

TOSHIBA

**32 ビット RISC マイクロコントローラ
TX03 シリーズ**

TMPM330FYWFG

Rev1.02

株式会社 **東芝**

セミコンダクター&ストレージ社

目次

1. 概要と特長	2
1.1 機能概要	2
1.2 ブロック図	5
2 ピン配置とピン機能	6
2.1 ピン配置図 (TOP VIEW)	6
2.2 ピン名称と機能	9
2.2.1 ピン番号順	9
2.2.2 ポート順	14
2.3 ピン名称と電源供給端子	19
2.4 ピン番号と電源供給端子	19
3 プロセッサコア	20
3.1 コアに関する情報	20
3.2 構成可能なオプション	20
3.3 例外/割り込み	20
3.3.1 割り込み本数	20
3.3.2 SysTick	21
3.3.3 SYSRESETREQ	21
3.3.4 LOCKUP	21
3.3.5 補助フォールトステータスレジスタ	21
3.4 イベント	21
3.5 電力管理	22
3.6 排他アクセス	22
3 割り込み優先度ビット数	21
4. デバッグインターフェース	23
4.1 仕様概要	23
4.2 SWJ-DP 特長.....	23
4.3 ETM 特長.....	23
4.4 端子情報	23
4.5 デバッグツールとの接続	24
5 メモリマップ	25
5.1 メモリマップ	26

6	リセット動作	27
6.1	コールドリセット時	27
6.2	ウォームリセット時	28
6.2.1	リセット期間	28
6.3	リセット解除後	28
7.	クロック/モード制御	29
7.1	特長	29
7.2	レジスタ説明	30
7.2.1	レジスタ一覧	30
7.2.2	レジスタ詳細	31
7.3	クロック制御	36
7.3.1	クロックの種類	36
7.3.2	リセット動作による初期値	36
7.3.3	クロック逡倍回路 (PLL)	38
7.3.4	ウォーミングアップ機能	39
7.3.5	システムクロック	41
7.3.6	プリスケーラクロック	41
7.3.7	システムクロックの端子出力機能	42
7.4	モードとモード遷移	43
7.4.1	モード状態遷移	43
7.5	動作モード	44
7.5.1	NORMAL モード	44
7.5.2	SLOW モード	44
7.6	低消費電力モード	44
7.6.1	IDLE モード	45
7.6.2	SLEEP モード	45
7.6.3	STOP モード	45
7.6.4	低消費電力モードの選択	47
7.6.5	各モードにおける動作状態	47
7.6.6	低消費電力モードの解除	48
7.6.7	ウォーミングアップ	49
7.6.8	モード遷移によるクロック動作	50
8	例外	52
8.1	概要	52
8.1.1	種類	52

8.1.2	処理の流れ	53
8.2	リセット例外	60
8.3	マスク不能割り込み (NMI)	60
8.4	SYSTICK	61
8.5	割り込み	62
8.5.1	要因	62
8.5.2	処理詳細	66
8.6	例外/割り込み関連レジスタ	73
8.6.1	レジスタ一覧	73
8.6.2	NVIC レジスタ	74
8.6.3	クロックジェネレータレジスタ	94
9	入出力ポート	101
9.1	ポートレジスタ概略説明	101
9.2	ポート機能詳細	102
9.2.1	STOPモード中のポート状態	102
9.2.2	STOP/SLEEPモード遷移の際の注意	102
9.2.3	ポートA (PA0~PA7)	103
9.2.4	ポートB (PB0~PB7)	105
9.2.5	ポートC (PC0~PC3)	107
9.2.6	ポートD (PD0~PD7)	108
9.2.7	ポートE (PE0~PE6)	109
9.2.8	ポートF (PF0~PF7)	111
9.2.9	ポートG (PG0~PG7)	113
9.2.10	ポートH (PH0~PH7)	115
9.2.11	ポートI (PI0~PI7)	117
9.2.12	ポートJ (PJ0~PJ7)	118
9.2.13	ポートK (PK0~PK2)	120
9.3	ポート回路図	122
9.3.1	ポートタイプ一覧	122
9.3.2	タイプT1	123
9.3.3	タイプT2	124
9.3.4	タイプT3	125
9.3.5	タイプT4	126
9.3.6	タイプT5	127
9.3.7	タイプT6	128
9.3.8	タイプT7	129

9.3.9	タイプT 8	130
9.3.10	タイプT 9	131
9.3.11	タイプT 10	132
9.3.12	タイプT 11	133
9.3.13	タイプT 12	134
9.3.14	タイプT 13	135
9.3.15	タイプT 14	136
9.3.16	タイプT 15	137
9.3.17	タイプT 16	138
9.3.18	タイプT 17	139
9.3.19	タイプT 18	140
10	16ビットタイマ/イベントカウンタ (TMRB)	141
10.1	概要	141
10.2	チャンネル別仕様相違点	142
10.3	構成	144
10.4	レジスタ説明	146
10.4.1	チャンネル別レジスタ一覧	146
10.5	回路別の動作説明	158
10.5.1	プリスケータ	158
10.5.2	アップカウンタ (UC0)	163
10.5.3	タイマレジスタ (TBORG0、TBORG1)	164
10.5.4	キャプチャ制御	164
10.5.5	キャプチャレジスタ (TBOCP0H/L、TBOCP1H/L)	165
10.5.6	アップカウンタキャプチャレジスタ (TBOUCH/L)	165
10.5.7	コンパレータ (CP0、CP1)	165
10.5.8	タイマフリップフロップ (TBOFF0)	165
10.5.9	キャプチャ割込み (INTCAP00、INTCAP01)	165
10.6	モード別動作説明	166
10.6.1	16ビットインタバルタイマモード	166
10.6.2	16ビットイベントカウンタモード	166
10.6.3	16ビットPPG (プログラマブル矩形波) 出力モード	166
10.7	タイマ同期モード	169
10.8	キャプチャ機能を利用した応用例	170
11	シリアル チャンネル (SIO/UART)	174
11.1	特長	174

11.2	ブロック図 (チャンネル0)	176
11.3	回路別の動作説明 (チャンネル0)	177
11.3.1	プリスケータ	177
11.3.2	ボーレート	182
11.3.3	シリアルクロック生成回路	186
11.3.4	受信カウンタ	186
11.3.5	受信制御部	186
11.3.6	受信バッファ	186
11.3.7	受信 FIFO バッファ	188
11.3.8	受信 FIFO の動作	188
11.3.9	送信カウンタ	190
11.3.10	送信制御部	190
11.3.11	送信バッファ	192
11.3.12	送信 FIFO BUFFER	193
11.3.13	送信 FIFO の動作	193
11.3.14	パリティ制御回路	195
11.3.15	エラーフラグ	195
11.3.16	データ転送方向	196
11.3.17	STOP ビットの長さ	196
11.3.18	ステータスフラグ	196
11.3.19	送受信バッファの構成	197
11.3.20	ソフトリセット	197
11.3.21	各信号発生タイミング	198
11.4	レジスタ説明 (チャンネル0 についてのみ説明します)	199
11.4.1	イネーブルレジスタ	199
11.4.2	バッファレジスタ	199
11.4.3	コントロールレジスタ	200
11.4.4	モードコントロールレジスタ 0	201
11.4.5	モードコントロールレジスタ 1	202
11.4.6	モードコントロールレジスタ 2	203
11.4.7	ボーレートジェネレータコントロールレジスタ (SCOBRCCR) ボーレートジェネレータコントロールレジスタ 2 (SCOBRADD)	205
11.4.8	FIFO コンフィグレジスタ	207
11.4.9	受信 FIFO コンフィグレジスタ	208
11.4.10	送信 FIFO コンフィグレジスタ	209
11.4.11	受信 FIFO ステータスレジスタ	210

11.4.12	送信 FIFO ステータスレジスタ	210
11.5	モード別動作説明	211
11.5.1	モード 0 (I/O インターフェースモード)	211
11.5.2	モード 1 (7 ビット UART モード)	221
11.5.3	モード 2 (8 ビット UART モード)	222
11.5.4	モード 3 (9 ビット UART)	223
12.	シリアルバスインタフェース (SBI)	225
12.1	構成	226
12.2	制御	227
12.3	I2C バスモード時のデータフォーマット	228
12.4	I2C バスモード時のコントロールレジスタ	229
12.5	I2C バスモード時の制御	237
12.5.1	アクノリッジメントモードの指定	237
12.5.2	転送ビット数の選択	237
12.5.3	シリアルクロック	237
12.5.4	スレーブアドレスとアドレス認識モードの設定	238
12.5.5	マスタ/スレーブの選択	238
12.5.6	トランスミッタ/レシーバの選択	239
12.5.7	スタート/ストップコンディションの発生	239
12.5.8	割り込みサービス要求と解除	240
12.5.9	シリアルバスインタフェースの動作モード	240
12.5.10	アービトレーションロスト検出モニタ	240
12.5.11	スレーブアドレス一致検出モニタ	241
12.5.12	ゼネラルコール検出モニタ	242
12.5.13	最終受信ビットモニタ	242
12.5.14	ソフトウェアリセット	242
12.5.15	シリアルバスインタフェースデータバッファレジスタ (SBIxDBR)	242
12.5.16	I2CBUS アドレスレジスタ (SBIxI2CAR)	242
12.5.17	IDLE 設定レジスタ (SBIxBR0)	242
12.6	I2C バスモード時のデータ転送手順	243
12.6.1	デバイスの初期化	243
12.6.2	スタートコンディション、スレーブアドレスの発生	243
12.6.3	1 ワードのデータ転送	244
12.6.4	ストップコンディションの発生	249
12.6.5	再スタートの手順	250
12.7	クロック同期式 8 ビット SIO モード時の制御	251

12.7.1	シリアルクロック	255
12.7.2	転送モード	257
13.	CEC機能	262
13.1	概要	262
13.1.1	受信	262
13.1.2	送信	262
13.1.3	注意事項	262
13.2	レジスタ説明	263
13.2.1	レジスタ一覧	263
13.2.2	CECイネーブルレジスタ《CECEN》	264
13.2.3	ロジカルアドレスレジスタ《CECADD》	264
13.2.4	ソフトウェアリセットレジスタ《CECRESET》	265
13.2.5	受信イネーブルレジスタ《CECEN》	266
13.2.6	受信バッファレジスタ《CECRBUF》	267
13.2.7	受信コントロールレジスタ1《CECRCR1》	268
13.2.8	受信コントロールレジスタ2《CECRCR2》	270
13.2.9	受信コントロールレジスタ3《CECRCR3》	271
13.2.10	送信イネーブルレジスタ《CECTEN》	273
13.2.11	送信バッファレジスタ《CECTBUF》	274
13.2.12	送信コントロールレジスタ《CECTCR》	275
13.2.13	受信割り込みステータスレジスタ《CECRSTAT》	277
13.2.14	送信割り込みステータスレジスタ《CECTSTAT》	278
13.2.15	CEC サンプリングクロック選択レジスタ《CECFSEL》	279
13.3	動作説明	280
13.3.1	受信	280
13.3.2	送信	291
13.3.3	ソフトウェアリセット	296
14	リモコン判定機能	297
14.1	概要	297
14.1.1	リモコン受信	297
14.2	レジスタ説明	298
14.2.1	レジスタ一覧	298
14.2.2	リモコンイネーブルレジスタ《RMCEN》	299
14.2.3	リモコン受信イネーブルレジスタ《RMCREN》	299
14.2.4	リモコン受信データバッファレジスタ1《RMCRBUF1》	300

14.2.5	リモコン受信データバッファレジスタ 2 《RMCRBUF2》	301
14.2.6	リモコン受信データバッファレジスタ 3 《RMCRBUF3》	301
14.2.7	リモコン受信コントロールレジスタ 1 《RMCRCR1》	302
14.2.8	リモコン受信コントロールレジスタ 2 《RMCRCR2》	304
14.2.9	リモコン受信コントロールレジスタ 3 《RMCRCR3》	306
14.2.10	リモコン受信コントロールレジスタ 4 《RMCRCR4》	307
14.2.11	リモコン受信ステータスレジスタ 《RMCSTAT》	308
14.3	動作説明	309
14.3.1	リモコン受信	309
15	アナログ/デジタルコンバータ	320
15.1	レジスタ一覧	321
15.2	レジスタ詳細	322
15.3	変換クロック	334
15.4	動作説明	335
15.4.1	アナログ基準電圧	335
15.4.2	アナログ入力チャネルの選択	335
15.4.3	A/D 変換開始	336
15.4.4	A/D 変換モードと A/D 変換終了割り込み	337
15.4.5	最優先変換モード	340
15.4.6	AD 監視機能	340
15.4.7	A/D 変換結果の格納と読み出し	340
15.4.8	データポーリング	341
16	ウォッチドッグタイマ (WDT)	342
16.1	構成	342
16.2	ウォッチドッグタイマ割り込み	343
16.3	コントロールレジスタ	344
16.3.1	ウォッチドッグ タイマ モードレジスタ (WDMOD)	344
16.3.2	ウォッチドッグ タイマ コントロールレジスタ (WDCR)	344
16.4	動作説明	346
17	リアルタイムクロック (RTC)	347
17.1	RTC の機能概略	347
17.2	ブロック図	347
17.3	コントロールレジスタ	348
17.4	コントロールレジスタの説明	349

17.5	動作説明	357
17.5.1	時計データをリードする場合	357
17.5.2	時計データをライトする場合	358
17.5.3	低消費電力モードへ遷移する場合	359
17.6	アラーム機能の説明	360
18.	FLASH 動作説明	362
18.1	フラッシュメモリ	362
18.1.1	特長	362
18.1.2	フラッシュ部ブロック図	363
18.2	動作モード	364
18.2.1	リセット動作	365
18.2.2	ユーザーブートモード (シングルチップモード)	366
18.2.3	シングルブートモード	373
18.3	オンボードプログラミングでのフラッシュメモリ書き込み/消去	405
18.3.1	フラッシュメモリ	405
19.	プロテクト/セキュリティ機能	422
19.1	概要	422
19.2	特長	422
19.2.1	内蔵ROM(Flash)のライト/消去プロテクト	422
19.2.2	セキュリティ機能	423
19.3	レジスタ	424
19.4	設定/解除方法	426
19.4.1	内蔵ROM(Flash)のライト/消去プロテクト	426
19.4.2	セキュリティビット	426
20.	電气的特性	427
20.1	絶対最大定格	427
20.2	DC 電气的特性 (1/2)	428
20.3	DC 電气的特性 (2/2)	429
20.4	10 ビット A/D コンバータ変換特性	430
20.5	AC 電气的特性	431
20.5.1	シリアルチャネルタイミング (SIO)	431
20.5.2	シリアルバスインターフェース (I2C)	433
20.5.3	イベントカウンタ	437
20.5.4	キャプチャ	437

20.5.5	外部割り込み (STOP 解除時以外)	437
20.5.6	外部割り込み (STOP 解除時)	437
20.5.7	SCOUT 端子 AC 特性	438
20.5.8	デバッグ通信	439
20.5.9	ETM トレース	440
20.6	発振回路	441
20.7	取り扱い上のご注意	442
20.7.1	電源の注意事項	442
20.7.1.1	電源投入時の注意事項	442
20.7.1.2	電源再投入時の注意事項	442
21.	ポート部等価回路図	443
22.	パッケージ寸法図	448
	製品取り扱い上のお願い	450

ARM, ARM Powered, AMBA, ADK, ARM9TDMI, TDMI, PrimeCell, RealView, Thumb, Cortex, Coresight, ARM9, ARM926EJ-S, Embedded Trace Macrocell, ETM, AHB, APB, and KEIL はARM LimitedのEUおよびその他の国における登録商標または商標です。



32 ビット RISC マイクロコントローラ TX03 シリーズ

TMPM330FYWFG

1. 概要と特長

TX03 シリーズは、ARM 社 Cortex™-M3 コアを内蔵した 32 ビット RISC マイクロコントローラです。TMPM330FYWFG の機能概要と特長は次のとおりです。

1.1 機能概要

(1) ARM 社製 Cortex-M3 コアを使用

①Thumb®-2 命令で、コード効率の向上を実現

- プログラムフロー改善のための新しい 16 ビット命令
- 性能とコードサイズ向上のための新しい 32 ビット命令
- 32 ビット命令/16 ビット命令の切り替えをコンパイラが自動で実行

②高性能化と低消費電力化を同時に実現

●高性能化

- 32 ビット乗算 ($32 \times 32 = 32$ ビット) を 1 クロックで実行
- 除算を 2 ~ 12 クロックで実行

●低消費電力化

- 低消費電力ライブラリを使用した最適化設計
- コアの動作を停止させるスタンバイ機能

③リアルタイム制御に向けた高速割り込み応答

- 実行時間の長い命令は割り込みで中断可能
- スタックへの PUSH をハードウェアで自動的に実行

(2) 内蔵プログラムメモリ／データメモリ

製品名	内蔵 Flash ROM	内蔵 RAM
TMPM330FYWFG	256Kbyte	16Kbyte

- (3) 16 ビットタイマ : 10 チャンネル
- 16 ビットインターバルタイマモード
 - 16 ビットイベントカウンタモード
 - 16 ビット PPG 出力
 - インプットキャプチャ機能
- (4) リアルタイムクロック (RTC) : 1 チャンネル
- 時計機能 (時間、分、秒)
 - カレンダー機能 (月日、週、うるう年)
 - +/-30 秒補正機能 (ソフトウェアによる補正)
 - アラーム機能 (アラーム出力)
 - アラーム割り込み発生
- (5) ウォッチドッグタイマ : 1 チャンネル
- ウォッチドックタイマアウト機能
- (6) 汎用シリアル・インタフェース : 3 チャンネル
- UART/同期式モード選択可能 (4byte FIFO 内蔵)
- (7) シリアルバスインタフェース : 3 チャンネル
- I²C バスモード/クロック同期式モード選択可能
- (8) CEC 機能 : 1 チャンネル
- 1 byte ごと送受信
- (9) リモコン判定機能 : 2 チャンネル
- 72bit まで一括受信
- (10) 10 ビット A/D コンバータ : 12 チャンネル
- 内部タイマトリガスタート/外部トリガスタートが可能
 - チャンネル固定/スキャンモード
 - シングル/リピートモード
 - AD 監視機能 2ch
 - 変換時間 1.15usec (f_{sys} = 40MHz 時)
- (11) 割り込み機能
- 内部 42 本 …… 7 レベルの優先順位設定可能 (ウォッチドッグタイマ割り込みを除く)
 - 外部 8 本 …… 7 レベルの優先順位設定可能
- (12) 入出力ポート
- 78 端子

-
- (13) スタンバイ機能
 - スタンバイモード : IDLE、SLOW、SLEEP、STOP
 - サブクロック動作 (32.768kHz) : SLOW、SLEEP

 - (14) クロックジェネレータ
 - PLL 内蔵 (4 通倍)
 - クロックギア機能: 高速クロックを 1/1, 1/2, 1/4, 1/8 に分周可能

 - (15) エンディアン
 - リトルエンディアン

 - (16) 最大動作周波数
 - 40MHz

 - (17) 動作電圧範囲
 - 2.7V~3.6V (内蔵レギュレータ使用)

 - (18) 温度範囲
 - -40°C~85°C (Flash W/E 時以外)
 - 0°C~70°C (Flash W/E 時)

 - (19) パッケージ
 - LQFP100-P-1414-0.50H (14mm × 14mm, 0.5mm ピッチ)
-

1.2 ブロック図

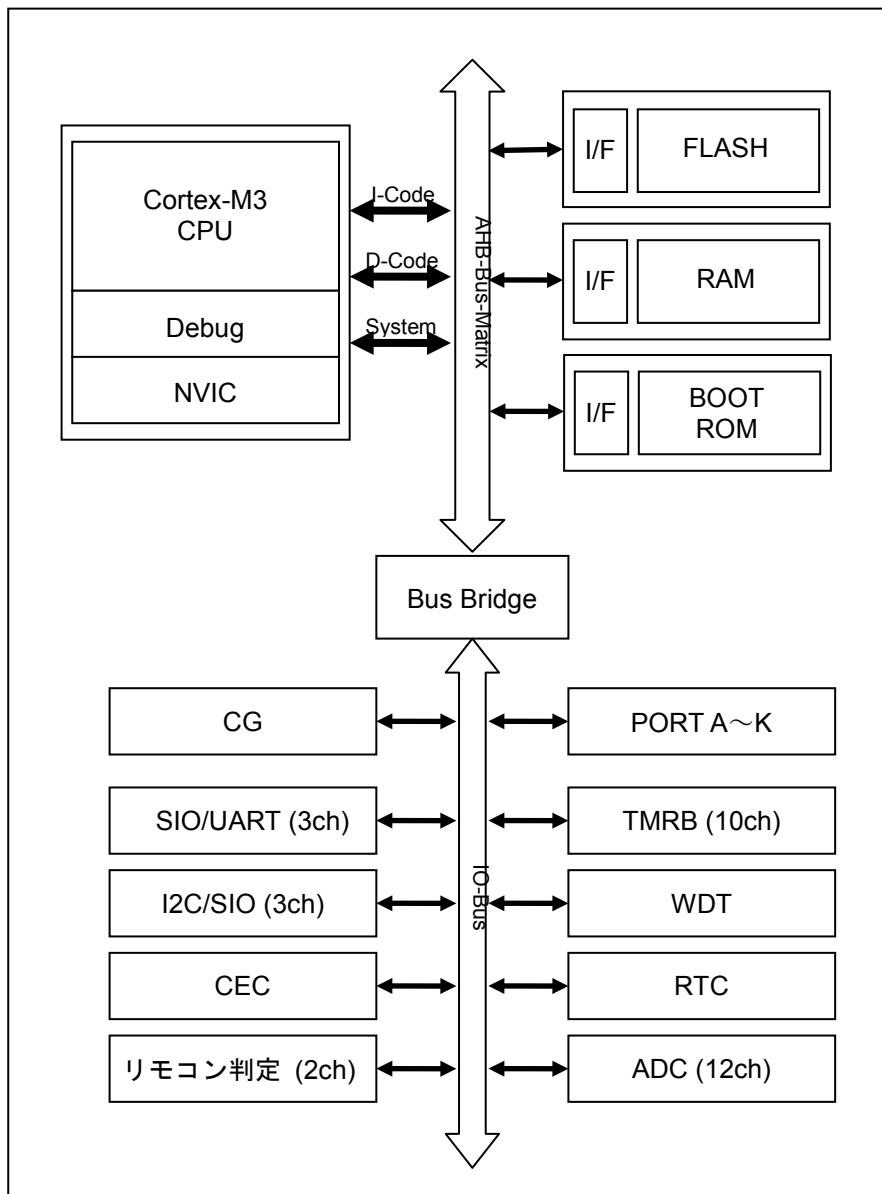


図 1-1 ブロック図

2 ピン配置とピン機能

TMPM330FYWFG のピン配置図および入出力ピンの名称と概略機能を示します。

2.1 ピン配置図 (Top view)

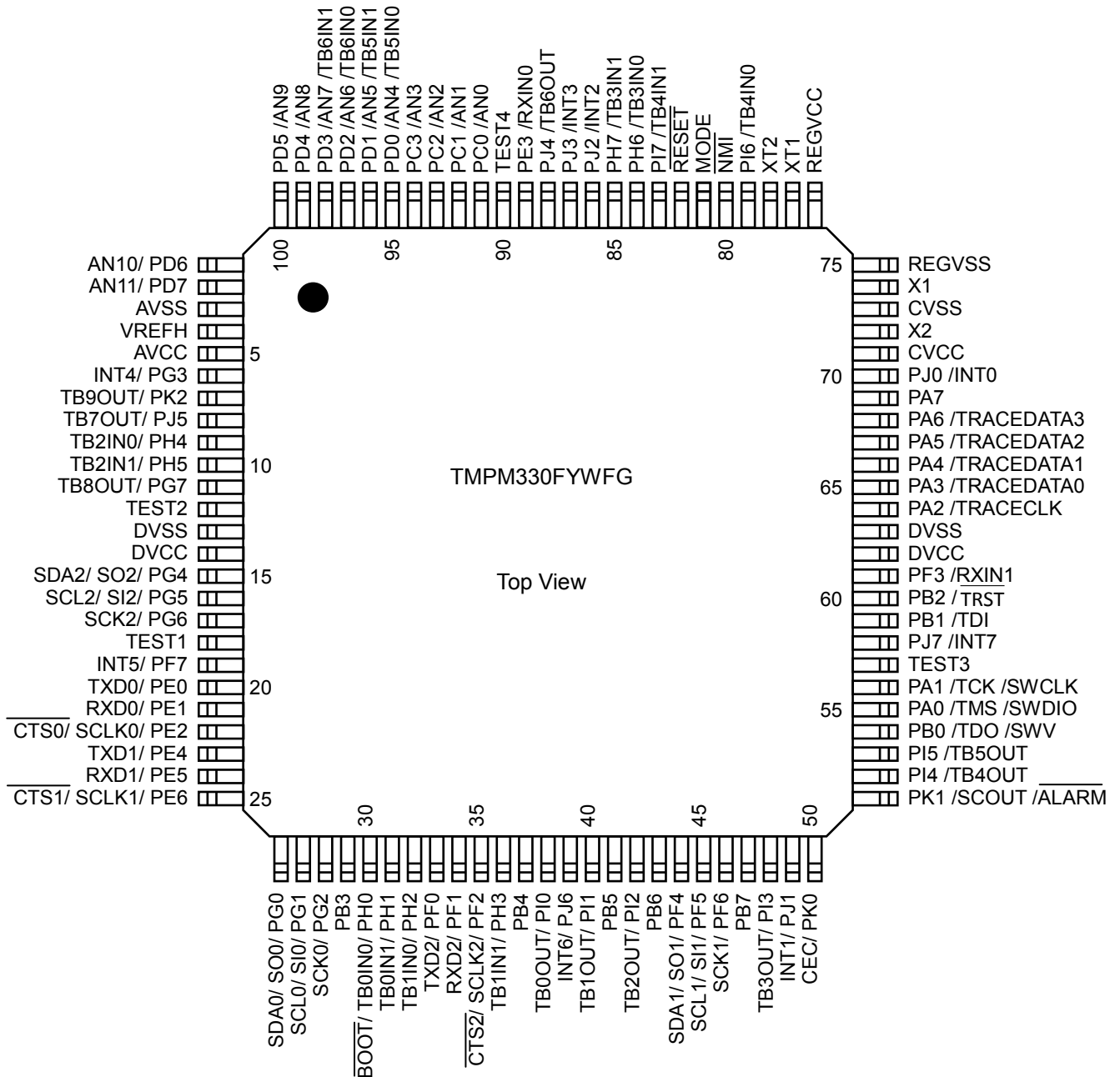


図 2-1 ピン配置図 (LQFP100)

表 2-1 ピンリスト(1/2)

ピン 番号	ピン機能	ピン 番号	ピン機能
1	PD6, AN10	26	PG0, S00, SDA0
2	PD7, AN11	27	PG1, SI0, SCL0
3	AVSS	28	PG2, SCK0
4	VREFH	29	PB3
5	AVCC	30	PH0, TB0IN0, $\overline{\text{BOOT}}$
6	PG3, INT4	31	PH1, TB0IN1
7	PK2, TB9OUT	32	PH2, TB1IN0
8	PJ5, TB7OUT	33	PF0, TXD2
9	PH4, TB2IN0	34	PF1, RXD2
10	PH5, TB2IN1	35	PF2, SCLK2, CTS2
11	PG7, TB8OUT	36	PH3, TB1IN1
12	TEST2	37	PB4
13	DVSS	38	PI0, TB0OUT
14	DVCC	39	PJ6, INT6
15	PG4, S02, SDA2	40	PI1, TB1OUT
16	PG5, SI2, SCL2	41	PB5
17	PG6, SCK2	42	PI2, TB2OUT
18	TEST1	43	PB6
19	PF7, INT5	44	PF4, S01, SDA1
20	PE0, TXD0	45	PF5, SI1, SCL1
21	PE1, RXD0	46	PF6, SCK1
22	PE2, SCLK0, CTS0	47	PB7
23	PE4, TXD1	48	PI3, TB3OUT
24	PE5, RXD1	49	PJ1, INT1
25	PE6, SCLK1, CTS1	50	PK0, CEC

表 2-1ピンリスト(2/2)

ピン 番号	ピン機能	ピン 番号	ピン機能
51	PK1, SCOUT, $\overline{\text{ALARM}}$	76	REGVCC
52	PI4, TB4OUT	77	XT1
53	PI5, TB5OUT	78	XT2
54	PB0, TDO, SWV	79	PI6, TB4IN0
55	PA0, TMS, SWDIO	80	$\overline{\text{NMI}}$
56	PA1, TCK, SWCLK	81	MODE
57	TEST3	82	$\overline{\text{RESET}}$
58	PJ7, INT7	83	PI7, TB4IN1
59	PB1, TDI	84	PH6, TB3IN0
60	PB2, $\overline{\text{TRST}}$	85	PH7, TB3IN1
61	PF3, RXIN1	86	PJ2, INT2
62	DVCC	87	PJ3, INT3
63	DVSS	88	PJ4, TB6OUT
64	PA2, TRACECLK	89	PE3, RXIN0
65	PA3, TRACEDATA0	90	TEST4
66	PA4, TRACEDATA1	91	PC0, AN0
67	PA5, TRACEDATA2	92	PC1, AN1
68	PA6, TRACEDATA3	93	PC2, AN2
69	PA7	94	PC3, AN3
70	PJ0, INTO	95	PDO, AN4, TB5IN0
71	CVCC	96	PD1, AN5, TB5IN1
72	X2	97	PD2, AN6, TB6IN0
73	CVSS	98	PD3, AN7, TB6IN1
74	X1	99	PD4, AN8
75	REGVSS	100	PD5, AN9

2.2 ピン名称と機能

TMPM330FYWFG の入出力ピン名称と機能を、ピン番号順、ポート順に示します。

2.2.1 ピン番号順

表 2-2 ピン名称と機能 <ピン番号順> (1/5)

分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
機能	1	PD6 AN10	入力 入力	入力ポート アナログ入力端子	P-up	—	—
	2	PD7 AN11	入力 入力	入力ポート アナログ入力端子	P-up	—	—
電源	3	AVSS	入力	A/D コンバータ用基準電源端子 ※A/D コンバータを使用しない場合でも GND に接続してください	—	—	—
	4	VREFH	入力	A/D コンバータ用基準電源端子 ※A/D コンバータを使用しない場合でも 電源に接続してください	—	—	—
	5	AVCC	入力	A/D コンバータ用電源端子 ※A/D コンバータを使用しない場合でも 電源に接続してください	—	—	—
機能	6	PG3 INT4	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	○
	7	PK2 TB9OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
	8	PJ5 TB7OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
	9	PH4 TB2IN0	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子	P-up	○	—
	10	PH5 TB2IN1	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子	P-up	○	—
	11	PG7 TB8OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	○
テスト	12	TEST2	—	テスト端子 ※必ず OPEN にしてください	—	—	—
電源	13	DVSS	—	GND 端子	—	—	—
	14	DVCC	—	電源端子	—	—	—
機能	15	PG4 SDA2/S02	入出力 入出力	入出力ポート I ² C モード送受信, S10 モード送信端子	P-up	○	○
	16	PG5 SCL2/S12	入出力 入出力	入出力ポート I ² C モードクロック, S10 モード受信端子	P-up	○	○
	17	PG6 SCK2	入出力 入出力	入出力ポート S10 モードクロック端子	P-up	○	○
テスト	18	TEST1	—	テスト端子 ※必ず OPEN にしてください	—	—	—
機能	19	PF7 INT5	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	○

表 2-2 ピン名称と機能 <ピン番号順> (2/5)

分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
機能	20	PE0 TXD0	入出力 出力	入出力ポート SIO 送信端子	P-up	—	○
	21	PE1 RXD0	入出力 入力	入出力ポート SIO 受信端子	P-up	○	○
	22	PE2 SCLK0 CTS0	入出力 入出力 入力	入出力ポート SIO クロック端子 SIO ハンドシェイク用端子	P-up	○	○
	23	PE4 TXD1	入出力 出力	入出力ポート SIO 送信端子	P-up	—	○
	24	PE5 RXD1	入出力 入力	入出力ポート SIO 受信端子	P-up	○	○
	25	PE6 SCLK1 CTS1	入出力 入出力 入力	入出力ポート SIO クロック端子 SIO ハンドシェイク用端子	P-up	○	○
	26	PG0 SDA0/S00	入出力 入出力	入出力ポート I ² C モード送受信, SIO モード送信端子	P-up	○	○
	27	PG1 SCL0/SIO	入出力 入出力	入出力ポート I ² C モードクロック, SIO モード受信端子	P-up	○	○
	28	PG2 SCK0	入出力 入出力	入出力ポート SIO モードクロック端子	P-up	○	○
	29	PB3	入出力	入出力ポート	P-up	—	—
機能/ 制御	30	PH0 TB0IN0 BOOT	入出力 入力 入力	入出力ポート タイマ B インพุットキャプチャ端子 BOOT モード端子 ※RESET 信号の立ち上がりで“L”をサンプリングしてシングルブートモードになります	P-up	○	—
機能	31	PH1 TB0IN1	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子	P-up	○	—
	32	PH2 TB1IN0	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子	P-up	○	—
	33	PF0 TXD2	入出力 出力	入出力ポート SIO 送信端子	P-up	—	○
	34	PF1 RXD2	入出力 入力	入出力ポート SIO 受信端子	P-up	○	○
	35	PF2 SCLK2 CTS2	入出力 入出力 入力	入出力ポート SIO クロック端子 SIO ハンドシェイク用端子	P-up	○	○
	36	PH3 TB1IN1	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子	P-up	○	—
	37	PB4	入出力	入出力ポート	P-up	—	—
	38	PI0 TB0OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
	39	PJ6 INT6	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
	40	PI1 TB1OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—

表 2-2 ピン名称と機能 <ピン番号順> (3/5)

分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
機能	41	PB5	入出力	入出力ポート	P-up	—	—
	42	PI2 TB2OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
	43	PB6	入出力	入出力ポート	P-up	—	—
	44	PF4 SDA1/S01	入出力 入出力	入出力ポート I ² C モード送受信, SIO モード送信端子	P-up	○	○
	45	PF5 SCL1/SI1	入出力 入出力	入出力ポート I ² C モードクロック, SIO モード受信端子	P-up	○	○
	46	PF6 SCK1	入出力 入出力	入出力ポート SIO モードクロック端子	P-up	○	○
	47	PB7	入出力	入出力ポート	P-up	—	—
	48	PI3 TB3OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
	49	PJ1 INT1	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
	50	PK0 CEC	入出力 入出力	入出力ポート CEC 端子	—	○	◎ (注 4)
	51	PK1 SCOUT ALARM	入出力 出力 出力	入出力ポート システムクロック出力 アラーム出力端子	P-up	—	—
	52	PI4 TB4OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
	53	PI5 TB5OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
機能/ デバッグ	54	PB0 TDO/SWV	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
	55	PA0 TMS/SWDIO	入出力 入出力	入出力ポート デバッグ用端子	P-up	○	—
	56	PA1 TCK/SWCLK	入出力 入力	入出力ポート デバッグ用端子	P-down	—	—
テスト	57	TEST3	—	テスト端子 ※必ず OPEN にしてください	—	—	—
機能	58	PJ7 INT7	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
機能/ デバッグ	59	PB1 TDI	入出力 入力	入出力ポート デバッグ用端子	P-up	—	—
	60	PB2 TRST	入出力 入力	入出力ポート デバッグ用端子	P-up	○	—
機能	61	PF3 RXIN1	入出力 入力	入出力ポート リモコン入力	P-up	○	○
電源	62	DVCC	—	電源端子	—	—	—
	63	DVSS	—	GND 端子	—	—	—

表 2-2 ピン名称と機能 <ピン番号順> (4/5)

分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
機能/ デバッグ	64	PA2 TRACECLK	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
	65	PA3 TRACEDATA0	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
	66	PA4 TRACEDATA1	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
	67	PA5 TRACEDATA2	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
	68	PA6 TRACEDATA3	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
機能	69	PA7	入出力	入出力ポート	P-up		
	70	PJ0 INT0	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
電源	71	CVCC	—	電源端子	—	—	—
クロック	72	X2	出力	高速発振子接続端子	—	—	—
電源	73	CVSS	—	GND 端子	—	—	—
クロック	74	X1	入力	高速発振子接続端子	—	○	—
電源	75	REGVSS	—	GND 端子	—	—	—
	76	REGVCC	—	電源端子	—	—	—
クロック	77	XT1	入力	低速発振子接続端子	—	○	—
	78	XT2	出力	低速発振子接続端子	—	—	—
機能	79	PI6 TB4IN0	入出力 入力	入出力ポート タイマ B インプットキャプチャ端子	P-up	○	—
	80	$\overline{\text{NMI}}$	入力	ノンマスクブル割り込み	—	○ ノイズフィルタ	—
制御	81	MODE	入力	モード端子 ※必ず GND に接続してください	—	○	—
機能	82	$\overline{\text{RESET}}$	入力	リセット入力端子	常時 P-up	○ ノイズフィルタ	—
	83	PI7 TB4IN1	入出力 入力	入出力ポート タイマ B インプットキャプチャ端子	P-up	○	—
	84	PH6 TB3IN0	入出力 入力	入出力ポート タイマ B インプットキャプチャ端子	P-up	○	—
	85	PH7 TB3IN1	入出力 入力	入出力ポート タイマ B インプットキャプチャ端子	P-up	○	—
	86	PJ2 INT2	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
	87	PJ3 INT3	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
	88	PJ4 TB6OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
	89	PE3 RXIN0	入出力 入力	入出力ポート リモコン入力	P-up	○	○
テスト	90	TEST4	—	テスト端子 ※必ず OPEN にしてください	—	—	—

表 2-2 ピン名称と機能 <ピン番号順> (5/5)

分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
機能	91	PC0 AN0	入力 入力	入力ポート アナログ入力端子	P-up	—	—
	92	PC1 AN1	入力 入力	入力ポート アナログ入力端子	P-up	—	—
	93	PC2 AN2	入力 入力	入力ポート アナログ入力端子	P-up	—	—
	94	PC3 AN3	入力 入力	入力ポート アナログ入力端子	P-up	—	—
	95	PD0 AN4 TB5IN0	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
	96	PD1 AN5 TB5IN1	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
	97	PD2 AN6 TB6IN0	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
	98	PD3 AN7 TB6IN1	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
	99	PD4 AN8	入力 入力	入力ポート アナログ入力端子	P-up	—	—
	100	PD5 AN9	入力 入力	入力ポート アナログ入力端子	P-up	—	—

- (注1) TEST1~4の端子処理は、必ずOPENにしてください。
- (注2) MODEの端子処理は、必ずGNDに接続してください。
- (注3) A/Dコンバータを使用しない場合でも、AVCC/VREFHは電源に、AVSSはGNDに接続してください。
- (注4) Nch オープンドレイン端子です。
- (注5) ノイズフィルタのノイズ除去幅は、Typ条件で約30nsです。

2.2.2 ポート順

表 2-3 ピン名称と機能 <ポート順> (1/5)

PORT	分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
PORT A	機能/ デバッグ	55	PA0 TMS/SWDIO	入出力 入出力	入出力ポート デバッグ用端子	P-up	○	—
		56	PA1 TCK/SWCLK	入出力 入力	入出力ポート デバッグ用端子	P-down	—	—
		64	PA2 TRACECLK	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
		65	PA3 TRACEDATA0	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
		66	PA4 TRACEDATA1	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
		67	PA5 TRACEDATA2	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
		68	PA6 TRACEDATA3	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
	機能	69	PA7	入出力	入出力ポート	P-up		
PORT B	機能/ デバッグ	54	PB0 TD0/SWV	入出力 出力	入出力ポート デバッグ用端子	P-up	—	—
		59	PB1 TDI	入出力 入力	入出力ポート デバッグ用端子	P-up	—	—
		60	PB2 $\overline{\text{TRST}}$	入出力 入力	入出力ポート デバッグ用端子	P-up	○	—
	機能	29	PB3	入出力	入出力ポート	P-up	—	—
		37	PB4	入出力	入出力ポート	P-up	—	—
		41	PB5	入出力	入出力ポート	P-up	—	—
		43	PB6	入出力	入出力ポート	P-up	—	—
47	PB7	入出力	入出力ポート	P-up	—	—		
PORT C	機能	91	PC0 AN0	入力 入力	入力ポート アナログ入力端子	P-up	—	—
		92	PC1 AN1	入力 入力	入力ポート アナログ入力端子	P-up	—	—
		93	PC2 AN2	入力 入力	入力ポート アナログ入力端子	P-up	—	—
		94	PC3 AN3	入力 入力	入力ポート アナログ入力端子	P-up	—	—

表 2-3 ピン名称と機能 <ポート順> (2/5)

PORT	分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
PORT D	機能	95	PD0 AN4 TB5IN0	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
		96	PD1 AN5 TB5IN1	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
		97	PD2 AN6 TB6IN0	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
		98	PD3 AN7 TB6IN1	入力 入力 入力	入力ポート アナログ入力端子 タイマBインプットキャプチャ端子	P-up	—	—
		99	PD4 AN8	入力 入力	入力ポート アナログ入力端子	P-up	—	—
		100	PD5 AN9	入力 入力	入力ポート アナログ入力端子	P-up	—	—
		1	PD6 AN10	入力 入力	入力ポート アナログ入力端子	P-up	—	—
		2	PD7 AN11	入力 入力	入力ポート アナログ入力端子	P-up	—	—
PORT E	機能	20	PE0 TXD0	入出力 出力	入出力ポート SIO送信端子	P-up	—	○
		21	PE1 RXD0	入出力 入力	入出力ポート SIO受信端子	P-up	○	○
		22	PE2 SCLK0 CTS0	入出力 入出力 入力	入出力ポート SIOクロック端子 SIOハンドシェイク用端子	P-up	○	○
		89	PE3 RXIN0	入出力 入力	入出力ポート リモコン入力	P-up	○	○
		23	PE4 TXD1	入出力 出力	入出力ポート SIO送信端子	P-up	—	○
		24	PE5 RXD1	入出力 入力	入出力ポート SIO受信端子	P-up	○	○
		25	PE6 SCLK1 CTS1	入出力 入出力 入力	入出力ポート SIOクロック端子 SIOハンドシェイク用端子	P-up	○	○
PORT F	機能	33	PF0 TXD2	入出力 出力	入出力ポート SIO送信端子	P-up	—	○
		34	PF1 RXD2	入出力 入力	入出力ポート SIO受信端子	P-up	○	○
		35	PF2 SCLK2 CTS2	入出力 入出力 入力	入出力ポート SIOクロック端子 SIOハンドシェイク用端子	P-up	○	○
		61	PF3 RXIN1	入出力 入力	入出力ポート リモコン入力	P-up	○	○

表 2-3 ピン名称と機能 <ポート順> (3/5)

PORT	分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
PORT F	機能	44	PF4 SDA1/S01	入出力 入出力	入出力ポート I ² C モード送受信, SIO モード送信端子	P-up	○	○
		45	PF5 SCL1/SI1	入出力 入出力	入出力ポート I ² C モードクロック, SIO モード受信端子	P-up	○	○
		46	PF6 SCK1	入出力 入出力	入出力ポート SIO モードクロック端子	P-up	○	○
		19	PF7 INT5	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	○
PORT G	機能	26	PG0 SDA0/S00	入出力 入出力	入出力ポート I ² C モード送受信, SIO モード送信端子	P-up	○	○
		27	PG1 SCL0/SIO	入出力 入出力	入出力ポート I ² C モードクロック, SIO モード受信端子	P-up	○	○
		28	PG2 SCK0	入出力 入出力	入出力ポート SIO モードクロック端子	P-up	○	○
		6	PG3 INT4	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	○
		15	PG4 SDA2/S02	入出力 入出力	入出力ポート I ² C モード送受信, SIO モード送信端子	P-up	○	○
		16	PG5 SCL2/SI2	入出力 入出力	入出力ポート I ² C モードクロック, SIO モード受信端子	P-up	○	○
		17	PG6 SCK2	入出力 入出力	入出力ポート SIO モードクロック端子	P-up	○	○
PORT H	機能/ 制御	30	PH0 TB0IN0 BOOT	入出力 入力 入力	入出力ポート タイマB インพุットキャプチャ端子 BOOT モード端子 ※RESET 信号の立ち上がりで“L”をサン プリングしてシングルブートモードに なります	P-up	○	—
	機能	31	PH1 TB0IN1	入出力 入力	入出力ポート タイマB インพุットキャプチャ端子	P-up	○	—
		32	PH2 TB1IN0	入出力 入力	入出力ポート タイマB インพุットキャプチャ端子	P-up	○	—
		36	PH3 TB1IN1	入出力 入力	入出力ポート タイマB インพุットキャプチャ端子	P-up	○	—
		9	PH4 TB2IN0	入出力 入力	入出力ポート タイマB インพุットキャプチャ端子	P-up	○	—
		10	PH5 TB2IN1	入出力 入力	入出力ポート タイマB インพุットキャプチャ端子	P-up	○	—
		84	PH6 TB3IN0	入出力 入力	入出力ポート タイマB インพุットキャプチャ端子	P-up	○	—
85	PH7 TB3IN1	入出力 入力	入出力ポート タイマB インพุットキャプチャ端子	P-up	○	—		

表 2-3 ピン名称と機能 <ポート順> (4/5)

PORT	分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
PORT I	機能	38	PI0 TB0OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
		40	PI1 TB1OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
		42	PI2 TB2OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
		48	PI3 TB3OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
		52	PI4 TB4OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
		53	PI5 TB5OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
		79	PI6 TB4IN0	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子	P-up	○	—
		83	PI7 TB4IN1	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子	P-up	○	—
PORT J	機能	70	PJ0 INT0	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
		49	PJ1 INT1	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
		86	PJ2 INT2	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
		87	PJ3 INT3	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
		88	PJ4 TB6OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—
		8	PJ5 TB7OUT	入出力 入力	入出力ポート タイマ B 出力端子	P-up	—	—
		39	PJ6 INT6	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
		58	PJ7 INT7	入出力 入力	入出力ポート 外部割込み端子	P-up	○ ノイズフィルタ	—
PORT K	機能	50	PK0 CEC	入出力 入出力	入出力ポート CEC 端子	—	○	◎ (注 4)
		51	PK1 SCOUT ALARM	入出力 出力 出力	入出力ポート システムクロック出力 アラーム出力端子	P-up	—	—
		7	PK2 TB9OUT	入出力 出力	入出力ポート タイマ B 出力端子	P-up	—	—

表 2-3 ピン名称と機能 <ポート順> (5/5)

分類	ピン番号	記号	入出力	機能	Pull-up Pull-down 制御選択可	Shmitt 入力 (ノイズフィルタ 有無)	Open Drain 出力選択可
機能	82	$\overline{\text{RESET}}$	入力	リセット入力端子	常時 P-up	○ ノイズフィルタ	—
	80	$\overline{\text{NMI}}$	入力	ノンマスカブル割り込み	—	○ ノイズフィルタ	—
制御	81	MODE	入力	モード端子 ※必ず GND に接続してください	—	○	—
クロック	72	X2	出力	高速発振子接続端子	—	—	—
	74	X1	入力	高速発振子接続端子	—	○	—
	77	XT1	入力	低速発振子接続端子	—	○	—
	78	XT2	出力	低速発振子接続端子	—	—	—
テスト	12	TEST2	—	テスト端子 ※必ず OPEN にしてください	—	—	—
	18	TEST1	—	テスト端子 ※必ず OPEN にしてください	—	—	—
	57	TEST3	—	テスト端子 ※必ず OPEN にしてください	—	—	—
	90	TEST4	—	テスト端子 ※必ず OPEN にしてください	—	—	—
電源	3	AVSS	入力	A/D コンバータ用基準電源端子 ※A/D コンバータを使用しない場合でも GND に接続してください	—	—	—
	4	VREFH	入力	A/D コンバータ用基準電源端子 ※A/D コンバータを使用しない場合でも 電源に接続してください	—	—	—
	5	AVCC	入力	A/D コンバータ用電源端子 ※A/D コンバータを使用しない場合でも 電源に接続してください	—	—	—
	13	DVSS	—	GND 端子	—	—	—
	14	DVCC	—	電源端子	—	—	—
	62	DVCC	—	電源端子	—	—	—
	63	DVSS	—	GND 端子	—	—	—
	71	CVCC	—	電源端子	—	—	—
	73	CVSS	—	GND 端子	—	—	—
75	REGVSS	—	GND 端子	—	—	—	
76	REGVCC	—	電源端子	—	—	—	

- (注 1) TEST1~4 の端子処理は、必ず OPEN にしてください。
- (注 2) MODE の端子処理は、必ず GND に接続してください。
- (注 3) A/D コンバータを使用しない場合でも、AVCC/VREFH は電源に、AVSS は GND に接続してください。
- (注 4) Nch オープンドレイン端子です。
- (注 5) ノイズフィルタのノイズ除去幅は、Typ条件で約 30ns です。

2.3 ピン名称と電源供給端子

表2-4 ピン名称と電源

ピン名称	電源
PA	DVCC
PB	DVCC
PC	AVCC
PD	AVCC
PE	DVCC
PF	DVCC
PG	DVCC
PH	DVCC
PI	DVCC
PJ	DVCC
PK	DVCC
X1, X2	CVCC
XT1, XT2	DVCC
$\overline{\text{RESET}}$	DVCC
$\overline{\text{NMI}}$	DVCC
MODE	DVCC

2.4 ピン番号と電源供給端子

表2-5 ピン番号と電源

電源	ピン番号	電圧範囲
DVCC	14, 62	2.7V~3.6V
AVCC	5	
REGVCC	76	
CVCC	71	

3 プロセッサコア

3.1 コアに関する情報

TX03 シリーズには、高性能 32 ビットプロセッサコア（ARM 社 Cortex-M3 コア）が内蔵されています。プロセッサコアの動作については、ARM 社からリリースされる“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。ここでは、製品固有の情報について説明します。

TMPM330FYWFG で使用している Cortex-M3 コアのリリースバージョンは以下のとおりです。コアのリリースバージョンによる動作の違いについては、ARM 社の発行するドキュメントを参照してください。

製品名	コアリリースバージョン
TMPM330FYWFG	r1p1-00rel0

Cortex-M3 コアは、一部のブロックについて実装するかどうかを選択することができます。r1p1 リリースバージョンで実装を選択可能なブロックは ETM と MPU です。TMPM330FYWFG では、ETM™ は実装していませんが、MPU は実装していません。

3.2 構成可能なオプション

Cortex-M3 コアは、一部のブロックについて実装するかどうかを選択することができます。r1p1 リリースバージョンで実装を選択可能なブロックは ETM™ と MPU です。TMPM330FYWFG での構成は以下のとおりです。

構成可能なオプション	実装
MPU	なし
ETM	あり

3.3 例外/割り込み

例外/割り込みに関連する製品固有の情報をまとめます。

3.3.1 割り込み本数

Cortex-M3 コアは割り込み本数を 1～240 本の間で任意に構成することができます。TMPM330FYWFG の割り込み本数は 50 本です。割り込み本数は NVIC レジスタの割り込みコントローラタイプレジスタの<INTLINESNUM[4:0]> ビットに反映され、本製品では“0y00001”が読み出されます。

3.3.2 割り込み優先度ビット数

Cortex-M3 コアは割り込み優先度ビット数を 3~8 ビットの間で任意に構成することができます。TMPM330FYWFG の割り込み優先度は 3 ビットです。このビット数は割り込み優先度レジスタとシステムハンドラ優先度レジスタのビット構成に反映されます。

3.3.3 SysTick

Cortex-M3 コアには SysTick と呼ばれるシステムタイマがあり、SysTick 例外を発生させることができます。

SysTick 例外の詳細については、例外の「SysTick」の章および、「NVIC レジスタ」の SysTick 関連レジスタの章を参照してください。

3.3.4 SYSRESETREQ

Cortex-M3 コアは、アプリケーション割り込みおよびリセット制御レジスタの<SYSRESETREQ>ビットがセットされると SYSRESETREQ 信号を出力します。

TMPM330FYWFG では SYSRESETREQ 信号が出力されるとウォームリセットと同様の動作になります。

注) SLOW モードでは<SYSRESETREQ>によるリセットは使用しないでください。

3.3.5 LOCKUP

回復不能な例外が発生すると Cortex-M3 コアは LOCKUP 信号を出力し、ソフトウェアに重大な誤りのあることを示します。

TMPM330FYWFG ではこの信号は未使用です。LOCKUP 状態から復帰する際にはマスク不能割り込み (NMI) またはリセットを使用する必要があります。

3.3.6 補助フォールトステータスレジスタ

Cortex-M3 コアにはソフトウェアに対して追加のシステムフォールト情報を提供するための補助フォールトステータスレジスタが準備されています。

TMPM330FYWFG ではこのレジスタに対して機能を定義していません。リードすると常に "0x0000_0000" が読み出されます。

3.4 イベント

Cortex-M3 コアにはイベント出力信号とイベント入力信号があります。イベント出力信号は、SEV 命令実行により出力されます。また、イベントが入力されると WFE 命令による低電力状態から復帰します。

TMPM330FYWFG では、イベント出力信号、イベント入力信号とも未使用です。SEV 命令、WFE 命令は使用しないでください。

3.5 電力管理

Cortex-M3 コアには電力管理のための信号として **SLEEPING** および **SLEEPDEEP** があります。**SLEEPDEEP** は、システム制御レジスタの<**SLEEPDEEP**>ビットがセットされている場合に出力されます。

これらの信号は、割り込み待ち(**WFI**)命令の実行、イベント待ち(**WFE**)命令の実行または、システム制御レジスタの<**SLEEPONEXIT**>ビットがセットされている場合の割り込みサービスルーチン(**ISR**)からの退出時に出力されます。

TMPM330FYWFG では、**SLEEPDEEP** 信号は使用していません。<**SLEEPDEEP**>ビットはセットしないでください。また、イベント信号も未使用のため、**WFE** 命令は使用しないでください。

電力管理については、「クロック/モード制御」の章を参照してください。

3.6 排他アクセス

Cortex-M3 コアの **DCode** バスおよびシステムバスは排他アクセスをサポートしていますが、**TMPM330FYWFG** ではこの機能を使用していません。

4. デバッグインターフェース

4.1 仕様概要

TMPM330 には ICE(In-Circuit Emulator)と接続するためのデバッグインターフェースとして SWJ-DP(Serial Wire JTAG Debug Port)ユニット、並びに 内部プログラムをトレース出力するための ETM(Embedded Trace Macrocell™)ユニットを搭載しております。トレース出力はマイコン内部の TPIU(Trace Port Interface Unit)を通じて専用端子 (TRACEDATA[0]~[3]、SWV) に出力されません。

SWJ-DP、ETM、TPIU の詳細に関しましては ARM 社からリリースされる “Cortex-M3 テクニカルリファレンスマニュアル” をご参照ください。

4.2 SWJ-DP 特長

2pin による シリアルワイヤデバッグポート (SWDCK, SWDIO) と、JTAG デバッグポート (TDI, TDO, TMS, TCK, TRST) をサポートしております。

4.3 ETM 特長

データ信号 4pin (TRACEDATA[0]~[3]) と クロック信号 1pin (TRACECLK) および、SWV によるトレース出力をサポートしております。

4.4 端子情報

デバッグインタフェース端子は汎用ポート機能と兼用です。

デバッグインタフェース端子のうち、PA0/PA1 端子は JTAG デバッグポート機能とシリアルワイヤデバッグポート機能で、PBO 端子は JTAG デバッグポート機能と SWV トレース出力機能で兼用しています。

PA0/ PA1/ PBO/ PB1/ PB2 ポートはリセット解除後デバッグインタフェース端子となっていますが、デバッグ時にユーザプログラムでリセット解除後すぐに汎用ポートに設定を変更すると、デバッグツールからの制御が受け付けられず、デバッグツールによるデバッグができなくなる場合があります。デバッグインタフェース端子の使用状況に合わせて、設定を変える場合は注意してください。

低消費電力モードを使用する場合には以下の注意事項に留意してください。

(注 1) PA0/ PBO が TMS/SWDIO 機能設定の場合、スタンバイコントロールレジスタ (GGSTBYCR) の <DRVE>ビットの状態によらず、STOP モード中も出力が有効な状態のまま保持されます。

(注 2) PA1 がデバッグ機能設定の場合、STOP/SLEEP モードで十分な低消費電力効果が得られません。デバッグ機能を使用しないときは、PA1 をポート設定にしてください。

以下にデバッグインタフェースの端子情報とリセット解除後のポートの設定をまとめます。

ピン番号	ポート (ビット名)	デバッグ機能	リセット解除後のポートの設定				
			機能 (PxFR)	入力 (PxIE)	出力 (PxCR)	プルアップ (PxPUP)	プルダウン (PxPDN)
55	PA0	TMS/SWDIO	○	○	○	○	-
56	PA1	TCK/SWCLK	○	○	×	-	○
54	PB0	TDO/SWV	○	×	○	×	-
59	PB1	TDI	○	○	×	○	-
60	PB2	TRST	○	○	×	○	-
64	PA2	TRCECLK	×	×	×	×	-
65	PA3	TRACEDATA0	×	×	×	×	-
66	PA4	TRACEDATA1	×	×	×	×	-
67	PA5	TRACEDATA2	×	×	×	×	-
68	PA6	TRACEDATA3	×	×	×	×	-

○：有効 ×：無効

4.5 デバッグツールとの接続

デバッグツールとの接続方法については、お使いになるツールメーカーが推奨する接続方法をご参照ください。

5 メモリマップ

TMPM330FYWFG のメモリマップは、ARM Cortex-M3 コアのメモリマップに沿って作られており、内蔵 ROM 領域は Cortex-M3 コアメモリマップの Code 領域、内蔵 RAM 領域は SRAM 領域、SFR 領域は Peripheral 領域に割り付けられています。SRAM 領域、内蔵 IO 領域はすべてビットバンド領域に含まれています。

CPU 内レジスタ領域はコア内部のレジスタ領域です。

各領域の詳細については、“Cortex-M3 テクニカルリファレンスマニュアル”を参照してください。

“Fault”と記載された領域は、アクセスするとメモリフォールトが有効な場合にはメモリフォールト、無効な場合にはハードフォールトが発生します。また、ベンダ固有領域はアクセスしないでください。

5.1 メモリマップ

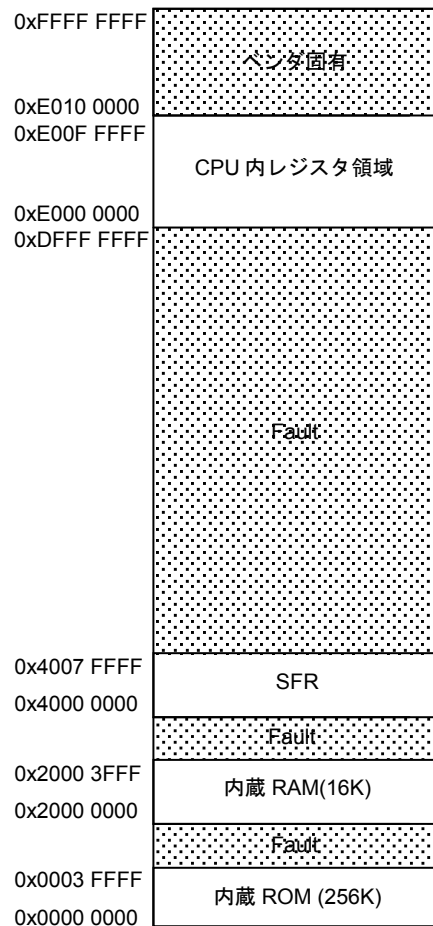


図 5.1 メモリマップ

(注) TMPM330FYWFG では 256KB の FLASH 領域とは別に、シングルブートモードの製品情報読み出しコマンドで使用する、データ領域およびパスワード領域がアドレス 0x0007_FF00 - 0x0007_FF80 に 128 ワード(1 ページ)分存在します。シングルブートモードについての詳細は、18 章のフラッシュ動作説明を参照してください。
0x0008_0000 からパスワード領域までの間の領域は、アクセスしても Fault は発生しません。書き込みは無視され、読み出される値は不定です。

6 リセット動作

リセットの種類として、外部リセット端子、WDT、SYSRESETREQ によるものがあります。WDT によるリセットについては WDT の章を参照してください。SYSRESETREQ によるリセットについては“Cortex-M3 テクニカルリファレンスマニュアルを参照してください。

注) SLOW モードでは、<SYSRESETREQ>によるリセットは使用しないでください。

6.1 コールドリセット時

電源投入の際には、内蔵レギュレータの安定のための時間と発振安定の時間を考慮する必要があります。本製品では、内蔵レギュレータ安定のための時間として $700\mu\text{s}$ が必要です。発振安定の時間はシステムにより異なります。コールドリセット時には、内蔵レギュレータと発振が安定するための十分な時間、外部リセット端子に“L”を入力する必要があります。

電源投入の手順を以下に示します。

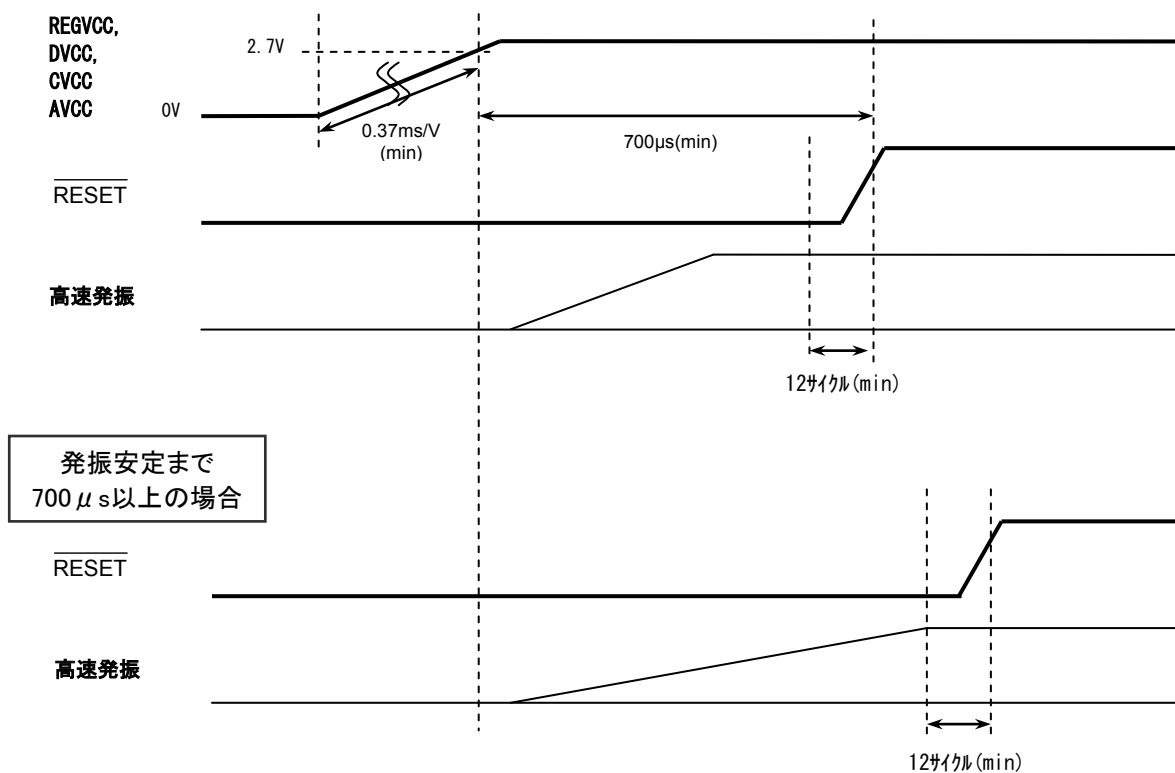


図6-1 コールドリセットシーケンス

- (注1) 電源立ち上げ(0V から 2.7V への立ち上げ)勾配は、 0.37ms/V 以上で使用してください。
- (注2) 電源投入はRESET 端子を”L”にした状態で行い、電源電圧が動作範囲で十分安定した状態でリセット解除させてください。

6.2 ウォームリセット時

6.2.1 リセット期間

本デバイスにリセットをかけるには、電源電圧が動作範囲内であり、かつ、内部高周波発振器の発振が安定した状態で、 $\overline{\text{RESET}}$ 入力を少なくとも 12 システムクロック間ローレベル”L”を入力してください。

6.3 リセット解除後

リセット解除後は、ほとんどの Cortex-M3 コアの制御レジスタや内蔵 I/O レジスタは初期化されます。コア内部のシステムデバッグコンポーネント（FPB、DWT、ITM）レジスタ、クロックジェネレータレジスタのリセットフラグレジスタおよび FLASH 関連レジスタのセキュリティビットレジスタはコールドリセットでのみ初期化されます。

リセット解除後は PLL 通倍回路が停止しているため、PLLSEL レジスタにて PLL 通倍回路使用の設定が必要です。

リセット例外処理を行った後、プログラムはリセットの割り込みサービスルーチンへ分岐します。

(注) リセット動作を行うと内蔵 RAM のデータは保証されません。

7. クロック/モード制御

7.1 特長

クロック/モード制御ブロックでは、クロックギアやプリスケラクロックの選択、PLL（逡倍回路）や発振器のウォーミングアップ等を設定することが可能です。

また、低消費電力モードがあり、モード遷移を行うことで電力の消費を抑えることが可能です。

本章では、クロックの制御および動作モードとモード遷移について説明します。

クロックに関連する機能としては以下のようなものがあります。

- システムクロックの制御
- プリスケラクロックの制御
- クロック逡倍回路（PLL）の制御
- ウォーミングアップタイマの制御

また、動作モードとして **NORMAL** モードと各種低消費電力モードがあり、使用方法に応じて消費電力を抑えることが出来ます。

7.2 レジスタ説明

7.2.1 レジスタ一覧

CG 関連のレジスタとアドレスを表 7-1 に示します。

表7-1 CG 関連レジスタ一覧

レジスタ名		アドレス
システムコントロールレジスタ	CGSYSCR	0x4004_0200
発振コントロールレジスタ	CGOSCCR	0x4004_0204
スタンバイコントロールレジスタ	CGSTBYCR	0x4004_0208
PLL セレクトレジスタ	CGPLLSEL	0x4004_020C
システムクロックセレクトレジスタ	CGCKSEL	0x4004_0210

7.2.2 レジスタ詳細

7.2.2.1 システムコントロールレジスタ

CGSYSCR		7	6	5	4	3	2	1	0	
	Bitsymbol	-	-	-	-	-	GEAR2	GEAR1	GEAR0	
	Read/Write	R					R/W	R/W	R/W	
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます					高速クロック (fc) のギア選択 000: fc 100: fc/2 001: reserved 101: fc/4 010: reserved 110: fc/8 011: reserved 111: reserved			
		15	14	13	12	11	10	9	8	
	Bitsymbol	-	-	-	FPSEL	-	PRCK2	PRCK1	PRCK0	
	Read/Write	R			R/W	R	R/W	R/W	R/W	
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます			fperiph 選択 0:fgear 1:fc	リード すると"0" が読め ます	プリスケラック選択 000: fperiph 100: fperiph/16 001: fperiph/2 101: fperiph/32 010: fperiph/4 110: Reserved 011: fperiph/8 111: Reserved			
		23	22	21	20	19	18	17	16	
	Bitsymbol	-	-	-	-	-	--	SCOSEL1	SCOSEL0	
	Read/Write	R						R/W	R/W	
	リセット後	0	0	0	0	0	0	0	1	
	機能	リードすると"0"が読めます						SCOUT 出力選択 00: fs 01: fsys/2 10: fsys 11: φT0		
		31	30	29	28	27	26	25	24	
Bitsymbol	-	-	-	-	-	-	-	-		
Read/Write	R									
リセット後	0	0	0	0	0	0	0	0		
機能	リードすると"0"が読めます									

<Bit 2:0><GEAR 2:0> : 高速クロック (fc) のギアを選択します。

<Bit 10:8><PRCK 2:0> : 周辺 I/O に供給するプリスケラックを選択します。

<Bit 12><FPSEL> : fperiph のソースクロックを選択します。

<Bit 17:16><SCOSEL1:0> : SCOUT 端子から設定したクロックを出力することが可能です。

7.2.2.2 発振コントロールレジスタ

	7	6	5	4	3	2	1	0	
CGOSCCR	Bitsymbol	-	WUPT2	WUPT1	WUPT0	WUPSEL	PLLON	WUEF	WUEON
	Read/Write	R	R/W	R/W	R/W	R/W	R/W	R	W
	リセット後	0	0	0	1	0	0	0	0
	機能	リードすると"0"が読めます	ウォーミングアップ時間選択 X1 選択時 000: WUP 無し 001: 2 ¹⁰ /入力周波数 010: 2 ¹¹ /入力周波数 011: 2 ¹² /入力周波数 100: 2 ¹³ /入力周波数 101: 2 ¹⁴ /入力周波数 110: 2 ¹⁵ /入力周波数 111: 2 ¹⁶ /入力周波数		XT1 選択時 000: WUP 無し 001: 2 ⁶ /入力周波数 010: 2 ⁷ /入力周波数 011: 2 ⁸ /入力周波数 100: 2 ⁹ /入力周波数 101: 2 ¹⁰ /入力周波数 110: 2 ¹¹ /入力周波数 111: 2 ¹² /入力周波数	ウォームアップカウンタ 0: X1 1: XT1	PLL 動作 0: 停止 1: 発振	ウォーミングアップタイマ (WUP) ステータス 0: WUP 終了 1: WUP 中	ウォーミングアップタイマ (WUP) 制御 0: don't care 1: WUP スタート
		15	14	13	12	11	10	9	8
	Bitsymbol	-	-	-	-	-	-	XTEN	XEN
	Read/Write	R		R/W		R		R/W	R/W
	リセット後	0	0	0	0	0	0	1	1
	機能	リードすると"0"が読めます		"0"を書いてください		リードすると"0"が読めます		低速発振器 0: 停止 1: 発振	高速発振器 0: 停止 1: 発振
		23	22	21	20	19	18	17	16
	Bitsymbol	-	-	-	-	-	-	-	-
	Read/Write	R							
	リセット後	0	0	0	0	0	0	0	0
	機能	リードすると"0"が読めます							
		31	30	29	28	27	26	25	24
	Bitsymbol	-	-	-	-	-	-	-	-
	Read/Write	R							
	リセット後	0	0	0	0	0	0	0	0
	機能	リードすると"0"が読めます							

- <Bit 0><WUEON> : ウォーミングアップタイマをスタートさせることが可能です。リードすると"0"が読めます。
- <Bit 1><WUEF> : ウォーミングアップタイマの状態を確認できます。
- <Bit 2><PLLON> : PLL (通倍回路) の動作を選択します。リセット解除後は "停止" 状態ですので設定が必要です。
- <Bit 3><WUPSEL> : ウォーミングアップさせたい発振器を選択します。選択された発振器のクロックでウォーミングアップタイマのカウントを行います。
- <Bit 6 : 4><WUPT2 : 0> : ウォーミングアップタイマの時間を選択します。
- <Bit 8><XEN> : 高速発振器の動作を選択します。
- <Bit 9><XTEN> : 低速発振器の動作を選択します。

7.2.2.3 スタンバイコントロールレジスタ

CGSTBYCR		7	6	5	4	3	2	1	0
	Bitsymbol	-	-	-	-	-	STBY2	STBY1	STBY0
	Read/Write	R					R/W	R/W	R/W
	リセット後	0	0	0	0	0	0	1	1
	機能	リードすると"0"が読めます					低消費電力モード選択 000 : Reserved 001 : STOP 010 : SLEEP 011 : IDLE 100 : Reserved 101 : Reserved 110 : Reserved 111 : Reserved		
		15	14	13	12	11	10	9	8
Bitsymbol	-	-	-	-	-	-	-	RXTEN	RXEN
Read/Write	R					R/W	R/W	R/W	
リセット後	0	0	0	0	0	0	0	0	1
機能	リードすると"0"が読めます					STOP モード解除後の低速発振器 0 : 停止 1 : 発振		STOP モード解除後の高速発振器 0 : 停止 1 : 発振	
		23	22	21	20	19	18	17	16
Bitsymbol	-	-	-	-	-	-	-	-	DRVE
Read/Write	R					R/W	R/W	R/W	
リセット後	0	0	0	0	0	0	0	0	0
機能	リードすると"0"が読めます					"0"を書いてください		STOP モード中の端子状態制御 0 : STOP モード中端子をドライブしません 1 : STOP モード中も端子をドライブします	
		31	30	29	28	27	26	25	24
Bitsymbol	-	-	-	-	-	-	-	-	-
Read/Write	R								
リセット後	0	0	0	0	0	0	0	0	0
機能	リードすると"0"が読めます								

<Bit 2 : 0><STBY2 : 0> : 低消費電力モードを選択します。

<Bit 8><RXEN> : STOP モード解除後の高速発振器動作を選択します。

<Bit 9><RXTEN> : STOP モード解除後の低速発振器動作を選択します。

<Bit 16><DRVE> : STOP モード中の端子ドライブ状態を選択します。

7.2.2.4 PLL セレクトレジスタ

CGPLLSEL		7	6	5	4	3	2	1	0	
	Bitsymbol	-	-	-	-	-	-	-	PLLSEL	
	Read/Write	R								R/W
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます								PLL 選択 0: X1 1: PLL 使用
		15	14	13	12	11	10	9	8	
	Bitsymbol	-	-	-	-	-	-	-	-	
	Read/Write	R								
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます								
		23	22	21	20	19	18	17	16	
	Bitsymbol	-	-	-	-	-	-	-	-	
	Read/Write	R								
リセット後	0	0	0	0	0	0	0	0		
機能	リードすると"0"が読めます									
	31	30	29	28	27	26	25	24		
Bitsymbol	-	-	-	-	-	-	-	-		
Read/Write	R									
リセット後	0	0	0	0	0	0	0	0		
機能	リードすると"0"が読めます									

<Bit 0><PLLSEL> : PLLにて逡倍されたクロックの使用可否を選択します。
リセット解除後は“X1”選択ですので、PLLを使用する場合は再設定が必要です。

7.2.2.5 システムクロックセレクトレジスタ

CGCKSEL		7	6	5	4	3	2	1	0
	Bitsymbol	-	-	-	-	-	-	SYSCK	SYSCKFLG
	Read/Write	R						R/W	R
	リセット後	0	0	0	0	0	0	0	0
	機能	リードすると"0"が読めます						システム クロック選 択 0: 高速 (fc) 1: 低速 (fs)	システム クロック ステータス 0 : 高速 (fc) 1 : 低速 (fs) <SYSCK> の値と同一 で安定して いる状態と なる
	15	14	13	12	11	10	9	8	
Bitsymbol	-	-	-	-	-	-	-	-	
Read/Write	R								
リセット後	0	0	0	0	0	0	0	0	
機能	リードすると"0"が読めます								
	23	22	21	20	19	18	17	16	
Bitsymbol	-	-	-	-	-	-	-	-	
Read/Write	R								
リセット後	0	0	0	0	0	0	0	0	
機能	リードすると"0"が読めます								
	31	30	29	28	27	26	25	24	
Bitsymbol	-	-	-	-	-	-	-	-	
Read/Write	R								
リセット後	0	0	0	0	0	0	0	0	
機能	リードすると"0"が読めます								

<Bit 0><SYSCKFLG> : システムクロック切り替え時の状態フラグを示します。
 <SYSCK>にて発振器の切り替えを行った場合、切り替え完了には時間差が発生します。
 <SYSCK>で選択した発振器が<SYSCKFLG>にて読み出されれば、発振器の切り替えが完了している状態です。

<Bit 1><SYSCK> : システムクロックの切り替えが可能です。
 <SYSCK>の切り替えを行う場合は、事前に OSCCR1<XEN>、<XTEN>を "1" に設定する必要があります。

7.3 クロック制御

7.3.1 クロックの種類

クロック系統図を図 7-1 に示します。各クロックの定義は以下のとおりです。

fosc	: X1, X2 端子より入力されるクロック
fs	: XT1, XT2 端子より入力されるクロック (低速クロック)
fpll	: PLL により通倍 (4 通倍) されたクロック
fc	: CGPLLSEL<PLLSEL>で選択されたクロック (高速クロック)
fgear	: CGSYSCR<GEAR[2:0]>で選択されたクロック
fsys	: CGCKSEL<SYSCK>で選択されたクロック (システムクロック)
fperiph	: CGSYSCR<FPSEL>で選択されたクロック
ΦT0	: CGSYSCR<PRCK[2:0]>で選択されたクロック (プリスケールクロック)

高速クロック fc と、プリスケールクロック ΦT0 は分周することが可能です。

高速クロック	: fc, fc/2, fc/4, fc/8
プリスケールクロック	: fperiph, fperiph/2, fperiph/4, fperiph/8, fperiph/16, fperiph/32

7.3.2 リセット動作による初期値

リセット動作により、クロックの設定は下記のような状態に初期化されます。

高速発振器	: ON (発振)
低速発振器	: ON (発振)
PLL (通倍回路)	: OFF (停止)
高速クロックギア	: fc (分周無し)

リセット動作により、低速クロック (fs) を除く全てのクロックの設定が fosc と同じになります。

fc	= fosc
fsys	= fosc
ΦT0	= fosc

例えば X1, X2 端子に 10MHz の発振子を接続している場合、システムクロック fsys は 10MHz となります。

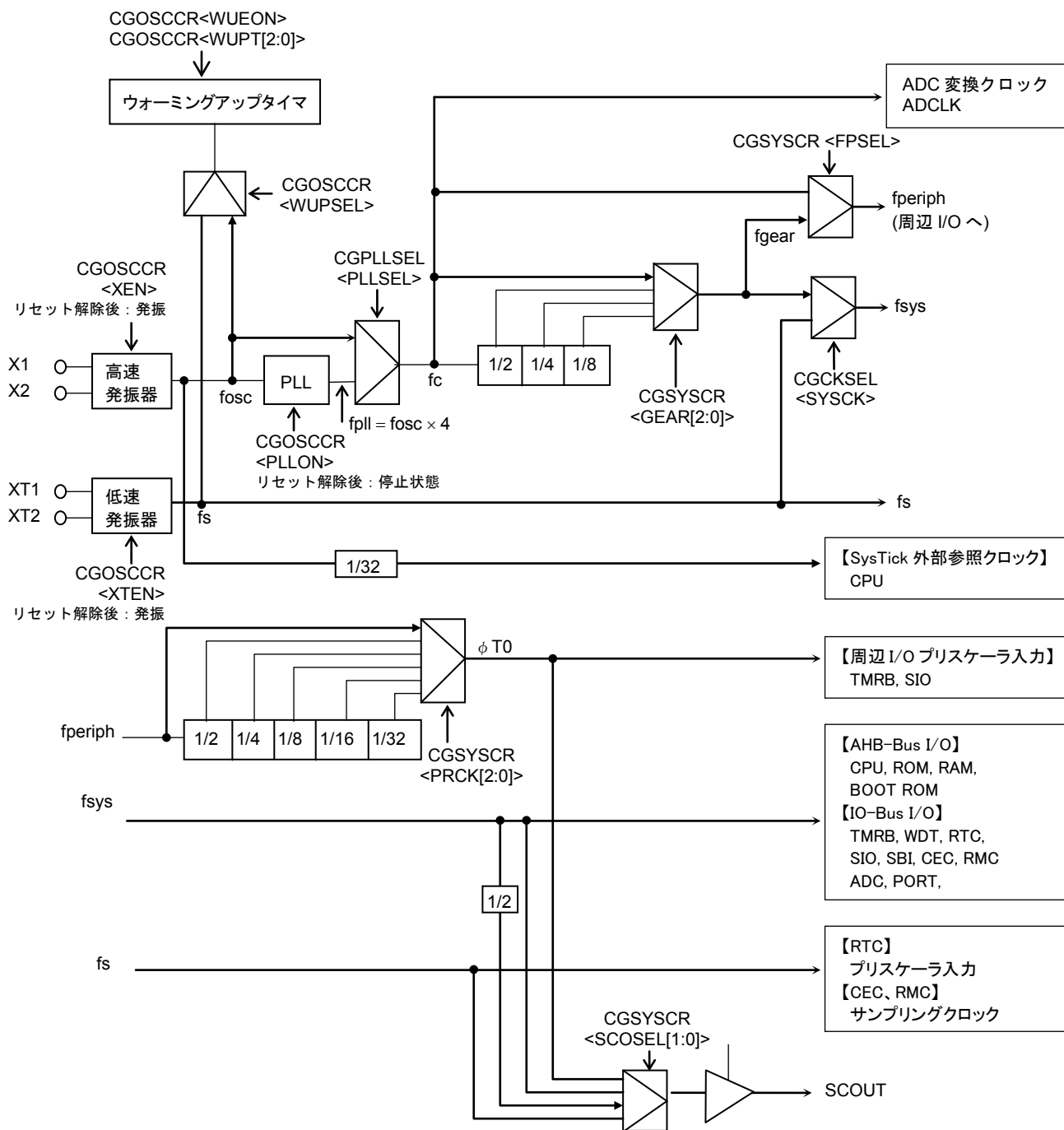


図7-1 クロック関連ブロック図

セレクタに入力されるクロックのうち、矢印つきのものがリセット後の初期状態として選択されます。

7.3.3 クロック逡倍回路 (PLL)

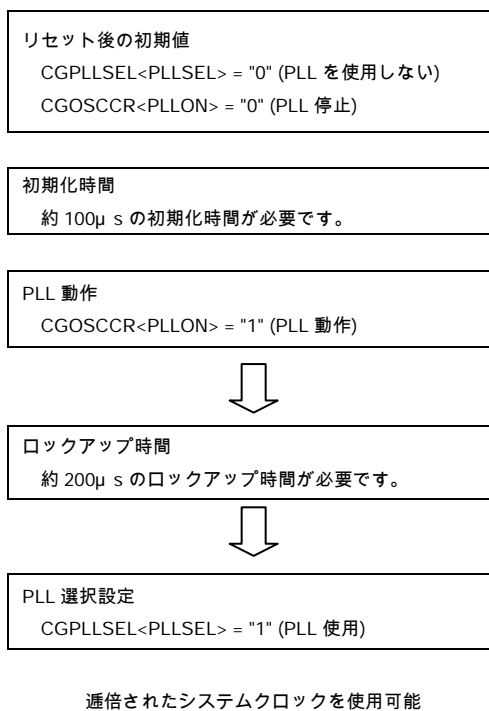
高速発振器の出力クロック f_{osc} を4逡倍した f_{pll} クロックを出力する回路です。これにより、発振器への入力周波数は低く内部クロックは高速にすることが可能です。

PLL はリセット解除後、ディセーブル状態です。PLL を有効にするためには、CGOSCCR<WUEF>ビットに“1”を設定します。また、PLL の動作が安定するまでの時間をウォーミングアップ機能等を用いて確保する必要があります。

(注) PLL の安定時間は、約 200 μ s です。

7.3.3.1 PLL 設定シーケンス

以下にリセット解除後の PLL 設定シーケンスを示します。



注) PLL を停止する場合は、CGPLLSEL<PLLSEL>="0" を設定後、CGPLLSEL<PLLSEL> をリードして"0" となっている事を確認した後に CGOSCCR<PLLON>=" 0" (PLL 停止) を設定してください。

7.3.4 ウォーミングアップ機能

ウォーミングアップ機能は、ウォーミングアップタイマを用いて発振子の発振安定時間や、PLL の安定時間を確保するための機能です。

ウォーミングアップ機能はまた、STOP/SLEEP モードからの復帰の際にも使用されます。この場合には低消費電力モードからの復帰割り込みが発生すると自動的にタイマがカウントを開始し、設定された時間のカウント終了後、システムクロックが出力され CPU が動作を開始します。

STOP/SLEEP モードでは PLL がディセーブルになるため、復帰の際には内部発振器の安定のほかに、PLL の安定時間も考慮してウォーミングアップ時間を設定してください。

ウォーミングアップ機能の設定方法を説明します。

まずウォーミングアップカウンタのカウントアップクロックを CGOSCCR<WUPSEL>ビットで選択します。

ウォーミングアップ時間は CGOSCCR<WUPT[2:0]>により選択できます。このウォーミングアップタイマのスタートおよび終了確認は、CGOSCCR<WUEON><WUEF>を使用してソフトウェア（命令）により行います。ウォーミングアップ終了を確認した後、CGCKSEL<SYSCK>にてシステムクロックの切り替えを行ってください。

クロック切り替えを行う場合、現在のシステムクロックは CGCKSEL<SYSCKFLG>をモニタすることで確認できます。

表 7-2にウォーミングアップ時間を示します。

表 7-2 ウォーミングアップ時間 (fosc=10 MHz, fs=32.768 kHz の場合)

ウォーミングアップ タイム選択 CGOSCCR<WUPT[2:0]>	高速クロック (fosc) CGOSCCR<WUPSEL>= "0"		低速クロック (fs) CGOSCCR<WUPSEL>= "1"	
	—	WUP 無し	—	WUP 無し
000	—	WUP 無し	—	WUP 無し
001	2 ¹⁰ /入力周波数	102.4 (μs)	2 ⁶ /入力周波数	1.953 (ms)
010	2 ¹¹ /入力周波数	204.8 (μs)	2 ⁷ /入力周波数	3.906 (ms)
011	2 ¹² /入力周波数	409.6 (μs)	2 ⁸ /入力周波数	7.813 (ms)
100	2 ¹³ /入力周波数	819.2 (μs)	2 ¹⁵ /入力周波数	1.0 (s)
101	2 ¹⁴ /入力周波数	1.638 (ms)	2 ¹⁶ /入力周波数	2.0 (s)
110	2 ¹⁵ /入力周波数	3.277 (ms)	2 ¹⁷ /入力周波数	4.0 (s)
111	2 ¹⁶ /入力周波数	6.554 (ms)	2 ¹⁸ /入力周波数	8.0 (s)

(注) ウォーミングアップタイムは発振クロックで動作しているため、発振周波数にゆらぎがある場合は誤差を含みます。従って概略時間としてとらえる必要があります。

以下に、ウォーミングアップ機能の設定例を示します。

<例 1> PLL の安定時間確保

CGOSCCR<WUPSEL>= "0" : ウォーミングアップカウンタ選択
 CGOSCCR<WUPT[2:0]>= "010" : ウォーミングアップ時間選択 (204.8 μs)
 CGOSCCR<WUEON>= "1" : ウォーミングアップタイム (WUP) スタート
 CGOSCCR<WUEF>リード : "0" (WUP 終了) になるまでウェイト

<例 2> NORMAL モードから SLOW モードへの移行

CGOSCCR<WUPSEL>= "1" : ウォーミングアップカウンタ選択
 CGOSCCR<WUPT[2:0]>= "xxx" : ウォーミングアップ時間選択
 CGOSCCR<XTEN>= "1" : 低速発振 (fs) イネーブル
 CGOSCCR<WUEON>= "1" : ウォーミングアップタイム (WUP) スタート
 CGOSCCR<WUEF>リード : "0" (WUP 終了) になるまでウェイト
 CGCKSEL<SYSCK>= "1" : システムクロックを低速 (fs) に切り替え
 CGCKSEL<SYSCKFLG>リード : "1" (現在のシステムクロックが fs) であることを確認
 CGOSCCR1<XEN>= "0" : 高速発振 (fosc) ディセーブル

<例 3> SLOW モードから NORMAL モードへの移行

CGOSCCR<WUPSEL>= "0" : ウォーミングアップカウンタ選択
 CGOSCCR<WUPT[2:0]>= "xxx" : ウォーミングアップ時間選択
 CGOSCCR<XEN>= "1" : 高速発振 (fosc) イネーブル
 CGOSCCR<WUEON>= "1" : ウォーミングアップタイム (WUP) スタート
 CGOSCCR<WUEF>リード : "0" (WUP 終了) になるまでウェイト
 CGCKSEL<SYSCK>= "0" : システムクロックを高速 (fgear) に切り替え
 CGCKSEL<SYSCKFLG>リード : "0" (現在のシステムクロックが fgear) であることを確認
 CGOSCCR<XTEN>= "0" : 低速発振 (fs) ディセーブル

(注) システムクロックを切り替える場合、CGSYSR<SYSCKFLG>をリードして、システムクロックが完全に切り変わっている事を確認してください。

7.3.5 システムクロック

システムクロックは高速クロックまたは低速クロックを選択できます。また、高速クロックは分周することが出来ます。

- X1, X2 入力周波数 : 8MHz~10MHz
- 発振子接続または外部クロック入力可能
- クロックギア : 1/1, 1/2, 1/4, 1/8 (リセット後は 1/1 分周)

表7-3 高速クロック範囲 (単位は MHz)

X1, X2 入力	最低動作 周波数	最大動作 周波数	リセット後 (PLL=OFF, CG=1/1)	クロックギア (CG) PLL=ON 時				クロックギア (CG) PLL=OFF 時			
				1/1	1/2	1/4	1/8	1/1	1/2	1/4	1/8
8MHz	1MHz	40MHz	8	32	16	8	4	8	4	2	1
10MHz			10	40	20	10	5	10	5	2.5	1.25

※ PLL=ON/OFF は CGOSCCR<PLLON>にて、クロックギア (CG) は CGSYSCR<GEAR2:0>にて設定できます。

- XT1, XT2 入力周波数

表7-4 低速クロック範囲

入力周波数範囲	最大動作周波数	最低動作周波数
30 ~ 34 (kHz)	34 kHz	30 kHz

(注1) クロックギアの切り替えは、CGSYSCR<GEAR[2:0]>レジスタへ値を書き込むことにより実行されます。書き込み後、実際に切り替わるまでに若干の時間を要します。

(注2) CEC 機能はサンプリングクロックとして低速クロックを使用しています。CEC 機能を使用する際に許容される誤差は、32.768kHz に対し約±4%です。

7.3.6 プリスケラクロック

周辺機能 (TMRB0~9, SIO0~2) には、それぞれにクロックを分周するプリスケラがあります。これらのプリスケラへ入力するクロック $\phi T0$ は、CGSYSCR<FPSEL>から選択されたクロック f_{periph} を更に CGSYSCR<PRCK[2:0]>にて分周することが可能です。リセット後の $\phi T0$ は $f_{periph}/1$ が選択されます。

(注) クロックギアを使用する場合、周辺 I/O の各ブロックのプリスケラ出力 ϕTn は、 $\phi Tn < f_{sys}$ を満足するように時間設定 (ϕTn が f_{sys} よりも遅くなるように) してください。また、タイマカウンタなどの周辺 I/O 動作中にクロックギアを切り替えないようにしてください。

7.3.7 システムクロックの端子出力機能

本製品には、システムクロックの端子出力機能があります。出力可能なクロックとして、低速クロック f_s 、システムクロックの 2 分周 $f_{sys}/2$ 、システムクロック f_{sys} 、プリスケーククロック $\phi T0$ を PK1/SCOUT 端子から出力できます。ポート K 関連レジスタ PKCR<PK1C>= “1”, PKFR1<PK1F1>= “1” に設定することにより、PK1/SCOUT 端子 (51pin) は SCOUT 出力端子になります。出力クロックの選択は CGSYSCR<SCOSEL[1:0]>によって設定します。

表 7-5に SCOUT 端子を SCOUT 出力に設定した場合のモード別端子状態を示します。

表7-5 モード別 SCOUT 出力状態

SCOUT 選択 CGSYSCR	モード	NORMAL	SLOW	低消費電力モード		
				IDLE	SLEEP	STOP
<SCOSEL[1:0]>= “00”		fs クロックを出力します				
<SCOSEL[1:0]>= “01”		fsys/2 クロックを出力します				
<SCOSEL[1:0]>= “10”		fsys クロックを出力します				
<SCOSEL[1:0]>= “11”		$\phi T0$ クロックを出力します	“0” または “1” に固定されます			

(注 1) SCOUT から出力されるシステムクロックは、内部クロックとの位相差 (AC タイミング) は保証できません。

(注 2) SCOUT に f_{sys} を選択しているときにクロックギアを切り替えると、切り替えた直後、 f_{sys} の波形が乱れます。波形の乱れがシステム上、問題となる場合は、クロックギア切り替え時に SCOUT 出力をディセーブルにするなど対策を施してください。

7.4 モードとモード遷移

7.4.1 モード状態遷移

動作モードとして、システムクロックに高速クロックを使用する場合と低速クロックを使用する場合があります、それぞれを **NORMAL** モード、**SLOW** モードと呼びます。

プロセッサコアの動作を停止して電力の消費を抑える低消費電力モードとして、**IDLE** モード、**SLEEP** モード、**STOP** モードがあります。

低速クロックを使用しない場合は **SLOW**/**SLEEP** モードは使用できません。

図 7-2 に状態遷移図を示します。

Sleep-on-exit については、“Cortex-M3 テクニカルリファレンスマニュアル” を参照してください。

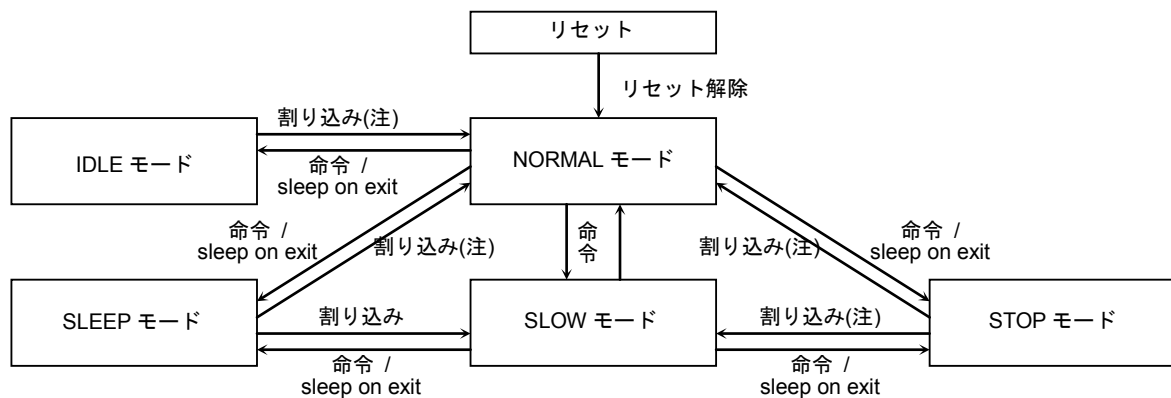


図7-2 モード状態遷移図

注) 復帰時にウォーミングアップが必要となります。ウォーミングアップ時間に関しては、低消費電力モードに入る前 (**NORMAL**) にて設定する必要があります。ウォーミングアップに関しては「6.3.4 ウォーミングアップ機能」を参照してください。

7.5 動作モード

動作モードにはNORMALモードとSLOWモードがあります。NORMAL, SLOWモードの特長は次の通りです。

7.5.1 NORMALモード

CPU コアおよび周辺ハードウェアを高速クロックで動作させるモードです。リセット解除後は、Normalモードになります。低速クロックを動作させることも可能です。

7.5.2 SLOWモード

高速クロック停止させ、CPU コア、周辺ハードウェアを低速クロックで動作させるモードです。NORMALモードに比べ消費電力を低減することができます。SLOWモードでは動作可能な周辺機能が限られます。使用できる周辺機能は、I/Oポート、リアルタイムクロック(RTC)、CEC機能、リモコン判定機能です。

(注1) CPU、リアルタイムクロック(RTC)、I/Oポート、CEC機能、リモコン判定機能以外の内蔵周辺機能はSLOWモードへ遷移する前に停止してください。

(注2) SLOWモードでは、Cortex-M3コアのNVICレジスタのアプリケーション割り込み、およびリセットレジスタ<SYSRESETREQ>によるリセットは使用しないでください。

7.6 低消費電力モード

低消費電力モードには、IDLE, SLEEP, STOPモードがあります。低消費電力モードに移行するには、システムコントロールレジスタCGSTBYCR<STBY[2:0]>にてモードを選択し、WFI(Wait For Interrupt)命令を実行します。WFI命令によって低消費電力モードへ移行した場合、低消費電力モードからの復帰はリセットまたは割り込み発生により行われます。割り込みで復帰する場合には、設定を行っておく必要があります。詳細は例外の章の割り込みを参照してください。

(注1) 本製品ではイベントによる復帰はサポートしていないため、WFE(Wait For Event)による低消費電力モードへの移行は行わないでください。

(注2) 本製品は、Cortex-M3コアのSLEEPDEEPによる低消費電力モードはサポートしていません。システム制御レジスタのSLEEPDEEPビットは設定しないでください。

IDLE, SLEEP, STOPモードの特長は次の通りです。

7.6.1 IDLE モード

CPUのみ停止するモードです。周辺機能は、各モジュールの中のレジスタに IDLE モード時の動作/停止設定レジスタを1ビット持ち、IDLE モードでの動作設定が可能です。IDLE モード時に動作停止に設定された周辺機能は、IDLE モードへ遷移した時の状態で停止します。

以下に IDLE モードでの動作を設定できる周辺機能の一覧を示します。設定方法は、各機能の章を参照してください。

- 16ビットタイマ/イベントカウンタ (TMRB)
- シリアルチャネル (SIO)
- シリアルバスインタフェース (SBI)
- アナログ/デジタルコンバータ (ADC)
- ウォッチドッグタイマ (WDT)

7.6.2 SLEEP モード

内部低速発振器と RTC、CEC 機能、リモコン判定機能が動作します。

SLEEP モードが解除されると、SLEEP モードへ移行する直前の動作モードへ復帰し、動作を開始します。

(注) PA1 (56pin) がデバッグ機能設定の場合、十分な低消費電力効果が得られません。デバッグ機能を使用しないときは、PA1 をポート設定にしてください。

7.6.3 STOP モード

STOP モードでは、内部発振器も含めてすべての内部回路が停止します。

STOP モードが解除されると、STOP モードへ移行する直前の動作モードへ復帰し、動作を開始します。

STOP モードでは、CGSTBYCR<DRVE>の設定により端子の状態を設定することができます。STOP モード時の端子状態を表 7-6 に示します。

(注) PA1 (56pin) がデバッグ機能設定の場合、十分な低消費電力効果が得られません。デバッグ機能を使用しないときは、PA1 をポート設定にしてください。

表7-6 STOP モード時の端子状態

	ピン名称	入/出力	<DRVE>=0	<DRVE>=1
ポート 以外	X1, XT1	入力専用	×	×
	X2, XT2	出力専用	“H” レベル出力	“H” レベル出力
	RESET, NMI, MODE	入力専用	○	○
ポート	PA0, PB0 (デバッグ機能設定 (PxFR<n>=1) かつ 出カイネーブル設定 (PxCR<n>=1) の場合)	入力	×	設定 (PxIE<n>) に従う
		出力	データ有効時はイネーブル データ無効時はディセーブル	データ有効時はイネーブル データ無効時はディセーブル
	PF7, PG3, PJ0-3, PJ6, PJ7 (割り込み機能設定 (PxFR<n>=1) かつ 入カイネーブル設定 (PxIE<n>=1) の場合)	入力	○	○
		出力	×	設定 (PxCR<n>) に従う
	上記以外のポート端子	入力	×	設定 (PxIE<n>) に従う
		出力	×	設定 (PxCR<n>) に従う

- ： 入力または出力がイネーブルであることを示します。
 ×： 入力または出力がディセーブルであることを示します。

7.6.4 低消費電力モードの選択

低消費電力モード選択は、スタンバイコントロールレジスタ (CGSTBYCR) の<STBY[2:0]>ビットの設定で選択されます。

表 7-7に<STBY[2:0]>の設定より選択されるモードを示します。

表7-7 低消費電力モードと設定

モード	CGSTBYCR <STBY[2:0]>
STOP	001
SLEEP	010
IDLE	011

(注) 上記の設定以外は行わないでください。

7.6.5 各モードにおける動作状態

各モードにおける動作状態を表 7-8に示します。

I/Oポートについては「○」は入出力の有効、「×」は無効を、その他の機能については「○」はクロックが供給されていることを、「×」は供給されていないことを示します。

表7-8 各動作モードにおける動作状態

ブロック	NORMAL	SLOW	IDLE	SLEEP	STOP
プロセッサコア	○	○	×	×	×
I/Oポート	○	○	○	○	* (注3)
ADC	○	× (注1)	モジュールご とに動作/停止 選択可能	×	×
SIO	○	× (注1)		×	×
SBI	○	× (注1)		×	×
TMRB	○	× (注1)		×	×
WDT	○	× (注1)		×	×
CEC	○	○	○	○	×
リモコン判定	○	○	○	○	×
RTC	○	○	○	○	×
CG	○	○	○	○	×
PLL	○	×	○	×	×
高速発振器 (fc)	○	* (注2)	○	×	×
低速発振器 (fs)	○	○	○	○	×

○ : 動作 × : 停止

(注1) ADC、SIO、SBI、TMRB、WDT は SLOW モードでは使用できないため停止してください。

(注2) 高速発振器は自動では停止しないため CGOSCCR<XEN>にて停止してください。

(注3) CGSTBYCR<DRVE>ビットの設定によります。

7.6.6 低消費電力モードの解除

低消費電力モードからの解除は、割り込み、NMI、リセットによって行うことができます。使用できるスタンバイ解除ソースは、低消費電力モードにより決まります。詳細を表 7-9 に示します。

表 7-9 解除ソースと解除可能なモード

低消費電力モード		IDLE	SLEEP	STOP
解除 ソース	INT0~7(注 1)	○	○	○
	INTRTC	○	○	×
	INTTBO~9	○	×	×
	INTCAPO0~60, 01~61	○	×	×
	INTRX0~2, INTTX0~2	○	×	×
	INTSBIO~2	○	×	×
	INTCECRX, INTCECTX	○	○	×
	INTRMCRX0, 1	○	○	×
	INTAD/INTADHP/INTADMO, 1	○	×	×
	SysTick 割り込み	○	×	×
NMI (INTWDT)	○	×	×	
NMI (NMI 端子)	○	○	○	
RESET (RESET 端子)	○	○	○	

○： 解除後、割り込み処理を開始します (RESET は LSI を初期化します)。

×： 解除に使用できません。

(注 1) レベルモードの割り込みによる解除を行う場合、割り込み処理が開始されるまでレベルを保持してください。それ以前でレベルを変化させた場合は、正しい割り込み処理を開始できません。

(注 2) 低消費電力モードへ移行する場合は、CPU で復帰要因以外の割り込みを禁止してください。禁止していないと復帰要因以外の割り込みで解除が行われる場合があります。

● 割り込み要求による解除

割り込みによって低消費電力モードを解除する場合、CPU で割り込みが検出されるよう準備しておく必要があります。SLEEP および STOP モードの解除に使用する割り込みは、CPU の設定のほかにクロックジェネレータで割り込み検出の設定を行う必要があります。

● NMI による解除

NMI の要因には、WDT 割り込み (INTWDT) と NMI 端子があります。INTWDT は IDLE モードでのみ使用可能です。NMI 端子による NMI ではすべての低消費電力モードを解除することができます。

● リセットによる解除

RESET 端子によるリセットですべての低消費電力モードからの解除を行うことができます。リセットで解除した場合には通常のリセット動作と同様になり、解除後は NORMAL モードで全てのレジスタが初期化された状態になります。

STOP モードの解除にリセットを使用する場合、自動ウォーミングアップが行われなため、発振器動作が安定するまでリセット信号を有効に保ってください。

● SysTick 割り込みによる解除

SysTick 割り込みは IDLE モードでのみ使用可能です。

割り込みの詳細に関しては「6. 割り込み」の項をご参照ください。

7.6.7 ウォーミングアップ

モード遷移の際には内部発振器の安定のためウォーミングアップが必要な場合があります。

STOP モードから NORMAL/SLOW モードへの遷移および、SLEEP モードから NORMAL モードへの遷移では、自動的にウォーミングアップ用カウンタが起動され、設定されているウォーミングアップ時間経過後にシステムクロックの出力が開始されます。このため、STOP/SLEEP モードに移行する命令を実行する前に CGOSCCR<WUPSEL>でウォーミングアップする発振器の選択および、<WUPT[2:0]>でウォーミングアップ時間の設定を行ってください。

(注) STOP/SLEEP モードでは PLL がディセーブルになるため、復帰の際には内部発振器の安定のほかに、PLL の安定時間も考慮してウォーミングアップ時間を設定してください。PLL の安定時間は、約 200 μ s です。

NORMAL モードから SLOW/SLEEP モードの遷移の場合、低速クロックがディセーブルの状態であれば低速クロックをイネーブルにした上で発振器の安定のためのウォーミングアップが必要になります。この場合、ソフトウェアでウォーミングアップを起動してください。

同様に、SLOW モードから NORMAL モードの遷移の場合で高速クロックがディセーブルの場合、高速クロックをイネーブルにし、ウォーミングアップを起動してください。

各動作モード遷移時におけるウォーミングアップ有無を表 7-10 に示します。

表7-10 各動作モード遷移時のウォーミングアップ

動作モード遷移	ウォーミングアップ
NORMAL→IDLE	不要
NORMAL→SLEEP	(注1)
NORMAL→SLOW	(注1)
NORMAL→STOP	不要
IDLE→NORMAL	不要
SLEEP→NORMAL	自動ウォーミングアップ
SLEEP→SLOW	不要
SLOW→NORMAL	(注2)
SLOW→SLEEP	不要
SLOW→STOP	不要
STOP→NORMAL	自動ウォーミングアップ(注3)
STOP→SLOW	自動ウォーミングアップ

(注1) 低速クロックがディセーブルの場合、イネーブルにした上でソフトウェアでウォーミングアップを起動する必要があります。

(注2) 高速クロックがディセーブルの場合、イネーブルにした上でソフトウェアでウォーミングアップを起動する必要があります。

(注3) リセットで解除する場合には自動ウォーミングアップを行わないため、発振動作が安定するまでリセット信号を有効にしてください。

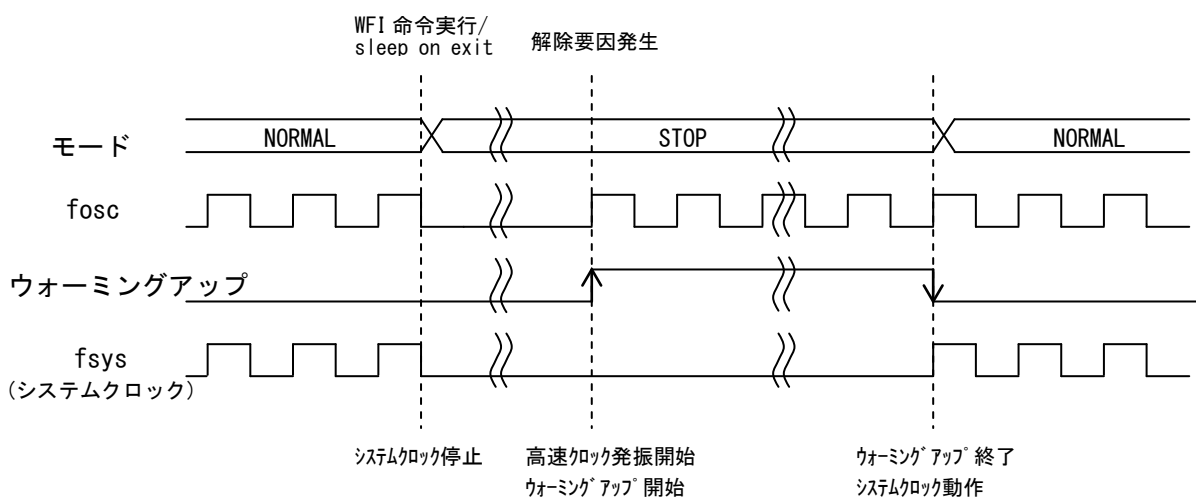
7.6.8 モード遷移によるクロック動作

モード遷移の際の、クロック動作について 5.6.8.1～5.6.8.4 に示します。

7.6.8.1 NORMAL→STOP→NORMAL 動作モード遷移

STOP モードから NORMAL モードへ復帰する場合、ウォーミングアップは自動的に起動します。STOP モードへ遷移する前にウォーミングアップ時間の設定を行ってください。

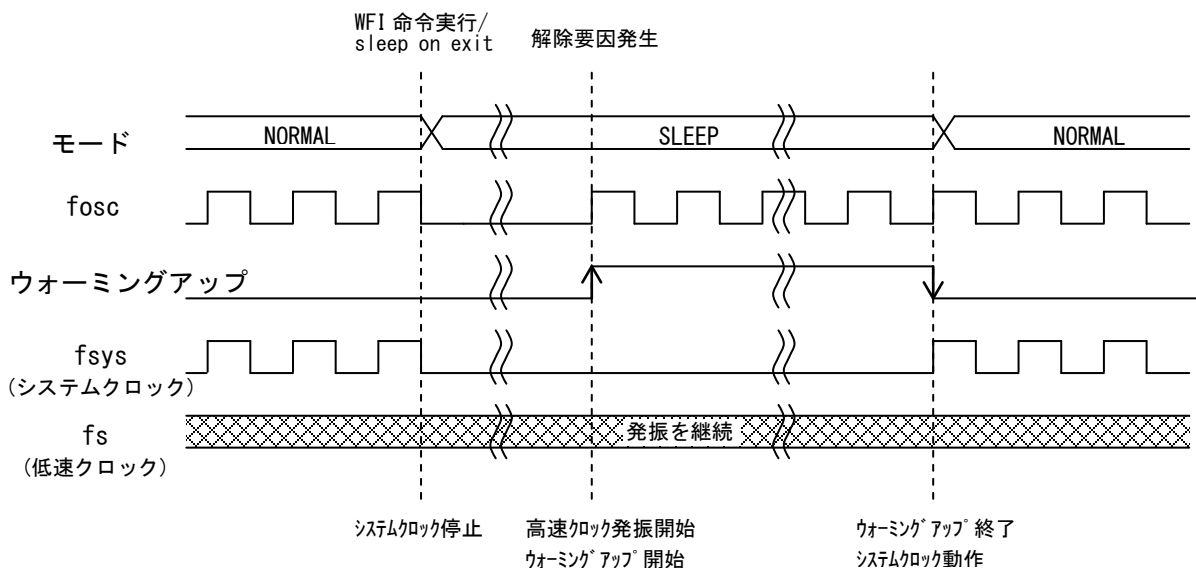
リセットで NORMAL モードへ復帰する場合にはウォーミングアップは行われませんので、発振動作が安定するまでリセット信号を有効に保ってください。



7.6.8.2 NORMAL→SLEEP→NORMAL 動作モード遷移

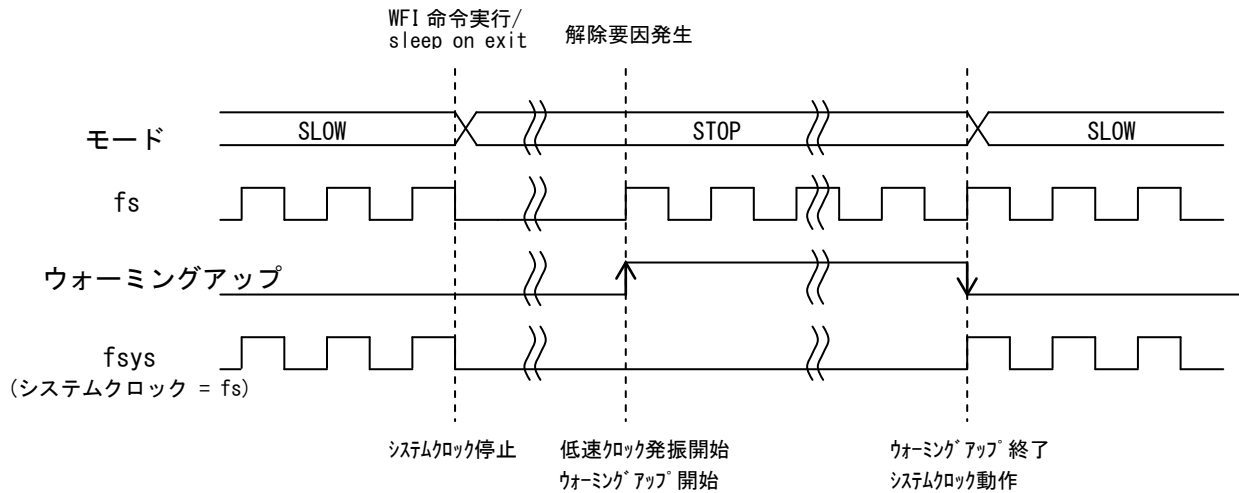
SLEEP モードから NORMAL モードへ復帰する場合、ウォーミングアップは自動的に起動します。SLEEP モードへ遷移する前にウォーミングアップ時間の設定を行ってください。

リセットで NORMAL モードへ復帰する場合にはウォーミングアップは行われませんので、発振動作が安定するまでリセット信号を有効に保ってください。



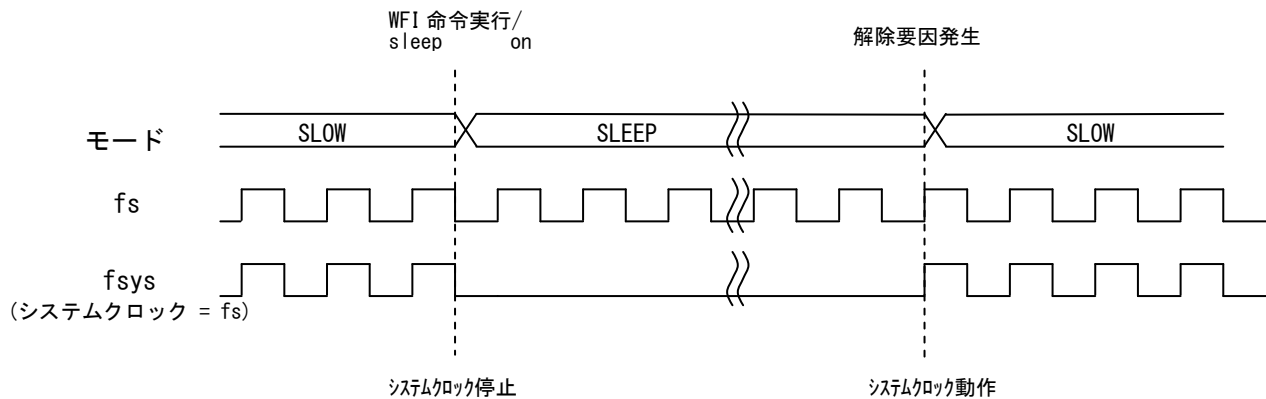
7.6.8.3 SLOW→STOP→SLOW 動作モード遷移

ウォーミングアップは自動的に起動します。STOP モードへ遷移する前にウォーミングアップ時間の設定を行ってください。



7.6.8.4 SLOW→SLEEP→SLOW 動作モード遷移

低速クロックは SLEEP モード中も継続して発振しているため、ウォーミングアップの必要はありません。



8 例外

この章では、例外の特徴、種類、処理について概略を説明します。

例外は CPU のアーキテクチャと深くかかわる部分ですので、必要に応じて Cortex-M3 テクニカルリファレンスマニュアルもご覧ください。

8.1 概要

例外は CPU に対し現在実行中の処理を中断して別の処理に移ることを要求するものです。

例外には、何らかの異常な状態が起こったときや例外を発生する命令を実行したときに発生するものと、外部端子や周辺機能からの割り込み要求信号といった、ハードウェアによる要因で発生する割り込みがあります。

全ての例外は優先度にしたがって CPU 内にあるネスト型ベクタ割り込みコントローラ (NVIC) によって処理されます。例外が発生すると、CPU はそのときの状態をスタックに退避し、割り込み処理ルーチンへ分岐します。割り込み処理ルーチンの実行後、スタックに退避した情報は自動的に復帰されます。



8.1.1 種類

例外には以下のようなものがあります。

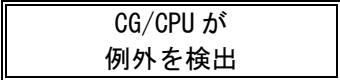



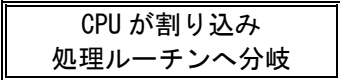

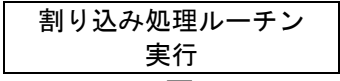

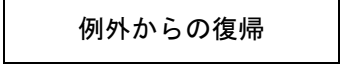
それぞれの例外の詳細な内容は、“Cortex-M3 テクニカルリファレンスマニュアル” をご覧ください。

- リセット
- マスク不能割り込み (NMI)
- ハードフォールト
- メモリ管理
- バスフォールト
- 用法フォールト
- SVCall (スーパーバイザコール)
- デバッグモニタ
- PendSV
- SysTick
- 外部割り込み

8.1.2 処理の流れ

例外／割り込みの処理の流れの概略を以下に示します。以下の説明で、 はハードウェアによる処理を、 はソフトウェアによる処理を示しています。

それぞれの処理の内容について、後続の章で説明します。

処理	内容	説明
 CG/CPU が 例外を検出	CG/CPU が例外要求を検出します。	8.1.2.1章
  CPU が 例外を処理	CPU が例外処理を行います。	8.1.2.2章
  CPU が割り込み 処理ルーチンへ分岐	検出した例外に応じた割り込み処理ルーチンへ分岐します。	
  割り込み処理ルーチン 実行	必要な処理を行います。	8.1.2.3章
  例外からの復帰	別の割り込み処理ルーチンまたはもとのプログラムに復帰します。	8.1.2.4章

8.1.2.1 例外要求と検出

(1) 例外要求の発生

例外は、CPU の命令実行、メモリアクセス、外部割り込み端子や周辺機能からの割り込み要求などにより発生します。

CPU の命令実行による例外の要求は、例外を発生する命令の実行や、命令実行中の異常が要因で発生します。

メモリアクセスによる例外の要求は、実行不可領域からの命令フェッチや、フォールト領域へのアクセスにより発生します。

外部割り込みは、外部割り込み端子からの信号入力や周辺機能から発生します。スタンバイ解除に使用する割り込みの場合、クロックジェネレータの設定も必要になります。詳細は「8.5 割り込み」の章で説明します。

(2) 例外の検出

複数の例外が同時に検出された場合には、CPU は優先度にしたがって最も優先度の高い例外を選択します。

各例外の優先度は以下のとおりです。“構成可能”と記載された例外は、優先度を設定することができます。また、メモリ管理、バスフォールト、用法フォールトは許可/禁止を選択することができます。禁止された例外が発生した場合にはハードフォールトとして扱われます。

表8-1 例外の種類と優先度

番号	例外	優先度	要因
1	リセット	-3 (最高)	リセット端子、WDT、SYSRESETREQ
2	マスク不能割り込み	-2	NMI 端子または WDT
3	ハードフォールト	-1	より優先度の高いフォールトの処理中、または禁止されているために発生できないフォールト
4	メモリ管理	構成可能	MPU(メモリ保護ユニット)からの例外 (注1) 実行不可 (XN) (Execute Never) 領域からの命令フェッチ
5	バスフォールト	構成可能	メモリマップの Hard Fault 領域に対するアクセス
6	用法フォールト	構成可能	未定義命令の実行や、命令実行によって発生するその他のエラー
7-10	予約		
11	SVCcall	構成可能	SVC 命令によるシステムサービスの呼び出し
12	デバッグモニタ	構成可能	コアがフォールト中でないときのデバッグモニタ
13	予約		
14	PendSV	構成可能	保留可能なシステムサービスへの要求
15	SysTick	構成可能	システムタイマからの通知
16-	外部割り込み	構成可能	外部割り込み端子や周辺機能 (注2)

(注1) 本製品はMPUを搭載していません。

(注2) 外部割込みは、製品により要因と番号が異なります。具体的な要因と番号については、8.5.1.4 要因一覧を参照してください。

(3) 優先度の設定

・優先度レベル

外部割込みの優先度は、割り込み優先度レジスタで、それ以外の例外はシステムハンドラ優先度レジスタで設定します。

優先度レジスタは、構成を変更することが可能になっており、優先度設定のためのビット数は製品により3ビット～8ビットのいずれかになります。このため、設定できる優先度の範囲も製品により異なります。

8ビットの構成の場合、優先度は0～255のレベルを設定できます。最も高い優先度は0です。

複数の要因を同じ優先度に設定した場合、番号の小さい例外の優先度が高くなります。

(注) 本製品では、割り込み優先度レジスタとシステムハンドラ優先度レジスタの優先度設定領域は3ビットの構成になっています。

・優先度のグループ化

優先度をグループ化することもできます。アプリケーション割り込みおよびリセット制御レジスタの<PRIGROUP>を設定することで、<PRI_n>を横取り優先度とサブ優先度に分割することができます。優先度はまず横取り優先度で判定され、横取り優先度が同じ場合サブ優先度で判定されます。サブ優先度も同じ場合は例外番号の小さいほうが優先度が高くなります。

以下に優先度のグループ化の設定についてまとめます。表中の横取り優先度数、サブ優先度数は、<PRI_n>が8ビット構成の場合の数です。

優先度のグループ化設定

<PRIGROUP[2:0]> の設定	<PRI_n[7:0]>		横取り優先度数	サブ優先度
	横取り フィールド	サブ優先度数 フィールド		
000	[7:1]	[0]	128	2
001	[7:2]	[1:0]	64	4
010	[7:3]	[2:0]	32	8
011	[7:4]	[3:0]	16	16
100	[7:5]	[4:0]	8	32
101	[7:6]	[5:0]	4	64
110	[7]	[6:0]	2	128
111	なし	[7:0]	1	256

注) <PRI_n>の構成が8ビットより小さい場合、下位ビットは"0"となります。

たとえば、3ビット構成の場合、<PRI_n[7:5]>で優先度が設定され、<PRI_n[4:0]>は"00000"になります。

8.1.2.2 例外の処理と割り込み処理ルーチンへの分岐（横取り）

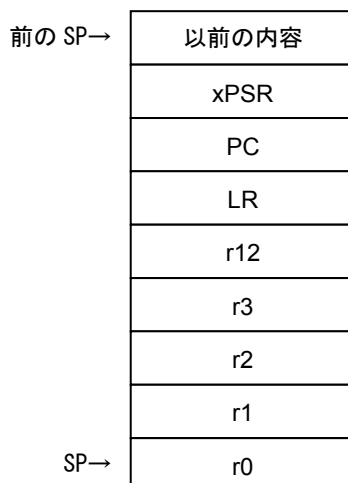
例外により、実行中の処理を中断して割り込み処理ルーチンへ分岐する動作を“横取り”と呼びます。

（1）レジスタの退避

例外を検出すると、CPU は 8 つのレジスタの内容を退避します。退避するレジスタと退避の順序は以下のとおりです。

- プログラムカウンタ（PC）
- プログラムステータスレジスタ（xPSR）
- r0 ~ r3
- r12
- リンクレジスタ（LR）

レジスタの退避が終了すると、SP は 8 ワード分減らされます。レジスタ退避終了後のスタックの状態は以下のようになっています。



（2）割り込み処理ルーチンのフェッチ

レジスタの退避と同時に CPU は割り込み処理ルーチンの命令フェッチを行います。

各例外の割り込み処理ルーチンの先頭番地をベクタテーブルに準備しておきます。ベクタテーブルはリセット後、コード領域の 0x0000_0000 番地に置かれます。ベクタテーブルは、ベクタテーブルオフセットレジスタを設定することでコード空間または SRAM 空間の任意のアドレスに置くことができます。

ベクタテーブルにはまた、メインスタックの初期値を設定します。

(3) 後着

割り込み処理ルーチンの実行前に、検出した例外よりも優先度の高い例外を検出した場合、CPU は優先度の高い例外の処理に移行します。これを後着と呼びます。

後着の場合、CPU は新たに検出された例外の割り込み処理ルーチンのフェッチを改めて行い、分岐しますが、再度レジスタの内容を退避することはありません。

(4) ベクタテーブルの構成

ベクタテーブルの構成は以下のとおりです。

最初の4ワード（スタックの先頭アドレス、リセット、NMI、ハードフォールトの割り込み処理ルーチンアドレス）は必ず設定する必要があります。その他の例外の割り込み処理ルーチンアドレスは、必要に応じて準備します。

オフセット	例外	内容	備考
0x00	リセット	メインスタックの初期値	必須
0x04	リセット	割り込み処理ルーチンアドレス	必須
0x08	マスク不能割り込み	割り込み処理ルーチンアドレス	必須
0x0C	ハードフォールト	割り込み処理ルーチンアドレス	必須
0x10	メモリ管理	割り込み処理ルーチンアドレス	任意
0x14	バスフォールト	割り込み処理ルーチンアドレス	任意
0x18	用法フォールト	割り込み処理ルーチンアドレス	任意
0x1C ~ 0x28	予約		
0x2C	SVCall	割り込み処理ルーチンアドレス	任意
0x30	デバッグモニタ	割り込み処理ルーチンアドレス	任意
0x34	予約		
0x38	PendSV	割り込み処理ルーチンアドレス	任意
0x3C	SysTick	割り込み処理ルーチンアドレス	任意
0x40	外部割り込み	割り込み処理ルーチンアドレス	任意

8.1.2.3 割り込み処理ルーチンの実行

割り込み処理ルーチンでは、発生した例外に応じて必要な処理を行います。割り込み処理ルーチンはユーザが準備します。

割り込み処理ルーチンでは、通常の処理プログラムに戻ったときに再度同じ割り込みが発生しないよう、割り込み要求の取り下げなどの処理が必要になる場合があります。

割り込みについての詳細は「8.5 割り込み」の項で説明します。

割り込み処理ルーチンの実行中に現在処理中の例外よりも優先度の高い例外を検出した場合、CPUは現在実行中の割り込み処理ルーチンを中断し新たに検出された例外の処理を行います。

8.1.2.4 例外からの復帰

(1) 割り込み処理ルーチンからの復帰先

割り込み処理ルーチン終了時の状態により復帰先が決まります。

- テールチェイン

保留中の例外が存在し、中断されている例外処理がないかまたは中断されているどの例外よりも優先度が高い場合、保留中の例外の割り込み処理ルーチンへ復帰します。

このとき、スタックの退避と復帰は省略されます。この動作をテールチェインと呼びます。

- 処理が中断されている割り込み処理ルーチンへ復帰

保留中の例外がない場合、または存在しても処理が中断されている例外の優先度が高い場合、中断されている例外の割り込み処理ルーチンへ復帰します。

- 元のプログラムへ復帰

保留中の例外も処理が中断されている例外もない場合、元のプログラムへ復帰します。

(2) 復帰処理

CPUは、割り込み処理ルーチンから復帰する際に以下の処理を行います。

- レジスタの復帰

退避していた8つのレジスタ(PC、xPSR、r0~r3、r12、LR)を復帰しSPを調整します。

- 割り込み番号のロード

退避していたxPSRから現在有効な割り込み番号をロードします。この割り込み番号によってCPUはどの割り込みに復帰するかを制御します。

- SPの選択

例外（ハンドラモード）へ復帰する場合、SPはSP_mainです。スレッドモードへ復帰する場合、SPはSP_mainまたはSP_processです。

8.2 リセット例外

リセット例外には、以下の3種類の要因があります。

リセットの要因を確認するためには、クロックジェネレータレジスタのリセットフラグレジスタ (RSTFLG) を参照してください。

・外部リセット端子

外部リセット端子を“L”にしたのち、“H”にすることによりリセット例外が発生します。

・WDTによるリセット例外

WDTにリセットを発生する機能があります。詳細はWDTの章をご覧ください。

・SYSRESETREQによるリセット例外

NVICレジスタの、アプリケーション割り込みおよびリセット制御レジスタのSYSRESETREQビットをセットすることで、リセットを発生させることができます。

注) SLOWモードでは<SYSRESETREQ>によるリセットは使用しないでください。

8.3 マスク不能割り込み (NMI)

マスク不能割り込みには、以下の2種類の要因があります。

マスク不能割り込みの要因を確認するためには、クロックジェネレータレジスタのNMIフラグレジスタ (NMIFLG) を参照してください。

・外部NMI端子

外部NMI端子を“H”から“L”にすることによりマスク不能割り込みが発生します。

・WDTによるマスク不能割り込み

WDTにマスク不能割り込みを発生する機能があります。詳細はWDTの章をご覧ください。

8.4 SysTick

SysTick は、CPU の持つシステムタイマを使用した割り込み機能です。

SysTick リロード値レジスタに値を設定し、SysTick 制御およびステータスレジスタで機能をイネーブルにすると、リロード値レジスタに設定された値がカウンタへリロードされカウントダウンを開始します。カウンタが“0”になると SysTick 例外が発生します。また、例外を保留しフラグでタイマが“0”になったことを確認することもできます。

SysTick 較正值レジスタには、システムタイマで 10ms を計測する際のリロード値が準備されています。製品により、カウントクロックの周期は異なるため、較正值レジスタに設定されている値も異なります。

(注) 本製品では、外部参照クロックとして fosc を 32 分周したクロックが使用されます。

8.5 割り込み

この章では、割り込みの伝わる経路、要因、必要な設定について説明します。

割り込みは、割り込み要因ごとの信号により CPU へ通知されます。

CPU は、優先順位付けを行い最も優先度の高い割り込みを発生します。

スタンバイ解除に使用する割り込み要因は、クロックジェネレータを経由して CPU に要因が伝わるため、クロックジェネレータの設定も必要です。

8.5.1 要因

8.5.1.1 経路

割り込み要求の経路を図 8-1 に示します。

周辺機能からの割り込み要求のうち、スタンバイ解除に使用されないものは直接 CPU に入力されます。(経路①)

スタンバイ解除に使用される周辺機能割り込み(経路②)および、外部割り込み端子からの割り込み(経路③)はクロックジェネレータに入力され、スタンバイ解除のロジックを経由して CPU に入力されます。(経路④⑤)

外部割り込み端子からの割り込みは、スタンバイ解除に使用しない場合スタンバイ解除ロジックを経由せずに CPU に入力されます。(経路⑥)

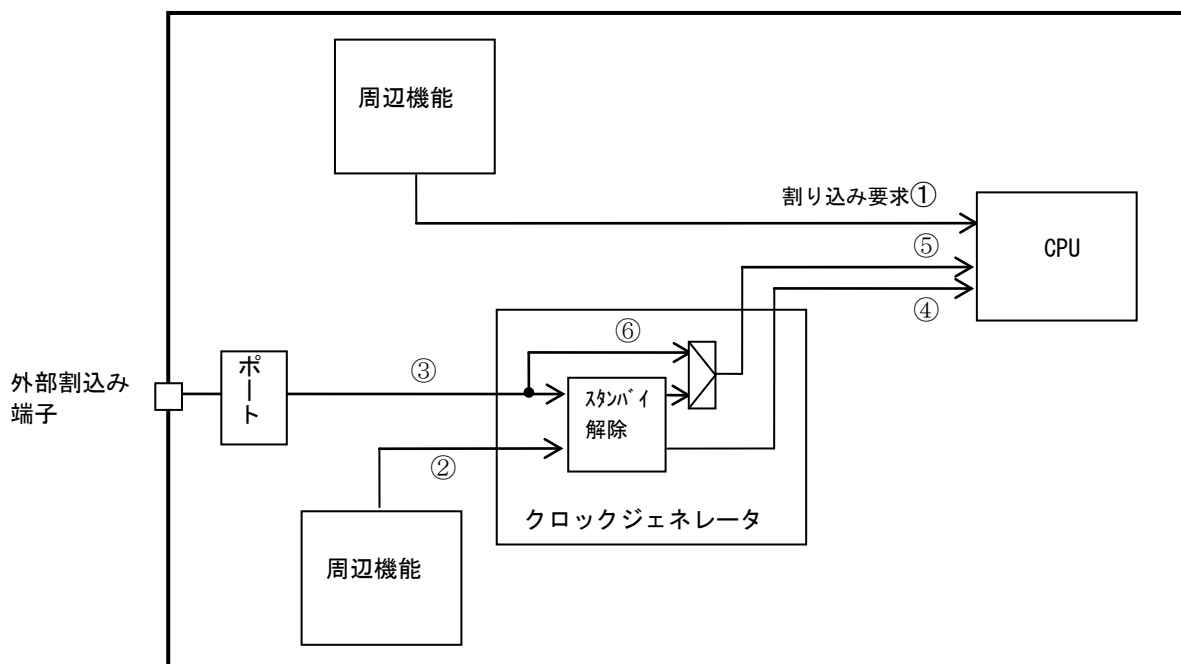


図8-1 割り込みの経路

8.5.1.2 割り込み要因の発生

割り込み要求は、割り込み要因に割り当てられた外部端子、周辺機能、NVIC レジスタの割り込み保留セットレジスタの設定により発生します。

・外部端子からの割り込み

外部割り込み端子を使用する場合、ポートの制御レジスタで端子を割り込み機能に設定します。

・周辺機能の割り込み

周辺機能の割り込みを使用する場合、使用する周辺機能で割り込みが出力されるよう設定する必要があります。

詳細は各章をご覧ください。

・割り込みの強制的な保留

割り込み保留セットレジスタの該当する割り込みのビットをセットすることで、割り込み要因を発生させることができます。

8.5.1.3 割り込み要因の伝達

外部端子／周辺機能から発生した割り込み要求のうち、スタンバイ解除要因にならないものは直接 CPU に接続されます。

スタンバイ解除要因として使用できる割り込みは、CPU を経由して割り込みコントローラに接続されるため、クロックジェネレータの設定が必要です。ただし、外部割り込みについてはスタンバイ解除要因として使用しない場合はクロックジェネレータの設定なしで使用することもできます。

8.5.1.4 外部割り込み端子を使用する際の注意

外部割り込みを使用する際には、予期しない割り込みが発生しないよう以下の点に留意してください。

外部割り込み端子からの入力信号は、入力ディセーブル(PxIE<PxIE>="0")の場合"High"となります。また、外部割り込みをスタンバイ解除要因として使用しない場合(「図 8-1 割り込みの経路」の⑥の経路)、外部割り込み端子からの入力信号がそのまま CPU に伝わります。CPU は"High"入力を割り込みとして認識しますので、入力ディセーブルのまま CPU で該当する割り込みを有効にすると割り込みが発生します。

外部割り込みをスタンバイ解除要因とせずに使用する際には、割り込み端子入力を"Low"レベルとして入力ディセーブルにし、その後 CPU で割り込み許可設定を行ってください。

8.5.1.5 要因一覧

割り込みの要因一覧を表 8-2 に示します。

表8-2 ハードウェア割り込み要因一覧(1/2)

番号	要因	アクティブレベル (スタンバイ解除)	CG 割り込みモード コントロールシグナ
0	INT0	割り込み端子 (PJ0/70pin)	IMCGA
1	INT1	割り込み端子 (PJ1/49pin)	
2	INT2	割り込み端子 (PJ2/86pin)	
3	INT3	割り込み端子 (PJ3/87pin)	
4	INT4	割り込み端子 (PG3/6pin)	
5	INT5	割り込み端子 (PF7/19pin)	IMCGB
6	INTRX0	シリアル受信 (channel.0)	
7	INTTX0	シリアル送信 (channel.0)	
8	INTRX1	シリアル受信 (channel.1)	
9	INTTX1	シリアル送信 (channel.1)	
10	INTSB10	シリアルインターフェース 0	
11	INTSB11	シリアルインターフェース 1	IMCGB
12	INTCECRX	GEC 受信	
13	INTCECTX	GEC 送信	
14	INTRMCRX0	リモコン受信 (channel.0)	「↑」エッジ
15	INTADHP	最優先 AD 変換終了割り込み	
16	INTADMO	AD 変換監視機能割り込み 0	
17	INTADM1	AD 変換監視機能割り込み 1	
18	INTTB0	16bitTMRB コンパレータ一致 0	
19	INTTB1	16bitTMRB コンパレータ一致 1	
20	INTTB2	16bitTMRB コンパレータ一致 2	
21	INTTB3	16bitTMRB コンパレータ一致 3	
22	INTTB4	16bitTMRB コンパレータ一致 4	
23	INTTB5	16bitTMRB コンパレータ一致 5	
24	INTTB6	16bitTMRB コンパレータ一致 6	
25	INTRTC	時計タイマ	「↓」エッジ
26	INTCAP00	16bitTMRB インพุットキャプチャ 00	
27	INTCAP01	16bitTMRB インพุットキャプチャ 01	
28	INTCAP10	16bitTMRB インพุットキャプチャ 10	
29	INTCAP11	16bitTMRB インพุットキャプチャ 11	
30	INTCAP50	16bitTMRB インพุットキャプチャ 50	
31	INTCAP51	16bitTMRB インพุットキャプチャ 51	
32	INTCAP60	16bitTMRB インพุットキャプチャ 60	
33	INTCAP61	16bitTMRB インพุットキャプチャ 61	
34	INT6	割り込み端子 (PJ6/39pin)	任意
35	INT7	割り込み端子 (PJ7/58pin)	
36	INTRX2	シリアル受信 (channel.2)	
37	INTTX2	シリアル送信 (channel.2)	
38	INTSB12	シリアルインターフェース 2	
39	INTRMCRX1	リモコン受信 (channel.1)	「↑」エッジ
40	INTTB7	16bitTMRB コンパレータ一致 7	
41	INTTB8	16bitTMRB コンパレータ一致 8	
42	INTTB9	16bitTMRB コンパレータ一致 9	

表 8-2 ハードウェア割り込み要因一覧 (2/2)

番号	要因		アクティブレベル (スタンバイ解除)	クロック ジェネレータ
43	INTCAP20	16bitTMRB インフットキャプチャ 20		
44	INTCAP21	16bitTMRB インフットキャプチャ 21		
45	INTCAP30	16bitTMRB インフットキャプチャ 30		
46	INTCAP31	16bitTMRB インフットキャプチャ 31		
47	INTCAP40	16bitTMRB インフットキャプチャ 40		
48	INTCAP41	16bitTMRB インフットキャプチャ 41		
49	INTAD	A/D 変換終了		

8.5.1.6 アクティブレベル

アクティブレベルはどのような信号変化を割り込み要因と見なすかを示しています。CPU は割り込み信号が「L」から「H」に変化することで割り込み要因とみなします。各種周辺機能から CPU へ直接割り込み信号が伝わるものは、割り込み要求として「H」パルスを出力するようになっています。

スタンバイ解除要因となる割り込みについては、クロックジェネレータに設定するアクティブレベルは、周辺機能からの割り込み要求は立ち上がり（「↑」）エッジまたは立ち下がり（「↓」）エッジとなり、割り込み端子からの割り込み要求は「H」レベル、「L」レベル、立ち上がり（「↑」）エッジ、立ち下がり（「↓」）エッジから選ぶことができます。

スタンバイ解除要因となる割り込みを使用するときにはクロックジェネレータレジスタの CGIMCGx<INTxEN>ビットを有効にし、CGIMCGx<EMCGx [2:0]>ビットにアクティブレベルを設定します。周辺機能からの割り込み要求のアクティブレベルは表 8-2 で指定されているとおりに設定してください。

クロックジェネレータで検出された割り込みは、「H」レベル信号で CPU に通知されます。



割り込み端子からの割り込み要求をスタンバイ解除要因として使用しない場合には、クロックジェネレータの設定をせずに使用することも可能です。ただし、CPU が割り込み要因として検出するためには、「H」パルスまたは「H」レベルの信号を入力する必要があります。

(注) GEC 受信/送信割り込み、リモコン受信割り込み、時計タイマ割り込みは、スタンバイ解除割り込みとして使用しない場合も INTxEN ビットを「1」に設定し、指定されたアクティブ状態の設定を行ってください。

8.5.2 処理詳細

8.5.2.1 処理の流れ

割り込みの処理の流れを以下に示します。

以下の説明で、 はハードウェアによる処理を、 はソフトウェアによる処理を示しています。

処理	内容	説明
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;">検出のための準備</div> <div style="border: 1px solid black; padding: 5px;">要因発生のための準備</div>	<p>割り込みを検出するための設定を、NVIC レジスタで行います。スタンバイ解除要因となる割り込みは、クロックジェネレータの設定も必要です。</p> <ul style="list-style-type: none"> ○共通 <ul style="list-style-type: none"> NVIC レジスタの設定 ○スタンバイ解除の設定 <ul style="list-style-type: none"> クロックジェネレータの設定 <p>割り込みの要因発生のための設定を行います。割り込みの種類により設定内容が異なります。</p> <ul style="list-style-type: none"> ○外部割り込み <ul style="list-style-type: none"> ポートの設定 ○各周辺機能からの割り込み <ul style="list-style-type: none"> 各周辺機能の設定（使用する周辺機能の章を参照ください） 	<div style="border: 1px solid black; padding: 5px;">8.5.2.2 準備</div>
<div style="border: 2px dashed black; padding: 5px; width: fit-content; margin: 0 auto;">割り込み要因の発生</div>	<p>ハードウェア割り込みの要因が発生します</p>	
<div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <div style="border: 1px solid black; padding: 2px; display: inline-block; margin-right: 10px;">スタンバイ解除 要因以外</div> <div style="border: 1px solid black; padding: 2px; display: inline-block;">スタンバイ解除 要因</div> </div> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;">CG が割り込みを検出 (スタンバイ解除要因)</div>	<p>スタンバイ解除要因となる割り込みは、クロックジェネレータを経由して CPU に接続されています。</p>	<div style="border: 1px solid black; padding: 5px;">8.5.2.3 検出 (CG)</div>

処理	内容	説明
<p style="text-align: center;">↓</p> <div style="border: 1px solid black; padding: 5px; text-align: center;">CPU が割り込みを検出</div>	<p>CPU が割り込みを検出します。</p> <p>複数の割り込み要因が存在する場合、優先順位に従って最も優先度の高い割り込み要因を検出します。</p>	<div style="border: 1px solid black; padding: 5px; text-align: center;">8.5.2.4 検出 (CPU)</div>
<p style="text-align: center;">↓</p> <div style="border: 1px solid black; padding: 5px; text-align: center;">CPU が割り込みを処理</div>	<p>CPU が割り込み処理を行います。</p> <p>スタックにレジスタの内容を退避し、割り込み処理ルーチンへ分岐します。</p>	<div style="border: 1px solid black; padding: 5px; text-align: center;">8.5.2.5 CPU の 処理</div>
<p style="text-align: center;">↓</p> <div style="border: 1px solid black; padding: 5px; text-align: center;">割り込み サービスルーチン実行</div>	<p>必要な処理をプログラミングしてください。</p> <p>必要に応じて割り込み要因の取り下げを行ってください。</p>	<div style="border: 1px solid black; padding: 5px; text-align: center;">8.5.2.6 割り込み サービス ルーチン</div>
<p style="text-align: center;">↓</p> <div style="border: 1px solid black; padding: 5px; text-align: center;">元のプログラムへ復帰</div>	<p>割り込み処理ルーチンから通常の処理プログラムに復帰します。</p>	

8.5.2.2 準備

割り込みの準備を行うときには、設定途中で不要な割り込みの発生を防ぐために設定の順番に注意が必要です。

割り込みの使用開始、または設定変更のときの基本的な順序は、まず、CPU で割り込みを禁止し、次に割り込み経路で CPU から遠いところから設定を行い、最後に CPU で割り込みを許可します。

クロックジェネレータの設定を行うときには、条件の設定を行った後、不要な割り込みが発生しないよう、クロックジェネレータ内部の割り込み情報をクリアしてから割り込み許可の設定を行います。

以下に設定の手順と、手順ごとの具体的な設定方法を示します。

- (1) CPU 割り込み禁止
- (2) CPU 割り込み設定
- (3) 要因の準備①（外部割り込み）
- (4) 要因の準備②（周辺機能からの割り込み）
- (5) 要因の準備③（割り込み保留セットレジスタ）
- (6) クロックジェネレータの設定
- (7) CPU 割り込み許可

(1) CPU 割り込み禁止

CPU を割り込み禁止状態にするには、PRIMASK レジスタに "1" をセットします。これにより、マスク不能割り込みとハードフォールト例外以外のすべての割り込みと例外がマスクされます。

このレジスタをセットするためには "MSR" 命令を使用します。

●割り込みマスクレジスタ		
PRIMASK	←	「1」（割り込み禁止）

(注) 「m」は該当ビットを示します。

(2) CPU 割り込み設定

NVIC レジスタの割り込み優先度レジスタで優先度の設定を行います。

このレジスタは、8 ビットごとに各割り込み要因に割り当てられていますが、製品ごとに構成するビット数が異なります。8 ビットの構成の場合「0」から「255」までの優先度を設定することができます。最も高い優先度は「0」です。複数の要因を同じ優先度に設定した場合、番号の小さい割り込みの優先度が高くなります。

●NVIC レジスタ		
割り込み優先度<m>	←	「優先度」

(3) 要因の準備①（外部割り込み）

外部割り込みを使用する場合、該当する端子のポートの設定を行います。機能端子として使用するため、該当するポートのファンクションレジスタ PnFRx[m]に「1」を、ポートを入力として使用するために PnIE[m]を「1」に設定します。

●ポートレジスタ		
PnFRx<PnmFRx>	←	「1」
PnIE<PnmIE>	←	「1」

(注) 「n」は該当ポート番号、「m」は該当ビット、「x」はファンクションレジスタ番号を示します。

STOP 以外のモードでは、PnIE で入力イネーブル設定であれば PnFR の設定によらず割り込みの入力が有効になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。

(4) 要因の準備②（周辺機能からの割り込み）

周辺機能からの割り込みを使用する場合、設定方法は周辺機能によって異なります。各周辺機能の章をご覧ください。

(5) 要因の準備③（割り込み保留セットレジスタ）

割り込み保留セットレジスタで割り込みを発生する場合、該当するビットに「1」をセットします。

●NVIC レジスタ		
割り込み保留セット<m>	←	「1」

(注) 「m」は該当ビットを示します。

(6) クロックジェネレータの設定

スタンバイ解除要因となる割り込みは、クロックジェネレータの CGIMCG レジスタでアクティブレベルと割り込み許可の設定を行います。CGIMCG レジスタは要因ごとの設定レジスタです。

割り込み許可の前に、不要な割り込み発生を防止するため割り込み要求のクリアを ICRCG レジスタで行います。CGICRCG レジスタは、要因に対応した値を書き込むことで保持されていた割り込み要求をクリアすることができます。具体的な値は、「8.6.3.5 CG 割り込み要求クリアレジスタ」を参照してください。

割り込み端子からの割り込み要求をスタンバイ解除要因として使用しない場合、クロックジェネレータの設定を行わずに使用することもできます。ただし、CPU が割り込み要因として検出するためには、「H」パルスまたは「H」レベルの信号を入力する必要があります。

●クロックジェネレータレジスタ		
CGIMCGn<EMCGm>	←	アクティブレベル
CGICRCG<ICRCG>	←	使用する要因に対応する値
CGIMCGn<INTmEN>	←	「1」（割り込み許可）

(注) 「n」はレジスタ番号、「m」は割り込み要因固有の番号を示します。

(7) CPU 割り込み許可

最後に CPU の割り込み許可の設定をします。

割り込み保留クリアレジスタで保留状態の割り込みをクリアし、割り込みイネーブルセットレジスタで割り込みを許可します。これらのレジスタは1ビットずつ各割り込み要因に割り当てられています。

割り込み保留クリアレジスタの該当する割り込みのビットに「1」を書くことで保留されている要因をクリアすることができ、割り込みイネーブルセットレジスタの該当する割り込みのビットに「1」を書くことで割り込みを許可することができます。

ただし、割り込み保留セットレジスタの設定で割り込みが発生する場合、この操作を行うと割り込み要因がクリアされてしまうため不要です。

●NVIC レジスタ		
割り込み保留クリア<m>	←	「1」
割り込みイネーブルセット<m>	←	「1」

●割り込みマスクレジスタ		
PRIMASK	←	「0」

(注) 「m」は該当ビットを示します。

8.5.2.3 検出（クロックジェネレータ）

スタンバイ解除要因となる割り込みは、クロックジェネレータに設定されたアクティブレベルにしたがって検出され CPU に伝えられます。

アクティブレベルが立ち上がりまたは立下りエッジの割り込み要因は、検出された後クロックジェネレータで要因が保持されますが、「H」レベルまたは「L」レベル設定の割り込み要因は、アクティブレベルから変化すると割り込み要因がなくなったとみなされるため、割り込み検出までレベルを保つ必要があります。

クロックジェネレータは割り込みを検出すると CG 割り込み要求クリア（CGICRCG）レジスタで解除されるまで「H」レベルの割り込み信号を CPU に出力します。解除を行わずに復帰すると再度同じ割り込みが検出されますので、割り込み処理ルーチン内で割り込みの解除を行ってください。

8.5.2.4 検出（CPU）

CPU は優先順位に従って最も優先度の高い割り込み要因を検出します。

8.5.2.5 CPU の処理

割り込みが検出されると、CPU はスタックへ PC、PSR、r0-r3、r12、LR を退避し、検出した割り込みの割り込み処理ルーチンへ分岐します。

8.5.2.6 割り込み処理ルーチンでの処理（要因の取り下げ）

割り込み処理ルーチンではアプリケーションにより必要な内容をプログラミングしますが、ここでは推奨する処理と要因の取り下げについて説明します。

（1）割り込み処理ルーチンでの処理

通常、割り込み処理ルーチンでは必要なレジスタの退避と割り込み処理を行います。Cortex-M3 コアは自動的に PC、PSR、r0-r3、r12、LR をスタックへ退避するため、これらのレジスタをユーザプログラムで退避する必要はありません。

その他のレジスタについては必要に応じて退避します。

割り込み処理ルーチン実行中でも、より高い優先度の割り込みや NMI などの例外は受け付けられます。そのため書き換わる可能性のある、汎用レジスタを退避することを推奨します。

(2) 割り込み要因の取り下げ

スタンバイ解除要因となる割り込みについては、CG 割り込み要求クリアレジスタ (CGICRCG) で割り込み要求を解除する必要があります。

アクティブレベルがレベル検出の割り込みの場合、要因そのものを取り下げない限り割り込み要求は存在し続けるため、まず要因を取り下げる必要があります。レベル検出の場合は、要因が取り下げられるとクロックジェネレータからの割り込み要求信号は自動的に取り下げられます。

エッジ検出の場合は CGICRCG レジスタに該当する割り込みの値を設定することで要因は取り下げられ、再度有効なエッジが発生したときに改めて要因として認識されます。

8.6 例外/割り込み関連レジスタ

以下に、本章で説明した CPU の NVIC レジスタとクロックジェネレータレジスタとアドレスを示します。

8.6.1 レジスタ一覧

●NVIC レジスタ	
SysTick 制御およびステータスレジスタ	0xE000_E010
SysTick リロード値レジスタ	0xE000_E014
SysTick 現在値レジスタ	0xE000_E018
SysTick 較正值レジスタ	0xE000_E01C
割り込みイネーブルセットレジスタ 1	0xE000_E100
割り込みイネーブルセットレジスタ 2	0xE000_E104
割り込みイネーブルクリアレジスタ 1	0xE000_E180
割り込みイネーブルクリアレジスタ 2	0xE000_E184
割り込み保留セットレジスタ 1	0xE000_E200
割り込み保留セットレジスタ 2	0xE000_E204
割り込み保留クリアレジスタ 1	0xE000_E280
割り込み保留クリアレジスタ 2	0xE000_E284
割り込み優先度レジスタ	0xE000_E400-0xE000_E430
ベクタテーブルオフセットレジスタ	0xE000_ED08
システムハンドラ優先度レジスタ	0xE000_ED18, 0xE000_ED1C, 0xE000_ED20
システムハンドラ制御および状態レジスタ	0xE000_ED24

●クロックジェネレータレジスタ		
CGICRCG	CG 割り込み要求クリアレジスタ	0x4004_0214
CGNMIFLG	NMI フラグレジスタ	0x4004_0218
CGRSTFLG	リセットフラグレジスタ	0x4004_021C
CGIMCGA	CG 割り込みモードコントロールレジスタ A	0x4004_0220
CGIMCGB	CG 割り込みモードコントロールレジスタ B	0x4004_0224
CGIMCGC	CG 割り込みモードコントロールレジスタ C	0x4004_0228
CGIMCGD	CG 割り込みモードコントロールレジスタ D	0x4004_022C

8.6.2 NVIC レジスタ

8.6.2.1 SysTick 制御およびステータスレジスタ

	7	6	5	4	3	2	1	0
bit Symbol						CLK SOURCE	TICKINT	ENABLE
Read/Write	R					R/W	R/W	R/W
リセット後	0					0	0	0
機能	リードすると"0"が読めます					0:外部参照クロック 1:コアクロック	0: SysTickを保留しない 1: SysTickを保留する	0: ディセーブル 1: イネーブル
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								COUNT FLAG
Read/Write	R							R/W
リセット後	0							0
機能	リードすると"0"が読めます							0: タイマは0になっていない 1: タイマが0になった
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

- <bit0> <ENABLE> 1 = リロード値レジスタの値をカウンタにロードし、動作を開始します。
0 = タイマは禁止されています。
- <bit1> <TICKINT> 1 = SysTick 例外を保留します。
0 = SysTick 例外を保留しません。
- <bit2> <CLKSOURCE> 0 = 外部参照クロック (fosc/32) (注)
1 = コアクロック (fsys)
- <bit16> <COUNTFLAG> 1 = 最後の読み出しの後にタイマが"0"になったことを示します。
このレジスタのいずれかの部分を読み出すとこのビットはクリアされます。

注) 本製品では外部参照クロックとして fosc を 32 分周したクロックが使用されます。

8.6.2.2 SysTick リロード値レジスタ

	7	6	5	4	3	2	1	0
bit Symbol	RELOAD							
Read/Write	R/W							
リセット後	不定							
機能	リロード値							
	15	14	13	12	11	10	9	8
bit Symbol	RELOAD							
Read/Write	R/W							
リセット後	不定							
機能	リロード値							
	23	22	21	20	19	18	17	16
bit Symbol	RELOAD							
Read/Write	R/W							
リセット後	不定							
機能	リロード値							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

<bit23:0> <RELOAD> タイマが“0”になったときに SysTick 現在値レジスタにロードする値を設定します。

8.6.2.3 SysTick 現在値レジスタ

	7	6	5	4	3	2	1	0
bit Symbol	CURRENT							
Read/Write	R/W							
リセット後	不定							
機能	[リード] SysTick タイマ現在値 [ライト] クリア							
	15	14	13	12	11	10	9	8
bit Symbol	CURRENT							
Read/Write	R/W							
リセット後	不定							
機能	[リード] SysTick タイマ現在値 [ライト] クリア							
	23	22	21	20	19	18	17	16
bit Symbol	CURRENT							
Read/Write	R/W							
リセット後	不定							
機能	[リード] SysTick タイマ現在値 [ライト] クリア							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

<bit23:0> <CURRENT> [リード] SysTick タイマ現在値です。

[ライト] 任意の値を書き込むとタイマカウントがクリアされます。
このレジスタをクリアすることで、SysTick 制御およびステータスレジスタの COUNTFLAG ビットもクリアされます。

8.6.2.4 SysTick 較正值レジスタ

	7	6	5	4	3	2	1	0
bit Symbol	TENMS							
Read/Write	R							
リセット後	1	1	0	0	0	1	0	0
機能	較正值 (注)							
	15	14	13	12	11	10	9	8
bit Symbol	TENMS							
Read/Write	R							
リセット後	0	0	0	0	1	0	0	1
機能	較正值 (注)							
	23	22	21	20	19	18	17	16
bit Symbol	TENMS							
Read/Write	R							
リセット後	0	0	0	0	0	0	0	0
機能	較正值 (注)							
	31	30	29	28	27	26	25	24
bit Symbol	NOREF	SKEW						
Read/Write	R	R	R					
リセット後	0	0	0					
機能	0: 参照クロックあり 1: 参照クロックなし	0: 較正值は 10ms 1: 較正值は 10ms でない	リードすると "0" が読めます					

<bit23:0> < TENMS > 外部参照クロックで 10ms をカウントするために使用するリロード値 (0x9C4) です。
(注)

<bit30> <SKEW> 1 = 較正值は正確に 10ms でないことを示します。

<bit31> <NOREF> 1 = カウントクロックが提供されていないことを示します。

(注) 本製品では、カウントクロックとして X1 端子から入力されるクロックを 32 分周したクロックが使用されます。
較正值レジスタには、X1 から 8MHz のクロックを入力した場合に 10ms を計測可能な値が設定されています。

(注) マルチショットで使用する場合、この値を-1 して使用してください。

8.6.2.5 割り込みイネーブルセットレジスタ 1

	7	6	5	4	3	2	1	0	
bit Symbol	SETENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号7 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号6 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号5 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号4 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号3 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号2 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号1 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号0 [ライト] 1:許可 [リード] 0:禁止 1:許可	
	15	14	13	12	11	10	9	8	
bit Symbol	SETENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号15 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号14 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号13 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号12 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号11 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号10 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号9 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号8 [ライト] 1:許可 [リード] 0:禁止 1:許可	
	23	22	21	20	19	18	17	16	
bit Symbol	SETENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号23 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号22 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号21 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号20 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号19 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号18 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号17 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号16 [ライト] 1:許可 [リード] 0:禁止 1:許可	
	31	30	29	28	27	26	25	24	
bit Symbol	SETENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号31 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号30 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号29 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号28 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号27 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号26 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号25 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号24 [ライト] 1:許可 [リード] 0:禁止 1:許可	

〈bit31:0〉 〈SETENA〉 割り込みを許可、またはどの割り込みが許可されているかを確認できます。

“1” をライトすることで該当する割り込みを許可します。“0” の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

0 = 禁止

1 = 許可

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.6 割り込みイネーブルセットレジスタ 2

	7	6	5	4	3	2	1	0	
bit Symbol	SETENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号 39 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 38 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 37 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 36 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 35 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 34 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 33 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 32 [ライト] 1:許可 [リード] 0:禁止 1:許可	
	15	14	13	12	11	10	9	8	
bit Symbol	SETENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号 47 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 46 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 45 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 44 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 43 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 42 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 41 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 40 [ライト] 1:許可 [リード] 0:禁止 1:許可	
	23	22	21	20	19	18	17	16	
bit Symbol							SETENA		
Read/Write	R						R/W		
リセット後	0						0	0	
機能	リードすると"0"が読めます						割り込み番号 49 [ライト] 1:許可 [リード] 0:禁止 1:許可	割り込み番号 48 [ライト] 1:許可 [リード] 0:禁止 1:許可	
	31	30	29	28	27	26	25	24	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると"0"が読めます								

<bit17:0> <SETENA> 割り込みを許可、またはどの割り込みが許可されているかを確認できます。

“1”をライトすることで該当する割り込みを許可します。“0”の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

0 = 禁止

1 = 許可

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.7 割り込みイネーブルクリアレジスタ 1

	7	6	5	4	3	2	1	0	
bit Symbol	CLRENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号7 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号6 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号5 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号4 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号3 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号2 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号1 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号0 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	
	15	14	13	12	11	10	9	8	
bit Symbol	CLRENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号15 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号14 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号13 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号12 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号11 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号10 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号9 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号8 [ライト] 1: 禁止許可 [リード] 0: 禁止 1: 許可	
	23	22	21	20	19	18	17	16	
bit Symbol	CLRENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号23 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号22 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号21 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号20 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号19 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号18 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号17 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号16 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	
	31	30	29	28	27	26	25	24	
bit Symbol	CLRENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号31 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号30 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号29 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号28 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号27 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号26 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号25 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号24 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	

〈bit31:0〉 〈CLRENA〉 割り込みを禁止、またはどの割り込みが禁止されているかを確認できます。

“1” をライトすることで該当する割り込みを禁止します。“0” の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

0 = 禁止

1 = 許可

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.8 割り込みイネーブルクリアレジスタ 2

	7	6	5	4	3	2	1	0	
bit Symbol	CLRENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号 39 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 38 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 37 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 36 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 35 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 34 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 33 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 32 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	
	15	14	13	12	11	10	9	8	
bit Symbol	CLRENA								
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	割り込み番号 47 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 46 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 45 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 44 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 43 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 42 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 41 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 40 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	
	23	22	21	20	19	18	17	16	
bit Symbol							CLRENA		
Read/Write	R						R/W		
リセット後	0						0	0	
機能	リードすると"0"が読めます						割り込み番号 49 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	割り込み番号 48 [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可	
	31	30	29	28	27	26	25	24	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると"0"が読めます								

<bit17:0> <CLRENA> 割り込みを禁止、またはどの割り込みが禁止されているかを確認できます。

“1”をライトすることで該当する割り込みを禁止します。“0”の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

0 = 禁止

1 = 許可

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.9 割り込み保留セットレジスタ 1

	7	6	5	4	3	2	1	0
bit Symbol	SETPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 7 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 6 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 5 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 4 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 3 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 2 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 1 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 0 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり
	15	14	13	12	11	10	9	8
bit Symbol	SETPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 15 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 14 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 13 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 12 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 11 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 10 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 9 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 8 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり
	23	22	21	20	19	18	17	16
bit Symbol	SETPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 23 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 22 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 21 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 20 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 19 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 18 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 17 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 16 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり
	31	30	29	28	27	26	25	24
bit Symbol	SETPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 31 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 30 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 29 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 28 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 27 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 26 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 25 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 24 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり

<bit31:0> <SETPEND> 割り込みを強制的に保留します。また、どの割り込みが保留されているかを確認できます。

“1”をライトすることで該当する割り込みを保留します。ただし、すでに保留されている割り込みおよび禁止されている割り込みに対しては無効です。“0”の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

0 = 保留されていない

1 = 保留されている

このレジスタのビットをクリアするには、割り込み保留クリアレジスタの該当するビットに“1”をセットします。

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.10 割り込み保留セットレジスタ 2

	7	6	5	4	3	2	1	0	
bit Symbol	SETPEND								
Read/Write	R/W								
リセット後	不定								
機能	割り込み番号 39 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 38 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 37 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 36 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 35 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 34 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 33 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 32 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	
	15	14	13	12	11	10	9	8	
bit Symbol	SETPEND								
Read/Write	R/W								
リセット後	不定								
機能	割り込み番号 47 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 46 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 45 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 44 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 43 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 42 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 41 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 40 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	
	23	22	21	20	19	18	17	16	
bit Symbol							SETPEND		
Read/Write	R						R/W		
リセット後	0						不定		
機能	リードすると"0"が読めます						割り込み番号 49 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	割り込み番号 48 [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり	
	31	30	29	28	27	26	25	24	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると"0"が読めます								

<bit17:0> <SETPEND> 割り込みを強制的に保留します。また、どの割り込みが保留されているかを確認できます。

“1”をライトすることで該当する割り込みを保留します。ただし、すでに保留されている割り込みおよび禁止されている割り込みに対しては無効です。“0”の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

0 = 保留されていない
1 = 保留されている

このレジスタのビットをクリアするには、割り込み保留クリアレジスタの該当するビットに“1”をセットします。

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.11 割り込み保留クリアレジスタ 1

	7	6	5	4	3	2	1	0
bit Symbol	CLRPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 7 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 6 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 5 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 4 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 3 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 2 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 1 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 0 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり
	15	14	13	12	11	10	9	8
bit Symbol	CLRPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 15 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 14 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 13 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 12 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 11 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 10 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 9 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 8 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり
	23	22	21	20	19	18	17	16
bit Symbol	CLRPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 23 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 22 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 21 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 20 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 19 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 18 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 17 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 16 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり
	31	30	29	28	27	26	25	24
bit Symbol	CLRPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 31 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 30 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 29 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 28 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 27 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 26 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 25 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 24 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり

<bit31:0> <GLRPEND> 保留された割り込みをクリアします。また、読み出すと保留されているか確認できます。

“1”をライトすることで該当する保留された割り込みをクリアします。ただし、すでに処理が開始されている割り込みに対しては無効です。“0”の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

0 = 保留されていない

1 = 保留されている

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.12 割り込み保留クリアレジスタ 2

	7	6	5	4	3	2	1	0
bit Symbol	CLRPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 39 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 38 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 37 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 36 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 35 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 34 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 33 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 32 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり
	15	14	13	12	11	10	9	8
bit Symbol	CLRPEND							
Read/Write	R/W							
リセット後	不定							
機能	割り込み番号 47 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 46 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 45 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 44 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 43 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 42 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 41 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 40 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり
	23	22	21	20	19	18	17	16
bit Symbol							CLRPEND	
Read/Write	R						R/W	
リセット後	0						不定	
機能	リードすると"0"が読めます						割り込み番号 49 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり	割り込み番号 48 [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

<bit17:0> <CLRPEND> 保留された割り込みをクリアします。また、読み出すと保留されているか確認できます。

“1”をライトすることで該当する保留された割り込みをクリアします。ただし、すでに処理が開始されている割り込みに対しては無効です。“0”の書き込みは意味を持ちません。

リードの場合、該当する割り込みが以下の状態であることを示します。

- 0 = 保留されていない
- 1 = 保留されている

(注) 割り込みの内容と割り込み番号については、8.5.1.4を参照してください。

8.6.2.13 割り込み優先度レジスタ

割り込み優先度レジスタは、各割り込みに対し 8 ビットごとの構成になっています。

割り込み番号と対応する割り込み優先度レジスタのアドレスは以下のとおりです。

	31	24	23	16	15	8	7	0
0xE000_E400	PRI_3		PRI_2		PRI_1			PRI_0
0xE000_E404	PRI_7		PRI_6		PRI_5			PRI_4
0xE000_E408	PRI_11		PRI_10		PRI_9			PRI_8
0xE000_E40C	PRI_15		PRI_14		PRI_13			PRI_12
0xE000_E410	PRI_19		PRI_18		PRI_17			PRI_16
0xE000_E414	PRI_23		PRI_22		PRI_21			PRI_20
0xE000_E418	PRI_27		PRI_26		PRI_25			PRI_24
0xE000_E41C	PRI_31		PRI_30		PRI_29			PRI_28
0xE000_E420	PRI_35		PRI_34		PRI_33			PRI_32
0xE000_E424	PRI_39		PRI_38		PRI_37			PRI_36
0xE000_E428	PRI_43		PRI_42		PRI_41			PRI_40
0xE000_E42C	PRI_47		PRI_46		PRI_45			PRI_44
0xE000_E430	-		-		PRI_49			PRI_48

各割り込みに割り当てられている 8 ビットのうち何ビットを優先度の設定に使用できるかは製品により異なります。本製品では、3 ビットで優先度を設定することができます。

以下に、代表として割り込み番号 0～3 の割り込み優先度レジスタの構成を示します。未使用のビットはリードすると“0”が読め、ライトは無視されます。

	7	6	5	4	3	2	1	0
bit Symbol	PRI_0							
Read/Write	R/W			R				
リセット後	0			0				
機能	割り込み番号 0 優先度			リードすると“0”が読めます				
	15	14	13	12	11	10	9	8
bit Symbol	PRI_1							
Read/Write	R/W			R				
リセット後	0			0				
機能	割り込み番号 1 優先度			リードすると“0”が読めます				
	23	22	21	20	19	18	17	16
bit Symbol	PRI_2							
Read/Write	R/W			R				
リセット後	0			0				
機能	割り込み番号 2 優先度			リードすると“0”が読めます				
	31	30	29	28	27	26	25	24
bit Symbol	PRI_3							
Read/Write	R/W			R				
リセット後	0			0				
機能	割り込み番号 3 優先度			リードすると“0”が読めます				

<bit7:5> <PRI_0> 割り込み番号0の優先度
<bit15:13> <PRI_1> 割り込み番号1の優先度
<bit23:21> <PRI_2> 割り込み番号2の優先度
<bit31:29> <PRI_3> 割り込み番号3の優先度

8.6.2.14 ベクタテーブルオフセットレジスタ

	7	6	5	4	3	2	1	0	
bit Symbol	TBLOFF								
Read/Write	R/W	R							
リセット後	0	0							
機能	オフセット値	リードすると”0”が読めます							
	15	14	13	12	11	10	9	8	
bit Symbol	TBLOFF								
Read/Write	R/W								
リセット後	0								
機能	オフセット値								
	23	22	21	20	19	18	17	16	
bit Symbol	TBLOFF								
Read/Write	R/W								
リセット後	0								
機能	オフセット値								
	31	30	29	28	27	26	25	24	
bit Symbol			TBLBASE	TBLOFF					
Read/Write	R		R/W	R/W					
リセット後	0		0	0					
機能	リードすると”0”が読めます		テーブルベース	オフセット値					

<bit28:7> <TBLOFF> TBLBASE で指定した領域の先頭からのオフセット値を設定します。オフセットはテーブルにある例外の数に基づいてアラインされる必要があります。16 個までの割り込みが使える、最小のアライメントは 32 ワードになります。割り込みの数がより多い場合は、次の 2 のべき乗まで切り上げて、アライメントを調整する必要があります。

<bit29> <TBLBASE> ベクタテーブルを置くメモリ空間を指定します。
 0 = コード空間
 1 = SRAM 空間

8.6.2.15 システムハンドラ優先度レジスタ

システムハンドラ優先度レジスタは、各例外に対し 8 ビットごとの構成になっています。
例外と対応する割り込み優先度レジスタのアドレスは以下のとおりです。

	31	24	23	16	15	8	7	0
0xE000_ED18	PRI_7		PRI_6 (用法フォールト)		PRI_5 (バスフォールト)		PRI_4 (メモリ管理)	
0xE000_ED1C	PRI_11 (SVCall)		PRI_10		PRI_9		PRI_8	
0xE000_ED20	PRI_15 (SysTick)		PRI_14 (PendSV)		PRI_13		PRI_12 デバッグモニタ	

各割り込みに割り当てられている 8 ビットのうち何ビットを優先度の設定に使用できるかは製品により異なります。本製品では、3 ビットで優先度を設定することができます。

以下に、代表として割り込み番号 0～3 の割り込み優先度レジスタの構成を示します。未使用のビットはリードすると“0”が読め、ライトは無視されます。

	7	6	5	4	3	2	1	0
bit Symbol	PRI_4							
Read/Write	R/W			R				
リセット後	0			0				
機能	メモリ管理 優先度			リードすると“0”が読めます				
	15	14	13	12	11	10	9	8
bit Symbol	PRI_5							
Read/Write	R/W			R				
リセット後	0			0				
機能	バスフォールト 優先度			リードすると“0”が読めます				
	23	22	21	20	19	18	17	16
bit Symbol	PRI_6							
Read/Write	R/W			R				
リセット後	0			0				
機能	用法フォールト 優先度			リードすると“0”が読めます				
	31	30	29	28	27	26	25	24
bit Symbol	PRI_7							
Read/Write	R/W			R				
リセット後	0			0				
機能	予約			リードすると“0”が読めます				

8.6.2.16 システムハンドラ制御および状態レジスタ

	7	6	5	4	3	2	1	0
bit Symbol	SVCALL ACT				USGFAULT ACT		BUSFAULT ACT	MEMFAULT ACT
Read/Write	R/W	R			R/W	R	R/W	R/W
リセット後	0	0			0	0	0	0
機能	SVCall 0:アクティブでない 1:アクティブ	リードすると"0"が読めます			用法フォールト 0:アクティブでない 1:アクティブ	リードすると"0"が読めます	バスフォールト 0:アクティブでない 1:アクティブ	メモリ管理 0:アクティブでない 1:アクティブ
	15	14	13	12	11	10	9	8
bit Symbol	SVCALL PENDED	BUSFAULT PENDED	MEMFAULT PENDED	USGFAULT PENDED	SYSTICK ACT	PENDSV ACT		MONITOR ACT
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
リセット後	0	0	0	0	0	0	0	0
機能	SVCall 0:保留なし 1:保留	バスフォールト 0:保留なし 1:保留	メモリ管理 0:保留なし 1:保留	用法フォールト 0:保留なし 1:保留	SysTick 0:アクティブでない 1:アクティブ	PendSV 0:アクティブでない 1:アクティブ	リードすると"0"が読めます	デバッグモニタ 0:アクティブでない 1:アクティブ
	23	22	21	20	19	18	17	16
bit Symbol						USGFAULT ENA	BUSFAULT ENA	MEMFAULT ENA
Read/Write	R					R/W	R/W	R/W
リセット後	0					0	0	0
機能	リードすると"0"が読めます					用法フォールト 0:禁止 1:許可	バスフォールト 0:禁止 1:許可	メモリ管理 0:禁止 1:許可
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

- <bit0> <MEMFAULTACT> メモリ管理例外がアクティブな場合“1”となります。
- <bit1> <BUSFAULTACT> バスフォールト例外がアクティブな場合“1”となります。
- <bit3> <USGFALACT> 用法フォールト例外がアクティブな場合“1”となります。
- <bit7> <SVCALLACT> SVCAll 例外がアクティブな場合“1”となります。
- <bit8> <MONITORACT> デバッグモニタ例外がアクティブな場合“1”となります。
- <bit10> <PENDSVACT> PendSV 例外がアクティブな場合、“1”となります。
- <bit11> <SYSTICKACT> SysTick 例外がアクティブな場合、“1”となります。
- <bit12> <USGFAULTPENDED> 用法フォールト例外が保留されている場合、“1”となります。
- <bit13> <MEMFAULTPENDED> メモリ管理例外が保留されている場合、“1”となります。
- <bit14> <BUSFAULTPENDED> バスフォールト例外が保留されている場合、“1”となります。
- <bit15> <SVCALLPENDED> SVCAll 例外が保留されている場合、“1”となります。

- <bit16> <MEMFAULTENA> メモリ管理例外を禁止する場合“0”、許可する場合“1”を設定します。
- <bit17> <BUSFAULTENA> バスフォールト例外を禁止する場合“0”、許可する場合“1”を設定します。
- <bit18> <USGFAULTENA> 用法フォールト例外を禁止する場合“0”、許可する場合“1”を設定します。

(注) アクティブビットの書き替えは、スタックの内容の更新等行いませんので注意して行ってください。

8.6.3 クロックジェネレータレジスタ

8.6.3.1 CG 割り込みモードコントロールレジスタ A

CGIMCGA

	7	6	5	4	3	2	1	0
bit Symbol		EMCG02	EMCG01	EMCG00	EMST01	EMST00		INT0EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	不定	0
機能	リードすると"0"が読めます	INT0スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT0スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると不定値が読まれます	INT0解除入力 0: ディセーブル 1: イネーブル
	15	14	13	12	11	10	9	8
bit Symbol		EMCG12	EMCG11	EMCG10	EMST11	EMST10		INT1EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	不定	0
機能	リードすると"0"が読めます	INT1スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT1スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると不定値が読まれます	INT1解除入力 0: ディセーブル 1: イネーブル
	23	22	21	20	19	18	17	16
bit Symbol		EMCG22	EMCG21	EMCG20	EMST21	EMST20		INT2EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	不定	0
機能	リードすると"0"が読めます	INT2スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT2スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると不定値が読まれます	INT2解除入力 0: ディセーブル 1: イネーブル
	31	30	29	28	27	26	25	24
bit Symbol		EMCG32	EMCG31	EMCG30	EMST31	EMST30		INT3EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	不定	0
機能	リードすると"0"が読めます	INT3スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT3スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると不定値が読まれます	INT3解除入力 0: ディセーブル 1: イネーブル

(注1) <EMSTx>は、<EMCGx[2:0]>が"100"の両エッジ設定のときのみ有効です。<EMSTx>を参照することにより、スタンバイ解除に使用されたアクティブ状態を確認することが出来ます。GGICRCGレジスタで割り込みをクリアすると<EMSTx>もクリアされます。

(注2) エッジ設定と同時に<INTxEN>を設定しないでください。エッジ設定を行ってから<INTxEN>を設定してください。

8.6.3.2 CG 割り込みモードコントロールレジスタ B

CGIMCGB

	7	6	5	4	3	2	1	0
bit Symbol		EMCG42	EMCG41	EMCG40	EMST41	EMST40		INT4EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INT4スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT4スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INT4解除入力 0: ディセーブル 1: イネーブル
	15	14	13	12	11	10	9	8
bit Symbol		EMCG52	EMCG51	EMCG50	EMST51	EMST50		INT5EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INT5スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT5スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INT5解除入力 0: ディセーブル 1: イネーブル
	23	22	21	20	19	18	17	16
bit Symbol		EMCG62	EMCG61	EMCG60	EMST61	EMST60		INT6EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INTCECRXスタンバイ解除要求のアクティブ状態を設定 以下の設定で使用してください 011: 立ち上がリエッジ			INTCECRXスタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INTCECRX解除入力 0: ディセーブル 1: イネーブル
	31	30	29	28	27	26	25	24
bit Symbol		EMCG72	EMCG71	EMCG70	EMST71	EMST70		INT7EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INTRMCRX0スタンバイ解除要求のアクティブ状態を設定 以下の設定で使用してください 011: 立ち上がリエッジ			INTRMCRX0スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INTRMCRX0解除入力 0: ディセーブル 1: イネーブル

(注1) EMSTxx を参照することにより、スタンバイ解除に使用されたアクティブ状態を確認することが出来ます。

(注2) エッジ設定と同時に<INTxEN>を設定しないでください。エッジ設定を行ってから<INTxEN>を設定してください。

8.6.3.3 CG 割り込みモードコントロールレジスタ C

CGIMCGC

	7	6	5	4	3	2	1	0
bit Symbol		EMCG82	EMCG81	EMCG80	EMST81	EMST80		INT8EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INTRTCスタンバイ解除要求のアクティブ状態を設定 以下の設定で使用してください 010: 立ち下がリエッジ			INTRTCスタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INTRTC解除入力 0: ディセーブル 1: イネーブル
	15	14	13	12	11	10	9	8
bit Symbol		EMCG92	EMCG91	EMCG90	EMST91	EMST90		INT9EN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INT6スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT6スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INT6解除入力 0: ディセーブル 1: イネーブル
	23	22	21	20	19	18	17	16
bit Symbol		EMCGA2	EMCGA1	EMCGA0	EMSTA1	EMSTA0		INTAEN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INT7スタンバイ解除要求のアクティブ状態を設定 (101~111:設定禁止) 000: "L" レベル 001: "H" レベル 010: 立ち下がリエッジ 011: 立ち上がリエッジ 100: 両エッジ			INT7スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INT7解除入力 0: ディセーブル 1: イネーブル
	31	30	29	28	27	26	25	24
bit Symbol		EMCGB2	EMCGB1	EMCGB0	EMSTB1	EMSTB0		INTBEN
Read/Write	R	R/W			R		R	R/W
リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INTRMCRX1スタンバイ解除要求のアクティブ状態を設定 以下の設定で使用してください 011: 立ち上がリエッジ			INTRMCRX1スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がリエッジ 10: 立ち下がリエッジ 11: 両エッジ		リードすると"0"が読めます	INTRMCRX1解除入力 0: ディセーブル 1: イネーブル

(注1) EMSTxx を参照することにより、スタンバイ解除に使用されたアクティブ状態を確認することが出来ます。

(注2) エッジ設定と同時に<INTxEN>を設定しないでください。エッジ設定を行ってから<INTxEN>を設定してください。

8.6.3.4 CG 割り込みモードコントロールレジスタ D

CGIMCGD		7	6	5	4	3	2	1	0
	bit Symbol		EMCGC2	EMCGC1	EMCGC0	EMSTC1	EMSTC0		INTCEN
	Read/Write	R	R/W			R		R	R/W
	リセット後	0	0	1	0	0	0	0	0
機能	リードすると"0"が読めます	INTCECTX スタンバイ解除要求のアクティブ状態を設定 以下の設定で使用してください 011: 立ち上がりエッジ			INTCECTX スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ		リードすると"0"が読めます	INTCECTX 解除入力 0: ディセーブル 1: イネーブル	
		15	14	13	12	11	10	9	8
bit Symbol									
Read/Write	R	R/W			R			R/W	
リセット後	0	0	1	0	0	0	0	0	
機能	リードすると"0"が読めます	任意の値をライトしてください			リードすると"0"が読めます			"0"を書いてください。	
		23	22	21	20	19	18	17	16
bit Symbol									
Read/Write	R	R/W			R			R/W	
リセット後	0	0	1	0	0	0	0	0	
機能	リードすると"0"が読めます	任意の値をライトしてください			リードすると"0"が読めます			"0"を書いてください。	
		31	30	29	28	27	26	25	24
bit Symbol									
Read/Write	R	R/W			R			R/W	
リセット後	0	0	1	0	0	0	0	0	
機能	リードすると"0"が読めます	任意の値をライトしてください			リードすると"0"が読めます			"0"を書いてください。	

(注1) EMSTxx を参照することにより、スタンバイ解除に使用されたアクティブ状態を確認することが出来ます。

(注2) エッジ設定と同時に<INTxEN>を設定しないでください。エッジ設定を行ってから<INTxEN>を設定してください。

8.6.3.6 NMI フラグレジスタ

CGNMIFLG

	7	6	5	4	3	2	1	0
bit Symbol							NMIFLG1	NMIFLG0
Read/Write	R							
リセット後	0	0	0	0	0	0	0	0
機能	リードすると”0”が読めます						NMI 起動要因 フラグ	NMI 起動要因 フラグ
							0:要因なし 1:NMI 端子 による 発生	0:要因なし 1:WDT による NMI 発生
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0	0	0	0	0	0	0	0
機能	リードすると”0”が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0	0	0	0	0	0	0	0
機能	リードすると”0”が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0	0	0	0	0	0	0	0
機能	リードすると”0”が読めます							

(注) <NMIFLG1:0>は読み出すと”0”にクリアされます。

8.6.3.7 リセットフラグレジスタ

CGRSTFLG	7	6	5	4	3	2	1	0
bit Symbol				SYSRSTF		WDRSTF	PINRSTF	PONRSTF
Read/Write	R			R/W	R/W	R/W	R/W	R/W
端子リセット後	0	0	0	0	0	0	1	1/0
機能	リードすると"0"が読めます			デバッグ リセット フラグ 0: 0 ライト 1: SYSRSTRQ によるリセ ットによる リセットフ ラグ (注2)	"0"を書 いてくださ い	WDT リセット フラグ 0: 0 ライト 1: WDT リセット によるリ セットフ ラグ	RESET 端子 フラグ 0: 0 ライト 1: RESET 端子によ るリセッ トフラグ	パワーオン フラグ 0: 0 ライト 1: Power On Resetによ るリセッ トフラグ
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
端子リセット後	0	0	0	0	0	0	0	0
機能	リードすると"0"が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
端子リセット後	0	0	0	0	0	0	0	0
機能	リードすると"0"が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
端子リセット後	0	0	0	0	0	0	0	0
機能	リードすると"0"が読めます							

(注1) 本製品はパワーオンリセット回路を内蔵しており、本レジスタはパワーオンリセットでのみ初期化されます。したがって、電源投入時の初期リセット後は<PONRSTF>がセットされますが、2度目以降のリセットでは<PONRSTF>はセットされません。また、本レジスタは自動的にクリアされませんので、“0”を書いてクリアしてください。

(注2) CPU の NVIC 内にある、アプリケーション割り込みおよびリセット制御レジスタの SYSRESETREQ ビットのセットにより発生したリセットであることを示します。

9 入出力ポート

9.1 ポートレジスタ概略説明

Px : ポートレジスタ

ポートのデータ読み込み、データ書込みを行います。

PxCR : コントロールレジスタ

入出力を制御します。

入力設定にした場合でも、PxIE レジスタにより入力をイネーブルにする必要が有ります。

PxFRn : ファンクションレジスタ

機能設定を行ないます。

“1” をセットすることにより割り当てられている機能を使用できるようになります。

PxOD : オープンドレイン制御レジスタ

プログラマブルオープンドレイン設定可能なレジスタに対して、入力を切り替えます。

PxPUP : プルアップ制御レジスタ

プログラマブルプルアップを制御します。

PxPDN : プルダウン制御レジスタ

プログラマブルプルダウンを制御します。

PxIE : 入力制御イネーブルレジスタ

入力の制御を行ないます。貫通電流対策のため、初期値は“0”で入力禁止になっています。

9.2 ポート機能詳細

9.2.1 STOPモード中のポート状態

STOP モード中の入力と出力の状態を、クロック/モード制御部のスタンバイコントロールレジスタ CGSTBYCR<DRVE>ビットで制御することができます。

PxIE、PxCR がイネーブルで、<DRVE>ビットを“1”に設定した場合、STOP モード中も入力、出力がイネーブルとなります。<DRVE>ビットを“0”に設定した場合、一部のポートを除き、PxIE、PxCR がイネーブルであっても STOP モード中は入力、出力がディセーブルになります。

以下に、設定による動作の違いをまとめます。

ポート	入/出力	<DRVE>=0	<DRVE>=1
PA0, PB0 (デバッグ機能設定 (PxFR<n>=1) かつ 出力イネーブル設定 (PxCR<n>=1) の場合)	入力	×	設定 (PxIE<n>) に従う
	出力	データ有効時はイネーブル データ無効時はディセーブル	データ有効時はイネーブル データ無効時はディセーブル
PF7, PG3, PJ0-3, PJ6, PJ7 (割り込み機能設定 (PxFR<n>=1) かつ 入力イネーブル設定 (PxIE<n>=1) の場合)	入力	○	○
	出力	×	設定 (PxCR<n>) に従う
上記以外のポート	入力	×	設定 (PxIE<n>) に従う
	出力	×	設定 (PxCR<n>) に従う

○： 入力または出力がイネーブルであることを示します。

×： 入力または出力がディセーブルであることを示します。

9.2.2 STOP/SLEEPモード遷移の際の注意

PA1 が TCK/SWCLK のデバッグ機能設定の場合十分な低消費電力効果が得られません。デバッグ機能を使用しないときは、PA1 をポート設定にしてください。

9.2.3 ポート A (PA0~PA7)

ポート A はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用入出力ポート機能以外にデバッグ通信機能、デバッグトレース出力機能があります。

リセットにより PA0 と PA1 はデバッグ通信機能となります。PA0 は TMS または SWDIO 機能で入力・出力・プルアップイネーブル、PA1 は TCK または SWCLK 機能で入力・プルダウンイネーブル、となります。

その他のビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力・プルアップはディセーブル状態になります。

- | |
|---|
| <p>(注 1) PA0、PA1 は初期状態が機能端子となり、入出力・プルアップ・プルダウンが有効な状態です。</p> <p>(注 2) PA0 が TMS/SWDIO 機能設定の場合、スタンバイコントロールレジスタ (CGSTBYCR) の<DRVE> ビットの状態によらず、STOP モード中も出力が有効な状態のまま保持されます。</p> <p>(注 3) PA1 がデバッグ機能設定の場合十分な低消費電力効果が得られません。デバッグ機能を使用しないときは、PA1 をポート設定にしてください。</p> |
|---|

ポート A 回路タイプ

	7	6	5	4	3	2	1	0
Type	T1	T9	T9	T9	T9	T9	T6	T12

ポート A レジスタ

PA (0x4000_0000)		7	6	5	4	3	2	1	0
	Bit Symbol	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
	Read/Write	R/W							
	リセット後	"0"							

ポート A コントロールレジスタ

PACR (0x4000_0004)		7	6	5	4	3	2	1	0
	Bit Symbol	PA7C	PA6C	PA5C	PA4C	PA3C	PA2C	PA1C	PA0C
	Read/Write	R/W							
	リセット後	0	0	0	0	0	0	0	1
機能	0:入力 1:出力								

ポート A ファンクションレジスタ 1

PAFR1 (0x4000_0008)		7	6	5	4	3	2	1	0
	Bit Symbol	—	PA6F1	PA5F1	PA4F1	PA3F1	PA2F1	PA1F1	PA0F1
	Read/Write	R	R/W						
	リセット後	0	0	0	0	0	0	1	1
機能	リードすると"0"が読めます	0:PORT 1: TRACE DATA3	0:PORT 1: TRACE DATA2	0:PORT 1: TRACE DATA1	0:PORT 1TRACE DATA0	0:PORT 1: TRACE CLK	0:PORT 1: TCK/ SWCLK	0:PORT 1: TMS/ SWDIO	

ポート A プルアップ制御レジスタ

PAPUP (0x4000_002C)		7	6	5	4	3	2	1	0	
	Bit Symbol	PA7UP	PA6UP	PA5UP	PA4UP	PA3UP	PA2UP	—	PA0UP	
	Read/Write	R/W							R	R/W
	リセット後	0	0	0	0	0	0	0	1	
機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	リードすると"0"が読めます	プルアップ 0:オフ 1:Pull-Up	

ポート A プルダウン制御レジスタ

PAPDN (0x4000_0030)		7	6	5	4	3	2	1	0	
	Bit Symbol	—	—	—	—	—	—	PA1DN	—	
	Read/Write	R							R/W	R
	リセット後	0	0	0	0	0	0	1	0	
機能	リードすると"0"が読めます							プルダウン 0:オフ 1:Pull-Down	リードすると"0"が読めます	

ポート A 入力イネーブル制御レジスタ

PAIE (0x4000_0038)		7	6	5	4	3	2	1	0
	Bit Symbol	PA7IE	PA6IE	PA5IE	PA4IE	PA3IE	PA2IE	PA1IE	PA0IE
	Read/Write	R/W							
	リセット後	0	0	0	0	0	0	1	1
機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

9.2.4 ポートB (PB0~PB7)

ポートBはビット単位で入出力の指定ができる8ビットの汎用入出力ポートです。汎用入出力ポート機能以外にデバッグ通信機能があります。

リセットによりPB0、PB1、PB2はデバッグ通信機能になります。PB0はTD0またはSWV機能で出力イネーブル、PB1はTDI機能で入力・プルアップイネーブル、PB2はTRST機能で入力・プルアップイネーブルとなります。

その他のビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力・プルアップはディセーブル状態になります。

(注1) PB0、PB1、PB2は初期状態が機能端子となり、入出力・プルアップが有効な状態です。

(注2) PB0がTD0/SWV機能設定の場合、スタンバイコントロールレジスタ(CGSTBYCR)の<DRVE>ビットの状態によらず、STOPモード中も出力が有効な状態のまま保持されます。

ポートB 回路タイプ

	7	6	5	4	3	2	1	0
Type	T1	T1	T1	T1	T1	T2	T2	T11

ポートB レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
Read/Write	RW							
リセット後	"0"							

PB
(0x4000_0040)

ポートB コントロールレジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PB7C	PB6C	PB5C	PB4C	PB3C	PB2C	PB1C	PB0C
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	1
機能	0:入力 1:出力							

PBCR
(0x4000_0044)

ポートB ファンクションレジスタ 1

	7	6	5	4	3	2	1	0
Bit Symbol	—	—	—	—	—	PB2F1	PB1F1	PB0F1
Read/Write	R					RW		
リセット後	0					1	1	1
機能	リードすると"0"が読めます					0:PORT 1: TRST	0:PORT 1: TDI	0:PORT 1: TDO/ SWV

PBFR1
(0x4000_0048)

ポートB プルアップ制御レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PB7UP	PB6UP	PB5UP	PB4UP	PB3UP	PB2UP	PB1UP	PB0UP
Read/Write	RW							
リセット後	0	0	0	0	0	1	1	0
機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up

PBPUP
(0x4000_006C)

ポートB 入力イネーブル制御レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PB7IE	PB6IE	PB5IE	PB4IE	PB3IE	PB2IE	PB1IE	PB0IE
Read/Write	RW							
リセット後	0	0	0	0	0	1	1	0
機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

PBIE
(0x4000_0078)

9.2.5 ポートC (PC0~PC3)

ポートCは4ビットの入力専用ポートです。汎用入力ポート機能以外に A/D コンバータのアナログ入力端子機能があります。

リセットによりすべてのビットのファンクションレジスタ設定は汎用入力ポートとなり、入力・プルアップはディセーブル状態になります。

入力ポートとして使用する場合には、入力イネーブル制御レジスタを設定してください。A/D コンバータのアナログ入力として使用する場合にはレジスタ設定は不要です。

(注) ポートC/ポートDを、アナログ入力とそれ以外の機能で混在して使用した場合、A/D 変換精度が悪化する場合があります。必ずご使用になるシステム上で問題のないことを確認してください。

ポートC 回路タイプ

	7	6	5	4	3	2	1	0
Type	—	—	—	—	T17	T17	T17	T17

ポートC レジスタ

	7	6	5	4	3	2	1	0
PC (0x4000_0080)	—	—	—	—	PC3	PC2	PC1	PC0
Read/Write	R				R			
リセット後	リードすると"0"が読めます				"1"			

ポートC プルアップ制御レジスタ

	7	6	5	4	3	2	1	0
PCPUP (0x4000_00AC)	—	—	—	—	PC3UP	PC2UP	PC1UP	PC0UP
Read/Write	R				R/W			
リセット後	0				0	0	0	0
機能	リードすると"0"が読めます				プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up

ポートC 入力イネーブル制御レジスタ

	7	6	5	4	3	2	1	0
PCIE (0x4000_00B8)	—	—	—	—	PC3IE	PC2IE	PC1IE	PC0IE
Read/Write	R				R/W			
リセット後	0				0	0	0	0
機能	リードすると"0"が読めます				入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

9.2.6 ポートD (PD0~PD7)

ポートDは8ビットの入力専用ポートです。汎用入力ポート機能以外に A/D コンバータのアナログ入力端子機能、16 ビットタイマ入力機能があります。

リセットによりすべてのビットのファンクションレジスタ設定は汎用入力ポートとなり、入力・プルアップはディセーブル状態になります。

入力ポートとして使用する場合には、入力イネーブル制御レジスタを設定してください。16 ビットタイマ入力として使用する場合には、ファンクションレジスタ 1 と入力イネーブル制御レジスタを設定してください。A/D コンバータのアナログ入力として使用する場合にはレジスタ設定は不要です。

(注) ポートC/ポートDを、アナログ入力とそれ以外の機能で混在して使用した場合、A/D 変換精度が悪化する場合があります。必ずご使用になるシステム上で問題のないことを確認してください。

ポートD 回路タイプ

	7	6	5	4	3	2	1	0
Type	T17	T17	T17	T17	T18	T18	T18	T18

ポートD レジスタ

PD (0x4000_00C0)		7	6	5	4	3	2	1	0
	Bit Symbol	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	Read/Write	R							
	リセット後	"1"							

ポートD ファンクションレジスタ 1

PDFR1 (0x4000_00C8)		7	6	5	4	3	2	1	0
	Bit Symbol	—	—	—	—	PD3F1	PD2F1	PD1F1	PD0F1
	Read/Write	R				R/W			
	リセット後	0				0	0	0	0
	機能	リードすると"0"が読めます				0:PORT 1: TB6IN1	0:PORT 1: TB6IN0	0:PORT 1: TB5IN1	0:PORT 1: TB5IN0

ポートD プルアップ制御レジスタ

PDPUP (0x4000_00EC)		7	6	5	4	3	2	1	0
	Bit Symbol	PD7UP	PD6UP	PD5UP	PD4UP	PD3UP	PD2UP	PD1UP	PD0UP
	Read/Write	R/W							
	リセット後	0	0	0	0	0	0	0	0
	機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up

ポートD 入力イネーブル制御レジスタ

PDIE (0x4000_00F8)		7	6	5	4	3	2	1	0
	Bit Symbol	PD7IE	PD6IE	PD5IE	PD4IE	PD3IE	PD2IE	PD1IE	PD0IE
	Read/Write	R/W							
	リセット後	0	0	0	0	0	0	0	0
	機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

9.2.7 ポートE (PE0~PE6)

ポートEはビット単位で入出力の指定ができる7ビットの汎用入出力ポートです。汎用ポート機能以外にシリアルインタフェース機能、リモコン判定入力機能があります。

リセットによりすべてのビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力・プルアップはディセーブル状態になります。

ポートE 回路タイプ

	7	6	5	4	3	2	1	0
Type	—	T16	T4	T10	T4	T16	T4	T10

ポートE レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	—	PE6	PE5	PE4	PE3	PE2	PE1	PE0
Read/Write	R	R/W						
リセット後	"0"	"0"						

PE
(0x4000_0100)

ポートE コントロールレジスタ

	7	6	5	4	3	2	1	0		
PECR (0x4000_0104)	Bit Symbol	—	PE6C	PE5C	PE4C	PE3C	PE2C	PE1C	PE0C	
	Read/Write	R	R/W							
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます	0:入力 1:出力							

ポートE ファンクションレジスタ 1

	7	6	5	4	3	2	1	0		
PEFR1 (0x4000_0108)	Bit Symbol	—	PE6F1	PE5F1	PE4F1	PE3F1	PE2F1	PE1F1	PE0F1	
	Read/Write	R	R/W							
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます	0:PORT 1:SCLK1	0:PORT 1:RXD1	0:PORT 1:TXD1	0:PORT 1:RXIN0	0:PORT 1:SCLK0	0:PORT 1:RXD0	0:PORT 1:TXD0	

ポートE ファンクションレジスタ 2

	7	6	5	4	3	2	1	0	
PEFR2 (0x4000_010C)	Bit Symbol	—	PE6F2	—	—	—	PE2F2	—	—
	Read/Write	R	R/W	R			R/W	R	
	リセット後	0	0	0			0	0	
	機能	リードすると"0"が読めます	0:PORT 1:CTS1	リードすると"0"が読めます			0:PORT 1:CTS0	リードすると"0"が読めます	

ポートE オープンドレイン制御レジスタ

	7	6	5	4	3	2	1	0	
PEOD (0x4000_0128)	Bit Symbol	—	PE6OD	PE5OD	PE4OD	PE3OD	PE2OD	PE1OD	PE0OD
	Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	リセット後	0	0	0	0	0	0	0	0
	機能	リードすると"0"が読めます	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン

ポートE プルアップ制御レジスタ

	7	6	5	4	3	2	1	0		
PEPUP (0x4000_012C)	Bit Symbol	—	PE6UP	PE5UP	PE4UP	PE3UP	PE2UP	PE1UP	PE0UP	
	Read/Write	R	R/W							
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	

ポートE 入力インエーブル制御レジスタ

	7	6	5	4	3	2	1	0		
PEIE (0x4000_0138)	Bit Symbol	—	PE6IE	PE5IE	PE4IE	PE3IE	PE2IE	PE1IE	PE0IE	
	Read/Write	R	R/W							
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると"0"が読めます	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	

9.2.8 ポート F (PF0~PF7)

ポート F はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用ポート機能以外にシリアルインタフェース機能、リモコン判定入力機能、シリアルバスインタフェース機能、外部割り込み入力機能があります。

割り込み入力を STOP モード解除に使用する場合、PxFR レジスタで機能設定にし、PxIE レジスタで入力設定をしてください。この設定では、クロック/モード制御部のスタンバイコントロールレジスタ CGSTBYCR<DRVE>ビットで STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

(注) STOP 以外のモードでは、PxIE で入力イネーブル設定であれば PxFR の設定によらず割り込みの入力が可能になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。

リセットによりすべてのビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力・プルアップはディセーブル状態になります。

ポート F 回路タイプ

	7	6	5	4	3	2	1	0
Type	T8	T13	T13	T13	T4	T16	T4	T10

ポート F レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
Read/Write	R/W							
リセット後	"0"							

PF
(0x4000_0140)

ポートF コントロールレジスタ

	7	6	5	4	3	2	1	0		
PFCR (0x4000_0144)	Bit Symbol	PF7C	PF6C	PF5C	PF4C	PF3C	PF2C	PF1C	PF0C	
	Read/Write	R/W								
	リセット後	0	0	0	0	0	0	0	0	
	機能	0:入力 1:出力								

ポートF ファンクションレジスタ 1

	7	6	5	4	3	2	1	0		
PFFR1 (0x4000_0148)	Bit Symbol	PF7F1	PF6F1	PF5F1	PF4F1	PF3F1	PF2F1	PF1F1	PF0F1	
	Read/Write	R/W								
	リセット後	0	0	0	0	0	0	0	0	
	機能	0:PORT 1:INT5	0:PORT 1:SCK1	0:PORT 1:SI1/ SCL1	0:PORT 1:SO1/ SDA1	0:PORT 1:RXIN1	0:PORT 1:SCLK2	0:PORT 1:RXD2	0:PORT 1:TXD2	

ポートF ファンクションレジスタ 2

	7	6	5	4	3	2	1	0	
PFFR2 (0x4000_014C)	Bit Symbol	—	—	—	—	—	PF2F2	—	—
	Read/Write	R					R/W	R	
	リセット後	0					0	0	
	機能	リードすると"0"が読めます					0:PORT 1:CTS2	リードすると"0"が読めます	

ポートF オープンドレイン制御レジスタ

	7	6	5	4	3	2	1	0	
PFOD (0x4000_0168)	Bit Symbol	PF7OD	PF6OD	PF5OD	PF4OD	PF3OD	PF2OD	PF1OD	PF0OD
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	リセット後	0	0	0	0	0	0	0	0
	機能	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン

ポートF プルアップ制御レジスタ

	7	6	5	4	3	2	1	0		
PFPUP (0x4000_016C)	Bit Symbol	PF7UP	PF6UP	PF5UP	PF4UP	PF3UP	PF2UP	PF1UP	PF0UP	
	Read/Write	R/W								
	リセット後	0	0	0	0	0	0	0	0	
	機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	

ポートF 入力イネーブル制御レジスタ

	7	6	5	4	3	2	1	0		
PFIE (0x4000_0178)	Bit Symbol	PF7IE	PF6IE	PF5IE	PF4IE	PF3IE	PF2IE	PF1IE	PF0IE	
	Read/Write	R/W								
	リセット後	0	0	0	0	0	0	0	0	
	機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	

9.2.9 ポートG (PG0~PG7)

ポートGはビット単位で入出力の指定ができる8ビットの汎用入出力ポートです。汎用ポート機能以外にシリアルバスインタフェース機能、外部割込み入力機能、16ビットタイマ出力機能があります。

割り込み入力をSTOPモード解除に使用する場合、PxFRレジスタで機能設定にし、PxIEレジスタで入力設定をしてください。この設定では、クロック/モード制御部のスタンバイコントロールレジスタCGSTBYCR<DRVE>ビットでSTOPモード中端子をドライブしない設定を行っていても割り込み入力可能です。

(注) STOP以外のモードでは、PxIEで入力イネーブル設定であればPxFRの設定によらず割り込みの入力が可能になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。

リセットによりすべてのビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力・プルアップはディセーブル状態になります。

ポートG 回路タイプ

	7	6	5	4	3	2	1	0
Type	T10	T13	T13	T13	T8	T13	T13	T13

ポートG レジスタ

PG
(0x4000_0180)

	7	6	5	4	3	2	1	0
Bit Symbol	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
Read/Write	R/W							
リセット後	"0"							

ポートG コントロールレジスタ

PGCR
(0x4000_0184)

	7	6	5	4	3	2	1	0
Bit Symbol	PG7C	PG6C	PG5C	PG4C	PG3C	PG2C	PG1C	PG0C
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	0:入力 1:出力							

ポートG ファンクションレジスタ 1

PGFR1
(0x4000_0188)

	7	6	5	4	3	2	1	0
Bit Symbol	PG7F1	PG6F1	PG5F1	PG4F1	PG3F1	PG2F1	PG1F1	PG0F1
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	0:PORT 1: TB8OUT	0:PORT 1: SCK2	0:PORT 1: SI2/ SCL2	0:PORT 1: SO2/ SDA2	0:PORT 1: INT4	0:PORT 1: SCK0	0:PORT 1: SI0/ SCL0	0:PORT 1: SO0/ SDA0

ポートG オープンドレイン制御レジスタ

PGOD
(0x4000_01A8)

	7	6	5	4	3	2	1	0
Bit Symbol	PG7OD	PG6OD	PG5OD	PG4OD	PG3OD	PG2OD	PG1OD	PG0OD
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット後	0	0	0	0	0	0	0	0
機能	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン	0:CMOS 1:オープン ドレイン

ポートG プルアップ制御レジスタ

PGPUP
(0x4000_01AC)

	7	6	5	4	3	2	1	0
Bit Symbol	PG7UP	PG6UP	PG5UP	PG4UP	PG3UP	PG2UP	PG1UP	PG0UP
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up

ポートG 入力イネーブル制御レジスタ

PGIE
(0x4000_01B8)

	7	6	5	4	3	2	1	0
Bit Symbol	PG7IE	PG6IE	PG5IE	PG4IE	PG3IE	PG2IE	PG1IE	PG0IE
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

9.2.10 ポートH (PH0~PH7)

ポートHはビット単位で入出力の指定ができる8ビットの汎用入出力ポートです。汎用ポート機能以外に16ビットタイマ入力機能、動作モード設定機能があります。

リセットが“0”の期間、PH0は入力とプルアップがイネーブルとなっており、リセット信号の立ち上がりでPH0が“1”の場合、シングルモードとなって内蔵FlashメモリからBOOTし、PH0が“0”の場合、シングルBOOTモードとなって内蔵BOOTプログラムからBOOTします。シングルBOOTモードの説明は、18章のFlash動作説明を参照してください。

リセット解除後、すべてのビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力はディセーブル状態になります。プルアップは、PH0はイネーブル、PH1~PH7はディセーブル状態になります。

ポートH 回路タイプ

	7	6	5	4	3	2	1	0
Type	T3	T3	T3	T3	T3	T3	T3	T5

ポートH レジスタ

	7	6	5	4	3	2	1	0	
PH (0x4000_01C0)	Bit Symbol	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
	Read/Write	RW							
	リセット後	"0"							

ポートH コントロールレジスタ

	7	6	5	4	3	2	1	0	
PHCR (0x4000_01C4)	Bit Symbol	PH7C	PH6C	PH5C	PH4C	PH3C	PH2C	PH1C	PH0C
	Read/Write	RW							
	リセット後	0	0	0	0	0	0	0	0
	機能	0:入力 1:出力							

ポートH ファンクションレジスタ 1

	7	6	5	4	3	2	1	0	
PHFR1 (0x4000_01C8)	Bit Symbol	PH7F1	PH6F1	PH5F1	PH4F1	PH3F1	PH2F1	PH1F1	PH0F1
	Read/Write	RW							
	リセット後	0	0	0	0	0	0	0	0
	機能	0:PORT 1:TB3IN1	0:PORT 1:TB3IN0	0:PORT 1:TB2IN1	0:PORT 1:TB2IN0	0:PORT 1:TB1IN1	0:PORT 1:TB1IN0	0:PORT 1:TB0IN1	0:PORT 1:TB0IN0

ポートH プルアップ制御レジスタ

	7	6	5	4	3	2	1	0	
PHPUP (0x4000_01EC)	Bit Symbol	PH7UP	PH6UP	PH5UP	PH4UP	PH3UP	PH2UP	PH1UP	PH0UP
	Read/Write	RW							
	リセット後	0	0	0	0	0	0	0	1
	機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up

ポートH 入力イネーブル制御レジスタ

	7	6	5	4	3	2	1	0	
PHIE (0x4000_01F8)	Bit Symbol	PH7IE	PH6IE	PH5IE	PH4IE	PH3IE	PH2IE	PH1IE	PH0IE
	Read/Write	RW							
	リセット後	0	0	0	0	0	0	0	0
	機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

9.2.11 ポート I (PI0~PI7)

ポート I はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用ポート機能以外に 16 ビットタイマ入力機能、16 ビットタイマ出力機能があります。

リセットによってすべてのビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力プルアップはディセーブル状態になります。

ポート I 回路タイプ

	7	6	5	4	3	2	1	0
Type	T3	T3	T9	T9	T9	T9	T9	T9

ポート I レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PI7	PI6	PI5	PI4	PI3	PI2	PI1	PI0
Read/Write	RW							
リセット後	"0"							

PI
(0x4000_0200)

ポート I コントロールレジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PI7C	PI6C	PI5C	PI4C	PI3C	PI2C	PI1C	PI0C
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	0:入力 1:出力							

PICR
(0x4000_0204)

ポート I ファンクションレジスタ 1

	7	6	5	4	3	2	1	0
Bit Symbol	PI7F1	PI6F1	PI5F1	PI4F1	PI3F1	PI2F1	PI1F1	PI0F1
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	0:PORT 1:TB4IN1	0:PORT 1:TB4IN0	0:PORT 1:TB5OUT	0:PORT 1:TB4OUT	0:PORT 1:TB3OUT	0:PORT 1:TB2OUT	0:PORT 1:TB1OUT	0:PORT 1:TB0OUT

PIFR1
(0x4000_0208)

ポート I プルアップ制御レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PI7UP	PI6UP	PI5UP	PI4UP	PI3UP	PI2UP	PI1UP	PI0UP
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up

PIPUP
(0x4000_022C)

ポート I 入力イネーブル制御レジスタ

	7	6	5	4	3	2	1	0
Bit Symbol	PI7IE	PI6IE	PI5IE	PI4IE	PI3IE	PI2IE	PI1IE	PI0IE
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

PIIE
(0x4000_0238)

9.2.12 ポート J (PJ0~PJ7)

ポート J はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用ポート機能以外に 16 ビットタイマ出力機能、外部割込み入力機能があります。

割り込み入力を STOP モード解除に使用する場合、PxFR レジスタで機能設定にし、PxIE レジスタで入力設定をしてください。この設定では、クロック/モード制御部のスタンバイコントロールレジスタ CGSTBYCR<DRVE>ビットで STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

(注) STOP 以外のモードでは、PxIE で入力イネーブル設定であれば PxFR の設定によらず割り込みの入力が可能になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。

リセットによってすべてのビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力・プルアップはディセーブル状態になります。

ポート J 回路タイプ

	7	6	5	4	3	2	1	0
Type	T7	T7	T9	T9	T7	T7	T7	T7

ポート J レジスタ

PJ
(0x4000_0240)

	7	6	5	4	3	2	1	0
Bit Symbol	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0
Read/Write	RW							
リセット後	"0"							

ポート J コントロールレジスタ

PJCR
(0x4000_0244)

	7	6	5	4	3	2	1	0
Bit Symbol	PJ7C	PJ6C	PJ5C	PJ4C	PJ3C	PJ2C	PJ1C	PJ0C
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	0:入力 1:出力							

ポート J ファンクションレジスタ 1

PJFR1
(0x4000_0248)

	7	6	5	4	3	2	1	0
Bit Symbol	PJ7F1	PJ6F1	PJ5F1	PJ4F1	PJ3F1	PJ2F1	PJ1F1	PJ0F1
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	0:PORT 1: INT7	0:PORT 1: INT6	0:PORT 1: TB7OUT	0:PORT 1: TB6OUT	0:PORT 1: INT3	0:PORT 1: INT2	0:PORT 1: INT1	0:PORT 1: INT0

ポート J プルアップ制御レジスタ

PJPUP
(0x4000_026C)

	7	6	5	4	3	2	1	0
Bit Symbol	PJ7UP	PJ6UP	PJ5UP	PJ4UP	PJ3UP	PJ2UP	PJ1UP	PJ0UP
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up

ポート J 入力イネーブル制御レジスタ

PJIE
(0x4000_0278)

	7	6	5	4	3	2	1	0
Bit Symbol	PJ7IE	PJ6IE	PJ5IE	PJ4IE	PJ3IE	PJ2IE	PJ1IE	PJ0IE
Read/Write	RW							
リセット後	0	0	0	0	0	0	0	0
機能	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

9.2.13 ポートK (PK0~PK2)

ポートKはビット単位で入出力の指定ができる3ビットの汎用入出力ポートです。汎用ポート機能以外に16ビットタイマ出力機能、CEC機能入力、クロック出力機能、アラーム出力機能があります。

リセットによってすべてのビットのファンクションレジスタ設定は汎用ポートとなり、入力・出力・プルアップはディセーブル状態になります。

(注) PK0端子はNチャンネルオープンドレインタイプの端子です。

ポートK 回路タイプ

	7	6	5	4	3	2	1	0
Type	—	—	—	—	—	T9	T15	T14

ポートK レジスタ

PK
(0x4000_0280)

	7	6	5	4	3	2	1	0
Bit Symbol	—	—	—	—	—	PK2	PK1	PK0
Read/Write	R					R/W		
リセット後	リードすると"0"が読めます					"0"		

ポートK コントロールレジスタ

PKCR
(0x4000_0284)

	7	6	5	4	3	2	1	0
Bit Symbol	—	—	—	—	—	PK2C	PK1C	PK0C
Read/Write	R					R/W		
リセット後	0					0	0	0
機能	リードすると"0"が読めます					0: 入力 1: 出力		

ポートK ファンクションレジスタ 1

PKFR1
(0x4000_0288)

	7	6	5	4	3	2	1	0
Bit Symbol	—	—	—	—	—	PK2F1	PK1F1	PK0F1
Read/Write	R					R/W		
リセット後	0					0	0	0
機能	リードすると"0"が読めます					0:PORT 1: TB9OUT	0:PORT 1: SCOUT	0:PORT 1: CEC

ポートK ファンクションレジスタ 2

PKFR2
(0x4000_028C)

	7	6	5	4	3	2	1	0
Bit Symbol	—	—	—	—	—	—	PK2F1	—
Read/Write	R						R/W	R
リセット後	0						0	0
機能	リードすると"0"が読めます						0:PORT 1: ALARM	リードすると"0"が読めます

ポートK プルアップ制御レジスタ

PKPUP
(0x4000_02AC)

	7	6	5	4	3	2	1	0
Bit Symbol	—	—	—	—	—	PK2UP	PK1UP	—
Read/Write	R					R/W		R
リセット後	0					0	0	0
機能	リードすると"0"が読めます					プルアップ 0:オフ 1:Pull-Up	プルアップ 0:オフ 1:Pull-Up	リードすると"0"が読めます

ポートK 入力イネーブル制御レジスタ

PKIE
(0x4000_02B8)

	7	6	5	4	3	2	1	0
Bit Symbol	—	—	—	—	—	PK2IE	PK1IE	PK0IE
Read/Write	R					R/W		
リセット後	0					0	0	0
機能	リードすると"0"が読めます					入力 0:禁止 1:許可	入力 0:禁止 1:許可	入力 0:禁止 1:許可

9.3 ポート回路図

9.3.1 ポートタイプ一覧

ポートには、以下のタイプがあります。それぞれの回路図を次ページから示します。

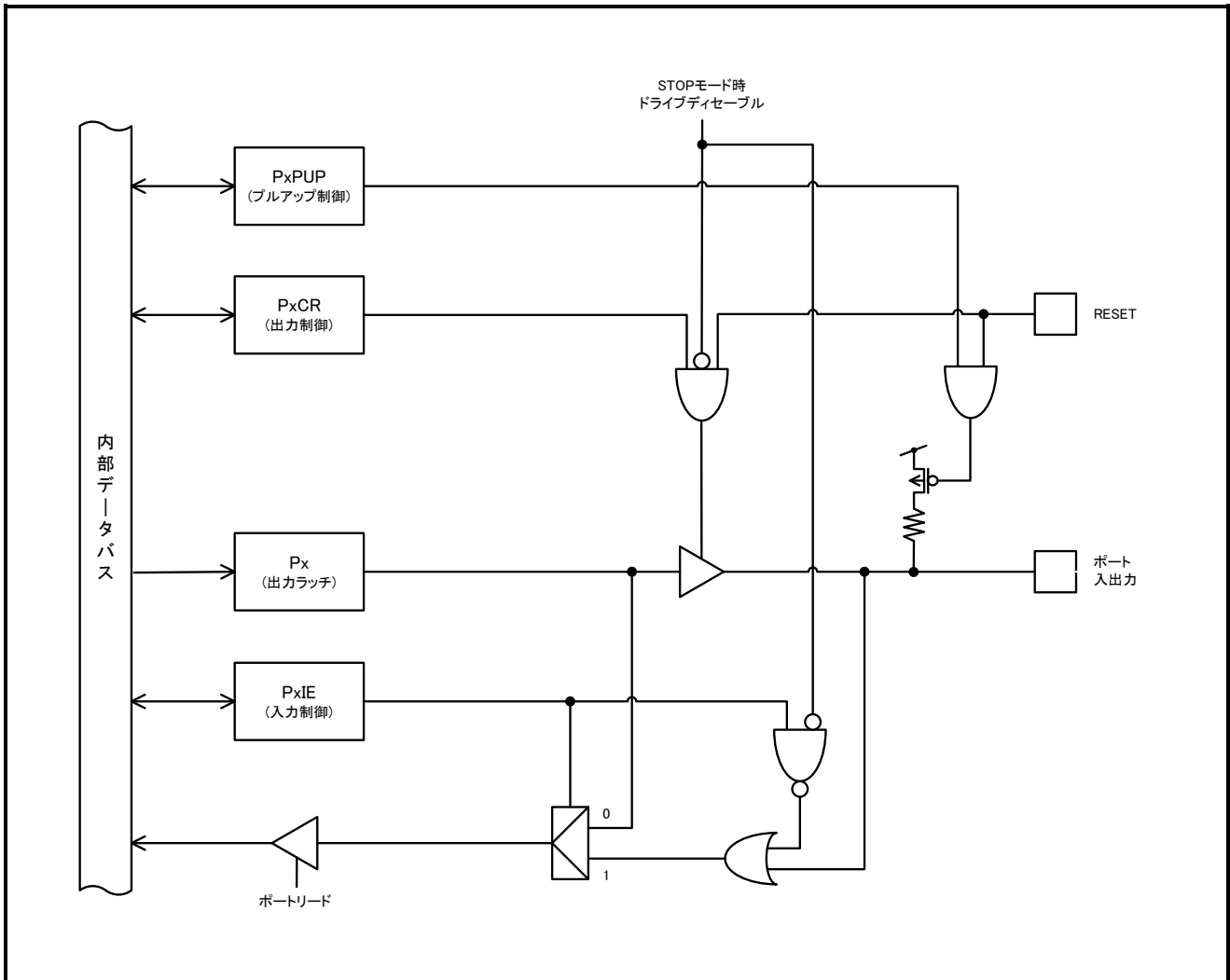
Type	汎用 ポート	機能 1	機能 2	アナログ	Pull up	Pull dn	OD	備考
T1	i/o	-	-	-	R	-	-	
T2	i/o	i	-	-	NoR	-	-	
T3	i/o	i	-	-	R	-	-	
T4	i/o	i	-	-	R	-	○	
T5	i/o	i	-	-	NoR	-	-	リセット中 BOOT 入力イネーブル
T6	i/o	i	-	-	-	NoR	-	
T7	i/o	i (int)	-	-	R	-	-	
T8	i/o	i (int)	-	-	R	-	○	
T9	i/o	o	-	-	R	-	-	
T10	i/o	o	-	-	R	-	○	
T11	i/o	o	-	-	R	-	-	機能出力に イネーブル付き
T12	i/o	i/o	-	-	NoR	-	-	機能出力に イネーブル付き
T13	i/o	i/o	-	-	R	-	○	
T14	i/o	i/o	-	-	-	-	◎	N チャネルオープン ドレイン端子
T15	i/o	o	o	-	R	-	-	
T16	i/o	i/o	i	-	R	-	○	
T17	i	-	-	○	R	-	-	
T18	i	i	-	○	R	-	-	

R : リセット中は強制的にディセーブル。

NoR : リセットでは制御されない。

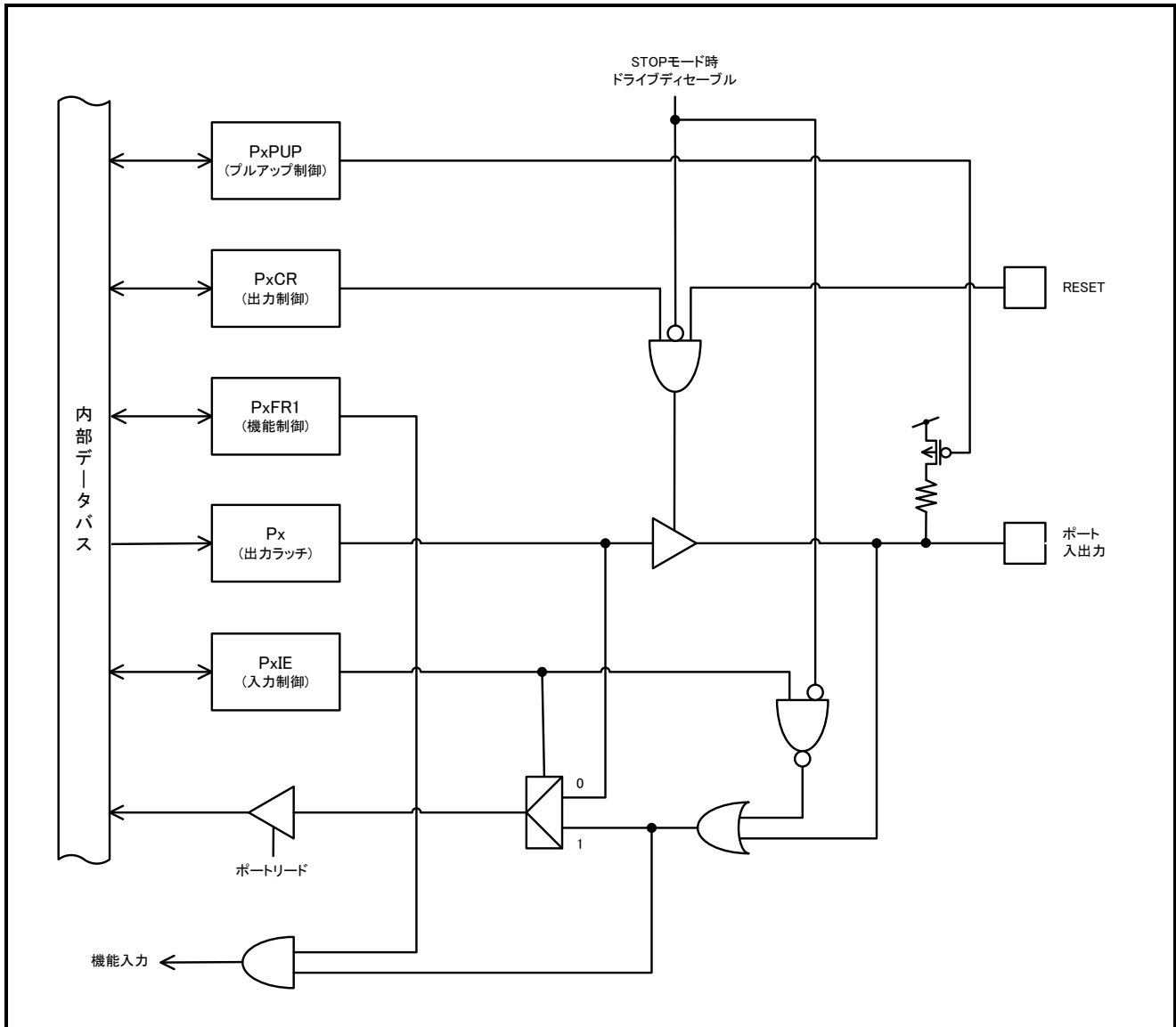
9.3.2 タイプ T1

タイプ T1 はプルアップ付きの汎用入出力ポートです。
 プルアップと出力は、リセット中はディセーブルになります。



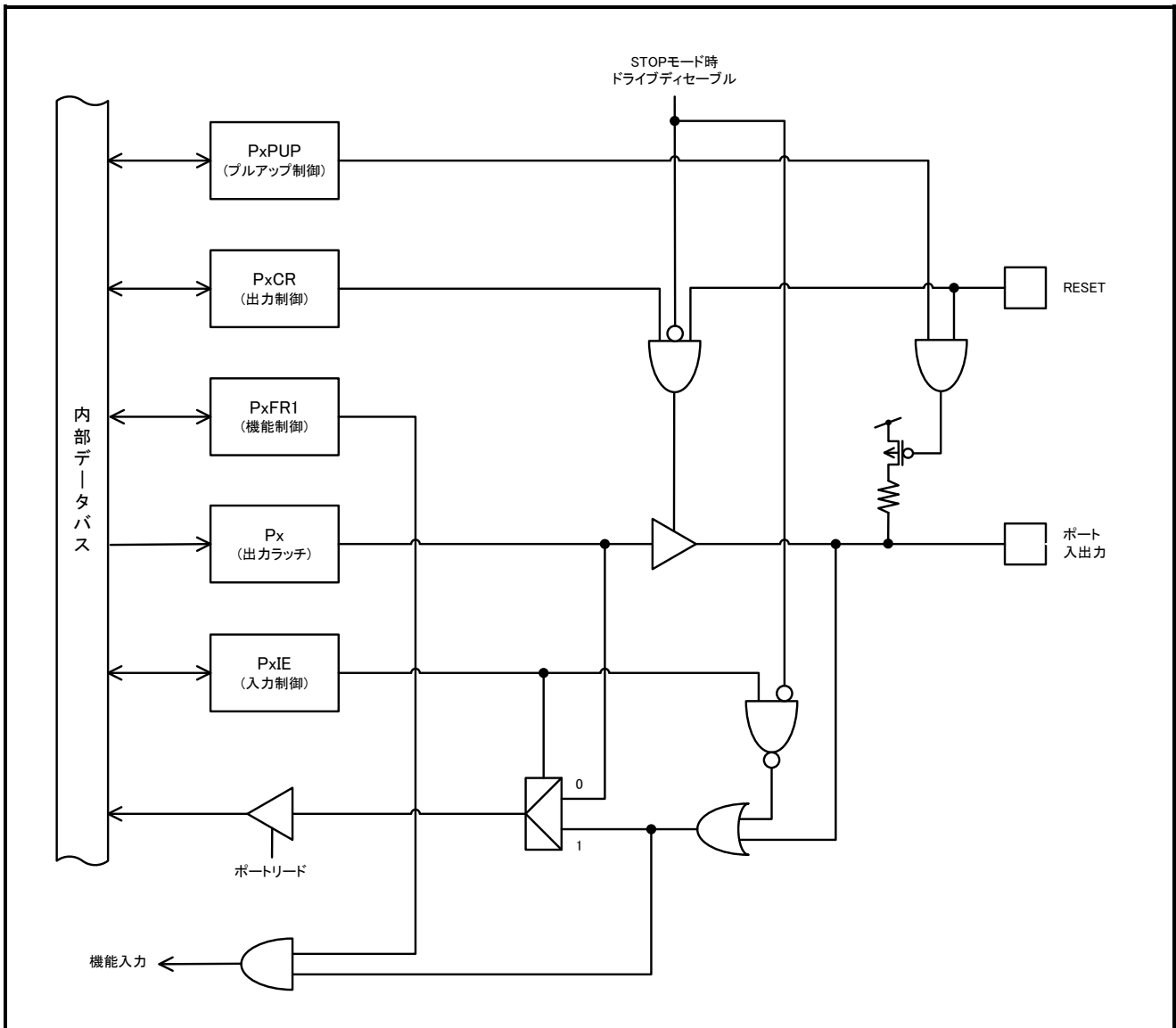
9.3.3 タイプ T2

タイプ T2 はプルアップ付きの汎用入出力ポートで、機能入力と兼用です。
出力は、リセット中はディセーブルになります。



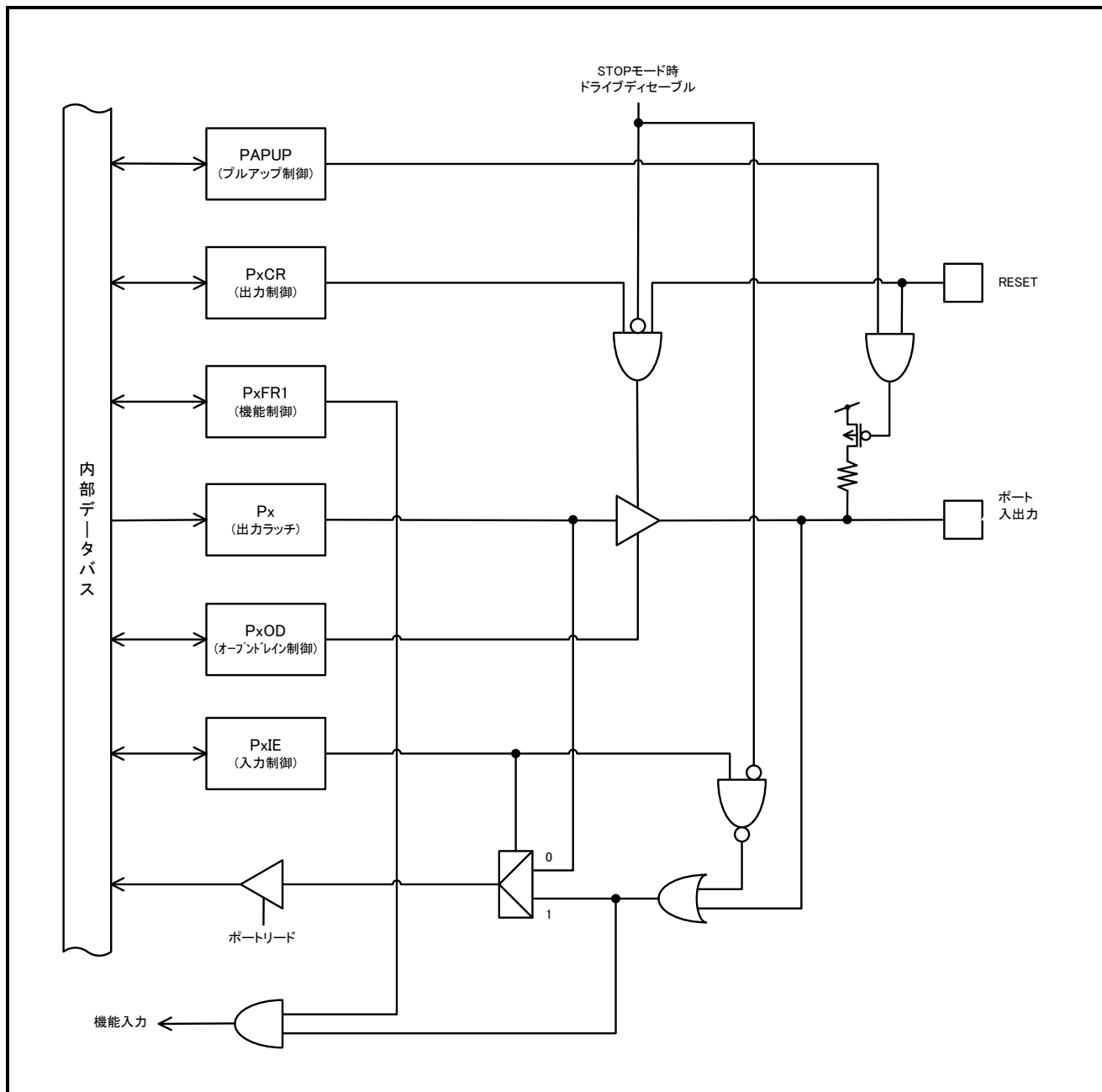
9.3.4 タイプ T3

タイプ T3 はプルアップ付きの汎用入出力ポートで、機能入力と兼用です。
 プルアップと出力は、リセット中はディセーブルになります。



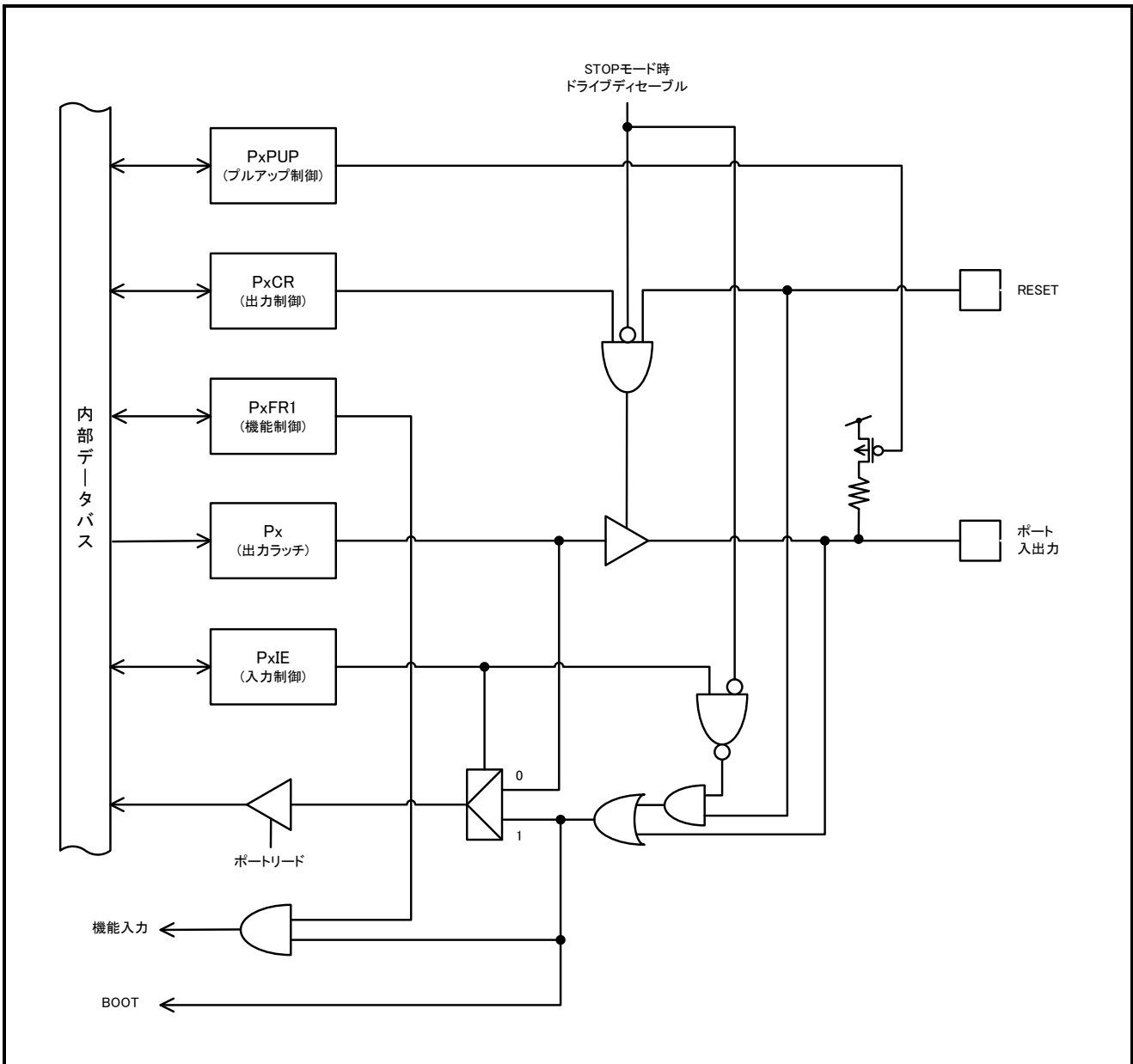
9.3.5 タイプ T 4

タイプ T4 はプルアップおよびオープンドレイン付きの汎用入出力ポートで、機能入力と兼用です。プルアップと出力は、リセット中はディセーブルになります。



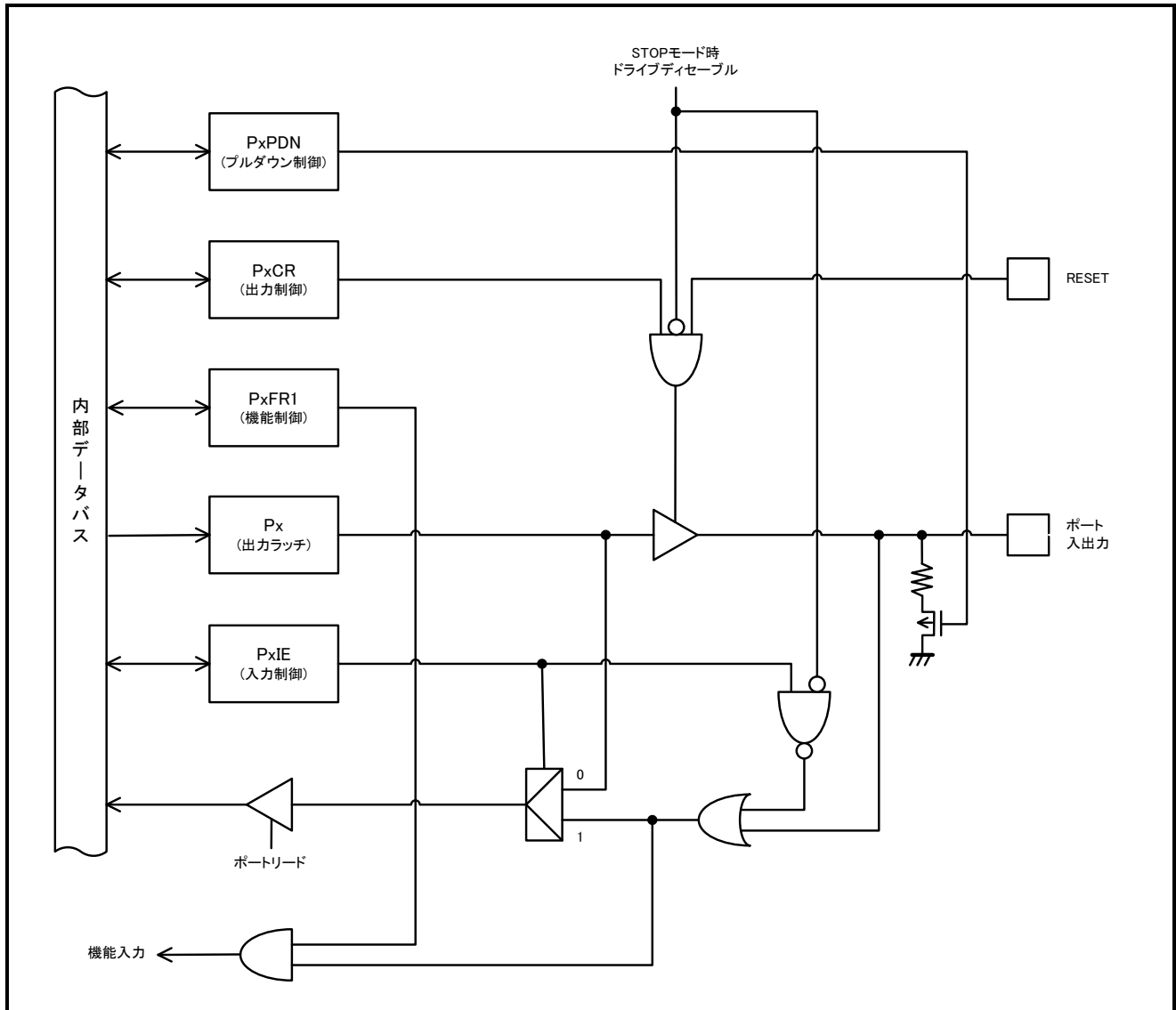
9.3.6 タイプ T5

タイプ T5 はプルアップ付きの汎用入出力ポートで、機能入力と兼用です。
 プルアップと出力は、リセット中はディセーブルになります。
 リセット中は BOOT 信号入力端子として機能します。



9.3.7 タイプ T6

タイプ T6 はプルダウン付きの汎用入出力ポートで、機能入力と兼用です。
出力はリセット中ディセーブルになります。



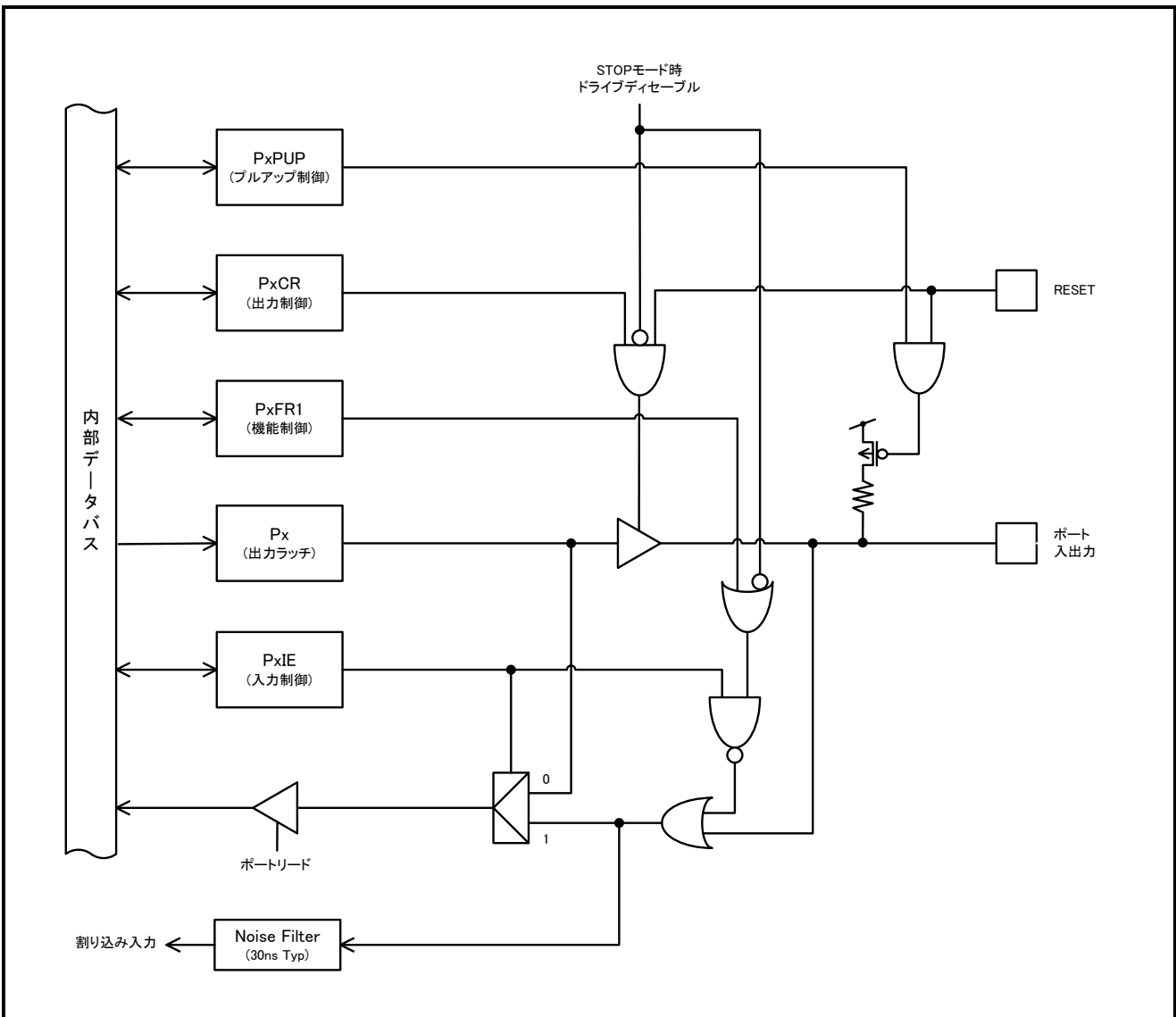
9.3.8 タイプ T7

タイプ T7 はプルアップ付きの汎用入出力ポートで、割り込み入力と兼用です。

プルアップと出力は、リセット中はディセーブルになります。

割り込み入力を STOP モード解除に使用する場合、PxFR1 レジスタで機能設定にし、PxIE レジスタで入力設定をしてください。この設定では、クロック/モード制御部のスタンバイコントロールレジスタ CGSTBYCR<DRVE>ビットで STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

(注) STOP 以外のモードでは、PxIE で入力イネーブル設定であれば PxFR の設定によらず割り込みの入力が有効になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。



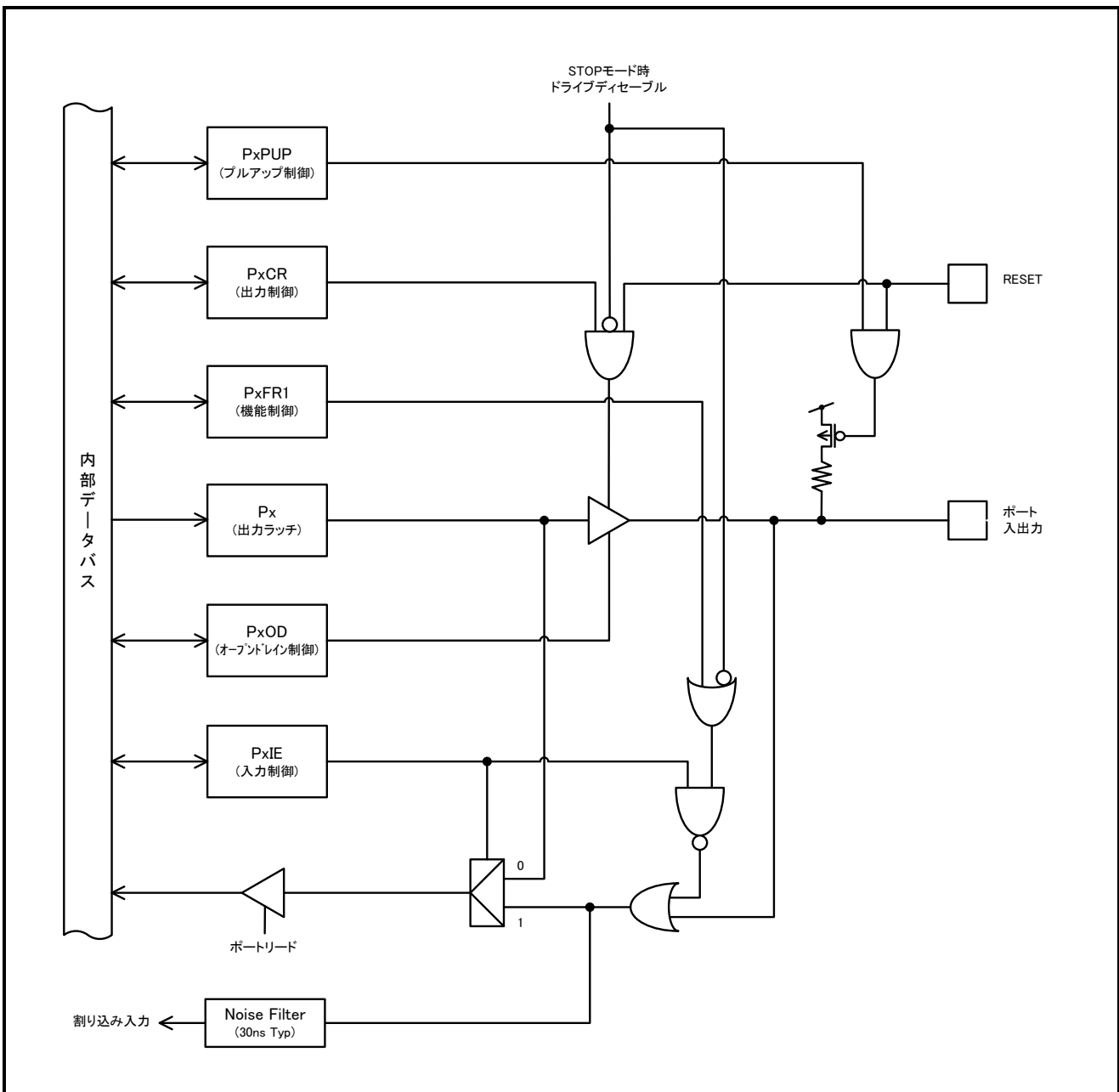
9.3.9 タイプ T8

タイプ T8 はプルアップおよびオープンドレイン付きの汎用入出力ポートで、割り込み入力と兼用です。

プルアップと出力は、リセット中はディセーブルになります。

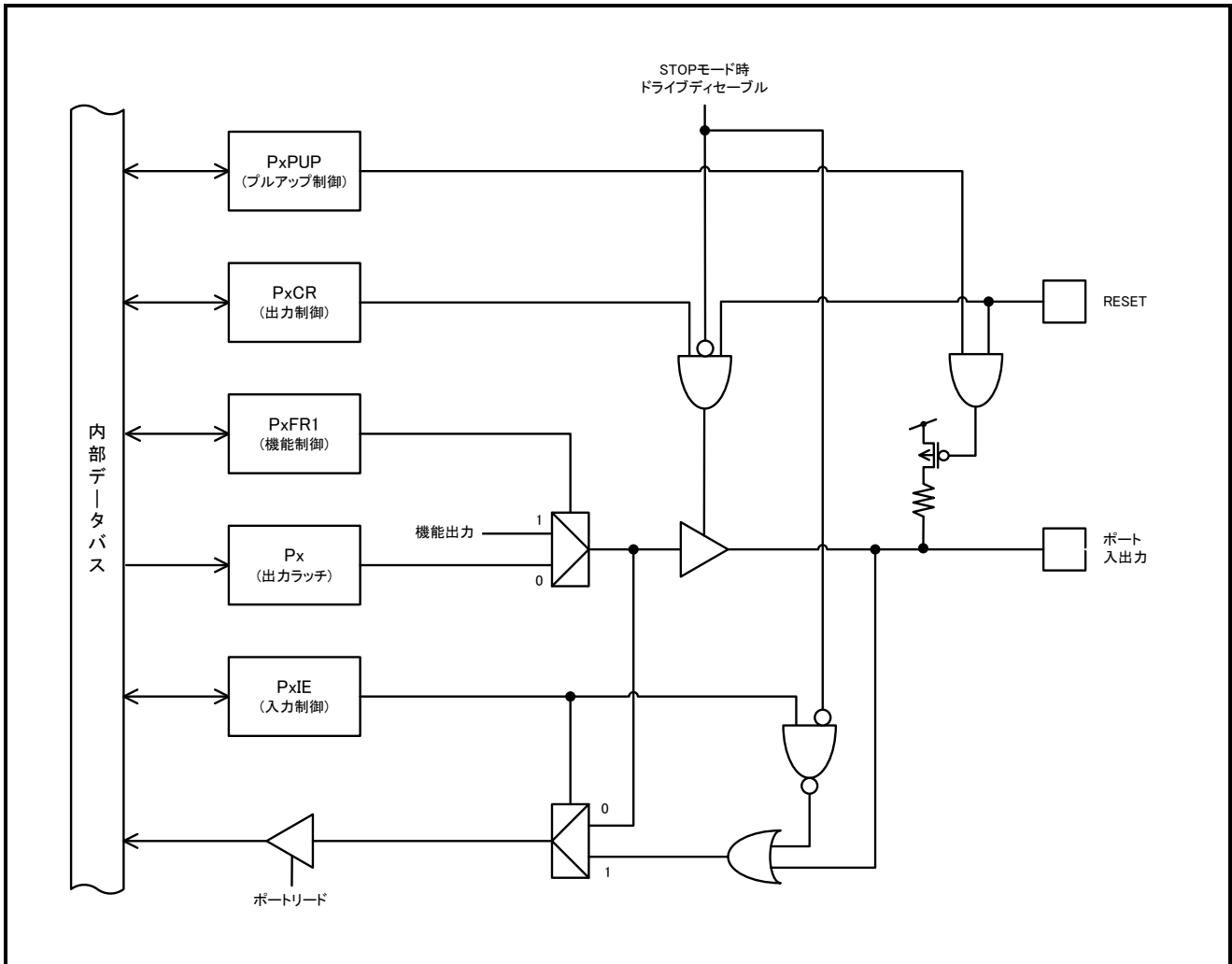
割り込み入力を STOP モード解除に使用する場合、PxFR1 レジスタで機能設定にし、PxIE レジスタで入力設定をしてください。この設定では、クロック/モード制御部のスタンバイコントロールレジスタ CGSTBYCR<DRVE>ビットで STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

(注) STOP 以外のモードでは、PxIE で入力イネーブル設定であれば PxFR の設定によらず割り込みの入力が有効になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。



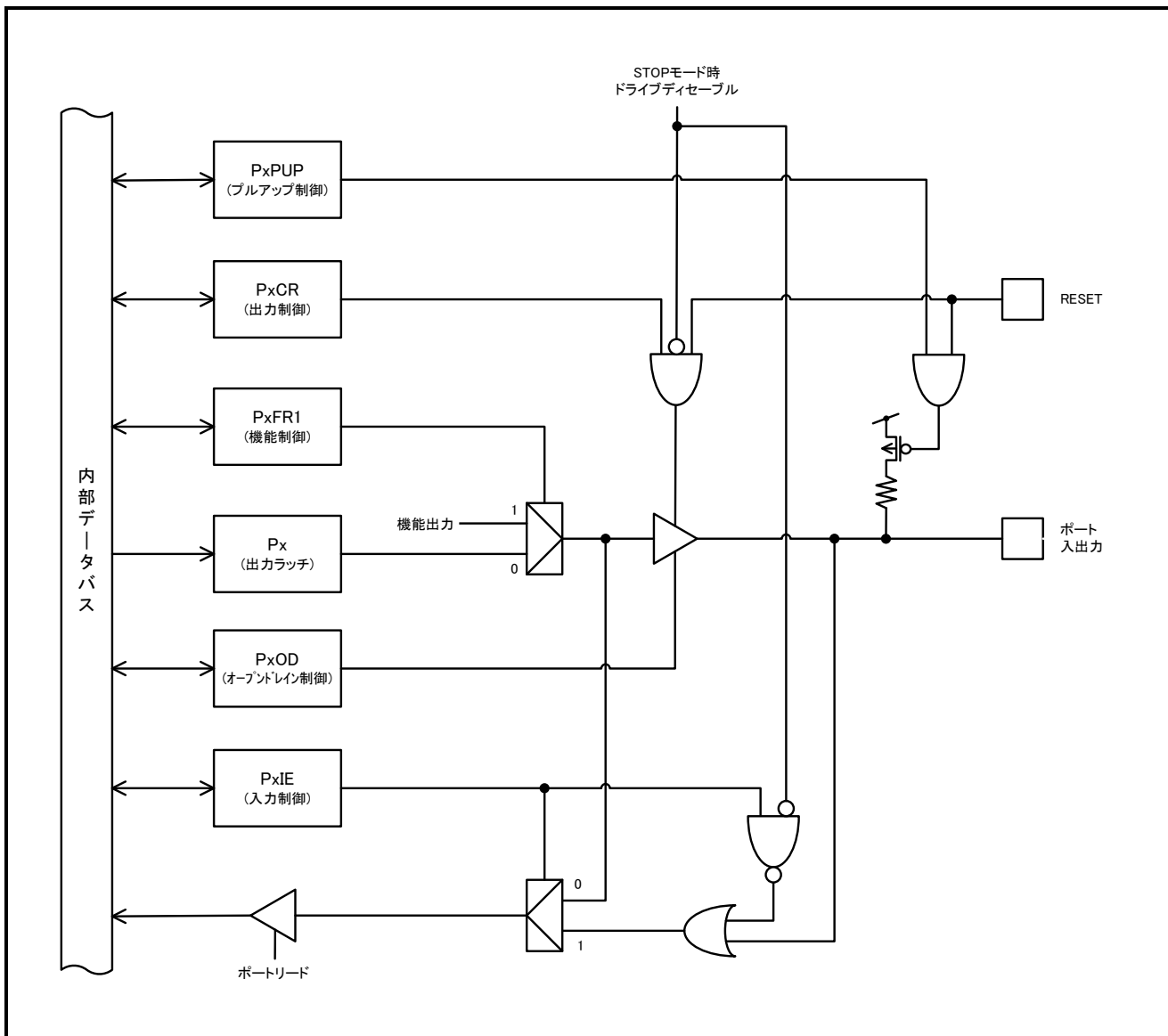
9.3.10 タイプ T9

タイプ T9 はプルアップ付きの汎用入出力ポートで、機能出力と兼用です。
 プルアップと出力は、リセット中はディセーブルになります。



9.3.11 タイプ T10

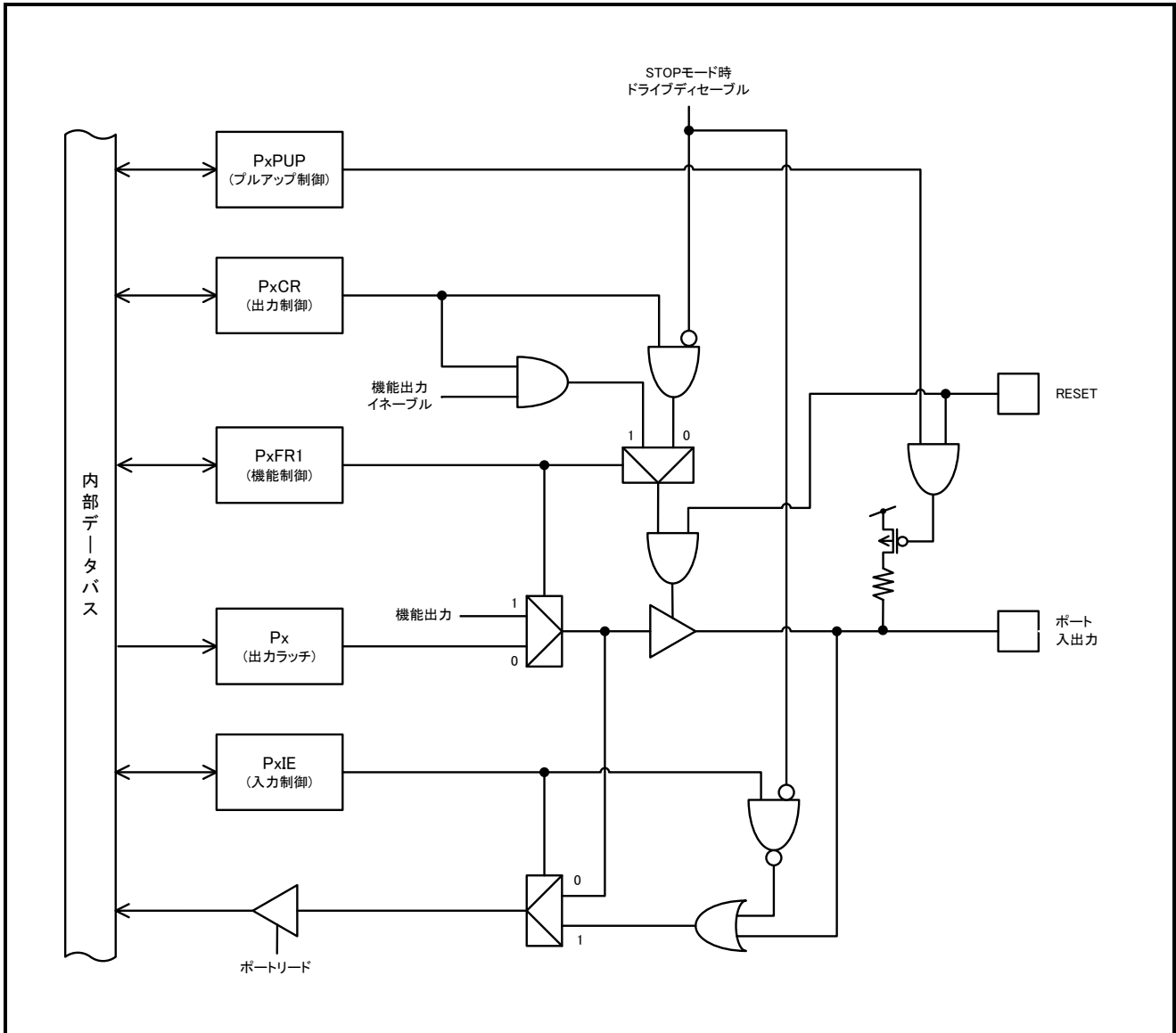
タイプ T10 はプルアップおよびオープンドレイン付きの汎用入出力ポートで、機能出力と兼用です。プルアップと出力は、リセット中はディセーブルになります。



9.3.12 タイプ T 1 1

タイプ T11 はプルアップつきの汎用入出力ポートで、機能出力と兼用です。機能出力はイネーブル信号で制御されており、有効なときのみ出力されます。

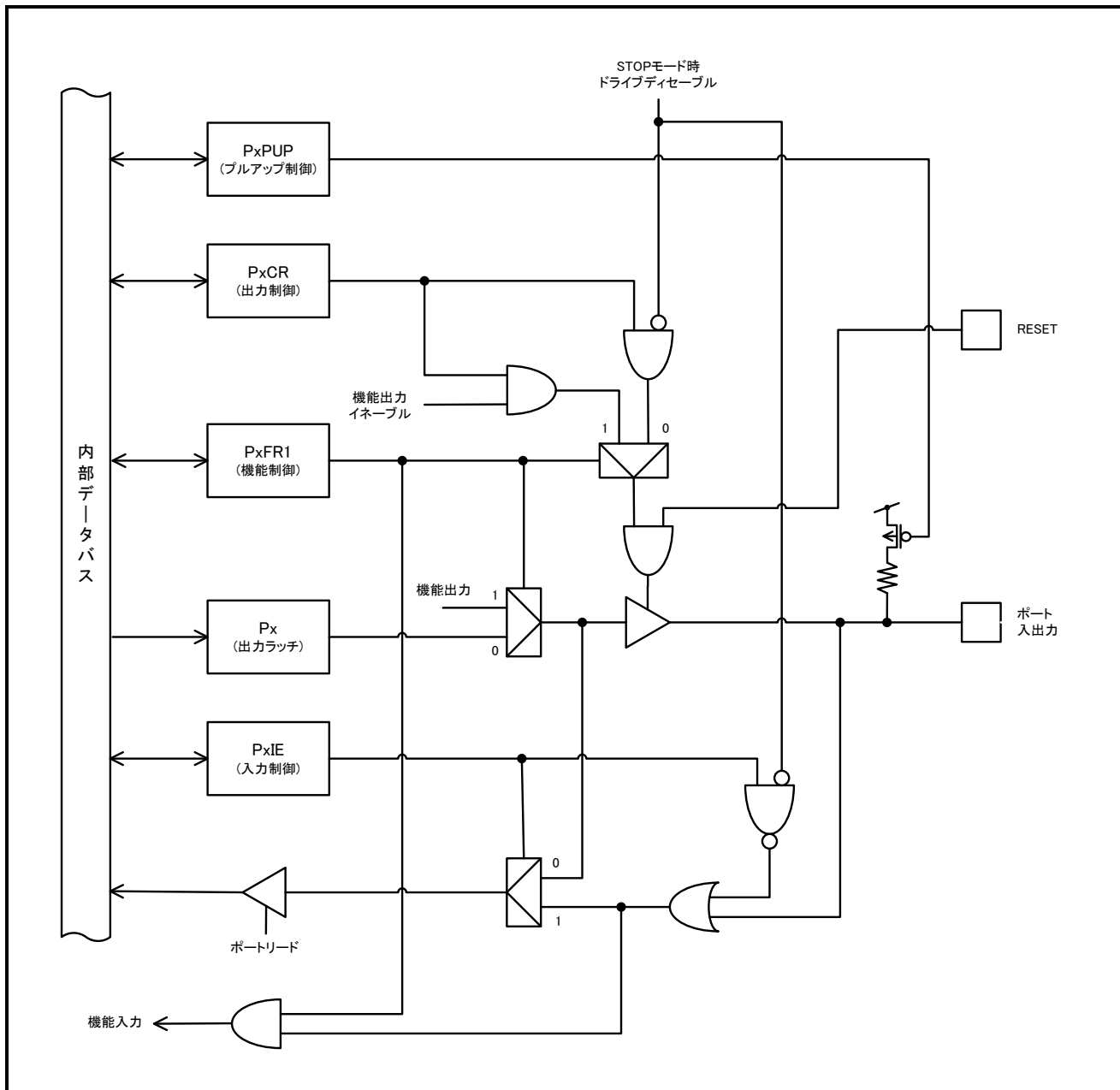
プルアップと出力は、リセット中はディセーブルになります。



9.3.13 タイプ T12

タイプ T12 はプルアップ付きの汎用入出力ポートで、機能入出力と兼用です。機能出力はイネーブル信号で制御されており、有効なときのみ出力されます。

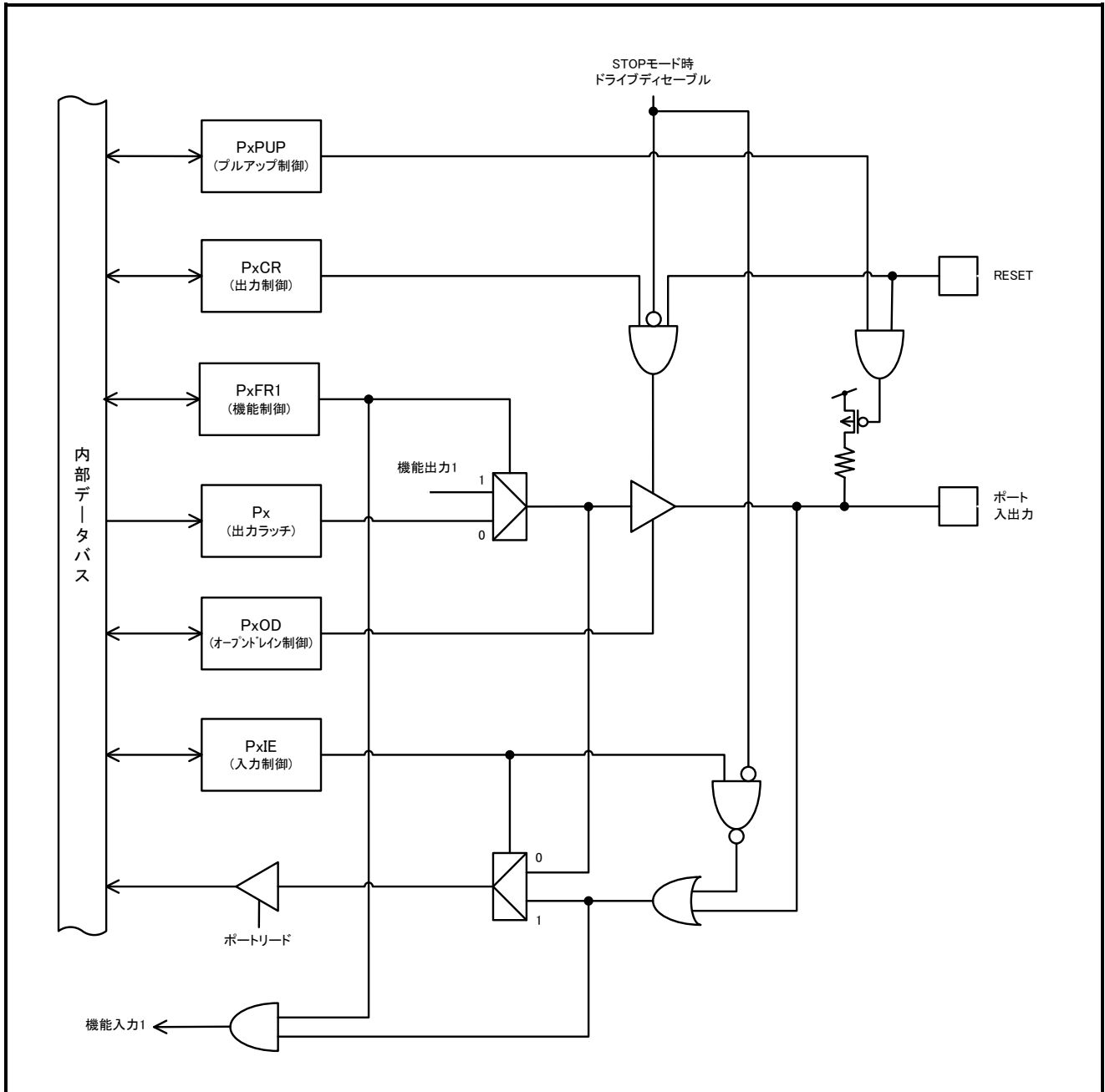
出力はリセット中ディセーブルになります。



9.3.14 タイプ T 1 3

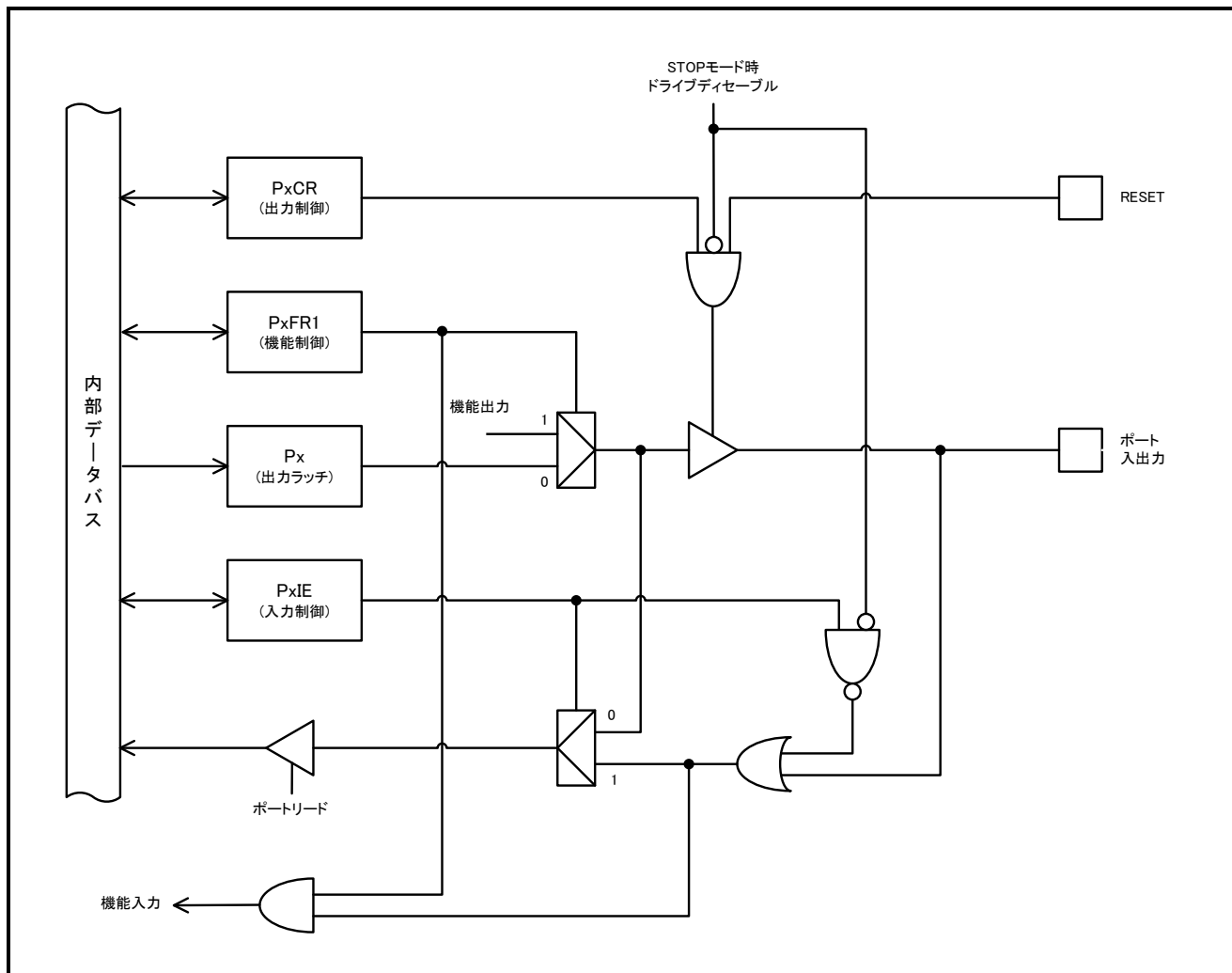
タイプ T13 はプルアップおよびオープンドレイン付きの汎用入出力ポートで、機能入出力と兼用です。

プルアップと出力は、リセット中はディセーブルになります。



9.3.15 タイプ T14

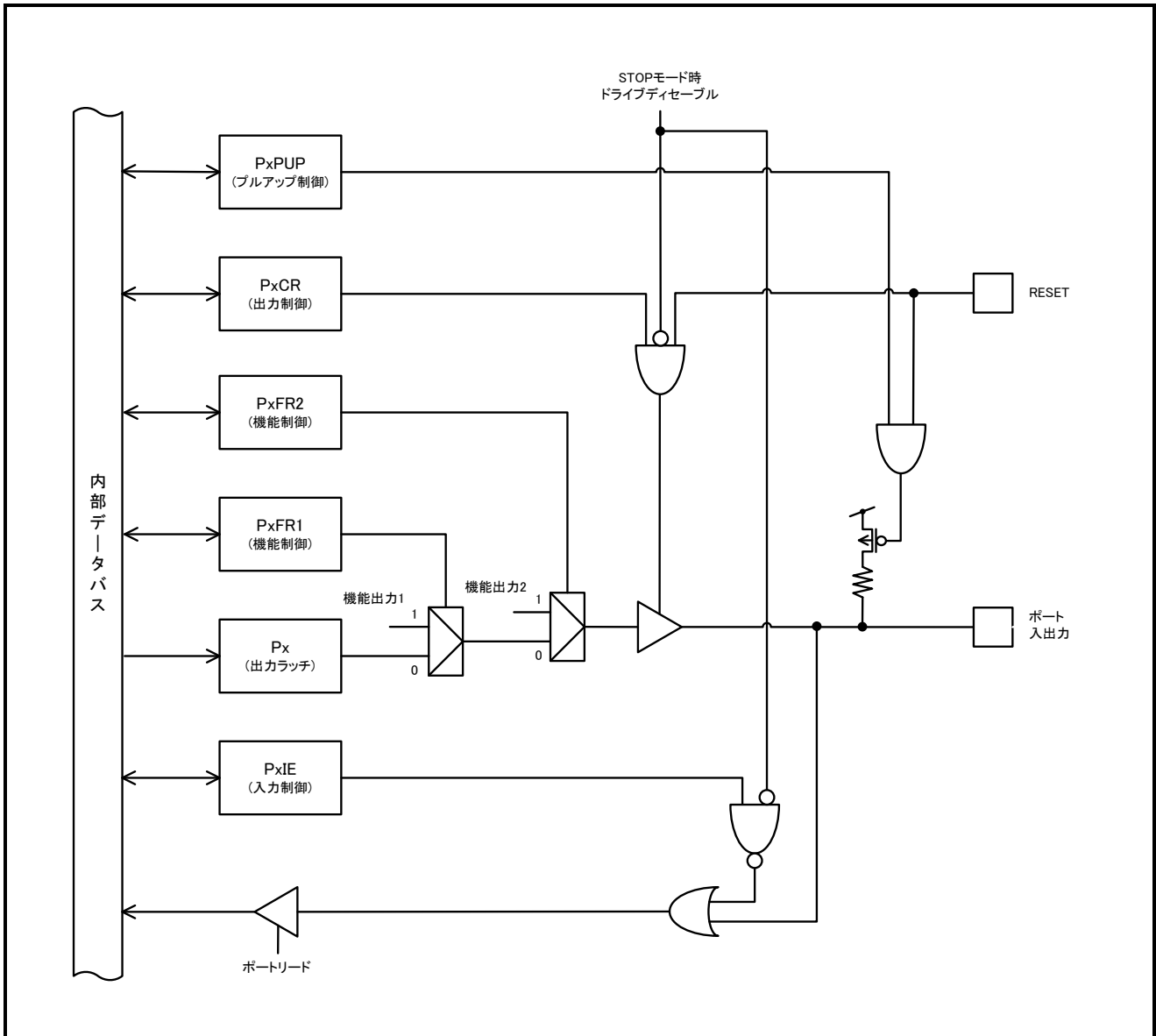
タイプ T14 は汎用入出力ポートで、機能入出力と兼用です。
出力はリセット中ディセーブルになります。



(注) タイプ T14 を使用している PK0 端子は N チャネルオープンドレインタイプの端子です。

9.3.16 タイプ T 1 5

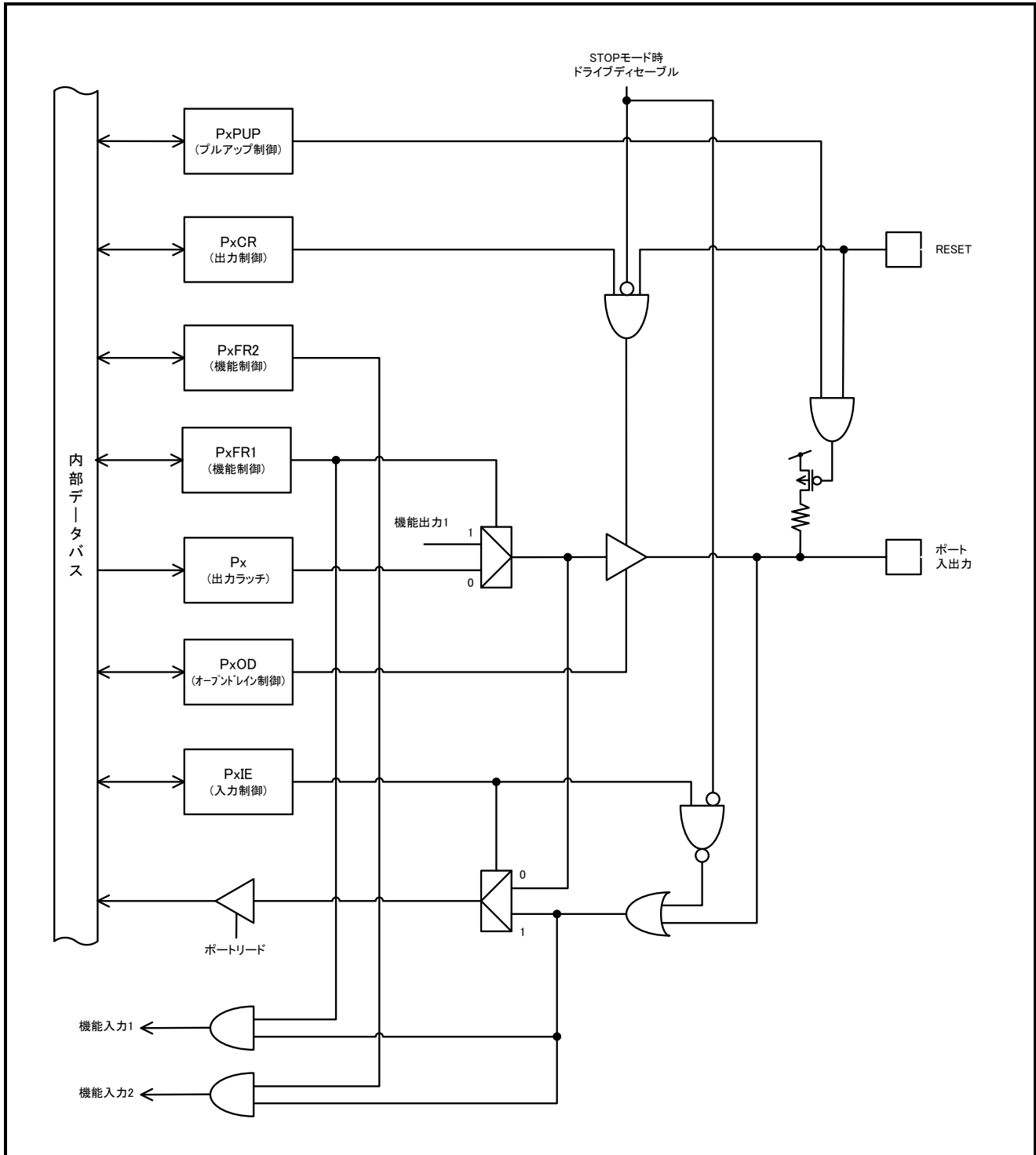
タイプ T15 はプルアップ付きの汎用入出力ポートで、2 種の機能出力と兼用です。
 プルアップと出力は、リセット中はディセーブルになります。



9.3.17 タイプ T16

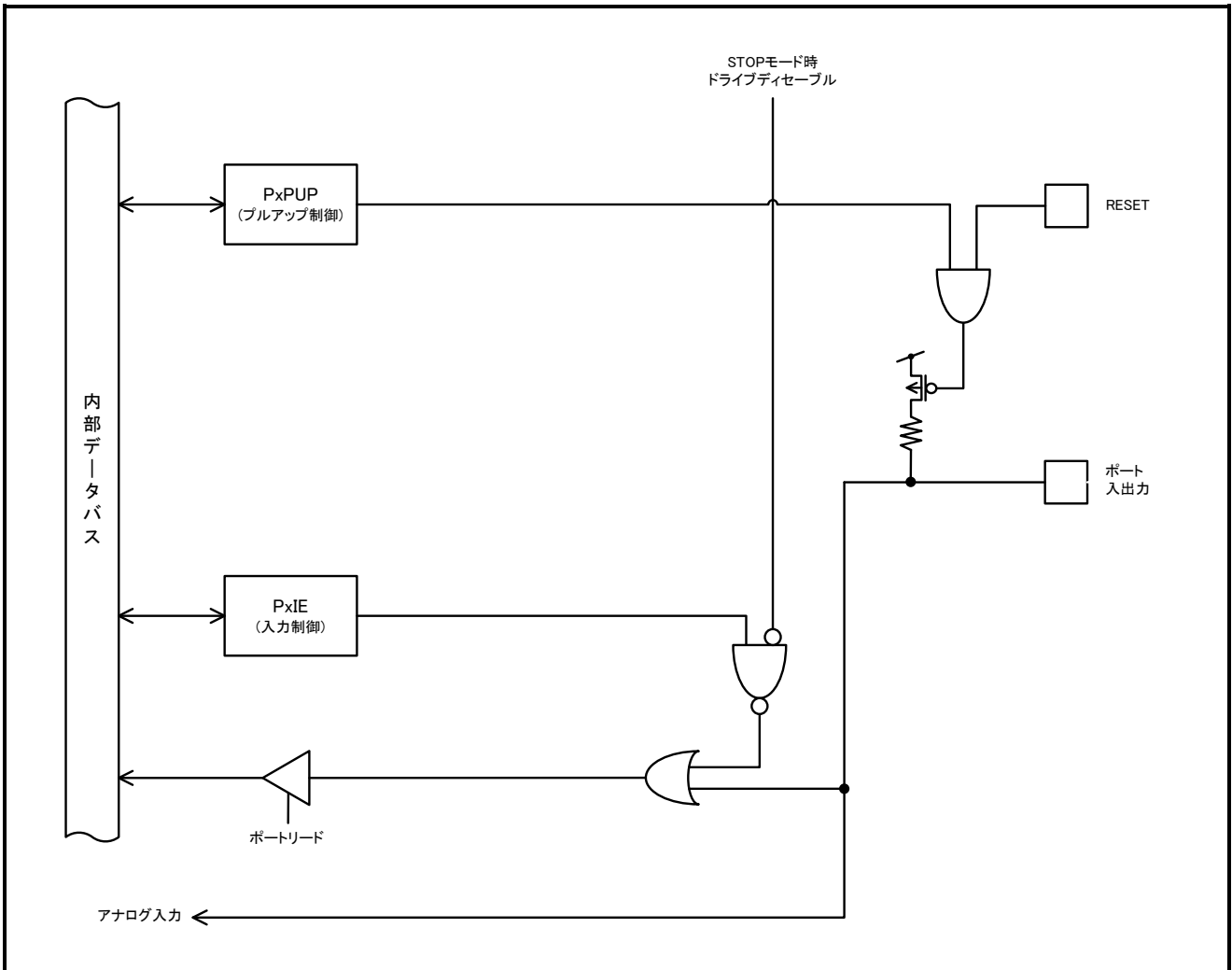
タイプ T16 はプルアップおよびオープンドレイン付きの汎用入出力ポートで、機能 1(入出力)、機能 2 (入力) と兼用です。

プルアップと出力は、リセット中はディセーブルになります。



9.3.18 タイプ T17

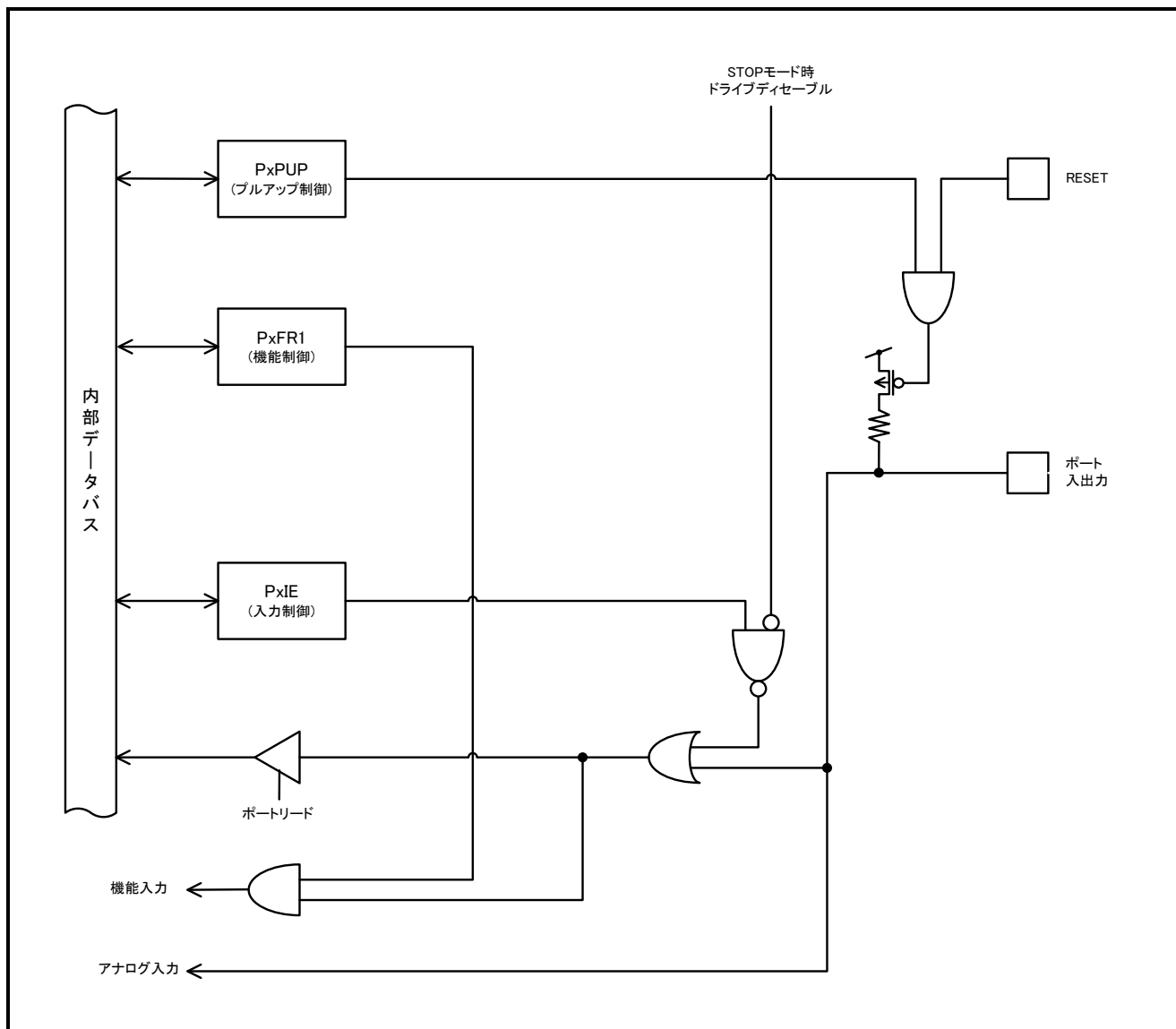
タイプ T17 はプルアップ付きの汎用入力ポートで、A/D コンバータ用アナログ信号入力と兼用です。
 プルアップは、リセット中はディセーブルになります。



9.3.19 タイプ T18

タイプ T18 はプルアップ付きの汎用入力ポートで、機能入力および A/D コンバータ用アナログ信号入力と兼用です。

プルアップは、リセット中はディセーブルになります。



10 16 ビットタイマ/イベントカウンタ (TMRB)

10.1 概要

多機能 16 ビットタイマ/イベントカウンタを 10 チャンネル (TMRB0~TMRB9) 内蔵しています。
TMRB は、次の 4 つの動作モードをもっています。

- 16 ビットインタバルタイマモード
- 16 ビットイベントカウンタモード
- 16 ビットプログラマブル矩形波出力 (PPG) モード
- タイマ同期モード (4ch 毎に同期出力設定可能)

また、キャプチャ機能を利用することで、次のような用途に使用することができます。

- 周波数測定
- パルス幅測定
- 時間差測定

10.2 チャンネル別仕様相違点

各チャンネル (TMRB0~TMRB9)はそれぞれ独立に動作します。いずれのチャンネルも表 10-1、表 10-2 に示される仕様相違点を除いて同一の動作をしますので、動作説明は TMRB0 の場合についてのみ説明します。

下記チャンネルはキャプチャトリガや、同期トリガに使用されます。

(1) TMRB7、TMRB8、TMRB9 のフリップフロップ出力を他のチャンネルのキャプチャトリガとして使用可能

- ・ TB7OUT => TMRB 0 ~ 1 で使用
- ・ TB8OUT => TMRB 2 ~ 4 で使用
- ・ TB9OUT => TMRB 5 ~ 6 で使用

(2) タイマ同期モードのスタートトリガ (TBxRUN を使用)

- ・ TMRB0 => TMRB0, 1, 2, 3 を同時スタート
- ・ TMRB4 => TMRB4, 5, 6, 7 を同時スタート

表 10-1 TMRB のチャンネル別仕様相違点(1)

仕様 チャンネル	外部端子		タイマ間トリガ機能	
	外部クロック/ キャプチャトリガ入力端子	タイマフリップフロップ 出力端子	キャプチャトリガ用タイマ	同期スタートトリガタイマ
TMRB0	TB0IN0 (PH0 と兼用) TB0IN1 (PH1 と兼用)	TB0OUT (PI0 と兼用)	TB7OUT	—
TMRB1	TB1IN0 (PH2 と兼用) TB1IN1 (PH3 と兼用)	TB1OUT (PI1 と兼用)	TB7OUT	TB0PRUN, TB0RUN
TMRB2	TB2IN0 (PH4 と兼用) TB2IN1 (PH5 と兼用)	TB2OUT (PI2 と兼用)	TB8OUT	TB0PRUN, TB0RUN
TMRB3	TB3IN0 (PH6 と兼用) TB3IN1 (PH7 と兼用)	TB3OUT (PI3 と兼用)	TB8OUT	TB0PRUN, TB0RUN
TMRB4	TB4IN0 (PI6 と兼用) TB4IN1 (PI7 と兼用)	TB4OUT (PI4 と兼用)	TB8OUT	—
TMRB5	TB5IN0 (PD0 と兼用) TB5IN1 (PD1 と兼用)	TB5OUT (PI5 と兼用)	TB9OUT	TB4PRUN, TB4RUN
TMRB6	TB6IN0 (PD2 と兼用) TB6IN1 (PD3 と兼用)	TB6OUT (PJ4 と兼用)	TB9OUT	TB4PRUN, TB4RUN
TMRB7	—	TB7OUT (PJ5 と兼用)	—	TB4PRUN, TB4RUN
TMRB8	—	TB8OUT (PG7 と兼用)	—	—
TMRB9	—	TB9OUT (PK2 と兼用)	—	—

表 10-2 TMRB のチャンネル別仕様相違点(2)

チャンネル	仕様	割込み	
		キャプチャ割込み	TMRB 割込み
TMRB0		INTCAP00 INTCAP01	INTTB0
TMRB1		INTCAP10 INTCAP11	INTTB1
TMRB2		INTCAP20 INTCAP21	INTTB2
TMRB3		INTCAP30 INTCAP31	INTTB3
TMRB4		INTCAP40 INTCAP40	INTTB4
TMRB5		INTCAP50 INTCAP51	INTTB5
TMRB6		INTCAP60 INTCAP61	INTTB6
TMRB7		—	INTTB7
TMRB8		—	INTTB8
TMRB9		—	INTTB9

10.3 構成

各チャンネルは、主に 16 ビットアップカウンタ、16 ビットタイマレジスタ 2 本 (ダブルバッファ構造)、16 ビットのキャプチャレジスタ 2 本、コンパレータ 2 個、および、キャプチャ入力制御、タイマフリップフロップとその制御回路で構成されています。タイマの動作モードやタイマフリップフロップはレジスタで制御されます。

- | |
|---|
| <p>(注1) チャンネル 0, 1 には TB7OUT, チャンネル 2, 3, 4 には TB8OUT, チャンネル 5, 6 には TB9OUT が入力されません。</p> <p>(注2) チャンネル 7~9 の TBxOUT は、端子に出力されません。
チャンネル 7~9 は、TBnIN0、TBnIN1 の端子入力がありません。
チャンネル 7~9 は、INTCAPn0、INTCAPn1 のキャプチャ割り込みを使用できません。</p> |
|---|

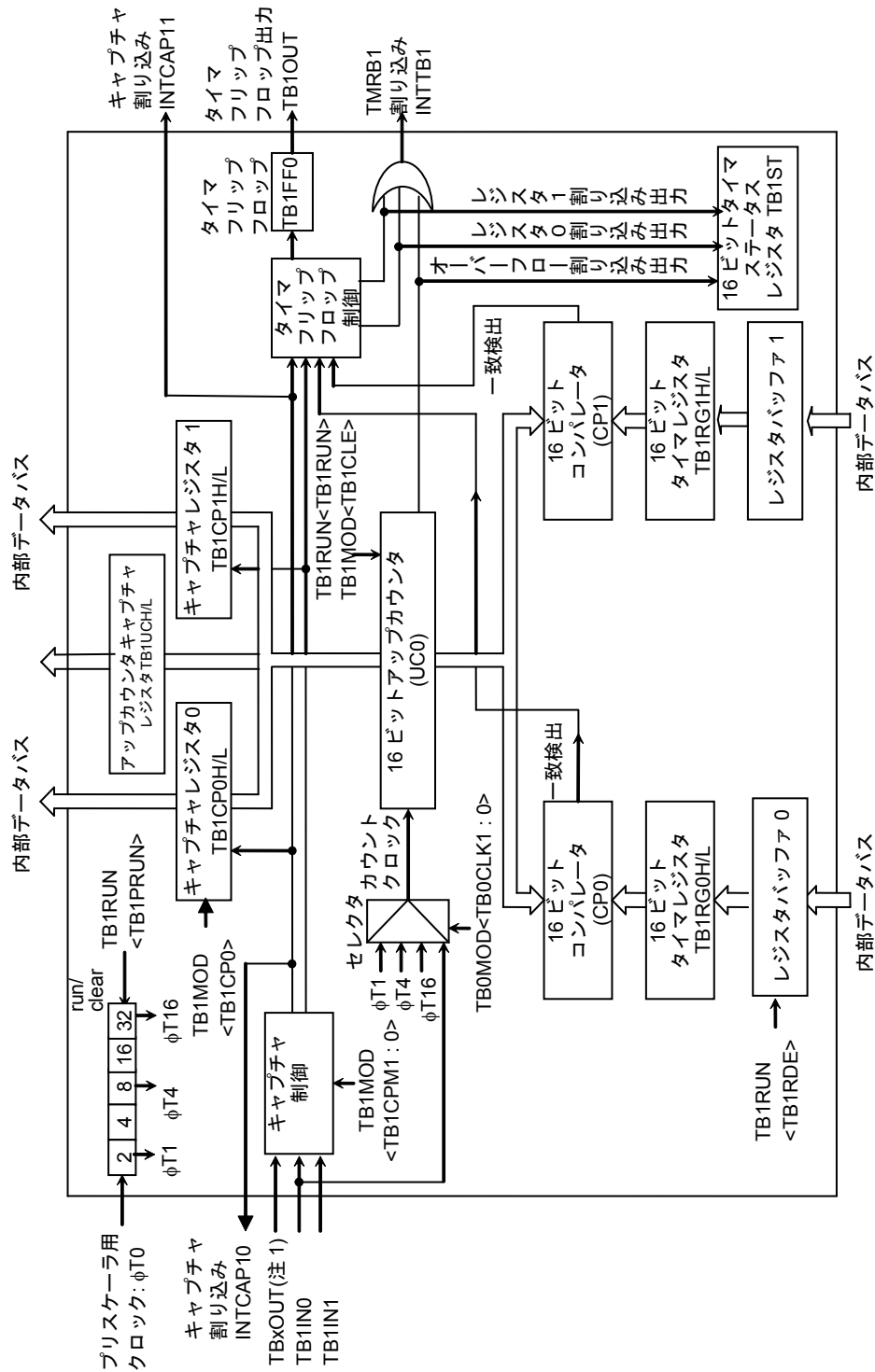


図 10-1 TMRB1 ブロック図 (チャンネル 0,2~9 も同様)

10.4 レジスタ説明

10.4.1 チャンネル別レジスタ一覧

各チャンネルのレジスタとアドレスを表 10-3に示します。

表 10-3 TMRB のチャンネル別レジスタ一覧 (1/2)

仕様		チャンネル		TMRB0		TMRB1		TMRB2		TMRB3	
レジスタ名 (アドレス)	タイマイネーブルレジスタ	TB0EN	0x4001_0000	TB1EN	0x4001_0040	TB2EN	0x4001_0080	TB3EN	0x4001_00C0		
	タイマ RUN レジスタ	TB0RUN	0x4001_0004	TB1RUN	0x4001_0044	TB2RUN	0x4001_0084	TB3RUN	0x4001_00C4		
	タイマコントロール レジスタ	TB0CR	0x4001_0008	TB1CR	0x4001_0048	TB2CR	0x4001_0088	TB3CR	0x4001_00C8		
	タイマモードレジスタ	TB0MOD	0x4001_000C	TB1MOD	0x4001_004C	TB2MOD	0x4001_008C	TB3MOD	0x4001_00CC		
	タイマフリップフロップ コントロールレジスタ	TB0FFCR	0x4001_0010	TB1FFCR	0x4001_0050	TB2FFCR	0x4001_0090	TB3FFCR	0x4001_00D0		
	タイマステータスレジスタ	TB0ST	0x4001_0014	TB1ST	0x4001_0054	TB2ST	0x4001_0094	TB3ST	0x4001_00D4		
	割り込みマスクレジスタ	TB0IM	0x4001_0018	TB1IM	0x4001_0058	TB2IM	0x4001_0098	TB3IM	0x4001_00D8		
	タイマアップカウンタ レジスタ	TB0UC	0x4001_001C	TB1UC	0x4001_005C	TB2UC	0x4001_009C	TB3UC	0x4001_00DC		
	タイマレジスタ	TB0RG0 TB0RG1	0x4001_0020 0x4001_0024	TB1RG0 TB1RG1	0x4001_0060 0x4001_0064	TB2RG0 TB2RG1	0x4001_00A0 0x4001_00A4	TB3RG0 TB3RG1	0x4001_00E0 0x4001_00E4		
	キャプチャレジスタ	TB0CP0 TB0CP1	0x4001_0028 0x4001_002C	TB1CP0 TB1CP1	0x4001_0068 0x4001_006C	TB2CP0 TB2CP1	0x4001_00A8 0x4001_00AC	TB3CP0 TB3CP1	0x4001_00E8 0x4001_00EC		

仕様		チャンネル		TMRB4		TMRB5		TMRB6		TMRB7	
レジスタ名 (アドレス)	タイマイネーブルレジスタ	TB4EN	0x4001_0100	TB5EN	0x4001_0140	TB6EN	0x4001_0180	TB7EN	0x4001_01C0		
	タイマ RUN レジスタ	TB4RUN	0x4001_0104	TB5RUN	0x4001_0144	TB6RUN	0x4001_0184	TB7RUN	0x4001_01C4		
	タイマコントロール レジスタ	TB4CR	0x4001_0108	TB5CR	0x4001_0148	TB6CR	0x4001_0188	TB7CR	0x4001_01C8		
	タイマモードレジスタ	TB4MOD	0x4001_010C	TB5MOD	0x4001_014C	TB6MOD	0x4001_018C	TB7MOD	0x4001_01CC		
	タイマフリップフロップ コントロールレジスタ	TB4FFCR	0x4001_0110	TB5FFCR	0x4001_0150	TB6FFCR	0x4001_0190	TB7FFCR	0x4001_01D0		
	タイマステータスレジスタ	TB4ST	0x4001_0114	TB5ST	0x4001_0154	TB6ST	0x4001_0194	TB7ST	0x4001_01D4		
	割り込みマスクレジスタ	TB4IM	0x4001_0118	TB5IM	0x4001_0158	TB6IM	0x4001_0198	TB7IM	0x4001_01D8		
	タイマアップカウンタ レジスタ	TB4UC	0x4001_011C	TB5UC	0x4001_015C	TB6UC	0x4001_019C	TB7UC	0x4001_01DC		
	タイマレジスタ	TB4RG0 TB4RG1	0x4001_0120 0x4001_0124	TB5RG0 TB5RG1	0x4001_0160 0x4001_0164	TB6RG0 TB6RG1	0x4001_01A0 0x4001_01A4	TB7RG0 TB7RG1	0x4001_01E0 0x4001_01E4		
	キャプチャレジスタ	TB4CP0 TB4CP1	0x4001_0128 0x4001_012C	TB5CP0 TB5CP1	0x4001_0168 0x4001_016C	TB6CP0 TB6CP1	0x4001_01A8 0x4001_01AC	TB7CP0 TB7CP1	0x4001_01E8 0x4001_01EC		

表 10-3 TMRB のチャンネル別レジスタ一覧 (2/2)

仕様	チャンネル	TMRB8		TMRB9	
		レジスタ名	アドレス	レジスタ名	アドレス
	タイマイネーブルレジスタ	TB8EN	0x4001_0200	TB9EN	0x4001_0240
	タイマ RUN レジスタ	TB8RUN	0x4001_0204	TB9RUN	0x4001_0244
	タイマコントロール レジスタ	TB8CR	0x4001_0208	TB9CR	0x4001_0248
	タイマモードレジスタ	TB8MOD	0x4001_020C	TB9MOD	0x4001_024C
	タイマフリップフロップ コントロールレジスタ	TB8FFCR	0x4001_0210	TB9FFCR	0x4001_0250
	タイマステータスレジスタ	TB8ST	0x4001_0214	TB9ST	0x4001_0254
	割り込みマスクレジスタ	TB8IM	0x4001_0218	TB9IM	0x4001_0258
	タイマアップカウンタ レジスタ	TB8UC	0x4001_021C	TB9UC	0x4001_025C
	タイマレジスタ	TB8RG0	0x4001_0220	TB9RG0	0x4001_0260
		TB8RG1	0x4001_0224	TB9RG1	0x4001_0264
	キャプチャレジスタ	TB8CP0	0x4001_0228	TB9CP0	0x4001_0268
		TB8CP1	0x4001_022C	TB9CP1	0x4001_026C

10.4.1.1 TMRBn イネーブルレジスタ (チャンネル0~9)

TMRBn コントロールレジスタ (n=0~9)

TnBnEN (0x4001_0xx0)		31	30	29	28	27	26	25	24
	bit Symbol								
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		23	22	21	20	19	18	17	16
	bit Symbol								
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		15	14	13	12	11	10	9	8
	bit Symbol								
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0	
bit Symbol	TnBnEN								
Read/Write	R/W	R							
リセット後	0	0							
機能	TMRBn 動作 0: 禁止 1: 許可	リードすると"0"が読めます							

<TnBnEN> : TMRB の動作を指定します。動作禁止の状態では TMRB モジュールの他のレジスタへクロックが供給されませんので消費電力の低減が可能です (他のレジスタへのリード、ライトはできません)。TMRB を使用する場合は、TMRB モジュールの各レジスタを設定する前に TMRB 動作許可 ("1") にしてください。TMRB を一旦動作させた後に、動作禁止した場合は各レジスタの設定は保持されます。

10.4.1.2 TMRB RUN レジスタ (チャンネル 0~9)

TMRBn RUN レジスタ (n=0~9)

TBnRUN
(0x4001_0xx4)

	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit Symbol						TBnPRUN		TBnRUN
Read/Write	R					R/W	R	R/W
リセット後	0					0	0	0
機能	リードすると"0"が読めます					Timer Run/Stop Control 0: 停止&クリア 1: カウント ※ 1 bit 目は"0"が読めます		

<TBnRUN> : TMRB0 のカウント動作を制御します。

<TBnPRUN> : TMRB0 のプリスケアラの動作を制御します。

(注) カウンタ停止状態(<TBRUN>="0") でアップカウンタキャプチャレジスタの TBxUC<TBUC[15:0]> をリードすると、カウンタ動作時に最後にキャプチャした値がリードされます。

10.4.1.3 TMRB コントロールレジスタ (チャンネル0~9)

TMRBn コントロールレジスタ (n=0~9)

	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit Symbol	TBnWBF		TBnSYNC		I2TBn			
Read/Write	R/W	R/W	R/W	R	R/W	R		
リセット後	0	0	0	0	0	0		
機能	Double Buffer 0: 禁止 1: 許可	“0”をラ イトして ください。	同期モード 切替 0: 個別動作 1: 同期動作	リ ー ド す る と “0” が 読 め ま す	IDLE 0: 停止 1: 動作	リードすると“0”が読めます		

<I2TBm> : IDLE モード時の動作を制御します。

<TBnSYNC> : 同期モードの切り替えを制御します。

“0” : タイマ毎のタイミングで動作を行います

“1” : 同期出力を行います。

<TBmWBF> : ダブルバッファの許可/禁止を制御します。

10.4.1.4 TMRB モードレジスタ (チャンネル 0~9)

TMRBn モードレジスタ (n=0~9)

	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit Symbol			TBnCP0	TBnCPM1	TBnCPM0	TBnCLE	TBnCLK1	TBnCLK0
Read/Write	R	R/W	W	R/W				
リセット後	0	0	1	0	0	0	0	0
機能	リードすると“0”が読めます	“0”をライトしてください。	ソフトウェアキャプチャ制御 0: ソフトキャプチャ 1: Don't care	キャプチャタイミング 00: ディセーブル 01: TBnIN0 ↑ TBnIN1 ↑ 10: TBnIN0 ↑ TBnIN0 ↓ 11: TBnOUT ↑ TBnOUT ↓	アップカウンタ制御 0: クリアディセーブル 1: クリアイネーブル	ソースクロック選択 00: TBnIN0 端子入力 01: φT1 10: φT4 11: φT16		

<TBnCLK[1:0]> : TMRBn のタイマカウンタクロックを選択します。

<TBnCLE> : TMRBn のアップカウンタのクリア制御をおこないます。

“0” : アップカウンタのクリア禁止

“1” : タイマレジスタ 1 (TBnRG1) との一致でクリア

<TBnCPM[1:0]> : TMRBn のキャプチャタイミングを設定します。

“00” : キャプチャ禁止

“01” : TBnIN0 端子入力の立ち上がりでキャプチャレジスタ 0 (TBnCP0) にカウント値を取り込み、TBnIN1 端子入力の立ち上がりでキャプチャレジスタ 1 (TBnCP1) にカウント値を取り込む

“10” : TBnIN0 端子入力の立ち上がりでキャプチャレジスタ 0 (TBnCP0) にカウント値を取り込み、TBnIN0 端子入力の立ち下がりでキャプチャレジスタ 1 (TBnCP1) にカウント値を取り込む

“11” : 16 ビットタイマ一致出力 (TBxOUT) の立ち上がりでキャプチャレジスタ 0 (TBnCP0) にカウント値を取り込み、TBxOUT の立ち下がりでキャプチャレジスタ 1 (TBnCP1) にカウント値を取り込む (TMRB0~1 : TB7OUT、TMRB2~4 : TB8OUT、TMRB5~6 : TB9OUT、)

<TBnCP0> : ソフトウェアキャプチャし、キャプチャレジスタ 0 (TBnCP0) にカウント値を取り込みます。

(注 1) TBnMOD のビット 5 は、リードすると“1”が読み出されます。

(注 2) TBnIN0 端子入力、TBnIN1 端子入力については、TMRB0~6 が対象です。TMRB7~9 では機能がありません。

10.4.1.5 TMRB フリップフロップコントロールレジスタ (チャンネル 0~9)

TMRBn フリップフロップコントロールレジスタ (n=0~9)

TnFFCR (0x4001_0xx0)		31	30	29	28	27	26	25	24
	bit Symbol	/							
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		23	22	21	20	19	18	17	16
	bit Symbol	/							
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		15	14	13	12	11	10	9	8
	bit Symbol	/							
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0	
bit Symbol	/		TBnC1T1	TBnC0T1	TBnE1T1	TBnE0T1	TBnFF0C1	TBnFF0C0	
Read/Write	R		R/W				R/W		
リセット後	1	1	0	0	0	0	1	1	
機能	読み出すと常に "11" になります。		TBnFF0 反転トリガ 0: トリガディセーブル 1: トリガイネーブル				TBnFF0 の制御 00: Invert 01: Set 10: Clear 11: Don't care		
			TBnCP1 へのアップ カウンタ値 取り込み時	TBnCP0 へのアップ カウンタ値 取り込み時	アップ カウンタと TBnRG1 と の一致時	アップ カウンタと TBnRG0 と の一致時	※ 読み出すと常に "11" になります。		

<TBnFF0C[1:0]> : タイマフリップフロップを制御します。

- "00" : TBnFF0 の値を反転 (ソフト反転) します
- "01" : TBnFF0 を"1"にセットします
- "10" : TBnFF0 を"0"にクリアします
- "11" : Don't care

<TBnE[1:0]> : アップカウンタとタイマレジスタ 0,1 (TBnRG0,1) との一致時にタイマフリップフロップを反転します。

<TBnC[1:0]> : アップカウンタの値がキャプチャレジスタ 0,1 (TBnCP0,1) に取り込まれた時にタイマフリップフロップを反転します。

10.4.1.6 TMRB ステータスレジスタ (チャンネル 0~9)

TMRBn ステータスレジスタ (n=0~9)

	31	30	29	28	27	26	25	24
TnST (0x4001_0xx4)	bit Symbol							
	Read/Write	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
	bit Symbol							
	Read/Write	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
	bit Symbol							
	Read/Write	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	bit Symbol					INTTBOFn	INTTBn1	INTTBn0
	Read/Write	R				R		
	リセット後	0				0	0	0
機 能	リードすると"0"が読めます					0: 割り込みは発生していない 1: 割り込みが発生	0: 割り込みは発生していない 1: 割り込みが発生	0: 割り込みは発生していない 1: 割り込みが発生

<INTTBn0> : タイマレジスタ 0 (TBnRG0) との一致割り込み発生

<INTTBn1> : タイマレジスタ 1 (TBnRG1) との一致割り込み発生

<INTTBOFn> : アップカウンタのオーバーフロー割り込み発生

(注 1) TBxIM でマスクせられていない要因のみ、CPU に対して割り込み要求が出力されます。
マスク設定されていても、フラグはセットされます。

(注 2) TnST レジスタをリードすると、フラグはクリアされます。
自動的にクリアされないため、TnST レジスタをリードしてフラグをクリアしてください。

10.4.1.7 TMRB 割込みマスクレジスタ (チャンネル 0~9)

TMRBn 割込みマスクレジスタ (n=0~9)

TBnIM
(0x4001_0xx8)

	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R	R	R	R	R	R/W	R/W	R/W
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit Symbol						TBIMOFn	TBIMn1	TBIMn0
Read/Write	R					R		
リセット後	0					0	0	0
機能	リードすると"0"が読めます					1: 割り込みをマスクする	1: 割り込みをマスクする	1: 割り込みをマスクする

<TBIMn0> : タイマレジスタ 0 (TBnRG0) との一致割り込みをマスクします。

<TBIMn1> : タイマレジスタ 1 (TBnRG1) との一致割り込みをマスクします。

<TBIMOFn> : アップカウンタのオーバーフロー割り込みをマスクします。

10.4.1.8 TMRB リードキャプチャレジスタ (チャンネル 0~9)

TBnUC リードキャプチャレジスタ (n=0~9)

TBnUC
(0x4001_0xxC)

	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit Symbol	UCn15	UCn14	UCn13	UCn12	UCn11	UCn10	UCn9	UCn8
Read/Write	R							
リセット後	0							
機 能	キャプチャによるカウンタ読み出しデータ 15-8 ビットデータ							
	7	6	5	4	3	2	1	0
bit Symbol	UCn7	UCn6	UCn5	UCn4	UCn3	UCn2	UCn1	UCn0
Read/Write	R							
リセット後	0							
機 能	キャプチャによるカウンタ読み出しデータ 7-0 ビットデータ							

(注) カウンタ動作時に TBxUC をリードすると、リード時のアップカウンタの値をキャプチャしリードすることができます。

10.4.1.9 TMRB タイマレジスタ (チャンネル 0~9)

TBnRG0 タイマレジスタ (n=0~9)

TBnRG0 (0x4001_0xx0)		31	30	29	28	27	26	25	24
	bit Symbol								
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		23	22	21	20	19	18	17	16
	bit Symbol								
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		15	14	13	12	11	10	9	8
	bit Symbol	TBnRG015	TBnRG014	TBnRG013	TBnRG012	TBnRG011	TBnRG010	TBnRG09	TBnRG08
	Read/Write	R/W							
	リセット後	0							
	機能	タイマカウント値 15-8 ビットデータ							
		7	6	5	4	3	2	1	0
	bit Symbol	TBnRG07	TBnRG06	TBnRG05	TBnRG04	TBnRG03	TBnRG02	TBnRG01	TBnRG00
	Read/Write	R/W							
	リセット後	0							
	機能	タイマカウント値 7-0 ビットデータ							

TBnRG1 タイマレジスタ (n=0~9)

TBnRG1 (0x4001_0xx4)		31	30	29	28	27	26	25	24
	bit Symbol								
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		23	22	21	20	19	18	17	16
	bit Symbol								
	Read/Write	R	R	R	R	R	R	R	R
	リセット後	0	0	0	0	0	0	0	0
		15	14	13	12	11	10	9	8
	bit Symbol	TBnRG115	TBnRG114	TBnRG113	TBnRG112	TBnRG111	TBnRG110	TBnRG19	TBnRG18
	Read/Write	R/W							
	リセット後	0							
	機能	タイマカウント値 15-8 ビットデータ							
		7	6	5	4	3	2	1	0
	bit Symbol	TBnRG17	TBnRG16	TBnRG15	TBnRG14	TBnRG13	TBnRG12	TBnRG11	TBnRG10
	Read/Write	R/W							
	リセット後	0							
	機能	タイマカウント値 7-0 ビットデータ							

10.4.1.10 TMRB キャプチャレジスタ (チャンネル 0~9)

TBnCP0 キャプチャレジスタ (n=0~9)

	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit Symbol	TBnCP015	TBnCP014	TBnCP013	TBnCP012	TBnCP011	TBnCP010	TBnCP09	TBnCP08
Read/Write	R							
リセット後	不定							
機能	タイマキャプチャ値 15-8 ビットデータ							
	7	6	5	4	3	2	1	0
bit Symbol	TBnCP07	TBnCP06	TBnCP05	TBnCP04	TBnCP03	TBnCP02	TBnCP01	TBnCP00
Read/Write	R							
リセット後	不定							
機能	タイマキャプチャ値 7-0 ビットデータ							

TBnCP1 キャプチャレジスタ (n=0~9)

	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit Symbol	TBnCP115	TBnCP114	TBnCP113	TBnCP112	TBnCP111	TBnCP110	TBnCP19	TBnCP18
Read/Write	R							
リセット後	不定							
機能	タイマキャプチャ値 15-8 ビットデータ							
	7	6	5	4	3	2	1	0
bit Symbol	TBnCP17	TBnCP16	TBnCP15	TBnCP14	TBnCP13	TBnCP12	TBnCP11	TBnCP10
Read/Write	R							
リセット後	不定							
機能	タイマキャプチャ値 7-0 ビットデータ							

10.5 回路別の動作説明

各チャンネルは表 10-1、表 10-2に示される仕様相違点を除いて同一の動作をしますので、チャンネル0を例に説明します。

10.5.1 プリスケーラ

アップカウンタ UCO のソースクロックを生成する4ビットのプリスケーラです。

プリスケーラへの入力クロック $\phi T0$ はCG部のCGSYSCR<PRCK[2:0]>にて選択した $f_{periph/1}$ 、 $f_{periph/2}$ 、 $f_{periph/4}$ 、 $f_{periph/8}$ 、 $f_{periph/16}$ 、 $f_{periph/32}$ のいずれかのクロックです。このペリフェラルクロック f_{periph} はCG部のCGSYSCR<FPSEL>で選択したクロック f_{gear} またはクロックギア分周前のクロック f_c のいずれかのクロックです。

プリスケーラはTB0RUN<TB0PRUN>により動作/停止の設定をします。“1”をライトするとカウント開始し“0”をライトするとクリアされ停止します。プリスケーラ出力クロックの分解能を、表 10-4、表 10-5に示します。

表 10-4 プリスケアラ出カクロック分解能 (fc=40MHz)

ペリフェラル クロック解除 <FPSEL>	クロックギア値 <GEAR[2:0]>	プリスケアラ クロック選択 <PRCK[2:0]>	プリスケアラ出カクロック分解能		
			$\phi T1$	$\phi T4$	$\phi T16$
0 (fgear)	000 (fc)	000 (fper iph/1)	$fc/2^1 (0.05\mu s)$	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$
		001 (fper iph/2)	$fc/2^2 (0.1\mu s)$	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		100 (fper iph/16)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		101 (fper iph/32)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$
	100 (fc/2)	000 (fper iph/1)	$fc/2^2 (0.1\mu s)$	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$
		001 (fper iph/2)	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$
		010 (fper iph/4)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		011 (fper iph/8)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		100 (fper iph/16)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$
		101 (fper iph/32)	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$	$fc/2^{11} (51.2\mu s)$
	101 (fc/4)	000 (fper iph/1)	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$
		001 (fper iph/2)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		010 (fper iph/4)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		011 (fper iph/8)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$
		100 (fper iph/16)	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$	$fc/2^{11} (51.2\mu s)$
		101 (fper iph/32)	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$	$fc/2^{12} (102.4\mu s)$
	110 (fc/8)	000 (fper iph/1)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		001 (fper iph/2)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		010 (fper iph/4)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$
		011 (fper iph/8)	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$	$fc/2^{11} (51.2\mu s)$
		100 (fper iph/16)	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$	$fc/2^{12} (102.4\mu s)$
		101 (fper iph/32)	$fc/2^9 (12.8\mu s)$	$fc/2^{11} (51.2\mu s)$	$fc/2^{13} (204.8\mu s)$
1 (fc)	000 (fc)	000 (fper iph/1)	$fc/2^1 (0.05\mu s)$	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$
		001 (fper iph/2)	$fc/2^2 (0.1\mu s)$	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		100 (fper iph/16)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		101 (fper iph/32)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$
	100 (fc/2)	000 (fper iph/1)	—	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$
		001 (fper iph/2)	$fc/2^2 (0.1\mu s)$	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		100 (fper iph/16)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		101 (fper iph/32)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$
	101 (fc/4)	000 (fper iph/1)	—	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$
		001 (fper iph/2)	—	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.2\mu s)$	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		100 (fper iph/16)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		101 (fper iph/32)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$
	110 (fc/8)	000 (fper iph/1)	—	—	$fc/2^5 (0.8\mu s)$
		001 (fper iph/2)	—	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$
		010 (fper iph/4)	—	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.4\mu s)$	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$
		100 (fper iph/16)	$fc/2^5 (0.8\mu s)$	$fc/2^7 (3.2\mu s)$	$fc/2^9 (12.8\mu s)$
		101 (fper iph/32)	$fc/2^6 (1.6\mu s)$	$fc/2^8 (6.4\mu s)$	$fc/2^{10} (25.6\mu s)$

- (注1) プリスケーラ出カクロック ϕTn は、かならず $\phi Tn < f_{sys}$ を満足するように (ϕTn が f_{sys} よりも遅くなるように) 選択してください。
- (注2) タイマ動作中はクロックギアの切り替えは行わないでください。
- (注3) 表中 “—” は設定禁止です。

表10-5 プリスケアラ出カクロック分解能 (fc=32MHz)

ペリフェラル クロック解除 <FPSEL>	クロックギア値 <GEAR[2:0]>	プリスケアラ クロック選択 <PRCK[2:0]>	プリスケアラ出カクロック分解能		
			$\phi T1$	$\phi T4$	$\phi T16$
0 (fgear)	000 (fc)	000 (fper iph/1)	$fc/2^1 (0.0625\mu s)$	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$
		001 (fper iph/2)	$fc/2^2 (0.125\mu s)$	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		100 (fper iph/16)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		101 (fper iph/32)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$
	100 (fc/2)	000 (fper iph/1)	$fc/2^2 (0.125\mu s)$	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$
		001 (fper iph/2)	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$
		010 (fper iph/4)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		011 (fper iph/8)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		100 (fper iph/16)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$
		101 (fper iph/32)	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$	$fc/2^{11} (64.0\mu s)$
	101 (fc/4)	000 (fper iph/1)	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$
		001 (fper iph/2)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		010 (fper iph/4)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		011 (fper iph/8)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$
		100 (fper iph/16)	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$	$fc/2^{11} (64.0\mu s)$
		101 (fper iph/32)	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$	$fc/2^{12} (128.0\mu s)$
	110 (fc/8)	000 (fper iph/1)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		001 (fper iph/2)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		010 (fper iph/4)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$
		011 (fper iph/8)	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$	$fc/2^{11} (64.0\mu s)$
		100 (fper iph/16)	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$	$fc/2^{12} (128.0\mu s)$
		101 (fper iph/32)	$fc/2^9 (16.0\mu s)$	$fc/2^{11} (64.0\mu s)$	$fc/2^{13} (256.0\mu s)$
1 (fc)	000 (fc)	000 (fper iph/1)	$fc/2^1 (0.0625\mu s)$	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$
		001 (fper iph/2)	$fc/2^2 (0.125\mu s)$	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		100 (fper iph/16)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		101 (fper iph/32)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$
	100 (fc/2)	000 (fper iph/1)	—	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$
		001 (fper iph/2)	$fc/2^2 (0.125\mu s)$	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		100 (fper iph/16)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		101 (fper iph/32)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$
	101 (fc/4)	000 (fper iph/1)	—	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$
		001 (fper iph/2)	—	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$
		010 (fper iph/4)	$fc/2^3 (0.25\mu s)$	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		100 (fper iph/16)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		101 (fper iph/32)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$
	110 (fc/8)	000 (fper iph/1)	—	—	$fc/2^5 (1.0\mu s)$
		001 (fper iph/2)	—	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$
		010 (fper iph/4)	—	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$
		011 (fper iph/8)	$fc/2^4 (0.5\mu s)$	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$
		100 (fper iph/16)	$fc/2^5 (1.0\mu s)$	$fc/2^7 (4.0\mu s)$	$fc/2^9 (16.0\mu s)$
		101 (fper iph/32)	$fc/2^6 (2.0\mu s)$	$fc/2^8 (8.0\mu s)$	$fc/2^{10} (32.0\mu s)$

- (注1) プリスケーラ出カクロック ϕT_n は、かならず $\phi T_n < f_{sys}$ を満足するように (ϕT_n が f_{sys} よりも遅くなるように) 選択してください。
- (注2) タイマ動作中はクロックギアの切り替えは行わないでください。
- (注3) 表中 “—” は設定禁止です。

10.5.2 アップカウンタ(UC0)

16 ビットのバイナリカウンタです。

- ソースクロック

ソースクロックは TBOMOD<TBOCLK[1:0]>で設定することができます。

プリスケアラ出力クロック ϕ T1、 ϕ T4、 ϕ T16、または、TB0IN0 端子の外部クロックのいずれかを選択できます。

- カウンタ動作の開始と停止

カウンタ動作は TBORUN<TBORUN>で行います。<TBORUN> = “1” でカウントを開始し、“0” でカウント停止と同時にカウンタのクリアを行います。

- カウンタクリアのタイミング

- ①コンペアー一致時

TBOMOD<TBOCLE> = “1” に設定することで、TBORG1 とのコンペアー一致とともにカウンタのクリアをすることができます。TBOMOD<TBOCLE> = “0” に設定するとカウンタはフリーランニングカウンタとして動作します。

- ②カウンタ停止時

TBORUN<TBORUN> = “0” に設定すると、カウンタが停止するとともにクリアされます。

- カウンタのオーバーフロー

UC0 がオーバーフローすると、オーバーフロー割り込み INTTB0 が発生します。

10.5.3 タイマレジスタ (TBORG0、TBORG1)

アップカウンタ UC0 と比較する値を設定するレジスタで、2 本内蔵されています。タイマレジスタに設定された値とアップカウンタの値をコンパレータで比較し、一致するとコンパレータが一致検出信号を出力します。

- 構成

タイマレジスタのうち、TBORG0/1 はダブルバッファ構成になっており、レジスタバッファとペアになっています。ダブルバッファのイネーブル/ディセーブルを制御は TBOCR<TBOWBF> によって行います。<TBOWBF> = “0” のときディセーブル、<TBOWBF> = “1” のときイネーブルとなります。ダブルバッファイネーブル時、UC0 と TBORG1 との一致時にレジスタバッファ 0 からタイマレジスタ TBORG0/1 へデータ転送が行われます。

- 初期状態

リセット動作により、TBORG0、TBORG1 は不定で、ダブルバッファはディセーブルになっています。

- 設定方法

- ①ダブルバッファを使用しない場合

タイマレジスタへのデータ設定は、2 バイトデータ転送命令を用いるか、1 バイトデータ転送命令を 2 回用いて下位 8 ビット、上位 8 ビットの順に行います。

- ②ダブルバッファを使用する場合

TBORG0/1 とレジスタバッファは、同じアドレスに割り付けられています。

<TBOWBF> = “0” のときは、TBORG0/1 とそれぞれのレジスタバッファに、同じ値が書き込まれ、<TBOWBF> = “1” のときは、それぞれのレジスタバッファにのみ書き込まれます。したがって、タイマレジスタに初期値を書き込むときには、レジスタバッファをディセーブルにしておき、タイマレジスタにデータを書き込んだ後、<TBOWBF> = “1” に設定し、レジスタバッファへ次のデータを書き込んでください。

10.5.4 キャプチャ制御

アップカウンタ UC0 の値をキャプチャレジスタ TBOCP0、TBOCP1 にラッチするタイミングを制御する回路です。キャプチャレジスタのラッチタイミングは、TBOMOD<TBOCPM[1:0]>で設定します。

また、ソフトウェアによってもアップカウンタ UC0 の値をキャプチャレジスタへ取り込むことができ、TBOMOD<TBOCP0> に “0” を書き込むたびに、その時点の UC0 の値をキャプチャレジスタ TBOCP0 へキャプチャします。

10.5.5 キャプチャレジスタ (TB0CP0H/L, TB0CP1H/L)

アップカウンタ UC0 の値をラッチする 16 ビットのレジスタです。

10.5.6 アップカウンタキャプチャレジスタ (TB0UCH/L)

キャプチャ制御回路によるキャプチャ機能のほかに、TB0UC レジスタを読み出すことにより、アップカウンタ UC0 の現在のカウント値をキャプチャする事ができます。

10.5.7 コンパレータ (CP0、CP1)

アップカウンタ UC0 と、タイマレジスタ TBORG0、TBORG1 への設定値とを比較し、一致を検出する 16 ビットコンパレータです。一致すると、INTTB0 を発生します。

10.5.8 タイマフリップフロップ (TB0FF0)

タイマフリップフロップ (TB0FF0) は、コンパレータからの一致信号、キャプチャレジスタへのラッチ信号によって反転するフリップフロップです。反転のディセーブル/イネーブルは、TB0FFCR<TB0C1T1, TB0C0T1, TB0E1T1, TB0E0T1>によって設定できます。

リセット後、TB0FF0 の値は不定となります。TB0FFCR<TB0FF0C1 : 0>に “00” を書き込むことで反転、“01” を書き込むことで “1” にセット、“10” を書き込むことで “0” にクリアすることが可能です。

TB0FF0 の値は、タイマ出力端子 TB0OUT 端子 (PIO と兼用) へ出力することができます。タイマ出力を行う場合、あらかじめポート I 関連レジスタ PICR、PIFR1 により、設定を行う必要があります。

10.5.9 キャプチャ割込み (INTCAP00、INTCAP01)

キャプチャレジスタ TB0CP0、TB0CP1 にラッチするタイミングで割込み INTCAP00、INTCAP01 をそれぞれ発生します。割込みの設定は CPU で行います。

10.6 モード別動作説明

10.6.1 16ビットインタバルタイマモード

一定周期の割り込みを発生させる場合

タイマレジスタ TBORG1 にインタバル時間を設定し、INTTB0 割り込みを発生します。

10.6.2 16ビットイベントカウンタモード

入力クロックを外部クロック (TB0IN0 端子入力) にすることでイベントカウンタにすることができます。

アップカウンタは TB0IN0 端子入力の立ち上がりエッジでカウントアップします。ソフトウェアキャプチャを行い、キャプチャ値をリードすることでカウント値を読むことができます。

イベントカウンタとして使用する場合も、プリスケアラは “RUN” にしてください (TBORUN<TBOPRUN> = “1”)。

10.6.3 16ビットPPG (プログラマブル矩形波) 出力モード

任意周波数、任意デューティの矩形波 (プログラマブル矩形波) を出力することができます。出力パルスは、ローアクティブ、ハイアクティブどちらでも可能です。

アップカウンタ (UC0) とタイマレジスタ (TBORG0、TBORG1) への設定値との一致によりタイマフリップフロップ (TBOFF) の反転トリガをかけることで、プログラマブル矩形波を TB0OUT 端子より出力することができます。ただし、TBORG0 と TBORG1 の設定値は次の条件を満たす必要があります。

$$(TBORG0 \text{ への設定値}) < (TBORG1 \text{ への設定値})$$

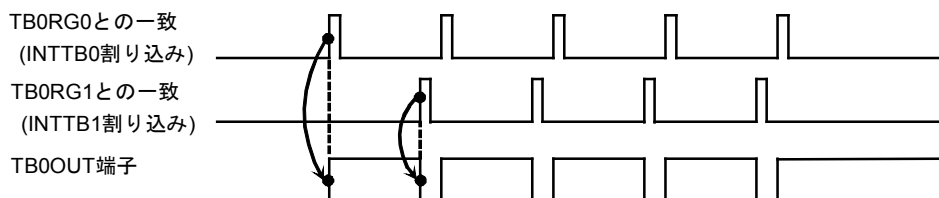


図 10-2 プログラマブル矩形波 (PPG) 出力波形例

このモードでは、TBORG0 のダブルバッファをイネーブルにすることにより、TBORG1 との一致で、レジスタバッファ 0 の値が TBORG0 へシフトインされます。これにより、小さいデューティへの対応が容易に行えます。

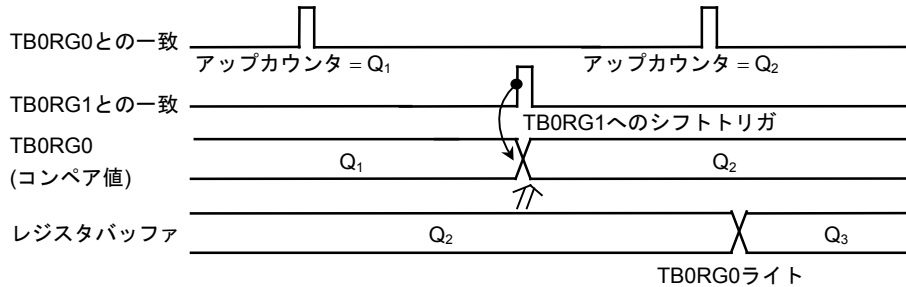


図 10-3 レジスタバッファの動作

このモードのブロック図を示します。

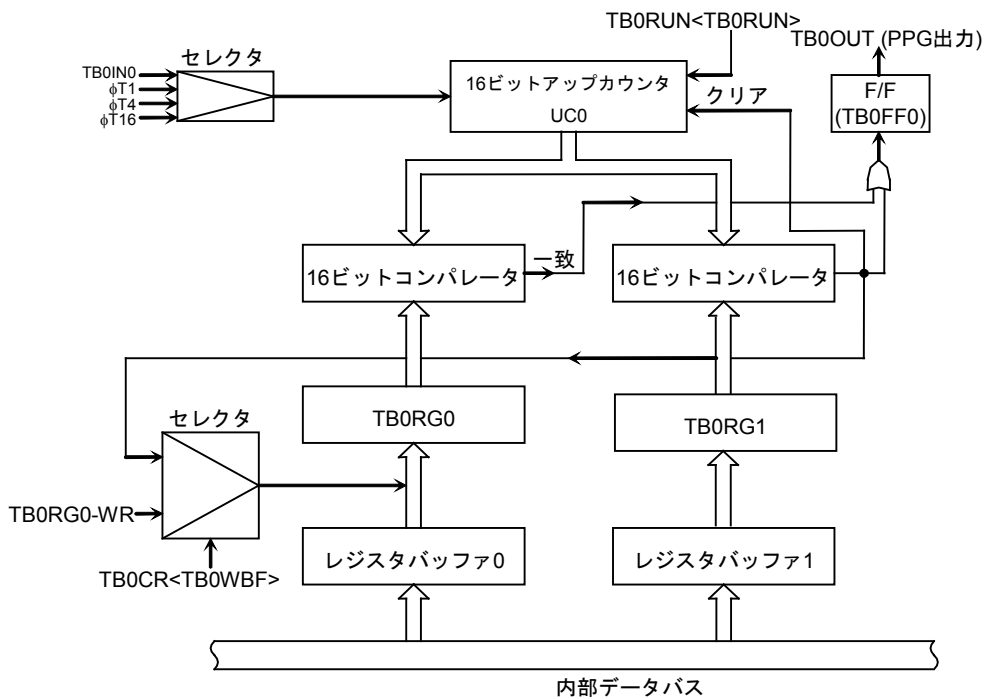


図 10-4 16 ビット PPG モードのブロック図

16ビット PPG 出力モード時の各レジスタは、次のように設定します。

		7	6	5	4	3	2	1	0	
TBOEN	←	1	X	X	X	X	X	X	X	TMRBO モジュールを起動します。
TBORUN	←	X	X	X	X	X	0	X	0	TMRBO を停止します。
TBORG0	←	*	*	*	*	*	*	*	*	デューティを設定します。
	←	*	*	*	*	*	*	*	*	(16ビット ※レジスタは32ビット)
TBORG1	←	*	*	*	*	*	*	*	*	周期を設定します。(16ビット※レジスタは32ビット)
	←	*	*	*	*	*	*	*	*	
TBOCR	←	1	0	X	0	0	0	0	0	TBORG0 のダブルバッファインーブル (INTTBO 割り込みでデューティ/周期の変更)
TBOFFCR	←	X	X	0	0	1	1	1	0	TBOFF0 を TBORG0、TBORG1 との一致検出で反転する ように設定します。また、TBOFF0 の初期値を “0” に します。
TBOMOD	←	0	0	1	0	0	1	*	*	入力クロックをプリスケアラ出力クロックにし、 キャプチャ機能ディセーブルにします。
								**	**	
										(** = 01, 10, 11)
PICR	←	-	-	-	-	-	-	-	1	} PIO を TBOOUT に割り付けます。
PIFR1	←	-	-	-	-	-	-	-	1	
TBORUN	←	*	*	*	*	*	1	X	1	

X: Don' t care -; no change

10.7 タイマ同期モード

タイマ同期モードを使用する事により、タイマ間のスタートの同期を取る事が可能となります。
PPG 出力にて同期モードを使用することによりモータ等の駆動に応用が可能です。

TBnCR<TBnSYNC>により、同期モードの切り替えを行います。

<TBnSYNC> = “0” : タイマの ch 毎のタイミングで動作を行います。

<TBnSYNC> = “1” : 同期出力を行います。

TMRB0~3, TMRB4~7 の 2つのブロックに分かれております。

<TBnSYNC> = “1” を設定した場合、ch 毎のタイマスタート TBmRUN<TBmPRUN, TBmRUN>=” 1, 1”
ではスタートせず、TMRB0, TMRB4 に同期する形でスタート致します。

- (注 1) 同期出力したい ch に関しては、TMRB0, TMRB4 でスタートする前に
TBmRUN<TBmPRUN, TBmRUN>=” 1, 1” にて予め同時にスタートをかけておいてください。
- (注 2) 同期出力モード時以外は TBnCR<TBnSYNC>= “0” に設定してください。
同期出力モードが設定されている場合、TMRB0, 4, 8 にてスタートが掛かるまで、
他の ch スタートは待たされます。
- (注 3) タイマ同期のマスターとなる TMRB0, TMRB4 は、常に TBnSYNC ビットを” 0” に
設定してください。
- (注 4) TMRB8, TMRB9 はタイマ同期モードの使用はできません。

TBnCR (0x4001_0xx8)		7	6	5	4	3	2	1	0
	bit Symbol	TBnWBF		TBnSYNC		I2TBn			
	Read/Write	R/W	R/W	R/W	R	R/W	R	R	R
	リセット後	0	0	0	0	0	0	0	0
	機能	Double Buffer 0: 禁止 1: 許可	“0” をラ イトして ください。	同期モード 0: 個別動作	リードす ると“0”が 読めます	IDLE 0: 停止 1: 動作	リードす ると“0”が 読めます	リードす ると“0”が 読めます	リードす ると“0”が 読めます

スレーブ側となるタイマは、TBnSYNC ビットを” 1” に設定してください。

TBnCR (0x4001_0xx8)		7	6	5	4	3	2	1	0
	bit Symbol	TBnWBF		TBnSYNC		I2TBn			
	Read/Write	R/W	R/W	R/W	R	R/W	R	R	R
	リセット後	0	0	0	0	0	0	0	0
	機能	Double Buffer 0: 禁止 1: 許可	“0” をラ イトして ください。	同期モード 1: 同期動作	リードす ると“0”が 読めます	IDLE 0: 停止 1: 動作	リードす ると“0”が 読めます	リードす ると“0”が 読めます	リードす ると“0”が 読めます

10.8 キャプチャ機能を利用した応用例

キャプチャ機能を利用することにより次に示す例をはじめ、多くの応用が可能です。

- ① 外部トリガパルスからのワンショットパルス出力
- ② 周波数測定
- ③ パルス幅測定
- ④ 時間差測定

①外部トリガパルスからのワンショットパルス出力

外部トリガパルスからのワンショットパルス出力は、次のように行います。

16ビットアップカウンタ UC5 をプリスケアラ出力クロックにてフリーランニングでカウントアップさせておきます。TB5IN0 端子より外部トリガパルスを入力し、キャプチャ機能を用いて、外部トリガパルスの立ち上がりで、アップカウンタ値をキャプチャレジスタ (TB5CP0) に取り込みます。

外部トリガパルスの立ち上がり時、割り込み INTCAP50 が発生するように CPU で設定します。この割り込みで、タイマレジスタ (TB5RG0) には、TB5CP0 の値 (c) とディレイタイム (d) を加算した値 (c + d) を設定します。

タイマレジスタ (TB5RG1) には、TB5RG0 の値とワンショットパルスのパルス幅 (p) を加算した値 (c + d + p) を設定します。(TB5RG1 の変更は次の一致までに完了してください)

さらに、タイマフリップフロップコントロールレジスタ (TB5FFCR<TB5E1T1, TB5E0T1>) に “11” を設定し、UC5 と TB5RG0 との一致、および、TB5RG1 との一致により、タイマフリップフロップ (TB5FF0) が反転するように、トリガイネーブルにします。ワンショットパルス出力後、INTTB5 の割り込み処理により、これをディセーブルに戻します。

なお、文中の (c)、(d)、(p) は、「図 10-5 ワンショットパルス出力 (ディレイあり)」の c、d、p と対応しています。

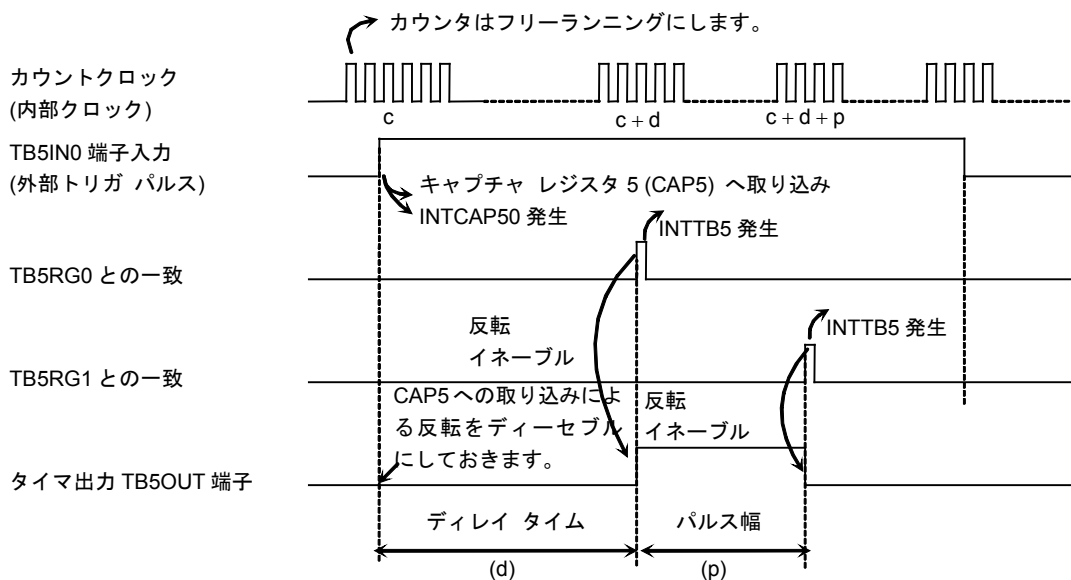


図 10-5 ワンショットパルス出力 (ディレイあり)

ディレイが不要な場合、TB5CP0 への取り込みによって TB5FF0 を反転させ、割り込み INT で TB5CP0 の値 (c) にワンショットパルスの幅 (p) を加算した値 (c + p) を TB5RG1 に設定します。(TB5RG1 の変更は次の一致までに完了してください。)TB5FF0 は、TB5RG1 と UC5 の一致によって反転するように、反転イネーブルを選択します。また、INTTB5 割り込みでこれをディセーブルに戻します。

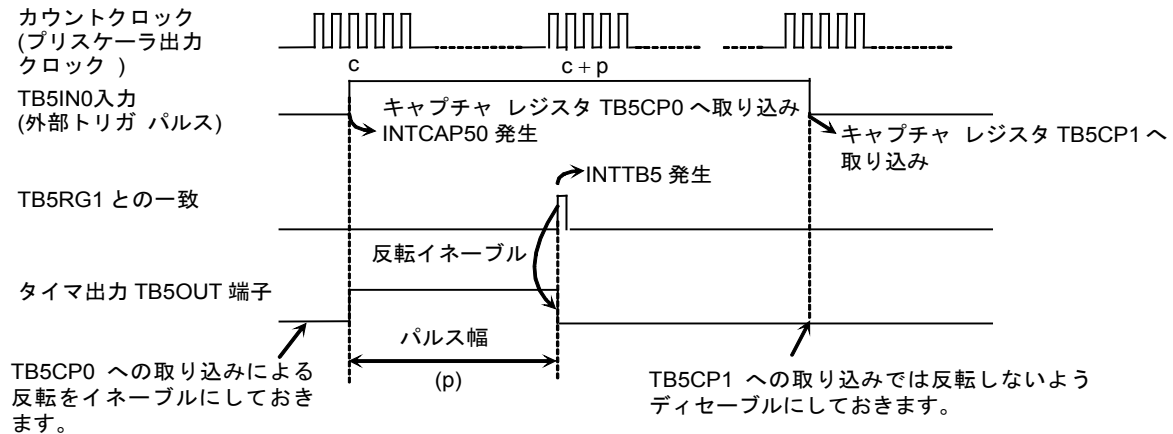


図 10-6 外部トリガパルスのワンショットパルス出力 (ディレイなし)

② 周波数測定

キャプチャ機能を用いて外部クロックの周波数測定を行うことができます。

周波数測定は、16 ビットイベントカウンタモードと他の 16 ビットタイマを組み合わせで行います。TMRB3 と TMRB8 を使う場合を例に説明します。TMRB8 の TB8OUT を測定時間の設定に用います。

TMRB3 のカウントクロックは TB3IN0 端子入力を選択し、外部クロック入力によるカウント動作を行います。TB3MOD<TB3CPM[1:0]> には “11” を設定します。この設定により、16 ビットタイマ (TMRB8) のタイマフリップフロップ出力 (TB8OUT) の立ち上がりで、キャプチャレジスタ (TB3CP0) に 16 ビットアップカウンタ UC3 のカウンタ値を取り込み、16 ビットタイマ (TMRB8) の TB8OUT の立ち下がりで、キャプチャレジスタ (TB3CP1) に UC3 のカウンタ値の取り込みを行います。

周波数は、16 ビットタイマの割り込み INTTB8 で測定時間を基準にして TB3CP0、TB3CP1 の差より求めます。

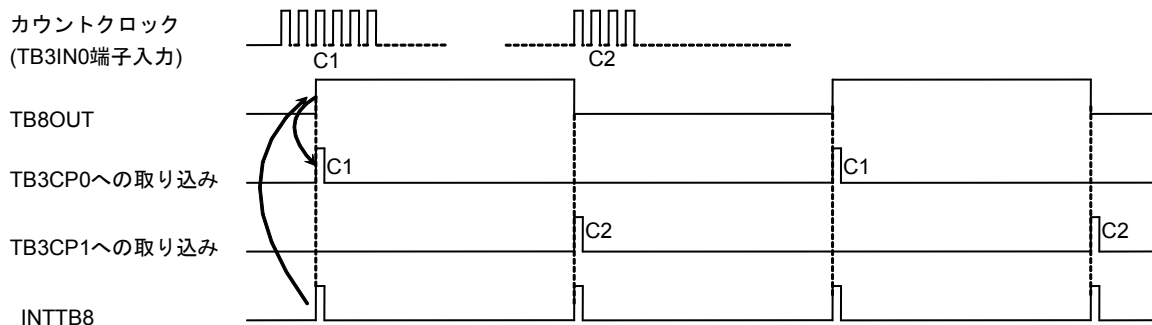


図 10-7 周波数測定

例えば、16ビットタイマによる TBOFF の “1” レベル幅の設定値が 0.5 s で、TB3CP0 と TB3CP1 の差が 100 であれば、周波数は $100 \div 0.5 \text{ s} = 200 \text{ Hz}$ となります。

③ パルス幅測定

キャプチャ機能を用いて、外部パルスの “H” レベル幅を測定することができます。TB5IN0 端子より外部パルスを入力し、アップカウンタ (UC5) をプリスケアラ出力クロックにてフリーランニングでカウントアップさせておきます。キャプチャ機能を用いて、外部パルスの立ち上がり/立ち下がり、それぞれのエッジでトリガをかけ、このときのアップカウンタ値をキャプチャレジスタ (TB5CP0, TB5CP1) に取り込みます。TB5IN0 端子の立ち下がりにより、INTCAP51 が発生するように CPU で設定します。

“H” レベルパルス幅は、TB5CP0 と TB5CP1 の差を求め、その値に内部クロックの周期をかけることにより、求めることができます。

例えば TB5CP0 と TB5CP1 の差が 100 で、プリスケアラ出力クロックの周期が $0.5 \mu\text{s}$ であれば、パルス幅は、 $100 \times 0.5 \mu\text{s} = 50 \mu\text{s}$ となります。

なお、クロックソースにより定まる UC5 の最大カウント時間を越えるパルス幅の測定を行う場合は、注意が必要です。この場合、ソフトウェアによる処理を行ってください。

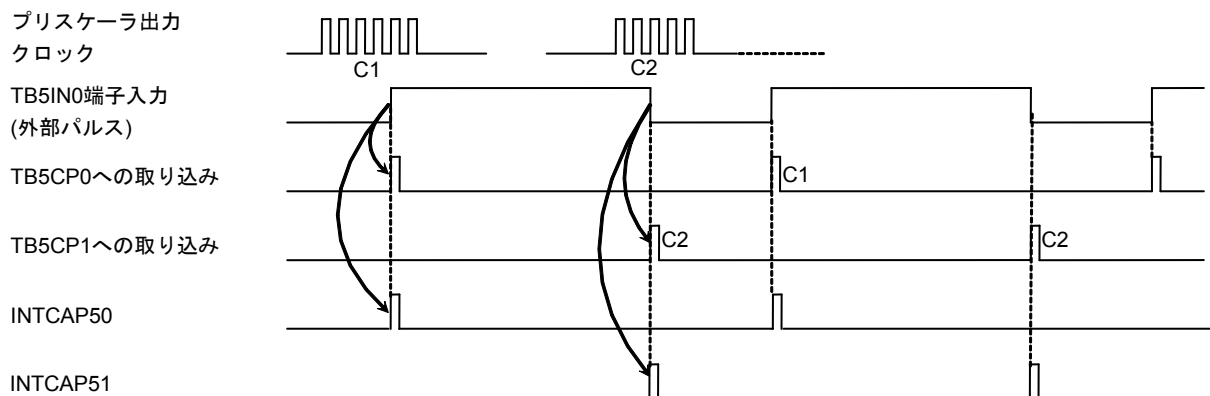


図 10-8 パルス幅測定

また、外部パルスの “L” レベル幅を測定することもできます。この場合、

「図 10-8 パルス幅測定」における、2 回目の INTCAP50 割り込み処理により、1 回目の C2 と 2 回目の C1 の差に、プリスケアラ出力クロックの周期をかけることにより、求めることができます。

④ 時間差測定

キャプチャ機能を用いて、2つの事象の時間差を測定することができます。プリスケアラ出力クロックを用いて、アップカウンタ（UC5）をフリーランニングでカウントアップさせておきます。TB5IN0 端子の入力パルスの立ち上がりエッジで、UC5 の値をキャプチャレジスタ（TB5CP0）に取り込みます。このとき、割り込み INTCAP50 が発生するように CPU で設定します。

TB5IN1 端子の入力パルスの立ち上がりエッジで、UC5 の値をキャプチャレジスタ TB5CP1 に取り込みます。このとき、割り込み INTCAP51 が発生するように CPU で設定します。

時間差は、キャプチャレジスタの値が取り込み終わった時点で、TB5CP1 から TB5CP0 を引いた値に、内部クロックの周期をかけて求めることができます。

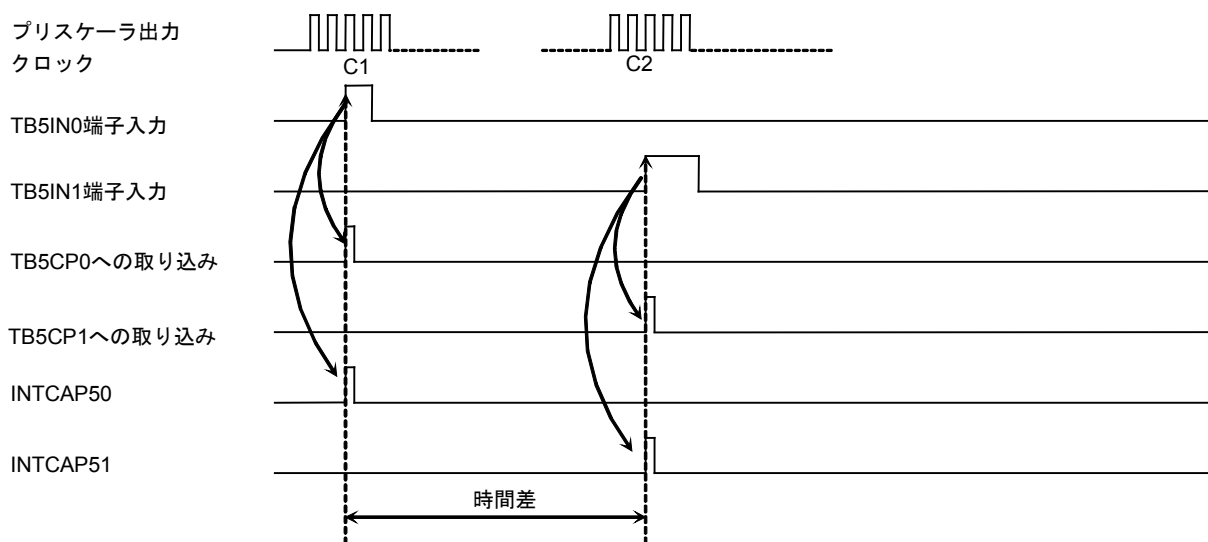


図 10-9 時間差測定

11 シリアル チャネル (SIO/UART)

11.1 特長

シリアル入出力を3チャンネル内蔵 (SI00~SI02) しています。各チャンネルは、下記に示すように UART モード (非同期通信) および I/O インターフェースモード (同期通信) を選択できます。

I/O インターフェースモード ———— モード 0: I/O を拡張するための I/O データの送受信とその同期信号 (SCLK) の送受信を行うモード

非同期通信 (UART) モード ————

- モード 1: 送受信データ長 7 ビット
- モード 2: 送受信データ長 8 ビット
- モード 3: 送受信データ長 9 ビット

このうち、モード 1 とモード 2 は、パリティビットの付加が可能で、モード 3 はマスタコントローラがシリアル リンク (マルチ コントローラ システム) でスレーブ コントローラを起動させるためのウェイクアップ機能を有しています。図 11-2 に、SI00 のブロック図を示します。

各チャンネルは主に、プリスケラ、シリアルクロック生成回路、受信バッファとその制御回路、送信バッファとその制御回路で構成されています。各チャンネルは、それぞれ独立に動作します。

SI00~SI02 は同一の動作をしますので、SI00 の場合についてのみ説明します。

表 11-1 SIO のチャンネル別仕様相違点

		チャンネル 0		チャンネル 1		チャンネル 2	
端子名称		TXD0 (PE0) RXD0 (PE1) CTS0/SCLK0 (PE2)		TXD1 (PE4) RXD1 (PE5) CTS1/SCLK1 (PE6)		TXD2 (PF0) RXD2 (PF1) CTS1/SCLK1 (PF2)	
割込み		INTRX0 INTTX0		INTRX1 INTTX1		INTRX2 INTTX2	
レジスタ名 (アドレス)	イネーブルレジスタ	SC0EN	0x4002_0080	SC1EN	0x4002_00C0	SC2EN	0x4002_0100
	送受信バッファレジスタ	SC0BUF	0x4002_0084	SC1BUF	0x4002_00C4	SC2BUF	0x4002_0104
	コントロールレジスタ	SC0CR	0x4002_0088	SC1CR	0x4002_00C8	SC2CR	0x4002_0108
	モードコントロール レジスタ 0	SC0MOD0	0x4002_008C	SC1MOD0	0x4002_00CC	SC2MOD0	0x4002_010C
	ポーレートジェネレータ コントロール	SC0BRCR	0x4002_0090	SC1BRCR	0x4002_00D0	SC2BRCR	0x4002_0110
	ポーレートジェネレータ コントロール 2	SC0BRADD	0x4002_0094	SC1BRADD	0x4002_00D4	SC2BRADD	0x4002_0114
	モードコントロール レジスタ 1	SC0MOD1	0x4002_0098	SC1MOD1	0x4002_00D8	SC2MOD1	0x4002_0118
	モードコントロール レジスタ 2	SC0MOD2	0x4002_009C	SC1MOD2	0x4002_00DC	SC2MOD2	0x4002_011C
	受信 FIFO コンフィグレジスタ	SC0RFC	0x4002_00A0	SC1RFC	0x4002_00E0	SC2RFC	0x4002_0120
	送信 FIFO コンフィグレジスタ	SC0TFC	0x4002_00A4	SC1TFC	0x4002_00E4	SC2TFC	0x4002_0124
	受信 FIFO ステータスレジスタ	SC0RST	0x4002_00A8	SC1RST	0x4002_00E8	SC2RST	0x4002_0128
	送信 FIFO ステータスレジスタ	SC0TST	0x4002_00AC	SC1TST	0x4002_00EC	SC2TST	0x4002_012C
	FIFO コンフィグレジスタ	SC0FCNF	0x4002_00B0	SC1FCNF	0x4002_00F0	SC2FCNF	0x4002_0130

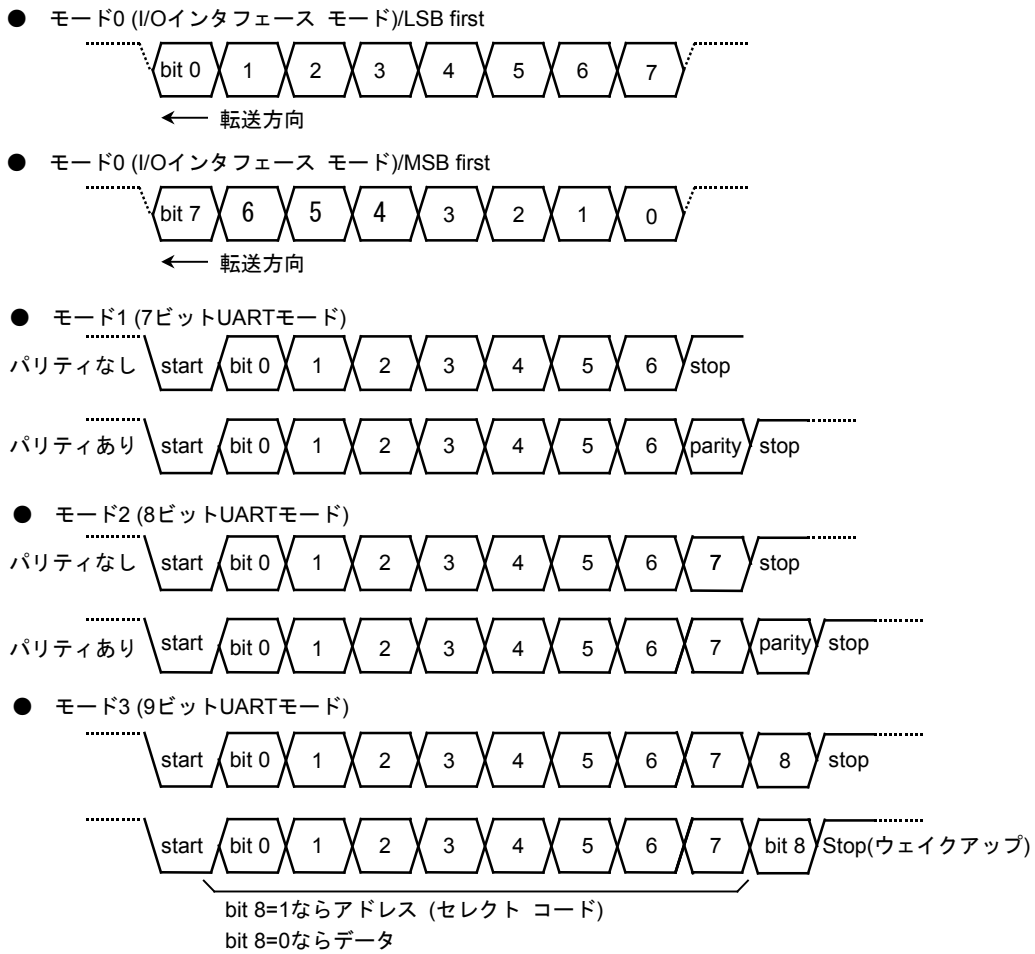


図11-1 データフォーマット

11.2 ブロック図 (チャンネル0)

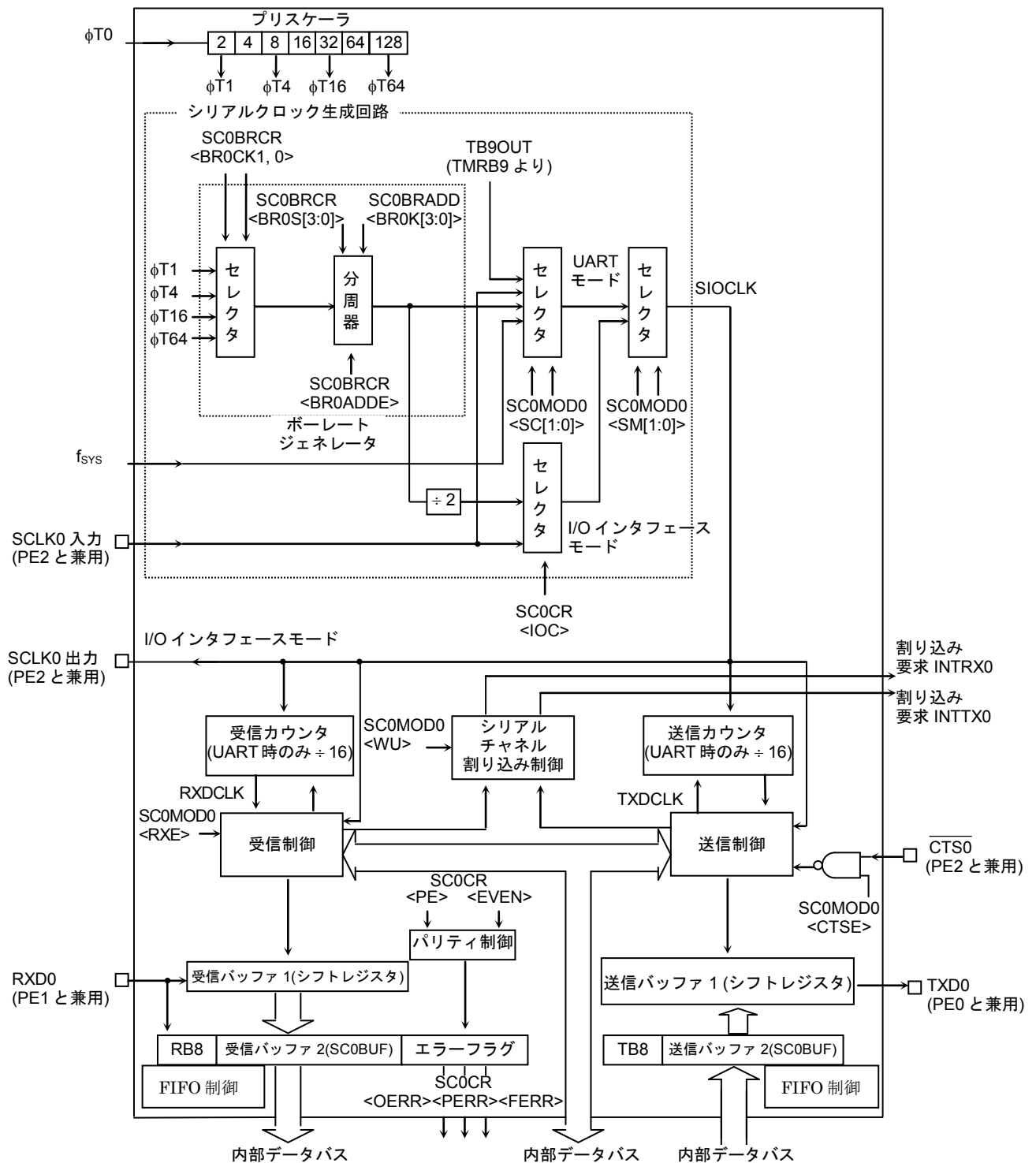


図11-2 SIO0 ブロック図

11.3 回路別の動作説明（チャンネル0）

11.3.1 プリスケータ

SI00の動作クロックを生成するために、7ビットプリスケータがあります。プリスケータの入力クロック $\phi T0$ は、CG部のCGSYSCR<PRCK[2:0]>にて選択したfperiph/1、fperiph/2、fperiph/4、fperiph/8、fperiph/16、fperiph/32のいずれかのクロックです。

fperiphはCG部のCGSYSCR<FPSEL>で選択されるクロックfgearまたはクロックギアで分周される前のクロックfcのいずれかです。

プリスケータは、シリアル転送クロックに、ボーレートジェネレータを選択した場合にのみ動作します。プリスケータ出力クロックの分解能を、表 11-2、表 11-3に示します。

表 11-2 ボーレートジェネレータへの入力クロック分解能 $f_c = 40\text{MHz}$

ペリフェラル クロック解除 <FPSEL>	クロックギア値 <GEAR[2:0]>	プリスケラ クロック選択 <PRCK[2:0]>	プリスケラ出力クロック分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
0 (fgear)	000 (fc)	000 (fper iph/1)	$f_c/2^1 (0.05\mu\text{s})$	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$
		001 (fper iph/2)	$f_c/2^2 (0.1\mu\text{s})$	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$
	100 (fc/2)	000 (fper iph/1)	$f_c/2^2 (0.1\mu\text{s})$	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$
		001 (fper iph/2)	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$
		010 (fper iph/4)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		011 (fper iph/8)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		100 (fper iph/16)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$
		101 (fper iph/32)	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$	$f_c/2^{13} (204.8\mu\text{s})$
	101 (fc/4)	000 (fper iph/1)	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$
		001 (fper iph/2)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		010 (fper iph/4)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		011 (fper iph/8)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$
		100 (fper iph/16)	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$	$f_c/2^{13} (204.8\mu\text{s})$
		101 (fper iph/32)	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$	$f_c/2^{14} (409.6\mu\text{s})$
	110 (fc/8)	000 (fper iph/1)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		001 (fper iph/2)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		010 (fper iph/4)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$
		011 (fper iph/8)	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$	$f_c/2^{13} (204.8\mu\text{s})$
		100 (fper iph/16)	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$	$f_c/2^{14} (409.6\mu\text{s})$
		101 (fper iph/32)	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$	$f_c/2^{13} (204.8\mu\text{s})$	$f_c/2^{15} (819.2\mu\text{s})$
1 (fc)	000 (fc)	000 (fper iph/1)	$f_c/2^1 (0.05\mu\text{s})$	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$
		001 (fper iph/2)	$f_c/2^2 (0.1\mu\text{s})$	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$
	100 (fc/2)	000 (fper iph/1)	—	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$
		001 (fper iph/2)	$f_c/2^2 (0.1\mu\text{s})$	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$
	101 (fc/4)	000 (fper iph/1)	—	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$
		001 (fper iph/2)	—	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.2\mu\text{s})$	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$
	110 (fc/8)	000 (fper iph/1)	—	—	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$
		001 (fper iph/2)	—	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$
		010 (fper iph/4)	—	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.4\mu\text{s})$	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (0.8\mu\text{s})$	$f_c/2^7 (3.2\mu\text{s})$	$f_c/2^9 (12.8\mu\text{s})$	$f_c/2^{11} (51.2\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (1.6\mu\text{s})$	$f_c/2^8 (6.4\mu\text{s})$	$f_c/2^{10} (25.6\mu\text{s})$	$f_c/2^{12} (102.4\mu\text{s})$

- (注1) プリスケーラ出力クロック ϕT_n は、かならず $\phi T_n < f_{sys}$ を満足するように (ϕT_n が f_{sys} よりも遅くなるように) 選択してください。
- (注2) SIO 動作中はクロックギアの切り替えは行わないでください。
- (注3) 表中 “-” は設定禁止です。

シリアルインタフェースボーレートジェネレータには、プリスケーラ出力クロックより $\phi T1$ 、 $\phi T4$ 、 $\phi T16$ 、 $\phi T64$ の4種類のクロックが用いられます。

表 11-3 ボーレートジェネレータへの入力クロック分解能 $f_c = 32\text{MHz}$

ペリフェラル クロック解除 <FPSEL>	クロックギア値 <GEAR[2:0]>	プリスケアラ クロック選択 <PRCK[2:0]>	プリスケアラ出力クロック分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
0 (fgear)	000 (fc)	000 (fper iph/1)	$f_c/2^1 (0.0625\mu\text{s})$	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$
		001 (fper iph/2)	$f_c/2^2 (0.125\mu\text{s})$	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$
	100 (fc/2)	000 (fper iph/1)	$f_c/2^1 (0.0625\mu\text{s})$	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$
		001 (fper iph/2)	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$
		010 (fper iph/4)	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$	$f_c/2^{13} (256.0\mu\text{s})$
	101 (fc/4)	000 (fper iph/1)	$f_c/2^1 (0.0625\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$
		001 (fper iph/2)	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		010 (fper iph/4)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$	$f_c/2^{13} (256.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$	$f_c/2^{14} (512.0\mu\text{s})$
	110 (fc/8)	000 (fper iph/1)	$f_c/2^1 (0.0625\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		001 (fper iph/2)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		010 (fper iph/4)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$	$f_c/2^{13} (256.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$	$f_c/2^{14} (512.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$	$f_c/2^{13} (256.0\mu\text{s})$	$f_c/2^{15} (1024\mu\text{s})$
1 (fc)	000 (fc)	000 (fper iph/1)	$f_c/2^1 (0.0625\mu\text{s})$	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$
		001 (fper iph/2)	$f_c/2^2 (0.125\mu\text{s})$	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$
	100 (fc/2)	000 (fper iph/1)	—	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$
		001 (fper iph/2)	$f_c/2^2 (0.125\mu\text{s})$	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$
	101 (fc/4)	000 (fper iph/1)	—	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$
		001 (fper iph/2)	—	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$
		010 (fper iph/4)	$f_c/2^3 (0.25\mu\text{s})$	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$
	110 (fc/8)	000 (fper iph/1)	—	—	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$
		001 (fper iph/2)	—	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$
		010 (fper iph/4)	—	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$
		011 (fper iph/8)	$f_c/2^4 (0.5\mu\text{s})$	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$
		100 (fper iph/16)	$f_c/2^5 (1.0\mu\text{s})$	$f_c/2^7 (4.0\mu\text{s})$	$f_c/2^9 (16.0\mu\text{s})$	$f_c/2^{11} (64.0\mu\text{s})$
		101 (fper iph/32)	$f_c/2^6 (2.0\mu\text{s})$	$f_c/2^8 (8.0\mu\text{s})$	$f_c/2^{10} (32.0\mu\text{s})$	$f_c/2^{12} (128.0\mu\text{s})$

- (注1) プリスケラ出力クロック ϕT_n は、かならず $\phi T_n < f_{sys}$ を満足するように (ϕT_n が f_{sys} よりも遅くなるように) 選択してください。
- (注2) S10 動作中はクロックギアの切り替えは行わないでください。
- (注3) 表中 “—” は設定禁止です。

シリアルインタフェースポーレートジェネレータには、プリスケラ出力クロックより $\phi T1$ 、 $\phi T4$ 、 $\phi T16$ 、 $\phi T64$ の4種類のクロックが用いられます。

11.3.2 ボーレート

ボーレートジェネレータは、シリアルチャネルの転送速度を定める送受信クロックを生成するための回路です。

ボーレートジェネレータへの入力クロックは7ビットプリスケアラより、 $\phi T1$ 、 $\phi T4$ 、 $\phi T16$ 、 $\phi T64$ を用います。この入力クロックの選択はボーレートジェネレータコントロールレジスタ SCOBRCR <BROCK[1:0]>で設定します。

ボーレートジェネレータは、 $1 + N + \frac{m}{16}$ ($N=2\sim 15$ 、 $m=0\sim 15$)、16分周が可能な分周器を内蔵しており、ボーレートジェネレータコントロールレジスタ SCOBRCR<BROADDE><BROS[3:0]>、SCOBRADE<BROK[3:0]>の設定に従い分周を行い、転送速度を決定します。

• UART モードの場合

1) SCOBRCR<BROADDE>=0 の場合

SCOBRADE<BROK[3:0]>の設定は無視され、SCOBRCR<BROS[3:0]>に設定された値“N”に従いN分周を行います。(N=1、2、3 … 16)

2) SCOBRCR<BROADDE>=1 の場合

$N + (16 - K) / 16$ 分周機能がイネーブルになり SCOBRCR<BROS[3:0]>に設定された値“N” (N = 2、3 … 15)、SCOBRADE<BROK[3:0]>に設定された値“K”に従い $N + (16 - K) / 16$ 分周を行います。(K=1、2、3 … 15)

(注) N=1 および 16 のときは $N + (16 - K) / 16$ 分周機能は禁止となりますので必ず SCOBRCR<BROADDE>= “0” に設定してください。

• I/O インタフェースモードの場合

I/O インタフェースモード時は $N + (16 - K) / 16$ 分周機能は使用できません。必ず SCOBRCR<BROADDE>= “0” に設定してN分周を行ってください。

• ボーレートジェネレータを使用した場合のボーレートの算出方法

1) UART モード

$$\text{ボーレート} = \frac{\text{ボーレートジェネレータの入力クロック}}{\text{ボーレートジェネレータの分周値}} \div 16$$

ボーレートジェネレータで生成される最高のボーレートは、 $\phi T1$ が 20MHz のときの 1.25Mbps です。

ボーレートジェネレータの出力ではない f_{sys} をシリアルクロックとして使用できます。この場合の最高ボーレートは 2.5Mbps@ $f_{\text{sys}} = 40\text{MHz}$

2) I/O インタフェースモード

$$\text{ボーレート} = \frac{\text{ボーレートジェネレータの入力クロック}}{\text{ボーレートジェネレータの分周値}} \div 2$$

ボーレートジェネレータで生成される最高のボーレートは、 $\phi T1$ が 20 MHz のときで、ダブルバッファを使用する場合は、ボーレートジェネレータの分周値=“1”の設定が可能なので、10Mbps になります (ダブルバッファを使用しない場合は、ボーレートジェネレータの分周値=“2”のときの、5.0Mbps が最高になります)。

- ボーレートの設定例

- 1) 整数分周 (N分周) の場合

fperiph に fc=39.321 MHz を選択し、φT0 を fperiph/16 に設定したとき、ボーレートジェネレータへの入力クロックをφT1、分周値 “N” (SCOBRCR<BROS[3:0]>)=4、SCOBRCR<BROADDE>= “0” の場合の UART モードのボーレートは、

※ クロック条件

システムクロック	:	高速 (fc)
高速クロックギア	:	1倍 (fc)
プリスケアラクロック	:	$f_{\text{periph}}/16$ ($f_{\text{periph}} = f_{\text{sys}}$)

$$\text{ボーレート} = \frac{fc/32}{4} \div 16$$

$$= 39.321 \times 10^6 \div 32 \div 4 \div 16 \doteq 19200 \text{ (bps) となります。}$$

(注) +(16-K)/16 分周機能は禁止に設定されるため SCOBRA DD <BROK[3:0]> の設定は無視されます。

- 2) N + (16-K)/16 分周 (UART モードのみ) の場合

fperiph に fc=9.6MHz を選択し、φT0 を fperiph/8 に設定したとき、ボーレートジェネレータへの入力クロックをφT1、分周値 “N” (SCOBRCR<BROS[3:0]>)=7、K” (SCOBRA DD <BROK[3:0]>)=3、SCOBRCR<BROADDE>=1 の場合のボーレートは、

※ クロック条件

システムクロック	:	高速 (fc)
高速クロックギア	:	1倍 (fc)
プリスケアラクロック	:	$f_{\text{periph}}/4$ ($f_{\text{periph}} = f_{\text{sys}}$)

$$\text{ボーレート} = \frac{fc/16}{7 + \frac{(16-3)}{16}} \div 16$$

$$= 9.6 \times 10^6 \div 16 \div \left(7 + \frac{13}{16}\right) \div 16 = 4800 \text{ (bps) となります。}$$

また、外部クロック入力をシリアルクロックに使用することもできます。この場合のボーレートの算出方法を示します。

- 外部クロック入力を使用した場合のボーレートの算出方法

- 1) UART モード

ボーレート = 外部クロック入力 ÷ 16

ただし、(外部クロック入力周期) > 2/fsys を満足する必要があります。

fsys=40MHz のときの最高ボーレートは、 $40 \div 2 \div 16 = 1.25$ (Mbps) になります。

- 2) I/O インタフェースモード

ボーレート = 外部クロック入力

ダブルバッファを使用するときは、

(外部クロック入力周期) > 6/fsys を満足する必要があります。

従って、fsys=40MHz のときの最高ボーレートは、 $40 \div 6 = 6.66$ (Mbps) 未満にする必要があります。

ダブルバッファを使用しないときは、

(外部クロック入力周期) > 8/fsys を満足する必要があります。

従って、fsys=40MHz のときの最高ボーレートは、 $40 \div 8 = 5.0$ (Mbps) 未満にする必要があります。

表 11-4～表 11-5に UART モードのボーレートの例を示します。

表 11-4 UART ボーレートの選択
(ボーレートジェネレータ使用、SCOBRCR <BROADE> = 0 の場合) 単位 (kbps)

f _c [MHz]	入力クロック				
	分周値 N (SCOBRCR <BROS[3:0]>に設定)	φT1 (f _c /4)	φT4 (f _c /16)	φT16 (f _c /64)	φT64 (f _c /256)
9.830400	2	76.800	19.200	4.800	1.200
↑	4	38.400	9.600	2.400	0.600
↑	8	19.200	4.800	1.200	0.300
↑	0	9.600	2.400	0.600	0.150

(注) 本表は、システムクロックとして f_c、クロックギアとして f_c/1、プリスケラ用クロックとして f_{periph}/2 を選択した場合の値です。

表 11-5 UART ボーレートの選択
(タイマ TMRB9 の出力 (内部 TB9OUT) 使用, タイマ TMRB9 の入力クロックが φT0 の場合)
単位 (kbps)

TBORG	f _c	32 MHz	9.8304 MHz	8 MHz
	1H		250	76.8
2H		125	38.4	31.25
3H			25.6	
4H		62.5	19.2	15.625
5H		50	15.36	12.5
6H			12.8	
8H		31.25	9.6	
AH		25	7.68	6.25
10H		15.625	4.8	
14H		12.5	3.84	3.125

ボーレートの算出方法 (タイマ TMRB9 を使用した場合)

$$\text{転送レート} = \frac{\text{CGSYSCR<PRCK[1:0]>で選択されたクロック周波数}}{(\text{TBORG} \times 2) \times 2 \times 16}$$

↑ (タイマ TMRB9 の入力クロックが φT0 の場合)
↑ (タイマフリップフロップ反転 2 回で 1 クロック周期となる)

- (注1) I/O インタフェースモードでは、タイマ TMRB9 からの出力信号 (内部) を転送クロックとして使用できません。
- (注2) 本表は、システムクロックとして f_c、クロックギアとして f_c/1、プリスケラ用クロックとして f_{periph}/4 を選択した場合の値です。

11.3.3 シリアルクロック生成回路

送受信基本クロックを生成する回路です。

- I/O インターフェースモードの場合

シリアルコントロールレジスタ SCOCR<IOC>= “0” の SCLK 出力モードのときは、前記ボーレートジェネレータの出力を2分周して、基本クロックをつくります。

SCOCR<IOC>= “1” の SCLK 入力モードのときは、SCOCR<SCLKS>の設定に従って立ち上がり/立ち下がりエッジを検出し、基本クロックをつくります。

- 非同期通信 (UART) モードの場合

シリアルモードコントロールレジスタ SCOMOD0<SC[1:0]>の設定により、前記ボーレートジェネレータからのクロックか、システムクロック (f_{SYS}) か、タイマ (TMRB9) の内部出力信号か、または外部クロック (SCLK0 端子)のいずれかを選択し、基本クロック SIOCLK を生成します。

11.3.4 受信カウンタ

受信カウンタは、非同期通信 (UART) モードで用いられる4ビットのバイナリカウンタで、SIOCLKでカウントアップされます。データ1ビットの受信にSIOCLKが16クロック用いられ7、8、9発目でデータをサンプリングします。3度のデータサンプリングによる多数決論理により受信データを判断しています。

11.3.5 受信制御部

- I/O インターフェースモードの場合

SCOCR<IOC>= “0” の SCLK 出力モードのときは、SCLK0 端子へ出力されるシフトクロックの立ち上がりで RXD0 端子をサンプリングします。

SCOCR <IOC>=1” の SCLK 入力モードのときは、SCOCR<SCLKS>の設定に従って、SCLK 入力の立ち上がり/立ち下がりエッジでシリアル受信データ RXD0 端子をサンプリングします。

- 非同期通信 (UART) モードの場合

受信制御部は、スタートビット検出回路を持ち、正常なスタートビットと判断して受信動作を開始します。

11.3.6 受信バッファ

受信バッファは、オーバランエラーを防ぐため二重構造となっています。受信バッファ1 (シフトレジスタ型) に受信データが1ビットずつ格納され、データがそろそろともう一方の受信バッファ2 (SCOBUF) へ移されます。また、同時に受信バッファの full Flag (SCOMOD2<RBFL>) が “1” にセットされ、受信バッファ2に有効データが格納されていることを示します。但し受信FIFOが有効にされている場合は、受信データは受信FIFOに移されこのFlagは直ぐにクリアされます。

受信FIFOが有効にされていない場合 (SCOFCNF<CNFG>=0 で且つ SCOMOD1<FDPX[1:0]>=01) は同時に割り込み INTRX0 が発生します。受信FIFOが有効にされている場合 (SCOFCNF<CNFG>=1 で且つ SCOMOD1<FDPX[1:0]>=01/11) は SCORFC<RIL[1:0]>の設定に従い割り込みが発生します。

CPUは受信バッファ2 (SCOBUF)または受信FIFO(アドレスは受信バッファに同じ)の方を読み出します。受信FIFOが有効にされていない場合は、この読み出しにより受信バッファのfull Flag <RBFLL>は“0”にクリアされます。CPUが受信バッファ2 (SCOBUF)または受信FIFOを読み出す前でも、次の受信データは受信バッファ1へ格納することができます。

また、IOインターフェースモードでSCLK出力に設定されている場合は、ダブルバッファ制御ビットSCOMOD2<WBUF>の設定により、受信バッファ2 (SCOBUF)を許可または不許可にできます。

受信バッファ2 (ダブルバッファ)を不許可にして、更に受信FIFOを有効にしない事(SCOFCNF<CNFG>=0で且つ<FDPX[1:0]>=01))により、通信相手とハンドシェイクを取ることが可能となり、1フレーム転送後にSCLK出力を停止します。この設定の場合には、CPUからの読み出し動作は受信バッファ1に対して行われます。読み出し動作が行なわれる事により、SCLK出力を再開します。

受信バッファ2 (ダブルバッファ)を許可にした場合で、受信FIFOが有効にされていない場合は最初の受信データが受信バッファ1から2に移され、次に受信データの受信が終了して、受信バッファ2及び1に有効データが存在した状態になると、SCLK出力を停止します。受信バッファ2の読み出しが行なわれると、受信バッファ1のデータが受信バッファ2に移され、受信割り込みINTRX0が発生すると同時にSCLK出力を再開します。従って、IOインターフェースモードのSCLK出力ではダブルバッファ制御ビットSCOMOD2<WBUF>の設定に関わらずオーバーランエラーは発生しません。

受信バッファ2 (ダブルバッファ)を許可にした場合で受信FIFOが有効にされている場合(SCNFCNF<CNFG>=1で且つ<FDPX[1:0]>=01/11)は受信FIFOがFULL (SCOFCNF<RFST>の設定に従う)になり、受信バッファ2及び1に有効データが存在した状態になるとSCLK出力を停止します。またこの状態でSCOFCNF<RXTXCNT>=1にしているとSCLK出力停止と同時に受信制御BIT:RXEが自動的にクリアされます。“0”の場合は自動クリアが行なわれません。

(注) このモードでは、SCOCRの<OEER> Flagは意味を持ちません。動作定義は不定となります。従ってSCLK出力モードから、他のモードへ切り替える時は事前にSCOCRをリードしてこのFlagを初期化してください。

その他の動作モードでは常に受信バッファ2の動作は有効であり、連続転送時のパフォーマンス向上を可能にしています。受信FIFOが有効にされていない場合は、受信バッファ1に次のデータが全ビット受信される前に受信バッファ2 (SCOBUF)を読み出さなければオーバーランエラーとなります。オーバーランエラーが発生した場合、受信バッファ2およびSCOCR<RB8>の内容は保存されていますが、受信バッファ1の内容は失われます。受信FIFOが有効にされている場合は受信FIFOがFULLになり、受信バッファ2に次のデータが移される前に受信FIFOを読み出さないと、受信FIFOのオーバーランが発生して受信FIFOのオーバーランエラーフラグがセットされます。この場合でも受信FIFOのデータは保存されます。

8ビットUARTのパリティ付加の場合、パリティビット9ビットUARTモードの場合の最上位ビットはSCOCR<RB8>に格納されます。

9ビットUARTの場合、ウェイクアップ機能SCOMOD0<WU>を“1”にすることによって、スレーブコントローラのウェイクアップ動作が可能で、SCOCR<RB8>=“1”のときのみ、割り込みINTRX0が発生します。

11.3.7 受信 FIFO バッファ

前記 Double buffer に加えて、受信 FIFO buffer を用いて、データの格納を行う事ができます。SC0FCNF レジスタの<CNFG>及び SCOMOD1 の<FDPX[1:0]>の設定で 4byte の受信 Buffer を有効にする事ができます。また、UART モード及び I/O インターフェースモードの場合は指定された fill レベルまでのデータが格納できます。受信 FIFO BUFFER を使用する場合はダブルバッファをイネーブルに設定しておいてください。

また UART モードでパリティビットを含むデータを受信する場合、受信ごとにパリティエラーチェックを行なう必要があります。

11.3.8 受信 FIFO の動作

① I/O インターフェースモードで SCLK 出力の場合

4Byte のデータを半二重受信する場合を例に説明します。

SIO の転送モードを半二重受信に設定します。

SCOMOD1<[6:5]>=01 : 転送モードを半二重受信に設定

SC0FCNF<[4:0]>=10111 : fill level 到達後の継続受信自動禁止

受信 FIFO の使用 Byte 数は割り込み発生 fill レベルに同じ

SCORFC<[1:0]>=00 : 割り込み発生を fill level 4 に設定

SCORFC<[7:6]>=01 : 受信 FIFO のクリアと割り込み発生条件の設定

この状態で転送モードを半二重受信に設定して RXE bit に 1 を書き込むと 4byte のデータ受信を開始します。4Byte 受信後に RXE を自動クリアして受信を終了(SCLK を STOP)します。

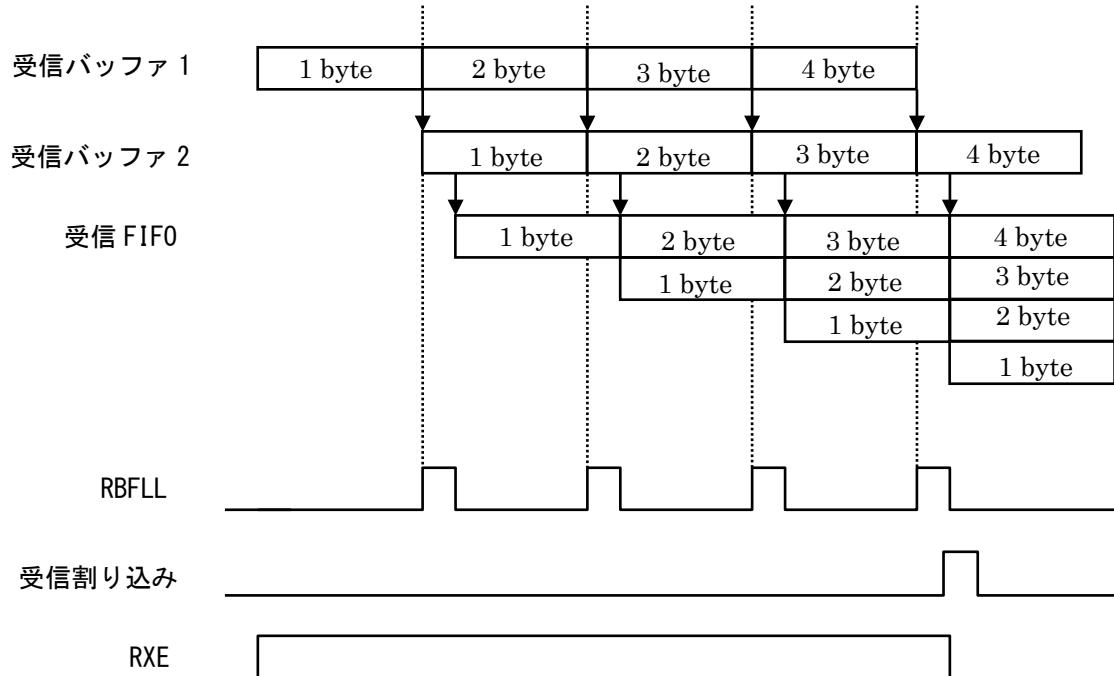


図11-3 受信 FIFO の動作

② I/O インターフェースモードで SCLK 入力の場合

SIO の転送モードを半二重受信に設定します。

SCOMOD1<[6:5]>=01 : 転送モードを半二重受信に設定

SCOF CNF<[4:0]>=10101 : fill level 到達後の継続受信自動許可

受信 FIFO の使用 Byte 数は最大構成

SCORFC<[1:0]>= 00 : 割り込み発生を fill level 4 に設定

SCORFC<[7:6]>=10 : 受信 FIFO のクリアと割り込み発生条件の設定

この状態で RXE bit に 1 を書き込むと入力クロックに応じて 4byte のデータ受信を開始します。4byte 受信後に受信 FIFO 割り込みを発生します。

また、この設定では次の受信に備えることもできます。FIFO 内のデータを読み出すことで、引き続き入力クロックに応じてデータ受信が可能となります。

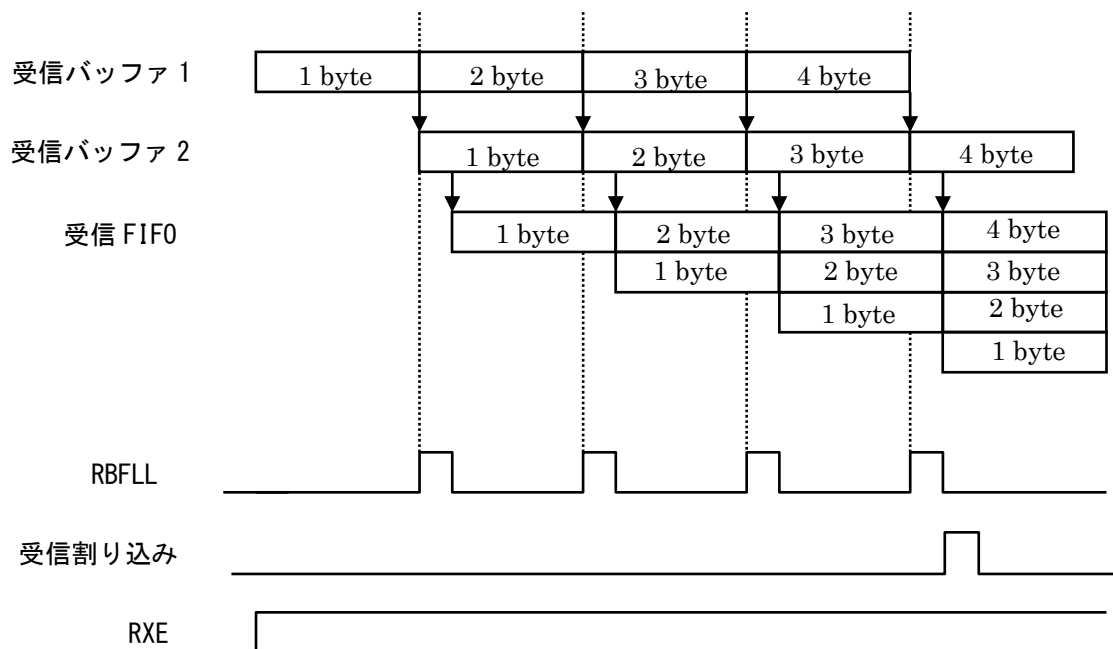


図 11-4 受信 FIFO の動作

11.3.9 送信カウンタ

送信カウンタは非同期通信 (UART) モードで用いられる 4 ビットのバイナリカウンタで受信カウンタ同様 SIOCLK でカウントされ、16 クロックごとに送信クロック (TXDCLK) を生成します。

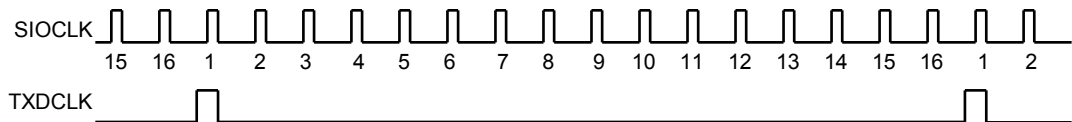


図 11-5 送信クロックの生成

11.3.10 送信制御部

- I/O インターフェースモードの場合

SC0CR<IOC>= “0” の SCLK 出力モードのときは、SCLK0 端子より出力されるシフトクロックの立ち上がりで送信バッファのデータを 1 ビットずつ TXD0 端子へ出力します。

SC0CR<IOC>= “1” の SCLK 入力モードのときは、SC0CR<SCLKS>の設定に従って SCLK 入力の立ち上がり/立ち下がりエッジで送信バッファのデータを 1 ビットずつシリアル送信データ TXD0 端子へ出力します。

- 非同期通信 (UART) モードの場合

送信バッファに CPU から送信データが書き込まれると、次の TXDCLK の立ち上がりエッジから送信を開始し、送信シフトクロック (TXDSFT) を生成します。

- ハンドシェイク機能

$\overline{\text{CTS}}$ 端子をもっており、この端子を使用することにより、1 データフォーマット単位での送信が可能となり、オーバランエラーの発生を防ぐことができます。この機能は SCOMOD0 <CTSE> によってイネーブル/ディセーブルできます。

送信は $\overline{\text{CTS}}$ 端子が “H” レベルになると、現在送信中のデータを送信完了後、 $\overline{\text{CTS}}$ 端子が “L” レベルに戻るまで送信を停止します。ただし、INTTX0 割り込みは発生し、次の送信データを CPU に要求し、送信バッファにデータを書き込み、送信待機します。

なお、 $\overline{\text{RTS}}$ 端子はありませんが、受信側にて受信が終了したとき（受信割り込みルーチン内）に $\overline{\text{RTS}}$ 機能に割り当てた任意の 1 ポートを “H” レベルにして、送信側に送信の一時停止を要求することにより容易にハンドシェイク機能を構築できます。

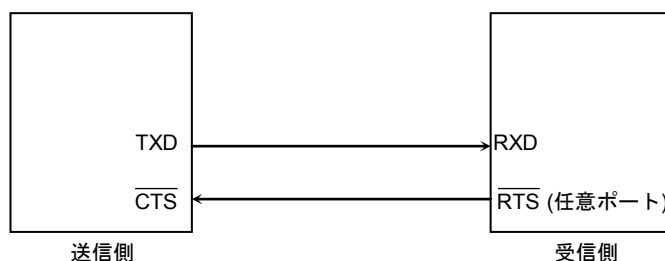


図 11-6 ハンドシェイク機能

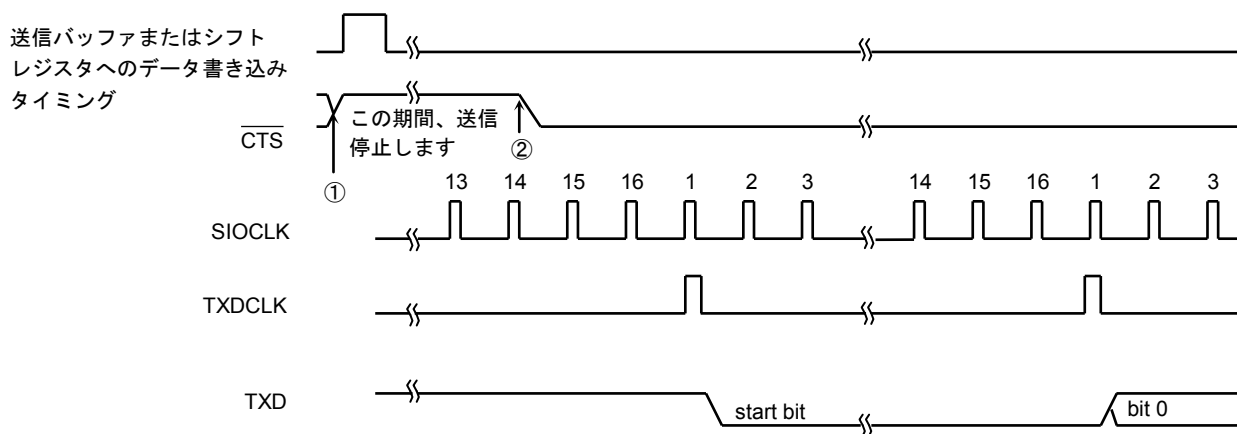


図 11-7 $\overline{\text{CTS}}$ (Clear to send) 信号のタイミング

- (注) ① 送信中に $\overline{\text{CTS}}$ 信号を立ち上げた場合は、送信終了後、次のデータの送信を停止します。
 ② $\overline{\text{CTS}}$ 信号立ち下がり後の最初の TXDCLK クロックから送信を開始します。

11.3.11 送信バッファ

送信バッファ (SCOBUF) は2重構造になっています。シリアルモードコントロールレジスタ 2 (SCOMOD2) のダブルバッファ制御ビット<WBUF>の設定により、ダブルバッファの有効/無効を制御できます。ダブルバッファが有効のときは、送信バッファ 2 (SCOBUF) ヘデータが書き込まれると、そのデータは送信バッファ 1 (シフトレジスタ) へ移されます。

送信 FIFO が有効にされていない場合 (SCOFCNF<CNFG>=0 or 1 で且つ SCOMOD1<FDPX[1:0]>=01) は同時に送信割り込み INTTXO が発生して、SCOMOD2 の送信バッファエンピティフラグ<TBEMP>が“1”にセットされます。これは、送信バッファ 2 が空になったことを示し、次の送信データの書き込みが可能になります。次の送信データが送信バッファ 2 に書き込まれると<TBEMP>フラグが“0”にクリアされます。

送信 FIFO が有効にされている場合 (SCNFCNF<CNFG>=1 で且つ SCOMOD1<FDPX[1:0]>=10/11) は送信 FIFO に送信データが存在する場合は直ぐに送信バッファ 2 へ移され、<TBEMP>フラグは直ぐに“0”にクリアされます。CPU は送信 Buffer 2 または送信 FIFO へデータを書き込みます。

また I/O インターフェースモードの SCLK 入力時に送信 FIFO が有効にされていない場合は、送信バッファ 1 のデータの送信が終了して次のフレームのクロックが入力される前に、送信バッファ 2 へデータがセットされないときは、アンダーランエラーになりシリアルコントロールレジスタ (SCOCR) のパリティ/アンダーランフラグ<PERR>がセットされます。

I/O インターフェースモードの SCLK 入力時で送信 FIFO が有効にされている場合は送信バッファ 1 のデータの送信が終了すると、送信バッファ 2 のデータが送信バッファ 1 へ移されると同時に送信 FIFO にデータが存在する場合は送信 FIFO から送信バッファ 2 へデータが移されます。

I/O インターフェースモードの SCLK 出力時に送信 FIFO が有効にされていない場合は、送信バッファ 2 のデータが送信バッファ 1 に移されて、そのデータの送信が終了すると SCLK 出力が停止します。従ってアンダーランエラーは発生しません。

I/O インターフェースモードの SCLK 出力時に送信 FIFO が有効にされている場合は送信 FIFO に有効データが無く、最後の送信バッファ 1 のデータ送信が終了すると SCLK 出力を停止します。

注) I/O インタフェース SCLK 出力モードでは、SCOCR の<PEER> Flag は意味を持ちません。動作定義は不定となります。従って SCLK 出力モードから、他のモードへ切り替える時は事前に SCOCR をリードしてこの Flag を初期化してください。

ダブルバッファを無効に設定している場合は、送信データの CPU からの書き込みは送信バッファ 1 に対して行われ、送信が終了すると送信割り込み INTTXO が発生します。

相手方とハンドシェイクの通信が必要な場合はダブルバッファ制御ビット<WBUF>="0" (不許可) に設定して送信バッファ 2 の機能を無効にして、送信 FIFO の設定は行わないで下さい。

11.3.12 送信 FIFO BUFFER

前記 Double buffer に加えて、送信 FIFO buffer を用いて、データの格納を行う事ができます。SCOFCNF レジスタの CNFG の設定及び SCOMOD1 の <FDPX[1:0]> で 4byte の送信 Buffer を有効にする事ができます。UART モード及び I/O インターフェースモードにおいて 4byte までのデータが格納できます。

また UART モードでパリティビットを含むデータを送信する場合、受信側は受信ごとにパリティエラーチェックを行なう必要があります。

注) 送信 FIFO バッファ使用時は、SIO の転送モード設定 (半二重/全二重)、FIFO 許可 (SCOFCNF<CNFG>=" 1") の後、必ず送信 FIFO クリアを実行して下さい。

11.3.13 送信 FIFO の動作

① I/O インターフェースモードで SCLK 出力の場合 (通常モード)

4Byte のデータを送信する場合を例に説明します。

SIO の転送モードを半二重受信に設定します。

SCOFCNF <[4:0]> = 01011 : fill level 到達後の継続送信の禁止

SCOTFC <[1:0]> = 00 : 割り込み発生時の fill level を 0 に設定

SCOTFC <[7:6]> = 01 : 送信 FIFO のクリアと割り込み発生条件の設定

この状態で転送モードを半二重送信に設定して、送信 FIFO に 4byte 分の送信データを書き込み、<TXE> BIT を 1 に設定する事によりデータ送信を開始します。最後の送信データが送信 BUFFER に移されると送信 FIFO 割り込みが発生して最後のデータの送信が終了するとクロック出力を停止して送信を終了します。

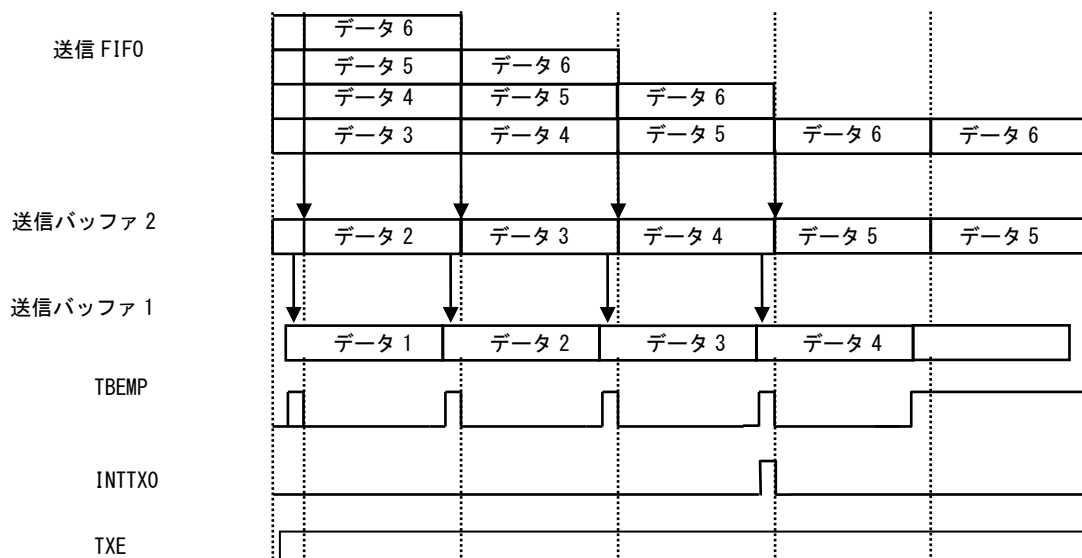


図 11-8 送信 FIFO の動作

② I/O インターフェースモードで SCLK 入力の場合（通常モード）

4Byte のデータを送信する場合を例に説明します。

SIO の転送モードを半二重受信に設定します。

SCOFCNF<[4:0]>=01001 : fill level 到達後の継続送信の許可

SCOTFC<[1:0]>=01 : 送信 FIFO のクリアと割り込み発生条件の設定

SCOTFC<[7:6]>=00 : 割り込み発生時の fill level を 0 に設定

この状態で、送信 FIFO に 4BYTE 分の送信データを書き込み、<TXE> BIT を 1 に設定する事によりクロック入力に応じてデータ送信を開始します。最後の送信データが送信 BUFFER に移されると送信 FIFO 割り込みが発生します。

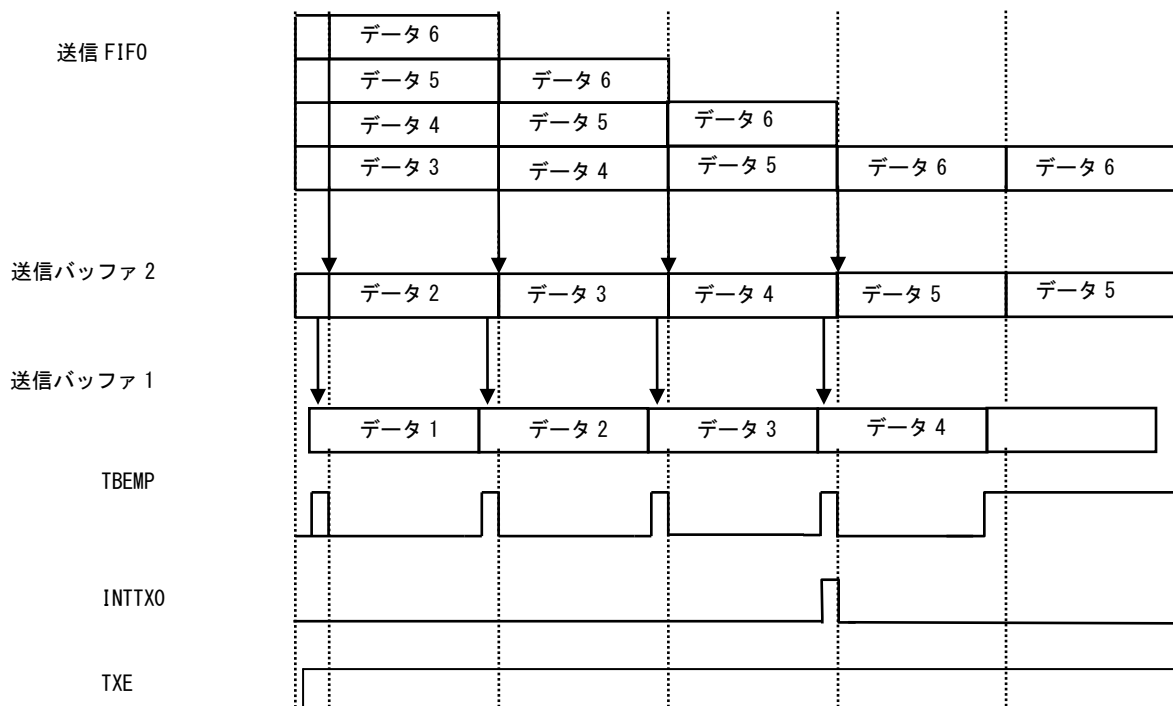


図 11-9 送信 FIFO の動作

11.3.14 パリティ制御回路

シリアルコントロールレジスタ (SCOCR) のパリティ付加ビット<PE>を“1”にするとパリティ付きの送信を行います。ただし、7ビットUARTまたは8ビットUARTモードのみパリティ付加が可能です。SCOCRの<EVEN>ビットによって偶数(奇数)パリティを選択することができます。

送信時、パリティ制御回路は送信バッファ(SCOBUF)に書き込まれたデータにより自動的にパリティを発生し、7ビットUARTモードのときはSCOBUFビット7<TB7>に、8ビットUARTモードのときはシリアルモードコントロールレジスタSCOMOD0のビット7<TB8>にパリティがデータ送信後に格納されます。なお、<PE>と<EVEN>の設定は、送信データを送信バッファに書き込む前に行ってください。

受信時、受信データが受信バッファ1にシフトインされ、受信バッファ2(SCOBUF)に移されることによりパリティを自動発生します。7ビットUARTモードのときは、SCOBUF<RB7>と、8ビットUARTモードのときは、SCOCRレジスタのビット7<RB8>のパリティと比較され、異なっているとパリティエラーが発生し、SCOCRレジスタの<PERR>フラグがセットされます。

I/Oインターフェースモードの時はSCOCR<PERR>はパリティフラグではなく、アンダーランエラーフラグの働きになります。

11.3.15 エラーフラグ

受信データの信頼性を上げるために3つのエラーフラグが用意されています。

1. オーバランエラー<OERR> : シリアルコントロールレジスタSCOCRのビット4

UART、I/Oインターフェースモード共に、受信バッファのデータを読み出す前に次のフレームの受信が終了するとエラーが発生して“1”にセットされます。受信FIFOを有効にしている場合は、受信FIFOへデータが自動的に移されるので、受信FIFOがFULL(使用BYTE数)になるまではオーバランエラーは発生しません。また、このフラグは読み出すと“0”にクリアされます。但し、I/OインターフェースモードのSCLK出力の設定ではオーバランエラーは発生しないために、このフラグは機能せず動作定義は未定です。

2. パリティエラー/アンダーランエラー<PERR> : SCOCRレジスタのビット3

UARTモード時はパリティエラーが発生すると“1”にセットされます。パリティエラーは受信したデータから生成されたパリティと受信したパリティが異なる場合に発生します。このフラグは読み出すと“0”にクリアされます。

I/Oインターフェースモード時はアンダーランエラーを示します。このフラグはシリアルモードコントロールレジスタ(SCOMOD2)のダブルバッファ制御ビット<WBUF>が“1”に設定されている状態で、SCLK入力モード時には、送信シフトレジスタのデータを送信終了し、次の転送クロックが入力される前に、送信ダブルバッファヘッダデータがセットされない場合に“1”にセットされ、アンダーランエラーが発生したことを示します。送信FIFOが有効にされている場合は送信FIFOからデータが移されます。送信FIFOのデータが空になり、送信ダブルバッファのデータ空になると、アンダーランエラーが発生します。SCLK出力モード時、アンダーランエラーは発生しないために、このフラグは機能せず動作定義は未定となります。送信バッファ2が無効の場合はアンダーランエラーフラグ<PERR>はセットされません。また、このフラグは読み出すと“0”にクリアされます。

3. フレーミングエラー <FERR> : SCOCR レジスタのビット 2

UART モード時にフレーミングエラーが発生すると“1”にセットされます。読み出すと“0”にクリアされます。フレーミングエラーは受信データのストップビットを中央付近でサンプリングし、結果が“0”の場合に発生します。シリアルモードコントロールレジスタ 2(SCOMOD2)の STOP ビット長設定ビット<SBLEN>の設定に関わらず、受信時の STOP ビットの判定は 1 ビットのみです。

動作モード	エラーフラグ	機能
UART	OERR	オーバーランエラーフラグ
	PERR	パリティエラーフラグ
	FERR	フレーミングエラーフラグ
IO インタフェース (SCLK 入力)	OERR	オーバーランエラーフラグ
	PERR	アンダーランエラーフラグ (WBUF=1)
		0 固定 (WBUF=0)
FERR	0 固定	
IO インタフェース (SCLK 出力)	OERR	動作未定義
	PERR	動作未定義
	FERR	0 固定

11.3.16 データ転送方向

シリアルモードコントロールレジスタ 2(SCOMOD2)の転送方向設定ビット<DRCHG>の設定により、I/O インターフェースモード時に転送方向を MSB ファーストまたは LSB ファーストを切り替えることができます。転送動作中の切り替えは行わないでください。

11.3.17 STOP ビットの長さ

SCOMOD2 レジスタのビット 4<SBLEN>の設定により UART 送信モード時の 送信データ中の STOP ビットの長さを 1 ビットまたは 2 ビットに設定できます。

11.3.18 ステータスフラグ

SCOMOD2 レジスタのビット 6<RBFLN>はダブルバッファ有効 (SCOMOD2<WBUF>=1) 時の受信バッファ full を示すフラグです。1 フレームの受信が終了して、受信データがバッファ 1 からバッファ 2 に移されるとバッファ 2 が full (データが格納されている状態)であることを示すために“1”にセットされます。CPU/DMAC により受信バッファを読み出すと“0”にクリアされます。<WBUF>=0 のときは意味を持ちませんのでステータスフラグとして使用しないでください。SCOMOD2 レジスタのビット 7<TBEMP>はダブルバッファ有効 (SCOMOD2<WBUF>=1) 時の送信バッファ 2 が空になったことを示すフラグです。送信バッファ 2 から送信バッファ 1 (シフトレジスタ) へデータが移されると、送信バッファ 2 が空になったことを示すために“1”がセットされます。CPU/DMAC により送信バッファにデータをセットすると“0”にクリアされます。<WBUF>=0 のときは意味を持ちませんのでステータスフラグとして使用しないでください。

11.3.19 送受信バッファの構成

		<WBUF> = 0	<WBUF> = 1
UART	送信	シングル	ダブル
	受信	ダブル	ダブル
I/O インタフェース (SCLK 入力)	送信	シングル	ダブル
	受信	ダブル	ダブル
I/O インタフェース (SCLK 出力)	送信	シングル	ダブル
	受信	シングル	ダブル

11.3.20 ソフトリセット

SCOMOD2 レジスタのビット 1, 0<SWRST[1:0]>を“10” → “01”と続けてのライトすることによりソフトウェアリセットが発生します。これにより、モードレジスタの SCOMOD0 <RXE>、SCOMOD1<TXE>、SCOMOD2<TBEMP>、<RBFL>、<TXRUN>、コントロールレジスタの SCOCR<OERR>、<PERR>、<FERR>及び内部回路が初期化されます。その他の状態は保持されます。

11.3.21 各信号発生タイミング

① UART モードの場合

受信

モード	9ビット	8ビット + パリティ	8ビット, 7ビット + パリティ, 7ビット
割り込み発生タイミング	第1ストップビットの中央付近	第1ストップビットの中央付近	第1ストップビットの中央付近
フレーミングエラー発生タイミング	ストップビットの中央付近	ストップビットの中央付近	ストップビットの中央付近
パリティエラー発生タイミング	—	最終ビット (パリティビット) の中央付近	最終ビット (パリティビット) の中央付近
オーバランエラー発生タイミング	ストップビットの中央付近	ストップビットの中央付近	ストップビットの中央付近

送信

モード	9ビット	8ビット + パリティ	8ビット, 7ビット + パリティ, 7ビット
割り込み発生タイミング (<WBUF> = 0)	ストップビット送出の直前	ストップビット送出の直前	ストップビット送出の直前
割り込み発生タイミング (<WBUF> = 1)	送信バッファ1ヘッダを移した直後 (スタートビット送出の直前)	送信バッファ1ヘッダを移した直後 (スタートビット送出の直前)	送信バッファ1ヘッダを移した直後 (スタートビット送出の直前)

② I/O インターフェースモードの場合

受信

割り込み発生タイミング (<WBUF> = 0)	SCLK 出力モード	最終 SCLK の立ち上がり直後
	SCLK 入力モード	最終 SCLK の立ち上がり直後 (立ち上がりモード)、立ち下がりモードでは立ち下がり直後
割り込み発生タイミング (<WBUF> = 1)	SCLK 出力モード	最終 SCLK の立ち上がり直後 (受信バッファ2ヘッダを移した直後) または受信バッファ2からデータを読み出した直後
	SCLK 入力モード	最終 SCLK の立ち上がり直後 (立ち上がりモード)、立ち下がりモードでは立ち下がり直後 (受信バッファ2ヘッダを移した直後)
オーバランエラー発生タイミング	SCLK 入力モード	最終 SCLK の立ち上がり直後 (立ち上がりモード)、立ち下がりモードでは立ち下がり直後

送信

割り込み発生タイミング (<WBUF> = 0)	SCLK 出力モード	最終 SCLK の立ち上がり直後
	SCLK 入力モード	最終 SCLK の立ち上がり直後 (立ち上がりモード)、立ち下がりモードでは立ち下がり直後
割り込み発生タイミング (<WBUF> = 1)	SCLK 出力モード	最終 SCLK の立ち上がり直後または送信バッファ1ヘッダを移した直後
	SCLK 入力モード	最終 SCLK の立ち上がり直後 (立ち上がりモード)、立ち下がりモードでは立ち下がり直後 または送信バッファ1ヘッダを移した直後
アンダーランエラー発生タイミング	SCLK 入力モード	次回 SCLK の立ち下がり直後 (立ち上がりモード)、立ち下がりモードでは立ち上がり直後

注1) 送信、受信 (受信許可の状態) に制御レジスタの変更は行わないでください

注2) 受信動作中に受信動作を停止 (SCOMOD0<RXE>="0") しないでください

注3) 送信動作中に送信動作を停止 (SCOMOD1<TXE>="0") しないでください

11.4 レジスタ説明（チャンネル0についてのみ説明します）

チャンネル0のレジスタについて説明します。他チャンネルについてもレジスタセットは同様です。

11.4.1 イネーブルレジスタ

	7	6	5	4	3	2	1	0	
SCOEN									
bit Symbol									SIOE
Read/Write									R/W
リセット後									0
機能	リードすると“0”が読めます。								SIO動作 0:禁止 1:許可

<SIOE>: SIOの動作を指定します。

SIOを使用する場合は、まずSIO動作許可にしてください。

動作禁止の状態では、イネーブルレジスタを除くSIOのすべてのクロックが停止しますので消費電力の低減が可能です。

SIOを一旦動作させた後に動作禁止にした場合は、SCOTFC[1:0]を除く各レジスタの設定は保持されます。

(注) $SCxEN<SIOE>=0$ (SIO動作禁止)、または $SCxMOD1<I2SC>=0$ でIDLEモードへの移行(IDLEモード中の動作停止)した場合は、必ず $SCxTFC$ の再設定を行ってください。

11.4.2 バッファレジスタ

バッファレジスタ (SC0BUF) は、書き込み時は送信バッファ、読み出し時は受信バッファとして機能します。

	7	6	5	4	3	2	1	0
SC0BUF								
bit Symbol	TB7/RB7	TB6/RB6	TB5/RB5	TB4/RB4	TB3/RB3	TB2/RB2	TB1/RB1	TB0/RB0
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	TB7~0 : 送信用バッファ/FIFO RB7~0 : 受信用バッファ/FIFO							

<TB[7:0]> 送信用バッファ。(書き込み時のみ)

<RB[7:0]> 受信用バッファ。(読み出し時のみ)

11.4.3 コントロールレジスタ

	7	6	5	4	3	2	1	0
bit Symbol	RB8	EVEN	PE	OERR	PERR	FERR	SCLKS	IOC
Read/Write	R	R/W		R (Readすると0クリアされます。)			R/W	
リセット後	0	0	0	0	0	0	0	0
機能	受信データ ビット8 (UART用)	パリティ (UART用) 0: Odd 1: Even	パリティ 付加 (UART用) 0: 禁止 1: 許可	0: 通常動作 1: エラー			0: SCLK0  1: SCLK0 	(I/O インタフェ-ス用) 0: ボーレートジェネレータ 1: SCLK0 端子入力
				オーバラン	パリティ / アンダーラン	フレーミング		

<RB8>: 9ビットUARTモード時の9ビット目の受信データです。

<EVEN>: パリティの条件を設定します。
“0”で奇数(Odd)パリティ、“1”で偶数(Even)パリティです。
パリティは、7ビットUARTモード、8ビットUARTモードで使用可能です。

<PE>: パリティ許可/禁止を制御するビットです。
パリティは、7ビットUARTモード、8ビットUARTモードで使用可能です。

<OERR>: エラーフラグです。(注)

<PERR>: それぞれ、オーバランエラー、パリティ/アンダーランエラー、フレーミングエラーを示します。

<SCLKS>: データ送受信で使用する入力クロックエッジを選択します。
0: SCLKxの立ち下がリエッジで送信バッファのデータを1bitずつTXDx端子へ出力します。SCLKxの立ち上がりエッジでRXDx端子のデータを1bitずつ受信バッファに取り込みます。この時、SCLKxはHighレベルからスタートします。
1: SCLKxの立ち上がりエッジで送信バッファのデータを1bitずつTXDx端子へ出力します。SCLKxの立ち下がリエッジでRXDx端子のデータを1bitずつ受信バッファに取り込みます。この時、SCLKxはLowレベルからスタートします。

<IOC>: I/Oインタフェースモード時のクロックを選択します。
“0”でボーレートジェネレータ、“1”でSCLK0端子入力を選択されます。

(注) エラーフラグは読み出されるとすべてクリアされます。

11.4.4 モードコントロールレジスタ 0

	7	6	5	4	3	2	1	0
bit Symbol	TB8	CTSE	RXE	WU	SM1	SM0	SC1	SC0
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	送信データビット 8	ハンドシェイク機能制御 0: CTS ディセーブル 1: CTS イネーブル	受信制御 0: 受信禁止 1: 受信許可	ウェイクアップ機能 0: ディセーブル 1: イネーブル	シリアル転送モード 00: I/O インタフェースモード 01: 7ビット長 UART モード 10: 8ビット長 UART モード 11: 9ビット長 UART モード	シリアル転送クロック (UART 用) 00: タイマ TB9OUT 01: ボーレートジェネレータ 10: 内部クロック f_{SYS} 11: 外部クロック (SCLKO 入力)		

<TB8>: 9ビット UART モード時の 9ビット目の送信データをセットします。

<CTSE>: ハンドシェイク機能の制御を行います。
イネーブルにすると \overline{CTS} 端子を用いたハンドシェイク機能が使用可能になります。

<RXE>: 受信制御を行うビットです。(注)
このビットは、各モードレジスタ (SCOMODO, SCOMOD1, SCOMOD2) を設定してから許可してください。

<WU>: ウェイクアップ機能の制御を行います。
9ビット UART モードの場合のみ有効で、その他のモードでは無視されます。

	9ビット UART モード	その他
0	受信すれば割り込み	don't care
1	受信データ 9ビット目が "1" のときのみ割り込み	

<SM[1:0]>: 転送モードを選択します。

<SC[1:0]>: UART モード時の転送クロックを選択します。
I/O インタフェースモード時の転送クロックは、コントロールレジスタ (SCOCR) で選択します。

(注) <RXE>="0" の状態で各モードレジスタ (SCOMODO、SCOMOD1、SCOMOD2) を設定してから最後に<RXE>="1" にしてください

11.4.5 モードコントロールレジスタ 1

	7	6	5	4	3	2	1	0
bit Symbol	I2S0	FDPX1	FDPX0	TXE	SINT2	SINT1	SINT0	—
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
リセット後	0	0	0	0	0	0	0	0
機能	IDLE 0: 停止 1: 動作	転送モード設定 00: 転送禁止 01: 半二重 (受信) 10: 半二重 (送信) 11: 全二重	送信制御 0: 禁止 1: 許可	連続転送時のインターバル時間 (I/O インタフェース用)		000: 無し 100: 8SCLK 001: 1SCLK 101: 16SCLK 010: 2SCLK 110: 32SCLK 011: 4SCLK 111: 64SCLK		0 を書き 込んでく ださい

<I2S0>: IDLE モード時の動作を指定します。

<FDPX[1:0]>: I/O インタフェースモード時の転送モードを設定します。また FIFO が許可されている場合は FIFO の構成を指定します。
UART モードの場合は FIFO 構成の指定のみ行われます。

<TXE>: 全転送モードに有効な送信許可ビットです。(注)
送信中にこのビットが禁止に設定された場合はそのフレームの転送が終了してから、禁止になります。

<SINT[2:0]>: I/O インタフェースモードで SCLK0 端子入力でない場合のみ有効なビットです。その他のモードでは意味を持ちません。
I/O インタフェースモードで、Double buffer または FIFO が許可されている時に連続転送のインターバル時間を指定します。

(注 1) <TXE>ビットは、他のビットを設定した後に許可してください。

(注 2) SCxEN<SIOE>=0 (SIO 動作禁止)、または SCxMOD1<I2SC>=0 で IDLE モードへの移行 (IDLE モード中の動作停止) した場合は、必ず SCxTFC の再設定を行ってください。

11.4.6 モードコントロールレジスタ 2

	7	6	5	4	3	2	1	0
bit Symbol	TBEMP	RBFLl	TXRUN	SBLEN	DRCHG	WBUF	SWRST1	SWRST0
Read/Write	R			R/W			W	
リセット後	1	0	0	0	0	0	0	0
機能	送信 buffer empty flag 0: full 1: Empty	受信 Buffer full flag 0: Empty 1: full	送信動作中 flag 0: 停止 1: 動作	STOP ビット (UART 用) 0: 1 ビット 1: 2 ビット	転送方向設定 0: LSB first 1: MSB first	W-buffer の許可 0: 不許可 1: 許可	SOFT RESET “10” → “01” のライトで RESET	

<TBEMP>: ダブルバッファ不許可の場合はこのフラグは意味を持ちません。送信ダブルバッファのデータ empty を示すフラグです。送信ダブルバッファのデータが送信シフトレジスタに移され empty になると “1” になり、送信データが書き込まれると “0” になります。

<RBFLl>: ダブルバッファ不許可の場合はこのフラグは意味を持ちません。受信ダブルバッファのデータ full を示すフラグです。受信動作が終了して、受信シフトレジスタから受信ダブルバッファへデータが格納されると “1” になり、読み出すと “0” になります。

<TXRUN>: 送信シフト動作中を示すステータスフラグです。
<TXRUN>と<TBEMP>ビットで以下のような状態を示します。

<TXRUN>	<TBEMP>	状態
1	-	送信動作中
0	1	送信が完全に終了
	0	送信バッファに次のデータがあり送信待ち

<SBLEN>: UART モード時の送信 STOP ビットの長さを指定します。受信の場合は設定に関わらず 1 ビットの STOP ビットで認識します。

<DRCHG>: I/O インタフェースモード時の転送方向を指定します。UART モード時は LSB first に設定してください。

<WBUF>: I/O インタフェースモードの送信 (SCLK 出力、入力)、受信 (SCLK 出力)、UART モードの送信時に、送信、受信の Double buffer の許可/不許可を指定します。I/O インタフェースモードの受信 (SCLK 入力)、UART モードの受信時は設定に関わらず、常に Double buffer は許可されます。

<SWRST[1:0]>: “10” → “01” の順に書き込むことでソフトウェアリセットが発生します。ソフトウェアリセットにより、以下のビットと内部回路が初期化されます。
(注1) (注2) (注3)

レジスタ名	ビット
SCOMODO	RXE
SCOMOD1	TXE
SCOMOD2	TBEMP, RBFLl, TXRUN,
SCOCR	OERR, PERR, FERR

- (注 1) 転送動作中にソフトリセットを実施する場合は2回連続して実行してください。
- (注 2) ソフトウェアリセット動作が完了するのに、命令実行後2クロックが必要です。
- (注 3) ソフトウェアリセットを実施した場合、他のビットは初期化されますので、モードレジスタ及びコントロールレジスタの再設定が必要となります。

11.4.7 ポーレートジェネレータコントロールレジスタ (SC0BRCR) ポーレートジェネレータコントロールレジスタ 2 (SC0BRADD)

	7	6	5	4	3	2	1	0
bit Symbol	—	BROADDE	BROCK1	BROCK0	BROS3	BROS2	BROS1	BROS0
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	“0” をラ イトしてく ださい	$N + (16 - K) / 16$ 分周機能 0: ディセー ブル 1: イネー ブル	ポーレートジェネレー タ入力クロック選択 00: $\phi T1$ 01: $\phi T4$ 10: $\phi T16$ 11: $\phi T64$	分周値 “N” の設定 0000: 16 分周 0001: 1 分周 0010: 2 分周 : 1111: 15 分周				

	7	6	5	4	3	2	1	0
bit Symbol					BROK3	BROK2	BROK1	BROK0
Read/Write	R				R/W			
リセット後	0				0	0	0	0
機能	リードすると “0” が読めます				$N + (16 - K) / 16$ 分周の K 値の設定 0000: 設定禁止 0001: K=1 0010: K=2 : 1111: K=15			

<RBOADDE>: $N + (16 - K) / 16$ 分周機能の設定を行うビットです。
この機能は、UART モードのときのみ使用可能です。

<RBROCK[1:0]>: ポーレートジェネレータ入力クロックを選択するビットです。

<RBROS[3:0]>: 分周値 “N” を設定するビットです。

<RBROK[3:0]>: $N + (16 - K) / 16$ 分周の “K” 値を設定するビットです。

ポーレートジェネレータの分周値は、上記の 2 つのレジスタで設定します。
ポーレートジェネレータ分周値の設定方法を表 11-6 にまとめます。

表 11-6 分周値の設定方法

	BROADDE=0 のとき	BROADDE=1 のとき (注1) (UART モードのみ使用可能)
BROS の設定	分周値 “N” を設定 (注2) (注3)	
BROK の設定	設定不要	“K” 値を設定 (注4)
分周値	N 分周	$N + \frac{(16 - K)}{16}$ 分周

- (注1) $N+(16-K)/16$ 分周機能を使用する場合、必ず BROK に “K” 値を設定後に BROADDE= “1” を設定してください。この機能は、UART モードのときのみ使用可能です。
- (注2) 分周値 “N” に 1 分周 (“0001”) を設定できるのは、以下の場合のみです。
- ・ UART モードでは、 $N+(16-K)/16$ 分周機能を使用しない場合。
 - ・ I/O インタフェースモードでは、ダブルバッファを使用する場合。
- (注3) $N+(16-K)/16$ 分周機能を使用する場合、分周値 “N” に 1 分周 (「0001」) と 16 分周 (「0000」) は設定できません。
- (注4) “K” 値に 「0」 を設定することはできません。

11.4.8 FIFO コンフィグレジスタ

	7	6	5	4	3	2	1	0
bit Symbol	Reserved	Reserved	Reserved	RFST	TFIE	RFIE	RXTXCNT	CNFG
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	必ず “000” を書き込んでください			受信 FIFO 使用 Byte 数 0:最大 1:受信 FIFO の FILL レベルに同じ	送信 FIFO 使用時の送信割り込み許可 0:禁止 1:許可	受信 FIFO 使用時の受信割り込み許可 0:禁止 1:許可	RXE/TXE の自動禁止 0:無し 1:自動禁止	FIFO の許可 0:禁止 1:許可

<RFST>: 受信 FIFO 使用 byte 数の設定ビットです。(注)
 “0” の場合、構成されている FIFO の最大の byte 数が使用可能です。
 (<CNFG>ビットの説明を参照してください)
 “1” の場合、受信 FIFO コンフィグレジスタ SCORFC<RIL[1:0]>で指定された FILL レベルの byte 数になります。

<TFIE>: 送信 FIFO が有効にされている時の送信割り込みの許可／不許可を切り替えます。

<RFIE>: 受信 FIFO が有効にされている時の受信割り込みの許可／不許可を切り替えます。

<RXTXCNT>: 送信/受信の自動禁止機能の制御ビットです。
 “1” に設定した場合、設定された通信方式により以下のように動作します。
 (通信方式はモードコントロールレジスタ 1 SCOMOD1<FDPX[1:0]>で設定します。)

半二重受信	受信 FIFO に指定された有効 byte 数分のデータが格納されると自動的に受信許可ビット (SCOMOD0<RXE>) が “0” となり、受信が禁止される。
半二重送信	送信 FIFO が Empty になると自動的に送信許可ビット (SCOMOD1<TXE>) が “0” となり、送信が禁止される。
全二重	上記の半二重受信または半二重送信のどちらかの条件が成立すると自動的に送信許可ビット、受信許可ビットとも “0” となり、送受信が禁止される。

<CNFG>: FIFO 使用の許可ビットです。
 “1” に設定すると FIFO が使用可能となります。設定された通信方式により FIFO の構成は以下ようになります。
 (通信方式はモードコントロールレジスタ 1 SCOMOD1<FDPX[1:0]>で設定します。)

半二重受信	受信 FIFO 4byte
半二重送信	送信 FIFO 4byte
全二重	受信 FIFO 2byte+送信 FIFO 2byte

(注) 送信 FIFO は常に構成されている FIFO の最大 BYTE 数を使用できます。
 使用 BYTE 数は送信 FIFO に書き込んだ BYTE 数になります。

11.4.9 受信 FIFO コンフィグレジスタ

	7	6	5	4	3	2	1	0
bit Symbol	RFCS	RFIS					RIL1	RIL0
Read/Write	W	R/W	R				R/W	
リセット後	0	0	0				0	0
機能	受信 FIFO クリア 1:クリア リードすると常に "0" が読めます	割り込み発生条件選択 0:fill レベル到達 1:fill レベル到達および新規データ読み出し時に fill レベルを超えている	リードすると "0" が読めます。				受信割り込みが発生する FIFO の fill レベル 00:4byte (全二重の場合は 2Byte) 01:1byte 10:2byte 11:3byte	

<RFCS>: 受信 FIFO クリアビットです。

"1"を書き込むと受信 FIFO がクリアされ、SCxRST<RLVL>="000"となります。また、リードポインタも初期化されます。

<RFIS>: 割り込み発生条件を選択します。

"0"で fill レベル到達時に割り込み発生、"1"で fill レベル到達時および、新規データ読み出し時に fill レベルを超えている場合に割り込みが発生します。

<RIL[1:0]>: FIFO の fill レベルを選択します。(注)

	全二重以外	全二重
00	4byte	2byte
01	1byte	1byte
10	2byte	2byte
11	3byte	1byte

(注) FDPX[1:0]=11 (全二重) の場合、RIL1 は無視されます

11.4.10 送信 FIFO コンフィグレジスタ

	7	6	5	4	3	2	1	0
bit Symbol	TFCS	TFIS					TIL1	TIL0
Read/Write	w	R/W	R				R/W	
リセット後	0	0	0				0	0
機能	送信 FIFO クリア 1:クリア リードすると常に "0" が読めます	割り込み発生条件選択 0:fill レベル到達 1:fill レベル到達 および新規データ書き込み時に fill レベルに達していない	リードすると "0" が読めます。				送信割り込みが発生する FIFO の fill レベル 00:Empty 01:1byte 10:2byte 11:3byte 注: FDPX1:0=11 (全二重) の場合、TIL1 は無視されます	

<TFCS>: 送信 FIFO クリアビットです。
"1" を書き込むと送信 FIFO がクリアされ、SCxTST<TLVL>="000" となります。また、ライトポインタも初期化されます。

<TFIS>: 割り込み発生条件を選択します。
"0" で fill レベル到達時に割り込み発生、"1" で fill レベル到達時および、新規データ書き込み時に fill レベルに達していない場合に割り込みが発生します。

<TIL[1:0]>: FIFO の fill レベルを選択します。(注)

	全二重以外	全二重
00	Empty	Empty
01	1byte	1byte
10	2byte	Empty
11	3byte	1byte

(注 1) FDPX[1:0]=11 (全二重) の場合、TIL1 は無視されます。

(注 2) SCOEN<0>=0 (SIO 動作禁止※クロック停止)、または SCOMOD1<7>=0 でスタンバイモード (IDLE, SLEEP, STOP) へ移行 (スタンバイモード中の動作停止※クロック停止) した場合は、必ず SCOTFC レジスタの再設定を行ってください。

11.4.11 受信 FIFO ステータスレジスタ

		7	6	5	4	3	2	1	0
SCORST	bit Symbol	ROR					RLVL2	RLVL1	RLVL0
	Read/Write	R	R				R		
	リセット後	0	0				0	0	0
	機能	受信 FIFO Overrun 1:発生	リードすると“0”が読めます。				受信 FIFO の fill レベルのステータス 000:Empty 001:1Byte 010:2Byte 011:3Byte 100:4Byte		

<ROR>: 受信 FIFO のオーバーランフラグです。
オーバーランが発生すると“1”にセットされます。(注)

<RLVL[2:0]>: 受信 FIFO の fill レベルを示すビットです。

(注) <ROR>ビットはバッファレジスタ (SCOBUF) を読み出すと“0”にクリアされます。

11.4.12 送信 FIFO ステータスレジスタ

		7	6	5	4	3	2	1	0
SCOTST	bit Symbol	TUR					TLVL2	TLVL1	TLVL0
	Read/Write	R	R				R		
	リセット後	1	0				0	0	0
	機能	送信 FIFO Under run 1:発生 FIFO ライトでクリア	リードすると“0”が読めます。				送信 FIFO の fill レベルのステータス 000:Empty 001:1Byte 010:2Byte 011:3Byte 100:4Byte		

<TUR>: 送信 FIFO のアンダーランフラグです。
アンダーランが発生すると“1”にセットされます。(注)

<TLVL[2:0]>: 送信 FIFO の fill レベルを示すビットです。

(注) <TUR>ビットはバッファレジスタ (SCOBUF) に書き込みを行うと“0”にクリアされます。

11.5 モード別動作説明

11.5.1 モード0 (I/O インターフェースモード)

このモードには、同期クロック SCLK を出力する SCLK 出力モードと、外部より同期クロック SCLK を入力する SCLK 入力モードがあります。以下に FIFO の使用が禁止されている状態での各動作の説明を行います。FIFO の動作については、前述の受信 FIFO 動作及び送信 FIFO 動作の項を参照してください。

① 送信

SCLK 出力モード

SCLK 出力モードでは SCOMOD2<WBUF>="0" で送信ダブルバッファが不許可の場合、CPU が送信バッファにデータを書き込むたびに、8 ビットのデータが TXD0 端子、同期クロックが SCLK0 端子より出力されます。データがすべて出力されると割り込み (INTTX0) が発生します。

SCOMOD2<WBUF>="1" で送信ダブルバッファが許可されている場合は、送信が停止している状態で CPU が送信バッファ 2 にデータを書き込んだとき、または送信バッファ 1 (シフトレジスタ) のデータ送出が終了したときに送信バッファ 2 より、送信バッファ 1 にデータが移されます。これと同時に送信バッファ empty フラグ SCOMOD2<TBEMP>が "1" にセットされます。同時に、割り込み (INTTX0) が発生します。このときに送信バッファ 2 に送信バッファ 1 へ移すデータが存在しない場合は、割り込み (INTTX0) を発生せず、SCLK0 出力も停止します。

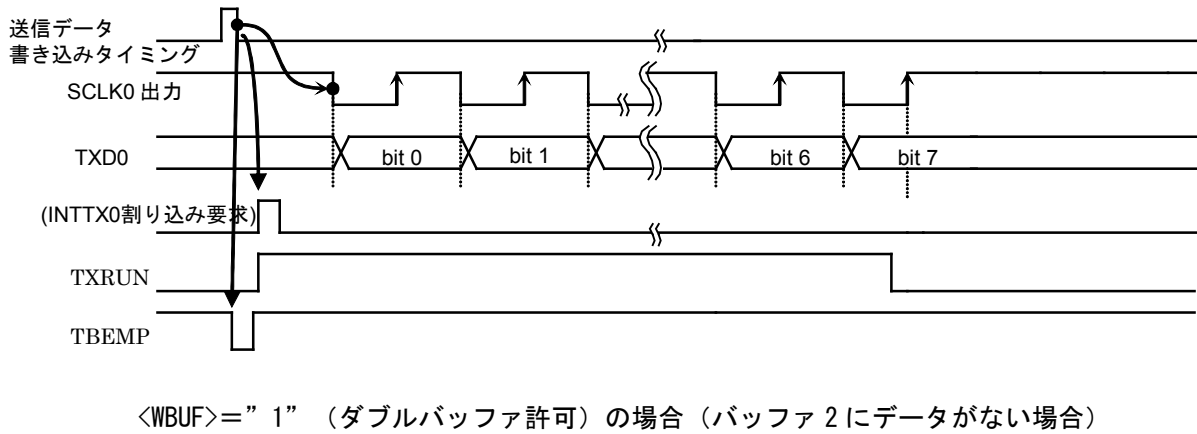
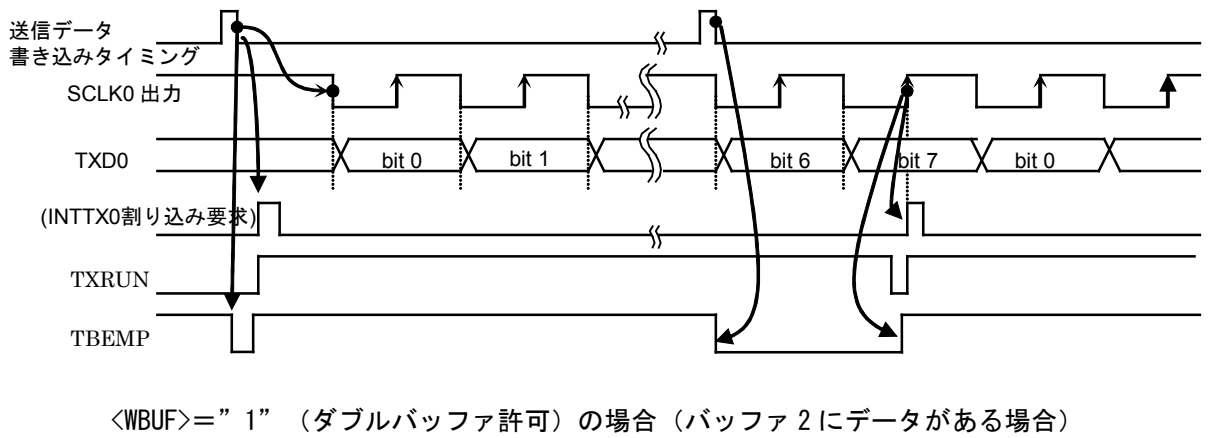
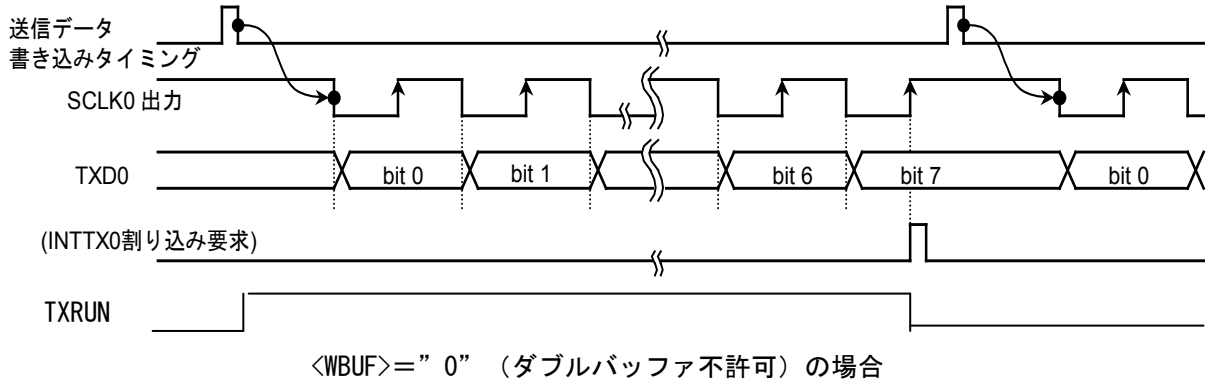
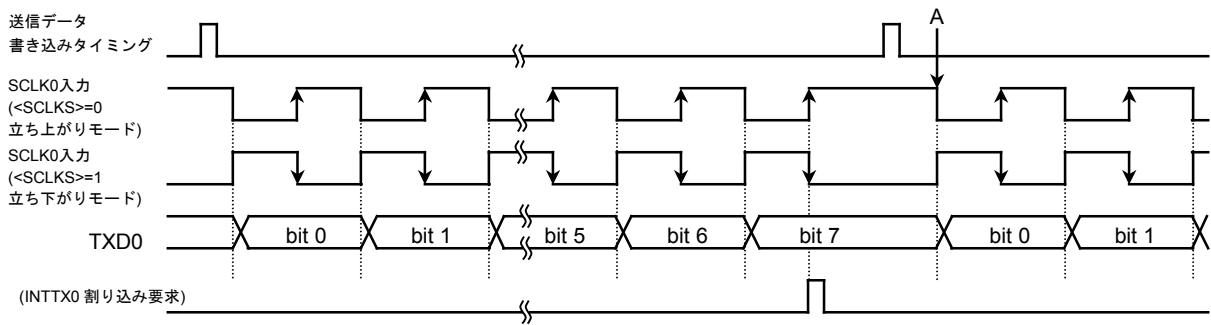


図 11-10 I/O インターフェースモード送信動作 (SCLK0 出力モード)

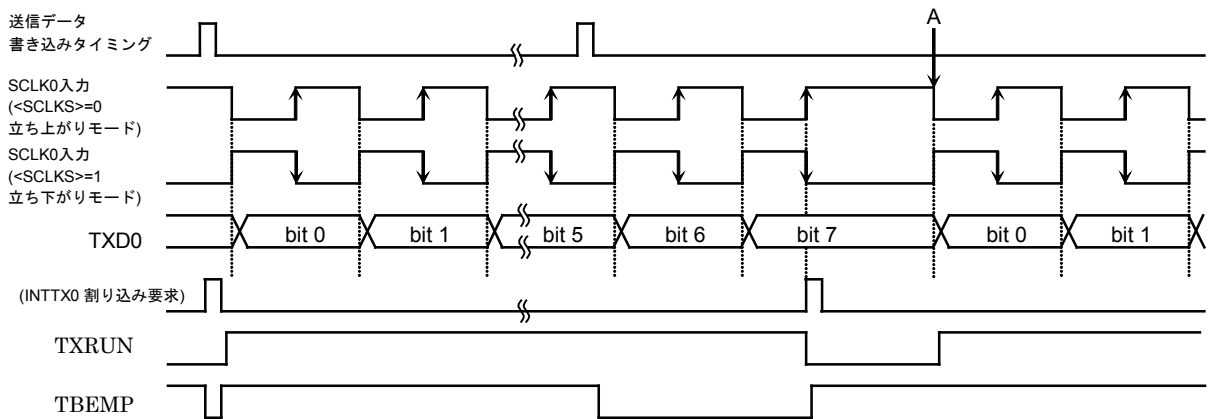
SCLK 入力モード

SCLK 入力モードでは、SCOMOD2<WBUF>=" 0" で送信ダブルバッファが不許可の場合は、送信バッファにデータが書き込まれている状態で SCLK0 入力アクティブになると、8 ビットのデータが TXD0 端子より出力されます。データがすべて出力されると割り込み INTTX0 が発生します。次の送信データは図 11.11 に示す A 点までに書き込んでください。

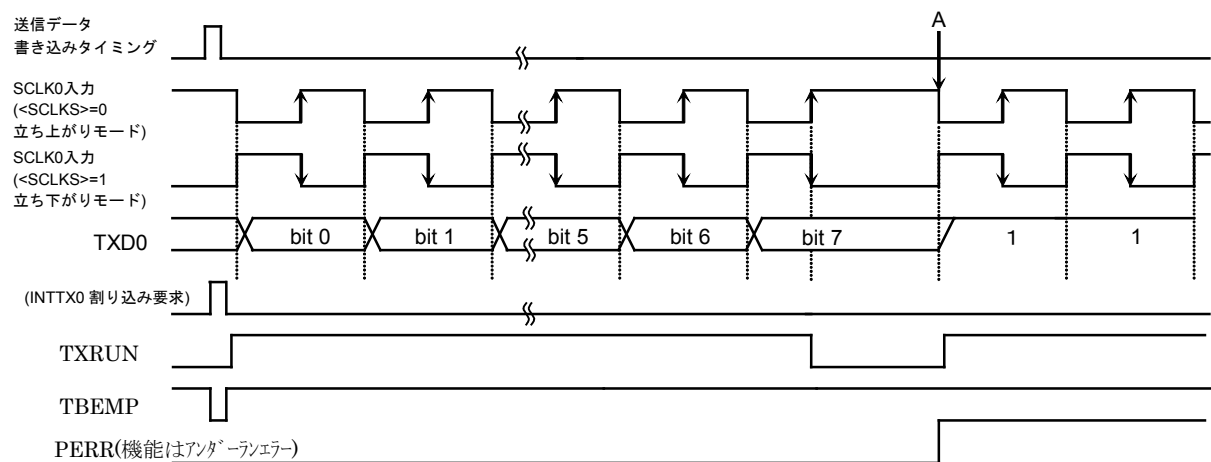
SCOMOD2<WBUF>=" 1" で送信ダブルバッファが許可されている場合は、SCLK0 入力アクティブになる前に送信バッファにデータを書き込んだとき、または送信バッファ 1 (シフトレジスタ) のデータ送出が終了したときに送信バッファ 2 のデータが送信バッファ 1 へ移されます。これと同時に送信バッファ empty フラグ SCOMOD2<TBEMP> が "1" にセットされ、割り込み (INTTX0) が発生します。送信バッファ 2 にデータが書き込まれていない状態で、SCLK0 入力アクティブになった場合は、内部ビット数カウンタはカウントを開始しますが、アンダーランエラーがセットされ、8 ビット分のダミーデータ (FFh) を送出します。



<WBUF>=" 0" (ダブルバッファ不許可) の場合



<WBUF>=" 1" (ダブルバッファ許可) の場合 (バッファ 2 にデータがある場合)



<WBUF>=" 1" (ダブルバッファ許可) の場合 (バッファ 2 にデータがない場合)

図 11-11 I/O インターフェースモード送信動作 (SCLK0 入力モード)

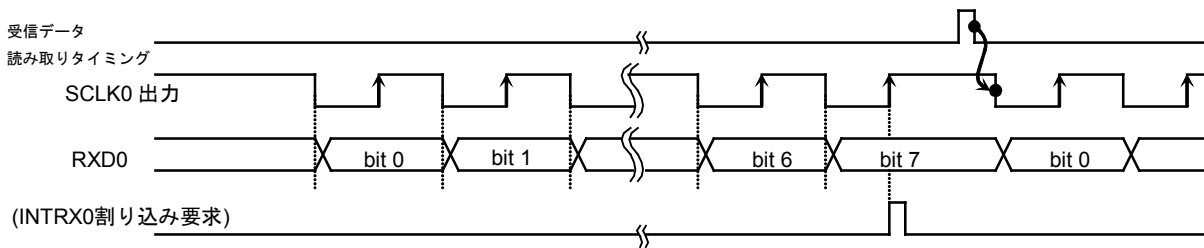
② 受信

SCLK 出力モード

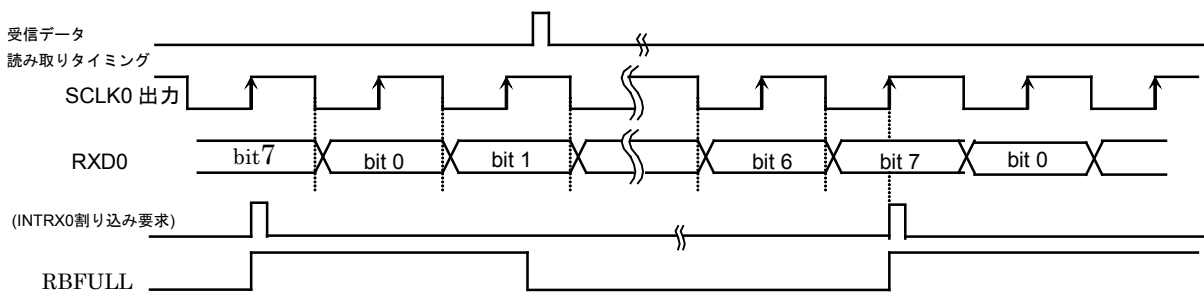
SCLK 出力モードでは SCOMOD2<WBUF>="0" で受信ダブルバッファが不許可の場合は、受信データが CPU に読み取られるたびに、SCLK0 端子より同期クロックが出力され次のデータが受信バッファ 1 にシフトインされます。8 ビットデータが受信されると、割り込み INTRX0 が発生します。

最初の SCLK 出力の開始は、受信許可ビット SCOMOD0<RXE>を "1" にセットすることで行います。また、SCOMOD2<WBUF>="1" で受信ダブルバッファが許可の場合は、最初に受信したフレームは 受信バッファ 2 に移され、続けて次のフレームを受信バッファ 1 で受信します。受信バッファ 1 から受信バッファ 2 にデータが移されると、受信バッファ full フラグ SCOMOD2<RBFULL>が "1" にセットされ、割り込み INTRX0 が発生します。

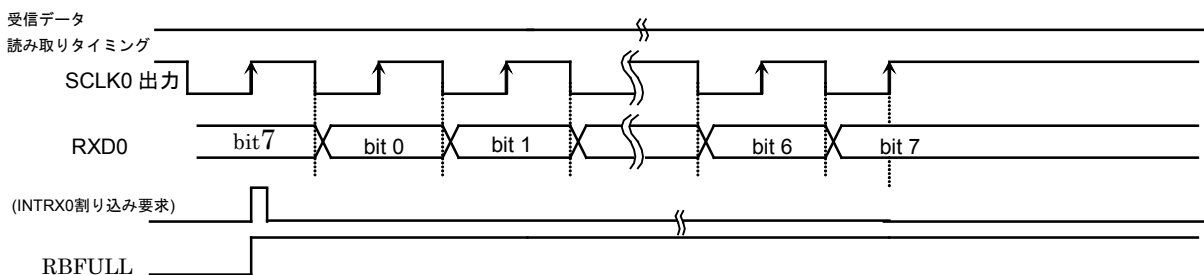
受信バッファ 2 にデータが移された状態で 8 ビット分のデータを受信完了する前に、CPU/DMAC で受信バッファ 2 のデータを読み出されない場合は、割り込み INTRX0 が発生せず、SCLK0 出力も停止します。この状態で受信バッファ 2 のデータを読み出すと、受信バッファ 1 のデータを受信バッファ 2 に移し、割り込み INTRX0 を発生して受信を再開します。



<WBUF>="0" (ダブルバッファ不許可) の場合



<WBUF>="1" (ダブルバッファ許可) の場合 (バッファ 2 のデータを読み出した場合)



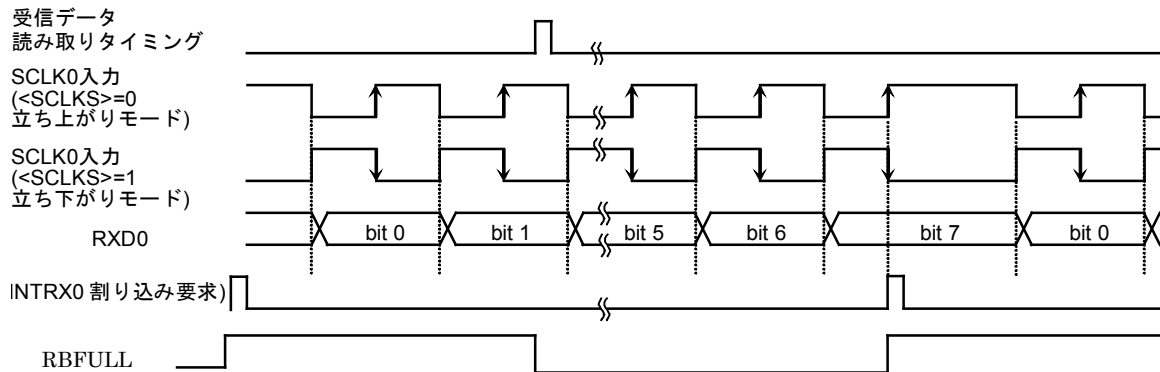
<WBUF>="1" (ダブルバッファ許可) の場合 (バッファ 2 のデータを読み出されない場合)

図 11-12 I/O インターフェースモード受信動作 (SCLK0 出力モード)

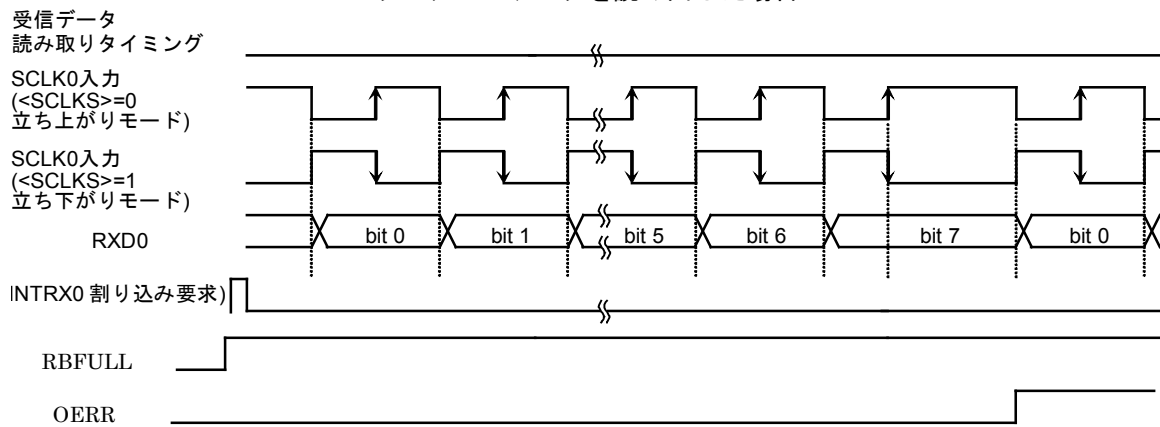
SCLK 入力モード

SCLK 入力モードでは常に受信ダブルバッファが許可されており、受信したフレームは受信バッファ 2 に移され、受信バッファ 1 で連続して次のフレームを受信することができます。

受信データが受信バッファ 2 へ移されるごとに受信割り込み INTRX が発生します。



バッファ 2 のデータを読み出した場合



バッファ 2 のデータが読み出されない場合

図 11-13 I/O インターフェースモード受信動作 (SCLK0 入力モード)

(注) 受信動作を行う場合には SCLK 入/出力のどちらのモードでも、受信イネーブル状態 (SCOMODO <RXE> = 1) にしておく必要があります。

③ 送受信（全二重）

シリアルモードコントロールレジスタ 1 (SCOMOD1) のビット 6 <FDPX0> に "1" をセットすることにより全二重モードでの通信が可能になります。

SCLK 出力モード

SCLK 出力モードでは SCOMOD2<WBUF>="0" で送受信共にダブルバッファが不許可の場合は、CPU が送信バッファにデータを書き込むと SCLK が出力され、8 ビットデータが受信バッファ 1 にシフトインされ、受信割り込み (INTRX0) が発生します。それと並行して送信バッファに書き込まれた 8 ビットデータが、TXD0 端子より出力され、全てのデータが送信されると送信割り込み (INTTX0) が発生します。この状態で SCLK の出力は停止します。この状態で CPU が受信バッファを読み出し、送信バッファヘデータを書き込むと次の送受信が開始されます。受信バッファの読み出しと送信バッファの書き込み順番は任意です。両方の条件が成立した場合に再開されます。

SCOMOD2<WBUF>="1" で送受信共にダブルバッファが許可されている場合は、CPU が送信バッファにデータを書き込むと SCLK が出力され、8 ビットデータが受信バッファ 1 にシフトインされると、データは受信バッファ 2 に移され割り込み (INTRX0) が発生します。8 ビットデータの受信と並行して 8 ビットデータが TXD0 端子より出力されます。データがすべて出力されると割り込み (INTTX0) が発生して次のデータが送信バッファ 2 から送信バッファ 1 移されます。この時に、送信バッファ 2 に移すデータが存在しない (SCOMOD2<TBEMP>="1") または受信バッファ 2 にデータが存在している (SCOMOD2<RBFULL>="1") 場合は SCLK 出力が停止します。その後は受信データの読み出しと送信データの書き込みの両方の条件が成立すると再度 SCLK の出力が開始されて次の送受信が始まります。

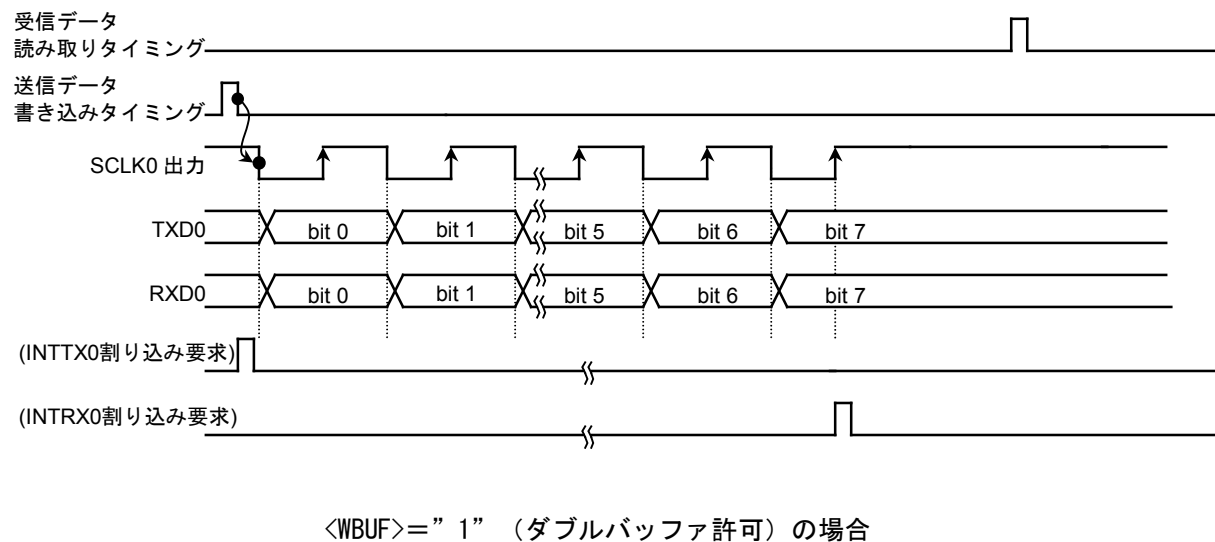
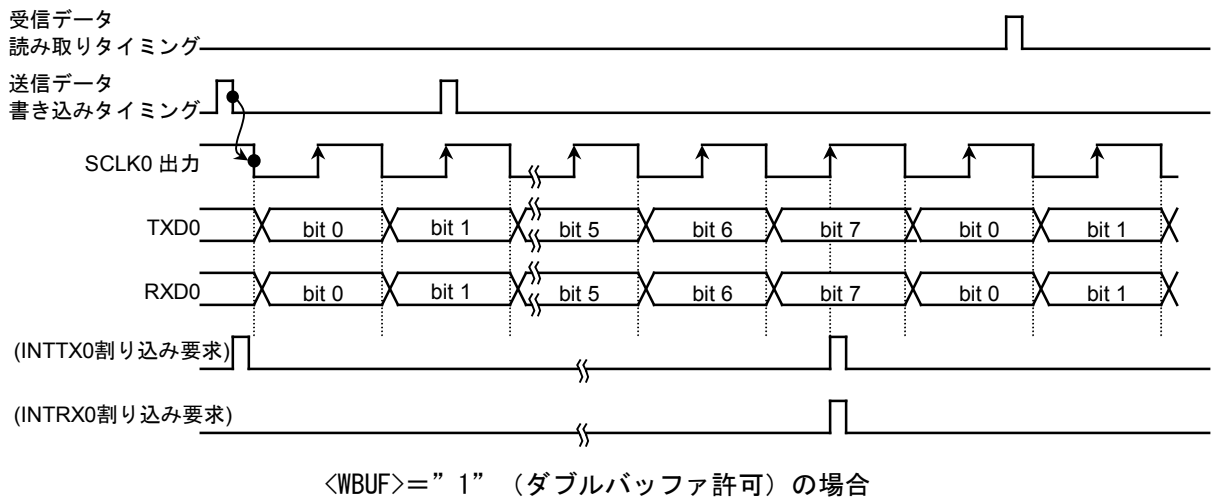
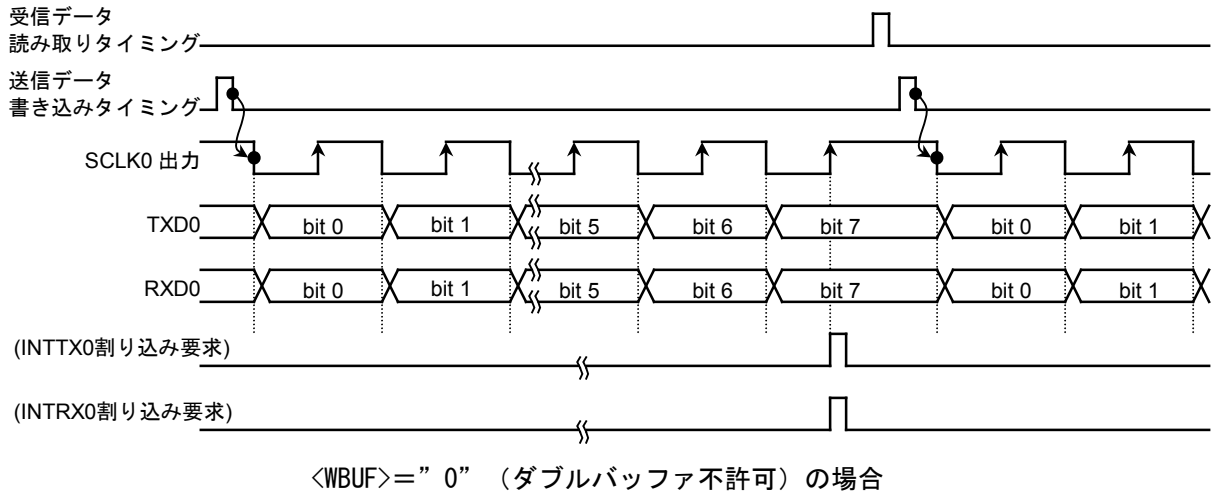


図 11-14 I/O インターフェースモード送受信動作 (SCLK0 出力モード)

SCLK 入力モード

SCLK 入力モードでは SCOMOD2<WBUF>=" 0" で送信ダブルバッファが不許可（受信は設定に関わらずダブルバッファ有効）の場合は、送信バッファにデータが書き込まれている状態で SCLK 入力アクティブになると、8 ビットのデータが TXD0 より出力されると同時に 8 ビットのデータが受信バッファへシフトインされます。データの送信が終了すると割り込み（INTTX0）が、受信が終了すると受信バッファ 1 から受信バッファ 2 へデータが移されると同時に割り込み（INTRX0）が発生します。次のフレームの SCLK が入力される前に送信データを送信バッファへ書き込む様にしてください。図 11.15 に示す A 点までに書き込んでください。受信データに関してはダブルバッファが有効になっている為に次のフレームの受信が終了する前に読み出してください。

SCOMOD2<WBUF>=" 1" で送受信共にダブルバッファが許可されている場合は、送信バッファ 1 のデータの送信が終了すると、送信バッファ 2 のデータが送信バッファ 1 へ移されると同時に割り込み（INTRX0）が発生します。また並行して受信した 8 ビットデータのバッファ 1 へのシフトインが終了すると、そのデータは受信バッファ 2 へ移され、割り込み（INTRX0）が発生します。続けて次のフレームの SCLK が入力されると、送信バッファ 2 から送信バッファ 1 へ移されたデータの送信が始まり、並行して受信バッファ 1 でのデータ受信が行なわれます。また、フレームの最終ビットの受信までに受信バッファ 2 のデータが読み出されていない場合はオーバーランエラーが発生します。また、次のフレームの SCLK 入力までに送信バッファ 2 へ転送データが書き込まれていない場合はアンダーランエラーが発生します。

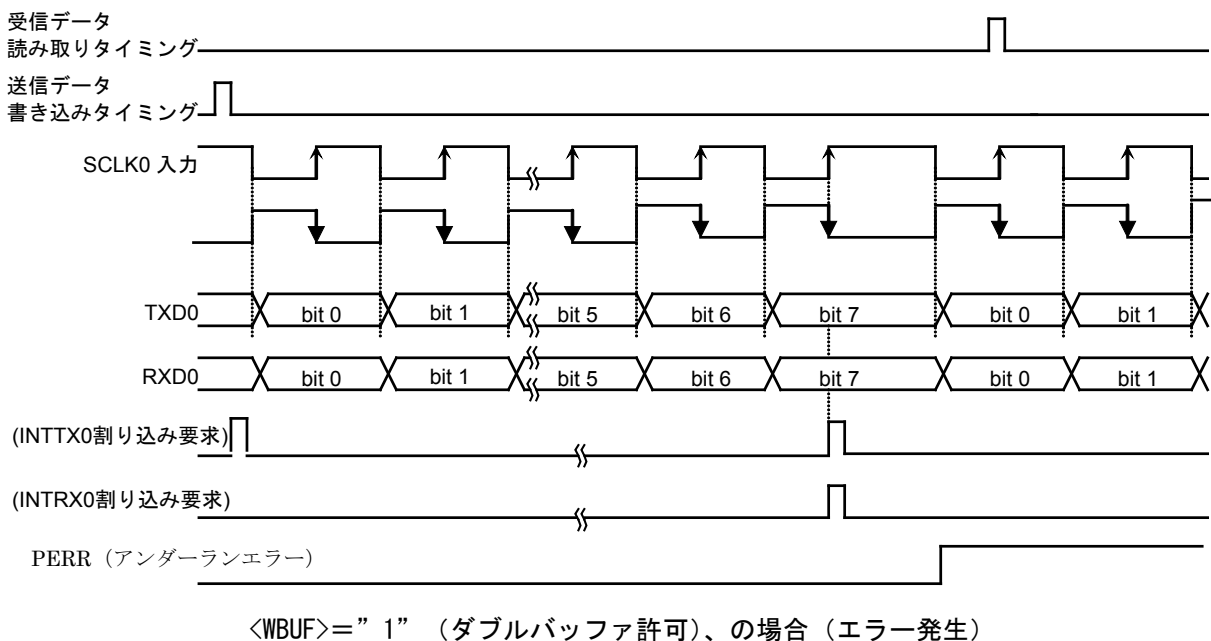
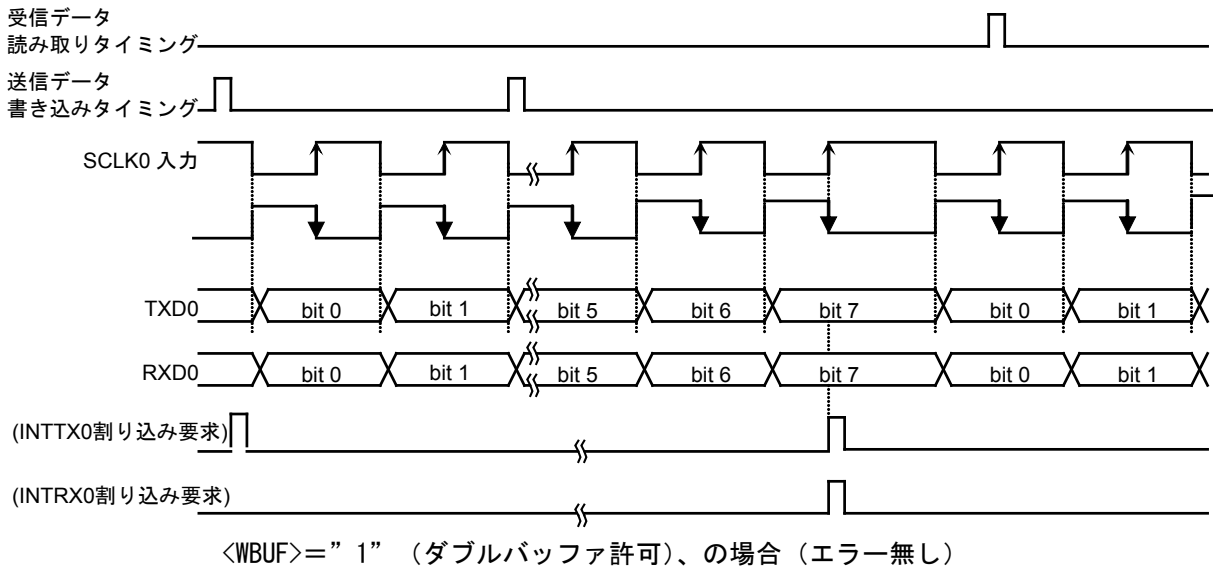
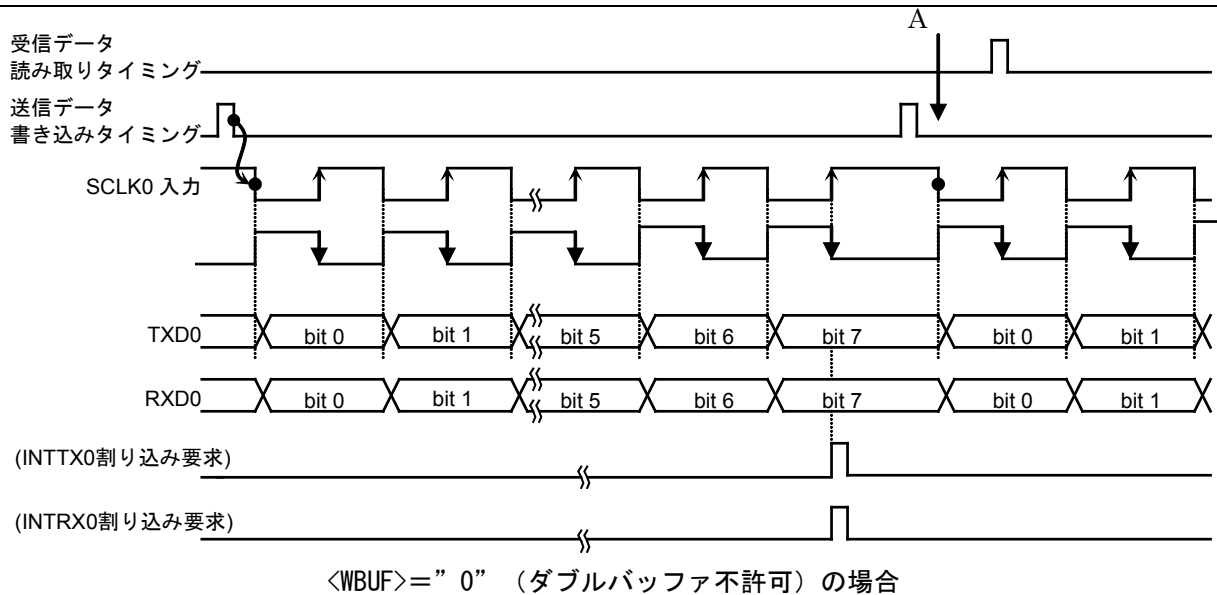


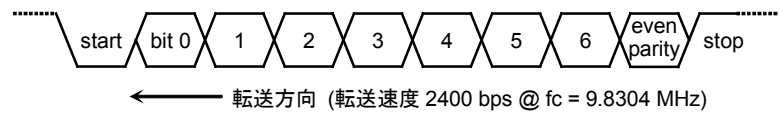
図 11-15 I/O インターフェースモード送受信動作 (SCLK0 入力モード)

11.5.2 モード1 (7ビットUARTモード)

シリアルモードコントロールレジスタ (SCOMOD0 <SM[1:0]>) を 01 にセットすると 7 ビット UART モードになります。

このモードではパリティビットの付加が可能で、シリアルモードコントロールレジスタ (SCOCR <PE>) でパリティビット付加のイネーブル/ディセーブルを制御しています。<PE>= “1” (イネーブル) のときは、SCOCR<EVEN>で偶数パリティ/奇数パリティを選択できます。STOP ビットの長さは SCOMOD2<SBLEN>で指定することができます。

例：下記フォーマットのデータを送信する場合の各コントロールレジスタの設定を示します。

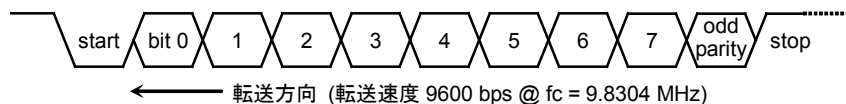


※ クロック条件	{	システムクロック : 高速 (fc) 高速クロックギア : 1倍 (fc) プリスケーラクロック : $f_{\text{periph}}/2$ ($f_{\text{periph}} = f_{\text{sys}}$)
----------	---	---

11.5.3 モード2 (8ビットUARTモード)

SCOMODO <SM[1:0]> を10にセットすると8ビットUARTモードになります。このモードでは、パリティビットの付加が可能で SCOCR <PE> でパリティビット付加のイネーブル/ディセーブルを制御できます。<PE> = “1” (イネーブル) のとき、SCOCR <EVEN> で偶数パリティ/奇数パリティの選択も可能です。

例：下記のフォーマットのデータを受信する場合の各コントロールレジスタの設定を示します。



※ クロック条件	{	システムクロック	:	高速 (fc)
		高速クロックギア	:	1倍 (fc)
		プリスケールクロック	:	$f_{\text{periph}}/2$ ($f_{\text{periph}} = f_{\text{sys}}$)

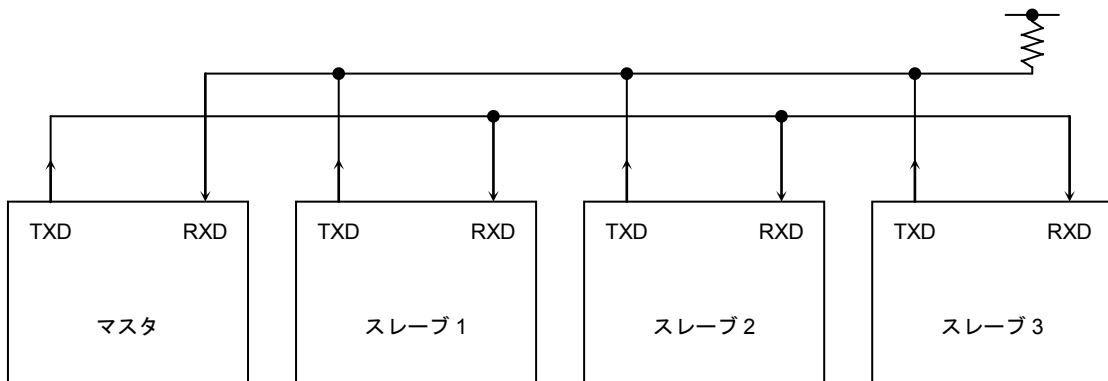
11.5.4 モード3 (9ビットUART)

SCOMOD0 <SM[1:0]> を “11” にセットすると9ビットUARTモードになります。このモードでは、パリティビットの付加を禁止 (SCOCR<PE> = “0”) してください。

最上位ビット (9ビット目) は、送信の場合シリアルモードコントロールレジスタ 0 (SCOMOD0)のビット7<TB8>に書き込み、受信の場合シリアルコントロールレジスタ SCOCRのビット7<RB8>に格納されます。また、バッファに対する書き込み、読み出しはかならず最上位ビットの方を先に行い、SCOBUFの方を後にします。STOPビットの長さはSCOMOD2<SBLEN>で指定することができます。

ウェイクアップ機能

9ビットUARTモードでは、ウェイクアップ機能制御ビット SCOMOD0<WU>を “1” にすることによって、スレーブコントローラのウェイクアップ動作が可能で、SCOCR<RB8>= “1” のときのみ割り込み (INTRX0) が発生します。



(注) スレーブコントローラのTXD端子は、かならずODEレジスタを設定してオープンドレイン出力モードにしてください。

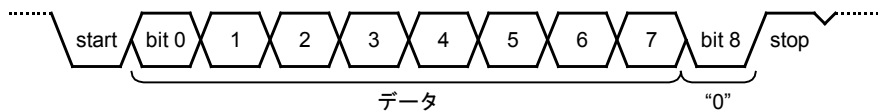
図 11-16 ウェイクアップ機能によるシリアルリンク

プロトコル

- ① マスタおよびスレーブコントローラは9ビットUARTモードにします。
- ② 各スレーブコントローラはSCOMODO〈WU〉を“1”にセットし、受信可能状態とします。
- ③ マスタコントローラは、スレーブコントローラのセレクトコード(8ビット)を含む1フレームを送信します。このとき最上位ビット(ビット8)〈TB8〉は“1”にします。

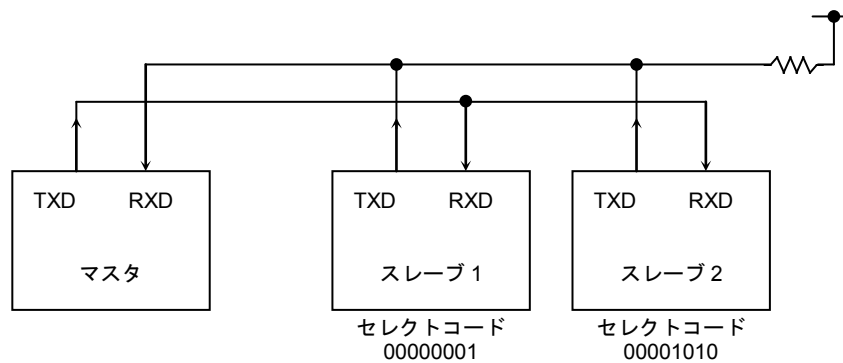


- ④ 各スレーブコントローラは、上記フレームを受信し、自分のセレクトコードと一致すれば、WUビットを“0”にクリアします。
- ⑤ マスタコントローラは指定したスレーブコントローラ(SCOMODO〈WU〉 = “0”にクリアされたコントローラ)に対しデータを送信します。このとき、最上位ビット(ビット8)〈TB8〉は“0”にします。



- ⑥ 〈WU〉=“1”のままのスレーブコントローラは、受信データの最上位ビット(ビット8)の〈RB8〉が“0”であるため、割り込み(INTRX0)が発生せず、受信データを無視します。また、〈WU〉 = “0”になったスレーブコントローラがマスタコントローラにデータを送信し、この送信データで受信終了をマスタコントローラに知らせることもできます。

設定例: 内部クロック f_{SYS} を転送クロックとして2つのスレーブコントローラとシリアルリンクさせる場合



12. シリアルバスインタフェース (SBI)

シリアルバスインタフェース (SBI) を 3 チャンネル内蔵しています。シリアルバスインタフェースは、下記の 2 つの動作モードを持っています。

- I²C バスモード (マルチマスタ)
- クロック同期式 8 ビット SIO モード

I²C バスモードでは、SCL および SDA を通して外部デバイスと接続されます。

クロック同期式 8 ビット SIO モードでは、SCK, SI, SO を通して外部デバイスと接続されます。

チャンネルごとの使用端子とポートの設定は、下記の通りです。

		端子 (PIN No.)	ポートファンクションレジスタ 設定	ポートコントロールレジスタ 設定	ポートインピットレジスタ 設定	ポートアウトプットレイン 出力設定
チャンネル 0	I ² C バスモード	SCL: PG1 (27) SDA: PG0 (26)	PGFR1[1:0] = 11	PGCR[1:0] = 11	PGIE[1:0] = 11	PGOD[1:0] = 11
	クロック同期式 SIO モード	SCK: PG2 (28) SI : PG1 (27) SO : PG0 (26)	PGFR1[2:0] = 111	PGCR[2:0] = 101	PGIE[2:0] = 110	PGOD[2:0] = xxx
チャンネル 1	I ² C バスモード	SCL: PF5 (45) SDA: PF4 (44)	PFFR1[5:4] = 11	PFGR[5:4] = 11	PFIE[5:4] = 11	PFOD[5:4] = 11
	クロック同期式 SIO モード	SCK: PF6 (46) SI : PF5 (45) SO : PF4 (44)	PFFR1[6:4] = 111	PFGR[6:4] = 101	PFIE[6:4] = 110	PFOD[6:4] = xxx
チャンネル 2	I ² C バスモード	SCL: PG5 (16) SDA: PG4 (15)	PGFR1[5:4] = 11	PGCR[5:4] = 11	PGIE[5:4] = 11	PGOD[5:4] = 11
	クロック同期式 SIO モード	SCK: PG6 (17) SI : PG5 (16) SO : PG4 (15)	PGFR1[6:4] = 111	PGCR[6:4] = 101	PGIE[6:4] = 110	PGOD[6:4] = xxx

x: Don't care

12.1 構成

構成を図 12-1に示します。

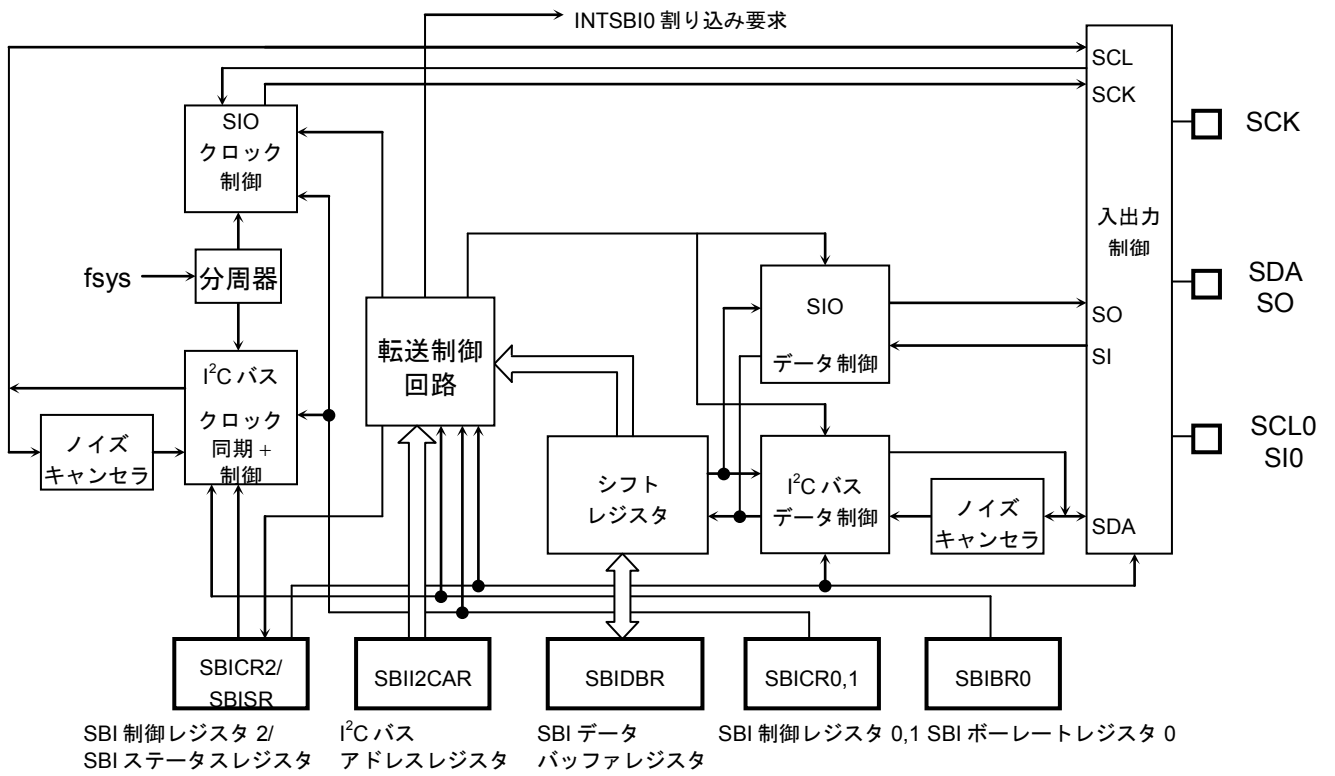


図 12-1 SBI のブロック図

12.2 制御

シリアルバスインタフェースの制御および動作状態のモニタは、以下のレジスタで行います。

- シリアルバスインタフェース制御レジスタ 0 (SBIxCR0)
- シリアルバスインタフェース制御レジスタ 1 (SBIxCR1)
- シリアルバスインタフェース制御レジスタ 2 (SBIxCR2)
- シリアルバスインタフェースバッファレジスタ (SBIxDBR)
- I²C バスアドレスレジスタ (SBIxI2CAR)
- シリアルバスインタフェースステータスレジスタ (SBIxSR)
- シリアルバスインタフェースポーレートレジスタ 0 (SBIxBR0)

上記レジスタは使用するモードによって、機能が異なります。詳細は「12.5 I²C バスモード時の制御」および「12.7 クロック同期式 8 ビット SIO モード時の制御」を参照してください。

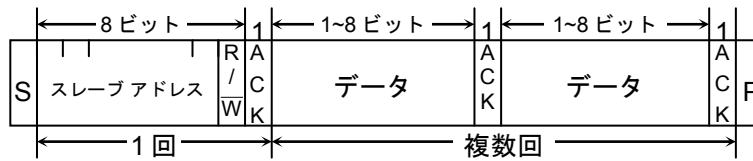
各レジスタのアドレスは以下のとおりです。

		チャンネル 0	チャンネル 1	チャンネル 2
レジスタ名 (アドレス)	シリアルバスインタフェース 制御レジスタ 0	SBI0CR0 0x4002_0000	SBI1CR0 0x4002_0020	SBI2CR0 0x4002_0040
	シリアルバスインタフェース 制御レジスタ 1	SBI0CR1 0x4002_0004	SBI1CR1 0x4002_0024	SBI2CR1 0x4002_0044
	シリアルバスインタフェース 制御レジスタ 2	SBI0CR2 (ライト時) 0x4002_0010	SBI1CR2 (ライト時) 0x4002_0030	SBI2CR2 (ライト時) 0x4002_0050
	シリアルバスインタフェース ステータスレジスタ	SBI0SR (リード時)	SBI1SR (リード時)	SBI2SR (リード時)
	シリアルバスインタフェース ポーレートレジスタ 0	SBI0BR0 0x4002_0014	SBI1BR0 0x4002_0034	SBI2BR0 0x4002_0054
	シリアルバスインタフェース データバッファレジスタ	SBI0DBR 0x4002_0008	SBI1DBR 0x4002_0028	SBI2DBR 0x4002_0048
	I ² C バスアドレスレジスタ	SBI0I2CAR 0x4002_000C	SBI1I2CAR 0x4002_002C	SBI2I2CAR 0x4002_004C

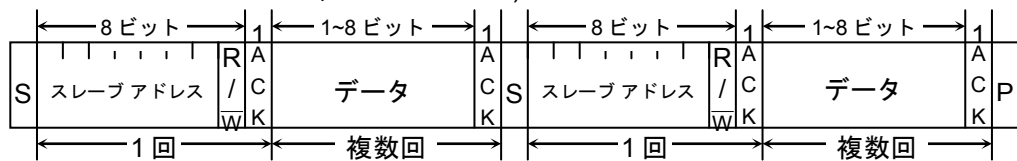
12.3 I²C バスモード時のデータフォーマット

I²C バスモード時のデータフォーマットを図 12-2に示します。

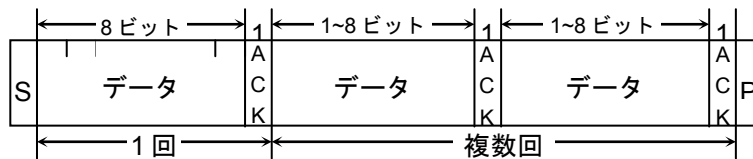
(a) アドレッシングフォーマット



(b) アドレッシングフォーマット (再スタート有り)



(c) フリーデータフォーマット (マスタデバイスからスレーブデバイスへデータを転送する転送フォーマット)



- 注) S: スタートコンディション
 R/ \bar{W} : 方向ビット
 ACK: アクノリッジビット
 P: ストップコンディション

図 12-2 I²C バスモード時のデータフォーマット

12.4 I²C バスモード時のコントロールレジスタ

シリアルバスインタフェース (SBI) を I²C バスモードで使用するときの制御、および動作状態のモニタは以下のレジスタで行います。

シリアルバスコントロールレジスタ 0

SBIxCR0

	7	6	5	4	3	2	1	0
bit Symbol	SBIEN							
Read/Write	R/W							
リセット後	0							
機能	SBI 動作 0: 禁止 1: 許可							
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

<SBIEN> : SBIを使用する場合は、SBIモジュールの各レジスタを設定する前にSBI動作許可("1")にしてください。

図 12-3 I²C バスモード関係のレジスタ

シリアルバスコントロールレジスタ 1

	7	6	5	4	3	2	1	0
bit Symbol	BC2	BC1	BC0	ACK		SCK2	SCK1	SCK0/ SWRMON
Read/Write	R/W			R/W	R	R/W		R/W
リセット後	0	0	0	0	1	0	0	1
機能	転送ビット数の選択 (注1)			アクノリツジメント クロック 0: 発生 しない 1: 発生 する	リードす ると"1" が読めま す	内部 SCL 出力クロックの 周波数選択 (注2) とリセット モニタ		
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0" が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0" が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0" が読めます							

<Bit 2:0><SCK[2:0]> : 内部 SCL 出力クロックの周波数選択。

内部 SCL 出力クロックの周波数選択 <SCK2 : 0> @ライト

000	n=5	384 kHz	システムクロック: fsys (=40 MHz) クロックギア : fc/1 周波数 = $\frac{f_{sys}}{2^n + 72}$ [Hz]
001	n=6	294 kHz	
010	n=7	200 kHz	
011	n=8	121 kHz	
100	n=9	68 kHz	
101	n=10	36 kHz	
110	n=11	18 kHz	
111		reserved	

<Bit 0><SWRMON:0> : リセットモニタ

ソフトウェアリセット状態モニタ <SWRMON> @リード

0	ソフトウェアリセット中
1	ソフトウェアリセット解除中

<Bit 7:5><BC[2:0]> : 転送ビット数の選択

転送ビット数の選択

<BC2:0>	<ACK> = 0 のとき		<ACK> = 1 のとき	
	クロック数	データ長	クロック数	データ長
000	8	8	9	8
001	1	1	2	1
010	2	2	3	2
011	3	3	4	3
100	4	4	5	4
101	5	5	6	5
110	6	6	7	6
111	7	7	8	7

図 12-4 I²C バスモード関係のレジスタ

- (注 1) クロック同期式 8 ビット SIO モードに切り替える前に<BC2:0>を “000” にクリアしてください。
- (注 2) SCL ラインクロックの周波数については、「12.5.3 シリアルクロック」を参照してください。
- (注 3) <SCK0/SWRMON>ビットは、リセット後 “1” が読み出されますが、SB1xCR2 レジスタにて SIO モードに設定した場合、<SCK0>ビットの初期値は “0” になります。
- (注 4) 読み出しの初期値とは関係なく、周波数選択の初期値は<SCK[2:0]>=000 です。
- (注 5) マスターモードで<BC[2:0]>=“001”で<ACK>=“0”のとき、ストップコンディション発生後の SCL ラインの立ち下りによって、SCL ラインが“L”に固定されることがあり、他のマスタデバイスがバスを使用できなくなります。複数のマスタデバイスがバスに接続されている場合、ストップコンディション発生前に、転送ビット数を“2”以上にしてください。

シリアルバスコントロールレジスタ 2

SBIxCR2

	7	6	5	4	3	2	1	0
bit Symbol	MST	TRX	BB	PIN	SBIM1	SBIM0	SWRST1	SWRST0
Read/Write	W				W (注 2)		W (注 1)	
リセット後	0	0	0	1	0	0	0	0
機能	マスタ/スレーブの選択 0: スレーブ 1: マスタ	送信/受信の選択 0: 受信 1: 送信	スタート/ストップ状態の発生 0: ストップ状態発生 1: スタート状態発生	INTSBIn 割り込み要求解除 0: - 1: 割り込み要求の解除	シリアルバスインタフェースの動作モード選択 (注 2) 00: ポートモード 01: SIO モード 10: I ² C バスモード 11: (予約)		ソフトウェアリセットの発生 最初に“10”、次に“01”をライとすると、リセットが発生します。	
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると“0”が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると“0”が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると“0”が読めます							

- <Bit 1:0><SWRST[1:0]> :最初に“10”、次に“01”をライとすると、リセットが発生します。
- <Bit 3:2><SBIM[1:0]> :シリアルバスインタフェースの動作モード選択

シリアルバスインタフェースの動作モード選択 (注 2)

00	ポートモード (シリアルバスインタフェースの出力禁止)
01	クロック同期式 8 ビット SIO モード
10	I ² C バスモード
11	(予約)

- <Bit 4><PIN> :INTSBIn 割り込み要求解除
- <Bit 5><BB> :状態発生の確認
- <Bit 6><TRX> :送受信の選択
- <Bit 7><MST> :マスタスレーブの選択

- (注 1) このレジスタをリードすると、SBInSR レジスタとして機能します。
- (注 2) ポートモードへの切り替えは、バスフリーを確認してから行ってください。また、ポートモードから I²C バスモード、クロック同期式 8 ビット SIO への切り替えは、ポートの状態が“H”になっていることを確認してから行ってください。
- (注 3) モードの切替えはシリアル転送が終了していることを確認してから行ってください。

図 12-5 I²C バスモード関係のレジスタ

表 12-1 ベースクロック 分解能

@fsys = 40 MHz

クロックギア値 <GEAR[1:0]>	ベースクロック 分解能
00 (fc)	$f_{\text{sys}}/2^2$ (0.1 μs)
01 (fc/2)	$f_{\text{sys}}/2^3$ (0.2 μs)
10 (fc/4)	$f_{\text{sys}}/2^4$ (0.4 μs)
11 (fc/8)	$f_{\text{sys}}/2^5$ (0.8 μs)

シリアルバスインタフェースステータスレジスタ

	7	6	5	4	3	2	1	0
SBIxSR	MST	TRX	BB	PIN	AL	AAS	ADO	LRB
Read/Write	R							
リセット後	0	0	0	1	0	0	0	0
機能	マスタ/スレーブ選択 モニター 0:スレーブ 1:マスタ	送信/受信 選択 モニター 0:受信 1:送信	I ² Cバス 状態 モニター 0:バス フリー 1:バス ビジー	INTSBIIn 割り込み 要求 モニター 0:割り込み 要求発生 状態 1:割り込み 要求解除 状態	アービトレー ション ロスト検出 0:－ 1:検出	スレーブ アドレス 一致検出 0:－ 1:検出 (ゼネラル コール時 もセット されます)	ゼネラル コール 検出 0:－ 1:検出	最終受信 ビット モニター 0:“0” 1:“1”
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると“0”が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると“0”が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると“0”が読めます							

(注) このレジスタをライトすると、SBI0CR2として機能します。

図 12-6 I²C バスモード関係のレジスタ

- <Bit 0><LRB> : 最終受信ビットモニター
- <Bit 1><ADO> : ゼネラルコール検出
- <Bit 2><AAS> : スレーブアドレス一致検出
- <Bit 3><AL> : アービトレーションロスト検出
- <Bit 4><PIN> : INTSBIIn 割り込み要求モニター
- <Bit 5><BB> : I²C 状態モニター
- <Bit 6><TRX> : 送信/受信選択モニター
- <Bit 7><MST> : マスタスレーブ選択モニター

シリアルバスインタフェースポーレートレジスタ 0

SBIxBR0

	7	6	5	4	3	2	1	0
bit Symbol	I2SBI							
Read/Write	R	R/W	R					R/W
リセット後	1	0	1					0
機能	リードすると”1”が読めます	IDLE 0: 停止 1: 動作	リードすると”1”が読めます					必ず”0”をライトしてください。
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると”0”が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると”0”が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると”0”が読めます							

<Bit 6><I2SBI0> : IDLE モード時の動作

シリアルバスインタフェースデータバッファレジスタ

SBIxDBR

	7	6	5	4	3	2	1	0
bit Symbol	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Read/Write	R (受信)/W (送信)							
リセット後	0							
機能	受信データ/送信データ							
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると”0”が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると”0”が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると”0”が読めます							

(注1) 送信データを書き込み時には、データを MSB (ビット 7) 側につめてライトしてください。また、受信データは LSB 側に格納されます。

(注2) SBIxDBR は書き込み用のバッファと読み出し用のバッファを個別に持っているため、書き込んだデータを読み出すことはできません。したがって、ビット操作などのリードモディファイライト命令は使用できません。

I²C バスアドレスレジスタ

SBIxI2CAR

	7	6	5	4	3	2	1	0	
bit Symbol	SA6	SA5	SA4	SA3	SA2	SA1	SA0	ALS	
Read/Write	R/W								
リセット後	0	0	0	0	0	0	0	0	
機能	スレーブデバイスとして動作するときのスレーブアドレスの設定							アドレス認識モードの指定	
	15	14	13	12	11	10	9	8	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると"0"が読めます								
	23	22	21	20	19	18	17	16	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると"0"が読めます								
	31	30	29	28	27	26	25	24	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると"0"が読めます								

<Bit 0><ALS> :アドレス認識モードの指定。

(注 1) I²C バスアドレスレジスタ SBIxI2CAR のビット 0<ALS>はフリーデータフォーマット使用時以外は必ず"0"に設定してください。
 "1"に設定した場合にはフリーデータフォーマットとして動作し、マスタ時は送信に、スレーブ時は受信に転送方向が固定されます。

(注 2) スレーブモード時に SBIxI2CAR を"0x00"に設定しないでください。("0x00"に設定した場合、スレーブモードでI2Cバス規格のSTARTバイト("0x01")を受信した時にスレーブアドレスが一致したと判断します。)

図 12-7 I²C バスモード関係のレジスタ

12.5 I2C バスモード時の制御

12.5.1 アクノリッジメントモードの指定

SBIxCR1 <ACK> を “1” にセットしておくで、アクノリッジメントモードとして動作します。マスタのときには、アクノリッジ信号のためのクロックを 1 クロック付加します。トランスミッタモードのときには、このクロックの期間中、SDA 端子を開放し、レシーバからのアクノリッジ信号を受信できる状態にします。レシーバモードのときにはこのクロック期間中、SDA 端子を “L” レベルに引き、アクノリッジ信号を発生します。

<ACK> を “0” に設定しておくで、非アクノリッジメントモードとして動作し、マスタのときにアクノリッジ信号のためのクロックを発生しません。

12.5.2 転送ビット数の選択

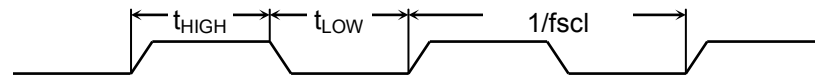
SBIxCR1 <BC[2:0]> により、次に送受信するデータのビット数を選択します。

<BC[2:0]> はスタートコンディションにより “000” にされるため、スレーブアドレス、方向ビットの転送はかならず 8 ビットで行われます。それ以外のときは <BC[2:0]> は一度設定された値を保持します。

12.5.3 シリアルクロック

① クロックソース

SBIxCR1 <SCK[2:0]> で、マスタモード時に SCL 端子から出力されるシリアルクロックの最大転送周波数を選択します。



$$t_{LOW} = 2^{n-1}/f_{sys} + 58/f_{sys}$$

$$t_{HIGH} = 2^{n-1}/f_{sys} + 14/f_{sys}$$

$$f_{SCL} = 1/(t_{LOW} + t_{HIGH})$$

$$= \frac{f_{sys}}{2^n + 72}$$

SBIxCR1 <SCK[2:0]>	n
000	5
001	6
010	7
011	8
100	9
101	10
110	11

図 12-8 クロックソース

通信規格上、標準モード/高速モードの最高速は 100KHz/400KHz です。内部 SCL クロックの周波数の設定は、使用される f_{sys} と上記計算式にて設定されますのでご注意ください。

② クロック同期化

I²C バスでは、端子の構造上バスをワイヤードアンドで駆動させるため、クロックラインを最初に “L” レベルに引いたマスタが、“H” レベルを出力しているマスタのクロックを無効にします。このため、“H” レベルを出力しているマスタは、これを検出し対応する必要があります。

クロック同期化機能をもっており、バス上に複数のマスタが存在する場合でも、正常に転送が行われます。

クロック同期の手順を、バス上に2つのマスタが同時に存在した場合を例に挙げて以下に示します。

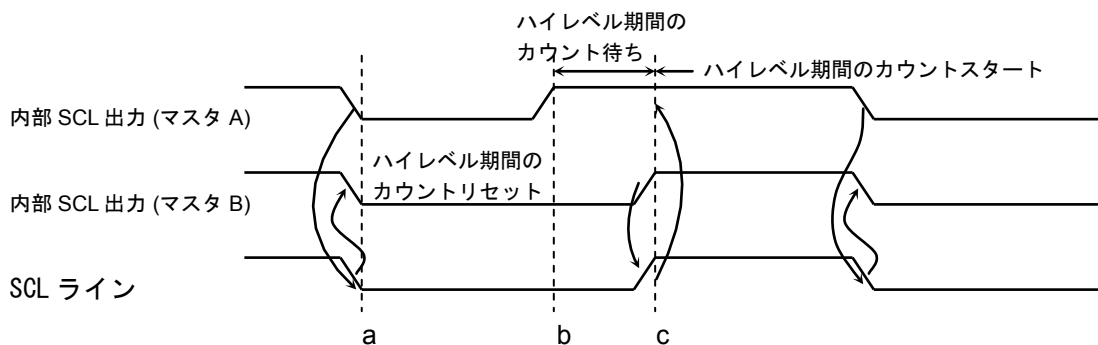


図 12-9 クロック同期化の例

a 点でマスタ A が内部 SCL 出力を “L” レベルに引くことで、バスの SCL ラインは “L” レベルになります。マスタ B はこれを検出し、マスタ B の “H” レベル期間のカウントをリセットし、内部 SCL 出力を “L” レベルに引きます。

b 点でマスタ A は “L” レベル期間のカウントを終わり、内部 SCL 出力を “H” レベルにします。しかし、マスタ B がバスの SCL ラインを “L” レベルに保持し続けているので、マスタ A は “H” レベル期間のカウントを止めます。マスタ A は、c 点でマスタ B が内部 SCL 出力を “H” レベルにし、バスの SCL ラインが “H” レベルになったことを検出後、“H” レベル期間のカウントを始めます。

以上のようにバス上のクロックは、バスに接続されているマスタの中で最も短い “H” レベル期間をもつマスタと最も長い “L” レベル期間をもつマスタによって決定されます。

12.5.4 スレーブアドレスとアドレス認識モードの設定

スレーブデバイスとして動作させるときは、SBIxI2CAR にスレーブアドレス <SA[6:0]> と <ALS> を設定します。<ALS> に “0” を設定すると、アドレス認識モードになります。

12.5.5 マスタ/スレーブの選択

SBIxCR2<MST> を “1” に設定すると、マスタデバイスとして動作します。

<MST> を “0” に設定すると、スレーブデバイスとして動作します。<MST> はバス上のストップコンディションの検出、またはアービトレーションロストの検出で、ハードウェアにより “0” にされます。

12.5.6 トランスミッタ/レシーバの選択

SBIxCR2 <TRX> を “1” に設定すると、トランスミッタとして動作し、<TRX> を “0” に設定すると、レシーバとして動作します。

スレーブモード時は

- アドレッシングフォーマットのデータ転送を行うとき、
- 受信したスレーブアドレスが I2CCR にセットした値と同じとき、
- ゼネラルコールを受信したとき、(スタートコンディション後の 8 ビットのデータがすべて “0”)

ハードウェアによりマスタデバイスから送られてくる方向ビット (R/\bar{W}) が “1” の場合、<TRX> は “1” にセットされ、“0” の場合、<TRX> は “0” にされます。

マスタモード時は、スレーブデバイスからアクリッジが返ってくると、ハードウェアにより、送信した方向ビットが “1” の場合、<TRX> は “0” に、方向ビットが “0” の場合、<TRX> は “1” に変化します。アクリッジが返ってこないときは、以前の状態を保ちます。

<TRX> はバス上のストップコンディションの検出、またはアービトレーションロストの検出で、ハードウェアにより “0” にされます。

12.5.7 スタート/ストップコンディションの発生

SBIxSR<BB> が “0” のときに、SBIxCR2 <MST, TRX, BB, PIN> に “1” を書き込むと、バス上にスタートコンディションと、8 ビットのデータが出力されます。あらかじめ、<ACK> に “1” を設定してください。

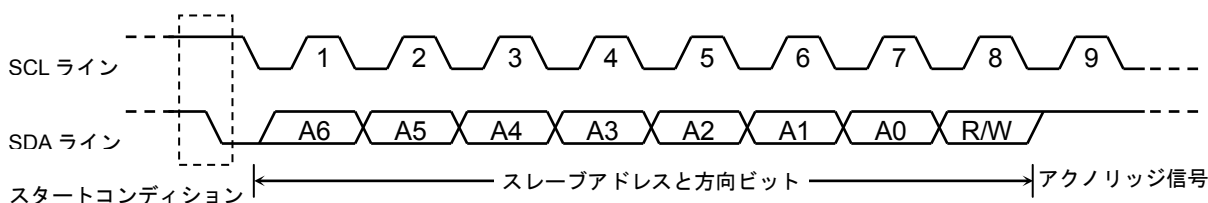


図 12-10 スタートコンディションの発生とスレーブアドレスの発生

<BB> = “1” のときに、<MST, TRX, PIN> に “1”、<BB> に “0” を書き込むと、バス上にストップコンディション出力のシーケンスを開始します。バス上にストップコンディションが発生するまで、<MST, TRX, BB, PIN> の内容を書き替えないでください。

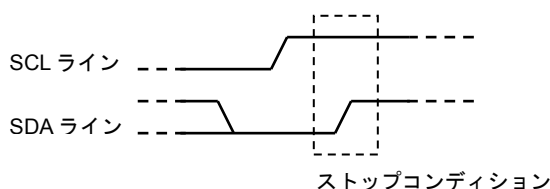


図 12-11 ストップコンディションの発生

また、SBIxSR<BB> を読み出すことで、バスの状態を知ることができます。<BB> は、バス上のスタートコンディションを検出すると “1” にセットされ (バスビジー状態)、ストップコンディションを検出すると “0” にされます (バスフリー状態)。

12.5.8 割り込みサービス要求と解除

マスタモードの時、〈BC〉と〈ACK〉によって設定されたクロック数の転送が終了すると、シリアルバスインタフェース割り込み要求 (INTSBIx) が発生します。

スレーブモードの場合は、以下のときに INTSBIx が発生します。

- ・受信したスレーブアドレスが SBIxI2CAR〈SA[6:0]〉に設定されたスレーブアドレスとアクノリッジ信号出力後
- ・ゼネラルコールを受信した時のアクノリッジ信号出力後
- ・スレーブアドレス一致、またはゼネラルコール受信後におけるデータ転送終了時

アドレス認識モード (〈ALS〉 = “0”) では、受信したスレーブアドレスが SBIxI2CAR にセットした値と同じとき、またはゼネラルコール (スタートコンディション後の 8 ビットのデータがすべて “0”) を受信したときに、〈PIN〉が “0” にされます。プログラムで SBIxCR2 〈PIN〉に “1” を書き込むと “1” にセットされますが、“0” を書き込んでも “0” にクリアされません。

シリアルバスインタフェース割り込み要求 (INTSBI0) が発生すると、SBIxCR2 〈PIN〉が “0” にされます。〈PIN〉が “0” の間、SCL ラインを “L” レベルに引きます。

〈PIN〉は 1 ワードの送信または受信が終了すると “0” にされ、SBIxDBR にデータを書き込むか、SBIxDBR からデータを読み出すと “1” にセットされます。〈PIN〉が “1” にセットされてから、SCL ラインが開放されるまで、 t_{LOW} の時間がかかります。

12.5.9 シリアルバスインタフェースの動作モード

SBIxCR2 〈SBIM[1:0]〉でシリアルバスインタフェースの動作モードを設定します。I²C バスモードで使用するときは、〈SBIM[1:0]〉を “10” に設定します。ポートモードへの切り替えは、バスがフリーであることを確認してから行ってください。

12.5.10 アービトレーションロスト検出モニタ

I²C バスではマルチマスタ (1 つのバス上で同時に 2 つ以上のマスタが存在する) が可能なため、転送されるデータの内容を保証するためにバスのアービトレーション手順が必要となります。

バスビジーの状態のときにスタートコンディションを出力しようとした場合は SCL、SDA ラインには出力されずにアービトレーションロストが発生します。I²C バスではバスのアービトレーションに SDA ラインのデータを使用します。

アービトレーションの手順を、バス上に 2 つのマスタが同時に存在した場合を例に挙げて以下に示します。a 点のビットまでマスタ A、マスタ B とも同じデータを出力し、a 点でマスタ A が “L” レベルを出力、マスタ B が “H” レベルを出力すると、バスの SDA ラインはワイヤードアンドで駆動されるためにマスタ A によって “L” レベルに引かれます。b 点でバスの SCL ラインが立ち上がると、スレーブデバイスは SDA ラインデータ、すなわち、マスタ A のデータを取り込みます。このとき、マスタ B の出力したデータは無効になります。マスタ B のこの状態を “アービトレーションロスト” と呼びます、マスタ B は SDA 端子を開放し、他のマスタの出力するデータに影響を及ぼさないようにします。また、複数のマスタが 1 ワード目でまったく等しいデータを送信した場合、アービトレーションの手順は 2 ワード目以降も継続されます。

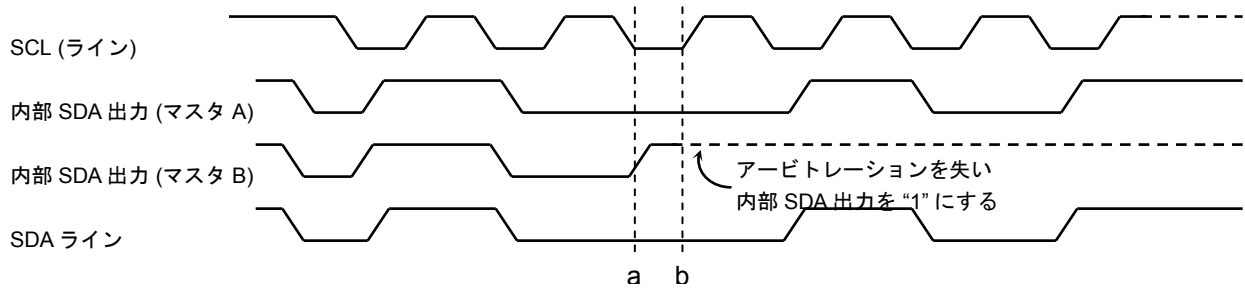


図 12-12 アービトレーションロスト

バスの SDA ラインのレベルと内部 SDA 出力のレベルの比較は、SCL ラインの立ち上がりで行います。この比較結果が不一致の場合アービトレーションロストになり、SBIxSR <AL> が “1” にセットされます。

<AL> が “1” にセットされると SBIxSR <MST, TRX> は “0” にされ、スレーブレシーバモードになります。<AL> は、SBIxDBR にデータを書き込むか、SBIxDBR からデータを読み込む、または SBIxCR2 にデータを書き込むと “0” にされます。

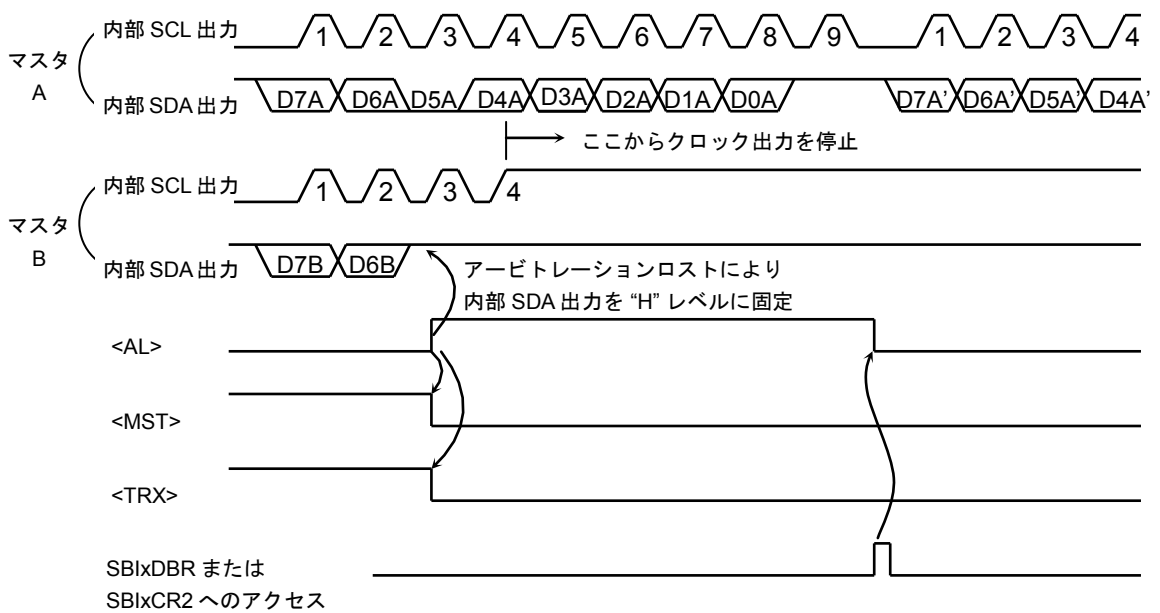


図 12-13 マスタ B の場合の例 (D7A = D7B, D6A = D6B)

12.5.11 スレーブアドレス一致検出モニタ

SBIxSR <AAS> は、スレーブモード時、アドレス認識モード (SBIxI2CAR <ALS> = “0”) のとき、ゼネラルコールまたは SBIxI2CAR にセットした値と同じスレーブアドレスを受信すると “1” にセットされます。<ALS> = “1” のときは、最初の 1 ワードが受信されると “1” にセットされます。<AAS> は SBIxDBR にデータを書き込むか、SBIxDBR からデータを読み出すと “0” にされます。

12.5.12 ゼネラルコール検出モニタ

SBIxSR <AD0> は、スレーブモード時、ゼネラルコール（スタートコンディション後の8ビットのデータがすべて“0”）を受信したとき“1”にセットされ、バス上のスタートコンディション、またはストップコンディションが検出されると“0”にされます。

12.5.13 最終受信ビットモニタ

SBIxSR <LRB> には、SCL ラインの立ち上がりで取り込まれた SDA ラインの値がセットされます。アクノリッジメントモードのとき、INTSBIx 割り込み要求発生直後に SBIxSR <LRB> を読み出すと、ACK 信号が読み出されます。

12.5.14 ソフトウェアリセット

シリアルバスインタフェース回路が、外部からのノイズによりロックした場合、ソフトウェアリセット機能を使い、シリアルバスインタフェース回路を初期化することができます。

SBIxCR2 <SWRST[1:0]> へ、最初に“10”、次に“01”をライトすると、シリアルバスインタフェース回路にリセット信号が入力され、回路が初期化されます。このとき、すべてのコントロールレジスタとステータスフラグはリセット直後の値となります。また、<SWRST> は、シリアルバスインタフェースを初期化すると、自動的に“0”にクリアされます。

(注) ソフトウェアリセットをかけると動作モード選択もリセットされ、I²C モードから PORT モードになります。

12.5.15 シリアルバスインタフェースデータバッファレジスタ (SBIxDBR)

SBIxDBR をリード/ライトすることで、受信データの読み出し/送信データの書き込みを行います。また、マスタモード時において、このレジスタにスレーブアドレスと方向ビットを設定後、スタートコンディションを発生します。

12.5.16 I2CBUS アドレスレジスタ (SBIxI2CAR)

SBIxI2CAR <SA[6:0]> は、スレーブデバイスとして動作する場合の、スレーブアドレスを設定するビットです。また、I2COAR <ALS> = “0” に設定すると、マスタデバイスから出力されるスレーブアドレスを認識し、データフォーマットはアドレッシングフォーマットとなります。<ALS> = “1” に設定すると、スレーブアドレスを認識せず、データフォーマットはフリーデータフォーマットとなります。

12.5.17 IDLE 設定レジスタ (SBIxBR0)

SBIxBR0 <I2SBI> は IDLE モードに遷移した際に動作の許可/禁止を設定するレジスタです。スタンバイモードに移る命令を実行する前に、あらかじめ設定してください。

12.6 I2C バスモード時のデータ転送手順

12.6.1 デバイスの初期化

最初に SBIxCR1<ACK, SCK[2:0]> を設定します。SBIxCR1 のビット 7～5 には、“0” を書き込んでください。

次に SBIxI2CAR にスレーブアドレス <SA[6:0]> と <ALS> (アドレッシングフォーマット時、<ALS> = “0”) を設定します。

それから、SBIxCR2 <MST, TRX, BB> に “0”、<PIN> に “1”、<SBIM[1:0]> に “10”、ビット 1, 0 に “0” を書き込み、初期状態をスレーブレシーバモードにします。

	7	6	5	4	3	2	1	0	
SBIxCR1	←	0	0	0	X	0	X	X	ACK および SCL クロックの設定をします。
SBIxI2CAR	←	X	X	X	X	X	X	X	スレーブアドレスおよびアドレス認識モードの設定をします。
SBIxCR2	←	0	0	0	1	1	0	0	スレーブレシーバモードにします。

(注) X: Don't care

12.6.2 スタートコンディション、スレーブアドレスの発生

① マスタモードの場合

マスタモード時は、スタートコンディションとスレーブアドレスを、次の手順で発生します。

はじめに、バスフリー状態 (<BB> = “0”) を確認します。次に、SBIxCR1 <ACK> に “1” を書き込んで、アクノリッジメントモードに設定します。また、SBIxDBR に、送信するスレーブアドレスと方向ビットのデータを書き込みます。

<BB> = “0” の状態で、SBIxCR2 <MST, TRX, BB, PIN> に “1111” を書き込むと、バス上にスタートコンディションが発生します。スタートコンディションの発生に次いで、SCL 端子から 9 クロックを出力します。最初の 8 クロックで、SBIxDBR に設定したスレーブアドレスと方向ビットを出力します。9 クロック目で、SDA ラインを解放し、スレーブデバイスからのアクノリッジ信号を受信します。

9 クロック目の立ち下がり、INTSBIx 割り込み要求が発生し、<PIN> = “0” にされます。マスタモード時は、<PIN> = “0” の間 SCL ラインを “L” レベルにひきます。また、スレーブデバイスからのアクノリッジ信号が返ってきたときのみ、INTSBIx 割り込み要求の発生により、送信した方向ビットに合わせて <TRX> は変化します。

メインルーチンでの設定

	7	6	5	4	3	2	1	0	
Reg.	←	SBIxSR							
Reg.	←	Reg. e 0x20							
if Reg.	≠	0x00							バスがフリー状態になるまで確認します。
Then									
SBIxCR1	←	X	X	X	1	0	X	X	アクノリッジメントモードに設定します。
SBIxDBR1	←	X	X	X	X	X	X	X	目的のスレーブのスレーブアドレスと方向をセットします。
SBIxCR2	←	1	1	1	1	1	0	0	スタートコンディションの発生を行います。

INTSBI0 割り込みルーチンでの処理例

割り込み要求クリア
処理
割り込み終了

② スレーブモードの場合

スレーブモードの場合は、スタートコンディションとスレーブアドレスを受信します。

マスタデバイスからのスタートコンディションを受信した後、SCL ラインの最初の 8 クロックで、マスタデバイスからのスレーブアドレスと方向ビットを受信します。ゼネラルコール、または SBIxI2CAR に設定されたスレーブアドレスと同一のアドレスを受信したとき、9 クロック目で SDA ラインを “L” レベルにひき、アクノリッジ信号を出力します。

9 クロック目の立ち下がりで、INTSBIx 割り込み要求が発生し、〈PIN〉 = “0” にされます。スレーブモード時は、〈PIN〉 = “0” の間 SCL ラインを “L” レベルにひきます。

(注) DMA 転送を使用する場合は

- ・ マスタスレーブが 1 対 1
- ・ 送信または受信が連続して可能

のときにのみ可能です。

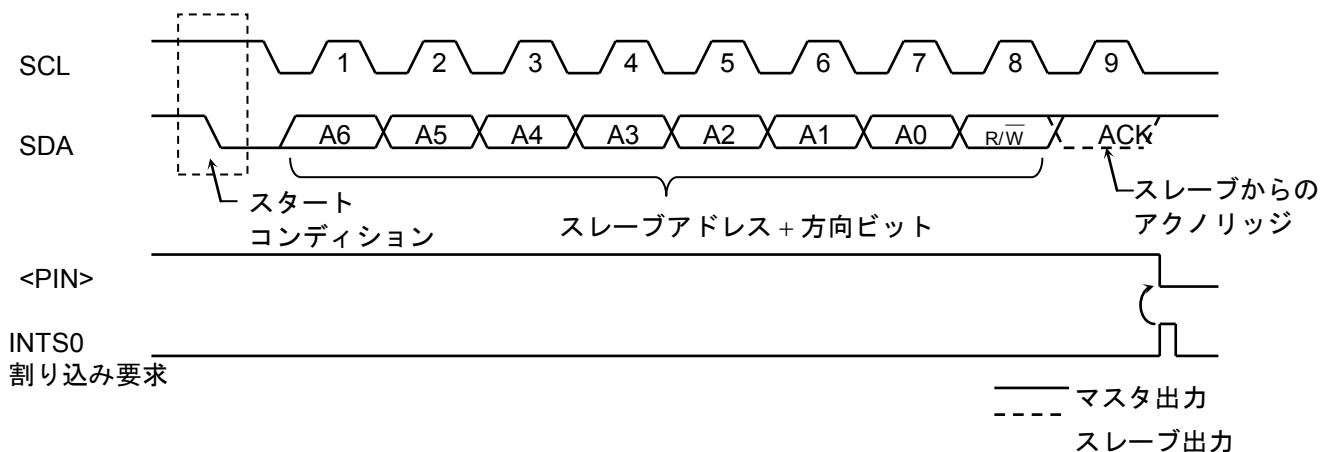


図 12-14 スタートコンディションとスレーブアドレスの発生

12.6.3 1 ワードのデータ転送

1 ワード転送終了の INTSBIx 割り込みの処理で 〈MST〉 をテストし、マスタモード/スレーブモードの判断をします。

① マスタモードの場合 (〈MST〉 = “1”)

〈TRX〉をテストし、トランスマッタ/レシーバの判断をします。

トランスマッタモードの場合 (〈TRX〉 = “1”)

〈LRB〉 をテストします。〈LRB〉 が “1” のとき、レシーバはデータを要求していないのでストップコンディションを発生する処理 (後記参照) を行ってデータ転送を終了します。

〈LRB〉 が “0” のときレシーバが次のデータを要求しています。次に転送するデータのビット数が 8 ビットるとき SBIxDBR に転送データを書き込みます。8 ビット以外ときは 〈BC[2:0]〉、〈ACK〉 を設定し、転送データを SBIxDBR に書き込みます。データを書き込むと 〈PIN〉 が “1” になり SCL 端子から次の 1 ワードデータ転送用のシリアルクロックが発生され、SDA 端子から 1 ワードのデータが転送されます。転送終了後 INTSBIx 割り込み要求が発生し、〈PIN〉 が “0” になり SCL 端子を “L” レベルに引きます。複数ワードの転送が必要な場合は上記 〈LRB〉 のテストから繰り返します。

INTSB_ix 割り込み

```

if MST = 0
Then スレーブモード時の処理へ移行
if TRX = 0
Then レシーバモード時の処理へ移行
if LRB = 0
Then ストップコンディションを発生する処理へ移行
SBIxCR1 ← X X X X 0 X X X      転送ビット数および ACK を設定します。
SBIxDBR ← X X X X X X X X      転送データを書き込みます。
割り込み処理終了
(注) X: Don't care
    
```

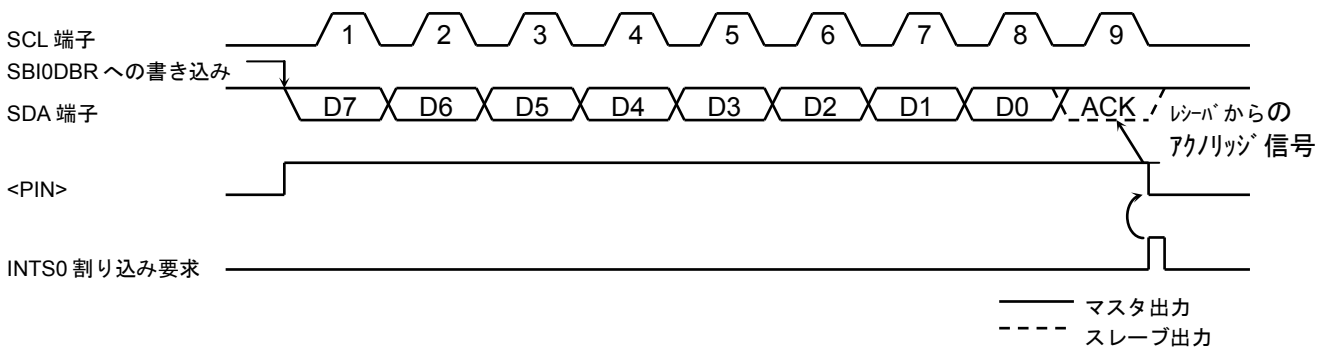


図 12-15 <BC[2:0]> = “000”, <ACK> = “1” の場合 (トランスミッタモード)

レシーバモードの場合 (<TRX> = “0”)

次に転送するデータのビット数が 8 ビットのときは SBIxDBR に転送データを書き込みます。8 ビット以外のときは <BC[2:0]>、<ACK> を設定し、SCL ラインを解放するために SBIxDBR から受信データを読み出します (スレーブアドレス送信直後のリードデータは不定です)。データを読み出すと <PIN> は “1” になり、次の 1 ワードデータ転送用のシリアルクロックを SCL 端子に出力します。最後のビットでアクノリッジ信号の “L” レベルのタイミングで “0” を SDA 端子に出力します。

その後、INTSB_ix 割り込み要求が発生し、<PIN> が “0” になり SCL 端子を “L” レベルに引きます。SBIxDBR から受信データを読み出すたびに 1 ワードの転送クロックとアクノリッジを出力します。

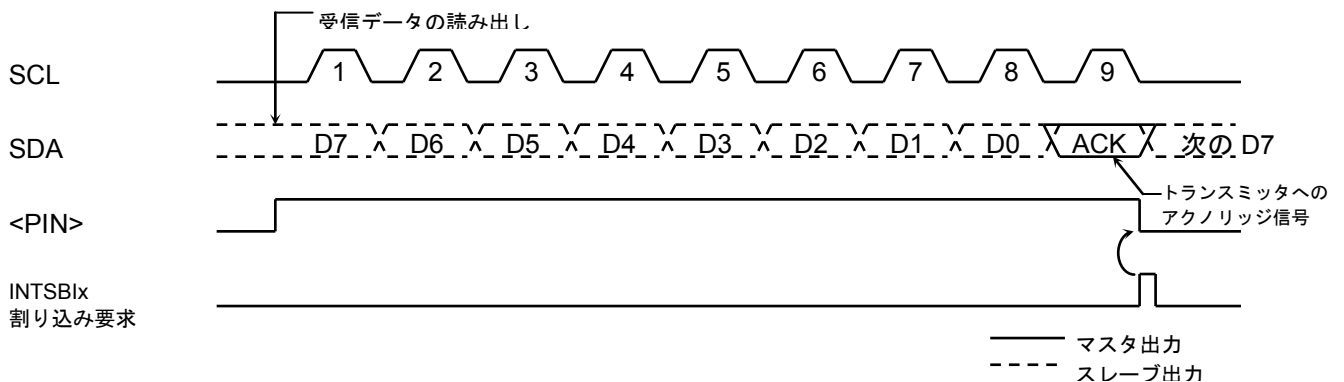


図 12-16 <BC[2:0]> = “000”, <ACK> = “1” のときの例 (レシーバモード)

トランスミッタに対してデータの送信を終了させるときは、最後に受信したいデータの1ワード手前のデータを読み出す前に <ACK> を “0” にします。これにより、最終データのアクノリッジのためのクロックは発生されなくなります。転送終了の割り込み要求発生後の処理で、<BC[2:0]> = “001” に設定し、データを読み出すと、1ビット転送のためのクロックを発生します。このときマスタはレシーバなのでバスの SDA ラインは “H” レベルを保ちます。トランスミッタは ACK 信号としてこの “H” レベルを受信するので、レシーバはトランスミッタへ送信終了を知らせることができます。

この1ビット転送の受信終了割り込み要求後の処理で、ストップコンディションを発生させ、データ転送を終了させます。

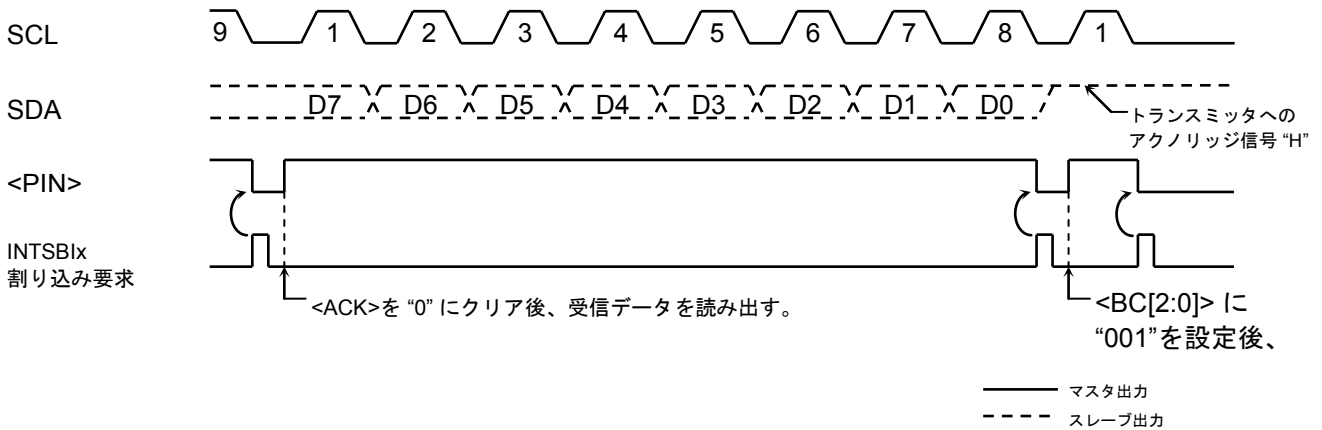


図 12-17 マスタレシーバモード時、データの送信を終了させるときの処理

例：データを N 回受信する場合

INTSBIx 割り込み（データ送信後）

```

          7 6 5 4 3 2 1 0
SBIxCR1 ← X X X X 0 X X X
Reg.     ← SBIxDBR
割り込み終了
    
```

受信データのビット数および ACK を設定します。
ダミーデータを取り込みます。

INTSBIx 割り込み（データ受信 1~ (N-2) 回目）

```

          7 6 5 4 3 2 1 0
Reg.     ← SBIxDBR
割り込み終了
    
```

1~ (N-2) 回目のデータを取り込みます。

INTSBIx 割り込み（データ受信 (N-1) 回目）

```

          7 6 5 4 3 2 1 0
SBIxCR1 ← X X X 0 0 X X X
Reg.     ← SBIxDBR
割り込み終了
    
```

アクノリッジ信号のクロックを発生しないようにします。
(N-1) 回目のデータを取り込みます。

INTSBIx 割り込み（データ受信 N 回目）

```

          7 6 5 4 3 2 1 0
SBIxCR1 ← 0 0 1 0 0 X X X
Reg.     ← SBIxDBR
割り込み終了
    
```

1ビット転送のためのクロックを発生します。
N 回目のデータを取り込みます。

INTSBIx 割り込み（データ受信後）

ストップコンディションを発生する処理 割り込み終了
データ転送を終了させます。

(注) X: Don't care

② スレーブモードの場合 (〈MST〉 = “0”)

スレーブモードのとき、マスタが送ったスレーブアドレス、またはゼネラルコールを受信したとき、もしくは、受信したスレーブアドレスが一致した後、またはゼネラルコールを受信した後のデータ転送終了時に INTSBIx 割り込み要求が発生します。また、マスタモードのとき、アービトレーションロストを検出するとスレーブモードとして動作し、アービトレーションロストを検出したワード転送の終了時に INTSBIx 割り込み要求が発生します。INTSBIx 割り込み要求が発生すると 〈PIN〉 が “0” にされ、SCL 端子を “L” レベルに引きます。SBIxDBR にデータを書き込む、SBIxDBR からデータを読み出す、または〈PIN〉に “1” を設定すると SCL 端子が t_{LOW} 後に開放されます。

スレーブモード時は、通常のスレーブモードとしての処理、またはアービトレーションロストを検出し、スレーブモードになったときの処理を行います。

SBIxSR 〈AL〉、〈TRX〉、〈AAS〉、〈ADO〉 をテストし、場合分けを行います。表 12-2 スレーブモード時の処理にスレーブモード時の状態と必要な処理を示します。

例：スレーブレシーバモード時スレーブアドレスが一致し、方向ビットが “1” の場合

INTSBIx 割り込み

```

if TRX = 0
Then その他処理へ移行
if AL = 1
Then その他処理へ移行
if AAS = 0
Then その他処理へ移行
SBIxCR1 ← X X X 1 0 X X X      送信ビット数を設定します。
SBIxDBR ← X X X X 0 X X X      送信データをセットします。

```

(注) X: Don't care

表 12-2 スレーブモード時の処理

<TRX>	<AL>	<AAS>	<AD0>	状 態	処 理
1	1	1	0	スレーブアドレス送信中にアービトレーションロストを検出し、他のマスタが送った方向ビットが“1”のスレーブアドレスを受信	1 ワードのビット数を <BC2:0> にセットし、送信するデータを SBIXDBR に書き込みます。
	0	1	0	スレーブレシーバモード時、マスタが送った方向ビットが“1”のスレーブアドレスを受信	
	0	0	0	スレーブトランスミッタモード時、1 ワードのデータの送信が終了	LRB をテストし、“1”にセットされていた場合、レシーバが次のデータを要求していないので <PIN> に“1”をセット、<TRX> を“0”にリセットしバスを開放します。<LRB> が“0”にリセットされていた場合、レシーバが次のデータを要求しているため1ワードのビット数を <BC[2:0]> にセットし、送信するデータを SBIXDBR に書き込みます。
0	1	1	1/0	スレーブアドレス送信中にアービトレーションロストを検出し、他のマスタが送った方向ビットが“0”のスレーブアドレス、またはゼネラルコールを受信	<PIN> を“1”にセットするために SBIXDBR を読み出します。(ダミー読み出し) または <PIN> に“1”を書き込みます。
		0	0	スレーブアドレスを送信中またはデータ送信中にアービトレーションロストを検出し、そのワードの転送が終了	
	0	1	1/0	スレーブレシーバモード時、マスタの送った方向ビットが“0”のスレーブアドレス、またはゼネラルコールを受信	
		0	1/0	スレーブレシーバモード時、1 ワードのデータの受信が終了	1ワードのビット数を <BC[2:0]> にセットし、受信データを SBIXDBR から読み出します。

12.6.4 ストップコンディションの発生

SBIxSR <BB> = “1” のときに、SBIxCR2 <MST, TRX, PIN> に “1”、<BB> に “0” を書き込むとバス上にストップコンディション出力のシーケンスを開始します。バス上にストップコンディションが発生するまで、<MST, TRX, BB, PIN> の内容を書き替えないでください。

なお、バスの SCL ラインが他のデバイスにより引かれていた場合、SCL ラインが開放されるのを待ち、SDA 端子が立ち上がり、ストップコンディションが発生します。

7 6 5 4 3 2 1 0
SBIxCR2 ← 1 1 0 1 1 0 0 0 ストップコンディションを発生させます。

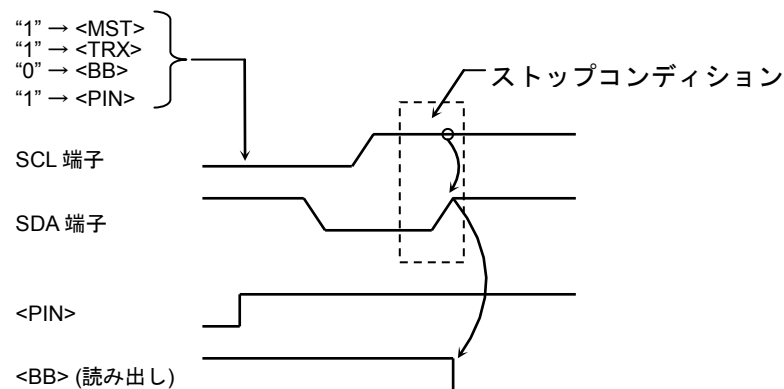


図 12-18 ストップコンディションの発生

12.6.5 再スタートの手順

再スタートはマスタデバイスがスレーブデバイスに対して、データ転送を終了させずに転送の方向を変化させるときに使用します。マスタモード時、再スタートを発生する場合の手順を以下に示します。

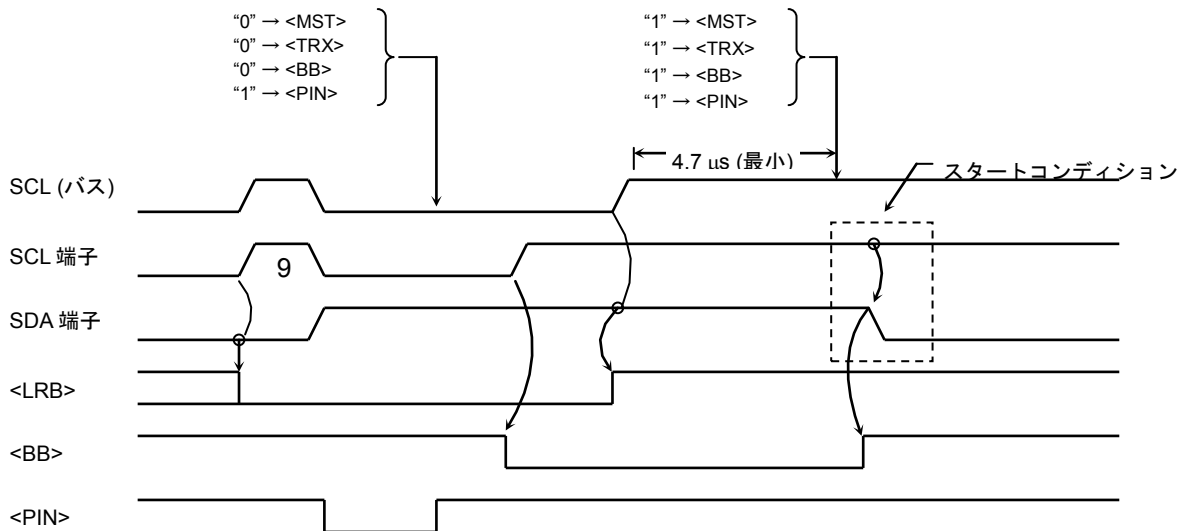
まず、SBIxCR2 <MST, TRX, BB> に “0”、<PIN> に “1” を書き込み、バスを開放します。このとき SDA 端子は “H” レベルを保ち、SCL 端子が開放され、バス上にストップコンディションが発生されないため、他のデバイスからみるとバスはビジー状態のままです。このあと、SBIxSR <BB> をテストして “0” になるまで待ち、SCL 端子が開放されたことを確認します。次に <LRB> をテストして “1” になるまで待ち、他のデバイスがバスの SCL ラインを “L” レベルに引いていないことを確認します。以上の手順によってバスがフリー状態になっていることを確認した後前記「12.6.2 スタートコンディション、スレーブアドレスの発生」の手順で、スタートコンディションの発生を行います。

なお、再スタート時のセットアップタイムを満たすために、バスフリーの確認からスタートの発生まで最低 4.7μs（標準モード時）のソフトウェアによる待ち時間が必要です。

```

┌─── 7 6 5 4 3 2 1 0
└── SBIxCR2 ← 0 0 0 1 1 0 0 0      バスを解放します。
┌── if SBISR<BB> ≠ 0                SCL 端子の解放を確認します。
└── Then
    ┌── if SBIxSR<LRB> ≠ 1           他のデバイスの SCL 端子 “L” レベルの確認を行います。
    └── Then
        4.7 μs Wait
        SBIxCR1 ← X X X 1 0 X X X   アクノリジメントモードに設定します。
        SBIxDBR ← X X X X X X X X   目的のスレーブのスレーブアドレスと方向をセットします。
        SBIxCR2 ← 1 1 1 1 1 0 0 0   スタートコンディションの発生を行います。
    
```

(注) X: Don't care



(注) <MST>="0" の状態の時に<MST>="0" をライトしないでください (再スタートできません)。

図 12-19 再スタートを発生する場合のタイミングチャート

12.7 クロック同期式 8 ビット SIO モード時の制御

シリアルバスインタフェースをクロック同期式 8 ビット SIO モードで使用するときの制御、および動作状態のモニタは、以下のレジスタで行います。

シリアルバスインタフェース制御レジスタ 0

		7	6	5	4	3	2	1	0
SBIXCR0	bit Symbol	SBIEN							
	Read/Write	R/W			R				
	リセット後	0			0				
	機能	SBI 動作 0: 禁止 1: 許可		リードすると"0"が読めます					
		15	14	13	12	11	10	9	8
bit Symbol									
Read/Write					R				
リセット後					0				
機能		リードすると"0"が読めます							
		23	22	21	20	19	18	17	16
bit Symbol									
Read/Write					R				
リセット後					0				
機能		リードすると"0"が読めます							
		31	30	29	28	27	26	25	24
bit Symbol									
Read/Write					R				
リセット後					0				
機能		リードすると"0"が読めます							

<SBIEN> : SIO (SBI) を使用する場合は、SIO (SBI) モジュールの各レジスタを設定する前に SIO (SBI) 動作許可 ("1") にしてください。

図 12-20 SIO モード関係のレジスタ

シリアルバスインタフェース制御レジスタ 1

SBiXCR1		7	6	5	4	3	2	1	0
	bit Symbol	SIOS	SIOINH	SIOM1	SIOM0		SCK2	SCK1	SCK0
	Read/Write	R/W				R	R/W		R/W
	リセット後	0	0	0	0	1	0	0	0
機能	転送の 開始/終了 0: 終了 1: 開始	転送の 強制停止 0: 転送継続 1: 強制停止	転送モードの選択 00: 送信モード 01: (予約) 10: 送受信モード 11: 受信モード		リー ド す る と ” 1 が 読 め ま す	シリアルクロック周波数の選択			
	15	14	13	12	11	10	9	8	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると”0”が読めます								
	23	22	21	20	19	18	17	16	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると”0”が読めます								
	31	30	29	28	27	26	25	24	
bit Symbol									
Read/Write	R								
リセット後	0								
機能	リードすると”0”が読めます								

シリアルクロック周波数の選択 <SCK[2:0]> @ライト

000	n = 3	2.5 MHz	$\left(\begin{array}{l} \text{システムクロック} : f_{\text{sys}} \\ \hspace{10em} (=40 \text{ MHz}) \\ \text{クロックギア} : f_c/1 \\ \text{周波数} = \frac{f_{\text{sys}}/2^n}{2} [\text{Hz}] \end{array} \right)$
001	n = 4	1.25 MHz	
010	n = 5	625 kHz	
011	n = 6	313 kHz	
100	n = 7	156 kHz	
101	n = 8	78 kHz	
110	n = 9	39 kHz	
111	—	外部クロック	

(注 1) <SCK[0]>ビットは、リセット後”1”が読みだされますが、SBiXCR2 レジスタにて SIO モードに設定後は”0”が初期値として読み出されます。ここでは初期状態で SIO モードに設定した後の値を「リセット後」欄に示します。なお、SBiXCR2 レジスタ、SBiXSR レジスタでも同様な記載をしています。

(注 2) 転送モード、シリアルクロックの設定時は、<SIOS> = “0”、および、<SIOINH> = “1” に設定してください。

図 12-21 SIO モード関係のレジスタ

シリアルバスインタフェースデータバッファレジスタ

SBlxDBR

	7	6	5	4	3	2	1	0
bit Symbol	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Read/Write	R (受信)/W (送信)							
リセット後	0							
機能	受信データ/送信データ							
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

シリアルバスインタフェース制御レジスタ 2

SBlxCR2

	7	6	5	4	3	2	1	0
bit Symbol					SBIM1	SBIM0		
Read/Write	R				W		R	
リセット後	1				0	0	1	
機能	リードすると"1"が読めます				シリアルバスインタフェースの動作モード選択 00: ポートモード 01: クロック同期式 8bit SIO モード 10: I ² C バスモード 11: (予約)		リードすると"1"が読めます	
	15	14	13	12	11	10	9	8
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							
	23	22	21	20	19	18	17	16
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							
	31	30	29	28	27	26	25	24
bit Symbol								
Read/Write	R							
リセット後	0							
機能	リードすると"0"が読めます							

図 12-22 SIO モード関係のレジスタ

シリアルバスインタフェースレジスタ

SBIXSR		7	6	5	4	3	2	1	0	
	bit Symbol					SIOF	SEF			
	Read/Write	R				R		R		
	リセット後	1				0	0	1		
	機能	リードすると"1"が読めます				シリアル 転送動作 状態モニタ 0: 転送終了 1: 転送中	シフト動作 状態モニタ 0: シフト 動作終了 1: シフト 転送中	リードすると"1"が読めます		
		15	14	13	12	11	10	9	8	
	bit Symbol									
	Read/Write	R								
	リセット後	0								
	機能	リードすると"0"が読めます								
		23	22	21	20	19	18	17	16	
	bit Symbol									
	Read/Write	R								
	リセット後	0								
	機能	リードすると"0"が読めます								
		31	30	29	28	27	26	25	24	
bit Symbol										
Read/Write	R									
リセット後	0									
機能	リードすると"0"が読めます									

シリアルバスインタフェースポーレートレジスタ 0

SBIXBR0		7	6	5	4	3	2	1	0	
	bit Symbol		I2SBI							
	Read/Write	R	R/W	R					W	
	リセット後	1	0	1					0	
	機能	リードすると"1"が読めます。	IDLE 0: 停止 1: 動作	リードすると"1"が読めます					必ず"0"をライトしてください。	
		15	14	13	12	11	10	9	8	
	bit Symbol									
	Read/Write	R								
	リセット後	0								
	機能	リードすると"0"が読めます								
		23	22	21	20	19	18	17	16	
	bit Symbol									
	Read/Write	R								
	リセット後	0								
	機能	リードすると"0"が読めます								
		31	30	29	28	27	26	25	24	
bit Symbol										
Read/Write	R									
リセット後	0									
機能	リードすると"0"が読めます									

図 12-23 SIO モード関係のレジスタ

12.7.1 シリアルクロック

①□ クロックソース

SBIxCR1 <SCK[2:0]> により、次の選択ができます。

内部クロック

内部クロックモードでは7種類の周波数が選択できます。シリアルクロックはSCK端子より外部に出力されます。なお、転送開始時SCK端子出力は“H”レベルになります。

プログラムでデータの書き込み（送信時）またはデータの読み出し（受信時）がこのシリアルクロックレートに追従できないときには、自動的にシリアルクロックを停止し、それらの処理が終了するまで次のシフト動作を待機させる自動ウェイト機能を有しています。

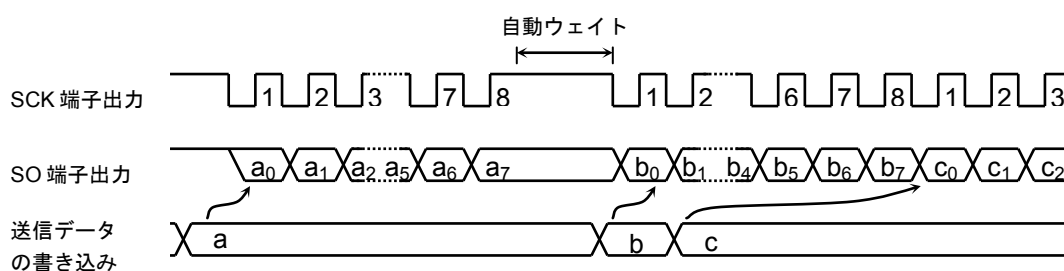


図 12-24 自動ウェイト機能

外部クロック（<SCK[2:0]> = “111”）

外部からSCK端子に供給されるクロックをシリアルクロックとして用います。なお、シフト動作を確実にを行うためには、シリアルクロックの“H”レベル、“L”レベル幅は下記に示すパルス幅が必要です。

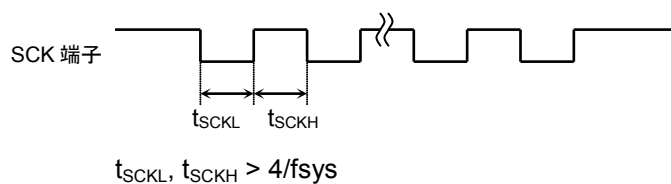


図 12-25 外部クロック入力時の最大転送周波数

② シフトエッジ

送信は前縁シフト，受信は後縁シフトになります。

前縁シフト

シリアルクロックの前縁 (SCK 端子入出力の立ち下がりエッジ) でデータをシフトします。

後縁シフト

シリアルクロックの後縁 (SCK 端子入出力の立ち上がりエッジ) でデータをシフトします。

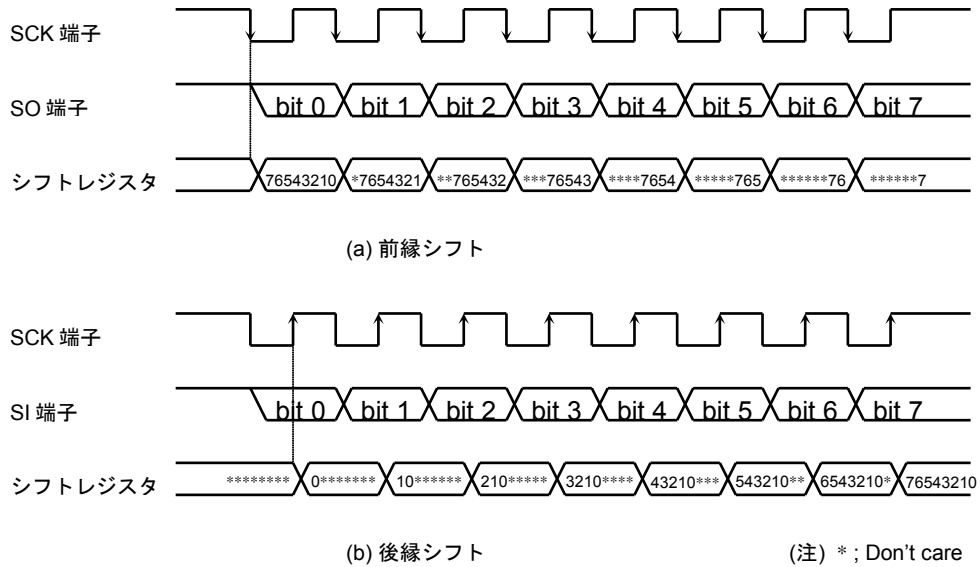


図 12-26 シフトエッジ

12.7.2 転送モード

SBIxCR1 <SIOM[1:0]> で、送信/受信/送受信モードを選択します。

① 8ビット送信モード

制御レジスタに送信モードをセットした後、送信データをSBIxDBRに書き込みます。

送信データの書き込み後、SBIxCR1 <SIOS> = “1” を書き込むことにより送信が開始されます。送信データは、SBIxDBRからシフトレジスタに移され、シリアルクロックに同期して最下位ビット (LSB) 側から SO 端子に出力されます。送信データがシフトレジスタに移されると、SBIxDBR が空になりますので、次の送信データを要求するINTSBIx (バッファエンプティ) 割り込み要求が発生します。

内部クロック動作の場合、8ビットのデータをすべて送信したあと、次のデータがセットされていないとシリアルクロックを停止して自動ウェイト動作を行います。次の送信データを書き込むと自動ウェイト動作は解除されます。

外部クロック動作の場合、次のデータのシフト動作に入る前に、SBIxDBRにデータが書き込まれている必要があります。したがって、転送速度は割り込み要求の発生から割り込みサービスプログラムにて、SBIxDBRにデータを書き込むまでの最大遅れ時間により決まります。

送信開始時、SBIxSR <SIOF> が “1” となってから SCK の立ち上がりエッジまでの間、前回送信したデータの最後のビットと同じ値が出力されます。

送信を終了させるには、INTSBIx 割り込みサービスプログラムで <SIOS> = “0” を書き込むか <SIOINH> = “1” を書き込みます。<SIOS> がクリアされると、データがすべて出力された時点で送信終了します。プログラムで送信の終了の確認は SBIxSR <SIOF>で行います。<SIOF> は送信の終了で “0” になります。<SIOINH> = “1” を書き込んだ場合はただちに送信を打ち切り、<SIOF> は “0” になります。

外部クロック動作では、次の送信データのシフト動作に入る前に <SIOS> を “0” にする必要があります。もしシフトアウトする前に <SIOS> が “0” にされなかった場合は、ダミーのデータの送信後、停止します。

	7 6 5 4 3 2 1 0	
SBIxCR1	← 0 1 0 0 0 X X X	送信モードをセットします。
SBIxDBR	← X X X X X X X X	送信データを書き込みます。
SBIxCR1	← 1 0 0 0 0 X X X	送信を開始します。

INTSBIx 割り込み

SBIxDBR	← X X X X X X X X	送信データを書き込みます。
---------	-------------------	---------------

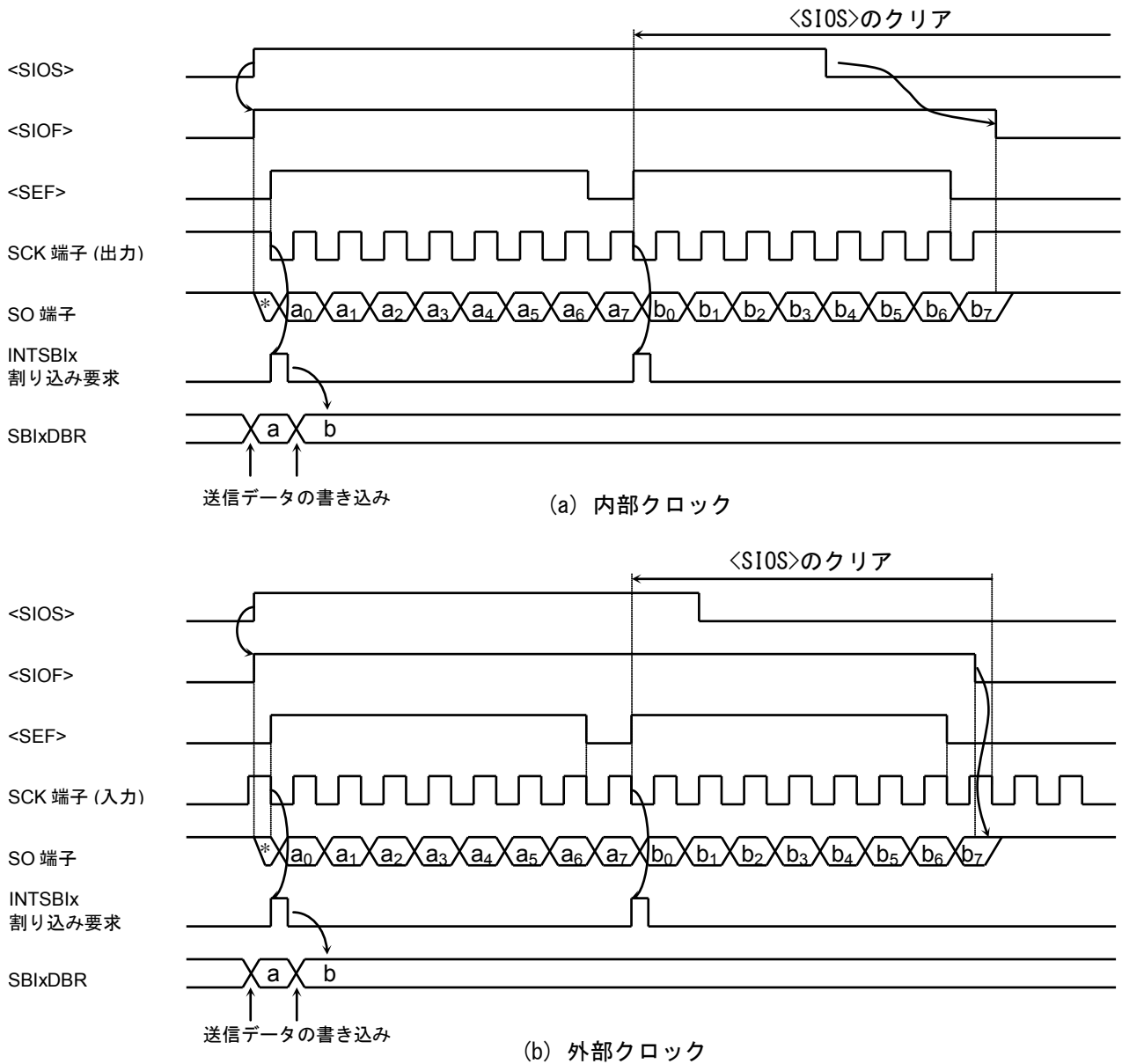


図 12-27 送信モード

例: <SIO> の送信終了指示 (外部クロックの場合)のプログラム例

	7 6 5 4 3 2 1 0	
→	if SBISR<SIOF> ≠ 0	転送の終了を確認します。
└	Then	
→	if SCK ≠ 1	ポートをモニタし、SCK 端子が “1” になったことを確認します。
└	Then	
	SBIXCR1 ← 0 0 0 0 0 1 1 1	<SIOS> = 0 を設定し送信を終了します。

② 8ビット受信モード

制御レジスタに受信モードをセットした後、SBIxCR1 <SIOS> = “1” を書き込むことにより受信可能となります。シリアルクロックに同期して、SI 端子より最下位ビット側からシフトレジスタへデータを取り込みます。8ビットのデータが取り込まれるとシフトレジスタから SBIxDBR に受信データが書き込まれ、受信データの読み出しを要求する INTSBIx (バッファフル) 割り込み要求が発生します。受信データは、割り込みサービスプログラムにて SBIxDBR から読み出します。

内部クロック動作の場合、受信データが SBIxDBR から読み出されるまでシリアルクロックを停止する自動ウェイト動作を行います。

外部クロック動作の場合は、シフト動作が外部から供給されるクロックに同期します。外部クロック動作での最大転送速度は、割り込み要求の発生から受信データの読み出しまでの最大遅れ時間により決まります。

受信を終了させるには INTSBIx 割り込みサービスプログラムで <SIOS> = “0” を書き込むか、<SIOINH> = “1” を書き込みます。<SIOS> がクリアされると、受信データが全ビット揃い、SBIxDBR への書き込みが完了した時点で受信が終了します。プログラムで受信の終了の確認は、SBIxSR <SIOF>で行います。<SIOF> は受信の終了で “0” にされます。受信終了の確認のあと最終受信データを読み出します。<SIOINH> = “1” を書き込んだ場合は、ただちに受信を打ち切り、<SIOF> は “0” になります (受信データは無効になりますので読み出す必要はありません)。

(注) 転送モードを切り替えると SBIxDBR の内容は保持されません。もし、転送モードの切り替えが必要な場合は、受信終了指示 (<SIOS> = “0” を書き込む) を行い、最終受信データを読み出したあとで切り替えてください。

```

          7 6 5 4 3 2 1 0
SBIxCR1 ← 0 1 1 1 0 X X X   受信モードをセットします。

```

```

SBIxCR1 ← 1 0 1 1 0 X X X   受信を開始します。

```

INTSBIx 割り込み

```

Reg.      ← SBIxDBR         受信データを取り込みます。

```

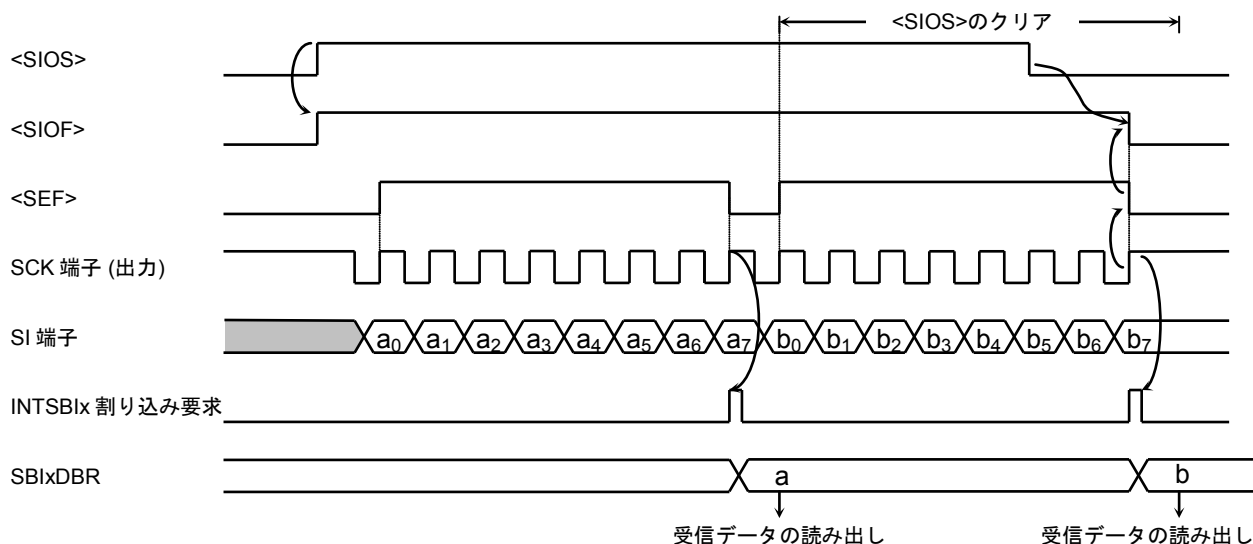


図 12-28 受信モード (例: 内部クロック)

③ 8ビット送受信モード

制御レジスタに送受信モードをセットした後、送信データを SBIxDBR に書き込みます。その後、SBIxCRI <SIOS> に “1” をセットすることにより送受信可能となります。最下位ビットから、シリアルクロックの立ち下がりで送信データが S0 端子から出力され、立ち上がりで受信データが SI 端子から取り込まれます。8ビットのデータが取り込まれると、シフトレジスタから SBIxDBR へ受信データが転送され、INTSBIx 割り込み要求が発生します。割り込みサービスプログラムにて受信データをデータバッファレジスタから読み出し、そのあと送信データを書き込みます。SBIxDBR は、送信/受信モードで兼用していますので、送信データは、かならず受信データを読み出してから書き込むようにしてください。

内部クロック動作の場合、受信データを読み出し、次の送信データを書き込むまで自動ウェイト動作を行います。

外部クロック動作の場合、シフト動作が外部から供給されるシリアルクロックに同期するので、次のシフト動作に入る前に受信データを読み出し、次の送信データを書き込む必要があります。外部クロック動作での最大転送速度は、割り込み要求の発生から受信データを読み出し、送信データを書き込むまでの最大遅れ時間により決まります。

送信開始時、<SIOF> が “1” となってから SCK の立ち下がりエッジまでの間、前回送信したデータの最後のビットと同じ値が出力されます。

送受信を終了させるには、INTSBIx 割り込みサービスプログラムで <SIOS> = “0” を書き込むか SBIxCRI <SIOINH> = “1” を書き込みます。<SIOS> がクリアされると、受信データが揃い、SBIxDBR への転送が完了した時点で送受信が終了します。プログラムで送受信の終了の確認は、SBIxSR <SIOF> で行います。<SIOF> は送受信の終了で “0” にされます。<SIOINH> をセットした場合は、ただちに送受信を打ち切り、<SIOF> は “0” にされます。

(注) 転送モードを切り替えると SBIxDBR の内容は保持されません。もし、転送モードの切り替えが必要な場合は、送受信終了指示 (<SIOS> = “0” を書き込む) を行い、最終受信データを読み出したあとで切り替えてください。

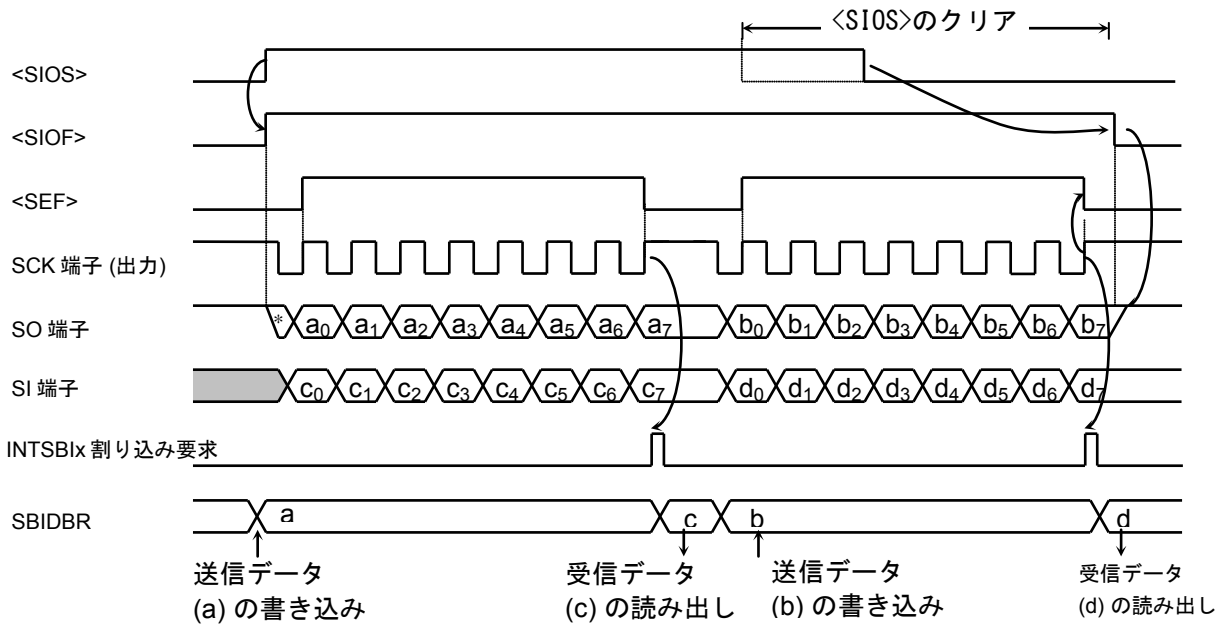


図 12-29 送受信モード (例: 内部クロック)

	7 6 5 4 3 2 1 0	
SBIxCR1	← 0 1 1 0 0 X X X	送受信モードをセットします。
SBIxDBR	← X X X X X X X X	送信データを書き込みます。
SBIxCR1	← 1 0 1 0 0 X X X	送受信を開始します。

INTSBIx 割り込み

Reg.	← SBIxDBR	受信データを取り込みます。
SBIxDBR	← X X X X X X X X	送信データを書き込みます。

13. CEC 機能

13.1 概要

CEC (Consumer Electronics Control) データの送受信を行います。
(HDMI 規格 Version 1.3a に準拠。)

13.1.1 受信

- 32KHz クロックでサンプリング
 - ・ ノイズキャンセル時間を調整可能
- 1byte ごとにデータを受信
 - ・ データサンプリングポイントを調整可能
 - ・ ディスティネーションアドレス不一致でも受信可能
- エラー検出
 - ・ 周期違反 (最小/最大)
 - ・ ACK 衝突
 - ・ 波形エラー

13.1.2 送信

- 1byte ごとにデータを送信
 - ・ バスフリーを自動判定し送信開始
- 送信波形の調整
 - ・ 立ち上がりタイミング、周期を調整可能
- エラー検出
 - ・ アービトレーションロスト
 - ・ ACK 違反

13.1.3 注意事項

受信動作に関し、以下の注意事項があります。

アドレス条件	設定	注意事項
ロジカルアドレス一致	—	送信側が EOM ビット “1” のブロックを送信せずに新たにメッセージをスタートビットから送信した場合、ACK ビットが最大周期違反と判定され割り込みが発生します。その後の受信は通常通り行われます。
ロジカルアドレス不一致	ロジカルアドレス不一致で受信する (CECR1<CECOTH>=“1”)	
ロジカルアドレス不一致	ロジカルアドレス不一致で受信しない (CECR1<CECOTH>=“0”)	送信側は必ず EOM= “1” のデータで送信を終了してください。EOM= “0” のデータで送信を終了した場合、その後の動作は保証できません。

13.2 レジスタ説明

13.2.1 レジスタ一覧

CEC 機能の制御レジスタとアドレスは以下のとおりです。

レジスタ名		アドレス
CEC イネーブルレジスタ	CECEN	0x4004_0300
ロジカルアドレスレジスタ	CECADD	0x4004_0304
ソフトウェアリセットレジスタ	CECRESET	0x4004_0308
受信イネーブルレジスタ	CECREN	0x4004_030C
受信バッファレジスタ	CECRBUF	0x4004_0310
受信コントロールレジスタ 1	CECRCR1	0x4004_0314
受信コントロールレジスタ 2	CECRCR2	0x4004_0318
受信コントロールレジスタ 3	CECRCR3	0x4004_031C
送信イネーブルレジスタ	CECTEN	0x4004_0320
送信バッファレジスタ	CECTBUF	0x4004_0324
送信コントロールレジスタ	CECTCR	0x4004_0328
受信割り込みステータスレジスタ	CECRSTAT	0x4004_032C
送信割り込みステータスレジスタ	CECTSTAT	0x4004_0330
CEC サンプリングクロック選択レジスタ	CECFSEL	0x4004_0334

13.2.2 CECイネーブルレジスタ《CECEN》

	7	6	5	4	3	2	1	0
bit Symbol	—						—	CECEN
Read/Write	R						R/W	R/W
リセット後	0						0	0
機能	リードすると“0”が読めます。						”1”をラ イトして ください。	CEC 動作 0:禁止 1:許可

<CECEN>: CEC 機能の動作を制御します。
 CEC 機能を使用する場合は、まず CEC 動作許可にしてください。
 動作禁止の状態では、イネーブルレジスタを除く CEC 機能のすべてのクロックが停止しま
 すので消費電力の低減が可能です。
 CEC 機能を一旦動作させた後に、動作禁止にした場合は各レジスタの設定は保持されます。

13.2.3 ロジカルアドレスレジスタ《CECADD》

	15	14	13	12	11	10	9	8
bit Symbol	CECADD15	CECADD14	CECADD13	CECADD12	CECADD11	CECADD10	CECADD9	CECADD8
Read/Write	R/W							
リセット後	0							
機能	ロジカル アドレス 15	ロジカル アドレス 14	ロジカル アドレス 13	ロジカル アドレス 12	ロジカル アドレス 11	ロジカル アドレス 10	ロジカル アドレス 9	ロジカル アドレス 8

	7	6	5	4	3	2	1	0
bit Symbol	CECADD7	CECADD6	CECADD5	CECADD4	CECADD3	CECADD2	CECADD1	CECADD0
Read/Write	R/W							
リセット後	0							
機能	ロジカル アドレス 7	ロジカル アドレス 6	ロジカル アドレス 5	ロジカル アドレス 4	ロジカル アドレス 3	ロジカル アドレス 2	ロジカル アドレス 1	ロジカル アドレス 0

<CECADD[15:0]>: 本デバイスに割り当てるロジカルアドレスを設定します。
 各ビットはそれぞれアドレスと対応しており、複数のアドレスを設定することが
 可能です。

(注) ブロードキャストメッセージはこのレジスタの設定によらず受信します。
 また、ロジカルアドレス15を設定した場合にはブロードキャストメッセージに対し論
 理“0”のACK応答を行います。

13.2.4 ソフトウェアリセットレジスタ 《CECRESET》

	7	6	5	4	3	2	1	0	
bit Symbol	—								CECRESET
Read/Write	R								W
リセット後	0								0
機能	リードすると“0”が読めます。								ソフトウェアリセット 0:禁止 1:許可 リードすると“0”が読めません。

<CECRESET>: CEC機能のすべての動作を停止し、レジスタを初期化します。
 このビットを“1”に設定すると、以下のように動作します。
 受信動作: 直ちに停止。受信データは破棄。
 送信動作: 直ちに停止。CEC信号出力も停止。
 レジスタ: CECEN以外のすべてのレジスタを初期化。

13.2.5 受信イネーブルレジスタ《CECREN》

	7	6	5	4	3	2	1	0
bit Symbol	—							CECREN
Read/Write	R							R/W
リセット後	0							不定
機 能	リードすると“0”が読めます。							受信制御 [ライト] 0:禁止 1:許可 [リード] 0:停止中 1:動作中

〈CECREN〉: CEC 機能の受信動作を制御します。
 このビットへライトすることにより、受信の許可／禁止の設定を行います。“1”をライトすることで受信待ち状態になります。
 このビットをリードすることで受信回路の状態をモニタすることができます。設定後にリードすることで設定が反映されたかどうかを確認できます。

(注1) 〈CECREN〉ビットは、受信コントロールレジスタ1、受信コントロールレジスタ2、受信コントロールレジスタ3を設定した後に許可してください。

(注2) 〈CECREN〉ビットへの設定が実際に回路に反映されるまでには若干の時間を要します。各種設定の変更を行う場合や、禁止にした後再び許可の設定を行う場合には、停止中であることを確認してから行ってください。

13.2.6 受信バッファレジスタ《CECRBUF》

	15	14	13	12	11	10	9	8
bit Symbol	—						CECACK	CECEOM
Read/Write	R						R	R
リセット後	0						0	0
機能	リードすると“0”が読めます。						ACK ビット	EOM ビット

	7	6	5	4	3	2	1	0
bit Symbol	CECRBUF7	CECRBUF6	CECRBUF5	CECRBUF4	CECRBUF3	CECRBUF2	CECRBUF1	CECRBUF0
Read/Write	R							
リセット後	0							
機能	受信データ							

<CECACK>: 受信した ACK ビットが読めます。

<CECEOM>: 受信した EOM ビットが読めます。

<CECRBUF[7:0]>: 受信した 1 バイト分のデータが読めます。ビット 7 が MSB です。

(注 1) このレジスタへの書き込み動作は無視されます。

(注 2) このレジスタは受信割り込み発生後なるべく早く読んでください。また、2 度目以降の読み出しデータの内容は保証しません。

13.2.7 受信コントロールレジスタ 1 《CECR1》

	31	30	29	28	27	26	25	24	
bit Symbol	—								CEACKDIS
Read/Write	R								R/W
リセット後	0								0
機能	リードすると“0”が読めます。								ACK 応答制御 0: 論理0 応答する 1: 論理0 応答しない
	23	22	21	20	19	18	17	16	
bit Symbol	—		CECHNC1	CECHNC0	—	CECLNC2	CECLNC1	CECLNC0	
Read/Write	R		R/W		R	R/W			
リセット後	0		0		0	0			
機能	リードすると“0”が読めます。		“1”検出ノイズキャンセル設定 00: サンプルング 1 回 01: サンプルング 2 回 10: サンプルング 3 回 11: サンプルング 4 回		リードすると“0”が読めます。		“0”検出ノイズキャンセル設定 000: サンプルング 1 回 001: サンプルング 2 回 010: サンプルング 3 回 011: サンプルング 4 回 100: — (Reserved) 101: — (Reserved) 110: — (Reserved) 111: — (Reserved)		
	15	14	13	12	11	10	9	8	
bit Symbol	—	CECMIN2	CECMIN1	CECMIN0	—	CECMAX2	CECMAX1	CECMAX0	
Read/Write	R	R/W			R	R/W			
リセット後	0	0			0	0			
機能	リードすると“0”が読めます。	最小周期違反検出時間 000: 2.05ms 001: 2.05ms+1cycle 010: 2.05ms+2cycle 011: 2.05ms+3cycle 100: 2.05ms-1cycle 101: 2.05ms-2cycle 110: 2.05ms-3cycle 111: 2.05ms-4cycle				リードすると“0”が読めます。	最大周期違反検出時間 000: 2.75ms 001: 2.75ms+1cycle 010: 2.75ms+2cycle 011: 2.75ms+3cycle 100: 2.75ms-1cycle 101: 2.75ms-2cycle 110: 2.75ms-3cycle 111: 2.75ms-4cycle		
	7	6	5	4	3	2	1	0	
bit Symbol	—	CECDAT2	CECDAT1	CECDAT0	CECTOUT1	CECTOUT0	CECRIHLD	CECOTH	
Read/Write	R	R/W			R/W		R/W	R/W	
リセット後	0	0			0		0	0	
機能	リードすると“0”が読めます。	データ 0/1 判別タイミング 000: 1.05ms 001: 1.05ms+2cycle 010: 1.05ms+4cycle 011: 1.05ms+6cycle 100: 1.05ms-2cycle 101: 1.05ms-4cycle 110: 1.05ms-6cycle 111: Reserved				タイムアウト判定時間 00: 1 bit 周期 01: 2 bit 周期 10: 3 bit 周期 11: Reserved	エラー割り込み保留 0: 保留しない 1: 保留する	ロジカルアドレス不一致時の動作 0: 受信しない 1: 受信する	

- <CECACKDIS>: ディスティネーションアドレスがロジカルアドレスレジスタに設定されたアドレスと一致する時に、データブロックに対して論理“0”のACK応答をするかどうかを設定します。
(ヘッダブロックに対しては、このビットの設定によらず、アドレスが一致すると論理“0”のACK応答を行います。)
- <CECHNC[1:0]>: “1”を検出する際のノイズキャンセル時間をサンプリングクロックサイクル単位で設定します。
設定されたサイクル数分の“1”がサンプリングされなければノイズとみなします。
- <CECLNC[2:0]>: “0”を検出する際のノイズキャンセル時間をサンプリングクロックサイクル単位で設定します。
設定されたサイクル数分の“0”がサンプリングされなければノイズとみなします。
- <CECMIN[2:0]>: 有効なビットとみなす最短の時間を設定します。
約 2.05ms を基準に、サンプリングクロックサイクル単位で-4 サイクルから+3 サイクルまで設定可能です。
1 ビットの周期が設定より短い場合は割り込みが発生し、CEC 信号へ約 3.6ms 間“0”を出力します。
- <CECMAX[2:0]>: 有効なビットとみなす最長の時間を設定します。
約 2.75ms を基準に、サンプリングクロックサイクル単位で-4 サイクルから+3 サイクルまで設定可能です。
1 ビットの周期が設定より長い場合は割り込みが発生します。
- <CECDAT[2:0]>: データの 0/1 判別を行うポイントを設定します。
約 1.05ms を基準に、サンプリングクロック 2 クロックサイクル単位で±6 サイクルまで設定可能です。
- <CECTOUT[1:0]>: タイムアウトとみなす時間を設定します。ビット周期単位で、1 ビットから 3 ビットまで設定可能です。
<CECRIHLD>が有効な場合のタイムアウト検出にこの設定が使用されます。
- <CECRIHLD>: 受信エラー割り込みを保留するかどうかを設定します。対象となる割り込みは、最大周期違反、バッファオーバーラン、波形エラーです。
この設定が“1”に設定されていると、エラー検出時点では割り込みは発生しません。エラー検出後 ACK ビットまでデータが継続した場合は ACK 応答を通常と反転の論理で行い、後続のビットがいずれかのタイミングで途切れた場合には<CECTOUT>の設定にしたがってタイムアウトと判定します。割り込みは ACK 応答後、またはタイムアウト判定後に発生します。
- <CECOTH>: ディスティネーションアドレスが、ロジカルアドレスレジスタに設定されたアドレスと異なる場合にもデータの受信を行うかどうかを設定します。

- (注 1) 送信動作でも、ACK 応答の受信の際にノイズキャンセル時間<CECHNC><CECLNC>、データ判別ポイント<CECDAT>の設定が使用されます。
- (注 2) 送受信動作中に設定が変更されると正しく送受信できない可能性があります。
設定を変更する場合には、CECREN レジスタ<CECREN>ビットで受信禁止の設定を行い、<CECREN>および CECTEN レジスタ<CECTEN>ビットをリードして停止中であることを確認後に変更することを推奨します。
- (注 3) ブロードキャストメッセージは、<CECOTH>の設定によらず受信します。
- (注 4) <CECLNC>は、必ず CECTCR<CECTCR>と同じ設定で使用してください。

13.2.8 受信コントロールレジスタ 2 《CECR2》

	15	14	13	12	11	10	9	8
bit Symbol	—	CECSWAV32	CECSWAV31	CECSWAV30	—	CECSWAV22	CECSWAV21	CECSWAV20
Read/Write	R	R/W			R	R/W		
リセット後	0	0			0	0		
機能	リードすると“0”が読めます。	スタートビット検出設定 3 000 : 4.7ms 001 : 4.7ms+1cycle 010 : 4.7ms+2cycle 011 : 4.7ms+3cycle 100 : 4.7ms+4cycle 101 : 4.7ms+5cycle 110 : 4.7ms+6cycle 111 : 4.7ms+7cycle			リードすると“0”が読めます。	スタートビット検出設定 2 000 : 4.3ms 001 : 4.3ms-1cycle 010 : 4.3ms-2cycle 011 : 4.3ms-3cycle 100 : 4.3ms-4cycle 101 : 4.3ms-5cycle 110 : 4.3ms-6cycle 111 : 4.3ms-7cycle		
	7	6	5	4	3	2	1	0
bit Symbol	—	CECSWAV12	CECSWAV11	CECSWAV10	—	CECSWAV02	CECSWAV01	CECSWAV00
Read/Write	R	R/W			R	R/W		
リセット後	0	0			0	0		
機能	リードすると“0”が読めます。	スタートビット検出設定 1 000 : 3.9ms 001 : 3.9ms+1cycle 010 : 3.9ms+2cycle 011 : 3.9ms+3cycle 100 : 3.9ms+4cycle 101 : 3.9ms+5cycle 110 : 3.9ms+6cycle 111 : 3.9ms+7cycle			リードすると“0”が読めます。	スタートビット検出設定 0 000 : 3.5ms 001 : 3.5ms-1cycle 010 : 3.5ms-2cycle 011 : 3.5ms-3cycle 100 : 3.5ms-4cycle 101 : 3.5ms-5cycle 110 : 3.5ms-6cycle 111 : 3.5ms-7cycle		

- <CECSWAV3 [2:0]>: スタートビット検出時の周期の条件を設定します。
- <CECSWAV2 [2:0]>: <CECSWAV3>で周期の最大値の条件を設定します。規定上の最大値 4.7ms に対し、サンプリングクロック単位で 0 から+7 サイクルまで設定可能です。
<CECSWAV2>で周期の最小値の条件を設定します。規定上の最小値 4.3ms に対し、サンプリングクロック単位で 0 から-7 サイクルまで設定可能です。
- <CECSWAV1 [2:0]>: スタートビット検出時の立ち上がりタイミングの条件を設定します。
- <CECSWAV0 [2:0]>: <CECSWAV1>で立ち上がりタイミングの最大値の条件を設定します。規定上の最大値 3.9ms に対し、サンプリングクロック単位で 0 から+7 サイクルまで設定可能です。
<CECSWAV0>で立ち上がりタイミングの最小値の条件を設定します。規定上の最小値 3.5ms に対し、サンプリングクロック単位で 0 から-7 サイクルまで設定可能です。

(注) 受信動作中に設定が変更されると正しく受信できない可能性があります。
設定を変更する場合には、CECREN レジスタ<CECREN>ビットで受信禁止の設定を行い、<CECREN>ビットをリードして停止中であることを確認後に変更することを推奨します。

13.2.9 受信コントロールレジスタ 3 《CECR3》

	23	22	21	20	19	18	17	16
bit Symbol	—	CECWAV32	CECWAV31	CECWAV30	—	CECWAV22	CECWAV21	CECWAV20
Read/Write	R	R/W			R	R/W		
リセット後	0	0			0	0		
機能	リードすると“0”が読めます。	波形確認設定3 000 : 1.7ms 001 : 1.7ms+1cycle 010 : 1.7ms+2cycle 011 : 1.7ms+3cycle 100 : 1.7ms+4cycle 101 : 1.7ms+5cycle 110 : 1.7ms+6cycle 111 : 1.7ms+7cycle			リードすると“0”が読めます。	波形確認設定2 000 : 1.3ms 001 : 1.3ms-1cycle 010 : 1.3ms-2cycle 011 : 1.3ms-3cycle 100 : 1.3ms-4cycle 101 : 1.3ms-5cycle 110 : 1.3ms-6cycle 111 : 1.3ms-7cycle		
	15	14	13	12	11	10	9	8
bit Symbol	—	CECWAV12	CECWAV11	CECWAV10	—	CECWAV02	CECWAV01	CECWAV00
Read/Write	R	R/W			R	R/W		
リセット後	0	0			0	0		
機能	リードすると“0”が読めます。	波形確認設定1 000 : 0.8ms 001 : 0.8ms+1cycle 010 : 0.8ms+2cycle 011 : 0.8ms+3cycle 100 : 0.8ms+4cycle 101 : 0.8ms+5cycle 110 : 0.8ms+6cycle 111 : 0.8ms+7cycle			リードすると“0”が読めます。	波形確認設定0 000 : 0.4ms 001 : 0.4ms-1cycle 010 : 0.4ms-2cycle 111 : 0.4ms-3cycle 100 : 0.4ms-4cycle 101 : 0.4ms-5cycle 110 : 0.4ms-6cycle 111 : 0.4ms-7cycle		
	7	6	5	4	3	2	1	0
bit Symbol	—							CECWAVEN
Read/Write	R							R/W
リセット後	0							0
機能	リードすると“0”が読めます。							波形エラー検出 1:許可 0:禁止

(注) 受信動作中に設定が変更されると正しく受信できない可能性があります。
設定を変更する場合には、CECREN レジスタ<CECREN>ビットで受信禁止の設定を行い、<CECREN>ビットをリードして停止中であることを確認後に変更することを推奨します。

- <CECWAV3 [2:0]>: <CECWAVEN>ビットが“1”に設定されている場合に有効になります。
論理“0”波形の立ち上がりタイミングより遅い場合にエラー検出を行うための設定です。規格上もっとも遅い立ち上がりタイミング 1.7ms に対し、サンプリングクロック単位で 0 から+7 サイクルまで設定可能です。
ビットのスタートポイントから<CECWAV3>の設定値までに立ち上がりが検出されなければエラーとなります。
- <CECWAV2 [2:0]>: <CECWAVEN>ビットが“1”に設定されている場合に有効になります。
<CECWAV1 [2:0]>: 論理“1”波形の立ち上がりタイミングより遅く、論理“0”波形の立ち上がりタイミングより早い場合にエラー検出を行うための設定です。
<CECWAV1>で論理“1”波形の、規格上もっとも遅い立ち上がりタイミング 0.8ms に対しサンプリングクロック単位で 0 から+7 サイクルまでの設定を行います。
<CECWAV2>で論理“0”波形の規格上もっとも早い立ち上がりタイミング 1.3ms に対し、サンプリングクロック単位で 0 から-7 サイクルまでの設定を行います。
<CECWAV2>と<CECWAV1>の設定値の間に立ち上がりを検出するとエラーとなります。
- <CECWAV0 [2:0]>: <CECWAVEN>ビットが“1”に設定されている場合に有効になります。
論理“1”波形の立ち上がりタイミングより早い場合にエラー検出を行うための設定です。規格上もっとも早い立ち上がりタイミング 0.4ms に対し、サンプリングクロック単位で 0 から-7 サイクルまで設定可能です。
ビットのスタートポイントから<CECWAV0>設定値の間に立ち上がりを検出するとエラーとなります。
- <CECWAVEN>: 受信データ波形が規格から外れたことを検出し、波形エラー割り込みを発生します。
許可の設定の場合、<CECWAV0><CECWAV1><CECWAV2><CECWAV3>の設定にしたがって波形エラーの検出を行います。

13.2.10 送信イネーブルレジスタ《CECTEN》

	7	6	5	4	3	2	1	0	
bit Symbol	—							CECTRANS	CECTEN
Read/Write	R							R	W
リセット後	0							0	不定
機能	リードすると“0”が読めます。							送信中 0: 送信していない 1: 送信中	送信制御 0: 禁止 1: 許可

〈CECTRANS〉: 送信中かどうかを示します。
 スタートビットの送信を開始すると“1”になり、送信終了割り込みまたはエラー割り込み発生で“0”になります。
 このビットに対する書き込み動作は無視されます。

〈CECTEN〉: GEC 機能の送信動作を制御します。
 このビットへライトすることにより送信の開始／禁止の設定を行います。“1”をライトすることで送信動作を開始します。送信終了割り込みまたはエラー割り込み発生で自動的に“0”にクリアされます。

(注1) 〈CECTEN〉ビットは、送信バッファレジスタ、送信コントロールレジスタを設定した後に開始の設定をしてください。

(注2) 各種設定の変更を行う場合は、停止中であることを確認してから行ってください。

13.2.11 送信バッファレジスタ《CECTBUF》

	15	14	13	12	11	10	9	8
bit Symbol	—							CECTEOM
Read/Write	R							R/W
リセット後	0							0
機 能	リードすると“0”が読めます。							EOM ビット
	7	6	5	4	3	2	1	0
bit Symbol	CECTBUF7	CECTBUF6	CECTBUF5	CECTBUF4	CECTBUF3	CECTBUF2	CECTBUF1	CECTBUF0
Read/Write	R/W							
リセット後	0							
機 能	送信データ							

<CECTEOM>: 送信する EOM ビットを設定します。

<CECTBUF[7:0]>: 送信する 1 バイト分のデータを設定します。ビット 7 が MSB です。

13.2.12 送信コントロールレジスタ《CECTCR》

	23	22	21	20	19	18	17	16
bit Symbol	—	CECSTRS2	CECSTRS1	CECSTRS0	—	CECSPRD2	CECSPRD1	CECSPRD0
Read/Write	R	R/W			R	R/W		
リセット後	0	0			0	0		
機能	リードすると“0”が読めません。	スタートビット波形立ち上がり時間調整 000: 基準値 001: 基準値-1cycle 010: 基準値-2cycle 011: 基準値-3cycle 100: 基準値-4cycle 101: 基準値-5cycle 110: 基準値-6cycle 111: 基準値-7cycle			リードすると“0”が読めません。	スタートビット波形周期時間調整 000: 基準値 001: 基準値-1cycle 010: 基準値-2cycle 011: 基準値-3cycle 100: 基準値-4cycle 101: 基準値-5cycle 110: 基準値-6cycle 111: 基準値-7cycle		
	15	14	13	12	11	10	9	8
bit Symbol	—	CECDTRS2	CECDTRS1	CECDTRS0	CECDPRD3	CECDPRD2	CECDPRD1	CECDPRD0
Read/Write	R	R/W			R/W			
リセット後	0	0			0			
機能	リードすると“0”が読めません。	データ波形立ち上がり時間調整 000: 基準値 001: 基準値-1cycle 010: 基準値-2cycle 011: 基準値-3cycle 100: — (Reserved) 101: — (Reserved) 110: — (Reserved) 111: — (Reserved)			データ波形周期時間調整 0000: 基準値 1000: 基準値-8cycle 0001: 基準値-1cycle 1001: 基準値-9cycle 0010: 基準値-2cycle 1010: 基準値-10cycle 0011: 基準値-3cycle 1011: 基準値-11cycle 0100: 基準値-4cycle 1100: 基準値-12cycle 0101: 基準値-5cycle 1101: 基準値-13cycle 0110: 基準値-6cycle 1110: 基準値-14cycle 0111: 基準値-7cycle 1111: 基準値-15cycle			
	7	6	5	4	3	2	1	0
bit Symbol	—			CEGBRD	CECFREE3	CECFREE2	CECFREE1	CECFREE0
Read/Write	R			R/W	R/W			
リセット後	0			0	0			
機能	リードすると“0”が読めません。			0: フロートキャスト送信設定 1: フロートキャスト送信でない	バスフリー待ち時間設定 0000 : 1 bit 周期 1000 : 9 bit 周期 0001 : 2 bit 周期 1001 : 10 bit 周期 0010 : 3 bit 周期 1010 : 11 bit 周期 0011 : 4 bit 周期 1011 : 12 bit 周期 0100 : 5 bit 周期 1100 : 13 bit 周期 0101 : 6 bit 周期 1101 : 14 bit 周期 0110 : 7 bit 周期 1110 : 15 bit 周期 0111 : 8 bit 周期 1111 : 16 bit 周期			

- <CECSTRS[2:0]>: スタートビットの立ち上がりタイミングの設定を行います。
約 3.7ms を基準に、サンプリングクロックサイクル単位で 0 サイクルから-7 サイクルまで設定可能です。
- <CECSPRD[2:0]>: スタートビットの周期の設定を行います。
約 4.5ms を基準に、サンプリングクロックサイクル単位で 0 サイクルから-7 サイクルまで設定可能です。
- <CECDTRS[2:0]>: データビットの立ち上がりタイミングの設定を行います。
約 0.6ms (論理 “1” の場合) または約 1.5ms (論理 “0” の場合) を基準に、サンプリングクロックサイクル単位で 0 サイクルから-7 サイクルまで設定可能です。
- <CECDPRD[3:0]>: データビットの周期の設定を行います。
約 2.4ms を基準に、サンプリングクロックサイクル単位で 0 サイクルから-15 サイクルまで設定可能です。
- <CECBRD>: ブロードキャストメッセージの送信を行う場合、このビットを “1” に設定します。
- <CECFREE[3:0]>: 送信開始前に確認するバスフリー時間の設定を行います。
設定されたビット周期分の CEC ラインが稼働されないことを確認し、送信を開始します。

(注)	<CECDTRS>は必ず “0” 検出ノイズキャンセル時間 CECRCR1<CECLNC>と同じ設定で使用してください。
-----	--

13.2.13 受信割り込みステータスレジスタ 《CECRSTAT》

	7	6	5	4	3	2	1	0
bit Symbol	—	CECRIWAV	CECRIOR	CECRIACK	CECRIMIN	CECRIMAX	CECRISTA	CECRIEND
Read/Write	R	R	R	R	R	R	R	R
リセット後	0	0	0	0	0	0	0	0
機能	リードすると“0”が読めません。	割り込みフラグ 1: 波形エラー	割り込みフラグ 1: 受信バッファオーバーラン	割り込みフラグ 1: ACK衝突	割り込みフラグ 1: 最小周期違反	割り込みフラグ 1: 最大周期違反	割り込みフラグ 1: スタートビット検出	割り込みフラグ 1: 1バイト受信完了

<CECRIWAV>: 波形エラーが発生したことを示します。CEGRCR3<CECWAVEN>ビットで許可の設定をしている場合に発生します。

<CECRIOR>: 受信バッファにデータがセットされた後、読み出される前に次のデータを受信したことを示します。

<CECRIACK>: ACKビットで“0”出力期間後に“0”が観測されたことを示します。

<CECRIMIN>: 1ビットの周期が、最小周期違反検出時間としてCEGRCR1<CECMIN>に設定された時間より短かったことを示します。

<CECRIMAX>: 1ビットの周期が、最大周期違反検出時間としてCEGRCR1<CECMAX>に設定された時間より長かったことを示します。

<CECRISTA>: スタートビットが検出されたことを示します。

<CECRIEND>: 1バイト分のデータの受信が完了したことを示します。

(注)	このレジスタへの書き込み動作は無視されます。
-----	------------------------

13.2.14 送信割り込みステータスレジスタ 《CECTSTAT》

	7	6	5	4	3	2	1	0
bit Symbol	—			CECTIUR	CECTIACK	CECTIAL	CECTIEND	CECTISTA
Read/Write	R			R	R	R	R	R
リセット後	0			0	0	0	0	0
機能	リードすると“0”が読めます。			割り込み フラグ 1: 送信バ ッファア ンダーラ ン	割り込み フラグ 1: ACK エ ラー検出	割り込み フラグ 1: アービ トレシヨ ンロスト 発生	割り込み フラグ 1: 全メッ セージ送 信終了	割り込み フラグ 1: 送信開 始

<CECTIUR>: 1バイトの送信開始から終了までの間に送信バッファに次のデータがセットされなかったことを示します。

<CECTIACK>: ACK 送信において、以下の状態が発生したことを示します。
 ・ 特定のアドレスを設定した送信で論理“0”が検出されなかった場合。
 ・ ブロードキャストメッセージの送信で、論理“1”が検出されなかった場合。

<CECTIAL>: “1”を送信中に“0”を検出したことを示します。

<CECTIEND>: EOM 付きのデータの送信が終了したことを示します。

<CECTISTA>: 1バイトデータの送信を開始したことを示します。

(注) このレジスタへの書き込み動作は無視されます。

13.2.15 CEC サンプリングクロック選択レジスタ 《CECFSSSEL》

	7	6	5	4	3	2	1	0
bit Symbol	—							CECCLK
Read/Write	R							R/W
リセット後	0							0
機能	リードすると“0”が読めます。							CLK 選択 0:低速 CLK 1:TBAOUT

<CECCLK>: サンプリングクロック選択
 0:低速クロック (fs)
 1:TBAOUT

CEC 機能のサンプリングクロックを設定します。
 CEC 機能のサンプリングクロックとして、低速クロック (fs)かタイマ出力 (TBAOUT) を
 選択することが可能です。
 TBAOUT にて設定できるタイマ出力範囲は 30KHz~34KHz です。

(注) CECFSSSEL レジスタにてサンプリングクロックを切り替える場合は、CECEN<CECEN>にて CEC 動作を一旦停止 (禁止)させ、再度動作 (許可)設定した後、他の CEC 関連レジスタよりも先に CECFSSSEL レジスタを設定してください。また、GECRESET レジスタによるソフトウェアリセット後、CECFSSSEL レジスタにてサンプリングクロックを切り替える場合にも同様に、他の CEC 関連レジスタよりも先に CECFSSSEL レジスタを設定してください。

13.3 動作説明

13.3.1 受信

13.3.1.1 サンプリングクロック

CEC 信号のサンプリングは、低速クロック (f_s) または 16 ビットフリップフロップ出力 TBxOUT で行います。

CECFSSSEL<CECCLK>で使用するクロックを選択してください。

13.3.1.2 基本動作

スタートビット検出後、スタートビット割り込みを発生します。スタートビット割り込みを発生すると、CECRSTAT<CECRISTA>がセットされます。

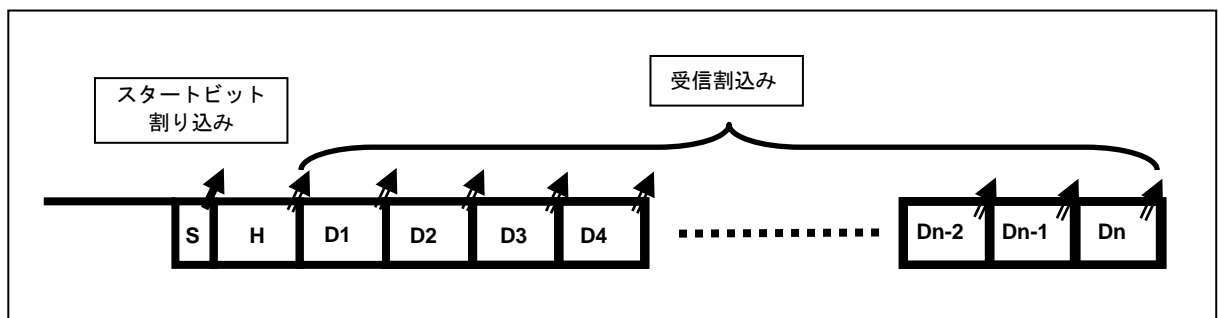
1 バイト分のデータと、EOM、ACK を受信すると受信したデータを CECRBUF レジスタに格納し、受信割り込みを発生します。受信割り込みが発生すると、CECRSTAT<CECRIEND>がセットされます。

CECRBUF レジスタには、データ 8 ビットと、EOM ビット、ACK ビットが格納されます。ACK ビットは CEC 回路内部で発生したものではなく、他のデータと同様 CEC 信号を観測した結果が格納されます。

1 つのデータブロック受信後、EOM ビットが "1" の最終データブロックまで継続して受信動作を行います。最終データブロックであることを検出すると、スタートビット待ち状態になります。

データ受信途中にエラーが検出されると、エラー割り込みを発生し、次のスタートビット待ち状態になります。エラーが発生した場合、取得したデータは破棄されます。

(注) 受信に際しては、13.1.3 の注意事項に留意してください。



13.3.1.3 受信の準備

受信を開始する前に、ロジカルアドレスレジスタ (CECADD)、受信コントロールレジスタ 1 (CECR1)、受信コントロールレジスタ 2 (CECR2)、受信コントロールレジスタ 3 (CECR3) で受信動作の設定を行います。

(1) ロジカルアドレスの設定

CECADD レジスタで、本デバイスに割り当てるロジカルアドレスを設定します。0 から 15 のビットがそれぞれアドレスと対応しており、複数のアドレスを設定することが可能です。

(注) ブロードキャストメッセージは、CECADD レジスタの設定によらず受信し、論理“1”の ACK 応答を行います。
ロジカルアドレス 15 を設定している場合には、ブロードキャストメッセージに対し論理“0”の ACK 応答を行います。

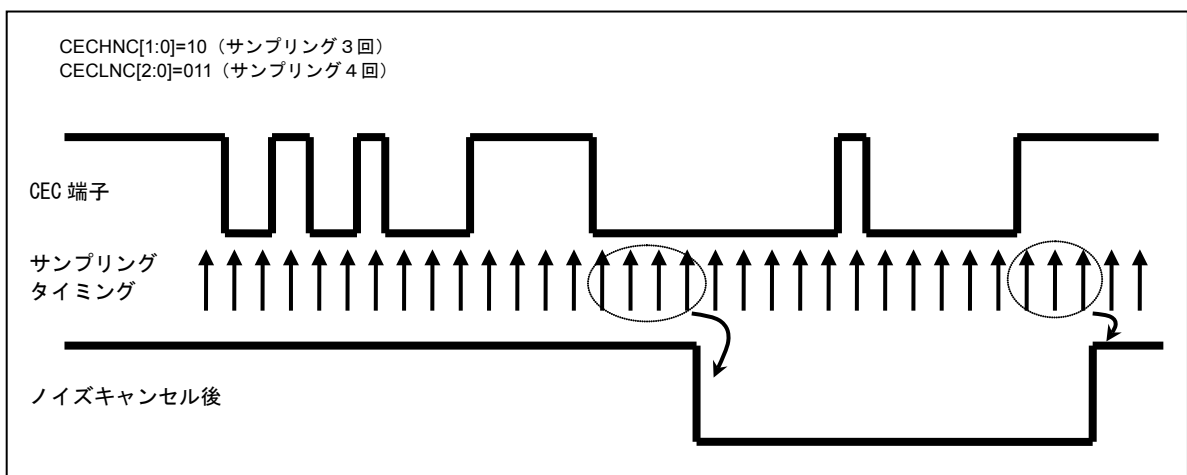
(2) ノイズキャンセル時間

CECR1 レジスタの<CECHNC[1:0]>、<CECLNC[2:0]>ビットでノイズキャンセル時間を設定します。設定されたサンプリング回数分の値が観測されなければノイズとみなされます。“1”観測用、“0”観測用を個別に設定可能です。

(注) <CECLNC>は、必ず CECTCR<CECDTRS>と同じ設定で使用してください。

ノイズキャンセルは、サンプリングクロックで CEC 信号をサンプリングした信号に対して行います。サンプリングクロックの立ち上がりエッジごとに CEC 信号を観測し、現在“1”であれば、<CECLNC>ビットで設定されたサンプリング回数の“0”が観測されたときに信号が“0”に変化したと認識し、現在“0”であれば、<CECHNC>ビットで設定されたサンプリング回数の“1”が観測されたとき“1”に変化したと認識します。

次の図は、ノイズキャンセルの設定を<CECHNC [1:0]>=10 (サンプリング 3 回)、<CECLNC [2:0]>=011 (サンプリング 4 回) とした場合の動作です。ノイズキャンセル後の信号は、“1”の状態から、“0”が 4 サイクル観測されたところで“0”に変化し、“0”の状態から、“1”が 3 サイクル観測されたところで“1”に変化します。



(3) 周期判定時間

CECR1 レジスタの<CECMIN[2:0]>、<CECMAX[2:0]>ビットで周期違反の検出のための設定を行います。

CEC 規格で決められた周期の最小値・最大値に対し、サンプリングクロックサイクル単位で-4クロックから+3クロックの間で周期違反の検出を行うことができます。

違反が検出された場合には割り込みが発生し、スタートビット待ちになります。割り込み発生までに受信したデータは破棄されます。

(4) データ判別タイミングの設定

CECR1 レジスタの<CECDAT>ビットでデータ判別を行うタイミングの設定を行います。

ビットのスタートポイントから約 1.05ms を基準に、サンプリングクロック 2クロックサイクル単位で±6 サイクルまで設定可能です。

(5) ACK 応答

CECR1 レジスタの<CECACKDIS>ビットで、ディスティネーションアドレスがロジカルアドレスレジスタに設定されたアドレスと一致した場合にデータブロックに対し、論理“0”のACK応答をする／しないを任意に設定することができます。

ヘッダブロックについては、このビットの設定によらずアドレスが一致した場合に論理“0”のACK応答を行います。

(6) 受信エラー割り込みの保留

CECR1 レジスタの<CECRIHLD>ビットで、受信エラー割り込みを保留するかどうかを設定します。対象となる割り込みは、最大周期違反、バッファオーバーラン、波形エラーです。

このビットが“1”に設定されていると、エラー検出時点では割り込みは発生しません。エラー検出後 ACK ビットまでデータが継続した場合は ACK 応答を通常と反転の論理で行い、後続のビットがいずれかのタイミングで途切れた場合には CECR1 レジスタ<CECTOUT>の設定にしたがってタイムアウトと判定します。割り込みは ACK 応答後、またはタイムアウト判定後に発生します。

(7) タイムアウトの設定

CECR1 レジスタ<CECTOUT>で、タイムアウト判定の設定を行います。

この設定は受信エラー保留の設定（CECR1 レジスタ<CECRIHLD>）が有効の場合に使用されます。

(8) ロジカルアドレスが一致しない場合の動作

CECRCR1 レジスタの<CECOTH>ビットで、ディスティネーションアドレスがロジカルアドレスレジスタに設定されたアドレスと一致しない場合にも受信するよう設定することができます。

この場合、受信動作は通常の場合と同様に行い、違反が検出されれば割り込みも発生しますが、ACK 応答はヘッダブロック、データブロックとも行いません。

(注1) ブロードキャストメッセージは、<CECOTH>ビットの設定によらず受信します。

(注2) 送信側が EOM ビット “1” のブロックを送信せずに新たにメッセージをスタートビットから送信した場合、ACK ビットが最大周期違反と判定され割り込みが発生します。その後の受信は通常通り行われます。

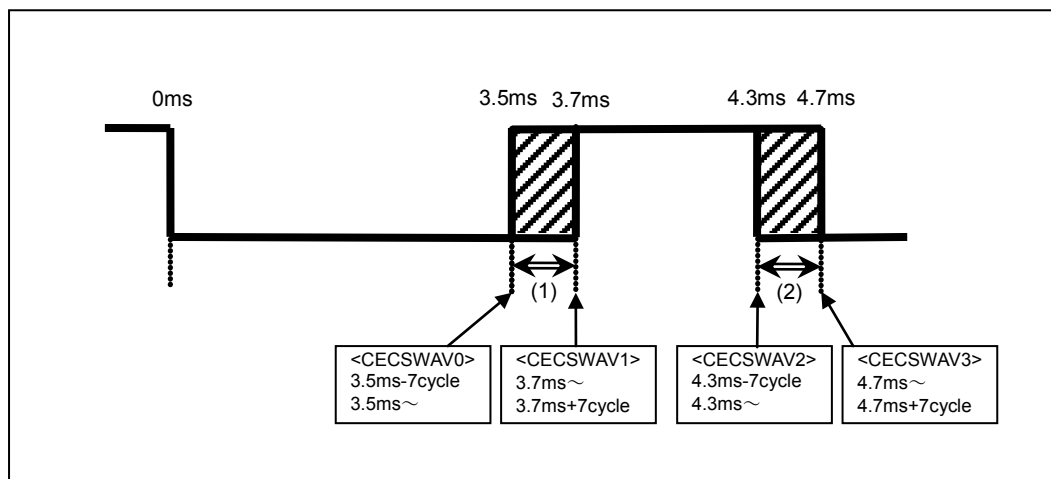
(9) スタートビット検出の設定

CECRCR2 レジスタでスタートビット検出の条件を設定することができます。立ち上がりのタイミングと周期をそれぞれ設定可能です。

<CECSWAV0>で立ち上がりのもっとも早いタイミング、<CECSWAV1>で立ち上がりのもっとも遅いタイミングを設定します。(下図の(1)の期間)。

<CECSWAV2>で周期のもっとも早いタイミング、<CECSWAV3>で周期のもっとも遅いタイミングを設定します。(下図の(2)の期間)。

(1) の期間に立ち上がり、(2) の期間に立下りが検出されると、有効なスタートビットとみなします。



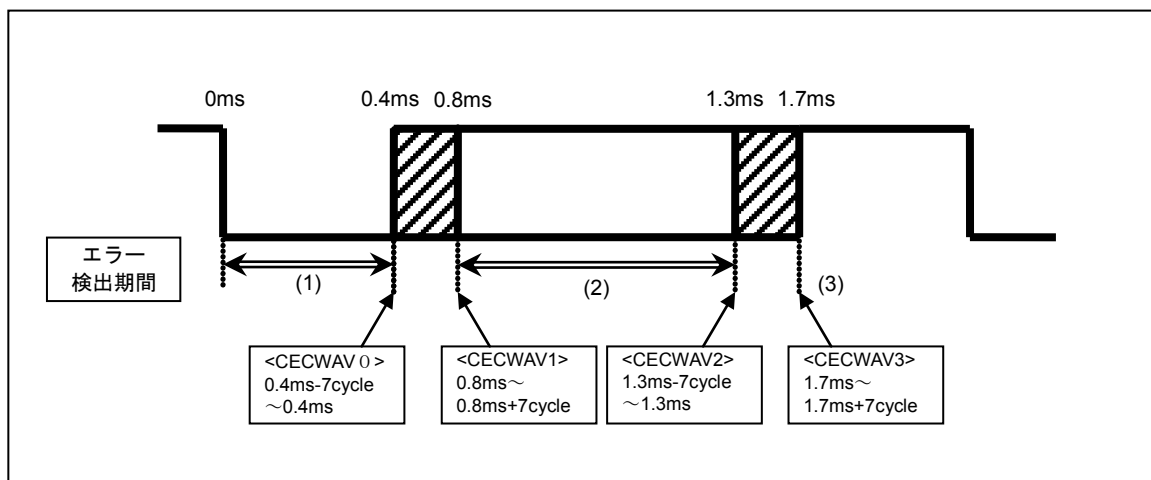
(10) 波形エラー検出の設定

受信波形が規格から外れた場合にエラーとして検出するには、CECRCR3 レジスタの設定を行います。

CECRCR3 レジスタ<CECWAVEN>ビットが許可の設定の場合にエラー検出を行います。検出期間は<CECWAV0><CECWAV1><CECWAV2><CECWAV3>ビットで調整が可能です。

以下の、(1) (2) の期間に立ち上がりがあった場合または、(3) のタイミングまでに立ち上がりなかった場合に波形エラー割り込みを発生します。

- (1) ビット開始から論理“1”の最も早い立ち上がりタイミングの間
- (2) 論理“1”のもっとも遅い立ち上がりタイミングと論理“0”のもっとも早い立ち上がりタイミングの間
- (3) 論理“0”のもっとも遅い立ち上がりタイミング



13.3.1.4 受信許可

CECADD、CECR1、CECR2、CECR3 レジスタの設定終了後、CECREN レジスタの<CECREN>ビットを受信許可に設定することで受信待ち状態になり、スタートビットを検出すると受信動作を開始します。

(注) 受信動作中に CECADD、CECR1、CECR2、CECR3 レジスタの設定が変更されると正しく受信できない可能性があります。
以下のレジスタ設定を変更する場合には、CECREN レジスタ<CECREN>ビットで受信禁止の設定を行い、<CECREN>および CECTEN レジスタ<CECTEN>ビットをリードして送受信とも停止中であることを確認後に変更することを推奨します。

CECADD	<CECADD[15:0]>	ロジカルアドレス
CECR1	<CECHNC><CECLNC>	ノイズキャンセル時間
	<CECMIN><CECMAX>	周期時間
	<CECOTH>	ロジカルアドレス不一致時の受信
CECR2	<CECSWAV0><CECSWAV1> <CECSWAV2><CECSWAV3>	スタートビット検出設定
CECR3 (波形エラー有効の場合)	<CECWAV0><CECWAV1> <CECWAV2><CECWAV3>	波形確認設定

13.3.1.5 受信動作

スタートビット検出後、スタートビット割り込みを発生します。スタートビット割り込みを発生すると、CECRSTAT レジスタの<CECRISTA>ビットがセットされます。

1 バイト分のデータと、EOM、ACK を受信すると受信したデータを CECRBUF レジスタに格納し、受信割り込みを発生します。受信割り込みが発生すると、CECRSTAT レジスタの<CECRIEND>ビットがセットされます。

CECRBUF レジスタには、データ 8 ビットと、EOM ビット、ACK ビットが格納されます。

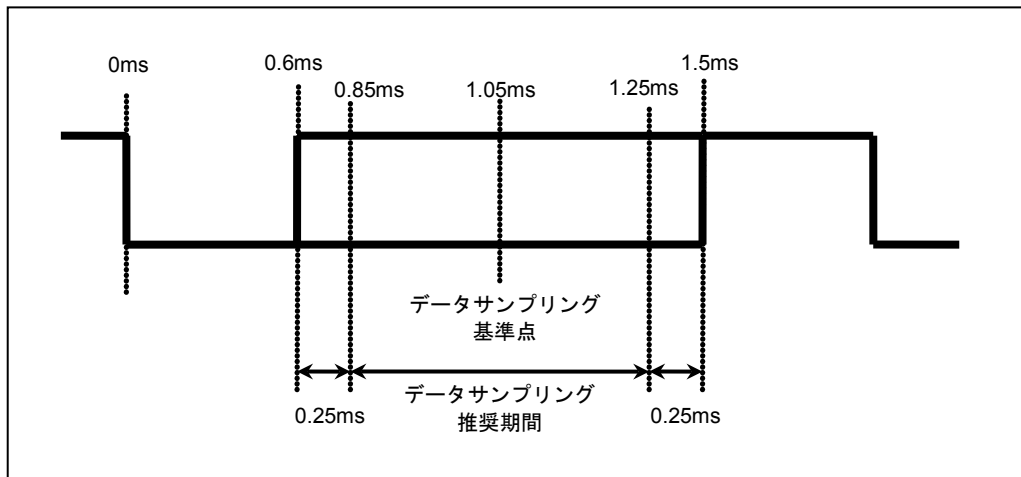
ACK ビットは CEC 回路内部で発生したものではなく、他のデータと同様 CEC 信号を観測した結果が格納されます。

1 つのデータブロック受信後、EOM ビットが“1”の最終データブロックまで継続して受信動作を行います。最終データブロックであることを検出すると、スタートビット待ち状態になります。

13.3.1.6 データ取り込みタイミング

受信時のデータ取り込みのタイミングを以下に示します。

データサンプリング基準点の 1.05ms のポイントを中心に、サンプリングクロック 2 サイクルごとに ±6cycle までサンプリングポイントを選択することができます。サンプリングポイントの選択は、CECR1 レジスタ<CECDAT>ビットで行います。



13.3.1.7 ACK 応答

CECR1 レジスタの<CECACKDIS>ビットで、ディスティネーションアドレスがロジカルアドレスレジスタに設定されたアドレスと一致した場合に、データブロックに対し論理 “0” の ACK 応答する／しないを設定することができます。

ヘッダブロックに対する ACK 応答はこのビットは関係しません。

以下に ACK 応答動作についてまとめます。

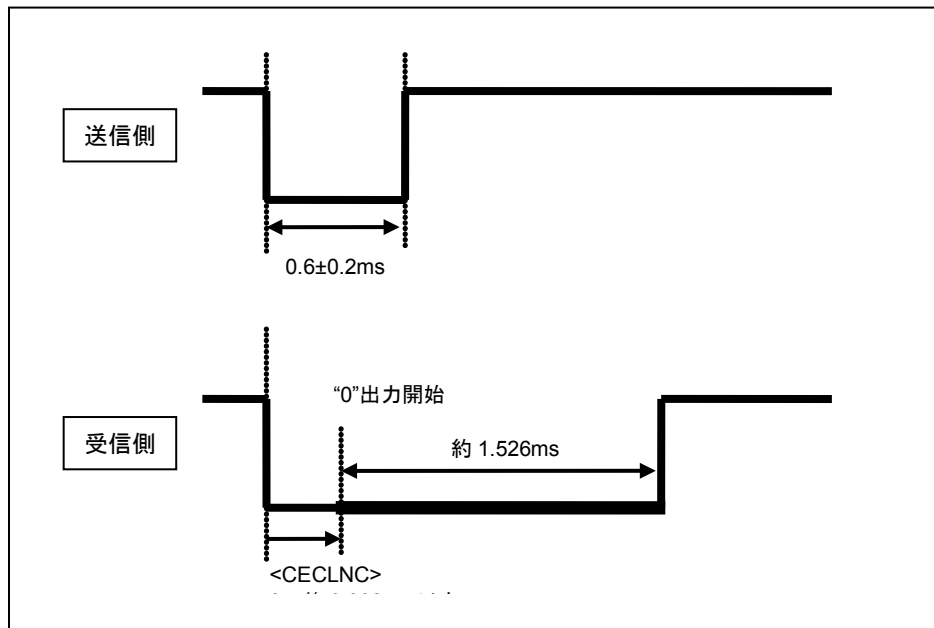
以下の表で、「○」は送信デバイスの ACK 送信に対し “0” 出力をする (ACK ビットは論理 “0” となる) こと、「×」は “0” 出力をしない (ACK ビットは論理 “1” となる) ことを示します。

レジスタ設定		ヘッダブロック		データブロック	
		アドレス一致	アドレス不一致	アドレス一致	アドレス不一致
CECR1 <CECACKDIS>	“0” (論理 0 応答する)	○	×	○	×
	“1” (論理 0 応答しない)			×	×

ACK 応答のタイミングを以下に示します。

送信側が出力した ACK ビットの立下りを検出すると、約 1.526ms 間 “0” を出力します。立下り検出までの時間は、“0” 検出ノイズキャンセル時間設定 (CECR1<CECLNC>) によって決まります。

(注) CECRCR1<CECLNC>は、必ず CECTCR<CECDTRS>と同じ設定で使用してください。



13.3.1.8 受信エラー検出

受信中にエラーが検出されると割り込みを発生し、受信動作を停止してスタートビット待ち状態になります。エラーが発生した受信データは破棄されます。

最大周期違反、受信バッファオーバーラン、波形エラーについては、割り込み発生を保留し受信動作を継続して ACK 応答を通常の反転の論理で行うことも可能です。

エラーの要因を確認するために、CECRSTAT レジスタに割り込みに対応した要因ビットが準備されており、割り込み要因を確認することが出来ます。

13.3.1.9 受信エラー詳細

(1) 周期違反

受信中、ビット開始の信号の立下りから次のビット開始の信号の立下りの周期を測定し、最小値／最大値の設定を違反している場合には割り込みを発生します。

周期の最大値／最小値の設定は CECRCR1 レジスタ<CECMIN[2:0]>、<CECMAX[2:0]>ビットで行います。CEC 規格で決められた周期の最小値 (2.045ms) / 最大値 (2.747ms) に対し、サンプリングクロックサイクル単位で-4クロックから+3クロックの間で周期違反の検出を行うことができます。

周期違反の割り込みが発生すると、CECRSTAT レジスタの<CECRIMIN>ビットまたは<CECRIMAX>ビットがセットされます。

また、周期最小値の違反が発生した場合には、CEC 信号へ約 3.63ms 間 “0” を出力します。

(注1) 周期最小値違反の際、“0” 検出ノイズキャンセル時間後から “0” 出力を開始します。

(注2) 送信側が EOM ビット “1” のブロックを送信せずに新たにメッセージをスタートビットから送信した場合、ACK ビットが最大周期違反と判定され割り込みが発生する場合があります。詳細は 13.1.3 項を参照してください。

(2) ACK 衝突

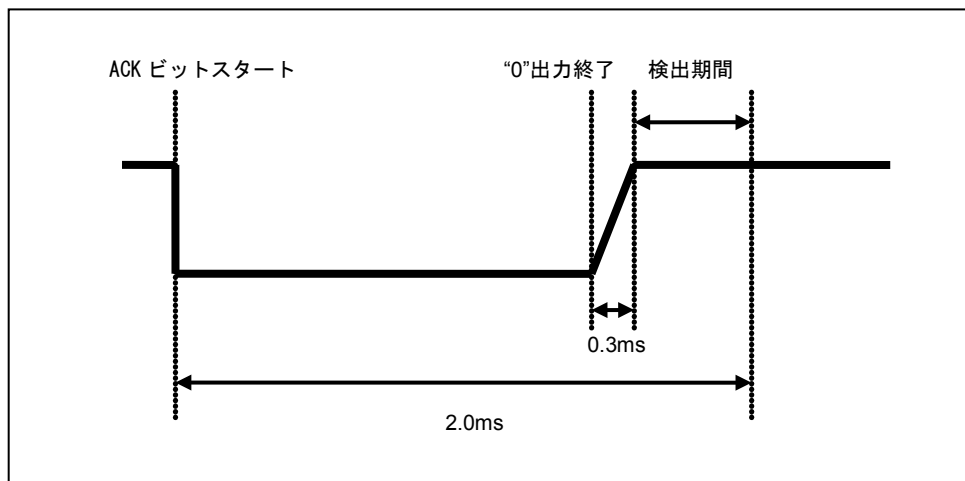
ACK 応答時、“0” 出力期間後に CEC 信号が “0” であることを検出すると、ACK 衝突割り込みまたは周期最小値違反の割り込みを発生します。

ACK 衝突割り込みが発生すると、CECRSTAT レジスタの<CECRIACK>ビットがセットされます。周期最小値の違反割り込みが発生すると、CECRSTAT レジスタの<CECRIMIN>ビットがセットされます。

検出期間と検出方法を以下に示します。

“0” 出力期間終了から約 0.3ms 後から検出を開始します。検出期間の終了は、ACK ビットのスタート (立下り) から約 2.0ms のところです。

“0” 出力期間終了から 0.3ms のポイントで CEC 信号が “0” であるかどうかを確認し、“0” の場合には ACK 衝突割り込みを発生します。このポイントで CEC 信号が “1” で、検出期間内に “0” を観測した場合には周期最小値の違反割り込みが発生し、CEC 信号へ約 3.63ms 間 “0” を出力します。



(3) 受信バッファオーバーラン

受信バッファに格納されたデータが読み出される前に次の 1 バイト分のデータの受信が完了すると、受信バッファのオーバーラン割り込みが発生します。

オーバーラン割り込みが発生すると、CECRSTAT レジスタの<CECRIOR>ビットがセットされます。

(4) 波形エラー

CECRCR3 レジスタで波形エラー検出を許可に設定している場合に発生します。受信波形が規格から外れた場合に波形エラーを検出し、割り込みが発生します。

波形エラー割り込みが発生すると、CECRSTAT レジスタの<CECRIWAV>ビットがセットされます。

(5) 受信エラー割り込みの保留

最大周期違反、受信バッファオーバーラン、波形エラーについては、エラー検出時に割り込みを発生させずに保留することができます。この設定は CECRCR1 レジスタの<CECRIHLD>で行います。また、この設定を使用するときには、CECRCR1 レジスタの<CECTOUT>でタイムアウトの設定も行います。

割り込み保留の設定が有効の場合、後続のデータビットを継続して受信し、ACK ビットまで受信できた場合には ACK 応答を通常の反転の論理で行った後に割り込みが発生します。このとき、CECRSTAT レジスタには、受信完了の<CECRIEND>と保留されたエラーのフラグがセットされます。

後続のデータビットがいずれかのタイミングで途切れた場合には、タイムアウトの計測を行い、タイムアウト後に割り込みが発生します。このとき、CECRSTAT レジスタにセットされるのは、保留されたエラーのフラグのみです。

タイムアウトの計測は、送信動作でのバスフリー待ち時間と同様、最後に受信したビットの終了時点から計測します。

割り込みが保留されているという情報は、EOM が “1” のデータ受信完了またはタイムアウトするまで保持されます。したがって、割り込みが保留された状態で複数バイトの受信が行われた場合、1 バイト受信ごとに割り込みが発生し、CECRSTAT レジスタには受信完了と保留された割り込みのフラグがセットされます。

- | |
|---|
| <p>(注1) 割り込み保留中に、後続の受信で最小周期違反が発生した場合には、直ちに最小周期違反割り込みが発生し、CEC 信号に約 3.6ms 間 “0” を出力します。
受信割り込みステータスレジスタには、保留された割り込みと最小周期違反のフラグがセットされます。</p> <p>(注2) 割り込み保留中に、後続の受信で最小周期違反以外のエラーが発生した場合には、ACK 応答またはタイムアウトまで処理が継続されます。
受信割り込みステータスレジスタには、検出されたすべての割り込みのフラグがセットされます。</p> |
|---|

13.3.1.10 受信の停止

CECREN レジスタの<CECREN>ビットを “0” (受信禁止) に設定すると受信動作を停止します。受信動作中に禁止の設定を行った場合、直ちに受信動作を停止し、それまでに受信したデータは破棄されます。

- | |
|--|
| <p>(注) 周期最小値の違反に対する “0” 出力中に受信禁止設定を行うと “0” 出力も停止します。</p> |
|--|

13.3.2 送信

13.3.2.1 基本動作

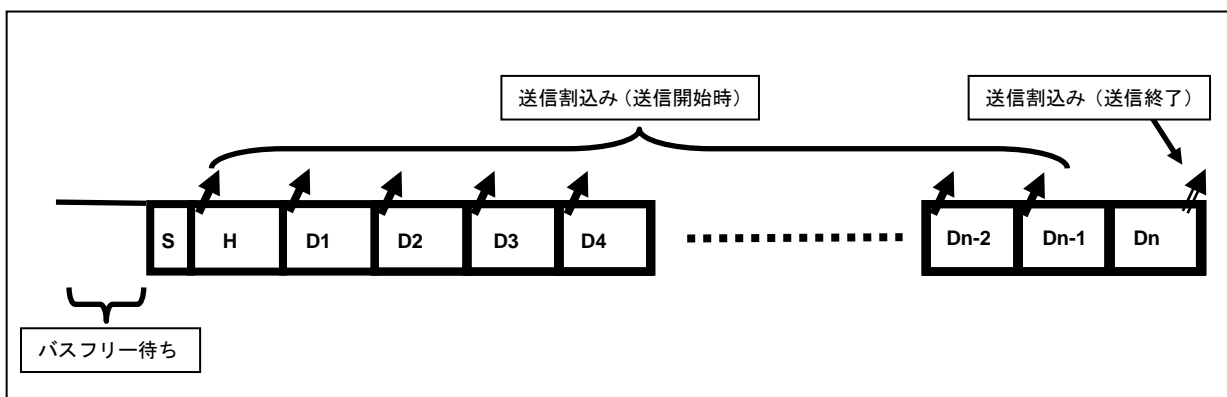
データバッファ設定後、送信開始ビットを設定すると、バスフリー期間の確認後スタートビット送信を開始します。

スタートビットに続けてデータの最初のビットの送信を開始すると送信割り込みを発生し、次のデータを送信バッファに設定可能であることを示します。データ 1 バイト (8bit)、EOM ビット送信後、ACK ビットを送信し、ACK 応答の検出を行います。

EOM ビットが“1”のデータが送信バッファに設定されるまで 1 バイトごとにデータ送信を続けます。EOM ビットが“1”のデータの送信が終了すると送信終了割り込みを発生します。

送信中にエラーが発生した場合、エラー割り込みを発生して送信動作を停止します。

送信中は受信許可状態であっても受信動作は行いません。



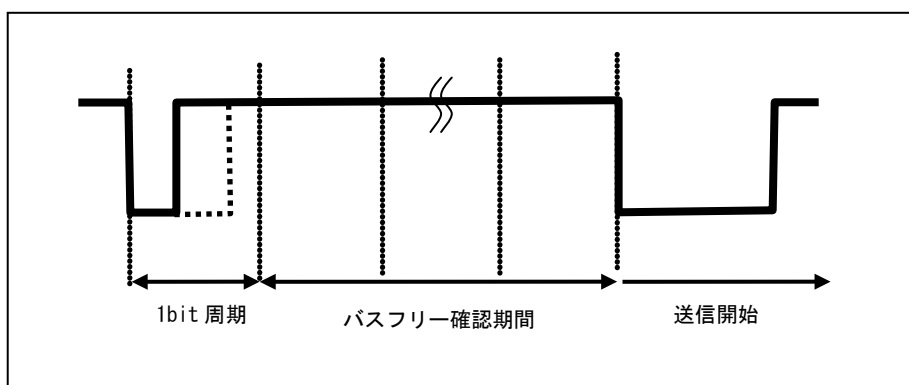
13.3.2.2 送信の準備

送信を開始する前に、送信コントロールレジスタ (CECTCR) と送信バッファ (CECTBUF) の設定を行います。

(1) バスフリー待ち時間

CECTCR レジスタの<CECFREE>ビットでバスフリー待ち時間を設定します。ビット周期単位で1ビット周期から16ビット周期まで設定可能です。

バスフリー状態の確認は最終ビットの立ち下がりの1ビット周期後から開始し、設定されたビット周期分立下りがなければ送信を開始します。



(2) ブロードキャストメッセージの送信

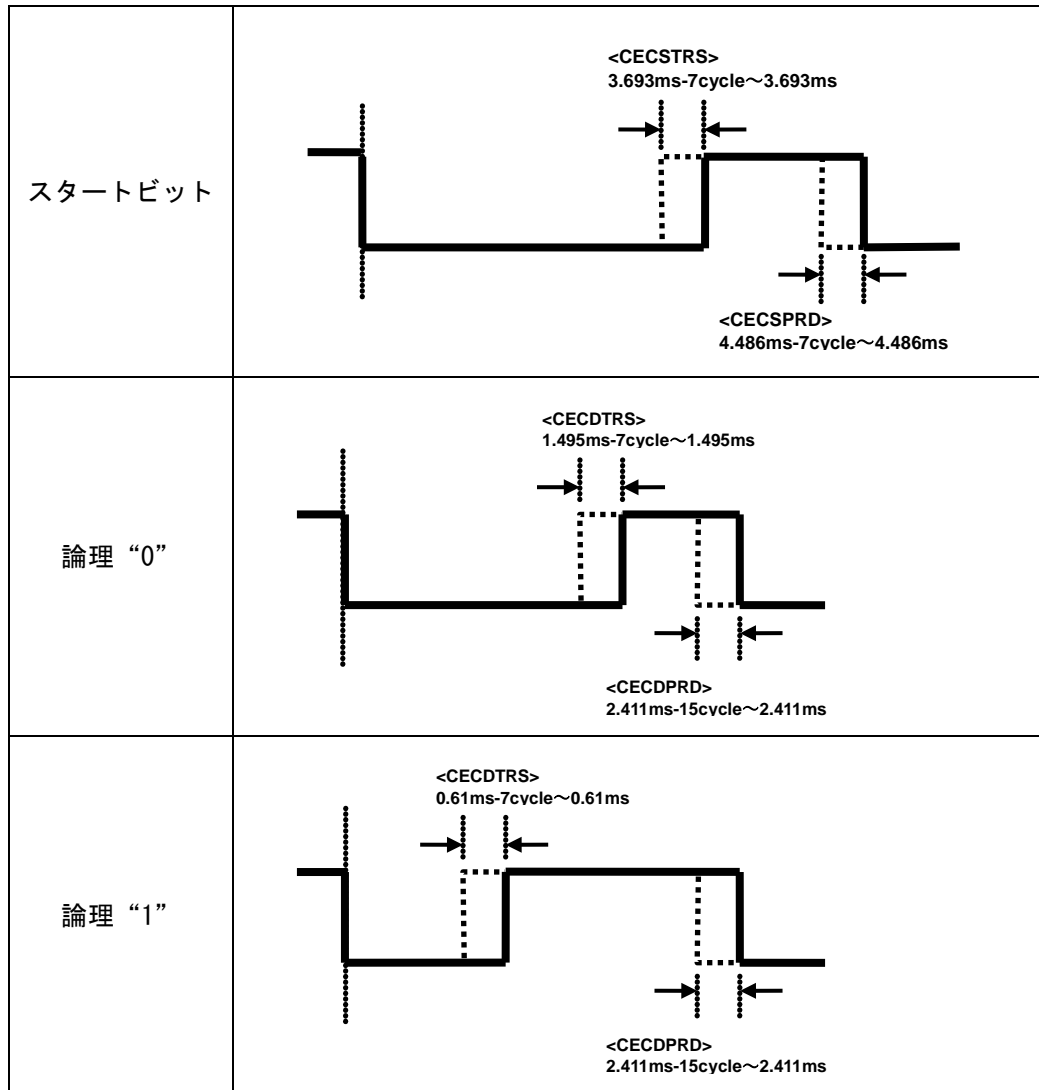
ブロードキャストメッセージの送信を行う場合は、CECTCR レジスタの<CECBRD>ビットを設定します。このビットがセットされているときには、ACK サイクルで“0”の応答があるとエラーになります。このビットがセットされていないときは、ACK サイクルで“1”の応答があるとエラーになります。

(3) 送信波形の調整

スタートビット/データビットとも立ち上がりタイミングと周期の調整が可能です。CECTCR レジスタの<CECSTRS><CECSPRD><CECDTRS><CECDPRD>ビットで設定を行います。規格で定められた、立ち上がりと周期の最も早いタイミングから標準値の間で調整することが出来ます。

(注) <CECDTRS>は、必ず CECRCR1<CECLNC>と同じ設定で使用してください。

以下に、スタートビット、論理“0”、論理“1”の設定による波形の違いを示します。



(4) 送信データの準備

1 バイト分の送信データと E OM のデータを CECTBUF レジスタに設定します。

13.3.2.3 送信開始

CECTCR レジスタと CECTBUF レジスタの設定終了後、CECTEN レジスタの<CECTEN>ビットを送信開始に設定することで送信の準備が整います。

<CECTEN>ビットは、送信終了割り込みまたはエラー割り込みの発生まで“0”にクリアされることはありません。したがって、1 バイトのデータ転送ごとに設定する必要はありません。

(注) 送信動作中に CECTCR レジスタの設定が変更されると正しく送信できない可能性があります。送信開始後に設定変更を行う場合は注意して行ってください。

13.3.2.4 送信動作

送信開始の設定を行うとまずバスフリー状態を確認します。設定されたビット周期分の CEC 信号 “1” の状態を確認後にスタートビットを送信します。バスフリー状態の確認は常時行っており、送信開始設定時に、設定されたビット周期分のバスフリー条件を満たしていればすぐに送信を開始します。

スタートビット送信後にバッファに設定された 1 バイトのデータと EOM データがシフトレジスタに送られ、データの送信を開始します。1 バイトのデータの最初のビットの送信が開始されると送信割り込みが発生し、CECTSTAT レジスタの<CECTISTA>ビットがセットされます。送信割り込み発生後、次の 1 バイトのデータを送信バッファに設定します。

8 ビットのデータ、EOM ビット、ACK ビット送信後、ACK 応答を確認し 1 バイト分のデータ転送が終了します。

EOM ビットに “1” が設定されるまで同様にデータ転送を行います。

EOM に “1” が設定されている場合、データ、EOM、ACK ビットの送信および、ACK 応答の確認後に送信終了割り込みを発生します。送信終了割り込みが発生すると、CECTSTAT レジスタの<CECTIEND>ビットがセットされます。

送信終了割り込み発生により一連の送信動作が終了し、CECTEN レジスタの<CECTEN>ビットはクリアされます。

13.3.2.5 ACK 送信と ACK 応答判定

CECTCR レジスタの<CECBRD>ビットにより判定方法が変わります。

このビットが設定されているときはブロードキャストメッセージの送信となり、ACK 応答が論理 “0” の場合にエラーとなります。設定されていない場合、ACK 応答が論理 “1” の場合にエラーとなります。

13.3.2.6 送信エラー検出

送信中にエラーが検出されると割り込みを発生し、送信動作を停止します。また、CECTEN レジスタの<CECTEN>ビットはクリアされます。

エラーの要因を確認するために、CECTSTAT レジスタに割り込みに対応した要因ビットが準備されており、割り込み要因を確認することが出来ます。

(注) エラーにより送信動作を停止する場合、エラー発生後直ちに出力を停止するため CEC 信号に不正な波形が出力される場合があります。

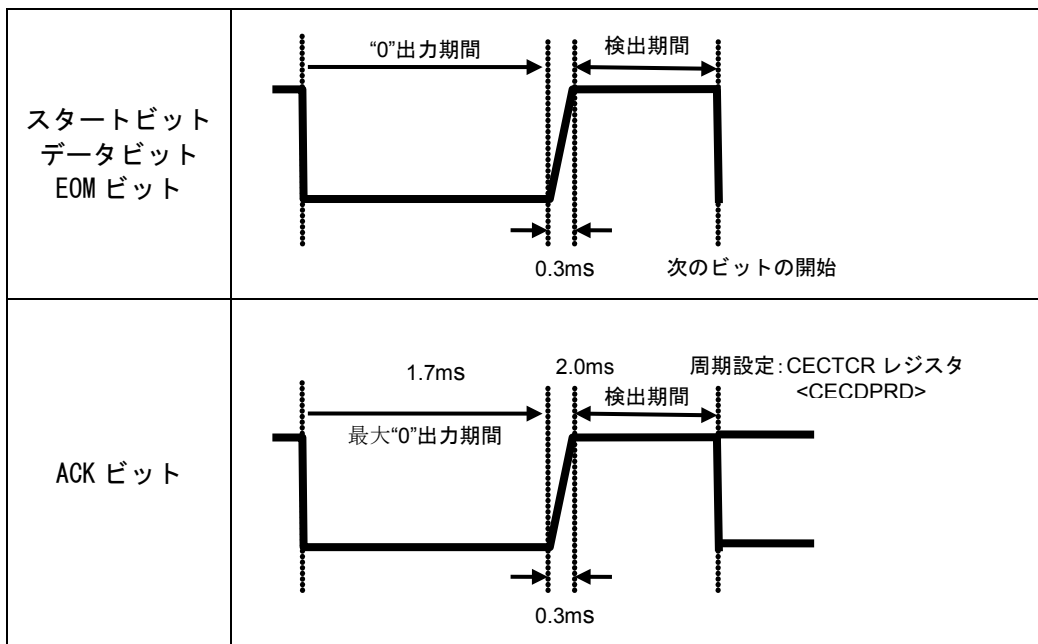
13.3.2.7 送信エラー詳細

(1) アービトレーションロスト

スタートビットおよびデータ送信中で“0”出力していない状態、またはACK応答後、CEC信号に“0”が検出された場合アービトレーションロストエラーが発生します。

アービトレーションロストが発生すると、CECTSTATレジスタの<CECTIAL>ビットがセットされます。

以下に、アービトレーションロスト検出期間を示します。



(2) ACK 違反

CECTCRレジスタの<CECBRD>ビットにしたがってACK応答の内容を確認し違反しているとACK違反割り込みが発生します。

ACK違反割り込みが発生すると、CECTSTATレジスタの<CECTIACK>ビットがセットされます。

ACK応答の確認は以下のように行います。

設定	確認内容
<CECBRD>=0 ブロードキャスト送信でない	ACK 応答が論理 “1” のときに ACK 違反
<CECBRD>=1 ブロードキャスト送信	ACK 応答が論理 “0” のときに ACK 違反

(3) 送信バッファアンダーラン

送信バッファのデータがシフトレジスタに送られ割り込みが発生した後、1 バイト分のデータを送信し、次の1 バイトデータの送信開始までに送信バッファにデータが設定されなかった場合、送信バッファのアンダーランエラーが発生します。

アンダーランエラーが発生すると、CECTSTAT レジスタの<CECTIUR>ビットがセットされます。

(4) ACK 違反と送信バッファアンダーランの発生順序

1 バイトデータ送信終了時点でACK違反と送信バッファアンダーランの両方の要因が存在する場合、割り込みは送信バッファアンダーラン→ACK違反の順に発生します。

13.3.2.8 送信の停止

送信動作を終了する場合、EOM ビットが“1”のデータを送信することで送信終了割り込みにより終了してください。

送信動作中に送信開始ビットを“0”に設定した場合の動作は保証しません。

13.3.2.9 再送信

エラーが検出されると送信動作を停止します。送信をやり直す場合には、改めて送信条件と送信データを設定し、送信開始の設定を行ってください。

13.3.3 ソフトウェアリセット

ソフトウェアで CEC 機能全体を初期化することが出来ます。

ソフトウェアリセットレジスタ (CECRESET) の<CECRESET>ビットに“1”をセットすることで以下のように動作します。

- 受信動作 : 直ちに停止。受信データは破棄。
- 送信動作 : 直ちに停止。CEC 信号出力も停止。
- レジスタ : CECEN レジスタ以外の全てのレジスタを初期化。

送信動作中にソフトウェアリセットを行うと、CEC 信号上の波形は規格を満たさないものになる可能性があるため注意が必要です。

14 リモコン判定機能

14.1 概要

搬送波が取り除かれたリモコン信号の受信を行います。

14.1.1 リモコン受信

- 32KHz クロックでサンプリング
- ノイズキャンセラ
- リーダ検出
- 最大 72bit まで一括受信

14.2 レジスタ説明

14.2.1 レジスタ一覧

リモコン受信の制御レジスタとアドレスは以下のとおりです。

レジスタ名		アドレス	
		チャンネル 0	チャンネル 1
リモコンイネーブルレジスタ	RMCxEN	0x4004_0400	0x4004_0440
リモコン受信イネーブルレジスタ	RMCxREN	0x4004_0404	0x4004_0444
リモコン受信データバッファレジスタ 1	RMCxRBUF1	0x4004_0408	0x4004_0448
リモコン受信データバッファレジスタ 2	RMCxRBUF2	0x4004_040C	0x4004_044C
リモコン受信データバッファレジスタ 3	RMCxRBUF3	0x4004_0410	0x4004_0450
リモコン受信コントロールレジスタ 1	RMCxRCR1	0x4004_0414	0x4004_0454
リモコン受信コントロールレジスタ 2	RMCxRCR2	0x4004_0418	0x4004_0458
リモコン受信コントロールレジスタ 3	RMCxRCR3	0x4004_041C	0x4004_045C
リモコン受信コントロールレジスタ 4	RMCxRCR4	0x4004_0420	0x4004_0460
リモコン受信ステータスレジスタ	RMCxRSTAT	0x4004_0424	0x4004_0464

14.2.2 リモコンイネーブルレジスタ 《RMCEN》

	7	6	5	4	3	2	1	0	
bit Symbol	—							—	RMCEN
Read/Write	R							R/W	R/W
リセット後	0							0	0
機能	リードすると“0”が読めます。							”1”をラ イトして ください。	RMC 動作 0:禁止 1:許可

<RMCEN>: リモコン判定機能の動作を制御します。
 リモコン判定機能を使用する場合は、まずリモコン動作許可にしてください。
 動作禁止の状態では、イネーブルレジスタを除く RMC 機能のすべてのクロックが停止しますので消費電力の低減が可能です。
 リモコン判定機能を一旦動作させた後に、動作禁止にした場合は各レジスタの設定は保持されます。

14.2.3 リモコン受信イネーブルレジスタ 《RMCREN》

	7	6	5	4	3	2	1	0	
bit Symbol	—							—	RMCREN
Read/Write	R							R/W	R/W
リセット後	0							0	0
機能	リードすると“0”が読めます。								受信 0:禁止 1:許可

<RMCREN>: リモコン判定機能の受信動作を制御します。
 このビットを“1”にすることで受信待ち状態になります。

(注) <RMCREN>ビットは、リモコン受信コントロールレジスタ 1, 2, 3 を設定した後に許可してください。

14.2.4 リモコン受信データバッファレジスタ 1 《RMCRBUF1》

	31	30	29	28	27	26	25	24
bit Symbol	RMCRBUF31	RMCRBUF30	RMCRBUF29	RMCRBUF28	RMCRBUF27	RMCRBUF26	RMCRBUF25	RMCRBUF24
Read/Write	R							
リセット後	0							
機 能	受信データ							
	23	22	21	20	19	18	17	16
bit Symbol	RMCRBUF23	RMCRBUF22	RMCRBUF21	RMCRBUF20	RMCRBUF19	RMCRBUF18	RMCRBUF17	RMCRBUF16
Read/Write	R							
リセット後	0							
機 能	受信データ							
	15	14	13	12	11	10	9	8
bit Symbol	RMCRBUF15	RMCRBUF14	RMCRBUF13	RMCRBUF12	RMCRBUF11	RMCRBUF10	RMCRBUF9	RMCRBUF8
Read/Write	R							
リセット後	0							
機 能	受信データ							
	7	6	5	4	3	2	1	0
bit Symbol	RMCRBUF7	RMCRBUF6	RMCRBUF5	RMCRBUF4	RMCRBUF3	RMCRBUF2	RMCRBUF1	RMCRBUF0
Read/Write	R							
リセット後	0							
機 能	受信データ							

<RMCRBUF[31:0]>: 受信した4バイト分のデータが読めます。

14.2.5 リモコン受信データバッファレジスタ 2 《RMCRBUF2》

	31	30	29	28	27	26	25	24
bit Symbol	RMCRBUF63	RMCRBUF62	RMCRBUF61	RMCRBUF60	RMCRBUF59	RMCRBUF58	RMCRBUF57	RMCRBUF56
Read/Write	R							
リセット後	0							
機 能	受信データ							
	23	22	21	20	19	18	17	16
bit Symbol	RMCRBUF55	RMCRBUF54	RMCRBUF53	RMCRBUF52	RMCRBUF51	RMCRBUF50	RMCRBUF49	RMCRBUF48
Read/Write	R							
リセット後	0							
機 能	受信データ							
	15	14	13	12	11	10	9	8
bit Symbol	RMCRBUF47	RMCRBUF46	RMCRBUF45	RMCRBUF44	RMCRBUF43	RMCRBUF42	RMCRBUF41	RMCRBUF40
Read/Write	R							
リセット後	0							
機 能	受信データ							
	7	6	5	4	3	2	1	0
bit Symbol	RMCRBUF39	RMCRBUF38	RMCRBUF37	RMCRBUF36	RMCRBUF35	RMCRBUF34	RMCRBUF33	RMCRBUF32
Read/Write	R							
リセット後	0							
機 能	受信データ							

<RMCRBUF[63:32]>: 受信した4バイト分のデータが読めます。

14.2.6 リモコン受信データバッファレジスタ 3 《RMCRBUF3》

	7	6	5	4	3	2	1	0
bit Symbol	RMCRBUF71	RMCRBUF70	RMCRBUF69	RMCRBUF68	RMCRBUF67	RMCRBUF66	RMCRBUF65	RMCRBUF64
Read/Write	R							
リセット後	0							
機 能	受信データ							

<RMCRBUF[71:64]>: 受信した1バイト分のデータが読めます。

- | | |
|------|--|
| (注1) | 受信データはリモコン受信データバッファレジスタ1の<RMCRBUF0>から順にリモコン受信データバッファレジスタ3の<RMCRBUF71>まで格納されます。 |
| (注2) | 受信データは、最初に受信したBitが本レジスタのMSBに、最後に受信したBitが本レジスタのLSB(Bit0)に格納されます。
LSB firstのリモコン信号を受信した場合、Bitの重みが逆順のデータが本レジスタへ格納されますので、ご注意ください。 |

14.2.7 リモコン受信コントロールレジスタ 1 《RMCR1》

	31	30	29	28	27	26	25	24
bit Symbol	RMCLCMAX7	RMCLCMAX6	RMCLCMAX5	RMCLCMAX4	RMCLCMAX3	RMCLCMAX2	RMCLCMAX1	RMCLCMAX0
Read/Write	R/W							
リセット後	0							
機能	リーダ検出の周期期間の上限設定 $RMCLCMAX \times 4 / fs [s]$							
	23	22	21	20	19	18	17	16
bit Symbol	RMCLCMIN7	RMCLCMIN6	RMCLCMIN5	RMCLCMIN4	RMCLCMIN3	RMCLCMIN2	RMCLCMIN1	RMCLCMIN0
Read/Write	R/W							
リセット後	0							
機能	リーダ検出の周期期間の下限設定 $RMCLCMIN \times 4 / fs [s]$							
	15	14	13	12	11	10	9	8
bit Symbol	RMCLLMAX7	RMCLLMAX6	RMCLLMAX5	RMCLLMAX4	RMCLLMAX3	RMCLLMAX2	RMCLLMAX1	RMCLLMAX0
Read/Write	R/W							
リセット後	0							
機能	リーダ検出の Low 期間の上限設定 $RMCLLMAX \times 4 / fs [s]$							
	7	6	5	4	3	2	1	0
bit Symbol	RMCLLMIN7	RMCLLMIN6	RMCLLMIN5	RMCLLMIN4	RMCLLMIN3	RMCLLMIN2	RMCLLMIN1	RMCLLMIN0
Read/Write	R/W							
リセット後	0							
機能	リーダ検出の Low 期間の下限設定 $RMCLLMIN \times 4 / fs [s]$							

<RMCLCMAX[7:0]>: リーダ検出の周期期間の上限の設定をします。
 上限期間の計算式は $RMCLCMAX \times 4 / fs [s]$ 。
 検出は上限期間以下となります。

<RMCLCMIN[7:0]>: リーダ検出の周期期間の下限の設定をします。
 下限期間の計算式は $RMCLCMIN \times 4 / fs [s]$ 。
 検出は下限期間以上となります。

<RMCLLMAX[7:0]>: リーダ検出の Low 期間の上限の設定をします。
 上限期間の計算式は $RMCLLMAX \times 4 / fs [s]$ 。
 検出は上限期間以下となります。

<RMCLLMIN[7:0]>: リーダ検出の Low 期間の下限の設定をします。
 下限期間の計算式は $RMCLLMIN \times 4 / fs [s]$ 。
 検出は下限期間以上となります。
 RMCR2<RMCLD> = 1 のときは、設定値未満の値をデータと判別します。

(注 1) リーダ検出の設定時には以下の関係式を守って下さい。

リーダ種類	関係式
Low 幅+High 幅	<RMCLCMAX[7:0]> > <RMCLCMIN[7:0]> <RMCLLMAX[7:0]> > <RMCLLMIN[7:0]> <RMCLCMIN[7:0]> > <RMCLLMAX[7:0]>
High 幅のみ	<RMCLCMAX[7:0]> > <RMCLCMIN[7:0]> <RMCLLMAX[7:0]> = 0x00000000 <RMCLLMIN[7:0]> = don' t care
リーダなし	<RMCLCMAX[7:0]> = 0x00000000 <RMCLCMIN[7:0]> = don' t care <RMCLLMAX[7:0]> = don' t care <RMCLLMIN[7:0]> = don' t care

14.2.8 リモコン受信コントロールレジスタ 2 《RMCR2》

	31	30	29	28	27	26	25	24
bit Symbol	RMCLIEN	RMCEDIEN	—	—	—	—	RMCLD	RMCPHM
Read/Write	R/W	R/W	R				R/W	R/W
リセット後	0	0	0				0	0
機能	リーダ検出割込み発生許可 0: 割込み発生しない 1: 割込み発生する	リモコン入力立下リエッジ割込み 0: 割込み発生しない 1: 割込み発生する	リードすると“0”が読めます。				リーダありとリーダなしのリモコン信号を両方受信可能なモード 0: 禁止 1: 許可	位相方式のリモコン受信の設定 0: 位相方式のリモコン信号を受信しない(周期方式で受信) 1: 周期固定の位相方式のリモコン信号を受信する
	23	22	21	20	19	18	17	16
bit Symbol	—	—	—	—	—	—	—	—
Read/Write	R							
リセット後	0							
機能	リードすると“0”が読めます。							
	15	14	13	12	11	10	9	8
bit Symbol	RMCLL7	RMCLL6	RMCLL5	RMCLL4	RMCLL3	RMCLL2	RMCLL1	RMCLL0
Read/Write	R/W							
リセット後	1							
機能	Low 幅の検出による受信終了/割込み発生のタイミングの設定 00000000~11111110: $RMCLL \times 1/fs[s]$ 11111111: Low 幅検出で受信終了/割込みの設定をしない							
	7	6	5	4	3	2	1	0
bit Symbol	RMCDMAX7	RMCDMAX6	RMCDMAX5	RMCDMAX4	RMCDMAX3	RMCDMAX2	RMCDMAX1	RMCDMAX0
Read/Write	R/W							
リセット後	1							
機能	データ bit の周期 MAX で受信終了/割込み発生の設定 00000000~11111110: $RMCDMAX \times 1/fs[s]$ で受信終了/割込み発生 11111111: データ bit の周期 MAX で受信終了/割込み発生しない							

<RMCLIEN>: リーダ検出での割込み発生の許可をします。

<RMCEDIEN>: リモコン入力立下リエッジ割込み発生の許可をします。

<RMCLD>: リーダありとリーダなしのリモコン信号を両方受信可能なモードに設定します。

<RMCPHM>: 位相方式の受信モードの設定をします。周期が固定の位相方式のリモコン信号の場合は、“1”を設定。

<RMCLL[7:0]>: Low 幅の検出による受信終了/割込み発生のタイミングの設定をします。
<RMCLL7:0> = 1111111b のときは検出しません。設定期間の計算式は $RMCLL \times$

$1/f_s[s]$ 。

<RMCDMAX[7:0]>: データ bit の周期 MAX 検出のしきい値を設定します。データ bit 周期の値がしきい値以上であれば、検出となります。<RMCMAX[7:0]> = 1111111b のときは検出しません。検出のしきい値の計算式は $RMCDMAX \times 1/f_s[s]$ 。

14.2.9 リモコン受信コントロールレジスタ 3 《RMCR3》

	15	14	13	12	11	10	9	8
bit Symbol	—	RMCDATH6	RMCDATH5	RMCDATH4	RMCDATH3	RMCDATH2	RMCDATH1	RMCDATH0
Read/Write	R	R/W						
リセット後	0	0						
機能	リードすると“0”が読めます。	データ bit の 3 値判定のしきい値上位 $RMCDATH \times 1/fs[s]$						
	7	6	5	4	3	2	1	0
bit Symbol	—	RMCDATL6	RMCDATL5	RMCDATL4	RMCDATL3	RMCDATL2	RMCDATL1	RMCDATL0
Read/Write	R	R/W						
リセット後	0	0						
機能	リードすると“0”が読めます。	データ bit の 0/1 判別及び 3 値判定のしきい値下位 $RMCDATL \times 1/fs[s]$						

<RMCDATH[6:0]>: 位相方式のリモコン信号の 3 値判定の 1.5T と 2T のしきい値の設定をします。データ bit の測定結果がしきい値以上でデータを”10”、しきい値未満でデータを”01”と判別します。しきい値の計算式は、 $RMCDATH \times 1/fs[s]$ 。

<RMCDATL[6:0]>: データ bit の 0/1 判定のしきい値及び、位相方式のリモコン信号の 3 値判定の 1T と 1.5T のしきい値の設定をします。データ bit の 0/1 判定の場合、測定結果がしきい値以上でデータを”1”、しきい値未満でデータを”0”と判別します。位相方式のリモコン信号の 3 値判定の場合、データ bit の測定結果がしきい値以上でデータを”01”、しきい値未満でデータを”00”と判別します。0/1 判別のしきい値の計算式は、 $RMCDATL \times 1/fs[s]$ 。

(注 1)	リモコン受信コントロールレジスタ 2 《RMCR2》の<RMCPHM>=”0”のとき、<RMCDATH[6:0]>ビットは有効になりません。<RMCPHM>=”1”のときのみ有効となります。
-------	---

14.2.10 リモコン受信コントロールレジスタ4 《RMCR4》

	7	6	5	4	3	2	1	0
bit Symbol	RMCP0	—	—	—	RMCNC3	RMCNC2	RMCNC1	RMCNC0
Read/Write	R/W	R			R/W			
リセット後	0	0			0			
機能	リモコン入力信号の極性選択 0: 正極 1: 負極	リードすると“0”が読めます。			ノイズキャンセラのノイズ除去時間の設定 0000: ノイズ除去しない 0001~1111: $RMCNC \times 1/f_s[s]$			

<RMCP0>: リモコン入力信号の入力極性の選択をします。

<RMCNC[3:0]>: ノイズキャンセラのノイズ除去時間の設定をします。<RMCNC[3:0]> = 0000b のときはノイズ除去しません。ノイズ除去時間の計算式は、 $RMCNC \times 1/f_s[s]$ 。

14.2.11 リモコン受信ステータスレジスタ《RMCRSTAT》

	15	14	13	12	11	10	9	8
bit Symbol	RMCLIF	RMCLOIF	RMCDMAXIF	RMCEDIF	—	—	—	—
Read/Write	R	R	R	R	R			
リセット後	0	0	0	0	0			
	リーダ検出割込み要因フラグ 0: 要因ではない 1: 要因	Low 幅検出割込み要因フラグ 0: 要因ではない 1: 要因	データ bit 周期 MAX 割込み要因フラグ 0: 要因ではない 1: 要因	リモコン入力の下リエッジ割込み要因フラグ 0: 要因ではない 1: 要因	リードすると“0”が読めます。			
	7	6	5	4	3	2	1	0
bit Symbol	RMCLDR	RMCRNUM6	RMCRNUM5	RMCRNUM4	RMCRNUM3	RMCRNUM2	RMCRNUM1	RMCRNUM0
Read/Write	R	R						
リセット後	0	0						
機能	リーダ検出 0: なし 1: あり	リモコン受信したデータ bit 数 0000000: データ bit なし(リーダのみ) 0000001~1001000: 1~72bit 受信 1001001~1111111: 73bit 以上						

<RMCLIF>: 割込み発生要因がリーダ検出であることを示します。

<RMCLOIF>: 割込み発生要因が Low 幅検出であることを示します。

<RMCDMAXIF>: 割込み発生要因がデータ bit 周期 MAX 検出であることを示します。

<RMCEDIF>: 割込み発生要因がリモコン入力信号の立下リエッジであることを示します。

<RMCLDR>: リモコン受信した波形のリーダ検出を示します。

<RMCRNUM[6:0]>: リモコン受信したデータ bit 数を示します。データ bit 数は受信終了後に格納され、受信途中の bit 数はモニタすることは出来ません。

(注 1) このレジスタは割込み発生時に最新状態に更新されます。

このレジスタへの書き込み動作は無視されます。

(注 2) 73bit 以上のデータを受信しても受信終了が発生しない場合は、データを受信し続けます。このとき、データバッファの内容については保証しません。

14.3 動作説明

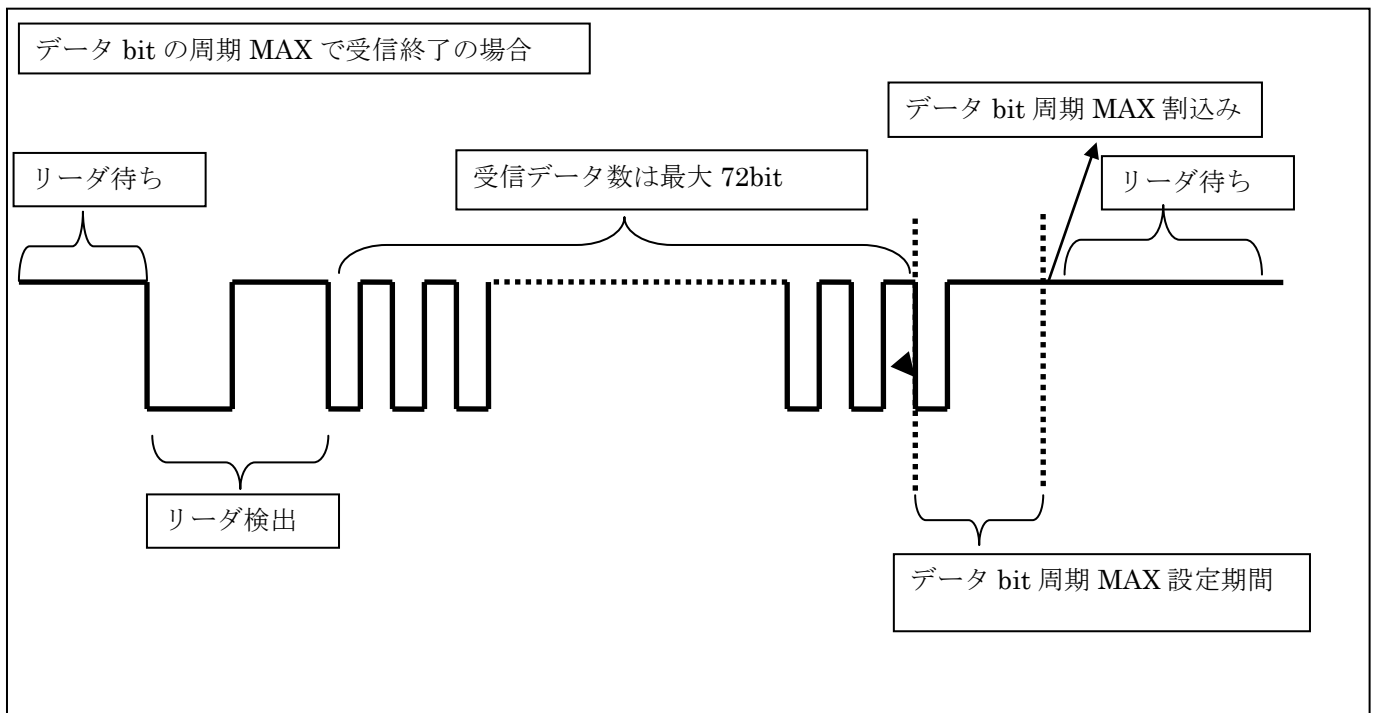
14.3.1 リモコン受信

14.3.1.1 サンプリングクロック

リモコン信号のサンプリングは、低速クロック (fs) で行います。

14.3.1.2 基本動作

リーダ待ち状態でリーダを検出するとリーダ検出割込みを発生し、データ bit の受信を開始します。データ bit の 0/1 判定は周期で判別され、最大 72bit まで受信します。受信終了はデータ bit の周期 MAX 検出、Low 幅の検出のどれかで行い、受信終了後リーダ待ちとなります。受信終了時にはデータバッファ、ステータスレジスタ共に最新の状態が保持されています。



14.3.1.3 リモコン受信の準備

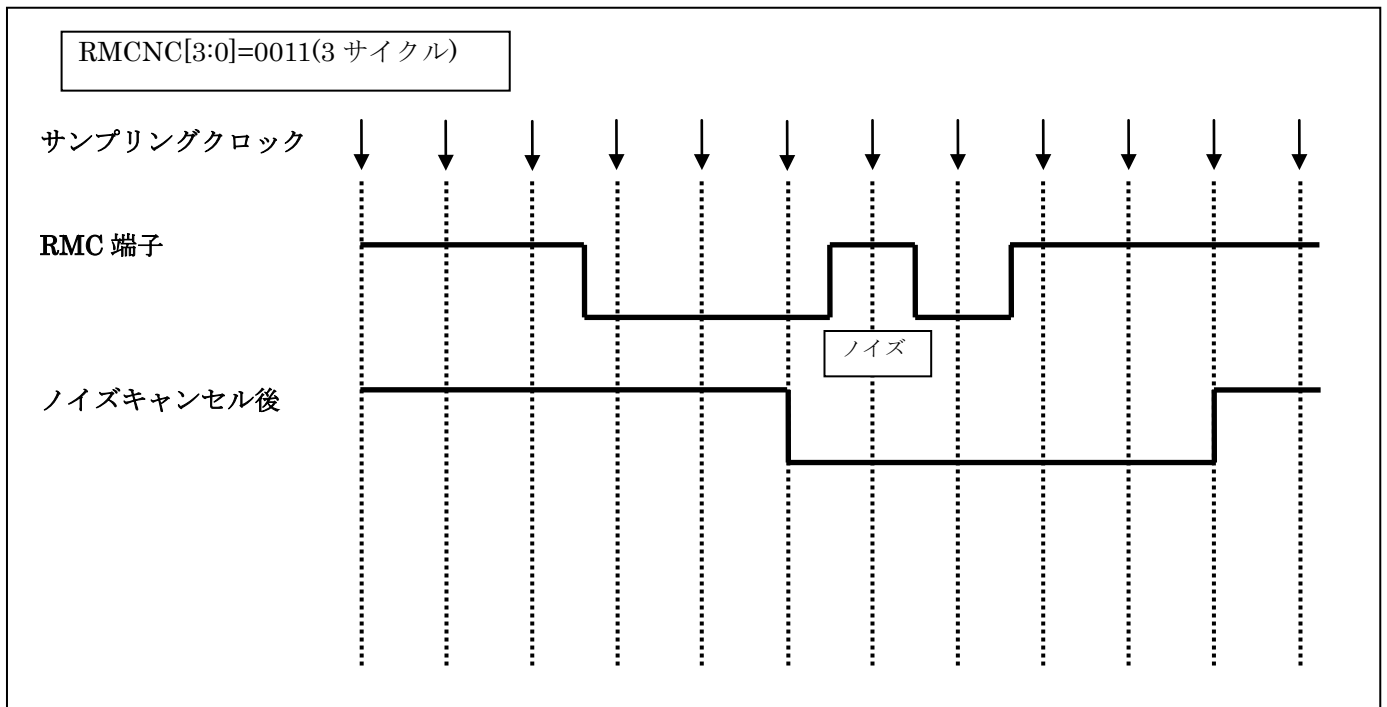
受信を開始する前に、リモコン受信コントロールレジスタ (RMCR1,2,3) でリモコン受信動作の設定を行います。

(1) ノイズキャンセル時間の設定

RMCR4 レジスタの<RMCNC[3:0]>ビットでノイズキャンセル時間を設定します。

サンプリングクロックの立ち上がりエッジごとにリモコン信号を観測し、現在“High”であれば、RMCNC で設定されたサイクル分の“Low”が観測されたときに信号が“Low”に変化したと認識し、現在“Low”であれば、RMCNC で設定されたサイクル分の“High”が観測されたとき“High”に変化したと認識します。

次の図は、ノイズキャンセルの設定を RMCNC[3:0]=0011 (3 サイクル) とした場合の動作です。ノイズキャンセル後の信号は、“High”の状態から、“Low”が3 サイクル観測されたところで“Low”に変化し、“Low”の状態から、“High”が3 サイクル観測されたところで“High”に変化します。

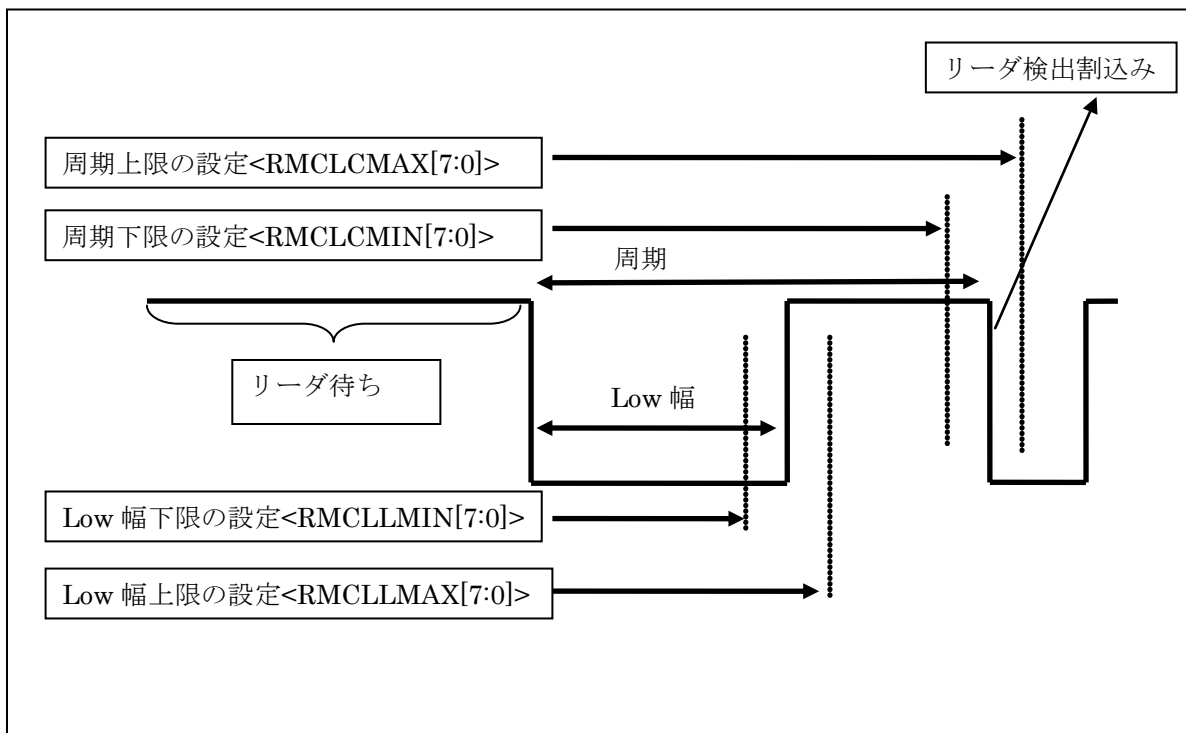


(2) リーダ検出の設定

リーダー検出の設定は、リーダーの周期と Low 幅の値を設定します。RMCRCR 1 レジスタの <RMCLLMAX[7:0]>、<RMCLLMAX[7:0]>、<RMCLCMIN[7:0]>、<RMCLCMAX[7:0]>ビットでリーダー検出のための設定を行います。RMCRCR1 レジスタを設定するときは、次の関係で行ってください。

リーダー種類	関係式
Low 幅+High 幅	<RMCLCMAX[7:0]> > <RMCLCMIN[7:0]> <RMCLLMAX[7:0]> > <RMCLLMIN[7:0]> <RMCLCMIN[7:0]> > <RMCLLMAX[7:0]>
High 幅のみ	<RMCLCMAX[7:0]> > <RMCLCMIN[7:0]> <RMCLLMAX[7:0]> = 0x00000000 <RMCLLMIN[7:0]> = don' t care
リーダーなし	<RMCLCMAX[7:0]> = 0x00000000 <RMCLCMIN[7:0]> = don' t care <RMCLLMAX[7:0]> = don' t care <RMCLLMIN[7:0]> = don' t care

次図はリーダー波形と RMCRCR 1 レジスタの設定について表しています。



リーダー検出時に割込みの発生を行いたい場合は、RMCRCR 2 レジスタの<RMCLIEN>ビットで設定します。リーダーなしのリモコン信号については、リーダー検出割込みの発生は出来ません。

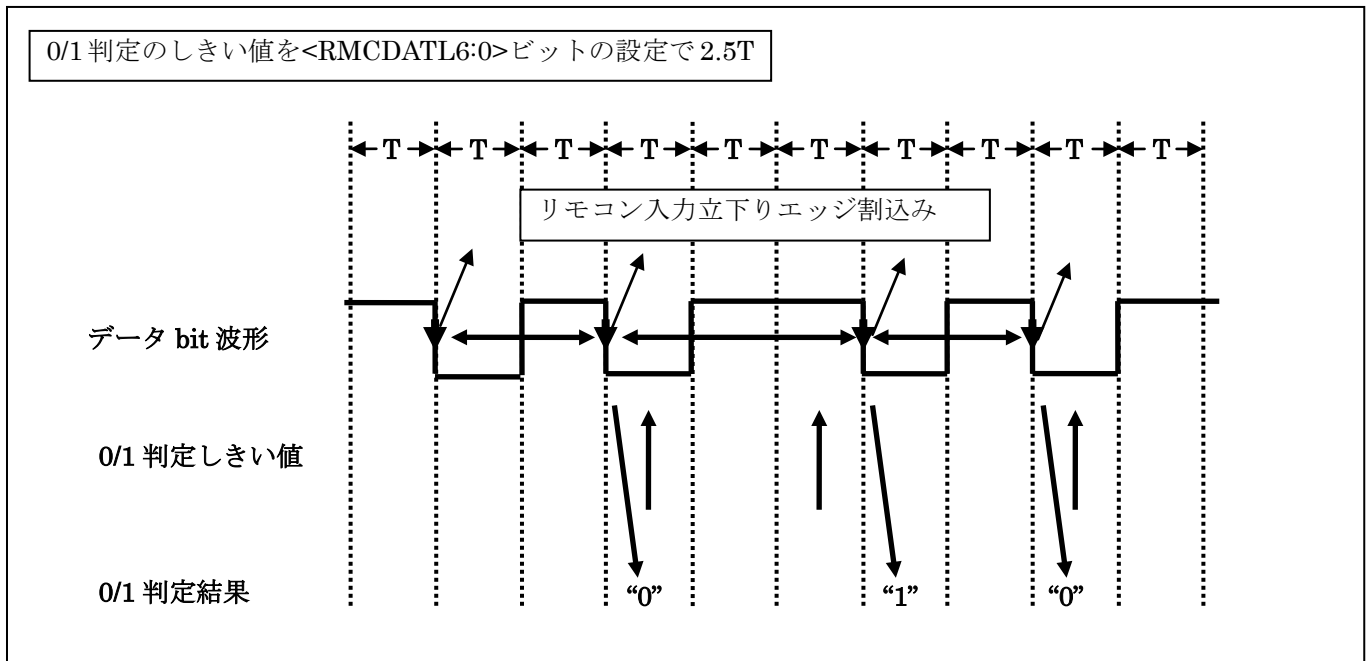
(3) データ bit の 0/1 判定の設定

データ bit の 0/1 判定は立下りエッジの周期に対して行います。

RMCRCR3 レジスタの<RMCDATL[6:0]>ビットでデータ bit の 0/1 判定のしきい値を設定します。0/1 判別はしきい値以上でデータ"1"、しきい値未満でデータ"0"となります。

RMCRCR2 レジスタの<RMCEDIEN>ビットを"1"にセットすることで、データ bit の立下りエッジ毎にリモコン入力立下りエッジ割込みを発生させることも出来ます。リモコン入力立下りエッジ割込みと 16 ビットタイマを用いることで、ソフトウェア的にリモコン判定を行うことも出来ます。

次図はデータ bit 0/1 判定について表しています。



位相方式のリモコン信号のデータ bit の判定は、14.3.1.10 周期固定の位相方式のリモコン信号の受信について で説明します。

(4) 受信終了の設定

受信終了はデータ bit の周期 MAX 検出及び、low 幅の検出を設定することで行います。複数要因設定の場合は、最初に検出した要因で受信終了となります。受信終了の設定は必ず行うようにして下さい。

①データ bit の周期 MAX で受信終了する場合

RMCR2 レジスタの<RMCDMAX[7:0]>ビットでデータ bit 周期 MAX の設定をします。データ bit の立下りエッジ周期の値が、<RMCDMAX[7:0]>ビットで設定されたデータ bit 周期 MAX のしきい値以上のときデータ bit 周期 MAX 検出で受信終了/割込み発生となります。

②Low 幅検出で受信終了する場合

RMCR2 レジスタの<RMCLL[7:0]>ビットで Low 幅の検出による受信終了の設定をします。データ bit が立下り後、設定期間を超えて Low のままだと Low 幅検出し、受信終了/割込み発生となります。

14.3.1.4 受信許可

RMCR1 レジスタ、RMCR2 レジスタ、RMCR3 レジスタ、RMCR4 レジスタの設定終了後、RMCREN レジスタの<RMCREN>ビットを受信許可に設定することで受信待ち状態になり、リーダを検出すると受信動作を開始します。

(注) 受信動作中に RMCR1, 2, 3, 4 レジスタの設定が変更されると正しく受信できない可能性があります。受信許可中に設定変更を行う場合は注意して行ってください。

14.3.1.5 受信動作

リーダ検出をすると、RMCRSTAT レジスタの<RMCRCLDR>ビットがセットされます。この時、RMCR2 レジスタの<RMCLIEN>ビットを設定しておくこと、リーダ検出時にリーダ検出割込みが発生します。リーダ割込み発生時には、RMCRSTAT レジスタの<RMCRLIF>がセットされます。

リーダ検出後、データ bit の 0/1 判定を順次行い、結果を RMCRBUF 1, 2, 3 レジスタに最大 72bit まで格納します。RMCR2 レジスタの<RMCEDIEN>ビットを設定しておくこと、データ bit の立下りエッジ毎にリモコン入力立下りエッジ割込みが発生します。リモコン入力立下りエッジ割込み発生時には、RMCRSTAT レジスタの<RMCEDIF>がセットされます。

受信動作はデータ bit 周期 MAX の検出及び、Low 幅の検出が設定値になったところで終了となり、受信終了/割込み発生となります。

受信終了時の状態は、リモコン受信ステータスレジスタを読み出すことで知ることが出来ます。

受信終了後、リーダ待ちとなります。

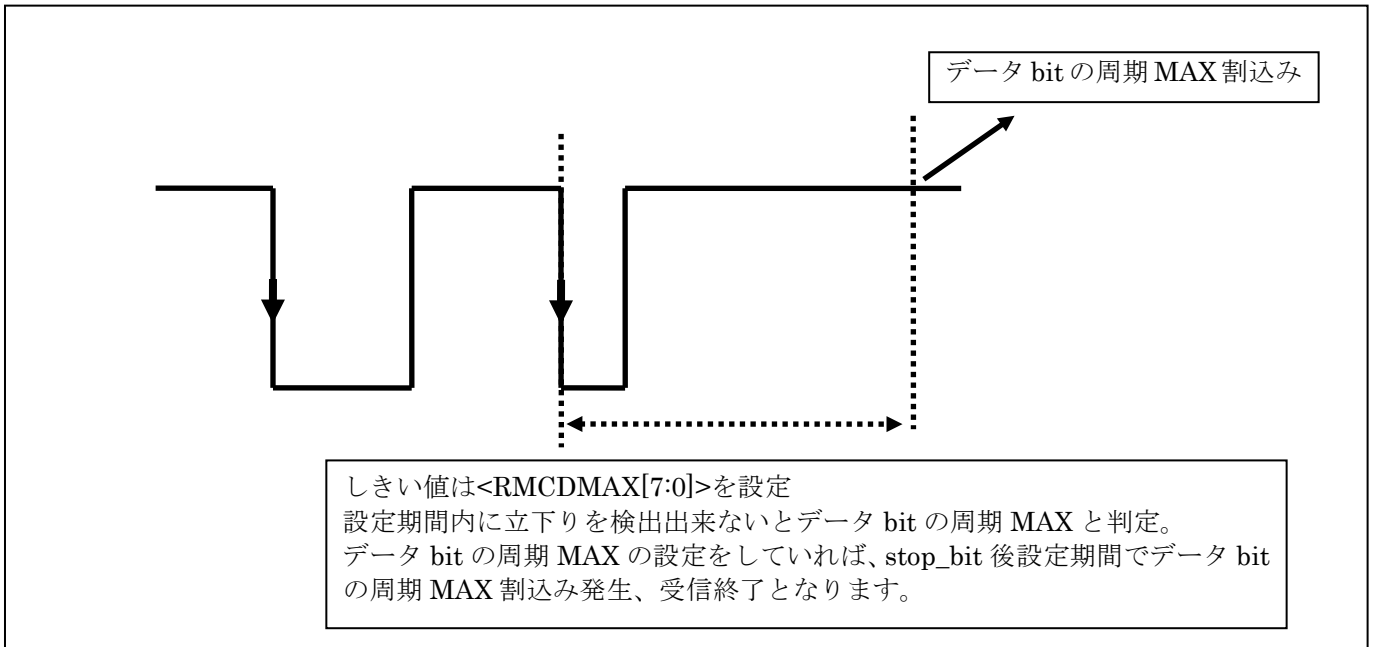
データ bit のみのリモコン信号を受信する設定をすると、リーダの検出はせずに最初からデータとして扱い、受信を行います。

受信したデータを読み出す前に、次の受信が終了すると受信データは書き換わります。

14.3.1.6 受信終了

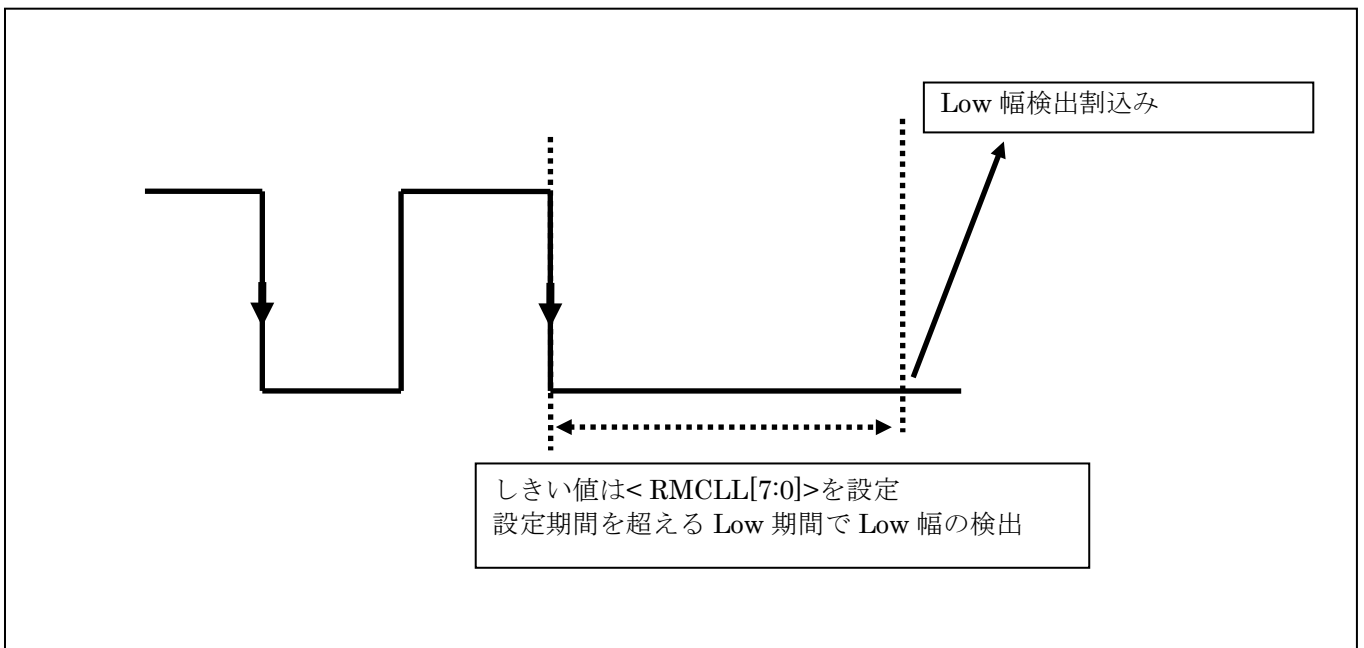
①データ bit の周期 MAX 検出で受信終了する場合

データ bit の周期 MAX の検出で受信終了/データ bit の周期 MAX 割込みを発生します。割込み発生後に RMCSTAT レジスタの値の <RMCDMAXIF> ビットがセットされます。



②Low 幅検出で受信終了する場合

Low 幅検出により、受信終了/Low 幅検出割込みを発生します。割込み発生後に RMCSTAT レジスタの <RMCLOIF> ビットに "1" がセットされます。



73bit 以上のデータを受信しても受信終了が発生しない場合は、データを受信し続けます。このとき、データバッファの内容については保証しません。

受信終了後、RMCRSTST レジスタを読み出すことで受信終了時の状態を知ることが出来ます。RMCRSTST レジスタと受信後の状態は次のようになっています。

<RMCRLDR>	<RMCRNUM[6:0]>	受信後の状態
0	0000001~1001000	リーダーなしリモコン信号受信(データ bit 数は 1~72bit)
0	1001001~1111111	リーダーなしリモコン信号受信(データ bit 数は 73bit 以上)
1	0000000	リーダーのみ
1	0000001~1001000	リーダーありリモコン信号受信(データ bit 数は 1~72bit)
1	1001001~1111111	リーダーありリモコン信号受信(データ bit 数は 73bit 以上)

14.3.1.7 受信の停止

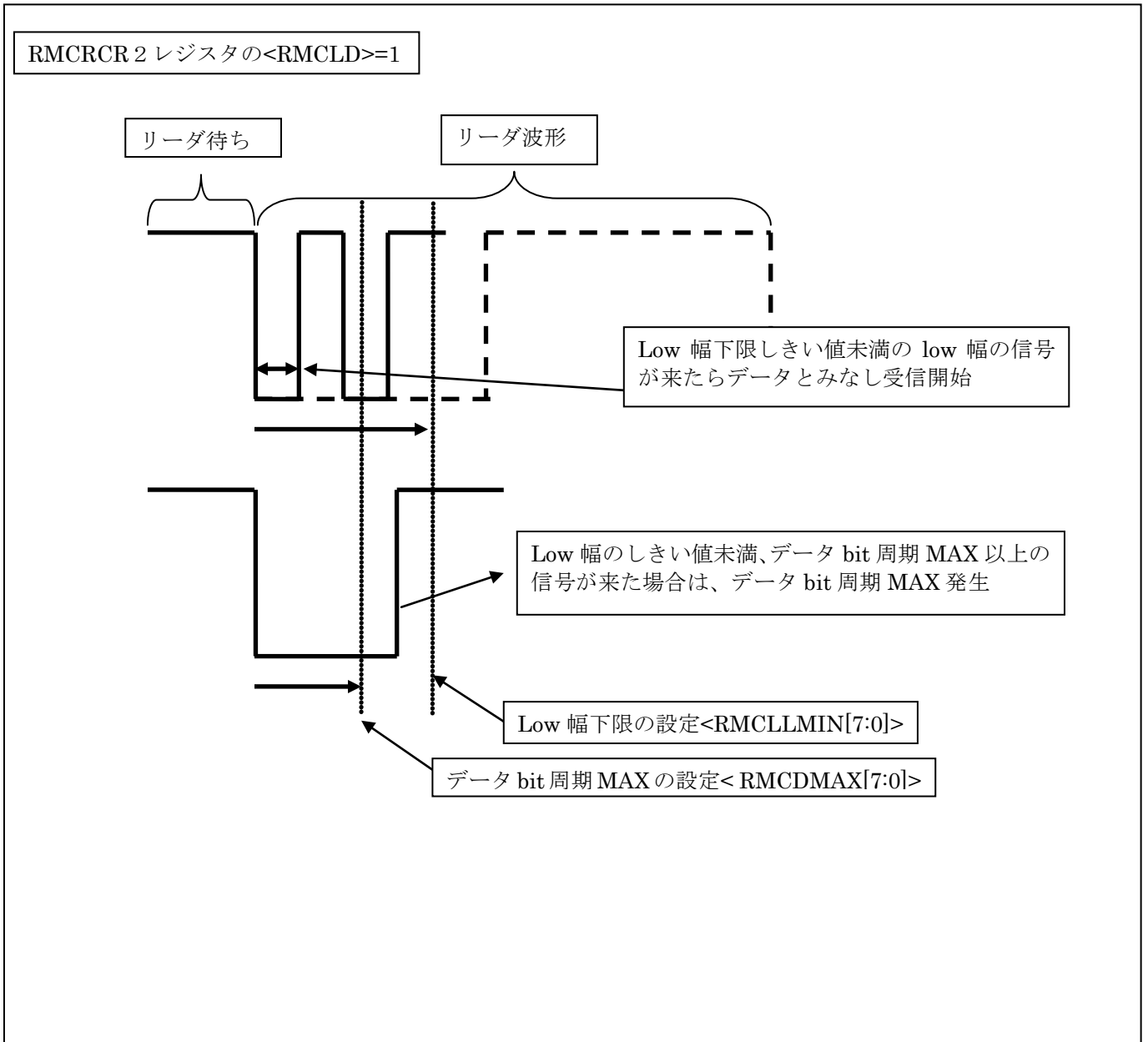
RMCREN レジスタの<RMCREN>ビットを“0” (受信禁止) に設定すると受信動作を停止します。

受信動作中に禁止の設定を行った場合、直ちに受信動作を停止し、それまでに受信したデータは破棄されます。

14.3.1.8 リーダ待ちの状態でのリーダなしのリモコン信号の受け方

RMCR2CR 2 レジスタの<RMCLD>を設定することでリーダありとリーダなしのリモコン信号を両方受信することが可能になります。RMCR2CR 2 レジスタの<RMCLD>を設定すると、RMCR1CR 1 レジスタの<RMCLLMAX[7:0]>以下の Low 幅の信号を受信するとデータと判断し受信を開始し、最終 bit まで受信を行います。

この設定を使用する場合、データ bit の 0/1 判定、エラー検出、受信終了の設定はリーダあり/リーダなしデータで全て共通となりますので、受信可能なリモコン信号には制限があります。



14.3.1.9 Low 幅のみのリーダで始まるリモコン信号の受信について

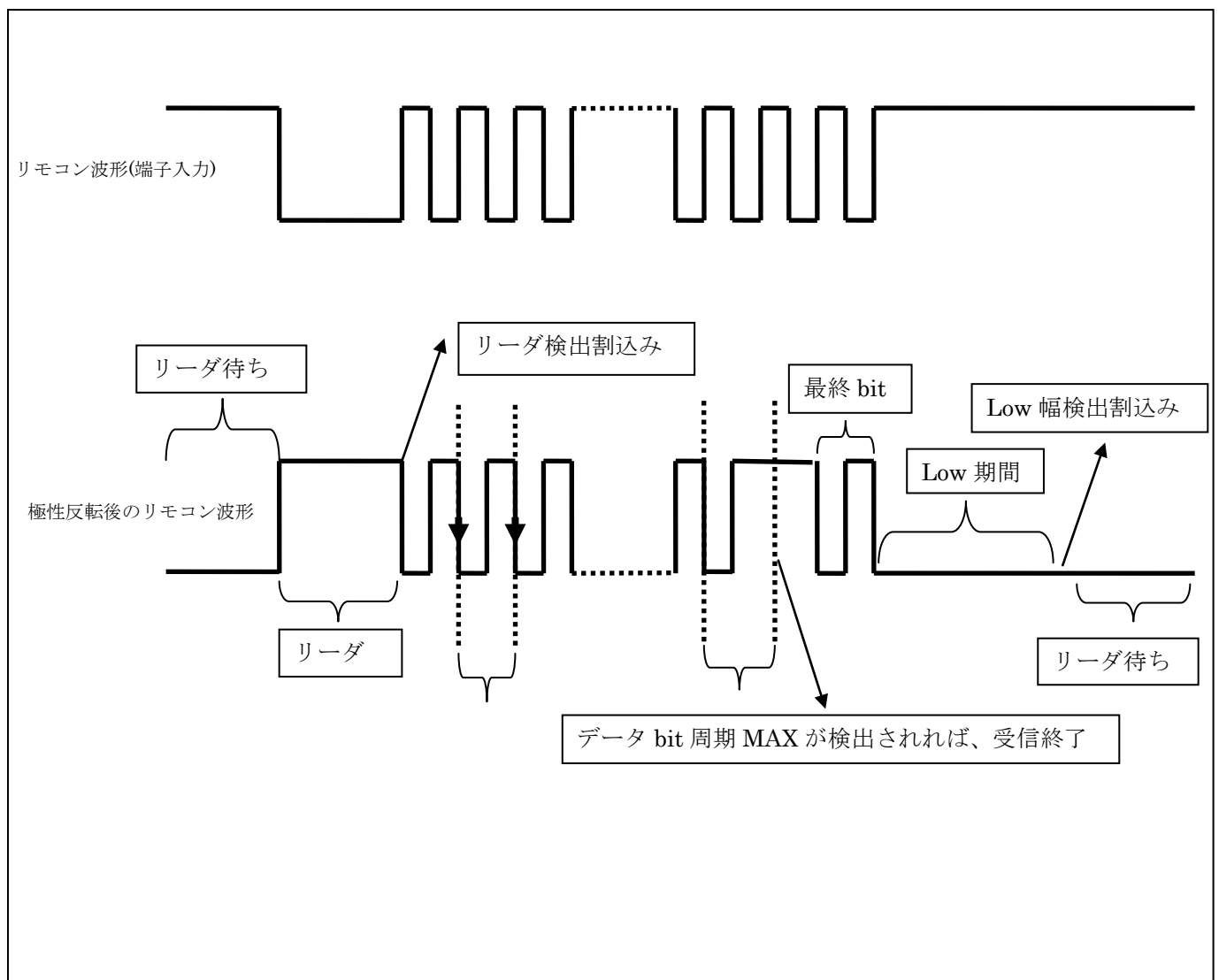
Low 幅のみのリーダで始まるリモコン信号は次図にあるように、リーダが Low 幅のみで始まり、データ bit の周期が立上り期間で構成される信号です。データ bit の測定を立下りエッジの周期で行う必要があるため、RMCR4 レジスタの<RMCP0>="1"に設定し、反転した信号にして入力します。

リーダの検出の設定は Low 幅のみのため、RMCR1 レジスタの設定を、<RMCLLMAX[7:0]> = 0x00000000、<RMCLCMAX[7:0]> > <RMCLCMIN[7:0]> の関係で設定します。この場合、<RMCLLMIN[7:0]>の値は don't care となります。

データ 0/1 判定の設定は、リモコン受信コントロールレジスタ 3 <RMCDATL[6:0]>で 0/1 判別のしきい値を設定します。

リモコン受信コントロールレジスタ 2 の<RMCDMAX[7:0]>でデータ bit の周期 MAX の設定をします。

受信終了の設定は、リモコン受信コントロールレジスタ 2 の<RMCDMAX[7:0]>でデータ bit の周期 MAX の設定と<RMCLL[7:0]>で Low 幅検出の設定をします。データ bit の周期 MAX 検出及び、最終 bit 後に設定された Low 期間確認後に受信終了/割込み発生し、リーダ待ちとなります。



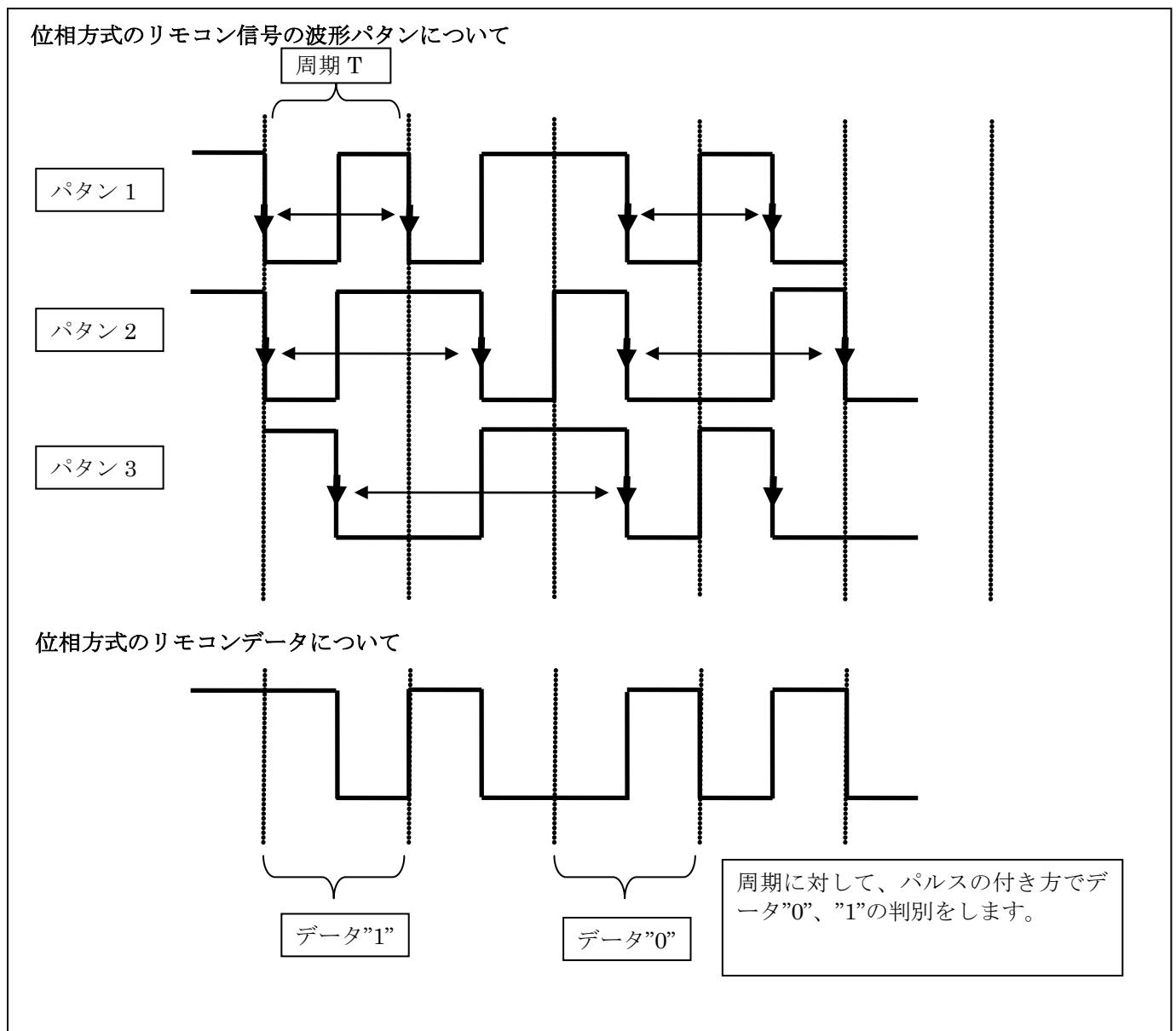
14.3.1.10 周期固定の位相方式のリモコン信号の受信について

周期固定の位相方式のリモコン信号は、データの変化のパターンが3通りに分けられます。しきい値を2つ設定することでパタンの判別を行い、ハードウェアでリモコンデータに変換を行い、受信終了後、受信データをリモコンデータの形式で RMCRCBUF1,2,3 に格納します。

周期固定の位相方式のリモコン受信は、RMCRCR2<RMCPHM>="1"に設定することで有効になります。しきい値の設定は、RMCRCR3 レジスタの<RMCDATL[6:0]>ビット、<RMCDATH[6:0]>ビットで行います。次図にある、3通りのデータの変化のパターンに対して、2つのしきい値を決めます。3通りのパターンはそれぞれ周期 T に対して、1T、1.5T、2T となり、しきい値の設定は下表の様になります。

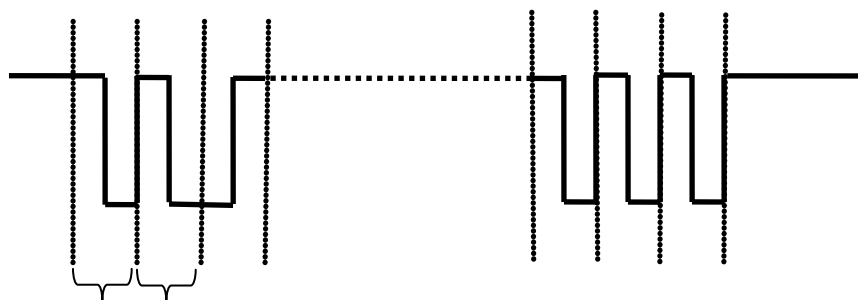
周期固定の位相方式のリモコン信号の判別は、3通りのデータの変化のパターンと直前の周期のデータが必要です。周期固定の位相方式のリモコン信号は必ず、データ "1" で始まる必要があります。

	判別内容	しきい値	設定レジスタ
しきい値1	パターン1とパターン2	1T~1.5T	RMCRCR3 レジスタの<RMCDATL[6:0]>
しきい値2	パターン2とパターン3	1.5T~2T	RMCRCR3 レジスタの<RMCDATH[6:0]>



位相方式のリモコン信号について

リモコン信号



開始の 2 bit のデータは必ず”11”である必要があります。

15 アナログ/デジタルコンバータ

TMPM330 は、12 チャンネルのアナログ入力を持つ、10 ビット逐次変換方式アナログ/デジタルコンバータ (A/D コンバータ) を内蔵しています。

図 15-1に、A/D コンバータのブロック図を示します。

12 チャンネルのアナログ入力端子 (ANO~AN11) は、入出力専用ポートと兼用です。

(注) IDLE、STOP モードにより電源電流を低減させる場合、以下の条件で使用される場合には、A/D コンバータの動作を停止して、スタンバイモードに移行する命令を実行してください。

- 1) ADMOD1<I2AD>=" 0" で IDLE モードへ移行する場合
- 2) STOP モードへ移行する場合

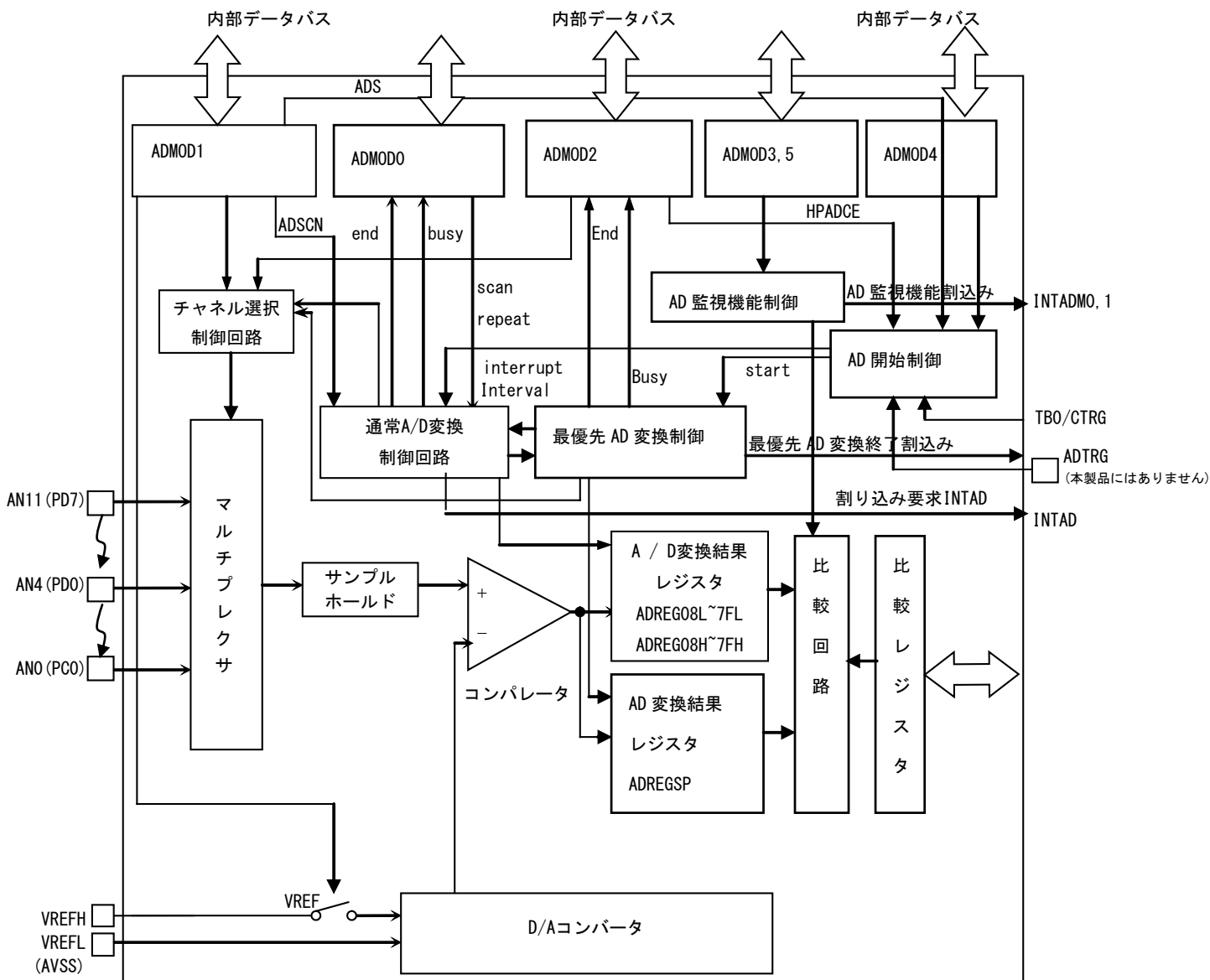


図15-1 A/D コンバータのブロック図

15.1 レジスタ一覧

A/D コンバータの制御レジスタとアドレスは以下のとおりです。

レジスタ名		アドレス
A/D 変換クロック設定レジスタ	ADCLK	0x4003_0000
A/D モードコントロールレジスタ 0	ADMOD0	0x4003_0004
A/D モードコントロールレジスタ 1	ADMOD1	0x4003_0008
A/D モードコントロールレジスタ 2	ADMOD2	0x4003_000C
A/D モードコントロールレジスタ 3	ADMOD3	0x4003_0010
A/D モードコントロールレジスタ 4	ADMOD4	0x4003_0014
A/D モードコントロールレジスタ 5	ADMOD5	0x4003_0018
A/D 変換精度設定レジスタ (注)	ADCBAS	0x4003_0020
A/D 変換結果下位レジスタ 08L	ADREG08L	0x4003_0030
A/D 変換結果上位レジスタ 08H	ADREG08H	0x4003_0031
A/D 変換結果下位レジスタ 19L	ADREG19L	0x4003_0034
A/D 変換結果上位レジスタ 19H	ADREG19H	0x4003_0035
A/D 変換結果下位レジスタ 2AL	ADREG2AL	0x4003_0038
A/D 変換結果上位レジスタ 2AH	ADREG2AH	0x4003_0039
A/D 変換結果下位レジスタ 3BL	ADREG3BL	0x4003_003C
A/D 変換結果上位レジスタ 3BH	ADREG3BH	0x4003_003D
A/D 変換結果下位レジスタ 4CL	ADREG4CL	0x4003_0040
A/D 変換結果上位レジスタ 4CH	ADREG4CH	0x4003_0041
A/D 変換結果下位レジスタ 5DL	ADREG5DL	0x4003_0044
A/D 変換結果上位レジスタ 5DH	ADREG5DH	0x4003_0045
A/D 変換結果下位レジスタ 6EL	ADREG6EL	0x4003_0048
A/D 変換結果上位レジスタ 6EH	ADREG6EH	0x4003_0049
A/D 変換結果下位レジスタ 7FL	ADREG7FL	0x4003_004C
A/D 変換結果上位レジスタ 7FH	ADREG7FH	0x4003_004D
A/D 変換結果下位レジスタ SP	ADREGSPL	0x4003_0050
A/D 変換結果上位レジスタ SP	ADREGSPH	0x4003_0051
A/D 変換結果比較下位レジスタ 0	ADCMPOL	0x4003_0054
A/D 変換結果比較上位レジスタ 0	ADCMP0H	0x4003_0055
A/D 変換結果比較下位レジスタ 1	ADCMP1L	0x4003_0058
A/D 変換結果比較上位レジスタ 1	ADCMP1H	0x4003_0059

注) A/D コンバータを使用する場合、変換精度を高くするために初期化後の設定の際に必ず下記設定 (ADCBAS レジスタへの 0x58h の書き込み) を1回行ってください。

0x4003_0020 = 0x58

		7	6	5	4	3	2	1	0
ADCBAS	bit Symbol								
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	リセット後	0	0	1	1	1	0	0	0
	機能	“0” をラ イトして ください。	“1” をラ イトして ください。	“0” をラ イトして ください。	“1” をラ イトして ください。	“1” をラ イトして ください。	“0” をラ イトして ください。	“0” をラ イトして ください。	“0” をラ イトして ください。

15.2 レジスタ詳細

A/D コンバータは、A/D モードコントロールレジスタ (ADMOD0、ADMOD1、ADMOD2、ADMOD3、ADMOD4、ADMOD5) により制御されています。また、A/D 変換結果は、A/D 変換結果上位/下位レジスタ ADREG08H/L、～ADREG7FH/L の 16 個のレジスタに格納されます。また、最優先変換結果は ADREGSPH/L に格納されます。

各レジスタの内容を説明します。

A/D モードコントロールレジスタ 0

	7	6	5	4	3	2	1	0
ADMOD0	bit Symbol	EOCFN	ADBFN	ITM1	ITM0	REPEAT	SCAN	ADS
	Read/Write	R		R	R/W			
	リセット後	0	0	0	0	0	0	0
機 能	通常 A/D 変換終了フラグ 0: 変換前または変換中 1: 終了	通常 A/D 変換 BUSY フラグ 0: 変換停止 1: 変換中	リードすると "0" が読めます	チャンネル固定リピート変換モード時の割り込み指定	チャンネル固定リピート変換モード時の割り込み指定	リピートモード指定 0: シングル変換モード 1: リピート変換モード	スキャンモード指定 0: チャンネル固定モード 1: チャンネルスキャンモード	A/D 変換スタート 0: Don't care 1: 変換開始 リードすると常に "0" が読み出されます。

→ チャンネル固定リピート変換モード時の A/D 変換割り込み指定

	チャンネル固定リピート変換モード <SCAN> = "0", <REPEAT> = "1"
00	1 回変換するごとに割り込み発生
01	4 回変換するごとに割り込み発生
10	8 回変換するごとに割り込み発生
11	設定禁止

図15-2 A/D モードコントロールレジスタ 0

(注) モード設定を行ってから、<ADS>ビットの設定を行ってください。

A/D モードコントロールレジスタ 1

	7	6	5	4	3	2	1	0
bit Symbol	VREFON	I2AD	ADSCN	—	ADCH3	ADCH2	ADCH1	ADCH0
Read/Write	R/W							
リセット後	0	0	0	0	0	0	0	0
機能	VREF 印加 制御 0: OFF 1: ON	IDLE 0: 停止 1: 動作	チャネルスキャン時の動作 モード設定 0: 4ch スキャン 1: 8ch スキャン	0 を記入してください。	アナログ入力チャネル選択			

アナログ入力チャネル選択

<ADCH[3:0]>	<SCAN>		
	0 チャネル固定	1 チャネルスキャン (ADSCN=0)	1 チャネルスキャン (ADSCN=1)
0000	AN0	AN0	AN0
0001	AN1	AN0~AN1	AN0~AN1
0010	AN2	AN0~AN2	AN0~AN2
0011	AN3	AN0~AN3	AN0~AN3
0100	AN4	AN4	AN0~AN4
0101	AN5	AN4~AN5	AN0~AN5
0110	AN6	AN4~AN6	AN0~AN6
0111	AN7	AN4~AN7	AN0~AN7
1000	AN8	AN 8	AN8
1001	AN9	AN8~AN9	AN8~AN9
1010	AN10	AN8~AN10	AN8~AN10
1011	AN11	AN8~AN11	AN8~AN11
1100	設定禁止		
1101			
1110			
1111			

- (注 1) AD 変換をスタートさせる場合は、かならず<VREFON>ビットに” 1” を書き込んだ後、内部基準電圧が安定するまでの 3 μ s 待ってから、ADMOD0<ADS>ビットに” 1” を書き込んでください。
- (注 2) AD 変換終了後にスタンバイモードへ移行する場合は、<VREFON>を” 0” に設定してください。

図15-3 A/D モードコントロールレジスタ 1

A/D モードコントロールレジスタ 2

	7	6	5	4	3	2	1	0
bit Symbol	EOCFHP	ADBFHP	HPADCE	—	HPADCH3	HPADCH2	HPADCH1	HPADCH0
Read/Write	R	R	R/W					
リセット後	0	0	0	0	0	0	0	0
機能	最優先 AD 変換終了 フラグ 0 : 変換前または変換中 1 : 終了	最優先 AD 変換 BUSY フラグ 0 : 変換停止 1 : 変換中	最優先変換の起動 0: Don't care 1: 変換開始 読み出しは常に0	"0"をかいてください。	最優先変換起動時のアナログ入力チャンネル選択			

<HPADCH[3:0]>	最優先変換時のアナログ入力チャンネル
0000	AN0
0001	AN1
0010	AN2
0011	AN3
0100	AN4
0101	AN5
0110	AN6
0111	AN7
1000	AN8
1001	AN9
1010	AN10
1011	AN11
1100	設定禁止
1101	
1110	
1111	

図15-4 A/D モードコントロールレジスタ 2

(注) チャンネル設定を行ってから、<HPADCE>ビットの設定を行ってください。

A/D モードコントロールレジスタ 3

ADMOD3

	7	6	5	4	3	2	1	0
bit Symbol			ADOBIC	REGS3	REGS2	REGS1	REGS0	ADOBISV
Read/Write	R/W	R	R/W					
リセット後	0	0	0	0	0	0	0	0
機能	0 をライトしてください	リードすると "0" が読めます	AD 監視機能割込みの設定 0 : 比較 Regi より小 1 : 比較 Regi より大	AD 監視機能が Enable 時に比較 Regi の内容と比較される A/D 変換結果格納 Regi の選択 BIT				AD 監視機能 0 : Disable 1 : Enable

<REGS[3:0]>	比較される AD 変換格納レジスタ
0000	ADREG08
0001	ADREG19
0010	ADREG2A
0011	ADREG3B
0100	ADREG4C
0101	ADREG5D
0110	ADREG6E
0111	ADREG7F
1XXX	ADREGSP

図15-5 A/D モードコントロールレジスタ 3

A/D モードコントロールレジスタ 5

ADMOD5		7	6	5	4	3	2	1	0	
	bit Symbol			ADOBIC	REGS3	REGS2	REGS1	REGS0	ADOBVS	
	Read/Write	R			R/W					
	リセット後	0	0	0	0	0	0	0	0	
	機能	リードすると "0"が読めます		AD 監視機能 割込みの設定 0 : 比較 Regi より小 1 : 比較 Regi より大	AD 監視機能が Enable 時に比較 Regi の内容と比較される A/D 変換結果格納 Regi の選択 BIT				AD 監視機能 0 : Disable 1 : Enable	

<REGS[3:0]>	比較される AD 変換格納レジスタ
0000	ADREG08
0001	ADREG19
0010	ADREG2A
0011	ADREG3B
0100	ADREG4C
0101	ADREG5D
0110	ADREG6E
0111	ADREG7F
1XXX	ADREGSP

図15-6 A/D モードコントロールレジスタ 5

A/D モードコントロールレジスタ 4

	7	6	5	4	3	2	1	0
bit Symbol	HADHS	HADHTG	ADHS	ADHTG			ADRST1	ADRST0
Read/Write	R/W				R		W	W
リセット後	0	0	0	0	0		0	0
機能	最優先 AD 変換の HW 起動ソース 0: 外部トリガ 1: TB5RG0 一致	最優先 AD 変換の HW 起動 0: Disable 1: Enable	通常 AD 変換の HW 起動ソース 0: 外部トリガ 1: TB6RG0 一致	通常 AD 変換の HW 起動 0: Disable 1: Enable	リードすると"0"が読めます		10→01のライトで ADC を software reset する	

(注 1) 16 ビットタイマの一致トリガ<ADHTG>、<HADHTG>に"1"を設定して H/W 起動リソースによる AD 変換を行う場合、

- ① タイマ停止中に
- ② H/W のソースを選択 <ADHS>、<HADHS>
- ③ AD 変換の H/W 起動をイネーブル <ADHTG>、<HADHTG>
- ④ タイマ動作

の順に設定することにより、一定間隔での AD 起動が可能となります。

(注 2) 最優先 AD 変換、通常 AD 変換設定は同時に行わないで下さい。

(注 3) 最優先 AD 変換の HW 起動ソースに外部トリガを使用しているときは、通常 AD 変換 HW 起動としては外部トリガを設定できません。

(注 4) ソフトウェアリセットを掛けた場合、他のビットは初期化されますので、モードレジスタの再設定が必要となります。

(注) 本製品には HW 起動ソースとしての外部トリガ入力はありませんので、<HADHS>、<ADHS>に"0"を設定することはできません。

図15-7 A/D モードコントロールレジスタ 4

A/D 変換結果下位レジスタ 08

	7	6	5	4	3	2	1	0
ADREG08L	bit Symbol	ADR01	ADR00				OVR0	ADR0RF
	Read/Write	R		R			R	R
	リセット後	0		0			0	0
	機能	A/D 変換結果下位 2ビット格納		リードすると"0"が読めます			Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有

A/D 変換結果上位レジスタ 08

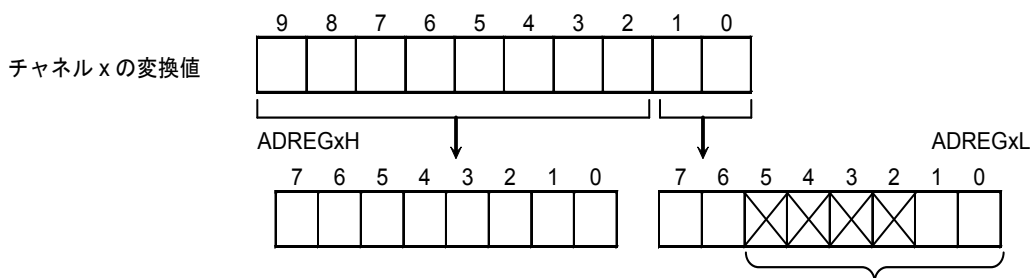
	7	6	5	4	3	2	1	0	
ADREG08H	bit Symbol	ADR09	ADR08	ADR07	ADR06	ADR05	ADR04	ADR03	ADR02
	Read/Write	R							
	リセット後	0							
	機能	A/D 変換結果上位 8ビット格納							

A/D 変換結果下位レジスタ 19

	7	6	5	4	3	2	1	0
ADREG19L	bit Symbol	ADR11	ADR10				OVR1	ADR1RF
	Read/Write	R		R			R	R
	リセット後	0		0			0	0
	機能	A/D 変換結果下位 2ビット格納		リードすると"0"が読めます			Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有

A/D 変換結果上位レジスタ 19

	7	6	5	4	3	2	1	0	
ADREG19H	bit Symbol	ADR19	ADR18	ADR17	ADR16	ADR15	ADR14	ADR13	ADR12
	Read/Write	R							
	リセット後	0							
	機能	A/D 変換結果上位 8ビット格納							



- ADREG08L/ADREG19L のビット 0 は、A/D 変換結果格納フラグ <ADRxRF> です。A/D 変換値が格納されると、“1”にセットされます。下位のレジスタ (ADREGxL) をリードすると、“0”にされます。
- ADREG08L/ADREG19L のビット 1 は over RUN flag <OVRx> です。両方の変換結果格納レジスタ (ADREGxH,ADREGxL) を Read する前に変換結果が上書きされると 1に Set されます。Flag の Read により 0にクリアされます。
- 変換結果格納レジスタをバイト単位で読む場合は、上位を先に読み出し、次に下位を読み出して下さい。

図15-8 A/D コンバータ変換結果レジスタ

A/D 変換結果下位レジスタ 2A

	7	6	5	4	3	2	1	0	
ADREG2AL	bit Symbol		ADR21	ADR20				OVR2	ADR2RF
	Read/Write		R		R			R	R
	リセット後		0		0			0	0
	機能		A/D 変換結果下位 2ビット格納			リードすると"0"が読めます		Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有

A/D 変換結果上位レジスタ 2A

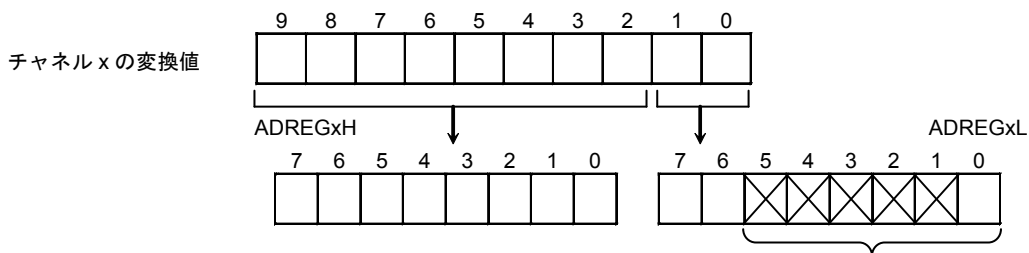
	7	6	5	4	3	2	1	0										
ADREG2AH	bit Symbol		ADR29	ADR28	ADR27	ADR26	ADR25	ADR24	ADR23	ADR22								
	Read/Write										R							
	リセット後										0							
	機能										A/D 変換結果上位 8ビット格納							

A/D 変換結果下位レジスタ 3B

	7	6	5	4	3	2	1	0	
ADREG3BL	bit Symbol		ADR31	ADR30				OVR3	ADR3RF
	Read/Write		R		R			R	R
	リセット後		0		0			0	0
	機能		A/D 変換結果下位 2ビット格納			リードすると"0"が読めます		Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有

A/D 変換結果上位レジスタ 3B

	7	6	5	4	3	2	1	0										
ADREG3BH	bit Symbol		ADR39	ADR38	ADR37	ADR36	ADR35	ADR34	ADR33	ADR32								
	Read/Write										R							
	リセット後										0							
	機能										A/D 変換結果上位 8ビット格納							



- ADREG2AL/ADREG3BL のビット 0 は、A/D 変換結果格納フラグ <ADR_xRF> です。A/D 変換値が格納されると、“1” にセットされます。下位のレジスタ (ADREG_xL) をリードすると、“0” にされます。
- ADREG2AL/ADREG3BL のビット 1 は over RUN flag<OVR_x>です。両方の変換結果格納レジスタ (ADREG_xH,ADREG_xL) を Read する前に変換結果が上書きされると 1 に Set されます。Flag の Read により 0 にクリアされます。
- 変換結果格納レジスタをバイト単位で読む場合は、上位を先に読み出し、次に下位を読み出してください。

図15-9 A/D コンバータ変換結果レジスタ

A/D 変換結果下位レジスタ 4C

	7	6	5	4	3	2	1	0
ADREG4CL	ADR41	ADR40					OVR4	ADR4RF
bit Symbol	R		R				R	R
Read/Write	0		0				0	0
リセット後	A/D 変換結果下位 2ビット格納		リードすると"0"が読めます				Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有
機能								

A/D 変換結果上位レジスタ 4C

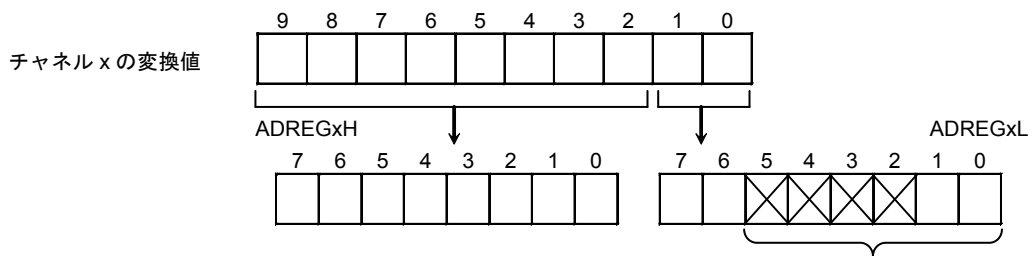
	7	6	5	4	3	2	1	0
ADREG4CH	ADR49	ADR48	ADR47	ADR46	ADR45	ADR44	ADR43	ADR42
bit Symbol	R							
Read/Write	0							
リセット後	A/D 変換結果上位 8ビット格納							
機能								

A/D 変換結果下位レジスタ 5D

	7	6	5	4	3	2	1	0
ADREG5DL	ADR51	ADR50					OVR5	ADR5RF
bit Symbol	R		R				R	R
Read/Write	0		0				0	0
リセット後	A/D 変換結果下位 2ビット格納		リードすると"0"が読めます				Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有
機能								

A/D 変換結果上位レジスタ 5D

	7	6	5	4	3	2	1	0
ADREG5DH	ADR59	ADR58	ADR57	ADR56	ADR55	ADR54	ADR53	ADR52
bit Symbol	R							
Read/Write	0							
リセット後	A/D 変換結果上位 8ビット格納							
機能								



- ADREG4CL/ADREG5DL のビット 0 は、A/D 変換結果格納フラグ <ADR_xRF> です。A/D 変換値が格納されると、“1”にセットされます。下位のレジスタ (ADREG_xL) をリードすると、“0”にされます。
- ADREG4CL/ADREG5DL のビット 1 は over Run flag<OVR_x>です。両方の変換結果格納レジスタ (ADREG_xH,ADREG_xL) を Read する前に変換結果が上書きされると 1に Set されます。Flag の Read により 0にクリアされます。
- 変換結果格納レジスタをバイト単位で読む場合は、上位を先に読み出し、次に下位を読み出してください。

図15-10 A/D コンバータ変換結果レジスタ

A/D 変換結果下位レジスタ 6E

	7	6	5	4	3	2	1	0
ADREG6EL	ADR61	ADR60					OVR6	ADR6RF
Read/Write	R		R				R	R
リセット後	0		0				0	0
機能	A/D 変換結果下位 2ビット格納		リードすると"0"が読めます				Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有

A/D 変換結果上位レジスタ 6E

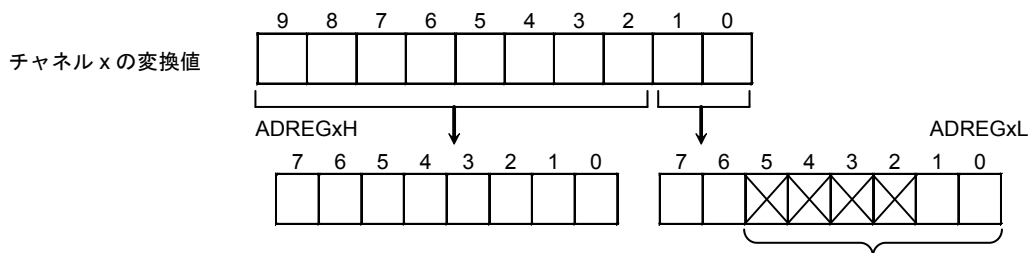
	7	6	5	4	3	2	1	0
ADREG6EH	ADR69	ADR68	ADR67	ADR66	ADR65	ADR64	ADR63	ADR62
Read/Write	R							
リセット後	0							
機能	A/D 変換結果上位 8ビット格納							

A/D 変換結果下位レジスタ 7F

	7	6	5	4	3	2	1	0
ADREG7FL	ADR71	ADR70					OVR7	ADR7RF
Read/Write	R		R				R	R
リセット後	0		0				0	0
機能	A/D 変換結果下位 2ビット格納		リードすると"0"が読めます				Over RUN Flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有

A/D 変換結果上位レジスタ 7F

	7	6	5	4	3	2	1	0
ADREG7FH	ADR79	ADR78	ADR77	ADR76	ADR75	ADR74	ADR73	ADR72
Read/Write	R							
リセット後	0							
機能	A/D 変換結果上位 8ビット格納							



- ADREG6EL/ADREG7FL のビット 0 は、A/D 変換結果格納フラグ <ADR_xRF> です。A/D 変換値が格納されると、“1”にセットされます。下位のレジスタ (ADREG_xL) をリードすると、“0”にされます。
- ADREG6EL/ADREG7FL のビット 1 は over Run flag <OVR_x> です。両方の変換結果格納レジスタ (ADREG_xH,ADREG_xL) を Read する前に変換結果が上書きされると 1 に Set されます。Flag の Read により 0 にクリアされます。
- 変換結果格納レジスタをバイト単位で読む場合は、上位を先に読み出し、次に下位を読み出してください。

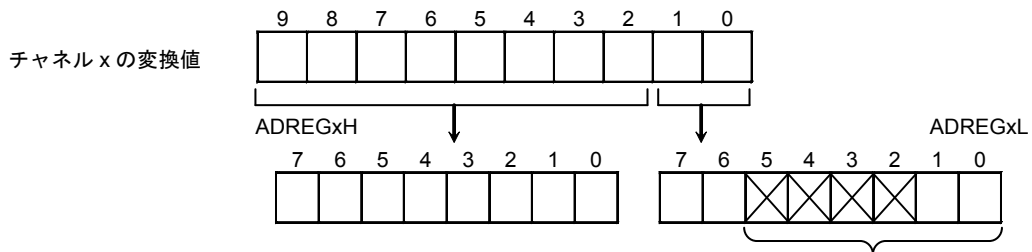
図15-11 A/D コンバータ変換結果レジスタ

A/D 変換結果下位レジスタ SP

	7	6	5	4	3	2	1	0
ADREGSPL	ADRSP1	ADRSP0					OVRSP	ADRSPRF
Read/Write	R		R				R	R
リセット後	0		0				0	0
機能	A/D 変換結果下位 2 ビット格納		リードすると"0"が読めます				Over RUN flag 0: 発生無し 1: 発生	A/D 変換結果 格納フラグ 1: 変換結果有

A/D 変換結果上位レジスタ SP

	7	6	5	4	3	2	1	0
ADREGSPH	ADRSP9	ADRSP8	ADRSP7	ADRSP6	ADRSP5	ADRSP4	ADRSP3	ADRSP2
Read/Write	R							
リセット後	0							
機能	A/D 変換結果上位 8 ビット格納							



- ADREGSP のビット 0 は、A/D 変換結果格納フラグ <ADR_xRF> です。A/D 変換値が格納されると、“1” にセットされます。下位のレジスタ (ADREG_xL) をリードすると、“0” にされます。
- ADREGSP のビット 1 は over RUN flag<OVR_x> です。両方の変換結果格納レジスタ (ADREG_xH,ADREG_xL) を Read する前に変換結果が上書きされると 1 に Set されます。Flag の Read により 0 にクリアされます。
- 変換結果格納レジスタをバイト単位で読む場合は、上位を先に読み出し、次に下位を読み出してください。

図15-12 A/D コンバータ変換結果レジスタ

A/D 変換結果比較下位レジスタ 0

	7	6	5	4	3	2	1	0
ADCMP0L	bit Symbol ADR021 ADR020							
	Read/Write R/W					R		
	リセット後 0					0		
	機能 A/D 変換結果比較下位 2ビット格納		リードすると"0"が読めます					

A/D 変換結果比較上位レジスタ 0

	7	6	5	4	3	2	1	0
ADCMP0H	ADR029	ADR028	ADR027	ADR026	ADR025	ADR024	ADR023	ADR022
	Read/Write R/W							
	リセット後 0							
	機能 A/D 変換結果比較上位 8ビット格納							

A/D 変換結果比較下位レジスタ 1

	7	6	5	4	3	2	1	0
ADCMP1L	bit Symbol ADR121 ADR120							
	Read/Write R/W					R		
	リセット後 0					0		
	機能 A/D 変換結果比較下位 2ビット格納		リードすると"0"が読めます					

A/D 変換結果比較上位レジスタ 1

	7	6	5	4	3	2	1	0
ADCMP1H	ADR129	ADR128	ADR127	ADR126	ADR125	ADR124	ADR123	ADR122
	Read/Write R/W							
	リセット後 0							
	機能 A/D 変換結果比較上位 8ビット格納							

(注) このレジスタへ値を設定する時、または値を変更する時は、AD 監視機能を禁止 (ADMOD3, 5<ADBSVx>=" 0") した状態で行ってください

図15-13 A/D コンバータ変換結果比較レジスタ

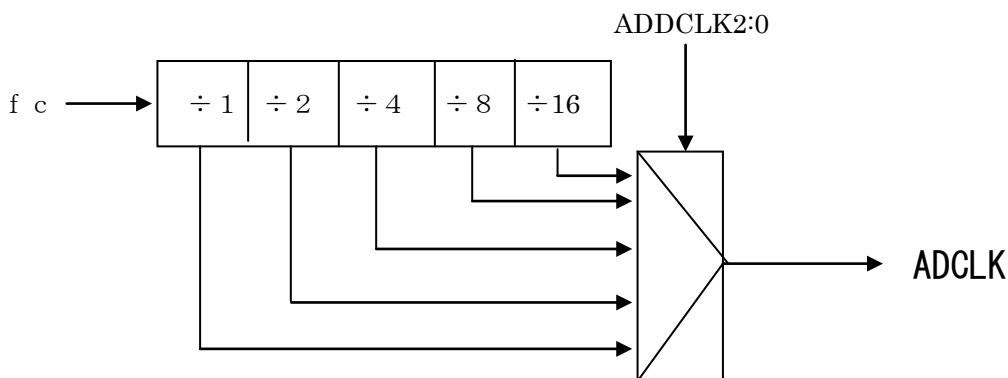
15.3 変換クロック

- 最短 46 変換クロックで変換されます。

A/D 変換クロック設定レジスタ

		7	6	5	4	3	2	1	0
ADCLK	bit Symbol	TSH3	TSH2	TSH1	TSH0	/	ADCLK2	ADCLK1	ADCLK0
	Read/Write	R/W	R/W	R/W	R/W		R	R/W	R/W
	リセット後	1	0	0	0	0	0	0	0
	機能	A/D サンプルホールド時間選択 1000: 8 変換クロック 1010: 24 変換クロック 0011: 64 変換クロック 1100: 128 変換クロック 上記以外の設定: reserved				1001:16 変換クロック 1011: 32 変換クロック 1101: 512 変換クロック	リードすると"0"が読めます	A/D プリスケアラ出力選択 000: fc 001: fc/2 010: fc/4 011: fc/8 100: fc/16 111:reserved	

図15-14 A/D 変換クロック設定レジスタ



(例) $f_{sys}=f_c=40\text{MHz}/32\text{MHz}$ の場合

Prescaler [ADDCLK2:0]	tconv. (変換時間) 40MHz	tconv. (変換時間) 32MHz
1	1.15 μs	1.44 μs
1/2	2.3 μs	2.88 μs
1/4	4.6 μs	5.75 μs

S/H 時間可変

変換 clock	S/H 時間	40MHz	tconv. (変換時間)
40MHz	変換 clk*8	(0.2 μs)	1.15 μs
	変換 clk*16	(0.4 μs)	1.35 μs
	変換 clk*24	(0.6 μs)	1.55 μs
	変換 clk*32	(0.8 μs)	1.75 μs
	変換 clk*64	(1.6 μs)	2.55 μs
	変換 clk*128	(3.2 μs)	4.15 μs
	変換 clk*512	(12.8 μs)	13.75 μs

(注) 「AD 変換中には、A/D 変換クロック設定を変更しないで下さい」

図15-15 A/D 変換時間

15.4 動作説明

15.4.1 アナログ基準電圧

アナログ基準電圧の“H”レベル側をVREFH端子に、“L”レベル側をVREFL端子に印加します。ADMOD1<VREFON>ビットに”0”を書き込むことにより、VREFH-VREFL間のスイッチをOFFできます。AD変換をスタートさせる場合は、かならず<VREFON>ビットに”1”を書き込んだ後、内部基準電圧が安定するまでの $3\mu s$ 待ってから、ADMOD0<ADS>ビットに”1”を書き込んでください。

15.4.2 アナログ入力チャネルの選択

アナログ入力チャネルの選択は、A/Dコンバータの動作モードによって異なります。

(1) 通常AD変換時

- アナログ入力チャネルを固定で使用する場合 (ADMOD0<SCAN>= “0”)
ADMOD1<ADCH[3:0]>の設定により、アナログ入力AIN0～AIN11端子の中から1チャネルを選択します。
- アナログ入力チャネルをスキャンで使用する場合 (ADMOD0<SCAN>= “1”)
ADMOD1 <ADCH[3:0]> の設定およびADSCNにより、12種類のスキャンモードの中から1つのスキャンモードを選択します。

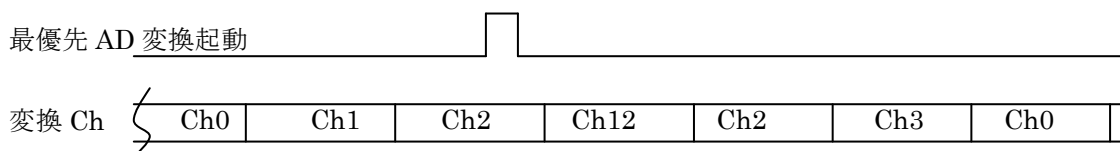
(2) 最優先AD変換時

ADMOD2<HPADCH[3:0]>の設定により、アナログ入力AIN0～AIN11端子の中から1チャネルを選択します。

リセット後はADMOD0<SCAN>は“0”にADMOD1<ADCH[3:0]>は“0000”に初期化され、これにより選択が行なわれますので、AIN0端子のチャネル固定入力を選択されます。なお、アナログ入力チャネルとして使用しない端子は、通常の入力ポートとして使用できます。

通常AD変換中に最優先AD変換の起動が掛かると、通常AD変換は中断し、最優先AD変換が実行されて終了後に通常AD変換を再開します。

例) ADMOD0<REPEAT : SCAN>= ”11”、ADMOD1<ADCH[3:0]>=0011 でチャネルAIN0～AIN3までのリピートスキャン変換中にADMOD2<HPADCH[3:0]>=1100でAIN12の最優先AD変換が起動された場合。



15.4.3 A/D 変換開始

A/D 変換には、通常 AD 変換と最優先 AD 変換の 2 種類があります。通常 AD 変換は ADMOD0<ADS> に “1” を設定することによりソフトで起動が掛かります。また、最優先 AD 変換は ADMOD2<HPADCE> に “1” を設定することによりソフトで起動が掛かります。通常 AD 変換は ADMOD0<[2:1]>で指定される 4 種類の動作モードから 1 つの動作モードが選択されます。最優先 AD 変換の動作モードはチャンネル固定のシングル変換のみです。また、通常 AD 変換は ADMOD4<ADHS>、最優先 AD 変換は ADMOD4<HADHS>で選択される HW 起動ソースにより起動を掛けることができます。このビットが ‘0’ の場合は、ADTRG 端子より立ち下がりエッジの入力により起動が掛かり、このビットが ‘1’ の場合、通常 AD 変換は 16 ビットタイマ 6 からの TB6RGO の一致で起動が掛かり、最優先 AD 変換の場合は 16 ビットタイマ 5 からの TB5RGO の一致で起動が掛かります。H/W 起動が許可された場合でもソフトウェア起動は有効です。

(注) 最優先 AD 変換の HW 起動ソースに外部トリガを使用しているときは、通常 AD 変換 HW 起動としては外部トリガを設定できません。

(注) 本製品には HW 起動ソースとしての外部トリガ入力はありませんので、<HADHS>、<ADHS>に “0” を設定することはできません。

通常 A/D 変換が開始されると、A/D 変換中を示す A/D 変換 Busy フラグ (ADMOD0<ADBF>) が “1” にセットされます。また、最優先 AD 変換が開始されると、AD 変換中を示す AD 変換 Busy フラグ (ADMOD2<ADBFHP>) が 1 にセットされます。このときに通常 AD 変換用の Busy フラグは、最優先 AD 変換の開始前の値を保持します。また、通常 AD 変換用の変換終了フラグ EOCFN も開始前の値を保持します。

(注) 最優先 A/D 変換中に通常 AD 変換を再起動させないでください (最優先 A/D 変換終了フラグがセットされません。また、以前の通常 A/D 変換のフラグがクリアされません)。

通常 A/D 変換実行中に通常 A/D 変換を再起動する場合はソフトウェアリセット (ADMOD3<ADRST[1:0]>) を行ってから起動してください。HW による通常 A/D 変換の再起動は行わないでください。

通常 AD 変換中に ADMOD2<HPADCE>に “1” を設定すると、現在変換中の AD 変換は中断され、最優先 AD 変換が始まり ADMOD2<[3:0]>で指定されるチャンネルの AD 変換 (チャンネル固定のシングル変換) が開始されます。この結果を格納レジスタ ADREGSP へ格納すると、続きから通常 AD 変換を再開します。

通常 AD 変換中に HW による最優先 AD 変換の起動が許可されている場合は、リソースからの起動条件が成立すると現在変換中の AD 変換は中断され、最優先 AD 変換が始まり ADMOD2<[3:0]>で指定されるチャンネルの AD 変換 (チャンネル固定のシングル変換) が開始されます。この結果を格納レジスタ ADREGSP へ格納すると、続きから通常 AD 変換を再開します。

15.4.4 A/D 変換モードと A/D 変換終了割り込み

A/D 変換には、次の 4 つの動作モードが用意されています。通常 AD 変換の場合は ADMOD0<2:1>の設定により選択ができます。最優先 AD 変換の場合は ADMOD0<2:1>の設定によらず、チャンネル固定のシングル変換のみの動作です。

- チャンネル固定シングル変換モード
- チャンネルスキャンシングル変換モード
- チャンネル固定リピート変換モード
- チャンネルスキャンリピート変換モード

(1) 通常 A/D 変換

動作モードの選択は、ADMOD0<REPEAT, SCAN>で行います。A/D 変換が開始されると ADMOD0<ADBFN>が“1”にセットされます。指定された AD 変換が終了すると、A/D 変換終了割り込み (INTAD) が発生し、A/D 変換終了を示す ADMOD0<EOCF>が“1”にセットされます。<ADBFN>は<REPEAT>=“0”の時は EOCF のセットと同時に“0”に戻りますが、<REPEAT>=“1”の時は“1”の状態を保持して変換を続けます。

① チャンネル固定シングル変換モード

ADMOD0 <REPEAT, SCAN> に“00”を設定すると、チャンネル固定シングル変換モードになります。

このモードでは、選択した 1 チャンネルの変換を 1 回だけ行います。変換が終了した後、ADMOD0<EOCF>が“1”にセット、ADMOD0<ADBFN>が“0”にされ、INTAD の割り込み要求が発生します。<EOCF>は読み出す事により 0 にクリアされます。

② チャンネルスキャンシングル変換モード

ADMOD0 <REPET, SCAN> に“01”を設定すると、チャンネルスキャンシングル変換モードになります。

このモードでは、選択したスキャンチャンネルの変換をそれぞれ 1 回だけ行います。スキャン変換が終了した後、ADMOD0<EOCF>が“1”にセット、ADMOD0<ADBFN>が“0”にされ、INTAD の割り込み要求が発生します。<EOCF>は読み出す事で 0 にクリアされます。

③ チャンネル固定リピート変換モード

ADMOD0<REPEAT, SCAN>に“10”を設定するとチャンネル固定リピート変換モードになります。

このモードでは、選択した 1 チャンネルの変換を繰り返し行います。変換が終了した後、ADMOD<EOCF>が“1”にセットされます。ADMOD0<ADBF>は“0”にされず“1”を保持します。INTADの割り込み要求発生タイミングはADMOD0<ITM[1:0]>の設定により選択できます。<EOCF>がSetされるタイミングも割り込みのタイミングに連動します。

<EOCF>は読み出す事により0にクリアされます。

<ITM[1:0]>を“00”に設定するとA/D変換が1回終了するごとに割り込み要求が発生します。この場合、変換結果は常に格納レジスタのADREG08に格納されます。格納時点でEOCFは1になります。

<ITM[1:0]>を“01”に設定するとA/D変換が4回終了するごとに割り込み要求が発生します。この場合、変換結果は格納レジスタのADREG08～ADREG3Bに順次格納されます。ADREG3Bに格納後<EOCF>は1にSETされ、再びADREG08から格納を始めます。<EOCF>は読み出す事により0にクリアされます。

<ITM[1:0]>を“10”に設定するとA/D変換が8回終了するごとに割り込み要求が発生します。この場合、変換結果は格納レジスタのADREG08～ADREG7Fに順次格納されます。ADREG7F格納後<EOCF>は1にsetされ、再びADREG08から格納を始めます。

<EOCF>は読み出す事によりクリアされます。

④ チャンネルスキャンリピート変換モード

ADMOD0<REPEAT, SCAN>に“11”を設定するとチャンネルスキャンリピート変換モードになります。

このモードでは、選択したスキャンチャンネルの変換を繰り返し行います。1回のスキャン変換が終了するごとにADMOD0<EOCF>が“1”にセットされ、INTAD割り込み要求が発生します。ADMOD0<ADBF>は“0”にされず“1”を保持します。<EOCF>は読み出す事により”0”にクリアされます。

リピート変換モード(③、④のモード)の動作を停止させたい場合は、ADMOD0<REPEAT>に“0”を書き込んでください。実行中の変換を終了した時点で、リピート変換モードは終了し、ADMOD0<ADBF>は“0”にされます。

IDLE、STOPモード等のスタンバイ状態へ移行する場合は、スタンバイ状態へ移行する前にA/D変換を停止していることを確認してから(または、停止させてから)移行してください。

(2) 最優先A/D変換

動作モードはチャンネル固定のシングル変換のみです。ADMOD0<REPEAT, SCAN>の設定は関係ありません。起動条件が成立すると、ADMOD2<HPADCH[3:0]>で指定されるチャンネルの変換を一度だけ行います。変換が終了すると、最優先AD変換終了割り込みが発生して、ADMOD2<EOCFHP>は1にセットされ、<ADBFHP>は0に戻ります。EOCFHPフラグは読み出すとクリアされます。

表15-1 A/D変換モードと割り込み発生タイミング、フラグ動作の関係

変換モード	割り込み発生 タイミング	EOCF セットタイミング (注)	ADBF (割り込み 発生後)	ADMOD0		
				ITM1:0	REPEAT	SCAN
チャンネル固定 シングル変換	変換終了後	変換終了後	0	—	0	0
チャンネル固定 リピート変換	1回変換毎	変換が1回終了後	1	00	1	0
	4回変換毎	変換が4回終了後	1	01		
	8回変換毎	変換が8回終了後	1	10		
チャンネルスキャン シングル変換	スキャン 変換終了後	スキャン変換終了後	0	—	0	1
チャンネルスキャン リピート変換	1回のスキャン 変換終了毎	1回のスキャン 変換終了後	1	—	1	1

(注) EOCF はリードするとクリアされます。

15.4.5 最優先変換モード

通常 AD 変換に割り込んで、最優先 AD 変換を行う事ができます。最優先 AD 変換は ADMOD2<HPADCE>に '1' を設定するソフトによる起動と、ADMOD3<[7:6]>の設定により HW リソースを用いた起動ができます。通常 AD 変換中に最優先 AD 変換が起動されると、現在変換中の AD 変換は中断され、ADMOD2<[3:0]>で指定されるチャンネルのシングル変換を行います。変換結果は ADREGSP へ格納され、最優先 AD 変換割り込みが発生します。その後通常 AD 変換が続きから再開されます。また、最優先 AD 変換中の最優先 AD 変換の起動は無視されます。

例えば チャンネル AN0~AN8 までのチャンネルリピート変換が起動されており、AN3 の変換中に<HPADCE>に "1" がセットされた場合は AN3 の変換が中断され、<HPADC[3:0]>で指定されたチャンネルの変換を行い、結果を ADREGSP へ格納後に AN3 からのチャンネルリピート変換を再開します。

15.4.6 AD 監視機能

ADMOD3,5<ADOBSVx>に 1 を設定すると AD 監視機能が有効になり、REGSx<[3:0]>で指定された変換結果格納レジスタの内容が比較レジスタの値より大または小 (ADOBIC で大か小は指定) になると AD 監視機能割り込みが発生します。また、この比較動作は該当変換結果格納レジスタへ結果が格納されるごとに行われ、条件が成立すると割り込みが発生します。また、AD 監視機能に割り当てている格納レジスタは通常ではソフトで読み出しは行われませんので、overrun フラグ<OVRx>は常にセットされていることになり、変換結果格納フラグ<ADRxF>もセットされている事になります。したがって、AD 監視機能を使用する場合は当該変換結果格納レジスタのフラグを使用しないでください。

15.4.7 A/D 変換結果の格納と読み出し

A/D 変換結果は、通常 AD 変換の A/D 変換結果上位/下位レジスタ (ADREG08H/L~ADRG7FH/L) に格納されます。

チャンネル固定リピート変換モードでは、A/D 変換結果は、ADREG08H/L から ADREG7FH/L へと順次格納されます。ただし、割り込み発生を<ITM[1:0]>で 1 回ごとに指定した場合は ADREG08H/L のみに格納され、<ITM[1:0]>で 4 回ごとに指定した場合は ADREG08H/L~ADREG3BH/L へと順次格納されます。

表 15-2にアナログ入力チャンネルと A/D 変換結果レジスタの対応を示します。

表 15-2 アナログ入力チャンネルと A/D 変換結果レジスタの対応

アナログ入力 チャンネル (ポート A)	A/D 変換結果レジスタ			
	右記以外の変 換モード	チャンネル固定リピート 変換モード (1 回)	チャンネル固定リピート 変換モード (4 回ごと)	チャンネル固定リピート 変換モード (8 回ごと)
AN0	ADREG08H/L	ADREG08H/L 固定	ADREG08H/L ←	ADREG08H/L ←
AN1	ADREG19H/L		↓	↓
AN2	ADREG2AH/L		↓	↓
AN3	ADREG3BH/L		ADREG3BH/L ←	↓
AN4	ADREG4CH/L			ADRE4CBH/L ←
AN5	ADREG5DH/L			
AN6	ADREG6EH/L			
AN7	ADREG7FH/L			
AN8	ADREG08H/L			
AN9	ADREG19H/L			
AN10	ADREG2AH/L			
AN11	ADREG3BH/L			
AN12	ADREG4CH/L			

15.4.8 データポーリング

割り込みを使用せずにデータポーリングで AD 変換結果を処理する場合は ADMOD0<EOCF>のポーリングをしてください。このフラグがセットされた場合は、所定の AD 変換結果格納レジスタに変換結果が格納されていますので、セットを確認後に AD 変換格納レジスタを読み出してください。この際に Over RUNを検出する為に、変換結果格納レジスタの上位を先に読み出し、次に下位を読み出してください。この結果、下位に存在する OVRn=" 0"、ADRnRF=" 1" であれば、正しい変換結果を得た事になります。

16 ウォッチドッグタイマ (WDT)

ウォッチドッグタイマ (WDT) は、ノイズなどの原因により CPU が誤動作 (暴走) を始めた場合これを検出し、正常な状態に戻すことを目的としています。暴走を検出するとノンマスクブル割り込みを発生し CPU に知らせ、外部へはウォッチドッグタイマアウト端子 (WDTOUT) より”0” を出力して周辺装置への暴走の検出を知らせます。

(注) 本製品にはウォッチドッグタイマアウト端子 (WDTOUT) はありません。

また、このウォッチドッグタイマアウトをリセット (チップ内部) へ接続することにより、強制的にリセット動作を行うことができます。

16.1 構成

図 16-1 にウォッチドッグタイマのブロック図を示します。

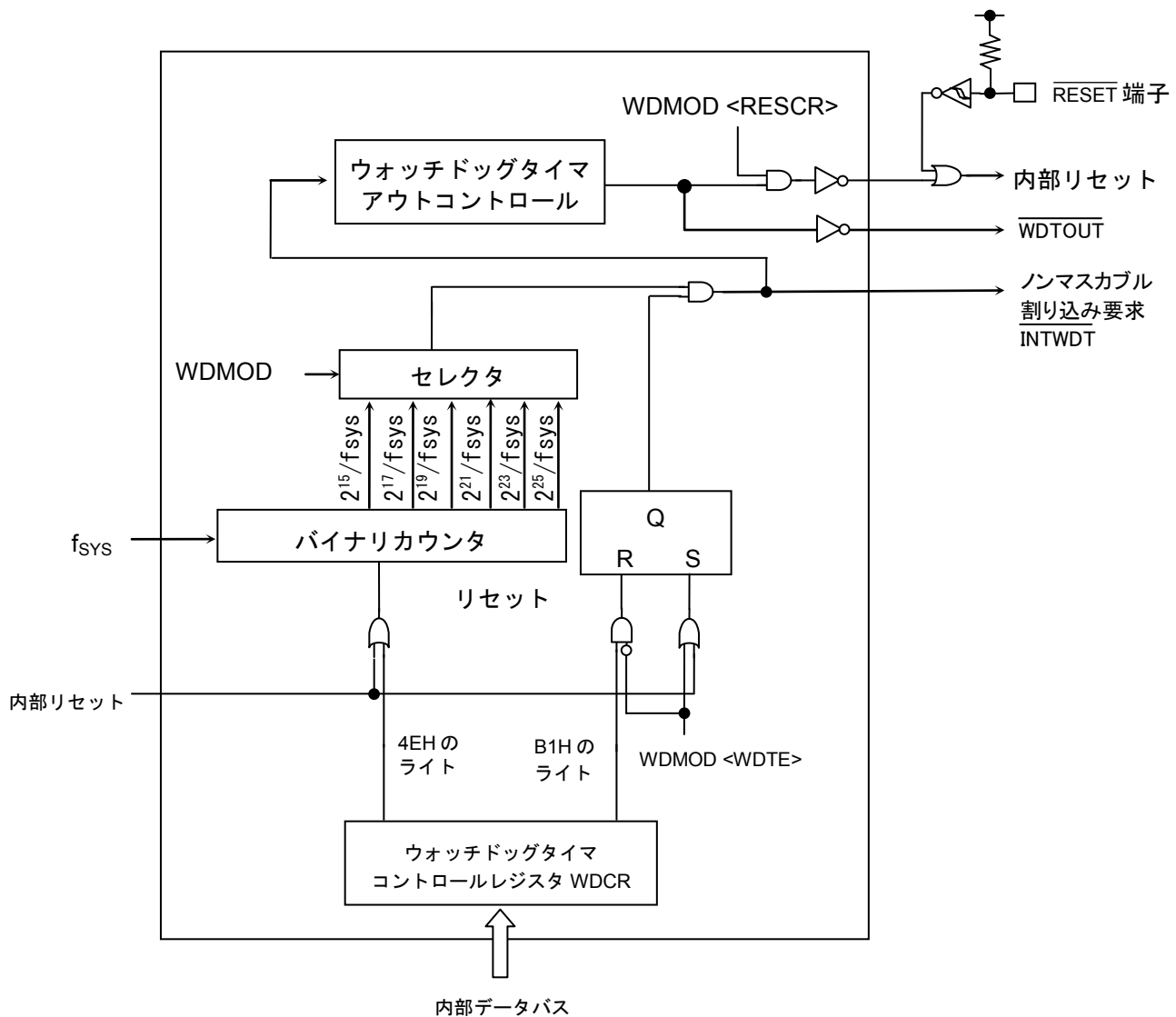


図16-1 ウォッチドッグタイマのブロック図

16.2 ウォッチドッグタイマ割り込み

ウォッチドッグタイマは、システムクロック f_{SYS} を入力クロックとする、バイナリカウンタで構成されています。バイナリカウンタの出力には 2^{15} 、 2^{17} 、 2^{19} 、 2^{21} 、 2^{23} および 2^{25} があります。このうちの 1 出力を WDMOD <WDTP[2:0]> で選択することにより、そのオーバーフロー時に、図 16-2 で示すように、ウォッチドッグタイマ割り込み INTWDT を発生し、ウォッチドッグタイマアウト (WDTOUT) を出力します。

INTWDT はノンマスクابل割り込みの要因となりますが、ノンマスクابل割り込みの要因は複数あるため、クロックジェネレータの NMI フラグレジスタ NMIFLG <NMIFLG0> ビットにて識別する必要があります。

また、ウォッチドッグタイマアウト端子 (WDTOUT) は、ウォッチドッグタイマのオーバーフローにより "0" を出力するため、周辺装置のリセットを行うことも可能です。この "0" 出力は、ウォッチドッグタイマのクリア (WDCR レジスタにクリアコード 4EH をライト) により、"1" にセットされます。すなわち、通常モードの場合、クリアコードが WDCR レジスタに書かれるまで、WDTOUT 端子は "0" を出力し続けます。

(注) 本製品にはウォッチドッグタイマアウト端子 (WDTOUT) はありません。

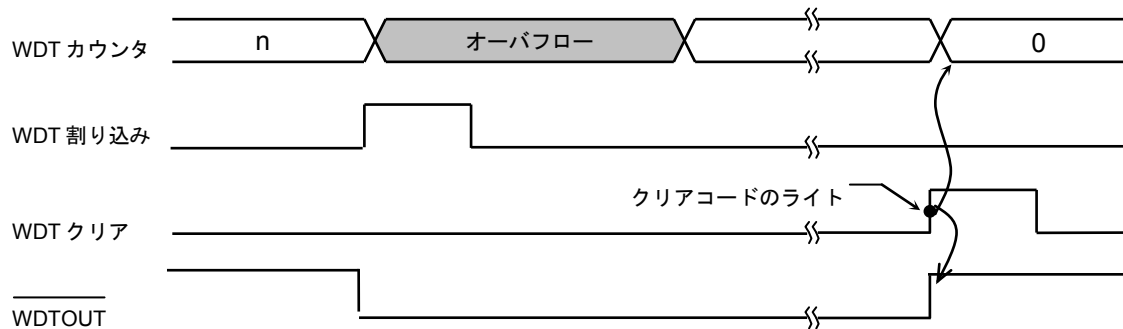


図16-2 通常モード

また、オーバーフロー時にチップ自身をリセットすることも選択可能です。この場合、図 16-3 で示すように 32 ステートの期間、リセットを行います。なお、この場合 (リセットされた場合)、入力クロック f_{SYS} は、高速発振器のクロック f_C をクロックギアで 1 分周したクロック f_{SYS} が使われます。

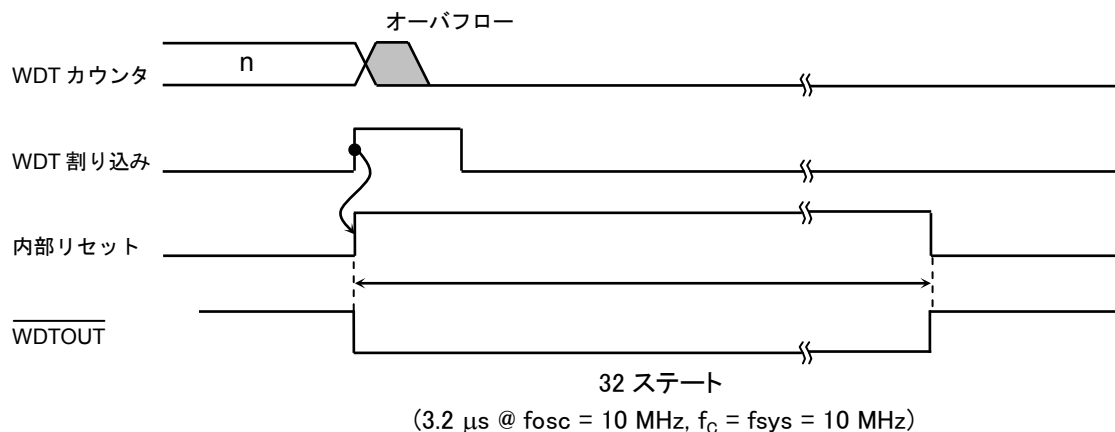


図16-3 リセットモード

16.3 コントロールレジスタ

ウォッチドッグ タイマ (WDT) は、2 つのコントロールレジスタ (WDMOD、WDCR) によって制御されています。

16.3.1 ウォッチドッグ タイマ モードレジスタ (WDMOD)

① ウォッチドッグ タイマ検出時間の設定 <WDTP[1:0]>

暴走検出のためのウォッチドッグ タイマ割り込み時間を設定する2ビットのレジスタです。リセット時 WDMOD <WDTP[2:0]> = “000” にイニシャライズされます。図 16-4 にウォッチドッグ タイマの検出時間を示します。

② ウォッチドッグ タイマのイネーブル/ディセーブル制御 <WDTE>

リセット時 WDMOD <WDTE> = “1” にイニシャライズされますので、ウォッチドッグ タイマはイネーブルになっています。

ディセーブルにするには、このビットを “0” にするとともに WDCR レジスタにディセーブル コード (B1H) を書き込む必要があります。この二重設定のため、暴走によるウォッチドッグ タイマのディセーブルが発生し難くなっています。

逆にディセーブル状態からイネーブル状態に戻す場合は、<WDTE> ビットを “1” にセットするだけでイネーブルとなります。

③ ウォッチドッグタイマアウトのリセット接続 <RESCR>

暴走検出により自分自身をリセットするかどうかを設定するレジスタです。リセット時 WDMOD <RESCR> = “1” に初期化されますので、カウンタのオーバフローによりリセットが発生します。

16.3.2 ウォッチドッグ タイマ コントロールレジスタ (WDCR)

ウォッチドッグ タイマ機能のディセーブルおよびバイナリカウンタのクリアを制御するレジスタです。

● ディセーブル制御

WDMOD <WDTE> を “0” にしたあと、この WDCR レジスタにディセーブル コード (B1H) を書き込むとウォッチドッグ タイマをディセーブルにすることができます。

WDMOD	← 0 - - - - -	WDTE を “0” クリアします。
WDCR	← 1 0 1 1 0 0 0 1	ディセーブルコード (B1H) を書き込みます。

● イネーブル制御

WDMOD <WDTE> を “1” にする。

● ウォッチドッグタイマのクリア制御

WDCR レジスタにクリア コード (4EH) を書き込むと、バイナリカウンタはクリアされ、再カウントします。

WDCR ← 0 1 0 0 1 1 1 0 クリア コード (4EH) を書き込みます。

(注) ディセーブルコード (B1H) を書き込むとバイナリカウンタはクリアされます。

ウォッチドッグタイマモードレジスタ

	7	6	5	4	3	2	1	0
bit Symbol	WDTE	WDTP2	WDTP1	WDTP0		I2WDT	RESOR	
Read/Write	R/W	R/W			R	R/W		R/W
リセット後	1	0	0	0		0	1	0
機能	WDT 制御 1: 許可	WDT 検出時間の選択 000: $2^{15}/f_{SYS}$ 001: $2^{17}/f_{SYS}$ 010: $2^{19}/f_{SYS}$ 011: $2^{21}/f_{SYS}$ 100: $2^{23}/f_{SYS}$ 101: $2^{25}/f_{SYS}$ 110: 設定禁止 111: 設定禁止			リードすると“0”が読めます。	IDLE 0: 停止 1: 動作	0: NMI 割込み発生 1: WDTOUT をリセットへ接続	“0”をライトしてください。

ウォッチドッグタイマアウトコントロール

0	NMI 割り込みが発生します
1	WDTOUT をリセットへ接続

ウォッチドッグタイマの検出時間

@ $f_c = 40 \text{ MHz}$

SYSCR1 クロック分値 <GEAR[2:0]>	Watch Dog Timer の検出時間					
	WDMOD<WDTP[2:0]>					
	000	001	010	011	100	101
000 (f_c)	0.82 ms	3.28 ms	13.11 ms	52.43 ms	209.72 ms	838.86 ms
100 ($f_c/2$)	1.63 ms	6.55 ms	26.21 ms	104.86 ms	419.43 ms	1.68 s
101 ($f_c/4$)	3.28 ms	13.11 ms	52.43 ms	209.72 ms	838.86 ms	3.36 s
110 ($f_c/8$)	6.55 ms	26.21 ms	104.86 ms	419.43 ms	1.68 s	6.71 s

ウォッチドッグタイマの禁止/許可制御

0	停止
1	許可

ウォッチドッグタイマコントロールレジスタ

	7	6	5	4	3	2	1	0
bit Symbol	—							
Read/Write	W							
リセット後	—							
機能	B1H : WDT ディセーブルコード 4EH : WDT クリアコード							

WDT のディセーブル&クリア

B1H	ディセーブルコード
4EH	クリアコード
上記以外	—

図16-4 ウォッチドッグタイマ関連レジスタ

16.4 動作説明

ウォッチドッグタイマは、WDMOD <WDTP[2:0]> レジスタで設定された検出時間後に割り込み (INTWDT) を発生させ、ウォッチドッグタイマアウト端子 (WDTOUT) より“L”レベルを出力させるタイマです。ソフトウェア (命令) でウォッチドッグタイマ用のバイナリカウンタを $\overline{\text{INTWDT}}$ 割り込みが発生する前にゼロクリアする必要があります。もし、CPU がノイズなどの原因で誤動作 (暴走) しバイナリカウンタをクリアする命令を実行しなければ、バイナリカウンタはオーバフローし、INTWDT によってノンマスカブル割り込みが発生します。CPU はノンマスカブル割り込みにより誤動作 (暴走) が発生したことを知り、誤動作 (暴走) 対策プログラムにより正常な状態に戻すことができます。また、ウォッチドッグタイマアウト端子を周辺装置のリセットなどへ接続することにより、CPU の誤動作 (暴走) に対処することができます。

(注) 本製品にはウォッチドッグタイマアウト端子 (WDTOUT) はありません。

ウォッチドッグタイマは、リセット解除後ただちに動作を開始します。

また、STOP モード中のウォッチドッグタイマはリセットされ停止しています。バス解放中 ($\overline{\text{BUSAK}} = \text{“L”}$) は、カウントを続けます。IDLE モードでは、WDMOD <I2WDT> の設定に依存します。必要に応じて、IDLE モードに入る前に WDMOD <I2WDT> を設定してください。

(注) 本製品には外部バス機能はありませんので、バス解放状態になりません。

例: ① バイナリカウンタをクリアします。

```

          7 6 5 4 3 2 1 0
WDCR    ← 0 1 0 0 1 1 1 0   クリアコード (4EH) の書き込み

```

② ウォッチドッグタイマ検出時間を $2^{21}/f_{\text{SYS}}$ に設定します。

```

          7 6 5 4 3 2 1 0
WDMOD   ← 1 0 1 1 - - - -

```

③ ウォッチドッグタイマをディセーブルします。

```

          7 6 5 4 3 2 1 0
WDMOD   ← 0 - - - - - -   WDTTE を “0” クリア
WDCR    ← 1 0 1 1 0 0 0 1   ディセーブルコード (B1H) の書き込み

```

(注1) 高周波発振子が止まっている状態では使用しないで下さい。

ウォッチドッグタイマにより、システムリセットが動作しても、高周波発振子の発振が安定していない為に、正常に動作いたしません。

(注2) デバックモード中は、ウォッチドッグタイマのカウンタが停止いたします。

17 リアルタイムクロック (RTC)

17.1 RTCの機能概略

- 1) 時計機能 (時間、分、秒)
- 2) カレンダー機能 (月日、週、うるう年)
- 3) 24 時間計と 12 時間計 (AM/PM) のいずれかを選択可能
- 4) +/-30 秒補正機能 (ソフトウェアによる補正)
- 5) アラーム機能 (アラーム出力)
- 6) アラーム割り込み発生

17.2 ブロック図

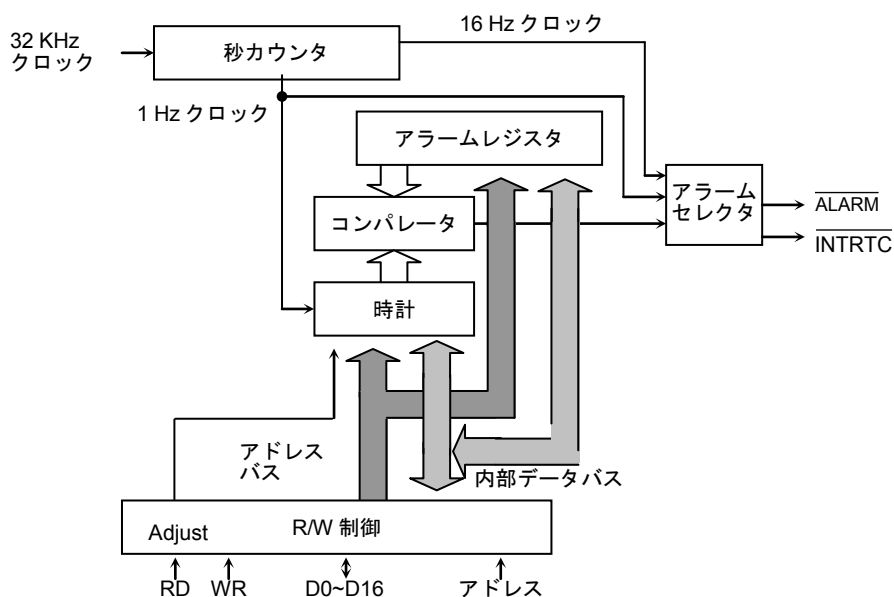


図 17-1 RTC ブロック図

(注 1) 西暦年桁について

本製品は、年桁を下 2 桁しか持っていません。そのため 99 年の翌年は 00 年として動作します。使用するシステムにおいて、西暦で年桁を取り扱う場合にはシステム側にて上 2 桁を管理してください。

(注 2) うるう年について

うるう年は、4 で割り切れる年ですが例外があり 100 で割り切れる年はうるう年ではありません。ただし、400 で割り切れる年はうるう年です。しかし、本製品は上記例外に対応していません。4 で割り切れる年のみをうるう年としていますのでこの点が問題であればシステム側にてあらかじめ対策してください。

17.3 コントロールレジスタ

- リセット動作により下記レジスタが初期化されます。PAGER<PAGE>, <ADJUST>, <INTENA>
- RESTR<RSTALM>, <RSTTMR>, <DIS16HZ>, <DIS1HZ>

これ以外の時計機能に関するレジスタはリセット動作が行なわれても初期化されません。
RTCを使用するには各レジスタに時刻/月日曜日年うるう年を設定後、動作を開始します。
時計データの設定、秒補正、時計リセットを行う場合注意が必要です。後述の“17.5.3 低消費電力モードへ遷移する場合”を参照してください。

表 17-1 PAGE0 (時計機能) レジスタ

Symbol	アドレス	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	設定内容	Read/Write
SECR	0x4004_0100H		40 秒	20 秒	10 秒	8 秒	4 秒	2 秒	1 秒	秒桁	R/W
MINR	0x4004_0101H		40 分	20 分	10 分	8 分	4 分	2 分	1 分	分桁	R/W
HOURL	0x4004_0102H			20 時 /PM/AM	10 時	8 時	4 時	2 時	1 時	時間桁	R/W
DAYR	0x4004_0104H						W2	W1	W0	曜日桁	R/W
DATER	0x4004_0105H			20 日	10 日	8 日	4 日	2 日	1 日	日桁	R/W
MONTHR	0x4004_0106H				10 月	8 月	4 月	2 月	1 月	月桁	R/W
YEARR	0x4004_0107H	80 年	40 年	20 年	10 年	8 年	4 年	2 年	1 年	年桁 (西暦下 2 桁)	R/W
PAGER	0x4004_0108H	割り込み許可			秒補正 設定	時計 許可	アラーム 許可		PAGE 設定	PAGE レジスタ	W, R/W
RESTR	0x4004_010CH	1Hz 許可	16Hz 許可	時計 リセット	アラーム リセット	“0” をライトしてください				リセットレジスタ	Wのみ

(注) PAGE0 の SECR, MINR, HOURL, DAYR, MONTHR, YEARR はリードすると現在の状態がリードされます。

表 17-2 PAGE1 (アラーム機能) レジスタ

Symbol	アドレス	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	設定内容	Read/Write
SECR	0x4004_0100H										
MINR	0x4004_0101H		40 分	20 分	10 分	8 分	4 分	2 分	1 分	アラーム分桁	R/W
HOURL	0x4004_0102H			20 時 /PM/AM	10 時	8 時	4 時	2 時	1 時	アラーム時間桁	R/W
DAYR	0x4004_0104H						W2	W1	W0	アラーム週桁	R/W
DATER	0x4004_0105H			20 日	10 日	8 日	4 日	2 日	1 日	アラーム日桁	R/W
MONTHR	0x4004_0106H								24/12	24 時間クロック モード	R/W
YEARR	0x4004_0107H								うるう年設定	うるう年モード	R/W
PAGER	0x4004_0108H	割り込み許可			秒補正 設定	時計 許可	アラーム 許可		PAGE 設定	PAGE レジスタ	W, R/W
RESTR	0x4004_010CH	1Hz 許可	16Hz 許可	時計 リセット	アラーム リセット	“0” をライトしてください				リセットレジスタ	Wのみ

(注1) PAGE1 の MINR, HOURL, DAYR, MONTHR, YEARR はリードすると現在の状態がリードされます。

(注2) PAGE0 の SEC, MIN, HOUR, DAY, DATE, MONTH, YEAR レジスタ、および PAGE1 の YEAR (うるう年) レジスタのリード動作は 2 回行い、比較処理を行ってください。

17.4 コントロールレジスタの説明

(1) 秒桁レジスタの設定 (PAGE0 のみ)

	7	6	5	4	3	2	1	0
Bit symbol	—	SE6	SE5	SE4	SE3	SE2	SE1	SE0
Read/Write	R	R/W						
リセット後	0	不定						
機能	“0” がリロードされます。	40 秒桁	20 秒桁	10 秒桁	8 秒桁	4 秒桁	2 秒桁	1 秒桁

設定例を下記に示します。

0	0	0	0	0	0	0	0 秒
0	0	0	0	0	0	1	1 秒
0	0	0	0	0	0	1	2 秒
0	0	0	0	0	0	1	3 秒
0	0	0	0	1	0	0	4 秒
0	0	0	0	1	0	1	5 秒
0	0	0	0	1	1	0	6 秒
0	0	0	0	1	1	1	7 秒
0	0	0	1	0	0	0	8 秒
0	0	0	1	0	0	1	9 秒
0	0	1	0	0	0	0	10 秒
:							
0	0	1	1	0	0	1	19 秒
0	1	0	0	0	0	0	20 秒
:							
0	1	0	1	0	0	1	29 秒
0	1	1	0	0	0	0	30 秒
:							
0	1	1	1	0	0	1	39 秒
1	0	0	0	0	0	0	40 秒
:							
1	0	0	1	0	0	1	49 秒
1	0	1	0	0	0	0	50 秒
:							
1	0	1	1	0	0	1	59 秒

注) 上記以外の設定はしないでください。

(2) 分析レジスタの設定 (PAGE0/1)

	7	6	5	4	3	2	1	0
MINR	—	MI6	MI5	MI4	MI3	MI2	MI1	MI0
Bit symbol	—	R/W						
Read/Write	R	R/W						
リセット後	0	不定						
機能	"0" がリードされます。	40分	20分	10分	8分	4分	2分	1分

設定例を下記に示します。

0	0	0	0	0	0	0	0分
0	0	0	0	0	0	1	1分
0	0	0	0	0	1	0	2分
0	0	0	0	0	1	1	3分
0	0	0	0	1	0	0	4分
0	0	0	0	1	0	1	5分
0	0	0	0	1	1	0	6分
0	0	0	0	1	1	1	7分
0	0	0	1	0	0	0	8分
0	0	0	1	0	0	1	9分
0	0	1	0	0	0	0	10分
:							
0	0	1	1	0	0	1	19分
0	1	0	0	0	0	0	20分
:							
0	1	0	1	0	0	1	29分
0	1	1	0	0	0	0	30分
:							
0	1	1	1	0	0	1	39分
1	0	0	0	0	0	0	40分
:							
1	0	0	1	0	0	1	49分
1	0	1	0	0	0	0	50分
:							
1	0	1	1	0	0	1	59分

注) 上記以外の設定はしないでください。

(3) 時間桁レジスタの設定 (PAGE0/1)

1. 24時間クロックモード (MONTHR<MO0> = “1”) の場合

HOURL		7	6	5	4	3	2	1	0
	Bit symbol	—		H05	H04	H03	H02	H01	H00
	Read/Write	R		R/W					
	リセット後	0		不定					
	機能	“0” がリードされます。		20時	10時	8時	4時	2時	1時

設定例を下記に示します。

0	0	0	0	0	0	0	0時
0	0	0	0	0	0	1	1時
0	0	0	0	0	1	0	2時
:							
0	0	1	0	0	0	0	8時
0	0	1	0	0	0	1	9時
0	1	0	0	0	0	0	10時
:							
0	1	1	0	0	0	1	19時
1	0	0	0	0	0	0	20時
:							
1	0	0	0	1	1	1	23時

注) 上記以外の設定はしないでください。

2. 12時間クロックモード (MONTHR<MO0> = “0”) の場合

HOURL		7	6	5	4	3	2	1	0
	Bit symbol	—		H05	H04	H03	H02	H01	H00
	Read/Write	R		R/W					
	リセット後	0		不定					
	機能	“0” がリードされます。		PM/AM	10時	8時	4時	2時	1時

設定例を下記に示します。

0	0	0	0	0	0	0	0時 (AM)
0	0	0	0	0	0	1	1時
0	0	0	0	0	1	0	2時
:							
0	0	1	0	0	0	1	9時
0	1	0	0	0	0	0	10時
0	1	0	0	0	0	1	11時
1	0	0	0	0	0	0	0時 (PM)
1	0	0	0	0	0	1	1時

注) 上記以外の設定はしないでください。

(4) 週桁レジスタの設定 (PAGE0/1)

	7	6	5	4	3	2	1	0
DAYR	—					WE2	WE1	WE0
Bit symbol	—					R/W		
Read/Write	R					R/W		
リセット後	0					不定		
機能	“0” がリードされます。					W2	W1	W0

設定例を下記に示します。

0	0	0	日曜日
0	0	1	月曜日
0	1	0	火曜日
0	1	1	水曜日
1	0	0	木曜日
1	0	1	金曜日
1	1	0	土曜日

注) 上記以外の設定はしないでください。

(5) 日桁レジスタ (PAGE0/1)

	7	6	5	4	3	2	1	0
DATER	—		DA5	DA4	DA3	DA2	DA1	DA0
Bit symbol	—		DA5	DA4	DA3	DA2	DA1	DA0
Read/Write	R		R/W					
リセット後	0		不定					
機能	“0” がリードされます。		20日	10日	8日	4日	2日	1日

設定例を下記に示します。

0	0	0	0	0	0	0	0日
0	0	0	0	0	0	1	1日
0	0	0	0	0	1	0	2日
0	0	0	0	0	1	1	3日
0	0	0	1	0	0	0	4日
:							
0	0	1	0	0	1	1	9日
0	1	0	0	0	0	0	10日
0	1	0	0	0	1	0	11日
:							
0	1	1	0	0	1	1	19日
1	0	0	0	0	0	0	20日
:							
1	0	1	0	0	1	1	29日
1	1	0	0	0	0	0	30日
1	1	0	0	0	1	1	31日

注1) 上記以外の設定はしないでください。

注2) 2月30日など、存在しない日は設定しないでください。

(6) 月桁レジスタの設定 (PAGE0 のみ)

	7	6	5	4	3	2	1	0
MONTHR	—			M04	M04	M02	M01	M00
Bit symbol	—			M04	M04	M02	M01	M00
Read/Write	R			R/W				
リセット後	0			不定				
機能	“0” がリードされます。			10月	8月	4月	2月	1月

設定例を下記に示します。

0	0	0	0	1	1月
0	0	0	1	0	2月
0	0	0	1	1	3月
0	0	1	0	0	4月
0	0	1	0	1	5月
0	0	1	1	0	6月
0	0	1	1	1	7月
0	1	0	0	0	8月
0	1	0	0	1	9月
1	0	0	0	0	10月
1	0	0	0	1	11月
1	0	0	1	0	12月

注) 上記以外の設定はしないでください。

(7) 24 時間時計、12 時間時計の選択 (PAGE1 のみ)

	7	6	5	4	3	2	1	0
MONTHR	—							M00
Bit symbol	—							M00
Read/Write	R							R/W
リセット後	0							不定
機能	“0” がリードされます。							1: 24 時間 0: 12 時間

(注) RTC 動作時 (PAGER<ENATMR>=“1”)には、MONTHR<M00>ビットを操作しないでください。

(8) 年桁レジスタの設定 (PAGE0 のみ)

	7	6	5	4	3	2	1	0	
YEARR	Bit symbol	YE7	YE6	YE5	YE4	YE3	YE2	YE1	YE0
	Read/Write	R/W							
	リセット後	不定							
	機能	80年	40年	20年	10年	8年	4年	2年	1年

設定例を下記に示します。

0	0	0	0	0	0	0	0	00年
0	0	0	0	0	0	0	1	01年
0	0	0	0	0	0	1	0	02年
0	0	0	0	0	0	1	1	03年
0	0	0	0	0	1	0	0	04年
0	0	0	0	0	1	0	1	05年
:								
1	0	0	1	1	0	0	1	99年

注) 上記以外の設定はしないでください。

(9) うるう年レジスタの設定 (PAGE1 のみ)

	7	6	5	4	3	2	1	0
YEARR	Bit symbol	—					LEAP1	LEAP0
	Read/Write	R					R/W	
	リセット後	0					不定	
	機能	“0” がリードされます。					00: うるう年 01: うるう年から1年目 10: うるう年から2年目 11: うるう年から3年目	

設定例を下記に示します。

0	0	現在の年 (今年) がうるう年
0	1	現在がうるう年から1年目
1	0	現在がうるう年から2年目
1	1	現在がうるう年から3年目

(10) PAGE レジスタの設定 (PAGE0/1)

	7	6	5	4	3	2	1	0
PAGER	INTENA	—		ADJUST	ENATMR	ENAALM	—	PAGE
	R/W	R		R/W	R/W		R	R/W
	リセット後	0		0	不定		0	0
リードモディファイライト できません	機能 INTRTC 0: 禁止 1: 許可	"0" がリードされま す。		[ライト] 0: Don' t care 1: ADJUST 要 求セット [リード] 0: ADJUST 要 求なし 1: ADJUST 要 求あり	時計 0: 禁止 1: 許可	ALARM 0: 禁止 1: 許可	"0" がリー ドされます。	PAGE 設定

(注 1) このレジスタはリードモディファイライトできません。
 (注 2) <ENATMR>および<ENAAML>の各々の割り込み許可ビットと、INTENA の割り込み許可ビットは
 下記の設定順番を守り、同時に設定しないようにしてください。
 (時計/アラーム許可と割り込み許可の設定間に時間差を設ける。)
 また、<ENATMR>、<ENAAML>の設定を変更する際には、<ENAAML>を禁止してから行って
 ください。

(例) 現時刻、アラーム設定

```

        7 6 5 4 3 2 1 0
PAGER   ← 0 0 0 0 1 1 0 0   時計、アラーム許可
PAGER   ← 1 0 0 0 1 1 0 0   割り込み許可
    
```

PAGE	0	Page0 が選択されます
	1	Page1 が選択されます

ADJUST	0	Don' t care
	1	秒を補正します。要求は秒カウンタのカウン アップ時にサンプリングされ、秒が 0~29 秒の 場合秒桁のみ "0" になります。また、30~59 秒のときは分を桁上げて秒を" 0" にします。 このビットを読み出すことで、要求のあり/な しを確認できます。

(11) リセットレジスタの設定 (PAGE0/1)

		7	6	5	4	3	2	1	0
RESTR	Bit symbol	DIS1HZ	DIS16HZ	RSTTMR	RSTALM	—			
	Read/Write	R/W				R			
	リセット後	1	1	0	0	0			
リードモディ ファイライト できません	機能	1 Hz 0: 許可 1: 禁止	16 Hz 0: 許可 1: 禁止	[ライト] 0: Don't care 1: 時計 リセット [リード] 0: リセット 要求なし 1: リセット 要求有り	0: Don't care 1: アラーム リセット	リードすると“0”が読めます。			

(注) このレジスタはリードモディファイライトできません。

RSTALM	0	未使用
	1	アラームレジスタ(分、時、日、週桁レジスタ)を初期化します。 初期化後は、00分、00時、01日、日曜日になります。

RSTTMR	0	未使用
	1	秒カウンタをリセットします。要求は低速クロックでサンプリングされます。 このビットを読み出すことで、要求のあり/なしを確認できます。

アラーム、1Hz 割り込み、16Hz 割り込みで使用する場合の<DIS1HZ>、<DIS16HZ>、RTCPAGER <ENAALM>の設定を以下に示します。

<DIS1HZ>	<DIS16HZ>	PAGER<ENAALM>	割り込みソース信号
1	1	1	アラーム
0	1	0	1Hz
1	0	0	16Hz
その他			“0”が出力されます。

17.5 動作説明

RTC 内部には 32.768 KHz の信号から 1 Hz の信号を発生する秒カウンタが内蔵されており、このカウンタの動作を考慮して使用する必要があります。

17.5.1 時計データをリードする場合

1. 1Hz 割り込みを利用する

秒カウンタのカウントアップに同期して 1Hz の割り込みが発生しますので、1Hz 割り込みを待って、データをリードすれば、正常にデータリードできます。

2. 2 度読みを利用する

時計データのリード動作の途中で内部カウンタの桁上げが起こると誤ったデータをリードする場合があります。従って、データを正しく読み込むために、下記の方法で 2 度以上リードしてください。

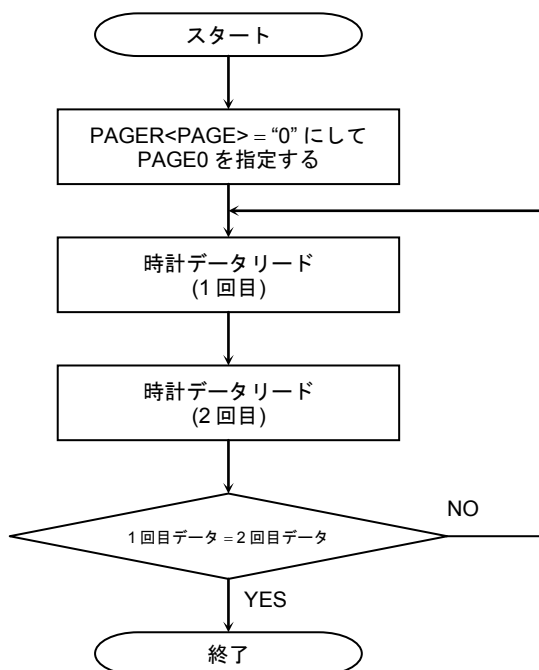


図 17-2 時計データのリードフロー

17.5.2 時計データをライトする場合

一連のデータライト動作の途中で桁上げ信号が入ってくると、期待するデータはライトできません。従って、データを正しくライトするためにはつぎの方法があります。

1. 1Hz 割り込みを利用する

秒カウンタのカウントアップに同期して 1Hz の割り込みとが発生しますので、1Hz 割り込みを待って、次の 1s カウントアップまでの間にデータをライトすれば、正常にデータライトできます。

2. カウンタをリセットする

秒カウンタをリセットした後にデータをライトします。

カウンタリセット後に 1Hz 割り込みを許可した場合、1 秒後に 1Hz 割り込みが発生しますので、1 秒以内に時間設定を行うようにしてください。

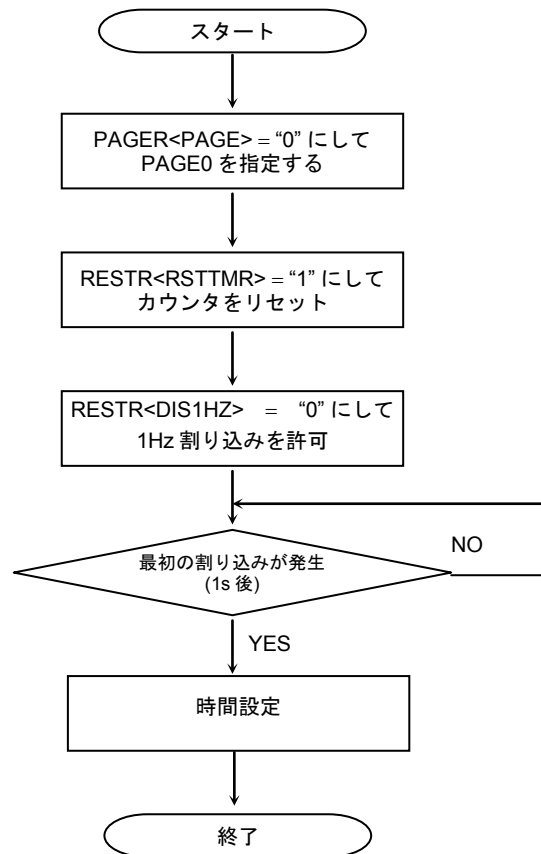


図 17-3 データライトのフロー

3. 時計を禁止する

PAGER<ENATMR>に “0” をライトすると、時計は禁止となって桁上げは禁止されま
す。

1Hz 割り込み発生後に時計を停止し（このとき秒カウンタは動作を継続）、次の 1Hz
割り込みが発生する前（1s 以内）に再度時計データを設定し時計機能を許可してくだ
さい。

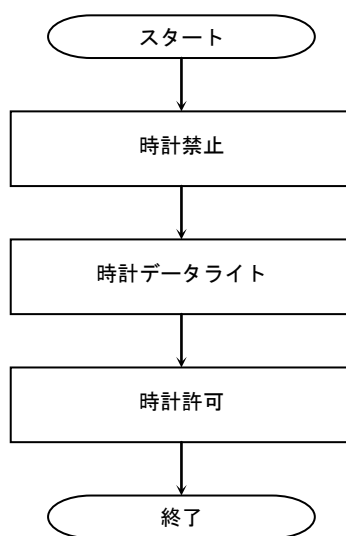


図 17-4 時計を禁止するフローチャート

17.5.3 低消費電力モードへ遷移する場合

時計データの設定、秒補正、時計リセット後にシステムクロックが停止するモード (SLEEP
モード) へ遷移する場合、必ず以下のどちらかの手順で行ってください。

1. レジスタ変更または ADJUST, RSTTMR 操作後に 1 秒割り込みの発生を待つ。
2. レジスタ変更または ADJUST, RSTTMR 操作後、時計レジスタ値/ADJUST フラグ/RSTTMR フ
ラグの値を Read し、反映を確認する。

17.6 アラーム機能の説明

PAGER<PAGE>に“1”をライトすることにより、PAGE1のレジスタ群でアラーム機能が使用できます。 $\overline{\text{ALARM}}$ 端子からは以下3つの信号のいずれかを出力できます。

- (1) アラームレジスタと時計の一致時、“0”パルスを出力
- (2) 1 Hz 周期の“0”パルスを出力
- (3) 16 Hz 周期の“0”パルスを出力

いずれの場合も、低速クロック 1 周期分のパルスを出力します。また、同時に INTRTC 割り込みの要求を出力します。INTRTC 割り込み信号は、立下りエッジが有効ですので、CG 割り込みモードコントロールレジスタのアクティブ状態の設定は、“立下りエッジ”に設定してください。

- (1) アラームレジスタと時計の一致時、 $\overline{\text{ALARM}}$ 端子からパルスを出力

PAGE1 のアラームレジスタと PAGE0 の時計の内容が一致したときに $\overline{\text{ALARM}}$ 端子に“0”パルスを出力するとともに、INTRTC 割り込みを発生し、その時刻になったことを知らせます。

アラームを使用する方法を下記に説明します。

まずアラーム禁止状態で、初期化を行います。アラームの初期化は RESTR<RSTALM>に“1”をライトすることにより行われ、アラーム分析、アラーム時桁、アラーム日桁、アラーム曜日桁は 00 分、00 時、01 日、日曜日になります。

PAGE1 のアラーム分析、アラーム時桁、アラーム日桁、アラーム曜日桁の設定を行い、PAGER<ENAALM>でアラームを許可します。

最後に PAGER<INTENA>で割り込みを許可します。

例えば、月曜日 5 日正午 (PM12:00) にアラームを出力させる場合のプログラムを下記に示します。

	7	6	5	4	3	2	1	0		
PAGER	←	0	0	0	0	1	0	0	1	アラーム禁止、PAGE1 設定
RESTR	←	1	1	0	1	0	0	0	0	アラーム初期化
DAYR	←	0	0	0	0	0	0	0	1	月曜日
DATAR	←	0	0	0	0	0	1	0	1	5 日
HOURR	←	0	0	0	1	0	0	1	0	12 時設定
MINR	←	0	0	0	0	0	0	0	0	00 分設定
PAGER	←	0	0	0	0	1	1	0	0	アラーム許可
PAGER	←	1	0	0	0	1	1	0	0	割り込み許可

アラーム設定は、低速クロックに同期して動作していますので、CPU が高周波で動作している場合、レジスタ設定してから有効になるまでに最大 32 kHz の 1 クロック分 (約 30 μs) の遅れが生じることがあります。

(注) 繰り返し設定 (例えば、毎週水曜 12 時 00 分と言った場合) を行なう場合、アラーム一致時に発生する INTRTC の割り込みルーチン内で、次回アラームの設定を行う必要があります。

(2) 1 Hz のパルスを出力する場合

PAGER<ENAALM> = “0”、RESTR<DIS1HZ> = “0”、<DIS16HZ> = “1” を設定後、PAGER<INTENA>=“1”を設定すると $\overline{\text{ALARM}}$ 端子に 1 Hz 周期の低速クロック 1 周期分の “0” パルスを出力します。また、同時に INTRTC 割り込みを出力します。

(3) 16 Hz のパルスを出力する場合

PAGER<ENAALM> = “0”、RESTR<DIS1HZ> = “1”、<DIS16HZ> = “0” を設定後、PAGER<INTENA>=“1”を設定すると $\overline{\text{ALARM}}$ 端子に 16 Hz 周期の低速クロック 1 周期分の “0” パルスを出力します。また、同時に INTRTC 割り込みを出力します。

18. Flash 動作説明

Flash 機能について、ハードウェアの構成およびその動作を説明します。

18.1 フラッシュメモリ

18.1.1 特長

1) メモリ容量

TMPM330FYWFG はフラッシュメモリを搭載しています。

各ブロック個別に書き込みを行うことができます。CPU から内蔵フラッシュメモリをアクセスする場合、データバス幅は 32 ビットとなります。

2) 書き込み/消去時間

書き込みはページ単位で行います。1 ページは 128 ワードです。

1 ページあたりの書き込み時間は、ワード数にかかわらず 1.25ms(Typ.)です。

消去時間は 1 ブロックあたり 0.1sec(Typ.)です。

1 チップあたりの書き込み時間、消去時間は以下ようになります。

製品	メモリ容量	ブロック構成				ワード数	書き込み時間	消去時間
		128KB	64KB	32KB	16KB			
TMPM330FYWFG	256KB	1	1	2	-	128	0.64sec	0.4sec

(注) 上記の値は理論時間を表しており、データ転送時間などは含まれていません。

チップ当たりの時間はユーザーの書き替え方法により異なります。

3) プログラミング方法

ユーザーのボード上で書き替えが可能なオンボードプログラミングモードがあります。

・オンボードプログラミングモード

3-1) ユーザーブートモード

ユーザー独自の書き替え方法をサポート

3-2) シングルブートモード

シリアル転送(当社オリジナル)での書き替え方法をサポート

4) 書き替え方式

本デバイス内蔵のフラッシュメモリは、一部の機能を除き JEDEC 標準機能に準拠しています。このため、外部メモリとしてフラッシュメモリをご使用になられている場合でも、本 LSI への移行が容易です。また、フラッシュメモリ内に書き込み、チップ消去など自動で行う回路を内蔵していますので、書き込み、消去動作自身に係わる複雑なフローをユーザーがプログラムで組む必要がありません。

5) プロテクト/セキュリティ機能

本デバイスでは、ライターでのフラッシュデータの読み出しを禁止する、セキュリティ機能を追加しています。一方、書き替え禁止を設定するライト/消去プロテクトは、コマンド(ソフトウェア)による対応のみで 12 V 電圧を印加して設定する方式(ハードウェア)には対応できません。プロテクト、セキュリティ機能の詳細については、プロテクト/セキュリティ機能の章を参照して下さい。

注) パスワードが消去データ(0xFF)の場合、容易にパスワードの照合が可能になり、セキュリティの確保が難しくなります。シングルブートモードを使用しない場合も固有の値を置くことを推奨します。

JEDEC 準拠の機能	変更、追加、削除した機能
<ul style="list-style-type: none"> • 自動プログラム • 自動チップ消去 • 自動ブロック消去 • データポーリング/トグルビット 	<p><変更> ブロック単位でのライト/消去プロテクト (ソフトウェアプロテクトのみサポート)</p> <p><削除> 消去レジューム/サスペンド機能</p>

18.1.2 フラッシュ部ブロック図

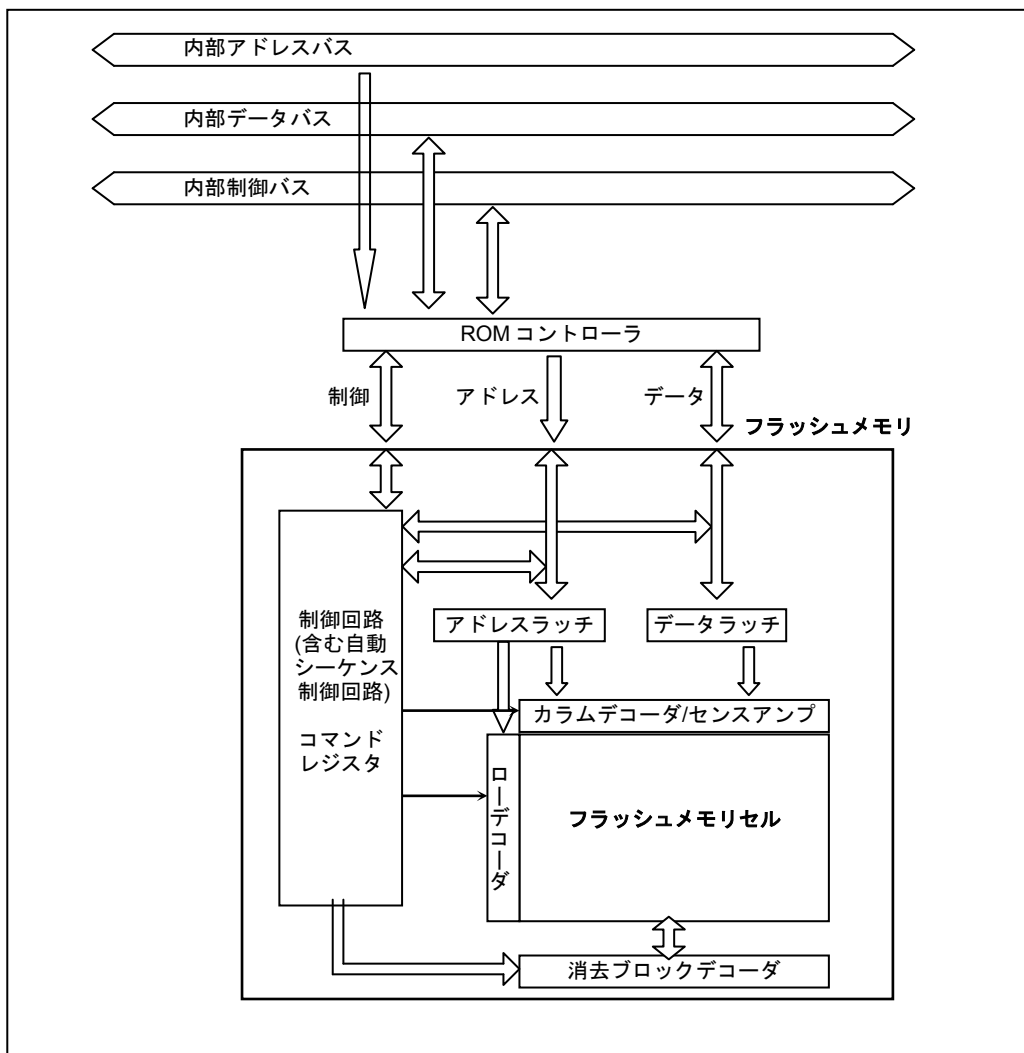


図 18-1 フラッシュ部ブロック図

18.2 動作モード

本デバイスは内蔵フラッシュメモリを使用しない場合を含めて、3通りの動作状態(モード)が存在します。

表 18-1 動作モード説明

動作モード名	動作の内容
シングルチップモード	リセット解除後、内蔵のフラッシュメモリから起動します。
ノーマルモード	本動作モードの中で、ユーザーのアプリケーションプログラムを実行するモードと、ユーザーのセット上でフラッシュメモリの書き替えを実行するモードとに分けて定義します。前者を「ノーマルモード」、後者を「ユーザーブートモード」と呼びます。
ユーザーブートモード	この両者の切り替えはユーザーが独自に設定できます。 例えばポート A0 が '1' のときノーマルモード、'0' のときにユーザーブートモードというように自由に設計することが可能です。 ユーザーはアプリケーションプログラムの一部に切り替えを判定するためのルーチンを準備してください。
シングルブートモード	リセット解除後、内蔵する Boot ROM (Mask ROM) から起動します。Boot ROM には、本デバイスのシリアルポートを経由してユーザーのセット上で書き替えを行うことができるアルゴリズムがプログラムされています。シリアルポートにより外部ホストと接続し、規定されたプロトコルでデータの転送を行うことで内蔵フラッシュの書き替えが実行できます。

上記表 18-1でプログラムが可能なフラッシュメモリの動作モードは ユーザーブートモード、シングルブートモードの2つです。ユーザーのセット上で内蔵フラッシュメモリの書き替えが可能なモードは、ユーザーブートモードとシングルブートモードで、この2つをオンボードプログラミングモードと定義します。

シングルチップ、シングルブートの各動作モードは、リセット状態で $\overline{\text{BOOT}}$ (PH0)端子のレベルを外部で設定することにより決定されます。

CPU は状態設定後リセットを解除することにより各動作モードで動作を開始します。 $\overline{\text{BOOT}}$ (PH0)端子については、それぞれモード設定後は動作中にレベルの変更がないようにしてください。以下に動作モードの設定方法とモード遷移図を示します。

表 18-2 動作モード設定表

動作モード	端子	
	$\overline{\text{RESET}}$	$\overline{\text{BOOT}}$ (PH0)
シングルチップモード	0 → 1	1
シングルブートモード	0 → 1	0

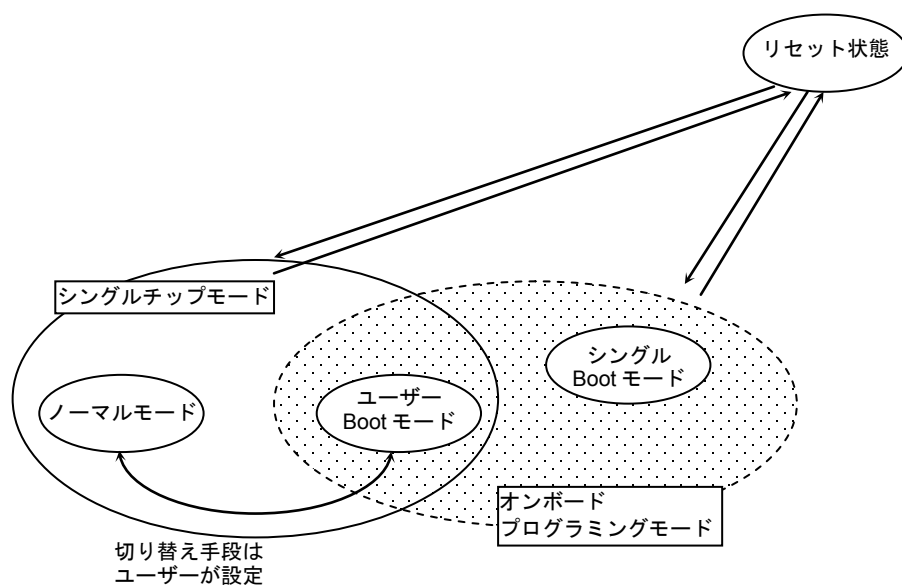


図 18-2 モード遷移図

18.2.1 リセット動作

本デバイスにリセットをかけるには、電源電圧が動作電圧範囲内で、かつ内部発振器の発振が安定した状態で、少なくとも 12 システムクロック間 (40MHz 動作で 0.3 μs (リセット後は、クロックギア 1/1 モード)) $\overline{\text{RESET}}$ 入力を“0”にしてください。

- (注 1) 電源投入後は、電源電圧および発振が安定した状態から 700 μs 以上経過してからリセット解除させてください。
- (注 2) 内蔵フラッシュの自動プログラム/消去動作中にハードウェアリセットを行う場合は、システムクロックによらず 0.5 μs 以上のリセット期間が必要となります。またこの場合、リセット解除後読み出しが可能になるまで 2ms 程度の時間がかかります。

18.2.2 ユーザーブートモード (シングルチップモード)

ユーザーブートモードは、ユーザー独自のフラッシュメモリ書き替えルーチンを使う方法です。ユーザーアプリケーション上で用意されているフラッシュメモリ書き替え用のプログラムで用いる、データ転送バスがシリアル I/O と異なる場合に使用します。動作はシングルチップモード上で行います。このため、シングルチップモードにおいて通常のユーザーアプリケーションプログラムが動作しているノーマルモードから、フラッシュを書き替えるためのユーザーブートモードに移行する必要があります。したがって、条件判定を行うプログラムをユーザーアプリケーションの中で、リセット処理プログラムの中に組み込んでください。

このモード切り替えの条件設定は、本デバイスの I/O を使用してユーザーのシステムセット条件に合わせて独自に構築してください。また、ユーザーブートモード移行後に使用するユーザー独自のフラッシュメモリ書き替えルーチンも同様にユーザーアプリケーションの中にあらかじめ組み込んでおき、ユーザーブートモード移行後にこれらのルーチンを使用して書き替えを行ってください。なお、内蔵フラッシュメモリは消去/書き込み動作モード中はフラッシュのデータを読み出せません。このため、書き替えルーチンはフラッシュメモリエリア外に格納して実行させる必要があります。また、シングルチップモード (通常動作モード) 中に誤ってフラッシュの内容を書き替えまいよう、書き替え処理が完了した後、必要なブロックにライト/消去プロテクトをかけておくことを推奨します。あわせて、ユーザーブートモード中は、ノンマスカブルを含めたすべての割り込み発生を禁止してください。

書き替えルーチンを内蔵フラッシュメモリに置く場合と、外部から転送する場合の 2 ケースを例に、以下 (1-A)、(1-B) にその手順を説明します。フラッシュメモリへの書き込み/消去方法の詳細は、「18.3 オンボードプログラミングでのフラッシュメモリ書き込み/消去」を参照してください。

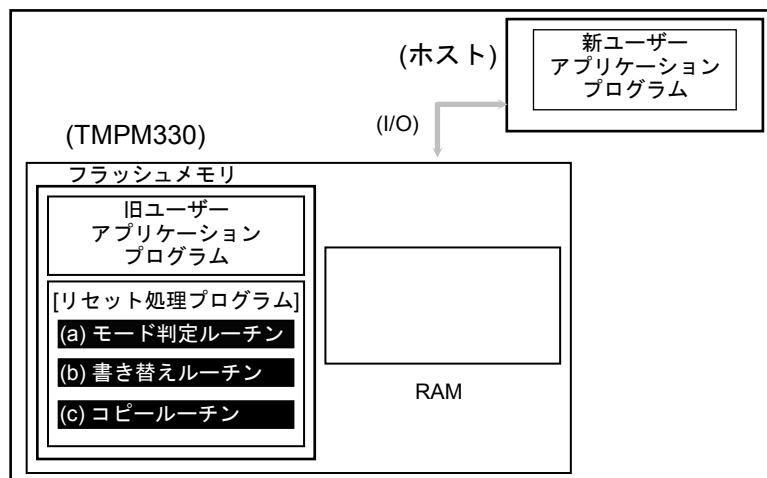
ユーザーブートモード

(1-A) 書き換えルーチンをフラッシュメモリに内蔵する場合の手順例

(Step-1)

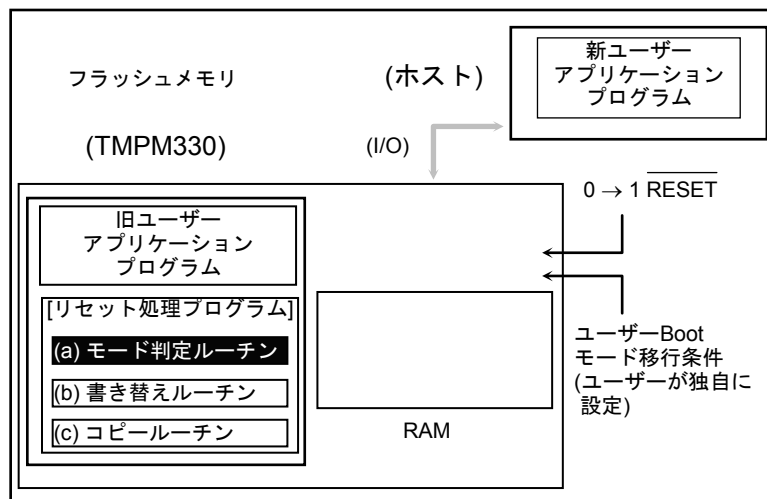
ユーザーは、あらかじめどのような条件 (例えば端子状態) に設定されたらユーザーブートモードに移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合った回路の設計、プログラムの作成を行います。ユーザーは本デバイスをボードに組み込む前に、あらかじめフラッシュメモリ上の任意のブロックにライターなどを使用して以下に示す 3つのプログラムを書き込んでおきます。

- (a) モード判定ルーチン: 書き換え動作に移るためのプログラム
- (b) フラッシュ書き換えルーチン: 書き換えデータを外部から取り込み、フラッシュメモリを書き替えるためのプログラム
- (c) コピールーチン: 上記 (b) を内蔵 RAM または外部メモリにコピーするためのプログラム



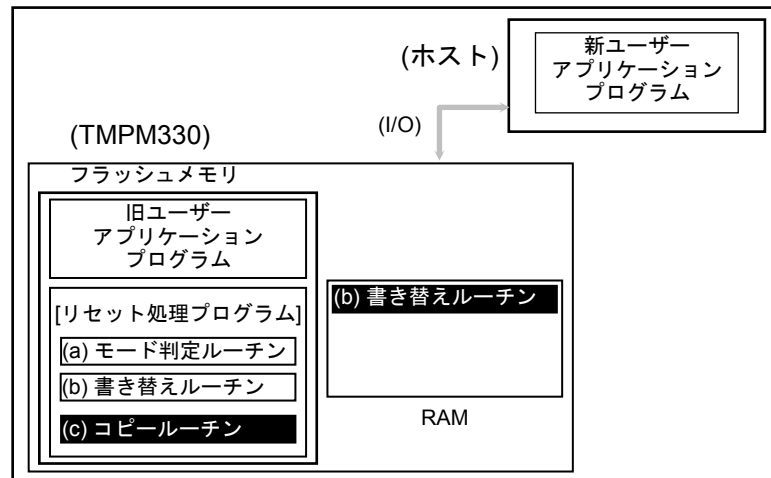
(Step-2)

以下リセット処理プログラム内にこれらのルーチンを組み込んだ場合について説明します。まず、リセット解除後のリセット処理プログラムにおいてユーザーブートモードへの移行を判定します。このとき、移行条件が整っていれば、プログラムは書き換えのためのユーザーブートモードに移ります。(ユーザーブートモードに移行した場合は、これ以降割り込みを使わないでください。)

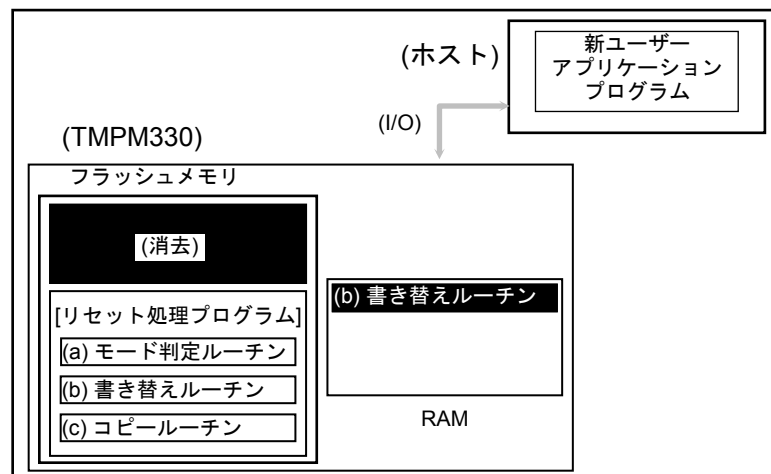


(Step-3)

ユーザーブートモードに移ると、(c) コピールーチンを使用して、(b) 書き替えルーチンを内部 RAM にコピーします。

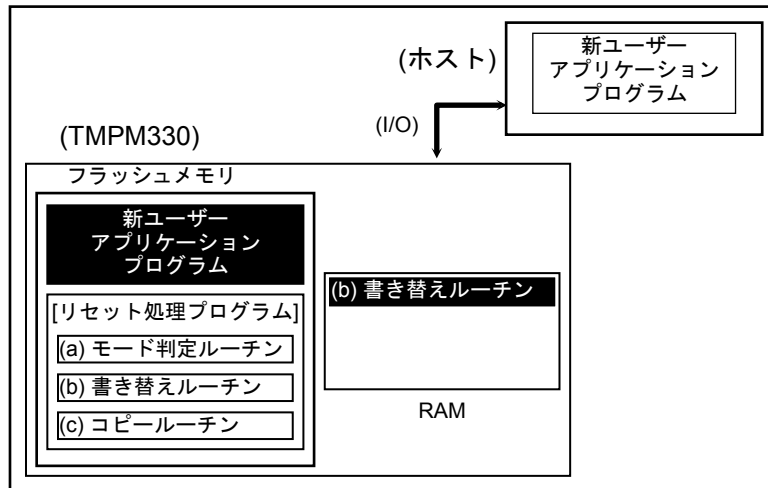
**(Step-4)**

RAM 上の書き替えルーチンへジャンプし、旧ユーザープログラムエリアのライト/消去プロテクトを解除して、消去（ブロック単位）を行います。



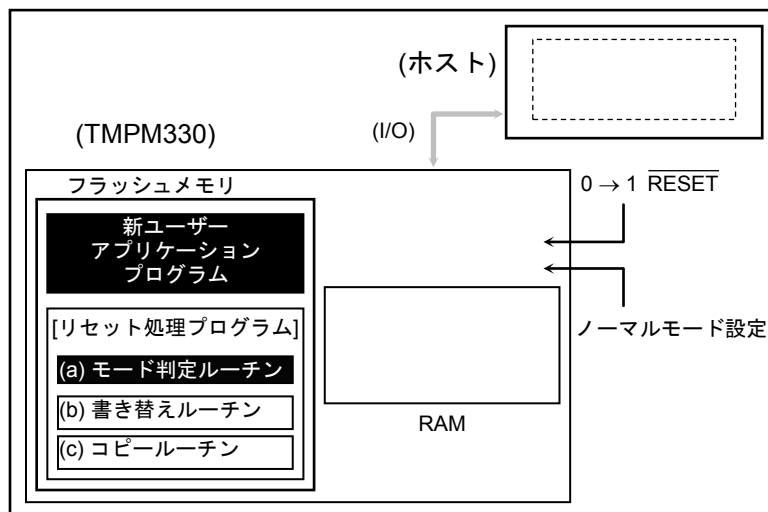
(Step-5)

さらに、RAM 上の書き替えルーチンを実行して、転送元 (ホスト) より新ユーザーアプリケーションプログラムのデータをロードし、フラッシュメモリの消去したエリアに書き込みを行います。書き込みが完了したら、ユーザープログラムエリアのライト/消去プロテクトをオンにします。



(Step-6)

$\overline{\text{RESET}}$ 入力端子を“0”にしてリセットを行い、設定条件をノーマルモードの設定にします。リセット解除後、新ユーザーアプリケーションプログラムで動作を開始します。



(1-B) 書き替えルーチンを外部から転送する手順例

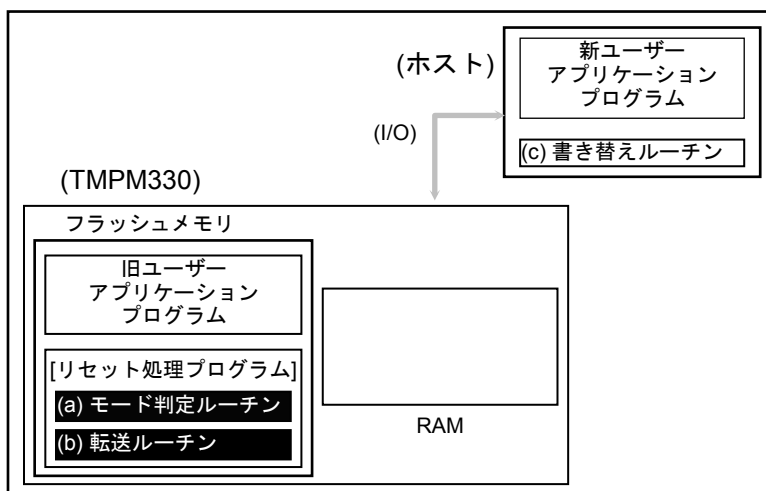
(Step-1)

ユーザーは、あらかじめどのような条件(例えば端子状態)に設定されたらユーザーブートモードに移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合った回路の設計、プログラムの作成を行います。ユーザーは本デバイスをボードに組み込む前に、あらかじめフラッシュメモリ上の任意のブロックにライタなどを使用して以下に示す 2 つのプログラムを書き込んでおきます。

- (a) モード判定ルーチン: 書き替え動作に移るためのプログラム
- (b) 転送ルーチン: 書き替えプログラムを外部から取り込むためのプログラム

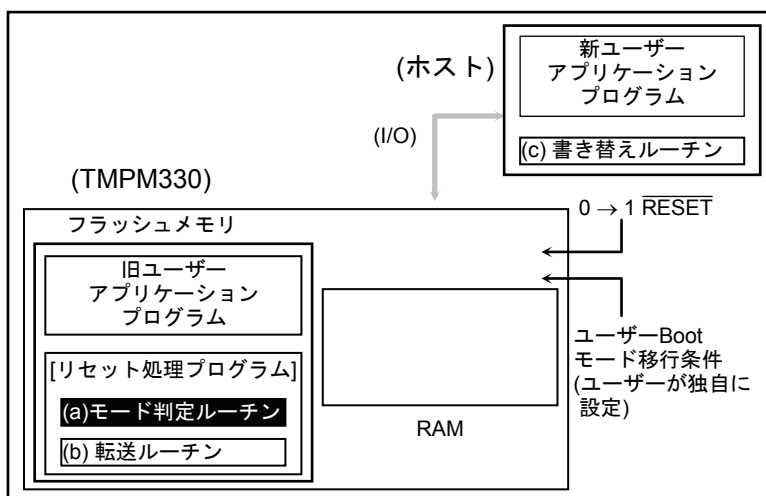
また、下記に示すプログラムはホスト上に用意します。

- (c) 書き替えルーチン: 書き替えを行うためのプログラム



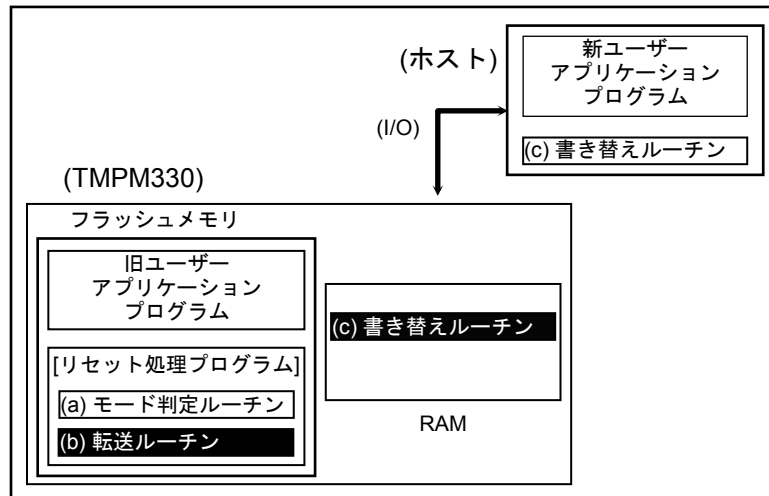
(Step-2)

以下リセット処理プログラム内にこれらのルーチンを組み込んだ場合について説明します。まず、リセット解除後のリセット処理プログラムにおいてユーザーブートモードへの移行を判定します。このとき、移行条件が整っていれば、プログラムは書き替えのためのユーザーブートモードに移ります。(ユーザーブートモードに移行した場合は、これ以降割り込みを使わないでください。)



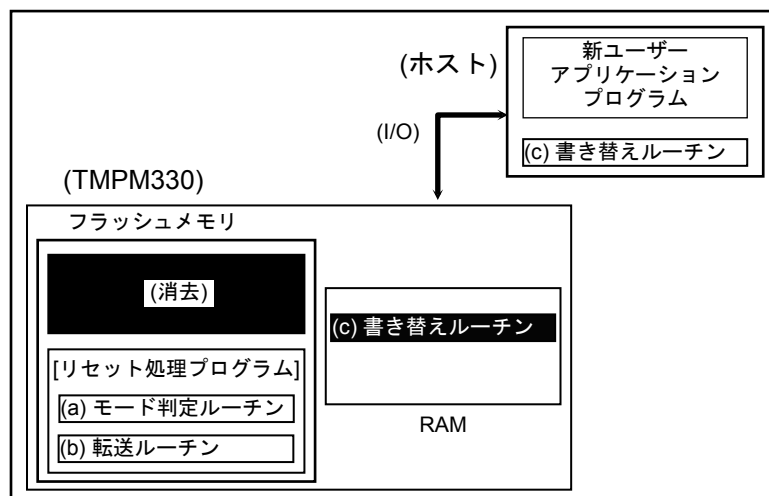
(Step-3)

ユーザーブートモードに移ると、(b) 転送ルーチンを使用して、転送元 (ホスト) より (c) 書き替えルーチンを内部 RAM にロードします。



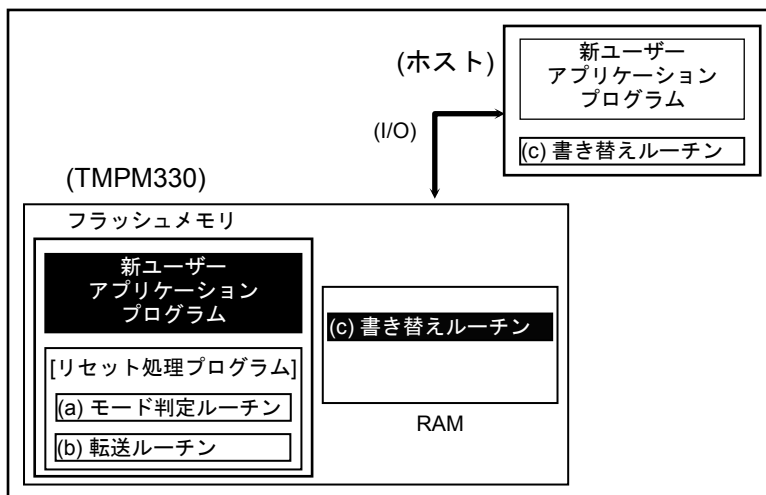
(Step-4)

RAM 上の書き替えルーチンへジャンプし、旧ユーザープログラムエリアのライト/消去プロテクトを解除して、消去 (ブロック単位) を行います。



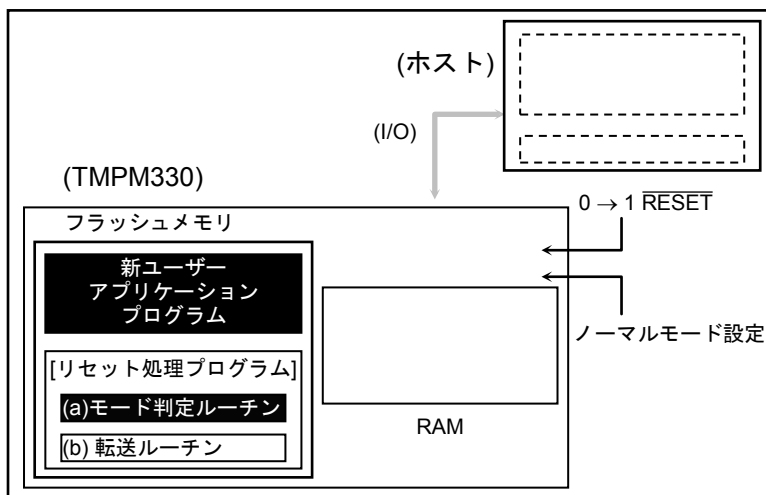
(Step-5)

さらに、RAM 上の (c) 書き替えルーチンを実行して、転送元 (ホスト) より新ユーザーアプリケーションプログラムのデータをロードし、消去したエリアに書き込みを行います。書き込みが完了したら、ユーザープログラムエリアのライト/消去プロテクトをオンにします。



(Step-6)

$\overline{\text{RESET}}$ 入力端子を“0”にしてリセットを行い、設定条件をノーマルモードに設定します。リセット解除後、新ユーザーアプリケーションプログラムで動作を開始します。



18.2.3 シングルブートモード

内蔵ブート ROM (マスク ROM) を起動して、ブート ROM のプログラムを利用してフラッシュメモリを書き替える方法です。このモードでは、内蔵ブート ROM が割り込みベクタテーブルを含む領域にマッピングされ、ブート ROM プログラムが実行されます。また、フラッシュメモリはブート ROM 領域とは別のアドレス空間にマッピングされます。

ブートモードでは、コマンドおよびデータをシリアル転送してフラッシュメモリの書き替えを行います。本デバイスの SIO (SIO0) と外部ホストを接続し、外部ホスト側から本デバイスの内蔵 RAM に書き替えプログラムをコピーし、RAM 上の書き替えルーチンを実行してフラッシュメモリの書き替えを行います。書き替えルーチンは、ホスト側からコマンドおよび書き替えデータを送出することにより実行します。ホスト側との通信の詳細は後述のプロトコルに従ってください。

RAM へのプログラム転送は、ユーザーの ROM データ、セキュリティ確保のため、実行に先立ちユーザーパスワードの照合を行います。パスワードが一致しない場合は、RAM 転送そのものが実行されません。なお、シングルブートモードでもユーザーブートモードと同様、割り込み禁止状態で行います。シングルブートモード時、ブート ROM プログラムは NORMAL モードで動作します。

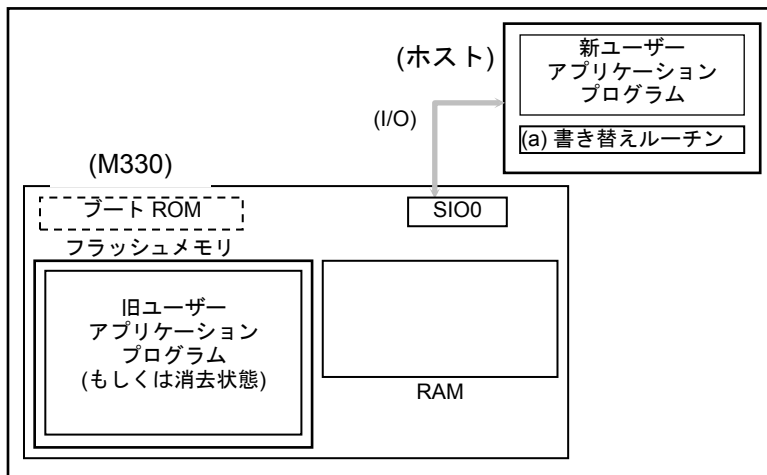
シングルチップモード (通常動作モード) 中に誤ってフラッシュメモリの内容を書き替えないう、書き替え処理が完了したら必要なブロックにライト/消去プロテクトをかけておくことを推奨します。

シングルブートモード

(2-A)内蔵ブート ROM の書き替えアルゴリズムを利用する場合

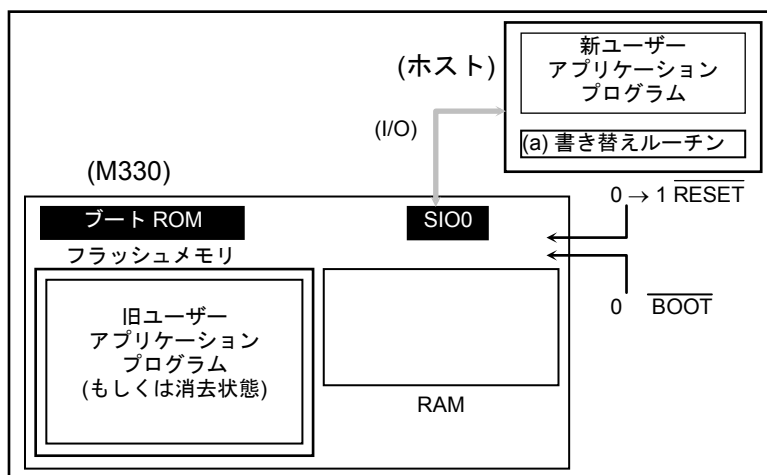
(Step-1)

フラッシュメモリの状態は旧バージョンのユーザープログラムが書かれた状態でも、消去されている状態でも構いません。書き替えルーチン、書き替えデータなどの転送は SIO (SIO0) を経由して行いますので、ボード上で本デバイスの SIO (SIO0) と外部ホストとをつなげます。書き替えを行うための (a) 書き替えルーチンはホスト上に用意します。



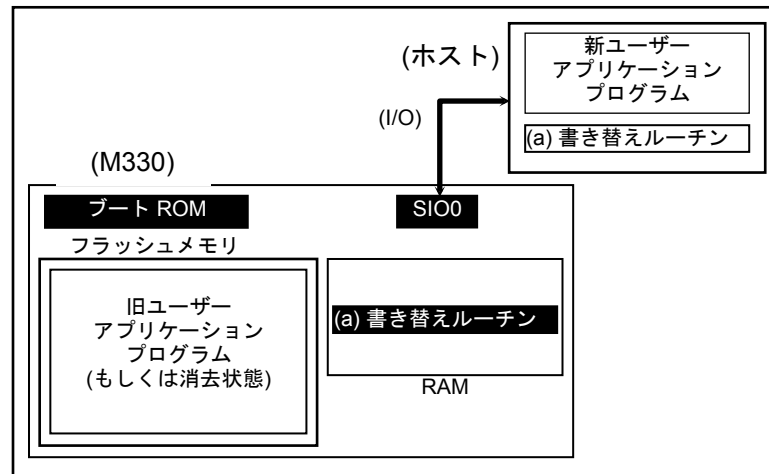
(Step-2)

ブートモードの端子条件設定でリセットを解除し、ブート ROM で起動します。ブートモードの手順に従い、SIO0 を経由して転送元 (ホスト) より (a) 書き替えルーチンの転送を行いますが、最初にユーザーアプリケーションプログラム上に記録されているパスワードとの照合を行います。(フラッシュメモリが消去されている状態でも、消去データ (FFH) をパスワードとして照合を行います。)

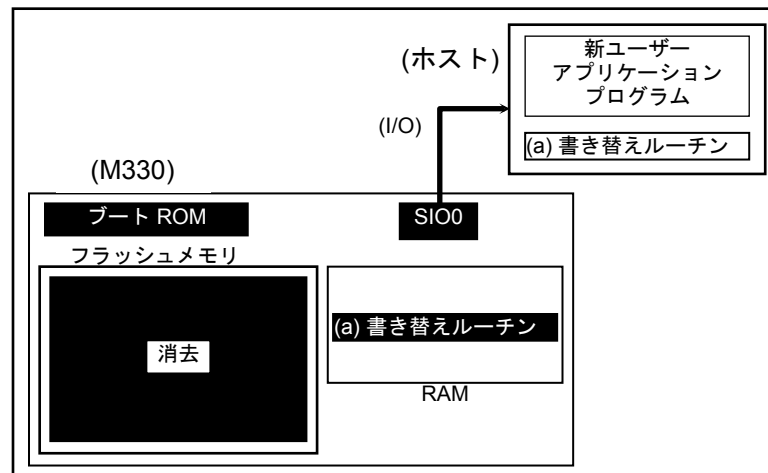


(Step-3)

パスワードの照合が終了すると、転送元（ホスト）から（a）書き替えルーチンを転送します。ブート ROM はそのルーチンを内部 RAM にロードします。ただし、RAM 上のアドレス 0x2000_0400 から RAM の最終番地の範囲に格納してください。

**(Step-4)**

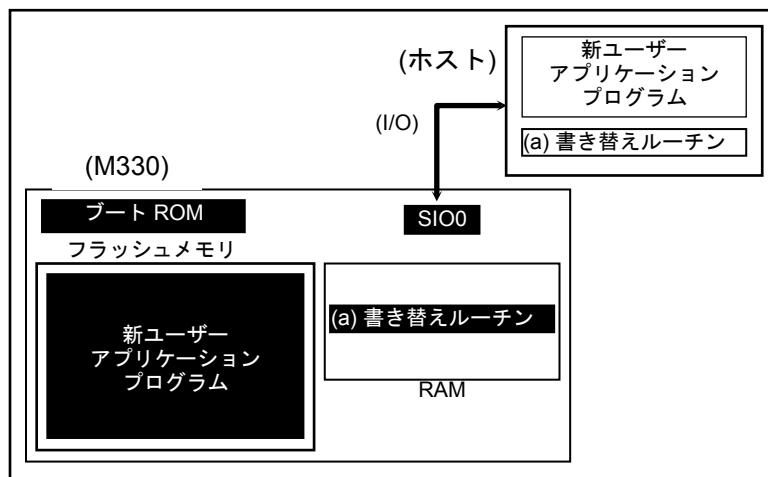
RAM 上の（a）書き替えルーチンへジャンプし、旧ユーザーアプリケーションプログラムエリアの消去を行います。（ブロック単位もしくは一括）



(Step-5)

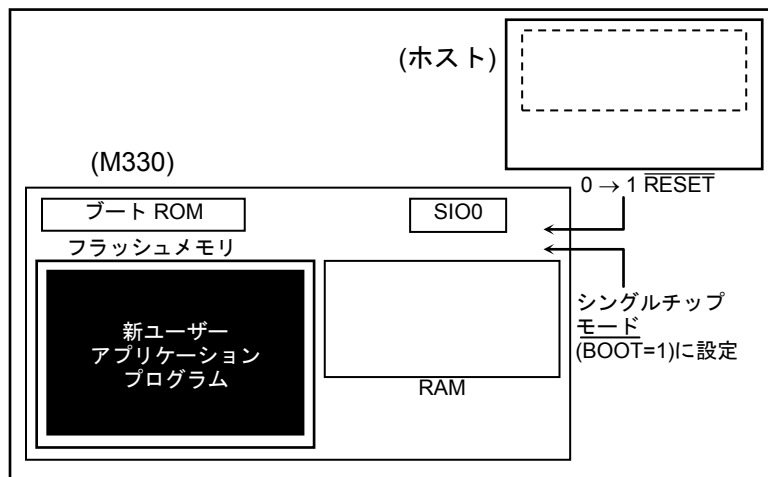
さらに、RAM 上の (a) 書き替えルーチンを実行して、転送元(ホスト)より新ユーザーアプリケーションプログラムのデータをロードし、フラッシュメモリの消去したエリアに書き込みを行います。書き込みが完了したら、ユーザープログラムエリアのライト/消去プロテクトをオンにします。

下の例の場合、書き替えルーチンを転送したときと同じホストおよび SIO0 経由で書き替えデータも転送されていますが、RAM 上で動作を開始した以降では、ユーザー独自にデータバスおよび転送元を設定することもできます。方法に応じて、ボードのハードおよび書き替えルーチンを組み立ててください。



(Step-6)

書き込みが完了したら、一度ボードの電源を落とし、ホストと接続していたケーブルをはずします。この後、再度電源を入れ直し、シングルチップモード(ノーマルモード)起動し、新しいユーザーアプリケーションプログラムを実行します。



(1)モード設定

オンボードプログラミングを実行するためには、本デバイスをシングルブートモードで立ち上げます。シングルブートモードで立ち上がるための設定を以下に示します。

$$\overline{\text{BOOT}} (\text{PH0}) = 0$$

$$\overline{\text{RESET}} = 0 \rightarrow 1$$

$\overline{\text{RESET}}$ 入力端子を“0”の状態にして、 $\overline{\text{BOOT}} (\text{PH0})$ 端子をあらかじめ上記条件に設定します。その後 $\overline{\text{RESET}}$ 解除を行うとシングルブートモードで起動します。

(2) メモリマップ

図 18-3にシングルチップモードとシングルブートモードのメモリマップの比較を示します。図のように、シングルブートモードでは、内蔵フラッシュメモリは 0x3F80_0000 番地からマッピングされます。また、0x0000_0000 番地から 0x0000_0FFF 番地にはブート ROM (マスク ROM) がマッピングされます。

製品	Flash サイズ	RAM サイズ	Flash アドレス (シングルチップ / シングルブートモード)	RAM アドレス
TMPM330FYWFG	256KB	16KB	0x0000_0000 - 0x0003_FFFF (0x0007_FE00 - 0x0007_FFFF) (注) 0x3F80_0000 - 0x3F83_FFFF (0x3F87_FE00 0x3F87_FFFF) (注)	0x2000_0000 - 0x2000_3FFF

(注) TMPM330FYWFG では、256KB の FLASH 領域とは別に、製品情報読み出しコマンド用のデータ領域およびパスワード領域が 128 ワード (1 ページ) 分存在します。

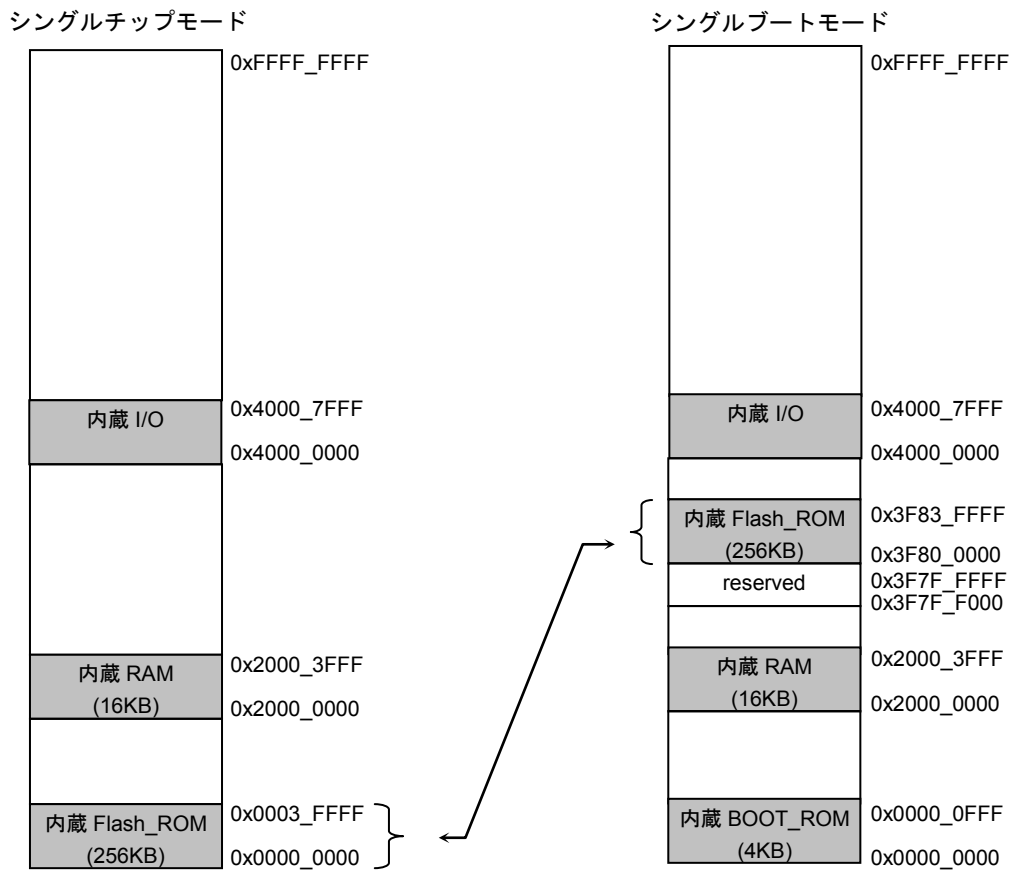


図 18-3 メモリマップの比較

(3) インタフェース仕様

シングルブートモードでの SIO 通信フォーマットを以下に示します。シリアル動作のモードは、UART (非同期通信) と I/O インタフェースモード両方に対応しています。オンボードプログラミングを実行するためには、書き込みコントローラ側の通信フォーマットも同様に設定する必要があります。

- UART で通信する場合

通信チャンネル	: SIO チャンネル 0
シリアル転送モード	: UART (非同期通信) モード、半二重通信、LSB ファスト
データ長	: 8 ビット
パリティビット	: なし
STOP ビット	: 1 ビット
ボーレート	: 任意のボーレート
- I/O インタフェースモードで通信する場合

通信チャンネル	: SIO チャンネル 0
シリアル転送モード	: I/O インタフェースモード、全二重通信、LSB ファスト
同期信号 (SCLK0)	: 入力モード
ハンドシェイク端子	: 出力モード PE4
ボーレート	: 任意のボーレート

表 18-3 端子の接続

端 子		インタフェース	
		UART	I/O インタフェースモード
電源系 端子	REGVCC	○	○
	AVCC	○	○
	DVCC	○	○
	CVCC	○	○
	REGVSS	○	○
	AVSS	○	○
	DVSS	○	○
	CVSS	○	○
モード設定端子	$\overline{\text{BOOT}}$ (PH0)	○	○
リセット端子	$\overline{\text{RESET}}$	○	○
通信端子	TXD0(PE0)	○	○
	RXD0(PE1)	○	○
	SCLK0(PE2)	×	○ (入力モード)
	PE4	×	○ (出力モード)

(4) データ転送フォーマット

動作コマンド、および各動作モード時のデータ転送フォーマットをそれぞれ表 18-4、表 18-6~表 18-9に示します。後述の「ブートプログラム動作説明」とあわせてお読みください。

表 18-4 動作コマンドデータ

動作コマンドデータ	動作モード
10H	RAM 転送
20H	フラッシュメモリ SUM
30H	製品情報読み出し
40H	フラッシュメモリチップ消去 及び プロテクトビット消去

(5) メモリの制約について

シングルブートモードでは、内蔵 RAM、内蔵 FlashROM に対して表 18-5 のように制約が付きます。

表 18-5 シングルブート時のメモリの制約

種別	制約内容
内蔵 RAM	0x2000_0000~0x2000_03FF 番地は BOOT_ROM のワークエリアになります。 RAM 転送のプログラムは 0x2000_0400 から RAM の最終番地に格納して下さい。
内蔵 ROM	以下の番地はソフトなどの ID 情報や password の格納エリアとなりますので、なるべくプログラムエリアとしての使用はさけて下さい。 TMPM330FYWFG : 0x3F87_FF00 - 0x3F87_FF0F

表 18-6 シングルブートプログラムの転送フォーマット [RAM 転送の場合]

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
ブート ROM	1 バイト目	シリアル動作モード & ボーレート設定 UART の場合 86H I/O インタフェースの場合 30H	指定された ボーレート (注 1)	—
	2 バイト目	—		シリアル動作モードに対する ACK 応答 UART の場合 正常 (設定可能) の場合 86H (ボーレートの設定が不可能と判断した場 合は動作停止) I/O インタフェースの場合 正常の場合 30H
	3 バイト目	動作コマンドデータ (10H)		—
	4 バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 × 1H 通信異常の場合 × 8H
	5 バイト目 ~ 16 バイト目	PASS WORD データ (12 バイト) 0x3F87_FF04~0x3F87_FF0F		—
	17 バイト目	5 ~ 16 バイト目の CHECK SUM 値		—
	18 バイト目	—		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 x1H 通信異常の場合 x8H
	19 バイト目	RAM 格納開始アドレス 31 ~ 24		—
	20 バイト目	RAM 格納開始アドレス 23 ~ 16		—
	21 バイト目	RAM 格納開始アドレス 15 ~ 8		—
	22 バイト目	RAM 格納開始アドレス 7 ~ 0		—
	23 バイト目	RAM 格納バイト数 15 ~ 8		—
	24 バイト目	RAM 格納バイト数 7 ~ 0		—
	25 バイト目	19 ~ 24 バイト目の CHECK SUM 値		—
	26 バイト目	—		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 x1H 通信異常の場合 x8H
	27 バイト目 ~ m バイト目	RAM 格納データ		—
m + 1 バイト目	27 ~ m バイト目の CHECK SUM 値	—		
m + 2 バイト目	—	CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 10H 異常の場合 x1H 通信異常の場合 x8H		
RAM	m + 3 バイト目	—	JUMP RAM 格納開始アドレス	

(注 1) I/O インタフェースモードの場合、1 バイト目と 2 バイト目のボーレートは、指定されたボーレート ÷ 16 で行ってください。

(注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。I/O インタフェースモードの場合は、通信異常の場合は発生しません。

(注 3) 19 バイト目 ~ 25 バイト目のデータは、RAM 上のアドレス 0x2000_0400 から RAM の最終番地の領域内に納まるようにプログラムしてください。

表 18-7 ブートプログラムの転送フォーマット [フラッシュメモリ SUM の場合]

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
ブート ROM	1バイト目	シリアル動作モード & ボーレート設定 UART の場合 86H I/O インタフェースの場合 30H	指定された ボーレート *1	—
	2バイト目	—		シリアル動作モードに対する ACK 応答 UART の場合 正常 (設定可能) の場合 86H (ボーレートの設定が不可能と判断した場 合は動作停止) I/O インタフェースの場合 正常の場合 30H
	3バイト目	動作コマンドデータ (20H)		—
	4バイト目	—		動作コマンドに対する ACK 応答 *2 正常の場合 20H 異常の場合 x1H 通信異常の場合 x8H
	5バイト目	—		SUM (上位)
	6バイト目	—		SUM (下位)
	7バイト目	—		5~6バイト目の CHECK SUM 値
	8バイト目	(次の動作コマンドデータ待ち)		—

*1: I/O インタフェースモードの場合、1バイト目と2バイト目のボーレートは、指定されたボーレート ÷ 16 で行ってください。

*2: 異常応答後は、動作コマンド (3バイト目) 待ちになります。I/O インタフェースモードの場合は、通信異常の場合は発生しません。

表 18-8 ブートプログラムの転送フォーマット [デバイス情報読み出しの場合] (1/3)

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
ブート ROM	1バイト目	シリアル動作モード & ボーレート設定 UARTの場合 86H I/O インタフェースの場合 30H	指定された ボーレート (注 1)	—
	2バイト目	—		シリアル動作モードに対する ACK 応答 UARTの場合 正常 (設定可能) の場合 86H (ボーレートの設定が不可能と判断した 場合は、動作停止) I/O インタフェースの場合 正常の場合 30H
	3バイト目	動作コマンドデータ (30H)		—
	4バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 30H 異常の場合 ×1H 通信異常の場合 ×8H
	5バイト目	—		フラッシュメモリデータ 0x3F87_FF00 番地
	6バイト目	—		フラッシュメモリデータ 0x3F87_FF01 番地
	7バイト目	—		フラッシュメモリデータ 0x3F87_FF02 番地
	8バイト目	—		フラッシュメモリデータ 0x3F87_FF03 番地
	9バイト目 ~ 20バイト目	—		製品名 (アスキーコード、12 バイト) 9 バイト目から 'TMPM330FD_ _'
	21バイト目 ~ 24バイト目	—		Password 比較開始アドレス (4 バイト) 21 バイト目から 04H, FFH, 87H, 3FH
	25バイト目 ~ 28バイト目	—		RAM 開始アドレス (4 バイト) 25 バイト目から 00H, 00H, 00H, 20H
	29バイト目 ~ 32バイト目	—		ダミーデータ (4 バイト) 29 バイト目から 00H, 00H, 00H, 00H
	33バイト目 ~ 36バイト目	—		RAM 終了アドレス (4 バイト) 33 バイト目から FFH, 7FH, 00H, 20H
	37バイト目 ~ 40バイト目	—		ダミーデータ (4 バイト) 37 バイト目から 00H, 00H, 00H, 00H
	41バイト目 ~ 44バイト目	—		ダミーデータ (4 バイト) 41 バイト目から 00H, 00H, 00H, 00H
	45バイト目 ~ 46バイト目	—		ダミーデータ (2 バイト) 45 バイト目から 00H, 00H

表 18-8 ブートプログラムの転送フォーマット [デバイス情報読み出しの場合] (2/3)

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
ブート ROM	47 バイト目 ~ 50 バイト目	—		フラッシュメモリ開始アドレス (4 バイト) 47 バイト目から 00H, 00H, 80H, 3FH
	51 バイト目 ~ 54 バイト目	—		フラッシュメモリ終了アドレス (4 バイト) 51 バイト目から FFH, FFH, 87H, 3FH
	55 バイト目 ~ 56 バイト目	—		フラッシュメモリブロック分割数情報(2 バイト) 55 バイト目から 06H, 00H
	57 バイト目 ~ 60 バイト目	—		フラッシュメモリ同一ブロックサイズ(16K)の 開始アドレス (4 バイト) 57 バイト目から 00H, 00H, 00H, 00H
	61 バイト目 ~ 64 バイト目	—		フラッシュメモリ同一ブロックサイズ(16K)の サイズ (ハーフワード表現) (4 バイト) 61 バイト目から 00H, 20H, 00H, 00H
	65 バイト目	—		フラッシュメモリ同一ブロックサイズ(16K)の 個数 (1 バイト) 00H
	66 バイト目 ~ 69 バイト目	—		フラッシュメモリ同一ブロックサイズ(32K)の 開始アドレス (4 バイト) 66 バイト目から 00H, 00H, 80H, 3FH
	70 バイト目 ~ 73 バイト目	—		フラッシュメモリ同一ブロックサイズ(32K)の サイズ (ハーフワード表現) (4 バイト) 70 バイト目から 00H, 40H, 00H, 00H
	74 バイト目	—		フラッシュメモリ同一ブロックサイズ(32K)の 個数 (1 バイト) 02H
	75 バイト目 ~ 78 バイト目	—		フラッシュメモリ同一ブロックサイズ(64K)の 開始アドレス (4 バイト) 75 バイト目から 00H, 00H, 81H, 3FH
	79 バイト目 ~ 82 バイト目	—		フラッシュメモリ同一ブロックサイズ(64K)の サイズ (ハーフワード表現) (4 バイト) 79 バイト目から 00H, 80H, 00H, 00H
	83 バイト目	—		フラッシュメモリ同一ブロックサイズ(64K)の 個数 (1 バイト) 83 バイト目から 01H

表 18-8 ブートプログラムの転送フォーマット [デバイス情報読み出しの場合] (3/3)

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
ブート ROM	84 バイト目 ~ 87 バイト目	—		フラッシュメモリ同一ブロックサイズ(128K)の 開始アドレス (4 バイト) 84 バイト目から 00H, 00H, 82H, 3FH
	88 バイト目 ~ 91 バイト目	—		フラッシュメモリ同一ブロックサイズ(128K)の サイズ (ハーフワード表現) (4 バイト) 70 バイト目から 00H, 00H, 01H, 00H
	92 バイト目	—		フラッシュメモリ同一ブロックサイズ(128K)の 個数 (1 バイト) 03H
	93 バイト目	—		5 ~ 92 バイト目の CHECK SUM 値
	94 バイト目	(次の動作コマンドデータ待ち)		—

(注 1) I/O インタフェースモードの場合、1 バイト目と 2 バイト目のボーレートは、指定されたボーレート ÷ 16 で行ってください。

(注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。I/O インタフェースモードの場合は、通信異常の場合は発生しません。

(注 3) TMPM330FYWFG のフラッシュメモリ終了アドレスは、0x3F83_FFFF です。

256KB の FLASH 領域とは別に、製品情報読み出しコマンド用のデータ領域およびパスワード領域が 128 ワード (1 ページ) 存在します。(0x3F87_FE00 - 0x3F87_FFFF)

表 18-9 ブートプログラムの転送フォーマット
[フラッシュメモリチップ消去 及び プロテクトビット消去の場合]

	転送バイト数	コントローラ→本デバイスへの 転送データ	ボーレート	本デバイス→コントローラへの 転送データ
ブート ROM	1バイト目	シリアル動作モード & ボーレート設定 UART の場合 86H I/O インタフェースの場合 30H	指定された ボーレート *1	—
	2バイト目	—		シリアル動作モードに対する ACK 応答 正常 (設定可能) の場合 ・UART の場合 86H ・I/O インタフェースの場合 30H (ボーレートの設定が不可能と判断した 場合は動作停止)
	3バイト目	動作コマンドデータ (40H)		—
	4バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 40H 異常の場合 x1H 通信異常の場合 x8H
	5バイト目	消去イネーブルコマンドデータ (54H)		—
	6バイト目	—		動作コマンドに対する ACK 応答 (注 2) 正常の場合 54H 異常の場合 x1H 通信異常の場合 x8H
	7バイト目	—		消去コマンドに対する ACK 応答 正常の場合 4FH 異常の場合 4CH
	8バイト目	(次の動作コマンドデータ待ち)		—

*1: I/O インタフェースモードの場合、1バイト目と2バイト目のボーレートは、指定されたボーレート ÷ 16 で行ってください。

*2: 異常応答後は、動作コマンド (3バイト目) 待ちになります。I/O インタフェースモードの場合は、通信異常の場合は発生しません。

(6) ブートプログラム 動作説明

シングルブートモードで立ち上げるとブートプログラムが起動し、以下の機能を提供します。詳細は、1) RAM 転送コマンド～4) フラッシュメモリチップ消去コマンドに記載してありますので、参照してください。

以下、特にことわりのない限りアドレスは仮想アドレスで表記します。

1. RAM 転送コマンド

RAM 転送は、コントローラから送られてくるデータを内蔵 RAM へ格納します。転送が正常に終了するとユーザープログラムの実行を開始します。ユーザープログラム領域として、ブートプログラムで使用する領域(0x2000_0000～0x2000_03FF)を除く、0x2000_0400以降を使用可能です。実行開始アドレスは、RAM 格納開始アドレスになります。

この RAM 転送機能により、ユーザー独自のオンボードプログラミング制御を行うことができます。ユーザープログラムでオンボードプログラミングを実行するためには、18.3章で説明するフラッシュメモリコマンドシーケンスを使う必要があります。RAM 転送コマンドは、実行に先立ちパスワードの照合結果をチェックします。パスワードが一致していない場合、実行されません。

注) パスワードが消去データ(0xFF)の場合、容易にパスワードの照合が可能になり、セキュリティの確保が難しくなります。シングルブートモードを使用しない場合も固有の値を置くことを推奨します。

2. フラッシュメモリ SUM コマンド

フラッシュメモリ全領域の SUM を計算しその結果を返します。ブートプログラムではフラッシュメモリの全エリアのデータを読み出す動作コマンドはサポートしていません。その代わりに、このフラッシュメモリ SUM コマンドがあります。SUM を読み出すことで、アプリケーションプログラムのレビジョンを管理することができます。

3. 製品情報読み出しコマンド

本デバイスの製品名やメモリ情報などを返します。デバイス情報読み出しコマンドでは、フラッシュメモリの一部エリアのデータを返します。フラッシュメモリ SUM コマンド以外に、このデータを用いることで、アプリケーションプログラムのレビジョンを管理することができます。

製品	対象エリア
TMPM330FYWFG	0x3F87_FF00 - 0x3F87_FF03

4. フラッシュメモリチップ消去 及び プロテクトビット消去

このコマンドはすべてのブロックのフラッシュメモリを消去します。ライト/消去プロテクト及び、セキュリティ状態にかかわらず、メモリセルの全てのブロックを消去し、全てのブロックのライト/消去プロテクトを消去します。コマンド終了後、SECBIT レジスタの<SECBIT>ビットは”1”になります。このコマンドは、パスワードを忘れた場合のブートプログラムの操作を回復する機能も兼用しているため、パスワード比較は行っていません。

1) RAM 転送コマンド (表 18-6参照)

- 1 バイト目のデータは、シリアル動作モードを判定するデータになります。シリアルの動作モードを認める方法は、後述の「シリアル動作モード判定」を参照してください。シリアルの動作モードで **UART** と判定した場合は、ボーレートの設定が可能かどうかを判定します。1 バイト目のデータは、受信を禁止した状態 ($SC0MOD0<RXE>=0$) にしています。

- **UART** で通信を行いたい場合

コントローラからターゲットボードへは、**UART** の設定で、所望のボーレートでデータを **86H** にして送信してください。シリアルの動作モードの判定で **UART** と判定した場合、ボーレートの設定が可能かどうかを判定します。設定が不可能と判断した場合は動作を停止するため、通信が行えなくなります。ボーレートの設定が可能かどうかを判定する方法については、後述の「ボーレートの設定方法」を参照してください。

- **I/O インタフェース** で通信を行いたい場合

コントローラからターゲットボードへは、同期式の設定で、所望のボーレート $\div 16$ でデータを **30H** にして送信してください。2 バイト目も同様に、所望のボーレート $\div 16$ にしてください。所望のボーレートで転送するのは、3 バイト目 (動作コマンドデータ) からにしてください。

I/O インタフェース の場合、CPU が受信端子を入力ポートとして見ており、その入力ポートのレベルの変化をモニタしています。したがって、ボーレートが早い場合や動作周波数が多い場合は、CPU はレベルの変化を判別できないことがあります。これを防ぐために **I/O インタフェース** の場合、ボーレートは所望のボーレート $\div 16$ で指定します。**I/O インタフェース** と判定した場合、**SCLK** 入力モードになります。コントローラは、**AC** タイミングを満足するボーレートで送信を行ってください。**I/O インタフェース** の場合、受信エラーフラグのチェックは行いません。したがって、**ACK** 応答データの通信異常 **ACK (bit 3) (x8H)** はありません。

- 2 バイト目の送信データは、1 バイト目のシリアル動作モード設定データに対する **ACK** 応答データになります。1 バイト目のデータが、**UART** と判定されボーレートの設定が可能な場合 **86H** を、**I/O インタフェース** と判定された場合 **30H** を送信します。

- **UART** と判定された場合

ボーレートの設定が可能かどうかを判定します。設定が可能と判定した場合、**SC0BRCCR** の値を書き替え、**86H** を送信し、設定が不可能と判定した場合、動作を停止するため何も送信しません。コントローラは、1 バイト目のデータの送信が終了した後、タイムアウト時間 (**5 秒**) を設けます。タイムアウト時間内に、データ (**86H**) を正常受信できなければ、通信不能と判断してください。受信を許可 ($SC0MOD0<RXE>=1$) するタイミングは、送信バッファにデータ (**86H**) を書き込む前に行っています。

- **I/O インタフェース** と判定された場合

I/O インタフェース の設定になるように **SC0MOD0**、**SC0CR** の値を書き替え、**SC0BUF** に **30H** を書き込み、**SCLK0** クロックを待ちます。コントローラは、1 バイト目のデータ送信が終了した後、アイドル時間 (数 m 秒) 後、**SCLK** クロックを出力してください。このときのボーレートは、所望のボーレート $\div 16$ で行

い、受信データが 30H なら、通信可能と判断してください。3 バイト目からは所望のボーレートで行ってください。受信を許可 (SCOMOD0<RXE>= 1) するタイミングは、送信バッファにデータ (30H) を書き込む前に行っています。

- 3 バイト目の受信データは、動作コマンドデータになります。この場合は、RAM 転送コマンドデータ (10H) になります。
- 4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。最初に、3 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。) なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、3 バイト目の受信データが、表 18-4 の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、10H をエコーバック送信して RAM 転送処理ルーチンに分岐します。このルーチンに分岐後、パスワードエリアのデータをチェックします。パスワードエリアのデータのチェック方法は、後述の「パスワードについて」を参照してください。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。)

- 5 バイト目 ~ 16 バイト目の受信データは、パスワードデータ (12 バイト) になります。5 バイト目の受信データから順に、フラッシュメモリの以下の表のアドレスと照合します。一致していない場合、パスワードエラーフラグをセットします。

製品	パスワード領域
TMPM330FYWFG	0x3F87_FF04 - 0x3F87_FF0F

- 17 バイト目の受信データは、CHECK SUM データになります。5 バイト目から 16 バイト目の送信データを符号なしの 8 ビット加算 (オーバフローを無視) して得られた下位 8 ビット値の 2 の補数をコントローラから送信してください。CHECK SUM データの計算方法は、後述の「CHECK SUM の計算方法」を参照してください。
- 18 バイト目の送信データは、5 バイト目 ~ 17 バイト目のデータに対する ACK 応答データ (CHECK SUM 値に対する ACK 応答) になります。最初に、5 バイト目 ~ 17 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 18H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので、「1」になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、17 バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、5 バイト目 ~ 16 バイト目までの受信データを符号なし 8 ビット加算 (オーバフローを無視) して得られた値の下位 8 ビットが、00H かどうかをチェックしています。00H 以外の場合、CHECK SUM エラーの ACK 応答データ (bit0) 11H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

最後に、パスワードの照合結果をチェックします。次の場合、パスワードエラーの ACK 応答データ (bit 0) 11H を送信して、次の動作コマンド (3 バイト目) データ待ち

になります。

- 5バイト目～16バイト目のパスワードデータの照合結果に関わらず、パスワードエリアの12バイトのデータが、FFH以外の同一データの場合。
- 5バイト目～16バイト目のパスワードデータの照合がすべて一致しない場合。

上記のチェックを終えて、すべて正常なら、正常 ACK 応答データ 10H を送信します。

8. 19バイト目～22バイト目までの受信データは、ブロック転送における格納先のRAMの開始アドレスを表します。19バイト目がアドレスの31ビット～24ビットに対応し、22バイト目が7ビット～0ビットに対応します。格納アドレスの開始アドレスは偶数アドレスにしてください。
9. 23バイト目、24バイト目の受信データは、ブロック転送するバイト数を表します。23バイト目が転送バイト数の15ビット～8ビット目に対応し、24バイト目が7ビット～0ビット目に対応します。
10. 25バイト目の受信データは、CHECK SUM データになります。19バイト目から24バイト目の送信データを符号なし8ビット加算（オーバーフローを無視）して得られた下位8ビット値の2の補数値をコントローラから送信してください。CHECK SUM データ計算方法は、後述の「CHECK SUM の計算方法」を参照してください。
11. 26バイト目の送信データは、19バイト目～25バイト目のデータに対するACK 応答データ（CHECK SUM 値に対するACK 応答）になります。最初に、19バイト目～25バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常のACK 応答データ（bit 3）18H を送信して、次の動作コマンド（3バイト目）データ待ちになります。送信データの上位4ビットは、直前の動作コマンドデータの上位4ビットになるので“1”になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、25バイト目のCHECK SUM データをチェックします。CHECK SUM データのチェック方法は、19バイト目～24バイト目までの受信データを符号なし8ビット加算（オーバーフローを無視）して得られた値の下位8ビットが、00H かどうかをチェックしています。00H 以外の場合、CHECK SUM エラーのACK 応答データ（bit 0）11H を送信して、次の動作コマンド（3バイト目）データ待ちになります。

- 19バイト目～25バイト目のデータはRAM 上のアドレス0x2000_0400 からRAM の最終番地の領域に納まるようにプログラムしてください。

上記のチェックを終えてすべて正常なら、正常 ACK 応答データ 10H を送信します。

12. 27バイト目～mバイト目の受信データは、RAM へ格納するデータになります。RAM に格納するデータを、19バイト目から22バイト目で指定されたアドレスから書き込み、23バイト目から24バイト目に指定されたバイト数分だけ書き込みます。

13. $m+1$ バイト目の受信データは、CHECK SUM データになります。27 バイト目～ m バイト目の送信データを符号なし 8 ビット加算（オーバフローを無視）して得られた下位 8 ビット値の 2 の補数をコントローラから送信してください。CHECK SUM データの計算方法は、後述の「CHECK SUM の計算方法」を参照してください。
14. $m+2$ バイト目の送信データは、27 バイト目～ $m+1$ バイト目のデータに対する ACK 応答データ（CHECK SUM に対する ACK 応答）になります。最初に 27 バイト目～ $m+1$ バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 18H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので“1”になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、 $m+1$ バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、27 バイト目～ m バイト目までの受信データを符号なし 8 ビット加算（オーバフローを無視）して得られた値の下位 8 ビットが、00H かどうかをチェックしています。00H 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 11H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。上記のチェックを終えてすべて正常なら、正常 ACK 応答データ 10H を送信します。

15. $m+2$ バイト目の ACK 応答データが正常 ACK 応答データの場合、正常 ACK 応答データ 10H を送信後、19 バイト目～22 バイト目で指定されたアドレスに分岐します。

2) フラッシュメモリ SUM コマンド (表 18-7参照)

1. 1 バイト目 ~ 2 バイト目までの送受信データは RAM 転送コマンドの場合と同一になります。
2. 3 バイト目の受信データは動作コマンドデータになります。この場合は、フラッシュメモリ SUM コマンドデータ (20 H) になります。
3. 4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。最初に、3 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。)なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、3 バイト目の受信データが、表 18-4 の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、20H をエコーバック送信して、フラッシュメモリ SUM 処理ルーチンに分岐します。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データ上位 4 ビットは不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。)

4. 5 バイト目の送信データは SUM 値の上位データ、6 バイト目の送信データは SUM 値の下位データになります。SUM の計算方法は、後述の「SUM の計算方法」を参照してください。
5. 7 バイト目の送信データは、CHECK SUM データになります。5 バイト目から 6 バイト目の送信データを符号なし 8 ビット加算 (オーバーフローを無視) を行い、得られた下位 8 ビット値の 2 の補数を送信します。
6. 8 バイト目の受信データは、次の動作コマンドデータになります。

3) 製品情報読み出しコマンド (表 18-8参照)

1. 1 バイト目 ~ 2 バイト目までの送受信データは RAM 転送コマンドの場合と同一になります。
2. 3 バイト目の受信データは、動作コマンドデータになります。この場合は、製品情報読み出しコマンドデータ (30H) になります。
3. 4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。最初に、3 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。) なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、3 バイト目の受信データが、表 18-4の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、30H をエコーバック送信して、製品情報読み出し処理ルーチンに分岐します。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。)

4. 5 バイト目 ~ 8 バイト目の送信データは、フラッシュメモリのデータ (下記の表のアドレスのデータ) になります。この番地にソフトなどの ID 情報を書き込んでおくことにより、書き込んだソフトのバージョン管理をすることができます。

製品	対象エリア
TMPM330FYWFG	0x3F87_FF00 - 0x3F87_FF03

5. 9 バイト目 ~ 20 バイト目の送信データは製品名になります。9 バイト目から、アスキーコードで、以下の表のとおり送信します。以下の表で、[] はスペースを示します。

製品	送信コード
TMPM330FYWFG	T, M, P, M, 3, 3, 0, F, D, _, [], _

6. 21 バイト目 ~ 24 バイト目の送信データはパスワード比較開始アドレスになります。21 バイト目から以下の値を送信します。

製品	パスワード比較開始アドレス
TMPM330FYWFG	04H, FFH, 87H, 3FH

7. 25 バイト目 ~ 28 バイト目の送信データは RAM 開始アドレスになります。25 バイト目から、00H、00H、00H、20H を送信します。
8. 29 バイト目 ~ 32 バイト目の送信データはダミーデータになります。29 バイト目から、00H、00H、00H、00H を送信します。
9. 33 バイト目 ~ 36 バイト目の送信データは RAM 終了アドレスになります。33 バイト目から以下の値を送信します。

製品	RAM 終了アドレス
TMPM330FYWFG	FFH、7FH、00H、20H

(注) TMPM330FYWFG の RAM 終了アドレスは、実際には 0x2000_3FFF です。

10. 37 バイト目 ~ 40 バイト目の送信データは、00H、00H、00H、00H になります。
41 バイト目 ~ 44 バイト目の送信データは、00H、00H、00H、00H になります。
11. 45 バイト目 ~ 46 バイト目の送信データは、00H、00H になります。
12. 47 バイト目 ~ 50 バイト目の送信データは、フラッシュメモリ開始アドレスになります。47 バイト目から、00H、00H、80H、3FH を送信します。
13. 51 バイト目 ~ 54 バイト目の送信データは、フラッシュメモリ終了アドレスになります。51 バイト目から以下の値を送信します。

製品	フラッシュメモリ終了アドレス
TMPM330FYWFG	FFH、FFH、87H、3FH

(注) TMPM330FYWFG のフラッシュメモリ終了アドレスは、0x3F83_FFFF です。256KB の FLASH 領域とは別に、製品情報読み出しコマンド用のデータ領域およびパスワード領域が 128 ワード(1 ページ)存在します。(0x3F87_FE00 - 0x3F87_FFFF)

14. 55 バイト目 ~ 56 バイト目の送信データは、フラッシュメモリのブロック分割数になります。55 バイト目から以下の値を送信します。

製品	ブロック分割数
TMPM330FYWFG	06H、00H

15. 57 バイト目 ~ 83 バイト目の送信データは、フラッシュメモリのブロック情報になります。ブロック情報は、フラッシュメモリ開始アドレスから見たとき、同一ブロックサイズが何ブロック続いているかを一単位とし、同一ブロックサイズの先頭の開始アドレスとブロックサイズ (ハーフワード単位) およびブロックの個数で表します。

57 バイト目 ~ 65 バイト目の送信データは、16k バイトのブロックを表します。66 バイト目 ~ 74 バイト目の送信データは、32k バイトのブロックを表します。75 バイト目 ~ 83 バイト目の送信データは、64k バイトのブロックを表します。84 バイト目 ~ 92 バイト目の送信データは、128k バイトのブロックを表します。

送信データについては、表 18-8を参照してください。

16. 93 バイト目の送信データは、CHECK SUM データになります。5 バイト目から 92 バイト目の送信データを符号なし 8 ビット加算 (オーバフローを無視) して、得られた下位 8 ビット値の 2 の補数を送信します。

17. 94 バイト目の受信データは、次の動作コマンドデータになります。

4) フラッシュメモリチップ消去 及び プロテクトビット消去コマンド(表 18-9 参照)

1. 1 バイト目 ~ 2 バイト目までの送受信データは RAM 転送コマンドの場合と同一になります。
2. コントローラ → デバイス
3 バイト目の受信データは動作コマンドデータになります。この場合は、フラッシュメモリチップ消去コマンドデータ (40 H) になります。
3. デバイス → コントローラ
4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。
最初に、3 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。
次に、3 バイト目の受信データが、表 18-4 の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、40H をエコーバック送信します。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。
4. コントローラ → デバイス
5 バイト目の受信データは消去イネーブルコマンドデータ (54 H) になります。
5. デバイス → コントローラ
6 バイト目の送信データは、5 バイト目の消去イネーブルコマンドデータに対する ACK 応答データになります。
最初に、5 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) x8H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。
次に、5 バイト目の受信データが、消去イネーブルコマンドデータに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、54H をエコーバック送信して、フラッシュメモリチップ消去処理ルーチンに分岐します。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) x1H を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります(直前の動作コマンドデータの上位 4 ビットになります)。

6. デバイス → コントローラ
7 バイト目の送信データはが正常に終了したかどうかを示します。
正常に終了した時は、終了コード(4FH)を返します。
消去Error が起きた場合は、エラーコード(4CH)を返します。

7. 9 バイト目の受信データは、次の動作コマンドデータになります。

5) ACK 応答データ

ブートプログラムは処理状況を各種コードによってコントローラに送信します。表 18-10から表 18-13に各受信データに対する ACK 応答データを示します。ACK 応答データの上位 4 ビットは、動作コマンドデータの上位 4 ビットになります。また 3 ビット目は受信エラーを表し、0 ビット目は動作コマンドエラー、CHECK SUM エラー、パスワードエラーの状態を表します。1 ビット目と 2 ビット目は常に 0 になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

表 18-10 シリアル動作判定データに対する ACK 応答データ

送信データ	送信データの意味
0x86	UART での通信が可能と判定した。(注 1)
0x30	I/O インタフェースでの通信が可能と判定した。

(注 1): UART の場合、ボーレートの設定が不可能と判定したら、何も送信しないで動作を停止します。

表 18-11 動作コマンドデータに対する ACK 応答データ

送信データ	送信データの意味
0x78 (注 1)	動作コマンドデータに受信エラーが発生した。
0x71 (注 1)	未定義の動作コマンドデータを正常受信した。
0x10	RAM 転送コマンドと判定した。
0x20	フラッシュメモリ SUM コマンドと判定した。
0x30	製品情報読み出しコマンドと判定した。
0x40	フラッシュメモリチップ消去コマンドと判定した。

(注 1) : 上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになります。

表 18-12 CHECK SUM データに対する ACK 応答データ

送信データ	送信データの意味
0xN8 (注 1)	受信エラーが発生していた。
0xN1 (注 1)	CHECK SUM エラーが発生した。あるいは、パスワードエラーが発生した。
0xN0 (注 1)	CHECK SUM 値は正常な値と判定した。

(注 1) : 上位 4 ビットは動作コマンドデータの上位 4 ビットになります。例えば、パスワードエラー発生時は 1(N=RAM 転送コマンドデータ[7:4])となります。

表 18-13 フラッシュメモリチップ消去 及び プロテクトビット消去動作に対する ACK 対応データ

送信データ	送信データの意味
54H	消去イネーブルコマンドと判定した。
4FH	消去コマンド終了
4CH	消去コマンドが不正に終了した。

6) シリアル動作モード判定

コントローラは、UART で通信したい場合、所望のボーレートで1バイト目を 86H にし、I/O インタフェースで通信したい場合、所望のボーレート ÷ 16 で1バイト目を 30H にして送信してください。図 18-4にそれぞれの場合の波形を示します。

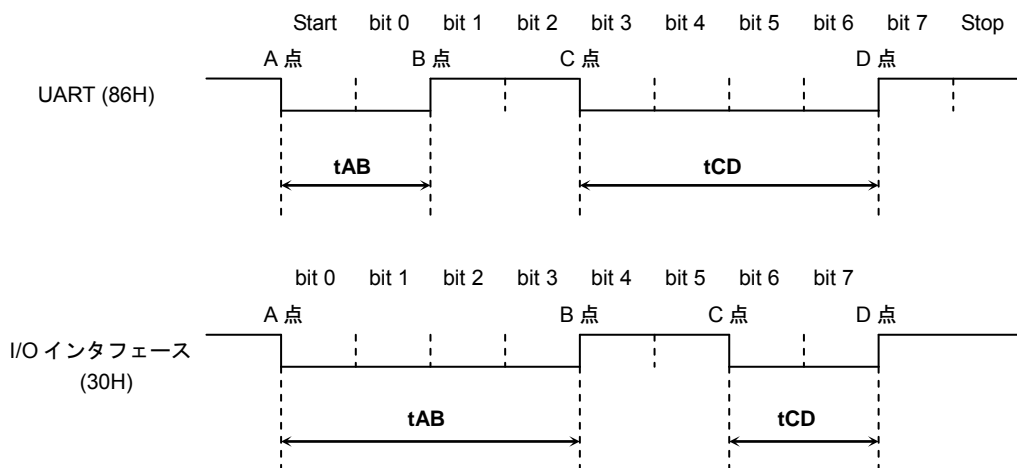


図 18-4 シリアル動作モード判定データ

ブートプログラムは、リセット解除後の1バイト目のシリアル動作モード判定データ (86H、30H) を受信禁止状態にして、図 18-5に示すフローチャートで、図 18-4の t_{AB} 、 t_{AC} と、 t_{AD} の時間を求めています。図 18-5のフローチャートに示すように、CPU が受信端子のレベルをモニタしてレベルの変化があると、そのときのタイマ値を取り込みます。このため、 t_{AB} 、 t_{AC} と、 t_{AD} のタイマ値には誤差が生じます。また、ボーレートが速いときには、CPU は受信端子のレベルの変化を判断できない場合がありますので注意してください。特に、I/O インタフェースは UART に比べボーレートが速いため、このような場合が発生しやすくなります。このようなことが起こらないようにするために、I/O インタフェースの場合、コントローラのボーレートは所望ボーレート ÷ 16 にして送信してください。

図 18-5のフローチャートに示すように、シリアル動作モードの判定は、受信端子が“L”レベルのときの時間幅の大小関係で判定しています。 $t_{AB} \leq t_{CD}$ の場合 UART と判定し、ボーレートの自動設定が可能かどうかを t_{AD} の時間から判定します。 $t_{AB} > t_{CD}$ の場合、I/O インタフェースと判定します。なお、先に述べたように、 t_{AB} 、 t_{AC} 、 t_{AD} のタイマ値には誤差が生じているため、ボーレートが速く、動作周波数が低い場合、各タイマ値が小さくなり、意図しない判断を行うことがありますので注意してください(書き換えルーチン内で UART の再設定を行ってください)。

例えば、コントローラは UART で通信したいのに、I/O インタフェースと判定してしまうことがあります。このようなことを考慮して、コントローラは UART で通信したい場合、1バイト目のデータを送信後、タイムアウト時間内にデータ 86H を正常受信できなければ通信不可能と判断してください。I/O インタフェースで通信したい場合は1バイト目のデータを送信後、アイドル時間後に SCLK クロックを出力してデータを受信し、受信データが 30H でなければ通信不可能と判断してください。

I/O インタフェースで通信したい場合は上記のとおり、 $t_{AB} > t_{CD}$ であれば1バイト目のデータは 0x30 でなくても構いません。A点とC点の立ち下がり、B点とD点の立ち上がりを判定できるように 0x91、0xA1 あるいは 0xB1 を1バイト目のデータとして送信できます。 $t_{AB} > t_{CD}$ が成立しており、動作モード判定結果 SIO が選択された場合、(1バイト目の送信データが 0x30 でない場合でも)2バイト目のデータは 0x30 となります(以下、I/O インタフェース判定用の1バイト目のデータは 0x30 を表記しています)。

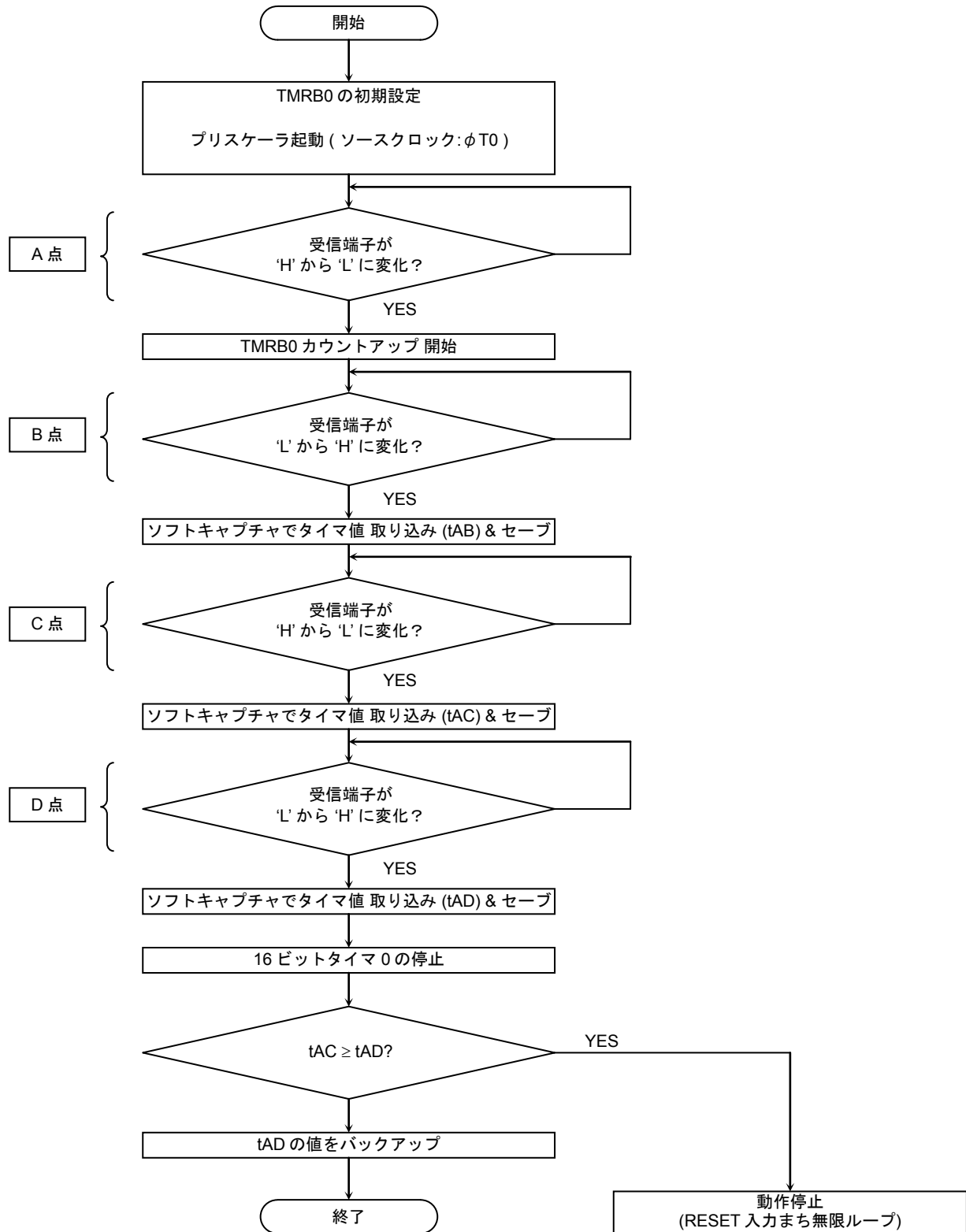


図 18-5 シリアル動作モード受信フローチャート

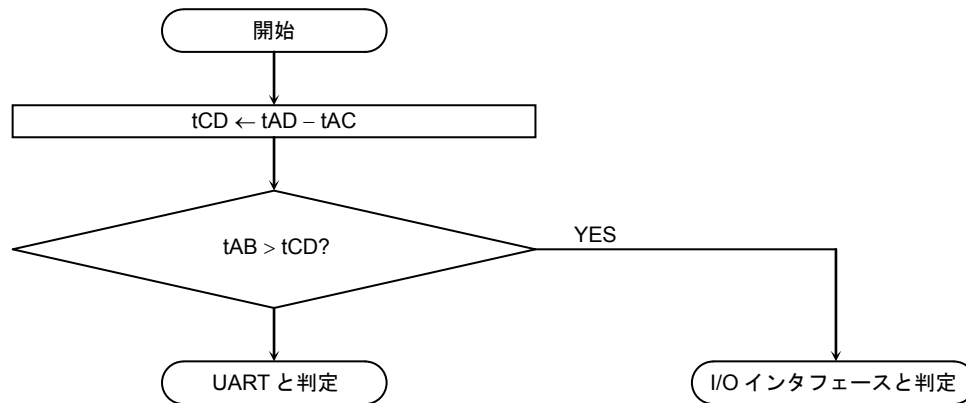


図 18-6 シリアル動作モード判定フローチャート

7) パスワードについて

動作コマンドデータが RAM 転送コマンド (10H) の場合、パスワードのチェックを行います。まず、動作コマンドデータをエコーバック送信 (10H) 後、パスワードエリアのデータ (12 バイト) をチェックします。パスワードエリアは以下のとおりです。

製品	パスワード領域
TMPM330FYWFG	0x3F87_FF04 - 0x3F87_FF0F

図 18-7に示すようにパスワードエリアのデータが、FFH 以外の同一データになっていた場合、パスワードエリアエラーと判定します。パスワードエリアエラーと判定された場合、パスワードデータの照合結果に関わらず、17 バイト目の CHECK SUM 値に対する ACK 対応は 11H を送信します。

次に、5 バイト目～16 バイト目の受信データ (パスワードデータ) の照合を行います。12 バイト分すべてが一致しないと、パスワードエラーになります。パスワードエラーと判定された場合、17 バイト目の CHECK SUM 値に対する ACK 応答は、パスワードエラーとなります。

セキュリティ機能が有効な状態でもパスワードの参照は行います。

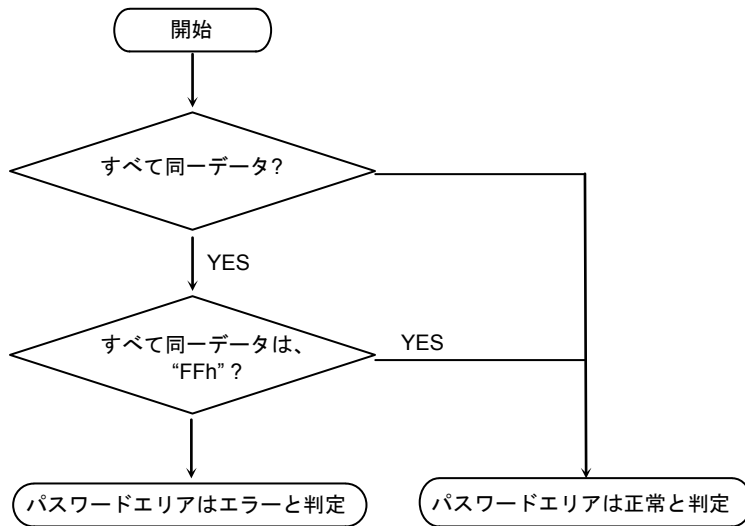


図 18-7 パスワードエリアチェックフローチャート

8) SUM の計算方法

SUM の計算方法は、バイト+バイト+バイト+.....+バイトの結果をハーフワードで返します。つまり、バイトでデータを読み出して符号なし 8 ビット加算を行い、計算結果をハーフワードで求めています。コントローラへは、SUM の上位 8 ビットデータ、下位 8 ビットデータの順番で送信します。SUM の計算対象のデータは、フラッシュメモリ全エリア (512k バイト) のデータになります。フラッシュメモリ SUM コマンドを実行したときに返される SUM は、本計算方法を使用しています。

(注) TMPM330FYWFG の場合、256KB の領域からパスワード領域までの間とパスワード領域以降 0x0007_FFFF までの間のデータは“0xFF”として計算されます。

例)

A1H
B2H
C3H
D4H

左記 4 バイトが計算対象データの場合、SUM の値は、
 $A1H + B2H + C3H + D4H = 02EAH$
 となるので、
 SUM の上位のデータは、02H、
 SUM の下位のデータは、EAH になります。
 したがって、コントローラには 02H、EAH の順番で送信します。

9) CHECK SUM の計算方法

CHECK SUM の計算方法は、送信データを符号なし 8 ビット加算 (オーバフローを無視) して得られた下位 8 ビット値の 2 の補数値を求めています。フラッシュメモリ SUM コマンド、製品情報読み出しコマンドを実行したときに返される CHECK SUM は、本計算方法を使用しています。また、コントローラは CHECK SUM 値を送信するときは、本計算方法を使用してください。

例) フラッシュメモリ SUM コマンドのときを例に説明します。

SUM の上位 8 ビットデータが E5H、下位 8 ビットデータが F6H の場合の CHECK SUM 値を求めます。まず、符号なし 8 ビット加算して得られた値を求めます。

$$E5H + F6H = 1DBH$$

この値の下位 8 ビットに対しての 2 の補数をとると以下のようになり、この値が CHECK SUM 値になります。したがって、コントローラには 25H を送信します。

$$0 - DBH = 25H$$

(7) ブートプログラム全体フローチャート

ブートプログラム全体フローチャートを示します。

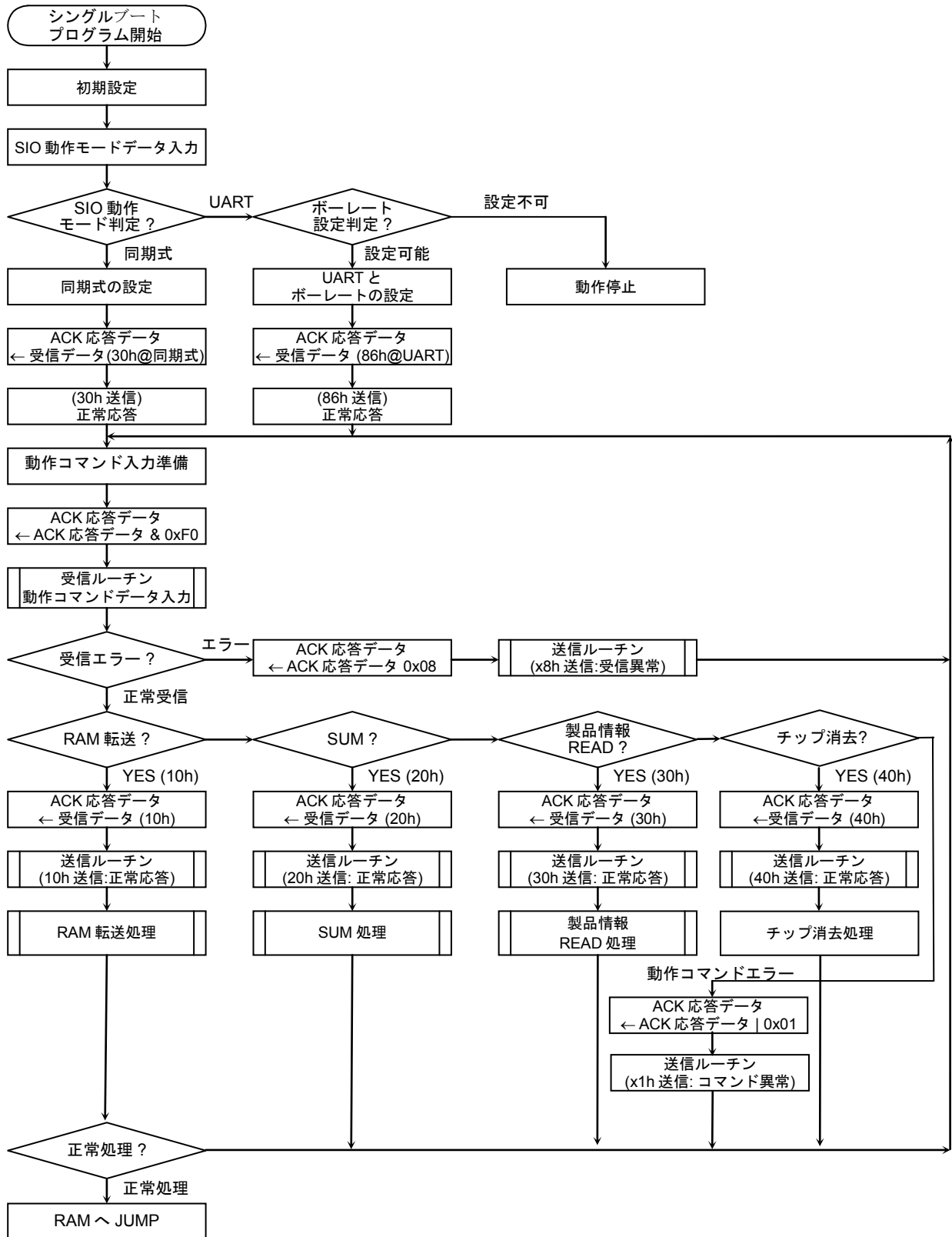


図 18-8 ブートプログラム全体フローチャート

18.3 オンボードプログラミングでのフラッシュメモリ書き込み/消去

オンボードプログラミングでは、CPU によりソフトウェア的にコマンドを実行することで、フラッシュの書き込み/消去を行います。この書き込み/消去の制御プログラムはユーザーがあらかじめ用意しておきます。フラッシュメモリの書き込み/消去を行っている間は、フラッシュメモリ自身の読み出しはできませんので、ユーザーブートモードに移行後、書き込み/消去制御プログラムは内蔵 RAM 上で実行してください。

18.3.1 フラッシュメモリ

一部の機能を除き、フラッシュメモリの書き込みおよび消去などは JEDEC 標準コマンドに準拠しています。書き込み、消去を行う場合、CPU の 32 ビット(ワード)のデータ転送命令を用いてフラッシュメモリへコマンドを入力します。コマンド入力後、書き込みおよび消去は内部で自動的に行われます。

表 18-14 フラッシュメモリの機能

主な機能	説明
自動ページプログラム	ページ単位でデータ書き込みを自動で行います。
自動チップ消去	フラッシュメモリ全エリアの一括消去を自動で行います。
自動ブロック消去	ブロック単位での消去を自動で行います。
ライト/消去プロテクト	ブロック単位ごとに書き込みおよび消去を禁止することができます。

CPU とのインタフェースの関係上、動作コマンドのアドレス指定が標準コマンドとは異なります。また、特に断りのない限りフラッシュメモリの書き込みは 32 ビット単位で行います。フラッシュメモリへの書き込みは、32 ビット(ワード)のデータ転送命令を用いてください。

(1) ブロック構成

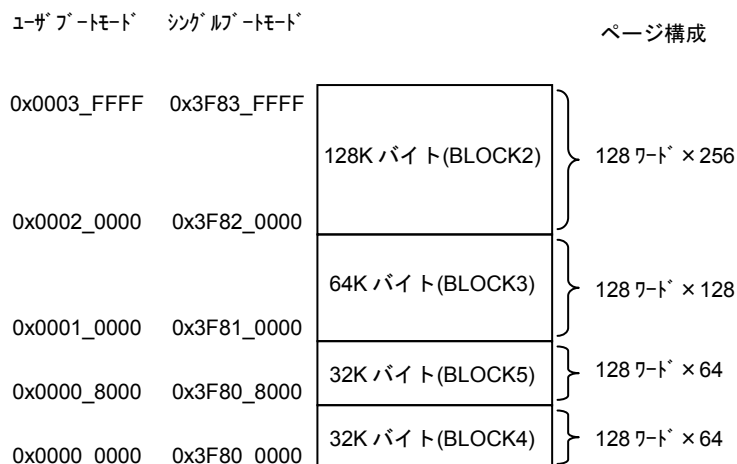


図 18-9 フラッシュメモリのブロック構成

(注) TMPM330FYWFG では 256KB の FLASH 領域とは別に、製品情報読み出しコマンド用のデータ領域およびパスワード領域が 128 ワード(1 ページ)存在します (0x3F87_FE00 - 0x3F87_FFFF)。この領域を消去する場合には自動チップ消去コマンドを使用するか、自動ブロック消去コマンドでブロック 0 を指定してください。

(2) 基本動作

このフラッシュメモリには、大きく分けて以下の2種類の動作モードがあります。

- メモリデータを読み出すモード (リードモード)
- メモリデータを自動的に消去/書き替えるモード (自動動作)

リードモード中にコマンドシーケンスを実行することで、自動動作に移ることができます。自動動作中は、フラッシュメモリデータの読み出しとフラッシュメモリ上の命令の実行ができません。自動動作中はハードウェアリセットを除いて例外が発生した場合、リードモードに移りません。自動動作中はデバックポート接続時のデバック例外とリセットを除いて、全ての例外を発生させないでください。ハードウェアリセットを除いて例外が発生した場合、リードモードに移行しません。

1) リード

データを読み出す場合、フラッシュメモリをリードモードにします。電源投入直後、CPUリセット解除後および自動動作の正常終了時に、フラッシュメモリはリードモードになります。自動動作の異常終了時や、他のモードからリードモードに復帰させるには、後述する Read/リセットコマンド(ソフトウェアリセット)もしくはハードウェアリセットを用います。フラッシュメモリに書かれた命令を実行する場合もリードモードでなければなりません。

- **Read/リセットコマンド及び Read コマンド(ソフトウェアリセット)**

ID-Read コマンドを実行した場合、マクロは自動的に Read モードに復帰せず、その状態で停止します。このような状態から Read モードに復帰させるために、Read/リセットコマンドを使用します。また、途中まで入力したコマンドをキャンセルしたい場合にも、Read/リセット コマンドを使用します。Read コマンドは、フラッシュメモリの任意のアドレスに 0x0000_00F0 データを 32 ビット (ワード) のデータ転送命令を実行してリードモードに復帰するコマンドです。

- **Read/リセットコマンドは第 3 バスライトサイクル終了後にリードモードになります。**

2) コマンドライト

このフラッシュメモリは、コマンドコントロール方式を用いています。コマンド実行は、フラッシュメモリに対してコマンドシーケンスを実行することで行います。フラッシュメモリは、入力されたアドレスとデータの組み合わせによって各自動動作コマンドを実行します(コマンドシーケンス参照)。

コマンドシーケンスの途中でコマンドライトをキャンセルしたい場合や、間違ったコマンドシーケンスを入力した場合は、Read/リセットコマンドを実行します。フラッシュメモリはコマンド実行を中止してリードモードになります。

各コマンドはいくつかのバスサイクルで構成されており、フラッシュメモリに対して 32 ビット (ワード) のデータ転送命令を実行するものを“バスライトサイクル”と呼びます。各バスライトサイクルには順番があり、フラッシュメモリはバスライトサイクルのアドレスとデータが規定の順番でコマンドライトされた時は自動動作を実施します。規定の順番でコマンドライトされなかったバスライトサイクルがあった場合にフラッシュメモリはコマンドの実行を中止してリードモードになります。

- (注意1) 各コマンドシーケンスは、フラッシュメモリ外のエリアから実施します。
- (注意2) 各バスライトサイクルは連続して、32ビット(ワード)のデータ転送命令で行って下さい。各コマンドシーケンスの実行中に、フラッシュメモリへのアクセスは実施しないで下さい。また、全ての割り込み(デバックポート接続時は、デバック例外を除く)を発生させないでください。
- 各バスライトサイクル及び、各コマンドシーケンスの実行中にフラッシュメモリに対して予期せぬリードアクセスが生じることになり、コマンドシーケンサがコマンドを正常に認識できない恐れがあります。各コマンドシーケンスは正常終了しない恐れがあると同時に、誤ったコマンドライトとして認識してしまう可能性が有ります。
- (注意3) コマンドシーケンサがコマンドを認識するために、コマンド開始前の状態がリードモードである必要があります。各コマンドシーケンスの第1バスライトサイクル前にFCFLCS<RDY/BSY>=1であることを確認してください。続いてReadコマンドを実行することを推奨します。
- (注意4) コマンド発行時、誤ったアドレスやデータをライトした場合は、必ずソフトウェアリセットを発行して、一度リードモードに戻して下さい。

(3) リセット

ハードウェアリセット

ハードウェアリセットは、自動プログラム/消去動作の強制的な実行中止や、自動動作が異常終了したとき、コマンドライトにより設定した動作モードの解除に使用します。

本フラッシュメモリはメモリブロックとしてリセット入力を持ち、この入力は CPU のリセット信号とつながっています。このため、本デバイスの RESET 入力端子が VIL となるか、ウォッチドッグタイマのオーバフローなどにより CPU のリセットがかかると、フラッシュメモリは自動動作の実行中であってもその動作を中止し、リードモードに戻ります。なお、自動動作の実行中にハードウェアリセットが入った場合は、データの書き替えが正常に行えませんので注意が必要です。再度、書き替えを行う処置をしてください。

CPU のリセット動作については、「18.2.1 リセット動作」を参照してください。所定のリセット入力後、CPU はフラッシュメモリよりリセットベクタデータをリードし、リセット解除後の動作を開始します。

(4) コマンド説明

1) 自動ページプログラム

フラッシュメモリへの書き込みは、“1” データセルを“0” データにすることです。“0” データセルを“1” データにすることはできません。“0” データセルを“1” データにするには消去動作を行う必要があります。

本デバイスの自動ページプログラムは、ページごとの書き込みとなります。1 ページは 128 ワードです。アドレス[31:9]が同じで、先頭アドレス[8:0] = 0、最後のアドレス[8:0] = 0x1FF のグループです。以降はページプログラムの単位をページと呼びます。

データセルへの書き込みは、内部シーケンサで自動的に行われ、CPU による外部からの制御を必要としません。自動ページプログラムの状態(書き込み動作中であるか)はレジスタ FCFLCS<RDY/BSY>にて確認できます。

また、自動ページプログラム中は、新たにコマンドシーケンスを受け付けません。自動ページプログラム動作を中止したい場合は、ハードウェアリセットを用います。これにより動作を中止させた場合、該当のページに対するデータの書き込みは正常に行われていないため、消去動作後に改めて自動ページプログラムを実行する必要があります。

自動ページプログラムは消去後のページに対して 1 回のみ可能で、“1” データセルであっても“0” データセルであってもページに対して 2 回以上の実行はできません。一度書き込み動作を行ったページに対して再度書き込みを行う場合は、自動ブロック消去または自動チップ消去コマンドを行った後に自動ページプログラムを実行しなおす必要がありますのでご注意ください。消去動作を伴わない同一ページへの 2 回以上ページプログラム実施はデバイス破損の可能性が有ります。

本デバイス内部で自動的なベリファイ動作は行いません。正常に書き込みができたか、実行後に読み出しをして確認してください。

自動ページプログラムは、コマンドサイクルの第3バスライトサイクル終了から開始します。第5バスライトサイクル以降は、第4バスライトサイクルで指定した次のアドレス（第4バスライトサイクルではページの先頭アドレスをコマンドライトします）から順番に書き込みを行います（データ入力は32ビット単位で行います）。第4バスライトサイクル以降のコマンドライトは必ず32ビット（ワード）のデータ転送命令を使用してください。このとき32ビット（ワード）のデータ転送命令はワード境界をまたいだ位置へ実施しないでください。第5バスライトサイクル以降は同一ページエリアに対してデータをコマンドライトします。また、ページの一部に書き込みをしたい場合でもページ単位で自動ページプログラムする必要があります。この場合も第4バスライトサイクルのアドレス入力はページ先頭アドレスにしてください。この時”0”データセルにしたくない箇所は入力データを”1”にしてコマンドライトします。例えば、あるページ先頭アドレスの書き込みをしない場合、第4バスライトサイクルのデータ入力を0xFFFFFFFFとしてコマンドライトします。

第3バスライトサイクルを実行すると自動プログラム動作中となります。このことはレジスタFCFLCS<RDY/BSY>（表18-15）をモニタすることで確認できます。自動プログラム動作中は、新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。動作を中止させた場合は、データの書き込みは正常に行きませんので注意してください。1ページのデータをコマンドライト後、ページ自動書き込みが正常終了した時にFCFLCS<RDY/BSY> = ”1”となり、リードモードに復帰します。

複数のページに対してデータの書き込みを行うときは、ページ毎にページプログラムコマンドを実行する必要があります（1回の自動ページプログラムコマンドで書き込めるサイズは1ページです）。ページを跨ったデータ入力の自動ページプログラムは出来ません。

ライト/消去プロテクトされたブロックへの書き込みはできません。自動プログラムが正常終了すると、自動的にリードモードに復帰します。このことはFCFLCS<RDY/BSY>（表18-15）をモニタすることで確認できます。自動プログラム動作が不良となった場合は、フラッシュメモリはこのモードのままロックされ、リードモードには復帰しません。リード状態に復帰させるにはハードウェアリセットでデバイスをリセットする必要があります。この場合、このアドレスへの書き込みが不良になっていますので、デバイスの使用を停止するか、以後このアドレスを含むブロックを使用しないことを推奨します。

注：自動ページプログラム第4バスライトサイクル以降のバスライトサイクルでは、ソフトウェアリセットが無効になります。

2) 自動チップ消去

自動チップ消去動作は、コマンドサイクルの第6バスライトサイクル終了から開始します。

FCFLCS<RDY/BSY>(表 18-15)をモニタすることで確認できます。本デバイス内部で自動的なベリファイ動作は行いませんので、正常に消去ができたか、実行後に読み出しをして確認してください。自動チップ消去動作中は、新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。動作を中止した場合、データの消去は正常に行えないので、再度自動チップ消去を行う必要があります。

また、ライト/消去プロテクトされているブロックがある場合、そのブロックの消去は行いません。すべてのブロックがライト/消去プロテクトされている場合は、自動チップ消去を実行せず、コマンドシーケンスの第6バスライトサイクルの完了後にリードモードになります。自動チップ消去が正常終了すると、自動的にリードモードに復帰します。自動チップ消去動作が不良となった場合は、フラッシュメモリはこのモードのままロックされ、リードモードには復帰しません。

リードモードに復帰させるにはハードウェアリセットでデバイスをリセットする必要があります。この場合は、不良が発生したブロックの検知はできません。デバイスの使用を停止するか、ブロック消去機能を利用して不良ブロックを特定し、以降不良ブロックを使用しないことを推奨します。

3) 自動ブロック消去(Block 単位)

自動ブロック消去は、コマンドサイクルの第6バスライトサイクル終了から開始します。

自動ブロック消去動作の状態は、FCFLCS<RDY/BSY>(表 18-15)をモニタすることで確認できます。本デバイス内部で自動的なベリファイ動作は行いませんので、正常に消去ができたか、実行後に読み出しをして確認してください。自動ブロック消去中は新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。この場合、データの消去は正常に行えませんが、再度自動ブロック消去を行う必要があります。

また、ライト/消去プロテクトされているブロックがある場合、そのブロックの消去を行いません。自動ブロック消去動作が不良になった場合、フラッシュメモリはこのモードのままロックされ、リードモードには復帰しません。ハードウェアリセットを用いてデバイスをリセットしてください。

4) 自動プロテクトビットプログラム(Block 単位)

本デバイスはプロテクトビットを内蔵し Block 単位で設定することが出来ます。Block とプロテクトビットの関係は表 18-20にあります。自動プロテクトビットプログラムは1ビット単位で実行します。ビットの指定は第7バスライトサイクルの PBA で行います。自動プロテクトビットプログラムにより、ブロックごとに書き込みと消去の動作を禁止(プロテクト)することができます。各ブロックのプロテクトの状態は FLCS レジスタの<BLPRO>ビットで確認できます。自動プロテクトビットプログラム動作の状態は、FCFLCS<RDY/BSY>(表 18-15)をモニタすることで確認できます。自動プロテクトビットプログラム動作中は新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。この場合、プロテクトの設定は正常に行えない可能性がありますので、再度ブロックプロテクトの動作をやり直す必要があります。全てのプロテクトビットをプログラムした後は、FLCS レジスタの全ての<BLPRO>ビットが“1”になっています。これ以降は全ての Block に対し、ライト/消去は出来ません。

注：自動プロテクトビットプログラム第7バスライトサイクルでは、ソフトウェアリセットが無効になります。FCFLCS<RDY/BSY> は、第7バスライトサイクル入力後から、FCFLCS<RDY/BSY> = “0”となります。

5) 自動プロテクトビット消去

セキュリティビットとプロテクトビットの状態によって、自動プロテクトビット消去コマンドの実行結果が異なります。FCSECBIT<SECBIT>=0x1 の場合、FCFLCS レジスタの全ての<BLPRO>が“1”か、それ以外の値かで動作が決まります。自動プロテクトビット消去コマンド実施前に必ず FCFLCS<BLPRO>の値を確認してください。セキュリティの詳細については 17 章プロテクト/セキュリティ機能を参照してください。

(注) TMPM330FYWFG はブロック 2～5 で構成されていますが、セキュリティ機能を使用する場合ブロック 0、1 もプロテクトビットをプログラムしてください。

・ FCFLCS<BLPRO> = all “1” (全てのプロテクトビットがプログラムされている) の場合

自動プロテクトビット消去コマンドをコマンドライトすると、自動的に本デバイス内部でフラッシュメモリの初期化を行います。第 7 バスライトサイクル終了後、フラッシュメモリ全エリアのデータセルの消去を行い、引き続きプロテクトビットの消去を行います。この動作に関しては FCFLCS<RDY/BSY>をモニタすることで確認できます。自動プロテクトビット消去動作が正常に終了した場合 FCFLCS=0x00000001 となります。本デバイス内部で自動的にベリファイ動作は行いませんので、正常に消去ができたか、実行後に読み出しをして確認してください。第 7 バスサイクル以降の自動動作中にリードモードに復帰させるにはハードウェアリセットでデバイスをリセットする必要があります。この場合リードモードに復帰後、FCFLCS<BLPRO>にてプロテクトビットの状態を確認して、必要に応じて再度自動プロテクトビット消去または、自動チップ消去或いは自動ブロック消去を実行する必要があります。

・ FCFLCS<BLPRO> ≠ all “1” (全てのプロテクトビットがプログラムされていない) の場合

自動プロテクトビット消去により、プロテクトの状態を解除することができます。本デバイスでは、表 18-21 のように Block 単位のプロテクトビットに対して消去は Block をまとめて行います。消去を行いたいプロテクトビットの指定は第 7 バスライトサイクルで行い、コマンド終了後は、Block をまとめて消去します。各ブロックのプロテクトの状態は後で説明する FCFLCS<BLPRO>で確認できます。自動プロテクトビットプログラム動作の状態は、FCFLCS<RDY/BSY>をモニタすることで確認できます。自動プロテクトビット消去動作が正常に終了した場合、FCFLCS<BLPRO>の消去選択したプロテクトビットの値が“0”となります。

いずれの場合も、自動プロテクトビット消去動作中は新たなコマンドシーケンスを受け付けません。動作を中止する場合はハードウェアリセットを用います。自動プロテクトビット消去動作が正常に終了した場合はリードモードに復帰します。

FCFLCS<RDY/BSY> ビットは自動動作中“0”、自動動作終了後“1”になります。

6) ID-Read

ID-Read コマンドを使用すると、本デバイスに内蔵しているフラッシュメモリのタイプ等の情報を知ることが出来ます。第 4 バスライトサイクル以降でのアドレス[15:14] の値によりロードされるデータが異なります(データ入力値は 0x00 推奨)。第 5 バスライトサイクル以降で任意のフラッシュメモリエリアの読み出しを行うと **ID** の値が読み出されます。**ID-Read** コマンド第 4 バスライトサイクル以降は自動的にリードモードに復帰しません。第 4 バスライトサイクルと **ID** の値の読み出しは繰り返し実行できます。リードモードへの復帰は **Read**/リセットコマンドまたはハードウェアリセットで行います。

(5) フラッシュコントロール/ステータスレジスタ

フラッシュメモリのステータスマニタと、ブロック単位のプロテクト状態を示すレジスタです。

表 18-15 フラッシュコントロールレジスタ

FCFLCS
0x4004_0520

	31	30	29	28	27	26	25	24
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機能	リードすると'0'が読めます							
	23	22	21	20	19	18	17	16
bit Symbol	-	-	BLPRO5	BLPRO4	BLPRO3	BLPRO2	-	-
Read/Write	R		R	R	R	R	R	R
リセット後	0		(注2)	(注2)	(注2)	(注2)	(注2)	(注2)
機能	リードすると'0'が読めます		Block5 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block4 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block3 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block2 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	プロテクト状態(注3) 0:プロテクト状態ではない 1:プロテクト状態	プロテクト状態(注3) 0:プロテクト状態ではない 1:プロテクト状態
	15	14	13	12	11	10	9	8
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機能	リードすると'0'が読めます							
	7	6	5	4	3	2	1	0
bit Symbol	-	-	-	-	-	-	-	RDY/BSY
Read/Write	R							R
リセット後	0							1
機能	リードすると'0'が読めます							Ready/Busy(注1) 0:自動動作中 1:自動動作終了

ビット 0: Ready/Busy フラグビット

自動動作の状態を認識する方法として、RDY/BSY 出力を備えています。本ビットはこの機能を CPU からモニタするための機能ビットです。フラッシュメモリが自動動作中は“0”を出力し、ビジー状態であるとことを示します。自動動作が終了するとレディ状態となり“1”を出力し、次のコマンドを受け付けます。自動動作の結果が不良であった場合、本ビットは“0”出力を継続します。ハードウェアリセットにより“1”に復帰します。

ビット[21:16]: プロテクトビット状態ビット

プロテクトビット値は各ブロックのプロテクト状態に対応します。該当ビットが“1”の時は対応するブロックがプロテクト状態であることを示します。プロテクト状態のブロックは書き換えはできません。

- (注1) コマンド発行は、必ずレディ状態であることを確認してから発行して下さい。
 ビジー中にコマンド発行を行なった場合、正常なコマンドが送られないだけでなく、それ以降のコマンドを入力出来なくなる可能性があります。その際は、ハードウェアリセットで復帰して下さい。
 ハードウェアリセットを行う場合は、システムクロックによらず 0.5 μ s 以上のリセット期間が必要となります。またこの場合、リセット解除後読み出しが可能になるまで 2ms 程度の時間がかかります。
- (注2) プロテクト状態に対応した値になります。
- (注3) Flash の Block は存在しませんが、プロテクトビットは存在します。セキュリティ機能を使用する場合ブロック 0、1 相当のプロテクトビットをプログラムしてください。

表 18-16 セキュリティビットレジスタ

FCSECBIT
0x4004_0500

	31	30	29	28	27	26	25	24
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機能	リードすると '0' が読めます							
	23	22	21	20	19	18	17	16
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機能	リードすると '0' が読めます							
	15	14	13	12	11	10	9	8
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機能	リードすると '0' が読めます							
	7	6	5	4	3	2	1	0
bit Symbol	-	-	-	-	-	-	-	SECBIT
Read/Write	R							R/W
リセット後	0							1
機能	リードすると '0' が読めます							セキュリティビット 0: セキュリティ機能設定不可 1: セキュリティ機能設定可能

(注) 本レジスタは、パワーオンリセットでのみ初期化されます。

(6) コマンドシーケンス一覧

表 18-17 内部 CPU によるフラッシュメモリアクセス

コマンド シーケンス	第 1 バス サイクル	第 2 バス サイクル	第 3 バス サイクル	第 4 バス サイクル	第 5 バス サイクル	第 6 バス サイクル	第 7 バス サイクル
	Addr.	Addr.	Addr.	Addr.	Addr.	Addr.	Addr.
	Data	Data	Data	Data	Data	Data	Data
Read	0xXX	—	—	—	—	—	—
	0xF0	—	—	—	—	—	—
Read/リセット	0x54XX	0xAAXX	0x54XX	RA	—	—	—
	0xAA	0x55	0xF0	RD	—	—	—
ID-Read	0x54XX	0xAAXX	0x54XX	IA	0xXX	—	—
	0xAA	0x55	0x90	0x00	ID	—	—
自動ページ プログラム	0x54XX	0xAAXX	0x54XX	PA	PA	PA	PA
	0xAA	0x55	0xA0	PD0	PD1	PD2	PD3
自動チップ消去	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	0x54XX	—
	0xAA	0x55	0x80	0xAA	0x55	0x10	—
自動 ブロック消去	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	BA	—
	0xAA	0x55	0x80	0xAA	0x55	0x30	—
自動プロテクトビット プログラム	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	0x54XX	PBA
	0xAA	0x55	0x9A	0xAA	0x55	0x9A	0x9A
自動プロテクトビット 消去	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	0x54XX	PBA
	0xAA	0x55	0x6A	0xAA	0x55	0x6A	0x6A

補足説明

- RA: リードアドレス
- RD: リードデータ
- IA: ID アドレス
- ID: ID データ
- PA: プログラム ページアドレス
PD: プログラムデータ(32 ビットデータ)
第 4 バスサイクル以降 1 ページ分をアドレス順にデータ入力
- BA: ブロックアドレス
- PBA: プロテクトビットアドレス

(注 1) 全バスサイクル、アドレスビット[1:0]へは常に"0"を設定して下さい。(ビット[7:2]への設定値は"0"推奨)

(注 2) Read コマンドの第 2 バスサイクル、Read/リセットコマンドの第 4 バスサイクル、ID-Read コマンドの第 5 バスサイクル以外は全て"バスライトサイクル"です。バスライトサイクルは 32 ビット(ワード)のデータ転送命令で実施します。各バスライトサイクルのアドレス[31:16]はコマンドシーケンス実施対象のフラッシュメモリのアドレス[31:16]を用います。アドレス[15:0]は表の"Addr."を用います。

(7) バスライトサイクル時のアドレスビット構成

表 18-18 バスライトサイクル時のアドレスビット構成

アドレス	Addr [31:19]	Addr [18]	Addr [17]	Addr [16]	Addr [15]	Addr [14]	Addr [13:11]	Addr [10]	Addr [9]	Addr [8]	Addr [7:0]	
通常 コマンド	通常のバスライトサイクルアドレス設定											
	フラッシュ領域	"0"推奨			コマンド					Addr[1:0]="0"固定、 他ビットは"0"推奨		
ID -READ	IA:ID アドレス (ID-READ の第 4 バスライトサイクルアドレス設定)											
	フラッシュ領域	"0"推奨			ID アドレス	Addr[1:0]="0"固定、他ビットは"0"推奨						
ブロック 消去	BA:ブロックアドレス (ブロック消去の第 6 バスライトサイクルアドレス設定)											
	ブロックアドレス(表 18-19)					Addr[1:0]="0"固定、他ビットは"0"推奨						
Auto ページプロ グラム	PA:プログラムページアドレス (ページプログラムの第 4 バスライトサイクルアドレス設定)											
	ページアドレス								Addr[1:0]="0"固定、 他ビットは"0"推奨			
プロテクト ビットプロ グラム	PBA:プロテクトビットアドレス (プロテクトビットプログラムの第 7 バスライトサイクルアドレス設定)											
	フラッシュ領域	プロテクトビット選択 (表 18-20)			"0"固定			プロテクトビット 選択(表 18-20)	Addr[1:0]="0"固定、 他ビットは"0"推奨			
プロテクト ビット消去	PBA:プロテクトビットアドレス (プロテクトビット消去の第 7 バスライトサイクルアドレス設定)											
	フラッシュ領域	プロテクトビット 選択 (表 18-21)			"0"推奨				プロテクトビット 選択 (表 18-21)	Addr[1:0]="0"固定、 他ビットは"0"推奨		

(注 1) 表 16-17 「内部 CPU によるフラッシュメモリアクセス」と併せてご使用願います。

(注 2) 第 1 バスサイクルから「通常のバスライトサイクルアドレス設定」に従い、アドレス設定を行なって下さい。

(注 3) 「"0"推奨」は適宜変更可能です。

表 18-19 ブロックアドレス表

Block	アドレス (ユーザーブートモード)	アドレス (シングルブートモード)	サイズ (Kbyte)
4	0x0000_0000-0x0000_7FFF	0x3F80_0000-0x3F80_7FFF	32
5	0x0000_8000-0x0000_FFFF	0x3F80_0000-0x3F80_FFFF	32
3	0x0001_0000-0x0001_FFFF	0x3F81_0000-0x3F81_FFFF	64
2	0x0002_0000-0x0003_FFFF	0x3F82_0000-0x3F83_FFFF	128

ブロックアドレスには、消去するブロックに含まれる任意のアドレスを指定します。

(注) 第1バスサイクルから第5バスサイクルまで上位側のアドレスは消去するブロックのアドレスを指定してください。

表 18-20 プロテクトビットプログラムアドレス表

Block	プロテクトビット	第7バスライトサイクルのアドレス						
		アドレス [18]	アドレス [17]	アドレス [16]	アドレス [15:11]	アドレス [10]	アドレス [9]	アドレス [8]
Block0	-	0	0	1	"0"固定	0	0	"0"推奨
Block1	-	0	0	1		0	1	
Block2	BLPRO2	0	0	1		1	0	
Block3	BLPRO3	0	0	1		1	1	
Block4	BLPRO4	0	1	0		0	0	
Block5	BLPRO5	0	1	0		0	1	

表 18-21 プロテクトビット消去アドレス表

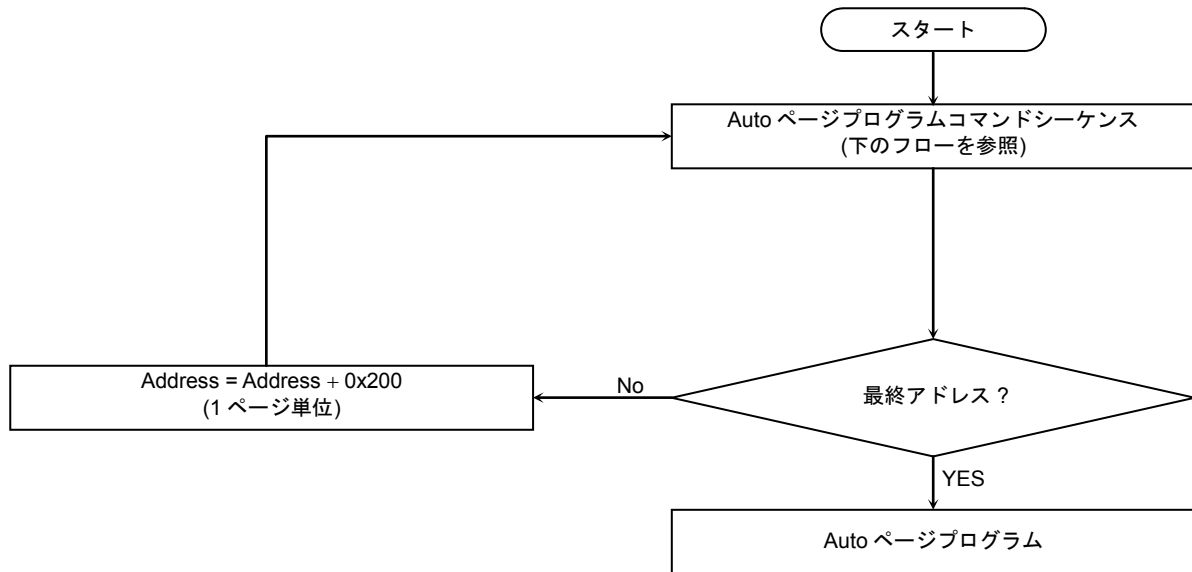
Block	プロテクトビット	第7バスライトサイクルのアドレス[18:17]	
		アドレス[18]	アドレス[17]
Block0~3	BLPRO0~3	0	0
Block4~5	BLPRO4~5	0	1

(注) プロテクトビット消去コマンドは プロテクトビット単位での消去は出来ません。

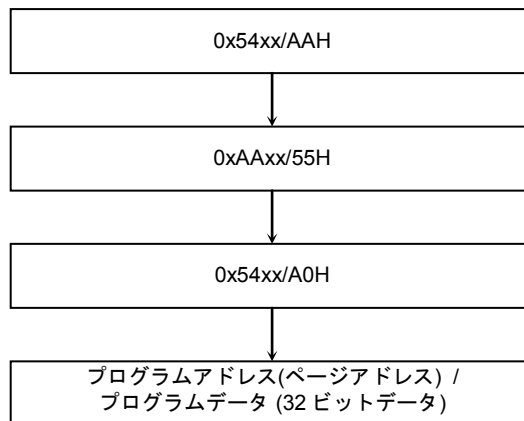
表 18-22 ID-Read コマンド第4バスライトサイクルのIDアドレス(IA)と
その後の32ビット転送命令で読み出せるデータ(ID)

IA [15:14]	ID [7: 0]	Code
00b	0x98	メーカーコード
01b	0x5A	デバイスコード
10b	Reserved	---
11b	0x12	マクロコード

(8) フローチャート



Auto ページプログラムコマンドシーケンス (アドレス/コマンド)



注) 0x54xx は 0x55xx でもコマンドシーケンスを実行します。

図 18-10 自動プログラム

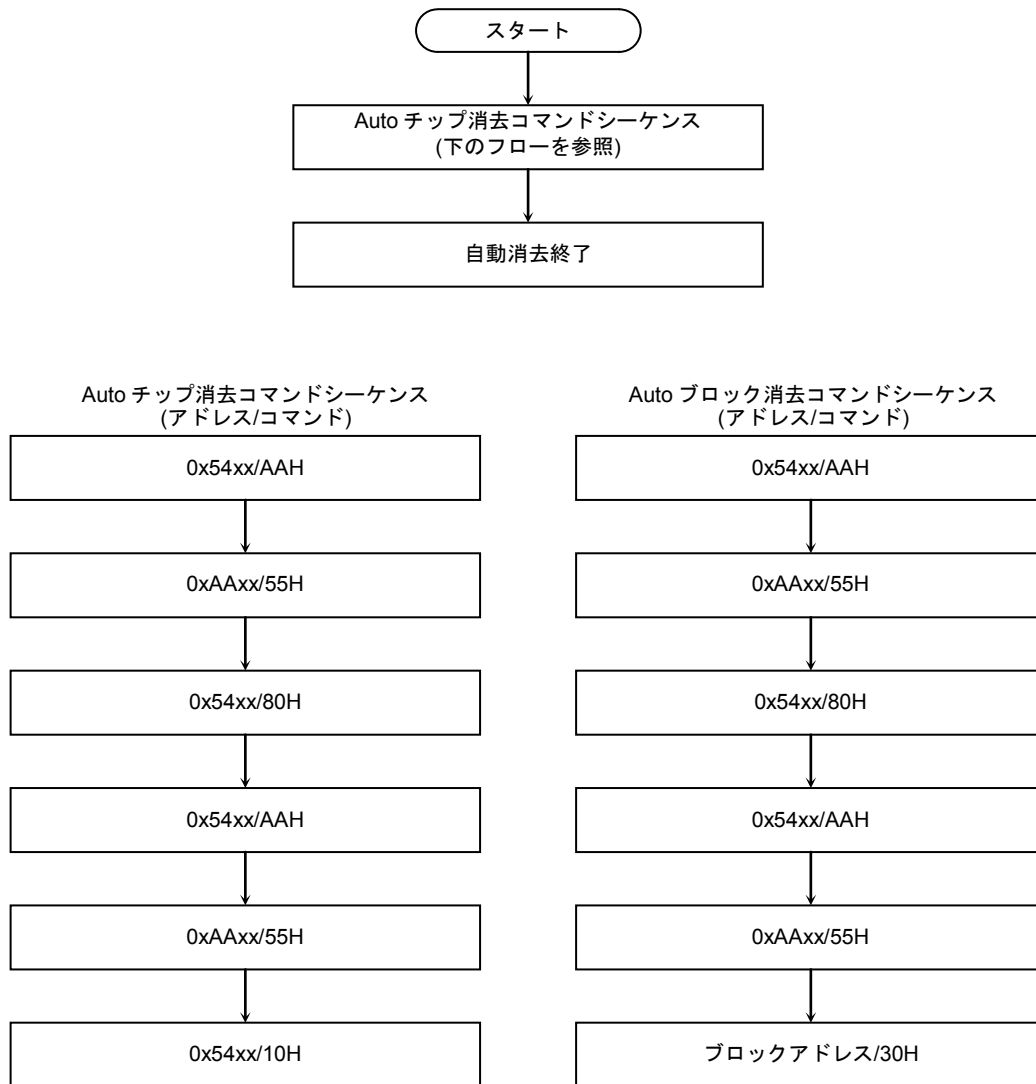


図 18-11 自動消去

注) 0x54xx は 0x55xx でもコマンドシーケンスを実行します。

19. プロテクト/セキュリティ機能

19.1 概要

本製品は内蔵 ROM(Flash)のライト/消去をプロテクトする機能、及びライターでの内蔵 ROM(Flash)領域を読み出し禁止に設定できるセキュリティ機能を内蔵しています。セキュリティ機能はデバック機能の使用制限も行います。プロテクト/セキュリティ機能として、次の2つの機能をもっています。

- 内蔵 ROM(Flash)のライト/消去プロテクト
- セキュリティ機能

19.2 特長

19.2.1 内蔵 ROM(Flash)のライト/消去プロテクト

内蔵フラッシュは、ブロック単位で書き込みと消去の動作を禁止することができます。この機能をライト/消去プロテクトと呼びます。

ライト/消去プロテクト機能を有効にするためには、プロテクトをかけたいブロックに対応するプロテクトビットを”1”にします。プロテクトビットを”0”にすることによりブロックプロテクトを解除することができます。(プログラム方法については、Flash 動作説明の章をご覧ください。)

プロテクトビットは、FCFLCS レジスタ<BLPRO>ビットでモニタすることができます。

ライト/消去プロテクトは、シングルチップモード、シングルブートモード、ライターモードで有効となります。

19.2.2 セキュリティ機能

内蔵フラッシュに対してのデータの読み出しの制限及び、デバック機能を制限することができます。この機能をセキュリティ機能と呼びます。

セキュリティ機能が有効になる条件を、以下に示します。

- 1) FCSECBIT レジスタの<SECBIT>ビットが”1”にセットされている。
- 2) ライト/消去プロテクト用の全てのプロテクトビット (FCFLCS レジスタ<BLPRO>ビット) が”1”にセットされている。

注)FCSECBIT レジスタの<SECBIT>ビットは電源投入直後のパワーオンリセット時に”1”にセットされます。

セキュリティ機能が有効な状態の制限内容を、表 19-1 に示します。

表 19-1 セキュリティ機能有効時の制限内容

項目	内容
1)ROM 領域のデータの読み出し	ライターモードでデータが読み出せなくなります。リード動作を行うとメーカーコードの 0x0098 が読み出されます。シングルチップモード、シングルブートモードでは読み出し可能です。
2)デバックポート	JTAG/SW、トレースの通信が出来なくなります。
3)フラッシュに対するコマンドの実行	フラッシュに対してのコマンドライトが受け付けられません。また、ライト/消去プロテクト用のプロテクトビットを消去しようとする、チップ消去が行われ、全てのプロテクトビットも消去されます。

19.3 レジスタ

フラッシュメモリのステータスマニタと、ブロック単位のプロテクト状態を示すレジスタです。

表 19-2 フラッシュコントロールレジスタ

FCFLCS
0x4004_0520

	31	30	29	28	27	26	25	24
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機能	リードすると0が読めます							
	23	22	21	20	19	18	17	16
bit Symbol	-	-	BLPRO5	BLPRO4	BLPRO3	BLPRO2	BLPRO1	BLPRO0
Read/Write	R		R	R	R	R	R	R
リセット後	0		(注2)	(注2)	(注2)	(注2)	(注2)	(注2)
機能	リードすると0が読めます	Block5 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block4 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block3 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block2 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block1 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	Block0 のプロテクト状態 0:プロテクト状態ではない 1:プロテクト状態	
	15	14	13	12	11	10	9	8
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機能	リードすると0が読めます							
	7	6	5	4	3	2	1	0
bit Symbol	-	-	-	-	-	-	-	RDY/BSY
Read/Write	R							R
リセット後	0							1
機能	リードすると0が読めます							Ready/Busy (注1) 0:自動動作中 1:自動動作終了

ビット 0: Ready/Busy フラグビット

自動動作の状態を認識する方法として、RDY/BSY 出力を備えています。本ビットはこの機能を CPU からモニタするための機能ビットです。フラッシュメモリが自動動作中は“0”を出力し、ビジー状態であることとを示します。自動動作が終了するとレディ状態となり“1”を出力し、次のコマンドを受け付けます。自動動作の結果が不良であった場合、本ビットは“0”出力を継続します。ハードウェアリセットにより“1”に復帰します。

ビット[21:16]: プロテクトビット状態ビット

プロテクトビット(6ビット)値は各ブロックのプロテクト状態に対応します。該当ビットが“1”の時は対応するブロックがプロテクト状態であることを示します。プロテクト状態のブロックは書き換えはできません。

(注1) コマンド発行は、必ずレディ状態であることを確認してから発行して下さい。
 ビジー中にコマンド発行を行なった場合、正常なコマンドが送られないだけでなく、
 それ以降のコマンドを入力出来なくなる可能性があります。その際は、ハードウェア
 リセットで復帰して下さい。
 ハードウェアリセットを行う場合は、システムクロックによらず 0.5μs 以上のリセッ
 ト期間が必要となります。またこの場合、リセット解除後読み出しが可能になるまで
 2ms 程度の時間がかかります。

(注2) プロテクト状態に対応した値が読めます。

表 19-3 セキュリティビットレジスタ

FCSECBIT
 0x4004_0500

	31	30	29	28	27	26	25	24
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機 能	リードすると '0' が読めます							
	23	22	21	20	19	18	17	16
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機 能	リードすると '0' が読めます							
	15	14	13	12	11	10	9	8
bit Symbol	-	-	-	-	-	-	-	-
Read/Write	R							
リセット後	0							
機 能	リードすると '0' が読めます							
	7	6	5	4	3	2	1	0
bit Symbol	-	-	-	-	-	-	-	SECBIT
Read/Write	R							R/W
リセット後	0							1
機 能	リードすると '0' が読めます							セキュリティ ビット 0: セキュリ ティ機能 設定不可 1: セキュリ ティ機能 設定可能

(注1) 本レジスタは、パワーオンリセットでのみ初期化されます。

19.4 設定／解除方法

19.4.1 内蔵 ROM(Flash)のライト/消去プロテクト

プロテクトビットの書き込みはブロック単位で行います。プロテクトビットの書き込みはコマンドシーケンスを用いて行います。

プロテクトビットの消去はブロックをまとめて行います。Block2～3 と Block4～5 の2つの単位で消去を行います。ただし、全 Block 全てのプロテクトビットが”1”のときは、FCSECBIT レジスタの<SECBIT>ビットを”0”にしてから、消去を行う必要があります。FCSECBIT レジスタの<SECBIT>ビットが”1”の状態プロテクトビットの消去を行おうとすると、チップ消去を行い、プロテクトビット全てを消去します。プロテクトビットの消去はコマンドシーケンスを用いて行います。

プロテクトビットのライト/消去は、シングルチップモード、シングルブートモード、ライターモードで行います。

詳細は Flash 動作説明章を参照して下さい。

19.4.2 セキュリティビット

セキュリティ機能を有効にする FCSECBIT レジスタの<SECBIT>ビットは電源投入時のパワーオンリセット時に”1”にセットされます。FCSECBIT レジスタの<SECBIT>ビットの書き換えは以下の手順で行います。セキュリティビットの書き換えは、シングルチップモード、シングルブートモードで行います。

- 1) FCSECBIT レジスタに対して特定のコード(0xa74a9d23)を書き込む。
- 2) 1)の書き込みから 16 クロック以内にデータを書き込む。

注)上記 1)、2)の書き込みは 32bit 転送命令のみ有効となります。

20. 電気的特性

20.1 絶対最大定格

項目		記号	定格	単位
電源電圧		DVCC (I/O)	- 0.3~3.9	V
		AVCC (A/D)	- 0.3~3.9	
		CVCC (CLK)	- 0.3~3.9	
		REGVCC	- 0.3~3.9	
入力電圧		V_{IN}	$- 0.3 \sim V_{CC} + 0.3$	V
低レベル 出力電流	1 端子	I_{OL}	5	mA
	合計	ΣI_{OL}	50	
高レベル 出力電流	1 端子	I_{OH}	- 5	
	合計	ΣI_{OH}	50	
消費電力($T_a = 85^\circ\text{C}$)		PD	600	mW
はんだ付け温度(10s)		T_{SOLDER}	260	$^\circ\text{C}$
保存温度		T_{STG}	- 55 ~ 125	$^\circ\text{C}$
動作温度	Flash W/E 時を除く	T_{OPR}	- 40 ~ 85	$^\circ\text{C}$
	Flash W/E 時		0 ~ 70	

(注) 絶対最大定格とは、瞬時たりとも超えてはならない規格であり、どの1つの項目も超えることができない規格です。絶対最大定格（電流、電圧、消費電力、温度）を超えると破壊や劣化の原因となり、破裂・燃焼による障害を負うことがあります。したがってかならず絶対最大定格を超えないように、応用機器の設計を行ってください。

20.2 DC 電気的特性 (1/2)

Ta = -40 ~ 85°C

項目		記号	条件	Min.	Typ. (注1)	Max.	単位	
電源電圧	DVCC = AVCC = CVCC = REGVCC (注2) DVSS = AVSS = 0V	DVCC AVCC CVCC REGVCC	fosc = 8 ~ 10MHz fsys = 1 ~ 40MHz fs = 30 ~ 34kHz	2.7	-	3.6	V	
	低レベル入力電圧	PC, PD	V _{IL1}	2.7V ≤ AVCC ≤ 3.6V	-0.3	-	0.3 AVCC	V
		ノーマルポート	V _{IL2}				0.3 DVCC	
		シュミット入力	V _{IL3}	2.7V ≤ DVCC ≤ 3.6V			0.2 DVCC	
		CEC	V _{IL4}				2.7V ≤ CVCC ≤ 3.6V	
X1		V _{IL5}						
高レベル入力電圧	PC, PD	V _{IH1}	2.7V ≤ AVCC ≤ 3.6V	0.7 AVCC	-	AVCC+0.3	V	
	ノーマルポート	V _{IH2}	2.7V ≤ DVCC ≤ 3.6V	0.7 DVCC		DVCC+0.3		
	シュミット入力	V _{IH3}		0.8 DVCC		3.6		
	CEC	V _{IH4}	2.7V ≤ CVCC ≤ 3.6V	0.9 CVCC		CVCC+0.3		
	X1	V _{IH5}						
低レベル出力電圧		V _{OL}	I _{OL} = 2mA DVCC ≥ 2.7V	-	-	0.4	V	
高レベル出力電圧		V _{OH}	I _{OH} = -2mA DVCC ≥ 2.7V	2.4	-	-	V	
入力リーク電流	CEC 以外	I _{LI1}	0.0 ≤ V _{IN} ≤ DVCC 0.0 ≤ V _{IN} ≤ AVCC	-	0.02	± 5	μA	
	CEC	I _{LI2}	0.0 ≤ V _{IN} ≤ 3.6					
出力リーク電流	CEC 以外	I _{LO1}	0.2 ≤ V _{IN} ≤ DVCC - 0.2 0.2 ≤ V _{IN} ≤ AVCC - 0.2	-	0.05	± 10		
	CEC	I _{LO2}	0.2 ≤ V _{IN} ≤ 3.4					
リセットプルアップ抵抗		RRST	DVCC = 2.7V ~ 3.6V	-	50	150	kΩ	
シュミット入力幅		V _{TH}	2.7V ≤ DVCC ≤ 3.6V	0.3	0.6	-	V	
プログラマブルプルアップ/ダウン抵抗		PKH	DVCC = 2.7V ~ 3.6V	-	50	150	kΩ	
Pin 容量 (電源端子を除く)		C _{IO}	fc = 1MHz	-	-	10	pF	

(注1) Typ 値は特に指定のない限り Ta=25°C, DVCC = REGVCC = AVCC = 3.3V の値です。

(注2) DVCC、AVCC、CVCC、REGVCC は同電圧で使用してください。

20.3 DC 電気的特性 (2/2)

(1) TMPM330FYWFG

DVCC = AVCC = CVCC = REGVCC = 2.7V~3.6V, Ta = -40~85°C

項目	記号	条件	Min.	Typ. (注1)	Max.	単位
NORMAL (注2) ギア比 1/1	I _{CC}	f _{sys} = 40 MHz (f _{osc} = 10 MHz)	-	32	42	mA
IDLE (注3)			-	8	13	
SLOW		-	2.5	6		
SLEEP		f _s = 32.768kHz	-	55	500	μA
STOP			-	45	490	

(注1) Typ 値は特に指定のない限り Ta=25°C, DVCC = AVCC = CVCC = REGVCC = 3.3V の値です。

(注2) I_{CC} NORMAL の測定条件 :

実行プログラム : ドライストン V2.1 (内蔵 FLASH 動作)
A/DC を除く内蔵周辺機能は全て動作

(注3) I_{CC} IDLE の測定条件 :

周辺機能は全て停止
I_{CC} には DVCC, AVCC, CVCC, REGVCC に流れる電流が含まれます。

20.4 10ビット A/D コンバータ変換特性

DVCC = AVCC = CVCC = REGVCC = VREFH = 2.7V~3.6V,

AVSS = DVSS, Ta = -40~85°C

AVCC 負荷容量 $\geq 3.3 \mu\text{F}$, VREFH 負荷容量 $\geq 3.3 \mu\text{F}$

項目	記号	条件	Min	Typ	Max	単位
アナログ基準電圧(+)	VREFH	-	2.7	3.3	3.6	V
アナログ入力電圧	VAIN	-	AVSS	-	VREFH	V
アナログ基準 電圧電源電流	A/D 変換時	IREF DVSS = AVSS	-	2.5	5.5	mA
	A/D 非変換時		-	± 0.02	± 5	μA
消費電流	A/D 変換時	-	-	-	3	mA
積分非直線性誤差	-	AIN 負荷抵抗 $\leq 600 \Omega$ AIN 負荷容量 $\leq 30\text{pF}$ 変換時間 $\geq 1.15 \mu\text{s}$	-	± 2	± 3	LSB
微分直線性誤差			-	± 1	± 2	
オフセット誤差			-	± 2	± 4	
フルスケール誤差			-	± 2	± 4	
積分非直線性誤差	-	AIN 負荷抵抗 $\leq 600 \Omega$ AIN 負荷容量 $\leq 0.1 \mu\text{F}$ 変換時間 $\geq 1.15 \mu\text{s}$	-	± 2	± 3	
微分直線性誤差			-	± 1	± 2	
オフセット誤差			-	± 2	± 4	
フルスケール誤差			-	± 2	± 4	
積分非直線性誤差	-	AIN 負荷抵抗 $\leq 10\text{k}\Omega$ AIN 負荷容量 $\geq 0.1 \mu\text{F}$ 変換時間 $\geq 2.30 \mu\text{s}$	-	± 2	± 3	
微分直線性誤差			-	± 1	± 2	
オフセット誤差			-	± 2	± 4	
フルスケール誤差			-	± 2	± 4	

(注) 1LSB = (VREFH - AVSS) / 1024[V]

20.5 AC 電気的特性

20.5.1 シリアルチャネルタイミング (SIO)

(1) I/O インタフェースモード

表中の x は SIO の動作クロックの周期を表します。SIO の動作クロックは、システムクロック f_{sys} と同じ周期です。この周期は、クロックギアの設定に依存します。

① □ SCLK 入力モード

[入力]

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
SCLK クロック High 幅(入力)	t_{SCH}	$4x$	-	100	-	ns
SCLK クロック Low 幅 (入力)	t_{SCL}	$4x$	-	100	-	
SCLK 周期	t_{SCY}	$t_{SCH} + t_{SCL}$	-	200	-	
有効 Data 入力 ← SCLK 立ち上がり/立ち下がり(注1)	t_{SRD}	30	-	30	-	
SCLK 立ち上がり → Input Data 保持/立ち下がり(注1)	t_{HSR}	$x + 30$	-	55	-	

[出力]

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
SCLK クロック High 幅(入力)	t_{SCH}	$4x$	-	120 (注3)	-	ns
SCLK クロック Low 幅 (入力)	t_{SCL}	$4x$	-	120 (注3)	-	
SCLK 周期	t_{SCY}	$t_{SCH} + t_{SCL}$	-	240	-	
Output Data ← SCLK 立ち上がり/立ち下がり (注1)	t_{OSS}	$t_{scv} / 2 - 2x - 45$	-	0 (注2)	-	
SCLK 立ち上がり → Output Data 保持/立ち下がり(注1)	t_{OHS}	$t_{scv} / 2$	-	120	-	

(注1) SCLK 立ち上がり/立ち下がり…SCLK 立ち上がりモードの場合は SCLK 立ち上がり、SCLK 立ち下がりモードの場合は SCLK 立ち下がりのタイミングです。

(注2) 計算値がマイナスにならない範囲の SCLK 周期で使用してください。

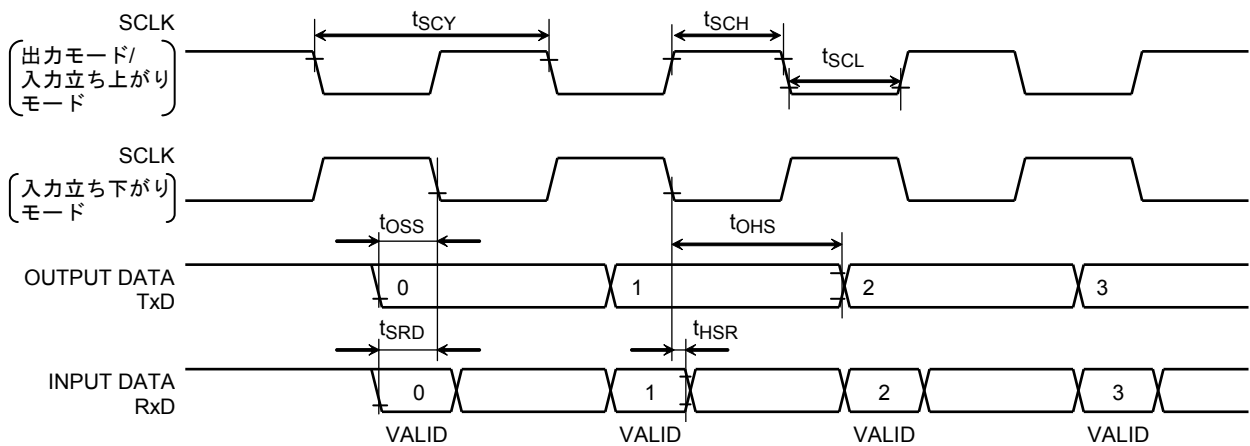
(注3) t_{OSS} がマイナスにならない最少値を示しています。

② SCLK 出力モード

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
SCLK 周期 (プログラマブル)	t_{SCY}	4x	-	100	-	ns
Output Data ← SCLK 立ち上がり	t_{OSS}	$t_{SCY} / 2 - 20$	-	30	-	
SCLK 立ち上がり → Output Data 保持	t_{OHS}	$t_{SCY} / 2 - 20$	-	30	-	
有効 Data 入力 ← SCLK 立ち上がり	t_{SRD}	45	-	45	-	
SCLK 立ち上がり → Input put Data 保持	t_{HSR}	0	-	0	-	

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, CL=30 pF

・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照



20.5.2 シリアルバスインターフェース (I2C)

(1) I2C モード

x は I2C の動作クロックの周期を表します。I2C の動作クロックは、システムクロック f_{sys} と同じ周期です。この周期は、クロックギアの設定に依存します。

n は SBInCR レジスタの SCK フィールドで指定した SCL 出力クロックの周波数選択値です。

項目	記号	計算式		標準モード		ファーストモード		単位
		Min	Max	Min	Max	Min	Max	
SCL クロック周波数	t _{SCL}	0	-	0	100	0	400	kHz
スタートコンディション保持	t _{HD:STA}	-	-	4.0	-	0.6	-	μs
SCL クロック Low 幅 (入力) (注1)	t _{LOW}	-	-	4.7	-	1.3	-	μs
SCL クロック High 幅 (入力) (注2)	t _{HIGH}	-	-	4.0	-	0.6	-	μs
再スタートコンディション セットアップ時間	t _{SU:STA}	ソフト (注5)	-	4.7	-	0.6	-	μs
データ保持時間 (入力) (注3,4)	t _{HD:DAT}	-	-	0.0	-	0.0	-	μs
データセットアップ時間	t _{SU:DAT}	-	-	250	-	100	-	ns
ストップコンディション セットアップ時間	t _{SU:STO}	-	-	4.0	-	0.6	-	μs
ストップコンディションとスタート コンディション間のバスフリー時間	t _{BUF}	ソフト (注5)	-	4.7	-	1.3	-	μs

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, CL=30 pF

・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照

注1) SCL クロック LOW 幅 (出力): $(2^{n-1} + 58) / x$

注2) SCL クロック HIGH 幅 (出力): $(2^{n-1} + 14) / x$

通信規格上、標準モード/高速モードの最高速は 100kHz/400kHz です。内部 SCL クロックの周波数の設定は、使用される f_{sys} と上記計算式にて設定されますのでご注意願います。

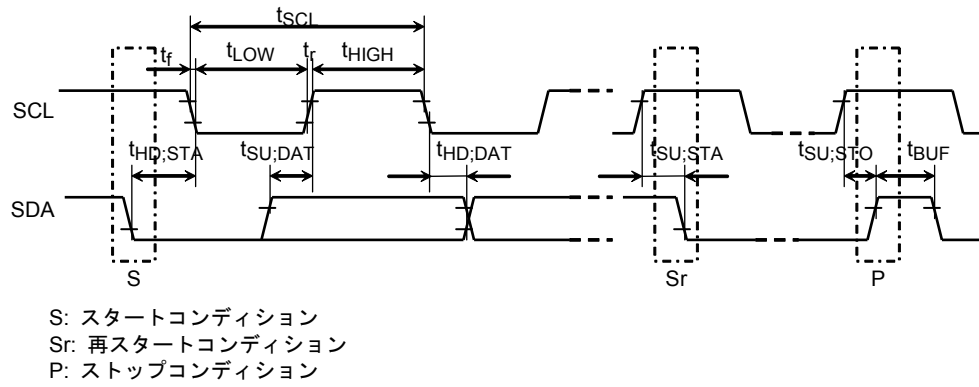
注3) データ保持時間 (出力) は内部 SCL から 4x の時間です。

注4) フィリップススペックでは内部で、SDA 入力時にデータホールド 300nsec を確保して SCL 立下り時の不安定状態を回避する事になっていますが、本製品では対応していません。また SCL のエッジスロープコントロール機能をもっていません。従って、SCL/SDA の tr/tf を含めて BUS 上で上表のデータ保持時間 (入力) を守る様に設計してください。

注5) ソフトウェアに依存します。

注6) フィリップススペックでは、ファーストモードデバイスへの電源供給が切れた場合、バスラインを妨げることがないように SDA および SCL 信号の I/O ピンと電源との接続が外れるようにする必要がありますが、本デバイスでは対応していません。

注) 通信規格上、標準モード/高速モードの最高速は 100kHz/400kHz です。
内部 SCL クロックの周波数の設定は、使用される f_{sys} と上記注1, 注2の計算式にて設定されますのでご注意願います。



(2) クロック同期式 8 ビット SIO モード

x は I2C の動作クロックの周期を表します。I2C の動作クロックは、システムクロック f_{sys} と同じ周期です。この周期は、クロックギアの設定に依存します。

① □ SCK 入力モード (SCK デューティ 50%の場合)

[データ入力]

項 目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
SCK クロック High 幅(入力)	t _{SCH}	4x	-	100	-	ns
SCK クロック Low 幅(入力)	t _{SCL}	4x	-	100	-	
SCK 周期	t _{SCY}	t _{SCH} + t _{SCL}	-	200	-	
有効 Data 入力 ← SCK 立ち上がり	t _{SRD}	30 - x	-	5	-	
SCK 立ち上がり → Input Data 保持	t _{HSR}	2x+30	-	80	-	

[データ出力]

項 目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
SCK クロック High 幅(入力)	t _{SCH}	4x	-	100	-	ns
SCK クロック Low 幅(入力)	t _{SCL}	4x	-	100	-	
SCK 周期	t _{SCY}	t _{SCH} + t _{SCL}	-	200	-	
Output Data ← SCK 立ち上がり	t _{OSS}	t _{SCY} /2 - 3x - 45	-	0 (注 1)	-	
SCK 立ち上がり → Output Data 保持	t _{OHS}	t _{SCY} /2 + x	-	125	-	

- AC 測定条件
- ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, CL=30 pF
 - ・ 入力レベル: DC 電气的特性の、低レベル入力電圧/高レベル入力電圧参照

(注 1) 計算値がマイナスにならない範囲の SCLK 周期で使用してください。

(注 2) t_{OSS} がマイナスにならない最小値を示しています。

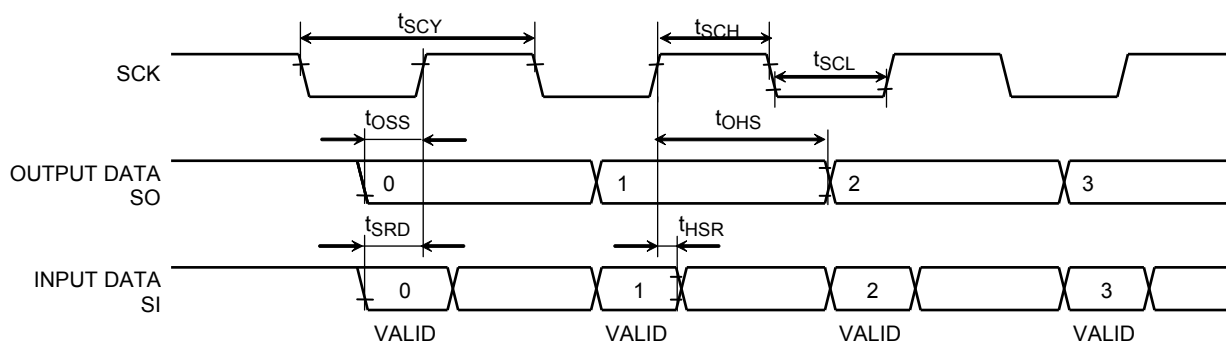
② SCK 出力モード

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
SCK 周期(プログラマブル)	t _{SCY}	16x'(注 1)	-	400	-	ns
Output Data ← SCK 立ち上がり	t _{OSS}	t _{SCY} /2 - 20 (注 2)	-	180	-	
SCK 立ち上がり → Output Data 保持	t _{OHS}	t _{SCY} /2 - 20	-	180	-	
有効 Data 入力 ← SCK 立ち上がり	t _{SRD}	x+45	-	70	-	
SCK 立ち上がり → Input Data 保持	t _{HSR}	0	-	0	-	

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, CL=30 pF
 ・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照

(注 1) 自動wait 後のSCK 周期が14x になります。

(注 2) 自動 wait 後の SO データ出力が、t_{SCY}/2- x- 20 になる場合があります。



20.5.3 イベントカウンタ

x は TMRB の動作クロックの周期を表します。TMRB の動作クロックは、システムクロック f_{sys} と同じ周期です。この周期は、クロックギアの設定に依存します。

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
クロック低レベルパルス幅	t_{VCKL}	$2x + 100$	-	150	-	ns
クロック高レベルパルス幅	t_{VCKH}	$2x + 100$	-	150	-	ns

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, $CL=30$ pF
 ・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照

20.5.4 キャプチャ

x は TMRB の動作クロックの周期を表します。TMRB の動作クロックは、システムクロック f_{sys} と同じ周期です。この周期は、クロックギアの設定に依存します。

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
低レベルパルス幅	t_{CPL}	$2x + 100$	-	150	-	ns
高レベルパルス幅	t_{CPH}	$2x + 100$	-	150	-	ns

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, $CL=30$ pF
 ・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照

20.5.5 外部割り込み (STOP 解除時以外)

表中の x はシステムクロック f_{sys} の周期を表します。

・ STOP 解除割り込み以外

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
NMI, INT0~7 低レベルパルス幅	t_{INTAL}	$x + 100$	-	125	-	ns
INT0~7 高レベルパルス幅	t_{INTAH}	$x + 100$	-	125	-	ns

・ STOP 解除割り込み

項目	記号	Min	Max	単位
NMI, INT0~7 低レベルパルス幅	t_{INTBL}	100	-	ns
INT0~7 高レベルパルス幅	t_{INTBH}	100	-	ns

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, $CL=30$ pF
 ・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照

20.5.6 外部割り込み (STOP 解除時)

項目	記号	Min	Max	単位
NMI 低レベルパルス幅	t_{INTBL}	100	-	ns

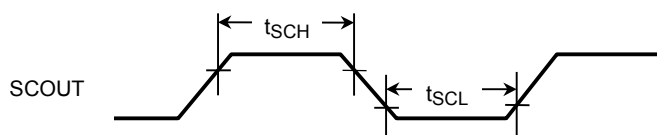
AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, $CL=30$ pF
 ・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照

20.5.7 SCOUT 端子 AC 特性

項目	記号	計算式		40MHz		単位
		Min	Max	Min	Max	
高レベルパルス幅	t_{SCH}	$0.5T - 5$		7.5		ns
低レベルパルス幅	t_{SCL}	$0.5T - 5$		7.5		ns

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, $C_L = 30$ pF
 ・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照

(注) 表中の「T」は、SCOUT 出力波形の周期を示します。



20.5.8 デバッグ通信

(1) SWD インタフェース

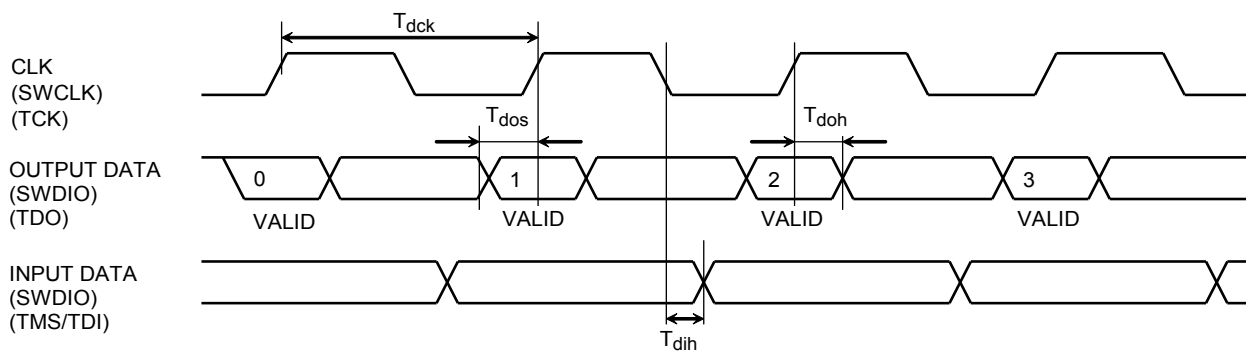
項目	記号	Min	Max	単位
CLK 周期	T_{dck}	83.33	-	ns
DATA 入力←CLK 立ち上がり	T_{dos}	12	-	ns
CLK 立ち上がり→DATA 保持	T_{doh}	4	-	ns
CLK 立下り→DATA 保持	T_{dih}	-	6	ns

AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, CL=30 pF
 ・ 入力レベル: DC 電气的特性の、低レベル入力電圧/高レベル入力電圧参照

(2) JTAG インタフェース

項目	記号	Min	Max	単位
CLK 周期	T_{dck}	100	-	ns
DATA 入力←CLK 立ち上がり	T_{dos}	3	-	ns
CLK 立ち上がり→DATA 保持	T_{doh}	25	-	ns
CLK 立下り→DATA 保持	T_{dih}	-	15	ns

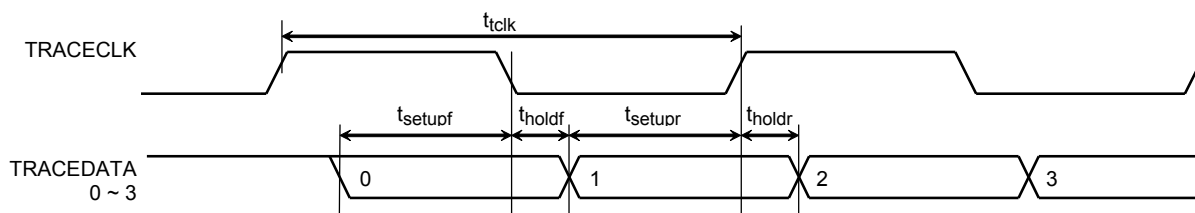
AC 測定条件 ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, CL=30 pF
 ・ 入力レベル: DC 電气的特性の、低レベル入力電圧/高レベル入力電圧参照



20.5.9 ETM トレース

項目	記号	Min	Max	単位
TRACECLK 周期	t_{clk}	50	-	ns
TRACEDATA 有効←DCLK 立ち上がり	t_{setupr}	2	-	ns
DCLK 立ち上がり→TRACEDATA 保持	t_{holdr}	1	-	ns
TRACEDATA 有効←DCLK 立ち下がり	t_{setupf}	2	-	ns
DCLK 立ち下がり→TRACEDATA 保持	t_{holdf}	1	-	ns

- AC 測定条件
- ・ 出力レベル: High 0.8DVCC3 V/Low 0.2DVCC3V, CL=30 pF
 - ・ 入力レベル: DC 電気的特性の、低レベル入力電圧/高レベル入力電圧参照



20.6 発振回路

本製品は、下記の発振子メーカーにて評価されております。発振子の選択時に活用願います。

(注) 発振端子の負荷容量は、接続する負荷容量 C1, C2 と実装基板上の浮遊容量の和になります。C1, C2 の定数を使用した場合でも実装基板により負荷容量が異なり発振器が誤動作する可能性があります。従って、基板設計の際には発振回路周辺のパターンが最短距離になるようにしてください。最終的に実装基板での発振子評価を推奨いたします。

(1) 発振子接続回路例

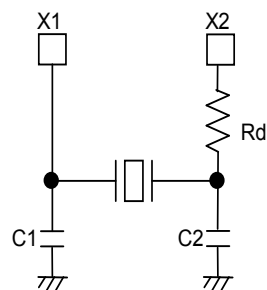


図 19.1 高周波発振器の接続図

(2) 本製品の推奨セラミック発振子

本製品は、(株)村田製作所製セラミック発振子を推奨しております。
詳細につきましては、下記 URL の同社ホームページを参照してください。
<http://www.murata.co.jp>

(3) 本製品の推奨水晶発振子

本製品は、京セラクリスタルデバイス株式会社製水晶発振子を推奨しております。
詳細につきましては、下記 URL の同社ホームページを参照してください。
<http://www.kinseki.co.jp>

20.7 取り扱い上のご注意

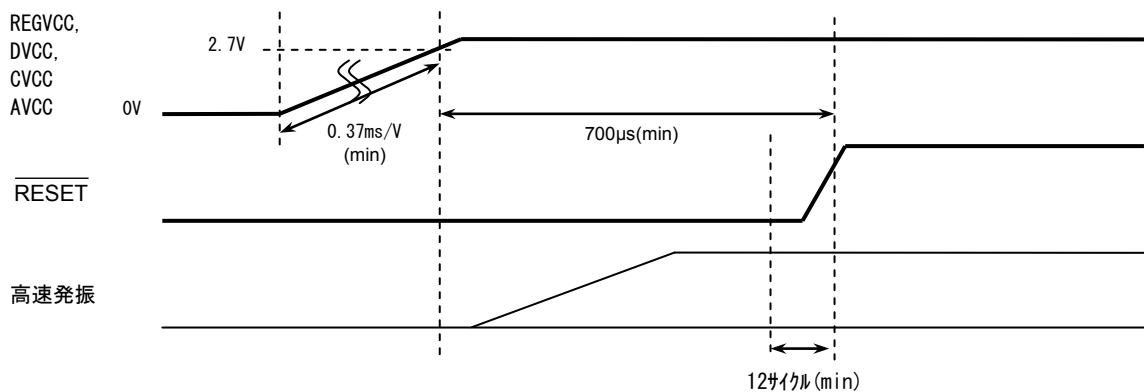
20.7.1 電源の注意事項

20.7.1.1 電源投入時の注意事項

電源立ち上げ(0V から 2.7V への立ち上げ)勾配は、0.37ms/V 以上で使用してください。

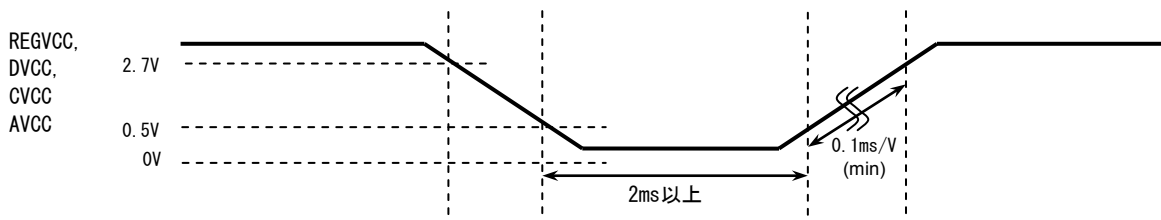
電源投入の際には、内蔵レギュレータの安定のための時間と発振安定の時間を考慮する必要があります。本製品では、内蔵レギュレータ安定のための時間として $700\mu\text{s}$ が必要です。また、内蔵レギュレータと発振が安定するための十分な時間、外部リセット端子に“L”を入力する必要があります。

電源投入の手順を以下に示します。



20.7.1.2 電源再投入時の注意事項

電源を遮断し、再投入する場合、電源の電圧を 0.5V 以下まで落とし、2ms 以上保持してください。その後、電源立ち上げ(2.7V への立ち上げ)勾配は 0.1ms/V で使用してください。リセット信号の入力期間は 20.7.2.1 を参照してください。



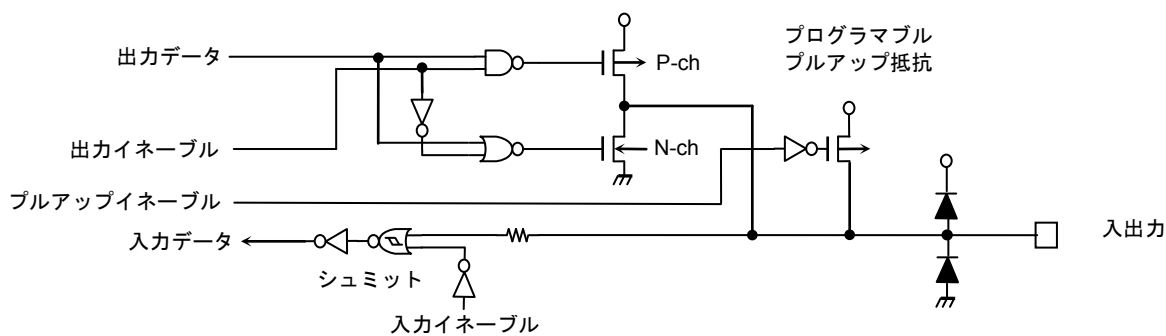
21. ポート部等価回路図

- 回路図の見方

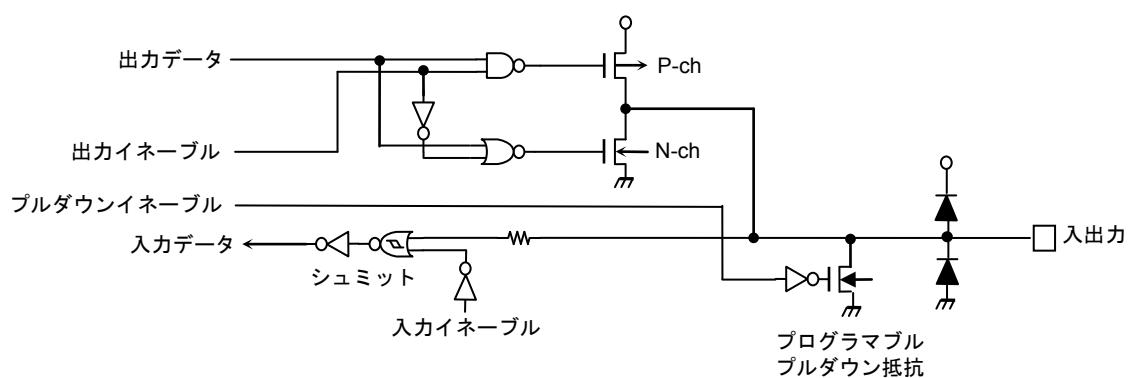
基本的に、標準 CMOS ロジック IC 「74HCxx」 シリーズと同じゲート記号を使って書かれています。

入力保護抵抗は、数十 Ω ~数百 Ω 程度です。X2,XT2 のダンピング抵抗値は、図中に typ 値を記入しています。

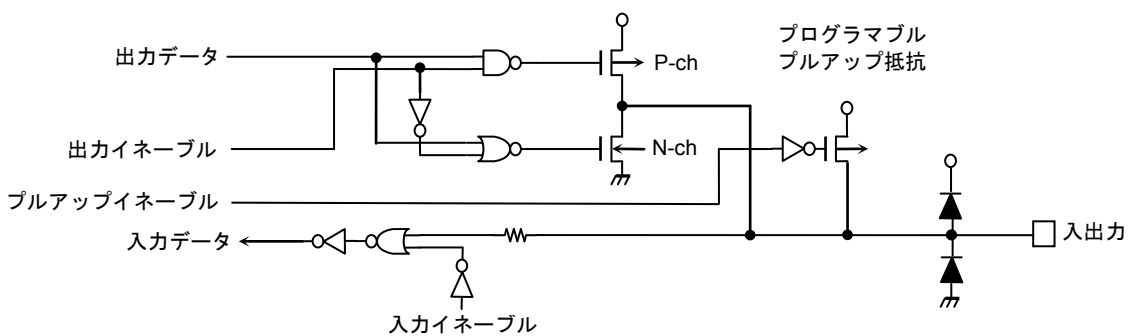
■ PA0, PB1-2, PE1-3, PE5-6, PF1-7, PG0-6, PH0-7, PI6-7, PJ0-3, PJ6-7



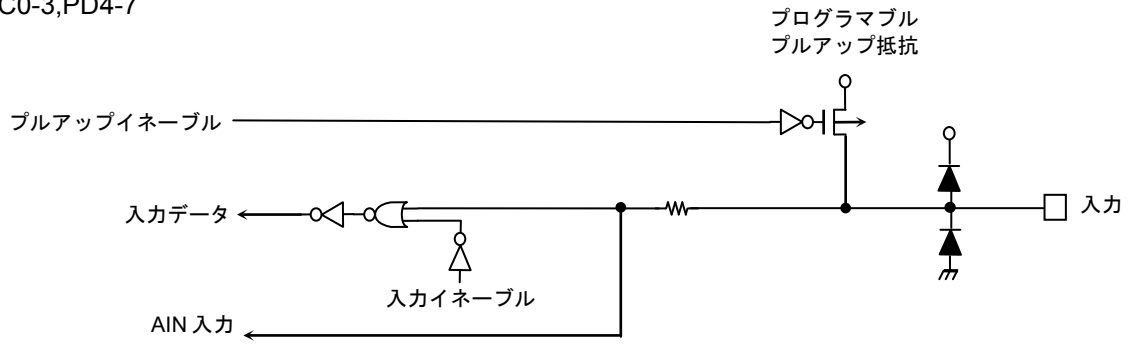
■ PA1



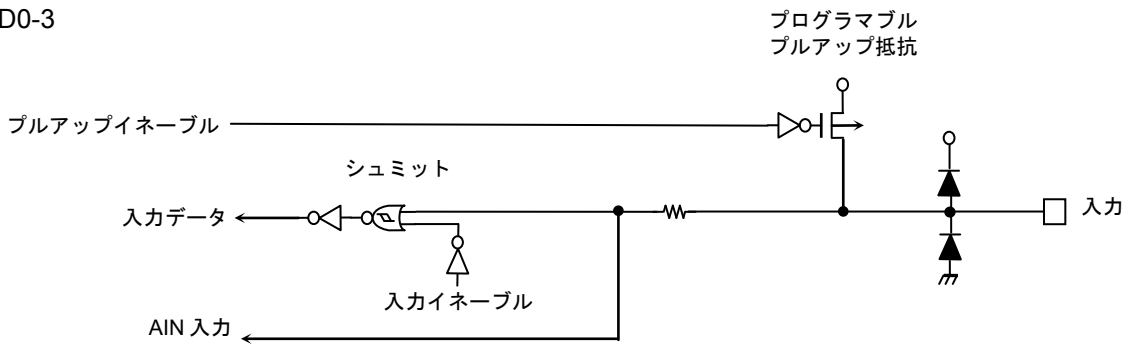
■ PA2-7, PB0, PB3-7, PE0, PE4, PF0, PG7, PI0-5, PJ4-5, PK1-2



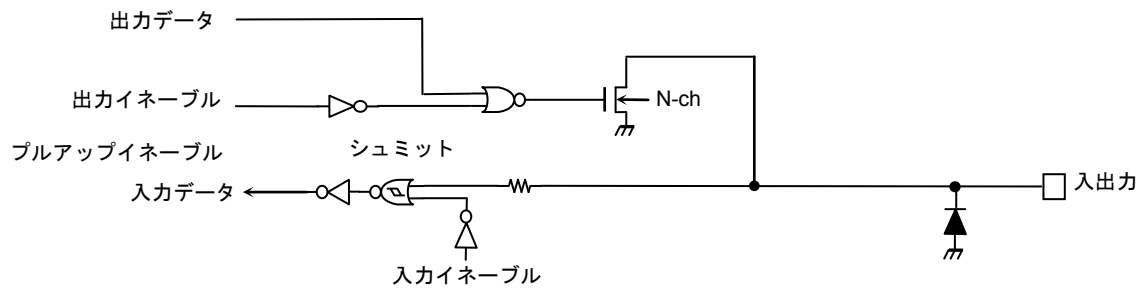
■ PC0-3,PD4-7



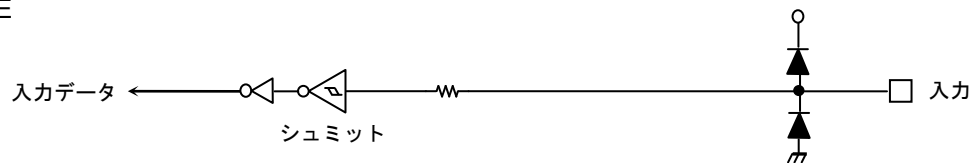
■ PD0-3



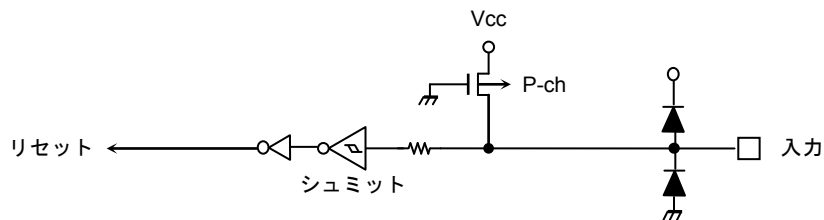
■ PK0



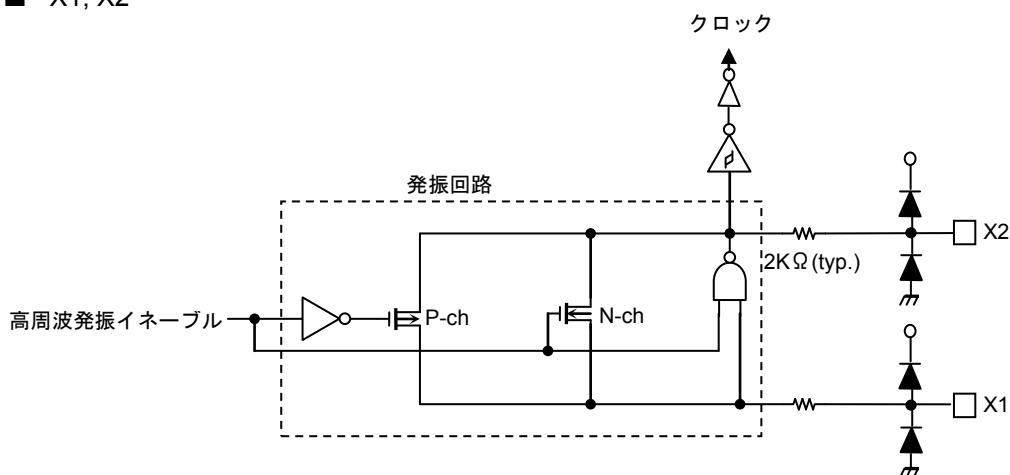
■ $\overline{\text{NMI}}$,MODE



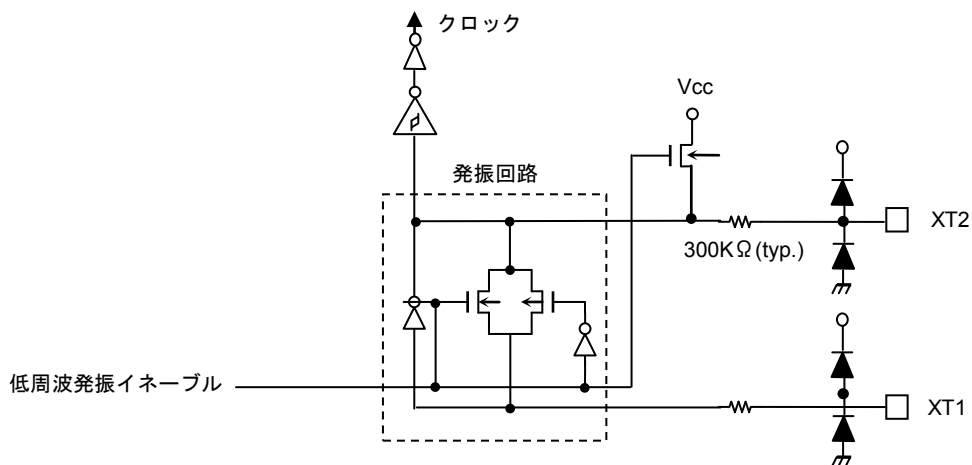
■ $\overline{\text{RESET}}$



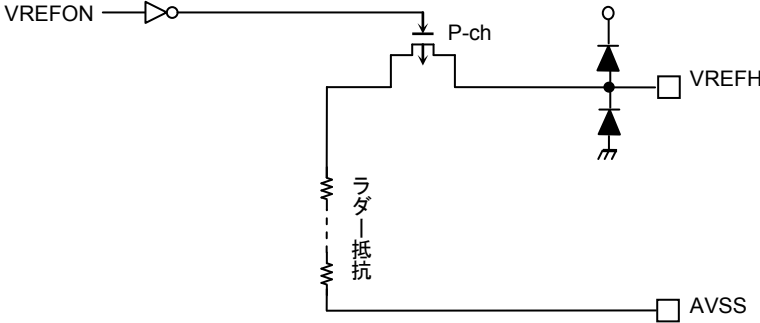
■ X1, X2



■ XT1, XT2



■ VREFH, AVSS

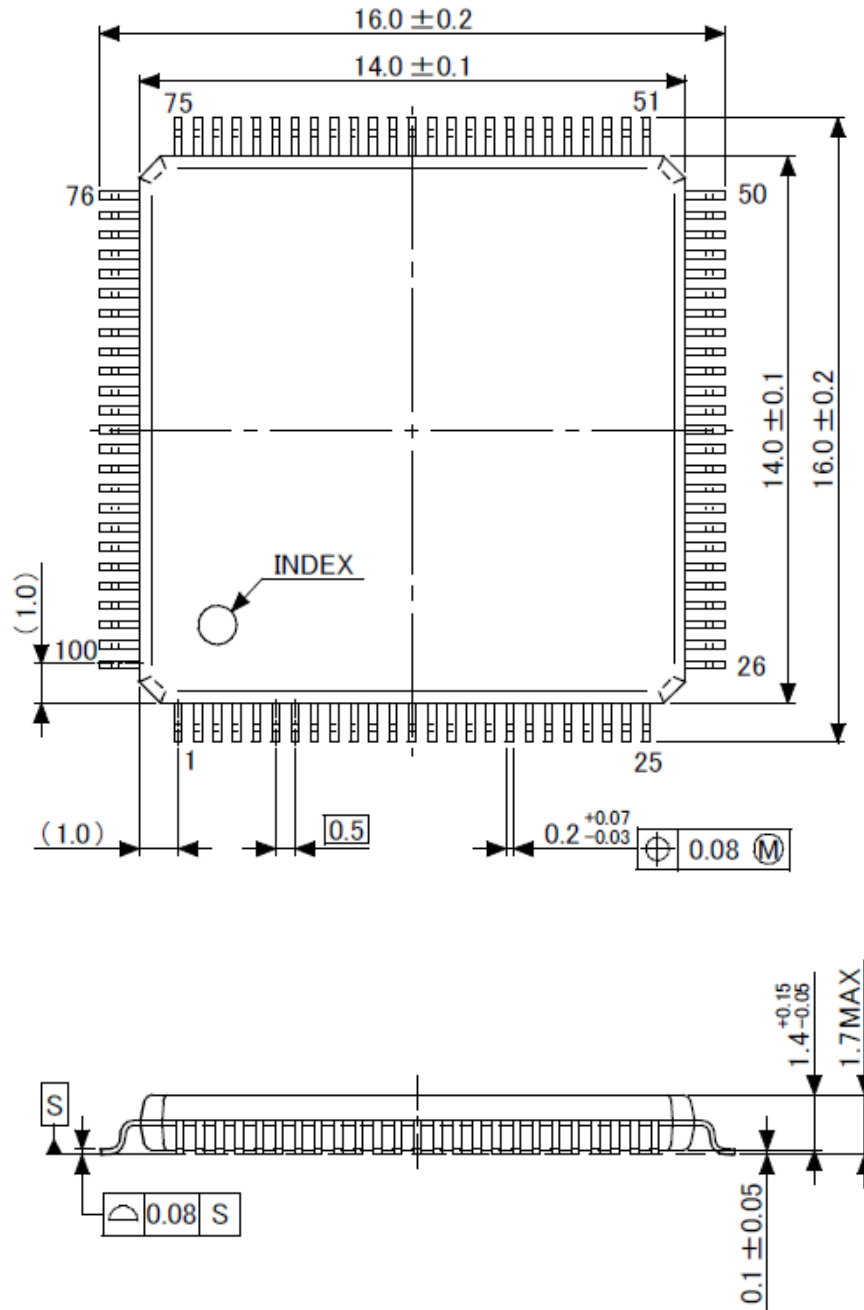


22. パッケージ寸法図

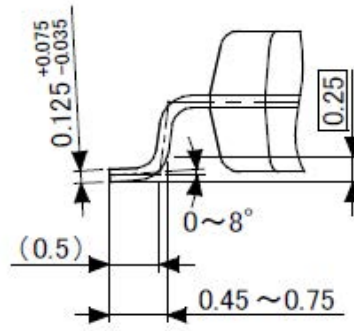
パッケージ型名 : LQFP100-P-1414-0.50H

外形寸法図

Unit : mm



端子先端形状詳細図



製品取り扱い上のお願い

- 本資料に掲載されているハードウェア、ソフトウェアおよびシステム（以下、本製品という）に関する情報等、本資料の掲載内容は、技術の進歩などにより予告なしに変更されることがあります。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。また、文書による当社の事前の承諾を得て本資料を転載複製する場合でも、記載内容に一切変更を加えたり、削除したりしないでください。
- 当社は品質、信頼性の向上に努めていますが、半導体・ストレージ製品は一般に誤作動または故障する場合があります。本製品をご使用頂く場合は、本製品の誤作動や故障により生命・身体・財産が侵害されることのないように、お客様の責任において、お客様のハードウェア・ソフトウェア・システムに必要な安全設計を行うことをお願いします。なお、設計および使用に際しては、本製品に関する最新の情報（本資料、仕様書、データシート、アプリケーションノート、半導体信頼性ハンドブックなど）および本製品が使用される機器の取扱説明書、操作説明書などをご確認の上、これに従ってください。また、上記資料などに記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を使用する場合は、お客様の製品単独およびシステム全体で十分に評価し、お客様の責任において適用可否を判断してください。
- 本製品は、特別に高い品質・信頼性が要求され、またはその故障や誤作動が生命・身体に危害を及ぼす恐れ、膨大な財産損害を引き起こす恐れ、もしくは社会に深刻な影響を及ぼす恐れのある機器（以下“特定用途”という）に使用されることは意図されていませんし、保証もされていません。特定用途には原子力関連機器、航空・宇宙機器、医療機器、車載・輸送機器、列車・船舶機器、交通信号機器、燃焼・爆発制御機器、各種安全関連機器、昇降機器、電力機器、金融関連機器などが含まれますが、本資料に個別に記載する用途は除きます。特定用途に使用された場合には、当社は一切の責任を負いません。なお、詳細は当社営業窓口までお問い合わせください。
- 本製品を分解、解析、リバースエンジニアリング、改造、改変、翻案、複製等しないでください。
- 本製品を、国内外の法令、規則及び命令により、製造、使用、販売を禁止されている製品に使用することはできません。
- 本資料に掲載してある技術情報は、製品の代表的動作・応用を説明するためのもので、その使用に際して当社及び第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。
- 別途、書面による契約またはお客様と当社が合意した仕様書がない限り、当社は、本製品および技術情報に関して、明示的にも黙示的にも一切の保証（機能動作の保証、商品性の保証、特定目的への合致の保証、情報の正確性の保証、第三者の権利の非侵害保証を含むがこれに限らない。）をしておりません。
- 本製品、または本資料に掲載されている技術情報を、大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」、「米国輸出管理規則」等、適用ある輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
- 本製品の RoHS 適合性など、詳細につきましては製品個別に必ず当社営業窓口までお問い合わせください。本製品のご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用ある環境関連法令を十分調査の上、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関して、当社は一切の責任を負いかねます。