

参考資料



RM0008

リファレンス・マニュアル

中容量及び大容量 STM32F101xx 及び STM32F103xx 高性能 ARM ベース 32bit MCU

はじめに

このリファレンス・マニュアルには、アプリケーション開発を行う開発者を対象として中容量及び大容量 STM32F101xx 及び STM32F103xx マイクロコントローラのメモリやペリフェラルについての詳細な情報が記載されています。特に記載がない限り、このマニュアルでは中容量及び大容量 STM32F101xx 及び STM32F103xx の総称として STM32F10xxx と記載されています。

STM32F10xxx 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ファミリ製品になります。

注文コード、外形寸法、電気特性の詳細については、中容量及び大容量 *STM32F101xx* 及び *STM32F103xx* データシートを参照してください。

内蔵されている Flash メモリの書き込み、消去、保護については、*STM32F10xxx Flash プログラミング・マニュアル*を参照してください。

ARM Cortex™-M3 コアの詳細な情報については、*Cortex™-M3 Technical Reference Manual*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 Technical Reference Manual*

次の資料が www.st.com から入手できます。

■ *STM32F101xx STM32F103xx データシート*

■ *STM32F10xxx Flash プログラミング・マニュアル*

参考資料

目次

RM0008

目次

1	このマニュアルにおける表記の規則	32
1.1	レジスタに関する略記	32
1.2	用語	32
2	メモリとバスの構成	33
2.1	システム・アーキテクチャ	33
2.2	メモリ構成	34
2.3	メモリ・マップ	35
2.3.1	ペリフェラルメモリ・マップ	36
2.3.2	内蔵 SRAM	38
2.3.3	ビット・バンディング	38
2.3.4	内蔵 Flash メモリ	39
2.4	ブート構成	42
3	CRC 計算ユニット	44
3.1	概要	44
3.2	主な機能	44
3.3	CRC 機能詳細	45
3.4	CRC レジスタ	45
3.4.1	データ・レジスタ (CRC_DR)	45
3.4.2	独立型データ・レジスタ (CRC_IDR)	46
3.4.3	制御レジスタ (CRC_CR)	46
3.4.4	CRC レジスタ・マップ	47
4	電源制御 (PWR)	48
4.1	電源供給	48
4.1.1	独立した A/D コンバータ電源と参照電圧	49
4.1.2	バッテリ・バックアップ・ドメイン	49
4.1.3	電圧レギュレータ	50
4.2	電源供給管理	50
4.2.1	パワー・オン・リセット (POR)/ パワー・ダウン・リセット (PDR)	50
4.2.2	プログラム可能な電圧検出回路 (PVD)	51
4.3	低電力モード	52



参考資料

4.3.1	クロック周波数の低減	52
4.3.2	ペリフェラル回路のクロック・スイッチ	53
4.3.3	SLEEP モード	53
4.3.4	STOP モード	55
4.3.5	STANDBY モード	56
4.3.6	低電力モードからの自動ウェークアップ (AWU : Auto-wakeup)	58
4.4	電源制御レジスタ	59
4.4.1	電源制御レジスタ (PWR_CR)	59
4.4.2	電源制御 / ステータス・レジスタ (PWR_CSR)	61
4.4.3	PWR レジスタ・マップ	62
5	バックアップ・レジスタ (BKP)	63
5.1	概要	63
5.2	主な機能	63
5.3	機能解説	64
5.3.1	タンパ検出	64
5.3.2	RTC 周波数調整	64
5.4	BKP レジスタ	65
5.4.1	バックアップ・データ・レジスタ x (BKP_DRx) (x = 1 ..42)	65
5.4.2	RTC クロック較正レジスタ (BKP_RTCCR)	66
5.4.3	バックアップ制御レジスタ (BKP_CR)	67
5.4.4	バックアップ制御 / ステータス・レジスタ (BKP_CSR)	68
5.4.5	BKP レジスタ・マップ	69
6	リセットとクロックの制御 (RCC)	72
6.1	リセット	72
6.1.1	システム・リセット	72
6.1.2	電源リセット	73
6.1.3	バックアップ・ドメイン・リセット	73
6.2	クロック	73
6.2.1	HSE クロック	75
6.2.2	HSI クロック	76
6.2.3	PLL 回路	76
6.2.4	LSE クロック	77
6.2.5	LSI クロック	77
6.2.6	システム・クロック (SYSCLK) の選択	78

参考資料

目次

RM0008

6.2.7	クロック・セキュリティシステム (CSS)	78
6.2.8	RTC クロック	79
6.2.9	ウォッチドッグ・クロック	79
6.2.10	クロック信号出力	79
6.3	RCC レジスタ	80
6.3.1	クロック制御レジスタ (RCC_CR)	80
6.3.2	クロック構成レジスタ (RCC_CFGR)	82
6.3.3	クロック割込みレジスタ (RCC_CIR)	85
6.3.4	APB2 ペリフェラル・リセット・レジスタ (RCC_APB2RSTR)	87
6.3.5	APB1 ペリフェラル・リセット・レジスタ (RCC_APB1RSTR)	89
6.3.6	AHB ペリフェラル・クロック・イネーブル・レジスタ (RCC_AHBENR)	91
6.3.7	APB2 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB2ENR)	93
6.3.8	APB1 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB1ENR)	95
6.3.9	バックアップ・ドメイン制御レジスタ (RCC_BDCR)	98
6.3.10	制御 / ステータス・レジスタ (RCC_CSR)	99
6.3.11	RCC レジスタ・マップ	101
7	汎用入出力とオルタネート機能入出力 (GPIO、AFIO)	102
7.1	GPIO 機能	102
7.1.1	汎用入出力 (GPIO)	105
7.1.2	アトミック・ビットのセット / リセット	105
7.1.3	外部割込み / ウェークアップ信号	105
7.1.4	オルタネート機能 (AF)	106
7.1.5	ソフトウェアによるオルタネート機能の入出力ピンの再割当て	106
7.1.6	GPIO ロック機構	106
7.1.7	入力回路の構成	107
7.1.8	出力回路の構成	108
7.1.9	オルタネート機能の構成	109
7.1.10	アナログ入力の構成	110
7.2	GPIO レジスタ	111
7.2.1	ポート構成レジスタ (下位) (GPIOx_CRL) (x=A..G)	111
7.2.2	ポート構成レジスタ (上位) (GPIOx_CRH) (x=A..G)	112
7.2.3	ポート入力データ・レジスタ (GPIOx_IDR) (x=A..G)	113
7.2.4	ポート出力データ・レジスタ (GPIOx_ODR) (x=A..G)	113
7.2.5	ポートビット・セット / リセット・レジスタ (GPIOx_BSRR) (x=A..G)	114

参考資料

7.2.6	ポートビット・リセット・レジスタ (GPIOx_BRR) (x=A..G)	114
7.2.7	ポート構成ロックレジスタ (GPIOx_LCKR) (x=A..G)	115
7.3	オルタネート入出力機能とデバッグ構成 (AFIO)	116
7.3.1	GPIO ポート PC14/PC15 を OSC32_IN/OSC32_OUT ピンとして 使用するには	116
7.3.2	GPIO ポート PD0/PD1 を OSC_IN/OSC_OUT ピンとして使用するには	116
7.3.3	CAN オルタネート機能のピン再割当て	116
7.3.4	JTAG/SWD オルタネート機能のピン割当て	117
7.3.5	ADC オルタネート機能のピン割当て	118
7.3.6	タイマ・オルタネート機能のピン割当て	119
7.3.7	USART オルタネート機能のピン割当て	121
7.3.8	I2C 回路 1 オルタネート機能のピン割当て	122
7.3.9	SPI 回路 1 オルタネート機能のピン割当て	122
7.4	AFIO レジスタ	123
7.4.1	イベント制御レジスタ (AFIO_EVCR)	123
7.4.2	オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR)	124
7.4.3	外部割込み構成レジスタ 1 (AFIO_EXTICR1)	127
7.4.4	外部割込み構成レジスタ 2 (AFIO_EXTICR2)	127
7.4.5	外部割込み構成レジスタ 3 (AFIO_EXTICR3)	128
7.4.6	外部割込み構成レジスタ 4 (AFIO_EXTICR4)	128
7.5	GPIO・AFIO レジスタ・マップ	129
8	割込みとイベント	130
8.1	ネスト化されたベクタ割込みコントローラ (NVIC)	130
8.1.1	SysTick 較正值レジスタ	130
8.1.2	割込み・例外ベクタ	130
8.2	外部割込み / イベントコントローラ (EXTI)	134
8.2.1	主な機能	134
8.2.2	ブロック図	134
8.2.3	ウェークアップ・イベント管理	135
8.2.4	機能詳細	135
8.2.5	外部割込み / イベントラインの割当て	136
8.3	EXTI レジスタ	137
8.3.1	割込みマスク・レジスタ (EXTI_IMR)	137
8.3.2	イベントマスク・レジスタ (EXTI_EMR)	137
8.3.3	立上りトリガ選択レジスタ (EXTI_RTSR)	138

参考資料

目次

RM0008

8.3.4	立下りトリガ選択レジスタ (EXTI_FTSR)	138
8.3.5	ソフトウェア割込みイベント・レジスタ (EXTI_SWIER)	139
8.3.6	ペンドィング・レジスタ (EXTI_PR)	139
8.3.7	EXTI レジスタ・マップ	140
9	DMA コントローラ (DMA)	141
9.1	概要	141
9.2	DMA の主な機能	141
9.3	DMA の機能詳細	142
9.3.1	DMA トランザクション	142
9.3.2	アービタ	143
9.3.3	DMA チャネル	143
9.3.4	エラー管理	144
9.3.5	割込み	145
9.3.6	DMA リクエストのマッピング	145
9.4	DMA レジスタ	149
9.4.1	DMA 割込みステータス・レジスタ (DMA_ISR)	149
9.4.2	DMA 割込みフラグ・クリア・レジスタ (DMA_IFCR)	150
9.4.3	DMA チャネル x 構成レジスタ (DMA_CCRx) (x = 1 ..7)	151
9.4.4	DMA チャネル x データ数レジスタ (DMA_CNDTRx) (x = 1 ..7)	152
9.4.5	DMA チャネル x ペリフェラル・アドレス・レジスタ (DMA_CPARx) (x = 1 ..7)	153
9.4.6	DMA チャネル x メモリ・アドレス・レジスタ (DMA_CMARx) (x = 1 ..7)	153
9.4.7	DMA レジスタ・マップ	153
10	A/D コンバータ (ADC)	156
10.1	概要	156
10.2	ADC の主な機能	156
10.3	ADC の機能詳細	157
10.3.1	ADC 電源の制御	158
10.3.2	ADC クロック	158
10.3.3	チャネル選択	158
10.3.4	シングル変換モード (Single conversion mode)	159
10.3.5	連続変換モード (Continuous conversion mode)	159
10.3.6	タイミング図	160
10.3.7	アナログ・ウォッチドッグ	160

参考資料

10.3.8	スキャン・モード (Scan mode)	161
10.3.9	インジェクト・チャネルの管理	161
10.3.10	分割モード (Discontinuous mode)	162
10.4	較正機能	163
10.5	データの配置	164
10.6	入力チャネルごとのサンプリングタイムの設定	165
10.7	外部トリガによる変換	165
10.8	DMA リクエスト	167
10.9	デュアル ADC モード	167
10.9.1	同時インジェクト・モード (Injected simultaneous mode)	169
10.9.2	同時レギュラ・モード (Regular simultaneous mode)	169
10.9.3	高速インターブ・モード (Fast interleaved mode)	170
10.9.4	低速インターブ・モード (Slow interleaved mode)	170
10.9.5	オルタネート・トリガ・モード (Alternate trigger mode)	171
10.9.6	独立モード (Independent mode)	172
10.9.7	同時レギュラ・モード + 同時インジェクト・モード	172
10.9.8	並列レギュラ・モード + オルタネート・トリガ・モード	173
10.9.9	同時インジェクト + インターブ	173
10.10	温度センサ	174
10.11	ADC 割込みリクエスト	175
10.12	ADC レジスタ	176
10.12.1	ADC ステータス・レジスタ (ADC_SR)	176
10.12.2	ADC 制御レジスタ 1 (ADC_CR1)	177
10.12.3	ADC 制御レジスタ 2 (ADC_CR2)	180
10.12.4	ADC サンプル・タイム・レジスタ 1 (ADC_SMPR1)	183
10.12.5	ADC サンプル・タイム・レジスタ 2 (ADC_SMPR2)	184
10.12.6	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)	184
10.12.7	ADC ウオッチドッグ上限スレッショルド・レジスタ (ADC_HTR)	185
10.12.8	ADC ウオッチドッグ下限スレッショルド・レジスタ (ADC_LTR)	185
10.12.9	ADC レギュラ・シーケンス・レジスタ 1 (ADC_SQR1)	185
10.12.10	ADC レギュラ・シーケンス・レジスタ 2 (ADC_SQR2)	186
10.12.11	ADC レギュラ・シーケンス・レジスタ 3 (ADC_SQR3)	187
10.12.12	ADC インジェクト・シーケンス・レジスタ (ADC_JSQR)	187
10.12.13	ADC インジェクト・データ・レジスタ x (ADC_JDRx) (x= 1..4)	188
10.12.14	ADC レギュラ・データ・レジスタ (ADC_DR)	189

参考資料

目次

RM0008

10.12.15 ADC レジスタ・マップ	189
11 D/A コンバータ (DAC)	191
11.1 概要	191
11.2 DAC の主な機能	191
11.3 DAC 機能詳細	193
11.3.1 DAC チャネル・イネーブル	193
11.3.2 DAC 出力バッファ・イネーブル	193
11.3.3 DAC データ・フォーマット	193
11.3.4 DAC 変換	194
11.3.5 DAC 出力電圧	195
11.3.6 DAC トリガ選択	195
11.3.7 DMA リクエスト	196
11.3.8 ノイズ生成	196
11.3.9 三角波生成	197
11.4 デュアル DAC チャネル変換	199
11.4.1 波形生成なしの独立トリガ	199
11.4.2 同じ LFSR 生成による独立トリガ	199
11.4.3 異なる LFSR 生成による独立トリガ	200
11.4.4 同じ三角波生成による独立トリガ	200
11.4.5 異なる三角波生成による独立トリガ	200
11.4.6 同時ソフトウェアスタート	201
11.4.7 波形生成なしによる同時トリガ	201
11.4.8 同じ LFSR 生成による同時トリガ	202
11.4.9 異なる LFSR 生成による同時トリガ	202
11.4.10 同じ三角生成による同時トリガ	202
11.4.11 異なる三角生成による同時トリガ	203
11.5 DAC レジスタ	204
11.5.1 DAC 制御レジスタ (DAC_CR)	204
11.5.2 DAC ソフトウェア・トリガ・レジスタ (DAC_SWTRIGR)	208
11.5.3 DAC チャネル 1 12bit 右寄せデータ保持レジスタ (DAC_DHR12R1)	208
11.5.4 DAC チャネル 1 12bit 左寄せデータ保持レジスタ (DAC_DHR12L1)	209
11.5.5 DAC チャネル 1 8bit 右寄せデータ保持レジスタ (DAC_DHR8R1)	209
11.5.6 DAC チャネル 2 12bit 右寄せデータ保持レジスタ (DAC_DHR12R2)	210



参考資料

11.5.7	DAC チャネル 2 12bit 左寄せデータ保持レジスタ (DAC_DHR12L2)	210
11.5.8	DAC チャネル 2 8bit 右寄せデータ保持レジスタ (DAC_DHR8R2)	211
11.5.9	デュアル DAC 12bit 右寄せデータ保持レジスタ (DAC_DHR12RD)	211
11.5.10	デュアル DAC 12bit 左寄せデータ保持レジスタ (DAC_DHR12LD)	212
11.5.11	デュアル DAC 8bit 右寄せデータ保持レジスタ (DAC_DHR8RD)	212
11.5.12	DAC チャネル 1 データ出力レジスタ (DAC_DOR1)	213
11.5.13	DAC チャネル 2 データ出力レジスタ (DAC_DOR2)	213
11.5.14	DAC レジスタ・マップ	214
12	高機能制御タイマ (TIM1 及び TIM8)	215
12.1	TIM1 及び TIM8 の概要	215
12.2	TIM1 及び TIM8 の主な機能	216
12.3	TIM1 及び TIM8 機能詳細	218
12.3.1	タイムベース・ユニット	218
12.3.2	カウンタ・モード	220
12.3.3	繰返しカウンタ (ダウンカウンタ)	229
12.3.4	クロック選択	231
12.3.5	キャプチャ / 比較チャネル	234
12.3.6	入力キャプチャ・モード	236
12.3.7	PWM 入力モード	237
12.3.8	出力強制駆動モード	238
12.3.9	出力比較モード	238
12.3.10	PWM モード	240
12.3.11	相補出力とデッドタイム挿入	243
12.3.12	ブレーク機能の利用	244
12.3.13	外部イベントによる OCxREF 信号のクリア	247
12.3.14	6 ステップ PWM 波形生成	248
12.3.15	単一パルス発生モード	249
12.3.16	エンコーダ・インターフェース・モード	250
12.3.17	タイマ入力 XOR 機能	253
12.3.18	ホール・センサ・インターフェース	253
12.3.19	TIMx と外部トリガ同期	255
12.3.20	タイマ間の同期	258

参考資料

目次

RM0008

12.3.21	デバッグ・モード	258
12.4	TIM1 及び TIM8 レジスタ	259
12.4.1	制御レジスタ 1 (TIMx_CR1)	259
12.4.2	制御レジスタ 2 (TIMx_CR2)	260
12.4.3	スレーブ・モード制御レジスタ (TIMx_SMCR)	262
12.4.4	DMA / 割込みリクエスト許可レジスタ (TIMx_DIER)	264
12.4.5	ステータス・レジスタ (TIMx_SR)	266
12.4.6	イベント生成レジスタ (TIMx_EGR)	267
12.4.7	キャプチャ / 比較モード・レジスタ 1 (TIMx_CCMR1)	269
12.4.8	キャプチャ / 比較モード・レジスタ 2 (TIMx_CCMR2)	272
12.4.9	キャプチャ / 比較イネーブル・レジスタ (TIMx_CCER)	273
12.4.10	カウンタ (TIMx_CNT)	276
12.4.11	プリスケーラ (TIMx_PSC)	276
12.4.12	自動再ロード・レジスタ (TIMx_ARR)	276
12.4.13	繰返しカウンタ・レジスタ (TIMx_RCR)	277
12.4.14	キャプチャ / 比較レジスタ 1 (TIMx_CCR1)	277
12.4.15	キャプチャ / 比較レジスタ 2 (TIMx_CCR2)	278
12.4.16	キャプチャ / 比較レジスタ 3 (TIMx_CCR3)	278
12.4.17	キャプチャ / 比較レジスタ 4 (TIMx_CCR4)	279
12.4.18	ブレーク / デッドタイムレジスタ (TIMx_BDTR)	279
12.4.19	DMA 制御レジスタ (TIMx_DCR)	281
12.4.20	バースト・モード DMA アドレス・レジスタ (TIMx_DMAR)	282
12.4.21	TIM1 及び TIM8 レジスタ・マップ	283
13	汎用タイマ (TIMx)	285
13.1	概要	285
13.2	TIMx の主な機能	285
13.3	TIMx の機能詳細	286
13.3.1	タイムベース・ユニット	286
13.3.2	カウンタ・モード	288
13.3.3	クロック選択	296
13.3.4	キャプチャ / 比較チャネル	299
13.3.5	入力キャプチャ・モード	301
13.3.6	PWM 入力モード	302
13.3.7	出力強制駆動モード	302
13.3.8	出力比較モード	303
13.3.9	PWM モード	304

参考資料

13.3.10	単一パルス発生モード	307
13.3.11	外部イベントによる OCxREF 信号のクリア	308
13.3.12	エンコーダ・インターフェース・モード	309
13.3.13	タイマ入力 XOR 機能	311
13.3.14	タイマの外部トリガ同期	311
13.3.15	タイマ間の同期	314
13.3.16	デバッグ・モード	320
13.4	TIMx レジスタ	321
13.4.1	制御レジスタ 1 (TIMx_CR1)	321
13.4.2	制御レジスタ 2 (TIMx_CR2)	322
13.4.3	スレーブ・モード制御レジスタ (TIMx_SMCR)	323
13.4.4	DMA/ 割込みイネーブル・レジスタ (TIMx_DIER)	326
13.4.5	ステータス・レジスタ (TIMx_SR)	327
13.4.6	イベント生成レジスタ (TIMx_EGR)	328
13.4.7	キャプチャ / 比較モード・レジスタ 1 (TIMx_CCMR1)	330
13.4.8	キャプチャ / 比較モード・レジスタ 2 (TIMx_CCMR2)	333
13.4.9	キャプチャ / 比較イネーブル・レジスタ (TIMx_CCER)	334
13.4.10	カウンタ (TIMx_CNT)	335
13.4.11	プリスケーラ (TIMx_PSC)	336
13.4.12	自動再ロード・レジスタ (TIMx_ARR)	336
13.4.13	キャプチャ / 比較レジスタ 1 (TIMx_CCR1)	336
13.4.14	キャプチャ / 比較レジスタ 2 (TIMx_CCR2)	337
13.4.15	キャプチャ / 比較レジスタ 3 (TIMx_CCR3)	337
13.4.16	キャプチャ / 比較レジスタ 4 (TIMx_CCR4)	338
13.4.17	DMA 制御レジスタ (TIMx_DCR)	338
13.4.18	バースト・モード DMA アドレス・レジスタ (TIMx_DMAR)	339
13.4.19	TIMx レジスタ・マップ	340
14	ベーシック・タイマ (TIM6 及び TIM7)	342
14.1	概要	342
14.2	TIM6 及び TIM7 の主な機能	342
14.3	TIM6 及び TIM7 の機能詳細	343
14.3.1	タイムベース・ユニット	343
14.3.2	カウンタ動作モード	345
14.3.3	クロック・ソース	347
14.3.4	デバッグ・モード	348
14.4	TIM6 及び TIM7 レジスタ	348

参考資料

目次

RM0008

14.4.1	制御レジスタ 1 (TIMx_CR1)	348
14.4.2	制御レジスタ 2 (TIMx_CR2)	350
14.4.3	DMA / 割込みリクエスト許可レジスタ (TIMx_DIER)	350
14.4.4	ステータス・レジスタ (TIMx_SR)	351
14.4.5	イベント生成レジスタ (TIMx_EGR)	351
14.4.6	カウンタ (TIMx_CNT)	352
14.4.7	プリスケーラ (TIMx_PSC)	352
14.4.8	自動再ロード・レジスタ (TIMx_ARR)	352
14.4.9	TIM6 及び TIM7 レジスタ・マップ	353
15	リアルタイム・クロック (RTC)	354
15.1	RTC の概要	354
15.2	RTC の主な機能	355
15.3	RTC の機能詳細	355
15.3.1	概要	355
15.3.2	RTC レジスタのリセット	356
15.3.3	RTC レジスタの読み出し	356
15.3.4	RTC レジスタの構成設定	357
15.3.5	RTC フラグセットの条件	358
15.4	RTC レジスタ	359
15.4.1	RTC 制御レジスタ (上位) (RTC_CRH)	359
15.4.2	RTC 制御レジスタ (下位) (RTC_CRL)	360
15.4.3	RTC プリスケーラ・ロード・レジスタ (RTC_PRLH / RTC_PRLL)	362
15.4.4	RTC プリスケーラ分周レジスタ (RTC_DIVH / RTC_DIVL)	363
15.4.5	RTC カウンタ・レジスタ (RTC_CNTH / RTC_CNTL)	364
15.4.6	RTC アラーム・レジスタ (上位) (RTC_ALRH / RTC_ALRL)	365
15.4.7	RTC レジスタ・マップ	366
16	独立型ウォッチドッグ (IWDG)	367
16.1	概要	367
16.2	IWDG の主な機能	367
16.3	IWDG の機能詳細	367
16.3.1	ハードウェアウォッチドッグ	367
16.3.2	レジスタ・アクセスに対する保護	368
16.3.3	デバッグ・モード	368
16.4	IWDG レジスタ	369

参考資料

16.4.1	キーレジスタ (IWDG_KR)	369
16.4.2	プリスケーラ・レジスタ (IWDG_PR)	369
16.4.3	再ロード値レジスタ (IWDG_RLR)	370
16.4.4	ステータス・レジスタ (IWDG_SR)	371
16.4.5	IWDG レジスタ・マップ	372
17	Window 型ウォッチドッグ (WWDG)	373
17.1	概要	373
17.2	WWDG の主な機能	373
17.3	WWDG の機能解説	374
17.4	ウォッチドッグによるタイムアウトを利用するには	376
17.5	デバッグ・モード	377
17.6	WWDG レジスタ	377
17.6.1	制御レジスタ (WWDG_CR)	377
17.6.2	構成レジスタ (WWDG_CFR)	378
17.6.3	ステータス・レジスタ (WWDG_SR)	378
17.6.4	WWDG レジスタ・マップ	379
18	フレキシブル・スタティック・メモリ・コントローラ (FSMC)	380
18.1	FSMC の主な機能	380
18.2	ブロック図	381
18.3	AHB インタフェース	382
18.3.1	サポートされるメモリ及びトランザクション	382
18.4	外部デバイス・アドレス・マッピング	383
18.4.1	NOR/PSRAM アドレス・マッピング	383
18.4.2	NAND/PC カード・アドレス・マッピング	384
18.5	NOR 型 Flash/PSRAM コントローラ	385
18.5.1	外部メモリ・インターフェース信号	386
18.5.2	サポートされるメモリ及びトランザクション	388
18.5.3	一般的なタイミング規則	389
18.5.4	NOR 型 Flash/PSRAM コントローラ・タイミング図	389
18.5.5	同期バースト読み込み	407
18.5.6	NOR/PSRAM 制御レジスタ	413
18.6	NAND 型 Flash/PC カード・コントローラ	419
18.6.1	外部メモリ・インターフェース信号	420

参考資料

目次

RM0008

18.6.2	NAND 型 Flash/PC カードがサポートされるメモリ及びトランザクション	421
18.6.3	NAND、ATA 及び PC カードに関するタイミング図	421
18.6.4	NAND 型 Flash レディ / ビジー管理	424
18.6.5	エラー訂正コード生成 ECC (NAND 型 Flash)	425
18.6.6	NAND 型 Flash/PC カード制御レジスタ	425
19	SDIO インタフェース (SDIO) 432	
19.1	SDIO の主な機能	432
19.2	SDIO バス・トポロジ	433
19.3	SDIO の機能詳細	435
19.3.1	SDIO アダプタ	436
19.3.2	SDIO AHB インタフェース	446
19.4	カード機能詳細	447
19.4.1	カード識別モード	447
19.4.2	カード・リセット	447
19.4.3	動作電圧範囲検証	447
19.4.4	カード識別プロセス	448
19.4.5	ブロック書き込み	449
19.4.6	ブロック読み出し	449
19.4.7	ストリーム・アクセス、ストリーム書き込み、ストリーム読み込み (MultiMediaCard のみ)	450
19.4.8	消去：グループ消去とセクタ消去	451
19.4.9	ワイドバス選択または非選択	451
19.4.10	保護管理	452
19.4.11	カード・ステータス・レジスタ	455
19.4.12	SD ステータス・レジスタ	458
19.4.13	SD I/O モード	463
19.4.14	コマンドとレスポンス	464
19.5	レスポンス・フォーマット	467
19.5.1	R1 (ノーマル・レスポンス・コマンド)	467
19.5.2	R1b	468
19.5.3	R2 (CID、CSD レジスタ)	468
19.5.4	R3 (OCR レジスタ)	468
19.5.5	R4 (高速 I/O)	469
19.5.6	R4b	469
19.5.7	R5 (割込みリクエスト)	470



参考資料

19.5.8	R6	470
19.6	SDIO I/O カード仕様動作	471
19.6.1	SDIO_D2 信号による SDIO I/O 読込みウェイト動作	471
19.6.2	SDIO_CK 停止による SDIO 読込みウェイト動作	471
19.6.3	SDIO サスPEND / リジューム動作	471
19.6.4	SDIO 割込み	472
19.7	CE-ATA 仕様動作	472
19.7.1	コマンド完了信号ディセーブル	472
19.7.2	コマンド完了信号イネーブル	472
19.7.3	CE-ATA 割込み	472
19.7.4	CMD61 のアボート	472
19.8	HW フロー制御	473
19.9	SDIO レジスタ	473
19.9.1	SDIO 電源制御レジスタ (SDIO_POWER)	473
19.9.2	SDIO クロック制御レジスタ (SDIO_CLKCR)	474
19.9.3	SDIO 引数レジスタ (SDIO_ARG)	475
19.9.4	SDIO コマンド・レジスタ (SDIO_CMD)	475
19.9.5	SDIO コマンド・レスポンス・レジスタ (SDIO_RESPCMD)	476
19.9.6	SDIO レスポンス 0..4 レジスタ (SDIO_RESPx)	477
19.9.7	SDIO データタイマレジスタ (SDIO_DTIMER)	477
19.9.8	SDIO データ長レジスタ (SDIO_DLEN)	478
19.9.9	SDIO データ制御レジスタ (SDIO_DCTRL)	478
19.9.10	SDIO データ・カウンタ・レジスタ (SDIO_DCOUNT)	480
19.9.11	SDIO ステータス・レジスタ (SDIO_STA)	480
19.9.12	SDIO 割込みクリア・レジスタ (SDIO_ICR)	482
19.9.13	SDIO マスク・レジスタ (SDIO_MASK)	483
19.9.14	SDIO FIFO カウンタ・レジスタ (SDIO_FIFOCNT)	486
19.9.15	SDIO データ FIFO レジスタ (SDIO_FIFO)	487
19.9.16	SDIO レジスタ・マップ	488
20	USB フルスピード・デバイス・インターフェース (USB)	489
20.1	概要	489
20.2	USB の主な機能	489
20.3	USB の機能詳細	490
20.3.1	USB ブロック	491
20.4	プログラミングの際の注意	493

参考資料

目次

RM0008

20.4.1	一般的な USB デバイスのプログラミング	493
20.4.2	システム・リセットとパワー・オン・リセット	493
20.4.3	ダブルバッファ・エンドポイント	499
20.4.4	アイソクロナス転送	501
20.4.5	サスPEND / レジュームイベント	502
20.5	USB レジスタ	504
20.5.1	一般レジスタ	504
20.5.2	エンドポイントに対応するレジスタ	511
20.5.3	バッファ・ディスクリプタ・テーブル	516
20.5.4	USB レジスタ・マップ	519
21	CAN (Controller Area Network) (bxCAN)	521
21.1	概要	521
21.2	bxCAN の主な機能	521
21.2.1	詳細	522
21.3	bxCAN の動作モード	524
21.3.1	初期化モード	524
21.3.2	通常モード	525
21.3.3	SLEEP モード (低消費電力)	525
21.3.4	テストモード	525
21.3.5	サイレント・モード	526
21.3.6	ループバック・モード	526
21.3.7	サイレント・モードとループバック・モードの組合わせ	527
21.4	bxCAN の機能詳細	527
21.4.1	送信動作	527
21.4.2	タイムトリガ通信モード	529
21.4.3	受信動作	529
21.4.4	ID によるフィルタリング	530
21.4.5	メッセージ・メモリ (メールボックス)	535
21.4.6	エラー管理	536
21.4.7	ビット・タイミング	537
21.5	bxCAN の割込み	538
21.6	CAN レジスタ	540
21.6.1	レジスタ・アクセスに対する保護	540
21.6.2	CAN 制御 / ステータス・レジスタ	541
21.6.3	メールボックス・レジスタ	553



参考資料

21.6.4	CAN フィルタ・レジスタ	560
21.6.5	bxCAN レジスタ・マップ	564
22	SPI (Serial peripheral interface)	567
22.1	SPI の概要	567
22.2	SPI と I ² S の主な機能	568
22.2.1	SPI 機能	568
22.2.2	I ² S 機能	569
22.3	SPI の機能詳細	570
22.3.1	概説	570
22.3.2	SPI スレーブ・モード	574
22.3.3	SPI マスタ・モード	575
22.3.4	一方向の通信	575
22.3.5	ステータス・フラグ	576
22.3.6	CRC の計算	577
22.3.7	DMA (direct memory addressing) を使用する SPI 通信	578
22.3.8	エラー・フラグ	578
22.3.9	SPI のディセーブル	579
22.3.10	SPI 割込み	579
22.4	I ² S 機能詳細	579
22.4.1	詳細	579
22.4.2	サポートされるオーディオ・プロトコル	581
22.4.3	クロック・ジェネレータ	588
22.4.4	I ² S マスタ・モード	589
22.4.5	I ² S スレーブ・モード	590
22.4.6	ステータス・フラグ	592
22.4.7	エラー・フラグ	593
22.4.8	I ² S 割込み	593
22.4.9	DMA 機能	593
22.5	SPI と I ² S レジスタ	594
22.5.1	SPI 制御レジスタ 1 (SPI_CR1) (I ² S モードでは使用しません。)	594
22.5.2	SPI 制御レジスタ 2 (SPI_CR2)	596
22.5.3	SPI ステータス・レジスタ (SPI_SR)	596
22.5.4	SPI データ・レジスタ (SPI_DR)	598
22.5.5	SPI CRC 多項式レジスタ (SPI_CRCPR) (I ² S モードでは使用しません。)	598
22.5.6	SPI Rx CRC レジスタ (SPI_RXCRCR) (I ² S モードでは使用しません。)	599

参考資料

目次

RM0008

22.5.7	SPI Tx CRC レジスタ (SPI_TXCRCR) (I ² S モードでは使用しません。)	599
22.5.8	SPI_I ² S 構成レジスタ (SPI_I2SCFGR)	600
22.5.9	SPI_I ² S プリスケーラ・レジスタ (SPI_I2SPR)	601
22.5.10	SPI レジスタ・マップ	602
23	I²C (Inter-integrated circuit) インタフェース	603
23.1	概要	603
23.2	I ² C の主な機能	604
23.3	I ² C の機能詳細	605
23.3.1	モード選択	605
23.3.2	I ² C スレーブ・モード	606
23.3.3	I ² C マスター・モード	609
23.3.4	エラー条件	613
23.3.5	SDA/SCL ライン制御	614
23.3.6	SMBus	615
23.3.7	DMA リクエスト	618
23.3.8	パケット・エラーの検出	620
23.4	I ² C 割込み	621
23.5	I ² C デバッグ・モード	623
23.6	I ² C レジスタ	623
23.6.1	制御レジスタ 1 (I2C_CR1)	623
23.6.2	制御レジスタ 2 (I2C_CR2)	625
23.6.3	自己アドレス・レジスタ 1 (I2C_OAR1)	627
23.6.4	自己アドレス・レジスタ 2 (I2C_OAR2)	627
23.6.5	データ・レジスタ (I2C_DR)	628
23.6.6	ステータス・レジスタ 1 (I2C_SR1)	629
23.6.7	ステータス・レジスタ 2 (I2C_SR2)	632
23.6.8	クロック制御レジスタ (I2C_CCR)	634
23.6.9	TRISE レジスタ (I2C_TRISE)	635
23.6.10	I ² C レジスタ・マップ	636
24	USART	637
24.1	概要	637
24.2	USART の主な機能	638
24.3	USART の機能詳細	639
24.3.1	USART の特性詳細	642

参考資料

24.3.2	送信回路	643
24.3.3	受信回路	645
24.3.4	ボーレート生成回路	649
24.3.5	複数プロセッサ間の通信	651
24.3.6	パリティ制御	652
24.3.7	LIN (local interconnection network) モード	653
24.3.8	USART 同期シリアル・モード	655
24.3.9	単線式半二重通信	657
24.3.10	スマートカード	658
24.3.11	IrDA SIR ENDEC ブロック	660
24.3.12	DMA を使用する連続的な通信	662
24.3.13	ハードウェア・フロー制御	663
24.4	USART 割込み	665
24.5	USART モード構成	666
24.6	USART レジスタ	666
24.6.1	ステータス・レジスタ (USART_SR)	666
24.6.2	データ・レジスタ (USART_DR)	669
24.6.3	ボーレート・レジスタ (USART_BRR)	670
24.6.4	制御レジスタ 1 (USART_CR1)	670
24.6.5	制御レジスタ 2 (USART_CR2)	673
24.6.6	制御レジスタ 3 (USART_CR3)	675
24.6.7	ガードタイム / プリスケーラ・レジスタ (USART_GTPR)	676
24.6.8	USART レジスタ・マップ	678
25	デバイス電子署名	679
25.1	メモリ・サイズ・レジスタ	679
25.1.1	Flash サイズ・レジスタ	679
25.1.2	RAM サイズ・レジスタ	679
25.2	ユニーク・デバイス ID レジスタ (96bit)	680
26	デバッグ・サポート (DBG)	682
26.1	概要	682
26.2	ARM ドキュメントの参照	684
26.3	SWJ デバッグ・ポート (シリアル・ワイヤと JTAG)	684
26.3.1	JTAG-DP もしくは SW-DP 選択メカニズム	685
26.4	ピン配置とデバッグ・ポートピン	685

参考資料

目次

RM0008

26.4.1	SWJ デバッグ・ポートピン	685
26.4.2	フレキシブルな SWJ-DP ピン配置	685
26.4.3	JTAG ピンの内部プルアップ / プルダウン	686
26.4.4	GPIO としてもシリアル・ワイヤの使用と未使用デバッグピンの解放	687
26.5	STM32F10xxx JTAG TAP 接続	688
26.6	ID コードとロックメカニズム	689
26.6.1	MCU デバイス ID コード	689
26.6.2	バウンダリ・スキャン TAP	690
26.6.3	Cortex-M3 TAP	690
26.6.4	Cortex-M3 JEDEC-106 ID コード	690
26.7	JTAG デバッグ・ポート	690
26.8	SW デバッグ・ポート	692
26.8.1	SW プロトコル概要	692
26.8.2	SW プロトコル手順	692
26.8.3	SW-DP ステートマシン（リセット、アイドルステート、ID コード）	693
26.8.4	DP 及び AP の読み出し / 書込みアクセス	694
26.8.5	SW-DP レジスタ	695
26.8.6	SW-AP レジスタ	696
26.9	AHB-AP (AHB アクセスポート) - JTAG-DP もしくは SW-DP で有効	696
26.10	コア・デバッグ	697
26.11	システム・リセット時のデバッグ・ホスト接続	697
26.12	FPB (Flash patch breakpoint : Flash パッチ・ブレークポイント)	698
26.13	DWT (data watchpoint trigger : データ・ウォッチポイント・トリガ)	698
26.14	ITM (instrumentation trace macrocell : 計測トレース・マクロセル)	699
26.14.1	概要	699
26.14.2	タイムスタンプ・パケット、同期及びオーバフロー・パケット	699
26.15	MCU デバッグ・コンポーネント (MCUDBG)	701
26.15.1	低電力モードにおけるデバッグのサポート	701
26.15.2	タイマ、ウォッチドッグ、bxCAN と I ² C に対する デバッグ・サポート	701
26.15.3	デバッグ MCU 構成レジスタ	702
26.16	TPIU (trace port interface unit)	704
26.16.1	概要	704
26.16.2	トレース・ピン配置	705
26.16.3	TPIU フォーマッタ	707
26.16.4	TPIU フレーム同期パケット	707



参考資料

26.16.5	同期フレームパケットの発行	708
26.16.6	同期モード	708
26.16.7	非同期モード	708
26.16.8	STM32F10xxx 内部の TRACECLKIN 接続	708
26.16.9	TPIU レジスタ	709
26.16.10	設定例	710
26.17	DBG レジスタ・マップ	711
27	改版履歴	712

表の一覧

表 1.	レジスタ・バウンダリ・アドレス	36
表 2.	Flash モジュールの構成（中容量デバイス）.....	39
表 3.	Flash モジュールの構成（大容量デバイス）.....	40
表 4.	ブート・モード	42
表 5.	CRC 計算ユニット・レジスタ・マップとリセット値	47
表 6.	低電力モード要約	52
表 7.	Sleep-now	54
表 8.	Sleep-on-exit.....	54
表 9.	STOP モード	56
表 10.	STANDBY モード	57
表 11.	PWR - レジスタ・マップとリセット時の値	62
表 12.	BKP - レジスタ・マップとリセット時の値.....	69
表 13.	RCC - レジスタ・マップとリセット時の値.....	101
表 14.	ポート・ビット構成表	104
表 15.	出力モードビット	104
表 16.	CAN オルタネート機能のピン割当て変更	116
表 17.	デバッグ・インターフェース信号	117
表 18.	デバッグ・ポートのピン割当て	117
表 19.	ADC1 外部トリガ・インジェクト変換オルタネート機能のピン割当て	118
表 20.	ADC1 外部トリガ・レギュラ変換オルタネート機能のピン割当て	118
表 21.	ADC2 外部トリガ・インジェクト変換オルタネート機能のピン割当て	118
表 22.	ADC2 外部トリガ・レギュラ変換オルタネート機能のピン割当て	118
表 23.	タイマ 5 オルタネート機能のピン割当て	119
表 24.	タイマ 4 オルタネート機能のピン割当て	119
表 25.	タイマ 3 オルタネート機能のピン割当て	119
表 26.	タイマ 2 オルタネート機能のピン割当て	119
表 27.	タイマ 1 オルタネート機能のピン割当て	120
表 28.	USART3 ピン割当て	121
表 29.	USART2 ピン割当て	121
表 30.	USART1 ピン割当て	121
表 31.	I2C1 ピン割当て	122
表 32.	SPI1 ピン割当て	122
表 33.	GPIO - レジスタ・マップとリセット時の値	129
表 34.	AFIO - レジスタ・マップとリセット時の値	129
表 35.	ベクターテーブル	130
表 36.	外部割込み / イベントコントローラ (EXTI) レジスタ・マップとリセット時の値	140
表 37.	DMA 割込みリクエスト	145
表 38.	チャネルに対する DMA1 リクエストの一覧	147
表 39.	チャネルに対する DMA2 リクエストの一覧	148
表 40.	DMA - レジスタ・マップとリセット時の値	153
表 41.	ADC ピン	158
表 42.	アナログ・ウォッチドッグ・チャネル選択	161
表 43.	ADC1 と ADC2 のレギュラ・チャネルのための外部トリガ	165
表 44.	ADC1 と ADC2 のインジェクト・チャネルのための外部トリガ	166
表 45.	ADC3 のレギュラ・チャネルのための外部トリガ	166
表 46.	ADC3 のインジェクト・チャネルのための外部トリガ	166
表 47.	ADC 割込み	175
表 48.	ADC - レジスタ・マップとリセット時の値	189

参考資料

表 49.	DAC ピン	192
表 50.	外部トリガ	195
表 51.	DAC レジスタ・マップ	214
表 52.	カウントの方向とエンコーダ信号	251
表 53.	TIMx 内部トリガ接続	264
表 54.	ブレーク機能をもつ相補出力 OC _x 及び OC _{xN} チャネルの出力制御ビット	275
表 55.	TIM1 及び TIM8 レジスタ・マップとリセット時の値	283
表 56.	カウントの方向とエンコーダ信号	310
表 57.	TIMx 内部トリガ接続	325
表 58.	標準 OC _x チャネルの出力制御ビット	335
表 59.	TIMx レジスタ・マップとリセット時の値	340
表 60.	TIM6 及び TIM7 レジスタ・マップとリセット時の値	353
表 61.	RTC レジスタ・マップとリセット時の値	366
表 62.	ウォッチドッグ・タイムアウト時間 (40 kHz クロック入力時)	368
表 63.	IWDG レジスタ・マップとリセット時の値	372
表 64.	WWDG レジスタ・マップとリセット時の値	379
表 65.	NOR/PSRAM バンク選択	383
表 66.	外部メモリ・アドレス	384
表 67.	メモリ・マッピングとタイミング・レジスタ	384
表 68.	NAND バンク選択	385
表 69.	プログラマブル NOR/PSRAM アクセス・パラメータ	386
表 70.	非マルチプレクス I/O NOR 型 Flash	386
表 71.	マルチプレクス I/O NOR 型 Flash	387
表 72.	PSRAM	387
表 73.	NOR 型 Flash /PSRAM がサポートするメモリ及びトランザクション	388
表 74.	FSMC_BCRx ビット・フィールド	391
表 75.	FSMC_TCRx ビット・フィールド	392
表 76.	FSMC_BCRx ビット・フィールド	394
表 77.	FSMC_TCRx ビット・フィールド	394
表 78.	FSMC_BWTRx ビット・フィールド	394
表 79.	FSMC_BCRx ビット・フィールド	398
表 80.	FSMC_TCRx ビット・フィールド	398
表 81.	FSMC_BWTRx ビット・フィールド	398
表 82.	FSMC_BCRx ビット・フィールド	400
表 83.	FSMC_TCRx ビット・フィールド	401
表 84.	FSMC_BWTRx ビット・フィールド	401
表 85.	FSMC_BCRx ビット・フィールド	403
表 86.	FSMC_TCRx ビット・フィールド	404
表 87.	FSMC_BWTRx ビット・フィールド	404
表 88.	FSMC_BCRx ビット・フィールド	406
表 89.	FSMC_TCRx ビット・フィールド	407
表 90.	FSMC_BCRx ビット・フィールド	410
表 91.	FSMC_TCRx ビット・フィールド	410
表 92.	FSMC_BCRx ビット・フィールド	412
表 93.	FSMC_TCRx ビット・フィールド	412
表 94.	プログラマブル NAND/PC カード・アクセス・パラメータ	419
表 95.	8bit NAND 型 Flash	420
表 96.	16bit NAND 型 Flash	420
表 97.	16bit PC カード	420
表 98.	サポートされるメモリ及びトランザクション	421
表 99.	ECC リザルト関連ビット	431
表 100.	SDIO I/O 定義	436

参考資料

表の一覧

RM0008

表 101.	コマンド・フォーマット	440
表 102.	ショート・レスポンス・フォーマット	441
表 103.	ロング・レスポンス・フォーマット	441
表 104.	コマンド・パス・ステータス・フラグ	441
表 105.	データ・トークン・フォーマット	444
表 106.	送信 FIFO ステータス・フラグ	445
表 107.	受信 FIFO ステータス・フラグ	446
表 108.	カード・ステータス	456
表 109.	SD ステータス	459
表 110.	速度クラス・フィールド	460
表 111.	パフォーマンス・ムーブ・フィールド	460
表 112.	AU_SIZE フィールド	461
表 113.	最大 AU サイズ	461
表 114.	消去サイズ・フィールド	461
表 115.	消去タイムアウト・フィールド	462
表 116.	消去オフセット・フィールド	462
表 117.	ブロック重視書き込みコマンド	465
表 118.	ブロック重視書き込み保護コマンド	465
表 119.	消去コマンド	466
表 120.	I/O モード・コマンド	466
表 121.	ロック・カード	466
表 122.	アプリケーション固有コマンド	467
表 123.	R1 レスpons	467
表 124.	R2 レスpons	468
表 125.	R3 レスpons	468
表 126.	R4 レスpons	469
表 127.	R4b レスpons	469
表 128.	R5 レスpons	470
表 129.	R6 レスpons	470
表 130.	レスponsタイプと SDIO_RESPx レジスタ	477
表 131.	SDIO -レジスタ・マップ	488
表 132.	ダブルバッファリング・バッファ・フラグ定義	499
表 133.	バルク・ダブルバッファリング・メモリ・バッファ	500
表 134.	アイソクロナス・メモリ・バッファ	501
表 135.	レジューム・イベント検出	503
表 136.	受信ステータス・エンコーディング	515
表 137.	エンドポイント・タイプ・エンコーディング	515
表 138.	EP_KIND ビットの意味	515
表 139.	送信ステータス・エンコーディング	515
表 140.	アロケートされたバッファ・メモリ・サイズの定義	519
表 141.	USB -レジスタ・マップとリセット時の値	519
表 142.	送信メールボックスのマッピング	535
表 143.	受信メールボックスのマッピング	535
表 144.	bxCAN -レジスタ・マップとリセット時の値	564
表 145.	SPI 割込みリクエスト	579
表 146.	I ² S 割込みリクエスト	593
表 147.	SPI -レジスタ・マップとリセット時の値	602
表 148.	SMBus -I ² C 比較	615
表 149.	I ² C 割込みリクエスト	621
表 150.	I ² C -レジスタ・マップとリセット時の値	636
表 151.	サンプルデータからのノイズ検出	647
表 152.	ボーレート設定と周波数の誤差	650



参考資料

表 153.	フレーム・フォーマット	652
表 154.	USART 割込みリクエスト	665
表 155.	USART モード設定	666
表 156.	USART レジスタ・マップとリセット時の値	678
表 157.	SWJ デバッグ・ポートピン	685
表 158.	融通性の高い SWJ-DP ピンの割り当て	686
表 159.	JTAG デバッグ・ポート・データ・レジスタ	691
表 160.	A[3:2] で選択される 32bit デバッグ・ポート・レジスタ	692
表 161.	パケット・リクエスト (8bit)	693
表 162.	ACK 応答 (3bit)	693
表 163.	DATA 転送 (33bit)	693
表 164.	SW-DP レジスタ	695
表 165.	Cortex-M3 AHB-AP レジスタ	696
表 166.	コア・デバッグ・レジスタ	697
表 167.	主な ITM レジスタ	700
表 168.	非同期トレース・ピン割り当て	705
表 169.	同期トレース・ピン割り当て	705
表 170.	融通性の高いトレース・ピン割り当て	706
表 171.	重要な TPIU レジスタ	709
表 172.	DBG - レジスタ・マップとリセット時の値	711
表 173.	改版履歴	712

参考資料

図の一覧

RM0008

図の一覧

図 1.	システム・アーキテクチャ	33
図 2.	メモリ・マップ	35
図 3.	CRC 計算ユニット・ブロック図	44
図 4.	電源系の概要	48
図 5.	パワー・オン・リセット / パワー・ダウン・リセット波形	50
図 6.	PVD スレッショルド	51
図 7.	リセット回路	73
図 8.	クロック構成図	74
図 9.	HSE / LSE クロック・ソース	75
図 10.	入出力ポートの基本構造	103
図 11.	5V 耐性入出力ポートの基本構造	103
図 12.	入力回路の構造 (フローティング / プルアップ / プルダウン)	107
図 13.	出力回路の構造	108
図 14.	オルタネート機能時のポートの構成	109
図 15.	ハイ・インピーダンス・アナログ入力の構成	110
図 16.	外部割込み / イベントコントローラ (EXTI) ブロック図	134
図 17.	外部割込み / イベントの GPIO 割当て	136
図 18.	DMA ブロック構成	142
図 19.	DMA1 リクエストのマッピング	146
図 20.	DMA2 リクエストのマッピング	148
図 21.	シングル ADC ブロック図	157
図 22.	タイミング図	160
図 23.	アナログ・ウォッチドッグの保護ドメイン	160
図 24.	インジェクト変換の遅延時間	162
図 25.	較正動作のタイミング図	164
図 26.	データの右寄せ	164
図 27.	データの左寄せ	164
図 28.	デュアル ADC ブロック図	168
図 29.	4 チャネルの同時インジェクト・モード	169
図 30.	16 チャネルの同時レギュラ変換モード	170
図 31.	連続変換モードにおける 1 チャネルの高速インタリーブ・モード	170
図 32.	1 チャネルの低速インタリーブ・モード	171
図 33.	オルタネート・トリガ : 各 ADC のインジェクト・チャネル・グループ	172
図 34.	オルタネート・トリガ : 分割モデルの (各 ADC ごとの) 4 インジェクト・チャネル	172
図 35.	オルタネート + 同時レギュラ	173
図 36.	インジェクト変換中にトリガが発生した場合	173
図 37.	インジェクト・シーケンス CH11、CH12 でのインタリーブ・シングル・チャネル	174
図 38.	温度センサ及び VREFINT チャネル・ブロック図	174
図 39.	DAC チャネル・ブロック図	192
図 40.	シングル DAC チャネル・モードのデータ・レジスタ	194
図 41.	デュアル DAC チャネル・モードのデータ・レジスタ	194
図 42.	トリガ・ディセーブル時 ($TEN = 0$) の変換タイミング図	195
図 43.	DAC LFSR レジスタ計算アルゴリズム	196
図 44.	LFSR 波形生成による DAC 変換 (SW トリガ・イネーブル)	197
図 45.	DAC 三角波生成	197
図 46.	三角波生成による DAC 変換 (SW トリガ・イネーブル)	198
図 47.	高機能制御タイマ・ブロック図	217
図 48.	プリスケーラ分周比が 1 から 2 に変化した時のカウンタ・タイミング図	219

参考資料

図 49.	プリスケーラ分周比が 1 から 4 に変化した時のカウンタ・タイミング図	219
図 50.	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	220
図 51.	内部クロック分周比が 2 の場合のカウンタ・タイミング図	221
図 52.	内部クロック分周比が 4 の場合のカウンタ・タイミング図	221
図 53.	内部クロック分周比が N の場合のカウンタ・タイミング図	221
図 54.	ARPE=0 時 (TIMx_ARR はプリロードなし)、更新イベント時のタイミング図	222
図 55.	ARPE=1 時 (TIMx_ARR はプリロード)、更新イベントのタイミング図	222
図 56.	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	224
図 57.	内部クロック分周比が 2 の場合のカウンタ・タイミング図	224
図 58.	内部クロック分周比が 4 の場合のカウンタ・タイミング図	224
図 59.	内部クロック分周比が N の場合のカウンタ・タイミング図	225
図 60.	繰返しカウンタが使用されないときの更新イベント時のタイミング図	225
図 61.	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 (TIMx_ARR=0x6)	227
図 62.	内部クロック分周比が 2 の場合のカウンタ・タイミング図	227
図 63.	内部クロック分周比が 4 の場合のカウンタ・タイミング図 (TIMx_ARR=0x36)	227
図 64.	内部クロック分周比が N の場合のカウンタ・タイミング図	228
図 65.	ARPE=1 時の更新イベントのタイミング図 (カウンタはアンダフロー)	228
図 66.	ARPE=1 時の更新イベントのタイミング図 (カウンタはオーバフロー)	229
図 67.	カウンタモードと TIMx_RCR レジスタの設定による更新状況の例	230
図 68.	内部クロック分周比 1 の場合の通常モードの制御回路	231
図 69.	TI2 外部クロックの接続例	232
図 70.	外部クロック・モード 1 の制御回路	232
図 71.	外部トリガ入力ブロック	233
図 72.	外部クロック・モード 2 の制御回路	233
図 73.	キャプチャ / 比較チャネル (例: チャネル 1 入力ステージ)	234
図 74.	キャプチャ / 比較チャネル 1 メイン回路	234
図 75.	キャプチャ / 比較チャネル (チャネル 1 - 3) の出力ステージ	235
図 76.	キャプチャ / 比較チャネル (チャネル 4) の出力ステージ	235
図 77.	PWM 入力モードのタイミング	237
図 78.	出力比較モード、OC1 出力のトグル	239
図 79.	エッジアライン PWM 波形 (ARR=8)	241
図 80.	センターアライン PWM 波形 (ARR=8)	242
図 81.	デッドタイム挿入のある相補出力	243
図 82.	負のパルスより長い遅延があるときのデッドタイムの波形	243
図 83.	正のパルスより長い遅延があるときのデッドタイムの波形	244
図 84.	ブレークに対する出力の動作	246
図 85.	OCxREF のクリア動作	247
図 86.	6 ステップ PWM 波形の生成と CO 例 (OSSR=1)	248
図 87.	単一パルス発生モードの例	249
図 88.	エンコーダ・インターフェース・モードにおけるカウンタの動作例	252
図 89.	TI1FP1 の極性を反転したエンコーダ・インターフェース・モードの例	252
図 90.	ホール・センサ・インターフェースの例	254
図 91.	リセット・モードにおける制御回路	255
図 92.	ゲート・モードの制御回路	256
図 93.	トリガ・モードの制御回路	257
図 94.	外部クロック・モード 2 + トリガ・モードの制御回路	258
図 95.	汎用タイマのブロック図	286
図 96.	プリスケーラ分周比が 1 から 2 に変化した時のカウンタ・タイミング図	287
図 97.	プリスケーラ分周比が 1 から 4 に変化した時のカウンタ・タイミング図	288
図 98.	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	289
図 99.	内部クロック分周比が 2 の場合のカウンタ・タイミング図	289
図 100.	内部クロック分周比が 4 の場合のカウンタ・タイミング図	289

参考資料

図の一覧

RM0008

図 101.	内部クロック分周比が N の場合のカウンタ・タイミング図	290
図 102.	ARPE=0 (TIMx_ARR はプリロードなし) の時の更新イベントのカウンタ・タイミング図	290
図 103.	ARPE=1 (TIMx_ARR はプリロード) の時の更新イベントのカウンタ・タイミング図	291
図 104.	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	292
図 105.	内部クロック分周比が 2 の場合のカウンタ・タイミング図	292
図 106.	内部クロック分周比が 4 の場合のカウンタ・タイミング図	292
図 107.	内部クロック分周比が N の場合のカウンタ・タイミング図	293
図 108.	繰り返しカウンタが使用されない時の更新イベントのカウンタ・タイミング図	293
図 109.	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 (TIMx_ARR=0x6)	294
図 110.	内部クロック分周比が 2 の場合のカウンタ・タイミング図	294
図 111.	ロード・レジスタ内部クロック分周比が 4 の場合のカウンタ・タイミング図 (TIMx_ARR=0x36)	295
図 112.	内部クロック分周比が N の場合のカウンタ・タイミング図	295
図 113.	ARPE=1 (カウンタはアンダフロー) の時の更新イベントのカウンタ・タイミング図	295
図 114.	ARPE=1 (カウンタはオーバフロー) の時の更新イベントのカウンタ・タイミング図	296
図 115.	内部クロック分周比が 1 の場合の、通常モードの制御回路	297
図 116.	TI2 外部クロックの接続例	297
図 117.	外部クロック・モード 1 の制御回路	298
図 118.	外部トリガ入力ブロック	298
図 119.	外部クロック・モード 2 の制御回路	299
図 120.	キャプチャ / 比較チャネル (例: チャネル 1 入力ステージ)	299
図 121.	キャプチャ / 比較チャネル 1 メイン回路	300
図 122.	キャプチャ / 比較チャネル (チャネル 1) の出力ステージ	300
図 123.	PWM 入力モードのタイミング	302
図 124.	出力比較モード、OC1 出力のトグル	304
図 125.	エッジアライン PWM 波形 (ARR=8)	305
図 126.	センターアライン PWM 波形 (ARR=8)	306
図 127.	単一パルス発生モード	307
図 128.	TIMx OCxREF のクリア動作	309
図 129.	エンコーダ・インターフェース・モードにおけるカウンタの動作例	310
図 130.	IC1FP1 の極性を反転したエンコーダ・インターフェース・モードの例	311
図 131.	リセット・モードの制御回路	312
図 132.	ゲート・モードの制御回路	313
図 133.	トリガ・モードの制御回路	313
図 134.	外部クロック・モード 2 + トリガ・モードの制御回路	314
図 135.	マスター / スレーブ タイマの例	315
図 136.	タイマ 1 の OC1REF によるタイマ 2 のゲート操作	316
図 137.	タイマ 1 のイネーブルによるタイマ 2 のゲート制御	317
図 138.	タイマ 1 のアップデートによるタイマ 2 のトリガ	318
図 139.	タイマ 1 のイネーブルによるタイマ 2 のトリガ	319
図 140.	タイマ 1 の TI1 入力によるタイマ 1 とタイマ 2 のトリガ	320
図 141.	基本タイマ・ブロック図	343
図 142.	プリスケーラ分周比が 1 から 2 に変化した時のカウンタ・タイミング図	344
図 143.	プリスケーラ分周比が 1 から 4 に変化した時のカウンタ・タイミング図	344
図 144.	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	345
図 145.	内部クロック分周比が 2 の場合のカウンタ・タイミング図	346
図 146.	内部クロック分周比が 4 の場合のカウンタ・タイミング図	346
図 147.	内部クロック分周比が N の場合のカウンタ・タイミング図	346
図 148.	ARPE=0 時 (TIMx_ARR はプリロードなし)、更新イベント時のタイミング図	347
図 149.	ARPE=1 時 (TIMx_ARR はプリロード)、更新イベントのタイミング図	347
図 150.	内部クロック分周比 1 の場合の、通常モードの制御回路	348
図 151.	RTC ブロック構成	356



参考資料

図 152.	RTC 秒及びアラーム波形 (PR=0003、ALARM=00004)STANDBY モード	358
図 153.	RTC オーバフロー波形 (PR=0003)	358
図 154.	独立型ウォッチドッグのブロック図	368
図 155.	ウォッチドッグブロック図	374
図 156.	Window 型ウォッチドッグのタイミング	376
図 157.	FSMC ブロック図	381
図 158.	FSMC メモリ・バンク	383
図 159.	モード 1 読込みアクセス	390
図 160.	モード 1 書込みアクセス	391
図 161.	モード A 読込みアクセス	392
図 162.	モード A 書込みアクセス	393
図 163.	モード 2/B 読込みアクセス	395
図 164.	モード 2 書込みアクセス	396
図 165.	モード B 書込みアクセス	397
図 166.	モード C 読込みアクセス	399
図 167.	モード C 書込みアクセス	400
図 168.	モード D 読込みアクセス	402
図 169.	モード D 書込みアクセス	403
図 170.	マルチプレクスした読み込みアクセス	405
図 171.	マルチプレクスした書き込みアクセス	406
図 172.	同期読み込みモード – NOR、CRAM	409
図 173.	同期書き込みモード – CRAM	411
図 174.	コモン・メモリ・アクセス用 NAND コントローラのタイミング	422
図 175.	‘CE don’t care’ でない NAND 型 Flash へのアクセス	424
図 176.	SDIO “no response” と “no data” 動作	433
図 177.	SDIO (複数) ブロック読み込み動作	433
図 178.	SDIO (複数) ブロック書き込み動作	434
図 179.	SDIO 連続読み込み動作	434
図 180.	SDIO 連続書き込み動作	434
図 181.	SDIO ブロック図	435
図 182.	SDIO アダプタ	436
図 183.	制御ユニット	437
図 184.	SDIO アダプタ・コマンド・パス	438
図 185.	コマンド・パス・ステート・マシン (CPSM)	439
図 186.	SDIO コマンド転送	440
図 187.	データ・パス	442
図 188.	データ・パス・ステート・マシン (DPSM)	443
図 189.	USB ペリフェラル・ブロック図	490
図 190.	バッファ・ディスクリプション・テーブル・ロケーションの パケット・バッファ・ドメイン	495
図 191.	CAN ネットワークのトポロジ	522
図 192.	CAN ブロック図	523
図 193.	bxCAN 動作モード	524
図 194.	bxCAN サイレント・モード	526
図 195.	bxCAN ループバック・モード	526
図 196.	bxCAN 組合せモード	527
図 197.	送信メールボックスのステート	528
図 198.	受信 FIFO のステート	529
図 199.	フィルタ・バンクのスケール設定 - レジスタ構成	532
図 200.	フィルタ番号の例	533
図 201.	フィルタのメカニズム例	534
図 202.	CAN エラー状態図	536

参考資料

図の一覧

RM0008

図 203.	ビット・タイミング	537
図 204.	CAN フレーム	538
図 205.	イベント・フラグと割込み生成	539
図 206.	SPI ブロック図	570
図 207.	シングル・マスタ / シングル・スレーブ・アプリケーション	571
図 208.	ハードウェア / ソフトウェア・スレーブ選択の管理	571
図 209.	データとクロックのタイミング	573
図 210.	I ² S ブロック図	580
図 211.	I ² S フィリップスプロトコル波形 (16/32bit フル精度、CPOL = 0)	582
図 212.	I ² S フィリップス標準波形 (24bit フレーム、CPOL = 0)	582
図 213.	0x8EAA33 の送信	582
図 214.	0x8EAA33 の受信	583
図 215.	I ² S フィリップス標準 (32bit パケットフレームに拡張された 16bit、CPOL = 0)	583
図 216.	例	583
図 217.	MSB づめ 16bit または 32bit フル精度長、CPOL = 0	584
図 218.	MSB づめ 24bit フレーム長、CPOL = 0	584
図 219.	MSB づめ 32bit パケットフレームに拡張された 16bit、CPOL = 0	584
図 220.	LSB づめ 16bit または 32bit フル精度、CPOL = 0	585
図 221.	LSB づめ 24bit フレーム、CPOL = 0	585
図 222.	0x3478AE を送信するのに必要な動作	585
図 223.	0x3478AE の受信に必要な動作	586
図 224.	LSB づめ 32bit パケットフレームへ拡張された 16bit、CPOL = 0	586
図 225.	例	586
図 226.	PCM 標準波形 (16bit)	587
図 227.	PCM 標準波形 (32bit パケットフレームに拡張された 16bit)	587
図 228.	オーディオサンプリング周波数定義	588
図 229.	I ² S クロック生成アーキテクチャ	588
図 230.	I ² C バスプロトコル	605
図 231.	I ² C ブロック図	606
図 232.	スレーブ・トランスマッタの交信手順	607
図 233.	スレーブ・リーバの交信手順	608
図 234.	マスタトランスマッタの交信手順	611
図 235.	マスタリーバの交信手順	612
図 236.	I ² C 割込みマッピング	622
図 237.	USART ブロック図	641
図 238.	ワード長の設定	642
図 239.	制御 STOP ビットの設定	644
図 240.	ノイズ検出のためのデータサンプリング	647
図 241.	アイドル・ライン検出を利用するミュートモード	651
図 242.	アドレスマーク検出を利用するミュートモード	652
図 243.	LIN モードにおけるブレーク検出 (11 ビットブレーク長 (LBDL=1))	654
図 244.	LIN モードにおけるブレーク検出とフレーミング・エラー検出	655
図 245.	USART 同期シリアル送信の例	656
図 246.	USART データクロックタイミング (M=0)	656
図 247.	USART データクロックタイミング (M=1)	657
図 248.	RX データセットアップ / ホールドタイム	657
図 249.	ISO 7816-3 非同期プロトコル	658
図 250.	1.5 STOP ビットを利用したパリティ・エラー検出	659
図 251.	IrDA SIR ENDEC - ブロック図	661
図 252.	IrDA データ変調 (3/16) - ノーマル・モード	661
図 253.	2 個の USART 間のハードウェア・フロー制御	663
図 254.	RTS フロー制御	664



参考資料

RM0008

図の一覧

図 255.	CTS フロー制御	664
図 256.	USART 割込みマッピング	665
図 257.	ブロック図 – STM32F10xxx レベルと Cortex-M3 レベルのデバッグ・サポート	683
図 258.	SWJ デバッグ・ポート	684
図 259.	JTAG TAP 接続	688
図 260.	TPIU ブロック図	704



参考資料

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

レジスタの各ビットに関しては、次の略記が使用されます。

read/write (rw)	ソフトウェアからビット (bit) の読み出し / 書込みができます。
read-only (r)	ソフトウェアでは、ビット (bit) の読み出しのみが可能です。
write-only (w)	ソフトウェアではビット (bit) への書込みのみが可能です。ビット (bit) の読み出しでは、常にリセット時の値が読み出されます。
read/clear (rc_w1)	ソフトウェアではビット (bit) に”1”を書込むと、読み出しとクリアができます。”0”を書込んだときにはビットの値は変化しません。
read/clear (rc_w0)	ソフトウェアではビット (bit) に”0”を書込むと、読み出しとクリアができます。”1”を書込んだときにはビットの値は変化しません。
read/set (rs)	ソフトウェアではビット (bit) の読み出しとセットができます。”0”を書込んだときにはビットの値は変化しません。
toggle(t)	ソフトウェアではビット (bit) に”1”を書込むとトグル（反転）のみができます。”0”を書込んだときにはビットの値は変化しません。

1.2 用語

- Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。
- Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

2 メモリとバスの構成

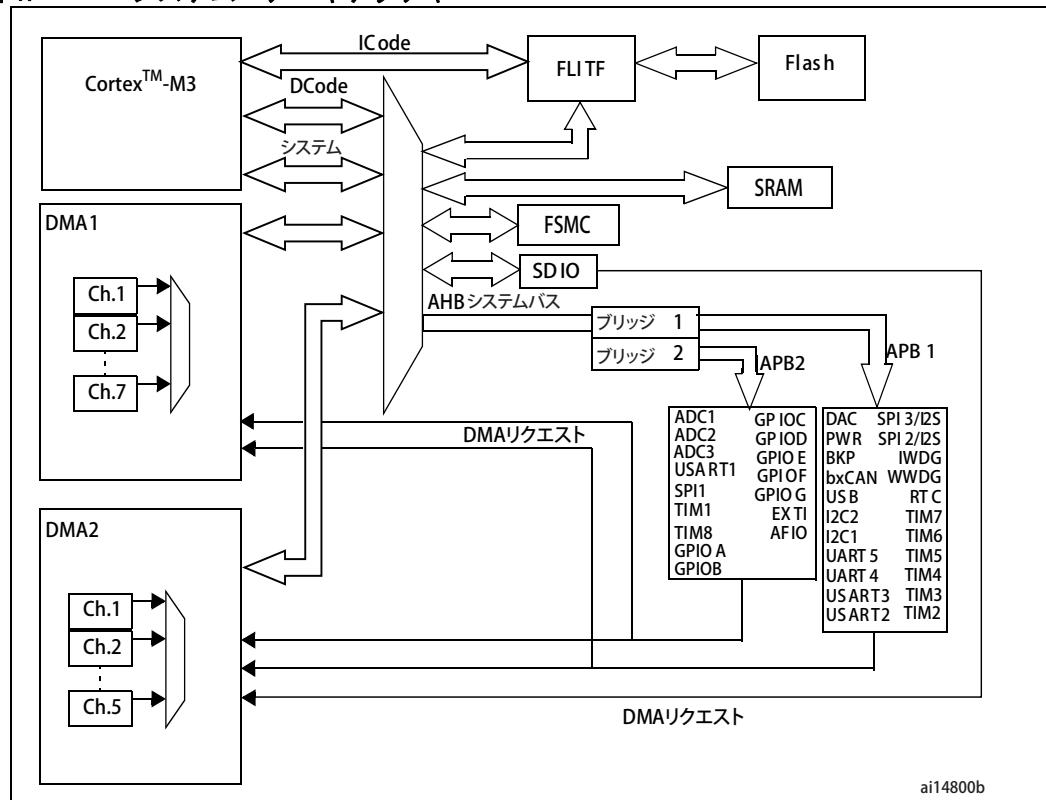
2.1 システム・アーキテクチャ

システムの中核は次の要素で構成されています。

- 5 個のマスタ回路 :
 - CortexTM-M3 コア、 ICode バス (I-bus)、 DCode バス (D-bus)、 システム・バス (S-bus)
 - GP-DMA1 及び 2 (汎用 DMA)
- 3 個のスレーブ回路 :
 - 内蔵 SRAM
 - 内蔵 Flash メモリ
 - すべての APB 周辺回路を接続するための AHB/APB ブリッジ (AHB2APBx)

これらは図 1 に示されているように、マルチレイヤ AHB バス・アーキテクチャで相互に接続されています。

図 1. システム・アーキテクチャ



ICode バス

このバスは CortexTM-M3 コアの命令語バス (Instruction bus) を Flash メモリの命令語インターフェースに接続しています。命令語のプリフェッчはこのバスを経由して行われます。

参考資料

DCode バス

このバスは CortexTM-M3 コアの Dcode バス（リテラルのロードとデバッグのためのアクセス）と Flash メモリ・インターフェースに接続しています。

システム・バス

このバスは CortexTM-M3 のシステム・バス（ペリフェラル・バス）をバス・マトリックスと接続し、コアと DMA の間のアクセスの競合を管理しています。

DMA バス

このバスは DMA の AHB マスタ・インターフェースとバス・マトリックスの間を接続し、CPU の DCode バスと DMA から SRAM や Flash メモリ、周辺回路へのアクセスを管理しています。

バス・マトリックス (BusMatrix)

バス・マトリックスはコアシステム・バスと DMA マスタバスの間のアクセス競合をラウンドロビン方式で管理しています。バス・マトリックスは 3 個のマスタ（CPU DCode バス、システム・バス、DMA バス）と 3 個のスレーブ（FLITF、SRAM、AHB/APB ブリッジ）から構成されています。

AHB ペリフェラルは DMA アクセスを可能とするため、バス・マトリックスを経由してシステム・バスに接続されています。

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。APB1 は上限周波数 36 MHz で、APB2 はフルスピード（デバイスに依存し、最大 72 MHz）で動作します。

各ブリッジに接続されている周辺回路のアドレス割当てに関しては [表 1 \(36 ページ\)](#) を参照してください。

各デバイス・リセットの後、すべてのペリフェラル・クロックはディセーブルになります。（SRAM と FLITF は除く）ペリフェラルを使う前に、RCC_AHBENR、RCC_APB2ENR または RCC_APB1ENR レジスタで、クロックをイネーブルにする必要があります。

注意： APB レジスタへ 16 または 8bit アクセスが行われると、アクセスは 32bit アクセスへ変換されます。ブリッジが 16 または 8bit データを複製し、32bit のベクタを提供します。

2.2 メモリ構成

プログラムメモリ、データメモリ、レジスタと入出力ポートは同じ 4 G バイトのリニアなアドレス空間に配置されています。

メモリの各バイトはリトル・エンディアン形式で、ワード内のもっとも小さなアドレスのバイトが最下位バイトに、もっとも大きなアドレスのバイトが最上位バイトになります。

STM32F10xxx のメモリ・マップは [図 2 \(35 ページ\)](#) に示されています。周辺回路のレジスタのアドレスに関しては、対応する個々の章を参照してください。

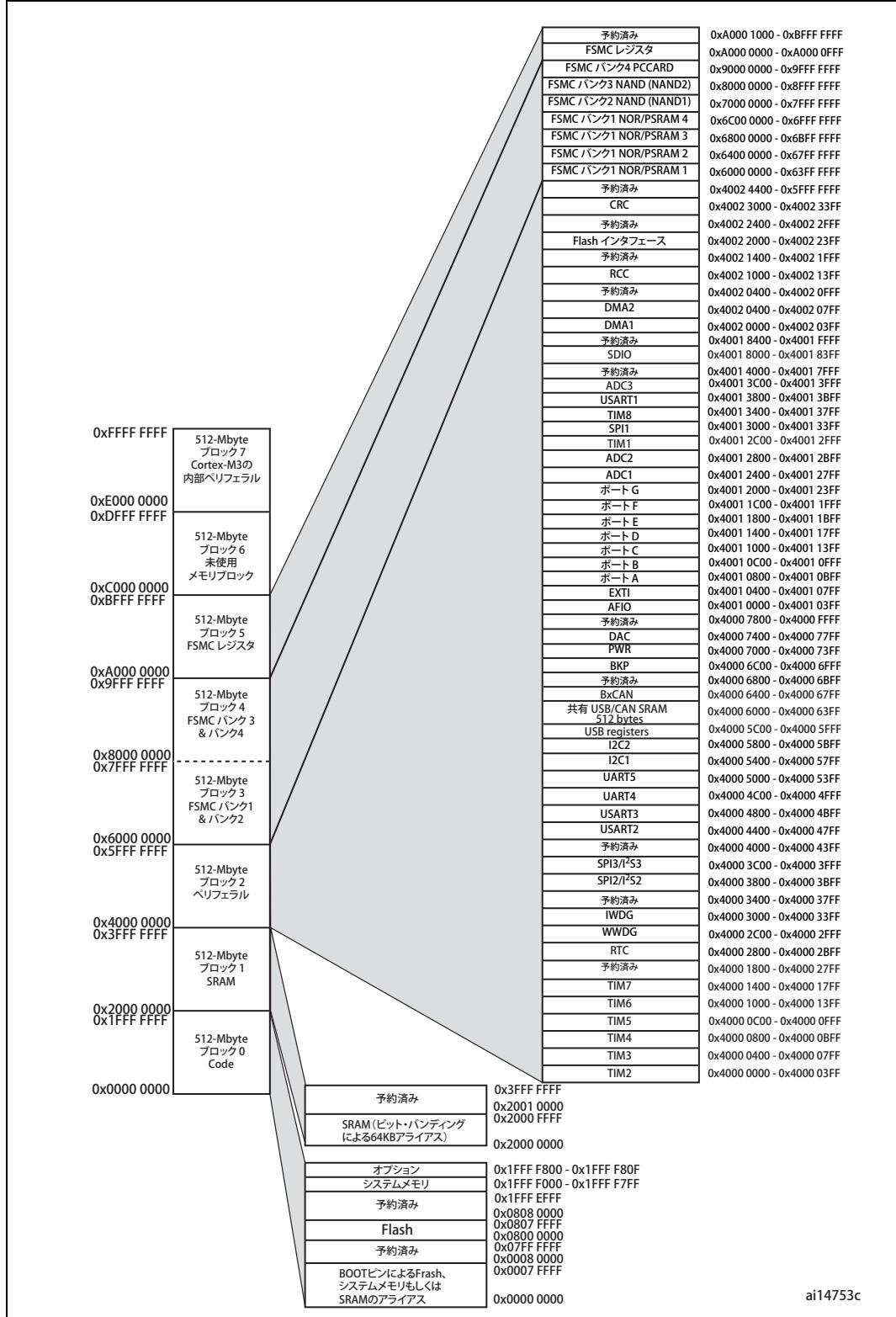
アドレス可能なメモリ空間は、8 個の、それぞれ 512 MB のブロックに分けられています。

チップ上のメモリやペリフェラル回路に割当てられていないメモリ・ドメインは、すべて「予約済み」になります。（このドメインは [図 2 \(35 ページ\)](#) では網掛けが施されています。）

参考資料

2.3 メモリ・マップ

図 2. メモリ・マップ



参考資料

2.3.1 ペリフェラルメモリ・マップ

表 1. レジスタ・バウンダリ・アドレス

アドレスの範囲	ペリフェラル回路	バス	レジスタ・マップ
0x4002 2400 – 0x4002 2FFF	予約済み	AHB	
0x4002 3000 – 0x4002 33FF	CRC		セクション 3.4.4 (47 ページ)
0x4002 2000 – 0x4002 23FF	Flash メモリ インターフェース		
0x4002 1400 – 0x4002 1FFF	予約済み		
0x4002 1000 – 0x4002 13FF	リセット・クロック制御 RCC		セクション 6.3.11 (101 ページ)
0x4002 0800 – 0x4002 0FFF	予約済み		
0x4002 0400 – 0x4002 07FF	DMA2		セクション 9.4.7 (153 ページ)
0x4002 0000 – 0x4002 03FF	DMA1		セクション 9.4.7 (153 ページ)
0x4001 8400 – 0x4001 FFFF	予約済み		
0x4001 8000 – 0x4001 83FF	SDIO		セクション 19.9.16 (488 ページ)
0x4001 4000 – 0x4001 7FFF	予約済み	APB2	
0x4001 3C00 – 0x4001 3FFF	ADC3		セクション 10.12.15 (189 ページ)
0x4001 3800 – 0x4001 3BFF	USART1		セクション 24.6.8 (678 ページ)
0x4001 3400 – 0x4001 37FF	TIM8 タイマ		セクション 12.4.21 (283 ページ)
0x4001 3000 – 0x4001 33FF	SPI1		セクション 22.5 (594 ページ)
0x4001 2C00 – 0x4001 2FFF	TIM1 タイマ		セクション 12.4.21 (283 ページ)
0x4001 2800 – 0x4001 2BFF	ADC2		セクション 10.12.15 (189 ページ)
0x4001 2400 – 0x4001 27FF	ADC1		セクション 10.12.15 (189 ページ)
0x4001 2000 – 0x4001 23FF	GPIO ポート G		セクション 7.5 (129 ページ)
0x4001 1C00 – 0x4001 1FFF	GPIO ポート F		セクション 7.5 (129 ページ)
0x4001 1800 – 0x4001 1BFF	GPIO ポート E		セクション 7.5 (129 ページ)
0x4001 1400 – 0x4001 17FF	GPIO ポート D		セクション 7.5 (129 ページ)
0x4001 1000 – 0x4001 13FF	GPIO ポート C		セクション 7.5 (129 ページ)
0x4001 0C00 – 0x4001 0FFF	GPIO ポート B		セクション 7.5 (129 ページ)
0x4001 0800 – 0x4001 0BFF	GPIO ポート A		セクション 7.5 (129 ページ)
0x4001 0400 – 0x4001 07FF	EXTI		セクション 8.3.7 (140 ページ)
0x4001 0000 – 0x4001 03FF	AFIO		セクション 7.5 (129 ページ)

参考資料

表 1. レジスタ・バウンダリ・アドレス (続き)

アドレスの範囲	ペリフェラル回路	バス	レジスタ・マップ
0x4000 7800 – 0x4000 FFFF	予約済み	APB1	
0x4000 7400 – 0x4000 77FF	DAC		セクション 11.5.14 (214 ページ)
0x4000 7000 – 0x4000 73FF	電源制御 PWR		セクション 4.4.3 (62 ページ)
0x4000 6C00 – 0x4000 6FFF	バックアップ・レジスタ BKP		セクション 5.4.5 (69 ページ)
0x4000 6800 – 0x4000 6BFF	予約済み		
0x4000 6400 – 0x4000 67FF	bxCAN		セクション 21.6.5 (564 ページ)
0x4000 6000 – 0x4000 63FF	USB/CAN 共用 SRAM 512bytes		
0x4000 5C00 – 0x4000 5FFF	USB レジスタ		セクション 20.5.4 (518 ページ)
0x4000 5800 – 0x4000 5BFF	I2C2		セクション 23.6.10 (636 ページ)
0x4000 5400 – 0x4000 57FF	I2C1		セクション 23.6.10 (636 ページ)
0x4000 5000 – 0x4000 53FF	UART5		セクション 24.6.8 (678 ページ)
0x4000 4C00 – 0x4000 4FFF	UART4		セクション 24.6.8 (678 ページ)
0x4000 4800 – 0x4000 4BFF	USART3		セクション 24.6.8 (678 ページ)
0x4000 4400 – 0x4000 47FF	USART2		セクション 24.6.8 (678 ページ)
0x4000 4000 – 0x4000 3FFF	予約済み		
0x4000 3C00 – 0x4000 3FFF	SPI3/I2S		セクション 22.5 (594 ページ)
0x4000 3800 – 0x4000 3BFF	SPI2/I2S		セクション 22.5 (594 ページ)
0x4000 3400 – 0x4000 37FF	Reserved		
0x4000 3000 – 0x4000 33FF	独立型ウォッチドッグ IWDG		セクション 16.4.5 (372 ページ)
0x4000 2C00 – 0x4000 2FFF	Window 型ウォッチドッグ WWDG		セクション 17.6.4 (379 ページ)
0x4000 2800 – 0x4000 2BFF	RTC		セクション 15.4.7 (366 ページ)
0x4000 1800 – 0x4000 27FF	予約済み		
0x4000 1400 – 0x4000 17FF	TIM7 タイマ		セクション 14.4.9 (353 ページ)
0x4000 1000 – 0x4000 13FF	TIM6 タイマ		セクション 14.4.9 (353 ページ)
0x4000 0C00 – 0x4000 0FFF	TIM5 タイマ		セクション 13.4.19 (340 ページ)
0x4000 0800 – 0x4000 0BFF	TIM4 タイマ		セクション 13.4.19 (340 ページ)
0x4000 0400 – 0x4000 07FF	TIM3 タイマ		セクション 13.4.19 (340 ページ)
0x4000 0000 – 0x4000 03FF	TIM2 タイマ		セクション 13.4.19 (340 ページ)

2.3.2 内蔵 SRAM

STM32F10xxx は 64Kbytes の SRAM を搭載しています。SRAM は、バイト、ハーフワード（16bit）、フルワード（32bit）によるアクセスが可能で、そのアドレスは 0x2000 0000 から始まります。

2.3.3 ビット・バンディング

CortexTM-M3 のメモリ・マップには 2 つのビットバンド・ドメインが組込まれています。これらのドメインでは、メモリのエイリアス・ドメインの各ワードをメモリのビットバンド・ドメインのビットに配置しています。エイリアス・ドメインのワードへの書き込みは、ビットバンド・ドメインの対象ビットに読み込み / モディファイ / 書き込み動作でアクセスすることと同じ効果をもちます。

STM32F10xxx では、周辺回路のレジスタも SRAM ドメインと同じビットバンド・ドメインにマッピングされています。これにより、ビットバンド・ドメインの書き込み・読み出し操作を同じように実行することができます。

マッピングの計算式は、ビットバンド・ドメインの対象ビットがどのようにエイリアス・ドメインのワードに対応させられるかを示しています。この式は、

$$\text{bit_word_addr} = \text{bit_band_base} + (\text{byte_offset} \times 32) + (\text{bit_number} - 4)$$

ここで、

bit_word_addr は対象ビットに対応させられるエイリアス・ドメインのワードのアドレスです。

bit_band_base はエイリアス・ドメイン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・ドメイン内のバイトのオフセット値です。

bit_number は対象ビットのビット位置 (0-7) です。

例：

以下の例は SRAM アドレス 0x20000300 のバイトがエイリアス・ドメインにどのように配置されているかを示しています。

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

アドレス 0x22006008 への書き込みは SRAM アドレス 0x20000300 のバイトのビット 2 に対して読み込み / モディファイ / 書き込みを実行することと同じ効果があります。

アドレス 0x22006008 の読み出しだけでは、SRAM アドレス 0x20000300 のバイトのビット 2 の状態が、ビットがセットされていれば 0x01 として、リセットされていれば 0x00 として読み出されます。

ビット・バンディングの詳細に関しては、CortexTM-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。

2.3.4 内蔵 Flash メモリ

高機能な Flash メモリ・モジュールは次の特徴をもっています。

- 容量 512Kbyte
- メモリ構成: Flash メモリはメインブロックと情報ブロックで構成されています。
 - 中容量デバイスは 16Kb × 64bits を 1Kbyte のページ 128 ページで分割します。
([表 2](#)を参照)
 - 大容量デバイスでは 64 Kb × 64bits を 2Kbytes の 256 ページへ分割します。
([表 3](#)を参照)
 - 情報ブロックは 258 × 64bits で、2Kbytes と 16bytes の 2 つのページに分割します。
([表 2](#)を参照)

Flash メモリ・インターフェース (ELITF) は次の機能を持っています。

- プリフェッч・バッファを持つ読み込みインターフェース (2 × 64bit ワード)
- オプションバイト・ローダ
- Flash プログラム書き込み / 消去操作
- 読込み / 書込み禁止機能

表 2. Flash モジュールの構成 (中容量デバイス)

ブロック	名称	アドレス	サイズ (bytes)
メイン・メモリ	ページ 0	0x0800 0000 – 0x0800 03FF	1 Kbyte
	ページ 1	0x0800 0400 – 0x0800 07FF	1 Kbyte
	ページ 2	0x0800 0800 – 0x0800 0BFF	1 Kbyte
	ページ 3	0x0800 0C00 – 0x0800 0FFF	1 Kbyte
	ページ 4	0x0800 1000 – 0x0800 13FF	1 Kbyte
	.	.	.
	ページ 127	0x0801 FC00 – 0x0801 FFFF	1 Kbyte
情報ブロック	システム・メモリ	0xFFFF F000 – 0xFFFF F7FF	2 Kbytes
	オプション・バイト	0xFFFF F800 – 0xFFFF F80F	16
Flash メモリ・インターフェース・レジスタ	FLASH_ACR	0x4002 2000 – 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 – 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 – 0x4002 200B	4
	FLASH_SR	0x4002 200C – 0x4002 200F	4
	FLASH_CR	0x4002 2010 – 0x4002 2013	4
	FLASH_AR	0x4002 2014 – 0x4002 2017	4
	予約済み	0x4002 2018 – 0x4002 201B	4
	FLASH_OBR	0x4002 201C – 0x4002 201F	4
	FLASH_WRPTR	0x4002 2020 – 0x4002 2023	4

参考資料

表 3. Flash モジュールの構成（大容量デバイス）

ブロック	名称	アドレス	サイズ (bytes)
メイン・ブロック	ページ 0	0x0800 0000 – 0x0800 07FF	2 Kbytes
	ページ 1	0x0800 0800 – 0x0800 0FFF	2 Kbytes
	ページ 2	0x0800 1000 – 0x0800 17FF	2 Kbytes
	ページ 3	0x0800 1800 – 0x0800 1FFF	2 Kbytes
	.	.	.
	ページ 255	0x0807 F800 – 0x0807 FFFF	2 Kbytes
	.	.	.
情報ブロック	システム・メモリ	0x1FFF F000 – 0x1FFF F7FF	2 Kbytes
	オプション・バイト	0x1FFF F800 – 0x1FFF F80F	16
Flash メモリ・インターフェース・レジスタ	FLASH_ACR	0x4002 2000 – 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 – 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 – 0x4002 200B	4
	FLASH_SR	0x4002 200C – 0x4002 200F	4
	FLASH_CR	0x4002 2010 – 0x4002 2013	4
	FLASH_AR	0x4002 2014 – 0x4002 2017	4
	予約済み	0x4002 2018 – 0x4002 201B	4
	FLASH_OBR	0x4002 201C – 0x4002 201F	4
	FLASH_WRPR	0x4002 2020 – 0x4002 2023	4

注意： Flash メモリ・インターフェース・レジスタの詳細は、 STM32F10xxx Flash プログラミングマニュアルを参照してください。

Flash メモリの読み出し

Flash メモリの操作やデータ・アクセスは AHB バスを経由して行われます。ICode バスを通しての命令語フェッチでは、プリフェッチ回路が使用されます。アクセス競合の管理は Flash メモリ・インターフェースで行われ、DCode バスを経由するデータ・アクセスが最も優先されます。

Flash メモリの読み出しの際には、次のオプションが指定できます。

- 遅延：読み込みアクセスの際のウェイト・ステート数は on-the-fly でプログラム設定ができます。
- プリフェッチ・バッファ ($2 \times 64\text{bit}$ ブロック)：この機能は、リセット後にイネーブルすることが出来ます。ブロック・サイズが Flash メモリのバンド幅に適合するように、全体のブロックは Flash メモリから一回の読み込みで置き換えることが出来ます。プリフェッチ・バッファがあるため、CPU がプリフェッチ・バッファですぐに利用できる、次のワードで一度に一つのワードをフェッチするため、より高速な CPU 動作を行うことが出来ます。
- ハーフ・サイクル：消費電力との関連で決めることができます。

注意： 1 これらのオプションは Flash メモリのアクセスタイムとの兼ね合いで選択されます。ウェイト・ステートは Flash メモリ・アクセスタイムに対する SYSCLK (システム・クロック) 周期の比率を表します。
 $0 < \text{SYSCLK} \leq 24\text{ MHz}$ の場合、ゼロウェイト・ステート
 $24\text{ MHz} < \text{SYSCLK} \leq 48\text{ MHz}$ の場合、1 ウェイト・ステート
 $48\text{ MHz} < \text{SYSCLK} \leq 72\text{ MHz}$ の場合、2 ウェイト・ステート

- 2 ハーフサイクルの構成は AHB 上のプリスケーラと同時に利用することはできません。システム・クロック (SYSCLK) は HCLK クロックと同じにする必要があります。このため、この機能は 8 MHz 以下の低周波数クロックのみで使用されます。また、クロックは HSI もしくは HSE から生成され、PLL からは生成されません。
- 3 AHB クロックでは 1 と異なるプリスケーラを使用する場合、プリフェッチ・バッファは維持される必要があります。
- 4 プリフェッチ・バッファは SYSCLK が 24MHz より低いときのみ、オン／オフしなければなりません。プリフェッチ・バッファは、通常初期化ルーチンでマイクロコントローラが内部 8MHzRC (HSI) オシレータで動作中にオン／オフされます。
- 5 DMA の利用：DMA 回路は DCode バスを経由して Flash メモリにアクセスしており、ICode からのアクセスより優先されます。DMA では、各転送動作のあとに 1 サイクル分の空き時間がおかれるよう制御しています。いくつかの命令語は DMA 転送と同時に実行されます。

Flash メモリの書き込みと消去

Flash メモリは一度に 16bit (ハーフ・ワード) ずつ書き込まれます。

Flash メモリの消去はページ単位か、もしくは Flash メモリ全体に対して（全体消去）実行されます。全体消去の場合でも、情報ブロックは消去されません。

正確な Flash 書込み時間を維持するため、Flash 書込み / 消去コントローラには固定されたのクロックが供給されています。

データ書き込みや消去の終了時に割込みリクエストを発生させることができます。FLITF クロックがイネーブルの場合は、この割込みトリガで WFI モードから終了することができます。FLITF クロックがディセーブルのときには WFI モードから終了した後に割込み処理動作が行われます。

注意： Flash メモリ動作とレジスタ設定についての詳細は、STM32F10xxx Flash プログラミングマニュアルを参照してください。

2.4 ブート構成

STM32F10xxx では 3 種のブート動作が可能で、次の [表 4](#) のように BOOT[1:0] ピンによって選択できます。

表 4. ブート・モード

ブート・モード選択 ピン		ブート・モード	摘要
BOOT1	BOOT0		
x	0	メイン Flash メモリ	メイン Flash メモリからブートが行われます。
0	1	システム・メモリ	システム・メモリからブートが行われます。
1	1	内蔵 SRAM	内蔵 SRAM からブートが行われます。

ブート・モードで指定された物理メモリは Block 000 (ブート・メモリ) に割付けられます。BOOT ピンの状態は、リセット後 SYSCLK の 4 番目の立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。

BOOT ピンは STANDBY モードを終了する場合にもサンプルされます。したがって、これらのピンは STANDBY モードの中で必要とされるブート・モードを指示するよう、信号レベルが維持する必要があります。

ブート・メモリ空間に割り当てられている関連するメモリ (Flash メモリもしくは SRAM メモリ) は、同時に、メモリとしての通常のアドレスからもアクセスが可能です。

起動時、所定の時間が経過した後、ブート・メモリの最下位アドレス 0x0000 0000 から CPU の命令実行が始まります。

内蔵ブートローダ

内蔵ブートローダは USART1 シリアル・インターフェースにより、Flash メモリを再プログラムするために使われます。このプログラムは ST により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。詳細については AN2606 を参照してください。

参考資料

RM0008

メモリとバスの構成



Rev 5 — 日本語版

43/720

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。

参考資料

3 CRC 計算ユニット

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、中容量デバイスです。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、大容量デバイスです。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

3.1 概要

CRC (Cyclic Redundancy Check) 計算ユニットは、32bit データワードと決められた生成多項式から CRC コードを取得します。

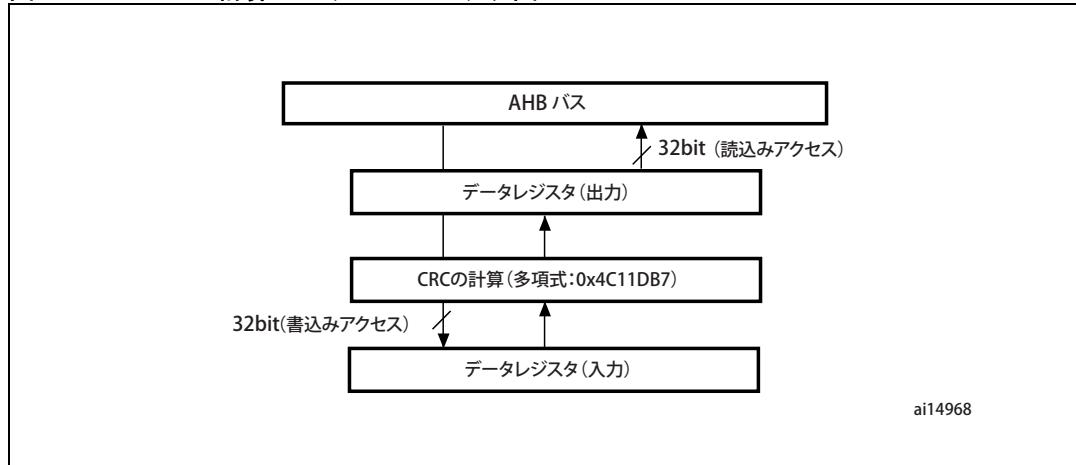
他のアプリケーションの中でも、CRC ベースのテクニックはデータ転送やストレージの整合性を実証するために用いられます。EN/IEC60335-1 標準の中で、Flash メモリの整合性を実証することが挙げられています。リンク時に生成され、与えられたメモリ・ロケーション上に保存されたリファレンス符号と比較するために、CRC 計算ユニットは、ランタイム時にソフトウェアの符号計算を支援します。

3.2 主な機能

- CRC-32 (Ethernet) 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 32bit データ・レジスタ
- CRC の計算は 4 AHB クロック・サイクル (HCLK) で完了
- 汎用 8bit レジスタ (テンポラリ・ストレージとして利用可能)

図 3 にブロック図を示します。

図 3. CRC 計算ユニット・ブロック図



3.3 CRC 機能詳細

CRC 計算ユニットは、主に次のような 1 つの 32bit データ・レジスタから構成されています。

- CRC 計算機に新たなデータを入力する（レジスタへ書込みする場合）入力レジスタとして使用
- 前の CRC 計算結果を保持（レジスタを読み込みする場合）

データ・レジスタへの各書込みオペレーションは、前の CRC 値と新たな値（CRC 計算は 32bit データ・ワード単位で行われ、バイト毎ではありません。）の組み合わせを作ります。

CPU は処理の間は停止しているため、ソフトウェア・ウェイト・サイクルを挿入せずに、back-to-back の書込みアクセスや、連続した書込みと読み込みアクセスを可能にします。

CRC 計算機は、CRC_CR レジスタの RESET 制御ビットで、FFFF FFFFh にリセットすることができます。このオペレーションは CRC_IDR レジスタの内容に影響しません。

3.4 CRC レジスタ

CRC 計算ユニットには、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。

3.4.1 データ・レジスタ (CRC_DR)

アドレス・オフセット : 0x00

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR [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
DR [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 データ・レジスタ・ビット

新たなデータを CRC 計算機に書込む時に入力レジスタとして使います。
読み込み時は、前の CRC 計算結果を保持しています。

参考資料

3.4.2 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]							
-	-	-	-	-	-	-	-	rW	rW	rW	rW	rW	rW	rW	rW

Bits 31:8 予約済み

Bits 7:0 汎用 8bit データ・レジスタ・ビット

1 バイトのテンポラリストレージとして使う事が出来ます。

このレジスタは、CRC_CR レジスタの RESET ビットにより生成される CRC リセットの影響を受けません。

3.4.3 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															RESET
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	w

Bits 31:1 予約済み

Bits 0 RESET ビット

CRC 計算ユニットをリセットし、データ・レジスタを FFFF FFFFh にセットします。

このビットはセットのみで、ハードウェアにより自動的にクリアされます。

参考資料

3.4.4 CRC レジスタ・マップ

次のテーブルは、CRC レジスタ・マップとリセット値を示します。

表 5. CRC 計算ユニットーレジスタ・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 リセット値	データ・レジスタ 0xFFFF FFFF											
0x04	CRC_IDR リセット値	予約済み				独立型データ・レジスタ 0x00							
0x08	CRC_CR リセット値	予約済み										予約済み 0	RESET 0

4 電源制御 (PWR)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

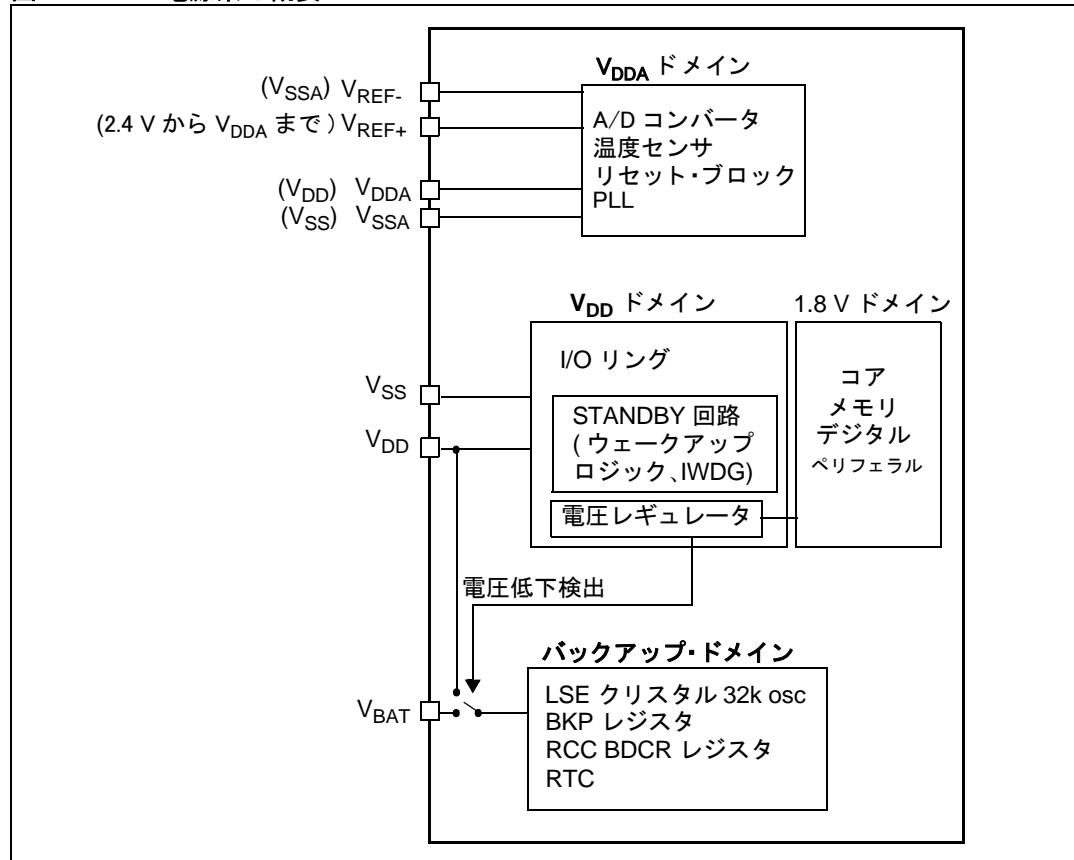
このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

4.1 電源供給

このデバイスは 2.0 ~ 3.6 V の V_{DD} 電源供給を必要とします。内部の 1.8 V デジタル電源を供給するために、レギュレータが搭載されています。

リアルタイム・クロック (RTC) とバックアップ・レジスタの電源は、主電源 V_{DD} がオフの時には V_{BAT} から供給されます。

図 4. 電源系の概要



注意: 1 V_{DDA} と V_{SSA} はそれぞれ V_{DD} と V_{SS} へ接続する必要があります。

4.1.1 独立した A/D コンバータ電源と参照電圧

AD 変換の精度を向上させるため、ADC には独立した電源が供給されます。この電源回路には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 電源の入力は他と分離した V_{DDA} ピンから供給されます。
 - ADC 電源のグランドは、独立した V_{SSA} ピンとして引出されています。
- パッケージにより V_{REF-} は、 V_{SSA} に接続する必要があります。

100 ピン及び 144 ピンのパッケージ

高精度を維持し入力電圧を低く保つため、独自の外部参照電圧を ADC の V_{REF+} 及び V_{REF-} に接続することができます。 V_{REF+} の電圧は 2.4V から V_{DDA} の範囲になります。

64 ピンのパッケージ

V_{REF+} と V_{REF-} は提供されていません。これらはチップ内で ADC 電源 (V_{DDA}) 及びグランド (V_{SSA}) と接続されています。

4.1.2 バッテリ・バックアップ・ドメイン

V_{DD} がオフのときにバックアップ・レジスタの内容を保持するのと RTC の機能を提供するため、 V_{BAT} ピンを電池などの待機電源により提供されるオプションの STANDBY 電源に接続することができます。

V_{BAT} ピンからは RTC 回路にも電源供給されており、LSE オシレータと PC13 ~ 15 の入出力はメインのデジタル電源 (V_{DD}) がオフの間も RTC が動作しつづけることができます。 V_{BAT} 電源への切替えはリセットブロックに組込まれているパワー・ダウン・リセット回路によって行われます。

要注意： V_{DD} スタート時の過渡時間 $t_{RSTTEMPO}$ の間、 V_{BAT}/V_{DD} の電源スイッチは V_{BAT} 側に接続されています。 V_{DD} の立ち上がりが早く、この時間内に所定の電圧に達する場合には、 V_{BAT} が $V_{DD}-0.6V$ より低くなると V_{DD} と V_{BAT} の間のダイオードを通して V_{BAT} に電流が流入します。 $t_{RSTTEMPO}$ の値に関してはデータシートを参照してください。

外部バッテリが使用されないアプリケーションでは、 V_{BAT} はチップ外で 100 nF の外部セラミック安定化コンデンサを経由して V_{DD} と接続します。(詳細は AN2586 を参照)

バックアップ・ドメインが V_{DD} (アナログスイッチを V_{DD} へ接続) によって供給を受けている場合には次の機能が利用できます。

- PC14 と PC15 は GPIO あるいは LSE ピンとして使うことができます
- PC13 は GPIO、TAMPER ピン、RTC 較正クロック、RTC アラームまたは第二出力として使うことができます (セクション 5: バックアップ・レジスタ (BKP) (63 ページ) を参照)

注意： スイッチに流すことができる電流の量は限られているので (3mA)、GPIO の PC13 から PC15 の利用は限られ、同時に 1 つの出力のみ使用することが出来ます。最大負荷 30 pF 時の最大速度は 2MHz で、これらの入出力は電流ソースとして使うことはできません。50 pF 負荷で 3.6V の平均消費電流は、2MHz で 0.4mA です。

参考資料

バックアップ・ドメインが V_{BAT} (V_{DD} は存在しないのでアナログスイッチは V_{BAT} へ接続) によって供給をうけている場合には、次の機能が利用できます。

- PC14 と PC15 は LSE ピンとしてのみ使うことができます。
- PC13 は TAMPER ピン、RTC アラームまたは第二出力として使うことができます。([セクション 5.4.2: RTC クロック較正レジスタ \(BKP_RTCCR\) \(66 ページ\)](#) を参照)

4.1.3 電圧レギュレータ

リセット後、電圧レギュレータは常に動作状態となり、アプリケーションの状態に応じて 3 種類のモードで動作します。

- Run モード：レギュレータは 1.8 V ドメイン（コア、メモリ、デジタルペリフェラル回路）に全電力を供給します。
- STOP モード：レギュレータは 1.8 V ドメインに対して、レジスタや SRAM の内容を保持できる、限定された電力を供給します。
- STANDBY モード：レギュレータはパワー・オフ状態です。STANDBY 回路とバックアップ・ドメイン以外のレジスタや SRAM の内容は失われます。

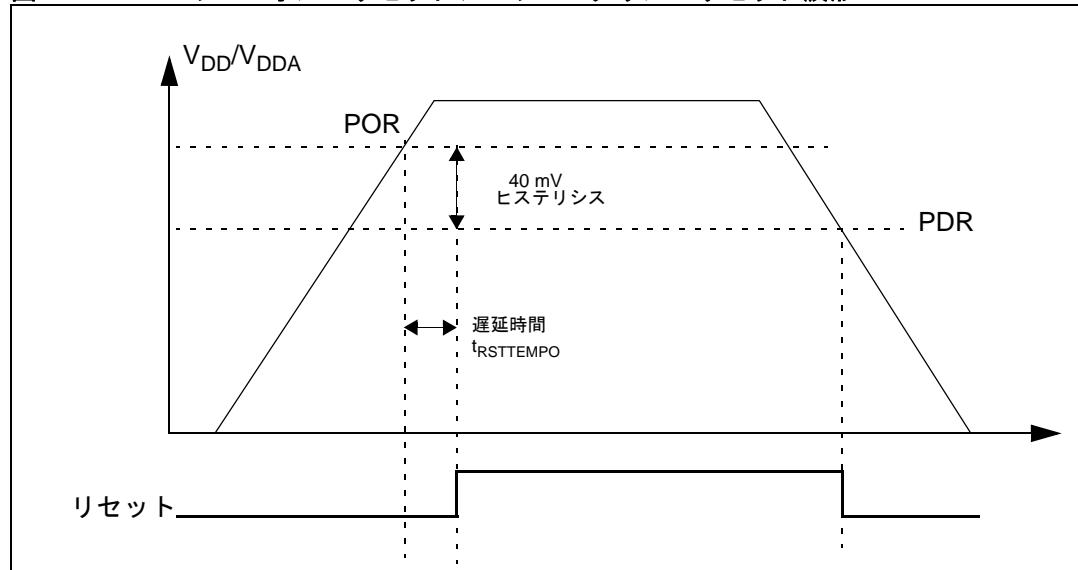
4.2 電源供給管理

4.2.1 パワー・オン・リセット (POR)/ パワー・ダウン・リセット (PDR)

このデバイスには、電源電圧が 2 V を超え、もしくは低下したときに必要な動作を行わせるための、POR/PDR 回路が搭載されています。

デバイスは V_{DD} / V_{DDA} が所定のスレッショルド ($V_{POR/PDR}$) を下回っている間は、外部のリセット回路を必要とせずに、リセット・モードを維持することができます。POR/PDR 閾値に関しては、データシートの電気特性の項をご参照ください。

図 5. パワー・オン・リセット / パワー・ダウン・リセット波形



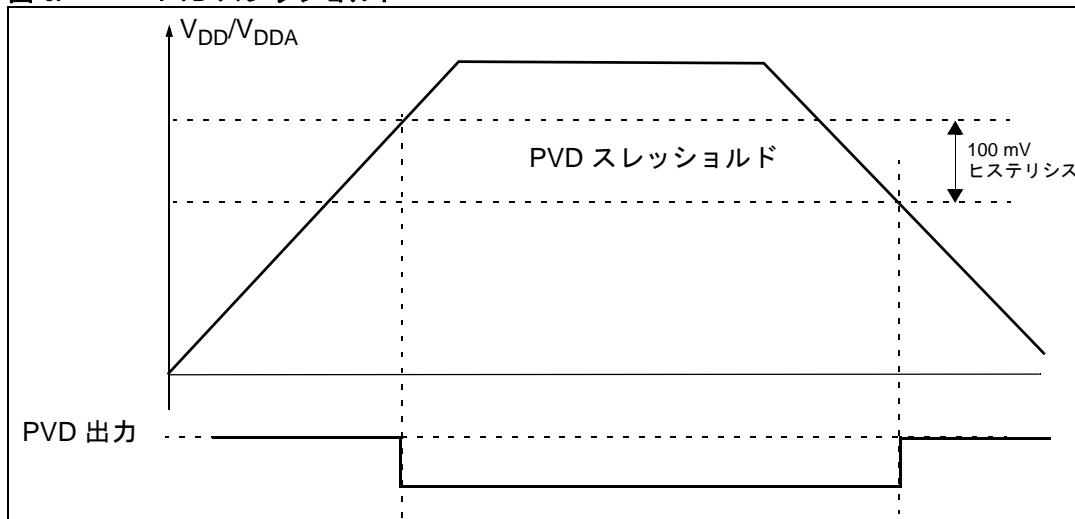
4.2.2 プログラム可能な電圧検出回路 (PVD)

電源電圧 V_{DD} / V_{DDA} と [電源制御レジスタ \(PWR_CR\)](#) の PLS[2:0] で指定されているスレッショルドを比べるために PVD を利用することができます。

PVD は PVDE ビットをセットするときに有効となります。

[電源制御 / ステータス・レジスタ \(PWR_CSR\)](#) の PVDO フラグは電源電圧 V_{DD} / V_{DDA} が PVD スレッショルドより高いか、低いかを示します。この状態はイベントとして EXTI のライン 16 に接続されており、EXTI レジスタで許可状態であれば、割込みリクエストを生成します。 V_{DD} / V_{DDA} が電圧スレッショルドを上回る場合にリクエストが生成するか、下回るときにリクエストが生成するかは、EXTI ライン 16 で割込みリクエストが発生する信号エッジの設定で決まります。これをを利用して、サービス・ルーチンで緊急停止処理を行わせることができます。

図 6. PVD スレッショルド



4.3 低電力モード

標準状態では、システム・リセットもしくは電源リセット後、マイクロコントローラは Run モードの状態です。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。外部からのイベント待ちなどで CPU がプログラムの実行が必要ないときに、消費電力を小さくするために、いくつかの低電力モードが用意されています。低電力モードは、消費電力と、ウェークアップのための最短スタートアップ時間とイベントとの関連で選択されます。

STM32F10xxx デバイスは次の 3 種の低電力モードを持っています。

- SLEEP モード (Cortex-M3 コア停止、ペリフェラルは動作継続)
- STOP モード (すべてのクロックが停止)
- STANDBY モード (1.8V 系ドメインの電源オフ)

加えて、Run モードの消費電力を低下させるために次の手段をとることができます。

- システム・クロック周波数の低減
- 使用しない APB、AHB のペリフェラル・クロックの停止

表 6. 低電力モード要約

モード名称	モード遷移イベント	ウェークアップイベント	1.8V ドメインのクロック	V_{DD} ドメインクロック	電圧レギュレータ
SLEEP (Sleep-now もしくは Sleep-on-exit)	WFI	割込み発生	CPU CLK OFF アナログ、及び 他のクロックへの影響なし	無し	ON
	WFE	ウェークアップ(Wake-up)イベント			
STOP	PDDS 及び LPDS ビット + SLEEPDEEP ビット + WFI または WFE	EXTI ライン (EXTI レジスタ設定に依存)	全 1.8V ドメイン クロック OFF	HSI、HSE オシレータ OFF	ON もしくは 低電力モード (電源制御レジスタ(PWR_CR) による)
STANDBY	PDDS ビット + SLEEPDEEP ビット + WFI または WFE	WKUP ピン立上り エッジ、RTC アラーム、NRST ピン外部リセット、 IWDG リセット			OFF

4.3.1 クロック周波数の低減

Run モードでは、プリスケーラ・レジスタのプログラム設定で Run モードのシステム・クロック (SYSCLK, HCLK, PCLK1, PCLK2) の周波数を低下させることができます。また、プリスケーラを使用して、SLEEP モードに入る前に、ペリフェラルの周波数を低下させておくこともできます。

この詳細は [セクション 6.3.2: クロック構成レジスタ \(RCC_CFGR\)](#) を参照してください。

4.3.2 ペリフェラル回路のクロック・スイッチ

Run モードで、個々のペリフェラルとメモリに対する HCLK 及び PCLK_x を任意に停止し、消費電力を低減することができます。

SLEEP モードにおける更なる電力低減のために、WFI 及び 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くことができます。

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)*、*APB1 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB1ENR)* 及び *APB2 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB2ENR)* で制御されます。

4.3.3 SLEEP モード

SLEEP モードに入るには

WFI (Wait For Interrupt) もしくは WFE (Wait for Event) 命令を実行することで、SLEEP モードに入ります。この際には、Cortex-M3 システム制御レジスタの SLEEPONEXIT ビットの設定に応じて、SLEEP モードに入るメカニズムを選択するために 2 つのオプションが利用できます。

- SLEEP-NOW: SLEEPONEXIT ビットがクリアされると、MCU は WFI、WFE 命令の実行と同時に 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時には、MCU は最も優先度の低い ISR を終了した時点で SLEEP モードに入ります。

SLEEP モードに入る際の動作の詳細は [表 7](#) 及び [表 8](#) を参照してください。

SLEEP モードを終了するには

WFI 命令によって SLEEP モードに入った時には、ペリフェラルの割込みリクエストがネスト化されたベクタ割込みコントローラ (NVIC) で受付けられると、デバイスは SLEEP モードを終了します。

WFE 命令により 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。ペリフェラル回路の制御レジスタで割込みリクエストとして許可されおり NVIC では受付けが許可されていないペリフェラル回路の割込みリクエストと、イベントモードが指定されている EXTI ラインからの信号が、このイベントとなります。

このモードは割込みの入り口と出口で余分な時間を必要としないため、ウェークアップ時間は最も短くなります。

SLEEP モードからの終了の詳細は、[表 7](#) 及び [表 8](#) を参照してください。

参考資料

表 7. Sleep-now

Sleep-now モード	詳細
モード遷移	次の条件下の WFI (Wait for Interrupt) もしくは WFE (Wait for Event) 命令 - SLEEPDEEP = 0 及び - SLEEPONEXIT = 0 Cortex™-M3 システム制御レジスタの説明を参照
モード終了	WFI 命令で SLEEP モードに入った場合、 割込み : 表 35: ベクタテーブル を参照 WFE 命令で SLEEP モードに入った場合、 ウェークアップ (Wake-up) イベント : セクション 8.2.3: ウェークアップ・イベント管理 を参照
ウェークアップ時の遅延	なし

表 8. Sleep-on-exit

Sleep-on-exit	摘要
モード遷移	次の条件下の WFI (Wait for Interrupt) もしくは WFE (Wait for Event) 命令 - SLEEPDEEP = 0 及び - SLEEPONEXIT = 1 Cortex™-M3 システム制御レジスタの説明を参照
モード終了	割込み : 表 35: ベクタテーブル を参照
ウェークアップ時の遅延	なし

4.3.4 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回路のクロック・スイッチ機能を組合わせたものです。電圧レギュレータは通常モードか、低電力モードのいずれかです。STOP モードでは 1.8 V ドメインのすべてのクロックは停止し、PLL、HSI、HSE RC 発振回路はディセーブル状態となります。SRAM とレジスタの内容は保持されています。

STOP モードに入るには

STOP モードへの入り方については [表 9](#) を参照してください。

STOP モードで、消費電力をより削減するには、内蔵されている電圧レギュレータを低電力モードにします。この設定は [電源制御レジスタ \(PWR_CR\)](#) の LPDS ビットで指定します。

Flash メモリの書き込みが進行中のときには、STOP モードへの遷移はメモリ・アクセスが終了するまで待たれます。

APB ドメインにアクセス中のときには、STOP モードへの遷移は APB アクセスが終了するまで待たれます。

STOP モードでは、次の回路で、個別の制御ビットによる機能設定ができます。

- 独立型ウォッチドッグ (IWDG) : IWDG の動作は、キーレジスタへの書き込み、もしくはハードウェアの設定で開始されます。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 16: 独立型ウォッチドッグ \(IWDG\) のセクション 16.3](#) 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 [バックアップ・ドメイン制御レジスタ \(RCC_BDCR\)](#) の RTCEN ビットで行います。
- 内蔵 RC オシレータ (LSI RC) : この設定は [制御/ステータス・レジスタ \(RCC_CSR\)](#) の LSION ビットで行います。
- 外部 32.768 kHz のオシレータ (LSE OSC) : この設定は [バックアップ・ドメイン制御レジスタ \(RCC_BDCR\)](#) の LSEON ビットで行います。

ADC や DAC は、STOP モードに入る前にディセーブルにされない限り、STOP モードの間は電力を消費します。ディセーブルするには、ADC_CR2 レジスタの ADON ビットと DAC_CR レジスタの ENx ビットの両方に 0 を書き込まなければなりません。

STOP モードを抜け出すには

STOP モードから抜け出す条件の詳細は [表 9](#) を参照してください。

割込みもしくはウェークアップ・イベントの発生で STOP モードを終了す場合、HSI RC オシレータがシステム・クロックとして選択されます。

電圧レギュレータが低電力モードの時には、STOP モードからのウェークアップの際に余分なスタートアップ時間を必要とします。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で、消費電力は増加しますがウェークアップに要する時間は短縮されます。

参考資料

表 9. STOP モード

STOP モード	摘要
モード遷移	次の条件下の WFI (Wait for Interrupt) もしくは WFE (Wait for Event) 命令 - Cortex TM -M3 システム制御レジスタの SLEEPDEEPbit をセット - 電力制御レジスタ (PWR_CR) の PDDE bit をクリア - PWR_CR レジスタの LPDS ビットで電圧レギュレータのモードを選択されている 注: STOP モードに入るには、EXTI ラインのペンドイングビット（ ペンドイング・レジスタ (EXTI_PR) ）すべてと RTC アラームフラグはリセットする必要があります。リセットしない場合、STOP モードに入る手順は無視され、プログラムの実行が続けられます。
モード終了	WFI 命令で STOP モードに入った場合、割込みモードに設定されている任意の EXTI ライン（対応する割込みベクタが NVIC で有効とされているライン）。 表 35: ベクターテーブル (130 ページ) を参照 WFE 命令で STOP モードに入った場合、イベントモードに設定されている任意の EXTI ライン。 セクション 8.2.3: ウェークアップ・イベント管理 (135 ページ) を参照
ウェークアップ時の遅延	HSI RC ウェークアップ時間 + 低電力モードからの通常起動時間

4.3.5 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がディセーブルで CortexTM-M3 のディープスリープ (deepsleep) モードで 1.8 V ドメインも電源オフ状態となります。また、PLL、HSI オシレータ及び HSE オシレータもオフ状態とされます。SRAM とレジスタの内容はバックアップドメインのレジスタと STANDBY 回路（[図 4 参照](#)）を除いて失われます。

STANDBY モードに入るには

STANDBY モードへの入り方については[表 10](#)を参照してください。

STANDBY モードでは次の回路で、個別の制御ビットによる機能設定ができます。

- 独立型ウォッチドッグ (IWDG): IWDG の動作は、キーレジスタへの書き込み、もしくはハードウェアの設定で開始されます。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 16: 独立型ウォッチドッグ \(IWDG\) のセクション 16.3](#) を参照してください。
- リアルタイム・クロック (RTC): これはバックアップ・ドメイン・レジスタ (RCC_BDCR) の RTCEN ビットで設定されます。
- 内蔵 RC オシレータ (LSI RC): これは制御 / ステータス・レジスタ (RCC_CSR) の LSION ビットで設定されます。
- 外部 32.768 kHz オシレータ (LSE OSC): これはバックアップ・ドメイン制御レジスタ (RCC_BDCR) レジスタの LSEON ビットで設定されます。

参考資料

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 ピン信号の立上がりエッジ、もしくは RTC アラームのいずれかの発生で、STANDBY モードを終了します。[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は、STANDBY モードからの起動時にリセットされます。

STANDBY モードからのウェークアップ後、プログラムはリセット直後と同様に（ブートピン信号のサンプル、リセット・ベクタのフェッチ、など）開始されます。[電源制御／ステータス・レジスタ \(PWR_CSR\)](#) の SBF ステータス・フラグは、MCU が STANDBY モードか否かを示します。

STANDBY モードから抜け出す条件の詳細は[表 10](#)を参照してください。

表 10. STANDBY モード

STANDBY モード	摘要
モード遷移	次の条件下の WFI (Wait for Interrupt) もしくは WFE (Wait for Event) 命令 - Cortex™-M3 システム制御レジスタの SLEEPDEEPbit をセット - 電力制御レジスタ (PWR_CR) の PDSDSbit をセット - 電力制御 / ステータス・レジスタ (PWR_CSR) の WUFbit をクリア
モード終了	WKUP ピン信号の立上がりエッジ、RTC アラーム、NRST ピンの外部リセット信号、IWDG リセット
ウェークアップ時の遅延	リセット状態からの通常のスタートアップ

STANDBY モードにおける入出力の状態

STANDBY モードでは、以下の信号を除いて、すべての入出力ピンはハイ・インピーダンス状態となります。

- リセットパッド（有効）
- TAMPE ピン（タンパもしくは較正出力に設定されている場合）
- WKUP ピン（イネーブル状態の場合）

デバッグ・モード

デバッグ機能が使用されているときに、アプリケーションの動作として MCU が STOP モードもしくは STANDBY モードに入ると、デバッグ接続は失われます。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。

しかし、DBGMCU_CR レジスタの構成ビットをセットすることにより低電力モードでも使用する場合、ソフトウェアはデバッグを行うことができます。この詳細は[セクション 26.15.1: 低電力モードにおけるデバッグのサポート](#)を参照してください。

4.3.6 低電力モードからの自動ウェークアップ (AWU : Auto-wakeup)

外部割込み (AWU モード) を使用せずに、RTC を利用して MCU を低電力モードからウェークアップすることができます (Auto-Wake-Up mode)。RTC は一定の時間間隔で STOP モードもしくは STANDBY モードからウェークアップするための設定可能なタイムベース機能を提供しています。この目的のため、3 種の RTC クロックのうち、次のいずれかを **バックアップ・ドメイン制御レジスタ (RCC_BDCR)** の RTCSEL[1:0] で選択することができます。：

- 低電力 32.768 kHz 外部クリスタル・オシレータ (LSE OSC)
このクロックは低電力（標準的な使用で $1\mu A$ 以下）で高精度のタイムベースを備えています。
- 低電力内蔵 RC オシレータ (LSI RC)
このクロックを利用することで、32.768 kHz のクリスタル発振子のコストを省くことができます。内部 RC オシレータは、オシレータの消費電力が最小となるよう設計されています。

RTC アラームにより STOP モードからウェークアップするためには、次の設定が必要です。

- EXTI ライン 17 を立上がりエッジが有効となるよう設定します。
- RTC 回路を、RTC アラームを発生するよう設定します。

STnadbay モードからのウェークアップでは、EXTI ライン 17 の設定は必要とされません。

4.4 電源制御レジスタ

4.4.1 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 : 0000 0000h (STANDBY モードからのウェークアップ時にリセットされます)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み						DBP	PLS[2:0]			PVDE	CSBF	CWUF	PDDS	LPDS	
Res						rw	rw	rw	rw	rw	rc_w1	rc_w1	rw	rw	

Bits 31:9 予約済み、常に 0 が読込まれます。

Bit 8 **DBP**: バックアップ・ドメイン書き込み禁止解除

リセット時には RTC レジスタとバックアップ・レジスタは過渡状態における書き込みアクセスから保護されています。これらのレジスタへの書き込みを可能とするには、このビットをセットしなければなりません。

0: RTC とバックアップ・レジスタへのアクセスは禁止されています。

1: RTC とバックアップ・レジスタへのアクセスが可能です。

Bits 7:5 **PLS[2:0]**: PVD レベル選択

これらのビットには、電源電圧検出回路 (PVD: Power Voltage Detector) の電圧スレッショルド値をソフトウェアで設定します。

000: 2.2V

001: 2.3V

010: 2.4V

011: 2.5V

100: 2.6V

101: 2.7V

110: 2.8V

111: 2.9V

注: 電気特性の詳細に関してはデータシートを参照してください。

Bit 4 **PVDE**: PVD (Power Voltage Detector) イネーブル

このビットはソフトウェアでセット / クリアができます。

0: PVD ディセーブル

1: PVD イネーブル

Bit 3 **CSBF**: STANDBY フラグのクリア

このビットの読み出しは常に 0 が戻されます。

0: ビットの値に影響を与えません。

1: SBF (STANDBY フラグ) をクリアします。(書き込み時)

Bit 2 **CWUF**: ウェークアップ (Wake-up) フラグのクリア

このビットの読み出しは常に 0 が戻されます。

0: ビットの値に影響を与えません。

1: WUF (ウェークアップ・フラグ) を 2 システム・クロック後にクリアします。(書き込み時)

参考資料

Bit 1 PDDS: パワー・ダウン・ディープスリープ *Power Down Deepsleep.*

このビットはソフトウェアでセット / クリアできます。このビットは LPDS ビットとの組合わせで動作します。

0: CPU がディープスリープに入ると、STOP モードになります。レギュレータの状態は LPDS ビットに依存します。

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

Bit 0 LPDS: 低電力・ディープスリープ *Low-Power Deepsleep.*

このビットはソフトウェアでセット / クリアできます。このビットは PDDS ビットとの組合わせで動作します

0: 電圧レギュレータは STOP モードの間もオン状態を保ちます。

1: 電圧レギュレータは STOP モードの間は低電力モードとなります。

参考資料

RM0008

電源制御 (PWR)

4.4.2 電源制御 / ステータス・レジスタ (PWR_CSR)

アドレスオフセット : 0x04h

リセット値 : 0000 0000h (STANDBY モードからのウェークアップ時にはリセットされません)

このレジスタを読込むためには、標準 APB 読込みに対する追加の 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															
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				EWUP	予約済み				PVDO	SBF	WUF				
Res.				rw	Res.				r	r	r				

Bits 31:9 予約済み、常に 0 が読込まれます。

Bit 8 EWUP: WKUP ピン・イネーブル

このビットはソフトウェアでセット / クリアできます。

0: WKUP ピンは汎用入出力(GPIO)として使用されます。この WKUP ピンのイベントは STANDBY モードからのウェークアップをするものではありません。

1: WKUP ピンは STANDBY モードからのウェークアップに使用されます。このピンは強制的にプルダウン入力モードに設定されます。(WKUP ピンの立上がりエッジで STANDBY モードからのウェークアップが発生します。)

注: このビットはシステム・リセットでリセットされます。

Bits 7:3 予約済み、常に 0 が読込まれます。

Bit 2 PVDO: PVD 出力

このビットのセット / クリアはハードウェアが行います。このビットは PVDE ビットの設定で PVD がイネーブル状態となっているときに限り有効です。

0: V_{DD} / V_{DDA} は PLS[2:0]bit で設定されている PVD スレッショルドより高い電圧です。

1: V_{DD} / V_{DDA} は PLS[2:0]bit で設定されている PVD スレッショルドより低い電圧です。

注: PVD は STANDBY モードで停止します。このため、このビットは STANDBY 及びリセット後は PVDE ビットがセットされるまで 0 を戻します。

Bit 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) もしくは [電源制御レジスタ \(PWR_CR\)](#) の CSBF ビットに 1 を書込むことでクリアされます。

0: デバイスは STANDBY モードではありません。

1: デバイスは STANDBY モードです。

Bit 0 WUF: ウェークアップ・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) もしくは [電源制御レジスタ \(PWR_CR\)](#) の CWUF ビット 1 を書込むことでクリアされます。

0: ウェークアップ (Wake-up) イベントは発生していません。

1: WKUP ピン、もしくは RTC アラームにより、ウェークアップ (Wake-up) イベントが受付けられました。

注: WKUP ピンのレベルが常にハイの場合、WKUP ピンがイネーブル (EWUPbit で設定) されると追加のウェークアップイベントが検出されます。



参考資料

4.4.3 PWR レジスタ・マップ

次の表は PWR レジスタの一覧です。

表 11. PWR - レジスタ・マップとリセット時の値

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 リセット値	予約済み															DBP 0	PLS[2:0] 0 0 0	PVDE 0	CSBF 0	CWUF 0	PVDO 0	SBF 0	PDDS 0	WUF 0	LPDS 0							
	PWR_CSR リセット値	予約済み															EWUP 0	予約済み															

レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。

5 バックアップ・レジスタ (BKP)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

5.1 概要

バックアップ・レジスタは、84bytes のユーザアプリケーションのデータを保存する事が可能な 42 個の 16bit レジスタで構成されています。これらのレジスタは、 V_{DD} 電源が遮断された時に、 V_{BAT} により電源を維持するバックアップ・ドメインに搭載されています。これらのレジスタは、デバイスが STANDBY からウェークアップした場合や、システム・リセットまたは電源リセットされてもリセットされません。

更に、BKP 制御レジスタはタンパ検出機能や RTC 周波数調整にも使用されます。

リセット後、バックアップ・レジスタと RTC へのアクセスはディセーブル状態で、バックアップ・ドメイン (BKP) は過渡的に発生し得る予期しない書き込みアクセスから保護されています。バックアップ・レジスタや RTC へのアクセスを可能とするには、次の手順が必要です。

- RCC_APB1ENR レジスタの PWREN と BKOPEN ビットをセットし、パワーとバックアップ・インターフェース・クロックをイネーブルにします。
- 電源制御レジスタ (PWR_CR) の DBP ビットをセットし、バックアップ・レジスタと RTC へのアクセスをイネーブルにします。

5.2 主な機能

- 20bytes のデータ・レジスタ（中容量デバイス）あるいは 84bytes データ・レジスタ（大容量デバイス）
- 割込みによるタンパ検出管理のためのステータス / 制御レジスタ
- RTC 周波数の調整値を格納する較正レジスタ
- RTC 較正クロック、RTC アラームパルス、もしくはタンパピン PC13 の 2 つ目のパルス（このピンがタンパ検出に使われていない場合）の出力

5.3 機能解説

5.3.1 タンパ検出

ANTI_TAMP ピンは、[バックアップ制御レジスタ \(BKP_CCR\)](#)の TPAL ビットに応じて、信号が 0 から 1 もしくは 0 から 1 に変化したときにタンパ検出イベントを発生させます。タンパ検出イベントは、全てのバックアップ・データ・レジスタの内容をリセットします。

しかし、タンパ・イベントの見落しを防ぐため、エッジ検出に使用される信号はタンパ・イネーブル・ビットとの論理積がとられています。これにより、タンパピンがイネーブルになる前にすでに発生していたタンパ・イベントも検出することができます。

- **TPAL=0 のとき**：タンパ・ピンが TPE ビットをセットすることでイネーブルとされる前からハイレベルであれば、タンパ・ピンがイネーブルにされると同時にタンパ・イベントが発生します。(このとき、TPE がセットされた後にはタンパ・ピンの信号には立上りエッジは存在していません。)
- **TPAL=1 のとき**：タンパ・ピンが TPE ビットをセットすることでイネーブルとされる前からローレベルであれば、タンパ・ピンがイネーブルにされると同時にタンパ・イベントが発生します。(このとき、TPE がセットされた後にはタンパ・ピンの信号には立下りエッジは存在していません。)

BKP_CSR レジスタの TPIE ビットをセットすることにより、タンパ検出イベントが発生した時に割込みが発生します。

タンパ・イベントが検出され、その後にクリアされたときには、タンパ・ピンは一旦ディセーブルとし、その後バックアップ・データ・レジスタ (BKP_DRx) に書き込みを行う前に TPE ビットでイネーブルとされなければなりません。これは、タンパ検出がイネーブルとされる前にタンパ・ピンの値がタンパ検出を示しているときに、ソフトウェアがバックアップ・データ・レジスタ (BKP_DRx) への書き込みを行ってしまうのを防ぎます。これはタンパ・ピンでレベル検出を行っているのと同じ働きをもちます。

注意：タンパ検出は VDD 電源がオフのときにも有効に動作します。バックアップ・データ・レジスタの内容が不用意にリセットされることを防ぐため、ANTI_TAMP ピンは外部で適当な信号レベルに接続しておくことが必要です。

5.3.2 RTC 周波数調整

クロック周波数を測定するため、32.768 kHz RTC クロックを タンパ・ピンに出力することができます。この動作は [RTC クロック較正レジスタ \(BKP_RTCRR\)](#) の CCO ビットをセットすることで指示します。

クロック周波数は CAL[6:0] の設定で最大 121 ppm まで低下させることができます。

RTC 周波数調整及び厳密なタイミング管理方法の詳細については、AN2604 「STM32Fxx と STM32F103xx RTC 周波数調整」を参照してください。

参考資料

5.4 BKP レジスタ

レジスタの詳細で使用されている略語については[セクション 1.1 \(32 ページ\)](#) を参照してください。

5.4.1 バックアップ・データ・レジスタ x (BKP_DRx) (x = 1 ..42)

アドレス・オフセット : 0x04 ~ 0x28、0x40 ~ 0xBC

リセット値 : 0x0000 0000

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

Bits 15:0 D[15:0]: バックアップ・データ

これらのビットには任意のユーザデータを書込むことができます。

注: BKP_DRx レジスタはシステム・リセットやパワー・リセット、もしくは STANDBY モードからのウェークリードではリセットされません。

レジスタの内容はバックアップ・ドメイン・リセットもしくはタンパク・イベント（この機能が有効な場合）によってリセットされます。

参考資料

5.4.2 RTC クロック較正レジスタ (BKP_RTCCR)

アドレスオフセット : 0x2C

リセット値 : 0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
予約済み	ASOS	ASOE	CCO	CAL[6:0]													
Res.	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		

Bits 15:10 予約済み、常に 0 が読出されます。

Bit 9 ASOS アラームまたはセカンド出力選択

ASOE がセットされた時、ASOS ビットを使ってタンパピンの信号出力を RTC セカンドパルス信号にするか、アラームパルス信号にするかを選択できます。

0: RTC アラームパルス出力を選択します。

1: RTC セカンドパルス出力を選択します。

注: このビットをリセットするのは、バックアップ・ドメインのリセットだけです。

Bit 8 ASOE アラームまたはセカンド出力イネーブル

このビットをセットすることで、RTC アラームパルス信号か、ASOS ビットの設定に応じてタンパピン上のセカンドパルス信号を出力する事が出来ます。

出力パルス周期は、1RTC クロック周期です。ASOE ビットがセットされている間は、タンパピンをイネーブルにしてはいけません。

注: このビットをリセットるのは、バックアップ・ドメインのリセットだけです。

Bit 7 CCO: 較正クロック出力

0: 影響なし

1: このビットをセットすると、64 分周された RTC クロックがタンパピンから出力されます。タンパ検出の誤動作を防ぐため、CCO ビットがセットされているときにはタンパピンをイネーブルにすることはできません。

注: このビットは V_{DD} 供給がオフとなった時にはクリアされます。

Bit 6:0 CAL[6:0] 較正值

この値は、2²⁰ クロックパルス毎に無視されるクロックパルス数を示します。これにより、RTC の較正が出来、クロックを 1000000/2²⁰ ppm のステップで下げる事が出来ます。

RTC クロックは、0 ~ 121 ppm の間で下げることが出来ます。

参考資料

RM0008

バックアップ・レジスタ (BKP)

5.4.3 バックアップ制御レジスタ (BKP_CR)

アドレス・オフセット : 0x30

リセット値 : 0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み												TPAL	TPE		
Res.												rw	rw		

Bits 15:2 予約済み、常に 0 が読出されます。

Bit 1 TPAL: タンパ・ピン・アクティブ・レベル

0: タンパ・ピンのハイレベルで全バックアップ・データ・レジスタがリセットされます。
(TPE ビットがセットされている場合)

1: タンパ・ピンのローレベルで全バックアップ・データ・レジスタがリセットされます。
(TPE ビットがセットされている場合)

Bit 0 TPE: タンパ・ピン・イネーブル

0: タンパ・ピンは汎用入出力 (GPIO) として使用することができます。

1: タンパ・オルタネート入出力機能が有効となります。

注意: TPAL と TPE ビットを同時にセットすることは常に可能ですが、両方を同時にクリアすることは副作用としてタンパ・イベントを発生させてしまうことがあります。このため、TPAL ビットの変更は TPE ビットがクリアされているときに限るようにしてください。



参考資料

5.4.4 バックアップ制御 / ステータス・レジスタ (BKP_CSR)

アドレス・オフセット : 0x34

リセット値 : 0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み	TIF	TEF	予約済み		TPIE	CTI	CTE								
Res.	r	r	Res.		rw	w	w								

Bits 15:10 予約済み、常に 0 が読出されます。

Bit 9 TIF: タンパ割込みリクエスト・フラグ

このビットはタンパ・イベントが検出され、TPIE ビットがセットされているときに、ハードウェアによってセットされます。このビットは CTI ビットに 1 を書込むことでクリアされます。また、TPIE ビットをクリアしたときにもクリアされます。

0: タンパ割込みリクエストは発生しません。

1: タンパ割込みリクエストが発生しました。

注: このビットはシステム・リセット、もしくは STANDBY モードからのウェークアップでのみリセットされます。

Bit 8 TEF: タンパ・イベント・フラグ

このビットはタンパ・イベントが検出されたときに、ハードウェアでセットされます。このビットをクリアするには、CTE ビットに 1 を書込みます。

0: タンパ・イベントは発生していません。

1: タンパ・イベントが発生しました。

注: タンパ・イベントはすべての BKP_DRx レジスタをリセットします。レジスタは TEF ビットがセットされている間、リセット状態に置かれます。フラグがセットされているときに BKP_DRx レジスタへの書込みを行っても、その値は格納されません。

Bits 7:3 予約済み、常に 0 が読出されます。

Bit 2 TPIE: タンパ・ピン割込みリクエスト許可

0: タンパ割込みリクエストは禁止されています。

1: タンパ割込みリクエストは許可されています。(BKP_CR レジスタの TPE ビットはセットされていなければなりません。)

注 1: タンパ割込みは、コアを低電力モードからウェークアップすることはありません。

注 2: このビットはシステム・リセット、もしくは STANDBY モードからのウェークアップでリセットされます。

Bit 1 CTI: タンパ割込みリクエスト・クリア

このビットは書き込み専用で、読み出しでは常に 0 が戻されます。

0: 影響なし

1: タンパ割込みリクエストと TIF タンパ割込みリクエスト・フラグをクリアします。

Bit 0 CTE: タンパ・イベント・クリア

このビットは書き込み専用で、読み出しでは常に 0 が戻されます。

0: 影響なし

1: タンパ・イベント・フラグをクリアします。(同時にタンパ検出回路もリセットされます。)

參考資料

RM0008

バックアップ・レジスタ (BKP)

5.4.5 BKP レジスタ・マップ

BKP レジスタは、次の表のように、すべて 16bit のレジスタです。

表 12. BKP - レジスタ・マップとリセット時の値

参考資料

表 12. BKP - レジスタ・マップとリセット時の値 (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	BKP_DR11	予約済み	D[15:0]
	リセット値		0 0
0x44	BKP_DR12	予約済み	D[15:0]
	リセット値		0 0
0x48	BKP_DR13	予約済み	D[15:0]
	リセット値		0 0
0x4C	BKP_DR14	予約済み	D[15:0]
	リセット値		0 0
0x50	BKP_DR15	予約済み	D[15:0]
	リセット値		0 0
0x54	BKP_DR16	予約済み	D[15:0]
	リセット値		0 0
0x58	BKP_DR17	予約済み	D[15:0]
	リセット値		0 0
0x5C	BKP_DR18	予約済み	D[15:0]
	リセット値		0 0
0x60	BKP_DR19	予約済み	D[15:0]
	リセット値		0 0
0x64	BKP_DR20	予約済み	D[15:0]
	リセット値		0 0
0x68	BKP_DR21	予約済み	D[15:0]
	リセット値		0 0
0x6C	BKP_DR22	予約済み	D[15:0]
	リセット値		0 0
0x70	BKP_DR23	予約済み	D[15:0]
	リセット値		0 0
0x74	BKP_DR24	予約済み	D[15:0]
	リセット値		0 0
0x78	BKP_DR25	予約済み	D[15:0]
	リセット値		0 0
0x7C	BKP_DR26	予約済み	D[15:0]
	リセット値		0 0
0x80	BKP_DR27	予約済み	D[15:0]
	リセット値		0 0

参考資料

表 12. BKP - レジスタ・マップとリセット時の値 (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	
0x84	BKP_DR28	予約済み	D[15:0]
	リセット値		0 0
0x88	BKP_DR29	予約済み	D[15:0]
	リセット値		0 0
0x8C	BKP_DR30	予約済み	D[15:0]
	リセット値		0 0
0x90	BKP_DR31	予約済み	D[15:0]
	リセット値		0 0
0x94	BKP_DR32	予約済み	D[15:0]
	リセット値		0 0
0x98	BKP_DR33	予約済み	D[15:0]
	リセット値		0 0
0x9C	BKP_DR34	予約済み	D[15:0]
	リセット値		0 0
0xA0	BKP_DR35	予約済み	D[15:0]
	リセット値		0 0
0xA4	BKP_DR36	予約済み	D[15:0]
	リセット値		0 0
0xA8	BKP_DR37	予約済み	D[15:0]
	リセット値		0 0
0xAC	BKP_DR38	予約済み	D[15:0]
	リセット値		0 0
0xB0	BKP_DR39	予約済み	D[15:0]
	リセット値		0 0
0xB4	BKP_DR40	予約済み	D[15:0]
	リセット値		0 0
0xB8	BKP_DR41	予約済み	D[15:0]
	リセット値		0 0
0xBC	BKP_DR42	予約済み	D[15:0]
	リセット値		0 0

レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。

6 リセットとクロックの制御 (RCC)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

6.1 リセット

リセット動作には、システム・リセット、電源リセット、バックアップ・ドメイン・リセットの 3 種類のリセット動作があります。

6.1.1 システム・リセット

システム・リセットは、クロック・コントローラ CSR レジスタのリセット・フラグとバックアップ・ドメインのレジスタ（図 4 を参照）を除いた、すべてのレジスタを初期値にリセットします。

システム・リセットは次のイベントのいずれかの発生時に実行されます。

1. NRST ピン信号のローレベル（外部リセット）
2. Window 型ウォッチドッグのカウント終了（WWDG リセット）
3. 独立型ウォッチドッグのカウント終了（IWDG リセット）
4. ソフトウェア・リセット（SW リセット）（セクション：ソフトウェア・リセットを参照）
5. 低電力管理リセット（セクション：低電力管理リセットを参照）

発生したリセットの要因は制御 / ステータス・レジスタのリセット・フラグから知ることができます。（セクション 6.3.10: 制御 / ステータス・レジスタ (RCC_CSR) を参照）

ソフトウェア・リセット

デバイス上でソフトウェア・リセットを実行するには、Cortex™-M3 のアプリケーション割込みレジスタ、リセット制御レジスタの SYSRESETREQ ビットがセットする必要があります。詳細は 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発生するには、2 種類の方法があります。

1. STANDBY モードに入る際のリセット動作
このリセットはユーザ・オプション・バイトの nRST_STDBY ビットをリセットすることで有効となります。この場合、STANDBY モードに入る動作が正常に実行されると、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 モードに入る際のリセット動作
このリセットはユーザ・オプション・バイトの nRST_STOP ビットをリセットすることで有効となります。この場合、STOP モードに入る動作が正常に実行されると、STOP モードに入る代わりにデバイスがリセットされます。

ユーザ・オプション・バイトの詳細に関しては STM32F10x Flash プログラミングマニュアルを参照してください。

参考資料

6.1.2 電源リセット

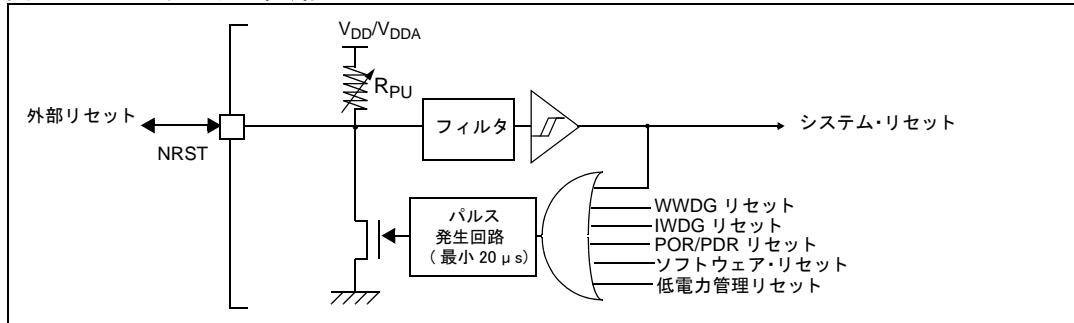
パワー・リセットは次のいずれかの場合に実行されます。

1. パワー・オン / パワー・ダウン・リセット (POR/PDR リセット)
2. STANDBY モードを抜け出す場合

パワー・リセットはバックアップ・ドメイン (図 4 を参照) 以外のすべてのレジスタを初期化します。

これらの信号は RESET ピンに対して働き、ピンを所定の時間ローレベルに保ちます。RESET サービスルーチンのベクタはメモリ・マップ上で固定された 0x0000_0000-0x0000_0004 に配置されています。詳細は [表 35: ベクタテーブル \(130 ページ\)](#) を参照してください。

図 7. リセット回路



バックアップ・ドメイン (図 4 を参照) は、このドメインにのみ有効な 2 種類のリセットがあります。

6.1.3 バックアップ・ドメイン・リセット

バックアップ・ドメインのリセットは、次のいずれかのイベントにより実行されます。

1. ソフトウェア・リセット、[バックアップ・ドメイン制御レジスタ \(RCC_BDCR\)](#) の BDRST ビットをセットすることでトリガされます。
2. V_{DD} もしくは V_{BAT} パワー・オン、ただし、両方の電源供給がともにオフ状態であった場合。

6.2 クロック

次の 3 信号が、システム・クロック (SYSCLK) の駆動源として利用できます。

- HSI オシレータ・クロック
- HSE オシレータ・クロック
- PLL クロック

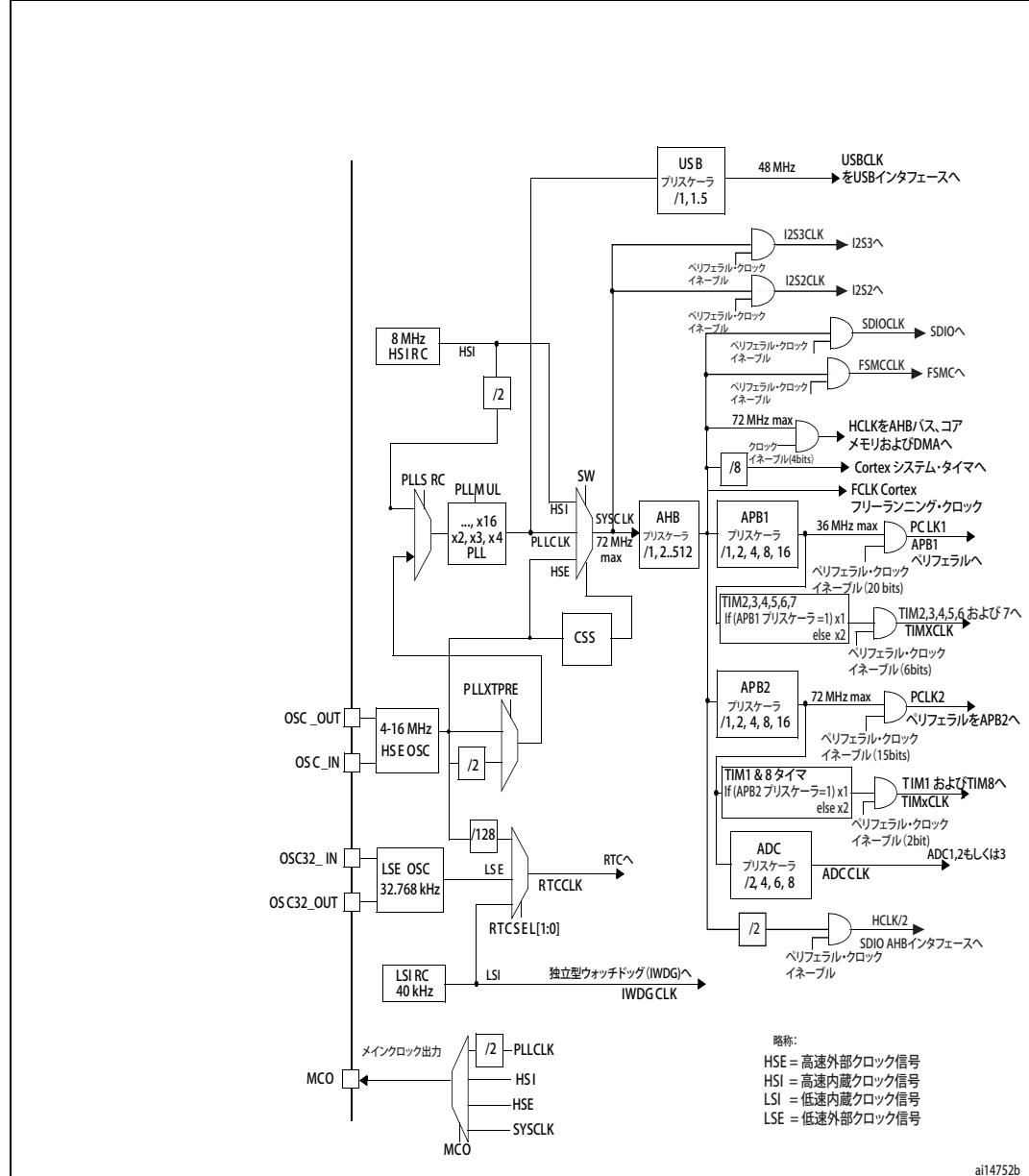
デバイスは、また、次の 2 種の信号を補助的なクロック・ソースとしてもできます。

- 40 kHz 低速内蔵 RC (LSI RC)、この信号は独立型ウォッチドッグを駆動しています。また、設定によっては STOP/STANDBY モードからの自動ウェークアップ (Auto Wake-up) に使用される RTC 回路を駆動することもできます。
- 32.768 kHz 低速外部発振子 (LSE クリスタル)、この信号は設定により、RTCCLK としてリアルタイム・クロックを駆動することができます。

それぞれのクロック・ソースは、未使用時の電力消費を省くために、個別にオン・オフが可能です。

参考資料

図 8. クロック構成図



1. HSI が PLL クロック入力として使用される場合、最大システム・クロック周波数は 64MHz です。

複数のプリスケーラを利用して、AHB 周波数、高速 APB (APB2) 及び低速 APB (APB1) ドメインの周波数を設定することができます。AHB 及び APB2 ドメインの最高周波数は 72 MHz で、APB1 ドメインの最高周波数は 36 MHz です。SDIO AHB インタフェースは HCLK/2 と等しい固定クロックで動作します。

RCC は Cortex システムタイマ (SysTick) 外部クロックに AHB クロックを 8 分周した信号を供給しています。SysTick はこの分周されたクロックか、Cortex クロック (AHB) のいずれかで動作することができ、これは SysTick 制御とステータス・レジスタで選択されます。ADC には高速 APB (APB2) クロックを 2、4、6 もしくは 8 分周したクロックが供給されます。

タイマ・クロック周波数はハードウェアで自動的に決定されます。これには、2つのケースがあります。

1. APB プリスケーラが 1 のときには、タイマの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になります。
2. その設定ではない場合、タイマ・クロックの周波数はタイマが接続されている APB ドメインのクロックの 2 倍です。

FCLK は Cortex™-M3 フリーラニングクロックとして動作します。詳細は 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。

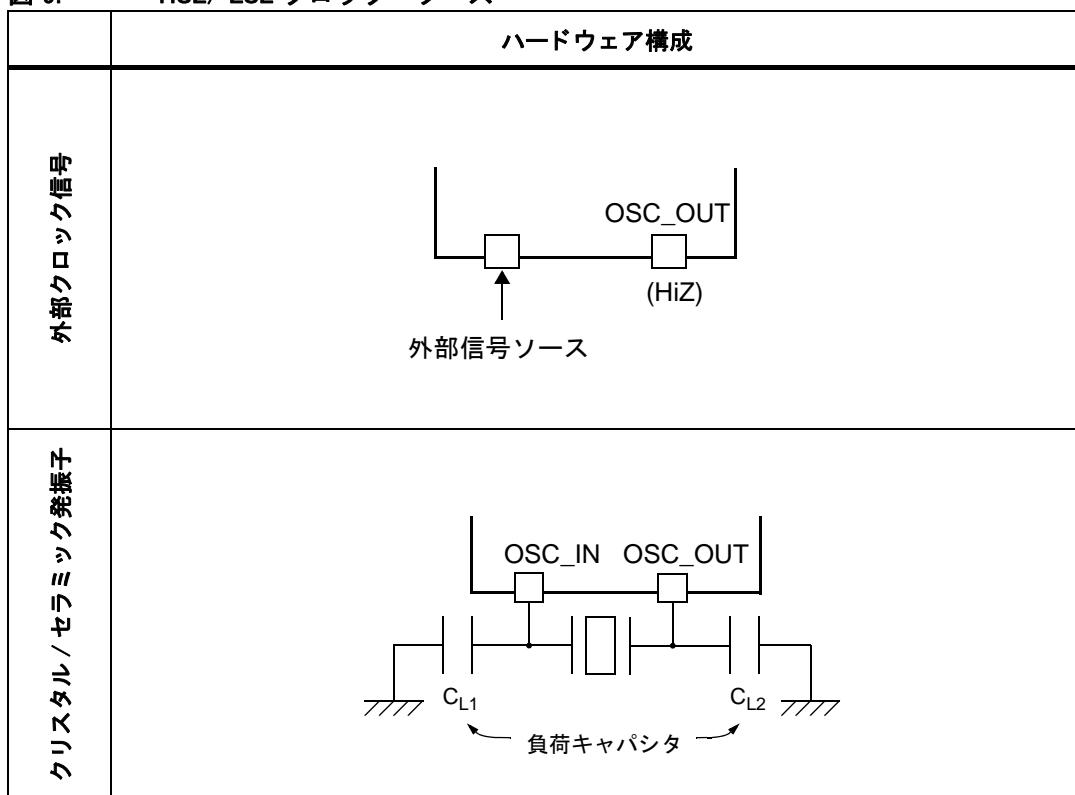
6.2.1 HSE クロック

高速外部クロック信号 (HSE) は次のいずれかの方法で生成することができます。

- HSE 外部クリスタル / セラミック発振子
- HSE 外部クロック信号

波形ひずみと発振開始時の安定動作のため、発振子と負荷キャパシタ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。同調キャパシタの容量は使用されている発振子にあわせて調整することが必要です。

図 9. HSE/ LSE クロック・ソース



外部ソース (HSE バイパス)

このモードは外部にクロック・ソースとなる 25 MHz 以下の信号が必要です。このモードは [クロック制御レジスタ \(RCC_CR\)](#) の HSEBYP ビットと HSEON ビットをセットすることで指定します。供給されるクロック信号はデューティサイクルがほぼ 50 % の矩形波、正

参考資料

弦波、三角波のいずれかで、OSC_IN ピンを駆動します。このとき OSC_OUT ピンはハイ・インピーダンスの状態とします。(図 9 を参照)

外部クリスタル / セラミック発振子 (HSE クリスタル)

4 ~ 16 MHz の発振子を外部に付加することにより、メインクロックの周波数を高精度とすることができる利点が生じます。

関連するハードウェアの構成は図 9 に示されています。詳細は、データシートの電気特性の項を参照してください。

クロック制御レジスタ (RCC_CR) の HSERDY フラグは外部オシレータが安定に動作しているか否かを示します。起動時にはこのビットがハードウェアでセットされるまで、クロック信号は送出されません。クロック割込みレジスタ (RCC_CIR) で許可されていれば、フラグのセット時に割込みリクエストが発生します。

HSE クリスタル・モードは、クロック制御レジスタ (RCC_CR) の HSEON ビットでオン・オフされます。

6.2.2 HSI クロック

HSI クロック信号は内蔵されている 8 MHz RC オシレータで生成され、直接にシステム・クロックとして、もしくは 2 分周され PLL 入力として直接使用されます。

HSI RC オシレータの利用は、低コスト（外付部品なし）でクロック信号を得られる利点があります。同時に、HSE クリスタル発振子を使用する場合に比べ、スタートアップに要する時間が短縮されますが、較正を実施していても、外部のクリスタル / セラミック発振子を使用する場合に比べて周波数の精度は劣ります。

周波数の調整（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、 $T_A=25^\circ\text{C}$ 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、クロック制御レジスタ (RCC_CR) の HSICAL[7:0] にロードされます。

RC オシレータの周波数に影響を与える電圧や温度の環境で利用されるときには、クロック制御レジスタ (RCC_CR) の HSITRIM[4:0] フィールドを使用して、アプリケーションで HSI 周波数の調整を行うことができます。

クロック制御レジスタ (RCC_CR) の HSIRDY フラグは HSI RC オシレータが安定状態か否かを示します。HSI RC 出力はこのフラグがハードウェアでセットされるまで HIS RC 信号は出力されません。

HSI RC 回路はクロック制御レジスタ (RCC_CR) の HSION ビットにより、オン・オフを切り替えることができます。

HSI 信号は HSE クリスタル・オシレータの動作が異常の場合はバックアップ（補助クロック）として使用することもできます。これに関してはセクション 6.2.7: クロック・セキュリティシステム (CSS) (78 ページ) を参照してください。

6.2.3 PLL 回路

HSI RC 出力や HSE クリスタル出力のクロック周波数を遙倍するために、内蔵されている PLL 回路を利用することができます。これに関しては図 8 とクロック制御レジスタ (RCC_CR) を参照してください。

PLL の設定 (PLL 入力クロックが HSI 2 分周信号もしくは HSE 信号かの選択、及び遅倍数) は PLL 回路がイネーブルになる前に行う必要があります。PLL 回路がイネーブルになると、PLL のパラメータは変更できません。

[クロック割込みレジスタ \(RCC_CIR\)](#) で許可されていれば、PLL が安定した動作状態に入ったときに割込みリクエストが生成されます。

アプリケーションで USB インタフェースが使用されるときには、48 MHz の USBCLK を得るために、PLL 出力は 48 MHz もしくは 72 MHz に設定されなければなりません。

6.2.4 LSE クロック

LSE クリスタルは 32.768 kHz 低速外部クリスタル / セラミック発振子を使用します。このクロックは、時計 / カレンダ、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 (RTC) に、低電力で高精度のクロック信号を供給できます。

LSE クリスタルは[バックアップ・ドメイン制御レジスタ \(RCC_BDCR\)](#) の LSEON ビットでオン・オフを切り替えることができます。

[バックアップ・ドメイン制御レジスタ \(RCC_BDCR\)](#) の LSERDY フラグは LSE オシレータが安定して動作しているか否かを表示しています。起動時に LSE クリスタル出力 クロック信号は、ハードウェアがこのビットをセットするまでは出力されません。[クロック割込みレジスタ \(RCC_CIR\)](#) で許可されている場合、フラグがセットされたときに割込みリクエストが発生します。

外部クロック・ソース (LSE バイパス)

このモードは外部にクロック・ソースとなる 32.768 kHz の信号が必要です。このモードは[バックアップ・ドメイン制御レジスタ \(RCC_BDCR\)](#) の LSEBYP ビットと LSEON ビットをセットすることで指定します。供給される外部クロック信号はデューティサイクルがほぼ 50 % の矩形波、正弦波、三角波のいずれかで、OSC_32IN ピンを駆動します。このとき OSC_32OUT ピンはハイ・インピーダンスの状態になります。(図 9 を参照)

6.2.5 LSI クロック

LSI RC は、独立型ウォッチドッグ (IWDG) や自動ウェークアップ・ユニット (AWU) のために STOP モード及び STANDBY モードで動作する低電力のクロック・ソースとして動作します。クロック周波数は約 40 kHz 前後 (30 kHz ~ 60kHz) になります。詳細はデータシートの電気特性の項をご参照してください。

LSI RC 回路は[制御 / ステータス・レジスタ \(RCC_CSR\)](#) の LSION ビットでオン・オフが切り替えられます。

[制御 / ステータス・レジスタ \(RCC_CSR\)](#) の LSIRDY フラグは内蔵オシレータが安定して動作しているか否かを表示しています。起動時、このクロックはハードウェアがフラグをセットするまでは、クロック信号は出力されません。[クロック割込みレジスタ \(RCC_CIR\)](#) で許可されている場合、フラグセット時に割込みリクエストが発生します。

注意 : LSI の較正は、大容量デバイスに限り利用可能です。

LSI 周波数調整 (較正)

低速内部 RC (LSI) 発振子の周波数のばらつきは、RTC 基準時間及び / もしくは妥当な精度の IWDG タイムアウト (これらのペリフェラルに LSI がクロック・ソースとして使われる時) にたいして較正する事が出来ます。

この較正は、TIM5 入力クロック (TIM5CLK) に対して LSI クロックを計測することで行われます。HSE 発振子の精度で行われるこの計測は、正確な基準時間を得る、または正確な IWDG タイムアウトを計算する為に、ソフトウェアは RTC の 20 ビットプリスケーラを調整する事が出来ます。

LSI を較正するには、次の手順に従ってください。

1. TIM5 タイマをイネーブルし、チャネル 4 を入力キャプチャ・モードに設定する。
2. 較正用に LSI クロックを内部で TIM5 チャネル 4 入力キャプチャへ接続するために、AFIO_MAPR レジスタの TIM5CH4_IREMAP ビットをセットする。
3. TIM5 キャプチャ / 比較 4 イベントあるいは割込みを使って、LSI クロックを計測する。
4. 必要な基準時間及び / もしくは、IWDG タイムアウトの計算により、RTC の 20bit プリスケーラを更新するために、計測した LSI 周波数を使う。

6.2.6 システム・クロック (SYSCLK) の選択

システム・リセットの直後は HSI オシレータがシステム・クロックとして選択されています。クロック・ソースが直接、もしくは 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・ソースに切り替えるときには、目的のクロック信号が動作中（スタートアップ遅延後にクロックが安定しているか、もしくは PLL がロックされている状態）でなければなりません。クロックが動作状態でなければ、このクロックの動作が安定するのを待って切り替えが行われます。クロック制御レジスタ (*RCC_CR*) のステータス・ビットは個々のクロックが動作状態であるか否かと、どのクロックがシステム・クロックとして利用可能であるかを示しています。

6.2.7 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で有効にできます。このとき、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の動作が開始され、オシレータが停止すると検出回路もディセーブル状態となります。

HSE オシレータのクロック信号に異常が検出されたときには、オシレータは自動的に停止し、クロック異常イベントが高機能制御タイマ TIM1 のブレーク入力に送られます。また、ソフトウェアに異常を通知し、割込みリクエスト（クロック・セキュリティシステム割込み (CSSI)）が生成され MCU が対応処理を行うことが許可されます。CSSI は Cortex™-M3 NMI (Non-Maskable Interrupt) 例外ベクタと対応しています。

注意：

CSS がイネーブルとなり、HSE クロック異常が発生すると、CSS 割込みがリクエストされ、NMI が自動的に発生します。NMI 動作は CSS 割込みペンドイングビットがクリアされない限り、無条件に実行します。つまり、NMI ISR の中でユーザがクロック割込みレジスタ (*RCC_CIR*) の CSSC ビットをセットし、CSS 割込みをクリアしなければなりません。

HSE オシレータが直接もしくは間接的（この信号が PLL 入力で、PLL 出力がシステム・クロックとして使用されている場合）にシステム・クロックとして使用されているときには、異常検出によりシステム・クロックは HSI オシレータに切り替わり、外部 HSE 発振回路はディセーブルとなります。HSE オシレータのクロックが PLL 入力で、PLL 出力がシステム・クロックとして使用されているときに異常が発生すると、PLL 回路もディセーブルになります。

6.2.8 RTC クロック

RTCCLK クロックは HSE/128、LSE、もしくは LSI クロックのいずれかから供給されます。これは [バックアップ・ドメイン制御レジスタ \(RCC_BDCR\)](#) の RTCSEL[1:0] ビットで選択ができます。この選択はバックアップ・ドメインをリセットしない限り変更することはできません。

LSE クロックはバックアップ・ドメインに属しますが、HSE と LSI クロックはそうではありません。このため、

- LSE が RTC クロックとして使われているときには、
 - V_{DD} 供給がオフとなっても V_{BAT} の供給が保たれている限り、RTC は動作を続けます。
- LSI が自動ウェークアップ (AWU) クロックとして使われているときには、
 - V_{DD} 供給がオフとなったとき、AWU の状態は保証されません。LSI 較正の詳細については [セクション 6.2.5: LSI クロック \(77 ページ\)](#) を参照してください。
- 128 分周した HSE クロックが RTC クロックとして使われているときには、
 - V_{DD} 供給がオフとなったとき、もしくは内蔵されている電圧レギュレータがオフとなり 1.8 V ドメインの電源供給が停止したときは、RTC の状態は保証されません。

6.2.9 ウオッチドッグ・クロック

独立型ウォッチドッグ (IWDG) がハードウェアのオプションとして、もしくはソフトウェアによって開始されたとき、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 オシレータの立上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。

6.2.10 クロック信号出力

マイクロコントローラ・クロック出力機能 (MCO) は、クロック信号を外部の MCO ピンに出力します。このとき、対応する GPIO ポートの設定レジスタはオルタネート機能が選択する必要があります。以下の 4 つの信号の一つが、MCU のクロックとして選択することができます。

- SYSCLK
- HSI
- HSE
- 2 分周された PLL クロック

出力されるクロックは [クロック構成レジスタ \(RCC_CFGR\)](#) の MCO[2:0] ビットで指定します。

参考資料

6.3 RCC レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#) 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 : 0x0000 XX83 X は不定

アクセス : ノーウェイト、ワード・ハーフワード・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				PLL RDY	PLLON	予約済み				CSS ON	HSE BYP	HSE RDY	HSE ON		
予約済み				r	rw	予約済み				rw	rw	r	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSICAL[7:0]								HSITRIM[4:0]				予約済み	HSI RDY	HSION	
r	r	r	r	r	r	r	r	rw	rw	rw	rw	r	rw		

Bits 31:26 予約済み、常に 0 が読出されます。

Bit 25 PLLRDY: PLL クロック・レディ・フラグ

ハードウェアでセットされ、PLL がロック状態であることを示しています。

0: PLL はロック状態ではありません。

1: PLL はロック状態です。

Bit 24 PLLON: PLL イネーブル

PLL をイネーブルとするため、ソフトウェアでセット / リセットができます。

STOP モード、STANDBY モードに入ると、ハードウェアによってクリアされます。このビットは PLL 出力がシステム・クロックとして使用されているとき、もしくはこれからシステム・クロックとして使用されようとしているときにはリセットできません。

0: PLL オフ

1: PLL オン

Bits 23:20 予約ビット、常に 0 が読出されます。

Bit 19 CSSON: クロック・セキュリティシステム・イネーブル

クロック検出回路をイネーブルとするため、ソフトウェアでセット / リセットができます。

0: クロック検出回路オフ

1: クロック検出回路オン (1-25MHz HSE オシレータの動作時のみ)

Bit 18 HSEBYP: 高速外部 (HSE) クロック・バイパス・モード

デバッグ時に、オシレータをバイパスして外部からのクロック信号の利用を指示するために、ソフトウェアでセット / リセットができます。このビットは 外部 1 ~ 25MHz オシレータがディセブルの時にのみ書込むことができます。

0: 外部 1 ~ 25MHz オシレータはバイパスされません。

1: 外部 1 ~ 25MHz 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。

参考資料

Bit 17 HSERDY: 高速外部 (HSE) クロック・レディ・フラグ

外部 1 ~ 25MHz オシレータが安定した動作状態であるか否かを示すよう、ハードウェアでセットされます。HSEON がリセットされてからフラグがクリアされるまで、外部 1 ~ 25MHz オシレータのクロックは 6 サイクル相当の時間を必要になります。

0: 外部 1 ~ 25MHz オシレータはレディ状態ではありません。

1: 外部 1 ~ 25MHz オシレータはレディ状態です。

Bit 16 HSEON: 高速外部 (HSE) クロック・イネーブル

ソフトウェアでセット / リセットされます。

STOP モード、STANDBY モードに入り 外部 1 ~ 25MHz が停止するとハードウェアによりリセットされます。外部 1 ~ 25MHz が直接、もしくは間接的にシステム・クロックとして使用されているとき、またはこれから使用されようとしているときには、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

Bits 15:8 HSICAL[7:0]: 高速内蔵 (HSI) クロック・トリミング

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。

Bits 7:3 HSITRIM[4:0]: 高速内蔵 (HSI) クロック・トリミング

これらのビットは、ユーザがプログラム可能なトリミング値を HSICAL[7:0] ビットに追加で書込む事が出来ます。これは電圧や温度の変化に対応するためにプログラムで変更することができ、内蔵 HSI RC の周波数に影響します。

デフォルト値は 16 で HSICAL 値を追加した場合は HSI を 8MHz ± 1% にトリムする必要があります。

Bit 2 予約済み、常に 0 が読出されます。

Bit 1 HSIRDY: 高速内蔵 (HSI) クロック・レディ・フラグ

内蔵 8 MHz RC オシレータが安定した動作状態であるか否かを示すために、ハードウェアでセットされます。HSION がリセットされてからフラグがクリアされるまで、内蔵 8MHz RC オシレータのクロックで 6 サイクル相当の時間を必要になります。

0: 内蔵 8 MHz RC オシレータはレディ状態ではありません。

1: 内蔵 8 MHz RC オシレータはレディ状態です。

Bit 0 HSION: 高速内蔵 (HSI) クロック・イネーブル

ソフトウェアでセット / リセットされます。

STOP モード、STANDBY モードを終了したとき、もしくは 外部 1 ~ 25MHz オシレータの クロックが直接 / 間接的にシステム・クロックとして使用され、この クロックに異常が検出されたときに、内蔵 8 MHz RC 回路を強制的にオンとするためにハードウェアによりセットされます。内蔵 8 MHz RC クロックが直接、もしくは間接的にシステム・クロックとして使用されているとき、またはこれから使用されようとしているときには、このビットをリセットできません。

0: 内蔵 8 MHz RC オシレータ OFF

1: 内蔵 8 MHz RC オシレータ ON

参考資料

6.3.2 クロック構成レジスタ (RCC_CFGR)

アドレスオフセット : 0x04

リセット値 : 0000 0000h

アクセス : 0 <= ウエイト数 <= 2、ワード・ハーフワード・バイトアクセス

クロック・ソースの切り替え中のアクセスに限り、1ないし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				MCO[2:0]				予約済み	USB PRE	PLLMUL[3:0]				PLL XTPRE	PLL SRC
予約済み				rw	rw	rw	予約済み	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC PRE[1:0]		PPRE2[2:0]			PPRE1[2:0]			HPRE[3:0]				SWS[1:0]		SW[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r	rw	rw

Bits 31:26 予約済み、常に 0 が読出されます。

Bits 26:24 MCO: マイクロコントローラクロック出力

ソフトウェアでセット / リセットができます。

0xx: クロック出力なし。

100: システム・クロックが出力されます。

101: 内蔵 8 MHz RC オシレータ (HSI) クロックが出力されます。

110: 外部 1–25 MHz オシレータ (HSE) クロックが出力されます。

111: 2 分周された PLL クロックが出力されます。

注:

- このクロック出力は、スタートアップ時もしくは MCO クロック・ソースがスイッチされる間に、サイクルを短縮することができます。
- MCO ピンにシステム・クロックが出力される場合、その周波数が最大入出力速度の 50 MHz を超えないよう、注意してください。

Bit 22 USBPRE: USB プリスケーラ

48 MHz の USB クロックを生成するために、ソフトウェアでセット / リセットができます。このビットは RCC_APB1ENR レジスタで USB クロックがイネーブルとされる前に有効な値がセットされていなければなりません。また、USB クロックがイネーブル状態のときに、このビットをリセットすることはできません。

0: PLL クロックは 1.5 分周されます。

1: PLL クロックは分周されません。

参考資料

Bits 21:18 PLLMUL: PLL 適倍数

これらのビットには、PLL の適倍数の指定をソフトウェアで書込みます。書込みは PLL がディセーブル状態のときに限られます。

注意 : PLL 出力周波数は 72 MHz を超えることはできません。

- 0000: PLL 入力周波数 × 2
- 0001: PLL 入力周波数 × 3
- 0010: PLL 入力周波数 × 4
- 0011: PLL 入力周波数 × 5
- 0100: PLL 入力周波数 × 6
- 0101: PLL 入力周波数 × 7
- 0110: PLL 入力周波数 × 8
- 0111: PLL 入力周波数 × 9
- 1000: PLL 入力周波数 × 10
- 1001: PLL 入力周波数 × 11
- 1010: PLL 入力周波数 × 12
- 1011: PLL 入力周波数 × 13
- 1100: PLL 入力周波数 × 14
- 1101: PLL 入力周波数 × 15
- 1110: PLL 入力周波数 × 16
- 1111: PLL 入力周波数 × 16

Bit 17 PLLXTPRE: PLL 入力に対する HSE 分周

PLL 入力信号として使用される HSE の分周比を設定するため、ソフトウェアでセット / リセットします。この設定は PLL がディセーブル状態の時にのみ書込みが可能です。

- 0: HSE クロックは分周されません。(直に入力されます。)
- 1: HSE クロックは 2 分周されます。

Bit 16 PLLSRC: PLL 入力クロック・ソース

PLL クロック・ソースを選択するため、ソフトウェアでセット / リセットができます。このビットは PLL がディセーブル状態のときにのみ書込みが可能です。

- 0: HSI クロック / 2 が PLL 入力となります。
- 1: HSE クロックが PLL 入力となります。

Bits 14:14 ADCPRE: ADC プリスケーラ

ソフトウェアで ADC のクロック周波数を選択します。

- 00: PLCK2 2 分周
- 01: PLCK2 4 分周
- 10: PLCK2 6 分周
- 11: PLCK2 8 分周

Bits 13:11 PPREG: APB 高速 プリスケーラ (APB2)

ソフトウェアで APB 高速クロック分周を制御するためにセット / リセットができます。

- 0xx: HCLK は分周されません (= HCLK)
- 100: HCLK 2 分周
- 101: HCLK 4 分周
- 110: HCLK 8 分周
- 111: HCLK 16 分周

参考資料

Bits 10:8 **PPRE1: APB 低速プリスケーラ (APB1)**

ソフトウェアで APB 低速 のクロック分周を制御するためのセット / リセットができます。

- 0xx: HCLK は分周しません。
- 100: HCLK 2 分周
- 101: HCLK 4 分周
- 110: HCLK 8 分周
- 111: HCLK 16 分周

Bits 7:4 **HPRE: AHB プリスケーラ**

ソフトウェアで AHB のクロック分周を制御するためのセット / リセットができます。

0xxx: SYSCLK は分周しません。 (= SYSCLK)

- 1000: SYSCLK 2 分周
- 1001: SYSCLK 4 分周
- 1010: SYSCLK 8 分周
- 1011: SYSCLK 16 分周
- 1100: SYSCLK 64 分周
- 1101: SYSCLK 128 分周
- 1110: SYSCLK 256 分周
- 1111: SYSCLK 512 分周

注: AHB クロックで 1 とは異なるプリスケーラを使う場合、プリフェッч・バッファは保持する必要があります。詳細については [Flash メモリの読み出し \(41 ページ\)](#) を参照してください。

Bits 3:2 **SWS: システム・クロック・スイッチ・ステータス**

どの信号がシステム・クロックを生成するために使われているかを表示するため、ハードウェアによってセット / リセットができます。

- 00: HSI オシレータがシステム・クロックとして使われています。
- 01: HSE オシレータがシステム・クロックとして使われています。
- 10: PLL がシステム・クロックとして使われています。
- 11: 未使用

Bits 1:0 **SW: システム・クロック・スイッチ**

ソフトウェアで、SYSCLK のソースとして使われる信号をセット / リセットします。

STOP モード、STANDBY モードを終了する時、もしくは HSE 発信回路が直接 / 間接的にシステム・クロックとして使用されており、この信号に異常が検出されたときに、ハードウェアによって値がセットされます。(後者はクロック・セキュリティシステムが有効な場合のみ)

- 00: HSI がシステム・クロックとして使用されます。
- 01: HSE がシステム・クロックとして使用されます。
- 10: PLL がシステム・クロックとして使用されます。
- 11: 設定禁止

参考資料

6.3.3 クロック割込みレジスタ (RCC_CIR)

アドレスオフセット : 0x08

リセット値 : 0000 0000h

アクセス : ノーウェイト、ワード・ハーフワード・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	予約済み		PLL RDYC	HSE RDYC	HSI RDYC	LSE RDYC	LSI RDYC
予約済み								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	PLL RDYIE	HSE RDYIE	HSI RDYIE	LSE RDYIE	LSI RDYIE	CSSF	予約済み		PLL RDYF	HSE RDYF	HSI RDYF	LSE RDYF	LSI RDYF		
予約済み	rw	rw	rw	rw	rw	r			r	r	r	r	r		r

Bits 31:24 予約済み、常に 0 が読出されます。

Bit 23 **CSSC** クロック・セキュリティシステム割込みクリア

このビットをソフトウェアでセットすることで、CSSF フラグがクリアされます。フラグ・クリア後に、このビットはハードウェアでリセットされます。

- 0: CSSF はクリアされません。
- 1: CSSF がクリアされます。

Bits 22:21 予約済み、常に 0 が読出されます。

Bit 20 **PLLRDYC: PLL** レディ割込みクリア

ソフトウェアでセットすることで、PLLRDYF フラグがクリアされます。フラグ・クリア後、このビットはハードウェアでリセットされます。

- 0: PLLRDYF はクリアされません。
- 1: PLLRDYF がクリアされます。

Bit 19 **HSERDYC: HSE** レディ割込みクリア

ソフトウェアでセットすることで、HSERDYF フラグがクリアされます。フラグ・クリア後、このビットはハードウェアでリセットされます。

- 0: HSERDYF はクリアされません。
- 1: HSERDYF がクリアされます。

Bit 18 **HSIRDYC: HSI** レディ割込みクリア

ソフトウェアでセットすることで、HSIRDYF フラグがクリアされます。フラグ・クリア後、このビットはハードウェアでリセットされます。

- 0: HSIRDYF はクリアされません。
- 1: HSIRDYF がクリアされます。

Bit 17 **LSERDYC: LSE** レディ割込みクリア

ソフトウェアでセットすることで、LSERDYF フラグがクリアされます。フラグ・クリア後、このビットはハードウェアでリセットされます。

- 0: LSERDYF はクリアされません。
- 1: LSERDYF がクリアされます。

参考資料

Bit 16 **LSIRDYC: LSI レディ割込みクリア**

ソフトウェアでセットすることで、LSIRDYF フラグがクリアされます。フラグをクリア後、このビットはハードウェアでリセットされます。

- 0: LSIRDYF はクリアされません。
- 1: LSIRDYF がクリアされます。

Bits 15:13 予約済み、常に 0 が読出されます。

Bit 12 **PLLRDYIE: PLL レディ割込みリクエストクリア**

PLL ロックにより発生する割込みリクエストを許可 / 禁止するために、ソフトウェアでセット / リセットができます。

- 0: PLL ロック割込みリクエスト禁止
- 1: PLL ロック割込みリクエスト許可

Bit 11 **HSERDYIE: HSE レディ割込みリクエスト許可**

外部 1 ~ 25MHz オシレータの安定動作により発生する割込みリクエストを許可 / 禁止するため、ソフトウェアでセット / リセットができます。

- 0: HSE レディ割込みリクエスト禁止
- 1: HSE レディ割込みリクエスト許可

Bit 10 **HSIRDYIE: HSI レディ割込みリクエスト許可**

内部 8 MHz RC オシレータの安定動作により発生する割込みリクエストを許可 / 禁止するため、ソフトウェアでセット / リセットができます。

- 0: HSI レディ割込みリクエスト禁止
- 1: HSI レディ割込みリクエスト許可

Bit 9 **LSERDYIE: LSE レディ割込みリクエスト許可**

外部 32 kHz オシレータの安定動作により発生する割込みリクエストを許可 / 禁止するため、ソフトウェアでセット / リセットができます。

- 0: LSE レディ割込みリクエスト禁止
- 1: LSE レディ割込みリクエスト許可

Bit 8 **LSIRDYIE: LSI レディ割込みリクエスト許可**

内蔵 RC 40kHz オシレータの安定動作により発生する割込みリクエストを許可 / 禁止するため、ソフトウェアでセット / リセットができます。

- 0: LSI レディ割込みリクエスト禁止
- 1: LSI レディ割込みリクエスト許可

Bit 7 **CSSF: クロック・セキュリティシステム割込みリクエスト・フラグ**

外部 1 ~ 25 MHz オシレータで異常が検出されたときにハードウェアでセットされ、ソフトウェアにより CSSC ビットに 1 を書込むことでリセットされます。

- 0: HSE クロック異常によるクロックセキュリティ割込みは発生していません。
- 1: HSE クロック異常によるクロックセキュリティ割込みが発生しました。

Bits 6:5 予約済み、常に 0 が読出されます。

Bit 4 **PLLRDYF: PLL レディ割込みリクエスト・フラグ**

PLLRDYIE ビットがセットされているとき、PLL がロック状態となるとハードウェアでセットされ、ソフトウェアで PLLRDYC ビットに 1 を書込むことでリセットされます。

- 0: PLL レディ割込みリクエストは発生していません。
- 1: PLL レディ割込みリクエストが発生しました。

Bit 3 **HSERDYF: HSE レディ割込みリクエスト・フラグ**

HSERDYIE ビットがセットされて、外部低速クロックの動作が安定するとハードウェアによってセットされ、ソフトウェアで HSERDYC ビットに 1 を書込むことでリセットされます。

- 0: HDE (1 ~ 25 MHz オシレータ) レディ割込みリクエストは発生していません。
- 1: HDE (1 ~ 25 MHz オシレータ) レディ割込みリクエストが発生しました。

参考資料

Bit 2 HSIRDYF: HSI レディ割込みリクエスト・フラグ

HSIRDYIE ビットがセットされ、内部高速クロックの動作が安定するとハードウェアによってセットされ、ソフトウェアで HSIRDYC ビットに 1 を書込むことでリセットされます。

- 0: HSI (内蔵 8 MHz RC オシレータ) レディ割込みリクエストは発生していません。
- 1: HSI (内蔵 8 MHz RC オシレータ) レディ割込みリクエストが発生しました。

Bit 1 LSERDYF: LSE レディ割込みリクエスト・フラグ

LSERDYIE ビットがセットされ、外部高速クロックの動作が安定するとハードウェアによってセットされ、ソフトウェアで LSERDYC ビットに 1 を書込むことでリセットされます。

- 0: LSE (外部 32 kHz オシレータ) レディ割込みリクエストは発生していません。
- 1: LSE (外部 32 kHz オシレータ) レディ割込みリクエストが発生しました。

Bit 0 LSIRDYF: LSI レディ割込みリクエスト・フラグ

LSIRDYIE ビットがセットされ、内部低速クロックの動作が安定するとハードウェアによってセットされ、ソフトウェアで LSIRDYC ビットに 1 を書込むことでリセットされます。

- 0: LSI (内蔵 32 kHz オシレータ) レディ割込みリクエストは発生していません。
- 1: LSI (内蔵 32 kHz オシレータ) レディ割込みリクエストが発生しました。

6.3.4 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 : 0x0C

リセット値 : 0000 0000h

アクセス : ノーウェイト、ワード・ハーフワード・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															
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3 RST	USART1 RST	TIM8 RST	SPI1 RST	TIM1 RST	ADC2 RST	ADC1 RST	IOPG RST	IOPF RST	IOPE RST	IOPD RST	IOPC RST	IOPB RST	IOPA RST	予約済み	AFIO RST
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	予約済み	rw

Bits 31:16 予約済み、常に 0 が読出されます。

Bit 15 ADC3RST ADC3 インタフェース・リセット

ソフトウェアでセット / リセットができます。

- 0: 影響なし
- 1: ADC3 をリセットします。

Bit 14 USART1RST: USART1 リセット

ソフトウェアでセット / リセットができます。

- 0: 影響なし
- 1: USART1 をリセットします。

Bit 13 TIM8RST TIM8 タイマ・リセット

ソフトウェアでセット / リセットができます。

- 0: 影響なし
- 1: TIM8 をリセットします。



参考資料

Bit 12 SPI1RST: SPI 1 リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: SPI 1 をリセットします。

Bit 11 TIM1RST: TIM1 タイマ・リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: TIM1 タイマをリセットします。

Bit 10 ADC2RST: ADC2 インタフェース・リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: ADC2 インタフェースをリセットします。

Bit 9 ADC1RST: ADC1 インタフェース・リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: ADC1 インタフェースをリセットします。

Bit 8 IOPGRST 入出力ポート G リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: 入出力ポート G をリセットします。

Bit 7 IOPFRST 入出力ポート F リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: 入出力ポート F をリセットします。

Bit 6 IOPERST: 入出力ポート E リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: 入出力ポート E をリセットします。

Bit 5 IOPDRST 入出力ポート D リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: 入出力ポート D をリセットします。

Bit 4 IOPCRST 入出力ポート C リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: 入出力ポート C をリセットします。

Bit 3 IOPBRST 入出力ポート B リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: 入出力ポート B をリセットします。

Bit 2 IOPARST: 入出力ポート A リセット

ソフトウェアでセット / リセットができます。
0: 影響なし
1: 入出力ポート A をリセットします。

Bit 1 予約済み、常に 0 が読出されます。

参考資料

Bit 0 **AFIORST:** オルタネート機能入出力リセット
ソフトウェアでセット / リセットができます。
0: 影響なし
1: オルタネート機能ポートをリセットします。

6.3.5 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 : 0x10
リセット値 : 0000 0000h

アクセス : ノーウェイト、ワード・ハーフワード・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	DAC RST	PWR RST	BKP RST	予約済み	CAN RST	予約済み	USB RST	I2C2 RST	I2C1 RST	UART 5 RST	UART 4 RST	USART 3 RST	USART 2 RST	予約済み	
予約済み	rw	rw	rw	予約済み	rw	予約済み	rw	rw	rw	rw	rw	rw	rw	rw	予約済み
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 RST	SPI2 RST	予約済み	WWDRST		予約済み					TIM7 RST	TIM6 RST	TIM5 RST	TIM4 RST	TIM3 RST	TIM2 RST
rw	rw	予約済み	rw		予約済み					rw	rw	rw	rw	rw	rw

Bits 31:30 予約済み、常に 0 が読出されます。

Bit 29 **DACRST:** DAC インタフェース・リセット
ソフトウェアでセット / リセットができます。
0: 影響なし
1: DAC インタフェースをリセットします。

Bit 28 **PWRRST:** 電源インターフェース・リセット
ソフトウェアでセット / リセットができます。
0: 影響なし
1: 電源インターフェースをリセットします。

Bit 27 **BKPRST:** バックアップ・インターフェース・リセット
ソフトウェアでセット / リセットができます。
0: 影響なし
1: バックアップ・インターフェースをリセットします。

Bit 26 予約済み、常に 0 が読出されます。

Bit 25 **CANRST:** CAN リセット
ソフトウェアでセット / リセットができます。
0: 影響なし
1: CAN をリセットします。

Bit 24 予約済み、常に 0 が読出されます。

Bit 23 **USBRST:** USB リセット
ソフトウェアでセット / リセットができます。
0: 影響なし
1: USB をリセットします。

参考資料

Bit 22 **I2C2RST: I₂C 2 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: I₂C2 をリセットします。

Bit 21 **I2C1RST: I₂C1 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: I₂C1 をリセットします。

Bit 20 **USART5RST: USART5 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: USART5 をリセットします。

Bit 19 **USART4RST: USART4 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: USART4 をリセットします。

Bit 18 **USART3RST: USART3 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: USART3 をリセットします。

Bit 17 **USART2RST: USART2 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: USART2 をリセットします。

Bits 16 予約済み、常に 0 が読出されます。

Bit 15 **SPI3RST: SPI 3 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: SPI 3 をリセットします。

Bit 14 **SPI2RST: SPI 2 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: SPI 2 をリセットします。

Bits 13:12 予約済み、常に 0 が読出されます。

Bit 11 **WWDGRST: Window 型ウォッチドッグ・リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: Window 型ウォッチドッグをリセットします。

Bits 10:6 予約済み、常に 0 が読出されます。

Bit 5 **TIM7RST: タイマ 7 リセット**

ソフトウェアでセット / リセットができます。
0: 影響なし
1: タイマ 7 をリセットします。

参考資料

RM0008

リセットとクロックの制御 (RCC)

Bit 4 TIM6RST: タイマ 6 リセット

ソフトウェアでセット / リセットができます。

0: 影響なし

1: タイマ 6 をリセットします。

Bit 3 TIM5RST: タイマ 5 リセット

ソフトウェアでセット / リセットができます。

0: 影響なし

1: タイマ 5 をリセットします。

Bit 2 TIM4RST: タイマ 4 リセット

ソフトウェアでセット / リセットができます。

0: 影響なし

1: タイマ 4 をリセットします。

Bit 1 TIM3RST: タイマ 3 リセット

ソフトウェアでセット / リセットができます。

0: 影響なし

1: タイマ 3 をリセットします。

Bit 0 TIM2RST: タイマ 2 リセット

ソフトウェアでセット / リセットができます。

0: 影響なし

1: タイマ 2 をリセットします。

6.3.6 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 : 0x14

リセット値 : 0000 0014h

アクセス : ノーウェイト、ワード・ハーフワード・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					SDIO EN	予約 済み	FSMC EN	予約 済み	CRCE N	予約 済み	FLITF EN	予約 済み	SRAM EN	DMA2 EN	DMA1 EN
					rw		rw		rW		rw		rw	rw	rw

Bits 31:11 予約済み、常に 0 が読出されます。

Bit 10 SDIOEN SDIO クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: SDIO にはクロックが供給されません。

1: SDIO にクロックが供給されます。

Bits 9 予約済み、常に 0 が読出されます。

Bit 8 FSMCEN FSMC クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: FSMC にはクロックが供給されません。

1: FSMC にクロックが供給されます。



参考資料

Bit 7 予約済み、常に 0 が読出されます。

Bit 6 **CRCEN**: CRC クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: CRC にはクロックが供給されません。

1: CRC にクロックが供給されます。

Bit 5 予約済み、常に 0 が読出されます。

Bit 4 **FLITFEN**: FLITF クロック・イネーブル

SLEEP モードにおける FLITF クロックのオン / オフを切り替えるため、ソフトウェアでセット / リセットができます。

0: SLEEP モードで、FLITF にはクロックが供給されません。

1: SLEEP モードで、FLITF にクロックが供給されます。

Bit 3 予約済み、常に 0 が読出されます。

Bit 2 **SRAMEN**: SRAM インタフェース・クロック・イネーブル

SLEEP モードにおける SRAM クロックのオン / オフを切り替えるため、ソフトウェアでセット / リセットができます。

0: SLEEP モードで、SRAM にはクロックが供給されません。

1: SLEEP モードで、SRAM にクロックが供給されます。

Bit 1 **DMA2EN**: DMA2 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: DMA2 にはクロックが供給されません。

1: DMA2 にクロックが供給されます。

Bit 0 **DMAEN**: DMA クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: DMA にはクロックが供給されません。

1: DMA にクロックが供給されます。

参考資料

6.3.7 APB2 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB2ENR)

アドレスオフセット : 18h

リセット値 : 0000 0000h

アクセス : ワード・ハーフワード・バイトアクセス

アクセスはノーウェイトで行われます。ただし、APB2 ドメインのペリフェラルにアクセス中の場合は、これが終了するまでウェー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															
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3 EN	USART T1EN	TIM8 EN	SPI1 EN	TIM1 EN	ADC2 EN	ADC1 EN	IOPG EN	IOPF EN	IOPE EN	IOPD EN	IOPC EN	IOPB EN	IOPA EN	予約 済み	AFIO EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	予約済 み	rw

Bits 31:16 予約済み、常に 0 が読出されます。

Bit 15 **ADC3EN** ADC 3 インタフェース・クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: ADC3 インタフェースにはクロックが供給されません。

1: ADC3 インタフェースにクロックが供給されます。

Bit 14 **USART1EN** USART1 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: USART1 にはクロックが供給されません。

1: USART1 にクロックが供給されます。

Bit 13 **TIM8EN** TIM8 タイマ・クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: TIM8 にはクロックが供給されません。

1: TIM8 にクロックが供給されます。

Bit 12 **SPI1EN**: SPI 1 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: SPI 1 にはクロックが供給されません。

1: SPI 1 にクロックが供給されます。

Bit 11 **TIM1EN**: TIM 1 タイマ・クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: TIM1 にはタイマ・クロックが供給されません。

1: TIM1 にタイマ・クロックが供給されます。

Bit 10 **ADC2EN**: ADC 2 インタフェース・クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: ADC 2 インタフェースにはクロックが供給されません。

1: ADC 2 インタフェースにクロックが供給されます。

参考資料

Bit 9 **ADC1EN: ADC 1 インタフェース・クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: ADC 1 インタフェースにはクロックが供給されません。

1: ADC 1 インタフェースにクロックが供給されます。

Bit 8 **IOPEEN: 入出力ポート G クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: 入出力ポート G にはクロックが供給されません。

1: 入出力ポート G にクロックが供給されます。

Bit 7 **IOPEEN: 入出力ポート F クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: 入出力ポート F にはクロックが供給されません。

1: 入出力ポート F にクロックが供給されます。

Bit 6 **IOPEEN: 入出力ポート E クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: 入出力ポート E にはクロックが供給されません。

1: 入出力ポート E にクロックが供給されます。

Bit 5 **IOPDEN: 入出力ポート D クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: 入出力ポート D にはクロックが供給されません。

1: 入出力ポート D にクロックが供給されます。

Bit 4 **IOPCEN: 入出力ポート C クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: 入出力ポート C にはクロックが供給されません。

1: 入出力ポート C にクロックが供給されます。

Bit 3 **IOPBEN: 入出力ポート B クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: 入出力ポート B にはクロックが供給されません。

1: 入出力ポート B にクロックが供給されます。

Bit 2 **IOPAEN: 入出力ポート A クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: 入出力ポート A にはクロックが供給されません。

1: 入出力ポート A にクロックが供給されます。

Bit 1 予約済み、常に 0 が読出されます。

Bit 0 **AFIOEN: オルタネート機能入出力クロック・イネーブル**

ソフトウェアでセット / リセットができます。

0: オルタネート機能入出力にはクロックが供給されません。

1: オルタネート機能入出力にクロックが供給されます。

参考資料

6.3.8 APB1 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB1ENR)

アドレスオフセット : 0x1C

リセット値 : 0000 0000h

アクセス : ワード・ハーフワード・バイトアクセス

アクセスはノーウェイトで行われます。ただし、APB1 ドメインのペリフェラルにアクセス中の場合は、これが終了するまでウェー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	DAC EN	PWR EN	BKP EN	予約済み	CAN EN	予約済み	USB EN	I2C2 EN	I2C1 EN	UART5E N	UART4 EN	USART3EN	USART2EN	予約済み	
Res	rw	rw	rw	Res	rw	Res	rw	rw	rw	rw	rw	rw	rw	rw	Res
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 EN	SPI2 EN	予約済み	WWD GEN		予約済み					TIM7 EN	TIM6 EN	TIM5 EN	TIM4 EN	TIM3 EN	TIM2 EN
rw	rw	Res	rw		Res					rw	rw	rw	rw	rw	rw

Bits 31:30 予約済み、常に 0 が読出されます。

Bit 29 **DACEN**: DAC インタフェース・イネーブル

ソフトウェアでセット / リセットができます。

0: DAC インタフェースにはクロックが供給されません。

1: DAC インタフェースにクロックが供給されます。

Bit 28 **PWREN**: 電源インターフェース・クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: 電源インターフェースにはクロックが供給されません。

1: 電源インターフェースにクロックが供給されます。

Bit 27 **BKOPEN**: バックアップ・インターフェース・クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: バックアップ・インターフェースにはクロックが供給されません。

1: バックアップ・インターフェースにクロックが供給されます。

Bit 26 予約済み、常に 0 が読出されます。

Bit 25 **CANEN**: CAN クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: CAN にはクロックが供給されません。

1: CAN にクロックが供給されます。

Bit 24 予約済み、常に 0 が読出されます。

Bit 23 **USBEN**: USB クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: USB にはクロックが供給されません。

1: USB にクロックが供給されます。

Bit 22 **I2C2EN**: I2C 2 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: I2C 2 にはクロックが供給されません。

1: I2C 2 にクロックが供給されます。

参考資料

Bit 21 **I2C1EN: I₂C 1 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: I₂C 1 にはクロックが供給されません。
1: I₂C 1 にクロックが供給されます。

Bit 20 **USART5EN: USART 5 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: USART 5 にはクロックが供給されません。
1: USART 5 にクロックが供給されます。

Bit 19 **USART4EN: USART 4 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: USART 4 にはクロックが供給されません。
1: USART 4 にクロックが供給されます。

Bit 18 **USART3EN: USART 3 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: USART 3 にはクロックが供給されません。
1: USART 3 にクロックが供給されます。

Bit 17 **USART2EN: USART 2 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: USART 2 にはクロックが供給されません。
1: USART 2 にクロックが供給されます。

Bits 16 予約済み、常に 0 が読出されます。

Bit 15 **SPI3EN: SPI 3 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: SPI 3 にはクロックが供給されません。
1: SPI 3 にクロックが供給されます。

Bit 14 **SPI2EN: SPI 2 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: SPI 2 にはクロックが供給されません。
1: SPI 2 にクロックが供給されます。

Bits 13:12 予約済み、常に 0 が読出されます。

Bit 11 **WWDGGEN: Window 型ウォッチドッグ・クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: Window 型ウォッチドッグにはクロックが供給されません。
1: Window 型ウォッチドッグにクロックが供給されます。

Bits 10:6 予約済み、常に 0 が読出されます。

Bit 5 **TIM7EN: タイマ 7 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: タイマ 7 にはクロックが供給されません。
1: タイマ 7 にクロックが供給されます。

Bit 4 **TIM6EN: タイマ 6 クロック・イネーブル**

ソフトウェアでセット / リセットができます。
0: タイマ 6 にはクロックが供給されません。
1: タイマ 6 にクロックが供給されます。

参考資料

Bit 3 **TIM5EN**: タイマ 5 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: タイマ 5 にはクロックが供給されません。

1: タイマ 5 にクロックが供給されます。

Bit 2 **TIM4EN**: タイマ 4 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: タイマ 4 にはクロックが供給されません。

1: タイマ 4 にクロックが供給されます。

Bit 1 **TIM3EN**: タイマ 3 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: タイマ 3 にはクロックが供給されません。

1: タイマ 3 にクロックが供給されます。

Bit 0 **TIM2EN** タイマ 2 クロック・イネーブル

ソフトウェアでセット / リセットができます。

0: タイマ 2 にはクロックが供給されません。

1: タイマ 2 にクロックが供給されます。

参考資料

6.3.9 バックアップ・ドメイン制御レジスタ (RCC_BDCR)

アドレスオフセット : 0x20

リセット値 : 0000 0000h (バックアップ・ドメイン・リセット信号でリセットされます。)

アクセス : 0 ≤ ウェート数 ≤ 3、ワード・ハーフワード・バイトアクセス

このレジスタへの連続したアクセスで、ウェートステートが挿入されます。

バックアップ・ドメイン制御レジスタ (RCC_BDCR) の LSEON, LSEBYP, RTCSEL, RTCEN の各ビットはバックアップ・ドメインに属しています。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 ビットをセットしなければなりません。詳細はセクション 5 (63 ページ) を参照してください。これらのビットは V_{BAT} 電源がオンでバックアップ・ドメイン・リセットの時にのみリセットされます。他のリセット機能はこれらのビットの値には影響を与えません。

注意 :

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															BDRST
Res															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTC EN	予約済み				RTCSEL[1:0]		予約済み				LSE BYP	LSE RDY	LSEON		
rw	Res				rw	rw	Res				rw	r	rw		

Bits 31:17 予約済み、常に 0 が読出されます。

Bit 16 BDRST: バックアップ・ドメイン・ソフトウェア・リセット

このビットはソフトウェアでセット / リセットができます。

0: リセットが有効ではありません。

1: バックアップ・ドメイン全体がリセットされます。

Bit 15 RTCEN: RTC クロック・イネーブル

このビットはソフトウェアでセット / リセットができます。

0: RTC にはクロックが供給されません。

1: RTC にクロックが供給されます。

Bits 14:10 予約済み、常に 0 が読出されます。

Bits 9:8 RTCSEL[1:0]: RTC クロック・ソースの選択

これらのビットはソフトウェアで RTC のクロック・ソースを選択するセットを行うことができます。RTC クロック・ソースの選択が行われるとバックアップドメインのリセット無しで変更することはできません。この内容のリセットは BDRST ビットで行います。

00: クロックなし

01: LSE オシレータのクロックが RTC クロックとして使用されます。

10: LSI オシレータのクロックが RTC クロックとして使用されます。

11: HSE オシレータのクロックを 128 分周した信号が RTC クロックとして使用されます。

Bits 7:3 予約済み、常に 0 が読出されます。

Bit 2 LSEBYP: 外部低速オシレータ・バイパス

デバッグ・モードで、オシレータをバイパスするためにソフトウェアでセット / リセットすることができます。このビットは外部 32kHz オシレータがディセーブルの時にのみ書き込みが出来ます。

0: LSE オシレータはバイパスされません。

1: LSE オシレータがバイパスされます。

参考資料

Bit 1 LSERDY: 外部低速オシレータ・レディ

外部 32kHz オシレータが安定した動作状態であるか否かを示すようにハードウェアでセット / リセットされます。LSEON がリセットされてからフラグがクリアされるまで、この外部低速オシレータ・クロックで 6 サイクル相当の時間を必要とします。

0: LSE オシレータはレディ状態ではありません。

1: LSE オシレータはレディ状態です。

Bit 0 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でセット / リセットができます。

0: 外部 32kHz オシレータ OFF

1: 外部 32kHz オシレータ ON

6.3.10 制御 / ステータス・レジスタ (RCC_CSR)

アドレスオフセット : 0x24

リセット値 : 0C00 0000h (システム・リセットでリセットされます。ただしリセット・フラグ類は電源リセット時のみリセットされます)

アクセス : 0 ≤ ウェイト数 ≤ 3、ワード・ハーフワード・バイトアクセス

このレジスタへの連続したアクセスで、ウェー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
LPWR RSTF	WWDG RSTF	IWDG RSTF	SFT RSTF	POR RSTF	PIN RSTF	予約済み	RMVF	予約済み							
rw	rw	rw	rw	rw	rw	Res	rw	予約済み							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み												LSI RDY	LSION		
予約済み												r	rw		

Bit 31 LPWRRSTF: 低電力リセット・フラグ

低電力管理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、ソフトウェアで RMVF ビットに 1 を書込むことでリセットされます。

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は [セクション : 低電力管理リセット](#) を参照してください。

Bit 30 WWDGRSTF: Window 型ウォッチドッグ・リセット・フラグ

Window 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でセットされ、ソフトウェアで RMVF ビットに 1 を書込むことでリセットされます。

0: Window 型ウォッチドッグ・リセットは発生していません。

1: Window 型ウォッチドッグ・リセットが発生しました。

Bit 29 IWDGRSTF: 独立型ウォッチドッグ・リセット・フラグ

V_{DD} メインでウォッチドッグ・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、ソフトウェアで RMVF ビットに 1 を書込むことでリセットされます。

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。

参考資料

Bit 28 **SFTRSTF: ソフトウェア・リセット・フラグ**

ソフトウェア・リセットが発生したときにハードウェアによってセットされ、ソフトウェアで RMVF ビットに 1 を書込むことでリセットされます。

- 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。

Bit 27 **PORRSTF: POR/PDR リセット・フラグ**

POR/PDR リセットが発生したときにハードウェアでセットされ、ソフトウェアで RMVF ビットに 1 を書込むことでリセットされます。

- 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。

Bit 26 **PINRSTF: ピン・リセット・フラグ**

NRST ピンでリセットが発生したときにハードウェアでセットされ、ソフトウェアで RMVF ビットに 1 を書込むことでリセットされます。

- 0: NRST ピンのリセットは発生していません。
- 1: NRST ピンのリセットが発生しました。

Bit 25 予約済み、常に 0 が読出されます。

Bit 24 **RMVF: リセット・フラグ解除**

リセット・フラグの値を初期値に戻すため、ソフトウェアでセット / リセットします。

- 0: リセット・フラグの初期化は行われません。
- 1: リセット・フラグの初期化を行います。

Bits 23:2 予約済み、常に 0 が読出されます。

Bit 1 **LSIRDY: 内蔵低速オシレータ・レディ**

内蔵 RC40kHz オシレータが安定した動作状態であるか否かを示すように、ハードウェアでセット / リセットされます。LSION がリセットされてからフラグがクリアされるまで、この内蔵 RC40kHz オシレータのクロックで 3 サイクル相当の時間を必要とします。

- 0: 内蔵 RC40kHz オシレータはレディ状態ではありません。
- 1: 内蔵 RC40kHz オシレータはレディ状態です。

Bit 0 **LSION: 内蔵低速オシレータ・イネーブル**

このビットはソフトウェアでセット / リセットができます。

- 0: 内蔵 RC40kHz オシレータ OFF
- 1: 内蔵 RC40kHz オシレータ ON

參考資料

6.3.11 RCC レジスタ・マップ

次の表は RCC レジスタ・マップ及びリセット値について示しています。

表 13. RCC - レジスタ・マップとリセット時の値

レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。

7 汎用入出力とオルタネート機能入出力(GPIO、AFIO)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

7.1 GPIO 機能

汎用入出力ポート(GPIO)には、それぞれに、32bit の構成レジスタ (GPIOx_CRL、GPIOx_CRH) と、2 個の 32bit データ・レジスタ (GPIOx_IDR、GPIOx_ODR)、32bit のセット / リセット・レジスタ (GPIOx_BSRR)、16bit のリセット・レジスタ (GPIOx_BRR)、32bit のロックレジスタ (GPIOx_LCKR) が備えられています。

個々の入出力ポートの特定のハードウェア特性に関しては、データシートに記載されています。汎用入出力(GPIO)ポートの各ビットは、個々に、ソフトウェアで以下の動作モードを選択できます。

- フローティング入力
- プルアップ付き入力
- プルダウン付き入力
- アナログ入力
- オープンドレイン出力
- プッシュプル出力
- オルタネート機能対応プッシュプル
- オルタネート機能対応オープンドレイン

個々の入出力ポートビットは任意に設定できますが、入出力ポート・レジスタは 32 ビットワードとしてアクセスしなければなりません。ハーフワードやバイト単位でのアクセスはできません。GPIOx_BSRR レジスタと GPIOx_BRR レジスタを利用すれば、GPIO レジスタにビット単位でアクセスし、値を変更することができます。これにより、一部ビットの変更を複数命令で行う際に生じる、読出しと変更の間の割込みリクエストの発生リスクを回避することができます。

[図 10](#)は、入出力ポートビットの基本的な構造を示しています。

参考資料

図 10. 入出力ポートの基本構造

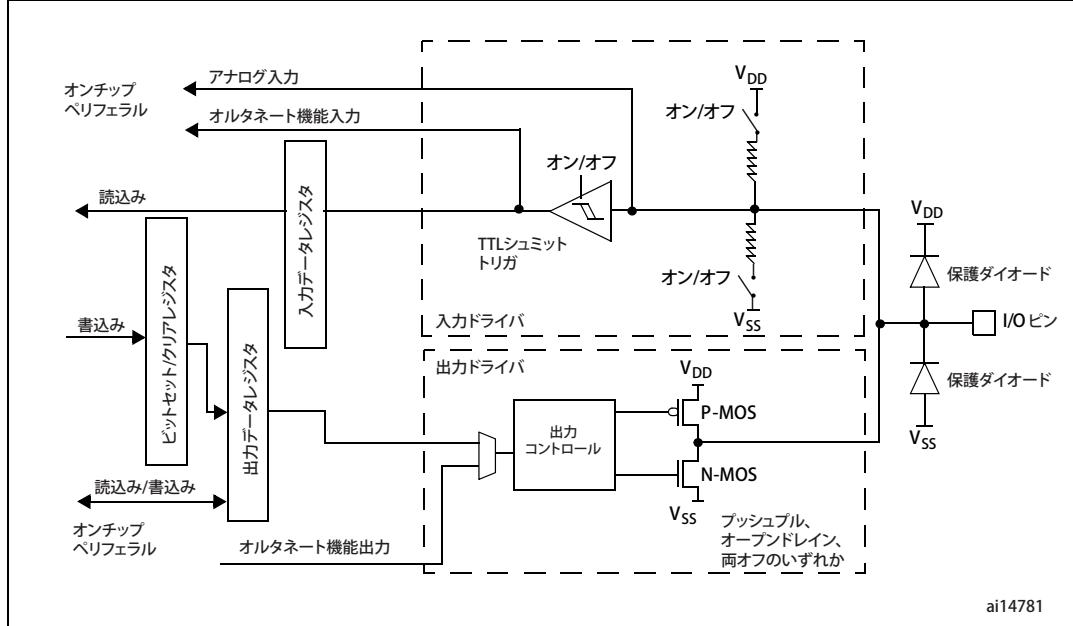
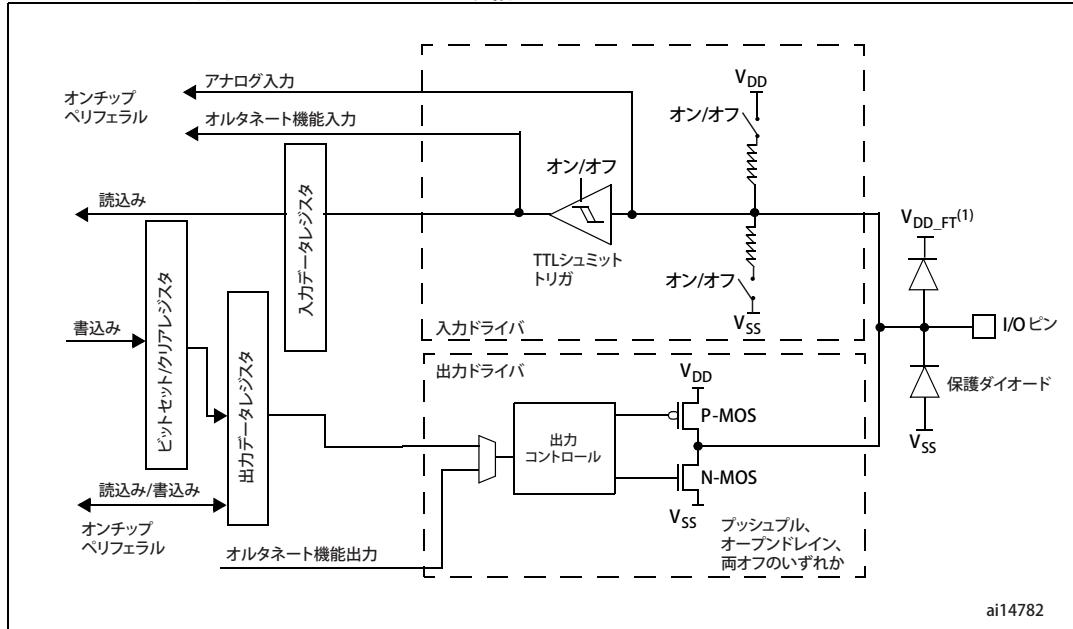


図 11. 5V 耐性入出力ポートの基本構造



1. V_{DD_FT} は、5V 耐性入出力の電位で、V_{DD} とは異なります。

参考資料

表 14. ポート・ビット構成表

ポート構成		CNF1	CNF0	MODE1	MODE0	PxODR レジスタ
汎用出力	プッシュプル	0	0	01 10 11		0 または 1
	オープンドレイン		1			0 または 1
オルタネート機能・ 出力	プッシュプル	1	0			無視
	オープンドレイン		1			無視
入力	アナログ入力	0	0	00		無視
	フローティング入力		1			無視
	入力・プルダウン付入力	1	0			0
	入力・プルアップ付入力					1

表 15. 出力モードビット

MODE[1:0]	意味
00	予約値
01	最大出力速度 10 MHz
10	最大出力速度 2 MHz
11	最大出力速度 50 MHz

7.1.1 汎用入出力(GPIO)

リセット中、及びリセット直後は、オルタネート機能は有効ではなく、入出力ポートはフローティング入力モード (CNFx[1:0]=01b, MODEx[1:0]=00b) に設定されています。

リセット直後、JTAG ピンはプルアップ / プルダウンを有する入力モードとなっています。

PA15: JTDI 入力 (プルアップ付き)

PA14: JTCK 入力 (プルダウン付き)

PA13: JTMS 入力 (プルアップ付き)

PB4: JNTRST 入力 (プルアップ付き)

ポートビットが出力に設定されると、出力データ・レジスタ (GPIOx_ODR) に書込まれていたデータの値がピンに出力されます。出力ドライバは、プッシュプル・モードかオープンドレイン・モード (出力が 0 の時に N-MOS のみがアクティブ (ON) となります) のいずれかを指定できます。

入力データ・レジスタ (GPIOx_IDR) は、APB2 クロック・サイクルごとに、入出力ピン上の信号の状態を反映します。

プルアップ / プルダウンが選択されているか否かにかかわらず、入力モードに設定された GPIO ピンは常に内部に高い抵抗値のプルアップとプルダウンを搭載しています。

7.1.2 アトミック・ビットのセット / リセット

GPIOx_ODR をビットレベルで操作する際に、ソフトウェアで割込みを禁止する必要はありません。一回のアトミック APB2 書込みアクセスのみで、1 もしくは数ビットのみを変更することができます。このためには、ビットセット / リセット・レジスタ (GPIOx_BSRR、もしくはリセットのみであれば GPIOx_BRR) の変更したいビットに対応する位置に 1 を書き込みます。選択されなかった (= 0) 出力ビットの値は変わりません。

7.1.3 外部割込み / ウェークアップ信号

各ポートで、外部割込みを発生させることができます。外部割込みを利用するには、そのポートは入力モードにする必要があります。詳細は、次のセクションを参照してください。

- セクション 8.2: 外部割込み / イベントコントローラ (EXTI) (134 ページ)
- セクション 8.2.3: 再起動イベント管理 (130 ページ)

7.1.4 オルタネート機能(AF)

オルタネート機能を使用するには、デフォルトのピン接続であっても、事前にポートビット構成レジスタの設定を行う必要があります。

- オルタネート機能の入力では、ポートは入力モード（フローティング、プルアップ、プルダウン）に設定し、入力ピンは外部から駆動しなければなりません。

注意：

汎用入出力コントローラをプログラムする事で、ソフトウェアで AF1 入力ピンをエミュレートすることが出来ます。この場合、ポートをオルタネート機能出力モードに設定する必要があります。この際明らかに、対応するポートは汎用入出力コントローラを使いソフトウェアにより駆動されるので、外部から駆動してはなりません。

- オルタネート機能の出力では、オルタネート機能出力（プッシュプル、オープンドレイン）を選択します。
- オルタネート機能の信号が両方向の場合は、ポートビットはオルタネート機能出力（プッシュプル、オープンドレイン）に設定します。このとき、入力ドライバはフローティング入力モードとなっています。

ポートビットをオルタネート機能出力に設定した時には、ピンは GPIO 出力レジスタから切離され、オンチップ・ペリフェラル回路の出力信号に接続されます。

ソフトウェアでオルタネート機能出力を選択した場合でペリフェラル回路がアクティブではないとき、その出力ピンの信号は確定しません。

7.1.5 ソフトウェアによるオルタネート機能の入出力ピンの再割当て

異なる種類のデバイスパッケージで複数のペリフェラル入出力機能を利用する場合、いくつかのペリフェラル回路を別のピンに割当ることにより適切な割当てにすることができます。この場合は、対応するレジスタの設定を行います。（[AFIO レジスタ \(123 ページ\)](#) を参照してください。）また、そのオルタネート機能では標準割当てのピンは使用できません。

7.1.6 GPIO ロック機構

ロック機構は I/O ポート構成の設定をロックします。ポートの個々のビットにロックの手順が適用されると、次のリセットまで、そのポートビットの値を変更できなくなります。

参考資料

RM0008

汎用入出力とオルタネート機能入出力(GPIO、AFIO)

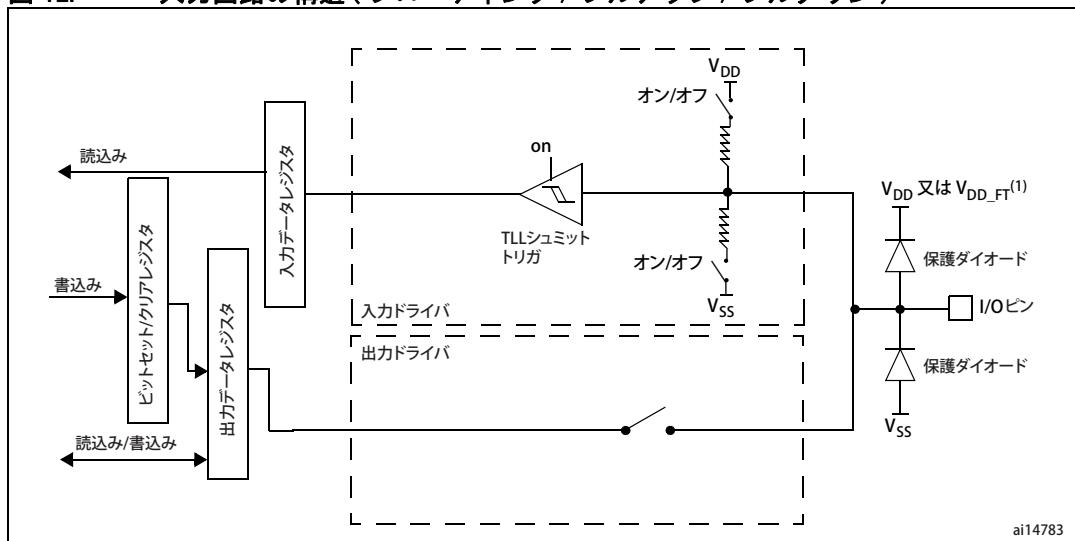
7.1.7 入力回路の構成

入出力ポートで入力モードが選択されたときには、

- 出力バッファは無効になります。
- シュミット・トリガ入力が有効となります。
- 入力のモード設定（プルアップ、プルダウン、フローティング）にかかわらず、高抵抗のプルアップ、プルダウン抵抗はアクティブです。
- 入出力ピン上のデータは、APB2 クロック・サイクルごとに入力データ・レジスタに読み込まれます。
- 入力データ・レジスタを読み出すことで、入出力ピンの状態を得ることができます。

図 12 (107 ページ) は、入力モード時の入出力ポートの状態を示しています。

図 12. 入力回路の構造(フローティング / プルアップ / プルダウン)



1. V_{DD_FT} は、5V 耐性入出力の電位で、 V_{DD} とは異なります。

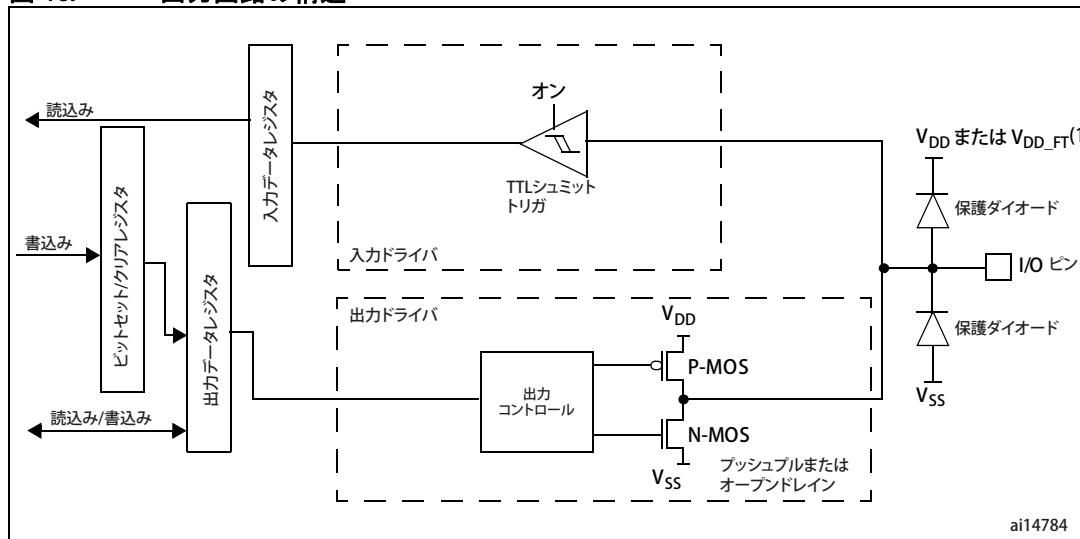
7.1.8 出力回路の構成

入出力ポートで出力モードが選択されたときには、

- 出力バッファが有効となります。
 - オープンドレイン・モード：出力レジスタが 0 のとき N-MOS がアクティブ(ON)となり、1 のときにはポートはハイ・インピーダンス状態となります。(P-MOS がアクティブと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 がアクティブ(ON)となり、1 のときには P-MOS がアクティブ(ON)となります。
- シュミット・トリガ入力は有効で動作します。
- 高抵抗のプルアップ、プルダウン抵抗は無効です。
- 入出力ピン上のデータは、APB2 クロック・サイクルごとに入力データ・レジスタに読み込まれます。
- オープンドレイン・モードでは、入力データ・レジスタを読み出すことで、入出力ピンの状態を得ることができます。
- プッシュプル・モードでは、出力データ・レジスタを読み出すことで、最後に書込まれたデータを得ることができます。

図 13 (108 ページ) は、出力モード時の入出力ポートの構造を示しています。

図 13. 出力回路の構造



1. V_{DD_FT} は、5V 耐性入出力の電位で、 V_{DD} とは異なります。

7.1.9 オルタネート機能の構成

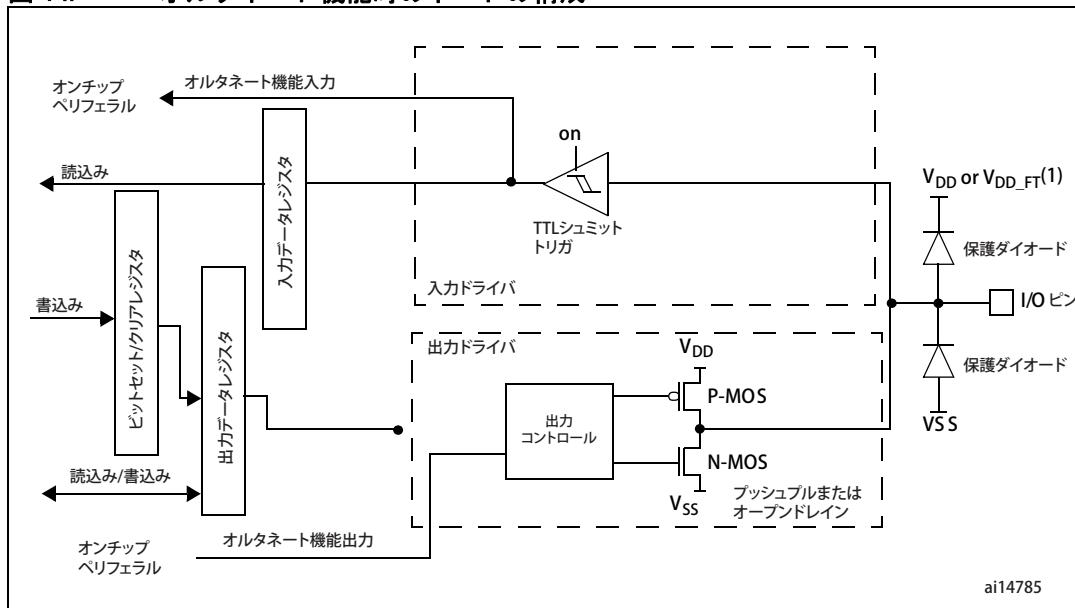
入出力ポートがオルタネート機能に設定されている場合には、

- 出力バッファはオープンドレインもしくはプッシュプル・モードで動作します。
- 出力バッファはペリフェラル回路からの信号で駆動されます。(オルタネート機能出力)
- シュミット・トリガ入力は有効です。
- 高抵抗のプルアップ、プルダウン抵抗は無効になります。
- 入出力ピン上のデータは、APB2 クロック・サイクルごとに、入力データ・レジスタに読み込まれます。
- オープンドレイン・モードでは、入力データ・レジスタを読み出すことで、入出力ピンの状態を得ることができます。
- プッシュプル・モードでは、出力データ・レジスタを読み出すことで、最後に出力レジスタに書込まれたデータを得ることができます。

[図 14 \(109 ページ\)](#) は、オルタネート機能を選択した時の入出力ポートの状態を示しています。同時に、[セクション 7.4: AFIO レジスタ \(123 ページ\)](#) をあわせて参照してください

オルタネート機能入出力 (AFIO) レジスタの設定で、オルタネート機能のピン割当てを変更することができます。

図 14. オルタネート機能時のポートの構成



1. V_{DD_FT} は、5V 耐性入出力の電位で、V_{DD} とは異なります。

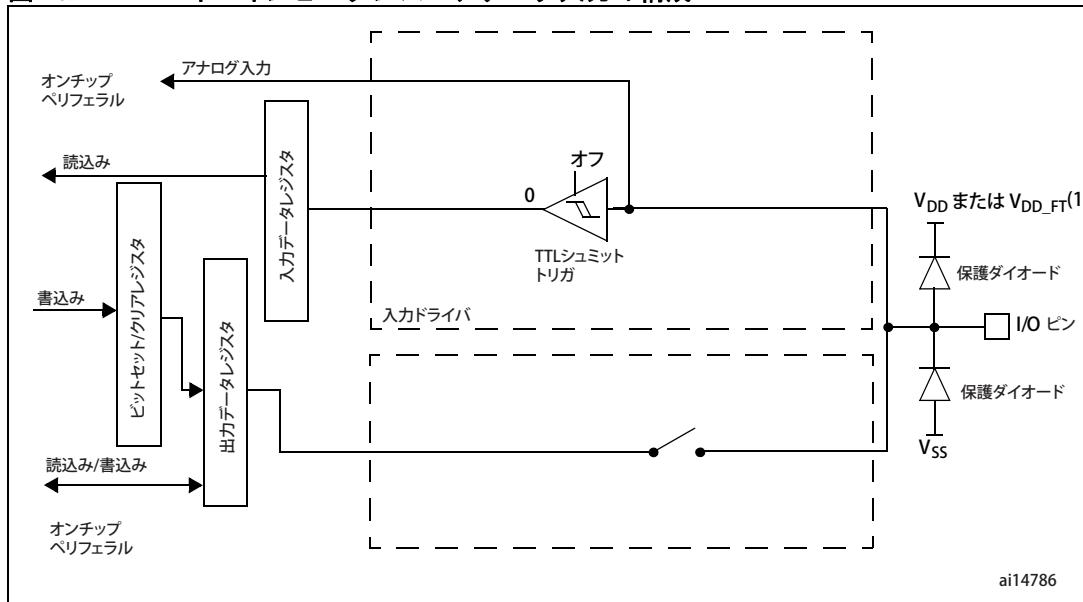
7.1.10 アナログ入力の構成

入出力ポートでアナログ入力が選択されたときには、

- 出力バッファはディセーブル状態になります。
- 入出力ピンに負荷を与えないため、シュミット・トリガ入力はオフ状態となります。シュミット・トリガ回路の出力は常に 0 となります。
- 高抵抗のプルアップ、プルダウン抵抗は存在しません。(ディセーブル状態です。)
- 入力データ・レジスタからは、0 が読出されます。

[図 15 \(110 ページ\)](#) は、ハイ・インピーダンス・アナログ入力の構成を示しています。

図 15. ハイ・インピーダンス・アナログ入力の構成



参考資料

RM0008

汎用入出力とオルタネート機能入出力(GPIO、AFIO)

7.2 GPIO レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#)を参照してください。

7.2.1 ポート構成レジスタ(下位)(GPIOx_CRL) (x=A..G)

アドレスオフセット : 0x00

リセット値 : 0x4444 4444

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF7[1:0]	MODE7[1:0]	CNF6[1:0]	MODE6[1:0]	CNF5[1:0]	MODE5[1:0]	CNF4[1:0]	MODE4[1:0]	CNF3[1:0]	MODE3[1:0]	CNF2[1:0]	MODE2[1:0]	CNF1[1:0]	MODE1[1:0]	CNF0[1:0]	MODE0[1:0]
RW	RW														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RW	RW														

Bits 31:30, 27:26, 23:22, **CNFx[1:0]**: ポート x 構成ビット ($x = 0 \dots 7$)
19:18, 15:14, 11:10, 7:6, 3:2 これらのビットは対応するポートの構成を指定するため、ソフトウェアで書込まれます。

[表 14: ポート・ビット構成表 \(104 ページ\)](#) を参照してください。

入力モード時 (MODE[1:0] = 00):

- 00: アナログ入力モード
- 01: フロー・ティング入力 (リセット時の値)
- 10: ブルアップ / ブルダウン付きの入力
- 11: 予約値

出力モード時 (MODE[1:0] > 00):

- 00: ブッシュプル・汎用出力
- 01: オープンドレイン・汎用出力
- 10: ブッシュプル・オルタネート機能出力
- 11: オープンドレイン・オルタネート機能

Bits 29:28, 25:24, 21:20, **MODEx[1:0]**: ポート x モードビット ($x = 0 \dots 7$)

17:16, 13:12, 9:8, 5:4, 1:0 これらのビットは対応するポートの構成を指定するため、ソフトウェアで書込まれます。

[表 14: ポート・ビット構成表 \(104 ページ\)](#) を参照してください。

00: 入力モード (リセット時の値)

01: 出力モード、最大速度 10 MHz

10: 出力モード、最大速度 2 MHz

11: 出力モード、最大速度 50 MHz



参考資料

7.2.2 ポート構成レジスタ(上位)(GPIOx_CRH) (x=A..G)

アドレスオフセット : 0x04

リセット値 : 0x4444 4444

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF15[1:0]		MODE15[1:0]		CNF14[1:0]		MODE14[1:0]		CNF13[1:0]		MODE13[1:0]		CNF12[1:0]		MODE12[1:0]	
RW	RW	RW	RW												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF11[1:0]		MODE11[1:0]		CNF10[1:0]		MODE10[1:0]		CNF9[1:0]		MODE9[1:0]		CNF8[1:0]		MODE8[1:0]	
RW	RW	RW	RW												

Bits 31:30, 27:26, 23:22, **CNF_x[1:0]: ポート_x構成ビット (x = 8 .. 15)**
19:18, 15:14, 11:10, 7:6, 3:2 これらのビットには対応するポートの構成を指定するため、ソフトウェアで書き込まれます。

[表 14: ポート・ビット構成表 \(104 ページ\)](#) を参照してください。

入力モード時 (MODE[1:0] = 00):

- 00: アナログ入力モード
- 01: フローティング入力 (リセット時の値)
- 10: プルアップ / プルダウン付きの入力
- 11: 予約値

出力モード時 (MODE[1:0] > 00):

- 00: プッシュプル・汎用出力
- 01: オープンドレイン・汎用出力
- 10: プッシュプル・オルタネート機能出力
- 11: オープンドレイン・オルタネート機能

Bits 29:28, 25:24, 21:20, **MODE_x[1:0]: ポート_xモードビット (x = 8 .. 15)**

17:16, 13:12, 9:8, 5:4, 1:0 これらのビットには対応するポートの構成を指定するため、ソフトウェアで書き込まれます。

[表 14: ポート・ビット構成表 \(104 ページ\)](#) を参照してください。

00: 入力モード (リセット時の値)

- 01: 出力モード、最大速度 10 MHz
- 10: 出力モード、最大速度 2 MHz
- 11: 出力モード、最大速度 50 MHz

参考資料

RM0008

汎用入出力とオルタネート機能入出力(GPIO、AFIO)

7.2.3 ポート入力データ・レジスタ (GPIOx_IDR) ($x=A..G$)

アドレスオフセット : 0x08h

リセット値 : 0x0000 XXXX

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 r r r r

Bits 31:16 予約済み、常に 0 が読出されます。

Bits 15:0 **IDRx[15:0]**: ポート入力データ ($x = 0 .. 15$)

これらのビットは読み出し専用で、そのアクセスはワードモードに限られます。これらのビットには対応するポートの入力値が格納されています。

7.2.4 ポート出力データ・レジスタ (GPIOx_ODR) ($x=A..G$)

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
ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0

rw rw

Bits 31:16 予約済み、常に 0 が読出されます。

Bits 15:0 **ODRx[15:0]**: ポート出力データ ($x = 0 .. 15$)

これらのビットはソフトウェアで読み出し・書き込みができ、そのアクセスはワードモードに限られます。

注: 個別のビットセット / リセット時には、GPIOx_BSRR ($x = A .. G$) レジスタ書き込みで、ODR ビットが個々にセット / リセットされます。



参考資料

7.2.5 ポートビット・セット / リセット・レジスタ (GPIOx_BSRR) (x=A..G)

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
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

Bits 31:16 **BRx**: ビット x リセット ($x = 0..15$)

これらのビットは書き込み専用で、そのアクセスはワードモードに限られます。

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（クリア）します。

注: BSx ビットと BRx ビットの両方が 1 の時には、BSx ビットの指定（セット）が優先します。

Bits 15:0 **BSx**: ビット x セット ($x = 0..15$)

これらのビットは書き込み専用で、そのアクセスはワードモードに限られます。

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。

7.2.6 ポートビット・リセット・レジスタ (GPIOx_BRR) (x=A..G)

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
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

Bits 31:16 予約済み

Bits 15:0 **BRx**: ビット x リセット ($x = 0..15$)

これらのビットは書き込み専用で、そのアクセスはワードモードに限られます。

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（クリア）します。

注: BSx ビットと BRx ビットの両方が 1 の時には、BSx ビットの指定（セット）が優先します。

参考資料

7.2.7 ポート構成ロックレジスタ (GPIOx_LCKR) ($x=A..G$)

このレジスタは、ビット 16 (LCKK) に正しい手順で書き込みを行い、ポートの個々のビットの構成をロックするために使用されます。ビット [15:0] の値は GPIO の構成をロックするために使われ、書き込み手順の間は LCKR[15:0] の値は同じでなければなりません。ポートビットに対する LOCK の手順が終了すると、次のリセットまで、ポートビットの値を変更することができません。

個々のロックビットは、制御レジスタ (CRL、CRH) の対応する 4bit の設定値を固定します。

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
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

Bits 31:17 予約済み

Bit 16 LCKK[15:0]: ロック・キー

このビットは常時読み出しができます。ビット値の変更はロックキー書き込み手順に従って行う必要があります。

0: ポート構成ロック・キーは無効です。

1: ポート構成ロック・キーが有効です。GPIOx_LCKR レジスタは MCU リセットが発生するまでロックされます。

ロック・キー書き込み手順:

1 を書込む

0 を書込む

1 を書込む

0 を読込む

1 を読込む (この読み出しは必ずしも必要ありませんが、ロックされたことを確認するために有効です。)

注:

ロックキー書き込み手順の間は、LCK[15:0] の値を変更することはできません。

ロック手順中にエラーが発生すると、手順は無効となります。

Bits 15:0 LCKy: 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、書き込み時の LCKK ビットは 0 にする必要があります。

0: ポート構成はロックされません。

1: ポート構成がロックされます。

7.3 オルタネート入出力機能とデバッグ構成(AFIO)

64 ピン、100 ピン及び 144 ピンのパッケージに搭載されているペリフェラル回路を必要数にあったものとするために、いくつかの回路機能を別のピンに割当てることができます。これはソフトウェアで [オルタネート機能の再割当てとデバッグ入出力構成レジスタ \(AFIO_MAPR\) \(124 ページ\)](#) の設定を行うことができます。この場合、オルタネート機能の、標準のピン割当ては無効となります。

7.3.1 GPIO ポート PC14/PC15 を OSC32_IN/OSC32_OUT ピンとして使用するには

LSE オシレータのピン OSC32_IN と OSC32_OUT は、LSE オシレータがオフの場合、汎用入出力 PC14 及び PC15 としてそれぞれ使用することができます。LSE は、GPIO 機能より高い優先順位を持っています。

- 注意： 1 *PC14/PC15 は、1.8V ドメインの電源がオフ (STANDBY 状態となる) の場合あるいは、 V_{BAT} (V_{DD} の供給が断たれる) によりバックアップ・ドメインに電力供給されている場合、その GPIO 機能は無効になります。この場合、入出力はアナログ入力モードにセットされます。*
- 2 *I/O の使い方の制限に関しては、セクション 4.1.2 (49 ページ) を参照してください。*

7.3.2 GPIO ポート PD0/PD1 を OSC_IN/OSC_OUT ピンとして使用するには

外部オシレータのピン OSC_IN と OSC_OUT は、[オルタネート機能の再割当てとデバッグ入出力構成レジスタ \(AFIO_MAPR\)](#) の PD01_REMAP ビットの設定で、汎用入出力 (GPIO) の PD0/PD1 信号に割当てることができます。

この割当て変更は、36 ピン、48 ピン、64 ピンのパッケージでのみ有効です。(100 ピン及び 144 ピンのパッケージでは PD0/PD1 には専用のピンが備えられているので、この割当て変更の必要はありません。)

- 注意： 外部割込み／イベント機能は、再割当てできません。36、48 及び 64 ピンのパッケージでは、PD0 と PD1 を外部割込み／イベント生成に使用することは出来ません。

7.3.3 CAN オルタネート機能のピン再割当て

CAN 信号は、[表 16](#) のように、ポート A、ポート B もしくはポート D のピンに割当て変更することができます。ポート D は、36、48 及び 64 ピンのパッケージのデバイスでは再割当てることは出来ません。

表 16. CAN オルタネート機能のピン割当て変更

オルタネート機能	CAN_REMAP[1:0] = "00"	CAN_REMAP[1:0] = "10" ⁽¹⁾	CAN_REMAP[1:0] = "11" ⁽²⁾
CANRX	PA11	PB8	PD0
CANTX	PA12	PB9	PD1

- 36 ピンパッケージではピン割当ての変更はできません。
- PD0 と PD1 が OSC_IN と OSC_OUT に再割当てされていない場合、ピン割当ての変更は、100 ピン及び 144 ピンパッケージの場合のみ可能です。

参考資料

7.3.4 JTAG/SWD オルタネート機能のピン割当て

デバッグ・インターフェース信号は、表 17 のように、GPIO ポートのピンに割当てられています。

表 17. デバッグ・インターフェース信号

オルタネート機能	GPIO ポート
JTMS / SWDIO	PA13
JTCK / SWCLK	PA14
JTDI	PA15
JTDO / TRACESWO	PB3
JNTRST	PB4
TRACECK	PE2
TRACED0	PE3
TRACED1	PE4
TRACED2	PE5
TRACED3	PE6

デバッグ時に GPIO で使用できるピンの割当てを最適化するため、オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR) の SWJ_CFG[1:0] ビットの設定で、複数のピン割当てから選ぶことができます。表 18 を参照してください。

表 18. デバッグ・ポートのピン割当て

SWJ_CFG [2:0]	利用可能なデバッグ・ポート	SWJ I/O ピンの割当て				
		PA.13 / JTMS/ SWDIO	PA.14 / JTCK/S WCLK	PA.15 / JTDI	PB.3 / JTDO/ TRACE SWO	PB.4/ JNTRST
000	フル SWJ (JTAG-DP + SW-DP) (リセット時の状態)	X	X	X	X	X
001	フル SWJ (JTAG-DP + SW-DP)、 JNTRST なし	X	X	X	x	自由
010	JTAG-DP ディセーブル及び SW-DP イネーブル	X	X	自由	自由 ⁽¹⁾	自由
100	JTAG-DP ディセーブル及び SW-DP ディセーブル	自由	自由	自由	自由	自由
その他	設定できません					

1. 非同期トレースを使用していないときにのみ、GPIO で使用できます。

参考資料

7.3.5 ADC オルタネート機能のピン割当て

オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR) を参照してください。

表 19. ADC1 外部トリガ・インジェクト変換オルタネート機能のピン割当て⁽¹⁾

オルタネート機能	ADC1_ETRGINJ_REMAP = 0	ADC1_ETRGINJ_REMAP = 1
ADC1 外部トリガ・インジェクト変換	ADC1 外部トリガ・インジェクト変換は EXTI15 へ接続	ADC1 外部トリガ・インジェクト変換は TIM8_CH4 へ接続

1. 割当て変更は、大容量デバイスに限り利用出来ます。

表 20. ADC1 外部トリガ・レギュラ変換オルタネート機能のピン割当て⁽¹⁾

オルタネート機能	ADC1_ETRGREG_REMAP = 0	ADC1_ETRGREG_REMAP = 1
ADC1 外部トリガ・レギュラ変換	ADC1 外部トリガ・レギュラ変換は EXTI11 へ接続	ADC1 外部トリガ・レギュラ変換は TIM8_TRGO へ接続

1. 割当て変更は、大容量デバイスに限り利用出来ます。

表 21. ADC2 外部トリガ・インジェクト変換オルタネート機能のピン割当て⁽¹⁾

オルタネート機能	ADC2_ETRGINJ_REMAP = 0	ADC2_ETRGINJ_REMAP = 1
ADC2 外部トリガ・インジェクト変換	ADC2 外部トリガ・インジェクト変換は EXTI15 へ接続	ADC2 外部トリガ・インジェクト変換は TIM8_CH4 へ接続

1. 割当て変更は、大容量デバイスに限り利用出来ます。

表 22. ADC2 外部トリガ・レギュラ変換オルタネート機能のピン割当て⁽¹⁾

オルタネート機能	ADC2_ETRGREG_REG = 0	ADC2_ETRGREG_REG = 1
ADC2 外部トリガ・レギュラ変換	ADC2 外部トリガ・レギュラ変換は EXTI11 へ接続	ADC2 外部トリガ・レギュラ変換は TIM8_TRGO へ接続

1. 割当て変更は、大容量デバイスに限り利用出来ます。

参考資料

7.3.6 タイマ・オルタネート機能のピン割当て

タイマ 4 のチャネル 1 ~ 4 は、ポート B とポート D のいずれかに割当てられます。他のタイマで変更可能なピン割当では表 25ないし表 27に示されています。同時にオルタネート機能の再割当てとデバッグ入出力構成レジスタ(AFIO_MAPR)を参照してください。

表 23. タイマ 5 オルタネート機能のピン割当て⁽¹⁾

オルタネート機能	TIM5CH4_IREMAP = 0	TIM5CH4_IREMAP = 1
TIM5_CH4	TIM5 チャネル 4 は PA3 へ接続	LSI の内部クロックは TIM5_CH4 入力へ較正の目的で接続

1. 割当て変更は、大容量デバイスに限り利用出来ます。

表 24. タイマ 4 オルタネート機能のピン割当て

オルタネート機能	TIM4_REMAP = 0	TIM4_REMAP = 1 ⁽¹⁾
TIM4_CH1	PB6	PD12
TIM4_CH2	PB7	PD13
TIM4_CH3	PB8	PD14
TIM4_CH4	PB9	PD15

1. 割当て変更は、100 ピンと 144 ピンパッケージに限り利用出来ます。

表 25. タイマ 3 オルタネート機能のピン割当て

オルタネート機能	TIM3_REMAP[1:0] = “00” (再割当てなし)	TIM3_REMAP[1:0] = “10” (部分的な再割当て)	TIM3_REMAP[1:0] = “11” (完全な再割当て) ⁽¹⁾
TIM3_CH1	PA6	PB4	PC6
TIM3_CH2	PA7	PB5	PC7
TIM3_CH3	PB0		PC8
TIM3_CH4	PB1		PC9

1. 割当て変更は、64 ピン、100 ピンと 144 ピンパッケージに限り利用出来ます。

表 26. タイマ 2 オルタネート機能のピン割当て

オルタネート機能	TIM2_REMAP[1:0] = “00” (再割当てなし)	TIM2_REMAP[1:0] = “01” (部分的な再割当て)	TIM2_REMAP[1:0] = “10” (部分的な再割当て) ⁽¹⁾	TIM2_REMAP[1:0] = “11” (完全な再割当て) ⁽¹⁾
TIM2_CH1_ETR ⁽²⁾	PA0	PA15	PA0	PA15
TIM2_CH2	PA1	PB3	PA1	PB3
TIM2_CH3	PA2		PB10	
TIM2_CH4	PA3		PB11	

1. 割当て変更は 36 ピンパッケージではできません。
2. TIM_CH と TIM_ETR は同じピンを共有しますが、同時に使用することは出来ません。(そのため TIM2_CH1_ETR という名称が付けられています)

参考資料

表 27. タイマ1 オルタネート機能のピン割当て

オルタネート機能 再割当て	TIM1_REMAP[1:0] = “00”(再割当てなし)	TIM1_REMAP[1:0] = “01”(部分的再割当て)	TIM1_REMAP[1:0] = “11” (完全な再割当て) ⁽¹⁾
TIM1_ETR	PA12		PE7
TIM1_CH1	PA8		PE9
TIM1_CH2	PA9		PE11
TIM1_CH3	PA10		PE13
TIM1_CH4	PA11		PE14
TIM1_BKIN	PB12 ⁽²⁾	PA6	PE15
TIM1_CH1N	PB13 ⁽²⁾	PA7	PE8
TIM1_CH2N	PB14 ⁽²⁾	PB0	PE10
TIM1_CH3N	PB15 ⁽²⁾	PB1	PE12

1. 割当て変更は、100 ピンと 144 ピンパッケージに限り利用出来ます。

2. 割当て変更は、36 ピンではできません。

7.3.7 USART オルタネート機能のピン割当て

オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR) の記述を参照してください。

表 28. USART3 ピン割当て

オルタネート機能	USART3_REMAP[1:0] = "00" (再割当てなし)	USART3_REMAP[1:0] = "01" (部分的再割当て) ⁽¹⁾	USART3_REMAP[1:0] = "11" (完全な再割当て) ⁽²⁾
USART3_TX	PB10	PC10	PD8
USART3_RX	PB11	PC11	PD9
USART3_CK	PB12	PC12	PD10
USART3_CTS		PB13	PD11
USART3_RTS		PB14	PD12

1. 割当て変更は、64 ピン、100 ピンと 144 ピンパッケージに限り利用出来ます。

2. 割当て変更は、100 ピンと 144 ピンパッケージに限り利用出来ます。

表 29. USART2 ピン割当て

オルタネート機能	USART2_REMAP = 0	USART2_REMAP = 1 ⁽¹⁾
USART2_CTS	PA0	PD3
USART2_RTS	PA1	PD4
USART2_TX	PA2	PD5
USART2_RX	PA3	PD6
USART2_CK	PA4	PD7

1. 割当て変更は、100 ピンと 144 ピンパッケージに限り利用出来ます。

表 30. USART1 ピン割当て

オルタネート機能	USART1_REMAP = 0	USART1_REMAP = 1
USART1_TX	PA9	PB6
USART1_RX	PA10	PB7

参考資料

7.3.8 I2C 回路 1 オルタネート機能のピン割当て

オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR) の記述を参照してください。

表 31. I2C1 ピン割当て

オルタネート機能	I2C1_REMAP = 0	I2C1_REMAP = 1 ⁽¹⁾
I2C1_SCL	PB6	PB8
I2C1_SDA	PB7	PB9

1. 割当て変更は、36 ピンではできません。

7.3.9 SPI 回路 1 オルタネート機能のピン割当て

オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR) の記述を参照してください。

表 32. SPI1 ピン割当て

オルタネート機能	SPI1_REMAP = 0	SPI1_REMAP = 1
SPI1_NSS	PA4	PA15
SPI1_SCK	PA5	PB3
SPI1_MISO	PA6	PB4
SPI1_MOSI	PA7	PB5

参考資料

RM0008

汎用入出力とオルタネート機能入出力(GPIO、AFIO)

7.4 AFIO レジスタ

レジスタの詳細記述で使用されている略語については [セクション 1.1 \(32 ページ\)](#) を参照してください。

7.4.1 イベント制御レジスタ (AFIO_EVCR)

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								EVOE	PORT[2:0]			PIN[3:0]			

Bits 31:8 予約済み

Bit 7 EVOE イベント出力カイネーブル

このビットはソフトウェアからセット / クリアができます。このビットがセットされているときには、Cortex の EVENT OUT 出力が、PORT[2:0] 及び PIN[3:0] bit で選択されたピンに出力されます。

Bits 6:4 PORT[2:0]: ポート選択

これらのビットは、Cortex EVENT OUT 信号を出力するポートを指定するため、ソフトウェアからセット / クリアができます。

000: PA が選択されます

001: PB が選択されます

010: PC が選択されます

011: PD が選択されます

100: PE が選択されます

Bits 3:0 PIN[3:0] ピン選択 ($x = A \dots E$)

これらのビットは、Cortex EVENT OUT 信号を出力するピンを指定するため、ソフトウェアからセット / クリアができます。

0000: Px0 が選択されます

0001: Px1 が選択されます

0010: Px2 が選択されます

0011: Px3 が選択されます

...

1111: Px15 が選択されます



参考資料

7.4.2 オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR)

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					SWJ_CFG[2:0]			予約済み			ADC2_ETRGR_EG_REMAP	ADC2_ETRGIN_J_REMAP	ADC1_ETRGR_EG_REMAP	ADC1_ETRGIN_J_REMAP	TIM5CH4_IREM_AP
w	w	w									rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD01_REMAP	CAN_REMAP[1:0]	TIM4_REMAP	TIM3_REMAP[1:0]	TIM2_REMAP[1:0]	TIM1_REMAP[1:0]	USART3_REMAP[1:0]	USART2_REMAP	USART1_REMAP	I2C1_REMAP	SPI1_REMAP					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:27 予約済み

Bits 26:24 SWJ_CFG[2:0]: シリアル・ワイヤ JTAG 構成

これらのビットは書き込みのみになります。(読み込む場合は値は未定義になります。) これらのビットは SWJ (Serial Wire JTAG) とトレース・オルタネート機能入出力を設定します。SWJ は JTAG もしくは SWD による Cortex デバッグ出力へのアクセスをサポートします。リセット直後のデフォルト値は SWJ オンで、トレースなしです。このとき JTMS/JTCK ピンに特定のシーケンスを送ることで、JTAG もしくは SW モードが有効となります。

000: フル SWJ (JTAG-DP + SW-DP) (リセット時の状態)

001: フル SWJ (JTAG-DP + SW-DP)、JNTRST なし

010: JTAG-DP ディセーブル及び SW-DP イネーブル

100: JTAG-DP ディセーブル及び SW-DP ディセーブル

他の値の設定はできません。

Bits 23:21 予約済み

Bits 20 ADC2_ETRGREG_REMAP ADC 2 外部トリガ・レギュラー変換割当

ソフトウェアでセット / クリアができます。このビットは、ADC2 外部トリガ・レギュラー変換に接続されているトリガ入力を制御します。このビットがセットされると、ADC2 外部トリガ・レギュラー変換は EXTI11 へ接続されます。このビットがリセットされると、ADC2 外部イベントレギュラー変換は TIM8_TRGO へ接続されます。

Bits 19 ADC2_ETRGINJ_REMAP ADC 2 外部トリガ・インジェクト変換割当

ソフトウェアでセット / クリアができます。このビットは、ADC2 外部トリガ入力変換に接続されているトリガ入力を制御します。このビットがセットされると、ADC2 外部トリガ入力変換は EXTI15 へ接続されます。このビットがリセットされると、ADC2 外部イベント入力変換は TIM8_Channel4 へ接続されます。

Bits 18 ADC1_ETRGREG_REMAP ADC 1 外部トリガ・レギュラー変換割当

ソフトウェアでセット / クリアができます。このビットは、ADC1 外部トリガ・レギュラー変換に接続されているトリガ入力を制御します。このビットがセットされると、ADC1 外部トリガ・レギュラー変換は EXTI11 へ接続されます。このビットがリセットされると、ADC1 外部イベントレギュラー変換は TIM8_TRGO へ接続されます。

Bits 17 ADC1_ETRGINJ_REMAP ADC 1 外部トリガ入力変換割当

ソフトウェアでセット / クリアができます。このビットは、ADC1 外部トリガ入力変換に接続されているトリガ入力を制御します。このビットがセットされると、ADC1 外部トリガ入力変換は EXTI15 へ接続されます。このビットがリセットされると、ADC1 外部イベント入力変換は TIM8_Channel4 へ接続されます。

参考資料

RM0008

汎用入出力とオルタネート機能入出力(GPIO、AFIO)

Bits 16 TIM5CH4_IREMAP TIM5 チャネル 4 内部割当て

ソフトウェアでセット / クリアができます。このビットは、TIM5_CH4 内部割当てを制御します。リセットされると、TIM5_CH4 は PA3 へ接続されます。セットされると、LSI 内部クロックは較正の目的で TIM5_CH4 へ接続されます。

Bit 15 PD01_REMAP: OSC_IN/OSC_OUT 上のポート D0/D1 のピン割当て

このビットはソフトウェアでセット / クリアができます。このビットは GPIO 機能の PD0 と PD1 のピン割当てを設定します。HSE オシレータが使用されないとき（アプリケーションが内部の 8 MHz RC で実行されているとき）には、OSC_IN と OSC_OUT ピンは PD0 及び PD1 として使用できます。この割当ては 36、48、64 ピンパッケージで利用できます。（100 ピン及び 144 ピンのパッケージでは別に PD0、PD1 ピンがあるため、この割当ては使用しません。）

0: PD0、PD1 は OSC_IN、OSC_OUT ピンに割当てができません。

1: OSC_IN ピンは PD0 として、OSC_OUT ピンは PD1 として割当てます。

Bits 14:13 CAN_REMAP[1:0]: CAN オルタネート機能のピン割当

これらのビットはソフトウェアでセット / クリアができ、オルタネート機能 CANRX と CANTX のピン割当てができます。

00: CANRX として PA11 ピンが、CABTX として PA12 ピンを割当てます。

01: 予約値

10: CANRX として PB8 ピンが、CABTX として PB9 ピンを割当てます。

（36 ピンパッケージでは使用できません。）

11: CANRX として PD0 ピンが、CABTX として PD1 ピンを割当てます。

Bit 12 TIM4_REMAP: TIM4 機能のピン割当

このビットはソフトウェアでセット / クリアができ、GPIO ポートの TIM4 のチャネル 1 ~ 4 のピン割当てが設定されます。

0: 割当て変更なし (TIM4_CH1/PB6、TIM4_CH2/PB7、TIM4_CH3/PB8、TIM4_CH4/PB9)

1: 割当て変更あり (TIM4_CH1/PD12、TIM4_CH2/PD13、TIM4_CH3/PD14、TIM4_CH4/PD15)

注 : PE0 ピンに割当てられている TIM4_ETR は再割当できません。

Bits 11:10 TIM3_REMAP[1:0]: TIM3 機能のピン割当

これらのビットはソフトウェアでセット / クリアができ、GPIO ポートの TIM3 のチャネル 1 ~ 4 のピン割当てが設定できます。

00: 割当て変更なし (CH1/PA6、CH2/PA7、CH3/PB0、CH4/PB1)

01: この値は指定できません。

10: 部分的な割当変更 (CH1/PB4、CH2/PB5、CH3/PB0、CH4/PB1)

11: 全面的な割当変更 (CH1/PC6、CH2/PC7、CH3/PC8、CH4/PC9)

注 : PE0 ピンに割当てられている TIM3_ETR は変更できません。

Bits 9:8 TIM2_REMAP[1:0]: TIM2 機能のピン割当

これらのビットはソフトウェアでセット / クリアができ、GPIO ポートの TIM2 のチャネル 1 ~ 4 と外部トリガ (ETR) のピン割当てが設定できます。

00: 割当て変更なし (CH1/ETR/PA0、CH2/PA1、CH3/PA2、CH4/PA3)

01: 部分的な割当変更 (CH1/ETR/PA15、CH2/PB3、CH3/PA2、CH4/PA3)

10: 部分的な割当変更 (CH1/ETR/PA0、CH2/PA1、CH3/PB10、CH4/PB11)

11: 全面的な割当変更 (CH1/ETR/PA15、CH2/PB3、CH3/PB10、CH4/PB11)



参考資料

Bits 7:6 TIM1_REMAP[1:0]: TIM1 機能のピン割当て

これらのビットはソフトウェアでセット / クリアができ、GPIO の TIM2 のチャネル 1 ~ 4、1N ~ 3N と外部トリガ (ETR)、ブレーク入力 (BKIN) のピン割当てが設定できます。

00: 割当て変更なし (ETR/PA12、CH1/PA8、CH2/PA9、CH3/PA10、CH4/PA11、BKIN/PB12、CH1N/PB13、CH2N/PB14、CH3N/PB15)

01: 部分的な割当て変更 (ETR/PA12、CH1/PA8、CH2/PA9、CH3/PA10、CH4/PA11、BKIN/PA6、CH1N/PA7、CH2N/PB0、CH3N/PB1)

10: この値は指定できません。

11: 全面的な割当て変更 (ETR/PE7、CH1/PE9、CH2/PE11、CH3/PE13、CH4/PE14、BKIN/PE15、CH1N/PE8、CH2N/PE10、CH3N/PE12)

Bits 5:4 USART3_REMAP[1:0]: USART3 機能のピン割当て

これらのビットはソフトウェアでセット / クリアができ、UART3 の CTS、RTS、CK、TX、RX ピンの割当てが設定できます。

00: 割当て変更なし (TX/PB10、RX/PB11、CK/PB12、CTS/PB13、RTS/PB14)

01: 部分的な割当て変更 (TX/PC10、RX/PC11、CK/PC12、CTS/PB13、RTS/PB14)

10: この値は設定できません。

11: 全面的な割当て変更 (TX/PD8、RX/PD9、CK/PD10、CTS/PD11、RTS/PD12)

Bit 3 USART2_REMAP: USART2 機能のピン割当て

これらのビットはソフトウェアでセット / クリアができ、UART2 の CTS、RTS、CK、TX、RX ピンの割当てが設定できます。

0: 割当て変更なし (CTS/PA0、RTS/PA1、TX/PA2、RX/PA3、CK/PA4)

1: 割当て変更あり (CTS/PD3、RTS/PD4、TX/PD5、RX/PD6、CK/PD7)

Bit 2 USART1_REMAP: USART1 機能のピン割当て

これらのビットはソフトウェアでセット / クリアができ、UART1 の TX、RX ピンの割当てが設定できます。

0: 割当て変更なし (TX/PA9、RX/PA10)

1: 割当て変更あり (TX/PB6、RX/PB7)

Bit 1 I2C1_REMAP: I²C 1 機能のピン割当て

これらのビットはソフトウェアでセット / クリアができ、I²C 1 の SCL と SDA ピンの割当てが設定できます。

0: 割当て変更なし (SCL/PB6、SDA/PB7)

1: 割当て変更あり (SCL/PB8、SDA/PB9)

Bit 0 SPI1_REMAP: SPI 1 機能のピン割当て

これらのビットはソフトウェアでセット / クリアができ、SPI 1 の NSS、SCK、MISO、MOSI ピンの割当てが設定できます。

0: 割当て変更なし (NSS/PA4、SCK/PA5、MISO/PA6、MOSI/PA7)

1: 割当て変更あり (NSS/PA15、SCK/PB3、MISO/PB3、MOSI/PB5)

参考資料

RM0008

汎用入出力とオルタネート機能入出力(GPIO、AFIO)

7.4.3 外部割込み構成レジスタ 1 (AFIO_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
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 予約済み

Bits 15:0 **EXTIx[3:0]: EXTI x 構成 (x = 0 ~ 3)**

これらのビットには EXTIx 外部割込みのソースとなる信号の選択を、ソフトウェアで書込むことができます。詳細は [セクション 8.2.5: 外部割込み / イベントラインの割当て \(136 ページ\)](#) を参照してください。

- 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: PF[x] ピン
- 0110: PG[x] ピン

7.4.4 外部割込み構成レジスタ 2 (AFIO_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
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 予約済み

Bits 15:0 **EXTIx[3:0]: EXTI x 構成 (x = 4 ~ 7)**

これらのビットには EXTIx 外部割込みのソースとなる信号の選択を、ソフトウェアで書込むことができます。

- 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: PF[x] ピン
- 0110: PG[x] ピン



参考資料

7.4.5 外部割込み構成レジスタ3 (AFIO_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
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 予約済み

Bits 15:0 **EXTIx[3:0]: EXTI x 構成** ($x = 8 \sim 11$)

これらのビットには EXTIx 外部割込みのソースとなる信号の選択を、ソフトウェアで書込むことができます。

- 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: PF[x] ピン
- 0110: PG[x] ピン

7.4.6 外部割込み構成レジスタ4 (AFIO_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
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 予約済み

Bits 15:0 **EXTIx[3:0]: EXTI x 構成** ($x = 12 \sim 15$)

これらのビットには EXTIx 外部割込みのソースとなる信号の選択を、ソフトウェアで書込むことができます。

- 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: PF[x] ピン
- 0110: PG[x] ピン

參考資料

7.5 GPIO・AFIO レジスタ・マップ

レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。次の表は、GPIO と AFIO レジスタ・マップとリセット値を示しています。

表 33. GPIO - レジスタ・マップとリセット時の値

表 34. AFIO レジスタ・マップとリセット時の値

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	
	AFIO_EVCR	予約済み																								PORT[2:0]				PIN[3:0]				
	リセット値																										EVOE							
	AFIO_MAPR	予約済み																									PORT[2:0]				PIN[3:0]			
	リセット値																										EVOE							
0x00	AFIO_EVCR	予約済み																									PORT[2:0]				PIN[3:0]			
0x04	AFIO_MAPR	予約済み																									EVOE							
0x08	AFIO_EXTICR1	予約済み																									PORT[2:0]				PIN[3:0]			
0x0C	AFIO_EXTICR2	予約済み																									EVOE							
0x10	AFIO_EXTICR3	予約済み																									PORT[2:0]				PIN[3:0]			
0x14	AFIO_EXTICR4	予約済み																									EVOE							

8 割込みとイベント

Flash メモリ容量が 32 ~ 128 キロバイトの STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512 キロバイトの STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

8.1 ネスト化されたベクタ割込みコントローラ (NVIC)

主な機能

- 60 個のマスク可能な割込みチャネル (Cortex™-M3 の 16 個の割込みラインは数に含まれていません。)
- プログラム可能な 16 の優先レベル (4bit の優先割込みを使用)
- 遅延時間の少ない例外及び割込みハンドリング
- 電力管理制御
- システム制御レジスタの実装

NVIC とプロセッサコアのインターフェースとは密接に結びついており、割込み処理の遅延時間を小さくし、遅れてリクエストされた割込みを効率的に処理できるように工夫されています。

コアの例外を含むすべての割込みは NVIC で管理されます。例外と NVIC のプログラミングに関しては、ARM Cortex™-M3 テクニカル・リファレンス・マニュアルの「5 章 例外」と「8 章 ネスト化された割込みコントローラ」の章を参照してください。

8.1.1 SysTick 較正值レジスタ

SysTick 較正值は 9000 に固定されており、SysTick クロックが 9 MHz (最大値 HCLK/8) の時に 1 ms のタイムベースが生成されます。

8.1.2 割込み・例外ベクタ

表 35. ベクタテーブル

位置	優先度	優先種別	項目 (略称)	詳細	アドレス
-	-	-		予約済み	0x0000_0000
-3	固定	Reset		リセット	0x0000_0004
-2	固定	NMI		ノンマスカブル割込み。RCC クロック・セキュリティ・システム (CSS) はここに接続されています。	0x0000_0008
-1	固定	HardFault		すべての異常	0x0000_000C
0	設定可能	MemManage		メモリ管理	0x0000_0010

参考資料

表 35. ベクターテーブル (続き)

位置	優先度	優先種別	項目 (略称)	詳細	アドレス
	1	設定可能	BusFault	プリフェッч異常、メモリ・アクセス異常	0x0000_0014
	2	設定可能	UsageFault	未定義命令もしくは異常状態	0x0000_0018
	-	-	-	予約済み	0x0000_001C - 0x0000_002B
	3	設定可能	SVCall	SWI 命令によるシステム・サービスコール	0x0000_002C
	4	設定可能	Debug Monitor	デバッグ・モニタ	0x0000_0030
	-	-	-	予約	0x0000_0034
	5	設定可能	PendSV	ペンドイ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	システム・ティック・タイマ	0x0000_003C
0	7	設定可能	WWDG	Window 型ウォッチドッグ割込み	0x0000_0040
1	8	設定可能	PVD	PVD 検出割込み (EXTI 経由)	0x0000_0044
2	9	設定可能	TAMPER	タンパ (改ざん) 割込み	0x0000_0048
3	10	設定可能	RTC	RTC グローバル 割込み	0x0000_004C
4	11	設定可能	FLASH	Flash グローバル割込み	0x0000_0050
5	12	設定可能	RCC	RCC グローバル割込み	0x0000_0054
6	13	設定可能	EXTI0	EXTI ライン 0 割込み	0x0000_0058
7	14	設定可能	EXTI1	EXTI ライン 1 割込み	0x0000_005C
8	15	設定可能	EXTI2	EXTI ライン 2 割込み	0x0000_0060
9	16	設定可能	EXTI3	EXTI ライン 3 割込み	0x0000_0064
10	17	設定可能	EXTI4	EXTI ライン 4 割込み	0x0000_0068
11	18	設定可能	DMA_Channel1	DMA チャネル 1 グローバル割込み	0x0000_006C
12	19	設定可能	DMA_Channel2	DMA チャネル 2 グローバル割込み	0x0000_0070
13	20	設定可能	DMA_Channel3	DMA チャネル 3 グローバル割込み	0x0000_0074
14	21	設定可能	DMA_Channel4	DMA チャネル 4 グローバル割込み	0x0000_0078
15	22	設定可能	DMA_Channel5	DMA チャネル 5 グローバル割込み	0x0000_007C
16	23	設定可能	DMA_Channel6	DMA チャネル 6 グローバル割込み	0x0000_0080
17	24	設定可能	DMA_Channel7	DMA チャネル 7 グローバル割込み	0x0000_0084
18	25	設定可能	ADC1_2	ADC グローバル割込み	0x0000_0088
19	26	設定可能	USB_HP_CAN_TX	USB 優先割込み、もしくは CAN TX 割込み	0x0000_008C
20	27	設定可能	USB_LP_CAN_RX0	USB 非優先割込み、もしくは CAN RX0 割込み	0x0000_0090
21	28	設定可能	CAN_RX1	CAN RX1 割込み	0x0000_0094

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お問い合わせ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。

参考資料

表 35. ベクターテーブル (続き)

位置	優先度	優先種別	項目 (略称)	詳細	アドレス
22	29	設定可能	CAN_SCE	CAN SCE 割込み	0x0000_0098
23	30	設定可能	EXTI9_5	EXTI ライン [9:5] 割込み	0x0000_009C
24	31	設定可能	TIM1_BRK	TIM1 ブレーク割込み	0x0000_00A0
25	32	設定可能	TIM1_UP	TIM1 更新割込み	0x0000_00A4
26	33	設定可能	TIM1_TRG_COM	TIM1 トリガとコミュニケーション割込み	0x0000_00A8
27	34	設定可能	TIM1_CC	TIM1 キャプチャ / 比較割込み	0x0000_00AC
28	35	設定可能	TIM2	TIM2 グローバル割込み	0x0000_00B0
29	36	設定可能	TIM3	TIM3 グローバル割込み	0x0000_00B4
30	37	設定可能	TIM4	TIM4 グローバル割込み	0x0000_00B8
31	38	設定可能	I2C1_EV	I ² C1 イベント割込み	0x0000_00BC
32	39	設定可能	I2C1_ER	I ² C1 エラー割込み	0x0000_00C0
33	40	設定可能	I2C2_EV	I ² C2 イベント割込み	0x0000_00C4
34	41	設定可能	I2C2_ER	I ² C2 エラー割込み	0x0000_00C8
35	42	設定可能	SPI1	SPI1 グローバル割込み	0x0000_00CC
36	43	設定可能	SPI2	SPI2 グローバル割込み	0x0000_00D0
37	44	設定可能	USART1	USART1 グローバル割込み	0x0000_00D4
38	45	設定可能	USART2	USART2 グローバル割込み	0x0000_00D8
39	46	設定可能	USART3	USART3 グローバル割込み	0x0000_00DC
40	47	設定可能	EXTI15_10	EXTI ライン [15:10] 割込み	0x0000_00E0
41	48	設定可能	RTCAlarm	RTC アラーム割込み (EXTI 経由)	0x0000_00E4
42	49	設定可能	USBWakeups	サスPENDからのUSBウェークアップ割込み (EXTI 経由)	0x0000_00E8
43	50	設定可能	TIM8_BRK	TIM8 ブレーク割込み	0x0000_00EC
44	51	設定可能	TIM8_UP	TIM8 更新割込み	0x0000_00F0
45	52	設定可能	TIM8_TRG_COM	TIM8 トリガとコミュニケーション割込み	0x0000_00F4
46	53	設定可能	TIM8_CC	TIM8 キャプチャ・比較割込み	0x0000_00F8
47	54	設定可能	ADC3	ADC3 グローバル割込み	0x0000_00FC
48	55	設定可能	FSMC	FSMC グローバル割込み	0x0000_0100
49	56	設定可能	SDIO	SDIO グローバル割込み	0x0000_0104
50	57	設定可能	TIM5	TIM5 グローバル割込み	0x0000_0108
51	58	設定可能	SPI3	SPI3 グローバル割込み	0x0000_010C

参考資料

表 35. ベクターテーブル (続き)

位置	優先度	優先種別	項目 (略称)	詳細	アドレス
52	59	設定可能	UART4	UART4 グローバル割込み	0x0000_0110
53	60	設定可能	UART5	UART5 グローバル割込み	0x0000_0114
54	61	設定可能	TIM6	TIM6 グローバル割込み	0x0000_0118
55	62	設定可能	TIM7	TIM7 グローバル割込み	0x0000_011C
56	63	設定可能	DMA2_Channel1	DMA2 チャネル 1 グローバル割込み	0x0000_0120
57	64	設定可能	DMA2_Channel2	DMA2 チャネル 2 グローバル割込み	0x0000_0124
58	65	設定可能	DMA2_Channel3	DMA2 チャネル 3 グローバル割込み	0x0000_0128
59	66	設定可能	DMA2_Channel4_5	DMA2 チャネル 4 と DMA2 チャネル 5 グローバル割込み	0x0000_012C

8.2 外部割込み / イベントコントローラ (EXTI)

外部割込み / イベントコントローラは、割込みリクエスト / イベントを発生させる 19 本のエッジ検出回路を構成します。入力ラインは個別に割込み種別（パルス、ペンドイング）及びトリガ種別（立上り、立下り、両エッジ）を選択することができます。また、ラインごとにマスクをかけることもできます。ペンドイング・レジスタは割込みリクエスト・ラインの状態を保持しています。

8.2.1 主な機能

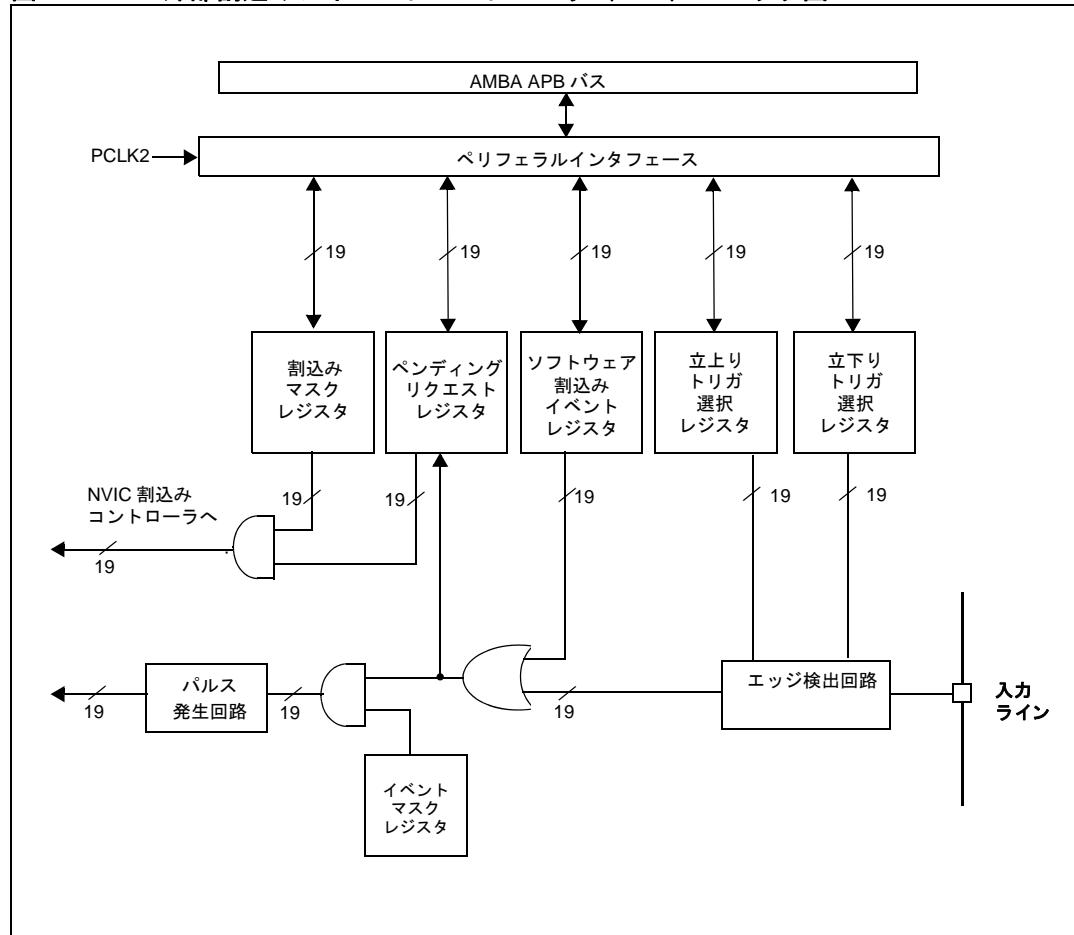
EXTI コントローラは次のような機能を持っています。

- 割込み / イベントラインごとに、独立したトリガとマスク設定が可能
- 割込みラインごとのステータス・ビットを装備
- 最大 19 本のソフトウェアイベント / 割込みリクエストの生成
- APB2 クロック周期より小さなパルス幅の外部信号の検出機能。具体的な詳細はデータシートの電気特性の項を参照してください。

8.2.2 ブロック図

ブロック構成は図 16 に示されています。

図 16. 外部割込み / イベントコントローラ (EXTI) ブロック図



8.2.3 ウェークアップ・イベント管理

Cortex™-M3 は外部イベントや内部イベントによりコア (WFE) をウェークアップさせることができます。外部ラインを入力ポートとして設定することで、RTC アラーム及び USB ウェークアップ・イベントを CPU のウェークアップ (WFE から終了) に利用することができます。

EXTI ラインをウェークアップ・イベントとして利用する方法については [セクション 8.2.4: 機能詳細](#) を参照してください。

8.2.4 機能詳細

割込みを生成するには、割込みリクエスト・ラインを設定し、有効にする必要があります。これには 2 個のトリガ・レジスタで必要なエッジ方向を指定し、割込みマスク・レジスタの対応すビットに 1 を書込み、リクエストを受付可能にします。外部割込みラインで指定されたエッジが生成されると割込みリクエストが発生し、ラインに対応するペンドイングビットがセットされます。リクエストはペンドイング・レジスタの対応ビットに 1 を書込むことでリセットすることができます。

イベントを生成されるには、イベントラインを設定し、有効にする必要があります。これには 2 個のトリガ・レジスタで必要なエッジ方向を指定し、イベントマスク・レジスタの対応すビットに 1 を書込み、リクエストを受付可能にします。イベントラインで指定されたエッジが生成されるとイベントパルスが発生します。このとき、イベントに対応するペンドイングビットはセットされません。

割込み / イベント・リクエストは、ソフトウェア割込み / イベント・レジスタの対応ビットに 1 を書込むことで、ソフトウェアから生成させることもできます。

ハードウェア割込みの選択

19 本の入力ラインを割込みリクエストとする設定は次のように行います。

- 19 本の割込みラインのマスク・ビットを設定します。(EXTI_IMR)
- 割込みラインのトリガ選択ビットの設定を行います。(EXTI_RTSR 及び EXTI_FTSR)
- NVIC IRQ チャネルが外部割込みコントローラ (EXTI) に割当てられるよう、イネーブル・ビット、マスク・ビットを設定し、19 本のラインからの割込みリクエストが正しく受けられるようにします。

ハードウェアイベントの選択

19 本の入力ラインをイベント要因とする設定は次のように行います。

- 19 本の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の設定を行います。(EXTI_RTSR, EXTI_FTSR)

ソフトウェア割込み / イベントの選択

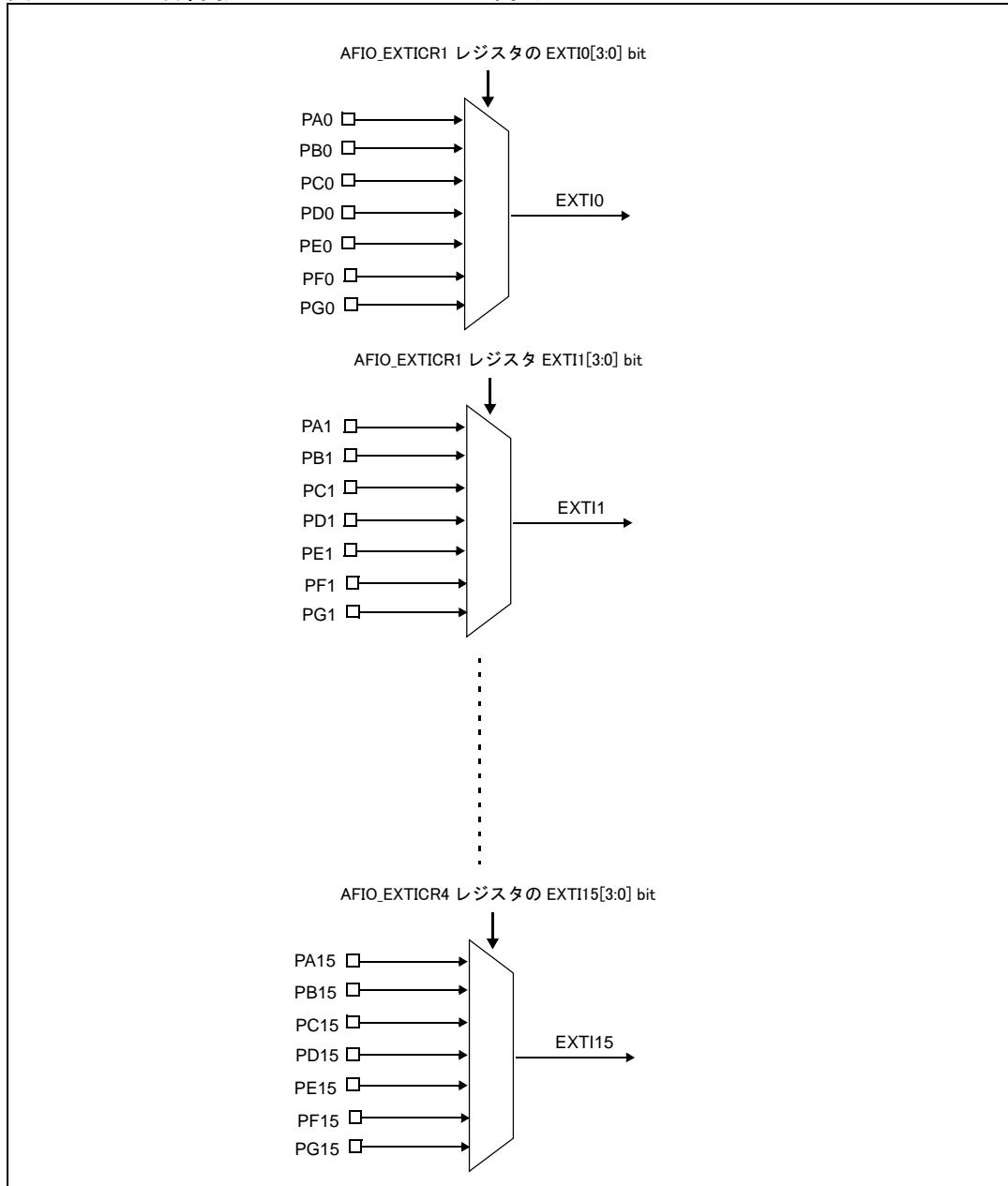
19 本の入力ラインは割込み / イベントラインのいずれかとして設定しておくことができます。ソフトウェア割込みは次の操作で生成します。

- 割込み / イベントラインのマスク・ビットを設定します。(EXTI_IMR, EXTI_EMR)
- ソフトウェア割込みレジスタ (EXTI_SWIER) の所定のビットをセットします。

8.2.5 外部割込み / イベントラインの割当て

112 本の GPIO 信号は、次の図のように、16 の外部割込み / イベントラインに割振られます。

図 17. 外部割込み / イベントの GPIO 割当て



他の 3 本の EXTI ラインは次のように接続されています。

- EXTI ライン 16 は PVD 出力と接続されています。
- EXTI ライン 17 は RTC アラームイベントと接続されています。
- EXTI ライン 18 は USB ウエークアップ・イベントと接続されています。

参考資料

RM0008

割込みとイベント

8.3 EXTI レジスタ

レジスタの詳細記述で使用されている略語については、[セクション 1.1 \(32 ページ\)](#) を参照してください。

8.3.1 割込みマスク・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														MR18	MR17	MR16
														rw	rw	rw
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	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 31:19 予約済み、常にリセット時の値 (0) を保持してください。

Bits 18:0 MR_x: 割込みマスク (ライン _x)

0: ライン _x の割込みリクエストは禁止 (マスク) されています。

1: ライン _x の割込みリクエストは許可されています。

8.3.2 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														MR18	MR17	MR16
														rw	rw	rw
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	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 31:19 予約済み、常にリセット時の値 (0) を保持してください。

Bits 18:0 MR_x: ライン _x上のイベント・マスク

0: ライン _x のイベント・リクエストは禁止 (マスク) されています。

1: ライン _x のイベント・リクエストは許可されています。



参考資料

割込みとイベント

RM0008

8.3.3 立上り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														TR18	TR17	TR16
														RW	RW	RW
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	
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

Bits 31:19 予約済み、常にリセット時の値(0)を保持してください。

Bits 18:0 TRx: ライン x 上の立上りエッジ・トリガ

- 0: 入力ラインの(イベントと割込みについて)立上りエッジはトリガとして有効ではありません。
1: 入力ラインの(イベントと割込みについて)立上りエッジはトリガとして有効です。

注意: 外部ウェークアップ・ラインはエッジ・トリガです。グリッチが無いようにする必要があります。EXTI_RTSR レジスタへの書き込み中に外部割込みラインの立上りエッジが発生したときには、ペンドイングビットはセットされません。

同じ割込みラインで、立上りエッジと立下りエッジの双方をセットすることができます。このときには、いずれのエッジであってもトリガ条件となります。

8.3.4 立下り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														TR18	TR17	TR16
														RW	RW	RW
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	
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	

Bits 31:19 予約済み、常にリセット時の値(0)を保持してください。

Bits 18:0 TRx: 立下りエッジトリガ (ライン x)

- 0: 入力ラインの(イベントと割込みについて)立下りエッジはトリガとして有効ではありません。
1: 入力ラインの(イベントと割込みについて)立下りエッジはトリガとして有効です。

注意: 外部ウェークアップ・ラインはエッジ・トリガです。グリッチが無いようにする必要があります。EXTI_FTSR レジスタへの書き込み中に外部割込みラインの立下りエッジが発生したときには、ペンドイングビットはセットされません。

同じ割込みラインで、立上りエッジと立下りエッジの双方をセットすることができます。このときには、いずれのエッジであってもトリガ条件となります。

参考資料

RM0008

割込みとイベント

8.3.5 ソフトウェア割込みイベント・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														SWIER 18	SWIER 17	SWIER 16
														rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SWIER 15	SWIER 14	SWIER 13	SWIER 12	SWIER 11	SWIER 10	SWIER 9	SWIER 8	SWIER 7	SWIER 6	SWIER 5	SWIER 4	SWIER 3	SWIER 2	SWIER 1	SWIER 0	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 31:19 予約済み、常にリセット時の値(0)を保持してください。

Bits 18:0 SWIERx: ライン_x上のソフトウェア割込み

このビットが0の時に1を書込むと、EXTI_PRレジスタの対応するペンドィングビットがセットされます。EXTI_IMR及びEXTI_EMRレジスタで、このラインの割込みリクエストが許可されれば、割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を書込む）と、このビットはクリアされます。

8.3.6 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 : 0xxxxx xxxx

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
予約済み														PR18	PR17	PR16
														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																

Bits 31:19 予約済み、常にリセット時の値(0)を保持してください。

Bits 18:0 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: トリガリクエストが発生しました。

外部割込みラインで指定されているトリガ条件が発生したとき、このビットがセットされます。このビットに1を書込んだとき、もしくはエッジ検出回路の設定を変更したとき、このビットはクリアされます。

注: STOPモードに入る1サイクル前に割込みリクエストが発生した時には、STOPモードから終了した時点でEXTI_PRレジスタが更新され、EXTI_IMRレジスタの対応ビットで許可されていれば割込みリクエストが発生します。



参考資料

8.3.7 EXTI レジスタ・マップ

次の表は EXTI レジスタ・マップとリセット値を示しています。

表 36. 外部割込み / イベントコントローラ (EXTI) レジスタ・マップとリセット時の値

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																																						
	リセット値																																						
0x04	EXTI_EMR																																						
	リセット値																																						
0x08	EXTI_RTSR																																						
	リセット値																																						
0x0C	EXTI_FTSR																																						
	リセット値																																						
0x10	EXTI_SWIER																																						
	リセット値																																						
0x14	EXTI_PR																																						
	リセット値																																						

レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。

9 DMA コントローラ (DMA)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

9.1 概要

DMA (Direct Memory Access) は、ペリフェラルとメモリ間、及びメモリ内で高速のデータ転送を行うために利用されます。DMA を利用することで CPU の動作を要せずデータをすばやく移動することができます。これにより、CPU の能力を他の動作のためにあけておくことができます。

DMA コントローラは合計 12 チャネル（7 つの DMA1、5 つの DMA2）を搭載し、それぞれがペリフェラルからのメモリ・アクセスを管理します。また、コントローラは DMA リクエスト間のアクセス順序を管理するアービタを内蔵しています。

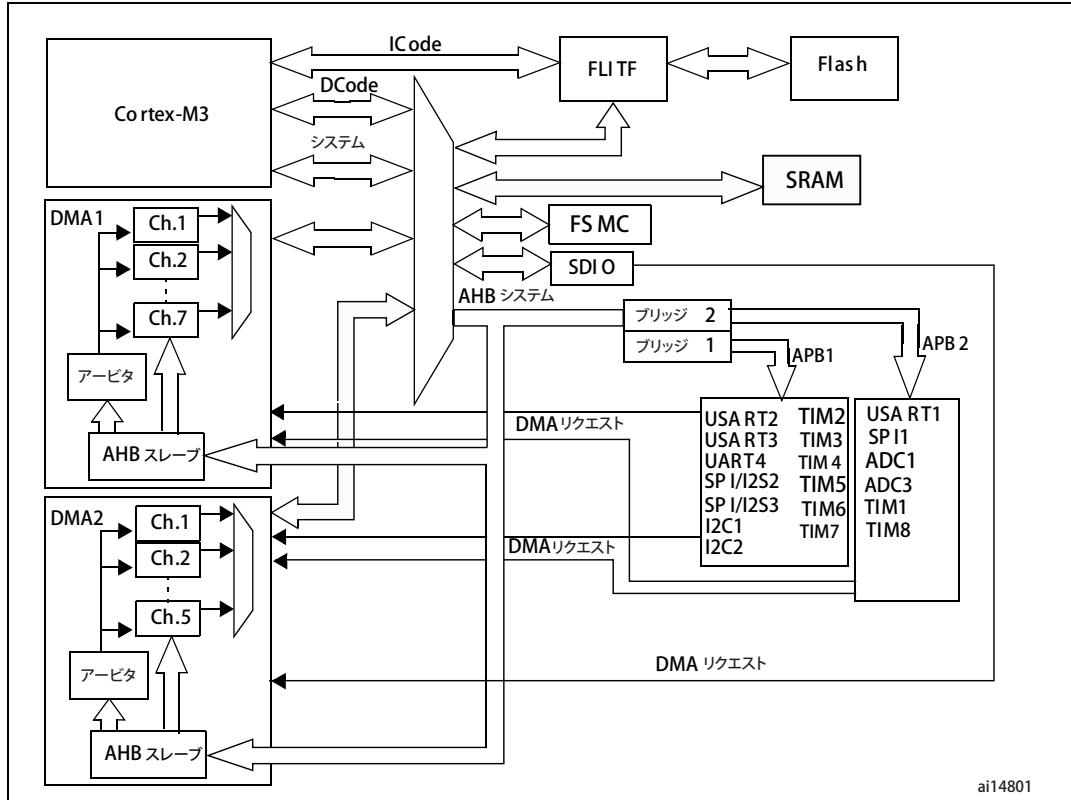
9.2 DMA の主な機能

- 個別に設定可能な 12 チャネル（リクエスト）：7 つの DMA1 及び 5 つの DMA2
- 12 チャネルはそれぞれに固有のハードウェア DMA リクエストと接続されています。また、ソフトウェア・トリガは各チャネルでサポートされています。設定はソフトウェアで行われます。
- 1 つの DMA のチャネル間のリクエストの優先度は、ソフトウェアで指定できます。（4 レベル、最高、高、中、低）また、ハードウェアによりチャネル番号の順とすることも可能です。（リクエスト 1 がリクエスト 2 に優先、など）
- 転送元と転送先で転送幅（バイト、ハーフワード、ワード）を個別に指定でき、この際にデータのパッキング、アンパッキングが行われます。
- サーキュラ・バッファ管理のサポート
- 3 個のイベントフラグ（DMA 1/2 転送終了、DMA 転送完了、DMA 転送エラー）、これらの論理和はチャネルごとに割込みリクエストを発生
- メモリ間転送機能
- ペリフェラルからメモリ、メモリからペリフェラル、及びペリフェラル相互間の転送機能
- Flash メモリ、SRAM、ペリフェラル SRAM、APB1、APB2 及び AHB ペリフェラルへのアクセス（転送元 / 転送先として）
- 最大 65536 まで、転送データ数の設定が可能

ブロック構成は [図 18](#) に示されています。

参考資料

図 18. DMA ブロック構成



1. DMA2 コントローラは大容量デバイスにのみ搭載されています。
2. ADC3、SPI/I2S3、UART4、SDIO、TIM5、TIM6、DAC、TIM7、TIM8 DMA リクエストは大容量デバイスでのみ備えられています。

9.3 DMA の機能詳細

DMA コントローラはシステム・バスを CortexTM-M3 コアと共に用しながら DMA 転送を実行します。CPU と DMA が同じ転送先 (RAM もしくはペリフェラル) のアクセスを試みたとき、CPU からシステム・バスへのアクセス動作が 1 バスサイクルの間、DMA によって停止させられることがあります。バス・マトリックスはラウンドロビン方式で競合を解決していますので、(RAM 及びペリフェラルの両方への) システム・バス帯域の少なくとも 1/2 が CPU のために確保されます。

9.3.1 DMA トランザクション

所定のイベント後、ペリフェラルは DMA コントローラにリクエスト信号を送ります。 DMA コントローラはチャネル間の優先順位に従って、これらのリクエストを処理します。 DMA コントローラがペリフェラルにアクセスすると、アクノレッジ信号がペリフェラルに送られます。ペリフェラルは DMA コントローラからのアクノレッジを受け取ると、DMA リクエスト信号を解除します。ペリフェラルからの DMA リクエストが解除されると、 DMA コントローラはアクノレッジ信号を解除します。別の DMA 転送が必要であれば、ペリフェラルは次のトランザクションを開始することができます。

個々の DMA 転送は次の 3 動作から構成されています。

- ペリフェラル・データ・レジスタ、もしくはメモリ上の所定の位置から、DMA_CMARx レジスタにデータをロードします。
- ロードされたデータを DMA_CMARx レジスタからペリフェラルのデータ・レジスタ、もしくはメモリ上の所定の位置に格納します。
- このあと予定される DMA トランザクションの数を保持している DMA_CNDTRx レジスタをカウントダウンします。

9.3.2 アービタ

アービタはアクセスリクエストをチャネルの優先順位に応じて管理し、ペリフェラルやメモリに対するアクセス動作を開始させます。

優先順位は 2 段階で管理されます。

- ソフトウェア：優先順位は DMA_CCRx レジスタで 4 段階のいずれかをチャネルごとに設定できます。
 - 最優先
 - 高優先度
 - 中優先度
 - 低優先度
- ハードウェア：ソフトウェア優先度が同じレベルのリクエストが複数ある場合、番号の小さなチャネルが優先します。たとえばチャネル 2 はチャネル 4 より優先されることになります。

注意：大容量デバイスでは、DMA1 コントローラは DMA2 コントローラより高い優先順位を持っています。

9.3.3 DMA チャネル

各チャネルは固定されたアドレスのペリフェラルレジスタとメモリとの間の DMA 転送を扱うことができます。転送されるデータの総量（最大 65535）はプログラムで設定可能です。この数を保持しているレジスタは、転送トランザクションごとにカウントダウンされます。

データ・サイズの設定

ペリフェラルやメモリとの転送時のデータ・サイズは MDA_CCRx の PSIZE 及び MSIZE フィールドで任意に設定できます。

ポインタの加算

ペリフェラルやメモリのポインタは、各転送動作のあとにカウントされるよう、MDA_CCRx の PINC 及び MINC で指定することができます。カウントアップが行われる場合、設定されているデータ・サイズに応じて、1、2、4 のいずれかが加算されます。最初の転送のアドレスとしてはソフトウェアで DMA_CPARx/DMA_CMARx に設定された値が使用されます。

チャネルの設定がサーキュラ・モードでないときには、転送終了後（転送データ数が 0 までカウントダウンされた後）は DMA リクエストに対する転送動作は行われません。

チャネル設定手順

DMA チャネルの設定は、次の手順で行います。(x はチャネル番号です。)

1. DMA_CPARx レジスタにペリフェラルレジスタのアドレスを設定します。ペリフェラルイベントごとに、このレジスタとメモリの間でデータが転送されることになります。
2. DMA_CPARx レジスタにメモリのアドレスを設定します。ペリフェラルイベントごとに、このメモリに書込み、もしくはこのメモリから読み出されることになります。
3. 転送されるデータの数を DMA_CNDTRx レジスタに設定します。個々のペリフェラルイベント後にこの値がカウントダウンされます。
4. DMA_CCRx レジスタの PL[1:0] フィールドでチャネルの優先度を指定します。
5. データ転送の方向、サーキュラ・モードの有無、ペリフェラルとメモリのアドレスカウントの有無、アクセス時のデータ・サイズ、1/2 終了もしくは全終了後の割込みの有無を DMA_CCRx レジスタに設定します。
6. DMA_CCRx レジスタの ENABLE ビットをセットし、チャネルの動作を有効とします。

チャネル動作が有効になると、そのチャネルに接続されているペリフェラルからの DMA リクエストにこたえることができます。

1/2 転送終了割込み許可ビット (HTIE) がセットされているとき、転送バイト数の半分が転送されると 1/2 転送終了割込みフラグ (HTIF) がセットされ、割込みリクエストが発生します。すべての転送が完了したときには、転送完了割込み許可ビット (TCIE) がセットされていれば 転送完了フラグ (TCIF) がセットされ、割込みリクエストが発生します。

サーキュラ・モード

サーキュラ・モードを指定することで、サーキュラ・バッファを管理し、連続的なデータの流れ (ADC スキャン・モードなど) を扱うことができます。この機能は DMA_CCRx レジスタの CIRC ビットをセットすることで有効となります。サーキュラ・モードが指定されると、チャネル設定時に指定された転送データ数は自動的にデータ数レジスタに再ロードされ、DMA リクエストは続けて処理されます。

メモリ間転送モード

DMA チャネルはペリフェラルからのリクエストによるトリガがなくとも動作させることができます。このモードはメモリ間転送と称されます。

DMA_CCRx レジスタの MEM2MEM ビットがセットされると、ソフトウェアで DMA_CCRx レジスタのイネーブル・ビット (EN) がセットされると同時にチャネルの動作が開始されます。転送は DMA_CNDTRx レジスタの値が 0 にカウントダウンされると停止します。メモリ間転送モードとサーキュラ・モードとを同時に指定することはできません。

9.3.4 エラー管理

DMA による読み込みもしくは書き込みアクセスでバスエラーが発生したとき、ハードウェアで対応するチャネル構成レジスタ (DMA_CCRx) の EN ビットをクリアするため、エラーを生じたチャネルは自動的にディセーブル状態となります。DMA_CCRx レジスタの 転送エラー割込み許可ビット (TEIE) がセットされていれば、DMA_IFR レジスタのチャネルに対応する転送エラー割込みフラグ (TEIF) がセットされ、割込みリクエストが発生します。

9.3.5 割込み

割込みは各 DMA チャネル毎に、ハーフトランスファ、トランスファ完了またはトランスファエラーにより発生します。高い融通性のために、別々の割込みを備えています。

表 37. DMA 割込みリクエスト

割込みイベント	イベント・フラグ	イネーブル制御ビット
1/2 転送	HTIF	HTIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

注意：大容量デバイスでは、DMA2 チャネル 4 と DMA2 チャネル 5 は、同じ割込みベクタへ割り当てられます。DMA1 と DMA2 の他の全ての割込みは、それぞれの割込みベクタを持っています。

9.3.6 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[1,2,3,4]、ADC1、SPI1、SPI/I2S2、I2Cx[1,2] 及び USARTx[1,2,3]) からの 7 種のリクエストは DMA1 コントローラに渡される前に論理和がとられます。このため、同時に 1 種類のリクエストのみが有効とされなければなりません [図 19: DMA1 リクエストのマッピング](#) を参照してください。

ペリフェラルからの DMA リクエストは、独自に、個々のペリフェラルのレジスタの DMA 制御ビットの設定で、許可 / 禁止することができます。

参考資料

図 19. DMA1 リクエストのマッピング

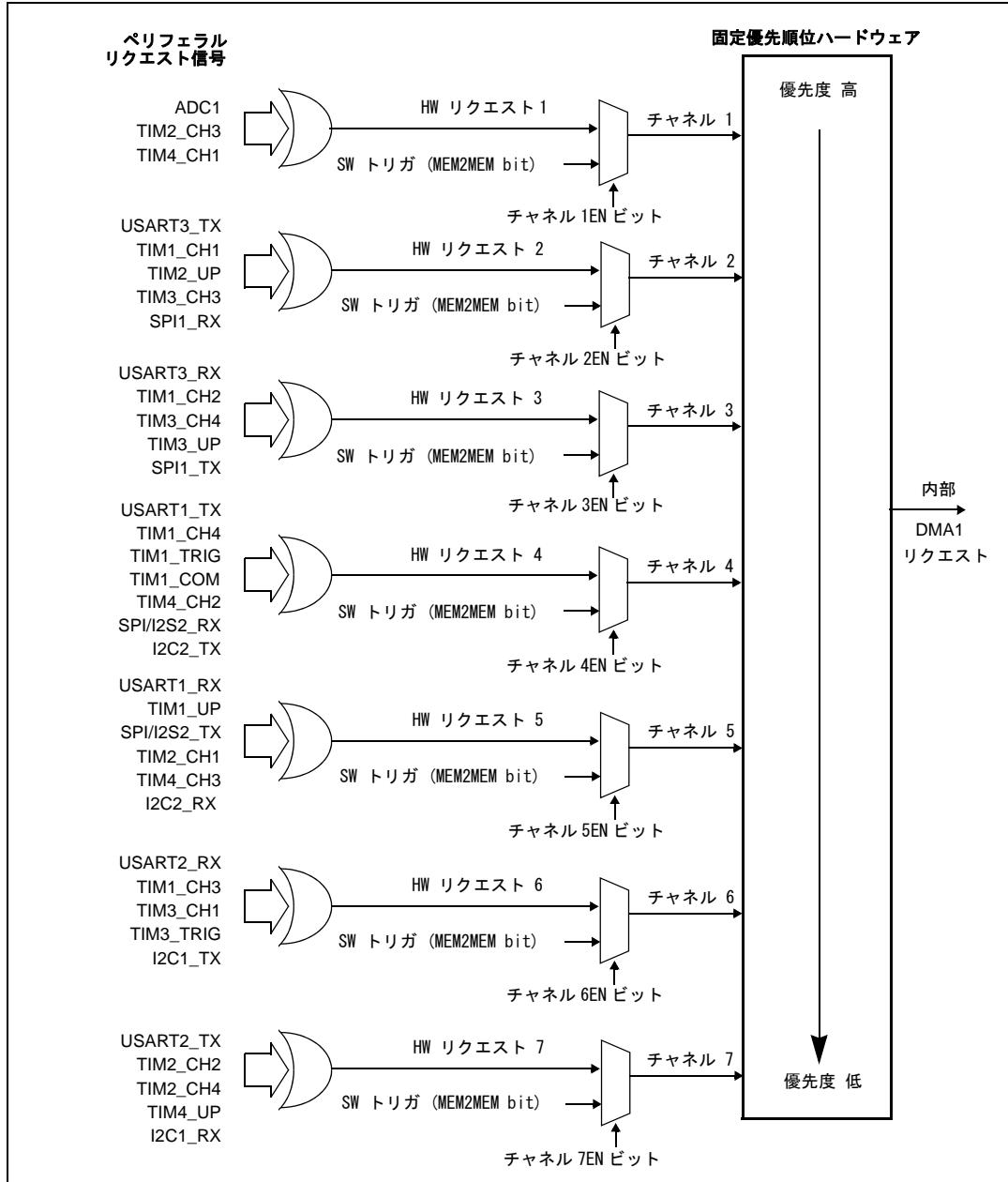


表 38 には各チャネルに対する DMA リクエストが表示されています。

参考資料

表 38. 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/I ² S		SPI1_RX	SPI1_TX	SPI/I2S2_RX	SPI/I2S2_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
TIM1		TIM1_CH1	TIM1_CH2	TIM1_CH4 TIM1_TRIG TIM1_COM	TIM1_UP	TIM1_CH3	
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

DMA2 コントローラ

ペリフェラルからの 5 種のリクエスト (TIMx[5,6,7,8]、ADC3、SPI/I2S3、UART4、DAC_Channel[1,2] と SDIO) は、DMA2 へ渡される前に論理和がとられます。このため、同時に 1 種類のリクエストのみが有効とされなければなりません [図 20: DMA2 リクエストのマッピング](#) を参照してください。

ペリフェラルからの DMA リクエストは、独自に、個々のペリフェラルのレジスタの DMA 制御ビットの設定で、許可 / 禁止することができます。

注意 : DMA2 と関連リクエストは、大容量デバイスでのみ利用できます。

参考資料

図 20. DMA2 リクエストのマッピング

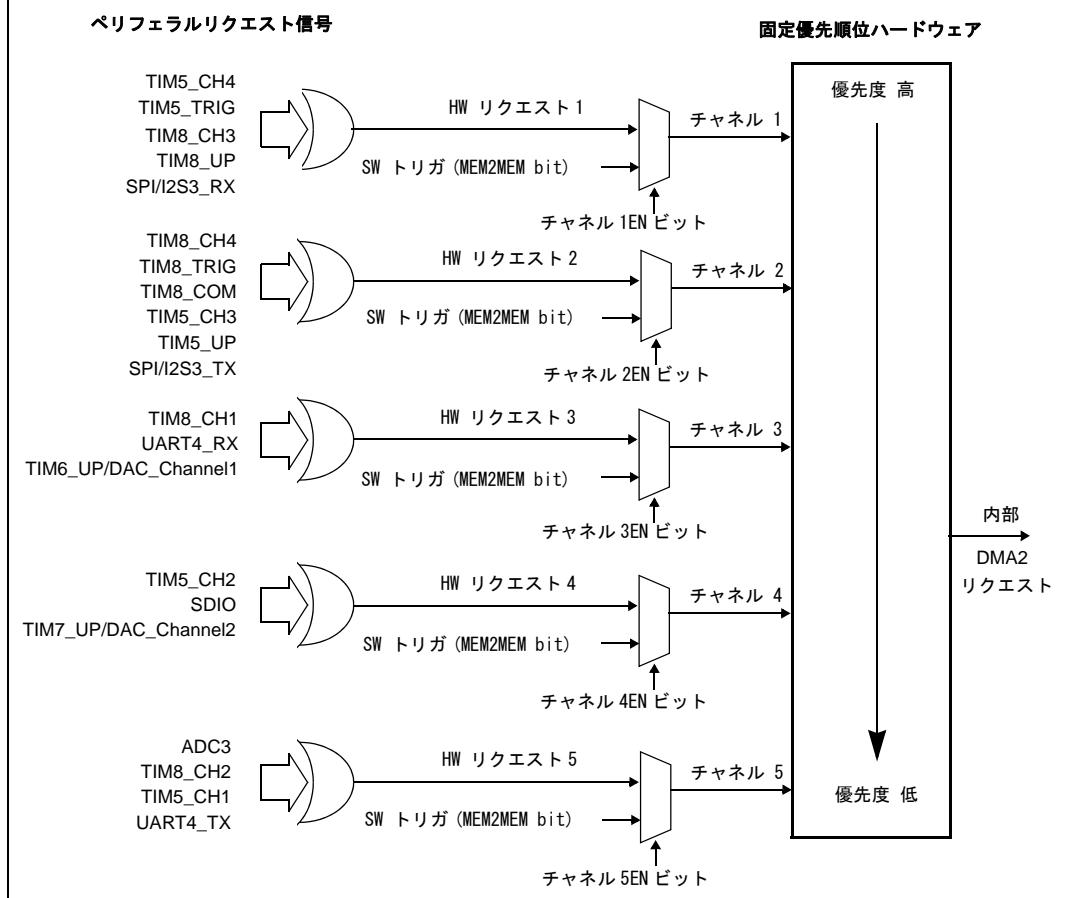


表 39 には各チャネルに対する DMA2 リクエストが表示されています。

表 39. チャネルに対する DMA2 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
ADC3					ADC3
SPI/I2S3	SPI/I2S3_RX	SPI/I2S3_TX			
UART4			UART4_RX		UART4_TX
SDIO				SDIO	
TIM5	TIM5_CH4 TIM5_TRIG	TIM5_CH3 TIM5_UP		TIM5_CH2	TIM5_CH1
TIM6/ DAC_Channel1			TIM6_UP/ DAC_Channel1		
TIM7/ DAC_Channel2				TIM7_UP/ DAC_Channel2	
TIM8	TIM8_CH3 TIM8_UP	TIM8_CH4 TIM8_TRIG TIM8_COM	TIM8_CH1		TIM8_CH2

9.4 DMA レジスタ

レジスタの詳細記述で使用されている略語については、[セクション 1.1 \(32 ページ\)](#) を参照してください。

注意： DMA2 は 5 つのチャネルしか持たないので、次に示すレジスタでは、チャネル 6 とチャネル 7 関連の全ビットは、DMA2 には適応されません。

9.4.1 DMA 割込みステータス・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	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

Bits 31:28 予約済み、常に 0 が読み出されます。

Bits 27, 23, TEIF x : チャネル x 転送エラー・フラグ ($x = 1..7$)

19, 15, 11, このビットはハードウェアでセットされます。DMA_IFCR レジスタの対応するビットに 1 を書込む
7, 3 ことで、このビットをクリアすることができます。
0: チャネル x で、転送エラー (TE) は生じていません。
1: チャネル x で、転送エラー (TE) が発生しました。

Bits 26, 22, HTIF x : チャネル x 1/2 転送終了フラグ ($x = 1..7$)

18, 14, 10, このビットはハードウェアでセットされます。DMA_IFCR レジスタの対応するビットに 1 を書込む
6, 2 ことで、このビットをクリアすることができます。
0: チャネル x で、1/2 転送終了 (HT) イベントは発生していません。
1: チャネル x で、1/2 転送終了 (HT) イベントが発生しました。

Bits 25, 21, TCIF x : チャネル x 転送完了フラグ ($x = 1..7$)

17, 13, 9, 5, このビットはハードウェアでセットされます。DMA_IFCR レジスタの対応するビットに 1 を書込む
1 ことで、このビットをクリアすることができます。
0: チャネル x で、転送完了 (TC) イベントは発生していません。
1: チャネル x で、転送完了 (TC) イベントが発生しました。

Bits 24, 20, GIF x : チャネル x 割込みリクエスト・フラグ ($x = 1..7$)

16, 12, 8, 4, このビットはハードウェアでセットされます。DMA_IFCR レジスタの対応するビットに 1 を書込む
0 ことで、このビットをクリアすることができます。
0: チャネル x で、TE, HT, TC イベントはいずれも発生していません。
1: チャネル x で、TE, HT, TC イベントのいずれかが発生しました。

参考資料

9.4.2 DMA 割込みフラグ・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				CTEIF 7	CHTIF 7	CTCIF 7	CGIF 7	CTEIF 6	CHTIF 6	CTCIF 6	CGIF 6	CTEIF 5	CHTIF 5	CTCIF 5	CGIF 5
				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
CTEIF 4	CHTIF 4	CTCIF 4	CGIF 4	CTEIF 3	CHTIF 3	CTCIF 3	CGIF 3	CTEIF 2	CHTIF 2	CTCIF 2	CGIF 2	CTEIF 1	CHTIF 1	CTCIF 1	CGIF 1
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Bits 31:28 予約済み、常に 0 が読み出されます。

Bits 27, 23, **CTEIFx**: チャネル x 転送エラークリア ($x = 1..7$)

19, 15, 11, 7, このビットはソフトウェアでセット / クリアができます。

3 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。

Bits 26, 22, **CHTIFx**: チャネル x 1/2 転送終了クリア ($x = 1..7$)

18, 14, 10, 6, このビットはソフトウェアでセット / クリアができます。

2 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。

Bits 25, 21, **CTCIFx**: チャネル x 転送完了クリア ($x = 1..7$)

17, 13, 9, 5, 1 このビットはソフトウェアでセット / クリアができます。

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。

Bits 24, 20, **CGIFx**: チャネル x 割込みリクエストクリア ($x = 1..7$)

16, 12, 8, 4, 0 このビットはソフトウェアでセット / クリアができます。

0: 影響なし

1: DMA_ISR レジスタの、対応するチャネルの GIF、TEIF、HTIF、TCIF フラグをクリアします。

参考資料

9.4.3 DMA チャネル x 構成レジスタ (DMA_CCRx) ($x = 1 \dots 7$)

アドレスオフセット : $0x08 + 20d \times \text{チャネル番号}$

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:15 予約済み、常に 0 が読み出されます。

Bit 14 MEM2MEM: メモリ間転送モード

このビットはソフトウェアでセット / クリアができます。

- 0: メモリ間転送モードは指定されていません。
- 1: メモリ間転送モードが有効です。

Bits 13:12 PL[1:0]: チャネル優先レベル

このビットはソフトウェアでセット / クリアができます。

- 00: 低
- 01: 中
- 10: 高
- 11: 最優先

Bits 11:10 MSIZE[1:0]: メモリ・サイズ

このビットはソフトウェアでセット / クリアができます。

- 00: 8 ビット
- 01: 16 ビット
- 10: 32 ビット
- 11: 予約値

Bits 9:8 PSIZE[1:0]: ペリフェラル・サイズ

このビットはソフトウェアでセット / クリアができます。

- 00: 8 ビット
- 01: 16 ビット
- 10: 32 ビット
- 11: 予約値

Bit 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でセット / クリアができます。

- 0: メモリ・インクリメント・モードは指定されていません。
- 1: メモリ・インクリメント・モードは有効です。

Bit 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でセット / クリアができます。

- 0: ペリフェラル・インクリメント・モードは指定されていません。
- 1: ペリフェラル・インクリメント・モードは有効です。

参考資料

Bit 5 CIRC: サーキュラ・モード

このビットはソフトウェアでセット / クリアができます。

0: サーキュラ・モードは指定されていません。

1: サーキュラ・モードは有効です。

Bit 4 DIR: データ転送方向

このビットはソフトウェアでセット / クリアができます。

0: ペリフェラルから読出されます。(ペリフェラル→メモリ)

1: メモリから読出されます。(メモリー→ペリフェラル)

Bit 3 TEIE: 転送エラー (TE) 割込みリクエスト許可

このビットはソフトウェアでセット / クリアができます。

0: TE 割込みリクエスト禁止

1: TE 割込みリクエスト許可

Bit 2 HTIE: 1/2 転送終了 (HT) 割込みリクエスト許可

このビットはソフトウェアでセット / クリアができます。

0: HT 割込みリクエスト禁止

1: HT 割込みリクエスト許可

Bit 1 TCIE: 転送完了割込み (TC) リクエスト許可

このビットはソフトウェアでセット / クリアができます。

0: TC 割込みリクエスト禁止

1: TC 割込みリクエスト許可

Bit 0 EN: チャネル・イネーブル

このビットはソフトウェアでセット / クリアができます。

0: チャネルはディセーブル状態です。

1: チャネルはイネーブル状態です。

9.4.4 DMA チャネル x データ数レジスタ (DMA_CNDTRx) (x = 1 ..7)

アドレスオフセット t: 0x0C + 20d × チャネル番号

リセット値 : 0x0000 0000

Bits 31:16 予約済み、常に 0 が読み出されます。

Bits 15:0 NDT[15:0]: 転送されるデータ数

転送されるデータの数 (0 から 最大 65535 バイト)。このチャネルへの書込みは、チャネルがディセーブル時に限られます。一旦チャネルが有効 (イネーブル) となると、このレジスタは読み出し専用となり、まだ転送されていないバイト数を示します。このレジスタの値は個々の DMA 転送動作後にカウントダウンされます。

すべての転送が終了すると、通常は、このレジスタの値は 0 にとどまります。ただし、自動再ロード・モードが設定されている時には、先にソフトウェアで設定した値が再びロードされます。

このレジスタの値が 0 のときには、チャネルがイネーブルか否かにかかわらず、DMA リクエストに応えることはありません。

参考資料

RM0008

DMA コントローラ (DMA)

9.4.5 DMA チャネル x ペリフェラル・アドレス・レジスタ (DMA_CPARx) (x = 1 ..7)

アドレスオフセット : $0x10 + 20d \times \text{チャネル番号}$

リセット値 : 0x0000 0000

Bits 31:0 PA[31:0]: ペリフェラル・アドレス

データが読出され、もしくは書込まれるペリフェラルレジスタのアドレス。(アドレスがカウントアップされる場合は、先頭アドレス)

9.4.6 DMA チャネル x メモリ・アドレス・レジスタ (DMA_CMARx) (x = 1 ..7)

アドレスオフセット : $0x14 + 20d \times \text{チャネル番号}$

リセット値 : 0x0000 0000

Bits 31:0 MA[31:0]: メモリ・アドレス

データが読出され、もしくは書込まれるメモリ領域のベース・アドレス。

9.4.7 DMA レジスタ・マップ

次の表は DMA レジスタ・マップとリセット値の一覧です。

表 40. DMA - レジスタ・マップとリセット時の値

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 リセット値	予約済み	0	0	TEIF7	HTIF7	TCIF7	GIF7	0	TEIF6	HTIF6	TCIF6	GIF6	0	TEIF5	HTIF5	TCIF5	GIF5	0	TEIF4	HTIF4	TCIF4	GIF4	0	TEIF3	HTIF3	TCIF3	GIF3	0	TEIF2	HTIF2	TCIF2	GIF2	0
0x004	DMA_IFCR リセット値	予約済み	0	0	CTEIF7	CHTIF7	CTCIF7	CGIF7	0	CTEIF6	CHTIF6	CTCIF6	CGIF6	0	CTEIF5	CHTIF5	CTCIF5	CGIF5	0	CTEIF4	CHTIF4	CTCIF4	CGIF4	0	CTEIF3	CHTIF3	CTCIF3	CGIF3	0	CTEIF2	CHTIF2	CTCIF2	CGIF2	0
0x008	DMA_CCR1 リセット値	予約済み	0	0	MEM2MEM	PL [1:0]	M SIZE [1:0]	PSIZE [1:0]	0	MEM2MEM	PL [1:0]	M SIZE [1:0]	PSIZE [1:0]	0	MEM2MEM	PL [1:0]	M SIZE [1:0]	PSIZE [1:0]	0	MEM2MEM	PL [1:0]	M SIZE [1:0]	PSIZE [1:0]	0	MEM2MEM	PL [1:0]	M SIZE [1:0]	PSIZE [1:0]	0	MEM2MEM	PL [1:0]	M SIZE [1:0]	PSIZE [1:0]	0
0x00C	DMA_CNDTR1 リセット値	予約済み	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x010	DMA_CPAR1 リセット値	PA[31:0]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x014	DMA_CMAR1 リセット値	MA[31:0]	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x018		予約済み	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x01C	DMA_CCR2 リセット値	予約済み	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x020	DMA_CNDTR2 リセット値	予約済み	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			



參考資料

表 40. DMA - レジスタ・マップとリセット時の値 (続き)

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。

參考資料

表 40. DMA - レジスタ・マップとリセット時の値 (続き)

レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。

10 A/D コンバータ (ADC)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

10.1 概要

ADC ペリフェラル回路の A/D コンバータは 12bit 逐次比較方式で、マルチプレクサで切り替えられる最大 18 チャネルの入力は、16 個の外部信号と 2 個の内部ソースの測定を行うことができます。複数のチャネルに対する測定は、ワنسキヤン・連続スキャン及び分割モードで行うことができます。変換結果は、16bit データ・レジスタの上位もしくは下位に格納されます。

アナログ・ウォッチドッグ機能では、ユーザが指定した上限もしくは下限から、入力が逸脱していないかを検出することができます。

10.2 ADC の主な機能

- 精度 12bit
- 変換終了時、インジェクト変換終了時、及びアナログ・ウォッチドッグ・イベント終了時に割込みを生成
- ワنسキヤン、及び連続変換モード
- チャネル 0 から チャネル n を自動変換するためのスキャン・モード
- 自己較正
- 変換データの性格に応じたビット寄せの選択
- チャネルごとのサンプリングタイムの設定
- レギュラ及びインジェクト変換の両方についての外部トリガ・オプション
- 分割モード
- デュアル・モード（2 個の ADC の協調動作）
- ADC 変換レート 1 MHz
 - STM32F103xx パフォーマンス・ライン・デバイス : 56MHz 時 $1\ \mu s$ (72MHz 時 $1.17\ \mu s$)
 - STM32F101xx アクセス・ライン・デバイス : 28MHz 時 $1\ \mu s$ (36MHz 時 $1.55\ \mu s$)
- ADC 電源供給 : 2.4V ~ 3.6V
- ADC 入力電圧 : $V_{REF-} \leq V_{IN} \leq V_{REF+}$
- レギュラ変換における DMA リクエストの生成

ADC 回路のブロック構成は [図 21](#) に示されています。

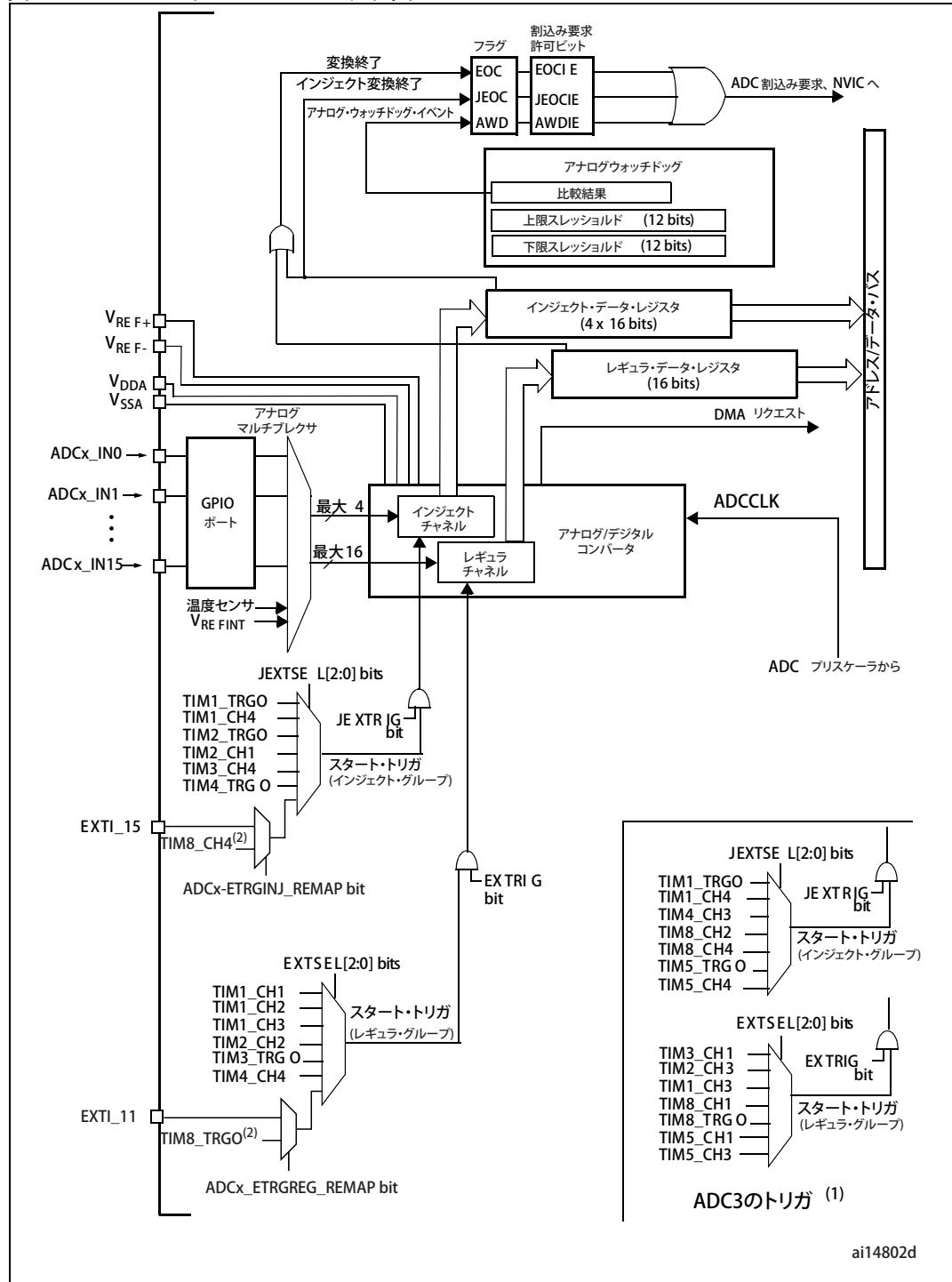
注意 : V_{REF-} ピンが用意されているパッケージでは、このピンを V_{SSA} に接続してください。

参考資料

10.3 ADC の機能詳細

図 21 は、1 回路分の ADC ブロック図を、表 41 は ADC ピン配列を示しています。

図 21. シングル ADC ブロック図



1. ADC3 は、ADC1 と ADC2 とは異なるレギュラとインジェクト変換トリガを持っています。
2. TIM8_CH4 と TIM8_TRGO に対応したビット寄せは、大容量デバイスに限り存在します。

表 41. ADC ピン

名称	信号種別	摘要
V_{REF+}	入力、 アナログ参照電圧 +	ADC の参照電圧、ハイレベル / 正電圧 $2.4V \leq V_{REF+} \leq V_{DDA}$
V_{DDA}	入力、 アナログ電源供給	アナログ回路電源、 $2.4V \leq V_{DDA} \leq V_{DD}$ (3.6V)
V_{REF-}	入力、 アナログ参照電圧 -	ADC の参照電圧、ローレベル / 負電圧 $V_{REF-} = V_{SSA}$
V_{SSA}	入力、 アナロググランド	V_{SS} と等しいアナログ回路電源のグランド
ADC _x _IN[15:0]	アナログ入力信号	16 チャネルのアナログ入力

10.3.1 ADC 電源の制御

ADC 回路は ADC_CR1 レジスタの ADON ビットをセットすることで、電源オンの状態となります。ADON ビットが最初にセットされたときには、ADC はパワー・ダウン・モードからの立上げとなります。

二度目に ADON ビットがソフトウェアでセットされると、ADC パワー・アップ・タイム (t_{STAB}) 後に変換動作が開始されます。

変換動作を停止し ADC をパワー・ダウン・モードとするには、ADON ビットをクリアします。この状態では ADC 回路は電源をほとんど消費しません。(数 μA 程度)

10.3.2 ADC クロック

PCLK2 (APB2 クロック) に同期した ADCCLK クロックはクロック・コントローラから供給されます。RCC コントローラは、ADC クロック専用のプリスケーラを持っています。詳細は、[リセットとクロックの制御 \(RCC\) \(72 ページ\)](#) クロック (CLK) の章を参照してください。

10.3.3 チャネル選択

ADC は 16 の多重化された入力チャネルを搭載しています。これらの入力の変換動作は、レギュラ (regular) とインジェクト (injected) の 2 グループに構成することができます。各グループは変換チャネルの順序を定めるもので、任意のチャネルを任意の順序で指定することができます。たとえば、Ch3、Ch8、Ch2、Ch2、Ch0、Ch2、Ch2、Ch15 のようなチャネルの選び方が可能です。

- レギュラ・グループ (regular group) には、最大 16 個の変換動作を指定することができます。レギュラ・グループのチャネルと順序は ADC_SQRx レジスタで指定し、グループに含まれる変換の数は ADC_SQR1 レジスタの L[3:0] フィールドで与えます。
- インジェクト・グループ (injected group) には、最大 4 個の変換動作を指定することができます。インジェクト・グループのチャネルと順序は ADC_JSQR レジスタで指定し、グループに含まれる変換の数は同じ ADC_JSQR レジスタの JL[1:0] フィールドで与えます。

変換動作中に ADC_SQRx もしくは ADC_JSQR レジスタが変更された時には、その時点で行われている変換はリセットされ、新たに設定されたグループの変換を開始するスタートパルスが ADC に送られます。

温度センサ / V_{REFINT} 内部チャネル

温度センサは ADC_x_IN16 へ、内部の参照電圧 V_{REFINT} は ADC_x_IN17 へ、チップ内部で ADC に接続されています。この両内部チャネルは、レギュラ・グループ、インジェクト・グループのいずれにおいても、一般的な入力チャネルと同様に選択することができます。

注意： 温度センサと V_{REFINT} はマスタ ADC1 のみで利用できます。

10.3.4 シングル変換モード (Single conversion mode)

シングル変換モードでは ADC は一回の変換を行います。このモードは CONT ビットが 0 のとき、ADC_CR2 レジスタの ADON ビットをセットする（レギュラ・チャネルのみ）か、もしくは外部トリガによって（レギュラもしくはインジェクト・チャネルの場合）変換が行われます。

指定されたチャネルの変換が終了すると、

- レギュラ・チャネルが変換した場合：
 - 変換されたデータは 16bit 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れば、割込みリクエストを発生します。
- インジェクト・チャネルが変換した場合：
 - 変換されたデータは 16bit の ADC_DRJ1 レジスタに格納されます。
 - JEOC (End Of Conversion Injected) フラグがセットされます。
 - JEDECIE ビットがセットされていれば、割込みリクエストが発生します。

この後、ADC の変換動作は停止します。

10.3.5 連続変換モード (Continuous conversion mode)

連続変換モードでは、ADC は一回の変換が終わると次の変換に移ります。このモードは CONT ビットが 1 のとき、外部トリガか、もしくは ADC_CR2 レジスタの ADON ビットをセットすることでスタートします。

グループの各入力チャネルの変換の後に、

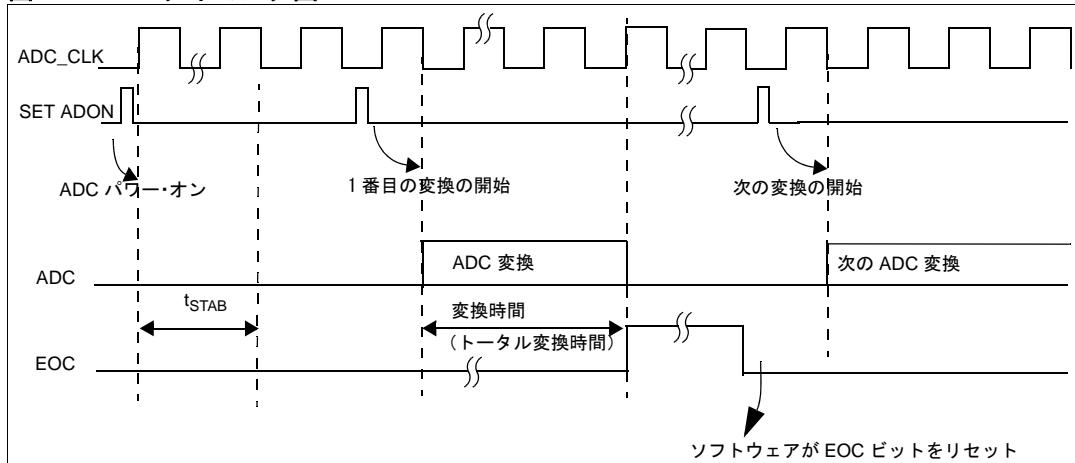
- レギュラ・チャネルが変換した場合：
 - 変換されたデータは 16bit の ADC_DR レジスタに格納されます。
 - EOC (End Of Conversion) フラグがセットされます。
 - EOCIE ビットがセットされていれば、割込みリクエストを発生します。
- インジェクト・チャネルが変換した場合：
 - 変換されたデータは 16bit の ADC_DRJ1 レジスタに格納されます。
 - JEOC (End Of Conversion Injected) フラグがセットされます。
 - JEDECIE ビットがセットされていれば、割込みリクエストが発生します。

参考資料

10.3.6 タイミング図

図 22 に見られるように、ADC は高精度の変換動作を開始する前に安定時間 t_{STAB} を必要とします。ADC 変換の開始後 14 クロック・サイクルで EOC フラグがセットされ、変換されたデータは 16bit の ADC データ・レジスタに格納されます。

図 22. タイミング図



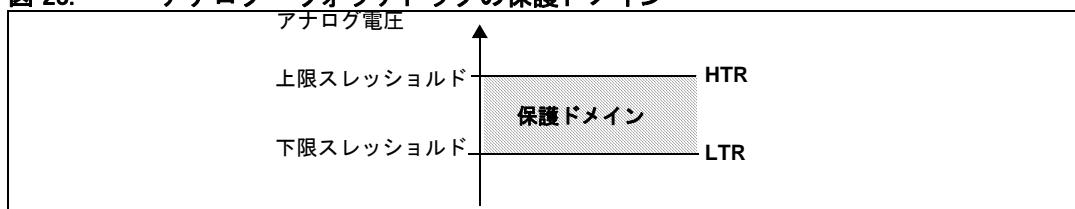
10.3.7 アナログ・ウォッチドッグ

アナログ・ウォッチドッグ (AWD) ステータス・ビットは、ADC で変換された値が下限スレッショルドを下回るか、もしくは上限スレッショルドを超えた時にセットされます。この上下限の値は、16bit レジスタ ADC_HTR 及び ADC_LTR レジスタの下位 12bit に設定します。ADC_CR1 レジスタの AWDIE ビットの設定で、割込みリクエストを発生させることができます。

上下限の値は ADC_CR2 レジスタの ALIGN ビットの設定の影響を受けません。上下限との比較はデータを 16bit 化する前に行われます。(セクション 10.5 を参照)

アナログ・ウォッチドッグ機能は表 42 のように、ADC_CR1 レジスタの設定に応じて、一変換もしくは一連の変換をチェック対象とすることができます。

図 23. アナログ・ウォッチドッグの保護ドメイン



参考資料

表 42. アナログ・ウォッチドッグ・チャネル選択

アナログ・ウォッチドッグで監視されるチャネル	ADC_CR1 レジスタ制御ビット (x = 無視)		
	AWDSDL 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・チャネル	0	0	1
すべてのレギュラ・チャネル	0	1	0
すべてのレギュラとインジェクト・チャネル	0	1	1
シングル ⁽¹⁾ インジェクト・チャネル	1	0	1
シングル ⁽¹⁾ レギュラ・チャネル	1	1	0
シングル ⁽¹⁾ レギュラまたはインジェクト・チャネル	1	1	1

1. AWDCH[4:0] フィールドで指定されます。

10.3.8 スキャン・モード (Scan mode)

このモードはアナログ入力のグループをスキャンする際に使用されます。

スキャン・モードは ADC_CR1 レジスタの SCAN ビットをセットしたときに選択されます。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・チャネル時）もしくは ADC_JSQR レジスタ（インジェクト・チャネル時）で選択されているすべてのチャネルをスキャンします。グループ内の各チャネルに対しては、1 回の変換が行われます。各変換が終了すると、グループ内の次の変換が自動的に行われます。CONT ビットがセットされていれば、グループの最後のチャネルのあとも変換動作は終了せず、そのグループの最初の項目から変換が続けられます。

DMA ビットがセットされていれば、各 EOC 時に、レギュラ・グループの変換データを SRAM に転送するために DMA コントローラを利用することができます。

インジェクト・チャネルの変換データは、常に ADC_JDRx レジスタに格納されます。

10.3.9 インジェクト・チャネルの管理

トリガ・インジェクション (Triggered injection)

トリガ・インジェクションを行うには、ADC_CR1 レジスタで JAUTO ビットがクリアされており、SCAN ビットがセットされていなければなりません。

- 外部トリガか、ADC_CR2 レジスタの ADON ビットをセットすることにより、レギュラ・グループのチャネルの変換を開始します。
- レギュラ・グループの変換中に、外部インジェクト・トリガが発生すると、その時点で行われている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・ワーム・モードで変換されます。
- その後、レギュラ・グループ・チャネル変換は、最後の割込みレギュラ変換から再開されます。レギュラ・イベントはインジェクト変換の間に発生し、割込みはありませんが、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 24](#) は、タイミング図を示しています。

参考資料

注意: トリガ・インジェクションを使用する場合は、トリガ・イベントの間隔はインジェクション・シーケンスより長くなければなりません。例えば、シーケンスの長さが 28 ADC クロック・サイクル（それぞれの 1.5 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、最小のトリガ間隔は 29ADC クロック・サイクルにする必要があります。

自動インジェクション (Auto injection)

JAUTO ビットがセットされているとき、インジェクト・グループ・チャネルはレギュラ・グループ・チャネルの後に自動的に変換されます。これにより、ADC_SQRx レジスタと ADC_JSQR レジスタを使用して最大 20 個の変換順序を設定できます。

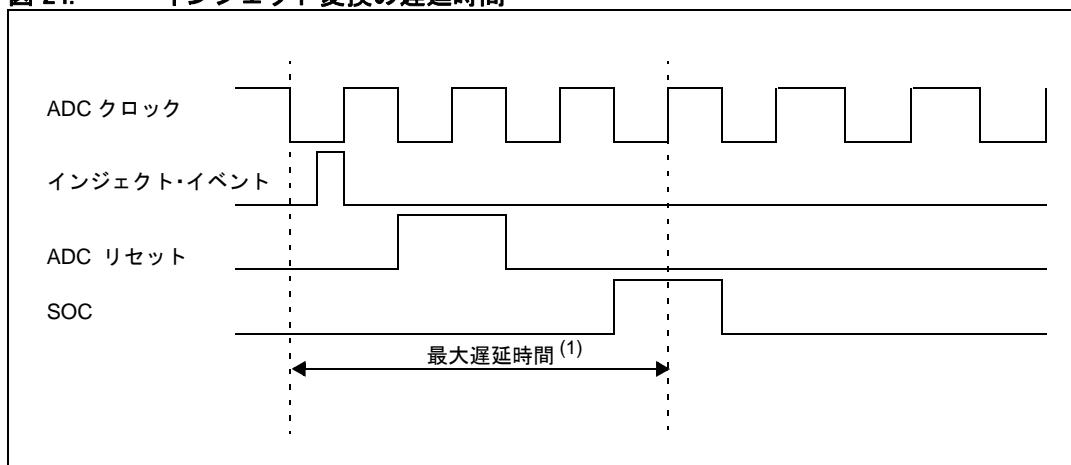
このモードではインジェクト・チャネルの外部トリガは無効にする必要があります。

JAUTO ビットとともに CONT ビットがセットされている場合は、レギュラ・チャネルの変換とインジェクト・チャネルの変換が交互に行われます。

範囲 4 から 8 までの ADC クロック・プリスケーラ用には、レギュラからインジェクト・シーケンス（同様にインジェクトからレギュラ）への切り替え時に、1ADC クロック周期の遅延時間が自動的に挿入されます。ADC クロック・プリスケーラが 2 に設定された場合、遅延時間は 2ADC クロック周期です。

注意: 自動インジェクト・モードと分割モードを同時に有効とすることはできません。

図 24. インジェクト変換の遅延時間



1. 最大遅延時間は、STM32F101 及び STM32F103xx の電気特性の資料に記されています。

10.3.10 分割モード (Discontinuous mode)

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で有効となります。このモードでは、ADC_SQRx レジスタで指定されている変換対象のうちの n ($n \leq 8$) 項目を一連の変換の対象とします。変換数 n は ADC_CR1 レジスタの DISCNUM[2:0] フィールドで指定します。

外部トリガが生じると、ADC_SQRx レジスタで選択されている項目がすべて終了するまで、この内の次の n 項目の変換が開始されます。ADC_SQRx の有効な設定の総数は ADC_SQR1 レジスタの L[3:0] で指定されています。

参考資料

RM0008

A/D コンバータ (ADC)

例：

- n = 3、変換対象となるチャネル = 0、1、2、3、6、7、9、10
- 最初のトリガによる変換の対象は、0、1、2
- 2番目のトリガによる変換の対象は、3、6、7
- 3番目のトリガによる変換の対象は、9、10、次に EOC イベントが生じます。
- 4番目のトリガによる変換の対象は、0、1、2

注意： レギュラ変換グループが分割モードで変換されるときには、指定項目の最後から直ちに最初の項目に戻ることはできません。

グループのすべての分割スキャンが終了すると、次のトリガではグループの最初からスキャンが行われます。上の例では、4番目のトリガで最初のサブグループ 0、1、2 が変換されます。

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となります。このモードでは、ADC_JSQR レジスタで指定されている変換対象を、外部トリガごとに 1 チャネルずつ変換します。

外部トリガが生じると、ADC_JSQR レジスタで選択されている項目がすべて終了するまで、この内の次の変換が開始されます。ADC_JSQR の有効な設定の総数は ADC_JSQR レジスタの JL[1:0] で指定されています。

例：

- n = 1、変換対象となるチャネル = 1、2、3
- 最初のトリガではチャネル 1 が変換され、
- 2番目のトリガではチャネル 2 が変換され、
- 3番目のトリガではチャネル 3 が変換され、EOC 及び JEOC イベントが生じます。
- 4番目のトリガではチャネル 1 が変換されます。

注意： 1 すべてのインジェクト・チャネルが変換されると、次のトリガではグループの最初から変換が行われます。上の例では、4番目のトリガで最初の項目、チャネル 1 が変換されます。
2 自動インジェクト・モードと分割モードを同時に有効とすることはできません。
3 レギュラとインジェクト・グループの両方で分割モードを設定してはいけません。このモードの設定は、いずれか一方のグループのみに限られます。

10.4 較正機能

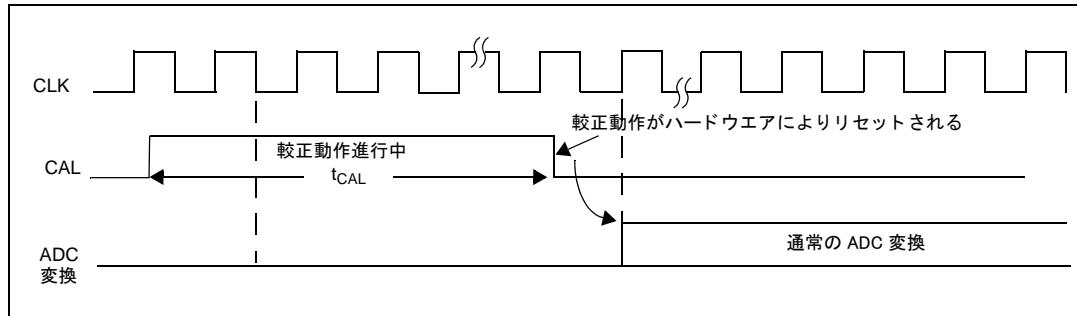
ADC には、自己較正機能が備えられています。この較正機能で、内部の複数のキャパシタのばらつきによる誤差を大幅に低減することができます。較正時に誤差の補正值（デジタル値）が個々のキャパシタに対して計算され、変換動作時にこの補正值を利用することでキャパシタによる誤差を取り除きます。

較正は ADC_CR2 レジスタの CAL ビットをセットすることで開始されます。較正が終了すると、CAL ビットはハードウェアでクリアされ、通常の変換動作が行われます。パワー・オン後に一度、ADC の較正を行うことが推奨されます。較正動作が終了すると、補正值が ADC_DR レジスタに格納されます。

注意： 1 パワー・アップ後には、毎回、較正を実行することをお勧めします。
2 較正動作を開始する前に、ADC は少なくとも 2 ADC クロック・サイクルの間、パワー・オフの状態 (ADON ビット = 0) になければなりません。



図 25. 較正動作のタイミング図



10.5 データの配置

ADC_CR2 レジスタの ALIGN ビットは、図 26 及び図 27 のように、変換データをデータ・レジスタに格納する際に左寄せするか、もしくは右寄せするかを指定します。

インジェクト・グループ・チャネルのデータの場合には、ADC_JOFRx レジスタにソフトウェアで任意に設定した値を引き算することができ、負の値を含む変換結果とすることができます。SEXT は符号が拡張されたビットです。

レギュラ・グループ・チャネルの変換結果ではオフセットの引き算は行われませんので、12bit のみが有効です。

図 26. データの右寄せ

インジェクト・グループ															
SEXT	SEXT	SEXT	SEXT	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
レギュラ・グループ															
0	0	0	0	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

図 27. データの左寄せ

インジェクト・グループ															
SEXT	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	0	0
レギュラ・グループ															
D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	0	0	0

10.6 入力チャネルごとのサンプリングタイムの設定

ADC は入力電圧を読込む際に、サンプル・タイムとして、ADCCLK サイクルを単位として ADC_SMPR1 及び ADC_SMPR2 の SMP[2:0] で指定されたサンプリングタイムを使用します。個々の入力チャネルには、個々に異なったサンプリングタイムを設定しておくことができます。

チャネルの変換に要する時間は次の式で計算されます。

$$T_{conv} = \text{サンプリングタイム} + 12.5 \text{ サイクル}$$

例：

ADCCLK = 14 MHz で、サンプリングタイムが 1.5 サイクルのとき、

$$T_{conv} = 1.5 + 12.5 = 14 \text{ サイクル} = 1 \mu\text{s}$$

10.7 外部トリガによる変換

タイマキャプチャ、外部割込みなどの外部イベントを変換のトリガとすることができます。EXTTRIG 制御ビットがセットされていれば、外部イベントは変換動作をトリガすることができます。EXTSEL[2:0] 及び JEXTSEL[2:0] フィールドはレギュラ及びインジェクト・グループのそれについて、8 種のイベントからトリガとして使用されるイベントを選択します。

注意： ADC のレギュラ及びインジェクト変換として外部トリガが選択されているとき、トリガ信号の立上がりリエッジのみが変換動作のトリガとなります。

表 43. ADC1 と ADC2 のレギュラ・チャネルのための外部トリガ

トリガ・ソース	信号種別	EXTSEL[2:0]
タイマ 1 CC1 出力	オンチップ・タイマからの信号	000
タイマ 1 CC2 出力		001
タイマ 1 CC3 出力		010
タイマ 2 CC2 出力		011
タイマ 3 TRGO 出力		100
タイマ 4 CC4 出力		101
外部割込み 11 /TIM8_TRGO イベント ⁽¹⁾⁽²⁾	外部ピン / オンチップ・タイマからの内部信号	110
SWSTART	ソフトウェア制御ビット	111

1. TIM8_TRGO イベントは、大容量デバイスにのみ搭載されています。

2. 外部トリガ EXTI_line11 やレギュラ・チャネルのための TIM8_TRGO イベントの選択は、ADC1 と ADC2 のそれぞれについて、構成ビット ADC1_ETRGREG_REMAP 及び ADC2_ETRGREG_REMAP により行われます。

参考資料

表 44. ADC1 と ADC2 のインジェクト・チャネルのための外部トリガ

トリガ・ソース	信号種別	JEXTSEL[2:0]
タイマ 1 TRGO 出力	オンチップ・タイマからの信号	000
タイマ 1 CC4 出力		001
タイマ 2 TRGO 出力		010
タイマ 2 CC1 出力		011
タイマ 3 CC4 出力		100
タイマ 4 TRGO 出力		101
外部割込み 15 /TIM8_CC4 イベント ⁽¹⁾⁽²⁾	外部ピン / オンチップ・タイマからの内部信号	110
JSWSTART	ソフトウェア制御ビット	111

1. TIM8_CC4 イベントは、大容量デバイスにのみ搭載されています。
2. 外部トリガ EXTI_line15 やインジェクト・チャネルの TIM8_CC4 イベントは、ADC1 と ADC2 のそれぞれについて、構成ビット ADC1_ETRGINJ_REMAP 及び ADC2_ETRGINJ_REMAP により行われます。

表 45. ADC3 のレギュラ・チャネルのための外部トリガ

トリガ・ソース	信号種別	EXTSEL[2:0]
タイマ 3 CC1 イベント	オンチップ・タイマからの内部信号	000
タイマ 2 CC3 イベント		001
タイマ 1 CC3 イベント		010
タイマ 8 CC1 イベント		011
タイマ 8 TRGO イベント		100
タイマ 5 CC1 イベント		101
タイマ 5 CC3 イベント		110
SWSTART	ソフトウェア制御ビット	111

表 46. ADC3 のインジェクト・チャネルのための外部トリガ

トリガ・ソース	信号種別	JEXTSEL[2:0]
タイマ 1 TRGO イベント	オンチップ・タイマからの内部信号	000
タイマ 1 CC4 イベント		001
タイマ 4 CC3 イベント		010
タイマ 8 CC2 イベント		011
タイマ 8 CC4 イベント		100
タイマ 5 TRGO イベント		101
タイマ 5 CC4 イベント		110
JSWSTART	ソフトウェア制御ビット	111

ソフトウェア・トリガ・イベントは、レジスタのビット (SWSTART 及び ADC_CR2 の JSWSTART) を設定することで行えます。

レギュラ・グループ変換は、インジェクト・トリガにより割込まれることが可能です。

10.8 DMA リクエスト

レギュラ・チャネルの変換では変換結果は 1 個のレジスタに格納されます。このため、複数のチャネルを変換する際にすでに ADC_DR レジスタに格納されているデータが失われるのを防ぐため、DMA の使用が必要とされる場合があります。

レギュラ・チャネルの変換終了時にのみ DMA リクエストが発生し、これにより ADC_DR レジスタからソフトウェアで指定した場所に変換データを移動することができます。

注意: この DMA 機能は、ADC1 と ADC3 だけが持っています。ADC2 により変換されたデータは、マスタである ADC1 によって、DMA を使いデュアル ADC モードで転送可能です。

10.9 デュアル ADC モード

ADC を 2 回路搭載したデバイスでは、デュアル ADC モードを使用することができます。
([図 28](#) を参照)

デュアル ADC モードでは、ADC1 (マスタ) と ADC2 (スレーブ) に、交互に、もしくは同時に変換開始トリガが与えられます。これは ADC_CR1 レジスタの DUALMOD[2:0] ビットで選択されます。

注意: デュアル ADC モードでは、外部イベントによる変換動作の設定はマスタ ADC のみに行います。スレーブでソフトウェア・トリガをセットすると、スレーブ ADC で誤った変換動作が行われてしまいます。但し、外部トリガを有効とする設定は、マスタとスレーブの両方の ADC に対して行われねばなりません。

デュアル ADC には、次の 6 種の動作モードがあります。

- 同時インジェクト・モード (Injected simultaneous mode)
- 同時レギュラ・モード (Regular simultaneous mode)
- 高速インターブ・モード (Fast interleaved mode)
- 低速インターブ・モード (Slow interleaved mode)
- オルタネート・トリガ・モード (Alternate trigger mode)
- 独立モード (Independent mode)

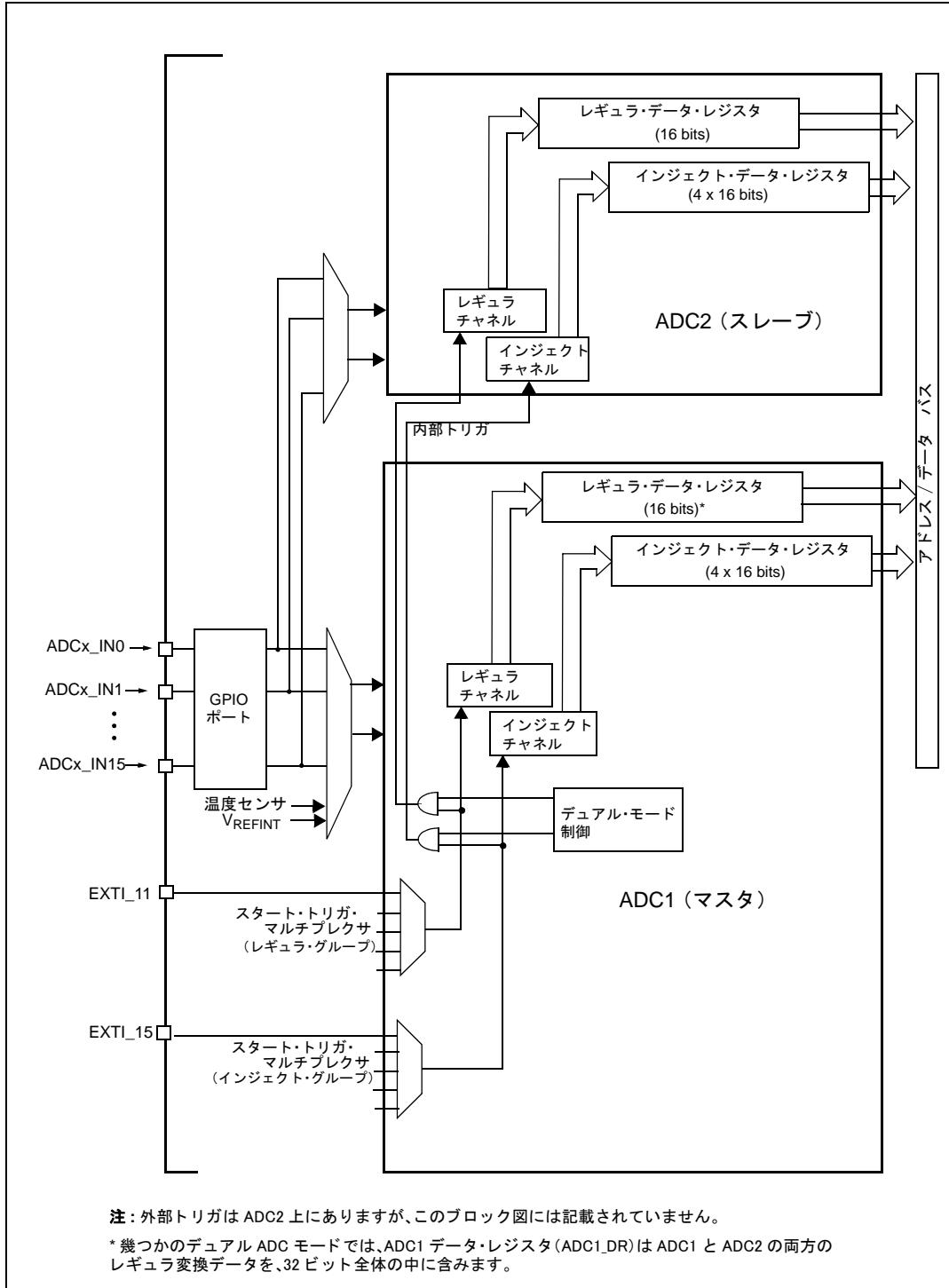
これらのモードが次のように組合わされる場合があります。

- 同時インジェクト・モード + 同時レギュラ・モード
- 同時レギュラ・モード + オルタネート・トリガ・モード
- 同時インジェクト・モード + インターブ・モード

注意: デュアル ADC モードでは、マスタのデータ・レジスタからスレーブの変換データを読出すには、変換されたレギュラ・チャネル・データの転送に使用されるか否かにかかわらず、DMA ビットがセットされていなければなりません。

参考資料

図 28. デュアル ADC ブロック図



参考資料

10.9.1 同時インジェクト・モード (Injected simultaneous mode)

このモードではインジェクト・チャネル・グループの変換を行います。外部トリガ信号は (ADC1_CR2 レジスタの JEXTSEL[2:0] によって選択された) ADC1 のインジェクト・グループ・マルチプレクサから来ます。同時トリガは ADC2 へ送られます。

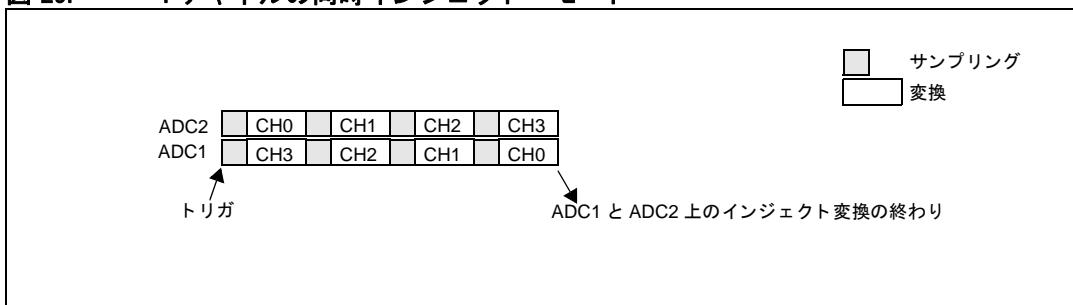
注意: 同じ入力チャネルを両方の ADC で変換してはいけません。(同じチャネルを両方の ADC で変換する時には、サンプリングタイムが重なってはいけません。)

ADC1 もしくは ADC2 における変換終了イベント時に、

- 変換されたデータは、各 ADC の ADC_JDRx レジスタに格納されています。
- ADC1/ADC2 のインジェクト・チャネルがすべて変換されたとき、いずれかの ADC で割込みリクエストが許可されていれば、JEOC 割込みリクエストが発生します。

注意: 同時モードでは、変換グループの長さが同一で、両方の変換グループで必要なトリガ間隔より、実際のトリガ間隔が長いことが確実でなければなりません。変換グループの長さが異なるときには、長いほうの ADC で変換が終了するまでに、短い方の ADC で設定の最初に戻っての変換が行われることがあります。

図 29. 4 チャネルの同時インジェクト・モード



10.9.2 同時レギュラ・モード (Regular simultaneous mode)

このモードではレギュラ・チャネル・グループの AD 変換を行います。外部トリガ信号は ADC1 のレギュラ変換トリガ・スイッチで、ADC1_CR2 レジスタの EXTSEL[2:0] によって選択されます。トリガ信号は ADC2 に同時に送られます。

注意: 同じ入力チャネルを両方の ADC で変換してはいけません。(同じチャネルを両方の ADC で変換する時には、サンプリングタイムが重なってはいけません。)

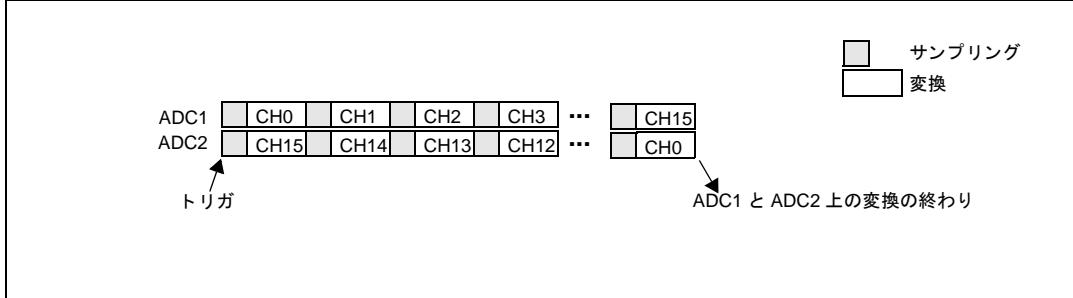
ADC1 もしくは ADC2 における変換終了イベント時に、

- DMA ビットがセットされているときには、32bit の ADC1_DR レジスタの内容を SRAM に転送するため、DMA 転送リクエストが発生します。ADC1_DR レジスタには、上位 16bit に ADC2 の変換データが、下位 16bit に ADC1 の変換データが格納されています。
- ADC1/ADC2 のレギュラ変換チャネルがすべて変換されたとき、いずれかの ADC で割込みリクエストが許可されていれば、EOC 割込みリクエストが発生します。

注意: 同時モードでは、変換グループの長さが同一で、両方の変換グループで必要なトリガ間隔より、実際のトリガ間隔が長いことが確実でなければなりません。変換グループの長さが異なるときには、長いほうの ADC で変換が終了するまでに、短い方の ADC で設定の最初に戻っての変換が行われることがあります。

参考資料

図 30. 16 チャネルの同時レギュラ変換モード



10.9.3 高速インタリーブ・モード (Fast interleaved mode)

このモードはレギュラ変換グループ（通常は一入力チャネル）でのみ実行されます。外部トリガ信号はADC1のレギュラ変換トリガから来ます。外部トリガ時には、

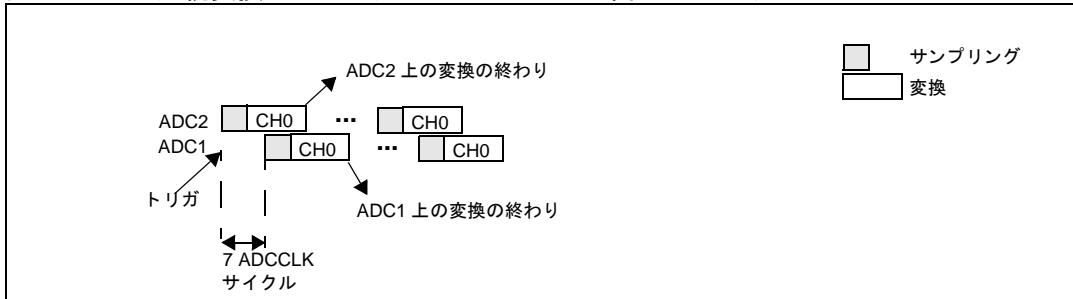
- ADC2は直ちにスタートします。
- ADC1は7 ADC クロック・サイクル遅れて変換がスタートします。

ADC1とADC2の両方でCONTビットがセットされていれば、それぞれのレギュラ・グループは連続的に変換されます。

ADC1でEOCIEビットがセットされておりEOC割込みリクエストが発生した後、DMAビットがセットされているときには、32bitのADC1_DRレジスタの内容をSRAMに転送するため DMA 転送リクエストが発生します。ADC1_DRレジスタには、上位16bitにADC2の変換データが、下位16bitにADC1の変換データが格納されています。

注意：同じ入力チャネルを変換する場合、ADC1とADC2のサンプリングタイムがオーバラップすることを防ぐため、使用可能な最大サンプリングタイムは7ADCCLKサイクル未満です。

図 31. 連続変換モードにおける1チャネルの高速インタリーブ・モード



10.9.4 低速インタリーブ・モード (Slow interleaved mode)

このモードはレギュラ変換グループ（一入力チャネルのみ）で実行されます。外部トリガ信号はADC1のレギュラ変換トリガから来ます。外部トリガ時には、

- ADC2は直ちにスタートします。
- ADC1は14 ADC クロック・サイクル遅れて変換がスタートします。
- ADC2は、更に14 クロック遅れて、二番目の変換を開始します。以降、この要領で変換が続きます。

注意：ADC1とADC2のサンプリングタイムがオーバラップすることを防ぐため、使用可能な最大サンプリングタイムは14ADCCLKサイクル未満です。

参考資料

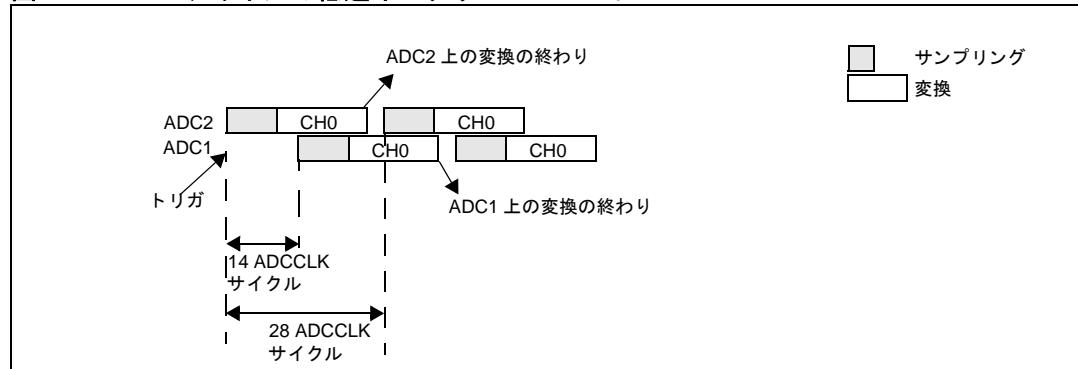
ADC1 で EOCIE ビットがセットされており EOC 割込みリクエストが発生した後、DMA ビットがセットされているときには、32bit の ADC1_DR レジスタの内容を SRAM に転送するため DMA 転送リクエストが発生します。ADC1_DR レジスタには、上位 16bit に ADC2 の変換データが、下位 16bit に ADC1 の変換データが格納されています。

28 ADC クロック・サイクル後には、ADC2 は自動的に新たな変換を開始します。

このモードでは指定されたレギュラ変換チャネルを連続的に変換するため、CONT ビットをセットすることはできません。

注意： アプリケーションでは、インタリーブ・モードがイネーブル状態のとき、インジェクト・チャネルのトリガが発生しないよう、注意することが必要とされます。

図 32. 1 チャネルの低速インタリーブ・モード



10.9.5 オルタネート・トリガ・モード (Alternate trigger mode)

このモードはインジェクト・チャネル・グループでのみ実行されます。外部トリガ信号は ADC1 のインジェクト・チャネル・マルチプレクサから供給されます。

- 最初のトリガが発生したとき、ADC1 すべてのインジェクト・グループ・チャネルの変換が開始されます。
- 二番目のトリガ発生で、ADC2 のすべてのインジェクト・グループ・チャネルの変換が開始されます。
- 以上の動作が繰返されます。

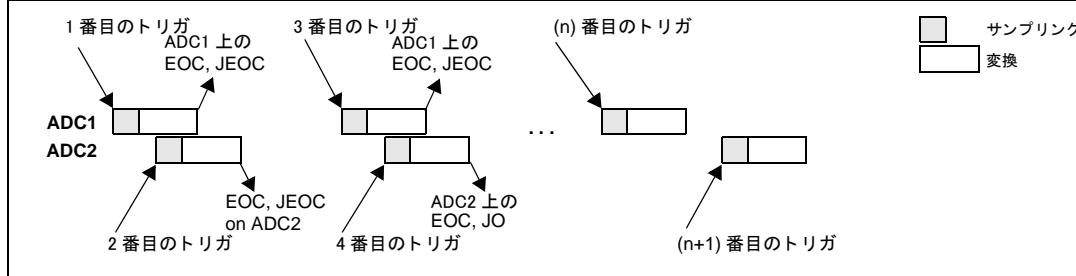
ADC1 すべてのインジェクト・グループ・チャネルの変換が完了すると、(割込みリクエストが許可されているとき) JEOC 割込みリクエストが発生します。

ADC2 すべてのインジェクト・グループ・チャネルの変換が完了すると、(割込みリクエストが許可されているとき) JEOC 割込みリクエストが発生します。

すべてのインジェクト・グループ・チャネルの変換された後に他の外部トリガが発生する場合、オルタネート・トリガ・プロセスは ADC1 インジェクト・グループ・チャネルの変換により再スタートします。

参考資料

図 33. オルタネート・トリガ：各 ADC のインジェクト・チャネル・グループ



ADC1、ADC2 の両方で、インジェクト分割モードが有効になっているときには、

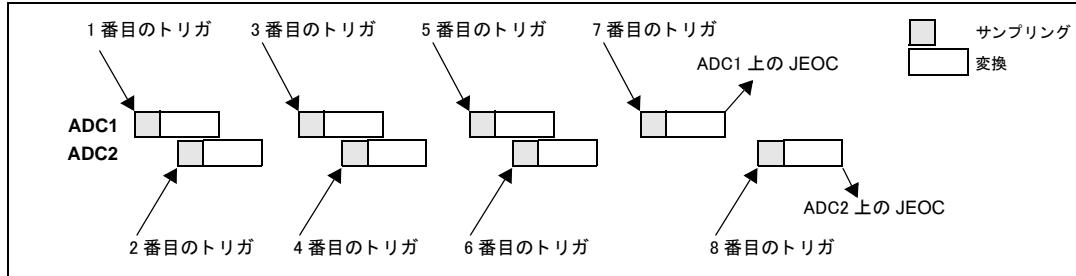
- 最初のトリガが発生したとき、ADC1 のインジェクト・チャネルが変換されます。
- 二番目のトリガが発生したとき、ADC2 のインジェクト・チャネルが変換されます。
- 以上の動作が繰返されます。

ADC1 すべてのインジェクト・グループ・チャネルの変換が完了すると、(割込みリクエストが許可されているとき) JEOC 割込みリクエストが発生します。

ADC2 すべてのインジェクト・グループ・チャネルの変換が完了すると、(割込みリクエストが許可されているとき) JEOC 割込みリクエストが発生します。

すべてのインジェクト・グループ・チャネルが変換された後に他の外部トリガが発生する場合、オルタネート・トリガ・プロセスは再スタートします。

図 34. オルタネート・トリガ：分割モデルの（各 ADC ごとの）4 インジェクト・チャネル



10.9.6 独立モード (Independent mode)

ADC 間の同期を行わず、各 ADC 回路を独立して動作します。

10.9.7 同時レギュラ・モード + 同時インジェクト・モード

レギュラ・グループの同時変換を中断中に、インジェクト・グループの同時変換を開始することができます。

注意： 同時レギュラ／インジェクト・モードの結合において、1つは、同じ長さでシーケンスを変換しなければなりません。もしくは、2シーケンス以上の長さのトリガ間隔が必要です。変換グループの長さが異なる場合には、シーケンスが長い方の ADC が以前の変換が完了するまでに、短いシーケンスの ADC が再スタートすることがあります。

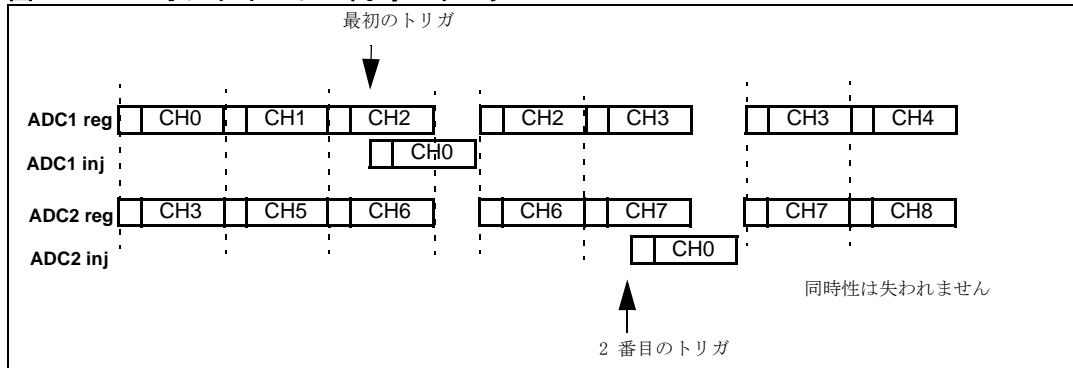
10.9.8 並列レギュラ・モード + オルタネート・トリガ・モード

レギュラ・グループの同時変換中断中に、インジェクト・グループのオルタネート・トリガ変換を開始することができます。図 35 は、オルタネート・トリガが同時レギュラ変換の割込み動作を示しています。

インジェクト・オルタネート変換は、インジェクト・イベントが発生すると直ちに開始されます。すでにレギュラ変換が動作している場合、インジェクト変換後に同時性を維持するため、両 ADC (マスター・スレーブ) におけるレギュラ変換は停止し、インジェクト変換の終了後に再開されます。

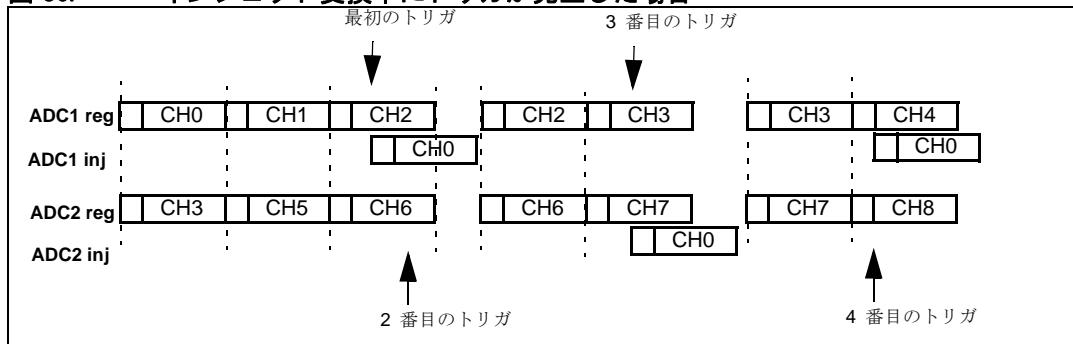
注意: 同時レギュラ+オルタネート・トリガ・モードの結合において、1つは、同じ長さでシーケンスを変換しなければなりません。もしくは、2シーケンス以上の長さのトリガ間隔が必要です。変換グループの長さが異なる場合には、シーケンスが長い方の ADC が以前の変換が完了するまでに、短いシーケンスの ADC が再スタートすることがあります。

図 35. オルタネート + 同時レギュラ



インジェクト変換がレギュラ変換へ割込みをしている間にトリガが発生する場合、このトリガは無視されます。図 36 はこの状況を図示しています。(2 番目のトリガが無視されています。)

図 36. インジェクト変換中にトリガが発生した場合



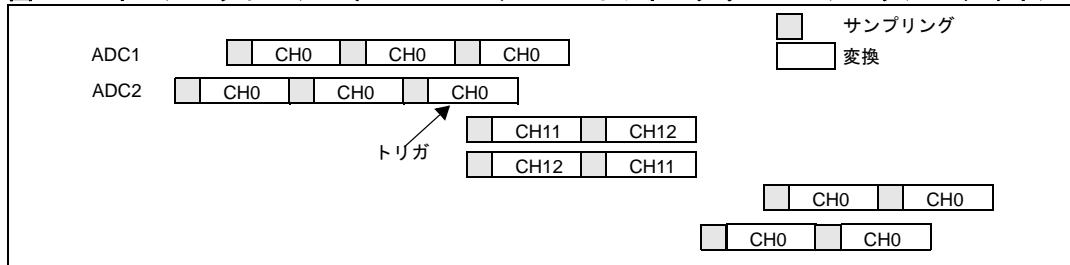
10.9.9 同時インジェクト + インタリープ

インタリープ・モードの変換を行っている際に、インジェクト・イベントで、これを中断することができます。この場合、インタリープ・モードの変換は中断され、インジェクト変換動作がスタートします。インジェクト変換の終了後は、インタリープ変換が再開されます。図 37 には、この使用例の動作が示されています。

参考資料

注意： ADC クロック・プリスケーラが 4 に設定されている場合、インターブ・モードは等間隔のサンプル期間ではリカバーしません：サンプリング間隔は、7 クロック周期に続く 7 クロック周期ではなく、6ADC クロック周期に続き 8ADC クロック周期となります。

図 37. インジェクト・シーケンス CH11、CH12 でのインターブ・シングル・チャネル



10.10 温度センサ

温度センサは、デバイスの周辺空気温度 (T_A) の測定を行うために利用することができます。

温度センサは内部で $ADCx_IN16$ 入力チャネルに接続されており、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。この入力のサンプリングタイムの推奨値は、 $17.1 \mu s$ です。

温度センサのブロック図は図 38 にあります。

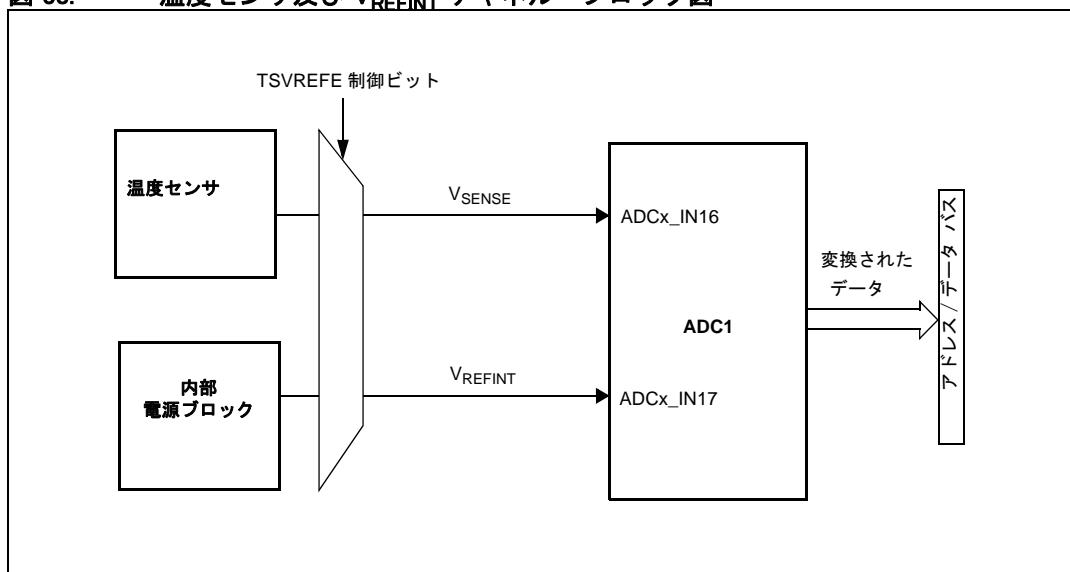
センサが使用されないときには、この回路をパワー・ダウン状態としておくことができます。

注意： 内部チャネル $ADCx_IN16$ (温度センサ) と $ADCx_IN17 (V_{REFINT})$ の変換を可能とするには、 $TSVREFE$ ビットをセットしておかねばなりません。

主な機能

- 温度範囲 : $-40 \sim 125^\circ C$
- 精度 : $\pm 1.5^\circ C$

図 38. 温度センサ及び V_{REFINT} チャネル・ブロック図



温度データの読み込み

温度センサを使用するには、

1. ADC_x_IN16 入力チャネルを選択します。
2. サンプリングタイムを $17.1 \mu\text{s}$ に設定します。
3. 温度センサをパワー・ダウン・モードから起動するため、[ADC 制御レジスタ 2 \(ADC_CR2\)](#) レジスタの TSVREFE ビットをセットします。
4. V_{SENSE} データの変換結果を ADC データ・レジスタから読出します。
5. 次の式を適用し、温度の値を得ます。

$$\text{温度 } (\text{ }^{\circ}\text{C}) = \{(V_{25} - V_{\text{SENSE}}) / \text{Avg_Slope}\} + 25$$

ここで、

$$V_{25} = 25 \text{ }^{\circ}\text{C} \text{における } V_{\text{SENSE}} \text{ の値}$$

Avg_Slope = 温度・ V_{SENSE} (mV/ $^{\circ}\text{C}$ もしくは $\mu\text{V}/ ^{\circ}\text{C}$) カーブの平均傾斜

6. V_{25} 及び Avg_Slope の実際の値に関しては、データシートのデバイスの電気特性を参照してください。

注意：センサがパワー・ダウン状態から起動し、その時点の有効な V_{SENSE} を出力するには、スタートアップタイムが必要とされます。ADC 自体もパワー・ダウン・モードからのスタートアップタイムを必要としていますので、これらの遅延時間を最小とするため、ADON ビットと TSVREFE ビットは同時のセットされることが効果的です。

10.11 ADC 割込みリクエスト

レギュラ・グループとインジェクト・グループの変換終了時、及びアナログ・ウォッチドッグ・ステータス・ビットがセットされたとき、割込みリクエストを発生させることができます。これらには、個別に割込みリクエスト許可ビットが備えられています。

注意：ADC1 と ADC2 割込みは、同じ割込みベクタへ配置されます。ADC3 割込みは別の割込みベクタへ配置されます。

ADC_SR レジスタには他に 2 個のフラグが備わっていますが、これらは割込みリクエストにはかかわりません。

- JSTRT (インジェクト・グループ・チャネルの変換開始)
- STRT (レギュラ・グループ・チャネルの変換開始)

表 47. ADC 割込み

割込みイベント	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

参考資料

10.12 ADC レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#) を参照してください。

10.12.1 ADC ステータス・レジスタ (ADC_SR)

アドレスオフセット : 00h

リセット値 : 00000000h

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								STRT	JSTRT	JEOC	EOC	rc_w0	rc_w0	rc_w0	rc_w0

Bits 31:5 予約済み、書き込み時は常に 0 としてください。

Bit 4 **STRT**: レギュラ変換チャネル・スタート・フラグ

このビットはレギュラ・チャネルの変換が開始されたとき、ハードウェアによってセットされます。フラグのクリアはソフトウェアで行います。

0: レギュラ・チャネルの変換は開始されていません。

1: レギュラ・チャネルの変換が開始されました。

Bit 3 **JSTRT**: インジェクト・チャネル・スタート・フラグ

このビットはインジェクト・チャネルの変換が開始されたとき、ハードウェアによってセットされます。フラグのクリアはソフトウェアで行います。

0: インジェクト・グループの変換は開始されていません。

1: インジェクト・グループの変換が開始されました。

Bit 2 **JEOC**: インジェクト・チャネル変換終了

このビットはインジェクト・グループ・チャネル変換がすべて終了したときに、ハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: 変換は完了していません。

1: 変換が完了しました。

Bit 1 **EOC**: 変換終了フラグ

このビットは、グループ・チャネル変換（レギュラもしくはインジェクト）が終了したときに、ハードウェアによってセットされます。フラグは ADC_DR レジスタを読み出したとき、もしくはソフトウェアによるステータス・レジスタへの書き込みで行います。

0: 変換は完了していません。

1: 変換が完了しました。

Bit 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、あらかじめ ADC_HTR レジスタ及び ADC_LTR レジスタで設定されている上下限を超えたときに、ハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: アナログ・ウォッチドッグ・イベントは発生していません。

1: アナログ・ウォッチドッグ・イベントが発生しました。

参考資料

RM0008

A/D コンバータ (ADC)

10.12.2 ADC 制御レジスタ 1 (ADC_CR1)

アドレスオフセット : 04h

リセット値 : 0000 0000h

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み								AWDEN	JAWDEN	予約済み		DUALMOD[3:0]			
								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	DISCEN	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

Bits 31:24 予約済み、書き込み時は常に 0 としてください。

Bit 23 AWDEN: アナログ・ウォッチドッグ有効 (レギュラ・チャネル)

このビットはソフトウェアでセット / クリアします。

0: レギュラ・チャネルに対するウォッチドッグ機能は無効です。

1: レギュラ・チャネルに対するウォッチドッグ機能は有効です。

Bit 22 JAWDEN: アナログ・ウォッチドッグ有効 (インジェクト・チャネル)

このビットはソフトウェアでセット / クリアします。

0: インジェクト・チャネルに対するウォッチドッグ機能は無効です。

1: インジェクト・チャネルに対するウォッチドッグ機能は有効です

Bits 21:20 予約済み、書き込み時は常に 0 としてください。

Bits 19:16 DUALMOD[3:0]: デュアル ADC モードの選択

これらのビットはソフトウェアで動作モードを選択するため、ソフトウェアで書込みます。

0000: 独立モード

0001: 組合せ動作 : 同時レギュラ + 同時インジェクト・モード

0010: 組合せ動作 : 同時レギュラ + オルタネート・トリガ・モード

0011: 組合せ動作 : 同時インジェクト + 高速インタリーブ・モード

0100: 組合せ動作 : 同時インジェクト + 低速インタリーブ・モード

0101: 同時インジェクト・モードのみ

0110: 同時レギュラ・モードのみ

0111: 高速インタリーブ・モードのみ

1000: 低速インタリーブ・モードのみ

1001: オルタネート・トリガ・モードのみ

注:

-これらのビットは ADC2 及び ADC3 では使用されません。(予約ビットです)

-チャネル選択の変更の際、動作の再スタートが行われ、デュアル ADC モードでは同期状態が崩れことがあります。チャネル選択を変更する前に、デュアル ADC モードを無効としておくことが勧められます。



参考資料

Bits 15:13 DISCNUM[2:0]: 分割モード・チャネル数

これらのビットには、レギュラ変換グループで分割スキャンを行う際の、個々トリガで変換されるチャネルの数をソフトウェアで設定します。

000: 1 チャネル
001: 2 チャネル
.....

111: 8 チャネル

Bit 12 JDISCEN: インジェクト・チャネル上の分割モード

このビットは、インジェクト・グループ・チャネルの分割モードを有効 / 無効にするために、ソフトウェアによりセット / クリアします。

0: インジェクト・チャネルの分割モードは無効です。
1: インジェクト・チャネルの分割モードは有効です。

Bit 11 DISCEN: レギュラ・チャネル上の分割モード

このビットは、レギュラ・チャネルの分割モードを有効 / 無効にするために、ソフトウェアによりセット / クリアします。

0: レギュラ・チャネルの分割モードは無効です。
1: レギュラ・チャネルの分割モードは有効です。

Bit 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・グループ変換後の自動インジェクト・グループ変換を有効 / 無効にするために、ソフトウェアによりセット / クリアします。

0: 自動インジェクト・グループ変換は無効です。
1: 自動インジェクト・グループ変換は有効です。

Bit 9 AWDSGL: スキャン・モードでのシングル・チャネル上のウォッチドッグ・イネーブル

このビットは、AWDCH[4:0] ビットにより指定されたチャネル上のアナログ・ウォッチドッグを有効 / 無効にするために、ソフトウェアでセット / クリアします。

0: アナログ・ウォッチドッグ機能はすべてのチャネルで有効です。
1: アナログ・ウォッチドッグ機能はシングル・チャネルでのみ有効です。

Bit 8 SCAN: スキャン・モード

このビットはスキャン・モードの有効 / 無効を切替えるため、ソフトウェアでセット / クリアします。スキャン・モードでは、ADC_SQRx 及び ADC_JSEQR レジスタで指定した入力の変換が行われます。

0: スキャン・モードは無効です。
1: スキャン・モードは有効です。

注: EOCIE もしくは JEOCIE ビットがセットされていれば、指定されている最後の変換が終了したとき、EOC もしくは JEOC 割込みリクエストが発生します。

Bit 7 JEOCIE: インジェクト・チャネルの割込み許可

このビットはインジェクト・チャネルについての変換割込み終了を有効 / 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: JEOC 割込み無効。
1: JEOC 割込み有効。JEOC ビットがセットされると割込みが発生します。

参考資料

RM0008

A/D コンバータ (ADC)

Bit 6 **AWDIE**: アナログ・ウォッヂドッグ割込み許可

このビットはアナログ・ウォッヂドッグ割込みの有効 / 無効を切替えるため、ソフトウェアでセット / クリアします。スキャン・モードで、ウォッヂドッグの上下限が逆転しているケースでは、このビットがセットされている場合に限ってスキャンがアポートされます。

0: アナログ・ウォッヂドッグ割込み無効。

1: アナログ・ウォッヂドッグ割込み有効。

Bit 5 **EOCIE**: EOC 割込み有効

このビットは、変換終了時の割込みの有効 / 無効を切替えるため、ソフトウェアでセット / クリアします。

0: EOC 割込み無効。

1: EOC 割込み有効。JEOC ビットがセットされると割込みが発生します。

Bits 4:0 **AWDCH[4:0]**: アナログ・ウォッヂドッグ・チャネル選択ビット

これらのビットはソフトウェアによりセット / クリアされます。これらは、アナログ・ウォッヂドッグにより保護される入力チャネルを選択します。

00000: ADC アナログ入力チャネル 0

00001: ADC アナログ入力チャネル 1

....

01111: ADC アナログ入力チャネル 15

10000: ADC アナログ入力チャネル 16

10001: ADC アナログ入力チャネル 17

他の値は予約されており、設定できません。

注:

- ADC1 アナログ入力チャネル 16 とチャネル 17 は、内部で温度センサと V_{REFINT} にそれぞれ接続されています。
- ADC2 アナログ入力チャネル 16 とチャネル 17 は、内部で V_{SS} へ接続されています。
- ADC3 アナログ入力チャネル 9、チャネル 14、チャネル 15、チャネル 16 とチャネル 17 は、 V_{SS} へ接続されています。



参考資料

10.12.3 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								TSVREFE FE	SWSTART ART	JSWSTART ART	EXTTRIG RIG	EXTSEL[2:0]		予約済み	
								RW	RW	RW	RW	RW	RW	RW	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JEXTTRIG	JEXTSEL[2:0]		ALIGN	予約済み	DMA	予約済み				RSTCAL	CAL	CONT	ADON		
RW	RW	RW	RW	RW	RW					RW	RW	RW	RW	RW	

Bits 31:24 予約済み、書き込み時は常に 0 としてください。

Bit 23 TSVREFE: 温度センサと V_{REFINT} 有効

このビットは、温度センサ及び V_{REFINT} チャネルを有効とするか否かを許可・禁止を切替えるため、ソフトウェアでセット / クリアします。2 回路の ADC を有するデバイスでは、このビットは ADC1 に対してのみ有効です。

0: 温度センサ及び V_{REFINT} チャネルは機能しません。

1: 温度センサ及び V_{REFINT} チャネル有効です。

Bit 22 SWSTART: レギュラ・チャネルの変換開始

このビットは、ソフトウェアでセットすることにより変換が開始され、変換開始直後にハードウェアによってクリアされます。EXTSEL[2:0] で外部トリガとして SWSTART が選択されれば、レギュラ・グループの変換が開始されます。

0: リセット状態

1: レギュラ・チャネルの変換が開始されます。

Bit 21 JSWSTART: インジェクト・チャネルの変換開始

このビットは、ソフトウェアでセットすることにより変換が開始され、変換開始直後にハードウェアによってクリアされます。JEXTSEL[2:0] で外部トリガとして SWSTART が選択されれば、インジェクト・チャネルのグループの変換が開始されます。

0: リセット状態

1: インジェクト・チャネルの変換が開始されます。

Bit 20 EXTTRIG: レギュラ・チャネルについての外部トリガ変換モード

このビットは、レギュラ・チャネルの変換を開始する外部トリガ機能の有効 / 無効を切替えるため、ソフトウェアでセット / クリアします。

0: 外部イベントによる変換は行われません。

1: 外部イベントによる変換が行われます。

参考資料

RM0008

A/D コンバータ (ADC)

Bits 19:17 EXTSEL[2:0]: レギュラ・チャネルについての外部イベント選択

これらのビットは、レギュラ・グループの変換を開始するトリガを使用するための外部イベントを選択します。

ADC1 及び ADC2 関連のトリガは、

- 000: タイマ 1 CC1 イベント
- 001: タイマ 1 CC2 イベント
- 010: タイマ 1 CC3 イベント
- 011: タイマ 2 CC2 イベント
- 100: タイマ 3 TRGO イベント
- 101: タイマ 4 CC4 イベント
- 110: EXTI ライン 11/TIM8_TRGO イベント (TIM8_TRGO は大容量デバイスのみで利用可能)
- 111: SWSTART

ADC3 関連のトリガは、

- 000: タイマ 3 CC1 イベント
- 001: タイマ 2 CC3 イベント
- 010: タイマ 1 CC3 イベント
- 011: タイマ 8 CC1 イベント
- 100: タイマ 8 TRGO イベント
- 101: タイマ 5 CC1 イベント
- 110: タイマ 5 CC3 イベント
- 111: SWSTART

Bit 16 予約済み、書込み時は常に 0 としてください。

Bit 15 JEXTTRIG: インジェクト・チャネルの外部トリガ変換モード

このビットは、インジェクト・チャネル・グループの変換を開始する外部トリガ機能の有効 / 無効を切替えるため、ソフトウェアでセット / クリアします。

- 0: 外部イベントによる変換は行われません。
- 1: 外部イベントによる変換が行われます。

Bits 14:12 JEXTSEL[2:0]: インジェクト・グループについての外部イベント選択

これらのビットは、インジェクト・グループの変換を開始するトリガを使用するための外部イベントを選択します。

ADC1 及び ADC2 関連のトリガは、

- 000: タイマ 1 TRGO イベント
- 001: タイマ 1 CC4 イベント
- 010: タイマ 2 TRGO イベント
- 011: タイマ 2 CC1 イベント
- 100: タイマ 3 CC4 イベント
- 101: タイマ 4 TRGO イベント
- 110: EXTI ライン 15/TIM8_CC4 イベント (TIM8_CC4 は大容量デバイスのみで利用可能)
- 111: JSWSTART

ADC3 関連のトリガは、

- 000: タイマ 1 TRGO イベント
- 001: タイマ 1 CC4 イベント
- 010: タイマ 4 CC3 イベント
- 011: タイマ 8 CC2 イベント
- 100: タイマ 8 CC4 イベント
- 101: タイマ 5 TRGO イベント
- 110: タイマ 5 CC4 イベント
- 111: JSWSTART



参考資料

Bit 11 ALIGN: データ・アライメント

このビットはソフトウェアでセット / クリアされます。図 26 及び図 27 を参照してください。

0: 右 (下位) 寄せ

1: 左 (上位) 寄せ

Bits 10:9 予約済み、書き込み時は常に 0 としてください。

Bit 8 DMA: DMA モード

このビットはソフトウェアでセット / クリアされます。詳細は DMA コントローラの章を参照してください。

0: DMA モードは無効です。

1: DMA モードは有効です。

注: ADC1 と ADC3 のみで DMA リクエストを生成することができます。

Bits 7:4 予約済み、書き込み時は常に 0 としてください。

Bit 3 RSTCAL: 較正値リセット

このビットはソフトウェアでリセットしますが、ハードウェアでクリアされることもあります。
較正値レジスタがクリアされると、このビットはクリアされます。

0: 較正値レジスタはすでに初期化されています。

1: 較正値レジスタの初期化を行います。

注: 変換の実行中に RSTCAL ビットをセットした場合は、較正レジスタをクリアするために、より多くのサイクルが必要とされます。

Bit 2 CAL: A/D 較正

このビットをソフトウェアでセットすると、較正が開始されます。較正が終了すると、ハードウェアによってこのビットがリセットされます。

0: 較正は終了しました。

1: 較正を開始します。

Bit 1 CONT: 連続変換

このビットは、ソフトウェアによりセット / クリアされます。ビットをリセットするまで連続変換が行われます。

0: シングル変換モード

1: 連続変換モード

Bit 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りセット / クリアされます。このビットが 0 であるときに 1 を書込むと、ADC をパワー・ダウン状態から起動します。

このビットが 1 の状態で、さらに 1 を書込むと、変換動作が開始されます。アプリケーションでは、起動から変換までに t_{STAB} の遅延が発生します。図 22 を参照してください。

0: ADC の変換 / 較正を中断し、パワー・ダウン状態に移行します。

1: ADC を有効とし、変換動作を開始します。

注: このレジスタの ADON 以外のビットが 変更されたときには、この書き込みによって変換が開始されることはありません。これにより、設定時に誤って変換が開始されることを防いでいます。

参考資料

10.12.4 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								SMP17[2:0]			SMP16[2:0]			SMP15[2:1]	
Res.								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP 15_0	SMP14[2:0]			SMP13[2:0]			SMP12[2:0]			SMP11[2:0]			SMP10[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:24 予約済み、書込み時は常に 0 してください。

Bits 23:0 SMPx[2:0]: チャネル x サンプル・タイム選択

これらのビットは、個々のチャネルごとにサンプル・タイムを選択するために、ソフトウェアで設定します。変換動作のサンプル・タイムの間は、これらのビットは変更することはできません。

- 000: 1.5 サイクル
- 001: 7.5 サイクル
- 010: 13.5 サイクル
- 011: 28.5 サイクル
- 100: 41.5 サイクル
- 101: 55.5 サイクル
- 110: 71.5 サイクル
- 111: 239.5 サイクル

注:

- ADC1 アナログ入力チャネル 16 とチャネル 17 は、内部的に温度センサと V_{REFINT} にそれぞれ接続されています。
- ADC2 アナログ入力チャネル 16 とチャネル 17 は、内部で V_{SS} と接続されています。
- ADC3 アナログ入力チャネル 14、チャネル 15、チャネル 16 とチャネル 17 は、V_{SS} へ接続されています。

参考資料

10.12.5 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	SMP9[2:0]		SMP8[2:0]		SMP7[2:0]		SMP6[2:0]		SMP5[2:1]						
	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								

Bits 31:30 予約済み、書込み時は常に 0 としてください。

Bits 29:0 SMPx[2:0]: チャネル x サンプル・タイム選択

これらのビットは、個々のチャネルごとにサンプル・タイムを選択するために、ソフトウェアで設定します。変換動作のサンプル・タイムの間は、これらのビットは変更することはできません。

000: 1.5 サイクル

001: 7.5 サイクル

010: 13.5 サイクル

011: 28.5 サイクル

100: 41.5 サイクル

101: 55.5 サイクル

110: 71.5 サイクル

111: 239.5 サイクル

注: ADC3 アナログ入力チャネル 9 は、V_{SS} へ接続されています。

10.12.6 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 : 0x14–0x20

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
予約済み	JOFFSETx[11:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:12 予約済み、書込み時は常に 0 としてください。

Bits 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場合に、生変換データから引かれたオフセットを定義するために、ソフトウェアにより設定されます。引き算後の変換結果は ADC_JDRx レジスタから読出されます。

参考資料

RM0008

A/D コンバータ (ADC)

10.12.7 ADC ウオッチドッグ上限スレッショルド・レジスタ (ADC_HTR)

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
予約済み				HT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:12 予約済み、書き込み時は常に 0 としてください。

Bits 11:0 HT[11:0] アナログ・ウォッチドッグ上限スレッショルド

これらのビットにはソフトウェアで、アナログ・ウォッチドッグ機能の上限スレッショルド値を設定します。

10.12.8 ADC ウオッチドッグ下限スレッショルド・レジスタ (ADC_LTR)

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
予約済み				LT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:12 予約済み、書き込み時は常に 0 としてください。

Bits 11:0 LT[11:0] アナログ・ウォッチドッグ下限スレッショルド

これらのビットにはソフトウェアで、アナログ・ウォッチドッグ機能の下限スレッショルド値を設定します。

10.12.9 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 : 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								L[3:0]				SQ16[4:1]			
Res.								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
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	rw	rw	rw



参考資料

Bits 31:24 予約済み、書き込み時は常に 0 としてください。

Bits 23:20 L[3:0]: レギュラ・チャネル・シーケンス長

これらのビットは、レギュラ・チャネル変換シーケンスの変換合計数を定義するために、ソフトウェアで設定します。

0000: 1 変換

0001: 2 変換

....

1111: 16 変換

Bits 19:15 SQ16[4:0]: 16 番目のレギュラ・シーケンス

これらのビットは、16 番目の変換シーケンスとしてチャネル番号 (0..17) を割当てるために、ソフトウェアにより設定されます。

Bits 14:10 SQ15[4:0]: 15 番目のレギュラ・シーケンス

Bits 9:5 SQ14[4:0]: 14 番目のレギュラ・シーケンス

Bits 4:0 SQ13[4:0]: 13 番目のレギュラ・シーケンス

10.12.10 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 : 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		SQ12[4:0]				SQ11[4:0]				SQ10[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
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

Bits 31:30 予約済み、書き込み時は常に 0 としてください。

Bits 29:26 SQ12[4:0]: 12 番目のレギュラ・シーケンス

これらのビットは、12 番目の変換シーケンスとしてチャネル番号 (0..17) を割当てるために、ソフトウェアにより設定されます。

Bits 24:20 SQ11[4:0]: 11 番目のレギュラ・シーケンス

Bits 19:15 SQ10[4:0]: 10 番目のレギュラ・シーケンス

Bits 14:10 SQ9[4:0]: 9 番目のレギュラ・シーケンス

Bits 9:5 SQ8[4:0]: 8 番目のレギュラ・シーケンス

Bits 4:0 SQ7[4:0]: 7 番目のレギュラ・シーケンス

参考資料

RM0008

A/D コンバータ (ADC)

10.12.11 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 : 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		SQ6[4:0]					SQ5[4:0]					SQ4[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
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

Bits 31:30 予約済み、書き込み時は常に 0 としてください。

Bits 29:25 SQ6[4:0]: 6 番目のレギュラ・シーケンス

これらのビットは、6 番目の変換シーケンスとしてチャネル番号 (0..17) を割当てるために、ソフトウェアにより設定されます。

Bits 24:20 SQ5[4:0]: 5 番目のレギュラ・シーケンス

Bits 19:15 SQ4[4:0]: 4 番目のレギュラ・シーケンス

Bits 14:10 SQ3[4:0]: 3 番目のレギュラ・シーケンス

Bits 9:5 SQ2[4:0]: 2 番目のレギュラ・シーケンス

Bits 4:0 SQ1[4:0]: 1 番目のレギュラ・シーケンス

10.12.12 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 : 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		JSQ4[4:1]					JL[1:0]					JSQ1[4:0]			
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
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



参考資料

Bits 31:22 予約済み、書き込み時は常に 0 としてください。

Bits 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変換合計数を定義するために、ソフトウェアで設定します。

00: 1 変換

01: 2 変換

10: 3 変換

11: 4 変換

Bits 19:15 JSQ4[4:0]: 4 番目のインジェクト・シーケンス

これらのビットは、4 番目の変換シーケンスとしてチャネル番号 (0..17) を割当てるために、ソフトウェアにより設定されます。

注: レギュラ変換シーケンスと異なり、JL[1:0] が 4 より小さなときには変換は (4-JL) 番目の項目から開始されます。

例: ADC_JSQR[21:0] = 10 00011 00011 00111 00010 のときには、7、3、3 の順序で変換が行われます。
(2、7、3 ではありません。)

Bits 14:10 JSQ3[4:0]: 3 番目のインジェクト・シーケンス

Bits 9:5 JSQ2[4:0]: 2 番目のインジェクト・シーケンス

Bits 4:0 JSQ1[4:0]: 1 番目のインジェクト・シーケンス

10.12.13 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 : 0x3C – 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
JDATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:16 予約済み、書き込み時は常に 0 としてください。

Bits 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。これらのビットには、インジェクト・チャネル x かの変換結果が格納されます。データは、図 26 及び図 27 で示されるように、左もしくは右に寄せられています。

参考資料

RM0008

A/D コンバータ (ADC)

10.12.14 ADC レギュラ・データ・レジスタ (ADC_DR)

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
ADC2DATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:16 ADC2DATA[15:0]: ADC2 データ

- ADC1 では、デュアル・モードのとき、これらのビットには ADC2 のレギュラ・データが格納されます。[セクション 10.9: デュアル ADC モード](#)を参照してください。
- ADC2 及び ADC3 では、これらのビットは使用されません。

Bits 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。これらのビットには、インジェクト・チャネル x の変換結果が格納されます。データは、[図 26](#) 及び [図 27](#) で示されるように、左もしくは右に寄せられています。

10.12.15 ADC レジスタ・マップ

次の表は ADC レジスタの一覧です。

表 48. ADC - レジスタ・マップとリセット時の値

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 リセット値	予約済み																0	SRTT	4	3	JSTRT	2	EOC	1	AWD	0	0					
0x04	ADC_CR1 リセット値	予約済み				AWDEN 0	JAWDEN 0	予約済み		DUALMOD [3:0] 0 0 0 0	DISC NUM [2:0] 0 0 0 0	JDISCEN 0 0 0 0	DISCEN 0 0 0 0	JAUTO 0 0 0 0	AWD SGL 0 0 0 0	SCAN 0 0 0 0	JEOCIE 0 0 0 0	AWDIE 0 0 0 0	ECIE 0 0 0 0	AWDCH[4:0]													
0x08	ADC_CR2 リセット値	予約済み				TSVREFE 0 0	SWSTART 0 0	JSWSTART 0 0	EXTTRIG 0 0 0 0	EXTSEL [2:0] 0 0 0 0	JEEXTSEL [2:0] 0 0 0 0	JEXTTRIG 0 0 0 0	JEXTSEL [2:0] 0 0 0 0	ALIGN 0 0 0 0	予約済み 0 0 0 0	DMA 0 0 0 0	予約済み				RSTCAL 0 0 0 0	CAL 0 0 0 0	CONT 0 0 0 0	ADON 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 リセット値	Sample time bits SMPx_x																															
0x10	ADC_SMPR2 リセット値	Sample time bits SMPx_x																															
0x14	ADC_JOFR1 リセット値	予約済み												JOFFSET1[11:0]																			
0x18	ADC_JOFR2 リセット値	予約済み												JOFFSET2[11:0]																			

参考資料

表 48. ADC - レジスタ・マップとリセット時の値 (続き)

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1C	ADC_JOFR3 リセット値	予約済み																JOFFSET3[11:0] 0															
0x20	ADC_JOFR4 リセット値	予約済み																JOFFSET4[11:0] 0															
0x24	ADC_HTR リセット値	予約済み																HT[11:0] 0															
0x28	ADC_LTR リセット値	予約済み																LT[11:0] 0															
0x2C	ADC_SQR1 リセット値	予約済み						L[3:0]		レギュラ・チャネル・シーケンス SQx_x bits 0																							
0x30	ADC_SQR2 リセット値	予約済み	レギュラ・チャネル・シーケンス SQx_x bits 0																														
0x34	ADC_SQR3 リセット値	予約済み	レギュラ・チャネル・シーケンス SQx_x bits 0																														
0x38	ADC_JSQR リセット値	予約済み						JL[1:0]	インジェクト・チャネル・シーケンス JSQx_x bits 0																								
0x3C	ADC_JDR1 リセット値	予約済み																JDATA[15:0] 0															
0x40	ADC_JDR2 リセット値	予約済み																JDATA[15:0] 0															
0x44	ADC_JDR3 リセット値	予約済み																JDATA[15:0] 0															
0x48	ADC_JDR4 リセット値	予約済み																JDATA[15:0] 0															
0x4C	ADC_DR リセット値	ADC2DATA[15:0]																Regular DATA[15:0] 0															

レジスタのアドレス範囲に関しては、表 1 (36 ページ) を参照してください。

11 D/A コンバータ (DAC)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、大容量デバイスにのみ適応されます。

11.1 概要

DAC モジュールは 12bit、電圧出力デジタルーアナログ・コンバータです。DAC は 8 または 12bit モードに設定され、DMA コントローラと共に使用することも出来ます。12bit モードでは、データは左または右寄せにすることが出来ます。DAC は 2 つの出力チャネルを持っており、それぞれがコンバータを搭載しています。デュアル DAC チャネル・モードにおいては、同期アップデート・オペレーションのために両チャネルがグループ化されなければ、変換は独立あるいは同時に進行が可能です。高分解能用に、入力基準ピン V_{REF+} が用意されています。

11.2 DAC の主な機能

- 2 つの DAC コンバータ：それぞれ 1 つの出力チャネル
- 8bit または 12bit の単調出力
- 12bit モードの左または右データ寄せ
- 同期アップデート機能
- ノイズ波形生成
- 三角波生成
- デュアル DAC チャネル独立または同時変換
- 両チャネル用 DMA
- 変換用外部トリガ
- 入力基準電圧 V_{REF+}

DAC チャネルのブロック図は [図 39](#) に、ピン記述は [表 49](#) に示されています。

参考資料

図 39. DAC チャネル・ブロック図

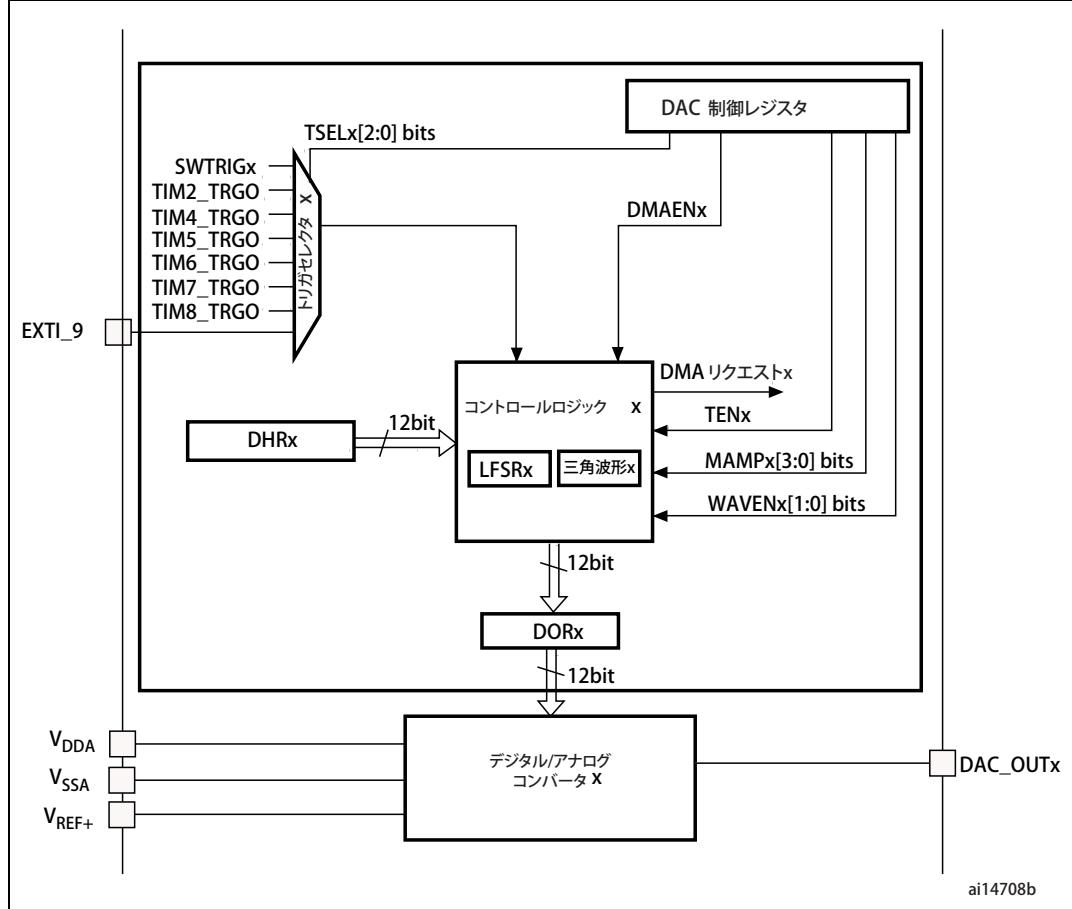


表 49. DAC ピン

名称	信号種別	摘要
V_{REF+}	入力、 アナログ基準電圧 +	DAC の基準電圧、ハイレベル / 正電圧 $2.4 \text{ V} \leq V_{REF+} \leq V_{DDA} (3.3 \text{ V})$
V_{DDA}	入力、 アナログ電源供給	アナログ回路電源
V_{SSA}	入力、 アナロググランド	アナログ回路電源のグランド
DAC_OUTx	アナログ出力信号	DAC チャネル x アナログ出力

注意：一度 DAC チャネルがイネーブルされると、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。寄生消費電力を防ぐために、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。

11.3 DAC 機能詳細

11.3.1 DAC チャネル・イネーブル

各 DAC チャネルは、DAC_CR レジスタの対応する EN_x ビットを設定する事により起動することができます。そして DAC チャネルは、t_{WAKEUP} 後にイネーブルされます。

注意： EN_x ビットは、アナログ DAC チャネルのマクロセルのみをイネーブルします。DAC チャネル x デジタル・インターフェースは、EN_x ビットがリセットされた場合でもイネーブルされます。

11.3.2 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すことができ、外部動作アンプなしに負荷を駆動できる 2 つの出力バッファを内蔵しています。各 DAC チャネル出力バッファは、DAC_CR レジスタの対応する BOFF_x ビットによりイネーブル / ディセーブルすることができます。

11.3.3 DAC データ・フォーマット

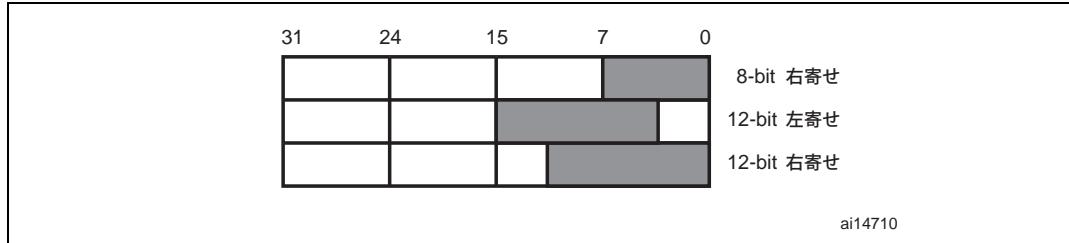
次のように、選択された構成モードにより、データは指定されたレジスタへ書込む必要があります。

- シングル DAC チャネル x では、3 つの状況が考えられます：
 - 8bit 右寄せ：ユーザは、データを DAC_DHR8Rx[7:0] ビット (DHRx[11 : 4] ビットへ保存) ヘロードしなければなりません。
 - 12bit 左寄せ：ユーザは、データを DAC_DHR12L x[15:4] ビット (DHRx[11 : 0] ビットへ保存) ヘロードしなければなりません。
 - 12bit 右寄せ：ユーザは、データを DAC_DHR12Rx[11:0] ビット (DHRx[11 : 0] ビットへ保存) ヘロードしなければなりません。

ロードされた DAC_DHRyyyx レジスタの内容により、ユーザが書込んだデータはシフトされ、DHRx (メモリへ配置されない内部レジスタであるデータ保持レジスタ x) へ保存されます。そして DHRx レジスタは、自動的または外部イベントによるソフトウェア・トリガにより、DORx レジスタへロードされます。

参考資料

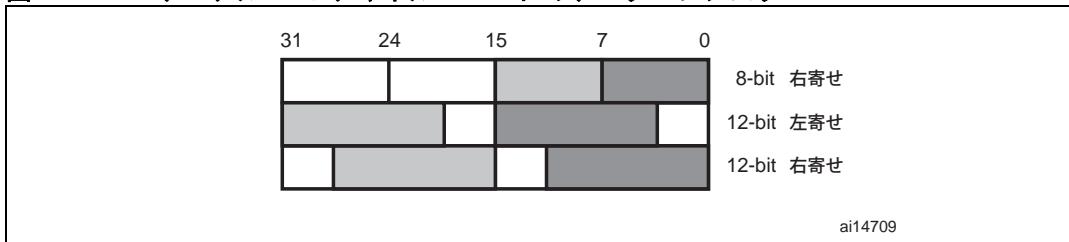
図 40. シングル DAC チャネル・モードのデータ・レジスタ



- デュアル DAC チャネルでは、3 つの状況が考えられます。
 - 8bit 右寄せ : DAC チャネル 1 へのデータは、データを DAC_DHR8RD[7:0] ビット (DHR1[11 : 4] ビットへ保存) へ、DAC チャネル 2 へのデータは、DAC_DHR8RD[15:8] ビット (DHR2[11 : 4] ビットへ保存) ヘロードされます。
 - 12bit 左寄せ : DAC チャネル 1 へのデータは、データを DAC_DHR12LD[15:4] ビット (DHR1[11 : 0] ビットへ保存) へ、DAC チャネル 2 へのデータは、DAC_DHR12LD[31:20] ビット (DHR2[11 : 0] ビットへ保存) ヘロードされます。
 - 12bit 右寄せ : DAC チャネル 1 へのデータは、データを DAC_DHR12RD[11:0] ビット (DHR1[11 : 0] ビットへ保存) へ、DAC チャネル 2 へのデータは、DAC_DHR12LD[27:16] ビット (DHR2[11 : 0] ビットへ保存) ヘロードされます。

ロードされた DAC_DHRyyxD レジスタの内容により、ユーザが書込んだデータはシフトされ、DHR1 と DHR2 (メモリへ配置されない内部レジスタであるデータ保持レジスタ) へ保存されます。そして DHR1 及び DHR2 レジスタは、自動的または外部イベントによるソフトウェア・トリガにより、DOR1 と DOR2 レジスタヘロードされます。

図 41. デュアル DAC チャネル・モードのデータ・レジスタ



11.3.4 DAC 変換

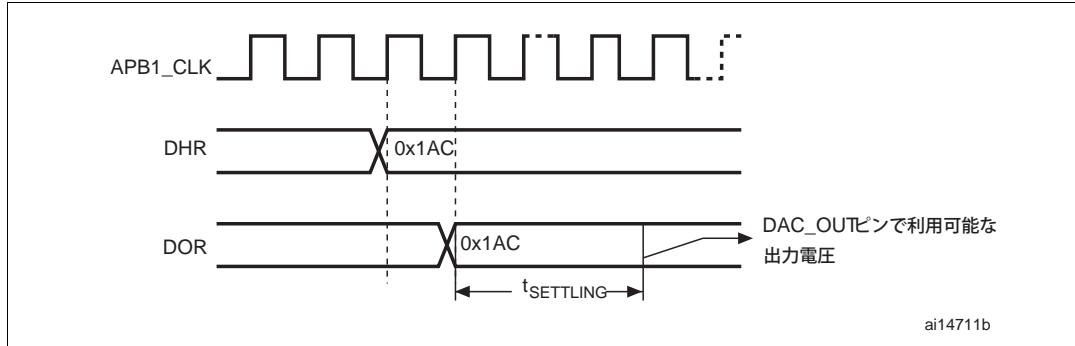
DAC_DORx へは、直接書込むことは出来ません。したがって、DAC チャネル x へのデータ転送は DAC_DHRx レジスタ (DAC_DHR8Rx、DAC_DHR12Lx、DAC_DHR12Rx、DAC_DHR8RD、DAC_DHR12LD または DAC_DHR12LD への書き込み) へのロードにより行う必要があります。

DAC_DHRx レジスタへ保存されたデータは、ハードウェアトリガが選択されていなければ、1APB1 クロック・サイクル後に DAC_DORx レジスタへ自動的に転送されます。しかしながら、ハードウェアトリガが選択されており (DAC_CR レジスタの TENx がセットされている)、尚且つトリガが発生すると、転送は 3APB1 クロック・サイクル後に行われます。

DAC_DORx に DAC_DHRx の内容がロードされると、出力電圧は、電源電圧とアナログ出力負荷に応じて決定される $t_{SETTLING}$ 時間後に利用可能となります。

参考資料

図 42. トリガ・ディセーブル時 (TEN = 0) の変換タイミング図



11.3.5 DAC 出力電圧

デジタル入力は、0 から V_{REF+} のリニア変換により出力電圧へ変換されます。

各 DAC チャネルピンのアナログ出力電圧は、次の式により決定されます。

$$DAC_{Output} = V_{REF} \times \frac{DOR}{4095}$$

11.3.6 DAC トリガ選択

TENx 制御ビットが外部イベント（タイマ・カウンタ、外部割込みピン）によりセットされると、TSELx[2 : 0] 制御ビットは、表 50 に示す 8 つの中のどのイベントが変換をトリガするか決定します。

表 50. 外部トリガ

発生元	種別	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
タイマ 8 TRGO イベント		001
タイマ 7 TRGO イベント		010
タイマ 5 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 line9	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 TRGO タイマ出力、あるいは選択された外部割込みライン 9 上に立上りエッジを検出する度に、DAC_DHRx レジスタに最後に保存されたデータは、DAC_DORx レジスタへ転送されます。DAC_DORx レジスタは、トリガが発生してから 3APB1 サイクル後にアップデートされます。

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 レジスタへロードされるとリセットされます。

- 注意:
- 1 ENx ビットがセットされていると、TSELx[2:0] ビットは変更することが出来ません。
 - 2 ソフトウェア・トリガが選択されていると、DAC_DHRx から DAC_DORx レジスタへの転送は僅か 1APB1 クロック・サイクルで行われます。

11.3.7 DMA リクエスト

各 DAC チャネルは、DMA を持っています。DAC チャネルの DMA リクエストは、2 つの DMA チャネルにより処理されます。

DAC DMA リクエストは、DMAENx ビットがセットされており外部トリガ（ソフトウェア・トリガでなく）が起きた時に発生します。その後 DAC_DHRx レジスタの値は、DAC_DORx レジスタへ転送されます。

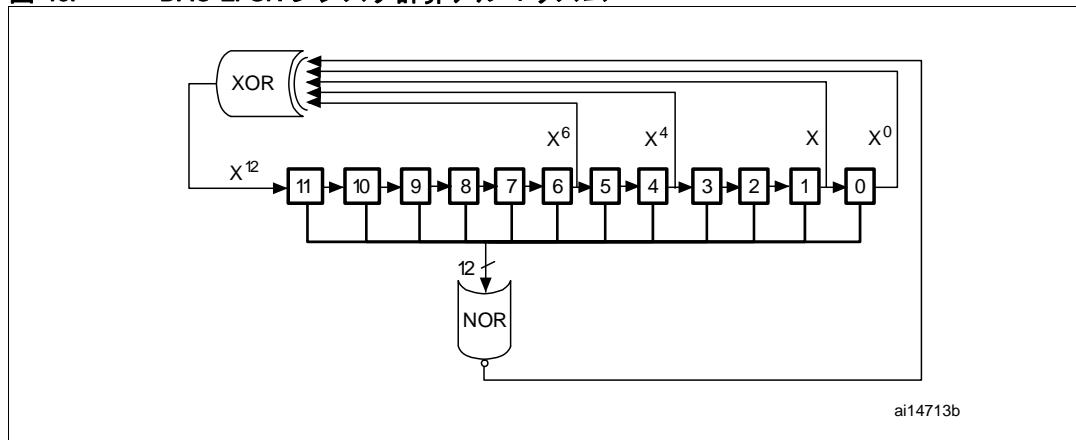
デュアル・モードでは、両方の DMAENx ビットがセットされていると、2 つの DMA リクエストが発生します。もし 1 つの DMA リクエストのみ必要ならば、対応する 1 つの DMAENx ビットをセットします。これにより、アプリケーションは各 DMA リクエストに対して個別の DMA チャネルを使う事で、両方の DAC チャネルをデュアル・モードで管理することが出来ます。

DAC DMA リクエストはキューされないので、最後の DMA リクエストへの応答の前に 2 つの外部トリガが発生すると、新たなリクエストは無視されエラーも報告されません。

11.3.8 ノイズ生成

可変振幅の擬似ノイズの生成には、リニア・フィードバック・シフト・レジスタ (LFSR) が利用できます。DAC ノイズ生成は、WAVEx[1:0] を 01 にセットすることで選択されます。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 3APB1 クロック・サイクル後に、特定のアルゴリズムに従ってアップデートされます。

図 43. DAC LFSR レジスタ計算アルゴリズム



ai14713b

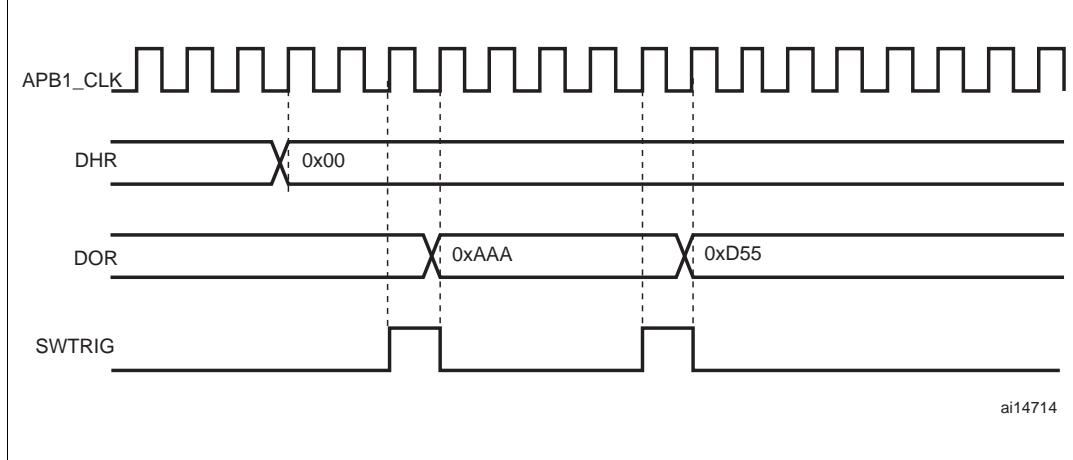
DAC_CR レジスタの MAMPx[3 : 0] ビットにより、部分的あるいは全体がマスクされる LFSR 値は、オーバフローなしに DAC_DHRx の内容に加算されます。そして、この値は DAC_DORx レジスタへ保存されます。

LFSR が 0x0000 の場合、1 が入れられます。(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事で、LFSR 波形生成のリセットが可能です。

参考資料

図 44. LFSR 波形生成による DAC 変換 (SW トリガ・イネーブル)



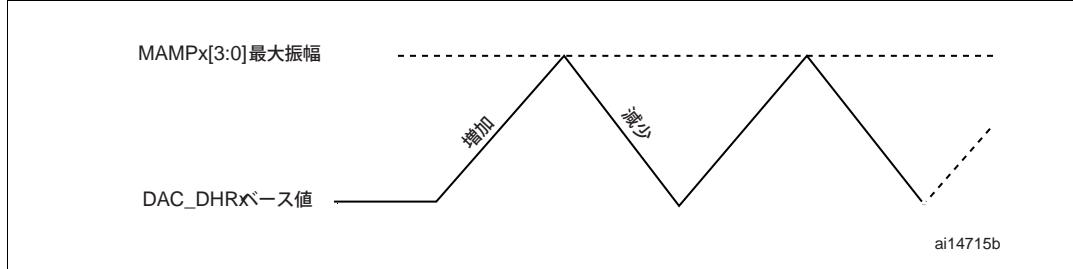
注意：ノイズ生成のためにには、*DAC_CR* レジスタの *TENx* ビットをセットし、*DAC* トリガをイネーブルしなければなりません。

11.3.9 三角波生成

DC または低周波上に、小さな振幅の三角波を追加することができます。DAC 三角波生成は、*WAVEx[1 : 0]* を 10 にセットすることで選択されます。振幅は *DAC_CR* レジスタの *MAMPx[3 : 0]* ビットを通じて設定されます。内部のカウンタは、各イベントの 3APB1 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、オーバフローなしに *DAC_DHRx* レジスタへ加えられ、合計は *DAC_DORx* レジスタへ保存されます。このカウンタは、*MAMPx[3 : 0]* ビットにより定義された最大振幅より、転送された値が小さかった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を行います。

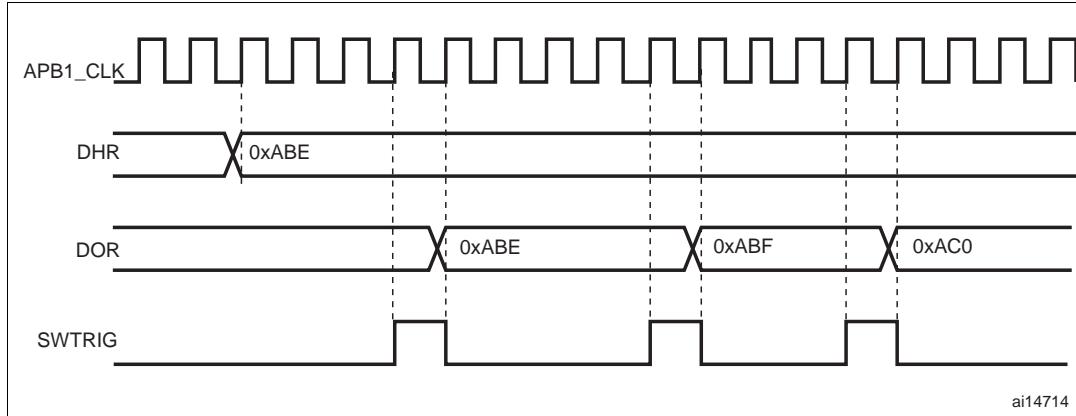
三角波生成は *WAVEx[1 : 0]* ビットをリセットする事で、リセット可能です。

図 45. DAC 三角波生成



参考資料

図 46. 三角波生成による DAC 変換 (SW トリガ・イネーブル)



- 注意:
- 1 ノイズ生成には、*DAC_CR* レジスタの *TENx* ビットをセットし *DAC* トリガをイネーブルしなければなりません。
 - 2 *DAC* をイネーブルする前に *MAMPx[3:0]* ビットを設定しなければ、それらを変更することは出来ません。

11.4 デュアル DAC チャネル変換

同時に 2 つの DAC チャネルを必要とするアプリケーションで、バスバンド幅を有効に使うために、DHR8RD、DHR12RD と DHR12LD の 3 つのデュアル・レジスタが搭載されています。2 つの DAC チャネルを同時に使うには、特別なレジスタ・アクセスが必要です。

2 つの DAC チャネルと 3 つのデュアル・レジスタを使うことで、11 の変換モードが利用可能です。全ての変換モードは、必要な場合には、別の DHRx レジスタを使っても利用できます。

全てのモードは、次の節に記載されています。

11.4.1 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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに異なる値をセットし、異なるトリガソースを設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

DAC チャネル 1 トリガが発生すると、DHR1 レジスタは DAC_DOR1 へ転送されます。
(3APB1 クロック・サイクル後)

DAC チャネル 2 トリガが発生すると、DHR2 レジスタは DAC_DOR2 へ転送されます。
(3APB1 クロック・サイクル後)

11.4.2 同じ 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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに異なる値をセットし、異なるトリガソースを設定
- 2 つの DAC チャネル WAVEEx[1 : 0] ビットを 01 に、同じ LFSR マスクを MAMPx[3 : 0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

DAC チャネル 1 トリガが発生すると、同じマスクで LFSR1 カウンタが DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR1 カウンタがアップデートされます。

DAC チャネル 2 トリガが発生すると、同じマスクで LFSR2 カウンタが DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR2 カウンタがアップデートされます。

11.4.3 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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに異なる値をセットし、異なるトリガソースを設定
- 2つの DAC チャネル WAVE_x[1:0] ビットを 01 に、異なる LFSR マスクを MAMP1[3:0] と MAMP2[3 : 0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

DAC チャネル 1 トリガが発生すると、MAMP1[3 : 0] により設定されたマスクで LFSR1 カウンタが DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR1 カウンタがアップデートされます。

DAC チャネル 2 トリガが発生すると、MAMP2[3 : 0] により設定されたマスクで LFSR2 カウンタが DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR2 カウンタがアップデートされます。

11.4.4 同じ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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに異なる値をセットし、異なるトリガソースを設定
- 2つの DAC チャネル WAVE_x[1:0] ビットを 1x に、同じ最大振幅値を MAMP_x[3:0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

DAC チャネル 1 トリガが発生すると、DAC チャネル 1 三角カウンタが同じ三角振幅で DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 1 三角カウンタがアップデートされます。

DAC チャネル 2 トリガが発生すると、DAC チャネル 2 三角カウンタが同じ三角振幅で DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 2 三角カウンタがアップデートされます。

11.4.5 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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに異なる値をセットし、異なるトリガソースを設定
- 2つの DAC チャネル WAVE_x[1:0] ビットを 1x に、異なる最大振幅値マスクを MAMP1[3:0] と MAMP2[3 : 0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

DAC チャネル 1 トリガが発生すると、MAMP1[3 : 0] により設定されたマスクで DAC チャネル 1 三角カウンタが DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 1 三角カウンタがアップデートされます。

DAC チャネル 2 トリガが発生すると、MAMP2[3 : 0] により設定されたマスクで DAC チャネル 2 三角カウンタが DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 2 三角カウンタがアップデートされます。

11.4.6 同時ソフトウェアスタート

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。

- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

このコンフィグレーションでは、DHR1 と DHR2 レジスタは、1APB1 クロック・サイクル後に DAC_DOR1 と DAC_DOR2 へそれぞれ転送されます。

11.4.7 波形生成なしによる同時トリガ

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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに同じ値をセットし、同じトリガソースを設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

トリガが発生すると、DHR1 と DHR2 レジスタは、DAC_DOR1 と DAC_DOR2 へそれぞれ転送されます。(3APB1 クロック・サイクル後)

11.4.8 同じ 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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに同じ値をセットし、同じトリガソースを設定
- 2つの DAC チャネル WAVE_x[1 : 0] ビットを 01 に、同じ LFSR マスクを MAMP_x[3 : 0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

トリガが発生すると、同じマスクで LFSR1 カウンタが DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR1 カウンタがアップデートされます。同時に、同じマスクで LFSR2 カウンタが DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR2 カウンタがアップデートされます。

11.4.9 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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに同じ値をセットし、同じトリガソースを設定
- 2つの DAC チャネル WAVE_x[1:0] ビットを 01 に、異なる LFSR マスクを MAMP1[3:0] と MAMP2[3 : 0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

トリガが発生すると、MAMP1[3 : 0] で設定されたマスクで LFSR1 カウンタが DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR1 カウンタがアップデートされます。

同時に、MAMP2[3 : 0] で設定されたマスクで LFSR2 カウンタが DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 LFSR2 カウンタがアップデートされます。

11.4.10 同じ三角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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに同じ値をセットし、同じトリガソースを設定
- 2つの DAC チャネル WAVE_x[1:0] ビットを 1x に、同じ最大振幅値を MAMP_x[3:0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

DAC チャネル 1 トリガが発生すると、DAC チャネル 1 三角カウンタが同じ三角振幅で DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 1 三角カウンタがアップデートされます。

同時に、DAC チャネル 2 三角カウンタが同じ三角振幅で DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 2 三角カウンタがアップデートされます。

11.4.11 異なる三角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 チャネル・トリガ・イネーブル・ビットをセット
- TSEL1[2:0] と TSEL2[2 : 0] ビットに同じ値をセットし、同じトリガソースを設定
- 2 つの DAC チャネル WAVEx[1:0] ビットを 1x に、異なる最大振幅値マスクを MAMP1[3:0] と MAMP2[3 : 0] に設定
- 希望する DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD または DAC_DHR8RD) へ、デュアル DAC チャネル・データをロード

トリガが発生すると、MAMP1[3 : 0] で設定された三角振幅で DAC チャネル 1 三角カウンタが DHR1 レジスタに加算され、合計は DAC_DOR1 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 1 三角カウンタがアップデートされます。同時に、MAMP2[3 : 0] で設定された三角振幅で DAC チャネル 2 三角カウンタが DHR2 レジスタに加算され、合計は DAC_DOR2 へ転送されます。(3APB1 クロック・サイクル後) その後 DAC チャネル 2 三角カウンタがアップデートされます。

参考資料

11.5 DAC レジスタ

11.5.1 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	DMA EN2	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	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み	DMA EN1	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	rw

Bits 31:29 予約済み

Bit 28 DMAEN2: DAC チャネル 2DMA イネーブル

このビットはソフトウェアによりセット / クリアされます。

0: DAC チャネル 2DMA モードはディセーブルです。

1: DAC チャネル 2DMA モードはイネーブルです。

Bit 27:24 MAMP2[3:0]: DAC チャネル 2 マスク / 振幅セレクタ

これらのビットは、波形生成モードのマスク、または三角生成モードの振幅を選択するために、ソフトウェアにより書込まれます。

0000: LFSR のマスクしないビット 0 / 三角振幅は 1 と等しい

0001: LFSR のマスクしないビット [1:0] / 三角振幅は 3 と等しい

0010: LFSR のマスクしないビット [2:0] / 三角振幅は 7 と等しい

0011: LFSR のマスクしないビット [3:0] / 三角振幅は 15 と等しい

0100: LFSR のマスクしないビット [4:0] / 三角振幅は 31 と等しい

0101: LFSR のマスクしないビット [5:0] / 三角振幅は 63 と等しい

0110: LFSR のマスクしないビット [6:0] / 三角振幅は 127 と等しい

0111: LFSR のマスクしないビット [7:0] / 三角振幅は 255 と等しい

1000: LFSR のマスクしないビット [8:0] / 三角振幅は 511 と等しい

1001: LFSR のマスクしないビット [9:0] / 三角振幅は 1023 と等しい

1010: LFSR のマスクしないビット [10:0] / 三角振幅は 2047 と等しい

1011: LFSR のマスクしないビット [11:0] / 三角振幅は 4095 と等しい

Bit 23:22 WAVE2[1:0]: DAC チャネル 2 ノイズ / 三角波生成イネーブル

これらのビットは、ソフトウェアによりセット / クリアされます。

00: 波形生成はディセーブルです。

01: ノイズ波生成はイネーブルです。

1x: 三角波生成はイネーブルです。

注 : ビット TEN2 = 1 の場合に限り使われます。(DAC チャネル 2 トリガ・イネーブル)

参考資料

RM0008

D/A コンバータ (DAC)

Bits 21:19 TSEL2[2:0]: DAC チャネル 2 トリガ選択

これらのビットは、DAC チャネル 2 をトリガするための外部イベントを選択します。

- 000: タイマ 6 TRGO イベント
- 001: タイマ 8 TRGO イベント
- 010: タイマ 7 TRGO イベント
- 011: タイマ 5 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 の場合に限り使われます。（DAC チャネル 2 トリガ・イネーブル）

Bit 18 TEN2: DAC チャネル 2 トリガ・イネーブル

このビットは、DAC チャネル 2 トリガをイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: DAC チャネル 2 トリガはディセーブルで、DAC_DHRx レジスタから転送されたデータは 1APB1 クロック・サイクル後に DAC_DOR2 レジスタへ転送されます。

1: DAC チャネル 2 トリガはイネーブルで、DAC_DHRx レジスタから転送されたデータは、3APB1 クロック・サイクル後に DAC_DOR2 レジスタへ転送されます。

注：ソフトウェア・トリガがイネーブルな場合、DAC_DHRx から DAC_DOR2 レジスタへの転送には、1APB1 クロック・サイクルしかかかりません。

Bit 17 BOFF2: DAC チャネル 2 出力バッファ・ディセーブル

このビットは、DAC チャネル 2 出力バッファをイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: DAC チャネル 2 出力バッファはイネーブルです。

1: DAC チャネル 2 出力バッファはディセーブルです。

Bit 16 EN2: DAC チャネル 2 イネーブル

このビットは、DAC チャネル 2 をイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: DAC チャネル 2 はディセーブルです。

1: DAC チャネル 2 はイネーブルです。

Bits 15:13 予約済み

Bit 12 DMAEN1: DAC チャネル 1DMA イネーブル

このビットは、ソフトウェアによりセット / リセットされます。

0: DAC チャネル 1DMA モードはディセーブルです。

1: DAC チャネル 1DMA モードはイネーブルです。



参考資料

Bits 11:8 MAMP1[3:0]: DAC チャネル 1 マスク / 振幅セレクタ

これらのビットは、波形生成モードのマスク、または三角生成モードの振幅を選択するために、ソフトウェアにより書き込まれます。

- 0000: LFSR のマスクしないビット 0 / 三角振幅は 1 と等しい
- 0001: LFSR のマスクしないビット [1:0] / 三角振幅は 3 と等しい
- 0010: LFSR のマスクしないビット [2:0] / 三角振幅は 7 と等しい
- 0011: LFSR のマスクしないビット [3:0] / 三角振幅は 15 と等しい
- 0100: LFSR のマスクしないビット [4:0] / 三角振幅は 31 と等しい
- 0101: LFSR のマスクしないビット [5:0] / 三角振幅は 63 と等しい
- 0110: LFSR のマスクしないビット [6:0] / 三角振幅は 127 と等しい
- 0111: LFSR のマスクしないビット [7:0] / 三角振幅は 255 と等しい
- 1000: LFSR のマスクしないビット [8:0] / 三角振幅は 511 と等しい
- 1001: LFSR のマスクしないビット [9:0] / 三角振幅は 1023 と等しい
- 1010: LFSR のマスクしないビット [10:0] / 三角振幅は 2047 と等しい
- 1011: LFSR のマスクしないビット [11:0] / 三角振幅は 4095 と等しい

Bits 7:6 WAVE1[1:0]: DAC チャネル 1 ノイズ / 三角波生成イネーブル

これらのビットはソ、フトウェアによりセット / クリアされます。

- 00: 波形生成はディセーブルです。
- 01: ノイズ波生成はイネーブルです。
- 1x: 三角波生成はイネーブルです。

注 : ビット TEN1 = 1 の場合に限り使われます。(DAC チャネル 1 トリガ・イネーブル)

Bits 5:3 TSEL1[2:0]: DAC チャネル 1 トリガ選択

これらのビットは、DAC チャネル 1 をトリガするための外部イベントを選択します。

- 000: タイマ 6 TRGO イベント
- 001: タイマ 8 TRGO イベント
- 010: タイマ 7 TRGO イベント
- 011: タイマ 5 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 の場合に限り使われます。(DAC チャネル 1 トリガ・イネーブル)

参考資料

Bit 2 **TEN1**: DAC チャネル 1 トリガ・イネーブル

このビットは、DAC チャネル 1 トリガをイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: DAC チャネル 1 トリガはディセーブルで、DAC_DHRx レジスタから転送されたデータは 1APB1 クロック・サイクル後に DAC_DOR1 レジスタへ転送されます。

1: DAC チャネル 1 トリガはイネーブルで、DAC_DHRx レジスタから転送されたデータは、3APB1 クロック・サイクル後に DAC_DOR1 レジスタへ転送されます。

注：ソフトウェア・トリガがイネーブルな場合、DAC_DHRx から DAC_DOR1 レジスタへの転送には、1APB1 クロック・サイクルしかかかりません。

Bit 1 **BOFF1**: DAC チャネル 1 出力バッファ・ディセーブル

このビットは、DAC チャネル 1 出力バッファをイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: DAC チャネル 1 出力バッファはイネーブルです。

1: DAC チャネル 1 出力バッファはディセーブルです。

Bit 0 **EN1**: DAC チャネル 1 イネーブル

このビットは、DAC チャネル 1 をイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: DAC チャネル 1 はディセーブルです。

1: DAC チャネル 1 はイネーブルです。

参考資料

D/A コンバータ (DAC)

RM0008

11.5.2 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															

Bits 31:2 予約済み

Bit 1 **SWTRIG2**: DAC チャネル2 ソフトウェア・トリガ

このビットは、ソフトウェア・トリガをイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: ソフトウェア・トリガはディセーブルです。

1: ソフトウェア・トリガはイネーブルです。

注：このビットは DAC_DHR2 レジスタ値が DAC_DOR2 レジスタへロードされると、ハードウェアによりリセットされます。(1APB1 クロック・サイクル後)

Bit 0 **SWTRIG1**: DAC チャネル1 ソフトウェア・トリガ

このビットは、ソフトウェア・トリガをイネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

0: ソフトウェア・トリガはディセーブルです。

1: ソフトウェア・トリガはイネーブルです。

注：このビットは DAC_DHR1 レジスタ値が DAC_DOR1 レジスタへロードされると、ハードウェアによりリセットされます。(1APB1 クロック・サイクル後)

11.5.3 DAC チャネル1 12bit 右寄せ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]													

Bits 31:12 予約済み

Bit 11:0 **DACC1DHR[11:0]**: DAC チャネル1 12bit 右寄せデータ

これらのビットは、DAC チャネル1 用の 12bit データを示すソフトウェアにより書き込まれます。

参考資料

RM0008

D/A コンバータ (DAC)

11.5.4 DAC チャネル 1 12bit 左寄せ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			

Bits 31:16 予約済み

Bit 15:4 DACC1DHR[11:0]: DAC チャネル 1 12bit 左寄せデータ

これらのビットは、DAC チャネル 1 用の 12bit データを示すソフトウェアにより書込まれます。

Bits 3:0 予約済み

11.5.5 DAC チャネル 1 8bit 右寄せ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							

Bits 31:8 予約済み

Bits 7:0 DACC1DHR[7:0]: DAC チャネル 1 8bit 右寄せデータ

これらのビットは、DAC チャネル 1 用の 8bit データを示すソフトウェアにより書込まれます。



参考資料

11.5.6 DAC チャネル 2 12bit 右寄せ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			

Bits 31:12 予約済み

Bits 11:0 DACC2DHR[11:0]: DAC チャネル 2 12bit 右寄せデータ

これらのビットは、DAC チャネル 2 用の 12bit データを示すソフトウェアにより書込まれます。

11.5.7 DAC チャネル 2 12bit 左寄せ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			

Bits 31:16 予約済み

Bits 15:4 DACC2DHR[11:0]: DAC チャネル 2 12bit 左寄せデータ

これらのビットは、DAC チャネル 2 用の 12bit データを示すソフトウェアにより書込まれます。

Bits 3:0 予約済み

参考資料

11.5.8 DAC チャネル 2 8bit 右寄せ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

Bits 31:8 予約済み

Bits 7:0 DACC2DHR[7:0]: DAC チャネル 2 8bit 右寄せデータ

これらのビットは、DAC チャネル 2 用の 8bit データを示すソフトウェアにより書き込まれます。

11.5.9 デュアル DAC 12bit 右寄せ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]											
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

Bits 31:28 予約済み

Bits 27:16 DACC2DHR[11:0]: DAC チャネル 2 12bit 右寄せデータ

これらのビットは、DAC チャネル 2 用の 12bit データを示すソフトウェアにより書き込まれます。

Bits 15:12 予約済み

Bits 11:0 DACC1DHR[11:0]: DAC チャネル 1 12bit 右寄せデータ

これらのビットは、DAC チャネル 1 用の 12bit データを示すソフトウェアにより書き込まれます。

参考資料

11.5.10 デュアル DAC 12bit 左寄せ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	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	rw	rw	rw	rw

Bits 31:20 DACC2DHR[11:0]: DAC チャネル 2 12bit 左寄せデータ

これらのビットは、DAC チャネル 2 用の 12bit データを示すソフトウェアにより書込まれます。

Bits 19:16 予約済み

Bits 15:4 DACC1DHR[11:0]: DAC チャネル 1 12bit 左寄せデータ

これらのビットは、DAC チャネル 1 用の 12bit データを示すソフトウェアにより書込まれます。

Bits 3:0 予約済み

11.5.11 デュアル DAC 8bit 右寄せ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

Bits 31:16 予約済み

Bits 15:8 DACC2DHR[7:0]: DAC チャネル 2 8bit 右寄せデータ

これらのビットは、DAC チャネル 2 用の 8bit データを示すソフトウェアにより書込まれます。

Bits 7:0 DACC1DHR[7:0]: DAC チャネル 1 8bit 右寄せデータ

これらのビットは、DAC チャネル 1 用の 8bit データを示すソフトウェアにより書込まれます。

参考資料

RM0008

D/A コンバータ (DAC)

11.5.12 DAC チャネル 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	r	r	r	r

Bits 31:12 予約済み

Bits 11:0 DACC1DOR[11:0]: DAC チャネル 1 データ出力

これらのビットは、読み取り専用で DAC チャネル 1 のデータ出力を保持します。

11.5.13 DAC チャネル 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	r	r	r	r

Bits 31:12 予約済み

Bits 11:0 DACC2DOR[11:0]: DAC チャネル 2 データ出力

これらのビットは、読み取り専用で DAC チャネル 2 のデータ出力を保持します。



参考資料

11.5.14 DAC レジスタ・マップ

次の表は DAC レジスタの一覧です。

表 51. DAC レジスタ・マップ

アドレスオフセット	名称	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		DMAEN2	MAMP2[3:0]	WAV E2[2:0]	TSEL2[2:0]	TEN2	BOFF2	EN2	予約済み	DMAEN1	MAMP1[3:0]	WAV E1[2:0]	TSEL1[2:0]	TEN1	SWTRIG2	BOFF1															
0x04	DAC_SWTRI GR	予約済み																									SWTRIG1		EN1	0			
0x08	DAC_DHR12 R1	予約済み												DACC1DHR[11:0]								DACC1DHR[11:0]								予約済み			
0x0C	DAC_DHR12 L1	予約済み												DACC1DHR[11:0]								DACC1DHR[7:0]								予約済み			
0x10	DAC_DHR8R 1	予約済み												DACC1DHR[7:0]								DACC2DHR[11:0]								DACC2DHR[7:0]			
0x14	DAC_DHR12 R2	予約済み												DACC2DHR[11:0]								DACC2DHR[7:0]								予約済み			
0x18	DAC_DHR12 L2	予約済み												DACC2DHR[7:0]								DACC1DHR[11:0]								予約済み			
0x1C	DAC_DHR8R 2	予約済み												DACC2DHR[7:0]								DACC1DHR[11:0]								DACC1DHR[11:0]			
0x20	DAC_DHR12 RD	予約済み	DACC2DHR[11:0]										予約済み	DACC1DHR[11:0]								DACC1DHR[7:0]								予約済み			
0x24	DAC_DHR12 LD	DACC2DHR[11:0]										予約済み	DACC1DHR[11:0]								DACC1DHR[7:0]								予約済み				
0x28	DAC_DHR8R D	予約済み												DACC2DHR[7:0]								DACC1DHR[11:0]								DACC1DHR[7:0]			
0x2C	DAC_DOR1	予約済み												DACC1DOR[11:0]								DACC2DOR[11:0]								DACC2DOR[11:0]			
0x30	DAC_DOR2	予約済み												DACC2DOR[11:0]								DACC1DOR[11:0]								DACC1DOR[7:0]			

注意：レジスタのアドレス範囲に関しては、[表 1 \(36 ページ\)](#) を参照してください。

12 高機能制御タイマ (TIM1 及び TIM8)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、中容量デバイスです。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、大容量デバイスです。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適用されます。

12.1 TIM1 及び TIM8 の概要

中容量デバイスは TIM1 を持ち、大容量デバイスは TIM1 と TIM8 の両方を持っています。

高性能制御タイマ (TIM1 及び TIM8) は、プログラマブル・プリスケーラによる 16bit 自動再ロード・カウンタを持っています。

このタイマは、入力信号のパルス幅の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、様々な目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロック・コントローラ・プリスケーラを使用して、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。

高機能タイマ (TIM1 及び TIM8) と汎用タイマ (TIMx) は互いに独立しており、リソースの一部を共有していることはありません。また [セクション 12.3.20](#) に記されているように、相互に同期させることもできます。

参考資料

12.2 TIM1 及び TIM8 の主な機能

TIM1 及び TIM8 タイマの主な機能は：

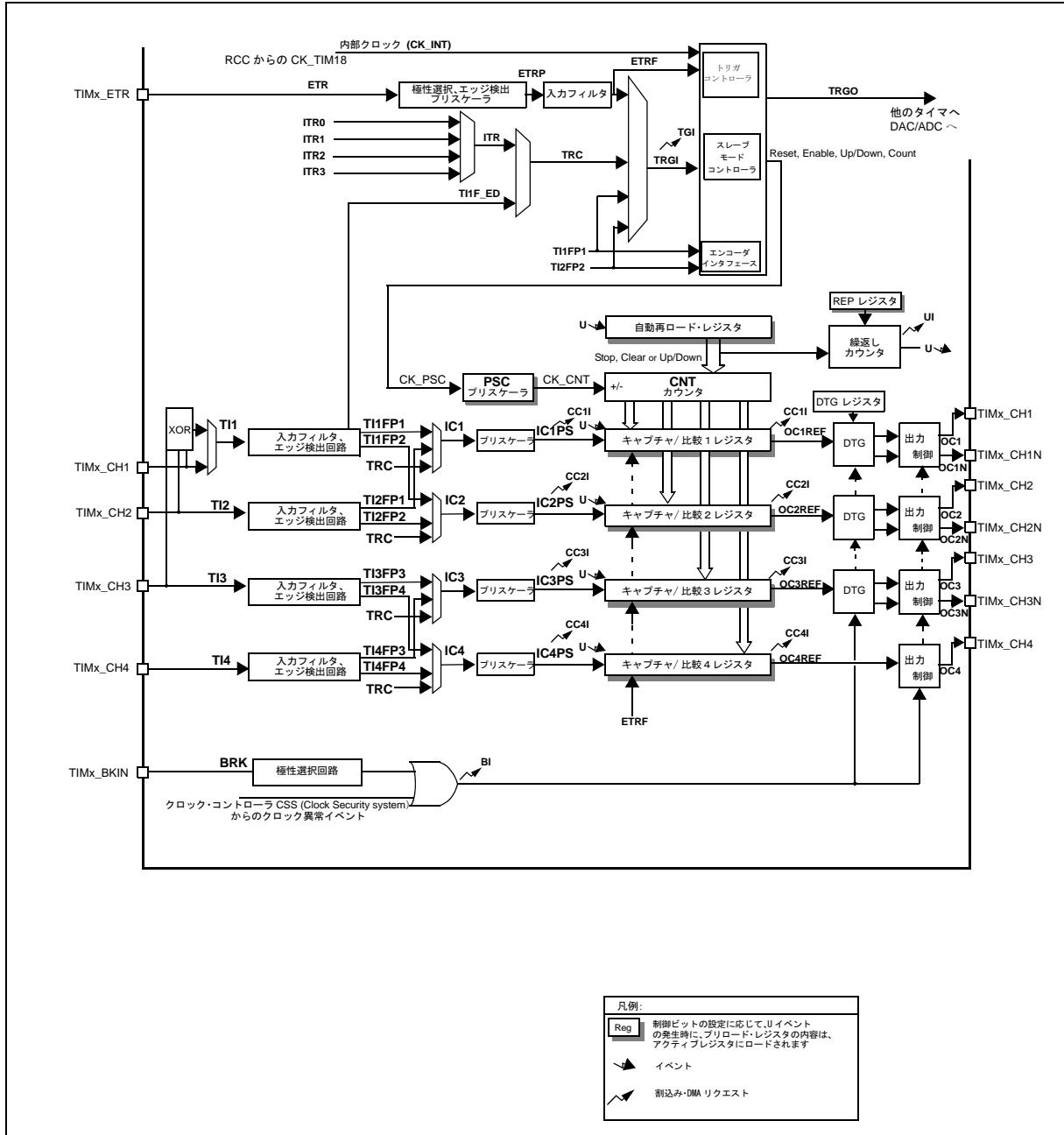
- 16bit アップ、ダウン自動再ロード・カウンタ
- 16bit プログラマブル・プリスケーラ。カウンタ・クロック周波数を 1 から 65535 の間で分周でき、分周比を動作中に変更することもできます。
- 次の機能を持つ、独立した最大 4 チャネル。
 - 入力キャプチャ
 - 出力比較
 - PWM 生成（エッジアライン・モードとセンターアライン・モード）
 - 単パルスモード出力
- 設定可能なデッドタイムを有する相補出力
- 外部信号でタイマを制御する同期回路と複数タイマ間の相互接続。
- 所定の回数だけタイマレジスタを更新する繰返しカウンタ。
- タイマの出力信号をリセット状態、もしくは指定した状態とするブレーク入力。
- 以下のイベントによる、割込み /DMA リクエストの発生。
 - 更新：カウンタ・オーバフロー / アンダフロー、カウンタの初期化（ソフトウェア指示、及び内部 / 外部トリガ）
 - トリガ・イベント（カウンタ・スタート、ストップ、初期化、及び内部 / 外部トリガによるカウント動作）
 - 入力キャプチャ
 - 出力比較
 - ブレーク入力

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

図 47. 高機能制御タイマ・ブロック図



参考資料

12.3 TIM1 及び TIM8 機能詳細

12.3.1 タイムベース・ユニット

プログラマブル高機能制御タイマの主要なブロックは自動再ロード・レジスタを持った 16bit カウンタです。カウンタはカウントアップ、カウントダウン、及びアップ・ダウンの動作が可能です。カウンタのクロックはプリスケーラによって分周されます。

カウンタ、自動再ロード・レジスタ、及びプリスケーラ・レジスタは、カウンタが動作中であっても、ソフトウェアによる読み書きができます。

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)
- プリスケーラ・レジスタ (TIMx_PSC)
- 自動再ロード・レジスタ (TIMx_ARR)
- 繰返しカウンタ・レジスタ (TIMx_RCR)

自動再ロード・レジスタはプリロード機能を持っています。自動再ロード・レジスタの読み書きはプリロード・レジスタへのアクセスとなります。プリロード・レジスタの内容が、TIMx_CR1 レジスタのプリロード・イネーブル・ビット (ARPE) の設定に応じて、常時もしくは更新イベント 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 であれば、更新イベントはカウンタがオーバフロー (ダウンカウント時はアンダフロー) に達したときに発生します。また、ソフトウェアで更新イベントを発生させることも可能です。この方法は、それぞれのタイマ構成のセクションに記されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタ・イネーブル・ビット (CEN) がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます。(カウンタをイネーブルとする方法の詳細に関しては、スレーブ・モード・コントローラの記述を合わせてお読みください。)

なお、実際のカウンタ・イネーブル信号 CNT_EN は CEN ビットがセットされてから 1 クロック遅れてセットされることに留意してください。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

プリスケーラ

プリスケーラはカウンタ・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。プリスケーラは 16bit レジスタ (TIMx_PSC) を通して制御される 16bit カウンタをベースとしています。制御レジスタにはバッファが備わっているので、プリスケーラが動作中であっても、レジスタ内容を変更することができます。新たに指定された分周比は次の更新イベントで効果があります。

図 49 と図 50 は動作中に分周比を変化させたときのカウンタの動作の例を示しています。

図 48. プリスケーラ分周比が 1 から 2 に変化した時のカウンタ・タイミング図

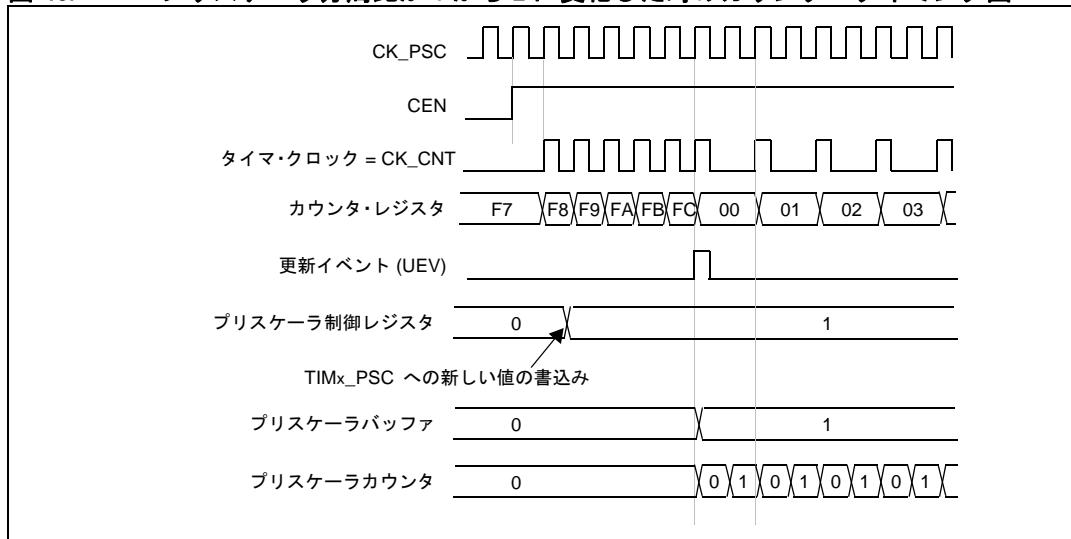
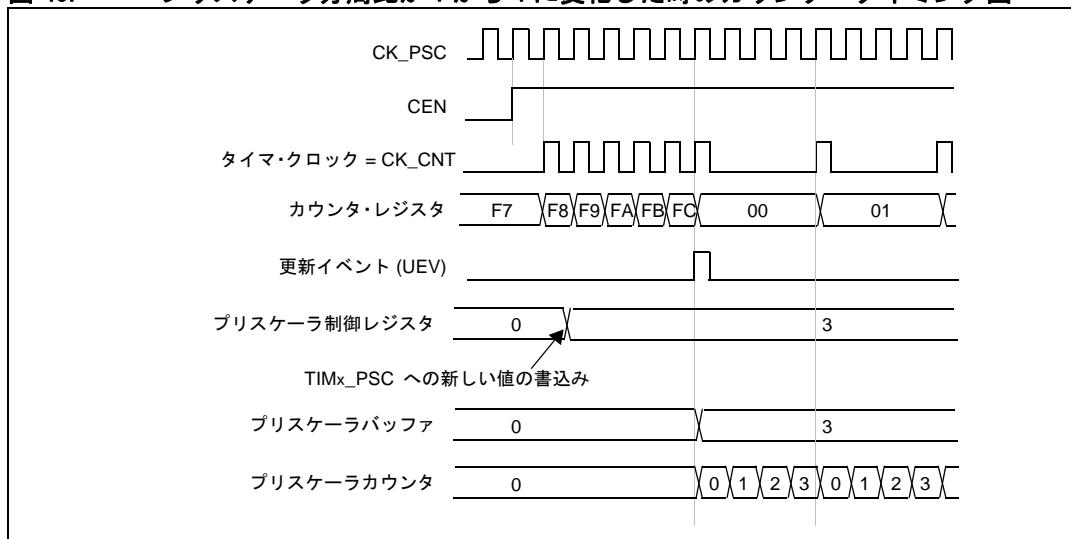


図 49. プリスケーラ分周比が 1 から 4 に変化した時のカウンタ・タイミング図



参考資料

12.3.2 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・レジスタの値 (TIMx_ARR レジスタの内容) までカウントを行い、その後 0 からのカウントを再開し、同時にカウンタ・オーバフロー・イベントを発生させます。

繰返しカウンタが使用されている場合には、繰返しカウンタ・レジスタ (TIMx_RCR) に設定されている回数までアップカウント動作が繰返され、その後に更新イベント (UEV) が発生します。繰返しカウンタが使用されていないときには、カウンタのオーバフローごとに更新イベントが発生します。

ソフトウェアによるレジスタ書き込みで、もしくはスレーブ・モード・コントローラによって TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントを発生させることができます。

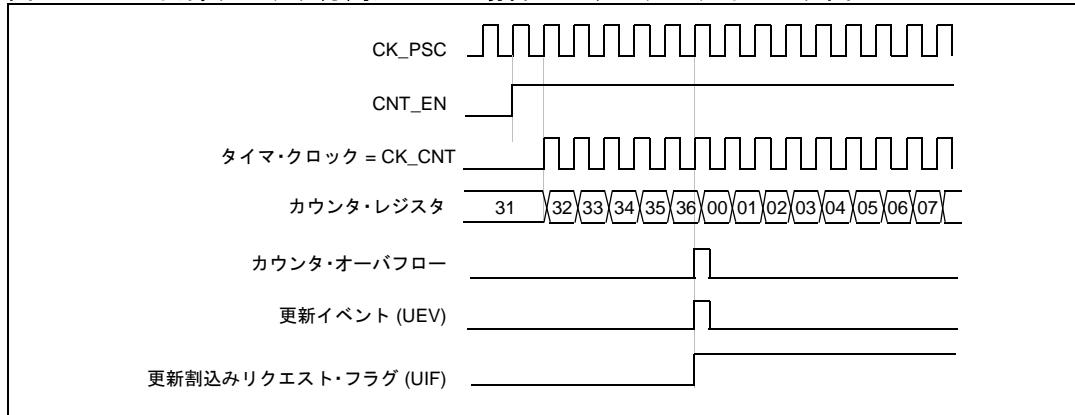
UEV イベントはソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることで、ディセーブルすることができます。これはプリロード・レジスタに新しい値を書込んでいるときにシャドウ・レジスタが更新されるのを防止します。この後 UDIS ビットに 0 が書込まれるまで、更新イベントは発生しません。しかし、プリスケーラのカウンタと同じく（ただしプリスケーラの比は変化しません）、カウンタは 0 からカウントを再開します。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットのセットが更新イベント UEV を引起こしますが、UIF フラグはセットされず、割込みや DMA リクエストは発生しません。これは、キャプチャイベントの際にカウンタをクリアすることで、更新とキャプチャ割込みがともに発生することを防ぎます。

更新イベントが発生したとき、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。

- 繰返しカウンタには TIMx_RCR レジスタの内容が再びロードされます。
- 自動再ロード・シャドウ・レジスタはプリロード・レジスタ (TIMx_ARR) の値に更新されます。
- プリスケーラのバッファにはプリロード値 (TIMx_PSC レジスタの値) が再びロードされます。

以下の図は、自動再ロード・レジスタ TIMx_ARR = 0x36 の場合の、異なるクロック周波数におけるカウンタの挙動の例を示しています。

図 50. 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



参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

図 51. 内部クロック分周比が 2 の場合のカウンタ・タイミング図

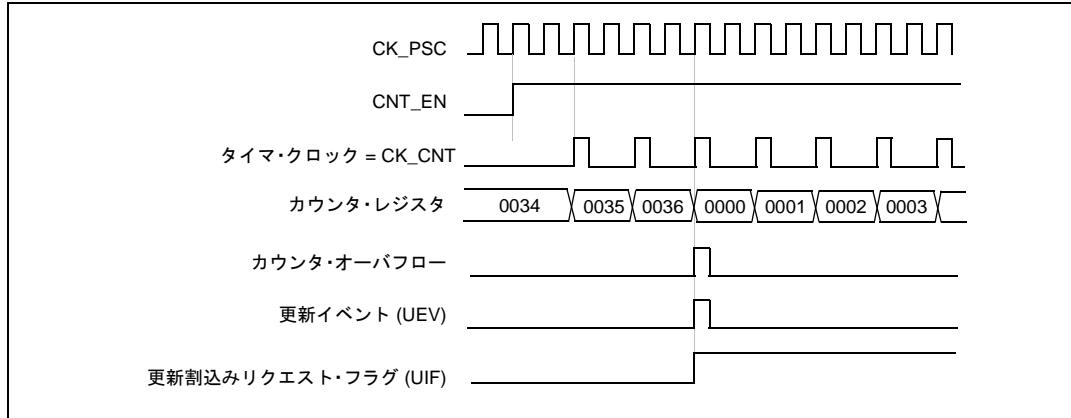


図 52. 内部クロック分周比が 4 の場合のカウンタ・タイミング図

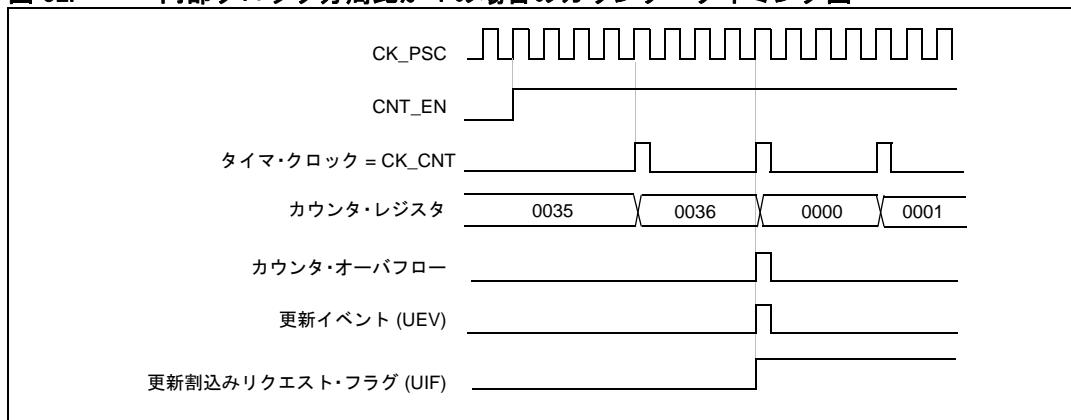
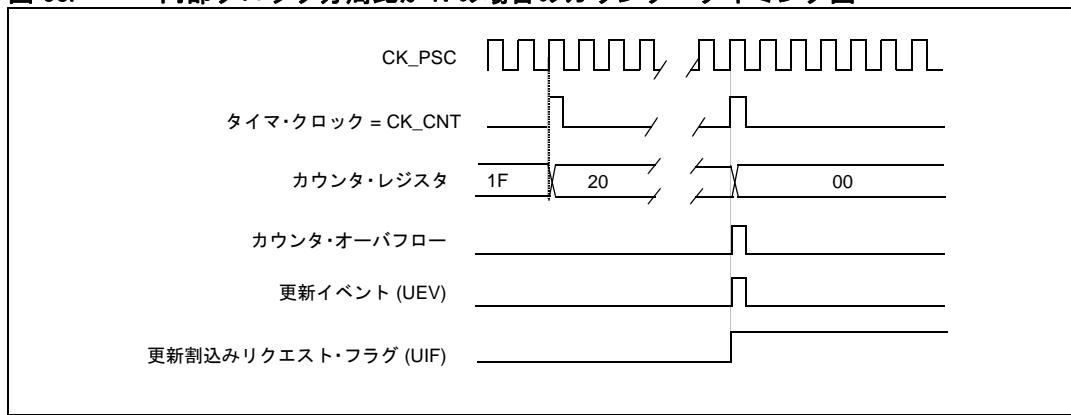


図 53. 内部クロック分周比が N の場合のカウンタ・タイミング図



参考資料

図 54. ARPE=0 時 (TIMx_ARR はプリロードなし)、更新イベント時のタイミング図

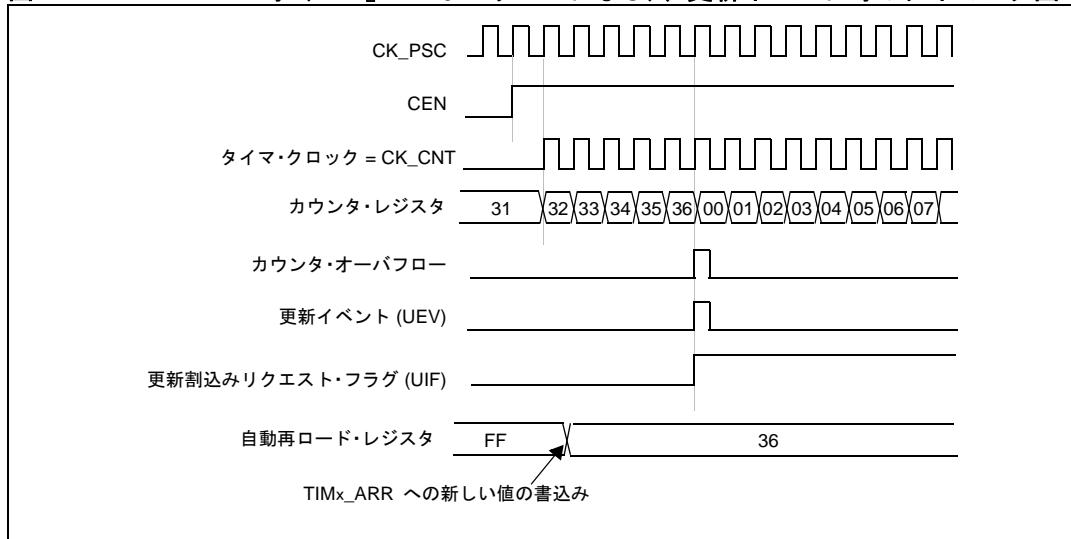
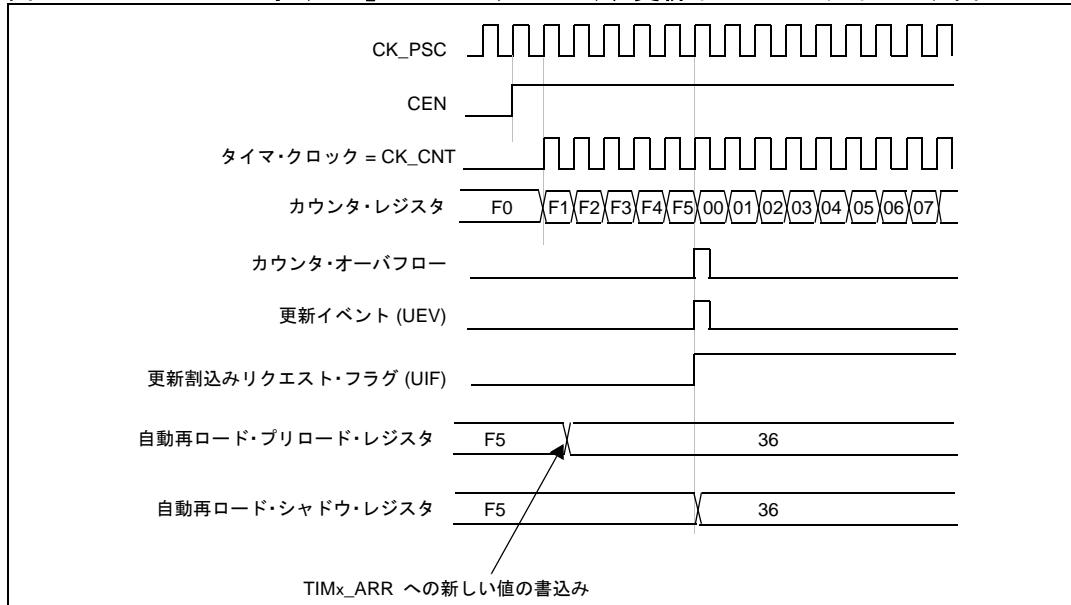


図 55. ARPE=1 時 (TIMx_ARR はプリロード)、更新イベントのタイミング図



参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

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・レジスタの値 (TIMx_ARR レジスタの内容) から 0 までのカウントダウンを行い、その後、自動再ロード・レジスタの値からのカウントダウンを再開し、同時にカウンタ・アンダフロー・イベントを発生させます。

繰返しカウンタが使用されている場合には、繰返しカウンタ・レジスタ (TIMx_RCR) に設定されている回数までダウンカウント動作が繰返され、その後に更新イベント (UEV) が発生します。繰返しカウンタが使用されていないときには、カウンタのアンダフローごとに更新イベントが発生します。

ソフトウェアによるレジスタ書き込みで、もしくはスレーブ・モード・コントローラによって TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントを発生させることができます。

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で、ディセーブルすることができます。これはプリロード・レジスタに新しい値を書込んでいたときにシャドウ・レジスタが更新されるのを防止します。この後 UDIS ビットに 0 が書込まれるまで、更新イベントは発生しません。しかし、プリスクーラのカウンタが 0 からカウントを再開するのに対し（ただしプリスクーラの比は変化しません）、カウンタはその時点の自動再ロード・レジスタの値からカウントダウンを再開します。

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 ビットのセットが更新イベント UEV を引起こしますが、UIF フラグはセットされず、割込みや DMA リクエストは発生しません。これは、キャプチャイベントの際にカウンタをクリアすることで、更新とキャプチャ割込みがともに発生することを防ぎます。

更新イベントが発生したとき、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。

- 繰返しカウンタには TIMx_RCR レジスタの内容が再びロードされます。
- プリスクーラ・バッファにはプリロード値 (TIMx_PSC レジスタの内容) が再びロードされます。
- 自動再ロード・アクティブレジスタはプリロード・レジスタ (TIMx_ARR) の値に更新されます。自動再ロード・レジスタの更新はカウンタの再ロードの前に行われますので、次の周期は希望の時間となります。

以下の図は、自動再ロード・レジスタ TIMx_ARR = 0x36 の場合の、異なるクロック周波数におけるカウンタの挙動の例を示しています。



参考資料

図 56. 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

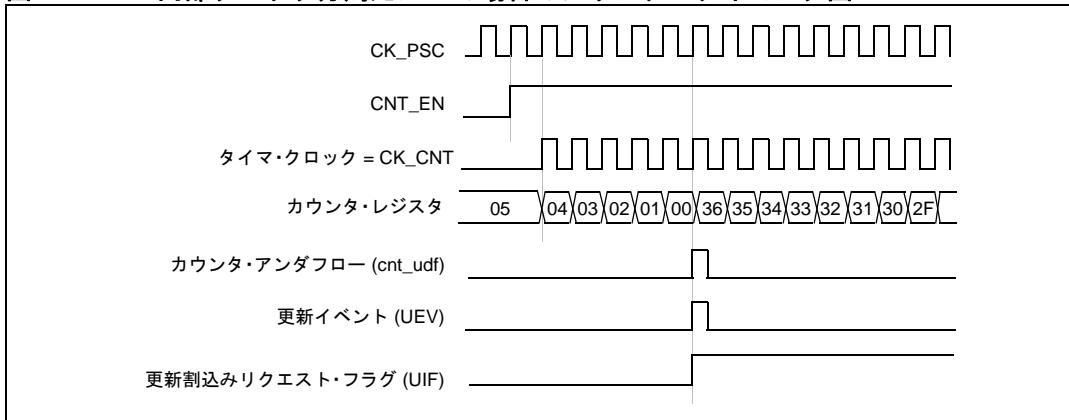


図 57. 内部クロック分周比が 2 の場合のカウンタ・タイミング図

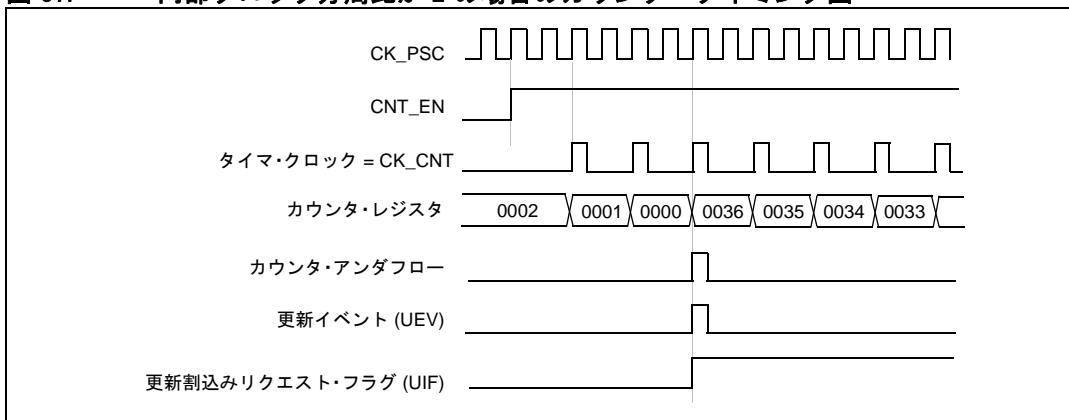
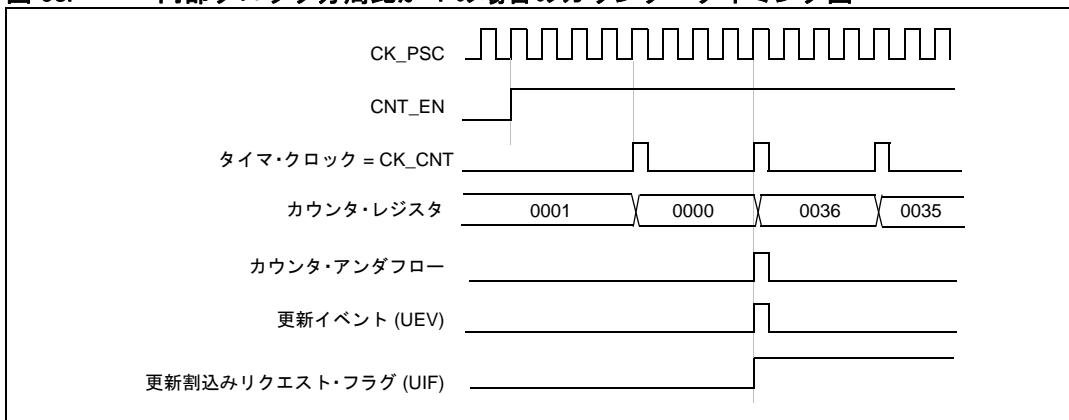


図 58. 内部クロック分周比が 4 の場合のカウンタ・タイミング図



参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

図 59. 内部クロック分周比が N の場合のカウンタ・タイミング図

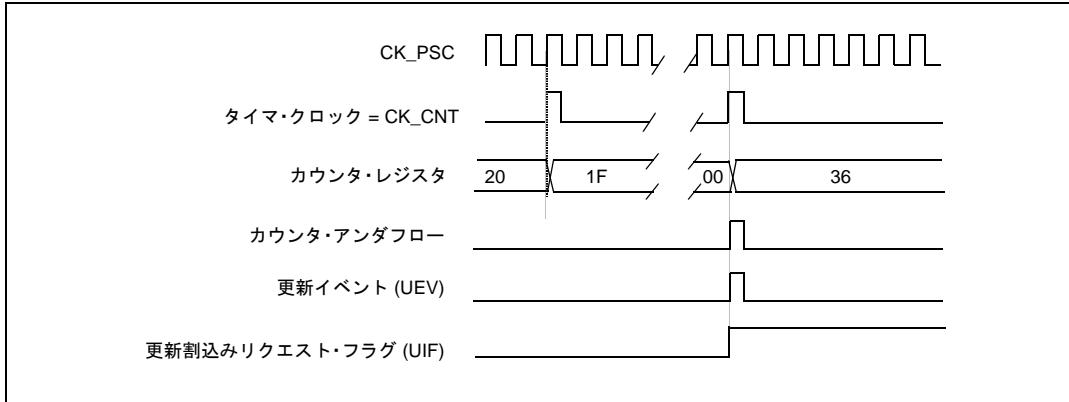
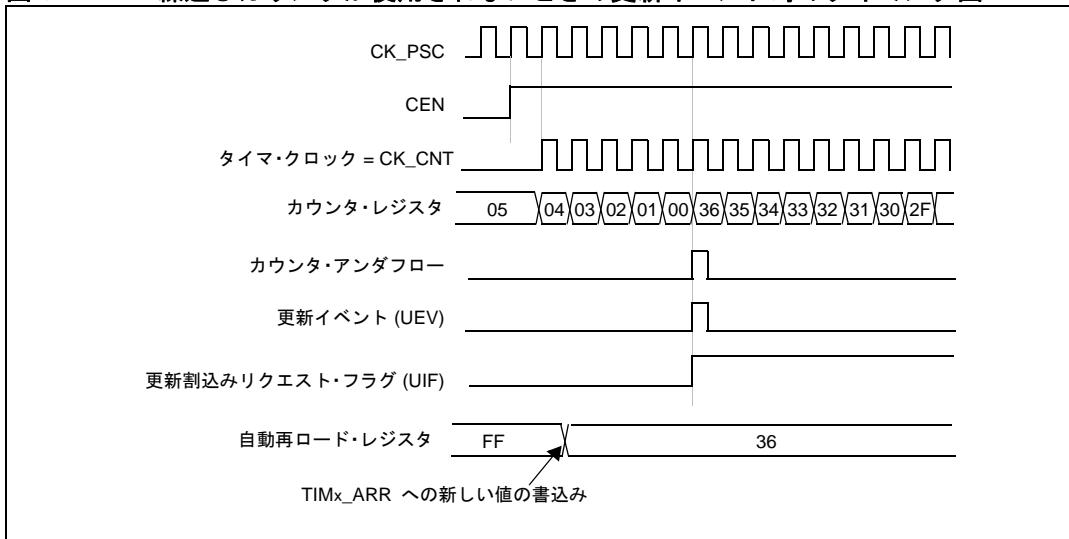


図 60. 繰返しカウンタが使用されないときの更新イベント時のタイミング図



参考資料

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・レジスタの値 (TIMx_ARR レジスタの内容) - 1 までカウントを行い、カウンタ・オーバフロー・イベントを発生させます。次に 1 までのカウントダウンを行い、カウンタ・アンダフロー・イベントを発生させます。その後 0 からのカウントを再開します。

このモードでは TIMx_CR1 レジスタの方向ビット (DIR) への書き込みはできません。このビットはハードウェアによって更新され、その時点でのカウンタのカウント方向を表示します。

更新イベントはカウンタがオーバフロー及びアンダフローを生じるごとに発生します。また、TIMx_EGR レジスタの UG ビットをセットすることで、ソフトウェアで、もしくはスレーブ・モード・コントローラにより、更新イベントを発生させることもできます。このときには、プリスケーラのカウンタと同じく、カウンタは 0 からのカウントアップを再開します。

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で、ディセーブルすることができます。これはプリロード・レジスタに新しい値を書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止します。この後 UDIS ビットに 0 が書込まれるまで、更新イベントは発生しません。しかし、カウンタはその時点の自動再ロード値を使用して、カウント動作（アップ / 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 ビットのセットが更新イベント UEV を引起こしますが、UIF フラグはセットされず、割込みや DMA リクエストは発生しません。これは、キャプチャイベントの際にカウンタをクリアすることで、更新とキャプチャ割込みがともに発生することを防ぎます。

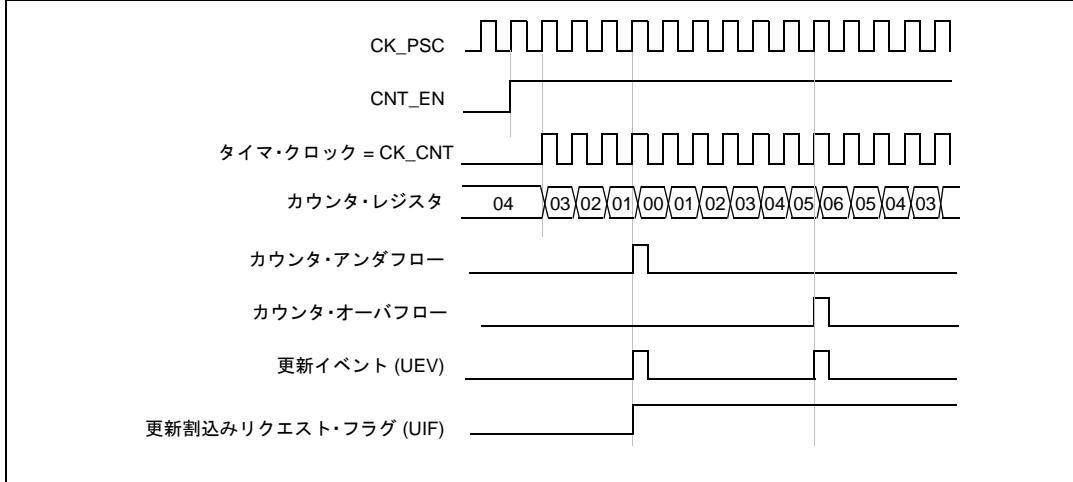
更新イベントが発生したとき、すべてのレジスタが更新され、URS ビットの設定に応じて更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- 繰返しカウンタには TIMx_RCR レジスタの内容が再びロードされます。
- プリスケーラ・バッファにはプリロード値 (TIMx_PSC レジスタの内容) が再びロードされます。
- 自動再ロード・アクティブ・レジスタはプリロード値に更新されます。更新の原因がカウンタのオーバフローであるときには、自動再ロード・レジスタの更新はカウンタの再ロードの前に行われる所以、次の周期は希望の時間となります。

以下の図は異なるクロック周波数におけるカウンタの挙動の例を示しています。

参考資料

図 61. 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 (TIMx_ARR=0x6)



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。(詳細は [セクション 12.4: TIM1 及び TIM8 レジスタ \(259 ページ\)](#) を参照してください)

図 62. 内部クロック分周比が 2 の場合のカウンタ・タイミング図

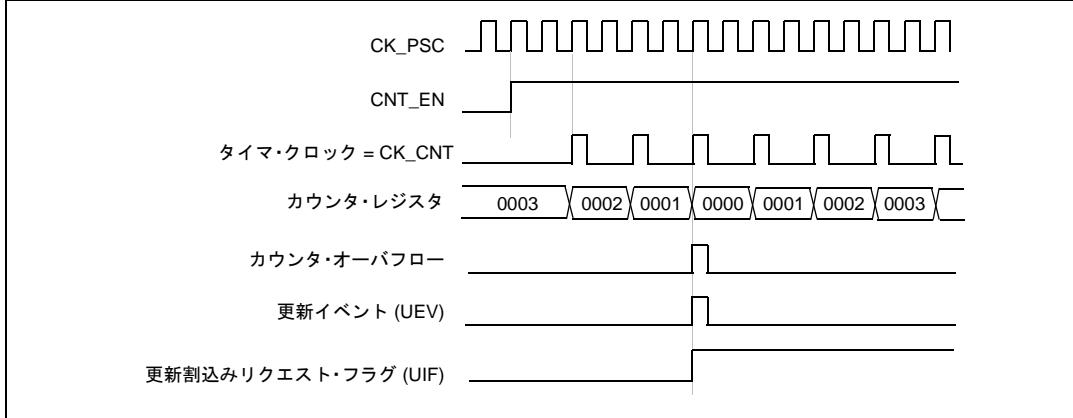
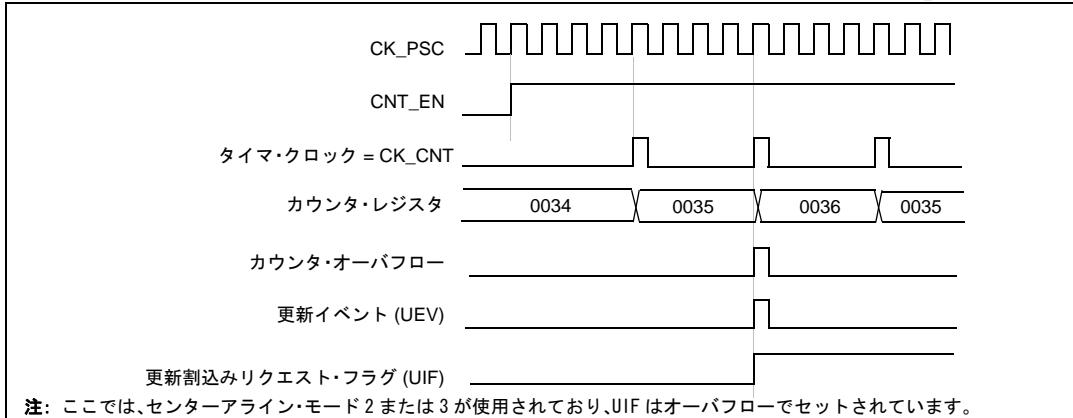


図 63. 内部クロック分周比が 4 の場合のカウンタ・タイミング図 (TIMx_ARR=0x36)



注: ここでは、センターアライン・モード 2 または 3 が使用されており、UIF はオーバフローでセットされています。

参考資料

図 64. 内部クロック分周比が N の場合のカウンタ・タイミング図

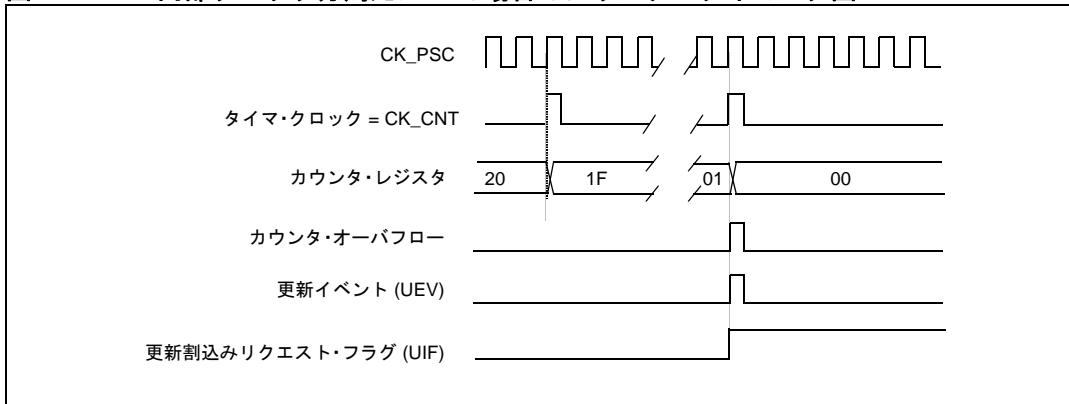
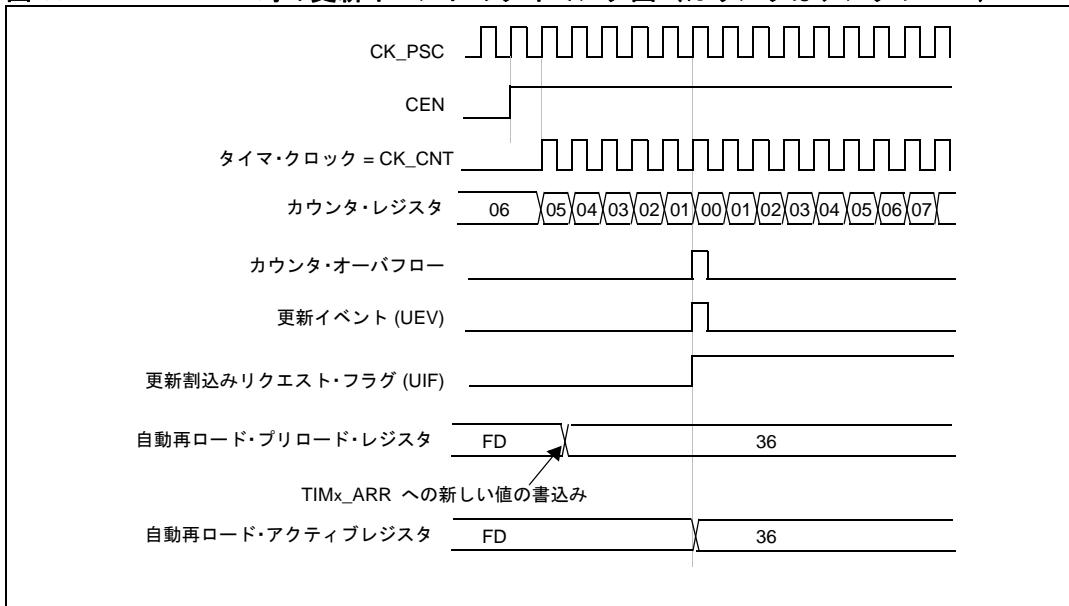
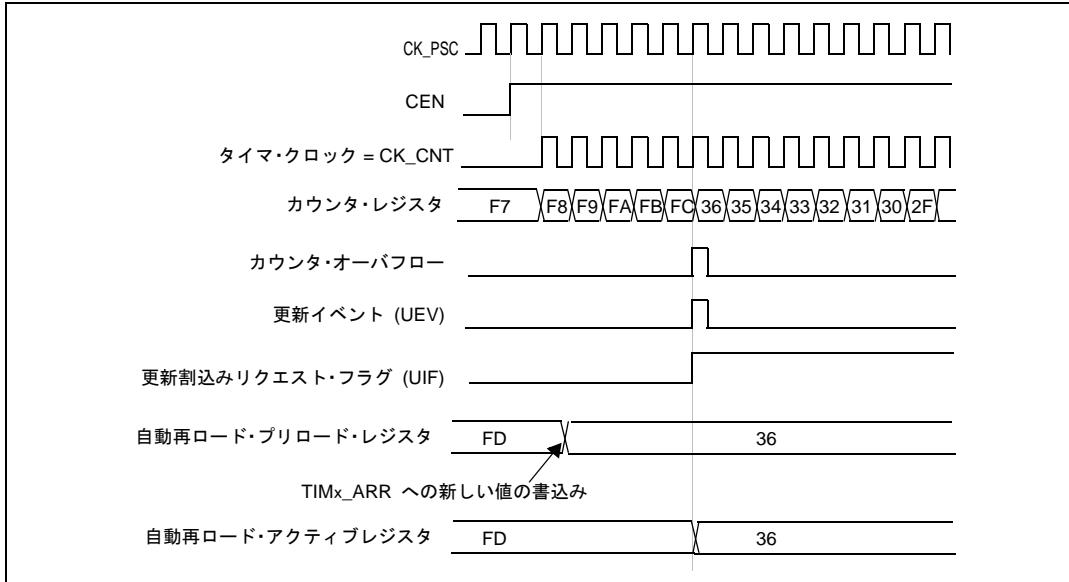


図 65. ARPE=1 時の更新イベントのタイミング図（カウンタはアンダーフロー）



参考資料

図 66. ARPE=1 時の更新イベントのタイミング図 (カウンタはオーバフロー)



12.3.3 繰返しカウンタ (ダウンカウンタ)

セクション 12.3.1: タイムベース・ユニット には、カウンタのオーバフローやアンダフローに伴って、どのように更新イベント (UEV) が発生するかが説明されています。実際には繰返しカウンタが 0 に達したときにのみ更新イベントが発生します。このことは PWM 信号を生成する際に役立ちます。

これは、 $\text{TIM}_x\text{_RCR}$ 繰返しカウンタ・レジスタの値を N とするとき、 N 回目のカウンタのオーバフローもしくはアンダフローごとに、プリロード・レジスタからシャドウ・レジスタにデータが移されることを意味しています。(対象レジスタ : $\text{TIM}_x\text{_ARR}$ 自動再ロード・レジスタ、 $\text{TIM}_x\text{_PSC}$ プリスケーラ・レジスタ、比較モードでの $\text{TIM}_x\text{_CCR}$ キャプチャ / 比較値レジスタ)

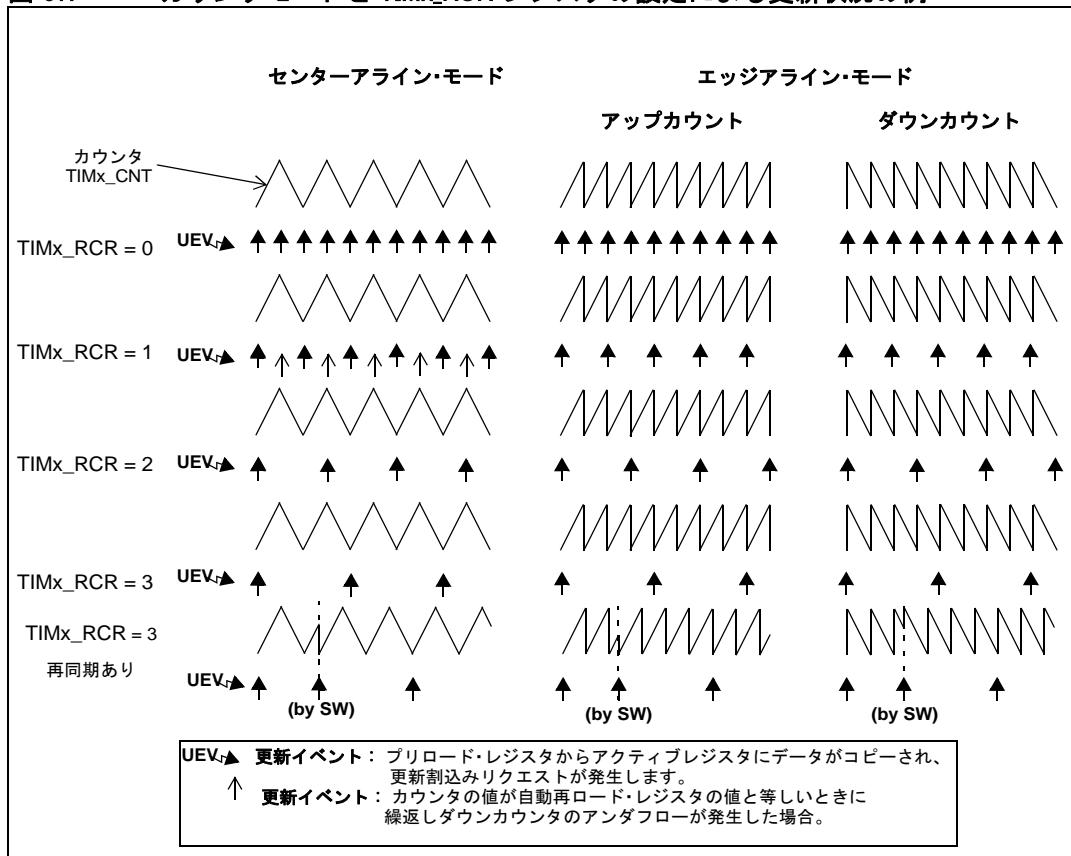
繰返しカウンタのカウントダウンは次の場合に行われます。

- アップカウント・モードで、カウンタ・オーバフローごと
- ダウンカウント・モードで、カウンタ・アンダフローごと
- センターライン・モードで、カウンタ・オーバフローとカウンタ・アンダフローごと
このとき、繰返し回数の上限は 128 PWM サイクルとなります。同時に、デューティサイクルを変更する機会が PWM 周期あたり 2 回あることになります。センターライン・モードで比較値レジスタの値を PWM 周期あたり 1 回のみ更新するときには、波形が対称であるため、最大精度は $2 \times T_{\text{ck}}$ となります。

繰返しダウンカウンタは自動再ロード方式で、繰返しの回数は $\text{TIM}_x\text{_RCR}$ レジスタの値(図 67 を参照)で定まります。ソフトウェアで ($\text{TIM}_x\text{_EGR}$ レジスタの UG ビットをセットすることで) もしくはスレーブ・モード・コントローラを介してハードウェアにより更新イベントが発生させられたとき、そのときの繰返しダウンカウンタの値にかかわらず直ちにイベントが発生し、繰返しダウンカウンタには $\text{TIM}_x\text{_RCR}$ レジスタの内容がロードされます。

参考資料

図 67. カウンタモードと TIMx_RCR レジスタの設定による更新状況の例



12.3.4 クロック選択

カウンタ・クロックは次の信号のいずれかから供給することができます。

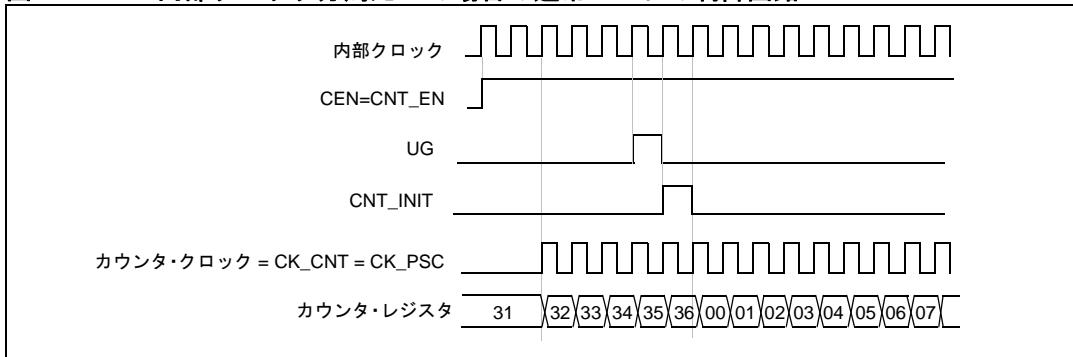
- 内部クロック (CK_INT)
- 外部クロック・モード 1: 外部入力ピン
- 外部クロック・モード 2: 外部トリガ入力 ETR
- 内部トリガ入力 (ITRx): あるタイマを別のタイマのプリスケーラとして使用します。
例えば、タイマ 1 がタイマ 2 のプリスケーラとして働くように設定することができます。詳細は [セクション : タイマを別のタイマのプリスケーラとするには \(319 ページ\)](#) を参照してください。

内部クロック (CK_INT)

スレーブ・モード・コントローラがディセーブル状態 (SMS=000) のときには、TIMx_CR1 レジスタの CEN、DIR ビットと TIMx_EGR レジスタの UG ビットが実際の制御ビットとなります。自動的にクリアされる UG ビット以外は、これらのビットはソフトウェアでのみ変更できます。CEN ビットに 1 が書込まれると、プリスケー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 68](#) には、プリスケー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が図示されています。

図 68. 内部クロック分周比 1 の場合の通常モードの制御回路

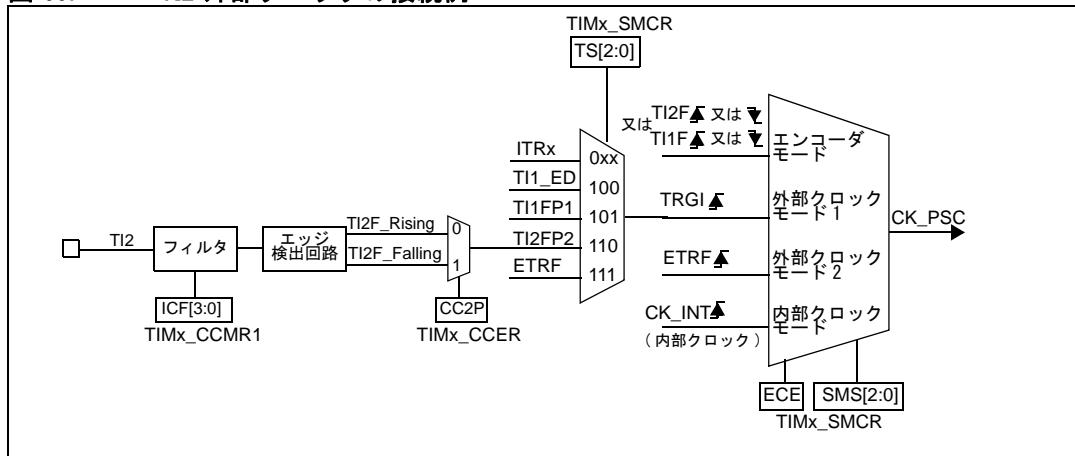


參考資料

外部クロック・モード 1

このモードは TIMx_SMCR レジスタの SMS=111 の時に選択されます。カウンタは選択された入力の立上り、もしくは立下りエッジでカウントを行います。

図 69. TI2 外部クロックの接続例



一例として、アップカウンタが TI2 入力の立上りエッジでカウントを行う構成は、次のように設定します。

1. TIM_x_CCMR1 レジスタの CC2S=01 とすることで、チャネル 2 が TI2 入力の立上りエッジを検出するように設定します。
 2. TIM_x_CCMR1 レジスタの IC2F[3:0] ビットへの書き込みで、入力フィルタの応答時間を設定します。(フィルタを使用しないときには、IC2F=0000 としておきます。)
 3. TIM_x_CCER レジスタの CC2P=0 を書き込み、立上りエッジ極性を選択します。
 4. TIM_x_SMCR レジスタの SMS=111 を書き込み、タイマを外部クロック・モード 1 に設定します。
 5. TIM_x_SMCR レジスタの TS=110 を書き込み、トリガ入力として TI2 を選択します。
 6. TIM_x_CR1 レジスタの CEN=1 を書き込み、カウンタをイネーブルとします。

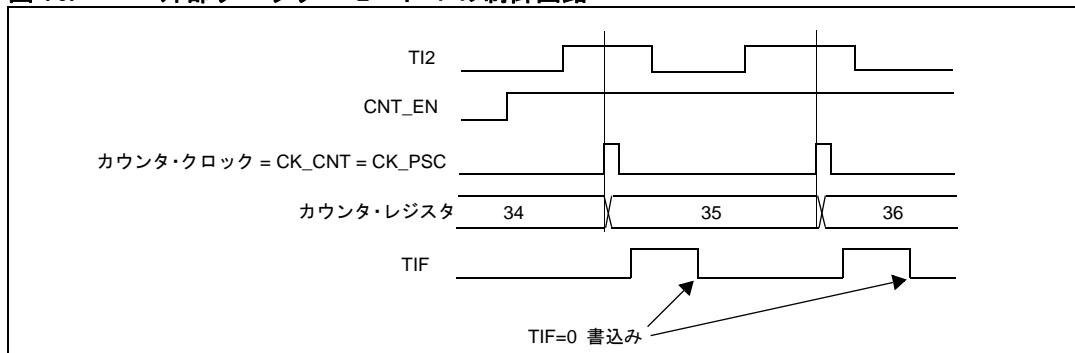
注意：

キャプチャ・プリスケーラはトリガとして使用されませんので、これに関する設定は不要です。

T12 の立上りエッジが生じたとき、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 の立上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起因する遅延が生じます。

図 70. 外部クロック・モード 1 の制御回路



参考資料

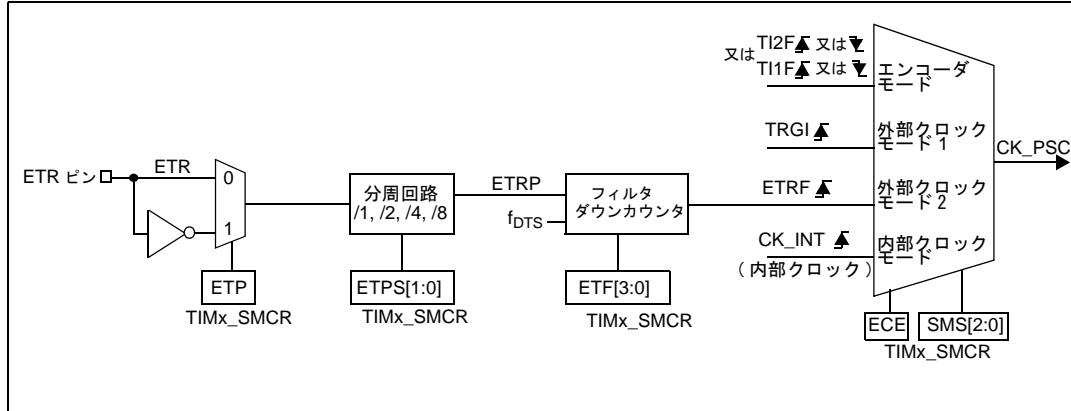
外部クロック・モード 2

このモードは TIM1_SMCR レジスタの ECE=1 とすることで選択されます。

カウンタは外部トリガ入力 ETR の立上りもしくは立下りエッジごとにカウントを行います。

図 71 には外部トリガ入力ブロックの概要が図示されています。

図 71. 外部トリガ入力ブロック



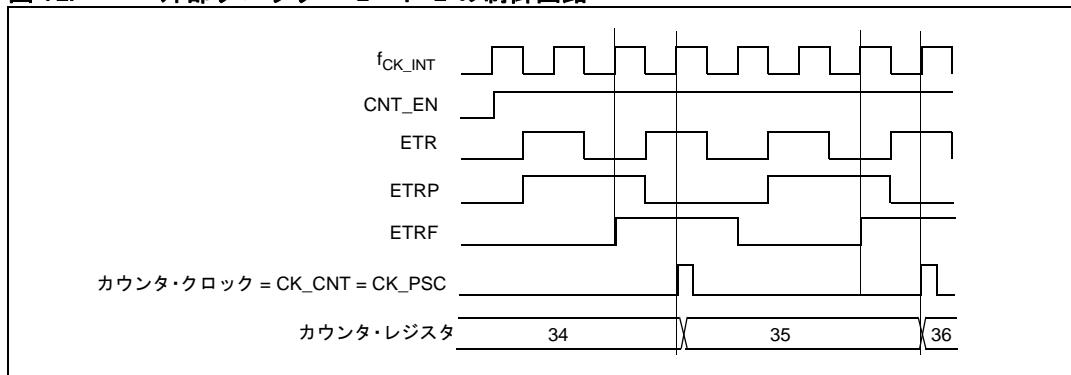
一例として、アップカウンタが ETR の 2 個の立上りエッジごとに 1 回カウントを行う構成は、次のように設定します。

1. この例ではフィルタは必要とされませんので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書込みます。
2. TIMx_SMCR レジスタの ETPS[1:0]=01 とすることで、プリスケーラの設定を行います。
3. TIMx_SMCR レジスタの ETP=0 とし、ETR ピンで立上りエッジを検出するように設定します。
4. TIMx_SMCR レジスタの ECE=1 を書込み、外部クロック・モード 2 を有効とします。
5. TIMx_CR1 レジスタの CEN=1 を書込み、カウンタをイネーブルとします。

カウンタは 2 個の立上りエッジごとに 1 回カウント動作を行います。

ETR の立上りエッジから実際のカウンタ・クロックまでの間には、ETRP 入力の再同期回路に起因する遅延が生じます。

図 72. 外部クロック・モード 2 の制御回路



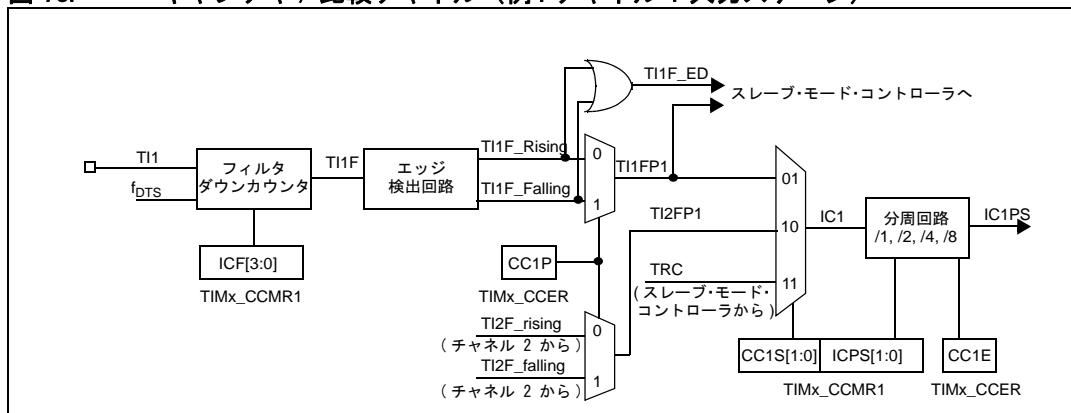
12.3.5 キャプチャ / 比較チャネル

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）、入力ステージ（デジタル・フィルタ、マルチプレクサ、プリスケーラ）、と出力ステージ（比較回路、出力制御）から構成されています。

図 73 から図 76 には、キャプチャ / 比較チャネル（1 チャネル）の概要が図示されています。

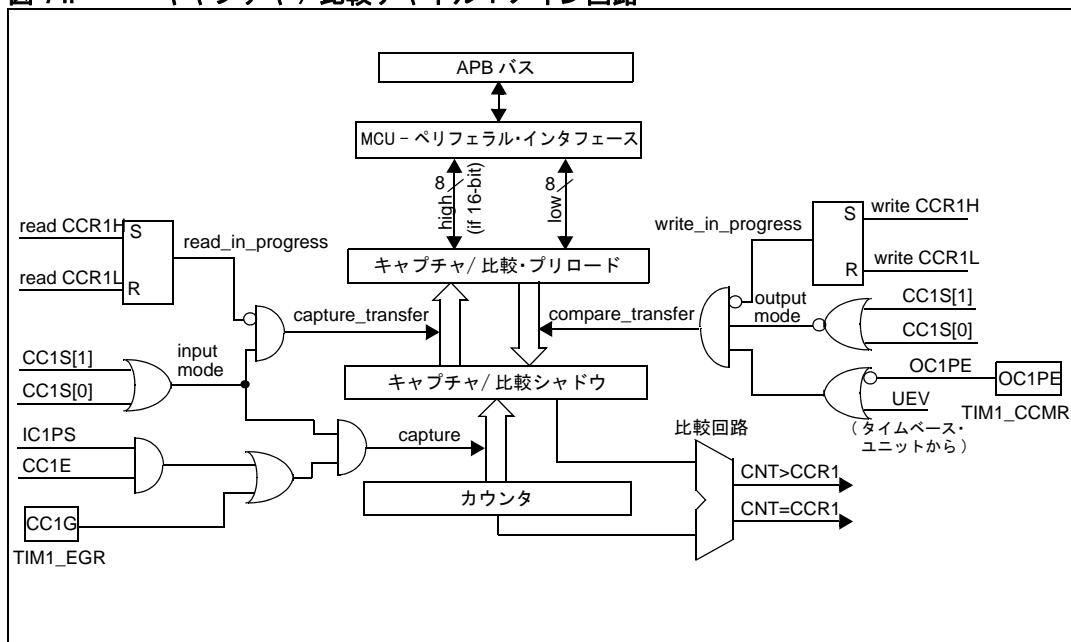
入力ステージは対応する TIx 入力信号をサンプルし、フィルタリングを行った TIxF 信号を生成します。次に、エッジ検出回路と極性選択回路が、スレーブ・モード・コントローラへのトリガ入力もしくはキャプチャ・コマンドとして使用される信号 (TIxFPx) を生成します。この信号はプリスケーラを通して (ICxPS) キャプチャ・レジスタに渡されます

図 73. キャプチャ / 比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは OCxREF（アクティブ・ハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 74. キャプチャ / 比較チャネル 1 メイン回路



参考資料

図 75. キャプチャ / 比較チャネル (チャネル 1 - 3) の出力ステージ

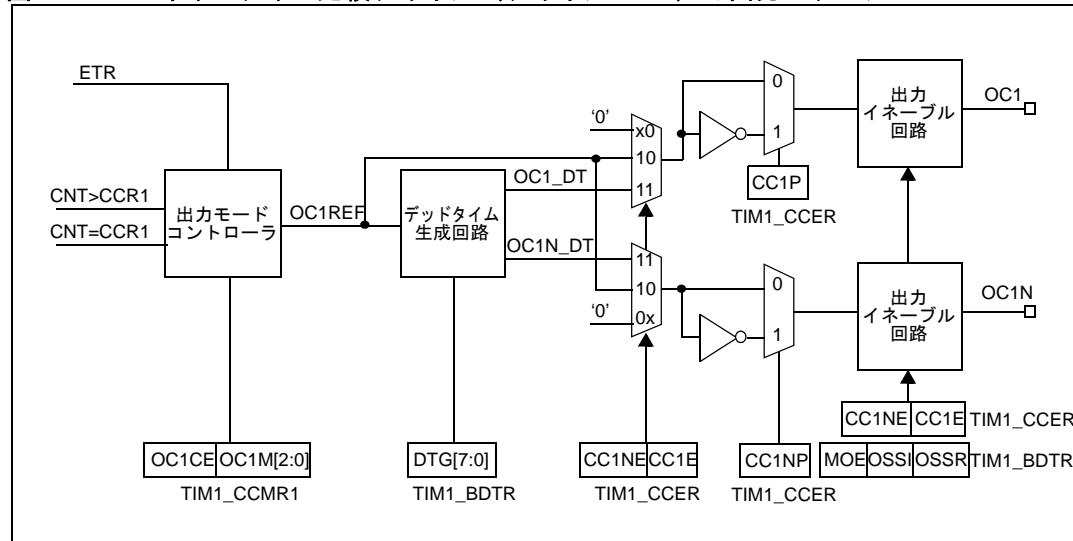
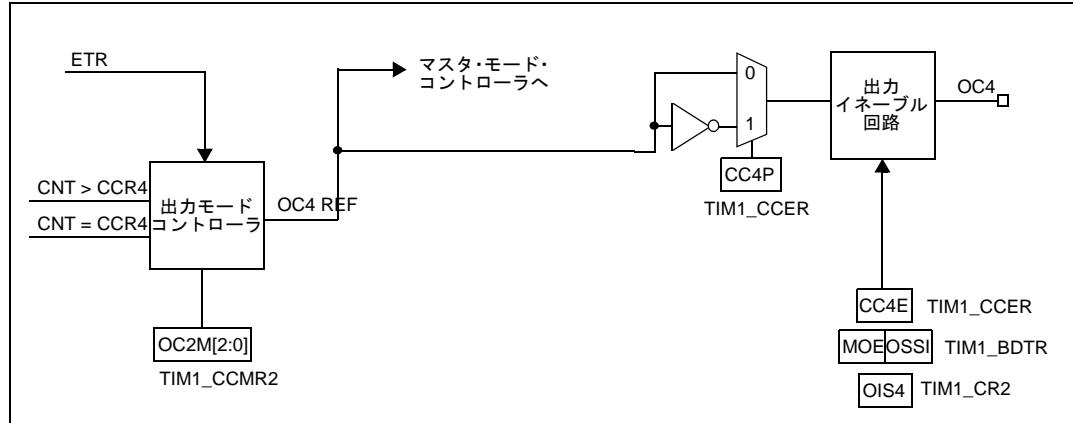


図 76. キャプチャ / 比較チャネル (チャネル 4) の出力ステージ



キャプチャ / 比較ブロックは 1 個のプリロード・レジスタと 1 個のシャドウ・レジスタで構成されています。ソフトウェアからの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、これがカウンタとの比較動作に使われます。

参考資料

12.3.6 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 IC_x 信号の変化が検出されたときに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 (TIM_x_CCR_x) が使用されます。キャプチャが発生すると、TIM_x_SR レジスタの 対応する CC_xIF フラグがセットされ、リクエスト発生が許可されていれば、割込みリクエストもしくは DMA リクエストが発生します。CC_xIF フラグがすでにセットされているときにキャプチャ動作が発生すると、TIM_x_SR レジスタのオーバキャプチャ・フラグ CC_xOF がセットされます。CC_xIF フラグはソフトウェアでこのビットに 0 を書込むか、もしくは TIM_x_CCR_x レジスタの内容を読出す (16bit レジスタの場合には下位バイトの読出し) ことでクリアされます。CC_xOF のクリアは、このビットに 0 を書込むことで行います。

次の例は、TI1 入力が立上がったときに、カウンタの値を TIM_x_CCR1 にキャプチャするものです。このためには次のように操作します。

- 有効な入力を選択します。TIM_x_CCR1 は TI1 入力とリンクされていなければならず、このためには TIM_x_CCMR1 レジスタの CC1S ビットに 01 を書込みます。CC1S の値が 00 から変化すると、チャネルは入力用に構成され、TIM_x_CCR1 レジスタは読出し専用となります。
- 入力フィルタでタイマに接続される信号に応じて必要な応答時間を設定します。
(TIM_x_CCMRx レジスタの ICxF ビット) 入力信号が反転したとき、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、フィルタの応答時間として 5 クロック・サイクルを超える長さを指定しなければなりません。一例として、連続した 8 個の入力サンプル値が新しい値を持っているときに、TI1 で有効な変化があったとすることができます。これを設定するには、TIM_x_CCMR1 レジスタの IC1F ビットに 0011 を書込みます。(この場合、f_{DTS} 周波数によるサンプル)
- TI1 チャ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、TIM_x_CCER レジスタの CC1P ビットに 0 を書込みます。(この場合、立上りエッジの選択)
- 入力プリスケーラを設定します。今回の例では有効な信号変化ごとにキャプチャを行いますので、プリスケーラは無効とします。(TIM_x_CCMR1 レジスタの IC1PS ビットに 00 を書込みます。)
- カウンタからキャプチャ・レジスタへのキャプチャ・モードをイネーブルとします。これには、TIM_x_CCER レジスタの CC1E ビットをセットします。
- 必要ならば、関連する割込みリクエストを許可状態とするために TIM_x_DIER レジスタの CC1IE の設定を行うか、DMA リクエストのために TIM_x_DIER レジスタの CC1DE レジスタの設定を行います。

入力キャプチャ動作が行われると、

- アクティブな信号変化で、TIM_x_CCR1 レジスタにはカウンタの値が格納されます。
- CC1IF フラグ (割込みリクエスト) がセットされます。フラグがクリアされていない状態で 2 回目以降のキャプチャが発生すると CC1OF ビットも同時にセットされます。
- CC1IE ビットの設定に応じて、割込みリクエストが発生します。
- CC1DE ビットの設定に応じて、DMA リクエストが発生します。

オーバキャプチャの対応では、オーバキャプチャ・フラグの読出し前にデータを読出すようにしてください。これにより、フラグ読出し後、データ読出し前に発生し得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。

注意 :

TIM_x_EGR レジスタの対応する CC_xG ビットをセットすることで、IC 割込みリクエストと DMA リクエストの一方、もしくは両方をソフトウェアで発生させることができます。

参考資料

12.3.7 PWM 入力モード

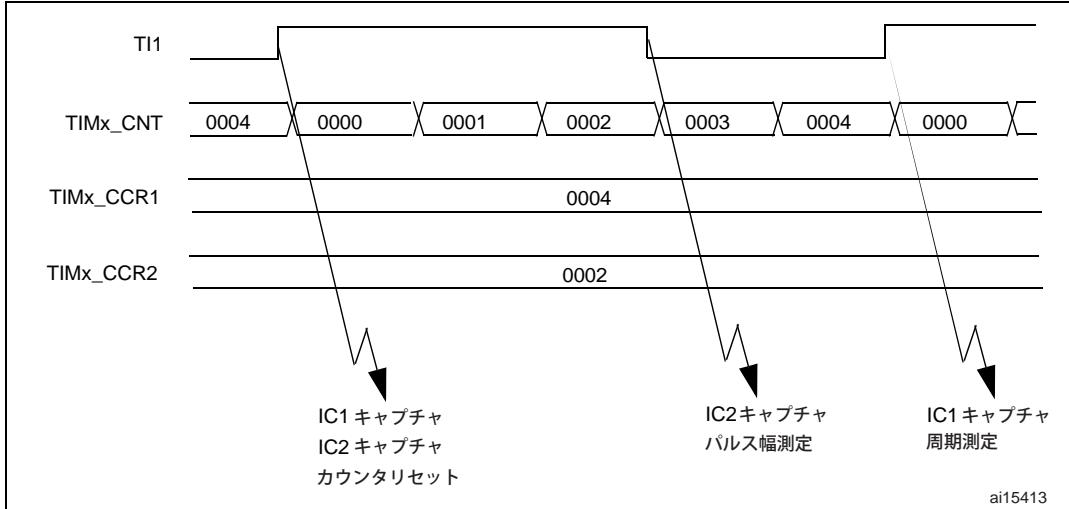
このモードは、入力キャプチャの特別な応用例です。操作手順は入力キャプチャ・モードと同様ですが、以下の点が異なっています。

- 2 個の ICx 信号が同じ TIx 入力に配置されます。
- この 2 個の ICx 信号は、アクティブエッジが逆の極性に設定されます。
- 2 個の TIxFP 信号の一方がトリガ入力として選択され、スレーブ・モード・コントローラがリセット・モードに設定されます。

一例として、TI1 上の PWM 信号の周期 (TIMx_CCR1 レジスタ) とデューティサイクル (TIMx_CCR2 レジスタ) を測定するものとします。このときには、以下の操作手順を実行します。(手順は CK_INT 周波数とプリスケーラの値によって、若干異なることがあります。)

- TIMx_CCR1 のアクティブな入力を選択します。ここでは、TIMx_CCMR1 レジスタの CC1S ビットに 01 を書込みます。(TI1 を選択)
- TI1FP1 のアクティブな極性を選択します。(TIMx_CCR1 キャプチャとカウンタクリアの両方に使用します。) これには CC1P ビットに 0 を書込みます。(立上りエッジ)
- TIMx_CCR2 のアクティブな入力を選択します。ここでは、TIMx_CCMR1 レジスタの CC2S ビットに 10 を書込みます。(TI1 を選択)
- TI1FP2 のアクティブな極性を選択します。(TIMx_CCR2 のキャプチャに使用します。) これには CC2P ビットに 1 を書込みます。(立下りエッジ)
- 有効なトリガ入力を選択します。ここでは、TIMx_SMCR レジスタの TS ビットに 101 を書込みます。(TI1FP1 を選択)
- スレーブ・モード・コントローラをリセット・モードとします。これには、TIMx_SMCR レジスタの SMS ビットを 100 とします。
- キャプチャ動作を開始します。これには、TIMx_CCER レジスタの CC1E ビットと CC2E ビットに 1 を書込みます。

図 77. PWM 入力モードのタイミング



参考資料

12.3.8 出力強制駆動モード

このモード (TIMx_CCMRx レジスタの CCxS=00) では、各出力比較信号 (OCxREF 及び OCx/OCxN) 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ルーレベルとするには、対応する TIMx_OCMRx レジスタの OCxM ビットに 101 を書込みます。これにより、OCxREF はハイレベル (OCxREF は常にアクティブ・ハイです) となり、OCx は CCxP 極性ビットと逆のレベルとなります。

つまり、CCxP=0 (OCx アクティブ・ハイ) \Rightarrow OCx はハイレベルとなります。

TIMx_CCMRx レジスタの OCxM ビットに 100 を書込むことで、OCxREF 信号をローレベルとすることができます。

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、これによるフラグのセットは行われます。フラグの状態によっては割込みリクエストや DMA リクエストが発生することもあります。これに関しては、次の出力比較モードのセクションに記されています。

12.3.9 出力比較モード

このモードは出力波形を制御する場合や、一定時間が経過したことを表示する場合に使用されます。

キャプチャ / 比較レジスタとカウンタの値が一致したとき、出力比較は次のように働きます。

- 対応する出力ピンに、プログラムで設定されている、出力比較モードとしての値 (TIMx_CCMRx レジスタの OCxM ビット) と出力極性 (TIMx_CCER レジスタの OCxP ビット) を適用した信号を出力します。値が一致したときに出力ピンは、固定レベルを維持 (OCxM=000)、アクティブルーレベルにセット (OCxM=001)、インアクティブルーレベルにセット (OCxM=010)、反転 (OCxM=011) などの状態とすることができます。
- 割込みステータス・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)
- 対応する割込みリクエストの許可ビット (TIMx_DIER レジスタの CCxIE ビット) がセットされていれば、割込みリクエストが発生します。
- 対応する DMA リクエストの許可ビット (TIMx_DIER レジスタの CCxDE ビット) がセットされていれば、DMA リクエストが発生します。(DMA リクエストの選択には、TIMx_CR2 レジスタの CCDS ビットが使われます。)

TIMx_CCRx レジスタの設定においてプリロード・レジスタが使用されるか否かは TIMx_OCMRx レジスタの OCxPE ビットで指示できます。

出力比較モードでは、更新イベント UEV は OCxREF 及び 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 カウント分となります。出力比較モードは 1 個のパルスを発生させる (単一パルス発生モード) ために使用することができます。

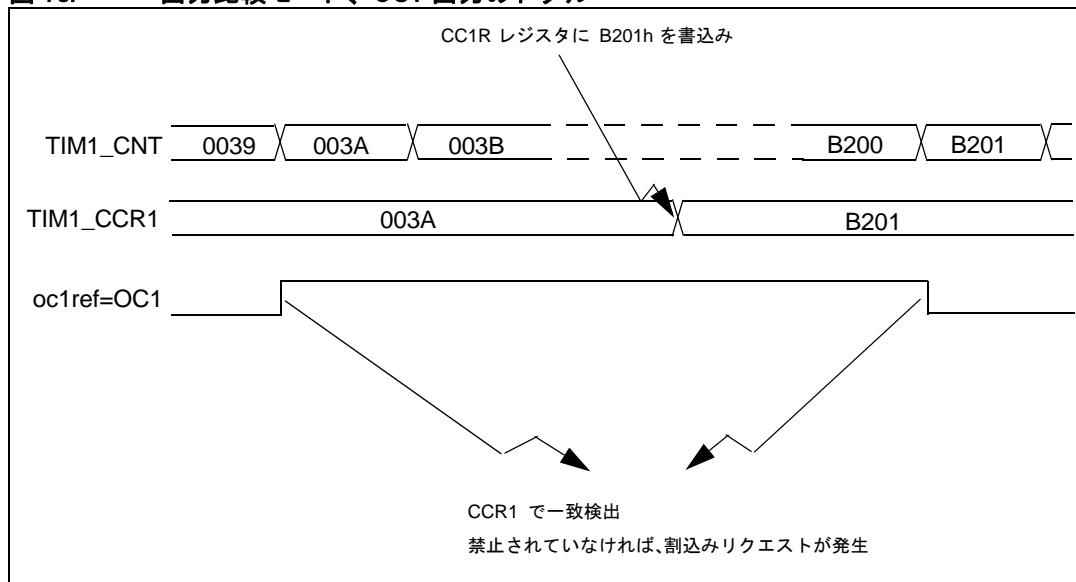
参考資料

操作手順 :

1. カウンタ・クロックを選択します。(内部、外部、プリスケーラ)
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに希望のデータを書き込みます。
3. 割込みリクエストの発生が必要であれば、CCxEIE ビットをセットします。
4. 出力モードを下の例のように選択します。
 - OCxM = 011 を書き込み、CNT と CCRx が一致したときに OCx 出力ピンを反転させます。
 - OCxPE = 0 を書き込み、プリロード・レジスタの使用を行わない状態とします。
 - CCxP = 0 を書き込み、極性としてアクティブ・ハイを選択します。
 - CCxE = 1 を書き込み、出力をイネーブルとします。
5. カウンタの動作を開始するには、TIMx_CR1 レジスタの CEN ビットをセットします。

プリロード・レジスタが使用されていません (OCxPE=0) ので、TIMx_CCRx レジスタは出力波形を制御するためソフトウェアで常時変更ができます。(OCxPE=1 の時には次の更新イベント UEV でのみシャドウ・レジスタが変更されます。) タイミングの一例は図 78 に示されます。

図 78. 出力比較モード、OC1 出力のトグル



参考資料

12.3.10 PWM モード

パルス幅変調 (PWM) モードでは、周波数が TIMx_ARR レジスタで定まり、デューティ比が TIMx_CCRx レジスタで指定される信号を生成することができます。

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) を書込むことで、独自に選択することができます。このとき、 TIMx_CCMRx レジスタの OCxPE ビットをセットすることで対応するチャネルのプリロード機能を有効としておくことが必要とされます。また、アップカウント・モードやセンターアライン・モードでは、 TIMx_CR1 レジスタの ARPE ビットをセットすることで、自動再ロード・プリロード・レジスタも有効にしておく必要も生じます。

プリロード・レジスタの内容は、更新イベントが発生したときにのみシャドウ・レジスタにコピーされるため、カウンタをスタートする前に、 TIMx_EGR レジスタの UG ビットのセットで、すべてのレジスタを初期化しておくことが必要とされます。

OCx 信号がアクティブ・ハイであるか、アクティブ・ローであるかの極性は、 TIMx_CCER レジスタの CCxP ビットを使用して、ソフトウェアで指定することができます。 OCx 出力はブレーク機能が組み込まれていない場合には、 TIMx_CCER レジスタの CCxE ビットによってイネーブルとされます。ブレーク機能を有するときには、(TIMx_CCER レジスタと TIMx_BDTR レジスタの) CCxE 、 CCxNE 、 MOE 、 OSSI 及び OSSR ビットの組合せによって指示されます。詳細は TIMx_CCERx レジスタの説明をお読みください。

PWM モード (1 及び 2) では、 $\text{TIMx_CCR} \leq \text{TIMx_CNT}$ か $\text{TIMx_CNT} \leq \text{TIMx_CCR}$ かを判別する (カウントの方向に依存します) ため、 TIMx_CNT と TIMx_CCR は常時比較されます。

タイマは PWM をエッジアライン・モードとセンターアライン・モードのいずれかで生成することができ、これは TIMx_CR1 レジスタの CMS ビットで指定します。

参考資料

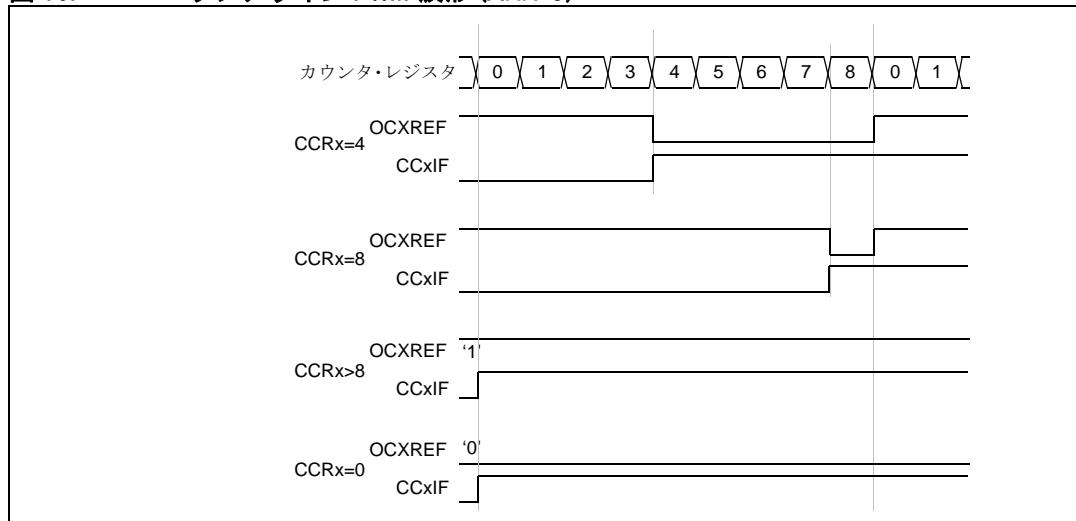
PWM エッジアライン・モード

- 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、アップカウント動作がアクティブとなっています。 [アップカウント・モード \(220 ページ\)](#) を参照してください。

以下の例では PWM モード 1 を想定します。PWM 基準信号 OCxREF は TIMx_CNT < TIMx_CCRx の間はハイレベルに、他の場合にはローレベルとなります。 TIMx_CCRx の比較値が自動再ロード値 (TIMx_ARR レジスタの値) より大きい場合には、 OCxREF は 1 を保ち、比較値が 0 のときには OCxREF は 0 を保ちます。[図 79](#) は TIMx_ARR=8 の時のエッジアライン PWM 波形の例を示しています。

図 79. 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となっています。 [ダウンカウント・モード \(223 ページ\)](#) の説明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 の間はローレベルとなり、他の場合にはハイレベルとなります。比較値 TIMx_CCRx が TIMx_ARR レジスタの自動再ロード値より大きなときには、OCxREF は 1 を保ちます。このモードでは デューティ比 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

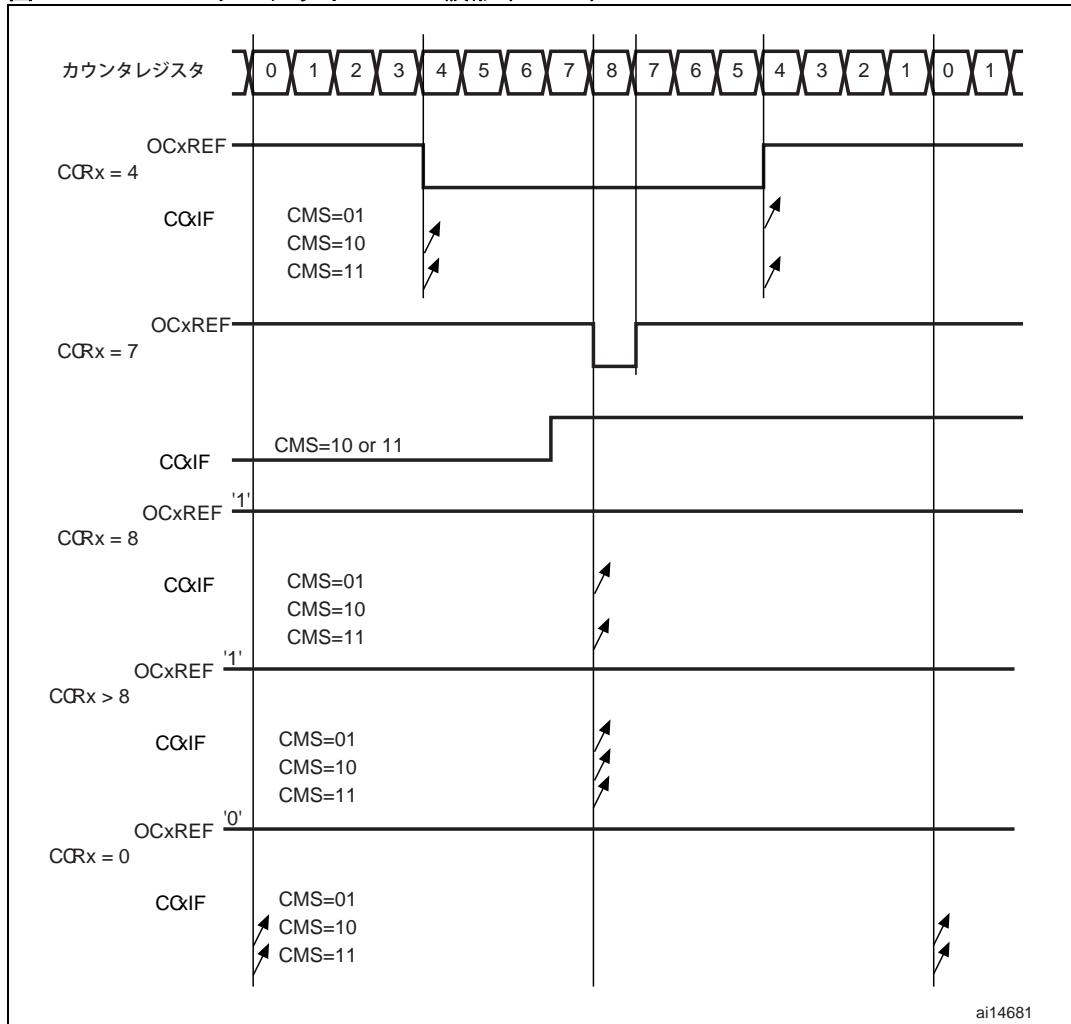
センターアライン・モードは TIMx_CR1 レジスタの CMS ビットが 00 ではないときにアクティブとなります。(OCxREF/OCx 信号に関しては、他の設定はすべて同じ効果を持っています。) CMS ビットの設定に応じて、カウンタがカウントアップを行っているとき、カウントダウンを行っているとき、その両方のとき、比較フラグがセットされます。 TIMx_CR1 レジスタの方向ビット (DIR) はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。 [センターアライン・モード \(アップ/ダウンカウント\) \(226 ページ\)](#) を参照してください。

図 80 は次の条件の例で、センターアライン PWM 波形を示しています。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- TIMx_CR1 レジスタの CMS=01 で選択されるセンターアライン・モード 1 に対応して、カウンタがカウントダウンのときにフラグがセットされます。

参考資料

図 80. センターライン PWM 波形 (ARR=8)



ai14681

センターライン・モードを使用する際に、

- センターライン・モードをスタートするときには、アップ・ダウン設定が使用されます。カウンタは TIMx_CR1 レジスタの DIR ビットに書込まれている値に応じて、カウントアップ、カウントダウンを実行します。さらに、DIR ビットと CMS ビットはソフトウェアで同時に変更することはできません。
- センターライン・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ため好ましくありません。特に、
 - 自動再ロード値より大きな値をカウンタに書込んだ場合 (TIMx_CNT > TIMx_ARR) には、方向は更新されません。カウントアップ中であれば、カウンタはその動作を続けます。
 - カウンタに 0 もしくは TIMx_ARR 値を書込んだときには、方向が更新されますか、更新イベント UEV は発生しません。
- センターライン・モードを使用する確実な方法は、カウンタをスタートする直前に TIMx_EGR レジスタの UG ビットをセットしてソフトウェアで更新イベントを発生させ、動作中はカウンタへの書き込みを行わないことです。

参考資料

12.3.11 相補出力とデッドタイム挿入

高機能制御タイマ TIM1 と TIM8 は 2 本の相補信号を出力することができ、出力におけるスイッチング・オンとスイッチング・オフ動作を管理することができます。

この時間は通常デッドタイムと称されているもので、出力に接続されているデバイスとの特性（レベルシフタの伝播遅延、パワースイッチにおける遅延、など）に応じて調整することができます。

出力（主出力 OC_x 及び補助出力 OC_{xN}）の極性は、出力ごとに独自に設定することができます。これは TIM_x_CCER レジスタの CC_{xP} ビット及び CC_{xNP} ビットへの書き込みで指定します。

相補信号 OC_x 及び OC_{xN} は、TIM_x_CCER レジスタの CC_{xE} ビット及び CC_{xNE} ビット、TIM_x_BDTR レジスタと TIM_xCR2 レジスタの MOE ビット、OIS_x ビット、OIS_{xN} ビット、OSSI ビット及び OSSR ビットといった複数の制御ビットの組み合わせで有効化されます。詳細は [表 54: ブレーク機能をもつ相補出力 OC_x 及び OC_{xN} チャネルの出力制御ビット \(275 ページ\)](#) を参照してください。特に、IDLE 状態に切替わるとき（MOE が 0 になると）にデッドタイムが挿入されます。

デッドタイム挿入は CC_{xE} ビットと CC_{xNE} ビットの両方がセットされており、ブレーク回路が組込まれている場合には、さらに MOE ビットがセットされているときにイネーブルとなります。個々のチャネルは 10 ビットのデッドタイム生成回路をもっています。この回路は基準波形 OC_{xREF} から 2 種の出力 OC_x と OC_{xN} を生成します。OC_x と OC_{xN} がアクティブ・ハイの時には、

- OC_x 出力信号は、立上りエッジが基準波形の立上りエッジから遅れている点を除けば、基準信号と同じです。
- OC_{xN} 出力信号は、立上りエッジが基準波形の立下りエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延時間がアクティブな出力 OC_x 及び OC_{xN} のパルス幅より大きな時には、対応するパルスは生成されません。

以下の図はデッドタイム生成回路の出力信号と基準信号 OC_{xREF} との関係を示しています。（例では、CC_{xP}=0、CC_{xNP}=0、MOE=1、CC_{xE}=1 及び CC_{xNE}=1 を想定しています。）

図 81. デッドタイム挿入のある相補出力

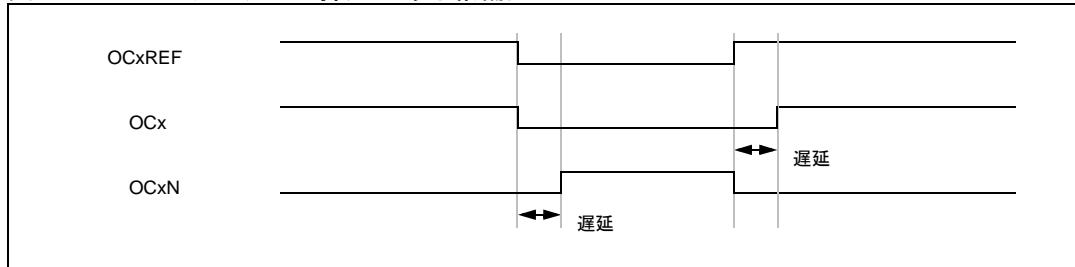
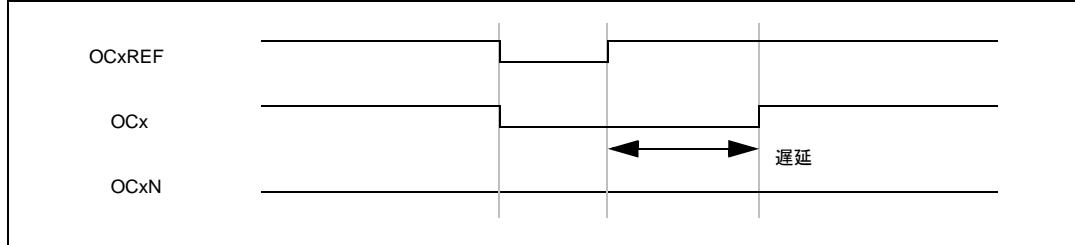
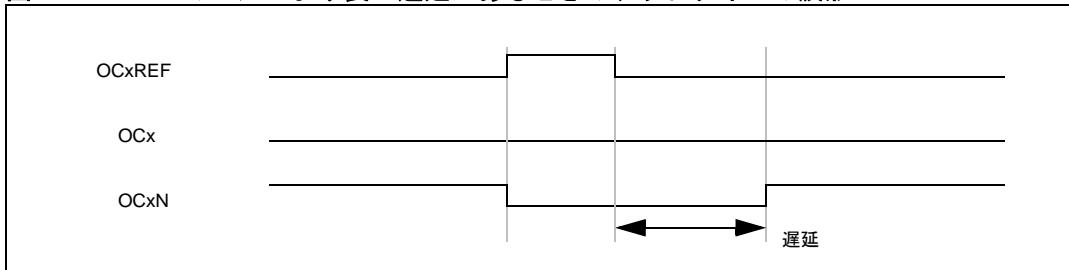


図 82. 負のパルスより長い遅延があるときのデッドタイムの波形



参考資料

図 83. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延時間はチャネル間で共通で、ソフトウェアから TIMx_BDTR レジスタの DTG ビットで指定できます。遅延時間の計算式に関しては [セクション 12.4.18: ブレーク / デッドタイムレジスタ \(\$\text{TIMx}_\text{BDTR}\$ \) \(279 ページ\)](#) に記されています。

OCxREF 信号から OCx 及び OCxN へ

出力モード（強制駆動、出力比較、PWM）では、OCxREF 信号が OCx 出力と OCxN 出力に送られますが、この出力は TIMx_CCER レジスタの CCxE ビット及び CCxNE ビットで設定されます。

この設定で、一方の出力に特定の波形（PWM もしくはアクティブレベルに固定）を出力し、他方をインアクティブレベルに固定することも可能です。他の例としては、両方の出力をインアクティブレベルとする場合や、両方をデッドタイムを有する有効な相補出力とすることができます。

注意：

$OCxN$ のみがイネーブルとされた ($CCxE=0, CCxNE=1$) 場合、この出力は相補的ではなく、 $OCxREF$ がハイレベルとなると、出力はアクティブとなります。例えば、 $CCxNP=0$ であれば $OCxN=OCxREF$ となります。他方、 OCx と $OCxN$ の両方がイネーブルとされた ($CCxE=CCxNE=1$) ときには、これらは相補的で、 $OCxREF$ がハイレベルになると OCx はアクティブとなり、 $OCxREF$ がローレベルとなると $OCxN$ はアクティブとなります。

12.3.12 ブレーク機能の利用

ブレーク機能を使用するとき、出力イネーブル信号とインアクティブレベルは追加された制御ビット (TIMx_BDTR レジスタの MOE、OSSI、OSSR、及び TIMx_CR2 レジスタの OISxN ビット) によって任意に設定することができます。ただし、 OCx と $OCxN$ 出力は両方を同時にアクティブレベルとすることはできません。詳細は [表 54: ブレーク機能をもつ相補出力 \$OCx\$ 及び \$OCxN\$ チャネルの出力制御ビット \(275 ページ\)](#) を参照してください。

ブレークの要因となる信号はブレーク入力ピンもしくはリセット・クロック・コントローラ (RCC) のクロック・セキュリティシステム (CSS) から発せられるクロック異常イベントです。クロック・セキュリティシステムの詳細は [セクション 6.2.7: クロック・セキュリティシステム \(CSS\) \(78 ページ\)](#) を参照してください。

リセットが解除されたとき、ブレーク回路はディセーブル状態で MOE ビットはローレベルとなっています。 TIMx_BDTR レジスタの BKE ビットをセットすることで、ブレーク機能を有効とすることができます。ブレーク入力の極性は同じレジスタの BKP ビットで指定できます。BKE ビットと BKP ビットは常時変更が可能です。

MOE の立下りエッジは任意の時点で生じるため、実際に出力に影響を与える信号と (TIMx_BDTR レジスタからアクセスできる) 同期制御ビットの間に、再同期回路がおかれています。このため、ローレベルの MOE に 1 を書き込んだとき、これを正しく読出すには所定の遅延（ダミー命令）をおかねばなりません。これはクロックと非同期の信号を書き込み、クロックに同期した信号を読出す状況に相当しています。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

ブレーク（ブレーク入力上の所定のレベル）が発生すると、

- MOE ビットは直ちにクリアされ、出力をインアクティブ状態とし、OSSI ビットで指示されているようにアイドル状態もしくはリセット状態とします。この機能は MCU 発振回路がオフであっても機能します。
- 個々の出力チャネルは MOE=0 となったとき、TIMx_CR2 レジスタの OISx ビットで指定されているレベルに駆動されます。OSSI=0 であれば、タイマはイネーブル状態の出力を解放し、OSSI=1 であればハイレベルを保ちます。
- 相補出力が使用されているときには、
 - 出力は先ずリセット状態のインアクティブ状態（極性に依存します）とされます。これはクロックとは非同期に行われ、タイマにクロックが供給されていないときであっても機能します。
 - タイマ・クロックが供給されているときには、デッドタイム後に OISx と OISxN ビットで指定されている所定のレベルに出力を駆動するため、デッドタイム発生回路が作動します。この場合でも、OCx 及び OCxN は同時にアクティブレベルに駆動することはできません。MOE のクロック同期に関連して、デッドタイムの時間は通常動作より 1 ビットタイム分 (2 CK_TIM クロック・サイクル程度) 長くなります。
 - OSSl=0 であればタイマは有効状態の出力を解放し、OSSI=1 であればそのままの値を維持、もしくは、CCxE もしくは CCxNE ビットがハイレベルになったときにハイレベルとなります。
- ブレーク状態フラグ (TIMx_SR レジスタの BIF ビット) がセットされます。TIMx_DIER レジスタの BIE ビットがセットされていれば割込みリクエストが発生し、BDE ビットがセットされていれば DMA リクエストが発生します。
- TIMx_BDTR レジスタの AOE ビットがセットされているとき、MOE ビットは次の更新イベント UEV で再び自動的にセットされます。一例として、これを制御目的で利用することもできます。AOE ビットがセットされていないときには、MOE ビットはソフトウェアで 1 を書込むまで 0 状態を維持します。これは、MOE 信号をセキュリティ目的で利用したり、ブレーク入力をパワー駆動回路や温度センサなどのアラームとして利用することができます。

注意：

ブレーク入力はレベル信号として機能します。このため、ブレーク入力が（自動的に、もしくはソフトウェアの操作で）アクティブの時には、MOE ビットがセットされることはありません。また、この間ステータス・フラグ BIF はクリアできません。

ブレークは、BRK 入力が所定のレベルにあり、TIMx_BDTR レジスタの BKE ビットで機能がイネーブルとされているときに発生します。

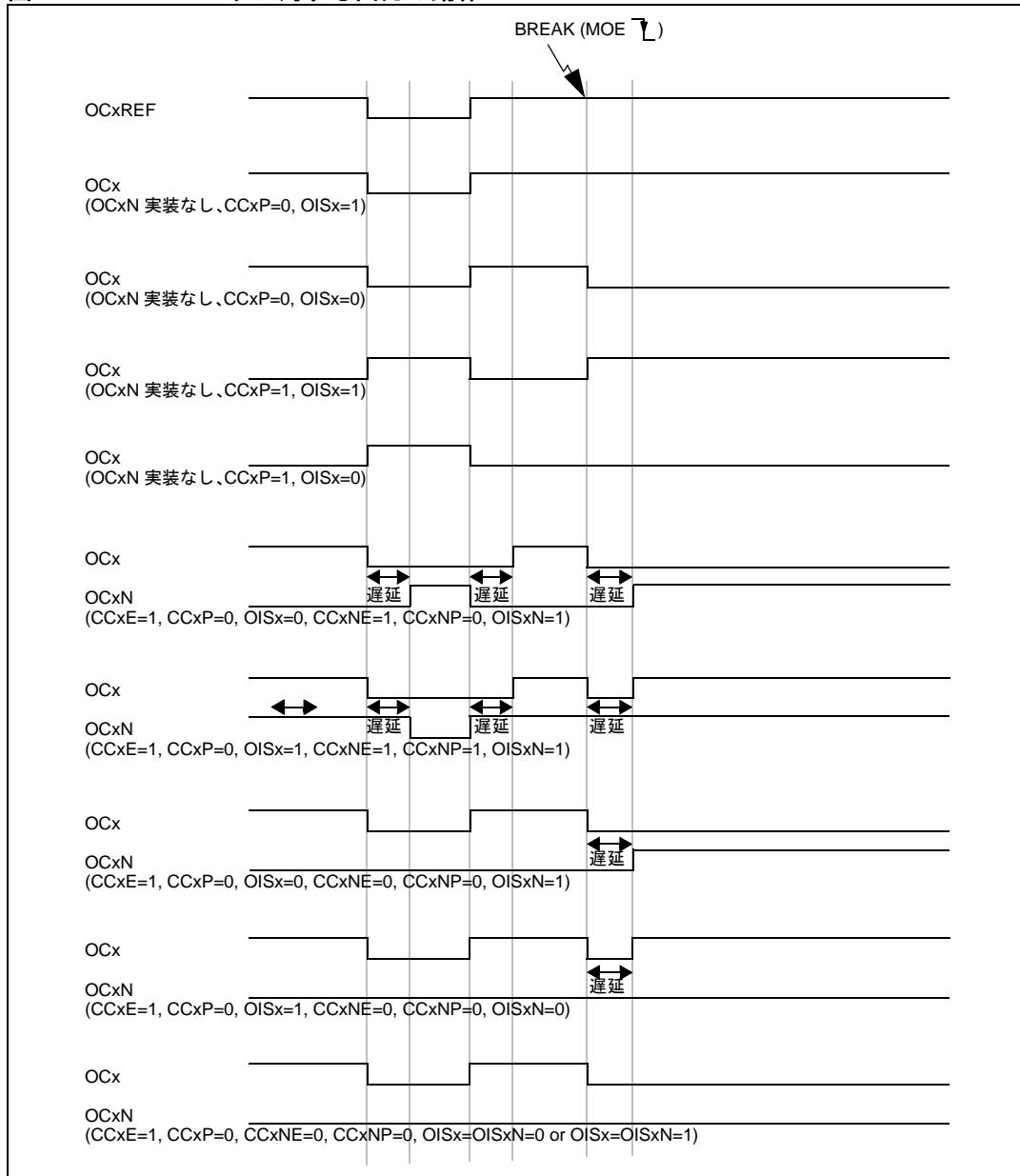
さらに、ブレーク入力と出力の管理は、アプリケーションに対する安全策として、ブレーク回路内で書き込み保護が施されています。これにより、いくつかのパラメータ（デッドタイム時間、OCx/OCxN 極性とディセーブル時の状態、OCxM 設定、ブレークイネーブルと極性）を固定することができます。このとき、TIMx_BDTR レジスタの LOCK ビットでレベル 3 の保護を指定します。これについては [セクション 12.4.18: ブレーク / デッドタイムレジスタ \(TIMx_BDTR\) \(279 ページ\)](#) を参照してください。LOCK ビットの書き込みはリセット後 1 回に限られます。

図 84 はブレークに対する出力の挙動の例を示しています。



参考資料

図 84. ブレークに対する出力の動作



参考資料

12.3.13 外部イベントによる OCxREF 信号のクリア

指定されたチャネルの OCxREF 信号は ETRF 入力をハイレベルとすることでローレベルとすることができます。(TIMx_CCMRx レジスタの対応する OCxCE イネーブル・ビットが 1 に設定されているとき) 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。

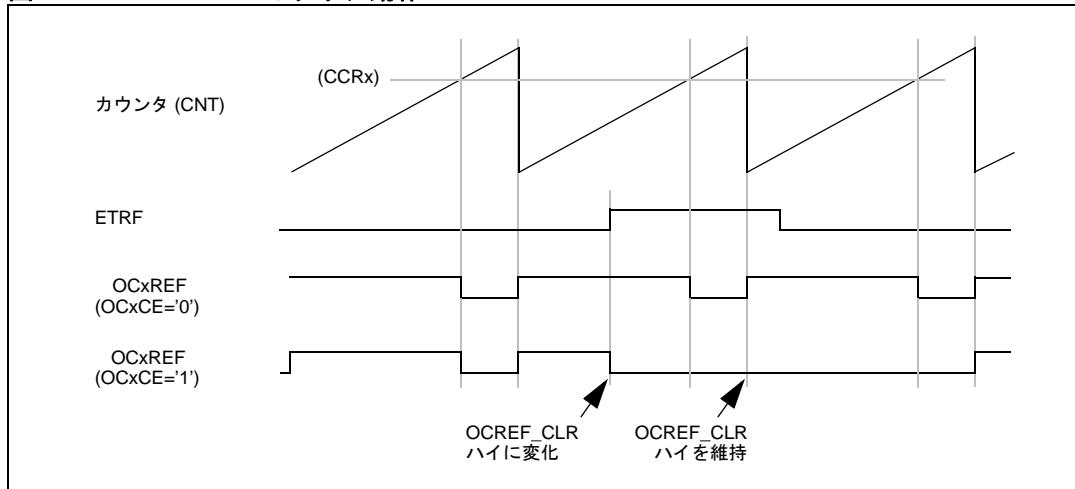
この機能は出力比較モードと PWM モードでのみ使用でき、強制駆動モードでは利用できません。

例えば、通常動作のため、OCxREF 信号を比較回路の出力に接続することができます。この場合、ETR は次のように設定しなければなりません。

1. 外部トリガ・プリスケーラはオフとしておきます。このため、TIMx_SMCR レジスタの ETPS[1:0] を 00 とします。
2. 外部クロック・モード 2 はディセーブルとします。TIMx_SMCR レジスタの ECE ビットは 0 にセットします。
3. 外部トリガ極性 (ETP) と外部トリガフィルタ (ETF) は必要に応じた設定とします。

図 85 はイネーブル・ビット OCxCE の値と、ETRF 入力がハイレベルになったときの OCxREF の挙動を示しています。この例では TIMx タイマは PWM モードで動作しています。

図 85. OCxREF のクリア動作



参考資料

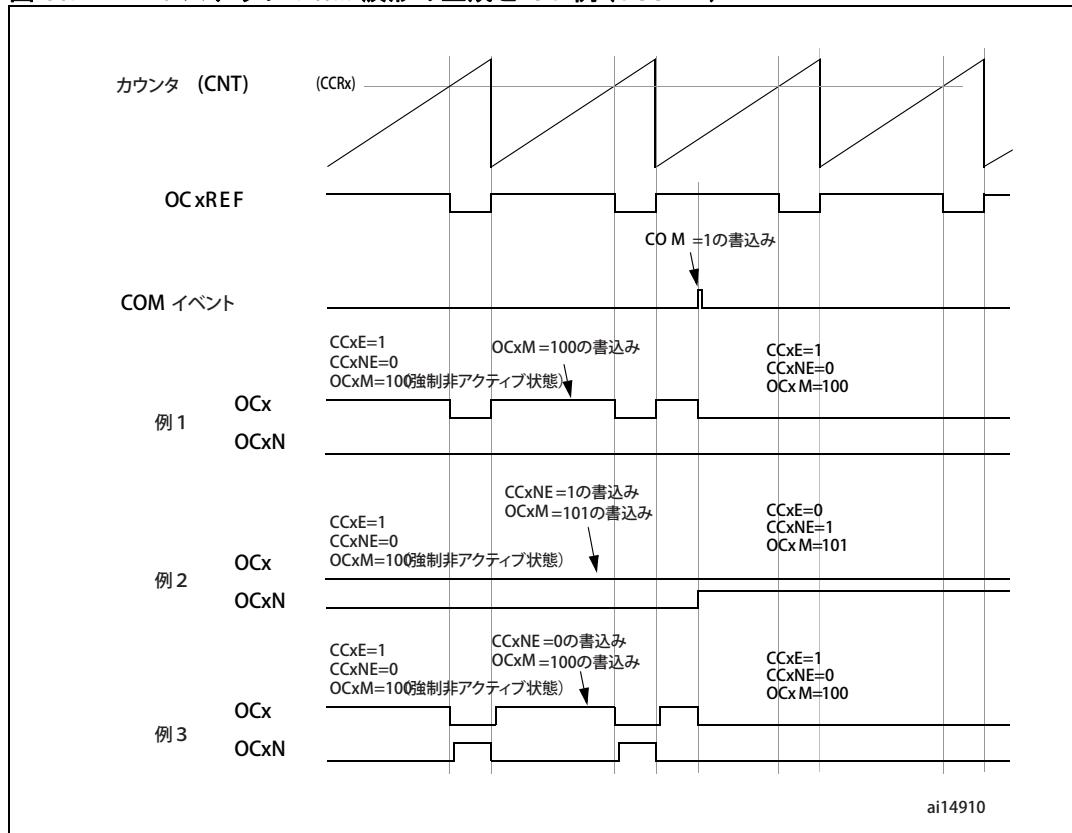
12.3.14 6 ステップ PWM 波形生成

チャネルで相補出力が使用されているとき、OCxM ビット、CCxE ビット及び CCxNE ビットでプリロード機能が有効となります。プリロードビットの値は COM 転流イベントでシャドウビットにコピーされます。これにより、次のステップの値をあらかじめ設定し、全チャネルの設定を同時に変更することができます。COM イベントはソフトウェアで TIM1_EGR レジスタの COM ビットをセットするか、TRGI 信号の立上りエッジでハードウェアによりセットすることで発生させることができます。

TIM1_SR レジスタの COMIF フラグは COM イベントが発生したときにセットされます。このフラグは (TIM1_DIER レジスタの COMIE ビットがセットされているとき) 割込みか、(TIM1_DIER レジスタの COMDE ビットがセットされているとき) DMA リクエストを発生することができます。

図 86 は COM イベントが発生したときの OCx と OCxN 出力の挙動を、設定の異なる 3 種の例で示しています。

図 86. 6 ステップ PWM 波形の生成と CO 例 (OSSR=1)



12.3.15 単一パルス発生モード

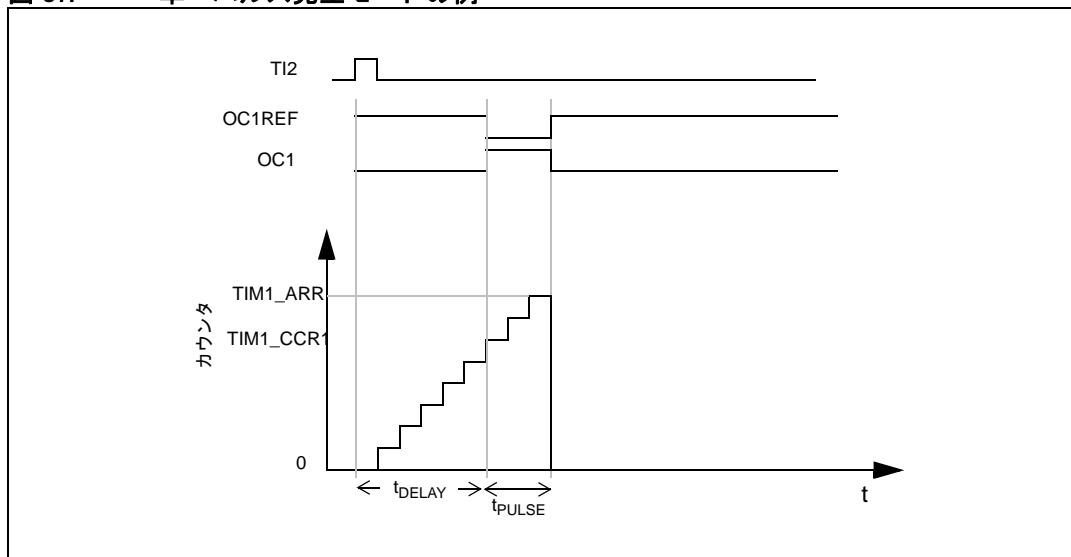
単一パルス発生モード (OPM: One Pulse Mode) はこれまでに説明したモードの一ケースです。このモードでは、トリガに応じてカウンタをスタートさせ、設定した遅延時間の後に、設定した幅のパルスを発生させます。

カウンタのスタートはスレーブ・モード・コントローラによって行うことができます。パルス波形の生成は出力比較モード、もしくは PWM モードによって行います。単一パルス発生モードは TIM1_CR1 レジスタの OPM ビットをセットすることで選択されます。このときには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ときのみ正しく生成されます。カウンタをスタートさせる（タイマがトリガ信号を待つ状態となる）前に、次の設定が必要とされます。

- アップカウント時 : $CNT < CCRx \leq ARR$ (特に $0 < CCRx$)
- ダウンカウント時 : $CNT > CCRx$

図 87. 単一パルス発生モードの例



一例として、TI2 入力ピンで立上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} 後に発生させることとします。

TI2FP2 をトリガ 1 として使用します。：

- TI2FP2 を TI2 に配置するために、 $TIMx_CCMR1$ レジスタの $CC2S=01$ を書き込みます。
- TI2FP2 で立上りエッジを検出します。 $TIMx_CCER$ レジスタの $CC2P=0$ 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) とします。このためには、 $TIMx_SMCR$ レジスタの $TS=110$ を書き込みます。
- TI2FP2 でカウンタをスタートさせます。このためには、 $TIMx_SMCR$ レジスタの SMS ビットに 110 (トリガ・モード) を書き込みます。

OPM 波形は比較値レジスタの設定（クロック周波数とプリスケーラに依存します）で定まります。

- 遅延時間 t_{DELAY} は $TIMx_CCR1$ レジスタに書込まれた値によって定義されます。
- パルス幅 t_{PULSE} は自動再ロード・レジスタと比較値の差 ($TIMx_ARR - TIMx_CCR1$) によって定義されます。

参考資料

- カウンタの値が比較値と一致したとき 0 から 1 に変化し、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。これを行うためには、
TIMx_CCMR1 レジスタの OC1M=111 をセットします。このとき TIMx_CCMR1 レジスタ
の OC1PE=1 とし、TIMx_CR1 レジスタの ARPE=1 とすることでプリロードを有効とす
ることもできます。この場合には、比較値を TIMx_CCR1 レジスタに、自動再ロード
値を TIMx_ARR レジスタに書込んだ上で、ソフトウェアで UG ビットをセットするこ
とで更新イベントを発生させ、TI2 上の外部トリガ・イベントを待ちます。この例で
は CC1P には 0 を書込みます。

上の例では TIMx_CR1 レジスタの DIR ビットと CMS ビットはローレベルでなければなりません。

発生するパルスは 1 個ですので、次の更新イベント（カウンタが自動再ロード値に達し
て、カウンタの値が 0 に戻る時点）でカウンタが停止するよう、TIMx_CR1 レジスタの
CPM ビットには 1 を書込みます。

応用例 : OCx 即時イネーブル

単一パルス発生モードでは、TIx 入力のエッジ検出はカウンタを有効化する CEN ビットを
セットします。その後、カウンタと比較値レジスタの値との比較で出力が反転させられま
す。この動作には数クロック・サイクルが必要とされますので、実現可能な遅延時間の最
小値は制限を受けます。

より小さな遅延時間が望まれるときには、TIMx_CCMRx レジスタの OCxFE ビットをセット
することができます。このとき OCxREF (及び OCx) は、比較動作を行うことなく、トリ
ガに応じてレベルが定まります。このときのレベルは比較が一致したときと同じです。
OCxFE はチャネルが PWM1 もしくは PWM2 モードに設定されたときにのみ有効です。

12.3.16 エンコーダ・インターフェース・モード

エンコーダ・インターフェース・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するのであれば SMS=001 に、TI1 エッジのみであれば
SMS=010 に、TI1 と TI2 の両方のエッジをカウントするのであれば SMS=011 とします。

TI1 と TI2 の極性は TIMx_CCER レジスタの CC1P ビットと CC2P ビットで指定できます。
必要ならば、入力フィルタの設定も同じように行います。

二つの入力 TI1 と TI2 はインクリメンタル・エンコーダとのインターフェースのために使用
されます。動作は [表 52](#) を参照してください。カウンタがイネーブル状態 (TIMx_CR1 レジ
スタの CEN ビットが 1) であれば、TI1FP1 もしくは TI2FP2 (フィルタを通し、極性を整
えた TI1 及び TI2 信号) の有効なレベル変化がカウンタのクロックとして使用されます。
両入力の変化の順序が、カウントパルスと同時に方向信号を与えます。変化の順序によっ
てカウンタがカウントアップ、カウントダウンを行うとき、TIMx_CR1 レジスタの DIR
ビットはこれに応じてハードウェアで変更されます。カウントが TI1 のみ、TI2 のみ、TI1
と TI2 の両方のいずれで行われるかにかかわらず、DIR ビットの値は TI1、TI2 いずれかの
変化を検出したときに設定されます。

エンコーダ・インターフェース・モードは、単に方向の選択を含む外部クロック方式として
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 (方向に応じて
0 から ARR、もしくは ARR から 0 の間で) 連続的にカウントを行います。したがって、
TIMx_ARR は他の場合と同じく開始前に設定しておかねばなりません。キャプチャ、比較、
プリスケーラ、繰返しカウンタ、トリガ出力の機能は通常と同じく機能しています。エン
コーダ・モードと外部クロック・モード 2 は競合する点がありますので、同時に設定して
はいけません。

参考資料

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、そしてそのコンテンツに応じて自動的に変更され、その値は常にエンコーダの位置を示しています。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。下の表は TI1 と TI2 の変化の組合せをまとめたものです。(TI1 と TI2 は同時に変化しないと想定しています。)

表 52. カウントの方向とエンコーダ信号

有効エッジ	他方の信号に対するレベル (TI1FP1 for TI2, TI2FP2 for TI1)	TI1FP1 信号		TI2FP2 信号	
		立上り	立下り	立上り	立下り
TI1 のみをカウント	High	Down	Up	カウントなし	カウントなし
	Low	Up	Down	カウントなし	カウントなし
TI2 のみをカウント	High	カウントなし	カウントなし	Up	Down
	Low	カウントなし	カウントなし	Down	Up
TI1 と TI2 をカウント	High	Down	Up	Up	Down
	Low	Up	Down	Down	Up

外部のインクリメンタル・エンコーダは外部にインターフェース回路を持つことなく、MCU に直接に接続されます。ただし、通常はコンパレータがエンコーダの作動出力をデジタル信号に変換するために使用されます。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エンコーダの 3 番目の出力は、外部割込み入力に接続することができ、カウンタのリセットをトリガすることができます。

図 88 はカウント信号の生成と方向制御を含むカウンタの動作を示しています。同時に、両エッジを選択したとき、ジッタがどのように除かれるかも示されています。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ます。この例では、次の設定がなされていることを想定しています。

- 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 (TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1)
- CC2P=0 (TIMx_CCER レジスタ、TI1FP2 非反転、TI1FP2=TI2)
- SMS=011 (TIMx_SMCR レジスタ、両入力で、立上りと立下りの両エッジをカウント)
- CEN=1 (TIMx_CR1 レジスタ、カウンタ・イネーブル)

参考資料

図 88. エンコーダ・インターフェース・モードにおけるカウンタの動作例

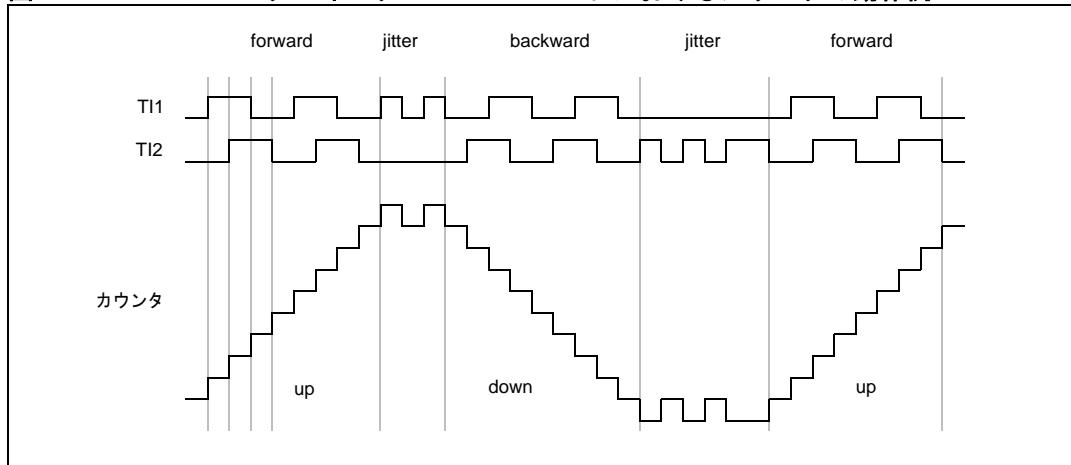
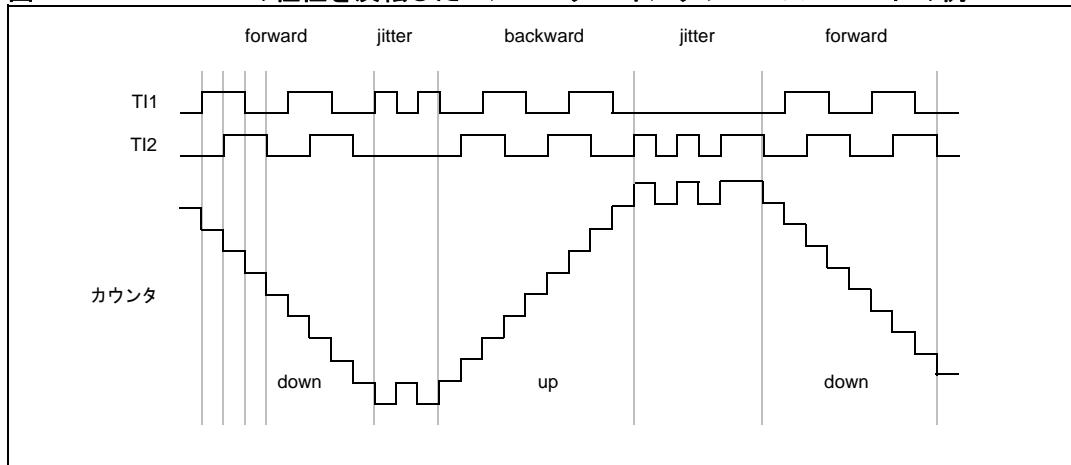


図 89 では、TI1FP1 の極性を反転した時のカウンタの動作を示しています。これは上の例とほぼ同じ設定ですが、CC1P=1 である点が異なっています。

図 89. TI1FP1 の極性を反転したエンコーダ・インターフェース・モードの例



エンコーダ・インターフェース・モードに設定されたタイマはセンサの位置を表示しています。キャプチャ・モードに設定した二番目のタイマを使用し、ふたつのエンコーダイベントの時間差を計ることで、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はこの目的に使用できます。ふたつのイベントの時間差によっては、カウンタを任意の時点で読出することもできます。ハードウェアに余裕があれば、カウンタの値を 3 番目の入力キャプチャ・レジスタにラッチすることができます。(さらに、キャプチャが周期的なときには、キャプチャ信号の発生に別のタイマを使うことも考えられます。) また、リアルタイム・クロックによって発生する DMA リクエストでこの値を読出することもできます。

12.3.17 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットで、チャネル 1 の入力フィルタを、TIMx_CH1、TIMx_CH2 及び TIMx_CH3 を入力とする 3 入力 XOR ゲートの出力に接続することができます。

XOR 出力は、トリガや入力キャプチャといったタイマ入力機能のすべてで利用することができます。この機能の利用の一例は [セクション 12.3.18](#) に記されているホール・センサ・インターフェースです。

12.3.18 ホール・センサ・インターフェース

これは、[図 90](#) で “インターフェース・タイマ” と呼ばれている他の TIMx (TIM2、TIM3、TIM4、または TIM5) タイマや、モータを駆動するための PWM 信号を生成する高機能タイマ (TIM1 または TIM8) を使用することによって行われます。“インターフェース・タイマ” は XOR を通して TI1 入力チャネルに接続されている 3 本のタイマ入力ピン (CC1、CC2、CC3) をキャプチャします。(TIMx_CR2 レジスタの TI1S ビットをセットすることで選択できます。)

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、TI1F_ED をスレーブ入力とします。3 本の入力のいずれかが反転するごとに、カウンタは 0 からのカウントを再開します。これがホール・センサ入力の変化によってトリガされるタイムベースを構成します。

“インターフェース・タイマ” のキャプチャ / 比較チャネル 1 は TRC をキャプチャ信号とするキャプチャ / モードに設定されます。[\(図 73: キャプチャ / 比較チャネル \(例: チャネル 1 入力ステージ\) \(234 ページ\)\)](#) を参照してください。キャプチャされた値はふたつの入力の変化の間の経過時間に対応しており、モータの速度情報を与えます

“インターフェース・タイマ” は、(COM イベントをトリガすることで) 高機能タイマ (TIM1 または TIM8) のチャネルの構成を変更するパルスを生成するために、出力モードで使用されます。TIM1 タイマはモータを駆動する PWM 信号を生成するために使用されます。このために、インターフェース・タイマ・チャネルは所定の時間後に正パルスを生成するよう (出力比較モードもしくは PWM モード) 設定されなければなりません。このパルスは TRGO 出力を通じて 高機能タイマ (TIM1 または TIM8) に送られます。

例 : TIMx に接続されているホール・センサ入力が変化するごとに、所定の時間が経過した後に高機能タイマ TIM1 の PWM 設定を変更するとします。

- 3 本のタイマ入力を TI1 入力チャネルに OR 接続します。このためには、TIMx_CR2 レジスタの TI1S に 1 を書込みます。
- タイムベースを設定します。TIMx_ARR に最大値を書き込み、TI1 の変化でカウンタがクリアされるようにします。プリスケーラはセンサのふたつの変化間の時間より長いカウント時間が得られるよう設定します。
- チャネル 1 をキャプチャ・モード (TRC 選択) とします。このために TIMx_CCMR1 レジスタの CC1S ビットに 01 を書込みます。必要ならば、デジタル・フィルタの設定も行います。
- チャネル 2 を PWM モード 2 とし、所定の遅延時間を設定します。このために、TIM1_CCMR1 レジスタの OC2M ビットに 111 を、CC2S ビットに 00 を書込みます。
- OC2REF を TRGO のトリガ出力とします。このために、TIMx_CR2 レジスタの MMS ビットに 101 を書込みます。

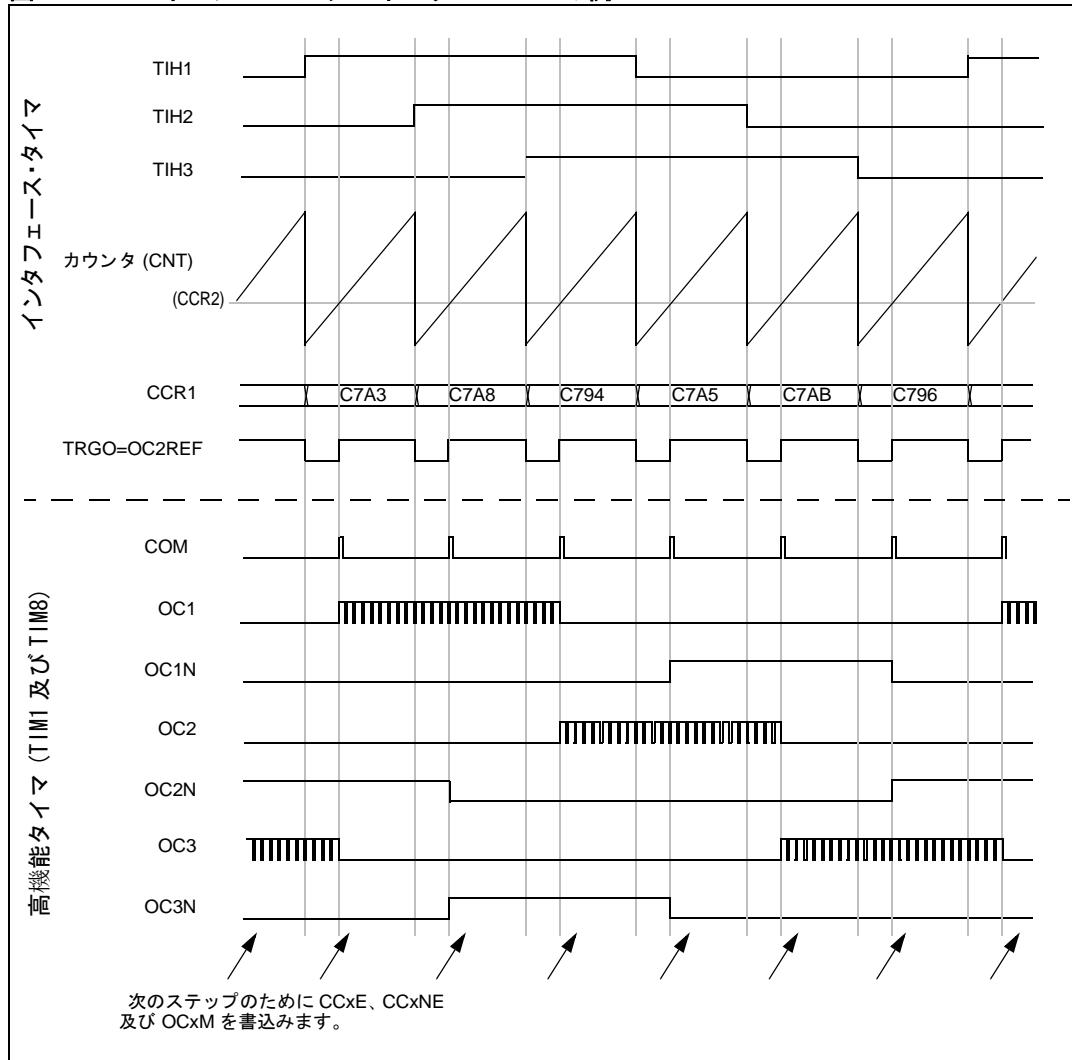
高機能タイマ TIM1 で、トリガ入力として適切な ITR 入力を選択し、タイマが PWM 信号を生成するよう設定し、(TIM1_CR2 レジスタの CCPC=1 で) キャプチャ / 比較制御信号がプリロードされ、(TIM1_CR2 レジスタの CCUS=1 で) COM イベントがトリガ入力で制御されるように設定します。PWM 制御ビット CCxE 及び OCxM は、COM イベントが発生した

参考資料

後に次のステップのために書き込まれます。(これは OC2REF の立上りエッジで生じる割込み処理ルーチンで行うことができます。)

図 90 には、この例の動作が図示されています。

図 90. ホール・センサ・インターフェースの例



12.3.19 TIMx と外部トリガ同期

TIMx タイマは、リセット・モード、ゲート・モード、トリガ・モードといった幾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

トリガ入力イベントによって、カウンタとそのプリスケーラの再初期化を行うことができます。さらに、 $\text{TIM}_x\text{_CR1}$ レジスタの URS ビットがローレベルであれば、更新イベントが発生します。これによりプリロード機能をもつレジスタ、 $\text{TIM}_x\text{_ARR}$ と $\text{TIM}_x\text{_CCR}_x$ が更新されます。

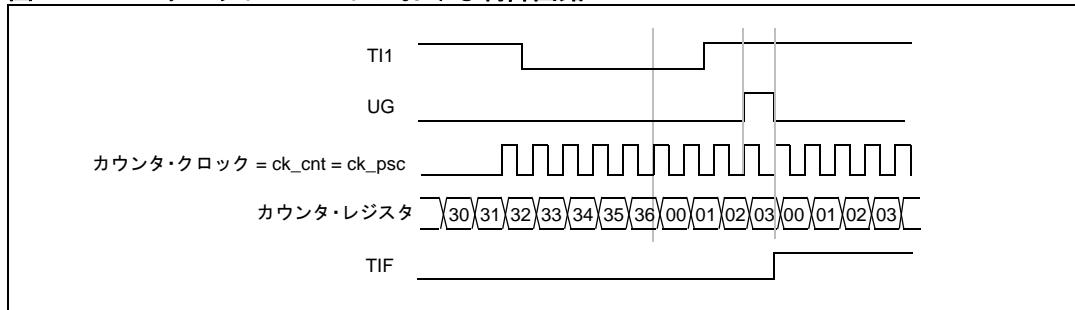
次の例では、TI1 入力の立上りエッジに応じて、アップカウンタがクリアされます。

- チャネル 1 が TI1 の立上りエッジを検出するように設定します。入力フィルタの応答時間を設定します。(この例ではフィルタ機能は必要ありません。IC1F=0000 としておきます。) トリガによる動作ではキャプチャ・プリスケーラは使用されませんので、設定の必要はありません。CC1S ビットは入力キャプチャの信号を選択するのみで、 $\text{TIM}_x\text{_CCMR1}$ レジスタの CC1S=01 とします。トリガの極性(立上りエッジのみを検出)を設定するために、 $\text{TIM}_x\text{_CCER}$ レジスタの CC1P=0 とします。
- $\text{TIM}_x\text{_SMCR}$ レジスタの SMS=100 で、タイマをリセット・モードとします。 $\text{TIM}_x\text{_SMCR}$ レジスタの TS=101 で入力信号として TI1 を選択します。
- $\text{TIM}_x\text{_CR1}$ レジスタの CEN=1 により、カウンタをスタートします。

カウンタは内部クロックでカウントを開始し、TI1 の立上りエッジが生じるまで通常の動作を行います。TI1 が立ち上がったとき、カウンタはクリアされ、0 からカウントが再開します。同時に、トリガフラグ ($\text{TIM}_x\text{_SR}$ レジスタの TIF ビット) がセットされ、割込みリクエスト、もしくは DMA リクエストが発生します。 $(\text{TIM}_x\text{_DIER}$ レジスタの TIE もしくは TDE ビットで許可されている場合。)

次の図は自動再ロード・レジスタ $\text{TIM}_x\text{_ARR}=0x36$ の場合の動作を示しています。TI1 の立上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 入力がクロックに同期させられることに起因します。

図 91. リセット・モードにおける制御回路



参考資料

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、カウンタが動作します。

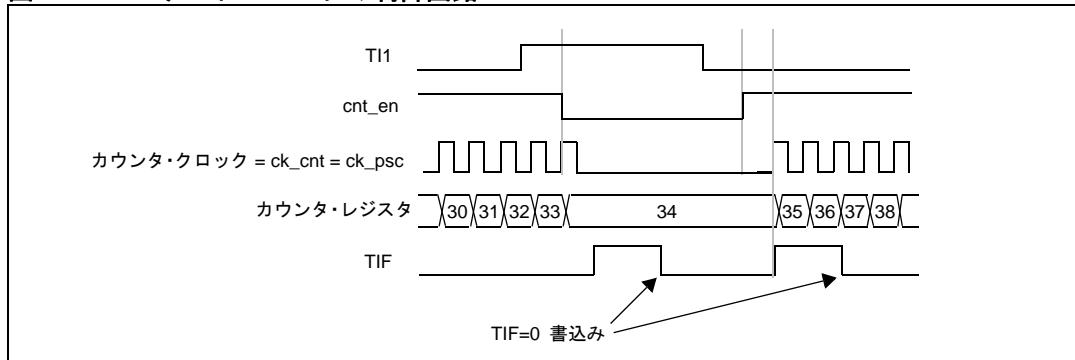
以下の例では、TI1 入力がロー・レベルのときにのみアップカウンタがカウント動作を行います。

- チャネル 1 で TI1 のロー・レベルを検出するよう設定します。入力フィルタの応答時間を設定します。(この例ではフィルタは必要ありませんので IC1F=0000 のままとします。) キャプチャ・プリスケーラはトリガでは使用されませんので、設定の必要はありません。TIMx_CCMR1 レジスタの CC1S ビットは CC1S=01 とし、入力キャプチャの信号を選択します。TIMx_CCER レジスタの CC1P=1 を書き込み、極性 (ロー・レベルのみを検出) を有効とします。
- タイマをゲート・モードに設定します。このためには、TIMx_SMCR レジスタで SMS=101 とします。TIMx_SMCR レジスタの TS=101 で TI1 を入力信号として選択します。
- TIMx_CR1 レジスタの CEN=1 により、カウンタをイネーブルとします。(ゲート・モードではトリガ入力のレベルにかかわらず、CEN=0 ではカウンタはスタートしません。)

カウンタは TI1 がロー・レベルの時にカウントを開始し、TI1 がハイ・レベルになると停止します。カウンタのスタート時とストップ時の両方で TIMx_SR レジスタの TIF フラグがセットされます。

TI1 の立上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 入力がクロックに同期させられることに起因します。

図 92. ゲート・モードの制御回路



参考資料

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応じて、カウンタがスタートします。

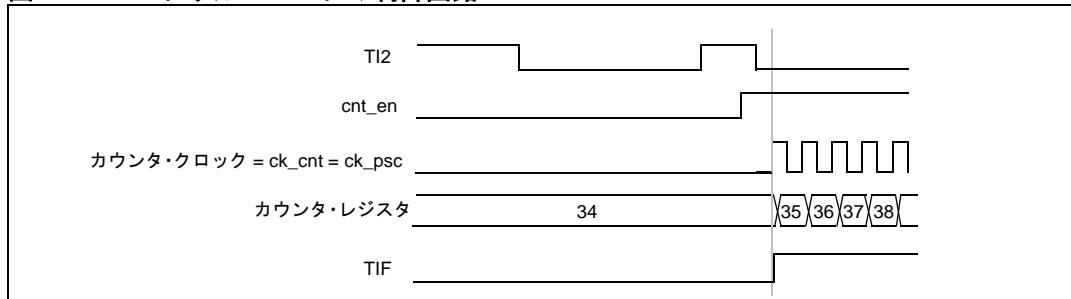
次の例では、TI2 入力の立上りエッジに応じて、アップカウンタがスタートします。

- チャネル 2 で TI2 の立上りエッジを検出するよう設定します。入力フィルタの応答時間を設定します。(この例ではフィルタは必要ありませんので IC2F=0000 のままとします。) キャプチャ・プリスケーラはトリガでは使用されませんので、設定の必要はありません。TIMx_CCMR1 レジスタの CC2S ビットは CC2S=01 とし、入力キャプチャの信号を選択します。TIMx_CCER レジスタの CC2P=1 を書き込み、極性（ロー・レベルをのみを検出）を有効とします。
- タイマをトリガ・モードに設定します。このためには、TIMx_SMCR レジスタで SMS=110 とします。TIMx_SMCR レジスタの TS=110 で TI2 を入力信号として選択します。

TI2 の立上りエッジでカウンタは内部クロックでカウントを開始し、同時に TIF フラグをセットします。

TI2 の立上りエッジから実際にカウンタがスタートするまでの遅延は、TI2 入力がクロックに同期させられることに起因します。

図 93. 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（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）使用することができます。このとき、ETR 信号は外部クロック入力として使用され、別の入力がトリガ入力（リセット・モード、ゲート・モード、及びトリガ・モード）として使用されます。TIMx_SMCR レジスタの TS ビットの設定では、TRGI として ETR を選択しないようにしてください。

以下の例では、TI1 の立上りエッジが発生したとき、アップカウンタは ETR 信号の立上りエッジによるカウントアップを開始します。

- TIMx_SMCR レジスタで次の設定を行い、外部トリガ入力回路の構成を行います。
 - ETF = 0000: フィルタなし
 - ETPS=00: プリスケーラは無効
 - ETP=0: ETR の立上りエッジを検出。ECE=1 で外部クロック・モード 2 を有効とします。

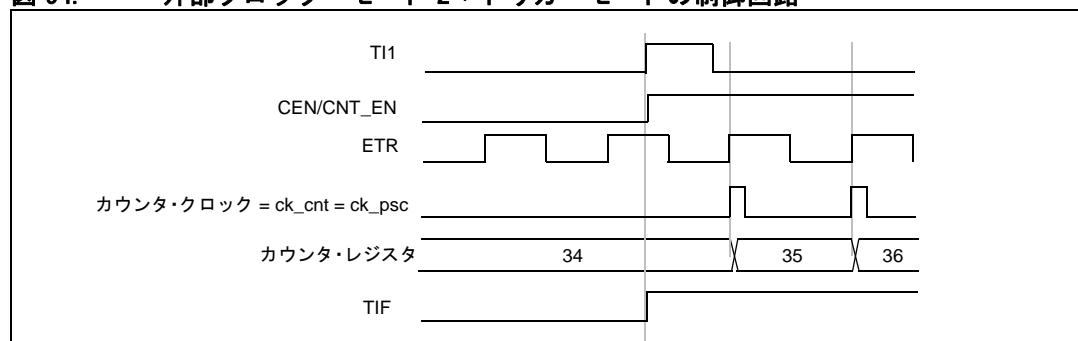
参考資料

2. TI1 の立上りエッジを検出するため、チャネル 1 で次の設定を行います。
 - IC1F=0000: フィルタなし
 - トリガに関してはキャプチャ・プリスケーラは使用されませんので設定は不要です。
 - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャ信号を選択します。
 - TIMx_CCER レジスタの CC1P=0 で、極性を有効とします。(同時に、立上りエッジの選択を指示します。)
3. タイマをトリガ・モードとするため、TIMx_SMCR レジスタの SMS=110 とします。入力として TI1 を選択するため、TIMx_SMCR レジスタの TS=101 を設定します。

TI1 の立上りエッジでカウンタがイネーブルとなり、TIF フラグがセットされます。カウンタは ETR の立上りエッジをカウントします。

ETR 信号の立上りエッジから実際にカウンタのリセットまでの遅延は、ETRP 入力がクロックに同期させられることに起因します。

図 94. 外部クロック・モード 2 + トリガ・モードの制御回路



12.3.20 タイマ間の同期

TIM タイマは同期もしくは連結のために、内部で互いに接続することができます。これに関する情報は [セクション 13.3.15: タイマ間の同期 \(314 ページ\)](#) で説明されています。

12.3.21 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 (Cortex-M3 コアは停止状態) に入ったとき、TIM1 カウンタが通常どおりに動作するか、停止するかを、DBG モジュールの DBG_TIM1_STOP 構成ビットで指定することができます。詳細は [セクション 26.15.2: タイマ、ウォッチドッグ、bxCAN と PCに対するデバッグ・サポート](#) に記載されています。

參考資料

12.4 TIM1 及び TIM8 レジスタ

レジスタの詳細記述で使用されている略語については [セクション 1.1 \(32 ページ\)](#) を参照してください。

12.4.1 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

Bits 15:10 予約済み、常に 0 が読出されます。

Bits 9:8 CKD[1:0]: クロック分周

これらのビットは、タイマ・クロック (CK_INT) 周波数と、デッドタイム、デッドタイム生成回路で使用されるサンプリング・クロック (t_{DTS}) 及びデジタル。フィルタ (ETR, TIx) との間の分周比を指定します。

- 00: $t_{DTS} = t_{CK_INT}$
 - 01: $t_{DTS} = 2*t_{CK_INT}$
 - 10: $t_{DTS} = 4*t_{CK_INT}$
 - 11: 約約値、設定できません。

Bit 7 ARPE: 自動再ロード・プリロード・イネーブル

- 0: TIM1_ARR レジスタはバッファを使用しません。
 - 1: TIM1_ARR レジスタはバッファを使用します。

Bits 6:5 CMS[1:0]: センターライン・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 / カウントダウンを行います。

01: センターライン・モード 1、カウンタは交互にアップ / ダウンを行います。出力設定 (TIM1_CCMRx レジスタの CCxS=00) の各チャネルの出力比較割込みリクエスト・フラグはカウントダウンの時にのみセットされます。

10: センターライン・モード 2、カウンタは交互にアップ / ダウンを行います。出力設定 (TIM1_CCMRx レジスタの CCxS=00) の各チャネルの出力比較割込みリクエスト・フラグはカウントアップの時にのみセットされます。

11: センターライン・モード 3、カウンタは交互にアップ / ダウンを行います。出力設定 (TIM1_CCMRx レジスタの CCxS=00) の各チャネルの出力比較割込みリクエスト・フラグはカウントアップ / カウントダウンの両方でセットされます。

注: カウンタがイネーブル状態 (CEN=1) のとき、エッジアライン・モードからセンターアライン・モードに切替えることはできません。

Bit 4 DIR: 方向

- 0: カウンタはカウントアップを行います。
1: カウンタはカウントダウンを行います。

注: このビットはセンター・ライン・モードもしくはエンコーダ・モードに設定されているときには読み出し専用となります。

Bit 3 OPM: 単一パルス・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動作を停止（CEN ビットをクリア）します。

参考資料

Bit 2 URS: 更新リクエストの要因

更新イベント (UEV) を発生する元となる信号を選択するため、ソフトウェアでセット / クリアができます。

0: 次のイベントのいずれかが更新割込み、もしくは DMA リクエストを発生します。(リクエストが許可されているときのみ)

-カウンタ・オーバフロー / アンダフロー

-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指示

1: カウンタのオーバフロー / アンダフローのみが更新割込み、もしくは DMA リクエストを発生します。(リクエストが許可されているときのみ)

Bit 1 UDIS: 更新動作ディセーブル

更新イベント (UEV) 発生をイネーブル / ディセーブルとするため、ソフトウェアでセット / クリアができます。

0: UEV イネーブル。更新イベント (UEV) は次のいずれかで発生します。

-カウンタ・オーバフロー / アンダフロー

-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指示

このとき、バッファを持つレジスタではプリロード値がロードされます。

1: UEV ディセーブル。更新イベントは発生せず、個々のシャドウ・レジスタ (ARR, PSC, CCRx) がそれぞれの値を保持しています。カウンタとプリスケーラは UG ビットがセットされたとき、及びスレーブ・モード・コントローラからハードウェアリセットが指示されたときに初期化されます。

Bit 0 CEN: カウンタ・イネーブル

0: カウンタ・ディセーブル

1: カウンタ・イネーブル

注: 外部クロック、ゲート・モード及びエンコーダ・モードは CEN ビットが事前にソフトウェアでセットされているときのみ動作します。トリガ・モードではハードウェアで自動的に CEN ビットをセットします。

12.4.2 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	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S		MMS[2:0]	CCDS	CCUS	予約済み	CCPC	Res.	RW

Bit 15 予約済み、常に 0 が読出されます。

Bit 14 OIS4: 出力アイドル状態 4 (OC4 出力)

OIS1 ビットの説明を参照してください。

Bit 13 OIS3N: 出力アイドル状態 3 (OC3N 出力)

OIS1N ビットの説明を参照してください。

Bit 12 OIS3: 出力アイドル状態 3 (OC3 出力)

OIS1 ビットの説明を参照してください。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

Bit 11 **OIS2N**: 出力アイドル状態 2 (OC2N 出力)

OIS1N ビットの説明を参照してください。

Bit 10 **OIS2**: 出力アイドル状態 2 (OC2 出力)

OIS1 ビットの説明を参照してください。

Bit 9 **OIS1N**: 出力アイドル状態 1 (OC1N 出力)

0: MOE=0 の時、デッドタイム後に OC1N=0

1: MOE=0 の時、デッドタイム後に OC1N=1

注: このビットは LOCK レベル 1、2 もしくは 3 が指定されているときには変更できません。
(TIM1_BKR レジスタの LOCK ビット)

Bit 8 **OIS1**: 出力アイドル状態 1 (OC1 出力)

0: MOE=0 の時 OC1=0 (OC1N がセットされていればデッドタイム後に)

1: MOE=0 の時 OC1=1 (OC1N がセットされていればデッドタイム後に)

注: このビットは LOCK レベル 1、2 もしくは 3 が指定されているときには変更できません。
(TIM1_BKR レジスタの LOCK ビット)

Bit 7 **TI1S**: TI1 選択

0: TIM1_CH1 ピンが TI1 入力に接続されます。

1: TIM1_CH1、CH2 及び CH3 ピンが TI1 入力に接続されます。(XOR 接続)

Bits 6:4 **MMS[1:0]**: 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・モードにおいてスレーブ・タイマに送られる情報 (TRGO) の選択を行います。

000: **Reset** – TIM1_EGR レジスタの UG ビットがトリガ出力 (TRGO) として使用されます。トリガ入力によってリセットが発生したとき (スレーブ・モード・コントローラがリセット・モードに設定されているとき) TRGO 信号は実際のリセットから遅れて発生します。

001: **Enable** – 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。これは幾つかのタイマを同時にスタートさせるときや、スレーブタイマの動作範囲を制御する時に有効です。カウンタ・イネーブル信号は、CEN 制御ビットとゲート・モードに設定されているときのトリガ入力との論理和で発生します。マスタ・スレーブ・モードに設定されているときを除いて TRGO 出力には遅延が存在します。(TIM1_SMCR レジスタの MSM ビットの説明をお読みください。)

010: **Update** – 更新イベントがトリガ出力 (TRGO) として選択されます。例えばマスタタイマがスレーブタイマのプリスケーラとなるときに使用されます。

011: **Compare Pulse** – キャプチャもしくは比較一致で CC1F フラグのセット条件が発生したとき (フラグがすでにハイレベルであっても)、正のパルスがトリガ出力 (TRGO) から送出されます。

100: **Compare** – OC1REF 信号がトリガ出力 (TRGO) として使用されます。

101: **Compare** – OC2REF 信号がトリガ出力 (TRGO) として使用されます。

110: **Compare** – OC3REF 信号がトリガ出力 (TRGO) として使用されます。

111: **Compare** – OC4REF 信号がトリガ出力 (TRGO) として使用されます。

Bit 3 **CCDS**: キャプチャ / 比較 DMA 選択

0: CCx イベント発生時に CCx DMA リクエストが送出されます。

1: 更新イベントの発生時に CCx DMA リクエストが送出されます。

Bit 2 **CCUS**: キャプチャ / 比較・制御更新選択

0: キャプチャ / 比較制御ビットは、プリロード機能を使用している (CCPC=1) 場合、COM ビットがセットされたときにのみ更新されます。

1: キャプチャ / 比較制御ビットは、プリロード機能を使用している (CCPC=1) 場合、COM ビットがセットされたときもしくは TRGI の立上りエッジで更新されます。

注: このビットは相補出力を有するチャネルに対してのみ有効に働きます。

Bit 1 予約済み、常に 0 が読出されます。



参考資料

高機能制御タイマ (TIM1 及び TIM8)

RM0008

Bit 0 CCPC: キャプチャ / 比較・プリロード制御

0: CCxE、CCxNE 及び OCxM ビットはプリロードされません。

1: CCxE、CCxNE 及び OCxM ビットはプリロード機能が有効で、書込み後に COM ビットがセットされたときに値が更新されます。

注: このビットは相補出力を有するチャネルに対してのみ有効に働きます。

12.4.3 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]			

Bit 15 ETP: 外部トリガ極性

このビットは外部トリガ動作に ETR と ETR のどちらかが使用されるかを選択します。

0: ETR は反転されず、ハイレベルもしくは立上りエッジがアクティブとなります。

1: ETR を反転し、ローレベルもしくは立下りエッジがアクティブとなります。

Bit 14 ECE: 外部クロック有効

このビットは外部クロック・モード 2 を有効とします。

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。ETRF 信号のアクティブエッジがカウンタのクロックとなります。

注 1: ECE ビットをセットすると、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) と同じ効果となります。

注 2: 外部クロック・モード 2 と同時に次のスレーブ・モードを指定できます。リセット・モード、ゲート・モード、トリガ・モード。ただし、TRGI は ETRF に接続されていてはいけません。(TS ビットは 111 でないことが必要。)

注 3: 外部クロック・モード 1 と外部クロック・モード 2 が同時に有効に設定されると、外部クロック入力は ETRF となります。

Bits 13:12 ETPS[1:0]: 外部トリガ・プリスケーラ

外部トリガ信号 ETRP の周波数は最大で TIMxCLK 周波数の 1/4 に制限されます。ETRP の周波数を低下させるためにプリスケーラが使用でき、高速の外部クロックが与えられる場合に役立ちます。

00: プリスケーラ・オフ

01: ETRP 分周比 2

10: ETRP 分周比 4

11: ETRP 分周比 8

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

Bits 11:8 ETF[3:0]: 外部トリガフィルタ

これらのビットは 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・フィルタの応答時間を定めます。デジタル・フィルタはイベントカウンタを主体としており、出力に有効な変化をもたらすには N 回のイベント発生が必要とされます。

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: $f_{SAMPLING} = f_{DTS}/32$ 、 $N=8$

Bit 7 MSM: マスター / スレーブ・モード

0: 無効

1: トリガ入力 (TRGI) は、タイマとそのスレーブ (TRGO に接続) との間で完全な同期をとるため、ある程度の遅延をおいて有効となります。これは複数のタイマを 1 個の外部イベントで同期する際に役立ちます。

Bits 6:4 TS[2:0]: トリガ選択

これらのビットはカウンタを同期させるため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: フィルタ後のタイマ入力 1 (TI1FP1)

110: フィルタ後のタイマ入力 2 (TI2FP2)

111: 外部トリガ入力 (ETRF)

各タイマの ITRx の意味についての詳細は [表 53: TIMx 内部トリガ接続 \(264 ページ\)](#) を参照してください。

注: 設定変更時に誤ったエッジ検出が行われるのを防ぐため、これらのビットの変更は、これが使用されていないとき（例えば SMS=000）に限られます。

Bit 3 予約済み、常に 0 が読出されます。



参考資料

Bits 2:0 SMS: スレーブ・モード選択

外部信号が選択されると、その信号の指定された極性が、トリガ信号 (TRGI) のアクティブエッジとなります。

000: スレーブ・モード・ディセーブル - CEN = 1 であれば、内部クロックが直接にプリスケーラの入力となります。

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 / カウントダウンを行います。

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 / カウントダウンを行います。

011: エンコーダ・モード 3 - カウンタは TI1FP1 のエッジと TI2FP2 のエッジの両方でカウントを行います。アップ / ダウンはエッジ検出時の他方の信号レベルで決められます。

100: リセット・モード - 選択されたトリガ入力 (TRGI) の立上りエッジがカウンタをリセットし、レジスタの更新イベントを発生します。

101: ゲート・モード - トリガ入力 (TRGI) がハイレベルの時、カウンタにクロックが供給されます。トリガ信号がローレベルとなるとカウンタは停止します。(リセットはされません。) カウンタのスタートとストップの両方の制御ができます。

110: トリガ・モード - トリガ入力 (TRGI) の立上りエッジでカウンタがスタートします。(このときリセットはされません。) カウンタのスタートのみが制御されます。

111: 外部クロックモード 1 - 選択された信号 (TRGI) の立上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 (TS=100) ときには、ゲート・モードを選択することはできません。TI1F_ED は TI1F の信号変化ごとに 1 パルスを出力します。その一方で、ゲート・モードはトリガ信号のレベルに依存した動作を行います。

表 53. TIMx 内部トリガ接続

Slave TIM	ITR0 (TS = 000)	ITR1 (TS = 001)	ITR2 (TS = 010)	ITR3 (TS = 011)
TIM1	TIM5	TIM2	TIM3	TIM4
TIM8	TIM1	TIM2	TIM4	TIM5

12.4.4 DMA / 割込みリクエスト許可レジスタ (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	COMD E	CC4D E	CC3D E	CC2D E	CC1D E	UDE	BIE	TIE	COMI E	CC4IE	CC3IE	CC2IE	CC1IE	UIE

Bit 15 予約済み、常に 0 が読出されます。

Bit 14 TDE: トリガ DMA リクエスト許可

- 0: トリガ DMA リクエスト禁止
- 1: トリガ DMA リクエスト許可

Bit 13 COMDE: COM DMA リクエスト許可

- 0: COM DMA リクエスト禁止
- 1: COM DMA リクエスト許可

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

Bit 12 **CC4DE:** キャプチャ / 比較 4 DMA リクエスト許可

- 0: CC4 DMA リクエスト禁止
- 1: CC4 DMA リクエスト許可

Bit 11 **CC3DE:** キャプチャ / 比較 3 DMA リクエスト許可

- 0: CC3 DMA リクエスト禁止
- 1: CC3 DMA リクエスト許可

Bit 10 **CC2DE:** キャプチャ / 比較 2 DMA リクエスト許可

- 0: CC2 DMA リクエスト禁止
- 1: CC2 DMA リクエスト許可

Bit 9 **CC1DE:** キャプチャ / 比較 1 DMA リクエスト許可

- 0: CC1 DMA リクエスト禁止
- 1: CC1 DMA リクエスト許可

Bit 8 **UDE:** 更新 DMA リクエスト許可

- 0: 更新 DMA リクエスト禁止
- 1: 更新 DMA リクエスト許可

Bit 7 **BIE:** ブレーク割込みリクエスト許可

- 0: ブレーク割込みリクエスト禁止
- 1: ブレーク割込みリクエスト許可

Bit 6 **TIE:** トリガ割込みリクエスト許可

- 0: トリガ割込みリクエスト禁止
- 1: トリガ割込みリクエスト許可

Bit 5 **COMIE:** COM 割込みリクエスト許可

- 0: COM 割込みリクエスト禁止
- 1: COM 割込みリクエスト許可

Bit 4 **CC4IE:** キャプチャ / 比較 4 割込みリクエスト許可

- 0: CC4 割込みリクエスト禁止
- 1: CC4 割込みリクエスト許可

Bit 3 **CC3IE:** キャプチャ / 比較 3 割込みリクエスト許可

- 0: CC3 割込みリクエスト禁止
- 1: CC3 割込みリクエスト許可

Bit 2 **CC2IE:** キャプチャ / 比較 2 割込みリクエスト許可

- 0: CC2 割込みリクエスト禁止
- 1: CC2 割込みリクエスト許可

Bit 1 **CC1IE:** キャプチャ / 比較 1 割込みリクエスト許可

- 0: CC1 割込みリクエスト禁止
- 1: CC1 割込みリクエスト許可

Bit 0 **UIE:** 更新割込みリクエスト許可

- 0: 更新割込みリクエスト禁止
- 1: 更新割込みリクエスト許可



参考資料

12.4.5 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 Res.	CC4OF rc_w0	CC3OF rc_w0	CC2OF rc_w0	CC1OF rc_w0	予約済み Res.	BIF rc_w0	TIF rc_w0	COMIF rc_w0	CC4IF rc_w0	CC3IF rc_w0	CC2IF rc_w0	CC1IF rc_w0	UIF rc_w0		

Bit 15:13 予約済み、常に 0 が読み出されます。

Bit 12 **CC4OF**: キャプチャ / 比較 4 オーバキャプチャ・フラグ

CC1OF の説明を参照してください。

Bit 11 **CC3OF**: キャプチャ / 比較 3 オーバキャプチャ・フラグ

CC1OF の説明を参照してください。

Bit 10 **CC2OF**: キャプチャ / 比較 2 オーバキャプチャ・フラグ

CC1OF の説明を参照してください。

Bit 9 **CC1OF**: キャプチャ / 比較 1 オーバキャプチャ・フラグ

このフラグのセットは、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に、ハードウェアで行われます。フラグのクリアはソフトウェアでこのビットに 0 を書込むことで行います。

0: オーバキャプチャ状況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にもかかわらず、カウンタの値が TIM1_CCR1 レジスタに書込まれました。

Bit 8 予約済み、常に 0 が読み出されます。

Bit 7 **BIF**: ブレーク割込みリクエスト・フラグ

このフラグはブレーク入力がアクティブとなったときにハードウェアでセットされます。フラグのクリアはブレーク入力がアクティブでない時に、ソフトウェアで行います。

0: ブレークイベントは発生していません。

1: ブレーク入力でアクティブルベルが検出されました。

Bit 6 **TIF**: トリガ割込みリクエスト・フラグ

このフラグはトリガ・イベント時（スレーブ・モード・コントローラのゲート・モード以外の全モードで TRGI 入力でアクティブエッジが検出されたとき、もしくはゲート・モードでいずれかのエッジが検出されたとき）にハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: トリガ・イベントは発生していません。

1: トリガ割込みリクエストがペンディング状態です。

Bit 5 **COMIF**: COM 割込みリクエスト・フラグ

このフラグは COM イベント時（キャプチャ / 比較制御ビット - CCxE, CCxNE, OCxM - が更新された時）にハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: COM イベントは発生していません。

1: COM 割込みリクエストはペンディング状態です。

Bit 4 **CC4IF**: キャプチャ / 比較 4 割込みリクエスト・フラグ

CC1IF の説明を参照してください。

Bit 3 **CC3IF**: キャプチャ / 比較 3 割込みリクエスト・フラグ

CC1IF の説明を参照してください。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

Bit 2 **CC2IF**: キャプチャ / 比較 2 割込みリクエスト・フラグ

CC1IF の説明を参照してください。

Bit 1 **CC1IF**: キャプチャ / 比較 1 割込みリクエスト・フラグ

チャネル CC1 が出力設定のとき :

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でセットされます。(センター・ライン・モードでは、例外状況もあります。TIM1_CR1 レジスタの CMS ビットの説明を参照してください。) フラグのクリアはソフトウェアで行います。

0: 一致なし。

1: カウンタ TIM1_CNT の内容が TIM1_CCR1 レジスタの内容と一致しました。

チャネル CC1 が入力設定のとき :

このフラグは、キャプチャが行われたときにハードウェアでセットされます。フラグはソフトウェアで TIM1_CCR1 レジスタを読込むことでクリアされます。

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 TIM1_CCR1 レジスタにキャプチャされました。(IC1 で所定のエッジが検出されました。)

Bit 0 **UIF**: 更新割込みリクエスト・フラグ

このビットは更新イベント発生時にハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: 更新は発生していません。

1: 更新割込みリクエストがペンディング状態です。このビットはレジスタが更新されたときにハードウェアでセットされます。

- TIM1_CR1 レジスタの UDIS=0 で、繰返しダウンカウンタに関してオーバフローもしくはアンダーフローが検出されたとき。(REP_CNT=0 で更新イベントが発生します。)

- TIM1_CR1 レジスタの URS=0 かつ UDIS=0 で、TIM1_EGR レジスタの UG ビットを使用し、ソフトウェアで CNT の再初期化が行われたとき。

- TIM1_CR1 レジスタの URS=0 かつ UDIS=0 で、トリガ・イベントで CNT が再初期化されたとき。(セクション 12.4.3: スレーブ・モード制御レジスタ (TIMx_SMCR) を参照してください。)

12.4.6 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								BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG

Bits 15:8 予約済み、常に 0 が読み出されます。

Bit 7 **BG**: ブレークイベント生成

このビットをセットすることで、ソフトウェアでイベントを発生させることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: ブレークイベントが発生し、MOE ビットがクリアされ、EIF フラグがセットされます。関連する許可ビットがセットされていれば、割込みリクエストや DMA リクエストが発生します。



参考資料

Bit 6 TG: トリガ・イベント生成

このビットをセットすることで、ソフトウェアでイベントを発生させることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: TIM1_SR レジスタの TIF フラグがセットされます。関連する許可ビットがセットされていれば、割込みリクエストや DMA リクエストが発生します。

Bit 5 COM: キャプチャ / 比較制御更新イベント生成

このビットはソフトウェアでセットすることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: CCPC ビットがセットされている場合、CCxE、CCxNE 及び OCxM ビットを更新します。

注: このビットは相補出力を有するチャネルでのみ有効に働きます。

Bit 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。

Bit 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。

Bit 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。

Bit 1 CC1G: キャプチャ / 比較 1 イベント生成

このビットをセットすることで、ソフトウェアでイベントを発生させることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: チャネル 1 でキャプチャ / 比較イベントが発生します。

CC1 チャネルが出力設定のとき:

CC1IF フラグがセットされます。関連する許可ビットがセットされていれば、割込みリクエストや DMA リクエストが発生します。

CC1 チャネルが入力設定のとき:

その時点のカウンタの値が TIM1_CCR1 レジスタに保存されます。CC1IF フラグがセットされ、関連する許可ビットがセットされていれば、割込みリクエストや DMA リクエストが発生します。

CC1IF フラグがすでにハイレベルであれば、CC1OF フラグがセットされます。

Bit 0 UG: 更新イベント生成

このビットはソフトウェアでセットすることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: カウンタを初期化し、レジスタの更新を行います。プリスケーラカウンタもクリアされます。
(プリスケーラの設定値は変化しません。) センタアライン・モードもしくは DIR=0 (カウントアップ) が選択されているときには、カウンタの値はクリアされます。DIR=1 (カウントダウン) であれば、自動再ロード・レジスタ (TIM1_ARR) の値がカウンタにロードされます。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

12.4.7 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

各チャネルは入力（キャプチャ・モード）もしくは出力（比較モード）で使用することができます。チャネルの方向は対応する CC_xS ビットで定まります。このレジスタの他のビットは入力モードと出力モードで異なる意味をもちます。ビット名が OC_{xx} のビットはチャネルが出力設定のときに有効で、IC_{xx} のビットは入力設定のときに有効です。同じビットが入力モードと出力モードで別の役割を果たしている点に、くれぐれも注意してください。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M[2:0]			OC2 PE	OC2 FE	CC2S[1:0]	OC1 CE	OC1M[2:0]			OC1 PE	OC1 FE	CC1S[1:0]		
	IC2F[3:0]		IC2PSC[1:0]				IC1F[3:0]			IC1PSC[1:0]					
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

出力比較モード :

Bit 15 OC2CE: 出力比較 2 クリア・イネーブル

Bits 14:12 OC2M[2:0]: 出力比較 2 モード

Bit 11 OC2PE: 出力比較 2 プリロード・イネーブル

Bit 10 OC2FE: 出力比較 2 高速イネーブル

Bits 9:8 CC2S[1:0]: キャプチャ / 比較 2 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC2 チャネルは出力設定です。

01: CC2 チャネルは入力設定で、IC2 には TI2 が選択されます。

10: CC2 チャネルは入力設定で、IC2 には TI1 が選択されます。

11: CC2 チャネルは入力設定で、IC2 には TRGI が選択されます。これは TS ビット (TIM1_SMCR レジスタ) で内部トリガ入力が選択されている時にのみ正しく働きます。

注: CC2S ビットはチャネルが OFF (TIM1_CCER レジスタで CC2E=0) のときにのみ書き込みができます。

Bit 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。



参考資料

Bits 6:4 OC1M: 出力比較 1 モード

これらのビットは OC1 及び OC1N の元になる出力基準信号 OC1REF 挑動を規定します。OC1 及び OC1N のアクティブレベルが CC1P 及び CC1NP ビットに依存するのに対し、OC1REF はアクティブ・ハイの信号です。

000: 凍結 – 出力比較レジスタ TIM1_CCR1 とカウンタ TIM1_CNT との間の比較結果は出力に影響しません。

001: 一致時にアクティブレベル – カウンタ TIM1_CNT がキャプチャ / 比較レジスタ TIM1_CCR1 と一致したとき、OC1REF 信号はハイレベルとされます。

010: 一致時にインアクティブレベル – カウンタ TIM1_CNT がキャプチャ / 比較レジスタ TIM1_CCR1 と一致したとき、OC1REF 信号はローレベルとされます。

011: 反転 – TIM1_CNT=TIM1_CCR1 のとき、OC1REF は反転します。

100: インアクティブレベル – OC1REF はローレベルとなります。

101: アクティブレベル – OC1REF はハイレベルとなります。

110: PWM モード 1 – カウントアップ時、 $TIM1_CNT < TIM1_CCR1$ であればチャネル 1 はアクティブに、これ以外はインアクティブになります。カウントダウン時は、 $TIM1_CNT > TIM1_CCR1$ であればインアクティブに、これ以外はアクティブになります。

111: PWM モード 2 – カウントアップ時、 $TIM1_CNT < TIM1_CCR1$ であればチャネル 1 はインアクティブに、これ以外はアクティブになります。カウントダウン時は、 $TIM1_CNT > TIM1_CCR1$ であればアクティブに、これ以外はインアクティブになります。

注 1: LOCK レベル 3 が指定されており (TIM1_BDTR レジスタの LOCK ビット) CC1S=00 (チャネルは出力設定) の場合には、これらのビットは変更できません。

注 2: PWM モード 1 もしくは 2 では、比較結果が変化したとき、もしく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時にのみ、OC1REF のレベルが変化します。

Bit 3 OC1PE: 出力比較 1 プリロード・イネーブル

0: TIM1_CCR1 のプリロード・レジスタはディセーブルです。TIM1_CCR1 は常時書き込みが可能で、書込まれた値は直ちに有効となります。

1: TIM1_CCR1 のプリロード・レジスタはイネーブル状態です。読み書きはプリロード・レジスタに対して行われます。TIM1_CCR1 プリロード値は更新イベント時にアクティブレジスタにロードされます。

注 1: LOCK レベル 3 が指定されており (TIM1_BDTR レジスタの LOCK ビット) CC1S=00 (チャネルは出力設定) の場合には、これらのビットは変更できません。

注 2: 単一パルス発生モードのときにのみ、プリロード機能を使用せずに PWM モードを利用することができます。連続的な PWM モードでは、その動作は保証されません。

Bit 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 CC1 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、トリガがオンであっても、通常はカウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 が有効となるまでに、5 クロック・サイクルを要します。

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、比較結果とは別に、OC は比較レベルにセットされます。トリガ入力をサンプルし CC1 出力を有効とする際の遅延は 3 クロック・サイクルに短縮されます。OCxFE ビットはチャネルが PWM1 もしくは PWM2 モードに設定されているような効果をもちます。

Bits 1:0 CC1S: キャプチャ / 比較 1 選択

これらのビットはチャネルの方向 (入力 / 出力) と入力時の機能を定めます。

00: CC1 チャネルは出力設定です。

01: CC1 チャネルは入力設定で、IC1 には TI1 が選択されます。

10: CC1 チャネルは入力設定で、IC2 には TI2 が選択されます。

11: CC1 チャネルは入力設定で、IC2 には TRGI が選択されます。これは TS ビット (TIM1_SMCR レジスタ) で内部トリガ入力が選択されている時にのみ正しく働きます。

注: CC1S ビットはチャネルが OFF (TIM1_CCER レジスタで CC1E=0) のときにのみ書き込みができます。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

入力キャプチャ・モード

Bits 15:12 **IC2F**: 入力キャプチャ 2 フィルタ

Bits 11:10 **IC2PSC[1:0]**: 入力キャプチャ 2 プリスケーラ

Bits 9:8 **CC2S**: キャプチャ / 比較 2 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC2 チャネルは出力設定です。

01: CC2 チャネルは入力設定で、IC2 には TI2 が選択されます。

10: CC2 チャネルは入力設定で、IC2 には TI1 が選択されます。

11: CC2 チャネルは入力設定で、IC2 には TRGI が選択されます。これは TS ビット (TIM1_SMCR レジスタ) で内部トリガ入力が選択されている時にのみ正しく働きます。

注: ビットはチャネルが OFF (TIM1_CCER レジスタで CC2E=0) のときにのみ書き込みができます。

Bits 7:4 **IC1F[3:0]**: 入力キャプチャ 1 フィルタ

これらのビットは TI1 入力をサンプルする周波数と、TI1 に対するデジタル・フィルタの応答を定めます。デジタル・フィルタは、出力に有効な変化をもたらすために N イベントを要するイベントカウンタで構成されています。

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

Bits 3:2 **IC1PSC**: 入力キャプチャ 1 プリスケーラ

これらのビットは CC1 入力 (IC1) で働くプリスケーラの分周比を指定します。

プリスケーラは TIM1_CCER レジスタの CC1E=0 でリセットされます。

00: プリスケーラなし、キャプチャ入力でエッジが検出されるたびにキャプチャが行われます。

01: イベント 2 回ごとにキャプチャが行われます。

10: イベント 4 回ごとにキャプチャが行われます。

11: イベント 8 回ごとにキャプチャが行われます。

Bits 1:0 **CC1S**: キャプチャ / 比較 1 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC1 チャネルは出力設定です。

01: CC1 チャネルは入力設定で、IC1 には TI1 が選択されます。

10: CC1 チャネルは入力設定で、IC1 には TI2 が選択されます。

11: CC1 チャネルは入力設定で、IC1 には TRGI が選択されます。これは TS ビット (TIM1_SMCR レジスタ) で内部トリガ入力が選択されている時にのみ正しく働きます。

注: CC1S ビットはチャネルが OFF (TIM1_CCER レジスタで CC1E=0) のときにのみ書き込みができます。



参考資料

12.4.8 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

前述の CCMR1 レジスタの詳細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
OC4 CE	OC4M[2:0]			OC4 PE	OC4 FE	CC4S[1:0]	OC3 CE.	OC3M[2:0]			OC3 PE	OC3 FE	CC3S[1:0]		
	IC4F[3:0]			IC4PSC[1:0]			IC3F[3:0]			IC3PSC[1:0]					

出力比較モード

Bit 15 OC4CE: 出力比較 4 クリア・イネーブル

Bits 14:12 OC4M: 出力比較 4 モード

Bit 11 OC4PE: 出力比較 4 プリロード・イネーブル

Bit 10 OC4FE: 出力比較 4 高速イネーブル

Bits 9:8 CC4S: キャプチャ / 比較 4 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC4 チャネルは出力設定です。

01: CC4 チャネルは入力設定で、IC4 には TI4 が選択されます。

10: CC4 チャネルは入力設定で、IC4 には TI3 が選択されます。

11: CC4 チャネルは入力設定で、IC4 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC4S ビットはチャネルが OFF (TIMx_CCER レジスタで CC4E=0) のときにのみ書き込みができます。

Bit 7 OC3CE: 出力比較 3 クリア・イネーブル

Bits 6:4 OC3M: 出力比較 3 モード

Bit 3 OC3PE: 出力比較 3 プリロード・イネーブル

Bit 2 OC3FE: 出力比較 3 高速イネーブル

Bits 1:0 CC3S: キャプチャ / 比較 3 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC3 チャネルは出力設定です。

01: CC3 チャネルは入力設定で、IC3 には TI3 が選択されます。

10: CC3 チャネルは入力設定で、IC3 には TI4 が選択されます。

11: CC3 チャネルは入力設定で、IC3 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC3S ビットはチャネルが OFF (TIMx_CCER レジスタで CC3E=0) のときにのみ書き込みができます。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

入力キャプチャ・モード

Bits 15:12 **IC4F**: 入力キャプチャ 4 フィルタ

Bits 11:10 **IC4PSC**: 入力キャプチャ 4 プリスケーラ

Bits 9:8 **CC4S**: キャプチャ / 比較 4 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC4 チャネルは出力設定です。

01: CC4 チャネルは入力設定で、IC4 には TI4 が選択されます。

10: CC4 チャネルは入力設定で、IC4 には TI3 が選択されます。

11: CC4 チャネルは入力設定で、IC4 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC4S ビットはチャネルが OFF (TIMx_CCER レジスタで CC4E=0) のときにのみ書き込みができます。

Bits 7:4 **IC3F**: 入力キャプチャ 3 フィルタ

Bits 3:2 **IC3PSC**: 入力キャプチャ 3 プリスケーラ

Bits 1:0 **CC3S**: キャプチャ / 比較 3 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC3 チャネルは出力設定です。

01: CC3 チャネルは入力設定で、IC3 には TI3 が選択されます。

10: CC3 チャネルは入力設定で、IC3 には TI4 が選択されます。

11: CC3 チャネルは入力設定で、IC3 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC3S ビットはチャネルが OFF (TIMx_CCER レジスタで CC3E=0) のときにのみ書き込みができます。

12.4.9 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	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E	
Res.	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 15:14 予約済み、常に 0 が読出されます。

Bit 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。

Bit 12 **CC4E**: キャプチャ / 比較 4 出力イネーブル

CC1E の説明を参照してください。

Bit 11 **CC3NP**: キャプチャ / 比較 3 補助出力極性

CC1NP の説明を参照してください。

Bit 10 **CC3NE**: キャプチャ / 比較 3 補助出力イネーブル

CC1NE の説明を参照してください。



参考資料

Bit 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。

Bit 8 **CC3E**: キャプチャ / 比較 3 出力イネーブル

CC1E の説明を参照してください。

Bit 7 **CC2NP**: キャプチャ / 比較 2 補助出力極性

CC1NP の説明を参照してください。

Bit 6 **CC2NE**: キャプチャ / 比較 2 補助出力イネーブル

CC1NE の説明を参照してください。

Bit 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。

Bit 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。

Bit 3 **CC1NP**: キャプチャ / 比較 1 補助出力イネーブル

0: OC1N アクティブ・ハイ

1: OC1N アクティブ・ロー

注: このビットは LOCK レベル 2 もしくは 3 が指定 (TIM1_BKR レジスタの LOCK ビット) されており、CC1S=00 (チャネルは出力設定) のときには書き込みできません。

Bit 2 **CC1NE**: キャプチャ / 比較 1 補助出力イネーブル

0: オフ - OC1N はアクティブではありません。OC1N のレベルは MOE、OSSI、OSSR、OIS1、OIS1N 及び CC1E の組合せによって決まります。

1: オン - OC1N 信号は MOE、OSSI、OSSR、OIS1、OIS1N 及び CC1E ビットに依存し、所定の出力ピンに出力される出力です。

Bit 1 **CC1P**: キャプチャ / 比較 1 出力極性

CC1 チャネルが出力設定のとき :

0: OC1 アクティブ・ハイ

1: OC1 アクティブ・ロー

CC1 チャネルが入力設定のとき :

このビットはトリガやキャプチャ動作に IC1 か $\overline{IC1}$ のいずれが使用されるかを選択します。

0: 非反転 : キャプチャは IC1 の立上りエッジで行われます。外部トリガとして使用される時には、IC1 は反転されません。

1: 反転 : キャプチャは IC1 の立下りエッジで行われます。外部トリガとして使用される時には、IC1 は反転されます。

注: このビットは LOCK レベル 2 もしくは 3 が指定 (TIM1_BKR レジスタの LOCK ビット) されているときには書き込みできません。

Bit 0 **CC1E**: キャプチャ / 比較 1 出力イネーブル

CC1 チャネルが出力設定のとき :

0: オフ - OC1 はアクティブではありません。OC1 のレベルは MOE、OSSI、OSSR、OIS1、OIS1N 及び CC1NE の組合せによって決まります。

1: オン - OC1 信号は MOE、OSSI、OSSR、OIS1、OIS1N 及び CC1NE ビットに依存し、所定の出力ピンに出力される出力です。

CC1 チャネルが入力設定のとき :

このビットはカウンタ値のキャプチャ / 比較レジスタ 1 (TIM1_CCR1) へのキャプチャ動作が行われるか否かを指定します。

0: キャプチャディセーブル

1: キャプチャイネーブル

参考資料

表 54. ブレーク機能をもつ相補出力 OCx 及び OCxN チャネルの出力制御ビット

制御ビット					出力状態 ⁽¹⁾	
MOE bit	OSSI bit	OSSR bit	CCxE bit	CCxNE bit	OCx 出力状態	OCxN 出力状態
1	X	0	0	0	出力ディセーブル (タイマで駆動されません) OCx=CCxP, OCx_EN=0	出力ディセーブル (タイマで駆動されません) OCxN=CCxNP, OCxN_EN=0
		0	0	1	出力ディセーブル (タイマで駆動されません) OCx=CCxP, OCx_EN=0	OCxREF + 極性指定 OCxN=OCxREF xor CCxNP, OCxN_EN=1
		0	1	0	OCxREF + 極性指定 OCx=OCxREF xor CCxP, OCx_EN=1	出力ディセーブル (タイマで駆動されません) OCxN=CCxNP, OCxN_EN=0
		0	1	1	OCREF + 極性指定 + デッドタイム OCx_EN=1	OCREF に対して相補極性指 定 + デッドタイム OCx_N_EN=1
		1	0	0	出力ディセーブル (タイマで駆動されません) OCx=CCxP, OCx_EN=0	出力ディセーブル (タイマで駆動されません) OCxN=CCxNP, OCxN_EN=0
		1	0	1	オフ状態 (出力はインアク ティブレベル) OCx=CCxP, OCx_EN=1	OCxREF + 極性指定 OCxN=OCxREF xor CCxNP, OCxN_EN=1
		1	1	0	OCxREF + 極性指定 OCx=OCxREF xor CCxP, OCx_EN=1	オフ状態 (出力はインアク ティブレベル) OCxN=CCxNP, OCxN_EN=1
		1	1	1	OCREF + 極性指定 + デッドタイム OCx_EN=1	OCREF に対して相補極性指 定 + デッドタイム OCx_N_EN=1
0	X	0	0	0	出力ディセーブル (タイマで駆動されません) 非同期 : OCx=CCxP, OCx_EN=0, OCxN=CCxNP, OCxN_EN=0 クロックがデッドタイム後に、OCx=OISx 及び OCxN=OISxN が供給されている 場合、OISx と OISxN は OCx 及び OCxN が共にアクティブであると指示しない。	
		0	0	1		
		0	1	0		
		0	1	1		
		1	0	0	オフ状態 (出力はインアクティブレベル) 非同期 : OCx=CCxP, OCx_EN=1, OCxN=CCxNP, OCxN_EN=1 クロックがデッドタイム後に、OCx=OISx 及び OCxN=OISxN が供給されている 場合、OISx と OISxN は OCx 及び OCxN が共にアクティブであると指示しない。	
		1	0	1		
		1	1	0		
		1	1	1		

- チャネルの両方の出力が使用されない場合 (CCxE=CCxNE=0)、OISx、OISxN、CCxP 及び CCxNP はクリアされたままでなければなりません。

注意 : 相補出力 OCx チャネル及び OCxN チャネルに接続されている外部入出力ピンの状態は、両チャネルの状態と、GPIO/AFIO レジスタの設定に依存します。

参考資料

12.4.10 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]															

Bits 15:0 CNT[15:0]: カウンタの値

12.4.11 プリスケーラ (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]															

Bits 15:0 PSC[15:0]: プリスケーラの値

カウンタ・クロック (CK_CNT) の周波数は $f_{CK_PSC} / (PSC[15:0] + 1)$ となります。

PSC は更新イベントごとにアクティブ・プリスケーラ・レジスタにロードされる値を格納しています。(更新イベントには、TIM1_EGR レジスタの UG ビットを使用して、もしくはリセット・モードのトリガコントローラを経由してカウンタをクリアしたときも含まれます。)

12.4.12 自動再ロード・レジスタ (TIMx_ARR)

アドレスオフセット : 0x2C

リセット値 : 0x0000

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

Bits 15:0 ARR[15:0]: 自動再ロード・レジスタの値

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 の更新と挙動の詳細は [セクション 12.3.1: タイムベース・ユニット \(218 ページ\)](#) を参照してください。

自動再ロード値が 0 のときには、カウンタへのロードはブロックされています。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

12.4.13 繰返しカウンタ・レジスタ (TIMx_RCR)

アドレスオフセット : 0x30

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								REP[7:0]							
Res.								rW	rW	rW	rW	rW	rW	rW	rW

Bits 15:8 予約済み、常に 0 が読出されます。

Bits 7:0 REP[7:0]: 繰返しカウンタの値

これらのビットはプリロード機能が有効なとき、比較レジスタの更新レートを構成するために使用されます。(周期的にプリロード・レジスタからアクティブレジスタにデータが移されます。) また、割込みリクエストが許可されていれば、割込みリクエストの発生の度合いも更新します。繰返しダウンカウンタ (REP_CNT) が 0 に達すると、更新イベントが発生し、REP 値からカウントを再開します。REP_CNT には、繰返し更新イベント U_RC でのみ、REP の値がロードされます。TIM1_RCR レジスタへの書込みは次の繰返し更新イベントまで効果をもちません。

PWM モードでは (REP+1) は次の意味をもちます。

- エッジアライン・モードでは、PWM 周期の数
- センターライン・モードでは、PWM の 1/2 周期の数 (周期の数の 2 倍)

12.4.14 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

Bits 15:0 CCR1[15:0]: キャプチャ / 比較 1 の値

CC1 チャネルが出力設定のとき :

CCR1 はアクティブキャプチャ / 比較レジスタ 1 にロードされる値 (プリロード値) です。

TIM1_CCMR1 レジスタの CC1PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード値がアクティブキャプチャ / 比較レジスタ 1 にロードされます。

アクティブキャプチャ / 比較レジスタは OC1 出力を送出するためカウンタ TIM1_CNT と比較される値を保持しています。

CC1 チャネルが入力設定のとき :

CCR1 は直近の入力キャプチャ 1 イベント (IC1) でラッチされたカウンタの値です。



参考資料

12.4.15 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]															

Bits 15:0 CCR2[15:0]: キャプチャ / 比較 2 の値

CC2 チャネルが出力設定のとき :

CCR2 はアクティブキャプチャ / 比較レジスタ 2 にロードされる値（プリロード値）です。

TIM1_CCMR2 レジスタの CC2PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード値がアクティブキャプチャ / 比較レジスタ 2 にロードされます。

アクティブキャプチャ / 比較レジスタは OC2 出力を送出するためにカウンタ TIM1_CNT と比較される値を保持しています。

CC2 チャネルが入力設定のとき :

CCR2 は直近の入力キャプチャ 2 イベント (IC2) でラッチされたカウンタの値です。

12.4.16 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]															

Bits 15:0 CCR3[15:0]: キャプチャ / 比較 3 の値

CC3 チャネルが出力設定のとき :

CCR3 はアクティブキャプチャ / 比較レジスタ 1 にロードされる値（プリロード値）です。

TIM1_CCMR3 レジスタの CC3PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード値がアクティブキャプチャ / 比較レジスタ 3 にロードされます。

アクティブキャプチャ / 比較レジスタは OC3 出力を送出するためにカウンタ TIM1_CNT と比較される値を保持しています。

CC3 チャネルが入力設定のとき :

CCR3 は直近の入力キャプチャ 3 イベント (IC3) でラッチされたカウンタの値です。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

12.4.17 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]															

Bits 15:0 CCR4[15:0]: キャプチャ / 比較 4 の値

CC4 チャネルが出力設定のとき :

CCR4 はアクティブキャプチャ / 比較レジスタ 4 にロードされる値（プリロード値）です。

TIM1_CCMR4 レジスタの CC4PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード値がアクティブキャプチャ / 比較レジスタ 4 にロードされます。

アクティブキャプチャ / 比較レジスタは OC4 出力を送出するためにカウンタ TIM1_CNT と比較される値を保持しています。

CC4 チャネルが入力設定のとき :

CCR4 は直近の入力キャプチャ 4 イベント (IC4) でラッチされたカウンタの値です。

12.4.18 ブレーク / デッドタイムレジスタ (TIMx_BDTR)

アドレスオフセット : 0x44

リセット値 : 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]	DTG[7:0]								

注意 : AOE、BKP、BKE、OSSI、OSSR 及び DTG[7:0] は LOCK 指定に応じて書き込みがロック（禁止）されますので、TIMx_BDTR レジスタへの最初のアクセスで、これらのすべてを一度に設定することが必要とされます。

Bit 15 MOE: メイン出力イネーブル

このビットはブレーク入力がアクティブとなると、ハードウェアによって随時クリアされます。このビットはソフトウェアでセットされます。また、AOE ビットの設定に応じてハードウェアにより自動的にセットされます。このビットの設定は、出力設定のチャネルに対してのみ有効です。

0: OC 及び OCN 出力はディセーブル状態、もしくは強制的にアイドル状態とされます。

1: OC 及び OCN 出力は、対応するイネーブル・ビット (TIMx_CCER レジスタの CCxE、CCxNE ビット) がセットされていればイネーブル状態となります。OC/OCN イネーブルの詳細に関しては、[セクション 12.4.9: キャプチャ / 比較イネーブル・レジスタ \(TIMx_CCER\) \(273 ページ\)](#) を参照してください。

Bit 14 AOE: 自動出力イネーブル

0: MOE はソフトウェアでのみセットされます。

1: MOE はソフトウェアの他、次の更新イベントでも自動的にセットされます。（ブレーク入力がアクティブレベルではない場合）

注: これらのビットは LOCK レベルが 1 に設定されたときには変更できません。（TIMx_BDTR レジスタの LOCK ビット）



参考資料

Bit 13 BKP: ブレーク極性

- 0: ブレーク入力 BRK はアクティブ・ローです。
- 1: ブレーク入力 BRK はアクティブ・ハイです。

注: これらのビットは LOCK レベルが 1 に設定されたときには変更できません。 (TIMx_BDTR レジスタの LOCK ビット)

Bit 12 BKE: ブレークイネーブル

- 0: ブレーク入力 (BRK と BRK_ACTH) はディセーブル状態です。
- 1: ブレーク入力 (BRK と BRK_ACTH) はイネーブル状態です。

注: これらのビットは LOCK レベルが 1 に設定されたときには変更できません。 (TIMx_BDTR レジスタの LOCK ビット)

Bit 11 OSSR: RUN モードにおけるオフ状態の選択

このビットは MOE=1 のとき、相補出力を有する出力設定のチャネルで使用されます。 OSSR は相補出力が組込まれていないタイマには組込まれていません。

詳細は [セクション 12.4.9: キャプチャ / 比較イネーブル・レジスタ \(TIMx_CCER\) \(273 ページ\)](#) にある OC/OCN イネーブルの説明を参照してください。

- 0: インアクティブのとき、OC/OCN 出力はディセーブルとなります。 (OC/OCN イネーブル出力信号 = 0)

- 1: インアクティブのとき、CCxE=1 もしくは CCxNE=1 になると、OC/OCN 出力はイネーブル状態でインアクティブルレベルとなります。 (OC/OCN イネーブル出力信号 = 1)

注: これらのビットは LOCK レベルが 2 に設定されたときには変更できません。 (TIMx_BDTR レジスタの LOCK ビット)

Bit 10 OSSI: アイドルモードにおけるオフ状態の選択

このビットは MOE=0 のとき、出力設定のチャネルで使用されます。

詳細は [セクション 12.4.9: キャプチャ / 比較イネーブル・レジスタ \(TIMx_CCER\) \(273 ページ\)](#) にある OC/OCN イネーブルの説明を参照してください。

- 0: インアクティブのとき、OC/OCN 出力はディセーブルとなります。 (OC/OCN イネーブル出力信号 = 0)

- 1: インアクティブのとき、CCxE=1 もしくは CCxNE=1 になると、OC/OCN 出力はアイドルレベルになります。 (OC/OCN イネーブル出力信号 = 1)

注: これらのビットは LOCK レベルが 2 に設定されたときには変更できません。 (TIMx_BDTR レジスタの LOCK ビット)

Bits 9:8 LOCK[1:0]: ロック設定

このビットはソフトウェアの誤りなどに対する書き込み保護を提供します。

00: LOCK オフ - 書込み保護は行われません。

01: LOCK レベル 1 = TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx 及び OISxN ビット、TIMx_BDTR レジスタの BKE、BKP、AOE ビットの書き込みができなくなります。

10: LOCK レベル 2 = LOCK レベル 1 に加え、TIMx_CCER レジスタの CC 極性ビット (CCxS ビットによりチャネルが出力設定のとき、TIMx_CCMRx レジスタの OCxM 及び OCxPE ビット)、及び OSSR と OSSI ビットの書き込みができなくなります。

11: LOCK レベル 3 = LOCK レベル 2 に加え、CC 制御ビット (CCxS ビットによりチャネルが出力設定のとき、TIMx_CCMRx レジスタの OCxM 及び OCxPE ビット) の書き込みができなくなります。

注: LOCK ビットはリセット後に一度だけ書き込みができます。一旦 TIMx_BDTR レジスタに書き込みが行われると、これらのビットの値は次のリセットまで凍結されます。

参考資料

Bits 7:0 DTG[7:0]: デッドタイム発生回路の設定

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間 (DT) は次の式で与えられます。

$$\begin{aligned} \text{DTG}[7:5]=0xx &\Rightarrow DT = DTG[7:0] \times t_{dtg} \text{ ここで } t_{dtg}=t_{DTS} \\ \text{DTG}[7:5]=10x &\Rightarrow DT = (64+DTG[5:0]) \times t_{dtg} \text{ ここで } t_{dtg}=2 \times t_{DTS} \\ \text{DTG}[7:5]=110 &\Rightarrow DT = (32+DTG[4:0]) \times t_{dtg} \text{ ここで } t_{dtg}=8 \times t_{DTS} \\ \text{DTG}[7:5]=111 &\Rightarrow DT = (32+DTG[4:0]) \times t_{dtg} \text{ ここで } t_{dtg}=16 \times t_{DTS} \end{aligned}$$

例 : $t_{DTS}=125\text{ns}$ (8MHz) のとき、デッドタイムとして設定可能な値は、

- 0 から 15875 ns、125 ns ステップ
- 16 μs から 31750 ns、250 ns ステップ
- 32 μs から 63 μs 、1 μs ステップ
- 64 μs から 126 μs 、2 μs ステップ

注: これらのビットは LOCK レベルが 1、2 もしくは 3 に設定されたときには変更できません。
(TIMx_BDTR レジスタの LOCK ビット)

12.4.19 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]				Reserved			DBA[4:0]				
Res.	rw	rw	rw	rw	rw	rw	rw	Res.	rw	rw	rw	rw	rw	rw	rw

Bits 15:13 予約済み、常に 0 が読出されます。

Bits 12:8 DBL[4:0]: DMA バースト長

これらのビットはバースト・モード (TIMx_DMAR のアドレスに対して行われる読み書きのアクセスはバースト転送として扱われます) で DMA 転送の際のデータ長を指示します。転送はハーフワードかバイトで行われます。(以下を参照ください)

- 00000: 1 転送
- 00001: 2 転送
- 00010: 3 転送

...
10001: 18 転送

例: 次のような転送を考えるとします : DBL = 7 バイトおよび DBA = TIM2_CR1

- もし DBL = 7 バイトと DBA = TIM2_CR1 が送信されるバイトのアドレスを表すとすれば、送信アドレスは次の式により導かれます :

(TIMx_CR1 アドレス) + DBA + (DMA インデックス)、DMA インデックス = DBL です。

この例では、7 バイトが (TIMx_CR1 アドレス) + DBA に加算されます。これによりコピーされるデータのアドレスを得ます。この場合、送信は次のアドレス : (TIMx_CR1 アドレス) + DBA から 7 つのレジスタへ行われます。

DMA データ・サイズの設定によっては、幾つかのケースが考えられます。:

- DMA データ・サイズをハーフワードで設定すると、16 ビットデータは 7 つのレジスタへそれぞれ送信されます。
- DMA データ・サイズをバイトに設定すると、データは 7 つのレジスタへ送信されます。: 最初のレジスタは最初の MSB バイト、2 番目のレジスタ、最初の LSB バイトと言う具合です。送信タイマについても、DMA により送信されたデータのサイズを指定しなければなりません。

Bits 7:5 予約済み、常に 0 が読出されます。

参考資料

Bits 4:0 DBA[4:0]: DMA ベース・アドレス

これらのビットはバースト・モード（読み書きのアクセスは TIMx_DMAR のアドレスを通して行われます）で DMA 転送のためのベース・アドレスを指示します。DBA は TIM1_CR1 レジスタのアドレスから始まるオフセット値として格納されます。

例：

00000: TIM1_CR1
00001: TIM1_CR2
00010: TIM1_SMCR

...

12.4.20 バースト・モード DMA アドレス・レジスタ (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
DMA[15:0]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits 15:0 DMAB[15:0]: DMA バーストアクセスレジスタ

DMAR レジスタへの読み書きのアクセスは、次のアドレスにあるレジスタへのアクセスとなります。
(TIM1_CR1 アドレス) + DBA + (DMA オフセット)、ここで

TIM1_CR1 アドレスは、制御レジスタ 1 のアドレスです。

DBA は、TIM1_DCR レジスタに設定されている DMA ベース・アドレスです。

DMA オフセットは、DMA 転送の際に自動的に制御されるオフセット値で、TIM1_DCR レジスタの転送長 DBL に依存した値となります。

参考資料

RM0008

高機能制御タイマ (TIM1 及び TIM8)

12.4.21 TIM1 及び TIM8 レジスタ・マップ

TIM1 及び TIM8 レジスタは次の表のように、すべて 16bit のレジスタです。

表 55. TIM1 及び TIM8 レジスタ・マップとリセット時の値

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]																	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x04	TIMx_CR2	予約済み												OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	OIS1S	MMMS[2:0]		DIR	OPM	CCDS	URS	Reserved	UDIS	CCPC			
	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			
0x08	TIMx_SMCR	予約済み												ETP	ETPS [1:0]	ETF[3:0]	MSM		TS[2:0]		SMS[2:0]		予約済み		CCUS		URS		Reserved		UDIS		
	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			
0x0C	TIMx_DIER	予約済み												TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UJE	UJE	UJE	UJE	UJE	
	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			
0x10	TIMx_SR	予約済み												CC4OF	CC3OF	CC2OF	CC1OF	Reserved	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UJF	UJF	UJF	UJF	UJF	UJF		
	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			
0x14	TIMx_EGR	予約済み												BG		TG		COM		CC4G		CC3G		CC2G		CC1G		UG		UG			
	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			
0x18	TIMx_CCMR1 Output Compare mode	予約済み												OC2CE	OC2M [2:0]	OC2PE	OC2FE	CC2S [1:0]	OC1CE	OC1M [2:0]	OC1PE	CC1S [1:0]		CC1S [1:0]									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x1C	TIMx_CCMR1 Input Capture mode	予約済み												IC2F[3:0]	IC2 PSC [1:0]	CC2S [1:0]	IC1F[3:0]		IC1 PSC [1:0]	CC1S [1:0]	CC1S [1:0]												
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x20	TIMx_CCMR2 Output Compare mode	予約済み												O24CE	OC4M [2:0]	OC4PE	OC4FE	CC4S [1:0]	OC3CE	OC3M [2:0]	OC3PE	CC3S [1:0]		CC3S [1:0]									
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x24	TIMx_CNT	予約済み												CNT[15:0]												CNT[15:0]		CNT[15:0]		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			
0x28	TIMx_PSC	予約済み												PSC[15:0]												PSC[15:0]		PSC[15:0]		PSC[15:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x2C	TIMx_ARR	予約済み												ARR[15:0]												ARR[15:0]		ARR[15:0]		ARR[15:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x30	TIMx_RCR	予約済み												REP[7:0]												0	0	0	0	0	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			

参考資料

表 55. TIM1 及び TIM8 レジスタ・マップとリセット時の値 (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	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			
0x38	TIMx_CCR2	予約済み												CCR2[15:0]																			
														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x3C	TIMx_CCR3	予約済み												CCR3[15:0]																			
														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x40	TIMx_CCR4	予約済み												CCR4[15:0]																			
														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x44	TIMx_BDTR	予約済み												MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]	DT[7:0]												
														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x48	TIMx_DCR	予約済み												DBL[4:0]				予約済み	DBA[4:0]														
														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x4C	TIMx_DMAR	予約済み												DMAB[15:0]																			
														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			

レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。

13 汎用タイマ (TIMx)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適用されます。

13.1 概要

汎用タイマは、プログラマブルなプリスケーラで駆動される 16bit 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、入力信号のパルス幅の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM）など、様々な目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロック・コントローラ・プリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

タイマは互いに独立しており、リソースの一部を共有していることはありません。また [セクション 13.3.15](#) に記されているように、相互に同期させることもできます。

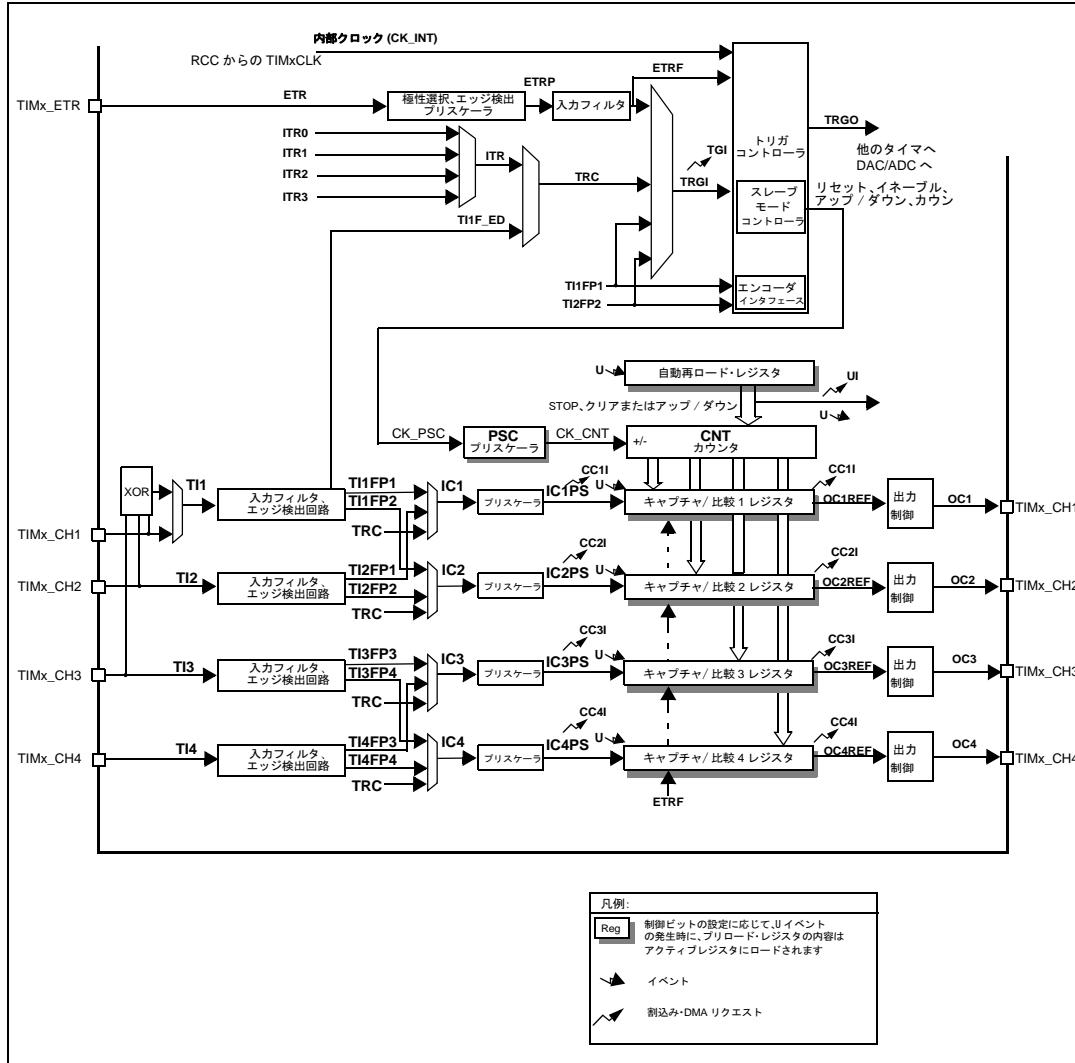
13.2 TIMx の主な機能

汎用 TIMx タイマ (TIM2、TIM3、TIM4、TIM5) は次の機能を搭載しています。

- 16bit アップカウンタ、ダウンカウンタ、アップ / ダウン再ロード・カウンタ。
- 16bit プログラマブル・プリスケーラ。カウンタ・クロック周波数を 1 から 65535 の間で分周でき、分周比を動作中に変更することもできます。
- 次の機能を持つ、独立した最大 4 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）
 - 単パルスモード出力
- 外部信号でタイマを制御する同期回路と複数タイマ間の相互接続。
- 以下のイベントによる、割込み / DMA リクエストの発生。
 - 更新：カウンタ・オーバフロー / アンダフロー、カウンタの初期化（ソフトウェア指示、及び内部 / 外部トリガ）
 - トリガ・イベント（カウンタ・スタート、ストップ、初期化、及び内部 / 外部トリガによるカウント動作）
 - 入力キャプチャ
 - 出力比較

参考資料

図 95. 汎用タイマのブロック図



13.3 TIMx の機能詳細

13.3.1 タイムベース・ユニット

プログラム可能なタイマの主要なブロックは自動再ロード・レジスタを持った 16bit カウンタです。カウンタはカウントアップ、カウントダウン、及びアップ・ダウンの動作が可能です。カウンタのクロックはプレスケーラによって分周されます。

カウンタ、自動再ロード・レジスタ、及びプリスケーラ・レジスタは、カウンタが動作中であっても、ソフトウェアによる読み書きができます。

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)
- プリスケーラ・レジスタ (TIMx_PSC):
- 自動再ロード・レジスタ (TIMx_ARR)

参考資料

自動再ロード・レジスタはプリロード機能を搭載しています。自動再ロード・レジスタの読み書きはプリロード・レジスタへのアクセスとなります。プリロード・レジスタの内容が、TIMx_CR1 レジスタのプリロード・イネーブル・ビット (ARPE) の設定に応じて、常にもしくは更新イベント 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 であれば、更新イベントはカウンタがオーバフロー (ダウンカウント時はアンダフロー) に達したときに発生します。また、ソフトウェアで更新イベントを発生させることも可能です。この方法は、それぞれのタイマ構成のセクションに記述されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタ・イネーブル・ビット (CEN) がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます。(カウンタをイネーブルとする方法の詳細に関しては、スレーブ・モード・コントローラの記述を合わせてお読みください。)

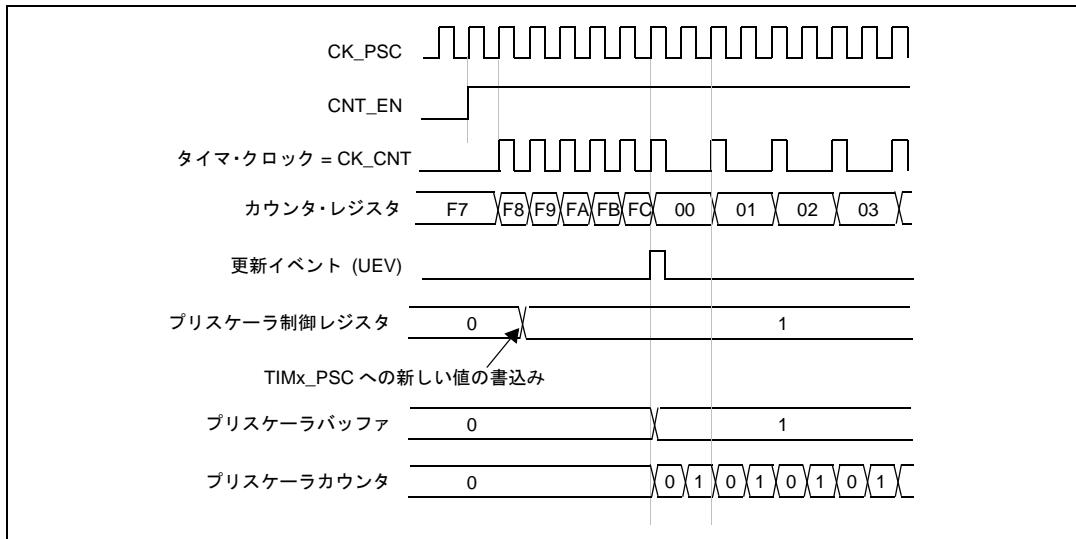
なお、実際のカウンタ・イネーブル信号 CNT_EN は CEN ビットがセットされてから 1 クロック遅れてセットされることに留意してください。

プリスケーラ

プリスケーラはカウンタ・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。プリスケーラは 16bit レジスタ (TIMx_PSC) を通して制御される 16bit カウンタをベースとしています。制御レジスタにはバッファが備わっていますので、プリスケーラが動作中であっても、レジスタ内容を変更することができます。新たに指定された分周比は次の更新イベントで効果を持ちます。

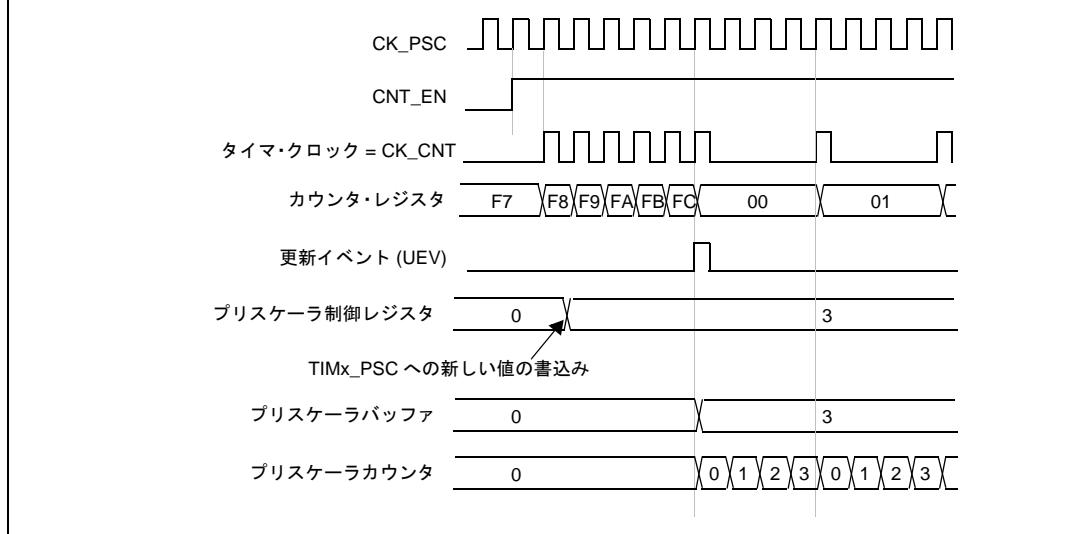
図 96 と図 97 は動作中に分周比を変化させたときのカウンタの動作の例を示しています。

図 96. プリスケーラ分周比が 1 から 2 に変化した時のカウンタ・タイミング図



参考資料

図 97. プリスケーラ分周比が 1 から 4 に変化した時のカウンタ・タイミング図



13.3.2 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・レジスタの値 (TIMx_ARR レジスタの内容) までカウントを行い、その後 0 からのカウントを再開し、同時にカウンタ・オーバフロー・イベントを発生させます。

カウンタのオーバフローごとに更新イベントが発生します。ソフトウェアによるレジスタ書き込みで、もしくはスレーブ・モード・コントローラによって TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントを発生させることができます。

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で、ディセーブルすることができます。これはプリロード・レジスタに新しい値を書込んでいたときにシャドウ・レジスタが更新されるのを防止します。この後 UDIS ビットに 0 が書込まれるまで、更新イベントは発生しません。しかし、プリスケーラのカウンタと同じく（ただしプリスケーラの比は変化しません）、カウンタは 0 からカウントを再開します。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 ビットのセットが更新イベント UEV を引起こしますが、UIF フラグはセットされず、割込みや DMA リクエストは発生しません。これは、キャプチャイベントの際にカウンタをクリアすることで、更新とキャプチャ割込みがともに発生することを防ぎます。

更新イベントが発生したとき、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。

- プリスケーラのバッファにはプリロード値 (TIMx_PSC レジスタの値) が再びロードされます。
- 自動再ロード・シャドウ・レジスタはプリロード・レジスタ (TIMx_ARR) の値に更新されます。

以下の図は、自動再ロード・レジスタ TIMx_ARR = 0x36 の場合の、異なるクロック周波数におけるカウンタの挙動の例を示しています。

参考資料

図 98. 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

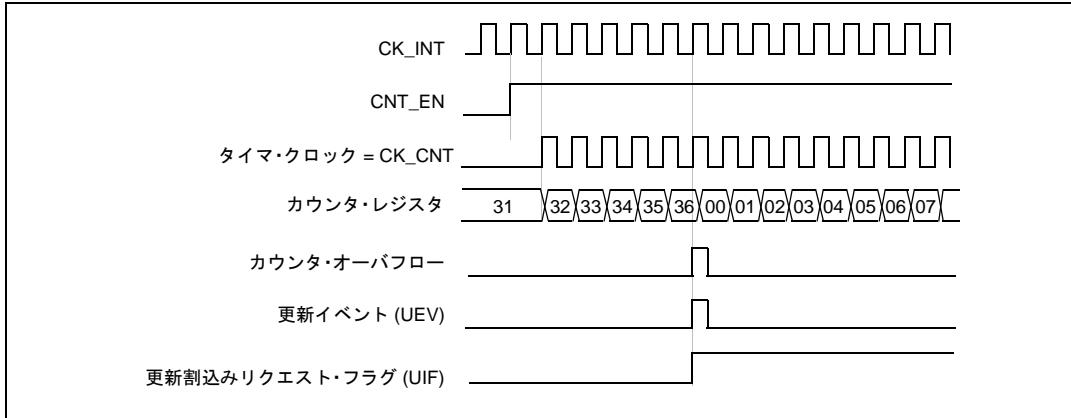


図 99. 内部クロック分周比が 2 の場合のカウンタ・タイミング図

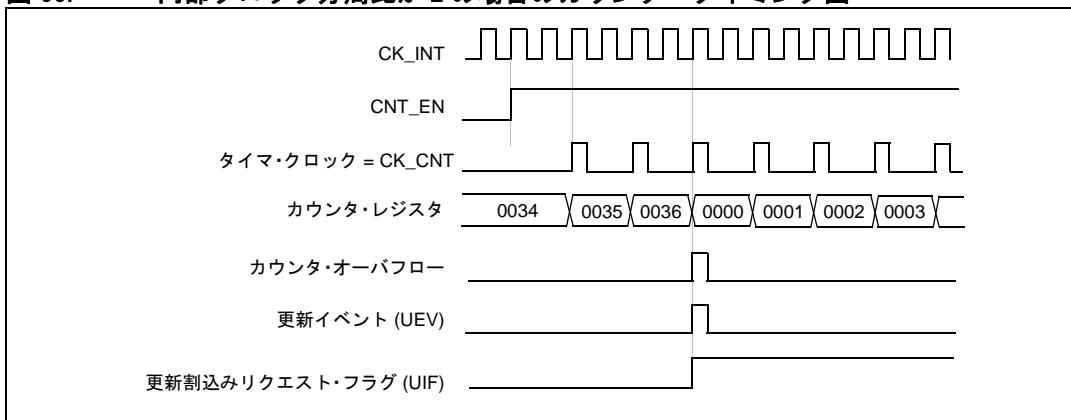
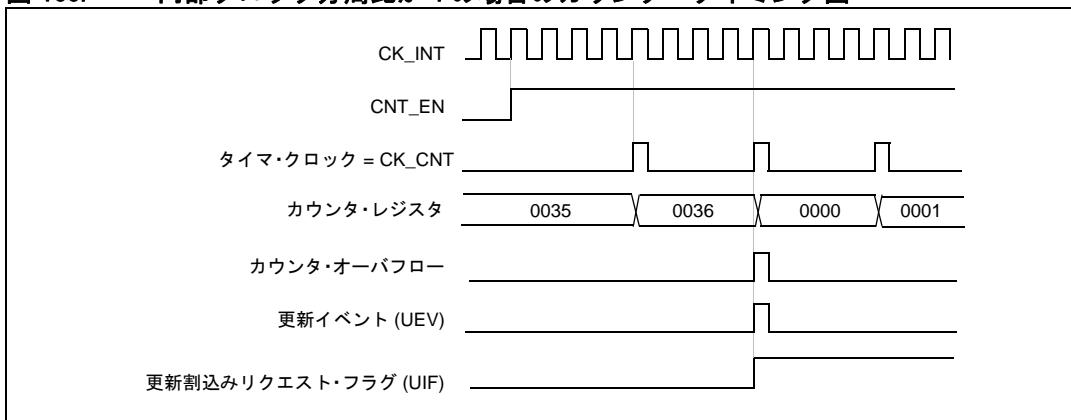


図 100. 内部クロック分周比が 4 の場合のカウンタ・タイミング図



参考資料

図 101. 内部クロック分周比が N の場合のカウンタ・タイミング図

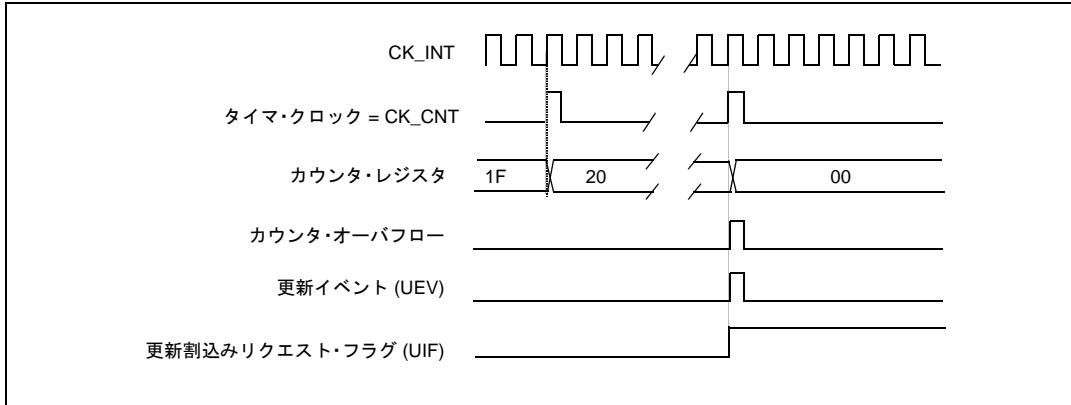
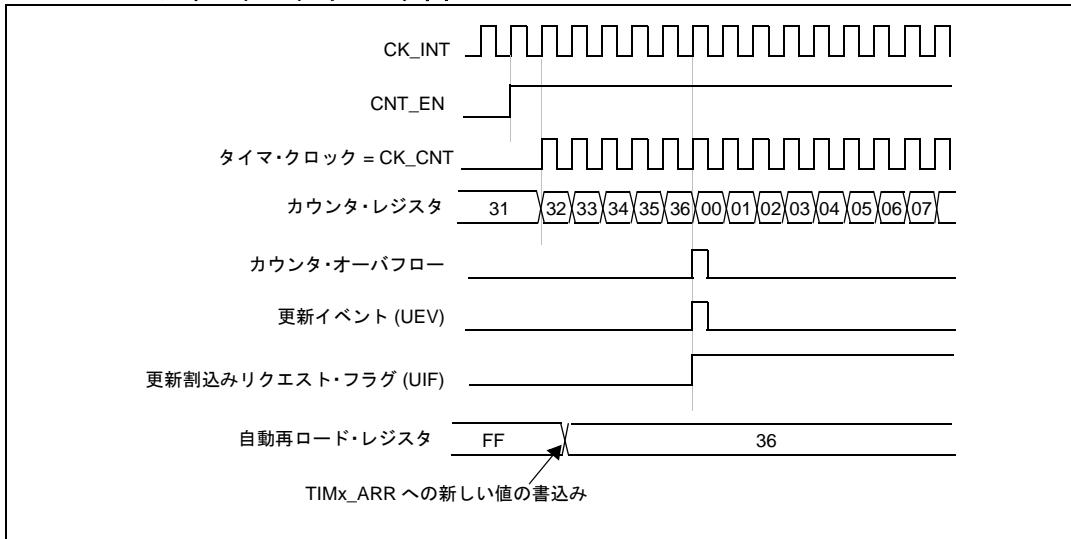
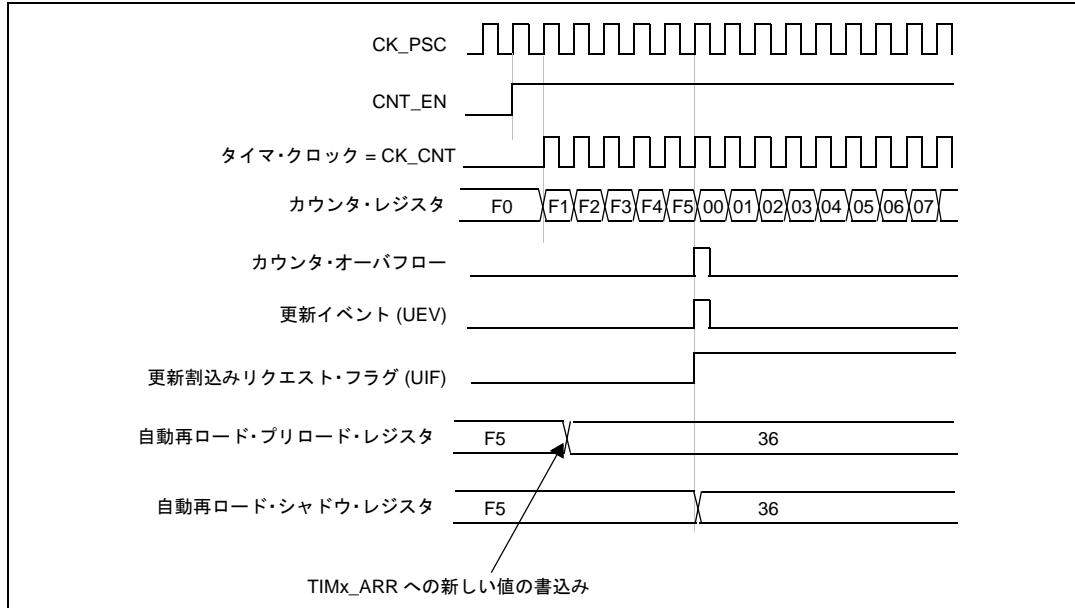


図 102. ARPE=0 (TIMx_ARR はプリロードなし) の時の更新イベントのカウンタ・タイミング図



参考資料

図 103. ARPE=1 (TIMx_ARR はプリロード) の時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・レジスタの値 (TIMx_ARR レジスタの内容) から 0 までのカウントダウンを行い、その後、自動再ロード・レジスタの値からのカウントダウンを再開し、同時にカウンタ・アンダフロー・イベントを発生させます。

カウンタのアンダフローごとに更新イベントが発生します。ソフトウェアによるレジスタ書き込みで、もしくはスレーブ・モード・コントローラによって TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントを発生させることができます。

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で、ディセーブルすることができます。これはプリロード・レジスタに新しい値を書込んでいたときにシャドウ・レジスタが更新されるのを防止します。この後 UDIS ビットに 0 が書込まれるまで、更新イベントは発生しません。しかし、プリスクーラのカウンタが 0 からカウントを再開するのに対し（ただしプリスクーラの比は変化しません）、カウンタはその時点の自動再ロード・レジスタの値からカウントダウンを再開します。

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 ビットのセットが更新イベント UEV を引起こしますが、UIF フラグはセットされず、割込みや DMA リクエストは発生しません。これは、キャプチャイベントの際にカウンタをクリアすることで、更新とキャプチャ割込みがともに発生することを防ぎます。

更新イベントが発生したとき、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。

- プリスクーラのバッファにはプリロード値 (TIMx_PSC レジスタの値) 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・レジスタ (TIMx_ARR) の値に更新されます。自動再ロード・レジスタの更新はカウンタの再ロードの前に行われますので、次の周期は希望の時間となります。

以下の図は、自動再ロード・レジスタ TIMx_ARR = 0x36 の場合の、異なるクロック周波数におけるカウンタの挙動の例を示しています。

参考資料

図 104. 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

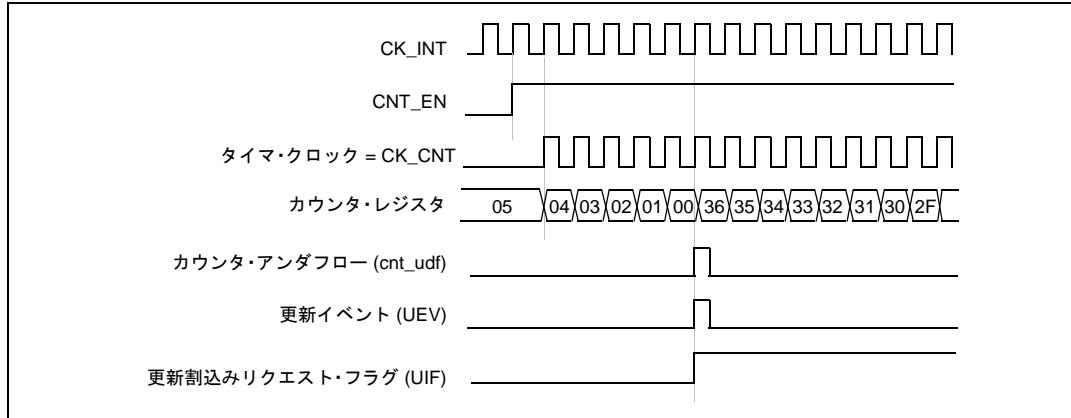


図 105. 内部クロック分周比が 2 の場合のカウンタ・タイミング図

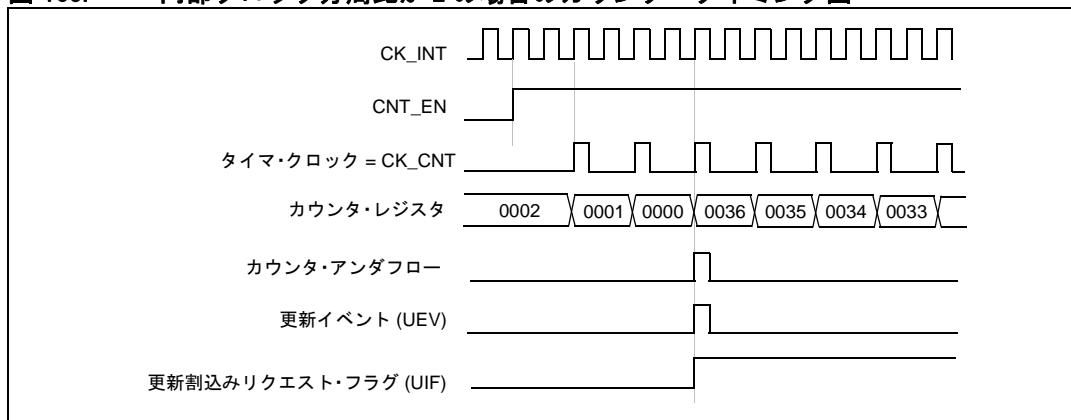
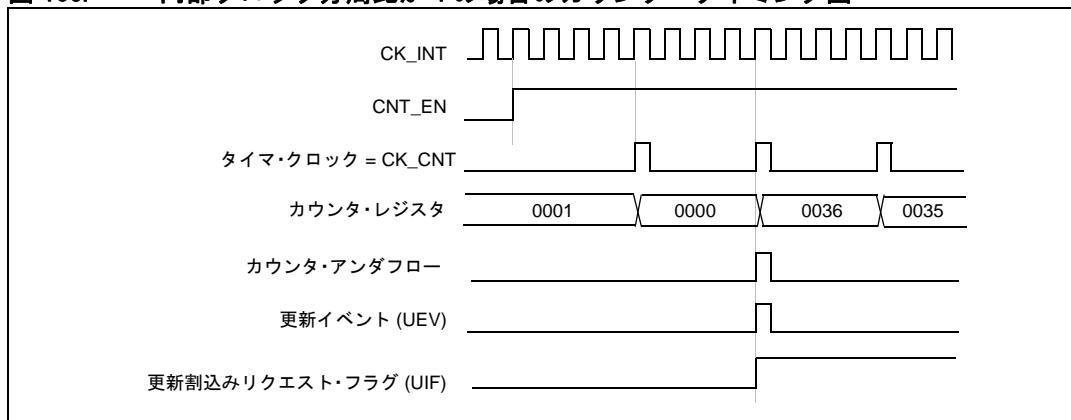


図 106. 内部クロック分周比が 4 の場合のカウンタ・タイミング図



参考資料

図 107. 内部クロック分周比が N の場合のカウンタ・タイミング図

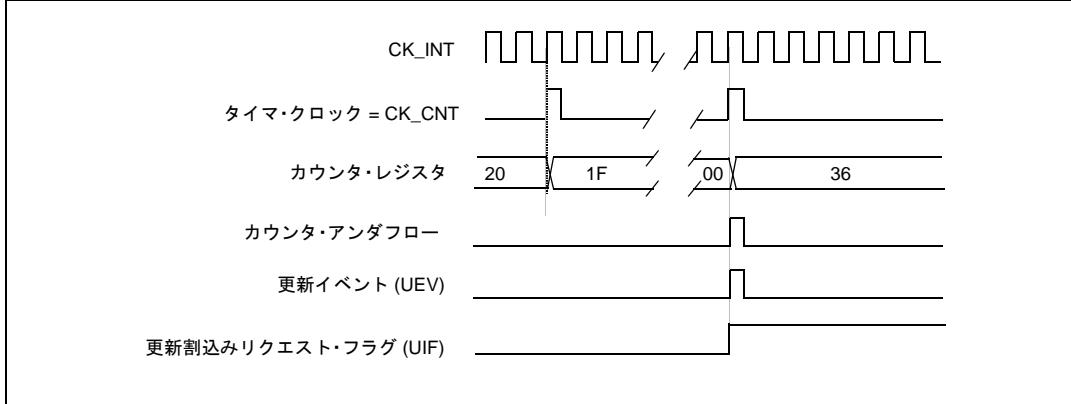
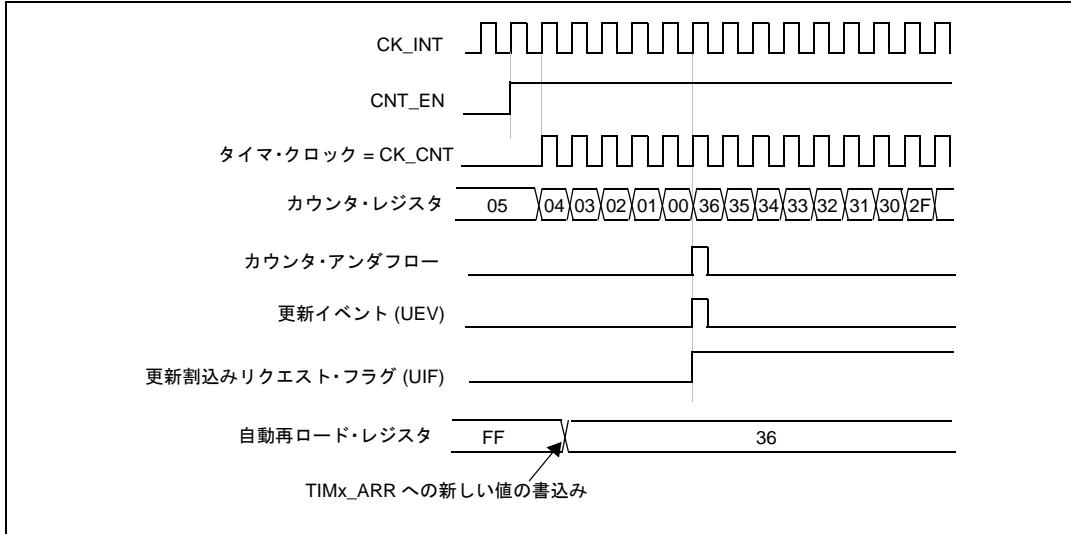


図 108. 繰り返しカウンタが使用されない時の更新イベントのカウンタ・タイミング図



センターライン・モード（アップ / ダウンカウント）

センタークライン・モードでは、カウンタは 0 から自動再ロード・レジスタの値 (TIMx_ARR レジスタの内容) - 1 までカウントを行い、カウンタ・オーバフロー・イベントを発生させます。次に 1 までのカウントダウンを行い、カウンタ・アンダフロー・イベントを発生させます。その後 0 からのカウントを再開します。

このモードでは TIMx_CR1 レジスタの方向ビット (DIR) への書込みはできません。このビットはハードウェアによって更新され、その時点でのカウンタのカウント方向を表示します。

更新イベントはカウンタがオーバフロー及びアンダフローを生じるごとに発生します。また、TIMx_EGR レジスタの UG ビットをセットすることで、ソフトウェアで、もしくはスレーブ・モード・コントローラにより、更新イベントを発生させることもできます。このときには、プリスケーラのカウンタと同じく、カウンタは 0 からのカウントアップを再開します。

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で、ディセーブルとすることができます。これはプリロード・レジスタに新しい値を書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止します。この後 UDIS ビットに 0 が書込まれるまで、更新イベントは発生しません。しかし、カウンタはその時点の自動再ロード値を使用して、カウント動作（アップ / ダウン）を続けます。

参考資料

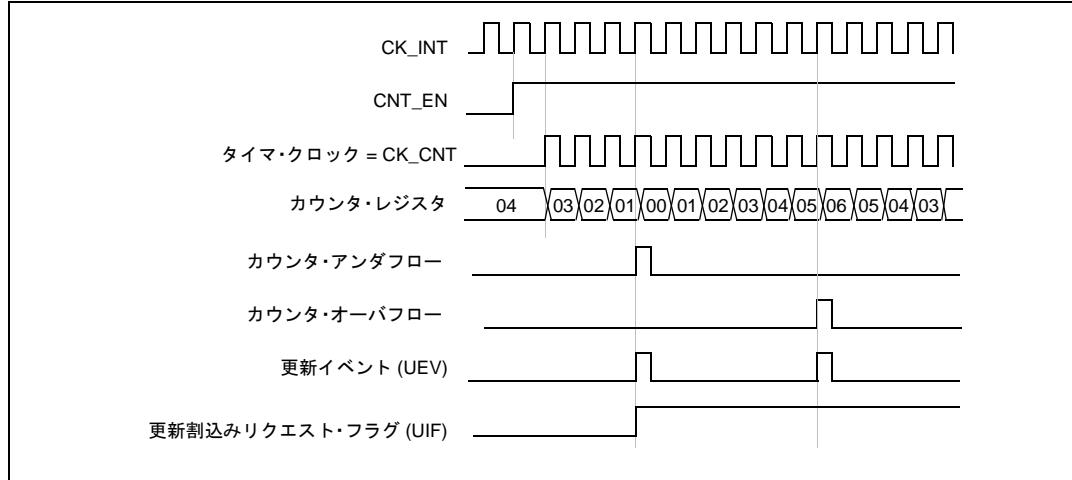
さらに、`TIMx_CR1` レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットのセットが更新イベント UEV を引起こしますが、UIF フラグはセットされず、割込みや DMA リクエストは発生しません。これは、キャプチャイベントの際にカウンタをクリアすることで、更新とキャプチャ割込みがともに発生することを防ぎます。

更新イベントが発生したとき、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。

- プリスケーラのバッファにはプリロード値 (`TIMx_PSC` レジスタの値) 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・レジスタ (`TIMx_ARR`) の値に更新されます。更新の原因がカウンタのオーバフローであるときには、自動再ロード・レジスタの更新はカウンタの再ロードの前に行われますので、次の周期は希望の時間となります。

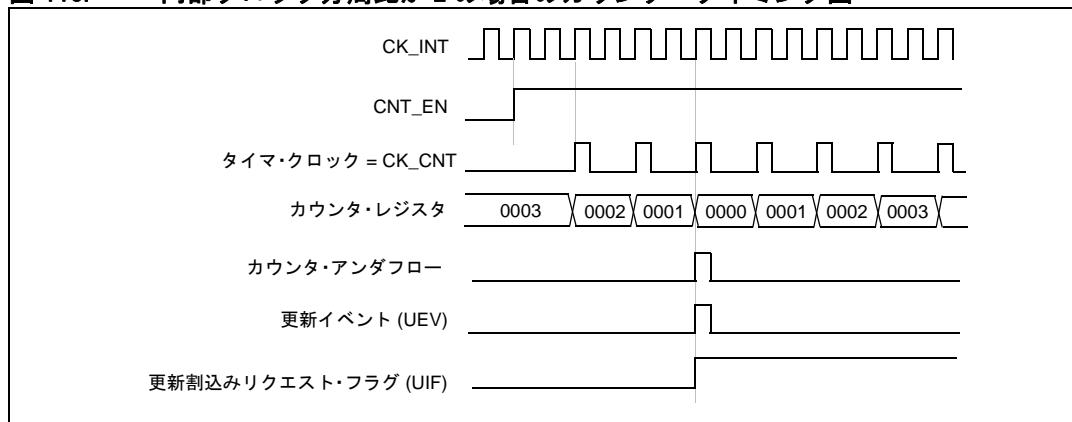
以下の図は異なるクロック周波数におけるカウンタの挙動の例を示しています。

図 109. 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 (`TIMx_ARR=0x6`)



1. ここでは、センター・アライン・モード 1 が使われています。(詳細は [セクション 13.4.1: 制御レジスタ 1 \(TIMx_CR1\) \(321 ページ\)](#) を参照してください。)

図 110. 内部クロック分周比が 2 の場合のカウンタ・タイミング図



参考資料

図 111. ロード・レジスタ内部クロック分周比が 4 の場合のカウンタ・タイミング図
(TIMx_ARR=0x36)

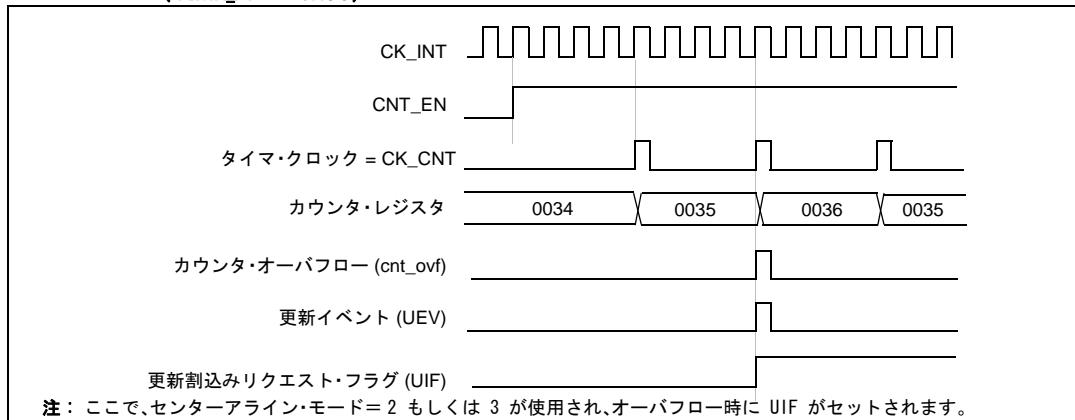


図 112. 内部クロック分周比が N の場合のカウンタ・タイミング図

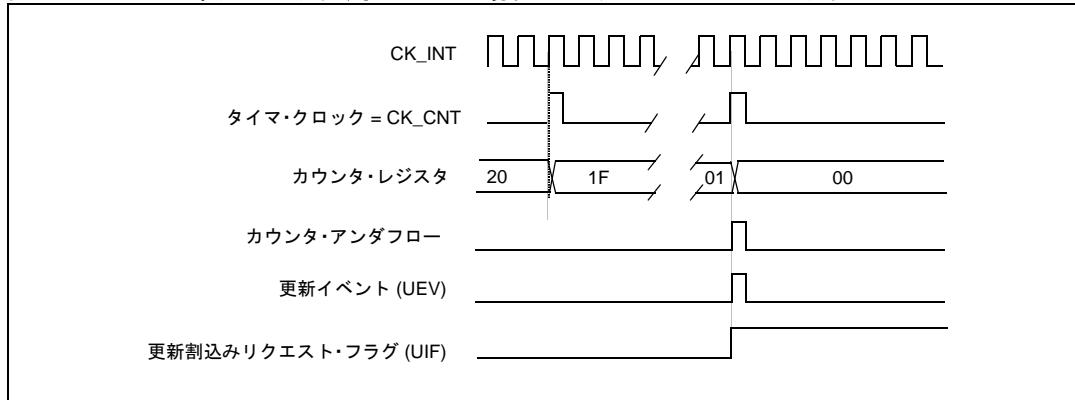
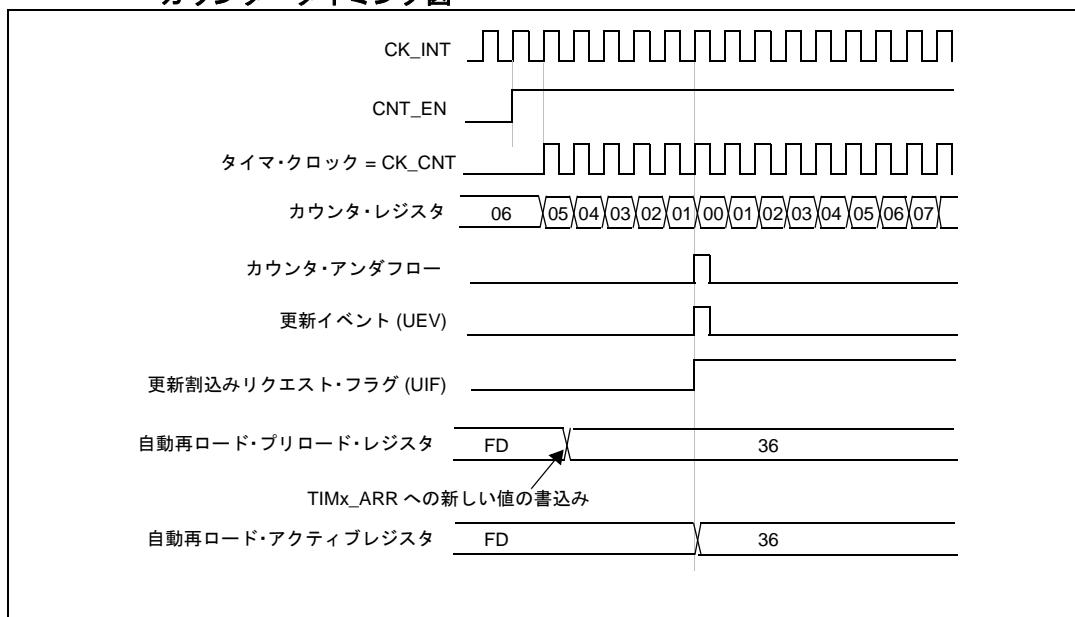
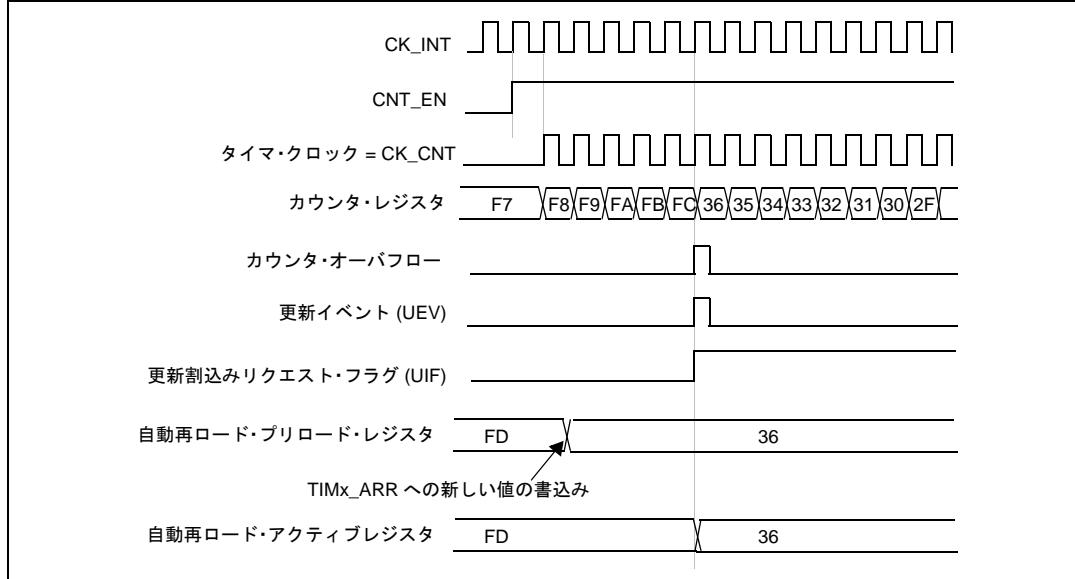


図 113. ARPE=1 (カウンタはアンダフロー) の時の更新イベントのカウンタ・タイミング図



参考資料

図 114. ARPE=1 (カウンタはオーバフロー) の時の更新イベントのカウンタ・タイミング図



13.3.3 クロック選択

カウンタ・クロックは次の信号のいずれかから供給することができます。

- 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ーラとして使用します。例えば、タイマ 1 がタイマ 2 のプリスケーラとして働くように設定することができます。詳細は [セクション: タイマを別のタイマのプリスケーラとするには \(315 ページ\)](#) を参照してください。

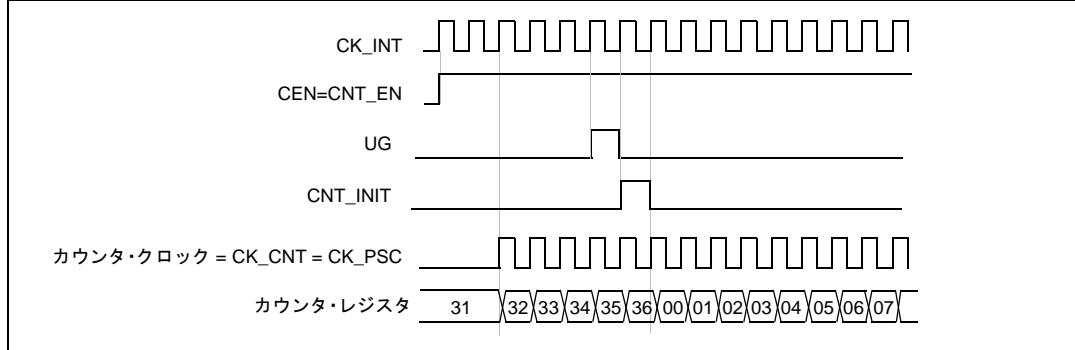
内部クロック (CK_INT)

スレーブ・モード・コントローラがディセーブル状態 (SMS=000) のときには、TIMx_CR1 レジスタの CEN、DIR ビットと TIMx_EGR レジスタの UG ビットが実際の制御ビットとなります。自動的にクリアされる UG ビット以外は、これらのビットはソフトウェアでのみ変更できます。CEN ビットに 1 が書込まれると、プリスケー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 115 には、プリスケー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が図示されています。

参考資料

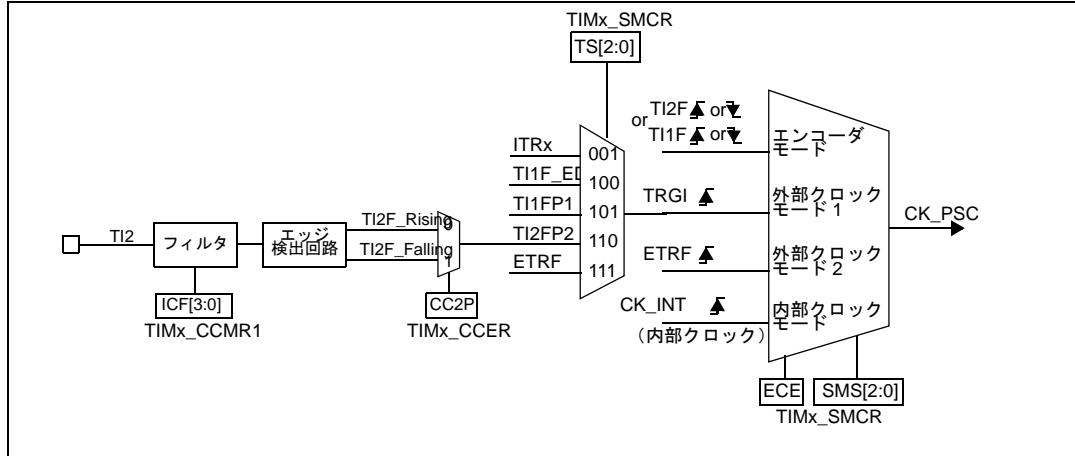
図 115. 内部クロック分周比が 1 の場合の、通常モードの制御回路



外部クロック・モード 1

このモードは TIMx_SMCR レジスタの SMS=111 の時に選択されます。カウンタは選択された入力の立上り、もしくは立下りエッジでカウントを行います。

図 116. TI2 外部クロックの接続例



一例として、アップカウンタが TI2 入力の立上りエッジでカウントを行う構成は、次のように設定します。

1. TIMx_CCMR1 レジスタの CC2S=01 とすることで、チャネル 2 が TI2 入力の立上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] ビットへの書き込みで、入力フィルタの応答時間を設定します。(フィルタを使用しないときには、IC2F=0000 としておきます。)

注意:

キャプチャ・プリスケーラはトリガとして使用されませんので、これに関する設定は不要です。

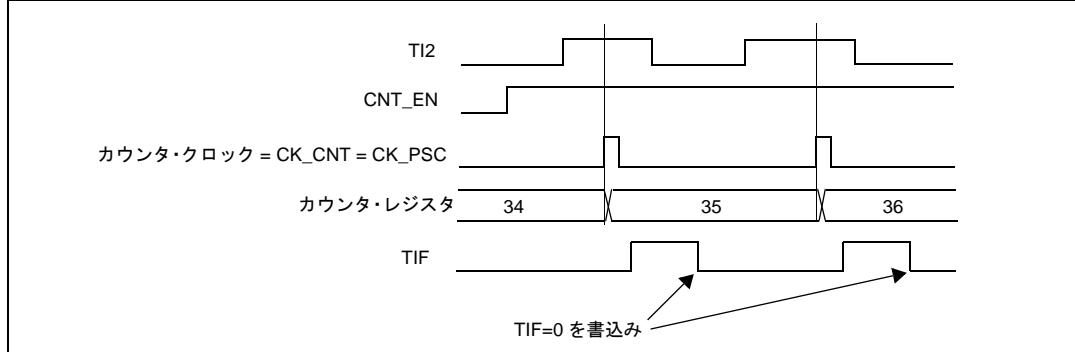
3. TIMx_CCER レジスタの CC2P=0 を書き込み、立上りエッジ極性を選択します。
4. TIMx_SMCR レジスタの SMS=111 を書き込み、タイマを外部クロック・モード 1 に設定します。
5. TIMx_SMCR レジスタの TS=110 を書き込み、トリガ入力として TI2 を選択します。
6. TIMx_CR1 レジスタの CEN=1 を書き込み、カウンタをイネーブルとします。

TI2 の立上りエッジが生じたとき、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 の立上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起因する遅延が生じます。

参考資料

図 117. 外部クロック・モード 1 の制御回路



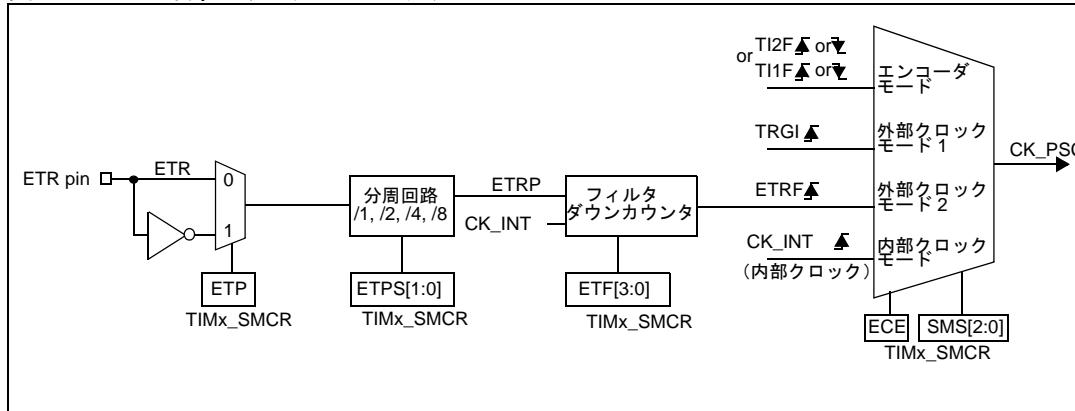
外部クロック・モード 2

このモードは TIMx_SMCR レジスタの ECE=1 とすることで選択されます。

カウンタは外部トリガ入力 ETR の立上りもしくは立下りエッジごとにカウントを行います。

図 118 には外部トリガ入力ブロックの概要が図示されています。

図 118. 外部トリガ入力ブロック



一例として、アップカウンタが ETR の 2 個の立上りエッジごとに 1 回カウントを行う構成は、次のように設定します。

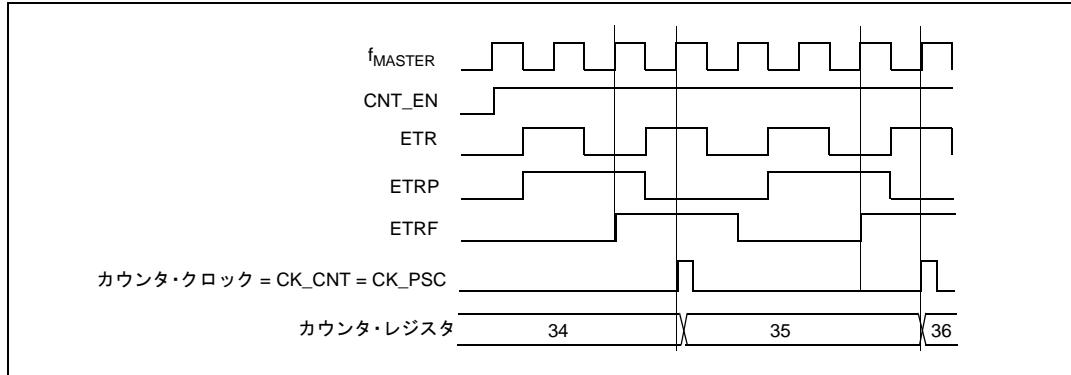
1. この例ではフィルタは必要とされませんので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書込みます。
2. TIMx_SMCR レジスタの ETPS[1:0]=01 とすることで、プリスケーラの設定を行います。
3. TIMx_SMCR レジスタの ETP=0 とし、ETR ピンで立上りエッジを検出するように設定します。
4. TIMx_SMCR レジスタの ECE=1 を書込み、外部クロック・モード 2 を有効とします
5. TIMx_CR1 レジスタの CEN=1 を書込み、カウンタをイネーブルとします。

カウンタは 2 個の立上りエッジごとに一回カウント動作を行います。

ETR の立上りエッジから実際のカウンタ・クロックまでの間には、ETRP 入力の再同期回路に起因する遅延が生じます。

参考資料

図 119. 外部クロック・モード 2 の制御回路



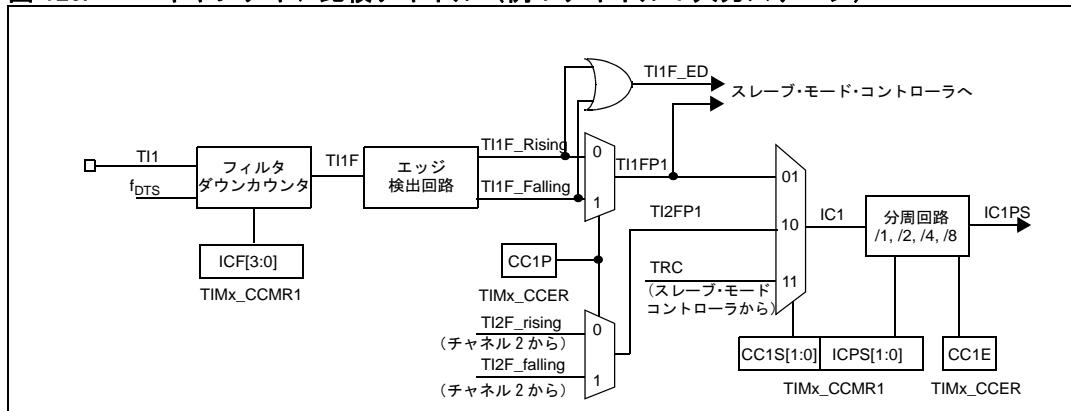
13.3.4 キャプチャ / 比較チャネル

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）、入力ステージ（デジタル・フィルタ、マルチプレクサ、プリスケーラ）、と出力ステージ（比較回路、出力制御）から構成されています。

次の図には、キャプチャ / 比較チャネルの概要が示されています。

入力ステージは対応する TIx 入力信号をサンプルし、フィルタリングを行った TIxF 信号を生成します。次に、エッジ検出回路と極性選択回路が、スレーブ・モード・コントローラへのトリガ入力もしくはキャプチャ・コマンドとして使用される信号 (TIxFPx) を生成します。この信号はプリスケーラを通して (ICxPS) キャプチャ・レジスタに渡されます。

図 120. キャプチャ / 比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは OCxRef (アクティブ・ハイ) として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

參考資料

図 121. キャプチャ / 比較チャネル 1 メイン回路

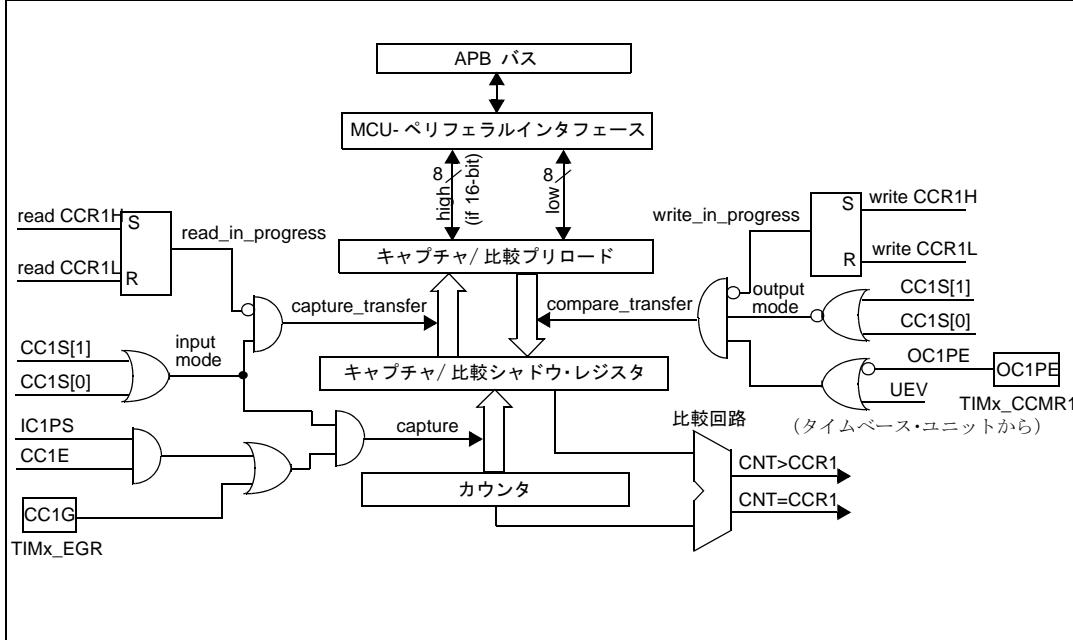
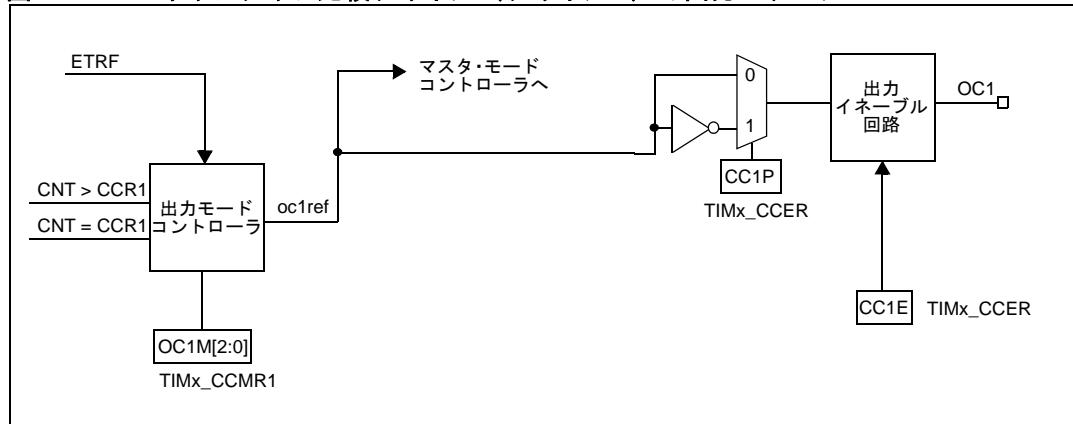


図 122. キャプチャ / 比較チャネル（チャネル 1）の出力ステージ



キャプチャ / 比較ブロックは 1 個のプリロード・レジスタと 1 個のシャドウ・レジスタで構成されています。ソフトウェアからの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、これがカウンタとの比較動作に使われます。

13.3.5 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 IC_x 信号の変化が検出されたときにカウンタの値をラッ奇するため、キャプチャ / 比較レジスタ (TIM_x_CCR_x) が使用されます。キャプチャが発生すると、TIM_x_SR レジスタの対応する CC_xIF フラグがセットされ、リクエスト発生が許可されれば、割込みリクエストもしくは DMA リクエストが発生します。CC_xIF フラグがすでにセットされているときにキャプチャ動作が発生すると、TIM_x_SR レジスタのオーバキャプチャ・フラグ CC_xOF がセットされます。CC_xIF フラグはソフトウェアでこのビットに 0 を書込むか、もしくは TIM_x_CCR_x レジスタの内容を読出す (16 ビットレジスタの場合には下位バイトの読出し) ことでクリアされます。CC_xOF のクリアは、このビットに 0 を書込むことで行います。

次の例は、TI1 入力が立上がったときに、カウンタの値を TIM_x_CCR1 にキャプチャするものです。操作は以下のとおりです。

- 有効な入力を選択します。TIM_x_CCR1 は TI1 入力とリンクされていなければならず、このためには TIM_x_CCMR1 レジスタの CC1S ビットに 01 を書込みます。CC1S の値が 00 から変化すると、チャネルは入力用に構成され、TIM_x_CCR1 レジスタは読出し専用となります。
- 入力フィルタでタイマに接続される信号に応じて必要な応答時間を設定します。(TIM_x_CCMRx レジスタの ICxF ビット) 入力信号が反転したとき、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、フィルタの応答時間として 5 クロック・サイクルを超える長さを指定しなければなりません。一例として、連続した 8 個の入力サンプル値が新しい値を持っているときに、TI1 で有効な変化があったとすることができます。これを設定するには、TIM_x_CCMR1 レジスタの IC1F ビットに 0011 を書込みます。(この場合、f_{DTS} 周波数によるサンプル)
- TI1 チャ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、TIM_x_CCER レジスタの CC1P ビットに 0 を書込みます。(この場合、立上りエッジの選択)
- 入力プリスケーラを設定します。今回の例では有効な信号変化ごとにキャプチャを行いますので、プリスケーラは無効とします。(TIM_x_CCMR1 レジスタの IC1PS ビットに 00 を書込みます。)
- カウンタからキャプチャ・レジスタへのキャプチャ・モードをイネーブルとします。これには、TIM_x_CCER レジスタの CC1E ビットをセットします。
- 必要ならば、関連する割込みリクエストを許可状態とするために TIM_x_DIER レジスタの CC1IE の設定を行うか、DMA リクエストのために TIM_x_DIER レジスタの CC1DE レジスタの設定を行います。

入力キャプチャ動作が行われると、

- アクティブな信号変化で、TIM_x_CCR1 レジスタにはカウンタの値が格納されます。
- CC1IF フラグ (割込みリクエスト) がセットされます。フラグがクリアされていない状態で 2 回目以降のキャプチャが発生すると CC1OF ビットも同時にセットされます。
- CC1IE ビットの設定に応じて、割込みリクエストが発生します。
- CC1DE ビットの設定に応じて、DMA リクエストが発生します。

オーバキャプチャの対応では、オーバキャプチャ・フラグの読出し前にデータを読出すようにしてください。これにより、フラグ読出し後、データ読出し前に発生し得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。

注意 : TIM_x_EGR レジスタの対応する CC_xG ビットをセットすることで、IC 割込みリクエストと DMA リクエストの一方、もしくは両方をソフトウェアで発生させることができます。

13.3.6 PWM 入力モード

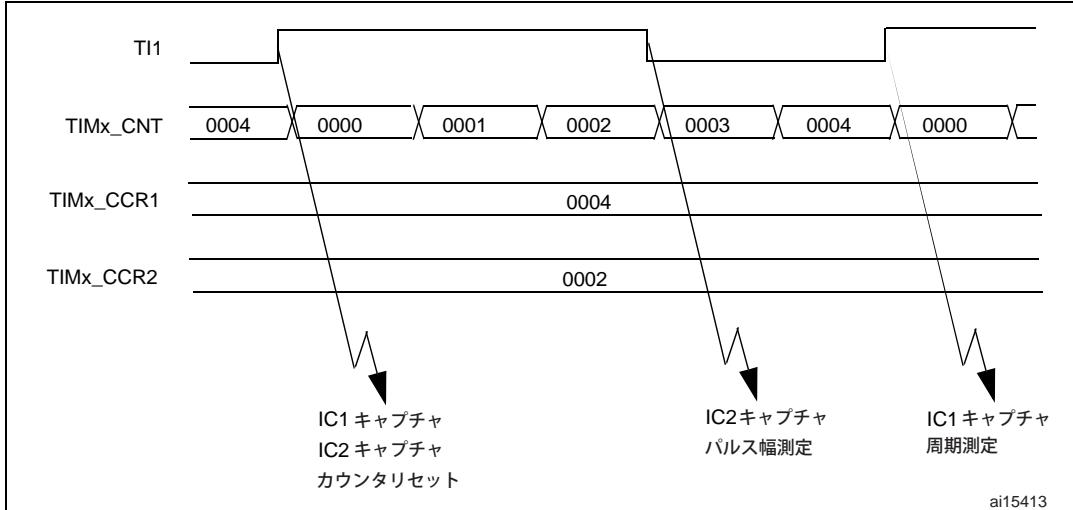
このモードは、入力キャプチャの特別な応用例です。操作手順は入力キャプチャ・モードと同様ですが、以下の点が異なっています。

- 2 個の IC_x 信号が同じ TI_x 入力に配置されます。
- この 2 個の IC_x 信号は、アクティブエッジが逆の極性に設定されます。
- 2 個の TI_xFP 信号の一方がトリガ入力として選択され、スレーブ・モード・コントローラがリセット・モードに設定されます。

一例として、TI1 上の PWM 信号の周期 (TIM_x_CCR1 レジスタ) とデューティサイクル (TIM_x_CCR2 レジスタ) を測定するものとします。このときには、以下の操作手順を実行します。(手順は CK_INT 周波数とプリスケーラの値によって、若干異なることがあります。)

- TIM_x_CCR1 のアクティブな入力を選択します。ここでは、TIM_x_CCMR1 レジスタの CC1S ビットに 01 を書込みます。(TI1 を選択)
- TI1FP1 のアクティブな極性を選択します。(TIM_x_CCR1 キャプチャとカウンタクリアの両方に使用します。) これには CC1P ビットに 0 を書込みます。(立上りエッジ)
- TIM_x_CCR2 のアクティブな入力を選択します。ここでは、TIM_x_CCMR1 レジスタの CC2S ビットに 10 を書込みます。(TI1 を選択)
- TI1FP2 のアクティブな極性を選択します。(TIM_x_CCR2 のキャプチャに使用します。) これには CC2P ビットに 1 を書込みます。(立下りエッジ)
- 有効なトリガ入力を選択します。ここでは、TIM_x_SMCR レジスタの TS ビットに 101 を書込みます。(TI1FP1 を選択)
- スレーブ・モード・コントローラをリセット・モードとします。これには、TIM_x_SMCR レジスタの SMS ビットを 100 とします。
- キャプチャ動作を開始します。これには、TIM_x_CCER レジスタの CC1E ビットと CC2E ビットに 1 を書込みます。

図 123. PWM 入力モードのタイミング



13.3.7 出力強制駆動モード

このモード (TIM_x_CCMRx レジスタの CC_xS=00) では、各出力比較信号 (OC_xREF、そして OC_x) 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ルーレベルとするには、対応する TIMx_OCMRx レジスタの OCxM ビットに 101 を書込みます。これにより、OCxREF はハイレベル (OCxREF は常にアクティブ・ハイです) となり、OCx は CCxP 極性ビットと逆のレベルとなります。

つまり、CCxP=0 (OCx アクティブ・ハイ) \Rightarrow OCx はハイレベルとなります。

TIMx_CCMRx レジスタの OCxM ビットに 100 を書込むことで、OCxREF 信号をローレベルとすることができます。

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。フラグの状態によっては割込みリクエストや DMA リクエストが発生することもあります。これに関しては、次の出力比較モードのセクションで説明されています。

13.3.8 出力比較モード

このモードは出力波形を制御する場合や、一定時間が経過したことを表示する場合に使用されます。

キャプチャ / 比較レジスタとカウンタの値が一致したとき、出力比較は次のように働きます。

- 対応する出力ピンに、プログラムで設定されている、出力比較モードとしての値 (TIMx_CCMRx レジスタの OCxM ビット) と出力極性 (TIMx_CCER レジスタの OCxP ビット) を適用した信号を出力します。値が一致したときに出力ピンは、固定レベルを維持 (OCxM=000)、アクティブルーレベルにセット (OCxM=001)、インアクティブルーレベルにセット (OCxM=010)、反転 (OCxM=011) などの状態とすることができます。
- 割込みステータス・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)
- 対応する割込みリクエストの許可ビット (TIMx_DIER レジスタの CCxEIE ビット) がセットされていれば、割込みリクエストが発生します。
- 対応する DMA リクエストの許可ビット (TIMx_DIER レジスタの CCxDE ビット) がセットされていれば、DMA リクエストが発生します。(DMA リクエストの選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます。)

TIMx_CCRx レジスタの設定においてプリロード・レジスタが使用されるか否かは TIMx_OCMRx レジスタの OCxPE ビットで指示できます。

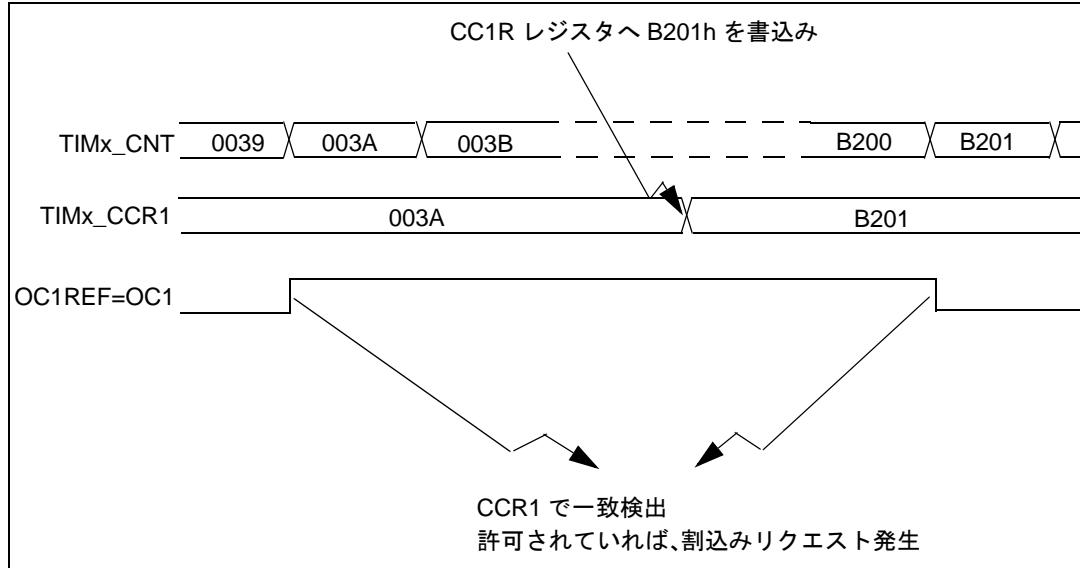
出力比較モードでは、更新イベント UEV は OCxREF 及び OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分となります。出力比較モードは 1 個のパルスを発生させる (単一パルス発生モード) ために使用することもできます。

操作手順 :

- カウンタ・クロックを選択します。(内部、外部、プリスケーラ)
- TIMx_ARR レジスタと TIMx_CCRx レジスタに希望のデータを書込みます。
- 割込みリクエスト /DMA リクエストの発生が必要であれば、CCxEIE ビット /CCxDE ビットをセットします。
- 出力モードを選択します。例えば、CNT が CCRx と一致したときに OCx 出力を反転させる、プリロード機能が使用されず、OCx が有効でアクティブ・ハイとするには、OCxM=011、OCxPE=0、CCxP=0、CCxE=1 を書込みます。
- カウンタの動作を開始するには、TIMx_CR1 レジスタの CEN ビットをセットします。

プリロード・レジスタが使用されていません (OCxPE=0) ので、TIMx_CCRx レジスタは出力波形を制御するためソフトウェアで常時変更ができます。(OCxPE=1 の時には次の更新イベント UEV でのみシャドウ・レジスタが変更されます。) タイミングの一例は図 124 に示されています。

図 124. 出力比較モード、OC1 出力のトグル.



13.3.9 PWM モード

パルス幅変調 (PWM) モードでは、周波数が TIMx_ARR レジスタで定まり、デューティ比が TIMx_CCRx レジスタで指定される信号を生成することができます。

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) を書込むことで、独自に選択することができます。このとき、TIMx_CCMRx レジスタの OCxPE ビットをセットすることで対応するチャネルのプリロード機能を有効としておくことが必要とされます。また、アップカウント・モードやセンターアライン・モードでは、TIMx_CR1 レジスタの ARPE ビットをセットすることで、自動再ロード・プリロード・レジスタも有効にしておくことが必要とされます。

プリロード・レジスタの内容は、更新イベントが発生したときにのみシャドウ・レジスタにコピーされるため、カウンタをスタートする前に、TIMx_EGR レジスタの UG ビットのセットで、すべてのレジスタを初期化しておくことが必要とされます。

OCx 信号がアクティブ・ハイであるか、アクティブ・ローであるかの極性は、TIMx_CCER レジスタの CCxP ビットを使用して、ソフトウェアで指定することができます。OCx 出力は TIMx_CCER レジスタの CCxE ビットによってイネーブルとされます。詳細は TIMx_CCERx レジスタの説明をお読みください。

PWM モード (1 または 2) では、TIMx_CCRx \leq TIMx_CNT かつ TIMx_CNT \leq TIMx_CCRx かを判別する (カウントの方向に依存します) ため、TIMx_CNT と TIMx_CCRx は常時比較されます。しかし、OCREF を次の PWM 周期まで ETR 信号による外部イベントでクリアすることができるという OCREF_CLR の機能を満たすため、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。
- 出力比較モード (TIMx_CCMRx レジスタの OCxM ビット) で、固定値 (OCxM=000) から PWM モードのいずれか (OCxM=110 または 111) への変更が行われたとき。

これにより、ソフトウェアによって動作中に PWM への切替えを行うことが可能となります。

タイマは PWM をエッジアライン・モードとセンターアライン・モードのいずれかで生成することができ、これは TIMx_CR1 レジスタの CMS ビットで指定します。

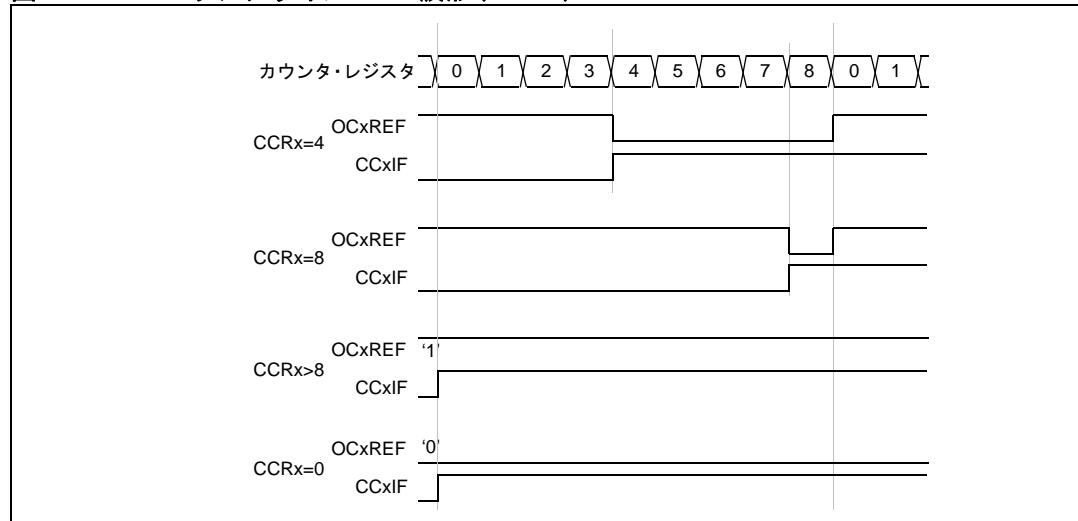
PWM エッジアライン・モード

アップカウント構成

TIMx_CR1 レジスタの DIR ビットが 0 のときには、アップカウント動作がアクティブとなっています。 [セクション：アップカウント・モード \(288 ページ\)](#) を参照してください。

以下の例では PWM モード 1 を想定します。PWM 基準信号 OCxREF は TIMx_CNT < TIMx_CCRx の間はハイレベルに、他の場合にはローレベルとなります。TIMx_CCRx の比較値が自動再ロード値 (TIMx_ARR レジスタの値) より大きい場合には、OCxREF は 1 を保ち、比較値が 0 のときには OCxREF は 0 を保ちます。[図 125](#) は TIMx_ARR=8 の時のエッジアライン PWM 波形の例を示しています。

図 125. 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 1 のときには、ダウンカウントがアクティブとなっています。 [ダウンカウントモード \(293 ページ\)](#) の説明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 の間はローレベルとなり、他の場合にはハイレベルとなります。比較値 TIMx_CCRx が TIMx_ARR レジスタの自動再ロード値より大きいときには、OCxREF は 1 を保ちます。このモードではデューティ比 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

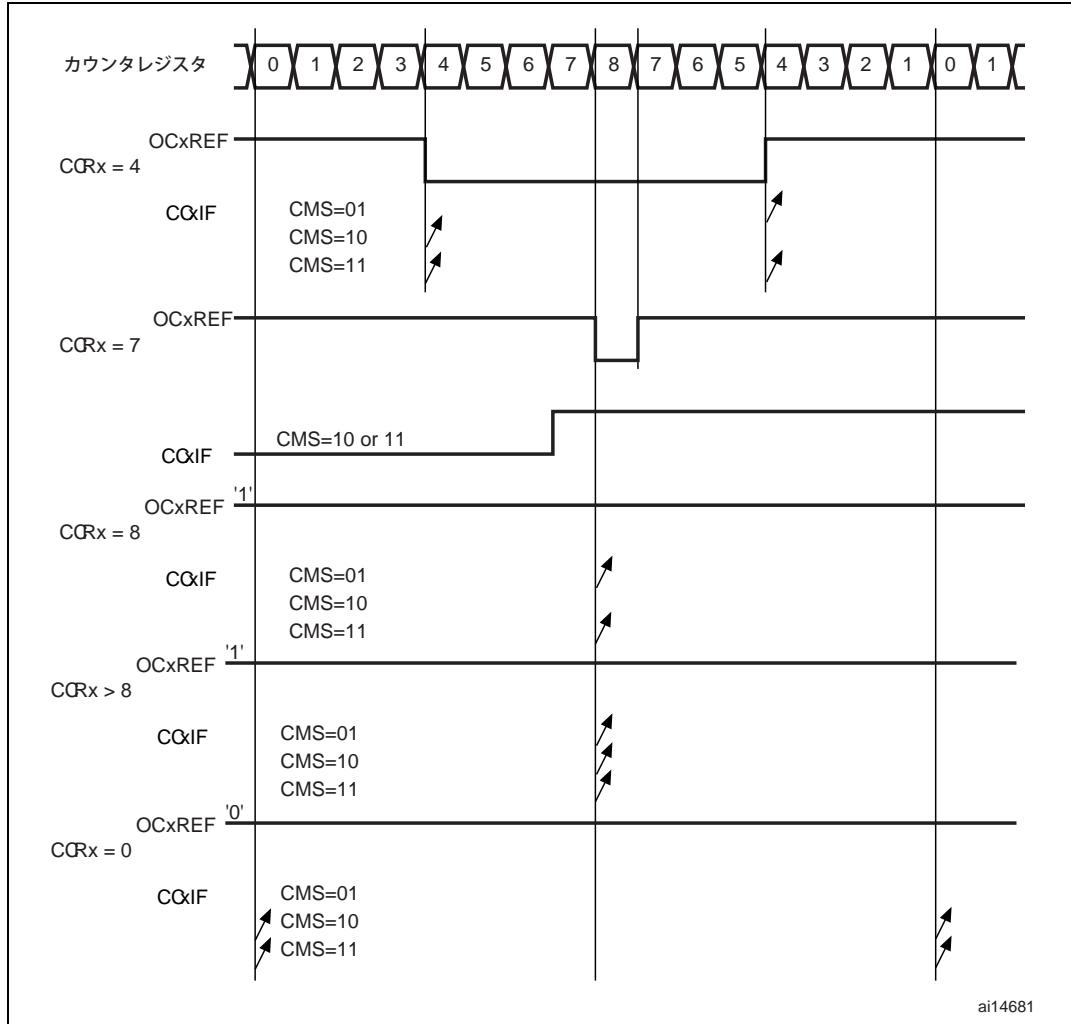
センターアライン・モードは TIMx_CR1 レジスタの CMS ビットが 00 ではないときにアクティブとなります。(OCxREF/OCx 信号に関しては、他の設定はすべて同じ効果を持っています。) CMS ビットの設定に応じて、カウンタがカウントアップを行っているとき、カウントダウンを行っているとき、カウントアップとカウントダウンの両方で、比較フラグがセットされます。TIMx_CR1 レジスタの方向ビット (DIR) はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。 [センターアラインモード \(アップ/ダウンカウント\) \(295 ページ\)](#)

[図 126](#) は次の条件の例で、センターアライン PWM 波形を示しています。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- TIMx_CR1 レジスタの CMS=01 で選択されるセンターアライン・モード 1 に対応して、カウンタがカウントダウンのときにフラグがセットされます。

参考資料

図 126. センターライン PWM 波形 (ARR=8)



センターライン・モードの使用について：

- センターライン・モードをスタートするときには、アップ・ダウン設定が使用されます。カウンタは TIMx_CR1 レジスタの DIR ビットに書込まれている値に応じて、カウントアップ、カウントダウンを実行します。さらに、DIR ビットと CMS ビットはソフトウェアで同時に変更することはできません。
- センターライン・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ため好ましくありません。特に、
 - 自動再ロード値より大きな値をカウンタに書込んだ場合 (TIMx_CNT > TIMx_ARR) には、方向は更新されません。カウントアップ中であれば、カウンタはその動作を続けます。
 - カウンタに 0 もしくは TIMx_ARR 値を書込んだときには、方向が更新されます。更新イベント UEV は発生しません。
- センターライン・モードを使用する確実な方法は、カウンタをスタートする直前に TIMx_EGR レジスタの UG ビットをセットしてソフトウェアで更新イベントを発生させ、動作中はカウンタへの書き込みを行わないことです。

13.3.10 単一パルス発生モード

単一パルス発生モード (OPM: One Pulse Mode) はこれまでに説明したモードの一ケースです。このモードでは、トリガに応じてカウンタをスタートさせ、設定した遅延時間の後に、設定した幅のパルスを発生させます。

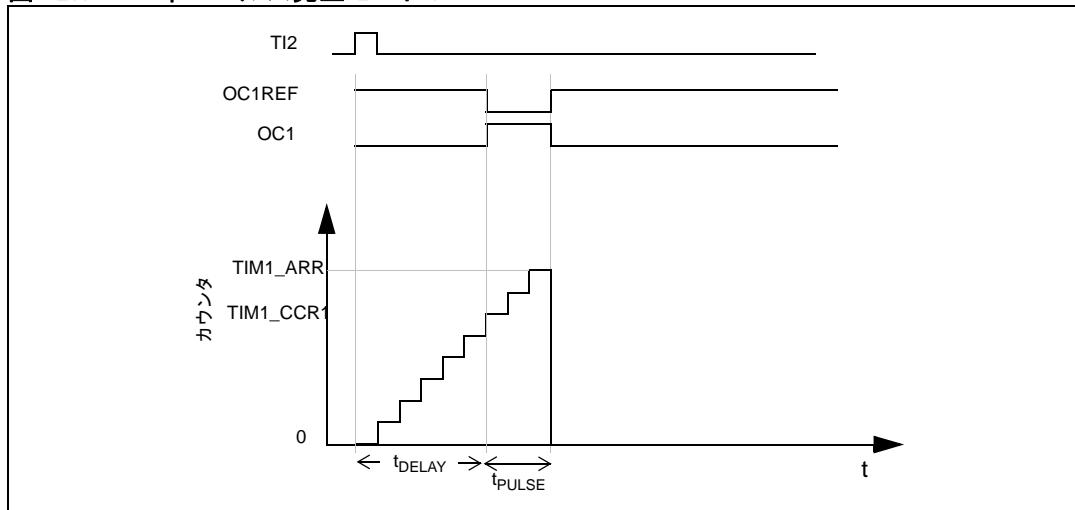
カウンタのスタートはスレーブ・モード・コントローラによって行うことができます。パルス波形の生成は出力比較モード、もしくは PWM モードによって行います。単一パルス発生モードは TIMx_CR1 レジスタの OPM ビットをセットすることで選択されます。このときにはカウンタは次の更新イベント UEV で自動的に停止します。

パルスは比較値がカウンタの初期値と異なるときにのみ正しく生成されます。カウンタをスタートさせる（タイマがトリガ信号を待つ状態となる）前に、次の設定が必要とされます。

アップカウント時 : $CNT < CCRx \leq ARR$ (特に $0 < CCRx$)

ダウンカウント時 : $CNT > CCRx$

図 127. 単一パルス発生モード .



一例として、TI2 入力ピンで立上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} 後に発生させることとします。

TI2FP2 をトリガ 1 として使用します。

- TI2FP2 を TI2 に配置するために、TIMx_CCMR1 レジスタの CC2S=01 とします。
- TI2FP2 で立上りエッジを検出します。TIMx_CCER レジスタの CC2P=0 を書込みます。
- TI2FP2 をスレーブ・モード・コントローラのトリガ (TRGI) とします。このため、TIMx_SMCR レジスタの TS=110 を書込みます。
- TI2FP2 でカウンタをスタートさせます。このためには、TIMx_SMCR レジスタの SMS ビットに 110 (トリガ・モード) を書込みます。

OPM 波形は比較値レジスタの設定（クロック周波数とプリスケーラに依存します）で定まります。

- 遅延時間 t_{DELAY} は TIMx_CCR1 レジスタに書込まれた値によって定まります。
- パルス幅 t_{PULSE} は自動再ロード・レジスタと比較値の差 ($TIMx_ARR - TIMx_CCR1$) によって定まります。
- カウンタの値が比較値と一致したとき 0 から 1 に変化し、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。これを行うためには、TIMx_CCMR1 レジスタの OC1M=111 をセットします。このとき TIMx_CCMR1 レジスタ

参考資料

の OC1PE=1 とし、TIMx_CR1 レジスタの ARPE=1 とすることでプリロードを有効とすることもできます。この場合には、比較値を TIMx_CCR1 レジスタに、自動再ロード値を TIMx_ARR レジスタに書込んだ上で、ソフトウェアで UG ビットをセットすることで更新イベントを発生させ、TI2 上の外部トリガ・イベントを待ちます。この例では CC1P には 0 を書き込みます。

上の例では TIMx_CR1 レジスタの DIR ビットと CMS ビットはローレベルでなければなりません。発生するパルスは 1 個ですので、次の更新イベント（カウンタが自動再ロード値に達して、カウンタの値が 0 に戻る時点）でカウンタが停止するよう、TIMx_CR1 レジスタの CPM ビットには 1 を書き込みます。

応用例：OCx 即時イネーブル：

単一パルス発生モードでは、TIx 入力のエッジ検出はカウンタをイネーブルとする CEN ビットをセットします。その後、カウンタと比較値レジスタの値との比較で出力が反転させられます。この動作には数クロック・サイクルが必要とされますので、実現可能な遅延時間の最小値は制限を受けます。

より小さな遅延時間が望まれるときには、TIMx_CCMRx レジスタの OCxFE ビットをセットすることができます。このとき OCxREF (及び OCx) は、比較動作を行うことなく、トリガに応じてレベルが定まります。このときのレベルは比較が一致したときと同じです。OCxFE はチャネルが PWM1 もしくは PWM2 モードに設定されたときにのみ有効です。

13.3.11 外部イベントによる OCxREF 信号のクリア

指定されたチャネルの OCxREF 信号は ETRF 入力をハイレベルとすることでローレベルとすることができます。(TIMx_CCMRx レジスタの対応する OCxCE イネーブル・ビットが 1 に設定されているとき) 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。

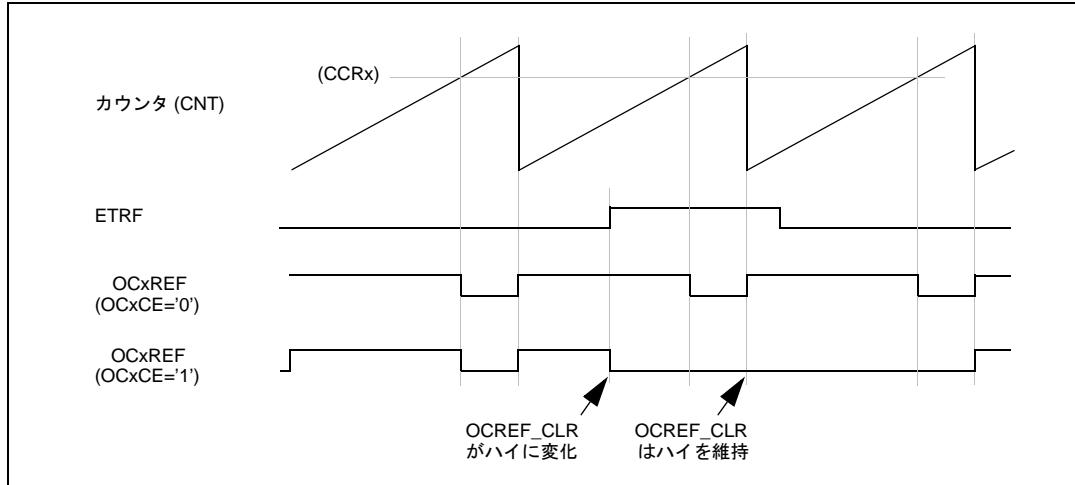
この機能は出力比較モードと PWM モードでのみ使用でき、強制駆動モードでは利用できません。

例えば、通常動作のため、OCxREF 信号を比較回路の出力に接続することができます。この場合、ETR は次のように設定しなければなりません。

1. 外部トリガ・プリスケーラはオフとしておきます。このため、TIMx_SMCR レジスタの ETPS[1:0] を 00 とします。
2. 外部クロック・モード 2 はディセーブルとします。TIMx_SMCR レジスタの ECE ビットは 0 にセットします。
3. 外部トリガ極性 (ETP) と外部トリガフィルタ (ETF) は必要に応じた設定とします。

[図 128](#) はイネーブル・ビット OCxCE の値と、ETRF 入力がハイレベルになったときの OCxREF の挙動を示しています。この例では TIMx タイマは PWM モードで動作しています。

図 128. TIMx OCxREF のクリア動作



13.3.12 エンコーダ・インターフェース・モード

エンコーダ・インターフェース・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するのであれば `SMS=001` に、`TI1` エッジのみであれば `SMS=010` に、`TI1` と `TI2` の両方のエッジをカウントするのであれば `SMS=011` とします。

`TI1` と `TI2` の極性は `TIMx_CCER` レジスタの `CC1P` ビットと `CC2P` ビットで指定できます。必要ならば、入力フィルタの設定も同じように行います。

ふたつの入力 `TI1` と `TI2` はインクリメンタル・エンコーダとのインターフェースのために使用されます。動作は表 56 を参照してください。カウンタがイネーブル状態 (`TIMx_CR1` レジスタの `CEN` ビットが 1) であれば、`TI1FP1` もしくは `TI2FP2` (フィルタを通し、極性を整えた `TI1` 及び `TI2` 信号) の有効なレベル変化がカウンタのクロックとして使用されます。両入力の変化の順序が、カウントパルスと同時に方向信号を与えます。変化の順序によってカウンタがカウントアップ、カウントダウンを行うとき、`TIMx_CR1` レジスタの `DIR` ビットはこれに応じてハードウェアで変更されます。カウントが `TI1` のみ、`TI2` のみ、`TI1` と `TI2` の両方のいずれで行われるかにかかわらず、`DIR` ビットの値は `TI1`、`TI2` いずれかの変化を検出したときに設定されます。

エンコーダ・インターフェース・モードは、単に方向の選択を含む外部クロック方式として動作します。カウンタは 0 と `TIMx_ARR` レジスタの自動再ロード値の間で（方向に応じて 0 から `ARR`、もしくは `ARR` から 0 の間で）連続的にカウントを行います。したがって、`TIMx_ARR` は他の場合と同じく開始前に設定しておかねばなりません。キャプチャ、比較、プリスケーラ、繰返しカウンタ、トリガ出力の機能は通常と同じく機能しています。

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、その値は常にエンコーダの位置を示しています。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。下の表は `TI1` と `TI2` の変化の組合わせをまとめたものです。（`TI1` と `TI2` は同時に変化しないと想定しています。）

参考資料

表 56. カウントの方向とエンコーダ信号

有効エッジ	他の信号に対するレベル (TI1FP1 for TI2, TI2FP2 for TI1)	TI1FP1 信号		TI2FP2 信号	
		立上り	立下り	立上り	立下り
TI1 のみをカウント	High	Down	Up	カウントなし	カウントなし
	Low	Up	Down	カウントなし	カウントなし
TI2 のみをカウント	High	カウントなし	カウントなし	Up	Down
	Low	カウントなし	カウントなし	Down	Up
TI1 と TI2 をカウント	High	Down	Up	Up	Down
	Low	Up	Down	Down	Up

外部のインクリメンタル・エンコーダは外部にインタフェース回路を持つことなく、MCU に直接に接続されます。ただし、通常はコンパレータがエンコーダの作動出力をデジタル信号に変換するために使用されます。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エンコーダの 3 番目の出力は、外部割込み入力に接続することができ、カウンタのリセットをトリガすることができます。

図 129 はカウント信号の生成と方向制御を含むカウンタの動作を示しています。同時に、両エッジを選択したとき、ジッタがどのように除かれるかも示されています。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ます。この例では、次の設定がなされていることを想定しています。

- 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 (TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1)
- CC2P=0 (TIMx_CCER レジスタ、TI1FP2 非反転、TI1FP2=TI2)
- SMS=011 (TIMx_SMCR レジスタ、両入力で、立上りと立下りの両エッジをカウント)
- CEN=1 (TIMx_CR1 レジスタ、カウンタ・イネーブル)

図 129. エンコーダ・インターフェース・モードにおけるカウンタの動作例.

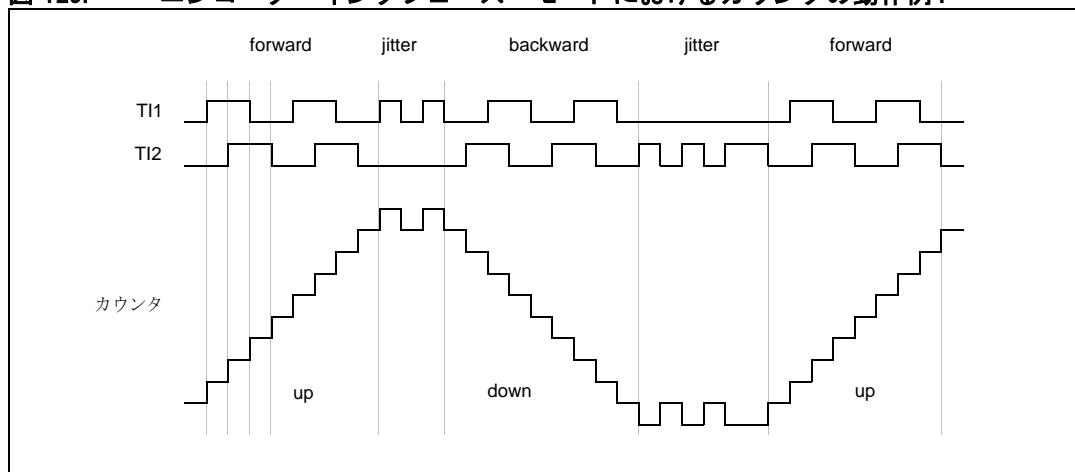
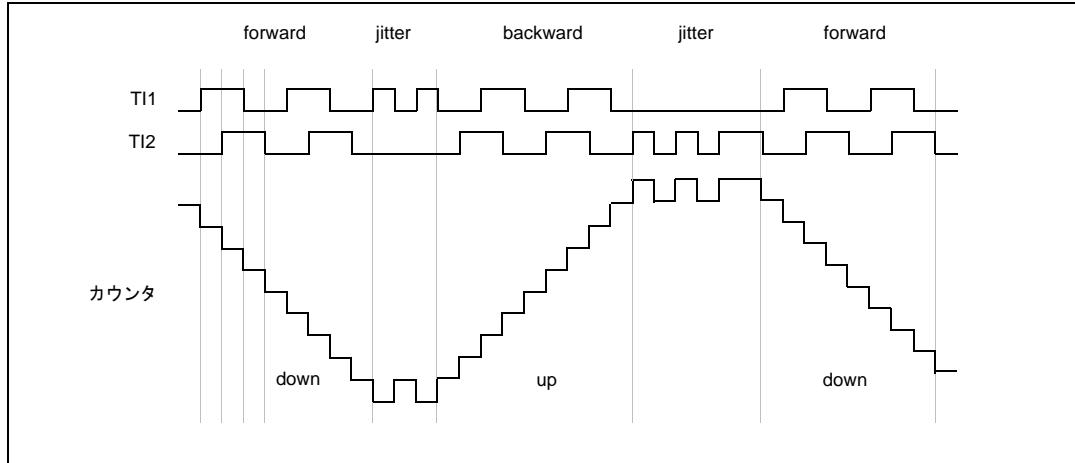


図 130 では、TI1FP1 の極性を反転した時のカウンタの動作を示しています。これは上の例とほぼ同じ設定ですが、CC1P=1 である点が異なっています。

図 130. IC1FP1 の極性を反転したエンコーダ・インターフェース・モードの例。



エンコーダ・インターフェース・モードに設定されたカウンタはセンサの位置を表示しています。キャプチャ・モードに設定した二番目のタイマを使用し、ふたつのエンコーダイベントの時間差を計ることで、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はこの目的に使用できます。ふたつのイベントの時間差によっては、カウンタを任意の時点で読出することもできます。ハードウェアに余裕があれば、カウンタの値を 3 番目の入力キャプチャ・レジスタにラッチすることができます。(さらに、キャプチャが周期的なときには、キャプチャ信号の発生に別のタイマを使うことも考えられます。) また、リアルタイム・クロックによって発生する DMA リクエストでこの値を読出することもできます。

13.3.13 タイマ入力 XOR 機能

TIM1_CR2 レジスタの TI1S ビットで、チャネル 1 の入力フィルタを、TIMx_CH1 から TIMx_CH3 を入力とする 3 入力 XOR ゲートの出力に接続することができます。

XOR 出力は、トリガや入力キャプチャといったタイマ入力機能のすべてで利用することができます。この機能の利用の一例は [セクション 12.3.18 \(253 ページ\)](#) に記されているホール・センサ・インターフェースです。

13.3.14 タイマの外部トリガ同期

TIMx タイマは、リセット・モード、ゲート・モード、トリガ・モードといった幾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

トリガ入力イベントによって、カウンタとそのプリスケーラの初期化を行うことができます。さらに、TIMx_CR1 レジスタの URS ビットがローレベルであれば、更新イベントが発生します。これによりプリロード機能をもつレジスタ、TIMx_ARR と TIMx_CCRx が更新されます。

次の例では、TI1 入力の立上りエッジに応じて、アップカウンタがクリアされます。

- チャネル 1 が TI1 の立上りエッジを検出するように設定します。入力フィルタの応答時間を設定します。(この例ではフィルタ機能は必要ありません。IC1F=0000 としておきます。) トリガによる動作ではキャプチャ・プリスケーラは使用されませんので、設定の必要はありません。CC1S ビットは入力キャプチャの信号を選択するのみで、TIMx_CCMR1 レジスタの

参考資料

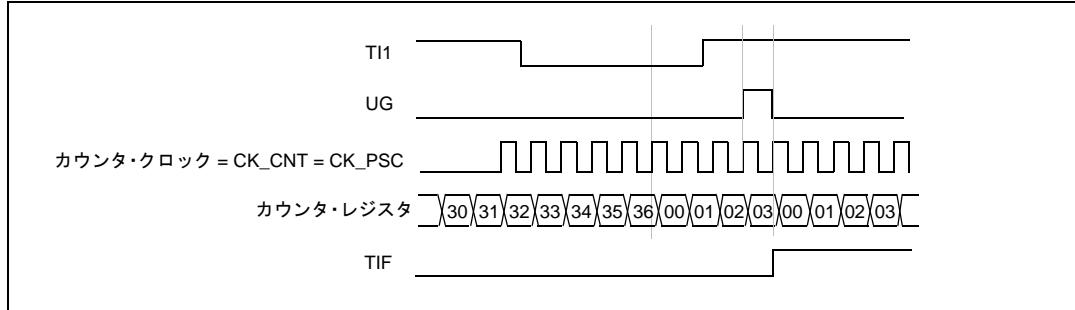
CC1S=01 とします。トリガの極性（立上りエッジを検出）を設定するために、TIMx_CCER レジスタの CC1P=0 とします。

- TIMx_SMCR レジスタの SMS=100 で、タイマをリセット・モードとします。
TIMx_SMCR レジスタの TS=101 で入力信号として TI1 を選択します。
- TIMx_CR1 レジスタの CEN=1 により、カウンタをスタートします。

カウンタは内部クロックでカウントを開始し、TI1 の立上りエッジが生じるまで通常の動作を行います。TI1 が立ち上がったとき、カウンタはクリアされ、0 からカウントが再開します。同時に、トリガフラグ（TIMx_SR レジスタの TIF ビット）がセットされ、割込みリクエスト、もしくは DMA リクエストが発生します。（TIMx_DIER レジスタの TIE もしくは TDE ビットで許可されている場合。）

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ています。TI1 の立上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 入力回路における同期クロックの変更によって生じます。

図 131. 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、カウンタが動作します。

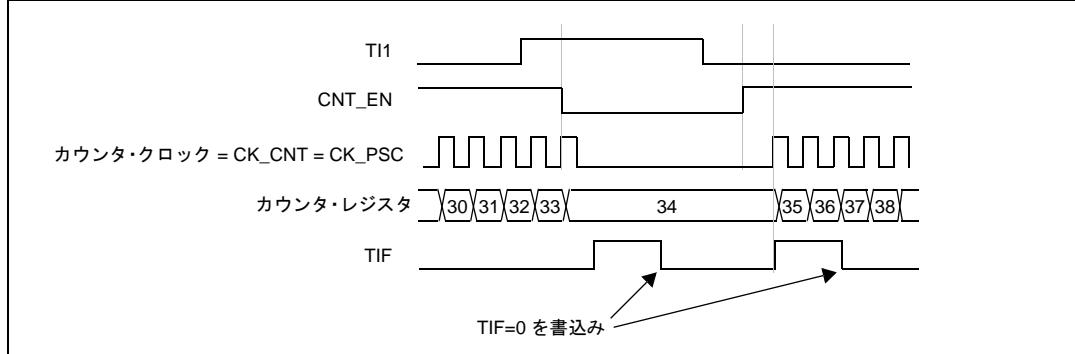
以下の例では、TI1 入力がローレベルのときにのみアップカウンタがカウント動作を行います。

- チャネル 1 で TI1 のローレベルを検出するよう設定します。入力フィルタの応答時間を設定します。（この例ではフィルタは必要ありませんので IC1F=0000 のままとします。）キャプチャ・プリスケーラはトリガでは使用されませんので、設定の必要はありません。TIMx_CCMR1 レジスタの CC1S ビットは CC1S=01 とし、入力キャプチャの信号を選択します。TIMx_CCER レジスタの CC1P=1 を書き込み、極性（ローレベルを検出）を有効とします。
- タイマをゲート・モードに設定します。このためには、TIMx_SMCR レジスタで SMS=101 とします。TIMx_SMCR レジスタの TS=101 で TI1 を入力信号として選択します。
- TIMx_CR1 レジスタの CEN=1 により、カウンタをイネーブルとします。（ゲート・モードではトリガ入力のレベルにかかわらず、CEN=0 ではカウンタはスタートしません。）

カウンタは TI1 がローレベルの時にカウントを開始し、TI1 がハイレベルとなると停止します。カウンタのスタート時とストップ時の両方で TIMx_SR レジスタの TIF フラグがセットされます。

TI1 の立上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 入力がクロックに同期させられることに起因します。

図 132. 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、カウンタがスタートします。

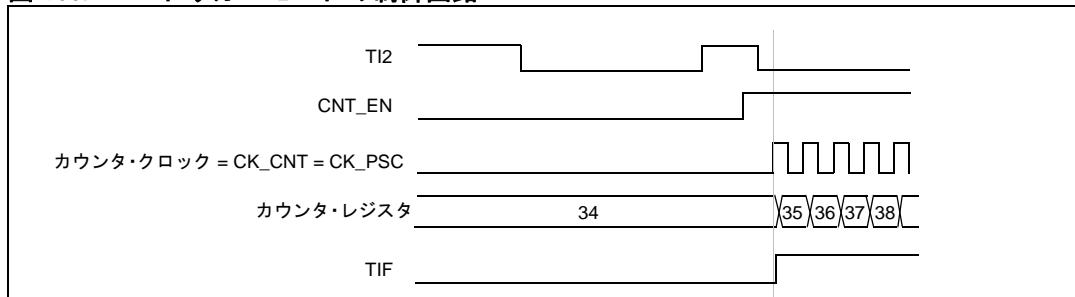
次の例では、TI2 入力の立上りエッジに応じて、アップカウンタがスタートします。

- チャネル 2 で TI2 の立上りエッジを検出するよう設定します。入力フィルタの応答時間を設定します。(この例ではフィルタは必要ありませんので IC2F=0000 のままとします。) キャプチャ・プリスケーラはトリガでは使用されませんので、設定の必要はありません。TIMx_CCMR1 レジスタの CC2S ビットは CC2S=01 とし、入力キャプチャの信号を選択します。TIMx_CCER レジスタの CC2P=1 を書込み、極性（ローレベルを検出）を有効とします。
- タイマをトリガ・モードに設定します。このためには、TIMx_SMCR レジスタで SMS=110 とします。TIMx_SMCR レジスタの TS=110 で TI2 を入力信号として選択します。

TI2 の立上りエッジでカウンタは内部クロックでカウントを開始し、同時に TIF フラグをセットします。

TI2 の立上りエッジから実際にカウンタがスタートするまでの遅延は、TI2 入力がクロックに同期させられることに起因します。

図 133. 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（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）とともに使用することができます。このとき、ETR 信号は外部クロック入力として使用され、別の入力がトリガ入力（リセット・モード、ゲート・モード、及びトリガ・モード）として使用されます。TIMx_SMCR レジスタの TS ビットの設定では、TRGI として ETR を選択しないようにしてください。

以下の例では、TI1 の立上りエッジが発生したとき、アップカウンタは ETR 信号の立上りエッジによるカウントアップを開始します。

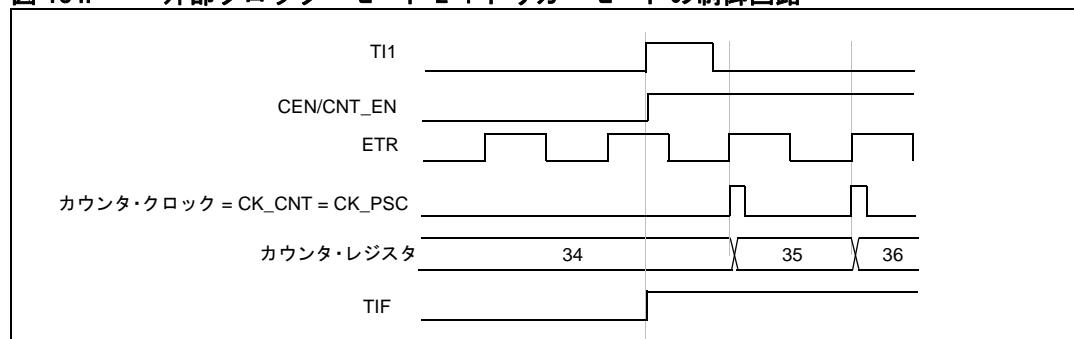
参考資料

1. TIMx_SMCR レジスタで次の設定を行い、外部トリガ入力回路の構成を行います。
 - ETF = 0000: フィルタなし
 - ETPS=00: プリスケーラは無効
 - ETP=0: ETR の立上りエッジを検出。ECE=1 で外部クロック・モード 2 を有効とします。
2. TI1 の立上りエッジを検出するため、チャネル 1 で次の設定を行います。
 - IC1F=0000: フィルタなし
 - トリガに関してはキャプチャ・プリスケーラは使用されませんので設定は不要です。
 - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャ信号を選択します。
 - TIMx_CCER レジスタの CC1P=0 で、極性を有効とします。(同時に、立上りエッジの選択を指示します。)
3. タイマをトリガ・モードとするため、TIMx_SMCR レジスタの SMS=110 とします。入力として TI1 を選択するため、TIMx_SMCR レジスタの TS=101 を設定します。

TI1 の立上りエッジでカウンタがインバブルとなり、TIF フラグがセットされます。カウンタは ETR の立上りエッジをカウントします。

ETR 信号の立上りエッジから実際にカウンタがカウントされるまでに、ETRP 入力がクロックに同期することに起因する遅延が存在します。

図 134. 外部クロック・モード 2 + トリガ・モードの制御回路



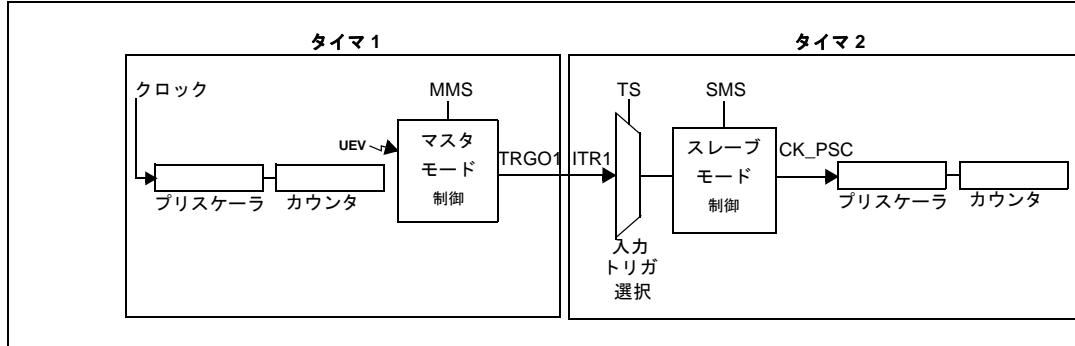
13.3.15 タイマ間の同期

タイマ間で同期をとったりチエイニングを行うなど、TIMx タイマを内部で相互に接続することができます。タイマがマスター・モードに設定されたときには、スレーブ・モードとされた別のタイマのリセット、スタート、ストップを行ったり、クロックとして機能させることができます。

次の図はトリガ選択とマスター・モード選択ブロックの概要を示しています。

タイマを別のタイマのプリスケーラとするには

図 135. マスタ / スレーブタイマの例



一例として、図 135 のようにタイマ 1 がタイマ 2 のプリスケーラとして動作させることができます。このためには次の操作を行います。

- タイマ 1 をマスタ・モードとし、更新イベント UEV ごとに周期的にトリガ信号を出力するように設定します。TIM1_CR2 レジスタの MMS=010 としたときには、更新イベントが発生するごとに TRGO1 出力に立上りエッジが出力されます。
- タイマ 1 の出力 TRGO1 をタイマ 2 に接続し、タイマ 2 が内部トリガ ITR1 を使用するスレーブ・モードに設定します。これは TIM2_SMCR レジスタの TS ビットを TS=001 することで設定されます。
- 次に、TIM2_SMCR レジスタで SMS=111 とすることで、スレーブ・モード・コントローラを外部クロック・モード 1 とします。これによりタイマ 2 は、タイマ 1 の周期的なトリガ信号の立上りエッジ（タイマ 1 カウンタのオーバフローに対応しています）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で、両タイマをイネーブル状態とします。

注意： タイマ 1 のトリガ出力として OCx が選択されたとき (MMS=1xx) には、タイマ 2 カウンタのクロックとして、その立上りエッジが使用されます。

タイマが別のタイマをイネーブルにするには

この場合には、タイマ 1 の出力比較 1 でタイマ 2 をイネーブルとします。接続は図 135 を参照してください。タイマ 1 の OC1REF がハイレベルの時にのみ、分周された内部クロックをタイマ 2 がカウントします。両方のカウンタ・クロック周波数は CK_INT をプリスケーラで 3 分周したものとします。 $(f_{CK_CNT} = f_{CK_INT}/3)$

- タイマ 1 をマスタ・モードとし、出力比較 1 の出力 (OC1REF) 信号をトリガ出力とします。これは TIM1_CR2 レジスタの MMS=100 で設定します。
- タイマ 1 の OC1REF の波形を設定します。(TIM1_CCMR1 レジスタ)
- タイマ 2 がタイマ 1 からのトリガ入力を受取るようにします。これは TIM2_SMCR レジスタの TS=001 で設定します。
- タイマ 2 をゲート・モードに設定します。(TIM2_SMCR レジスタの SMS=101)
- TIM2_CR1 レジスタの CEN ビットに 1 を書込み、タイマ 2 をイネーブルとします。
- TIM1_CR1 レジスタの CEN ビットに 1 を書込み、タイマ 1 をスタートさせます。

注意： タイマ 2 のクロックはカウンタ 1 と同期していません。このモードではタイマ 2 カウンタのイネーブル信号を操作するのみです。

参考資料

図 136. タイマ 1 の OC1REF によるタイマ 2 のゲート操作

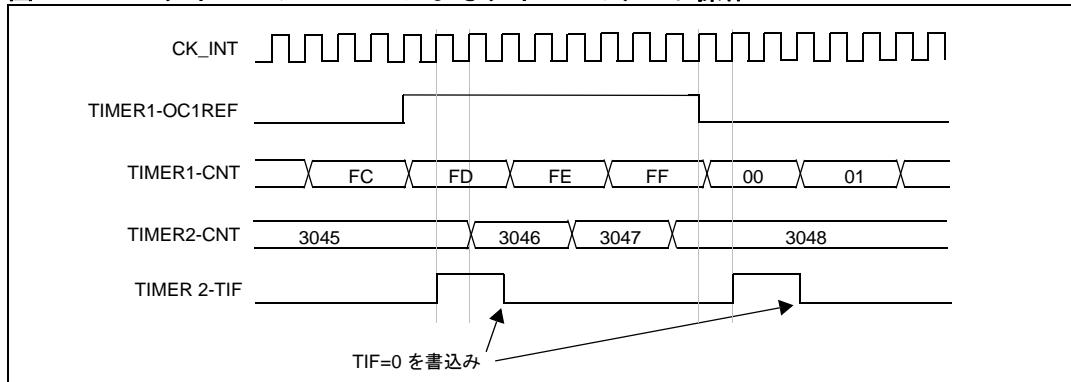


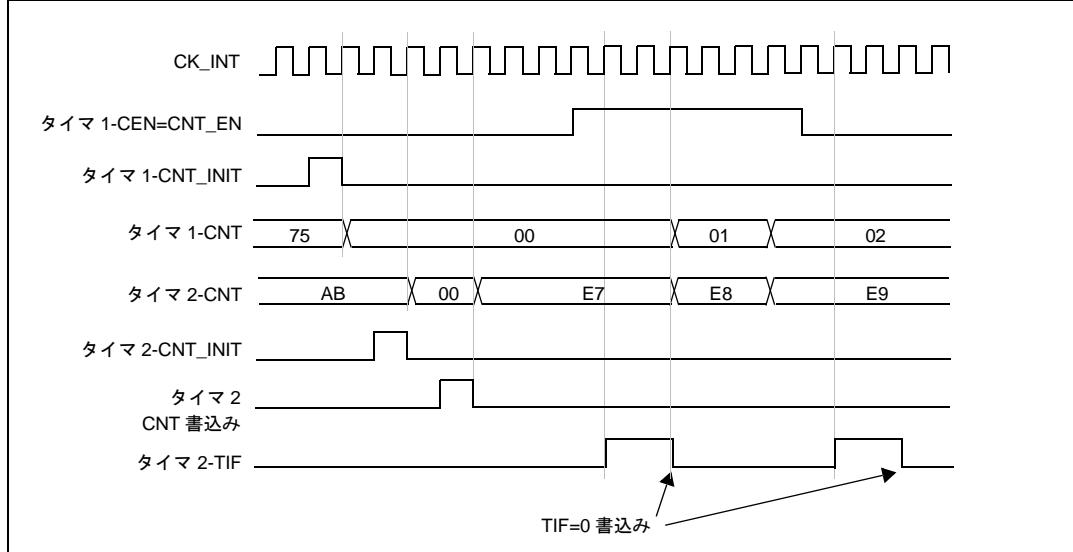
図 136 の例では、タイマ 2 のカウンタとプリスケーラはスタート前に初期化されておらず、その時点の値からカウントを開始しています。タイマ 1 をスタートする前に、両タイマに任意の値を設定しておくことができます。タイマは TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで容易にリセットすることができます。

次の例では、タイマ 1 とタイマ 2 を同期させます。タイマ 1 はマスターとして 0 からカウントを開始します。タイマ 2 はスレーブとして 0xE7 からスタートします。プリスケーラの分周比は両タイマで同じとします。TIM1_CR1 レジスタの CEN ビットが 0 でタイマ 1 がディセーブルの時にはタイマ 2 は停止します。

- タイマ 1 をマスター・モードとし、出力比較 1 の出力 (OC1REF) 信号をトリガ出力とします。(TIM1_CR2 レジスタの MMS=100)
- タイマ 1 の OC1REF 波形を指定します。(TIM1_CCMR1 レジスタ)
- タイマ 2 がタイマ 1 からの入力トリガを使用するようにします。(TIM2_SMCR レジスタの TS=001)
- タイマ 2 をゲート・モードとします。(TIM2_SMCR レジスタの SMS=101)
- TIM1_EGR レジスタの UG ビットに 1 を書き込み、タイマ 1 をリセットします。
- TIM2_EGR レジスタの UG ビットに 1 を書き込み、タイマ 2 をリセットします。
- タイマ 2 のカウンタ (TIM2_CNTL) に 0xE7 を書き込み、タイマ 2 を 0xE7 初期化します。
- TIM2_CR1 レジスタの CEN ビットを 1 とし、タイマ 2 をイネーブルとします。
- TIM1_CR1 レジスタの CEN ビットを 1 とし、タイマ 1 をスタートします。
- TIM1_CR1 レジスタの CEN ビットを 0 とし、タイマ 1 をストップします。

参考資料

図 137. タイマ 1 のイネーブルによるタイマ 2 のゲート制御



参考資料

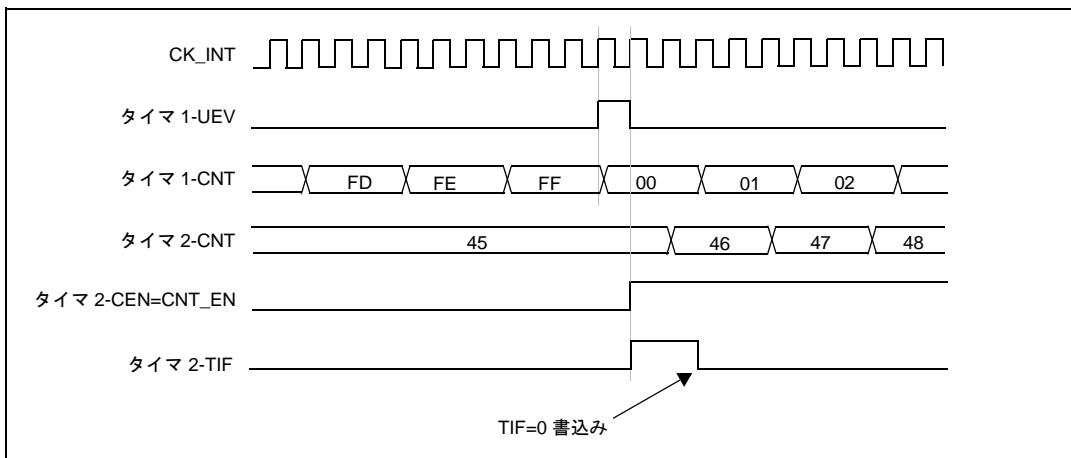
タイマが別のタイマをスタートさせるには

この場合、タイマ 1 の更新イベントでタイマ 2 をイネーブルにセットします。接続については図 135 を参照してください。タイマ 2 はタイマ 1 が発生させた更新イベントから、その時点の値（ゼロである必要はありません）から分周された内部クロックでカウントを開始します。タイマ 2 がトリガ信号を受取ると CEN ビットは自動的にセットされ、TIM2_CR1 レジスタの CEN ビットに 0 を書込むまでカウント動作が行われます。両方のカウンタのクロック周波数は CK_INT をプリスケーラで 3 分周した値とします。

($f_{CK_CNT} = f_{CK_INT}/3$)

- タイマ 1 をマスター・モードとし、更新イベント (UEV) をトリガ出力とします。
(TIM1_CR2 レジスタの MMS=010)
- タイマ 1 の周期を設定します。
(TIM1_ARR レジスタ)
- タイマ 2 がタイマ 1 からのトリガ入力を受取る設定とします。
(TIM1_SMCR レジスタの TS=001)
- タイマ 2 をトリガ・モードとします。
(TIM2_SMCR レジスタの SMS=110)
- TIM1_CR1 レジスタの CEN ビットに 1 を書き込み、タイマ 1 をスタートします。

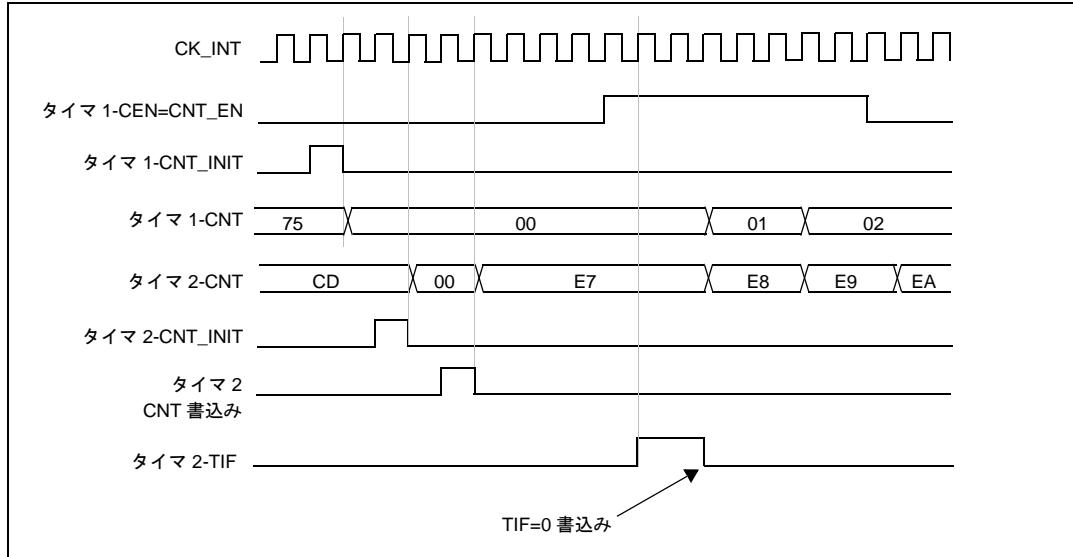
図 138. タイマ 1 のアップデートによるタイマ 2 のトリガ



先の例では、カウントを開始する前に両カウンタの初期化が可能でした。図 139 は図 138 と同じ構成ですが、タイマ 2 でゲート・モードではなくトリガ・モードとしています。
(TIM2_SMCR レジスタの SMS=110)

参考資料

図 139. タイマ 1 のイネーブルによるタイマ 2 のトリガ



タイマを別のタイマのプリスケーラとするには

この例は、タイマ 1 をタイマ 2 のプリスケーラとして動作するよう、設定します。接続は [図 135](#) を参照してください。

- タイマ 1 をマスタ・モードとし、更新イベント (UEV) をトリガ出力とします。
(TIM1_CR2 レジスタの MMS=010) これにより、カウンタのオーバフローごとに周期的な信号が出力されます。
- タイマ 1 の周期を設定します。(TIM1_ARR レジスタ)
- タイマ 2 がタイマ 1 からのトリガ入力を受取るよう設定します。
(TIM2_SMCR レジスタの TS=001)
- タイマ 2 を外部クロック・モード 1 とします。(TIM2_SMCR レジスタの SMS=111)
- TIM2_CR1 レジスタの CEN ビットに 1 を書込み、タイマ 2 をスタートします。
- TIM1_CR1 レジスタの CEN ビットに 1 を書込み、タイマ 1 をスタートします。

共通の外部トリガによって 2 個のタイマを同時にスタートさせるには

この例では、TI1 入力の立上りエッジでタイマ 1 をイネーブルとし、タイマ 1 がイネーブルとなったことを受けてタイマ 2 をイネーブルとします。接続は [図 135](#) を参照してください。カウンタの同時性を確保するため、タイマ 1 はマスタ / スレーブ・モードとします。(TI1 に対してはスレーブ動作、タイマ 2 に対してはマスタ動作)

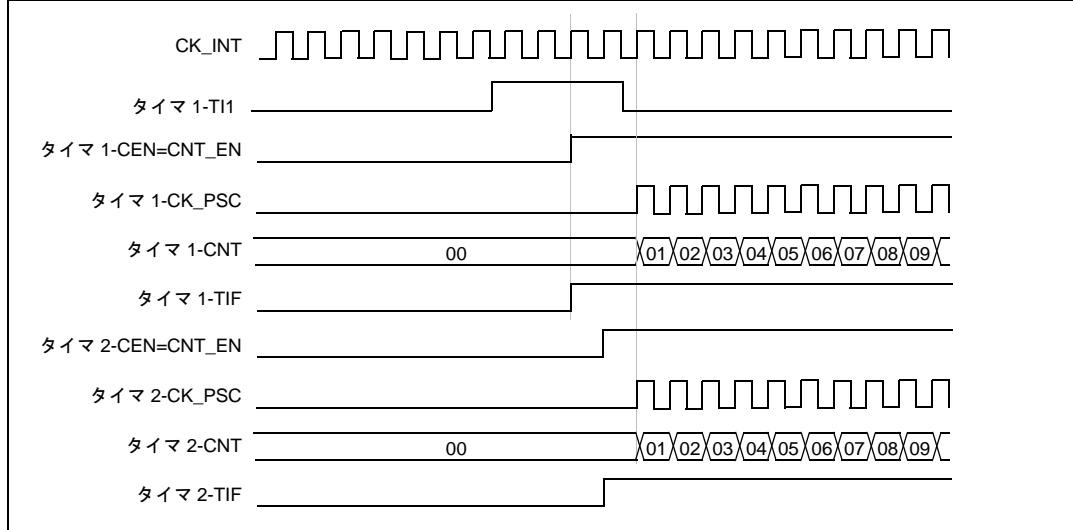
- タイマ 1 をマスタ・モードとし、トリガ出力として ENABLE を出力します。
(TIM1_CR2 レジスタの MMS=001)
- タイマ 1 がスレーブ・モードで TI1 をトリガ入力として受取るよう設定します。
(TIM1_SMCR レジスタの TS=100)
- タイマ 1 をトリガ・モードとします。(TIM1_SMCR レジスタの SMS=110)
- タイマ 1 をマスタ / スレーブ・モードとします。(TIM1_SMCR レジスタの MSM=1)
- タイマ 2 がタイマ 1 からのトリガ入力を受取る設定とします。
(TIM2_SMCR レジスタの TS=001)
- タイマ 2 をトリガ・モードとします。(TIM2_SMCR レジスタの SMS=1)

参考資料

タイマ 1 の TI1 で立上りエッジが検出されると、両方のカウンタは同時に内部クロックによるカウントをスタートします。このとき両方のタイマで TIF フラグがセットされます。

注意： この例では両方のタイマは（それぞれの UG ビットをセットすることで）スタート前に初期化されます。両方のカウンタは 0 からスタートしますが、カウンタ・レジスタ (TIMx_CNT) の一方もしくは両方に書き込みを行うことで、カウント値に差をつけることもできます。マスター/スレーブ・モードでは、タイマ 1 の CNT_EN と CK_PSC の間に遅延時間が存在します。

図 140. タイマ 1 の TI1 入力によるタイマ 1 とタイマ 2 のトリガ.



13.3.16 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 (Cortex-M3 コアは停止状態) に入ったとき、TIMx カウンタが通常どおりに動作するか、停止するかを、DBG モジュールの DBG_TIMx_STOP 構成ビットで指定することができます。詳細は [セクション 26.15.2: タイマ、ウォッチドッグ、bxCAN と PEC に対するデバッグ・サポート](#) に記載されています。

13.4 TIMx レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#) を参照してください。

13.4.1 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	rw	rw	rw	rw		

Bits 15:10 予約済み、常に 0 が読出されます。

Bits 9:8 CKD: クロック分周比

これらのビットは、タイマ・クロック (CK_INT) と、デジタル・フィルタ (ETR、TIx) のサンプリング・クロックとの間の分周比を指定します。

00: $T_{DTS} = T_{ck_tim}$

01: $T_{DTS} = 2 * T_{ck_tim}$

10: $T_{DTS} = 4 * T_{ck_tim}$

11: 予約値、設定できません。

Bit 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を使用しません。

1: TIMx_ARR レジスタはバッファを使用します。

Bits 6:5 CMS: センターライン・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 / カウントダウンを行います。

01: センターライン・モード 1、カウンタは交互にアップ / ダウンを行います。出力設定 (TIMx_CCMRx レジスタの CCxS=00) の各チャネルの出力比較割込みリクエスト・フラグはカウントダウンの時にのみセットされます。

10: センターライン・モード 2、カウンタは交互にアップ / ダウンを行います。出力設定 (TIMx_CCMRx レジスタの CCxS=00) の各チャネルの出力比較割込みリクエスト・フラグはカウントアップの時にのみセットされます。

11: センターライン・モード 3、カウンタは交互にアップ / ダウンを行います。出力設定 (TIMx_CCMRx レジスタの CCxS=00) の各チャネルの出力比較割込みリクエスト・フラグはカウントアップ / カウントダウンの両方でセットされます。

注: カウンタがイネーブル状態 (CEN=1) のとき、エッジアライン・モードからセンターライン・モードに切替えることはできません。

Bit 4 DIR: 方向

0: カウンタはカウントアップを行います。

1: カウンタはカウントダウンを行います。

注: このビットはセンターライン・モードもしくはエンコーダ・モードに設定されているときには読出し専用となります。

Bit 3 OPM: 単一パルス発生モード

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動作を停止 (CEN ビットをクリア) します。

参考資料

Bit 2 URS: 更新リクエストの要因

更新イベント (UEV) を発生する元となる信号を選択するため、ソフトウェアでセット / クリアができます。

0: 次のイベントのいずれかが更新割込みリクエストもしくは DMA リクエストを発生します。(リクエストが許可されているときのみ)

-カウンタ・オーバフロー / アンダフロー

-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指示

1: カウンタのオーバフロー / アンダフローのみが更新割込みリクエストもしくは DMA リクエストを発生します。(リクエストが許可されているときのみ)

Bit 1 UDIS: 更新動作ディセーブル

更新イベント (UEV) 発生をイネーブル / ディセーブルとするため、ソフトウェアでセット / クリアができます。

0: UEV イネーブル。更新イベント (UEV) は次のいずれかで発生します。

-カウンタ・オーバフロー / アンダフロー

-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指示

このとき、バッファを持つレジスタではプリロード値がロードされます。

1: UEV ディセーブル。更新イベントは発生せず、個々のシャドウ・レジスタ (ARR, PSC, CCRx) がそれぞれの値を保持しています。カウンタとプリスケーラは UG ビットがセットされたとき、及びスレーブ・モード・コントローラからハードウェアリセットが指示されたときに初期化されます。

Bit 0 CEN: カウンタ・イネーブル

0: カウンタ・ディセーブル

1: カウンタ・イネーブル

注: 外部クロック、ゲート・モード及びエンコーダ・モードは CEN ビットが事前にソフトウェアでセットされているときのみ動作します。トリガ・モードではハードウェアで自動的に CEN ビットをセットします。

単一パルス発生モードでは、更新イベント発生時に CEN は自動的にクリアされます。

13.4.2 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			

Bits 15:8 予約済み、常に 0 が読出されます。.

Bit 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 及び CH3 ピンが TI1 入力に接続されます。(XOR 接続)

[セクション 12.3.18: ホール・センサ・インターフェース \(253 ページ\)](#) を参照してください。

参考資料

RM0008

汎用タイマ (TIMx)

Bits 6:4 MMS: 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・モードから送られる情報 (TRGO) の選択を行います。

000: **Reset** – TIMx_EGR レジスタの UG ビットがトリガ出力 (TRGO) として使用されます。トリガ入力によってリセットが発生したとき (スレーブ・モード・コントローラがリセット・モードに設定されているとき) TRGO 信号は実際のリセットから遅れて発生します。

001: **Enable** – 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。これは幾つかのタイマを同時にスタートさせるときや、スレーブタイマの動作範囲を制御する時に有効です。カウンタ・イネーブル信号は、CEN 制御ビットとゲート・モードに設定されているときのトリガ入力との論理和で発生します。

カウンタ・イネーブル信号がトリガ入力で制御されているとき、マスタ / スレーブ・モードに設定されているときを除いて TRGO 出力には遅延が存在します。(TIMx_SMCR レジスタの MSM ビットの説明をお読みください。)

010: **Update** – 更新イベントがトリガ出力 (TRGO) として選択されます。例えばマスタタイマがスレーブタイマのプリスケーラとなるときに使用されます。

011: **Compare Pulse** – キャプチャもしくは比較一致で CC1F フラグのセット条件が発生したとき (フラグがすでにハイレベルであっても)、正のパルスがトリガ出力 (TRGO) から送出されます。

100: **Compare** – OC1REF 信号がトリガ出力 (TRGO) として使用されます。

101: **Compare** – OC2REF 信号がトリガ出力 (TRGO) として使用されます。

110: **Compare** – OC3REF 信号がトリガ出力 (TRGO) として使用されます。

111: **Compare** – OC4REF 信号がトリガ出力 (TRGO) として使用されます。

Bit 3 CCDS: キャプチャ / 比較 DMA 選択

0: CCx イベント発生時に CCx DMA リクエストが送出されます。

1: 更新イベントの発生時に CCx DMA リクエストが送出されます。

Bits 2:0 予約済み、常に 0 が読出されます。

13.4.3 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]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 15 ETP: 外部トリガ極性

このビットは外部トリガ動作に ETR と \overline{ETR} のいずれが使用されるかを選択します。

0: ETR は反転されず、ハイレベルもしくは立上りエッジがアクティブとなります。

1: ETR を反転し、ローレベルもしくは立下りエッジがアクティブとなります。



参考資料

Bit 14 ECE: 外部クロック有効

このビットは外部クロック・モード 2 を有効とします。

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。ETRF 信号のアクティブエッジがカウンタのクロックとなります。

注 1: ECE ビットをセットすると、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) と同じ効果となります。

注 2: 外部クロック・モード 2 と同時に次のスレーブ・モードを指定できます。リセット・モード、ゲート・モード、トリガ・モード。ただし、TRGI は ETRF に接続されていてはいけません。(TS ビットは 111 ないことが必要。)

注 3: 外部クロック・モード 1 と外部クロック・モード 2 が同時に有効に設定されると、外部クロック入力は ETRF となります。

Bits 13:12 ETPS: 外部トリガ・プリスケーラ

外部トリガ信号 ETRP の周波数は最大で CK_INT 周波数の 1/4 に制限されます。ETRP の周波数を低下させるためにプリスケーラが使用でき、高速の外部クロックが与えられる場合に役立ちます。

00: プリスケーラ・オフ

01: ETRP 分周比 2

10: ETRP 分周比 4

11: ETRP 分周比 8

Bits 11:8 ETF[3:0]: 外部トリガフィルタ

これらのビットは 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・フィルタの応答時間を定めます。デジタル・フィルタはイベントカウンタを主体としており、出力に有効な変化をもたらすには N 回のイベント発生が必要とされます。

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: $f_{SAMPLING} = f_{DTS}/32$ 、 $N=8$

Bit 7 MSM: マスター/スレーブ・モード

0: 無効

1: トリガ入力 (TRGI) は、タイマとそのスレーブ (TRGO に接続) との間で完全な同期をとるため、ある程度の遅延をもつて有効となります。これは複数のタイマを 1 個の外部イベントで同期する際に役立ちます。

参考資料

RM0008

汎用タイマ (TIMx)

Bits 6:4 TS: トリガ選択

これらのビットはカウンタを同期させるために使用されるトリガ入力を選択します。内部入力の接続に関しては、個々の製品仕様書を参照してください。

- 000: 内部トリガ 0 (ITR0)、TIM1
- 001: 内部トリガ 1 (ITR1)、TIM2
- 010: 内部トリガ 2 (ITR2)、TIM3
- 011: 内部トリガ 3 (ITR3)、TIM4
- 100: TI1 エッジ検出回路 (TI1F_ED)
- 101: フィルタ後のタイマ入力 1 (TI1FP1)
- 110: フィルタ後のタイマ入力 2 (TI2FP2)
- 111: 外部トリガ入力 (ETRF)

各タイマへの ITRx の意味の詳細は [表 57: TIMx 内部トリガ接続 \(325 ページ\)](#) を参照してください。

注: 設定変更時に誤ったエッジ検出が行われるのを防ぐため、これらのビットの変更は、これが使用されていないとき（例えば SMS=000）に限られます。

Bit 3 予約済み、常に 0 が読出されます。

Bits 2:0 SMS: スレーブ・モード選択

外部信号が選択されると、その信号の指定された極性が、トリガ信号 (TRGI) のアクティブエッジとなります。

- 000: スレーブ・モード・ディセーブル - CEN = 1 であれば、内部クロックが直接にプリスケーラの入力となります。
- 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 / カウントダウンを行います。
- 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 / カウントダウンを行います。
- 011: エンコーダ・モード 3 - カウンタは TI1FP1 のエッジと TI2FP2 のエッジの両方でカウントを行います。アップ / ダウンはエッジ検出時の他方の信号レベルで決められます。
- 100: リセット・モード - 選択されたトリガ入力 (TRGI) の立上りエッジがカウンタをリセットし、レジスタの更新イベントを発生します。
- 101: ゲート・モード - トリガ入力 (TRGI) がハイレベルの時、カウンタにクロックが供給されます。トリガ信号がローレベルとなるとカウンタは停止します。（リセットはされません。）カウンタのスタートとストップの両方の制御ができます。
- 110: トリガ・モード - トリガ入力 (TRGI) の立上りエッジでカウンタがスタートします。（このときリセットはされません。）カウンタのスタートのみが制御されます。
- 111: 外部クロックモード 1 - 選択された信号 (TRGI) の立上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 (TS=100) ときには、ゲート・モードを選択することはできません。TI1F_ED は TI1F の信号変化ごとに 1 パルスを出力します。他方、ゲート・モードはトリガ信号のレベルに依存した動作を行います。

表 57. 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	TIM1	TIM8	TIM3	TIM4
TIM3	TIM1	TIM2	TIM5	TIM4
TIM4	TIM1	TIM2	TIM3	TIM8
TIM5	TIM2	TIM3	TIM4	TIM8



参考資料

13.4.4 DMA/ 割込みイネーブル・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	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

Bit 15 予約済み、常に 0 が読出されます。

Bit 14 TDE: トリガ DMA リクエスト許可

- 0: トリガ DMA リクエスト禁止
- 1: トリガ DMA リクエスト許可

Bit 13 予約済み、常に 0 が読出されます。

Bit 12 CC4DE: キャプチャ／比較 4 DMA リクエスト許可

- 0: CC4 DMA リクエスト禁止
- 1: CC4 DMA リクエスト許可

Bit 11 CC3DE: キャプチャ／比較 3 DMA リクエスト許可

- 0: CC3 DMA リクエスト禁止
- 1: CC3 DMA リクエスト許可

Bit 10 CC2DE: キャプチャ／比較 2 DMA リクエスト許可

- 0: CC2 DMA リクエスト禁止
- 1: CC2 DMA リクエスト許可

Bit 9 CC1DE: キャプチャ／比較 1 DMA リクエスト許可

- 0: CC1 DMA リクエスト禁止
- 1: CC1 DMA リクエスト許可

Bit 8 UDE: 更新 DMA リクエスト許可

- 0: 更新 DMA リクエスト禁止
- 1: 更新 DMA リクエスト許可

Bit 7 予約済み、常に 0 が読出されます。

Bit 6 TIE: トリガ割込みリクエスト許可

- 0: トリガ割込みリクエスト禁止
- 1: トリガ割込みリクエスト許可

Bit 5 予約済み、常に 0 が読出されます。

Bit 4 CC4IE: キャプチャ／比較 4 割込みリクエスト許可

- 0: CC4 割込みリクエスト禁止
- 1: CC4 割込みリクエスト許可

Bit 3 CC3IE: キャプチャ／比較 3 割込みリクエスト許可

- 0: CC3 割込みリクエスト禁止
- 1: CC3 割込みリクエスト許可

Bit 2 CC2IE: キャプチャ／比較 2 割込みリクエスト許可

- 0: CC2 割込みリクエスト禁止
- 1: CC2 割込みリクエスト許可

参考資料

RM0008

汎用タイマ (TIMx)

Bit 1 **CC1IE:** キャプチャ / 比較 1 割込みリクエスト許可

- 0: CC1 割込みリクエスト禁止
- 1: CC1 割込みリクエスト許可

Bit 0 **UIE:** 更新割込みリクエスト許可

- 0: 更新割込みリクエスト禁止
- 1: 更新割込みリクエスト許可

13.4.5 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				CC4 OF	CC3 OF	CC2 OF	CC1 OF		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	rc_w0	rc_w0

Bit 15:13 予約済み、常に 0 が読出されます。

Bit 12 **CC4OF:** キャプチャ / 比較 4 オーバキャプチャ・フラグ

CC1OF の説明を参照してください。

Bit 11 **CC3OF:** キャプチャ / 比較 3 オーバキャプチャ・フラグ

CC1OF の説明を参照してください。

Bit 10 **CC2OF:** キャプチャ / 比較 2 オーバキャプチャ・フラグ

CC1OF の説明を参照してください。

Bit 9 **CC1OF:** キャプチャ / 比較 1 オーバキャプチャ・フラグ

このフラグのセットは、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に、ハードウェアで行われます。フラグのクリアはソフトウェアでこのビットに 0 を書込むことで行います。

0: オーバキャプチャ状況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にもかかわらず、カウンタの値が TIMx_CCR1 レジスタに書込まれました。

Bits 8:7 予約済み、常に 0 が読出されます。

Bit 6 **TIF:** トリガ割込みリクエスト・フラグ

このフラグはトリガ・イベント時（スレーブ・モード・コントローラのゲート・モード以外の全モードで TRGI 入力でアクティイブエッジが検出されたとき、もしくはゲート・モードでいずれかのエッジが検出されたとき）にハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: トリガ・イベントは発生していません。

1: トリガ割込みリクエストがペンディング状態です。

Bit 5 予約済み、常に 0 が読出されます。

Bit 4 **CC4IF:** キャプチャ / 比較 4 割込みリクエスト・フラグ

CC1IF の説明を参照してください。

Bit 3 **CC3IF:** キャプチャ / 比較 3 割込みリクエスト・フラグ

CC1IF の説明を参照してください。

Bit 2 **CC2IF:** キャプチャ / 比較 2 割込みリクエスト・フラグ

CC1IF の説明を参照してください。



参考資料

Bit 1 CC1IF: キャプチャ／比較 1 割込みリクエスト・フラグ

チャネル CC1 が出力設定のとき：

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でセットされます。(センターライン・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。) フラグのクリアはソフトウェアで行います。

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。

チャネル CC1 が入力設定のとき：

このフラグは、キャプチャが行われたときにハードウェアでセットされます。フラグはソフトウェアで TIMx_CCR1 レジスタを読み込むことでクリアされます。

0: 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 で所定のエッジが検出されました。)

Bit 0 UIF: 更新割込みリクエスト・フラグ

このビットは更新イベント発生時にハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: 更新は発生していません。

1: 更新割込みリクエストがペンディング状態です。このビットはレジスタが更新されたときにハードウェアでセットされます。

-TIMx_CR1 レジスタの UDIS=0 で、繰返しダウンカウンタに関してオーバフローもしくはアンダーフローが検出されたとき。(REP_CNT=0 で更新イベントが発生します。)

-TIMx_CR1 レジスタの URS=0 かつ UDIS=0 で、TIMx_EGR レジスタの UG ビットを使用し、ソフトウェアで CNT の初期化が行われたとき。

- TIMx_CR1 レジスタの URS=0 かつ UDIS=0 で、トリガ・イベントで CNT が初期化されたとき。

13.4.6 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	Res.	w	w	w	w	w	w

Bits 15:7 予約済み、常に 0 が読出されます。

Bit 6 TG: トリガ・イベント生成

このビットをセットすることで、ソフトウェアでイベントを発生させることができます。このビットはハードウェアで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。関連する許可ビットがセットされていれば、割込みリクエストや DMA リクエストが発生します。

Bit 5 予約済み、常に 0 が読出されます。

Bit 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。

Bit 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。

参考資料

Bit 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。

Bit 1 CC1G: キャプチャ / 比較 1 イベント生成

このビットをセットすることで、ソフトウェアでイベントを発生させることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: チャネル 1 でキャプチャ / 比較イベントが発生します。

CC1 チャネルが出力設定のとき :

C1IF フラグがセットされます。関連する許可ビットがセットされていれば、割込みリクエストや DMA リクエストが発生します。

チャネルが入力設定のとき :

その時点のカウンタの値が TIMx_CCR1 レジスタに保存されます。CC1IF フラグがセットされ、関連する許可ビットがセットされていれば、割込みリクエストや DMA リクエストが発生します。CC1IF フラグがすでにハイレベルであれば、CC1OF フラグがセットされます。

Bit 0 UG: 更新イベント生成

このビットはソフトウェアでセットすることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: カウンタを初期化し、レジスタの更新を行います。プリスケーラカウンタもクリアされます。（プリスケーラの設定値は変化しません。）センター・アライン・モードもしくは DIR=0（カウントアップ）が選択されているときには、カウンタの値はクリアされます。DIR=1（カウントダウン）であれば、自動再ロード・レジスタ (TIMx_ARR) の値がカウンタにロードされます。

参考資料

13.4.7 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

各チャネルは入力（キャプチャ・モード）もしくは出力（比較モード）で使用することができます。チャネルの方向は対応する CC_xS ビットで定まります。このレジスタの他のビットは入力モードと出力モードで異なる意味をもちます。ビット名が OC_{xx} のビットはチャネルが出力設定のときに有効で、IC_{xx} のビットは入力設定のときに有効です。同じビットが入力モードと出力モードで別の役割を果たしている点に、くれぐれも注意してください。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M[2:0]			OC2 PE	OC2 FE	CC2S[1:0]		OC1 CE	OC1M[2:0]			OC1 PE	OC1 FE	CC1S[1:0]	
IC2F[3:0]			IC2PSC[1:0]		IC1F[3:0]			IC1PSC[1:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

Bit 15 OC2CE: 出力比較 2 クリア・イネーブル

Bits 14:12 OC2M[2:0]: 出力比較 2 モード

Bit 11 OC2PE: 出力比較 2 プリロード・イネーブル

Bit 10 OC2FE: 出力比較 2 高速イネーブル

Bits 9:8 CC2S[1:0]: キャプチャ / 比較 2 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC2 チャネルは出力設定です。

01: CC2 チャネルは入力設定で、IC2 には TI2 が選択されます。

10: CC2 チャネルは入力設定で、IC2 には TI1 が選択されます。

11: CC2 チャネルは入力設定で、IC2 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC2S ビットはチャネルが OFF (TIMx_CCER レジスタで CC2E=0) のときにのみ書き込みができます。

Bit 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。

参考資料

Bits 6:4 OC1M: 出力比較 1 モード

これらのビットは OC1 及び OC1N の元になる出力基準信号 OC1REF 挑動を規定します。OC1 及び OC1N のアクティブルレベルが CC1P 及び CC1NP ビットに依存するのに対し、OC1REF はアクティブ・ハイの信号です。

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: 一致時にアクティブルレベル – カウンタ TIMx_CNT がキャプチャ / 比較レジスタ TIMx_CCR1 と一致したとき、OC1REF 信号はハイレベルとされます。

010: 一致時にインアクティブルレベル – カウンタ TIMx_CNT がキャプチャ / 比較レジスタ TIMx_CCR1 と一致したとき、OC1REF 信号はローレベルとされます。

011: 反転 – TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

100: インアクティブルレベル – OC1REF はローレベルとなります。

101: アクティブルレベル – OC1REF はハイレベルとなります。

110: PWM モード 1 – カウントアップ時、TIMx_CNT < TIMx_CCR1 であればチャネル 1 はアクティブに、これ以外はインアクティブになります。カウントダウン時は、TIMx_CNT > TIMx_CCR1 であればインアクティブに、これ以外はアクティブになります。

111: PWM モード 2 – カウントアップ時、TIMx_CNT < TIMx_CCR1 であればチャネル 1 はインアクティブに、これ以外はアクティブになります。カウントダウン時は、TIMx_CNT > TIMx_CCR1 であればアクティブに、これ以外はインアクティブになります。

注 1: LOCK レベル 3 が指定されており (TIMx_BDTR レジスタの LOCK ビット) CC1S=00 (チャネルは出力設定) の場合には、これらのビットは変更できません。

注 2: PWM モード 1 もしくは 2 では、比較結果が変化したとき、もしく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時にのみ、OC1REF のレベルが変化します。

Bit 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・レジスタはディセーブルです。TIMx_CCR1 は常時書き込みが可能で、書き込まれた値は直ちに有効となります。

1: TIMx_CCR1 のプリロード・レジスタはイネーブル状態です。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時にアクティブルレジスタにロードされます。

注 1: LOCK レベル 3 が指定されており (TIMx_BDTR レジスタの LOCK ビット) CC1S=00 (チャネルは出力設定) の場合には、これらのビットは変更できません。

注 2: 単一パルス発生モード (TIMx_CR1 レジスタの OPM=1) のときにのみ、プリロード機能を使用せずに PWM モードを利用することができます。連続的な PWM モードでは、その動作は保証されません。

Bit 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 CC1 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、トリガがオンであっても、通常はカウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 が有効となるまでに、5 クロック・サイクルを要します。

1: トリガ入力のアクティブルエッジは、CC1 出力に対して、比較一致のように働きます。このため、比較結果とは別に、OC は比較レベルにセットされます。トリガ入力をサンプルし CC1 出力を有効とする際の遅延は 3 クロック・サイクルに短縮されます。OCxFE ビットはチャネルが PWM1 もしくは PWM2 モードに設定されているような効果を効果をもちます。

Bits 1:0 CC1S: キャプチャ / 比較 1 選択

これらのビットはチャネルの方向 (入力 / 出力) と入力時の機能を定めます。

00: CC1 チャネルは出力設定です。

01: CC1 チャネルは入力設定で、IC1 には TI1 が選択されます。

10: CC1 チャネルは入力設定で、IC1 には TI2 が選択されます。

11: CC1 チャネルは入力設定で、IC1 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC1S ビットはチャネルが OFF (TIMx_CCER レジスタで CC1E=0) のときにのみ書き込みができます。

参考資料

入力キャプチャ・モード

Bits 15:12 **IC2F**: 入力キャプチャ 2 フィルタ

Bits 11:10 **IC2PSC[1:0]**: 入力キャプチャ 2 プリスケーラ

Bits 9:8 **CC2S[1:0]**: キャプチャ／比較 2 選択

これらのビットはチャネルの方向（入力／出力）と入力時の機能を定めます。

00: CC2 チャネルは出力設定です。

01: CC2 チャネルは入力設定で、IC2 には TI2 が選択されます。

10: CC2 チャネルは入力設定で、IC2 には TI1 が選択されます。

11: CC2 チャネルは入力設定で、IC2 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC2S ビットはチャネルが OFF (TIMx_CCER レジスタで CC2E=0) のときにのみ書き込みができます。

Bits 7:4 **IC1F**: 入力キャプチャ 1 フィルタ

これらのビットは TI1 入力をサンプルする周波数と、TI1 に対するデジタル・フィルタの応答を定めます。デジタル・フィルタは、出力に有効な変化をもたらすために N イベントを要するイベントカウンタで構成されています。

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: $f_{SAMPLING} = f_{DTS}/32$ 、 $N=8$

注: 現在のシリコン・リビジョンでは、ICxF[3:0]=1、2 または 3 の場合、fDTS は式内で CK_INT により置き換えられます。

Bits 3:2 **IC1PSC**: 入力キャプチャ 1 プリスケーラ

これらのビットは CC1 入力 (IC1) で働くプリスケーラの分周比を指定します。

プリスケーラは TIMx_CCER レジスタの CC1E=0 でリセットされます。

00: プリスケーラなし、キャプチャ入力でエッジが検出されるたびにキャプチャが行われます。

01: イベント 2 回ごとにキャプチャが行われます。

10: イベント 4 回ごとにキャプチャが行われます。

11: イベント 8 回ごとにキャプチャが行われます。

参考資料

RM0008

汎用タイマ (TIMx)

Bits 1:0 CC1S: キャプチャ / 比較 1 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC1 チャネルは出力設定です。

01: CC1 チャネルは入力設定で、IC1 には TI1 が選択されます。

10: CC1 チャネルは入力設定で、IC1 には TI2 が選択されます。

11: CC1 チャネルは入力設定で、IC1 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC1S ビットはチャネルが OFF (TIMx_CCER レジスタで CC1E=0) のときにのみ書き込みができます。

13.4.8 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

CCMR1 レジスタの詳細については上記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
OC4 CE	OC4M[2:0]			OC4 PE	OC4 FE	CC4S[1:0]		OC3 CE.	OC3M[2:0]			OC3 PE	OC3 FE	CC3S[1:0]	
IC4F[3:0]			IC4PSC[1:0]		IC3F[3:0]			IC3PSC[1:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

Bit 15 OC4CE: 出力比較 4 クリア・イネーブル

Bits 14:12 OC4M: 出力比較 4 モード

Bit 11 OC4PE: 出力比較 4 プリロード・イネーブル

Bit 10 OC4FE: 出力比較 4 高速イネーブル

Bits 9:8 CC4S: キャプチャ / 比較 4 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC4 チャネルは出力設定です。

01: CC4 チャネルは入力設定で、IC4 には TI4 が選択されます。

10: CC4 チャネルは入力設定で、IC4 には TI3 が選択されます。

11: CC4 チャネルは入力設定で、IC4 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC4S ビットはチャネルが OFF (TIMx_CCER レジスタで CC4E=0) のときにのみ書き込みができます。

Bit 7 OC3CE: 出力比較 3 クリア・イネーブル

Bits 6:4 OC3M: 出力比較 3 モード

Bit 3 OC3PE: 出力比較 3 プリロード・イネーブル

Bit 2 OC3FE: 出力比較 3 高速イネーブル

Bits 1:0 CC3S: キャプチャ / 比較 3 選択

これらのビットはチャネルの方向（入力 / 出力）と入力時の機能を定めます。

00: CC3 チャネルは出力設定です。

01: CC3 チャネルは入力設定で、IC3 には TI3 が選択されます。

10: CC3 チャネルは入力設定で、IC3 には TI4 が選択されます。

11: CC3 チャネルは入力設定で、IC3 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC3S ビットはチャネルが OFF (TIMx_CCER レジスタで CC3E=0) のときにのみ書き込みができます。



参考資料

汎用タイマ (TIMx)

RM0008

入力キャプチャ・モード

Bits 15:12 **IC4F:** 入力キャプチャ 4 フィルタ

Bits 11:10 **IC4PSC:** 入力キャプチャ 4 プリスケーラ

Bits 9:8 **CC4S:** キャプチャ／比較 4 選択

これらのビットはチャネルの方向（入力／出力）と入力時の機能を定めます。

00: CC4 チャネルは出力設定です。

01: CC4 チャネルは入力設定で、IC4 には TI4 が選択されます。

10: CC4 チャネルは入力設定で、IC4 には TI3 が選択されます。

11: CC4 チャネルは入力設定で、IC4 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC4S ビットはチャネルが OFF (TIMx_CCER レジスタで CC4E=0) のときにのみ書き込みができます。

Bits 7:4 **IC3F:** 入力キャプチャ 3 フィルタ

Bits 3:2 **IC3PSC:** 入力キャプチャ 3 プリスケーラ

Bits 1:0 **CC3S:** キャプチャ／比較 3 選択

これらのビットはチャネルの方向（入力／出力）と入力時の機能を定めます。

00: CC3 チャネルは出力設定です。

01: CC3 チャネルは入力設定で、IC3 には TI3 が選択されます。

10: CC3 チャネルは入力設定で、IC3 には TI4 が選択されます。

11: CC3 チャネルは入力設定で、IC3 には TRGI が選択されます。これは 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時にのみ正しく働きます。

注: CC3S ビットはチャネルが OFF (TIMx_CCER レジスタで CC3E=0) のときにのみ書き込みができます。

13.4.9 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	CC4P	CC4E	予約済み	CC3P	CC3E	予約済み	CC2P	CC2E	予約済み	CC1P	CC1E				
	rw	rw													

Bits 15:14 予約済み、常に 0 が読出されます。

Bit 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。

Bit 12 **CC4E:** キャプチャ／比較 4 出力イネーブル

CC1E の説明を参照してください。

Bits 11:10 予約済み、常に 0 が読出されます。

Bit 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。

Bit 8 **CC3E:** キャプチャ／比較 3 出力イネーブル

CC1E の説明を参照してください。

Bits 7:6 予約済み、常に 0 が読出されます。

Bit 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。

参考資料

RM0008

汎用タイマ (TIMx)

Bit 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。

Bits 3:2 予約済み、常に 0 が読出されます。

Bit 1 CC1P: キャプチャ / 比較 1 出力極性

CC1 チャネルが出力設定のとき :

0: OC1 アクティブ・ハイ

1: OC1 アクティブ・ロー

CC1 チャネルが入力設定のとき :

このビットはトリガやキャプチャ動作に IC1 か $\overline{IC1}$ のいずれが使用されるかを選択します。

0: 非反転 : キャプチャは IC1 の立上りエッジで行われます。外部トリガとして使用される時には、IC1 は反転されていません。

1: 反転 : キャプチャは IC1 の立下りエッジで行われます。外部トリガとして使用される時には、IC1 は反転されます。

Bit 0 CC1E: キャプチャ / 比較 1 出力イネーブル

CC1 チャネルが出力設定のとき :

0: オフ - OC1 はアクティブではありません。

1: オン - OC1 は所定のピンに出力されます。

CC1 チャネルが入力設定のとき :

このビットはカウンタ値のキャプチャ / 比較レジスタ 1 (TIMx_CCR1) へのキャプチャ動作が行われるか否かを指定します。

0: キャプチャディセーブル

1: キャプチャイネーブル

表 58. 標準 OCx チャネルの出力制御ビット

CCxE ビット	OCx 出力状態
0	出力ディセーブル (OCx=0、OCx_EN=0)
1	OCx=OCxREF + 極性、OCx_EN=1

注意 : 相補出力 OCx チャネル及び OCxN チャネルに接続されている外部入出力ピンの状態は、両チャネルの状態と、GPIO/AFIO レジスタの設定に依存します。

13.4.10 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

Bits 15:0 CNT[15:0]: カウンタの値



参考資料

13.4.11 プリスケーラ (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

Bits 15:0 **PSC[15:0]: プリスケーラの値**

カウンタ・クロック (CK_CNT) の周波数は $f_{CK_PSC} / (PSC[15:0] + 1)$ となります。

PSC は更新イベントごとにアクティブ・プリスケーラ・レジスタにロードされる値を格納しています。

13.4.12 自動再ロード・レジスタ (TIMx_ARR)

アドレスオフセット : 0x2C

リセット値 : 0x0000

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

Bits 15:0 **ARR[15:0]: 自動再ロード・レジスタの値**

ARR は実際の（シャドウ）自動再ロード・レジスタにロードされる値です。

ARR の更新と挙動の詳細は [セクション 13.3.1: タイムベース・ユニット \(286 ページ\)](#) を参照してください。

自動再ロード値が 0 のときには、カウンタへのロードはブロックされています。

13.4.13 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

Bits 15:0 **CCR1[15:0]: キャプチャ / 比較 1 の値**

CC1 チャネルが出力設定のとき :

CCR1 はアクティブキャプチャ / 比較レジスタ 1 にロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの CC1PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード値がアクティブキャプチャ / 比較レジスタ 1 にロードされます。

アクティブキャプチャ / 比較レジスタは OC1 出力を送出するためにカウンタ TIMx_CNT と比較される値を保持しています。

CC1 チャネルが入力設定のとき :

CCR1 は最近の入力キャプチャ 1 イベント (IC1) でラッチされたカウンタの値です。

参考資料

13.4.14 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

Bits 15:0 CCR2[15:0]: キャプチャ / 比較 2 の値

CC2 チャネルが出力設定のとき :

CCR2 はアクティブキャプチャ / 比較レジスタ 2 にロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの CC2PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード値がアクティブキャプチャ / 比較レジスタ 2 にロードされます。

アクティブキャプチャ / 比較レジスタは OC2 出力を送出するためにカウンタ TIMx_CNT と比較される値を保持しています。

CC2 チャネルが入力設定のとき :

CCR2 は最近の入力キャプチャ 2 イベント (IC2) でラッチされたカウンタの値です。

13.4.15 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

Bits 15:0 CCR3[15:0]: キャプチャ / 比較 3 の値

CC3 チャネルが出力設定のとき :

CCR3 はアクティブキャプチャ / 比較レジスタ 3 にロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの CC3PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード値がアクティブキャプチャ / 比較レジスタ 3 にロードされます。

アクティブキャプチャ / 比較レジスタは OC3 出力を送出するためにカウンタ TIMx_CNT と比較される値を保持しています。

CC3 チャネルが入力設定のとき :

CCR3 は最近の入力キャプチャ 3 イベント (IC3) でラッチされたカウンタの値です。

参考資料

汎用タイマ (TIMx)

RM0008

13.4.16 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

Bits 15:0 CCR4[15:0]: キャプチャ / 比較 4 の値

1/CC4 チャネルが出力設定 (CC4S ビット) のとき :

CCR4 はアクティブキャプチャ / 比較レジスタ 4 にロードされる値 (プリロード 値) です。

TIMx_CCMR4 レジスタの CC4PE ビットでプリロード機能が有効とされていないときには、この値は直ちにロードされます。プリロード機能が有効なときには、更新イベントが発生した時にプリロード 値がアクティブキャプチャ / 比較レジスタ 4 にロードされます。

アクティブキャプチャ / 比較レジスタは OC4 出力を送出するためにカウンタ TIMx_CNT と比較される値を保持しています。

2/CC4 チャネルが入力設定 (TIMx_CCMR4 レジスタの CC4S ビット) のとき :

CCR4 は最近の入力キャプチャ 4 イベント (IC4) でラッチされたカウンタの値です。

13.4.17 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]					rw	rw	rw	rw
	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw				

Bits 15:13 予約済み、常に 0 が読出されます。

Bits 12:8 DBL[4:0]: DMA バースト長

これらのビットはバースト・モード (TIMx_DMAR のアドレスに対して行われる読書きのアクセスはバースト転送として扱われます) で DMA 転送の際のデータ長を指示します。転送されるバイト数は、次のように指定されます。

00000: 1byte

00001: 2bytes

00010: 3bytes

...

10001: 18bytes

Bits 7:5 予約済み、常に 0 が読出されます。

参考資料

Bits 4:0 DBA[4:0]: DMA ベース・アドレス

これらのビットはバースト・モード（読み書きのアクセスは TIMx_DMAR のアドレスを通して行われます）で DMA 転送のためのベース・アドレスを指示します。DBA は TIMx_CR1 レジスタのアドレスから始まるオフセット値として格納されます。

例：

00000: TIMx_CR1

00001: TIMx_CR2

00010: TIMx_SMCR

..

13.4.18 バースト・モード DMA アドレス・レジスタ (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

Bits 15:0 DMAB[15:0]: DMA バーストアクセスレジスタ

DMAR レジスタへの読み書きのアクセスは、次のアドレスにあるレジスタへのアクセスとなります。

(TIMx_CR1 アドレス) + DBA + (DMA オフセット)、ここで

TIMx_CR1 アドレスは、制御レジスタ 1 のアドレスです。

DBA は、TIMx_DCR レジスタに設定されている DMA ベース・アドレスです。

DMA オフセットは、DMA 転送の際に自動的に制御されるオフセット値で、TIMx_DCR レジスタの転送長 DBL に依存した値となります。

参考資料

汎用タイマ (TIMx)

RM0008

13.4.19 TIMx レジスタ・マップ

TIMx レジスタは、次の表のように、すべて 16bit のレジスタです。

表 59. TIMx レジスタ・マップとリセット時の値

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 リセット値	予約済み												CKD [1:0] 0 0	ARPE 0	CMS [1:0] 0 0	DIR 0	OPM 0	CCDS 0	URS 0	UDIS 0	CEN 0																				
0x04	TIMx_CR2 リセット値	予約済み												T1S 0	MMS[2:0] 0 0 0	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	
0x08	TIMx_SMCR リセット値	予約済み						ETP 0	ECE 0	ETPS [1:0] 0 0	ETP[3:0] 0 0 0 0	MSW 0	TS[2:0] 0 0 0	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み	预約済み							
0x0C	TIMx_DIER リセット値	予約済み												TDE 0	COMDE 0	CC4DE 0	CC3DE 0	CC2DE 0	CC1DE 0	UDE 0	TIE 0	COMIE 0	CC4IE 0	CC3IE 0	CC2IE 0	CC1IE 0	UJF 0	UG 0														
0x10	TIMx_SR リセット値	予約済み												OC4OF 0	OC3OF 0	OC2OF 0	OC1OF 0	OC1OF 0	OC1OF 0	OC1OF 0	TIF 0	COMIF 0	CC4IF 0	CC3IF 0	CC2IF 0	CC1IF 0	UIF 0	UG 0														
0x14	TIMx_EGR リセット値	予約済み												IC2F[3:0] 0 0 0 0	IC2PSC [1:0] 0 0	CC2S [1:0] 0 0	CC1S [1:0] 0 0	IC1F[3:0] 0 0 0 0	IC1PSC [1:0] 0 0	CC1S [1:0] 0 0																						
0x18	TIMx_CCMR1 Output Compare mode リセット値	予約済み						OC2CE 0	OC2M [2:0] 0 0 0	OC2PE 0	OC2FE 0	CC2S [1:0] 0 0	CC1CE 0	OC1M [2:0] 0 0 0	OC1PE 0	OC1FE 0	CC1S [1:0] 0 0																									
	TIMx_CCMR1 Input Capture mode リセット値	予約済み						IC2F[3:0] 0 0 0 0	IC2PSC [1:0] 0 0	CC2S [1:0] 0 0	CC1F[3:0] 0 0 0 0	IC1PSC [1:0] 0 0	CC1S [1:0] 0 0																													
0x1C	TIMx_CCMR2 Output Compare mode リセット値	予約済み						O24CE 0	OC4M [2:0] 0 0 0	OC4PE 0	OC4FE 0	CC4S [1:0] 0 0	CC3CE 0	OC3M [2:0] 0 0 0	OC3PE 0	OC3FE 0	CC3S [1:0] 0 0																									
	TIMx_CCMR2 Input Capture mode リセット値	予約済み						IC4F[3:0] 0 0 0 0	IC4PSC [1:0] 0 0	CC4S [1:0] 0 0	IC3F[3:0] 0 0 0 0	IC3PSC [1:0] 0 0	CC3S [1:0] 0 0																													
0x20	TIMx_CCER リセット値	予約済み						CC4P 0	CC4E 0	CC4E 0	CC4E 0	预約済み 0	CC3E 0	预約済み 0	CC2E 0	预約済み 0	CC1E 0																									
0x24	TIMx_CNT リセット値	予約済み						CNT[15:0]																																		
0x28	TIMx_PSC リセット値	予約済み						PSC[15:0]																																		
0x2C	TIMx_ARR リセット値	予約済み						ARR[15:0]																																		
0x30		予約済み																																								

參考資料

表 59. TIMx レジスタ・マップとリセット時の値 (続き)

レジスタのアドレス範囲に関しては表1(36ページ)を参照してください。

14 ベーシック・タイマ (TIM6 及び TIM7)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、大容量デバイスのみに適応されます。

14.1 概要

ベーシック・タイマ TIM6 及び TIM7 は、プログラマブルなプリスケーラで駆動される 16bit 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する事が出来ますが、これらのタイマは、デジタルーアナログ変換 (DAC) 専用としても使われます。実際、これらのタイマは内部で DAC と接続されており、トリガ出力を通じて使う事が可能です。

タイマは完全に独立しており、いかなるリソースも共有しません。

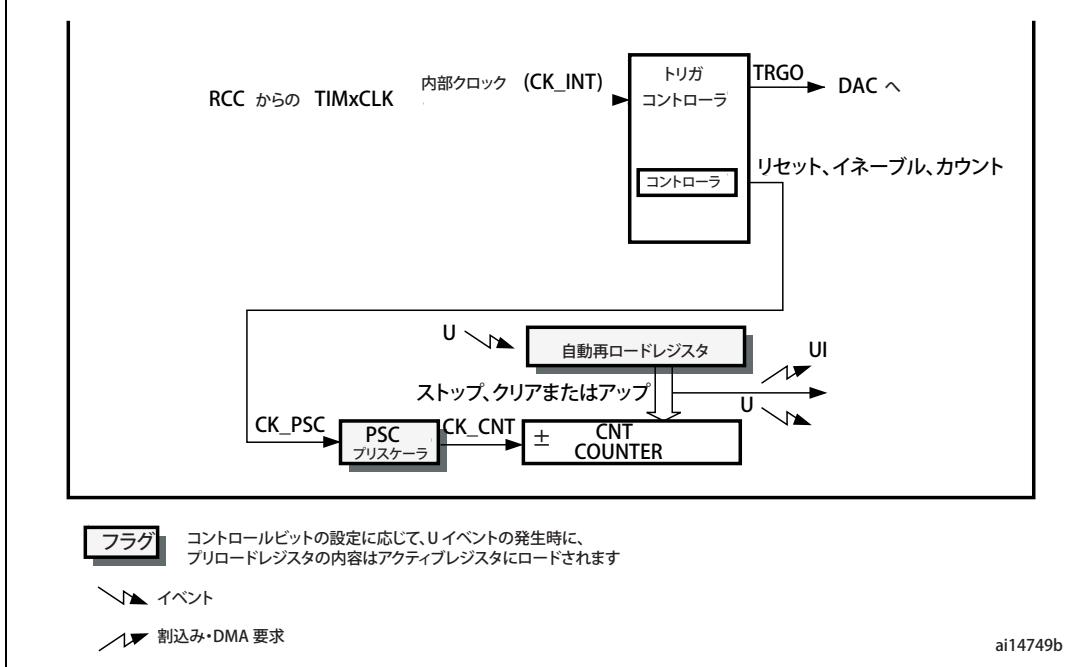
14.2 TIM6 及び TIM7 の主な機能

ベーシック・タイマ (TIM6 及び TIM7) は、次のような機能を搭載しています。

- 16bit 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 65535 の間で分周するための 16bit プログラマブル・プリスケーラ
- DAC をトリガする同期回路
- アップデート・イベントにおける割込み /DMA 生成：カウンタ・オーバフロー

参考資料

図 141. 基本タイマ・ブロック図



14.3 TIM6 及び TIM7 の機能詳細

14.3.1 タイムベース・ユニット

プログラム可能なタイマの主要なブロックは自動再ロード・レジスタを持った 16 bit カウンタです。カウンタはカウントアップ、カウントダウン、アップ・ダウンの動作が可能です。カウンタのクロックはプレスケーラによって分周されます。

カウンタ、自動再ロード・レジスタ、プリスケーラ・レジスタは、カウンタが動作中であっても、ソフトウェアによる読み書きができます。

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)
- プリスケーラ・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機能を搭載しています。自動再ロード・レジスタの読み書きはプリロード・レジスタへのアクセスとなります。プリロード・レジスタの内容が、TIMx_CR1 レジスタのプリロード・イネーブル・ビット (ARPE) の設定に応じて、常に更新イベント 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 であれば、更新イベントはカウンタがオーバフロー（ダウンカウント時はアンダフロー）に達したときに発生します。また、ソフトウェアで更新イベントを発生させることも可能です。この方法は、それぞれのタイマ構成のセクションに記されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタ・イネーブル・ビット (CEN) がセットされているときにのみ、プリスケーラ出力 CK_CNT から供給されます。（カウンタをイネーブルとする方法の詳細に関しては、スレーブ・モード・コントローラの記述を合わせてお読みください。）

参考資料

なお、実際のカウンタ・イネーブル信号 CNT_EN は CEN ビットがセットされてから 1 クロック遅れてセットされることに留意してください。

プリスケーラ

プリスケーラはカウンタ・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。プリスケーラは 16bit レジスタ (TIMx_PSC) を通して制御される 16 bit カウンタをベースとしています。制御レジスタにはバッファが備わっていますので、プリスケーラが動作中であっても、レジスタ内容を変更することができます。新たに指定された分周比は次の更新イベントで効果をもちます。

図 142 と図 143 は、動作中に分周比を変化させたときのカウンタの動作の例を示しています。

図 142. プリスケーラ分周比が 1 から 2 に変化した時のカウンタ・タイミング図

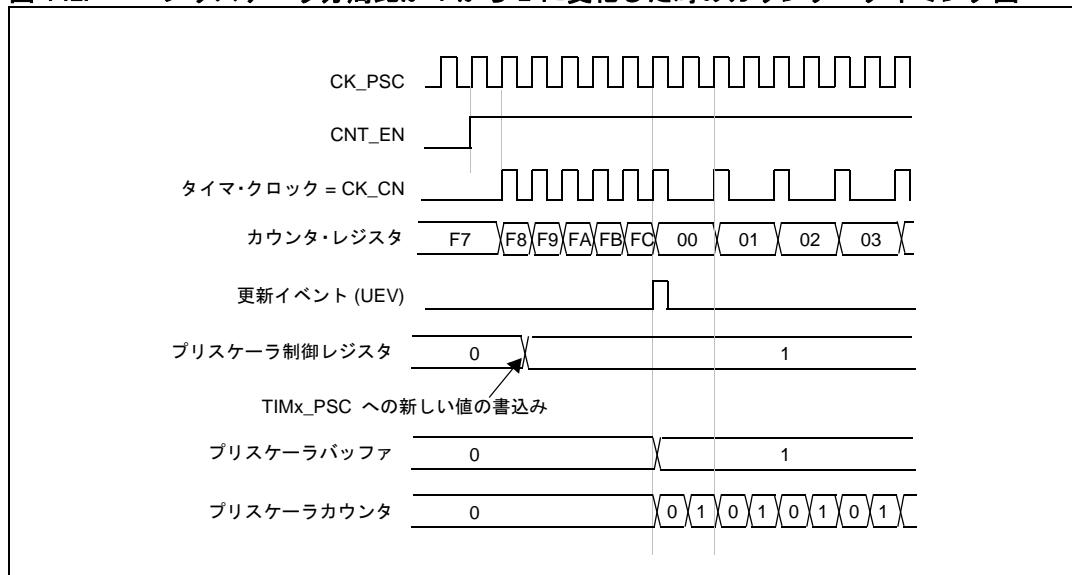
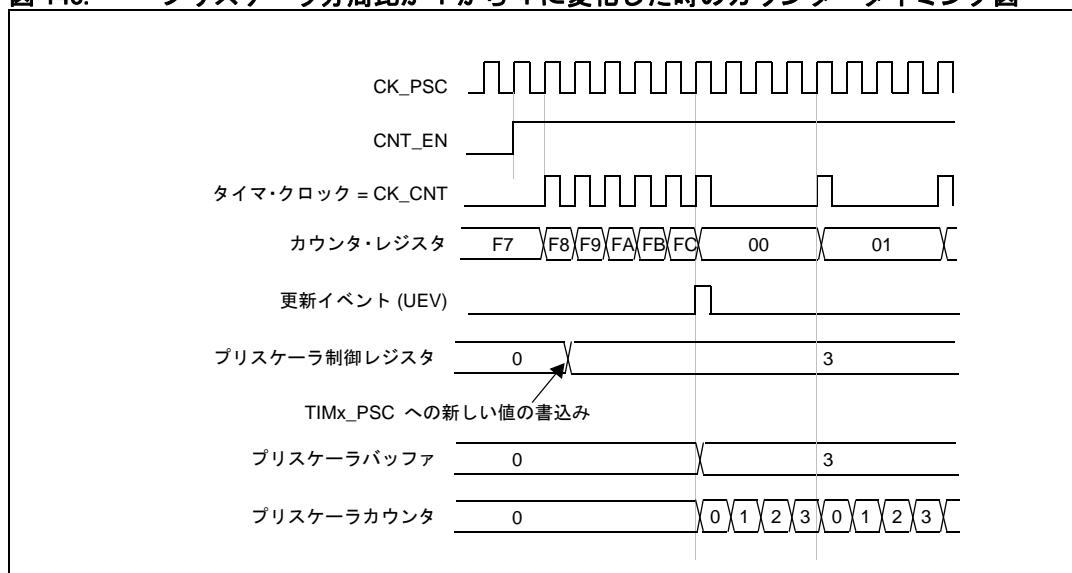


図 143. プリスケーラ分周比が 1 から 4 に変化した時のカウンタ・タイミング図



14.3.2 カウンタ動作モード

カウンタは、0 から自動再ロード・レジスタの値 (TIMx_ARR レジスタの内容) までカウントを行い、その後 0 からのカウントを再開し、同時にカウンタ・オーバフロー・イベントを発生させます。

カウンタのオーバフローごとに更新イベントが発生します。ソフトウェアによるレジスタ書き込みで、もしくはスレーブ・モード・コントローラによって TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントを発生させることができます。

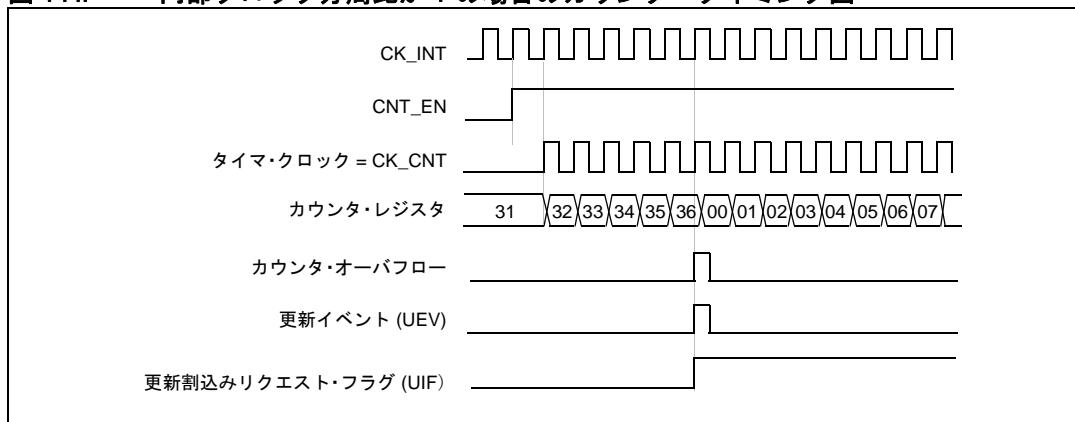
UEV イベントはソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることで、ディセーブルすることができます。これはプリロード・レジスタに新しい値を書込んでいるときにシャドウ・レジスタが更新されるのを防止します。この後 UDIS ビットに 0 が書込まれるまで、更新イベントは発生しません。しかし、プリスケーラのカウンタと同じく（ただしプリスケーラの比は変化しません）、カウンタは 0 からカウントを再開します。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットのセットが更新イベント UEV を引起こしますが、UIF フラグはセットされず、割込みや DMA リクエストは発生しません。これは、キャプチャイベントの際にカウンタをクリアすることで、更新とキャプチャ割込みがともに発生することを防ぎます。

更新イベントが発生したとき、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。

- プリスケーラのバッファにはプリロード値 (TIMx_PSC レジスタの値) が再びロードされます。
- 自動再ロード・シャドウ・レジスタはプリロード・レジスタ (TIMx_ARR) の値に更新されます。

以下の図は、自動再ロード・レジスタ TIMx_ARR = 0x36 の場合の、異なるクロック周波数におけるカウンタの挙動の例を示しています。

図 144. 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



参考資料

図 145. 内部クロック分周比が 2 の場合のカウンタ・タイミング図

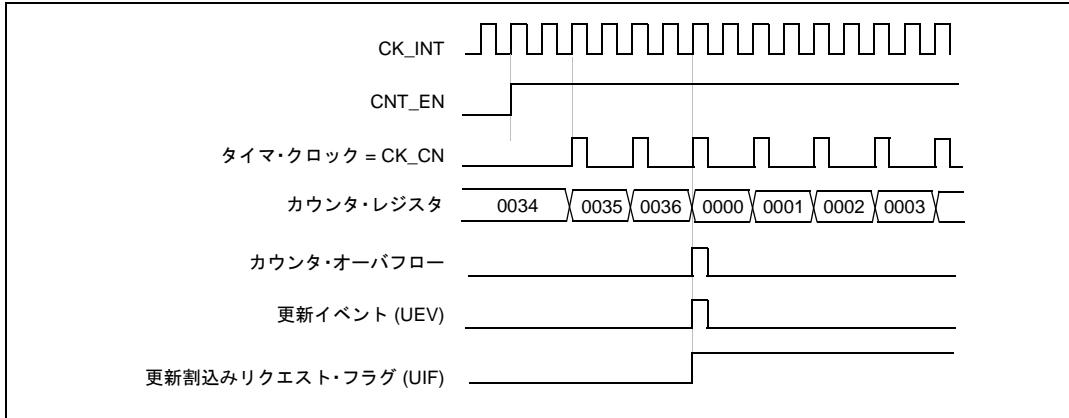


図 146. 内部クロック分周比が 4 の場合のカウンタ・タイミング図

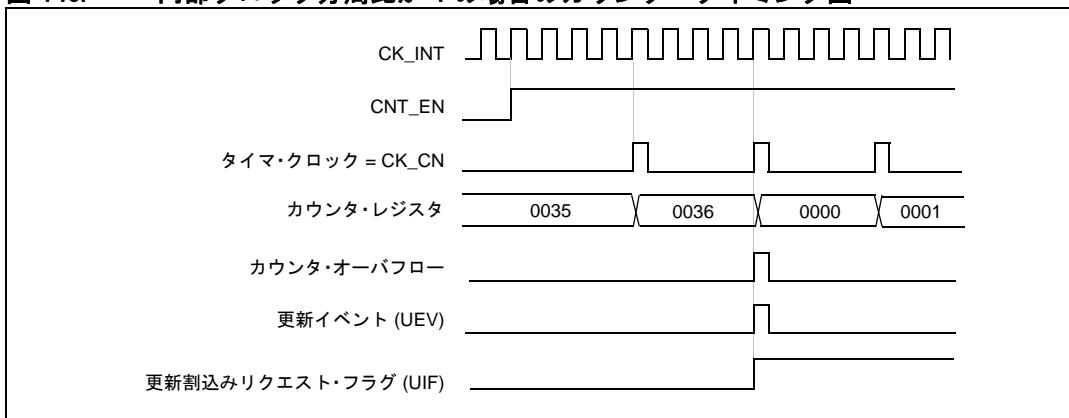
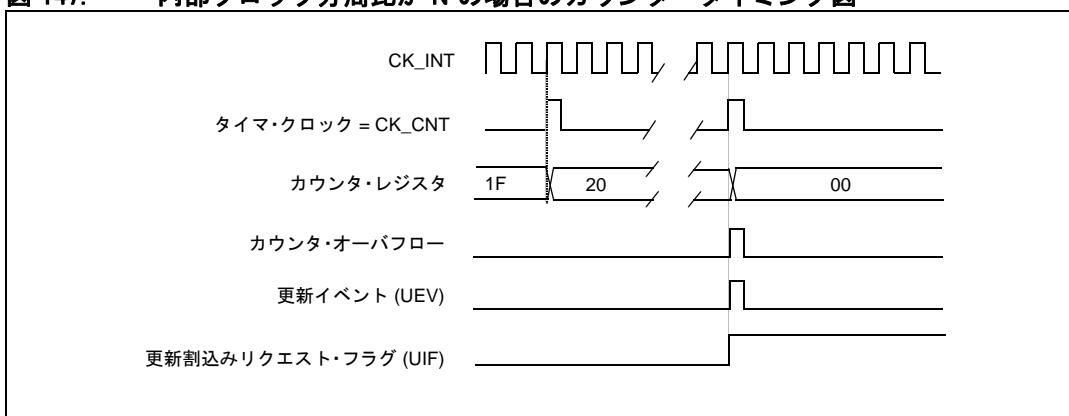


図 147. 内部クロック分周比が N の場合のカウンタ・タイミング図



参考資料

図 148. ARPE=0 時 (TIMx_ARR はプリロードなし)、更新イベント時のタイミング図

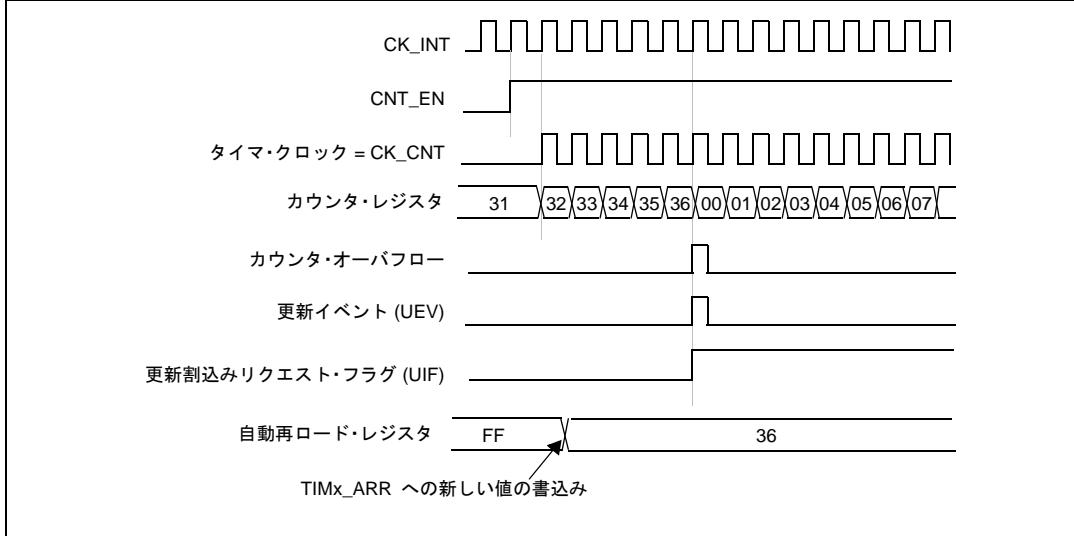
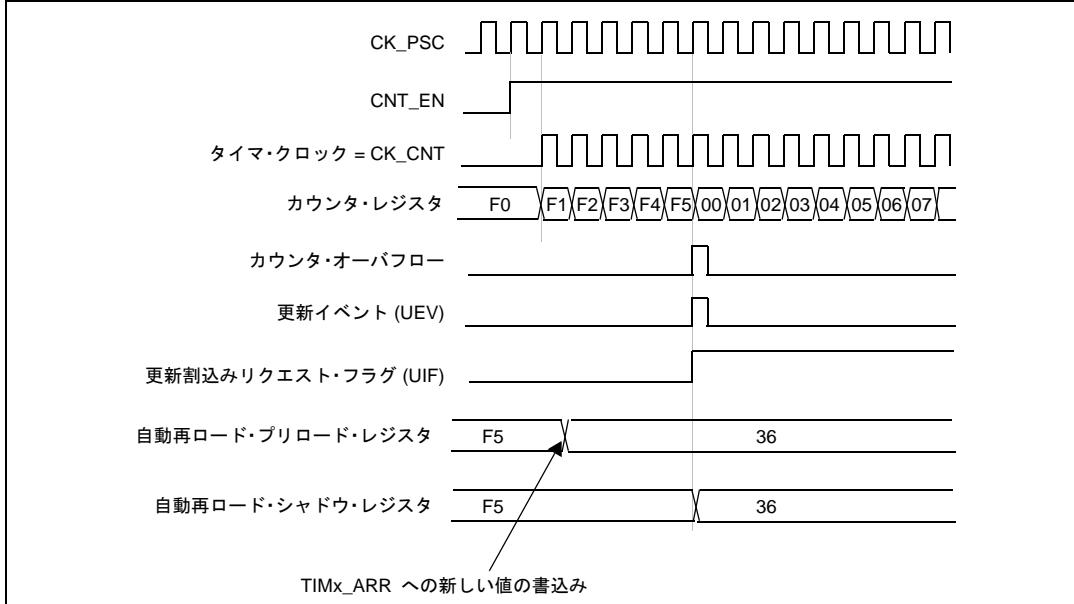


図 149. ARPE=1 時 (TIMx_ARR はプリロード)、更新イベント時のタイミング図



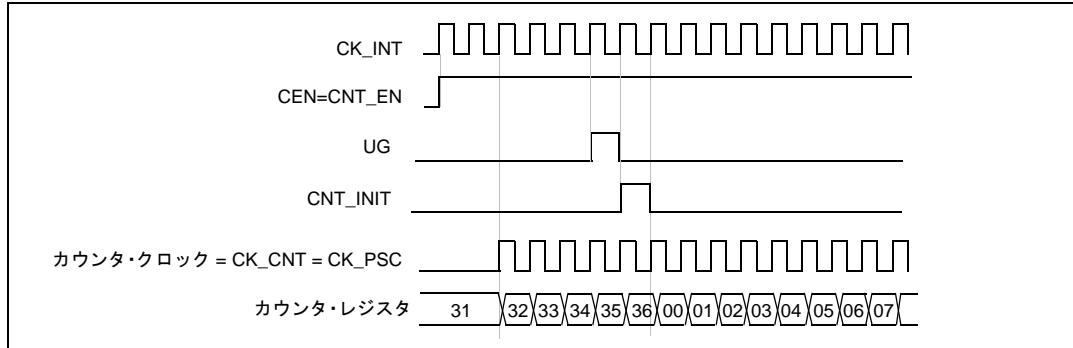
14.3.3 クロック・ソース

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、DIR ビットと TIMx_EGR レジスタの UG ビットが実際の制御ビットとなります。自動的にクリアされる UG ビット以外は、これらのビットはソフトウェアでのみ変更できます。CEN ビットに 1 が書込まれると、プリスケー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 150 は、プリスケー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が図示されています。

図 150. 内部クロック分周比 1 の場合の、通常モードの制御回路



14.3.4 デバッグ・モード

マイクロコントローラがデバッグ・モードに入った時 (Cortex-M3 コア停止)、DBG モジュールの DBG_TIMx_STOPTIMx 構成ビットに応じて、カウンタは通常通り動作を続けるか、または停止します。詳細については、[セクション 26.15.2: タイマ、ウォッチドッグ、bxCAN と PC に対するデバッグ・サポート](#)を参照してください。

14.4 TIM6 及び TIM7 レジスタ

レジスタの詳細記述で使用されている略語については、[セクション 1.1 \(32 ページ\)](#) を参照してください。

14.4.1 制御レジスタ 1 (TIMx_CR1)

アドレスオフセット : 00h

リセット値 : 0000h

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			
Res.		rw		Res.		rw		rw		rw		rw			

Bits 15:8 予約済み、常に 0 が読出されます。

Bit 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を使用しません。

1: TIMx_ARR レジスタはバッファを使用します。

Bits 6:4 予約済み、常に 0 が読出されます。

Bit 3 **OPM: 単一パルス発生モード**

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動作を停止 (CEN ビットをクリア) します。

参考資料

Bit 2 URS: 更新リクエストの要因

更新イベント (UEV) を発生する元となる信号を選択するため、ソフトウェアでセット / クリアができます。

0: 次のイベントのいずれかが更新割込みリクエストもしくは DMA リクエストを発生します。(リクエストが許可されているときのみ)

-カウンタ・オーバフロー / アンダフロー

-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指示

-1: カウンタのオーバフロー / アンダフローのみが更新割込みリクエストもしくは DMA リクエストを発生します。(リクエストが許可されているときのみ)

Bit 1 UDIS: 更新動作ディセーブル

更新イベント (UEV) 発生をイネーブル / ディセーブルとするため、ソフトウェアでセット / クリアができます。

0: UEV イネーブル。更新イベント (UEV) は次のいずれかで発生します。

-カウンタ・オーバフロー / アンダフロー

-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指示

このとき、バッファを持つレジスタではプリロード値がロードされます。

1: UEV ディセーブル。更新イベントは発生せず、個々のシャドウ・レジスタ (ARR, PSC, CCRx) がそれぞれの値を保持しています。カウンタとプリスケーラは、UG ビットがセットされたとき、及びスレーブ・モード・コントローラからハードウェアリセットが指示されたときに初期化されます。

Bit 0 CEN: カウンタ・イネーブル

0: カウンタ・ディセーブル

1: カウンタ・イネーブル

注: 外部クロック、ゲート・モード及びエンコーダ・モードは CEN ビットが事前にソフトウェアでセットされているときのみ動作します。トリガ・モードではハードウェアで自動的に CEN ビットをセットします。

単一パルス発生モードでは、更新イベント発生時に CEN は自動的にクリアされます。

参考資料

14.4.2 制御レジスタ 2 (TIMx_CR2)

アドレスオフセット : 04h

リセット値 : 0000h

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			
Res.								rw	rw	rw	rw	Res.			

Bits 15:7 予約済み、常に 0 が読出されます。

Bits 6:4 MMS: 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・モードから送られる情報 (TRGO) の選択を行います。

000: **Reset** – TIMx_EGR レジスタの UG ビットがトリガ出力 (TRGO) として使用されます。トリガ入力によってリセットが発生したとき (スレーブ・モード・コントローラがリセット・モードに設定されているとき) TRGO 信号は実際のリセットから遅れて発生します。

001: **Enable** – 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。これは幾つかのタイマを同時にスタートさせるときや、スレーブタイマの動作範囲を制御する時に有効です。カウンタ・イネーブル信号は、CEN 制御ビットとゲート・モードに設定されているときのトリガ入力との論理和で発生します。

カウンタ・イネーブル信号がトリガ入力で制御されているとき、マスタ / スレーブ・モードに設定されているときを除いて TRGO 出力には遅延が存在します。(TIMx_SMCR レジスタの MSM ビットの説明をお読みください。)

010: **Update** – 更新イベントがトリガ出力 (TRGO) として選択されます。例えばマスタタイマがスレーブタイマのプリスケーラとなるときに使用されます。

Bits 3:0 予約済み、常に 0 が読出されます。

14.4.3 DMA / 割込みリクエスト許可レジスタ (TIMx_DIER)

アドレスオフセット : 0Ch

リセット値 : 0000h

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		
Res.								rw	Res.				rw		

Bit 15:9 予約済み、常に 0 が読出されます。

Bit 8 UDE: 更新 DMA リクエスト許可

- 0: 更新 DMA リクエスト禁止
- 1: 更新 DMA リクエスト許可

Bit 7:1 予約済み、常に 0 が読出されます。

Bit 0 UIE: 更新割込みリクエスト許可

- 0: 更新割込みリクエスト禁止
- 1: 更新割込みリクエスト許可

参考資料

RM0008

ベーシック・タイマ(TIM6 及び TIM7)

14.4.4 ステータス・レジスタ (TIMx_SR)

アドレスオフセット : 10h

リセット値 : 0000h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								UIF							
Res.								rc_w0							

Bits 15:1 予約済み、常に 0 が読出されます。

Bit 0 **UIF: 更新割込みリクエスト・フラグ**

このビットは更新イベント発生時にハードウェアでセットされます。フラグのクリアはソフトウェアで行います。

0: 更新は発生していません。

1: 更新割込みリクエストがペンディング状態です。このビットはレジスタが更新されたときにハードウェアでセットされます。

-TIMx_CR1 レジスタの UDIS=0 で、繰返しダウンカウンタに関してオーバフローもしくはアンダーフローが検出されたとき。(REP_CNT=0 で更新イベントが発生します。)

-TIMx_CR1 レジスタの URS=0 かつ UDIS=0 で、TIMx_EGR レジスタの UG ビットを使用し、ソフトウェアで CNT の初期化が行われたとき。

14.4.5 イベント生成レジスタ (TIMx_EGR)

アドレスオフセット : 14h

リセット値 : 0000h

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								予約済み .							
Res.								Res.							

Bits 15:1 予約済み、常に 0 が読出されます。

Bit 0 **UG: 更新イベント生成**

このビットはソフトウェアでセットすることができます。このビットはハードウェアで自動的にクリアされます。

0: 影響なし。

1: カウンタを初期化し、レジスタの更新を行います。プリスケーラカウンタもクリアされます。
(プリスケーラの設定値は変化しません。)



参考資料

14.4.6 カウンタ (TIMx_CNT)

アドレスオフセット : 24h

リセット値 : 0000h

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

Bits 15:0 CNT[15:0]: カウンタの値

14.4.7 プリスケーラ (TIMx_PSC)

アドレスオフセット : 28h

リセット値 : 0000h

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

Bits 15:0 PSC[15:0]: プリスケーラの値

カウンタ・クロック (CK_CNT) の周波数は $f_{CK_PSC} / (PSC[15:0] + 1)$ となります。

PSC は更新イベントごとにアクティブ・プリスケーラ・レジスタにロードされる値を格納しています。

14.4.8 自動再ロード・レジスタ (TIMx_ARR)

アドレスオフセット : 2Ch

リセット値 : 0000h

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

Bits 15:0 ARR[15:0]: 自動再ロード・レジスタの値

ARR は実際の（シャドウ）自動再ロード・レジスタにロードされる値です。

ARR の更新と挙動の詳細は、[セクション 14.3.1: タイムベース・ユニット \(343 ページ\)](#) を参照してください。

自動再ロード値が 0 のときには、カウンタへのロードはブロックされています。

参考資料

14.4.9 TIM6 及び TIM7 レジスタ・マップ

TIMx レジスタは、次の表のよう、すべて 16bit のレジスタです。

表 60. TIM6 及び TIM7 - レジスタ・マップとリセット時の値

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 リセット値	予約済み																								ARPE 0	予約済み 0	OPM 0	URS 0	UDIS 0	CEN 0		
0x04	TIMx_CR2 リセット値	予約済み																							MMS[2:0] 0 0 0	予約済み 0							
0x08		予約済み																															
0x0C	TIMx_DIER リセット値	予約済み																							UDE 0	予約済み 0			UIE 0				
0x10	TIMx_SR リセット値	予約済み																											UIF 0				
0x14	TIMx_EGR リセット値	予約済み																											UG 0				
0x18		予約済み																															
0x1C		予約済み																															
0x20		予約済み																															
0x24	TIMx_CNT リセット値	予約済み																							CNT[15:0]								
0x28	TIMx_PSC リセット値	予約済み																							PSC[15:0]								
0x2C	TIMx_ARR リセット値	予約済み																							ARR[15:0]								

レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。

15 リアルタイム・クロック (RTC)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

15.1 RTC の概要

リアルタイム・クロック (RTC) は他とは独立したタイマです。RTC は連続的に動作しているカウンタで構成されており、何らかのソフトウェアを利用することで、カレンダ・クロックとしての利用を目的としています。カウンタにはシステムの日付や時刻を書込むことができます。

RTC コアとクロック構成 (RCC_BDCR レジスタ) はバックアップ・ドメインにあります。つまり、RTC の設定と時間はリセットや STANDBY モードからの起動後にも保存されることを意味します。

リセット後のバックアップ・レジスタへのアクセスは、ディセーブルされておりバックアップ・ドメインは不正な書き込みアクセスから保護されます。バックアップ・レジスタと RTC へのアクセスをイネーブルするには次の手順に従ってください。

- RCC_APB1ENR レジスタの PWREN と BKREN ビットをセットし、電源及びバックアップ・インタフェース・クロックをイネーブルします。
- 電源制御レジスタ (PWR_CR) の DBP ビットをセットし、バックアップ・レジスタと RTC へのアクセスをイネーブルします。

15.2 RTC の主な機能

- プログラマブルなプリスケーラ：分周比は最大 2^{20}
- 32bit プログラマブルカウンタで長時間の動作が可能
- 2 種の独立したクロック供給：APB1 インタフェースのための PCLK1 と RTC クロック (PCLK1 の 1/4 以下の周波数であることが必要)
- RTC のクロック・ソースは次の 3つがあげられます。
 - 128 で分周された HSE クロック
 - LSE 発振子クロック
 - LSI 発振子クロック（詳細は [セクション 6.2.8: RTC クロック](#) を参照）
- 2 種のリセット動作：
 - APB1 インタフェースはシステム・リセットでリセットされます。
 - RTC コア（プリスケーラ、アラーム、カウンタ、分周回路）はバックアップ・ドメインリセットでのみリセットされます [セクション 6.1.3: バックアップ・ドメイン・リセット \(73 ページ\)](#) を参照）
- 3 種のマスク可能な割込みリクエストライン：
 - アラーム割込み、ソフトウェアで自由に設定可能なアラーム割込みが生成できます。
 - 秒割込み、最大 1 秒までを設定可能な周期割込み信号が生成できます。
 - オーバフロー割込み、内部のカウンタがオーバフローし、カウンタが 0 となったことを検出できます。

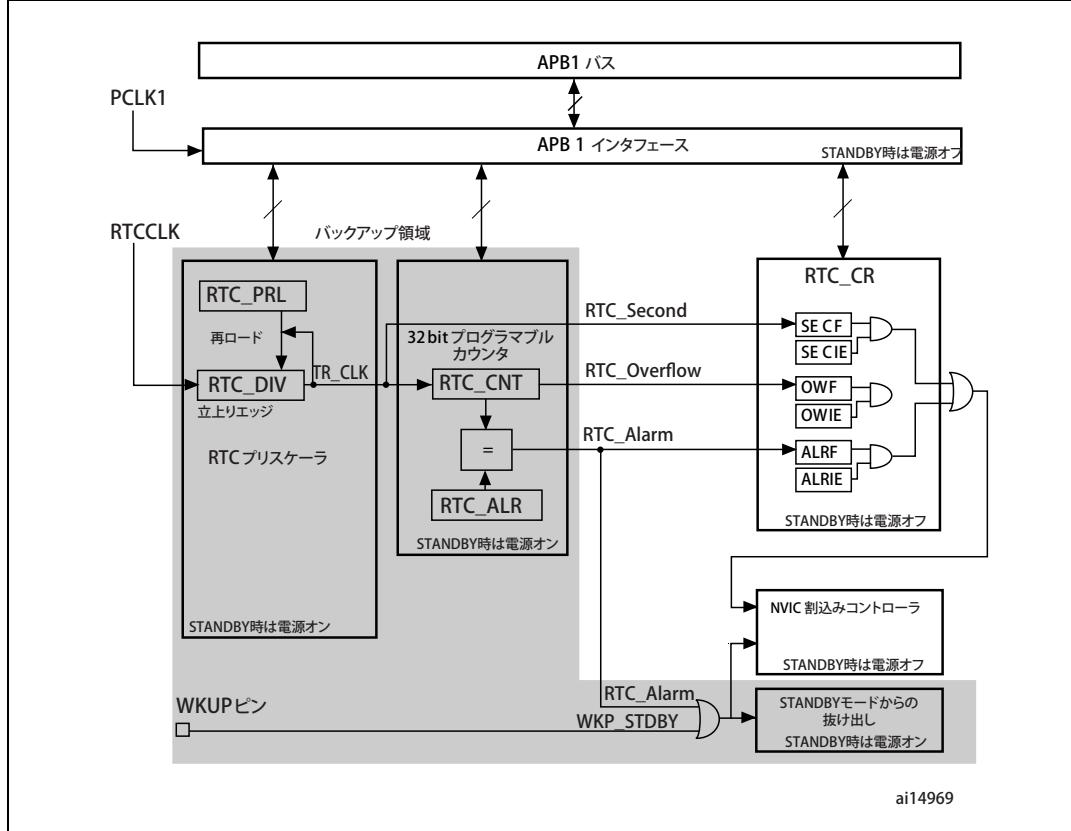
15.3 RTC の機能詳細

15.3.1 概要

RTC は 2 つの基本的なユニットから構成されています。（[図 151 \(356 ページ\) を参照](#)）一方のユニットは APB1 インタフェースで、APB1 バスとのインターフェースとして動作します。このユニットには APB1 バスから読み書き可能な 16bit レジスタがあります。（詳細は [セクション 15.4: RTC レジスタ \(359 ページ\) を参照してください](#)）APB1 インタフェースでは、APB1 バスとの接続のため、APB1 バスクロックが使用されています。

他のユニットは RTC コアで、2 つのブロックに分かれたプログラマブルなカウンタがつながれた構造をしています。一方のブロックは最大 1 秒までの周期を設定可能な RTC プリスケーラで、RTC のタイムベース信号 TR_CLK を生成します。プリスケーラはソフトウェアで設定可能な 20 ビット分周回路で構成されています。RTC_CR レジスタで割込みリクエストが許可されれば、RTC は TR_CLK ごとに割込み（秒割込み）リクエストを発生します。他方のユニットはシステム時刻がセットされる 32 bit のカウンタです。システム時刻は TR_CLK ごとにカウントされ、RTC_ALR レジスタに格納されている時刻と比較されます。一致時には、RTC_CR レジスタで割込みリクエストが許可されれば、アラーム割込みリクエストが発生します。

図 151. RTC ブロック構成



15.3.2 RTC レジスタのリセット

RTC_PRL、RTC_ALR、RTC_CNT 及び RTC_DIV 以外のシステム・レジスタは、システム・リセットもしくはパワー・リセットにより随時リセットされます。

RTC_PRL、RTC_ALR、RTC_CNT 及び RTC_DIV レジスタは、バックアップ・ドメインリセットでのみリセットされます。詳細は [セクション 6.1.3 \(73 ページ\)](#) を参照してください。

15.3.3 RTC レジスタの読み出し

RTC コアは RTC APB1 インタフェースとは完全に独立しています。

ソフトウェアからは、RTC プリスケーラ、カウンタ及びアラーム設定値には APB1 インタフェースを通してアクセスしますが、実際に読み出されるレジスタは内部的に RTC クロックの立上りエッジで更新されており、RTC APB1 クロックに再同期させられます。フラグに関しても、同じ操作がなされています。

このため、APB1 インタフェースがこれまでディセーブル状態であり、イネーブル状態とされた直後に読み込みが行われたときには、内部での更新動作が行われておらず、RTC

APB1 レジスタの最初の読み出しは正しく行われないことがあります。(通常は 0 が読み出されます) これは次の場合に発生する可能性があります。

- システム・リセット、もしくはパワー・リセットが行われたとき。
- MCU が STANDBY モードからウェークアップされた直後。([セクション 4.3: 低電力モード](#) を参照してください。)
- MCU が STOP モードからウェークアップされた直後 ([セクション 4.3: 低電力モード](#) を参照してください。)

上記の場合、RTC コアは動作を続けていますが、APB1 インタフェースはディセーブル状態 (リセット、クロック・オフ、もしくはパワー・オフ) となっています。

したがって、RTC_APB1 インタフェースがディセーブル状態であった後に RTC レジスタを読み出す場合、まずソフトウェアでは、RTC_CRL レジスタのレジスタ同期フラグ (RSF) がハードウェアによってセットされるまで待つことが必要とされます。

なお、RTC_APB1 インタフェースは WFI 及び WFE 命令による低電力モードには影響されないことに注意を払ってください。

15.3.4 RTC レジスタの構成設定

RTC_PRL、RTC_CNT、RTC_ALR レジスタへの書き込みを行うには、ペリフェラルは設定モードになければなりません。RTC_CRL レジスタの CNF ビットをセットすると、このモードに入ります。

更に、RTC レジスタへの書き込みは、前に行った書き込み操作が終了しているときにのみ可能です。この条件をソフトウェアで確認するため、RTC_CR レジスタの RTOFF ステータス・ビットで、レジスタの更新が進行中であるか否かを表示しています。RTOFF ステータス・ビットが 1 の時にのみ、RTC レジスタへの新たな書き込み操作を行うことができます。

モード設定手順:

1. RTOFF をポールし、値が 1 になるまで待ちます。
2. CNF ビットをセットし、設定モードに入れます。
3. RTC レジスタへの書き込み操作を行います。(複数の書き込みを行うことができます。)
4. CNF ビットをクリアし、設定モードを終了します。
5. 書込み完了を確認するため、RTOFF をポールし、値が 1 になるまで待ちます。

書き込み操作は CNF ビットがクリアされているときにのみ行うことができます。書き込み操作は少なくとも 3 RTCCLK サイクルを必要とします。

参考資料

15.3.5 RTC フラグセットの条件

RTC 秒フラグ (SECF) は RTC コアのクロックごとに、RTC カウンタがカウントされる前に、セットされます。

RTC オーバフロー・フラグ (OWF) は、カウンタの値が 0 になる直前の RTC コアクロックでセットされます。

RTC_Alarm 信号と RTC アラームフラグ (ALRF) (図 152 を参照) は、カウンタの値がアラーム・レジスタに格納されている RTC アラームの値より 1 だけ大きな値 (RTC_ALR + 1) に達する直前の RTC コアクロックでセットされます。RTC アラームと RTC 秒フラグへの書き込み操作は次のいずれかの方法でタイミングを合わせて行わねばなりません。

- RTC アラーム割込みを利用し、割込み処理ツーチンの中で RTC アラーム及び RTC カウンタ・レジスタのいずれか、もしくは双方を更新します。
- RTC 制御レジスタの SECF ビットがセットされるのを待ち、RTC アラーム及び RTC カウンタ・レジスタのいずれか、もしくは双方を更新します

図 152. RTC 秒及びアラーム波形 (PR=0003, ALARM=00004)STANDBY モード

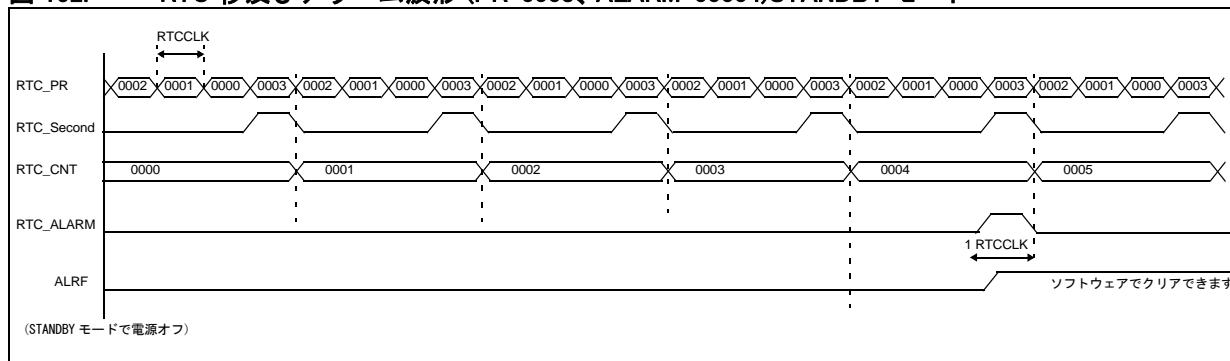
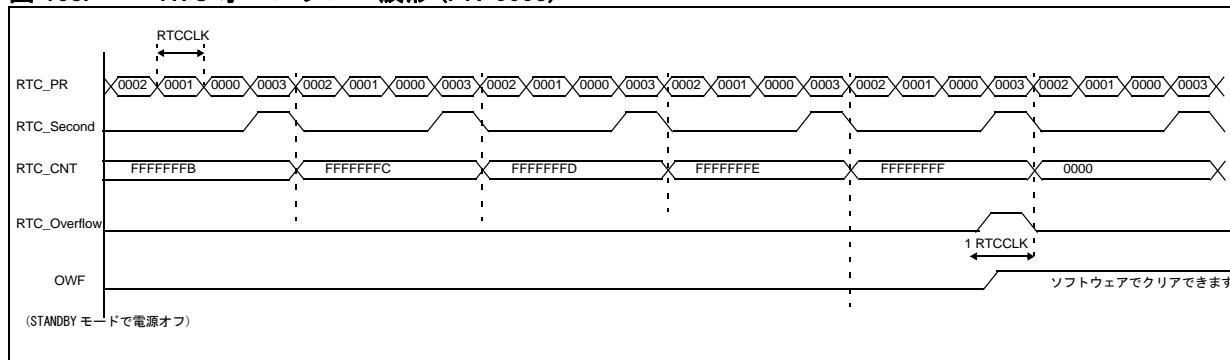


図 153. RTC オーバフロー波形 (PR=0003)



15.4 RTC レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#)を参照してください。

15.4.1 RTC 制御レジスタ（上位）(RTC_CRH)

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												OWIE	ALRIE	SECIE	
												RW	RW	RW	

Bits 15:3 予約済み、ハードウェアで 0 に固定されています。

Bit 2 **OWIE**: オーバフロー割込みリクエスト許可

- 0: オーバフロー割込みリクエスト禁止
- 1: オーバフロー割込みリクエスト許可

Bit 1 **ALRIE**: アラーム割込みリクエスト許可

- 0: アラーム割込みリクエスト禁止
- 1: アラーム割込みリクエスト許可

Bit 0 **SECIE**: 秒割込みリクエスト許可

- 0: 秒割込みリクエスト禁止
- 1: 秒割込みリクエスト許可

これらのビットは割込みリクエストをマスクするために使用されます。リセット直後はすべての割込みリクエストは禁止状態となっており、初期化後に割込みリクエストがペンドィングになっていないことを確かにするための RTC レジスタへの書き込みができます。RTC_CRH レジスタには、前の書き込み操作が完了 (RTOFF=0) するまで、書き込みを行うことができません。[\(セクション 15.3.4 \(357 ページ\) を参照\)](#)

RTC の機能はこの制御レジスタによって制御されています。ビットによっては、特定の書き込み手順が必要とされるものもあります。[\(モード設定手順: を参照\)](#)

参考資料

15.4.2 RTC 制御レジスタ (下位) (RTC_CRL)

アドレスオフセット : 0x04

リセット値 : 0x0020

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								RTOFF	CNF	RSF	OWF	ALRF	SECF		
								r	rw	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	

Bits 15:6 予約済み、ハードウェアで 0 に固定されています。

Bit 5 RTOFF: RTC 動作オフ

RTC は、最後に行われた書き込み操作が完了したか否かの状況を、このビットを使って表示します。このビットが 0 の間は、RTC レジスタへの書き込みはできません。このビットは読み出し専用です。

0: RTC レジスタへの最後の書き込み動作はまだ実行中です。

1: RTC レジスタへの最後の書き込み動作は終了しました。

Bit 4 CNF: 設定モード・フラグ

RTC_CNT、RTC_ALR もしくは RTC_PRL レジスタに書き込みを行うための設定モードに入るには、このビットをソフトウェアでセットしなければなりません。書き込み動作は、ソフトウェアでこのビットを再度クリアした後行われます。

0: 設定モードを終了します。(RTC レジスタの更新が始まります。)

1: 設定モードに入ります。

Bit 3 RSF: レジスタ同期フラグ

RTC_CNT と RTC_DIV レジスタが更新されたとき、もしくはソフトウェアでクリアされたときに、このビットはハードウェアでセットされます。APB1 リセットや APB1 クロック停止の直後の読み出し操作の前には、RTC_CNT、RTC_ALR もしくは RTC_PRL レジスタの内容が同期していることを確認するため、このビットがセットされるまで待たねばなりません。

0: レジスタ内容はまだ同期していません。

1: レジスタ内容は同期しました。

Bit 2 OWF: オーバフロー・フラグ

32bit カウンタがオーバフローしたときに、ハードウェアによってセットされます。RTC_CRH レジスタの OWIE=1 であれば割込みリクエストが発生します。このビットはソフトウェアでクリアできます。1 を書込んだときには、ビットの値に影響を与えません。

0: オーバフローは発生していません。

1: 32 ビットカウンタのオーバフローが発生しました。

Bit 1 ALRF: アラームフラグ

32bit カウンタが RTC_ALR レジスタにセットされている値に達したときに、ハードウェアによってセットされます。RTC_CRH レジスタの ALRIE=1 であれば割込みリクエストが発生します。このビットはソフトウェアでクリアできます。1 を書込んだときには、ビットの値に影響を与えません。

0: アラームは発生していません。

1: アラームが発生しました。

Bit 0 SECF: 秒フラグ

このビットは 32bit のプリスケーラがオーバフローし、RTC カウンタがカウントされ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 RTC カウンタの分解能（多くの場合 1 秒）を周期とする一定間隔の信号を提供します。RTC_CRH レジスタの SECIE=1 であれば割込みリクエストが発生します。このビットはソフトウェアでクリアできます。1 を書込んだときには、ビットの値に影響を与えません。

0: 秒フラグ条件は発生していません。

1: 秒フラグ条件が発生しました。

参考資料

RTC の機能はこの制御レジスタによって制御されています。RTC_CR レジスタには、前の書込み操作が完了 (RTOFF=0) するまで、書込みを行うことができません。([セクション 15.3.4 \(357 ページ\)](#) を参照)

- 注:
- 1 *RTC_CR* リクエスト・フラグがソフトウェアでクリアされるまで、フラグはペンディング状態を保ち、割込みリクエストが発生していることを示します。
 - 2 リセット直後は割込みリクエストは禁止状態で、ペンディング状態の割込みリクエストはありません。また、RTC レジスタへの書込みが可能となっています。
 - 3 *OWF*、*ALRF*、*SECF* 及び *RSF* ビットは APB1 クロックが動作していないときには更新されません。
 - 4 *OWF*、*ALRF*、*SECF* 及び *RSF* ビットはハードウェアでセットされ、ソフトウェアでクリアされます。
 - 5 *ALRF = 1*かつ*ALRIE = 1*のとき、RTC のグローバル割込みはイネーブルとなっています。このとき *EXTI* ライン 17 も *EXTI* コントローラを通してイネーブルであれば、RTC グローバル割込みと RTC アラーム割込みの両方がイネーブルとなります。
 - 6 *ALRF = 1*のとき、*EXTI* ライン 17 が *EXTI* コントローラを通して割込みモードでイネーブルであれば、RTC アラーム割込みもイネーブルとなります。*EXTI* ライン 17 がイベントモードであれば、このラインにパルスが発生します。このとき RTC アラーム割込みは発生しません。

参考資料

15.4.3 RTC プリスケーラ・ロード・レジスタ (RTC_PRLH / RTC_PRLL)

プリスケーラ・ロード・レジスタは RTC プリスケーラの周期カウント値を保持します。この値は RTC_CR レジスタの RTOFF ビットで書き込み保護されており、RTOFF の値が 1 のときのみ書き込み可能となります。

RTC プリスケーラ・ロード・レジスタ (上位) (RTC_PRLH)

アドレスオフセット : 0x08
書き込みのみ ([セクション 15.3.4 \(357 ページ\)](#) 参照)
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								PRL[19:16]							
W								W							

Bits 15:4 予約済み、ハードウェアで 0 に固定されています。

Bits 3:0 PRL[19:16]: RTC プリスケーラ再ロード値 (上位)

これらのビットはカウンタ・クロックの周波数を定めるために使用されます。周波数は次の式で決まります。

$$f_{TR_CLK} = f_{RTCCLK} / (PRL[19:0] + 1)$$

注意: 0 を設定することは好ましくありません。RTC 割込みリクエストとフラグが正しくセットされません。

RTC プリスケーラ・ロード・レジスタ (下位) (RTC_PRLL)

アドレスオフセット : 0x0C
書き込みのみ ([セクション 15.3.4 \(357 ページ\)](#) 参照)
リセット値 : 0x8000

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

Bits 15:0 PRL[15:0]: RTC プリスケーラ再ロード値 (下位)

これらのビットはカウンタ・クロックの周波数を定めるために使用されます。周波数は次の式で決まります。

$$f_{TR_CLK} = f_{RTCCLK} / (PRL[19:0] + 1)$$

注意: 入力クロックの周波数 (f_{RTCCLK}) が 32.768 kHz の場合、1 秒の周期を得るにはこのレジスタに 7FFFh をセットします。

15.4.4 RTC プリスケーラ分周レジスタ (RTC_DIVH / RTC_DIVL)

TR_CLK の個々の周期ごとに、RTC プリスケーラ内部のレジスタには RTC_PRL レジスタの値が再設定されます。精度の高い時間単位を得るために、動作を停止することなく、RTC_DIV レジスタに格納されているその時点のプリスケーラの値を読出することができます。このレジスタは読み出し専用であり、RTC_PRL もしくは RTC_CNT レジスタが変更されたときには、その値がハードウェアによってロードされます。

RTC プリスケーラ分周レジスタ (上位) (RTC_DIVH)

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												RTC_DIV[19:16]			
r r r r r r r r r r r r												r r r r			

Bits 15:4 予約済み

Bits 3:0 RTC_DIV[19:16]: RTC クロック分周カウンタ (上位)

RTC プリスケーラ分周レジスタ (下位) (RTC_DIVL)

アドレスオフセット : 0x14
リセット値 : 0x8000

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

Bits 15:0 RTC_DIV[15:0]: RTC クロック分周カウンタ (下位)

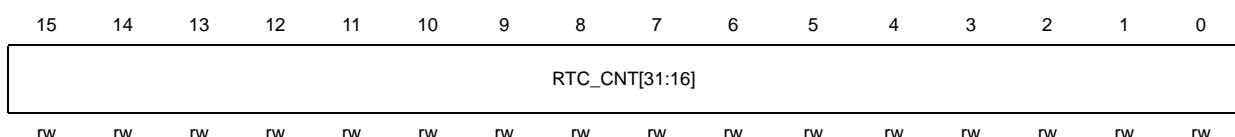
参考資料

15.4.5 RTC カウンタ・レジスタ (RTC_CNTH / RTC_CNTL)

RTC コアはプリスケーラで生成された TR_CLK タイムベースで駆動される 32bit のカウンタを持っており、これは 2 個の 16bit レジスタを通してアクセスされます。2 個の RTC_CNTL レジスタは、このカウンタの値を保持しています。このレジスタは RTC_CR レジスタの RTOFF ビットで書き込み保護がなされており、RTOFF ビットが 1 の時にのみ書き込むことができます。上位 (RTC_CNTH) 及び下位 (RTC_CNTL) のレジスタへの書き込み操作は、対応しているカウンタへの直接の書き込みとなり、その際に RTC プリスケーラに再ロードが行われます。読み出しへはカウンタの値（システム時刻）が戻されます。

RTC カウンタ・レジスタ (上位) (RTC_CNTH)

アドレスオフセット : 0x18
リセット値 : 0x0000

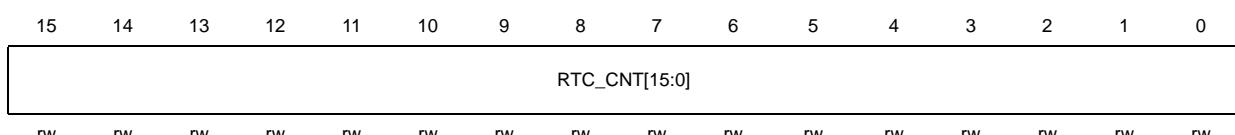


Bits 15:0 RTC_CNT[31:16]: RTC カウンタ (上位)

RTC_CNTH レジスタの読み出しへは、RTC カウンタの上位の値が戻されます。このレジスタへの書き込みには、RTC_CR レジスタの RTOFF ビットで設定モードにされていなければなりません。（[セクション 15.3.4: RTC レジスタの構成設定 \(357 ページ\)](#) 参照）

RTC カウンタ・レジスタ (下位) (RTC_CNTL)

アドレスオフセット : 0x1C
リセット値 : 0x0000



Bits 15:0 RTC_CNT[15:0]: RTC カウンタ (下位)

RTC_CNTL レジスタの読み出しへは、RTC カウンタの下位の値が戻されます。このレジスタへの書き込みには、RTC_CR レジスタの RTOFF ビットで設定モードにされていなければなりません。（[セクション 15.3.4: RTC レジスタの構成設定 \(357 ページ\)](#) を参照）

15.4.6 RTC アラーム・レジスタ（上位）(RTC_ALRH / RTC_ALRL)

32bit カウンタの値が RTC_ALR レジスタに格納されている値に達すると、アラームが生じ、RTC_alarm 割込みリクエストが発生します。このレジスタは RTC_CR レジスタの RTOFF ビットによって書き込み保護がなされており、RTOFF ビットの値が 1 の時にのみ書き込み操作が可能です。

RTC アラーム・レジスタ（上位）(RTC_ALRH)

アドレスオフセット : 0x20

書き込みのみ ([セクション 15.3.4 \(357 ページ\)](#) を参照)

リセット値 : 0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTC_ALR[31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Bits 15:0 RTC_ALR[31:16]: RTC アラーム（上位）

ソフトウェアでアラーム時刻の上位の値を書き込みます。このレジスタへの書き込みは RTC_CR レジスタの RTOFF ビットで指示される 設定モードになければなりません。([セクション 15.3.4: RTC レジスタの構成設定 \(357 ページ\)](#) を参照)

RTC アラーム・レジスタ（下位）(RTC_ALRL)

アドレスオフセット : 0x24

書き込みのみ ([セクション 15.3.4 \(357 ページ\)](#) を参照)

リセット値 : 0xFFFF

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

Bits 15:0 RTC_ALR[15:0]: RTC アラーム（下位）

ソフトウェアでアラーム時刻の下位の値を書き込みます。このレジスタへの書き込みは RTC_CR レジスタの RTOFF ビットで指示される 設定モードになければなりません。([セクション 15.3.4: RTC レジスタの構成設定 \(357 ページ\)](#) を参照)

參考資料

15.4.7 RTC レジスタ・マップ

RTC レジスタは、次の表のように、すべて 16bit のレジスタです。

表 61. RTC - レジスタ・マップとリセット時の値

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	RTC_CRH リセット値																																			
0x004	RTC_CRL リセット値																																			
0x008	RTC_PRLH リセット値																																PRL[19:16]			
0x00C	RTC_PRLL リセット値																																			
0x010	RTC_DIVH リセット値																														DIV[31:16]					
0x014	RTC_DIVL リセット値																														DIV[15:0]					
0x018	RTC_CNTH リセット値																														CNT[13:16]					
0x01C	RTC_CNTL リセット値																														CNT[15:0]					
0x020	RTC_ALRH リセット値																														ALR[31:16]					
0x024	RTC_ALRL リセット値																														ALR[15:0]					

レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。

16 独立型ウォッチドッグ (IWDG)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適用されます。

16.1 概要

STM32F10xxx は、高い安全レベルと時間精度が高く自由度の大きな、2 個のウォッチドッグ回路を搭載しています。両ウォッチドッグ回路（独立型とウィンドウ）はソフトウェアの異常に起因する誤動作を検出及び解消し、カウンタが与えられたタイムアウト値に達したときに割込みもしくはシステム・リセットをトリガする役割を果たします。

独立型ウォッチドッグ (IWDG) は独自の低速クロック (40kHz) を使用し、メインクロックに異常があったときにも有効に動作します。Window 型ウォッチドッグ (WWDG) は APB1 クロックをプリスケーラを通して使用して時間の窓 (time-window) を作り、ソフトウェアで設定したアプリケーション動作の速すぎ / 遅れを検出します。

IWDG はアプリケーションとは独立して動作するウォッチドッグを必要とする場合に最も適していますが、時間精度が高くないという制約があります。WWDG は正確なタイムスロットで処理を行わなければならないアプリケーションに適しています。

Window 型ウォッチドッグの詳細に関しては [セクション 17 \(373 ページ\)](#) を参照してください。

16.2 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 発振器からのクロック供給 (STANDBY 及び STOP モードで動作可能)
- ダウンカウンタが 0x000 に達した時のリセット (ウォッチドッグ・アクティベート時)

16.3 IWDG の機能詳細

[図 154](#) に独立型ウォッチドッグ回路の機能ブロックが示されています。

キーレジスタ (IWDG_KR) に CCCCh が書込まれることで独立型ウォッチドッグがスタートすると、カウンタはリセット時の値 FFFh からのカウントダウンを開始します。カウンタの値が最終値 (000h) に達するとリセット信号 (IWDG_RESET) が発生します。

IWDG_KR レジスタに AAAAh が書込まれたときには、IWDG_RLR レジスタの値が再びカウンタにロードされ、ウォッチドッグによるリセット生成は中止されます。

16.3.1 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で「ハードウェアウォッチドッグ」機能が有効とされているときには、ウォッチドッグは電源投入と同時に自動的に有効となり、カウンタが最終値に達する前にソフトウェアによるキーレジスタへの書込みが行われないとときにリセットが発生します。

参考資料

16.3.2 レジスタ・アクセスに対する保護

IWDG_PR 及び IWDG_RLR レジスタに対する書き込みアクセスは保護の対象となっており、書き込みアクセスを行うには事前に IWDG_KR レジスタに 5555h を書き込まねばなりません。この IWDG_KR レジスタに他の値を書込むと書き込みシーケンスはアボートされ、レジスタへの書き込みが再び保護の対象となります。通常は、再ロード指示 (AAAAh) が書き込まれることが想定されています。

ステータス・レジスタには、プリスケーラの更新やダウンカウンタへの再ロードが行われていることが表示されます。

16.3.3 デバッグ・モード

マイクロコントローラがデバッグ・モードに入った (Cortex-M3 コアはホールト状態) とき、IWDG は動作を続けるか、停止するかのいずれかとなります。これは DBG モジュールの DWG_IWDG_STOP ビットの指定で選択できます。詳細は [セクション 26.15.2: タイマ、ウォッチドッグ、bxCAN と PEC に対するデバッグ・サポート](#) を参照してください。

図 154. 独立型ウォッチドッグのブロック図



注意:

ウォッチドッグ機能は STOP モード及び STANDBY モードでも機能し得る VDD 電源供給ドメインに組込まれています。

表 62. ウォッチドッグ・タイムアウト時間 (40 kHz クロック入力時)⁽¹⁾

プリスケーラ分周比	PR[2:0] ビット	タイムアウト最小値 (ms) RL[11:0]= 000h	タイムアウト最大値 (ms) RL[11:0]= FFFh
/4	0	0.1	409.6
/8	1	0.2	819.2
/16	2	0.4	1638.4
/32	3	0.8	3276.8
/64	4	1.6	6553.6
/128	5	3.2	13107.2
/256	6 (または 7)	6.4	26214.4

1. このタイミングは 40kHz クロックが供給されたときの値です。ただし、内蔵 RC クロックの周波数は 30kHz から 60kHz の間で変化することがあります。更に、RC クロック周波数を正確に与えても、タイムアウト時間は APB インタフェース・クロックと RC クロック 40kHz の位相のずれにも影響され、最大で RC クロック 1 周期分の不確定さが生じます。

LSI は許容範囲の精度で IWDG タイムアウトを計算するために補正することができます。詳しくは [LSI 周波数調整 \(校正\) \(77 ページ\)](#) を参照してください。

参考資料

RM0008

独立型ウォッチドッグ (IWDG)

16.4 IWDG レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#)を参照してください。

16.4.1 キーレジスタ (IWDG_KR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000 (STANDBY モードでリセットされます)

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
KEY[15:0]															
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bits 31:16 予約済み、0 が読出されます。

Bits 15:0 KEY[15:0]: キー (書き込みのみ有効、読出しでは0)

これらのビットには一定の間隔でソフトウェアが値 AAAAh のキーを書き込まれなければなりません。書き込みがないと、カウンタの値が 0 に達し、ウォッチドッグはリセットが発生します。

キー 5555h の書き込みは IWDG_PR 及び IWDG_RLR レジスタへのアクセスを可能とします。(セクション 16.3.2 を参照)

キー CCCCh の書き込みはウォッチドッグの動作を開始させます。(ハードウェアウォッチドッグがオプションとして選択されていない場合。)

16.4.2 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bits 31:3 予約済み、0 が読出されます。



参考資料

Bits 2:0 PR[2:0]: プリスケーラ分周比

セクション 16.3.2 にあるように、これらのビットは書込みアクセスの保護がなされています。これらのビットにはソフトウェアでプリスケーラの分周比の指定を行います。プリスケーラの分周比の変更は、IWDG_SR レジスタの PVU ビットがクリアされているときでなければなりません。

- 000: 分周比 /4
- 001: 分周比 /8
- 010: 分周比 /16
- 011: 分周比 /32
- 100: 分周比 /64
- 101: 分周比 /128
- 110: 分周比 /256
- 111: 分周比 /256

注: このレジスタの読み込みでは、VDD 電源供給ドメインからプリスケーラの値が読み出されます。その時点でのこのレジスタへの書き込み動作が進行中のときには、読み出された値が更新されていない、もしくは無効な値が読み出されることがあります。読み出された値が正しいことは、IWDG_SR レジスタの PVU ビットがクリアされているときにのみ保証されます。

16.4.3 再ロード値レジスタ (IWDG_RLR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0FFF (STANDBY モードでリセットされます)

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		RL[11:0]													

Bits 31:12 予約済み、0 が読み出されます。

Bits 11:0 RL[11:0]: ウォッチドッグ・カウンタ再ロード値

セクション 16.3.2 にあるように、これらのビットは書込みアクセスの保護がなされています。これらのビットには、IWDG_KR レジスタに AAAAh が書込まれたときにウォッチドッグ・カウンタにロードされる値を、ソフトウェアで書込みます。ウォッチドッグ・カウンタはこの値からカウントダウンされますので、タイムアウトまでの時間はこの値とクロック・プリスケーラの設定値によって決まります。(表 62 を参照)

この再ロード値の変更は IWDG_SR レジスタの RVU ビットがクリアされているときに行わねばなりません。

注: このレジスタの読み込みでは、VDD 電源供給ドメインからプリスケーラの値が読み出されます。その時点でのこのレジスタへの書き込み動作が進行中のときには、読み出された値が更新されていない、もしくは無効な値が読み出されることがあります。読み出された値が正しいことは、IWDG_SR レジスタの RVU ビットがクリアされているときにのみ保証されます。

参考資料

RM0008

独立型ウォッチドッグ (IWDG)

16.4.4 ステータス・レジスタ (IWDG_SR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000 (STANDBY モード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												RVU	PVU	r	r

Bits 31:2 予約済み

Bit 1 RVU: ウォッチドッグ・カウンタ再ロード値の更新

このビットは、再ロード値の書き換え動作が進行中であることを示すため、ハードウェアによりセットされ、VDD ドメインの再ロード値の書換えが完了するとクリアされます。(書換えには RC40kHz クロックで最大 5 サイクルを要します。)

再ロード値の変更（レジスタへの書き込み操作）は RVU ビットがクリアされているときにのみ行うことができます。

Bit 0 PVU: ウォッチドッグ・プリスケーラ値の更新

このビットは、プリスケーラ値の書き換え動作が進行中であることを示すため、ハードウェアによりセットされ、VDD ドメインのプリスケーラ値の書換えが完了するとクリアされます。(書換えには RC40kHz クロックで最大 5 サイクルを要します。)

プリスケーラ値の変更（レジスタへの書き込み操作）は PVU ビットがクリアされているときにのみ行うことができます。

注意：

複数の再ロード値やプリスケーラ値をアプリケーションで使用することが必要な場合、値の変更の前に RVU もしくは PVU ビットがクリアされるのを待つことが必要とされます。他方、変更のための書き込み操作後は、プログラムコードの実行を続けるにあたって、RVU もしくは PVU がクリアされることを待つ必要はありません。（低電力モードに入った場合でも書換え動作は有効で、最後まで続けられます。）



參考資料

16.4.5 IWDG レジスタ・マップ

次の表は IWDG レジスタ・マップとリセット値を示しています。

表 63. IWDG レジスタ・マップとリセット時の値

レジスタのアドレス範囲に関しては [表1 \(36ページ\)](#) を参照してください。

17 Window 型ウォッチドッグ (WWDG)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリー全てに適用されます。

17.1 概要

Window 型ウォッチドッグ (Window Watchdog) は、主に外部の影響や予期しない論理条件などによってアプリケーションを通常の手順から逸脱させることにより発生するソフトウェアの異常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でダウンカウンタの値を更新しない限り、指定された時間の経過後に MCU リセットを発生します。更に、7 ビットのダウンカウンタの値（制御レジスタ内）がウィンドウレジスタの値に達する前にダウンカウンタの値が更新されたときにも MCU リセットが発生します。すなわち、リセットを発生しないためには、カウンタは限られた時間枠 (time-window) の間に更新されなければなりません。

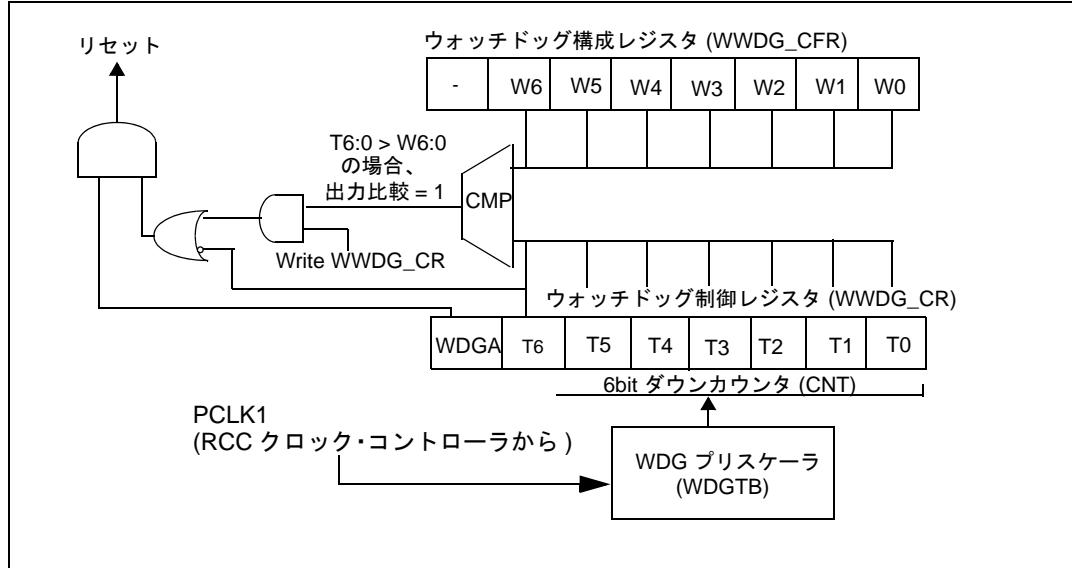
17.2 WWDG の主な機能

- 設定可能な自走ダウンカウンタ
- リセット発生の条件（ウォッチドッグが有効な場合）
 - ダウンカウンタの値が 40h より小さくなったとき。
 - ダウンカウンタに時間枠の外で値がロードされたとき（図 156 を参照）
- アーリーウエーカップ割込み (EWI): ダウンカウンタが 40h になった時トリガ（イネーブルで、ウォッチドッグがアクティベートされている場合）されます。カウンタの再ロード及び WWDG リセットを防ぐために使えます。

17.3 WWDG の機能解説

ウォッチドッグが有効化 (WWDG_CR レジスタの WDGA ビットがセット) されており、7 ビットのダウンカウンタ (T[6:0] フィールド) が 40h から 3Fh に変化した (T6 がクリアされた) とき、リセット動作が開始されます。また、ダウンカウンタの値がウィンドウレジスタの値より大きな間にソフトウェアによるカウンタへのロード操作が行われたときにも、リセットが発生します。

図 155. ウォッチドッグブロック図



MCU リセットを発生させないためには、アプリケーションプログラムの通常の動作の中で一定の間隔で WWDG_CR レジスタへの書き込みを行わねばなりません。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なときに限られます。WWDG_CR レジスタに格納される値は FFh から C0h の間でなければなりません。

- ウォッチドッグを有効とするには :

ウォッチドッグはリセット後は常に無効状態です。これを作動状態とするには、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はウォッチドッグを無効とすることはできません。

- ダウンカウンタを制御するには :

ダウンカウンタは常に働いており、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ています。ウォッチドッグを有効とするときには、直後にリセットが発生するのを防ぐため、T6 ビットはセットしておかねばなりません。

T[5:0] ビットはウォッチドッグがリセットを生じるまでの時間をカウント数で示しています。実際の時間は WWDG_CR レジスタ書き込みの際のプリスケーラの状態によって若干変化します。WWDG_CR レジスタへ書きこむ時のプリスケーラの状態は不明なので、タイミングの値は最小値と最大値の間で様々です。(図 156 を参照)

構成レジスタ (WWDG_CFR) は時間枠を定める上限値を格納しています。リセット発生を防ぐため、ダウンカウンタの更新はカウンタの値が ウィンドウレジスタの値より小さく、3Fh より大きいときに行わねばなりません。図 156 は Window 型ウォッチドッグの動作を説明しています。

参考資料

RM0008

Window 型ウォッチドッグ (WWDG)

カウンタを再ロードするもう 1 つの方法は、アーリー・ウェークアップ割込み (EWI) を使うことです。この割込みは WWDG_CFR レジスタの EWI ビットをセットすることでイネーブルされます。ダウンカウンタが 40h に達すると、この割込みが発生し対応する割込みサービス・ルーチン (ISR) を使いカウンタを再ロードすることで、WWDG リセットを防ぐことが出来ます。この割込みは、WWDG_SR レジスタの EWIF ビットに 0 を書込む事でクリアされます。

注意 : T6 ビットはソフトウェアでリセットを発生させるために利用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)

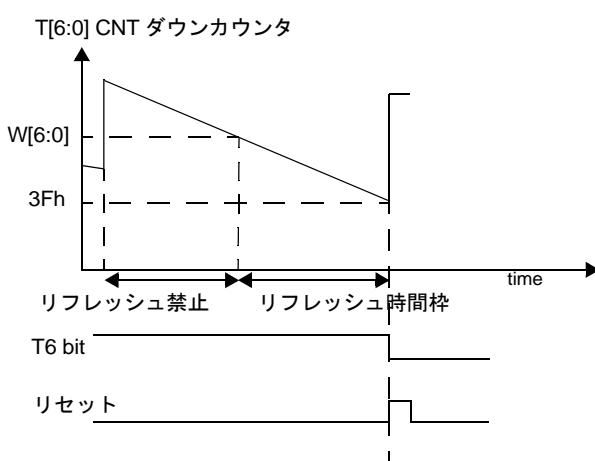


17.4 ウォッチドッグによるタイムアウトを利用するには

図 156 はウォッチドッグ・カウンタ(CNT)にロードされる 6bit の値とこれにより得られるタイムアウト時間(ms 単位)の関係を示しています。これはタイミング変化を考慮しない簡単な計算に使用することができます。タイムアウト時間を算出するには**図 156**の式を使用してください。

要注意: 副作用によるリセット発生を防ぐため、WWDG_CR レジスタへの書き込みでは T6 ビットには常に 1 を書き込んでください。

図 156. Window 型ウォッチドッグのタイミング



タイムアウト時間は次の式で算出されます。:

$$T_{\text{WWDG}} = T_{\text{PCLK1}} \times 4096 \times 2^{\text{WDGTB}} \times (T[5:0] + 1) \quad ;(\text{ms})$$

ここで :

T_{WWDG} : WWDG タイムアウト時間

T_{PCLK1} : APB1 クロック周期 (ms)

タイムアウトの最小値と最大値 @36MHz (PCLK1)

タイムベース : WDGTB	タイムアウト 最小値	タイムアウト 最大値
0	113 μs	7.28 ms
1	227 μs	14.56 ms
2	455 μs	29.12 ms
3	910 μs	58.25 ms

17.5 デバッグ・モード

マイクロコントローラがデバッグ・モードに入った（Cortex-M3 コアはホールト状態）とき、WWDG は動作を続けるか、停止するかのいずれかとなります。これは DBG モジュールの DWG_WWDG_STOP ビットの指定で選択できます。詳細は [セクション 26.15.2: タイマ、ウォッチドッグ、bxCAN と PC に対するデバッグ・サポート](#) を参照してください。

17.6 WWDG レジスタ

レジスタの詳細記述で使用されている略語については [セクション 1.1 \(32 ページ\)](#) を参照してください。

17.6.1 制御レジスタ (WWDG_CR)

アドレスオフセット : 0x00

リセット値 : 0x7F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								WDGA	T6	T5	T4	T3	T2	T1	T0
Res.								rs	rw						

Bits 31:8 予約済み

Bit 7 WDGA: 有効化ビット

このビットはソフトウェアでセットされますが、クリアはリセット後にハードウェアで行われるのみです。WDGA = 1 のとき、ウォッチドッグはリセットを行うことができます。

0: ウォッチドッグは無効（ディセーブル）です。

1: ウォッチドッグは有効（イネーブル）です。

Bits 6:0 T[6:0]: 7bit カウンタ (MSB から LSB)

これらのビットはウォッチドッグのカウンタです。カウンタは (4096×2^{WDGTB}) PCLK1 サイクルごとにカウントダウンされます。カウンタが 40h から 3Fh に変化したとき（T6 がクリアされたとき）にリセットが行われます。

参考資料

17.6.2 構成レジスタ (WWDG_CFR)

アドレスオフセット : 0x04

リセット値 : 0x7F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み					EWI	WDG TB1	WDG TB0	W6	W5	W4	W3	W2	W1	W0	
Res.					rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31:10 予約済み

Bit 9 **EWI:** 先行ウェークアップ割込みリクエスト許可

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 40h に達したときに割込みリクエストが発生します。この割込みリクエストはリセット後にハードウェアによってクリアされるのみです。

Bits 8:7 **WDGTB[1:0]:** タイムベース

プリスケーラのタイムベースを次のように指定します。

- 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

Bits 6:0 **W[6:0]:** 7bit ウィンドウ値

これらのビットはダウンカウンタの値と比較されるウィンドウ値が格納されます。

17.6.3 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 : 0x00

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

Bits 31:1 予約済み

Bit 0 **EWIF:** 先行ウェークアップ割込みリクエスト・フラグ

このビットはカウンタの値が 40h に達したときにハードウェアによってセットされます。このビットはソフトウェアで 0 を書込むことでクリアしなければなりません。1 を書込んだときには、ビットの値に影響を与えません。このビットは割込みリクエストが許可されていないときもセットされます。

参考資料

17.6.4 WWDG レジスタ・マップ

次の表は WWDG のレジスタ・マップとリセット値を示しています。

表 64. WWDG レジスタ・マップとリセット時の値

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 リセット値	予約済み																		WDGA	T[6:0]												
	WWDG_CFR リセット値	予約済み																		EWI	WDGTB1	WDGTB0	W[6:0]										
0x08	WWDG_SR リセット値	予約済み																		EWIF	EWIF								0				

レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。

18 フレキシブル・スタティック・メモリ・コントローラ(FSMC)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、中容量デバイスです。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、大容量デバイスです。

このセクションは、大容量デバイスのみに適応されます。

18.1 FSMC の主な機能

FSMC ブロックは同期、非同期及び 16bit PC メモリカードとの接続が可能です。その主な目的には、次のようなものがあります。

- 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・リクエストへの対応

すべての外部メモリはアドレス、データ及び制御信号をコントローラと共有します。それぞれの外部デバイスは、個別のチップセレクトによりアクセスされます。FSMC は外部デバイスに対して、一度に一回のみのアクセスを行います。

FSMC は以下のような主な機能を搭載します。

- 次のようなスタティック・メモリ・マップ・デバイスとの接続
 - スタティック・ランダム・アクセス・メモリ (SRAM)
 - リード・オンリー・メモリ (ROM)
 - NOR 型 Flash メモリ
 - PSRAM (4 メモリ・バンク)
- Cellular RAM 及び COSMO RAM への同期及び非同期のランダムアクセス
- CRAM としての OneNAND™ Flash による独立した同期及び非同期読み込み及び書き込み
- ECC ハードウェアと 2 バンクの NAND 型フラッシュにより 8Kbytes までのデータをチェック
- 16bit PC カード
- 同期デバイスへのバースト・モードアクセス (NOR 型 Flash 及び PSRAM)
- 8 あるいは 16bit 幅のデータ・バス
- 各メモリ・バンクへの独立したチップセレクト
- 各メモリ・バンクの個別設定
- 以下のようなプログラマブル・タイミングによる広範囲なデバイスサポート
 - プログラマブル・ウェイト・ステート (15 まで)
 - プログラマブル・バス・ターンアラウンド・サイクル (15 まで)
 - プログラマブルな出力イネーブル及び書き込みイネーブル遅延 (15 まで)
 - 独立した読み込み、書き込みタイミング及びプロトコルにより、広範囲なメモリ及びタイミングをサポート
- PSRAM 及び SRAM デバイス用書き込みイネーブルとバイト・レーン・セレクト出力
- 16bit または 8bit 外部デバイスのために、32bit 幅の AHB トランザクションの 16bit あるいは 8bit アクセスへの連続変換

- 32bit 幅ワードが 16 段の書き込み FIFO により、遅いメモリへのアクセスを可能とし、AHB をいち早く開放する事ができます。FSMC への新たなトランザクションが開始されると、まず FIFO からデータを取り出します。
- 外部同期ウェイト制御

外部デバイスタイプ及びその特性を定義する FSMC のレジスタは、起動時に通常設定され、次のリセットまたはパワー・アップまで変更しません。しかしながら、設定はいつでも変更可能です。

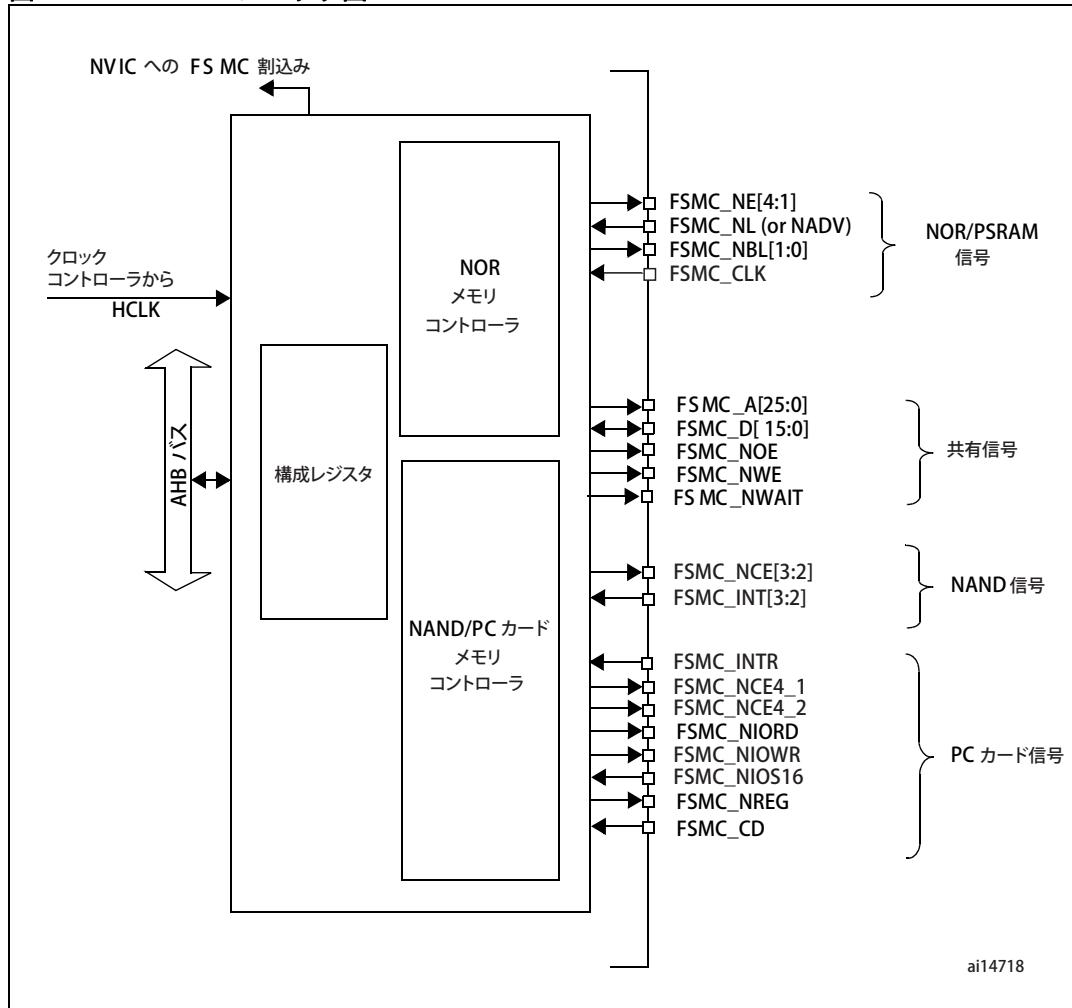
18.2 ブロック図

FSMC には 4 つの主なブロックがあります。：

- AHB インタフェース (FSMC 構成レジスタ含む)
- NOR 型 Flash / PSRAM コントローラ
- NAND 型 Flash / PC カード・コントローラ
- 外部デバイスインターフェース

ブロック図は [図 157](#) を参照してください。

図 157. FSMC ブロック図



18.3 AHB インタフェース

AHB スレーブインターフェースは、外部スタティック・メモリへアクセスするために、内部CPU及び他のバス・マスター・ペリフェラルをイネーブルします。

AHB トランザクションは、外部デバイス・プロトコルへ変換されます。特に、選択された外部メモリが 16bit または 8bit 幅の場合には、AHB の 32bit 幅のトランザクションは 16bit または 8bit の連続したアクセスへ分割されます。

AHB クロック (HCLK) は FSMC へのリファレンスクロックです。

18.3.1 サポートされるメモリ及びトランザクション

一般的なトランザクション規則

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、32bit 幅の何れかをリクエストできます。これは、転送に矛盾を引き起こすかもしれません。

そのため、次のような幾つかの単純な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・サイズが同じであれば、問題はありません。
- AHB トランザクションサイズがメモリ・サイズより大きいと、FSMC は外部データ幅に合わせて AHB トランザクションを連続した小さなメモリ・アクセスへ分割します。
- AHB トランザクションサイズがメモリ・サイズより小さいと、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かもしれません。
 - SRAM、ROM、PSRAM 等のバイトセレクトを持つデバイスへの非同期アクセスの場合、FSMC は読み込み、書き込みトランザクションを許可し、バイト・レーン BL[1:0] を通して正しいデータへアクセスします。
 - NOR 型や 16bit NAND 型 Flash 等のバイトセレクトを持たないデバイスへの非同期アクセスのように、16bit 幅の Flash メモリへのバイトアクセスでは、明らかにデバイスへのバイトアクセスはできません。(Flash メモリへのアクセスは、16bit 単位の読み込み、書き込みのみです) そのため、
 - a) 書込みトランザクションは許可されません。
 - b) 読込みトランザクションは許可されます。(コントローラは 16bit メモリワード全てを読み込み、必要なバイトのみを使います。)

構成レジスタ

FSMC はレジスタセットにより設定可能です。NOR 型 Flash / PSRAM 制御レジスタの詳細設定については [セクション 18.5.6](#) を参照してください。NAND 型 Flash / CompactFlash 制御レジスタの詳細設定については [セクション 18.6.6](#) を参照してください。

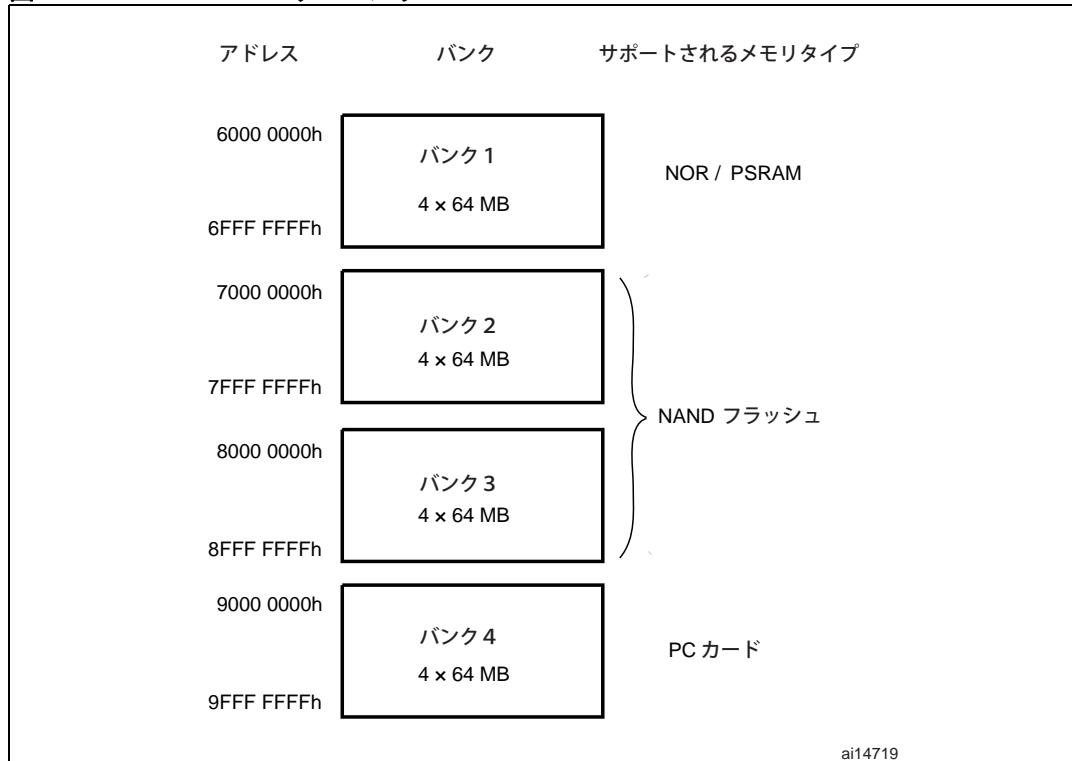
18.4 外部デバイス・アドレス・マッピング

FSMC から見ると、外部メモリは各 256Mbytes の決まった容量の 4 つのバンクに分けられます。(図 158 を参照)

- バンク 1 は、4つまでの NOR 型 Flash または PSRAM メモリデバイスのアドレスに使われます。このバンクは 4 つの独立したチップセレクトにより 4 つの NOR/PSRAM 領域に分割されます。
- バンク 2 と 3 は、NAND 型 Flash デバイス(バンク毎に 1 デバイス)のアドレスに使われます。
- バンク 4 は、PC カードデバイスのアドレスに使われます。

各バンクで使われるメモリのタイプは、構成レジスタで設定されます。

図 158. FSMC メモリ・バンク



18.4.1 NOR/PSRAM アドレス・マッピング

HADDR[27:26] ビットは、表 65 に示されるように 4 つのメモリ・バンクの選択を指定します。

表 65. NOR/PSRAM バンク選択

HADDR[27:26]	選択されるバンク
00	Bank 1 NOR/PSRAM 1
01	Bank 1 NOR/PSRAM 2
10	Bank 1 NOR/PSRAM 3
11	Bank 1 NOR/PSRAM 4

参考資料

HADDR[25:0] は、外部メモリ・アドレスです。HADDR はメモリがワード単位でアクセスされるのに反してバイト・アドレスであるので、実際に使われるアドレスは次の表に示されるようにメモリのデータ幅により変わってきます。

表 66. 外部メモリ・アドレス

メモリ幅	メモリへ供給されるデータアドレス	メモリ最大容量(ビット)
8bit	HADDR[25:0]	64Mbytes × 8 = 512Mbit
16bit	HADDR[25:1] >> 1	64Mbytes/2 × 16 = 512Mbit
32bit	HADDR[25:2] >> 2	64Mbytes/4 × 32 = 512Mbit

NOR型Flash/PSRAM用のラップサポート

各 NOR 型 Flash/PSRAM メモリ・バンクは、ラップ・アクセスをサポートするよう設定可能です。

メモリ側では、同期と非同期の 2 つのアクセス・モードのケースについて考える必要があります。

- 非同期モード**：このケースでは、全てのシングル・アクセスについてアドレスが与えられている限り、ラップ・アドレスはフルにサポートされます。
- 同期モード**：このケースでは、FSMC は一度だけアドレスを供給し、その後は FSMC クロックの CLK によるバースト転送が行われます。

幾つかの NOR 型メモリはラップ・アラウンド・アクセスにより、連続したモジュロ N (N は通常 8、16 または 32 で、NOR 型 Flash の構成レジスタで設定されます) アドレスから決まったワード数を読み出すリニアバーストをサポートします。このケースでは、メモリ・ラップ・モードを AHB マスタ・ラップ・モードと同じに設定することができます。

そうでなければ、メモリ・ラップ・モードと AHB マスタ・ラップ・モードが同じに設定できない場合には、ラッピングを無効にする (FSMC 構成レジスタの該当するビットにより) 必要があります。そしてラップ・トランザクションは 2 つの連続したリニア・トランザクションに分割されます。

18.4.2 NAND/PC カード・アドレス・マッピング

このケースでは 3 つのバンクが使用可能で、それぞれのバンクは、表 67 に示されるように分割されます。

表 67. メモリ・マッピングとタイミング・レジスタ

スタートアドレス	エンドアドレス	FSMC バンク	メモリ・スペース	タイミング・レジスタ
0x9C00 0000	0x9FFF FFFF	バンク 4 - PC カード	I/O	FSMC_PIO4 (0xB0)
0x9800 0000	0x9BFF FFFF		属性	FSMC_PATT4 (0xAC)
0x9000 0000	0x93FF FFFF		コモン	FSMC_PMEM4 (0xA8)
0x8800 0000	0x8BFF FFFF	バンク 3 - NAND 型 Flash	属性	FSMC_PATT3 (0x8C)
0x8000 0000	0x83FF FFFF		コモン	FSMC_PMEM3 (0x88)
0x7800 0000	0x7BFF FFFF	バンク 2 - NAND 型 Flash	属性	FSMC_PATT2 (0x6C)
0x7000 0000	0x73FF FFFF		コモン	FSMC_PMEM2 (0x68)

参考資料

NAND 型 Flash メモリでは、コモン及び属性メモリ・スペースは下位 256Kbyte に位置する 3 つのセクション（表 68 を参照）に分割されます。

- データ・セクション（コモン / 属性メモリ・スペースの最初の 64Kbyte）
- コマンド・セクション（コモン / 属性メモリ・スペースの 2 番目の 64Kbyte）
- アドレス・セクション（コモン / 属性メモリ・スペースの次の 128Kbyte）

表 68. NAND バンク選択

セクション名	HADDR[17:16]	アドレス範囲
アドレス・セクション	1X	0x020000-0x03FFFF
コマンド・セクション	01	0x010000-0x01FFFF
データ・セクション	00	0x000000-0x0FFFF

アプリケーション・ソフトウェアは、NAND 型 Flash メモリへのアクセスに 3 つのセクションを使います。

- NAND 型 Flash メモリへコマンドを送る**：ソフトウェアは、コマンド・セクションの何れかのメモリ・ロケーションへ、コマンドの値を書込む必要があります。
- 読み込みまたは書き込みが必要な NAND 型 Flash アドレスを指定する**：ソフトウェアは、アドレス・セクションの何れかのメモリ・ロケーションへ、アドレスの値を書込む必要があります。アドレスは 4 または 5 バイト長（実際のメモリ・サイズに依存）であるため、フルアドレスを指定するためにはアドレス・セクションへの幾つかの連続した書き込みが必要となります。
- データの読み込みまたは書き込み**：ソフトウェアは、データ・セクションの何れかのロケーションへの読み込みまたは書き込みが必要となります。

NAND 型 Flash メモリは自動的にアドレスをインクリメントするので、連続したメモリ・ロケーションへのアクセスのために、データ・セクションのアドレスをインクリメントする必要はありません。データ・セクションの同じロケーションへの読み込みまたは書き込みは NAND 型 Flash の次のアドレス上のデータに影響します。

18.5 NOR 型 Flash/PSRAM コントローラ

FSMC は、以下のメモリ・タイプを駆動するための最適な信号タイミングを生成します。

- 非同期 SRAM 及び ROM
 - 8bit
 - 16bit
 - 32bit
- PSRAM（Cellular RAM 及び Cosmo RAM）と OneNAND
 - 非同期モード
 - バースト・モード
- NOR 型 Flash
 - 非同期モードまたはバースト・モード
 - マルチプレクスまたは非マルチプレクス

FSM は、各バンク毎に個別のチップセレクト NE[4:1] を出力します。その他の信号（アドレス、データ、制御）は共有されます。

参考資料

同期アクセスでは、FSMCは選択された外部デバイスへクロック(CLK)を送出します。このクロックは、AHBクロックと同一かそれを分割したものです。各バンクのサイズは固定で 64Mbytes です。

各バンクは、個別に割り当てられたレジスタで設定されます。([セクション 18.6.6 を参照](#))

プログラマブル・メモリ・パラメータは、アクセスタイミング ([表 69 を参照](#)) 及びラップとウェイト管理 (PSRAM とバースト・モードによる NOR 型 Flashへのアクセス) のサポートを含みます。

表 69. 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	ユニット	最小	最大
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)	1	16
アドレス・ホールド	ドレスホールド・フェーズ時間	非同期、多重 I/O	AHB クロック・サイクル(HCLK)	1	16
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	バス・ターンアラウンド・フェーズ時間	非同期及び同期 読み込み	AHB クロック・サイクル(HCLK)	1	16
クロック 分割比	1つのメモリ・クロック・サイクル(CLK)に必要な AHB クロック・サイクル数(HCLK)	同期	AHB クロック・サイクル(HCLK)	1	16
データ遅延	バーストの最初のデータ以前にメモリへ供給するクロック・サイクル数	同期	メモリ・クロック・サイクル(CLK)	2	17

18.5.1 外部メモリ・インターフェース信号

[表 70](#)、[表 71](#)、[表 72](#)は、NOR 型 Flash 及び PSRAM とのインターフェースに通常使用する信号のリストです。

注意: Prefix “N” は関連信号がアクティブ・ローである事を示しています。

NOR 型 Flash、非マルチプレクス I/O

表 70. 非マルチプレクス I/O NOR 型 Flash

FSMC 信号名	I/O	機能
CLK	O	クロック (同期バースト用)
A[25:0]	O	アドレス・バス
D[15:0]	I/O	双方向データ・バス
NE[x]	O	チップセレクト $x = 1..4$
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NWAIT	I	FSMC への NOR 型 Flash ウェイト入力信号

参考資料

NOR型Flashメモリは、16bitワード単位でアドレスされます。最大容量は、512Mbit（26アドレス・ライン）です。

NOR型Flash、マルチプレクスI/O

表 71. マルチプレクスI/O NOR型Flash

FSMC信号名	I/O	機能
CLK	O	クロック（同期バースト用）
A[25:16]	O	アドレス・バス
AD[15:0]	I/O	16bitマルチプレクスト、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	出力イネーブル
NWE	O	書き込みイネーブル
NL(=NADV)	O	ラッチ・イネーブル（このシグナルは、幾つかのNOR型Flashデバイスではアドレス・バリッド、NADV等と呼ばれます）
NWAIT	I	FMSCへのNOR型Flashウェイト入力信号

NOR型Flashメモリは16bitワード単位でアドレスされます。最大容量は、512Mbit（26アドレス・ライン）です。

PSRAM

表 72. PSRAM

FSMC信号名	I/O	機能
CLK	O	クロック（同期バースト用）
A[25:0]	O	アドレス・バス
D[15:0]	I/O	双方向データ・バス
NE[x]	O	チップセレクト $x = 1..4$ （CellularRAMではNCE、CosmoRAMではNCE1と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NL(=NADV)	O	アドレスバリットPSRAM入力（メモリ信号名NADV）
NWAIT	I	FMSCへのPSRAMウェイト入力信号
NBL[1]	O	上位バイトイネーブル（メモリ信号名NUB）
NBL[0]	O	下位バイトイネーブル（メモリ信号名NLB）

PSRAMメモリは、16bitワード単位でアドレスされます。最大容量は512Mbit（26アドレス・ライン）

参考資料

18.5.2 サポートされるメモリ及びトランザクション

次の表73は、サポートされるデバイス、アクセス・モード及びトランザクションを示しています。FSMCで無効な（またはサポートされない）トランザクションはグレイで示します。

表 73. NOR 型 Flash /PSRAM がサポートするメモリ及びトランザクション

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	有効 / 無効	コメント
NOR 型 Flash (マックスト I/O 及びノンマックス ト 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 アクセスへ 分割
	非同期	W	32	16	Y	2つの FSMC アクセスへ 分割
	非同期ページ	R	-	16	N	サポートされないモード
	同期	R	8	16	N	
	同期	R	16	16	Y	
	同期	R	32	16	Y	
PSRAM	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 アクセスへ 分割
	非同期	W	32	16	Y	2つの FSMC アクセスへ 分割
	非同期ページ	R	-	16	N	サポートされないモード
	同期	R	8	16	N	
	同期	R	16	16	Y	
SRAM と ROM	非同期	R	8 / 16 / 32	8 / 16 / 32	Y	バイト・レーン NBL[1:0] の使用
	非同期	W	8 / 16 / 32	8 / 16 / 32	Y	バイト・レーン NBL[1:0] の使用

18.5.3 一般的なタイミング規則

信号の同期

- 全てのコントローラ出力信号は、内部クロック (HCLK) の立上がりエッジで変化します。
- 同期の書き込みモード (PSRAM デバイス) では、出力信号はメモリ・クロック (CLK) の立下りエッジで変化します。

フィードバック・クロック

同期モードでアクセスされるメモリ (NOR 型 Flash と PSRAM) は、コントローラから供給されるクロックと、外部メモリに対する実際の信号間の遅延に反映するフィードバック・クロック・メカニズムが実装されています。

18.5.4 NOR 型 Flash/PSRAM コントローラ・タイミング図

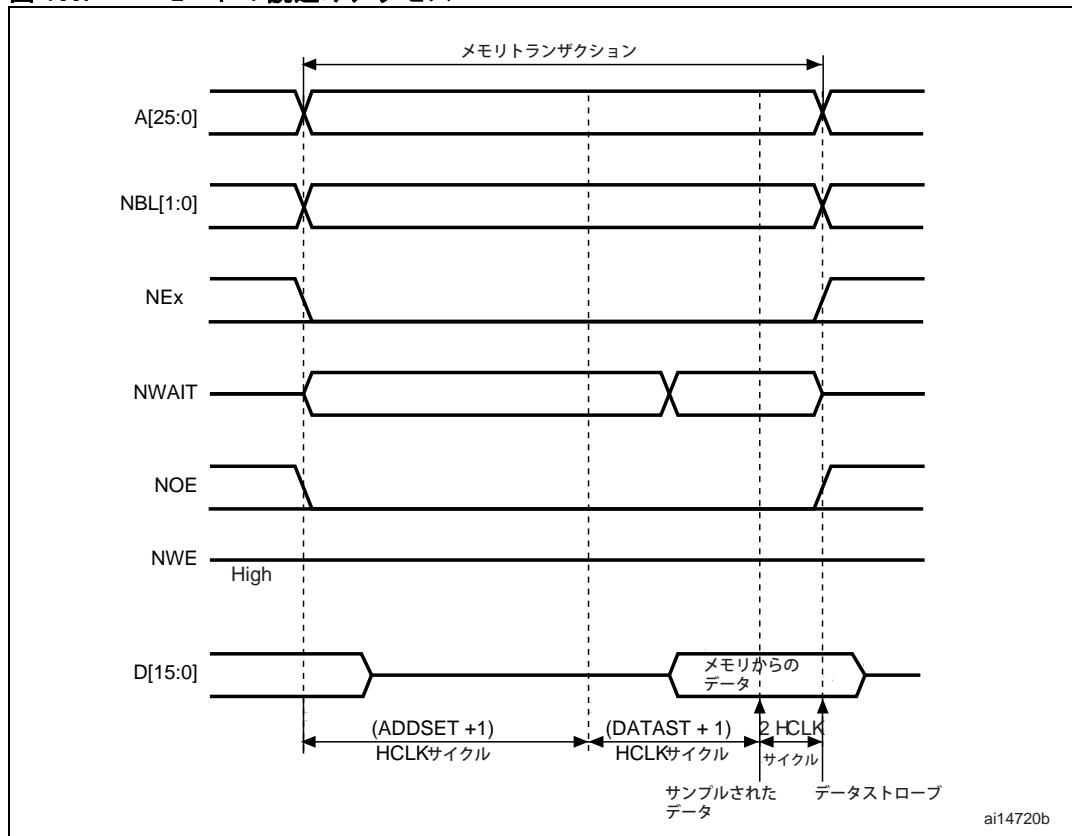
非同期スタティック・メモリ (NOR 型 Flash、SRAM、CRAM)

- 信号は内部クロック HCLK と同期しています。このクロックはメモリには供給されません。
- 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ルします。これにより、メモリデータホールドタイミングに関する規則を満たすことを保証します。(通常、チップイネーブルハイからデータ遷移は最低 0ns です)
- 拡張モードがセットされていると、読み込み及び書き込みにおいて A、B、C、と D モードがミックスされる可能性があります。(例えば、モード A で読み込み、モード B で書き込むことが可能です)

参考資料

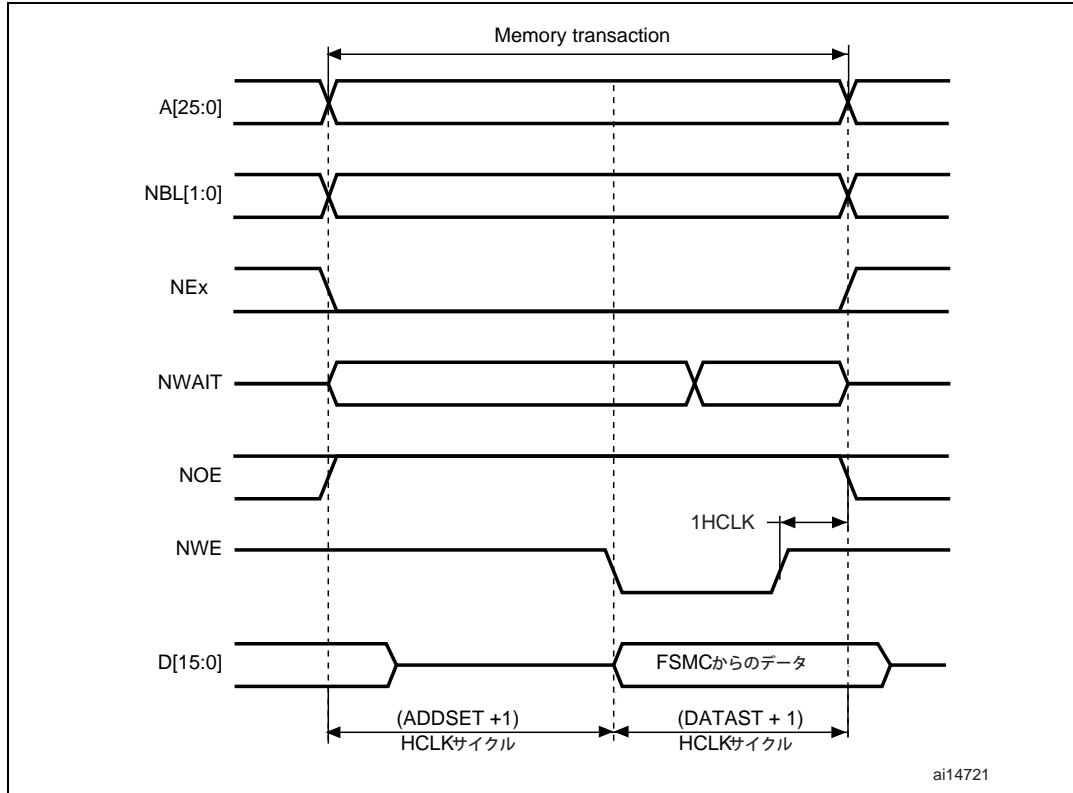
モード 1 – SRAM/CRAM

図 159. モード 1 読込みアクセス



参考資料

図 160. モード 1 書込みアクセス



書込みトランザクションの最後の 1HCLK サイクルは、NWE の立上がりエッジ後のアドレス及びデータホールド時間を保証するのに役立ちます。この 1 つの HCLK サイクルのため DATAST の値はゼロより大きく (DATAST > 0) なければなりません。

表 74. FSMC_BCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-16		0x0000
15	ASYNCWAIT	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。
14-10		0x0
9	WAITPOL	ビット 15 が 1 の場合に限り意味を持ちます。
8	BURSTEN	0x0
7		-
6	FACCEN	-
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、10 (NOR 型 Flash) を除きます。
1	MUXEN	0x0
0	MBKEN	0x1

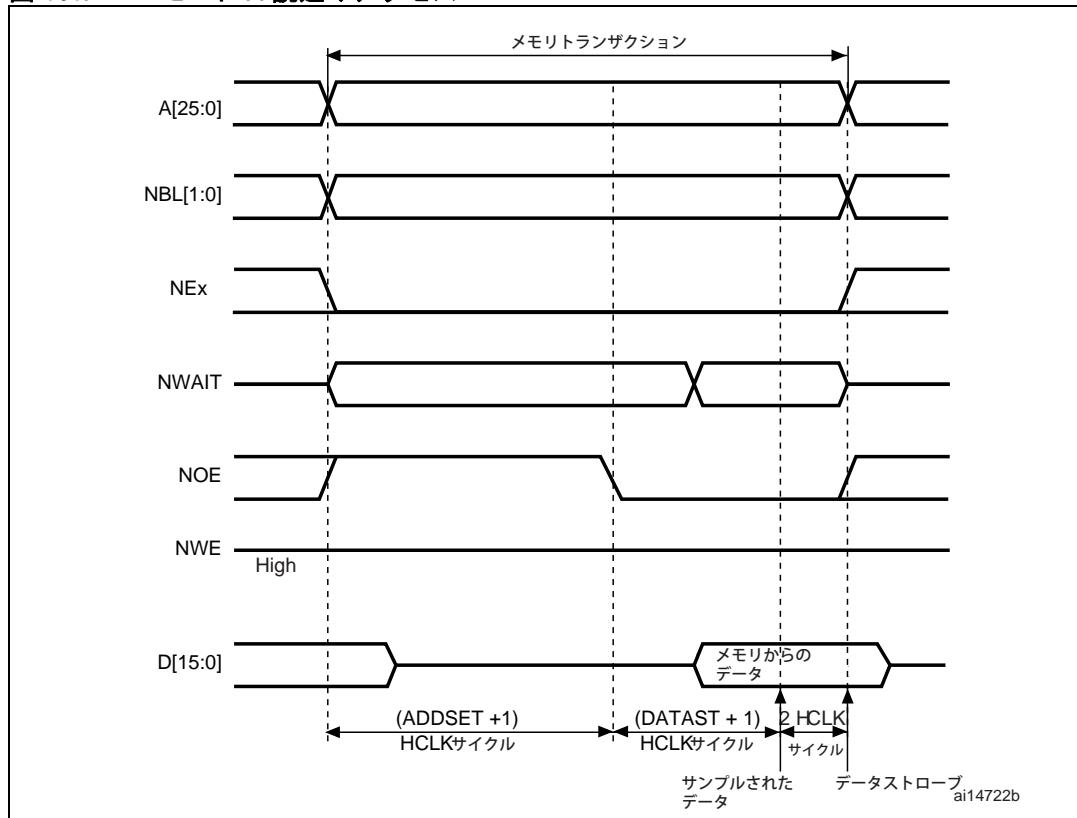
参考資料

表 75. FSMC_TCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-16		0x0000
15-8	DATAST	書込み（読み込みの場合は、DATAST+3HCLK）における2番目のアクセスフェーズ（DATAST + 1HCLK サイクル）の時間です。この値は 0 にはできません。（最低値は 1 です。）
7-4		0x0
3-0	ADDSET	1番目のアクセスフェーズの時間です。（ADDSET+1HCLK サイクル）

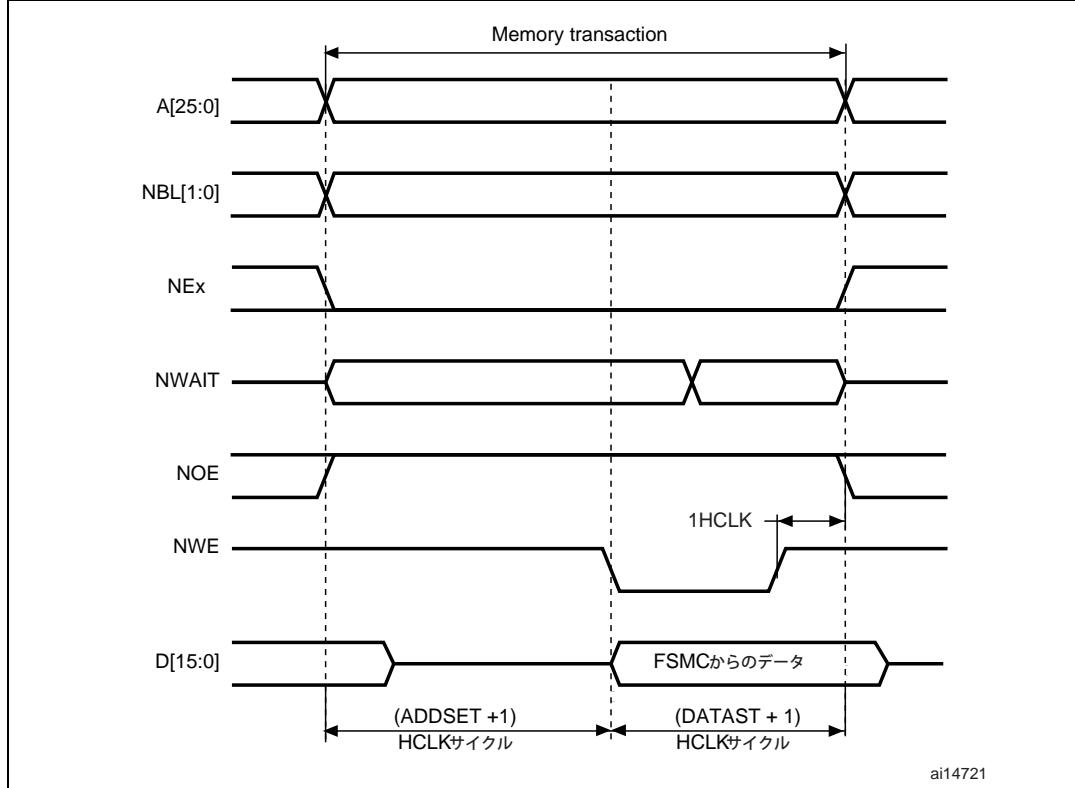
モード A - SRAM/CRAM OE トグリング

図 161. モード A 読込みアクセス



参考資料

図 162. モード A 書込みアクセス



モード 1 と比較した場合の相違点は、NOE のトグリング及び独立した読み込みと書き込みのタイミングです。

参考資料

表 76. FSMC_BCRx ビット・フィールド

ビットナンバ	名称	設定値
31-16		0x0000
15	ASYNCPOL	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 のまとします。
14	EXTMOD	0x1
13-10		0x0
9	WAITPOL	ビット 15 が 1 の場合に限り意味を持ちます。
8	BURSTEN	0x0
7		-
6	FACCEN	-
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、10 (NOR 型 Flash) を除きます。
1	MUXEN	0x0
0	MBKEN	0x1

表 77. FSMC_TCRx ビット・フィールド

ビットナンバ	名称	設定値
31-30		0x0
29-28	ACCMOD	0x0
27-16		0x000
15-8	DATAST	読み込みにおける 2 番目のアクセスフェーズ (DATAST + 3HCLK サイクル) の時間です。この値は 0 には設定できません。(最低値は 1 です。)
7-4		0x0
3-0	ADDSET	読み込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK) の時間

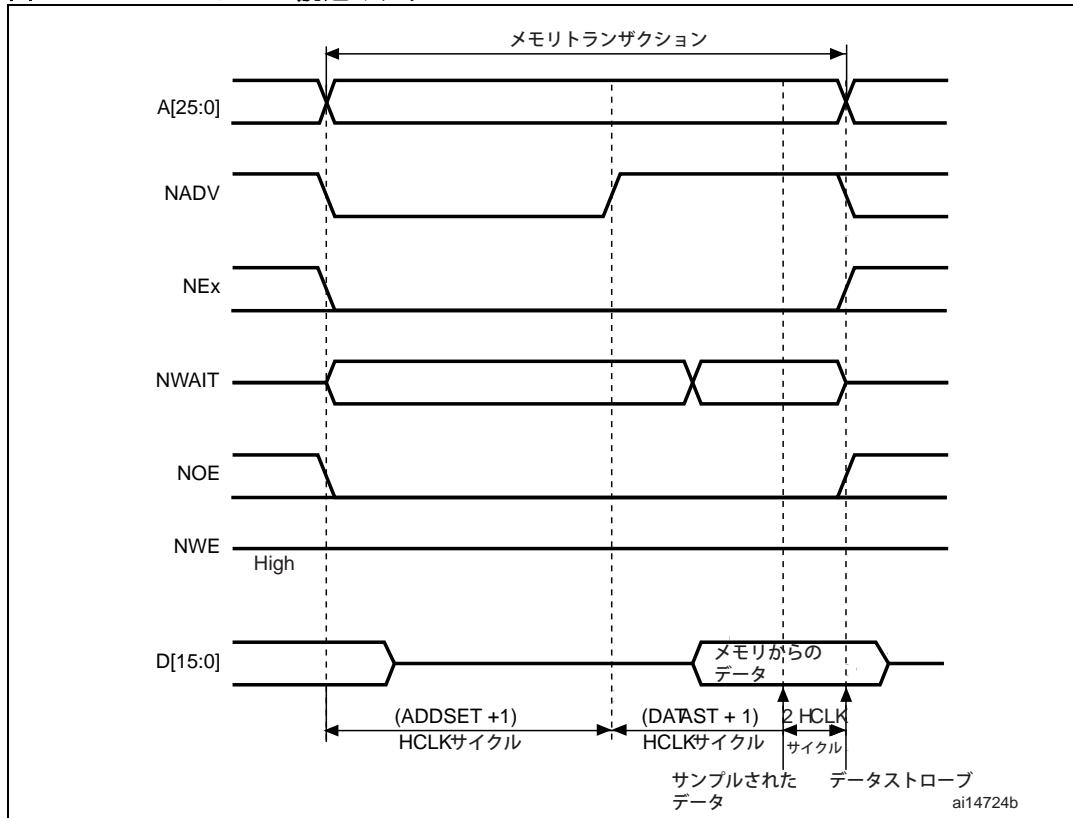
表 78. FSMC_BWTRx ビット・フィールド

ビットナンバ	名称	設定値
31-30		0x0
29-28	ACCMOD	0x0
27-16		0x000
15-8	DATAST	書き込みにおける 2 番目のアクセスフェーズ (DATAST + 1HCLK サイクル) の時間です。この値は 0 にはできません。(最低値は 1 です。)
7-4		0x0
3-0	ADDSET	書き込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK) の時間

参考資料

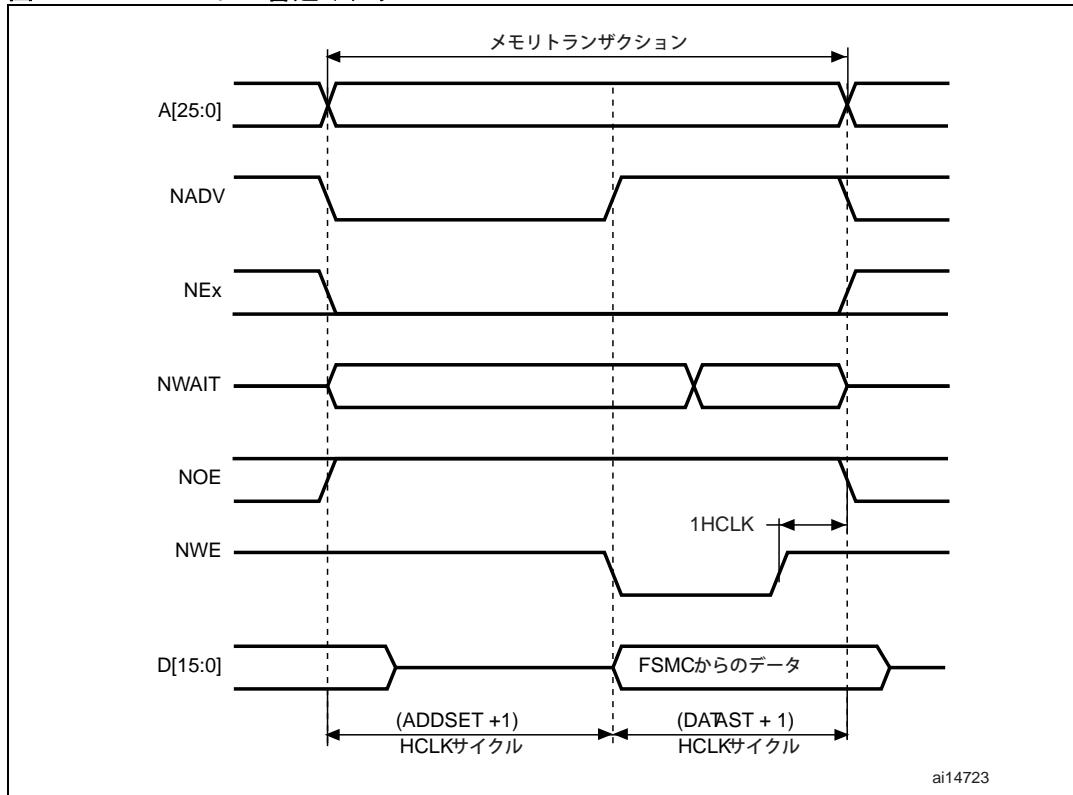
モード 2/B - NOR 型 Flash

図 163. モード 2/B 読込みアクセス



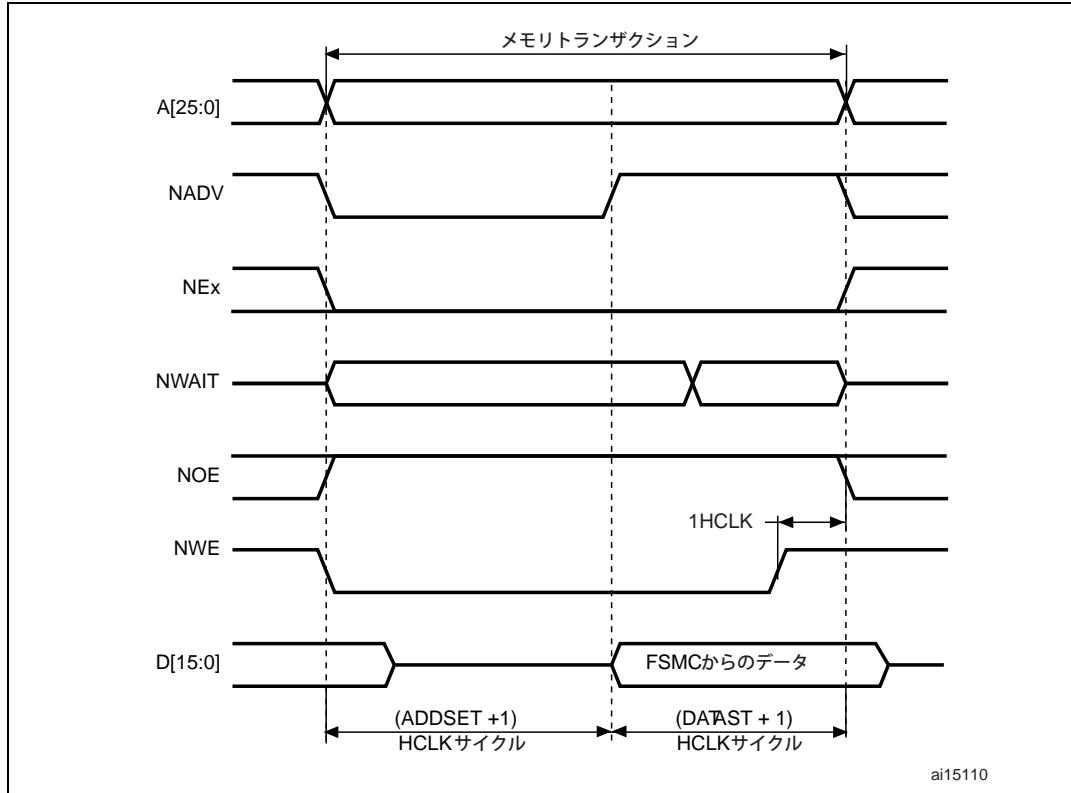
参考資料

図 164. モード 2 書込みアクセス



参考資料

図 165. モード B 書込みアクセス



モード 1 と比較した場合の相違点は、NADV のトグリングと拡張モードがセットされた場合（モード B）の独立した読み込み及び書き込みのタイミングです。

参考資料

表 79. FSMC_BCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-16		0x0000
15	ASYNCEWAIT	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。
14	EXTMOD	モード B では 0x1、モード 2 では 0x0
13-10		0x0
9	WAITPOL	ビット 15 が 1 の場合に限り意味を持ちます。
8	BURSTEN	0x0
7		-
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	10 (NOR 型 Flash)
1	MUXEN	0x0
0	MBKEN	0x1

表 80. FSMC_TCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-30		0x0
29-28	ACCMOD	拡張モードがセットされている場合は 0x1
27-16		0x000
15-8	DATAST	読み込みにおける 2 番目のアクセスフェーズ (DATAST + 3HCLK サイクル) の時間です。この値はゼロには設定できません。(最低値は 1 です。)
7-4		0x0
3-0	ADDSET	読み込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK) の時間

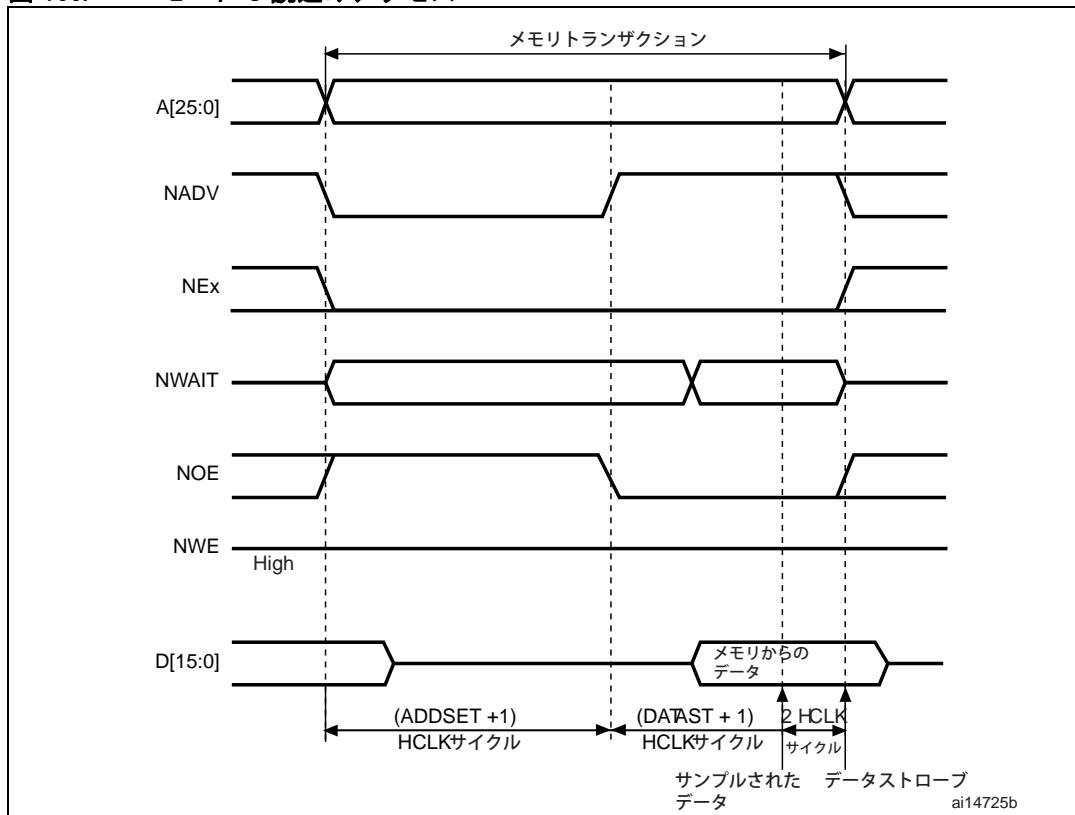
表 81. FSMC_BWTRx ビット・フィールド

ビット ナンバ	名称	設定値
31-30		0x0
29-28	ACCMOD	拡張モードがセットされている場合は 0x1
27-16		0x000
15-8	DATAST	書き込みにおける 2 番目のアクセスフェーズ (DATAST + 1HCLK サイクル) の時間です。この値はゼロにはできません。(最低値は 1 です。)
7-4		0x0
3-0	ADDSET	書き込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK) の時間

注意: *FSMC_BWTRx* レジスタは、拡張モードがセット (モード B) されている場合に限り有効です。それ以外の場合には、その内容は意味を持ちません。

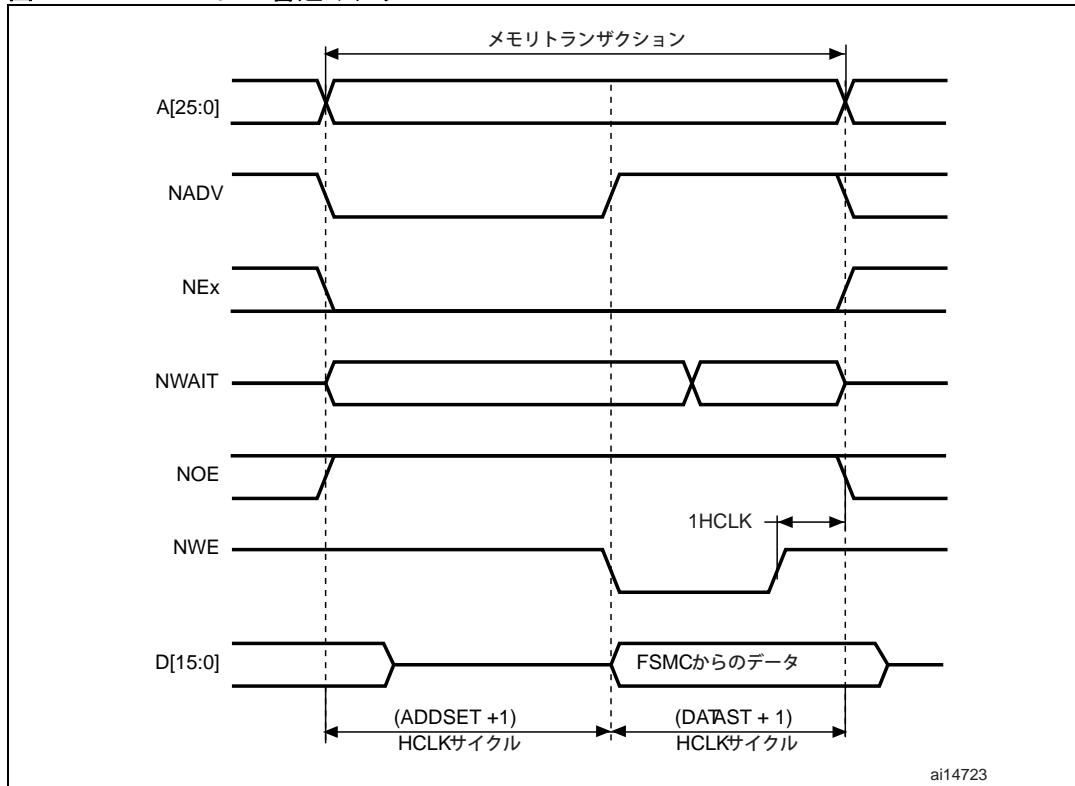
モード C - NOR 型 Flash - OE トグリング

図 166. モード C 読込みアクセス



参考資料

図 167. モード C 書込みアクセス



モード 1 と比較した場合の相違点は、NOE のトグリングと NADV、及び独立した読み書きタイミングです。

表 82. FSMC_BCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-16		0x0000
15	ASYNCWAIT	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。
14	EXTMOD	1
13-10		0x0
9	WAITPOL	ビット 15 が 1 の場合に限り意味を持ちます。
8	BURSTEN	0
7		-
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	10 (NOR 型 Flash)
1	MUXEN	0
0	MBKEN	1

参考資料

表 83. FSMC_TCRx ピット・フィールド

ピットナンバ	名称	設定値
31-30		0x0
29-28	ACCMOD	0x2
27-16		0x000
15-8	DATAST	読み込みにおける 2 番目のアクセスフェーズ (DATAST + 3HCLK サイクル) の時間です。この値は 0 には設定できません。(最低値は 1 です。)
7-4		0x0
3-0	ADDSET	読み込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK) の時間

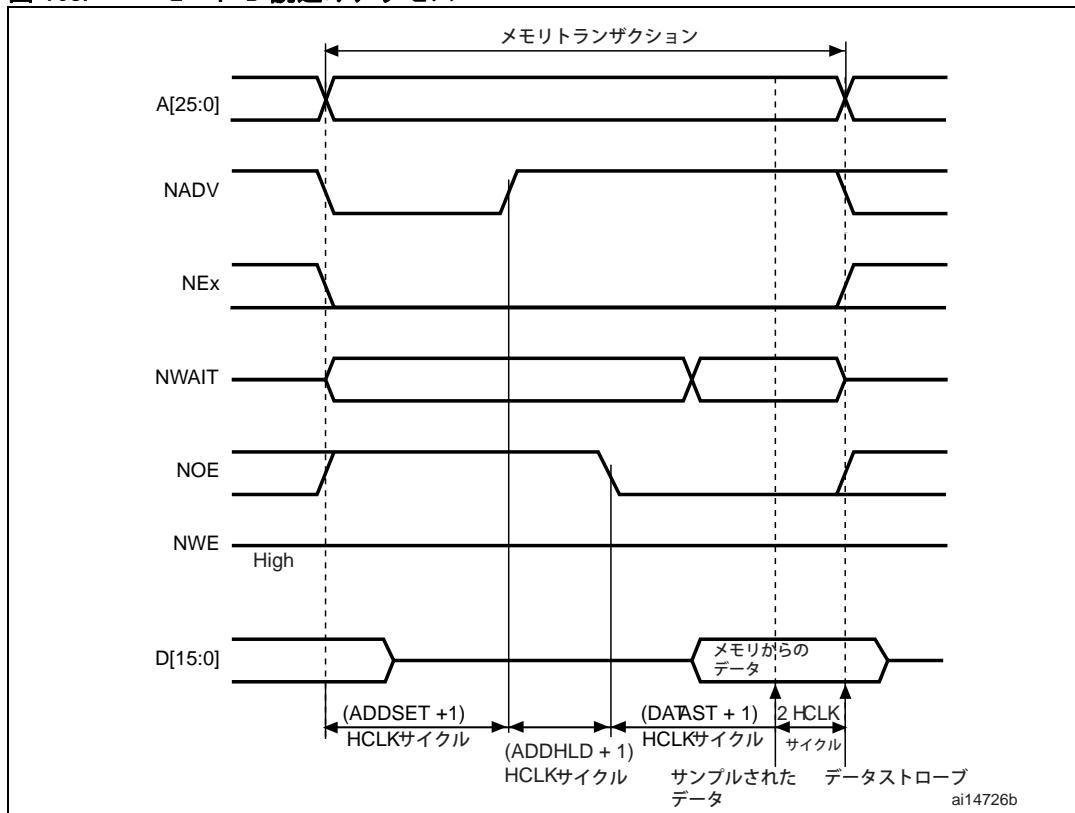
表 84. FSMC_BWTRx ピット・フィールド

ピットナンバ	名称	設定値
31-30		0x0
29-28	ACCMOD	0x2
27-16		0x000
15-8	DATAST	書き込みにおける 2 番目のアクセスフェーズ (DATAST + 1HCLK サイクル) の時間です。この値は 0 にはできません。(最低値は 1 です。)
7-4		0x0
3-0	ADDSET	書き込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK サイクル) の時間

参考資料

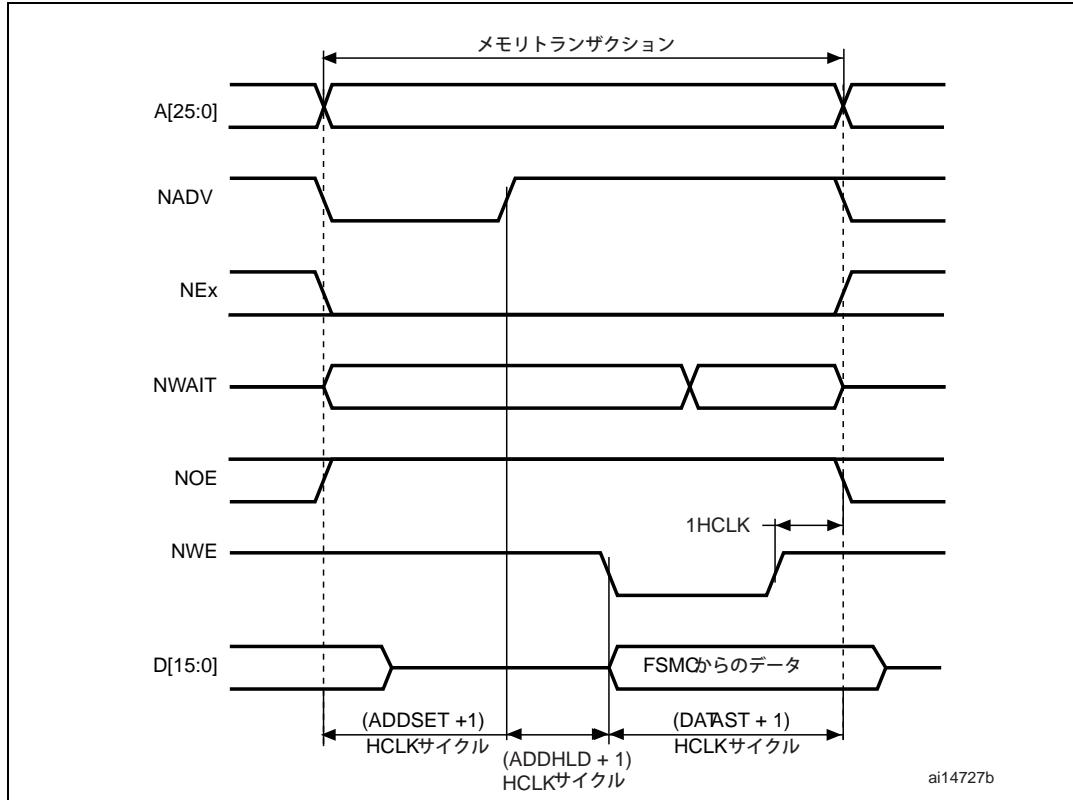
モード D - 拡張アドレスによる非同期アクセス

図 168. モード D 読込みアクセス



参考資料

図 169. モード D 書込みアクセス



モード 1 と比較した場合の相違点は、NADV のトグリング、NADV が変化した後にトグリングする NOE 及び独立した読み書きタイミングです。

表 85. FSMC_BCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-16		0x0000
15	ASYNCWAIT	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 のままでします。
14	EXTMOD	0x1
13-10		0x0
9	WAITPOL	ビット 15 が 1 の場合に限り意味を持ちます。
8	BURSTEN	0x0
7		-
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。
1	MUXEN	0x0
0	MBKEN	0x1

参考資料

表 86. FSMC_TCRx ビット・フィールド

ビットナンバ	名称	設定値
31–30		0x0
29–28	ACCMOD	0x2
27–16		0x000
15–8	DATAST	読み込みにおける 2 番目のアクセスフェーズ (DATAST + 3HCLK サイクル) の時間です。この値は 0 には設定できません。(最低値は 1 です。)
7–4	ADDHLD	読み込みにおけるアクセスの中間フェーズ (ADDHLD + 1HCLK サイクル) の時間
3–0	ADDSET	読み込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK サイクル) の時間

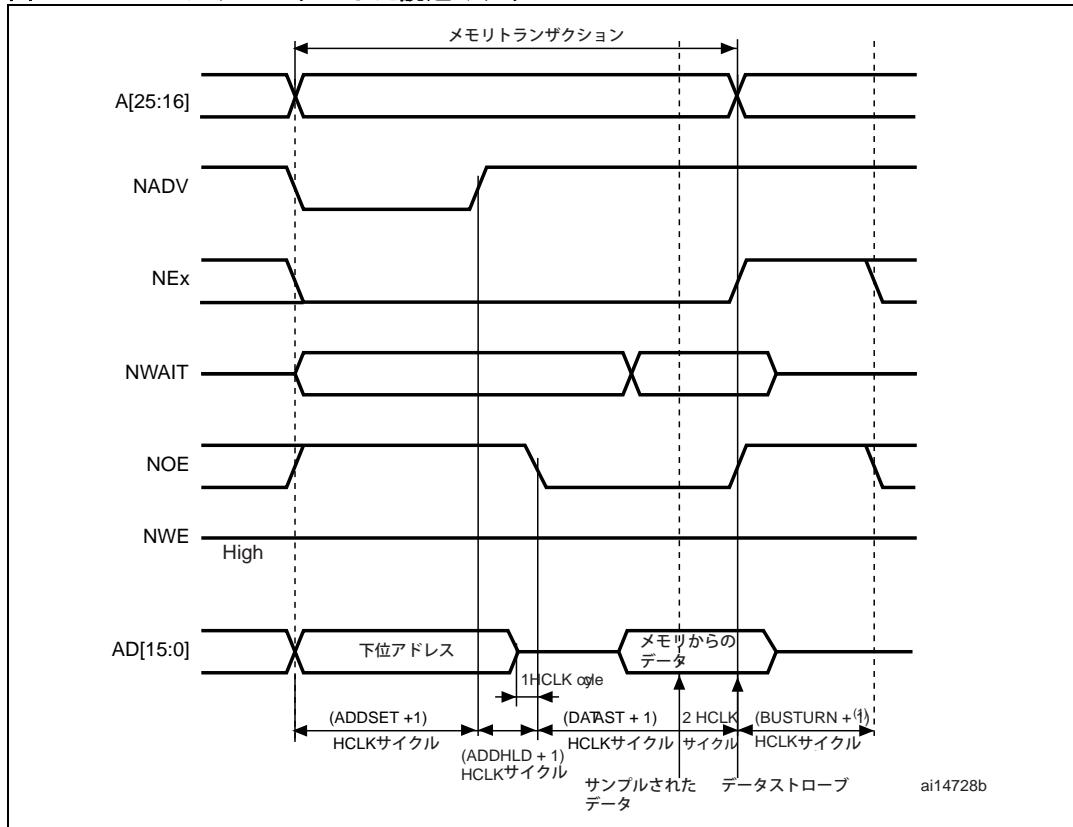
表 87. FSMC_BWTRx ビット・フィールド

ビットナンバ	名称	設定値
31–30		0x0
29–28	ACCMOD	0x2
27–16		0x000
15–8	DATAST	書き込みにおける 2 番目のアクセスフェーズ (DATAST + 1HCLK サイクル) の時間です。この値は 0 にはできません。(最低値は 1 です。)
7–4	ADDHLD	書き込みにおけるアクセスフェーズ中 (ADDHLD + 1HCLK サイクル) の時間
3–0	ADDSET	書き込みにおける最初のアクセスフェーズ (ADDSET + 1HCLK サイクル) の時間

参考資料

マルチプレクス・モード - マルチプレクスした NOR 型 Flash 非同期アクセス

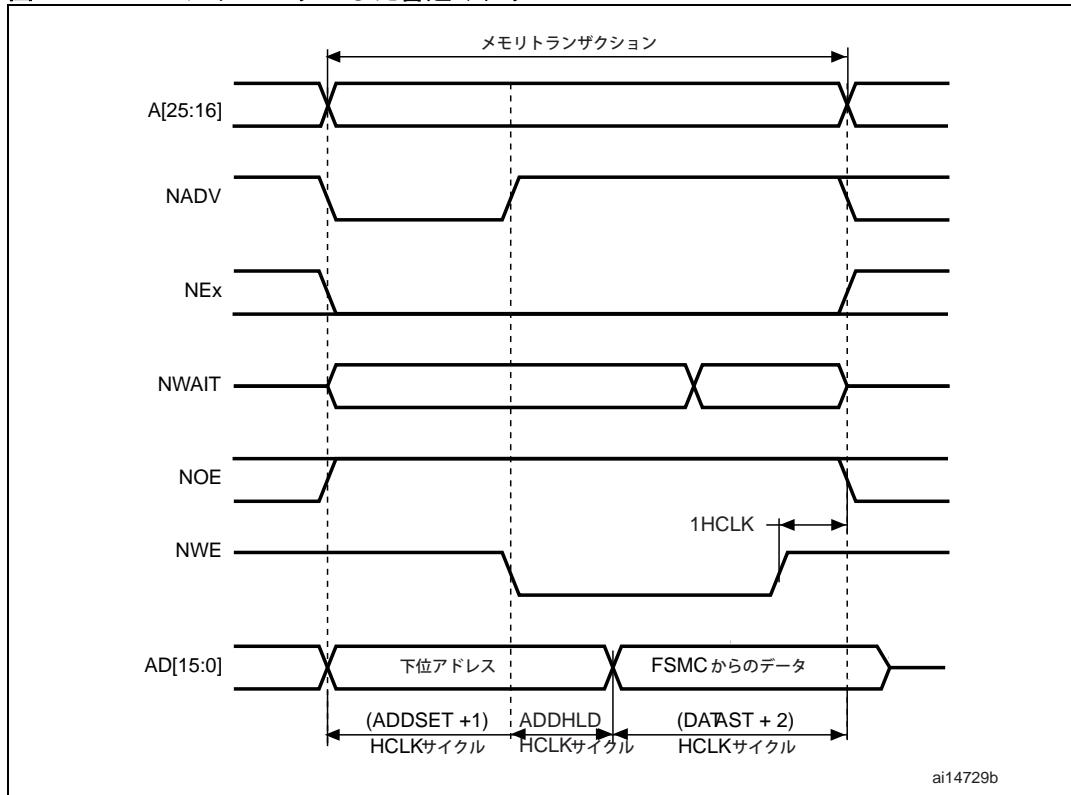
図 170. マルチプレクスした読み込みアクセス



1. バス・ターンアラウンド遅延 (BUSTURN + 1) 及びサイドバイサイド・トランザクション・オーバーラップ間の遅延、すなわち $BUSTURN \leq 5$ は影響しません。

参考資料

図 171. マルチプレクスした書き込みアクセス



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 88. FSMC_BCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-16		0x0000
15	ASYNCWAIT	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。
14	EXTMOD	0x0
13-10		0x0
9	WAITPOL	ビット 15 が 1 の場合に限り意味を持ちます。
8	BURSTEN	0x0
7		-
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	10 (NOR)
1	MUXEN	0x1
0	MBKEN	0x1

表 89. FSMC_TCRx ビット・フィールド

ビット ナンバ	名称	設定値
31-30		0
29-20		
19-16	BUSTURN	アクセスの最後のフェーズの時間 (BUSTURN+1HCLK)
15-8	DATAST	2番目のアクセスフェーズの時間 (読み込み時は DATAST + 3HCLK サイクル、書き込み時は DATAST+1HCLK サイクル) この値は 0 には設定できません。(最低値は 1 です。)
7-4	ADDHLD	アクセスの中間フェーズの時間 (ADDHLD+1HCLK サイクル) この値は 0 には設定できません。(最低値は 1 です。)
3-0	ADDSET	最初のアクセスフェーズの時間 (ADDSET+1HCLK サイクル)

18.5.5 同期バースト読み込み

メモリ・クロックである CLK は、内部クロック HCLK または CLKDIV パラメータ値により HCLK を分割したものと等しくなります。

NOR 型 Flash メモリは NADV のアサーションから CLK のハイ状態までの最小時間を指定します。この規則を満たすために FSMC は、同期アクセスの最初の内部クロック・サイクルにおいて (NADV アサーション前) クロックをメモリへ供給しません。これにより、メモリ・クロックの立上りエッジが、NADV ロウパルスの中間で発生することを保証します。

これらの例では、ウェイト・シグナルはアクティブ・ローであり、尚且つメモリによりアサートされないことを前提としています。

データ遅延と NOR 型 Flash 遅延

データ遅延は、データをサンプルする前のウェイトサイクル数です。DATLAT の値は、NOR 型 Flash 構成レジスタで指定される遅延の値と一致しなければなりません。FMSC は NADV がローである場合、データ遅延カウントにクロック・サイクルを含めません。

要注意 : 幾つかの NOR 型 Flash メモリは、データ遅延カウントに NADV ローサイクルを含めます。つまり NOR 型 Flash 遅延と FMSC DATALAT パラメータの厳密な関係は次の何れかとなり得ます。

- NOR 型 Flash 遅延 = DATLAT + 2
- NOR 型 Flash 遅延 = DATLAT + 3

CLK と HCLK の周期が同一である場合には、NADV は CLK と同時にハイとなります。出力遅延の負荷バランスが不確定であるために、CLK の立上りエッジが NADV の遷移の前後どちらで発生するかによっては、NOR 型 Flash により記録された遅延の値を壊すかもしれません。これを避けるために、NADV がハイとなって開始されたサイクル中は、メモリへのクロックの供給は行われません。(図 164 を参照)

新しいメモリの幾つかは、遅延フェーズにおいて NWAIT をアサートします。そのような場合には、DATLAT は最小値に設定することができます。その結果 FSMC はデータをサンプルし、データが有効であるかどうか判断するのに十分な時間ウェイトすることになります。このようにして FSMC は、メモリが遅延から抜け出す事を検出し、本来のデータが引き取られます。

他のメモリは、遅延中に NWAIT をアサートしません。このような場合には、FSMC とメモリに対して遅延が正しく設定されなければなりません。そうでなければ、正しいデータの

代わりに無効なデータが引き取られたり、正しいデータがメモリサイクルの初期フェーズで失われることになります。

シングルバースト転送

選択されたバンクが同期バースト・モードに設定されている場合、AHB シングルバーストランザクションがリクエストされると、FSMC は 1 つ (AHB 転送が 16bit の場合) または 2 つ (AHB 転送が 32bit、つまり 2 つの 16bit アクセスに分割される場合) のバースト読みを行います。そして最後のデータが検出された時点で、チップセレクトをディアサートします。

サイクル効率の観点から見ると、このような転送は明らかに非効率的です。(非同期読みと比較した場合) それにもかかわらず、ランダム非同期アクセスは、より時間のかかるメモリ・アクセス・モードの再プログラムを最初に必要とします。

ウェイト管理

同期バーストの NOR 型 Flash では、NWAIT はプログラムされた遅延期間、(DATALAT+1) CLK サイクル後に参照されます。

もし NWAIT がアクティブ (WAITPOL=0 の場合ローレベル、WAITPOL=1 の場合ハイレベル) であると検出されると、NWAIT がインアクティブ (WAITPOL=0 の場合ハイレベル、WAITPOL=1 の場合ローレベル) であることが検出されるまでウェイト・ステートが挿入されます。

WAIT がインアクティブである時データは、直ちに (ビット WAITCFG=1) あるいは次のクロックエッジ (ビット WAITCFG=0) で有効であると判断されます。

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のクロックパルスを送り続け、チップセレクトと出力イネーブル信号を有効に保ちます。そして、データが有効であるとは認識しません。

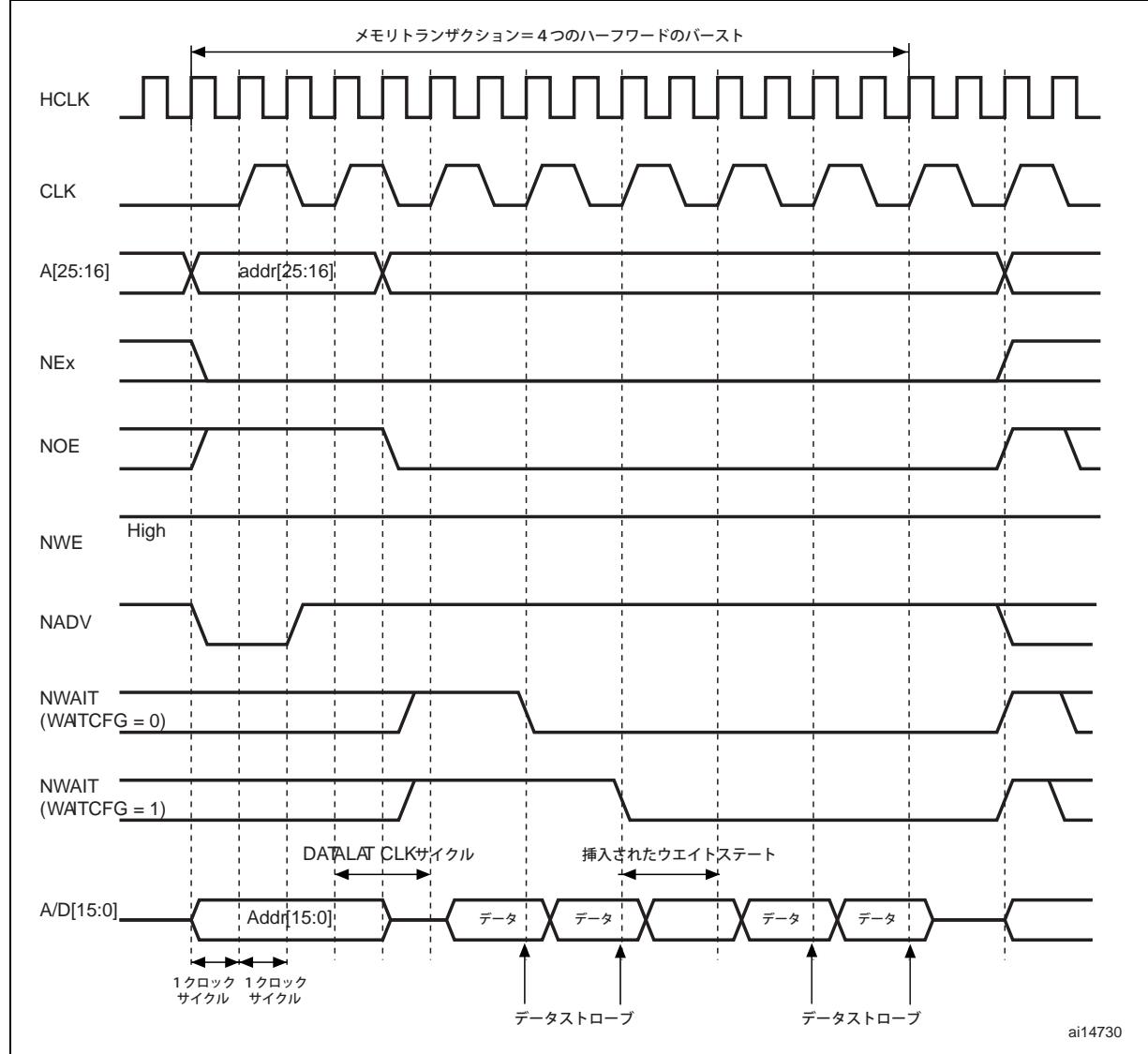
バースト・モードにおける NOR 型 Flash の NWAIT 信号には 2 つのタイミング構成があります。

- Flash メモリは、ウェイト・ステート (リセット後のデフォルト) の 1 データサイクル前に NWAIT 信号をアサートします。
- Flash メモリは、ウェイト・ステートの間 NWAIT 信号をアサートします。

FSMC は、これらの 2 つの NOR 型 Flash ウェイト・ステートの構成を、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毎にサポートします。

参考資料

図 172. 同期読み込みモード – NOR、CRAM



1. バイト・レーン出力の BL は図中に示されていません。NOR アクセス時には、それらはハイ状態に保たれ CRAM 及び COSMO RAM アクセスにはロー状態に保たれます。

参考資料

表 90. FSMC_BCRx ピット・フィールド

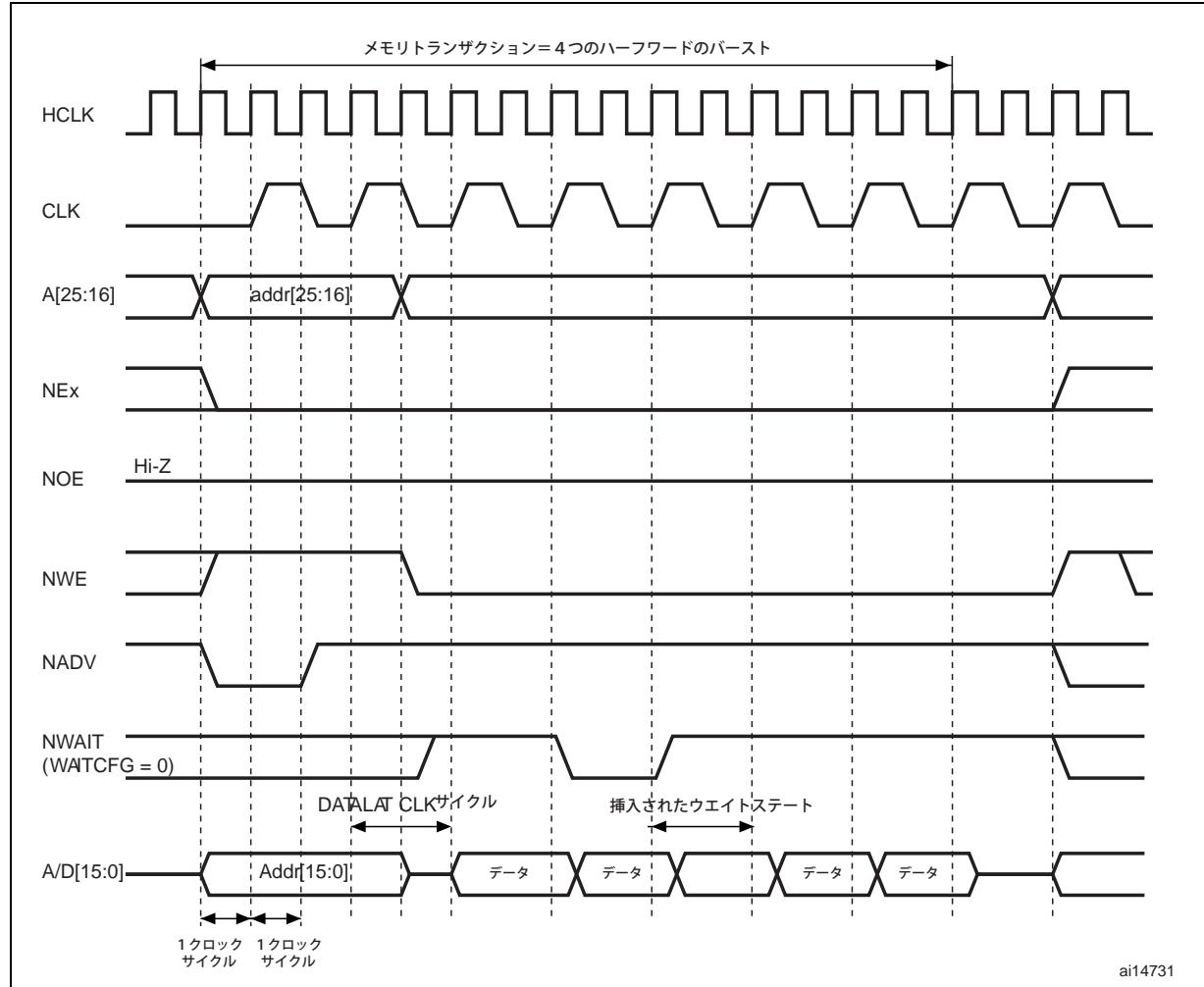
ピット ナンバ	名称	設定値
31–20		0x0000
19	CBURSTRW	同期読み込み時は影響しません。
18–16		0x0
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	ハイ状態の場合には、遅延期間後の最初のデータは、メモリ値によるウェイトに関わらず常に有効であると認識されます。
12	WREN	同期読み込み時は影響しません。
11	WAITCFG	メモリに応じて設定します。
10	WRAPMOD	メモリに応じて設定します。
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	0x1
7	FWPRLVL	意図しない書き込みアクセスからメモリを保護するために設定します。
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。
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 91. FSMC_TCRx ピット・フィールド

ピット ナンバ	名称	設定値
27–24	DATLAT	データ遅延
23–20	CLKDIV	0 の場合、CLK = HCLK となります。(サポートされません) 1 の場合、CLK = 2 × HCLK となります。
19–16	BUSTURN	影響しません。
15–8	DATAST	影響しません。
7–4	ADDHLD	影響しません。
3–0	ADDSET	影響しません。

参考資料

図 173. 同期書き込みモード – CRAM



1. メモリは、あらかじめ NWAIT を 1 サイクル発行しなければなりません、それに伴い WAITCFG を 0 にプログラムする必要があります。
2. バイト・レーン出力 (NBL) は、図中に示されていません。それらは NEx がアクティブの間、ロー状態に保たれます。

参考資料

表 92. FSMC_BCRx ピット・フィールド

ピット ナンバ	名称	設定値
31–20		0x0000
19	CBURSTRW	0x1
18–16		0x0
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	ハイ状態の場合には、遅延期間後の最初のデータは、メモリ値によるウェイトに関わらず常に有効であると認識されます。
12	WREN	同期読み込み時は影響しません。
11	WAITCFG	0x0
10	WRAPMOD	メモリに応じて設定します。
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	同期書き込み時は影響しません。
7	FWPRLVL	意図しない書き込みからメモリを保護するために設定します。
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	01 または 11
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 93. FSMC_TCRx ピット・フィールド

ピット ナンバ	名称	設定値
31–30	–	0x0
27–24	DATLAT	データ遅延
23–20	CLKDIV	0 の場合、CLK = HCLK となります。(サポートされません) 1 の場合、CLK = 2 × HCLK となります。
19–16	BUSTURN	影響しません。
15–8	DATAST	影響しません。
7–4	ADDHLD	影響しません。
3–0	ADDSET	影響しません。

参考資料

18.5.6 NOR/PSRAM 制御レジスタ

SRAM/NOR-Flash 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 30XX

このレジスタは、SRAM、ROM 及び非同期あるいはバースト NOR 型 Flash メモリの各メモリ・バンクの制御の情報を含みます。

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

Bit 19 CBURSTRW: 書込みバースト・イネーブル

Cellular RAM と COSMO では、このビットは書込み動作において同期バースト・プロトコルを有効にします。バースト・モードの Flash メモリ・アクセスでは、このビットは NWAIT 信号を使ったウェイト・ステートの挿入のイネーブルあるいはディセーブルを指定します。読み込みアクセスにおける同期バースト・プロトコルのイネーブル・ビットは、FSMC_BCRx レジスタの BURSTEN ビットです。

0: 書込み動作は、常に非同期モードで実行されます。

1: 書込み動作は、同期モードで実行されます。

Bit 15 ASYNCWAIT: 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においても NWAIT 信号を使うように FSMC を設定します。そのような場合 FSMC はそれを考慮して、アクセスがクローズされる前にディアサートされるまで待ちます。

0: 非同期プロトコル動作中（リセット後のデフォルト）には、NWAIT 信号は考慮されません。

1: 非同期プロトコル動作中に NWAIT 信号は考慮されます。

Bit 14 EXTMOD: 拡張モード・イネーブル

このビットは、FSMC が FSMC_BWTR レジスタの内部をプログラムするよう設定します。それにより、読み込み及び書き込み時の異なるタイミングを実現します。

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 レジスタ内部の値は考慮されます。

Bit 13 WAITEN: ウェイト・イネーブル・ビット

バースト・モードの Flash メモリ・アクセスには、このビットは NWAIT 信号を使ったウェイト・ステートの挿入のイネーブルあるいはディセーブルを指定します。

0: NWAIT 信号はディセーブルされます。（そのレベルは考慮されず、プログラムされた Flash 遅延期間後にウェイト・ステートは挿入されません）（リセット後のデフォルト）

1: NWAIT 信号はイネーブルされます。（アサートされた場合、プログラムされた Flash 遅延期間後にそのレベルは考慮されます）（リセット後のデフォルト）

Bit 12 WREN: 書込みイネーブル・ビット

このビットは、FSMC によるバンクへの書き込み動作がイネーブルされているかディセーブルされているかを表します。

0: FSMC によるバンクへの書き込み動作はディセーブルされています。AHB エラーは報告されます。

1: FSMC によるバンクへの書き込み動作はイネーブルされています。（リセット後のデフォルト）

参考資料

Bit 11 WAITCFG: ウェイト・タイミング構成

Flash メモリのバースト・モードアクセスでは、ウェイト・ステートが挿入されなければならない場合、NWAIT 信号は Flash メモリからのデータが有効であるかどうかを示しています。この構成ビットは、NWAIT がウェイト・ステート中あるいはウェイト・ステートが始まる 1 クロック前に Flash メモリによりアサートされるかどうかを定義します。

0: NWAIT 信号はウェイト・ステートの 1 データサイクル手前でアクティブです。

(リセット後のデフォルト)

1: NWAIT 信号は、ウェイト・ステートの間アクティブです。(Celluar RAM は除く)

Bit 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 つの連続したアクセスへ分割するか否かを定義します。Flash メモリをバースト・モードでアクセスする場合にのみ有効です。

0: ダイレクト・ラップト・バーストはイネーブルではありません。(リセット後のデフォルト)

1: ダイレクト・ラップト・バーストはイネーブルされています。

Bit 9 WAITPOL: ウェイト信号極性ビット

Flash メモリからのウェイト信号の極性を定義します。Flash メモリをバースト・モードでアクセスする場合にのみ有効です。

0: NWAIT アクティブ・ロー (リセット後のデフォルト)

1: NWAIT アクティブ・ハイ

Bit 8 BURSTEN: バースト・イネーブル・ビット

Flash メモリへのバースト・アクセス・モードをイネーブルします。同期バースト Flash メモリに限り有効です。

0: バースト・アクセス・モードはディセーブルです。(リセット後のデフォルト)

1: バースト・アクセス・モードはイネーブルです。

Bit 7 予約済み

Bit 6 FACCEN: Flash アクセス・イネーブル

NOR 型 Flash メモリ・アクセス・オペレーションをイネーブルします。

0: 対応する NOR 型 Flash メモリ・アクセスはディセーブルです。

1: 対応する NOR 型 Flash メモリ・アクセスはイネーブルです。

Bits 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。全てのタイプのメモリにたいして有効です。リセット後のバンク 1 の値は、リセット時のペリフェラル入力ピンである ExtDevWidth に依存します。

00: 8bit

01: 16bit

10: 予約、使わないでください。

11: 予約、使わないでください。

Bits 3:2 MTYP: メモリ・タイプ

対応するメモリ・バンクへ接続される外部メモリのタイプを定義します。

00: SRAM, ROM (Bank 2...4 のリセット後のデフォルト)

01: Cellular RAM

10: NOR Flash (バンク 1 のリセット後のデフォルト)

11: COSMO RAM

Bit 1 MUXEN: アドレス / データ・マルチプレクシング・イネーブル・ビット

このビットがセットされている場合、データ・バス上でアドレスとデータはマルチプレクスされます。NOR 型 Flash に限り有効です。

0: 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)

參考資料

Bit 0 **MBKEN**: メモリ・バンク・イネーブル・ビット

メモリ・バンクをイネーブルします。リセット後はバンク1はイネーブルされ、その他はディセーブルされます。ディセーブルされたバンクへのアクセスは AHB バス上で ERROR が発生します。

- 0: 対応するメモリ・バンクはディセーブルです。
1: 対応するメモリ・バンクはイネーブルです。

SRAM/NOR-Flash 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: 0xFFFF FFFF

このレジスタは SRAM、ROM 及び NOR 型 Flash メモリのための制御の情報を含みます。もし FSMC_BCRx レジスタの EXTMOD ビットがセットされると、このレジスタは書込みと読み込みアクセスに分割されます。つまり、2 つのレジスタが利用できます。1 つは読み込みアクセスを設定し（このレジスタ）、もう 1 つは書込みアクセスを設定します。
(FSMC_BWTRx レジスタ)

Bits 29:28 ACCMOD: アクセス・モード

タイミングダイアグラムに示されるように非同期アクセスについて規定します。これらのビットは、FSMC BCRxレジスタのEXTMODビットが1の時に限り有効されます。

- 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

Bits 27:24 DAT1 AT (下記のビット記述を参照) · データ遅延 (NOR 型 Flash の同期バースト用)

同期バースト・モードがイネーブルされた NOR 型 Flash の場合、最初のデータを得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。

このパラメータは HCLK 周期で表されません。非同期の NOR 型 Flash メモリ、SRAM あるいは ROM アクセスについては、この値は無関係です。CRAM の場合には、この値は 0 でなければなりません。

- 0000: 最初のバーストアクセスのための 2CLK 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のための 17CLK 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)

Bits 23:20 CI_KDIV: クロック分周比 (CIK 信号に関して)

CLK1クロック出力信号の周期を定義します。HCLKサイクル数で表します

- CLK フレーバー出力信号の周期を定義します。USER フラグが数で表します。
0000: 予約
0001: CLK 周期 = $2 \times$ HCLK 周期
0010: CLK 周期 = $3 \times$ HCLK 周期
1111: CLK 周期 = $16 \times$ HCLK 周期（リセット後のデフォルト）
非同期 NOR 型 Flash, SRAM あるいは ROM アクセスでは、この値は無関係です

参考資料

Bits 19:16 **BUSTURN:** バス・ターンアラウンド・フェーズ期間

これらのビットは、コントローラが次のサイドバイサイドトランザクションのためにデータ・バス上にアドレスを出力する必要がある場合に、バスの衝突を避ける目的で、読み込みアクセス（マルチプレクス NOR 型 Flash メモリに限り）後のバス・ターンアラウンド遅延を設定するために、ソフトウェアにより書込まれます。BUSTURN は、メモリシステムがマルチプレクスメモリを含まないか、最も遅いメモリがデータ・バスをハイ・インピーダンス状態にするために、6HCLK クロック・サイクル以上費やさない場合に、最小値に設定できます。

0000: バス・ターンアラウンド期間 = 1 × HCLK クロック・サイクル

...

1111: バス・ターンアラウンド期間 = 16 × HCLK クロック・サイクル（リセット後のデフォルト）

Bits 15:8 **DATAST:** データ・フェーズ期間

これらのビットは、データ・フェーズの期間を定義するために、（図 157 から 図 169 を参照）ソフトウェアにより書込まれます。SRAM、ROM 及び非同期マルチプレクス NOR 型 Flash アクセスで使用されます。

0000 0000: DATAST フェーズ期間 = 1 × HCLK クロック・サイクル

...

0000_1111: DATAST フェーズ期間 = 16 × HCLK クロック・サイクル（リセット後のデフォルト）

Bits 7:4 **ADDHLD:** アドレス・ホールド期間

これらのビットは、アドレス・ホールドの期間（図 166 から 図 169 を参照）を定義するためにソフトウェアにより書込まれます。SRAM、ROM 及び非同期マルチプレクス NOR 型 Flash アクセスで使用されます。

0000: ADDHLD フェーズ期間 = 1 × HCLK クロック・サイクル

...

1111: ADDHLD フェーズ期間 = 16 × HCLK クロック・サイクル（リセット後のデフォルト）

注意：同期 NOR 型 Flash アクセスでは、この値は使われません。アドレス・ホールド フェーズは、常に 1Flash クロック周期の期間です。

Bits 3:0 **ADDSET:** アドレス・セットアップ・フェーズ期間

これらのビットは、アドレス・セットアップ・フェーズの期間（図 166 から 図 169 を参照）を HCLK で定義する場合に、ソフトウェアにより書込まれます。SRAM、ROM 及び非同期マルチプレクス NOR 型 Flash で使用されます。

0000: ADDSET フェーズ期間 = 1 × HCLK クロック・サイクル

...

1111: ADDSET フェーズ期間 = 16 × HCLK クロック・サイクル（リセット後のデフォルト）

注意：同期 NOR 型 Flash アクセスでは、この値は使われません。アドレス・ホールド フェーズは、常に 1Flash クロック周期の期間です。

注意：
RAM は内部リフレッシュのために可変の遅延を持っています。そのため、これらのメモリは遅延を延長するために、必要に応じて NWAIT 信号を遅延フェーズの間常に発行します。

CRAM では、ファイルされた DATLAT は 0 に設定しなければなりません。すると FSMC は直ちに遅延フェーズを終了し、メモリからの NWAIT 信号をサンプルし始めます。そしてメモリがレディ状態となると、読み込みまたは書き込みを開始します。

この方法は、古い世代の Flash メモリと異なり NWAIT 信号を発行する最新の同期 Flash メモリでも使用することができます。（使用する Flash メモリのデータシートをご確認ください。）

参考資料

SRAM/NOR-Flash チップセレクト・タイミング・レジスタ 1..4 (FSMC_BWTR1..4)

アドレスオフセット : $0xA000\ 0000 + 0x104 + 8 * (x - 1)$, $x = 1 \dots 4$

リセット値 : 0xFFFF FFFF

このレジスタは、SRAM、ROM 及び NOR 型 Flash メモリのための各メモリ・バンクの制御の情報を含みます。FSMC_BCRx レジスタの EXTMOD ビットがセットされると、このレジスタは書き込みアクセスのためにアクティブとなります。

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	ACCMOD		DATLAT		CLKDIV		BUSTURN																									

Bits 29:28 **ACCMOD:** アクセス・モード

次のタイミングダイアグラムに示すように、非同期アクセス・モードについて規定します。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限り、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

Bits 27:24 **DATLAT:** データ遅延 (同期バースト NOR 型 Flash 用)

同期バースト・モードがイネーブルされた NOR 型 Flash については、最初のデータを得る前にメモリへ送るメモリ・クロック・サイクル数 (+2) を定義します。

0000: (0x0) 最初のバーストアクセスのための 2CLK サイクルのデータ遅延

...

1111: (0xF) 最初のバーストアクセスのための 7CLK 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)

注意：このパラメータは HCLK 周期で表されません。

注意：非同期の NOR 型 Flash メモリ、SRAM または ROM アクセスについては、この値は無関係です。

注意：CRAM の場合には、この値は 0 でなければなりません。

Bits 23:20 **CLKDIV:** クロック分周比 (CLK 信号に関して)

CLKI クロック出力信号の周期を定義します。HCLK サイクル数で表します。

0000: 予約

0001: CLK 周期 = 2 × HCLK 周期

0010: CLK 周期 = 3 × HCLK 周期

1111: CLK 周期 = 16 × HCLK 周期 (リセット後のデフォルト)

非同期 NOR 型 Flash、SRAM あるいは ROM アクセスでは、この値は無関係です。

Bits 19:16 **BUSTURN:** バス・ターンアラウンド・フェーズ期間

これらのビットは、あらゆるタイプのメモリへについて、現在のメモリ・バンクが次のサイクルでアドレスされない時の、バス・ターンアラウンド・フェーズの期間を定義する際、ソフトウェアにより書込まれます。

0000: バス・ターンアラウンド期間 = 1 × HCLK クロック・サイクル

...

1111: バス・ターンアラウンド期間 = 16 × HCLK クロック・サイクル (リセット後のデフォルト)

参考資料

Bits 15:8 DATAST: データ・フェーズ期間

これらのビットは、データ・フェーズの期間を定義するために、(図 157 から 図 169 を参照) ソフトウェアにより書き込まれます。SRAM、ROM 及び非同期マルチプレクス NOR 型 Flash アクセスで使用されます。

0000 0000: DATAST フェーズ期間 = 1 × HCLK クロック・サイクル

...

0000_1111: DATAST フェーズ期間 = 16 × HCLK クロック・サイクル (リセット後のデフォルト)

Bits 7:4 ADDHLD: アドレス・ホールド期間

これらのビットは、アドレス・ホールドの期間 (図 166 から 図 169 を参照) を定義するためにソフトウェアにより書き込まれます。SRAM、ROM 及び非同期マルチプレクス NOR 型 Flash アクセスで使用されます。

0000: ADDHLD フェーズ期間 = 1 × HCLK クロック・サイクル

...

1111: ADDHLD フェーズ期間 = 16 × HCLK クロック・サイクル (リセット後のデフォルト)

注意 : 同期 NOR 型 Flash アクセスでは、この値は使われません。アドレス・ホールド・フェーズは、常に 1Flash クロック周期の期間です。

Bits 3:0 ADDSET: アドレス・セットアップ・フェーズ期間

これらのビットは、アドレス・セットアップ・フェーズの期間 (図 166 から 図 169 を参照) を HCLK で定義する場合に、ソフトウェアにより書き込まれます。SRAM、ROM 及び非同期マルチプレクス NOR 型 Flash で使用されます。

0000: ADDSET フェーズ期間 = 1 × HCLK クロック・サイクル

...

1111: ADDSET フェーズ期間 = 16 × HCLK クロック・サイクル (リセット後のデフォルト)

注意 : 同期 NOR 型 Flash アクセスでは、この値は使われません。アドレス・ホールド・フェーズは、常に 1Flash クロック周期の期間です。

18.6 NAND 型 Flash/PC カード・コントローラ

FSMC は、次のようなタイプのデバイスを駆動するための最適なタイミングを生成します。

- NAND 型 Flash
 - 8bit
 - 16bit
- PC カード 16 互換デバイス

NAND/PC カード・コントローラは、3 つの外部バンクを制御可能です。バンク 2 及びバンク 3 は、NAND 型 Flash デバイスをサポートします。バンク 4 は PC カードデバイスをサポートします。

各バンクは独立したレジスタ ([セクション 18.6.6](#)) により設定されます。プログラマブルメモリパラメータは、アクセスタイミング ([表 94](#) に示す) と ECC 構成を含みます。

表 94. プログラマブル NAND/PC カード・アクセス・パラメータ

パラメーター	機能	アクセス・モード	単位	最小	最大
メモリ・セットアップ・タイム	コマンド・アサーション前にアドレスをセットアップするためのクロック・サイクル (HCLK) 数	読み込み / 書込み	AHB クロック・サイクル (HCLK)	1	256
メモリ・ウェイト	コマンド・アサーションの最小期間 (HCLK クロック・サイクル)	読み込み / 書込み	AHB クロック・サイクル (HCLK)	1	256
メモリ・ホールド	コマンド・ディアサーション後にアドレス（書き込みアクセスの場合にはデータも）をホールドするためのクロック・サイクル数 (HCLK)	読み込み / 書込み	AHB クロック・サイクル (HCLK)	1	255
メモリ・データ・バス・ハイ・インピーダンス	書き込みアクセス開始後にデータ・バスがハイ・インピーダンス状態に保たれるクロック・サイクル数 (HCLK)	書き込み	AHB クロック・サイクル (HCLK)	0	255

参考資料

18.6.1 外部メモリ・インタフェース信号

次の表は、NAND型Flash及びPCカードとの接続に通常使われる信号をリストしたものです。

注意：Prefix “N”は対応する信号をアクティブ・ローとして示しています。

8bit NAND型Flash

表 95. 8bit NAND型Flash

FSMC信号名	I/O	機能
A[17]	O	NAND型Flashアドレス・ラッチ・イネーブル(ALE)信号
A[16]	O	NAND型Flashコマンド・ラッチ・イネーブル(CLE)
D[7:0]	I/O	8bitマルチプレクスト、双方向アドレス/データ・バス
NOE(=NRE)	O	出力イネーブル(メモリ信号名：読み込みイネーブル、NRE)
NWE	O	書き込みイネーブル
NWAIT	I	FSMCへのNAND型Flashレディ/ビジー入力信号

FSMCは必要とされるアドレスサイクルを管理できるので、容量に関しては理論上の限界はありません。

16bit NAND型Flash

表 96. 16bit NAND型Flash

FSMC信号名	I/O	機能
A[17]	O	NAND型Flashアドレス・ラッチ・イネーブル(ALE)信号
A[16]	O	NAND型Flashコマンド・ラッチ・イネーブル(CLE)
D[15:0]	I/O	16bitマルチプレクスト、双方向アドレス/データ・バス
NOE(=NRE)	O	出力イネーブル(メモリ信号名：読み込みイネーブル、NRE)
NWE	O	書き込みイネーブル
NWAIT	I	FSMCへのNAND型Flashレディ/ビジー入力信号

FSMCは必要とされるアドレスサイクルを管理できるので、容量に関しては理論上の限界はありません。

表 97. 16bit PCカード

FSMC信号名	I/O	機能
A[10:0]	O	アドレス・バス
NIOS16	I	I/Oスペースのデータ転送幅(16bitあるいは8bit転送)
NIORD	O	I/Oスペースの出力イネーブル
NIOWR	O	I/Oスペースの書き込みイネーブル
NREG	O	コモン、アトリビュート・スペース何れのアクセスかを表すレジスタ信号
D[15:0]	I/O	双方向データ・バス
NCE4_1	O	チップセレクト1

参考資料

表 97. 16bit PC カード (続き)

FSMC 信号名	I/O	機能
NCE4_2	O	チップセレクト 2 (アクセスが 16bit か 8bit かを表す)
NOE	O	出力イネーブル
NWE	O	書き込みイネーブル
NWAIT	I	FSMC への PC カードウェイト入力信号
CD	I	PC カードの有無の検出

18.6.2 NAND 型 Flash/PC カードがサポートされるメモリ及びトランザクション

以下の表 98 は、サポートされるデバイス、アクセス及びトランザクションを示しています。NAND 型 Flash/PC カードにより許されていない(またはサポートされない)トランザクションはグレイで示されています。

表 98. サポートされるメモリ及びトランザクション

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 / 不許可	コメント
NAND 8bit	非同期	R	8	8	許可	
	非同期	W	8	8	許可	
	非同期	R	16	8	許可	2 つの FSMC アクセスへ分割
	非同期	W	16	8	許可	2 つの FSMC アクセスへ分割
	非同期	R	32	8	許可	4 つの FSMC アクセスへ分割
	非同期	W	32	8	許可	4 つの FSMC アクセスへ分割
NAND 16bit	非同期	R	8	16	許可	
	非同期	W	8	16	不許可	
	非同期	R	16	16	許可	
	非同期	W	16	16	許可	
	非同期	R	32	16	許可	2 つの FSMC アクセスへ分割
	非同期	W	32	16	許可	2 つの FSMC アクセスへ分割

18.6.3 NAND、ATA 及び PC カードに関するタイミング図

CompactFlash 及び NAND 型 Flash メモリの各バンクは、レジスタセットにより管理されます。

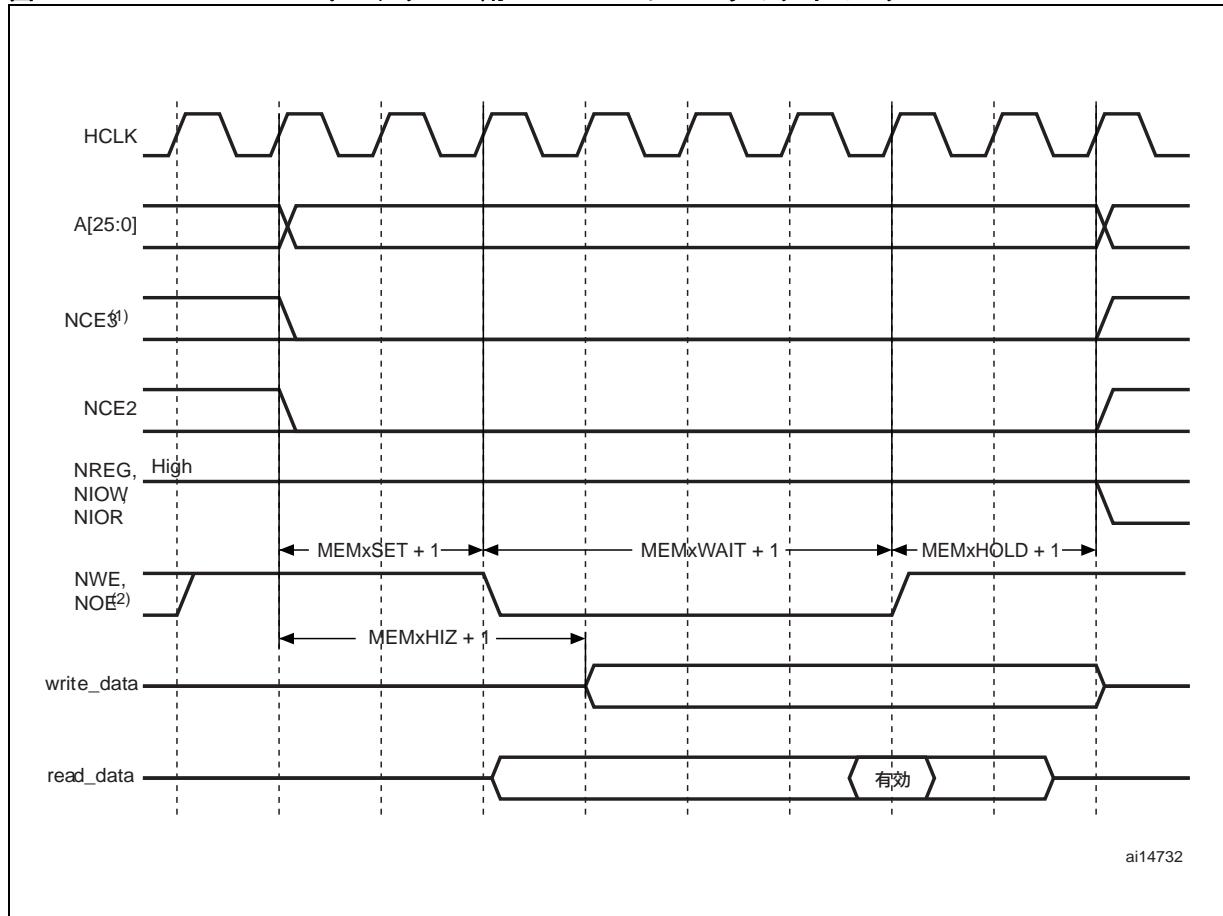
- 制御レジスタ : FSMC_PCRx
- 割込みステータス・レジスタ : FSMC_SRx
- ECC レジスタ : FSMC_ECCRx
- コモン・メモリ・スペース用タイミング・レジスタ : FSMC_PMEMx
- アトリビュート・メモリ・スペース用タイミング・レジスタ : FSMC_PATTx
- I/O スペース用タイミング・レジスタ : FSMCPIOx

各タイミング構成レジスタは、あらゆる CompactFLASH または NAND 型 Flash アクセスの 3 つのフェーズのための HCLK サイクル数を定義するために、3 つのパラメータと、書込

参考資料

みの際にデータ・バスの駆動を開始するタイミングを定義する1つのパラメータを含みます。[図174](#)は、他のタイミングが類似していることから、コモン・メモリ・アクセス用のタイミングパラメータに関する定義を示します。

図174. コモン・メモリ・アクセス用NANDコントローラのタイミング



1. NCE2 は 8bit アクセスの際、ハイ状態（インアクティブ）が保たれます。
2. NOE は書き込みアクセス時ハイ状態（インアクティブ）が保たれます。読み込み時 NWE は、ハイ状態（インアクティブ）が保たれます。

NAND 型 Flash オペレーション

これまでの説明にあるように、NAND 型 Flash デバイスのコマンド・ラッチ・イネーブル (CLE) とアドレス・ラッチ・イネーブル信号は、FSMC コントローラの幾つかのアドレス信号により駆動されます。これは NAND 型 Flash メモリヘコマンドまたはアドレスを送るには、CPU は自身のメモリ・スペース内のアドレスに書き込みを行う必要があることを意味します。

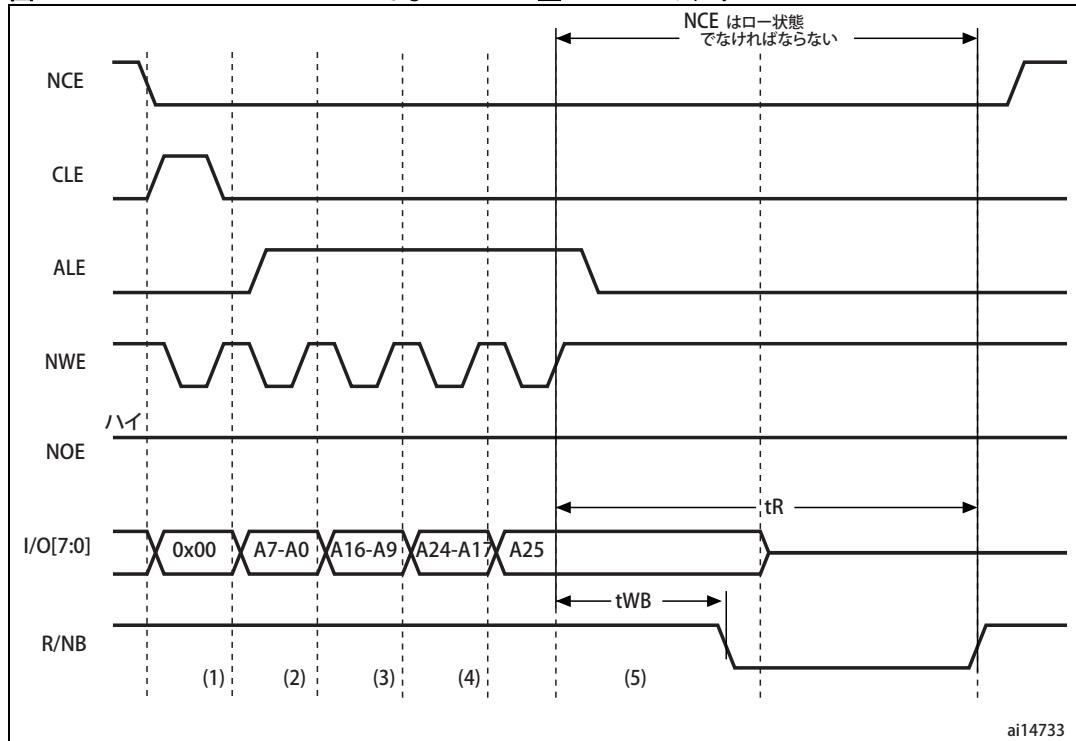
NAND 型 Flash デバイスからの一般的なページ読み込みオペレーションは、次のようにになります。

1. NAND 型 Flash の特性 (NAND 型 Flash のデータ・バス幅用に PWID ビット、PTYP=1、PWAITEN=1、PBKEN=1 タイミング構成に関してはセクション [コモン・メモリ・スペース・タイミング・レジスタ 2..4 \(FSMC_PMEM2..4\) \(428 ページ\)](#) を参照) に応じて FSMC_PCRx と FSMC_PMEMx レジスタを設定し (幾つかのデバイスでは、加えて FSMC_PATTx、[ページ 403 のセクション 18.6.4: NAND 型 Flash レディ/ビジー管理 \(424 ページ\)](#) を参照)、対応するメモリ・バンクをプログラムしイネーブルにします。
2. CPU は、1 つの Flash コマンドバイトと等しいデータバイト (例えば Samsung NAND 型 Flash デバイスでは 0x00) を、コモン・メモリ・スペースへ、バイト書き込みします。NAND 型 Flash の CLE 入力は、書き込みストローブ (NWE 上のローパルス) の間アクティブです。このようにして書き込まれたバイトは、NAND 型 Flash によりコマンドとして認識されます。コマンドは、NAND 型 Flash に一旦ラッチされると、次のページ読み込みオペレーションのために書き込む必要はありません。
3. CPU は、STARTAD[7:0]、STARTAD[16:9]、STARTAD[24:17]、最後に STARTAD[25] (64Mb ビット x8 ビット NAND 型 Flash) をコモン・メモリかアトリビュート・スペースへ 4bytes (小容量デバイスでは 3) 書き込みすることで、読み込みオペレーションのためのスタートアドレス (STARTAD) を送る事が出来ます。NAND 型 Flash の ALE 入力は、書き込みストローブ (NWE 上のローパルス) の間アクティブです。このようにして、書き込まれたバイトは読み込みオペレーションのスタートアドレスとして認識されます。制御メモリ・スペースを使う事で、FSMC の異なるタイミング構成を使うことは、幾つかの NAND 型 Flash メモリで必要とされるブリウェイト機能の実現を可能とします。(詳細は [セクション 18.6.4: NAND 型 Flash レディ/ビジー管理 \(424 ページ\)](#) を参照)
4. コントローラは、新たなアクセス (同一または異なるバンクにたいして) を開始する前にアクティブとなる準備が出来るまで (R/NB 信号のハイ状態) ウェイトします。コントローラは、ウェイトする間 NCE 信号をアクティブ (ロー状態) に保ちます。
5. CPU は、NAND 型 Flash ページ (データフィールド+スペアフィールド) をバイト単位で読み込みるためにコモン・メモリ・スペースへバイト読み込みオペレーションを行います。
6. 次の NAND 型 Flash ページは、CPU コマンドやアドレス書き込みオペレーションなしに、3 つの異なる方法で読み込む事が出来ます。
 - ステップ 5 に記述されたオペレーションを実行する。
 - ステップ 3 においてオペレーションを再スタートすることで、新たなランダムアドレスにアクセスできる。
 - ステップ 2 において再スタートする事で、NAND 型 Flash デバイスへ新たなコマンドを送る事が出来る。

18.6.4 NAND 型 Flash レディ / ビジー管理

幾つかの NAND 型 Flash デバイスでは、図 175 に示されるように、最後のアドレス入力の後及びビジーステート（R/NB ピンがロー状態）中に、NCE 信号がロー状態を保持する事が必要とされます。

図 175. ‘CE don’t care’ でない NAND 型 Flash へのアクセス



1. CPU はアドレス 0x7001 0000 にバイト 0x00 を書込む。
2. CPU はアドレス 0x7002 0000 にバイト A7 ~ A0 を書込む。
3. CPU はアドレス 0x7002 0000 にバイト A16 ~ A9 を書込む。
4. CPU はアドレス 0x7002 001100 にバイト A24 ~ A17 を書込む。
5. CPU は wrote byte A25 アドレス 0x7802 0000 にバイト A25 を書込む : FSMC は ATTHOLD \geq 7 ((7+1) \times HCLK = 112 ns $>$ t_{WB} max を供給する) FSMC_PATT2 タイミング定義を使って書き込みアクセスを実行します。これにより、R/NB がローから再びハイ状態となるまでの間、NCE がロー状態を保持する事を保証します。(NCE が Don't care でない NAND 型 Flash メモリへのリクエストのみ)

この機能が必要な場合には、T_{WB} タイミングを満たすために MEMHOLD 値をプログラムする事で、保証する事が出来ます。しかしながら、NAND 型 Flash へのいかなる CPU 読込みや書き込みアクセスも、NWE 信号の立上がりリエッジから次のアクセスへ挿入される HCLK サイクルのホールド遅延 (MEMHOLD+1) を持っています。

このタイミング制約を回避するために、t_{WB} タイミングを満たす ATTHOLD 値をタイミング・レジスタにプログラムし、MEMHOLD 値を最小とする事でアトリビュート・スペースを使う事が出来ます。そして CPU は、アトリビュート・メモリ・スペースへ書込まなければならない最後のアドレスバイトを NAND 型 Flash デバイスへ書込む場合を除き、全ての NAND 型 Flash 読込みと書き込みアクセスに関して、コモン・メモリ・スペースを使う必要があります。

全ての場合及びアクセスについて、R/NB のロー状態は FSMC コントローラを NCE ロー状態に保ちます。

18.6.5 エラー訂正コード生成 ECC (NAND 型 Flash)

FSMC の PC カード・コントローラは、2 つのエラー訂正コード処理ハードウェア・ブロックをメモリ・バンク毎に持っています。それらは、システムのソフトウェアでエラー訂正コードを処理する際に、ホスト CPU の負荷を軽減するために使われます。

これら 2 つの同一レジスタは、それぞれバンク 2 とバンク 3 に割り当てられます。結果として、バンク 4 に接続されるメモリは、ハードウェア ECC 処理を利用できません。

FSMC に搭載されるエラー訂正コード (ECC) のアルゴリズムは、NAND 型 Flash への 256、512、1024、2048、4096 または 8192bytes の読み込みまたは書き込み毎に、1bit 訂正と 2bit エラー検出を処理できます。

ECC モジュールは、NAND 型 Flash メモリ・バンクがアクティブになる度に、NAND 型 Flash データ・バス及び読み込み / 書込み信号 (NCE と NWE) を監視します。

機能的な動作は：

- NAND 型 Flash へのアクセスがバンク 2 またはバンク 3 に行われた場合、AD[15:0] バス上のデータはラッチされ ECC 処理に使われます。
- その他のアドレス上で発生する NAND 型 Flash へのアクセスは、ECC ロジックはアイドル状態であり、何の処理も行いません。すなわち、コマンドの定義や NAND 型 Flash へのアドレスのための書き込み動作は ECC 処理の対象となりません。

ホスト CPU によりリクエストされたバイト数が、一旦読み込み / 書込みされると、処理された値を取り出すためには、FSMC_ECCR2/3 レジスタが読み込まれなければなりません。

ECC ブロックを自動的にディセーブル、リイネーブルするには、これらのレジスタはクリアされなければなりません。これは、FSMC_PCR2/3 レジスタの ECCEN ビットを 1 にセットすることにより行います。

18.6.6 NAND 型 Flash/PC カード制御レジスタ

PC Card/NAND 型 Flash 制御レジスタ 2..4 (FSMC_PCR2..4)

アドレスオフセット : $0xA0000000 + 0x40 + 0x20 * (x - 1)$, $x = 2..4$

リセット値 : 0x0000 0018

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

Bits 19:17 ECCPS: ECC ページサイズ

外部 ECC のページサイズを定義します：

- 000: 256 bytes
- 001: 512 bytes
- 010: 1024 bytes
- 011: 2048 bytes
- 100: 4096 bytes
- 101: 8192 bytes

参考資料

Bits 16:13 TAR: ALE から RE 間の遅延

ALE ロー状態から RE ロー状態までの時間を AHB クロック・サイクル (HCLK) 数でセットします。

時間は $t_{ar} = (TAR + SET + 4) \times THCLK$ で、THCLK は HCLK クロック周期です。

0000: 1 HCLK サイクル (デフォルト)

1111: 16 HCLK サイクル

注意 : SET は、アドレスされるスペースに応じて MEMSET か ATTSETSET です。

Bits 12:9 TCLR: CLE から RE 間の遅延

CLE ロー状態から RE ロー状態までの時間を AHB クロック・サイクル (HCLK) 数でセットします。

時間は $t_{clr} = (TCLR + SET + 4) \times THCLK$ で、THCLK は HCLK クロック周期です。

0000: 1 HCLK サイクル (デフォルト)

1111: 16 HCLK サイクル

注意 : SET は、アドレスされるスペースに応じて MEMSET か ATTSETSET です。

Bits 8 ADLOW: アドレス・ロービット・デリバリー

PC カード /NAND 型 Flash コントローラのどちらのアドレスビットが A[24:16] 信号上へ送られるかを定義します。

0: ダイレクト・マッピング : A[24:16] は AHB アドレス・ライン [24:16] を送ります。

1: 下位アドレス・マッピング : A[24:16] は AHB アドレス・ライン [8:0] を送ります。

Bit 7 予約済み

Bits 6 ECCEN: ECC 処理ロジック・イネーブル・ビット

0: ECC ロジックはディセーブルでリセット状態です。(リセット後のデフォルト)

1: ECC ロジックはイネーブルです。

Bits 5:4 PWID: NAND 型 Flash データ・バス幅

外部 NAND 型 Flash メモリデバイス幅を定義します。PTYP が NAND 型 Flash の時に限り有効です。

00: 8bit (リセット後のデフォルト)

01: 16bit

10: 予約されています。使わないでください。

11: 予約されています。使わないでください。

Bit 3 PTYP: メモリ・タイプ

対応するメモリ・バンクに接続されるデバイスのタイプを定義します。

0: PC カード、CompactFlash、CF+ または PCMCIA

1: NAND 型 Flash (リセット後のデフォルト)

Bit 2 PBKEN: PC カード /NAND 型 Flash メモリ・バンク・イネーブル・ビット

メモリ・バンクをイネーブルします。ディセーブルされたメモリ・バンクへのアクセスは AHB バス上で ERROR を発生させます。

0: 対応するメモリ・バンクはディセーブルです。(リセット後のデフォルト)

1: 対応するメモリ・バンクはイネーブルです。

Bit 1 PWAITEN: ウェイト機能イネーブル・ビット

PC カード /NAND 型 Flash メモリに対してウェイト機能をイネーブルします。

0: ディセーブル

1: イネーブル

Bit 0 予約済み

参考資料

RM0008

フレキシブル・スタティック・メモリ・コントローラ (FSMC)

FIFO ステータス及び割込みレジスタ 2.4 (FSMC_SR2.4)

アドレスオフセット : 0xA000 0000 + 0x44 + 0x20 * (x-1), x = 2..4

リセット値 : 0x0000 0040

このレジスタは、FIFO ステータスと割込みの情報を含みます。FSMC は、16 ワードまでのデータを、AHB からメモリへ書き込み、保存する時に使う FIFO を持っています。

これは、FSMC が FIFO のデータをメモリへ送る際に、FSMC 以外のペリフェラルへのトランザクションのために、いち早く AHB へ書き込みするために使われます。このレジスタの 1 つのビットは、ECC のための FIFO のステータスを表します。

ECC は、データがメモリへ書き込まれる際に計算処理されます。つまり正しい ECC を読み込むためには、ソフトウェアは FIFO が空になるまで待たなければなりません。

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
未 定 義																				FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS					

r

r

r/w

r/w

r/w

r/w

r/w

r/w

r/w

Bit 6 FEMPT: FIFO が空の状態

FIFO のステータスを表示する読み取り専用ビット

- 0: FIFO が空でない状態
- 1: FIFO が空である状態

Bit 5 IFEN: 割込み立下りエッジ検出イネーブル・ビット

- 0: 割込み立下りエッジ検出リクエストはディセーブル状態
- 1: 割込み立下りエッジ検出リクエストはイネーブル状態

Bit 4 ILEN: 割込みレベル検出イネーブル・ビット

- 0: 割込みレベル検出リクエストはディセーブル状態
- 1: 割込みレベル検出リクエストはイネーブル状態

Bit 3 IREN: 割込み立上りエッジ検出イネーブル・ビット

- 0: 割込み立上りエッジ検出リクエストはディセーブル状態
- 1: 割込み立上りエッジ検出リクエストはイネーブル状態

Bit 2 IFS: 割込み立下りエッジステータス

- 0: 割込み立下りエッジは発生しませんでした
- 1: 割込み立下りエッジが発生しました

Bit 1 ILS: 割込みレベルステータス

- 0: 割込みレベルは発生しませんでした
- 1: 割込みレベルが発生しました

Bit 0 IRS: 割込み立上りエッジステータス

- 0: 割込み立上りエッジは発生しませんでした
- 1: 割込み立上りエッジが発生しました



参考資料

コモン・メモリ・スペース・タイミング・レジスタ 2..4 (FSMC_PMEM2..4)

アドレスオフセット : Address: 0xA000 0000 + 0x48 + 0x20 * (x - 1), x = 2..4

リセット値 : 0xFCFC FCFC

各 FSMC_PMEMx(x=2..4) 読込み / 書込みレジスタは、16bit PC カード /CompactFlash のコモン・メモリ・スペースへのアクセス、コマンド用の NAND 型 Flash へのアクセス、アドレス書込みアクセスやデータ読み込み / 書込みアクセス等の用途に、PC カードまたは NAND 型 Flash メモリ・バンク x のタイミング情報を含みます。

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
MEMHIZx								MEMHOLDx								MEMWAITx								MEMSETx							
r/w								r/w								r/w								r/w							

Bits 31:24 MEMHIZx: コモン・メモリ x データ・バス・ハイ・インピーダンス時間

ソケット x 上のコモン・メモリ・スペースへの PC カード /NAND 型 Flash による書込みアクセス後に、データ・バスをハイ・インピーダンス状態に保つ間の HCLK (NAND に限り +1) クロック・サイクル数を定義します。これは、書込みトランザクションに限り有効です。

0000 0000: (0x00) 0 HCLK サイクル

1111 1111: (0xFF) 255 HCLK サイクル (リセット後のデフォルト)

Bits 23:16 MEMHOLDx: コモン・メモリ x ホールド時間

ソケット x 上のコモン・メモリ・スペースへの PC カード /NAND 型 Flash による読み込みまたは書き込みアクセスのために、コマンドデアサーション (NWE、NOE) 後にアドレス (書き込みアクセス時はデータも) ホールドする HCLK クロック・サイクル数を定義します。

0000 0000: 予約されています。使わないでください。

0000 0001: 1 HCLK サイクルから 255 HCLK サイクル (リセット後のデフォルト)

1111 1111: (0xFF)

Bits 15:8 MEMWAITx: コモン・メモリ x ウェイト時間

ソケット x 上のコモン・メモリ・スペースへの PC カード /NAND 型 Flash による読み込みまたは書き込みアクセスのために、コマンド (NEW、NOE) をアサートするための HCLK (+1) クロック・サイクル数を定義します。プログラムされた HCLK 値の後にウェイト信号 (NWAIT) がアクティブ (ロー状態) な場合、コマンド・アサーションの期間は延長されます。

0000 0000: 1 HCLK サイクル (+ NWAIT のデアサーションによるウェイトサイクル)

1111 1111: 256 HCLK サイクル (+カードが NWAIT をデアサートする事によるウェイトサイクル) (リセット後のデフォルト)

Bits 7:0 MEMSETx: コモン・メモリ x セットアップ時間

ソケット x 上のコモン・メモリ・スペースへの PC カード /NAND 型 Flash による読み込みまたは書き込みアクセスのために、コマンド・アサーション (NWE、NOE) 前にアドレスをセットアップするための HCLK (PC カードでは +1、NAND では +2) クロック・サイクル数を定義します。

0000 0000: 1 HCLK サイクル

1111 1111: 256 HCLK サイクル (リセット後のデフォルト)

参考資料

RM0008

フレキシブル・スタティック・メモリ・コントローラ (FSMC)

アトリビュート・メモリ・スペース・タイミング・レジスタ 2..4 (FSMC_PATT2..4)

アドレスオフセット : $0xA000\ 0000 + 0x4C + 0x20 * (x - 1)$, $x = 2..4$

リセット値 : 0xFCFC FCFC

各 FSMC_PATT x ($x=2..4$) 読込み / 書込みレジスタは、16bit PC カード /CompactFlash のアトリビュート・メモリ・スペースへのアクセス、他のアクセスとタイミングが異なる場合には（読み込み / 書込みマネージメントについては [セクション 18.6.4: NAND 型 Flash レディ / ビジー管理](#)を参照）NAND 型 Flash への最後のアドレス書込みアクセスの用途に、PC カードまたは NAND 型 Flash メモリ・バンク x のタイミング情報を含みます。

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
ATTHIZ x								ATTHOLD x								ATTWAIT x								ATTSET x							
r/w								r/w								r/w								r/w							

Bits 31:24 ATTHIZ x : アトリビュート・メモリ x データ・バス・ハイ・インピーダンス時間

ソケット x 上のアトリビュート・メモリ・スペースへの PC カード /NAND 型 Flash による書込みアクセス後に、データ・バスをハイ・インピーダンス状態に保つ間の HCLK (NAND に限り +1) クロック・サイクル数を定義します。このビットは、書込みトランザクションに限り有効です。

0000 0000: 0 HCLK サイクル

1111 1111: 255 HCLK サイクル (リセット後のデフォルト)

Bits 23:16 ATTHOLD x : アトリビュート・メモリ x ホールド時間

ソケット x 上のコモン・メモリ・スペースへの PC カード /NAND 型 Flash による読み込みまたは書き込みアクセスのために、コマンド・デアサーション (NWE、NOE) 後にアドレス (書き込みアクセス時はデータも) ホールドする HCLK (+1) クロック・サイクル数を定義します。

0000 0000: 1 HCLK サイクル

1111 1111: 256 HCLK サイクル (リセット後のデフォルト)

Bits 15:8 ATTWAIT x : アトリビュート・メモリ x ウェイト時間

ソケット x 上のアトリビュート・メモリ・スペースへの PC カード /NAND 型 Flash による読み込みまたは書き込みアクセスのために、コマンド (NEW、NOE) をアサートするための HCLK (+1) クロック・サイクル数を定義します。プログラムされた HCLK 値の後にウェイト信号 (NWAIT) がアクティブ (ロー状態) な場合、コマンド・アサーションの期間は延長されます。

0000 0000: 1 HCLK サイクル (+ NWAIT のデアサーションによるウェイトサイクル)

1111 1111: 256 HCLK サイクル (+ カードが NWAIT をデアサートする事によるウェイトサイクル) (リセット後のデフォルト)

Bits 7:0 ATTSET x : アトリビュート・メモリ x セットアップ時間

ソケット x 上のアトリビュート・メモリ・スペースへの PC カード /NAND 型 Flash による読み込みまたは書き込みアクセスのために、コマンド・アサーション (NWE、NOE) 前にアドレスをセットアップするための HCLK (+1) クロック・サイクル数を定義します。

0000 0000: 1 HCLK サイクル

1111 1111: 256 HCLK サイクル (リセット後のデフォルト)



参考資料

I/O スペースタイミング・レジスタ 4 (FSMC_PIO4)

アドレスオフセット : 0xA000 0000 + 0xB0

リセット値 : 0xFCFCFCFC

FSMC_PIO4 読込み / 書込みレジスタは、16bit PC カード /CompactFlash の I/O スペースへのアクセスを得るためのタイミング情報を含みます。

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
IOHZx						IOHOLDx						IOWAITx						IOSETx													
r/w						r/w						r/w						r/w													

Bits 31:24 IOHZx: I/O データ・バス・ハイ・インピーダンス時間

ソケット x 上の I/O スペースへの PC カード書込みアクセス後に、データ・バスをハイ・インピーダンス状態に保つ間の HCLK クロック・サイクル数を定義します。このビットは、書込みトランザクションに限り有効です。

0000 0000: 0 HCLK サイクル

1111 1111: 255 HCLK サイクル (リセット後のデフォルト)

Bits 23:16 IOHOLDx: I/Ox ホールド時間

ソケット x 上の I/O スペースへの PC カードによる読み込みまたは書き込みアクセスのために、コマンドデアサーション (NWE、NOE) 後にアドレス (書き込みアクセス時はデータも) ホールドする HCLK (+ 1) クロック・サイクル数を定義します。

0000 0000: 1 HCLK サイクル

1111 1111: 256 HCLK サイクル (リセット後のデフォルト)

Bits 15:8 IOWAITx: I/Ox ウェイト時間

ソケット x 上の I/O スペースへの PC カードによる読み込みまたは書き込みアクセスのために、コマンド (SMNWE、SMNOE) をアサートするための HCLK (+ 1) クロック・サイクル数を定義します。プログラムされた HCLK 値の後にウェイト信号 (NWAIT) がアクティブ (ロー状態) な場合、コマンド・アサーションの期間は延長されます。

0000 0000: 1 HCLK サイクル (+ NWAIT のデアサーションによるウェイトサイクル)

1111 1111: 256 HCLK サイクル (+ カードが NWAIT をデアサートする事によるウェイトサイクル) (リセット後のデフォルト)

Bits 7:0 IOSETx: I/Ox セットアップ時間

ソケット x 上の I/O スペースへの PC カードによる読み込みまたは書き込みアクセスのために、コマンド・アサーション (NWE、NOE) 前にアドレスをセットアップするための HCLK (+ 1) クロック・サイクル数を定義します。

0000 0000: 1 HCLK サイクル

1111 1111: 256 HCLK サイクル (リセット後のデフォルト)

参考資料

RM0008

フレキシブル・スタティック・メモリ・コントローラ (FSMC)

ECC リザルト・レジスタ 2/3 (FSMC_ECCR2/3)

アドレスオフセット : 0xA000 0000 + 0x54 + 0x20 * (x - 1), x = 2 or 3

リセット値 : 0x0000 0000

これらのレジスタは、FSMC コントローラの ECC 処理モジュールにより計算された、現在のエラー訂正コードを含みます。(NAND 型 Flash メモリ・バンク毎に 1 つ) CPU が、NAND 型 Flash メモリページ上の、正しいアドレスからデータを読み込む場合、(セクション 18.6.5: エラー訂正コード生成 ECC (NAND 型 Flash) を参照) NAND 型 Flash から読み込みまたは書き込まれたデータは、ECC 処理モジュールにより自動的に処理されます。X バイトの読み込みが完了すると、(FSMC_PCRx レジスタの ECCPS フィールドに応じて)、CPU は処理された ECC の値を FSMC_ECCx レジスタから読みみしなければなりません。そしてページ内容が有効であるか、必要があれば訂正するために、これらの処理されたパリティデータと、スペアエリアに記録されたパリティデータが一致するか確認します。FMSMC_ECCRx レジスタは、読み込まれた後 ECCEN ビットを 0 にセットすることで、クリアされなければなりません。新たなデータクロックを処理するには、ECCEN ビットを 1 にセットしなければなりません。

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
ECCx																															

r

Bits 31:0 ECCx: ECC リザルト

これらのビットは、ECC 処理ロジックにより計算された値を提供します。表 99 に、これらのビット・フィールドの内容を示します。

表 99. ECC リザルト関連ビット

ECCPS[2:0]	ページサイズ (バイト単位)	ECC ビット
000	256	ECC[21:0]
001	512	ECC[23:0]
010	1024	ECC[25:0]
011	2048	ECC[27:0]
100	4096	ECC[29:0]
101	8192	ECC[31:0]



19 SDIO インタフェース (SDIO)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、中容量デバイスです。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、大容量デバイスです。

このセクションは、大容量デバイスのみに適応されます。

19.1 SDIO の主な機能

SD/SDIO MMC カードホストインターフェース (SDIO) は、AHB ペリフェラルバスと MultiMediaCard (MMC)、SD メモリカード、SDIO カード及び CE-ATA デバイスとのインターフェースを提供します。

MMCA テクニカル委員会から発行される MultiMediaCard システム仕様書は、MultiMediaCard Association のウェブサイト www.mmcs.org から入手できます。

SD メモリカードと SDIO カードシステム仕様書は、SD card Association のウェブサイト www.sdcards.org から入手できます。

CE-ATA システム仕様書は、CE-ATA workgroup のウェブサイト www.ce-ata.org 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 への完全準拠。3 つの異なるデータ・バス・モード用カードサポート : 1bit (デフォルト)、4bit 及び 8bit
- 以前のバージョンの MultiMediaCards との完全互換 (フォワード互換)
- SD Memory Card Specifications Version 2.0 への完全準拠
- SD I/O Card Specification Version 2.0 への完全準拠。2 つの異なるモード用カードサポート : 1bit (デフォルト) 及び 4bit
- CE-ATA 機能の完全サポート (CE-ATA digital protocol Rev1.1 への完全準拠)
- 8bit モードの 48MHz までのデータ転送
- 双方向ドライバを制御するデータとコマンド出力信号

注意： 1 SDIO は SPI 互換の通信モードを備えていません。

2 SD メモリカードプロトコルは、MultiMediaCard system specification V2.11 で定義されているように、MultiMediaCard プロトコルのサブセットです。SD メモリカードに必要ないいくつかのコマンドは、SD I/O デバイスやコンボカードの I/O 部分でサポートされません。消去等の SD I/O デバイスで使うことがない幾つかのコマンドは、SDIO でサポートされません。さらに、幾つかのコマンドは、SD メモリカードと SD I/O カードで異なる為 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 関数を利用するプロトコルを使った MMC 電気接続によりサポートされます。インターフェース電気信号定義は、MMC 基準で定められています。

MultiMediaCard/SD バスはカードをコントローラへ接続します。

現在の SDIO は 1 度に 1 つの SD/SDIO/MMC 4.2 カード、そして MMC4.1 あるいは以前のスタックだけをサポートします。

参考資料

19.2 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 バス上の基本的な処理は、コマンド / レスポンス処理です。これらの種類のバス処理は、直接コマンド内で、またはレスポンス機構により情報を転送します。さらに、幾つかの動作はデータトークンを持っています。

SD/SDIO メモリカードに対するデータ転送は、データブロックにより行われます。MMC に対するデータ転送は、データブロックまたはストリームにより行われます。CE-ATA デバイスに対するデータ転送は、データブロックにより行われます。

図 176. SDIO “no response” と “no data” 動作

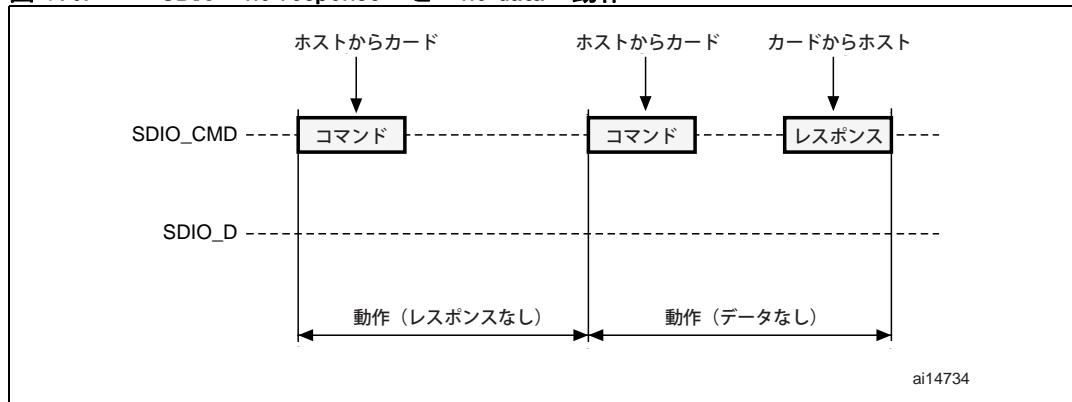
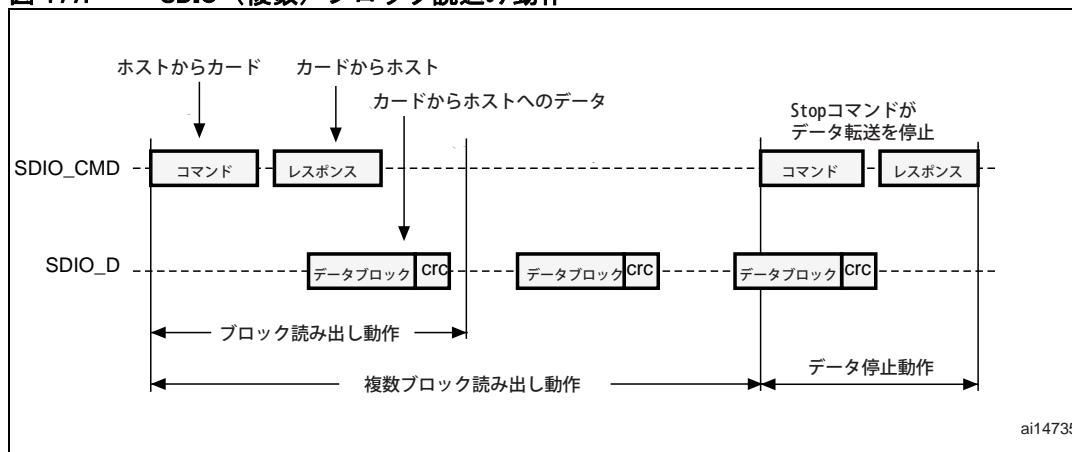
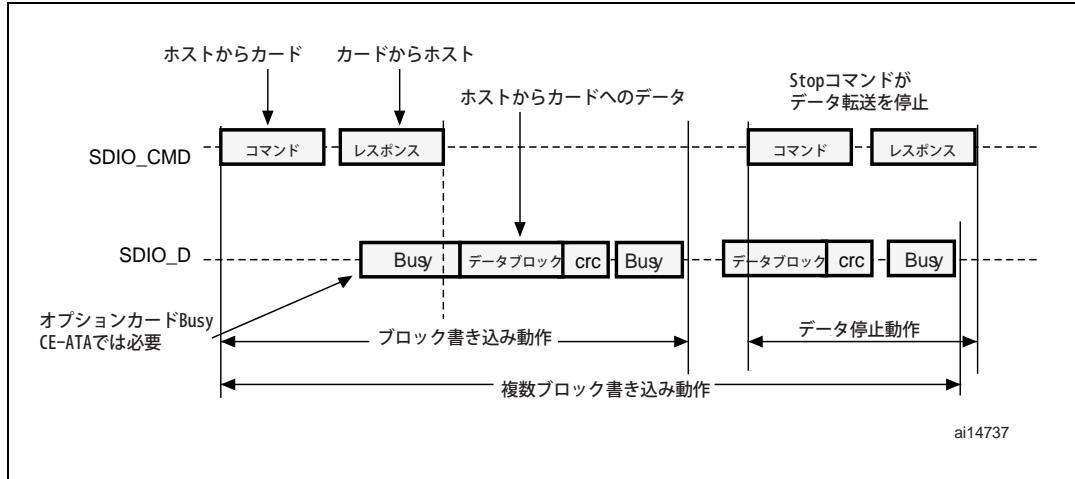


図 177. SDIO (複数) ブロック読み込み動作



参考資料

図 178. SDIO (複数) ブロック書き込み動作



注意: SDIO は Busy 信号がアサートされている限り (SDIO_D0 をロー状態へプルダウン)、データを送信しません。

図 179. SDIO 連続読み込み動作

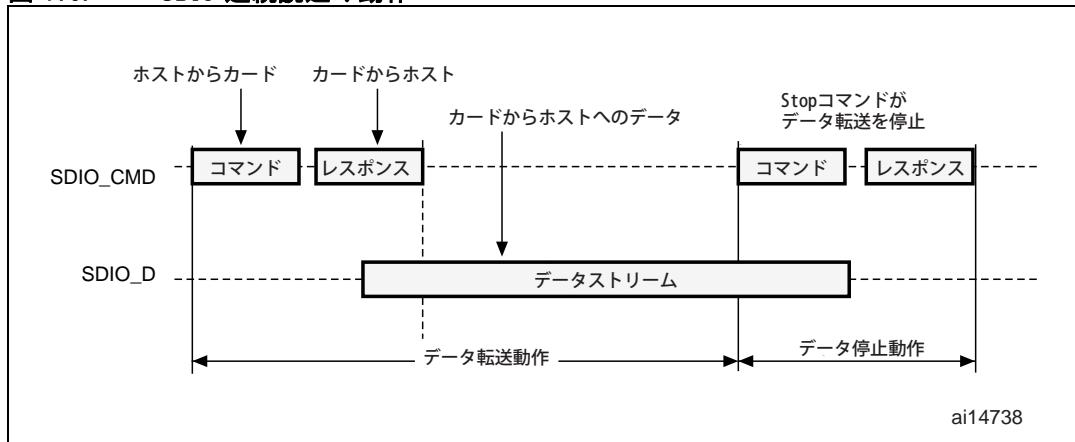
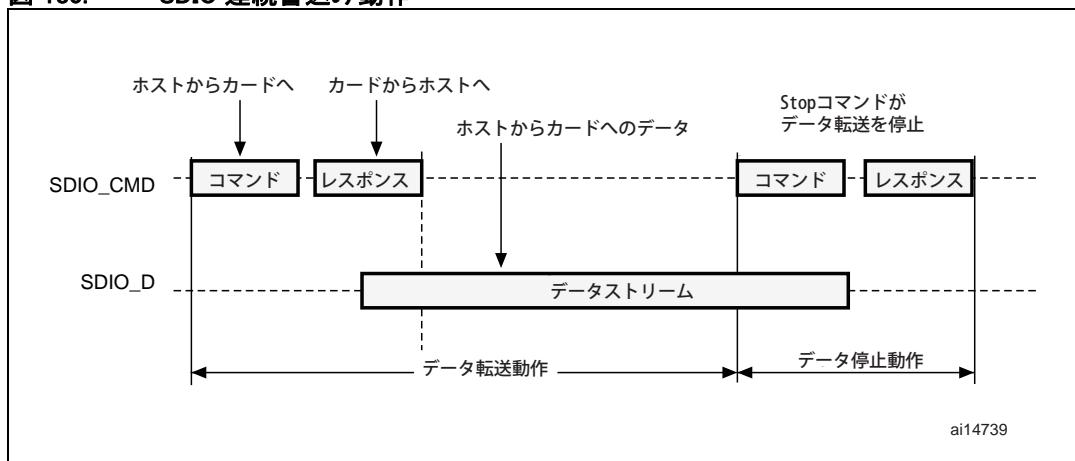


図 180. SDIO 連続書き込み動作

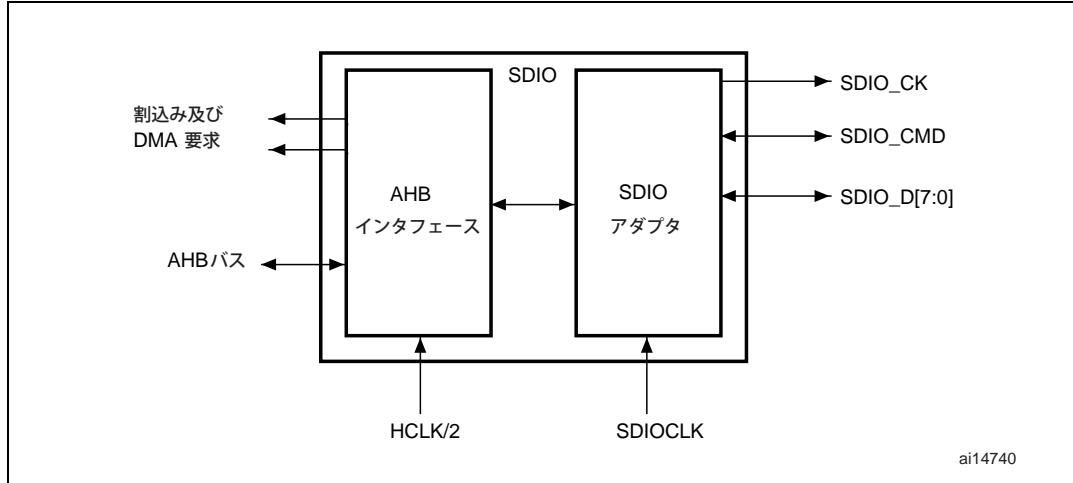


19.3 SDIO の機能詳細

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、データ等の MMC/SD/SDIO カードに関する全ての機能を提供します。
- AHB インタフェースは、SDIO アダプタレジスタへアクセスし割込みと DMA リクエスト信号を生成します。

図 181. SDIO ブロック図



ai14740

デフォルトで SDIO_D0 はデータ転送に使われます。初期化終了後、ホストはデータ・バス幅を変更することができます。

もし、MultiMediaCard がバスに接続されている場合、SDIO_D0、SDIO_D[3:0]、SDIO_D[7:0] はデータ転送に使用できます。MMCv3.31 またはそれ以前では、1bit のデータのみサポートするので、SDIO_D0 だけが使用できます。

もし SD または SD I/O カードがバスに接続されている場合、データ転送は SDIO_D0 または SDIO_D[3:0] を使うように、ホストにより設定することができます。全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。：

- 初期化用オープンドレイン (MMCv3.31 またはそれ以前)
- コマンド転送用プッシュプル (SD/SD I/O カード MMC4.2 もまた初期化にプッシュプル ドライバを使います)

SDIO_CK はカードへのクロックです：各クロック・サイクルでコマンドとデータライン上で 1bit 転送される。クロック周波数は 0MHz から 20MHz で可変 (MultiMediaCard V3.31 用)、MultiMediaCard V4.0/4.2 用で 0 から 48MHz、SD/SD I/O カードでは 0 から 25MHz の間です。

SDIO は 2 つのクロック信号を使います。：

- SDIO アダプタクロック (SDIOCLK=HCLK)
- AHB バスクロック (HCLK/2)

MultiMediaCard/SD/SD I/O カードバス上の信号は [表 100](#) に示されています。

参考資料

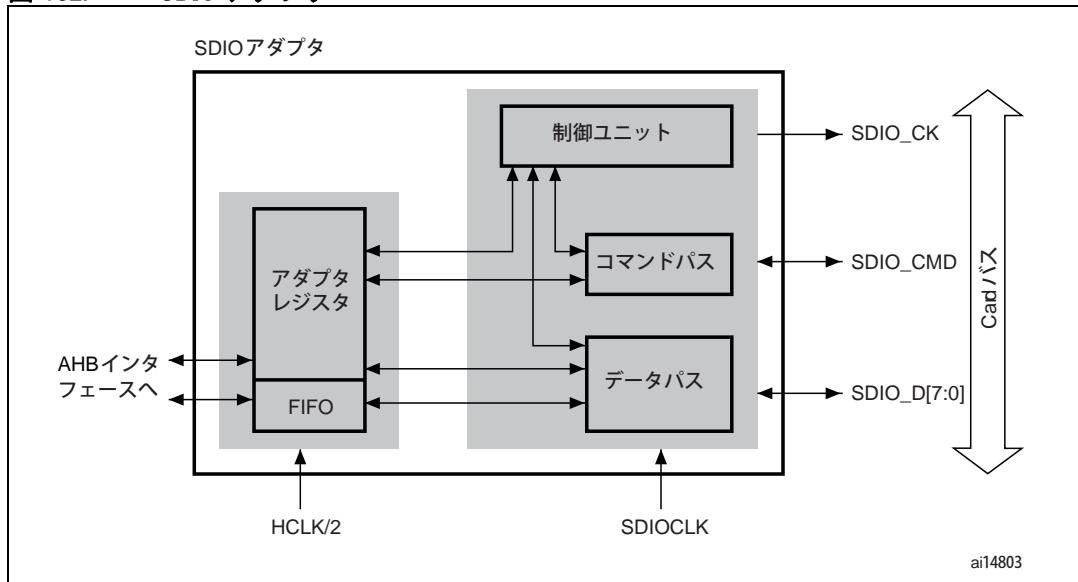
表 100. SDIO I/O 定義

ピン	方向	摘要
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のデータ・バスです。

19.3.1 SDIO アダプタ

図 182 は、SDIO アダプタの簡略化されたブロック図です。

図 182. 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・カードとの接続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・マスターです。それは次の 5 つのサブユニットから構成されています。：

- 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

注意：アダプタレジスタと FIFO は AHB バスクロック・ドメイン (HCLK/2) を使用します。制御ユニット、コマンド・バス及びデータ・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、全てのシステム・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を生成しま

す。クリア信号は、SDIO クリア・レジスタの対応するビットに 1 を書込むと生成されます。

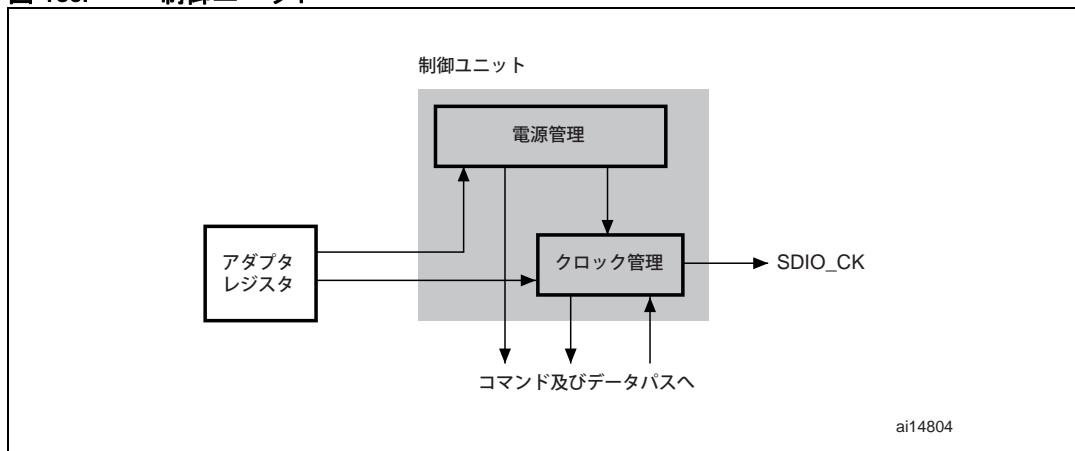
制御ユニット

制御ユニットは、パワー管理機能とメモリカード・クロックのクロック分周器を持っています。

3 つの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 183. 制御ユニット



制御ユニットは、[図 183](#) に示されています。それには、パワー管理のサブユニットとクロック管理サブユニットが含まれます。

パワー管理サブユニットは、パワー・オフとパワー・アップ・フェーズの間、カードバス出力信号をディセーブルします。

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、クロック分周かクロック・バイパス・モードのいずれかを使うことができます。クロック出力は次の場合インアクティブとなります。

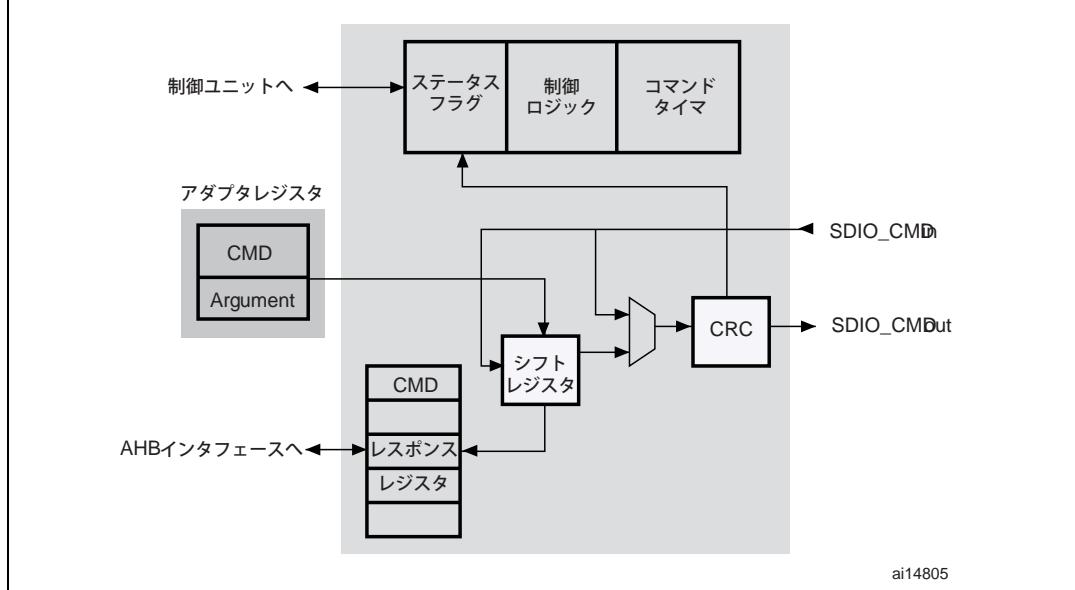
- リセット後
- パワー・オフまたはパワー・アップ・フェーズ中
- パワー節約モードがイネーブルで、カードバスがアイドル状態（コマンドをデータ・バス・サブユニットがアイドル状態となってから 8 クロック周期）

参考資料

コマンド・パス

コマンドとパスユニットは、カードに対してコマンドを送りレスポンスを受け取ります。

図 184. SDIO アダプタ・コマンド・パス

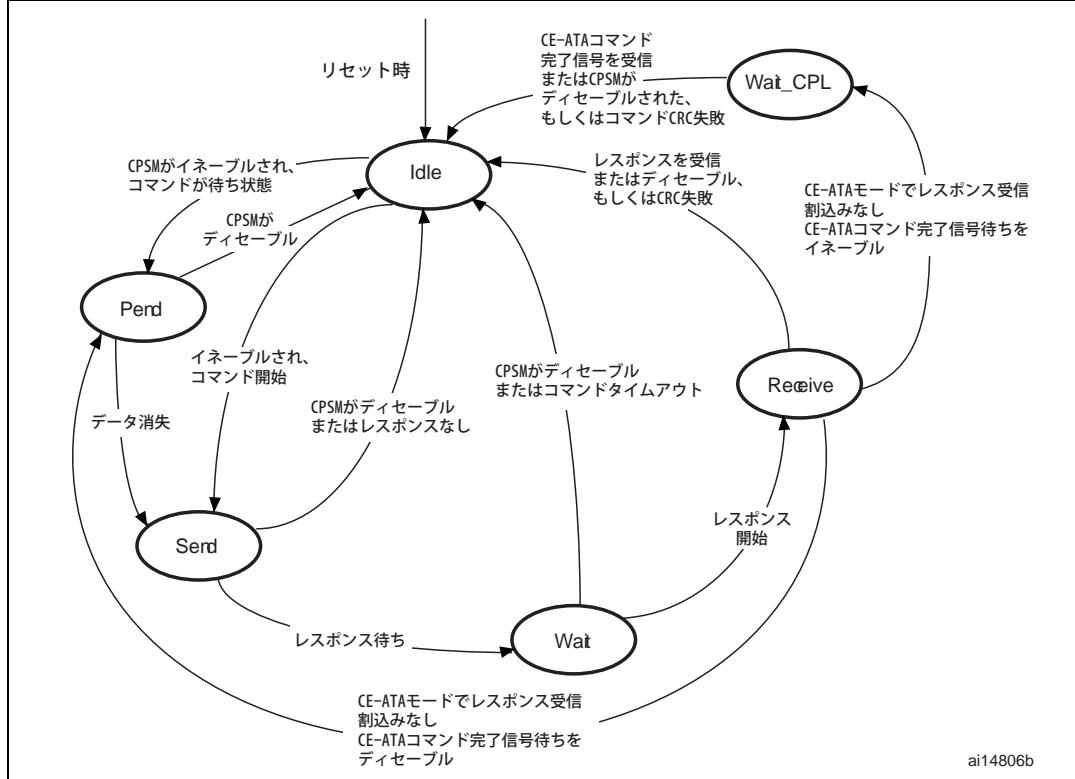


- コマンド・パス・ステート・マシン (CPSM)

- コマンド・レジスタが書き込みされイネーブル・ビットがセットされると、コマンド転送が開始します。コマンドが送られるとコマンド・パス・ステート・マシン (CPSM) はステータス・フラグをセットし、レスポンスが必要な場合はアイドルステートに入ります。もしレスポンスが必要な場合には、レスポンスを待ちます。(図 185 (439 ページ)) レスポンスが受け取られると、受信した CRC コードと内部生成されたコードを比較し、適切なステータス・フラグをセットします。

参考資料

図 185. コマンド・パス・ステート・マシン (CPSM)



ウェイト・ステートに入ると、コマンドタイマが動き出します。もし CPSM が受信状態に移る前にタイムアウトに達すると、タイムアウトフラグがセットされアイドルステートに入ります。

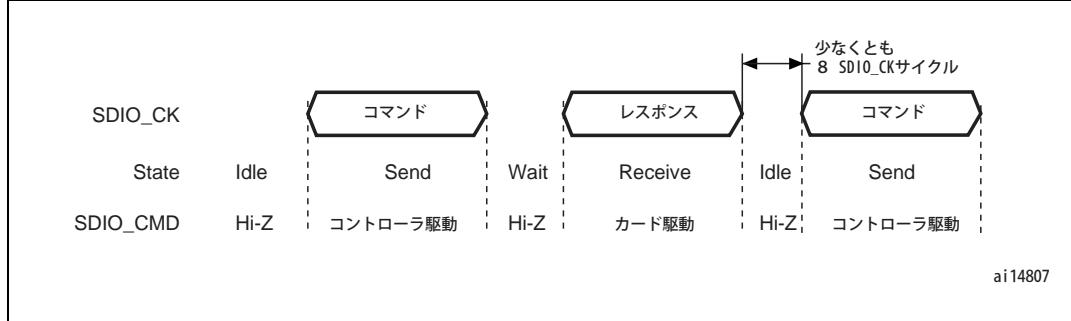
注意：コマンドタイムアウトは、 $64SDIO_CK$ クロック周期の固定された値を持っています。

もしコマンド・レジスタの割込みビットがセットされると、タイマはディセーブルされ CPSM はカードの中の 1 つからの割込みリクエストを待ちます。もしコマンド・レジスタのペンドイングビットがセットされると、CPSM はペンドイングステートに入り、データ・パスからの CmdPend 信号を待ちます。CmdPend が検出されると、CPSM は Send ステートへ移ります。これは、コマンド転送のストップをトリガするコマンドカウンタをイネーブルします。

注意：CPSM は、 N_{CC} と N_{RC} タイミング規定を満たすために、少なくとも $8SDIO_CK$ 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、 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時間です。

参考資料

図 186. 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、動作を開始するためのトークンです。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）または接続された全てのカードへ送られます。（ブロードキャストコマンドは MMCV3.31 かそれ以前で利用可能です）コマンドは、CMD ライン上でシリアルに送られます。全てのコマンドは固定の 48 ビットの長さです。MultiMediaCard、SD メモリカード及び SDIO カードのコマンドの一般的なフォーマットは [表 101](#) に示します。CE-ATA コマンドは MMC コマンド V4.2 の拡張部分であり、同じフォーマットを持っています。
コマンド・バスは半多重モードで動作するので、コマンドを送りレスポンスを受け取ることができます。もし CPSM が Send ステートにない場合、SDIO_CMD は [図 186 \(440 ページ\)](#) 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 の立上がりエッジと同期します。[表 101](#) はコマンド・フォーマットを示しています。

表 101. コマンド・フォーマット

ビット位置	幅	値	摘要
47	1	0	スタート・ビット
46	1	1	転送ビット
[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	エンド・ビット

- レスpons : レスpons は、アドレス指定されたカード（MMC V3.31 もしくはそれ以前では、全てカードから同期して）が、既に受け取ったコマンドへレスponsするために送られるトークンです。レスpons は CMD ライン上でシリアルに転送されます。

SDIO は 2 種類のレスpons をサポートします。どちらも CRC エラーチェックを使います。

- 48bit ショートレスpons
- 136bit ロングレスpons

注意： レスpons が CRC (CMD1 レスpons) を含まない場合、デバイスドライバは CRC 失敗ステータスを無視する必要があります。

参考資料

表 102. ショート・レスポンス・フォーマット

ビット位置	幅	値	摘要
47	1	0	スタート・ビット
46	1	0	転送ビット
[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	エンド・ビット

表 103. ロング・レスポンス・フォーマット

ビット位置	幅	値	摘要
135	1	0	スタート・ビット
134	1	0	転送ビット
[133:128]	6	111111	受信した
[127:1]	127	-	CID または CSD (内部 CRC7 含む)
0	1	1	エンド・ビット

コマンド・レジスタは、コマンド・インデックス（カードに送られる 6bit）とコマンドタイプを含みます。これらはコマンドがレスポンスを必要とするか否か、そしてレスポンスが 48 か 136bit 長であるかを決定します。（[セクション 19.9.4 \(475 ページ\)](#) を参照）コマンド・パスは [表 104](#) に示すようにステータス・フラグを提供します。

表 104. コマンド・パス・ステータス・フラグ

フラグ	摘要
CMDREND	CRC レスponsが OK であればセット
CCRCFAIL	レスポンス CRC が失敗すればセット
CMDSENT	コマンド（レスポンスを必要としない）が送られた時セット
CTIMEOUT	レスポンスタイムアウト
CMDACT	コマンド転送中

CRC 生成器は、CRC コード手前の全てのビットについて CRC チェックサムを計算します。これは、スタート・ビット、転送ビット、コマンド・ビットとコマンド・引数（またはカード・ステータス）を含みます。CRC チェックサムは、ロング・レスポンス・フォーマットの CID または CSD の最初の 120bit について計算されます。スタート・ビット、転送ビット及び 6 つの予約ビットは CRC の計算に使われない事に注意してください。

CRC チェックサムは 7bit の値 :

$$\text{CRC}[6:0] = \text{Remainder} [(\text{M}(x) * x^7) / \text{G}(x)]$$

$$\text{G}(x) = x^7 + x^3 + 1$$

$$\text{M}(x) = (\text{スタート・ビット}) * x^{39} + \dots + (\text{CRC 手前の最後のビット}) * x^0 \text{ または、}$$

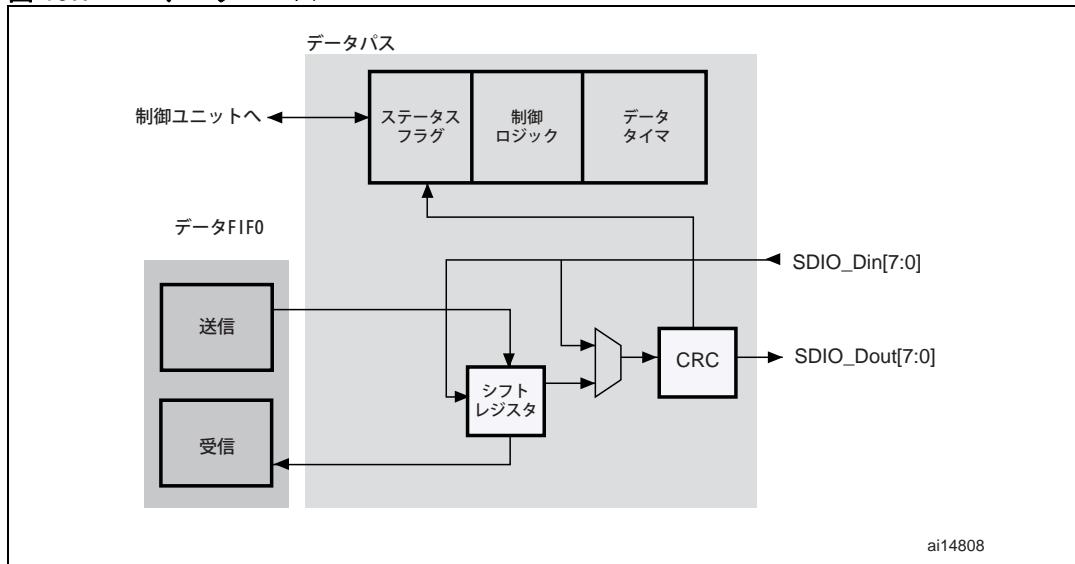
$$\text{M}(x) = (\text{スタート・ビット}) * x^{119} + \dots + (\text{CRC 手前の最後のビット}) * x^0$$

参考資料

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 187 はデータ・パスのブロック図を示しています。

図 187. データ・パス



ai14808

カード・データ・バス幅は、クロック制御レジスタでプログラムすることができます。4bit 幅バスモードがイネーブルであると、データは全 4 つのデータ信号上 (SDIO_D[3:0]) でクロック・サイクル毎に 4bits 転送されます。8bit 幅バスモードがイネーブルされると、データは全 8 つのデータ信号上 (SDIO_D[7:0]) でクロック・サイクル毎に 8bits 転送されます。ワイド・バス・モードがイネーブルでない場合、SDIO_D0 上でクロック・サイクル毎に 1bit だけが転送されます。

転送の方向によって（送信か受信）、データ・パス・ステート・マシン (DPSM) はイネーブルされた時に Wait_S または Wait_R 状態へ移行します。：

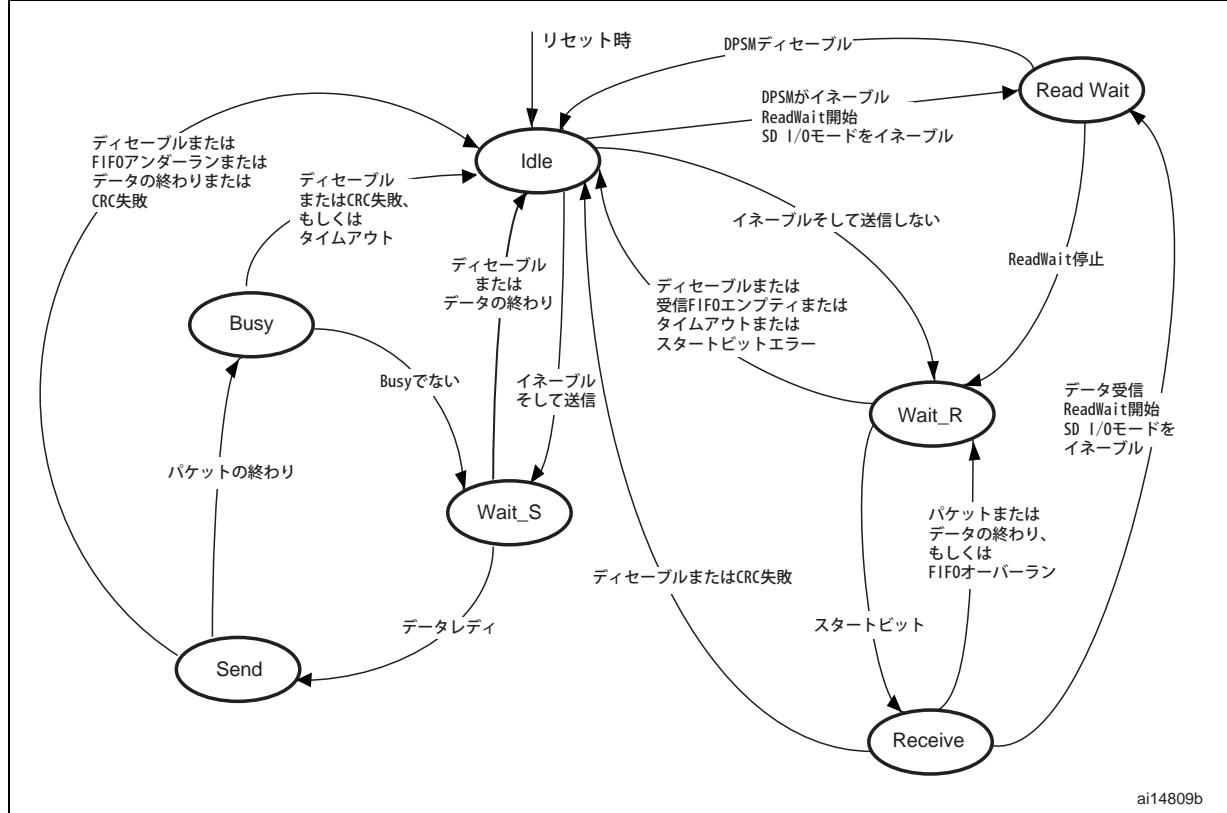
- 送信:DPSM は Wait_S 状態へ移行します。転送 FIFO にデータがある場合、DPSM は Send 状態へ移行しデータ・パス・サブユニットはカードへデータ転送を開始します。
- 受信 : DPSM は Wait_R 状態へ移行しスタート・ビットを待ちます。スタート・ビットを受信すると DPSM は受信状態へ移行し、データ・パス・サブユニットはカードからデータを受信し始めます。

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 の立上りエッジと同期しています。DPSM は、図 188: データ・パス・ステート・マシン (DPSM) に示すような 6 つの状態を持っています。

参考資料

図 188. 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、SDIO_D[7:0] 出力はハイ・インピーダンスの状態です。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・カウンタに新たな値をロードし、データの方向に応じて 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には、DPSM は SDIO_D 上のスタート・ビットを待ちます。タイムアウト前にスタート・ビットを受信すると、データ・ブロック・カウンタをロードします。スタート・ビットを検出する前にタイムアウトに達する、あるいはスタート・ビット・エラーが発生すると、アイドル状態へ移行しタイムアウト・ステータス・フラグをセットします。
- 受信：カードから受信する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またはストリームのモードとすることができます。
 - 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 コードと一致すると、DPSM は Wait_R 状態へ移行します。そうでなければ、CRC 失敗ステータス・フラグをセットし 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。カウンタがゼロに達すると、シフト・レジスタ中の残りのデータはデータ FIFO へ書き込まれ、DPSM は Wait_R 状態へ移行します。
- FIFO オーバーランが発生すると。DPSM は FIFO エラー・フラグをセットし、アイドル状態へ移行します。
- Wait_S:DPSM はデータ・カウンタがゼロになるとアイドル状態へ移行します。そうでなければ、データ FIFO 空フラグがディアサートされるまで待ちます。

参考資料

注意: DPSM は N_{WR} タイミングのリクエストを満たすため、少なくとも 2 クロック周期の間 Wait_S 状態を保ちます。このとき 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間のクロック・サイクル数です。

- **送信 :** DPSM は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、データ転送モードはブロックまたはストリームとなります。
 - 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 コードとエンド・ビットを送信し Busy 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・ビットがハイ状態でデータ・カウンタがゼロでない間、データをカードへ送信します。その後アイドル状態へ移行します。もし FIFO アンダーラン・エラーが発生すると、DPSM は FIFO エラー・フラグをセットし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 ステータスが受信されないと、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 がロー状態（カードがビジーでない）であれば 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すると、DPSM はデータタイムアウトフラグをセットしアイドル状態へ移行します。
- データタイムアウトは、DPSM が Wait_R またはビジー状態の場合にイネーブルされ、データタイムアウトを生成します。
- データ送信中に、プログラムされたタイムアウト期間以上に DPSM がビジー状態にあると、タイムアウトが発生します。
 - データ受信中に、データの最後が真でなくプログラムされたタイムアウト期間以上に DPSM が Wait_R 状態にあると、タイムアウトが発生します。
- **データ :** データはカードからホストへ転送可能です。データはデータラインにより転送されます。データは 32 ワードの FIFO へ保存されます。各ワードは 32bits 幅です。

表 105. データ・トークン・フォーマット

記述	スタート・ビット	データ	CRC16	エンド・ビット
ブロックデータ	0	-	はい	1
ストリームデータ	0	-	いいえ	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 32bit 幅、32 ワードの深さのデータ・バッファ、そして受信回路を持っています。データ FIFO は AHB クロック・ドメイン (HCLK/2) 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全ての信号は、再同期されます。

参考資料

TXACT と RXACT フラグに応じて FIFO はディセーブル、送信イネーブルまたは受信イネーブルとする事ができます。TXACT と RXACT はデータ・パス・サブユニットにより駆動され、相互排他です。

- 送信 FIFO は、TXACT がアサートされると送信回路とデータ・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 が送信用にイネーブルされているとき AHB インタフェースを通じて書込む事ができます。

送信 FIFO は 32 ビットの連続アドレスでアクセス可能です。転送 FIFO は、読み込みポインタで示されたデータワードを保持するデータ出力レジスタを持っています。データ・パス・サブユニットがシフト・レジスタをロードすると、読み込みポインタをインクリメントし新たなデータを駆動します。

もし送信 FIFO がディセーブルであると、全てのステータス・フラグはディアサートされます。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 106. 送信 FIFO ステータス・フラグ

フラグ	摘要
TXFIFOF	全 32 送信 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。
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 またはそれ以上の送信 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う事ができます。
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 TXFIFIOE フラグの反転です。
TXUNDERR	アンダーランが発生すると、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りクリアされます。

- 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、受信 FIFO は書き込みデータ・バス上にデータを出力します。書き込みポインタは、書き込み動作が完了するとインクリメントされます。読み出しでは、現在の読み込みポインタで示される FIFO ワードの内容は読み込みバス上に出力されます。もし、受信 FIFO がディセーブルであると全てのステータス・フラグはディアサートされ、読み込みと書き込みポインタはリセットされます。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 107](#)は受信 FIFO のステータス・フラグを列挙したものです。受信 FIFO は 32 の連続アドレスでアクセス可能です。

参考資料

表 107. 受信 FIFO ステータス・フラグ

フラグ	摘要
RXFIFOF	全ての 32 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 またはそれ以上の受信 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うことができます。
RXDAVL	受信 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 RXFIFOE フラグの反転です。
RXOVERR	オーバーランが発生すると、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りクリアされます。

19.3.2 SDIO AHB インタフェース

AHB インタフェースは、割込みと DMA リクエストを生成し SDIO アダプタレジスタとデータ FIFO をアクセスします。AHB インタフェースは、データ・パス、レジスタデコーダそして割込み /DMA 回路で構成されています。

SDIO 割込み

割込み回路は、選択されたステータス・フラグの少なくとも 1 つがハイ状態のとき、アサートされた割込み信号を生成します。マスク・レジスタは、割込みを生成する条件の選択を可能とします。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ると割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、SDIO ホストコントローラから MMC (CMD24 (WRITE_BLOCK) を使った 512bytes) への転送です。SDIO FIFO は DMA コントローラを使いメモリに保存されたデータで埋められます。

1. カードの識別を行う。
2. SDIO_CK 周波数を上げる。
3. CMD7 を送りカードを選択する。
4. 次の手順で DMA2 を設定する：
 - a) DMA2 コントローラをイネーブルし、待ち状態の割込みをクリアする。
 - b) DMA2_Channel4 ソース・アドレス・レジスタにメモリ・ロケーション・ベース・アドレスを、DMA2_Channel4 デスティネーション・アドレス・レジスタに SDIO FIFO レジスタ・アドレスをプログラムする。
 - c) DMA2_Channel4 制御レジスタをプログラムする。(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_Channel4 をイネーブルする。
5. 次次の手順で CMD24 (WRITE_BLOCK) を送る。：
 - a) SDIO データ長レジスタをプログラムする。(SDIO データタイマレジスタはカード識別手順の前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ロケーションをプログラムする。

- c) SDIO コマンド・レジスタをプログラムする。: CmdInde を 24 (WRITE_BLOCK) で、WaitResp を 1 で (SDIO カードホストはレスポンスを待つ)。他のフィールドはリセット値。
 - d) SDIO_STA[6] = CMDREND 割込みを待つ。割込みそして SDIO データ制御レジスタをプログラムする : DTEN を 1 で (SDIO カードホストはデータ送信するためにイネーブル)、DTDIR を 0 で (コントローラからカード)、DMAEN を 1 で (DMA イネーブル)、DTMODE を 0x09 で (512bytes)。他のフィールドは無視。
 - e) SDIO_STA[10] = DBCKEND を待つ
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、チャネルがまだイネーブルされていない事を確認。

19.4 カード機能詳細

19.4.1 カード識別モード

カード識別モードでは、ホストは全てのカードをリセット、動作電圧の実証、カードを識別しバス上のカードに相対カード・アドレス (RCA) をセットします。カード識別モードでは、全ての通信にはコマンドライン (CMD) のみが使われます。

19.4.2 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、MultiMediaCard と SD メモリをアイドル状態へ移行させます。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 の後全てのカードの出力バスドライバはハイ・インピーダンス状態にあり、カードはデフォルトの相対アドレス (RCA=0x0001) とデフォルトドライバステージレジスタ設定 (低スピード、最高駆動可能電流) で初期化されます。

19.4.3 動作電圧範囲検証

全てのカードは、仕様の範囲内のあらゆる動作電圧で SDIO カードホストと通信することができます。サポートされる最小、最大 V_{DD} 値は、カード上の動作状態レジスタ (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) とカード特定データ (CSD) を保存するカードは、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が非互換な V_{DD} 範囲を持つ場合、カードは識別サイクルを完了する事ができないので、CSD データを送る事はできません。これには、SDIO カードホストのリクエストする V_{DD} 範囲を満たさないカードを識別し、排除する特別なコマンド SEND_OP_COND (CMD1)、SD_APP_OP_COND (SD メモリ用 ACMD41 と 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、これらのコマンド動作によりリクエストされる V_{DD} 電圧範囲を送ります。指定された範囲でデータ転送を行えないカードは、バスから切断しインアクティブ状態へ移行します。

電圧をオペランドとして含まないこれらのコマンドを使うことで、SDIO カードホストは各カードへの問い合わせをし、範囲外のカードをインアクティブ状態へさせる前に、共通の電圧範囲を決定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電圧範囲を選択できる場合や、カードが使用不可能であるとの通知をユーザが必要とする場合に有効です。

19.4.4 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、識別過程は F_{od} クロック速度で開始します。SDIO_CMD ラインドライバは、オープンドレインでこの過程での平行動作を容認します。登録過程は次のように実現されます。：

1. バスがアクティベートされます。
2. SDIO カードホストは、動作条件を受信するために SEND_OP_COND (CMD1) をブロードキャストします。
3. レスポンスは、全てのカードからの動作条件をワイヤード AND した動作です。
4. 非互換な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、全てのアクティブなカードに ALL_SEND_CID (CMD2) をブロードキャストします。
6. アクティブなカードは、自身の CID 番号をシリアルに同時送信します。出力した CID ビットがコマンドライン上のビットと一致しないカードは、転送をストップし次の識別サイクルを待ちます。1つのカードが全ての CID の SDIO カードホストへの転送に成功し、識別状態へ移行します。
7. SDIO カードホストは、SET_RELATIVE_ADDR (CMD3) をそのカードに発行します。この新たなアドレスは、相対カード・アドレス (RCA) と呼ばれます。それは CID より短く、カードをアドレス指定します。アサインされたカードは、STANDBY 状態へ変化し、その後の識別過程にレスポンスすることはありません。そしてその出力は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手順 5 から 7 を繰り返します。

SD カードでは、識別過程は F_{od} クロック速度で開始し 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です。登録過程は次のように実現されます。：

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. カードは自身の動作条件レジスタの内容でレスポンスします。
4. 非互換な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、全てのアクティブなカードに ALL_SEND_CID (CMD2) をブロードキャストします。
6. カードは、自身のユニークカード識別子番号 (CID) を送り返し、識別状態へ入ります。
7. SDIO カードホストは、SET_RELATIVE_ADDR (CMD3) をアクティブなカードに発行します。この新たなアドレスは、相対カード・アドレス (RCA) と呼ばれます。それは CID より短く、カードをアドレス指定します。アサインされたカードは、STANDBY 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 RCA を変更することができます。カードの RCA は最後にアサインされた値です。
8. SDIO カードホストは、全てのアクティブなカードについて手順 5 から 7 を繰り返します。

SD I/O カードでは、登録過程は次のように実現されます。：

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) を送ります。
3. カードは自身の動作条件レジスタの内容でレスポンスします。
4. 非互換な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、SET_RELATIVE_ADDR (CMD3) をアクティブなカードに発行します。この新たなアドレスは、相対カード・アドレス (RCA) と呼ばれます。それは CID より短く、カードをアドレス指定します。アサインされたカードは、STANDBY 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 RCA を変更する事ができます。カードの RCA は最後にアサインされた値です。

19.4.5 ブロック書き込み

ブロック書き込み (CMD24-27) では、ホストにより各ブロックの後に CRC が追加されたデータが、ホストからカードへ転送されます。ブロック書き込みをサポートするカードは、WRITE_BL_LEN で定義されたデータのブロックは、常に受け入れる事ができます。もし CRC が失敗すると、カードは SDIO_D ライン上に失敗を表し転送されたデータは破棄され書きません。そして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) は無視されます。

ホストが、累積した長さがブロックのアラインメントでない不完全なブロックを使い、ブロックのミスアラインメントが許容されない場合 (CSD パラメータ WRITE_BL_MISALIGN がセットされていない)、カードはミスアラインされたブロックの最初で、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 ビットをセット) ホストが書き込み保護されたエリアに書き込みを行おうとすれば、同様に書き込み動作はアボートされます。しかしながら、この場合カードは WP_VIOLATION ビットをセットします。

CID と CSD レジスタのプログラミングは、以前のブロック長の設定を必要としません。転送されたデータは、CRC 保護されます。もし CSD や CID レジスタの一部が ROM に保存されていれば、この不变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。これが一致しないと、カードはエラーを報告しレジスタの内容を一切変更しません。幾つかのカードでは、長く予測不能な回数のデータブロックを書き込む必要があるかもしれません。データの受信と CRC 確認が終了した後、カードは書き込みバッファがフルで新たな WRITE_BLOCK コマンドによる書き込みを受け入れられない場合、書き込みを開始し SDIO_D ラインをロー状態に保ちます。ホストは、SEND_STATUS コマンド (CMD13) でカードのステータスをいつでもポーリングすることができ、カードは自身のステータスでレスポンスします。READY_FOR_DATA ステータス・ビットは、カードが新たなデータを受け入れられるか否か、または書き込み動作がまだ続行中であるかを表示します。ホストは、CMD7 (異なるカードを選ぶため) を発行し切断状態に移行させカードをディセレクトし、書き込み動作に割り込む事なしに SDIO_D ラインを開放するかもしれません。再びカードを選択するときホストは、プログラミングが続行中で書き込みバッファが利用不可であると、SDIO_D をロー状態に引き上げビジーである事を再表示します。

19.4.6 ブロック読み出し

ブロック読み出しでは、データ転送の基本部分は、最大サイズが CSD (READ_BL_LEN) で定義されたブロックです。もし READ_BL_PARTIAL がセットされると、スタートとエンドアドレスが 1 つの物理ブロック (READ_BL_LEN で定義された) に収まる小さなブロックは転送される場合があります。CRC は、各ブロックの後に追加されたデータの整合性を確かにします。CMD17 (READ_SINGLE_BLOCK) はブロック読み込みを行い、転送が終わると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、複数ブロック動作においてその種類に関係なく、読み込みをいつでも中止する事ができます。処理のアボートは、ストップ転送コマンドを送ることで行われます。

カードが複数の読み込み動作中にエラー (例えば範囲外、アドレスのミスアラインメントまたは内部エラー) を検出すると、ホストはデータ転送をストップしデータ状態に留まります。そしてホストは、ストップ転送コマンドを送り動作をアボートします。読み込みエラーはストップ転送コマンドへのレスポンスの中で報告されます。

ホストが、予め設定されたブロック数による複数ブロックの動作で、最後のブロックを転送する前にストップ転送コマンドを送った場合、カードは、もはやデータ状態にはないのと無効なコマンドとしてレスポンスします。もしホストが累積した長さがブロックのアラインメントでない不完全なブロックを使い、ブロックのミスマッチメントが許容されない場合、カードはミスマッチメント最初で、ブロックミスマッチメントエラーを検出します。(ステータス・レジスタの ADDRESS_ERROR ビットをセット)

19.4.7 ストリーム・アクセス、ストリーム書き込み、ストリーム読み込み (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、各ブロックの後に CRC は追加されません。

ストリーム書き込み (MultiMediaCard のみ)

WRITE_DAT_UNTIL_STOP (CMD20) は、指定されたアドレスから SDIO カードホストがストップコマンドを発行するまでの、SDIO カードホストからカードへのデータ転送を開始します。不完全なブロックが許容されている場合 (CSD パラメータ WRITE_BL_PARTIAL がセット)、データストリームはカード・アドレス空間内であれば、どのアドレスにおいても開始及び停止することができます。そうでなければ、ブロック境界上でのみ開始及び停止することができます。データ量を予め設定する事ができないので、CRC を使う事ができません。データ転送中にメモリ範囲の最後に達して、SD カードホストから停止コマンドが送られなかった場合、その後転送されたデータは破棄されます。

ストリーム書き込みの最大クロック周波数は、カード特定データ・レジスタの次の式で与えられます。：

$$\text{Maximumspeed} = \text{MIN}(\text{TRANSPEED}, \frac{(8 \times 2^{\text{writeblen}})(-\text{NSAC})}{\text{TAAC} \times \text{R2WFACTOR}})$$

- Maximumspeed = 書込み最大周波数
- TRANSPEED = データ転送最大速度
- writeblen = 書込みデータブロック最大長
- NSAC = CLK サイクルによるデータ読み込みアクセス時間 2
- TAAC = データ読み込みアクセス時間 1
- R2WFACTOR = 書込み速度ファクタ

もしホストがより高い周波数を使おうとした場合、カードはデータを処理できずにプログラミングを停止、ステータス・レジスタに OVERRUN エラー・ビットをセットし、それ以降のデータ転送を無視しながら停止コマンドを待つかもしれません。書き込み動作も、ホストが書き込み保護されたエリアに書き込みを行おうとするとアボートされます。しかしながらこの場合は、WP_VIOLATION ビットがセットされます。

ストリーム読み込み (MultiMediaCard のみ)

READ_DAT_UNTIL_STOP (CMD11) は、ストリーム関連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るようカードに指示します。停止コマンドは、シリアルコマンドの転送と、データ転送が停止コマンドの最後のビットの後に停止する理由から、実行遅延時間を持っています。データ送信中 SDIO カードホストにより停止コマンドが送られず、メモリ範囲の最後に達した場合、それ以降に送られたデータは不定と認識されます。

参考資料

ストリーム読み込みのクロックの最大周波数は、カード特定データ・レジスタのフィールドを使い次の式により与えられます。.

$$\text{Maximumspeed} = \text{MIN}(\text{TRANSPEED}, \frac{(8 \times 2^{\text{readblen}})(-\text{NSAC})}{\text{TAAC} \times \text{R2WFACTOR}})$$

- Maximumspeed = 読込み最大周波数
- TRANSPEED = データ転送最大速度
- readblen = 読込みデータブロック最大長
- writeblen = 書込みデータブロック最大長
- NSAC = CLK サイクルによるデータ読み込みアクセス時間 2
- TAAC = データ読み込みアクセス時間 1
- R2WFACTOR = 書込み速度ファクタ

もしホストがより高い周波数を使おうとした場合、カードはデータ転送を維持する事ができません。もしこれが発生すると。カードはステータス・レジスタに UNDRRUN エラー・ビットをセットし、転送をアボートしデータ状態で停止コマンドを待ちます。

19.4.8 消去 : グループ消去とセクタ消去

MultiMediaCard の消去可能ユニットは、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ユニットである書き込みブロック単位です。消去ブロックのサイズは、CSD に定義されるカード特定パラメータです。

ホストは連続した範囲の消去グループを消去する事ができます。消去過程は 3 ステップの順序です。

まず初めにホストは、ERASE_GROUP_START (CMD35) コマンドを使って開始アドレス範囲を定義します。次に ERASE_GROUP_END (CMD36) コマンドを使い最終アドレス範囲を定義し、最後に ERASE (CMD38) を発行し消去の過程を開始します。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以下の LSB を無視し、効果的にアドレスを消去グループ境界に合わせます。

消去コマンドが誤った順序で受信されると、カードはステータス・レジスタの ERASE_SEQ_ERROR ビットをセットし順序全体をリセットします。

誤った順序（消去コマンドでなく SEND_STATUS を除く）のコマンドを受信すると、カードはステータス・レジスタの ERASE_RESET ビットをセットし、消去順序をリセットし最後のコマンドを実行します。

もし消去範囲に書き込み保護ブロックが含まれていると、それらの部分はそのままに、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 ビットがセットされます。

カードは SDIO_D をロー状態に保持する事で、消去が進行中であることを表示します。実際の消去時間はかなり長く、ホストが CMD7 を発行しカードをディセレクトする場合があります。

19.4.9 ワイドバス選択または非選択

ワイドバス（4bit バス幅）動作は、SET_BUS_WIDTH (ACMD6) によってセレクトまたはディセレクトされます。パワー・アップ後または GO_IDLE_STATE (CMD0) が 1bit のデフォルトバス幅は、転送状態でのみ有効です。つまりバス幅は、カードが SELECT/DESELECT_CARD (CMD7) により選択された後に限り変更することができます。

19.4.10 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（カード側の責任において）
2. 機械的書き込み保護スイッチ（SDIO カードホスト側のみの責任）
3. パスワード保護されたカードロック動作

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 に永久的もしくは一時的に書き込み保護ビットをセットすることで、製造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。CSD の WP_GRP_ENABLE ビットをセットすることで、書き込み保護をセクタのグループでサポートするカードは、データの一部を保護することができます。そして、書き込み保護はアプリケーションにより変更する事が可能です。書き込み保護は、CSD に示されるように WP_GRP_SIZE セクタの単位です。SET_WRITE_PROT と 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 は 1 ブロックの読み込みコマンドと似ています。カードは、32 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ります。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、グループサイズより下の全ての 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ディングタブは、カードの書き込み保護をユーザがセット／クリアできるようにします。スライディングタブがウインドウオープンの位置にあると、カードは書き込み保護されます。そしてウインドウが閉じると、カードの内容は変更できます。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事を SDIO カードホストへ表示します。SDIO カードホストは、カードを保護する責任があるわけです。書き込み保護スイッチの位置は、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は、SDIO カードホストモジュールがカードをパスワードでロック、アンロックすることをイネーブルします。パスワードは 128bit PWD レジスタへ保存され、そのサイズは 8bit PWD_LEN レジスタへセットされます。これらのレジスタは不揮発性ですので、パワーサイクルで消去されません。ロックされたカードは、決まったコマンドにレスポンスし実行します。これは SDIO カードホストモジュールがリセット、初期化、選択そしてステータスの問い合わせを行うことを確認することを意味します。しかしながら、カード上のデータへのアクセスは許されません。パスワードがセットされると (PWD_LEN の値がゼロ以外を表す)、カードはパワー・アップ後に自動的にロックされます。CSD と CID レジスタのように、書き込みコマンド、ロック / アンロックコマンドは転送状態でのみ利用できます。この状態では、コマンドはアドレス引数を含みません。そしてカードは使う前の選択され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バス処理タイプと構造を持っています。転送されたデータブロックは、コマンドに必要な全ての情報（パスワード設定モード、PWD そのもの及びカードロック / アンロック）を含んでいます。コマンド・データブロック・サイズは、カードにロック / アンロックコマンドを送る前に、SDIO カード・ホスト・モジュールにより定義されます。それは [表 121](#) のような構造を持っています。

ビット設定は、次のように行います：

- ERASE : セットすると消去動作を強制します。他の全てのビットは、ゼロでなければなりません。そしてコマンドバイトだけが送られます。
- 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 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 : パスワードの長さをバイト単位で定義します。
- PWD : パスワード（コマンドに応じて、新しい又は現在使用中の）

次のセクションは、パスワードをセット / リセット、カードをロック / アンロック、そして消去を強制するコマンド順序を説明します。

パスワードの設定

1. 既に選択されていなければ、カードを選択します。（SELECT/DESELECT_CARD、CMD7）
2. 8bit カードロック / アンロックモード、8bit PWD_LEN、新たなパスワードのバイト数を送るためのブロック長（SET_BLOCKLEN, CMD16）を定義します。パスワード書き換えが行われるときは、古いものと新しいパスワードの両方がコマンドで送られる事を考慮したブロックサイズでなければなりません。
3. データライン上の 16bit CRC を含む適切なデータブロックサイズと共に LOCK/UNLOCK (CMD42) を送ります。データブロックはモード（SET_PWD=1）、長さ（PWD_LEN）とパスワード（PWD）自身を表示します。パスワードの書き換えが完了すると、長さの値（PWD_LEN）は古いものと新規の両方のパスワードの長さを含みます。PWD フィールドは、古いパスワード（現在使用中）に続き新しいパスワードを含みます。
4. パスワードが一致すると、新たなパスワードとそのサイズが PWD と PWD_LEN にそれぞれ保存されます。送られた古いパスワードが期待するパスワードと一致しないと（サイズまたは内容が）、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、現在パスワードがセットされているかどうかを表します。これらのビットがゼロ以外である場合、パスワードがセットされておりカードはパワー・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カードをロックする為の追加のコマンドを送れば、現在のパワーセッションにおいて直ちにカードをロックする事が可能です。

パスワードのリセット

1. 既に選択されていなければ、カードを選択します。（SELECT/DESELECT_CARD、CMD7）
2. データライン上の 16bit CRC を含む適切なデータブロックサイズと共に LOCK/UNLOCK (CMD42) を送ります。データブロックはモード（CLR_PWD=1）、長さ（PWD_LEN）とパスワード（PWD）自身を表示します。LOCK_UNLOCK ビットは無視されます。
3. データライン上の 16bit CRC を含む適切なデータブロックサイズと共に LOCK/UNLOCK (CMD42) を送ります。データブロックはモード（CLR_PWD=1）、長さ（PWD_LEN）とパスワード（PWD）自身を表示します。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。送られたパスワードが期待するパスワードと一致しないと（サイズまたは内容が）、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。

カードのロック

- 既に選択されていなければ、カードを選択します。(SELECT/DESELECT_CARD、CMD7)
- 8bit カードロック / アンロックモード ([表 121](#) のバイト 0)、8bit PWD_LEN、現在のパスワードのバイト数を送るためのブロック長(SET_BLOCKLEN、CMD16)を定義します。
- データライン上の 16bit CRC を含む適切なデータブロックサイズと共に LOCK/UNLOCK (CMD42) を送ります。データブロックはモード (LOCK_UNLOCK=1)、長さ (PWD_LEN) とパスワード (PWD) 自身を表示します。
- 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。送られたパスワードが期待するパスワードと一致しないと (サイズまたは内容が)、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、ロックは失敗します。

同じ順序の中で、パスワードのセットとカードのロックを行うことは可能です。この場合、SDIO カードホストモジュールはパスワードのセットに必要な全てのステップを実行します。[\(パスワードの設定 \(453 ページ\) を参照\)](#) しかしながら、ステップ 3 で新たなパスワードコマンドが送られる際に、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セットされていると (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再度ロックしようとしたり、パスワードを持たないカードをロックしようとすると、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ます。

カードのアンロック

- 既に選択されていなければ、カードを選択します。(SELECT/DESELECT_CARD、CMD7)
- 8bit カードロック / アンロックモード ([表 121](#) のバイト 0)、8bit PWD_LEN、現在のパスワードのバイト数を送るためのブロック長(SET_BLOCKLEN、CMD16)を定義します。
- データライン上の 16bit CRC を含む適切なデータブロックサイズと共に LOCK/UNLOCK (CMD42) を送ります。データブロックはモード (LOCK_UNLOCK = 0)、長さ (PWD_LEN) とパスワード (PWD) 自身を表示します。
- 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 ビットがクリアされます。送られたパスワードが期待するパスワードと一致しないと (サイズまたは内容が)、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、カードはロックされたままとなります。

アンロッキング機能は、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でない場合、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再度アンロックしようとすると、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ます。

強制消去

もしユーザがパスワード (PWD の内容) を忘れてしまったら、カード上の全データをクリアした後アクセスすることができます。この強制消去動作は、全てのカード・データ及び全てのパスワードデータを消去します。

- 既に選択されていなければ、カードを選択します。(SELECT/DESELECT_CARD、CMD7)
- ブロック長(SET_BLOCKLEN, CMD16) を 1byte にセットします。8bit カードロック / アンロックバイト ([表 121](#) のバイト 0) だけが送られます。

3. データライン上の 16bit CRC を含む適切なデータブロックサイズと共に LOCK/UNLOCK (CMD42) を送ります。データブロックはモード (ERASE = 1) を表示します。他の全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唯一セットされたビットであるならば、PWD と PWD_LEN フィールドを含む全てのカード内容は消去されます。そしてカードはロックされません。他のあらゆるビットがセットされていると、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、カードは全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とすると、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ます。

19.4.11 カード・ステータス・レジスタ

レスポンス・フォーマット R1 は、カード・ステータスと呼ばれる 32bit フィールドを持っています。これらのビットはカード・ステータス情報 (ローカル・ステータス・レジスタへ保存されるかもしれない) をホストへ転送するためのものです。断りがない限り、ステータス・エントリは、以前に発行されたコマンドに常に応じます。

表 108 はステータスの異なるエントリを定義します。テーブル中のタイプとクリア条件 フィールドは、次のように省略されます。：

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、セットされます。
- X : コマンド実行中に検出、セットされます。SDIO カードホストは、これらのビットを読み込むためのステータスコマンドでカードを監視し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 : 常に以前のコマンド関連しています。有効なコマンドの受信でクリアします。
(1 コマンドの遅延で)
- C : 読込みでクリアします。

参考資料

表 108. カード・ステータス

ビット	識別子	タイプ	値	摘要	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は、このカードに許容された範囲を超えてています。 複数ブロックまたはストリーム読み込み / 書込み動作が、(有効なアドレスから開始しているが) カード容量を超えた読み込みか書き込みを行おうとしています。	C
30	ADDRESS_MISALIGN		'0' = エラーなし '1' = エラー	コマンドアドレス引数が (現在セットされているブロック長と一致して) 最初のデータブロックをカードの物理ブロックに対してミスマッチした場所に置いています。 複数ブロック読み込み / 書込み動作が、(有効なアドレスから開始しているが) カードの物理ブロックに合っていないデータブロックの読み込みか書き込みを行おうとしています。	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で無効です。(例、ホストが書き込みコマンドを発行、現在のブロック長がカードの許容する最大値より小さい、そして部分的なブロックの書き込みが許されない)	C
28	ERASE_SEQ_ERROR		'0' = エラーなし '1' = エラー	消去コマンドの過程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	消去のために消去グループの無効な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へプログラムしようとしました。	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りロックされたことを表します。	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、過程エラーまたはパスワードエラーが発生した時にセットされます。	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	カード状態には、そのコマンドは合法ではありません。	B
21	CARD_ECC_FAILED	E X	'0' = 成功 '1' = 失敗	カードの内部 ECC が適応されました が、訂正に失敗しました。	C

参考資料

表 108. カード・ステータス (続き)

ビット	識別子	タイプ	値	摘要	クリア条件
20	CC_ERROR	E R	'0' = エラーなし '1' = エラー	(規格では定義されていません) ホストコマンドに関係し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	(規格では定義されていません) 最後のホストコマンド (実行に関係した (実行中に検出した) 一般カードエラーです。 (例、読み込みまたは書き込みの失敗)	C
18	予約済み				
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あり得る： - CID レジスタは既に書き込まれており上書きできません。 - CSD の読み出し専用部分が、カードの内容と一致しません。 - コピー (オリジナルとして) または永久的な WP ビット (保護しない)への入れ替えが試みられました。	C
15	WP_ERASE_SKIP	E X	'0' = 保護されない '1' = 保護されている	既存の書き込みが理由で、部分的なアドレスだけが消去されました。	C
14	CARD_ECC_DISABLED	S X	'0' = イネーブル '1' = ディセーブル	内部 ECC なしで、コマンドが実行されました。	A
13	ERASE_RESET		'0' = クリアされた '1' = セットされた	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、消去の順序は実行前にクリアされました。	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	コマンドを受信したときのカードの状態。コマンド実行がカードの状態を変更するような場合、次のコマンドのレスポンスで状態はホストへ見えるようになります。4つのビットは2進で0から15を示します。	B
8	READY_FOR_DATA	S R	'0' = レディでない '1' = レディ	バス上のバッファの空信号を表します。します。	
7	SWITCH_ERROR	E X	'0' = n 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 コマンドでリクエストされた期待するモードへの切り替えを行わなかった事を意味します。	B

参考資料

表 108. カード・ステータス (続き)

ビット	識別子	タイプ	値	摘要	クリア条件
6	予約済み				
5	APP_CMD	S R	'0' = ディセーブル '1' = イネーブル	カードは ACMD を期待しているか、コマンドが ACMD として認識された事を表します。	C
4	予約済み for SD I/O Card				
3	AKE_SEQ_ERROR	E R	'0' = エラーなし '1' = エラー	認証過程の順序にエラーがあります。	C
2	アプリケーション特定のコマンド用に予約されています。				
1	製造者のテストモード用に予約されています。				
0					

19.4.12 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、及び将来のアプリケーション固有の使い方に関するステータス・ビットを含んでいます。SD Status のサイズは 512bits です。このレジスタの内容は、ACMD13 が送られると (CMD55 に続き CMD13) SDIO カードホストへ転送されます。ACMD13 は、転送状態にある場合に限り送る事ができます。(カードが選択されている)

表 109 は、SD ステータス・レジスタの異なるエントリを定義したものです。テーブル内のタイプとクリア条件フィールドは、次のように省略されます。：

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、セットされます。
- X : コマンド実行中に検出、セットされます。SDIO カードホストは、これらのビットを読み込むためのステータスコマンドでカードを監視し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 : 常に以前のコマンド関連しています。有効なコマンドの受信でクリアします。
(1 コマンドの遅延で)
- C : 読込みでクリアします。

参考資料

表 109. SD ステータス

ビット	識別子	タイプ	値	摘要	クリア条件
511: 510	DAT_BUS_WIDTH	S R	'00' = 1 (デフォルト) '01' = 予約 '10' = 4bit 幅 '11' = 予約	SET_BUS_WIDTH コマンドにより設定された、現在定義されているバス幅	A
509	SECURED_MODE	S R	'0' = そのモードにありません '1' = Secured Mode にあります	カードは Secured Mode モードの動作にあります。（“SD Security Specification”を参照）	A
508: 496	予約済み				
495: 480	SD_CARD_TYPE	S R	'00xxh' = Physical Spec Ver1.01-2.00 で定義された SD メモリカード（'x' = 無視）次のカードが現在定義されています： '0000' = 通常の SD RD/WR カード '0001' = SD ROM カード	将来、8つの LSB は異なる種類の SD メモリカードを定義するためには使われます。（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われます。	A
479: 448	SIZE_OF_PROTECTED_AREA	S R	保護エリアのサイズ（下記参照）	（下記参照）	A
447: 440	SPEED_CLASS	S R	カードの速度クラス（下記参照）	（下記参照）	A
439: 432	PERFORMANCE_MOVE	S R	1[MB/s] 単位による移動性能（下記参照）	（下記参照）	A
431:428	AU_SIZE	S R	AU のサイズ（下記参照）	（下記参照）	A
427:424	予約済み				
423:408	ERASE_SIZE	S R	一度に消去される AU の数	（下記参照）	A
407:402	ERASE_TIMEOUT	S R	UNIT_OF_ERASE_AU で指定されたエリアを消去するためのタイムアウト値	（下記参照）	A
401:400	ERASE_OFFSET	S R	消去時間に追加される固定オフセット	（下記参照）	A
399:312	予約済み				
311:0	製造者のために予約されています				

SIZE_OF_PROTECTED_AREA

これらのビットをセットする事は、標準と大容量カードで異なります。標準容量のカードでは、保護されるエリアの容量は次のように計算されます。：

$$\text{保護されるエリア} = \text{SIZE_OF_PROTECTED_AREA_} * \text{MULT} * \text{BLOCK_LEN}$$

SIZE_OF_PROTECTED_AREA は MULT*BLOCK_LEN 単位で示されます。

大容量のカードでは、保護されるエリアの容量はこれらのビットに示されます。：

$$\text{保護されるエリア} = \text{SIZE_OF_PROTECTED_AREA}$$

SIZE_OF_PROTECTED_AREA はバイト単位で示されます。

参考資料

SPEED_CLASS

この 8bit フィールドは速度クラスを表します。値は $P_W/2$ (P_W は書き込み性能) で計算されます。

表 110. 速度クラス・フィールド

SPEED_CLASS	値の定義
00h	Class 0
01h	Class 2
02h	Class 4
03h	Class 6
04h – FFh	予約済み

PERFORMANCE_MOVE

この 8bit フィールドは、 P_m (パフォーマンス・ムーブ) を表します。値は 1 [MB/sec] 毎にセットされます。カードが使われた RU (記録ユニット) を移動しない場合、 P_m は無限大であると考えます。FFh にセットすると無限大である事を意味します。

表 111. パフォーマンス・ムーブ・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	無限大

参考資料

AU_SIZE

この 4bit フィールドは、AU サイズを表します。16KB から 2 の累乗で値を選択する事ができます。

表 112. 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

カード容量に依存した [表 113](#) で定義される AU の最大サイズ。カードは、RU サイズと最大 AU サイズの間で。あらゆる AU サイズにセットできます。

表 113. 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 16bit フィールドは、N_ERASE を表します。N_ERASE 個の AU が消去されたとき、タイムアウト値は ERASE_TIMEOUT ([ERASE_TIMEOUT](#) を参照) で指定されます。ホストは、1 回の動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、ホストは消去の進行状況を表示することができます。これらのビットが 0 にセットされると消去タイムアウトの計算はサポートされません。

表 114. 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

参考資料

ERASE_TIMEOUT

この 6bit フィールドは、This 6-bit field indicates T_ERASE を表します。その値は ERASE_SIZE で指定された複数の AU が消去される時に、オフセットからの消去タイムアウトを表します。ERASE_TIMEOUT の範囲は、63 秒までの間で定義することができ、製造者は用途に応じて、ERASE_SIZE と ERASE_TIMEOUT のあらゆる組み合わせを選ぶことができます。

ERASE_TIMEOUT を決定すると ERASE_SIZE が決定します。

表 115. 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 2bit フィールドは、T_OFFSET を表し、4 つの中から 1 つの値を選択できます。これらのビットは ERASE_SIZE と ERASE_TIMEOUT フィールドが 0 にセットされたときは意味を持ちません。

表 116. 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]

19.4.13 SD I/O モード

SD I/O 割込み

SD I/O カードから MultiMediaCard/SD モジュールへの割込みを実現するために、SD インタフェースのピンに割込み機能が用意されています。4 ビット SD モジュール動作において SD_D1 として使われるピン 8 は、MultiMediaCard/SD モジュールへのカード割込みを表します。割込みの使用は、各カードまたはカードの機能にとってオプションです。SD I/O 割込みはレベルに敏感ですので、割込みラインは認識され MultiMediaCard/SD モジュールによって割込みが処理される、あるいは割込み期間が過ぎてディアサートされるまで、アクティブ（ロー状態）に保持する必要があります。MultiMediaCard/SD が割込み処理を終えると、割込みステータス・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 書込みによりクリアされます。全ての SD I/O カードの割込み出力はアクティブ・ローで、MultiMediaCard/SD モジュールは全てのデータライン (SDIO_D[3:0]) 上に、プルアップ抵抗を提供します。MultiMediaCard/SD モジュールは割込み期間の間に限って、割込み検出器へのピン 8(SDIO_D/IRQ) のレベルをサンプルします。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。

割込み期間は、メモリと I/O モジュールの両方の動作に適用されます。1 つのブロックによる動作の割込み期間の定義は、複数ブロックのデータ転送の定義と異なります。

SD I/O サスペンド（停止）とリジューム（再開）

マルチファンクション SD I/O または I/O とメモリ機能を持つたカードでは、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。複数デバイスによる MMC/SD モジュールへのアクセス共有のために、SD I/O とコンボカードではオプションでサスペンド / リジュームの概念が実装されます。カードがサスペンド / リジュームをサポートするとき、MMC/SD モジュールは異なる機能やメモリへのより優先順位の高い転送用にバスを開放するために、一時的にデータ転送を中止することができます。この優先順位の高い転送が完了すると、本来の転送がアボートした所からリジューム（再開）されます。サスペンド / リジュームのサポートはカード毎のオプションです。MMC/SD バス上でサスペンド / リジューム動作を行うには、MMC/SD モジュールは次の手順を行います。：

1. SDIO_D [3:0] ラインを使っている機能を特定します。
2. 低優先順位や遅い処理にサスペンドするようリクエストします。
3. サスペンド処理の完了を待ちます。
4. 高い優先順位の処理を開始します。
5. 高い優先順位の処理の終了を待ちます。
6. サスペンドされた処理を復帰させます。

SD I/O ReadWait

オプションである ReadWait (RW) 動作は、SD-1 ビットと 4 ビットモジュールに限り定義されています。ReadWait 動作は、MMC/SD モジュールが、SD I/O デバイスの機能へコマンドを送っている間に、一時的にデータ・レジスタをストールさせるために複数のレジスタを読み込んでいる事をカードに知らせるようにします。カードが ReadWait プロトコルをサポートする事を知るには、MMC/SD モジュールは内部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、割込み期間に基づいています。

19.4.14 コマンドとレスポンス

アプリケーション固有と一般コマンド

SD カードホストモジュールは、様々な種類のアプリケーションのために標準インターフェースを提供するよう設計されています。この環境では、特定の顧客 / アプリケーションのためのリクエストがあります。これらの機能を装備するために、標準では 2 つの一般的なコマンドが定義されています。: アプリケーション固有コマンド (ACMD) と一般コマンド (GEN_CMD) です。

カードが APP_CMD(CMD55) を受け取ると、カードは次のコマンドにアプリケーション固有のコマンドを期待します。ACMD は、通常の MultiMediaCard コマンドと同じ構造を持っており、同じ CMD 番号を持つ事ができます。カードは、コマンドが APP_CMD(CMD55) の後に現れることから ACMD であると判断します。APP_CMD(CMD55) のすぐ後に続くコマンドが定義されたアプリケーション固有コマンドでない場合、標準コマンドが使われます。例えば、カードが SD_STATUS (ACMD13) の定義を持っており、APP_CMD(CMD55) の直後の CMD13 を受け取ると、これは SD_STATUS (ACMD13) であると理解されます。しかしながら、カードが APP_CMD(CMD55) の直後に CMD7 を受け取りカードが ACMD7 の定義を持っていない場合、これは、標準 (SELECT/DESELECT_CARD) CMD7 であると理解されます。

1 つの製造者固有の ACMD を使うには、SD カードホストは次の手順を踏まなければなりません。:

1. APP_CMD (CMD55) を送る

カードは MultiMediaCard/SD モジュールにレスポンスし、APP_CMD ビットがセットされ ACMD が期待されている事を表します。

2. 必要な ACMD を送る

カードは MultiMediaCard/SD モジュールにレスポンスし、APP_CMD ビットがセットされ受け取ったコマンドが、ACMD であると理解された事を表します。ACMD でないコマンドを受け取ると、カードはそれを通常の MultiMediaCard コマンドとして処理され、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。

無効なコマンドが送られると (ACMD でも CMD でもない)、それは標準の MultiMediaCard 違反コマンドエラーとして処理されます。

GEN_CMD のバス処理は、1 つのブロック読み込みまたは書込みコマンド (WRITE_BLOCK、CMD24 または READ_SINGLE_BLOCK、CMD17) と同じです。この場合には、アドレスでなく引数がデータ転送の方向を指示します。そして、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) を送る前に選択 (転送状態で) されなければなりません。データブロックサイズは、SET_BLOCKLEN (CMD16) で定義されます。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と一般コマンドは、次の 4 つのタイプに分かれます。:

- **ブロードキャストコマンド (BC)** : 全てのカードに送信、レスポンスを返しません。
- **ブロードキャストコマンドとレスポンス (BCR)** : 全てのカードに送信、レスポンス全てのカードから同時に受信されます。
- **アドレス (ポイントトゥポイント) 指定コマンド (AC)** : 選択されたカードへ送信、SDIO_D ライン上のデータ転送を含みません。
- **アドレス (ポイントトゥポイント) 指定データ転送コマンド (ADTC)** : 選択されたカードへ送信、SDIO_D ライン上のデータ転送を含みます。

参考資料

コマンド・フォーマット

コマンド・フォーマットは [表 101 \(440 ページ\)](#) を参照してください。

MultiMediaCard/SD モジュール用コマンド

表 117. ブロック重視書き込みコマンド

CMD インデックス	タイプ	引数	レスポンス・フォーマット	省略	適用
CMD23	ac	[31:16] を 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込みまたは書き込み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 コマンドで選択されたブロックサイズを書き込みます。
CMD25	adtc	[31:0] データ・アドレス	R1	WRITE_MULTIPLE_BLOCK	続く STOP_TRANSMISSION または、リクエスト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] スタッフ・ビット	R1	PROGRAM_CID	カード識別子のプログラミング。このコマンドはカードにつき 1 度だけ発行しなければなりません。カードは初めのプログラミングの後、この動作を防ぐためのハードウェアを搭載します。通常このコマンドは製造者用に予約されます。
CMD27	adtc	[31:0] スタッフ・ビット	R1	PROGRAM_CSD	CSD のプログラマブルビットをプログラミングします。

表 118. ブロック重視書き込み保護コマンド

CMD インデックス	タイプ	引数	レスponsス・フォーマット	省略	摘要
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。書き込み保護の属性は、カード固有データ (WP_GRP_SIZE) に符号化されます。
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提供する場合、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提供する場合、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リクエストします。
CMD31	予約済み				

参考資料

表 119. 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	省略	摘要
CMD32 ... CMD34	予約済み。これらのコマンド・インデックスは、様々な古いバージョンの MultiMediaCard との下位互換性を保つために予約されており、使うことはできません。				
CMD35	ac	[31:0] データ・アドレス	R1	ERASE_GROUP_START	範囲内における最初の消去グループの消去用アドレスをセットします。
CMD36	ac	[31:0] データ・アドレス	R1	ERASE_GROUP_END	連続する範囲内における最後の消去グループの消去用アドレスをセットします。
CMD37	予約済み。このコマンド・インデックスは、様々な古いバージョンの MultiMediaCard との下位互換性を保つために予約されており、使うことはできません。				
CMD38	ac	[31:0] スタッフ・ビット	R1	ERASE	以前に選択された全ての書き込みブロックを消去します。

表 120. 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	省略	摘要
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	8bit (レジスタ) データフィールドの読み込み書込みに使われます。コマンドはカードとレジスタにアドレス指定し、書き込みフラグがセットされれば、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込ま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固有のレジスタへアクセスします。
CMD40	bcr	[31:0] スタッフ・ビット	R5	GO_IRQ_STATE	システムを割込みモードにします。
CMD41	予約済み				

表 121. ロック・カード

CMD インデックス	タイプ	引数	レスポンス・フォーマット	省略	摘要
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 コマンドでセットされます。
CMD43 ... CMD54	予約済み				

参考資料

表 122. 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	省略	摘要
CMD55	ac	[31:16] RCA [15:0] スタッフ・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事をカードに表示します。
CMD56	adtc	[31:1] スタッフ・ビット [0]: RD/WR			データブロックをカードへ転送する、または汎用 / アプリケーション固有コマンド用にカードからデータを得るために使います。データブロックのサイズは、SET_BLOCK_LEN コマンドでセットされます。
CMD57 ... CMD59		予約済み			
CMD60 ... CMD63		製造者用に予約されています。			

19.5 レスポンス・フォーマット

全てのレスポンスは MCCMD 経由で送られます。レスポンスの転送は、常に対応するレスポンスコードワード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、常にスタート・ビット（常に 0）から始まり、転送の方向を表示するビットが続きます。（Card=0）下記のテーブルで x で示される値は、可変エントリを表示しています。R3 レスポンスを除く全てのレスポンスは、CRC で保護されます。全てのコマンドコードワードは、エンド・ビット（常に 1）で終了します。

5 つのタイプのレスポンスがあり、それらは次のように定義されます。：

19.5.1 R1 (ノーマル・レスポンス・コマンド)

コード長 = 48bits。45:40bits はレスポンスされるコマンドのインデックスを表します。この値は 2 進コード番号（0 から 63）と解釈されます。カードのステータスは 32bits に符号化されます。

表 123. R1 レスポンス

ビット位置	幅(ビット)	値	摘要
47	1	0	スタート・ビット
46	1	0	転送ビット
[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	エンド・ビット

参考資料

19.5.2 R1b

R1 と同じですが、オプションとしてデータライン上にビジー信号を出力します。以前のコマンド受信状態によっては、これらのコマンドの後、カードはビジーとなる場合があります。

19.5.3 R2 (CID、CSD レジスタ)

コード長 = 136bits。CID レジスタの内容は、CMD2 と CMD10 コマンドのレスポンスとして送られます。CSD レジスタの内容は、CMD9 のレスポンスとして送られます。CID と 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、レスポンスの最終ビットで置き換えられます。カードは、MCDAT をロー状態に維持する事で消去が進行中であることを表します。実際の消去時間はかなり長く、ホストは CMD7 を発行しカードをディセレクトする場合があります。

表 124. R2 レスポンス

ビット位置	幅 (ビット)	値	摘要
135	1	0	スタート・ビット
134	1	0	転送ビット
[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	エンド・ビット

19.5.4 R3 (OCR レジスタ)

コード長 : 48bit。OCR レジスタの内容は、CMD1 へのレスポンスとして送られます。レベル符号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 125. R3 レスポンス

ビット位置	幅 (ビット)	値	摘要
47	1	0	スタート・ビット
46	1	0	転送ビット
[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	エンド・ビット

参考資料

19.5.5 R4 (高速 I/O)

コード長 : 48bits。引数フィールドはアドレス指定されたカード、読み込みあるいは書き込みレジスタ・アドレス及びその内容の RCA を含みます。

表 126. R4 レスポンス

ビット位置	幅 (ビット)	値	摘要
47	1	0	スタート・ビット
46	1	0	転送ビット
[45:40]	6	'111111'	予約済み
[39:8] 引数フィールド	[31:16]	16	X
	[15:8]	8	X
	[7:0]	8	X
	[7:1]	7	'1111111'
0	1	1	エンド・ビット

19.5.6 R4b

SD I/O 専用です。CMD5 を受信する SDIO カードは独自の SDIO レスポンスである R4b レスponsします。そのフォーマットは :

表 127. R4b レスポンス

ビット位置	幅 (ビット)	値	摘要
47	1	0	スタート・ビット
46	1	0	転送ビット
[45:40]	6	x	予約済み
[39:8] 引数フィールド	39	16	カードがレディ状態
	[38:36]	3	I/O 機能数
	35	1	メモリあり
	[34:32]	3	スタッフ・ビット
	[31:8]	24	I/O ORC
	[7:1]	7	予約
0	1	1	エンド・ビット

SD I/O カードが一旦 CMD5 を受信すると、カードの I/O 部分は以降の全てのコマンドに、通常通りにレスポンスするようイネーブルされます。I/O カードの機能の、この I/O イネーブルはリセット、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 がカードにより受信されるまで、セットされたままとなります。SD メモリオンリー・カードは、CMD5 にレスポンスする場合があることに注意してください。メモリだけのカードの正しいレスポンスは、メモリあり = 1 と I/O 機能数 = 0 です。SD Memory Card specification version 1.0 に準拠するために作られたメモリオンリー・カードは、CMD5 を不正なコマンドとして認識しレスポンスしません。I/O を認識できるホストは CMD5 を送ります。もしカードが R4 レスポンスでレスポンスすると、ホストは R4 レスポンスに含まれるデータから、カードの構成を決定します。

参考資料

19.5.7 R5 (割込みリクエスト)

MultiMediaCard 専用です。コード長 : 48bits。ホストによりレスポンスが生成されると、引数の RCA フィールドは 0x0 となります。

表 128. R5 レスポンス

ビット位置	幅(ビット)	値	摘要
47	1	0	スタート・ビット
46	1	0	転送ビット
[45:40]	6	'111111'	CMD40
[39:8] 引数フィールド	[31:16]	16	獲得したカードまたはホストの RCA [31:16]
	[15:0]	16	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	エンド・ビット

19.5.8 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です。[表 129](#) に示されます。

表 129. R6 レスポンス

ビット位置	幅(ビット)	値	摘要
47	1	0	スタート・ビット
46	1	0	転送ビット
[45:40]	6	'101000'	CMD40
[39:8] 引数フィールド	[31:16]	16	獲得したカードあるいはホストの RCA [31:16]
	[15:0]	16	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	エンド・ビット

カード [23:8] ステータス・ビットは、CMD3 が I/O オンリーカードへ送られると 変更されます。この場合、レスポンスの 16bits は SD I/O オンリーカードの値です。：

- Bit [15] COM_CRC_ERROR
- Bit [14] ILLEGAL_COMMAND
- Bit [13] ERROR
- Bits [12:0] 予約

19.6 SDIO I/O カード仕様動作

次の機能は SD I/O 固有動作です。：

- SDIO_D2 信号による SDIO 読込みウェイト動作
- クロックの停止による SDIO 読込みウェイト動作
- SDIO サスPEND / リジューム動作 (書き込み及び読み込みサスPEND)
- SDIO 割込み

SDIO は SDIO_DCTRL[11] ビットがセットされた場合に限り、これらの機能をサポートします。
尚、特定のハードウェア設計を必要としない読み込みサスPENDは、これに該当しません。

19.6.1 SDIO_D2 信号による SDIO I/O 読込みウェイト動作

最初のブロックを受信する前に、ReadWait 間隔を開始させる事ができます。：データ・パスがイネーブル (SDIO_DCTRL[0] ビットがセット)、SDIO 固有動作がイネーブル (SDIO_DCTRL[11] がセット)、読み込みウェイト開始 (SDIO_DCTRL[10] =0 で SDI_DCTRL[8] =1)、データ方向はカードから SDIO (SDIO_DCTRL[1] = 1) なとき DPSM はアイドルから ReadWait へ直接移行します。ReadWait では DPSM は 2SDIO_CK クロック・サイクル後、SDIOD2 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 SDIO_D2 を 1 クロック・サイクルの間 1 に駆動するために (SDIO 仕様に合わせて)、さらに 2SDIO_CK クロック・サイクル待ちます。そして DPSM は、カードからデータを受信するまで再びウェイトを開始します。DPSM は読み込みウェイトがセットされていても、ブロックを受信中は ReadWait 間隔を開始しません。ReadWait 間隔は、CRC の受信後に開始します。新たな読み込みウェイト動作を開始するには、RWSTOP ビットはクリアされなければなりません。ReadWait 間隔中は、SDIO は SDIO_D1 上で SDIO 割込みを検出する事ができます。

19.6.2 SDIO_CK 停止による SDIO 読込みウェイト動作

SDIO カードが先の読み込みウェイト方法をサポートしない場合、SDIO は SDIO_CK を停止する事で (SDIO_DCTRL は [セクション 19.6.1](#) に記載される方法と同じですが SDIO_DCTRL[10] =1 です) 読込みウェイトを実行する事ができます。：DPSM は現在受信されるブロックの最後のビットから 2SDIO_CK サイクル後にクロックを停止し、読み込みウェイト・ビットがセットされると再びクロックをスタートします。

SDIO_CK が停止されると、どの様なコマンドもカードに発行する事ができます。ReadWait 間隔中は、SDIO は SDIO_D1 上 SDIO 割込みを検出できます。

19.6.3 SDIO サスPEND / リジューム動作

データをカードに送信する間、SDIO は書き込みをサスPENDすることができます。SDIO_CMD[11] ビットがセットされ、現在のコマンドがサスPENDコマンドであることを CPSM に表示します。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PENDが受け入れられた) 現在のブロックの CRC トークンを受信した後にアイドル状態となる DPSM への承認をします。

ハードウェアは、サスPEND動作 (リジューム) を完了するために送信される残りのブロック数を保存しません。

書き込み動作は、カードからサスPENDコマンドの ACK を受信したときに DPSM をディセーブルするだけで (SDIO_DCTRL[0] =0)、ソフトウェアによりサスPENDすることができます。そして DPSM はアイドル状態へ入ります。

読み込みをサスPENDする：サスPENDされる機能が、データ処理を停止する直前に完全なパケットを送るよう、DPSM は Wait_r 状態でウェイトします。アプリケーションは FIFO

が空になるまで、Rx FIFO を読み込み続けます。そして DPSM は自動的にアイドル状態となります。

19.6.4 SDIO 割込み

SDIO 割込みは一旦 SDIO_DCTRL[11] ビットがセットされると、SDIO_D1 ライン上で検出されます。

19.7 CE-ATA 仕様動作

次の機能は CE-ATA 固有の動作です。

- CE-ATA デバイスをディセーブルするために、コマンド完了信号を送る。
- CE-ATA デバイスからコマンド完了信号を受信する。
- ステータス・ビットと / または割込みを使い、CPU へ CE-ATA コマンドの完了を知らせる。

SDIO は SDIO_CMD[14] がセットされると、これらの動作を CE-ATA CMD61 用に限ってサポートします。

19.7.1 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、short レスポンスの受信から 8 ビットサイクル後に ‘enable CMD completion’ ビット、SDIO_CMD[12] がセットされておらず ‘not interrupt Enable’ ビット、SDIO_CMD[13] がセットされている場合に送られます。

CPSM は Pend 状態へ入りシフト・レジスタへディセーブル順序” 00001 ”を、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 Send 状態へ移動します。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となります。

19.7.2 コマンド完了信号イネーブル

もし ‘enable CMD completion’ ビット、SDIO_CMD[12] がセットされ、‘not interrupt Enable’ ビット SDIO_CMD[13] がセットされると、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 はアイドル状態へ入ります。7 ビットサイクルの間、新たなコマンドを送る事はできません。そのうち 5 サイクル (7 サイクル中) 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。

19.7.3 CE-ATA 割込み

コマンド完了は、ステータス・ビット SDIO_STA[23] により CPU へ知らせられます。この静的なビットはクリア・ビット SDIO_ICR[23] によりクリアする事ができます。

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、各割込みライン上に割込みを生成することができます。

19.7.4 CMD61 のアボート

もしコマンド完了ディセーブル信号が送信されておらず、CMD61 がアボートされる必要があると、コマンドステートマシンはディセーブルされなければなりません。するとアイドル状態となり、CMD12 コマンドを送信する事ができます。動作中、コマンド完了ディセーブル信号は送信されません。

19.8 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) とオーバーラン (RX モード) を回避するために使われます。

その動作は、SDIO_CK を停止し SDIO ステートマシンを止まらせるものです。データ転送は FIFO のデータ送受信が不可能な間停止します。SDIOCLK のクロックで動作するステートマシンだけが停止し、AHB インタフェースは生きてます。そのため FIFO はフロー制御がアクティブであっても、満たされたり空になったりします。

HW フロー制御をイネーブルするには、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。リセット後はフロー制御はディセーブルです。

19.9 SDIO レジスタ

デバイスは、AHB 経由でアクセス可能な 32bit 幅の制御レジスタによりシステムと通信します。

19.9.1 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																														PWRCTRL	

r

r/w

Bits 31:2 予約済み、常に 0 が読み込まれます。

[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を定義するために使われます。 :

00: パワー・オフ : カードへのクロックは停止します。

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。

注意 : データ書き込みの後、データは 7HCLK クロック周期の間書きこむことができません。

参考資料

19.9.2 SDIO 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	WIDBUS	BYPASS	PWRSAV	CLKEN	CLKDIV													
r												r/w	r/w	r/w	r/w	r/w	r/w	r/w													

Bits 31:15 予約済み、常に 0 が読み込まれます。

Bit 14 HWFC_EN: HW フロー制御イネーブル

0b: HW フロー制御はディセーブルです。

1b: HW フロー制御はイネーブルです。

HW フロー制御がイネーブルのとき TXFIFOE と RXFIFOF 割込み信号を意味します。セクション 19.9.11 の SDIO ステータス・レジスタ定義を参照してください。

Bit 13 NEGEDGE: SDIO_CK 反位相選択ビット

0b: SDIOCLK マスタークロックの立上がりエッジで生成された SDIO_CK。

1b: SDIOCLK マスタークロックの立下がりエッジで生成された SDIO_CK。

Bits 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 が使われます。

01: 4 ワイド・バス・モード : SDIO_D[3:0] が使われます。

10: 8 ワイド・バス・モード : SDIO_D[7:0] が使われます。

Bit 10 BYPASS: クロック分周器バイパス・イネーブル・ビット

0: バイパスをディセーブル : SDIOCLK は SDIO_CK を駆動する前に CLKDIV の値で分周されます。

1: バイパスをイネーブル : SDIOCLK は直接 SDIO_CK 出力信号を駆動します。

Bit 9 PWRSAV: パワーセーブ構成ビット

パワーセービングのためには、SDIO_CK クロック出力は、バスがアイドル状態のとき PWRSAV をセットすることでディセーブルすることができます。:

0: SDIO_CK クロックは常にイネーブルです。

1: SDIO_CK はバスがアクティブなときに限りイネーブルです。

Bit 8 CLKEN: クロック・イネーブル・ビット

0: SDIO_CK はディセーブルです。

1: SDIO_CK はイネーブルです。

Bits 7:0 CLKDIV: クロック分周ファクタ

これらのビットは、入力クロック (SDIOCLK) と出力クロック (SDIO_CK) の間で分周ファクタを定義します。: SDIO_CK 周波数 = SDIOCLK / [CLKDIV + 2].

注意: 1 SD/SDIO カードまたは MultiMediaCard が識別モードにある場合、SDIO_CK 周波数は 400 kHz 以下でなければなりません。

2 クロック周波数は、相対アドレスが全てのカードに割り当てられている場合に、最大カードバス周波数に変更することができます。

3 データ書き込みの後、データは 7HCLK クロック周期の間、書きこむことはできません。SDIO_CK も SD I/O カードのための読み込みウェイト間隔の間停止されます。: この場合 SDIO_CLKR レジスタは SDIO_CK を制御しません。

参考資料

19.9.3 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、コマンドの一部としてカードへ送られる 32bit 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																															
r/w																															

Bits 31:0 **CMDARG:** コマンド引数

コマンドの一部としてカードに送られるコマンド引数です。コマンドが引数を含む場合、コマンドをコマンド・レジスタへ書きこむ前に、このレジスタへロードされなければなりません。

19.9.4 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・インデックスとコマンド・タイプ・ビットを持っていきます。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へ送信されます。コマンド・タイプ・ビットは、コマンド・パス・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									r/w		

Bits 31:15 予約済み、常に 0 が読込まれます。

Bit 14 **ATACMD:** CE-ATA コマンド

ATACMD がセットされると CPSM は CMD61 を送信します。

Bit 13 **nIEN:** 割込みイネーブルでない

このビットが 0 のとき CE-ATA の割込みはイネーブルです。

Bit 12 **ENCMDcompl:** CMD 完了イネーブル

このビットがセットされると、コマンド完了信号がイネーブルされます。

Bit 11 **SDIOSuspend:** SD I/O サスPENDコマンド

このビットがセットされると、送信されるコマンドはサスPENDされます。（SDIO カード専用です）

Bit 10 **CPSMEN:** コマンド・パス・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ると、CPSM がイネーブルされます。

Bit 9 **WAITPEND:** CPSM のデータ転送の最後待ち (CmdPend 内部信号)

このビットがセットされると、CPSM はコマンドを送信する前に、データ転送の最後を待ちます。

参考資料

Bit 8 WAITINT: CPSM の割込みリクエスト待ち

このビットがセットされると、CPSM はコマンドタイムアウトをディセーブルし、割込みリクエストを待ちます。

Bits 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、そしてもし待つ場合にはそのレスポンスの種類を設定するために使われます。

00: レスpons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pons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。

Bit 5:0 CMDINDEX: 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送信されます。

注意: 1 データ書き込みの後、データは 7HCLK クロック周期の間、書きこむことはできません。

2 MultiMediaCard は 2 種類のレスポンスを送信できます。: 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、136bits 長。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ます。引数はレスポンスのタイプにより様々です。: 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だけを送信します。

19.9.5 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と（ロングまたは OCR レスpons）、本来は 111111b（レスponsの予約フィールドの値）を含むはずですが RESPCMD フィールドは不定です。

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																RESPCMD															

Bits 31:6 予約済み、常に 0 が読み込まれます。

Bits 5:0 RESPCMD: レスpons・コマンド・インデックス。読み取り専用ビット。最後に受信されたコマンド・レスponsのコマンド・インデックスを含みます。

参考資料

19.9.6 SDIO レスポンス 0..4 レジスタ (SDIO_RESPx)

アドレスオフセット : $(0x14 + (4*x))$; $x = 0..4$

リセット値 : 0x0000 0000

SDIO_RESP0/1/2/3/4 レジスタは、受信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

Bits 31:0 CARDSTATUSx: [表 130](#) を参照

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 127bits となります。

表 130. 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	カード・ステータス [31:0]	カード・ステータス [127:96]
SDIO_RESP2	未使用	カード・ステータス [95:64]
SDIO_RESP3	未使用	カード・ステータス [63:32]
SDIO_RESP4	未使用	カード・ステータス [31:1]0b

カード・ステータスの最下位ビットがはじめに受信されます。SDIO_RESP3 レジスタ LSB は常に 0b です。

19.9.7 SDIO データタイマレジスタ (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、カードバスロック周期によるデータタイムアウト期間を含みます。

カウンタは SDIO_DTIMER レジスタから値をロードします。そして、データ・パス・ステート・マシンが Wait_R またはビジー状態に入ったとき、デクリメントを開始します。DPSM がこれらの何れかの状態にある間にタイマが 0 に達すると、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																															

r/w

Bits 31:0 DATETIME: データタイムアウト期間

カードバスロック周期で表されたデータタイムアウト期間です。

注意: データ転送は、データ制御レジスタへの書き込みの前にデータタイマ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。



参考資料

19.9.8 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したときにデータ・カウンタへロードされます。

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																

Res.

r/w

Bits 31:25 予約済み、常に 0 が読み込まれます。

Bits 24:0 DATALENGTH: データ長の値

転送されるデータバイト数

注意: 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) データ転送はデータ制御レジスタへ書込まれる前に、データタイマレジスタとデータ長レジスタに書込まれなければなりません。

19.9.9 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								

Res.

r/w

r/w

r/w

r/w

r/w

r/w

Bits 31:12 予約済み、常に 0 が読み込まれます。

Bit 11 SDIOEN: SD I/O イネーブル機能

このビットがセットされると、DPSM は SD I/O カード固有の動作を行います。

Bit 10 RWMOD: 読込みウェイトモード

0: 読込みウェイト制御は SDIO_CK を停止します。

1: SDIO_D2 を使った読み込みウェイトです。

Bit 9 RWSTOP: 読込みウェイト停止

0: RWSTART ビットがセットされていると、読み込みウェイトは進行中です。

1: RWSTART ビットがセットされていると、読み込みウェイト用イネーブルです。

Bit 8 RWSTART: 読込みウェイト開始

このビットがセットされていると、読み込みウェイトは開始します。

参考資料

Bits 7:4 DBLOCKSIZE: データブロックサイズ

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: (0 10 進) ブロック長 = $2^0 = 1$ byte
- 0001: (1 10 進) ブロック長 = $2^1 = 2$ bytes
- 0010: (2 10 進) ブロック長 = $2^2 = 4$ bytes
- 0011: (3 10 進) ブロック長 = $2^3 = 8$ bytes
- 0100: (4 10 進) ブロック長 = $2^4 = 16$ bytes
- 0101: (5 10 進) ブロック長 = $2^5 = 32$ bytes
- 0110: (6 10 進) ブロック長 = $2^6 = 64$ bytes
- 0111: (7 10 進) ブロック長 = $2^7 = 128$ bytes
- 1000: (8 10 進) ブロック長 = $2^8 = 256$ bytes
- 1001: (9 10 進) ブロック長 = $2^9 = 512$ bytes
- 1010: (10 10 進) ブロック長 = $2^{10} = 1024$ bytes
- 1011: (11 10 進) ブロック長 = $2^{11} = 2048$ bytes
- 1100: (12 10 進) ブロック長 = $2^{12} = 4096$ bytes
- 1101: (13 10 進) ブロック長 = $2^{13} = 8192$ bytes
- 1110: (14 10 進) ブロック長 = $2^{14} = 16384$ bytes
- 1111: (15 10 進) 予約済み

Bit 3 DMAEN: DMA イネーブル・ビット

- 0: DMA はディセーブルです。
- 1: DMA はイネーブルです。

Bit 2 DTMODE: データ転送モード選択

- 0: ブロックデータ転送
- 1: ストリームデータ転送

Bit 1 DTDIIR: データ転送方向選択

- 0: コントローラからカード
- 1: カードからコントローラ

[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 DTDIIR に応じて、DPSM は Wait_S、Wait_R 状態、転送の初めに直ちに RWs スタート・ビットがセットされると ReadWait 状態へ移行します。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、新たなデータ転送をイネーブルするために、SDIO_DCTRL はアップデートされなければなりません。

注意： データ書き込みの後は 7HCLK クロック周期の間、データをこのレジスタに書きこむ事はできません。

参考資料

19.9.10 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 状態へ移行したとき、データ長レジスタから (SDIO_DLEN を参照) 値をロードします。データが転送されると、カウンタは 0 に達するまで値をデクリメントします。そして DPSM はアイドル状態へ移り、データステータスエンドフラグである 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																
Res.															r																

Bits 31:25 予約済み、常に 0 が読みされます。

Bits 24:0 **DATACOUNT:** データカウント値

このビットが読み込まれたとき、転送される残りのバイト数が返されます。書き込みには影響しません。

注意: このレジスタを読み込みするのは、データ転送が完了したときだけです。

19.9.11 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です。このレジスタは 2 つのタイプのフラグを持っています。:

- 静的フラグ (ビット [23:22,10:0]) : これらのビットは SDIO 割込みクリア・レジスタ (SDIO_ICR を参照) に書きこむ事によりクリアされるまで、アサートされたままであります。
- 動的フラグ (ビット [21:11]) : これらのフラグは基礎回路の状態により、状態を変化させます。(例えば、FIFO フルとエンプティ (空) フラグはデータが FIFO に書込まれることでアサート及びディアサートされます)

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	TXFIFOHF	RXFIFOHE	TXFIFOHE	RXACT	TXACT	CMDACT	DBCKEND	STBITERR	DATAEND	CMDSENT	CMDREND	RXOVERR	TXUNDER	DTIMEOUT	CTIMEOUT	DCRCFAIL	CCRCFAIL
Res.										r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r			

Bits 31:24 予約済み、常に 0 が読みされます。

Bit 23 **CEATAEND:** CMD61 用に受信された CE-ATA コマンド完了信号

Bit 22 **SDIOIT:** SDIO 割込みが受信されました。

Bit 21 **RXDAVL:** 受信 FIFO にデータがあります。

Bit 20 **TXDAVL:** 送信 FIFO にデータがあります。

Bit 19 **RXFIFOE:** FIFO が空です。

参考資料

Bit 18 **TXFIFOE**: 送信 FIFO が空です。

HW フロー制御がイネーブルのとき、FIFO に 2 ワード含まれると TXFIFOE 信号が有効となります。

Bit 17 **RXFIFOF**: 受信 FIFO がフルです。

HW フロー制御がイネーブルのとき、FIFO がフルになる 2 ワード前に RXFIFOF 信号が有効となります。

Bit 16 **TXFIFOF**: 送信 FIFO がフルです。

Bit 15 **RXFIFOHF**: FIFO. 受信 FIFO が半フル状態です。少なくとも 8 ワードが FIFO にあります。

Bit 14 **TXFIFOHE**: 送信 FIFO が半エンプティ（空）状態です。少なくとも 8 ワードを FIFO に書きこむ事ができます。

Bit 13 **RXACT**: データ受信中です。

Bit 12 **TXACT**: データ転送中です。

Bit 11 **CMDACT**: コマンド転送中です。

Bit 10 **DBCKEND**: データブロック送受信 (CRC 確認成功)

Bit 9 **STBITERR**: ワイド・バス・モードにおいて全てのデータ信号上にスタート・ビットが検出されませんでした。

Bit 8 **DATAEND**: データエンド (データ・カウンタ SDIDCOUNT がゼロです。)

Bit 7 **CMDSENT**: コマンドが送信されました。(レスポンス必要なし)

Bit 6 **CMDREND**: コマンド・レスポンスが受信されました。(CRC 確認成功)

Bit 5 **RXOVERR**: FIFO オーバーラン・エラーを受け取りました。

Bit 4 **TXUNDERR**: 送信 FIFO アンダーラン・エラーです。

Bit 3 **DTIMEOUT**: データタイムアウトです。

Bit 2 **CTIMEOUT**: コマンド・レスポンス・タイムアウトです。

コマンド・タイムアウト期間は、固定値の 64SDIO_CK クロック周期です。

Bit 1 **DCRCFAIL**: データブロック送受信 (CRC 確認失敗)

Bit 0 **CCRCFAIL**: コマンド・レスポンス受信 (CRC 確認失敗)

参考資料

19.9.12 SDIO 割込みクリア・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 1b を書込むと、SDIO_STA ステータス・レジスタの対応するビットをクリアします。

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					CEATAENDC	SDIOITC									予約済み					DBCKENDC	STBITERRC	DATAENDC	CMDSENTC	CMDRENDC	RXOVERRC	TXUNDERC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC	
Res.					r/w	r/w									Res					r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	

Bits 31:24 予約済み、常に 0 が読み込まれます。

Bit 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、ソフトウェアによりセットされます。

0: CEATAEND はクリアされません。

1: CEATAEND はクリアされます。

Bit 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、ソフトウェアによりセットされます。

0: SDIOIT はクリアされません。

1: SDIOIT はクリアされます。

Bits 21:11 予約済み、常に 0 が読み込まれます。

Bit 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、ソフトウェアによりセットされます。

0: DBCKEND はクリアされません。

1: DBCKEND はクリアされます。

Bit 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、ソフトウェアによりセットされます。

0: STBITERR はクリアされません。

1: STBITERR はクリアされます。

Bit 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、ソフトウェアによりセットされます。

0: DATAEND はクリアされません。

1: DATAEND はクリアされます。

Bit 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、ソフトウェアによりセットされます。

0: CMDSENT はクリアされません。

1: CMDSENT はクリアされます。

Bit 6 CMDRENDC: CMDREND フラグ・クリア・ビット

CMDREND フラグをクリアするために、ソフトウェアによりセットされます。

0: CMDREND はクリアされません。

1: CMDREND はクリアされます。

参考資料

Bit 5 RXOVERRC: RXOVERR フラグ・クリア・ビット

RXOVERR フラグをクリアするために、ソフトウェアによりセットされます。

0: RXOVERR はクリアされません。

1: RXOVERR はクリアされます。

Bit 4 TXUNDERRC: TXUNDERR フラグ・クリア・ビット

TXUNDERR フラグをクリアするために、ソフトウェアによりセットされます。

0: TXUNDERR はクリアされません。

1: TXUNDERR はクリアされます。

Bit 3 DTIMEOUTC: DTIMEOUT フラグ・クリア・ビット

DTIMEOUT フラグをクリアするために、ソフトウェアによりセットされます。

0: DTIMEOUT はクリアされません。

1: DTIMEOUT はクリアされます。

Bit 2 CTIMEOUTC: CTIMEOUT フラグ・クリア・ビット

CTIMEOUT フラグをクリアするために、ソフトウェアによりセットされます。

0: CTIMEOUT はクリアされません。

1: CTIMEOUT はクリアされます。

Bit 1 DCRCFAILC: DCRCFAIL フラグ・クリア・ビット

DCRCFAIL フラグをクリアするために、ソフトウェアによりセットされます。

0: DCRCFAIL はクリアされません。

1: DCRCFAIL はクリアされます。

Bit 0 CCRCFAILC: CCRCFAIL フラグ・クリア・ビット

CCRCFAIL フラグをクリアするために、ソフトウェアによりセットされます。

0: CCRCFAIL はクリアされません。

1: CCRCFAIL はクリアされます。

19.9.13 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 にセットする事で、どのステータス・フラグが割込みを生成するか決定します。

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
Res.		予約済み ※約定	CEATAENDIE	SDIOITIE	RXDAVIE	TXDAVIE	RXFIFOIE	TXFIFOIE	RXFIFOIE	TXFIFOIE	RXFIFOIE	TXFIFOIE	RXFIFOIE	TXFIFOIE	RXFIFOIE	TXACTIE	TXACTIE	CMDACTIE	DBCKENDIE	STBITERRIE	DATAENDIE	CMDSENTEIE	CMDRENDEIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE		

Bits 31:24 予約済み、常に 0 が読込まれます。

Bit 23 CEATAENDIE: CE-ATA コマンド完了信号は割込みイネーブルを受信しました。

CE-ATA コマンド完了信号を受信したときに生成される割込みを、イネーブル / ディセーブルするためにはソフトウェアによりセット / リセットされます。

0: CE-ATA command completion signal received interrupt disabled

1: CE-ATA command completion signal received interrupt enabled

参考資料

Bit 22 SDIOITIE: SDIO モード割込み受信割込みイネーブル

SDIO モード割込みを受信したときに生成される割込みを、イネーブル / ディセーブルするため
にソフトウェアによりセット / リセットされます。

- 0: SDIO モード割込み受信による割込みは、ディセーブルです。
- 1: SDIO モード割込み受信による割込みは、イネーブルです。

Bit 21 RXDAVIE: Rx FIFO データ有無割込みイネーブル

Rx FIFO 内のデータの有無を受信したときに生成される割込みを、イネーブル / ディセーブルす
るためにソフトウェアによりセット / リセットされます。

- 0: Rx FIFO のデータの有無による割込みは、ディセーブルです。
- 1: Rx FIFO のデータの有無による割込みは、イネーブルです。

Bit 20 TXDAVIE: Tx FIFO データ有無割込みイネーブル

Tx FIFO 内のデータの有無を受信したときに生成される割込みを、イネーブル / ディセーブルす
るためにソフトウェアによりセット / リセットされます。

- 0: Tx FIFO のデータの有無による割込みは、ディセーブルです。
- 1: Tx FIFO のデータの有無による割込みは、イネーブルです。

Bit 19 RXFIFOEIE: Rx FIFO エンプティ（空）割込みイネーブル

Rx FIFO がエンプティ（空）になった事を受信したときに生成される割込みを、イネーブル /
ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: Rx FIFO がエンプティ（空）となった事による割込みは、ディセーブルです。
- 1: Rx FIFO がエンプティ（空）となった事による割込みは、イネーブルです。

Bit 18 TXFIFOEIE: Tx FIFO エンプティ（空）割込みイネーブル

Tx FIFO がエンプティ（空）となった事を受信したときに生成される割込みを、イネーブル / ディ
セーブルするためにソフトウェアによりセット / リセットされます。

- 0: Tx FIFO がエンプティ（空）となった事による割込みは、ディセーブルです。
- 1: Tx FIFO がエンプティ（空）となった事による割込みは、イネーブルです。

Bit 17 RXFIFOFIE: Rx FIFO フル割込みイネーブル

Rx FIFO がフルになった事を受信したときに生成される割込みを、イネーブル / ディセーブルす
るためにソフトウェアによりセット / リセットされます。

- 0: Rx FIFO がフルとなった事による割込みは、ディセーブルです。
- 1: Rx FIFO がフルとなった事による割込みは、イネーブルです。

Bit 16 TXFIFOFIE: Tx FIFO フル割込みイネーブル

Tx FIFO がフルになった事を受信したときに生成される割込みを、イネーブル / ディセーブルす
るためにソフトウェアによりセット / リセットされます。

- 0: Tx FIFO がフルとなった事による割込みは、ディセーブルです。
- 1: Tx FIFO がフルとなった事による割込みは、イネーブルです。

Bit 15 RXFIFOHIE: Rx FIFO 半フル状態割込みイネーブル

Rx FIFO が半フル状態になった事を受信したときに生成される割込みを、イネーブル / ディセー
ブルするためにソフトウェアによりセット / リセットされます。

- 0: Rx FIFO が半フル状態となった事による割込みは、ディセーブルです。
- 1: Rx FIFO が半フル状態となった事による割込みは、イネーブルです。

Bit 14 TXFIFOHEIE: Tx FIFO 半エンプティ（空）状態割込みイネーブル

Tx FIFO が半エンプティ（空）状態になった事を受信したときに生成される割込みを、イネーブル /
ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: Tx FIFO が半エンプティ（空）状態となった事による割込みは、ディセーブルです。
- 1: Tx FIFO が半エンプティ（空）状態となった事による割込みは、イネーブルです。

参考資料

Bit 13 RXACTIE: データ受信動作割込みイネーブル

データが受信された動作（データ受信アクティング）により生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: データ受信動作割込みは、ディセーブルです。
- 1: データ受信動作割込みは、イネーブルです。

Bit 12 TXACTIE: データ送信動作割込みイネーブル

データが送信された動作（データ送信アクティング）により生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: データ送信動作割込みは、ディセーブルです。
- 1: データ送信動作割込みは、イネーブルです。

Bit 11 CMDACTIE: コマンド動作割込みイネーブル

コマンドが送信された動作（コマンド動作アクティング）により生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: コマンド動作割込みは、ディセーブルです。
- 1: コマンド動作割込みは、イネーブルです。

Bit 10 DBCKENDIE: データブロックエンド割込みイネーブル

データブロックエンドにより生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: データブロックエンド割込みは、ディセーブルです。
- 1: データブロックエンド割込みは、イネーブルです。

Bit 9 STBITERRIE: スタート・ビット・エラー割込みイネーブル

スタート・ビット・エラーにより生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: スタート・ビット・エラー割込みは、ディセーブルです。
- 1: スタート・ビット・エラー割込みは、イネーブルです。

Bit 8 DATAENDIE: データエンド割込みイネーブル

データエンドにより生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: データエンド割込みは、ディセーブルです。
- 1: データエンド割込みは、イネーブルです。

Bit 7 CMDSENTIE: コマンド送信割込みイネーブル

コマンド送信により生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: コマンド送信割込みは、ディセーブルです。
- 1: コマンド送信割込みは、イネーブルです。

Bit 6 CMDRENDIE: コマンド・レスポンス受信割込みイネーブル

コマンド・レスポンス受信により生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: コマンド・レスポンス受信割込みは、ディセーブルです。
- 1: コマンド・レスポンス受信割込みは、イネーブルです。

Bit 5 RXOVERRIE: Rx FIFO オーバーラン・エラー割込みイネーブル

Rx FIFO オーバーラン・エラーにより生成される割込みを、イネーブル / ディセーブルするためにソフトウェアによりセット / リセットされます。

- 0: Rx FIFO オーバーラン・エラー割込みは、ディセーブルです。
- 1: Rx FIFO オーバーラン・エラー割込みは、イネーブルです。

参考資料

Bit 4 TXUNDERRIE: Tx FIFO アンダーラン・エラー割込みイネーブル

Tx FIFO アンダーラン・エラーにより生成される割込みを、イネーブル / ディセーブルするため
にソフトウェアによりセット / リセットされます。

- 0: Tx FIFO アンダーラン・エラー割込みは、ディセーブルです。
- 1: Tx FIFO アンダーラン・エラー割込みは、イネーブルです。

Bit 3 DTIMEOUTIE: データタイムアウト割込みイネーブル

データタイムアウトにより生成される割込みを、イネーブル / ディセーブルするためソフ
トウェアによりセット / リセットされます。

- 0: データタイムアウト割込みは、ディセーブルです。
- 1: データタイムアウト割込みは、イネーブルです。

Bit 2 CTIMEOUTIE: コマンドタイムアウト割込みイネーブル

コマンドタイムアウトにより生成される割込みを、イネーブル / ディセーブルするためソフ
トウェアによりセット / リセットされます。

- 0: コマンドタイムアウト割込みは、ディセーブルです。
- 1: コマンドタイムアウト割込みは、イネーブルです。

Bit 1 DCRCFAILIE: データ CRC 失敗割込みイネーブル

データ CRC 失敗により生成される割込みを、イネーブル / ディセーブルするためソフ
トウェアによりセット / リセットされます。

- 0: データ CRC 失敗割込みは、ディセーブルです。
- 1: データ CRC 失敗割込みは、イネーブルです。

Bit 0 CCRCFAILIE: コマンド CRC 失敗割込みイネーブル

コマンド CRC 失敗により生成される割込みを、イネーブル / ディセーブルするためソフ
トウェアによりセット / リセットされます。

- 0: コマンド CRC 失敗割込みは、ディセーブルです。
- 1: コマンド CRC 失敗割込みは、イネーブルです。

19.9.14 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 へ書込む、または読出す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データ長レジスタ
(SDIO_DLLEN を参照) からロードされます。もし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 3bytes は 1 つのワードとしてみなされます。

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															

Bits 31:24 予約済み、常に 0 が読み込まれます。

Bits 23:0 FIFOCOUNT: FIFO へ書込まれる、または読み出される残りのバイト数

参考資料

19.9.15 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

受信及び送信 FIFO は、32bit 幅のレジスタとして読み書きする事ができます。FIFO は、32 の連続したアドレスで 32 個のエントリを含みます。これにより、CPU は FIFO への読み書きを行うための複数の演算値のロード及びストアに使う事ができます。

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																															
r/w																															

bits 31:0 **FIFOData:** 受信及び送信 FIFO データ

FIFO データは、アドレス SDIO ベース + 0x080 ~ SDIO ベース + 0xFC から、32 個の 32bit ワードのエントリを占有します。

参考資料

19.9.16 SDIO レジスタ・マップ

次の表は SDIO レジスタを要約したものです。

表 131. SDIO レジスタ・マップ

アドレスオフセット	名称	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																														PWRCTRL		
0x04	SDIO_CLKCR																														HWFC_EN NEGEDGE		
0x08	SDIO_ARG																														CMDARG		
0x0C	SDIO_CMD																														CE-ATACMD nIEN ENCM/Dempl SDIO Suspend CP SMEN WAITPEND WAITINT WAITRESP CMDINDEX		
0x10	SDIO_RESPCMD																														RESPCMD		
0x14	SDIO_RESP1																														CARDSTATUS1		
0x18	SDIO_RESP2																														CARDSTATUS2		
0x1C	SDIO_RESP3																														CARDSTATUS3		
0x20	SDIO_RESP4																														CARDSTATUS4		
0x24	SDIO_DTIMER																														DATATIME		
0x28	SDIO_DLEN	予約済み																													DATALENGTH		
0x2C	SDIO_DCTRL																														CLKDIV		
0x30	SDIO_DCOUNT	予約済み																													DATACOUNT		
0x34	SDIO_STA	予約済み																													SDIOEN		
0x38	SDIO_ICR	予約済み																													RWMOD		
0x3C	SDIO_MASK	予約済み																													RWSTOP		
0x48	SDIO_FIFOCNT	予約済み																													FIFOCOUNT		
0x80	SDIO_FIFO																														FIFOData		

注意： レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。

20 USB フルスピード・デバイス・インターフェース (USB)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適用されます。

20.1 概要

USB ペリフェラルは、フルスピード USB 2.0 バスと APB1 バスとのインターフェースです。

USB のサスペンド / レジューム動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。

20.2 USB の主な機能

- USB 規格バージョン 2.0 フルスピードに準拠
- 1 ~ 8 個の間でエンドポイント数を設定可能
- Cyclic Redundancy Check (CRC) 生成 / 検査、Non-Return-to-Zero Inverted (NRZI) エンコード / デコードと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 / レジューム動作
- フレームに同期したクロックパルスの生成

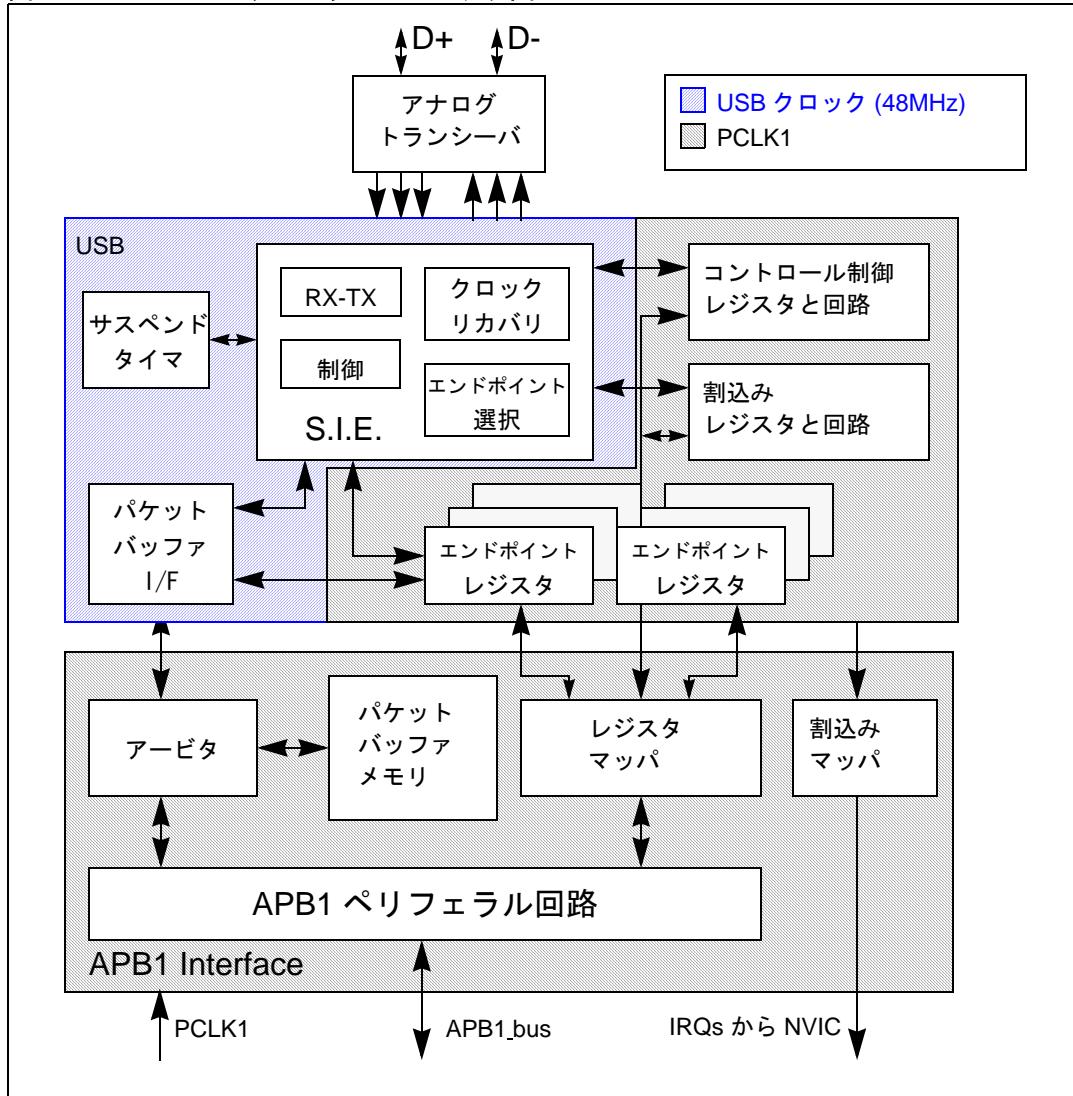
注意 :

USB と CAN は専用の 512byte SRAM メモリをデータ送受信のために共有しており、両者を同時に使用することはできません。(SRAM はいずれか一方からのアクセスに限られています。) ただし、同時になければ、同じアプリケーションの中で USB と bxCAN の両方を使用することは可能です。

20.3 USB の機能詳細

図 189 は USB ペリフェラルのブロック図を示しています。

図 189. USB ペリフェラル・ブロック図



USB 回路は、ホスト PC とマイクロコントローラにより実現される機能との間の、USB 仕様に準じた接続を提供しています。ホスト PC とシステム・メモリとの間のデータ転送は、USB 回路から直にアクセスできる専用のパケット・バッファ・メモリを通して実行されます。専用メモリは 512 bytes のサイズで、最大 16 個の一方向、もしくは最大 8 個の双方向エンドポイントをサポートできます。この USB 回路は USB ホストと接続され標準仕様に準じた動作、例えばトークンパケットの解析、データ送受信、ハンドシェークパケット処理を行います。CRC 生成とチェックを含むトランザクションの構成はハードウェアによって行われます。

個々のエンドポイントは、バッファ・ディスクリプタ・ブロックと対応しています。バッファ・ディスクリプタ・ブロックには、エンドポイントに関するメモリ・ドメインの所在、サイズや転送すべきデータのバイト数が格納されています。有効なコマンドとエンドポイントの組み合わせを示すトークンが USB 回路で認識されると、これに伴うデータ転送が（必要で、かつエンドポイントが正しく設定されているときには）行われます。USB 回路に蓄えられているデータは内部の 16bit レジスタにロードされ、専用バッファへのメモリ・アクセスが始まります。すべてのデータが転送されると、必要ならば、データ転送の方向に応じて、USB のハンドシェイクパケットが、生成され、もしくは待受けられます。

処理の最後で、エンドポイントに特有の割込みリクエストが発生します。マイクロコントローラはステータス・レジスタを読み、それに応じた割込み処理ルーチンを実行し、次の項目を確認します。

- どのエンドポイントがサービス対象か。
- ビットスタッフィング、フォーマット、CRC、プロトコル、ACKなし、オーバラン / アンダランなどのエラーが生じた場合、どのような処理が実行されたのか。

アイソクロナス転送と高速のバルク転送のために、ダブルバッファ方式が用意されており、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 回路は他方のバッファを使用することができます。

USB 回路は、必要なときに制御レジスタに書込みを行うことで、低消費電力（サスペンション・モード）の状態とすることができます。このとき、すべての静的な電力消費が抑えられ、USB クロックを低速化、もしくは停止することができます。低消費電力モードで USB 入力の動作が検出されると、USB 回路は直ちにウェークアップされます。特別の割込みリクエストは直にウェークアップ・ライン（第 4 章を参照してください）に接続することができ、システムを直ちに通常のクロック生成状態に戻し、USB 回路外から供給されるクロックのオン・オフをサポートすることができます。

20.3.1 USB ブロック

USB 回路には、次のようなブロックにより、USB 接続に関連するすべての機能が組込まれています。

- シリアル・インターフェース・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 生成及び検査、PID 確認 / 生成、及びハンドシェイクの実行です。ブロックは USB トランシーバとのインターフェースとなり、ローカルなメモリとしてパケット・バッファ・インターフェースで提供される仮想バッファを使用します。このブロックは、また、USB 回路イベントに応じた信号を生成します。イベントには、フレーム開始 (SOF)、USB リセット、データエラーなどと、送信完了や正常なパケット受信などのエンドポイント関係のイベントが含まれます。これらの信号は割込みリクエストを発生させるために使用されます。
- タイマ : このブロックは、フレーム開始同期を要する外部デバイスのために、フレームに同期したクロックパルスを生成します。このブロックは、また、トラフィックが 3 ms 受信されないときには、ホストからのグローバルサスペンドが検出されたものとします。
- パケット・バッファ・インターフェース : このブロックは送受信で使用されるバッファ・ドメインとして利用されるローカルメモリの管理を行います。このブロックは SIE からのリクエストにしたがって、適当な種類のバッファを、エンドポイントレジスタが指すメモリ・アドレスに配置します。パケットの終了まで、データの転送後にアドレスを加算し、転送バイト数を数え、バッファが最大サイズを超えてオーバランすることを防ぎます。

参考資料

- エンドポイント関連レジスタ：個々のエンドポイントは、そのタイプと現在の状態を格納しているレジスタと関連付けられています。一方向・シングルバッファのエンドポイントでは、1 個のレジスタが独立した 2 個のエンドポイントの情報を格納することができます。レジスタは全部で 8 個あり、16 個の一方向・シングルバッファエンドポイントか 7 個のダブルバッファエンドポイント、もしくはそれらの組合わせを実現できます。たとえば、USB 回路を 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構成することができます。
- 制御レジスタ：これらのレジスタは USB 回路全体のステータス情報を格納し、パワー・ダウンやレジュームといった USB イベントを生じさせるために使用されます。
- 割込みレジスタ：これらのレジスタは割込みリクエストマスクとイベントの発生状態を格納しています。レジスタは割込みの要因を確認するため割込み状態を読出したり、ペンドイングの割込みリクエストをクリアするために利用されます。

注意：

* エンドポイント 0 は常にシングルバッファモードのコントロール転送に設定されます。

USB 回路は APB1 インタフェースを介して APB1 バスに接続されています。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：このブロックは物理的にパケット・バッファを格納するローカルメモリです。このメモリはパケット・バッファ・インターフェースで使用されます。また、アプリケーションソフトウェアからも直接アクセスできます。パケット・メモリのサイズは 512bytes で、256 ワードの 16bit ワード編成となっています。
- アービタ：このブロックは APB1 バスからと USB インタフェースからのメモリ・アクセスを受け付け、APB1 からのアクセスを優先し、同時にメモリ帯域幅の 1/2 をすべての USB 転送を完了させるために確保することで、アクセス間の競合を解消します。この時分割方式は USB トランザクションが発生したときにも、仮想的なデュアルポート RAM としての働きを実現しています。任意サイズの複数ワード APB1 転送はこの方式でも利用できます。
- レジスタマッパ：このブロックは USB 回路のバイト幅及びビット単位のレジスタを集めて、APB1 空間のアドレスを付された 16bit 幅のワードレジスタに構成します。
- 割込みマッパ：このブロックは USB イベントを割込みコントローラ (NVIC) の割込みリクエスト (IRQ) ラインに配置します。
- APB1 周辺回路：このブロックはメモリとレジスタのために APB1 とのインターフェースを提供します。また、USB 回路全体を APB1 アドレス空間に配置します。

20.4 プログラミングの際の注意

このセクションでは、アプリケーションプログラムの作成を容易にするため、アプリケーションプログラムに期待される USB 回路の操作について記述してあります。

20.4.1 一般的な USB デバイスのプログラミング

このセクションでは、USB 規格に準じた動作をさせるためにアプリケーションソフトウェアにリクエストされる主な役割について記述してあります。一般的な USB イベントに対する操作が対象となっており、いくつかの段落は、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について説明しています。システム・リセット以外では、操作はすべて USB 回路の下記のような USB イベントによって開始されます。

20.4.2 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、パワー・オン・リセットの時に、アプリケーションソフトウェアが最初に行う操作は、USB 回路で必要とされるクロック信号を供給し、その後にリセット信号を解除し、レジスタにアクセスできるようにすることです。初期化手順は以下のとおりです。

最初の段階として、アプリケーションソフトウェアは USB 回路で使用するレジスタマイクロセルクロックが供給されるようにし、次に、所定の制御ビット操作でマイクロセルのリセット信号を解除します。

次に、USB_CNTR レジスタの PDWN ビットを使用して USB トランシーバなどのアナログ部分をオン状態とします。この際、特別な操作が必要となります。PDWN ビットはトランシーバに電源を供給している内部の電源をオンとします。この回路はあるスタートアップタイム（データシートに示される $t_{STARTUP}$ ）を有しており、この間のトランシーバの動作は定まらないため、PDWN ビットをセットした後この時間を待ち、その後に USB 回路のリセット状態を解除する（USB_CNTR レジスタの FRES ビットのクリア）ことが必要とされます。また、USB 回路の他の部分の動作を開始する前に、割込みステータス・レジスタ USB_ISTR をクリアすることで、この間にセットされている無効な割込みリクエストを除去することができます。

最後のステップとして、USB に固有の 48 MHz クロックを有効としなければなりません。これはデバイスのクロック管理ロジックにある所定の制御ビットの操作で行います。

システム・リセット時には、USB 回路が適切な割込みとデータ転送を行えるよう、マイクロコントローラは全レジスタとパケット・バッファ・ディスクリプタ・テーブルを初期化しなければなりません。エンドポイントに対応するレジスタ以外の全レジスタは、アプリケーションの必要に応じて、初期化されねばなりません。（有効な割込みの選択やパケット・バッファのアドレスなど）初期化処理は USB リセットイベントの処理（次項）に引き継がれます。

USB リセット (RESET 割込み)

このイベントが発生したとき、USB 回路は、前項で述べられているシステム・リセットに対する初期化後と同じ状態、つまりすべてのエンドポイントレジスタで交信が停止され USB 回路はパケットに応答しない状態に置かれます。USB リセットイベントに対する処理として、USB の機能はイネーブルとされ、USB アドレスが 0 で、デフォルトのコントロールエンドポイント（エンドポイント 0）のみが有効な状態とされなければなりません。これは、USB_DADDR レジスタの EF (enable function) ビットをセットし、USB_EP0R レジスタと対応するパケット・バッファを初期化することで実現できます。USB の番号付けプロセスで、ホストはデバイスにユニークなアドレスを付与し、このアドレスは USB_DADDR レジスタの ADD[6:0] ビットに書込まれ、リクエストされるエンドポイントの設定が行われます。

RESET 割込みリクエストが受けられると、アプリケーションソフトウェアはデフォルト エンドポイントの USB 機能 0 を、割込みをトリガしたリセットシーケンスの終端から 10 msec. 以内に有効とすることが必要です。

パケット・バッファの構造と役割

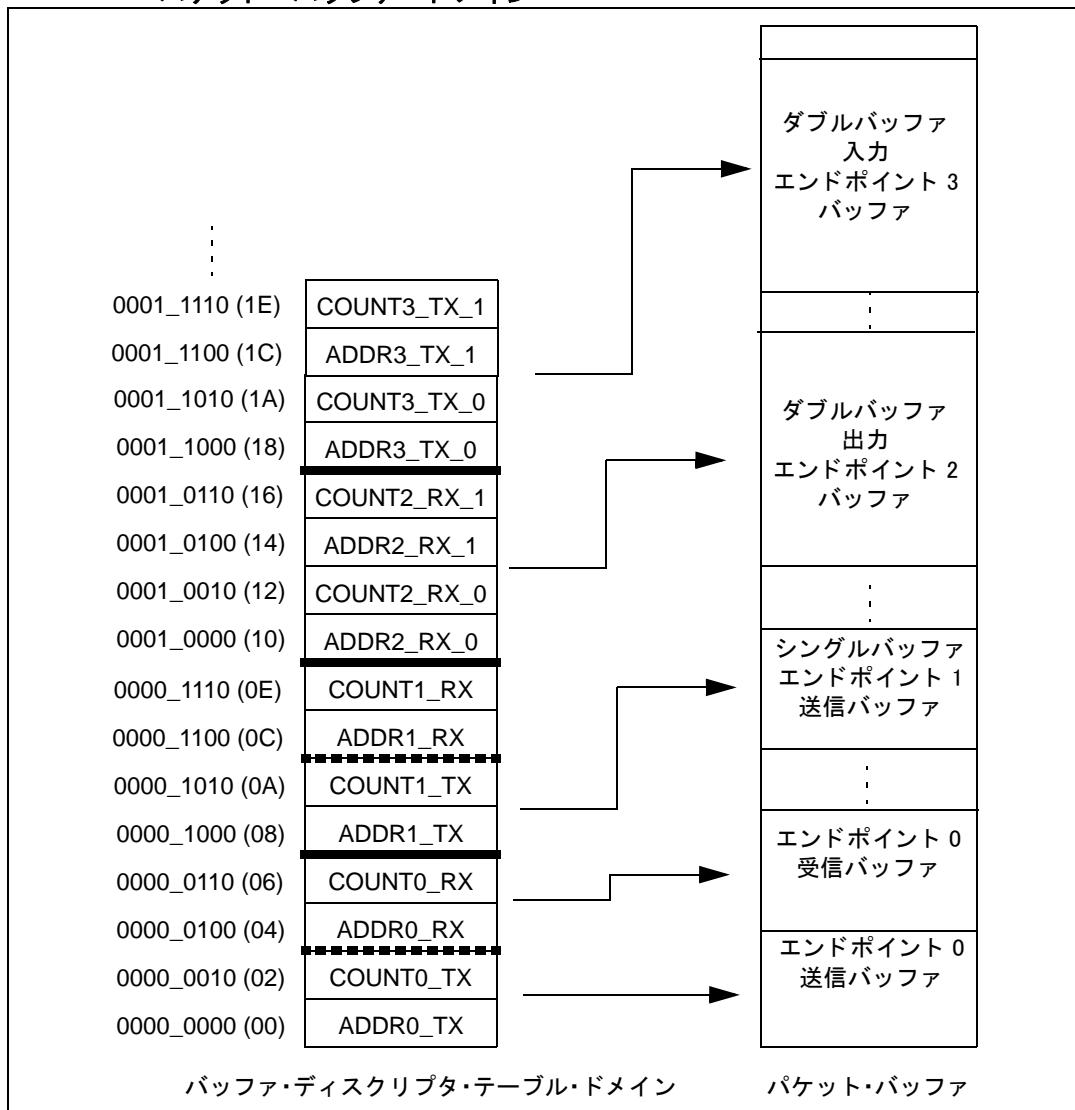
個々の両方向エンドポイントはホストとの間でデータの送受信を行うことができます。受信されたデータはそのエンドポイント専用のメモリ・バッファに格納され、もうひとつのメモリ・バッファがこのエンドポイントから送信されるデータを保持しています。このメモリへのアクセスはパケット・バッファ・インターフェース・ロックにより行われ、メモリ・アクセスリクエストが発行され、応答が待たれます。パケット・バッファ・メモリはマイクロコントローラからもアクセスされるため、アービタ回路が、APB1 サイクルの 1/2 をマイクロコントローラからのアクセスに当て、残りの 1/2 を USB 回路のアクセスに配置する手法で、アクセスの競合を管理しています。このようにして、一方でマイクロコントローラがアクセスを行っている間も、パケット・メモリが競合をまったく生じないデュアルポート SRAM 上にあるかのように動作を行うことができます。USB 回路は専用のクロックを使用します。このクロックの周波数は USB 規格を満たすため 48 MHz に固定されており、APB1 バスとのインターフェース・クロックと同じである必要はありません。APB1 クロック周波数は USB 回路のクロックより高くも、低くも選ぶことができます。

注意: USB のデータレートとパケット・メモリ・インターフェースのリクエストから、データのオーバラン / アンダランを避けるためには、APB1 クロック周波数は 8 MHz 以上でなければなりません。

個々のエンドポイントは 2 個のバッファ（多くの場合、送信と受信でそれぞれ 1 個）と関連付けられています。バッファのアドレスとサイズは、同じメモリ上にありアドレスが USB_BTABLE レジスタで指定されるバッファディスクリプションテーブルに格納されています。そのため、バッファはパケット・メモリ上の任意の場所に置くことができます。テーブルの各エントリはエンドポイントレジスタに対応しており、それぞれ 4 個の 16bit ワードで構成されています。テーブルのアドレスは 8 bytes 境界に位置付けられます。（USB_BTABLE レジスタの最下位 3bit は常に 0 となります。）バッファ・ディスクリプタ・テーブルの構造は [セクション 20.5.3: バッファ・ディスクリプタ・テーブル](#) に記載されています。エンドポイントが一方向で、アイソクロナスもしくはダブルバッファのバルク転送でなければ、パケット・バッファは 1 個のみが必要とされます。サポートされていない転送方向、もしくは使用されていないエンドポイントに対応するテーブルエントリはユーザが利用することができます。アイソクロナスもしくはダブルバッファのバルク転送のエンドポイントでは、特定のパケット・バッファの取扱いが必要とされます。（アイソクロナス転送に関しては [セクション 20.4.4: アイソクロナス転送を、ダブルバッファバルク転送に関しては](#) [セクション 20.4.3: 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・ディスクリプタ・テーブルとパケット・バッファの関係については [図 190](#) に図示されています。

参考資料

図 190. パッファ・ディスクリプション・テーブル・ロケーションの
パケット・パッファ・ドメイン



個々のパケット・バッファは、送受信に、バッファの先頭から使用されます。USB 回路は設定されているメモリ・バッファ以外のメモリの内容を変更することはありません。用意されているバッファより大きなパケットが受信されたとき（バッファオーバラン）には、データはバッファの最後のアドレスまでが書き込まれます。

エンドポイントの初期化

エンドポイントを初期化する最初の手順は、バッファ・ディスクリプタ・テーブルの ADDR_n_TX/ADDR_n_RX フィールドに適当な値を書き込み、USB 回路が、既にバッファに用意されており送信すべきデータ、もしくは受信データが入れられるバッファを見出せるようにします。USB_EPnR レジスタの EP_TYPE ビットはエンドポイントの種類にしたがってセットされていなければなりません。また、そのエンドポイントの特別な特徴を有効にするために EP_KIND ビットを使用します。送信の場合、エンドポイントは USB_EPnR レジスタの STAT_TX ビットを使用してイネーブル状態とされており、COUNT_n_TX は初期化されていなければなりません。受信の場合には、STAT_RX ビットがイネーブル状態にセットされ、COUNT_n_RX は BL_SIZE と NUM_BLOCK フィールドから算出されたバッファサイズが書き込まれていなければなりません。アイソクロナス、もしくはダブルバッファのバルク転送を除く一方向のエンドポイントでは、目的とする転送方向に関するビットやレジスタの初期化のみが必要とされます。一旦、送受信動作がイネーブルとされると、USB_EPnR レジスタと ADDR_n_TX/ADDR_n_RX、COUNT_n_TX/COUNT_n_RX は、アプリケーションプログラムでは変更することはできません。ハードウェアのみが動作時にこれらの値を変更できます。データ転送が終了すると、CTR 割込みイベントによって通知され、これらを次の動作のための値に更新することができます。

IN パケット (データ送信)

IN トークンパケットを受信したとき、設定されており有効なエンドポイントにアドレスが一致すると、USB 回路はバッファ・ディスクリプタ・テーブルの ADDR_n_TX と COUNT_n_TX にアクセスし、エンドポイントのアドレスを得ます。これらはソフトウェアからは見えない内部の 16 ビットレジスタ ADDR と COUNT に格納されます。パケット・メモリは送信すべき最初のワードを読出すためにアクセスされ ([パケット・バッファの構造と役割 \(494 ページ\)](#) を参照)、USB_EPnR の DTOG_TX ビットにより、DATA0 もしくは DATA1 PID の送信が始まります。PID 送出が完了すると、メモリ・バッファから読み出された最初のバイトが出力シフト・レジスタに移され、USB バスに送出されます。最後のバイトが送出されると、CRT が算出され、送出されます。エンドポイントのアドレスが有効でなければ、USB_EPnR レジスタの STAT_TX ビットに応じて、データパケットの代わりに NAK もしくは 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 はバッファ・メモリ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を数えるために使用されます。パケット・バッファ・メモリから読み出されたワードは、下位のバイトから送出されます。送信バッファ・メモリから読出されたそれぞれのワードは USB バスに下位バイトから送出されます。送信バッファ・メモリは ADDR_n_TX で示されるアドレスから順に COUNT_n_TX / 2 ワードが読出されます。送信パケットが奇数個のバイトで構成されるときには、最後のワードの下位部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、DTOG_TX ビットは反転され、STAT_TX を 10b (NAK) としてエンドポイントを一旦無効とし、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先ず、USB_ISTR レジスタの EP_ID と DIR ビットを読出して対象のエンドポイントを確認しなければなりません。CTR_TX イベントへの対応は割込みリクエストビットをクリアすることが始まります。アプリケーションソフトウェアは他方のバッファにデータを書き込み、テーブルの COUNT_n_TX に次の転送で送信すべきデータのバイト数を格納し、最後に STAT_TX を 11b (VALID) に設定して送信動作を再び可能とします。STA_TX が 10b (NAK) である間は、このエンドポイントに対する IN リクエストにはフロー制御のひとつである NAK を返します。USB ホストは正常終了するまでトランザクションの実行を繰り返し試みます。CTR 割込みリクエストをトリガしたトランザクションに続き、同じエンドポイントに対する次の IN トランザクションの通知を失わないためには、上記の操作手順をこの順序で実行することが必要です。

OUT パケットと SETUP パケット (データ受信)

これらの 2 種のトーケンは USB 回路では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、次のコントロール転送に関する項に記述されています。OUT/SETUP PID を受信したとき、その対象が有効なエンドポイントであれば、USB 回路はバッファ・ディスクリプタ・テーブルのエンドポイントに対応する ADDRn_RX と COUNTn_RX にアクセスします。ADDRn_RX の値はそのまま内部の ADDR レジスタに読み込まれます。COUNT レジスタはリセットされ、バッファオーバーラン条件をチェックする内部の 16bit カウンタ BUF_COUNT は COUNTn_RX レジスタから読み込んだ BL_SIZE と NUM_BLOCK の値に基づき初期化されます。(これらの内部レジスタはソフトウェアからアクセスすることはできません。) USB 回路で続いて受信されたデータバイトは、はじめに受信されたバイトが下位バイトとなるように、ワード幅にパックされ、パケット・バッファの ADDR レジスタが示すアドレスに書き込まれ、バイト転送ごとに BUF_COUNT がカウントダウン、COUNT がカウントアップされます。データ終了パケットが検出されると、受信された CRC がチェックされ、受信中にエラーが発生していないときにのみ ACK ハンドシェイクパケットがホストに送り返されます。CRC もしくは他のエラー (ビットスタッフィングエラー、フレーム・エラーなど) が検出されたときには、データは、少なくともエラーが検出された位置までは、バッファにコピーされていますが、ACK パケットは戻されず、USBISTR レジスタの ERR ビットがセットされます。この場合には、通常はソフトウェアでの処理は必要ありません。USB 回路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。有効ではないエンドポイントが指定されていたときには、ACK ではなく、USB_EPnR レジスタの STAT_RX により NAK もしくは STALL ハンドシェイクパケットが送り返されます。このときには、受信メモリ・バッファにはデータは書き込まれません。

受信メモリ・バッファには、ADDRn_RX レジスタに格納されていたアドレスから、CRC を含む受信データのサイズに対応するバイト数 (ペイロードサイズ + 2) が、BL_SIZE と NUM_BLOCKS で決まるバッファサイズを上限として、データが書き込まれます。USB 回路は用意されているバッファのサイズを越えてデータを書き込むことはありません。データパケットのペイロードのサイズがバッファのサイズより大きな場合、USB 回路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。このときには割込みリクエストは発生せず、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常に終了したときには、ACK ハンドシェイクが戻され、内部レジスタ COUNT はバッファ・ディスクリプタ・テーブルの COUNTn_RX に、BL_SIZE と NUM_BLOCK フィールドには影響を与えないように、書き戻されます。USB_EPnR レジスタでは、DTOG_RX ビットは反転され、STAT_RX を 10b (NAK) としてエンドポイントを一旦無効とし、CTR_RX ビットがセットされます。トランザクションでエラーもしくはバッファオーバーランが発生したときには、先にあげられたような動作は行われません。アプリケーションソフトウェアは先ず、USBISTR レジスタの EP_ID と DIR ビットを読み出して対象のエンドポイントを確認しなければなりません。CTR_RX イベントへの対応は、USB_EPnR レジスタの SETUP ビットでトランザクションの種類を確かめることから始まります。アプリケーションソフトウェアは割込みリクエストビットをクリアし、テーブルの COUNTn_RX レジスタから受信バイト数を読み出します。受信データの処理が終わると、USB_EPnR レジスタの STAT_RX を 11b (VALID) に設定して受信動作を再び可能とします。STAT_RX が 10b (NAK) である間は、このエンドポイントに対する OUT リクエストにはフロー制御のひとつである NAK を返します。USB ホストは正常終了するまでトランザクションの実行を繰り返し試みます。CTR 割込みリクエストをトリガしたトランザクションに続き、同じエンドポイントに対する次の OUT トランザクションの通知を失わないためには、上記の操作手順をこの順序で実行することが必要です。

コントロール転送

コントロール転送は、(逆方向で転送サイズが 0 の) ステータステージに続き、0 か任意の数の同一方向データステージを伴う、SETUP トランザクションで実行されます。SETUP トランザクションはコントロールエンドポイントのみが扱うことができ、OUT トランザクションと類似していますが、対象エンドポイントの DTOG_TX と DTOG_RX ビットの値がそれぞれ 1 と 0 にセットされており、コントロール転送の初期化のため、STAT_TX と STAT_RX は共に 10b (NAK) に設定されています。これによりソフトウェアは続くトランザクションが IN か OUT かを、SETUP トランザクションの内容により指定することができます。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、通常の OUT トランザクションと SETUP トランザクションを区別しなければなりません。USB デバイスは、SETUP ステージで転送されたデータを解析することで、データステージの転送方向とバイト数を知ることができ、エラーの場合には STALL を戻すことが必要とされます。これを実行するため、最終データステージ以外のすべてのデータステージで、使用されない転送方向のステータスは STALL にセットされなければなりません。こうすることで、ホストの転送方向切替えが早すぎるときには、データステージとして STALL が戻されることになります。最後のデータステージの設定では、反対方向のステータスは NAK とされなければならず、これによりホストが転送方向を切替えようとしたときにコントロール転送の完了を待たせることができます。制御操作が正常に終了したときには、ソフトウェアはステータスの NAK を VALID で置き換え、正常終了でなければ STALL とします。同時に、データステージが OUT であれば、EP_KIND の STATUS_OUT がセットされねばならず、これにより、ステータスとトランザクションのデータ・サイズが 0 でなければ、エラーが発生します。データストラップトランザクションがサービスされたとき、アプリケーションは STATUS_OUT ビットをクリアし、新たなコマンドを受付けるため、STAT_RX を VALID にセットし、次の SETUP に続くステータステージを遅らせるために、STAT_TX を NAK とします。

USB 仕様から SETUP パケットには ACK 以外の応答を行えないため、USB 回路のエンドポイントは、先に発行されたコマンドをアボートするため、ホストから受け取った SETUP トーカンに対して NAK もしくは STALL パケットを戻すことは許されません。

STAT_RX が 01b (STALL) もしくは 10b (NAK) で SETUP トーカンが受信されると、USB 回路は必要なデータ転送を実行し、ACK を返します。エンドポイントで、前にセットされた CRT_RX 割込みリクエストが未だアプリケーションで処理していないとき (CRT_RX ビットが先の受信完了でセットされたままのとき) には、USB 回路は SETUP トランザクションを捨て、その状態によらず、一切の応答を返さないことで受信エラーのように振る舞い、ホストが再度 SETUP トーカンを送信するようにします。これにより、このエンドポイントで CRT_RX 割込みリクエストをトリガしたトランザクションに続く次の SETUP トランザクションの通知が失われることを避けることができます。

20.4.3 ダブルバッファ・エンドポイント

USB仕様で定義されているすべてのエンドポイントはそれぞれ異なったデータトラフィックモデルを持っており、仕様では、そのデータ転送操作の一般的なリクエスト事項が述べられています。ホストPCとUSBデバイスの間で大量のデータが転送される用途では、バルクタイプのエンドポイントがもっとも適しています。これは、ホストがバルクトランザクションを、USBデバイスが扱える最大の転送速度でフレーム内の利用可能なバンド域全体のすべてを利用するようスケジュールしているためです。トランザクションの出現時に、前のトランザクションでUSBデバイスがビジー状態であれば、NAK応答を返します。このとき、ホストPCは同じトランザクションを、ビジー状態が解消するまで、再送信に使用される帯域幅の点で実際の転送レートを低下させながら、繰り返し試みます。このため、ダブルバッファリングと呼ばれる手法をバルク転送のエンドポイントで利用することができます。

ダブルバッファリング方式では、データトグルシーケンスが、USB回路がデータ転送を行うためにどのバッファを使用することにも利用されます。この際、送信と受信双方のパケット・メモリを利用し、一方をアプリケーションが使用している間USB回路が他方のバッファを使えるよう、トランザクションが正常に終了するごとにバッファを交換するよう管理します。例えば、受信ダブルバッファエンドポイントを対象とするOUTトランザクションでは、USBホストからのデータを一方のバッファに書きながら、他方のバッファをマイクロコントローラのソフトウェアで利用することができます。送信エンドポイントとINトランザクションでも、同様の扱いがなされます。

バッファ交換管理は、割り当てられているバッファのアドレスとサイズを示すバッファ・ディスクリプタ・テーブルの4ワードすべてを必要とします。ダブルバッファバルクエンドポイントを対象とするUSB_EPnRレジスタでは、無条件に、データ転送方向は一方向とされます。したがって、受信エンドポイントではSTAT_RXが、送信エンドポイントではSTAT_TXがというように、一方のSTATフィールドのみが00b(Disabled)ではない値に設定されます。ダブルバッファエンドポイントで送信と受信の両方向のデータ転送を行うには、2個のUSB_EPnRレジスタが必要とされます。

ダブルバッファ機能のメリットは、可能な最大限の転送レートを達成できることです。前の項で記されているエンドポイントのフロー制御は、トランザクションの正常終了ごとにステータスをNAKにセットする代わりに、USB回路とアプリケーションの間でバッファの競合が発生した場合にのみNAKをセットするよう、変更されます。USB回路で使用されているメモリ・バッファはDTOGビット（受信エンドポイントではUSB_EPnRレジスタのDTOG_RX、送信エンドポイントではDTOG_TXビット）で指定されます。

フロー制御では、どのバッファがアプリケーションソフトウェアで使用されているかを知る必要があります。USB_EPnRレジスタには2個のDTOGビットがありますが、ダブルバッファ方式では一方向の転送のみが行い得るという制約から、その一方のみがデータとバッファの管理で使用され、他方はアプリケーションソフトウェアがどちらのバッファを利用しているかを表示するために使用されます。この新しいフラグはSW_BUFと名づけられています。以下の表は、ダブルバッファ方式の送信と受信で、USB_EPnRレジスタのDTOG/SW_BUFビットの定義を示したものです。

表 132. ダブルバッファリング・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 bit 6)	DTOG_RX (USB_EPnR bit 14)
SW_BUF	USB_EPnR bit 14	USB_EPnR bit 6

参考資料

USB 回路で現在使用されているメモリ・バッファは DTOG バッファ・フラグで指定され、アプリケーションソフトウェアで使用されているメモリ・バッファは SW_BUF フラグで表示されます。バッファ・フラグの値とパケット・バッファの対応は、いずれの場合も同様で、次の表に示されています。

表 133. バルク・ダブルバッファリング・メモリ・バッファ

エンドポイントの種類	DTOG	SW_BUF	USB 回路で使用される パケット・バッファ	アプリケーションソフトウェア で使用されるパケット・バッファ
IN	0	1	ADDRn_TX_0 / COUNTn_TX_0 Buffer description table locations.	ADDRn_TX_1 / COUNTn_TX_1 Buffer description table locations.
	1	0	ADDRn_TX_1 / COUNTn_TX_1 Buffer description table locations.	ADDRn_TX_0 / COUNTn_TX_0 Buffer description table locations.
	0	0	None ⁽¹⁾	ADDRn_TX_0 / COUNTn_TX_0 Buffer description table locations.
	1	1	None ⁽¹⁾	ADDRn_TX_0 / COUNTn_TX_0 Buffer description table locations.
OUT	0	1	ADDRn_RX_0 / COUNTn_RX_0 Buffer description table locations.	ADDRn_RX_1 / COUNTn_RX_1 Buffer description table locations.
	1	0	ADDRn_RX_1 / COUNTn_RX_1 Buffer description table locations.	ADDRn_RX_0 / COUNTn_RX_0 Buffer description table locations.
	0	0	None ⁽¹⁾	ADDRn_RX_0 / COUNTn_RX_0 Buffer description table locations.
	1	1	None ⁽¹⁾	ADDRn_RX_1 / COUNTn_RX_1 Buffer description table locations.

1. NAK 状態のエンドポイント

バルクエンドポイントにおけるダブルバッファ方式は次のように指定します。

- USB_RPnR レジスタの EP_TYPE フィールドに 00b を設定し、バルクエンドポイントであることを指示します。
- 同じレジスタの EP_KIND ビットに 1 (DBL_BUF) をセットします。

アプリケーションソフトウェアは DTOG ビットと SW_BUF ビットを、第 1 バッファを指すように初期化しなければなりません。この際、これらのビットは反転（トグル）のみが可能であることに注意が必要です。

DBL_BUF がセットされた状態で、最初のトランザクションが完了したとき、ダブルバッファ特有の制御が始まります。この内容は、同じエンドポイントで DBL_BUF がセットされている限り繰り返されます。トランザクションの完了ごとに、転送方向に応じて、そのエンドポイントの USB_EPnR レジスタの CRT_RX もしくは CRT_TX ビットがセットされます。同時に、DTOG ビットがハードウェアにより反転され、USB 回路のバッファ切替えはソフトウェアとは独立に行われます。一般的なトランザクションとは異なり、DBL_BUF がセットされた最初のトランザクションの完了時には、ステータスは変化せず 11b (Valid) のままでですが、次のトーケンパケットを受取ったとき、USB 回路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る (DTOG と SW_BUF が同じ値、[表 133 \(500 ページ\)](#) を参照) と、ステータスに 10b (NAK) がセットされます。アプリケーションソフトウェアは CTR イベント通知に応じて、割込みリクエスト・フラグをクリアし、トランザクション完了に必要な操作を行わねばなりません。アプリケーションパケット・バッファの操作が終ったとき、ソフトウェアでは SW_BUF ビットに 1 を書き込み、このビットを反転し、これにより USB 回路にバッファが使用可能であることを通知します。このように、

NAK を返すトランザクションはアプリケーションがデータを評価する時間によってのみに限られます。データの評価時間が USB バス上でトランザクションが完了するまでの時間より短ければ、フロー制御による再送信は発生せず、転送レートはホスト PC によってのみ制限されることになります。

アプリケーションソフトウェアはステータスに 11b (Valid) 以外の値をセットすることで、常にダブルバッファ機能のフロー制御に優先した制御を行うことができます。このときには、USB 回路はバッファの使用状況にかかわらず、セットされたステータスに従った動きをします。

20.4.4 アイソクロナス転送

USB 標準規格では、一定で精度の高いデータ生成 / 消費周波数を必要とするフルスピードの周辺デバイスを、アイソクロナスとしてサポートしています。典型的なデータの例としては、オーディオ・圧縮されたビデオ・その他の時間的に正確な転送がリクエストされるデータがあげられます。エンドポイントがアドレス付けのフェーズでアイソクロナスと定められたときには、USB ホストはフレーム内に必要な帯域を配置し、エンドポイントの転送方向により、フレームごとに 1 個の IN もしくは OUT パケットを送信します。帯域幅を制限する目的から、アイソクロナス転送では、転送が失敗したときに再送信を行うことはできません。このことは、アイソクロナストランザクションにはハンドシェークフェーズが存在せず、データパケットの後に ACK パケットがリクエストされないことを意味します。同じ理由で、アイソクロナス転送ではデータトグルはサポートされておらず、データパケットでは常に 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は、USB_EPnR レジスタの EP_TYPE フィールドを 10b にセットすることで選択されます。ハンドシェイクが行われませんので、STAT_RX、STAT_TX ビットは 00b (Disabled) もしくは 11b (Valid) のいずれかです。他の値は USB 仕様に従わない動作を引き起こすことがあります。アイソクロナスエンドポイントではダブルバッファ方式を採用しており、「送信用」と「受信用」パケット・メモリ・ドメインを処理の正常な終了時にバッファ切替えを管理するために利用し、アプリケーションプログラムにおける操作を容易にしています。これにより、例えば受信の場合、アプリケーションでデータが格納されているバッファを取り扱うと同時に、USB 回路で他方のバッファに受信しているデータを書込む動作ができます。

USB 回路で使用されているメモリ・バッファは、エンドポイントの DTOG ビット（送信エンドポイントで DTOG_TX、受信エンドポイントでは DTOG_RX）で [表 134](#) にしたがって示されます。

表 134. アイソクロナス・メモリ・バッファ

エンドポイントの種類	DTOG ビットの値	USB 回路で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で使用されるパケット・バッファ
IN	0	ADDRn_TX_0 / COUNTn_TX_0 buffer description table locations.	ADDRn_TX_1 / COUNTn_TX_1 buffer description table locations.
	1	ADDRn_TX_1 / COUNTn_TX_1 buffer description table locations.	ADDRn_TX_0 / COUNTn_TX_0 buffer description table locations.
OUT	0	ADDRn_RX_0 / COUNTn_RX_0 buffer description table locations.	ADDRn_RX_1 / COUNTn_RX_1 buffer description table locations.
	1	ADDRn_RX_1 / COUNTn_RX_1 buffer description table locations.	ADDRn_RX_0 / COUNTn_RX_0 buffer description table locations.

ダブルバッファ方式のバルク転送エンドポイントと同じく、アイソクロナスエンドポイントの USB_EPnR レジスタは常に一方向とされます。送信と受信の双方が可能なエンドポイントでは、2 個の USB_EPnR レジスタが必要とされます。

アプリケーションプログラムは、DTOG ビットを第 1 バッファが使用されるよう、初期化することが必要です。この際、これらのビットは反転（トグル）のみが可能であることに注意が必要です。トランザクションの完了ごとに、転送方向に応じて、そのエンドポイントの USB_EPnR レジスタの CRT_RX もしくは CRT_TX ビットがセットされます。同時に、DTOG ビットがハードウェアにより反転され、USB 回路のバッファ切替えはソフトウェアとは独立に行われます。アイソクロナス転送にはハンドシェークフェーズがなく、フロー制御が行われないため、STAT フィールドは影響を受けず、エンドポイントは 11b (Valid) のままとなります。CRC エラー、もしくはバッファオーバランの状況がアイソクロナス出力転送で検出された場合でも、正常な転送が行われたものとして扱われ、常に CTR_RX イベントがトリガれます。ただし、CRC エラーは常に USB_ISTR レジスタの ERR ビットをセットし、データにエラーがある可能性をソフトウェアに通知します。

20.4.5 サスペンド / レジュームイベント

USB 規格では、サスペンド (SUSPEND) と呼ばれる、バスからの電流供給が平均 $500 \mu\text{A}$ を超えない特殊な状態を定義しています。（セルフパワーのデバイスではこの消費電力の制限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 3 msec. の間 USB バス上でトラフィックを発生させないことで、このモードを通知します。通常の動作では 1 msec. ごとに SOF パケットが送られますが、デバイスはこの SOF パケットが続けて 3 回存在しないときにホストからの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、許可ビットがセットされていれば、割込みリクエストを発生します。デバイスがサスペンドの状態では、いわゆる RESUME 手順によって通常の動作状態に戻ります。この手順は PC からか、デバイスが直接に開始しますが、手順の終わりは常にホスト PC によって行われます。デバイスでは、この手順は通常の USB リセットイベントと同じように扱われます。

USB 回路をサスペンドとする際の具体的な手順は、全体の消費電力を低減する操作がデバイス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。

サスペンドに入る手順は一般に、アプリケーションソフトウェアの USB に関連したルーチンが USB 回路の SUSP 通知に対応する形で、次のようにになります。：

1. USB_CNTR レジスタの FSUSP ビットを 1 にセットします。この操作は USB 回路内でサスペンド・モードを開始します。サスペンド・モードが開始されると、SUSP 割込みリクエストが重複して発生するないように、SOF 受信チェックは停止します。
2. USB 以外の回路の電力消費を低減、もしくはカットします。
3. USB_CNTR レジスタの LP_MODE ビットを 1 にセットし、USB トランシーバアナログ回路の静的な電力消費をオフとしますが、レジューム検出機能は動作状態を維持します。
4. アプリケーションによっては、外部の発信回路とデバイス PLL をオフとし、USB 回路内の動作を停止させることもできます。

サスペンド・モードで USB イベントが発生したとき、クロック状態と USB 動作を通常の状態に戻すため、レジューム手順を実行することが必要です。ウェークアップのイベントが USB リセットの場合には、この処理が 10 msec. 以内に終わるよう、特別の注意を払わねばなりません。（詳細は Universal Serial Bus Specification を参照してください。）USB 回路がサスペンド状態でレジュームもしくはリセット手順を開始するには、USB_CNTR レジスタの LP_MODE ビットを直ちにクリアします。WKUP 割込みリクエストが許可された状態で割込みリクエストを発することができる場合でも、システム・クロックが回復するまでの大きな遅れ時間には十分に注意を払わねばなりません。クロックが回復するまでの遅

参考資料

れを短縮するため、レジューム手順のルーチンをサスPENDルーチンの直後に置き、システム・クロックの回復後直ちに実行されるようにすることをお勧めします。ウェークアップ系からの ESD 放電や他のノイズを避けるために、サスPEND中はデータラインの動作検出に、70 nsec. 以下のパルスを排除するような、何らかのアナログフィルタを入れることが望されます。

レジューム手続きとして実行すべき内容は、次のとおりです。：

1. 外部の発振回路とデバイス PLL をオンとします。(必要な場合のみ)
2. USB_CNTR レジスタの FSUSP ビットをクリアします。
3. レジュームをトリガしているイベントを確認します。このためには、USB_FNR レジスタの RXDP ビットと RXDM ビットから表 135 のように判断・対応します。必要ならば、上記のビットがバスのアイドル状態を示す 10b となることを検出することで、レジュームやリセット手順の終了を知ることができます。更にリセット状態の終わりは、USB_ISTR レジスタの RESET ビットが 1 にセットされることでも知ることができます。マスクされていなければ割込みリクエストを発生させることができますので、通常の手順で処理することができます。

表 135. レジューム・イベント検出

[RXDP, RXDM] 状態	ウェークアップ・イベント	必要とされるソフトウェア操作
00	ルートリセット	なし
10	なし (バス上のノイズ)	サスPEND・モードに戻る
01	ルートレジューム	なし
11	なし (バス上のノイズ)	サスPEND・モードに戻る

マウスの移動でシステム全体を起動するように、USB プロトコルに直接かかわるものでなくとも、デバイスがサスPEND・モードを抜け出すことが必要とされることがあります。この場合、レジューム手順は USB_CNTR レジスタの RESUME ビットを 1 にセットすることで開始することができます。1 – 15 msec. が経過した後 0 にリセットします。この時間は、システム・クロックが通常の状態であれば 1 msec. 間隔で発生する ESOF 割込みを利用して確認することができます。一旦 RESUME ビットがクリアされると、レジューム手順がホスト PC によって終了され、これを USB_FNR レジスタの RXDP、RXDM ビットで知ることができます。

注意： USB_CNTR レジスタの FSUSP ビットを 1 にセットして USB 回路がサスPEND・モードに入ってからでなければ、RESUME ビットは操作できません。

参考資料

20.5 USB レジスタ

USB 回路のレジスタは、次のグループに分類されます。：

- 一般レジスタ：割込み、及び制御レジスタ
- 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・バッファとして使用するパケット・メモリの位置

すべてのレジスタのアドレスは、バッファ・ディスクリプタ・テーブルの位置を除いては、USB 回路レジスタ・ベース・アドレス 0xC000 8000 を基準とする相対値で示されています。バッファ・ディスクリプタ・テーブルのアドレスは、USB_BTABLE レジスタの値を基準とするパケット・メモリ内のアドレスで示されています。すべてのレジスタのアドレスは、レジスタが 16bit であっても、32bit ワード境界に配置されています。パケット・メモリ上の配置に関しても同様です。ソフトウェアから見たパケット・メモリはアドレス 0xC000 8800 に置かれています。

レジスタの詳細記述で使用されている略語については [セクション 1.1 \(32 ページ\)](#) を参照してください。

20.5.1 一般レジスタ

これらレジスタは USB 回路の一般的な動作に影響を与えるもので、動作モードの指定、割込みリクエストの取扱い、ホスト PC から与えられるデバイス・アドレスとフレーム番号へのアクセスが対象となります。

USB コントロール制御レジスタ (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	PMAO_VRM	ERRM	WKUP_M	SUSP_M	RESE_TM	SOFM	ESOF_M	予約済み	Res.	RESUME	FSUS_P	LP_MODE	PDWN	FRES	

Bit 15 CTRM: 転送の正常終了・割込みリクエスト許可

0: 転送正常終了 (CTR) 割込みリクエスト禁止。

1: 転送正常終了 (CTR) 割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

Bit 14 PMAOVRM: パケット・メモリ・ドメイン・オーバ／アンダラン割込みリクエスト許可

0: PMAOVR 割込みリクエスト禁止。

1: PMAOVR 割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

Bit 13 ERRM: エラー割込みリクエスト許可

0: エラー割込みリクエスト禁止。

1: エラー割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

Bit 12 WKUPM: ウェークアップ (WKUP: wake-up) 割込みリクエスト許可

0: WKUP 割込みリクエスト禁止。

1: WKUP 割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

参考資料

Bit 11 SUSPM: サスPEND・モード割込みリクエスト許可

0: サスPEND・モード割込みリクエスト禁止。

1: サスPEND・モード割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

Bit 10 RESETM: USB リセット割込みリクエスト許可

0: RESET 割込みリクエスト禁止。

1: RESET 割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

Bit 9 SOFM: フレーム開始 (SOF) 割込みリクエスト許可

0: SOF 割込みリクエスト禁止。

1: SOF 割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

Bit 8 ESOFM: フレーム開始欠落 (Expected SOF) 割込みリクエスト許可

0: フレーム開始欠落 (ESOF) 割込みリクエスト禁止

1: フレーム開始欠落 (ESOF) 割込みリクエスト許可。USBISTR レジスタの対応するビットがセットされたとき、割込みリクエストが送出されます。

Bits 7:5 予約済み

Bit 4 RESUME: ウェークアップ・リクエスト

マイクロコントローラは、ホストに対してレジューム信号を送るために、このビットをセットすることができます。USB 仕様によれば、1 - 15 msec. の間レジューム信号が送出されねばなりません。これにより、ホスト PC がレジューム手順全体を扱えるようになります。

Bit 3 FSUSP: サスPEND・モード

ソフトウェアは USB バス上に 3 msec. 以上トライフィックが無いときに生じる SUSP 割込みが発生した際に、このビットをセットしなければなりません。

0: 影響なし

1: サスPEND・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な電力消費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PEND状態とする必要があれば、このビットのセットの後に、LP_MODE ビットをセットしなければなりません。

Bit 2 LP_MODE: 低消費電力モード

このモードは、サスPEND・モードでの消費電力の制約が、外部のプルアップ抵抗以外のすべての静的な消費電力を抑えることを要求している場合、このモードが使用されます。USB サスPEND 条件を満たすために、すべてのアプリケーションがシステム・クロックを停止するか、速度を低下させる用意が整ったときに、このモードに入ります。サスPEND・モード中の USB バスの動作検出 (WKUP イベント) は、直ちにこのビットをクリアします。このビットはソフトウェアでクリアすることもできます。

0: 低消費電力モードではありません。(低消費電力モードを終了します。)

1: 低消費電力モードに入ります。

Bit 1 PDWN: パワー・ダウン

このビットは、何らかの理由で USB 回路全体を動作停止状態とする際に、USB 回路のアナログ部分を完全にスイッチオフの状態とします。このビットがセットされたときには、USB 回路はトランシーバから切離され、使用することができなくなります。

0: パワー・ダウン状態から復帰します。

1: パワー・ダウン状態に入ります。

Bit 0 FRES: USB 回路の強制リセット

0: USB リセットを解除します。

1: USB 回路を、USB 上の RESET 通知と同様に、強制的にリセットします。USB 回路は、ソフトウェアでこのビットをクリアしない限り、リセット状態に保たれます。USB RESET 割込みリクエストが許可されていれば、リクエストが発せられます。

参考資料

USB 割込みステータス・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 Res. r r r r r

このレジスタはすべての割込み要因の状態を集めたもので、どの要因が割込みの原因であるかをアプリケーションソフトウェアで確認しやすいように用意されています。

レジスタの上位ビットはビット対応で個々の割込み要因の状態を表示しています。これらはイベントが発生したときにハードウェアがセットします。USB_CNTR レジスタの対応するビットがセットされていれば、USB 回路からの割込みリクエストとして送出されます。割込みルーチンでは各ビットをチェックし、必要とされる操作を行った上で、そ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、割込みリクエストはまだペンドィングであるとみなされ、割込みリクエストが再び送出されます。いくつかのビットが同時にセットされていても、割込みリクエストは 1 ラインにまとめて送られます。

エンドポイントの処理完了は、割込み応答の遅れを軽減するため、異なる手順で扱われます。エンドポイントで処理が正常に終了したときには CTR ビットがセットされ、USB_CNTR レジスタの対応するビットがセットされていれば一般割込みリクエストが発生します。これとは別に、USB_CNTR レジスタの CTRM ビットがセットされていれば、エンドポイントに固有の割込み条件が発生します。ソフトウェアで対応する USB_EPnR レジスタのペンドィングビットをクリアする (CTR ビットは表示専用です) まで、両方の割込みリクエストが発生した状態が保たれます。エンドポイント関係の割込みリクエストについては、DIR (Direction of Transaction) ビットと EP_ID ビット (表示専用) で最後に割込みリクエストの原因となったエンドポイントを確認し、対応する割込み処理ルーチンを呼び出すことができます。

ユーザは同時に発生している USB_ISTR イベントに、割込み処理ルーチンでチェックする順序により、相対的な優先順位をつけることができます。処理ルーチンではサービス対象となったイベントのみをクリアします。割込み処理ルーチンが終了すると、未処理の別のイベントから割込みリクエストが発生しますので、残った条件の処理を行います。

イベントクリアに関する誤動作を防ぐため、変更されないビットには 1 を、クリアされるビットには 0 を書込むロード命令が使用されます。(これらのビットはソフトウェアでのみクリアされます。) ビットのクリアに読み込み - モディファイ - 書込みサイクルを使用することは、読み込みと書き込みの間にハードウェアによりビットがセットされ、このイベントがマイクロプロセッサで処理されていないにもかかわらず、次の書き込み動作でこのビックをクリアしてしまうことがあるため、好ましくありません。

以下は、個々のビットの詳細です。:

Bit 15 CTR: 転送の正常終了・割込みリクエスト

このビットは、エンドポイントが処理を正常に完了したときに、ハードウェアでセットされます。ソフトウェアで DIR と EP_ID ビットを参照すれば、どのエンドポイントが割込みリクエストを発生したかを確認することができます。このビットは読み出しのみができます。

参考資料

RM0008

USB フルスピード・デバイス・インターフェース (USB)

Bit 14 PMAOVR: パケット・メモリ・ドメイン・オーバー/アンダラン割込みリクエスト

このビットは、USB メモリからのリクエストにマイクロコントローラが一定時間内に対応できなかったときにセットされます。USB 回路はこのイベントを次のように扱います。受信時、ACK ハンドシェイクが送信されないとき、もしくは送信時にビットスタッフ・エラーが発生したときは、ホストはトランザクションを繰り返します。PMAOVR 割込みリクエストは通常の動作では決して発生しません。失敗に終わった処理がホストによって再試行されたとき、ソフトウェアはこの割込みに応じて次の試行の準備をすることで、デバイスのスピードアップを図ることができます。アイソクロナス転送では再試行は行われませんので、このような状況は発生しません。このビットはソフトウェアで読み出しとクリアのみができます。0 を書込むとビットはクリアされます。1 を書込んだときにはビットの値には影響を与えません。

Bit 13 ERR: エラー割込みリクエスト

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 (No Answer): ホストからの応答がタイムアウトしました。
CRC (CRC error): トーカン、もしくはデータのいずれかの CRC でエラーが検出されました。
BST (Bit Stuffing error): PID、データ、CRC のいずれかでビットスタッフィングエラーが検出されました。
FVIO (Framing format Violation): 規格外のフレームが受信されました。(EOP が正しい位置にない、トークンビット列の誤り、など)
USB 回路とホストはソフトウェアから見えない形で再試行を行いますので、USB ソフトウェアは通常これらのエラーを無視することができます。この割込みリクエストはソフトウェア開発段階、もしくは USB バスの転送の質を監視し、問題の可能性をユーザに指摘するなどに有効です。このビットはソフトウェアで読み出しとクリアのみができます。0 を書込むとビットはクリアされます。1 を書込んだときにはビットの値には影響を与えません。

Bit 12 WKUP: ウエークアップ (wake up) 割込みリクエスト

このビットはサスペンド・モードのときに、USB 回路をウエークアップする状態が検出されると 1 にセットされます。このイベントは直ちに USB_CNTR レジスタの LP_MODE ビットをクリアし、レジューム手順が開始されたことを回路の他の部分に伝える USB_WAKEUP 信号を送出 (activate) します。このビットはソフトウェアで読み出しとクリアのみができます。0 を書込むとビットはクリアされます。1 を書込んだときにはビットの値には影響を与えません。

Bit 11 SUSP: サスペンド・モード割込みリクエスト

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・モードリクエストを検出されたときにセットされます。サスペンド条件のチェックは USB リセット後に直ちに開始され、サスペンド・モード (FSUSP=1) に入つてからレジューム手順の終わりまで停止します。このビットはソフトウェアで読み出しとクリアのみができます。0 を書込むとビットはクリアされます。1 を書込んだときにはビットの値には影響を与えません。

Bit 10 RESET: USB リセット割込みリクエスト

USB 回路が入力として USB RESET 信号を検出したときにセットされます。USB 回路は RESET に対して、内部のプロトコルへの対応状態をリセットし、USB_CNTR レジスタのマスク・ビットの RESETM がセットされていれば割込みリクエストを発生します。RESET ビットがクリアされるまで送受信動作は停止されます。このビットはソフトウェアで明確にクリアすることが必要とされます。(これは RESET 割込みが確実に通知され、RESET 直後の処理が終了させられたことを確実にするためです。) リクエスト機能のアドレスとエンドポイントを保持しているレジスタは USB リセットイベントでリセットされます。

このビットは読み出しとクリアのみが可能です。0 を書込むとクリアされますが、1 の書込みはビットの値に影響を与えません。

Bit 9 SOF: フレーム開始 (SOF) 割込みリクエスト

このビットは USB バスの新たなフレームの開始を示すために、SOF パケットが到着したときにセットされます。割込みサービスルーチンでは、このイベントを USB ホストとの 1 msec. ごとの同期をとり、SOF パケット受信で更新される USB_FNR レジスタを確実に読み出すために使用することができます。このビットはソフトウェアで読み出しとクリアのみができます。0 を書込むとビットはクリアされます。1 を書込んだときにはビットの値には影響を与えません。



参考資料

Bit 8 **ESOF**: フレーム開始欠落 (*Expected SOF*) 割込みリクエスト

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ときに、ハードウェアでセットされます。ホストは 1 msec.ごとに SOF パケットを送出しますが、これをハブが正しく受取っていない場合、サスPEND タイマはこの割込みリクエストをセットします。トライフィックがまったく無く ESOF イベントが連続して 3 回連続して発生した（3 個の SOF パケットが失われた）ときには、SUSP 割込みリクエストがセットされます。このビットは、また、サスPEND タイマの同期が確立していないときに SOF パケットが失われたときにリセットされます。このビットはソフトウェアで読み出しとクリアのみができます。0 を書込むとビットはクリアされます。1 を書込んだときにはビットの値には影響を与えません。

Bits 7:5 予約済み

Bit 4 **DIR**: トランザクションのデータ転送方向

このビットは、正常に終了し割込みリクエストを発生させたトランザクションのデータ転送の方向を示すためにハードウェアが書き込みます。

DIR=0: IN トランザクションが USB_EPnR レジスタの CTR_TX ビットをセットしました。（データ転送の方向は USB 回路からホスト PC に向けてです。）

DIR=1: OUT トランザクションが USB_EPnR レジスタの CTR_RX ビットをセットしました。（データ転送の方向は ホスト PC から USB 回路に向けてです。）もしくは、処理待ちのトランザクションは 2 個あるときには、CTR_RX もセットされます。

この情報はアプリケーションソフトウェアが割込みを発生したエンドポイントにアクセスする際に利用されます。このビットは読み出し専用です。

Bits 3:0 **EP_ID[3:0]**: エンドポイント番号

これらのビットは割込みリクエストを発生させたトランザクションのエンドポイントを示すためにハードウェアが書き込みます。いくつかのエンドポイントのトランザクションがペンディング状態であれば、次の方法で、優先度の高いエンドポイントの番号を書き込みます。アイソクロナス、及びダブルバッファ方式のバルクエンドポイントが、他のものに優先します。同じグループ内では、エンドポイントレジスタの番号が小さなものが優先します。アプリケーションソフトウェアで処理が優先されるべきエンドポイントからレジスタを割り当てることで、割込み時の処理の優先順位を妥当なものとすることができます。これらのビットは読み出し専用です。

参考資料

RM0008

USB フルスピード・デバイス・インターフェース (USB)

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 where X is undefined

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]

Bit 15 RXDP: データ + ライン状態

このビットはデータ + ラインの状態を監視するもので、受信データとアップストリームの両方がモニタできます。このビットは "end-of-suspend" ルーチンでウェークアップ・イベントを監視する際にも使用できます。

Bit 14 RXDM: データ - ライン状態

このビットはデータ - ラインの状態を監視するもので、受信データとアップストリームの両方がモニタできます。このビットは "end-of-suspend" ルーチンでウェークアップ・イベントを監視する際にも使用できます。

Bit 13 LCK: ロック

このビットは、USB リセット条件もしくは USB レジューム手順が終わった後、少なくとも 2 個の SOF パケットが検出されたときに、ハードウェアによりセットされます。一旦この状態となると、USB リセットもしくは USB サスペンドが発生するまで、フレームタイマはこの状態を維持します。

Bits 12:11 LSOF[1:0]: SOF 欠落カウント

これらのビットは、フレーム開始欠落イベント (ESOF) が発生したときに、連続したその回数をカウントするために、ハードウェアで書き込まれます。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。

Bits 10:0 FN[10:0]: フレーム番号

これらのビットは最後に受信された SOF パケットのフレーム番号 (11bits) を表示します。フレーム番号はホストからの SOF 送信ごとにカウントアップされるものでアイソクロナス転送の処理で役立ちます。これらのビットは、SOF 割込みリクエストが発生したときに更新されます。



参考資料

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
Res.								RW	RW	RW	RW	RW	RW	RW	RW

Bits 15:8 予約済み

Bit 7 EF: USB 機能有効

このビットは USB デバイスを有効とするためにセットされます。このデバイスのアドレスは次の ADD[6:0] フィールドに格納されます。この EF ビットが 0 であれば、USB_EPnR レジスタの設定にかかわらず、トランザクションの処理は行われません。

Bits 6:0 ADD[6:0]: デバイス・アドレス

これらのビットは番号付けプロセスでホスト PC から割付けられた USB アドレスが格納されます。エンドポイントに対するリクエストが処理されるためには、USB トークン内の情報と、これらのビットと USB_EPnR レジスタの EA (Endpoint Address) フィールドがマッチしていなければなりません。

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	RW	RW	Res.

Bits 15:3 BTABLE[15:3]: バッファ・テーブル・アドレス

これらのビットは専用パケット・メモリ上のバッファ・ディスクリプタ・テーブルの先頭アドレスが格納されています。このテーブルはエンドポイントバッファの位置とサイズを格納しているもので、8 バイト境界に置かれていなければなりません。(アドレスの下位 3 ビットは 0 となります。) このデバイスに対する個々のトランザクションの最初に、USB 回路は対象のエンドポイントに対応するテーブルの要素を読み出し、バッファのアドレスとサイズを得ます。[\(パケット・バッファの構造と役割 \(494 ページ\)\)](#) を参照してください。

Bits 2:0 予約済み、ハードウェアで 0 に固定されています。

20.5.2 エンドポイントに対応するレジスタ

このレジスタの数は USB 回路が扱おうとしているエンドポイントの数によって変ります。USB 回路は最大 8 個の両方向エンドポイントをサポートできます。USB デバイスはエンドポイント番号 (EA フィールド) が 0 に設定されるコントロールエンドポイントをサポートしていかなければなりません。複数のエンドポイントが同じ番号をもって動作状態とされた場合には、この USB 回路の動作は規定されません (このような設定を行ってはいけません)。エンドポイントごとに個別の情報を格納している 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 – 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 リセットが受信されたときや、USB_CNTR レジスタの FRES ビットによりリセットが行われたときには、USB_EPnR レジスタもリセットされます。ただし、リセット直後の正規のパケット通知を見逃さないよう、CTR_RX、CTR_TX ビットはリセットされません。エンドポイントごとに個別の情報を格納している USB_EPnR レジスタが用意されています。

ビットのクリアに読み込み - モディファイ - 書込みサイクルを使用することは、読み込みと書き込みの間にハードウェアによりビットがセットされ、このイベントがマイクロプロセッサで処理されていないにもかかわらず、次の書き込み動作でこのビックをクリアしてしまうことがあるため、好ましくありません。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のために「影響なし」という値が用意されています。ハードウェアでのみ変更されるビットを含むレジスタを変更する場合、これらのビットに対してはビット値「影響なし」を書込むロード命令を使用してください。

Bit 15 CTR_RX: 受信転送完了

このビットは、このエンドポイントに対する OUT/SETUP 処理が正常に終了したとき、ハードウェアによってセットされます。ソフトウェアでは、このビットのクリアのみができます。

USB_CNTR レジスタの CTRM ビットがセットされていれば、マスクできないエンドポイント割込みリクエストとともに、一般割込みリクエストが発生します。処理が OUT か SETUP かは、このレジスタの SETUP ビットで知ることができます。

NAK もしくは STALL ハンドシェイクで終わったトランザクションは、プロトコルエラーもしくはデータトグルの不一致同様、データが実際に転送されていませんので、このビットはセットされません。このビットは読み出しとクリアが可能です。

このビットは 0 を書込むとクリアされます。1 の書き込みはビット値に影響を与えません。

参考資料

Bit 14 DTOG_RX: 受信データトグル

エンドポイントがアイソクロナスでないときには、このビットは次のデータ受信の際のデータトグルビットの値 (0=DATA0, 1=DATA1) を格納しています。PID が一致するデータ受信に続いて ACK ハンドシェイクが USB ホストに送信されたとき、ハードウェアはこのビットを反転します。エンドポイントのタイプがコントロールの時には、このエンドポイントを示す SETUP PID の受信時にハードウェアは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利用している場合には、このビットはバッファの交換にも利用されます。(セクション 20.4.3: ダブルバッファ・エンドポイントを参照)

エンドポイントのタイプがアイソクロナスであれば、データトグルは使用されず、DATA0 パケットのみが送信されるため、このビットはパケット・バッファの交換にのみ使用されます。(セクション 20.4.4: 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、データパケット受信の終わりで、ハードウェアはこのビットを反転します。

このビットは値を初期化するためにソフトウェアで反転することもできます。(コントロールエンドポイント以外は、必ず必要とされます。) もしくは、特定のデータトグル・パケット・バッファの使用に固定します。ソフトウェアで 1 を書込むとビットの値は反転します。0 を書込んだときには、値は変化しません。

Bits 13:12 STAT_RX[1:0]: 受信ステータス・ビット

このビットはエンドポイントのステータスを、表 136: 受信ステータス・エンコーディング (514 ページ) にしたがって格納しています。値を初期化するために、このビットはソフトウェアで (ビットごとに) 反転することができます。ソフトウェアで 1 を書込むとビットの値は反転し、0 を書込むとビット値は変化しません。

このエンドポイントに対する OUT もしくは SETUP トランザクションに対してデータ転送が正常に行われたとき (CTR_RX=1) にハードウェアによって STAT_RX は NAK にセットされます。このため、次のトランザクションで確認応答を返す前に、受信したデータをソフトウェアで確認する時間が得られます。

ダブルバッファのバルク転送では、通常とは異なるフロー制御が行われ、使用可能なバッファに基づいてステータスが制御されます。セクション 20.4.3: 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れば、ステータスは VALID か DISABLED のいずれかで、転送が正常に行われた後、このステータスがハードウェアで変更されることはありません。アイソクロナスエンドポイントで、ソフトウェアでステータスを STALL もしくは NAK にセットしたときの USB 回路の動作は規定されていません。

Bit 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、最後に終了したトランザクションが SETUP であったときに一度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においてのみ値が変ります。受信トランザクションが完了したとき (CTR_RX イベント) には、トランザクションの種類をこのビットで確かめねばなりません。割込み処理ルーチンにとって、次の SETUP トランザクションでビットの値が処理前に変化することを防ぐため、このビットの値は CTR_RX ビットが 1 時間は値が保持され、CTR_RX ビットが 0 になったときに値が変化します。このビットは表示専用です。

Bits 10:9 EP_TYPE[1:0]: エンドポイントタイプ

このビットはエンドポイントの動作を定めるもので、その値は表 137: エンドポイント・タイプ・エンコーディング (514 ページ) のように定められます。エンドポイント 0 はコントロールエンドポイントでなければなりませんが、必要ならば他のエンドポイントもコントロールに設定することができます。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。他のタイプ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は NAK もしくは 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のステータスが NAK の場合には、USB 回路は応答を返さず、転送方向が受信の先には、受信エラーが発生したものとして扱われます。転送方向が受信でステータスが STALL の場合には、SETUP パケットが受けられ、データ転送が行われた後 CTR 割込みリクエストが発生します。OUT トランザクションの受信は、コントロールエンドポイントであっても正常に実行されます。

バルクもしくはインタラプトエンドポイントは、ほとんど同じように動作しますが、EP_KIND ビットで選択され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扱いは、セクション 20.4.4: アイソクロナス転送に記載されています。

参考資料

RM0008

USB フルスピード・デバイス・インターフェース (USB)

Bit 8 EP_KIND: エンドポイント機能選択

このビットはソフトウェアで設定します。ビットの意味は、EP_TYPE フィールドで選択されるエンドポイントタイプによって異なっています。エンドポイントごとの機能は [表 130](#) にあります。DBL_BUF: このビットは、バルクエンドポイントでダブルバッファ機能を使用することを指定します。ダブルバッファバルクエンドポイントの詳細は [セクション 20.4.3: 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サービスされることを指定します。この場合、データ・サイズが 0 ではない OUT トランザクションには ACK ではなく STALL 応答が返されます。このビットはコントロール転送のプロトコレラーの際のアプリケーションプログラムの負担を軽減するために利用することもできます。この利用はコントロールエンドポイントに限られます。このビットがクリアされているときには、必要に応じて、任意のバイト数の OUT トランザクションを扱うことができます。

Bit 7 CTR_TX: 送信転送完了

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終了したとき、ハードウェアによってセットされます。ソフトウェアでは、このビットのクリアのみができます。USB_CNTR レジスタの CTRM ビットがセットされていれば、マスクできないエンドポイント割込みリクエストとともに、一般割込みリクエストが発生します。

NAK もしくは STALL ハンドシェイクで終わったトランザクションは、プロトコレラーもしくはデータトグルの不一致同様、データが実際に転送されていませんので、このビットはセットされません。

このビットは読み出しとクリアが可能です。このビットは 0 を書込むとクリアされます。1 の書き込みはビット値に影響を与えません。

Bit 6 DTOG_TX: 送信データトグル

エンドポイントがアイソクロナスでないときには、このビットは次のデータ送信の際に必要なデータトグルビットの値 (0=DATA0, 1=DATA1) を格納しています。ACK ハンドシェイクが USB ホストから受信されたとき、ハードウェアはこのビットを反転します。エンドポイントのタイプがコントロールの時には、このエンドポイントを示す SETUP PID の受信時にハードウェアはビットを 1 にセットします。

エンドポイントがダブルバッファ機能を利用している場合には、このビットはバッファの交換にも利用されます。(セクション 20.4.3: ダブルバッファ・エンドポイントを参照)

エンドポイントのタイプがアイソクロナスであれば、データトグルは使用されず、DATA0 パケットのみが送信されるため、このビットはパケット・バッファの交換にのみ使用されます。セクション 20.4.4: 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、データパケット受信の終わりで、ハードウェアはこのビットを反転します。

このビットは値を初期化するためにソフトウェアで反転することもできます。(コントロールエンドポイント以外は、必ず必要とされます。) もしくは、特定のデータトグル・パケット・バッファの使用に固定します。ソフトウェアで 1 を書込むとビットの値は反転します。0 を書込んだときには、値は変化しません。

Bits 5:4 STAT_RX [1:0]: 受信ステータス・ビット

このビットはエンドポイントのステータスを、[表 130](#) にしたがって格納しています。値を初期化するために、このビットはソフトウェアで (ビットごとに) 反転することができます。ソフトウェアで 1 を書込むとビットの値は反転し、0 を書込むとビット値は変化しません。このエンドポイントに対する IN もしくは SETUP トランザクションに対してデータ転送が正常に行われたとき (CTR_RX=1) にハードウェアによって STAT_RX は NAK にセットされます。この後、次に送信されるデータがソフトウェアで用意されるのを待ちます。

ダブルバッファのバルク転送では、通常とは異なるフロー制御が行われ、使用可能なバッファに基づいてステータスが制御されます。(セクション 20.4.3: 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れば、ステータスは VALID か DISABLED のいずれかで、転送が正常に行われた後、このステータスがハードウェアで変更されることはありません。アイソクロナスエンドポイントで、ソフトウェアでステータスを STALL もしくは NAK にセットしたときの USB 回路の動作は規定されていません。

ソフトウェアで 1 を書込むとビットの値は反転します。0 を書込んだときには、値は変化しません。



参考資料

Bits 3:0 EA[3:0]: エンドポイントアドレス

これらのビットには、トランザクションがエンドポイントを識別するための 4bit のアドレスを、エンドポイントの動作を開始する前に、ソフトウェアで書込みねばなりません。

表 136. 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 STALL 状態で、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 状態で、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。

表 137. 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	コントロール
10	アイソクロナス
11	インターラプト

表 138. EP_KIND ビットの意味

EP_TYPE[1:0]	EP_KIND ビットの意味	
00	バルク	DBL_BUF
01	コントロール	STATUS_OUT
10	アイソクロナス	使用されません
11	インターラプト	使用されません

表 139. 送信ステータス・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に対する送信は無視されます。
01	STALL: エンドポイントは STALL 状態で、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 状態で、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。

20.5.3 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・メモリ内に置かれていますが、そのエントリは、USB マクロセルと SYM32F10xxx との間でデータ交換をする際に使用するパケット・バッファの、ロケーションとサイズを設定するために使う追加のレジスタとして考えることができます。

共通 APB ブリッジのワードアドレス指定に関する制限により、全てのパケット・メモリ・ロケーションは、実際に USB_BTABLE レジスタ及びバッファディスクリプションテーブル ロケーション用に USB ペリフェラルによって利用されるメモリ・ロケーションアドレスではなく、32bit アラインされたアドレスで APB によりアクセスされます。

次のページには 2 つのロケーションアドレスが記載されています。: 1 つはパケット・メモリのアクセス中に、アプリケーションソフトウェアにより使われます。そして USB ペリフェラルアクセスのための相対的な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・メモリへアクセスする時に使う正しい STM32F10xxx メモリ・アドレスの値を得るために、実際のメモリ・ロケーション・アドレスを 2 倍する必要があります。最初のパケット・メモリ・ロケーションは、0x4000 6000 に位置します。USB_EPnR レジスタに対応するバッファ・ディスクリプタ・テーブル・エントリを下記に記載します。

パケット・バッファとバッファ・ディスクリプタ・テーブルのより詳しい記述は [パケット・バッファの構造と役割 \(494 ページ\)](#) に記載されています。

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 local address: [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	-

Bits 15:1 ADDRn_TX[15:1]: 送信バッファ・アドレス

これらのビットにはパケットバッファの先頭アドレスが格納されます。バッファには USB_EPnR レジスタに対応するエンドポイントが次の IN トークンで送信するデータが書込まれています。

Bit 0 パケット・メモリはワード幅の構成でバッファはワード境界に配置されていなければならぬため、このビットには常に 0 を書き込みます。

参考資料

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 local Address: [USB_BTABLE] + n*8 + 2

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-								COUNTn_TX[9:0]							
								RW	RW	RW	RW	RW	RW	RW	RW

Bits 15:10 USB 仕様によるパケットサイズの上限が 1023 バイトであるため、このビットは使用されません。USB 回路はこれらのビットの値を無視します。

Bits 9:0 COUNTn_TX[9:0]: 送信バイト・カウント

これらのビットには、次の IN トークンで USB_EPnR レジスタに対応するエンドポイントによって送信されるべきデータのバイト数が書込まれています。

注意: ダブルバッファ及びアイソクロナス IN エンドポイントでは、2 個のカウントフィールド USB_COUNTn_TX_0 と USB_COUNTn_TX_1 を使用します。この場合、バイト・カウントワードは次の形式となります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
-								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
-								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 local Address: [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	RW

Bits 15:1 ADDRn_RX[15:1]: 受信バッファ・アドレス

これらのビットにはパケットバッファの先頭アドレスが格納されます。バッファは USB_EPnR レジスタに対応するエンドポイントが次の OUT/SETUP トークンで受信するデータが書込まれます。

Bit 0 パケット・メモリはワード幅の構成でバッファはワード境界に配置されていなければならないため、このビットには常に 0 を書込みます。

参考資料

RM0008

USB フルスピード・デバイス・インターフェース (USB)

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 local Address: [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

このテーブルの場所にはパケット受信時に必要となる 2 種の異なった値を格納しています。最上位ビットはバッファサイズの単位を示しており、バッファ・ドメインにオーバフロー検出を可能としています。下位のフィールドは受信の完了時に USB 回路で書き戻されるもので、実際に受け取ったバイト数を示しています。レジスタサイズの制限から、バッファサイズはブロックサイズを単位として表現されています。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、通常は最初のアドレス付けの段階で maxPacketSize パラメータに基づいて設定されます。
(Universal Serial Bus Specifications を参照してください。)

Bit 15 BL_SIZE: ブロックサイズ

このビットはバッファ・ドメインの指定に使用されるメモリのブロックサイズを選択します。BL_SIZE=0、メモリブロックは 2byte で、これはワード単位で配置する際の最小単位です。この設定で指定できるブロックサイズは 2 ~ 62bytes となります。BL_SIZE=1、メモリブロックは 32byte で、USB 仕様の最大パケットサイズのバッファを指定できます。この設定で指定できるブロックサイズは 32 ~ 1024bytes となります。

Bits 14:10 NUM_BLOCK[4:0]: ブロックの数

このビットはパケット・バッファに割り当てられているメモリブロックの数が書込まれています。実際のメモリのサイズは、[表 140](#) に示されているように、BL_SIZE の値に依存します。

Bits 9:0 COUNTn_RX[9:0]: 受信バイト・カウント

これらのビットには、最後の OUT/SETUP トランザクションで USB_EPnR レジスタに対応するエンドポイントによって受信されたデータのバイト数が書込まれています。

注意: ダブルバッファ及びアイソクロナス OUT エンドポイントでは、2 個のカウントフィールド *USB_COUNTn_RX_0* と *USB_COUNTn_RX_1* を使用します。この場合、バイト・カウントワードは次の形式となります。

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



参考資料

表 140. アロケートされたバッファ・メモリ・サイズの定義

NUM_BLOCK[4:0] の値	BL_SIZE=0 の時のバッファ容量	BL_SIZE=1 の時のバッファ容量
0 (00000b)	設定禁止	32 bytes
1 (00001b)	2 bytes	64 bytes
2 (00010b)	4 bytes	96 bytes
3 (00011b)	6 bytes	128 bytes
...
15 (01111b)	30 bytes	512 bytes
16 (10000b)	32 bytes	N/A
17 (10001b)	34 bytes	N/A
18 (10010b)	36 bytes	N/A
...
30 (11110b)	60 bytes	N/A
31 (11111b)	62 bytes	N/A

20.5.4 USB レジスタ・マップ

次の表は USB のレジスタ・マップとリセット値を示しています。

表 141. USB レジスタ・マップとリセット時の値

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	0	CTR_RX	0	STAT_RX [1:0]	0	SETUP	0	EP_TYPE [1:0]	0	EP_KIND	0	CTR_TX	0	STAT_TX [1:0]	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			
0x04	USB_EP1R	予約済み												CTR_RX	0	CTR_RX	0	STAT_RX [1:0]	0	SETUP	0	EP_TYPE [1:0]	0	EP_KIND	0	CTR_TX	0	STAT_TX [1:0]	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			
0x08	USB_EP2R	予約済み												CTR_RX	0	CTR_RX	0	STAT_RX [1:0]	0	SETUP	0	EP_TYPE [1:0]	0	EP_KIND	0	CTR_TX	0	STAT_TX [1:0]	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			
0x0C	USB_EP3R	予約済み												CTR_RX	0	CTR_RX	0	STAT_RX [1:0]	0	SETUP	0	EP_TYPE [1:0]	0	EP_KIND	0	CTR_TX	0	STAT_TX [1:0]	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			
0x10	USB_EP4R	予約済み												CTR_RX	0	CTR_RX	0	STAT_RX [1:0]	0	SETUP	0	EP_TYPE [1:0]	0	EP_KIND	0	CTR_TX	0	STAT_TX [1:0]	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			
0x14	USB_EP5R	予約済み												CTR_RX	0	CTR_RX	0	STAT_RX [1:0]	0	SETUP	0	EP_TYPE [1:0]	0	EP_KIND	0	CTR_TX	0	STAT_TX [1:0]	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			



参考資料

表 141. USB - レジスタ・マップとリセット時の値 (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	0	CTR_RX	0	STAT_RX [1:0]	0	EP_TYPE [1:0]	0	STAT_TX [1:0]	0	EA[3:0]																	
		リセット値										DTOG_RX	0	DTOG_RX	0	STAT_RX [1:0]	0	EP_TYPE [1:0]	0	STAT_TX [1:0]	0	0 0 0 0 0 0																	
0x1C	USB_EP7R	予約済み										CTR_RX	0	CTR_RX	0	STAT_RX [1:0]	0	EP_TYPE [1:0]	0	STAT_TX [1:0]	0	EA[3:0]																	
		リセット値										DTOG_RX	0	DTOG_RX	0	STAT_RX [1:0]	0	EP_TYPE [1:0]	0	STAT_TX [1:0]	0	0 0 0 0 0 0																	
0x20-0x3F		予約済み																																					
0x40	USB_CNTR	予約済み										CTR	0	CTR	0	PMAOVRM	0	ERRM	0	WKUPM	0	SUSPM	0	RESETM	0	SOFM	0	ESOFM	0	RESUME	0	FSUSP	0	LPMODE	0	PDWN	0	FRES	0
		リセット値										RXDP	0	RXDM	0	LCK	0	ERR	0	WKUP	0	SUSP	0	RESET	0	SOF	0	ESOF	0	DIR	0	EP_ID[3:0]	0	0	0	0	0	0	
0x44	USB_ISTR	予約済み										LSOF	[1:0]	FN[10:0]																									
		リセット値										0	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x			
0x48	USB_FNR	予約済み										予約済み										EF	ADD[6:0]							0	0	0	0	0	0				
		リセット値										0	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x				
0x4C	USB_DADDR	予約済み										予約済み										0	0	0	0	0	0	0	0	0	0	0	0	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				
0x50	USB_BTABLE	予約済み										BTABLE[15:3]															予約済み	0	0	0	0	0	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			

注意： レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。

参考資料

21 CAN (Controller Area Network) (bxCAN)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

21.1 概要

Basic Extended CAN ペリフェラル（略称 **bxCAN**）は CAN ネットワークとのインターフェースで、CAN プロトコル・バージョン 2.0A 及び B をサポートしています。この回路は小さな CPU 負荷で多くの受信メッセージを能率よく取り扱えるように設計されています。また、送信メッセージの優先順位に関するリクエストも満たしています。

安全性が重視される (safety-critical) アプリケーションのために、CAN コントローラは CAN タイムトリガ通信 (CAN Time Triggered Communication) オプションをサポートするハードウェアを備えています。

21.2 bxCAN の主な機能

- CAN プロトコル、バージョン 2.0 A, B サポート
- ビットレート、最大 1 Mbit/s
- タイムトリガ通信オプションのサポート

送信

- 3 個の送信メールボックス
- 設定可能な送信優先順位
- SOF 送信のタイムスタンプ

受信

- 2 個の 3 ステージ受信 FIFO
- 14 個のスケーラブル・フィルタ・バンク /CAN セル - CAN セル間で共用可能
- ID (アイデンティファイア) リスト機能
- FIFO オーバラン時の対応選択
- SOF 受信時のタイムスタンプ

タイムトリガ通信オプション

- 自動再送信の禁止
- 16bit 自走タイマ
- タイマ分解能の選択
- タイムスタンプを最後の 2bytes として送信

通信管理

- マスク可能な割込み機能
- 固有のアドレス空間内の、ソフトウェアの効率のよいメールボックス配置

参考資料

注意： USB と CAN は専用の 512byte SRAM メモリをデータ送受信のために共有しており、両者を同時に使用することはできません。（SRAM はいずれか一方からのアクセスに限られています。）ただし、同時になければ、同じアプリケーションの中で USB と bxCAN の両方を使用することは可能です。

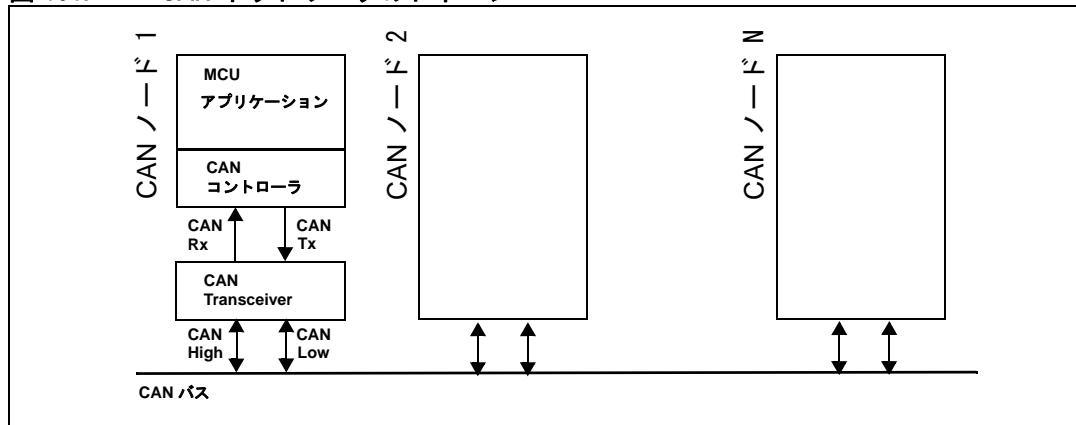
21.2.1 詳細

今日の CAN アプリケーションでは、ネットワークに含まれるノードの数は増加し、時には複数のネットワークがゲートウェイを通して接続されるケースもあります。特にシステム内の（そして各ノードが扱う）メッセージの数が大きく増加しています。また、アプリケーションメッセージのほかに、ネットワーク管理や診断のためのメッセージも導入されています。

- 高度のフィルタ機能が各種のメッセージを扱うために必要とされています。
- 更に、アプリケーションタスクがより多くの CPU タイムを必要としており、メッセージ受信によるリアルタイムへの制約の解消が必要とされています。
- 受信 FIFO の採用は、メッセージを失うことなく、CPU が長い時間アプリケーションタスクに専念することを可能とします。

標準的な CAN ドライバに基づく標準の上位レイヤープロトコル (HLP: Higher Layer Protocol) は CAN コントローラとの効率的なインターフェースを必要としています。

図 191. CAN ネットワークのトポロジ



CAN 2.0B アクティブコア

bxCAN モジュールは CAN メッセージの送受信を自律的に実行することができます。標準 ID (11bit) と拡張 ID (29bit) がハードウェアで完全にサポートされています。

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

制御レジスタ、ステータス・レジスタ、構成レジスタ

アプリケーションでは次の目的でこれらのレジスタを使用します。

- CAN パラメータ (ボーレートなど) の設定
- 送信リクエスト
- 受信処理
- 割込み管理
- 診断情報の入手

送信メールボックス

ソフトウェアでメッセージを組立てるため、3 個のメールボックスが用意されています。いずれのメールボックスから送信されるかは、送信スケジューラが決定します。

受信フィルタ

bxCAN は、着信メッセージからソフトウェアが必要なものを選び出し、他を破棄するためには、14 個のスケーラブルで設定可能な ID フィルタ・バンクを備えています。

受信 FIFO

2 個の受信 FIFO はハードウェアが受け取ったメッセージを格納するために使用されます。各 FIFO には 3 個のメッセージが格納できます。FIFO の管理はすべてハードウェアによって行われます。

図 192. CAN ブロック図

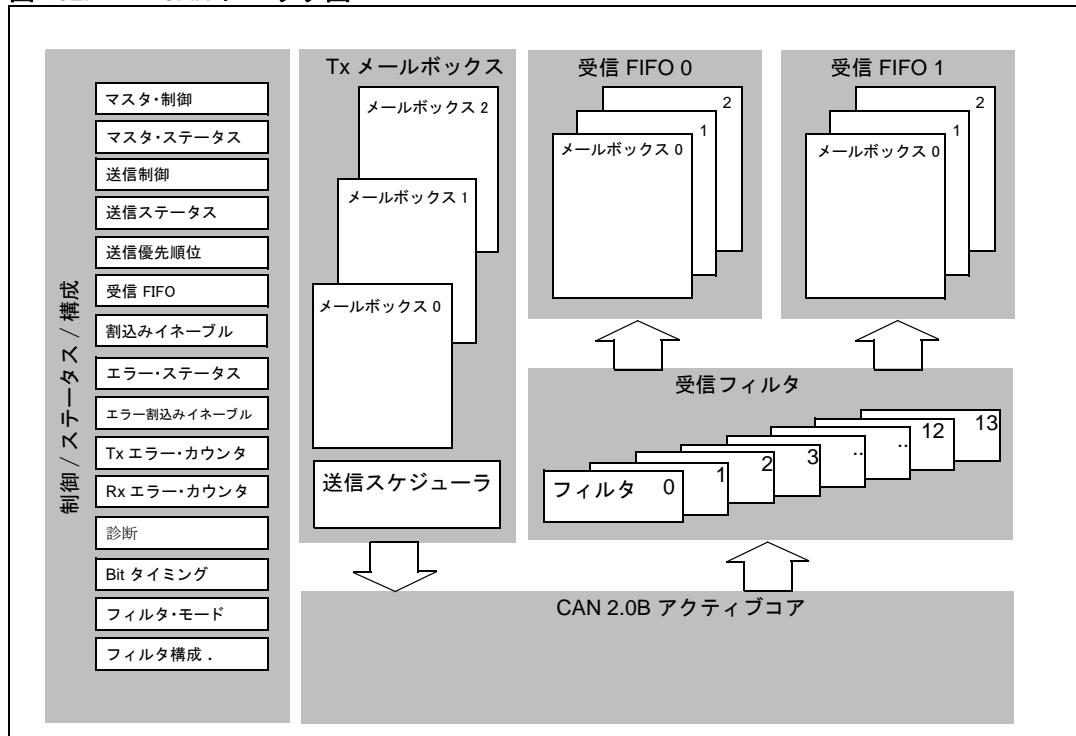
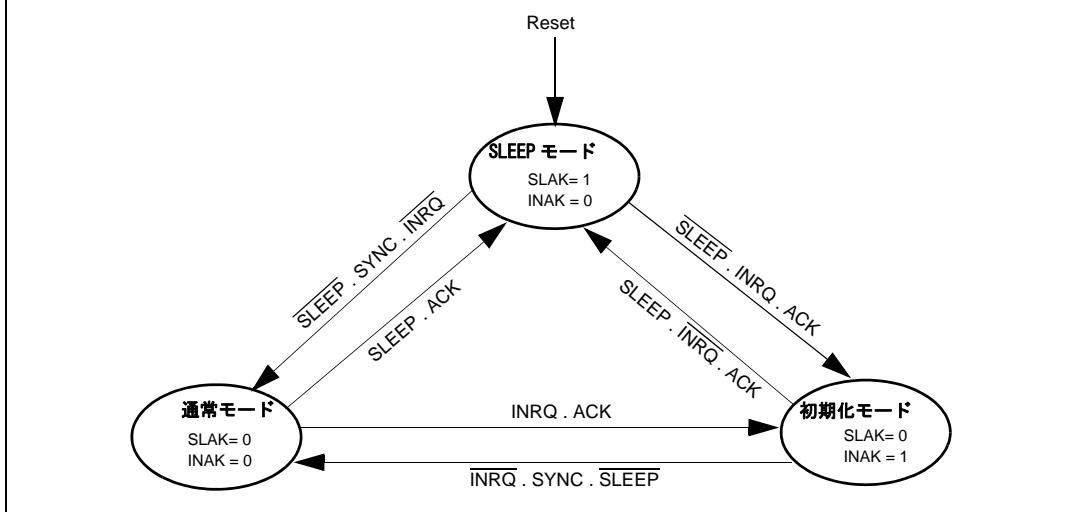


図 193. bxCAN 動作モード



- 注意：
- 1 *ACK* = ハードウェアが CAN_MSR レジスタの INAK もしくは SLAK ビットをセットすることで、指示された状態に実際にに入ったことを示すまでの待ちの状態です。
 - 2 *SYNC* = CAN バスがアイドル状態になる、つまり 11 個の連続したリセッショブ・ビットが CANRX 上で観察されるのを bxCAN が待っている状態です。

21.3 bxCAN の動作モード

bxCAN には初期化 (initialization)、通常 (normal) 及びスリープ (SLEEP) の 3 つの動作モードがあります。リセット後は bxCAN は消費電力を低減し、CANTX の内部のプルアップ抵抗が有効な状態のスリープモードに入っています。ソフトウェアは CAN_MCR レジスタの INRQ もしくは SLEEP ビットをセットすることで、bxCAN が 初期化モードもしくは SLEEP モードに入るよう指示します。bxCAN は、CAN_MSR レジスタの INAK もしくは SLAK ビットをセットし、その動作モードに入ったことを表示し、同時に内蔵されているプルアップ抵抗を無効とします。INAK と SLAK のいずれもがセットされていないときには、bxCAN は通常モードです。bxCAN は通常モードに入る前に、CAN バスと同期をとります。このためには bxCAN は、CAN バスがアイドル状態になる、つまり 11 個の連続したリセッショブ・ビットが CANRX 上で観察されるのを待ちます。

21.3.1 初期化モード

ハードウェアが初期化モードにある間に、ソフトウェアの初期化も行うことができます。このモードに入るには、ソフトウェアは CAN_MCR レジスタの INRQ ビットをセットし、ハードウェアがこれに答える形で CAN_MSR レジスタの INAK ビットをセットするのを待ちます。

初期化モードを離れるには、ソフトウェアで INRQ ビットをクリアします。INAK ビットがクリアされた時点で、bxCAN は初期化モードを終了します。

初期化モードでは CAN バスとの間のすべてのメッセージ転送は停止し、CAN バス出力 CANTX 信号はリセッショブ（ハイレベル）状態となっています。

初期化モードに入る際には、設定などに関するレジスタの内容は変化しません。

CAN コントローラを初期化するには、ソフトウェアで ビット・タイミング (CAN_BTR) と CAN オプション (CAN_MCR) の設定を行わねばなりません。

CAN フィルタ・バンクに関するレジスタ（モード、スケール、FIFO 配置・有効化とフィルタの値）の初期化には、ソフトウェアで FINIT ビット (CAN_FMR) をセットしなければなりません。フィルタの初期化は、初期化モード以外でも行うことができます。

- 注意 : FINIT=1 のとき、CAN 受信機能は動作していません。
フィルタの値は CAN_FA0R レジスタの対応するフィルタアクティブビットをクリアすることでも変更できます。
フィルタ・バンクが使用されていない場合、これらをアクティブではない状態（対応する FACT ビットをクリア）としておくことが勧められます。

21.3.2 通常モード

初期化が終了すると、ハードウェアを、CAN バスに同期し送受信動作を開始するよう、通常モードに入る指示を出さねばなりません。通常モードに入るには CAN_MCR レジスタの INRQ ビットをクリアし、CAN_MSR レジスタの INAK ビットがクリアされるのを待ちます。その後 bxCAN は、バス動作に加わりメッセージ転送を行う前に、11 個の連続したリセッショブ・ビットの発生（バスアイドル状態）を待ち、CAN バス上のデータ転送動作にタイミングを合わせます。

フィルタの値の初期化は初期化モードとは別に行われますが、フィルタが有効となっていない（フィルタに対応する FACTx ビットがクリアされている）ときに行わねばなりません。フィルタのスケールとモードは通常モードに入る前に設定しておかねばなりません。

21.3.3 SLEEP モード（低消費電力）

消費電力を節約するため、bxCAN は SLEEP モードと呼ばれる低電力モードを備えています。このモードに入るにはソフトウェアで CAN_MCR レジスタの SLEEP ビットをセットします。このモードでは bxCAN のクロックは停止しますが、ソフトウェアから bxCAN のメールボックスにアクセスすることは可能です。

bxCAN が SLEEP モードのときにソフトウェアが INRQ ビットをセットして初期化モードに入ることを指示するには、同時に SLEEP ビットをクリアしなければなりません。

ソフトウェアで SLEEP ビットをクリアしたとき、または CAN バス上で動作状態が検出されたときには、bxCAN は SLEEP モードを終了し、ウェークアップします。

CAN バス動作の検出では、CAN_MCR レジスタの AWUM ビットがセットされれば、ハードウェアは SLEEP ビットをクリアし、ウェークアップを実行します。AWUM ビットがクリアされているときには、SLEEP モードを抜け出す代わりにウェークアップ割込みリクエストが発生します。

- 注意 : bxCAN が自動的にウェークアップ動作を行う場合でも、ウェークアップ割込みリクエストが許可状態（CAN_IER レジスタの WKUIE ビットがセットされている）にあれば CAN 動作が検出されたときにウェークアップ割込みリクエストが発生します。

図 193: bxCAN 動作モード にあるように、SLEEP ビットがクリアされた後、bxCAN が CAN バスに同期すると SLEEP モードを終了します。実際に SLEEP モードから抜け出すのは、ハードウェアが SLAK ビットをクリアしたときです。

21.3.4 テストモード

テストモードは CAN_BTR レジスタの SLM ビットと LBKM ビットで指定できます。これらのビットの設定は、bxCAN が初期化モードの時に限られます。テストモードが選択された後に通常モードに戻るには、CAN_MCR レジスタの INRQ ビットをクリアせねばなりません。

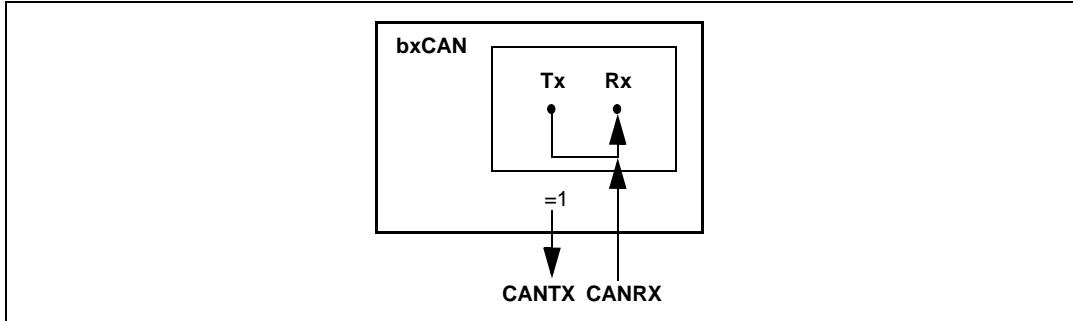
参考資料

21.3.5 サイレント・モード

bxCAN は CAN_BTR レジスタの SILEM ビットをセットすることでサイレント・モードとなります。

サイレント・モードでは、bxCAN は有効なデータフレームやリモートフレームを受信することができますが、送信動作を開始することはできません。bxCAN がドミナント・ビットを送出する必要があるとき（ACK ビット、オーバロードフラグ、アクティブ・エラー・フラグ）には、CAN コアは内部的にこのドミナント・ビットを見ることができますが、CAN バスではリセッショブ状態が保たれます。サイレント・モードは、ドミナント・ビットの送出（ACK ビット、エラーフレーム）で CAN バスに影響を与えることなく、CAN バス上のトランザクションを解析するために利用されます。

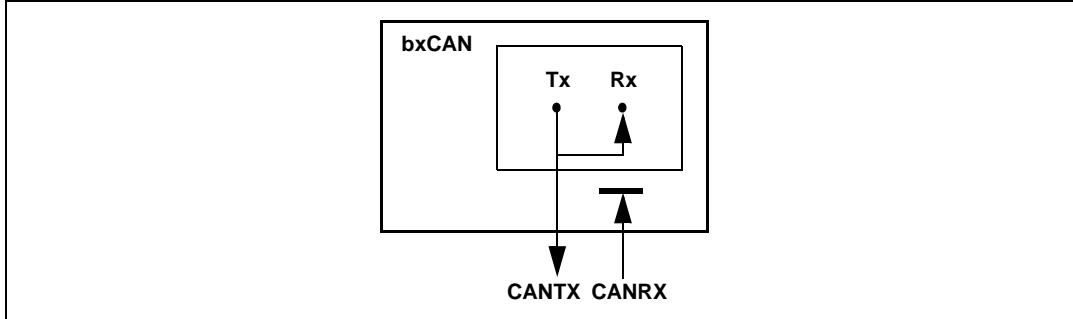
図 194. bxCAN サイレント・モード



21.3.6 ループバック・モード

bxCAN は CAN_BTR レジスタの LBKM ビットをセットすることでループバック・モードとなります。ループバック・モードでは、bxCAN は自己の送信メッセージを受信メッセージとして扱い、受信フィルタを通過したときには受信メールボックスに格納します。

図 195. bxCAN ループバック・モード

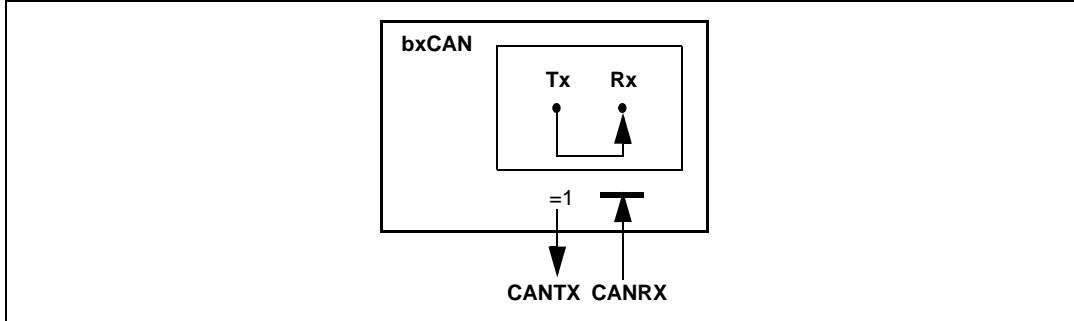


このモードは自己テスト機能のために用意されています。外部のイベントから独立させるために、ループバック・モードで CAN コアはデータフレーム / リモートフレームの ACK スロットにおけるドミナント・ビットのサンプルを行わないことで、ACK エラーを無視します。このモードでは bxCAN は内部で Tx 出力から Rx 入力にフィードバックしています。CANRX ピンの実際の値は bxCAN では無視されます。送信メッセージは CANTX ピンに送出されます。

21.3.7 サイレント・モードとループバック・モードの組合せ

CAN_BTR レジスタの LBKM ビットと SILEM ビットの両方をセットすることで、ループバック・モードとサイレント・モードを組合せることができます。このモードは “Hot Selftest” で使用されるもので、bxCAN は CANTX、CANRX ピンに接続されている CAN システムに影響を与えるずにループバック・モードと同様のテストを行うことができます。このモードでは CANRX ピンは bxCAN から切離されており、CANTX ピンはリセシシブ状態に固定されています。

図 196. bxCAN 組合せモード



21.4 bxCAN の機能詳細

21.4.1 送信動作

メッセージを送信するには、アプリケーションでは、EMPTY 状態のメールボックスを一個選び、ID やデータ長コード (DLC) 及びデータを設定し、最後に CAN_TTxR レジスタの TXRQ ビットをセットして送信を指示します。メールボックスが EMPTY 状態を離れると、ソフトウェアはこのメールボックス・レジスタへの書き込みができなくなります。TXRQ ビットがリセットされると、メールボックスは PENDING 状態となり、メールボックスの間で最も優先順位が高くなるのを待ちます。メールボックスの優先順位が最も高くなると、メールボックスは SCHEDULED 状態となります。SCHEDULED 状態のメールボックスの送信は CAN バスがアイドル状態となると開始されます。(TRANSMIT 状態に入ります。) メールボックスの内容が正常に送信されると、そのメールボックスは再び EMPTY 状態となります。ハードウェアは CAN_TSR レジスタの PQCP ビットと TXOK ビットをセットし、送信が正常に終了したことを表示します。

送信が失敗すると、送信権喪失の場合には CAN_TSR レジスタの ALST ビットで、送出エラーの場合には TERR ビットにより、送信が正常に行われなかったことを示します。

送信優先順位

ID による優先順位

複数の送信メールボックスが PENDING 状態にあるとき、送信される順序はメールボックスに格納されている ID によって決められます。CAN プロトコルのアービトレーションの方式の結果として、最も小さな ID 値を持つメッセージの優先順位が最も高くなります。ID の値が同じであれば、より小さなメールボックス番号のメールボックスの送信が優先されます。

送信リクエストの順序による優先順位

送信メールボックスは CAN_MCR レジスタの TXFP ビットがセットされると、送信 FIFO におかれます。この状態では、優先順位は送信指示が発せられた順となります。

このモードは分割送信の場合に役立ちます。

アポート

ユーザは、CAN_TSR レジスタの ABRQ ビットをセットすることで、送信指示をアポートすることができます。PENDING もしくは SCHEDULED 状態のときには、メールボックスのアポートは直ちに行われます。メールボックスが送信状態のときにアポート指示が出されると、次の二者のいずれかの結果となります。メールボックスの送信が正常に行われたときには、CAN_TSR レジスタの TXOK ビットがセットされ、メールボックスは EMPTY 状態となります。送信に異常が生じ、メールボックスが SCHEDULED 状態になると送信動作はアポートされ、TXOK ビットがクリアされた状態でメールボックスは EMPTY 状態となります。いずれの場合にも、進行中の送出動作が終了すると、少なくともそのメールボックスは EMPTY 状態となります。

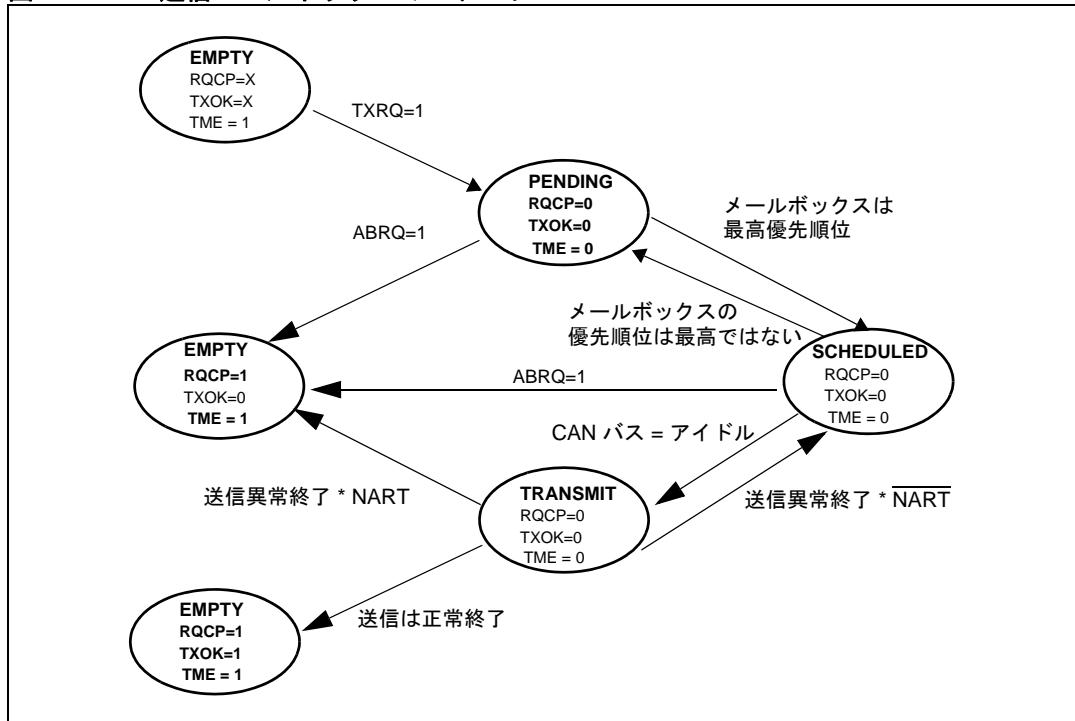
自動再送信の禁止

このモードは CAN 規格のタイムトリガ通信 (Time Triggered Communication) オプションを完全に満たすために組み込まれています。ハードウェアをこのモードに設定するには CAN_MCR レジスタの NART ビットをセットします。

このモードでは、個々の送出動作は一度だけ行われます。最初の送出動作が送信権喪失やエラーのために正常に終了しなかったときには、ハードウェアによる自動最送出は行われません。

送出を最初に試みた後、ハードウェアは送信指示が完了したものとして CAN_TSR レジスタの TQCP ビットをセットします。送信動作の結果は CAN_TSR レジスタの TXOK、ALST、TERR のビットで表示されます。

図 197. 送信メールボックスのステート



21.4.2 タイムトリガ通信モード

このモードでは CAN ハードウェアに内蔵されたカウンタが有効となり、受信メールボックス / 送信メールボックスの CAN_RDTxR/CAN_TDTxR レジスタのタイムスタンプ値が生成されます。送受信ともに、カウンタの値はフレーム開始 (SOF) ビットのサンプルポイントで読出されます。

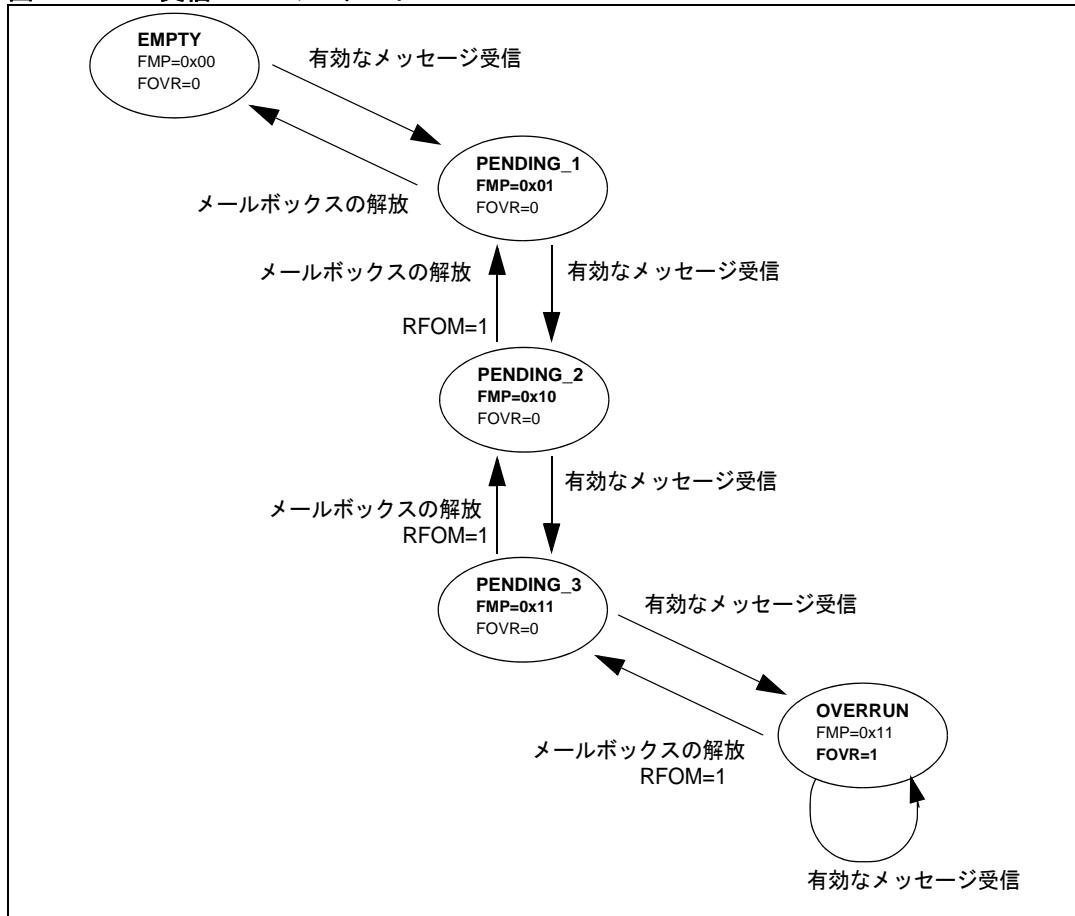
21.4.3 受信動作

CAN メッセージの受信では、3 個のメールボックスが FIFO の形で用意されています。CPU の負荷の軽減、ソフトウェアの単純化、データの一貫性の保証のため、FIFO の管理はハードウェアで行われます。アプリケーションからは、FIFO の出力メールボックスを通して FIFO に格納されているメッセージにアクセスします。

有効なメッセージ

受信されたメッセージは CAN プロトコル (no error until the last but one bit of the EOF field) に準じて正しく受信され、ID フィルタリングに該当したときに、有効なメッセージとみなされます。ID フィルタリングに関しては [セクション 21.4.4: ID によるフィルタリング](#)をお読みください。

図 198. 受信 FIFO のステート



FIFO 管理

EMPTY 状態の後、受信された有効なメッセージが最初に FIFO に格納されると PENDING_1 状態となります。ハードウェアは CRFR レジスタの FMP[1:0] フィールドに 01b をセットし、この変化を表示します。メッセージは FIFO 出力メールボックスから読出することができます。ソフトウェアはこのメールボックスの内容を読み出し、CRFR レジスタの RFOM ビットをセットすることで、このメールボックスの解放を指示します。このとき、FIFO は再び EMPTY 状態となります。この間に新たに有効なメッセージが受信されると、FIFO は PENDING_1 状態にとどまり、新しいメッセージが出力メールボックスからアクセスできるようになります。

アプリケーションがメールボックスを解放していなければ、次の有効なメッセージが FIFO に格納されると、FIFO は PENDING_2 状態 (FMP[1:0] = 10b) となります。次の有効メッセージに対しても格納動作は繰り返され、FIFO は PENDING_3 (FMP[1:0] = 11b) 状態となります。この時点ではソフトウェアは RFOM ビットをセットし出力メールボックスを解放しなければなりません。これにより、メールボックスは未使用状態となり、次のメッセージを格納することが可能となります。これが行われないときには、受信された次の有効メッセージは失われてしまいます。

[セクション 21.4.5: メッセージ・メモリ（メールボックス）](#) をあわせてお読みください。

オーバラン

FIFO が PENDING_3 状態 (3 個のメッセージボックスが満杯) のとき、次の有効メッセージが受信されると OVERRUN 状態が発生し、このメッセージは失われます。ハードウェアは CRFR レジスタの FOVR ビットをセットすることで、オーバラン状態の発生を通知します。失われるメッセージは、FIFO の設定に依存します。

- FIFO ロックが無効 (CAN_MCR レジスタの RFLM ビットがクリア) であれば、最後に FIFO に格納されたメッセージが新たに受信されたメッセージで上書きされます。このケースでは、アプリケーションは常に最後に受信されたメッセージを得ることができます。
- FIFO ロックが有効 (CAN_MCR レジスタの RFLM ビットがセット) であれば、最新のメッセージが破棄され、ソフトウェアからは FIFO 内にある 3 個の古いメッセージにアクセスできます。

受信関連の割込み

メッセージが FIFO に格納されると、FMP[1:0] の値が更新され、CAN_IER レジスタの FMPIE ビットがセットされていれば割込みリクエストが発生します。

FIFO がいっぱいになったとき (3 個目のメッセージが格納されたとき)、CRFR レジスタの FULL ビットがセットされ、CAN_IER レジスタの FFIE ビットがセットされれば割込みリクエストが発生します。

オーバラン状態が発生したとき FOVR ビットがセットされ、CAN_IER レジスタの FOVIE ビットがセットされれば、割込みリクエストが発生します。

21.4.4 ID によるフィルタリング

CAN プロトコルではメッセージの識別コード (ID) はノードのアドレスを指定するものではなく、メッセージの内容に関連したものとなっています。また、送信回路はメッセージをすべての受信回路に対して送信します。メッセージが受信されたとき、受信ノードはメッセージの識別コード (ID) に基づいて、このメッセージが自ノードに必要なものか否かを見定めます。ソフトウェアの介入なしで、このメッセージが必要であれば SRAM にコピーされ、必要なないメッセージは破棄されます。

この目的のため、bxCAN コントローラには 14 個の構成可能・スケーラブルなフィルタ・バンク (13-0) が用意されており、ソフトウェアが必要としているメッセージのみを受信します。このハードウェアフィルタリング機能は、ソフトウェアでフィルタリングを行った場合に必要とされる CPU の負荷を軽減します。個々のフィルタ・バンク x は 2 個の 32bit レジスタ CAN_FxR0 及び CAN_FxR1 から構成されています。

フィルタ幅の選択

フィルタをアプリケーションの必要に応じた無駄のないものとするため、個々のフィルタ・バンクは個別にスケールを決めることができます。フィルタのスケールに応じて、各フィルタ・バンクは次のように構成されます。

- 1 個の 32bit フィルタ、STDID[10:0]、EXTID[17:0]、IDE、RTR ビットを有します。
- 2 個の 16bit フィルタ、STDID[10:0]、RTR、IDE、EXTID[17:15] ビットを有します。

[図 199](#) を参照してください。

さらに、フィルタはマスクモード、ID リストモードのいずれかのモードとすることができます。

マスクモード

マスクモードでは、ID のどのビットが一致させるか、どのビットが比較されないかを定めるマスク・レジスタと ID レジスタが対応付けられています。

ID リストモード

ID リストモードでは、マスク・レジスタが ID レジスタとして使用されます。フィルタとして ID とマスクを指定する代わりに、2 個の ID を指定することができ、設定できる ID の数を倍にしています。受信メッセージの全 ID ビットがフィルタ・レジスタで指定された値と一致する必要があります。

フィルタ・バンクの大きさとモード設定

フィルタ・バンクは対応する CFMR レジスタ (CAN_FM0R レジスタ) によって構成が決まります。フィルタ・バンクを構成する際には、CAN_FAR レジスタの FACT ビットをクリアし、フィルタが無効の状態としておかなければなりません。フィルタのスケールは [図 199](#) にあるように CFSCR レジスタ (CAN_FS0R レジスタ) の対応する FSC x ビットにより指定されます。マスク /ID レジスタが ID リストモードか ID マスクモードかの選択は CFMR レジスタ (CAN_FM0R レジスタ) の対応する FBM x ビットによって行われます。

一群の ID のフィルタとして働くには、マスク /ID レジスタをマスクモードに設定します。

個別の ID を指定するには、マスク /ID レジスタを ID リストモードに設定します。

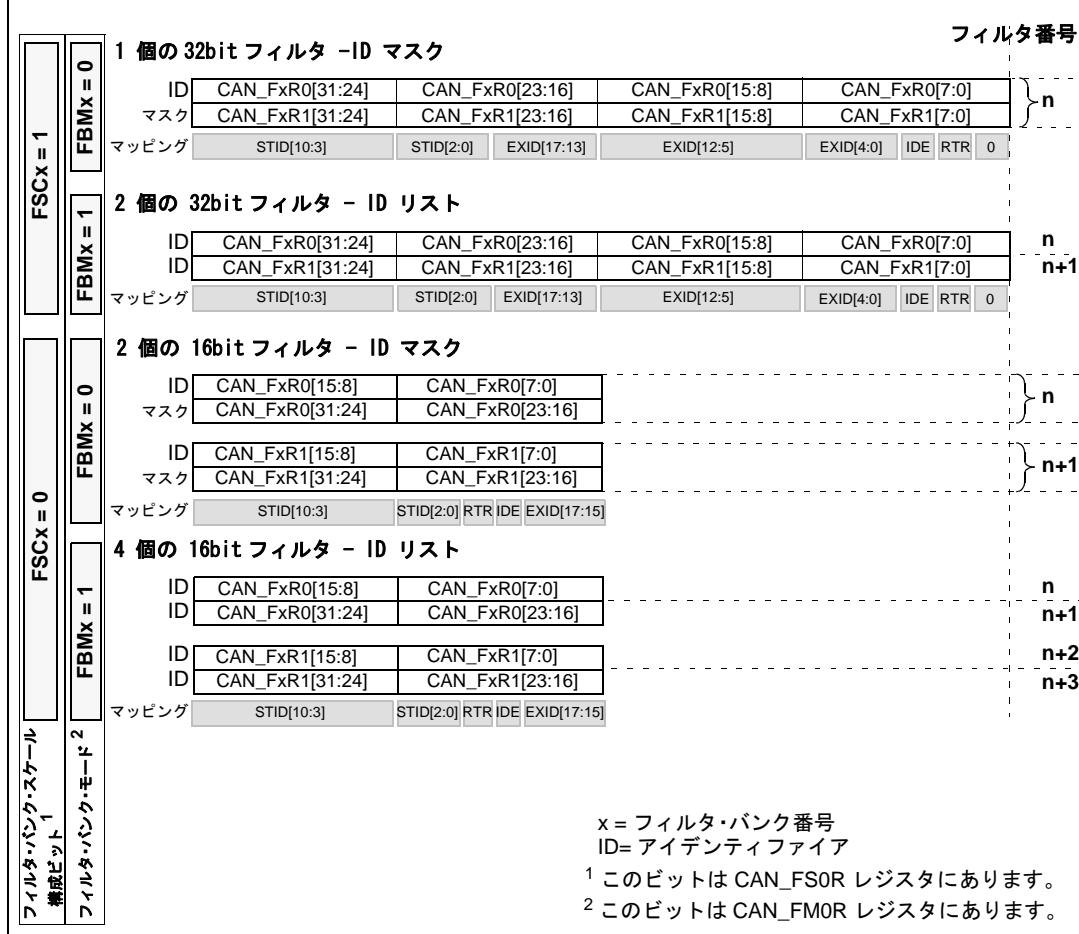
アプリケーションで使用されないフィルタは無効状態としておきます。

フィルタ・バンクの各フィルタには 0 から、14 個のフィルタ・バンクのそれぞれで設定されているモードとスケールに応じた最大値までの番号 (フィルタ番号) が付されています。

フィルタの構成に関しては [図 199](#) を参照してください。

参考資料

図 199. フィルタ・バンクのスケール設定 - レジスタ構成



該当フィルタ索引

メッセージが FIFO 内に受信されると、アプリケーションからアクセス可能となります。通常、アプリケーションのデータは SRAM のいずれかの位置にコピーされますが、データをアプリケーションの正しい位置にコピーするためには ID の値によりデータを識別しなければなりません。この操作を省き、SRAM 上の場所へのアクセスを容易にするために、CAN コントローラは該当フィルタ索引 (Filter Match Index) を搭載しています。

この索引はフィルタ優先規則に応じて、メールボックスにメッセージとともに格納されます。受信されたメッセージは個々に該当フィルタ索引を搭載しています。

該当フィルタ索引は次の二通りに使用されます。

- 該当フィルタ索引の値を、希望する値のリストと比較します。
- 該当フィルタ索引の値を、データを格納する場所を示す配列の添字として利用します。

マスクモードではないフィルタに関しては、ソフトウェアでこれ以上の ID 比較を行う必要はありません。

マスクモードのフィルタでは、シフトウェアの手間は、マスクされたビットの比較に限られます。

フィルタ番号の索引値はフィルタ・バンクの有効化の影響を受けません。さらに、個々の FIFO ごとに独立した番号付けを行うことができます。図 200 には、その一例が示されています。

参考資料

図 200. フィルタ番号の例

フィルタ バンク	FIFO0	フィルタ 番号 .	フィルタ バンク	FIFO1	フィルタ 番号
0	ID リスト (32bit)	0 1	2	ID マスク (16bit)	0 1
1	ID マスク (32bit)	2	4	ID リスト (32bit)	2 3
3	ID リスト (16bit)	3 4 5 6	7	無効化されている ID マスク (16bit)	4 5
5	無効化されている ID リスト (32bit)	7 8	8	ID マスク (16bit)	6 7
6	ID マスク (16bit)	9 10	10	無効化されている ID リスト (16bit)	8 9 10 11
9	ID リスト (32bit)	11 12	11	ID リスト (32bit)	12 13
13	ID マスク (32bit)	13	12	ID マスク (32bit)	14

ID=Identifier

フィルタ優先順位の規則

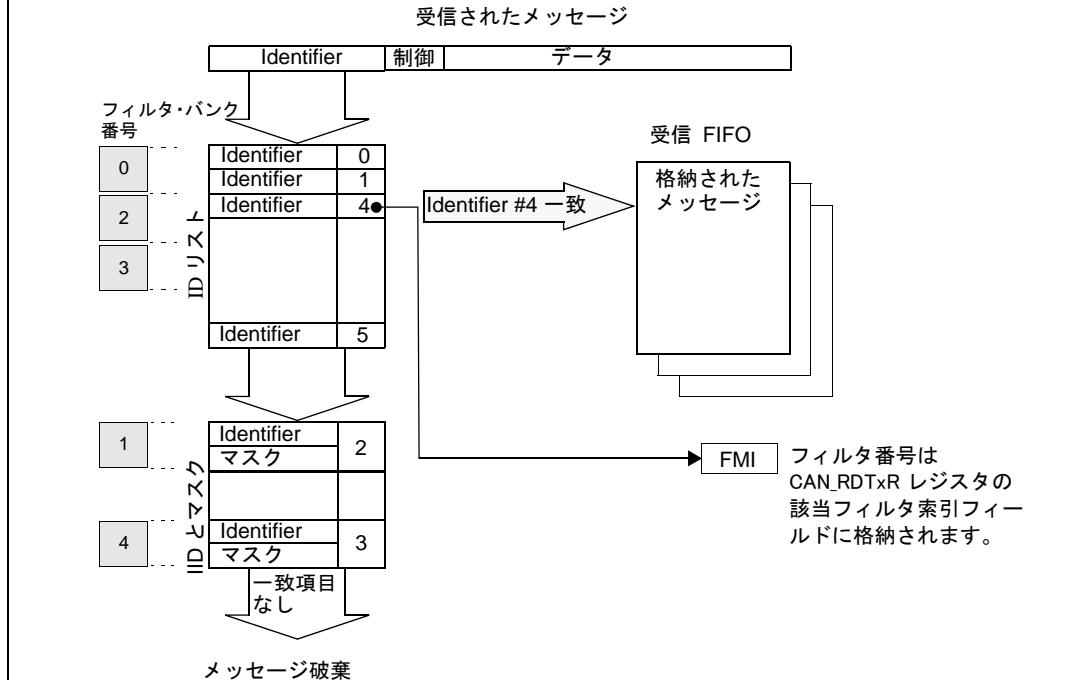
フィルタの組み合わせに応じて、ひとつの ID が複数のフィルタを通過できることがあります。このとき、受信メールボックスに格納される番号は次のように決められます。

- 32bit フィルタが 16bit フィルタより優先します。
- 同じスケールのフィルタでは、ID リストモードが ID マスクモードより優先します。
- スケールとモードが同じフィルタでは、フィルタ番号のより小さなフィルタが優先します。

参考資料

図 201. フィルタのメカニズムー例

32bit ID リストモード・フィルタ 3 フィルタ・バンクと
他を 32bit ID マスクモード・フィルタとした場合の例



上の例は bxCAN のフィルタリングの基本を示しています。メッセージが受信されると、ID はまず ID リストモードのフィルタと比較されます。一致するものがあれば、メッセージはその FIFO に格納され、一致したフィルタの索引は該当フィルタ索引フィールドに格納されます。例では、ID は ID #2 と一致し、FMI には 2 がセットされ、メッセージの本体が FIFO に格納されます。

一致する項目がなければ、受信メッセージの ID は次にマスクモードのフィルタと比較されます。

ID がいずれのモードのフィルタにも一致しなければ、そのメッセージはハードウェアにより破棄され、ソフトウェアが介入する必要はありません。

21.4.5 メッセージ・メモリ（メールボックス）

ソフトウェアとハードウェアの間で CAN メッセージを受渡すインターフェースはメールボックスという形式で実現されています。メールボックスには、ID、データ、制御、ステータス、タイムスタンプといった、メッセージに関するすべての情報が格納されます。

送信メールボックス

ソフトウェアでは送信しようとするメッセージをからの送信メールボックスにセットアップします。送信時のステータスはハードウェアによって CAN_TSR レジスタに書込まれます。

表 142. 送信メールボックスのマッピング

送信メールボックス内のオフセット	レジスタ名
0	CAN_TIxR
4	CAN_TDTxR
8	CAN_TDLxR
12	CAN_TDhxR

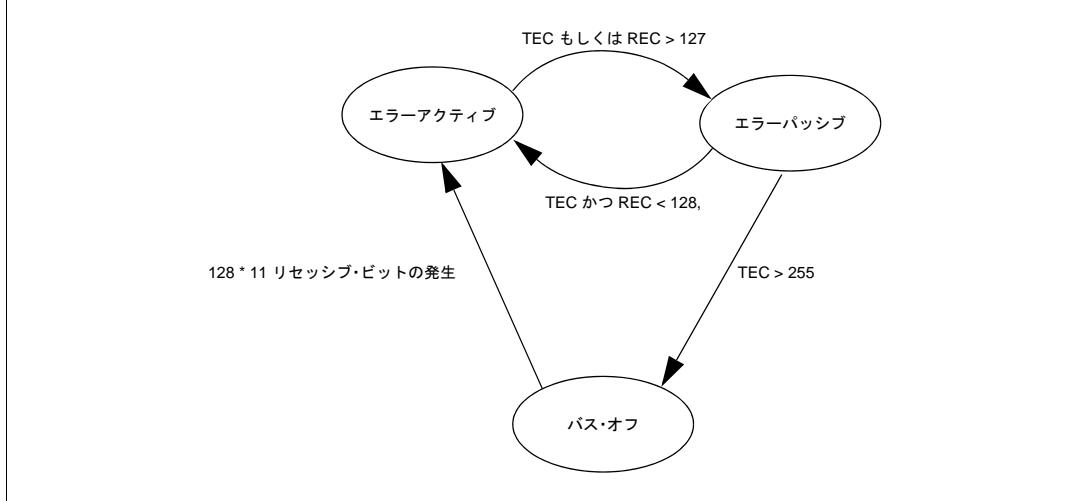
受信メールボックス

メッセージが受信されると、ソフトウェアは FIFO 出力メールボックスを通してこれにアクセスすることができます。ソフトウェアでのメッセージの読み込みなどの処理が終わると、ソフトウェアは CRFR レジスタ (CAN_RFxR レジスタ) の RFOM ビットを使用して FIFO 出力メールボックスを解放し、次の受信メッセージにアクセスできるようになります。該当フィルタ索引は CAN_RDTxR レジスタの MFMI フィールドに格納され、16 ビットタイムスタンプの値は CAN_RDTxR レジスタの TIME[15:0] フィールドに格納されます。

表 143. 受信メールボックスのマッピング

受信メールボックス内のオフセット	レジスタ名
0	CAN_RIxR
4	CAN_RDTxR
8	CAN_RDLxR
12	CAN_RDhxR

図 202. CAN エラー状態図



21.4.6 エラー管理

CAN プロトコルに記述されているエラー管理は、エラーの状況によってカウントアップ / ダウンされる送信エラーカウンタ (CAN_ESR レジスタの TEC 値) と受信エラーカウンタ (CAN_ESR レジスタの REC 値) を使用して、すべてハードウェアで行われます。TEC 及び REC の管理の詳細に関しては、CAN 規格を参照してください。

両カウンタはバスの安定性を確認するため、ソフトウェアで読み出すことができます。さらに、CAN ハードウェアはその都度 CAN_ESR レジスタに詳細なエラーステータスをセットします。CAN_IER の (ERRIE ビットなどの) 設定に応じて、指定した種類のエラーが検出されたときに割込みリクエストを発生させることができます。

バス・オフからの回復

TEC が 255 より大きくなるとバス・オフ状態となります。この状態は CAN_ESR レジスタの BOFF ビットで示されます。バス・オフ状態では bxCAN はメッセージの送受信を行うことができません。

CAN_MCR レジスタの ABOM ビットに応じて、bxCAN はバス・オフからの（エラーアクティブ状態への）回復を、自動的に、もしくはソフトウェアの指示で実行します。いずれの場合でも bxCAN は CAN 規格で定められた回復手順 (CANRX 上で 11 個の連続したりセッショップ・ビットの状態が 128 回出現) を待たねばなりません。

ABOM がセットされていれば、バス・オフ状態に入った後、bxCAN は自動的に回復手順を開始します。

ABOM がクリアされているときには、bxCAN が一旦初期化モードに入り、さらにここから抜け出すことをソフトウェアで指示し、回復手順を開始させねばなりません。

注意:

初期化モードでは bxCAN は CANRX 信号の監視は行いません。このため、この間は回復手順を利用することはできません。状態の復帰には、bxCAN は通常モードでなければなりません。

21.4.7 ビット・タイミング

ビット・タイミング回路はシリアルバスラインを監視し、ビットのサンプリング動作と、スタート・ビットのエッジによる同期と続くエッジによる再同期で、サンプリングポイントの調整を行います。

この動作を、ビットタイムを3セグメントに分けて簡単に説明します。

- 同期セグメント (SYNC_SEG): ビット変化がこの時間内に生じなければなりません。この時間の長さは1時間単位 (quantum) ($1 \times t_{CAN}$) に固定されています。
- ビットセグメント 1 (BS1): サンプルポイントの位置を定めます。この部分には CAN 規格の PROP_SEG と PHASE_SEG1 が含まれます。セグメントの時間は1から16時間単位の間でプログラムによる設定が可能で、ネットワークのノード間の周波数の相違による正の位相ドリフトを補正するために自動的に延長されます。
- ビットセグメント 2 (BS2): 送信ポイントの位置を定めます。この部分は CAN 規格の PHASE_SEG2 に相当します。セグメントの時間は1から8時間単位の間でプログラムによる設定が可能で、負の位相ドリフトを補正するため自動的に短縮されます。

再同期ジャンプ幅 (SJW) はビットセグメントを延長 / 短縮する際の上限を定めており、プログラムで1から4時間単位の間で設定します。

有効なエッジは、コントローラ自体はリセシブ・ビットを送出していない状態で、ビットタイムの中の、ドミナントからリセシブへのバスレベルの最初の変化と定義されます。

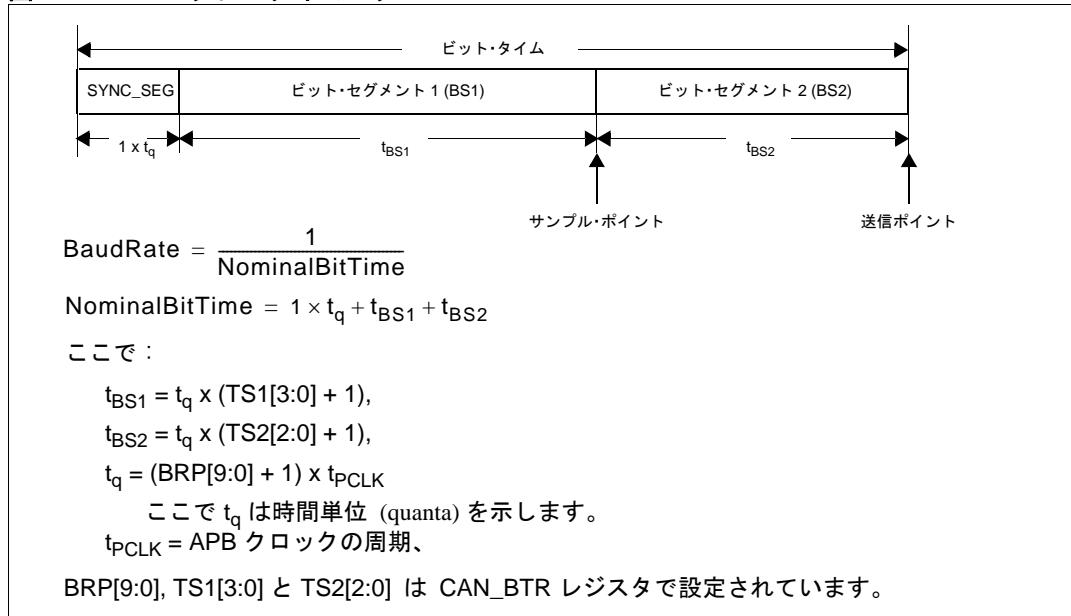
SYNC_SEG ではなく BS1において有効なエッジが検出されたときには、BS1は最大で SJW だけ延長され、結果としてサンプルポイントは遅くなります。

これに対して、SYNC_SEG ではなく BS2において有効なエッジが検出されたときには、BS2は最大で SJW だけ短縮され、結果として送信ポイントは早くなります。

プログラミングの誤りに対する安全策として、ビット・タイミング・レジスタ (CAN_BTR) は STANDBY モードの時にのみ設定ができます。

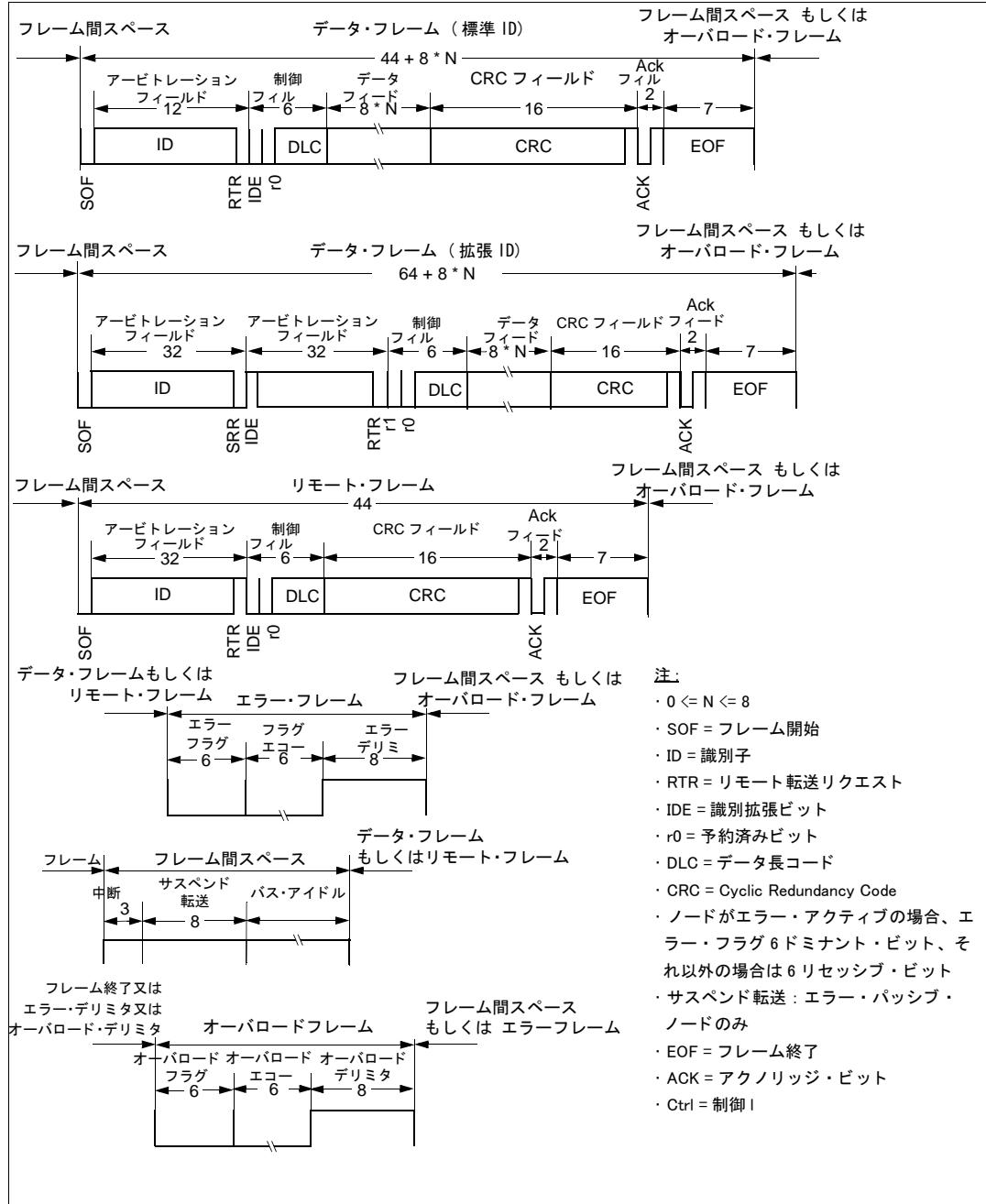
注意: CAN ビット・タイミングと再同期のメカニズムの詳細に関しては ISO11898 規格を参照してください。

図 203. ビット・タイミング



参考資料

図 204. CAN フレーム

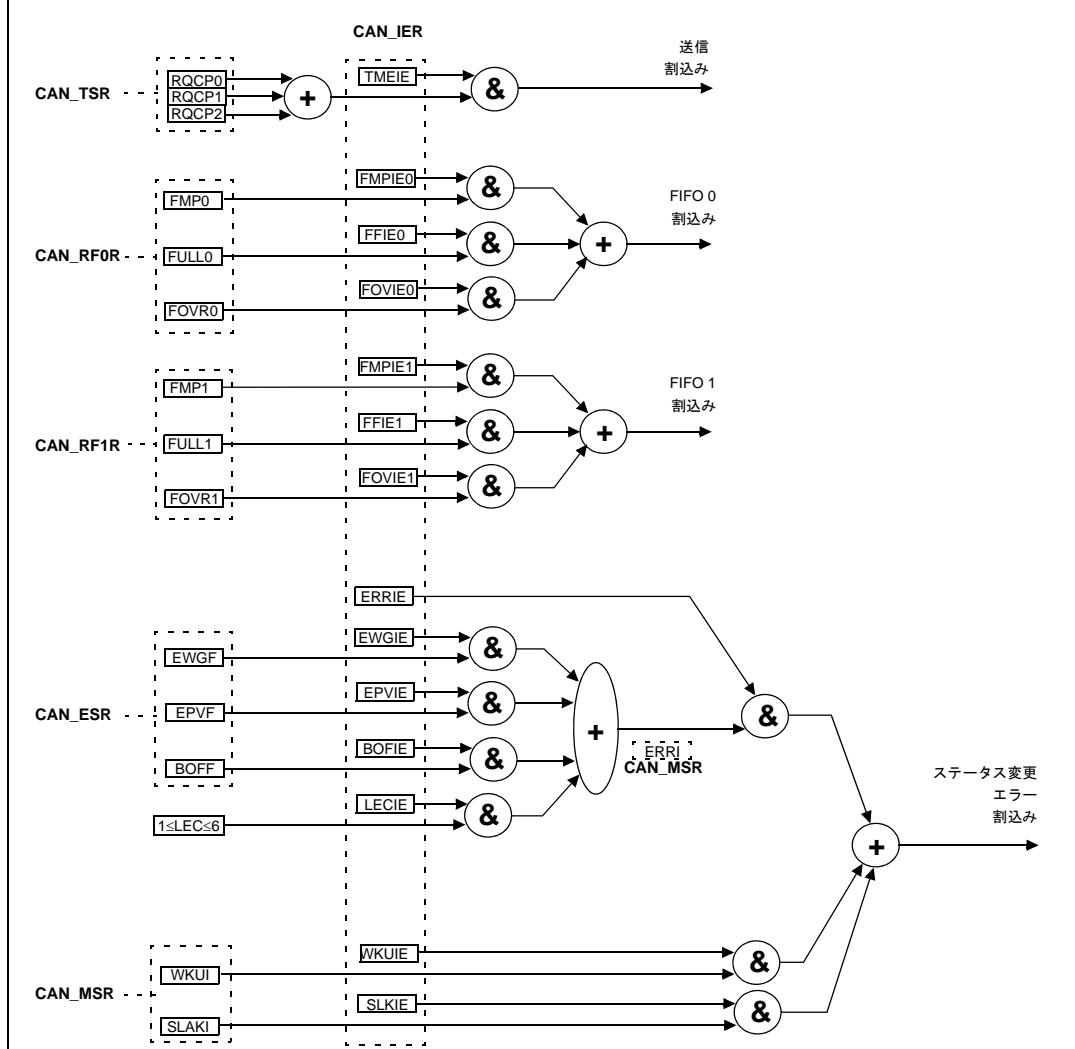


21.5 bxCAN の割込み

4 個の割込みベクタが bxCAN に配置されています。それぞれの割込み要因は CAN 割込みリクエスト許可レジスタ (CAN_IER) で独自に割込み許可 / 禁止の設定ができます。

参考資料

図 205. イベント・フラグと割込み生成



- **送信割込み**は次のイベントで発生します。
 - 送信メールボックス 0 が EMPTY になり、CAN_TSR レジスタの RQCP0 ビットがセットされたとき。
 - 送信メールボックス 1 が EMPTY になり、CAN_TSR レジスタの RQCP1 ビットがセットされたとき。
 - 送信メールボックス 2 が EMPTY になり、CAN_TSR レジスタの RQCP2 ビットがセットされたとき。
- **FIFO 0 割込み**は次のイベントで発生します。
 - 新たなメッセージの受信、CAN_RF0R レジスタの FMP0 が 0 でなくなったとき。
 - FIFO0 フル、CAN_RF0R レジスタの FULL0 ビットがセットされたとき。
 - FIFO0 オーバラン、CAN_RF0R レジスタの FOVR0 ビットがセットされたとき。
- **FIFO 1 割込み**は次のイベントで発生します。
 - 新たなメッセージの受信、CAN_RF1R レジスタの FMP1 が 0 でなくなったとき。
 - FIFO1 フル、CAN_RF1R レジスタの FULL1 ビットがセットされたとき。
 - FIFO1 オーバラン、CAN_RF1R レジスタの FOVR1 ビットがセットされたとき。

- エラー / 状態変化割込みは次のイベントで発生します。
 - エラー発生、エラーの詳細は CAN エラー・ステータス・レジスタ (CAN_ESR) を参照してください。
 - ウェークアップ、CAN Rx 信号で SOF が観察されたとき。
 - SLEEP モードへの遷移。

21.6 CAN レジスタ

21.6.1 レジスタ・アクセスに対する保護

レジスタへの誤ったアクセスでハードウェアが一時的に CAN ネットワーク全体の動作を妨げることがあります。このため、CAN_BTR レジスタは CAN ハードウェアが初期化モードのときにのみソフトウェアから変更ができます。

誤ったデータの送信が CAN ネットワークレベルの問題を引き起こすことはありませんが、アプリケーションには致命的な影響を与えることがあります。ソフトウェアによる送信メールボックスへの書き込みは、メールボックスが EMPTY 状態にあるときに限られます。詳細は [図 197: 送信メールボックスのステート](#) を参照してください。

フィルタの設定値の変更は、そのフィルタ・バンクが無効とされているか、もしくは FINIT ビットがセットされているときに限られます。さらに、CAN_FMxR、CAN_FSxR、CAN_RRAR レジスタにあるフィルタ構成（スケール、モード、FIFO 配置）の変更は CAN_FMR レジスタでのフィルタ初期化モードが設定されている (FINIT = 1) ときに限られます。

参考資料

21.6.2 CAN 制御 / ステータス・レジスタ

レジスタの詳細記述で使用されている略語については [セクション 1.1 \(32 ページ\)](#) を参照してください。

CAN マスタ制御レジスタ (CAN_MCR)

アドレスオフセット : 0x00
リセット値 : 0x0001 0002

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
RE SET	予約済み							TTCM	ABOM	AWUM	NART	RFLM	TXFP	SLEEP	INRQ

Bits 31:16 予約済み、ハードウェアで 0 に固定されています。

Bit 15 **RESET:** bxCAN ソフトウェアマスタリセット

0: 通常動作

1: bxCAN のマスタリセット動作を実行します。この際、リセット後はスリープモードに入ります。(CAN_MCR レジスタの FMP ビットはリセット時の値に戻されます。) このビットは自動的に 0 に戻ります。

Bits 14:8 予約済み、ハードウェアで 0 に固定されています。

Bit 7 **TTCM:** タイムトリガ通信モード

0: タイムトリガ通信モードに設定されていません。

1: タイムトリガ通信モードに設定されています。

注: タイムトリガ通信モードの詳細は [セクション 21.4.2: タイムトリガ通信モード](#) を参照してください。

Bit 6 **ABOM:** 自動バス・オフ管理

このビットは CAN ハードウェアが バス・オフ状態を離れる際の動作を指定します。

0: バス・オフ状態からの終了はソフトウェアの指示によります。11 個の連続したリセッショブ・ビットが 128 回検出されたとき、ソフトウェアは CAN_MCR レジスタの INRQ ビットを、まずセットし、次にクリアしなければなりません。

1: バス・オフ状態からの終了は、11 個の連続したリセッショブ・ビットが 128 回検出されたときにハードウェアによって自動的に行われます。

バス・オフ状態に関する詳細は [セクション 21.4.6: エラー管理](#) を参照してください。

Bit 5 **AWUM:** 自動ウェークアップ・モード

このビットは SLEEP モードにおいてメッセージを受取った時の CAN ハードウェアの動作を選択します。

0: ソフトウェアで CAN_MCR レジスタの SLEEP ビットをクリアしたときに SLEEP モードから終了します。

1: CAN メッセージを受け取ると自動的に SLEEP モードから終了します。

CAN_MCR レジスタの SLEEP ビットと CAN_MSR レジスタの SLAK ビットはハードウェアによってクリアされます。

Bit 4 **NART:** 自動再送信禁止

0: CAN ハードウェアは、CAN 規格に準じてメッセージが正常に送信されるまで、自動的にメッセージの再送信を行います。

1: 送信結果（正常、エラー、送信権喪失）にかかわらず、メッセージ送信は 1 回だけ行われます。

参考資料

Bit 3 RFLM: 受信 FIFO ロックモード

- 0: 受信 FIFO はオーバランでロックされません。受信 FIFO が一杯になると、次の着信メッセージは最後のメッセージに上書きされます。
- 1: 受信 FIFO はオーバランに対してロックされます。受信 FIFO が一杯になると、次の着信メッセージは破棄されます。

Bit 2 TXFP: 送信 FIFO 優先順位

- このビットは、複数のメールボックスが同時に PENDING 状態のときの送信順序を指定します。
- 0: メッセージの ID によって送信順序が定められます。
- 1: ソフトウェアにより送信リクエストが発せられた順に送信されます。

Bit 1 SLEEP: SLEEP モード指示

CAN ハードウェアが SLEEP モードに入ることを指示するため、このビットをソフトウェアでセットします。進行中の CAN 動作 (CAN フレームの送受信) が完了した時点で、ハードウェアは SLEEP モードに入れます。
ソフトウェアで SLEEP モードの終了を指示するには、このビットをクリアします。
また、AWUM ビットがセットされているときには、このビットは CAN Rx 信号上で SOF ビットが検出されたときにハードウェアによってクリアされます。
このビットはリセット直後はセットされており、CAN は SLEEP モードからスタートします。

Bit 0 INRQ: 初期化モード指示

ハードウェアが通常モードに入ることを指示するため、ソフトウェアでこのビットをクリアします。Rx 信号上で 11 個の連續したリセシシブ・ビットを検出したとき、ハードウェアはこれに同期し、送受信動作が可能な状態となります。ハードウェアはこれを CAN_MSR レジスタの INAK ビットをクリアすることで表示します。
ソフトウェアはこのビットをセットし、ハードウェアが初期化モードに入ることを指示することができます。ソフトウェアによって INRQ ビットがセットされると、CAN ハードウェアは進行中の CAN 動作 (CAN フレームの送受信) が完了するのを待って初期化モードに入れます。ハードウェアはこの時点での CAN_MSR レジスタの INAK ビットをセットします。

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

CAN マスタ・ステータス・レジスタ (CAN_MSR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0C02

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み	RX	SAMP	RXM	TXM	予約済み	SLAKI	WKUI	ERRI	SLAK	INAK	rc_w1	rc_w1	rc_w1	r	r
	r	r	r	r											

Bits 31:12 予約済み、ハードウェアで 0 に固定されています。

Bit 11 **RX: CAN Rx 信号**

CAN_RX ピンの実際の値を表示しています。

Bit 10 **SAMP: 最終サンプル値**

最後のサンプルポイントにおける RX の値（受信されたビットの値）を表示しています。

Bit 9 **RXM: 受信モード**

CAN ハードウェアがレシーバとして動作していることを示しています。

Bit 8 **TXM: 送信モード**

CAN ハードウェアがトランスマッタとして動作していることを示しています。

Bits 7:5 予約済み、ハードウェアで 0 に固定されています。

Bit 4 **SLAKI: SLEEP モード確認割込みリクエスト**

SLKIE=1 のとき、ハードウェアはこのビットをセットし bxCAN が SLEEP モードに入ったことを通知します。CAN_IER レジスタの SLKIE ビットがセットされていれば、このビットがセットされると 状態変化割込みが発生します。

注: SLKIE=0 の場合には、SLAKI ビットのポーリングは正しく機能しません。ポーリングには SLAK ビットを使用してください。

Bit 3 **WKUI: ウエークアップ割込みリクエスト**

CAN ハードウェアが SLEEP モードのとき、SOF ビットが検出されるとハードウェアはこのビットをセットして通知します。CAN_IER レジスタの WKUIE ビットがセットされていれば、このビットがセットされると 状態変化割込みが発生します。

このビットはソフトウェアでクリアします。

Bit 2 **ERRI: エラー割込みリクエスト**

エラーが検出によって CAN_ESR ビットがセットされ、CAN_IER レジスタの対応する割込みリクエストが許可されていれば、ハードウェアによってこのビットがセットされます。CAN_IER レジスタの ERRIE ビットがセットされていれば、このビットがセットされると 状態変化割込みが発生します。

このビットはソフトウェアでクリアします。



参考資料

Bit 1 SLAK: SLEEP モード・アクノレッジ

このビットは CAN ハードウェアが SLEEP モードにあることを表示するため、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からの SLEEP モード指示 (CAN_MCR レジスタの SLEEP ビットをセット) に対する肯定応答となります。

このビットは CAN ハードウェアが SLEEP モードを終了したときにハードウェアによってクリアされます。この動作は CAN バスに同期しており、CAN RX 信号上で 11 個の連続したリセッшиб・ビットが見出されるまで待たれます。

注: SLEEP モードを抜け出す動作は CAN_MCR レジスタの SLEEP ビットがクリアされることによって開始されます。SLEEP ビットのクリアに関しては、CAN_MCR レジスタの AWUM ビットの説明を参照してください。

Bit 0 IINAK: 初期化モード・アクノレッジ

このビットは CAN ハードウェアが初期化モードにあることを表示するため、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からの初期化モード指示 (CAN_MCR レジスタの INRQ ビットをセット) に対する肯定応答となります。

このビットは CAN ハードウェアが初期化モードを終了したときにハードウェアによってクリアされます。この動作は CAN バスに同期しており、CAN RX 信号上で 11 個の連続したリセッшиб・ビットが見出されるまで待たれます。

CAN 送信ステータス・レジスタ (CAN_TSR)

アドレスオフセット : 0x08

リセット値 : 0x1C00 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOW2	LOW1	LOW0	TME2	TME1	TME0	CODE[1:0]	ABRQ2	予約済み		TERR2	ALST2	TXOK2	RQCP2		
r	r	r	r	r	r	r	r	rs		rc_w1	rc_w1	rc_w1	rc_w1		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRQ1	予約済み		TERR1	ALST1	TXOK1	RQCP1	ABRQ0	予約済み		TERR0	ALST0	TXOK0	RQCP0		
rs		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rs			rc_w1	rc_w1	rc_w1	rc_w1		

Bit 31 LOW2: メールボックス 2 最低優先順位フラグ

複数のメールボックスが PENDING 状態で、メールボックス 2 が最低優先順位の時には、ハードウェアがこのビットをセットします。

Bit 30 LOW1: メールボックス 1 最低優先順位フラグ

複数のメールボックスが PENDING 状態で、メールボックス 1 が最低優先順位の時には、ハードウェアがこのビットをセットします。

Bit 29 LOW0: メールボックス 0 最低優先順位フラグ

複数のメールボックスが PENDING 状態で、メールボックス 0 が最低優先順位の時には、ハードウェアがこのビットをセットします。

注: LOW[2:0] ビットは PENDING 状態のメールボックスが 1 個の時には 0 となります。

Bit 28 TME2: 送信メールボックス 2 EMPTY

メールボックス 2 が EMPTY 状態ないとき、ハードウェアがこのビットをセットします。

Bit 27 TME1: 送信メールボックス 1 EMPTY

メールボックス 1 が EMPTY 状態ないとき、ハードウェアがこのビットをセットします。

Bit 26 TME0: 送信メールボックス 0 EMPTY

メールボックス 0 が EMPTY 状態ないとき、ハードウェアがこのビットをセットします。

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

Bits 25:24 **CODE[1:0]: メールボックス・コード**

少なくとも 1 個の送信メールボックスが EMPTY 状態のとき、次の EMPTY 状態の送信メールボックスの番号がコードの値となります。

すべての送信メールボックスが PENDING 状態のとき、最低優先順位の送信メールボックスの番号がコードの値となります。

Bit 23 **ABRQ2: メールボックス 2 送信中止リクエスト**

対応するメールボックスの送信リクエストを中止するには、ソフトウェアでこのビットをセットします。

メールボックスが EMPTY 状態になると、ハードウェアがこのビットをクリアします。

メールボックスが PENDING 状態にないときには、このビットをセットしても影響はありません。

Bits 22:20 予約済み、ハードウェアで 0 に固定されています。

Bit 19 **TERR2: メールボックス 2 送信エラー**

エラーにより送信 (TX) が失敗すると、このビットがセットされます。

Bit 18 **ALST2: メールボックス 2 送信権喪失**

送信権喪失により送信 (TX) が失敗すると、このビットがセットされます。

Bit 17 **TXOK2: メールボックス 2 送信 OK**

送信を試みた後、毎回、ハードウェアはこのビットをセットします。

0: 送信は失敗しました。

1: 送信は正常に終了しました。

メールボックス 2 の送信が正常に終了したときに、ハードウェアはこのビットをセットします。

[図 197](#) を参照してください。

Bit 16 **RQCP2: メールボックス 2 リクエスト完了**

最後のリクエスト（送信、もしくは中止）の実行が完了したときに、ハードウェアはこのビットをセットします。

ソフトウェアで 1 を書込んだとき、このビットはクリアされます。また、CAN_TMRD2R レジスタの TXRQ2 ビットがセットされ、送信がリクエストされたときにハードウェアがこのビットをクリアします。

このビットをクリアするとメールボックス 2 のすべてのステータス・ビット (TXOK2、ALST2 及び TERR2) がクリアされます。

Bit 15 **ABRQ1: メールボックス 1 送信中止リクエスト**

対応するメールボックスの送信リクエストを中止するには、ソフトウェアでこのビットをセットします。

メールボックスが EMPTY 状態になると、ハードウェアがこのビットをクリアします。

メールボックスが PENDING 状態にないときには、このビットをセットしても影響はありません。

Bits 14:12 予約済み、ハードウェアで 0 に固定されています。

Bit 11 **TERR1: メールボックス 1 送信エラー**

エラーにより送信 (TX) が失敗すると、このビットがセットされます。

Bit 10 **ALST1: メールボックス 1 送信権喪失**

送信権喪失により送信 (TX) が失敗すると、このビットがセットされます。

Bit 9 **TXOK1: メールボックス 1 送信完了**

送信を試みた後、毎回、ハードウェアはこのビットをセットします。

0: 送信は失敗しました。

1: 送信は正常に終了しました。

メールボックス 1 の送信が正常に終了したときに、ハードウェアはこのビットをセットします。

[図 197](#) を参照してください。



参考資料

Bit 8 RQCP1: メールボックス 1 リクエスト完了

最後のリクエスト（送信、もしくは中止）の実行が完了したときに、ハードウェアはこのビットをセットします。

ソフトウェアで 1 を書込んだとき、このビットはクリアされます。また、CAN_TMRD1R レジスタの TXRQ1 ビットがセットされ、送信がリクエストされたときにハードウェアがこのビットをクリアします。

このビットをクリアするとメールボックス 1 のすべてのステータス・ビット (TXOK1, ALST1 及び TERR1) がクリアされます。

Bit 7 ABRQ0: メールボックス 0 送信中止リクエスト

対応するメールボックスの送信リクエストを中止するには、ソフトウェアでこのビットをセットします。

メールボックスが EMPTY 状態になると、ハードウェアがこのビットをクリアします。

メールボックスが PENDING 状態にないときには、このビットをセットしても影響はありません。

Bits 6:4 予約済み、ハードウェアで 0 に固定されています。

Bit 3 TERR0: メールボックス 0 送信エラー

エラーにより送信 (TX) が失敗すると、このビットがセットされます。

Bit 2 ALST0: メールボックス 0 送信権喪失

送信権喪失により送信 (TX) が失敗すると、このビットがセットされます。

Bit 1 TXOK0: メールボックス 0 送信完了

送信を試みた後、毎回、ハードウェアはこのビットをセットします。

0: 送信は失敗しました。

1: 送信は正常に終了しました。

メールボックス 0 の送信が正常に終了したときに、ハードウェアはこのビットをセットします。

[図 197](#) を参照してください。

Bit 0 RQCP0: メールボックス 0 リクエスト完了

最後のリクエスト（送信、もしくは中止）の実行が完了したときに、ハードウェアはこのビットをセットします。

ソフトウェアで 1 を書込んだとき、このビットはクリアされます。また、CAN_TMRD0R レジスタの TXRQ0 ビットがセットされ、送信がリクエストされたときにハードウェアがこのビットをクリアします。

このビットをクリアするとメールボックス 0 のすべてのステータス・ビット (TXOK0, ALST0 及び TERR0) がクリアされます。

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

CAN 受信 FIFO 0 レジスタ (CAN_RF0R)

アドレスオフセット : 0x0C

リセット値 : 0x00

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								RFOM0	FOVR0	FULL0	予約済み	FMP0[1:0]			r	r

Bit 31:6 予約済み、ハードウェアで 0 に固定されています。

Bit 5 RFOM0: FIFO 0 出力メールボックスの解放

ソフトウェアでセットし、FIFO の出力メールボックスを解放を指示します。出力メールボックスは、FIFO 内の最後の 1 個のメッセージが PENDING 状態のときにのみ解放できます。FIFO が空のときにビットをセットしても影響はありません。複数のメッセージが FIFO 内で PENDING 状態の時には、次のメッセージにアクセスするにはソフトウェアで出力メールボックスを解放しなければなりません。

このビットは出力メールボックスが解放されるとハードウェアによってクリアされます。

Bit 4 FOVR0: FIFO 0 オーバラン

FIFO が一杯のときに新しいメッセージが受信されフィルタを通過すると、ハードウェアはこのビットをセットします。

このビットのクリアはソフトウェアで行います。

Bit 3 FULL0: FIFO 0 フル

FIFO に 3 個のメッセージが格納されると、ハードウェアはこのビットをセットします。

このビットのクリアはソフトウェアで行います。

Bit 2 予約済み、ハードウェアで 0 に固定されています。

Bits 1:0 FMP0[1:0]: FIFO 0 メッセージ PENDING

これらのビットは受信 FIFO で PENDING 状態になっているメッセージの数を表示しています。

FMP はハードウェアが FIFO に新しいメッセージを格納した際にカウントアップされ、ソフトウェアが RFOM0 ビットをセットすることで出力メールボックスを解放したときにカウントダウンされます。

参考資料

CAN 受信 FIFO 1 レジスタ (CAN_RF1R)

アドレスオフセット : 0x10

リセット値 : 0x00

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								RFOM1	FOVR1	FULL1	予約済み	FMP1[1:0]			r	r

Bits 31:6 予約済み、ハードウェアで 0 に固定されています。

Bit 5 RFOM1: FIFO 1 出力メールボックスの解放

ソフトウェアでセットし、FIFO の出力メールボックスを解放を指示します。出力メールボックスは、FIFO 内の最後の 1 個のメッセージが PENDING 状態のときにのみ解放できます。FIFO が空のときにビットをセットしても影響はありません。複数のメッセージが FIFO 内で PENDING 状態の時には、次のメッセージにアクセスするにはソフトウェアで出力メールボックスを解放しなければなりません。

このビットは出力メールボックスが解放されるとハードウェアによってクリアされます。

Bit 4 FOVR1: FIFO 1 オーバラン

FIFO が一杯のときに新しいメッセージが受信されフィルタを通過すると、ハードウェアはこのビットをセットします。

このビットのクリアはソフトウェアで行います。

Bit 3 FULL1: FIFO 1 フル

FIFO に 3 個のメッセージが格納されると、ハードウェアはこのビットをセットします。

このビットのクリアはソフトウェアで行います。

Bit 2 予約済み、ハードウェアで 0 に固定されています。

Bits 1:0 FMP1[1:0]: FIFO 1 メッセージ PENDING

これらのビットは受信 FIFO で PENDING 状態になっているメッセージの数を表示しています。FMP はハードウェアが FIFO に新しいメッセージを格納した際にカウントアップされ、ソフトウェアが RFOM1 ビットをセットすることで出力メールボックスを解放したときにカウントダウンされます。

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

CAN 割込みリクエスト許可レジスタ (CAN_IER)

アドレスオフセット : 0x14

リセット値 : 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み														SLKIE	WKUIE
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRIE	予約済み		LEC IE	BOF IE	EPV IE	EWG IE	予約済み	FOV IE1	FF IE1	FMP IE1	FOV IEO	FF IEO	FMP IEO	TME IE	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:18 予約済み、ハードウェアで 0 に固定されています。

Bit 17 **SLKIE**: Sleep 割込みリクエスト許可

0: SLAKI ビットがセットされたとき、割込みリクエストは発生しません。

1: SLAKI ビットがセットされたとき、割込みリクエストが発生します。

Bit 16 **WKUIE**: ウェークアップ割込みリクエスト許可

0: WKUI ビットがセットされたとき、割込みリクエストは発生しません。

1: WKUI ビットがセットされたとき、割込みリクエストが発生します。

Bit 15 **ERRIE**: エラー割込みリクエスト許可

0: CAN_ESR にエラー条件がセットされているとき、割込みリクエストは発生しません。

1: CAN_ESR にエラー条件がセットされているとき、割込みリクエストが発生します。

Bits 14:12 予約済み、ハードウェアで 0 に固定されています。

Bit 11 **LECIE**: 最終エラーコード割込みリクエスト許可

0: エラーが検出され、ハードウェアが LEC[2:0] にエラーコードがセットされたとき、ERRIE ビットはセットされません。

1: エラーが検出され、ハードウェアが LEC[2:0] にエラーコードがセットされたとき、ERRIE ビットがセットされます。

Bit 10 **BOFIE**: バス・オフ割込みリクエスト許可

0: BOFF がセットされたとき、ERRIE ビットはセットされません。

1: BOFF がセットされたとき、ERRIE ビットがセットされます。

Bit 9 **EPVIE**: エラーパッシブ割込みリクエスト許可

0: EPVF がセットされたとき、ERRIE ビットはセットされません。

1: EPVF がセットされたとき、ERRIE ビットがセットされます。

Bit 8 **EWGIE**: エラーオーナリング割込みリクエスト許可

0: EWGF がセットされたとき、ERRIE ビットはセットされません。

1: EWGF がセットされたとき、ERRIE ビットがセットされます。

Bit 7 予約済み、ハードウェアで 0 に固定されています。

Bit 6 **FOVIE1**: FIFO 1 オーバラン割込みリクエスト許可

0: FOVR ビットがセットされたとき、割込みリクエストは発生しません。

1: FOVR ビットがセットされたとき、割込みリクエストが発生します。

Bit 5 **FFIE1**: FIFO 1 フル割込みリクエスト許可

0: FULL ビットがセットされたとき、割込みリクエストは発生しません。

1: FULL ビットがセットされたとき、割込みリクエストが発生します。



参考資料

Bit 4 **FMP1E1**: FIFO 1 メッセージ PENDING 割込みリクエスト許可

- 0: FMP[1:0] が 0 でないとき、割込みリクエストは発生しません。
- 1: FMP[1:0] が 0 でないとき、割込みリクエストが発生します。

Bit 3 **FOVIE0**: FIFO 0 オーバラン割込みリクエスト許可

- 0: FOVR ビットがセットされたとき、割込みリクエストは発生しません。
- 1: FOVR ビットがセットされたとき、割込みリクエストが発生します。

Bit 2 **FFIE0**: FIFO 0 フル割込みリクエスト許可

- 0: FULL ビットがセットされたとき、割込みリクエストは発生しません。
- 1: FULL ビットがセットされたとき、割込みリクエストが発生します。

Bit 1 **FMP1E0**: FIFO 0 メッセージ PENDING 割込みリクエスト許可

- 0: FMP[1:0] が 0 でないとき、割込みリクエストは発生しません。
- 1: FMP[1:0] が 0 でないとき、割込みリクエストが発生します。

Bit 0 **TMEIE**: 送信メールボックス EMPTY 割込みリクエスト許可

- 0: RQCPx ビットがセットされているとき、割込みリクエストは発生しません。
- 1: RQCPx ビットがセットされているとき、割込みリクエストが発生します。

注 : [セクション 21.5: bxCAN の割込み](#)を参照してください。

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

CAN エラー・ステータス・レジスタ (CAN_ESR)

アドレスオフセット : 0x18

リセット値 : 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REC[7:0]								TEC[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								LEC[2:0]			予約済み	BOFF	EPVF	EWGF	
								rw	rw	rw		r	r	r	

Bits 31:24 REC[7:0]: 受信エラーカウンタ

CAN プロトコルの「フォルトコンファインメント」の実装です。受信の際にエラーが検出されると、CAN 規格の規定に準じて、1 もしくは 8 がカウントされます。受信が正常に行われるたびに、このカウンタは 1 カウントダウンされるか、(カウンタの値が 128 を超えているときには) 120 にリセットされます。カウンタの値が 127 を超えると、CAN コントローラはエラーパッシブの状態に入ります。

Bits 23:16 TEC[7:0]: 9bit 送信エラーカウンタの下位バイト

CAN プロトコルの「フォルトコンファインメント」の実装です。

Bits 15:7 予約済み、ハードウェアで 0 に固定されています。

Bits 6:4 LEC[2:0]: 最終エラーコード

これらのビットには、CAN バス上で検出された最後のエラーの状況を表示するコードを、ハードウェアで格納します。メッセージが正常に転送（送受信とも）された時には、これらのビットは 0 にクリアされます。

コード値 7 は未使用ですが、ハードウェアで更新状態をチェックするために書き込まれことがあります。

000: エラーなし

001: スタッフ・エラー

010: フーム・エラー

011: ACK エラー

100: リセッショナル・ビット・エラー

101: ドミナント・ビット・エラー

110: CRC エラー

111: ソフトウェアでセットされます。

Bit 3 予約済み、ハードウェアで 0 に固定されています。

Bit 2 BOFF: バス・オフ・フラグ

バス・オフの状態になると、ハードウェアはこのビットをセットします。バス・オフ状態は TEC オーバーフロー時に（カウント値が 255 を超えたとき）発生します。詳細は [セクション 21.4.6 \(536 ページ\)](#) を参照してください。

Bit 1 EPVF: エラー・パッシブ・フラグ

エラーパッシブ状態（受信エラーカウンタもしくは送信エラーカウンタ > 127）に達したとき、ハードウェアがこのビットをセットします。

Bit 0 EWGF: エラー・ウォーニング・フラグ

エラーワーニング状態（受信エラーカウンタもしくは送信エラーカウンタ ³ 96）に達したとき、ハードウェアがこのビットをセットします。



参考資料

CAN ビット・タイミング・レジスタ (CAN_BTR)

アドレスオフセット : 0x1C

リセット値 : 0x0123 0000

注意 : このレジスタにソフトウェアからアクセスできるのは CAN ハードウェアが初期化モードの時に限られます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
SILM	LBKM	予約済み			SJW[1:0]	予約済み	TS2[2:0]			TS1[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			BRP[9:0]													

Bit 31 **SILM:** サイレント・モード (デバッグ用)

0: 通常動作

1: サイレント・モード

Bit 30 **LBKM:** ループバック・モード (デバッグ用)

0: ループバック・モードは設定されません。

1: ループバック・モードは有効です。

Bits 29:26 予約済み、ハードウェアで 0 に固定されています。

Bits 25:24 **SJW[1:0]:** 再同期ジャンプ幅

これらのビットでは、CAN ハードウェアが再同期の際にビット幅の伸縮を行うことができる幅を時間単位 (quanta) の数で指定します。

$$t_{RJW} = t_{CAN} \times (SJW[1:0] + 1)$$

Bit 23 予約済み、ハードウェアで 0 に固定されています。

Bits 22:20 **TS2[2:0]:** タイムセグメント 2

これらのビットでは、タイムセグメント 2 の長さを時間単位 (quanta) の数で指定します。

$$t_{BS2} = t_{CAN} \times (TS2[2:0] + 1)$$

Bits 19:16 **TS1[3:0]:** タイムセグメント 1

これらのビットでは、タイムセグメント 1 の長さを時間単位 (quanta) の数で指定します。

$$t_{BS1} = t_{CAN} \times (TS1[3:0] + 1)$$

ビット・タイミングの詳細に関しては [セクション 21.4.7: ビット・タイミング \(537 ページ\)](#) を参照してください。

Bits 15:10 予約済み、ハードウェアで 0 に固定されています。

Bits 9:0 **BRP[9:0]:** ポーレートプリスケーラ

このフィールドは時間単位 (time quanta) の長さを指定します。

$$t_q = (BRP[9:0] + 1) \times t_{PCLK}$$

21.6.3 メールボックス・レジスタ

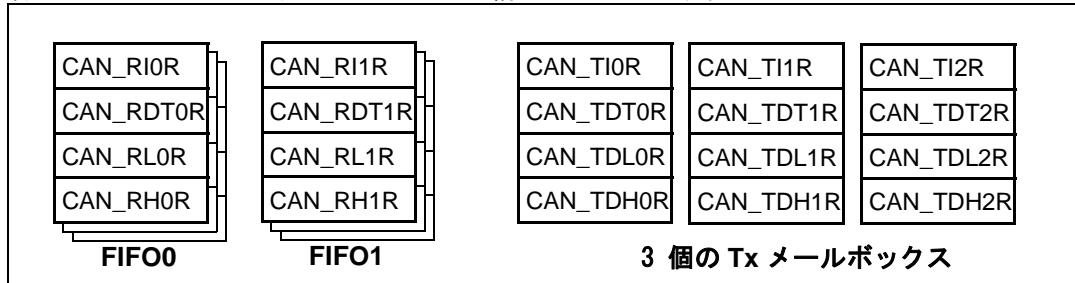
このセクションでは受信及び送信メールボックスのレジスタに関して記してあります。レジスタマッピングについては[セクション 21.4.5: メッセージ・メモリ \(メールボックス\) \(535 ページ\)](#) を参照してください。

メールボックスは送信と受信で類似した構造ですが、次の点が異なっています。

- CAN_RDTxR レジスタの FMI フィールド
- 受信メールボックスは常に書き込み禁止
- 送信メールボックスは、CAN_TSR レジスタの TME ビットの状態に応じて、EMPTY 状態のときのみ書き込み可能

送信メールボックスが 3 個、受信メールボックスが 2 個あります。受信メールボックスは、それぞれが 3 レベルの深さの FIFO となっており、FIFO 内の最も古いメッセージにアクセスします。

各メールボックスは 4 個のレジスタから構成されています。



CAN 送信メールボックス ID レジスタ (CAN_TIxR) (x=0..2)

アドレスオフセット : 180h, 190h, 1A0h
リセット値 : 不定 (ただし、ビット 0 TXRQ = 0)

- 注意 :
- 1 送信レジスタはメールボックスが送信待ち (TME_x クリア) の時には書き込みができません。
 - 2 このレジスタには送信リクエスト制御ビット (ビット 0) が組込まれています。
- リセット値 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
STID[10:0]/EXID[28:18]														EXID[17:13]		
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	
EXID[12:0]														IDE	RTR	TXRQ
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 31:21 STID[10:0]/EXID[28:18]: 標準あるいは拡張 ID

IDE ビットの値に応じて、標準 ID あるいは拡張 ID の上位ビット

Bit 20:3 EXID[17:0]: 拡張 ID

拡張 ID の下位ビット

参考資料

Bit 2 IDE: 拡張 ID 有効

このビットはメールボックス・メッセージの ID の種類を指定します。

0: 標準 ID

1: 拡張 ID

Bit 1 RTR: リモート送信リクエスト

0: データフレーム

1: リモートフレーム

Bit 0 TXRQ: 送信メールボックス送信リクエスト

対応するメールボックスの送信をリクエストするには、ソフトウェアでこのビットをセットします。

メールボックスが EMPTY 状態になると、ハードウェアがこのビットをクリアします。

CAN 送信メールボックス・データ長・タイムスタンプ・レジスタ (CAN_TDTxR) (x=0..2)

メールボックスが EMPTY 状態でないときには、このレジスタの全ビットは書き込み禁止状態です。

アドレスオフセット : 0x184、0x194、0x1A4

リセット値 : 0xXX X は不定

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIME[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
予約済み				TGT	予約済み				DLC[3:0]				rw	rw	rw
rw															

Bits 31:16 TIME[15:0]: メッセージ・タイムスタンプ

SOF 送信時に得られた 16bit タイマの値を格納します。

Bits 15:9 予約済み

Bit 8 TGT: 送信グローバルタイム

このビットはハードウェアが、CAN_MCR レジスタの TTCM ビットがセットされているタイムトリガ通信モードのときのみ有効です。

0: タイムスタンプ TIME[15:0] は送信されません。

1: タイムスタンプ TIME[15:0] の値は、8 バイトメッセージの最終 2 データバイトとして送信されます。このとき TIME[7:0] はデータバイト 6 として、TIME[15:8] はデータバイト 7 として、CAN_TDTHxR[31:16] レジスタ (DATA6[7:0]、DATA7[7:0]) の内容の代わりに送信されます。これら 2 バイトが CAN バス上に送出されるには、DLC フィールドが 8 に設定されていなければなりません。

Bits 7:4 予約済み

Bits 3:0 DLC[3:0]: データ長コード

これらのビットは、データフレーム内のデータバイト数、もしくはリモートフレームリクエストのデータバイト数を定めます。

メッセージは DLC フィールドの値に応じて、0 から 8 個のデータバイトを含むことができます。



参考資料

CAN 送信メールボックス・ データ・レジスタ (下位) (CAN_TDLxR) (x=0..2)

メールボックスが EMPTY 状態でないときには、このレジスタの全ビットは書き込み禁止状態です。

アドレスオフセット : 0x188、0x198、0x1A8
リセット値 : 0xXX X は不定

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA3[7:0]								DATA2[7:0]							
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
DATA1[7:0]								DATA0[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:24 DATA3[7:0]: データバイト 3

メッセージのデータバイト 3 の値

Bits 23:16 DATA2[7:0]: データバイト 2

メッセージのデータバイト 2 の値

Bits 15:8 DATA1[7:0]: データバイト 1

メッセージのデータバイト 1 の値

Bits 7:0 DATA0[7:0]: データバイト 0

メッセージのデータバイト 0 の値

メッセージは 0 ないし 8 個のデータバイトを含むことができます。データバイトはバイト 0 から送出されます。

参考資料

CAN 送信メールボックス・データ・レジスタ (上位) (CAN_TDHzR) (x=0..2)

メールボックスが EMPTY 状態でないときには、このレジスタの全ビットは書き込み禁止状態です。

アドレスオフセット : 0x18C、0x19C、0x1AC
リセット値 : 0xXX X は不定

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA7[7:0]								DATA6[7:0]							
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
DATA5[7:0]								DATA4[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:24 DATA7[7:0]: データバイト 7

メッセージのデータバイト 7 の値

注: メッセージの TGT と TTCM が有効であれば、DATA7 及び DATA6 の代わりにタイムスタンプ値 TIME が送出されます。

Bits 23:16 DATA6[7:0]: データバイト 6

メッセージのデータバイト 6 の値

Bits 15:8 DATA5[7:0]: データバイト 5

メッセージのデータバイト 5 の値

Bits 7:0 DATA4[7:0]: データバイト 4

メッセージのデータバイト 4 の値

参考資料

CAN 受信 FIFO メールボックス ID レジスタ (CAN_RXR) (x=0..1)

アドレスオフセット : 0x1B0、0x1C0

リセット値 : 0xXX Xは不定

注意 : 受信レジスタはすべて書き込みできません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STID[10:0]/EXID[28:18]												EXID[17:13]			
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
EXID[12:0]												IDE	RTR	予約済み	
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:21 STID[10:0]/EXID[28:18]: 標準あるいは拡張 ID

IDE ビットの値に応じて、標準 ID あるいは拡張 ID の上位ビット

Bits 20:3 EXID[17:0]: 拡張 ID

拡張 ID の下位ビット

Bit 2 IDE: 拡張 ID 有効

このビットはメールボックス・メッセージの ID の種類を指定します。

0: 標準 ID

1: 拡張 ID

Bit 1 RTR: リモート送信リクエスト

0: データフレーム

1: リモートフレーム

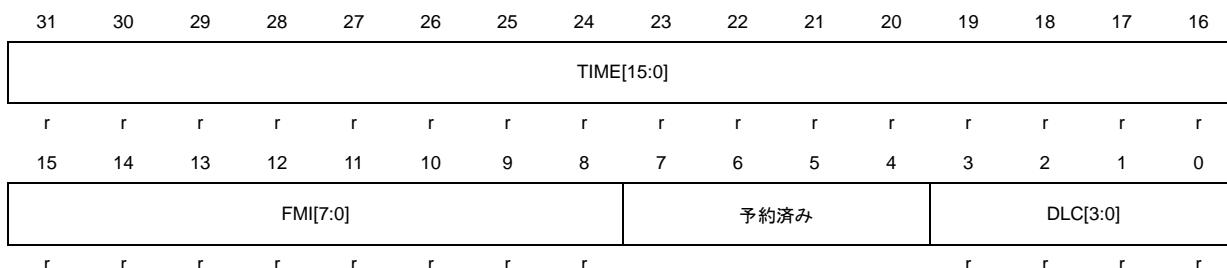
Bit 0 予約済み

参考資料

CAN 受信 FIFO メールボックス・データ長・タイムスタンプ・レジスタ (CAN_RDTxR) ($x=0..1$)

アドレスオフセット : 0x1B4、0x1C4
リセット値 : 0xXX X は不定

注意：受信レジスタはすべて書き込みできません。



Bits 31:16 TIME[15:0]: メッセージ・タイムスタンプ

SOF 検出時に得られた 16 ビットタイマの値を格納しています。

Bits 15:8 FMI[7:0]: 該当フィルタ索引

これらのビットは、メールボックスに格納されているメッセージに合致するフィルタの番号が格納されます。ID フィルタリングの詳細に関しては [セクション 21.4.4: ID によるフィルタリング \(530 ページ\)](#) の該当フィルタ索引の項を参照してください。

Bits 7:4 予約済み、ハードウェアで 0 に固定されています。

Bits 3:0 DLC[3:0]: データ長コード

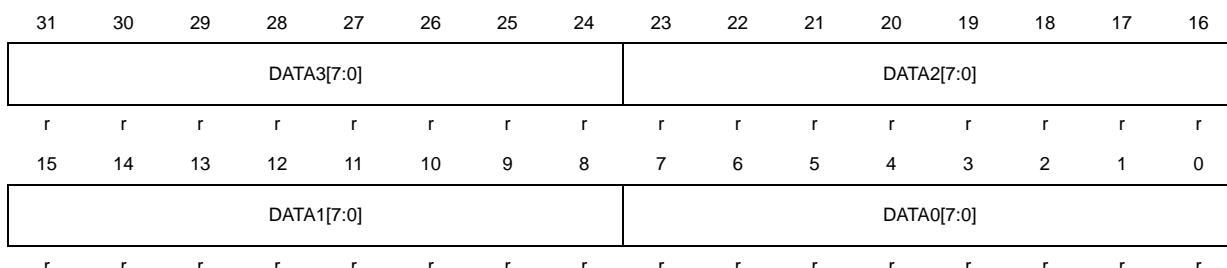
これらのビットにはデータフレームが保持しているデータバイト数 (0 - 8) が格納されます。リモートフレームリクエストの時には、これらのビットは 0 となります。

CAN 受信 FIFO メールボックス・データ・レジスタ (下位) (CAN_RDLxR) ($x=0..1$)

メールボックスが空の状態でない場合、このレジスタの全てのビットは書き込みできません。

アドレスオフセット : 0x1B8、0x1C8
リセット値 : 0xXX X は不定

注意：受信レジスタはすべて書き込みできません。



Bits 31:24 DATA3[7:0]: データバイト 3

メッセージのデータバイト 3 の値

参考資料

Bits 23:16 DATA2[7:0]: データバイト 2

メッセージのデータバイト 2 の値

Bits 15:8 DATA1[7:0]: データバイト 1

メッセージのデータバイト 1 の値

Bits 7:0 DATA0[7:0]: データバイト 0

メッセージのデータバイト 0 の値

メッセージは 0 ないし 8 個のデータバイトを含むことができます。データバイトはバイト 0 から始まります。

CAN 受信 FIFO メールボックス・データ・レジスタ (上位) (CAN_RDHxR) (x=0..1)

アドレスオフセット : 0x1BC、0x1CC

リセット値 : 0xXX X は不定

注意 : 受信レジスタはすべて書き込みできません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA7[7:0]								DATA6[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA5[7:0]								DATA4[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:24 DATA7[7:0]: データバイト 7

メッセージのデータバイト 3 の値

Bits 23:16 DATA6[7:0]: データバイト 6

メッセージのデータバイト 2 の値

Bits 15:8 DATA5[7:0]: データバイト 5

メッセージのデータバイト 1 の値

Bits 7:0 DATA4[7:0]: データバイト 4

メッセージのデータバイト 0 の値

参考資料

21.6.4 CAN フィルタ・レジスタ

CAN フィルタ・マスター・レジスタ (CAN_FMR)

アドレスオフセット : 0x200
リセット値 : 0x2A1C 0E01

注意 : このレジスタの全ビットはソフトウェアでセット／クリアができます。

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															

rw

Bits 31:1 予約済み、ハードウェアでリセット値に固定されています。

Bit 0 **FINIT**: フィルタ初期化モード
フィルタ・バンクの初期化モード
0: フィルタの機能を有効とします。
1: フィルタの初期化モード。

CAN フィルタ・モード・レジスタ (CAN_FM0R)

アドレスオフセット : 0x204
リセット値 : 0x00

注意 : このレジスタへの書き込みはフィルタ初期化モード (CAN_FMR レジスタの FINIT = 1) のときのみ可能です。

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
Reserved	FBM13	FBM12	FBM11	FBM10	FBM9	FBM8	FBM7	FBM6	FBM5	FBM4	FBM3	FBM2	FBM1	FBM0	

rw rw

注意 : 図 199: フィルタ・バンクのスケール設定 - レジスタ構成 (532 ページ) を参照してください。

Bits 31:14 予約済み、ハードウェアで 0 に固定されています。

Bits 13:0 **FBMx**: フィルタモード
フィルタ × レジスタのモード
0: フィルタ・バンク × の 2 個の 32 ビットレジスタは ID マスクモードです。
1: フィルタ・バンク × の 2 個の 32 ビットレジスタは ID リストモードです。

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

CAN フィルタ・スケール・レジスタ (CAN_FSOR)

アドレスオフセット : 0x20C

リセット値 : 0x00

注意 : このレジスタへの書き込みはフィルタ初期化モード (CAN_FMR レジスタの FINIT = 1) のときのみ可能です。

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	FSC13	FSC12	FSC11	FSC10	FSC9	FSC8	FSC7	FSC6	FSC5	FSC4	FSC3	FSC2	FSC1	FSC0	

注意 : [図 199: フィルタ・バンクのスケール設定 - レジスタ構成 \(532 ページ\)](#) を参照してください。

Bits 31:14 予約済み、ハードウェアで 0 に固定されています。

Bits 13:0 **FSCx**: フィルタスケール構成

これらのビットはフィルタ 13-0 のスケール構成を指定します。

0: デュアル 16 ビットスケール構成

1: シングル 32 ビットスケール構成

CAN フィルタ FIFO 配置レジスタ (CAN_FFA0R)

アドレスオフセット : 0x214

リセット値 : 0x00

注意 : このレジスタへの書き込みはフィルタ初期化モード (CAN_FMR レジスタの FINIT = 1) のときのみ可能です。

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	FFA13	FFA12	FFA11	FFA10	FFA9	FFA8	FFA7	FFA6	FFA5	FFA4	FFA3	FFA2	FFA1	FFA0	

Bits 31:14 予約済み、ハードウェアで 0 に固定されています。

Bits 13:0 **FFAx**: フィルタ x FIFO 配置

このフィルタを通ったメッセージが格納される FIFO を指定します。

0: フィルタは FIFO 0 に配置されます。

1: フィルタは FIFO 1 に配置されます。



参考資料

CAN フィルタ・有効化レジスタ (CAN_FA0R)

アドレスオフセット : 0x21C
リセット値 : 0x00

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	FACT13	FACT12	FACT11	FACT10	FACT9	FACT8	FACT7	FACT6	FACT5	FACT4	FACT3	FACT2	FACT1	FACT0	

Bits 31:14 予約済み、ハードウェアで 0 に固定されています。

Bits 13:0 FACT_x フィルタアクティブ

ソフトウェアでこのビットをセットすると、フィルタ x がアクティブとなります。フィルタ x レジスタ (CAN_FxR[0:7]) を変更するには、FACT_x ビットがクリアされているか、CAN_FMR の FINIT ビットがセットされていなければなりません。

0: フィルタ x はアクティブではありません。

1: フィルタ x はアクティブです。

CAN フィルタ・バンク x レジスタ (CAN_FxR[1:0]) ($x=0..13$)

アドレスオフセット : 0x240..0x2AC

リセット値 : 0xXX X は不定

注意: 14 個のフィルタ・バンク ($x = 0..13$) が存在します。それぞれのフィルタ・バンクは、2 個の 32 ビットレジスタ CAN_FxR[1:0] からなっています。

このレジスタは、CAN_FAxR レジスタの FACT_x ビットがクリアされているか、もしくは CAN_FMR レジスタの FINIT ビットがセットされているときのみ、変更ができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FB31	FB30	FB29	FB28	FB27	FB26	FB25	FB24	FB23	FB22	FB21	FB20	FB19	FB18	FB17	FB16
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FB15	FB14	FB13	FB12	FB11	FB10	FB9	FB8	FB7	FB6	FB5	FB4	FB3	FB2	FB1	FB0
rw															

参考資料

全てのコンフィグレーションにおいて：

Bits 31:0 **FB[31:0]**: フィルタ・ビット

ID

レジスタの個々のビットは、希望する ID のバス上での信号レベルを指定します。

0: ドミナント・ビット

1: リセッショブ・ビット

マスク

レジスタの個々のビットは、対応する ID レジスタのビットと受信 ID のビットが一致しなければいけないか、一致の必要がないかを指定します。

0: 無視、このビットは比較の対象とはなりません。

1: 受信 ID のビットはフィルタの対応する ID ビットと同じ値でなければなりません。

注意：

フィルタのスケールとモードの設定に応じて、個々のレジスタの機能が異なることがあります。フィルタマッピング機能とマスク・レジスタの関連に関しては、[セクション 21.4.4: ID によるフィルタリング \(530 ページ\)](#) を参照してください。

マスクモードにおけるマスク / ID レジスタは ID リストモードの場合と同じビットマッピングです。

フィルタ・バンクのレジスタマッピング / アドレスに関しては、[表 144 \(564 ページ\)](#) を参照してください。

參考資料

21.6.5 bxCAN レジスタ・マップ

レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。

表 144. bxCAN - レジスタ・マップとリセット時の値

参考資料

RM0008

CAN (Controller Area Network) (bxCAN)

表 144. bxCAN - レジスタ・マップとリセット時の値 (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
0x194	CAN_TDT1R																																
	リセット値	x x																															
0x198	CAN_TDL1R																																
	リセット値	x x																															
0x19C	CAN_TD1R																																
	リセット値	x x																															
0x1A0	CAN_TI2R																																
	リセット値	x x																															
0x1A4	CAN_TDT2R																																
	リセット値	x x																															
0x1A8	CAN_TDL2R																																
	リセット値	x x																															
0x1AC	CAN_TD2R																																
	リセット値	x x																															
0x1B0	CAN_RI0R																																
	リセット値	x x																															
0x1B4	CAN_RDT0R																																
	リセット値	x x																															
0x1B8	CAN_RDL0R																																
	リセット値	x x																															
0x1BC	CAN_RDH0R																																
	リセット値	x x																															
0x1C0	CAN_RI1R																																
	Reset value	x x																															
0x1C4	CAN_RDT1R																																
	リセット値	x x																															
0x1C8	CAN_RDL1R																																
	リセット値	x x																															
0x1CC	CAN_RDH1R																																
	リセット値	x x																															
0x1D0-0x1FF																																	

参考資料

表 144. bxCAN - レジスタ・マップとリセット時の値 (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
0x200	CAN_FMR リセット値																															FINT 1	
0x204	CAN_FM1R リセット値																														FBM[13:0]		
0x208																															予約済み		
0x20C	CAN_FS1R リセット値																														FSC[13:0]		
0x210																															予約済み		
0x214	CAN_FFA1R リセット値																														FFA[13:0]		
0x218																															予約済み		
0x21C	CAN_FA1R リセット値																														FACT[13:0]		
0x220																															予約済み		
0x224- 0x23F																															予約済み		
0x240	CAN_F0R1 リセット値	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	FB[31:0]			
0x244	CAN_F0R2 リセット値	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	FB[31:0]			
0x248	CAN_F1R1 リセット値	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	FB[31:0]			
0x24C	CAN_F1R2 リセット値	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	FB[31:0]			
.		
0x2A8	CAN_F13R1 リセット値	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	FB[31:0]			
0x2AC	CAN_F13R2 リセット値	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	FB[31:0]			

22 SPI (Serial peripheral interface)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

22.1 SPI の概要

大容量デバイスの SPI インタフェースは、SPI プロトコルあるいは I²S オーディオ・プロトコルの何れかを得る事ができる柔軟性を提供します。デフォルトでは SPI 機能が選択されます。ソフトウェアにより SPI から I²S へインタフェースを切り替えることもできます。中容量デバイスでは I²S プロトコルは利用できません。

SPI (Serial peripheral interface) では、外部デバイスとの間で、半二重 / 全二重、同期式の通信を行うことができます。インタフェース回路は、外部のスレーブデバイスに対して通信クロック (SCK) を供給するマスタとなるよう設定することができます。また、複数のマスターが存在する方式にも対応できます。

インタフェース回路は、両方向のデータ転送を行うことができる 2 本の信号線上の單一方向の同期転送や、CRC チェックを採用したより信頼性の高い方式など、種々の目的に利用することができます。

I²S は同期、3 ピンのプロトコルによるシリアル通信インターフェースです。それは、I²S フィリップス標準、MSB と LSB-justified 標準及び PCM 標準の 4 つの標準に対応することができます。I²S は、半二重通信のスレーブまたはマスター・モードになることができます。I²S が通信マスターとして構成されていると、マスタークロックはインターフェースにより外部スレーブ部品に提供されることがあります。

要注意： **幾つかの SPI3/I2S3 ピンは JTAG ピン (JTDI による SPI3_NSS/I2S3_WS と JTDO により SPI3_SCK/I2S3_CK) と共有されるので、I/O コントローラにより制御されず JTAG 用に予約されます。(各リセットの後) SPI3/I2S3 ピンの設定前のこの目的により、ユーザは JTAG をディセーブルし SWD インタフェース (アプリケーションでデバッグする際) を使うか、JTAG/SWD の両方のインターフェース (単体アプリケーション用) をディセーブルしなければなりません。JTAG/SWD インタフェースピンの設定に関する詳細は [セクション 7.3.4: JTAG/SWD オルタネート機能のピン割当て](#)を参照してください。**

22.2 SPI と I²S の主な機能

22.2.1 SPI 機能

- 3 本の信号線上で、全二重同期転送が可能。
- データ線の両方向機能の有無にかわらず、2 本の信号線で単向の同期転送が可能。
- 8 または 16 bit 転送フレームの選択。
- マスタ及びスレーブ動作。
- 複数マスタ・モードに対応。
- マスタ・モードにおけるボーレートプリスケーラ周波数（最大 $f_{PCLK}/2$ ）
- スレーブ・モード周波数（最大 $f_{PCLK}/2$ ）
- マスタ、スレーブ・モードともに、より早い通信が可能：SPI 速度は最大で 18 MHz
- マスタ、スレーブ・モードともに、ハードウェア及びソフトウェアによる NSS 管理が可能：マスタ・モードとスレーブ・モードを動作時に切り替え可能。
- クロック極性と位相をプログラムで設定可能。
- MSB から際に送出するか、LSB から送出するかの設定が、プログラムから可能。
- 送信・受信で独立したフラグを持ち、割込みリクエストが可能。
- SPI バスビジーステータスを示すフラグ。
- 信頼性の高い通信のための、ハードウェアによる CRC 機能：
 - 送信時には、CRC の値は最小バイトとして送信されます。
 - 全二重モードでは、最終バイトをチェックコードとして CRC エラーチェックを自動的に実行。
- マスタ・モードのエラー、オーバラン、CRC エラーが個別のフラグを持ち、割込みのリクエストが可能。
- 送受信ともに DMA 機能を持つ 1 バイトのバッファを持ち、個別にリクエストを発生。

22.2.2 I²S 機能

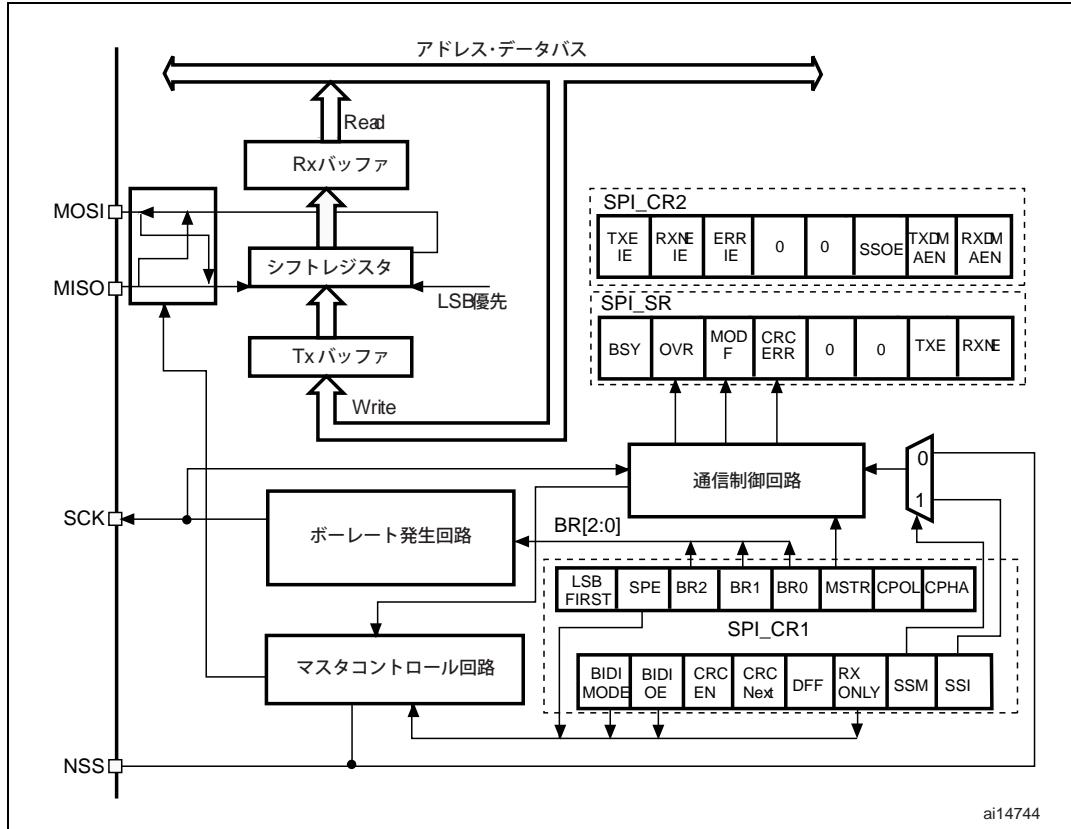
- 一方向通信（送信機か受信機のみ）
- マスタまたはスレーブ処理
- 正確なオーディオサンプル周波数（8KHz から 48 kHz）のための 8bit プログラマブルリニアプリスケーラ
- データ・フォーマットは 16bit、24bit または 32bit
- パケットフレームはオーディオチャネルにより 16bit（16bit データ・フレーム）または 32bit（16bit、24bit、32bit データ・フレーム）固定
- プログラマブルクロック位相（安定状態）
- スレーブ送信モードのアンダーランフラグ及び受信モードのオーバーランフラグ（マスタとスレーブ）
- 送信と受信用 16bit レジスタ及び両チャネルサイド用の 1 つのデータ・レジスタ
- サポートされる I²S プロトコル：
 - I²S フィリップス標準
 - MSB づめ（左寄せ）
 - LSB づめ（右寄せ）
 - PCM 標準（16bit チャネルフレームまたは 32bit チャネルフレームに拡張された 16bit データ・フレーム上のショート及びロングフレーム同期）
- データ方向は、常に MSB が先
- 送信及び受信（16bit）用 DMA
- 外部オーディオ部品を駆動するためのマスタクロックの出力。比率は 256xF_S 固定（F_S はオーディオサンプリング周波数）

22.3 SPI の機能詳細

22.3.1 概説

図 206 は、この SPI 回路のブロック図です。

図 206. SPI ブロック図



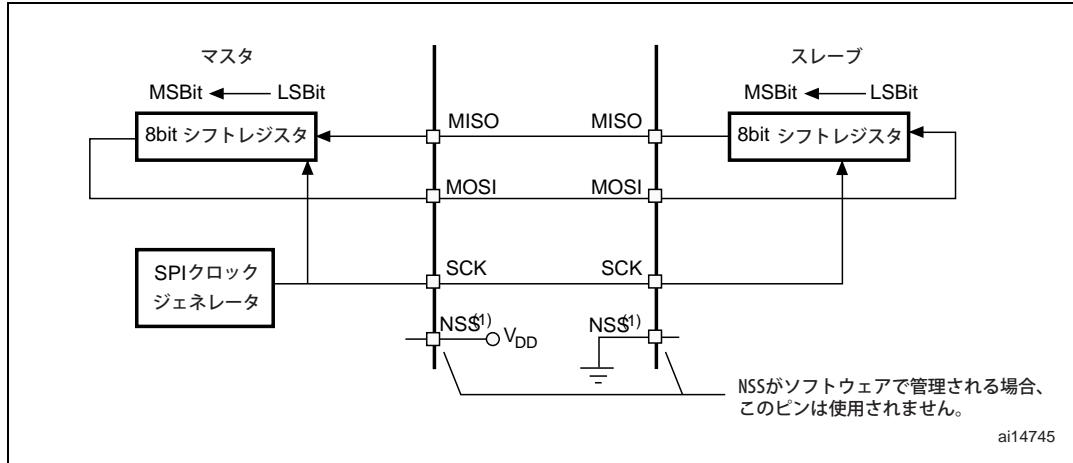
通常、SPI は 4 ピンで外部のデバイスと接続されます。

- MISO : Master In / Slave Out data – このピンはマスター・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されます。
- MOSI : Master Out / Slave In data – このピンはマスター・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されます。
- SCK : シリアル・クロック (Serial Clock) – SPI マスターではシリアル・クロックの出力に、スレーブではシリアル・クロックの入力に使用されます。
- NSS : スレーブ選択 (Slave Select) – このピンはマスター・モード / スレーブ・モードの選択に使用されるオプションピンです。SPI マスターが複数のスレーブと個別に通信を行うシステムでは、このピンはデータ線の競合を防ぐために「チップセレクト」のように動作します。この際、スレーブの NSS 入力はマスターデバイスの一般的な入出力ポートでドライブすることができます。また、SPI がマスター・モードのとき、SSOE ビットで NSS 出力イネーブルとされていれば、NSS ピンはローレベルの出力ピンとなります。このとき、マスターの NSS ピンに接続されているすべてのデバイスは、NSS ハードウェアモードに設定されている場合、このローレベルを受けてスレーブ・モードとなります。

1 個のマスターと 1 個のスレーブの間の基本的な接続は図 207 に示されています。

参考資料

図 207. 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 ピンは入力に設定されます。

MOSI ピン、MISO ピンはそれぞれが一緒につなぎ合わされます。この図では、マスタとスレーブの間で、データは（最上位ビットから）順に送られます。

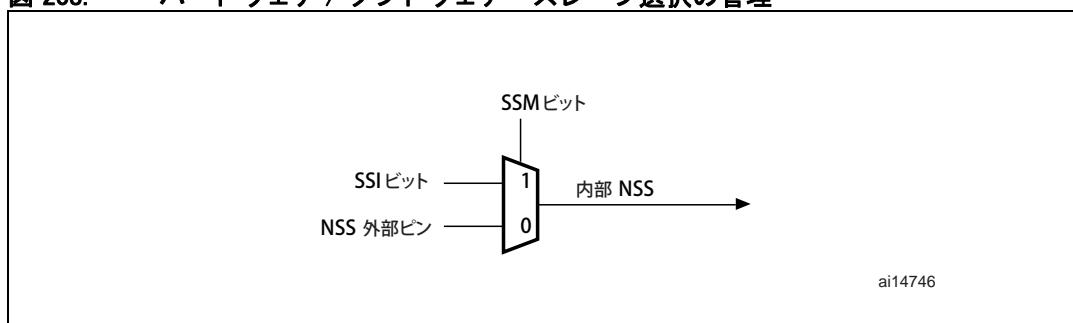
通信は常にマスタ主導で行われます。マスタデバイスがスレーブデバイスに MOSI ピンを通してデータを送信しているとき、スレーブデバイスは MISO ピンを通して応答を返します。これにより、マスタデバイスの SCK ピンから出力される同じクロックに同期した全二重通信が行われることになります。

スレーブ選択 (NSS) ピンの管理

NSS モードには 2 つあります。：

- ソフトウェア NSS モード：このモードは SPI_CR1 レジスタの SSM ビットをセットすることでイネーブルされます。（図 208 を参照）このモードでは、外部 NSS ピンはアプリケーションで自由に使うことができます。そして、内部 NSS 信号レベルは SPI_CR1 レジスタの SSI ビットに書込むことで駆動されます。
- ハードウェア NSS モード：2 つの場合があります：
 - NSS 出力はイネーブル：STM32F10xxx がマスタとして動作中で NSS 出力ピンが SPI_CR2 レジスタの SSOE ビットでイネーブルされていると、NSS ピンはロー状態に駆動されマスタ NSS ピンに接続されたデバイスの全ての NSS ピンは、ローレベルを認識し、それらが NSS ハードウェアモードに設定されているとスレーブとなります。
 - NSS 出力はディセーブル：複数マスタ機能が利用可能です。

図 208. ハードウェア / ソフトウェア・スレーブ選択の管理



クロックの位相と極性

SPI_CR1 レジスタの CPOL ビットと CPHA ビットの設定で、ソフトウェアにより、タイミングに関して 4 種類の選択が可能です。CPOL (クロック極性) ビットは、データが送られていない時のクロック信号の状態を定めます。この指定は、マスタ・モードとスレーブ・モードの両方で有効です。CPOL がクリアされているときには、アイドル状態の SCK ピンはローレベルとなります。CPOL がセットされていれば、アイドル状態の SCK ピンはハイレベルです。

CPHA (クロック位相) ビットがセットされているときには、SCK ピンの二番目のエッジ (CPOL がクリアされれば立下り、セットされれば立上りエッジ) で MSBit が読み込まれます。データは最初のクロック変化でラッチされます。CPHA ビットがクリアされれば、SCK ピンの最初のエッジ (CPOL がクリアされれば立上り、セットされれば立下りエッジ) が MSBit 読込みのストローブとなります。データは二番目のクロック変化でラッチされます。

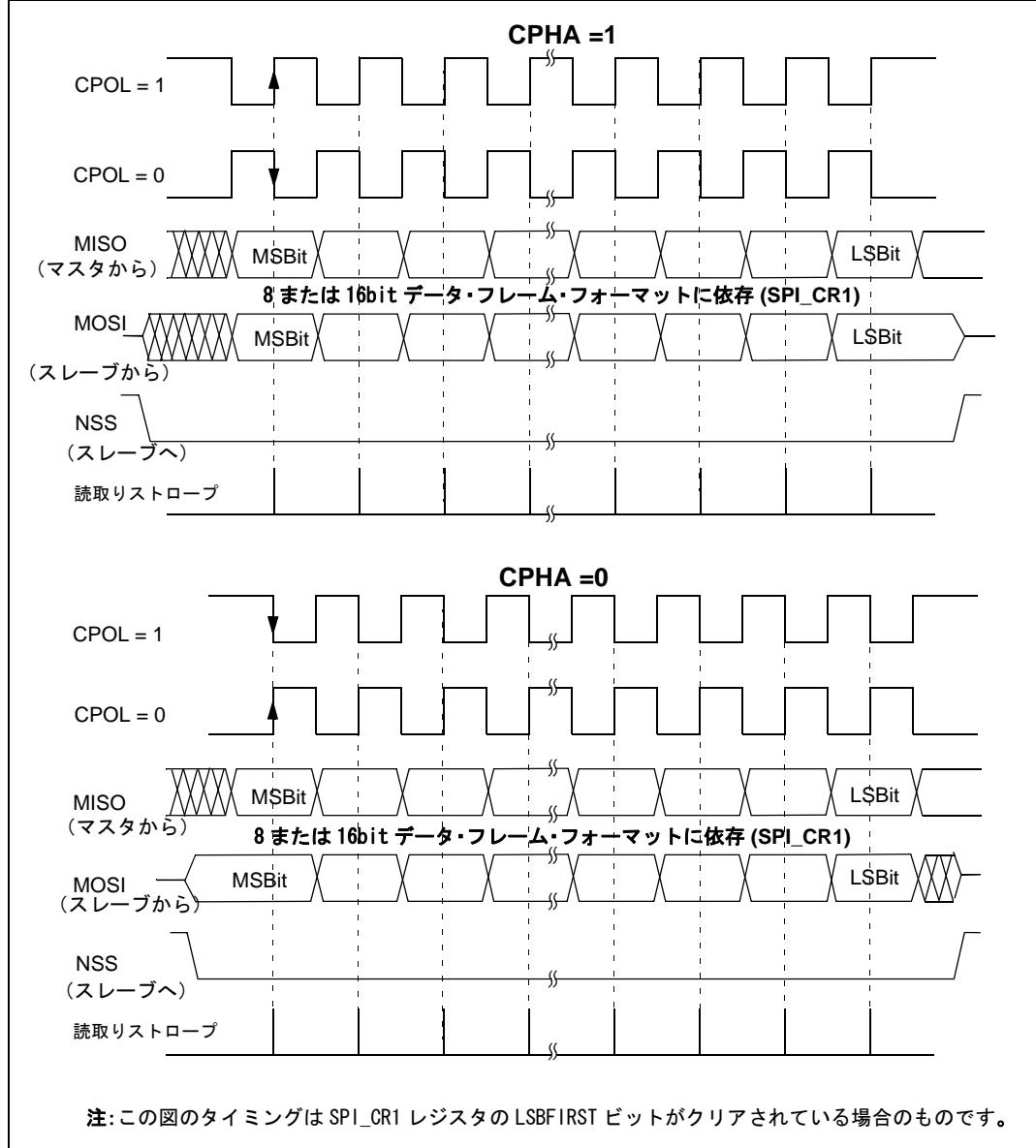
CPOL (クロック極性) と CPHA (クロック位相) の組合せで、データを読み込むクロックのエッジを選択することができます。

[図 209](#) は CPHA と CPOL ビットの組合せ 4 とおりの場合の SPI 転送の状況を示しています。この図は SCK ピン、MISO ピン、MOSI ピンがマスタ / スレーブ間でそれぞれ直接に接続されている場合のマスタもしくはスレーブのタイミング図ともなっています。

- 注意:
- 1 CPOL/CPHA ビットを設定する前に、SPE ビットをクリアすることで、SPI はディセーブル状態となっていかなければなりません。
 - 2 マスタとスレーブは、タイミングモードに関して同じ設定でなければなりません。
 - 3 アイドル状態における SCK は、SPI_CR1 レジスタの極性指定に依存します。(CPOL=1 であれば SCK はプルアップ状態、CPOL=0 であればプルダウン状態です。)
 - 4 データ・フレーム・フォーマット (8 または 16bit) は、SPI_CR1 レジスタの DFF ビットで指定されます。送受信の際のデータ長はこれによって決まります。

参考資料

図 209. データとクロックのタイミング



- これらのタイミングは SPI_CR1 レジスタの LSBFIRST ビットがクリアされている場合のものです。

データ・フレーム・フォーマット

SPI_CR1 レジスタの LSBFIRST ビットの設定によって、データが MSB から先に送出されるか、LSB から先に送出されるかを選択することができます。

個々のデータ・フレームは、SPI_CR1 レジスタの DFF ビットで指定されたデータ長に応じて、8bit もしくは 16bit なります。設定されたデータ・フレーム・フォーマットは、送受信の双方に適用されます。

参考資料

22.3.2 SPI スレーブ・モード

スレーブ・モードでは、マスタデバイスからのシリアル・クロックは SCK ピンから読み込まれます。SPI_CR1 レジスタの BR[1:0] に設定されている値はデータ転送の速度に影響しません。

設定

1. データ・フレームが 8bit か 16bit かを、DFF ビットに設定します。
2. 転送データとシリアル・クロックの 4 種の関係（[図 209 を参照](#)）のいずれかを CPOL ビットと CPHA ビットの組み合わせで選択します。データ転送が正しく行われるためには、CPOL ビットと CPHA ビットの設定が、スレーブデバイスとマスタデバイスで同じでなければなりません。
3. フレーム・フォーマット（SPI_CR1 レジスタの LSBFIRST ビットの値で指定される、MSB からの送信か、LSB からの送信かの設定）は、マスタデバイスと同じでなければなりません。
4. ハードウェアモード（[スレーブ選択 \(NSS\) ピンの管理 \(571 ページ\)](#) を参照）では、1 フレームの送信が完了するまでの間、NSS ピンはローレベルの信号に接続されなければなりません。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。
5. 所定のピンがオルタネート機能で使用できるよう、MSTR ビットをクリアし、SPE ビットをセットします。（これらはいずれも SPI_CR1 レジスタにあります。）

この設定で、MOSI ピンがデータ入力に、MISO ピンがデータ出力になります。

送信動作

書き込みサイクルにおいて、フレームデータは送信バッファにパラレルにロードされます。

送信動作はスレーブデバイスがクロック信号と、MOSI ピン上の最上位ビットデータを受けとるところから始まります。残りのビット（8bit フレームでは 7bit、16bit フレームでは 15bit）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られると、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れば割込みリクエストが発生します。

受信動作

受信動作においては、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送られ、SPI_SR レジスタの RXNE ビットがセットされます。
- SPI_CR2 レジスタの RXEIE ビットがセットされれば、割込みリクエストが発生します。

最後のサンプリング・クロック・エッジの直後に RXNE ビットがセットされ、シフト・レジスタ内の受信データが受信バッファに移されます。SPI_DR レジスタの読みしが行われると、SPI ペリフェラル回路はこのデータを戻します。

SPI_DR レジスタの読みしがにより、RXNE ビットはクリアされます。

22.3.3 SPI マスタ・モード

マスタ・モード設定のときには自らシリアル・クロックを生成し、これを SCK ピンに送出します。

設定

- シリアル・クロックの速度を決めるため BR[2:0] ビットを設定します。（SPI_CR1 レジスタを参照）
- 転送データとシリアル・クロックの 4 種の関係（図 209 を参照）のいずれかを CPOL ビットと CPHA ビットの組み合わせで選択します。
- データ・フレームが 8bit または 16bit を、DFF ビットに設定します。
- フレーム・フォーマットを指定するため、SPI_CR1 レジスタの LSBFIRST ビットを設定します。
- NSS ピンを入力モードとしておく場合、ハードウェアモードであれば、このピンをフレーム送信の間ハイレベルを保っているいずれかの信号と接続します。ソフトウェアモードのときには SPI_CR1 レジスタの SSM ビットと SSI ビットをセットします。 NSS ピンを出力モードとする必要があれば、SSOE ビットのみをセットします。
- MSTR ビットと SPE ビットはセットされていなければなりません。（これらのビットは NSS ピンがハイレベルに接続されているときにのみセット状態を維持します。）

この設定で、MOSI ピンがデータ出力に、MISO ピンがデータ入力になります。

送信動作

送信動作は、送信バッファにフレームデータを書込むことで開始されます。

データは、最初のビットの送出時に、内部バスを通してシフト・レジスタにパラレルに書込まれ、SPI_CR1 レジスタの LSBFIRST ビットの設定に応じて、上位ビットもしくは下位ビットから順に MOSI ピンに送出されます。送信バッファの内容がシフト・レジスタに移された時点で 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れば、割込みリクエストが発生します。

受信動作

受信動作においては、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送られ、RXNE ビットがセットされます。
- SPI_CR2 レジスタの RXEIE ビットがセットされいれば、割込みリクエストが発生します。

最後のサンプリング・クロック・エッジの直後に RXNE ビットがセットされ、シフト・レジスタ内の受信データが受信バッファに移されます。SPI_DR レジスタの読みしが行われると、SPI ペリフェラル回路はこのデータを戻します。

SPI_DR レジスタの読みしがにより、RXNE ビットはクリアされます。

いったん送信動作が開始されると、次のフレームとして送信されるデータが送信バッファに書込まれていれば、送信動作は連続的に行われます。送信バッファにデータを書込む際には、TXE フラグが 1 でなければならないことに注意してください。

22.3.4 一方向の通信

次の二種の環境で、SPI を一方向通信モードで使用することができます。

- 1 本のクロック線と 1 本の両方向データ線
- 1 本のクロック線と 1 本のデータ線（受信専用、もしくは全二重）

1 本のクロック線と 1 本の両方向データ線

このモードは SPI_CR1 レジスタの BIDIMODE ビットがセットされているときに有効になります。このモードでは SCK ピンはクロックとして、マスタにおける MOSI ピンもしくはスレーブにおける MISO ピンがデータの通信に使用されます。データの送信方向は SPI_CR1 レジスタの BIDIOE ビットによって制御され、このビットが 1 のときにデータピンは出力モードに、0 のときに入力モードになります。

1 本のクロック線と 1 本のデータ線（受信専用、もしくは全二重）

出入力ピンの使用を避け、このピンを他の目的に使用できるようにするために、SPI_CR1 レジスタの RXONLY ビットをセットすることで、SPI 回路のデータ出力機能を停止することができます。このとき、SPI 回路は受信のみのモードで動作します。RXONLY ビットがクリアされているときには、SPI 回路は全二重モードで動作します。

受信専用モードで通信をスタートするには、まず SPI 回路をイネーブル状態とすることが必要です。マスタ・モードでは、SPI 回路をイネーブル状態とすることで、通信は直ちに始まり、SPE ビットがクリアされ、その時点できわめてフレーム受信が完了すると通信は停止します。スレーブ・モードの場合は、NSS 信号がプルダウンされており（もしくは SSI ビットがクリアされており）SCK が働いている限り、SPI は受信動作を続けます。

注意： SPI_CR1 レジスタの RXONLY ビットがクリアされているときには、SPI 回路を送信専用として使用することができます。このとき、受信信号ピン（マスタ・モードでは MISO ピン、スレーブ・モードでは MOSI ピン）は GPIO として使用することができます。この場合、読み出し側のデータ・レジスタには受信データは読み込まれていません。

22.3.5 ステータス・フラグ

プログラムで SPI 接続の状態を知るため、3 個のステータス・フラグが用意されています。

ビジー・フラグ (BSY)

このフラグは SPI 回路の通信部の状態を示しています。このフラグがセットされているときには、SPI は通信中であるか、送信バッファに送出を待っているデータ格納していることをあらわします。このフラグは SPI バスが通信中であることを示すもので、次のタイミングでセットされます。

1. マスタ・モードで、データが SPI_DR レジスタに書込まれたとき。
2. スレーブ・モードで、SCK クロック（の有効エッジ）が検出されたとき。

ビジー・フラグは、フレームの送受信が完了したときにクリアされます。フラグのセット／クリアはハードウェアによって行われ、書込みの競合を避けるために利用することができます。このフラグへのソフトウェアによる書込みは無視されます。このフラグは、SPE ビットがセットされているときのみ有効な意味を持ちます。

注意： マスタ受信モード（1 ライン双方向）では、BUSY フラグをチェックしてはいけません。

送信バッファ・空きフラグ (TxE)

このフラグは送信バッファがからであり、次に送信されるべきデータを書込むことが可能であることを示しています。データが送信バッファに書込まれると、このフラグはクリアされます。また、このフラグは SPI 回路がディセーブル（SPE ビットがクリア）状態の時にもクリアされています。

受信バッファ・データフラグ (RXNE)

このフラグは有効な受信データが受信バッファに読込まれたことを示します。このフラグは SPI データ・レジスタがソフトウェアで読み出されるとクリアされます。

22.3.6 CRC の計算

CRC 計算機能は全二重モード（のみ）で、信頼度の高い通信を実現するために組込まれました。CRC 計算機能は、送信と受信のそれぞれに対して用意されています。CRC は各ビットにプログラムで設定された多項式を適用することで計算されます。CRC の計算は、SPI_CR1 レジスタの CPHA ビットと CPOL ビットで指定されるサンプリングエッジで行われます。

注意： このチップの SPI 回路では、データ・フレーム・フォーマットに従い、標準規格に準じた方式として、8bit データでは CR8、16bit データでは CRC16-CCITT をデフォルトで備えています。

CRC 計算機能は SPI_CR1 レジスタの CRCEN ビットをセットすることで有効となります。このビットをセットすることで、送受信 CRC レジスタ (SPI_RXCRCR and SPI_TXCRCR) をリセットします。SPI_CR1 レジスタの CRCNEXT ビットがセットされると、その時点で送信中のフレームに続いて SPI_TXCRCR レジスタの値が送信されます。

送信バッファにデータが存在すれば、このデータを送信後に CRC の値が送信されます。CRC の送信中は CRC の計算はオフとなり、CRC レジスタの値はそのまま残されます。

SPI_TXCRCR 値の送信時に、シフト・レジスタに受信された値が SPI_RXCRCR レジスタの値と一致していない場合には、SPI_SR レジスタの CRCERR フラグがセットされます。

注意： この機能が搭載されているか否かについては、個々の製品の仕様をご確認ください。

CRC を使用した SPI 通信は、次の操作で有効となります。

- 計算多項式を SPI_CRCPOLYR レジスタに設定します。
- SPI_CR1 レジスタの CRCEN ビットをセットし、CRC 計算機能をイネーブルとします。このとき、SPI_RXCRCR レジスタと SPI_TXCRCR レジスタはクリアされます。
- CPOL、CPHA、LSBfirst、DFF、BR、SSM、SSI、MSTR ビットの設定を行います。
- SPI_CR1 レジスタの SPE ビットをセットし、SPI をイネーブルとします。
- 通信を開始し、最後の 1 フレームを残して、すべてのフレームの送受信が終了するまで待ちます。
- 最後のフレームを送信バッファに書込むとともに、SPI_CR1 レジスタの CRCNext ビットをセットし、最終フレームの送信後に CRC を送出することを指示します。CRC の送出中は CRC の計算は行われません。
- 最終フレームの送出後に SPI は CRC を送出し、CRCNext ビットをクリアします。同時に、CRC が受信され、SPI_RXCRCR レジスタの値と比較されます。値が一致しないければ、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れば、割込みリクエストが発生します。

注意： ビットレート周波数が高いときには、CRC 送信に注意が必要です。CRC 送信フェーズにおける CPU サイクルはできるだけ少なくしなければならず、最終データ・フレームと CRC 受信におけるエラーを避けるため、CRC 送信手順におけるソフトウェア関数の呼び出しは禁止されます。

高い周波数のビットレートでは、CPU アクセスにより SPI 通信速度の低下することを避けるため、DMA モードの利用が強く勧められます。

22.3.7 DMA (direct memory addressing) を使用する SPI 通信

SPI で最大の速度を得るには、送信データを供給し、オーバランを生じることなく受信データを受信バッファから読み取らねばなりません。これを容易に実現するため、SPI 回路にはシンプルなリクエスト / アクノレッジ方式の DMA 機能が組込まれています。DMA は SPI_CR2 レジスタのイネーブル・ビットをセットすることで有効となります。送信バッファと受信バッファは、それぞれ独立した DMA リクエストを発生することができます。

CRC による DMA 機能

DMA モードで CRC 機能を使用する全二重モード SPI 通信が行われるとき、通信の最後にある CRC の送受信動作は、ソフトウェアが介在することなく自動的に行われます。

データ及び CRC の転送が終わり、転送中のデータに誤りが検出されると、SPI_SR レジスタの CRCERR ビットがセットされます。

22.3.8 エラー・フラグ

マスター・モード・フォールト (MODF)

マスター・モード・フォールトは、マスタデバイスの NSS ピンがローレベルにプルダウンされているか（ハードウェアモード）もしくはSSI ビットがクリアされている（ソフトウェアモード）ときに発生し、自動的に MODF ビットがセットされます。マスター・モード・フォールトは SPI 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れば割込みリクエストが発せられます。
- SPE ビットがクリアされます。これによりデバイスからのすべての出力が禁止され、SPI インタフェースはディセーブル状態となります。
- MSTR ビットがクリアされ、デバイスは強制的にスレーブ・モードとされます。

MODF ビットをクリアするには、次の手順を実行します。

1. MODF ビットがセットされているときに SPI_SR レジスタに読み出し、もしくは書き込みを行います。
2. 次に、SPI_CR1 レジスタへの書き込みを行います。

複数 MCU が存在するシステムで、スレーブ間の干渉を避けるために、MODF ビットをクリアする操作中は NSS ピンはハイレベルにプルアップされていかなければなりません。SPE ビットと MSTR ビットは、この手順と同時にもしくは手順実行後に、元の値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ときに SPE ビット及び MSTR ビットをセットすることはハードウェアで禁止されています。

スレーブデバイスでは MODF ビットがセットされることはありません。複数マスターのシステムでは MODF ビットがセットされた状態で、デバイスがスレーブ・モードとなりことがあります。このとき、MODF ビットは複数のマスターの間で競合が生じた可能性を示しています。この状態から抜けるために、割込み処理ルーチン内でリセット動作やデフォルト状態への復帰を行うことができます。

オーバラン状態

マスタデバイスがデータを送信したときに、スレーブデバイスで前のデータによりセットされた RXNE ビットがいまだクリアされていないと、オーバラン状態が発生します。このとき、

- OVR ビットがセットされ、ERRIE ビットがセットされていれば割込みリクエストが生じます。

この場合、受信バッファにはマスタデバイスからの新しいデータは格納されていません。受信バッファから読み出されるのは前のデータです。これ以降の送信データは失われます。

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で行われます。

CRC エラー

このフラグは SPI_CR1 レジスタの CRCEN ビットがセットされているときに、受信されたデータの有効性を示しています。全二重モードでは SPI_SR レジスタの CRCERR ビットは、(SPI_TXCRCR レジスタの値が送出された後に) シフト・レジスタに受信された値が SPI_RXCRCR レジスタの値と一致しないときにセットされます。

22.3.9 SPI のディセーブル

転送が終了されると、アプリケーションは SPI インタフェースをディセーブルし通信を停止することができます。これは SPE ビットをリセットすることで行われます。最後のデータ転送が進行中の間に SPI ペリフェラルをディセーブルすることは、デバイスがマスタ送信モードでなければデータの信頼性に影響はありません。

注意： マスタ送信モード（全二重または一方向送信のみ）では、アプリケーションは SPI マスタをディセーブルする前に SPI_SR レジスタの BSY フラグをチェックし、データ転送が進行中でないことを確認しなければなりません。

22.3.10 SPI 割込み

表 145. SPI 割込みリクエスト

割込みイベント	イベントフラグ	割込みリクエスト許可ビット
送信バッファ・空きフラグ	TXE	TXEIE
受信バッファ・データフラグ	RXNE	RXNEIE
マスタ・モード・フォールト・イベント	MODF	
オーバラン・エラー	OVR	ERRIE
CRC エラー・フラグ	CRCERR	

22.4 I²S 機能詳細

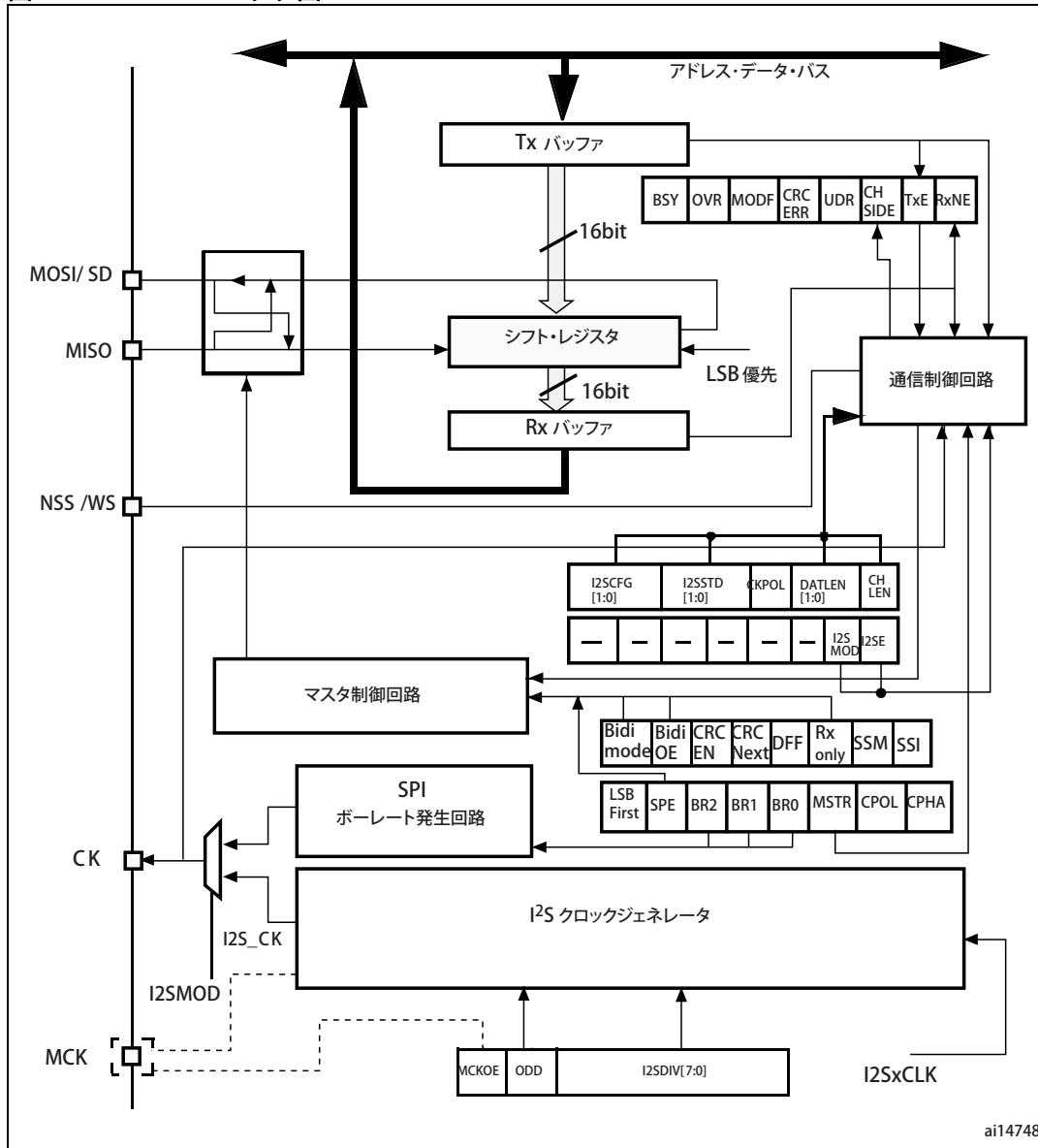
I²S オーディオ・プロトコルは、中容量デバイスでは利用できません。このセクションは大容量デバイスについてのみ適用されます。

22.4.1 詳細

図 210 は、I²S 回路のブロック図です。

参考資料

図 210. I²S ブロック図



ai14748

SPI は I²S 機能がパイネーブルされていると (SPI_I2SCFGR レジスタの I²SMOD ビットをセット)、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。このインターフェースは SPI とほぼ同じピン、フラグ及び割込みを使います。

- SD : 2 つの時間多重化データチャネル (1 方向モードのみ) を送受信するシリアルデータ (MOSI ピン上に配置された) です。
- WS : ワードセレクト (NSS ピン上に配置された) はデータ制御信号入出力 (マスタ・モード時出力、スレーブ・モード時入力) です。
- CK : シリアル・クロック (SCK ピン上に配置された) はシリアル・クロック入出力 (マスタ時出力、スレーブ時入力) です。

幾つかの外部デバイス用にマスタクロック出力が必要な場合、追加のピンを使う事ができます。：

- MCK : I²S がマスタ・モードに設定されているとき（そして 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ル周波数) 設定済み周波数で生成された追加のクロックを出力するために、マスタクロック（別に配置された）が使われます。

I²S はマスタ・モードにセットされているとき、自身のクロック・ジェネレータを使い通信クロックを作り出します。このクロック・ジェネレータはマスタクロック出力のソースとしても使われます。I²S では 2 つの追加レジスタが利用できます。1 つはクロック・ジェネレータ設定 SPI_I2SPR とリンクしていて、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、クロック極性その他）です。

SPI_CR1 レジスタと全ての CRC レジスタは、I²S モードでは使われません。同様に、SPI_CR2 の SSOE ビットと SPI_SR の MODF 及び CRCERR ビットも使われません。

I²S は同じ SPI レジスタを 16bit 幅モードでデータ転送 (SPI_DR) に使用します。

22.4.2 サポートされるオーディオ・プロトコル

3 ラインバスは一般的に 2 つのチャネル上で時間多重されたオーディオデータのみを扱わなければなりません。：右チャネルと左チャネルです。しかしながら、送信または受信用に 1 つの 16bit レジスタしかありません。つまり対応するチャネルに適した値をデータ・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、SPI_SR レジスタの CHSIDE ビットをチェックする事で対応するチャネルを識別するのはソフトウェアに任せられます。左チャネルは、常に右チャネルの後に送信されます。(CHSIDE は PCM プロトコルでは意味を持ちません)

4 つのデータとパケットフレームが利用できます。データは次のフォーマットで送信されます：

- 16bit フレーム内の 16bit データ
- 32bit フレーム内の 16bit データ
- 32bit フレーム内の 24bit データ
- 32bit フレーム内の 32bit データ

32bit パケットに拡張された 16bit データを使うとき、ソフトウェアや DMA リクエスト（読み込み / 書込み動作のみ）の介在なしに最初の 16bit (MSB) は上位ビット、16bit LSB は強制的に 0 になります。

24bit と 32bit データ・フレームは、アプリケーションが DMA を使う場合 SPI_DRとの間で 2 つの読み込みまたは書込み動作、または 2 つの DMA 動作を必要とします。24bit データ・フレームに限っては、8 つの下位ビットはハードウェアにより 0 により 32bit へ拡張されます。

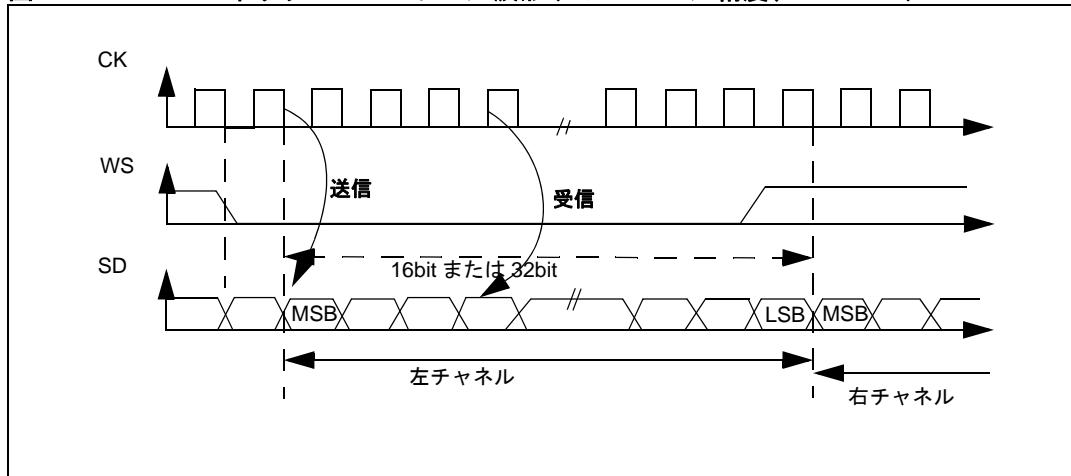
全てのデータ・フォーマットと通信標準において、最上位ビットは常に最初に送信されます。(MSB が先)

I²S インターフェースは、SPI_I2SCFGR レジスタの I2SSTD[1:0] と PCMSYNC ビットを使い 4 つのオーディオ標準をサポートします。

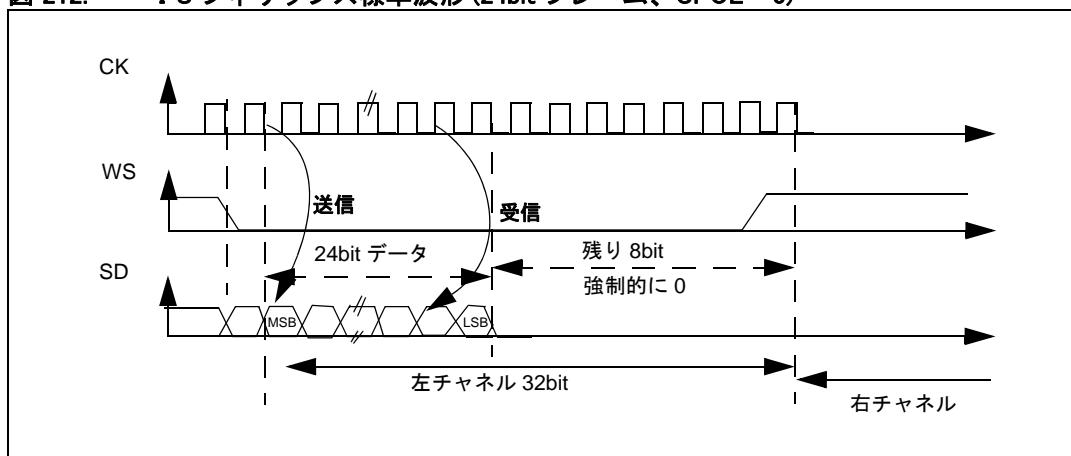
I²S フィリップス標準

この標準では、WS 信号を使いどちらのチャネルが送信されているか表します。それは最初のビットが有効になる 1CK クロック・サイクル手前でアクティブとなります。

参考資料

図 211. I²S フィリップスプロトコル波形 (16/32bit フル精度、CPOL = 0)

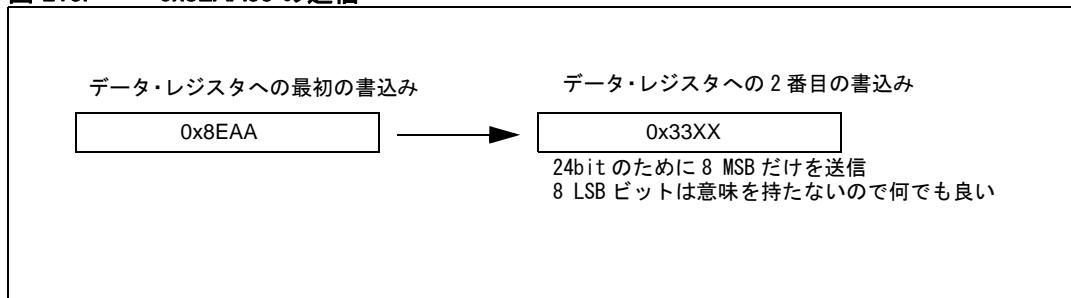
データは CK の立下りエッジ（送信時）でラッチされ、立上がりエッジで読み出されます。
（受信側）WS 信号も CK の立下りエッジでラッチされます。

図 212. I²S フィリップス標準波形 (24bit フレーム、CPOL = 0)

このモードでは SPI_DR にたいして 2 つの書き込みまたは読み込み動作が必要です。

- 送信モードでは：
8EAA33 を送信しなければならない場合 (24bit) :

図 213. 0x8EAA33 の送信



参考資料

- 受信モードでは：
データ 0x8EAA33 が受信されると：

図 214. 0x8EAA33 の受信

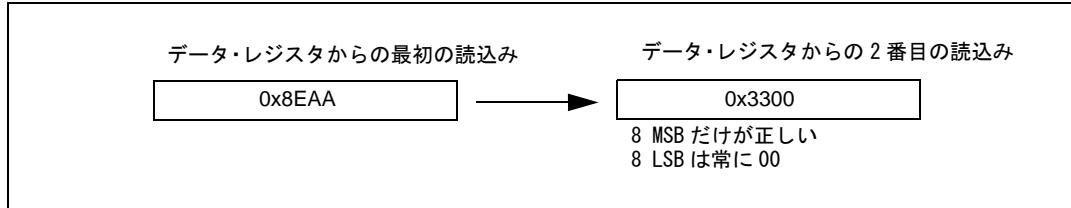
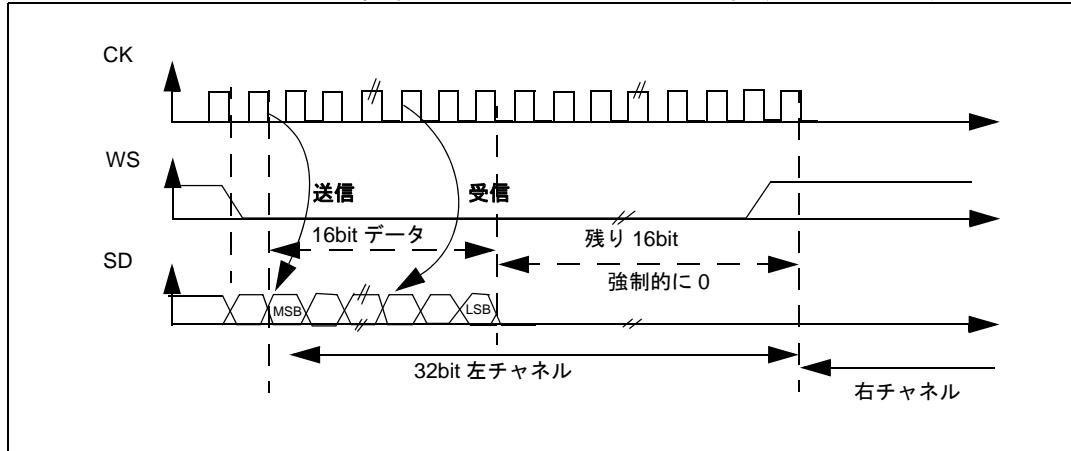


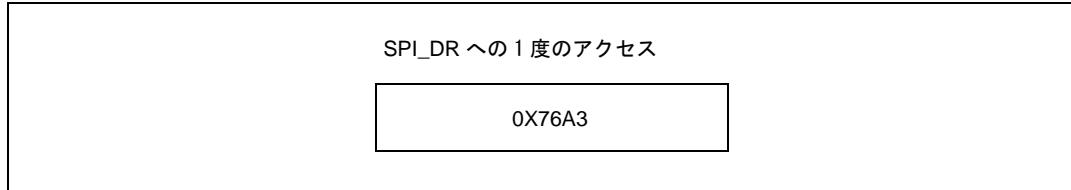
図 215. I²S フィリップス標準(32bit パケットフレームに拡張された 16bit、CPOL = 0)



I²S 設定フェーズで、16bit データ・フレームの 32bit チャネルフレームへの拡張が選択されたとき、1つの SPI_DR アクセスだけが必要です。残りの 16bit は、データを 32bit フォーマットに拡張するために、ハードウェアにより強制的に 0x0000 となります。

送信するデータまたは受信したデータが 0x76A3 (32bit に拡張された 0x76A30000) であるとき必要な動作を [図 216](#) に示します。

図 216. 例



送信では SPI_DR に MSB が書込まれる度に、SPI_DR に新たな値をロードするために、TXE フラグがセットされ、可能であれば割込みが生成されます。これはハードウェアにより行われるので、0x0000 がまだ送信されていなくとも行われます。

受信では最初の 16MSB ハーフワードが受信されると RXNE フラグがセットされ可能であれば割込みが生成されます。

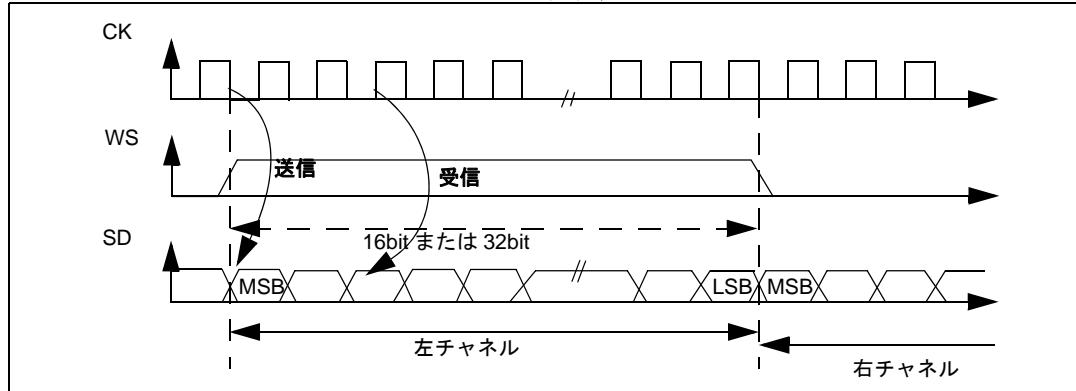
この方法では 2 つの書き込みあるいは読み込み動作間で、より多くの時間を規定することでアンダーランまたはオーバーラン状態（データ転送の方向により）を防ぎます。

参考資料

MSB づめ標準

この標準では WS 信号は、MSB ビットである最初のデータビットと同時に生成されます。

図 217. MSB づめ 16bit または 32bit フル精度長、CPOL = 0



データは CK の立下りエッジでラッチ（送信時）、または立上がりエッジで読み出されます。（受信時）

図 218. MSB づめ 24bit フレーム長、CPOL = 0

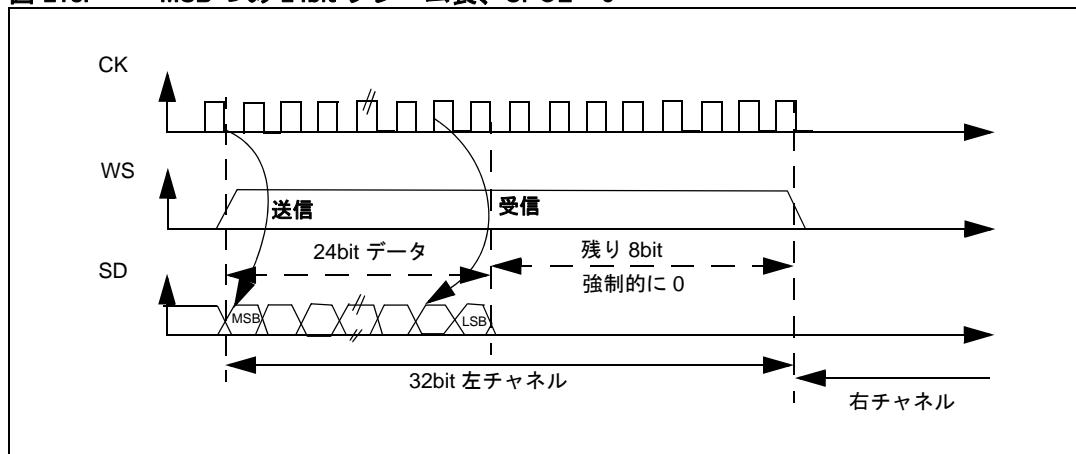
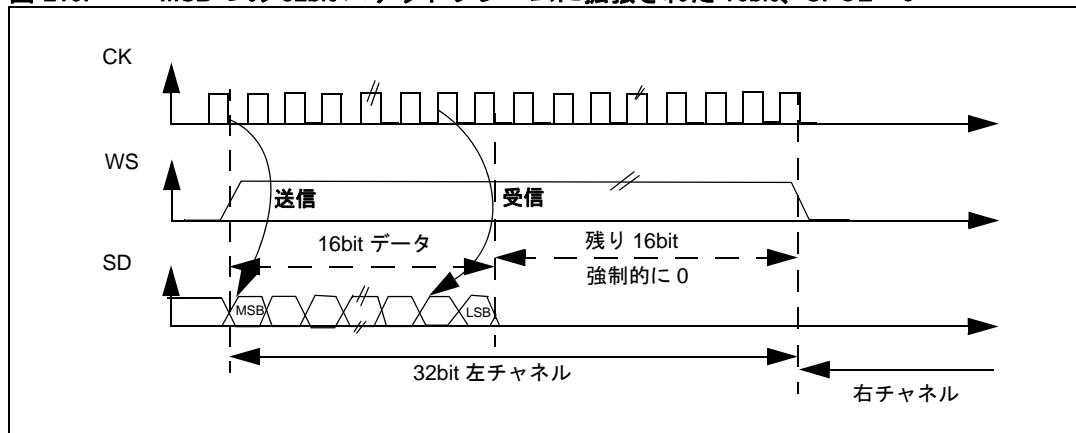


図 219. MSB づめ 32bit パケットフレームに拡張された 16bit、CPOL = 0



参考資料

LSB づめ標準

この標準は MSB づめ標準と似通っています。(16bit と 32bit 高精度フレーム・フォーマットに違いはありません)

図 220. LSB づめ 16bit または 32bit フル精度、CPOL = 0

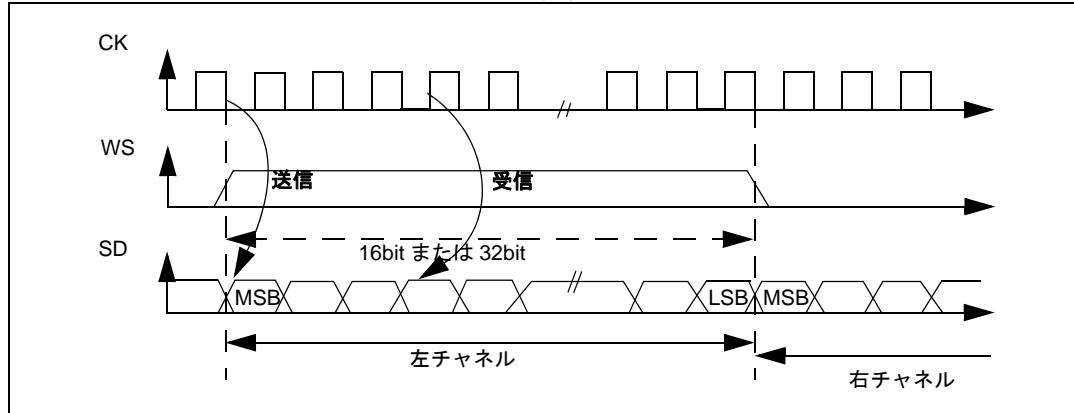
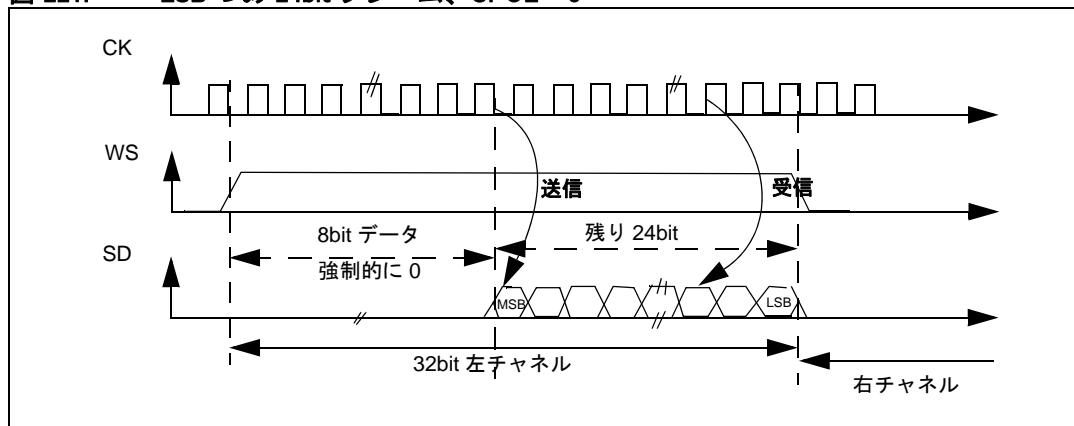


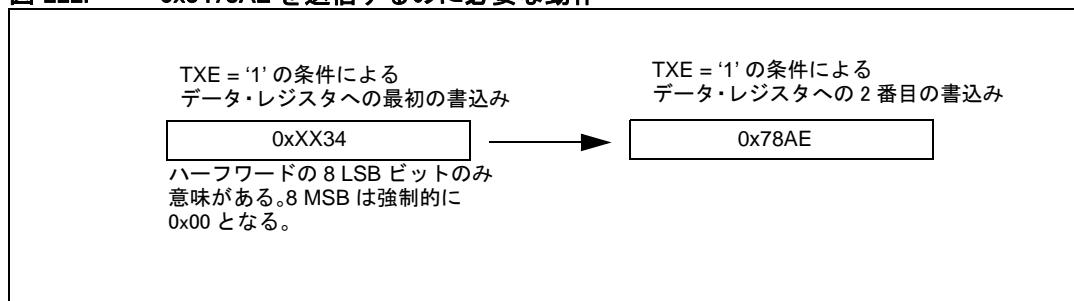
図 221. LSB づめ 24bit フレーム、CPOL = 0



- 送信モードでは :

データ 0x3478AE を送信しなければならない場合、ソフトウェアあるいは DMA による 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 222. 0x3478AE を送信するのに必要な動作



- 受信モードでは :

データ 0x3478AE が受信されると、各 RXNE イベント毎に SPI_DR への 2 つの読み込み動作が成功する必要がある。

参考資料

図 223. 0x3478AE の受信に必要な動作

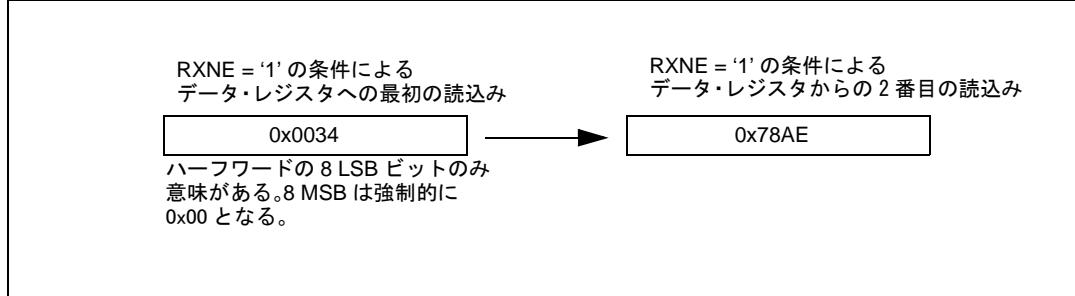
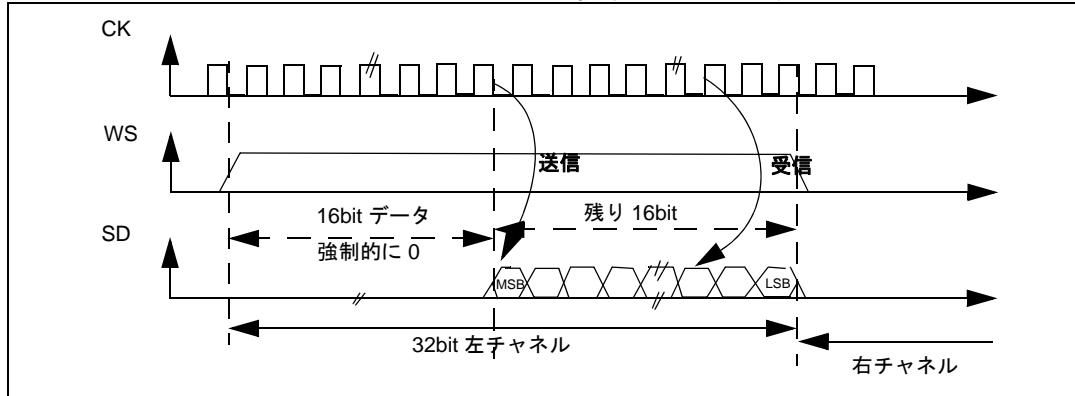


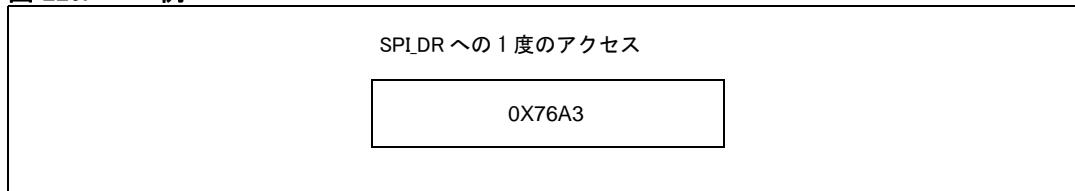
図 224. LSB づめ 32bit パケットフレームへ拡張された 16bit、CPOL = 0



32bit のチャネル・フレームに拡張された 16bit のデータ・フレームが、I²S 設定フェーズ中に選択されている場合、SPI_DR への 1 度のアクセスだけが必要です。残りの 16bit は、ハードウェアによって強制的に 0x0000 となりデータは 32bit フォーマットへ拡張されます。この場合、それはハーフワード MSB に相当します。

送信データあるいは受信データが 0x76A3(32bit まで延長された 0x0000 76A3) である場合、図 225 で示される動作が必要です。

図 225. 例



送信モードで、TXE がアサートされたとき、アプリケーションは送信する（この場合 0x76A3）データを書かなければなりません。0x000 フィールドは、最初に (32bit 上の拡張) 送信されます。有効なデータ (0x76A3) が SD の上で送られるとすぐに、TXE は再びアサートされます。

受信モードにおいて、上位ハーフワードが受け取られる（かつ 0x0000 フィールドではない）とすぐに、RXNE はアサートされます。

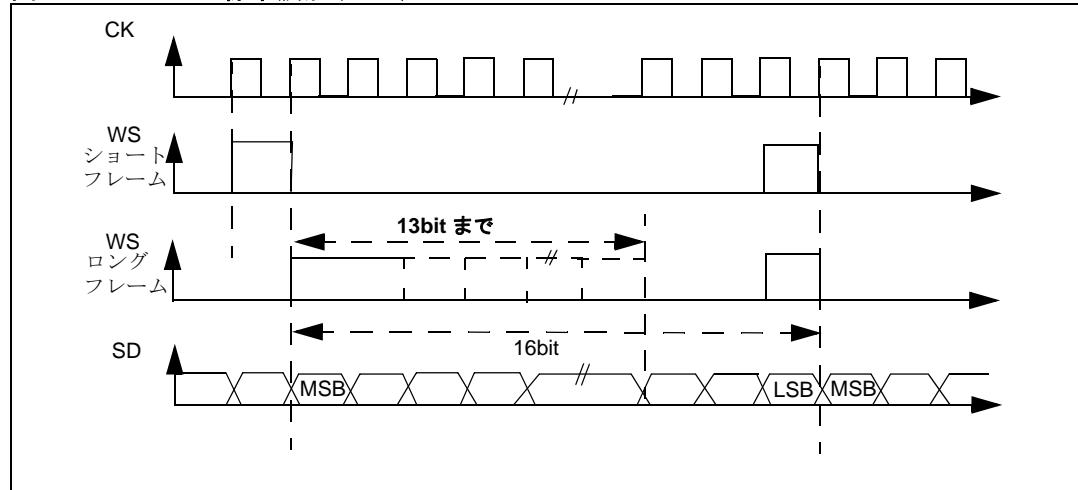
このように 2 つの書き込みあるいは読み込み動作間で、より多くの時間を規定することでアンダーランまたはオーバーラン状態を防ぎます。

参考資料

PCM 標準

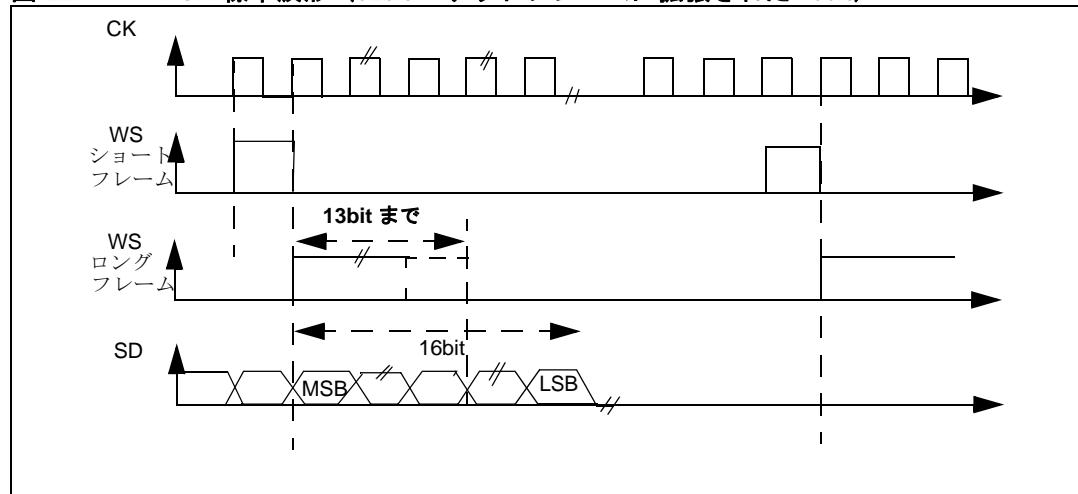
PCM 標準では、チャネルサイド情報を使用する必要はありません。SPI_I2SCFGR の PCMSYNC ビットを使用して、2つのPCM モード（ショート及びロングフレーム）を設定利用することができます。

図 226. PCM 標準波形 (16bit)



ロングフレーム同期については、WS 信号アサーション時間はマスタ・モードで 13bit 固定です。
ショートフレーム同期については、WS 同期信号は 1 サイクル長のみです。

図 227. PCM 標準波形 (32bit パケットフレームに拡張された 16bit)



注意： 両方のモード（マスターとスレーブ）、及び両方の同期（ショート及びロング）については、
スレーブ・モードでさえ連続した 2 つのデータ（2 つの同期信号）間のビットの数を指定す
る必要があります (SPI_I2SCFGR レジスタの DATLEN と CHLEN ビット)。

参考資料

22.4.3 クロック・ジェネレータ

I²S ビットレートは、I²S データライン上のデータフローと I²S クロック信号周波数を決定します。

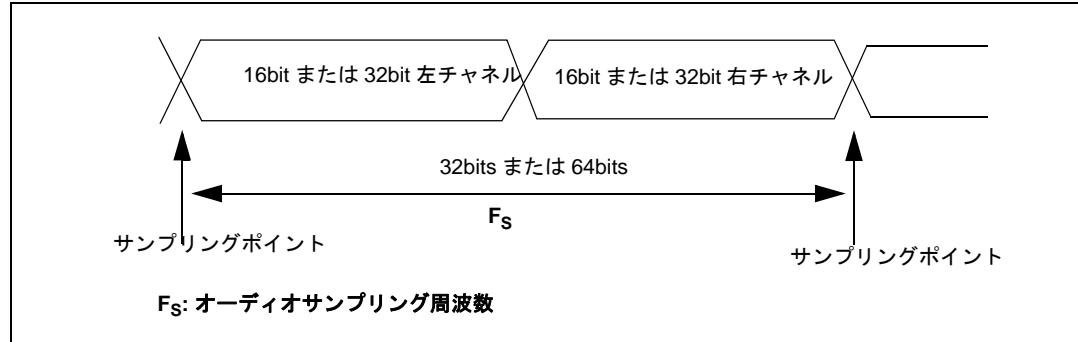
I²S ビットレート = チャネル毎のビット数 × チャネル数 × サンプリングオーディオ周波数

16bit オーディオ、左と右チャネル、I²S ビットレートは次のように計算されます：

$$I^2S \text{ ビットレート} = 16 \times 2 \times F_S$$

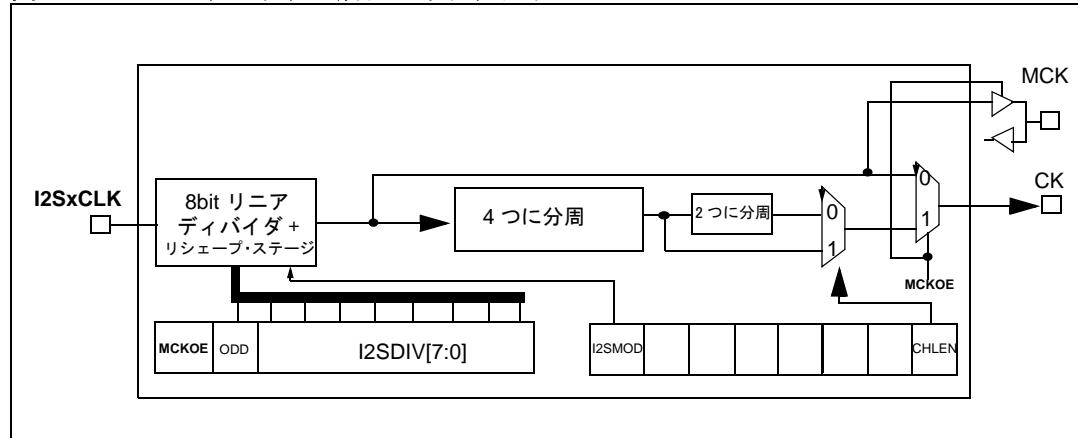
パケット長が 32bit 幅の場合、それは I²S ビットレート = $32 \times 2 \times F_S$ となります。

図 228. オーディオサンプリング周波数定義



マスタ・モードが設定される場合、希望のオーディオ周波と通信するためにリニアディバイダーを正しくプログラムするために特別なアクションを行う必要があります。

図 229. I²S クロック生成アーキテクチャ



1. x は 2 または 3 となり得る

図 228 は通信クロック構造を示しています。I2SxCLK ソースはシステム・クロックです。(HSI、HSE または PLL により提供され AHB クロックを供給する)

オーディオサンプリング周波数は 48 kHz、44.1 kHz、22.05 kHz、16 kHz または 8 kHz です。希望する周波数に達するには、リニアディバイダは次の式によりプログラムされる必要があります。：

マスタクロックが生成されるとき (SPI_I2SPR レジスタの MCKOE ビットをセット) :

$$F_S = I2SxCLK / [(16*2)*((2*I2SDIV)+ODD)*8)] \text{ チャネルフレームが 16bit 幅の場合}$$

$$F_S = I2SxCLK / [(32*2)*((2*I2SDIV)+ODD)*4)] \text{ チャネルフレームが 32bit 幅の場合}$$

マスタクロックがディセーブルのとき (MCKOE ビットはクリア) :

$$F_S = I2SxCLK / [(16*2)*((2*I2SDIV)+ODD)] \text{ チャネルフレームが } 16\text{bit 幅の場合}$$

$$F_S = I2SxCLK / [(32*2)*((2*I2SDIV)+ODD)] \text{ チャネルフレームが } 32\text{bit 幅の場合}$$

22.4.4 I²S マスタ・モード

I²S はマスタ・モードに設定できます。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、出力となる時もあればそうでない場合もあります。

設定

- 正しいオーディオサンプリング周波数を得るためにシリアル・クロックの速度を決めるため SPI_I2SPR レジスタの I2SDIV[7:0] ビットをセレクトします。
- 通信クロック用の安定したレベルを定義するために CKPOL ビットをセレクトします。マスタクロック MCK を外部 DAC/ADC オーディオ部品に供給する必要があるならば、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される必要があります。詳細は [セクション 22.4.3: クロック・ジェネレータ](#) を参照
- SPI_I2SCFGR の I2SMOD ビットをセットして I²S 機能をアクティベートし、I2SSSTD[1:0] と PCMSYNC ビットを通じて I²S 標準を、DATLEN[1:0] を通じてデータ長を、そして CHLEN ビットを設定しチャネル毎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（送信または受信）もセレクトします。
- 必要があれば、SPI_CR2 レジスタに書込むことにより全ての起こりうる割込みソース及び DMA 機能をセレクトします。
- SPI_I2SCFGR レジスタの I2SE ビットをセットしなければなりません。

WS と CK は出力モードに設定します。SPI_I2SPR の MCKOE ビットがセットされていると MCK も出力となります。

送信手順

ハーフワードが送信バッファに書かれたとき、送信手順は始まります。

送信バッファに書込まれた最初のデータが、左チャネルデータに対応すると仮定します。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き、TXE はセットされます。そして送信バッファに右チャネルに対応するデータを書かなければなりません。CHSIDE フラグは、どちらのチャネルが送信されるかを表します。CHSIDE フラグは TXE がハイ状態となったとき更新されるので、それは TXE フラグがセットされたとき意味を持ちます。

フルフレームとは左チャネル・データ送信と後続の右チャネル・データ送信を考える必要があります。左チャネルだけが送信された部分的なフレームというのは不可能です。

データハーフワードは、最初のビットの送信時に 16bit シフト・レジスタに同時にロードされ、シリアルに MOSI/SD ピンに MSB から先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・レジスタへの各送信後にセットされ、SPI_CR2 レジスタの TXEIE ビットがセットされていると割込みが生成されます。

セレクトされた I²S 標準モードに応じた書き込み動作の詳細は [セクション 22.4.2: 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の送信を行うには、現在の送信が終了する前に、SPI_DR に次の送信データを書きこむ事が必須です。

参考資料

I²S をクリアすることで I²S をスイッチオフするために、TXE = 0 と BSY = 0 を待つことが必要です。

受信手順

動作モードは、I2SCFG[1:0] ビットを通じてマスタ受信モードを設定するというポイント 3 を除いて送信モードと同じです。そこでは、設定は I2SCFG[1:0] ビットによってマスタ受信モードをセットするべきです。

どのようなデータまたはチャネル長でも、オーディオデータは 16bit のパケットによって受信されます。これは、受信バッファがフルとなる毎に 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る場合、割込みが生成されることを意味します。データ及びチャネル長の設定により、右左のチャネルのために受信したオーディオ値は、受信バッファへの 1 つあるいは 2 つの受信により行われるかもしれません。

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 は各受信後に更新されます。それは、I²S セルにより生成される WS 信号に影響されやすいといえます。

セレクトされた I²S 標準モードに応じた読み込み動作の詳細は [セクション 22.4.2: 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込まれていない間にデータが受け取られた場合、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ると、エラーを表示するために割込みが発生します。

I²S 受信モードをスイッチオフするためには、データ受信中及び最後のデータ受信前に I2SE をクリアしなければなりません。最後のデータが転送されている最中に I2SE がスイッチオフされたとしても、現在のデータ送信が終わるまでクロックと転送は継続されます。

22.4.5 I²S スレーブ・モード

スレーブ設定については、I²S は送信または受信モードに設定することができます。

動作モードは主として I²S マスター設定のために記述されるのと同じ規則に従っています。

スレーブ・モードでは、I²S インタフェースにより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る外部マスタからの入力です。ユーザがクロックを設定する必要はありません。

設定の手順を次に示します。：

1. I²S 機能のために SPI_I2SCFGR レジスタの I2SMOD ビットをセットし、I2SSTD[1:0] ビットを通じて I²S 標準を選択、DATLEN[1:0] を通じてデータ長を、そして CHLEN ビットを設定しフレームのチャネル毎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、スレーブ・モードと方向（送信または受信）もセレクトします。
2. 必要があれば、SPI_CR2 レジスタに書込むことにより全ての起こりうる割込みソース及び DMA 機能をセレクトします。
3. SPI_I2SCFGR レジスタの I2SE ビットをセットしなければなりません。

送信手順

ハーフワードが送信バッファに書かれたとき、送信手順は始まります。送信バッファに書込まれた最初のデータが、左チャネルデータに対応すると仮定します。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き、TXE はセットされます。そして送信バッファに右チャネルに対応するデータを書かなければなりません。CHSIDE フラグは、どちら

参考資料

らのチャネルが送信されるかを表します。スレーブ・モードでは、CHSIDE は外部マスターからの WS 信号に影響されやすいといえます。これはスレーブがマスターによりクロックが生成される前に、最初のデータ送信準備を整えておく必要があることを意味します。左チャネルに対応する WS アサーションが先に送信されます。

注意: I²S は、マスターの初めのクロックが CK 上に来る少なくとも 2PCLK サイクル前に、書かれなければなりません。

データハーフワードは、最初のビットの送信時に 16bit シフト・レジスタに同時にロードされ、シリアルに MOSI/SD ピンに MSB から先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・レジスタへの各送信後にセットされ、SPI_CR2 レジスタの TXEIE ビットがセットされていると割込みが発生します。

送信バッファへの書き込みの前に、TXE フラグが 1 となることを確認するべきであることに注意してください。

セレクトされた I²S 標準モードに応じた書き込み動作の詳細は [セクション 22.4.2: 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の送信を行うには、現在の送信が終了する前に、SPI_DR に次の送信データを書きこむ事が必須です。次のデータ通信の最初のクロックエッジの前にデータが SPI_DR レジスタへ書き込まれないと、アンダーランフラグがセットされ割込みが生成されます。これは、ソフトウェアに対して送信されたデータが、誤ったものである事を表します。SPI_CR2 レジスタの RXNEIE ビットがセットされると、SPI_SR レジスタの UDR フラグがハイ状態となると割込みが発生します。この場合、I²S をスイッチオフし左チャネルからデータ転送をリスタートしなければなりません。

受信手順

動作モードは、SPI_I2SCFGR レジスタの I2SCFG[1:0] ビットを通じてマスター受信モードを設定するというポイント 1 を除いて送信モードと同じです。

どのようなデータまたはチャネル長でも、オーディオデータは 16bit のパケットによって受信されます。これは、受信バッファがフルとなる毎に 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る場合、割込みが発生することを意味します。データ及びチャネル長の設定により、右左のチャネルのために受信したオーディオ値は、受信バッファへの 1 つあるいは 2 つの受信により行われるかもしれません。

CHSIDE は各受信後に更新されます。それは、外部マスター部品により管理される外部 WS ラインに影響されやすいといえます。

RXNE ビットのクリアは SPI_DR レジスタを読み込むことにより行われます。

セレクトされた I²S 標準モードに応じた読み込み動作の詳細は [セクション 22.4.2: 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込まれていない間にデータが受け取られた場合、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 RXNEIE ビットがセットされると、エラーを表示するために割込みが発生します。

I²S 受信モードをスイッチオフするためには、データ受信中及び最後のデータ受信前に I²SE をクリアしなければなりません。最後のデータが転送されている最中に I²SE がスイッチオフされたとしても、現在のデータ送信が終わるまでクロックと転送は継続されます。

注意: 外部マスター部品は、オーディオチャネルを経由し 16bit または 32bit パケットでデータの送受信可能でなければなりません。

22.4.6 ステータス・フラグ

3つのステータス・フラグが、I²S バスの状態を完全にモニタするためにアプリケーションに提供されます。

ビジー・フラグ (BSY)

このフラグは I²S 通信レイヤーの状態を表します。それは、I²S が通信するにはビジーであるそして / または送信バッファに有効なデータハーフワードが送信待ちである事を表すためにセットされます。このフラグの目的は I²S バス上に通信が実行中であるか否かを表すことです。このフラグは次の場合にセットされます。：

1. マスタ・モードで SPI_DR レジスタにデータが書込まれた場合
2. スレーブ・モードで CK クロックがある場合

ビジー・フラグはハーフワードが送信 / 受信されると直ぐにリセットされます。それはハードウェアによりセット / リセットされます。このフラグは書き込み衝突エラーを防ぐためにモニターできます。フラグへの書き込みは影響しません。このフラグは、SPI_I2SCFGR レジスタの I2SE ビットがセットされている時に限り意味を持ちます。

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、送信バッファがエンプティ（空）であり、次に送信されるデータがロード可能であることを表します。TXE フラグは、送信バッファに送信データが含まれるときにリセットされます。また、I²S がディセーブルされたときにもリセットされます。(I2SE ビットはリセット)

受信バッファエンプティフラグ (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表します。このフラグは、SPI_DR レジスタが読み込ま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 がハイ状態となるとリフレッシュされます。それは SD 上のデータが、どちらのチャネルサイドに属するものか表します。スレーブ送信モードのアンダーランエラーイベントの場合、このフラグは信頼性に欠けます。そして通信を再開するためには、I²S をスイッチオフ、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。それは SD 上のデータが、どちらのチャネルサイドで受信されたかを表します。エラー (OVR 等) が発生した場合、このフラグは意味を持たず、I²S はディセーブルしイネーブルすることでリセット（変更が必要なら設定する）しなければならない事に注意してください。

このフラグは、PCM 標準では意味を持ちません。（ショートとロングフレームの両モード）

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、割込みが発生します。この割込みは SPI_SR ステータス・レジスタを読み出すことでクリアできます。（いったん割込みソースがクリアされると）

22.4.7 エラー・フラグ

I²S セルには 2 つのエラー・フラグがあります。

アンダーランフラグ (UDR)

スレーブ転送モードでは、ソフトウェアが SPI_DR にまだ値をロードしていない間に、データ送信のための最初のクロックが現われる場合、このフラグはセットされます。それは SPI_I2SCFGR の I2SMOD ビットがセットされると利用できます。SPI_CR2 の ERRIE ビットがセットされると割込みが発生します。

UDR ビットは SPI_SR レジスタ上の読み込み動作によりクリアされます。

オーバーランフラグ (OVR)

このフラグは、データが受信され、先のデータが SPI_DR からまだ読み出されていないときにセットされます。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ると割込みが発生します。

この場合、受信バッファの内容は送信デバイスからの最新データでアップデートされません。SPI_DR レジスタへの読み込み動作は、先の正常受信データを返します。それ以降の全ての送信ハーフワードは失われます。

OVR ビットのクリアは、SPI_DR レジスタの読み込み動作と続く SPI_SR レジスタへの読み込みアクセスにより行われます。

22.4.8 I²S 割込み

表 146 は I²S 割込みのリストです。

表 146. I²S 割込みリクエスト

割込みイベント	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・エンプティでないフラグ	RXNE	RXNEIE
オーバーラン・エラー	OVR	ERRIE
アンダーラン・エラー	UDR	

22.4.9 DMA 機能

DMA は SPI モードと全く同じ動作をします。I²S 上に何も違いはありません。I²S モードではデータ転送保護システムがないので、CRC 機能だけが利用できません。

参考資料

22.5 SPI と I²S レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#) を参照してください。

22.5.1 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 MODE	BIDI OE	CRC EN	CRC NEXT	DFF	RX ONLY	SSM	SSI	LSB FIRST	SPE	BR [2:0]	MSTR	CPOL	CPHA		

Bit 15 **BIDIMODE: 両方向データモード**

0: 2 信号線、一方向データモードが選択されます。

1: 1 信号線、両方向データモードが選択されます。

注 : I²S モードでは使用しません。

Bit 14 **BIDIOE: 両方向データモード、出力イネーブル**

BIDIMODE ビットで両方向モードが選択されているとき、このビットでデータ転送の方向を指定します。

0: 出力ディセーブル (受信専用モード)

1: 出力イネーブル (送信専用モード)

注 : マスター・モードでは MOSI ピンが、スレーブ・モードでは MISO ピンが使用されます。

I²S モードでは使用しません。

Bit 13 **CRCEN: ハードウェア CRC 機能**

0: CRC 機能は無効です。

1: CRC 機能は有効です。

注 : このビットへの書込みは SPI がディセーブル状態 (SPE = 0) の時に行わねばなりません。

I²S モードでは使用しません。

Bit 12 **CRCNEXT: CRC 送信指示**

0: 次の送信データは送信バッファから送信されます。

1: 次の送信データは送信 CRC レジスタから送出されます。

注 : このビットは、最後のデータが SPI_DR レジスタに書込まれた直後に、設定されねばなりません。

I²S モードでは使用しません。

Bit 11 **DFF: データ・フレームのフォーマット**

0: 送受信で 8bit データ・フレームが選択されます。

1: 送受信で 16bit データ・フレームが選択されます。

注 : このビットへの書込みは SPI がディセーブル状態 (SPE = 0) の時に行わねばなりません。

I²S モードでは使用しません。

Bit 10 **RXONLY: 受信専用**

このビットと BIDI モードビットとの組み合わせで、単方向 2 信号線を使用する際の送信方向を指定します。また、このビットは複数のスレーブがある環境でこのスレーブが選択されていないときに、出力が本来の通信を妨げないためにも使用されます。

0: 全二重モード (送受信)

1: 出力ディセーブル (受信専用モード)

注 : I²S モードでは使用しません。

参考資料

RM0008

SPI (Serial peripheral interface)

Bit 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には、SSI ビットの値が NSS ピン入力の代わりに使用されます。

- 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が有効です。

注 : I²S モードでは使用しません。

Bit 8 SSI: ソフトウェアスレーブ選択

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となります。NSS ピンの入力信号は無視され、このビットの値が代わりに NSS として使用されます。

注 : I²S モードでは使用しません。

Bit 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 : 通信が行われているときに、このビットを変更することはできません。

I²S モードでは使用しません。

Bit 6 SPE: SPI 回路イネーブル

- 0: SPI 回路ディセーブル
- 1: SPI 回路イネーブル

注 : I²S モードでは使用しません。

Bits 5:3 BR[2:0]: ポーラート選択

- 000: f_{CPU}/2
- 001: f_{CPU}/4
- 010: f_{CPU}/8
- 011: f_{CPU}/16
- 100: f_{CPU}/32
- 101: f_{CPU}/64
- 110: f_{CPU}/128
- 111: f_{CPU}/256

注 : 通信が行われているときに、このビットを変更することはできません。

I²S モードでは使用しません。

Bit 2 MSTR: マスター・モード

- 0: スレーブ・モード選択
- 1: マスター・モード選択

注 : 通信が行われているときに、このビットを変更することはできません。

I²S モードでは使用しません。

Bit1 CPOL: クロック極性

- 0: アイドル時には SCK は 0 となります。
- 1: アイドル時には SCK は 1 となります。

注 : 通信が行われているときに、このビットを変更することはできません。

I²S モードでは使用しません。

Bit 0 CPHA: クロック位相

- 0: クロックの最初の変化エッジで、データが読み取られます。
- 1: クロックの二番目の変化エッジで、データが読み取られます。

注 : 通信が行われているときに、このビットを変更することはできません。

I²S モードでは使用しません。



参考資料

22.5.2 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		RXNE IE		ERRIE		予約済み		SSOE		TXDMA EN		RXDMA EN	
								RW	RW	RW			RW	RW	RW

Bits 15:8 予約済み、ハードウェアで 0 に固定されています。

Bit 7 **TXEIE**: 送信(TX)バッファ空き(empty)割込み許可

0: TXE 割込みはマスクされます。

1: TXE 割込み許可。TXE フラグがセットされると、割込みリクエストが発せられます。

注: 正しく機能させるためには、TXEIE ビットと TXDMAEN ビットを同時にセットしてはいけません。

Bit 6 **RXNEIE**: 受信(RX)バッファデータ(not empty)割込み許可

0: RXNE 割込みはマスクされます。

1: RXNE 割込み許可。RXNE フラグがセットされると、割込みリクエストが発せられます。

注: 正しく機能させるためには、RXNEIE ビットと RXDMAEN ビットを同時にセットしてはいけません。

Bit 5 **ERRIE**: エラー割込み許可

このビットはエラー (SPI モード時の CRCERR, OVR, MODF 及び I2S モード時の UDR, OVR) が発生したときに、割込みリクエストを発生させるか否かを制御します。

0: エラー割込みはマスクされます。

1: エラー割込みのリクエストが発生します。

Bits 4:3 予約済み、ハードウェアで 0 に固定されています。

Bit 2 **SSOE**: SS 出力イネーブル

0: マルチマスター構成で動作させるため、マスター・モードで SS 出力 (NSS ピン) はディセーブル状態となります。

1: マスター・モードでセルがイネーブルとされたとき、SS 出力 (NSS ピン) はイネーブル状態となります。マルチマスターの環境で使用することはできません。

注: I²S モードでは使用しません。

Bit 1 **TXDMAEN**: 送信(Tx)バッファ DMA イネーブル

このビットがセットされていると、TXE フラグがセットされたときに DMA リクエストが発生します。

0: 送信バッファ DMA ディセーブル

1: 送信バッファ DMA イネーブル

Bit 0 **RXDMAEN**: 受信(Rx)バッファ DMA イネーブル

このビットがセットされていると、RXNE フラグがセットされたときに DMA リクエストが発生します。

0: 受信バッファ DMA ディセーブル

1: 受信バッファ DMA イネーブル

22.5.3 SPI ステータス・レジスタ (SPI_SR)

アドレスオフセット : 08h

参考資料

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		BSY	OVR	MODF	CRC ERR	UDR	CHSID E	TXE	RXNE

Bits 15:8 予約済み、ハードウェアで 0 に固定されています。

Bit 7 BSY: ビジー・フラグ

- 0: SPI はビジー状態ではありません。
 - 1: SPI は通信ビジー状態、もしく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でセット / クリアされます。

注: マスタ受信モードでは (1 ライン双方向) BSY フラグをチェックしてはいけません。

Bit 6 OVR: オーバラン・フラグ

- 0: オーバランは発生していません。
 - 1: オーバランが発生しました。
- このフラグはハードウェアでセットされ、ソフトウェアでクリアします。フラグ・クリアの操作手順は [セクション 22.4.7 \(593 ページ\)](#) を参照してください。

Bit 5 MODF: モードフォールト (Mode fault)

- 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でセットされ、ソフトウェアでクリアします。フラグ・クリアの操作手順は [セクション 22.3.8 \(578 ページ\)](#) を参照してください。

注: I²S モードでは使われません。

Bit 4 CRCERR: CRC エラー・フラグ

- 0: 受信された CRC の値は SPI_RXCRCR と一致しました。
 - 1: 受信された CRC の値は SPI_RXCRCR と異なりました。(CRC エラー)
- このフラグはハードウェアでセットされ、ソフトウェアで 0 を書込むことでクリアされます。

注: I²S モードでは使用しません。

Bit 3 UDR: アンダーランフラグ

- 0: アンダーランは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でセットされ、ソフトウェア手順でリセットされます。ソフトウェア手順については [セクション 22.4.7 \(593 ページ\)](#) を参照してください。

注: SPI モードでは使用しません。

Bit 2 CHSIDE : チャネルサイド

- 0: 左チャネルは送信または受信されなければなりません。
- 1: 右チャネルは送信または受信されなければなりません。

注: SPI モードでは使用しません。

PCM モードでは意味を持ちません。

Bit 1 TXE: 送信 (Tx) バッファ・空き

- 0: 送信バッファは空ではありません。
- 1: 送信バッファが空です。

Bit 0 RXNE: 受信 (Rx) バッファ・データ

- 0: 受信バッファは空です。
- 1: 受信バッファにはデータがあります (空ではありません)。

参考資料

22.5.4 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]															

Bits 15:0 DR[15:0]: データ・レジスタ

受信されたデータ、及び送信されるデータ。

データ・レジスタは 2 個のバッファから成っています。レジスタ書込み時には一方（送信バッファ）が、レジスタ読出し時には他方（受信バッファ）が対象となります。データ・レジスタへの書き込みでは送信 (Tx) バッファにデータが書込まれ、データ・レジスタからの読出しへは受信 (Rx) バッファの内容が読み出されます。

SPI モードについての注意 :

フレーム・フォーマットビット（SPI_CR1 レジスタの DFF ビット）の指定に応じて、送受信されるデータは 8 もしくは 16bit となります。この指定は SPI 回路をイネーブルとする前に設定しておかねばなりません。

8 ビットデータ・フレームの時には、バッファは 8bit で、レジスタの下位バイト (SPI_DR[7:0]) のみが有効となります。データの読出しへは上位バイト (SPI_DR[15:8]) は 0 となります。

16bit データ・フレームでは、データ・レジスタ全体 (SPI_DR[15:0]) が送受信されます。

22.5.5 SPI CRC 多項式レジスタ (SPI_CRCPR) (I²S モードでは使用しません。)

アドレスオフセット : 0x10

リセット値 : 0x0007

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

Bits 15:0 CRCPOLY[15:0]: CRC 多項式レジスタ

このレジスタは CRC 算出時の多項式を指定します。

CRC 多項式 0007h が、このレジスタのリセット時の値です。他の値も必要に応じて設定することができます。

注 : I²S モードでは使用しません。

参考資料

22.5.6 SPI Rx CRC レジスタ (SPI_RXCRCR) (I²S モードでは使用しません。)

アドレスオフセット : 0x14

リセット値 : 0x0000

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

Bits 15:0 RxCRC[15:0]: 受信 CRC レジスタ

CRC 計算がイネーブルの状態のとき、RxCRC[15:0] は、受信されたデータから算出された CRC の値を保持しています。このレジスタは SPI_CR1 レジスタの CRCEN ビットに 1 が書込まれたときにリセットされます。CRC の値は SPI_CRCPR レジスタの値にしたがって、順に計算されます。

データ・フォーマットとして 8bit が選択されている (SPI_CR1 レジスタの DFF ビットが 0) ときには、下位 8bit のみが有効で、CRC8 方式に基づく計算が行われます。

16 ビットデータ・フレームが選択されている (SPI_CR1 レジスタの DFF ビットが 1) ときには、16bit レジスタ全体が有効で、CRC16-CCITT 標準に準じた計算が行われます。

注: ビジー・フラグ (BSY) がセットされているときにこのレジスタを読み出すと、本来と異なる値が読み出されることがあります。

I²S モードでは使用しません。

22.5.7 SPI Tx CRC レジスタ (SPI_TXCRCR) (I²S モードでは使用しません。)

アドレスオフセット : 0x18

リセット値 : 0x0000

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

Bits 15:0 TxCRC[15:0]: 送信 CRC レジスタ

CRC 計算がイネーブルの状態のとき、TxCRC[15:0] は、送信されたデータから算出された CRC の値を保持しています。このレジスタは SPI_CR1 レジスタの CRCEN ビットに 1 が書込まれたときにリセットされます。CRC の値は SPI_CRCPR レジスタの値にしたがって、順に計算されます。

データ・フォーマットとして 8bit が選択されている (SPI_CR1 レジスタの DFF ビットが 0) ときには、下位 8bit のみが有効で、CRC8 方式に基づく計算が行われます。

16bit データ・フレームが選択されている (SPI_CR1 レジスタの DFF ビットが 1) ときには、16bit レジスタ全体が有効で、CRC16-CCITT 標準に準じた計算が行われます。

注: ビジー・フラグ (BSY) がセットされているときにこのレジスタを読み出すと、本来と異なる値が読み出されることがあります。

I²S モードでは使用しません。

参考資料

22.5.8 SPI_I²S 構成レジスタ (SPI_I2SCFGR)

アドレスオフセット : 1Ch

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 D	I2SE	I2SCFG	PCMS YNC	予約 済み	I2SSTD	CKPO L	DATLEN	CHLE N						

rw rw rw rw rw rw rw rw

Bits 15:12 予約済み、ハードウェアで 0 に固定されています。

Bit 11 I2SMOD: I²S モード選択

- 0: SPI モードがセレクトされています。
- 1: I²S モードがセレクトされています。

注: このビットは SPI または I²S がディセーブルのとき設定します。

Bit 10 I2SE: I²S イネーブル

- 0: I²S ペリフェラルがディセーブルです。
- 1: I²S ペリフェラルがイネーブルです。

注: SPI モードでは使用しません。

Bit 9:8 I2SCFG: I²S 設定モード

- 00: スレーブ - 送信
- 01: スレーブ - 受信
- 10: マスター - 送信
- 11: マスター - 受信

注: このビットは I²S がディセーブルなとき設定します。

SPI モードでは使用しません。

Bit 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) の時に限り意味を持ちます。

SPI モードでは使用しません。

Bit 6 予約済み、ハードウェアで 0 に固定されています。

Bit 5:4 I2SSTD: I²S 標準セレクト

- 00: I²S フィリップス標準
- 01: MSB 寄せ標準 (左寄せ)
- 10: LSB 寄せ標準 (右寄せ)
- 11: PCM 標準

I²S 標準の詳細は [セクション 22.4.2 \(581 ページ\)](#) を参照してください。

注: 正しい動作には、このビットは I²S がディセーブルのとき設定します。

SPI モードでは使用しません。

Bit 3 CKPOL: 安定状態クロック極性

- 0: I²S クロック安定状態はローレベル
- 1: I²S クロック安定状態はハイレベル

注: 正しい動作には、このビットは I²S がディセーブルのとき設定します。

SPI モードでは使用しません。

Bit 2:1 **DATLEN:** 転送されるデータ長

- 00: 16bit データ長
- 01: 24bit データ長
- 10: 32bit データ長
- 11: 設定してはいけません。

注: 正しい動作には、このビットは I²S がディセーブルのとき設定します。
SPI モードでは使用しません。

Bit 0 **CHLEN:** チャネル長（オーディオチャネル毎のビット数）

- 0: 16bit 幅
- 1: 32bit 幅

このビットは DATLEN=00 のときに限り意味を持ちます。そうでなければ、書込まれた値に関係なくチャネル長はハードウェアにより 32bit 固定です。

注: 正しい動作には、このビットは I²S がディセーブルのとき設定します。
SPI モードでは使用しません。

22.5.9 SPI_I²S プリスケーラ・レジスタ (SPI_I2SPR)

アドレスオフセット : 20h

リセット値 : 0000 0010 (0002h)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				MCKOE	E	ODD	I2SDIV								
				rw		rw					rw				

Bits 15:10 予約済み、ハードウェアで 0 に固定されています。

Bit 9 **MCKOE:** マスタクロック出力イネーブル

- 0: マスタクロック出力はディセーブルです。
- 1: マスタクロック出力はイネーブルです。

注: このビットは I²S がディセーブルのとき設定します。I²S がマスター・モードの時に限り使われます。

SPI モードでは使用しません。

Bit 8 **ODD:** プリスケーラ用奇数ファクタ

- 0: 本当のディバイダ値は = I2SDIV *2
- 1: 本当のディバイダ値は = (I2SDIV * 2)+1

[セクション 22.4.3 \(588 ページ\)](#) を参照してください。

注: このビットは I²S がディセーブルのとき設定します。I²S がマスター・モードの時に限り使われます。

SPI モードでは使用しません。

Bit 7:0 **I2SDIV:** I²S リニアプリスケーラ

I2SDIV [7:0] = 0 または I2SDIV [7:0] = 1 は使ってはいけない値です。

[セクション 22.4.3 \(588 ページ\)](#) を参照してください。

注: このビットは I²S がディセーブルのとき設定します。I²S がマスター・モードの時に限り使われます。

SPI モードでは使用しません。

参考資料

22.5.10 SPI レジスタ・マップ

表は SPI のレジスタ・マップとリセット値を示しています。

表 147. SPI レジスタ・マップとリセット時の値

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 リセット値															BIDIMODE 0	BIDIOE 0	CRCEN 0	CRCNEXT 0	DFF 0	RXONLY 0	SSM 0	SSI 0	LSBFIRST 0	BR [2:0] 0	0	0	0	0	0	0	0	0		
0x04	SPI_CR2 リセット値																																		
0x08	SPI_SR リセット値																																		
0x0C	SPI_DR リセット値																																		
0x10	SPI_CRCPR リセット値																																		
0x14	SPI_RXCRCR リセット値																																		
0x18	SPI_TXCRCR リセット値																																		
0x1C	SPI_I2SCFGR リセット値															I2SMOD 0	I2SE 0	I2SCFG 0	PCMSYNC 0	预約済み 0	BSY 0	TXEIE 0	RXNEIE 0	OVR 0	MODF 0	ERRIE 0	预約済み 0	预約済み 0	预約済み 0	预約済み 0	预約済み 0	预約済み 0	预約済み 0	预約済み 0	
0x20	SPI_I2SPR リセット値															MCKOE 0	ODD 0	预約済み 0	预約済み 0	预約済み 0	I2SSTD 0	CKPOL 0	DATLEN 0	CHLEN 0	I2SDIV 1	TXE 0	TXDMAEN 0	CPOL 1	TXDMAEN 0	CPHA 0	CPOL 1	TXDMAEN 0	CPHA 0	预約済み 0	预約済み 0

注意： レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。

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

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

23.1 概要

I²C(Inter-Integrated Circuit) バスインターフェースはマイクロコントローラと I²C バスとの間のインターフェースとして動作します。インターフェースは複数マスタ機能をサポートしており、I²C バスで定められたすべての動作手順、アービトレーションとタイミング機能を備えています。インターフェースは標準モードと高速モード、さらに SMBus 2.0 互換をサポートしています。

インターフェースは CRC 生成と検査機能を持っており、SMBus (System Management Bus) や PMBus (Power Management Bus) など、さまざまな目的で利用することができます。

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 機能が利用できます。

23.2 I²C の主な機能

- 並列バス /I²C プロトコル間のコンバータ
- 複数マスタ構成をサポート、同じインターフェースでマスタとスレーブの両機能
- I²C マスタ機能：
 - クロック生成
 - START 信号、STOP 信号の送出
- I²C スレーブ機能：
 - 設定可能な I²C アドレス検出
 - 2 個のスレーブアドレスを受付けるデュアルアドレス機能
 - STOP 信号検出
- 7bit/10bit アドレスと同報 (General Call) の送出と検出
- 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)
- ステータス・フラグ：
 - 送信 / 受信モード・フラグ
 - End-of-byte 送信フラグ
 - I²C ビジー・フラグ
- エラー・フラグ：
 - マスタ・モードにおけるバス占有権喪失
 - アドレス / データ送信における確認応答 (acknowledgement) エラー
 - START 信号、STOP 信号のタイミングエラー検出
 - クロック伸張機能が無効のときのオーバラン / アンダラン
- 2 個の割込みベクタ：
 - 正常なアドレス / データ通信割込み
 - エラー条件割込み
- オプションとしてのクロック伸張
- DMA 利用可能な 1 バイトバッファ
- 設定可能な PEC (Packet Error Checking) 生成と検査：
 - 送信モードで、PEC 値を最終バイトとして送信可能
 - 最終バイトで PEC エラーチェック
- SMBus 2.0 互換：
 - 25 ms クロックロー・タイムアウト
 - 10 ms マスタクロックロー累積・拡張タイムアウト
 - 25 ms スレーブクロックロー累積・拡張タイムアウト
 - ハードウェアによる PEC 生成 / 検査と ACK 制御
 - ARP (Address Resolution Protocol) サポート
- PMBus 互換

注意： 上記の機能の幾つかは、製品によってはサポートされていないことがあります。特定のチップの I²C インタフェースの実装で、特定の機能がサポートされているか否かについては製品のデータシートを参照してください。

23.3 I²C の機能詳細

データの送受信に加え、この I²C インタフェースモジュールはデータのシリアルからパラレルへの変換などを行います。割込みリクエストはソフトウェアで許可・禁止を選ぶことができます。I²C インタフェースは I²C バスとはデータピン (SDA) とクロックピン (SCL) で接続されます。I²C インタフェースは、ソフトウェアによる選択で、標準 I²C (最大 100 kHz) と高速 I²C (最大 400 kHz) のいずれにも接続できます。

23.3.1 モード選択

インターフェースは次の 4 モードのいずれかで動作できます。

- スレーブ・トランシッタ
- スレーブ・レシーバ
- マスタ・トランシッタ
- マスタ・レシーバ

デフォルトではインターフェースはスレーブ・モードで動作します。インターフェースは複数マスタ構成を実現するために、START 信号を送出後にスレーブからマスタに、バス占有権を喪失もしくは STOP 信号を送出後にマスタからスレーブに自動的に切り替わります。

交信フロー

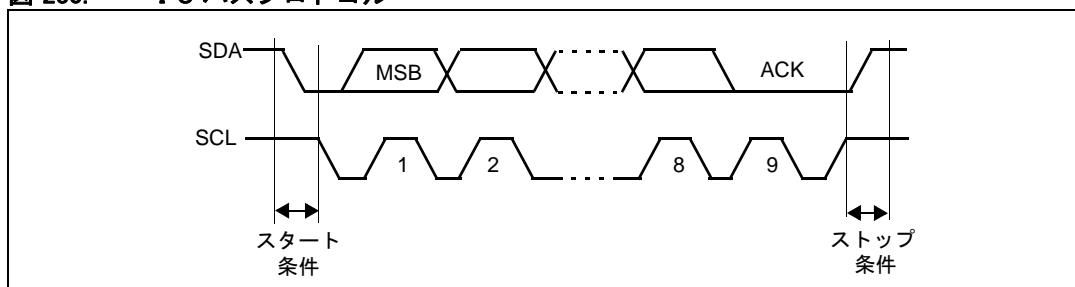
マスタ・モードでは、I²C インタフェースはクロックを生成し、データの転送手順を開始します。個々の手順は START 信号 (条件) で始まり、STOP 信号 (条件) で終わります。START 信号と STOP 信号はマスタ・モードでソフトウェアによって送出されます。

スレーブ・モードでは、I²C インタフェースは自己のアドレス (7 もしくは 10bit) と同報アドレス (general call address) を認識することができます。同報アドレスの認識はソフトウェアでオン・オフの切替えができます。

アドレスとデータは 8bit bytes で、上位ビットから順に転送されます。START 信号後の先頭のバイト (7bit アドレスでは 1byte、10bit アドレスでは 2bytes) にはアドレスが含まれています。アドレスはマスタ・モードの I²C インタフェースから送信されます。

バイト転送の 8 番目のクロックに続く 9 番目のクロックでは、レシーバが確認応答 (acknowledge bit) をトランシッタに送り返さねばなりません

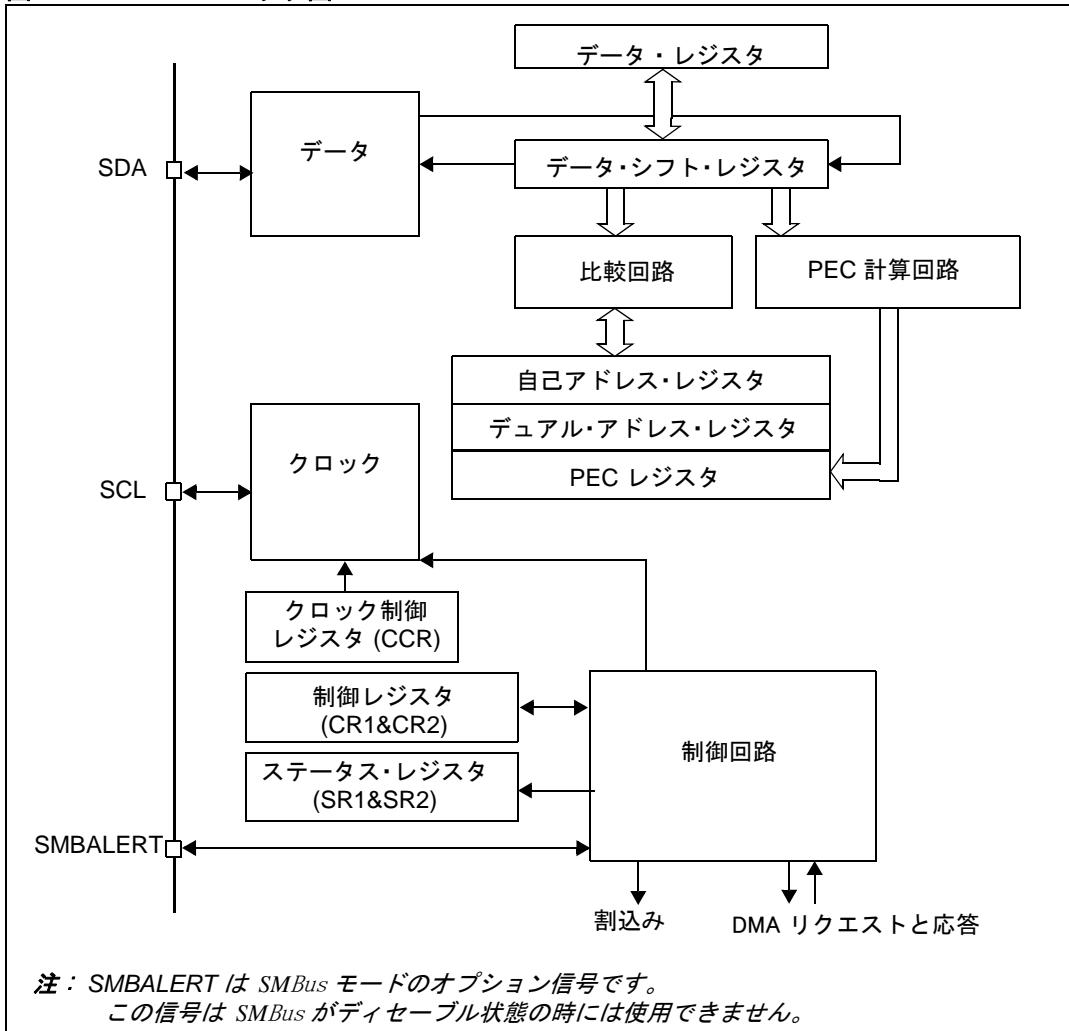
図 230. I²C バスプロトコル



確認応答の送出は、ソフトウェアでオン・オフができます。I²C インタフェース・アドレスの指定 (7bit/10bit アドレス) と同報アドレス受け入れのオン・オフが可能です。

I²C インタフェースのブロック図は [図 231](#) にあります。

参考資料

図 231. I²C ブロック図

23.3.2 I²C スレーブ・モード

正しいタイミングを生成するために周辺回路の入力クロックは I²C_CR2 レジスタで設定されねばなりません。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

START 信号が検出されると、アドレスが SDA ラインから受信され、シフト・レジスタに送られます。受信されたアドレスは自己のアドレス (OAR1) と OAR2 (ENDUAL=1 の時)、及び (ソフトウェアで機能が有効とされていれば) 同報アドレスと比較されます。

注意： 10bit アドレスモードでは、比較対象にはヘッダビット列 (11110xx0) とアドレスの上位 2 bit が含まれます。(xx 部)

ヘッダもしくはアドレスの不一致： インタフェースはこれを無視し、次の START 信号を待ちます。

ヘッダー一致 (10bit モードのみ) : ACK ビットがセットされていれば、確認応答を送り返し、スレーブアドレスの 8bit を待ちます。

参考資料

アドレス一致：インター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れば確認応答を返します。
- ADDR ビットがハードウェアによりセットされ、ITEVFEN ビットがセットされていれば割込みリクエストが発生します。
- ENDUAL=1 の場合、ソフトウェアで DUALF ビットを読み、どちらのスレーブアドレスが使われたかを確認しなければなりません。

10bit モードでは、アドレスビット列を受信後はスレーブは常に受信モードとなります。スレーブインターフェースは、START 信号の繰り返しが送られ、自己のアドレスに一致したアドレスビットとアドレスの最下位 2bit を含むヘッダビット列が受信された後、送信モードに入ります。

TRA ビットはスレーブが受信モードにあるか、送信モードであるかを示します。

スレーブトランスマッタ

アドレス受信に続いて ADDR がクリアされると、スレーブは DR レジスタのバイトデータを、内部のシフト・レジスタを経由して SDA ラインに送出します。

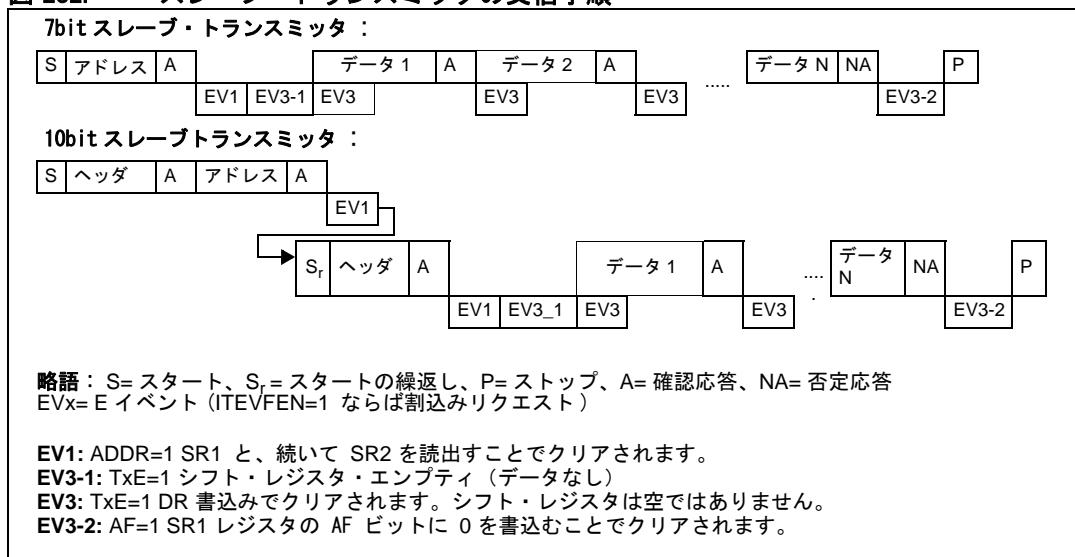
スレーブは SCL のローレベルを、ADDR がクリアされ DR に送信データが書込まれるまで延長します。（図 232 交信手順 EV1 EV3 を参照）

確認応答が受信されると、

- TxE ビットがハードウェアによりセットされます。ITEVFEN ビットと ITBUFEN ビットがセットされていれば、このとき割込みリクエストが発生します。

TxE ビットがセットされ、次のデータ送信までにデータが DR レジスタに書込まれていなければ BTF ビットがセットされ、インターフェースは SCL のローレベルを延長し DR レジスタにデータが書込まれるのを待ちます。

図 232. スレーブ・トランスマッタの交信手順



参考資料

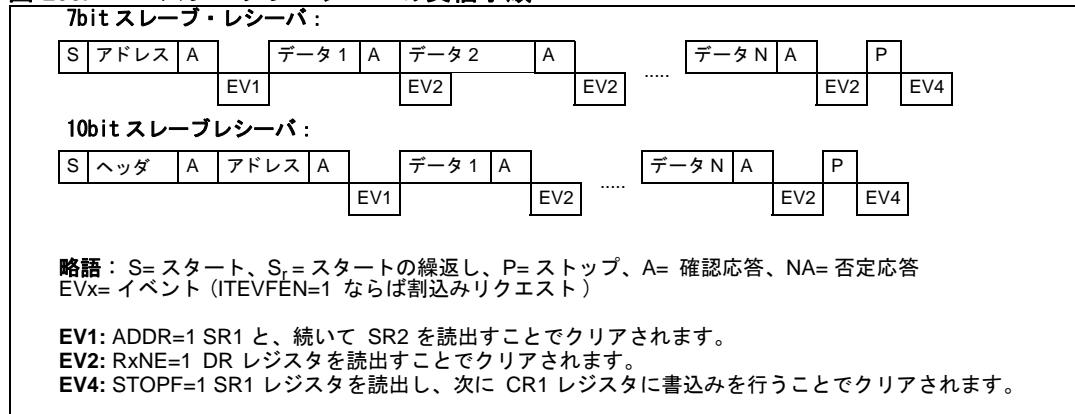
スレーブレシーバ

アドレス受信に続いて ADDR がクリアされると、スレーブは SDA ラインからシフト・レジスタを経由して、DR レジスタにデータバイトを読み込みます。1byte が受信されるごとに、

- ACK ビットがセットされていれば、確認応答を送出します。
- RxNE ビットがハードウェアによってセットされます。と ITEVFEN ビットと ITBUFEN ビットがセットされていれば、このとき割込みリクエストが発せられます。

RxNE ビットがセットされ、次のデータ受信の前に DR レジスタのデータが読出されていないときには BTF ビットがセットされ、インターフェースは SCL のローレベルを延長して DR レジスタが読み出されるのを待ちます。（図 233 交信手順を参照）

図 233. スレーブレ・シーバの交信手順



スレーブ交信の終了

最後のデータバイトが転送されると、マスタは STOP 信号を送出します。インターフェースはこれを認識し、

- STOPF ビットをセットします。このとき ITEVFEN ビットがセットされていれば割込みがリクエストされます。

次に、インターフェースは SR1 レジスタが読み出され、さらに CR1 レジスタが書き込まれるのを待ちます。（図 233 交信手順 EV4 を参照）

23.3.3 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手順を開始し、クロック信号を送出します。シリアルデータ転送は常に START 信号（条件）で開始され、STOP 信号（条件）で終わります。START ビットのよりバス上で START 信号が送出されると同時に、マスタ・モードが選択されます。

マスタ・モードでは次の手順がリクエストされます。

- 正しいタイミングを生成するため、I2C_CR2 レジスタでペリフェラル入力クロックの設定を行います。
- クロック制御レジスタの設定を行います。
- 立上がり時間レジスタを設定します。
- I2C_CR1 レジスタの書き込みでペリフェラルをイネーブル状態とします。
- I2C_CR2 レジスタの START ビットをセットし、START 信号を送出します。

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

START 信号 (Start condition)

BUSY ビットがクリアされているときに START ビットをセットすると、インターフェースはマスタ・モードに切替わり (M/SL=1)、START 信号が送出されます。

注意 : マスタ・モードでは、START ビットをセットするとインターフェースは、現在のバイト転送の最後で再スタートを生成します。

START 信号が送出されると、

- SB ビットがハードウェアによりセットされます。このとき、ITEVFEN ビットがセットされていれば割込みがリクエストされます。

次に、マスタは SR1 レジスタが読出され、DR レジスタにスレーブアドレスが書込まれるのを待ちます。（図 234 と図 235 交信手順 EV5 を参照）

参考資料

スレーブアドレス送信

スレーブアドレスが内部のシフト・レジスタを経由して SDA ラインから送出されます。

- 10bit アドレスモードでは、ヘッダビット列が送出されると、
 - ADD10 ビットがハードウェアによりセットされます。このとき ITEVFEN ビットがセットされていれば割込みがリクエストされます。
- マスターは SR1 レジスタが読出され、続いて DR レジスタに 2 番目のアドレスバイトが書込まれるのを待ちます。（図 234 と図 235 交信手順を参照）
 - ADDR ビットがハードウェアによりセットされ、ITEVFEN ビットがセットされていれば割込みがリクエストされます。
- マスターは SR1 レジスタが読出され、続いて SR2 レジスタが読出されるのを待ちます。（図 234 と図 235 交信手順を参照）
- 7bit アドレスモードでは 1 個のアドレスバイトが送出されます。
アドレスバイトが送出されると、
 - ADDR ビットがハードウェアによりセットされ、ITEVFEN ビットがセットされていれば割込みがリクエストされます。
- マスターは SR1 レジスタが読出され、続いて SR2 レジスタが読出されるのを待ちます。（図 234 と図 235 交信手順を参照）

送信したスレーブアドレスの最下位ビットにより、マスターはトランスマッタ・モードに入るか、レシーバ・モードに入るかが決められます。

- 7bit アドレスモード
 - トランスマッタ・モードに入る場合、マスターは最下位ビットがクリアされたスレーブアドレスを送信します。
 - レシーバ・モードに入る場合、マスターは最下位ビットがセットされたスレーブアドレスを送信します。
- 10bit アドレスモード
 - トランスマッタ・モードに入る場合、マスターはまずヘッダ (11110xx0) を送り、次に最下位ビットがクリアされた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・モードに入る場合、マスターはまずヘッダ (11110xx0) を送り、次に最下位ビットがクリアされたスレーブアドレスを送信します。さらに、マスターはヘッダ (11110xx1) を伴った START 信号を繰返し送出します。（xx はアドレスの上位 2 ビットです。）

TRA ビットはマスターが受信モードにあるか、送信モードかを示します。

マスタートランスマッタ

アドレス送信に続いて ADDR がクリアされると、マスターは DR レジスタのバイトを内部のシフト・レジスタを経由して SDA ラインに送出します。

マスターは次に TxE がクリアされるまで待ちます。（図 234 転送手順 EV8 を参照）

確認応答が受信されると

- TxE ビットがハードウェアによりセットされるのを待ちます。このとき ITEVFEN ビットと ITBUFEN ビットがセットされていれば、割込みがリクエストされます。

TxE がセットされ、次のバイトの送信までに DR レジスタへの書込みが行われていないときには BTF ビットがセットされ、インターフェースは BTF がクリアされるまで待ちます。

参考資料

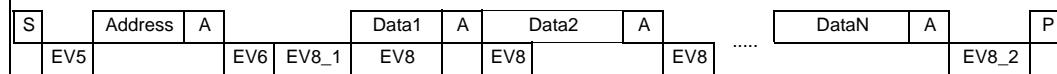
マスタトランスマッタ交信の終了

交信を終了するには：最後のバイトを DR レジスタに書込んだ後に、ソフトウェアで STOP ビットをセットし STOP 信号を送出させます。（図 234 交信手順）インターフェースは自動的にスレーブ・モードに戻ります。（M/SL ビットがクリアされます。）

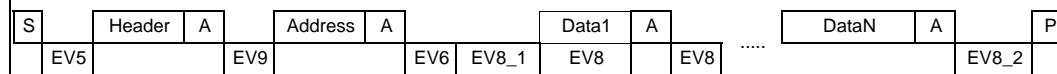
注意： TxE もしくは BTF ビットがセットされたとき、ソフトウェアで STOP 送出が指示されなければなりません。

図 234. マスタトランスマッタの交信手順

7bit マスタトランスマッタ



10bit マスタトランスマッタ



略語：S=スタート、S_r=スタートの繰返し、P=ストップ、A=確認応答、NA=否定応答
EVx= イベント (ITEVFEN=1 ならば割込みリクエスト)

EV5: SB=1 SR1 レジスタを読み出し、次に DR レジスタにアドレスを書込むことでクリアされます。

EV6: ADDR=1 SR1 を読み出し、次に SR2 レジスタを読み出すことでクリアされます。

EV8_1: TxE=1 シフト・レジスタ・エンプティ（データなし）

EV8: TxE=1 DR レジスタに書込むことでクリアされます。

EV8_2: TxE=1、BTF = 1 STOP 信号でハードウェアによりクリアされます。

EV9: ADD10=1 SR1 レジスタを読み出し、次に DR レジスタに書き込みを行うことでクリアされます。

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ーフェースは SDA ラインから内部のシフト・レジスタを経由して DR レジスタにデータバイトを読み込みます。バイトが読み込まれるごとに、インターフェースは

- ACK ビットがセットされていれば、確認応答を送出します。
- RxNE ビットがセットされます。このとき ITEVFEN ビットと ITBUFEN ビットがセットされていけば割込みがリクエストされます。（図 235 交信手順 EV7）

次のバイトを受信する前に、RxNE ビットがセットされ DR レジスタのデータが読み出されていないときには BTF ビットがハードウェアによってセットされ、インターフェースは DR レジスタが読み出されるまで待ちます。

参考資料

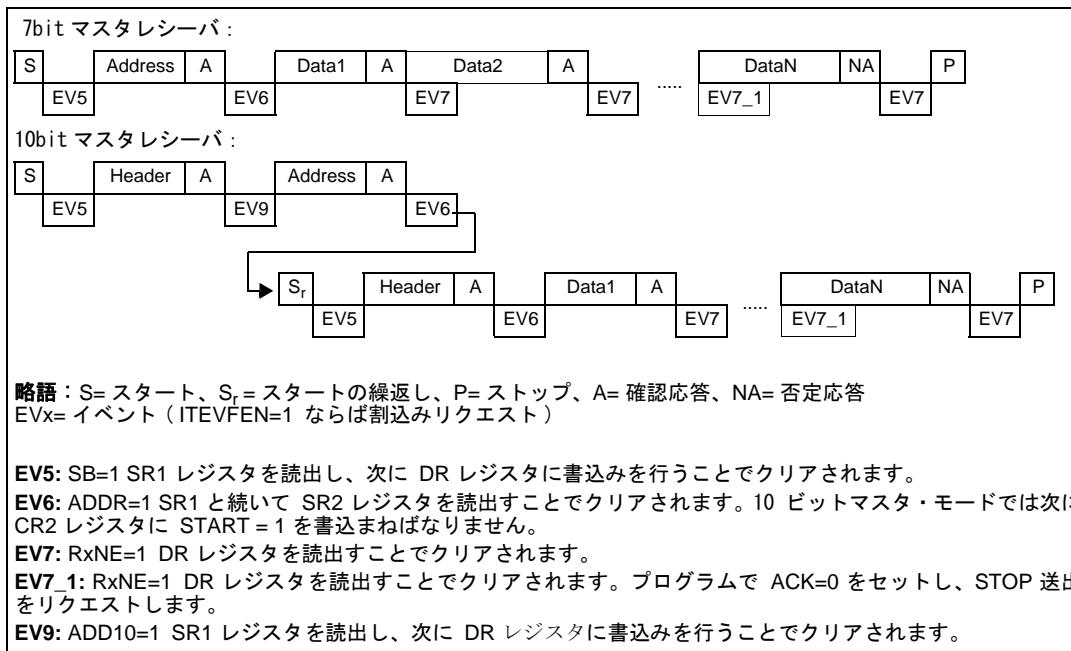
マスタレシーバ交信の終了

マスタはスレーブからの最終バイトの受信に対して NACK を返します。スレーブは NACK を受取ると SCL と SDA ラインの制御を解除します。マスタは次に STOP 信号もしくは RESTART 信号を送出することができます。

- 最後に受信されたバイトの後で否定応答 (NACK) パルスを発生するには、その前のデータバイトの後 (RxNE イベントの後) で ACK ビットがクリアされなければなりません。
- STOP 信号もしくは RESTART 信号を発生するには、2 番目のデータバイトの後 (2 番目の RxNE イベントの後) で STOP/START ビットをソフトウェアでセットしなければなりません。
- シングルバイトを受信する場合には、確認応答ディセーブルと STOP 信号の生成は EV6 で行われます。

STOP 信号の送出後にインターフェースは自動的にスレーブ・モードに戻ります。(M/SL ビットがクリアされます。)

図 235. マスタレシーバの交信手順



23.3.4 エラー条件

以下は通信に誤りをもたらす可能性のあるエラー条件です。

バスエラー (BERR)

I²C インタフェースがバイト転送中に STOP もしくは START 信号（ビット条件）を検出したときに、このエラーが発生します。このとき、

- BERR ビットがセットされます。ITERREN ビットがセットされていれば割込みがリクエストされます。
- スレーブでは、データは破棄され、ラインはハードウェアによって解放されます。
 - START が予期しないタイミングで現れた場合、スレーブはこれを RESTART 信号とみなし、アドレスもしくは STOP 信号が送られてくるのを待ちます。
 - STOP が予期しないタイミングで現れた場合、スレーブは STOP 信号が送られたものとみなし、ハードウェアによりバスラインを解放します。

確認応答異常 (AF)

インターフェースが否定応答ビットを検出したとき、このエラーが発生します。このとき、

- AF ビットがセットされます。ITERREN ビットがセットされていれば割込みがリクエストされます。
- NACK を受信したトランスマッタは交信状態をリセットしなければなりません。
 - スレーブ：バスラインはハードウェアにより解放されます。
 - マスター：ソフトウェアで STOP 信号を送出しなければなりません。

バス占有権喪失 (ARLO)

I²C インタフェースがバス占有権の喪失を検出した時には、このエラーが発生します。このとき、

- ARLO ビットがハードウェアによりセットされます。ITERREN ビットがセットされていれば割込みがリクエストされます。
- 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。（M/SL ビットはクリアされます。）
- バスラインはハードウェアによって解放されます。

参考資料

オーバラン / アンダランエラー (OVR)

クロック伸張がディセーブル状態で I²C インタフェースがデータを受信しているときには、スレーブでオーバ・ラン・エラーが発生することがあります。次のバイトが受信された時点で、インターフェースがすでにバイトを受信しており (RxNE=1)、DR レジスタのデータがまだ読出されていないときには、

- 最後に受信されたバイトは失われます。
- オーバ・ラン・エラーが発生したときには、ソフトウェアで RxNE ビットをクリアし、トランスマッタは最後のバイトを再送信しなければなりません。

クロック伸張がディセーブル状態で I²C インタフェースがデータを送信しているときには、スレーブでアンダランエラーが発生することがあります。次のバイト送信のためのクロックが現れたときに、DR レジスタに次に送信すべきデータがまだ格納されていない (TxNE=1) 時には、

- DR レジスタに残っているバイトが再び送出されます。
- アンダランエラー時にレシーバ側で受信されたデータが捨てられ、I²C バス規格で定められたクロックがローレベルの時間内に次のバイトが書込まれることを確認しなければなりません。

23.3.5 SDA/SCL ライン制御

- クロックの伸張機能がイネーブルのとき、
 - トランスマッタ・モード、TxNE=1、BTF=1 のとき：インターフェースは送信動作の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 SR1 を読み出し、さらにデータ・レジスタへの書き込みを待ちます。
 - レシーバ・モード、RxNE=1、BTF=1 のとき：インターフェースは受信動作の後で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 SR1 を読み出し、さらにデータ・レジスタからの読み出しを待ちます。
- クロックの伸張機能がディセーブルの場合、スレーブでは、
 - オーバ・ラン・エラー：RxNE=1 で次のバイトが受信される前に 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。
 - アンダランエラー：TxNE=1 で次のバイトの送出が行われる前に DR に書き込みが行われていない場合、同じバイトが再び送出されます。
 - 書き込みの競合は管理されていません。

参考資料

23.3.6 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、もしくは他のすべてと通信できる 2 線インターフェースです。このバスは I²C の基本動作に基づいていますが、SMBus は複数のタスクに関連するシステムの制御やパワーマネジメントの機能を搭載しています。システムでは、個別の制御ラインを切り替えながら使用することなく、SMBus でデバイス間のメッセージ交換を行うことができます。

System Management Bus 仕様では、3 種類のデバイスが規定されています。スレーブデバイスはコマンドを受信し、これに応答するデバイスです。マスタデバイスはコマンドを発行し、クロックを送出し、転送を終了させることができるデバイスです。ホストはシステム CPU とのメインインターフェースとしての機能を有する特殊なマスタデバイスです。ホストはマスターとスレーブの両機能を持ち、SMBus ホスト通知プロトコルをサポートしていかなければなりません。ホストはシステム全体で 1 個に限られています。

SMBus と I²C の類似点

- 2 線バスプロトコル (1 Clk, 1 Data)、SMBus アラートラインの追加されています。
- マスター - スレーブ通信、マスターがクロックを供給します。
- 複数マスター構成
- SMBus データ・フォーマットは I²C 7bit アドレスフォーマットに類似しています。(図 230 を参照)

SMBus と I²C の相違点

次の表に SMBus と I²C の相違点が示されています。

表 148. SMBus・I²C 比較

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
35 ms クロックのローレベルのタイムアウト	タイムアウトなし
論理電圧値は固定	論理電圧値は VDD に依存
種々のアドレスタイプ (予約、ダイナミック、など)	7bit、10bit、同報 (general call) スレーブアドレス
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 の用途

SMBus を利用し、デバイスはシステムに対して、製造者情報、モデル / パーツ番号、サスペンド時の状態セーブ、多種のエラー情報、対応できる制御パラメータ、デバイスの状態などを提供することができます。また、SMBus はシステムの制御バスとして、また複数のタスクにかかるパワーマネジメント機能を提供します。

参考資料

デバイスの識別子

SMBus 上に接続されたデバイスは、スレーブとしてユニークなアドレスを有します。スレーブアドレスのうち予約されている値は SMBus 仕様 ver. 2.0 (<http://smbus.org/specs/> を参照してください)

バスプロトコル

SMBus 仕様には最大 9 種のプロトコルが規定されています。プロトコルの詳細とアドレスタイプに関しては、SMBus 仕様 ver. 2.0 (<http://smbus.org/specs/>) を参照してください。プロトコルはユーザのソフトウェアで実現しなければなりません。

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当てることで解決できます。アドレス解決プロトコル (ARP: Address Resolution Protocol) は次の特徴を持っています。

- アドレスの割当てに標準の SMBus 物理層アービトレーション機能を利用します。
- 割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。電源オフの間もアドレスを維持することも認められています。
- アドレス割当て後は、新たな SMBus パケットのオーバヘッドは生じません。(アドレスが割当てられたスレーブデバイスへのアクセスは、アドレスが固定しているデバイスへのアクセスと同じオーバヘッドとなります。)
- いずれの SMBus マスタもバスの番号付けを行うことができます。

ユニーク・デバイス識別子 (UDID: Unique device identifier)

アドレス割当てのために個々のデバイスを区別する手段を提供するために、各デバイスはユニーク・デバイス識別子 (UDID: Unique device identifier) を組込まねばなりません。

128bit UDID の詳細と、ARP に関する更なる情報については SMBus 仕様 ver. 2.0 (<http://smbus.org/specs/>) を参照してください。

SMBus アラートモード

SMBus アラートは、ピンを通してマスタと機能を共用するデバイスにおいて、割込み機能を伴うオプションの信号です。SMBALERT は SCL や SDA 信号と同じく分散論理積 (wired-AND) 接続された信号です。SMBus から呼び出される SMBus General Call Address. Messages とともに使用される SMBALERT は 2bytes の長さです。

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、通信の必要があることを伝えるため、SMBALERT を通してホストに信号を送ることができます。ホストは割込みを処理し、同時に全 SMBALERT デバイスにアラート応答アドレス (ARA と称され、値は 0001100X) によりアクセスします。SMBALERT をローレベルにプルダウンしているデバイスのみがアラーと応答アドレスを受けます。この状態は I2C_SR1 レジスタの SMBALERT ステータス・フラグにより識別されます。ホストは部分的に変更されたバイト受信動作を実行します。スレーブ送信デバイスの 7bit デバイス・アドレスはバイトの上位 7bit に置かれます。8 番目のビットは 0 もしくは 1 のいずれのこともあります。

複数のデバイスが SMBALERT をローにプルダウンしている場合、最も優先度の高い (アドレスが小さい) デバイスがスレーブアドレス転送時の通常のアービトレーションの方法で通信の権利を得ます。スレーブアドレスが確認されたとき、デバイスは SMBALERT のプルダウンを解消します。メッセージ転送が終了したとき、SMBALERT のローレベルがホストでまだ観察される時には、再び ARA からの読み込みを行います。

SMBALERT 信号の機能が組み込まれていないホストは、周期的に ARA にアクセスすることができます。

SMBUs アラートモードの詳細は、SMBus 仕様 ver. 2.0 (<http://smbus.org/specs/>) 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関して相違点があります。

SMBus ではクロックのローレベルのタイムアウト、「35 ms タイムアウト」が定義されています。さらに SMBus ではスレーブデバイスにとってのクロックのローレベルの時間を累積した TLOW:SEXT が定められています。SMBus ではマスタデバイスにとってのクロックのローレベルの時間を累積した TLOW:MEXT も定められています。タイムアウトの値の詳細は、SMBus 仕様 ver. 2.0 (<http://smbus.org/specs/>) を参照してください。

I²C_SR1 レジスタのタイムアウト・ステータス・フラグは、この機能のステータスを示しています。

SMBus モードの使い方

I²C モードから SMBus モードに切り替えるには、次の手順を実行します。

- I²C_CR1 レジスタの SMBus ビットをセットします。
- I²C_CR1 レジスタの SMBTYPE ビットと ENARP ビットを、アプリケーションの必要にあわせて設定します。

デバイスをマスタとして設定する場合、セクション 23.3.3: I²C マスタ・モードに記述されている START 信号送出の手順を、他の場合にはセクション 23.3.2: I²C スレーブ・モードにある手順を実行します。

アプリケーションでは、ソフトウェアにより様々な SM バスプロトコルを制御しなければなりません。

- SMB デバイスデフォルトアドレス、ENARP=1 と SMBTYPE=0 で指定。
- SMB ホストヘッダ、ENARP=1 と SMBTYPE=1 で指定。
- SMB アラート応答アドレス、SMBALERT=1 で指定。

23.3.7 DMA リクエスト

DMA 利用がイネーブルとされている時には、DMA はデータ転送のためにのみリクエストされます。DMA リクエストは送信時にデータ・レジスタが空になったとき、及び受信時にデータ・レジスタでデータが有効となったときに発生します。データ転送の数が所定の DMA チャネルに設定されていた値に達したとき、DMA コントローラは転送終了 (EOT) を I²C インタフェースに通知し、割込みリクエストが許可されていれば、転送終了割込みリクエストを発生させます。

- マスタトランスマッタ : EOT 割込みの処理ルーチンでは、DMA リクエストを禁止し、BTM イベントの発生を待って STOP 信号を送出します。
- マスタレシーバ : DMA コントローラは「バイト数 - 1」に相当するハードウェア信号 EOT_1 を送ります。I²C_CR2 レジスタで LAST ビットがセットされていると、I²C は自動的に EOT_1 の後 NACK を送出します。割込みリクエストが許可状態のとき、DMA 転送完了割込みの処理ルーチンで STOP 信号を送出することも可能です。

注意 :

DMA コントローラが搭載されているか否かは、個々の製品の仕様書を参照してください。DMA 機能が搭載されていない製品では、I²C はセクション 1.4 に記載されている方法で利用しなければなりません。連続的な送受信を実現するために、割込み処理ルーチンの中で TxE/RxNE フラグをクリアすることもできます。

DMA を利用した送信

DMA モードを送信のためにイネーブルとするには、I²C_CR2 レジスタの DMAEN ビットをセットします。DMAEN ビットはアドレスシーケンスを受信し ADDR がクリアされた後でセットされねばなりません。データは TxE ビットがセットされるごとに、DMA ペリフェラルで設定されている (DMA 仕様を参照) メモリ・ドメインから I²C_DR レジスタにロードされます。I²C 送信のための DMA チャネルのマッピングは次の手順で行われます。(ここで x はチャネル番号をあらわしています。)

1. I²C_DR レジスタのアドレスを DMA_CPARx レジスタにセットします。データは TxE イベントごとに、メモリからこのアドレスに転送されます。
2. メモリ・アドレスを DMA_CPARx レジスタにセットします。データは TxE イベントごとに、このメモリから I²C_DR に転送されます。
3. 転送されるバイトの総数を DMA_CNDTRx レジスタに設定します。TxE イベントごとにこの値はカウントダウンされます。
4. チャネルの優先順位を DMA_CCRx レジスタの PL[1:0] に設定します。
5. DMA_CCRx レジスタの DIR ビットをセットし、ソフトウェアの必要に応じて 1/2 転送割込み、転送完了割込みの設定を行います。
6. DMA_CCRx レジスタの EN ビットをセットすることで、DMA 動作を有効とします。

転送バイト数が DMA コントローラのレジスタに設定されていた値に達したとき、DMA コントローラは転送終了信号 EOT/EOT_1 を I²C インタフェースに送ります。このとき、割込みリクエストが許可されていれば、DMA チャネルの割込みベクタに対する割込みリクエストが発生します。

注意 :

送信動作に DMA が利用されるときには I²C_CR2 レジスタの ITEVTEN ビットを割込みリクエスト許可に設定してはいけません。

参考資料

DMA を利用した受信

DMA モードを受信のためにイネーブルとするには、I²C_CR2 レジスタの DMAEN ビットをセットします。DMAEN ビットはアドレスシーケンスを受信し ADDR がクリアされた後でセットされねばなりません。1 バイトのデータが受信されるごとに、I²C_DR レジスタから DMA ペリフェラルで設定されている (DMA 仕様を参照) メモリ・ドメインに転送されます。I²C 受信のための DMA チャネルのマッピングは次の手順で行われます。(ここで x はチャネル番号をあらわしています。)

1. I²C_DR レジスタのアドレスを DMA_CPARx レジスタにセットします。データは RxNE イベントごとに、このアドレスからメモリに転送されます。
2. メモリ・アドレスを DMA_CPARx レジスタにセットします。データは RxNE イベントごとに、I²C_DR レジスタからメモリに転送されます。
3. 転送されるバイトの総数を DMA_CNDTRx レジスタに設定します。RxNE イベントごとにこの値はカウントダウンされます。
4. チャネルの優先順位を DMA_CCRx レジスタの PL[1:0] に設定します。
5. DMA_CCRx レジスタの DIR ビットをクリアし、ソフトウェアの必要に応じて 1/2 転送割込み、転送完了割込みの設定を行います。
6. DMA_CCRx レジスタの EN ビットをセットすることで、DMA 動作を有効とします。

転送バイト数が DMA コントローラのレジスタに設定されていた値に達したとき、DMA コントローラは転送終了信号 EOT/EOT_1 を I²C インタフェースに送ります。このとき、割込みリクエストが許可されていれば、DMA チャネルの割込みベクタに対する割込みリクエストが発生します。

注意: 受信動作に DMA が利用されるときには I²C_CR2 レジスタの ITEVTEN ビットを割込みリクエスト許可に設定してはいけません。

23.3.8 パケット・エラーの検出

通信の信頼度を上げるために PEC 計算回路が搭載されています。PEC はプログラムで設定された多項式を個々のビットに対して順に適用して計算されます。

- PEC 計算機能は I2C_CR1 レジスタの ENPEC ビットをセットすることで有効となります。PEC はアドレスと R/W ビットを含むメッセージ全体に CRC-8 を適用して計算されます。
 - 送信時、最後の TxE イベントが発生したとき、I2C_CR1 レジスタの PEC 転送ビットをセットします。PEC はその時点で送出中のバイトの後で送出されます。
 - 受信時、最後の RxNE イベントが発生したとき、I2C_CR1 レジスタの PEC ビットをセッショナルし、次に受信されるバイトが内部で計算されている PEC の値と等しくないときに受信回路に NACK を送出させます。複数のレシーバが存在する構成では、チェックの結果にかかわらず、PEC の後に NACK を送出されねばなりません。
- PECERR エラー・フラグ / 割込みリクエストは I2C_SR1 レジスタにあります。
- DMA と PEC 計算機能がともに有効とされるときには、
 - 送信時、I²C インタフェースは DMA コントローラからの EOT 信号を受取ると、最終バイトの次に自動的に PEC を送出します。
 - 受信時、I²C インタフェースが DMA コントローラから EOT_1 信号を受取ると、自動的に次に受信されるバイトが PEC であるとみなし、その検査を行います。PEC 受信後も DMA リクエストが発生します。
- 途中での PEC 転送を可能とするため、I2C_CR2 レジスタに、真に最終 DMA 転送か否かを指定する制御ビット (LAST ビット) が用意されています。マスタレシーバの最終 DMA リクエストであれば、受信されたバイトに対して自動的に NACK が送出されます。
- バス占有権の喪失によって PEC の計算は正しく続けられなくなります。

参考資料

23.4 I²C 割込み

次の表は I²C 割込みリクエストのリストを示しています。

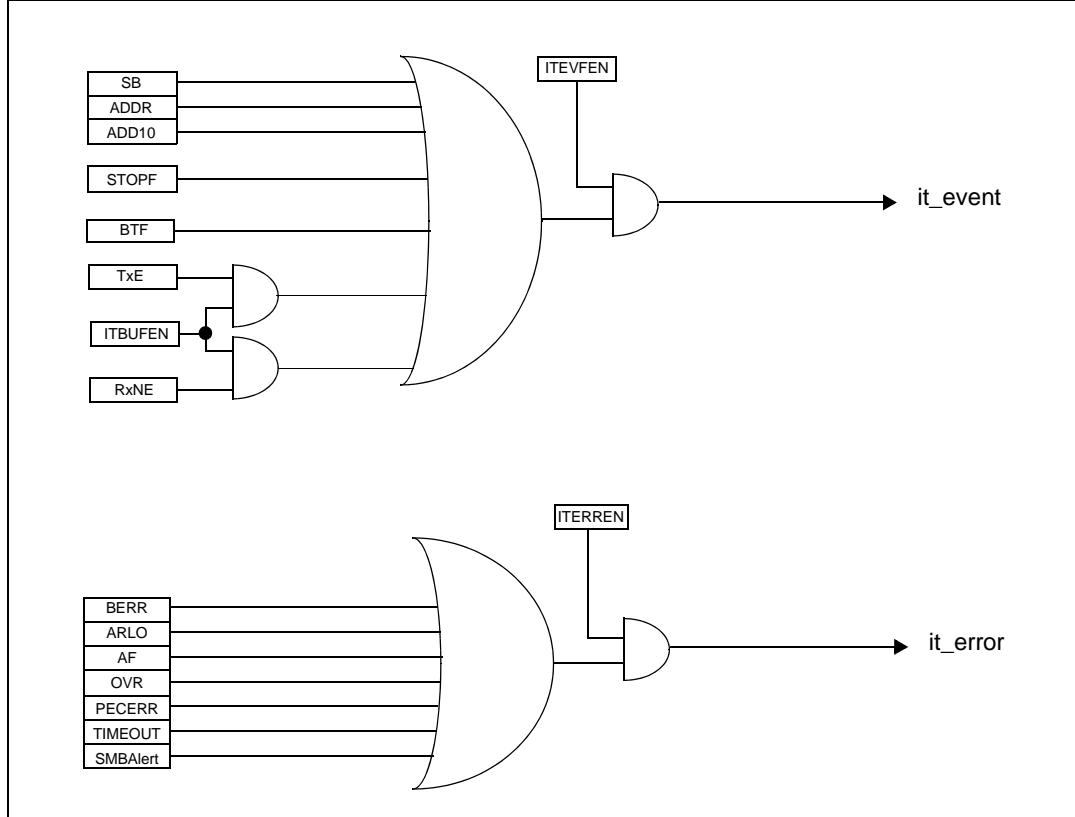
表 149. I²C 割込みリクエスト

割込みイベント	イベントフラグ	リクエスト許可制御ビット
START ビット送出 (マスタ)	SB	ITEVFEN
アドレス送出 (マスタ)、アドレス一致 (スレーブ)	ADDR	
10bit ヘッダ送出 (マスタ)	ADD10	
STOP 検出 (スレーブ)	STOPF	
データバイト転送終了	BTF	
送信バッファ・エンプティ (無し)	RxNE	ITEVFEN と ITBUFEN
受信バッファ・エンプティ	TxE	
バス・エラー	BERR	ITERREN
バス占有喪失 (マスタ)	ARLO	
否定応答	AF	
オーバラン / アンダラン	OVR	
PEC エラー	PECERR	
タイムアウト / T ローエラー	TIMEOUT	
SMBus アラート	SMBALERT	

注意： 1 *SB、ADDR、ADD10、STOPF、BTF、RxNE、TxE* のビットは論理和がとられ、同じ割込みリクエストチャネルに接続されています。

2 *BERR、ARLO、AF、OVR、PECERR、TIMEOUT、SMBALERT* のビットは論理和がとられ、同じ割込みチャネルに接続されています。

参考資料

図 236. I²C 割込みマッピング

参考資料

23.5 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 構成ビットに応じて、SMBUS タイムアウトは通常通りの動作を続行、あるいは停止します。詳細については、[セクション 26.15.2: タイマ、ウォッチドッグ、bxCAN と PC に対するデバッグ・サポート \(701 ページ\)](#) を参照してください。

23.6 I²C レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#) を参照してください。

23.6.1 制御レジスタ 1 (I2C_CR1)

アドレスオフセット t: 0x00

リセット値 : 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SW RST	予約済み	ALERT	PEC	POS	ACK	STOP	STAR T	NO STRETCH	ENGC	EN PEC	EN ARP	SMB TYPE	予約済み	SM BUS	PE

Bit 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 はリセット状態に置かれます。このビットをクリアする前に、I²C ラインが解放されており、バスが使用されていないことを確認する必要があります。

0: I²C ペリフェラルはリセット状態にありません。

1: I²C ペリフェラルはリセット状態です。

注: このビットはバス上で STOP 信号が検出されていなければ、BUSY ビットがセットされているときに使用できます。

Bit 14 予約済みビット、ハードウェアで 0 に固定されています。

Bit 13 ALERT SMBus アラート

このビットはソフトウェアでセット / クリアができます。また、PE=0 の時にはハードウェアでクリアされます。

0: SMBAlert 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には NACK が続きます。

1: SMBAlert 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には ACK が続きます。

Bit 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でセット / クリアができます。また、PEC が転送されたとき、START 信号、STOP 信号が検出されたとき、及び PE=0 のときにハードウェアでクリアされます。

0: PEC は転送されません。

1: PEC が転送されます。(送信・受信モードとも)

注: バス占有の喪失により、PEC の計算は妨げられることがあります。

参考資料

Bit 11 POS: 確認応答 (Acknowledge)/PEC 位置 (データ受信)

このビットはソフトウェアでセット / クリアができます。また、PE=0 の時、ハードウェアでクリアされます。

0: ACK ビットがシフト・レジスタに受信されたバイトの (N)ACK を制御します。PEC ビットは、シフト・レジスタに受信されたバイトが PEC であることを示します。

1: ACK ビットは次にシフト・レジスタに受信されるバイトの (N)ACK を制御します。PEC ビットは、次にシフト・レジスタに受信されるバイトが PEC であることを示します。

注:

このビットはデータ受信が開始される前に設定されなければなりません。

この設定は 2 個のデータバイトのみが存在する ADDR 拡張イベントの時にのみ使用できます。

Bit 10 ACK: 確認応答 (Acknowledge) 送出

このビットはソフトウェアでセット / クリアされます。また PE=0 の時にハードウェアがクリアします。

0: 確認応答は戻されません。

1: 一致するアドレスバイト、データバイトを受信したとき、確認応答を送出します。

Bit 9 STOP: STOP 信号送出

このビットはソフトウェアでセット / クリアができます。また、STOP 信号が検出されたときにハードウェアでクリアされ、タイムアウトエラーが検出されるとセットされます。

マスター・モード:

0: STOP 信号は発生しません。

1: 実行中のバイト送信、もしくは START 信号の送出の後、STOP 信号を送出します。

スレーブ・モード:

0: STOP 信号は発生しません。

1: 実行中のバイト転送後に SCL ライン及び SDA ラインを解放します。

注:

マスター・モードでは、STOP がリクエストされたときには I2C_SR1 レジスタの BTF ビットをクリアしなければなりません。

Bit 8 START: START 信号発生

このビットはソフトウェアでセット / クリアができます。また、START 信号が送られたとき、もしくは PE=0 の時にハードウェアがクリアします。

マスター・モード:

0: START 信号は発生しません。

1: RESTART 信号 (START 信号の繰返し) が送出されます。

スレーブ・モード:

0: START 信号は発生しません。

1: バスが解放状態であれば、START 信号を送出します。

Bit 7 NOSTRETCH: クロック伸張禁止 (スレーブ・モード)

このビットはスレーブ・モードで ADDR もしくは BTF ビットがセットされているときのクロックの伸張を禁止します。この状態はソフトウェアでこのビットをクリアするまで続きます。

0: クロック伸張許可

1: クロック伸張禁止

Bit 6 ENGC: General Call 有効

0: General Call は無効です。アドレス 00h は受け付けられません。

1: General Call は有効です。アドレス 00h は受け付けられます。

Bit 5 ENPEC: PEC イネーブル

0: PEC 計算はディセーブル状態です。

1: PEC 計算はイネーブル状態です。

参考資料

RM0008

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

Bit 4 ENARP: ARP イネーブル

0: ARP はディセーブル状態です。

1: ARP はイネーブル状態です。

SMBTYPE=0 のとき SM バスのデフォルトアドレスは受け入れられます。

SMBTYPE=1 のとき SM バスホストアドレスが受け入れられます。

Bit 3 SMBTYPE: SMBus タイプ

0: SMBus デバイス

1: SMBus ホスト

Bit 2 予約済みビット、ハードウェアで 0 に固定されています。

Bit 1 SMBUS: SMBus モード

0: I²C モード

1: SMBus モード

Bit 0 PE: ペリフェラルイネーブル

0: ペリフェラルはディセーブル状態。

1: ペリフェラルはイネーブル状態、入出力ピンは SMBus ビットの設定に応じたオルタネート機能が選択されます。

注: 通信中にこのビットがクリアされると、進行中の通信が終わりアイドル状態に戻った時点でペリフェラルはディセーブル状態となります。このとき、すべてのビットはリセットされます。
マスタ・モードでは、通信終了までこのビットをクリアしてはいけません。

23.6.2 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	ITBUF EN	ITEVT EN	ITER REN	予約済み						FREQ[5:0]			

rw rw

Bits 15:13 予約済みビット、ハードウェアで 0 に固定されています。

Bit 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: このビットはマスタ・モードのレシーバで、最終受信データに対して NACK を送出するために使用されます。

Bit 11 DMAEN: DMA リクエスト許可

0: DMA リクエスト禁止

1: TxE=1 もしくは RxNE =1 のとき DMA リクエストを許可します。

Bit 10 ITBUFEN: バッファ割込みリクエスト許可

0: TxE = 1 もしくは RxNE = 1 は割込みリクエストを生じません。

1: TxE = 1 もしくは RxNE = 1 はイベント割込みをリクエストします。(DMAEN ビットの状態には依存しません。)



参考資料

Bit 9 ITEVTEN: イベント割込みリクエスト許可

0: イベント割込みリクエスト禁止

1: イベント割込みリクエスト許可

この割込みリクエストは以下の場合に発生します。

-SB = 1 (マスター)

-ADDR = 1 (マスター / スレーブ)

-ADD10= 1 (マスター)

-STOPF = 1 (スレーブ)

-BTF = 1、ただし TxE もしくは 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

Bit 8 ITERREN: エラー割込みリクエスト許可

0: エラー割込みリクエスト禁止

1: エラー割込みリクエスト許可

この割込みリクエストは以下の場合に発生します。

-BERR = 1

-ARLO = 1

-AF = 1

-OVR = 1

-PECERR = 1

-TIMEOUT = 1

-SMBAlert = 1

Bits 7:6 予約済みビット、ハードウェアで 0 に固定されています。

Bits 5:0 FREQ[5:0]: ペリフェラル・クロック周波数

入力クロック周波数は正しいタイミングを生成するように設定する必要があります。

設定可能な周波数範囲は 2 MHz から 36MHz までです。

000000: 設定禁止

000001: 設定禁止

000010: 2 MHz

...

100100: 36MHz

100101 以上 : 設定禁止

参考資料

23.6.3 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	Res.	予約済み		ADD[9:8]		ADD[7:1]		ADD0							
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 15 ADDMODE: アドレスモード (スレーブ・モード)

0: 7bit スレーブアドレス (10bit アドレスは受け付けません。)

1: 10bit スレーブアドレス (7bit アドレスは受け付けません。)

Bit 14 1にセットし、この値を保持しなければなりません。

Bits 13:10 予約済みビット、ハードウェアで0に固定されています。

Bits 9:8 ADD[9:8]: インタフェース・アドレス

7bit アドレスモード: 無視

10bit アドレスモード: アドレスの上位 (bits 9:8)

Bits 7:1 ADD[7:1]: インタフェース・アドレス

アドレスの下位 (bits 7:1)

Bit 0 ADD0: インタフェース・アドレス

7bit アドレスモード: 無視

10bit アドレスモード: アドレスの最下位 (bit 0)

23.6.4 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	rw	rw	rw	rw	rw	rw	rw	rw

Bits 15:8 予約済みビット、ハードウェアで0に固定されています。

Bits 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のアドレス値 (bits 7:1)

Bit 0 ENDUAL: デュアルアドレスモード

0: 7bit アドレスモードで、OAR1のみがアドレスとして受けられます。

1: 7bit アドレスモードで、OAR1とOAR2の両方がアドレスとして受けられます。

参考資料

23.6.5 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	RW

Bits 15:8 予約済みビット、ハードウェアで 0 に固定されています。

Bits 7:0 DR[7:0]: 8bit データ・レジスタ ⁽¹⁾⁽²⁾⁽³⁾

受信されたバイト、もしくはバスに送出されるバイト。

- トランスマッタ・モード : バイトが DR レジスタに書込まれると、送信動作は自動的に始まります。送信が始まった後 ($TxE=1$) は、次に送信されるデータが DR に書込まれると連続的な送信動作が続けられます。
- レシーバ・モード : 受信されたデータは DR にコピーされます。 $(RxNE=1)$ DR レジスタの受信データは次のデータが受信される前に読み出されねばなりません。読み出されないとオーバ・ラン・エラーが発生し、後で受信されたバイトが失われます。

1. スレーブ・モードでは、アドレスは DR レジスタにコピーされません。
2. 書込みの競合には対応していません。(DR の書き込みは $TxE=0$ の時に限られます。)
3. ACK パルスで ARLO イベントが発生すると、受信バイトは DR レジスタにコピーされないため、これを読み出すことはできません。

参考資料

23.6.6 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	TIME OUT	予約済み	PEC ERR	OVR	AF	ARLO	BERR	TxE	RxNE	予約済み	STOP F	ADD10	BTB	ADDR	SB

rc_w0 rc_w0 rc_w0 rc_w0 rc_w0 rc_w0 rc_w0 r r r STOP F ADD10 BTB ADDR SB

Bit 15 SMBALERT: SMBus アラート

SMBus ホストモードで、

0: no SMBAlert は発生していません。

1: SMBAlert イベントがピン上で検出されています。

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 応答アドレスヘッダは受信された SMBAlert LOW を指しています。

- ソフトウェアで 0 を書込むとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

Bit 14 TIMEOUT: タイムアウト、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 が 25 ms の間 LOW レベルを保っています。(タイムアウト)

もしくは

マスターで、クロックのローレベルの累積時間が 10 ms を超えました。(タイムアウト)

もしくは

スレーブで、クロックのローレベルの累積時間が 25 ms を超えました。(タイムアウト)

- スレーブ・モードでは: 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した。
- マスター・モードでは: STOP 信号がハードウェアによって送出されました。
- ソフトウェアで 0 を書込むとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

Bit 13 予約済みビット、ハードウェアで 0 に固定されています。

Bit 12 PECERR: PEC 受信時のエラー検出

0: PEC エラーなし。PEC 受信に対して ACK を返しました。(ACK=1 のとき)

1: PEC エラー発生。PEC 受信に対して NACK を返しました。(ACK は任意)

- ソフトウェアで 0 を書込むとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

Bit 11 OVR: オーバラン / アンダラン

0: オーバラン / アンダランは発生していません。

1: オーバラン、もしくはアンダランが発生。

- スレーブ・モードで NOSTRETCH=1 の場合、次の条件でハードウェアによりセットされます。

- 受信時、新たなバイト (ACK パルスを含む) が受信され、DR レジスタが読み出されていないとき。
受信バイトは失われていません。

- 送信時、新たなバイト送信が可能で、DR レジスタへの書き込みがまだ行われていないとき。同じ
バイトが二度送出されます。

- ソフトウェアで 0 を書込むとクリアされます。また、PE=0 の時にはハードウェアがこのビット
をクリアします。

注: DR への書き込みが SCL の立上りエッジ近くで行われると、送出されるデータは不定で、ホールド
タイムエラーが発生します。

参考資料

Bit 10 AF: 送信後の確認応答 (Acknowledge) なし

- 0: 確認応答のエラーは発生していません。
- 1: 確認応答のエラーが発生しました。
 - 確認応答が戻されないとき、ハードウェアによってセットされます。
 - ソフトウェアで 0 を書込むとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

Bit 9 ARLO: バス占有 (Arbitration) の喪失 (マスタ・モード)

- 0: バス占有喪失は検出されていません。
- 1: バス占有の喪失が検出されました。
 - インターフェースが他のマスタにバスの占有権渡したとき、ハードウェアによってセットされます。
 - ソフトウェアで 0 を書込むとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

ARLO イベント後、インターフェースは自動的にスレーブ・モード (M/SL=0) に切り替わります。

注: SMBUS では、スレーブ・モードにおけるデータのアービトレーションはデータ・フェーズもしくは確認応答でのみで生じます。(アドレスに対する確認応答では生じません。)

Bit 8 BERR: バスエラー - 不正な START/STOP 信号

- 0: START もしくは STOP 信号のタイミング不良は発生していません。
- 1: START もしくは STOP 信号のタイミング不良が発生しました。
 - START もしくは STOP 信号のタイミング不良が検出されたとき、ハードウェアによってセットされます。
 - ソフトウェアで 0 を書込むとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

Bit 7 TxE: データ・レジスタ・エンプティ (送信時)

- 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。
 - 送信動作で 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。
 - ソフトウェアで DR レジスタへの書き込みを行うとクリアされます。また、START 信号、STOP 信号の後、及び PE=0 の時にはハードウェアがこのビットをクリアします。

注: TxE は NACK が受信された時、及び次に送出されるバイトが PEC の時 (PEC=1) にはセットされません。

Bit 6 RxNE: データ・レジスタ・データ有効 (受信時)

- 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。
 - 受信時、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うとクリアされます。また、START 信号、STOP 信号の後、及び PE=0 の時にはハードウェアがこのビットをクリアします。

注: RxNE は ARLO イベント時にはセットされません。

Bit 5 予約済みビット、ハードウェアで 0 に固定されています。

参考資料

Bit 4 STOPF: STOP 信号検出 (スレーブ・モード)

- 0: STOP 信号は検出されていません。
- 1: STOP 信号が検出されました。
 - 確認応答後に (ACK=1 のとき) スレーブがバス上で STOP 信号を検出し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 CR1 レジスタに書き込みを行うとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

注: STOPF ビットは NACK が受信されたときにはセットされません。

Bit 3 ADD10: 10 ビットアドレスリング (マスター・モード)

- 0: ADD10 イベントは発生していません。
- 1: マスターは最初のアドレスバイト (ヘッダ) を送出しました。
 - マスターが 10bit アドレスモードの最初のバイトを送出すると、ハードウェアがこのビットをセットします。
 - 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 番目のバイトを書き込むとクリアされます。また、PE=0 の時にはハードウェアがこのビットをクリアします。

注: ADD10 ビットは NACK 受信後はセットされません。

Bit 2 BTF: バイト転送の完了

- 0: バイト転送は完了していません。
- 1: データバイトの転送が正常に終了しました。
 - NOSTRETCH=0 のとき、及び次の場合にハードウェアによりセットされます。
 - 受信では、新たなバイトが (ACK パルスを含めて) 受信され、DR がまだ読み出されていないとき (RxNE=1)
 - 送信では、新たなバイトの送信が可能で、データが DR にまだ書き込まれていないとき (TxE=1)
 - ソフトウェアで SR1 を読み出し、続いて DR の読み書きを行ったとき、または送信時の START もしくは STOP 状態後や PE=0 のときハードウェアでセットされます。

注:

BTB ビットは NACK 受信後はセットされません。

BTB ビットは次に送信されるバイトが PEC (I2C_SR2 レジスタの TRA=1 かつ I2C_CR1 レジスタの PEC=1) であるときはセットされません。

Bit 1 ADDR: アドレス送信 (マスター) / アドレス一致 (スレーブ)

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、PE=0 の時にはハードウェアがこのビットをクリアします。

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: 受取ったアドレスが一途しました。
 - 受取ったアドレスが OAR レジスタの内容と一致したとき、もしくは同報 (general call)、SMBus デバイスデフォルトアドレス、SMBus ホスト、SMBus アラートのいずれかが検出されたときに、ハードウェアによりセットされます。(それぞれ、イネーブルとなる設定の場合)

アドレス送信 (マスター)

- 0: アドレス送信は終了していません。
- 1: アドレス送信が終了しました。
 - 10 ビットアドレスモードでは、2 番目のバイトの ACK でセットされます。
 - 7 ビットアドレスモードでは、アドレスバイトの ACK でセットされます。

注: ADDR は NACK が戻されたときにはセットされません。

参考資料

Bit 0 **SB:** START 信号送出完了 (マスタ・モード)
0: START 信号送出なし
1: START 信号が送出されました。
-START 条件が生成されたときにセットされます。
-ソフトウェアで SR1 レジスタを読出し、続いて DR レジスタへの書き込みを行ったとき、及び PE=0 のときにハードウェアでクリアされます。

23.6.7 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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
							PEC[7:0]	DUALF	SMB HOST	SMB DEF AULT	GEN CALL	予約済み	TRA	BUSY	MSL

- Bits 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 を保持しています。
- Bit 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
1: 受信アドレスが OAR2 と一致
-STOP 信号、START 信号の繰返しの後、及び PE=0 の時にハードウェアがこのビットをクリアします。
- Bit 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 信号、START 信号の繰返しの後、及び PE=0 の時にハードウェアがこのビットをクリアします。
- Bit 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は受信されていません。
1: ENARP=1 のとき SMBus デバイスデフォルトアドレスが受信されました。
-STOP 信号、START 信号の繰返しの後、及び PE=0 の時にハードウェアがこのビットをクリアします。
- Bit 4 **GENCALL:** General Call アドレス (スレーブ・モード)
0: General Call は受信されていません。
1: ENGC=1 のとき、General Call アドレスが受信されました。
-STOP 信号、START 信号の繰返しの後、及び PE=0 の時にハードウェアがこのビットをクリアします。
- Bit 3 予約済みビット、ハードウェアで 0 に固定されています。
- Bit 2 **TRA:** トランスマッタ / レシバ
0: データバイトが受信されました。
1: データバイトが送信されました。
このビットはアドレス・フェーズの終わりで、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、ハードウェアは、STOP 条件が検出されたとき (STOPF=1)、再スタート条件、バス占有の喪失 (ARLO=1)、もしくは PE=0 の時にこのビットをクリアします。

参考資料

Bit 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 もしくは SCL がローレベルのとき、ハードウェアでセットされます。
 - STOP 条件が検出されたとき、ハードウェアでクリアされます。
- このビットはバス上で通信が進行中であることを示します。このビットはインターフェースがディセーブル状態 (PE=0) のときでも更新されます。

Bit 0 MSL: マスタ / スレーブ

- 0: スレーブ・モード
- 1: マスタ・モード
 - インターフェースがマスタ・モードなる (SB=1) とハードウェアでセットされます。
 - バス上で STOP 条件が検出されたときやバス占有権を失ったとき (ARLO=1)、もしくは PE=0 のときにハードウェアでリセットされます。

参考資料

23.6.8 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

- 注意 : 1 F_{PCLK1} は 400kHz の Fast クロックを生成するのに必要な 10MHz の倍数です。
2 CCR レジスタは I2C がディセーブル ($PE=0$) の場合に限り設定されなければなりません。

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

Bit 15 F/S I²C マスタ・モード選択

0: 標準 I²C モード (Standard Mode I²C)

1: 高速 I²C モード (Fast Mode I²C)

Bit 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)

Bits 13:12 予約済みビット、ハードウェアで 0 に固定されています。

Bits 11:0 CCR[11:0]: 高速 / 標準モード・クロック制御レジスタ (マスタ・モード)

マスタ・モードにおける SCL クロックを制御します。

標準モード、もしくは SMBus:

$$T_{high} = CCR * T_{CK}$$

$$T_{ow} = CCR * T_{CK}$$

高速モード:

DUTY = 0 のとき :

$$T_{high} = CCR * T_{CK}$$

$$T_{ow} = 2 * CCR * T_{CK}$$

DUTY = 1: (上限 400 kHz) のとき

$$T_{high} = 9 * CCR * T_{CK}$$

$$T_{ow} = 16 * CCR * T_{CK}$$

例えば、標準モードで 100kHz の SCL クロックを得るには、

FREQR = 08、 $T_{ck} = 125\text{ns}$ のとき CCR には 28h が設定されます。

(28h <= 40d x 125ns = 5000 ns.)

注:

1. 設定できる最小値は 04h です。ただし、FAST DUTY モードでは最小値は 01h となります。
2. t_{high} には SCLH の立上りエッジが含まれます。
3. t_{low} には SCLH の立下りエッジが含まれます。
4. 記述されているタイミングはフィルタを通さないものです。
5. CCR レジスタは I2C がディセーブル ($PE=0$) の時に限り設定します。
6. $f_{CK}=10\text{MHz}$ の倍数は 400kHz の高速クロックを生成する時に必要です。

参考資料

23.6.9 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	rw	rw

Bits 15:6 予約済みビット、ハードウェアで 0 に固定されています。

Bits 5:0 TRISE[5:0]: 高速 / 標準モードにおける最大立上り時間 (マスター・モード)

これらのビットには、I²C バス仕様で与えられている SCL 最大立上り時間より 1 だけ大きな値を設定しなければなりません。

例えば、標準モードで許される SCL の最大立上り時間は 1000 ns です。

I²C_CR2 レジスタの FREQ[5:0] の値が 08h で、t_{CK} = 125 ns のとき、TRISE[5:0] フィールドには 09h が設定されねばなりません。

(1000 ns / 125 ns = 8 + 1)

フィルタ値も TRISE[5:0] に加えることができます。

計算値が整数でないときには、t_{HIGH} パラメータを考慮し、TRISE[5:0] にはその整数部を設定します。

注: TRISE[5:0] の設定は I²C がディセーブル状態 (PE=0) の時に限られます。

参考資料

23.6.10 I²C レジスタ・マップ

次の表は I²C のレジスタ・マップとリセット値を示しています。

表 150. I²C - レジスタ・マップとリセット時の値

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																																
	リセット値																																
0x04	I2C_CR2																																
	リセット値																																
0x08	I2C_OAR1																																
	リセット値																																
0x0C	I2C_OAR2																																
	リセット値																																
0x10	I2C_DR																																
	リセット値																																
0x14	I2C_SR1																																
	リセット値																																
0x18	I2C_SR2																																
	リセット値																																
0x1C	I2C_CCR																																
	リセット値																																
0x20	I2C_TRISE																																
	リセット値																																

レジスタのアドレス範囲に関しては [表 1 \(36 ページ\)](#) を参照してください。



24 USART

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、中容量デバイスです。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、大容量デバイスです。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適用されます。

24.1 概要

USART (Universal Synchronous Asynchronous Receiver Transmitter) は、標準的な NRZ 調歩式シリアルデータ形式を必要とする外部機器との間の全二重データ交換の手段を提供しています。USART は端数に対応できるボーレート生成回路を使用し、非常に広範囲のボーレートへの対応を可能としています。

USART は同期式の一方向通信、単線半二重通信もサポートにも対応します。また、LIN (Local Interconnection Network)、スマートカード・プロトコル、IrDA (Infrared Data Association) SIR ENDEC 規格、及びモデム接続 (CTS/RTS) にも対応することができます。また、マルチプロセッサ通信も可能です。

マルチバッファ構成を実現する DMA を利用し、高速のデータ通信が可能です。

24.2 USART の主な機能

- 全二重、調歩式通信
- NRZ 標準フォーマット（マーク / スペース）
- ボーレート生成における端数のサポート
 - 送受信共通のボーレートは最高 4.5 MBit/s まで設定することができます。
- プログラム設定可能なデータ語長（8 もしくは 9bit）
- 選択可能な STOP ビット数 -1 及び 2 STOP ビットをサポート
- LIN マスター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 モードで動作しているときの、13 ビットブレーク生成と 10/11 ビットブレーク検出
- 同期シリアル・モード送信における送信クロックの出力
- IrDA SIR エンコーダ・デコーダ
 - ノーマル・モードにおける 3/16bit パルス幅のサポート
- スマートカード・エミュレーション機能
 - スマートカード・インターフェースは ISO 7816-3 規格で定められている非同期スマートカード・プロトコルをサポート
 - スマートカード・モードのための 0.5, 1.5 STOP ビット
- 単線半二重通信
- DMA (Direct Memory Access) を利用したマルチバッファ通信
 - 内蔵標準 DMA 機能を使用し、送受信バイトを所定の SRAM ドメインにバッファリング
- 送信回路、受信回路個別のイネーブル・ビット制御
- データ転送関係のフラグ
 - 受信バッファ・フル
 - 送信バッファ・エンプティ
 - 送信完了フラグ
- パリティ制御
 - パリティビットの送信
 - 受信データにおけるパリティ検査
- 4 種のエラー検出フラグ
 - オーバ・ラン・エラー
 - ノイズ・エラー
 - フレーム・エラー
 - パリティ・エラー
- 10 種の割込み要因フラグ
 - CTS 信号変化
 - 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
- マルチプロセッサ間通信 – アドレスが不一致でミュートモードに遷移
- ミュートモードからのウェークアップ（アイドル・ライン検出 / アドレスマーク検出）
- 2種類の受信回路ウェークアップ・モード：アドレスビット（MSB の 9 ビット目）、アイドルライン

24.3 USART の機能詳細

インタフェースは 3 個のピンを用いて他の外部デバイスと接続されます。（図 237 を参照） USART 両方向通信では、少なくとも受信データ入力 (RX) と送信データ出力 (TX) の 2 個のピンを必要とします。

RX: 受信データ入力はシリアルデータの入力です。有効な入力データとノイズを区別し、データを回復するために、多重サンプリング手法が使用されています。

TX: 送信データ出力。送信回路がディセーブル状態の時には、出力ピンの状態は入出力ポートとしての構成に戻ります。送信回路がイネーブル状態で、送信すべきデータがないときには、TX ピンはハイレベルとなります。単線モード及びスマートカード・モードのときには、データの送受信の双方に使用されます。（USART レベルでは、データは SW_RX から受信されています。）

これらのピンを通してシリアルデータの送受信が行われます。通常の USART モードでは、フレームは次の部分から構成されています。

- 送信もしくは受信に先立つアイドル・ライン
- スタート・ビット
- データワード（8 もしくは 9bit）、下位ビットが先に送出されます。
- フレームの終わりを示す 0.5、1、1.5、2 のいずれかの幅の STOP ビット
- このインターフェースが使用するボーレート生成回路は 12bit の整数部と 4bit の小数部をもち、端数の指定が可能です。
- 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) – 12bit の整数部と 4bit の小数部
- ガードタイムレジスタ (USART_GTPR) – スマートカード・モードで使用されます。

レジスタの個々のビットの説明は [セクション 24.6: USART レジスタ \(666 ページ\)](#) を参照してください。

次のピンは同期シリアル・モードの接続で必要になります。

- SCLK: 送信クロック出力。このピンからの出力は SPI マスター・モード（スタート・ビット / STOP ビットがなく、最後のデータビットに対するクロックパルスの送出はソフトウェアに依存するオプション）と通信するための同期シリアル・モード送信における送信データクロックです。これと並行して RX 上のデータを受信することができます。これはシフト・レジスタを有する周辺機器（例：LCD ドライバ）を制御する際に使用することができます。クロックの位相と極性はソフトウェアで指定できます。スマートカード・モードでは SCLK はスマートカードのクロックとして使用することができます。

参考資料

次のピンは IrDA モードのインターフェースの際に必要になります。

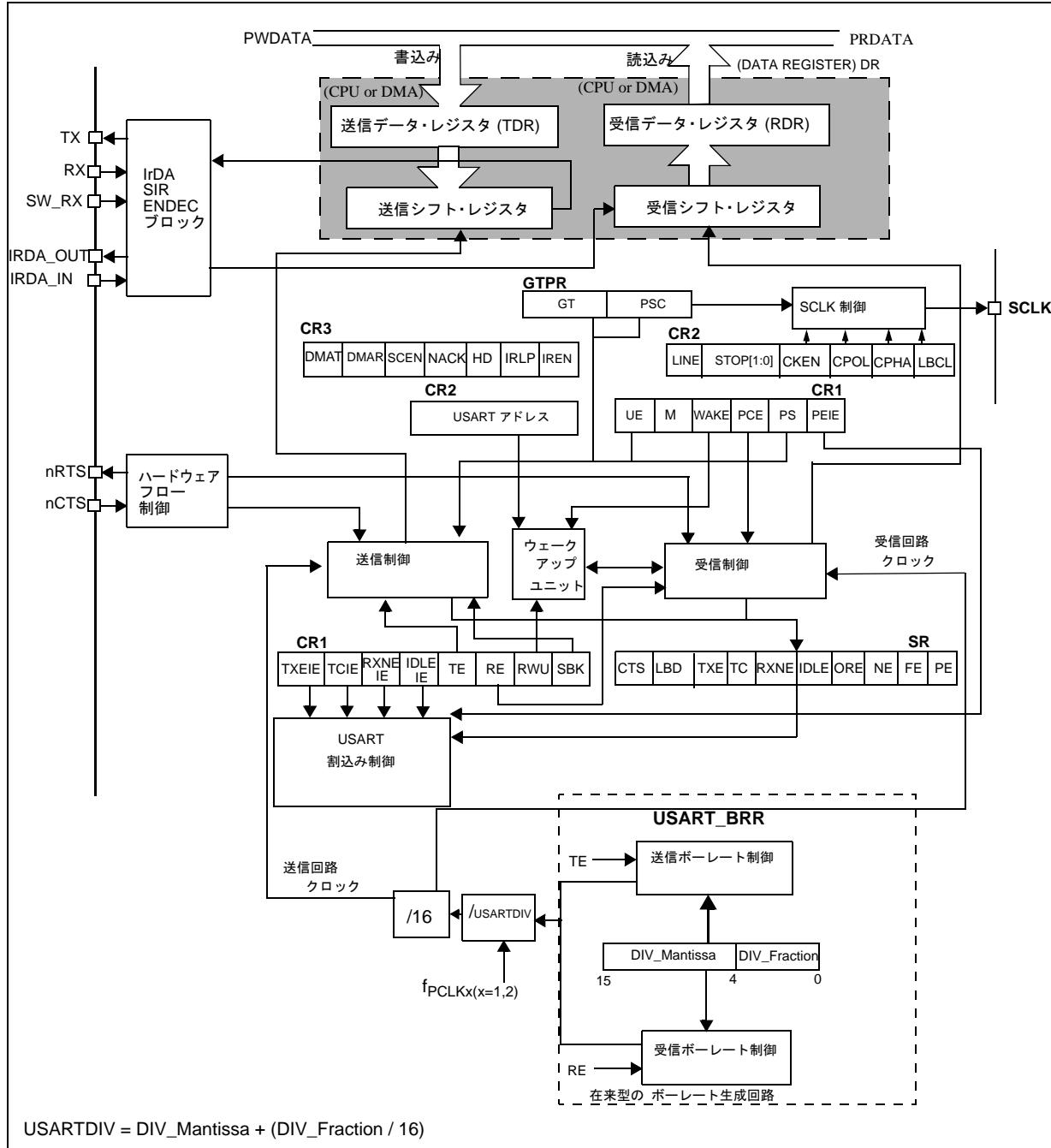
- IrDA_RXD: 受信データ入力。IrDA モードにおけるデータ入力となります。
- IrDA_TXD: 送信データ出力。IrDA モードにおけるデータ出力です。

以下のピンはハードウェア・フロー制御で必要になります。

- nCTS: (Clear To Send) ハイレベルのとき、実行中のバイト送出が終わると、送信動作を停止させます。
- nRTS: (Request To Send) USART がデータを受ける用意があることをローレベルで示します。

参考資料

図 237. USART ブロック図



USARTDIV = DIV_Mantissa + (DIV_Fraction / 16)

24.3.1 USART の特性詳細

ワード長は USART_CR1 レジスタの M ビットをプログラムで設定することで、8bit もしくは 9bit のいずれかを選択することができます。(図 238 を参照)

TX ピンはスタート・ビットでローレベルとなり、STOP ビットでハイレベルとなります。

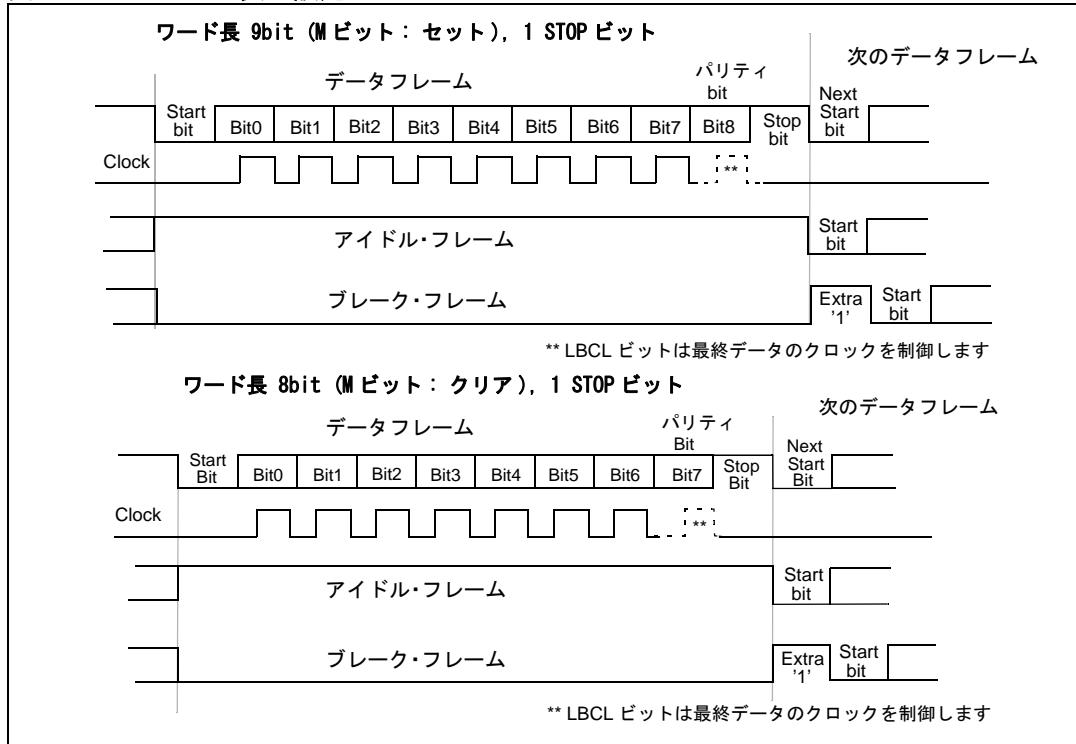
データを含む次のフレームのスタート・ビットまでのすべて 1 のフレーム (STOP ビットも含まれます) をアイドル・フレームと称します。

1 フレーム分に相当する時間の間 0 を受信した場合、これをブレーク・フレームと称します。ブレーク・フレームの終わりには、送信回路で 1 もしくは 2 個の STOP ビット (論理値としては 1 のビット) が挿入されます。

送信回路・受信回路は同じボーレート・ジェネレータで駆動されますが、それぞれのインターブル・ビットに対応して、送受信回路それぞれにクロックが送られます。

次の図には各フレームをブロックが図示されています。

図 238. ワード長の設定



24.3.2 送信回路

送信回路は M ビットの状態に応じて 8 もしくは 9bit ワードのデータを送出します。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・レジスタのデータは TX ピンから出力され、対応するクロックパルスが SCLK ピンから出力されます。

送信データ

USART 送信の間、データは下位ビットから順に TX ピンに送出されます。このモードではバッファ (TDR) からなる USART_DR レジスタが内部バスと送信シフト・レジスタの間におかれていています。(図 237 を参照)

どの特性も 1 つのビットピリオドについてロジックのレベルがローのスタート・ビットによりプリスケールされます。その特性は、STOP ビットの構成数によりターミネートされます。

以下の STOP ビットは USART によりサポートされます。

- 注意： 1 TE ビットはデータの送信中にクリアしてはいけません。送信中の TE ビットのクリアではボーレート生成回路が停止するため、TX ピンのデータに悪影響を与えます。送信中のデータは失われます。
2 アイドル・フレームは TE ビットがイネーブル状態にセットされた後に送出されます。

STOP ビット数の選択

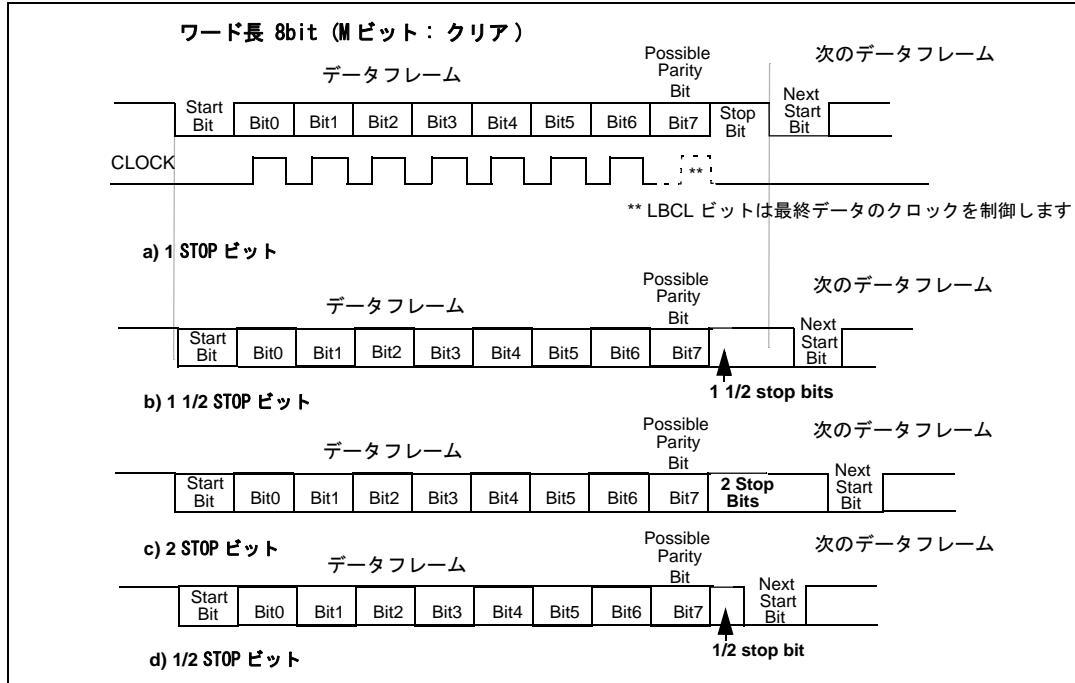
キャラクタごとに送出される STOP ビットの数は制御レジスタ 2 のビット 13:12 で選択できます。

- 1 STOP ビット : STOP ビット数のデフォルト値です。
- 2 STOP ビット : 通常の USART モード、単線モード、及びモデムモードでサポートされています。
- 0.5 STOP ビット : スマートカード・モードにおける受信動作で使用されます。
- 1.5 STOP ビット : スマートカード・モードの送信動作で使用されます。

アイドル・フレームの送出は STOP ビットを含んでいます。

ブレーク送信では、10 個 (M = 0 の時) もしくは 11 個 (M = 1 の時) のローレベルビットの後に、設定されている数の STOP ビットを送出します。10 もしくは 11bit 分より長いブレーク信号を送出することはできません。

図 239. 制御 STOP ビットの設定



操作手順 :

1. USART_CR1 レジスタの UE ビットに 1 を書き込み、USART をイネーブル状態とします。
2. 語長を選択するため、USART_CR1 レジスタの M ビットを設定します。
3. USART_CR2 レジスタで STOP ビット数を選択します。
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・ビット (DMAT) を設定します。また、マルチバッファ通信の項の説明にしたがって、DMA レジスタの設定を行います。
5. USART_CR1 の TE ビットをセットし、最初の送信としてアイドル・フレームを送出します。
6. USART_BRR レジスタを使用して、希望のボーレートを選択します。
7. 送信するデータを USART_DR レジスタに書込みます。これによって TXE ビットがクリアされます。シングルバッファ動作の場合、データごとにこれを繰返します。

バイト単位通信

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 ビットがハードウェアによってセットされたときには、次のことを示しています。

- データが TDR からシフト・レジスタに移され、送出動作が開始されました。
- TDR レジスタは空の状態です。
- 前のデータを上書きすることなく、次のデータを USART_DR レジスタに書込むことができます。

TXEIE ビットがセットされていれば、フラグは割込みリクエストを発生させます。

送信動作中は、USART_DR レジスタへの書き込み命令はデータを TDR に格納します。このデータは動作中の送信動作が終わり次第、シフト・レジスタに送られます。

送信動作中でなければ、USART_DR レジスタへの書込み命令はデータを直にシフト・レジスタに書込みます。これによりデータの送信が開始され、TXE ビットが直ちにセットされます。

フレームの送信が（ステップビットの送出まで）完了すると TC ビットがセットされ、USART_CR1 レジスタの TCIE ビットがセットされいれば割込みリクエストが発生します。

TC ビットはソフトウェアで次の操作を行うことでクリアされます。

1. USART_SR レジスタの読出し
2. USART_DR レジスタへの書込み

注意： TC ビットは、直接 0 を書込んだ場合もクリアされます。マルチバッファ通信の場合には、この方法でクリアすることがお勧められます。

ブレーク・フレーム

SBK ビットをセットするとブレーク・フレームが送出されます。ブレーク・フレームの長さは M ビットの設定に依存します。（図 238 を参照）

SBK ビットに 1 がセットされると、進行中の送信動作が終わり次第、TX ラインにブレーク・フレームが送出されます。このビットはブレーク・フレームのストップピットが送出されたときにハードウェアでクリアされます。次のフレームのスタート・ビットが間違なく認識されるよう、USART は最後のブレーク・フレームの後に論理値 1 のビットを挿入します。

注意： ブレーク・フレームの送出前に SBK ビットをソフトウェアでクリアすると、ブレーク・フレームは送出されません。2 個のブレークを連続して送出するには、最初のブレーク・フレームの STOP ビットの後で SBK をセットしなければなりません。

アイドル・フレーム

TE ビットをセットすることで、USART は最初のデータフレームの前にアイドル・フレームを送出します。

24.3.3 受信回路

USART は、USART_CR1 レジスタの M ビットの応じて、8 もしくは 9bit のデータを受信することができます。

文字データの受信

USART の受信動作では、下位ビットから順に RX ピンを通して読み込まれます。このモードではバッファ (RDR) からなる USART_DR レジスタが内部バスと受信シフト・レジスタの間にあります。

受信手順：

1. USART_CR1 レジスタの UE ビットに 1 を書き込み、USART をイネーブル状態とします。
2. 語長を選択するため、USART_CR1 レジスタの M ビットを設定します。
3. USART_CR2 レジスタで STOP ビット数を選択します。
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・ビット (DMAT) を設定します。また、マルチバッファ通信の項の説明にしたがって、DMA レジスタの設定を行います。
5. USART_BRR レジスタを使用して、希望のボーレートを選択します。
6. USART_CR1 レジスタの RE ビットをセットすることで受信回路をイネーブル状態とし、スタート・ビットの待ち受けを開始します。

文字が受信されると、

- RXNE ビットがセットされます。このビットはシフト・レジスタの内容が RDR レジスタに移されたことを示しています。すなわち、データが受信され、読出しが可能であることを示しています。(同時に、この受信のエラー・フラグも読出しができます。)
- RXNEIE ビットがセットされていれば、割込みリクエストが発生します。
- 受信時にフレーム・エラー、ノイズ・エラー、オーバ・ラン・エラーが検出されると、エラー・フラグがセットされます。
- マルチバッファ動作では、RXNE ビットはバイトが受信されるごとにセットされ、DMA がデータ・レジスタを読み出すごとにクリアされます。
- シングルバッファモードでは、RXNE ビットはソフトウェアで USART_DR レジスタを読み出すことでクリアされます。RXNE ビットは 0 を書き込むことでクリアすることもできます。オーバ・ラン・エラーを生じないためには、次の文字の受信が完了する前に、RXNE ビットがクリアされていなければなりません。

注意： RE ビットはデータを受信中にクリアしてはいけません。受信動作中に RE ビットがクリアされると、このバイトの受信動作はアボートされます。

ブレーク・フレーム

ブレーク・フレームが受信されると、USART はこれをフレーミング・エラーの発生とみなします。

アイドル・フレーム

アイドル・フレームが検出されると、データ受信と同じ動作が生じ、IDLEIE ビットがセットされていれば割込みリクエストが発生します。

オーバ・ラン・エラー

RXNE ビットがクリアされていない状態で文字が受信されると、オーバ・ラン・エラーが発生します。RXNE ビットがクリアされるまで、データがシフト・レジスタから RDR レジスタに移されることはありません。

RXNE フラグはバイトが受信されるごとにセットされます。次のデータが受信されたときに RXNE フラグがクリアされていないとき、もしくは DMA リクエストが実行されていないときにはオーバ・ラン・エラーが発生します。オーバ・ラン・エラーが発生すると、

- ORE ビットがセットされます。
- RDR の内容は失われます。USART_DR の読み出しが行われたときには、前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。これにより、オーバラン中に受信されたデータは失われます。
- RXNEIE ビットがセットされているか、もしくは EIE ビットと DMAR ビットの両方がセットされた時には、割込みリクエストが発生します。
- ORE ビットは、USART_SR レジスタを読み出し、次に USART_DR レジスタを読み出す操作でクリアされます。

注意 : *ORE* ビットがセットされたときには、少なくとも 1 個のデータが失われています。この際、次の 2 通りのケースがあります。

- RXNE=1 の場合には、有効な最後のデータは RDR レジスタに格納されており、これを読み出すことができます。
- RXNE=0 の場合には、有効な最後のデータはすでに読み出されており、RDR レジスタに読み出すべきデータがないことを示しています。このケースは RDR からの有効なデータの読み出しと、新たなデータの受信が同時に発生した時に生じます。また、読み出し操作中 (USART_SR レジスタと USART_DR レジスタの読み出しの間) に新たなデータが受信された場合にも発生します。

ノイズ・エラー

有効な受信データとノイズを区別しデータを復元するため、多重サンプリング手法が採られています。

図 240. ノイズ検出のためのデータサンプリング



表 151. サンプルデータからのノイズ検出

サンプルされた値	NE ステータス	受信ビットの値	データの有効性
000	0	0	有効
001	1	0	無効
010	1	0	無効
011	1	1	無効
100	1	0	無効
101	1	1	無効
110	1	1	無効
111	0	1	有効

フレーム内でノイズが検出された時には :

- RXNE ビットの立上りエッジで、NE ビットがセットされます。
- 誤りを含んだデータがシフト・レジスタから USART_DR レジスタへ移されます。
- バイト単位通信の時には割込みリクエストは発生しませんが、同時にセットされる RXNE ビットは割込みリクエストを発生します。マルチバッファ通信動作では、USART_CR3 レジスタの EIE ビットがセットされていれば、割込みリクエストが発生します。

NE ビットは、USART_SR レジスタが読み出され、続いて USART_DR レジスタが読み出された時にクリアされます。

フレーミング・エラー

受信時に同期はずれや過度のノイズのため、所定の位置で STOP ビットが検出されなかつたときには、フレーミング・エラーとみなされます。

フレーミング・エラーが検出されたときには、

- ハードウェアにより FE ビットがセットされます。
- 誤りを含んでいると思われるデータがシフト・レジスタから USART_DR レジスタに移されます。
- バイト単位通信の時には割込みリクエストは発生しませが、同時にセットされる RXNE ビットは割込みリクエストを発生します。マルチバッファ通信動作では USART_CR3 レジスタの EIE ビットがセットされていれば、割込みリクエストが発生します。

FE ビットは、USART_SR レジスタが読み出され、続いて USART_DR レジスタが読み出されたときにクリアされます。

受信時の STOP ビット数

受信 STOP ビット数は制御レジスタ 2 を通して設定できます – STOP ビット数は通常の動作では 1 もしくは 2 個ですが、スマートカード・モードでは 0.5 及び 1.5 が使用されます。

1. 0.5 STOP ビット（スマートカード・モードの受信時）：0.5 STOP ビットの時間には、サンプリングは行われません。0.5 STOP ビットが選択されているときには、フレーミング・エラーやブレーク・フレーム検出の条件は発生しません。
2. 1 STOP ビット : 8 番目、9 番目、10 番目のいずれかとして、1 STOP ビットのサンプリングが行われます。
3. 1.5 STOP ビット（スマートカード・モードの送信時）：スマートカード・モードの送信では、データが正しく送出されたかをデバイスがチェックしなければなりません。このため、受信回路がイネーブル状態（USART_CR1 レジスタの RE=1）とされ、スマートカードがパリティ・エラーを検出したか否かを確認するため、STOP ビットがチェックされます。パリティ・エラーの発生時には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 - NAK 信号 - とし、フレーミング・エラーを通知します。これにより FE フラグが 1.5 STOP ビットの終わりで RXNE ビットとともにセットされます。1.5 STOP ビットのサンプリングは 16 番目、17 番目ないしは 18 番目のサンプル（STOP ビット開始後の 1 ポークロック時間後）で行われます。1.5 STOP ビットは 2 部分に分けられます。最初の 0.5 ポークロックでは何も行われません。次の 1 個の通常の STOP ビット期間では、ほぼその中間でサンプリングが行われます。詳細は [セクション 24.3.10: スマートカード \(658 ページ\)](#) をお読みください。
4. 2 STOP ビット : 2 STOP ビットのサンプリングは、8 番目、9 番目もしくは 10 番目のサンプリング時に、最初の STOP ビットがサンプルされます。最初の STOP ビットでフレーミング・エラーが検出されたときには、フレーミング・エラー・フラグがセットされます。2 番目の STOP ビットはフレーミング・エラーの検査対象とはなりません。RXNE フラグは最初の STOP ビットの終わりでセットされます。

24.3.4 ボーレート生成回路

受信回路と送信回路のボーレートは同じ値 USARTDIV がセットされます。設定値は USART_DIV レジスタに整数部と小数部とに分けて設定されます。

$$\text{Tx/Rx baud} = \frac{f_{CK}}{(16 * \text{USARTDIV})}$$

凡例: $f_{PCLKx(x=1,2)}$ - ペリフェラルへの入力クロック (USART2, 3, 4, 5 用に PCLK1 または USART1 用に PCLK)

USARTDIV は符号のない固定小数点の値として扱われ、USBART_BRR レジスタに設定されます。

BRR レジスタから USARTDIV を求めるには :

例 1:

DIV_Mantissa = 27d かつ DIV_Fraction = 12d (BRR=1BCh) のとき、

仮数として (USARTDIV) = 27d

割合は (USARTDIV) = 12/16 = 0.75d

そのため USARTDIV = 27.75d になります。

例 2:

USARTDIV = 25.62d を設定するには、

DIV_Fraction = 16*0.62d = 9.92d、最も近い整数値は 10d = Ah

DIV_Mantissa = 整数部 (25.620d) = 25d = 19h

したがって、設定値 BRR = 19Ah になります。

例 3:

USARTDIV = 50.99d を設定するには、

DIV_Fraction = 16*0.99d = 15.84d => 最も近い整数値は 16d = 10h

DIV_Mantissa = 整数部 (50.990d) = 50d = 32h

注意 : USART_BRR に書き込みが行われた後、ボーレート・レジスタの新しい値がボーレートカウンタにロードされます。ボーレートカウンタの値は動作中 (トランザクションの間) に変更することはできません。

参考資料

表 152. ポーレート設定と周波数の誤差

ポーレート		$f_{CPU} = 36\text{MHz}$			$f_{CPU} = 72\text{MHz}$		
S.No	Kbps 単位	設定される値	ポーレートレジスタ設定値	誤差 (%) = (設定値 - 目標値) / 目標値	設定される値	ポーレートレジスタ設定値	誤差 (%)
1.	2.4	2.400	937.5	0%	2.4	1875	0%
2.	9.6	9.600	234.375	0%	9.6	468.75	0%
3.	19.2	19.2	117.1875	0%	19.2	234.375	0%
4.	57.6	57.6	39.0625	0%	57.6	78.125	0%
5.	115.2	115.384	19.5	0.15%	115.2	39.0625	0%
6.	230.4	230.769	9.75	0.16%	230.769	19.5	0.16%
7.	460.8	461.538	4.875	0.16%	461.538	9.75	0.16%
8.	921.6	923.076	2.4375	0.16%	923.076	4.875	0.16%
9.	2250	2250	1	0%	2250	2	0%
10.	4600	NA	NA	NA%	45005	1	0%

- 注意： 1 CPU クロック周波数を低下させると、ポーレートの精度が低下します。ポーレートの最大周波数はこのデータから得られます。
2 USART1だけがPCLK2 (72 MHz Max) の供給を受けます。他の USART は PCLK1 (36 MHz Max) が供給されます。

24.3.5 複数プロセッサ間の通信

ネットワークに複数の USART を接続するという方法で、複数のプロセッサ間の通信を USART で行うことができます。たとえば、ある 1 個の USART をマスターとして、その TX 出力を他の USART の RX 入力と接続することが考えられます。マスター以外の USART はスレーブとして働き、その TX 出力の論理積をとった上でマスターの RX 入力に接続します。

マルチプロセッサ通信では、メッセージの送信先のみがメッセージ全体を受信することが望ましい場合が多く、これにより対象外の USART のオーバヘッドを軽減することができます。

アドレスの対象外となったデバイスは、ミュート機能でミュートモードにおかれます。ミュートモードでは、

- 受信ステータス・ビットはセットされません。
- 受信関連の割込みリクエストは禁止されます。
- USART_CR1 レジスタの RWU ビットがセットされます。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、状況に応じてソフトウェアが書込むこともできます。

USART は、USART_CR1 レジスタの WAKE ビットの設定に応じて、次の 2 通りのいずれかでミュートモードの出入りを制御することができます。

- WAKE ビットがクリアされているとき - アイドル・ライン検出
- WAKE ビットがセットされているとき - アドレスマーク検出

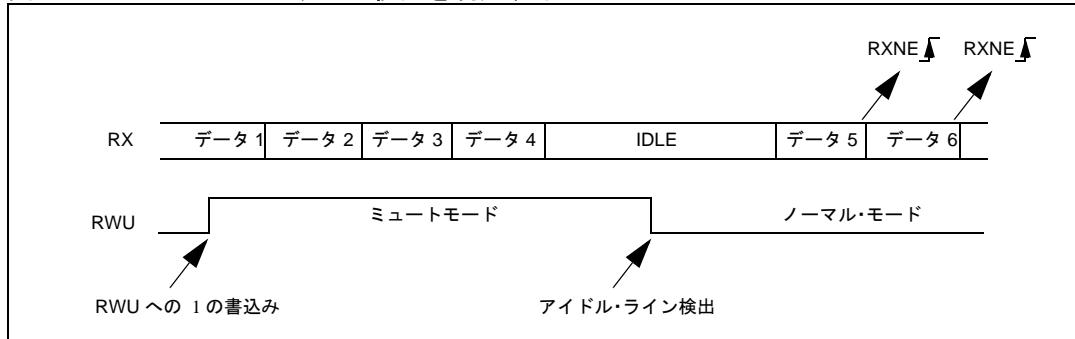
アイドル・ライン検出 (WAKE=0)

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・フレームの検出でミュートモードから抜出します。このとき RWU ビットはハードウェアによりクリアされますが、USART_SR レジスタの IDLE ビットはセットされません。RWU ビットは、ソフトウェアで 0 を書込むこともできます。

アイドル・ライン検出を利用するミュートモードの動作は図 241 に示されています。

図 241. アイドル・ライン検出を利用するミュートモード



アドレスマーク検出 (WAKE=1)

このモードでは、MSB が 1 のバイトがアドレス、0 のバイトがデータとして認識されます。アドレスバイトでは、送り先のアドレスが下位 4 ビットに置かれます。受信側では、この 4 ビットは USART_CR2 レジスタの ADD フィールドに設定されている自己のアドレスと比較されます。

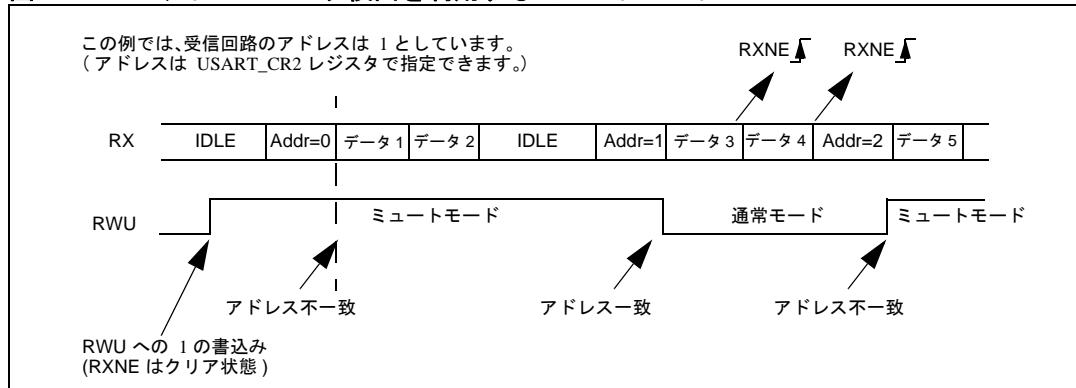
USART はアドレスバイトが設定されている自己のアドレスと一致しないときにミュートモードに入ります。USART がミュートモードに入るため、このアドレスバイトに対しては RXNE フラグはセットされず、割込みや DMA リクエストは発せられません。

設定されている自己アドレスと一致するアドレスバイトが受信されると、USART はミュートモードを終了します。RWU ビットはクリアされ、続くバイトは通常どおりに受信されます。このとき、RWU ビットがクリアされますので、アドレスバイトに対して RXNE ビットがセットされます。

受信バッファにデータが存在しない (USART_SR の RXNE=0) とき、RWU ビットには 0 や 1 を書込むことができます。受信バッファが空でないときには、書き込みは無視されます。

アドレスマークを利用したミュートモードの動作は図 242 に示されています。

図 242. アドレスマーク検出を利用するミュートモード



24.3.6 パリティ制御

USART_CR1 レジスタの PCE ビットをセットすることで、パリティ制御（送信時のパリティビット生成と受信時のパリティ検査）を有効とすることができます。M ビットで指定されている語長に応じて、表 153 にあるフレーム形式が指定できます。

表 153. フレーム・フォーマット

M ビット	PCE ビット	USART フレーム
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

略語 : SB: スタート・ビット、STB: STOP ビット、PB: パリティビット

注意 : アドレスマークでウェークアップを行う場合、データの最上位ビットはこの目的のために使用され、パリティビットとはなりません。

偶数パリティ : 下位 7 もしくは 8bit とパリティビットからなるフレーム内の 1 のビットが偶数になるよう、パリティビットがセットされます。

例 : data=00110101、4bit が値 1 => 偶数パリティが選択されている (USART_CR1 レジスタの PS=0) ときには、パリティビットは 0 になります。

奇数パリティ : 下位 7 もしくは 8bit とパリティビットからなるフレーム内の 1 のビットが奇数になるよう、パリティビットがセットされます。

例 : data=00110101、4bit が値 1 => 奇数パリティが選択されている (USART_CR1 レジスタの PS=1) ときには、パリティビットは 1 になります。

送信モード: USART_CR1 の PCE ビットがセットされていると、データ・レジスタに書込まれた最上位ビットは、パリティビットに置換えられた上で送信されます。パリティ検査で誤りがあると、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には、割込みリクエストが発生します。

24.3.7 LIN (local interconnection network) モード

LIN モードは USART_CR2 レジスタの LINEN ビットをセットすると設定されます。LIN モードでは次のビットはクリアされた状態としておかねばなりません。

- USART_CR2 レジスタの CLKEN ビット
- USART_CR3 レジスタの STOP[1:0]、SCEN、HDSEL 及び IREN ビット

LIN 送信

LIN マスターの送信では、[セクション 24.3.2](#) で記述されている手順と同じ操作が使用されますが、通常の USART 送信とは次の点が異なります。

- M ビットをクリアし、語長を 8bit とします。
- LIN モードに入るには LINEN ビットをセットします。このとき、SBK ビットをセットすると 13 個の 0bit からなるブレーク・フレームが送出されます。この後、1 の値のビットが送出され、次のスタート・ビット検出を開始します。

LIN 受信

LIN モードがイネーブルとされたとき、ブレーク検出回路が動作状態となります。この回路は通常の USART 受信回路とは独立しています。ブレークは、アイドルステートや通常のデータフレームなど、ラインの状態にかかわらず検出されます。

受信回路がイネーブルとなる (USART_CR1 レジスタの RE=1) と、RX 入力上のスタート・ビットを待ち受けます。スタート・ビットの検出方法は、ブレーク・フレームやデータと同じです。スタート・ビットが検出されると、通常のデータの場合と同じく、続くビットがサンプルされます。(8、9、10 番目のサンプル) 10 個 (USART_CR2 レジスタの LBDL=0) もしくは 11 個 (USART_CR2 レジスタの LBDL=1) の連続した 0 とこの後の区切り文字が続くと、USART_SR レジスタの LBD ビットがセットされます。このとき、LBDIE ビットが 1 であれば割込みリクエストが発生します。ブレークが有効となる前に、RX ラインがハイレベルに戻ったことが確認されると、ブレーク検出は有効となります。

10 もしくは 11 個の 1 がサンプルされると、ブレーク検出動作はキャンセルされ、スタート・ビットの待ち受けが開始されます。

LIN モードがディセーブル状態 (LINEN=0) のときには、受信回路は通常の USART として動作し、ブレーク検出は行われません。

LIN モードがイネーブル状態 (LINEN=1) のときにフレーム・エラーが発生 (STOP ビットの位置で 0 を検出、ブレーク・フレームの場合にも発生します。) すると、ブレーク検出回路が他の値 1 のビットを受け取るまで受信回路は停止します。

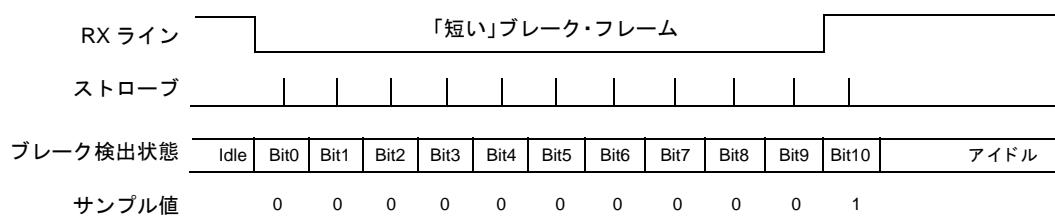
ブレーク検出回路の動作とブレーク・フラグの状態は[図 243: LIN モードにおけるブレーク検出 \(11 ビットブレーク長 \(LBDL=1\)\) \(654 ページ\)](#) に示されています。

ブレーク・フレームの例は[図 244: LIN モードにおけるブレーク検出とフレーミング・エラー検出 \(655 ページ\)](#) にあります。

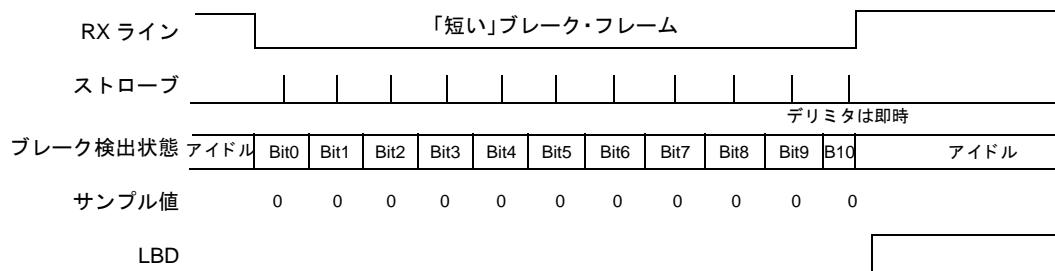
参考資料

図 243. LIN モードにおけるブレーク検出 (11 ビットブレーク長 (LBDL=1))

Case 1: 不十分な長さのブレーク・フレーム => ブレークはアポートされ、LBD はセットされません。



Case 2: 不十分な長さのブレーク・フレーム => ブレークが検出され、LBD がセットされます。



Case 3: 十分な長さのブレーク・フレーム => ブレークが検出され、LBD がセットされます。

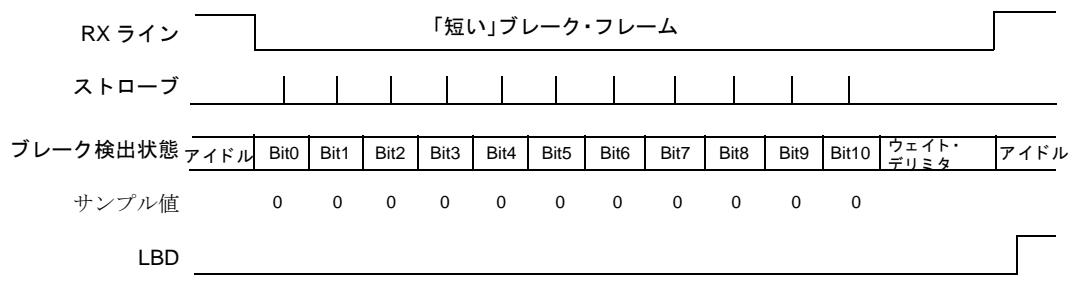
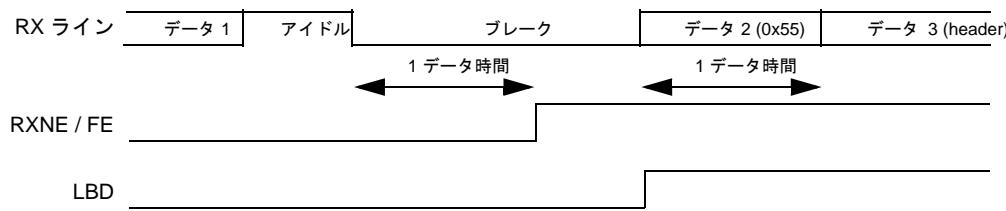


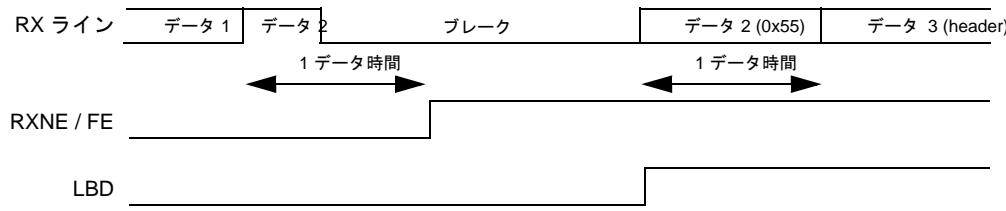
図 244. LIN モードにおけるブレーク検出とフレーミング・エラー検出

この例では、LBDL=1 (11bit ブレーク長)、M=0 (8bit データ長) が想定されています。

Case 1: アイドル後のブレーク発生



Case 1: データ受信中のブレーク発生



24.3.8 USART 同期シリアル・モード

同期シリアル・モードは USART_CR2 レジスタの CLKEN ビットに 1 を書込むことで指定されます。同期シリアル・モードでは次のビットはクリアされた状態でなければなりません。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの SCEN、HDSEL 及び IREN ビット

USART は、両方向の同期シリアル通信のマスター・モードとして、利用することができます。SCLK ピンは USART 送信クロックの出力となります。スタート・ビット、STOP ビットに対しては、SCLK ピンからクロックパルスが送出されません。USART_CR2 レジスタの LBCL ビットの状態によっては、最後の有効データビット（アドレスマーク）に対してクロックを送出しないことも可能です。USART_CR2 レジスタの CPOL ビットで外部クロックの極性を、CPHA ビットではクロックの位相を選択することができます。（図 245、図 246 及び 図 247 を参照）

アイドル状態、プリアンブルとブレーク送出では、外部 SCLK クロックは送出されません。

同期シリアル・モードでは USART 送信回路は通常の調歩式とまったく同じように動作しますが、TX からデータが、SCLK から TX に同期したクロックが出力されます。

このモードでは USART 受信回路は調歩式と異なった動きをします。RE=1 であれば、データは多重サンプリングではなく、SCLK (CPOL、CPHA で指定される立上り、立下りエッジ) によってサンプルされます。このとき、ボーレートに応じた 1/16 ビットタイムのセットアップ及びホールドタイムが必要とされます。

参考資料

- 注意： 1 *SCLK* ピンは *TX* ピンとともに動作します。クロックは送信回路がイネーブル状態 (*TE=1*) の時にのみ出力され、*USART_DR* に書込まれたデータが送信されます。データ送信を行わずに同期シリアルデータを受信することはできません。
- 2 クロックパルスが正常に機能するには、*LBCL*、*CPOL* 及び *CPHA* ビットの設定は送受信回路がともにディセーブル状態の時に限られます。これらのビットは少なくとも送信・受信回路が動作状態の時に変更することはできません。
- 3 受信回路のセットアップとホールドタイムを最小とするため、*TE* ビットと *RE* ビットを同じ 1 個の命令で *I* セットすることをお勧めします。
- 4 USART はマスタ・モードのみをサポートしています。*SCLK* は常に出力で、外部からのクロックを受けて送受信を行うことはできません。

図 245. USART 同期シリアル送信の例

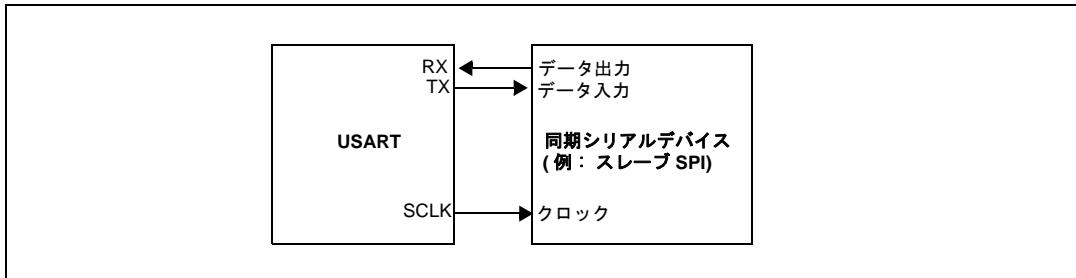
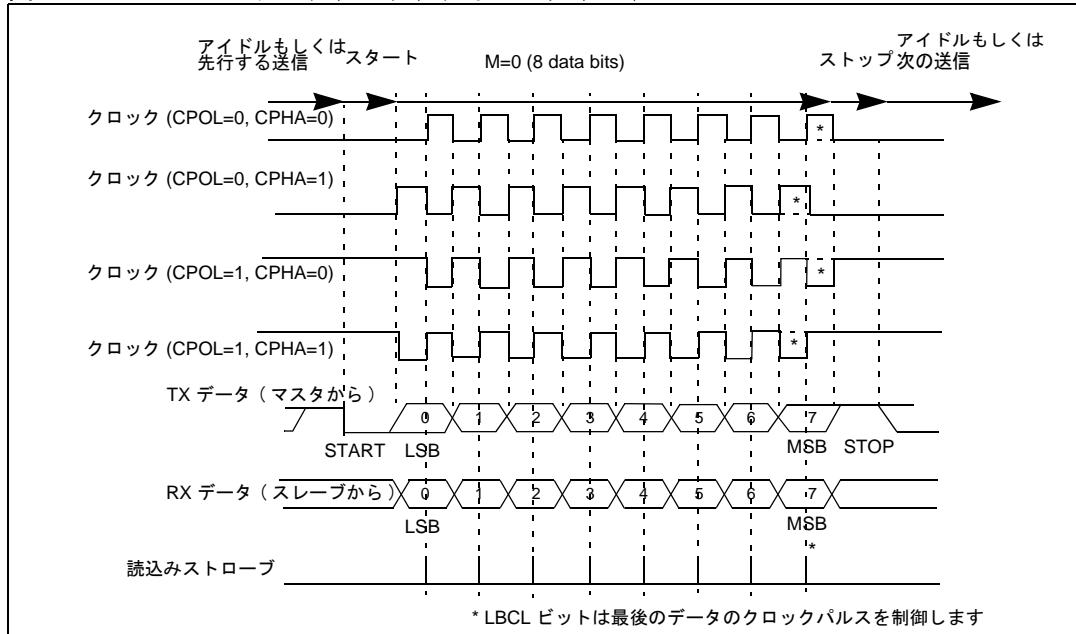


図 246. USART データクロックタイミング (M=0)



参考資料

図 247. USART データクロックタイミング (M=1)

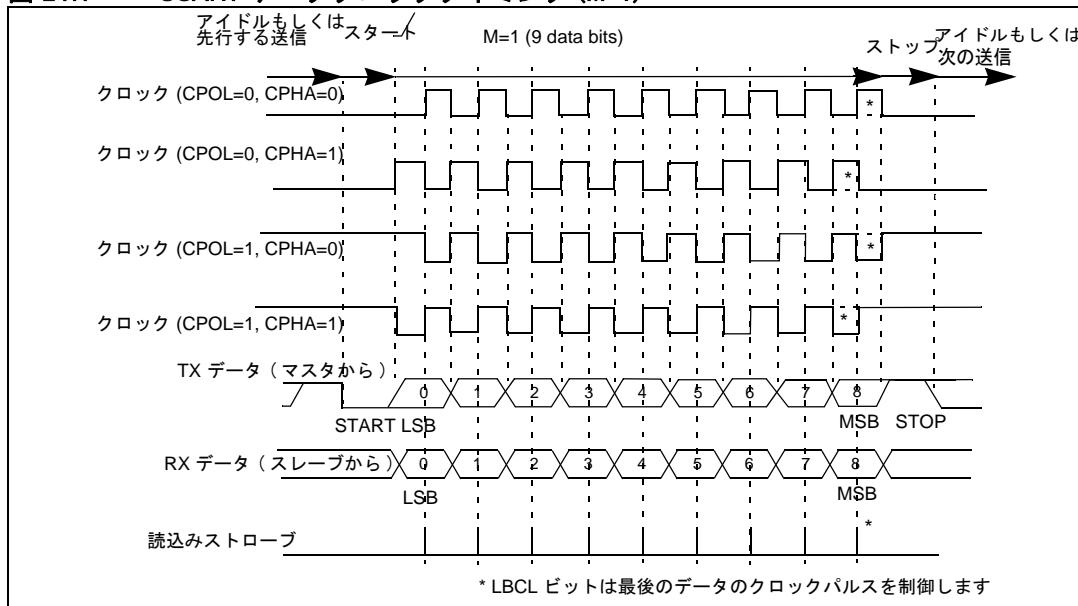
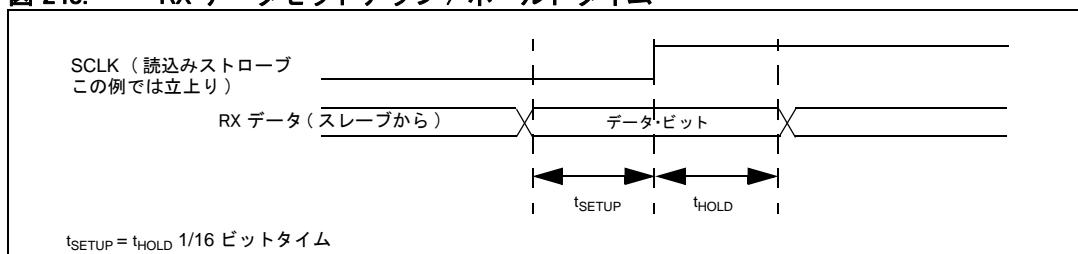


図 248. RX データセットアップ / ホールドタイム



注意: SCLK の機能はスマートカード・モードとは異なっています。詳細はスマートカード・モードの章を参照してください。

24.3.9 単線式半二重通信

単線半二重モードは USART_CR3 レジスタの HDSEL ビットで選択されます。このモードでは、次のビットはクリアされた状態でなければなりません。

- USART_CR2 レジスタの LINEN 及び CLKEN ビット
- USART_CR3 レジスタの SCEN 及び IREN ビット

USART は単線半二重のプロトコルに従うよう、設定することができます。半二重通信と全二重通信の選択は USART_CR3 レジスタの HDSEL 制御ビットで行います。

HDSEL ビットに 1 が書込まれると、

- RX は使用されません。
- TX はデータが送信されないときには常に解放されます。つまり、アイドルもしくは受信状態のときには TX ピンは標準入出力として動作します。したがって、TX が USART に駆動されていないときにはフローティング入力（もしくはオープンドレン出力）となるよう、入出力ポートとして設定しておかねばなりません。

これとは別に、通信自体は通常の USART モードと同様の通信が行われます。ラインの競合は（中央にアービタを置くなど）ソフトウェアで管理しなければなりません。特に、TE

ビットがセットされている限り、送信動作がハードウェアで停止させられることはなく、データがデータ・レジスタに書込まれる限り続けます。

24.3.10 スマートカード

スマートカード・モードは USART_CR3 レジスタの SCEN ビットで指定されます。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を保たねばなりません。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの HDSEL 及び IREN ビット

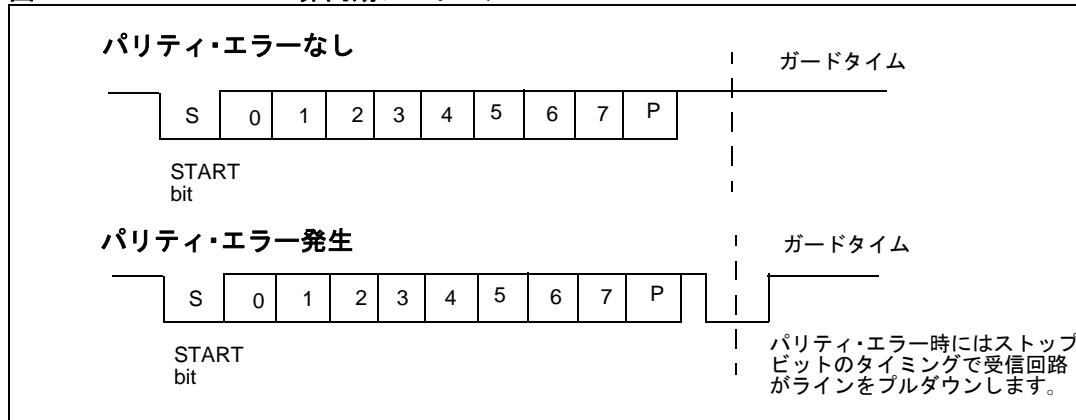
さらに、スマートカードにクロックを供給するために CLKEN ビットをセットすることができます。

スマートカード・インターフェースは ISO7816-3 規格で定められている非同期プロトコル・スマートカードをサポートするよう設計されています。USART は次のように設定されねばなりません。

- USART_CR1 レジスタの M=1、PCE=1 で、語長 8bit、パリティありを指定し、さらに次のいずれかを指定します。
 - 受信では、USART_CR2 レジスタの STOP=01b とし、0.5 STOP ビットを選択します。
 - 送信では、USART_CR2 レジスタの STOP=11b とし、1.5 STOP ビットを選択します。

図 249 はパリティ・エラーの有無によりデータラインがどのように観察されるかを図示しています。

図 249. 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き、USART の TX 出力はスマートカードの出力ともなる両方向ラインを駆動します。このためには、製品レベルで、SW_RX は TX と同じ入出力に接続されていることが必要です。TX_EN はスタート・ビットとデータの送出時にアサートされて送信回路の出力が有効となり、STOP ビット時にはアサートされ出力は弱いプルアップ状態となります。このため、パリティ・エラー時に受信回路がラインをプルダウンすることができます。TX_EN が使用されないときには、STOP ビット時に TX はハイレベルとなります。TX がオープンドレーンに設定されている場合に限り、受信回路がこのラインをドライブできます。

スマートカードは単線、半二重通信のプロトコルを使用します。

- 送信シフト・レジスタからのデータの送信は少なくとも 1/2 ボークロック遅れていることが保証されています。通常の USART 動作ではシフト・レジスタからの送出は次

参考資料

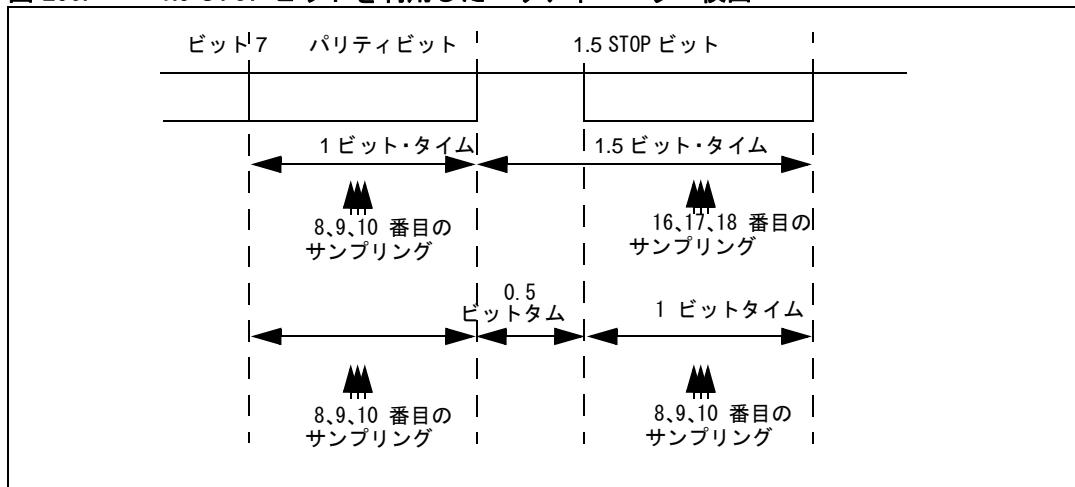
のポークロックのエッジで開始されます。スマートカード・モードでは、この送出動作はさらに 1/2 ポークロック分だけ遅くなります。

- 1/2 STOP ビットに設定されたフレームの受信時にパリティ・エラーが検出されると、受信フレームが終わった後（1/2 STOP ビットの終了時）に 1 ポークロックの間、送信ラインがローレベルにプルダウンされます。これが、USART に送られてき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示すことになります。この、送信ラインを 1 ポークロックの間プルダウンするという NACK 信号は 1.5 STOP ビットに設定されている送信側でフレーミング・エラーを生じます。アプリケーションはプロトコルに従ったデータ再送信を実行することができます。NACK 制御ビットがセットされていれば、パリティ・エラーは受信回路の否定応答を生じます。NACK ビットがセットされていなければ、否定応答は送信されません。
- ガードタイムレジスタの設定により、TC フラグのセットは遅れます。通常 TC フラグはシフト・レジスタが空になり、他に送信するデータが無いときにセットされます。スマートカード・モードではシフト・レジスタが空になるとガードタイムカウントが、ガードタイムレジスタに設定されている値までカウントアップし、このカウントアップの間は TC フラグ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トが設定値に達すると、TC フラグがセットされます。
- スマートカード・モードでは、TC フラグのクリアは行われません。
- 送信終了時に受信回路からの NACK 似よりフレーミング・エラーが発生した時、送信側の受信回路で NACK がスタート・ビットと認識されることはありません。ISO プロトコルによれば、受信される NACK 信号の幅は 1 もしくは 2 ポークロック時間と定められています。
- 受信側では、パリティ・エラーが検出され NACK が送出されたとき、この NACK が受信回路でスタート・ビットとして扱われることはありません。

- 注意： 1 スマートカード・モードではブレーク・フレームは重要ではありません。フレーミング・エラー発生時の値 $00h$ のデータは、ブレークではなくデータとして処理されます。
2 TE ビットを反転したとき、IDLE フレームは送出されません。ISO プロトコルではアイドル・フレームは定義されていません。

NACK 信号が USART でどのようにサンプルされるかの詳細が図 250 に示されています。この例では USART がデータと 1.5 STOP ビットを送出しています。USART の受信回路はデータの内容の確認と NACK 信号のチェックのために動作状態となっています。

図 250. 1.5 STOP ビットを利用したパリティ・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することができます。スマートカード・モードでは SCLK は通信動作と関連しておらず、単に内部のペリフェラル入力クロックを 5 ビットのプリスケーラに通した信号が出力されます。この分周比はプリスケーラ・レジスタ USART_GTPR で指定されます。ペリフェラル入力クロックの周波数を f_{CK} としたとき、SCLK の周波数は $f_{CK}/2$ から $f_{CK}/62$ の間で選択することができます。

24.3.11 IrDA SIR ENDEC ブロック

IrDA モードは USART_CR3 レジスタの IREN ビットをセットすることで指定されます。IrDA モードでは次のビットはクリア状態が保たれねばなりません。

- USART_CR2 レジスタの LINEN、STOP 及び CLKEN ビット
- USART_CR3 レジスタの SCEN 及び HDSEL ビット

IrDA SIR 物理層仕様論理は、値 0 を赤外パルスとした RZI (Return to Zero, Inverted) 変調方式を採用しています。(図 251 を参照)

SIR 送信エンコーダは USART 出力を NRZ (Non Return to Zero) 送信ビット波形に変調します。出力パルス列は外部の出力ドライバから赤外 LED に送られます。SIR ENDEC では USART は最高 115.2 kbps のビット列のみをサポートしています。ノーマル・モードでは、送信されるパルスの幅はビット時間の 3/16 と定められています。

SIR 受信デコーダは赤外検出素子からの RZ (Return to Zero) ビット列の復調を行い、NRZ シリアルビット列を USART に出力します。デコーダの入力はアイドル状態でノーマルハイ（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となっています。デコーダ入力のローレベルで、スタート・ビットが検出されます。

- IrDA は半二重の通信プロトコルです。送信回路が動作中 (USART が IrDA エンコーダに送信データを送っている間) は、IrDA 受信ライン上のデータは IrDA デコーダで無視され、受信回路が動作中 (USART がデコードされたデータを受信している間) は TX 上の USART から IrDA へのデータは IrDA でエンコードされません。データ受信中は送信データが破壊される場合があるため、送信は避けねばなりません。
- 論理値 0 がパルスとして送信され、値 1 がパルスなしとなります。ノーマル・モードにおけるパルスの幅は、選択されているビット時間の 3/16 と規定されています。(図 252 を参照)
- SIR デコーダは IrDA に準じた受信信号を USART のためのビット列に変換します。
- SIR 受信ロジックはハイ状態を論理値 1 に、ロー状態パルスを論理値 0 に変換します。
- 送信エンコーダ出力はデコーダ入力と逆の極性をもっています。SIR 出力はアイドル時にロー状態となります。
- IrDA 仕様は $1.41 \mu s$ より大きなパルスを受け付けることをリクエストしています。受け入れられるパルスの幅はプログラムで指定できます。受信側のグリッヂ検出ロジックは 2 PCS 時間 (PCS は IrDA 低電力モード・ボーレート・レジスタ USART_GTPR に設定されるプリスケーラの値です。) より狭いパルスをフィルタで除いています。1 PSC より狭いパルスは常に除かれますが、幅が 1 から 2 の間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。IrDA エンコーダ / デコーダは PSC=0 では動作しません。
- 受信回路は低電力送信回路と通信が可能です
- IrDA モードでは USART_CR2 レジスタの STOP ビットは、1 STOP ビットが選択されなければなりません。

IrDA 低電力モード

送信回路:

低電力モードではパルスの幅はビット時間の 3/16 を守っていません。これに代わって、パルス幅は低電力時のボーレート（最小で 1.42 MHz）周期の 3 倍となっています。一般的な値は 1.8432 MHz ($1.42 \text{ MHz} < \text{PSC} < 2.12 \text{ MHz}$) です。低電力モードの分周回路では、この条件を満たすようシステム・クロックを分周しています。

受信回路:

低電力モードの受信は、ノーマル・モードの受信と変わることはありません。グリッヂ検出では USART は 1/PSC より狭いパルスを無視しなければなりません。有効なローレベルはパルス幅が IrDA 低電力ボーレートクロック周期 (PSC) の 2 倍以上のもののみを受け入れます。

- 注意：
- 1 1 PSC 時間から 2 PSC 時間の間の幅のパルスは、無視されることも、無視されないこともあります。
 - 2 受信回路のセットアップタイムはソフトウェアで管理されねばなりません。IrDA 物理層仕様では送信と受信の間に最小 10 ms の時間を置くことになっています。(IrDA は半二重のプロトコルです。)

図 251. IrDA SIR ENDEC - ブロック図

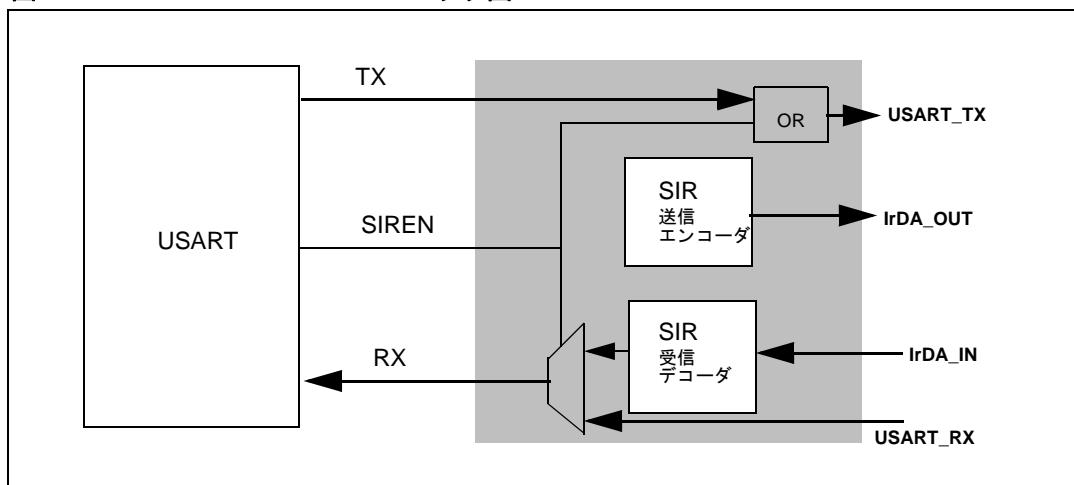
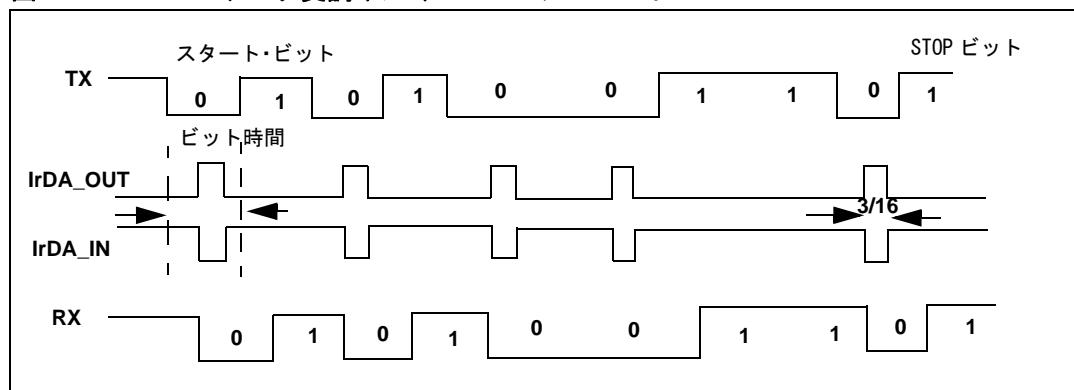


図 252. IrDA データ変調 (3/16) - ノーマル・モード



24.3.12 DMA を使用する連続的な通信

連続的に通信を継続するため、USART は DMA を利用することができます。受信バッファに対する DMA リクエストと送信バッファに対する DMA リクエストは個別に発生させることができます。

注意： DMA コントローラが使用できるか否かについては、個々の製品の仕様書を参照してください。目的の製品で DMA の利用が不可能であれば、USART はセクション 24.3.2 及び 24.3.3 で説明されているように使用しなければなりません。連続した通信を行うには、USART2_SR レジスタの TXE/ RXNE フラグをクリアしなければなりません。

DMA を利用した送信動作

DMA モードによる送信動作は USART_CR3 レジスタの DMAT ビットをセットすることで、実現できます。データは TXE ビットがセットされるごとに、DMA ペリフェラルで設定されている SRAM ドメインから USART_DR レジスタにロードされます。DMA チャネルを USART 送信に割当てるには、次の手順を実行します。(ここで x はチャネル番号を意味します。)

1. USART_DR レジスタのアドレスを、転送の受け側となるよう DMA 制御レジスタに書き込みます。データは TXE イベントごとにメモリからこのアドレスに移されます。
2. メモリ・アドレスを、転送の送り側となるよう DMA 制御レジスタに書き込みます。データは TXE イベントごとに、このメモリ・ドメインから USART_DR レジスタにロードされます。
3. DMA 制御レジスタに、転送すべきバイト数を設定します。
4. DMA レジスタにチャネルの優先順位を設定します。
5. アプリケーションで必要とされる 1/2 転送、転送完了時の DMA 割込み生成を設定します。
6. DMA レジスタでチャネルをアクティブにします。

DMA コントローラに設定された転送データ数に達したとき、DMA コントローラは、DMA チャネルの割込みベクタに対する割込みリクエストを発生します。割込み処理ルーチンの中では、USART_CR3 レジスタの DMAT ビットをソフトウェアでクリアしなければなりません。

注意： 送信に DMA を使用するときには、TXIE ビットをイネーブル（割込みリクエスト許可）としてはいけません。

DMA を利用した受信動作

DMA モードによる受信動作は USART_CR3 レジスタの DMAR ビットをセットすることで実現できます。データは データバイトが受信されるごとに、USART_DR レジスタから DMA ペリフェラルで設定されている SRAM ドメインに格納されます。DMA チャネルを USART 受信に割当てるには、次の手順を実行します。(ここで x はチャネル番号を意味します。)

1. USART_DR レジスタのアドレスを、転送の送り側となるよう DMA 制御レジスタに書込みます。データは RXNE イベントごとにこのアドレスからメモリに移されます。
2. メモリ・アドレスを、転送の受け側となるよう DMA 制御レジスタに書込みます。データは RXNE イベントごとに、USART_DR レジスタからこのメモリ・ドメインに格納されます。
3. DMA 制御レジスタに、転送すべきバイト数を設定します。
4. DMA レジスタにチャネルの優先順位を設定します。
5. アプリケーションで必要とされる 1/2 転送、転送完了時の DMA 割込み生成を設定します。
6. DMA レジスタでチャネルをアクティブにします。

DMA コントローラに設定された転送データ数に達したとき、DMA コントローラは、DMA チャネルの割込みベクタに対する割込みリクエストを発生します。割込み処理ルーチンの中では、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。

注意：受信に DMA を使用するときには、RXEIE ビットをイネーブル（割込みリクエスト許可）としてはいけません。

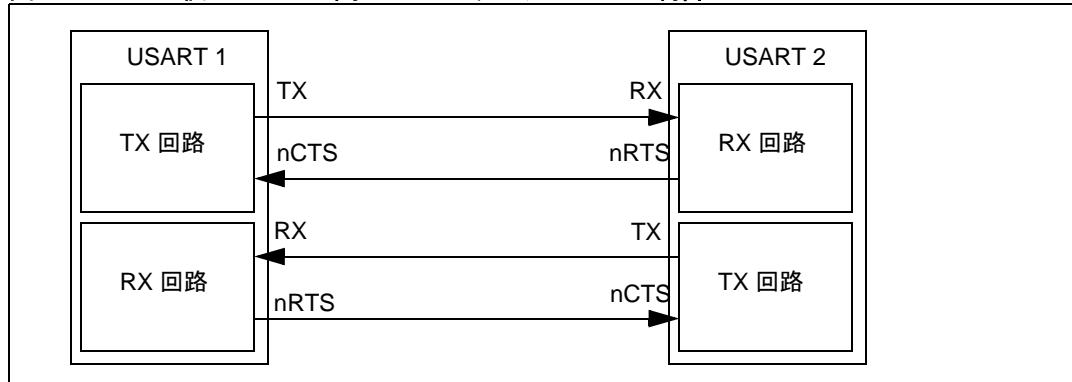
マルチバッファ通信における割込みリクエストとエラー・フラグ

マルチバッファ通信では、トランザクションでエラーが発生すると、そのバイトの後にエラー・フラグがセットされます。割込みリクエストの許可ビットがセットされていれば、このとき割込みリクエストが発生します。フレーミング・エラー、オーバ・ラン・エラー、及びノイズ・フラグは、バイト単位の転送では RXNE と同時にセットされますが、マルチバッファ通信では別の割込みリクエスト許可ビット（USART_CR3 レジスタの EIE ビット）が用意されており、セットされていれば、対象バイトの後に割込みリクエストが発生します。

24.3.13 ハードウェア・フロー制御

2 個のデバイス間のシリアルデータフローの制御を、nCTS 入力と nRTS 出力を利用して行なうことができます。[図 253](#) には、このときのデバイス間の接続が図示されています。

図 253. 2 個の USART 間のハードウェア・フロー制御

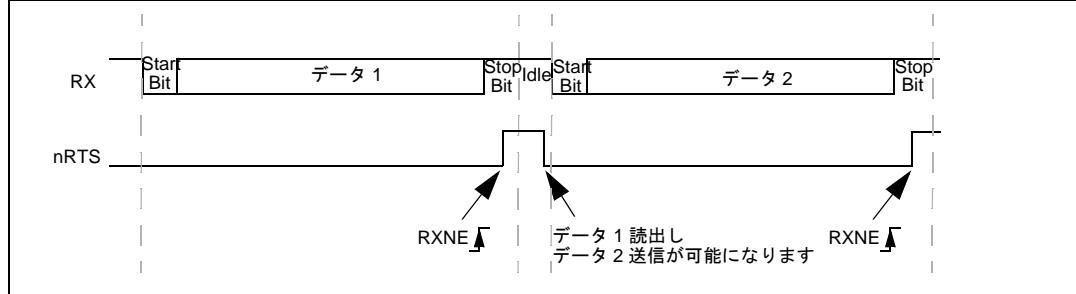


RTS・CTS フロー制御は USART_CR3 レジスタの RTSE ビット、CTSE ビットに 1 を書込むことで、個別に有効化することができます。

RTS フロー制御

RTS フロー制御が有効とされた (RTSE=1) 場合、USART 受信回路が新たなデータを受け取ることができるとき nRTS がアサートされています。受信レジスタが空でなくなると nRTS がデアサートされ、進行中の送信が終わり次第、送信動作を停止する必要があることを示します。図 254 は RTS フロー制御が有効なときの通信の例です。

図 254. RTS フロー制御

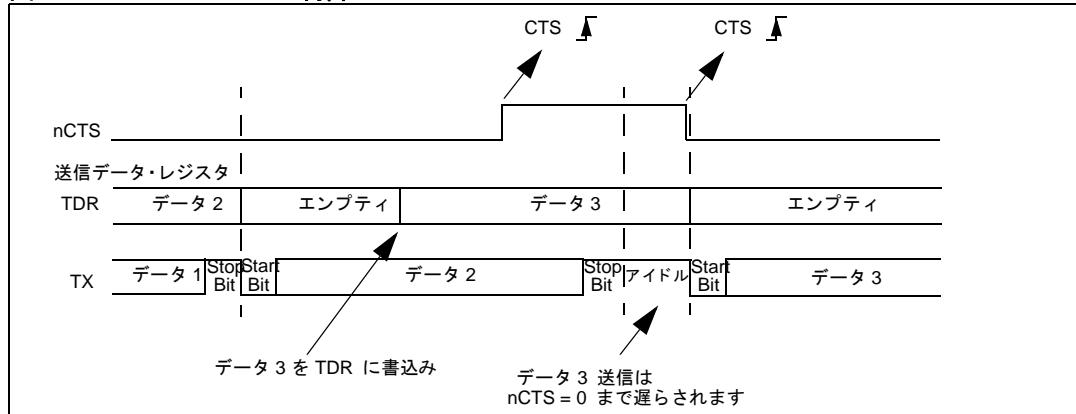


CTS フロー制御

CTS フロー制御が有効とされた (CTSE=1) 場合、送信回路は次のフレームを送出する前に nCTS 入力をチェックします。nCTS がアサートされているときには、(送信すべきデータがあるとき、すなわち TXE=0 のとき) 次のデータが送信され、アサートされていなければ送信は行われません。送信の際に nCTS がデアサートされているとき、進行中のフレーム送出が完了すると送信回路は停止します。

CTSE=1 の場合、nCTS 入力が反転すると CTSIF ステータス・ビットは直ちにハードウェアによって自動的にセットされます。このフラグは受信の用意ができているか否かを示します。このとき USART_CR3 レジスタの CTSIE ビットがセットされていれば、割込みリクエストが発生します。下図は CTS フロー制御が有効な時の通信状態の例を示しています。

図 255. CTS フロー制御



24.4 USART 割込み

表 154. USART 割込みリクエスト

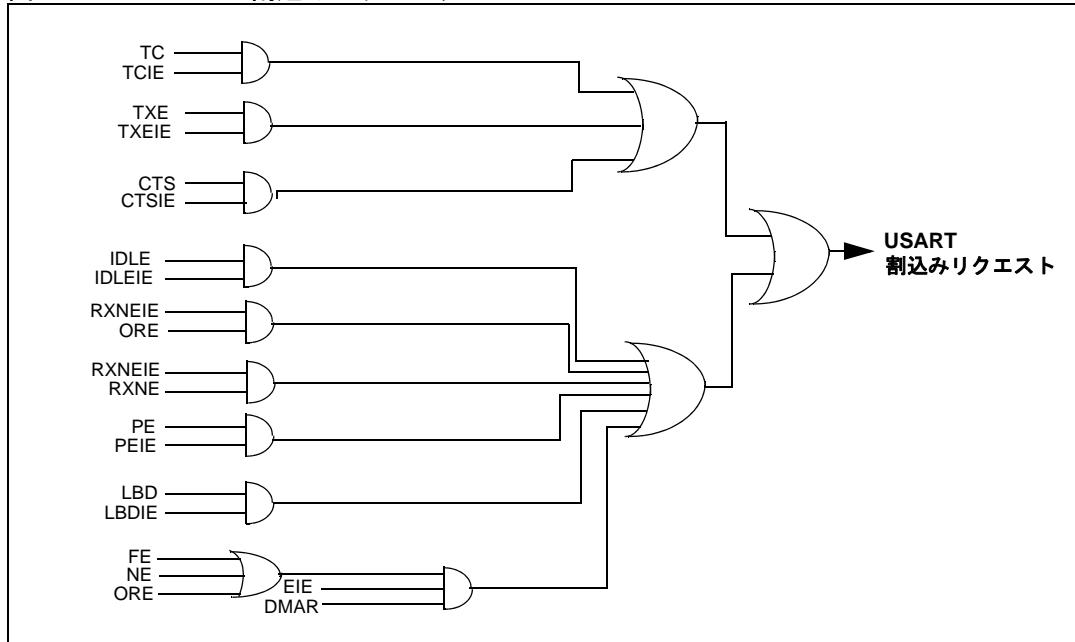
割込み要因（イベント）	イベントフラグ	割込みリクエスト許可ビット
送信データ・レジスタ・エンプティ	TXE	TXEIE
CTS フラグ	CTS	CTSIE
送信完了	TC	TCIE
受信データ・読み出し可能	RXNE	RXNEIE
オーバーラン・エラー検出	ORE	
アイドル・ライン検出	IDLE	IDLEIE
パリティ・エラー	PE	PEIE
ブレーク・フラグ	LBD	LBDIE
マルチバッファ通信動作時のノイズ・フラグ、オーバーラン・エラー、フレーミング・エラー	NE、ORE、FE	EIE

USART 割込みイベントは 1 個の同じ割込みベクタに集められています。(図 256 を参照)

- 送信関連：送信完了割込み、CTS 割込み、送信データ・レジスタ・エンプティ割込み
- 受信関連：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・データ有効、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）、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、対応する割込みリクエスト許可ビットがセットされていれば、割込みリクエストを発生させます。

図 256. USART 割込みマッピング



24.5 USART モード構成

表 155. 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	NA
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2重 (シングル・ワイヤ・モード)	X	X	X	X	X
IrDA	X	X	X	X	X
LIN	X	X	X	X	X

1. X = サポートされる ; NA = 適用されない

24.6 USART レジスタ

レジスタの詳細記述で使用されている略語については[セクション 1.1 \(32 ページ\)](#) を参照してください。

24.6.1 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

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	NE	FE	PE	r	r
rc_w0				rc_w0	rc_w0	r	rc_w0	rc_w0	r	r	r	r	r	r	r

Bits 31:10 予約済み、ハードウェアで 0 に固定されています。

Bit 9 CTS: CTS フラグ

CTSE ビットがセットされている場合、nCTS 入力信号が反転すると、このビットはハードウェアによってセットされます。このビットのクリアはソフトウェアで 0 を書込むことで行います。USART_CR3 レジスタの CTSIE=1 であれば、フラグビットがセットされると割込み要求が発生します。

0: nCTS ステータスラインの変化はありません。

1: nCTS ステータスラインに変化がありました。

注：このビットは UART4 及び UART5 では利用できません。

参考資料

Bit 8 LBD: LIN ブレーク検出フラグ

LIN ブレーク検出フラグ（ステータス・フラグ）

0: LIN ブレークは検出されていません。

1: LIN ブレークが検出されました。

注: LBDIE=1 の場合、LBD=1 になると割込み要求が発生します。

Bit 7 TXE: 送信データ・レジスタ・エンプティ

TDR レジスタの内容がシフト・レジスタに移されると、このビットがハードウェアによりセットされます。USART_CR1 レジスタの TXEIE=1 であれば、このとき割込み要求が発生します。

USRART_DR レジスタへの書き込みが行われると、このビットはクリアされます。

0: データはシフト・レジスタに移されていません。

1: データがシフト・レジスタに移されました。

注: このビットは、シングルバッファ送信の場合に使用されます。

Bit 6 TC: 送信完了

データを含むフレームの送信が完了すると、このビットがハードウェアによりセットされます。

USART_CR1 レジスタの TCIE=1 であれば、このとき割込み要求が発生します。このビットはソフトウェアで USART_SR レジスタを読み出し、その後に USART_DR レジスタへの書き込みを行うとクリアされます。

0: 送信は完了していません。

1: 送信が完了しました。

Bit 5 RXNE: 受信データ・レジスタ・データ有効

RDR シフト・レジスタの内容が USART_DR レジスタに移されると、このビットがハードウェアでセットされます。このとき、USART_CR1 レジスタの RXNEIE=1 であれば、割込み要求が発生します。このビットはソフトウェアで USART_SR レジスタを読み出し後に USART_DR レジスタを読み出すことでクリアされます。

0: データは受信されていません。

1: 受信データを USART_DR から読み出すことができます。

Bit 4 IDLE: アイドルライン検出

アイドルラインが検出されると、このビットがハードウェアでセットされます。このとき、USART_CR1 レジスタの IDLEIE=1 であれば、割込み要求が発生します。このビットはソフトウェアで USART_SR レジスタを読み出し後に USART_DR レジスタを読み出すことでクリアされます。

0: アイドルラインは検出されていません。

1: アイドルラインが検出されました。

注: RXNE ビットがセットされるまで（次のアイドルラインが発生するまで）、IDLE ビットが再びセットされることはありません。

Bit 3 ORE: オーバーラン・エラー

RXNE=1 の時に、シフト・レジスタに受信されたワードが RDR に移され得る状態となると、ハードウェアはこのビットをセットします。このとき、USART_CR1 レジスタの RXNEIE=1 であれば、割込み要求が発生します。このビットはソフトウェアで USART_SR レジスタを読み出し後に USART_DR レジスタを読み出すことでクリアされます。

0: オーバランは発生していません。

1: オーバランが発生しました。

注: このビットがセットされたとき、RDR の内容が失われてはいませんが、シフト・レジスタはこのあと重ね書きされます。マルチバッファ通信で EIE ビットがセットされているに時には ORE フラグによる割込み要求が発生します。

参考資料

Bit 2 NE: ノイズ・エラー・フラグ

受信フレームでノイズが検出されると、ハードウェアはこのビットをセットします。このビットはソフトウェアで USART_SR レジスタを読み出し後に USART_DR レジスタを読み出すことでクリアされます。

0: ノイズは検出されていません。

1: ノイズが検出されました。

注: 割込み要求を発生させることができる RXNE ビットがセットされるときに、同じタイミングでこのビットの値が決まります。このフラグが独自に割込み要求を発生させることはできません。

マルチバッファ通信で EIE ビットがセットされているに時には、割込み要求が発生します。

Bit 1 FE: フレーミング・エラー

同期はずれ、過度なノイズ、ブレーク信号が検出されたときに、このビットがハードウェアでセットされます。このビットはソフトウェアで USART_SR レジスタを読み出し後に USART_DR レジスタを読み出すことでクリアされます。

0: フレーミング・エラーは検出されていません。

1: フレーミング・エラー、もしくはブレーク信号が検出されました。

注: 割込み要求を発生させることができる RXNE ビットがセットされるときに、同じタイミングでこのビットの値が決まります。このフラグが独自に割込み要求を発生させることはできません。受信されたワードがフレームエラーとオーバ・ラン・エラーを同時に生じたときには、データが送られ、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に時には、FE フラグがセットされると割込み要求が発生します。

Bit 0 PE: パリティ・エラー

受信モードでパリティ・エラーが発生すると、このビットがハードウェアでセットされます。このとき、USART_CR1 レジスタの PEIE=1 ビットがセットされていれば、割込み要求が発生します。このビットはソフトウェアで USART_SR レジスタを読み出し後に USART_DR レジスタを読み出すことでクリアされます。

0: パリティ・エラーは発生していません。

1: パリティ・エラーが発生しました。

参考資料

24.6.2 データ・レジスタ (USART_DR)

アドレスオフセット : 0x04

リセット値 : 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
予約済み								DR[8:0]							
								rW	rW	rW	rW	rW	rW	rW	rW

Bits 31:9 予約済み、ハードウェアで 0 に固定されています。

Bits 8:0 DR[8:0]: データ値

レジスタの読み出しか、書き込みかに応じて、受信、もしくは送信キャラクタを保持しています。

データ・レジスタは送信 (TDR) と受信 (RDR) の 2 個のレジスタからなっており、読み出し、書き込みに応じて別の役割を持っています。

TDR は内部バスと出力シフト・レジスタとの間の並列インターフェースを提供しています。（図 1 を参照）

RDR は入力シフト・レジスタから内部バスへの並列インターフェースとして動作します。

パリティビット有効な設定 (USART_CR1 レジスタの PCE ビットが 1) で送信が行われるときは、MSB (データ長に応じてビット 7 もしくは 8) に書込まれた値は無視され、パリティビットで置き換えられます。

パリティビット有効な設定で受信が行われるときには、受信されたパリティビットが MSB ビットとして読み出されます。

参考資料

24.6.3 ボーレート・レジスタ (USART_BRR)

注意： TE もしくは RE ビットが無効を指示しているときには、対応するボーレートカウンタの動作は停止します。

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
DIV_Mantissa[11:0]												DIV_Fraction[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 予約済み、ハードウェアで 0 に固定されています。

Bits 15:4 DIV_Mantissa[11:0]: 分周比の整数部

この 12bits は USART の分周回路の整数部の値を設定します。

Bits 3:0 DIV_Fraction[3:0]: 分周比の小数部

この 4bits は USART の分周回路の小数部の値を設定します。

24.6.4 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

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	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNE IE	IDLEIE	TE	RE	RWU	SBK	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:14 予約済み、ハードウェアで 0 に固定されています。

Bit 13 UE: USART イネーブル

このビットがクリアされると、消費電力を低減するため、USART プリスケーラと信号出力は進行中の転送動作が終わった時点で停止します。このビットはソフトウェアでセット / クリアができます。

0: USART プリスケーラと出力信号はディセーブル状態。

1: USART イネーブル（動作）状態。

Bit 12 M: ワード長

このビットはワード長を指定します。このビットはソフトウェアでセット / クリアができます。

0: 1 スタート・ビット、8 データビット、n STOP ビット

1: 1 スタート・ビット、9 データビット、1 STOP ビット

注: M ビットは、送受信データの転送中に変更することはできません。

参考資料

Bit 11 WAKE: ウエークアップ (Wake-Up) 方法

このビットは USART のウェークアップ方法を指定します。このビットはソフトウェアでセット / クリアができます。

0: アイドル・ライン。
1: アドレスマーク。

Bit 10 PCE: パリティ制御有効

このビットはハードウェアによるパリティ制御（生成と検査）を有効とします。

パリティ制御が有効なときには、生成されたパリティビットが MSB 位置 ($M=1$ ならばビット 9、 $M=0$ ではビット 8) におかれ、受信データではパリティビットの検査が行われます。このビットはソフトウェアでセット / クリアができます。このビットは送受信中のバイトの次のバイトから有効となります。

0: パリティ制御無効。
1: パリティ制御有効。

Bit 9 PS: パリティ選択

このビットは、パリティ制御が有効なとき ($PCE=1$) に、奇数パリティか、偶数パリティを使用するかを指定します。このビットはソフトウェアでセット / クリアができます。このビットは送受信中のバイトの次のバイトから有効となります。

0: 偶数パリティ。
1: 奇数パリティ。

Bit 8 PEIE: パリティ・エラー (PE) 割込みリクエスト許可

このビットはソフトウェアでセット / クリアができます。

0: 割込みリクエスト禁止。
1: USART_SR レジスタの $PE=1$ のとき、USART 割込みリクエストを発生します。

Bit 7 TXEIE: 送信データ・レジスタ・エンプティ (TXE) 割込みリクエスト許可。

このビットはソフトウェアでセット / クリアができます。

0: 割込みリクエスト禁止。
1: USART_SR レジスタの $TXE=1$ のとき、USART 割込みリクエストを発生します。

Bit 6 TCIE: 送信完了割込みリクエスト許可。

このビットはソフトウェアでセット / クリアができます。

0: 割込みリクエスト禁止。
1: USART_SR レジスタの $TC=1$ のとき、USART 割込みリクエストを発生します。

Bit 5 RXNEIE: 受信データ・レジスタ・データ有効 (RXNE) 割込みリクエスト許可

このビットはソフトウェアでセット / クリアができます。

0: 割込みリクエスト禁止。
1: USART_SR レジスタの $RXNE=1$ もしくは $ORE=1$ のとき、USART 割込みリクエストを発生します。

Bit 4 IDLEIE: アイドル・ライン検出 (IDLE) 割込みリクエスト許可

このビットはソフトウェアでセット / クリアができます。

0: 割込みリクエスト禁止。
1: USART_SR レジスタの $IDLE=1$ のとき、USART 割込みリクエストを発生します。

Bit 3 TE: 送信イネーブル

このビットをセットすることで送信回路がイネーブル状態となります。このビットはソフトウェアでセット / クリアができます。

- 0: 送信回路ディセーブル。
- 1: 送信回路イネーブル。

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 を書込み、続いて 1 を書込む) と、送信中のワードの後に、プリアンブル (アイドル・ライン) を送出します。

2: TE ビットがセットから送信が実際に開始されるまで 1 ビット分の時間を要します。

Bit 2 RE: 受信イネーブル

このビットをセットすることで受信回路がイネーブル状態となります。このビットはソフトウェアでセット / クリアができます。

- 0: 受信回路ディセーブル。
- 1: 受信回路イネーブル。スタート・ビットの待ちうけが開始されます。

Bit 1 RWU: 受信回路ウェークアップ (Wake-Up)

このビットは USART がミュートモードであるか否かを定めます。このビットはソフトウェアでセット / クリアができます。また、ウェークアップ手順が認識されたときに、ハードウェアがこのビットをクリアします。

- 0: 受信回路はアクティブモード (通常動作) です。
- 1: 受信回路はミュートモードです。

注:

1: ミュートモードを指示する (RWU ビットをセットする) 前にまずデータバイトを受信しなければなりません。これがないと、ミュートモードで、アイドル・ライン検出によるウェー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によるウェークアップ (WAKE=1) の設定では、RXNE ビットがセットされているときには、ソフトウェアで RWU ビットを変更することはできません。

Bit 0 SBK: ブレーク送出

このビットをセットすると、ブレーク信号が送出されます。このビットはソフトウェアでセット / クリアができます。このビットはソフトウェアでセットされ、ブレーク信号の STOP ビット送出でハードウェアによりクリアされます。

- 0: ブレーク信号は送出されません。
- 1: ブレーク信号が送出されます。

24.6.5 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

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]	CLK EN	COPOL	CPHA	LBCL	予約済み	LBDIE	LBDL	予約済み		ADD[3:0]			

Bits 31:15 予約済み、ハードウェアで 0 に固定されています。

Bit 14 LINEN: LIN モードイネーブル

このビットはソフトウェアでセット / クリアができます。

0: LIN モードディセーブル

1: LIN モードイネーブル

LIN モードでは USART_CR1 レジスタの SBK ビットによる LIN 同期ブレーク (13 ローレベルビット) の送出と、LIN 同期ブレークの検出が可能となります。

Bits 13:12 STOP: STOP ビット

これらのビットは STOP ビットの指定に使用されます。

00: 1 STOP ビット

01: 0.5 STOP ビット

10: 2 STOP ビット

11: 1.5 STOP ビット

注: 0.5 STOP ビットと 1.5 STOP ビットは UART4 及び UART5 では利用できません。

Bit 11 CLKEN: クロック・イネーブル

このビットをセットすることで、SCLK ピン出力を有効とすることができます。

0: SCLK ピン・ディセーブル

1: SCLK ピン・イネーブル

注: このビットは UART4 及び UART5 では利用できません。

Bit 10 COPOL: クロック極性

このビットは、同期シリアル・モードにおける SCLK ピンのクロック出力の極性を指定します。

このビットは CPHA ビットとともに、希望のデータ・クロック間のタイミングを得るために使用されます。

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。

Bit 9 CPHA: クロック位相

このビットは、同期シリアル・モードにおける SCLK ピンのクロック出力の位相を指定します。

このビットは CPOL ビットとともに、希望のデータ・クロック間のタイミングを得るために使用されます。(図 246 から 図 247)

0: 最初のクロック遷移で最初のデータサンプルを行います。

1: 二番目のクロック遷移で最初のデータサンプルを行います。

注: このビットは UART4 及び UART5 では利用できません。

参考資料

Bit 8 **LBCL**: 最終ビットクロック

このビットは、同期シリアル・モードにおける最終データビット (MSB) 送信時にクロックが SCLK ピンに出力されるか否かを指定します。

- 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。

注 1 : 最後のビットが 8 番目のデータビットか 9 番目かは、USART_CR1 レジスタの M ビットで指定された語長の指定に依存します。

注 2 : このビットは UART4 及び UART5 では利用できません。

Bit 7 予約済み、ハードウェアで 0 に固定されています。

Bit 6 **LBDIE**: LIN ブレーク検出割込みリクエスト許可

ブレーク割込みのマスク・ビット（ブレーク区切りを使用したブレーク検出）

- 0: 割込みリクエスト禁止。
- 1: USART_SR レジスタの LBD=1 のとき、割込みリクエストが発生します。

Bit 5 **LBDL**: LIN ブレーク検出長

このビットは 10 ビットブレーク検出か、11 ビットブレーク検出かを指定します。

- 0: 10 ビットブレークを検出。
- 1: 11 ビットブレークを検出。

Bit 4 予約済み、ハードウェアで 0 に固定されています。

Bits 3:0 **ADD[3:0]**: USART ノードアドレス

これらのビットでは USART ノードのアドレスを指定します。

このアドレスはミュートモードにおけるマルチプロセッサ間の通信で、アドレスマークによる ウェークアップ (wake-up) に使用されます。

注意: *CPOL, CPHA, LBCL の 3bits は送信回路がイネーブルの状態で変更することはできません。*

参考資料

24.6.6 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

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				CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HD SEL	IRLP	IREN	EIE	

Bits 31:11 予約済み、ハードウェアで 0 に固定されています。

Bit 10 **CTSIE**: CTS 割込みリクエスト許可

0: 割込みリクエスト禁止

1: USART_SR レジスタで CTS=1 となったとき、割込みリクエストが発生します。

注: このビットは UART4 及び UART5 では利用できません。

Bit 9 **CTSE**: CTS イネーブル

0: CTS ハードウェア・フロー制御は行われません。

1: CTS モードが有効で、nCTS 入力がアサートされている（0 に接続）ときのみデータ送信が行われます。データの送信中に nCTS がアサート状態でなくなったときには、そのデータの送出が完了した時点で送信が停止します。nCTS がアサート状態でないときにデータ・レジスタにデータが書き込まれると、データの送出は nCTS がアサート状態になるまで待たれます。

注: このビットは UART4 及び UART5 では利用できません。

Bit 8 **RTSE**: RTS イネーブル

0: RTS ハードウェア・フロー制御は行われません。

1: RTS 割込みリクエストが許可されれば、受信バッファに空きがあるときにのみデータがリクエストされます。データの送信は、データの送出が完了した時点で送信が停止することが求められます。データ受信が可能なときに nRTS 出力はアサートされます。

注: このビットは UART4 及び UART5 では利用できません。

Bit 7 **DMAT**: 送信 DMA 有効

このビットはソフトウェアでセット / クリアができます。

0: 送信動作で DMA モードは使用されません。

1: 送信動作で DMA モードが使用されます。

注: このビットは UART5 では利用できません。

Bit 6 **DMAR**: 受信 DMA 有効

このビットはソフトウェアでセット / クリアができます。

0: 受信動作で DMA モードは使用されません。

1: 受信動作で DMA モードが使用されます。

注: このビットは UART5 では利用できません。

Bit 5 **SCEN**: スマートカード・モード・イネーブル

このビットをセットすることで、スマートカード・モードが有効となります。

0: スマートカード・モードはデイセーブル状態です。

1: スマートカード・モードはイネーブル状態です。

注: このビットは UART4 及び UART5 では利用できません。

参考資料

Bit 4 NACK: スマートカード NACK イネーブル

0: パリティ・エラーの際の NACK 送信は行われません。

1: パリティ・エラーの際は NACK が送信されます。

注: このビットは UART4 及び UART5 では利用できません。

Bit 3 HDSEL: 半二重選択

1 線半二重モードが指定されます。

0: 半二重モードは指定されていません。

1: 半二重モードが指定されています。

Bit 2 IRLP: IrDA 低電力モード

IrDA モードでは、このビットで通常動作と低電力モードの選択ができます。

0: 通常モード

1: 低電力モード

Bit 1 IREN: IrDA モードイネーブル

このビットはソフトウェアでセット / クリアができます。

0: IrDA モードはディセーブル状態です。

1: IrDA モードがイネーブル状態です。

Bit 0 EIE: エラー割込みリクエスト許可

マルチバッファ通信 (USART_CR3 の DMAR=1) で EIE ビットがセットされているに時、フレーミング・エラー、オーバラン、ノイズ・エラー (USART_SR レジスタの FE=1、ORE=1、NE=1) で割込みリクエストを発生させるには、エラー割込みリクエストを許可しておくことが必要です。

0: 割込みリクエスト禁止。

1: USART_CR3 レジスタの DMAR=1 の場合、USART_SR レジスタの FE=1、ORE=1、NE=1 のいずれかで、割込みリクエストを発生させます。

24.6.7 ガードタイム / プリスケーラ・レジスタ (USART_GTPR)

アドレスオフセット : 0x18

リセット値 : 0x0000

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

Bits 31:16 予約済み、ハードウェアで 0 に固定されています。

Bits 15:8 GT[7:0]: ガードタイム値

これらのビットではガードタイムの時間を、ボーレートクロックの数で指定します。これらのビットはスマートカード・モードで使用され、送信完了フラグはガードタイム経過後にセットされます。

注: このビットは UART4 及び UART5 では利用できません。

参考資料

Bits 7:0 PSC[7:0]: プリスケーラ値

- IrDA 低電力モードで、

PSC[7:0] = IrDA 低電力モード時のボーレート

低電力モードの時の周波数を得るため、プリスケーラがシステム・クロックを分周するよう指示します。

入力クロックは指定された値 (8bits) で分周されます。

00000000: 予約 - この値を設定してはいけません。

00000001: 入力クロックは 1 分周されます。

00000010: 入力クロックは 2 分周されます。

...

- IrDA 通常モードで、PSC は 00000001 に設定されていなければなりません。

- スマートカード・モードで、

PSC[4:0]: プリスケーラ値

システム・クロックからスマートカード・クロックを得るためにプリスケーラに設定する分周比を指定します。

このレジスタに設定した値 (5 ビット) を 2 倍した値が入力クロックの分周比となります。

00000: 予約 - この値を設定してはいけません。

00001: 入力クロックは 2 分周されます。

00010: 入力クロックは 4 分周されます。

00011: 入力クロックは 6 分周されます。

...

注 1: スマートカード・モードではビット [7:5] は無視されます。

注 2: このビットは UART4 及び UART5 では利用できません。

參考資料

24.6.8 USART レジスタ・マップ

表は USART のレジスタ・マップとリセット値を示しています。

表 156. USART レジスタ・マップとリセット時の値

レジスタのアドレス範囲に関しては表 1 (36 ページ) を参照してください。

25 デバイス電子署名

デバイス電子署名は、Flash メモリ・モジュールのシステム・メモリ・エリアに保存され、CPU か JTAG/SWD を使い読み込みする事が出来ます。デバイス電子署名は、ユーザ・ファームウェアや外部デバイスが、自身のインターフェースを STM32F10xxx マイクロコントローラの特性と自動的に一致させる事が出来るように、工場で予めプログラムされた識別データを含んでいます。

25.1 メモリ・サイズ・レジスタ

25.1.1 Flash サイズ・レジスタ

ベース・アドレス : 0xFFFF F7E0

読み取り専用 = 0xXXXX X は工場でプログラムされた値です。

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

Bits 15:0 **F_SIZE:** Flash メモリ・サイズ

これらのビットの値は、デバイスの Flash メモリ・サイズをキロバイト (Kbyte) で示します。
例 : 0x0080 = 128Kbyte

25.1.2 RAM サイズ・レジスタ

ベース・アドレス : 0xFFFF F7E2

読み取り専用 = 0xXXXX X は工場でプログラムされた値です。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R_SIZE															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 15:0 **R_SIZE:** RAM サイズ

この値は、デバイスの RAM サイズをキロバイト (Kbyte) で示します。
例 : 0x000A = 10 Kbyte

参考資料

25.2 ユニーク・デバイス ID レジスタ (96bit)

このユニーク・デバイス識別子は、次のような用途に適しています。

- シリアル番号として使用（最終的なアプリケーションや USB 文字列シリアル番号用）
- 内部 Flash メモリをプログラムする前に、このユニーク ID をソフトウェア暗号関数とプロトコルと組み合わせて使う事で、Flash 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ーキーとして使用。
- セキュア起動プロセスを起動する等、その他

96bit のユニーク識別子は、いかなる状況下でもあらゆるデバイスにたいして、ユニークなリファレンス番号を提供する。これらのビットは、ユーザにより変更することが出来ません。

96bit のユニーク・デバイス識別子は、シングルバイト、ハーフワード、ワード等の異なる方法で読み込みし、カスタムアルゴリズムと組み合わせる事が出来る。

ベース・アドレス : 0x1FFF F7E8

アドレスオフセット : 0x00

読み取り専用 = 0xXXXX X は工場でプログラムされた値

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

Bits 15:0 U_ID(15:0): 15:0 ユニーク ID ビット

アドレスオフセット : 0x02

読み取り専用 = 0xXXXX X は工場でプログラムされた値

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID(31:16)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 15:0 U_ID(31:16): 31:16 ユニーク ID ビット

これらのビットは、将来の機能用に予約されています。

アドレスオフセット : 0x04

読み取り専用 = 0xXXXX XXXX X は、工場でプログラムされた値

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID(63:48)															
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(47:32)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:0 U_ID(63:32): 63:32 ユニーク ID ビット

参考資料

RM0008

デバイス電子署名

アドレスオフセット : 0x08

読み取り専用 = 0xXXXX XXXX X は工場でプログラムされた値

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID(95:80)															
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(79:64)															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:0 U_ID(95:64): 95:64 ユニーク ID ビット



26 デバッグ・サポート (DBG)

Flash メモリ容量が 32 ~ 128Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**中容量デバイス**です。

Flash メモリ容量が 256 ~ 512Kbyte の STM32F101xx 及び STM32F103xx マイクロコントローラは、**大容量デバイス**です。

このセクションは、注意書きがない限り、STM32F10xxx ファミリ全てに適応されます。

26.1 概要

STM32F10xxx は Cortex-M3 コアを中心構成されており、高度なデバッグ機能のためのハードウェアの拡張機能を含んでいます。デバッグ拡張機能は特定の命令語のフェッチ（ブレークポイント）やデータ・アクセス（ウォッチポイント）コアの動作を停止することを可能としています。動作が停止しているときには、コアの内部状態とシステムの外部環境を吟味することができます。これが終わると、コアとシステムでは元の状態が復元され、プログラムの実行が再開します。

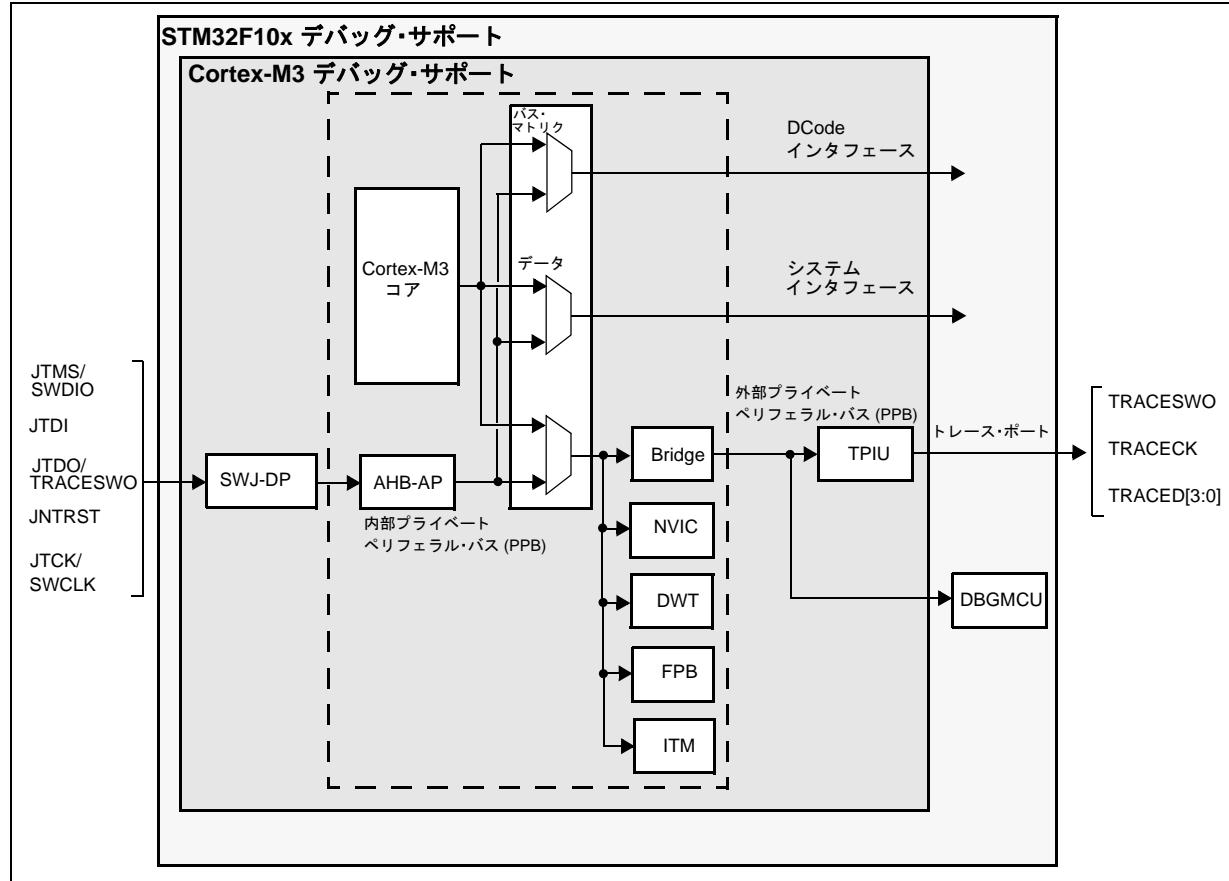
デバッグ機能は接続されているデバッグ・ホストと STM32F10x MCU によって利用されます。

デバッグのために、次の 2 種のインターフェースが利用できます。

- シリアル・ワイヤ
- JTAG デバッグ・ポート

参考資料

図 257. ブロック図 - STM32F10xxx レベルと Cortex-M3 レベルのデバッグ・サポート



注意: 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 コアは以下のブロックから成り立っているオンチップ・デバッグ・サポートを内蔵しています。

- 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: インスツルメンテーション・トレース・マイクロセル
- FPB: Flash パッチ・ブレーク・ポイント
- DWT: データ・ウォッチ・ポイント・トリガ
- TPIU: トレースポートユニット・インターフェース (対応ピンを有する大型パッケージに組み込まれています。)

次の STM32F10xxx 専用のデバッグ機能も内蔵されています。

- 融通性の高いデバッグ・ピンアウト割付け
- MCU デバッグ・ボックス (低電力モード、ペリフェラル・クロックの制御、など)

注意: ARM Cortex-M3 コアでサポートされているデバッグ機能の更なる情報は Cortex-M3 r1p1 Technical Reference Manual (TRM) と CoreSight Design Kit r1p0 TRM を参照してください。

26.2 ARM ドキュメントの参照

- *Cortex™-M3 r1p1 Technical Reference Manual (TRM)*
- *ARM Debug Interface V5*
- *ARM CoreSight Design Kit revision r1p0 Technical Reference Manual*

26.3 SWJ デバッグ・ポート (シリアル・ワイヤと JTAG)

STM32F10xxx コアにはシリアル・ワイヤ / JTAG デバッグ・ポート (SWJ-DP) が組込まれています。これは JTAG-DP (5-pin) インタフェースと SW-DP (2-pin) インタフェースの組合せからなる ARM standard CoreSight デバッグ・ポートです。

- JTAG デバッグ・ポート (JTAG-DP) は AHP-AP への 5 ピンの標準 JTAG インタフェースを備えています。
- シリアル・ワイヤ・デバッグ・ポート (SW-DP) は AHP-AP への 2 ピン (クロック + データ) のインターフェースを備えています。

SWJ-DP では、SW-DP の 2 個の JTAG ピンは JTAG-JP の 5 個の JTAG ピンと共に共用となっています。

図 258. SWJ デバッグ・ポート

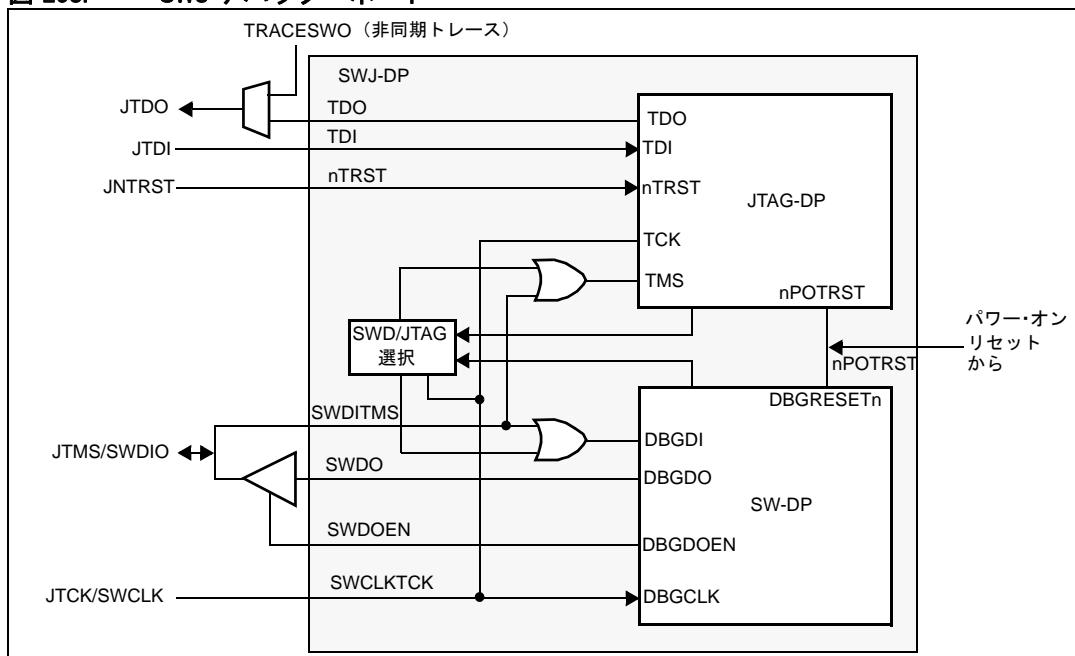


図 258 は、非同期 TRACE 出力 (TRACESWO) が TDO とマルチプレックスされている事を示しています。つまり、非同期トレースは JTAG-DP ではなく、SW-DP とだけ使うことが出来ます。

26.3.1 JTAG-DP もしくは SW-DP 選択メカニズム

デフォルトでは JTAG デバッグ・ポートが有効となっています。

デバッガ・ホストから SW-DP に切り替えるときには、TMS/TCK（それぞれ SWDIO/SWCLK に対応）で、JTAG-DP を無効とし SW-DP を有効とする専用の JTAG 手順を実行しなければなりません。この方法で SWCLK と SWDIO ピンのみを使用して、SW-DP を有効とすることができます。

この手順は、

1. TMS (SWDIO) =1 として 50 以上の TCK サイクルを送ります。
2. TMS (SWDIO) に 16 bit の列 0111100111100111 を MSB から順に送ります。
3. TMS (SWDIO) =1 として 50 以上の TCK サイクルを送ります。

26.4 ピン配置とデバッグ・ポートピン

STM32F10xxx MCU はピン数の異なる様々なパッケージで組込まれています。このため、ピンに依存する幾つかの機能はパッケージによって異なっています。

26.4.1 SWJ デバッグ・ポートピン

STM32F10xxx 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に備わっています。

表 157. SWJ デバッグ・ポートピン

SWJ-DP ピン名称	JTAG デバッグ・ポート		SW デバッグ・ポート		ピン配置
	種類	用途	種類	デバッグ割り当て	
JTMS/SWDIO	I	JTAG Test モード選択	I/O	シリアル・ワイヤ・データ入出力	PA13
JTCK/SWCLK	I	JTAG Test クロック	I	シリアル・ワイヤ・クロック	PA14
JTDI	I	JTAG Test データ入力	-	-	PA15
JTDO/TRACESWO	O	JTAG Test データ出力	-	非同期トレースが有効な時、TRACESWO	PB3
JNTRST	I	JTAG Test nReset	-	-	PB4

26.4.2 フレキシブルな SWJ-DP ピン配置

RESET (SYSRESETn/PORESETn) 後、デバッグ・ホストが利用できるように SWJ-DP の 5 本のピンはデバッグの用途に配置されています。(トレース出力ピンはデバッグ・ホストから所定の手順を実行しない限り配置されないことに注意してください。)

しかし STM32F10xxx MCU には SWJ-DP ポートの一部もしくはすべてを汎用入出力の目的で使用するようにするための REMAP_DBGAFR レジスタが用意されています。このレジスタは Cortex-M3 システム・バスに接続されている APB ブリッジにマップされています。このレジスタの設定は、デバッグ・ホストからではなく、ユーザソフトウェアで行われます。

参考資料

3 個の制御ビットで SWJ-DP ピン配置が設定されます。これらのビットはシステム・リセット時にリセットされます。

- REMAP_AF_REG (@0x40010004、STM32F10xxx MCU 内)
 - 読込み : APB - ウェイト・ステートなし
 - 書込み : APB - AHB-APB ブリッジのバッファがフルのとき、1 ウェイト

Bit 26:24 = **SWJ_CFG[2:0]**

ソフトウェアでセット / クリアができます。

この 3 bits は SWJ デバッグ・ポートとして使用されるピンの数を指定します。目的は、小さなパッケージを使用する際に、できるだけ多くのピンを汎用入出力として使用できるように解放することです。

リセット直後のデフォルト値は 0000 です。(5 ピンすべてが JTAG-DP 接続に配置されています。) この 3 bits 中の 1 bit のみがセットできます。(複数のビットをセットすることは禁止されています。)

表 158. 融通性の高い SWJ-DP ピンの割り当て

SWJ_CFG [2:0]	使用可能なデバッグ・ポート	SWJ 入出力ピン配置				
		PA13 / JTMS / SWDIO	PA14 / JTCK / SWCLK	PA15 / JTDI	PB3 / JTDO	PB4 / JNTRST
000	全 SWJ (JTAG-DP + SW-DP) 、Reset State	X	X	X	X	X
001	全 SWJ (JTAG-DP + SW-DP) 、JNTRST なし	X	X	X	X	
010	JTAG-DP 無効、SW-DP 有効	X	X			
100	JTAG-DP 無効、SW-DP 無効					解放
他	設定禁止					

注意 : REMAP_AF レジスタの書込みでは、APB ブリッジの書込みバッファがフルのとき、1 APB サイクルが余分に必要となります。これは、JTAGSW ピンの無効化が 2 サイクルで実行され、コアの nTRST と TCK 入力信号のクリーンレベルを保証するためです。

- サイクル 1: コアへの JTAGSW 入力信号は 1 もしくは 0 レベルに (TRST, TDI, TMS が 1 に、TCK が 0 に) 接続されます。
- サイクル 2: GPIO コントローラが SWJTAG 入出力ピンの制御信号 (方向、プルアップ・ダウン、シュミット・トリガ入力の有効化、など) を操作します。

26.4.3 JTAG ピンの内部プルアップ / プルダウン

JTAG 入力ピンはデバッグ・モードの機能を制御するフリップフロップに直接に接続されているため、これらがフロート状態でないことを保証することが必要となります。 SWCLK/TCK ピンはこれらの幾つかのフリップフロップのクロックと直接につながっているため、特に注意が必要とされます。

有効ではない入出力レベルを避けるため、STM32F10xxx では JTAG 入力ピンに内部でプルアップとプルダウンを組み込んであります。

- JNTRST: 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

ユーザソフトウェアで JTAG 入出力が解放されると、GPIO コントローラが再び制御を得ます。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。

- JNTRST: 入力、プルアップ
- JTDI: 入力、プルアップ
- JTMS/SWDIO: 入力、プルアップ
- JTCK/SWCLK: 入力、プルダウン
- JTDO: 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。

注意: JTAG IEEE 規格では TDI、TMS、nTRST にプルアップを付すことが推奨されていますが、TCK に関しては規定されていません。しかし STM32F10xxx では JTCK にプルダウンが組込まれています。

プルアップ、プルダウンを内蔵しているため、外部に抵抗を追加する必要がなくなります。

26.4.4 GPIO としてもシリアル・ワイヤの使用と未使用デバッグピンの解放

シリアル・ワイヤ DP を使用し、幾つかの GPIO 入出力を解放するには、ユーザソフトウェアはリセット直後に SWJ_CFG=010 をセットしなければなりません。この設定で PA15、PB3、PB4 が GPIO として使用できるようになります。

デバッグ時には、デバッグ・ホストは次の操作を行います。

- システム・リセット時、すべての SWJ ピンが配置されています。(JTAG-DP + SW-DP)
- システム・リセット時、デバッガ・ホストは JTAG-DP から SW-DP に切り替える JTAG シーケンスを送ります。
- システム・リセット時、さらに、デバッガはブレークポイントやベクタリセットを設定します。
- システム・リセットが解除されますが、コアは停止しています。
- これ以降デバッグのための通信は SW-DP を使用して行われます。他の JTAG ピンはソフトウェアで GPIO として配置することができます。

注意: ソフトウェアを考えるにあたっては、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、ソフトウェアでピンを解放するまでの間、ピンが入力・プルアップ (nTRST、TMS、TDI) もしくは入力・プルダウン (TDO) に設定されていることに留意する必要があります。

デバッグピン (JTAG、SW、TRACE) が配置されているとき、IOPORT コントローラにおける入出力ピンの設定の変更は影響を与えません。

26.5 STM32F10xxx JTAG TAP 接続

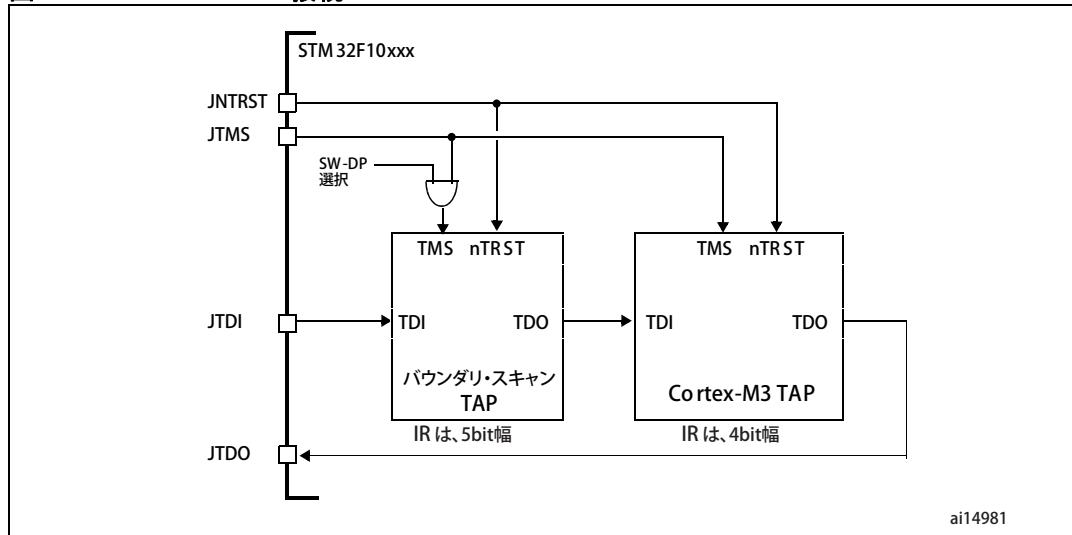
STM32F10xxx MCU は JTAG TAP の 2 通りのシリアル接続、テストに使用される TMC TAP (IR は 5 ビット幅) と、Cortex-M3 TAP (IR は 4 ビット幅) とを内蔵しています。

デバッグの目的で Cortex-M3 の TAP にアクセスするには、

1. 最初に、TMC TAP の BYPASS 命令をシフトしなければなりません。
2. 次に、個々の IR シフトごとに、スキャンチェインは 9 (=5+4) ビットからなっており、未使用の TAP 命令は BYPASS 命令でシフトされねばなりません。
3. データ・シフトごとに、BYPASS モードにある未使用の TAP にはデータスキャンチェインで余分なビットが 1 ビット付されます。

注意: **重要:** 専用の ARM JTAG シーケンスを使って Serial-Wire が選択されると、TMC TAP は自動的に無効となります。(JTMS は強制的にハイレベルにされます。)

図 259. JTAG TAP 接続



26.6 ID コードとロックメカニズム

STM32F10xxx MCU は内部に幾つかの ID コードを持っています。ツールの設計にあたっては、アドレス 0xE0042000 に配置されている外部の PPB メモリにある MCU DEVICE ID コードを利用して、デバッガをロックされるよう、ST 社は強く推奨します。

26.6.1 MCU デバイス ID コード

MCU STM32F10xxx は MCU ID コードを持っています。この ID は ST MCU のパートナンバーとダイのリビジョンを識別するものです。これは DBG_MCU 要素の一部であり、外部 PPB バス（[セクション 26.15 \(701 ページ\)](#) を参照）に配置されています。このコードは JTAG デバッグ・ポート（4 ~ 5 ピン）、SW デバッグ・ポート（2 つのピン）を使用し、もしくはユーザソフトウェアからアクセスすることができます。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 : 0xE0042000

32bit アクセスのみサポートされます。このレジスタは、読み取り専用です。

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				DEV_ID											
Res.				r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:16 REV_ID(15:0) リビジョン識別子

これらのビットはデバイスのリビジョンを表します。

中容量デバイスは :

- 0x0000 = リビジョン A
- 0x2000 = リビジョン B
- 0x2001 = リビジョン Z
- 0x2003 = リビジョン Y

大容量デバイスは :

- 0x1000 = リビジョン A
- 0x1001 = リビジョン Z

Bits 27:12 予約ビット

Bits 11:0 DEV_ID(11:0): デバイス ID

これらのビットはデバイスの ID コードを表示しています。

中容量デバイスのデバイス ID は 0x410 です。

大容量デバイスのデバイス ID は 0x414 です。

26.6.2 バウンダリ・スキャン TAP

JTAGID コード

STM32F10xxx BSC (バウンダリ・スキャン) の TAP は、下記と等しい JTAG ID コードを統合します。

- 中容量デバイスでは：
 - 0x06410041 = リビジョン A
 - 0x16410041 = リビジョン B とリビジョン Z
- 大容量デバイスでは：
 - 0x06414041 = リビジョン A

26.6.3 Cortex-M3 TAP

ARM Cortex-M3 の TAP は JTAG ID コードを持っています。ID コードは ARM のデフォルトで、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、その値は 0x3BA00477 (Cortex-M3 r1p1 相当) です。

デバッガ / プログラミングツールでは DEV_ID(15:0) のみを識別に使用できます。

26.6.4 Cortex-M3 JEDEC-106 ID コード

ARM Cortex-M3 は JEDEC-106 ID コードを持っています。これは内部 PPB バスのアドレス 0xE00FF000_0xE00FFFFF に配置されている 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 ピン)、SW デバッグ・ポート (2 つのピン)、もしくはユーザソフトウェアからアクセスできます。

26.7 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。完全な詳細情報に関しては、*Cortex-M3 r1p1 Technical Reference Manual (TRM)* を参照してください。

参考資料

表 159. JTAG デバッグ・ポート・データ・レジスタ

IR(3:0)	データ レジスタ	摘要
1111	BYPASS [1 bit]	
1110	IDCODE [32 bits]	ID コード 0x3BA00477 (ARM Cortex-M3 r1p1 ID コード)
1010	DPACC [35 bits]	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とします。 - 入力転送時： Bits 34:3 = DATA[31:0] = 書込みリクエストで転送される 32 ビットデータ Bits 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 Bit 0 = RnW = 読出しリクエスト (1) もしくは書込みリクエスト (0). - 出力転送時： Bits 34:3 = DATA[31:0] = 読込みリクエストにより読み出される 32 ビットデータ Bits 2:0 = ACK[2:0] = 3 ビット確認応答 010 = OK/FAULT 001 = WAIT OTHER = 予約値 A(3:2) ビットに関しては 表 160 を参照してください。
1011	APACC [35 bits]	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とします。 - 入力転送時： Bits 34:3 = DATA[31:0] = 書込みリクエストでシフトされる 32bit データ Bits 2:1 = A[3:2] = 2bit アドレス (サブアドレス AP レジスタ) Bit 0 = RnW= 読出しリクエスト (1) もしくは書込みリクエスト (0) - 出力転送時： Bits 34:3 = DATA[31:0] = 読出しリクエストにより読み出される 32bit データ Bits 2:0 = ACK[2:0] = 3bit 確認応答 010 = OK/FAULT 001 = WAIT OTHER = 予約値 次の項目の組合せで、多くの AP レジスタ (AHB-AP を参照) アドレスが指定できます。 - シフトされた A[3:2] - DP SELECT レジスタの現在の内容
1000	ABORT [35 bits]	アボートレジスタ - Bits 31:1 = 予約済み - Bit 0 = DAPABORT: DAP アボートを発生するには 1 を書込みます。

参考資料

表 160. A[3:2] で選択される 32bit デバッグ・ポート・レジスタ

アドレス	A(3:2) の値	摘要
0x0	00	予約済み
0x4	01	DP CTRL/STAT レジスタ . 次の目的で使用されます。 - システムもしくはデバッグパワー・アップ・リクエスト - AP アクセスのための転送動作の設定 - プッシュト比較もしくはプッシュト・ベリファイ操作の制御 - ステータス・フラグ（オーバラン、パワー・アップ確認応答）の読出し
0x8	10	DP SELECT レジスタ : アクセスするポートとアクティブな 4 ワード・レジスタ・ウィンドウの選択に使用されます。 - Bits 31:24: APSEL: カレント AP の選択 - Bits 23:8: 予約済み - Bits 7:4: APBANKSEL: カレント AP で有効な 4 ワード・レジスタ・ウィンドウの選択 - Bits 3:0: 予約済み
0xC	11	DP RDBUFF レジスタ : デバッガが一連の操作の結果を（新たな JTAG-DP 操作をリクエストせずに）得ることを可能とするために使用されます。

26.8 SW デバッグ・ポート

26.8.1 SW プロトコル概要

この同期式シリアルプロトコルでは 2 ピンが使用されます。

- SWCLK: ホストからターゲットへのクロック
- SWDIO: 両方向

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) の読み書きができます。

ビットは LSB から先に転送されます。

SWDIO 両方向管理のためには、ラインはボード上でプルアップ (ARM では 100 KΩ が推奨されています) されている必要があります。

プロトコルで SWDIO の転送方向が変わることに、ラインがホストからもターゲットからも駆動されていない切替え時間が挿入されます。デフォルトではこの切替え時間は 1 ビットタイムですが、SWCLK の周波数にあわせて設定することができます。

26.8.2 SW プロトコル手順

各手順は 3 フェーズからなっています。

1. ホストによるパケット・リクエストの送出 (8bit)
2. ターゲットによる確認応答の送出 (3bit)
3. ホストもしくはターゲットによるデータ転送フェーズ (33bit)

参考資料

表 161. パケット・リクエスト (8bit)

ビット	名称	摘要
0	Start	“1”でなければなりません
1	APnDP	0: DP アクセス 1: AP アクセス
2	RnW	0: 書込みリクエスト 1: 読出しリクエスト
4:3	A(3:2)	DP/AP レジスタのアドレス (表 160 を参照)
5	Parity	先行しているビットに対するパリティビット
6	Stop	0
7	Park	ホストは駆動しません。プルアップの効果で、ターゲットで 1 と読まれねばなりません。

DPACC レジスタと APACC レジスタの詳細に関しては、*Cortex-M3 r1p1 TRM* を参照してください。

パケット・リクエストの後には、常に、ホスト・ターゲットともラインを駆動していない切替え時間（デフォルト 1bit）がおかれます。

表 162. ACK 応答 (3bit)

ビット	名称	摘要
0..2	ACK	001: FAULT 010: WAIT 100: OK

READ トランザクション、もしくは WAIT、FAULT 応答の場合にのみ、ACK 応答の後に切替え時間が必要とされます。

表 163. DATA 転送 (33bit)

ビット	名称	摘要
0..31	WDATA もしくは RDATA	書込み / 読出しデータ
32	Parity	32bit データに対するパリティビット

READ トランザクションの場合、DATA 転送の後には切替え時間が必要とされます。

26.8.3 SW-DP ステートマシン (リセット、アイドルステート、ID コード)

SW-DP のステートマシンは、SW-DP を識別する、JEP-106 規格に準じた 内部 ID コードを持っています。この ID コードは ARM のデフォルトで、値は 0x1BA01477 (Cortex-M3 r1p1) となっています。

注意: ターゲットがこの ID コードを読出すまで、SW-DP ステートマシンが動作状態にないことに注意してください。

- パワー・オン・リセット直後、JTAG もしくは SWD から DP に切替えられた直後、もしくはラインが 50 サイクル以上の間ハイレベルを保っていたときには、SW-DP ステートマシンは RESET STATE になっています。
- RESET 状態で、少なくとも 2 サイクルの間ラインがローレベルとされると、SW-DP ステートマシンは IDLE STATE になります。
- RESET 状態で、まず IDLE 状態に入り、SW-DP ID CODE レジスタを読出す操作が必要とされます。これが行われていないと、ターゲットは他のトランザクションに対して FAULT 応答を返します。

SW-DP ステートマシンの詳細情報は *Cortex-M3 r1p1 TRM* 及び *CoreSight Design Kit r1p0 TRM* に記載されています。

26.8.4 DP 及び AP の読出し / 書込みアクセス

- DP への読出しアクセスはポストされません。ターゲットは、ACK=OK であれば直ちに、ACK=WAIT であれば遅れて応答します。
- AP への読出しアクセスはポストされます。このため、アクセスの結果は次の転送時に戻されます。次のアクセスが AP アクセスでないときには、結果を得るには DP-RDBUFF レジスタを読出します。
DP-CTRL/STAT レジスタの READOK フラグは、AP 読出しが正しく行われたか否かを表示するため、AP 読出しアクセス及び RDBUFF 読出しリクエストごとに更新されます。
- SW-DP はバッファへの書込み (DP 及び AP への書込み) をサポートしており、他のトランザクションが処理中であっても書込み操作を受けます。書込みバッファがフルのときには、ターゲットは WAIT 応答を返します。ただし、IDCODE 読出し、CTRL/STAT 読出し、ABORT 書込みはバッファがフルであっても受けられます。
- クロック・ドメイン SWCLK と HCLK が同期していないため、書込みトランザクション (パリティビット後) が内部で有効となるには 2 SWCLK が余分に必要とされます。このサイクルの間はラインをローレベルに駆動しなければなりません。(IDLE ステート)
これはパワー・アップ・リクエストのために CTRL/STAT 書込みを行う際に特に重要です。パワー・アップ状態が必要な次のトランザクションが直ちに発行されると、これは不成功に終わります。

参考資料

26.8.5 SW-DP レジスタ

これらのレジスタへのアクセスは APnDP=0 の時に開始されます。

表 164. 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	備考
00	Read		IDCODE	製造者コードは ST 社のコードではありません。値は 0x1BA01477 です。(SW-DP を識別します。)
00	Write		ABORT	
01	Read/Write	0	DP- CTRL/STAT	用途は、 - システム / デバッグパワー・アップ・リクエスト - AP アクセス時の転送動作の設定 - プッシュト比較操作、プッシュト・ベリファイ操作の制御 - ステータス・フラグの読み出し（オーバラン、パワー・アップ確認）
01	Read/Write	1	WIRE CONTROL	シリアルポート物理層プロトコルの設定を行います。（切替え時間など）
10	Read		READ RESEND	元の AP 転送を再度行うことなく、デバッグ転送でエラーとなった読み出しデータの修復を可能とします。
10	Write		SELECT	アクセスポートと有効な 4 ワードレジスタ ウィンドウの選択を行います。
11	Read/Write		READ BUFFER	AP アクセスがポストされますので、読み出しバッファの使用が効果的です。（AP 読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。） 読み出しバッファは AP からの先の読み出しによるデータを、新たなトランザクションを起動しなくとも保持します。

参考資料

26.8.6 SW-AP レジスタ

これらのレジスタへのアクセスは APnDP=1 の時に開始されます。

次の項目の組合せで、多くの AP レジスタ (AHB-AP を参照) アドレスが指定できます。

- シフトされた A[3:2]
- DP SELECT レジスタの現在の内容

26.9 AHB-AP (AHB アクセスポート) – JTAG-DP もしくは SW-DP で有効

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 及び JTAG-DP の両方からの AHB-AP へのアクセス
- AHB-AP はバス・マトリックスで AHB マスターとなります。また、AHB-AP は ICode を除くすべてのバス (Dcode バス、システム・バス、内部 PPB バス及び外部 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ています。
- AHB-AP トランザクションは FPB を迂回します。

32bits AHB-AP レジスタは 6bits 幅 (最大 64 ワード、もしくは 256 bytes) で、以下の構成となっています。

- f) Bits [8:4] = DP_SELECT レジスタのビット [7:4] APBANKSEL
- g) Bits [3:2] = SW-DP 35bit パケット・リクエストの 2bits アドレス

Cortex-M3 の AHB-AP は、9 個の 32bits レジスタから構成されています。

表 165. Cortex-M3 AHB-AP レジスタ

アドレスオフセット	レジスタ名	備考
0x00	AHB-AP 制御 / ステータスワード	AHB インタフェースを介した転送の設定と制御を行います。(サイズ、hprot、転送のステータス、アドレス計算タイプ)
0x04	AHB-AP 転送アドレス	
0x0C	AHB-AP データ読み込み / 書込み	
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ることなく、4 個の整列された データワードを直接に配置します。
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・インターフェースのベース・アドレス
0xFC	AHB-AP ID レジスタ	

詳細は *Cortex-M3 r1p1 TRM* を参照してください。

26.10 コア・デバッグ

コア・デバッグはコア・デバッグ・レジスタを通してアクセスされます。デバッグはこれらのレジスタに *Advanced High-performance Bus (AHB-AP)* ポートを介してアクセスします。プロセッサはこれらのレジスタに内部の *Private Peripheral Bus (PPB)* を通して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 166. コア・デバッグ・レジスタ

レジスタ	摘要
DHCSR	デバッグ・ホールト制御 / ステータス・レジスタ (32bit) このレジスタはプロセッサの状態に関する情報を表示し、コア・デバッグ・ホールトの有効化とプロセッサのステップ動作を制御します。
DCRSR	デバッグ・コア・レジスタ選択レジスタ (17bit) このレジスタは、データの読み書きを行うプロセッサのレジスタを選択します。
DCRDR	デバッグ・コア・レジスタ・データ・レジスタ (32bit) このレジスタは、DCRSR (セレクタ) レジスタで選択されたプロセッサレジスタの読み書きの際のデータを保持します。
DEMCR	デバッグ例外 / モニタ制御レジスタ (32bit) このレジスタはベクタキャッシングとデバッグモニタ制御を提供します。このレジスタは TRACE の使用を有効とする TRCENA ビットが用意されています。

注意: **重要:** これらのレジスタはシステム・リセットでリセットされません。レジスタのリセットはパワー・オン時のリセットに限られます。

詳細は *Cortex-M3 r1p1 TRM* 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 bit 0 (VC_CORRESET) をイネーブルに設定します。
- デバッグ・ホールト制御 / ステータス・レジスタの bit 0 (C_DEBUGEN) をイネーブルに設定します。

26.11 システム・リセット時のデバッグ・ホスト接続

STM32F10xxx MCU のリセットシステムは次のリセット要因で構成されています。

- POR (パワー・オン・リセット) - 電源オンごとに RESET が発生します。
- 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 (概ね PORRESETn) は他の部分 (SYSRESETn) とは別になっています。

このため、リセット・ベクタをフェッチする時点でコアの動作を停止するするようコア・デバッグ・レジスタで指定することで、デバッガはシステム・リセット状態でも接続が可能となります。このときホストはシステム・リセットを発することができ、コアは命令を実行すことなく直ちに停止状態となります。さらに、システム・リセット下でデバッグ機能を使用することができます。

注意： システム・リセット下でもデバッガ・ホストの接続を維持する（リセット・ベクタにブレークポイントを設定する）ことが推奨されます。

26.12 FPB

(Flash patch breakpoint : Flash パッチ・ブレークポイント)

FPB (Flash Patch Breakpoint) ユニットは

- 複数のハードウェア・ブレークポイントを実現しています。
- コードスペースからデータスペースに、コードとデータをパッチします。この機能はコードメモリ空間に置かれたソフトウェアのバグの訂正を可能とします。

ソフトウェア・パッチとハードウェア・ブレークポイントの利用は二者択一です。

FPB は以下から構成されています。

- コード空間からのリテラルロードをチェックし、システム空間の対応するドメインに再配置する 2 個のリテラルコンパレータ。
- コードスペースからの命令フェッチをチェックする 6 個のコンパレータ。対応するシステム空間への再配置、もしくはコアへのブレークポイント命令の生成のいずれかに使用されます。

26.13 DWT

(data watchpoint trigger : データ・ウォッチポイント・トリガ)

DWT (Data Watchpoint Trigger) は 4 個のコンパレータから構成されており、次の用途に設定されます。

- ハードウェア・ウォッチポイント
- ETM へのトリガ
- PC サンプラー
- データアドレスサンプラー

また、DWT はプロファイル情報を得る手段を備えています。例えば、次の数を得るために幾つかのカウンタにアクセスできます。

- クロック・サイクル数
- 処理中の命令数
- ロード・ストアユニット (LSU) の動作
- スリープサイクル数
- CPI (命令あたりのクロック数)
- 割込みオーバヘッド

26.14 ITM

(instrumentation trace macrocell : 計測トレース・マクロセル)

26.14.1 概要

ITM (Instrumentation Trace Macrocell) はアプリケーションにより操作されるトレース機能で、OS やアプリケーションのイベントをトレースしシステムの診断情報を出力する、printf 方式のデバッグをサポートしています。パケット化される ITM からの診断情報としては、

- ソフトウェアトレース : ソフトウェアは直接 ITM レジスタに書き込みを行い、パケットを発生させることができます。
- ハードウェアトレース : DWT はパケットを生成し、ITM がこれを出力します。
- タイムスタンプ : タイムスタンプはパケットとの相対値で出力されます。ITM はタイムスタンプを生成するために 21bit のカウンタを搭載しています。Cortex-M3 クロックもしくは *Serial Wire Viewer* (SWV) 出力のビットクロックがカウンタのクロックとなります。

ITM によってパケットは TPIU (Trace Port Interface Unit) に出力されます。TPIU のフォーマットは幾つかのパケットを追加 (TPIU の項を参照) し、デバッガ・ホストに完全なパケット列として送出します。

ITM を使用するには、デバッグ例外・モニタ制御レジスタの TRCEN ビットをイネーブルにセットしておかねばなりません。

26.14.2 タイムスタンプ・パケット、同期及びオーバフロー・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及び同期情報をコード化して保持しています。このために、適当なプリスケーラに駆動され、タイムスタンプ・パケットの発行ごとにリセットされる 21bit のタイムスタンプ・カウンタを使用します。カウンタは CPU クロックもしくは SWV クロックのいずれかで駆動されます。

同期パケットは 6 バイトの値 0x80 00 00 00 00 00 からなっており、TPIU には下位バイトが順に 00 00 00 00 00 80 のように送られます。

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、DWT トリガの都度、発行されます。

このためには、DWT は ITM へのトリガを、DWT 制御レジスタの CYCCNTENA (bit 0) をセットすることで、設定しておかねばなりません。

注意 : SYNENA ビットがセットされていないときには、DWT は TPIU に対する同期トリガを発し、TPIU が TPIU 同期パケットを送りますが、ITM 同期パケットは発行されません。

オーバフロー・パケットは、データを書込もうとした際に FIFO がフルであるときに発行される特殊なタイムスタンプを保持しています。

参考資料

表 167. 主な ITM レジスタ

アドレス	レジスタ	詳細
@E0000FB0	ITM アクセスロック	他の ITM レジスタへの書き込みを可能とするには、このレジスタに 0xC5ACCE55 を書き込みます。
@E0000E80	ITM トレース制御	Bits 31–24 = 常時 0
		Bits 23 = BUSY
		Bits 22–16 = トレースデータの元を識別する 7bit の ATB ID
		Bits 15–10 = 常時 0
		Bits 9:8 = TSPrescale: タイムスタンプ・プリスケーラ
		Bits 7–5 = 予約済み
		Bit 4 = SWOENA: SWV 動作（タイムスタンプ・カウンタを SWV クロックで駆動）をイネーブルとします。
		Bit 3 = DWTENA: DWT へのトリガ動作をイネーブルとします。
		Bit 2 = SYNCENA: DWT が同期トリガを発し、TPIU が同期パケットを発行できるようにするには、このビットは 1 でなければなりません。
		Bit 1 = TSENA: タイムスタンプ・イネーブル
		Bit 0 = ITMENA: ITM 全体のイネーブル・ビット
@E0000E40	ITM トレース優先度	Bit 3: ポート 31:24 のトレースを禁止します。
		Bit 2: ポート 23:16 のトレースを禁止します。
		Bit 1: ポート 5:8 のトレースを禁止します。
		Bit 0: ポート 7:0 のトレースを禁止します。
@E0000E00	ITM トレースイネーブル	個々のビットはトレースを発生させるポートを有効とします。
@E0000000-E000007C	ステイミュラスポートレジスタ 0 – 31	トレース出力時にポートに出力する 32 ビットのデータを選択されたステイミュラスポート（32 利用可能）へ書き込みます。

設定例

単純な値を TPIU に出力するには、

- TPIU を設定し、DBGMCU_CR を設定することで TRACE I/O を割り当てます。（[セクション 26.16.2: トレース・ピン配置](#)と[セクション 26.15.3: デバッグ MCU 構成レジスタ](#)を参照）
- 0xC5ACCE55 を ITM アクセス・ロック・レジスタに書き込み、ITM レジスタへの書き込みロックを解除します。
- 0x00010005 を ITM トレース制御レジスタに書き込み、ITM の同期パケットの発行をイネーブルとし ATB ID を 0x00 以外の値とします。
- 0x1 を ITM トレース・イネーブル・レジスタに書き込み、ポート 0 を有効とします。
- 0x1 を ITM トレース優先度レジスタに書き込み、ポート 7:0 のトレースを可能とします。
- ソフトウェアにより（printf 関数を使用して）ポート 0 レジスタに出力する値を書き込みます。

26.15 MCU デバッグ・コンポーネント (MCUDBG)

MCU デバッグ・コンポーネントは次のサポートを行うことでデバッガの働きを助けます。

- 低電力モード
- ブレークポイントにおける、タイマ、ウォッチドッグ、bxCAN のクロック制御
- トレース・ピンの配置制御

26.15.1 低電力モードにおけるデバッグのサポート

低電力モードに入るには、WFI もしくは WFE 命令の実行が必要とされます。

MCU には CPU クロックを停止し CPU の電力を削減する幾つかの低電力モードが組込まれています。

デバッグセッションで、コアの FCLK 及び HCLK をオフにすることはできません。これらはデバッガとの接続を維持するのに必要なため、アクティブな状態を保たねばなりません。MCU はユーザが低電力モードのソフトウェアをデバッグできるようにするために、特別の手段を用意しています。

この目的のため、デバッガ・ホストは幾つかのデバッグ構成レジスタの設定を行い、低電力モードでの挙動を変更しておかねばなりません。

- SLEEP モードでは、DBGMCU_CR レジスタの DBG_SLEEP ビットはデバッガによって事前にセットされなければなりません。これにより HCLK に FCLK と同じクロックが供給されます。(ソフトウェアで事前にシステム・クロックを設定しておくことが必要です。)
- STOP モードでは、デバッガにより事前に DBG_STOP ビットをセットしておくことが必要です。これにより内部の RC 発振回路が STOP モードにおける FCLK と HCLK を供給します。

26.15.2 タイマ、ウォッチドッグ、bxCAN と I²C に対するデバッグ・サポート

ブレークポイントにおいて、タイマのカウンタやウォッチドッグがいずれの状態となるかを指定することが必要とされます。

- ブレークポイントの中でもカウントを続けます。これは、一例としてモーターを制御している PWM で必要とされます。
- ブレークポイントではカウントを停止できます。これはウォッチドッグで必要となります。

bxCAN の場合は、ブレークポイントでは受信レジスタを更新しないという選択を行うことができます。

I²C の場合は、ブレークポイントでは SMBUS タイムアウトをブロックするという選択を行うことができます。

参考資料

26.15.3 デバッグ MCU 構成レジスタ

このレジスタは DEBUG 時の次のことをに関する MCU の構成を行います。

- 低電力モードのサポート
- タイマ及びウォッチドッグ・カウンタのサポート
- bxCAN 通信サポート
- トレース・ピンの配置

この DBGMCU_CR レジスタはアドレス 0xE0042004 にある外部 PPB バス上に配置されています。

PORESET によるリセット動作は、他とは独立に発生します。システム・リセット中に、デバッガによる書き込みで、このリセットを発生させることもできます。

デバッガ・ホストがこの機能をサポートしていないときには、ユーザソフトウェアによるレジスタへの書き込みでリセットを発生させることも可能です。

DBGMCU_CR

アドレス : 0xE0042004

32 ビットアクセスのみが可能です。

POR リセット値 : 0x00000000 (システム・リセットではリセットされません)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
											DBG_TIM8_STOP	DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_I2C2_SMBUS_TIMEOUT
											Res.	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DBG_I2C1_SM_BUS_TIMEOUT	DBG_CAN_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP	DBG_TIM1_STOP	DBG_WWDG_STOP	DBG_IWDG_STOP	TRACE_MODE [1:0]	TRACE_IOEN	予約済み	DBG_STANDBY	DBG_STOP	DBG_SLEEP		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	Res.	rw	rw	rw		

Bits 31:21 予約済み、0 の状態を保たねばなりません。

Bits 20:17 **DBG_TIMx_STOP:** コア停止時は TIMx カウンタ停止 ($x=8..5$)

0: コアの停止中もタイマ・カウンタへのクロックは供給され、出力は通常動作です。

1: コアの停止中はタイマ・カウンタへのクロックも停止します。出力はディセーブルとなります。
(ブレークイベントへの緊急停止の応答と同様)

Bit 16 **DBG_I2C2_SMBUS_TIMEOUT:** コア停止時の SMBUS タイムアウト・モード

0: 通常と同じ動作です。

1: SMBUS タイムアウトは停止します。

Bit 15 **DBG_I2C1_SMBUS_TIMEOUT:** コア停止時の SMBUS タイムアウト・モード停止

0: 通常と同じ動作です。

1: SMBUS タイムアウトは停止します。

Bit 14 **DBG_CAN_STOP:** コア停止時は CAN 停止

0: 通常の状態と同じ動作をします。

1: CAN の受信レジスタは同じ状態を保ちます。(frozen)

参考資料

Bits 13:10 **DBG_TIMx_STOP:** コア停止時はタイマ x 停止 (x=4..1)

- 0: コアの停止中もタイマ x カウンタのクロックは供給されます。
- 1: コアの停止中はタイマ x カウンタのクロックも停止します。

Bit 9 **DBG_WWDG_STOP:** コア停止時は Window 型ウォッチドッグ停止

- 0: コアの停止中も Window 型ウォッチドッグ・カウンタのクロックは供給されます。
- 1: コアの停止中は Window 型ウォッチドッグ・カウンタのクロックも停止します。

Bit 8 **DBG_IWDG_STOP:** コア停止時は独立型ウォッチドッグ停止

- 0: コアの停止中もウォッチドッグ・カウンタのクロックは供給されます。
- 1: コアの停止中はウォッチドッグ・カウンタのクロックも停止します。

Bits 7:5 **TRACE_MODE[1:0] and TRACE_IOEN:** トレース・ピンの配置

- **TRACE_IOEN=0 の時**

TRACE_MODE=xx: TRACE ピンは配置されません。(デフォルト)

- **TRACE_IOEN=1 の時**

TRACE_MODE=00: TRACE ピン配置は非同期モードです。

TRACE_MODE=01: TRACE ピン配置は同期モードです。(TRACEDATA サイズ 1)

TRACE_MODE=10: TRACE ピン配置は同期モードです。(TRACEDATA サイズ 2)

TRACE_MODE=11: TRACE ピン配置は同期モードです。(TRACEDATA サイズ 4)

Bit 4:3 予約済み、0 の状態を保たねばなりません。

Bit 2 **DBG_STANDBY:** デバッグ STANDBY モード

0: (FCLK=Off, HCLK=Off) デジタル部分全体が電源オフとなります。

ソフトウェアから見て、STANDBY モードからの終了は、リセット・ベクタのフェッチと同じ動作をします。(いくつかのステータス・ビットが STANDBY モードからの終了であることを示す点で異なります。)

1: (FCLK=On, HCLK=On) デジタル部分は電源オフ状態ではなく、FCLK と HCLK がアクティブ状態の内蔵 RC 発振回路から供給されます。さらに、MCU は STANDBY モード中に MCU はシステム・リセットを発生します。これにより、STANDBY モードを終了した時の状態がリセット・ベクタのフェッチと同等となります。

Bit 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・コントローラがディセーブル状態となり、HCLK と FCLK を含むすべてのクロックが停止します。STOP モードからの終了は、クロックの設定状態がリセット直後と同じになります。(CPU は内蔵 8 MHz 発振回路 (HSI) から供給されます。) したがって、ソフトウェアでクロック・コントローラの設定を再度行い、PLL やクリスタル発振子を動作状態としなければなりません。

1: (FCLK=On, HCLK=On) STOP モードに入ったとき、HCLK と FCLK は STOP モードでもアクティブ状態にある内蔵 RC 発振回路から供給されます。STOP モードからの終了は、ソフトウェアでクロック・コントローラの設定を再度行い、PLL やクリスタル発振子を動作状態としなければなりません。(DBG_STOP=0 の場合と同様です。)

Bit 0 **DBG_SLEEP:** デバッグ SLEEP モード

0: (FCLK=On, HCLK=Off) SLEEP モードでは FCLK にはソフトウェアで指定されているようにシステム・クロックが供給され、HCLK はオフとなります。

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、ソフトウェアで設定された状態が残ります。したがって、STOP モードから抜け出す際にクロック・コントローラを再び設定する必要はありません。

1: (FCLK=On, HCLK=On) SLEEP モードに入る際には、HCLK にも FCLK と同じクロック（事前に設定されているシステム・クロック）が供給されます。

26.16 TPIU (trace port interface unit)

26.16.1 概要

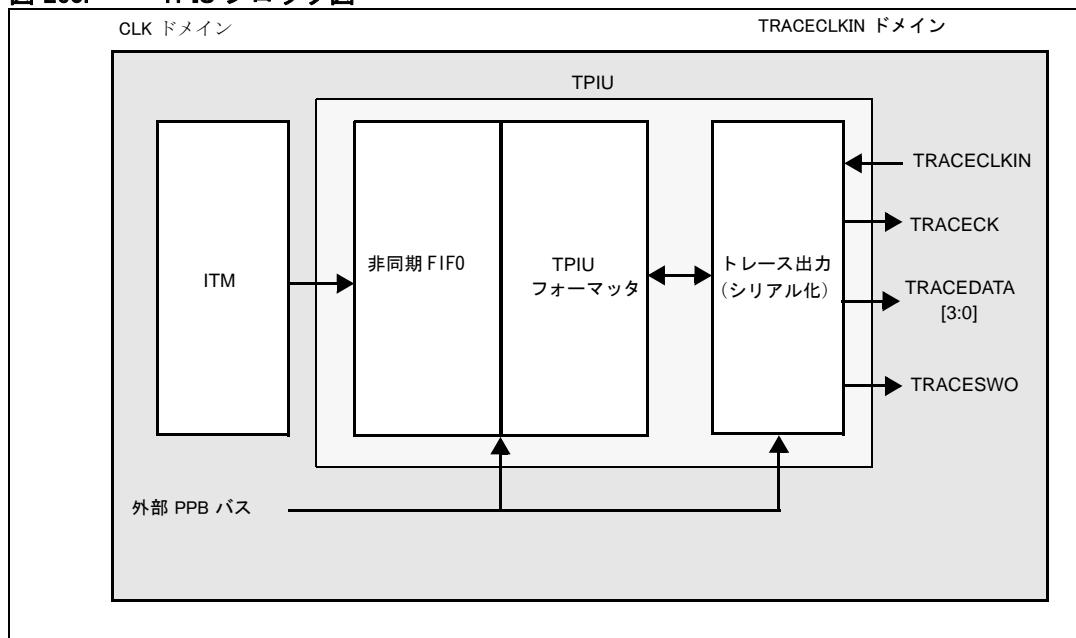
TPIU (Trace Port Interface Unit) は ITM からのオンチップ・トレースデータとのブリッジとしての役割を果たします。

TPA (Trace Port Analyzer) に渡される 出力データの流れには、トレース元の ID が含まれています。

低コストのデバッグを可能とするため、コアには単純な TPIU (特別バージョンの CoreSight TPIU) が内蔵されています。

TPIU は ITM からの情報出力のみを行う ITM デバッグトレースのみをサポートしています。

図 260. TPIU ブロック図



26.16.2 トレース・ピン配置

- **非同期モード**

非同期モードでは 1 ピンのみが必要とされ、すべてのパッケージで利用できます。このモードはシリアル・ワイヤ・モード時のみ使用できます。(JTAG モードでは不可。)

表 168. 非同期 トレース・ピン割り当て

TPIU ピン名称	非同期モード		STM32F10xxx ピン配置
	種別	摘要	
TRACESWO	O	TRACE 非同期データ出力	PB3

- **同期モード**

同期モードはデバッグトレースサイズに応じて 2 ~ 6 ピンを必要とし、大型パッケージでのみ使用できます。このモードは JTAG モード、シリアル・ワイヤ・モードのいずれでも利用でき、非同期モードより大きなバンド幅を備えています。

表 169. 同期 トレース・ピン割り当て

TPIU ピン名称	同期モード		STM32F10xxx ピン割り当て
	種別	摘要	
TRACECK	O	TRACE クロック	PE2
TRACED[3:0]	O	TRACE 同期データ出力 サイズは 1, 2, 4 のいずれか	PE[6:3]

TPIU トレース・ピン配置

デフォルトでは、これらのピンは配置されていません。これらのピンは **MCU Debug 要素構成レジスタ** の IOTRACEN ビット及び IOTRACEMODE ビットの指定で配置されます。この設定はデバッガ・ホストで行わねばなりません。

さらに、ピンの数はトレースの設定（非同期 / 同期）によって変化します。

- **非同期モード** : 1 ピンが必要とされます。
- **同期モード** : トレースポート・レジスタの設定によるデータ・サイズに応じて、2 ~ 5 ピンが必要となります。
 - TRACECK
 - TRACED(0)、ポート・サイズが 1, 2, 4bit の設定時
 - TRACED(1)、ポート・サイズが 2, 4bit の設定時
 - TRACED(2)、ポート・サイズが 4bit の設定時
 - TRACED(3)、ポート・サイズが 4bit の設定時

TRACE ピンを配置するには、デバッガ・ホストはデバッグ MCU 構成レジスタ (DBGMCU_CR) の TRACE_IOEN ビットと TRACE_MODE[1:0] ビットを設定しなければなりません。デフォルトでは TRACE ピンは配置されていません。

このレジスタは外部 PPB 上に配置されており、PORESET でリセットされます。(システム・リセットではリセットされません。) このレジスタはシステム・リセット中でもデバッガで書き込みができます。

参考資料

表 170. 融通性の高いトレース・ピン割り当て

DBGMCU_CR レジスタ		ピン配置モード	TRACE I/O ピン配置					
TRACE_I OEN	TRACE_ MODE[1:0]		PB3 / JTDO/ TRACESW O	PE2 / TRACE CK	PE3 / TRACE D[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	TRACESW O					解放 (GPIO として使用可)
1	01	同期トレース 1 bit		TRACE CK	TRACE D[0]			
1	10	同期トレース 2 bit	解放 ⁽¹⁾	TRACE CK	TRACE D[0]	TRACED [1]		
1	11	同期トレース 4 bit		TRACE CK	TRACE D[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。ただし、JTAG モードが使用されるときには、JTDO ピンとして使用されます。

注意: 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た後 2 クロック・サイクル後に HCLK に接続されます。

デバッガは TPIU の SPP_R (Selected Pin Protocol) レジスタの PROTOCOL[1:0] ビットに書込むことで、トレース・モードを設定しなければなりません。

- PROTOCOL=00: トレースポート・モード (同期モード)
- PROTOCOL=01 もしくは 10: シリアル・ワイヤ (マンチェスター、もしくは NRZ) モード (非同期モード)。デフォルト値は 01 です。

次に、TPIU の CPSPS_R (カレント同期ポート・サイズ・レジスタ) のビット [3:0] で、トレースポート・サイズを設定しなければなりません。

- 0x1: 1 ピン (デフォルト状態)
- 0x2: 2 ピン
- 0x4: 4 ピン

26.16.3 TPIU フォーマッタ

フォーマッタ・プロトコルは 16 バイトフレームとしてデータを出力します。

- 7bytes、データ
- 8bytes、以下の内容をもちます。
 - 1bit (LSB)、DATA バイトを 0 で、ID バイトを 1 で表示します。
 - 7bit (MSB)、データ、もしくは新しいトレース元 ID
- 1byte、各ビットが 8bytes のそれぞれに対応する補助バイトで、以下の意味を持ちます。
 - 対応するバイトがデータであれば、このビットはデータバイトのビット 0 を保持しています。
 - 対応するバイトが ID 変更であれば、このビットはその ID の変化が有効であることを示しています。

注意: 更なる情報に関しては、*ARM CoreSight Architecture Specification v1.0 (ARM IHI 0029B)* を参照してください。

STM32F10xxx MCU フォーマッタの使用

STM32F10xxx MCU では、TRACE 元として ITM 一種類のみが存在します。しかし、フォーマッタをディセーブルすることはできず、TRACECTL ピンが配置されていないため、バイパス・モードを使用しなければなりません。Trace Port Analyzer はトリガの位置を識別するため、フォーマッタ・プロトコルの一部をデコードしていることがあります。

26.16.4 TPIU フレーム同期パケット

TPIU は 2 種の同期パケットを発生することができます。

- フレーム同期パケット（フルワード同期パケット）

パケットはワード 0x7F FF FF FF (LSB から先に送出) から構成されています。このシーケンスは発行元 ID コード 0x7F が使用されていない限り発行されることはありません。

このパケットはフレーム間に周期的に出力されます。

連続モードでは TPA は同期フレームが見出されたフレームを破棄しなければなりません。
- ハーフワード同期パケット

このパケットはハーフワード 0x7F FF (LSB から先に送出) から構成されています。

パケットはフレーム間、及びフレーム内で周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、TRACE ポートが IDLE モードにある (TRACE がキャプチャされていない) ことを TPA が検出することをイネーブルとします。TPA がパケットを検出すると、このパケットは捨てられます。

26.16.5 同期フレームパケットの発行

コアの TPIU には同期カウンタ・レジスタは組み込まれていません。したがって、同期トリガは DWT によってのみ生成されます。DWT 制御レジスタ (SYNCTAP[11:10] ビット) と DWT カレント PC サンプラー・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。

- TPIU リセットの解除の際、その都度。このリセットは TRACECLKIN クロックの立上がりエッジに同期してリリースされます。このためパケットは DBGMCU_CFG レジスタの IO_TRACEN ビットがセットされるごとに一度ずつ発行されます。このとき、ワード 0x7F FF FF FF にフォーマットされたパケットが続くことはありません。
- DWT トリガごとに。(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いるとき、ワード 0x7F_FF_FF_FF が他のフォーマットされた情報を含むことなく送出されます。
 - ITM の SYNENA ビットがセットされているとき、ITM 同期パケットは TPIU でフォーマットされた 0x80 00 00 00 00 00 を伴います。

26.16.6 同期モード

トレースデータの出力サイズが 4、2、1 ピンから選択ができます。(TRACED[3:0])

出力クロックはデバッガに対して出力されます。(TRACECK)

なお、TRACECLKIN は内部で駆動され、TRACE が使用されるときのみ HCLK に接続されます。

注意: 同期モードでは、必ずしもクロック周波数が安定している必要はありません。

TRACE I/O (TRACECK を含む) は TRACELKIN (HCLK と同等) の立上がりエッジで駆動されます。このため、TRACECK の出力周波数は HCLK/2 となります。

26.16.7 非同期モード

これは 1 ピン (非同期出力ピン TRACESWO ピン) 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、そのバンド幅は制限されます。

TRACESWO ピンは SW-DP ピンを使用するときには JTDO ピンと共に用されています。このモードはすべての STM32F10xxx ピッケージで提供されています。

この非同期モードでは TRACELKIN の周波数は一定していかなければなりません。標準的な UART (NRZ) の受信機構では、5 % の周波数精度が必要とされます。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。

26.16.8 STM32F10xxx 内部の TRACELKIN 接続

STM32F10xxx では、この TRACELKIN 入力は内部で HCLK に接続されています。このため、非同期トレース・モードでは、アプリケーションのタイムフレームの使用は CPU 周波数が安定しているときに限られます。

注意: 重要: 同期トレースを使用するとき、次の点に留意しなければなりません。

STM32F10xxx MCU のデフォルトクロックは内部 RC 発振回路から供給されています。リセット下の周波数はリセット解除後の周波数と異なっています。これは、リセット中は

RC 補正值としてデフォルト値が使用されるのに対して、リセット後にその値が更新されるためです。

したがって、リセット解除後に送出されるトレースパケットと異なるビットタイムを持つ同期フレームパケットが挿入されることになるため、*Trace Port Analyzer (TPA)* がシステム・リセット中にトレースをイネーブルとする (*IOTRACEN* ビット) ことは避けるべきです。

26.16.9 TPIU レジスタ

TPIU APB レジスタはデバッグ例外・モニタ制御レジスタ (DEMCR) の TRCENA ビットがセットされているときにのみ読書きができます。このビットがセットされていないときは、レジスタの読み出しへでは 0 が戻されます。(このビットの出力は TPIU の PCLK をイネーブルとします。)

表 171. 重要な TPIU レジスタ

アドレス	レジスタ	摘要
0xE0040004	カレントポート サイズ	トレースポート・サイズの選択 Bit 0: ポート・サイズ = 1 Bit 1: ポート・サイズ = 2 Bit 2: ポート・サイズ = 3 サポートしません。 Bit 3: ポート・サイズ = 4 1 個のビットのみがセットできます。デフォルトではポート・サイズは 1 ビット (設定値は 0x00000001) です。
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 Bit 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 - マンチェスター (デフォルト値) 10: シリアル・ワイヤ出力 - NRZ 11: 予約値
0xE0040304	フォーマッタ・ Flash 制御	Bit 31-9 = 常時 0 Bit 8 = TrigIn: = 常時 1 で、トリガが表示されることを示します。 Bit 7-4 = 常時 0 Bit 3-2 = 常時 0 Bit 1 = EnFCont: 同期トレース・モード (選択ピンプロトコルレジスタ、ビット 1:0 = 00) では、このビットは強制的に 1 とされています。フォーマッタは自動的に連続モードでイネーブル状態とされます。 非同期モード (選択ピンプロトコルレジスタ、ビット 1:0 < 00) では、フォーマッタをアクティブ / インアクティブするために書き込みを行うことができます。 Bit 0 = 常時 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 ピンがチップ外に引き出されていませんので、フォーマッタは常に連続モードでイネーブル状態となります。このためフォーマッタはトレースパケットの元を示す制御パケットを挿入します。
0xE0040300	フォーマッタ・ Flash 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。

参考資料

26.16.10 設定例

- デバッグ例外・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込みます。(デフォルト値は 0x1 でポート・サイズは 1 ビットとなります。)
- TPIU フォーマッタ・Flash 制御レジスタに 0x102 (デフォルト値) を書込みます。
- TPIU 選択ピンプロトコルレジスタの書き込みで同期 / 非同期モードを選択します。例えば設定値 0x2 で非同期・NRZ モード (UART 類似) が指定されます。
- 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) が送出されます。
- ITM を設定し、ITM トリガ・レジスタに出力する値を設定します。

参考資料

26.17 DBG レジスタ・マップ

次の表はデバッグレジスタの一覧です。

表 172. DBG - レジスタ・マップとリセット時の値

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	0	0	0	0	0	0	0	0	0	0	0			0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xE0042004	DBGMCU_CR	予約済み												DBG_I2C2_SMBUS_TIMEOUT	DBG_I2C1_SMBUS_TIMEOUT	DBG_CAN_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP	DBG_TIM1_STOP	DBG_WWDG_STOP	DBG_IWDG_STOP	TRACE_MODE [1:0]	TRACE_OEN	予約済み	DBG_STANDBY	DBG_STOP	DBG_Sleep	0	0	0	0	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

参考資料

改版履歴

RM0008

27 改版履歴

本書は、「RM0008 Rev 5（2008年7月28日発行）」を日本語に翻訳したものです。次の表に本データシートの改版履歴を示します。

表 173. 改版履歴

日付	版	変更内容
2007年6月1日	Rev. 1	英語版 発行

参考資料

RM0008

改版履歴

表 173. 改版履歴 (続き)

日付	版	変更内容
2007 年 10 月 19 日	Rev. 1	<p>ドキュメントリファレンス番号を UM0306 から RM008 へ変更。2007 年 1 月 1 日発行の 1 版から下記を変更。</p> <p>EXTSEL[2:0] 及び JEXTSEL[2:0] が 表 41: ADC ピン (158 ページ) から削除され、備考欄の V_{REF+} 範囲が変更。</p> <p>セクション 10.3.9 (161 ページ)、セクション 10.9.2 (169 ページ)、セクション 10.9.7 (172 ページ) とセクション 10.9.9 (173 ページ) へ注意事項が追加。</p> <p>1 本のクロック線と 1 本の両方向データ線 (576 ページ) で SPI_CR2 が SPI_CR1 に接続。</p> <p>セクション 22.2: SPI と μS の主要機能 (568 ページ) で f_{CPU} 周波数が f_{PCLK} へ変更。セクション 22.3.6: CRC の計算 (577 ページ) 及びセクション 22.3.7: DMA (<i>direct memory addressing</i>) を使用する SPI 通信 (578 ページ) が変更。</p> <p>セクション 22.5.1: SPI 制御レジスタ 1 (SPI_CR1) (μS モードでは使用しません。) (594 ページ) でビット 13 の記述へ追加された注意が変更。セクション 22.5.3: SPI ステータス・レジスタ (SPI_SR) (596 ページ) でビット 4 の注意事項が変更。</p> <p>64 ピンのパッケージ (47 ページ) が変更。セクション 7.3.2: GPIO ポート PD0/PD1 を OSC_IN/OSC_OUT ピンとして使用するには (116 ページ) が更新。図 2: メモリ・マップ (35 ページ) and 表 1: レジスタ・バンクアドレスでアドレス 0x4000 6000 の SRAM 記述が変更。セクション 20.2: USB の主要機能 (489 ページ) 及びセクション 21.2: bxCAN の主要機能 (521 ページ) で注意事項が追加。</p> <p>図 4: 電源系の概要及び 100 ピン及び 144 ピンのパッケージ が変更。CAN ビット・タイミング・レジスタ (CAN_BTR) (552 ページ) でビット 25 : 24 の記述に式が追加。</p> <p>セクション 9.3: DMA の機能詳細 (142 ページ) が変更。設定例 (700 ページ) が変更。</p> <p>セクション 7.2.2: ポート構成レジスタ (上位) (GPIOx_CRH) ($x=A..G$) (107 ページ) の MODEx[1 : 0] ビットの定義が訂正。</p> <p>ダウンカウント・モード (223 ページ) が変更。</p> <p>図 76: キャプチャ/比較チャネル (チャネル 4) の出力ステージ (235 ページ) 及び 図 78: 出力比較モード、OC1 出力のトグル が変更。セクション 12.3.10: PWM モード (240 ページ) で OCx 出力イネーブル条件の定義が変更。</p> <p>セクション 12.3.19: TIMx と外部トリガ同期 (255 ページ) のタイトルが変更。</p> <p>セクション 12.4.7: キャプチャ/比較モード・レジスタ 1 (TIMx_CCMR1) 及び セクション 12.4.8: キャプチャ/比較モード・レジスタ 2 (TIMx_CCMR2) で OC1S、CC2S、CC3S と (1, 1) ビット設定用に変更された CC4S の定義が変更。</p> <p>セクション 13.4.7: キャプチャ/比較モード・レジスタ 1 (TIMx_CCMR1) 及び セクション 13.4.8: キャプチャ/比較モード・レジスタ 2 (TIMx_CCMR2) で OC1S、CC2S、CC3S と (1, 1) ビット設定用の CC4S 定義が変更。</p> <p>表 34: AFIO レジスタ・マップとリセット時の値 (129 ページ) で AFIO_EVCR ピンが変更。セクション 12.3.6: 入力キャプチャ・モード (236 ページ) が変更。</p>



参考資料

改版履歴

RM0008

表 173. 改版履歴 (続き)

日付	版	変更内容
2007 年 10 月 19 日 (続き)	Rev. 1 (続き)	<p>図 109: 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 (TIMx_ARR=0x6) 及び図 124: 出力比較モード、OC1 出力のトグルが変更。セクション 13.4.1: 制御レジスタ 1 (TIMx_CR1) の CKD 定義が変更。セクション 5.4.2: RTC クロック較正レジスタ (BKP_RTCCCR) にビット 8 とビット 9 が追加。</p> <p>DBGMCU_CR (702 ページ)、セクション 23.5: μC デバッグ・モード (623 ページ) にビット 15 とビット 16 が追加。</p> <p>表 6: 低電力モード要約の STOP と STANDBY が変更。</p> <p>表 8: Sleep-on-exit が変更。デバッグ・モード (57 ページ) が変更。</p> <p>セクション 6.3.1: クロック制御レジスタ (RCC_CR) の HSITRIM[4:0] ビットが変更。セクション 6.3.2: クロック構成レジスタ (RCC_CFGR) の MCO 定義の注意事項が変更。RCC - レジスタ・マップとリセット時の値 (101 ページ) の RCC_CR 列が変更。</p> <p>セクション 7.2.6: ポートビット・リセット・レジスタ (GPIOx_BRR) ($x=A..G$) のビット 15 : 0 記述が変更。内蔵ブートローダ (42 ページ) が追加。</p> <p>図 10、図 12、図 13、図 14 及び図 15 が変更。</p> <p>セクション 2.3.4: 内蔵 Flash メモリ (39 ページ) が変更。</p> <p>DBGMCU_CR (702 ページ) に REV_ID ビット記述が追加。</p> <p>クロック制御レジスタ (RCC_CR) (80 ページ) のリセット値が変更され HSITRIM[4:0] 記述が変更。</p> <p>セクション 7.1.1 (105 ページ) が変更。セクション 7.2: GPIO レジスタ (111 ページ) のビット定義が変更。表 9: STOP モードの起動遅延の記述が変更。</p> <p>クロック制御レジスタ (RCC_CR) のリセット値が変更。</p> <p>5.4.2 (66 ページ) の ASOS 及び ASOE ビット記述に注意事項が追加。</p> <p>セクション 26.15.2: タイマ、ウォッチドッグ、bxCAN と μC に対するデバッグ・サポートが変更。表 172: DBG - レジスタ・マップとリセット時の値が更新。</p> <p>セクション 20.5.3: バッファ・ディスクリプタ・テーブルを明示。</p> <p>センターライン・モード (アップ/ダウンカウント) (226 ページ) 及びセンターラインモード (アップ/ダウンカウント) (295 ページ) が更新。</p> <p>図 80: センターライン PWM 波形 (ARR=8) (242 ページ) 及び図 126: センターライン PWM 波形 (ARR=8) (306 ページ) が変更。</p> <p>セクション 10.12.3: ADC 制御レジスタ 2 (ADC_CR2) の RSTCAL 記述が変更。</p> <p>表 62: ウォッチドッグ・タイムアウト時間 (40 kHz クロック入力時) の下の注意事項が変更。図 8: クロック構成図の下に注意事項が追加。</p> <p>セクション 10.2: ADC の主な機能の ADC 変換時間が変更。</p> <p>自動インジェクション (Auto injection) (162 ページ) が更新。</p>

参考資料

RM0008

改版履歴

表 173. 改版履歴 (続き)

日付	版	変更内容
2007 年 10 月 19 日 (続き)	Rev. 1 (続き)	<p>セクション 10.9.9: 同時インジェクト + インタリーブに注意事項が追加。セクション 7.3.2: GPIO ポート PD0/PD1 を OSC_IN/OSC_OUT ピンとして使用するには注意事項が追加。若干のテキスト変更。内部 LSI RC 周波数が 32 から 40 kHz に変更。表 62: オッヂドッグ・タイムアウト時間 (40 kHz クロック入力時) が更新されました。図 2: メモリ・マップと表 2: Flash モジュールの構成 (中容量デバイス) でオプションバイト・アドレスが訂正。セクション 2.3.4: 内蔵 Flash メモリで情報ブロック構成に変更。</p> <p>ADC 変換をトリガする外部イベントは外部割込みではなく、EXTI ラインです (詳細はセクション 10: A/D コンバータ (ADC) を参照) Appendix A: 重要事項 (500 ページ) が追加。</p>
2007 年 11 月 20 日	Rev. 2	<p>図 237: USART ブロック図を変更。</p> <p>文字データの受信 (645 ページ) の手順を変更。</p> <p>セクション 24.3.4: ポーレート生成回路内で、</p> <ul style="list-style-type: none">- 方程式記号を変更。- 表 152: ポーレート設定と周波数の誤差を変更。- 注意事項を追加。 <p>テキストの変更。CAN ビット・タイミング・レジスタ (CAN_BTR) (552 ページ) で、ビット 15 を予約済みに変更。</p> <p>Flash メモリ構成を訂正。表 2: Flash モジュールの構成 (中容量デバイス) がセクション 2.3.4: 内蔵 Flash メモリで変更。</p> <p>セクション 4.1: 電源供給の図 4: 電源系の概要の下に注意事項を追加。</p> <p>バックアップ・ドメイン制御レジスタ (RCC_BDCR) の RTCSEL[1:0] ビット記述を変更。</p> <p>表 13: RCC - レジスタ・マップとリセット時の値の RCC_APB1RSTR 及び RCC_APB1ENR のビット [0 : 2] の名前を訂正。</p> <p>セクション 22.5.1: SPI 制御レジスタ 1 (SPLCR1) (μS モードでは使用しません。) の BR[2:0] 記述を訂正。</p> <p>アップデート・イベント発生時のプリスケーラ・バッファの動作の記載を追加。(アップカウント・モード (288 ページ)、ダウンカウント・モード (291 ページ) 及びセンターライン・モード (アップ/ダウンカウント) (293 ページ) を参照のこと。)</p> <p>セクション 10.12.2: ADC 制御レジスタ 1 (ADC_CR1) の AWDCH[4:0] を変更。セクション 10.12.4: ADC サンプル・タイム・レジスタ 1 (ADC_SMPR1) のビット [26:24] を予約済みに変更。</p> <p>表 144: bxCAN - レジスタ・マップとリセット時の値の CAN_BTR ビット 8 を予約済みに変更。CAN マスター制御レジスタ (CAN_MCR) (541 ページ) を訂正。</p> <p>表 41: ADC ピン及び 100 ピン及び 144 ピンのパッケージ (49 ページ) の V_{REF+} 範囲を訂正。</p> <p>START 信号 (Start condition) (609 ページ) を更新。表 16: CAN オルタネート機能のピン割当て変更の注意事項を削除。表 24: タイマ 4 オルタネート機能のピン割当てに注意事項を追加。</p>



参考資料

改版履歴

RM0008

表 173. 改版履歴 (続き)

日付	版	変更内容
2007 年 11 月 20 日 (続き)	Rev. 2 (続き)	<p>セクション 7.4.2: オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR) で USART2_REMAP = 0 のビット定義が変更。</p> <p>セクション 7.4.3: 外部割込み構成レジスタ 1 (AFIO_EXTICR1) で SPI1_REMAP = 0 のビット定義が変更。</p> <p>表 171: 重要な TPIU レジスタの 0xE0040004 でビット 2 のセットはサポートしません。</p> <p>TPIU トレース・ピン配置 (705 ページ) の TRCAE ポート・サイズ設定が訂正。図 10、図 12、図 13、図 14 及び図 15 が変更。図 11: 5V 耐性入出力ポートの基本構造が追加。</p> <p>表 7.3.1: GPIO ポート PC14/PC15 を OSC32_IN/OSC32_OUT ピンとして使用するには (116 ページ) が追加。</p> <p>セクション 15.4.5 とセクション 15.4.6 でビット記述が変更。</p> <p>セクション 26.6.2: バウンダリ・スキャン TAP (690 ページ) の JTAG ID コードが訂正。</p> <p>変更: セクション 17.2: WWDG の主な機能、セクション 5.2: 主な機能、セクション 5.3.1: タンパー検出、セクション 5.3.2: RTC 周波数調整、セクション 20.3: USB の機能詳細、ダウンカウンタを制御するには: (374 ページ)、セクション 4.1.2: バッテリ・バックアップ・ドメイン、セクション 8.2: はじめに</p> <p>セクション 5.4.2: RTC クロック較正レジスタ (BKP_RTCCR) の ASOE ビット記述が変更。</p>
2008 年 1 月 1 日	Rev. 1 – 日本語版	日本語翻訳版 発行
2008 年 5 月 22 日	Rev. 4	<p>512KbyteまでのFlashメモリ(大容量デバイス)を持つデバイスに対応するためにリファレンスマニュアルが更新。ドキュメントが再編成されました。若干のテキスト変更。中容量及び大容量デバイスの定義が全セクションに追加。</p> <p>セクション 2: メモリとバスの構成 (33 ページ) 内で</p> <ul style="list-style-type: none">図 1: システム・アーキテクチャ (33 ページ)、図 2: メモリ・マップ (35 ページ)、表 1: レジスタ・バウンダリ・アドレス (36 ページ) が更新。AHB/APB ブリッジ (APB) (34 ページ) に注意事項とテキストがを追加。セクション 2.3.2: 内蔵 SRAM (38 ページ) の SRAM サイズセクション 2.3.4: 内蔵 Flash メモリ (39 ページ) は (フラッシュサイン、ページサイズ、ページ数、Flash メモリの読み出し、表 3: Flash モジュールの構成 (大容量デバイス) (40 ページ) 追加) を更新。Flash メモリの読み出しにプリフェッч・バッファのオンオフを記載。 <p>セクション 2.3.3: ビット・バンディング (38 ページ) のビット番号定義を変更。</p> <p>セクション 3: CRC 計算ユニット (44 ページ) を追加。(表 1: レジスタ・バウンダリ・アドレス (36 ページ) 更新、図 2: メモリ・マップ (35 ページ) 更新、そしてセクション 6.3.6: AHB ペリフェラル・クロック・イネーブル・レジスタ (RCC_AHBENR) (91 ページ) に CRCEN ビットを追加)</p> <p>Stop モードに入るには (52 ページ) を記載。</p> <p>セクション 5: バックアップレジスタ (BKP) (59 ページ) の更新: バックアップ・レジスタ数、利用可能メモリ・サイズ及びセクション 5.1: 概要、セクション 5.4.2: RTC クロック較正レジスタ (BKP_RTCCR) (66 ページ) の ASOE 定義の変更。</p>

参考資料

RM0008

改版履歴

表 173. 改版履歴 (続き)

日付	版	変更内容
2008 年 5 月 22 日 (続き)	Rev. 4 (続き)	<p>セクション 6: リセットとクロックの制御 (RCC) (72 ページ) で :</p> <ul style="list-style-type: none">- LSI 周波数調整 (較正) (77 ページ) 追加。- 図 7: リセット回路 (73 ページ) 更新。- APB2 ペリフェラル・リセット・レジスタ (RCC_APB2RSTR) (87 ページ) 更新。- APB1 ペリフェラル・リセット・レジスタ (RCC_APB1RSTR) (89 ページ) 更新。- AHB ペリフェラル・クロック・イネーブル・レジスタ (RCC_AHBENR) 更新。- APB2 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB2ENR) 更新。- APB1 ペリフェラル・クロック・イネーブル・レジスタ (RCC_APB1ENR) (95 ページ) 更新。(セクション 表 13.: RCC - レジスタ・マップとリセット時の値を参照)- クロック割込みレジスタ (RCC_CIR) の LSERDYIE 定義変更。- クロック制御レジスタ (RCC_CR) の HSITRIM[4:0] 定義変更。 <p>セクション 7: 汎用入出力とオルタネート機能入出力 (GPIO、AFIO) (102 ページ) で :</p> <ul style="list-style-type: none">- GPIO ポート F 及び G 追加。- In セクション 7.3: オルタネート入出力機能とデバッグ構成 (AFIO) (116 ページ) で大容量デバイスのリマッピング追加、セクション 7.3.2 の下の注意事項変更、セクション 7.3.3 (116 ページ) 変更。- オルタネート機能の再割当てとデバッグ入出力構成レジスタ (AFIO_MAPR) (124 ページ) 更新。 <p>セクション 8: 割込みとイベント (130 ページ) : で更新 :</p> <ul style="list-style-type: none">- マスク可能割込みチャネル数。- GPIO 数 (図 17: 外部割込み / イベントの GPIO 割当てを参照) <p>セクション 9: DMA コントローラ (DMA) (141 ページ) で :</p> <ul style="list-style-type: none">- DMA コントローラ数と設定可能 DMA チャネル数更新。- 図 18: DMA ブロック構成 (142 ページ) を更新、注意事項追加。- セクション 9.3.2: アービタ (143 ページ) の注意事項更新。- セクション 9.3.5: 割込み (145 ページ) 注意事項更新。- 図 19: DMA1 リクエストのマッピング (146 ページ) 更新。- DMA2 コントローラ (147 ページ) 追加。 <p>セクション 10: A/D コンバータ (ADC) (156 ページ) で :</p> <ul style="list-style-type: none">- ADC3 追加 (図 21: シングル ADC ブロック図 (157 ページ) 更新、表 46: ADC3 のインジェクト・チャネルのための外部トリガ (166 ページ) 追加、他。) <p>セクション 11: D/A コンバータ (DAC) (191 ページ) 追加。</p> <p>セクション 12: 高機能制御タイマ (TIM1 及び TIM8) (215 ページ) で :</p> <ul style="list-style-type: none">- 高性能タイマ TIM8 追加。(図 47: 高機能制御タイマ・ブロック図 (217 ページ) 参照)- セクション 12.4.3: スレーブ・モード制御レジスタ (TIMx_SMCR) (262 ページ) で TS[2:0] 変更。 <p>セクション 13: 汎用タイマ (TIMx) (285 ページ) で :</p> <ul style="list-style-type: none">- TIM5 追加。- 図 95: 汎用タイマのブロック図 (286 ページ) 更新。- 表 57: TIMx 内部トリガ接続 (325 ページ) 変更。



参考資料

改版履歴

RM0008

表 173. 改版履歴 (続き)

日付	版	変更内容
2008 年 5 月 22 日 (続き)	Rev. 4 (続き)	<p>セクション 14: ベーシック・タイマ (TIM6 及び TIM7) (342 ページ) 追加。</p> <p>セクション 15.2: RTC の主な機能 (355 ページ) に RTC クロック・ソースを記載。セクション 15.1: RTC の概要変更。</p> <p>セクション 18: フレキシブル・スタティック・メモリ・コントローラ (FSMC) (380 ページ) 追加。</p> <p>セクション 19: SDIO インタフェース (SDIO) (432 ページ) 追加。</p> <p>図 204: CAN フレーム (538 ページ) 変更、CAN 送信メールボックス ID レジスタ (CAN_TIxR) ($x=0..2$) (553 ページ) のビット 31 : 21 及びビット 20 : 3 変更、CAN 受信 FIFO メールボックス ID レジスタ (CAN_RIxR) ($x=0..1$) (557 ページ) のビット 31 : 21 及びビット 20 : 3 変更、セクション 23.3.7: DMA リクエスト (618 ページ) 変更、セクション 23.6.2: 制御レジスタ 2 (I2C_CR2) (625 ページ) の DMAEN ビット 11 記述変更、クロックの位相と極性 (572 ページ) 変更、送信動作 (584 ページ) 変更、受信動作 (575 ページ) 追加、受信手順 (591 ページ) 変更、アンダーランフラグ (UDR) (593 ページ) 変更。</p> <p>I²S 機能追加。(セクション 22: SPI (Serial peripheral interface) 回路 (577 ページ) を参照)</p> <p>セクション 26: デバッグ・サポート (DBG) (682 ページ) で:</p> <ul style="list-style-type: none">- DBGMCU_IDCODE (689 ページ) and DBGMCU_CR (702 ページ) 更新。- TMC TAP をバウンダリ・スキャン TAP へ変更。- セクション 26.15.3: デバッグ MCU 構成レジスタ (702 ページ) の DBGMCU_CR へ配置されるアドレスの変更。 <p>セクション 25: デバイス電子署名 (679 ページ) 追加。</p> <p>セクション 26.6.1: MCU デバイス ID コード (689 ページ) で REV_ID(15 : 0) 定義変更。</p>

参考資料

表 173. 改版履歴 (続き)

日付	版	変更内容
2008 年 7 月 28 日	Rev. 5	<p>セクション 3.2: 主な機能 (44 ページ) の説明した多項式を更新しました。図 4: 電源系の概要 (48 ページ) 変更。</p> <p>セクション 4.1.2: バッテリ・バックアップ・ドメイン (49 ページ) 変更。</p> <p>セクション 6.2.5: LSI クロック (77 ページ) 記載。</p> <p>セクション 7.1.4: オルタネート機能 (AF) (106 ページ) を明確化。表 26: タイマ 2 オルタネート機能のピン割当て (119 ページ) へ注意事項を追加。</p> <p>セクション 9.4.2: DMA 割込みフラグ・クリア・レジスタ (DMA_IFCR) (150 ページ) のビットは書き込み専用。</p> <p>セクション 10.3.1: ADC 電源の制御 (158 ページ) のレジスタ名が変更。</p> <p>セクション 10.10: 温度センサ (174 ページ) に推奨サンプリング時間が記載。</p> <p>セクション 10.12.1: ADC ステータス・レジスタ (ADC_SR) (176 ページ) のビット属性が変更。</p> <p>セクション 10.12.4: ADC サンプル・タイム・レジスタ 1 (ADC_SMPR1) (183 ページ) のビット 23 : 0 の注意事項が変更。</p> <p>セクション 11.2: DAC の主な機能 (191 ページ) に注意事項が追加。</p> <p>セクション 11.3.5: DAC 出力電圧 (195 ページ) の式が更新。</p> <p>セクション 12.3.19: TIMx と外部トリガ同期 (255 ページ) の DBL[4 : 0] 記述が変更。</p> <p>図 77 (237 ページ) and 図 123 (302 ページ) が変更。</p> <p>セクション 22.5.3: SPI ステータス・レジスタ (SPI_SR) (596 ページ) が変更。</p> <p>マスター レシーバ交信の終了 (612 ページ) が更新。</p> <p>セクション 23.6.8: クロック制御レジスタ (I2C_CCR) (634 ページ) に注意事項が追加。セクション 23.6.8 及びセクション 23.6.9 の TCK が T_PCLK1 で置き換え。</p> <p>図 256: USART 割込みマッピング (665 ページ) の OVR が ORE へ変更。</p> <p>セクション 24.6.1: USART ステータス・レジスタ (USART_SR) (666 ページ) 更新。</p> <p>スレーブ選択 (NSS) ピンの管理 (571 ページ) が明確化。</p> <p>若干のテキスト変更。</p>
2009 年 2 月	Rev. 5 – 日本語版	日本語翻訳版 発行

機密保持義務 :

この文書には機密情報が含まれます。

この文書の配布については、機密保持契約書（NDA）を締結していることが条件となります。

「RESTRICTED DISTRIBUTION (配布限定資料)」として機密にお取扱いください。

この文書の取扱いに関して、お客様は常に以下のセキュリティ条件を遵守する責任を負うものとします。

（義務の詳細については NDA をご参照ください。）：

この文書の全部または一部の複写もしくは複製を禁じます。

この文書は、施錠できる場所で保管してください。

コピーが必要な場合、必要箇所に限定して配布することが可能ですので、お客様の最寄りの ST セールスオフィスへご連絡ください。

必ずお読み下さい :

本書記載の内容は ST 製品に関してのみ適用されるものです。STMicroelectronics NV およびその子会社（以下、ST）は、本書、本製品ならびに本書に記載されたサービスの内容を予告なく変更、修正、改定もしくは改良する権利を留保します。

すべての ST 製品は ST の販売条件に従って販売されます。

本書記載の ST 製品およびサービスの選択並びに使用については購入者が全ての責任を負うものとし、本書記載の ST 製品およびサービスの選択並びに使用に関して ST は一切の責任を負いません。

本書は、明示されているか否かに問わらず、また禁反言によるとよらずに問わらず、いかなる知的財産権の実施権を許諾するためのものではありません。本書で第三者の製品もしくはサービスに言及する場合、そのような言及は、ST が当該第三者の製品、サービスまたはそれらに含まれる知的財産権を利用する実施権を許諾したとみなすものではなく、またいかなる用法であれ当該第三者の製品、サービスまたはそれらに含まれる知的財産権の使用を保証するものではありません。

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默示の保証を含め、ST 製品の使用または販売に関する明示または默示の保証をすべて放棄します。

権限のある ST の代表者による書面での明示の許可がある場合を除き、ST 製品を軍事用、航空技術、宇宙用、救命用、若しくは生命維持用に用いること、または不具合若しくは誤動作により負傷または死亡事故、深刻な財産上または環境上の損害を招くおそれのある製品またはシステムへ使用することは、推奨、認可、保証されておりません。「自動車向け」として指定されていない ST 製品を自動車用に用いることは、使用者自身の責任においてなされるものとします。

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。

© 2009 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 – Singapore – Spain – Sweden – Switzerland – United Kingdom – United States of America

www.st.com (英語)

www.st-japan.co.jp (日本語)