

**TOSHIBA**

32 ビット RISC マイクロコントローラ  
TX03 シリーズ

TMPM330FDFG/FYFG/FWFG

株式会社 **東芝** セミコンダクター社





\*\*\*\*\*  
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。  
\*\*\*\*\*



## はじめに(本仕様書での SFR 表記に関する注意点)

各周辺機能回路(IP)には、SFR(Special Function Register)と呼ばれる制御レジスタが準備されています。

メモリマップの章に各 IP の SFR アドレス一覧を記載しており、各 IP の章では SFR の詳細を説明しています。

本仕様書では、SFR に関して以下のルールに従って表現しています。

### a. IP 別 SFR の一覧表(一例)

- 各 IP の章における SFR の一覧表では、レジスタ名称、アドレス、簡単な説明が表現されています。
- すべてのレジスタには、32bit で表現されるユニークなアドレスが割り振られており、各レジスタのアドレスは「Base Address + (固有)アドレス」で表現されています。(一部例外有)

		Base Address = 0x0000_0000
レジスタ名		Address(Base+)
コントロールレジスタ	SAMCR	0x0004
		0x000C

注) SAMCR レジスタのアドレスは 0x0000\_0004 番地「Base Address(0x00000000 番地)+固有アドレス(0x0004 番地)」から 32 ビット分となります。

注) 本レジスタは記述説明用のサンプルです。本マイコンには存在しません。

### b. 各 SFR(レジスタ)の説明

- 各レジスタは、基本的にすべて 32bit のレジスタで構成されています(一部例外有)。
- 各レジスタの説明では、対象ビット、ビットシンボル、タイプ、リセット後の初期値、機能説明が表現されています。

## 1.2.2 SAMCR(コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	MODE	
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	MODE	TDATA						
リセット後	0	0	0	1	0	0	0	0

Bit	Bit Symbol	Type	機能
31-10	-	R	リードすると"0"が読めます。
9-7	MODE[2:0]	R/W	動作モード設定 000 : サンプルモード 0 に設定 001 : サンプルモード 1 に設定 010 : サンプルモード 2 に設定 011 : サンプルモード 3 に設定 上記以外 : Reserved
6-0	TDATA[6:0]	W	送信データ

注) Type は基本的に下記 3 種類となります。

R / W : READ WRITE      読み出し/書き込み可能  
R : READ                  読み出しのみ可能  
W : WRITE                書き込みのみ可能

## c. データ表記について

SFR の説明において使用しているシンボルには以下のようなものがあります。

- ・ x:チャンネル番号/ポート
- ・ n,m:ビット番号

## d. レジスタの表現

説明文においてレジスタを以下のように表現しています。

- ・ レジスタ名<Bit Symbol>  
例: SAMCR<MODE>="000"または SAMCR<MODE[2:0]>="000"  
<MODE[2:0]>はビットシンボル MODE(3 ビット幅)の 2~0 ビット目を意味します。
- ・ レジスタ名[Bit]  
例: SAMCR[9:7]="000"  
レジスタ SAMCR(32 ビット幅)の 9~7 ビット目を意味します。



## 改訂履歴

日付	版	改訂理由
2010/4/24	1	First Release
2010/6/8	2	内容改訂
2010/10/6	3	内容改訂





# 目次

---

---

はじめに(本仕様書での SFR 表記に関する注意点)

---

---

## TMPM330FDFG/FYFG/FWFG

---

---

1.1	機能概要	1
1.2	ブロック図	4
1.3	ピン配置図(Top view)	5
1.4	ピン名称と機能	6
1.4.1	ピン番号順	6
1.4.2	ポート順	11
1.5	電源の種類と供給端子	16

---

---

## 第2章 プロセッサコア

---

---

2.1	コアに関する情報	17
2.2	構成可能なオプション	17
2.3	例外/割り込み	17
2.3.1	割り込み本数	17
2.3.2	割り込み優先度ビット数	18
2.3.3	SysTick	18
2.3.4	SYSRESETREQ	18
2.3.5	LOCKUP	18
2.3.6	補助フォールトステータスレジスタ	18
2.4	イベント	19
2.5	電力管理	19
2.6	排他アクセス	19

---

---

## 第3章 デバッグインタフェース

---

---

3.1	仕様概要	21
3.2	SWJ-DP	21
3.3	ETM	21
3.4	端子情報	22
3.5	ホールドモード中の周辺機能	23
3.6	リセットベクタブレイク	23
3.7	デバッグツールとの接続	23
3.7.1	接続方法	23
3.7.2	デバッグインタフェース端子を汎用ポートとして使用する際の注意	23

---

---

## 第4章 メモリマップ

---

---

<b>4.1</b>	<b>メモリマップ</b> .....	<b>25</b>
4.1.1	TMPM330FDFG メモリマップ.....	26
4.1.2	TMPM330FYFG メモリマップ.....	27
4.1.3	TMPM330FWFG メモリマップ.....	28
<b>4.2</b>	<b>SFR 領域詳細</b> .....	<b>29</b>

---

## 第5章 リセット動作

---

<b>5.1</b>	<b>コールドリセット時</b> .....	<b>31</b>
<b>5.2</b>	<b>ウォームリセット時</b> .....	<b>32</b>
5.2.1	リセット期間.....	32
5.2.2	リセット解除後.....	32

---

## 第6章 クロック/モード制御

---

<b>6.1</b>	<b>特長</b> .....	<b>33</b>
<b>6.2</b>	<b>レジスタ説明</b> .....	<b>34</b>
6.2.1	レジスタ一覧.....	34
6.2.2	CGSYSCR(システムコントロールレジスタ).....	35
6.2.3	CGOSCCR(発振コントロールレジスタ).....	36
6.2.4	CGSTBYCR(スタンバイコントロールレジスタ).....	37
6.2.5	CGPLLSEL(PLL セレクトレジスタ).....	38
6.2.6	CGCKSEL(システムクロックセレクトレジスタ).....	39
<b>6.3</b>	<b>クロック制御</b> .....	<b>40</b>
6.3.1	クロックの種類.....	40
6.3.2	リセット動作による初期値.....	40
6.3.3	クロック系統図.....	41
6.3.4	クロック通倍回路(PLL).....	42
6.3.5	ウォーミングアップ機能.....	42
6.3.6	システムクロック.....	44
6.3.6.1	高速クロック.....	
6.3.6.2	低速クロック.....	
6.3.7	プリスケラクロック.....	45
6.3.8	システムクロックの端子出力機能.....	45
<b>6.4</b>	<b>モードとモード遷移</b> .....	<b>46</b>
6.4.1	モード状態遷移.....	46
<b>6.5</b>	<b>動作モード</b> .....	<b>47</b>
6.5.1	NORMAL モード.....	47
6.5.2	SLOW モード.....	47
<b>6.6</b>	<b>低消費電力モード</b> .....	<b>48</b>
6.6.1	IDLE モード.....	48
6.6.2	SLEEP モード.....	48
6.6.3	STOP モード.....	49
6.6.4	低消費電力モードの選択.....	49
6.6.5	各モードにおける動作状態.....	50
6.6.6	低消費電力モードの解除.....	50
6.6.7	ウォーミングアップ.....	52
6.6.8	モード遷移によるクロック動作.....	53
6.6.8.1	NORMAL → STOP → NORMAL 動作モード遷移.....	
6.6.8.2	NORMAL → SLEEP → NORMAL 動作モード遷移.....	
6.6.8.3	SLOW → STOP → SLOW 動作モード遷移.....	
6.6.8.4	SLOW → SLEEP → SLOW 動作モード遷移.....	

---

## 第7章 例外

---

<b>7.1</b>	<b>概要</b> .....	<b>55</b>
7.1.1	種類.....	55

7.1.2	処理の流れ.....	56
7.1.2.1	例外要求と検出	
7.1.2.2	例外の処理と割り込み処理ルーチンへの分岐(横取り)	
7.1.2.3	割り込み処理ルーチンの発行	
7.1.2.4	例外からの復帰	
<b>7.2</b>	<b>リセット例外.....</b>	<b>62</b>
<b>7.3</b>	<b>マスク不能割り込み(NMI).....</b>	<b>62</b>
<b>7.4</b>	<b>SysTick.....</b>	<b>63</b>
<b>7.5</b>	<b>割り込み.....</b>	<b>64</b>
7.5.1	要因.....	64
7.5.1.1	経路	
7.5.1.2	割り込み要因の発生	
7.5.1.3	割り込み要因の伝達	
7.5.1.4	外部割り込み端子を使用する際の注意	
7.5.1.5	要因一覧	
7.5.1.6	アクティブレベル	
7.5.2	処理詳細.....	68
7.5.2.1	処理の流れ	
7.5.2.2	準備	
7.5.2.3	検出(クロックジェネレータ)	
7.5.2.4	検出(CPU)	
7.5.2.5	CPUの処理	
7.5.2.6	割り込み処理ルーチンでの処理(要因の取り下げ)	
<b>7.6</b>	<b>例外/割り込み関連レジスタ.....</b>	<b>74</b>
7.6.1	レジスタ一覧.....	74
7.6.2	NVIC レジスタ.....	75
7.6.2.1	SysTick 制御およびステータスレジスタ	
7.6.2.2	SysTick リロード値レジスタ	
7.6.2.3	SysTick 現在値レジスタ	
7.6.2.4	SysTick 較正值レジスタ	
7.6.2.5	割り込みイネーブルセットレジスタ 1	
7.6.2.6	割り込みイネーブルセットレジスタ 2	
7.6.2.7	割り込みイネーブルクリアレジスタ 1	
7.6.2.8	割り込みイネーブルクリアレジスタ 2	
7.6.2.9	割り込み保留セットレジスタ 1	
7.6.2.10	割り込み保留セットレジスタ 2	
7.6.2.11	割り込み保留クリアレジスタ 1	
7.6.2.12	割り込み保留クリアレジスタ 2	
7.6.2.13	割り込み優先度レジスタ	
7.6.2.14	ベクタテーブルオフセットレジスタ	
7.6.2.15	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	
7.6.2.16	システムハンドラ優先度レジスタ	
7.6.2.17	システムハンドラ制御および状態レジスタ	
7.6.3	クロックジェネレータレジスタ.....	93
7.6.3.1	CGIMCGA(CG 割り込みモードコントロールレジスタ A)	
7.6.3.2	CGIMCGB(CG 割り込みモードコントロールレジスタ B)	
7.6.3.3	CGIMCGC(CG 割り込みモードコントロールレジスタ C)	
7.6.3.4	CGIMCGD(CG 割り込みモードコントロールレジスタ D)	
7.6.3.5	CGICRCG(CG 割り込み要求クリアレジスタ)	
7.6.3.6	CGNMIFLG(NMI フラグレジスタ)	
7.6.3.7	CGRSTFLG(リセットフラグレジスタ)	

---

## 第 8 章 入出力ポート

---

<b>8.1</b>	<b>ポート機能.....</b>	<b>103</b>
8.1.1	機能一覧.....	103
8.1.2	ポートレジスタ概略説明.....	106
8.1.3	STOP モード中のポート状態.....	107
8.1.4	STOP/SLEEP モード遷移の際の注意.....	107
<b>8.2</b>	<b>ポート機能詳細.....</b>	<b>108</b>
8.2.1	ポート A (PA0-PA7).....	108
8.2.1.1	ポート A 回路タイプ	
8.2.1.2	ポート A レジスタ一覧	
8.2.1.3	PADATA (ポート A データレジスタ)	
8.2.1.4	PACR (ポート A 出力コントロールレジスタ)	
8.2.1.5	PAFRI (ポート A ファンクションレジスタ 1)	
8.2.1.6	PAPUP (ポート A ブルアップコントロールレジスタ)	

8.2.1.7	PAPDN (ポート A ブルダウンコントロールレジスタ)	
8.2.1.8	PAIE (ポート A 入力コントロールレジスタ)	
8.2.2	ポート B (PB0~PB7).....	113
8.2.2.1	ポート B 回路タイプ	
8.2.2.2	ポート B レジスタ一覧	
8.2.2.3	PBDATA (ポート B データレジスタ)	
8.2.2.4	PBCR (ポート B 出力コントロールレジスタ)	
8.2.2.5	PBFR1 (ポート B ファンクションレジスタ 1)	
8.2.2.6	PBPUP (ポート B ブルアップコントロールレジスタ)	
8.2.2.7	PBIE (ポート B 入力コントロールレジスタ)	
8.2.3	ポート C (PC0~PC3).....	117
8.2.3.1	ポート C 回路タイプ	
8.2.3.2	ポート C レジスタ一覧	
8.2.3.3	PCDATA (ポート C データレジスタ)	
8.2.3.4	PCPUP (ポート C ブルアップコントロールレジスタ)	
8.2.3.5	PCIE (ポート C 入力コントロールレジスタ)	
8.2.4	ポート D (PD0~PD7).....	120
8.2.4.1	ポート D 回路タイプ	
8.2.4.2	ポート D レジスタ一覧	
8.2.4.3	PDDATA (ポート D データレジスタ)	
8.2.4.4	PDFR1 (ポート D ファンクションレジスタ 1)	
8.2.4.5	PDPUP (ポート D ブルアップコントロールレジスタ)	
8.2.4.6	PDIE (ポート D 入力コントロールレジスタ)	
8.2.5	ポート E (PE0~PE6).....	123
8.2.5.1	ポート E 回路タイプ	
8.2.5.2	ポート E レジスタ一覧	
8.2.5.3	PEDATA (ポート E データレジスタ)	
8.2.5.4	PECR (ポート E 出力コントロールレジスタ)	
8.2.5.5	PEFR1 (ポート E ファンクションレジスタ 1)	
8.2.5.6	PEFR2 (ポート E ファンクションレジスタ 2)	
8.2.5.7	PEOD (ポート E オープンドレインコントロールレジスタ)	
8.2.5.8	PEPUP (ポート E ブルアップコントロールレジスタ)	
8.2.5.9	PEIE (ポート E 入力コントロールレジスタ)	
8.2.6	ポート F (PF0~PF7).....	128
8.2.6.1	ポート F 回路タイプ	
8.2.6.2	ポート F レジスタ一覧	
8.2.6.3	PFDATA (ポート F データレジスタ)	
8.2.6.4	PFCR (ポート F 出力コントロールレジスタ)	
8.2.6.5	PFFR1 (ポート F ファンクションレジスタ 1)	
8.2.6.6	PFFR2 (ポート F ファンクションレジスタ 2)	
8.2.6.7	PFOD (ポート F オープンドレインコントロールレジスタ)	
8.2.6.8	PFUPUP (ポート F ブルアップコントロールレジスタ)	
8.2.6.9	PFIE (ポート F 入力コントロールレジスタ)	
8.2.7	ポート G (PG0~PG7).....	133
8.2.7.1	ポート G 回路タイプ	
8.2.7.2	ポート G レジスタ一覧	
8.2.7.3	PGDATA (ポート G データレジスタ)	
8.2.7.4	PGCR (ポート G 出力コントロールレジスタ)	
8.2.7.5	PGFR1 (ポート G ファンクションレジスタ 1)	
8.2.7.6	PGOD (ポート G オープンドレインコントロールレジスタ)	
8.2.7.7	PGPUP (ポート G ブルアップコントロールレジスタ)	
8.2.7.8	PGIE (ポート G 入力コントロールレジスタ)	
8.2.8	ポート H (PH0~PH7).....	138
8.2.8.1	ポート H 回路タイプ	
8.2.8.2	ポート H レジスタ一覧	
8.2.8.3	PHDATA (ポート H データレジスタ)	
8.2.8.4	PHCR (ポート H 出力コントロールレジスタ)	
8.2.8.5	PHFR1 (ポート H ファンクションレジスタ 1)	
8.2.8.6	PHPUP (ポート H ブルアップコントロールレジスタ)	
8.2.8.7	PHIE (ポート H 入力コントロールレジスタ)	
8.2.9	ポート I (PI0~PI7).....	142
8.2.9.1	ポート I 回路タイプ	
8.2.9.2	ポート I レジスタ一覧	
8.2.9.3	PIDATA (ポート I データレジスタ)	
8.2.9.4	PICR (ポート I 出力コントロールレジスタ)	
8.2.9.5	PIFR1 (ポート I ファンクションレジスタ 1)	
8.2.9.6	PIPUP (ポート I ブルアップコントロールレジスタ)	
8.2.9.7	PIIE (ポート I 入力コントロールレジスタ)	
8.2.10	ポート J (PJ0~PJ7).....	146
8.2.10.1	ポート J 回路タイプ	
8.2.10.2	ポート J レジスタ一覧	
8.2.10.3	PJDATA (ポート J データレジスタ)	
8.2.10.4	PJCR (ポート J 出力コントロールレジスタ)	
8.2.10.5	PJFR1 (ポート J ファンクションレジスタ 1)	

8.2.10.6	PJPUP(ポートJプルアップコントロールレジスタ)	
8.2.10.7	PJIE(ポートJ入力コントロールレジスタ)	
8.2.11	ポート K (PK0~PK2)	150
8.2.11.1	ポート K 回路タイプ	
8.2.11.2	ポート K レジスタ一覧	
8.2.11.3	PKDATA(ポート K データレジスタ)	
8.2.11.4	PKCR(ポート K 出力コントロールレジスタ)	
8.2.11.5	PKFR1(ポート K ファンクションレジスタ 1)	
8.2.11.6	PKFR2(ポート K ファンクションレジスタ 2)	
8.2.11.7	PKPUP(ポート K プルアップコントロールレジスタ)	
8.2.11.8	PKIE(ポート K 入力コントロールレジスタ)	
<b>8.3</b>	<b>ポート回路図</b>	<b>154</b>
8.3.1	ポートタイプ一覧	154
8.3.2	タイプ T1	155
8.3.3	タイプ T2	156
8.3.4	タイプ T3	157
8.3.5	タイプ T4	158
8.3.6	タイプ T5	159
8.3.7	タイプ T6	160
8.3.8	タイプ T7	161
8.3.9	タイプ T8	162
8.3.10	タイプ T9	163
8.3.11	タイプ T10	164
8.3.12	タイプ T11	165
8.3.13	タイプ T12	166
8.3.14	タイプ T13	167
8.3.15	タイプ T14	168
8.3.16	タイプ T15	169
8.3.17	タイプ T16	170
8.3.18	タイプ T17	171
8.3.19	タイプ T18	172
<b>8.4</b>	<b>付録 (ポート設定一覧)</b>	<b>173</b>
8.4.1	ポート A 設定	173
8.4.2	ポート B 設定	174
8.4.3	ポート C 設定	175
8.4.4	ポート D 設定	175
8.4.5	ポート E 設定	176
8.4.6	ポート F 設定	177
8.4.7	ポート G 設定	178
8.4.8	ポート H 設定	179
8.4.9	ポート I 設定	180
8.4.10	ポート J 設定	181
8.4.11	ポート K 設定	182

---

## 第9章 16ビットタイマ/イベントカウンタ(TMRB)

---

<b>9.1</b>	<b>概要</b>	<b>183</b>
<b>9.2</b>	<b>チャンネル別仕様相違点</b>	<b>184</b>
<b>9.3</b>	<b>構成</b>	<b>185</b>
<b>9.4</b>	<b>レジスタ説明</b>	<b>186</b>
9.4.1	チャンネル別レジスタ一覧	186
9.4.2	TBxEN(イネーブルレジスタ)	187
9.4.3	TBxRUN(RUN レジスタ)	188
9.4.4	TBxCR(コントロールレジスタ)	189
9.4.5	TBxMOD(モードレジスタ)	190
9.4.6	TBxFFCR(フリップフロップコントロールレジスタ)	191
9.4.7	TBxST(ステータスレジスタ)	192
9.4.8	TBxIM(割り込みマスクレジスタ)	193
9.4.9	TBxUC(アップカウンタキャプチャレジスタ)	193
9.4.10	TBxRG0(タイマレジスタ 0)	194
9.4.11	TBxRG1(タイマレジスタ 1)	194
9.4.12	TBxCP0(キャプチャレジスタ 0)	195
9.4.13	TBxCPI(キャプチャレジスタ 1)	195

<b>9.5</b>	<b>回路別の動作説明</b> .....	196
9.5.1	プリスケーラ.....	196
9.5.2	アップカウンタ(UC).....	200
9.5.3	タイマレジスタ(TBxRG0, TBxRG1).....	200
9.5.4	キャプチャ制御.....	201
9.5.5	キャプチャレジスタ(TBxCP0, TBxCP1).....	201
9.5.6	アップカウンタキャプチャレジスタ(TBxUC).....	201
9.5.7	コンパレータ(CP0, CP1).....	201
9.5.8	タイマフリップフロップ(TBxFF0).....	201
9.5.9	キャプチャ割り込み(INTCAPx0, INTCAPx1).....	201
<b>9.6</b>	<b>モード別動作説明</b> .....	202
9.6.1	16ビットインタバルタイマモード.....	202
9.6.2	16ビットイベントカウンタモード.....	202
9.6.3	16ビットPPG(プログラマブル矩形波)出力モード.....	203
9.6.4	タイマ同期モード.....	205
<b>9.7</b>	<b>キャプチャ機能を利用した応用例</b> .....	206
9.7.1	外部トリガパルスからのワンショットパルス出力.....	206
9.7.2	周波数測定.....	208
9.7.3	パルス幅測定.....	208
9.7.4	時間差測定.....	209

---

## 第10章 シリアルチャネル(SIO/UART)

---

<b>10.1</b>	<b>概要</b> .....	211
<b>10.2</b>	<b>チャネル別仕様相違点</b> .....	211
<b>10.3</b>	<b>構成</b> .....	212
<b>10.4</b>	<b>レジスタ説明</b> .....	213
10.4.1	チャネル別レジスタ一覧.....	213
10.4.2	SCxEN (イネーブルレジスタ).....	214
10.4.3	SCxBUF (バッファレジスタ).....	215
10.4.4	SCxCR (コントロールレジスタ).....	216
10.4.5	SCxMOD0 (モードコントロールレジスタ 0).....	217
10.4.6	SCxMOD1 (モードコントロールレジスタ 1).....	218
10.4.7	SCxMOD2 (モードコントロールレジスタ 2).....	219
10.4.8	SCxBRCR (ボーレートジェネレータコントロールレジスタ), SCxBRADD (ボーレートジェネレータコントロールレジスタ 2).....	221
10.4.9	SCxFCNF (FIFO コンフィグレジスタ).....	223
10.4.10	SCxRFC (受信 FIFO コンフィグレジスタ).....	225
10.4.11	SCxTFC (送信 FIFO コンフィグレジスタ)(注 2).....	226
10.4.12	SCxRST (受信 FIFO ステータスレジスタ).....	227
10.4.13	SCxTST (送信 FIFO ステータスレジスタ).....	228
<b>10.5</b>	<b>動作モード</b> .....	229
<b>10.6</b>	<b>データフォーマット</b> .....	230
10.6.1	データフォーマット一覧.....	230
10.6.2	パリティ制御.....	231
10.6.2.1	送信.....	
10.6.2.2	受信.....	
10.6.3	STOP ビット長.....	231
<b>10.7</b>	<b>クロック制御</b> .....	232
10.7.1	プリスケーラ.....	232
10.7.2	シリアルクロック生成回路.....	236
10.7.2.1	ボーレートジェネレータ.....	
10.7.2.2	クロック選択回路.....	
<b>10.8</b>	<b>送信/受信バッファと FIFO</b> .....	240
10.8.1	構成.....	240
10.8.2	送信/受信バッファ.....	240
10.8.3	FIFO.....	240
<b>10.9</b>	<b>ステータスフラグ</b> .....	241
<b>10.10</b>	<b>エラーフラグ</b> .....	241
10.10.1	OERR フラグ.....	242

10.10.2	PERR フラグ	242
10.10.3	FERR フラグ	242
<b>10.11</b>	<b>受信</b>	<b>243</b>
10.11.1	受信カウンタ	243
10.11.2	受信制御部	243
10.11.2.1	IO インタフェースモードの場合	
10.11.2.2	UART モードの場合	
10.11.3	受信動作	243
10.11.3.1	受信バッファの動作	
10.11.3.2	受信 FIFO の動作	
10.11.3.3	IO インタフェースモード、SCLK 出力での受信	
10.11.3.4	受信データの読み出し	
10.11.3.5	ウェイクアップ機能	
10.11.3.6	オーバランエラー	
<b>10.12</b>	<b>送信</b>	<b>248</b>
10.12.1	送信カウンタ	248
10.12.2	送信制御部	248
10.12.2.1	IO インタフェースモードの場合	
10.12.2.2	UART モードの場合	
10.12.3	送信動作	249
10.12.3.1	送信バッファの動作	
10.12.3.2	送信 FIFO の動作	
10.12.3.3	IO インタフェースモード、SCLK 出力での送信	
10.12.3.4	アンダーランエラー	
<b>10.13</b>	<b>ハンドシェイク機能</b>	<b>252</b>
<b>10.14</b>	<b>割り込み/エラー発生タイミング</b>	<b>253</b>
10.14.1	受信割り込み	253
10.14.1.1	シングルバッファ/ダブルバッファ構成の場合	
10.14.1.2	FIFO 使用の場合	
10.14.2	送信割り込み	254
10.14.2.1	シングルバッファ/ダブルバッファ構成の場合	
10.14.2.2	FIFO 使用の場合	
10.14.3	エラー発生	255
10.14.3.1	UART モード	
10.14.3.2	IO インタフェースモード	
<b>10.15</b>	<b>ソフトリセット</b>	<b>255</b>
<b>10.16</b>	<b>モード別動作説明</b>	<b>256</b>
10.16.1	モード 0 (IO インタフェースモード)	256
10.16.1.1	送信	
10.16.1.2	受信	
10.16.1.3	送受信(全二重)	
10.16.2	モード 1 (7 ビット UART モード)	267
10.16.3	モード 2 (8 ビット UART モード)	267
10.16.4	モード 3 (9 ビット UART モード)	268
10.16.4.1	ウェイクアップ機能	
10.16.4.2	プロトコル	

---

## 第 11 章 シリアルバスインタフェース(I2C/SIO)

---

<b>11.1</b>	<b>構成</b>	<b>272</b>
<b>11.2</b>	<b>レジスタ説明</b>	<b>273</b>
11.2.1	チャンネル別レジスタ一覧	273
<b>11.3</b>	<b>I2C バスモード時のデータフォーマット</b>	<b>274</b>
<b>11.4</b>	<b>I2C バスモード時のコントロールレジスタ</b>	<b>275</b>
11.4.1	SBIxCR0(コントロールレジスタ 0)	275
11.4.2	SBIxCRI(コントロールレジスタ 1)	276
11.4.3	SBIxCR2(コントロールレジスタ 2)	278
11.4.4	SBIxSR(ステータスレジスタ)	279
11.4.5	SBIxBR0(ボーレートレジスタ 0)	280
11.4.6	SBIxDBR(データバッファレジスタ)	281
11.4.7	SBIxI2CAR(I2C バスアドレスレジスタ)	282
<b>11.5</b>	<b>I2C バスモード時の制御</b>	<b>283</b>
11.5.1	シリアルクロック	283

11.5.1.1	クロックソース	
11.5.1.2	クロック同期化	
11.5.2	アクノリッジメントモードの指定	284
11.5.3	転送ビット数の選択	284
11.5.4	スレーブアドレスとアドレス認識モードの設定	284
11.5.5	動作モード	285
11.5.6	トランスミッタ/レシーバの選択	285
11.5.7	マスタ/スレーブの選択	285
11.5.8	スタート/ストップコンディションの発生	286
11.5.9	割り込みサービス要求と解除	286
11.5.10	アービトレーションロスト検出モニタ	287
11.5.11	スレーブアドレス一致検出モニタ	288
11.5.12	ゼネラルコール検出モニタ	288
11.5.13	最終受信ビットモニタ	289
11.5.14	データバッファレジスタ(SBIxDBR)	289
11.5.15	ボーレートレジスタ(SBIxBR0)	289
11.5.16	ソフトウェアリセット	289
<b>11.6</b>	<b>I2C バスモード時のデータ転送手順</b>	<b>290</b>
11.6.1	デバイスの初期化	290
11.6.2	スタートコンディション, スレーブアドレスの発生	290
11.6.2.1	マスタモードの場合	
11.6.2.2	スレーブモードの場合	
11.6.3	1ワードのデータ転送	292
11.6.3.1	マスタモードの場合(<MST>="1")	
11.6.3.2	スレーブモードの場合(<MST>="0")	
11.6.4	ストップコンディションの発生	296
11.6.5	再スタートの手順	296
<b>11.7</b>	<b>SIO モード時のコントロールレジスタ</b>	<b>298</b>
11.7.1	SBIxCR0(コントロールレジスタ 0)	298
11.7.2	SBIxCR1(コントロールレジスタ 1)	299
11.7.3	SBIxDBR(データバッファレジスタ)	300
11.7.4	SBIxCR2(コントロールレジスタ 2)	301
11.7.5	SBIxSR(ステータスレジスタ)	302
11.7.6	SBIxBR0(ボーレートレジスタ 0)	303
<b>11.8</b>	<b>SIO モード時の制御</b>	<b>304</b>
11.8.1	シリアルクロック	304
11.8.1.1	クロックソース	
11.8.1.2	シフトエッジ	
11.8.2	転送モード	306
11.8.2.1	8ビット送信モード	
11.8.2.2	8ビット受信モード	
11.8.2.3	8ビット送受信モード	
11.8.2.4	送信終了時の最終ビット保持時間	

---

## 第12章 CEC 機能

---

<b>12.1</b>	<b>概要</b>	<b>311</b>
12.1.1	受信	311
12.1.2	送信	311
12.1.3	注意事項	311
<b>12.2</b>	<b>ブロック図</b>	<b>312</b>
<b>12.3</b>	<b>レジスタ説明</b>	<b>313</b>
12.3.1	レジスタ一覧	313
12.3.2	CECEN(CEC イネーブルレジスタ)	314
12.3.3	CECADD(ロジカルアドレスレジスタ)	315
12.3.4	CECRESET(ソフトウェアリセットレジスタ)	316
12.3.5	CECREN(受信イネーブルレジスタ)	316
12.3.6	CECRBUF(受信バッファレジスタ)	317
12.3.7	CECR1(受信コントロールレジスタ 1)	318
12.3.8	CECR2(受信コントロールレジスタ 2)	320
12.3.9	CECR3(受信コントロールレジスタ 3)	322
12.3.10	CECTEN(送信イネーブルレジスタ)	324
12.3.11	CECTBUF(送信バッファレジスタ)	325



12.3.12	CECTCR(送信コントロールレジスタ)	326
12.3.13	CECRSTAT(受信割り込みステータスレジスタ)	328
12.3.14	CECTSTAT(送信割り込みステータスレジスタ)	329
<b>12.4</b>	<b>動作説明</b>	<b>330</b>
12.4.1	サンプリングクロック	330
12.4.2	受信	330
12.4.2.1	基本動作	
12.4.2.2	受信の準備	
12.4.2.3	受信許可	
12.4.2.4	受信エラー検出	
12.4.2.5	受信エラー詳細	
12.4.2.6	受信の停止	
12.4.3	送信	339
12.4.3.1	基本動作	
12.4.3.2	送信の準備	
12.4.3.3	送信エラー検出	
12.4.3.4	送信エラー詳細	
12.4.3.5	送信の停止	
12.4.3.6	再送信	
12.4.4	ソフトウェアリセット	344

---

## 第13章 リモコン判定機能(RMC)

---

<b>13.1</b>	<b>概要</b>	<b>345</b>
13.1.1	リモコン受信	345
<b>13.2</b>	<b>ブロック図</b>	<b>345</b>
<b>13.3</b>	<b>レジスタ説明</b>	<b>346</b>
13.3.1	レジスタ一覧	346
13.3.2	RMCxEN(イネーブルレジスタ)	347
13.3.3	RMCxREN(受信イネーブルレジスタ)	348
13.3.4	RMCxRBUF1(受信データバッファレジスタ 1)	349
13.3.5	RMCxRBUF2(受信データバッファレジスタ 2)	349
13.3.6	RMCxRBUF3(受信データバッファレジスタ 3)	350
13.3.7	RMCxRCR1(受信コントロールレジスタ 1)	351
13.3.8	RMCxRCR2(受信コントロールレジスタ 2)	352
13.3.9	RMCxRCR3(受信コントロールレジスタ 3)	353
13.3.10	RMCxRCR4(受信コントロールレジスタ 4)	354
13.3.11	RMCxRSTAT(受信ステータスレジスタ)	355
<b>13.4</b>	<b>動作説明</b>	<b>356</b>
13.4.1	リモコン受信	356
13.4.1.1	サンプリングブロック	
13.4.1.2	基本動作	
13.4.1.3	リモコン受信の準備	
13.4.1.4	受信許可	
13.4.1.5	受信の停止	
13.4.1.6	リーダ待ちの状態でのリーダなしのリモコン信号の受信	
13.4.1.7	Low 幅のみのリーダで始まるリモコン信号の受信	
13.4.1.8	周期固定の位相方式のリモコン信号の受信	

---

## 第14章 アナログ/デジタルコンバータ(ADC)

---

<b>14.1</b>	<b>概要</b>	<b>365</b>
<b>14.2</b>	<b>構成</b>	<b>366</b>
<b>14.3</b>	<b>レジスタ説明</b>	<b>367</b>
14.3.1	レジスタ一覧	367
14.3.2	ADCBAS(変換精度設定レジスタ)	368
14.3.3	ADCLK(変換クロック設定レジスタ)	369
14.3.4	ADM0D0(モードコントロールレジスタ 0)	371
14.3.5	ADM0D1(モードコントロールレジスタ 1)	372
14.3.6	ADM0D2(モードコントロールレジスタ 2)	374
14.3.7	ADM0D4(モードコントロールレジスタ 4)	375

14.3.8	ADMOD3 (AD モードコントロールレジスタ 3).....	376
14.3.9	ADMOD5 (AD モードコントロールレジスタ 5).....	377
14.3.10	ADREG08 (変換結果レジスタ 08).....	378
14.3.11	ADREG19 (変換結果レジスタ 19).....	379
14.3.12	ADREG2A (変換結果レジスタ 2A).....	380
14.3.13	ADREG3B (変換結果レジスタ 3B).....	381
14.3.14	ADREG4C (変換結果レジスタ 4C).....	382
14.3.15	ADREG5D (変換結果レジスタ 5D).....	383
14.3.16	ADREG6E (変換結果レジスタ 6E).....	384
14.3.17	ADREG7F (変換結果レジスタ 7F).....	385
14.3.18	ADREGSP (変換結果レジスタ SP).....	386
14.3.19	ADCMP0 (変換結果比較レジスタ 0).....	387
14.3.20	ADCMP1 (変換結果比較レジスタ 1).....	387
<b>14.4</b>	<b>動作説明</b> .....	<b>388</b>
14.4.1	アナログ基準電圧.....	388
14.4.2	AD 変換モード.....	388
14.4.2.1	通常 AD 変換.....	
14.4.2.2	最優先 AD 変換.....	
14.4.3	AD 監視機能.....	389
14.4.4	入力チャネルの選択.....	390
14.4.5	AD 変換動作詳細.....	390
14.4.5.1	AD 変換の起動.....	
14.4.5.2	AD 変換動作.....	
14.4.5.3	通常 AD 変換中の最優先変換要求.....	
14.4.5.4	リビート変換モードの停止.....	
14.4.5.5	通常 AD 変換の再起動.....	
14.4.5.6	変換終了.....	
14.4.5.7	割り込み発生タイミングと変換結果格納レジスタ.....	

---

## 第 15 章 ウォッチドッグタイマ(WDT)

---

<b>15.1</b>	<b>構成</b> .....	<b>397</b>
<b>15.2</b>	<b>レジスタ一覧</b> .....	<b>398</b>
15.2.1	WDMOD(ウォッチドッグタイマモードレジスタ).....	398
15.2.2	WDCR(ウォッチドッグタイマコントロールレジスタ).....	399
<b>15.3</b>	<b>動作説明</b> .....	<b>400</b>
15.3.1	基本動作.....	400
15.3.2	動作モードと動作状態.....	400
<b>15.4</b>	<b>暴走検出時の動作</b> .....	<b>401</b>
15.4.1	INTWDT 割り込み発生の場合.....	401
15.4.2	内部リセット発生の場合.....	402
<b>15.5</b>	<b>コントロールレジスタ</b> .....	<b>403</b>
15.5.1	ウォッチドッグタイマモードレジスタ(WDMOD).....	403
15.5.2	ウォッチドッグタイマコントロールレジスタ(WDCR).....	403
15.5.3	設定例.....	404
15.5.3.1	ディセーブル制御.....	
15.5.3.2	イネーブル制御.....	
15.5.3.3	ウォッチドッグタイマのクリア制御.....	
15.5.3.4	ウォッチドッグタイマ検出時間の設定.....	

---

## 第 16 章 リアルタイムクロック(RTC)

---

<b>16.1</b>	<b>RTC の機能概略</b> .....	<b>405</b>
<b>16.2</b>	<b>ブロック図</b> .....	<b>405</b>
<b>16.3</b>	<b>レジスタ説明</b> .....	<b>406</b>
16.3.1	レジスタ一覧.....	406
16.3.2	コントロールレジスタ.....	406
16.3.3	レジスタ詳細.....	408
16.3.3.1	RTCSECR(秒桁レジスタ(PAGE0 のみ)).....	
16.3.3.2	RTCMINR(分析レジスタ(PAGE0/1)).....	
16.3.3.3	RTCHOURR(時間桁レジスタ(PAGE0/1)).....	

16.3.3.4	RTCDAYR(曜日桁レジスタ(PAGE0/1))	
16.3.3.5	RTCDATER(日桁レジスタ(PAGE0/1))	
16.3.3.6	RTCMONTHR(月桁レジスタ(PAGE0のみ))	
16.3.3.7	RTCMONTHR(24時間時計, 12時間時計の選択レジスタ(PAGE1のみ))	
16.3.3.8	RTCYEARR(年桁レジスタ(PAGE0のみ))	
16.3.3.9	RTCYEARR(うるう年レジスタ(PAGE1のみ))	
16.3.3.10	RTCPAGER(PAGEレジスタ(PAGE0/1))	
16.3.3.11	RTCRESTR(リセットレジスタ(PAGE0/1))	
<b>16.4</b>	<b>動作説明</b>	<b>415</b>
16.4.1	時計データをリードする場合	415
16.4.2	時計データをライトする場合	415
16.4.3	低消費電力モードへ遷移する場合	417
<b>16.5</b>	<b>アラーム機能の説明</b>	<b>418</b>
16.5.1	アラームレジスタと時計の一致時、ALARM端子からパルスを出力	418
16.5.2	1 Hz のパルスを出力する場合	419
16.5.3	16 Hz のパルスを出力する場合	419

---

## 第17章 Flash 動作説明

---

<b>17.1</b>	<b>フラッシュメモリ</b>	<b>421</b>
17.1.1	特長	421
17.1.2	フラッシュ部ブロック図	423
<b>17.2</b>	<b>動作モード</b>	<b>424</b>
17.2.1	リセット動作	425
17.2.2	ユーザーブートモード(シングルチップモード)	425
17.2.2.1	(1-A)書き替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	
17.2.2.2	(1-B)書き替えルーチンを外部から転送する場合の手順例	
17.2.3	シングルブートモード	434
17.2.3.1	(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	
17.2.4	モード設定	437
17.2.5	メモリマップ	437
17.2.6	インタフェース仕様	439
17.2.7	データ転送フォーマット	440
17.2.8	メモリの制約について	440
17.2.9	ブートプログラムの転送フォーマット	440
17.2.9.1	RAM転送	
17.2.9.2	フラッシュメモリSUM	
17.2.9.3	デバイス情報読み出し	
17.2.9.4	フラッシュメモリチップ消去およびプロテクトビット消去	
17.2.10	ブートプログラム動作説明	447
17.2.10.1	RAM転送コマンド	
17.2.10.2	フラッシュメモリSUMコマンド	
17.2.10.3	製品情報読み出しコマンド	
17.2.10.4	フラッシュメモリチップ消去およびプロテクトビット消去コマンド	
17.2.10.5	ACK応答データ	
17.2.10.6	シリアル動作モード判定	
17.2.10.7	パスワードについて	
17.2.10.8	SUMの計算方法	
17.2.10.9	CHECKSUMの計算方法	
17.2.11	ブートプログラム全体フローチャート	460
<b>17.3</b>	<b>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</b>	<b>461</b>
17.3.1	フラッシュメモリ	461
17.3.1.1	ブロック構成	
17.3.1.2	基本動作	
17.3.1.3	リセット(ハードウェアリセット)	
17.3.1.4	コマンド説明	
17.3.1.5	フラッシュコントロール/ステータスレジスタ	
17.3.1.6	コマンドシーケンス一覧	
17.3.1.7	バスライトサイクル時のアドレスビット構成	
17.3.1.8	フローチャート	

---

## 第18章 プロテクト/セキュリティ機能

---

<b>18.1</b>	<b>概要</b>	<b>477</b>
-------------	-----------	------------

<b>18.2</b>	<b>特長</b> .....	477
18.2.1	内蔵 ROM (Flash)のライト/消去プロテクト.....	477
18.2.2	セキュリティ機能.....	477
<b>18.3</b>	<b>レジスタ</b> .....	478
18.3.1	レジスタ一覧.....	478
18.3.2	FCFLCS(フラッシュコントロールレジスタ).....	479
18.3.3	FCSECBIT(セキュリティビットレジスタ).....	480
<b>18.4</b>	<b>設定/解除方法</b> .....	481
18.4.1	内蔵 ROM (Flash)のライト/消去プロテクト.....	481
18.4.2	セキュリティビット.....	481

---

## 第19章 電気的特性

---

<b>19.1</b>	<b>絶対最大定格</b> .....	483
<b>19.2</b>	<b>DC 電気的特性(1/3)</b> .....	484
<b>19.3</b>	<b>DC 電気的特性(2/3)</b> .....	485
<b>19.4</b>	<b>DC 電気的特性(3/3)</b> .....	486
19.4.1	TMPM330FDFG/TMPM330FYFG.....	486
19.4.2	TMPM330FWFG.....	486
<b>19.5</b>	<b>10ビット AD コンバータ変換特性</b> .....	487
<b>19.6</b>	<b>AC 電気的特性</b> .....	488
19.6.1	AC 測定条件.....	488
19.6.2	シリアルチャネル(SIO/UART).....	488
19.6.2.1	I/O インタフェースモード	
19.6.3	シリアルバスインタフェース(I2C/SIO).....	490
19.6.3.1	I2C モード	
19.6.3.2	クロック同期式 8ビット SIO モード	
19.6.4	イベントカウンタ.....	492
19.6.5	キャプチャ.....	492
19.6.6	外部割り込み.....	492
19.6.7	NMI.....	493
19.6.8	SCOUT 端子 AC 特性.....	493
19.6.9	デバッグ通信.....	494
19.6.9.1	SWD インタフェース	
19.6.9.2	JTAG インタフェース	
19.6.10	ETM トレース.....	495
<b>19.7</b>	<b>フラッシュ特性</b> .....	495
19.7.1	書き込み特性.....	495
<b>19.8</b>	<b>発振回路</b> .....	496
19.8.1	セラミック発振子.....	496
19.8.2	水晶発振子.....	496
<b>19.9</b>	<b>取り扱い上のご注意</b> .....	497
19.9.1	はんだ濡れ性についての注意事項.....	497
19.9.2	電源投入時の注意事項.....	497

---

## 第20章 ポート部等価回路図

---

<b>20.1</b>	<b>PA0,PB1~2,PE1~3,PE5~6,PF1~7,PG0~6,PH0~7,PI6~7,PJ0~3,PJ6~7</b> .....	499
<b>20.2</b>	<b>PA1</b> .....	499
<b>20.3</b>	<b>PA2~7,PB0,PB3~7,PE0,PE4,PF0,PG7,PI0~5,PJ4~5,PK1~2</b> .....	500
<b>20.4</b>	<b>PC0~3, PD4~7</b> .....	500
<b>20.5</b>	<b>PD0~3</b> .....	500
<b>20.6</b>	<b>PK0</b> .....	501
<b>20.7</b>	<b>NMI, MODE</b> .....	501
<b>20.8</b>	<b>RESET</b> .....	501



---

20.9	X1, X2.....	502
20.10	XT1, XT2.....	502
20.11	VREFH, AVSS.....	502

---

---

**第 21 章 パッケージ寸法図**

---



# TMPM330FDFG/FYFG/FWFG

TMPM330FDFG/FYFG/FWFG は、ARM 社 Cortex™-M3 コアを内蔵した 32 ビット RISC マイクロプロセッサです。

製品名	ROM (FLASH)	RAM	パッケージ
TMPM330FDFG	512 Kbyte	32 Kbyte	LQFP100-P-1414-0.50H
TMPM330FYFG	256 Kbyte	16 Kbyte	
TMPM330FWFG	128 Kbyte	8 Kbyte	

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

#### a. Thumb®-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/16 ビット混在の命令セットでコード効率を向上

#### b. 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
- ・プロセッサコアの動作を停止させるスタンバイ機能

#### c. 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
TMPM330FDFG	512 Kbyte	32 Kbyte
TMPM330FYFG	256 Kbyte	16 Kbyte
TMPM330FWFG	128 Kbyte	8 Kbyte

### 3. 16 ビットタイマ(TMRB) : 10 チャンネル

- ・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 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(WDT) : 1 チャンネル
  - リセットまたはマスク不能割り込み(NMI)発生
6. シリアルチャンネル(SIO/UART) : 3 チャンネル
  - UART/クロック同期式モード選択可能(4 byte FIFO 内蔵)
7. シリアルバスインタフェース(I2C/SIO) : 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(CEC) : 1 チャンネル
  - 1 byte ごと送受信
9. リモコン判定機能(RMC) : 2 チャンネル
  - 72 bit まで一括受信
- 10.10 ビット AD コンバータ(ADC) : 12 チャンネル
  - ・ 内部タイマ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 1.15  $\mu$ sec ( $f_{sys} = 40$  MHz 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. マスク不能割り込み(NMI)
  - ウォッチドッグタイマまたは  $\overline{NMI}$  端子により発生
13. 入出力ポート(PORT) : 78 端子
14. 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(32.768 kHz) : SLOW, SLEEP
15. クロックジェネレータ(CG)
  - ・ PLL 内蔵(4 逓倍)
  - ・ 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
16. エンディアン
  - リトルエンディアン
17. 最大動作周波数 : 40 MHz
18. 動作電圧範囲
  - 2.7 V~3.6 V (内蔵レギュレータ使用)
19. 温度範囲
  - ・  $-20^{\circ}\text{C} \sim 85^{\circ}\text{C}$  (Flash W/E 時以外)



- ・ 0°C ~ 70°C (Flash W/E 時)

20.パッケージ

LQFP100-P-1414-0.50H (14 mm × 14 mm, 0.5 mm ピッチ)

## 1.2 ブロック図

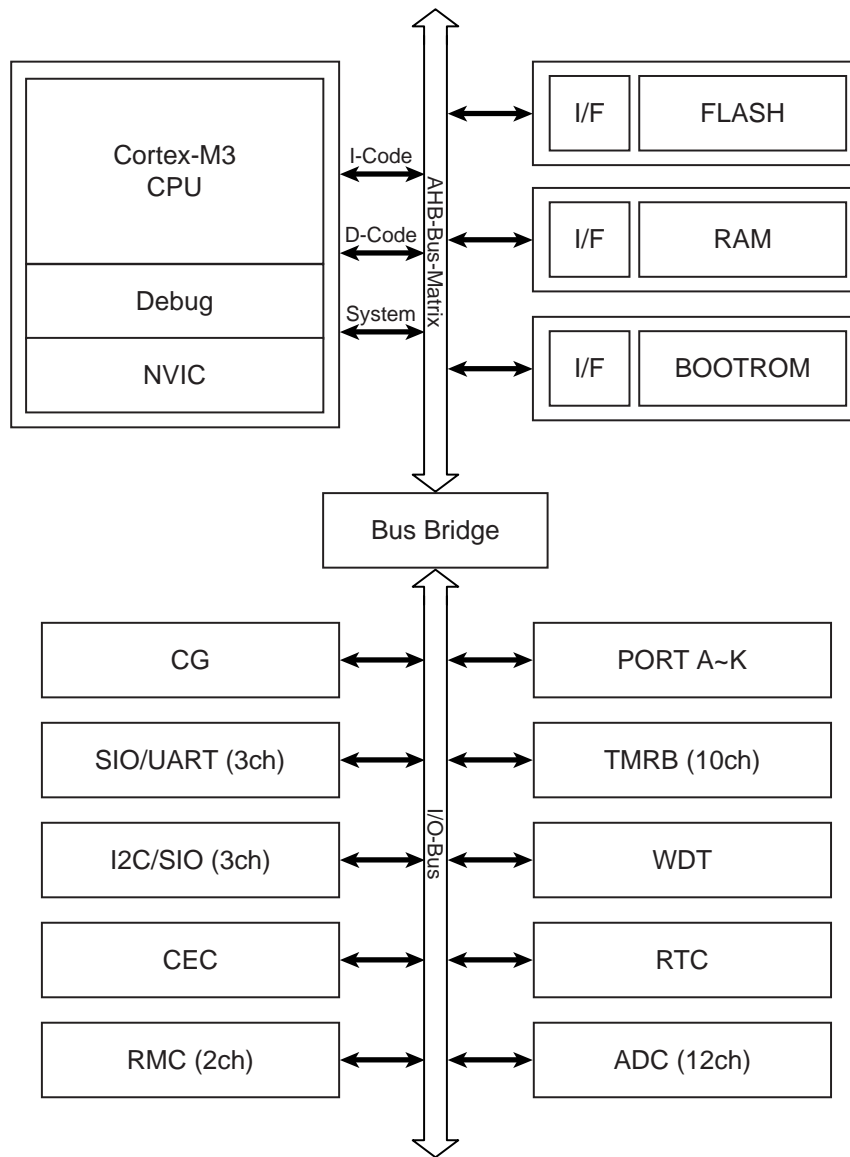


図 1-1 TMPM330FDFG/FYFG/FWFG ブロック図

### 1.3 ピン配置図(Top view)

TMPM330FDFG/FYFG/FWFG のピン配置図は、図 1-2 のとおりです。

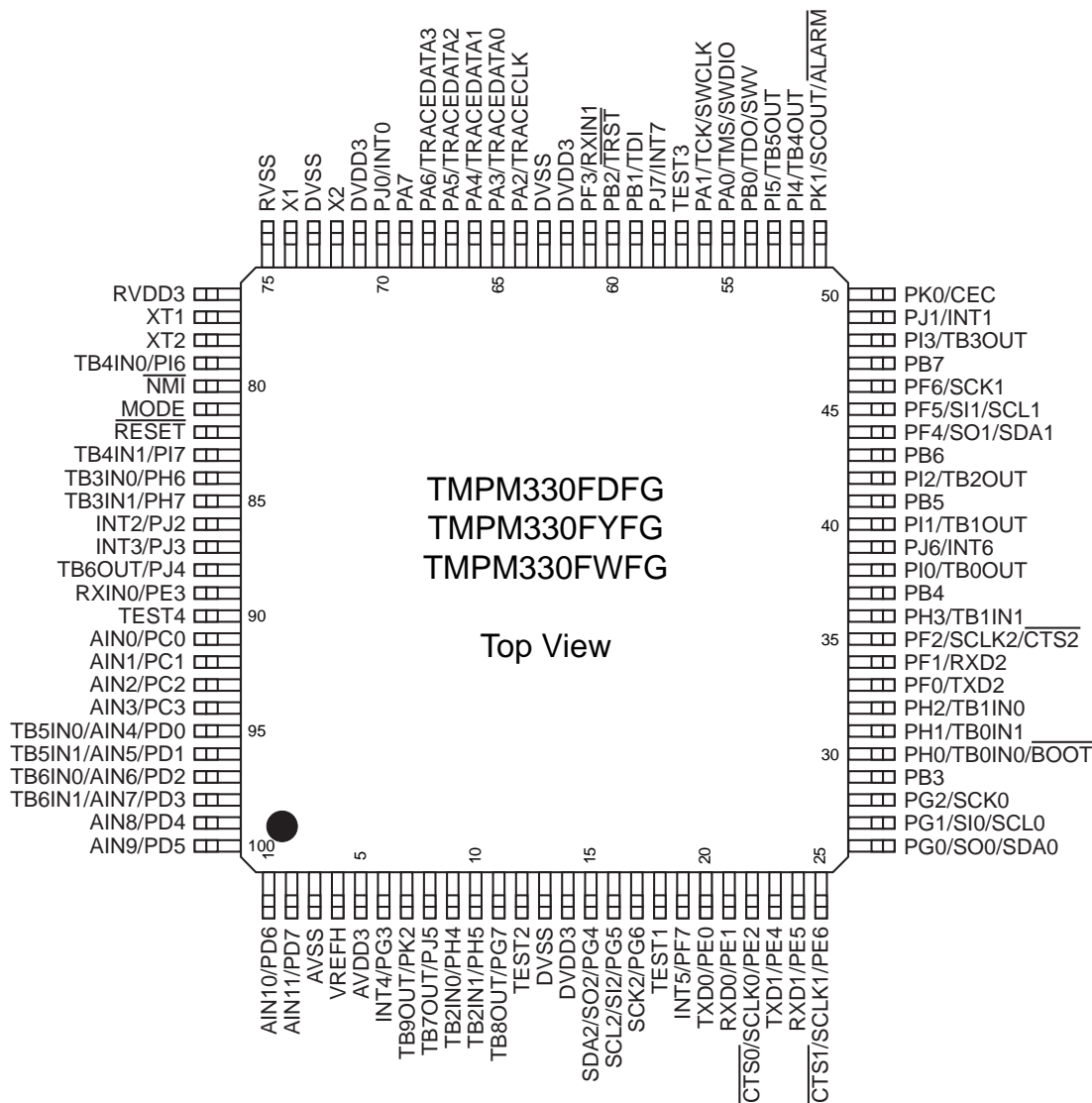


図 1-2 ピン配置図(LQFP100)

## 1.4 ピン名称と機能

TMPM330FDFG/FYFG/FWFG の入出力ピン名称と機能は、表 1-1、表 1-2 の通りです。

### 1.4.1 ピン番号順

表 1-1 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
機能	1	PD6 AIN10	入力 入力	入力ポート アナログ入力端子
機能	2	PD7 AIN11	入力 入力	入力ポート アナログ入力端子
電源	3	AVSS	入力	AD コンバータ用基準電源端子 (注)AD 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。
電源	4	VREFH	入力	AD コンバータ用基準電源端子 (注)AD 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。
電源	5	AVDD3	入力	AD コンバータ用電源端子 (注)AD 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 外部割り込み端子
機能	7	PK2 TB9OUT	入出力 出力	入出力ポート タイマ B 出力端子
機能	8	PJ5 TB7OUT	入出力 出力	入出力ポート タイマ B 出力端子
機能	9	PH4 TB2IN0	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
機能	10	PH5 TB2IN1	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
機能	11	PG7 TB8OUT	入出力 出力	入出力ポート タイマ B 出力端子
テスト	12	TEST2	-	テスト端子 (注)必ず OPEN にしてください。
電源	13	DVSS	-	GND 端子
電源	14	DVDD3	-	電源端子
機能	15	PG4 SDA2/SO2	入出力 入出力	入出力ポート I2C モード送受信、SIO モード送信端子
機能	16	PG5 SCL2/SI2	入出力 入出力	入出力ポート I2C モードクロック、SIO モード受信端子
機能	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
テスト	18	TEST1	-	テスト端子 (注)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 外部割り込み端子
機能	20	PE0 TXD0	入出力 出力	入出力ポート SIO 送信端子

表 1-1 ピン名称と機能&lt;ピン番号順&gt; (2/5)

分類	ピン番号	記号	入出力	機能
機能	21	PE1 RXD0	入出力 入力	入出力ポート SIO 受信端子
機能	22	PE2 SCLK0 $\overline{\text{CTS0}}$	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
機能	23	PE4 TXD1	入出力 出力	入出力ポート SIO 送信端子
機能	24	PE5 RXD1	入出力 入力	入出力ポート SIO 受信端子
機能	25	PE6 SCLK1 $\overline{\text{CTS1}}$	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
機能	26	PG0 SDA0/SO0	入出力 入出力	入出力ポート I2C モード送受信、SIO モード送信端子
機能	27	PG1 SCL0/SI0	入出力 入出力	入出力ポート I2C モードクロック、SIO モード受信端子
機能	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
機能	29	PB3	入出力	入出力ポート
機能/制御	30	PH0 TB0IN0 $\overline{\text{BOOT}}$	入出力 入力 入力	入出力ポート タイマ B インプットキャプチャ端子 BOOT モード端子 (注)RESET 信号の立ち上がりで"Low" をサンプリングしてシングルブートモードになります。
機能	31	PH1 TB0IN1	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
機能	32	PH2 TB1IN0	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
機能	33	PF0 TXD2	入出力 出力	入出力ポート SIO 送信端子
機能	34	PF1 RXD2	入出力 入力	入出力ポート SIO 受信端子
機能	35	PF2 SCLK2 $\overline{\text{CTS2}}$	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
機能	36	PH3 TB1IN1	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
機能	37	PB4	入出力	入出力ポート
機能	38	PI0 TB0OUT	入出力 出力	入出力ポート タイマ B 出力端子
機能	39	PJ6 INT6	入出力 入力	入出力ポート 外部割り込み端子

表 1-1 ピン名称と機能&lt;ピン番号順&gt; (3/5)

分類	ピン番号	記号	入出力	機能
機能	40	PI1 TB1OUT	入出力 出力	入出力ポート タイマ B 出力端子
機能	41	PB5	入出力	入出力ポート
機能	42	PI2 TB2OUT	入出力 出力	入出力ポート タイマ B 出力端子
機能	43	PB6	入出力	入出力ポート
機能	44	PF4 SDA1/SO1	入出力 入出力	入出力ポート I2C モード送受信、SIO モード送信端子
機能	45	PF5 SCL1/SI1	入出力 入出力	入出力ポート I2C モードクロック、SIO モード受信端子
機能	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
機能	47	PB7	入出力	入出力ポート
機能	48	PI3 TB3OUT	入出力 出力	入出力ポート タイマ B 出力端子
機能	49	PJ1 INT1	入出力 入力	入出力ポート 外部割り込み端子
機能	50	PK0 CEC	入出力 入出力	入出力ポート CEC 端子 (注)Nch オープンドレイン端子です。
機能	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
機能	52	PI4 TB4OUT	入出力 出力	入出力ポート タイマ B 出力端子
機能	53	PI5 TB5OUT	入出力 出力	入出力ポート タイマ B 出力端子
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
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
機能/ デバッグ	56	PA1 TCK/SWCLK	入出力 入力	入出力ポート デバッグ用端子
テスト	57	TEST3	-	テスト端子 (注)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 外部割り込み端子
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

表 1-1 ピン名称と機能&lt;ピン番号順&gt; (4/5)

分類	ピン番号	記号	入出力	機能
機能/ デバッグ	60	PB2 $\overline{\text{TRST}}$	入出力 入力	入出力ポート デバッグ用端子
機能	61	PF3 RXIN1	入出力 入力	入出力ポート リモコン入力
電源	62	DVDD3	-	電源端子
電源	63	DVSS	-	GND 端子
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
機能/ デバッグ	65	PA3 TRACEDATA0	入出力 出力	入出力ポート デバッグ用端子
機能/ デバッグ	66	PA4 TRACEDATA1	入出力 出力	入出力ポート デバッグ用端子
機能/ デバッグ	67	PA5 TRACEDATA2	入出力 出力	入出力ポート デバッグ用端子
機能/ デバッグ	68	PA6 TRACEDATA3	入出力 出力	入出力ポート デバッグ用端子
機能	69	PA7	入出力	入出力ポート
機能	70	PJ0 INT0	入出力 入力	入出力ポート 外部割り込み端子
電源	71	DVDD3	-	電源端子
クロック	72	X2	出力	高速発振子接続端子
電源	73	DVSS	-	GND 端子
クロック	74	X1	入力	高速発振子接続端子
電源	75	RVSS	-	GND 端子
電源	76	RVDD3	-	電源端子
クロック	77	XT1	入力	低速発振子接続端子
クロック	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
機能	80	$\overline{\text{NMI}}$	入力	マスク不能割り込み (注)ノイズフィルタ(Typ.条件で約 30ns)付きです。
制御	81	MODE	入力	モード端子 (注)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 (注)プルアップおよびノイズフィルタ(Typ.条件で約 30ns)付きです。

表 1-1 ピン名称と機能&lt;ピン番号順&gt; (5/5)

分類	ピン番号	記号	入出力	機能
機能	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
機能	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
機能	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
機能	86	PJ2 INT2	入出力 入力	入出力ポート 外部割り込み端子
機能	87	PJ3 INT3	入出力 入力	入出力ポート 外部割り込み端子
機能	88	PJ4 TB6OUT	入出力 出力	入出力ポート タイマB出力端子
機能	89	PE3 RXIN0	入出力 入力	入出力ポート リモコン入力
テスト	90	TEST4	-	テスト端子 (注)必ず OPEN にしてください。
機能	91	PC0 AIN0	入力 入力	入力ポート アナログ入力端子
機能	92	PC1 AIN1	入力 入力	入力ポート アナログ入力端子
機能	93	PC2 AIN2	入力 入力	入力ポート アナログ入力端子
機能	94	PC3 AIN3	入力 入力	入力ポート アナログ入力端子
機能	95	PD0 AIN4 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
機能	96	PD1 AIN5 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
機能	97	PD2 AIN6 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
機能	98	PD3 AIN7 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
機能	99	PD4 AIN8	入力 入力	入力ポート アナログ入力端子
機能	100	PD5 AIN9	入力 入力	入力ポート アナログ入力端子



## 1.4.2 ポート順

表 1-2 ピン名称と機能&lt;ポート順&gt; (1/5)

PORT	分類	ピン 番号	記号	入出力	機能
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
PORT A	機能/ デバッグ	56	PA1 TCK/SWCLK	入出力 入力	入出力ポート デバッグ用端子
PORT A	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
PORT A	機能/ デバッグ	65	PA3 TRACEDATA 0	入出力 出力	入出力ポート デバッグ用端子
PORT A	機能/ デバッグ	66	PA4 TRACEDATA 1	入出力 出力	入出力ポート デバッグ用端子
PORT A	機能/ デバッグ	67	PA5 TRACEDATA 2	入出力 出力	入出力ポート デバッグ用端子
PORT A	機能/ デバッグ	68	PA6 TRACEDATA 3	入出力 出力	入出力ポート デバッグ用端子
PORT A	機能	69	PA7	入出力	入出力ポート
PORT B	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
PORT B	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
PORT B	機能/ デバッグ	60	PB2 $\overline{\text{TRST}}$	入出力 入力	入出力ポート デバッグ用端子
PORT B	機能	29	PB3	入出力	入出力ポート
PORT B	機能	37	PB4	入出力	入出力ポート
PORT B	機能	41	PB5	入出力	入出力ポート
PORT B	機能	43	PB6	入出力	入出力ポート
PORT B	機能	47	PB7	入出力	入出力ポート
PORT C	機能	91	PC0 AIN0	入力 入力	入力ポート アナログ入力端子
PORT C	機能	92	PC1 AIN1	入力 入力	入力ポート アナログ入力端子
PORT C	機能	93	PC2 AIN2	入力 入力	入力ポート アナログ入力端子
PORT C	機能	94	PC3 AIN3	入力 入力	入力ポート アナログ入力端子
PORT D	機能	95	PD0 AIN4 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

表 1-2 ピン名称と機能&lt;ポート順&gt; (2/5)

PORT	分類	ピン 番号	記号	入出力	機能
PORT D	機能	96	PD1 AIN5 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
PORT D	機能	97	PD2 AIN6 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
PORT D	機能	98	PD3 AIN7 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
PORT D	機能	99	PD4 AIN8	入力 入力	入力ポート アナログ入力端子
PORT D	機能	100	PD5 AIN9	入力 入力	入力ポート アナログ入力端子
PORT D	機能	1	PD6 AIN10	入力 入力	入力ポート アナログ入力端子
PORT D	機能	2	PD7 AIN11	入力 入力	入力ポート アナログ入力端子
PORT E	機能	20	PE0 TXD0	入出力 出力	入出力ポート SIO 送信端子
PORT E	機能	21	PE1 RXD0	入出力 入力	入出力ポート SIO 受信端子
PORT E	機能	22	PE2 SCLK0 $\overline{\text{CTS0}}$	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
PORT E	機能	89	PE3 RXIN0	入出力 入力	入出力ポート リモコン入力
PORT E	機能	23	PE4 TXD1	入出力 出力	入出力ポート SIO 送信端子
PORT E	機能	24	PE5 RXD1	入出力 入力	入出力ポート SIO 受信端子
PORT E	機能	25	PE6 SCLK1 $\overline{\text{CTS1}}$	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
PORT F	機能	33	PF0 TXD2	入出力 出力	入出力ポート SIO 送信端子
PORT F	機能	34	PF1 RXD2	入出力 入力	入出力ポート SIO 受信端子
PORT F	機能	35	PF2 SCLK2 $\overline{\text{CTS2}}$	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
PORT F	機能	61	PF3 RXIN1	入出力 入力	入出力ポート リモコン入力

表 1-2 ピン名称と機能&lt;ポート順&gt; (3/5)

PORT	分類	ピン 番号	記号	入出力	機能
PORT F	機能	44	PF4 SDA1/SO1	入出力 入出力	入出力ポート I2C モード送受信、SIO モード送信端子
PORT F	機能	45	PF5 SCL1/SI1	入出力 入出力	入出力ポート I2C モードクロック、SIO モード受信端子
PORT F	機能	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
PORT F	機能	19	PF7 INT5	入出力 入力	入出力ポート 外部割り込み端子
PORT G	機能	26	PG0 SDA0/SO0	入出力 入出力	入出力ポート I2C モード送受信、SIO モード送信端子
PORT G	機能	27	PG1 SCL0/SI0	入出力 入出力	入出力ポート I2C モードクロック、SIO モード受信端子
PORT G	機能	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
PORT G	機能	6	PG3 INT4	入出力 入力	入出力ポート 外部割り込み端子
PORT G	機能	15	PG4 SDA2/SO2	入出力 入出力	入出力ポート I2C モード送受信、SIO モード送信端子
PORT G	機能	16	PG5 SCL2/SI2	入出力 入出力	入出力ポート I2C モードクロック、SIO モード受信端子
PORT G	機能	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
PORT G	機能	11	PG7 TB8OUT	入出力 出力	入出力ポート タイマ B 出力端子
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 (注)RESET 信号の立ち上がりで"Low" をサンプリングしてシングルブートモードになります。
PORT H	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
PORT H	機能	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
PORT H	機能	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
PORT H	機能	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
PORT H	機能	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
PORT H	機能	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
PORT H	機能	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

表 1-2 ピン名称と機能&lt;ポート順&gt; (4/5)

PORT	分類	ピン 番号	記号	入出力	機能
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
PORT I	機能	40	PI1 TB1OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT I	機能	42	PI2 TB2OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT I	機能	48	PI3 TB3OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT I	機能	52	PI4 TB4OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT I	機能	53	PI5 TB5OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT I	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
PORT I	機能	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
PORT J	機能	70	PJ0 INT0	入出力 入力	入出力ポート 外部割り込み端子
PORT J	機能	49	PJ1 INT1	入出力 入力	入出力ポート 外部割り込み端子
PORT J	機能	86	PJ2 INT2	入出力 入力	入出力ポート 外部割り込み端子
PORT J	機能	87	PJ3 INT3	入出力 入力	入出力ポート 外部割り込み端子
PORT J	機能	88	PJ4 TB6OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT J	機能	8	PJ5 TB7OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT J	機能	39	PJ6 INT6	入出力 入力	入出力ポート 外部割り込み端子
PORT J	機能	58	PJ7 INT7	入出力 入力	入出力ポート 外部割り込み端子
PORT K	機能	50	PK0 CEC	入出力 入出力	入出力ポート CEC 端子 (注)Nch オープンドレイン端子です。
PORT K	機能	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
PORT K	機能	7	PK2 TB9OUT	入出力 出力	入出力ポート タイマ B 出力端子
-	機能	82	$\overline{\text{RESET}}$	入力	リセット入力端子 (注)プルアップおよびノイズフィルタ(Typ.条件で約 30ns)付きです。

表 1-2 ピン名称と機能&lt;ポート順&gt; (5/5)

PORT	分類	ピン 番号	記号	入出力	機能
-	機能	80	NMI	入力	マスク不能割り込み (注)ノイズフィルタ(Typ.条件で約 30ns)付きです。
-	制御	81	MODE	入力	モード端子 (注)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
-	クロック	74	X1	入力	高速発振子接続端子
-	クロック	77	XT1	入力	低速発振子接続端子
-	クロック	78	XT2	出力	低速発振子接続端子
-	テスト	12	TEST2	-	テスト端子 (注)必ず OPEN にしてください。
-	テスト	18	TEST1	-	テスト端子 (注)必ず OPEN にしてください。
-	テスト	57	TEST3	-	テスト端子 (注)必ず OPEN にしてください。
-	テスト	90	TEST4	-	テスト端子 (注)必ず OPEN にしてください。
-	電源	3	AVSS	入力	AD コンバータ用基準電源端子 (注)AD 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。
-	電源	4	VREFH	入力	AD コンバータ用基準電源端子 (注)AD 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。
-	電源	5	AVDD3	入力	AD コンバータ用電源端子 (注)AD 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。
-	電源	13	DVSS	-	GND 端子
-	電源	14	DVDD3	-	電源端子
-	電源	62	DVDD3	-	電源端子
-	電源	63	DVSS	-	GND 端子
-	電源	71	DVDD3	-	電源端子
-	電源	73	DVSS	-	GND 端子
-	電源	75	RVSS	-	GND 端子
-	電源	76	RVDD3	-	電源端子

## 1.5 電源の種類と供給端子

表 1-3 電源の種類と供給端子

電源の種類	電圧範囲	ピン番号	電源供給端子
DVDD3	2.7 ~ 3.6V	14, 62, 71	PA, PB, PE, PF, PG, PH, PI, PJ, PK, X1, X2, XT1, XT2, RESET, NMI, MODE
AVDD3		5	PC, PD
RVDD3		76	-

## 第2章 プロセッサコア

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。

### 2.1 コアに関する情報

TMPM330FDFG/FYFG/FWFG で使用している Cortex-M3 コアのリビジョンは以下のとおりです。

CPU コア部、アーキテクチャなどの詳細は、ARM 社の下記 URL より"Cortex-M series processors"のマニュアルを参照してください。

<http://infocenter.arm.com/help/index.jsp>

製品名	コアリビジョン
TMPM330FDFG	r1p1-00rel0
TMPM330FYFG	
TMPM330FWFG	r1p1-01rel0

### 2.2 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 リビジョンで実装を選択可能なブロックは ETM™と MPU です。TMPM330FDFG/FYFG/FWFG 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

### 2.3 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。

#### 2.3.1 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。

TMPM330FDFG/FYFG/FWFG 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。

### 2.3.2 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。

TMPM330FDFG/FYFG/FWFG 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。

### 2.3.3 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。

TMPM330FDFG/FYFG/FWFG では、SysTick タイマの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。SysTick 較正值レジスタには、10ms を計測するための値が設定されますが、本製品では X1 から 8MHz を入力した場合に 10ms を計測可能な値 (0x9C4) が設定されています。また、SysTick 較正值レジスタの<NOREF>ビットと<SKEW>ビットはともに"0"が読み出され、カウントクロックが提供されていること、較正值が正確に 10ms であることを示します。

### 2.3.4 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。

TMPM330FDFG/FYFG/FWFG 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。

### 2.3.5 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。

TMPM330FDFG/FYFG/FWFG 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。

### 2.3.6 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。

TMPM330FDFG/FYFG/FWFG 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。



## 2.4 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。

TMPM330FDFG/FYFG/FWFG 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。

## 2.5 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。

TMPM330FDFG/FYFG/FWFG 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。

## 2.6 排他アクセス

Cortex-M3 コアの DCode バスは排他アクセスをサポートしていますが、TMPM330FDFG/FYFG/FWFG ではこの機能を使用していません。



## 第3章 デバッグインタフェース

### 3.1 仕様概要

TMPM330FDFG/FYFG/FWFG はデバッグツールと接続するためのデバッグインタフェースとして SWJ-DP (Serial Wire JTAG Debug Port)ユニット、なら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います。トレース出力はマイコン内部の TPIU (Trace Port Interface Unit)を通じてデバッグ用端子(TRACE DATA[3:0], SWV)に出力されます。

SWJ-DP, ETM, TPIU の詳細につきましては 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"を参照してください。

### 3.2 SWJ-DP

シリアルワイヤデバッグポート(SWCLK, SWDIO)と、JTAG デバッグポート(TDI, TDO, TMS, TCK, TRST)をサポートしています。

### 3.3 ETM

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

## 3.4 端子情報

デバッグインタフェース端子は汎用ポートと兼用です。

デバッグインタフェース端子と兼用される汎用ポートのうち、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、PB0 端子は JTAG デバッグポート機能と SWV トレース出力機能となります。

表 3-1 SWJ-DP,ETM デバッグ機能

SWJ-DP 端子名	汎用 ポート名	JTAG デバッグ機能		SW デバッグ機能	
		I/O	説明	I/O	説明
TMS / SWDIO	PA0	Input	JTAG Test Mode Selection	I/O	Serial Wire Data Input/Output
TCK / SWCLK	PA1	Input	JTAG Test Check	Input	Serial Wire Clock
TDO / SWV	PB0	Output	JTAG Test Data Output	(Output)(注)	(Serial Wire Viewer Output)
TDI	PB1	Input	JTAG Test Data Input	-	-
TRST	PB2	Input	JTAG Test RESET	-	-
TRACECLK	PA2	Output	TRACE Clock Output		
TRACEDATA0	PA3	Output	TRACE DATA Output0		
TRACEDATA1	PA4	Output	TRACE DATA Output1		
TRACEDATA2	PA5	Output	TRACE DATA Output2		
TRACEDATA3	PA6	Output	TRACE DATA Output3		

### 注) SWV 機能を許可した場合

リセット解除後、PA0/PA1/PB0/PB1/PB2 はデバッグポート端子となりますが、その他のデバッグインタフェース端子は汎用ポートです。必要に応じてデバッグ端子を使用する設定を行ってください。

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 と PB0 が機能設定(PA0:TMS/SWDIO,PB0:TDO/SWV)の場合、CGSTBYCR<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で保持されます。
- 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。

表 3-2 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。

表 3-2 デバッグインタフェース端子とリセット解除後のポート設定

ポート名 (ビット名)	デバッグ機能	リセット解除後のポートの設定値				
		機能 (PxFR)	入力 (PxIE)	出力 (PxCR)	プルアップ (PxPUP)	プルダウン (PxPDN)
PA0	TMS/SWDIO	1	1	1	1	-
PA1	TCK/SWCLK	1	1	0	-	1
PB0	TDO/SWV	1	0	1	0	-
PB1	TDI	1	1	0	1	-
PB2	TRST	1	1	0	1	-
PA2	TRACECLK	0	0	0	0	-
PA3	TRACEDATA0	0	0	0	0	-
PA4	TRACEDATA1	0	0	0	0	-
PA5	TRACEDATA2	0	0	0	0	-
PA6	TRACEDATA3	0	0	0	0	-

- : Don't care

### 3.5 ホールトモード中の周辺機能

Cortex-M3 コアがホールトモードに入ると、ウォッチドッグタイマ(WDT)が自動的に停止します。その他の周辺機能は動作を続けます。

### 3.6 リセットベクタブレーク

TMPM330DFDG/FYFG/FWFG は、 $\overline{\text{RESET}}$  端子からのリセットが有効な間デバッグツールとの通信が禁止されています。リセットベクタでブレークをかける場合、リセット解除後にデバッグツールからブレークの設定を行い、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>ビットをセットして再度リセットを発生させてください。

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。

### 3.7 デバッグツールとの接続

#### 3.7.1 接続方法

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参照してください。また、デバッグインタフェース端子はプルアップ/プルダウン抵抗を内蔵した端子です。外部にプルアップ/プルダウン抵抗を接続する際は注意してください。

#### 3.7.2 デバッグインタフェース端子を汎用ポートとして使用する際の注意

TMPM330DFDG/FYFG/FWFG は、 $\overline{\text{RESET}}$  端子からのリセットが有効な間デバッグツールとの通信が禁止されているためリセット中にデバッグモードに移行することができません。

PA0、PA1、PB0、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。

表 3-3 デバッグインタフェース端子の使用例

	使用するデバッグインタフェース端子						
	TRST	TDI	TDO / SWV	TCK / SWCLK	TMS / SWDIO	TRACE DATA[3:0]	TRACE CLK
JTAG+SW(リセット解除時)	o	o	o	o	o	x	x
JTAG+SW(TRST なし)	x	o	o	o	o	x	x
JTAG+TRACE	o	o	o	o	o	o	o
SW	x	x	x	o	o	x	x
SW+SWV	x	x	o	o	o	x	x
デバッグ機能禁止	x	x	x	x	x	x	x

o: 許可 x: 禁止(汎用ポートとして使用可能)



## 第4章 メモリマップ

### 4.1 メモリマップ

TMPM330FDFG/FYFG/FWFG 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、特殊機能レジスタ(SFR)領域は Peripheral 領域に割り付けられています。特殊機能レジスタ(SFR : Special function register)とは、入出力ポートおよび周辺機能のコントロールレジスタを示します。SRAM 領域、SFR 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。

### 4.1.1 TMPM330FDFG メモリマップ

TMPM330FDFG のメモリマップを図 4-1 に示します。

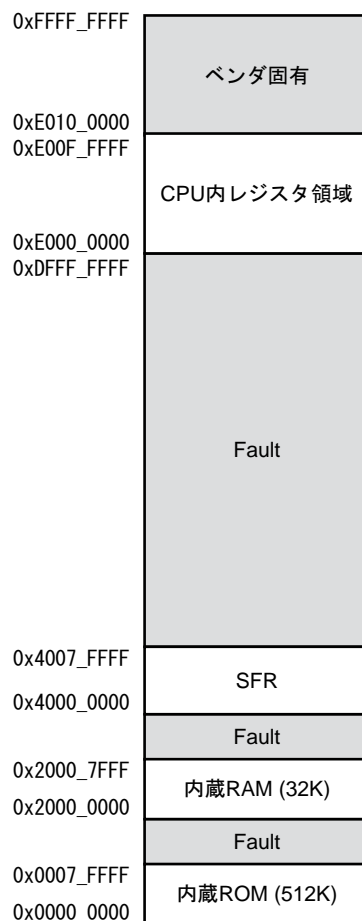


図 4-1 メモリマップ(TMPM330FDFG)



#### 4.1.2 TMPM330FYFG メモリマップ

TMPM330FYFG のメモリマップを図 4-2 に示します。

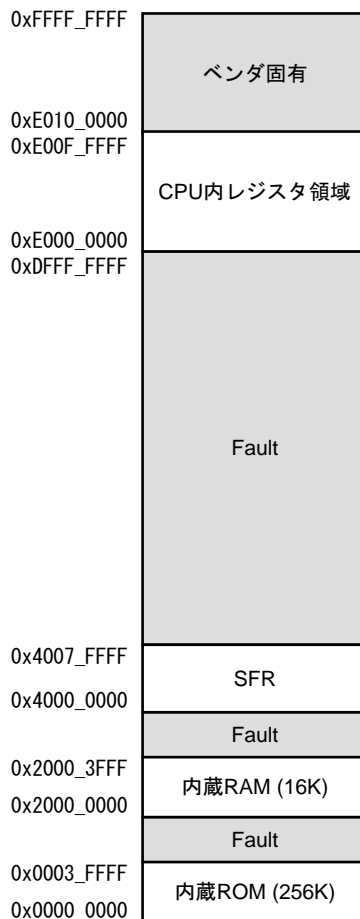


図 4-2 メモリマップ(TMPM330FYFG)

注) TMPM330FYFG 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\_FE00 ~ 0x0007\_FFFF 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、「フラッシュ動作説明」の章を参照してください。

0x0004\_0000 からパスワード領域までの間の領域はアクセスしないでください。

### 4.1.3 TMPM330FWFG メモリマップ

TMPM330FWFG のメモリマップを図 4-3 に示します。

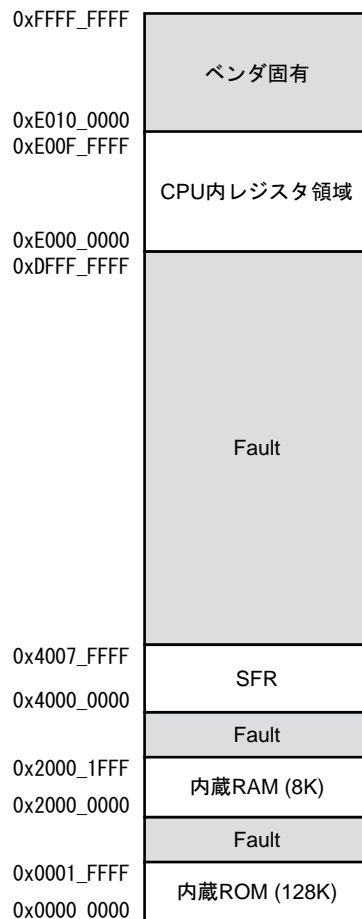


図 4-3 メモリマップ(TMPM330FWFG)

## 4.2 SFR 領域詳細

SFR 領域(0x4000\_0000~0x4007\_FFFF)のうち、周辺機能別に割り当てられているアドレス一覧を示します。

表 4-1 の予約領域および Reserved 欄に記載されているアドレスにはアクセスしないでください。また、SFR 領域で表 4-1 に記載のない領域については、読み出される値は不定となり書き込みは無視されます。

表 4-1 SFR 領域詳細

Start Address	End Address	Peripheral	Reserved
0x4000_0000	0x4000_02BF	PORT(A-K)	0x4000_0190 - 0x4000_0193 0x4000_01D0 - 0x4000_01D3 0x4000_0210 - 0x4000_0213 0x4000_0250 - 0x4000_0253
0x4001_0000	0x4001_027F	TMRB(10ch)	
0x4002_0000	0x4002_007F	I2C/SIO(3ch)	
0x4002_0080	0x4002_013F	SIO/UART(3ch)	
0x4003_0000	0x4003_007F	ADC(12ch)	0x4003_0024 - 0x4003_002F
0x4004_0000	0x4004_003F	WDT	
0x4004_0100	0x4004_013F	RTC	0x4004_010D
0x4004_0200	0x4004_023F	CG	0x4004_0230 - 0x4004_023F
0x4004_0300	0x4004_033F	CEC	
0x4004_0400	0x4004_047F	RMC(2ch)	0x4004_0428 - 0x4004_0433 0x4004_0468 - 0x4004_0473
0x4004_0500	0x4004_053F	FLASH	0x4004_0504 - 0x4004_0507 0x4004_0524 - 0x4004_052B
0x4004_0540	0x4004_05BF	予約領域	0x4004_0540 - 0x4004_0547 0x4004_0550 - 0x4004_0553 0x4004_0560 - 0x4004_0593
0x4004_0700	0x4004_073F	予約領域	0x4004_0700 - 0x4004_0707



## 第5章 リセット動作

リセットの種類として、外部リセット端子( $\overline{\text{RESET}}$ )、ウォッチドッグタイマ(WDT)、CPU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>ビットの設定によるものがあります。

ウォッチドッグタイマによるリセットについては「ウォッチドッグタイマ(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。

### 5.1 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"Low"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。

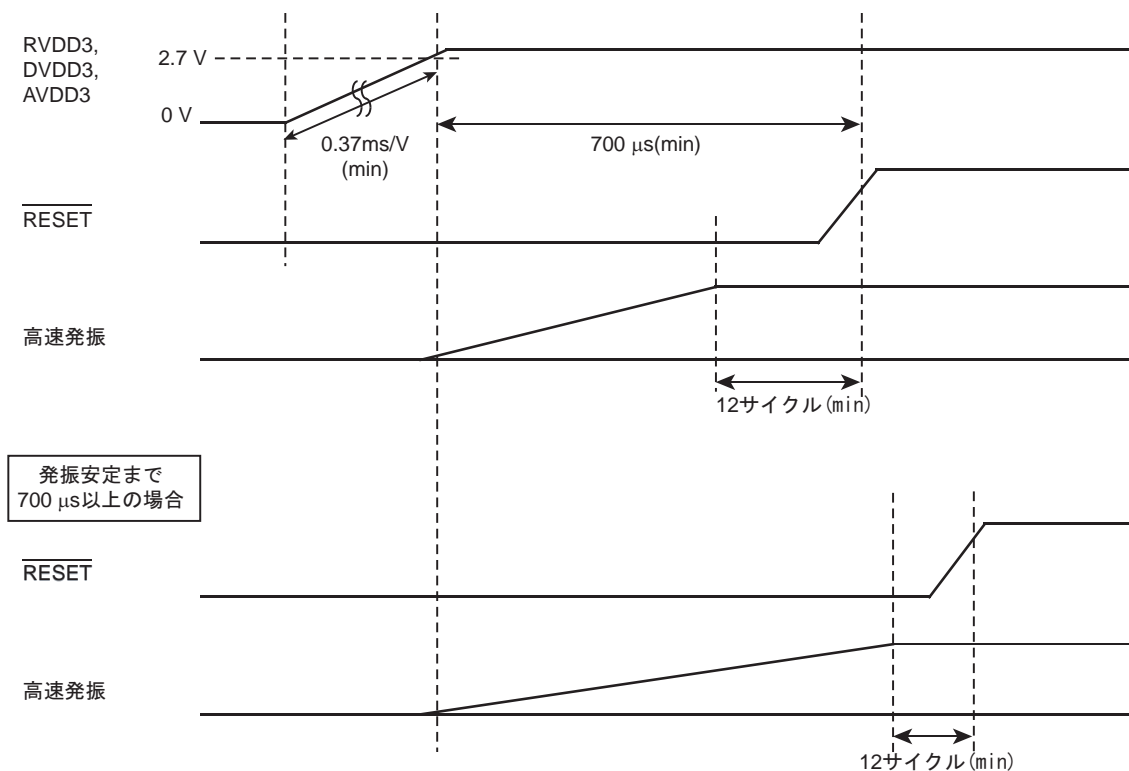


図 5-1 コールドリセットシーケンス

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  $\overline{\text{RESET}}$  端子を"Low"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。

## 5.2 ウォームリセット時

### 5.2.1 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、RESET 入力を少なくとも 12 システムクロック間ローレベル"Low"を入力してください。

### 5.2.2 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周辺機能の制御レジスタ(SFR)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 CGRSTFLG レジスタおよび FLASH 関連レジスタの FCSECBIT 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 逡倍回路が停止しているため、PLL 逡倍回路を使用する場合は CGPLLSEL レジスタにて PLL 逡倍回路の設定が必要です。

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。

## 第 6 章 クロック/モード制御

### 6.1 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(通倍回路)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
- ・ クロック通倍回路 (PLL) の制御
- ・ ウォーミングアップタイムの制御

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できます。

## 6.2 レジスタ説明

### 6.2.1 レジスタ一覧

CG 関連のレジスタとアドレスを以下に示します。

Base Address = 0x4004\_0200

レジスタ名		Address(Base+)
システムコントロールレジスタ	CGSYSCR	0x0000
発振コントロールレジスタ	CGOSCCR	0x0004
スタンバイコントロールレジスタ	CGSTBYCR	0x0008
PLL セレクトレジスタ	CGPLLSEL	0x000C
システムクロックセレクトレジスタ	CGCKSEL	0x0010



6.2.2 CGSYSCR(システムコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	SCOSEL	
リセット後	0	0	0	0	0	0	0	1
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	FPSEL	-	PRCK		
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	GEAR		
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-18	-	R	リードすると"0"が読めます。
17-16	SCOSEL[1:0]	R/W	SCOUT 出力選択 00: fs 01: fsys/2 10: fsys 11: φT0 SCOUT 端子から出力するクロックを設定します。
15-13	-	R	リードすると"0"が読めます。
12	FPSEL	R/W	fperiph 選択 0: fgear 1: fc fperiph のソースクロックを選択します。
11	-	R	リードすると"0"が読めます。
10-8	PRCK[2:0]	R/W	プリスケラック選択 000: fperiph 001: fperiph/2 010: fperiph/4 011: fperiph/8 100: fperiph/16 101: fperiph/32 110: Reserved 111: Reserved 周辺 I/O に供給するプリスケラックを選択します。
7-3	-	R	リードすると"0"が読めます。
2-0	GEAR[2:0]	R/W	高速クロック (fc) のギア選択 000: fc 001: Reserved 010: Reserved 011: Reserved 100: fc/2 101: fc/4 110: fc/8 111: Reserved

## 6.2.3 CGOSCCR(発振コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	XTEN	XEN
リセット後	0	0	0	0	0	0	1	1
	7	6	5	4	3	2	1	0
bit symbol	-	WUPT			WUPSEL	PLLON	WUEF	WUEON
リセット後	0	0	0	1	0	0	0	0

Bit	Bit Symbol	Type	機能																		
31-14	-	R	リードすると"0"が読めます。																		
13-12	-	R/W	"0"を書いてください。																		
11-10	-	R	リードすると"0"が読めます。																		
9	XTEN	R/W	低速発振器 0: 停止 1: 発振																		
8	XEN	R/W	高速発振器 0: 停止 1: 発振																		
7	-	R	リードすると"0"が読めます。																		
6-4	WUPT[2:0]	R/W	ウォーミングアップ時間選択  <table style="width: 100%; border: none;"> <tr> <td style="width: 50%;">X1 選択時</td> <td style="width: 50%;">XT1 選択時</td> </tr> <tr> <td>000: WUP なし</td> <td>000: WUP なし</td> </tr> <tr> <td>001: 2<sup>10</sup>/入力周波数</td> <td>001: 2<sup>9</sup>/入力周波数</td> </tr> <tr> <td>010: 2<sup>11</sup>/入力周波数</td> <td>010: 2<sup>7</sup>/入力周波数</td> </tr> <tr> <td>011: 2<sup>12</sup>/入力周波数</td> <td>011: 2<sup>8</sup>/入力周波数</td> </tr> <tr> <td>100: 2<sup>13</sup>/入力周波数</td> <td>100: 2<sup>15</sup>/入力周波数</td> </tr> <tr> <td>101: 2<sup>14</sup>/入力周波数</td> <td>101: 2<sup>16</sup>/入力周波数</td> </tr> <tr> <td>110: 2<sup>15</sup>/入力周波数</td> <td>110: 2<sup>17</sup>/入力周波数</td> </tr> <tr> <td>111: 2<sup>16</sup>/入力周波数</td> <td>111: 2<sup>18</sup>/入力周波数</td> </tr> </table>	X1 選択時	XT1 選択時	000: WUP なし	000: WUP なし	001: 2 <sup>10</sup> /入力周波数	001: 2 <sup>9</sup> /入力周波数	010: 2 <sup>11</sup> /入力周波数	010: 2 <sup>7</sup> /入力周波数	011: 2 <sup>12</sup> /入力周波数	011: 2 <sup>8</sup> /入力周波数	100: 2 <sup>13</sup> /入力周波数	100: 2 <sup>15</sup> /入力周波数	101: 2 <sup>14</sup> /入力周波数	101: 2 <sup>16</sup> /入力周波数	110: 2 <sup>15</sup> /入力周波数	110: 2 <sup>17</sup> /入力周波数	111: 2 <sup>16</sup> /入力周波数	111: 2 <sup>18</sup> /入力周波数
X1 選択時	XT1 選択時																				
000: WUP なし	000: WUP なし																				
001: 2 <sup>10</sup> /入力周波数	001: 2 <sup>9</sup> /入力周波数																				
010: 2 <sup>11</sup> /入力周波数	010: 2 <sup>7</sup> /入力周波数																				
011: 2 <sup>12</sup> /入力周波数	011: 2 <sup>8</sup> /入力周波数																				
100: 2 <sup>13</sup> /入力周波数	100: 2 <sup>15</sup> /入力周波数																				
101: 2 <sup>14</sup> /入力周波数	101: 2 <sup>16</sup> /入力周波数																				
110: 2 <sup>15</sup> /入力周波数	110: 2 <sup>17</sup> /入力周波数																				
111: 2 <sup>16</sup> /入力周波数	111: 2 <sup>18</sup> /入力周波数																				
3	WUPSEL	R/W	ウォームアップカウンタ 0: X1 1: XT1  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。																		
2	PLLON	R/W	PLL 動作 0: 停止 1: 発振  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。PLL を使用する場合はこのビットをセットします。																		
1	WUEF	R	ウォーミングアップタイマ(WUP)ステータス 0: ウォーミングアップ終了 1: ウォーミングアップ中  このビットを読むことでウォーミングアップタイマの動作状態を確認できます。																		
0	WUEON	W	ウォーミングアップタイマ(WUP)制御 0: don't care 1: ウォーミングアップスタート  このビットをセットすることでウォーミングアップタイマがスタートします。 リードすると"0"が読めます。																		

6.2.4 CGSTBYCR(スタンバイコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	DRVE
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	RXTEN	RXEN
リセット後	0	0	0	0	0	0	0	1
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	STBY		
リセット後	0	0	0	0	0	0	1	1

Bit	Bit Symbol	Type	機能
31-18	-	R	リードすると"0"が読めます。
17	-	R/W	"0"を書いてください。
16	DRVE	R/W	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
15-10	-	R	リードすると"0"が読めます。
9	RXTEN	R/W	STOP モード解除後の低速発振器の動作選択 0: 停止 1: 発振
8	RXEN	R/W	STOP モード解除後の高速発振器の動作選択 0: 停止 1: 発振
7-3	-	R	リードすると"0"が読めます。
2-0	STBY[2:0]	R/W	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

## 6.2.5 CGPLLSEL(PLL セレクトレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	PLLSEL
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	PLLSEL	R/W	PLL 選択 0: X1 1: PLL 使用 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 を使用する場合はこのビットの設定が必要です。

6.2.6 CGCKSEL(システムクロックセレクトレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	SYSCK	SYSCKFLG
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-2	-	R	リードすると"0"が読めます。
1	SYSCK	R/W	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 CGOSCCR<XEN>, <XTEN>を "1" に設定する必要があります。
0	SYSCKFLG	R	システムクロックステータス 0: 高速(fc) 1: 低速(fs) 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、クロックの切り替えが完了している状態です。

## 6.3 クロック制御

### 6.3.1 クロックの種類

クロックの一覧を以下に示します。

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)
fppl	: PLL により逡倍(4 逡倍)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 (プリスケラクロック)

高速クロック fc と、プリスケラクロック φT0 は以下のように分周することが可能です。

高速クロック	: fc, fc/2, fc/4, fc/8
プリスケラクロック	: fperiph, fperiph/2, fperiph/4, fperiph/8, fperiph/16, fperiph/32

CPU へは以下のクロックが入力されています。HCLK/FCLK は低消費電力モード(IDLE、SLEEP、STOP)では停止します。

HCLK, FCLK	: fsys
STCLK(Systick タイマ)	: fosc/32

### 6.3.2 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	: 発振
低速発振器	: 発振
PLL (逡倍回路)	: 停止
高速クロックギア	: fc (分周なし)

リセット動作により、低速クロック(fs)を除くすべ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 10 MHz の発振子を接続している場合、システムクロック fsys は 10 MHz となります。

### 6.3.3 クロック系統図

クロック系統図を図 6-1 に示します。

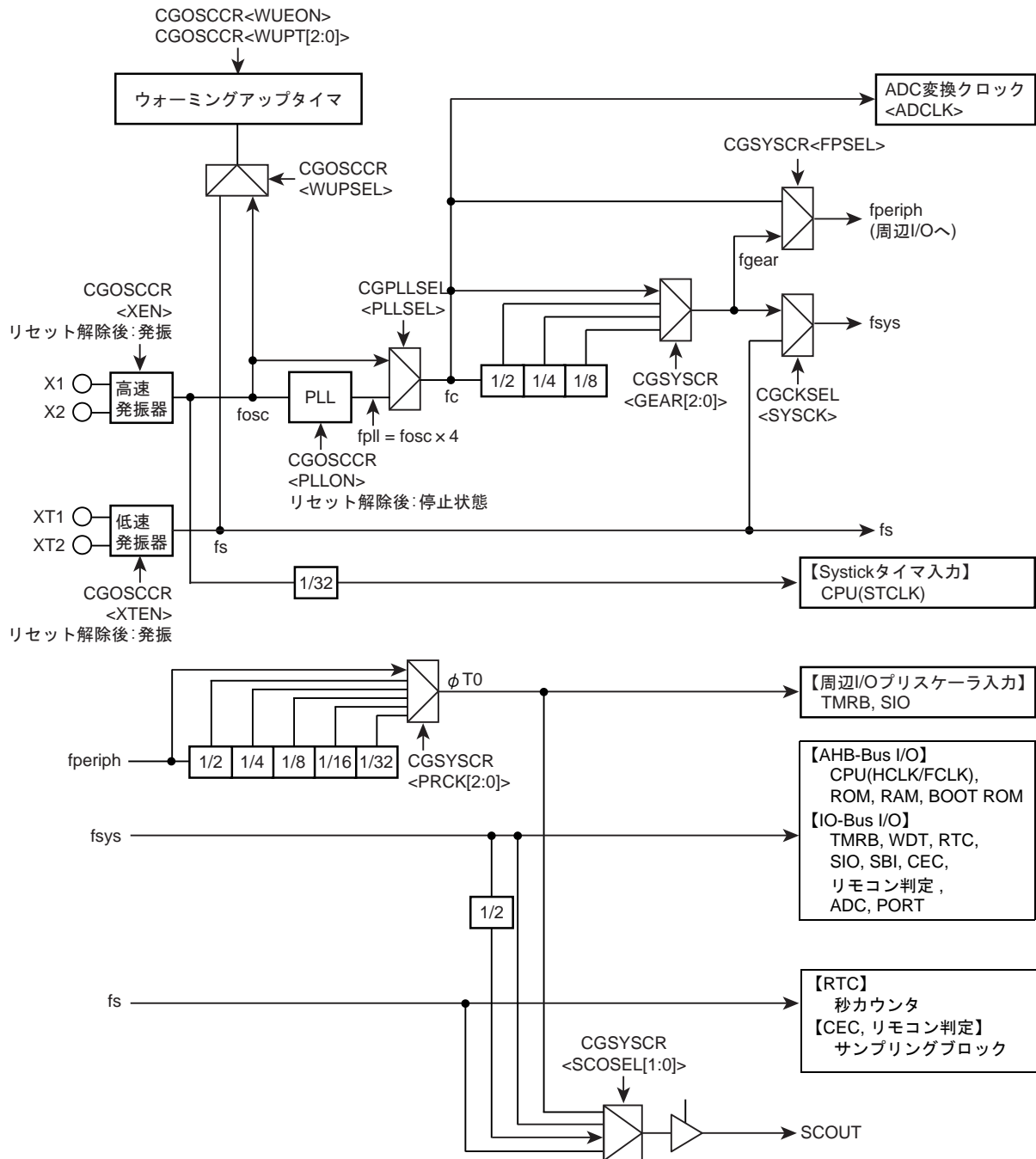


図 6-1 クロック系統図

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

### 6.3.4 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<PLLON>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  $\mu$ s です。

### 6.3.5 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、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>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 6-1 にウォーミングアップ時間を示します。



表 6-1 ウォーミングアップ時間(fosc = 10MHz, fs = 32.768kHz 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"	
000	-	WUP なし	-	WUP なし
001	2 <sup>10</sup> /入力周波数	102.4 (μs)	2 <sup>9</sup> /入力周波数	1.953 (ms)
010	2 <sup>11</sup> /入力周波数	204.8 (μs)	2 <sup>7</sup> /入力周波数	3.906 (ms)
011	2 <sup>12</sup> /入力周波数	409.6 (μs)	2 <sup>8</sup> /入力周波数	7.813 (ms)
100	2 <sup>13</sup> /入力周波数	819.2 (μs)	2 <sup>15</sup> /入力周波数	1.0 (s)
101	2 <sup>14</sup> /入力周波数	1.638 (ms)	2 <sup>16</sup> /入力周波数	2.0 (s)
110	2 <sup>15</sup> /入力周波数	3.277 (ms)	2 <sup>17</sup> /入力周波数	4.0 (s)
111	2 <sup>16</sup> /入力周波数	6.554 (ms)	2 <sup>18</sup> /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  
 CGOSCCR<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、CGSYSCR<SYSCKFLG>をリードして、システムクロックが完全に切り替わっていることを確認してください。

### 6.3.6 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できます。

- 注 1) 高速クロックの分周の切り替えは、CGSYSCR<GEAR[2:0]>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、32.768 kHz に対し約±4%です。

#### 6.3.6.1 高速クロック

- ・ X1, X2 入力周波数: 8 MHz ~ 10 MHz
- ・ 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 6-2 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	1 MHz	40 MHz	8	32	16	8	4	8	4	2	1
10MHz			10	40	20	10	5	10	5	2.5	1.25

注) PLL = ON/OFF は CGOSCCR<PLLON>にて、クロックギア(CG)は CGSYSCR<GEAR[2:0]>にて設定できます。

#### 6.3.6.2 低速クロック

XT1,XT2 に入力可能な周波数は以下の通りです。

表 6-3 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

### 6.3.7 プリスケーラクロック

周辺機能には、それぞれにクロックを分周するプリスケーラがあります。これらのプリスケー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  $f_{periph}$  をさらに 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

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ーラ出力  $\phi Tn$  は、 $\phi Tn < f_{sys}$  を満足するように時間設定( $\phi Tn$  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。

### 6.3.8 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  $f_{sys}$ 、プリスケーラクロック  $\phi T0$  を PK1/SCOUT 端子(51pin)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 6-4 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 6-4 モード別 SCOUT 出力状態

SCOUT 選択 CGSYSCR	モード	低消費電力モード			
	NORMAL	SLOW	IDLE	SLEEP	STOP
<SCOSEL[1:0]> = "00"	fs クロックを出力します				
<SCOSEL[1:0]> = "01"	f <sub>sys</sub> /2 クロックを出力します				
<SCOSEL[1:0]> = "10"	f <sub>sys</sub> 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) 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。

## 6.4 モードとモード遷移

### 6.4.1 モード状態遷移

動作モードとして NORMAL モードと SLOW モードがあります。NORMAL モードはシステムクロックに高速クロックを使用し、SLOW モードは低速クロックを使用します。

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 6-2 にモード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。

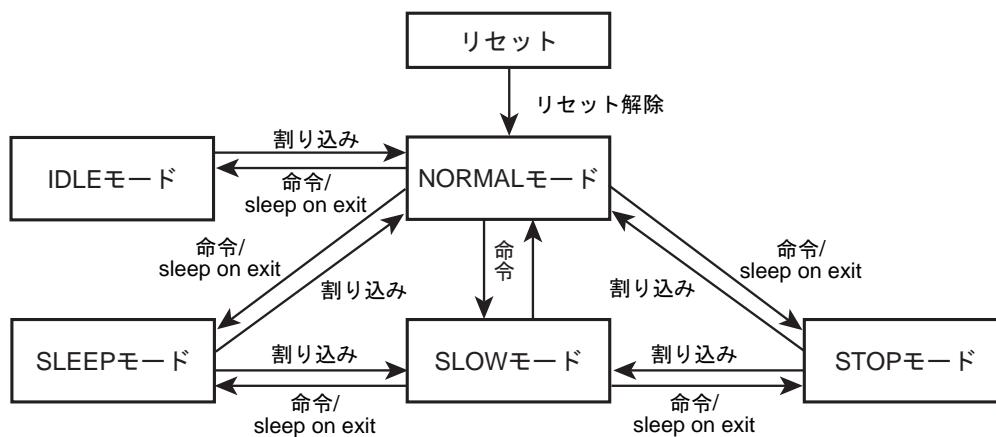


図 6-2 モード状態遷移図

## 6.5 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とおりです。

### 6.5.1 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。

### 6.5.2 SLOW モード

高速クロックを停止させ、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。

## 6.6 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とおりです。

### 6.6.1 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/UART)
- ・ シリアルバスインタフェース(I2C/SIO)
- ・ CEC 機能(CEC)
- ・ リモコン判定機能(RMC)
- ・ 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)

### 6.6.2 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。

### 6.6.3 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 6-5 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 6-5 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	"High"レベル出力	"High"レベル出力
	RESET, NMI, MODE	入力専用	o	o
ポート	PA0, PB0 (デバッグ機能設定 PxFRn<PxmFn>=1 かつ出力許可設定 PxCR<PxmC>=1 の場合)(注)	入力	×	設定(PxIE[m])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	
	PF7, PG3, PJ0, PJ1, PJ2, PJ3, PJ6, PJ7 (割り込み機能設定 PxFRn<PxmFn>=1 かつ入力許可設定 PxIE<PxmlE>=1 の場合)(注)	入力	o	o
		出力	×	設定(PxCR[m])に従う
	上記以外のポート端子	入力	×	設定(PxIE[m])に従う
	出力	×	設定(PxCR[m])に従う	

o: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。

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

### 6.6.4 低消費電力モードの選択

低消費電力モード選択は、CGSTBYCR<STBY[2:0]>の設定で選択されます。

表 6-6 に<STBY[2:0]>の設定より選択されるモードを示します。

表 6-6 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。

### 6.6.5 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 6-7 に示します。

I/Oポートについては「o」は入出力の有効、「×」は無効を、その他の機能については「o」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 6-7 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	o	o	×	×	×
I/Oポート	o	o	o	o	*(注3)
ADC	o	×(注1)	モジュール ごとに動作/ 停止選択可能	×	×
SIO	o	×(注1)		×	×
SBI	o	×(注1)		×	×
TMRB	o	×(注1)		×	×
WDT	o	×(注1)		×	×
CEC	o	o	o	o	×
リモコン判定	o	o	o	o	×
RTC	o	o	o	o	×
CG	o	o	o	o	×
PLL	o	×	o	×	×
高速発振器(fc)	o	*(注2)	o	×	×
低速発振器(fs)	o	o	o	o	×

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>の設定によります。

### 6.6.6 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、マスク不能割り込み(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 6-8 に示します。



表 6-8 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	割り 込み	INT0~7 (注 1)	○	○	○
		INTRTC	○	○	×
		INTTB0~9	○	×	×
		INTCAP00~60, 01~61	○	×	×
		INTRX0~2, INTTX0~2	○	×	×
		INTSBI0~2	○	×	×
		INTCECRX, INTCECTX	○	○	×
		INTRMCRX0, 1	○	○	×
		INTAD/INTADHP/INTADM0, 1	○	×	×
	マスク不能割り込み (INTWDT)	○	×	×	
マスク不能割り込み (NMI 端子)	○	○	○		
RESET (RESET 端子)	○	○	○		

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

×: 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。

- ・ マスク不能割り込み(NMI)による解除

マスク不能割り込みの要因には、WDT 割り込み(INTWDT)と  $\overline{\text{NMI}}$  端子があります。INTWDT は IDLE モードでのみ使用可能です。 $\overline{\text{NMI}}$  端子によるマスク不能割り込み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

$\overline{\text{RESET}}$  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 モードですべ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。

割り込みの詳細に関しては、「例外」の章の「割り込み」をご参照ください。

### 6.6.7 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  $\mu$ 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 6-9 に示します。

表 6-9 各動作モード遷移時のウォーミングアップ設定

動作モード遷移	ウォーミングアップ設定
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。

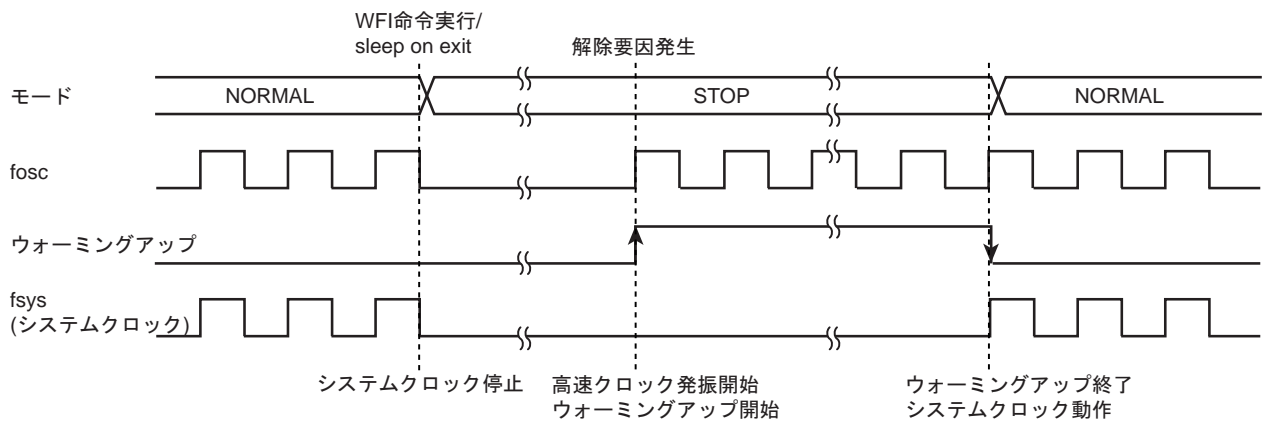
### 6.6.8 モード遷移によるクロック動作

モード遷移の際の、クロック動作について 6.6.8.1~6.6.8.4 に示します。

#### 6.6.8.1 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。

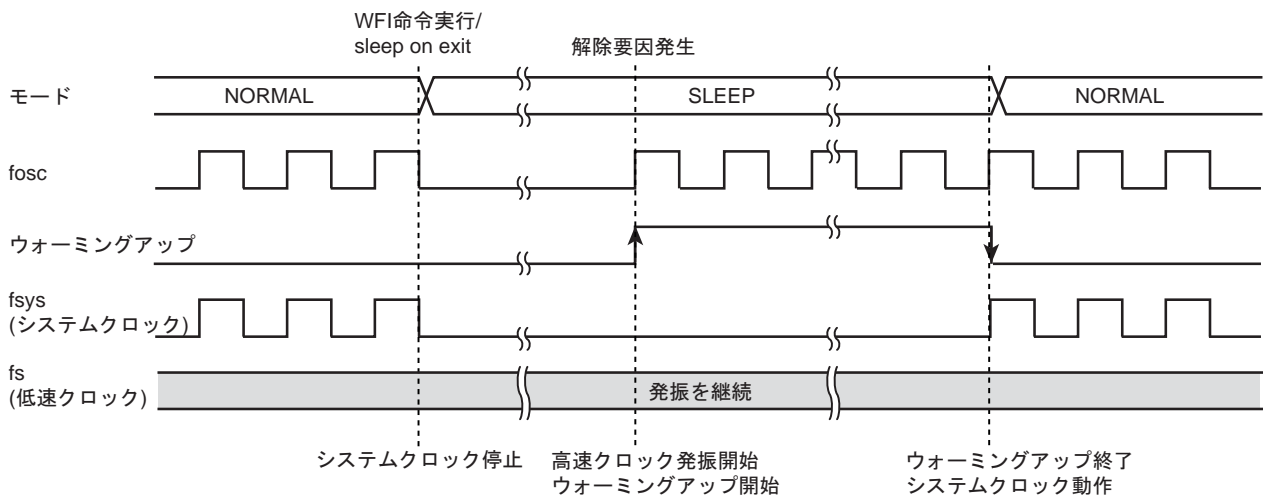
リセットで NORMAL モードへ復帰する場合にはウォーミングアップは行われませんので、発振動作が安定するまでリセット信号を有効に保ってください。



#### 6.6.8.2 NORMAL → SLEEP → NORMAL 動作モード遷移

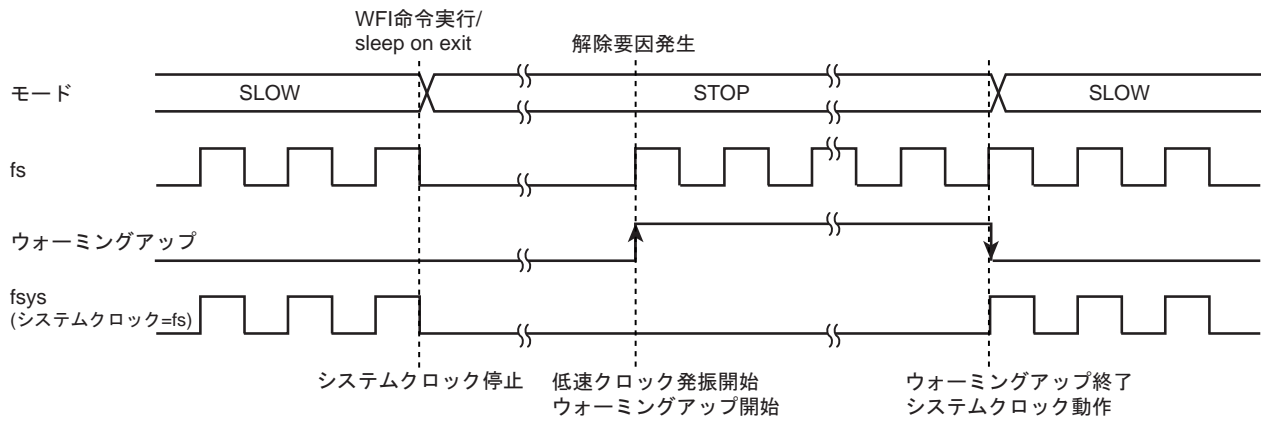
SLEEP モードから NORMAL モードへ復帰する場合、ウォーミングアップは自動的に起動します。SLEEP モードへ遷移する前にウォーミングアップ時間の設定を行ってください。

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。



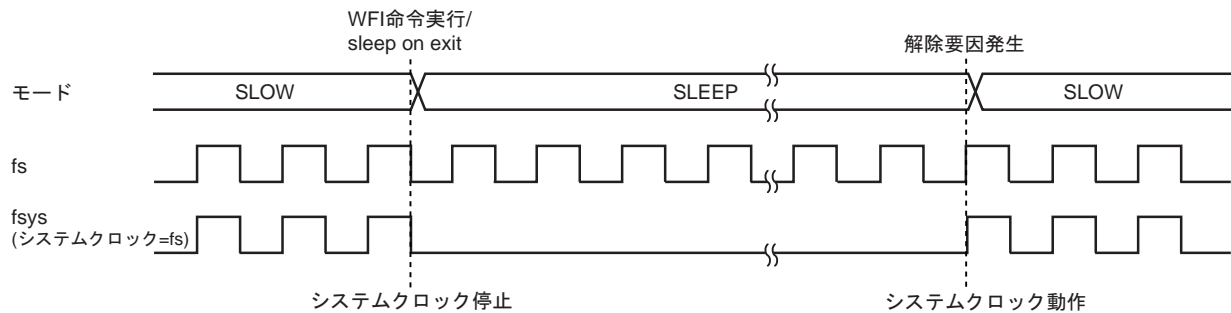
## 6.6.8.3 SLOW → STOP → SLOW 動作モード遷移

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。



## 6.6.8.4 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。



## 第7章 例外

この章では、例外の特長、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。

### 7.1 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。

すべ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。

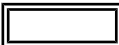
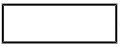
#### 7.1.1 種類

例外には以下のようなものがあります。

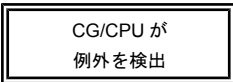
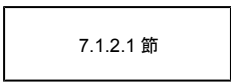

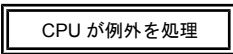


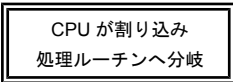

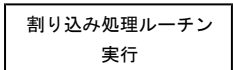
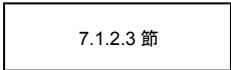

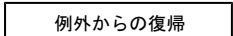
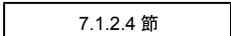
それぞれの例外の詳細な内容は、"Cortex-M3 テクニカルリファレンスマニュアル"をご覧ください。

- ・ 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
- ・ SVCcall (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

## 7.1.2 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、後続の節で説明します。

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。	 7.1.2.1節
		
 CPUが例外を処理	CPUが例外処理を行います。	 7.1.2.2節
		
 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。	 7.1.2.3節
		
 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。	 7.1.2.4節

## 7.1.2.1 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。

割り込みの要求は、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「7.5 割り込み」の節で説明します。

## (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 7-1 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	メモリマップのハードフォールト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	CPU 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) 外部割り込みは、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、「7.5.1.5 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

外部割り込みの優先度は、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該当する<PRI\_n>ビットに設定します。

<PRI\_n>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、<PRI\_n>ビットは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。

表 7-2 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。

表 7-2 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	サブ優先度数
	横取り フィールド	サブ優先度 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。



### 7.1.2.2 例外の処理と割り込み処理ルーチンへの分岐(横取り)

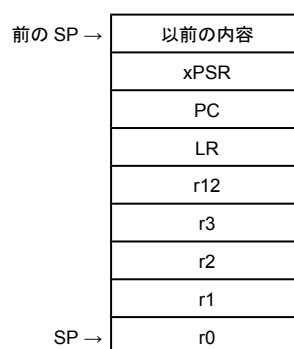
例外により、実行中の処理を中断して割り込み処理ルーチンへ分岐する動作を"横取り"と呼びます。

#### (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

## 7.1.2.3 割り込み処理ルーチンの発行

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「7.5 割り込み」の節で説明します。

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。

#### 7.1.2.4 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。

## 7.2 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 **CGRSTFLG** 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"Low"にしたのち、"High"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「ウォッチドッグタイマ」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。

## 7.3 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 **CGNMIFLG** を参照してください。

- ・ 外部  $\overline{\text{NMI}}$  端子

外部  $\overline{\text{NMI}}$  端子を"High"から"Low"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「ウォッチドッグタイマ」の章をご覧ください。

## 7.4 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 10 ms 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、システムタイマの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 8 MHz のクロックを入力した場合に 10 ms を計測可能な値(0x9C4)が設定されています。

## 7.5 割り込み

この節では、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。

### 7.5.1 要因

#### 7.5.1.1 経路

割り込み要求の経路を図 7-1 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)

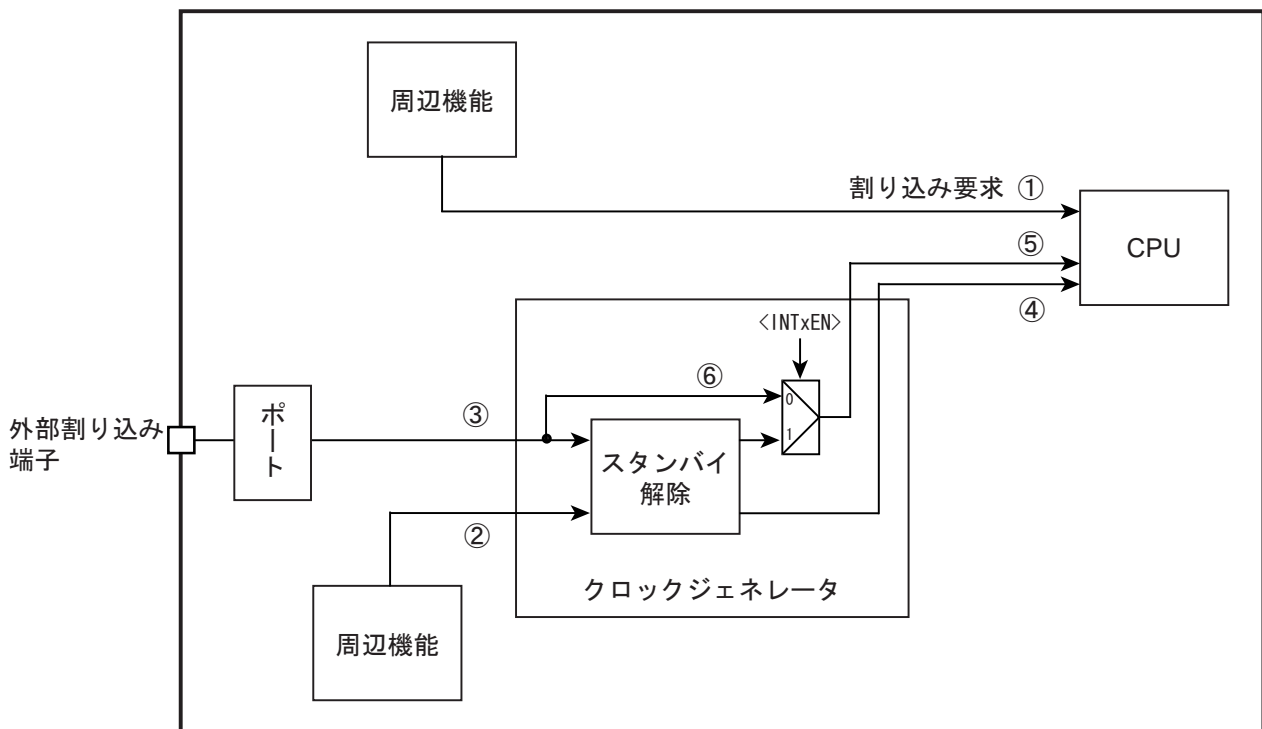


図 7-1 割り込みの経路

### 7.5.1.2 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。

### 7.5.1.3 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、クロックジェネレータを経由して CPU 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。この場合、次項の注意事項に留意して使用してください。

### 7.5.1.4 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<PxmiE>="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 7-1 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"レベルとして入力イネーブルにし、その後 CPU で割り込み許可設定を行ってください。

7.5.1.5 要因一覧

割り込みの要因一覧を表 7-3 に示します。

表 7-3 割り込み要因一覧

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 コントロールレジスタ		
0	INT0	割り込み端子(PJ0/70pin)	任意	CGIMCGA		
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)				
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	INTSBI0	シリアルバスインタフェース 0				
11	INTSBI1	シリアルバスインタフェース 1	「↑」エッジ	CGIMCGB		
12	INTCECRX	CEC 受信		CGIMCGD		
13	INTCECTX	CEC 送信		CGIMCGB		
14	INTRMCRX0	リモコン受信(channel.0)				
15	INTADHP	最優先 AD 変換終了割り込み				
16	INTADM0	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	16-bit TMRB コンペア一致 0				
19	INTTB1	16-bit TMRB コンペア一致 1				
20	INTTB2	16-bit TMRB コンペア一致 2				
21	INTTB3	16-bit TMRB コンペア一致 3				
22	INTTB4	16-bit TMRB コンペア一致 4				
23	INTTB5	16-bit TMRB コンペア一致 5				
24	INTTB6	16-bit TMRB コンペア一致 6				
25	INTRTC	時計タイマ			「↓」エッジ	CGIMCGC
26	INTCAP00	16-bit TMRB インพุットキャプチャ 00				
27	INTCAP01	16-bit TMRB インพุットキャプチャ 01				
28	INTCAP10	16-bit TMRB インพุットキャプチャ 10				
29	INTCAP11	16-bit TMRB インพุットキャプチャ 11				
30	INTCAP50	16-bit TMRB インพุットキャプチャ 50				
31	INTCAP51	16-bit TMRB インพุットキャプチャ 51				
32	INTCAP60	16-bit TMRB インพุットキャプチャ 60				
33	INTCAP61	16-bit TMRB インพุットキャプチャ 61				
34	INT6	割り込み端子(PJ6/39pin)	任意	CGIMCGC		
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	INTSBI2	シリアルバスインタフェース 2				
39	INTRMCRX1	リモコン受信(channel.1)			「↑」エッジ	CGIMCGC



表 7-3 割り込み要因一覧

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 コントロールレジスタ
40	INTTB7	16-bit TMRB コンペアー致 7		
41	INTTB8	16-bit TMRB コンペアー致 8		
42	INTTB9	16-bit TMRB コンペアー致 9		
43	INTCAP20	16-bit TMRB インพุットキャプチャ 20		
44	INTCAP21	16-bit TMRB インพุットキャプチャ 21		
45	INTCAP30	16-bit TMRB インพุットキャプチャ 30		
46	INTCAP31	16-bit TMRB インพุットキャプチャ 31		
47	INTCAP40	16-bit TMRB インพุットキャプチャ 40		
48	INTCAP41	16-bit TMRB インพุットキャプチャ 41		
49	INTAD	AD 変換終了		

#### 7.5.1.6 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 は割り込み信号の"High"を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"High"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"High"レベル、"Low"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>を有効にし、CGIMCGx<EMCGx>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 7-3 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、"High"レベル信号で CPU に通知されます。

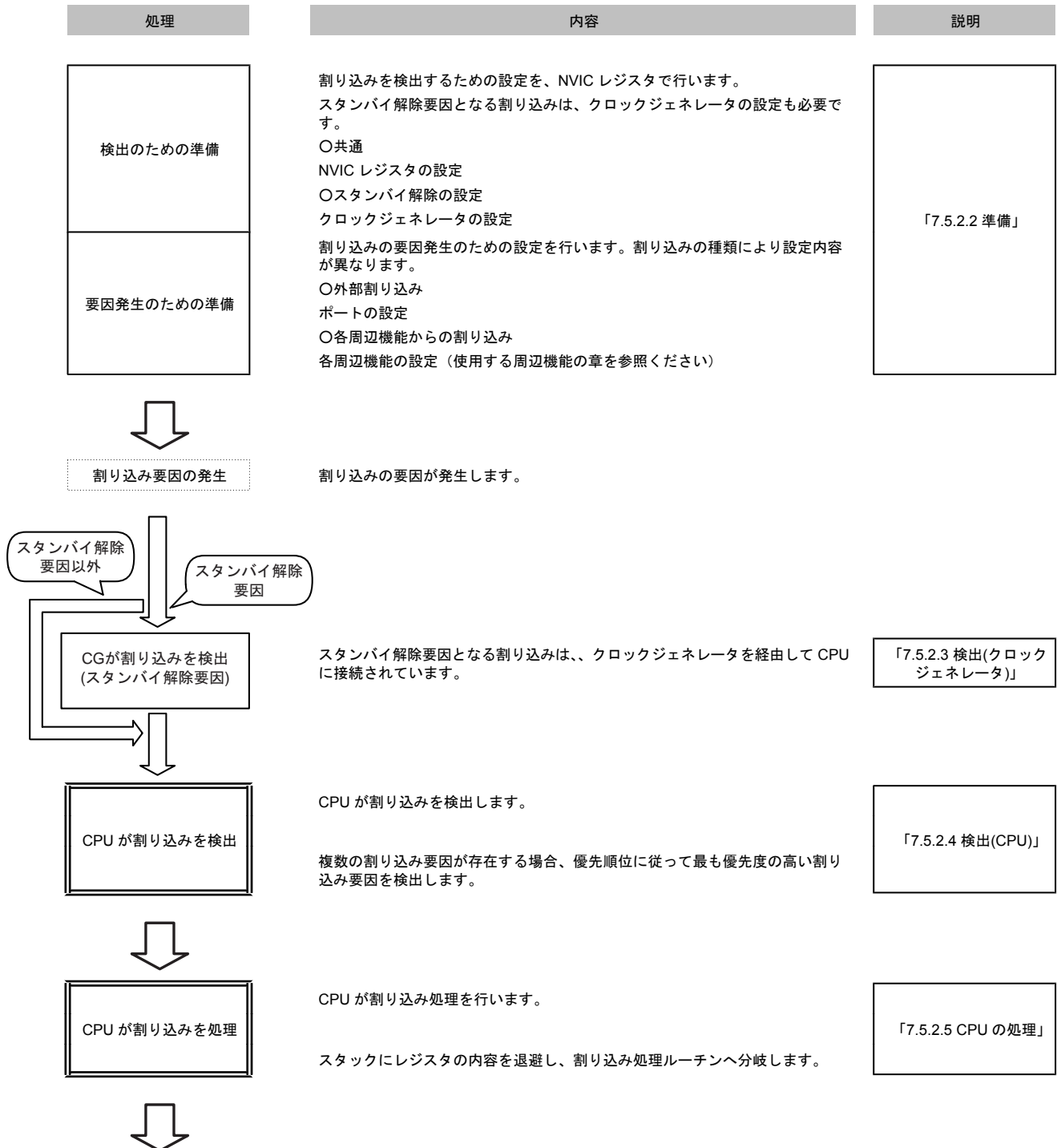
注) CEC 受信/送信割り込み, リモコン受信割り込み, 時計タイマ割り込みは、スタンバイ解除割り込みとして使用しない場合も<INTxEN>を"1"に設定し、指定されたアクティブ状態の設定を行ってください。

## 7.5.2 処理詳細

### 7.5.2.1 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
<p>割り込み サービスルーチン実行</p>	<p>必要な処理をプログラミングしてください。 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>	<p>「7.5.2.6 割り込み処理 ルーチンでの処理(要因の 取り下げ)」</p>
<p style="text-align: center;">↓</p> <p>元のプログラムへ復帰</p>	<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>	

### 7.5.2.2 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1) PRIMASKレジスタは、ユーザ・アクセス・レベルではセットできません。

注2) PRIMASKレジスタに"1"がセットされているとき、フォールトが発生するとハードフォールトとして扱われます。

#### (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<PRI\_n>に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。

グループ優先度を設定する場合に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<PRIGROUP>も設定します。

NVIC レジスタ		
<PRI_n>	←	「優先度」
<PRIGROUP>	←	「グループ優先度」(必要に応じて設定してください)

注) 「n」は該当する例外/割り込みの番号を示します。

本製品では割り込み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。

### (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 PxFRn[m]を"1"に、ポートを入力として使用するために PxIE[m]を"1"に設定します。

ポートレジスタ		
PxFRn<PxmFn>	←	"1"
PxIE<PxmlE>	←	"1"

注) 「x」は該当ポート番号、「m」は該当ビット、「n」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、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。また、「7.5.1.4 外部割り込み端子を使用する際の注意」の記載事項に注意してください。

### (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 CGICRCG 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、「7.6.3.5 CGICRCG(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、"High"パルスまたは"High"レベルの信号を入力する必要があります。また、「7.5.1.4 外部割り込み端子を使用する際の注意」の記載事項に注意してください。

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

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を発生する場合、割り込み保留クリアを行うと割り込み要因そのものが失われるため、この操作は不要です。

最後に、PRIMASKレジスタを"0"にクリアします。

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1) 「m」は該当ビットを示します。

注2) PRIMASKレジスタは、ユーザ・アクセス・レベルではセットできません。

#### 7.5.2.3 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ち下が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、"High"レベルまたは"Low"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るとみなされるため、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レジスタ(CGICRCG)で解除されるまで"High"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。

#### 7.5.2.4 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。

#### 7.5.2.5 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。

#### 7.5.2.6 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、CGICRCG レジスタ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。

## 7.6 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。

### 7.6.1 レジスタ一覧

NVIC レジスタ Base Address = 0xE000\_E000

レジスタ名	Address
SysTick 制御およびステータスレジスタ	0x0010
SysTick リロード値レジスタ	0x0014
SysTick 現在値レジスタ	0x0018
SysTick 較正值レジスタ	0x001C
割り込みイネーブルセットレジスタ 1	0x0100
割り込みイネーブルセットレジスタ 2	0x0104
割り込みイネーブルクリアレジスタ 1	0x0180
割り込みイネーブルクリアレジスタ 2	0x0184
割り込み保留セットレジスタ 1	0x0200
割り込み保留セットレジスタ 2	0x0204
割り込み保留クリアレジスタ 1	0x0280
割り込み保留クリアレジスタ 2	0x0284
割り込み優先度レジスタ	0x0400 ~ 0x0430
ベクタテーブルオフセットレジスタ	0x0D08
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	0x0D0C
システムハンドラ優先度レジスタ	0x0D18, 0x0D1C, 0x0D20
システムハンドラ制御および状態レジスタ	0x0D24

クロックジェネレータレジスタ Base Address = 0x4004\_0200

レジスタ名	Address
CG 割り込み要求クリアレジスタ	CGICRCG 0x0014
NMI フラグレジスタ	CGNMIFLG 0x0018
リセットフラグレジスタ	CGRSTFLG 0x001C
CG 割り込みモードコントロールレジスタ A	CGIMCGA 0x0020
CG 割り込みモードコントロールレジスタ B	CGIMCGB 0x0024
CG 割り込みモードコントロールレジスタ C	CGIMCGC 0x0028
CG 割り込みモードコントロールレジスタ D	CGIMCGD 0x002C
Reserved	- 0x0030
Reserved	- 0x0034
Reserved	- 0x0038
Reserved	- 0x003C

注) "Reserved"表記のアドレスにはアクセスしないでください。



## 7.6.2 NVIC レジスタ

## 7.6.2.1 SysTick 制御およびステータスレジスタ

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	COUNTFLAG
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	CLKSOURCE	TICKINT	ENABLE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-17	-	R	リードすると"0"が読めます。
16	COUNTFLAG	R/W	0: タイマは0になっていない 1: タイマが0になった "1"の場合、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。
15-3	-	R	リードすると"0"が読めます。
2	CLKSOURCE	R/W	0: 外部参照クロック 1: CPU クロック
1	TICKINT	R/W	0: SysTick を保留しない 1: SysTick を保留する
0	ENABLE	R/W	0: ディセーブル 1: イネーブル "1"をセットすると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。

## 7.6.2.2 SysTick リロード値レジスタ

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	RELOAD							
リセット後	不定							
	15	14	13	12	11	10	9	8
bit symbol	RELOAD							
リセット後	不定							
	7	6	5	4	3	2	1	0
bit symbol	RELOAD							
リセット後	不定							

Bit	Bit Symbol	Type	機能
31-24	-	R	リードすると"0"が読めます。
23-0	RELOAD	R/W	リロード値 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。

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。

## 7.6.2.3 SysTick 現在値レジスタ

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	CURRENT							
リセット後	不定							
	15	14	13	12	11	10	9	8
bit symbol	CURRENT							
リセット後	不定							
	7	6	5	4	3	2	1	0
bit symbol	CURRENT							
リセット後	不定							

Bit	Bit Symbol	Type	機能
31-24	-	R	リードすると"0"が読めます。
23-0	CURRENT	R/W	[リード] SysTick タイマ現在値 [ライト] クリア 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>もクリアされます。

## 7.6.2.4 SysTick 較正值レジスタ

	31	30	29	28	27	26	25	24
bit symbol	NOREF	SKEW	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TENMS							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TENMS							
リセット後	0	0	0	0	1	0	0	1
	7	6	5	4	3	2	1	0
bit symbol	TENMS							
リセット後	1	1	0	0	0	1	0	0

Bit	Bit Symbol	Type	機能
31	NOREF	R	0: 参照クロックあり 1: 参照クロックなし
30	SKEW	R	0: 較正值は 10 ms 1: 較正值は 10 ms でない
29-24	-	R	リードすると"0"が読めます。
23-0	TENMS	R	較正值 10 ms をカウントするために使用するリロード値(0x9C4)です。(注)

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 8 MHz のクロックを入力した場合に 10 ms を計測可能な値が設定されています。

7.6.2.5 割り込みイネーブルセットレジスタ 1

	31	30	29	28	27	26	25	24
bit symbol	SETENA (割り込み 31)	SETENA (割り込み 30)	SETENA (割り込み 29)	SETENA (割り込み 28)	SETENA (割り込み 27)	SETENA (割り込み 26)	SETENA (割り込み 25)	SETENA (割り込み 24)
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	SETENA (割り込み 23)	SETENA (割り込み 22)	SETENA (割り込み 21)	SETENA (割り込み 20)	SETENA (割り込み 19)	SETENA (割り込み 18)	SETENA (割り込み 17)	SETENA (割り込み 16)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	SETENA (割り込み 15)	SETENA (割り込み 14)	SETENA (割り込み 13)	SETENA (割り込み 12)	SETENA (割り込み 11)	SETENA (割り込み 10)	SETENA (割り込み 9)	SETENA (割り込み 8)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SETENA (割り込み 7)	SETENA (割り込み 6)	SETENA (割り込み 5)	SETENA (割り込み 4)	SETENA (割り込み 3)	SETENA (割り込み 2)	SETENA (割り込み 1)	SETENA (割り込み 0)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	SETENA	R/W	割り込み番号[31:0] [ライト] 1: 許可 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しています。 "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、該当する割り込みの許可/禁止状態が確認できます。

注) 割り込みの内容と割り込み番号については、「7.5.1.5 要因一覧」を参照してください。

## 7.6.2.6 割り込みイネーブルセットレジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	SETENA (割り込み 49)	SETENA (割り込み 48)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	SETENA (割り込み 47)	SETENA (割り込み 46)	SETENA (割り込み 45)	SETENA (割り込み 44)	SETENA (割り込み 43)	SETENA (割り込み 42)	SETENA (割り込み 41)	SETENA (割り込み 40)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SETENA (割り込み 39)	SETENA (割り込み 38)	SETENA (割り込み 37)	SETENA (割り込み 36)	SETENA (割り込み 35)	SETENA (割り込み 34)	SETENA (割り込み 33)	SETENA (割り込み 32)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-18	-	R	リードすると"0"が読めます。
17-0	SETENA	R/W	割り込み番号[49:32] [ライト] 1: 許可 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しています。 "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、該当する割り込みの許可/禁止状態が確認できます。

注) 割り込みの内容と割り込み番号については、「7.5.1.5 要因一覧」を参照してください。

7.6.2.7 割り込みイネーブルクリアレジスタ 1

	31	30	29	28	27	26	25	24
bit symbol	CLRENA (割り込み 31)	CLRENA (割り込み 30)	CLRENA (割り込み 29)	CLRENA (割り込み 28)	CLRENA (割り込み 27)	CLRENA (割り込み 26)	CLRENA (割り込み 25)	CLRENA (割り込み 24)
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	CLRENA (割り込み 23)	CLRENA (割り込み 22)	CLRENA (割り込み 21)	CLRENA (割り込み 20)	CLRENA (割り込み 19)	CLRENA (割り込み 18)	CLRENA (割り込み 17)	CLRENA (割り込み 16)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	CLRENA (割り込み 15)	CLRENA (割り込み 14)	CLRENA (割り込み 13)	CLRENA (割り込み 12)	CLRENA (割り込み 11)	CLRENA (割り込み 10)	CLRENA (割り込み 9)	CLRENA (割り込み 8)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CLRENA (割り込み 7)	CLRENA (割り込み 6)	CLRENA (割り込み 5)	CLRENA (割り込み 4)	CLRENA (割り込み 3)	CLRENA (割り込み 2)	CLRENA (割り込み 1)	CLRENA (割り込み 0)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	CLRENA	R/W	割り込み番号[31:0] [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しており、割り込みを禁止したり、禁止されているかどうかを確認できます。 "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、該当する割り込みの許可/禁止の状態を確認できます。

注) 割り込みの内容と割り込み番号については、「7.5.1.5 要因一覧」を参照してください。

## 7.6.2.8 割り込みイネーブルクリアレジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	CLRENA (割り込み 49)	CLRENA (割り込み 48)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	CLRENA (割り込み 47)	CLRENA (割り込み 46)	CLRENA (割り込み 45)	CLRENA (割り込み 44)	CLRENA (割り込み 43)	CLRENA (割り込み 42)	CLRENA (割り込み 41)	CLRENA (割り込み 40)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CLRENA (割り込み 39)	CLRENA (割り込み 38)	CLRENA (割り込み 37)	CLRENA (割り込み 36)	CLRENA (割り込み 35)	CLRENA (割り込み 34)	CLRENA (割り込み 33)	CLRENA (割り込み 32)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-18	-	R	リードすると"0"が読めます。
17-0	CLRENA	R/W	割り込み番号[49:32] [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しており、割り込みを禁止したり、禁止されているかどうかを確認できます。 "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、該当する割り込みの許可/禁止の状態を確認できます。

注) 割り込みの内容と割り込み番号については、「7.5.1.5 要因一覧」を参照してください。



7.6.2.9 割り込み保留セットレジスタ 1

	31	30	29	28	27	26	25	24
bit symbol	SETPEND (割り込み 31)	SETPEND (割り込み 30)	SETPEND (割り込み 29)	SETPEND (割り込み 28)	SETPEND (割り込み 27)	SETPEND (割り込み 26)	SETPEND (割り込み 25)	SETPEND (割り込み 24)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	23	22	21	20	19	18	17	16
bit symbol	SETPEND (割り込み 23)	SETPEND (割り込み 22)	SETPEND (割り込み 21)	SETPEND (割り込み 20)	SETPEND (割り込み 19)	SETPEND (割り込み 18)	SETPEND (割り込み 17)	SETPEND (割り込み 16)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	15	14	13	12	11	10	9	8
bit symbol	SETPEND (割り込み 15)	SETPEND (割り込み 14)	SETPEND (割り込み 13)	SETPEND (割り込み 12)	SETPEND (割り込み 11)	SETPEND (割り込み 10)	SETPEND (割り込み 9)	SETPEND (割り込み 8)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	7	6	5	4	3	2	1	0
bit symbol	SETPEND (割り込み 7)	SETPEND (割り込み 6)	SETPEND (割り込み 5)	SETPEND (割り込み 4)	SETPEND (割り込み 3)	SETPEND (割り込み 2)	SETPEND (割り込み 1)	SETPEND (割り込み 0)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
31-0	SETPEND	R/W	割り込み番号[31:0] [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり 各ビットが指定された番号の割り込みに対応しており、割り込みを強制的に保留したり、保留されているかどうかを確認できます。 "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保留されているかどうかを示します。 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対応するビットに"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、「7.5.1.5 要因一覧」を参照してください。

## 7.6.2.10 割り込み保留セットレジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	SETPEND (割り込み 49)	SETPEND (割り込み 48)
リセット後	0	0	0	0	0	0	不定	不定
	15	14	13	12	11	10	9	8
bit symbol	SETPEND (割り込み 47)	SETPEND (割り込み 46)	SETPEND (割り込み 45)	SETPEND (割り込み 44)	SETPEND (割り込み 43)	SETPEND (割り込み 42)	SETPEND (割り込み 41)	SETPEND (割り込み 40)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	7	6	5	4	3	2	1	0
bit symbol	SETPEND (割り込み 39)	SETPEND (割り込み 38)	SETPEND (割り込み 37)	SETPEND (割り込み 36)	SETPEND (割り込み 35)	SETPEND (割り込み 34)	SETPEND (割り込み 33)	SETPEND (割り込み 32)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
31-18	-	R	リードすると"0"が読めます。
17-0	SETPEND	R/W	<p>割り込み番号[49:32] [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり</p> <p>各ビットが指定された番号の割り込みに対応しており、割り込みを強制的に保留したり、保留されているかどうかを確認できます。</p> <p>"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。</p> <p>リードの場合、該当する割り込みが保留されているかどうかを示します。</p> <p>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対応するビットに"1"をセットします。</p>

注) 割り込みの内容と割り込み番号については、「7.5.1.5 要因一覧」を参照してください。

7.6.2.11 割り込み保留クリアレジスタ 1

	31	30	29	28	27	26	25	24
bit symbol	CLRPEND (割り込み 31)	CLRPEND (割り込み 30)	CLRPEND (割り込み 29)	CLRPEND (割り込み 28)	CLRPEND (割り込み 27)	CLRPEND (割り込み 26)	CLRPEND (割り込み 25)	CLRPEND (割り込み 24)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	23	22	21	20	19	18	17	16
bit symbol	CLRPEND (割り込み 23)	CLRPEND (割り込み 22)	CLRPEND (割り込み 21)	CLRPEND (割り込み 20)	CLRPEND (割り込み 19)	CLRPEND (割り込み 18)	CLRPEND (割り込み 17)	CLRPEND (割り込み 16)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	15	14	13	12	11	10	9	8
bit symbol	CLRPEND (割り込み 15)	CLRPEND (割り込み 14)	CLRPEND (割り込み 13)	CLRPEND (割り込み 12)	CLRPEND (割り込み 11)	CLRPEND (割り込み 10)	CLRPEND (割り込み 9)	CLRPEND (割り込み 8)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	7	6	5	4	3	2	1	0
bit symbol	CLRPEND (割り込み 7)	CLRPEND (割り込み 6)	CLRPEND (割り込み 5)	CLRPEND (割り込み 4)	CLRPEND (割り込み 3)	CLRPEND (割り込み 2)	CLRPEND (割り込み 1)	CLRPEND (割り込み 0)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
31-0	CLRPEND	R/W	割り込み番号[31:0] [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 各ビットが指定された番号の割り込みに対応しており、保留された割り込みをクリアしたり、保留されているかどうかを確認できます。 "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保留されているかどうかを示します。

注) 割り込みの内容と割り込み番号については、「7.5.1.5 要因一覧」を参照してください。

7.6.2.12 割り込み保留クリアレジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	CLRPEND (割り込み 49)	CLRPEND (割り込み 48)
リセット後	0	0	0	0	0	0	不定	不定
	15	14	13	12	11	10	9	8
bit symbol	CLRPEND (割り込み 47)	CLRPEND (割り込み 46)	CLRPEND (割り込み 45)	CLRPEND (割り込み 44)	CLRPEND (割り込み 43)	CLRPEND (割り込み 42)	CLRPEND (割り込み 41)	CLRPEND (割り込み 40)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	7	6	5	4	3	2	1	0
bit symbol	CLRPEND (割り込み 39)	CLRPEND (割り込み 38)	CLRPEND (割り込み 37)	CLRPEND (割り込み 36)	CLRPEND (割り込み 35)	CLRPEND (割り込み 34)	CLRPEND (割り込み 33)	CLRPEND (割り込み 32)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
31-18	-	R	リードすると"0"が読めます。
17-0	CLRPEND	R/W	割り込み番号[49:32] [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 各ビットが指定された番号の割り込みに対応しており、保留された割り込みをクリアしたり、保留されているかどうかを確認できます。 "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保留されているかどうかを示します。

注) 割り込みの内容と割り込み番号については、「7.5.1.5 要因一覧」を参照してください。

7.6.2.13 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。

	31	30	29	28	27	26	25	24
bit symbol	PRI_3			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	PRI_2			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	PRI_1			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PRI_0			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-29	PRI_3	R/W	割り込み番号 3 優先度
28-24	-	R	リードすると"0"が読めます。
23-21	PRI_2	R/W	割り込み番号 2 優先度
20-16	-	R	リードすると"0"が読めます。
15-13	PRI_1	R/W	割り込み番号 1 優先度
12-8	-	R	リードすると"0"が読めます。
7-5	PRI_0	R/W	割り込み番号 0 優先度
4-0	-	R	リードすると"0"が読めます。

## 7.6.2.14 ベクタテーブルオフセットレジスタ

	31	30	29	28	27	26	25	24
bit symbol	-	-	TBLBASE	TBLOFF				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TBLOFF							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBLOFF							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBLOFF	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-30	-	R	リードすると"0"が読めます。
29	TBLBASE	R/W	テーブルベース 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
28-7	TBLOFF	R/W	オフセット値 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。
6-0	-	R	リードすると"0"が読めます。

7.6.2.15 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

	31	30	29	28	27	26	25	24
bit symbol	VECTKEY/VECTKEYSTAT							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	VECTKEY/VECTKEYSTAT							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENDIANESS	-	-	-	-	PRIGROUP		
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	SYSRESET REQ	VECTCLR ACTIVE	VECTRESET
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	VECTKEY (ライト) VECTKEYSTAT (リード)	R/W	レジスタキー [ライト]このレジスタへ書き込みを行うには、<VECTKEY>に"0x05FA"を書き込む必要があります。 [リード]リードすると"0xFA05"が読めます。
15	ENDIANESS	R/W	エンディアン形式ビット(注 1) 1: ビッグエンディアン 0: リトルエンディアン
14-11	-	R	リードすると"0"が読めます。
10-8	PRIGROUP	R/W	割り込み優先度グループ分け 000: 横取り優先度 7bit、サブ優先度 1bit 001: 横取り優先度 6bit、サブ優先度 2bit 010: 横取り優先度 5bit、サブ優先度 3bit 011: 横取り優先度 4bit、サブ優先度 4bit 100: 横取り優先度 3bit、サブ優先度 5bit 101: 横取り優先度 2bit、サブ優先度 6bit 110: 横取り優先度 1bit、サブ優先度 7bit 111: 横取り優先度 0bit、サブ優先度 8bit 割り込み優先度レジスタ<PRI_n>を、横取り優先度とサブ優先度分けする際のビット構成を設定します。
7-3	-	R	リードすると"0"が読めます。
2	SYSRESET REQ	R/W	システムリセットリクエスト "1"をセットすると CPU が SYSRESETREQ 信号を出力します。(注 2)
1	VECTCLR ACTIVE	R/W	アクティブなベクタのクリア 1: アクティブな NMI、フォールト、割り込みのすべての状態の情報をクリアします。 0: クリアしません。 このビットは自身の動作によりクリアされます。 スタックの再初期化はアプリケーションで行う必要があります。
0	VECTRESET	R/W	システムリセット 1: システムをリセットします。 0: システムをリセットしません。 "1"をセットするとデバッグコンポーネント(FPB,DWT,ITM)以外の CPU 内部をリセットし、本ビットもクリアされます。

注 1) 本製品はリトルエンディアン固定です。

注 2) 本製品では、SYSRESETREQ が出力されるとウォームリセットが発生します。ウォームリセットにより<SYSRESETREQ>はクリアされます。

## 7.6.2.16 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4~7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。

	31	30	29	28	27	26	25	24
bit symbol	PRI_7			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	PRI_6			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	PRI_5			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PRI_4			-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-29	PRI_7	R/W	予約
28-24	-	R	リードすると"0"が読めます。
23-21	PRI_6	R/W	用法フォールト 優先度
20-16	-	R	リードすると"0"が読めます。
15-13	PRI_5	R/W	バスフォールト 優先度
12-8	-	R	リードすると"0"が読めます。
7-5	PRI_4	R/W	メモリ管理 優先度
4-0	-	R	リードすると"0"が読めます。



7.6.2.17 システムハンドラ制御および状態レジスタ

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	USGFAULT ENA	BUSFAULT ENA	MEMFAULT ENA
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	SVCALL PENDED	BUSFAULT PENDED	MEMFAULT PENDED	USGFAULT PENDED	SYSTICKACT	PENDSVACT	-	MONITOR ACT
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SVCALLACT	-	-	-	USGFAULT ACT	-	BUSFAULT ACT	MEMFAULT ACT
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-19	-	R	リードすると"0"が読めます。
18	USGFAULT ENA	R/W	用法フォールト 0: 禁止 1: 許可
17	BUSFAUL TENA	R/W	バスフォールト 0: 禁止 1: 許可
16	MEMFAULT ENA	R/W	メモリ管理 0: 禁止 1: 許可
15	SVCALL PENDED	R/W	SVCall 0: 保留されていない 1: 保留されている
14	BUSFAULT PENDED	R/W	バスフォールト 0: 保留されていない 1: 保留されている
13	MEMFAULT PENDED	R/W	メモリ管理 0: 保留されていない 1: 保留されている
12	USGFAULT PENDED	R/W	用法フォールト 0: 保留されていない 1: 保留されている
11	SYSTICKACT	R/W	SysTick 0: アクティブでない 1: アクティブ
10	PENDSVACT	R/W	PendSV 0: アクティブでない 1: アクティブ
9	-	R	リードすると"0"が読めます。
8	MONITORACT	R/W	デバッグモニタ 0: アクティブでない 1: アクティブ

Bit	Bit Symbol	Type	機能
7	SVCALLACT	R/W	SVCall 0: アクティブでない 1: アクティブ
6-4	-	R	リードすると"0"が読めます。
3	USGFAULT ACT	R/W	用法フォールト 0: アクティブでない 1: アクティブ
2	-	R	リードすると"0"が読めます。
1	BUSFAULT ACT	R/W	バスフォールト 0: アクティブでない 1: アクティブ
0	MEMFAULT ACT	R/W	メモリ管理 0: アクティブでない 1: アクティブ

注) アクティブビットの書き換えは、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。

### 7.6.3 クロックジェネレータレジスタ

#### 7.6.3.1 CGIMCGA(CG 割り込みモードコントロールレジスタ A)

	31	30	29	28	27	26	25	24
bit symbol	-	EMCG3			EMST3		-	INT3EN
リセット後	0	0	1	0	0	0	不定	0
	23	22	21	20	19	18	17	16
bit symbol	-	EMCG2			EMST2		-	INT2EN
リセット後	0	0	1	0	0	0	不定	0
	15	14	13	12	11	10	9	8
bit symbol	-	EMCG1			EMST1		-	INT1EN
リセット後	0	0	1	0	0	0	不定	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCG0			EMST0		-	INT0EN
リセット後	0	0	1	0	0	0	不定	0

Bit	Bit Symbol	Type	機能
31	-	R	リードすると"0"が読めます。
30-28	EMCG3[2:0]	R/W	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: "Low"レベル 001: "High"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
27-26	EMST3[1:0]	R	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
25	-	R	リードすると不定値が読まれます。
24	INT3EN	R/W	INT3 解除入力 0: ディセーブル 1: イネーブル
23	-	R	リードすると"0"が読めます。
22-20	EMCG2[2:0]	R/W	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: "Low"レベル 001: "High"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
19-18	EMST2[1:0]	R	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
17	-	R	リードすると不定値が読まれます。
16	INT2EN	R/W	INT2 解除入力 0: ディセーブル 1: イネーブル
15	-	R	リードすると"0"が読めます。

Bit	Bit Symbol	Type	機能
14-12	EMCG1[2:0]	R/W	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: "Low"レベル 001: "High"レベル 010: 立ち下がリエッジ 011: 立ち上がりエッジ 100: 両エッジ
11-10	EMST1[1:0]	R	INT1 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: 立ち下がリエッジ 11: 両エッジ
9	-	R	リードすると不定値が読まれます。
8	INT1EN	R/W	INT1 解除入力 0:ディセーブル 1: イネーブル
7	-	R	リードすると"0"が読めます。
6-4	EMCG0[2:0]	R/W	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: "Low"レベル 001: "High"レベル 010: 立ち下がリエッジ 011: 立ち上がりエッジ 100: 両エッジ
3-2	EMST0[1:0]	R	INT0 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: 立ち下がリエッジ 11: 両エッジ
1	-	R	リードすると不定値が読まれます。
0	INT0EN	R/W	INT0 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できます。CGICRCG 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。

7.6.3.2 CGIMCGB(CG 割り込みモードコントロールレジスタ B)

	31	30	29	28	27	26	25	24
bit symbol	-	EMCG7			EMST7		-	INT7EN
リセット後	0	0	1	0	0	0	不定	0
	23	22	21	20	19	18	17	16
bit symbol	-	EMCG6			EMST6		-	INT6EN
リセット後	0	0	1	0	0	0	不定	0
	15	14	13	12	11	10	9	8
bit symbol	-	EMCG5			EMST5		-	INT5EN
リセット後	0	0	1	0	0	0	不定	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCG4			EMST4		-	INT4EN
リセット後	0	0	1	0	0	0	不定	0

Bit	Bit Symbol	Type	機能
31	-	R	リードすると"0"が読めます。
30-28	EMCG7[2:0]	R/W	INTRMCRX0 スタンバイ解除要求のアクティブ状態を設定 <b>以下の設定で使用してください。</b> 011: 立ち上がりエッジ
27-26	EMST7[1:0]	R	INTRMCRX0 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
25	-	R	リードすると不定値が読まれます。
24	INT7EN	R/W	INTRMCRX0 解除入力 0:ディセーブル 1: イネーブル
23	-	R	リードすると"0"が読めます。
22-20	EMCG6[2:0]	R/W	INTCECRX スタンバイ解除要求のアクティブ状態を設定 <b>以下の設定で使用してください。</b> 011: 立ち上がりエッジ
19-18	EMST6[1:0]	R	INTCECRX 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
17	-	R	リードすると不定値が読まれます。
16	INT6EN	R/W	INTCECRX 解除入力 0:ディセーブル 1: イネーブル
15	-	R	リードすると"0"が読めます。
14-12	EMCG5[2:0]	R/W	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: "Low"レベル 001: "High"レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ

Bit	Bit Symbol	Type	機能
11-10	EMST5[1:0]	R	INT5 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
9	-	R	リードすると不定値が読めます。
8	INT5EN	R/W	INT5 解除入力 0:ディセーブル 1: イネーブル
7	-	R	リードすると"0"が読めます。
6-4	EMCG4[2:0]	R/W	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: "Low"レベル 001: "High"レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
3-2	EMST4[1:0]	R	INT4 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
1	-	R	リードすると不定値が読めます。
0	INT4EN	R/W	INT4 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できます。CGICRCG 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。

7.6.3.3 CGIMCGC(CG 割り込みモードコントロールレジスタ C)

	31	30	29	28	27	26	25	24
bit symbol	-	EMCGB			EMSTB		-	INTBEN
リセット後	0	0	1	0	0	0	不定	0
	23	22	21	20	19	18	17	16
bit symbol	-	EMCGA			EMSTA		-	INTAEN
リセット後	0	0	1	0	0	0	不定	0
	15	14	13	12	11	10	9	8
bit symbol	-	EMCG9			EMST9		-	INT9EN
リセット後	0	0	1	0	0	0	不定	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCG8			EMST8		-	INT8EN
リセット後	0	0	1	0	0	0	不定	0

Bit	Bit Symbol	Type	機能
31	-	R	リードすると"0"が読めます。
30-28	EMCGB[2:0]	R/W	INTRMCRX1 スタンバイ解除要求のアクティブ状態を設定 <b>以下の設定で使用してください。</b> 011: 立ち上がりエッジ
27-26	EMSTB[1:0]	R	INTRMCRX1 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
25	-	R	リードすると不定値が読まれます。
24	INTBEN	R/W	INTRMCRX1 解除入力 0:ディセーブル 1: イネーブル
23	-	R	リードすると"0"が読めます。
22-20	EMCGA[2:0]	R/W	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: "Low"レベル 001: "High"レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
19-18	EMSTA[1:0]	R	INT7 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
17	-	R	リードすると不定値が読まれます。
16	INTAEN	R/W	INT7 解除入力 0:ディセーブル 1: イネーブル
15	-	R	リードすると"0"が読めます。
14-12	EMCG9[2:0]	R/W	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: "Low"レベル 001: "High"レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ

Bit	Bit Symbol	Type	機能
11-10	EMST9[1:0]	R	INT6 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
9	-	R	リードすると不定値が読まれます。
8	INT9EN	R/W	INT6 解除入力 0:ディセーブル 1: イネーブル
7	-	R	リードすると"0"が読めます。
6-4	EMCG8[2:0]	R/W	INTRTC スタンバイ解除要求のアクティブ状態を設定 <b>以下の設定で使用してください。</b> 010: 立ち下がりエッジ
3-2	EMST8[1:0]	R	INTRTC 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
1	-	R	リードすると不定値が読まれます。
0	INT8EN	R/W	INTRTC 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できます。CGICRCG 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。



7.6.3.4 CGIMCGD(CG 割り込みモードコントロールレジスタ D)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	1	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	1	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	1	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCGC			EMSTC		-	INTCEN
リセット後	0	0	1	0	0	0	不定	0

Bit	Bit Symbol	Type	機能
31	-	R	リードすると"0"が読めます。
30-28	-	R/W	任意の値をライトしてください。
27-25	-	R	リードすると"0"が読めます。
24	-	R/W	"0"を書いてください。
23	-	R	リードすると"0"が読めます。
22-20	-	R/W	任意の値をライトしてください。
19-17	-	R	リードすると"0"が読めます。
16	-	R/W	"0"を書いてください。
15	-	R	リードすると"0"が読めます。
14-12	-	R/W	任意の値をライトしてください。
11-9	-	R	リードすると"0"が読めます。
8	-	R/W	"0"を書いてください。
7	-	R	リードすると"0"が読めます。
6-4	EMCGC[2:0]	R/W	INTCECTX スタンバイ解除要求のアクティブ状態を設定 <b>以下の設定で使用してください。</b> 011: 立ち上がりエッジ
3-2	EMSTC[1:0]	R	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
1	-	R	リードすると不定値が読まれます。
0	INTCEN	R/W	INTCECTX 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できます。CGICRCG 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。

## 7.6.3.5 CGICRCG(CG 割り込み要求クリアレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	ICRCG				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると"0"が読めます。
4-0	ICRCG[4:0]	W	割り込み要求をクリア 0_0000: INT0 0_0001: INT1 0_0010: INT2 0_0011: INT3 0_0100: INT4 0_0101: INT5 0_0110: INTCECRX 0_0111: INTRMCRX0 0_1000: INTRTC 0_1001: INT6 0_1010: INT7 0_1011: INTRMCRX1 0_1100: INTCECTX ※0_1101~1_1111: 設定禁止 ※リードすると"0"が読めます。

## 7.6.3.6 CGNMIFLG(NMI フラグレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	NMIFLG1	NMIFLG0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-2	-	R	リードすると"0"が読めます。
1	NMIFLG1	R	NMI 起動要因フラグ 0: 要因なし 1: $\overline{\text{NMI}}$ 端子による発生
0	NMIFLG0	R	NMI 起動要因フラグ 0: 要因なし 1: WDT による NMI 発生

注) <NMIFLG>は読み出すと"0"にクリアされます。

## 7.6.3.7 CGRSTFLG(リセットフラグレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
端子リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
端子リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
端子リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	SYSRSTF	-	WDTRSTF	PINRSTF	PONRSTF
端子リセット後	0	0	0	0	0	0	1	1 / 0

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると"0"が読めます。
4	SYSRSTF	R/W	デバッグリセットフラグ(注 1) 0: 0 ライト 1: SYSRESETREQ によるリセットによるリセットフラグ
3	-	R/W	"0"を書いてください。
2	WDTRSTF	R/W	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
1	PINRSTF	R/W	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
0	PONRSTF	R/W	パワーオンフラグ 0: 0 ライト 1: 電源投入時のリセットによるリセットフラグ

注 1) 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。

注 2) 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。

## 第 8 章 入出力ポート

### 8.1 ポート機能

#### 8.1.1 機能一覧

TMPM330FDFG/FYFG/FWFG には 78 本のポートがあり、ポート機能のほかに内蔵する周辺機能に対する入出力端子としても使用されます。

表 8-1、表 8-2、表 8-3 にポート機能の一覧を示します。

表 8-1 ポート機能一覧(ポート A ~ ポート C)

ポート	端子名	入出力	Pull-up Pull-down	Schmitt 入力	ノイズフィ ルタ	プログラ ムオープ ンドレイン	機能端子名
ポート A	PA0	入出力	Pull-up	o	-	-	TMS/ SWDIO
	PA1	入出力	Pull-down	-	-	-	TCK/ SWCLK
	PA2	入出力	Pull-up	-	-	-	TRACECLK
	PA3	入出力	Pull-up	-	-	-	TRACEDATA0
	PA4	入出力	Pull-up	-	-	-	TRACEDATA1
	PA5	入出力	Pull-up	-	-	-	TRACEDATA2
	PA6	入出力	Pull-up	-	-	-	TRACEDATA3
ポート B	PA7	入出力	Pull-up	-	-	-	-
	PB0	入出力	Pull-up	-	-	-	TDO/ SWV
	PB1	入出力	Pull-up	-	-	-	TDI
	PB2	入出力	Pull-up	o	-	-	TRST
	PB3	入出力	Pull-up	-	-	-	-
	PB4	入出力	Pull-up	-	-	-	-
	PB5	入出力	Pull-up	-	-	-	-
ポート C	PB6	入出力	Pull-up	-	-	-	-
	PB7	入出力	Pull-up	-	-	-	-
	PC0	入力	Pull-up	-	-	-	AIN0
	PC1	入力	Pull-up	-	-	-	AIN1
	PC2	入力	Pull-up	-	-	-	AIN2
	PC3	入力	Pull-up	-	-	-	AIN3

o: あり

-: なし

注) ノイズフィルタのノイズ除去幅は、Typ.条件で約 30ns です。

表 8-2 ポート機能一覧(ポート D ~ ポート G)

ポート	端子名	入出力	Pull-up Pull-down	Schmitt 入力	ノイズフィ ルタ	プログラ ムオープ ンドレイン	機能端子名
ポート D	PD0	入力	Pull-up	-	-	-	AIN4, TB5IN0
	PD1	入力	Pull-up	-	-	-	AIN5, TB5IN1
	PD2	入力	Pull-up	-	-	-	AIN6, TB6IN0
	PD3	入力	Pull-up	-	-	-	AIN7, TB6IN1
	PD4	入力	Pull-up	-	-	-	AIN8
	PD5	入力	Pull-up	-	-	-	AIN9
	PD6	入力	Pull-up	-	-	-	AIN10
	PD7	入力	Pull-up	-	-	-	AIN11
ポート E	PE0	入出力	Pull-up	-	-	o	TXD0
	PE1	入出力	Pull-up	o	-	o	RXD0
	PE2	入出力	Pull-up	o	-	o	SCLK0, CTS0
	PE3	入出力	Pull-up	o	-	o	RXIN0
	PE4	入出力	Pull-up	-	-	o	TXD1
	PE5	入出力	Pull-up	o	-	o	RXD1
	PE6	入出力	Pull-up	o	-	o	SCLK1, CTS1
ポート F	PF0	入出力	Pull-up	-	-	o	TXD2
	PF1	入出力	Pull-up	o	-	o	RXD2
	PF2	入出力	Pull-up	o	-	o	SCLK2, CTS2
	PF3	入出力	Pull-up	o	-	o	RXIN1
	PF4	入出力	Pull-up	o	-	o	SDA1/ SO1
	PF5	入出力	Pull-up	o	-	o	SCL1/ SI1
	PF6	入出力	Pull-up	o	-	o	SCK1
	PF7	入出力	Pull-up	o	o	o	INT5
ポート G	PG0	入出力	Pull-up	o	-	o	SDA0/ SO0
	PG1	入出力	Pull-up	o	-	o	SCL0/ SI0
	PG2	入出力	Pull-up	o	-	o	SCK0
	PG3	入出力	Pull-up	o	o	o	INT4
	PG4	入出力	Pull-up	o	-	o	SDA2/ SO2
	PG5	入出力	Pull-up	o	-	o	SCL2/ SI2
	PG6	入出力	Pull-up	o	-	o	SCK2
	PG7	入出力	Pull-up	-	-	o	TB8OUT

o: あり

-: なし

注) ノイズフィルタのノイズ除去幅は、Typ.条件で約 30ns です。

表 8-3 ポート機能一覧(ポート H ~ ポート K)

ポート	端子名	入出力	Pull-up Pull-down	Schmitt 入力	ノイズフィ ルタ	プログラマ ブルオーブ ンドレイン	機能端子名
ポート H	PH0	入出力	Pull-up	o	-	-	TB0IN0, BOOT
	PH1	入出力	Pull-up	o	-	-	TB0IN1
	PH2	入出力	Pull-up	o	-	-	TB1IN0
	PH3	入出力	Pull-up	o	-	-	TB1IN1
	PH4	入出力	Pull-up	o	-	-	TB2IN0
	PH5	入出力	Pull-up	o	-	-	TB2IN1
	PH6	入出力	Pull-up	o	-	-	TB3IN0
	PH7	入出力	Pull-up	o	-	-	TB3IN1
ポート I	PI0	入出力	Pull-up	-	-	-	TB0OUT
	PI1	入出力	Pull-up	-	-	-	TB1OUT
	PI2	入出力	Pull-up	-	-	-	TB2OUT
	PI3	入出力	Pull-up	-	-	-	TB3OUT
	PI4	入出力	Pull-up	-	-	-	TB4OUT
	PI5	入出力	Pull-up	-	-	-	TB5OUT
	PI6	入出力	Pull-up	o	-	-	TB4IN0
	PI7	入出力	Pull-up	o	-	-	TB4IN1
ポート J	PJ0	入出力	Pull-up	o	o	-	INT0
	PJ1	入出力	Pull-up	o	o	-	INT1
	PJ2	入出力	Pull-up	o	o	-	INT2
	PJ3	入出力	Pull-up	o	o	-	INT3
	PJ4	入出力	Pull-up	-	-	-	TB6OUT
	PJ5	入出力	Pull-up	-	-	-	TB7OUT
	PJ6	入出力	Pull-up	o	o	-	INT6
	PJ7	入出力	Pull-up	o	o	-	INT7
ポート K	PK0	入出力	-	o	-	o (注 1)	CEC
	PK1	入出力	Pull-up	-	-	-	SCOUT, ALARM
	PK2	入出力	Pull-up	-	-	-	TB9OUT

o: あり

-: なし

注 1) Nch オープンドレイン端子です。

注 2) ノイズフィルタのノイズ除去幅は、Typ.条件で約 30ns です。

### 8.1.2 ポートレジスタ概略説明

ポートを使用するには以下のレジスタを設定する必要があります。

- **PxDATA**: ポート x データレジスタ  
ポートのデータ読み込み、データ書き込みを行います。
- **PxCR**: ポート x 出力コントロールレジスタ  
出力の制御を行います。  
入力の制御は **PxIE** で設定してください。
- **PxFRn**: ポート x ファンクションレジスタ n  
機能設定を行いません。  
"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**: ポート x オープンドレインコントロールレジスタ  
プログラマブルオープンドレインの制御を行います。  
プログラマブルオープンドレインは、**PxOD** の設定により、出力データが"1"の場合に出力バッファをディセーブルにし、擬似的にオープンドレインを実現する機能です。
- **PxPUP**: ポート x プルアップコントロールレジスタ  
プログラマブルプルアップを制御します。
- **PxPDN**: ポート x プルダウンコントロールレジスタ  
プログラマブルプルダウンを制御します。
- **PxIE**: ポート x 入力コントロールレジスタ  
入力の制御を行いません。貫通電流対策のため、初期状態は入力禁止になっています。



### 8.1.3 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 CGSTBYCR<DRVE>で制御することができます。

PxIE, PxCR が許可で、<DRVE>を"1"に設定した場合、STOP モード中も入力, 出力が許可となります。<DRVE>を"0"に設定した場合、一部のポートを除き、PxIE, PxCR が許可であっても STOP モード中は入力, 出力が禁止になります。

STOP モード時の端子状態を表 8-4 に示します。

表 8-4 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	"High"レベル出力	"High"レベル出力
	RESET, NMI, MODE	入力専用	o	o
ポート	PA0, PB0 (デバッグ機能設定 PxFRn<PxmFn>=1 かつ出力許可設定 PxCR<PxmC>=1 の場合)(注)	入力	×	設定(PxIE[m])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	
	PF7, PG3, PJ0~3, PJ6, PJ7 (割り込み機能設定 PxFRn<PxmFn>=1 かつ入力許可設定 PxIE<PxmlE>=1 の場合)(注)	入力	o	o
		出力	×	設定(PxCR[m])に従う
	上記以外のポート端子	入力	×	設定(PxIE[m])に従う
出力		×	設定(PxCR[m])に従う	

o: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。

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

### 8.1.4 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まま STOP/SLEEP モードに遷移すると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。

## 8.2 ポート機能詳細

本章では、各ポートのレジスタの詳細について説明します。

回路構成については、本章では「回路タイプ」のみ記載しています。具体的な回路図は「8.3 ポート回路図」に記載していますので、そちらを参照してください。

### 8.2.1 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インタフェース、デバッグトレース出力があります。

リセット後 PA0 と PA1 はデバッグインタフェースとなります。PA0 は TMS または SWDIO で入力・出力・プルアップ許可、PA1 は TCK または SWCLK で入力・プルダウン許可となります。PA2 ~ PA7 は汎用ポートとなり、入力・出力・プルアップは禁止となります。

注 1) PA0 が TMS/SWDIO 設定の場合、CGSTBYCR<DRVE>の設定によらず、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 TCK/SWCLK 設定の場合十分な低消費電力効果が得られません。TCK/SWCLK を使用しないときは、PA1 をポート設定にしてください。

#### 8.2.1.1 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

#### 8.2.1.2 ポート A レジスタ一覧

Base Address = 0x4000\_0000

レジスタ名		Address(Base+)
ポート A データ レジスタ	PADATA	0x0000
ポート A 出力コントロールレジスタ	PACR	0x0004
ポート A ファンクションレジスタ 1	PAFR1	0x0008
ポート A プルアップコントロールレジスタ	PAPUP	0x002C
ポート A プルダウンコントロールレジスタ	PAPDN	0x0030
ポート A 入力コントロールレジスタ	PAIE	0x0038

## 8.2.1.3 PADATA (ポート A データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PA7-PA0	R/W	ポート A データレジスタ

## 8.2.1.4 PACR (ポート A 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7C	PA6C	PA5C	PA4C	PA3C	PA2C	PA1C	PA0C
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PA7C-PA0C	R/W	出力 0: 禁止 1: 許可

## 8.2.1.5 PAFR1 (ポート A ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PA6F1	PA5F1	PA4F1	PA3F1	PA2F1	PA1F1	PA0F1
リセット後	0	0	0	0	0	0	1	1

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6	PA6F1	R/W	0: PORT 1: TRACEDATA3
5	PA5F1	R/W	0: PORT 1: TRACEDATA2
4	PA4F1	R/W	0: PORT 1: TRACEDATA1
3	PA3F1	R/W	0: PORT 1: TRACEDATA0
2	PA2F1	R/W	0: PORT 1: TRACECLK
1	PA1F1	R/W	0: PORT 1: TCK/SWCLK
0	PA0F1	R/W	0: PORT 1: TMS/SWDIO

## 8.2.1.6 PAPUP (ポート A ブルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7UP	PA6UP	PA5UP	PA4UP	PA3UP	PA2UP	-	PA0UP
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-2	PA7UP-PA2UP	R/W	ブルアップ 0: 禁止 1: 許可
1	-	R	リードすると"0"が読めます。
0	PA0UP	R/W	ブルアップ 0: 禁止 1: 許可

## 8.2.1.7 PAPDN (ポート A ブルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PA1DN	-
リセット後	0	0	0	0	0	0	1	0

Bit	Bit Symbol	Type	機能
31-2	-	R	リードすると"0"が読めます。
1	PA1DN	R/W	ブルダウン 0: 禁止 1: 許可
0	-	R	リードすると"0"が読めます。

## 8.2.1.8 PAIE (ポート A 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7IE	PA6IE	PA5IE	PA4IE	PA3IE	PA2IE	PA1IE	PA0IE
リセット後	0	0	0	0	0	0	1	1

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PA7IE-PA0IE	R/W	入力 0: 禁止 1: 許可

## 8.2.2 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インタフェースがあります。

リセット後 PB0, PB1, PB2 はデバッグインタフェースになります。PB0 は TDO または SWV で出力許可、PB1 は TDI で入力・プルアップ許可、PB2 は  $\overline{\text{TRST}}$  で入力・プルアップ許可となります。

PB3 ~ PB7 は汎用ポートとなり、入力・出力・プルアップは禁止となります。

注) PB0 が TDO/SWV 設定の場合、CGSTBYCR<DRVE>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。

### 8.2.2.1 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

### 8.2.2.2 ポート B レジスタ一覧

		Base Address = 0x4000_0040
レジスタ名		Address(Base+)
ポート B データレジスタ	PBDATA	0x0000
ポート B 出力コントロールレジスタ	PBCR	0x0004
ポート B ファンクションレジスタ 1	PBFR1	0x0008
ポート B プルアップコントロールレジスタ	PBPUP	0x002C
ポート B 入力コントロールレジスタ	PBIE	0x0038

## 8.2.2.3 PBDATA (ポート B データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PB7-PB0	R/W	ポート B データ レジスタ

## 8.2.2.4 PBCR (ポート B 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PB7C	PB6C	PB5C	PB4C	PB3C	PB2C	PB1C	PB0C
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PB7C-PB0C	R/W	出力 0: 禁止 1: 許可



8.2.2.5 PBFR1 (ポート B ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PB2F1	PB1F1	PB0F1
リセット後	0	0	0	0	0	1	1	1

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2	PB2F1	R/W	0: PORT 1: $\overline{\text{TRST}}$
1	PB1F1	R/W	0: PORT 1: TDI
0	PB0F1	R/W	0: PORT 1: TDO/SWV

## 8.2.2.6 PBPUP (ポート B プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PB7UP	PB6UP	PB5UP	PB4UP	PB3UP	PB2UP	PB1UP	PB0UP
リセット後	0	0	0	0	0	1	1	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PB7UP-PB0UP	R/W	プルアップ 0: 禁止 1: 許可

## 8.2.2.7 PBIE (ポート B 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PB7IE	PB6IE	PB5IE	PB4IE	PB3IE	PB2IE	PB1IE	PB0IE
リセット後	0	0	0	0	0	1	1	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PB7IE-PB0IE	R/W	入力 0: 禁止 1: 許可

### 8.2.3 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 AD コンバータのアナログ入力端子機能があります。

リセット後は汎用入力ポートとなり、入力・プルアップは禁止となります。

AD コンバータのアナログ入力として使用する場合、PCIE で入力禁止、PCPUP でプルアップ禁止にしてください。

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、AD 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。

#### 8.2.3.1 ポート C 回路タイプ

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

#### 8.2.3.2 ポート C レジスタ一覧

Base Address = 0x4000\_0080

レジスタ名		Address(Base+)
ポート C データレジスタ	PCDATA	0x0000
ポート C プルアップコントロールレジスタ	PCPUP	0x002C
ポート C 入力コントロールレジスタ	PCIE	0x0038

## 8.2.3.3 PCDATA (ポート C データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PC3	PC2	PC1	PC0
リセット後	0	0	0	0	1	1	1	1

Bit	Bit Symbol	Type	機能
31-4	-	R	リードすると"0"が読めます。
3-0	PC3-PC0	R	ポート C データレジスタ

## 8.2.3.4 PCPUP (ポート C プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PC3UP	PC2UP	PC1UP	PC0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-4	-	R	リードすると"0"が読めます。
3-0	PC3UP-PC0UP	R/W	プルアップ 0: 禁止 1: 許可

8.2.3.5 PCIE (ポート C 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PC3IE	PC2IE	PC1IE	PC0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-4	-	R	リードすると"0"が読めます。
3-0	PC3IE-PC0IE	R/W	入力 0: 禁止 1: 許可

## 8.2.4 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 AD 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。

リセット後は汎用入力ポートとなり、入力・プルアップは禁止となります。

16 ビットタイマ入力として使用する場合には、PDFR1 と PDIE を設定してください。

AD コンバータのアナログ入力として使用する場合、PDIE で入力禁止、PDPUP でプルアップ禁止にして下さい。

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、AD 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。

### 8.2.4.1 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

### 8.2.4.2 ポート D レジスタ一覧

Base Address = 0x4000\_00C0

レジスタ名		Address(Base+)
ポート D データレジスタ	PDDATA	0x0000
ポート D ファンクションレジスタ 1	PDFR1	0x0008
ポート D プルアップコントロールレジスタ	PDPUP	0x002C
ポート D 入力コントロールレジスタ	PDIE	0x0038

8.2.4.3 PDDATA (ポート D データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
リセット後	1	1	1	1	1	1	1	1

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PD7-PD0	R	ポート D データレジスタ

8.2.4.4 PDFR1 (ポート D ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PD3F1	PD2F1	PD1F1	PD0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-4	-	R	リードすると"0"が読めます。
3	PD3F1	R/W	0: PORT 1: TB6IN1
2	PD2F1	R/W	0: PORT 1: TB6IN0
1	PD1F1	R/W	0: PORT 1: TB5IN1
0	PD0F1	R/W	0: PORT 1: TB5IN0

## 8.2.4.5 PDPUP (ポート D プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PD7UP	PD6UP	PD5UP	PD4UP	PD3UP	PD2UP	PD1UP	PD0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PD7UP-PD0UP	R/W	プルアップ 0: 禁止 1: 許可

## 8.2.4.6 PDIE (ポート D 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PD7IE	PD6IE	PD5IE	PD4IE	PD3IE	PD2IE	PD1IE	PD0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PD7IE-PD0IE	R/W	入力 0: 禁止 1: 許可



## 8.2.5 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。

リセット後は汎用ポートとなり、入力・出力・プルアップは禁止となります。

ポート E では 2 種類のファンクションレジスタがあります。汎用ポート機能として使用する場合は、両方のファンクションレジスタの該当するビットに"0"を設定してください。汎用ポート機能以外として使用する場合には、ファンクションレジスタの該当ビットに"1"を設定してください。両方のファンクションレジスタを同時に"1"を設定しないで下さい。

### 8.2.5.1 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

### 8.2.5.2 ポート E レジスタ一覧

Base Address = 0x4000\_0100

レジスタ名		Address(Base+)
ポート E データ レジスタ	PEDATA	0x0000
ポート E 出力コントロールレジスタ	PECR	0x0004
ポート E ファンクションレジスタ 1	PEFR1	0x0008
ポート E ファンクションレジスタ 2	PEFR2	0x000C
ポート E オープンドレインコントロールレジスタ	PEOD	0x0028
ポート E プルアップコントロールレジスタ	PEPUP	0x002C
ポート E 入力コントロールレジスタ	PEIE	0x0038

## 8.2.5.3 PEDATA (ポート E データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PE6	PE5	PE4	PE3	PE2	PE1	PE0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6-0	PE6-PE0	R/W	ポート E データレジスタ

## 8.2.5.4 PECR (ポート E 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PE6C	PE5C	PE4C	PE3C	PE2C	PE1C	PE0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6-0	PE6C-PE0C	R/W	出力 0: 禁止 1: 許可

8.2.5.5 PEFR1(ポート E ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PE6F1	PE5F1	PE4F1	PE3F1	PE2F1	PE1F1	PE0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6	PE6F1	R/W	0: PORT 1: SCLK1
5	PE5F1	R/W	0: PORT 1: RXD1
4	PE4F1	R/W	0: PORT 1: TXD1
3	PE3F1	R/W	0: PORT 1: RXIN0
2	PE2F1	R/W	0: PORT 1: SCLK0
1	PE1F1	R/W	0: PORT 1: RXD0
0	PE0F1	R/W	0: PORT 1: TXD0

## 8.2.5.6 PEFR2(ポート E ファンクションレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PE6F2	-	-	-	PE2F2	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6	PE6F2	R/W	0: PORT 1: CTS1
5-3	-	R	リードすると"0"が読めます。
2	PE2F2	R/W	0: PORT 1: CTS0
1-0	-	R	リードすると"0"が読めます。

## 8.2.5.7 PEOD(ポート E オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PE6OD	PE5OD	PE4OD	PE3OD	PE2OD	PE1OD	PE0OD
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6-0	PE6OD- PE0OD	R/W	0: CMOS 1: オープンドレイン

## 8.2.5.8 PEPUP(ポート E プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PE6UP	PE5UP	PE4UP	PE3UP	PE2UP	PE1UP	PE0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6-0	PE6UP-PE0UP	R/W	プルアップ 0: 禁止 1: 許可

## 8.2.5.9 PEIE(ポート E 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PE6IE	PE5IE	PE4IE	PE3IE	PE2IE	PE1IE	PE0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6-0	PE6IE-PE0IE	R/W	入力 0: 禁止 1: 許可

## 8.2.6 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。

リセット後は汎用ポートとなり、入力・出力・プルアップは禁止となります。

ポート F では 2 種類のファンクションレジスタがあります。汎用ポート機能として使用する場合は、両方のファンクションレジスタの該当するビットに"0"を設定してください。汎用ポート機能以外として使用する場合には、ファンクションレジスタの該当ビットに"1"を設定してください。両方のファンクションレジスタを同時に"1"を設定しないで下さい。

割り込み入力を STOP モード解除に使用する場合、PFFR1 で機能設定にし、PFIE 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 CGSTBYCR<DRVE>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、PFIE で入力許可設定であれば PFFR 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。

### 8.2.6.1 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

### 8.2.6.2 ポート F レジスタ一覧

Base Address = 0x4000\_0140

レジスタ名		Address(Base+)
ポート F データレジスタ	PFDATA	0x0000
ポート F 出力コントロールレジスタ	PFCR	0x0004
ポート F ファンクションレジスタ 1	PFFR1	0x0008
ポート F ファンクションレジスタ 2	PFFR2	0x000C
ポート F オープンドレインコントロールレジスタ	PFOD	0x0028
ポート F プルアップコントロールレジスタ	PFPUP	0x002C
ポート F 入力コントロールレジスタ	PFIE	0x0038

## 8.2.6.3 PFDATA (ポート F データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PF7-PF0	R/W	ポート F データレジスタ

## 8.2.6.4 PFCR(ポート F 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PF7C	PF6C	PF5C	PF4C	PF3C	PF2C	PF1C	PF0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PF7C-PF0C	R/W	出力 0: 禁止 1: 許可

## 8.2.6.5 PFFR1(ポート F ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PF7F1	PF6F1	PF5F1	PF4F1	PF3F1	PF2F1	PF1F1	PF0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	PF7F1	R/W	0: PORT 1: INT5
6	PF6F1	R/W	0: PORT 1: SCK1
5	PF5F1	R/W	0: PORT 1: SI1/SCL1
4	PF4F1	R/W	0: PORT 1: SO1/SDA1
3	PF3F1	R/W	0: PORT 1: RXIN1
2	PF2F1	R/W	0: PORT 1: SCLK2
1	PF1F1	R/W	0: PORT 1: RXD2
0	PF0F1	R/W	0: PORT 1: TXD2



8.2.6.6 PFFR2(ポート F ファンクションレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PF2F2	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2	PF2F2	R/W	0: PORT 1: $\overline{\text{CTS2}}$
1-0	-	R	リードすると"0"が読めます。

8.2.6.7 PFOOD(ポート F オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PF7OD	PF6OD	PF5OD	PF4OD	PF3OD	PF2OD	PF1OD	PF0OD
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PF7OD-PF0OD	R/W	0: CMOS 1: オープンドレイン

## 8.2.6.8 PFPUP(ポートF プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PF7UP	PF6UP	PF5UP	PF4UP	PF3UP	PF2UP	PF1UP	PF0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PF7UP-PF0UP	R/W	プルアップ 0: 禁止 1: 許可

## 8.2.6.9 PFIE(ポートF 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PF7IE	PF6IE	PF5IE	PF4IE	PF3IE	PF2IE	PF1IE	PF0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PF7IE-PF0IE	R/W	入力 0: 禁止 1: 許可

## 8.2.7 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, 外部割り込み入力機能, 16 ビットタイマ出力機能があります。

リセット後は汎用ポートとなり、入力・出力・プルアップは禁止となります。

割り込み入力を STOP モード解除に使用する場合、PGFR1 で機能設定にし、PGIE 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 CGSTBYCR<DRVE>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 で入力許可設定であれば 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。

### 8.2.7.1 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

### 8.2.7.2 ポート G レジスタ一覧

Base Address = 0x4000\_0180

レジスタ名		Address(Base+)
ポート G データレジスタ	PGDATA	0x0000
ポート G 出カコントロールレジスタ	PGCR	0x0004
ポート G ファンクションレジスタ 1	PGFR1	0x0008
Reserved	-	0x0010
ポート G オープンドレインコントロールレジスタ	PGOD	0x0028
ポート G プルアップコントロールレジスタ	PGPUP	0x002C
ポート G 入カコントロールレジスタ	PGIE	0x0038

注) "Reserved"表記のアドレスにはアクセスしないでください。

## 8.2.7.3 PGDATA(ポート G データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PG7-PG0	R/W	ポート G データレジスタ

## 8.2.7.4 PGCR(ポート G 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PG7C	PG6C	PG5C	PG4C	PG3C	PG2C	PG1C	PG0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PG7C-PG0C	R/W	出力 0: 禁止 1: 許可

8.2.7.5 PGFR1(ポート G ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PG7F1	PG6F1	PG5F1	PG4F1	PG3F1	PG2F1	PG1F1	PG0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	PG7F1	R/W	0: PORT 1: TB8OUT
6	PG6F1	R/W	0: PORT 1: SCK2
5	PG5F1	R/W	0: PORT 1: SI2/SCL2
4	PG4F1	R/W	0: PORT 1: SO2/SDA2
3	PG3F1	R/W	0: PORT 1: INT4
2	PG2F1	R/W	0: PORT 1: SCK0
1	PG1F1	R/W	0: PORT 1: SI0/SCL0
0	PG0F1	R/W	0: PORT 1: SO0/SDA0

## 8.2.7.6 PGOD(ポート G オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PG7OD	PG6OD	PG5OD	PG4OD	PG3OD	PG2OD	PG1OD	PG0OD
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PG7OD- PG0OD	R/W	0: CMOS 1: オープンドレイン

## 8.2.7.7 PGPUP(ポート G プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PG7UP	PG6UP	PG5UP	PG4UP	PG3UP	PG2UP	PG1UP	PG0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PG7UP- PG0UP	R/W	プルアップ 0: 禁止 1: 許可

8.2.7.8 PGIE(ポート G 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PG7IE	PG6IE	PG5IE	PG4IE	PG3IE	PG2IE	PG1IE	PG0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PG7IE-PG0IE	R/W	入力 0: 禁止 1: 許可

## 8.2.8 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"Low"の期間、PH0( $\overline{\text{BOOT}}$ )は入力とプルアップが許可となっており、リセット信号の立ち上がりで PH0 が"High"の場合、シングルチップモードとなって内蔵 Flash メモリから起動し、PH0 が"Low"の場合、シングル BOOT モードとなって内蔵 BOOT ROM から起動します。シングル BOOT モードの説明は、「Flash 動作説明」の章を参照してください。

リセット後は PH0~PH7 は汎用ポートとなり、入力・出力は禁止となります。プルアップは、PH0 は許可, PH1~PH7 は禁止となります。

### 8.2.8.1 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

### 8.2.8.2 ポート H レジスタ一覧

Base Address = 0x4000\_01C0

レジスタ名		Address(Base+)
ポート H データレジスタ	PHDATA	0x0000
ポート H 出カコントロールレジスタ	PHCR	0x0004
ポート H ファンクションレジスタ 1	PHFR1	0x0008
Reserved	-	0x0010
ポート H プルアップコントロールレジスタ	PHPUP	0x002C
ポート H 入カコントロールレジスタ	PHIE	0x0038

注) "Reserved"表記のアドレスにはアクセスしないでください。



8.2.8.3 PHDATA (ポート H データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PH7-PH0	R/W	ポート H データレジスタ

8.2.8.4 PHCR(ポート H 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PH7C	PH6C	PH5C	PH4C	PH3C	PH2C	PH1C	PH0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PH7C-PH0C	R/W	出力 0: 禁止 1: 許可

## 8.2.8.5 PHFR1(ポートH ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PH7F1	PH6F1	PH5F1	PH4F1	PH3F1	PH2F1	PH1F1	PH0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	PH7F1	R/W	0: PORT 1: TB3IN1
6	PH6F1	R/W	0: PORT 1: TB3IN0
5	PH5F1	R/W	0: PORT 1: TB2IN1
4	PH4F1	R/W	0: PORT 1: TB2IN0
3	PH3F1	R/W	0: PORT 1: TB1IN1
2	PH2F1	R/W	0: PORT 1: TB1IN0
1	PH1F1	R/W	0: PORT 1: TB0IN1
0	PH0F1	R/W	0: PORT 1: TB0IN0

8.2.8.6 PHPUP(ポート H プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PH7UP	PH6UP	PH5UP	PH4UP	PH3UP	PH2UP	PH1UP	PH0UP
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PH7UP-PH0UP	R/W	プルアップ 0: 禁止 1: 許可

8.2.8.7 PHIE(ポート H 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PH7IE	PH6IE	PH5IE	PH4IE	PH3IE	PH2IE	PH1IE	PH0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PH7IE-PH0IE	R/W	入力 0: 禁止 1: 許可

## 8.2.9 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。

リセット後は汎用ポートとなり、入力・出力・プルアップは禁止となります。

### 8.2.9.1 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

### 8.2.9.2 ポート I レジスタ一覧

Base Address = 0x4000\_0200

レジスタ名		Address(Base+)
ポート I データレジスタ	PIDATA	0x0000
ポート I 出力コントロールレジスタ	PICR	0x0004
ポート I ファンクションレジスタ 1	PIFR1	0x0008
Reserved	-	0x0010
ポート I プルアップコントロールレジスタ	PIPUP	0x002C
ポート I 入力コントロールレジスタ	PIIE	0x0038

注) "Reserved"表記のアドレスにはアクセスしないでください。

## 8.2.9.3 PIDATA(ポートIデータレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PI7	PI6	PI5	PI4	PI3	PI2	PI1	PI0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PI7-PI0	R/W	ポートIデータレジスタ

## 8.2.9.4 PICR(ポートI出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PI7C	PI6C	PI5C	PI4C	PI3C	PI2C	PI1C	PI0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PI7C-PI0C	R/W	出力 0: 禁止 1: 許可

## 8.2.9.5 PIFR1(ポートIファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PI7F1	PI6F1	PI5F1	PI4F1	PI3F1	PI2F1	PI1F1	PI0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	PI7F1	R/W	0: PORT 1: TB4IN1
6	PI6F1	R/W	0: PORT 1: TB4IN0
5	PI5F1	R/W	0: PORT 1: TB5OUT
4	PI4F1	R/W	0: PORT 1: TB4OUT
3	PI3F1	R/W	0: PORT 1: TB3OUT
2	PI2F1	R/W	0: PORT 1: TB2OUT
1	PI1F1	R/W	0: PORT 1: TB1OUT
0	PI0F1	R/W	0: PORT 1: TB0OUT

## 8.2.9.6 PIPUP(ポートIプルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PI7UP	PI6UP	PI5UP	PI4UP	PI3UP	PI2UP	PI1UP	PI0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PI7UP-PI0UP	R/W	プルアップ 0: 禁止 1: 許可

## 8.2.9.7 PIIE(ポートI入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PI7IE	PI6IE	PI5IE	PI4IE	PI3IE	PI2IE	PI1IE	PI0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PI7IE-PI0IE	R/W	入力 0: 禁止 1: 許可

### 8.2.10 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、外部割り込み入力機能があります。

リセット後は汎用ポートとなり、入力・出力・プルアップは禁止状態になります。

割り込み入力を STOP モード解除に使用する場合、PJFR1 で機能設定にし、PJIE 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 CGSTBYCR<DRVE>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、PJIE で入力許可設定であれば PJFR 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。

#### 8.2.10.1 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

#### 8.2.10.2 ポート J レジスタ一覧

Base Address = 0x4000\_0240

レジスタ名		Address(Base+)
ポート J データ レジスタ	PJDATA	0x0000
ポート J 出カコントロールレジスタ	PJCR	0x0004
ポート J ファンクションレジスタ 1	PJFR1	0x0008
Reserved	-	0x0010
ポート J プルアップコントロールレジスタ	PJPUP	0x002C
ポート J 入力コントロールレジスタ	PJIE	0x0038

注) "Reserved"表記のアドレスにはアクセスしないでください。



8.2.10.3 PJDATA (ポート J データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PJ7-PJ0	R/W	ポート J データレジスタ

8.2.10.4 PJCR(ポート J 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PJ7C	PJ6C	PJ5C	PJ4C	PJ3C	PJ2C	PJ1C	PJ0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PJ7C-PJ0C	R/W	出力 0: 禁止 1: 許可

## 8.2.10.5 PJFR1(ポート J ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PJ7F1	PJ6F1	PJ5F1	PJ4F1	PJ3F1	PJ2F1	PJ1F1	PJ0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	PJ7F1	R/W	0: PORT 1: INT7
6	PJ6F1	R/W	0: PORT 1: INT6
5	PJ5F1	R/W	0: PORT 1: TB7OUT
4	PJ4F1	R/W	0: PORT 1: TB6OUT
3	PJ3F1	R/W	0: PORT 1: INT3
2	PJ2F1	R/W	0: PORT 1: INT2
1	PJ1F1	R/W	0: PORT 1: INT1
0	PJ0F1	R/W	0: PORT 1: INT0

8.2.10.6 PJUP(ポート J プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PJ7UP	PJ6UP	PJ5UP	PJ4UP	PJ3UP	PJ2UP	PJ1UP	PJ0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PJ7UP-PJ0UP	R/W	プルアップ 0: 禁止 1: 許可

8.2.10.7 PJIE(ポート J 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PJ7IE	PJ6IE	PJ5IE	PJ4IE	PJ3IE	PJ2IE	PJ1IE	PJ0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PJ7IE-PJ0IE	R/W	入力 0: 禁止 1: 許可

### 8.2.11 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。

リセット後は汎用ポートとなり、入力・出力・プルアップは禁止となります。

注) 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。

#### 8.2.11.1 ポート K 回路タイプ

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

#### 8.2.11.2 ポート K レジスタ一覧

Base Address = 0x4000\_0280

レジスタ名		Address(Base+)
ポート K データ レジスタ	PKDATA	0x0000
ポート K 出力コントロールレジスタ	PKCR	0x0004
ポート K ファンクションレジスタ 1	PKFR1	0x0008
ポート K ファンクションレジスタ 2	PKFR2	0x000C
ポート K プルアップコントロールレジスタ	PKPUP	0x002C
ポート K 入力コントロールレジスタ	PKIE	0x0038

8.2.11.3 PKDATA(ポート K データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PK2	PK1	PK0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2-0	PK2-PK0	R/W	ポート K データレジスタ

8.2.11.4 PKCR(ポート K 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PK2C	PK1C	PK0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2-0	PK2C-PK0C	R/W	出力 0: 禁止 1: 許可

## 8.2.11.5 PKFR1(ポート K ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PK2F1	PK1F1	PK0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2	PK2F1	R/W	0: PORT 1: TB9OUT
1	PK1F1	R/W	0: PORT 1: SCOUT
0	PK0F1	R/W	0: PORT 1: CEC

## 8.2.11.6 PKFR2(ポート K ファンクションレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PK1F2	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-2	-	R	リードすると"0"が読めます。
1	PK1F2	R/W	0: PORT 1: ALARM
0	-	R	リードすると"0"が読めます。

## 8.2.11.7 PKPUP(ポート K プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PK2UP	PK1UP	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2-1	PK2UP-PK1UP	R/W	プルアップ 0: 禁止 1: 許可
0	-	R	リードすると"0"が読めます。

## 8.2.11.8 PKIE(ポート K 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PK2IE	PK1IE	PK0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2-0	PK2IE-PK0IE	R/W	入力 0: 禁止 1: 許可

## 8.3 ポート回路図

### 8.3.1 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。

図中の点線は「ポート部等価回路図」で記されている等価回路の範囲を示します。

表 8-5 機能一覧

Type	汎用ポート	機能 1	機能 2	アナログ	Pull-up	Pull-down	プログラマブルオープンドレイン	備考
T1	入出力	-	-	-	R	-	-	
T2	入出力	入力	-	-	NoR	-	-	
T3	入出力	入力	-	-	R	-	-	
T4	入出力	入力	-	-	R	-	o	
T5	入出力	入力	-	-	NoR	-	-	リセット中 $\overline{\text{BOOT}}$ 入力イネーブル
T6	入出力	入力	-	-	-	NoR	-	
T7	入出力	入力 (int)	-	-	R	-	-	
T8	入出力	入力 (int)	-	-	R	-	o	
T9	入出力	出力	-	-	R	-	-	
T10	入出力	出力	-	-	R	-	o	
T11	入出力	出力	-	-	R	-	-	機能出力にイネーブル付き
T12	入出力	入出力	-	-	NoR	-	-	機能出力にイネーブル付き
T13	入出力	入出力	-	-	R	-	o	
T14	入出力	入出力	-	-	-	-	-	N チャネルオープンドレイン端子
T15	入出力	出力	出力	-	R	-	-	
T16	入出力	入出力	入力	-	R	-	o	
T17	入力	-	-	o	R	-	-	
T18	入力	入力	-	o	R	-	-	

int : 割り込み入力

- : なし

o : 有り

R: リセット中は強制的に禁止

NoR: リセットでは制御されない。



8.3.2 タイプ T1

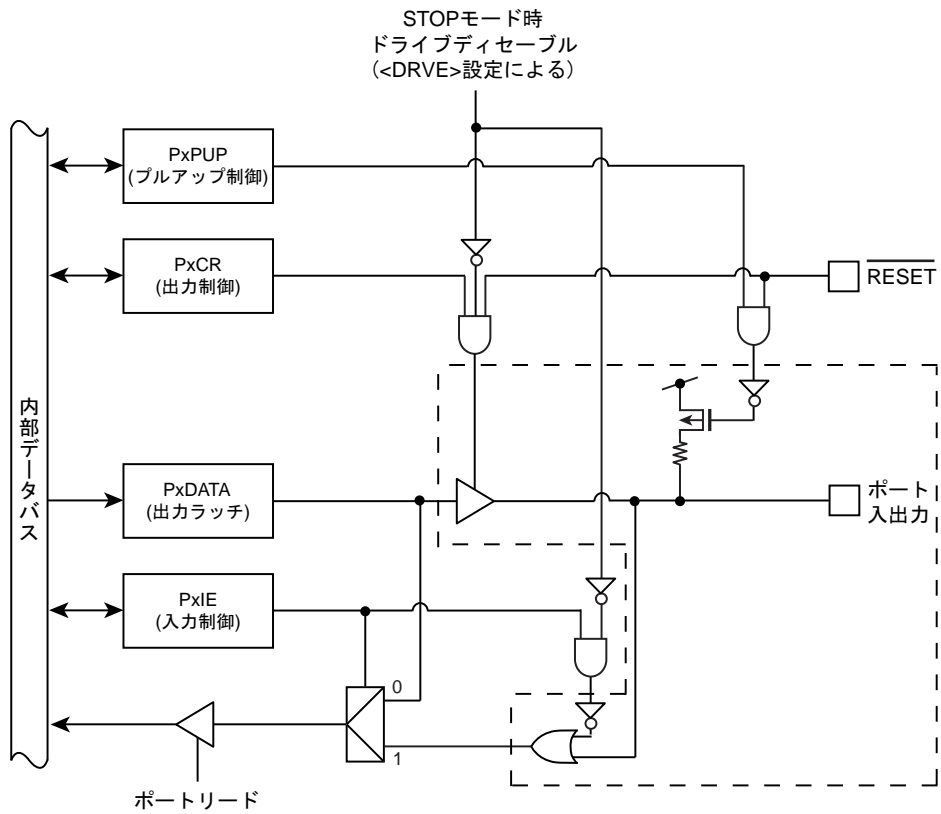


図 8-1 ポートタイプ T1

8.3.3 タイプ T2

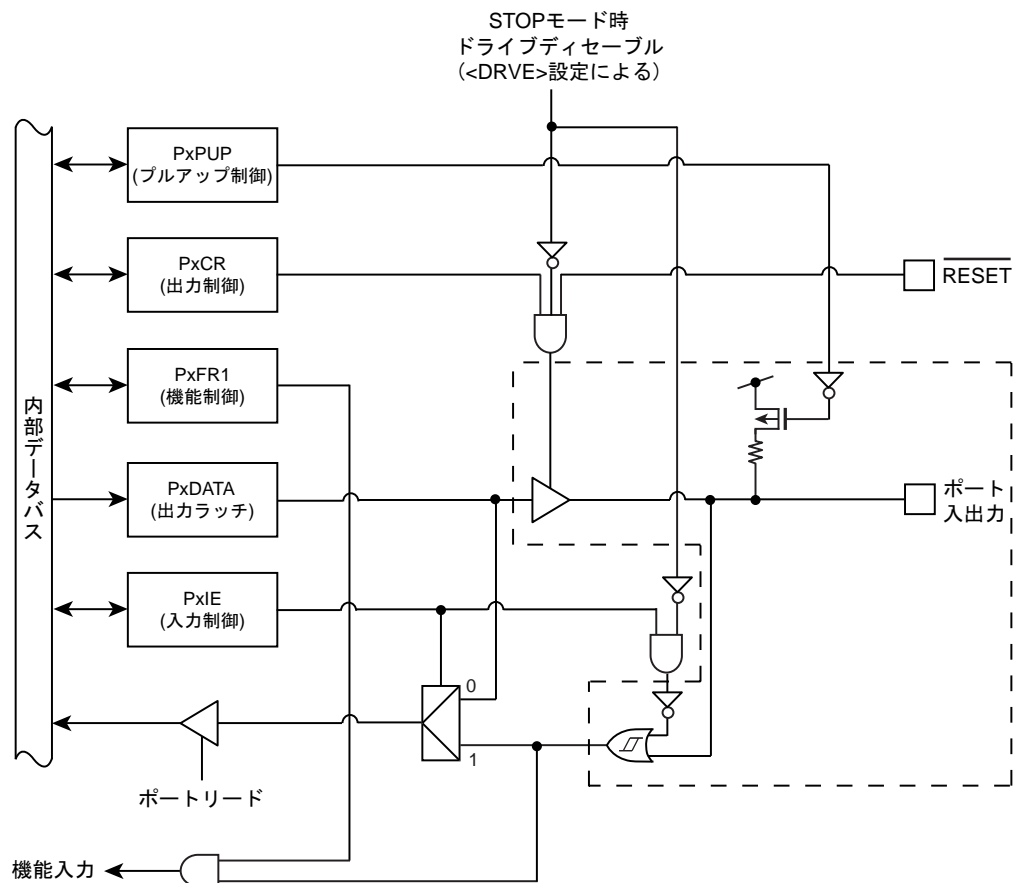


図 8-2 ポートタイプ T2

8.3.4 タイプ T3

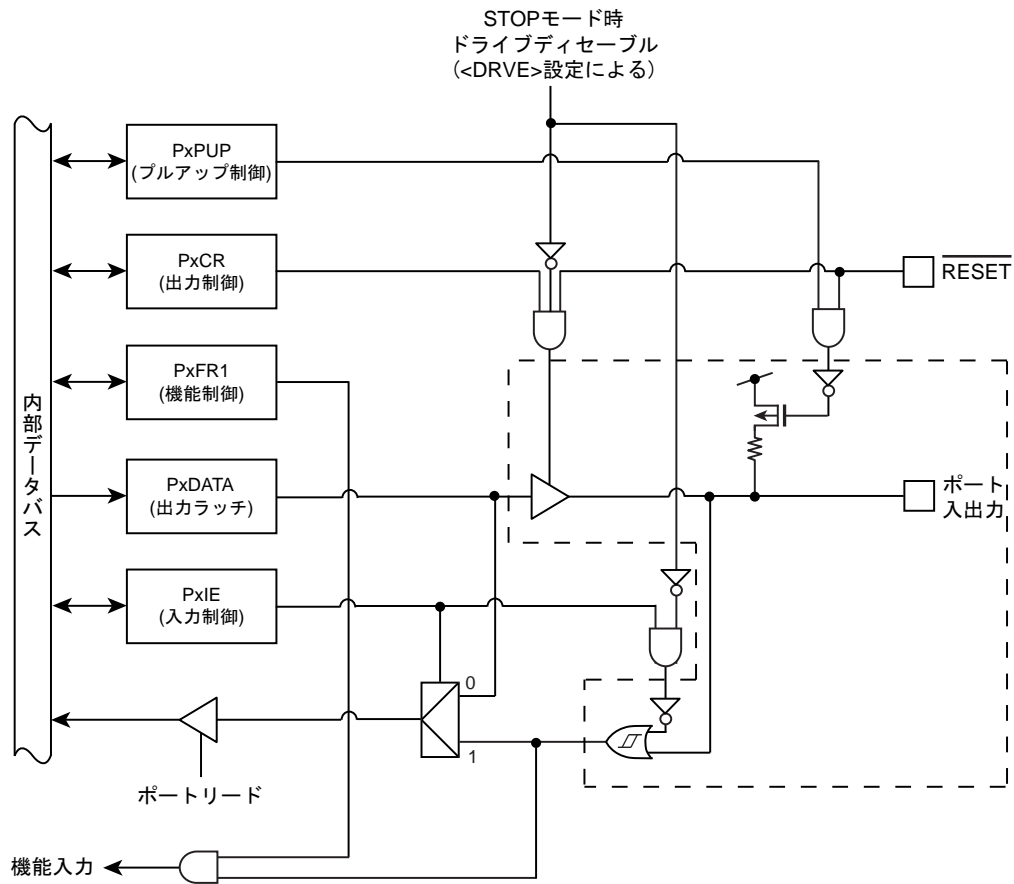


図 8-3 ポートタイプ T3

8.3.5 タイプ T4

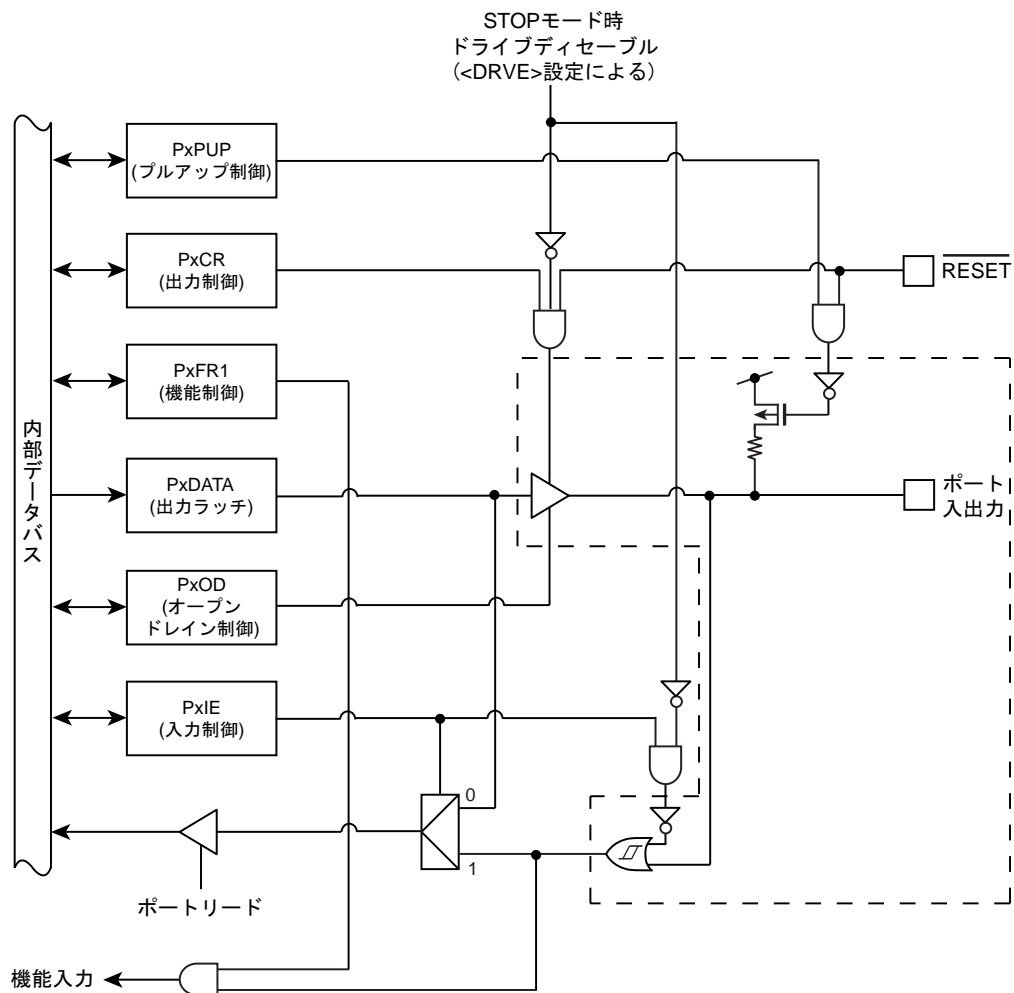


図 8-4 ポートタイプ T4

8.3.6 タイプ T5

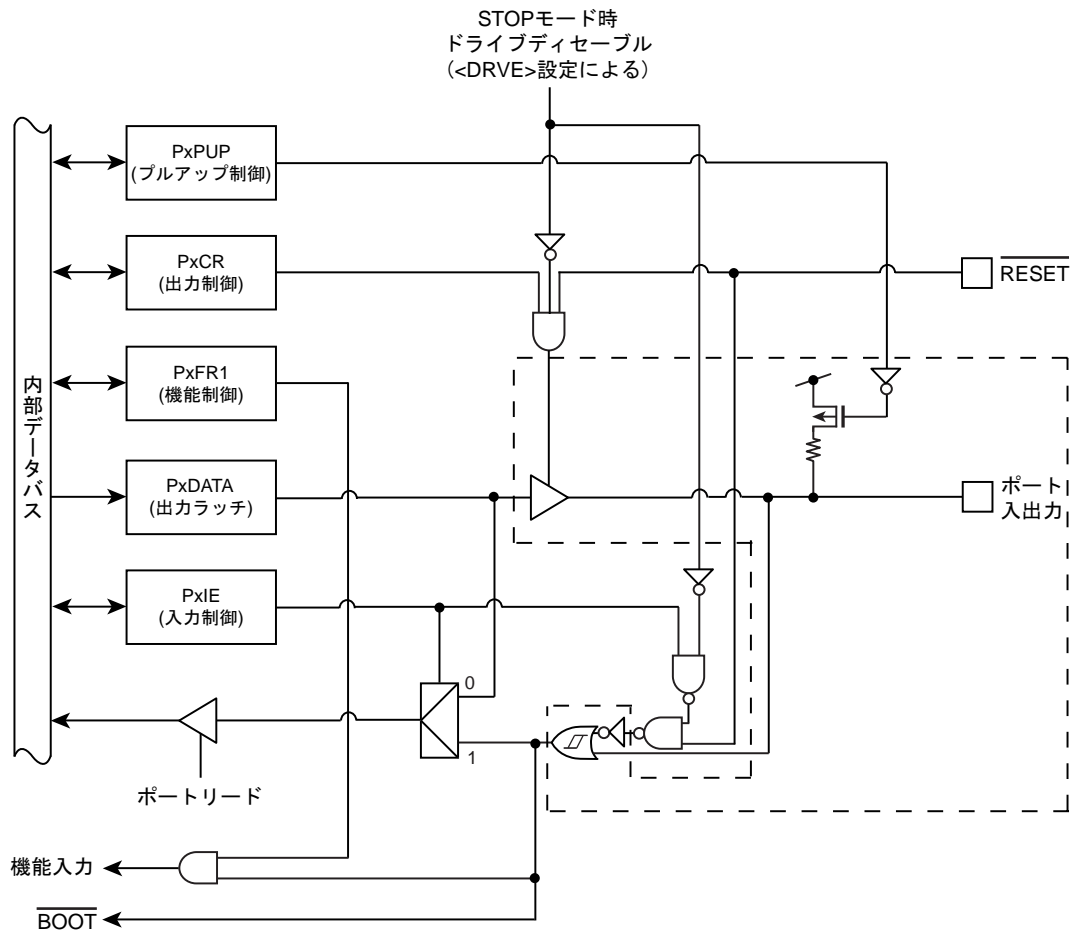


図 8-5 ポートタイプ T5

8.3.7 タイプ T6

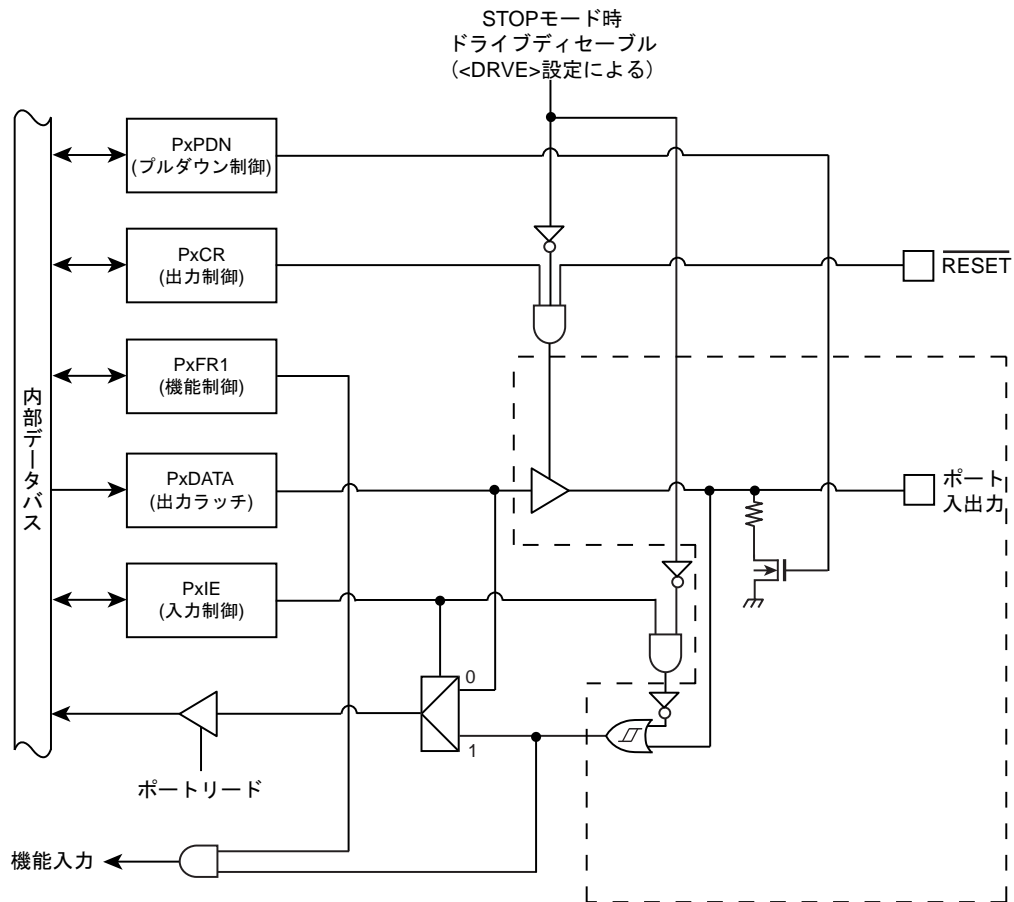


図 8-6 ポートタイプ T6

8.3.8 タイプ T7

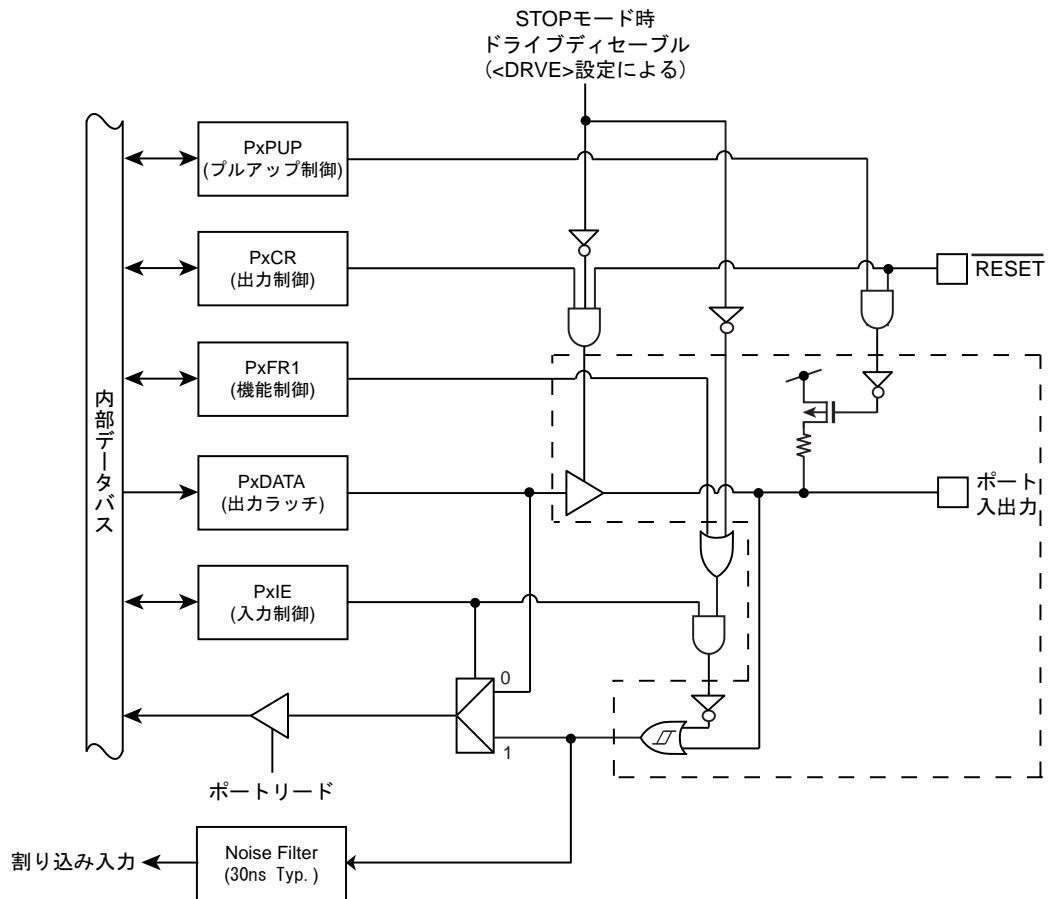


図 8-7 ポートタイプ T7

8.3.9 タイプ T8

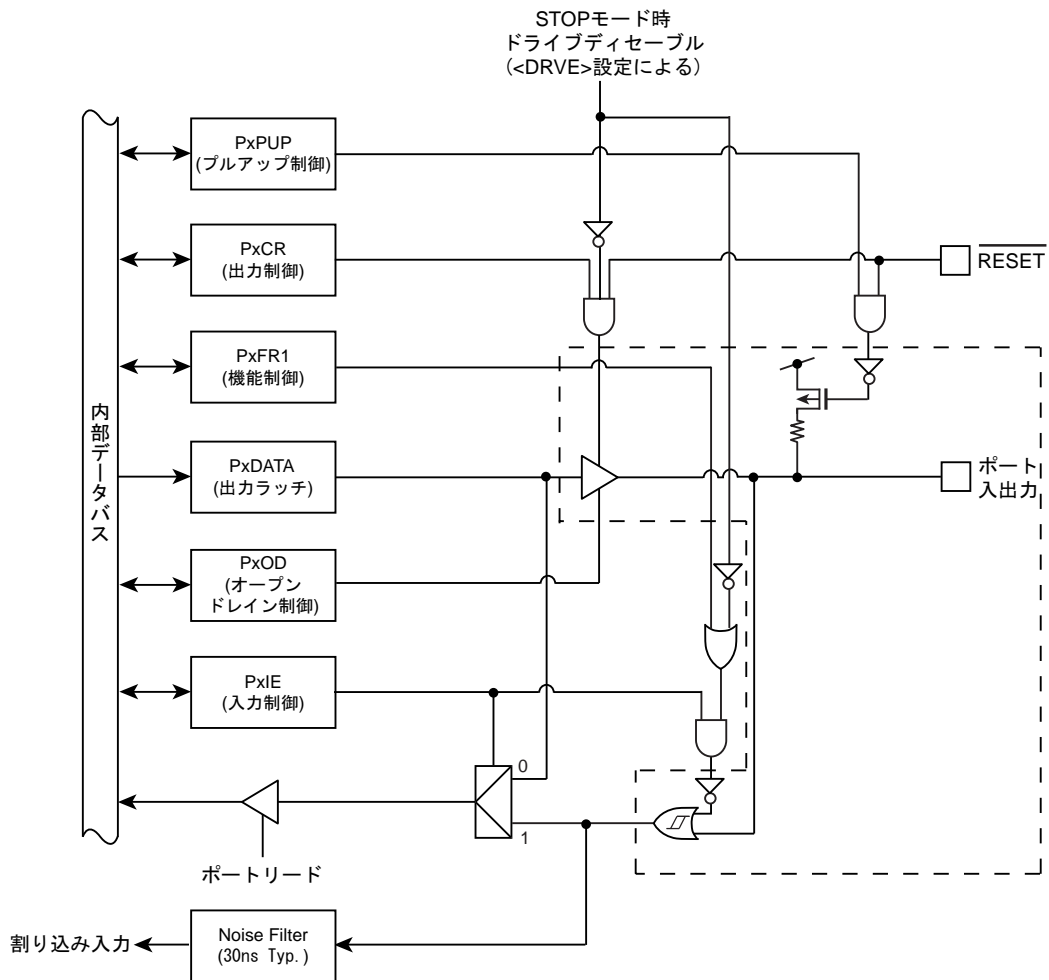


図 8-8 ポートタイプ T8



8.3.10 タイプ T9

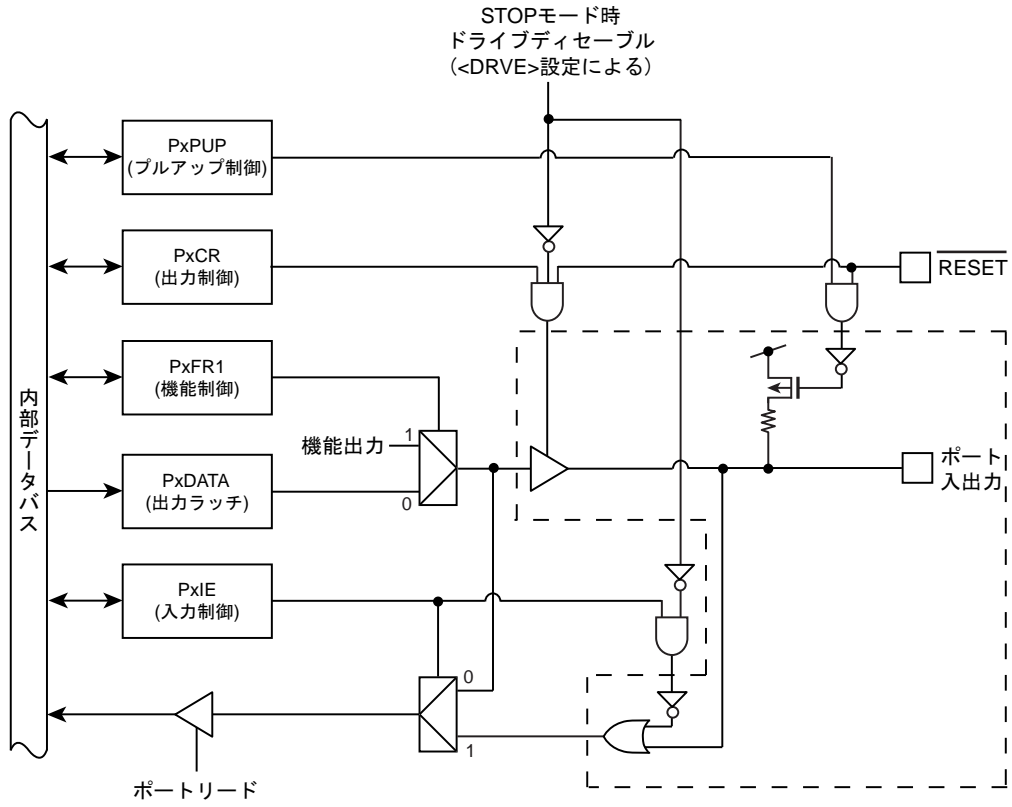


図 8-9 ポートタイプ T9

8.3.11 タイプ T10

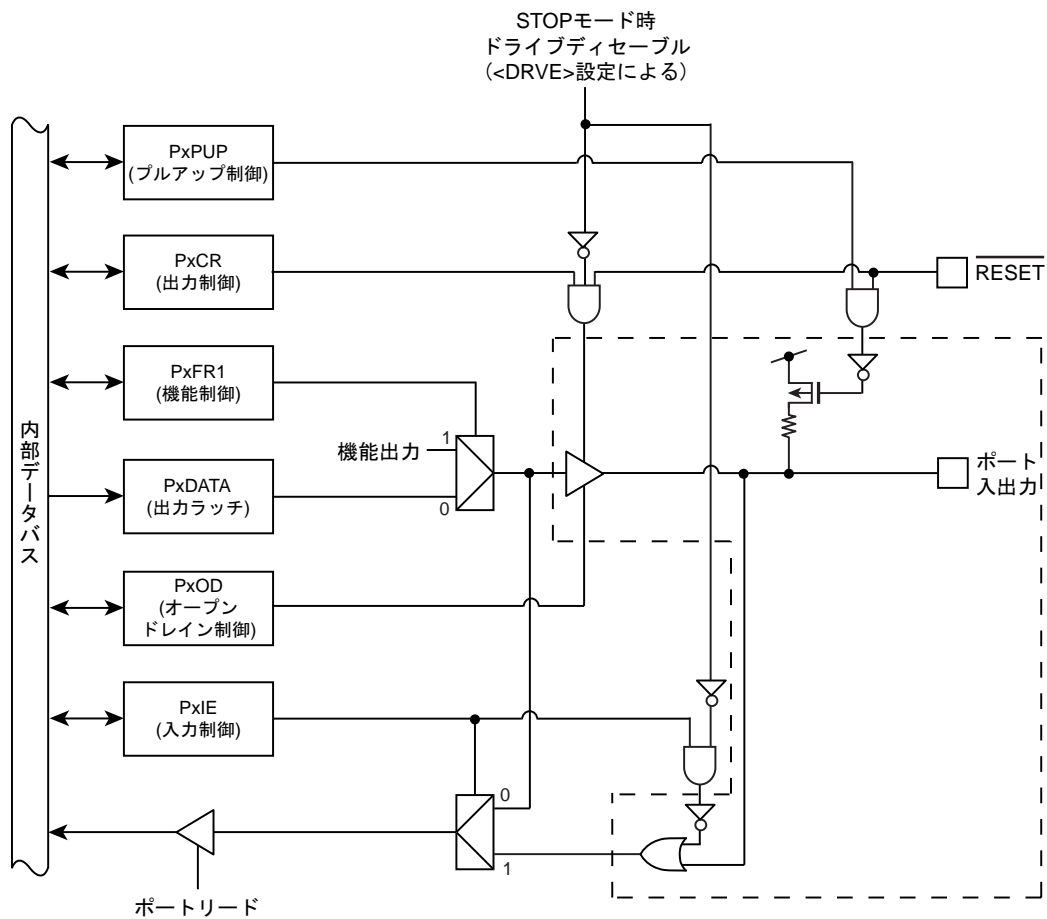


図 8-10 ポートタイプ T10

8.3.12 タイプ T11

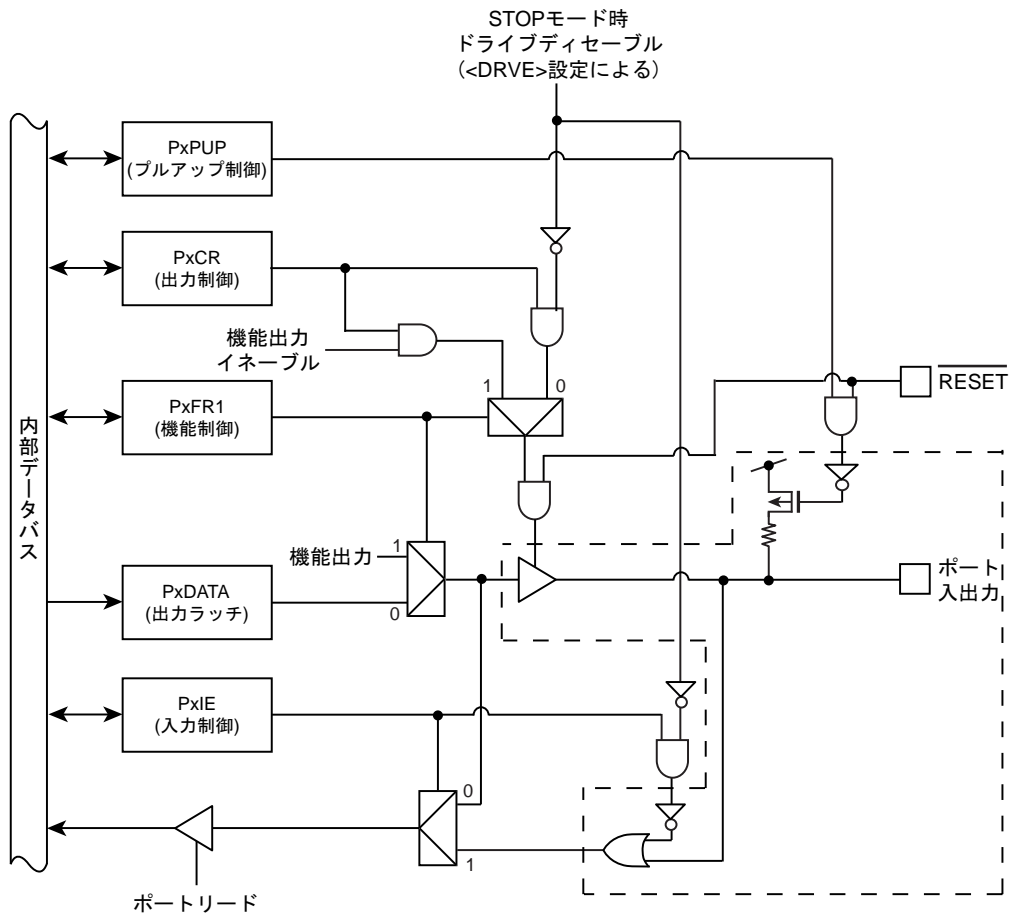


図 8-11 ポートタイプ T11

8.3.13 タイプT12

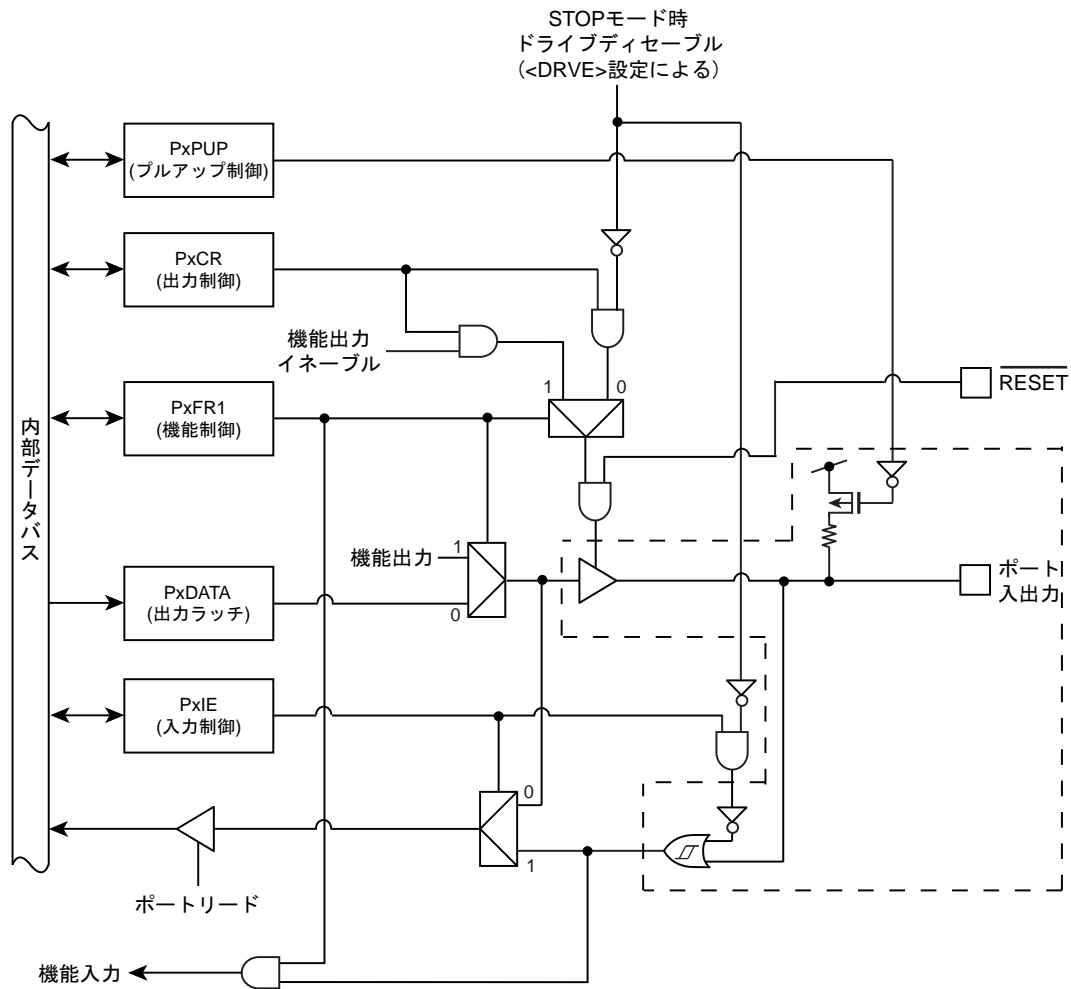


図 8-12 ポートタイプ T12

8.3.14 タイプ T13

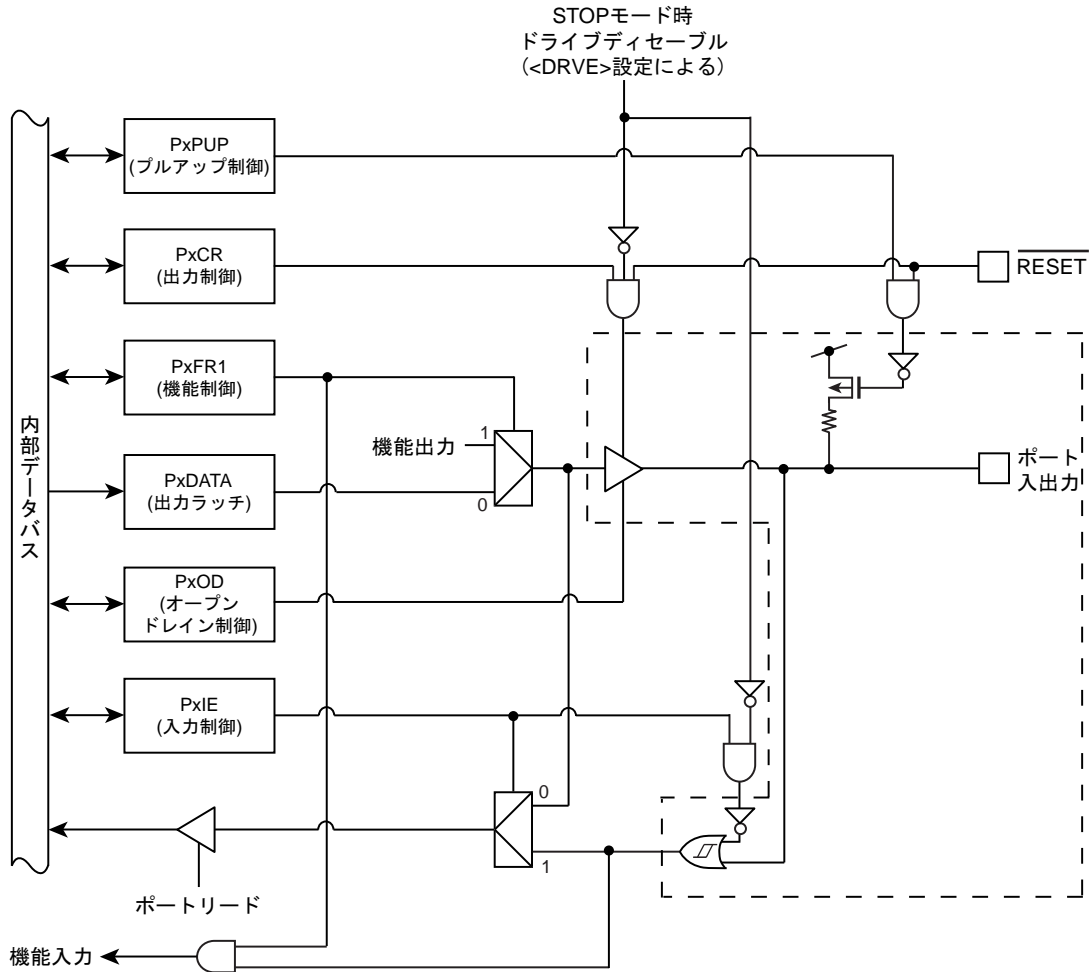


図 8-13 ポートタイプ T13

8.3.15 タイプ T14

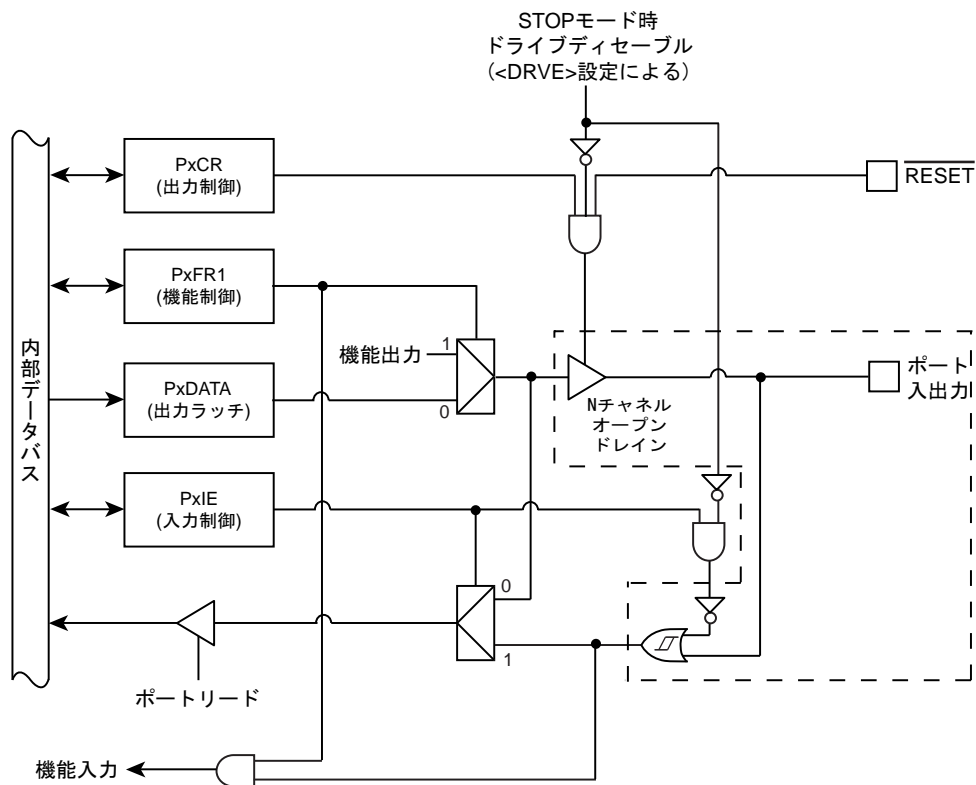


図 8-14 ポートタイプ T14

8.3.16 タイプ T15

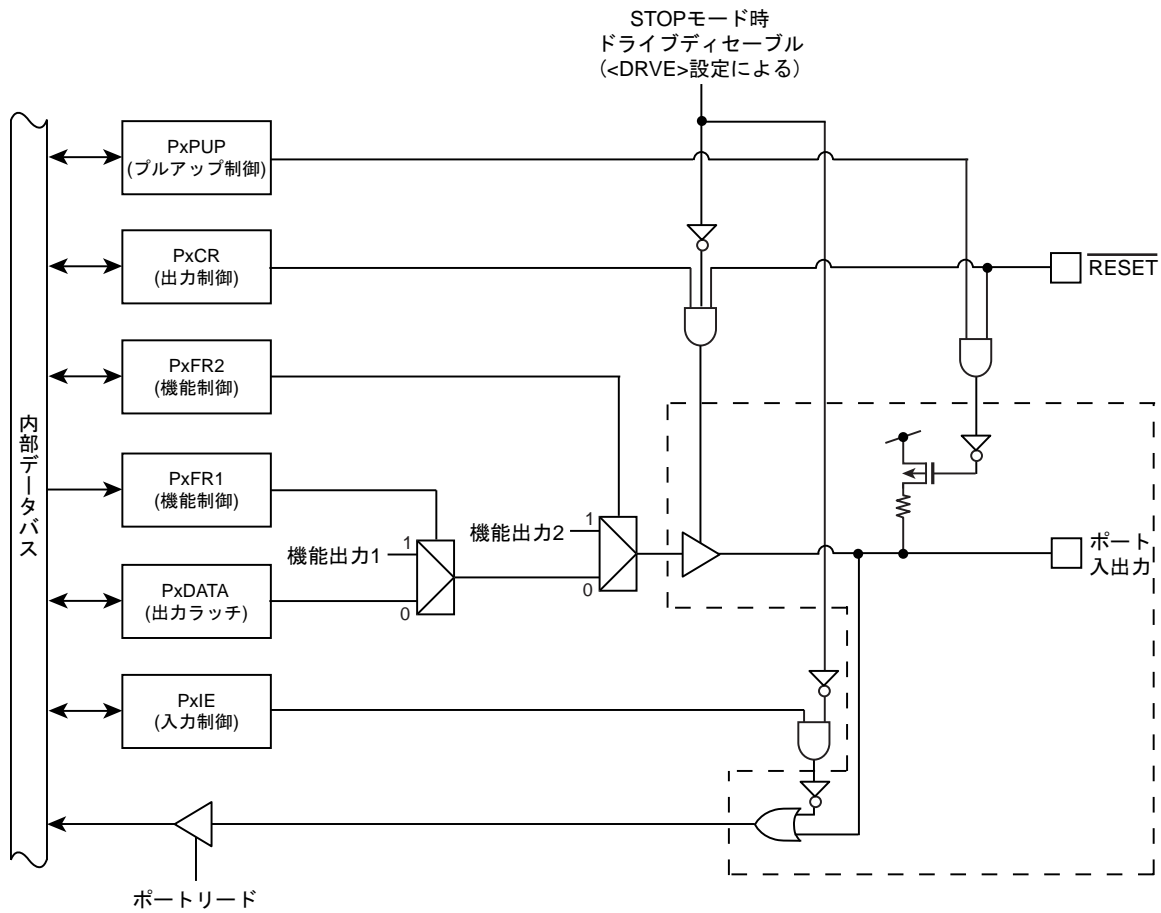


図 8-15 ポートタイプ T15

8.3.17 タイプ T16

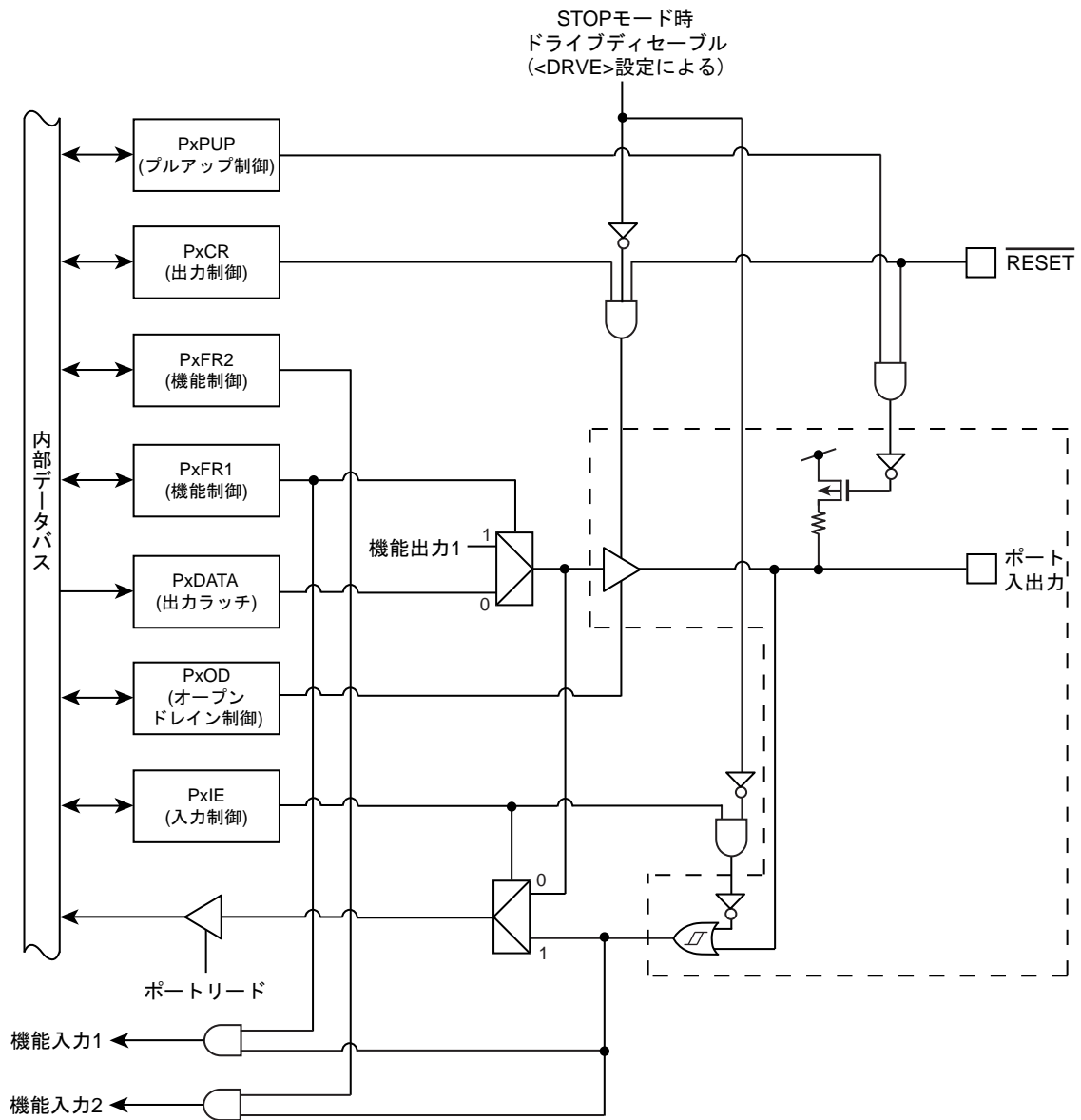


図 8-16 ポートタイプ T16



8.3.18 タイプ T17

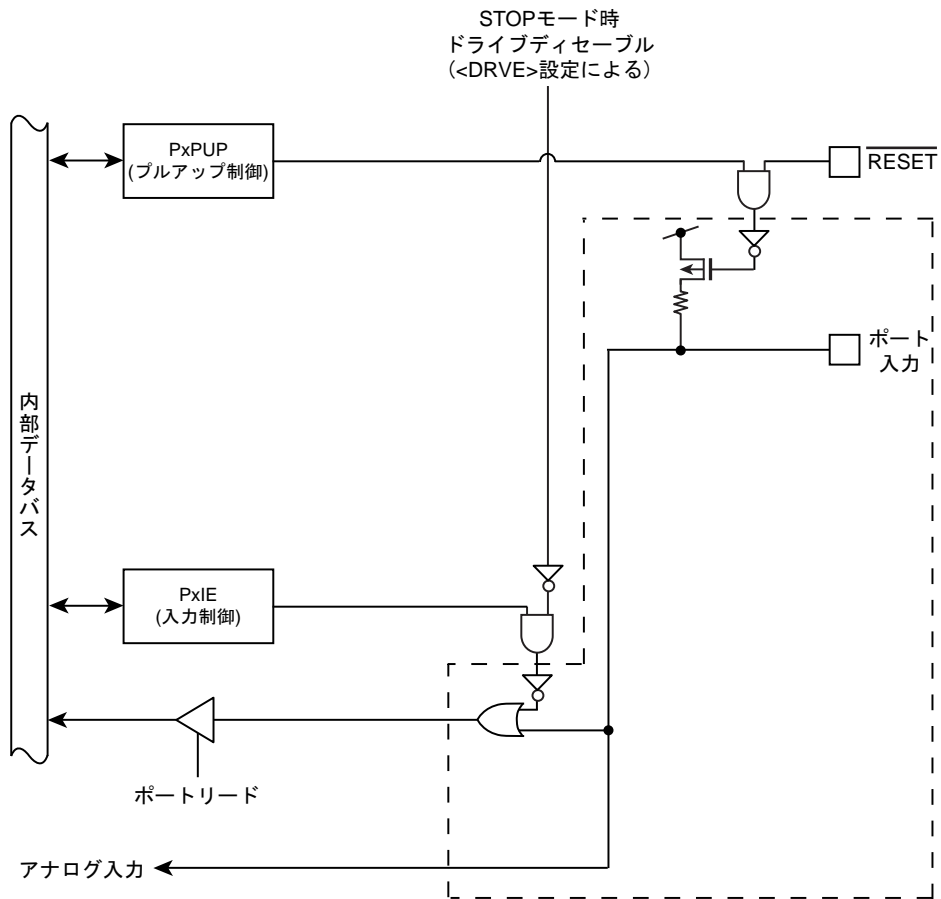


図 8-17 ポートタイプ T17

8.3.19 タイプ T18

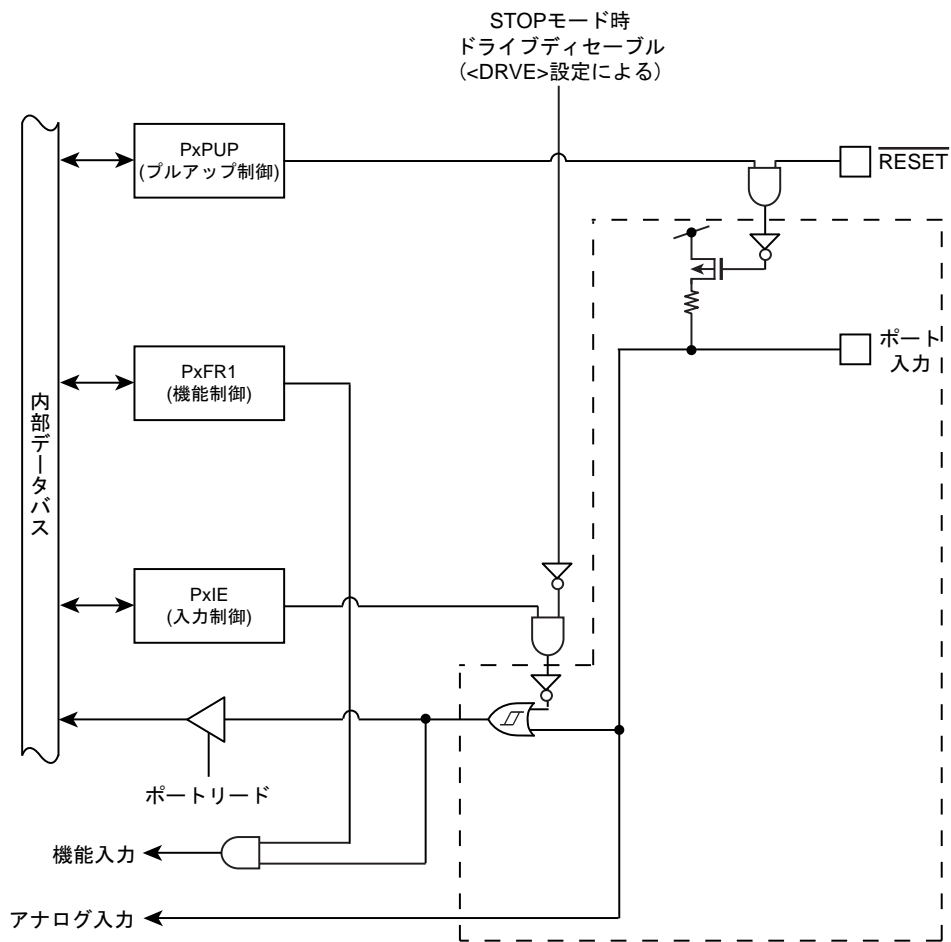


図 8-18 ポートタイプ T18

## 8.4 付録（ポート設定一覧）

機能ごとのレジスタ設定一覧を以下に示します。

初期設定欄に「・」のないポートの初期設定は、すべてのレジスタ設定が"0"となっています。

"x"のビット設定は任意に行ってください。

### 8.4.1 ポート A 設定

表 8-6 ポート設定一覧(ポート A)

端子名	ポート タイプ	機能	初期 設定	PACR	PAFR1	PAPUP	PAPDN	PAIE
PA0	T12	入力ポート		0	0	x	0	1
		出力ポート		1	0	x	0	0
		TMS(入力)/ SWDIO(入出力)	・	1	1	1	0	1
PA1	T6	入力ポート		0	0	0	x	1
		出力ポート		1	0	0	x	0
		TCK(入力)/ SWCLK(入力)	・	0	1	0	1	1
PA2	T9	入力ポート		0	0	x	0	1
		出力ポート		1	0	x	0	0
		TRACECLK(出力)		1	1	x	0	0
PA3	T9	入力ポート		0	0	x	0	1
		出力ポート		1	0	x	0	0
		TRACEDATA0(出力)		1	1	x	0	0
PA4	T9	入力ポート		0	0	x	0	1
		出力ポート		1	0	x	0	0
		TRACEDATA1(出力)		1	1	x	0	0
PA5	T9	入力ポート		0	0	x	0	1
		出力ポート		1	0	x	0	0
		TRACEDATA2(出力)		1	1	x	0	0
PA6	T9	入力ポート		0	0	x	0	1
		出力ポート		1	0	x	0	0
		TRACEDATA3(出力)		1	1	x	0	0
PA7	T1	入力ポート		0	0	x	0	1
		出力ポート		1	0	x	0	0

## 8.4.2 ポート B 設定

表 8-7 ポート設定一覧(ポート B)

端子名	ポート タイプ	機能	初期 設定	PBCR	PBFR1	PBPUP	PBIE
PB0	T11	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TDO(出力)/ SWV(出力)	.	1	1	0	0
PB1	T2	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TDI(入力)	.	0	1	1	1
PB2	T2	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		$\overline{\text{TRST}}$ (入力)	.	0	1	1	1
PB3	T1	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
PB4	T1	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
PB5	T1	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
PB6	T1	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
PB7	T1	入力ポート		0	0	x	1
		出力ポート		1	0	x	0

## 8.4.3 ポート C 設定

表 8-8 ポート設定一覧(ポート C)

端子名	ポート タイプ	機能	初期 設定	PCPUP	PCIE
PC0	T17	入力ポート		x	1
		アナログ入力	.	0	0
PC1	T17	入力ポート		x	1
		アナログ入力	.	0	0
PC2	T17	入力ポート		x	1
		アナログ入力	.	0	0
PC3	T17	入力ポート		x	1
		アナログ入力	.	0	0

## 8.4.4 ポート D 設定

表 8-9 ポート設定一覧(ポート D)

端子名	ポート タイプ	機能	初期 設定	PDFR1	PDPUP	PDIE
PD0	T18	入力ポート		0	x	1
		TB5IN0(入力)		1	x	1
		アナログ入力	.	x	0	0
PD1	T18	入力ポート		0	x	1
		TB5IN1(入力)		1	x	1
		アナログ入力	.	x	0	0
PD2	T18	入力ポート		0	x	1
		TB6IN0(入力)		1	x	1
		アナログ入力	.	x	0	0
PD3	T18	入力ポート		0	x	1
		TB6IN1(入力)		1	x	1
		アナログ入力	.	x	0	0
PD4	T17	入力ポート		0	x	1
		アナログ入力	.	x	0	0
PD5	T17	入力ポート		0	x	1
		アナログ入力	.	x	0	0
PD6	T17	入力ポート		0	x	1
		アナログ入力	.	x	0	0
PD7	T17	入力ポート		0	x	1
		アナログ入力	.	x	0	0

## 8.4.5 ポートE設定

表 8-10 ポート設定一覧(ポートE)

端子名	ポート タイプ	機能	初期 設定	PECR	PEFR1	PEFR2	PEOD	PEPUP	PEIE
PE0	T10	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		TXD0(出力)		1	1	0	x	x	0
PE1	T4	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		RXD0(入力)		0	1	0	x	x	1
PE2	T16	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		SCLK0(入力)		0	1	0	x	x	1
		SCLK0(出力)		1	1	0	x	x	0
		CTS0(入力)		0	0	1	x	x	1
PE3	T4	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		RXIN0(入力)		0	1	0	x	x	1
PE4	T10	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		TXD1(出力)		1	1	0	x	x	0
PE5	T4	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		RXD1(入力)		0	1	0	x	x	1
PE6	T16	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		SCLK1(入力)		0	1	0	x	x	1
		SCLK1(出力)		1	1	0	x	x	0
		CTS1(入力)		0	0	1	x	x	1

## 8.4.6 ポート F 設定

表 8-11 ポート設定一覧(ポート F)

端子名	ポート タイプ	機能	初期 設定	PFCR	PFFR1	PFFR2	PFOD	PFPUP	PFIE
PF0	T10	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		TXD2(出力)		1	1	0	x	x	0
PF1	T4	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		RXD2(入力)		0	1	0	x	x	1
PF2	T16	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		SCLK2(入力)		0	1	0	x	x	1
		SCLK2(出力)		1	1	0	x	x	0
		CTS2(入力)		0	0	1	x	x	1
PF3	T4	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		RXIN1(入力)		0	1	0	x	x	1
PF4	T13	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		SO1(出力)		1	1	0	x	x	0
		SDA1(入出力)		1	1	0	1	x	1
PF5	T13	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		SI1(入力)		0	1	0	x	x	1
		SCL1(入出力)		1	1	0	1	x	1
PF6	T13	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		SCK1(入力)		0	1	0	x	x	1
		SCK1(出力)		1	1	0	x	x	0
PF7	T8	入力ポート		0	0	0	x	x	1
		出力ポート		1	0	0	x	x	0
		INT5(入力)		0	1	0	x	x	1

## 8.4.7 ポート G 設定

表 8-12 ポート設定一覧(ポート G)

端子名	ポート タイプ	機能	初期 設定	PGCR	PGFR1	PGOD	PGPUP	PGIE
PG0	T13	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		SO0(出力)		1	1	x	x	0
		SDA0(入出力)		1	1	1	x	1
PG1	T13	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		SI0(入力)		0	1	x	x	1
		SCL0(入出力)		1	1	1	x	1
PG2	T13	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		SCK0(入力)		0	1	x	x	1
		SCK0(出力)		1	1	x	x	0
PG3	T8	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		INT4(入力)		0	1	x	x	1
PG4	T13	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		SO2(出力)		1	1	x	x	0
		SDA2(入出力)		1	1	1	x	1
PG5	T13	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		SI2(入力)		0	1	x	x	1
		SCL2(入出力)		1	1	1	x	1
PG6	T13	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		SCK2(入力)		0	1	x	x	1
		SCK2(出力)		1	1	x	x	0
PG7	T10	入力ポート		0	0	x	x	1
		出力ポート		1	0	x	x	0
		TB8OUT(出力)		1	1	x	x	0



## 8.4.8 ポート H 設定

表 8-13 ポート設定一覧(ポート H)

端子名	ポート タイプ	機能	初期 設定	PHCR	PHFR1	PHPUP	PHIE
PH0	T5	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB0IN0(入力)		0	1	x	1
PH1	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB0IN1(入力)		0	1	x	1
PH2	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB1IN0(入力)		0	1	x	1
PH3	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB1IN1(入力)		0	1	x	1
PH4	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB2IN0(入力)		0	1	x	1
PH5	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB2IN1(入力)		0	1	x	1
PH6	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB3IN0(入力)		0	1	x	1
PH7	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB3IN1(入力)		0	1	x	1

注) PH0 は  $\overline{\text{RESET}}$  が "Low" の間 Pull-up と入力が可能になっており、 $\overline{\text{BOOT}}$  入力端子として機能します。

## 8.4.9 ポートI設定

表 8-14 ポート設定一覧(ポートI)

端子名	ポート タイプ	機能	初期 設定	PICR	PIFR1	PIPUP	PIIE
PI0	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB0OUT(出力)		1	1	x	0
PI1	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB1OUT(出力)		1	1	x	0
PI2	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB2OUT(出力)		1	1	x	0
PI3	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB3OUT(出力)		1	1	x	0
PI4	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB4OUT(出力)		1	1	x	0
PI5	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB5OUT(出力)		1	1	x	0
PI6	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB4IN0(入力)		0	1	x	1
PI7	T3	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB4IN1(入力)		0	1	x	1

## 8.4.10 ポート J 設定

表 8-15 ポート設定一覧(ポート J)

端子名	ポート タイプ	機能	初期 設定	PJCR	PJFR1	PJPUP	PJIE
PJ0	T7	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		INT0(入力)		0	1	x	1
PJ1	T7	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		INT1(入力)		0	1	x	1
PJ2	T7	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		INT2(入力)		0	1	x	1
PJ3	T7	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		INT3(入力)		0	1	x	1
PJ4	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB6OUT(出力)		1	1	x	0
PJ5	T9	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		TB7OUT(出力)		1	1	x	0
PJ6	T7	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		INT6(入力)		0	1	x	1
PJ7	T7	入力ポート		0	0	x	1
		出力ポート		1	0	x	0
		INT7(入力)		0	1	x	1

## 8.4.11 ポート K 設定

表 8-16 ポート設定一覧(ポート K)

端子名	ポート タイプ	機能	初期 設定	PKCR	PKFR1	PKFR2	PKPUP	PKIE
PK0	T14	入力ポート		0	0	0	0	1
		出力ポート		1	0	0	0	0
		CEC(入出力)		1	1	0	0	1
PK1	T15	入力ポート		0	0	0	x	1
		出力ポート		1	0	0	x	0
		SCOUT(出力)		1	1	0	x	0
		ALARM(出力)		1	0	1	x	0
PK2	T9	入力ポート		0	0	0	x	1
		出力ポート		1	0	0	x	0
		TB9OUT(出力)		1	1	0	x	0

注) PK0 は N チャネルオープンドレインタイプの端子です。

## 第9章 16ビットタイマ/イベントカウンタ(TMRB)

### 9.1 概要

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

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

以下の説明中、"x"はチャンネル番号を表します。

## 9.2 チャンネル別仕様相違点

TMPM330DFDG/FYFG/FWFG は、10 チャンネルの TMRB を内蔵しています。

各チャンネルはそれぞれ独立に動作します。いずれのチャンネルも表 9-1 に示される仕様相違点を除いて同一の動作をします。

また、一部のチャンネルから他のチャンネルへキャプチャトリガや同期トリガをかけることができます。

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 → TMRB0~1 で使用
  - ・ TB8OUT → TMRB2~4 で使用
  - ・ TB9OUT → TMRB5~6 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 9-1 TMRB のチャンネル別仕様相違点

仕様 チャンネル	外部端子				タイマ間トリガ機能		割り込み	
	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	同期スタート トリガチャンネル	キャプチャ 割り込み	TMRB 割り込み
	信号名	ポート (ピン番号)	信号名	ポート (ピン番号)				
TMRB0	TB0IN0 TB0IN1	PH0 (30) PH1 (31)	TB0OUT	PI0 (38)	TB7OUT	-	INTCAP00 INTCAP01	INTTB0
TMRB1	TB1IN0 TB1IN1	PH2 (32) PH3 (36)	TB1OUT	PI1 (40)	TB7OUT	TMRB0	INTCAP10 INTCAP11	INTTB1
TMRB2	TB2IN0 TB2IN1	PH4 (9) PH5 (10)	TB2OUT	PI2 (42)	TB8OUT	TMRB0	INTCAP20 INTCAP21	INTTB2
TMRB3	TB3IN0 TB3IN1	PH6 (84) PH7 (85)	TB3OUT	PI3 (48)	TB8OUT	TMRB0	INTCAP30 INTCAP31	INTTB3
TMRB4	TB4IN0 TB4IN1	PI6 (79) PI7 (83)	TB4OUT	PI4 (52)	TB8OUT	-	INTCAP40 INTCAP41	INTTB4
TMRB5	TB5IN0 TB5IN1	PD0 (95) PD1 (96)	TB5OUT	PI5 (53)	TB9OUT	TMRB4	INTCAP50 INTCAP51	INTTB5
TMRB6	TB6IN0 TB6IN1	PD2 (97) PD3 (98)	TB6OUT	PJ4 (88)	TB9OUT	TMRB4	INTCAP60 INTCAP61	INTTB6
TMRB7	-	-	TB7OUT	PJ5 (8)	-	TMRB4	-	INTTB7
TMRB8	-	-	TB8OUT	PG7 (11)	-	-	-	INTTB8
TMRB9	-	-	TB9OUT	PK2 (7)	-	-	-	INTTB9

### 9.3 構成

各チャネルは、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。

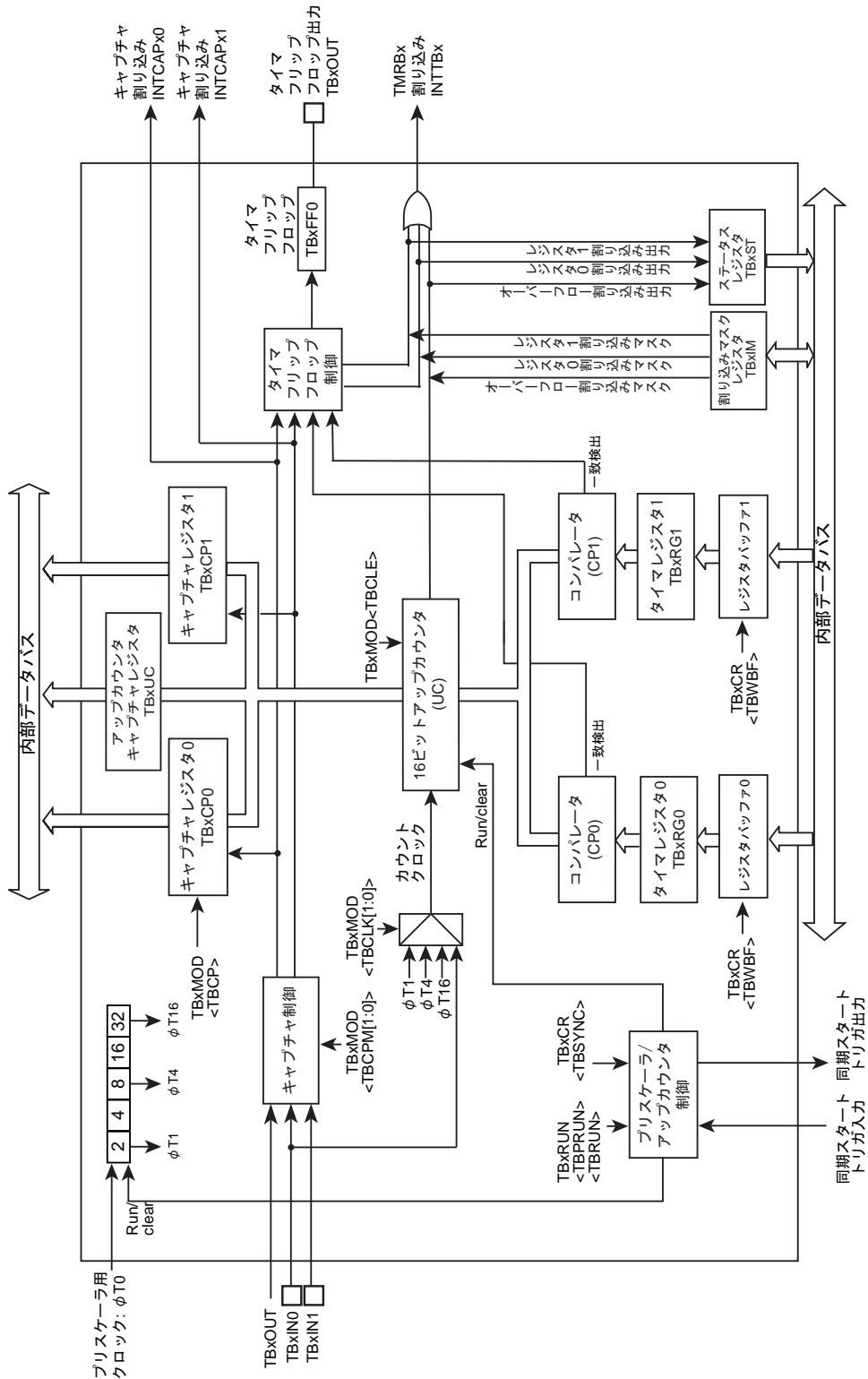


図 9-1 TMRBx ブロック図(x= 0~9)

## 9.4 レジスタ説明

### 9.4.1 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以下に示します。

Channel x	Base Address
Channel0	0x4001_0000
Channel1	0x4001_0040
Channel2	0x4001_0080
Channel3	0x4001_00C0
Channel4	0x4001_0100
Channel5	0x4001_0140
Channel6	0x4001_0180
Channel7	0x4001_01C0
Channel8	0x4001_0200
Channel9	0x4001_0240

レジスタ名(x=0~9)		Address(Base+)
イネーブルレジスタ	TBxEN	0x0000
RUN レジスタ	TBxRUN	0x0004
コントロールレジスタ	TBxCR	0x0008
モードレジスタ	TBxMOD	0x000C
フリップフロップコントロールレジスタ	TBxFFCR	0x0010
ステータスレジスタ	TBxST	0x0014
割り込みマスクレジスタ	TBxIM	0x0018
アップカウンタキャプチャレジスタ	TBxUC	0x001C
タイマレジスタ 0	TBxRG0	0x0020
タイマレジスタ 1	TBxRG1	0x0024
キャプチャレジスタ 0	TBxCP0	0x0028
キャプチャレジスタ 1	TBxCP1	0x002C



## 9.4.2 TBxEN(イネーブルレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBEN	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	TBEN	R/W	<p>TMRBx 動作</p> <p>0: 禁止</p> <p>1: 許可</p> <p>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(この状態では、TBxEN レジスタ以外のレジスタへのリード、ライトはできません)。</p> <p>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 をいったん動作させた後に、動作禁止した場合は各レジスタの設定は保持されます。</p>
6-0	-	R	リードすると"0"が読めます。

## 9.4.3 TBxRUN(RUN レジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	TBPRUN	-	TBRUN
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2	TBPRUN	R/W	プリスケアラ動作 0: 停止&クリア 1: カウント
1	-	R	リードすると"0"が読めます。
0	TBRUN	R/W	カウンタ動作 0: 停止&クリア 1: カウント

9.4.4 TBxCR(コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBWBF	-	TBSYNC	-	I2TB	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	TBWBF	R/W	ダブルバッファ 0: 禁止 1: 許可
6	-	R/W	"0"をライトしてください。
5	TBSYNC	R/W	同期モード切替 0: 個別動作(チャンネルごと) 1: 同期動作
4	-	R	リードすると"0"が読めます。
3	I2TB	R/W	IDLE 時の動作 0: 停止 1: 動作
2-0	-	R	リードすると"0"が読めます。

### 9.4.5 TBxMOD(モードレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	TBCP	TBCPM		TBCLE	TBCLK	
リセット後	0	0	1	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6	-	R/W	"0"をライトしてください。
5	TBCP	W	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 "0"を書き込むとキャプチャレジスタ 0 (TBxCP0)にカウント値を取り込みます。 リードすると"1"が読めます。
4-3	TBCPM[1:0]	R/W	キャプチャタイミング 00: ディセーブル 01: TBxIN0↑ TBxIN1↑ TBxIN0 端子入力の立ち上がりでキャプチャレジスタ 0 (TBxCP0)にカウント値を取り込み、 TBxIN1 端子入力の立ち上がりでキャプチャレジスタ 1 (TBxCP1)にカウント値を取り込む 10: TBxIN0↑ TBxIN0↓ TBxIN0 端子入力の立ち上がりでキャプチャレジスタ 0 (TBxCP0)にカウント値を取り込み、 TBxIN0 端子入力の立ち下がりでキャプチャレジスタ 1 (TBxCP1)にカウント値を取り込む 11: TBxOUT↑ TBxOUT↓ 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 (TBxCP0)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 (TBxCP1)にカウント値を取り込む (TMRB0~1: TB7OUT、TMRB2~4: TB8OUT、TMRB5~6: TB9OUT)
2	TBCLE	R/W	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 アップカウンタのクリア制御を行います。 "0"でクリア禁止、"1"でタイマレジスタ 1 (TBxRG1)との一致時にクリアします。
1-0	TBCLK[1:0]	R/W	TMRBx のソースクロック選択 00: TBxIN0 端子入力 01: φT1 10: φT4 11: φT16

9.4.6 TBxFFCR(フリップフロップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	TBC1T1	TBC0T1	TBE1T1	TBE0T1	TBFF0C	
リセット後	1	1	0	0	0	0	1	1

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-6	-	R	リードすると"1"が読めます。
5	TBC1T1	R/W	TBxCP1 へのアップカウンタ値取り込み時の TBxFF0 反転トリガ 0: トリガディセーブル 1: トリガイネーブル "1"をセットすると、アップカウンタの値がキャプチャレジスタ 1 (TBxCP1)に取り込まれた時にタイムフリップフロップを反転します。
4	TBC0T1	R/W	TBxCP0 へのアップカウンタ値取り込み時の TBxFF0 反転トリガ 0: トリガディセーブル 1: トリガイネーブル "1"をセットすると、アップカウンタの値がキャプチャレジスタ 0 (TBxCP0)に取り込まれた時にタイムフリップフロップを反転します。
3	TBE1T1	R/W	アップカウンタと TBxRG1 との一致時の TBxFF0 反転トリガ 0: トリガディセーブル 1: トリガイネーブル "1"をセットすると、アップカウンタとタイマレジスタ 1 (TBxRG1)との一致時にタイムフリップフロップを反転します。
2	TBE0T1	R/W	アップカウンタと TBxRG0 との一致時の TBxFF0 反転トリガ 0: トリガディセーブル 1: トリガイネーブル "1"をセットすると、アップカウンタとタイマレジスタ 0 (TBxRG0)との一致時にタイムフリップフロップを反転します。
1-0	TBFF0C[1:0]	R/W	TBxFF0 の制御 00: Invert TBxFF0 の値を反転(ソフト反転)します。 01: Set TBxFF0 を"1"にセットします。 10: Clear TBxFF0 を"0"にクリアします。 11: Don't care ※リードすると"11" が読めます。

## 9.4.7 TBxST(ステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	INTTBOF	INTTB1	INTTB0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2	INTTBOF	R	オーバフローフラグ 0: オーバフローは発生していない 1: オーバフローが発生 アップカウンタのオーバフローが発生すると"1"がセットされます。
1	INTTB1	R	一致フラグ(TBxRG1) 0: 一致検出していない 1: TBxRG1 との一致を検出した タイマレジスタ 1 (TBxRG1)との一致を検出すると"1"がセットされます。
0	INTTB0	R	一致フラグ(TBxRG0) 0: 一致検出していない 1: TBxRG0 との一致を検出した タイマレジスタ 0 (TBxRG0)との一致を検出すると"1"がセットされます。

注 1) TBxIM でマスク設定されていない要因のみ、CPU に対し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) フラグは自動的にクリアされません。クリアするためには本レジスタをリードしてしてください。

## 9.4.8 TBxIM(割り込みマスクレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	TBIMOF	TBIM1	TBIM0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2	TBIMOF	R/W	オーバフロー割り込みマスク 0:割り込みをマスクしない 1:割り込みをマスクする アップカウンタのオーバフロー割り込みをマスクする/しないを設定します。
1	TBIM1	R/W	一致割り込みマスク(TBxRG1) 0:割り込みをマスクしない 1:割り込みをマスクする タイマレジスタ 1 (TBxRG1)との一致割り込みをマスクする/しないを設定します。
0	TBIM0	R/W	一致割り込みマスク(TBxRG0) 0:割り込みをマスクしない 1:割り込みをマスクする タイマレジスタ 0 (TBxRG0)との一致割り込みをマスクする/しないを設定します。

## 9.4.9 TBxUC(アップカウンタキャプチャレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBUC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBUC							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-0	TBUC[15:0]	R	アップカウンタ読み出しによりキャプチャした値 TBxUCをリードすると、現在のアップカウンタの値をキャプチャすることができます。

## 9.4.10 TBxRG0(タイマレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBRG0							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBRG0							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-0	TBRG0[15:0]	R/W	アップカウンタと比較する値を設定します。

## 9.4.11 TBxRG1(タイマレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBRG1							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBRG1							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-0	TBRG1[15:0]	R/W	アップカウンタと比較する値を設定します。



9.4.12 TBxCP0(キャプチャレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBCP0							
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	7	6	5	4	3	2	1	0
bit symbol	TBCP0							
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-0	TBCP0[15:0]	R	アップカウンタをキャプチャした値が読めます。

9.4.13 TBxCP1(キャプチャレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBCP1							
リセット後	不定	不定	不定	不定	不定	不定	不定	不定
	7	6	5	4	3	2	1	0
bit symbol	TBCP1							
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-0	TBCP1[15:0]	R	アップカウンタをキャプチャした値が読めます。

## 9.5 回路別の動作説明

各チャンネルは表 9-1 に示される仕様相違点を除いて同一の動作をします。

### 9.5.1 プリスケーラ

アップカウンタ UC 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 TBxRUN<TBPRUN> 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 9-2、表 9-3 に示します。

表 9-2 プリスケーラ出力クロック分解能( $f_c = 40\text{MHz}$ )

ペリフェラル クロック選択 CGSYSCR <FPSEL>	クロックギア値 CGSYSCR <GEAR[2:0]>	プリスケーラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケーラ出力クロック機能		
			$\phi T1$	$\phi T4$	$\phi T16$
0 ( $f_{gear}$ )	000 ( $f_c$ )	000 ( $f_{periph/1}$ )	$f_c/2^1$ (0.05 $\mu\text{s}$ )	$f_c/2^3$ (0.2 $\mu\text{s}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )
		001 ( $f_{periph/2}$ )	$f_c/2^2$ (0.1 $\mu\text{s}$ )	$f_c/2^4$ (0.4 $\mu\text{s}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )
		010 ( $f_{periph/4}$ )	$f_c/2^3$ (0.2 $\mu\text{s}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )
		011 ( $f_{periph/8}$ )	$f_c/2^4$ (0.4 $\mu\text{s}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )
		100 ( $f_{periph/16}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )
		101 ( $f_{periph/32}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )	$f_c/2^{10}$ (25.6 $\mu\text{s}$ )
	100 ( $f_c/2$ )	000 ( $f_{periph/1}$ )	$f_c/2^2$ (0.1 $\mu\text{s}$ )	$f_c/2^4$ (0.4 $\mu\text{s}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )
		001 ( $f_{periph/2}$ )	$f_c/2^3$ (0.2 $\mu\text{s}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )
		010 ( $f_{periph/4}$ )	$f_c/2^4$ (0.4 $\mu\text{s}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )
		011 ( $f_{periph/8}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )
		100 ( $f_{periph/16}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )	$f_c/2^{10}$ (25.6 $\mu\text{s}$ )
		101 ( $f_{periph/32}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )	$f_c/2^{11}$ (51.2 $\mu\text{s}$ )
	101 ( $f_c/4$ )	000 ( $f_{periph/1}$ )	$f_c/2^3$ (0.2 $\mu\text{s}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )
		001 ( $f_{periph/2}$ )	$f_c/2^4$ (0.4 $\mu\text{s}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )
		010 ( $f_{periph/4}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )
		011 ( $f_{periph/8}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )	$f_c/2^{10}$ (25.6 $\mu\text{s}$ )
		100 ( $f_{periph/16}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )	$f_c/2^{11}$ (51.2 $\mu\text{s}$ )
		101 ( $f_{periph/32}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )	$f_c/2^{10}$ (25.6 $\mu\text{s}$ )	$f_c/2^{12}$ (102.4 $\mu\text{s}$ )
	110 ( $f_c/8$ )	000 ( $f_{periph/1}$ )	$f_c/2^4$ (0.4 $\mu\text{s}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )
		001 ( $f_{periph/2}$ )	$f_c/2^5$ (0.8 $\mu\text{s}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )
		010 ( $f_{periph/4}$ )	$f_c/2^6$ (1.6 $\mu\text{s}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )	$f_c/2^{10}$ (25.6 $\mu\text{s}$ )
		011 ( $f_{periph/8}$ )	$f_c/2^7$ (3.2 $\mu\text{s}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )	$f_c/2^{11}$ (51.2 $\mu\text{s}$ )
		100 ( $f_{periph/16}$ )	$f_c/2^8$ (6.4 $\mu\text{s}$ )	$f_c/2^{10}$ (25.6 $\mu\text{s}$ )	$f_c/2^{12}$ (102.4 $\mu\text{s}$ )
		101 ( $f_{periph/32}$ )	$f_c/2^9$ (12.8 $\mu\text{s}$ )	$f_c/2^{11}$ (51.2 $\mu\text{s}$ )	$f_c/2^{13}$ (204.8 $\mu\text{s}$ )

表 9-2 プリスケーラ出カクロック分解能( $f_c = 40\text{MHz}$ )

ペリフェラル クロック選択 CGSYSCR <FPSEL>	クロックギア値 CGSYSCR <GEAR[2:0]>	プリスケーラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケーラ出カクロック機能		
			$\phi T1$	$\phi T4$	$\phi T16$
1 (fc)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.05 $\mu\text{s}$ )	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.1 $\mu\text{s}$ )	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )
	100 (fc/2)	000 (fperiph/1)	-	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.1 $\mu\text{s}$ )	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )
	101 (fc/4)	000 (fperiph/1)	-	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )
	110 (fc/8)	000 (fperiph/1)	-	-	$fc/2^5$ (0.8 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )
		010 (fperiph/4)	-	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )

- 注 1) プリスケーラ出カクロック  $\phi T_n$  は、必ず  $\phi T_n < f_{\text{sys}}$  を満足するように( $\phi T_n$  が  $f_{\text{sys}}$  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 9-3 プリスケアラ出カクロック分解能( $f_c = 32\text{MHz}$ )

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

表 9-3 プリスケーラ出カクロック分解能( $f_c = 32\text{MHz}$ )

ペリフェラル クロック選択 CGSYSCR <FPSEL>	クロックギア値 CGSYSCR <GEAR[2:0]>	プリスケーラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケーラ出カクロック機能		
			$\phi T1$	$\phi T4$	$\phi T16$
1 (fc)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.0625 $\mu\text{s}$ )	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.125 $\mu\text{s}$ )	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
	100 (fc/2)	000 (fperiph/1)	-	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.125 $\mu\text{s}$ )	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
	101 (fc/4)	000 (fperiph/1)	-	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
	110 (fc/8)	000 (fperiph/1)	-	-	$fc/2^5$ (1.0 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )
		010 (fperiph/4)	-	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )

注 1) プリスケーラ出カクロック  $\phi T_n$  は、必ず  $\phi T_n < f_{\text{sys}}$  を満足するように( $\phi T_n$  が  $f_{\text{sys}}$  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。

## 9.5.2 アップカウンタ(UC)

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 TBxMOD<TBCLK[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、TBxIN0 入力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 TBxRUN<TBRUN>で行います。<TBRUN>="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. コンペア一致時  
TBxMOD<TBCLE>="1"に設定することで、TBxRG1 とのコンペア一致とともにカウンタのクリアをすることができます。TBxMOD<TBCLE>="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  
TBxRUN<TBRUN>="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。
- ・ カウンタのオーバフロー  
アップカウンタ UC がオーバフローすると、オーバフロー割り込み INTTBx が発生します。

## 9.5.3 タイマレジスタ(TBxRG0, TBxRG1)

アップカウンタ UC 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。

TBxRG0/1 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。初期状態では、ダブルバッファはディセーブルです。

ダブルバッファのイネーブル/ディセーブル制御は TBxCR<TBWBF>によって行います。<TBWBF>="0" のときディセーブル、<TBWBF>="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、UC と TBxRG1 との一致時にレジスタバッファ 0/1 からタイマレジスタ TBxRG0/1 ヘデータ転送が行われます。また、ダブルバッファがイネーブルでもカウンタが停止しているときはシングルバッファ動作となり、TBxRG0/1 に直接値を書き込むことができます。

#### 9.5.4 キャプチャ制御

アップカウンタ UC の値をキャプチャレジスタ TBxCP0, TBxCP1 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、TBxMOD<TBxCPM[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 UC の値をキャプチャレジスタへ取り込むことができ、TBxMOD<TBxCP>に "0" を書き込むたびに、その時点の UC の値をキャプチャレジスタ TBxCP0 へキャプチャします。

#### 9.5.5 キャプチャレジスタ(TBxCP0, TBxCP1)

アップカウンタ UC の値をキャプチャするレジスタです。

#### 9.5.6 アップカウンタキャプチャレジスタ(TBxUC)

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

#### 9.5.7 コンパレータ(CP0, CP1)

アップカウンタ UC と、タイマレジスタ TBxRG0, TBxRG1 への設定値とを比較し、一致を検出します。一致すると、INTTBx を発生します。

#### 9.5.8 タイマフリップフロップ(TBxFF0)

タイマフリップフロップ (TBxFF0) 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、TBxFFCR<TBC1T1, TBC0T1, TBE1T1, TBE0T1>によって設定できます。

リセット後、TBxFF0 の値は不定となります。TBxFFCR<TBFF0C[1:0]>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。

TBxFF0 の値は、タイマ出力端子 TBxOUT 端子へ出力することができます。タイマ出力を行う場合、あらかじめ該当するポートの設定を行う必要があります。

#### 9.5.9 キャプチャ割り込み(INTCAPx0, INTCAPx1)

キャプチャレジスタ TBxCP0, TBxCP1 にラッチするタイミングで割り込み INTCAPx0, INTCAPx1 をそれぞれ発生します。割り込みの設定は CPU で行います。

## 9.6 モード別動作説明

### 9.6.1 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 TBxRG1 にインタバル時間を設定することで INTTBx 割り込みを発生します。

	7	6	5	4	3	2	1	0	
TBxEN	← 1	X	X	X	X	X	X	X	TMRBx モジュールを起動します。
TBxRUN	← X	X	X	X	X	0	X	0	TMRBx を停止します
割り込みイネーブルセットレジスタ	← *	*	*	*	*	*	*	*	INTTBx 割り込みに対応するビットを"1"にし、割り込みを許可します。
TBxFFCR	← X	X	0	0	0	0	1	1	TB x FF0 反転トリガをディセーブルします。
TBxMOD	← X	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)			
TBxRG1	← *	*	*	*	*	*	*	*	インタバル時間を設定します。(16 ビット)
	← *	*	*	*	*	*	*	*	
TBxRUN	← *	*	*	*	*	1	X	1	TMRBx を起動します。

注) X; Don't care -; No change

### 9.6.2 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(TBxIN0 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 TBxIN0 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。

	7	6	5	4	3	2	1	0	
TBxEN	← 1	X	X	X	X	X	X	X	TMRBx モジュールを起動します。
TBxRUN	← X	X	X	X	X	0	X	0	TMRBx を停止します
PxIE[m]	←							1	該当ポートを TBxIN0 に割り付けます。
PxFR1[m]	←							1	
TBxFFCR	← X	X	0	0	0	0	1	1	TB x FF0 反転トリガをディセーブルします。
TBxMOD	← X	0	1	0	0	0	0	0	入力クロックを TBxIN0 にします。
TBxRUN	← *	*	*	*	*	1	X	1	TMRBx を起動します。
TBxMOD	← X	0	0	0	0	0	0	0	ソフトウェアキャプチャを行います。

注 1) "m"はポートの該当ビットを示します

注 2) X; Don't care

-; No change



### 9.6.3 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 (UC) とタイマレジスタ (TBxRG0, TBxRG1) への設定値との一致によりタイマフリップフロップ (TBxFF) の反転トリガをかけることで、プログラマブル矩形波を TBxOUT 端子より出力することができます。ただし、TBxRG0 と TBxRG1 の設定値は次の条件を満たす必要があります。

$$(TBxRG0 \text{ 設定値}) < (TBxRG1 \text{ 設定値})$$

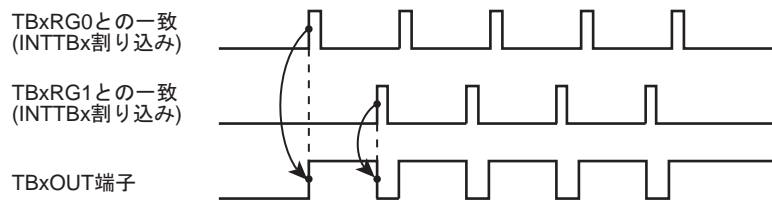


図 9-2 プログラマブル矩形波 (PPG) 出力波形例

このモードでは、TBxRG0 のダブルバッファをイネーブルにすることにより、TBxRG1 との一致で、レジスタバッファ 0 の値が TBxRG0 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。

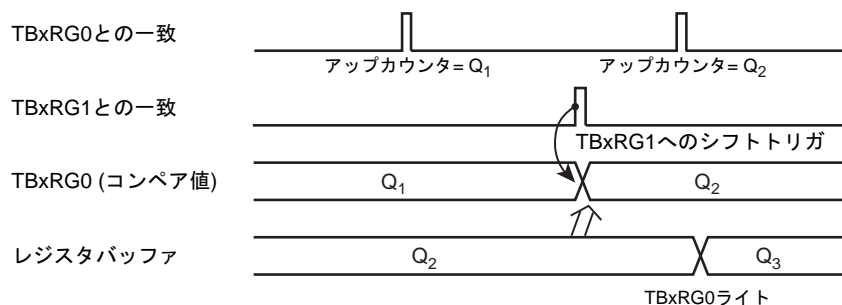


図 9-3 レジスタバッファの動作

このモードのブロック図を示します。

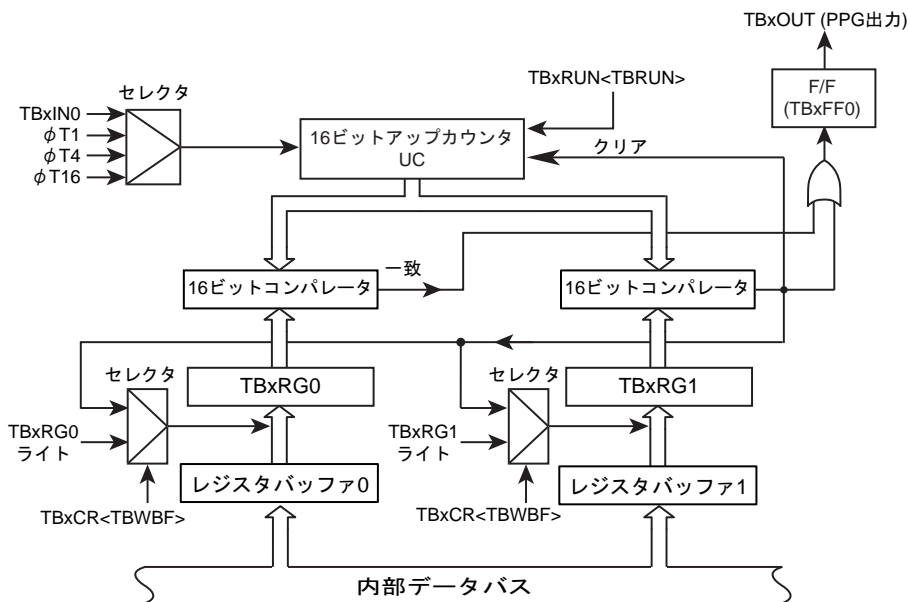


図 9-4 16 ビット PPG モードのブロック図

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	
TBxEN	← 1	X	X	X	X	X	X	X	TMRBx モジュールを起動します。
TBxRUN	← X	X	X	X	X	0	X	0	TMRBx を停止します
TBxCR	← 0	0	-	X	-	X	X	X	ダブルバッファをディセーブルします。
TBxRG0	← *	*	*	*	*	*	*	*	デューティを設定します。(16 ビット)
TBxRG1	← *	*	*	*	*	*	*	*	周期を設定します。(16 ビット)
TBxCR	← 1	0	X	0	0	0	0	0	TBxRG0 のダブルバッファイネーブル (INTTBx 割り込みでデューティ/周期の変更)
TBxFFCR	← X	X	0	0	1	1	1	0	TBxFF0 を TBxRG0, TBxRG1 との一致検出で反転するように設定します。また、TBxFF0 の初期値を "0" にします。
TBxMOD	← X	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)
PxCR[m]	←							1	該当ポートを TBxOUT に割り付けます。
PxFR1[m]	←							1	
TBxRUN	← *	*	*	*	*	1	X	1	TMRBx を起動します。

注 1) "m"はポートの該当ビットを示します

注 2) X; Don't care  
-; No change

#### 9.6.4 タイマ同期モード

タイマ同期モードを使用することにより、タイマ間のスタートの同期を取ることが可能となります。

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。

4 チャンルの TMRB が組になっており、4 チャンルのうちの 1 チャンルのスタートに、他の 3 チャンルのスタートを同期させることができます。TMPM330FDFG/FYFG/FWFG では以下の組み合わせで使用可能です。

スタートのトリガをかけるチャンネル (マスタチャンネル)	同期して動作するチャンネル (スレーブチャンネル)
TMRB0	TMRB1, TMRB2, TMRB3
TMRB4	TMRB5, TMRB6, TMRB7

TBxCR<TBSYNC>ビットの設定により、同期モードの切り替えを行います。

- ・ <TBSYNC>="0" : チャンネルごとの個別動作
- ・ <TBSYNC>="1" : 同期動作

マスタチャンネルの<TBSYNC>ビットは"0"を設定してください。

スレーブチャンネルの<TBSYNC>ビットに "1"を設定するとマスタチャンネルのスタートに同期して動作がスタートします。スレーブチャンネルの TBxRUN<TBPRUN, TBRUN>ビットの設定は不要です。

## 9.7 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

### 9.7.1 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 UC 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。TBxIN0 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(TBxCP0)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 INTCAPx0 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(TBxRG0)には、TBxCP0 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(TBxRG1)には、TBxRG0 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。(TBxRG1 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(TBxFFCR<TBE1T1, TBE0T1>)に "11" を設定し、TBxUC と TBxRG0 との一致、および、TBxRG1 との一致により、タイマフリップフロップ(TBxFF0)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、INTTBx 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 9-5 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。

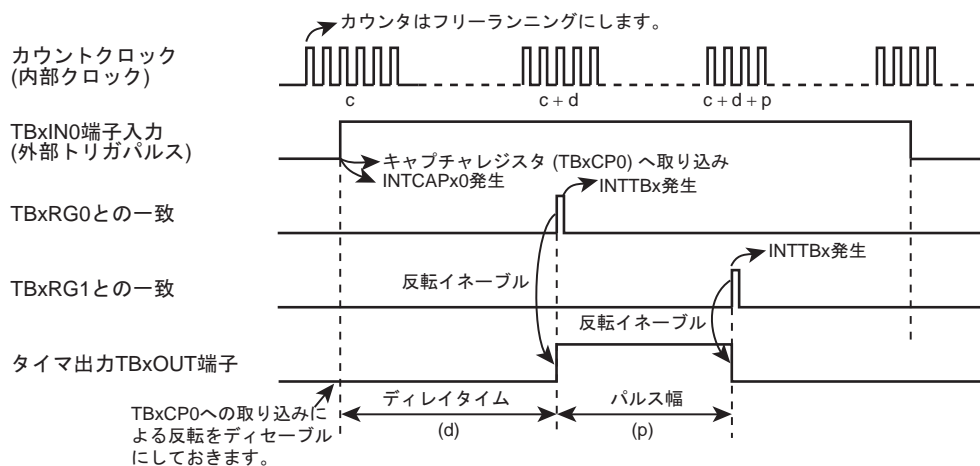


図 9-5 ワンショットパルス出力(ディレイあり)

TBxIN0 入力の立ち上がりトリガとして、3ms 後に 2ms 幅のワンショットパルスを出力する場合の設定例を以下に示します。ここではソースクロックに  $\Phi T1$  を使用しています。

	7	6	5	4	3	2	1	0		
[メイン処理] TBxIN0でのキャプチャ設定										
PxE[m]	←								1	該当ポートをTBxIN0に割り付けます。
PxFR1[m]	←								1	
TBxEN	←	1	X	X	X	X	X	X	X	TMRBxモジュールを起動します。
TBxRUN	←	X	X	X	X	X	0	X	0	TMRBxを停止します
TBxMOD	←	X	0	1	0	1	0	0	1	ソースクロックをΦT1にし、TBxIN0立ち上がりでTBxCP0へカウント値を取り込みます。
TBxFFCR	←	X	X	0	0	0	0	1	0	TB x FF0反転トリガをクリアし、ディセーブルします。
PxCR[m]	←								1	
PxFR1[m]	←								1	該当ポートをTBxOUTに割り付けます。
割り込みイネーブルセットレジスタ	←	*	*	*	*	*	*	*	*	INTCAPx0割り込みに対応するビットを"1"にし、割り込みを許可します。
TBxRUN	←	*	*	*	*	*	1	X	1	TMRBxを起動します。
[INTCAPx0 割り込みサービスルーチンでの処理] パルス出力設定										
TBxRG0	←	*	*	*	*	*	*	*	*	カウント値を設定します。(TBxCAP0 + 3ms/ΦT1)
TBxRG1	←	*	*	*	*	*	*	*	*	カウント値を設定します。(TBxCAP0 + (3+2)ms/ΦT1)
TBxFFCR	←	X	X	-	-	1	1	-	-	TBxRG0, TBxRG1 との一致でTBxFF0を反転します。
TBxIM	←	X	X	X	X	X	1	0	1	TBxRG1 との一致割り込み以外をマスクします。
割り込みイネーブルセットレジスタ	←	*	*	*	*	*	*	*	*	INTTBx 割り込みに対応するビットを"1"にし、割り込みを許可します。
[INTTBx 割り込みサービスルーチンでの処理] 出力ディセーブル										
TBxFFCR	←	X	X	-	-	0	0	-	-	TB x FF0反転トリガ設定をクリアします。
割り込みイネーブルクリアレジスタ	←	*	*	*	*	*	*	*	*	INTTBx 割り込みに対応するビットを"1"にし、割り込みを禁止します。

- 注 1) "m"はポートの該当ビットを示します
- 注 2) X; Don't care
- ; No change

ディレイが不要な場合、TBxCP0 への取り込みによって TBxFF0 を反転させ、割り込み INTCAPx0 で TBxCP0 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 TBxRG1 に設定します。(TBxRG1 の変更は次の一致までに完了してください。) TBxFF0 は、TBxRG1 と UC 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、INTTBx 割り込みでこれをディセーブルに戻します。

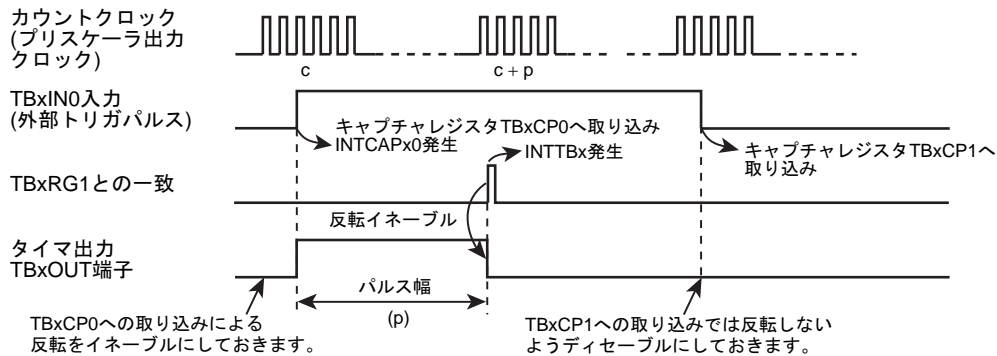


図 9-6 外部トリガパルスのワンショットパルス出力(ディレイなし)

### 9.7.2 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<TBCPM[1:0]>には"11"を設定することで、TB8OUTの立ち上がりでTB3CP0にカウンタ値取り込み、立下りでTB3CP1にカウンタ値取り込みます。

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UC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、キャプチャレジスタ(TB3CP1)にUC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。

例えば、TB8OUTの"1"レベル幅の設定値が0.5s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。

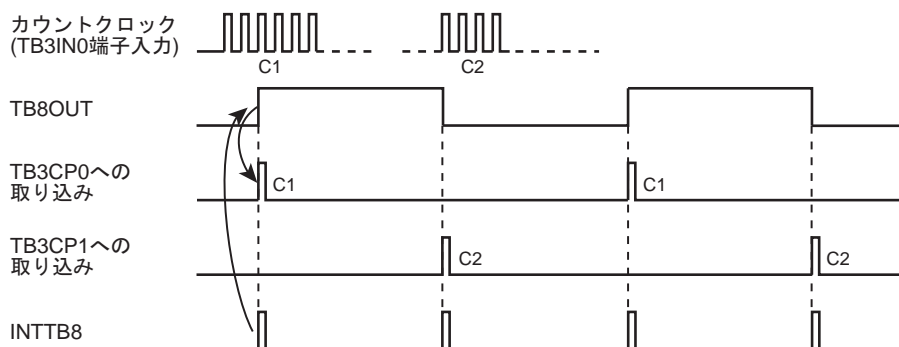


図 9-7 周波数測定

### 9.7.3 パルス幅測定

キャプチャ機能を用いて、外部パルスの"High"レベル幅を測定することができます。TBxIN0端子より外部パルスを入力し、アップカウンタ(UC)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(TBxCP0, TBxCP1)に取り込みます。TBxIN0端子の立ち下がりにより、INTCAPx1が発生するようにCPUで設定します。

"High"レベルパルス幅は、TBxCP0とTBxCP1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TBxCP0とTBxCP1の差が100で、プリスケアラ出力クロックの周期が0.5μs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UC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。

また、外部パルスの"Low"レベル幅を測定することもできます。この場合、「図 9-8 パルス幅測定」における、2 回目の INTCAPx0 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。

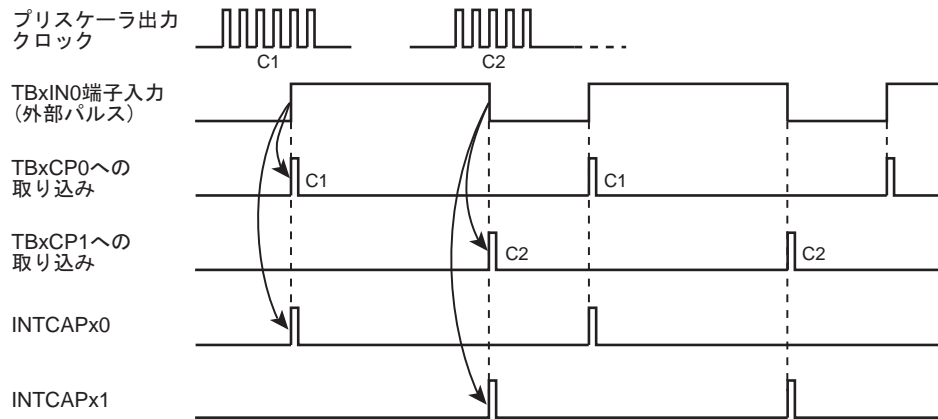


図 9-8 パルス幅測定

### 9.7.4 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(UC)をフリーランニングでカウントアップさせておきます。

TBxIN0 端子の入力パルスの立ち上がりエッジで、UC の値をキャプチャレジスタ(TBxCP0)に取り込みます。このとき、割り込み INTCAPx0 が発生するように CPU で設定します。

TBxIN1 端子の入力パルスの立ち上がりエッジで、UC の値をキャプチャレジスタ(TBxCP1)に取り込みます。このとき、割り込み INTCAPx1 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、TBxCP1 から TBxCP0 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。

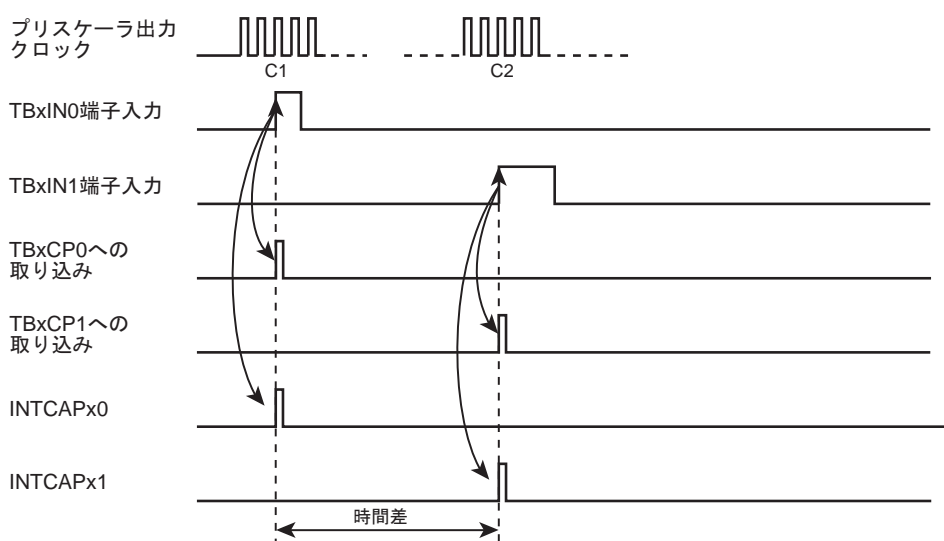


図 9-9 時間差測定





## 第 10 章 シリアルチャネル(SIO/UART)

### 10.1 概要

シリアルチャネル(SIO)は同期通信モード(IO インタフェースモード)と非同期通信モード(UART モード)の 2 つのモードを持っています。特長は以下のとおりです。

- ・ 転送クロック
  - プリスケアラでペリフェラルクロック( $\Phi T0$ )を 1/2、1/8、1/32、1/128 分周
  - プリスケアラ出力クロックに対し、1~16 分周が可能
  - プリスケアラ出力クロックに対し、 $N + m/16$  ( $N = 2 \sim 15, m = 1 \sim 15$ )分周が可能 (UART モードのみ)
  - システムクロックを使用可能(UART モードのみ)
- ・ ダブルバッファ/FIFO
  - ダブルバッファおよび、送受信あわせて最大 4 バイトの FIFO を使用可能
- ・ IO インタフェースモード
  - 転送モード：半二重(受信/送信)、全二重
  - クロック：出力(立ち上がりエッジ固定)/入力(立ち上がり/立ち下がりエッジ選択)
  - 連続転送時のインターバル時間設定が可能
- ・ UART モード
  - データ長：7, 8, 9 ビット
  - パリティ付加(9 ビット長では不可)
  - シリアルリンクでのウエイクアップ機能
  - $\overline{CTS}$  端子を用いたハンドシェイク機能

以下の説明中、"x"はチャネル番号をあらわします。

### 10.2 チャネル別仕様相違点

TMPM330FDFG/FYFG/FWFG は 3 チャネルの SIO を内蔵しています。

各チャネルはそれぞれ独立に動作します。チャネルごとの使用端子と割り込みを以下にまとめます。

表 10-1 SIO のチャネル別仕様相違点

		チャネル 0	チャネル 1	チャネル 2
端子	TXD	PE0(20pin)	PE4(23pin)	PF0(33pin)
	RXD	PE1(21pin)	PE5(24pin)	PF1(34pin)
	$\overline{CTS}$ /SCLK	PE2(22pin)	PE6(25pin)	PF2(35pin)
割り込み	受信割り込み	INTRX0	INTRX1	INTRX2
	送信割り込み	INTTX0	INTTX1	INTTX2

### 10.3 構成

図 10-1 に SIO のブロック図を示します。

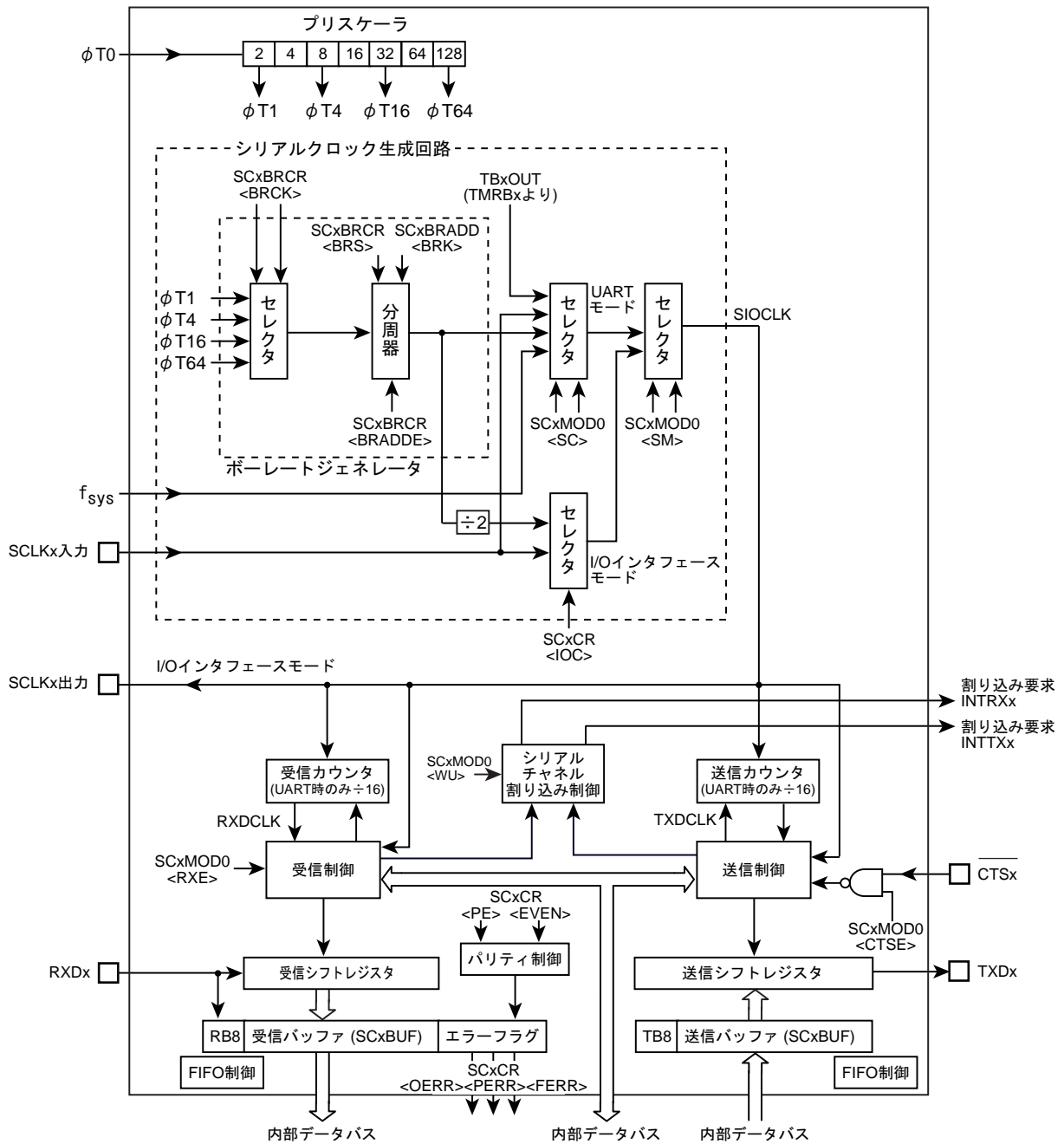


図 10-1 SIO ブロック図

## 10.4 レジスタ説明

### 10.4.1 チャネル別レジスタ一覧

各チャネルのレジスタとアドレスを示します。

Channel x	Base Address
Channel0	0x4002_0080
Channel1	0x4002_00C0
Channel2	0x4002_0100

レジスタ名(x=0~2)		Address(Base+)
イネーブルレジスタ	SCxEN	0x0000
送受信バッファレジスタ	SCxBUF	0x0004
コントロールレジスタ	SCxCR	0x0008
モードコントロールレジスタ 0	SCxMOD0	0x000C
ボーレートジェネレータコントロールレジスタ	SCxBRCR	0x0010
ボーレートジェネレータコントロールレジスタ 2	SCxBRADD	0x0014
モードコントロールレジスタ 1	SCxMOD1	0x0018
モードコントロールレジスタ 2	SCxMOD2	0x001C
受信 FIFO コンフィグレジスタ	SCxRFC	0x0020
送信 FIFO コンフィグレジスタ	SCxTFC	0x0024
受信 FIFO ステータスレジスタ	SCxRST	0x0028
送信 FIFO ステータスレジスタ	SCxTST	0x002C
FIFO コンフィグレジスタ	SCxFCNF	0x0030

注) 送信中、受信中に制御レジスタを書き換えないでください。

## 10.4.2 SCxEN (イネーブルレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	SIOE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	SIOE	R/W	<p>SIO 動作</p> <p>0: 禁止</p> <p>1: 動作</p> <p>SIO の動作を指定します。SIO を使用する場合は、まず&lt;SIOE&gt;に"1"をセットしてください。</p> <p>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。</p> <p>SIO を一旦動作させた後に動作禁止にした場合は、SCxTFC&lt;TIL[1:0]&gt;を除くレジスタの設定は保持されます。</p>

### 10.4.3 SCxBUF (バッファレジスタ)

SCxBUF は、書き込み時は送信バッファまたは FIFO、読み出し時は受信バッファまたは FIFO として機能します。

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TB / RB							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	TB[7:0] / RB [7:0]	R/W	[ライト] TB : 送信用バッファ / FIFO [リード] RB : 受信用バッファ / FIFO

## 10.4.4 SCxCR (コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RB8	EVEN	PE	OERR	PERR	FERR	SCLKS	IOC
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	RB8	R	受信データビット 8 (UART 用) 9 ビット UART モード時の 9 ビット目の受信データです。
6	EVEN	R/W	パリティ(UART 用) 0: Odd 1: 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。
5	PE	R/W	パリティ付加(UART 用) 0: 禁止 1: 許可 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。
4	OERR	R	オーバランエラー (注) 0: エラーではない 1: エラー
3	PERR	R	パリティ/アンダーランエラー (注) 0: エラーではない 1: エラー
2	FERR	R	フレーミングエラー (注) 0: エラーではない 1: エラー
1	SCLKS	R/W	入力クロックエッジ選択(IO インタフェース用) 0: 立ち上がり 1: 立ち下がり データ送受信で使用する入力クロックのエッジを選択します。 クロック出力モード時は、"0"を設定してください。
0	IOC	R/W	クロック選択(IO インタフェース用) 0: ボーレートジェネレータ 1: SCLK 端子入力

注) エラーフラグ(OERR, PERR, FERR)は読み出すとクリアされます。

10.4.5 SCxMOD0 (モードコントロールレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TB8	CTSE	RXE	WU	SM		SC	
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	TB8	R/W	送信データビット 8 (UART 用) 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。
6	CTSE	R/W	ハンドシェイク機能制御 (UART 用) 0: CTS ディセーブル 1: CTS イネーブル 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。
5	RXE	R/W	受信制御(注) 0: 禁止 1: 許可
4	WU	R/W	ウェイクアップ機能 (UART 用) 0: ディセーブル 1: イネーブル 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。 イネーブルの場合、受信データ 9 ビット目が"1"のときのみ割り込みが発生します。
3-2	SM[1:0]	R/W	シリアル転送モード 00: IO 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
1-0	SC[1:0]	R/W	シリアル転送クロック (UART 用) 00: タイマ TB9OUT 01: ポーレートジェネレータ 10: 内部クロック fsys 11: 外部クロック (SCLK 入力) (IO 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 (SCxCR) で選択します。)

注 1) <RXE>ビットは、各モードレジスタ(SCxMOD0, SCxMOD1, SCxMOD2)を設定してから許可してください。

注 2) 受信中に動作を停止(SCxMOD0<RXE>を"0"にクリア)しないでください。

## 10.4.6 SCxMOD1 (モードコントロールレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	I2SC	FDPX		TXE	SINT			-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	I2SC	R/W	IDLE モード時の動作 0: 停止 1: 動作 IDLE モード時の動作を指定します。
6-5	FDPX[1:0]	R/W	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 IO 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。
4	TXE	R/W	送信制御(注) 0: 禁止 1: 許可 全転送モードに有効な送信許可ビットです。
3-1	SINT[2:0]	R/W	連続転送時のインターバル時間(IO インタフェース用) 000: なし 001: 1SCLK 010: 2SCLK 011: 4SCLK 100: 8SCLK 101: 16SCLK 110: 32SCLK 111: 64SCLK IO インタフェースモードで SCLK 出力の場合に有効なビットです。その他のモードでは意味を持ちません。 IO インタフェースモードで、ダブルバッファまたは 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。
0	-	R/W	"0"をライトしてください。

注 1) <TXE>ビットは、すべての設定を行った後に許可してください。

注 2) 送信中に動作を停止(<TXE>を"0"にクリア)しないでください。



10.4.7 SCxMOD2 (モードコントロールレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBEMP	RBFL	TXRUN	SBLN	DRCHG	WBUF	SWRST	
リセット後	1	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能											
31-8	-	R	リードすると"0"が読めます。											
7	TBEMP	R	送信バッファエンプティフラグ 0: Full 1: Empty 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。											
6	RBFL	R	受信バッファ full フラグ 0: Empty 1: Full 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。											
5	TXRUN	R	送信動作中フラグ 0: 停止 1: 動作 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。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>&lt;TXRUN&gt;</th> <th>&lt;TBEMP&gt;</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>-</td> <td>送信動作中</td> </tr> <tr> <td rowspan="2">0</td> <td>1</td> <td>送信が完全に終了</td> </tr> <tr> <td>0</td> <td>送信バッファに次のデータがあり送信待ち</td> </tr> </tbody> </table>	<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
<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												
4	SBLN	R/W	送信 STOP ビット長(UART 用) 0: 1 ビット 1: 2 ビット 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。											
3	DRCHG	R/W	転送方向設定 0: LSB first 1: MSB first IO 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。											

Bit	Bit Symbol	Type	機能										
2	WBUF	R/W	ダブルバッファの許可 0: 不許可 1: 許可 IO 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、受信のダブルバッファの許可/不許可を指定します。 IO 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、常にダブルバッファは許可されます。										
1-0	SWRST[1:0]	R/W	ソフトリセット "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、以下のビットが初期化されます。また、送受信回路と FIFO は初期状態になります。(注 1)(注 2) <table border="1" data-bbox="571 622 1098 808"> <thead> <tr> <th>レジスタ名</th> <th>ビット</th> </tr> </thead> <tbody> <tr> <td>SCxMOD0</td> <td>RXE</td> </tr> <tr> <td>SCxMOD1</td> <td>TXE</td> </tr> <tr> <td>SCxMOD2</td> <td>TBEMP, RBFL, TXRUN</td> </tr> <tr> <td>SCxCR</td> <td>OERR, PERR, FERR</td> </tr> </tbody> </table>	レジスタ名	ビット	SCxMOD0	RXE	SCxMOD1	TXE	SCxMOD2	TBEMP, RBFL, TXRUN	SCxCR	OERR, PERR, FERR
レジスタ名	ビット												
SCxMOD0	RXE												
SCxMOD1	TXE												
SCxMOD2	TBEMP, RBFL, TXRUN												
SCxCR	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。

10.4.8 SCxBRCR (ポーレートジェネレータコントロールレジスタ),  
SCxBRADD (ポーレートジェネレータコントロールレジスタ 2)

ポーレートジェネレータの分周値は、下記の2つのレジスタで設定します。

SCxBRCR

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	BRADDE	BRCK		BRS			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	-	R/W	"0"をライトしてください。
6	BRADDE	R/W	$N + (16 - K)/16$ 分周機能(UART 用) 0: ディセーブル 1: イネーブル この機能は、UART モードのときのみ使用可能です。
5-4	BRCK[1:0]	R/W	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: $\phi T64$
3-0	BRS[3:0]	R/W	分周値"N"の設定 0000: 16 分周 0001: 1 分周 0010: 2 分周 : 1111: 15 分周

SCxBRADD

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	BRK			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-4	-	R	リードすると"0"が読めます。
3-0	BRK[3:0]	R/W	N + (16 - K)/16 分周の K 値の設定(UART 用) 0000: 設定禁止 0001: K = 1 0010: K = 2 : 1111: K = 15

ボーレートジェネレータ分周値の設定方法を表 10-2 にまとめます。

表 10-2 分周値の設定方法

	<BRADDE> = "0"のとき	<BRADDE> = "1"のとき (注 1) (UART モードのみ使用可能)
<BRS>の設定	分周値"N"を設定 (注 2) (注 3)	
<BRK>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<BRK>に"K"値を設定後に<BRADDE> = "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) UART モードで 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 3) IO インタフェースモードの場合、分周値"N"に 1 分周("0001")を設定できるのはダブルバッファを使用する場合のみです。
- 注 4) "K"値に"0"を設定することはできません。

10.4.9 SCxFCNF (FIFO コンフィグレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	RFST	TFIE	RFIE	RXXCNT	CNFG
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能						
31-8	-	R	リードすると"0"が読めます。						
7-5	-	R/W	必ず"000"を書き込んでください。						
4	RFST	R/W	受信 FIFO 使用バイト数 0: 最大 1: 受信 FIFO の FILL レベルに同じ 受信 FIFO 使用バイト数の設定ビットです。(注 1) "0"の場合、構成されている FIFO の最大のバイト数(<CNFG>ビットの説明を参照。)が使用可能です。 "1"の場合、SCxRFC<RIL[1:0]>で指定された FILL レベルのバイト数になります。						
3	TFIE	R/W	送信 FIFO 使用時の送信割り込み許可 0: 禁止 1: 許可 送信 FIFO が有効にされている時の送信割り込みの禁止/許可を切り替えます。						
2	RFIE	R/W	受信 FIFO 使用時の受信割り込み許可 0: 禁止 1: 許可 受信 FIFO が有効にされている時の受信割り込みの禁止/許可を切り替えます。						
1	RXTXCNT	R/W	RXE/TXE の自動禁止 0: なし 1: 自動禁止 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。 <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">半二重受信</td> <td>受信シフトレジスタ、受信バッファおよび受信 FIFO すべてにデータが格納されると自動的に受信許可ビット(SCxMOD0 &lt;RXE&gt;)が"0"となり、受信が禁止される。</td> </tr> <tr> <td>半二重送信</td> <td>送信 FIFO、送信バッファおよび送信シフトレジスタのすべてのデータ送信が終了すると自動的に送信許可ビット(SCxMOD1 &lt;TXE&gt;)が"0"となり、送信が禁止される。</td> </tr> <tr> <td>全二重</td> <td>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。</td> </tr> </table>	半二重受信	受信シフトレジスタ、受信バッファおよび受信 FIFO すべてにデータが格納されると自動的に受信許可ビット(SCxMOD0 <RXE>)が"0"となり、受信が禁止される。	半二重送信	送信 FIFO、送信バッファおよび送信シフトレジスタのすべてのデータ送信が終了すると自動的に送信許可ビット(SCxMOD1 <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。
半二重受信	受信シフトレジスタ、受信バッファおよび受信 FIFO すべてにデータが格納されると自動的に受信許可ビット(SCxMOD0 <RXE>)が"0"となり、受信が禁止される。								
半二重送信	送信 FIFO、送信バッファおよび送信シフトレジスタのすべてのデータ送信が終了すると自動的に送信許可ビット(SCxMOD1 <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。								
0	CNFG	R/W	FIFO の許可 0: 禁止 1: 許可 FIFO 使用の許可ビットです。(注 2) "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 の構成は以下のようになります。 <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;">半二重受信</td> <td>受信 FIFO 4 バイト</td> </tr> <tr> <td>半二重送信</td> <td>送信 FIFO 4 バイト</td> </tr> <tr> <td>全二重</td> <td>受信 FIFO 2 バイト + 送信 FIFO 2 バイト</td> </tr> </table>	半二重受信	受信 FIFO 4 バイト	半二重送信	送信 FIFO 4 バイト	全二重	受信 FIFO 2 バイト + 送信 FIFO 2 バイト
半二重受信	受信 FIFO 4 バイト								
半二重送信	送信 FIFO 4 バイト								
全二重	受信 FIFO 2 バイト + 送信 FIFO 2 バイト								

注 1) 送信 FIFO は常に構成されている FIFO の最大バイト数を使用できます。

注 2) 9 ビット UART モードでは FIFO は使用できません。

10.4.10 SCxRFC (受信 FIFO コンフィグレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RFCS	RFIS	-	-	-	-	RIL	
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能															
31-8	-	R	リードすると"0"が読めます。															
7	RFCS	W	受信 FIFO クリア(注) 1: クリア "1"を書き込むと受信 FIFO がクリアされます。リードすると"0"が読めます。															
6	RFIS	R/W	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															
5-2	-	R	リードすると"0"が読めます。															
1-0	RIL[1:0]	R/W	受信割り込みが発生する FIFO の fill レベル <table border="1" style="margin-left: 20px;"> <thead> <tr> <th></th> <th>半二重</th> <th>全二重</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>4 バイト</td> <td>2 バイト</td> </tr> <tr> <td>01</td> <td>1 バイト</td> <td>1 バイト</td> </tr> <tr> <td>10</td> <td>2 バイト</td> <td>2 バイト</td> </tr> <tr> <td>11</td> <td>3 バイト</td> <td>1 バイト</td> </tr> </tbody> </table>		半二重	全二重	00	4 バイト	2 バイト	01	1 バイト	1 バイト	10	2 バイト	2 バイト	11	3 バイト	1 バイト
	半二重	全二重																
00	4 バイト	2 バイト																
01	1 バイト	1 バイト																
10	2 バイト	2 バイト																
11	3 バイト	1 バイト																

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

## 10.4.11 SCxTFC (送信 FIFO コンフィグレジスタ) (注 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TFCS	TFIS	-	-	-	-	TIL	
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能															
31-8	-	R	リードすると"0"が読めます。															
7	TFCS	W	送信 FIFO クリア(注 1) 1: クリア "1"を書き込むと送信 FIFO がクリアされます。リードすると"0"が読めます。															
6	TFIS	R/W	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															
5-2	-	R	リードすると"0"が読めます。															
1-0	TIL[1:0]	R/W	送信割り込みが発生する FIFO の fill レベル <table border="1" style="margin-left: 20px;"> <thead> <tr> <th></th> <th>半二重</th> <th>全二重</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>Empty</td> <td>Empty</td> </tr> <tr> <td>01</td> <td>1 バイト</td> <td>1 バイト</td> </tr> <tr> <td>10</td> <td>2 バイト</td> <td>Empty</td> </tr> <tr> <td>11</td> <td>3 バイト</td> <td>1 バイト</td> </tr> </tbody> </table>		半二重	全二重	00	Empty	Empty	01	1 バイト	1 バイト	10	2 バイト	Empty	11	3 バイト	1 バイト
	半二重	全二重																
00	Empty	Empty																
01	1 バイト	1 バイト																
10	2 バイト	Empty																
11	3 バイト	1 バイト																

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

注 2) 以下の動作を行った際は、SCxTFC レジスタを再度設定してください。

- ・ SCxEN<SIOE>="0" (SIO 動作停止)

- ・ SCxMOD1<I2SC>="0" (IDLE モード時動作禁止)設定で、WFI 命令による低消費電力モードへの遷移からの復帰後



## 10.4.12 SCxRST (受信 FIFO ステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ROR	-	-	-	-	RLVL		
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	ROR	R	受信 FIFO オーバラン(注) 0: オーバランは発生していない 1: オーバラン発生
6-3	-	R	リードすると"0"が読めます。
2-0	RLVL[2:0]	R	受信 FIFO の fill レベル 000: Empty 001: 1 バイト 010: 2 バイト 011: 3 バイト 100: 4 バイト

注) <ROR>ビットはバッファレジスタ(SCxBUF)を読み出すと"0"にクリアされます。

## 10.4.13 SCxTST (送信 FIFO ステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TUR	-	-	-	-	TLVL		
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	TUR	R	送信 FIFO アンダーラン(注) 0: アンダーランは発生していない 1: アンダーラン発生
6-3	-	R	リードすると"0"が読めます。
2-0	TLVL[2:0]	R	送信 FIFO の fill レベル 000: Empty 001: 1 バイト 010: 2 バイト 011: 3 バイト 100: 4 バイト

注) <TUR>ビットはバッファレジスタ(SCxBUF)に書き込みを行うと"0"にクリアされます。

## 10.5 動作モード

表 10-3 にモードとデータフォーマットをまとめます。

表 10-3 モードとデータフォーマット

モード	種類	データ長	転送方向	パリティ付加	STOP ビット長(送信)
モード 0	同期通信モード (IO インタフェースモード)	8 ビット	LSB first/MSB first	-	-
モード 1	非同期通信モード (UART モード)	7 ビット	LSB first	○	1 ビットまたは 2 ビット
モード 2		8 ビット		○	
モード 3		9 ビット		×	

モード 0 は同期通信モードで、IO を拡張するために使用できます。SCLK に同期してデータの送受信を行います。SCLK は入力/出力いずれも使用できます。

転送方向は、LSB first と MSB first から選択可能です。パリティ付加機能はなく、STOP ビットも使用しません。

モード 1 からモード 3 は非同期通信モードです。転送方向は LSB first 固定です。

モード 1 とモード 2 はパリティビットの付加が可能です。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エイクアップ機能を有しています。

送信時の STOP ビットを 1 ビットまたは 2 ビットから選択できます。受信時の STOP ビット長は 1 ビット固定です。

## 10.6 データフォーマット

### 10.6.1 データフォーマット一覧

図 10-2 にデータフォーマットを示します。

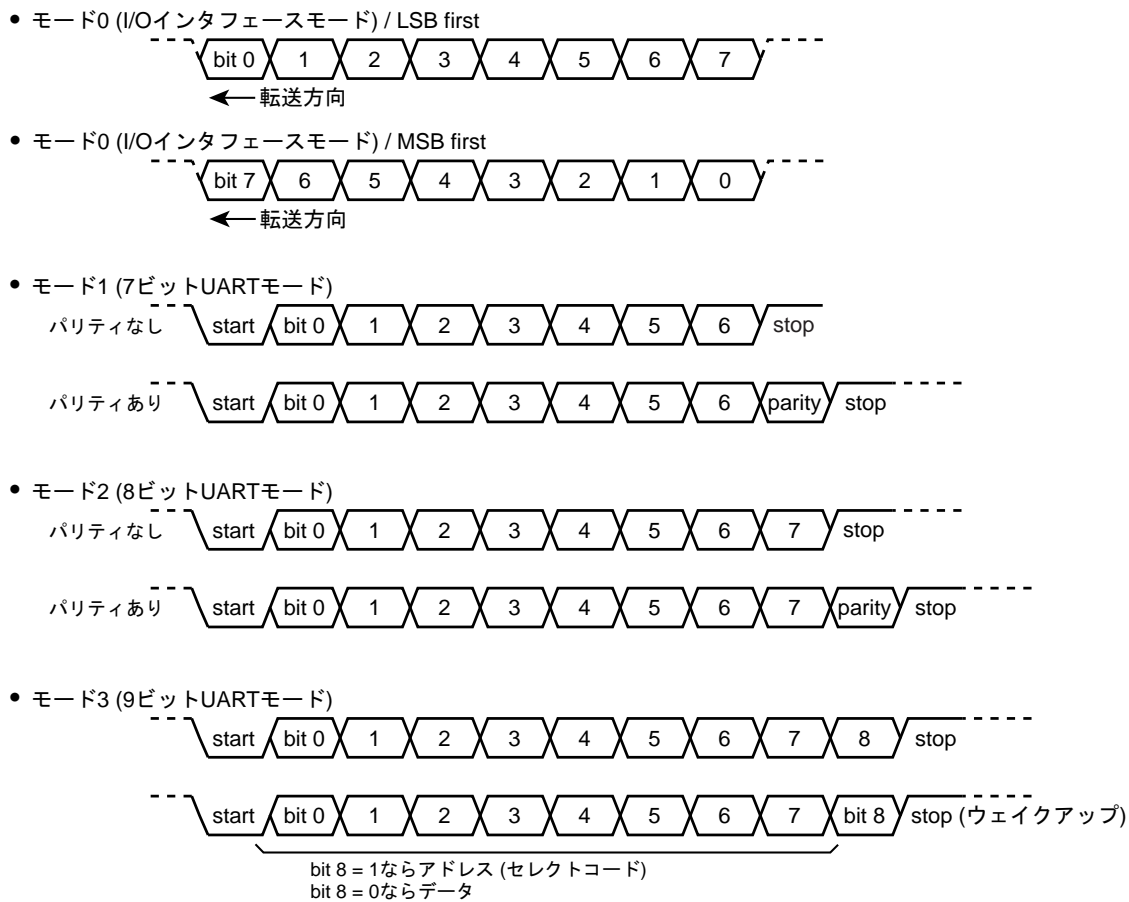


図 10-2 データフォーマット

## 10.6.2 パリティ制御

7ビットUARTモードまたは8ビットUARTモードでは送信データにパリティビットを付加することができます。

SCxCR<PE>に"1"を設定するとパリティが有効になります。SCxCR<EVEN>で偶数/奇数パリティを選択することができます。

### 10.6.2.1 送信

送信時、パリティ制御回路は送信バッファのデータに対して自動的にパリティを発生し、7ビットUARTモードのときはSCxBUF<TB7>に、8ビットUARTモードのときはSCxMOD<TB8>にパリティが格納されます。

なお<PE>と<EVEN>の設定は、送信データをバッファレジスタに書き込む前に行ってください。

### 10.6.2.2 受信

受信データが受信シフトレジスタから受信バッファに移されるとパリティを自動発生し、7ビットUARTモードのときはSCxBUF<RB7>と、8ビットUARTモードのときはSCxCR<RB8>と生成したパリティを比較します。異なっているとパリティエラーが発生し、SCxCR<PERR>がセットされます。

FIFOを使用する場合、<PERR>は受信したいずれかのデータでパリティエラーが発生したことを示します。

## 10.6.3 STOP ビット長

SCxMOD2<SBLN>で、UART送信モードのSTOPビット長を1ビットまたは2ビットに設定できます。受信の場合にはこのビットの設定にかかわらず1ビットのSTOPビット長として認識します。

## 10.7 クロック制御

### 10.7.1 プリスケーラ

7ビットのプリスケーラを実装しており、 $\Phi T0$  の 2 / 8 / 32 / 128 分周のクロックを生成します。

プリスケーラの入力クロック  $\Phi T0$  は、クロック/モード制御部の CGSYSCR レジスタで選択します。

プリスケーラは、 $SCxMOD0<SC[1:0]>="11"$  でボーレートジェネレータを転送クロックとして選択した場合に動作します。

ボーレートジェネレータへの入力クロック分解能を、表 10-4 (40MHz 動作時)、表 10-5 (32MHz 動作時)に示します。

表 10-4 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 クロック選択 CGSYSCR <FPSEL>	クロック ギア値 CGSYSCR <GEAR[2:0]>	プリスケーラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケーラ出カクロック 分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
0 (fgear)	000 (fc)	000 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 (fperiph/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 $\mu\text{s}$ )

表 10-4 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 クロック選択 CGSYSCR <FPSEL>	クロック ギア値 CGSYSCR <GEAR[2:0]>	プリスケアラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケアラ出カクロック 分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
1 (fc)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.05 $\mu\text{s}$ )	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.1 $\mu\text{s}$ )	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )	$fc/2^{11}$ (51.2 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )	$fc/2^{12}$ (102.4 $\mu\text{s}$ )
	100 (fc/2)	000 (fperiph/1)	–	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.1 $\mu\text{s}$ )	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )	$fc/2^{11}$ (51.2 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )	$fc/2^{12}$ (102.4 $\mu\text{s}$ )
	101 (fc/4)	000 (fperiph/1)	–	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		001 (fperiph/2)	–	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.2 $\mu\text{s}$ )	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )	$fc/2^{11}$ (51.2 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )	$fc/2^{12}$ (102.4 $\mu\text{s}$ )
	110 (fc/8)	000 (fperiph/1)	–	–	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )
		001 (fperiph/2)	–	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )
		010 (fperiph/4)	–	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.4 $\mu\text{s}$ )	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.8 $\mu\text{s}$ )	$fc/2^7$ (3.2 $\mu\text{s}$ )	$fc/2^9$ (12.8 $\mu\text{s}$ )	$fc/2^{11}$ (51.2 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (1.6 $\mu\text{s}$ )	$fc/2^8$ (6.4 $\mu\text{s}$ )	$fc/2^{10}$ (25.6 $\mu\text{s}$ )	$fc/2^{12}$ (102.4 $\mu\text{s}$ )

注 1) プリスケアラ出カクロック  $\phi T_n$  は、必ず  $\phi T_n \leq f_{\text{sys}}/2$  を満足するように ( $\phi T_n$  が  $f_{\text{sys}}$  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。

表 10-5 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 クロック選択 CGSYSCR <FPSEL>	クロック ギア値 CGSYSCR <GEAR[2:0]>	プリスケアラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケアラ出カクロック 分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
0 (fgear)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.0625 $\mu\text{s}$ )	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.125 $\mu\text{s}$ )	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )
	100 (fc/2)	000 (fperiph/1)	$fc/2^1$ (0.0625 $\mu\text{s}$ )	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )	$fc/2^{13}$ (256.0 $\mu\text{s}$ )
	101 (fc/4)	000 (fperiph/1)	$fc/2^1$ (0.0625 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )	$fc/2^{13}$ (256.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )	$fc/2^{14}$ (512.0 $\mu\text{s}$ )
	110 (fc/8)	000 (fperiph/1)	$fc/2^1$ (0.0625 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )	$fc/2^{13}$ (256.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )	$fc/2^{14}$ (512.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )	$fc/2^{13}$ (256.0 $\mu\text{s}$ )	$fc/2^{15}$ (1024 $\mu\text{s}$ )



表 10-5 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 クロック選択 CGSYSCR <FPSEL>	クロック ギア値 CGSYSCR <GEAR[2:0]>	プリスケアラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケアラ出カクロック 分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
1 (fc)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.0625 $\mu\text{s}$ )	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.125 $\mu\text{s}$ )	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )
	100 (fc/2)	000 (fperiph/1)	-	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.125 $\mu\text{s}$ )	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )
	101 (fc/4)	000 (fperiph/1)	-	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.25 $\mu\text{s}$ )	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )
	110 (fc/8)	000 (fperiph/1)	-	-	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )
		010 (fperiph/4)	-	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.5 $\mu\text{s}$ )	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (1.0 $\mu\text{s}$ )	$fc/2^7$ (4.0 $\mu\text{s}$ )	$fc/2^9$ (16.0 $\mu\text{s}$ )	$fc/2^{11}$ (64.0 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (2.0 $\mu\text{s}$ )	$fc/2^8$ (8.0 $\mu\text{s}$ )	$fc/2^{10}$ (32.0 $\mu\text{s}$ )	$fc/2^{12}$ (128.0 $\mu\text{s}$ )

- 注 1) プリスケアラ出カクロック  $\phi T_n$  は、必ず  $\phi T_n \leq f_{\text{sys}}/2$  を満足するように ( $\phi T_n$  が  $f_{\text{sys}}$  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。

## 10.7.2 シリアルクロック生成回路

送受信クロック(SIOCLK)を生成するブロックで、ボーレートジェネレータとモードによりクロックを選択する回路で構成されています。

### 10.7.2.1 ボーレートジェネレータ

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。

#### (1) ボーレートジェネレータ入力クロック

ボーレートジェネレータの入力クロックは、プリスケアラ出力の 2 / 8 / 32 / 128 分周から選択します。入力クロックの選択は SCxBRCR<BRCK>で行います。

#### (2) ボーレートジェネレータ出力クロック

ボーレートジェネレータの出力クロック分周値は、SCxBRCR、SCxBRADD で設定します。

IO インタフェースモードでは N 分周、UART モードでは N 分周または  $N + (16-K)/16$  分周が使用できます。

以下に設定可能な分周値をまとめます。

モード	分周機能設定 SCxBRCR<BRADDE>	N 分周値 SCxBRCR<BRS>	K 値 SCxBRADD<BRK>
IO インタフェース	N 分周	1 ~ 16 (注)	-
UART	N 分周	1 ~ 16	-
	$N + (16-K)/16$ 分周	2 ~ 15	1 ~ 15

注) 1 分周は、ダブルバッファ許可時のみ使用できます。

### 10.7.2.2 クロック選択回路

モードとレジスタ設定により、クロックが選択されます。

モードは SCxMOD0<SM>で指定します。

IO インタフェースモード時のクロックは、SCxCR で設定します。

UART モード時のクロックは、SCxMOD0<SC>で設定します。

#### (1) IO インタフェースモードの転送クロック

表 10-6 に IO インタフェースモードで可能なクロックを示します。

表 10-6 IO インタフェースモードのクロック選択

モード SCxMOD0<SM>	入出力選択 SCxCR<IOC>	クロックエッジ選択 SCxCR<SCLKS>	使用クロック
IO インタフェース モード	SCLK 出力	"0"で使用 (立ち上がり固定)	ボーレートジェネレータ出力の 2分周
	SCLK 入力	立ち上がり	SCLK 入力立ち上がりエッジ
		立下り	SCLK 入力立ち下がりエッジ

ボーレートジェネレータを使用する場合、以下の設定が最高ボーレートとなります。

注) AC 電気的特性を満足することを確認のうえ、クロック設定を決定してください。

- ・ クロック/モード制御部の設定
  - fc = 40MHz
  - fgear = 40MHz (CGSYSCR<GEAR[2:0]> = "000" : fc 選択)
  - ΦT0 = 40MHz (CGSYSCR<PRCK[2:0]> = "000" : 1 分周)
- ・ SIO の設定(ダブルバッファ使用の場合)
  - クロック選択 (SCxBRCR<BRCK[1:0]> = "00" : ΦT1 選択) = 20MHz
  - 分周値 (SCxBRCR<BRS[3:0]> = "0001" : 1 分周) = 20MHz

ダブルバッファ使用の場合、1 分周が選択できます。ボーレートは 20MHz が 2 分周され、10Mbps となります。
- ・ SIO の設定(ダブルバッファ未使用の場合)
  - クロック選択(SCxBRCR<BRCK[1:0]> = "00" : ΦT1 選択) = 20MHz
  - 分周値(SCxBRCR<BRS[3:0]> = "0010" : 2 分周) = 10MHz

ダブルバッファ未使用の場合は、2 分周が最速になります。ボーレートは 10MHz が 2 分周され、5Mbps となります。

SCLK 入力を使用する場合、以下の条件を満足する必要があります。

- ・ ダブルバッファ使用の場合
  - SCLK 周期 > 6/fsys

最高ボーレートは、 $40 \div 6 = 6.66$  Mbps 未満となります。
- ・ ダブルバッファ未使用の場合
  - SCLK 周期 > 8/fsys

最高ボーレートは、 $40 \div 8 = 5.0$  Mbps 未満となります。

## (2) UART モードの転送クロック

表 10-7 に UART モードの場合のクロック選択を示します。UART モードでは、選択されたクロックを受信/送信カウンタでさらに 16 分周して使用します。

表 10-7 UART モードのクロック選択

モード SCxMOD0<SM>	クロック選択 SCxMOD0<SC>
UART モード	タイマ出力
	ボーレートジェネレータ
	f <sub>sys</sub>
	SCLK 入力

それぞれのクロックでのボーレート例を示します。

- ・ ボーレートジェネレータを使用する場合
  - f<sub>c</sub> = 40MHz
  - f<sub>gear</sub> = 40MHz (CGSYSCR<GEAR[2:0]> = "000" : f<sub>c</sub> 選択)
  - ΦT0 = 40MHz (CGSYSCR<PRCK[2:0]> = "000" : 1 分周)
  - クロック選択 = ΦT1 = 20MHz (SCxBRCR<BRCK[1:0]> = "00" : ΦT1 選択)

最高ボーレートは 20MHz が 16 分周され、1.25Mbps となります。

表 10-8 に以下のクロック設定でボーレートジェネレータを使用する場合のボーレート例を示します。

- ・ f<sub>c</sub> = 9.8304MHz
- ・ f<sub>gear</sub> = 9.8304MHz (CGSYSCR<GEAR[2:0]> = "000" : f<sub>c</sub> 選択)
- ・ ΦT0 = 4.9152MHz (CGSYSCR<PRCK[2:0]> = "001" : 2 分周)

表 10-8 UART モードのボーレート例(ボーレートジェネレータ使用)

f <sub>c</sub> [MHz]	分周値 N (SCxBRCR<BRS[3:0]>)	φ T1 (f <sub>c</sub> /4)	φ T4 (f <sub>c</sub> /16)	φ T16 (f <sub>c</sub> /64)	φ T64 (f <sub>c</sub> /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
	16	9.600	2.400	0.600	0.150

単位 : kbps

- ・ SCLK 入力を使用する場合

SCLK 入力を使用する場合、以下の条件を満足する必要があります。

  - SCLK 周期 > 2/f<sub>sys</sub>

最高ボーレートは、 $40 \div 2 \div 16 = 1.25$  Mbps 未満にする必要があります。
- ・ f<sub>sys</sub> を使用する場合

f<sub>sys</sub> の最高が 40MHz ですので、最高ボーレートは、 $40 \div 16 = 2.5$  Mbps となります。
- ・ タイマ出力を使用する場合

タイマの出力を使用する場合、カウンタと TBxRG0 の一致でタイマフリップフロップ出力を反転させる設定とします。SIOCLK クロック周期は「TBxRG0 設定値 × 2」となります。

ボーレートは以下の計算式で求められます。

## ボーレートの算出方法

$$\text{転送レート} = \frac{\text{CGSYSCR}\langle\text{PRCK}[1:0]\rangle \text{で選択されたクロック周波数}}{(\text{TBxRG0} \times 2) \times \underline{2} \times 16}$$

↑ (タイマプリスケールクロックφT1 (2分周) を選択した場合)  
 ↑ (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)

表 10-9 に以下のクロック設定でタイマ出力を使用する場合のボーレート例を示します。

- ・ fc = 32MHz / 9.8304MHz / 8MHz
- ・ fgear = 32MHz / 9.8304MHz / 8MHz (CGSYSCR<GEAR[2:0]> = "000" : fc 選択)
- ・ φT0 = 16MHz / 4.9152MHz / 4MHz (CGSYSCR<PRCK[2:0]> = "001" : 2 分周)
- ・ タイマカウントクロック  
= 4MHz / 1.2287MHz / 1MHz (TBxMOD<TBCLK[1:0]> = "01" : φT1 選択)

表 10-9 UART モードのボーレート例(タイマ出力使用)

TBxRG0 設定	fc		
	32MHz	9.8304MHz	8MHz
0x0001	250	76.8	62.5
0x0002	125	38.4	31.25
0x0003	-	25.6	-
0x0004	62.5	19.2	15.625
0x0005	50	15.36	12.5
0x0006	-	12.8	-
0x0008	31.25	9.6	-
0x000A	25	7.68	6.25
0x0010	15.625	4.8	-
0x0014	12.5	3.84	3.125

単位 : kbps

## 10.8 送信/受信バッファと FIFO

### 10.8.1 構成

送信/受信バッファと FIFO の構成を図 10-3 に示します。

バッファと FIFO を使用するには設定が必要です。また、モードによっては構成が決まっている場合があります。

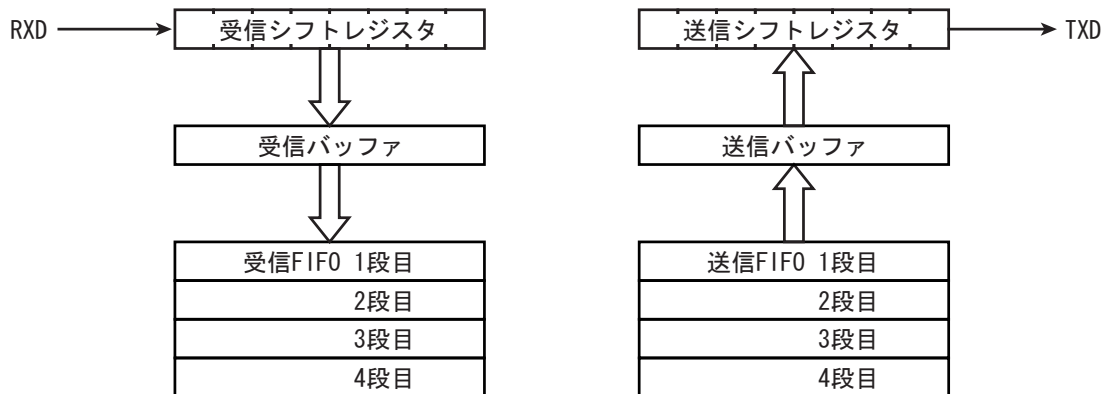


図 10-3 バッファと FIFO の構成

### 10.8.2 送信/受信バッファ

送信および受信バッファはダブルバッファ構造となっています。バッファ構成の設定は、SCxMOD2<WBUF>で行います。

受信の場合、IO インタフェースモードで SCLK 入力の場合と、UART モードでは<WBUF>の設定によらずダブルバッファ構成になります。その他のモードでは<WBUF>の設定に従います。

表 10-10 にモードとバッファ構成の関係をまとめます。

表 10-10 モードとバッファ構成

モード		SCxMOD2<WBUF>	
		"0"	"1"
UART	送信	シングル	ダブル
	受信	ダブル	ダブル
IO 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
IO 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

### 10.8.3 FIFO

ダブルバッファに加えて、4byte の FIFO を使用することができます。

FIFO を有効にするには SCxMOD2<WBUF>を"1"としてダブルバッファをイネーブルにし、SCxFCNF<CNFG>に"1"をセットします。FIFO バッファの構成は SCxMOD1<FDPX>で設定します。

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

表 10-11 にモードと FIFO 構成の関係をまとめます。

表 10-11 モードと FIFO 構成

	SCxMOD1<FDPX[1:0]>	受信 FIFO	送信 FIFO
半二重受信	"01"	4byte	-
半二重送信	"10"	-	4byte
全二重	"11"	2byte	2byte

## 10.9 ステータスフラグ

SCxMOD2 レジスタに 2 種類のステータスフラグが準備されています。これらのフラグはダブルバッファ許可設定時のみ意味を持ちます。

<RBFL>は、受信バッファ full を示すフラグです。1 データの受信が終了し、データが受信シフトレジスタから受信バッファに移されると"1"にセットされます。受信バッファを読み出すと"0"にクリアされます。

<TBEMP>は、送信バッファ empty を示すフラグです。送信バッファから送信シフトレジスタへデータが移されると、"1"がセットされます。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。

## 10.10 エラーフラグ

SCxCR レジスタに 3 種類のエラーフラグが準備されています。フラグによってはモードにより意味が変わります。以下にモードごとのフラグの意味をまとめます。

これらのフラグは、SCxCR レジスタを読み出すと"0"にクリアされます。

モード	フラグ		
	<OERR>	<PERR>	<FERR>
UART	オーバランエラー	パリティエラー	フレーミングエラー
IO インタフェース (SCLK 入力)	オーバランエラー	アンダーランエラー (ダブルバッファ または FIFO 使用時)	"0"固定
		"0"固定 (ダブルバッファ および FIFO 未使用時)	
IO インタフェース (SCLK 出力)	不定	不定	"0"固定

### 10.10.1 OERR フラグ

UART モード、IO インタ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"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 ヘッダが自動的に移されるので、受信 FIFO が full (使用バイト数) になるまではフラグはセットされません。

IO インタフェースモードの SCLK 出力の設定では、フラグのセットとともに SCLK 出力が停止します。

注) IO インタフェース SCLK 出力モードから他のモードへ切り替える際には、SCxCR レジスタを読み出し、オーバーランフラグをクリアしてください。

### 10.10.2 PERR フラグ

UART モードではパリティエラーを、IO インタフェースモードではアンダーランエラーまたは送信終了を示します。

パリティエラーは UART モードで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"1"にセットされます。

アンダーランエラーは、IO インタフェースモードでダブルバッファが有効な場合に以下の条件で"1"にセットされます。

SCLK 入力の場合、送信シフトレジスタのデータを送信終了後、送信バッファにデータがない状態で次の転送クロックが入力されるとセットされます。

SCLK 出力の場合、すべてのデータ送信が終了するとセットされクロック出力を停止します。

注) IO インタフェース SCLK 出力モードから他のモードへ切り替える際には、SCxCR レジスタを読み出し、アンダーランフラグをクリアしてください。

### 10.10.3 FERR フラグ

フレーミングエラーは UART モードの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。SCxMOD2<SBLN>でのストップビット長設定に関わらず、判定は第 1 ストップビットで行われます。

IO インタフェースモードではこのビットは"0"固定です。



## 10.11 受信

### 10.11.1 受信カウンタ

受信カウンタは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。

UARTモードでは、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。

### 10.11.2 受信制御部

#### 10.11.2.1 IOインタフェースモードの場合

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

SCxCR<IOC>="1"のSCLK入力モードのときは、SCxCR<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RXD端子をサンプリングします。

#### 10.11.2.2 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を判断して受信動作を開始します。

### 10.11.3 受信動作

#### 10.11.3.1 受信バッファの動作

受信シフトレジスタに受信データが1ビットずつ格納され、データがそろろうと割り込みINTRXxが発生します

ダブルバッファ設定の場合は、データは受信バッファ(SCxBUF)へ移され受信バッファのfullフラグ(SCxMOD2<RBFL>)が"1"にセットされます。受信バッファfullフラグは、受信バッファを読み出すと"0"にクリアされます。シングルバッファの場合、受信バッファfullフラグは意味を持ちません。

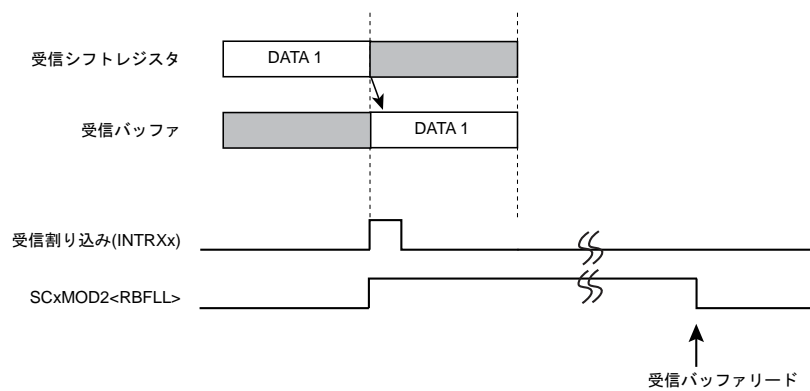


図 10-4 受信バッファの動作

10.11.3.2 受信 FIFO の動作

FIFO が許可されている場合、受信データは受信バッファから受信 FIFO に移され、受信バッファ full フラグはただちにクリアされます。割り込みは SCxRFC<RIL[1:0]>の設定に従って発生します。

注) UART モードで、FIFO 使用時にパリティビット付きのデータ受信する場合、パリティエラーは受信したいずれかのデータで発生したことを示します。

以下に、半二重受信の設定と動作を示します。

- SCxMOD1[6:5] = 01 : 転送モードを半二重受信に設定
- SCxFCNF[4:0] = 10111 : fill レベル到達後の継続受信自動禁止  
受信 FIFO の使用バイト数は割り込み発生 fill レベルに同じ
- SCxRFC[1:0] = 00 : 受信割り込みが発生する FIFO の fill レベルを 4 バイトに設定
- SCxRFC[7:6] = 11 : 受信 FIFO のクリアと割り込み発生条件の設定

上記の FIFO 構成の設定後、SCxMOD0<RXE> に 1 を書き込むとデータ受信を開始します。受信シフトレジスタ、受信バッファ、受信 FIFO すべてにデータが格納されると<RXE>を自動クリアして受信を終了します。

上記の設定で、fill レベル到達後の継続受信を許可にしておくと、FIFO のデータを読み出すことにより継続して受信動作を行うことができます。

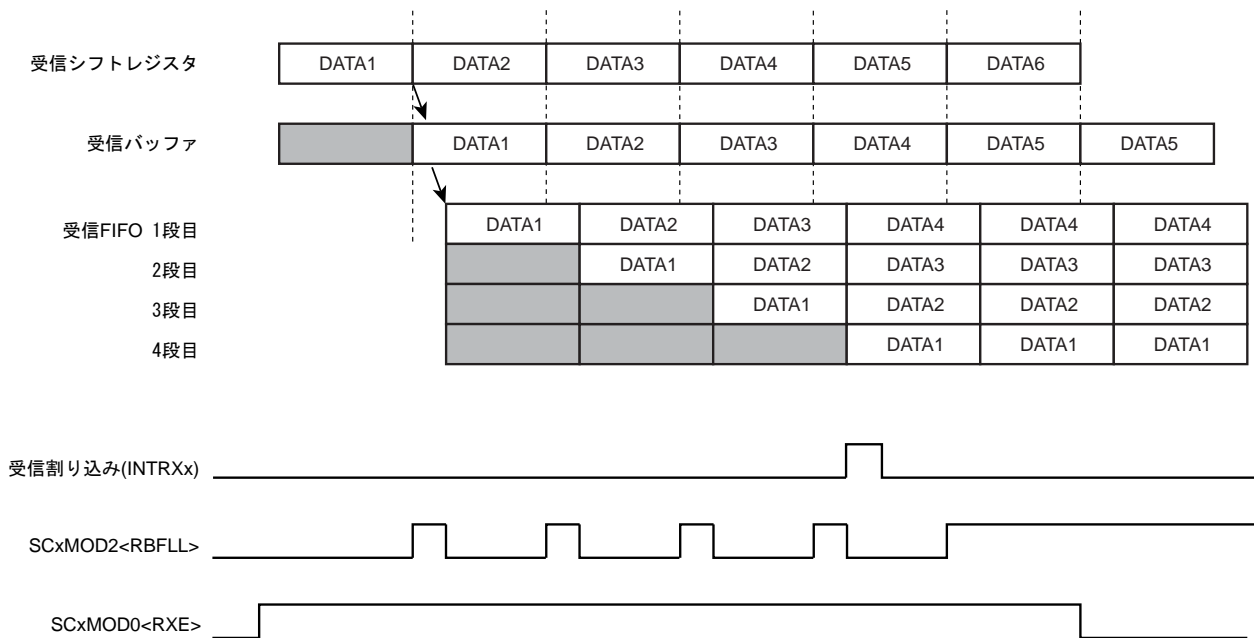


図 10-5 受信 FIFO の動作

### 10.11.3.3 IO インタフェースモード、SCLK 出力での受信

IO インタフェースモードで SCLK 出力設定の場合、使用可能な受信バッファ/FIFO にすべてデータが格納されると SCLK 出力が停止されます。このため、このモードではオーバランエラーフラグは意味を持ちません。

SCLK 出力の停止/再開のタイミングはバッファ/FIFO の使用状況によって変わります。

#### (1) シングルバッファの場合

1 データ受信後に SCLK 出力を停止します。このため、通信相手と 1 データごとのハンドシェイクが可能です。バッファからデータが読み出されると SCLK 出力を再開します。

#### (2) ダブルバッファの場合

受信シフトレジスタ、受信バッファともにデータが格納されると、SCLK 出力を停止します。1 データが読み出されると SCLK 出力を再開します。

#### (3) FIFO の場合

受信シフトレジスタ、受信バッファ、FIFO すべてにデータが格納されると SCLK 出力を停止します。1 データ読み出されると受信バッファから FIFO へ、受信シフトレジスタから受信バッファへデータが転送され、SCLK 出力を再開します。

また、SCxFCNF<RXTXCNT>がセットされていると、SCLK 停止とともに SCxMOD0<RXE>ビットがクリアされ受信動作を停止します。

### 10.11.3.4 受信データの読み出し

FIFO の有効/無効にかかわらず受信バッファ (SCxBUF) からデータを読み出します。

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 フラグ SCxMOD2<RBFL>は"0"にクリアされます。受信バッファを読み出す前でも、次の受信データは受信シフトレジスタに格納することができます。8 ビット UART モードでパリティ付加の場合と 9 ビット UART モードの場合、最上位ビットは SCxCR <RB8> に格納されます。

受信 FIFO が有効な場合、FIFO に格納できるデータは最大 8 ビットですので、9 ビット UART モードは使用できません。8 ビット UART モードでパリティ付加の場合、パリティビットは失われますがエラー判定は行われ、結果が SCxCR<PERR>に格納されます。

### 10.11.3.5 ウェイクアップ機能

9 ビット UART モードの場合、ウェイクアップ機能 SCxMOD0<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す。SCxCR<RB8>="1"のときのみ、割り込み INTRXx を発生させることができます。

### 10.11.3.6 オーバランエラー

受信 FIFO が有効にされていない場合は、受信シフトレジスタに次のデータが全ビット受信される前に受信バッファ(SCxBUF)を読み出さなければオーバランエラーとなります。オーバランエラーが発生した場合、受信バッファおよび SCxCR <RB8> の内容は保存されていますが、受信シフトレジスタ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 のオーバランが発生してオーバラン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。

IO インタフェースモードの SCLK 出力の設定では、クロック出力が自動的に停止するためこのフラグは意味を持ちません。

注) IO インタフェース SCLK 出力モードから他のモードへ切り替える際には、SCxCR レジスタを読み出し、オーバランフラグをクリアしてください。

## 10.12 送信

### 10.12.1 送信カウンタ

送信カウンタは4ビットのバイナリカウンタで、受信カウンタ同様 SIOCLK でカウントされます。

UART モードでは、16 クロックごとに送信クロック (TXDCLK) を生成します。

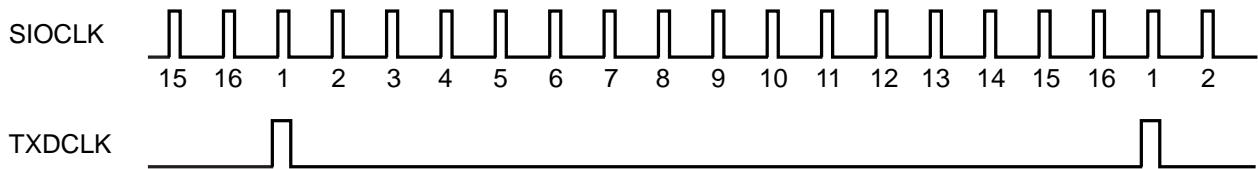


図 10-6 UART モード送信クロックの生成

### 10.12.2 送信制御部

#### 10.12.2.1 IO インタフェースモードの場合

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

SCxCR<IOC> = "1" の SCLK 入力モードのときは、SCxCR<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 TXD 端子へ出力します。

#### 10.12.2.2 UART モードの場合

送信バッファに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を生成します。

## 10.12.3 送信動作

### 10.12.3.1 送信バッファの動作

ダブルバッファ無効の場合、送信データの書き込みは送信シフトレジスタに対して行われ、送信が終了すると送信割り込み INTTXx が発生します。

ダブルバッファ有効の場合(送信 FIFO が有効な場合も含む)、送信バッファへ書き込まれたデータは送信シフトレジスタに転送されます。同時に送信割り込み INTTXx が発生し、送信バッファエンプティフラグ(SCxMOD2<TBEMP>)がセットされ、次のデータが書き込み可能であることを示します。次のデータを送信バッファに書き込むと<TBEMP>は"0"にクリアされます。

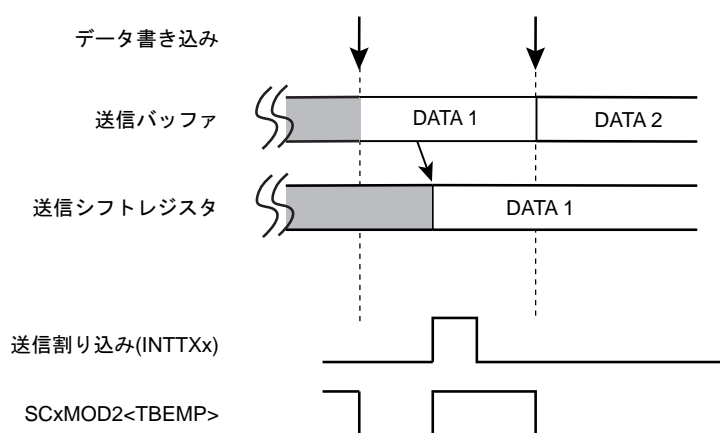


図 10-7 送信バッファの動作(ダブルバッファ有効時)

### 10.12.3.2 送信 FIFO の動作

FIFO が許可されている場合、送信バッファと FIFO で最大 5 バイトのデータを格納することができます。送信を許可すると、送信バッファから送信シフトレジスタにデータが転送されて送信を開始するとともに、FIFO にデータが存在する場合はただちに送信バッファへ移され、<TBEMP>フラグは"0"にクリアされます。

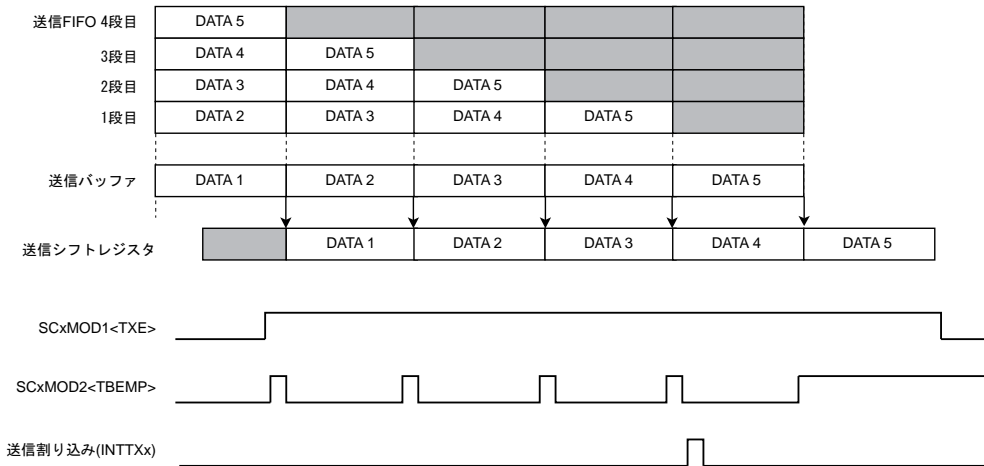
注) 送信 FIFO バッファ使用時は、SIO の転送モード設定(半二重/全二重)、FIFO 許可(SCxFCNF <CNFG>="1")の後、必ず送信 FIFO クリアを実行して下さい。

以下に、4 バイトのデータを半二重送信する場合の設定と動作を示します。

SCxMOD1[6:5] = 10	: 転送モードを半二重送信に設定
SCxFCNF[4:0] = 11011	: FIFO が空になると送信を自動的に禁止 受信 FIFO の使用バイト数は割り込み発生 fill レベルに同じ
SCxTFC[1:0] = 00	: 割り込み発生時の fill レベル を 0 に設定
SCxTFC[7:6] = 11	: 送信 FIFO のクリアと割り込み発生条件の設定

上記の設定後、送信バッファ/FIFO に 5 バイト分の送信データを書き込み、SCxMOD1<TXE> ビットを 1 に設定すること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バッファに移されると送信 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送信を終了します。

上記の設定で、送信の自動禁止としなければ送信データを書き込むことにより継続して送信を行うことができます。



### 10.12.3.3 IO インタフェースモード、SCLK 出力での送信

IO インタフェースモードで SCLK 出力の場合、設定されたデータがすべて送信されると SCLK 出力は自動的に停止します。このため、このモードではアンダーランエラーは発生しません。

バッファ/FIFO 使用状況によって SCLK 出力の停止/再開のタイミングが変わります。

#### (1) シングルバッファの場合

1 データ送信後に SCLK 出力を停止します。このため、通信相手と 1 データごとのハンドシェイクが可能です。バッファに次のデータが書き込まれると SCLK 出力を再開します。

#### (2) ダブルバッファの場合

送信シフトレジスタと送信バッファのデータがすべて送信されると SCLK 出力を停止します。バッファに次のデータが書き込まれると SCLK 出力を再開します。

#### (3) FIFO の場合

送信シフトレジスタ、送信バッファ、FIFO すべてのデータの送信が終了すると SCLK 出力を停止します。次のデータが書き込まれると SCLK 出力を再開します。

また、SCxFCNF<RXTXCNT>がセットされていると、SCLK 停止とともに SCxMOD0<TXE>ビットがクリアされ送信動作を停止します。



#### 10.12.3.4 アンダーランエラー

IO インタフェースモードの SCLK 入力時に送信 FIFO が有効にされていない場合、送信シフトレジスタのデータの送信が終了し、次の転送クロックが入力される前に送信バッファヘ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 SCxCR<PERR>に"1"がセットされます。

IO インタフェースモードの SCLK 出力の設定では、クロック出力が自動的に停止するためこのフラグは意味を持ちません。

注) IO インタフェース SCLK 出力モードから他のモードへ切り替える際には、SCxCR レジスタを読み出し、アンダーランフラグをクリアしてください。

## 10.13 ハンドシェイク機能

ハンドシェイク機能は CTS (Clear to send)端子を用いて 1 データ単位での送信を行う機能で、この機能を使うことでオーバランエラーの発生を防ぐことができます。ハンドシェイク機能は SCxMOD0 <CTSE> によってイネーブル/ディセーブルを設定できます。

$\overline{\text{CTS}}$  端子が "High" 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 "Low" レベルに戻るまで送信を停止します。ただし、INTTXx 割り込みは通常のタイミングで発生しますので、次の送信データを送信バッファに書き込み、送信待機状態にすることができます。

- 注) ① 送信中に  $\overline{\text{CTS}}$  信号を立ち上げた場合、送信終了後に停止します。  
 ② CTS 信号立下り後の最初の TXDCLK クロックから送信を開始します。

なお、RTS 端子はありませんが、任意のポートの 1 ビットを RTS 機能に割り当て、受信終了時に (受信割り込みルーチン内で)このポートを "High" レベルにし、送信側に送信の一時停止を要求することで容易にハンドシェイク機能を構築できます。

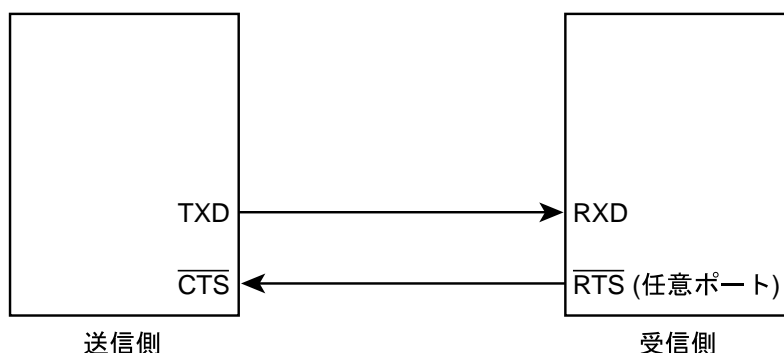


図 10-8 ハンドシェイク機能接続

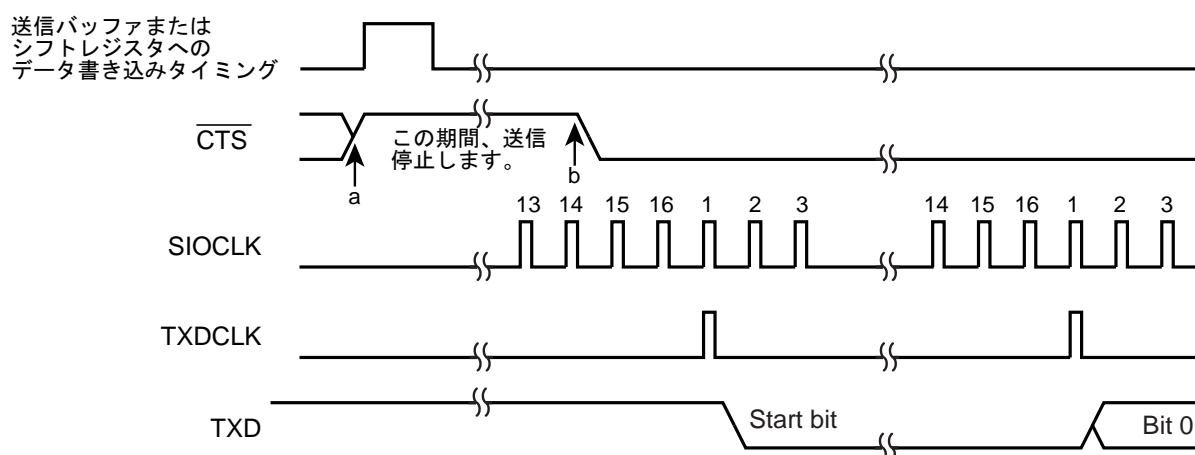


図 10-9  $\overline{\text{CTS}}$  信号のタイミング

## 10.14 割り込み/エラー発生タイミング

### 10.14.1 受信割り込み

受信動作のデータの流れと読み出しの経路を図 10-10 に示します。

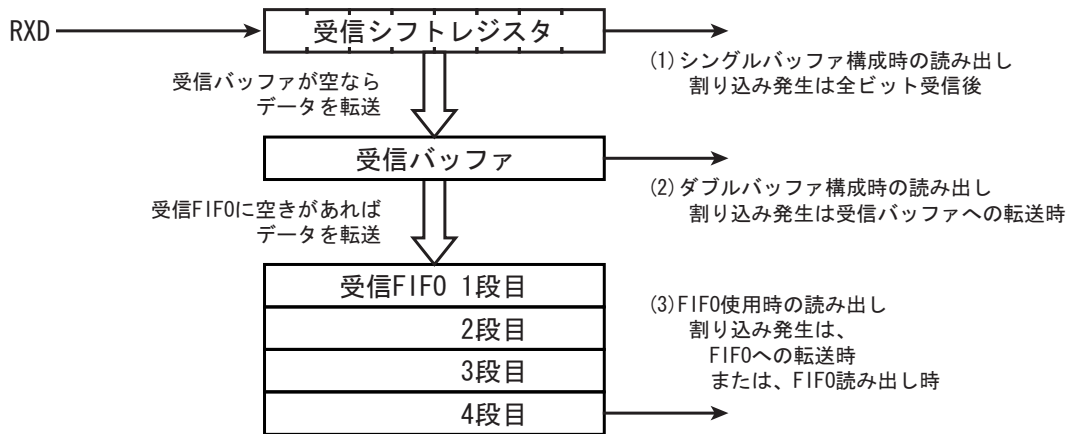


図 10-10 受信バッファ/FIFO 構成図

#### 10.14.1.1 シングルバッファ/ダブルバッファ構成の場合

受信割り込みは、転送モードとバッファ構成により以下のタイミングで発生します。

バッファ構成	UART モード	IO インタフェースモード
シングルバッファ	-	・最終ビットの SCLK 立ち上がり/立ち下がり直後 (立ち上がり/立ち下がりは、SCxCR<SCLKS>設定による)
ダブルバッファ	・第1ストップビットの中央付近	・最終ビットの SCLK 立ち上がり/立ち下がり直後 (立ち上がり/立ち下がりは、SCxCR<SCLKS>設定による) ・バッファのリードによる、シフトレジスタからバッファへのデータ転送時

注) オーバーランエラー成立時は割り込みは発生しません。

#### 10.14.1.2 FIFO 使用の場合

FIFO 使用の場合の受信割り込みは、以下の動作で SCxRFC<RFIS>の設定で決まる条件を満たした場合に発生します。

- ・ 1 フレームの全ビットの受信終了
- ・ FIFO をリードしたとき

発生条件は、SC x RFC<RFIS>の設定により表 10-12 のようになります。

表 10-12 FIFO 使用時の受信割り込み発生条件

SCxRFC<RFIS>	割り込み発生条件
"0"	FIFO fill レベル=割り込み発生 fill レベルが成立したとき
"1"	FIFO fill レベル≥割り込み発生 fill レベルとなったとき

### 10.14.2 送信割り込み

送信動作のデータの流れと読み出しの経路を示します。

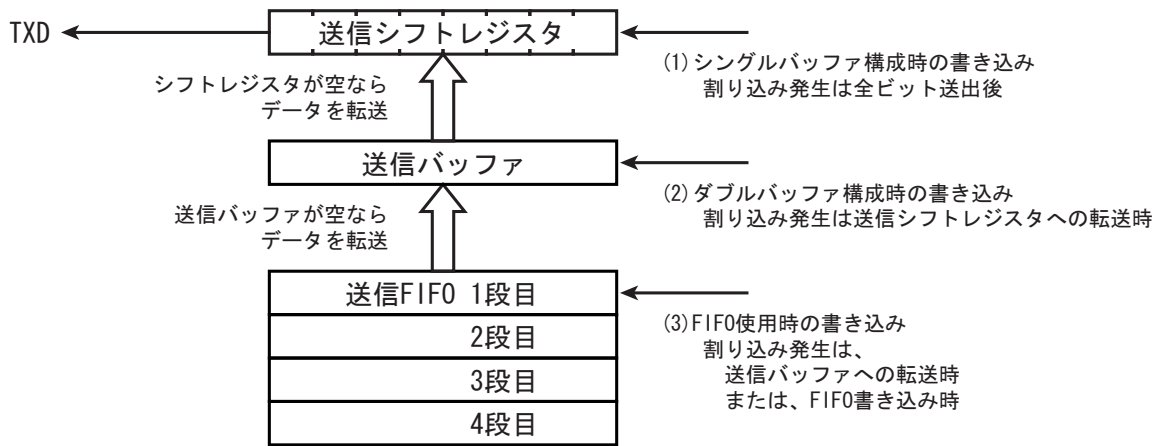


図 10-11 送信バッファ/FIFO 構成図

#### 10.14.2.1 シングルバッファ/ダブルバッファ構成の場合

送信割り込みは、転送モードとバッファ構成により以下のタイミングで発生します。

バッファ構成	UART モード	IO インタフェースモード
シングルバッファ	ストップビット送出の直前	最終ビットの SCLK 立ち上がり/立ち下がり直後 (立ち上がり/立ち下がりは、SCxCR<SCLKS>設定による)
ダブルバッファ	送信バッファから送信シフトレジスタへのデータ転送時	

注) ダブルバッファ有効の場合、送信バッファ書き込みによってバッファからシフトレジスタへデータが転送された場合も割り込みが発生します。

#### 10.14.2.2 FIFO 使用の場合

FIFO 使用の場合の送信割り込みは、以下の動作で SCxTFC<TFIS>の設定で決まる条件を満たした場合に発生します。

- ・ 1 フレームの全ビットの送信終了
- ・ FIFO への書き込み

発生条件は、SC x TFC<TFIS>の設定により表 10-13 のようになります。

表 10-13 FIFO 使用時の送信割り込み発生条件

SCxTFC<TFIS>	割り込み発生条件
"0"	FIFO fill レベル=割り込み発生 fill レベルが成立したとき
"1"	FIFO fill レベル<=割り込み発生 fill レベルとなったとき

### 10.14.3 エラー発生

#### 10.14.3.1 UART モード

モード	9 ビット	7 ビット 8 ビット 7 ビット+パリティ 8 ビット+パリティ
フレーミングエラー オーバランエラー	ストップビットの中央付近	
パリティエラー	-	パリティビットの中央付近

#### 10.14.3.2 IO インタフェースモード

オーバランエラー	最終ビットの SCLK 立ち上がり/立ち下がり直後 (立ち上がり/立ち下がりは、SCxCR<SCLKS>設定による)
アンダーランエラー	次回 SCLK の立ち上がり/立ち下がり直後 (立ち上がり/立ち下がりは、SCxCR<SCLKS>設定による)

注) SCLK 出力モードではオーバランエラー、アンダーランエラーフラグは意味を持ちません。

## 10.15 ソフトリセット

SCxMOD2<SWRST[1:0]>に"10" → "01"の順で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、SCxMOD0<RXE>、SCxMOD1<TXE>、SCxMOD2<TBEMP><RBFL><TXRUN>、SCxCR<OERR><PERR><FERR>が初期化され、送受信回路と FIFO は初期状態になります。その他の状態は保持されます。

## 10.16 モード別動作説明

### 10.16.1 モード 0 (IO インタフェースモード)

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 動作および送信 FIFO 動作の項を参照してください。

#### 10.16.1.1 送信

##### (1) SCLK 出力モード

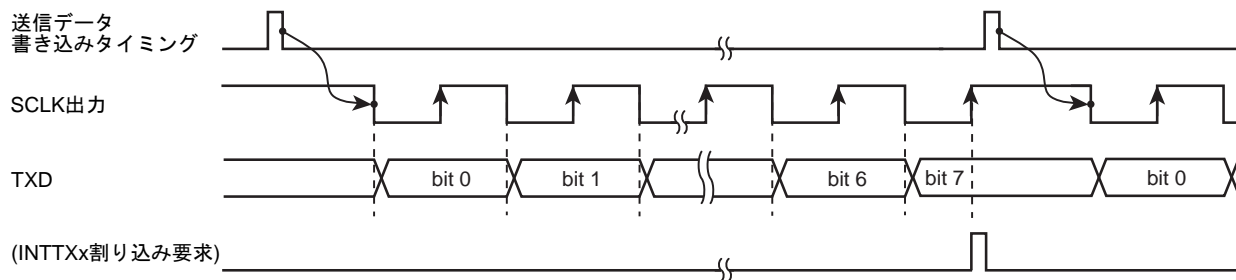
- ・ ダブルバッファ不許可(SCxMOD2<WBUF> = "0")の場合

送信バッファにデータを書き込むたびに、データが TXD 端子から、クロックが SCLK 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(INTTXx)が発生します。

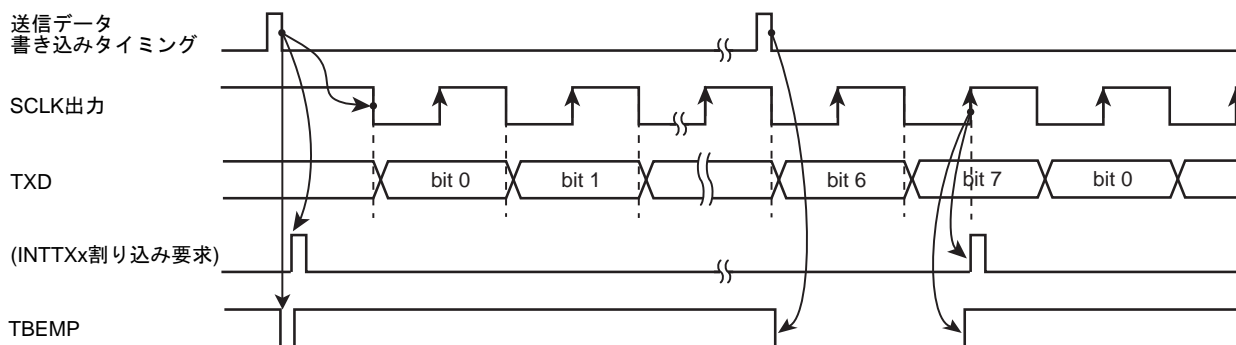
- ・ ダブルバッファ許可(SCxMOD2<WBUF> = "1")の場合

送信停止の状態です送信バッファにデータを書き込んだとき、または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よりシフトレジスタ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 SCxMOD2 <TBEMP>が"1"にセットされ割り込み(INTTXx)が発生します。

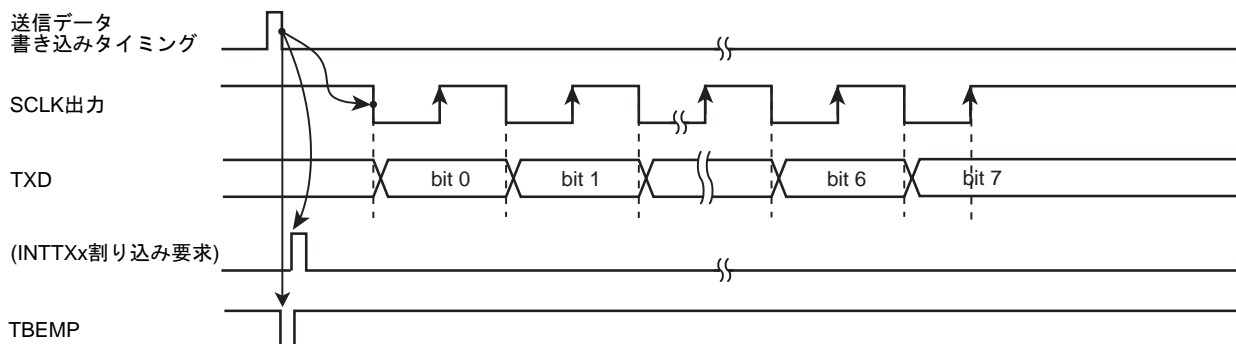
シフトレジスタのデータ送出終了時に送信バッファにデータが存在しない場合は、割り込み(INTTXx)が発生せず、SCLK 出力も停止します。



<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(バッファにデータがない場合)

図 10-12 IO インタフェースモード送信動作(SCLK 出力モード)

## (2) SCLK 入力モード

- ・ ダブルバッファ不許可(SCxMOD2<WBUF> = "0")の場合

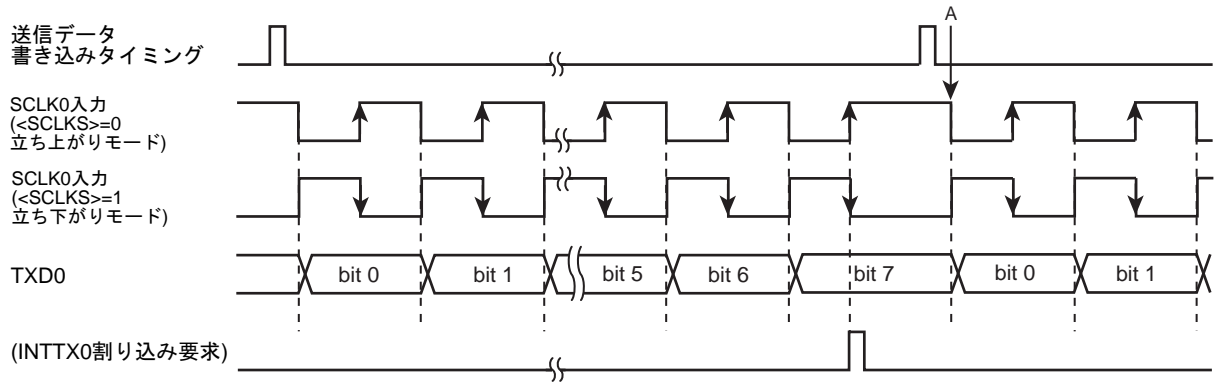
送信バッファにデータが書き込まれている状態で SCLK が入力されると、8 ビットのデータが TXD 端子より出力されます。データがすべて出力されると割り込み INTTXx が発生します。次の送信データは図 10-13 に示す A 点までに書き込んでください。

- ・ ダブルバッファ許可(SCxMOD2<WBUF> = "1")の場合

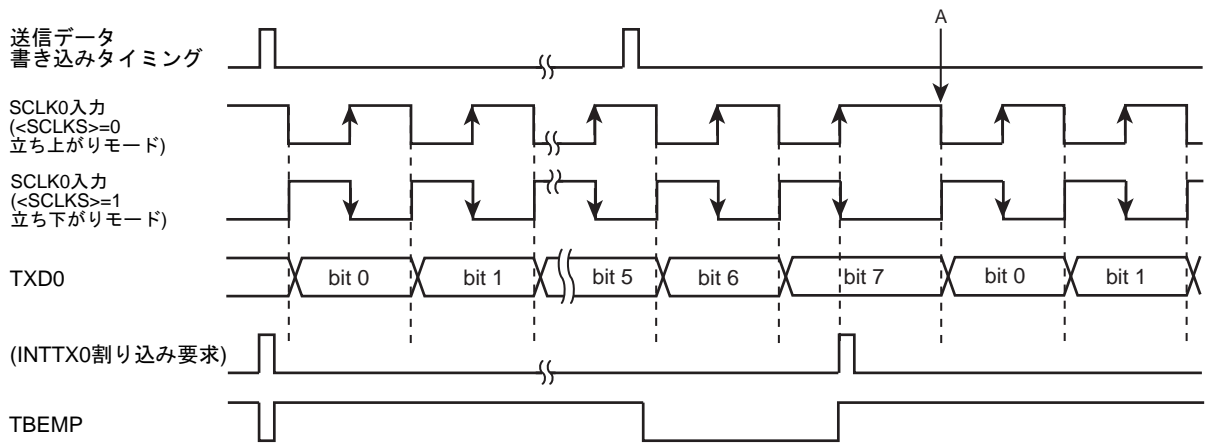
SCLK が入力される前に送信バッファにデータを書き込んだとき、または送信シフトレジスタのデータ送出が終了したときに送信バッファのデータがシフトレジスタへ移されます。これと同時に送信バッファ empty フラグ SCxMOD2<TBEMP> が"1"にセットされ、割り込み(INTTXx)が発生します。

送信バッファにデータが書き込まれていない状態で、SCLK が入力された場合、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(0xFF)を送出します。

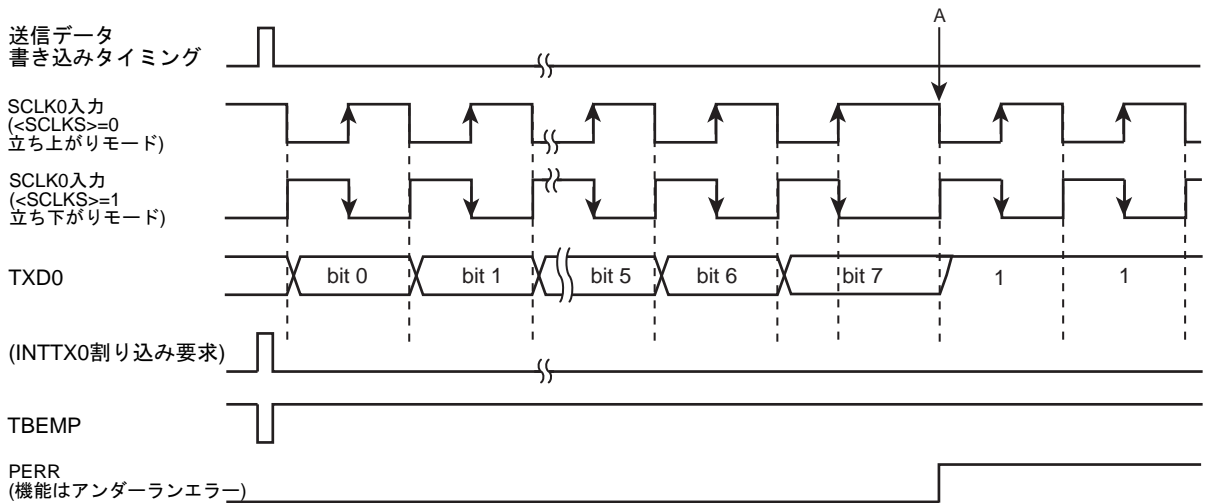




<WBUF> = "0" (ダブルバッファ不許可)の場合



<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 10-13 IO インタフェースモード送信動作(SCLK 入力モード)

### 10.16.1.2 受信

#### (1) SCLK 出力モード

受信許可ビット SCxMOD0<RXE>を"1"にセットすることで SCLK 出力が開始されます。

- ・ ダブルバッファ不許可(SCxMOD2<WBUF> = "0")の場合

受信データが読み出されるごとに、SCLK 端子よりクロックが出力され次のデータがシフトレジスタに格納されます。8 ビットデータが受信されると、割り込み INTRXx が発生します。

- ・ ダブルバッファ許可(SCxMOD2<WBUF> = "1")の場合

シフトレジスタに格納されたデータは受信バッファに移され、続けて次のフレームを受信することができます。シフトレジスタから受信バッファにデータが移されると、受信バッファ full フラグ SCxMOD2<RBFL>が"1"にセットされ、割り込み INTRXx が発生します。

受信バッファにデータが存在する状態で、次の 8 ビット分のデータを受信完了する前に受信バッファのデータが読み出されない場合、割り込み INTRXx は発生せず、SCLK 出力は停止します。この状態で受信バッファのデータを読み出すと、シフトレジスタのデータを受信バッファに移し、割り込み INTRXx を発生して受信を再開します。

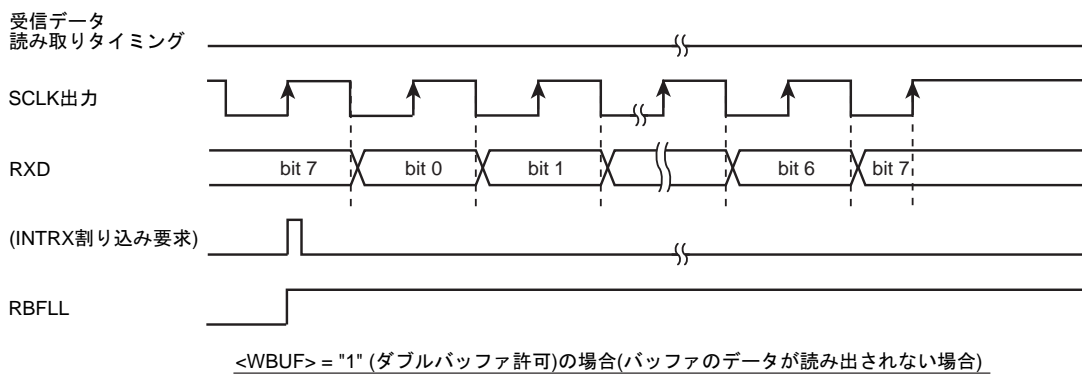
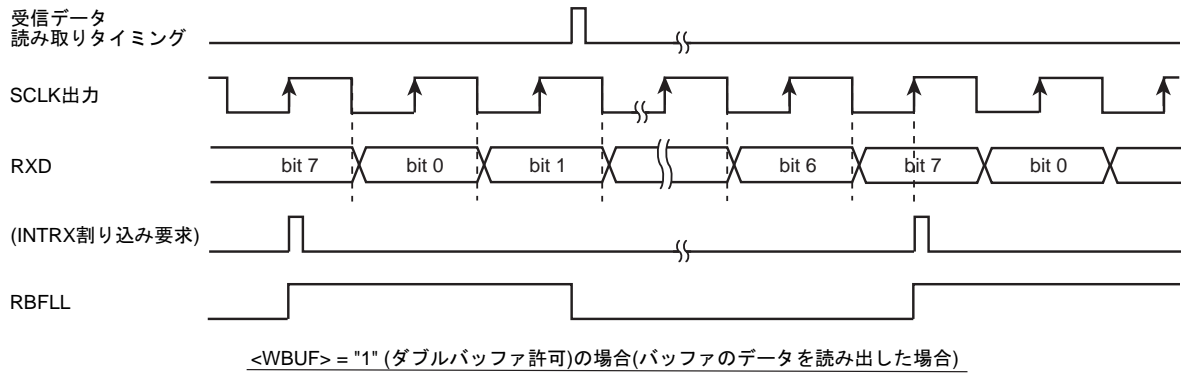
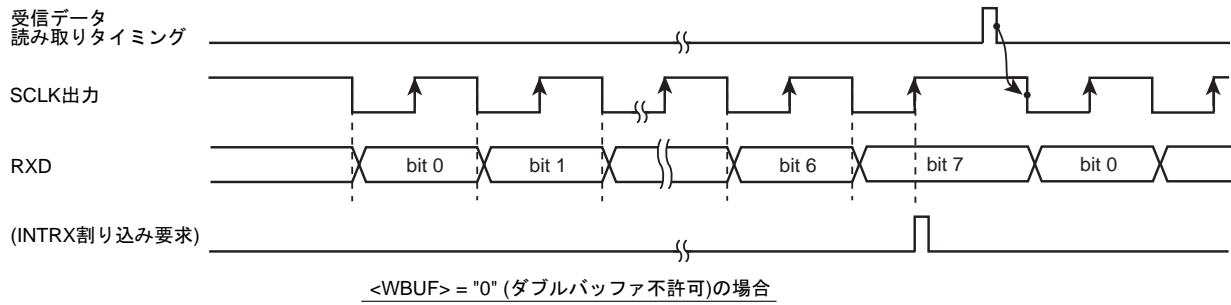


図 10-14 IO インタフェースモード受信動作(SCLK 出力モード)

(2) 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シフトレジスタから受信バッファに移され、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へ移されるごとに受信割り込み INTRXx が発生します。

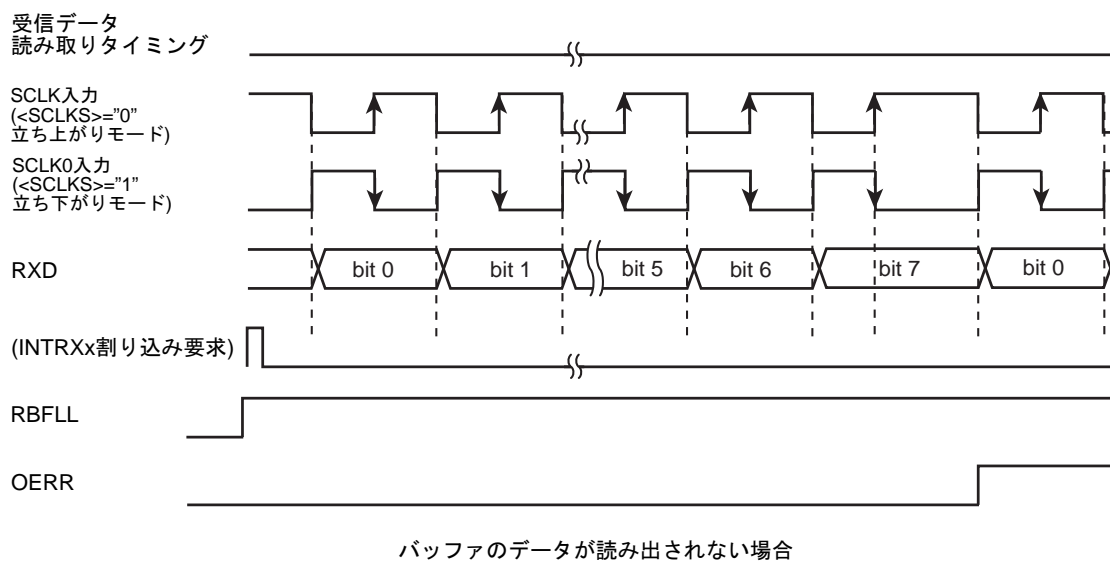
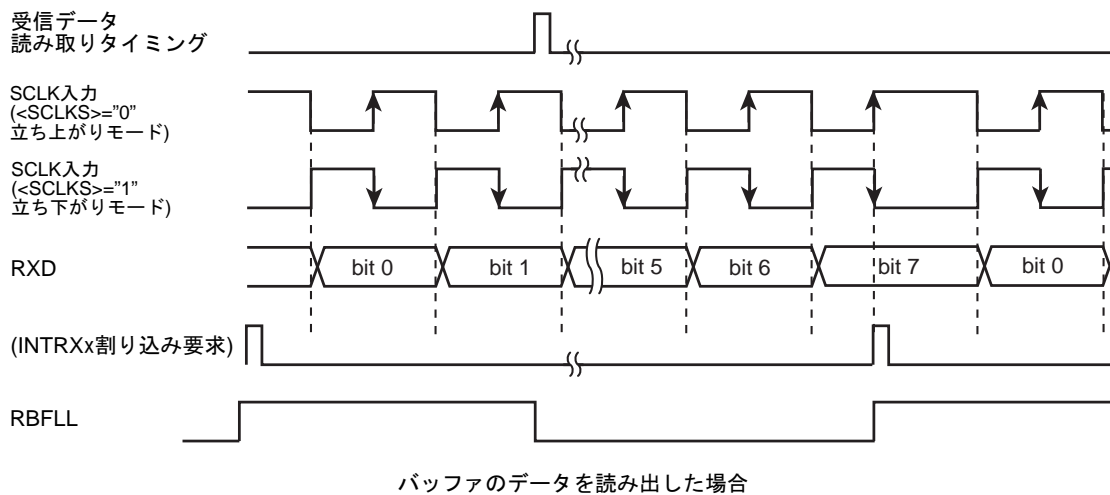


図 10-15 IO インタフェースモード受信動作(SCLK 入力モード)

## 10.16.1.3 送受信(全二重)

## (1) SCLK 出力モード

- ・ ダブルバッファ不許可(SCxMOD2<WBUF> = "0")の場合

送信バッファにデータを書き込むと SCLK 出力を開始します。

SCLK の出力により 8 ビットデータが受信シフトレジスタにシフトインされ、受信割り込み(INTRXx)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、TXD 端子より出力され、すべてのデータが送信されると送信割り込み(INTTXx)が発生します。この状態で SCLK の出力は停止します。

受信バッファの読み出しと送信バッファへのデータ書き込みを行う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。

- ・ ダブルバッファ許可(SCxMOD2<WBUF> = "1")の場合

送信バッファにデータを書き込むと SCLK が出力を開始します。

8 ビットデータが受信シフトレジスタにシフトインされるとデータは受信バッファに移され、割り込み(INTRXx)が発生します。受信と並行して 8 ビットデータが TXD 端子より出力され、データがすべて出力されると割り込み(INTTXx)が発生して次のデータが送信バッファから送信シフトレジスタに移されます。

この時、送信バッファに移すデータが存在しない(SCxMOD2 <TBEMP> = "1")または受信バッファにデータが存在している(SCxMOD2 <RBFL> = "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。

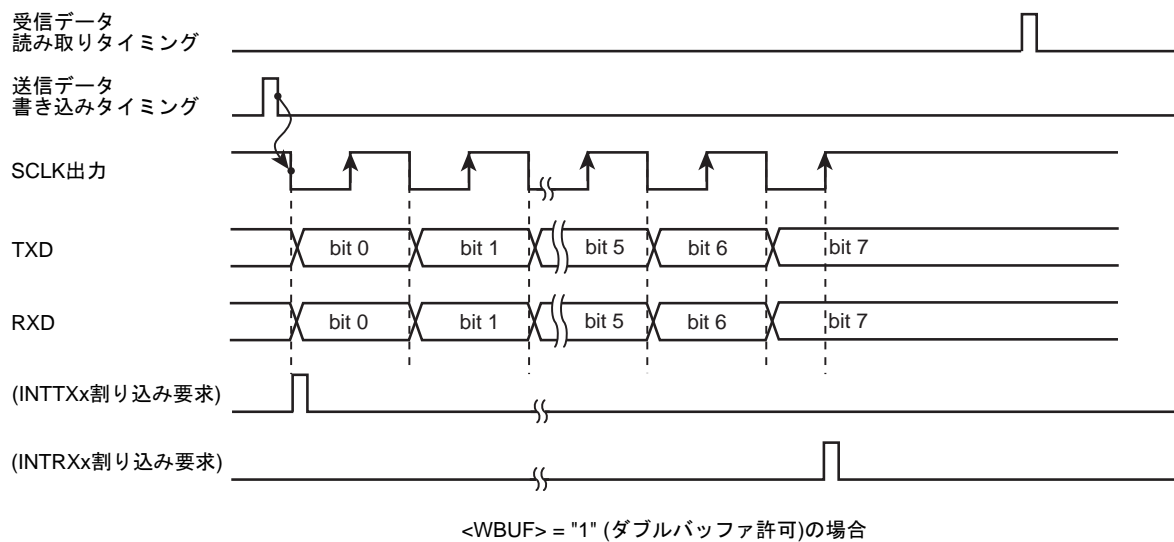
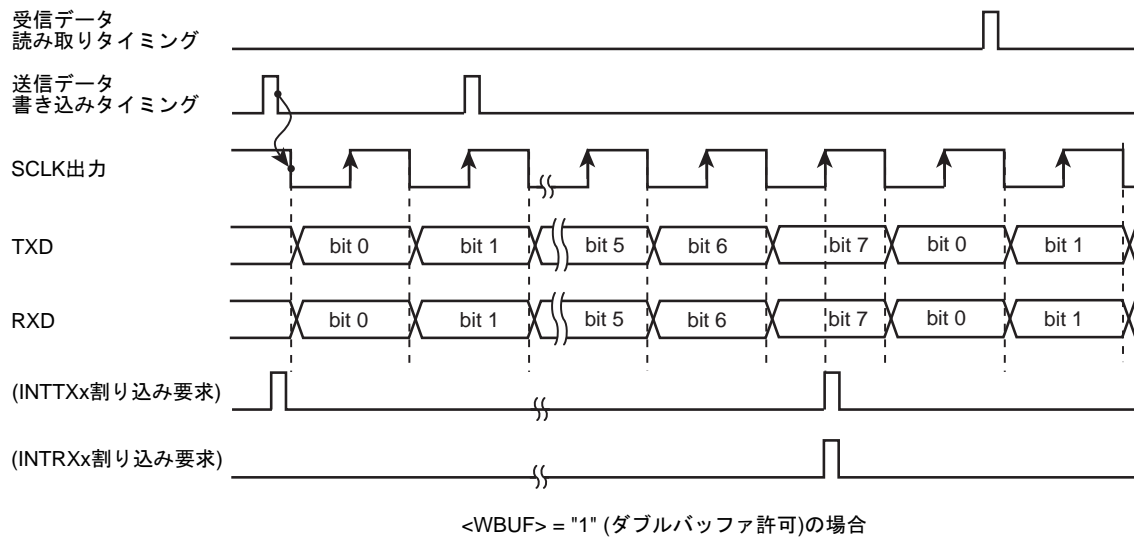
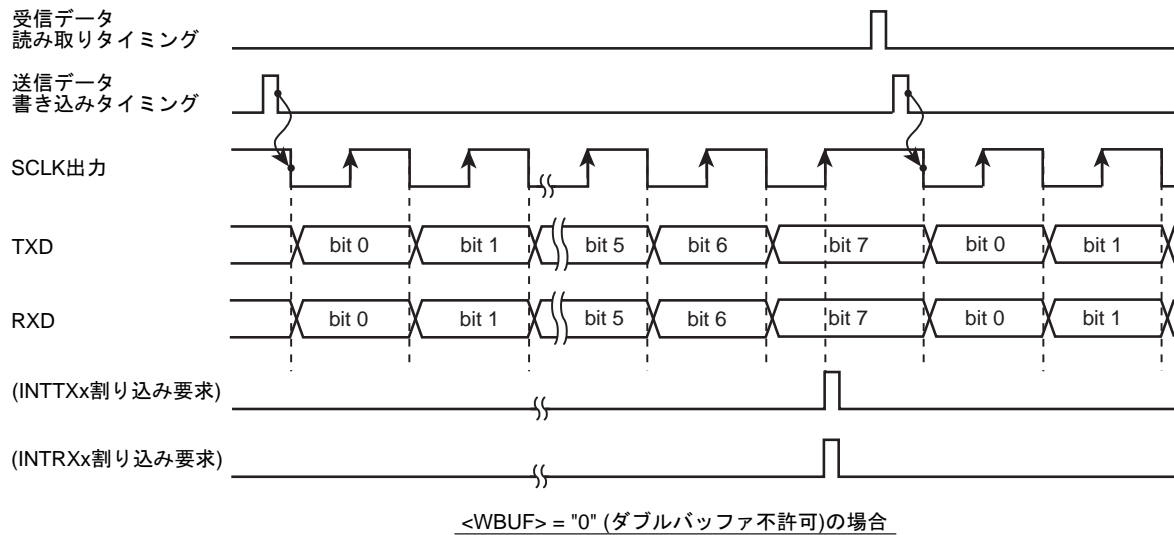


図 10-16 IO インタフェースモード送受信動作(SCLK 出力モード)

## (2) SCLK 入力モード

- ・ 送信ダブルバッファ不許可(SCxMOD2<WBUF> = "0")の場合

受信は SCxMOD2<WBUF>の設定に関わらずダブルバッファが有効になります。

送信バッファにデータが書き込まれている状態で SCLK が入力されると、8 ビットのデータが TXD 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。送信が終了すると割り込み(INTTXx)が発生し、受信が終了すると受信シフトレジスタから受信バッファへデータが移されると同時に割り込み(INTRXx)が発生します。

次のフレームの SCLK が入力される前にデータを送信バッファへ書き込むようにしてください。(図 10-17 に示す A 点までに書き込んでください)。受信データは、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。

- ・ ダブルバッファ許可(SCxMOD2<WBUF> = "1")の場合

送信シフトレジスタのデータの送信が終了すると、送信バッファのデータが送信シフトレジスタへ移されると同時に割り込み(INTRXx)が発生します。平行して受信が行われ、シフトレジスタにデータが揃うと受信バッファへ移され、割り込み(INTRXx)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ようにしてください。(図 10-17 に示す A 点までに書き込んでください)。受信データは、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。

続けて次のフレームの SCLK が入力されると、送信バッファから送信シフトレジスタにデータを移して送信が始まり、並行して受信シフトレジスタでのデータ受信が行なわれます。

フレームの最終ビットの受信までに受信バッファのデータが読み出されていない場合はオーバ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。

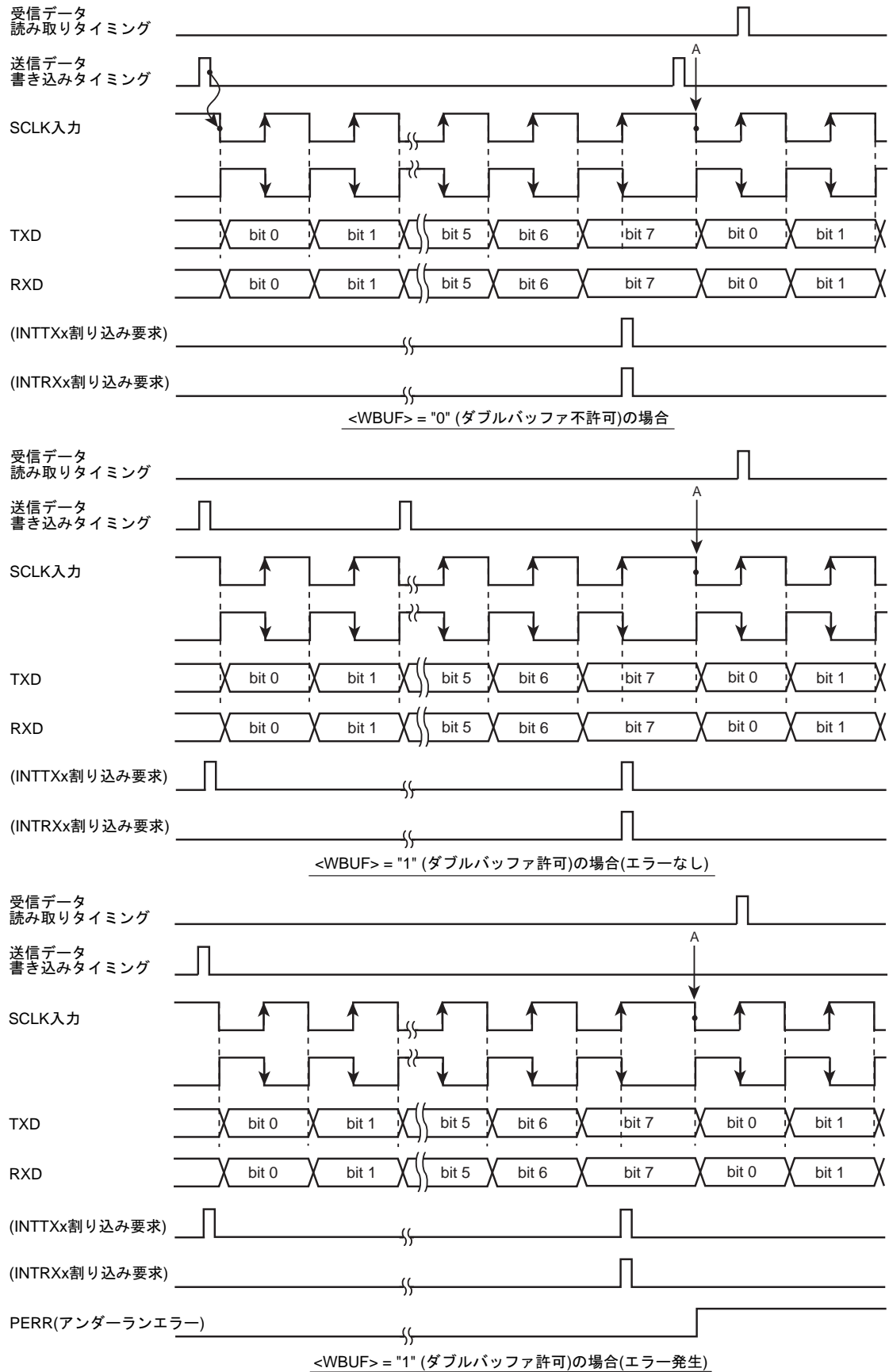


図 10-17 IO インタフェースモード送受信動作(SCLK 入力モード)

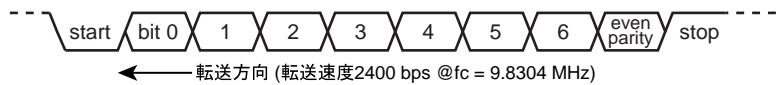


### 10.16.2 モード 1 (7 ビット UART モード)

シリアルモードコントロールレジスタ(SCxMOD0 <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 (SCxCR <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、SCxCR<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 SCxMOD2<SBLEN>で指定することができます。

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。



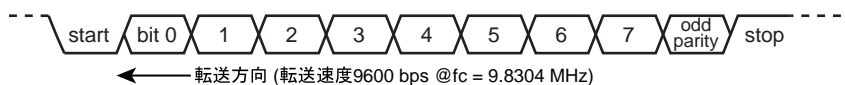
クロック条件	システムクロック:            高速 (fc) 高速クロックギア:            1 倍 (fc) プリスケアラクロック:        fperiph/2 (fperiph = fsys)									
SCxMOD0	←	x	0	-	0	0	1	0	1	7 ビット UART モードに設定
SCxCR	←	x	1	1	x	x	x	0	0	偶数パリティイネーブルに設定
SCxBRCR	←	0	0	1	0	0	1	0	0	2400bps に設定
SCxBUF	←	*	*	*	*	*	*	*	*	送信データを設定

x : don't care - : no change

### 10.16.3 モード 2 (8 ビット UART モード)

SCxMOD0 <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 SCxCR <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、SCxCR <EVEN> で偶数パリティ/奇数パリティの選択も可能です。

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) プリスケアラクロック:        fperiph/2 (fperiph = fsys)	
--------	--	--

	7	6	5	4	3	2	1	0	
SCxMOD0	← x	0	0	0	1	0	0	1	8ビット UART モードに設定
SCxCR	← x	0	1	x	x	x	0	0	奇数パリティネーブルに設定
SCxBRCR	← 0	0	0	1	0	1	0	0	9600bps に設定
SCxMOD0	← -	-	1	-	-	-	-	-	受信許可

x : don't care - : no change

### 10.16.4 モード 3 (9 ビット UART モード)

SCxMOD0 <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(SCxCR<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 SCxMOD0 <TB8> に書き込み、受信の場合 SCxCR <RB8> に格納されます。また、バッファに対する書き込み、読み出しは必ず最上位ビットの方を先に行い、SCxBUF の方を後にします。STOP ビットの長さは SCxMOD2 <SBLEN> で指定することができます。

#### 10.16.4.1 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 SCxMOD0 <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、SCxCR <RB8> = "1" のときのみ割り込み (INTRXx) が発生します。

注) スレーブコントローラの TXD 端子は、必ず 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。

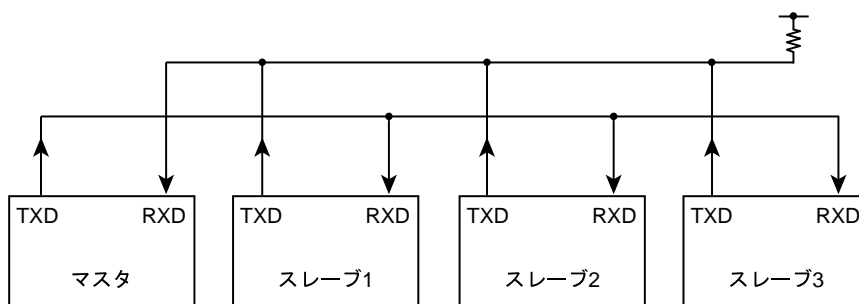


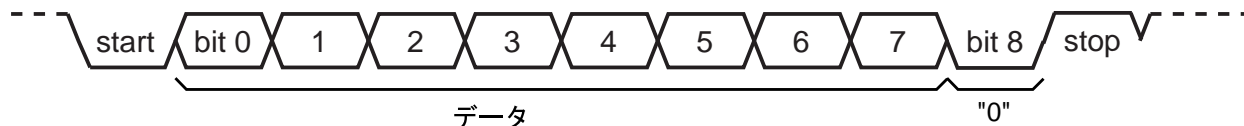
図 10-18 ウェイクアップ機能によるシリアルリンク

## 10.16.4.2 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SCxMOD0<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>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(<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(INTRXx)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。



## 第 11 章 シリアルバスインタフェース(I2C/SIO)

シリアルバスインタフェースを 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。

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、下記のとおりです。

表 11-1 シリアルバスインタフェース使用時のポート設定

チャンネル	動作モード	端子	ポート ファンクション レジスタ設定	ポート 出力コントロール レジスタ設定	ポート 入力コントロール レジスタ設定	ポート オーブンドレイン コントロール レジスタ設定
SBI0	I2C バスモード	SCL0 :PG1 SDA0 :PG0	PGFR1[1:0] = 11	PGCR[1:0] = 11	PGIE[1:0] = 11	PGOD[1:0] = 11
	SIO モード	SCK0 :PG2 SI0 :PG1 SO0 :PG0	PGFR1[2:0] = 111	PGCR[2:0] = 101(SCK0 出力) PGCR[2:0] = 001(SCK0 入力)	PGIE[2:0] = 010(SCK0 出力) PGIE[2:0] = 110(SCK0 入力)	PGOD[2:0] = xxx
SBI1	I2C バスモード	SCL1 :PF5 SDA1 :PF4	PFFR1[5:4] = 11	PFCR[5:4] = 11	PFIE[5:4] = 11	PFOD[5:4] = 11
	SIO モード	SCK1 :PF6 SI1 :PF5 SO1 :PF4	PFFR1[6:4] = 111	PFCR[6:4] = 101(SCK1 出力) PFCR[6:4] = 001(SCK1 入力)	PFIE[6:4] = 010(SCK1 出力) PFIE[6:4] = 110(SCK1 入力)	PFOD[6:4] = xxx
SBI2	I2C バスモード	SCL2 :PG5 SDA2 :PG4	PGFR1[5:4] = 11	PGCR[5:4] = 11	PGIE[5:4] = 11	PGOD[5:4] = 11
	SIO モード	SCK2 :PG6 SI2 :PG5 SO2 :PG4	PGFR1[6:4] = 111	PGCR[6:4] = 101(SCK2 出力) PGCR[6:4] = 001(SCK2 入力)	PGIE[6:4] = 010(SCK2 出力) PGIE[6:4] = 110(SCK2 入力)	PGOD[6:4] = xxx

注) x: Don't care

## 11.1 構成

構成を図 11-1 に示します。

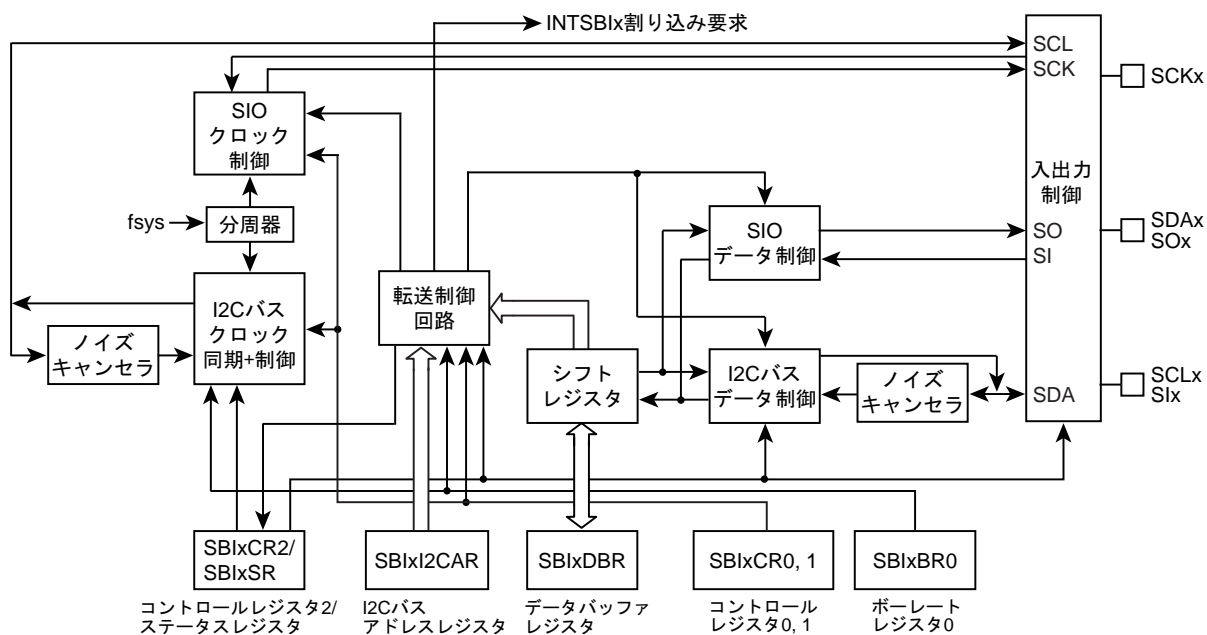


図 11-1 シリアルバスインタフェースブロック図

## 11.2 レジスタ説明

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を行うレジスタとアドレスは以下のとおりです。

下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「11.4 I2C バスモード時のコントロールレジスタ」および「11.7 SIO モード時のコントロールレジスタ」を参照してください。

### 11.2.1 チャネル別レジスタ一覧

各チャネルのレジスタとアドレスを示します。

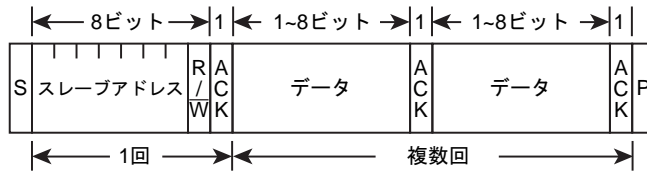
Channel x	Base Address
Channel0	0x4002_0000
Channel1	0x4002_0020
Channel2	0x4002_0040

レジスタ名(x=0~2)		Address(Base+)
コントロールレジスタ 0	SBIxCR0	0x0000
コントロールレジスタ 1	SBIxCR1	0x0004
データバッファレジスタ	SBIxDBR	0x0008
I2C バスアドレスレジスタ	SBIxI2CAR	0x000C
コントロールレジスタ 2	SBIxCR2(ライト時)	0x0010
ステータスレジスタ	SBIxSR(リード時)	
ボーレートレジスタ 0	SBIxBR0	0x0014

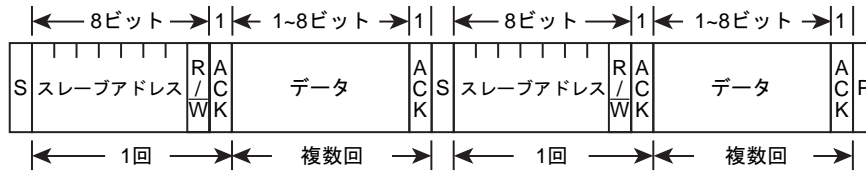
### 11.3 I2C バスモード時のデータフォーマット

I2C バスモード時のデータフォーマットを図 11-2 に示します。

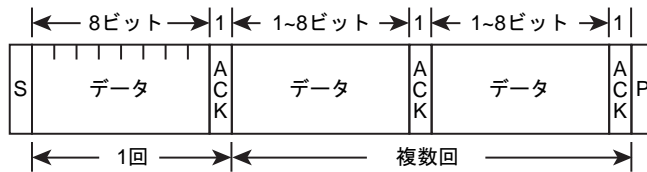
(a) アドレッシングフォーマット



(b) アドレッシングフォーマット (再スタートあり)



(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/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 11-2 I2C バスモード時のデータフォーマット



## 11.4 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 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。

### 11.4.1 SBIXCR0(コントロールレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SBIEN	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	SBIEN	R/W	シリアルバスインタフェース動作 0: 禁止 1: 許可 シリアルバスインタフェースを使用する場合、まずこのビットを許可にしてください。 禁止の場合、SBIXCR0 を除く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。 いったん許可した後に禁止にした場合、各レジスタの設定は保持されます。
6-0	-	R	リードすると"0"が読めます。

11.4.2 SBiXCR1(コントロールレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	BC			ACK	-	SCK2	SCK1	SCK0 / SWRMON
リセット後	0	0	0	0	1	0	0	1(注3)

Bit	Bit Symbol	Type	機能																																																	
31-8	-	R	リードすると"0"が読めます。																																																	
7-5	BC[2:0]	R/W	転送ビット数の選択(注 1) <table border="1" style="margin-left: 20px;"> <thead> <tr> <th rowspan="2">&lt;BC&gt;</th> <th colspan="2">&lt;ACK&gt; = 0 のとき</th> <th colspan="2">&lt;ACK&gt; = 1 のとき</th> </tr> <tr> <th>クロック数</th> <th>データ長</th> <th>クロック数</th> <th>データ長</th> </tr> </thead> <tbody> <tr><td>000</td><td>8</td><td>8</td><td>9</td><td>8</td></tr> <tr><td>001</td><td>1</td><td>1</td><td>2</td><td>1</td></tr> <tr><td>010</td><td>2</td><td>2</td><td>3</td><td>2</td></tr> <tr><td>011</td><td>3</td><td>3</td><td>4</td><td>3</td></tr> <tr><td>100</td><td>4</td><td>4</td><td>5</td><td>4</td></tr> <tr><td>101</td><td>5</td><td>5</td><td>6</td><td>5</td></tr> <tr><td>110</td><td>6</td><td>6</td><td>7</td><td>6</td></tr> <tr><td>111</td><td>7</td><td>7</td><td>8</td><td>7</td></tr> </tbody> </table>	<BC>	<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
<BC>	<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																																																
4	ACK	R/W	マスターモード 0: アクノリッジのためのメントクロックを発生しない 1: アクノリッジのためのメントクロックを発生する ..... スレーブモード 0: アクノリッジのためのメントクロックをカウントしない 1: アクノリッジのためのメントクロックをカウントする																																																	
3	-	R	リードすると"1"が読めます。																																																	
2-1	SCK[2:1]	R/W	内部 SCL 出カクロックの周波数選択<SCK[2:0]>@ライト (注 2)																																																	
0	SCK[0]	W	<table border="1" style="margin-left: 20px;"> <tbody> <tr><td>000</td><td>n = 5</td><td>385 kHz</td></tr> <tr><td>001</td><td>n = 6</td><td>294 kHz</td></tr> <tr><td>010</td><td>n = 7</td><td>200 kHz</td></tr> <tr><td>011</td><td>n = 8</td><td>122 kHz</td></tr> <tr><td>100</td><td>n = 9</td><td>68 kHz</td></tr> <tr><td>101</td><td>n = 10</td><td>36 kHz</td></tr> <tr><td>110</td><td>n = 11</td><td>19 kHz</td></tr> <tr><td>111</td><td></td><td>reserved</td></tr> </tbody> </table> <div style="margin-left: 100px;"> <math display="block">\left. \begin{array}{l} \text{システムクロック: } f_{\text{sys}} \\ \text{クロックギア: } f_c/1 \\ \text{周波数} = \frac{f_{\text{sys}}}{2^n + 72} \text{ [Hz]} \end{array} \right\} \begin{array}{l} (= 40 \text{ MHz}) \end{array}</math> </div>	000	n = 5	385 kHz	001	n = 6	294 kHz	010	n = 7	200 kHz	011	n = 8	122 kHz	100	n = 9	68 kHz	101	n = 10	36 kHz	110	n = 11	19 kHz	111		reserved																									
000	n = 5	385 kHz																																																		
001	n = 6	294 kHz																																																		
010	n = 7	200 kHz																																																		
011	n = 8	122 kHz																																																		
100	n = 9	68 kHz																																																		
101	n = 10	36 kHz																																																		
110	n = 11	19 kHz																																																		
111		reserved																																																		
	SWRMON	R	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中ではない																																																	

- 注 1) SIO モードに切り替える前に<BC[2:0]>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、「11.5.1 シリアルクロック」を参照してください。
- 注 3) <SCK[0]/SWRMON>は、リセット後"1"が読み出されますが、SBIXCR2 レジスタにて SIO モードに設定した場合、<SCK[0]>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。

## 11.4.3 SBIXCR2(コントロールレジスタ 2)

このレジスタをリードすると、SBIXSR レジスタとして機能します。

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	MST	TRX	BB	PIN	SBIM		SWRST	
リセット後	0	0	0	1	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	MST	W	マスタモード/スレーブモードの選択 0: スレーブモード 1: マスタモード
6	TRX	W	送信/受信の選択 0: 受信 1: 送信
5	BB	W	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
4	PIN	W	INTSBIX 割り込み要求解除 0: - 1: 割り込み要求の解除
3-2	SBIM[1:0]	W	シリアルバスインタフェースの動作モード選択(注) 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: SIO モード 10: I2C バスモード 11: Reserved
1-0	SWRST[1:0]	W	ソフトウェアリセットの発生 最初に"10"、次に"01"をライトすると、ソフトウェアリセットが発生します。

注) 通信中はモードを切り替えないでください。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 バスモードへの切り替えは、ポートの状態が"High"になっていることを確認してから行ってください。

11.4.4 SBiXSR(ステータスレジスタ)

このレジスタをライトすると、SBiXCR2として機能します。

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	MST	TRX	BB	PIN	AL	AAS	ADO	LRB
リセット後	0	0	0	1	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	MST	R	マスタ/スレーブ選択状態モニタ 0: スレーブモード 1: マスタモード
6	TRX	R	トランスミッタ/レシーバ選択状態モニタ 0: レシーバ 1: トランスミッタ
5	BB	R	I2C バス状態モニタ 0: バスフリー 1: バスビジー
4	PIN	R	INTSBiX 割り込み要求状態モニタ 0: 割り込みサービス要求中 1: 割り込みサービス要求解除中
3	AL	R	アービトラクションロスト検出モニタ 0: - 1: 検出
2	AAS	R	スレーブアドレス一致検出モニタ 0: - 1: 検出 (ゼネラルコール検出時もセットされます。)
1	ADO	R	ゼネラルコール検出モニタ 0: - 1: 検出
0	LRB	R	最終受信ビットモニタ 0: 最終受信ビット "0" 1: 最終受信ビット "1"

## 11.4.5 SBiXBR0(ポーレートレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	I2SBI	-	-	-	-	-	-
リセット後	1	0	1	1	1	1	1	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	-	R	リードすると"1"が読めます。
6	I2SBI	R/W	IDLE モード時の動作 0: 停止 1: 動作
5-1	-	R	リードすると"1"が読めます。
0	-	R/W	必ず"0"をライトしてください。

11.4.6 SBixDBR(データバッファレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	DB							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	DB[7:0]	R	受信データ
		W	送信データ

- 注 1) 送信データ書き込み時は、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。

## 11.4.7 SBiXl2CAR(I2C バスアドレスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SA							ALS
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-1	SA[6:0]	R/W	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
0	ALS	R/W	アドレス認識モードの指定 0:スレーブアドレスを認識する 1:スレーブアドレスを認識しない(フリーデータフォーマット)

- 注 1) <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 SBiXl2CAR 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。)

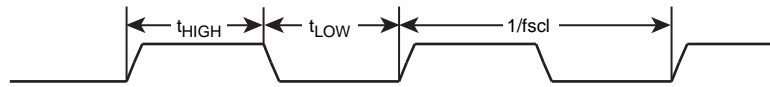


## 11.5 I2C バスモード時の制御

### 11.5.1 シリアルクロック

#### 11.5.1.1 クロックソース

SBIxCR1 <SCK[2:0]>で、マスタモード時に SCLx 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 11-3 クロックソース

注) 通信規格上、標準モード/高速モードの最高速は 100 kHz/400 kHz です。内部 SCL クロックの周波数の設定は、使用される fsys と上記計算式にて設定されますのでご注意ください。

#### 11.5.1.2 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"Low"レベルに引いたマスタが、"High"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、"High"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。

シリアルバスインタフェース回路は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。

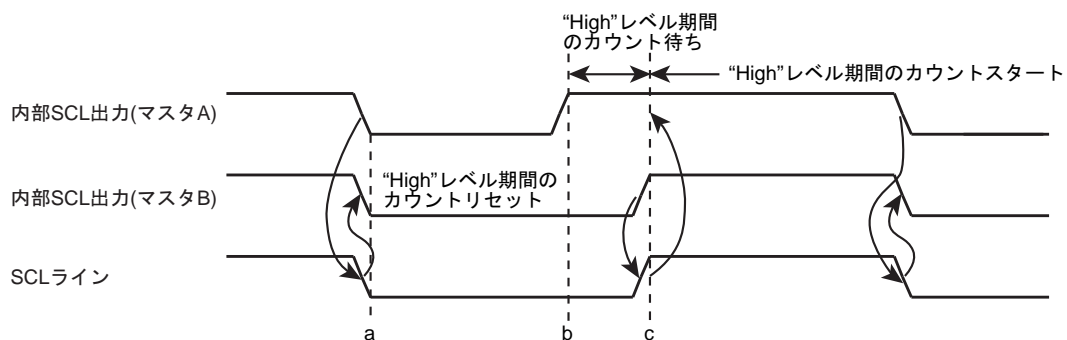


図 11-4 クロック同期化の例

a 点でマスタ A が内部 SCL 出力を"Low"レベルに引くことで、バスの SCL ラインは"Low"レベルになります。マスタ B はこれを検出し、マスタ B の"High"レベル期間のカウントをリセットし、内部 SCL 出力を"Low"レベルに引きます。

b 点でマスタ A は"Low"レベル期間のカウントを終わり、内部 SCL 出力を"High"レベルにします。しかし、マスタ B がバスの SCL ラインを"Low"レベルに保持し続けているので、マスタ A は"High"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"High"レベルにし、バスの SCL ラインが"High"レベルになったことを検出後、"High"レベル期間のカウントを始めます。その後、"High"レベル期間のカウントを終了したマスタ A が SCL 端子を"Low"に引くことでバスの SCL ラインは"Low"レベルになります。

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"High"レベル期間をもつマスタと最も長い"Low"レベル期間をもつマスタによって決定されます。

### 11.5.2 アクノリッジメントモードの指定

SBIxCR1<ACK>を"1"に設定するとアクノリッジメントモードとして動作します。マスタモードのときには、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。スレーブモードのときはアクノリッジ信号のためのクロックをカウントします。トランスミッタモードのときには、アクノリッジのためのクロック期間中 SDAx 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はクロック期間中 SDAx 端子を"Low"レベルに引き、更に、スレーブモードのときにゼネラルコールを受信した場合にもアクノリッジのためのクロック期間中、SDAx 端子を"Low"レベルに引き、アクノリッジ信号を発生します。

<ACK>を"0"に設定すると、非アクノリッジメントモードとして動作し、マスタモード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。スレーブモードのときはアクノリッジ信号のためのクロックをカウントしません。

### 11.5.3 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"にクリアされるため、スレーブアドレス、方向ビットの転送は必ず 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。

### 11.5.4 スレーブアドレスとアドレス認識モードの設定

スレーブアドレスを認識するアドレッシングフォーマットで動作させるときは、SBIxI2CAR<ALS>に"0"を設定し、SBIxI2CAR <SA[6:0]>にスレーブアドレスを設定します。

スレーブアドレスを認識しないフリーデータフォーマットで使用するときには<ALS>を"1"に設定します。なお、フリーデータフォーマットで使用情况の場合、スレーブアドレスと方向ビットの認識は行われず、スタートコンディション直後からデータとして扱われます。

### 11.5.5 動作モード

SBIxCR2<SBIM[1:0]>で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、シリアルバスインタフェース端子の状態が"High"になっていることを確認後、<SBIM[1:0]>を"10"に設定してください。また、ポートモードへの切り替えは、バスフリーであることを確認してから行ってください。

### 11.5.6 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 SBIxI2CAR にセットした値と同じとき
- ・ 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

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/W)が"1"の場合、<TRX>は"1"にセットされ、"0"の場合、<TRX>は"0"にクリアされます。

マスタモード時は、スレーブデバイスからアクノ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。アクノ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"にクリアされます。

フリーデータフォーマットで使用する場合、<TRX>はハードウェアによって変化することはありません。

### 11.5.7 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"にクリアされます。

### 11.5.8 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、あらかじめデータバッファレジスタに書き込んだスレーブアドレスと方向ビットが出力されます。あらかじめ、<ACK>に"1"を設定しておいてください。

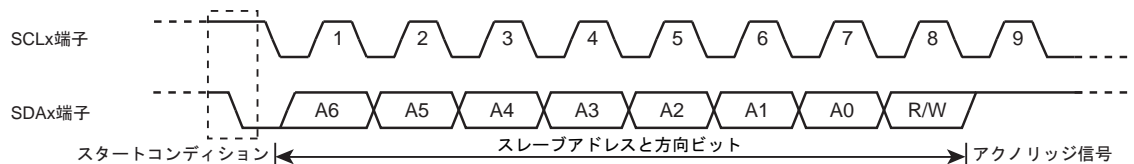


図 11-5 スタートコンディションの発生とスレーブアドレスの発生

<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が開始されます。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。

ストップコンディション発生時にバスの SCL ラインがほかのデバイスにより"Low"に引かれていた場合、SCL ラインが開放された後にストップコンディションが発生します。

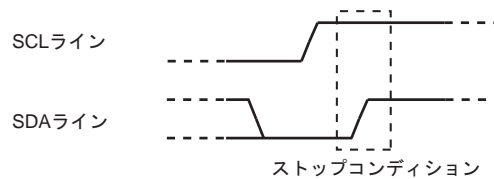


図 11-6 ストップコンディションの発生

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"にクリアされます(バスフリー状態)。

### 11.5.9 割り込みサービス要求と解除

マスターモードの時、<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 INTSBIx が発生します。

割り込み要求(INTSBIx)が発生すると、SBIxCR2<PIN>が"0"にクリアされます。<PIN>が"0"の間、SCL ラインを"Low"レベルに引きます。

<PIN>は 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。プログラムで<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。

注) マスタモードでアービトレーションロストが発生した時、スレーブアドレスが一致しなかった場合は、<PIN>は"0"にクリアされません。(INTSBIx は発生します。)

### 11.5.10 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、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"Low"レベルを出力、マスタ B が"High"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"Low"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。

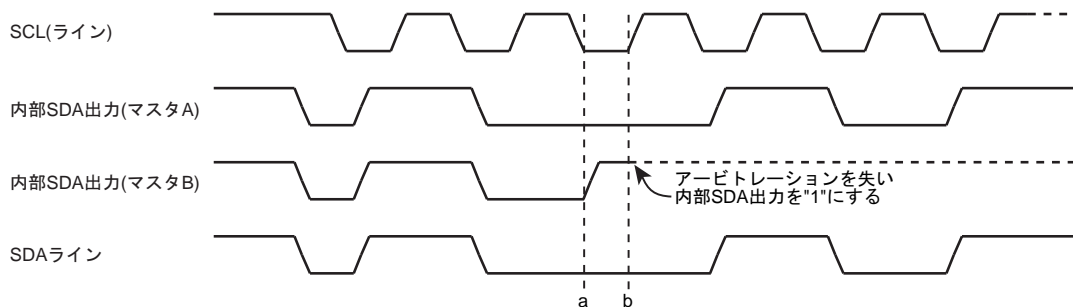


図 11-7 アービトレーションロスト

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"にクリアされ、スレーブレシーバモードになります。そのため、<AL>が"1"にセットされた後のデータ転送ではシリアルバスインタフェース回路はクロック出力を停止します。

<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"にクリアされます。

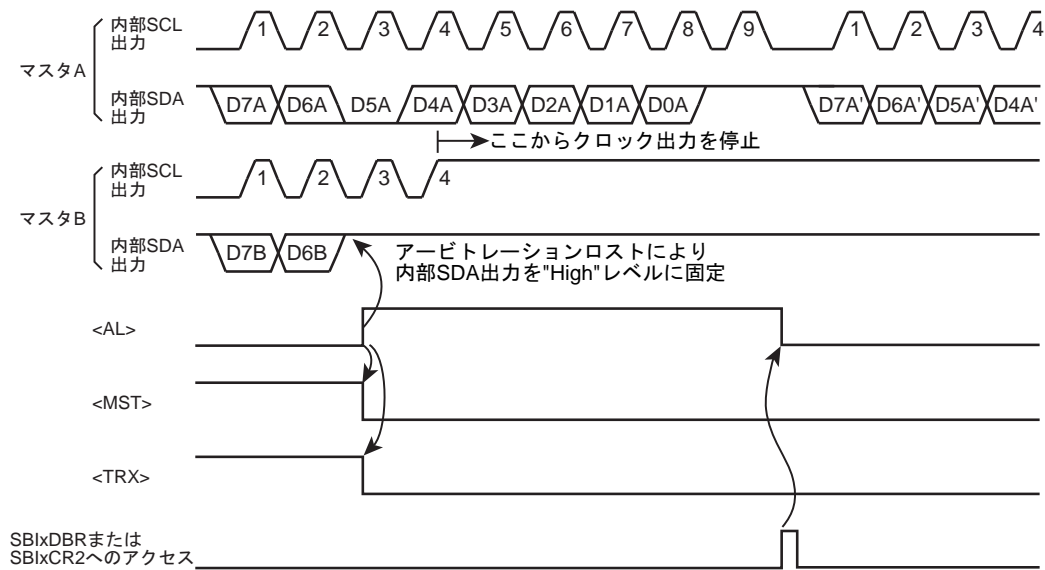


図 11-8 マスタ B の場合の例(D7A=D7B,D6A=D6B)

### 11.5.11 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。フリーデータフォーマット(<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"にクリアされます。

### 11.5.12 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"にセットされます。また、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"にクリアされます。

### 11.5.13 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。

### 11.5.14 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が発生します。

### 11.5.15 ボーレートレジスタ(SBIxBR0)

SBIxBR0<I2SBI>は IDLE モードに遷移した際にシリアルバスインタフェース回路の動作の許可/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。

### 11.5.16 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。

## 11.6 I2C バスモード時のデータ転送手順

### 11.6.1 デバイスの初期化

最初に SBIxCR1<ACK, SCK[2:0]>を設定します。SBIxCR1[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、シリアルバスインタフェース端子の状態が"High"になっていることを確認し、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。

注) シリアルバスインタフェース回路の初期化は、バスに接続されているすべてのデバイスが初期化された後、どのデバイスも一定期間スタートコンディションを発生しない期間を設け、その期間内に終了させてください。この制約が守られない場合、シリアルバスインタフェース回路の初期化が終了する前にほかのデバイスが転送を開始することがあり、正常にデータを受信することができません。

	7	6	5	4	3	2	1	0	
SBIxCR1	← 0	0	0	X	0	X	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	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	0	スレーブレシーバモードにします。

注) X; Don't care

### 11.6.2 スタートコンディション, スレーブアドレスの発生

#### 11.6.2.1 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、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、SCLx 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"Low"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。

注) スレーブアドレスを出力するために SBIxDBR に書き込む時は事前にソフトウェアによってバスフリーを検出してから行ってください。この制約が守られない場合、現在出力中のバス上のデータが破壊されることがあります。



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	X	アクノリッジモードに設定します。
SBIXDR1	←	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。

INTSBIX 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

11.6.2.2 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 "Low" レベルに引き、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 "Low" レベルに引きます。

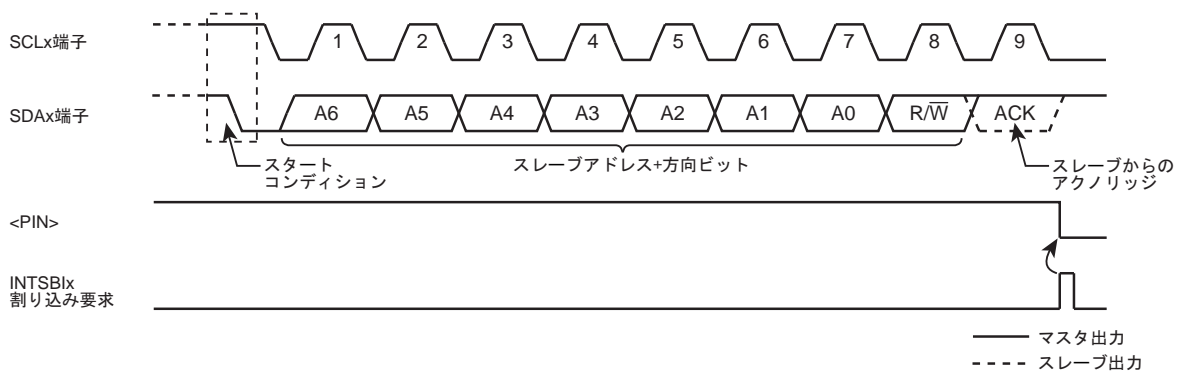


図 11-9 スタートコンディションとスレーブアドレスの発生

### 11.6.3 1ワードのデータ転送

1ワード転送終了の INTSB<sub>I</sub>x 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。

#### 11.6.3.1 マスタモードの場合(<MST> = "1")

<TRX>をテストし、トランスミッタ/レシーバの判断をします。

##### (1) トランスミッタモードの場合(<TRX> = "1")

<LRB>をテストします。<LRB> = "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> = "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ビットのとき 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、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 INTSB<sub>I</sub>x 割り込み要求が発生し、<PIN>が"0"になり SCL 端子を"Low"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<sub>I</sub>x 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

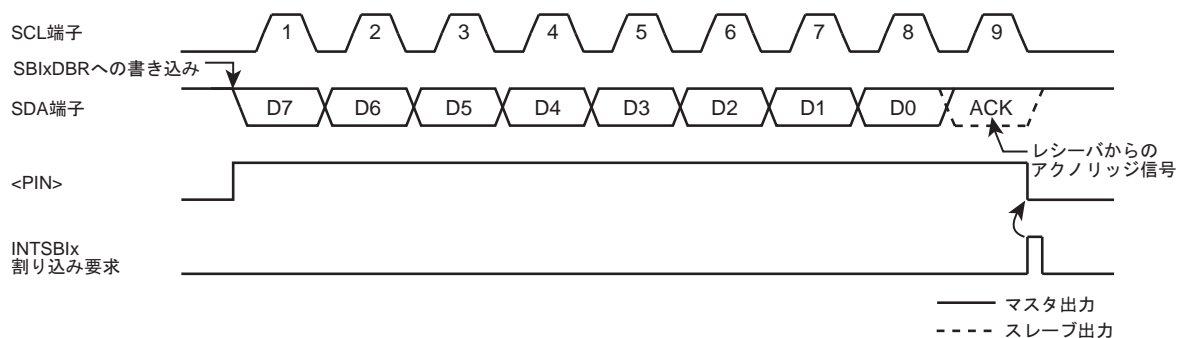


図 11-10 <BC[2:0]>="000", <ACK>="1"の場合 (トランスミッタモード)

(2) 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 SCLx 端子に出力します。最後のビットでアクノリッジ信号の"Low"レベルのタイミングで"0"を SDAx 端子に出力します。

その後、INTSBIx 割り込み要求が発生し、<PIN>が"0"になり SCLx 端子を"Low"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。

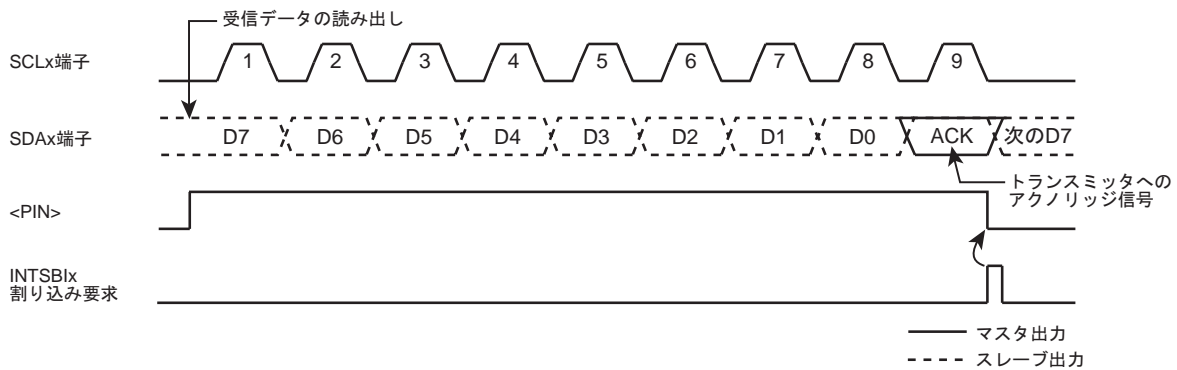


図 11-11 <BC[2:0]>="000", <ACK>="1"のときの例 (レシーバモード)

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が発生します。このときマスタはレシーバなのでバスの SDA ラインは"High"レベルを保ちます。トランスミッタは ACK 信号としてこの"High"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。

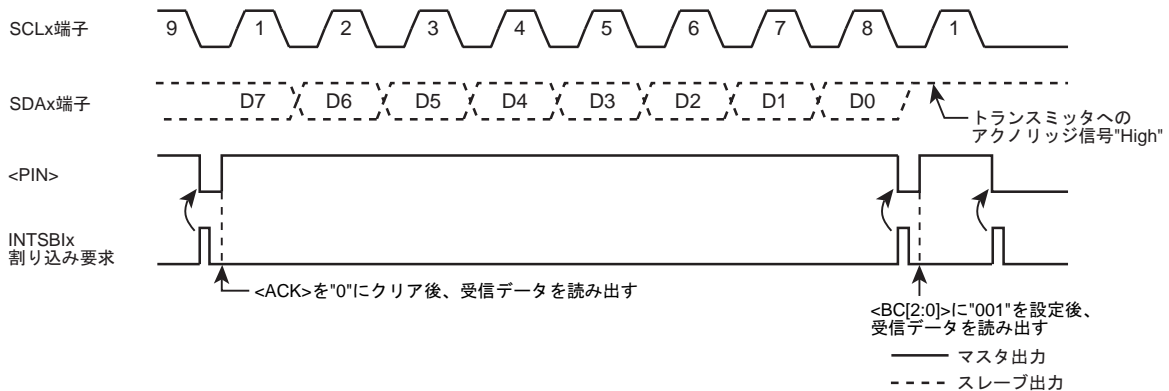


図 11-12 マスタレシーバモード時、データの送信を終了させるときの処理

例:データを N 回受信する場合

#### INTSB<sub>ix</sub> 割り込み(データ送信後)

		7	6	5	4	3	2	1	0
SB <sub>ix</sub> CR1	←	X	X	X	X	0	X	X	X
Reg.	←	SB <sub>ix</sub> DBR							

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。

#### INTSB<sub>ix</sub> 割り込み(データ受信 1~(N - 2)回目)

		7	6	5	4	3	2	1	0
Reg.	←	SB <sub>ix</sub> DBR							

割り込み終了

1~(N - 2)回目のデータを取り込みます。

#### INTSB<sub>ix</sub> 割り込み(データ受信 (N - 1)回目)

		7	6	5	4	3	2	1	0
SB <sub>ix</sub> CR1	←	X	X	X	0	0	X	X	X
Reg.	←	SB <sub>ix</sub> DBR							

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。

#### INTSB<sub>ix</sub> 割り込み(データ受信 N 回目)

		7	6	5	4	3	2	1	0
SB <sub>ix</sub> CR1	←	0	0	1	0	0	X	X	X
Reg.	←	SB <sub>ix</sub> DBR							

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。

#### INTSB<sub>ix</sub> 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

11.6.3.2 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、SCLx 端子を"Low"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 SCLx 端子が t<sub>LOW</sub> 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>, <AD0>をテストし、場合分けを行います。「表 11-2 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 = 0
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 11-2 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<BC[2:0]>にセットし、送信するデータを SBIxDBR に書き込みます。
		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ので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	

### 11.6.4 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 ラインが開放されてから、SDAx 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	ストップコンディションを発生させます。

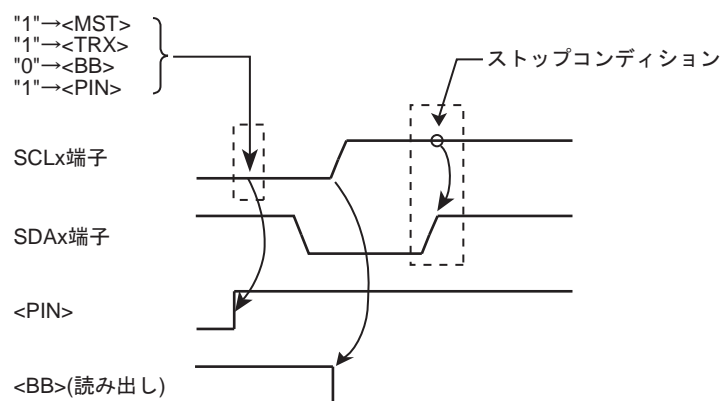


図 11-13 ストップコンディションの発生

### 11.6.5 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、再スタートが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 SDAx 端子は"High"レベルを保ち、SCLx 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、SCLx 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"Low"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に前記「11.6.2 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 4.7 μs (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。

注 1) <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)。

注 2) マスタデバイスがレシーバのとき、再スタートを発生させる前にトランスミッタとなっているスレーブデバイスからのデータ送信を終了させる必要があります。データ送信を終了させるために、"High"レベルのアクノリッジ信号をスレーブデバイスに受信させます。このため、再スタート発生前の<LBR>は"1"となり、再スタートの手順で<LBR>="1"であることを確認しても、SCL ラインの立ち上がりを確認できません。SCL ラインの状態を確認するにはポートを読み出してください。

	7 6 5 4 3 2 1 0	
→	SBIxCR2 ← 0 0 0 1 1 0 0 0	バスを開放します。
	if SBIxSR<BB> ≠ 0	SCL 端子の開放を確認します。
→	Then	
	if SBIxSR<LRB> ≠ 1	他のデバイスの SCL 端子"Low"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	アクリ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

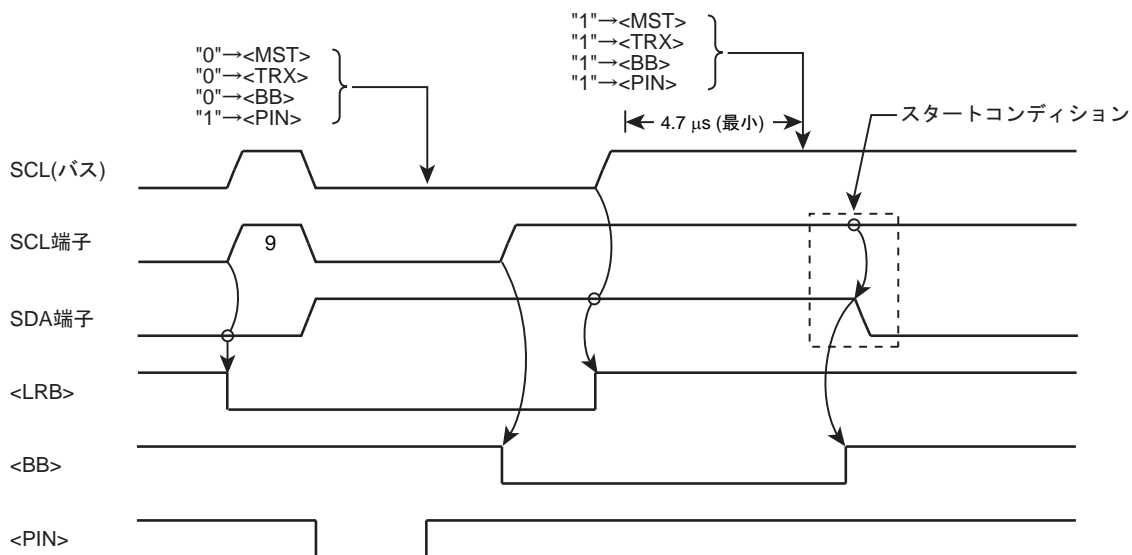


図 11-14 再スタートを発生する場合のタイミングチャート

## 11.7 SIO 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。

### 11.7.1 SBIXCR0(コントロールレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SBIEN	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	SBIEN	R/W	シリアルバスインタフェース動作 0: 禁止 1: 許可 シリアルバスインタフェースを使用する場合、まずこのビットを許可にしてください。 禁止の場合、SBIXCR0 を除く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。 いったん許可した後に禁止にした場合、各レジスタの設定は保持されます。
6-0	-	R	リードすると"0"が読めます。



11.7.2 SBiXCR1(コントロールレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SIOS	SIOINH	SIOM		-	SCK		
リセット後	0	0	0	0	1	0	0	0(注 1)

Bit	Bit Symbol	Type	機能																								
31-8	-	R	リードすると"0"が読めます。																								
7	SIOS	R/W	転送の開始/終了 0: 終了 1: 開始																								
6	SIOINH	R/W	転送の強制停止 0: 転送継続 1: 強制停止																								
5-4	SIOM[1:0]	R/W	転送モードの設定 00: 送信モード 01: Reserved 10: 送受信モード 11: 受信モード																								
3	-	R	リードすると"1"が読めます。																								
2-0	SCK[2:0]	R/W	シリアルクロックの周波数選択<SCK[2:0]>@ライト(注 1) <table border="1" style="margin-left: 20px;"> <tr> <td>000</td> <td>n = 3</td> <td>2.5 MHz</td> </tr> <tr> <td>001</td> <td>n = 4</td> <td>1.25 MHz</td> </tr> <tr> <td>010</td> <td>n = 5</td> <td>625 kHz</td> </tr> <tr> <td>011</td> <td>n = 6</td> <td>313 kHz</td> </tr> <tr> <td>100</td> <td>n = 7</td> <td>156 kHz</td> </tr> <tr> <td>101</td> <td>n = 8</td> <td>78 kHz</td> </tr> <tr> <td>110</td> <td>n = 9</td> <td>39 kHz</td> </tr> <tr> <td>111</td> <td>-</td> <td>外部クロック</td> </tr> </table> <div style="margin-left: 20px;"> <math display="block">\left. \begin{array}{l} \text{システムクロック: } f_{\text{sys}} \\ \text{クロックギア: } fc/1 \\ \text{周波数} = \frac{f_{\text{sys}}/2}{2^n} \text{ [Hz]} \end{array} \right\} (= 40 \text{ MHz})</math> </div>	000	n = 3	2.5 MHz	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
000	n = 3	2.5 MHz																									
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"が読み出されますが、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。

## 11.7.3 SBiXDBR(データバッファレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	DB							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	DB[7:0]	R	受信データ
		W	送信データ

- 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。

11.7.4 SBIXCR2(コントロールレジスタ 2)

このレジスタをリードすると、SBIXSR レジスタとして機能します。

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	SBIM		-	-
リセット後	1(注 1)	1(注 1)	1(注 1)	1(注 1)	0	0	1(注 1)	1(注 1)

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-4	-	R	リードすると"1"が読めます。(注 1)
3-2	SBIM[1:0]	W	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: I2C バスモード 11: Reserved
1-0	-	R	リードすると"1"が読めます。(注 1)

注 1) ここでは SIO モードに設定後の初期値を「リセット後」欄に示します。

注 2) 通信中はモードを切り替えないでください。

## 11.7.5 SBIXSR(ステータスレジスタ)

このレジスタをライトすると、SBIXCR2 として機能します。

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	SIOF	SEF	-	-
リセット後	1(注 1)	1(注 1)	1(注 1)	1(注 1)	0	0	1(注 1)	1(注 1)

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-4	-	R	リードすると"1"が読めます。(注 1)
3	SIOF	R	シリアル転送動作状態モニタ 0: 転送終了 1: 転送中
2	SEF	R	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
1-0	-	R	リードすると"1"が読めます。(注 1)

注 1) ここでは SIO モードに設定後の初期値を「リセット後」欄に示します。

11.7.6 SBiBR0(ポーレートレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	I2SBI	-	-	-	-	-	-
リセット後	1	0	1	1	1	1	1	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	-	R	リードすると"1"が読めます。
6	I2SBI	R/W	IDLE モード時の動作 0: 停止 1: 動作
5-1	-	R	リードすると"1"が読めます。
0	-	R/W	必ず"0"をライトしてください。

## 11.8 SIO モード時の制御

### 11.8.1 シリアルクロック

#### 11.8.1.1 クロックソース

SBIxCR1 <SCK[2:0]> により、次の選択ができます。

##### (1) 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 SCKx 端子より外部に出力されます。なお、転送開始時 SCKx 端子出力は "High" 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。

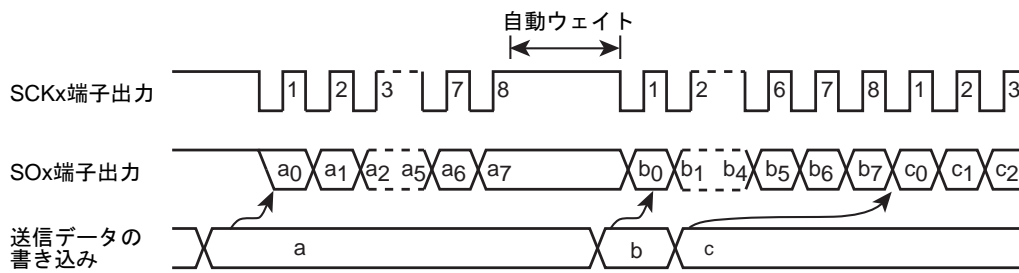


図 11-15 自動ウェイト機能

##### (2) 外部クロック (<SCK[2:0]> = "111")

外部から SCKx 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 "High" レベル、"Low" レベル幅は下記に示すパルス幅が必要です。

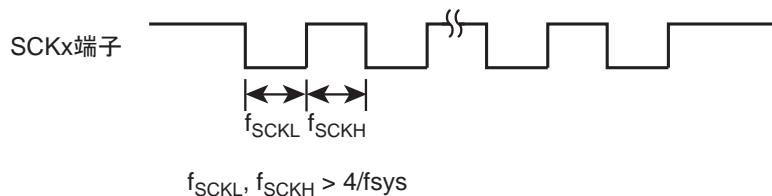


図 11-16 外部クロック入力時の最大転送周波数

11.8.1.2 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(SCKx 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(SCKx 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。

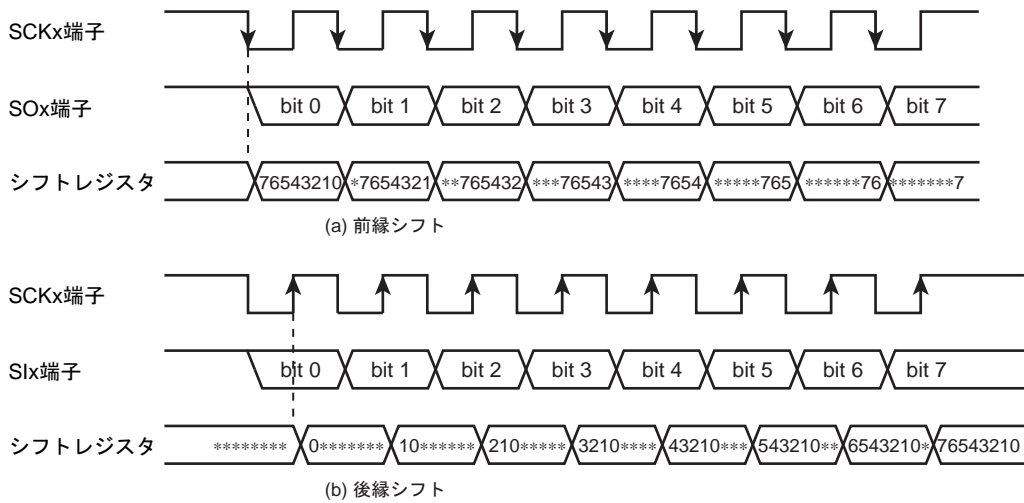


図 11-17 シフトエッジ

## 11.8.2 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。

### 11.8.2.1 8ビット送信モード

コントロール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"にクリアする必要があります。もしシフトアウトする前に<SIOS>が"0"にクリア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。
---------	-----	---	---	---	---	---	---	---	---------------



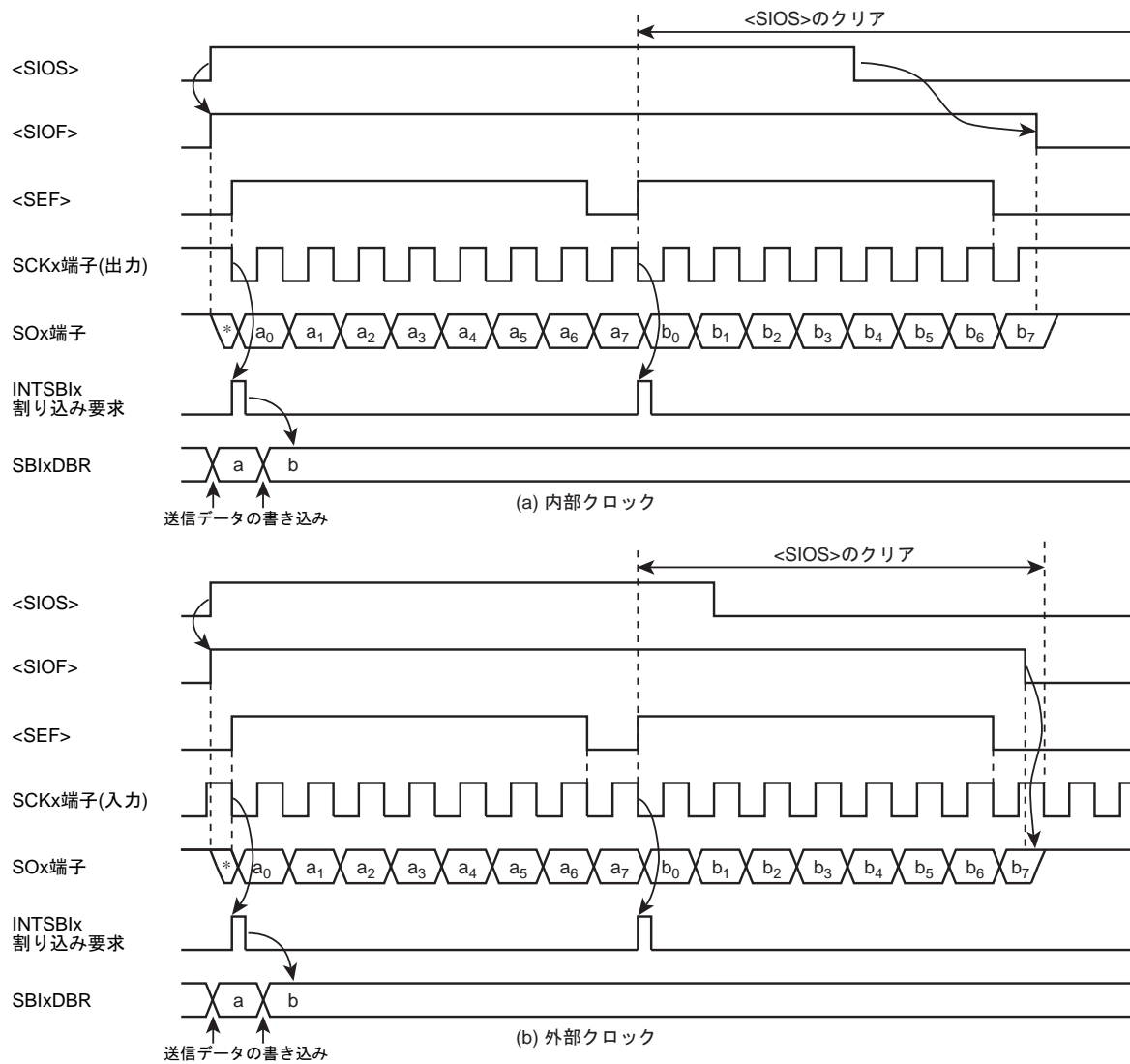


図 11-18 送信モード

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 SBIxSR<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、SCKx 端子が"1"になったことを確認
    Then                            します。
    SBIxCR1 ← 0 0 0 0 0 0 1 1 1  <SIO> = 0 を設定し送信を終了します。
  
```

### 11.8.2.2 8ビット受信モード

コントロール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。
------	---	---------	---------------

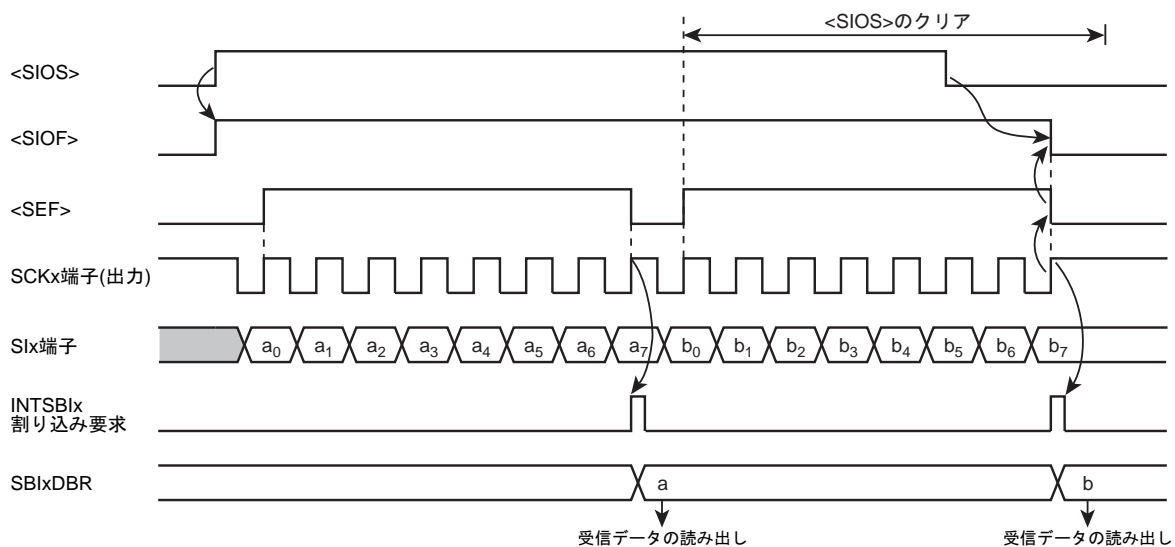


図 11-19 受信モード(例: 内部クロック)

11.8.2.3 8ビット送受信モード

コントロール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、SBIxCR1<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 SOx 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、必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は、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 SCKx 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 SBIxCR1 <SIOINH> = "1"を書き込みます。<SIOS>が"0"にクリアされると、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"にクリアされます。<SIOINH>を"1"にセットした場合は、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。

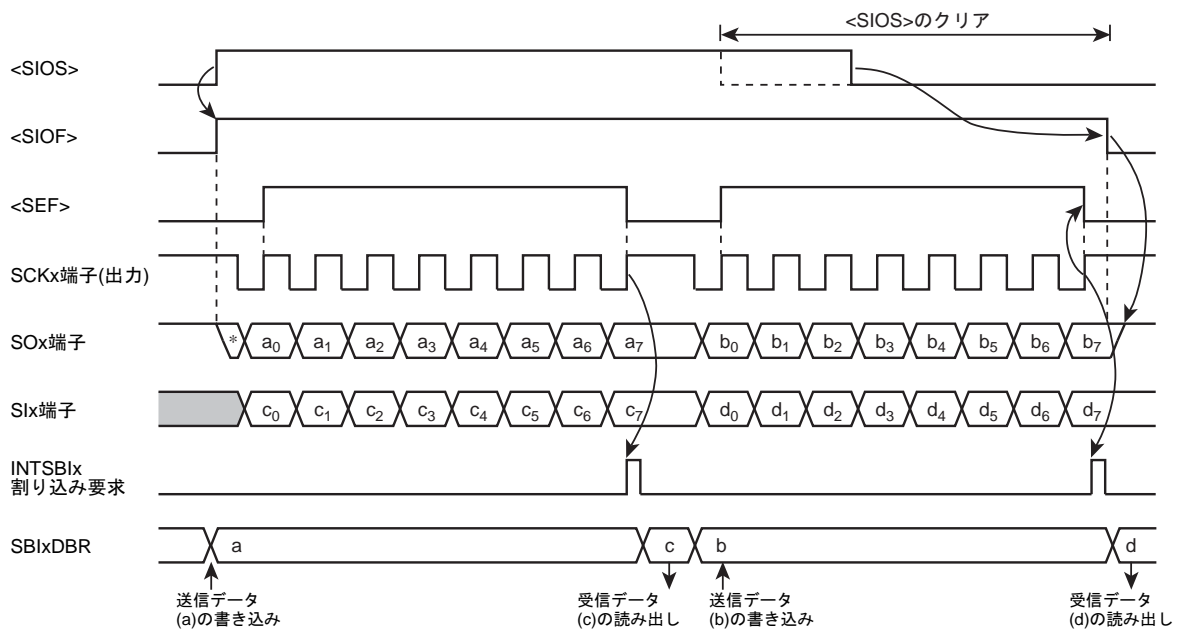


図 11-20 送受信モード(例: 内部クロック)

		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。

11.8.2.4 送信終了時の最終ビット保持時間

SBIxCR1<SIOS>="0"の状態では、送信データの最終ビットの SCK 立ち上がりに対するデータ保持時間は以下ようになります。送信モード、送受信モードとも同様です。

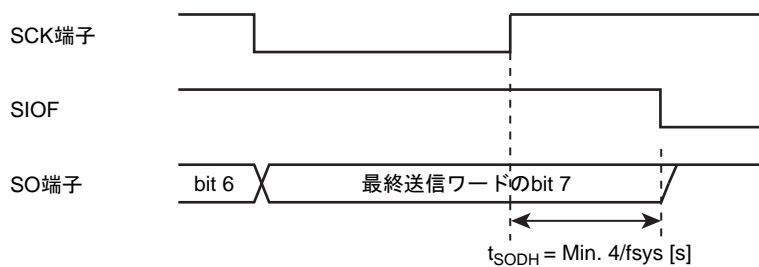


図 11-21 送信終了時の最終ビット保持時間

## 第 12 章 CEC 機能

### 12.1 概要

CEC (Consumer Electronics Control)データの送受信を行います。

HDMI 規格 Version 1.3a に準拠した動作が可能です。

#### 12.1.1 受信

- ・ 32.768kHz 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

#### 12.1.2 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

#### 12.1.3 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	-	
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")	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。

## 12.2 ブロック図

図 12-1 に CEC のブロック図を示します。

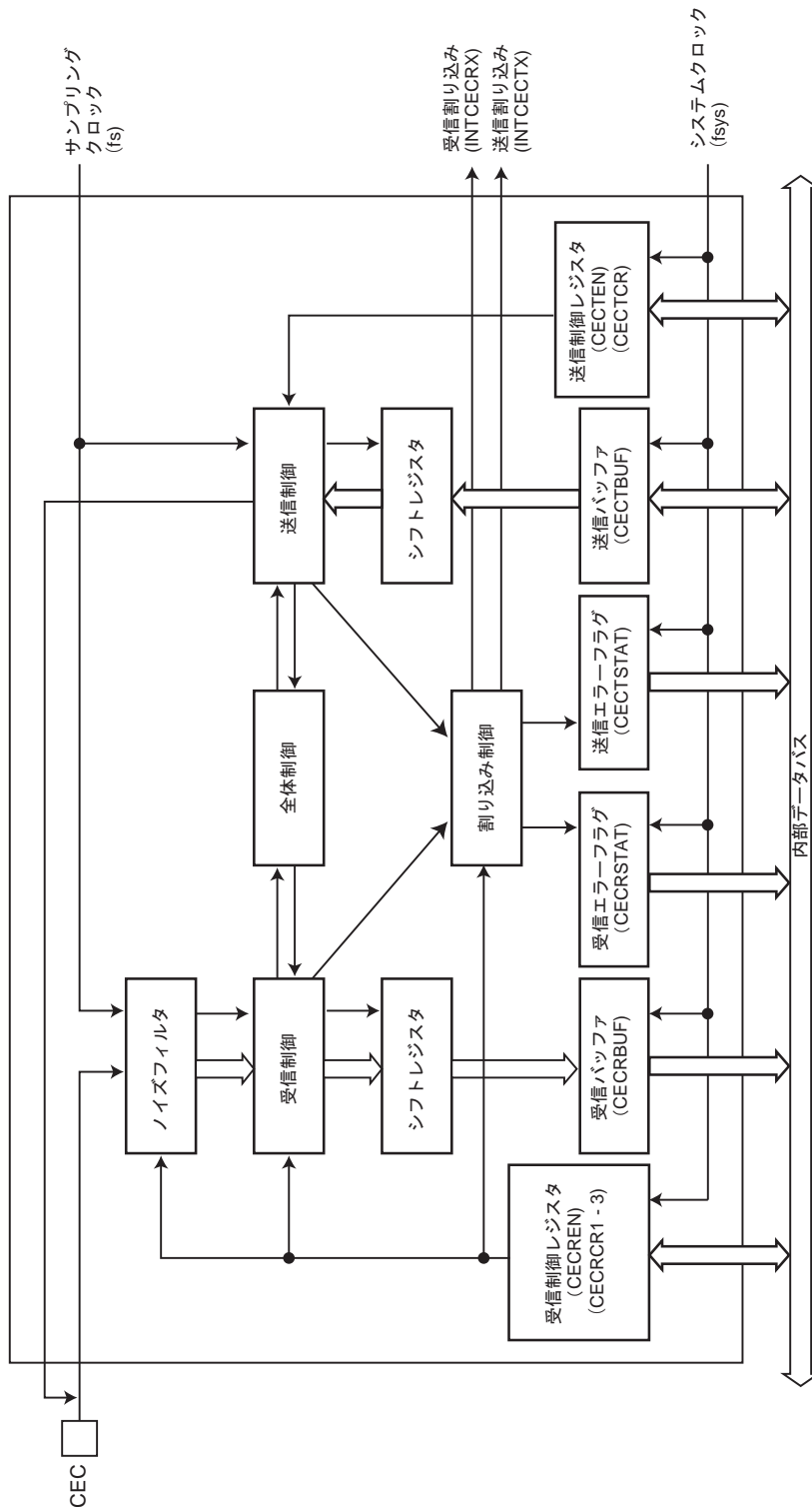


図 12-1 CEC ブロック図

## 12.3 レジスタ説明

### 12.3.1 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。

Base Address = 0x4004\_0300

レジスタ名		Address(Base+)
CEC イネーブルレジスタ	CECEN	0x0000
ロジカルアドレスレジスタ	CECADD	0x0004
ソフトウェアリセットレジスタ	CECRESET	0x0008
受信イネーブルレジスタ	CECREN	0x000C
受信バッファレジスタ	CECRBUF	0x0010
受信コントロールレジスタ 1	CECR1	0x0014
受信コントロールレジスタ 2	CECR2	0x0018
受信コントロールレジスタ 3	CECR3	0x001C
送信イネーブルレジスタ	CECTEN	0x0020
送信バッファレジスタ	CECTBUF	0x0024
送信コントロールレジスタ	CECTCR	0x0028
受信割り込みステータスレジスタ	CECRSTAT	0x002C
送信割り込みステータスレジスタ	CECTSTAT	0x0030

## 12.3.2 CECEN(CEC イネーブルレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	I2CEC	CECEN
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-2	-	R	リードすると"0"が読めます。
1	I2CEC	R/W	IDLE 時の動作 0: 禁止 1: 動作 IDLE モード時の CEC 機能の動作を制御します。IDLE モードで CEC を使用するときには、"1"を設定してください。 このビットは<CECEN>と同時に設定できます。
0	CECEN	R/W	CEC 動作 0: 禁止 1: 動作 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、CECEN 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 機能をいったん動作させた後に、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。



## 12.3.3 CECADD(ロジカルアドレスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	CECADD[15:8]							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CECADD[7:0]							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-0	CECADD[15:0]	R/W	ロジカルアドレス 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。

## 12.3.4 CECRESET(ソフトウェアリセットレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	CECRESET
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	CECRESET	W	ソフトウェアリセット 0: 禁止 1: 許可 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。 このビットをリードすると"0"が読めます。

## 12.3.5 CECREN(受信イネーブルレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	CECREN
リセット後	0	0	0	0	0	0	0	不定

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	CECREN	R/W	受信制御 [ライト] 0:禁止 1:許可 [リード] 0:停止中 1:動作中  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>は、CECR1, CECRCR2, CECRCR3 レジスタを設定した後に許可してください。

注 2) <CECREN>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。

### 12.3.6 CECRBUF(受信バッファレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	CECACK	CECEOM
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CECRBUF							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-10	-	R	リードすると"0"が読めます。
9	CECACK	R	ACK ビット 受信した ACK ビットが読めます。
8	CECEOM	R	EOM ビット 受信した EOM ビットが読めます。
7-0	CECRBUF[7:0]	R	受信データ 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。

12.3.7 CECRCR1(受信コントロールレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	CECACKDIS
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	CECHNC		-	CECLNC		
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	CECMIN			-	CECMAX		
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	CECDAT			CECTOUT		CECRIHLD	CECOTH
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-25	-	R	リードすると"0"が読めます。
24	CECACKDIS	R/W	ACK 応答制御 0: 論理 0 応答する 1: 論理 0 応答しない  ディステーションアドレスが CECADD 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。)
23-22	-	R	リードすると"0"が読めます。
21-20	CECHNC[1:0]	R/W	"High"検出ノイズキャンセル時間(注 1) 00: なし (fs クロック 1 回観測) 01: 1/fs (fs クロック 2 回連続観測) 10: 2/fs (fs クロック 3 回連続観測) 11: 3/fs (fs クロック 4 回連続観測)  "High"を検出する際のノイズキャンセル時間を 1/fs 単位で設定します。 設定されたサイクル数分の"High"が観測されなければノイズとみなします。
19	-	R	リードすると"0"が読めます。
18-16	CECLNC[2:0]	R/W	"Low"検出ノイズキャンセル時間(注 1)(注 4) 000: なし (fs クロック 1 回観測)      100: - (Reserved) 001: 1/fs (fs クロック 2 回連続観測)      101: - (Reserved) 010: 2/fs (fs クロック 3 回連続観測)      110: - (Reserved) 011: 3/fs (fs クロック 4 回連続観測)      111: - (Reserved)  "Low"を検出する際のノイズキャンセル時間を 1/fs 単位で設定します。 設定されたサイクル数分の"Low"がサンプリングされなければノイズとみなします。
15	-	R	リードすると"0"が読めます。
14-12	CECMIN[2:0]	R/W	最小周期違反検出時間 000: 67/fs(約 2.045ms)      100: 67/fs - 1/fs 001: 67/fs + 1/fs      101: 67/fs - 2/fs 010: 67/fs + 2/fs      110: 67/fs - 3/fs 011: 67/fs + 3/fs      111: 67/fs - 4/fs  有効なビットとみなす最短の時間を設定します。 67/fs(約 2.045) ms を基準に、1/fs 単位で-4/fs から+3/fs 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 3.63 ms 間"Low"を出力します。
11	-	R	リードすると"0"が読めます。

Bit	Bit Symbol	Type	機能																
10-8	CECMAX[2:0]	R/W	<p>最大周期違反検出時間</p> <table border="0"> <tr> <td>000:</td> <td>90/fs(約 2.747ms)</td> <td>100:</td> <td>90/fs - 1/fs</td> </tr> <tr> <td>001:</td> <td>90/fs + 1/fs</td> <td>101:</td> <td>90/fs - 2/fs</td> </tr> <tr> <td>010:</td> <td>90/fs + 2/fs</td> <td>110:</td> <td>90/fs - 3/fs</td> </tr> <tr> <td>011:</td> <td>90/fs + 3/fs</td> <td>111:</td> <td>90/fs - 4/fs</td> </tr> </table> <p>有効なビットとみなす最長の時間を設定します。 90/fs(約 2.747 ms)を基準に、1/fs 単位で-4/fs から+3/fs 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。</p>	000:	90/fs(約 2.747ms)	100:	90/fs - 1/fs	001:	90/fs + 1/fs	101:	90/fs - 2/fs	010:	90/fs + 2/fs	110:	90/fs - 3/fs	011:	90/fs + 3/fs	111:	90/fs - 4/fs
000:	90/fs(約 2.747ms)	100:	90/fs - 1/fs																
001:	90/fs + 1/fs	101:	90/fs - 2/fs																
010:	90/fs + 2/fs	110:	90/fs - 3/fs																
011:	90/fs + 3/fs	111:	90/fs - 4/fs																
7	-	R	リードすると"0"が読めます。																
6-4	CECDAT[2:0]	R/W	<p>データ 0/1 判別タイミング(注 1)</p> <table border="0"> <tr> <td>000:</td> <td>34/fs(約 1.038ms)</td> <td>100:</td> <td>34/fs - 2/fs</td> </tr> <tr> <td>001:</td> <td>34/fs + 2/fs</td> <td>101:</td> <td>34/fs - 4/fs</td> </tr> <tr> <td>010:</td> <td>34/fs + 4/fs</td> <td>110:</td> <td>34/fs - 6/fs</td> </tr> <tr> <td>011:</td> <td>34/fs + 6/fs</td> <td>111:</td> <td>Reserved</td> </tr> </table> <p>データの論理"0"/論理"1"判別を行うポイントを設定します。 34/fs(約 1.038 ms)を基準に、2/fs 単位で±6/fs まで設定可能です。</p>	000:	34/fs(約 1.038ms)	100:	34/fs - 2/fs	001:	34/fs + 2/fs	101:	34/fs - 4/fs	010:	34/fs + 4/fs	110:	34/fs - 6/fs	011:	34/fs + 6/fs	111:	Reserved
000:	34/fs(約 1.038ms)	100:	34/fs - 2/fs																
001:	34/fs + 2/fs	101:	34/fs - 4/fs																
010:	34/fs + 4/fs	110:	34/fs - 6/fs																
011:	34/fs + 6/fs	111:	Reserved																
3-2	CECTOUT[1:0]	R/W	<p>タイムアウト判定時間</p> <table border="0"> <tr> <td>00:</td> <td>1bit 周期</td> </tr> <tr> <td>01:</td> <td>2bit 周期</td> </tr> <tr> <td>10:</td> <td>3bit 周期</td> </tr> <tr> <td>11:</td> <td>Reserved</td> </tr> </table> <p>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。 &lt;CECRIHLD&gt;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。</p>	00:	1bit 周期	01:	2bit 周期	10:	3bit 周期	11:	Reserved								
00:	1bit 周期																		
01:	2bit 周期																		
10:	3bit 周期																		
11:	Reserved																		
1	CECRIHLD	R/W	<p>エラー割り込み保留</p> <p>0: 保留しない 1: 保留する</p> <p>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&lt;CECTOUT&gt;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。</p>																
0	CECOTH	R/W	<p>ロジカルアドレス不一致時の動作(注 3)</p> <p>0: 受信しない 1: 受信する</p> <p>ディステーションアドレスが、CECADD 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。</p>																

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<CECREN>で受信禁止の設定を行い、<CECREN>および CECTEN<CECTRANS>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<CECDTRS>と同じ設定で使用してください。

12.3.8 CECRCR2(受信コントロールレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	CECSWAV3			-	CECSWAV2		
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	CECSWAV1			-	CECSWAV0		
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-15	-	R	リードすると"0"が読めます。
14-12	CECSWAV3[2:0]	R/W	スタートビット検出設定 3 000: 154/fs(約 4.700 ms) 100: 154/fs + 4/fs 001: 154/fs + 1/fs 101: 154/fs + 5/fs 001: 154/fs + 2/fs 110: 154/fs + 6/fs 011: 154/fs + 3/fs 111: 154/fs + 7/fs
11	-	R	リードすると"0"が読めます。
10-8	CECSWAV2[2:0]	R/W	スタートビット検出設定 2 000: 141/fs(約 4.303 ms) 100: 141/fs - 4/fs 001: 141/fs - 1/fs 101: 141/fs - 5/fs 001: 141/fs - 2/fs 110: 141/fs - 6/fs 011: 141/fs - 3/fs 111: 141/fs - 7/fs
7	-	R	リードすると"0"が読めます。
6-4	CECSWAV1[2:0]	R/W	スタートビット検出設定 1 000: 128/fs(約 3.906 ms) 100: 128/fs + 4/fs 001: 128/fs + 1/fs 101: 128/fs + 5/fs 001: 128/fs + 2/fs 110: 128/fs + 6/fs 011: 128/fs + 3/fs 111: 128/fs + 7/fs
3	-	R	リードすると"0"が読めます。
2-0	CECSWAV0[2:0]	R/W	スタートビット検出設定 0 000: 115/fs(約 3.510 ms) 100: 115/fs - 4/fs 001: 115/fs - 1/fs 101: 115/fs - 5/fs 001: 115/fs - 2/fs 110: 115/fs - 6/fs 011: 115/fs - 3/fs 111: 115/fs - 7/fs

<CECSWAV3>:/	スタートビット検出時の周期の条件を設定します。
<CECSWAV2>:	<CECSWAV3>で周期の最大値の条件を設定します。154/fs(約 4.700ms)を基準に、1/fs 単位で 0 から+7/fs 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。141/fs(約 4.303ms)を基準に、1/fs 単位で 0 から-7/fs まで設定可能です。
<CECSWAV1>:/	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>:	<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。128/fs(約 3.906ms)を基準に、1/fs 単位で 0 から+7/fs 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。115/fs(約 3.510ms)を基準に、1/fs 単位で 0 から-7/fs 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<CECREN>で受信禁止の設定を行い、<CECREN>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>: <CECWAVEN>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。56/fs(約 1.709ms)に対し、1/fs 単位で 0 から+7/fs 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>/ <CECWAVEN>が"1"に設定されている場合に有効になります。
- <CECWAV1>: 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 26/fs(約 0.793ms)に対し、1/fs 単位で 0 から+7/fs まで設定が可能です。  
<CECWAV2>で 43/fs(約 1.312ms)に対し、1/fs 単位で 0 から-7/fs まで設定を可能です。  
<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>: <CECWAVEN>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。13/fs(約 0.396ms)に対し、1/fs 単位で 0 から-7/fs 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。

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<CECREN>で受信禁止の設定を行い、<CECREN>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。

## 12.3.10 CECTEN(送信イネーブルレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	CECTRANS	CECTEN
リセット後	0	0	0	0	0	0	0	不定

Bit	Bit Symbol	Type	機能
31-2	-	R	リードすると"0"が読めます。
1	CECTRANS	R	送信中 0: 送信していない 1: 送信中 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。
0	CECTEN	W	送信制御 0: 禁止 1: 許可 CEC 機能の送信動作を制御します。 このビットを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>は、CECTBUF、CECTCR 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。

12.3.11 CECTBUF(送信バッファレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	CECTEOM
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CECTBUF							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-9	-	R	リードすると"0"が読めます。
8	CECTEOM	R/W	EOM ビット 送信する EOM ビットを設定します。
7-0	CECTBUF[7:0]	R/W	送信データ 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。

12.3.12 CECTCR(送信コントロールレジスタ)

	31	30	29	28	27	26	25	24	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	23	22	21	20	19	18	17	16	
bit symbol	-	CECSTRS				-	CECSPRD		
リセット後	0	0	0	0	0	0	0	0	
	15	14	13	12	11	10	9	8	
bit symbol	-	CECDTRS				CECDPRD			
リセット後	0	0	0	0	0	0	0	0	
	7	6	5	4	3	2	1	0	
bit symbol	-	-	-	CECBRD	CECFREE				
リセット後	0	0	0	0	0	0	0	0	

Bit	Bit Symbol	Type	機能
31-23	-	R	リードすると"0"が読めます。
22-20	CECSTRS[2:0]	R/W	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 - 1/fs 010: 基準値 - 2/fs 011: 基準値 - 3/fs 100: 基準値 - 4/fs 101: 基準値 - 5/fs 110: 基準値 - 6/fs 111: 基準値 - 7/fs 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。 121/fs(約 3.693 ms)を基準に、1/fs 単位で 0 から -7/fs まで設定可能です。
19	-	R	リードすると"0"が読めます。
18-16	CECSPRD[2:0]	R/W	スタートビット波形周期時間調整 000: 基準値 001: 基準値 - 1/fs 010: 基準値 - 2/fs 011: 基準値 - 3/fs 100: 基準値 - 4/fs 101: 基準値 - 5/fs 110: 基準値 - 6/fs 111: 基準値 - 7/fs スタートビットの周期の設定を行います。 147/fs(約 4.486 ms)を基準に、1/fs 単位で 0 から -7/fs まで設定可能です。
15	-	R	リードすると"0"が読めます。
14-12	CECDTRS[2:0]	R/W	データ波形立ち上がり時間調整(注) 000: 基準値 001: 基準値 - 1/fs 010: 基準値 - 2/fs 011: 基準値 - 3/fs 100: Reserved 101: Reserved 110: Reserved 111: Reserved 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。 20/fs(約 0.610 ms、論理"1"の場合)または 49/fs(約 1.495 ms、論理"0"の場合)を基準に、1/fs 単位で 0 から -3/fs まで設定可能です。
11-8	CECDPRD[2:0]	R/W	データ波形周期時間調整 0000: 基準値 0001: 基準値 - 1/fs 0010: 基準値 - 2/fs 0011: 基準値 - 3/fs 0100: 基準値 - 4/fs 0101: 基準値 - 5/fs 0110: 基準値 - 6/fs 0111: 基準値 - 7/fs 1000: 基準値 - 8/fs 1001: 基準値 - 9/fs 1010: 基準値 - 10/fs 1011: 基準値 - 11/fs 1100: 基準値 - 12/fs 1101: 基準値 - 13/fs 1110: 基準値 - 14/fs 1111: 基準値 - 15/fs データビットの周期の設定を行います。 79/fs(約 2.411 ms)を基準に、1/fs 単位で 0 から -15/fs サイクルまで設定可能です。
7-5	-	R	リードすると"0"が読めます。

Bit	Bit Symbol	Type	機能
4	CECBRD	R/W	ブロードキャスト送信設定 0: ブロードキャスト送信でない 1: ブロードキャスト送信 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。
3-0	CECFREE[3:0]	R/W	バスフリー待ち時間設定 0000: 1bit 周期                               1000: 9bit 周期 0001: 2bit 周期                               1001: 10bit 周期 0010: 3bit 周期                               1010: 11bit 周期 0011: 4bit 周期                               1011: 12bit 周期 0100: 5bit 周期                               1100: 13bit 周期 0101: 6bit 周期                               1101: 14bit 周期 0110: 7bit 周期                               1110: 15bit 周期 0111: 8bit 周期                               1111: 16bit 周期 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"Low"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。

## 12.3.13 CECRSTAT(受信割り込みステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	CECRIWAV	CECRIOR	CECRIACK	CECRIMIN	CECRIMAX	CECRISTA	CECRIEND
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6	CECRIWAV	R	割り込みフラグ 0: 波形エラー割り込みは発生していない 1: 波形エラー割り込みが発生 波形エラーが発生したことを示します。CECRCR3<CECWAVEN>で許可の設定をしている場合に発生します。
5	CECRIOR	R	割り込みフラグ 0: 受信バッファオーバーランは発生していない 1: 受信バッファオーバーランが発生 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。
4	CECRIACK	R	割り込みフラグ 0: ACK 衝突は発生していない 1: ACK 衝突が発生 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。
3	CECRIMIN	R	割り込みフラグ 0: 最小周期違反は発生していない 1: 最小周期違反が発生 1 ビットの周期が、最小周期違反検出時間として CECRCR1<CECMIN>に設定された時間より短かったことを示します。
2	CECRIMAX	R	割り込みフラグ 0: 最大周期違反は発生していない 1: 最大周期違反が発生 1 ビットの周期が、最大周期違反検出時間として CECRCR1<CECMAX>に設定された時間より長かったことを示します。
1	CECRISTA	R	割り込みフラグ 0: スタートビットは検出していない 1: スタートビットを検出した スタートビットが検出されたことを示します。
0	CECRIEND	R	割り込みフラグ 0: 1 バイト受信完了していない 1: 1 バイト受信完了した 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。

## 12.3.14 CECTSTAT(送信割り込みステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	CECTIUR	CECTIACK	CECTIAL	CECTIEND	CECTISTA
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると"0"が読めます。
4	CECTIUR	R	割り込みフラグ 0: 送信バッファアンダーランは発生していない 1: 送信バッファアンダーランが発生  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。
3	CECTIACK	R	割り込みフラグ 0: ACK エラーは検出していない 1: ACK エラーを検出  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。
2	CECTIAL	R	割り込みフラグ 0: アービトレーションロストは発生していない 1: アービトレーションロストが発生  "High"を出力中に"Low"を検出したことを示します。
1	CECTIEND	R	割り込みフラグ 0: 全メッセージの送信を終了していない 1: 全メッセージの送信終了  EOM 付きのデータの送信が終了したことを示します。
0	CECTISTA	R	割り込みフラグ 0: 送信を開始していない 1: 送信開始  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。

## 12.4 動作説明

### 12.4.1 サンプリングクロック

CEC 信号のサンプリングは、32.768kHz の低速クロック (fs)で行います。

### 12.4.2 受信

#### 12.4.2.1 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。

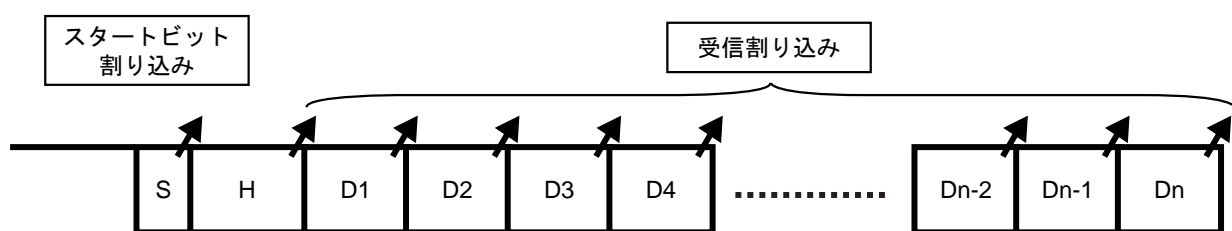
1 バイト分のデータと、EOM, ACK を受信すると受信したデータを CECRBUF レジスタに格納し、受信割り込みを発生します。受信割り込みが発生すると、CECRSTAT<CECRIEND>がセットされます。

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 12.1.3 の注意事項に留意してください。





### 12.4.2.2 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 (CECRCR1)、受信コントロールレジスタ 2 (CECRCR2)、受信コントロールレジスタ 3 (CECRCR3)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

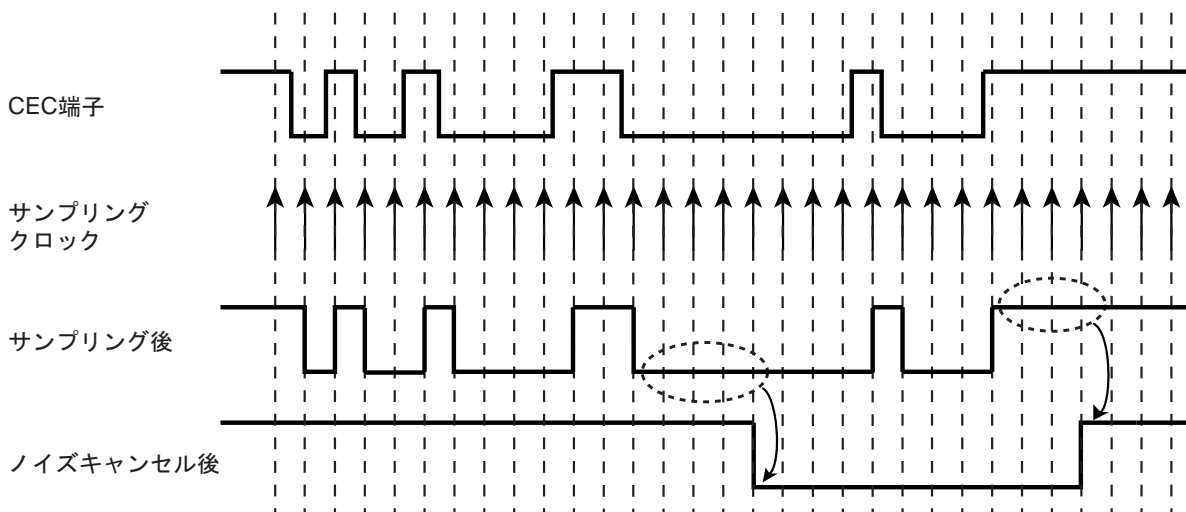
CECRCR1 レジスタの<CECHNC>、<CECLNC>でノイズキャンセル時間を設定します。設定されたサンプリング回数分の状態が観測されなければノイズとみなされます。"High"観測用、"Low"観測用を個別に設定可能です。

- 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ごとにサンプリング後の信号を観測し、現在"High"であれば、<CECLNC>で設定されたサンプリング回数の"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、<CECHNC>で設定されたサンプリング回数の"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<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、"High"の状態から、"Low"が 3 回観測されたところで"Low"に変化し、"Low"の状態から、"High"が 3 回観測されたところで"High"に変化します。

<CECHNC[1:0]> = 10 (サンプリング3回)  
<CECLNC[2:0]> = 011 (サンプリング4回)



## (3) 周期判定時間

CECRCR1 レジスタの<CECMIN>, <CECMAX>で周期違反の検出のための設定を行います。

1 ビット周期の最小値/最大値に対し、最小値については  $67/f_s$ (約 2.045ms)、最大値については  $90/f_s$ (約 2.747ms)を基準に、 $1/f_s$  単位で $-4/f_s$  から $+3/f_s$  の間で周期違反の検出を行うことができます。

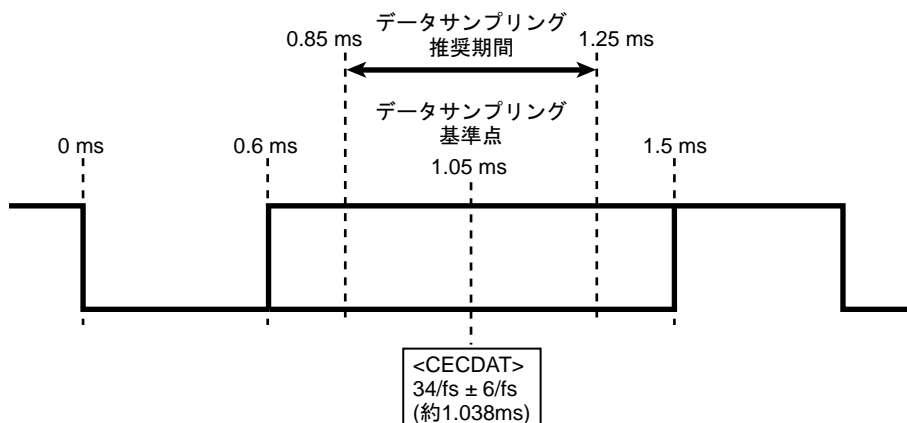
違反が検出された場合には割り込みが発生し、スタートビット待ちになります。割り込み発生までに受信したデータは破棄されます。

## (4) データ判別タイミングの設定

CECRCR1<CECDAT>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  $34/f_s$ (約 1.038ms)を基準に、 $2/f_s$  単位で $\pm 6/f_s$  まで設定可能です。

## 規格推奨のデータサンプリングタイミング



## (5) ACK 応答

CECRCR1<CECACKDIS>で、ディスティネーションアドレスが CECADD 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。

ヘッダブロックに対しては、<CECACKDIS>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。

以下に ACK 応答動作についてまとめます。

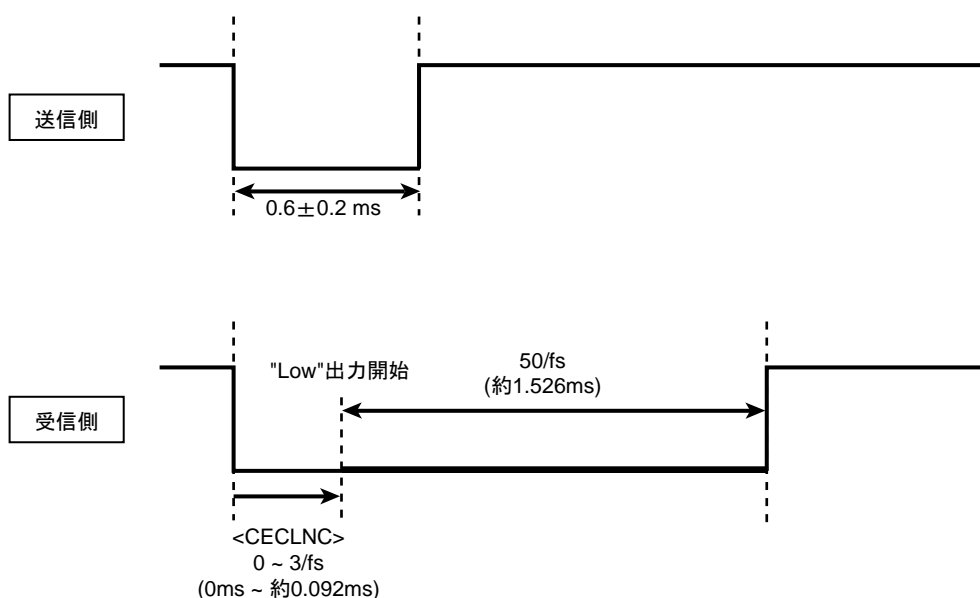
以下の表で、「○」は送信デバイスの ACK 送信に対し"0"出力をする(ACK ビットは論理"0"となる)こと、「×」は"0"出力をしない(ACK ビットは論理"1"となる)ことを示します。

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 <CEACKDIS>	"0" (論理0 応答する)	0	×	0	×
	"1" (論理0 応答しない)	0	×	×	×

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 ビットの立ち下がりを検出すると約 1.526ms 間"Low"を出力します。立ち下りの検出までの時間は、"Low"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) CECR1<CECLNC>は、必ず CECTCR<CEDTRS>と同じ設定で使用してください。



#### (6) 受信エラー割り込みの保留

CECR1<CECRIHLD>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。

<CECRIHLD>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<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<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割り込み保留の設定(CECR1<CECRIHLD>)が"1"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<CECOTH>で、ディスティネーションアドレスが CECADD 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>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。

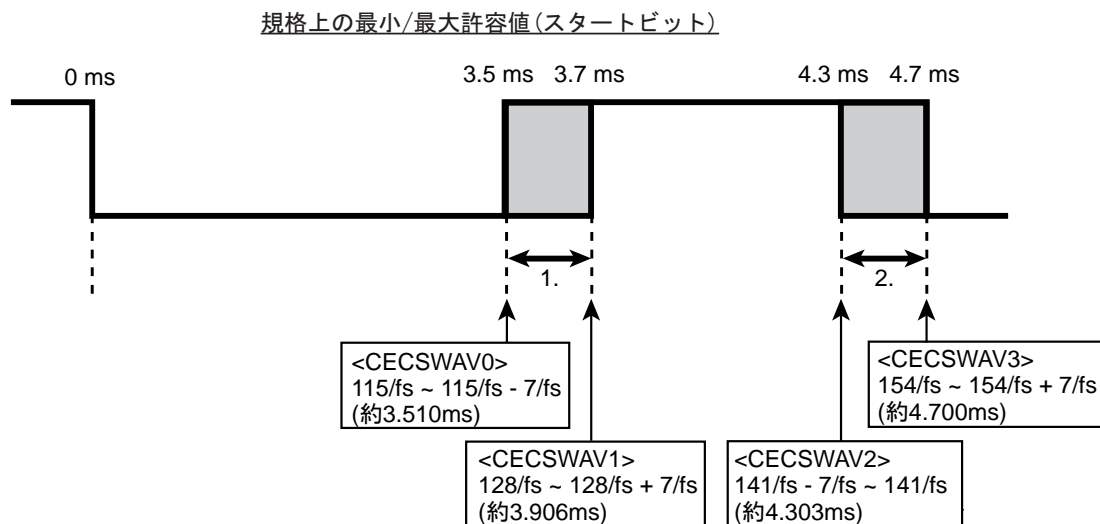
### (9) スタートビット検出の設定

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>で立ち上がりの最も早いタイミング、<CECSWAV1>で立ち上がりの最も遅いタイミングを設定します。(下図の 1.の期間)。

<CECSWAV2>で周期の最も早いタイミング、<CECSWAV3>で周期の最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.の期間に立ち下がりが検出されると、有効なスタートビットとみなします。



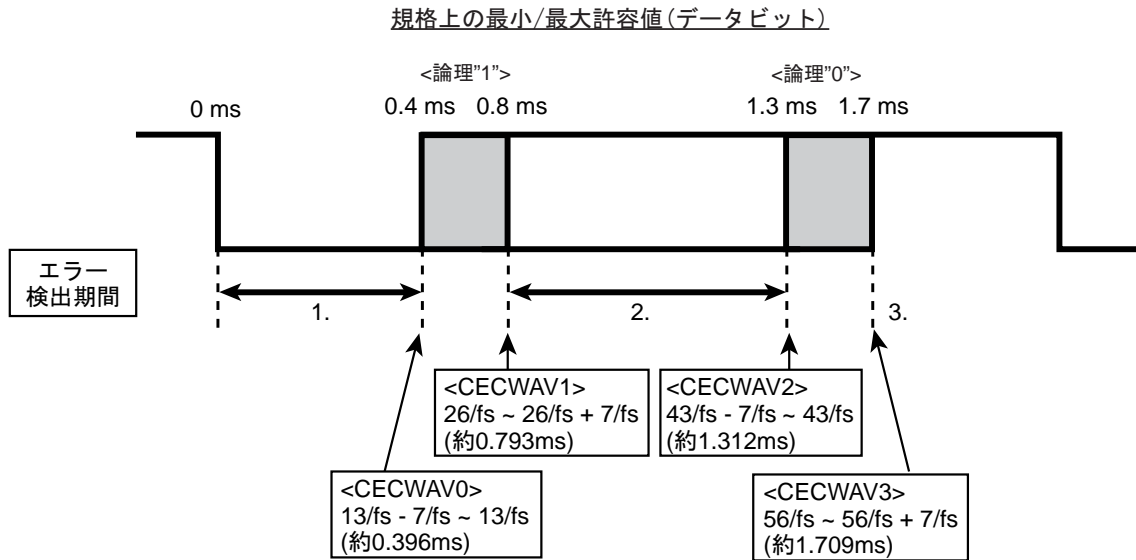
### (10) 波形エラー検出の設定

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<CECWAVEN>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>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がなかった場合に波形エラー割り込みが発生します。

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"の最も遅い立ち上がりタイミングと論理"0"の最も早い立ち上がりタイミングの間
3. 論理"0"の最も遅い立ち上がりタイミング



### 12.4.2.3 受信許可

CECADD, CECRCR1, CECRCR2, CECRCR3 レジスタの設定終了後、CECREN<CECREN>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, CECRCR1, CECRCR2, CECRCR3 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<CECREN>で受信禁止の設定を行い、<CECREN>および CECTEN<CECTRANS>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。

レジスタ名	Bit Symbol	設定項目
CECADD	<CECADD[15:0]>	ロジカルアドレス
CECRCR1	<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
CECRCR2	<CECSWAV0><CECSWAV1> <CECSWAV2><CECSWAV3>	スタートビット検出設定
CECRCR3 (波形エラー有効の場合)	<CECWAV0><CECWAV1> <CECWAV2><CECWAV3>	波形確認設定

### 12.4.2.4 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できます。

### 12.4.2.5 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ち下がりから次のビット開始の信号の立ち下がりの周期を測定し、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<CECMIN>, <CECMAX>で行います。周期の最小値/最大値に対し、最小値については  $67/f_s$ (約 2.045ms)、最大値については  $90/f_s$ (約 2.747ms) を基準に、 $1/f_s$  単位で  $-4/f_s$  から  $+3/f_s$  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<CECRIMIN>または<CECRIMAX>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 3.6 3ms 間"Low"を出力します。

- 注 1) 周期最小値違反の際、"Low"検出ノイズキャンセル時間後から"Low"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 12.1.3 を参照してください

## (2) ACK 衝突

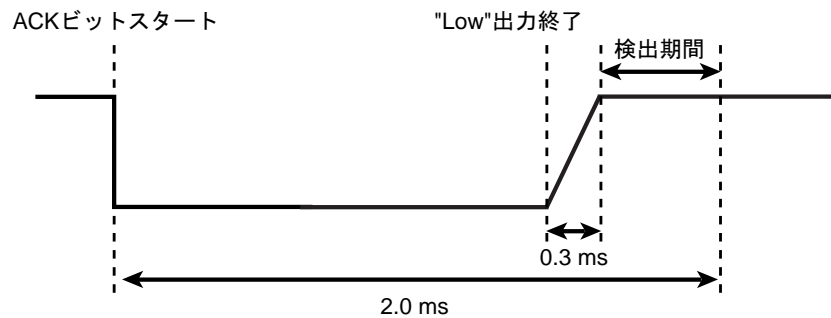
ACK 応答時、"Low"出力期間後に CEC 信号が"Low"であることを検出すると、ACK 衝突割り込みまたは周期最小値違反の割り込みが発生します。

ACK 衝突割り込みが発生すると、CECRSTAT<CECRIACK>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<CECRIMIN>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。

"Low"出力期間終了から約 0.3 ms 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(立ち下がり)から約 2.0 ms のところです。

"Low"出力期間終了から 0.3 ms のポイントで CEC 信号が"Low"であるかどうかを確認し、"Low"の場合には ACK 衝突割り込みが発生します。このポイントで CEC 信号が"High"で、検出期間内に"Low"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 3.63 ms 間"Low"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<CECRIOR>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<CECRIWAV>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<CECRIHLD>で行います。また、この設定を使用するときには、CECRCR1<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を発生します。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を発生します。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。

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を発生し、CEC 信号に約 3.63 ms 間"Low"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。

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。

#### 12.4.2.6 受信の停止

CECREN<CECREN>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。

注) 周期最小値の違反に対する"Low"出力中に受信禁止設定を行うと"Low"出力も停止します。



## 12.4.3 送信

### 12.4.3.1 基本動作

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 信号の立下りがないこと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<CECTISTA>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。

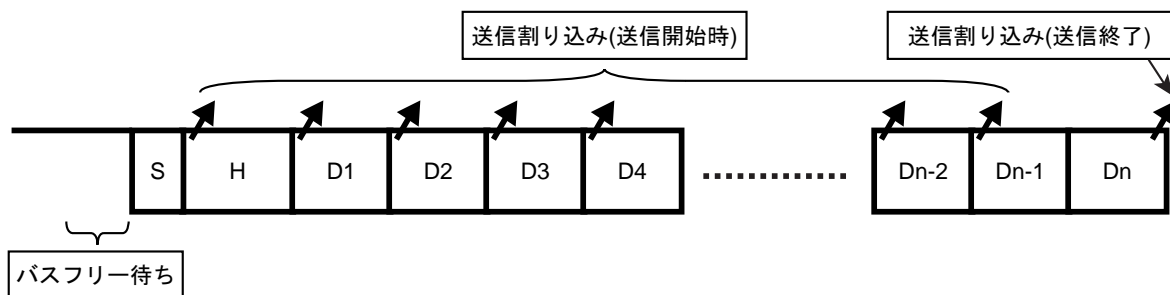
EOM ビットに"1"が設定されるまで同様にデータ転送を行います。

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<CECTIEND>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<CECTEN>はクリアされます。

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。



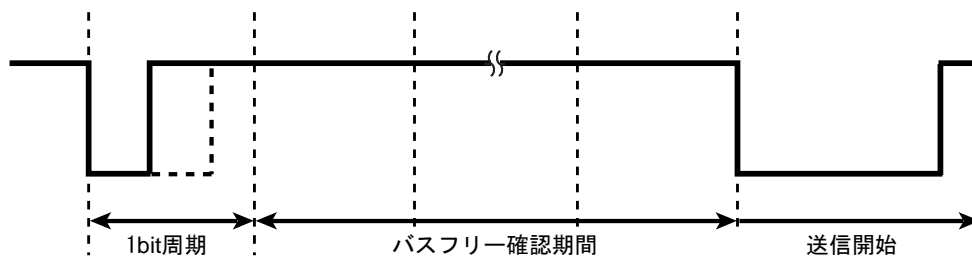
### 12.4.3.2 送信の準備

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<CECFREE>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ち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

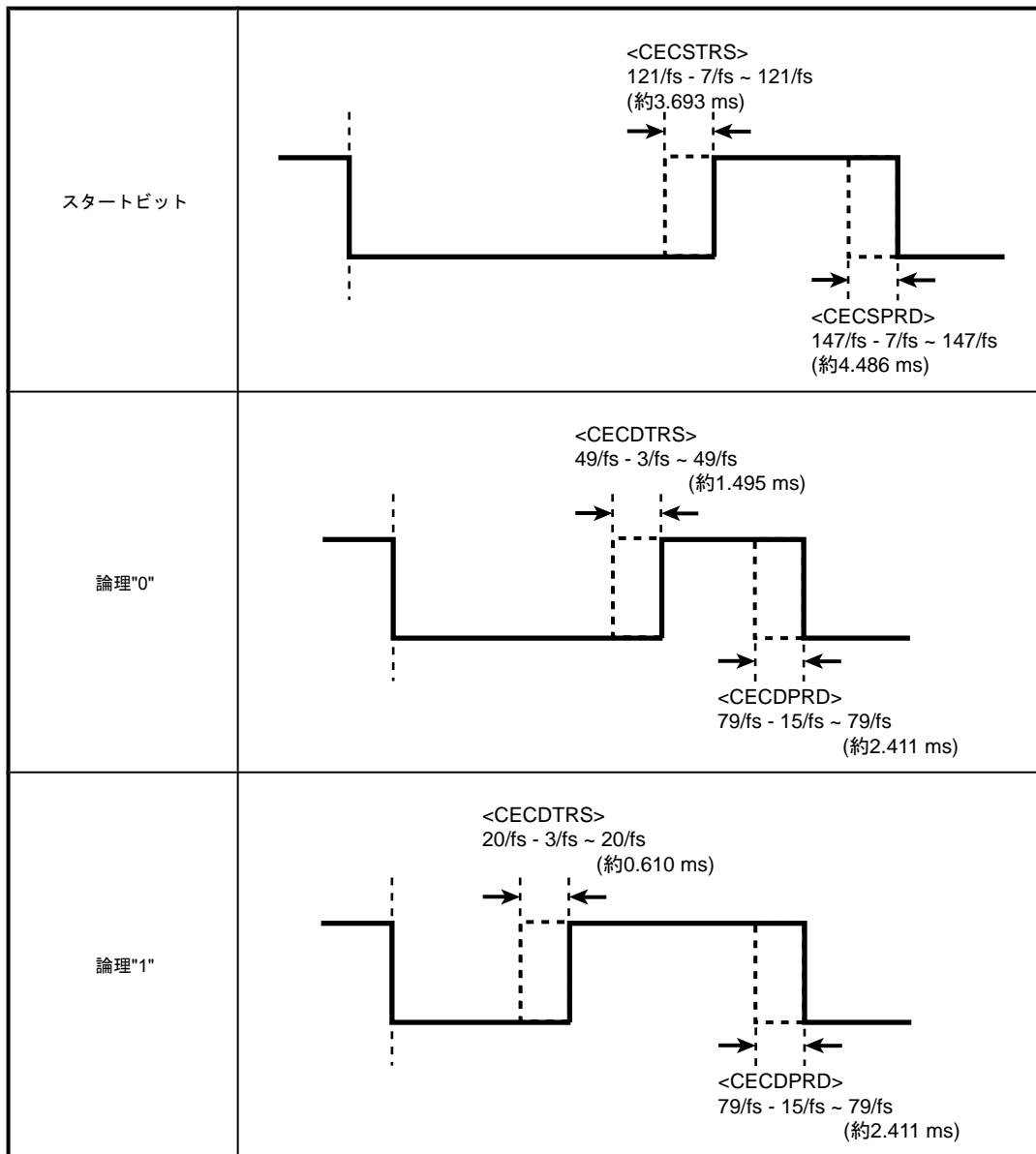
ブロードキャストメッセージの送信を行う場合は、CECTCR<CECBRD>を設定します。このビットがセットされているときには、ACK サイクルで論理"0"の応答があるとエラーになります。このビットがセットされていないときは、ACK サイクルで論理"1"の応答があるとエラーになります。

#### (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>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できます。

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 EOM のデータを CECTBUF レジスタに設定します。

### 12.4.3.3 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<CECTEN>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できます。

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。

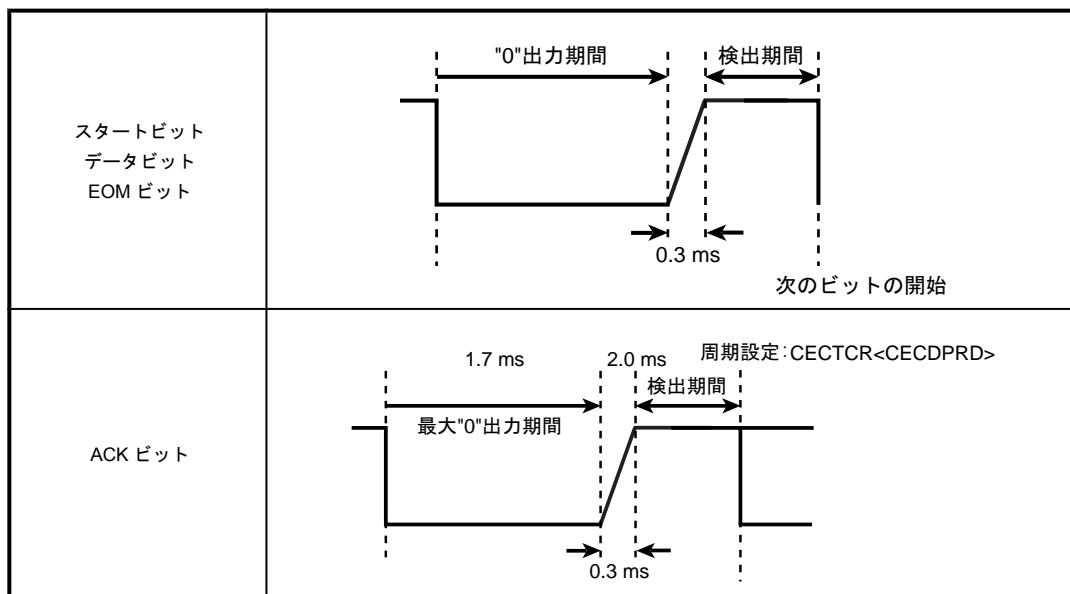
### 12.4.3.4 送信エラー詳細

#### (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"Low"出力していない状態、または ACK 応答後、CEC 信号に"Low"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<CECTIAL>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<CECBRD>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<CECTIACK>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<CECTIUR>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。

**12.4.3.5 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。

**12.4.3.6 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。

#### 12.4.4 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できます。

CECRESET<CECRESET>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 レジスタ以外のすべ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ので注意が必要です。

## 第 13 章 リモコン判定機能(RMC)

### 13.1 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。

#### 13.1.1 リモコン受信

- ・ 32.768 kHz 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
- ・ リーダ検出
- ・ 最大 72bit まで一括受信

### 13.2 ブロック図

図 13-1 にリモコン判定機能のブロック図を示します。

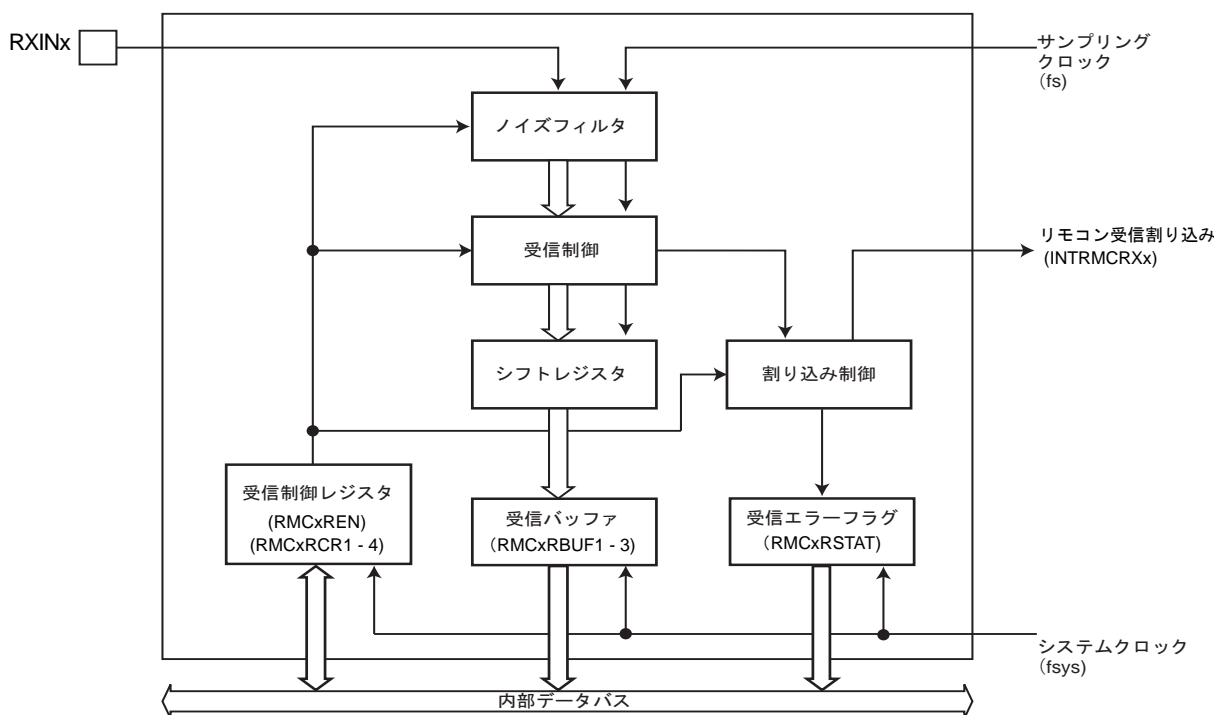


図 13-1 リモコン判定機能ブロック図

## 13.3 レジスタ説明

### 13.3.1 レジスタ一覧

リモコン判定機能の制御レジスタとアドレスは以下のとおりです。

Channel x	Base Address
Channel0	0x4004_0400
Channel1	0x4004_0440

レジスタ名(x=0~1)		Address(Base+)
リモコンイネーブルレジスタ	RMCxEN	0x0000
受信イネーブルレジスタ	RMCxREN	0x0004
受信データバッファレジスタ 1	RMCxRBUF1	0x0008
受信データバッファレジスタ 2	RMCxRBUF2	0x000C
受信データバッファレジスタ 3	RMCxRBUF3	0x0010
受信コントロールレジスタ 1	RMCxRCR1	0x0014
受信コントロールレジスタ 2	RMCxRCR2	0x0018
受信コントロールレジスタ 3	RMCxRCR3	0x001C
受信コントロールレジスタ 4	RMCxRCR4	0x0020
受信ステータスレジスタ	RMCxRSTAT	0x0024
Reserved	-	0x0028
Reserved	-	0x002C
Reserved	-	0x0030

注) "Reserved"表記のアドレスにはアクセスしないでください。



13.3.2 RMCxEN(イネーブルレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	I2RMC	RMCEN
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-2	-	R	リードすると"0"が読めます。
1	I2RMC	R/W	IDLE 時の動作 0: 禁止 1: 許可 IDLE モード時の機能の動作を制御します。 IDLE モードでリモコン判定機能を使用するときには、"1"を設定してください。 このビットは<RMCEN>と同時に設定できます。
0	RMCEN	R/W	リモコン判定機能動作 0: 禁止 1: 許可 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、まずこのビットを許可に設定してください。 動作禁止の状態では、イネーブルレジスタを除くリモコン判定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いったん動作させた後に、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。

## 13.3.3 RMCxREN(受信イネーブルレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	RMCREN
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	RMCREN	R/W	受信 0: 禁止 1: 許可 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、RMCxRCR1, 2, 3 を設定した後に許可してください。

13.3.4 RMCxRBUF1(受信データバッファレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	RMCxRBUF(受信データ 31 ~ 24 ビット)							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	RMCxRBUF(受信データ 23 ~ 16 ビット)							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	RMCxRBUF(受信データ 15 ~ 8 ビット)							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RMCxRBUF(受信データ 7 ~ 0 ビット)							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	RMCxRBUF[31:0]	R	受信データ(31 ビット ~ 0 ビット) 受信した 4 バイト分のデータ(31 ビット ~ 0 ビット)が読めます。

13.3.5 RMCxRBUF2(受信データバッファレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	RMCxRBUF(受信データ 63 ~ 54 ビット)							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	RMCxRBUF(受信データ 55 ~ 48 ビット)							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	RMCxRBUF(受信データ 47 ~ 40 ビット)							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RMCxRBUF(受信データ 39 ~ 32 ビット)							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	RMCxRBUF[63:32]	R	受信データ(63 ビット ~ 32 ビット) 受信した 4 バイト分のデータ(63 ビット ~ 32 ビット)が読めます。

## 13.3.6 RMCxRBUF3(受信データバッファレジスタ 3)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RMCRBUF(受信データ 71 ~ 64 ビット)							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	RMCRBUF[71:64]	R	受信データ(71 ビット~64 ビット) 受信した 1 バイト分のデータ(71 ビット~64 ビット)が読めます。

注) 受信データは、最初に受信したビットがデータバッファレジスタの MSB 側に、最後に受信したビットが 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、ビットの重みが逆順のデータがレジスタへ格納されますので、ご注意ください。

13.3.7 RMCxRCR1(受信コントロールレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	RMCLCMAX							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	RMCLCMIN							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	RMCLLMAX							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RMCLLMIN							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-24	RMCLCMAX[7:0]	R/W	リーダ検出の周期期間の上限設定 上限期間の計算式 : $\langle \text{RMCLCMAX} \rangle \times 4/\text{fs}[\text{s}]$
23-16	RMCLCMIN[7:0]	R/W	リーダ検出の周期期間の下限設定 下限期間の計算式 : $\langle \text{RMCLCMIN} \rangle \times 4/\text{fs}[\text{s}]$
15-8	RMCLLMAX[7:0]	R/W	リーダ検出の Low 期間の上限設定 上限期間の計算式 : $\langle \text{RMCLLMAX} \rangle \times 4/\text{fs}[\text{s}]$
7-0	RMCLLMIN[7:0]	R/W	リーダ検出の Low 期間の下限設定 下限期間の計算式 : $\langle \text{RMCLLMIN} \rangle \times 4/\text{fs}[\text{s}]$ RMCxRCR2<RMCLD> = 1 のときは、Low 幅が設定値未満の場合データビットと判別します。

注) リーダ検出の設定では以下の関係式を守ってください。

リーダ種類	関係式
Low 幅 + High 幅	$\langle \text{RMCLCMAX}[7:0] \rangle > \langle \text{RMCLCMIN}[7:0] \rangle$ $\langle \text{RMCLLMAX}[7:0] \rangle > \langle \text{RMCLLMIN}[7:0] \rangle$ $\langle \text{RMCLCMIN}[7:0] \rangle > \langle \text{RMCLLMAX}[7:0] \rangle$
High 幅のみ	$\langle \text{RMCLCMAX}[7:0] \rangle > \langle \text{RMCLCMIN}[7:0] \rangle$ $\langle \text{RMCLLMAX}[7:0] \rangle = 0x00$ $\langle \text{RMCLLMIN}[7:0] \rangle = \text{don't care}$
リーダなし	$\langle \text{RMCLCMAX}[7:0] \rangle = 0x00$ $\langle \text{RMCLCMIN}[7:0] \rangle = \text{don't care}$ $\langle \text{RMCLLMAX}[7:0] \rangle = \text{don't care}$ $\langle \text{RMCLLMIN}[7:0] \rangle = \text{don't care}$

## 13.3.8 RMCxRCR2(受信コントロールレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	RMCLIEN	RMCEDIEN	-	-	-	-	RMCLD	RMCPHM
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	RMCLL							
リセット後	1	1	1	1	1	1	1	1
	7	6	5	4	3	2	1	0
bit symbol	RMCDMAX							
リセット後	1	1	1	1	1	1	1	1

Bit	Bit Symbol	Type	機能
31	RMCLIEN	R/W	リーダ検出割り込み発生許可 0: 割り込み発生しない 1: 割り込み発生する
30	RMCEDIEN	R/W	リモコン入力立ち下がりエッジ割り込み発生許可 0: 割り込み発生しない 1: 割り込み発生する
29-26	-	R	リードすると"0"が読めます。
25	RMCLD	R/W	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設定 0: 禁止 1: 許可
24	RMCPHM	R/W	位相方式のリモコン受信モードの設定 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 周期が固定の位相方式のリモコン信号の場合は、"1"を設定します。
23-16	-	R	リードすると"0"が読めます。
15-8	RMCLL[7:0]	R/W	Low 幅の検出による受信終了/割り込み発生タイミングの設定 0000_0000~1111_1110: <RMCLL> × 1/fs[s]で受信終了/割り込み発生 1111_1111: Low 幅検出で受信終了/割り込みの設定をしない
7-0	RMCDMAX[7:0]	R/W	データビットの周期 MAX で受信終了/割り込み発生設定 0000_0000~1111_1110: <RMCDMAX> × 1/fs[s]で受信終了/割り込み発生 1111_1111: データビットの周期 MAX で受信終了/割り込み発生しない

13.3.9 RMCxRCR3(受信コントロールレジスタ 3)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	RMCDATH						
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	RMCDATL						
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-15	-	R	リードすると"0"が読めます。
14-8	RMCDATH[6:0]	R/W	データビットの3値判定のしきい値上位設定 しきい値の計算式： $\langle \text{RMCDATH} \rangle \times 1/\text{fs}[\text{s}]$ 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。 データビット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。
7	-	R	リードすると"0"が読めます。
6-0	RMCDATL[6:0]	R/W	データビットの0/1判別および3値判定のしきい値下位設定 しきい値の計算式： $\langle \text{RMCDATL} \rangle \times 1/\text{fs}[\text{s}]$ データビットの0/1判定のしきい値および、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。 データビットの0/1判定の場合、測定結果がしきい値以上でデータ"1"、しきい値未満でデータ"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、データビット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"00"と判別します。

注) RMCxRCR2<RMCPHM> = "0"のとき、<RMCDATH[6:0]>は有効になりません。<RMCPHM> = "1"のときのみ有効となります。

## 13.3.10 RMCxRCR4(受信コントロールレジスタ 4)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RMCP0	-	-	-	RMCNC			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	RMCP0	R/W	リモコン入力信号の極性選択 0: 正極 1: 負極
6-4	-	R	リードすると"0"が読めます。
3-0	RMCNC[3:0]	R/W	ノイズ除去時間の設定 0000: ノイズ除去しない 0001~1111: ノイズ除去 ノイズ除去時間の計算式 : $\langle \text{RMCNC} \rangle \times 1/\text{fs}[\text{s}]$



13.3.11 RMCxRSTAT(受信ステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	RMCRlif	RMCLOIF	RMCDMAXIF	RMCEDIF	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RMCRldr	RMCrNUM						
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15	RMCRlif	R	割り込み要因フラグ 0: リーダ検出割り込みは発生していない 1: リーダ検出割り込みが発生
14	RMCLOIF	R	割り込み要因フラグ 0: Low 幅検出割り込みは発生していない 1: Low 幅検出割り込みが発生
13	RMCDMAXIF	R	割り込み要因フラグ 0: データビット周期 MAX 割り込みは発生していない 1: データビット周期 MAX 割り込みが発生
12	RMCEDIF	R	割り込み要因フラグ 0: 立ち下がりエッジ割り込みは発生していない 1: 立ち下がりエッジが発生
11-8	-	R	リードすると"0"が読めます。
7	RMCRldr	R	リーダー検出 0: リーダ検出なし 1: リーダ検出あり
6-0	RMCrNUM[6:0]	R	リモコン受信したデータビット数 000_0000: データビットなし(リーダーのみ) 000_0001~100_1000: 1~72bit 受信 100_1001~111_1111: 73bit 以上 リモコン受信したデータビット数を示します。データビット数は受信終了後に格納され、受信途中のビット数はモニタすることはできません。

注 1) このレジスタは割り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の条件を満たす波形が入力され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。

## 13.4 動作説明

### 13.4.1 リモコン受信

#### 13.4.1.1 サンプリングブロック

リモコン信号のサンプリングは、32.768kHz の低速クロック (fs)で行います。

#### 13.4.1.2 基本動作

リーダを検出すると、RMCxRSTAT<RMCRLDR>がセットされます。

この時、RMCxRCR2<RMCLIEN>を設定しておくこと、リーダ検出時にリーダ検出割り込みが発生します。リーダ割り込み発生時には、RMCxRSTAT<RMCRLIF>がセットされます。

リーダ検出後、データビットの 0/1 判定を順次行い、結果を RMCxRBUF1, 2, 3 レジスタに最大 72bit まで格納します。RMCxRCR2<RMCEDIEN>を設定しておくこと、データビットの立ち下がりエッジごとにリモコン入力立ち下がりエッジ割り込みが発生します。リモコン入力立ち下がりエッジ割り込み発生時には、RMCxRSTAT<RMCEIDIF>がセットされます。

受信動作はデータビット周期 MAX の検出および、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/割り込み発生となります。

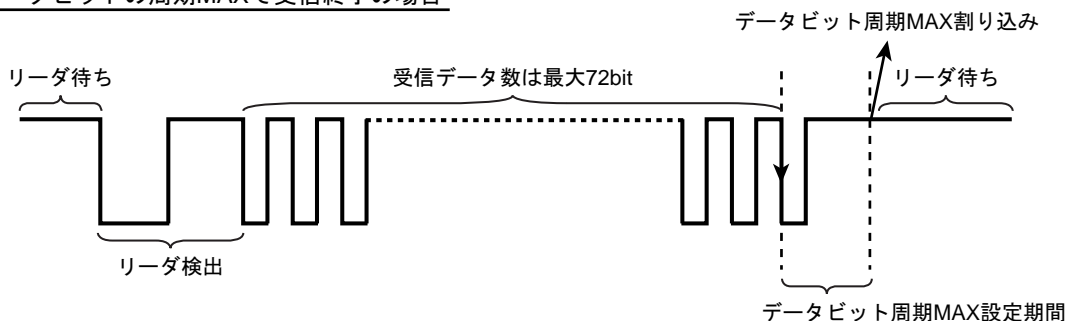
受信終了時の状態は、リモコン受信ステータスレジスタを読み出すことで知ることができます。

受信終了後、リーダ待ちとなります。

データビット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。

#### データビットの周期MAXで受信終了の場合



### 13.4.1.3 リモコン受信の準備

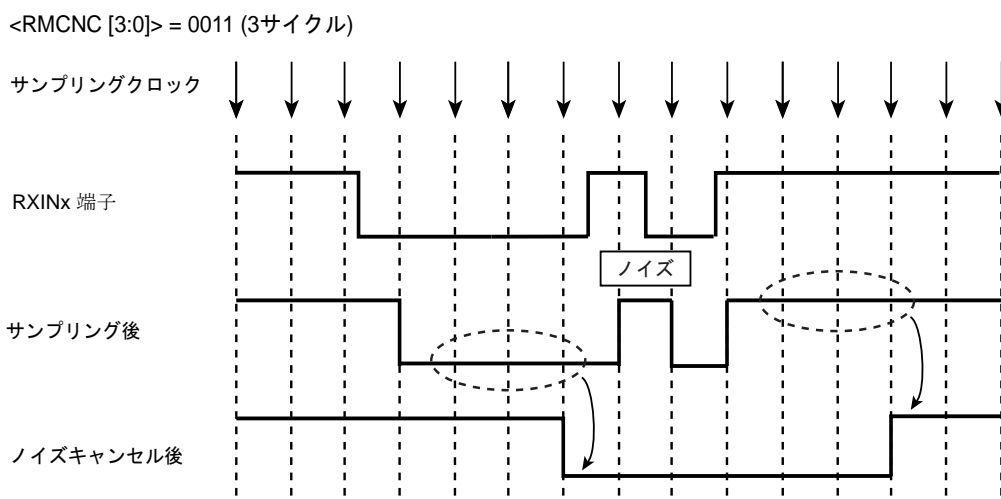
受信を開始する前に、リモコン受信コントロールレジスタ(RMCxRCR1, 2, 3, 4)でリモコン受信動作の設定を行います。

#### (1) ノイズキャンセル時間の設定

RMCxRCR4<RMCNC[3:0]>でノイズキャンセル時間を設定します。

ノイズキャンセルは、サンプリングクロックでリモコン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サンプリング後の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。

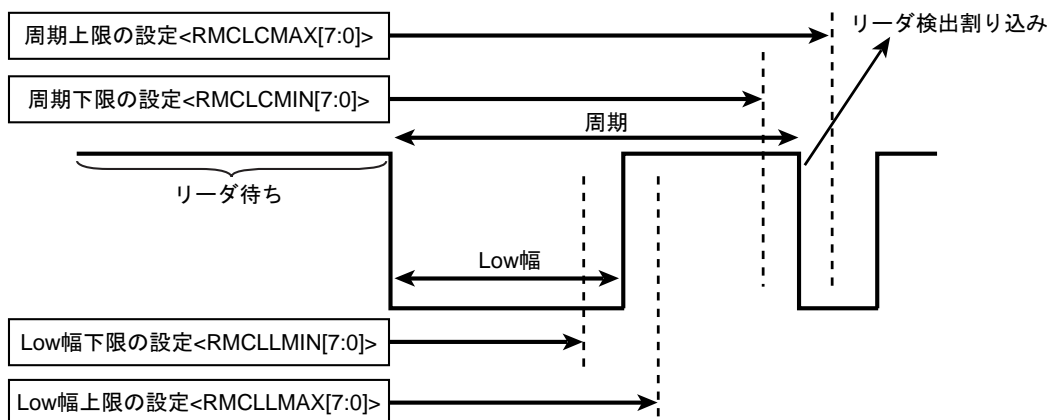


(2) リーダ検出の設定

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。RMCxRCR1 レジスタの <RMCLLMIN[7:0]>, <RMCLLMAX[7:0]>, <RMCLCMIN[7:0]>, <RMCLCMAX[7:0]> にリーダー検出のための設定を行います。RMCxRCR1 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]> = 0y00000000 <RMCLLMIN[7:0]> = don't care
リーダーなし	<RMCLCMAX[7:0]> = 0y00000000 <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 RMCxRCR1 の設定について表しています。



リーダー検出時に割り込みを発生させたい場合は、RMCxRCR2<RMCLIEN>で設定します。リーダーなしのリモコン信号については、リーダー検出割り込みの発生はできません。

### (3) データビットの 0/1 判定の設定

データビットの 0/1 判定は立ち下がりエッジの周期に対して行います。以下の 2 種類の判定方法があります。

#### 1. しきい値による判断

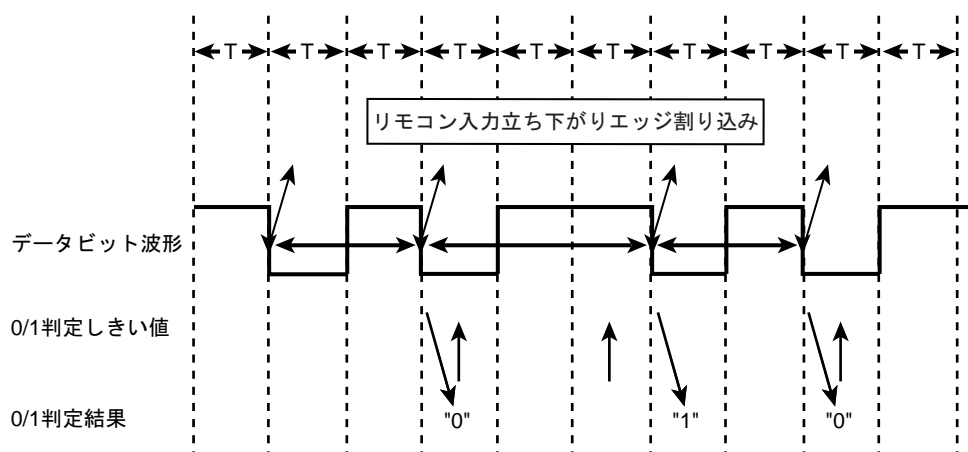
RMCxRRCR3<RMCDATL[6:0]>でデータビットの 0/1 判定のしきい値を設定します。0/1 判別はしきい値以上でデータ"1"、しきい値未満でデータ"0"となります。

#### 2. 立下りエッジ割り込みによる判断

RMCxRRCR2<RMCEDIEN>を"1"にセットすることで、データビットの立ち下がりエッジごとにリモコン入力立ち下がりエッジ割り込みを発生させることもできます。リモコン入力立ち下がりエッジ割り込みと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ができます。

次図はデータビットの 0/1 判定について表しています。

#### 0/1判定のしきい値を<RMCDATL[6:0]>ビットの設定で2.5T



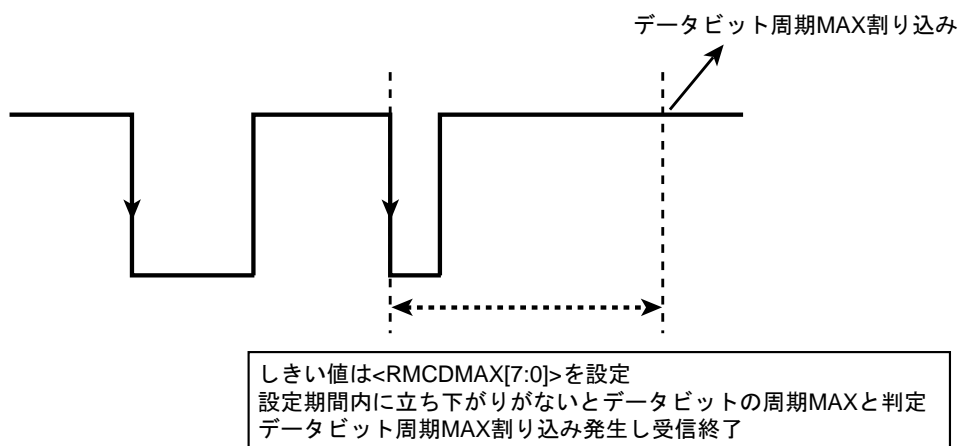
位相方式のリモコン信号のデータビットの判定は、「13.4.1.8 周期固定の位相方式のリモコン信号の受信」で説明します。

## (4) 受信終了の設定

受信終了はデータビットの周期 MAX 検出および、Low 幅の検出を設定することで行います。複数の要因を設定した場合は、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ください。

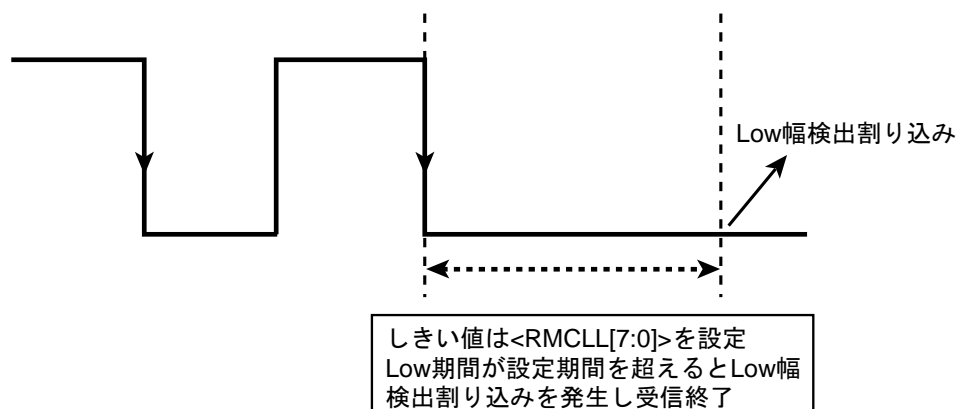
## 1. データビットの周期 MAX で受信終了する場合

RMCxRCR2<RMCDMAX[7:0]>でデータビット周期 MAX の設定をします。データビットの立ち下がりエッジ周期の値が、<RMCDMAX[7:0]>で設定されたデータビット周期 MAX のしきい値以上のときデータビット周期 MAX 検出で受信終了/割り込み発生となります。割り込み発生後に RMCxRSTAT<RMCDMAXIF>がセットされます。



## 2. Low 幅検出で受信終了する場合

RMCxRCR2<RMCLL[7:0]>で Low 幅の検出による受信終了の設定をします。データビットが立ち下がり後、設定期間を超えて Low のままだと Low 幅検出し、受信終了/割り込み発生となります。割り込み発生後に RMCxRSTAT<RMCLOIF>に"1"がセットされます。



### 13.4.1.4 受信許可

RMCxRCR1 レジスタ, RMCxRCR2 レジスタ, RMCxRCR3 レジスタ, RMCxRCR4 レジスタの設定終了後、RMCxREN<RMCREN>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 RMCxRCR1, 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。

### 13.4.1.5 受信の停止

RMCxREN<RMCREN>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。

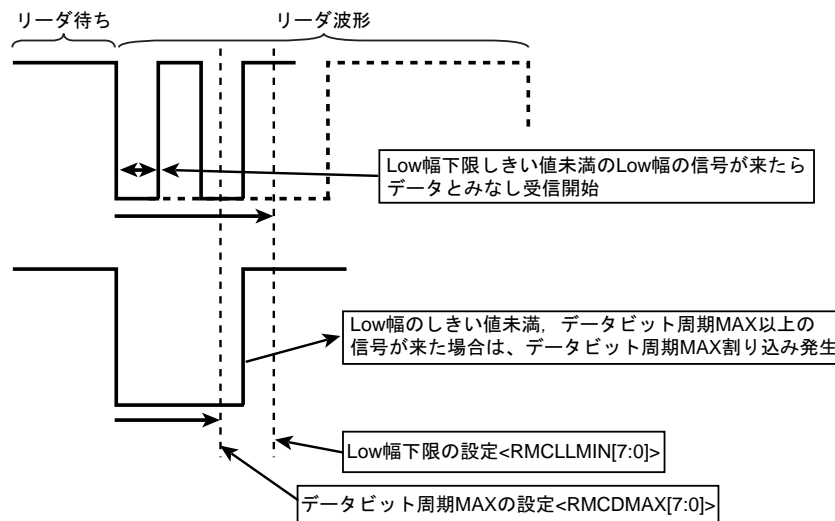
### 13.4.1.6 リーダ待ちの状態でリーダなしのリモコン信号の受信

RMCxRCR2<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。

RMCxRCR2<RMCLD>を設定すると、RMCxRCR1<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、最終ビットまで受信を行います。

この設定を使用する場合、データビットの 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すべ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。

RMCxRCR2<RMCLD> = 1



### 13.4.1.7 Low 幅のみのリーダで始まるリモコン信号の受信

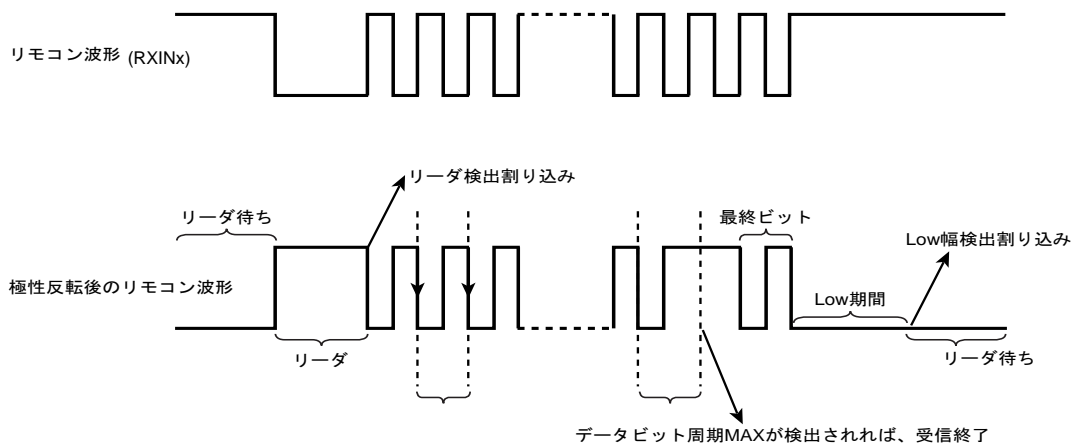
Low 幅のみのリーダで始まるリモコン信号は次図にあるように、リーダが Low 幅のみで始まり、データビットの周期が立ち上がり期間で構成される信号です。データビットの測定を立ち下がりエッジの周期で行う必要があるため、RMCxRCR4<RMCPO> = "1"に設定し、反転した信号にして入力します。

リーダの検出の設定は Low 幅のみのため、RMCxRCR1 の設定を、<RMCLLMAX[7:0]> = 0y0000\_0000, <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、RMCxRCR3<RMCDATL[6:0]>で 0/1 判別のしきい値を設定します。

RMCxRCR2 の<RMCDMAX[7:0]>でデータビットの周期 MAX の設定をします。

受信終了の設定は、RMCxRCR2 の<RMCDMAX[7:0]>でデータビットの周期 MAX の設定と <RMCLL[7:0]>で Low 幅検出の設定をします。データビットの周期 MAX 検出および、最終ビット後に設定された Low 期間確認後に受信終了/割り込み発生し、リーダ待ちとなります。



### 13.4.1.8 周期固定の位相方式のリモコン信号の受信

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ーンの判別を行い、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 RMCxRBUF1, 2, 3 に格納します。

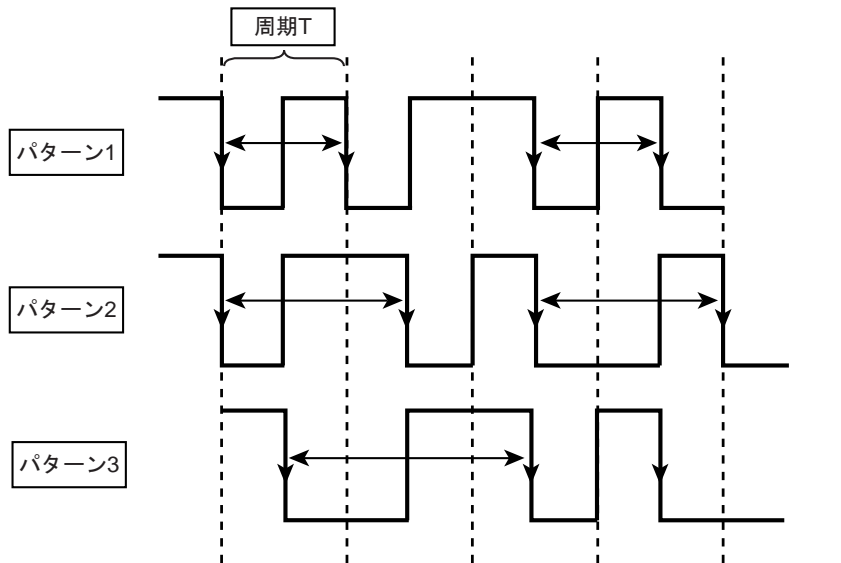
周期固定の位相方式のリモコン受信は、RMCxRCR2<RMCPHM> = "1"に設定することで有効になります。しきい値の設定は、RMCxRCR3<RMCDATL[6:0]>, <RMCDATH[6:0]>で行います。次図にある、3 通りのデータの変化のパターンに対して、2 つのしきい値を決めます。3 通りのパターンはそれぞれ周期 T に対して、1T, 1.5T, 2T となり、しきい値の設定は下表のようになります。



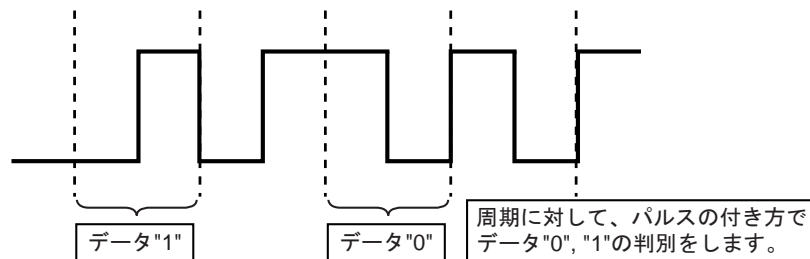
	判別内容	しきい値	設定レジスタ
しきい値 1	パターン 1 とパターン 2	1T~1.5T	RMCxRCR3<RMCDATL[6:0]>
しきい値 2	パターン 2 とパターン 3	1.5T~2T	RMCxRCR3<RMCDATH[6:0]>

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"11"で始まる必要があります。

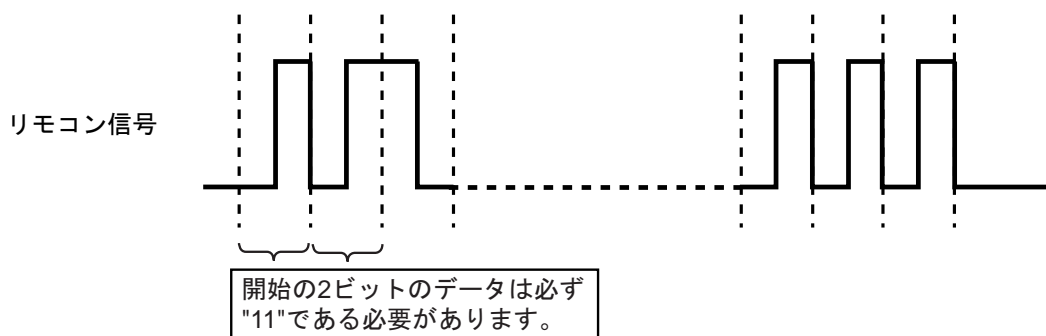
位相方式のリモコン信号の波形パターン



位相方式のリモコンデータ例



位相方式のリモコン信号について





## 第 14 章 アナログ/デジタルコンバータ(ADC)

### 14.1 概要

TMPM330FDFG/FYFG/FWFG 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 (ADC) を内蔵しています。

12 チャンネルのアナログ入力端子 (AIN0~AIN11) は、入力ポートと兼用です。

- 注 1) 変換精度を保証するために、ADCBAS レジスタに必ず指定された値を設定してください。
- 注 2) IDLE,STOP モードにより電源電流を低減させる場合、以下の条件で使用する時には、AD コンバータの動作を停止して、スタンバイモードに遷移する命令を実行してください。
  - 1.ADMOD1<I2AD> = "0"で IDLE モードへ遷移する場合
  - 2.STOP モードへ遷移する場合

## 14.2 構成

図 14-1 に AD コンバータのブロック図を示します。

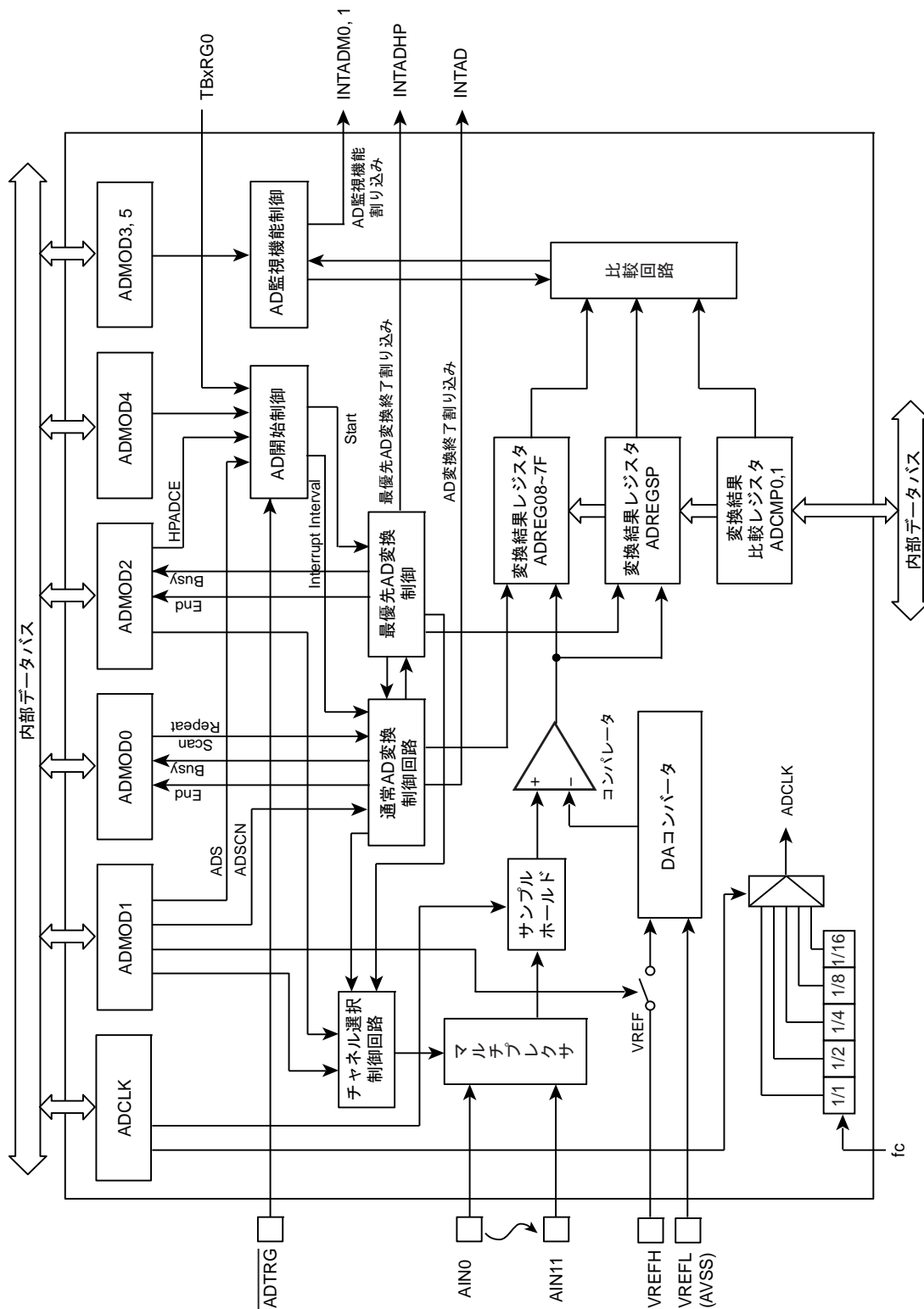


図 14-1 AD コンバータのブロック図

## 14.3 レジスタ説明

### 14.3.1 レジスタ一覧

AD 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。

AD コンバータは、モードコントロールレジスタ (ADMOD0 ~ ADMOD5) により制御されています。変換結果は、変換結果レジスタ ADREG08 ~ ADREG7F の 8 個のレジスタに格納されます。また、最優先変換結果は ADREGSP に格納されます。

変換精度を保証するために、ADCBAS レジスタに必ず指定された値を設定してください。

Base Address = 0x4003\_0000

レジスタ名		Address(Base+)
変換クロック設定レジスタ	ADCLK	0x0000
モードコントロールレジスタ 0	ADMOD0	0x0004
モードコントロールレジスタ 1	ADMOD1	0x0008
モードコントロールレジスタ 2	ADMOD2	0x000C
モードコントロールレジスタ 3	ADMOD3	0x0010
モードコントロールレジスタ 4	ADMOD4	0x0014
モードコントロールレジスタ 5	ADMOD5	0x0018
変換精度設定レジスタ	ADCBAS	0x0020
Reserved	-	0x0024
Reserved	-	0x0028
変換結果レジスタ 08	ADREG08	0x0030
変換結果レジスタ 19	ADREG19	0x0034
変換結果レジスタ 2A	ADREG2A	0x0038
変換結果レジスタ 3B	ADREG3B	0x003C
変換結果レジスタ 4C	ADREG4C	0x0040
変換結果レジスタ 5D	ADREG5D	0x0044
変換結果レジスタ 6E	ADREG6E	0x0048
変換結果レジスタ 7F	ADREG7F	0x004C
変換結果レジスタ SP	ADREGSP	0x0050
変換結果比較レジスタ 0	ADCMP0	0x0054
変換結果比較レジスタ 1	ADCMP1	0x0058

注) "Reserved"表記のアドレスにはアクセスしないでください。

## 14.3.2 ADCBAS (変換精度設定レジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADCBAS							
リセット後	0	0	1	1	1	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	ADCBAS[7:0]	R/W	"0x58"をライトしてください。

注) 変換精度を保证するために、ADCBAS レジスタに必ず指定された値(0x0000\_0058)を設定してください。

14.3.3 ADCLK (変換クロック設定レジスタ)

	31	30	29	28	27	26	25	24	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	23	22	21	20	19	18	17	16	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	15	14	13	12	11	10	9	8	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	7	6	5	4	3	2	1	0	
bit symbol	TSH				-	ADCLK			
リセット後	1	0	0	0	0	0	0	0	

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-4	TSH[3:0]	R/W	AD サンプルホールド時間選択 1000: 8 変換クロック 1001: 16 変換クロック 1010: 24 変換クロック 1011: 32 変換クロック 0011: 64 変換クロック 1100: 128 変換クロック 1101: 512 変換クロック 上記以外の設定: Reserved
3	-	R	リードすると"0"が読めます。
2-0	ADCLK[2:0]	R/W	AD 変換クロック選択 000: fc 001: fc/2 010: fc/4 011: fc/8 100: fc/16 111: Reserved

変換にかかるクロック数は最短で 46 クロックです。

以下にサンプルホールド時間、変換時間の例を示します。

(fc = 40MHz の場合)

<TSH[3:0]>	サンプルホールド 時間	変換時間(<ADCLK[2:0]>設定)				
		000 (fc)	001 (fc/2)	010 (fc/4)	011 (fc/8)	100 (fc/16)
1000 (8 変換クロック)	0.2 μs	1.15 μs	2.3 μs	4.6 μs	9.2μs	18.4μs
1001 (16 変換クロック)	0.4 μs	1.35 μs	2.7μs	5.4μs	10.8μs	21.6μs
1010 (24 変換クロック)	0.6 μs	1.55 μs	3.1μs	6.2μs	12.4μs	24.8μs
1011 (32 変換クロック)	0.8 μs	1.75 μs	3.5μs	7.0μs	14.0μs	28.0μs
0011 (64 変換クロック)	1.6 μs	2.55 μs	5.1μs	10.2μs	20.4μs	40.8μs
1100 (128 変換クロック)	3.2 μs	4.15 μs	8.3μs	16.6μs	33.2μs	66.4μs
1101 (512 変換クロック)	12.8 μs	13.75 μs	27.5μs	55.0μs	110.0μs	220.0μs

注) AD 変換中に、AD 変換クロック設定を変更しないで下さい。



14.3.4 ADMOD0 (モードコントロールレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	EOCFN	ADBFN	-	ITM		REPEAT	SCAN	ADS
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	EOCFN	R	通常 AD 変換終了フラグ(注 1) 0: 変換前または変換中 1: 終了
6	ADBFN	R	通常 AD 変換 BUSY フラグ 0: 変換停止 1: 変換中
5	-	R	リードすると"0"が読めます。
4-3	ITM[1:0]	R/W	チャンネル固定リピート変換モード時の割り込み指定(下表および注 2 参照)
2	REPEAT	R/W	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
1	SCAN	R/W	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
0	ADS	R/W	AD 変換スタート(注 3) 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 AD 変換割り込み指定

<ITM[1:0]>	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

注 1) ADMOD0 レジスタを読み出すと"0"にクリアされます。

注 2) チャンネル固定リピートモード(<REPEAT> = "1"、<SCAN> = "0")のときのみ有効です

注 3) モード設定を行ってから変換を開始してください。

## 14.3.5 ADMOD1 (モードコントロールレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VREFON	I2AD	ADSCN	-	ADCH			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	VREFON	R/W	VREF 印加制御(注 1、注 2) 0: OFF 1: ON
6	I2AD	R/W	IDLE モード時の動作設定 0: 停止 1: 動作
5	ADSCN	R/W	チャンネルスキャンモード時の動作設定 0: 4ch スキャン 1: 8ch スキャン
4	-	R/W	0 をライトしてください。
3-0	ADCH[3:0]	R/W	アナログ入力チャンネル選択(下表参照)

## アナログ入力チャネル選択

ADMOD0<SCAN> ADMOD1<ADCH[3:0]>	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	AIN0	AIN0	AIN0
0001	AIN1	AIN0~AIN1	AIN0~AIN1
0010	AIN2	AIN0~AIN2	AIN0~AIN2
0011	AIN3	AIN0~AIN3	AIN0~AIN3
0100	AIN4	AIN4	AIN0~AIN4
0101	AIN5	AIN4~AIN5	AIN0~AIN5
0110	AIN6	AIN4~AIN6	AIN0~AIN6
0111	AIN7	AIN4~AIN7	AIN0~AIN7
1000	AIN8	AIN8	AIN8
1001	AIN9	AIN8~AIN9	AIN8~AIN9
1010	AIN10	AIN8~AIN10	AIN8~AIN10
1011	AIN11	AIN8~AIN11	AIN8~AIN11
1100	設定禁止		
1101			
1110			
1111			

注 1) AD 変換をスタートさせる場合は、必ず<VREFON>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>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。

14.3.6 ADMOD2 (モードコントロールレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	EOCFHP	ADBFHP	HPADCE	-	HPADCH			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	EOCFHP	R	最優先 AD 変換終了フラグ(注 1) 0: 変換前または変換中 1: 終了
6	ADBFHP	R	最優先 AD 変換 BUSY フラグ 0: 変換停止 1: 変換中
5	HPADCE	R/W	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 が読めます。
4	-	R/W	0 をライトしてください。
3-0	HPADCH[3:0]	R/W	最優先変換起動時のアナログ入力チャンネル選択(下表参照)

<HPADCH[3:0]>	最優先変換時のアナログ入力チャンネル
0000	AIN0
0001	AIN1
0010	AIN2
0011	AIN3
0100	AIN4
0101	AIN5
0110	AIN6
0111	AIN7
1000	AIN8
1001	AIN9
1010	AIN10
1011	AIN11
1100	設定禁止
1101	
1110	
1111	

注 1) ADMOD2 レジスタを読み出すと"0"にクリアされます。

## 14.3.7 ADMOD4 (モードコントロールレジスタ 4)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	HADHS	HADHTG	ADHS	ADHTG	-	-	ADRST	
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	HADHS	R/W	最優先 AD 変換のハードウェア起動ソース選択 0: 外部トリガ 1: タイマレジスタ 0 一致(TB5RG0)
6	HADHTG	R/W	最優先 AD 変換のハードウェア起動 0: 無効 1: 有効
5	ADHS	R/W	通常 AD 変換のハードウェア起動ソース(注 1) 0: 外部トリガ 1: タイマレジスタ 0 一致(TB6RG0)
4	ADHTG	R/W	通常 AD 変換のハードウェア起動 0: 無効 1: 有効
3-2	-	R	リードすると"0"が読めます。
1-0	ADRST[1:0]	W	10 → 01 のライトで ADC をソフトウェアリセットします。(注 2)

注 1) 最優先 AD 変換のハードウェア起動ソースに外部トリガを使用しているときは、通常 AD 変換のハードウェア起動ソースとして外部トリガを設定できません。

注 2) ソフトウェアリセットをかけると、ADCLK<ADCLK>以外のレジスタの内容はすべて初期化されます。

注) 本製品にはハードウェア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。

## 14.3.8 ADMOD3 (AD モードコントロールレジスタ 3)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	ADOBIC0	ADREGS0				ADOBSV0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	-	R/W	0 をライトしてください。
6	-	R	リードすると"0"が読めます。
5	ADOBIC0	R/W	AD 監視機能割り込み 0 の設定 0: 変換結果レジスタの値が、変換結果比較レジスタ 0 の値より小さい場合割り込み発生 1: 変換結果レジスタの値が、変換結果比較レジスタ 0 の値より大きい場合割り込み発生
4-1	ADREGS0[3:0]	R/W	AD 監視機能 0 使用時に、比較対象とする変換結果レジスタの選択(下表参照)
0	ADOBSV0	R/W	AD 監視機能 0 0: 無効 1: 有効

<ADREGS0[3:0]>	比較される AD 変換結果レジスタ	<ADREGS0[3:0]>	比較される AD 変換結果レジスタ
0000	ADREG08	0100	ADREG4C
0001	ADREG19	0101	ADREG5D
0010	ADREG2A	0110	ADREG6E
0011	ADREG3B	0111	ADREG7F
-	-	1xxx	ADREGSP

14.3.9 ADMOD5 (AD モードコントロールレジスタ 5)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	ADOBIC1	ADREGS1				ADOBSV1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-6	-	R	リードすると"0"が読めます。
5	ADOBIC1	R/W	AD 監視機能割り込み 1 の設定 0: 変換結果レジスタの値が、変換結果 比較レジスタ 1 の値より小さい場合割り込み発生 1: 変換結果レジスタの値が、変換結果比較レジスタ 1 の値より大きい場合割り込み発生
4-1	ADREGS1[3:0]	R/W	AD 監視機能 1 使用時に、比較対象とする変換結果レジスタの選択(下表参照)
0	ADOBSV1	R/W	AD 監視機能 1 0: 無効 1: 有効

<ADREGS1[3:0]>	比較される AD 変換結果レジスタ	<ADREGS1[3:0]>	比較される AD 変換結果レジスタ
0000	ADREG08	0100	ADREG4C
0001	ADREG19	0101	ADREG5D
0010	ADREG2A	0110	ADREG6E
0011	ADREG3B	0111	ADREG7F
-	-	1xxx	ADREGSP

## 14.3.10 ADREG08 (変換結果レジスタ 08)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADRO							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADRO		-	-	-	-	OVR0	ADR0RF
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADRO[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR0	R	オーバーランフラグ 0: 発生していない 1: 発生した ＜ADRO＞を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR0RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます

注) 本レジスタはハーフワードまたはワードアクセスしてください。



14.3.11 ADREG19 (変換結果レジスタ 19)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADR1							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR1		-	-	-	-	OVR1	ADR1RF
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADR1[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR1	R	オーバーランフラグ 0: 発生していない 1: 発生した ＜ADR1＞を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR1RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。

## 14.3.12 ADREG2A (変換結果レジスタ 2A)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADR2							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR2		-	-	-	-	OVR2	ADR2RF
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADR2[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR2	R	オーバーランフラグ 0: 発生していない 1: 発生した <ADR2>を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR2RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。

14.3.13 ADREG3B (変換結果レジスタ 3B)

	31	30	29	28	27	26	25	24	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	23	22	21	20	19	18	17	16	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	15	14	13	12	11	10	9	8	
bit symbol	ADR3								
リセット後	0	0	0	0	0	0	0	0	
	7	6	5	4	3	2	1	0	
bit symbol	ADR3			-	-	-	-	OVR3	ADR3RF
リセット後	0	0	0	0	0	0	0	0	

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADR3[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャンネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR3	R	オーバーランフラグ 0: 発生していない 1: 発生した <ADR3>を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR3RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。

## 14.3.14 ADREG4C (変換結果レジスタ 4C)

	31	30	29	28	27	26	25	24	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	23	22	21	20	19	18	17	16	
bit symbol	-	-	-	-	-	-	-	-	
リセット後	0	0	0	0	0	0	0	0	
	15	14	13	12	11	10	9	8	
bit symbol	ADR4								
リセット後	0	0	0	0	0	0	0	0	
	7	6	5	4	3	2	1	0	
bit symbol	ADR4			-	-	-	-	OVR4	ADR4RF
リセット後	0	0	0	0	0	0	0	0	

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADR4[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャンネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR4	R	オーバーランフラグ 0: 発生していない 1: 発生した <ADR4>を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR4RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。

## 14.3.15 ADREG5D (変換結果レジスタ 5D)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADR5							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR5		-	-	-	-	OVR5	ADR5RF
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADR5[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャンネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR5	R	オーバーランフラグ 0: 発生していない 1: 発生した <ADR5>を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR5RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。

## 14.3.16 ADREG6E (変換結果レジスタ 6E)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADR6							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR6		-	-	-	-	OVR6	ADR6RF
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADR6[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャンネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR6	R	オーバーランフラグ 0: 発生していない 1: 発生した <ADR6>を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR6RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。

## 14.3.17 ADREG7F (変換結果レジスタ 7F)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADR7							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR7		-	-	-	-	OVR7	ADR7RF
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADR7[9:0]	R	AD 変換結果 変換結果が格納されます。変換チャンネルと結果レジスタの関係は 14.4.5.7 章の表 14-2 を参照してください。
5-2	-	R	リードすると"0"が読めます。
1	OVR7	R	オーバーランフラグ 0: 発生していない 1: 発生した <ADR7>を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADR7RF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。

## 14.3.18 ADREGSP (変換結果レジスタ SP)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADRSP							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADRSP		-	-	-	-	OVRSP	ADRSPRF
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADRSP[9:0]	R	AD 変換結果 最優先 AD 変換の結果が格納されます。
5-2	-	R	リードすると"0"が読めます。
1	OVRSP	R	オーバーランフラグ 0: 発生していない 1: 発生した <ADRSP>を読み出す前に変換結果が上書きされると"1"がセットされます。 このビットは読み出すと"0"にクリアされます。
0	ADRSPRF	R	AD 変換結果格納フラグ 0: 変換結果が格納されていない 1: 変換結果が格納された 変換結果が格納されると"1"がセットされます。 このビットは変換結果を読み出すと"0"にクリアされます。

注) 本レジスタはハーフワードまたはワードアクセスしてください。



## 14.3.19 ADCMP0 (変換結果比較レジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADCOM0							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADCOM0		-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADCOM0[9:0]	R/W	AD 監視機能 0 が有効な時、ADMOD3<ADREGS0>で指定された変換結果レジスタの値と比較する値を設定します。
5-0	-	R	リードすると"0"が読めます。

注) 本レジスタへの書き込みは、AD 監視機能 0 が無効な状態(ADMOD3<ADBSV0> = "0")で行ってください。

## 14.3.20 ADCMP1 (変換結果比較レジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ADCOM1							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADCOM1		-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めます。
15-6	ADCOM1[9:0]	R/W	AD 監視機能 1 が有効な時、ADMOD5<ADREGS1>で指定された変換結果レジスタの値と比較する値を設定します。
5-0	-	R	リードすると"0"が読めます。

注) 本レジスタへの書き込みは、AD 監視機能 1 が無効な状態(ADMOD5<ADBSV1>="0")で行ってください。

## 14.4 動作説明

### 14.4.1 アナログ基準電圧

アナログ基準電圧の"High"レベル側を VREFH 端子に、"Low"レベル側を VREFL 端子に印加します。

AD 変換をスタートさせる場合は、必ず<VREFON>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 待つてから、ADMOD0<ADS>に"1"を書き込んで変換を開始してください。

ADMOD1<VREFON>に"0"を書き込むことにより、VREFH - VREFL 間のスイッチを OFF できません。低消費電力モードに移行する場合、変換終了後に<VREFON>を"0"にしてください。

注) TMPM330FDFG/FYFG/FWFG では、VREFL を AVSS と共用しています。

### 14.4.2 AD 変換モード

AD 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 変換には 4 つの動作モードがあります。

#### 14.4.2.1 通常 AD 変換

通常 AD 変換には、次の 4 つの動作モードが用意されており、ADMOD0<REPEAT, SCAN>の設定により選択ができます。

- ・ 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) 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<EOCFN>が"1"、ADMOD0<ADBFN>が"0"となり、変換終了割り込み要求(INTAD)が発生します。<EOCFN>は読み出すことにより"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<EOCFN>が"1"にセット、ADMOD0<ADBFN>が"0"にされ、変換終了割り込み要求(INTAD)が発生します。<EOCFN>は読み出すことで 0 にクリアされます。

### (3) チャ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、ADMOD0<EOCFN>が"1"にセットされます。ADMOD0<ADBFN>は"0"にクリアされず"1"を保持します。変換終了割り込み要求(INTAD)発生タイミングはADMOD0<ITM>の設定により選択できます。<EOCFN>がセット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。

<EOCFN>は読み出すことにより0にクリアされます。

### (4) チャ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<EOCFN>が"1"にセットされ、変換終了割り込み要求(INTAD)が発生します。ADMOD0<ADBFN>は"0"にクリアされず"1"を保持します。<EOCFN>は読み出すことにより"0"にクリアされます。

#### 14.4.2.2 最優先 AD 変換

最優先 AD 変換は、通常 AD 変換に割り込んで AD 変換を行うことができます。

動作モードはチャ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>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、最優先変換終了割り込み要求(INTADHP)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。

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。

#### 14.4.3 AD 監視機能

AD 監視機能を2チャンネル持っています。

ADMOD3<ADOBSV0>, ADMOD5<ADOBSV1>に"1"をセットすると AD 監視機能が有効になり、ADMOD3<ADREGS0>, ADMOD5<ADREGS1>で指定された変換結果レジスタの内容が比較レジスタの値より大または小(ADMOD3<ADOBIC0>, ADMOD5<ADBIC1>で大か小は指定)になると AD 監視機能割り込み要求(INTADM0, INTADM1)が発生します。この比較動作は該当変換結果レジスタへ結果が格納されるごとに行われます。

AD 監視機能に割り当てている変換結果レジスタの変換結果を読み出さないで連続的に使用した場合、変換結果が上書きされ変換結果格納フラグ<ADR<sub>x</sub>RF>およびオーバーランフラグ<OVR<sub>x</sub>

#### 14.4.4 入力チャネルの選択

リセット後は ADMOD0<REPEAT,SCAN> は "00" に ADMOD1<ADCH[3:0]> は "0000" に初期化されます。

変換するチャネルは、AD コンバータの動作モードにより以下のように選択されます。

##### 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>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> の設定および ADSCN により、スキャンモードを選択します。

##### 2. 最優先 AD 変換時

ADMOD2<HPADCH>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。

#### 14.4.5 AD 変換動作詳細

##### 14.4.5.1 AD 変換の起動

通常 AD 変換は ADMOD0<ADS> に "1" をセットすることにより起動されます。また、最優先 AD 変換は ADMOD2<HPADCE> に "1" をセットすることにより起動されます。

通常 AD 変換は ADMOD0<REPEAT,SCAN> 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> で選択されるハードウェア起動ソースにより起動することができます。<ADHS>、<HADHS> が "0" の場合は、ADTRG 端子より立ち下がリエッジの入力により起動され、このビットが "1" の場合、通常 AD 変換は 16 ビットタイマのチャネル 6 の TB6RG0 一致で起動され、最優先 AD 変換の場合はチャネル 5 の TB5RG0 一致で起動されます。

ハードウェア起動を許可するには、通常 AD 変換では ADMOD4<ADHTG>、最優先 AD 変換では ADMOD4<HADHTG> に "1" をセットします。

ハードウェア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 変換のハードウェア起動ソースに外部トリガを使用しているときは、通常 AD 変換ハードウェア起動としては外部トリガを設定できません。

注) 本製品にはハードウェア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。

#### 14.4.5.2 AD 変換動作

通常 AD 変換が開始されると、AD 変換中を示す AD 変換 BUSY フラグ(ADMOD0<ADBFN>)に"1"がセットされます。

また、最優先 AD 変換が開始されると、最優先 AD 変換中を示す最優先 AD 変換 BUSY フラグ(ADMOD2<ADBFHP>)に 1 がセットされます。このとき、通常 AD 変換用の BUSY フラグ ADMOD0<ADBFN>と通常 AD 変換用の変換終了フラグ ADMOD0<EOCFN>は最優先 AD 変換の開始前の値を保持します

注) 最優先 AD 変換中に通常 AD 変換を再起動させないでください

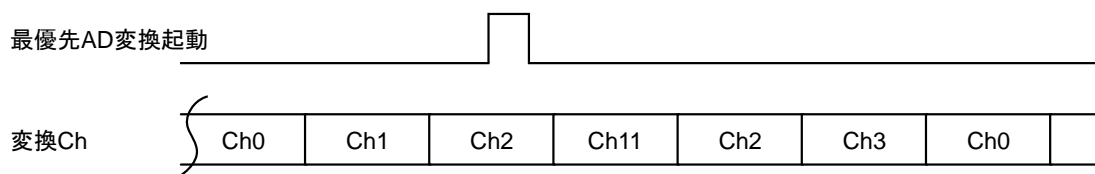
#### 14.4.5.3 通常 AD 変換中の最優先変換要求

通常 AD 変換中に最優先 AD 変換が起動されると、通常 AD 変換を中断し、最優先 AD 変換の終了後に通常 AD 変換を再開します。

通常 AD 変換中に ADMOD2<HPADCE>に"1"をセットすると、現在変換中の AD 変換は中断されて最優先 AD 変換を起動し、ADMOD2<HPADCH>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)が行われます。この結果を変換結果レジスタ ADREGSP へ格納すると、中断した通常 AD 変換を再開します。

通常 AD 変換中にハードウェアによる最優先 AD 変換の起動が許可されている場合は、ハードウェア起動ソース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<HPADCH>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。この結果を変換結果レジスタ ADREGSP へ格納すると、中断したチャンネルから通常 AD 変換を再開します。

例えばチャンネル AIN0~AIN3 までのチャンネルリピート変換が起動されており、AIN2 の変換中に <HPADCE>に"1"がセットされた場合は AIN2 の変換が中断され、<HPADCH>で指定されたチャンネルの変換(下図の場合 AIN11)を行い、結果を ADREGSP へ格納後に AIN2 からチャンネルリピート変換を再開します。



#### 14.4.5.4 リピート変換モードの停止

リピート変換モード(チャンネル固定リピートまたはチャンネルスキャンリピート変換モード)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<ADBFN>は"0"にクリアされます。

#### 14.4.5.5 通常 AD 変換の再起動

通常 AD 変換実行中に通常 AD 変換を再起動する場合はソフトウェアリセット (ADM0D3<ADRST>)を行ってから起動してください。ハードウェアによる通常 AD 変換の再起動は行わないでください。

#### 14.4.5.6 変換終了

##### (1) 通常 AD 変換の終了

通常 AD 変換が終了すると、変換終了割り込み要求(INTAD)が発生します。また、AD 変換結果がレジスタに格納され、AD 変換終了を示す ADM0D0<EOCFN> と変換中を示す ADM0D0<ADBFN>が変化します。変換モードにより、割り込み要求発生タイミング、変換結果レジスタ、<EOCFN><ADBFN>の変化タイミングは異なります。

チャンネル固定リピート変換モード以外のモードでは、変換結果はチャンネルに対応した変換結果レジスタ(ADREG08~ADRG7F)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、ADREG08 から ADREG7F へと順次格納されます。ただし、割り込み発生を<ITM>で 1 回ごとに指定した場合は ADREG08 のみに格納され、<ITM>で 4 回ごとに指定した場合は ADREG08H~ADREG3B へと順次格納されます。

モードごとの割り込み要求発生、フラグ変化、変換結果格納レジスタは以下の通りです。

- ・ チャンネル固定シングル変換モード

変換が終了した後、ADM0D0<EOCFN>が"1"にセット、ADM0D0<ADBFN>が"0"にクリアされ、変換終了割り込み要求が発生します。

変換結果はチャンネルに対応する変換結果レジスタに格納されます。

- ・ チャンネルスキップシングル変換モード

スキップ変換が終了した後、ADM0D0<EOCFN>が"1"、ADM0D0<ADBFN>が"0"となり、変換終了割り込み要求が発生します。

変換結果はチャンネルに対応する変換結果レジスタに格納されます。

- ・ チャンネル固定リピート変換モード

ADM0D0<ADBFN>は"0"とはならず"1"を保持します。割り込み要求発生タイミングは ADM0D0<ITM>の設定により選択できます。ADM0D0<EOCFN>がセット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

- a. 1 回変換

<ITM>を"00"に設定すると AD 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、変換結果は常に変換結果レジスタの ADREG08 に格納されます。格納時点で<EOCFN>は 1 になります。

- b. 4 回変換

<ITM>を"01"に設定すると AD 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、変換結果は変換結果レジスタの ADREG08~ADREG3B に順次格納されます。ADREG3B に格納後<EOCFN>は"1"にセットされ、再び ADREG08 から格納を始めます。

- c. 8 回変換

<ITM>を"10"に設定すると AD 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、変換結果は変換結果レジスタの ADREG08~ADREG7F に順次格納されます。ADREG7F 格納後<EOCFN>は"1"にセットされ、再び ADREG08 から格納を始めます。

- ・ チャンネルスキャンリピート変換モード

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<ADBFN>は"0"にならず"1"を保持します。

変換結果はチャンネルに対応する変換結果レジスタに格納されます。

## (2) 最優先 AD 変換の終了

最優先 AD 変換が終了すると、最優先変換終了割り込み要求(INTADHP)が発生し、最優先 AD 変換終了を示す ADMOD2<EOCFHP> が"1"にセットされます。

変換結果は変換結果レジスタ SP に格納されます。

## (3) データポーリング

割り込みを使用せずに、ポーリングで変換終了を確認することもできます。変換が終了すると ADMOD0<EOCFN>に"1"がセットされますのでこのビットをポーリングすることで変換終了を確認し変換結果を読み出してください。

変換結果レジスタは、ハーフワードまたはワードアクセスで読んでください。<OVRx>="0"、<ADR<sub>x</sub>RF>="1"であれば、正しい変換結果が得られたこととなります。

14.4.5.7 割り込み発生タイミングと変換結果格納レジスタ


表 14-1 に、変換モードと割り込み発生タイミング、フラグの関係を、表 14-2 にアナログ入力チャンネルと変換結果レジスタの対応をまとめます。

表 14-1 AD 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		スキャン/リピートモード設定 (ADMOD0)			割り込み発生 タイミング	<EOCFN>/ <EOCFHP> セットタイミング (注)	ADMOD0	ADMOD2
		<REPEAT>	<SCAN>	<ITM[1:0]>			<ADBFN> (割り込み 発生後)	<ADBFHP>
通常変換	チャンネル固定 シングル変換	0	0	-	変換終了後	変換終了後	0	-
	チャンネル固定 リピート変換	1	0	00	1回変換ごと	変換 1 回終了後	1	-
				01	4回変換ごと	変換 4 回終了後	1	-
				10	8回変換ごと	変換 8 回終了後	1	-
チャンネルスキャン シングル変換	0	1	-	スキャン変換 終了後	スキャン変換 終了後	0	-	
チャンネルスキャン リピート変換	1	1	-	1回のスキャン 変換終了ごと	1回のスキャン 変換終了後	1	-	
最優先変換		-	-	-	変換終了後	変換終了	-	0

注) ADMOD0<EOCFN>,ADMOD2<EOCFHP>はリードするとクリアされます。

表 14-2 アナログ入力チャンネルと AD 変換結果レジスタの対応

アナログ入力 チャンネル	通常変換				最優先変換
	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)	
AIN0	ADREG08	ADREG08 固定			ADREGSP
AIN1	ADREG19				
AIN2	ADREG2A				
AIN3	ADREG3B				
AIN4	ADREG4C				
AIN5	ADREG5D				
AIN6	ADREG6E				
AIN7	ADREG7F				
AIN8	ADREG08				
AIN9	ADREG19				
AIN10	ADREG2A				
AIN11	ADREG3B				

注) 変換結果レジスタは、ハーフワードまたはワードアクセスしてください。



**AD コンバータ使用時の注意**

電源電圧の変動や周囲のノイズの影響によって AD 変換結果がばらつくことがあります。AD 変換中に AD 入力に兼用となっている端子への入力及び端子出力を変化させる、または AD 変換中に出力ポートに設定している端子の出力電流が変動すると AD 変換精度が低下することがあります。プログラムで複数回の変換結果の平均値をとるなどして対策して下さい。



## 第 15 章 ウォッチドッグタイマ(WDT)

ウォッチドッグタイマは、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。

暴走を検出した場合、INTWDT 割り込みを発生またはマイコンをリセットします。

注) INTWDT 割り込みはマスク不能割り込み(NMI)要因のひとつです。

また、外部周辺装置に対しては、ウォッチドッグタイマアウト端子(WDTOUT)より"Low"を出力して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。

### 15.1 構成

図 15-1 にウォッチドッグタイマのブロック図を示します。

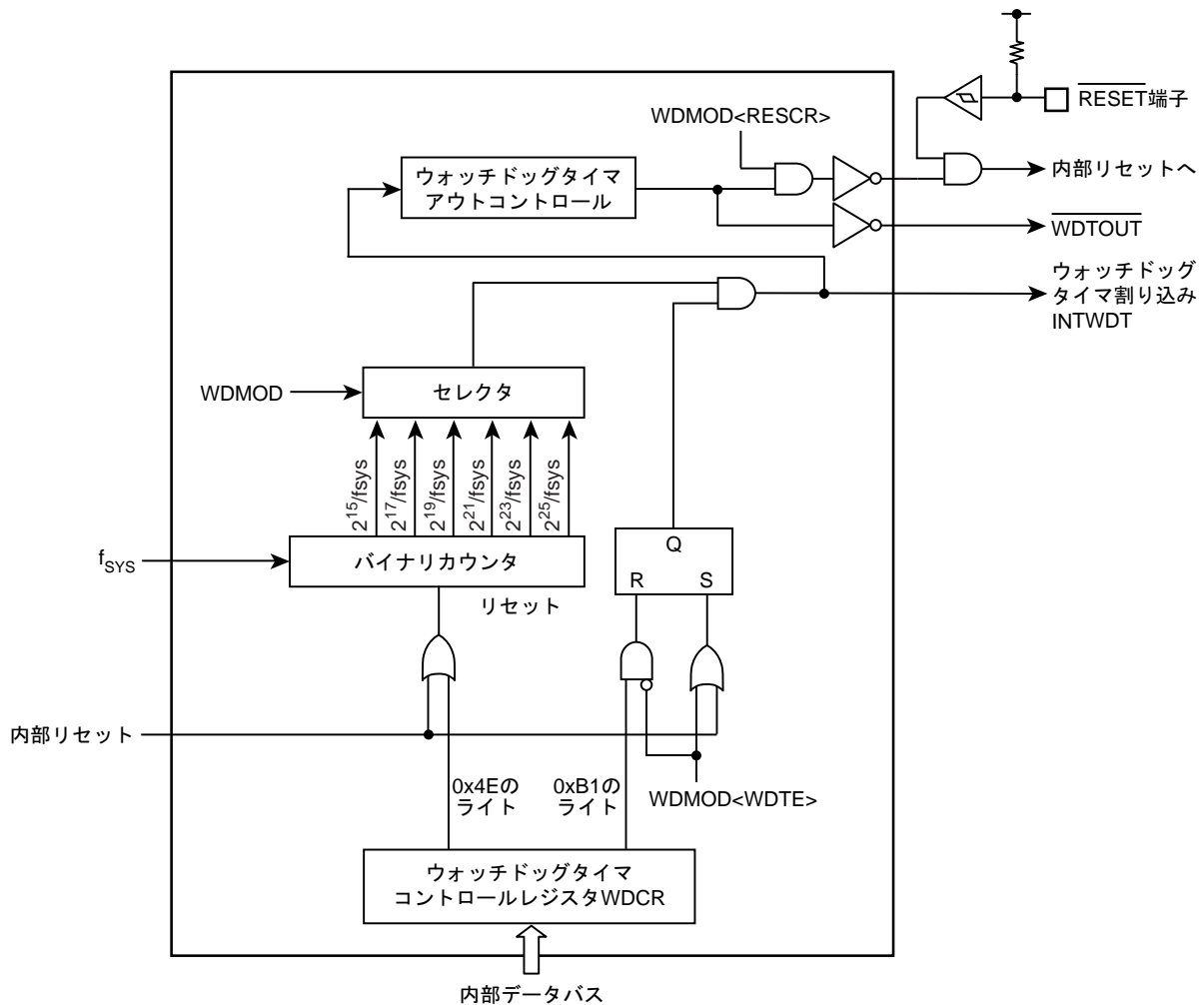


図 15-1 ウォッチドッグタイマのブロック図

## 15.2 レジスタ一覧

ウォッチドッグタイマの制御レジスタとアドレスは以下の通りです。

		Base Address = 0x4004_0000
レジスタ名		Address(Base+)
ウォッチドッグタイマモードレジスタ	WDMOD	0x0000
ウォッチドッグタイマコントロールレジスタ	WDCR	0x0004

### 15.2.1 WDMOD(ウォッチドッグタイマモードレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	WDTE	WDTP			-	I2WDT	RESCR	-
リセット後	1	0	0	0	0	0	1	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	WDTE	R/W	許可/禁止制御 0: 禁止 1: 許可
6-4	WDTP[2:0]	R/W	検出時間の選択(表 15-1 を参照) 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
3	-	R	リードすると"0"が読めます。
2	I2WDT	R/W	IDLE 時の動作 0: 停止 1: 動作
1	RESCR	R/W	暴走検出後の動作 0: INTWDT 割り込み要求を発生します。(注) 1: マイコンをリセットします。
0	-	R/W	"0"をライトしてください。

注) INTWDT 割り込みはマスク不能割り込み(NMI)要因のひとつです。

表 15-1 ウォッチドッグタイマの検出時間 (fc = 40 MHz)

クロックギア値 CGSYSCR<GEAR[2:0]>	WDMOD<WDTP[2:0]>					
	000	001	010	011	100	101
000 (fc)	0.82 ms	3.28 ms	13.11 ms	52.43 ms	209.72 ms	838.86 ms
100 (fc/2)	1.63 ms	6.55 ms	26.21 ms	104.86 ms	419.43 ms	1.68 s
101 (fc/4)	3.28 ms	13.11 ms	52.43 ms	209.72 ms	838.86 ms	3.36 s
110 (fc/8)	6.55 ms	26.21 ms	104.86 ms	419.43 ms	1.68 s	6.71 s

### 15.2.2 WDCR(ウォッチドッグタイマコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	WDCR							
リセット後	-	-	-	-	-	-	-	-

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	WDCR	W	ディセーブル/クリアコード 0xB1: ディセーブルコード 0x4E: クリアコード 上記以外:Reserved

## 15.3 動作説明

### 15.3.1 基本動作

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。検出時間は  $WDMOD<WDTP[2:0]>$  によって  $2^{15}$ ,  $2^{17}$ ,  $2^{19}$ ,  $2^{21}$ ,  $2^{23}$  および  $2^{25}$  から選択します。検出時間経過後にウォッチドッグタイマ割り込み(INTWDT)が発生し、ウォッチドッグタイマアウト端子( $\overline{WDTOUT}$ )より"Low"が出力されます。

ノイズなどの原因による CPU の暴走を検出するために、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 INTWDT 割り込みが発生する前にクリアするように設定しておきます。クリアが行われなかった場合、INTWDT によってマスク不能割り込みが発生しますので、CPU は暴走を検知し、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外部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( $\overline{WDTOUT}$ )はありません。

### 15.3.2 動作モードと動作状態

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。使用しない場合はディセーブルの処理を行ってください。

ウォッチドッグタイマは高速クロックが停止するモード(STOP、SLEEP、SLOW)では使用できません。モード遷移する前にディセーブルしてください。IDLE モード中は  $WDMOD<I2WDT>$  の設定に従います。

また、デバッグモード中は自動的にバイナリカウンタが停止します。

## 15.4 暴走検出時の動作

### 15.4.1 INTWDT 割り込み発生の場合

図 15-2 に INTWDT 割り込み発生(WDMOD<RESCR>="0")の場合の動作を示します。

バイナリカウンタのオーバーフローにより INTWDT 割り込みが発生します。INTWDT 割り込みはマスク不能割り込み(NMI)の要因であるため、CPU はマスク不能割り込みを検出し処理を行います。

マスク不能割り込み要因は複数あり、CGNMIFLG レジスタでマスク不能割り込み要因を識別できます。INTWDT 割り込みの場合、CGNMIFLG<NMIFLG0>がセットされます。

INTWDT 割り込み発生と同時にウォッチドッグタイマアウト( $\overline{\text{WDTOUT}}$ )より"Low"を出力します。 $\overline{\text{WDTOUT}}$  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 0x4E をライト)により"High"に戻ります。

注) 本製品にはウォッチドッグタイマアウトの外部出力端子はありません。

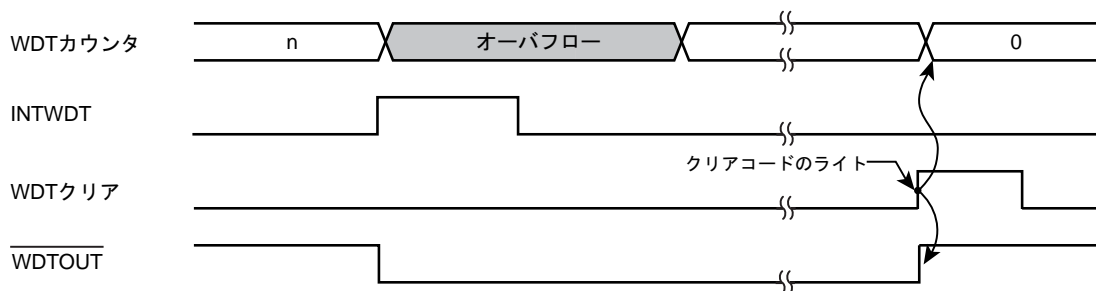


図 15-2 INTWDT 割り込み発生

### 15.4.2 内部リセット発生の場合

図 15-3 に内部リセット発生(WDMOD<RESCR>="1")の場合の動作を示します。

バイナリカウンタのオーバーフローによりマイコンをリセットします。この場合、32 ステートの期間、リセットを行います。クロックの設定も初期化され、入力クロック  $f_{SYS}$  と高速発振器のクロック  $f_{OSC}$  の関係は、 $f_{SYS} = f_{OSC}$  となります。

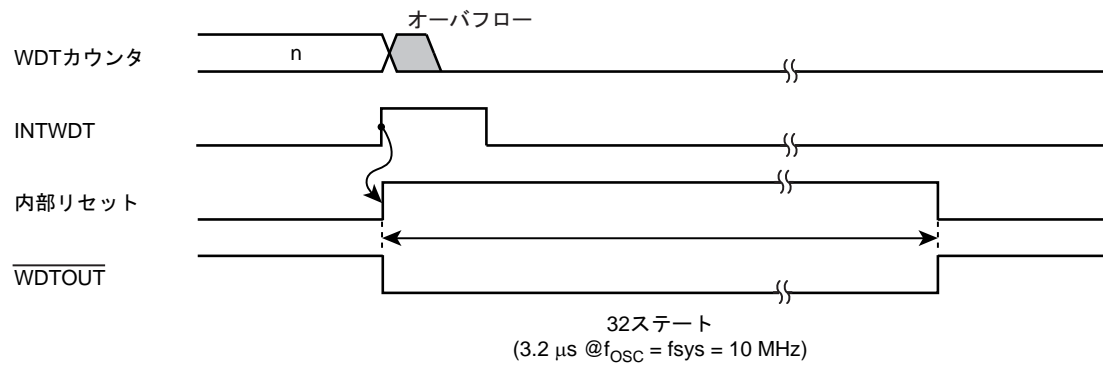


図 15-3 内部リセット発生



## 15.5 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。

### 15.5.1 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[2:0]>

ウォッチドッグタイマ検出時間を設定します。リセット時 WDMOD<WDTP[2:0]> = "000" に初期化されます。

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" に初期化されますので、ウォッチドッグタイマはイネーブルになっています。

暴走による誤書き込みを防止するため、ディセーブルにするには、このビットを"0"にした後で、WDCR にディセーブルコード(0xB1)を書き込む必要があります。

ディセーブル状態からイネーブル状態に戻す場合は、WDMOD<WDTE>を"1"に設定します。

3. ウォッチドッグタイマアウトのリセット接続<RESCR>

WDTOUT を内部リセットとして使用するか割り込みとして使用する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、バイナリカウンタのオーバーフローにより内部リセットが発生します。

### 15.5.2 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。

### 15.5.3 設定例

#### 15.5.3.1 ディセーブル制御

WDMOD<WDTE>に"0"を設定したあと、WDCR レジスタにディセーブルコード(0xB1)を書き込むとウォッチドッグタイマはディセーブルになり、バイナリカウンタはクリアされます。

		7	6	5	4	3	2	1	0	
WDMOD	←	0	-	-	-	-	-	-	-	<WDTE>に"0"を設定します。
WDCR	←	1	0	1	1	0	0	0	1	ディセーブルコード(0xB1)を書き込みます。

#### 15.5.3.2 イネーブル制御

WDMOD<WDTE>に"1"を設定します。

		7	6	5	4	3	2	1	0	
WDMOD	←	1	-	-	-	-	-	-	-	<WDTE>に"1"を設定します。

#### 15.5.3.3 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 (0x4E) 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。

		7	6	5	4	3	2	1	0	
WDCR	←	0	1	0	0	1	1	1	0	クリアコード(0x4E)を書き込みます。

#### 15.5.3.4 ウォッチドッグタイマ検出時間の設定

検出時間を  $2^21/f_{SYS}$  に設定する場合、WDMOD<WDTP[2:0]>に"011"を設定します。

		7	6	5	4	3	2	1	0	
WDMOD	←	1	0	1	1	-	-	-	-	

## 第 16 章 リアルタイムクロック(RTC)

### 16.1 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

### 16.2 ブロック図

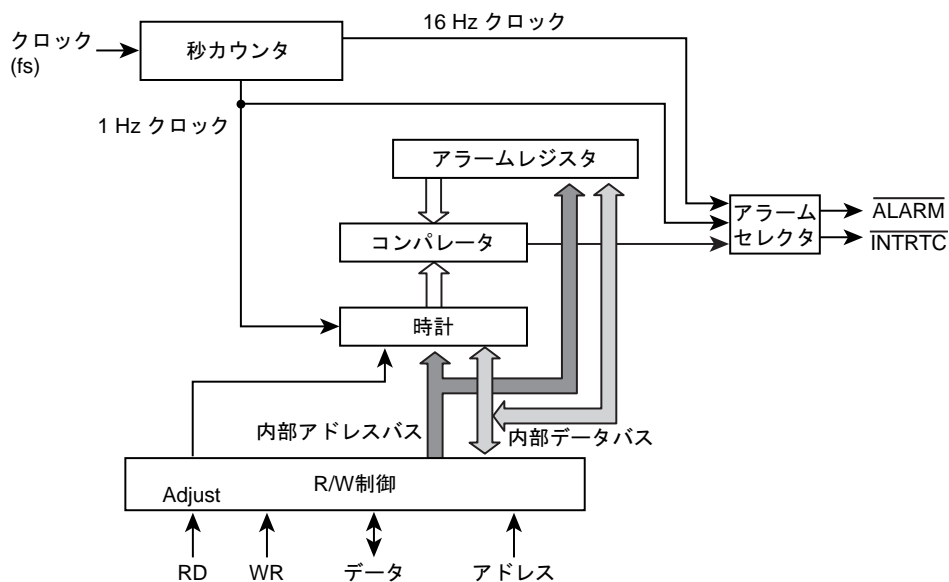


図 16-1 RTC ブロック図

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。

## 16.3 レジスタ説明

### 16.3.1 レジスタ一覧

RTC 関連のレジスタとアドレスを以下に示します。

RTC には PAGE0(時計機能)と PAGE1(アラーム機能)の 2 つの機能があり、一部のレジスタを共用しています。PAGE の選択は、RTCPAGER<PAGE>で行います。

Base Address = 0x4004\_0100

レジスタ名		Address(Base+)
秒桁レジスタ(PAGE0のみ)	RTCSECR	0x0000
分析レジスタ	RTCMINR	0x0001
時間桁レジスタ	RTCHOURR	0x0002
- (注 1)	-	0x0003
曜日桁レジスタ	RTCDAYR	0x0004
日桁レジスタ	RTCDATER	0x0005
月桁レジスタ(PAGE0)	RTCMONTHR	0x0006
24 時間時計、12 時間時計の選択レジスタ(PAGE1)		
年桁レジスタ(PAGE0)	RTCYEARR	0x0007
うるう年レジスタ(PAGE1)		
PAGE レジスタ	RTCPAGER	0x0008
- (注 1)	-	0x0009
- (注 1)	-	0x000A
- (注 1)	-	0x000B
リセットレジスタ	RTCRESTR	0x000C
Reserved	-	0x000D
- (注 1)	-	0x000E
- (注 1)	-	0x000F

注 1) リードすると"0"が読めます。また、書き込みは無視されます。

注 2) "Reserved"表記のアドレスにはアクセスしないでください。

### 16.3.2 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。

- ・ RTCPAGER<PAGE>, <ADJUST>, <INTENA>
- ・ RTCRESTR<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 を使用する際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「16.4.3 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 16-1 PAGE0 (時計機能)レジスタ

Symbol	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	設定内容
RTCSECR	-	40 秒	20 秒	10 秒	8 秒	4 秒	2 秒	1 秒	秒桁
RTCMINR	-	40 分	20 分	10 分	8 分	4 分	2 分	1 分	分桁
RTCHOURR	-	-	20 時/ PM/AM	10 時	8 時	4 時	2 時	1 時	時間桁
RTCDAYR	-	-	-	-	-	曜日設定			曜日桁
RTCDATER	-	-	20 日	10 日	8 日	4 日	2 日	1 日	日桁
RTCMONTHR	-	-	-	10 月	8 月	4 月	2 月	1 月	月桁
RTCYEARR	80 年	40 年	20 年	10 年	8 年	4 年	2 年	1 年	年桁(西暦下 2 桁)
RTCPAGER	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
RTCRESTR	1 Hz 許可	16 Hz 許可	時計 リセット	アラーム リセット	-	(FD/FY) - (FW) "1"をライトしてください。			リセット レジスタ

注) PAGE0 の RTCSECR, RTCMINR, RTCHOURR, RTCDAYR, RTCMONTHR, RTCYEARR はリードすると現在の状態がリードされます。

表 16-2 PAGE1 (アラーム機能)レジスタ

Symbol	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	設定内容
RTCSECR	-	-	-	-	-	-	-	-	-
RTCMINR	-	40 分	20 分	10 分	8 分	4 分	2 分	1 分	アラーム分桁
RTCHOURR	-	-	20 時/ PM/AM	10 時	8 時	4 時	2 時	1 時	アラーム時間桁
RTCDAYR	-	-	-	-	-	曜日設定			アラーム曜日桁
RTCDATER	-	-	20 日	10 日	8 日	4 日	2 日	1 日	アラーム日桁
RTCMONTHR	-	-	-	-	-	-	-	24/12	24 時間クロックモード
RTCYEARR	-	-	-	-	-	-	うるう年設定		うるう年モード
RTCPAGER	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
RTCRESTR	1 Hz 許可	16 Hz 許可	時計 リセット	アラーム リセット	-	(FD/FY) - (FW) "1"をライトしてください。			リセット レジスタ

注 1) PAGE1 の RTCMINR, RTCHOURR, RTCDAYR, RTCMONTHR, RTCYEARR はリードすると現在の状態がリードされます。

注 2) PAGE0 の RTCSECR, RTCMINR, RTCHOURR, RTCDAYR, RTCDATER, RTCMONTHR, RTCYEARR および PAGE1 の RTCYEARR(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。

注) 表 16-1 および表 16-2 の"FD"は TMPM330FDFG を、"FY"は TMPM330FYFG を、"FW"は TMPM330FWFG を示します。

## 16.3.3 レジスタ詳細

## 16.3.3.1 RTCSECR(秒桁レジスタ(PAGE0のみ))

	7	6	5	4	3	2	1	0
bit symbol	-	SE						
リセット後	0	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
7	-	R	リードすると"0"が読めます。
6-0	SE	R/W	秒桁設定 000_0000 : 00 秒      001_0000 : 10 秒      010_0000 : 20 秒 000_0001 : 01 秒      001_0001 : 11 秒      . 000_0010 : 02 秒      001_0010 : 12 秒      011_0000 : 30 秒 000_0011 : 03 秒      001_0011 : 13 秒      . 000_0100 : 04 秒      001_0100 : 14 秒      100_0000 : 40 秒 000_0101 : 05 秒      001_0101 : 15 秒      . 000_0110 : 06 秒      001_0110 : 16 秒      101_0000 : 50 秒 000_0111 : 07 秒      001_0111 : 17 秒      . 000_1000 : 08 秒      001_1000 : 18 秒      . 000_1001 : 09 秒      001_1001 : 19 秒      101_1001 : 59 秒

注) 上記以外の設定はしないでください。

## 16.3.3.2 RTCMINR(分桁レジスタ(PAGE0/1))

	7	6	5	4	3	2	1	0
bit symbol	-	MI						
リセット後	0	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
7	-	R	リードすると"0"が読めます。
6-0	MI	R/W	分桁設定 000_0000 : 00 分      001_0000 : 10 分      010_0000 : 20 分 000_0001 : 01 分      001_0001 : 11 分      . 000_0010 : 02 分      001_0010 : 12 分      011_0000 : 30 分 000_0011 : 03 分      001_0011 : 13 分      . 000_0100 : 04 分      001_0100 : 14 分      100_0000 : 40 分 000_0101 : 05 分      001_0101 : 15 分      . 000_0110 : 06 分      001_0110 : 16 分      101_0000 : 50 分 000_0111 : 07 分      001_0111 : 17 分      . 000_1000 : 08 分      001_1000 : 18 分      . 000_1001 : 09 分      001_1001 : 19 分      101_1001 : 59 分

注) 上記以外の設定はしないでください。

16.3.3.3 RTCHOURR(時間桁レジスタ(PAGE0/1))

(1) 24 時間クロックモード(RTCMONTHR<MO0> = "1")の場合

	7	6	5	4	3	2	1	0
bit symbol	-	-	HO					
リセット後	0	0	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
7-6	-	R	リードすると"0"が読めます。
5-0	HO	R/W	時間桁設定 00_0000 : 00 時      01_0000 : 10 時      10_0000 : 20 時 00_0001 : 01 時      01_0001 : 11 時      10_0001 : 21 時 00_0010 : 02 時      01_0010 : 12 時      10_0010 : 22 時 00_0011 : 03 時      01_0011 : 13 時      10_0011 : 23 時 00_0100 : 04 時      01_0100 : 14 時 00_0101 : 05 時      01_0101 : 15 時 00_0110 : 06 時      01_0110 : 16 時 00_0111 : 07 時      01_0111 : 17 時 00_1000 : 08 時      01_1000 : 18 時 00_1001 : 09 時      01_1001 : 19 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(RTCMONTHR<MO0> = "0")の場合

	7	6	5	4	3	2	1	0
bit symbol	-	-	HO					
リセット後	0	0	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
7-6	-	R	リードすると"0"が読めます。
5-0	HO	R/W	時間桁設定 (AM)                      (PM) 00_0000 : 00 時      10_0000 : 00 時 00_0001 : 01 時      10_0001 : 01 時 00_0010 : 02 時      10_0010 : 02 時 00_0011 : 03 時      10_0011 : 03 時 00_0100 : 04 時      10_0100 : 04 時 00_0101 : 05 時      10_0101 : 05 時 00_0110 : 06 時      10_0110 : 06 時 00_0111 : 07 時      10_0111 : 07 時 00_1000 : 08 時      10_1000 : 08 時 00_1001 : 09 時      10_1001 : 09 時 01_0000 : 10 時      11_0000 : 10 時 01_0001 : 11 時      11_0001 : 11 時

注) 上記以外の設定はしないでください。

## 16.3.3.4 RTCDAYR(曜日桁レジスタ(PAGE0/1))

	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	WE		
リセット後	0	0	0	0	0	不定	不定	不定

Bit	Bit Symbol	Type	機能
7-3	-	R	リードすると"0"が読めます。
2-0	WE	R/W	曜日桁設定 000 : 日曜日 001 : 月曜日 010 : 火曜日 011 : 水曜日 100 : 木曜日 101 : 金曜日 110 : 土曜日

注) 上記以外の設定はしないでください。

## 16.3.3.5 RTCDATER(日桁レジスタ(PAGE0/1))

	7	6	5	4	3	2	1	0
bit symbol	-	-	DA					
リセット後	0	0	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
7-6	-	R	リードすると"0"が読めます。
5-0	DA	R/W	日桁設定 00_0000 : 10 日      01_0000 : 10 日      10_0000 : 20 日      11_0000 : 30 日 00_0001 : 01 日      01_0001 : 11 日      10_0001 : 21 日      11_0001 : 31 日 00_0010 : 02 日      01_0010 : 12 日      10_0010 : 22 日 00_0011 : 03 日      01_0011 : 13 日      10_0011 : 23 日 00_0100 : 04 日      01_0100 : 14 日      10_0100 : 24 日 00_0101 : 05 日      01_0101 : 15 日      10_0101 : 25 日 00_0110 : 06 日      01_0110 : 16 日      10_0110 : 26 日 00_0111 : 07 日      01_0111 : 17 日      10_0111 : 27 日 00_1000 : 08 日      01_1000 : 18 日      10_1000 : 28 日 00_1001 : 09 日      01_1001 : 19 日      10_1001 : 29 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。



16.3.3.6 RTCMONTHR(月桁レジスタ(PAGE0のみ))

	7	6	5	4	3	2	1	0
bit symbol	-	-	-	MO				
リセット後	0	0	0	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能
7-5	-	R	リードすると"0"が読めます。
4-0	MO	R/W	月桁設定 0_0001 : 1月      0_0111 : 7月 0_0010 : 2月      0_1000 : 8月 0_0011 : 3月      0_1001 : 9月 0_0100 : 4月      1_0000 : 10月 0_0101 : 5月      1_0001 : 11月 0_0110 : 6月      1_0010 : 12月

注) 上記以外の設定はしないでください。

16.3.3.7 RTCMONTHR(24 時間時計, 12 時間時計の選択レジスタ(PAGE1のみ))

	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	MO0
リセット後	0	0	0	0	0	0	0	不定

Bit	Bit Symbol	Type	機能
7-1	-	R	リードすると"0"が読めます。
0	MO0	R/W	0 : 12 時間 1 : 24 時間

注) RTC 動作時(RTCPAGER<ENATMR> = "1")には、RTCMONTHR<MO0>を操作しないでください。

## 16.3.3.8 RTCYEARR(年桁レジスタ(PAGE0 のみ))

	7	6	5	4	3	2	1	0
bit symbol	YE							
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

Bit	Bit Symbol	Type	機能		
7-0	YE	R/W	年桁設定		
			0000_0000 : 00 年	0001_0000 : 10 年	0110_0000 : 60 年
			0000_0001 : 01 年	.	.
			0000_0010 : 02 年	0010_0000 : 20 年	0111_0000 : 70 年
			0000_0011 : 03 年	.	.
			0000_0100 : 04 年	0011_0000 : 30 年	1000_0000 : 80 年
			0000_0101 : 05 年	.	.
			0000_0110 : 06 年	0100_0000 : 40 年	1001_0000 : 90 年
			0000_0111 : 07 年	.	.
			0000_1000 : 08 年	01001_0000 : 50 年	.
			0000_1001 : 09 年	.	1001_1001 : 99 年

注) 上記以外の設定はしないでください。

## 16.3.3.9 RTCYEARR(うるう年レジスタ(PAGE1 のみ))

	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	LEAP	
リセット後	0	0	0	0	0	0	不定	不定

Bit	Bit Symbol	Type	機能
7-2	-	R	リードすると"0"が読めます。
1-0	LEAP	R/W	00 : 現在の年(今年)がうるう年 01 : 現在がうるう年から 1 年目 10 : 現在がうるう年から 2 年目 11 : 現在がうるう年から 3 年目

## 16.3.3.10 RTCPAGER(PAGE レジスタ(PAGE0/1))

	7	6	5	4	3	2	1	0
bit symbol	INTENA	-	-	ADJUST	ENATMR	ENAALM	-	PAGE
リセット後	0	0	0	0	不定	不定	0	0

Bit	Bit Symbol	Type	機能
7	INTENA	R/W	INTRTC 0: 禁止 1: 許可
6-5	-	R	リードすると"0"が読めます。
4	ADJUST	R/W	[ライト] 0: Don't care 1: ADJUST 要求セット 秒を補正します。要求は秒カウンタのカウント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。 [リード] 0: ADJUST 要求なし 1: ADJUST 要求あり "1"の場合 ADJUST 実行中を示し、"0"で処理が終了したことを示します。
3	ENATMR	R/W	時計 0: 禁止 1: 許可
2	ENAALM	R/W	ALARM 0: 禁止 1: 許可
1	-	R	リードすると"0"が読めます。
0	PAGE	R/W	PAGE 設定 0:Page0 が選択されます。 1:Page1 が選択されます。

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<ENAALM>のおおの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>,<ENAALM>の設定を変更する際には、<INTENA>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	
RTCPAGER ←	0	0	0	0	1	1	0	0	時計、アラーム許可
RTCPAGER ←	1	0	0	0	1	1	0	0	割り込み許可

## 16.3.3.11 RTCRESTR(リセットレジスタ(PAGE0/1))

	7	6	5	4	3	2	1	0
bit symbol	DIS1HZ	DIS16HZ	RSTTMR	RSTALM	-	-	-	-
リセット後	1	1	0	0	0	0(FD/FY) 1(FW)	0(FD/FY) 1(FW)	0(FD/FY) 1(FW)

Bit	Bit Symbol	Type	機能
7	DIS1HZ	R/W	1 Hz 割り込み 0: 許可 1: 禁止
6	DIS16HZ	R/W	16 Hz 割り込み 0: 許可 1: 禁止
5	RSTTMR	R/W	[ライト] 0: Don't care 1: 秒カウンタリセット 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。 [リード] 0: リセット要求なし 1: リセット要求あり "1"の場合リセット実行中を示し、"0"で処理が終了したことを示します。
4	RSTALM	R/W	0: Don't care 1: アラームリセット 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。
3	-	R	リードすると"0"が読めます。
2-0	-	R	FD/FY(注 2) : リードすると"0"が読めます
		R/W	FW(注 2) : "1"をライトしてください

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) "FD"は TMPM330FDFG を、"FY"は TMPM330FYFG を、"FW"は TMPM330FWFG を示します。

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>	RTCPAGER <ENAALM>	割り込みソース信号
1	1	1	アラーム
0	1	0	1 Hz
1	0	0	16 Hz
その他			割り込みは発生しません。

## 16.4 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。

### 16.4.1 時計データをリードする場合

1. 1 Hz 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 1 Hz の割り込みが発生しますので、1 Hz 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。

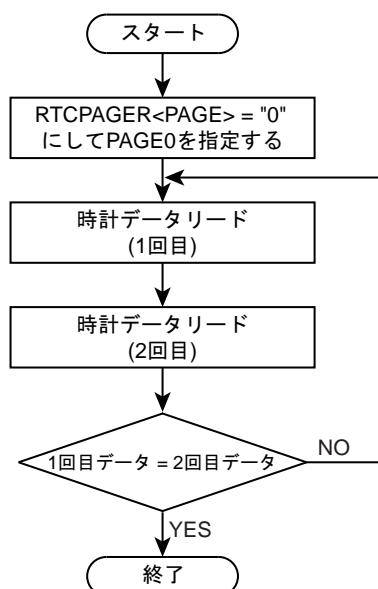


図 16-2 時計データのリードフロー

### 16.4.2 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次の方法があります。

1. 1 Hz 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 1 Hz の割り込みが発生しますので、1 Hz 割り込みを待って、次の 1 s 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 1 Hz 割り込みを許可した場合、1 秒後に 1 Hz 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。

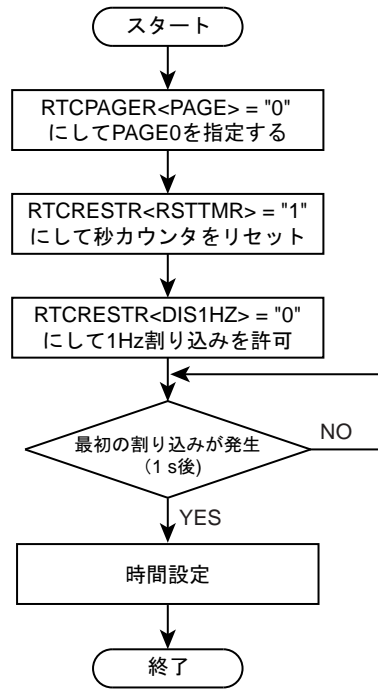


図 16-3 データライトのフロー

## 3. 時計を禁止する

RTCPAGER<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。

1 Hz 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 1 Hz 割り  
込みが発生する前(1 s 以内)に再度時計データを設定し時計を許可してください。

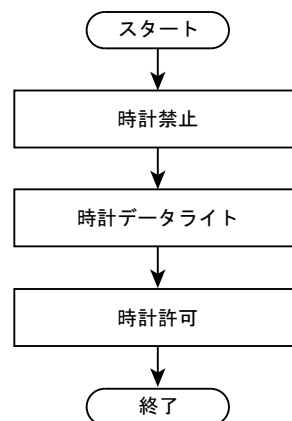


図 16-4 時計を禁止するフローチャート

### 16.4.3 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. 時計データの変更または<ADJUST>, <RSTTMR>操作後に 1 秒割り込みの発生を待つ。
2. 時計データの変更または<ADJUST>, <RSTTMR>操作後、時計レジスタ値<ADJUST>/<RSTTMR>の値を Read し、反映を確認する。

## 16.5 アラーム機能の説明

RTCPAGER<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、"Low"パルスを出力
2. 1 Hz 周期の"Low"パルスを出力
3. 16 Hz 周期の"Low"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、立ち下が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、"立ち下がりエッジ"に設定してください。

### 16.5.1 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"Low"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 RTCRESTR<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、RTCPAGER<ENAALM>でアラームを許可します。

最後に RTCPAGER<INTENA>で割り込みを許可します。

例えば、月曜日 5 日正午(12:00)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	
RTCPAGER	← 0	0	0	0	1	0	0	1	アラーム禁止, PAGE1 設定
RTCRESTR	← 1	1	0	1	0	0	0	0	アラーム初期化
RTCDAYR	← 0	0	0	0	0	0	0	1	月曜日
RTCDATER	← 0	0	0	0	0	1	0	1	5 日
RTCHOURR	← 0	0	0	1	0	0	1	0	12 時設定
RTCMINR	← 0	0	0	0	0	0	0	0	00 分設定
RTCPAGER	← 0	0	0	0	1	1	0	0	アラーム許可
RTCPAGER	← 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 fs 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 分といった場合)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。



### 16.5.2 1 Hz のパルスを出力する場合

RTCPAGER<ENAALM> = "0", RTCRESTR<DIS1HZ> = "0", <DIS16HZ> = "1" を設定後、RTCPAGER<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 "Low"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。

### 16.5.3 16 Hz のパルスを出力する場合

RTCPAGER<ENAALM> = "0", RTCRESTR<DIS1HZ> = "1", <DIS16HZ> = "0" を設定後、RTCPAGER<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 "Low"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。



## 第 17 章 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。

### 17.1 フラッシュメモリ

#### 17.1.1 特長

##### 1. メモリ容量

TMPM330FDFG/FYFG/FWFG はフラッシュメモリを搭載しています。それぞれの製品のメモリ容量と構成は下記の表のとおりです。

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にアクセスする場合、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 TMPM330FDFG/TMPM330FYFG では 128 ワード、TMPM330FWFG では 64 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 0.1 sec (Typ.)です。

1 チップあたりの書き込み時間、消去時間は以下のようになります。

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
		128 KB	64 KB	32 KB	16 KB			
TMPM330FDFG	512 KB	3	1	2	–	128	1.28 sec	0.4 sec
TMPM330FYFG	256 KB	1	1	2	–	128	0.64 sec	0.4 sec
TMPM330FWFG	128 KB	–	1	1	2	64	0.64 sec	0.2 sec

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には以下の 2 種類のモードがあります。

###### a. 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

###### b. 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。

JEDEC 準拠の機能	変更, 追加, 削除した機能
<ul style="list-style-type: none"> <li>・自動プログラム</li> <li>・自動チップ消去</li> <li>・自動ブロック消去</li> <li>・データポーリング/トグルビット</li> </ul>	<p>&lt;変更&gt;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>&lt;削除&gt;消去レジューム/サスペンド機能</p>

#### 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」の章を参照してください。

- 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。

17.1.2 フラッシュ部ブロック図

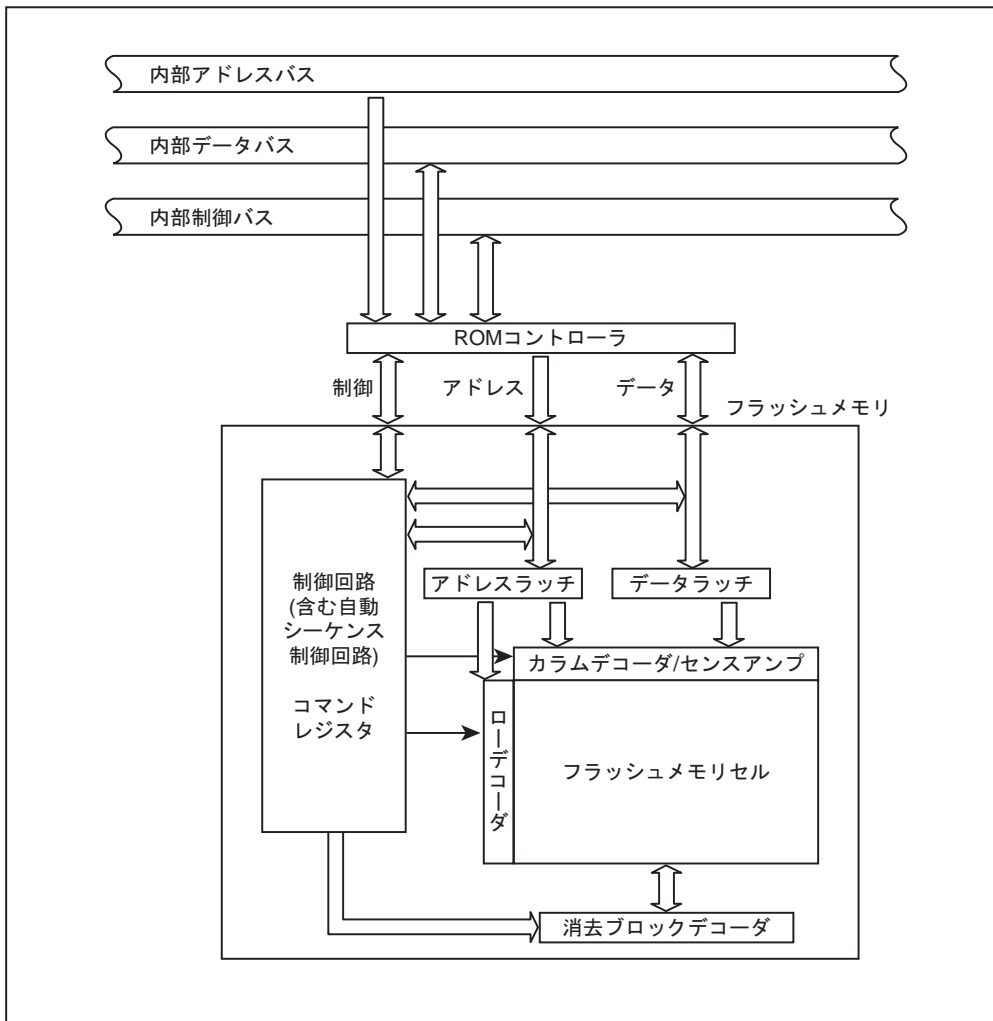


図 17-1 フラッシュ部ブロック図

## 17.2 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 17-1 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 17-1 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。

表 17-2 動作モード設定表

動作モード	端子	
	RESET	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

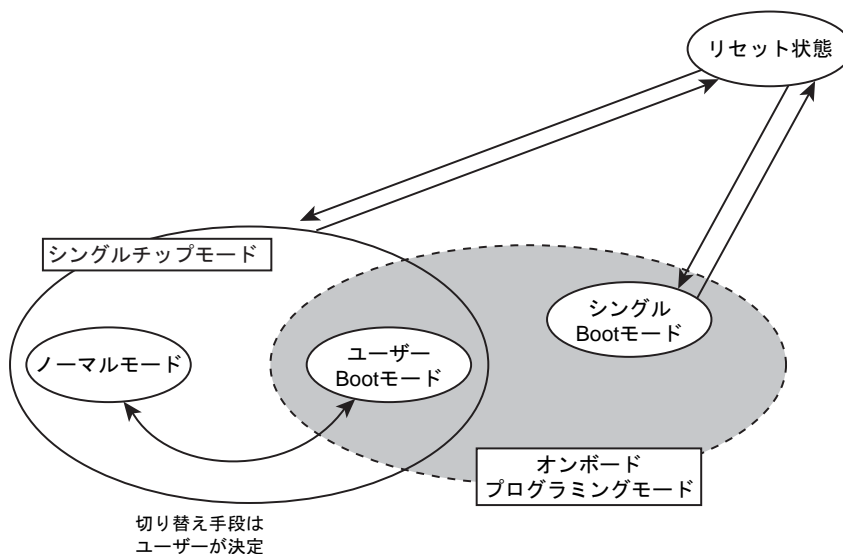


図 17-2 モード遷移図

### 17.2.1 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(40 MHz 動作で 0.3  $\mu$ s (リセット後は、クロックギア 1/1 モード)) RESET 入力を"0"にしてください。

注 1) 電源投入後は、電源電圧が安定した状態から 700  $\mu$ s 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  $\mu$ 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 2 ms 程度の時間がかかります。

### 17.2.2 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。従って、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ないよう、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。

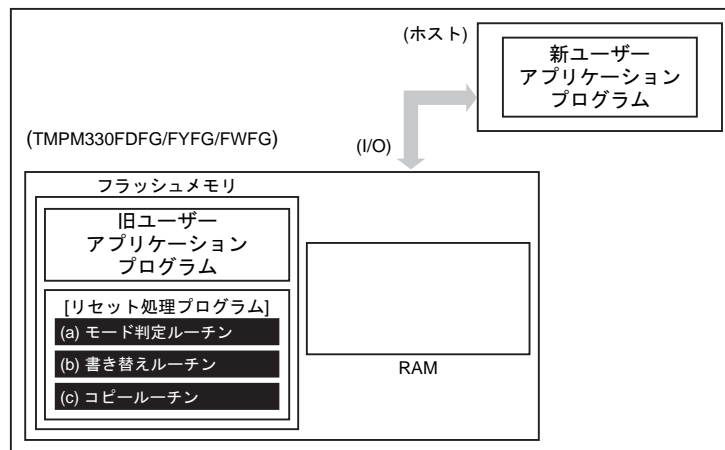
書き替えルーチンを内蔵フラッシュメモリに置く場合と、外部から転送する場合の 2 ケースを例に、以下 (1-A), (1-B)にその手順を説明します。フラッシュメモリへの書き込み/消去方法の詳細は、「17.3 オンボードプログラミングでのフラッシュメモリ書き込み/消去」を参照してください。

## 17.2.2.1 (1-A)書き替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

## (1) 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 3 つのプログラムを書き込んでおきます。

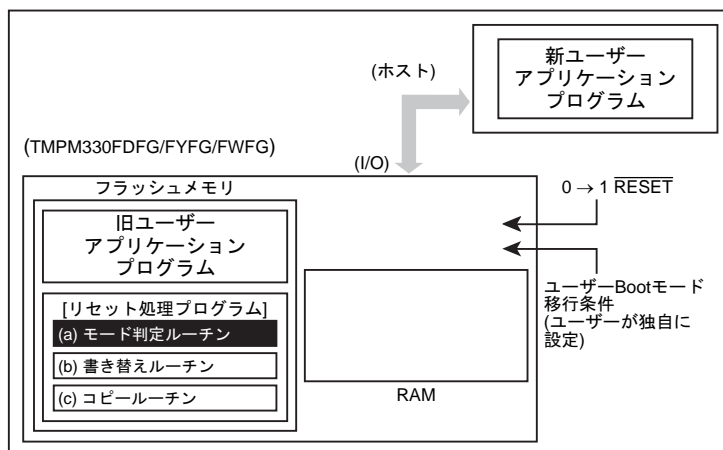
- |                    |   |
|--------------------|---|
| (a) モード判定ルーチン:     | 書き替え動作に移るためのプログラム                       |
| (b) フラッシュ書き替えルーチン: | 書き替えデータを外部から取り込み、フラッシュメモリを書き替えるためのプログラム |
| (c) コピールーチン:       | 上記(b)を内蔵 RAM または外部メモリにコピーするためのプログラム     |





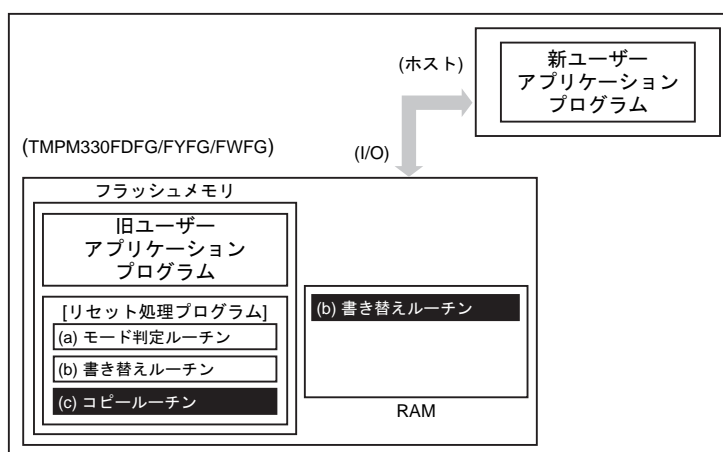
(2) Step-2

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。)



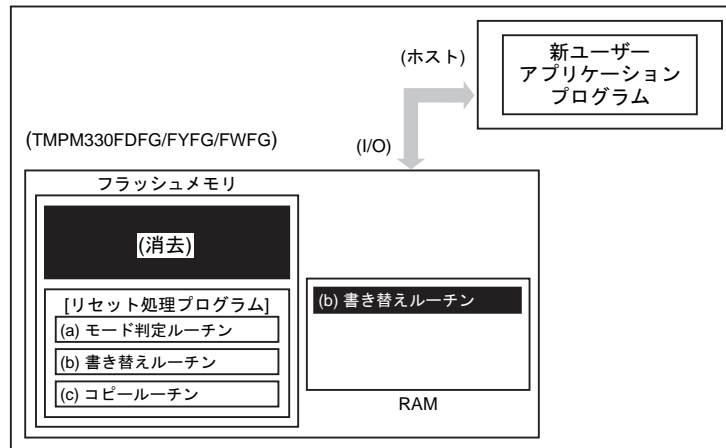
(3) Step-3

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。



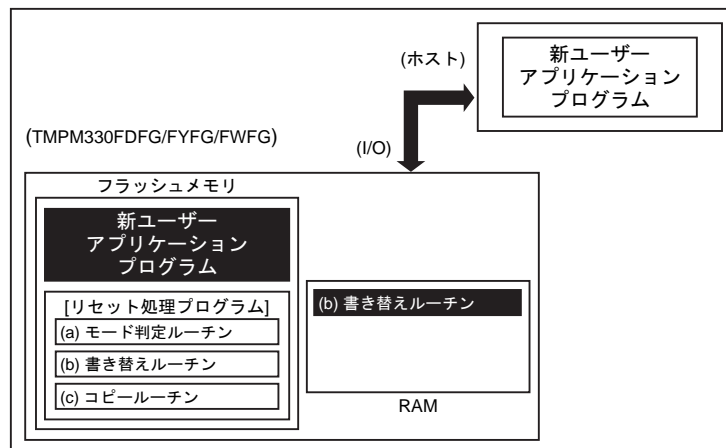
## (4) Step-4

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。



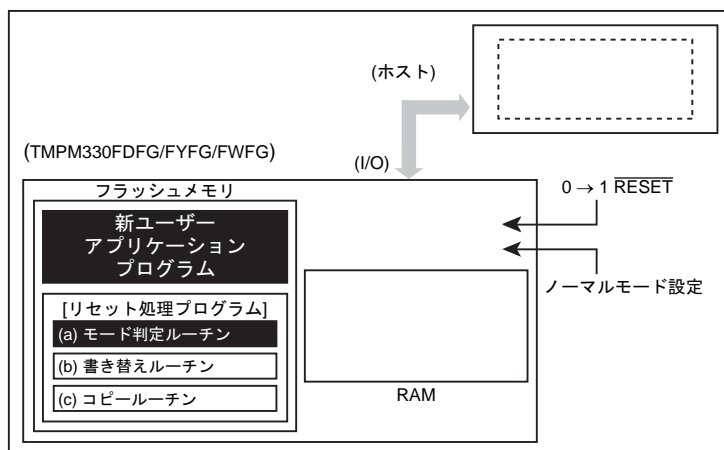
## (5) Step-5

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。



(6) Step-6

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。



## 17.2.2.2 (1-B)書き替えルーチンを外部から転送する手順例

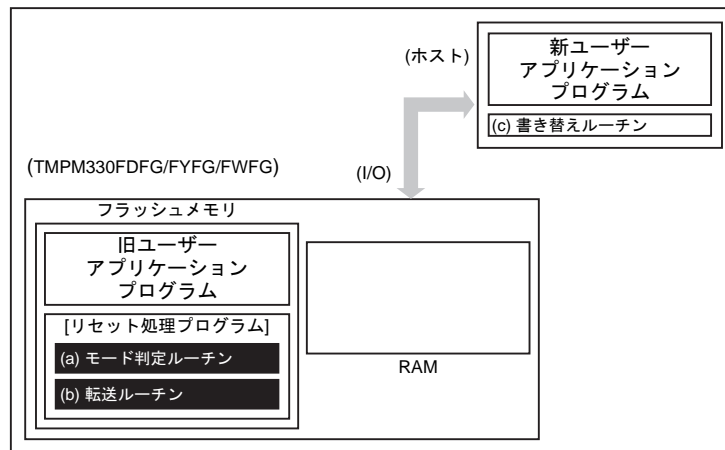
## (1) Step-1

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

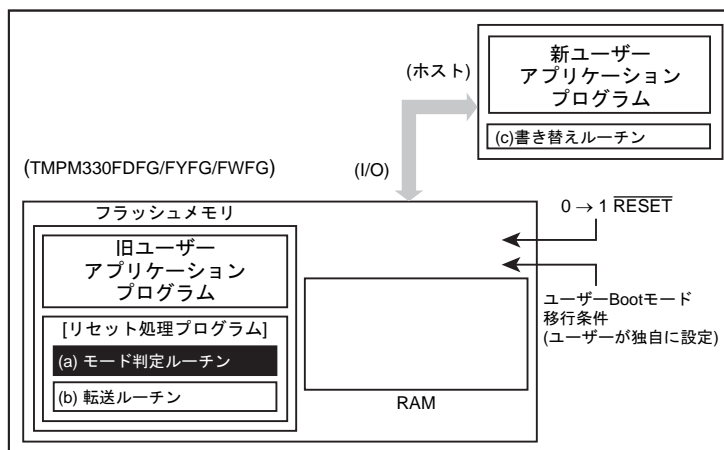
また、下記に示すプログラムはホスト上に用意します。

- (c) 書き替えルーチン: 書き替えを行うためのプログラム



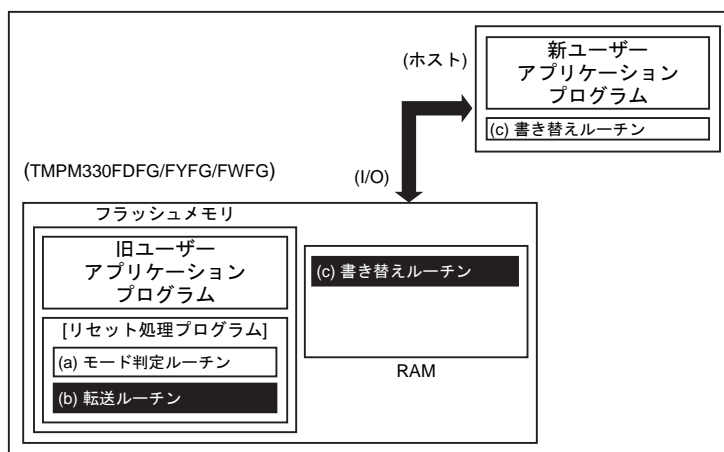
(2) Step-2

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。)



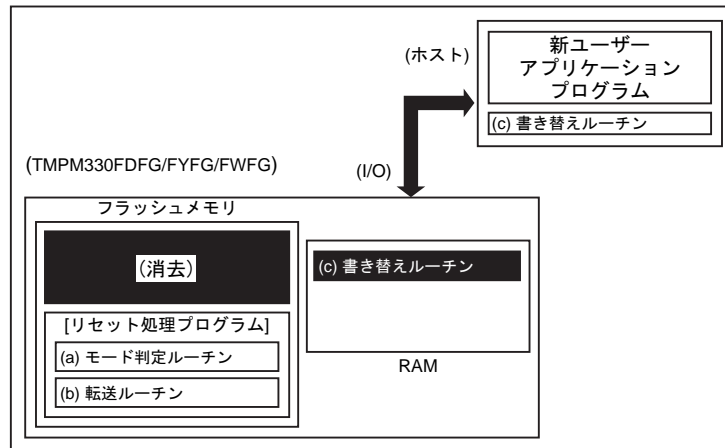
(3) Step-3

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。



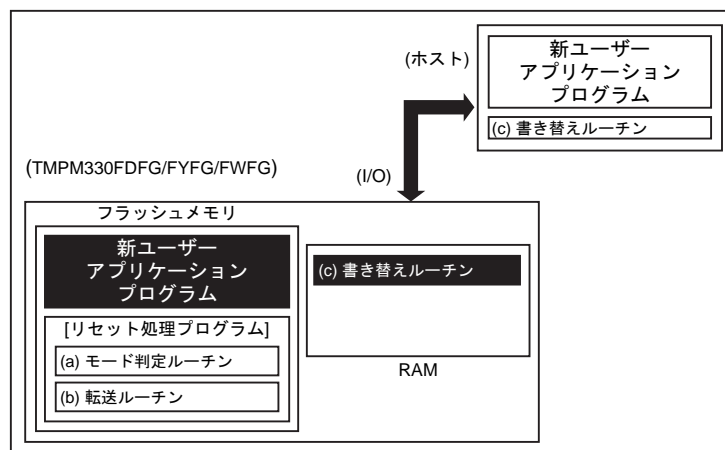
## (4) Step-4

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。



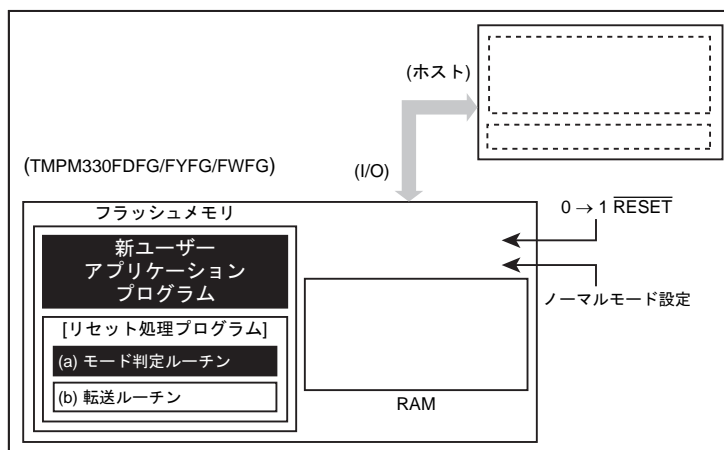
## (5) Step-5

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)より新用户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。



(6) 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。



### 17.2.3 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。

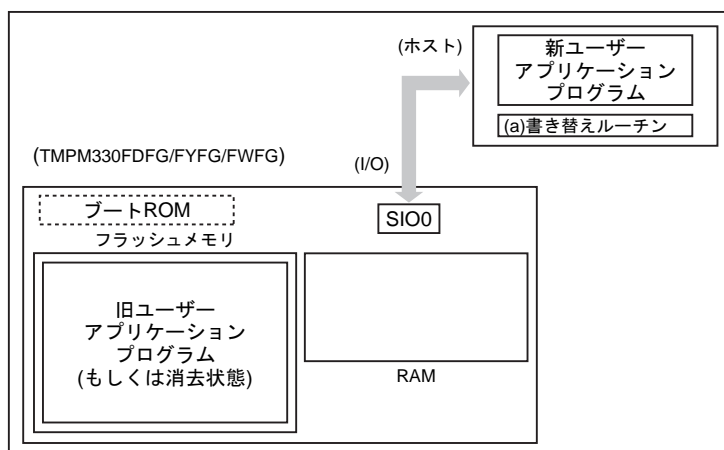
RAM へのプログラム転送は、ユーザーの ROM データ、セキュリティ確保のため、実行に先立ちユーザーパスワードの照合を行います。パスワードが一致しない場合は、RAM 転送そのものが実行されません。なお、シングルブートモードでもユーザーブートモードと同様、割り込み禁止状態で行います。シングルブートモード時、ブート ROM プログラムは NORMAL モードで動作します。

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よう、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。

#### 17.2.3.1 (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

##### (1) Step-1

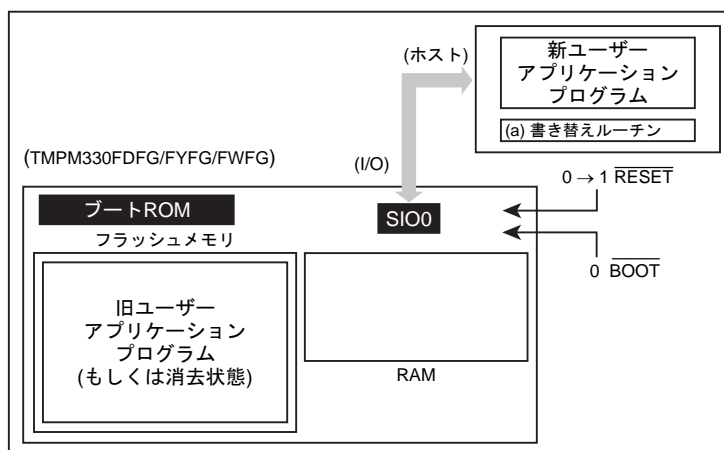
フラッシュメモリの状態は旧バージョンのユーザープログラムが書かれた状態でも、消去されている状態でも構いません。書き替えルーチン、書き替えデータなどの転送は SIO (SIO0) を経由して行いますので、ボード上で本デバイスの SIO (SIO0) と外部ホストとをつなげます。書き替えを行うための(a)書き替えルーチンはホスト上に用意します。





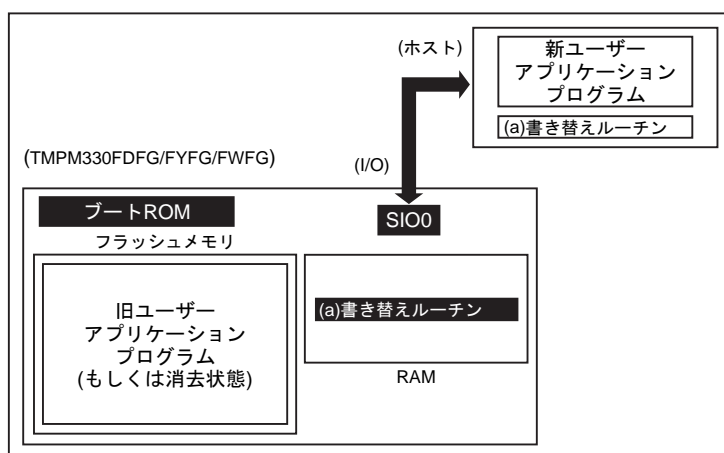
## (2) Step-2

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。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(0xFF)をパスワードとして照合を行います。)



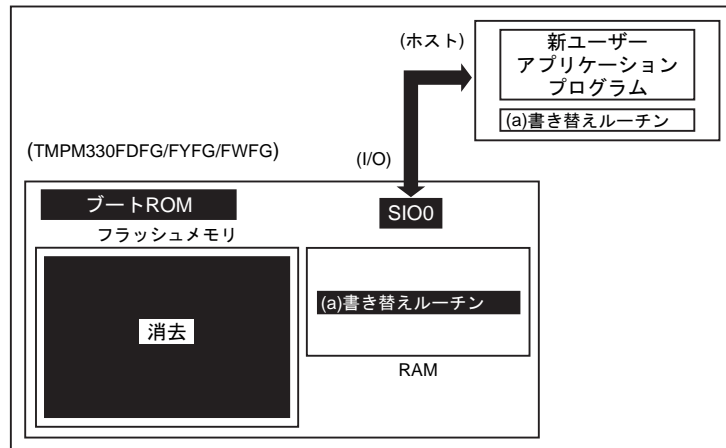
## (3) Step-3

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。



## (4) Step-4

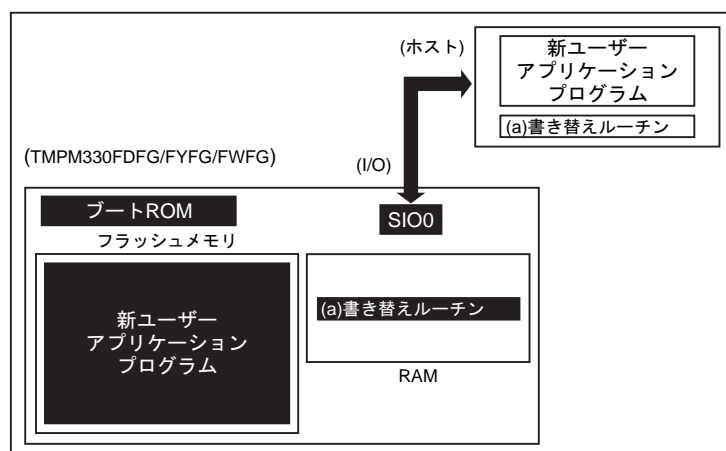
RAM 上の(a)書き替えルーチンへジャンプし、旧ユーザーアプリケーションプログラムエリアの消去を行います。(ブロック単位もしくは一括)



## (5) Step-5

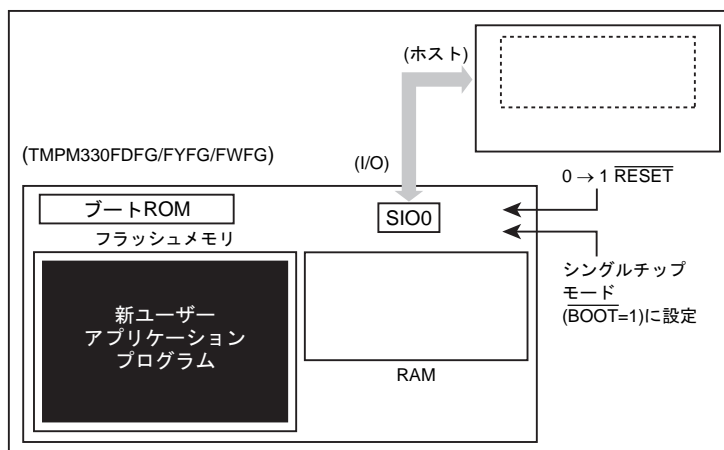
さらに、RAM 上の(a)書き替えルーチンを実行して、転送元(ホスト)より新ユーザーアプリケーションプログラムのデータをロードし、フラッシュメモリの消去したエリアに書き込みを行います。書き込みが完了したら、ユーザープログラムエリアのライト/消去プロテクトをオンにします。

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。



## (6) 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。



## 17.2.4 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。

## 17.2.5 メモリマップ

図 17-3 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。

製品ごとの内蔵フラッシュメモリと RAM のマッピングは以下のとおりです。

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
TMPM330FDFG	512 KB	32 KB	0x0000_0000 ~ 0x0007_FFFF 0x3F80_0000 ~ 0x3F87_FFFF	0x2000_0000 ~ 0x2000_7FFF
TMPM330FYFG	256 KB	16 KB	0x0000_0000 ~ 0x0003_FFFF (0x0007_FE00 ~ 0x0007_FFFF) (注) 0x3F80_0000 ~ 0x3F83_FFFF (0x3F87_FE00 ~ 0x3F87_FFFF) (注)	0x2000_0000 ~ 0x2000_3FFF
TMPM330FWFG	128 KB	8 KB	0x0000_0000 ~ 0x0001_FFFF 0x3F80_0000 ~ 0x3F81_FFFF	0x2000_0000 ~ 0x2000_1FFF

注) TMPM330FYFG 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。

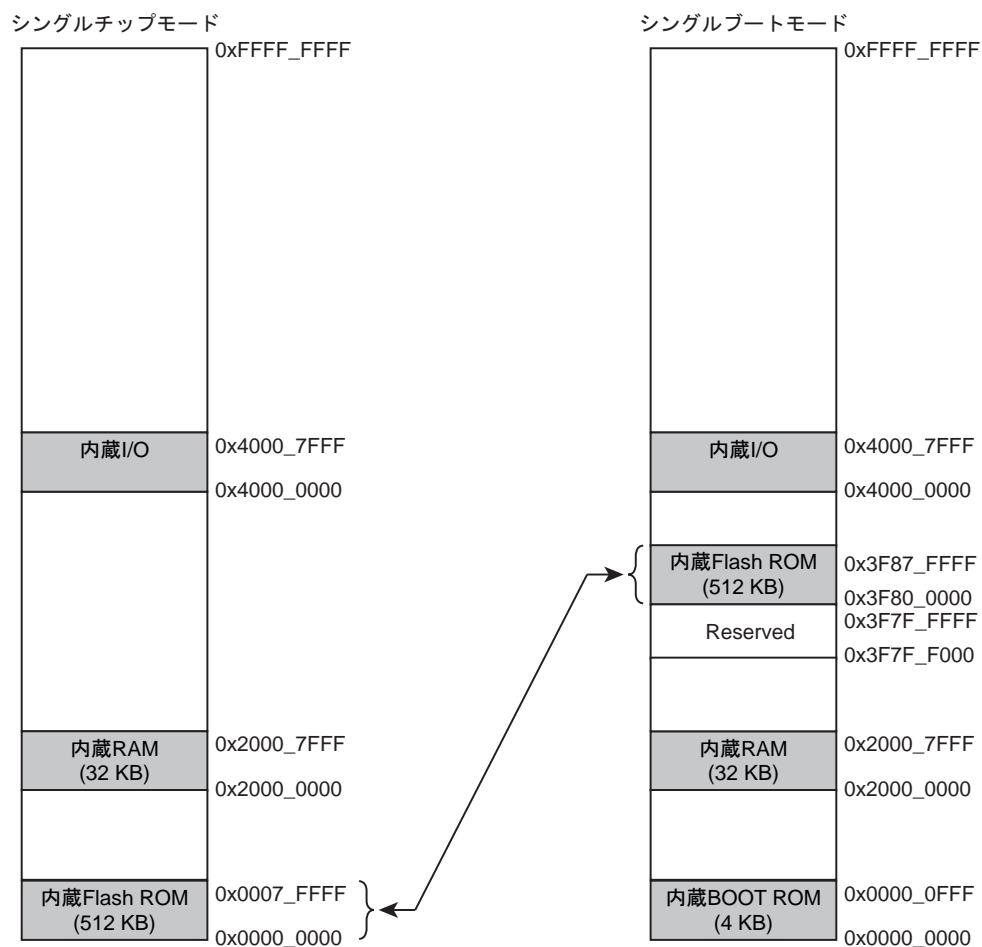


図 17-3 メモリマップの比較(TMPM330FDFG)

## 17.2.6 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 17-3 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	RVDD3	o	o
	AVDD	o	o
	DVDD3	o	o
	RVSS	o	o
	AVSS	o	o
	DVSS	o	o
モード設定端子	BOOT (PH0)	o	o
リセット端子	RESET	o	o
通信端子	TXD0 (PE0)	o	o
	RXD0 (PE1)	o	o
	SCLK0 (PE2)	x	o (入力モード)
	PE4	x	o (出力モード)

### 17.2.7 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 17-4, 表 17-6 ~ 表 17-9 に示します。「17.2.10 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 17-4 動作コマンドデータ

動作コマンドデータ	動作モード
0x10	RAM 転送
0x20	フラッシュメモリ SUM
0x30	製品情報読み出し
0x40	フラッシュメモリチップ消去およびプロテクトビット消去

### 17.2.8 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 Flash ROM に対して表 17-5 のように制約が付きま

す。

表 17-5 シングルブート時のメモリの制約

メモリ	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ください。
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ください。 TMPM330FDFG: 0x3F87_FF00 ~ 0x3F87_FF0F TMPM330FYFG: 0x3F87_FF00 ~ 0x3F87_FF0F TMPM330FWFG: 0x3F81_FF00 ~ 0x3F81_FF0F

### 17.2.9 ブートプログラムの転送フォーマット

各コマンドのブートプログラムの転送フォーマットを示します。「17.2.10 ブートプログラム動作説明」とあわせて参照してください。

17.2.9.1 RAM 転送

表 17-6 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: 0x86 I/O インタフェースの場合: 0x30	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: 0x86 (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: 0x30	
	3 バイト目	動作コマンドデータ (0x10)		-	
	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: 0x10 異常の場合: 0xX1 通信異常の場合: 0xX8	
	5 バイト目 ~ 16 バイト目	PASS WORD データ(12 バイト) 0x3F87_FF04 ~ 0x3F87_FF0F (FD/FY) 0x3F81_FF04 ~ 0x3F81_FF0F (FW)		-	
	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: 0x10 異常の場合: 0xX1 通信異常の場合: 0xX8	
	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: 0x10 異常の場合: 0xX1 通信異常の場合: 0xX8	
	27 バイト目 ~ m バイト目	RAM 格納データ		-	
	m + 1 バイト目	27 ~ m バイト値の 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: 0x10 異常の場合: 0xX1 通信異常の場合: 0xX8	
	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。

注 4) 表中の FD/ FY/ FW の表記は、TMPM330FDFG/ TMPM330FYFG/ TMPM330FWFG をあらわします。

## 17.2.9.2 フラッシュメモリ SUM

表 17-7 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: 0x86 I/O インタフェースの場合: 0x30	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: 0x86 (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: 0x30
	3 バイト目	動作コマンドデータ (0x20)		-
	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: 0x20 異常の場合: 0xX1 通信異常の場合: 0xX8
	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 CHECK SUM 値
	8 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。



17.2.9.3 デバイス情報読み出し

表 17-8 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]

	転送バイト数	コントローラ → 本デバイスの転送データ	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: 0x86 I/O インタフェースの場合: 0x30	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:0x86 (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: 0x30
	3バイト目	動作コマンドデータ(0x30)		-
	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: 0x30 異常の場合: 0xX1 通信異常の場合: 0xX8
	5バイト目	-		フラッシュメモリデータ 0x3F87_FF00 番地(FD/FY) 0x3F81_FF00 番地(FW)
	6バイト目	-		フラッシュメモリデータ 0x3F87_FF01 番地(FD/FY) 0x3F81_FF01 番地(FW)
	7バイト目	-		フラッシュメモリデータ 0x3F87_FF02 番地(FD/FY) 0x3F81_FF02 番地(FW)
	8バイト目	-		フラッシュメモリデータ 0x3F87_FF03 番地(FD/FY) 0x3F81_FF03 番地(FW)
	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_ '(FD/FY) 'TMPM330FW_ '(FW)
	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 0x04, 0xFF, 0x87, 0x3F (FD/FY) 0x04, 0xFF, 0x81, 0x3F (FW)
	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 0x00, 0x00, 0x00, 0x20 (FD/FY/FW)
	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 0x00, 0x00, 0x00, 0x00 (FD/FY/FW)
	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 0xFF, 0x7F, 0x00, 0x20 (FD/FY) (注 4) 0xFF, 0x1F, 0x00, 0x20 (FW)
	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 0x00, 0x00, 0x00, 0x00 (FD/FY/FW)
	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 0x00, 0x00, 0x00, 0x00 (FD/FY/FW)

表 17-8 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]

	転送バイト数	コントローラ → 本デバイスの転送データ	ポーレート	本デバイス → コントローラへの転送データ
	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 0x00, 0x00 (FD/FY/FW)
	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 0x00, 0x00, 0x80, 0x3F (FD/FY/FW)
	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 0xFF, 0xFF, 0x87, 0x3F (FD/FY) (注 5) 0xFF, 0xFF, 0x81, 0x3F (FW)
	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 0x06, 0x00 (FD/FY) 0x04, 0x00 (FW)
	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 0x00, 0x00, 0x00, 0x00 (FD/FY) 0x00, 0x00, 0x80, 0x3F (FW)
	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 0x00, 0x20, 0x00, 0x00 (FD/FY/FW)
	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) 0x00 (FD/FY) 0x02 (FW)
	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 0x00, 0x00, 0x80, 0x3F (FD/FY) 0x00, 0x80, 0x80, 0x3F (FW)
	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 0x00, 0x40, 0x00, 0x00 (FD/FY/FW)
	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) 0x02 (FD/FY) 0x01 (FW)
	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 0x00, 0x00, 0x81, 0x3F (FD/FY/FW)
	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 0x00, 0x80, 0x00, 0x00 (FD/FY/FW)

表 17-8 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]

転送バイト数	コントローラ → 本デバイスの転送データ	ボーレート	本デバイス → コントローラへの転送データ
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 0x01 (FD/FY/FW)
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 0x00, 0x00, 0x82, 0x3F (FD/FY) 0x00, 0x00, 0x00, 0x00 (FW)
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 0x00, 0x00, 0x01, 0x00 (FD/FY/FW)
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) 0x03 (FD/FY) (注 6) 0x00 (FW)
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) 表中の FD/ FY/ FW の表記は、TMPM330FDFG/ TMPM330FYFG/ TMPM330FWFG をあらわします。
- 注 4) TMPM330FYFG の RAM 終了アドレスは、実際には 0x2000\_3FFF です。
- 注 5) TMPM330FYFG のフラッシュメモリ終了アドレスは、実際には 0x3F83\_FFFF です。また、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)
- 注 6) TMPM330FYFG の 128KB ブロック個数は、実際には 1 個です。

## 17.2.9.4 フラッシュメモリチップ消去およびプロテクトビット消去

表 17-9 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およ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: 0x86 I/O インタフェースの場合: 0x30	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 正常(設定可能)の場合 ・ UART の場合: 0x86 ・ I/O インタフェースの場合: 0x30 (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(0x40)		-
	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: 0x40 異常の場合: 0xX1 通信異常の場合: 0xX8
	5 バイト目	消去イネーブルコマンドデータ(0x54)		-
	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: 0x54 異常の場合: 0xX1 通信異常の場合: 0xX8
	7 バイト目	-		消去コマンドに対する ACK 応答 正常の場合: 0x4F 異常の場合: 0x4C
	8 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。

## 17.2.10 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。

### 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、17.3 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。製品ごとの対象エリアは以下のとおりです。

製品	対象エリア
TMPM330DFDG TMPM330FYFG	0x3F87_FF00 ~ 0x3F87_FF03
TMPM330FWFG	0x3F81_FF00 ~ 0x3F81_FF03

### 4. フラッシュメモリチップ消去およ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/消去プロテクトおよび、セキュリティ状態にかかわらず、メモリセルのすべてのブロックを消去し、すべ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、FCSECBIT <SECBIT>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。

### 17.2.10.1 RAM 転送コマンド

データ転送フォーマットは表 17-6 を参照してください。

- 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  $0x86$  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  $0x30$  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。従って、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。従って、ACK 応答データの通信異常 ACK (bit 3) ( $0xX8$ ) 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  $0x86$  を I/O インタフェースと判定された場合  $0x30$  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、SC0BRCR の値を書き替え、 $0x86$  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 ( $0x86$ ) 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 ( $0x86$ ) 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  $0x30$  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 ms)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  $0x30$  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 ( $SC0MOD0<RXE>=1$ ) するタイミングは、送信バッファにデータ ( $0x30$ ) 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 ( $0x10$ ) 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)  $0xX8$  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 17-4 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、0x10 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) 0xX1 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
TMPM330DFDG TMPM330FYFG	0x3F87_FF04 ~ 0x3F87_FF0F
TMPM330FWFG	0x3F81_FF04 ~ 0x3F81_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) 0x18 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 (オーバーフローを無視) して得られた値の下位 8 ビットが、0x00 かどうかをチェックしています。0x00 以外の場合、CHECK SUM エラーの ACK 応答データ(bit 0) 0x11 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) 0x11 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、0xFF 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 0x10 を送信します。

- 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。
- 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) 0x18 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(オーバフローを無視)して得られた値の下位 8 ビットが、0x00 かどうかをチェックしています。0x00 以外の場合、CHECK SUM エラーの ACK 応答データ(bit 0) 0x11 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 0x10 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) 0x18 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、0x00 かどうかをチェックしています。0x00 以外の場合、CHECK SUM エラーの ACK 応答データ(bit 0) 0x11 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 0x10 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 0x10 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。

### 17.2.10.2 フラッシュメモリ SUM コマンド

データ転送フォーマットは表 17-7 を参照してください。

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(0x20)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) 0xX8 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 17-4 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、0x20 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) 0xX1 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。

### 17.2.10.3 製品情報読み出しコマンド

データ転送フォーマットは表 17-8 を参照してください。

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(0x30)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) 0xX8 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 17-4 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、0x30 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) 0xX1 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
TMPM330DFDG TMPM330FYFG	0x3F87_FF00 ~ 0x3F87_FF03
TMPM330FWFG	0x3F81_FF00 ~ 0x3F81_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
TMPM330FDFG TMPM330FYFG	T, M, P, M, 3, 3, 0, F, D, [, ], _
TMPM330FWFG	T, M, P, M, 3, 3, 0, F, W, [, ], _

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、製品ごとに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
TMPM330FDFG TMPM330FYFG	0x04, 0xFF, 0x 87, 0x3F
TMPM330FWFG	0x04, 0xFF, 0x 81, 0x3F

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、0x00, 0x00, 0x00, 0x20 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、0x00, 0x00, 0x00, 0x00 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、製品ごとに以下の値を送信します。

製品	RAM 終了アドレス
TMPM330FDFG TMPM330FYFG	0xFF, 0x7F, 0x00, 0x20
TMPM330FWFG	0xFF, 0x1F, 0x00, 0x20

注) TMPM330FYFG 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、0x00, 0x00, 0x00, 0x00 になります。  
41 バイト目～44 バイト目の送信データは、0x00, 0x00, 0x00, 0x00 になります。
11. 45 バイト目～46 バイト目の送信データは、0x00, 0x00 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、0x00, 0x00, 0x80, 0x3F 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、製品ごとに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
TMPM330FDFG TMPM330FYFG	0xFF, 0xFF, 0x87, 0x3F
TMPM330FWFG	0xFF, 0xFF, 0x81, 0x3F

注) TMPM330FYFG のフラッシュメモリ終了アドレスは、実際には 0x3F83\_FFFF です。また、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、製品ごとに以下の値を送信します。

製品	ブロック分割数
TMPM330DFDG	0x06, 0x00
TMPM330FYFG	0x06, 0x00
TMPM330FWFG	0x04, 0x00

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 17-8 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。

#### 17.2.10.4 フラッシュメモリチップ消去およびプロテクトビット消去コマンド

データ転送フォーマットは表 17-9 を参照してください。

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(0x40)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) 0xX8 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 17-4 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、0x40 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) 0xX1 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(0x54)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) 0xX8 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、0x54 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) 0xX1 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(0x4F)を返します。

消去 Error が起きた場合は、エラーコード(0x4C)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。

## 17.2.10.5 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 17-10 から表 17-13 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 17-10 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)
0x30	I/O インタフェースでの通信が可能と判定した。

注) 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 17-11 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
0x?8 (注)	動作コマンドデータに受信エラーが発生した。
0x?1 (注)	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) 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 17-12 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)	受信エラーが発生していた。
0xN1 (注)	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)	CHECK SUM 値は正常な値と判定した。

注) 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 17-13 フラッシュメモリチップ消去およ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
0x54	消去イネーブルコマンドと判定した。
0x4F	消去コマンド終了
0x4C	消去コマンドが不正に終了した。

## 17.2.10.6 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 0x86 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 0x30 にして送信してください。図 17-4 にそれぞれの場合の波形を示します。

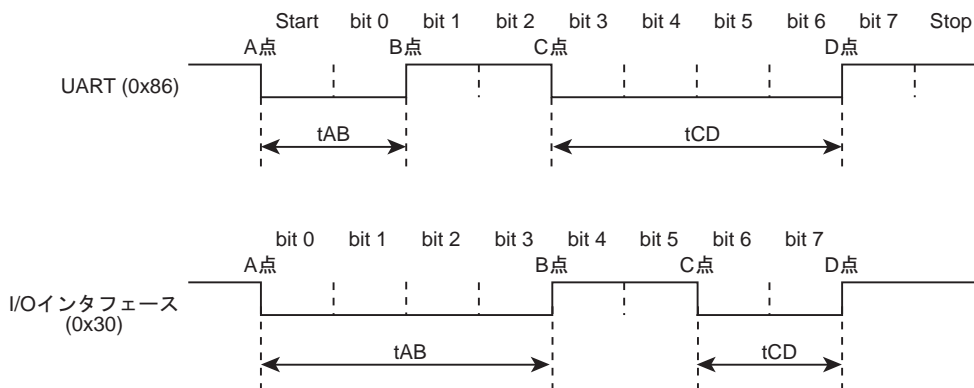


図 17-4 シリアル動作モード判定データ

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(0x86, 0x30)を受信禁止状態にして、図 17-5 に示すフローチャートで、図 17-4 の  $t_{AB}$ ,  $t_{AC}$  と、 $t_{AD}$  の時間を求めています。図 17-5 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 17-5 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 0x86 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 0x30 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)。

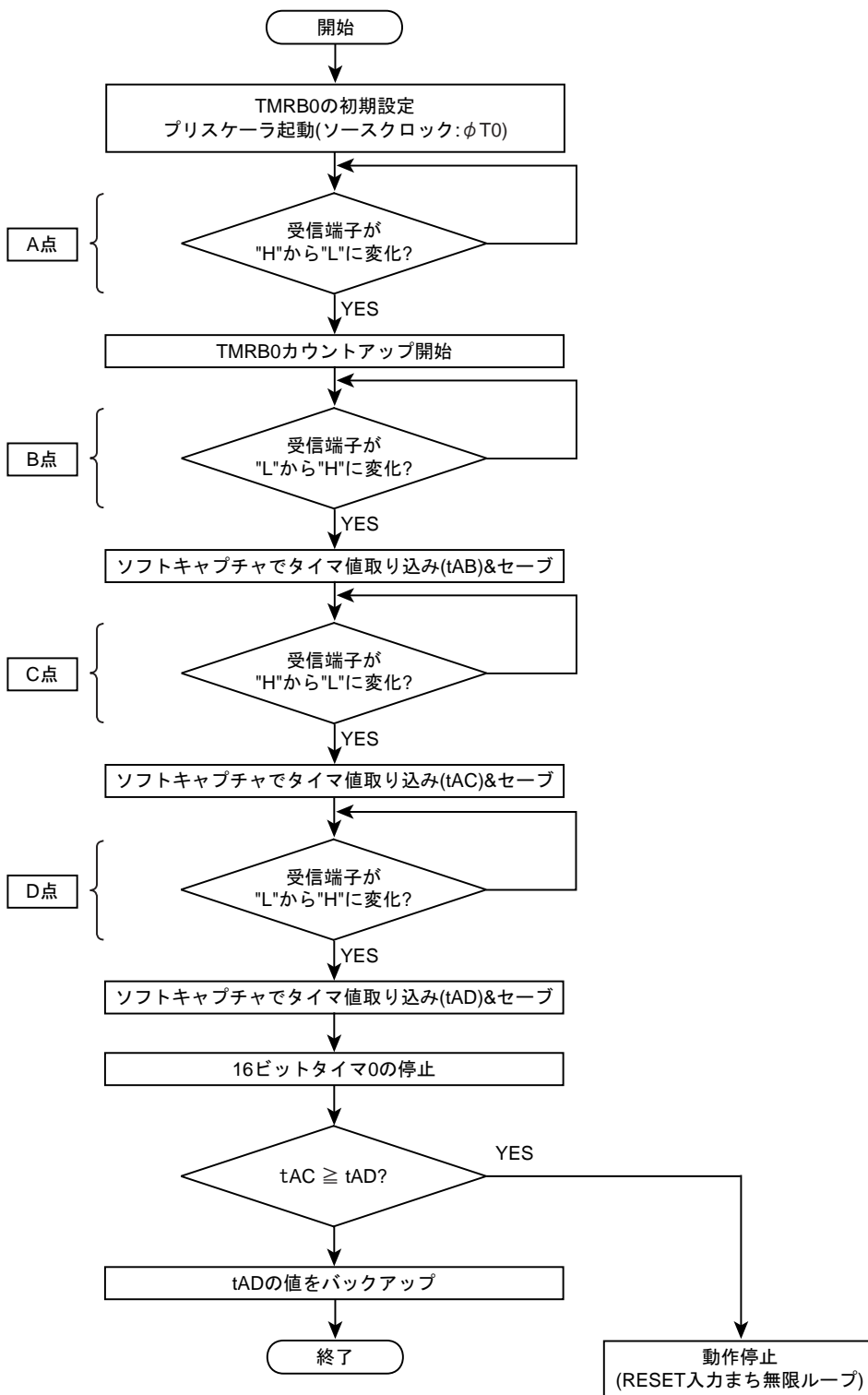


図 17-5 シリアル動作モード受信フローチャート

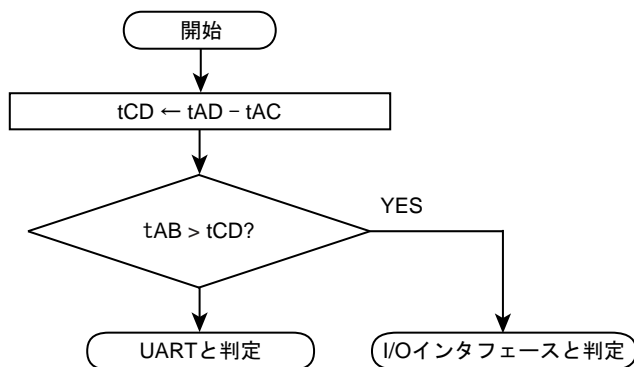


図 17-6 シリアル動作モード判定フローチャート

## 17.2.10.7 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(0x10)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(0x10)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
TMPM330FDFG TMPM330FYFG	0x3F87_FF04 ~ 0x3F87_FF0F
TMPM330FWFG	0x3F81_FF04 ~ 0x3F81_FF0F

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。

図 17-7 に示すようにパスワードエリアのデータが、0xFF 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 0x11 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。

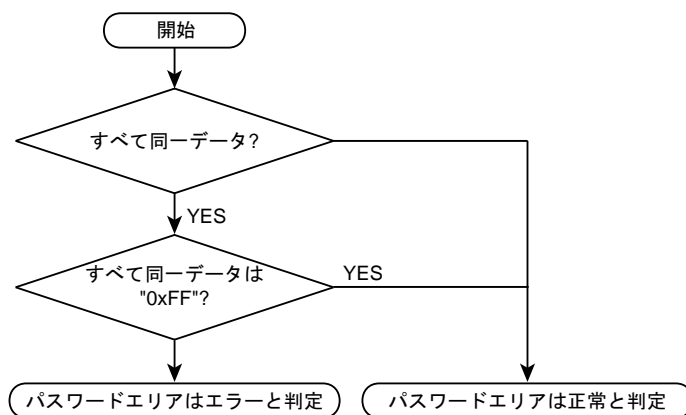


図 17-7 パスワードエリアチェックフローチャート



### 17.2.10.8 SUM の計算方法

フラッシュメモリ SUM コマンドでの 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のデータになります。

注) TMPM330FYFG 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)

0xA1
0xB2
0xC3
0xD4

左記4バイトが計算対象データの場合、SUMの値は、  
 $0xA1 + 0xB2 + 0xC3 + 0xD4 = 0x02EA$   
 となるので、  
 SUMの上位のデータは、0x02、  
 SUMの下位のデータは、0xEAになります。  
 従って、コントローラは0x02, 0xEAの順番で送信します。

### 17.2.10.9 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 0xE5、下位 8 ビットデータが 0xF6 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。

$$0xE5 + 0xF6 = 0x1DB$$

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。従って、コントローラには 0x25 を送信します。

$$0 - 0xDB = 0x25$$

### 17.2.11 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。

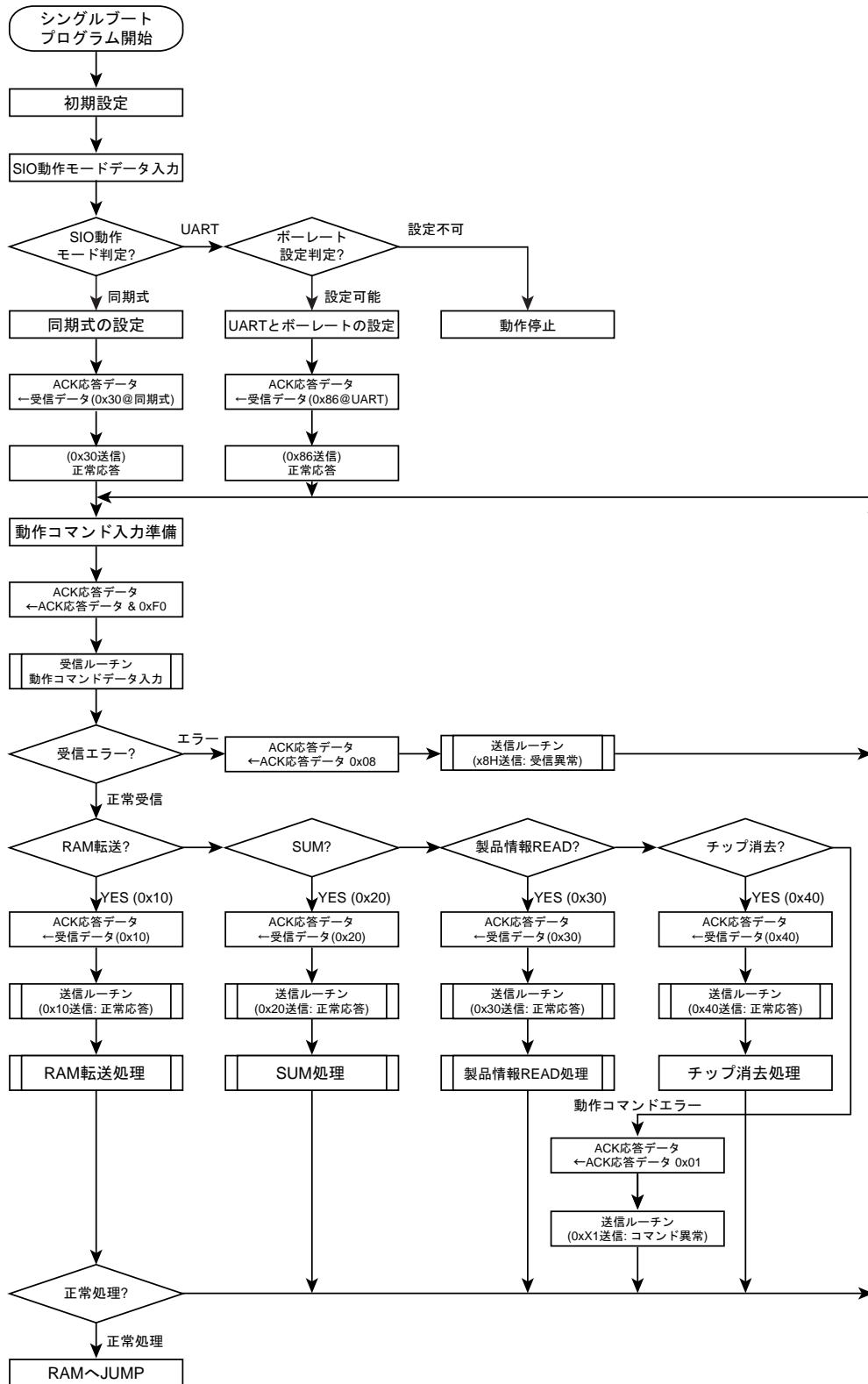


図 17-8 ブートプログラム全体フローチャート

## 17.3 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、フラッシュメモリの書き込み/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。

### 17.3.1 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。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。

書き込み、消去を行う場合、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 17-14 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	データ書き込みを自動で行います。
自動チップ消去	フラッシュメモリの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。

#### 17.3.1.1 ブロック構成

##### (1) TMPM330FDFG

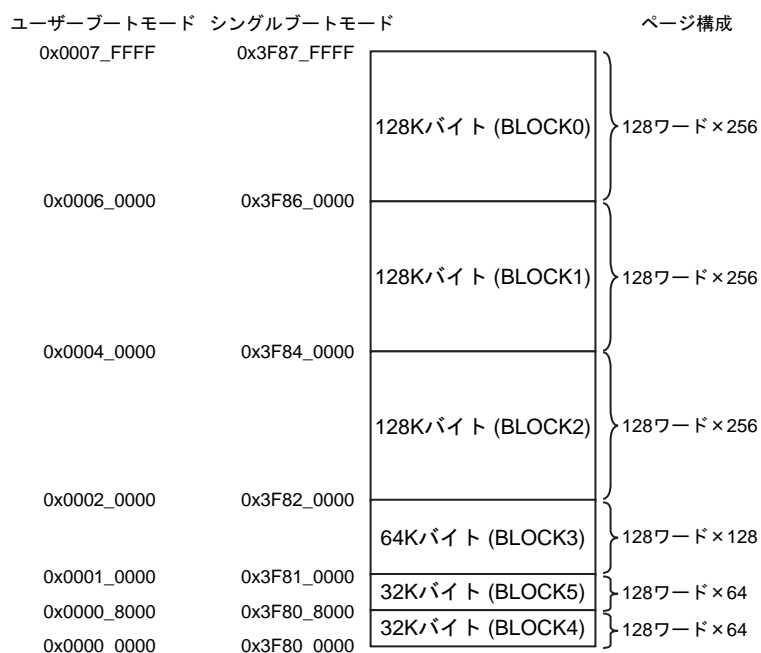


図 17-9 ブロック構成(TMPM330FDFG)

(2) TMPM330FYFG

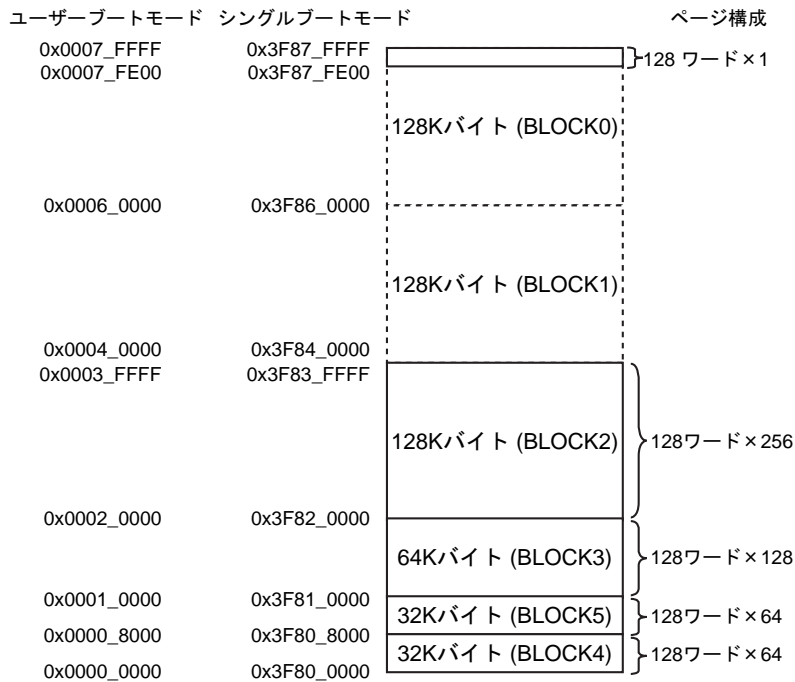


図 17-10 ブロック構成(TMPM330FYFG)

注) TMPM330FYFG 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。

(3) TMPM330FWFG

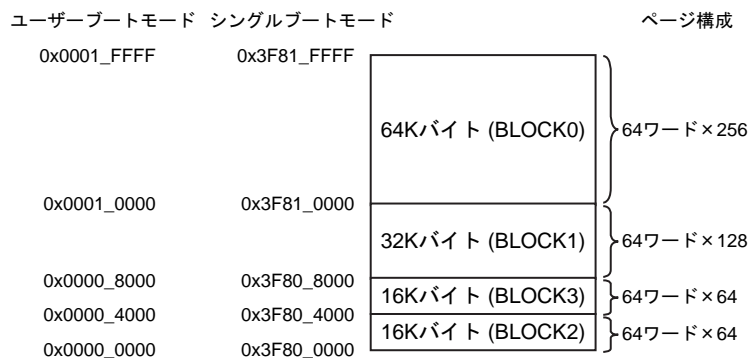


図 17-11 ブロック構成(TMPM330FWFG)

### 17.3.1.2 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、すべ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、リードモードに移りません。

#### (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/リセットコマンドおよび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。

フラッシュメモリに対する32ビット(ワード)のデータ転送命令を"バスライトサイクル"と呼びます。各コマンドはいくつかのバスサイクルで構成されています。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、すべての割り込み(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および、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、コマンドシーケンス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あります。

- 注 3) コマンドシーケンス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 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。

### 17.3.1.3 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  $\overline{\text{RESET}}$  入力端子が "Low" 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、「17.2.1 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。

### 17.3.1.4 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、TMPM330FDFG/TMPM330FYFG では 128 ワード、TMPM330FWFG では 64 ワードです。1 ページ 128 ワードの場合アドレス[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、1 ページ 64 ワードの場合アドレス[31:8]が同じで、先頭アドレス[7:0]=0、最後のアドレス[7:0]=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 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の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、あるページの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 FCFLCS<RDY/BSY>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えませんが注意してください。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、ページごと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できません。

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>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。

自動チップ消去動作は、FCFLCS<RDY/BSY>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>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できます。Block とプロテクトビットの関係は表 17-18 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 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。すべてのプロテクトビットをプログラムした後は、FCFLCS レジスタのすべての<BLPRO>ビットが"1"になっています。これ以降はすべての Block に対し、ライト/消去はできません。

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 レジスタのすべての<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「プロテクト/セキュリティ機能」の章を参照してください。

注) TMPM330FYFG 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" (すべ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、引き続いて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。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、自動チップ消去ある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" (すべ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



プロテクトビットを消去することで、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 17-18 のように Block 単位でプロテクトビットをプログラムするのに対して消去は 4 ビット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ます。各ブロックのプロテクトビッ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。

注) 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できます。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。

## 17.3.1.5 フラッシュコントロール/ステータスレジスタ

Base Address = 0x4004\_0500

レジスタ名		Address(Base+)
セキュリティビットレジスタ	FCSECBIT	0x0000
Reserved	-	0x0004
フラッシュコントロールレジスタ	FCFLCS	0x0020
Reserved	-	0x0024
Reserved	-	0x0028

注) "Reserved"表記のアドレスにはアクセスしないでください。

## (1) FCFLCS(フラッシュコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	BLPRO5	BLPRO4	BLPRO3	BLPRO2	BLPRO1	BLPRO0
リセット後	0	0	(注 2)	(注 2)	(注 2)	(注 2)	(注 2)	(注 2)
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	RDY/BSY
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-22	-	R	リードすると"0"が読めます。
21-16	BLPRO5- BLPRO0	R	Block5 ~ 0 のプロテクト状態(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。
15-1	-	R	リードすると"0"が読めます。
0	RDY/BSY	R	Ready/Busy (注 1) 0: 自動動作中 1: 自動動作終了 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。

- 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ください。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でき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ください。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 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 2 ms 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) ビット 21, 20 は TMPM330FWFG にはありません。リードすると"0"が読めます。

(2) FCSECBIT(セキュリティビットレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	SECBIT
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	SECBIT	R/W	セキュリティビット 0:セキュリティ機能設定不可 1:セキュリティビット設定可能

注) 本レジスタは、コールドリセットで初期化されます。

## 17.3.1.6 コマンドシーケンス一覧

各コマンドの、アドレスとデータを表 17-15 に示します。

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すべて"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。(表では、データの下位 8 ビットのデータのみ示しています。)

アドレスの詳細は、表 17-16 を参照してください。「コマンド」と記載された、Addr[15:8]に下記の値を使用します。

注) 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。

表 17-15 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

17.3.1.7 バスライトサイクル時のアドレスビット構成

表 17-16 は「表 17-15 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。

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、アドレス設定を行ってください。「"0"推奨」は適宜変更可能です。

表 17-16 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]
------	-----------------	--------------	--------------	--------------	--------------	--------------	-----------------	--------------	-------------	-------------	---------------

[TMPM330DFDG/FYFG/FWFG 共通]

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							

[TMPM330DFDG/FYFG]

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 17-17)						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 17-18)	"0"固定				プロテクトビ ット 選択 (表 17-18)	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 17-19)	"0"推奨				Addr[1:0] = "0"固定、 他ビットは"0"推奨				

[TMPM330FWFG]

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 17-17)						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 17-18)	"0"固定				プロテクトビ ット 選択 (表 17-18)	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 17-19)	"0"固定				Addr[1:0] = "0"固定、 他ビットは"0"推奨				

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。

表 17-17 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)
-------	----------------------	----------------------	----------------

## [TMPM330DFDG/FYFG]

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
1	0x0004_0000 ~ 0x0005_FFFF	0x3F84_0000 ~ 0x3F85_FFFF	128
0	0x0006_0000 ~ 0x0007_FFFF	0x3F86_8000 ~ 0x3F87_FFFF	128

## [TMPM330FWFG]

2	0x0000_0000 ~ 0x0000_3FFF	0x3F80_0000 ~ 0x3F80_3FFF	16
3	0x0000_4000 ~ 0x0000_7FFF	0x3F80_4000 ~ 0x3F80_7FFF	16
1	0x0000_8000 ~ 0x0000_FFFF	0x3F80_8000 ~ 0x3F80_FFFF	32
0	0x0001_0000 ~ 0x0001_FFFF	0x3F81_0000 ~ 0x3F81_FFFF	64

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 17-18 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]

## [TMPM330DFDG/FYFG]

Block0	<BLPRO[0]>	0	0	1	"0"固定	0	0	"0"推奨
Block1	<BLPRO[1]>	0	0	1		0	1	
Block2	<BLPRO[2]>	0	0	1		1	0	
Block3	<BLPRO[3]>	0	0	1		1	1	
Block4	<BLPRO[4]>	0	1	0		0	0	
Block5	<BLPRO[5]>	0	1	0		0	1	

## [TMPM330FWFG]

Block0	<BLPRO[0]>	0	0	"0"固定	0	0
Block1	<BLPRO[1]>	0	0		0	1
Block2	<BLPRO[2]>	0	0		1	0
Block3	<BLPRO[3]>	0	0		1	1

表 17-19 プロテクトビット消去アドレス表

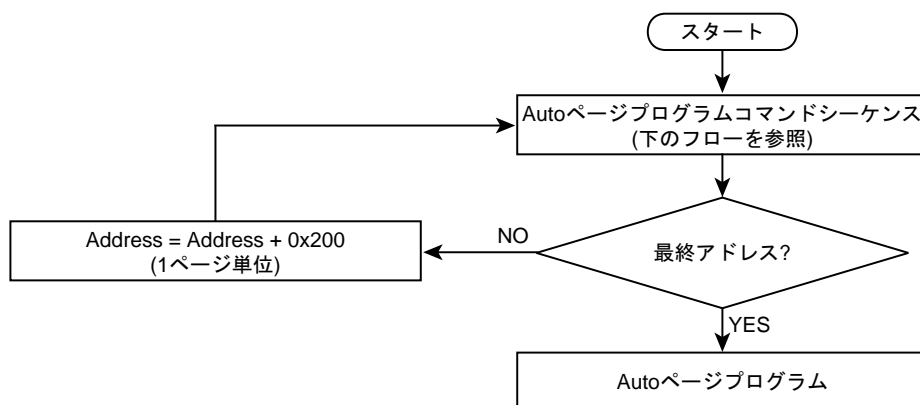
Block	プロテクトビット	第 7 バスライトサイクルのアドレス [18:17]	
		アドレス[18]	アドレス[17]
Block0 ~ 3	<BLPRO[0:3]>	0	0
Block4 ~ 5	<BLPRO[4:5]>	0	1

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できません。

表 17-20 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

IA[15:14]	ID[7:0]	Code
0y00	0x98	メーカーコード
0y01	0x5A	デバイスコード
0y10	Reserved	-
0y11	0x12 (TMPM330FDFG) 0x12 (TMPM330FYFG) 0x11 (TMPM330FWFG)	マクロコード

## 17.3.1.8 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)

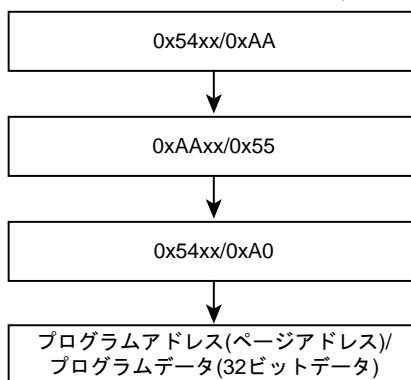


図 17-12 自動プログラム

注) 0x54xx は 0x55xx でもコマンドシーケンスを実行します。



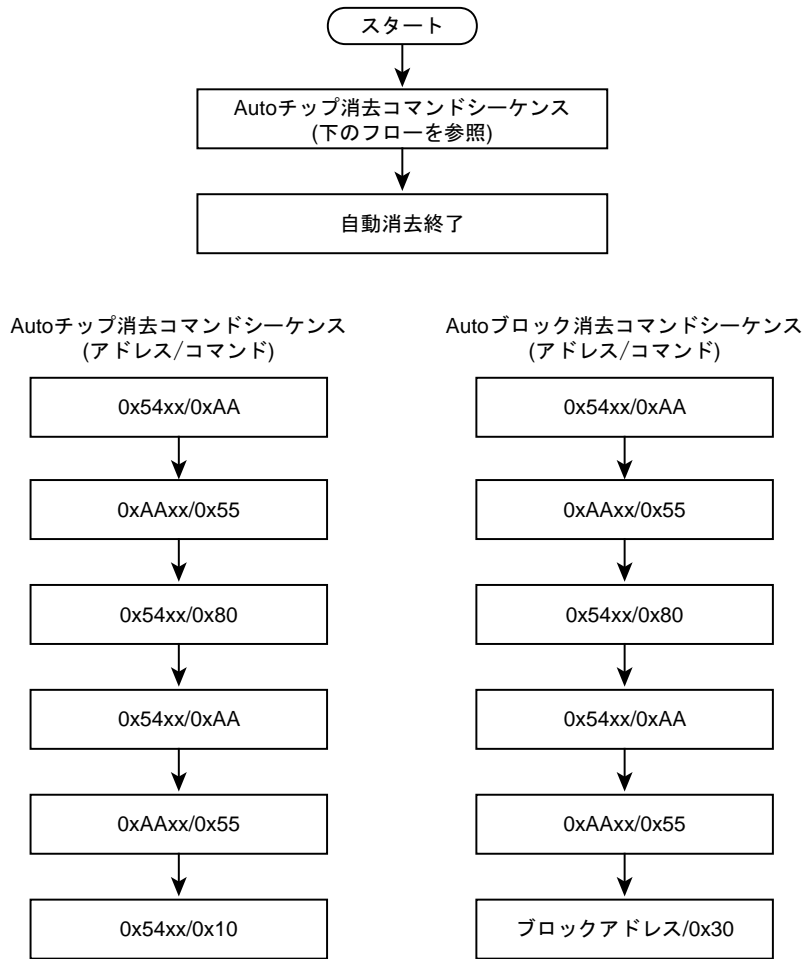


図 17-13 自動消去

注) 0x54xx は 0x55xx でもコマンドシーケンスを実行します。



## 第 18 章 プロテクト/セキュリティ機能

### 18.1 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、およ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グ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

### 18.2 特長

#### 18.2.1 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<BLPRO[5:0]>でモニタすることができます。

#### 18.2.2 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および、デバッグ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<SECBIT>が"1"にセットされている。
2. ライト/消去プロテクト用のすべてのプロテクトビット(FCFLCS<BLPRO>)が"1"にセットされている。

注) FCSECBIT<SECBIT>はコールドリセットで"1"にセットされます。

注) TMPM330FYFG 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。

セキュリティ機能が有効な状態の制限内容を、表 18-1 に示します。

表 18-1 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	CPU からの読み出しは可能です。
2) デバッグポート	JTAG/SW, トレースの通信ができなくなります。
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と、チップ消去が行われ、すべてのプロテクトビットも消去されます。

## 18.3 レジスタ

### 18.3.1 レジスタ一覧

Base Address = 0x4004\_0500

レジスタ名		Address(Base+)
セキュリティビットレジスタ	FCSECBIT	0x0000
Reserved	-	0x0004
フラッシュコントロールレジスタ	FCFLCS	0x0020
Reserved	-	0x0024
Reserved	-	0x0028

注) "Reserved"表記のアドレスにはアクセスしないでください。

## 18.3.2 FCFLCS(フラッシュコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	BLPRO5	BLPRO4	BLPRO3	BLPRO2	BLPRO1	BLPRO0
リセット後	0	0	(注 2)	(注 2)	(注 2)	(注 2)	(注 2)	(注 2)
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	RDY/BSY
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-22	-	R	リードすると"0"が読めます。
21-16	BLPRO5 - BLPRO0	R	Block5 ~ 0 のプロテクト状態(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。
15-1	-	R	リードすると"0"が読めます。
0	RDY/BSY	R	Ready/Busy フラグ(注 1) 0: 自動動作中 1: 自動動作終了 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。

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ください。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でき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ください。

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  $\mu$ 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 2 ms 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。

注 3) ビット 21, 20 は TMPM330FWFG にはありません。リードすると"0"が読めます。

## 18.3.3 FCSECBIT(セキュリティビットレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	SECBIT
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	SECBIT	R/W	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、コールドリセットで初期化されます。

## 18.4 設定/解除方法

### 18.4.1 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と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書き込みは1ブロックごと、消去はBlock0~3 と Block4~5 の2つの単位で行います。

全Blockすべてのプロテクトビットが"1"でかつ、FCSECBIT<SECBIT>が"1"の場合、セキュリティ機能が有効になっているので注意が必要です。この状態でプロテクトビットの消去を行うと、チップ消去を行った上でプロテクトビットすべてを消去します。このため、FCSECBIT<SECBIT>を"0"にしてからプロテクトビットの消去を行う必要があります。

コマンドシーケンスの詳細は「Flash 動作説明」の章を参照してください。

### 18.4.2 セキュリティビット

セキュリティ機能を有効にする FCSECBIT<SECBIT>は電源投入時のリセットで"1"にセットされます。FCSECBIT<SECBIT>の書き替えは以下の手順で行います。

1. FCSECBIT 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 転送命令で行ってください。





## 第 19 章 電気的特性

### 19.1 絶対最大定格

項目		記号	定格	単位
電源電圧		DVDD3	-0.3 ~ 3.9	V
		AVDD3	-0.3 ~ 3.9	
		RVDD3	-0.3 ~ 3.9	
入力電圧		V <sub>IN</sub>	-0.3 ~ VDD + 0.3	V
低レベル 出力電流	1 端子	I <sub>OL</sub>	5	mA
	合計	ΣI <sub>OL</sub>	50	
高レベル 出力電流	1 端子	I <sub>OH</sub>	-5	
	合計	ΣI <sub>OH</sub>	-50	
消費電力(Ta = 85 °C)		PD	600	mW
はんだ付け温度(10 s)		T <sub>SOLDER</sub>	260	°C
保存温度		T <sub>STG</sub>	-40 ~ 125	°C
動作温度	Flash W/E 時を除く	T <sub>OPR</sub>	-20 ~ 85	°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。従って必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。

Ta = -20 ~ 85 °C

## 19.2 DC 電気的特性(1/3)

項目		記号	条件	Min	Typ. (注 1)	Max	単位	
電源電圧	DVDD3 = AVDD3 = RVDD3 (注 2) DVSS = AVSS = 0V	DVDD3 AVDD3 RVDD3	f <sub>OSC</sub> = 8 ~ 10 MHz f <sub>sys</sub> = 1 ~ 40 MHz f <sub>s</sub> = 30 ~ 34 kHz	2.7	-	3.6	V	
低レベル 入力電圧	PC0~3, PD4~7	V <sub>IL1</sub>	2.7 V ≤ AVDD3 ≤ 3.6 V	-0.3	-	0.3 AVDD3	V	
	PD0~3	V <sub>IL2</sub>				0.2 AVDD3		
	PA2~7, PB0, PB3~7, PE0, PE4, PF0, PG7, PI0~5, PJ4~5, PK1~2	V <sub>IL3</sub>	2.7 V ≤ DVDD3 ≤ 3.6 V			0.3 DVDD3		
	PA0~1, PB1~2, PE1~3, PE5~6, PF1~7, PG0~6, PH0~7, PI6~7, PJ0~3, PJ6~7 RESET, NMI, MODE	V <sub>IL4</sub>				0.2 DVDD3		
	PK0	V <sub>IL5</sub>				0.1 DVDD3		
	X1	V <sub>IL6</sub>						
高レベル 入力電圧	PC0~3, PD4~7	V <sub>IH1</sub>	2.7 V ≤ AVDD3 ≤ 3.6 V	0.7 AVDD3	-	AVDD3 + 0.3	V	
	PD0~3	V <sub>IH2</sub>		0.8 AVDD3				
	PA2~7, PB0, PB3~7, PE0, PE4, PF0, PG7, PI0~5, PJ4~5, PK1~2	V <sub>IH3</sub>	2.7 V ≤ DVDD3 ≤ 3.6 V	0.7 DVDD3		DVDD3 + 0.3		
	PA0~1, PB1~2, PE1~3, PE5~6, PF1~7, PG0~6, PH0~7, PI6~7, PJ0~3, PJ6~7 RESET, NMI, MODE	V <sub>IH4</sub>		0.8 DVDD3				
	PK0	V <sub>IH5</sub>		0.9 DVDD3		3.6		
	X1	V <sub>IH6</sub>				DVDD3 + 0.3		
低レベル出力電圧	V <sub>OL</sub>	I <sub>OL</sub> = 2 mA	DVDD3 ≥ 2.7 V	-	-	0.4	V	
高レベル出力電圧	V <sub>OH</sub>	I <sub>OH</sub> = -2 mA	DVDD3 ≥ 2.7 V	2.4	-	-	V	
入力リーク 電流	PK0 以外	I <sub>LI1</sub>	0.0 ≤ V <sub>IN</sub> ≤ DVDD3 0.0 ≤ V <sub>IN</sub> ≤ AVDD3	-	0.02	±5	μA	
	PK0	I <sub>LI2</sub>	0.0 V ≤ V <sub>IN</sub> ≤ 3.6 V					
出力リーク 電流	PK0 以外	I <sub>LO1</sub>	0.2 ≤ V <sub>IN</sub> ≤ DVDD3 - 0.2 0.2 ≤ V <sub>IN</sub> ≤ AVDD3 - 0.2	-	0.05	±10		
	PK0	I <sub>LO2</sub>	0.2 V ≤ V <sub>IN</sub> ≤ 3.4 V					
リセットプルアップ抵抗	RRST	DVDD3 = 2.7 V ~ 3.6 V		30	50	150		kΩ
シュミット入力幅	VTH	2.7 V ≤ DVDD3 ≤ 3.6 V		0.3	0.6	-		V
プログラマブルプルアップ/ダウン抵抗	PKH	DVDD3 = 2.7 V ~ 3.6 V		-	50	150	kΩ	
Pin 容量(電源端子を除く)	C <sub>IO</sub>	f <sub>c</sub> = 1 MHz		-	-	10	pF	

注 1) Typ 値は特に指定のない限り Ta = 25 °C, DVDD3 = RVDD3 = AVDD3 = 3.3 V の値です。

注 2) DVDD3, AVDD3, RVDD3 は同電圧で使用してください。

19.3 DC 電気的特性(2/3)

DVDD3 = AVDD3 = RVDD3 = 2.7 V ~ 3.6 V, Ta = -20 ~ 85 °C

項目	記号	条件	Min	Typ.	Max	単位
低レベル出力電流	$I_{OL}$	1 端子ごと	-	-	2	mA
	$\Sigma I_{OL1}$	グループ単位 GrL1 = 6~12,89pin <PE3 / PG3,7 / PH4~5 / PJ5 / PK2> GrL2 = 15~37pin <PB3~4 / PE0~2,4~6 / PF0~2,7, PG0~2,4~6 / PH0~3> GrL3 = 15=61pin <PA0,1 / PB0~2,5~7 / PF4~6 / PI0~5 / PJ1,6,7 / PK0,1> GrL4 = 15~37pin <PA2~7 / PF3 / PH6,7 / PI6,7 / PJ0,2~4>	-	-	18	mA
	$\Sigma I_{OL2}$	全端子	-	-	35	mA
高レベル出力電流	$I_{OH}$	1 端子ごと	-	-	-2	mA
	$\Sigma I_{OH1}$	グループ単位 GrL1 = 6~12,89pin <PE3 / PG3,7 / PH4,5 / PJ5 / PK2> GrL2 = 15~37pin <PB3,4 / PE0~2,4~6 / PF0~2,7, PG0~2,4~6 / PH0~3> GrL3 = 15=61pin <PA0,1 / PB0~2,5~7 / PF4~6 / PI0~5 / PJ1,6,7 / PK0,1> GrL4 = 15~37pin <PA2~7 / PF3 / PH6,7 / PI6,7 / PJ0,2~4>	-	-	-18	mA
	$\Sigma I_{OH2}$	全端子	-	-	-35	mA

注) DVDD3, AVDD3, RVDD3 は同電圧で使用してください。

## 19.4 DC 電気的特性(3/3)

### 19.4.1 TMPM330FDFG/TMPM330FYFG

DVDD3 = AVDD3 = RVDD3 = 2.7 V ~ 3.6 V, Ta = -20 ~ 85 °C

項目	記号	条件	Min	Typ. (注 1)	Max	単位
NORMAL (注 2) ギア比 1/1	I <sub>DD</sub>	fsys = 40 MHz (fosc = 10 MHz)	-	32	42	mA
IDLE (注 3)			-	8	13	
SLOW		-	2.5	6		
SLEEP		fs = 32.768 kHz	-	55	950	μA
STOP		-	-	55	900	

注 1) Typ 値は特に指定のない限り Ta = 25 °C, DVDD3 = AVDD3 = RVDD3 = 3.3 V の値です。

注 2) I<sub>DD</sub> 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)

AD コンバータを除く内蔵周辺機能はすべて動作

注 3) I<sub>DD</sub> IDLE の測定条件:

周辺機能はすべて停止

I<sub>DD</sub> には DVDD3, AVDD3, RVDD3 に流れる電流が含まれます。

### 19.4.2 TMPM330FWFG

DVDD3 = AVDD3 = RVDD3 = 2.7 V ~ 3.6 V, Ta = -20 ~ 85 °C

項目	記号	条件	Min	Typ. (注 1)	Max	単位
NORMAL (注 2) ギア比 1/1	I <sub>DD</sub>	fsys = 40 MHz (fosc = 10 MHz)	-	26	32	mA
IDLE (注 3)			-	8	12.5	
SLOW		-	2.5	5.5		
SLEEP		fs = 32.768 kHz	-	55	900	μA
STOP		-	-	45	850	

注 1) Typ 値は特に指定のない限り Ta = 25 °C, DVDD3 = AVDD3 = RVDD3 = 3.3 V の値です。

注 2) I<sub>DD</sub> 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)

AD コンバータを除く内蔵周辺機能はすべて動作

注 3) I<sub>DD</sub> IDLE の測定条件:

周辺機能はすべて停止

I<sub>DD</sub> には DVDD3, AVDD3, RVDD3 に流れる電流が含まれます。

19.5 10ビットADコンバータ変換特性

DVDD3 = AVDD3 = RVDD3 = VREFH = 2.7 V ~ 3.6 V  
 AVSS = DVSS, Ta = -20 ~ 85 °C

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	AD変換時	IREF DVSS = AVSS	-	2.5	5.5	mA
	AD非変換時		-	±0.02	±5	µA
消費電流	AD変換時	-	IREFを除く	-	3	mA
積分非直線性誤差	-	AIN 負荷抵抗 ≤ 600 Ω AIN 負荷容量 ≤ 30 pF 変換時間 ≥ 1.15 µs	-	±2	±3	LSB
微分非直線性誤差			-	±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 ≤ 600 Ω AIN 負荷容量 ≤ 0.1 µF 変換時間 ≥ 1.15 µs	-	±2	±3	
微分非直線性誤差			-	±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 ≤ 10 kΩ AIN 負荷容量 ≥ 0.1 µF 変換時間 ≥ 2.30 µs	-	±2	±3	
微分非直線性誤差			-	±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]

注) ADコンバータ単体動作の時の特性です。

## 19.6 AC 電気的特性

### 19.6.1 AC 測定条件

本章に記載されている AC 特性は、特に指定のない限り以下の条件での測定結果です。

- ・ 出力レベル：High =  $0.8 \times DVDD3$ , Low =  $0.2 \times DVDD3$
- ・ 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
- ・ 負荷容量：CL = 30pF

### 19.6.2 シリアルチャネル(SIO/UART)

#### 19.6.2.1 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 fsys 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

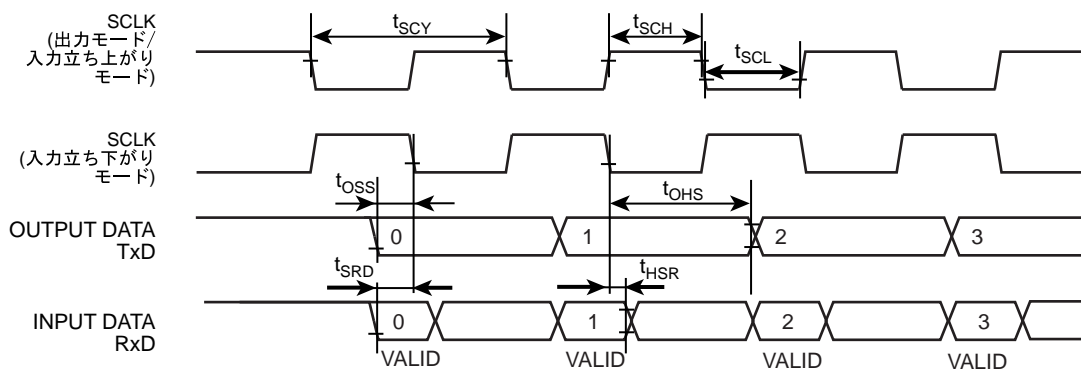
項目	記号	計算式		40 MHz		単位
		Min	Max	Min	Max	
SCLK クロック High 幅(入力)	t <sub>SCH</sub>	3x	-	75	-	ns
SCLK クロック Low 幅(入力)	t <sub>SCL</sub>	3x	-	75	-	
SCLK 周期	t <sub>SCY</sub>	t <sub>SCH</sub> + t <sub>SCL</sub>	-	150	-	
Output Data ← SCLK 立ち上がり/立ち下がり(注 1)	t <sub>OSS</sub>	t <sub>SCY</sub> /2 - 3x - 45	-	-45 (注 2)	-	
SCLK 立ち上がり → Output Data 保持/立ち下がり(注 1)	t <sub>OHS</sub>	t <sub>SCY</sub> /2	-	75	-	
有効 Data 入力 ← SCLK 立ち上がり/立ち下がり(注 1)	t <sub>SRD</sub>	30	-	30	-	
SCLK 立ち上がり → Input Data 保持/立ち下がり(注 1)	t <sub>HSR</sub>	x + 30	-	55	-	

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。

(2) SCLK 出力モード

項目	記号	計算式		40 MHz		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 Data 保持	$t_{HSR}$	0	-	0	-	



### 19.6.3 シリアルバスインタフェース(I2C/SIO)

#### 19.6.3.1 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 fsys 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 SBIxCR<SCK>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 <sub>SCL</sub>	0	-	0	100	0	400	kHz
スタートコンディション保持	t <sub>HD; STA</sub>	-	-	4.0	-	0.6	-	μs
SCL クロック Low 幅(入力) (注 1)	t <sub>LOW</sub>	-	-	4.7	-	1.3	-	μs
SCL クロック High 幅(入力) (注 2)	t <sub>HIGH</sub>	-	-	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 <sub>SU; STA</sub>	ソフト (注 5)	-	4.7	-	0.6	-	μs
データ保持時間(入力) (注 3, 4)	t <sub>HD; DAT</sub>	-	-	0.0	-	0.0	-	μs
データセットアップ時間	t <sub>SU; DAT</sub>	-	-	250	-	100	-	ns
ストップコンディションセットアップ時間	t <sub>SU; STO</sub>	-	-	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パスフリー時間	t <sub>BUF</sub>	ソフト (注 5)	-	4.7	-	1.3	-	μs

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):  $(2^{n-1} + 12)/x$

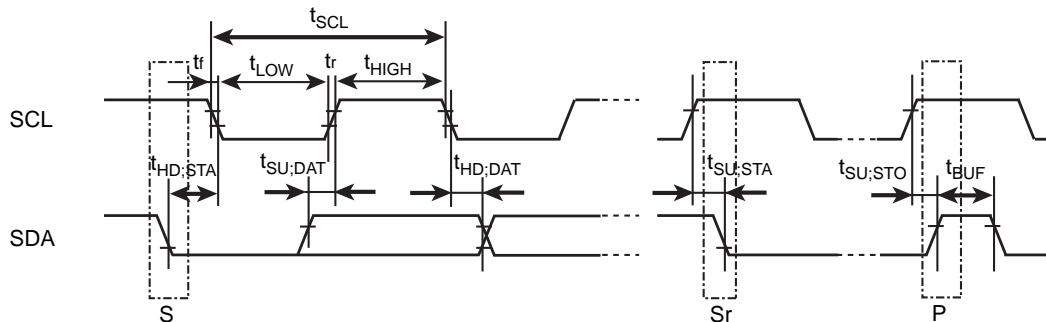
通信規格上、標準モード/高速モードの最高速は 100 kHz/400 kHz です。内部 SCL クロックの周波数の設定は、使用される fsys と上記注 1,注 2 の計算式にて設定されますのでご注意ください。

注 3) データ保持時間(出力)は内部 SCL から 12x の時間です。

注 4) フィリップス仕様では内部で、SDA 入力時にデータホールド 300nsec を確保して SCL 立ち下がり時の不安定状態を回避することになっていますが、本製品では対応していません。また SCL のエッジスローブコントロール機能を持っていません。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)を守るよう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) フィリップス仕様では、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。



S: スタートコンディション  
 Sr: 再スタートコンディション  
 P: ストップコンディション



19.6.3.2 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。

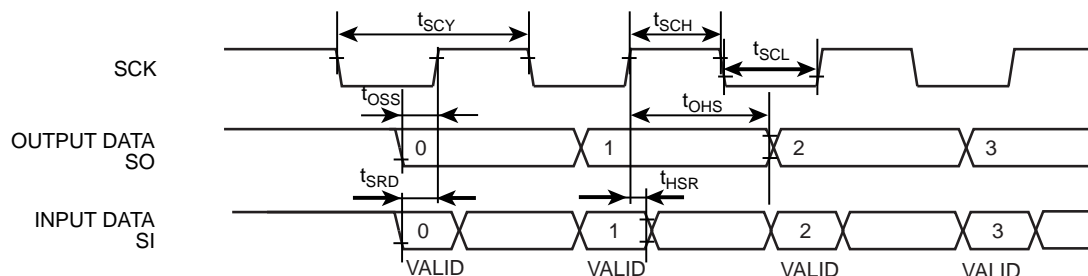
(1) SCK 入力モード(SCK デューティ 50%の場合)

項目	記号	計算式		40 MHz		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	$t_{OSS}$	$t_{SCY}/2 - 3x - 45$	-	-20 (注)	-	
SCK 立ち上がり → Output Data 保持	$t_{OHS}$	$t_{SCY}/2 + x$	-	125	-	
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	$t_{HSR}$	30	-	30	-	

注) 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。

(2) SCK 出力モード(SCK デューティ 50%の場合)

項目	記号	計算式		40 MHz		単位
		Min	Max	Min	Max	
SCK 周期(プログラマブル)	$t_{SCY}$	$16x$	-	400	-	ns
Output Data ← SCK 立ち上がり	$t_{OSS}$	$t_{SCY}/2 - 20$	-	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	$t_{SRD}$	45	-	45	-	
SCK 立ち上がり → Input Data 保持	$t_{HSR}$	0	-	0	-	



### 19.6.4 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		40 MHz		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

### 19.6.5 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		40 MHz		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

### 19.6.6 外部割り込み

表中の x はシステムクロック  $f_{sys}$  の周期を表します。

#### 1. STOP 解除割り込み以外

項目	記号	計算式		40 MHz		単位
		Min	Max	Min	Max	
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

#### 2. STOP 解除割り込み

項目	記号	Min	Max	単位
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

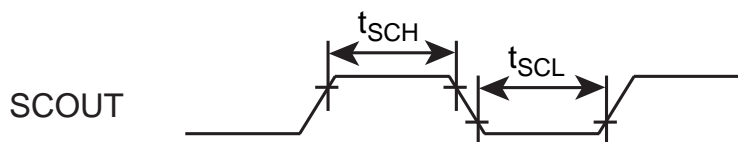
19.6.7  $\overline{\text{NMI}}$

項目	記号	Min	Max	単位
NMI 低レベルパルス幅	$t_{\text{NTCL}}$	100	-	ns

19.6.8 SCOUT 端子 AC 特性

項目	記号	計算式		40 MHz		単位
		Min	Max	Min	Max	
高レベルパルス幅	$t_{\text{SCH}}$	$0.5T - 5$	-	7.5	-	ns
低レベルパルス幅	$t_{\text{SCL}}$	$0.5T - 5$	-	7.5	-	ns

注) 表中の「T」は SCOUT 出力波形の周期を示します。



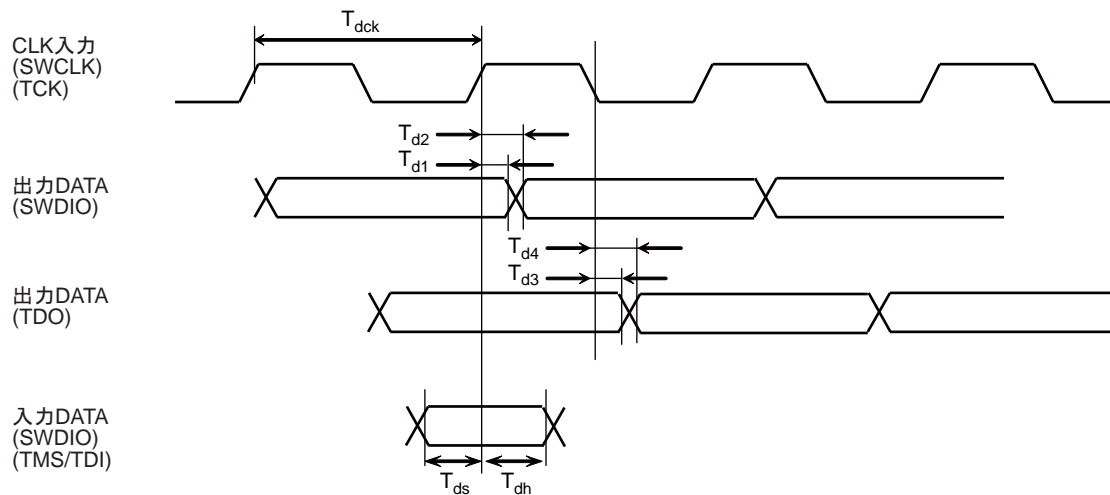
### 19.6.9 デバッグ通信

#### 19.6.9.1 SWD インタフェース

項目	記号	Min	Max	単位
CLK 周期	$T_{dck}$	83.33	-	ns
CLK 立ち上がり → 出力データ保持	$T_{d1}$	4	-	ns
CLK 立ち上がり → 出力データ有効	$T_{d2}$	-	30	ns
入力データ有効 ← CLK 立ち上がり	$T_{ds}$	20	-	ns
CLK 立ち上がり → 入力データ保持	$T_{dh}$	15	-	ns

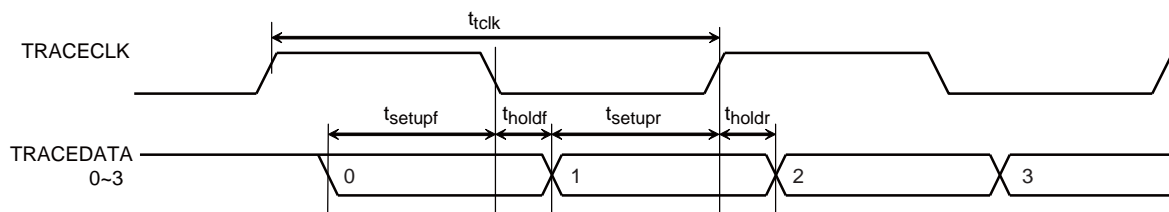
#### 19.6.9.2 JTAG インタフェース

項目	記号	Min	Max	単位
CLK 周期	$T_{dck}$	100	-	ns
CLK 立ち下がり → 出力データ保持	$T_{d3}$	4	-	ns
CLK 立ち下がり → 出力データ有効	$T_{d4}$	-	50	ns
入力データ有効 ← CLK 立ち上がり	$T_{ds}$	20	-	ns
CLK 立ち上がり → 入力データ保持	$T_{dh}$	15	-	ns



### 19.6.10 ETM トレース

項目	記号	Min	Max	単位
TRACECLK 周期	$t_{clk}$	50	-	ns
TRACEDATA 有効 ← TRACECLK 立ち上がり	$t_{setupr}$	2	-	ns
TRACECLK 立ち上がり → TRACEDATA 保持	$t_{holdr}$	1	-	ns
TRACEDATA 有効 ← TRACECLK 立ち下がり	$t_{setupf}$	2	-	ns
TRACECLK 立ち下がり → TRACEDATA 保持	$t_{holdf}$	1	-	ns



## 19.7 フラッシュ特性

### 19.7.1 書き込み特性

項目	条件	Min	Typ.	Max	単位
フラッシュメモリ書き換え保証回数	DVDD3 = AVDD3 = RVDD3 = 2.7 V ~ 3.6 V Ta = 0 ~ 70 °C	-	-	100	回

## 19.8 発振回路

発振子の接続回路例を以下に示します。

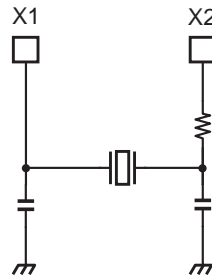


図 19-1 高周波発振回路例

注) 発振の安定には、発振子の位置、負荷容量を適切にする必要があります。これらは基板パターンにより大きな影響を受けます。安定した発振を得るため、ご使用される基板での評価をされるようお願いいたします。

本製品は、下記のメーカーの発振子を用いて評価しています。発振回路設計時に発振子の選択に活用願います。

### 19.8.1 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用いて評価しています。

(株)村田製作所の製品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>

### 19.8.2 水晶発振子

本製品は京セラキンセキ(株)製水晶発振子を用いて評価しています。

京セラキンセキ(株)の製品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>

## 19.9 取り扱い上のご注意

### 19.9.1 はんだ濡れ性についての注意事項

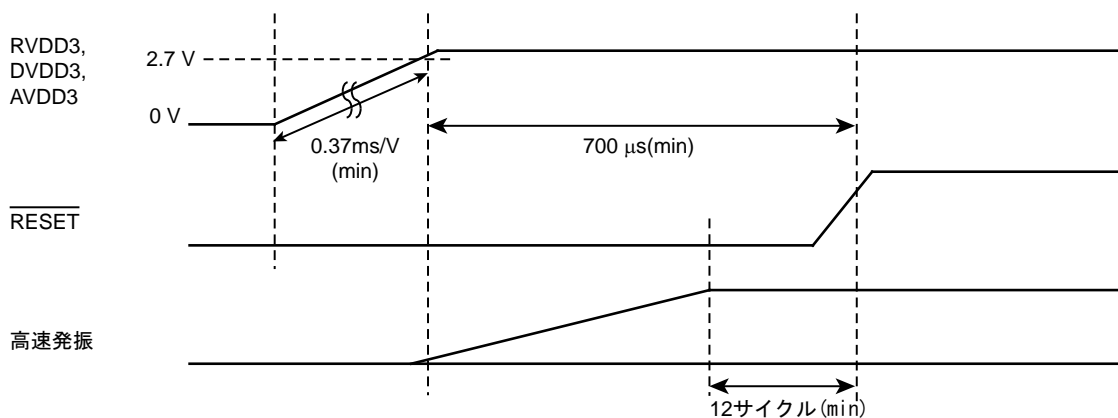
試験項目	試験条件	条件
はんだ付け性	230°C5 秒間 1 回 R タイプフラックス使用 (Sn-37Pb 鉛はんだ使用時)	フォーミングまでの半田付着 率 95%を良品とする
	245°C5 秒間 1 回 R タイプフラックス使用 (Sn-3.0Ag-0.5Cu はんだ使用時)	

### 19.9.2 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 $\mu$ 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。





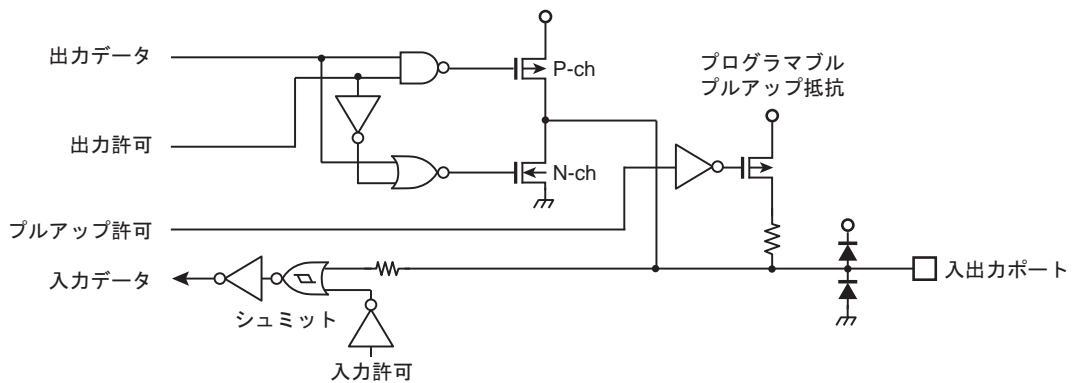


## 第 20 章 ポート部等価回路図

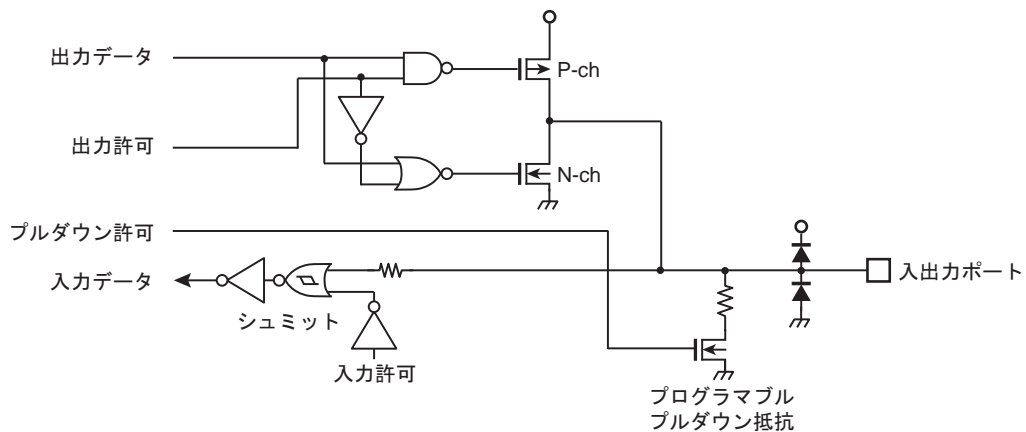
基本的に標準 CMOS ロジック IC 「74HCxx」 シリーズと同じゲート記号を使って書かれています。

入力保護抵抗は、数十  $\Omega$  ~ 数百  $\Omega$  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。

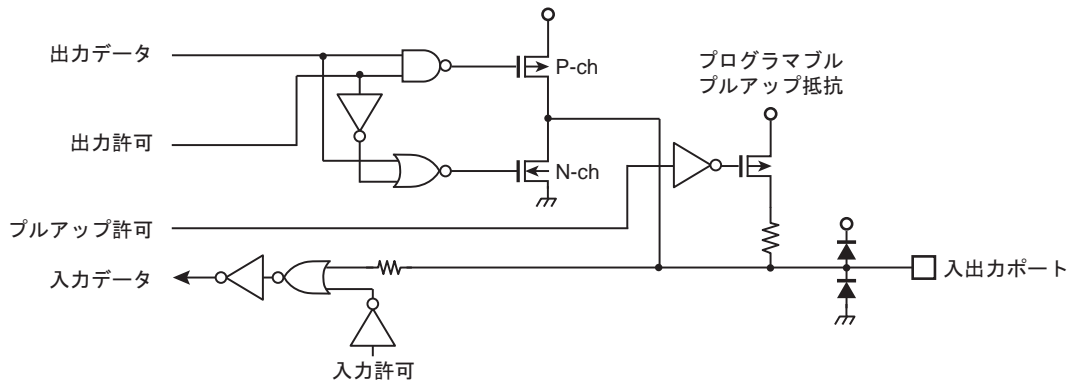
### 20.1 PA0, PB1~2, PE1~3, PE5~6, PF1~7, PG0~6, PH0~7, PI6~7, PJ0~3, PJ6~7



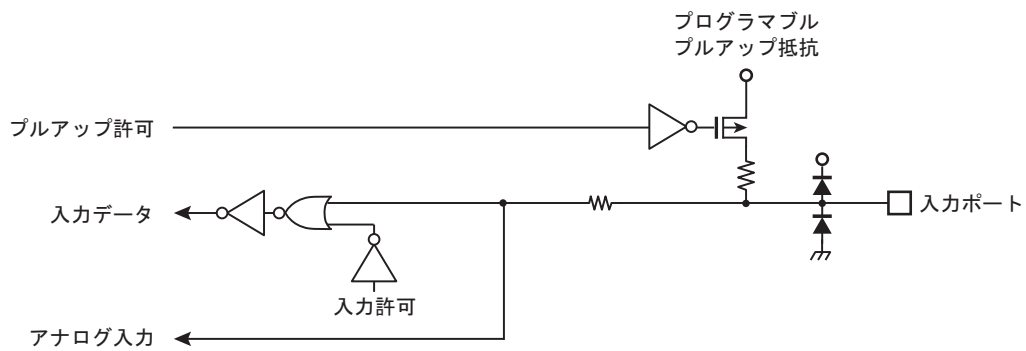
### 20.2 PA1



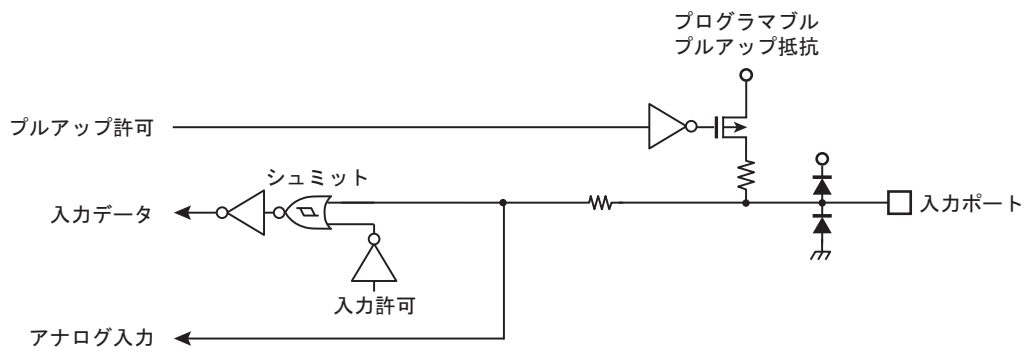
### 20.3 PA2~7,PB0,PB3~7,PE0,PE4,PF0,PG7,PIO~5,PJ4~5,PK1~2



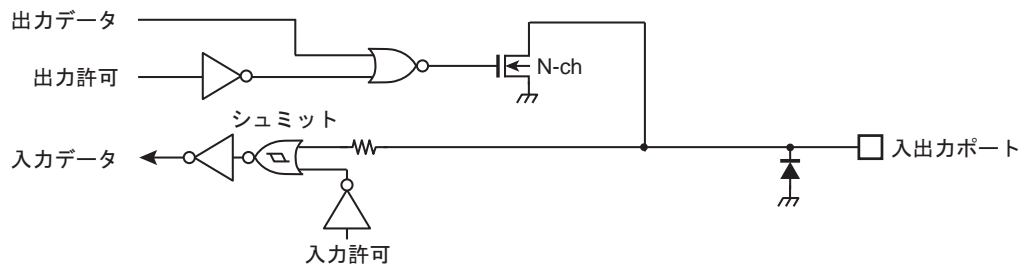
### 20.4 PC0~3, PD4~7



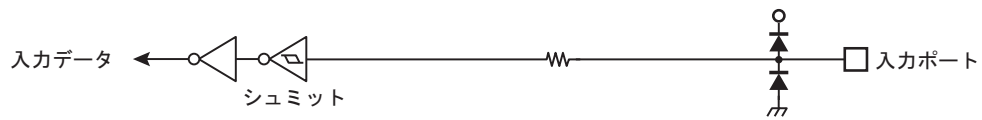
### 20.5 PD0~3



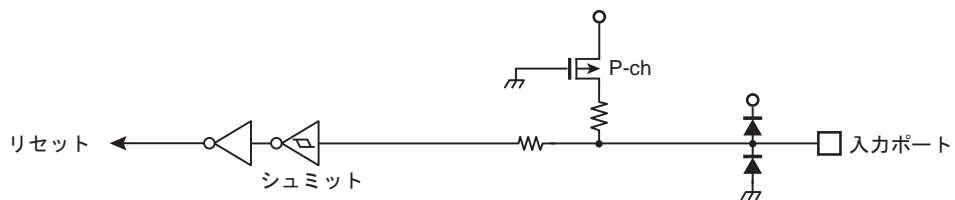
20.6 PK0



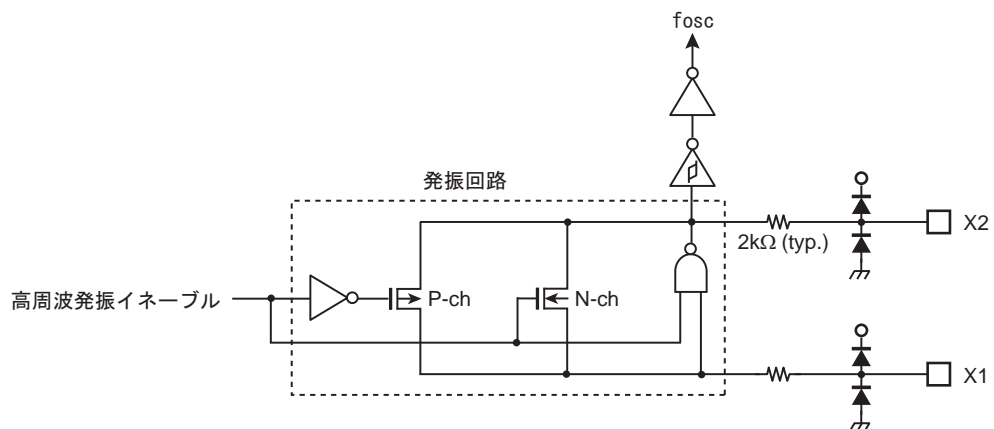
20.7  $\overline{\text{NMI}}$ , MODE



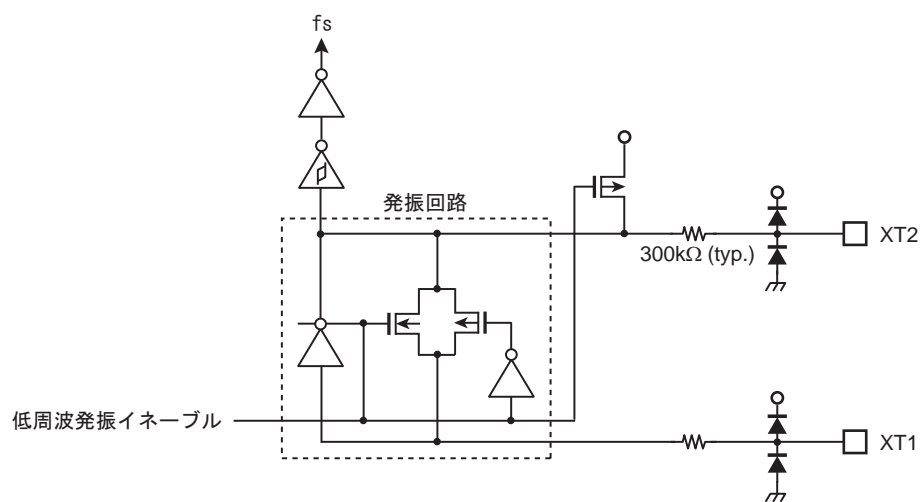
20.8  $\overline{\text{RESET}}$



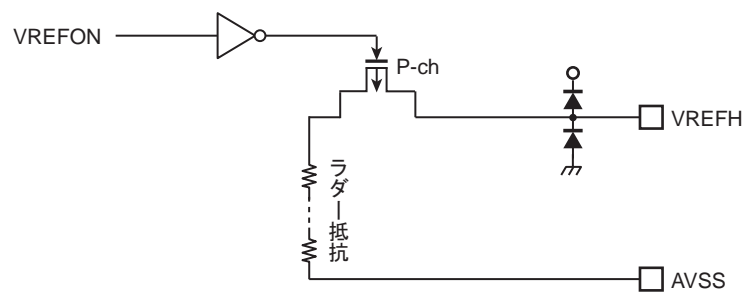
## 20.9 X1, X2



## 20.10 XT1, XT2



## 20.11 VREFH, AVSS

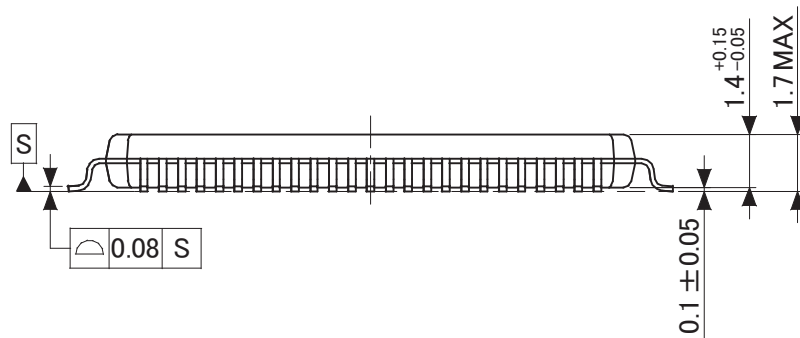
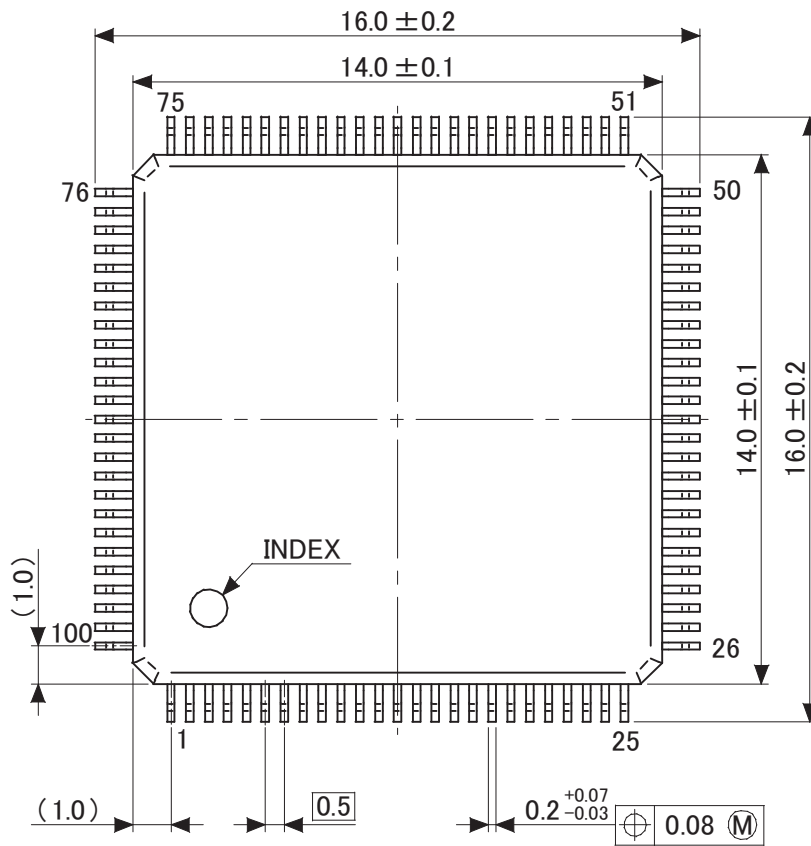


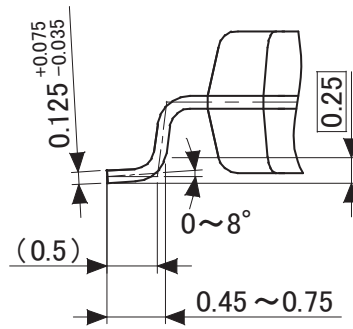
# 第 21 章 パッケージ寸法図

パッケージ型名 : LQFP100-P-1414-0.50H

Unit: mm

外形寸法図



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、一般的電子機器（コンピュータ、パーソナル機器、事務機器、計測機器、産業用ロボット、家電機器など）または本資料に個別に記載されている用途に使用されることが意図されています。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。本資料に個別に記載されている場合を除き、本製品を特定用途に使用しないでください。
- 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弊社営業窓口までお問合せください。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。

