TOSHIBA

32 ビット RISC マイクロコントローラ
TX04 シリーズ

TMPM470FDFG/FZFG/FYFG

株式会社 東芝

セミコンダクター & ストレージ社

©2026 Toshiba Corporation

****
ARM, Cortex および Thumb はARM Limited(またはその子会社)のEUまたはその他の国における
登録商標です。All rights reserved.
****

製品ご使用上の注意点について

本資料に掲載されている製品について、使用上の注意点を説明します。

なお、本項目と本資料での記述について、異なる場合は、本資料の記述が優先されます。

1. 電源投入時の動作について

   電源投入時、本資料に掲載されている製品の内部は不定状態となります。

   このため、リセットが有効となるまで、端子の状態は不定となります。

   外部リセット端子を使用してリセットする製品の場合、電源投入後外部から入力するリセットが有効になるまでの間、端子の状態は不定となります。

   また、内蔵パワーオンリセットを使用してリセットする製品の場合、電源電圧が電源投入から内蔵パワーオンリセットが有効となる電圧に上昇するまでの間、端子の状態は不定となります。

2. 未使用端子の処置について

   本資料に掲載されている製品では、未使用の入出力ポートは、入出力禁止となり、端子はハイインピーダンスです。一般にハイインピーダンスの端子を開放状態で製品を動作させると、外部からのノイズを受け誘起電圧が発生して LSI 内部で静電破壊やラッチアップが発生することがあります。

   未使用端子については、1 本ずつ、抵抗を通して電源端子または GND 端子に固定することを推奨します。

3. クロック発振の安定について

   リセットはクロック発振が安定してから解除してください。プログラム動作中にクロックを切り替える場合、切り替える先のクロック発振が安定している状態で切り替えてください。

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
2014/12/09	Tentative 1	First Release
-	-	-
2015/09/11	1	First Release
2015/10/13	2	Contents Revised
2015/12/24	2.1	Contents Revised
2016/02/18	2.2	Contents Revised
2016/03/10	2.3	Contents Revised
2021/10/15	2.4	Contents Revised
2022/03/31	2.5	Contents Revised
2022/06/01	2.6	Contents Revised
2023/07/14	2.7	Contents Revised
2023/07/31	2.8	Contents Revised

目　次

製品ご使用上の注意点について

TMPM470FDFG/FZFG/FYFG

1.1 機能概要 1
1.2 ブロック図 5
1.3 ピン配置図(Top view) 6
1.4 ピン名称と機能 7
1.4.1 機能端子名称と機能 7
1.4.1.1 周辺機能端子名称
1.4.1.2 デバッグ端子名称
1.4.1.3 制御端子名称
1.4.1.4 電源端子名称
1.4.1.5 電源間コンデンサ
1.4.2 ピン名称と機能 11
1.4.2.1 表の見方
1.4.2.2 PORT / デバッグ端子
1.4.2.3 制御端子
1.4.2.4 電源端子

第 2 章　製品情報

2.1 各周辺機能の情報 18
2.1.1 DMA コントローラ(DMAC) 18
2.1.2 16 ビットタイマ/イベントカウンタ(TMRB) 18
2.1.3 シリアルチャネル(SIO/UART) 19
2.1.4 シリアルバスインタフェース(I2C) 20
2.1.5 ベクトルエンジン (A-VE) 20
2.1.6 モータ制御回路 (PMD) 21
2.1.7 エンコーダ (A-ENC) 21
2.1.8 アナログ/デジタルコンバータ(ADC) 22
2.1.9 ウォッチドッグタイマ(WDT) 22
2.1.10 デバッグインタフェース 22

第 3 章　プロセッサコア

3.1 コアに関する情報 25
3.2 構成可能なオプション 25
3.3 例外/割り込み 26
3.3.1 割り込み本数 26
3.3.2 割り込み優先度ビット数 26
3.3.3 SysTick 26
3.3.4 SYSRESETREQ 26
3.3.5 LOCKUP 26
3.3.6 補助フォールトステータスレジスタ 26
3.4 イベント 27
3.5 電力管理 27

3.6 排他アクセス.....27
3.7 浮動小数点演算装置(FPU).....27

第 4 章 メモリマップ

4.1 メモリマップ.....29
4.2 バスマトリクス.....33
4.2.1 構成.....34
4.2.1.1 シングルチップモード
4.2.1.2 シングルブートモード
4.2.2 接続表.....36
4.2.2.1 Code 領域/ SRAM 領域
4.2.2.2 Peripheral 領域
4.2.3 周辺機能ベースアドレス一覧.....38

第 5 章 リセット動作

5.1 コールドリセット時.....42
5.1.1 POR によるリセット(RESET 端子を使用しない場合).....42
5.1.2 VLTD によるリセット(RESET 端子を使用しない場合).....43
5.1.3 RESET 端子によるリセット(POR によるリセットが有効な場合).....44
5.1.4 RESET 端子によるリセット(VLTD によるリセットが有効な場合).....45
5.2 ウォームリセット時.....46
5.2.1 リセット期間.....46
5.3 リセット解除後.....46

第 6 章 クロック/モード制御

6.1 特長.....47
6.2 レジスタ説明.....48
6.2.1 レジスタ一覧.....48
6.2.2 CGSYSCR(システムコントロールレジスタ).....49
6.2.3 CGOSCCR(発振コントロールレジスタ).....50
6.2.4 CGSTBYCR(スタンバイコントロールレジスタ).....52
6.2.5 CGPLLSEL(PLL セレクトレジスタ).....53
6.3 クロック制御.....54
6.3.1 クロックの種類.....54
6.3.2 リセット動作による初期値.....54
6.3.3 クロック系統図.....55
6.3.4 クロック逓倍回路(PLL).....56
6.3.4.1 安定時間
6.3.4.2 PLL 設定シーケンス
6.3.5 ウォーミングアップ機能.....58
6.3.6 システムクロック.....60
6.3.7 プリスケーラクロック.....61
6.4 動作モードとモード遷移.....62
6.4.1 動作モード状態遷移.....62
6.5 動作モード.....62
6.5.1 NORMAL モード.....62
6.6 低消費電力モード.....63
6.6.1 IDLE モード.....63
6.6.2 STOP モード.....64
6.6.3 低消費電力モードの選択.....65
6.6.4 各モードにおける動作状態.....65
6.6.5 低消費電力モードの解除.....65
6.6.6 ウォーミングアップ.....67

6.6.7 モード遷移によるクロック動作....68
6.6.7.1 NORMAL → STOP → NORMAL 動作モード遷移

第 7 章 内蔵高速発振調整機能

7.1 構成....69
7.2 レジスタ説明....70
7.2.1 レジスタ一覧....70
7.2.2 TRMOSCxPRO (プロテクトレジスタ)....70
7.2.3 TRMOSCxEN (イネーブルレジスタ)....71
7.2.4 TRMOSCxINIT (初期トリミング値モニタレジスタ)....72
7.2.5 TRMOSCxSET (トリミング値設定レジスタ)....73
7.3 動作説明....74
7.3.1 概要....74
7.3.2 調整範囲....74

第 8 章 例外

8.1 概要....75
8.1.1 種類....75
8.1.2 処理の流れ....76
8.1.2.1 例外要求と検出
8.1.2.2 例外の処理と割り込み処理ルーチンへの分岐(横取り)
8.1.2.3 割り込み処理ルーチンの発行
8.1.2.4 例外からの復帰
8.2 リセット例外....82
8.3 マスク不能割り込み(NMI)....83
8.4 SysTick....83
8.5 割り込み....84
8.5.1 要因....84
8.5.1.1 経路
8.5.1.2 割り込み要求の発生
8.5.1.3 割り込み要因の伝達
8.5.1.4 外部割り込み端子を使用する際の注意
8.5.1.5 要因一覧
8.5.1.6 アクティブレベル
8.5.2 処理詳細....89
8.5.2.1 処理の流れ
8.5.2.2 準備
8.5.2.3 検出(クロックジェネレータ)
8.5.2.4 検出(CPU)
8.5.2.5 CPU の処理
8.5.2.6 割り込み処理ルーチンでの処理(要因の取り下げ)
8.6 例外/割り込み関連レジスタ....95
8.6.1 レジスタ一覧....95
8.6.2 NVIC レジスタ....96
8.6.2.1 SysTick 制御およびステータスレジスタ
8.6.2.2 SysTick リロード値レジスタ
8.6.2.3 SysTick 現在値レジスタ
8.6.2.4 SysTick 較正値レジスタ
8.6.2.5 割り込み制御用レジスタ
8.6.2.6 割り込み優先度レジスタ
8.6.2.7 ベクタテーブルオフセットレジスタ
8.6.2.8 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
8.6.2.9 システムハンドラ優先度レジスタ
8.6.2.10 システムハンドラ制御および状態レジスタ
8.6.3 クロックジェネレータレジスタ....114
8.6.3.1 CG 割り込みモードコントロールレジスタ
8.6.3.2 CGICRCG(CG 割り込み要求クリアレジスタ)
8.6.3.3 CGRSTFLG(リセットフラグレジスタ)

第 9 章　デジタルノイズフィルタ回路(DNF)

9.1　構成.....121
9.2　レジスタ説明.....122
9.2.1　レジスタ一覧.....122
9.2.1.1　NFCKCR(ノイズフィルタ制御レジスタ)
9.2.1.2　NFENCR(ノイズフィルタ許可レジスタ)
9.3　動作説明.....125
9.3.1　構成.....125
9.3.2　動作.....125
9.3.3　使用可能な動作モード.....125
9.3.4　STOP モードを使用する場合の注意点.....125
9.3.5　最小ノイズ除去時間.....125

第 10 章　μDMA コントローラ（μDMAC）

10.1　概要.....127
10.1.1　機能一覧.....127
10.2　ブロック図.....128
10.3　レジスタ説明.....129
10.3.1　レジスタ一覧.....129
10.3.2　DMAxStatus (DMAC Status Register).....130
10.3.3　DMAxCfg (DMAC Configuration Register).....131
10.3.4　DMAxCtrlBasePtr (Channel control data base pointer Register).....132
10.3.5　DMAxAltCtrlBasePtr (Channel alternate control data base pointer Register).....132
10.3.6　DMAxChnlSwRequest(Channel software request Register).....133
10.3.7　DMAxChnlUseburstSet(Channel useburst set Register).....134
10.3.8　DMAxChnlUseburstClr(Channel useburst clear Register).....135
10.3.9　DMAxChnlReqMaskSet(Channel request mask set Register).....136
10.3.10　DMAxChnlReqMaskClr(Channel request mask clear Register).....137
10.3.11　DMAxChnlEnableSet(Channel enable set Register).....138
10.3.12　DMAxChnlEnableClr(Channel enable clear Register).....139
10.3.13　DMAxChnlPriAltSet(Channel primary-alternate set Register).....140
10.3.14　DMAxChnlPriAltClr(Channel primary-alternate clear Register).....141
10.3.15　DMAxChnlPrioritySet(Channel priority set Register).....142
10.3.16　DMAxChnlPriorityClr(Channel priority clear Register).....143
10.3.17　DMAxErrClr(Bus error clear Register).....144
10.3.18　DMAIFFLGx(DMA Flag Register).....145
10.4　動作説明.....146
10.4.1　チャネル制御データメモリマップ.....146
10.4.2　チャネル制御データの構造.....147
10.4.2.1　転送データ最終アドレス
10.4.2.2　転送先の最終アドレス
10.4.2.3　制御データ設定
10.4.3　動作モード.....149
10.4.3.1　無効
10.4.3.2　基本モード
10.4.3.3　自動要求モード
10.4.3.4　ピンポンモード
10.4.3.5　メモリスキャッターギャザーモード
10.4.3.6　周辺スキャッターギャザーモード
10.5　使用上の注意.....156
10.5.1　SIO/UART、TMRB、ADC を使用する場合.....156

第 11 章　入出力ポート

11.1　レジスタ説明.....157

11.1.1 レジスター覧.....157
11.1.2 ポート機能と設定一覧.....159
11.1.2.1 PORT A
11.1.2.2 PORT B
11.1.2.3 PORT C
11.1.2.4 PORT D
11.1.2.5 PORT E
11.1.2.6 PORT F
11.1.2.7 PORT G
11.1.2.8 PORT H
11.1.2.9 PORT J
11.1.2.10 PORT K
11.1.2.11 PORT L
11.1.2.12 PORT N
11.1.2.13 PORT P
11.2 ポート回路図.....171
11.2.1 タイプ FT1.....171
11.2.2 タイプ FT2.....172
11.2.3 タイプ FT3.....173
11.2.4 タイプ FT4.....174
11.2.5 タイプ FT5.....175
11.2.6 タイプ FT6.....176
11.2.7 タイプ FT7.....177
11.2.8 タイプ FT8.....178

第 12 章 16 ビットタイマ/イベントカウンタ(TMRB)

12.1 概要.....179
12.2 構成.....180
12.3 レジスタ説明.....181
12.3.1 レジスター覧.....181
12.3.2 TBxEN(イネーブルレジスタ).....182
12.3.3 TBxRUN(RUN レジスタ).....183
12.3.4 TBxCR(コントロールレジスタ).....184
12.3.5 TBxMOD(モードレジスタ).....185
12.3.6 TBxFFCR(フリップフロップコントロールレジスタ).....186
12.3.7 TBxST(ステータスレジスタ).....187
12.3.8 TBxIM(割り込みマスクレジスタ).....188
12.3.9 TBxUC(アップカウンタキャプチャレジスタ).....189
12.3.10 TBxRG0(タイマレジスタ 0).....190
12.3.11 TBxRG1(タイマレジスタ 1).....190
12.3.12 TBxCP0(キャプチャレジスタ 0).....191
12.3.13 TBxCP1(キャプチャレジスタ 1).....191
12.4 回路別の動作説明.....192
12.4.1 プリスケーラ.....192
12.4.2 アップカウンタ(UC).....192
12.4.2.1 ソースクロック
12.4.2.2 動作の開始と停止
12.4.2.3 カウンタクリアのタイミング
12.4.2.4 カウンタのオーバフロー
12.4.3 タイマレジスタ(TBxRG0, TBxRG1).....193
12.4.4 キャプチャ制御.....193
12.4.5 キャプチャレジスタ(TBxCP0, TBxCP1).....194
12.4.6 アップカウンタキャプチャレジスタ(TBxUC).....194
12.4.7 コンパレータ(CP0, CP1).....194
12.4.8 タイマフリップフロップ(TBxFF0).....194
12.4.9 キャプチャ割り込み(INTCAPx0, INTCAPx1).....194
12.5 モード別動作説明.....195
12.5.1 16 ビットインタバルタイマモード.....195
12.5.2 16 ビットイベントカウンタモード.....195
12.5.3 16 ビット PPG (プログラマブル矩形波)出力モード.....196
12.5.4 外部トリガ PPG(プログラマブル矩形波)出力モード.....198
12.6 キャプチャ機能を利用した応用例.....199
12.6.1 外部トリガパルスからのワンショットパルス出力.....199

12.6.2 パルス幅測定 201

第 13 章 4 バイト FIFO 付きシリアルチャネル(SIO/UART)

13.1 概要 203
13.2 構成 204
13.3 レジスタ説明 205
13.3.1 レジスター覧 205
13.3.2 SCxEN (イネーブルレジスタ) 206
13.3.3 SCxBUF (バッファレジスタ) 207
13.3.4 SCxCR (コントロールレジスタ) 208
13.3.5 SCxMOD0 (モードコントロールレジスタ 0) 210
13.3.6 SCxMOD1 (モードコントロールレジスタ 1) 211
13.3.7 SCxMOD2 (モードコントロールレジスタ 2) 212
13.3.8 SCxBRCR (ボーレートジェネレータコントロールレジスタ) 214
13.3.9 SCxBRADD (ボーレートジェネレータコントロールレジスタ 2) 215
13.3.10 SCxFCNF (FIFO コンフィグレジスタ) 216
13.3.11 SCxRFC (受信 FIFO コンフィグレジスタ) 218
13.3.12 SCxTFC (送信 FIFO コンフィグレジスタ) 219
13.3.13 SCxRST (受信 FIFO ステータスレジスタ) 220
13.3.14 SCxTST (送信 FIFO ステータスレジスタ) 221
13.4 動作モード 222
13.5 データフォーマット 223
13.5.1 データフォーマット一覧 223
13.5.2 パリティ制御 224
13.5.2.1 送信
13.5.2.2 受信
13.5.3 STOP ビット長 224
13.6 クロック制御 225
13.6.1 プリスケーラ 225
13.6.2 シリアルクロック生成回路 225
13.6.2.1 ボーレートジェネレータ
13.6.2.2 クロック選択回路
13.7 送信/受信バッファと FIFO 229
13.7.1 構成 229
13.7.2 送信/受信バッファ 229
13.7.3 送信バッファの初期化 230
13.7.4 FIFO 230
13.8 ステータスフラグ 231
13.9 エラーフラグ 231
13.9.1 OERR フラグ 231
13.9.2 PERR フラグ 232
13.9.3 FERR フラグ 232
13.10 受信 233
13.10.1 受信カウンタ 233
13.10.2 受信制御部 233
13.10.2.1 I/O インタフェースモードの場合
13.10.2.2 UART モードの場合
13.10.3 受信動作 233
13.10.3.1 受信バッファの動作
13.10.3.2 受信 FIFO の動作
13.10.3.3 I/O インタフェースモード、クロック出力モードでの受信
13.10.3.4 受信データの読み出し
13.10.3.5 ウエイクアップ機能
13.10.3.6 オーバランエラー
13.11 送信 237
13.11.1 送信カウンタ 237
13.11.2 送信制御部 237
13.11.2.1 I/O インタフェースモードの場合
13.11.2.2 UART モードの場合
13.11.3 送信動作 238
13.11.3.1 送信バッファの動作

13.11.3.2 送信 FIFO の動作
13.11.3.3 I/O インタフェースモード、クロック出力モードでの送信
13.11.3.4 I/O インタフェースモード時の最終ビット出力後の SCxTXD 端子の状態
13.11.3.5 アンダーランエラー
13.11.3.6 I/O インタフェースモード、クロック入力モードでのデータのホールド時間
13.12 ハンドシェイク機能 242
13.13 割り込み/エラー発生タイミング 243
13.13.1 受信割り込み 243
13.13.1.1 シングルバッファ/ダブルバッファ構成の場合
13.13.1.2 FIFO 使用の場合
13.13.2 送信割り込み 244
13.13.2.1 シングルバッファ/ダブルバッファ構成の場合
13.13.2.2 FIFO 使用の場合
13.13.3 エラー発生 245
13.13.3.1 UART モード
13.13.3.2 I/O インタフェースモード
13.14 DMA 要求 246
13.15 ソフトウエアリセット 247
13.16 モード別動作説明 248
13.16.1 モード 0 (I/O インタフェースモード) 248
13.16.1.1 送信
13.16.1.2 受信
13.16.1.3 送受信(全二重)
13.16.2 モード 1 (7 ビット UART モード) 259
13.16.3 モード 2 (8 ビット UART モード) 259
13.16.4 モード 3 (9 ビット UART モード) 260
13.16.4.1 ウェイクアップ機能
13.16.4.2 プロトコル

第 14 章 シリアルバスインタフェース(I2C/SIO)

14.1 構成 263
14.2 レジスタ説明 264
14.2.1 レジスター覧 264
14.3 I2C バスモード 265
14.3.1 I2C バスモード時のコントロールレジスタ 265
14.3.1.1 SBIxCR0(コントロールレジスタ 0)
14.3.1.2 SBIxCR1(コントロールレジスタ 1)
14.3.1.3 SBIxCR2(コントロールレジスタ 2)
14.3.1.4 SBIxSR(ステータスレジスタ)
14.3.1.5 SBIxBR0(ボーレートレジスタ 0)
14.3.1.6 SBIxDBR(データバッファレジスタ)
14.3.1.7 SBIxI2CAR(I2C バスアドレスレジスタ)
14.3.2 制御 273
14.3.2.1 動作モードの設定
14.3.2.2 シリアルクロック
14.3.2.3 アクノリッジメントモードの指定
14.3.2.4 転送ビット数の選択
14.3.2.5 スレーブアドレスとアドレス認識モードの設定
14.3.2.6 マスタ/スレーブの選択
14.3.2.7 トランスミッタ/レシーバの選択
14.3.2.8 バスビジーモニタ
14.3.2.9 割り込みサービス要求と解除
14.3.2.10 アービトレーションロスト検出モニタ
14.3.2.11 スレーブアドレス一致検出モニタ
14.3.2.12 ゼネラルコール検出モニタ
14.3.2.13 最終受信ビットモニタ
14.3.2.14 データバッファレジスタ(SBIxDBR)
14.3.2.15 ボーレートレジスタ(SBIxBR0)
14.3.2.16 ソフトウエアリセット
14.3.3 データ転送手順 279
14.3.3.1 デバイスの初期化
14.3.3.2 スタートコンディション, スレーブアドレスの発生
14.3.3.3 1 ワードのデータ転送
14.3.3.4 ストップコンディションの発生
14.3.3.5 再スタートの手順
14.3.4 データフォーマット 287

14.3.5 マルチマスタで使用する際の注意点 287
14.4 SIO モード 288
14.4.1 SIO モード時のコントロールレジスタ 288
14.4.1.1 SBIxCR0(コントロールレジスタ 0)
14.4.1.2 SBIxCR1(コントロールレジスタ 1)
14.4.1.3 SBIxDBR(データバッファレジスタ)
14.4.1.4 SBIxCR2(コントロールレジスタ 2)
14.4.1.5 SBIxSR(ステータスレジスタ)
14.4.1.6 SBIxBR0(ボーレートレジスタ 0)
14.4.2 制御 294
14.4.2.1 シリアルクロック
14.4.2.2 転送モード

第 15 章 12 ビットアナログ/デジタルコンバータ

15.1 機能と特徴 303
15.2 ブロック図 304
15.3 レジスタ一覧 305
15.4 レジスタ詳細 307
15.4.1 ADxCLK (変換クロック設定レジスタ) 307
15.4.2 ADxMOD0 (モード設定レジスタ 0) 309
15.4.3 ADxMOD1 (モード設定レジスタ 1) 310
15.4.4 ADxMOD2 (モード設定レジスタ 2) 311
15.4.5 ADxMOD3 (モード設定レジスタ 3) 312
15.4.6 ADxMOD4 (モード設定レジスタ 4) 313
15.4.7 ADxMOD5 (モード設定レジスタ 5) 314
15.4.8 ADxCMPCR0(監視割り込み設定レジスタ 0) 315
15.4.9 ADxCMPCR1(監視割り込み設定レジスタ 1) 316
15.4.10 ADxCMP0(変換結果比較レジスタ 0) 317
15.4.11 ADxCMP1(変換結果比較レジスタ 1) 317
15.4.12 ADxREG0(変換結果格納レジスタ 0) 318
15.4.13 ADxREG1(変換結果格納レジスタ 1) 319
15.4.14 ADxREG2(変換結果格納レジスタ 2) 320
15.4.15 ADxREG3(変換結果格納レジスタ 3) 321
15.4.16 ADxREG4(変換結果格納レジスタ 4) 322
15.4.17 ADxREG5(変換結果格納レジスタ 5) 323
15.4.18 ADxREG6(変換結果格納レジスタ 6) 324
15.4.19 ADxREG7(変換結果格納レジスタ 7) 325
15.4.20 ADxREG8(変換結果格納レジスタ 8) 326
15.4.21 ADxREG9(変換結果格納レジスタ 9) 327
15.4.22 ADxREG10(変換結果格納レジスタ 10) 328
15.4.23 ADxREG11(変換結果格納レジスタ 11) 329
15.4.24 PMD トリガ用プログラムレジスタ 330
15.4.24.1 ADxPSEL0 ~ ADxPSEL11(PMD トリガ用プログラム番号選択レジスタ 0 ~ 11)
15.4.24.2 ADxPINTS0 ~ 5(PMD トリガ用割り込み選択レジスタ 0 ~ 5)
15.4.24.3 ADxPSET0 ~ 5(PMD トリガ用プログラム選択レジスタ 0 ~ 5)
15.4.25 ADxTSET03 / ADxTSET47 / ADxTSET811(タイマトリガ用プログラムレジスタ) 349
15.4.26 ADxSSET03 / ADxSSET47 / ADxSSET811(ソフトウエアトリガ用プログラムレジスタ) 353
15.4.27 ADxASET03 / ADxASET47 / ADxASET811(常時変換用プログラムレジスタ) 357
15.5 動作説明 361
15.5.1 アナログ基準電圧 361
15.5.2 AD 変換開始 361
15.5.3 AD 監視機能 362
15.6 AD 変換タイミングチャート 363
15.6.1 ソフトウエア AD 変換 363
15.6.2 常時 AD 変換 364
15.6.3 トリガによる AD 変換開始 365
15.7 使用方法の例 367
15.7.1 PMD (3 シャント)、AD コンバータ × 1、順次変換方式 367
15.7.2 PMD A(3 シャント)、AD コンバータ × 2、同時変換方式 368
15.7.3 PMD 0(3 シャント)、PMD 1(1 シャント)、AD コンバータ × 2、順次変換方式 369
15.7.4 PMD (1 シャント)、AD コンバータ × 1、順次変換方式 370

15.8 AD コンバータ使用時の注意.....371

第 16 章 モータ制御回路(PMD : Programmable Motor Driver)

16.1 PMD 回路構成.....375
16.2 PMD レジスタ一覧.....376
16.2.1 PMDxMDEN(PMD イネーブルレジスタ).....377
16.2.2 PMDxPORTMD(ポート出力モードレジスタ).....378
16.2.3 PMDxMODESEL (モード選択レジスタ).....379
16.2.4 パルス幅変調回路.....380
16.2.4.1 PMDxMDCR (PMD コントロールレジスタ)
16.2.4.2 PMDxCNTSTA (PWM カウンタステータスレジスタ)
16.2.4.3 PMDxMDCNT(PWM カウンタレジスタ)
16.2.4.4 PMDxMDPRD(PWM 周期レジスタ)
16.2.4.5 PMDxCMPU (U 相用 PWM コンペアレジスタ)
16.2.4.6 PMDxCMPV (V 相用 PWM コンペアレジスタ)
16.2.4.7 PMDxCMPW (W 相用 PWM コンペアレジスタ)
16.2.5 通電制御回路.....392
16.2.5.1 PMDxMDPOT (PMD 出力設定レジスタ)
16.2.5.2 PMDxMDOUT(PMD 通電制御レジスタ)
16.2.6 保護制御回路.....397
16.2.6.1 保護制御回路(EMG 入力部)
16.2.6.2 PMDxEMGREL (EMG 解除レジスタ)
16.2.6.3 PMDxEMGCR (EMG コントロールレジスタ)
16.2.6.4 PMDxEMGSTA (EMG ステータスレジスタ)
16.2.6.5 保護制御回路(OVV 入力部)
16.2.6.6 PMDxOVVCR (OVV コントロールレジスタ)
16.2.6.7 PMDxOVVSTA (OVV ステータスレジスタ)
16.2.7 デッドタイム制御回路.....406
16.2.7.1 PMDxDTR (デッドタイムレジスタ)
16.2.8 同期トリガ生成回路.....409
16.2.8.1 PMDxTRGCMP0 (トリガコンペアレジスタ 0)
16.2.8.2 PMDxTRGCMP1 (トリガコンペアレジスタ 1)
16.2.8.3 PMDxTRGCMP2 (トリガコンペアレジスタ 2)
16.2.8.4 PMDxTRGCMP3 (トリガコンペアレジスタ 3)
16.2.8.5 PMDxTRGCR (トリガコントロールレジスタ)
16.2.8.6 PMDxTRGSYNCR (トリガ更新タイミング設定レジスタ)
16.2.8.7 PMDxTRGMD (トリガ出力モード設定レジスタ)
16.2.8.8 PMDxTRGSEL (トリガ出力選択レジスタ)

第 17 章 ベクトルエンジン(A-VE)

17.1 概要.....421
17.1.1 特徴.....421
17.1.2 主な機能.....422
17.2 構成.....422
17.2.1 ベクトルエンジンとモータ制御回路および A/D 変換器の関連.....423
17.3 レジスタ説明.....425
17.3.1 レジスタ一覧.....425
17.3.2 VE 制御レジスタ.....428
17.3.2.1 VExEN(ベクトルエンジン動作許可／禁止レジスタ)
17.3.2.2 VExCPURUNTRG(CPU 起動トリガ選択レジスタ)
17.3.2.3 VExTASKAPP(タスク指定レジスタ)
17.3.2.4 VExACTSCH(動作スケジュール選択レジスタ)
17.3.2.5 VExREPTIME(動作スケジュール繰り返し回数指定レジスタ)
17.3.2.6 VExTRGMODE(起動トリガモード設定レジスタ)
17.3.2.7 VExERRINTEN(エラー割り込み許可／禁止設定レジスタ)
17.3.2.8 VExCOMPEND(VE 強制終了レジスタ)
17.3.2.9 VExERRDET(エラー検出レジスタ)
17.3.2.10 VExSCHTASKRUN(スケジュール動作状態／実行中タスク番号レジスタ)
17.3.2.11 VExTMPREG0 ~ 5(テンポラリレジスタ)
17.3.3 専用レジスタ.....441
17.3.3.1 VExMODE(タスク制御モードレジスタ)
17.3.3.2 VExFMODE(フロー制御レジスタ)
17.3.3.3 VExTPWM(PWM 周期レート設定レジスタ)

17.3.3.4 VExOMEGA(回転速度設定レジスタ)
17.3.3.5 VExTHETA(モータ位相設定レジスタ)
17.3.3.6 VExCOS/VExSIN/VExCOSM/VExSINM(SIN/COS レジスタ)
17.3.3.7 VExIDREF/VExIQREF(d 軸/q 軸基準電流値設定レジスタ)
17.3.3.8 VExVD/VExVQ(d 軸/q 軸電圧設定レジスタ)
17.3.3.9 VExCIDKI/VExCIDKP/VExVCIQKI/VExCIQKP(PI 制御係数レジスタ)
17.3.3.10 VExVDIH/VExVDILH/VExVQIH/VExVQILH(PI 制御積分項保持レジスタ)
17.3.3.11 VExMCTLF(異常/判定結果保持レジスタ)
17.3.3.12 VExFPWMCHG(PWM 切り替え速度設定レジスタ)
17.3.3.13 VExMDPRD(PWM 周期設定レジスタ)
17.3.3.14 VExMINPLS(最小パルス幅差設定レジスタ)
17.3.3.15 VExSECTOR/VExSECTORM(セクタ情報レジスタ)
17.3.3.16 VExIAO/VExIBO/VExICO(ゼロ電流レジスタ)
17.3.3.17 VExIAADC/VExIBADC/VExICADC(電流 ADC 結果レジスタ)
17.3.3.18 VExVDC/VExVDCL(電源電圧レジスタ)
17.3.3.19 VExID/VExIQ(d 軸/q 軸電流レジスタ)
17.3.3.20 VExTADC(ADC 変換時間設定レジスタ)
17.3.3.21 VExCMPU/ VExCMPV/ VExCMPW(PWM DUTY レジスタ)
17.3.3.22 VExOUTCR(6 相出力制御レジスタ)
17.3.3.23 VExTRGCRC(同期トリガ補正量設定レジスタ)
17.3.3.24 VExTRGCMP0/VExTRGCMP1(トリガタイミング設定レジスタ)
17.3.3.25 VExTRGSEL(同期トリガ指定レジスタ)
17.3.3.26 VExEMGRS(EMG 復帰設定レジスタ)
17.3.3.27 VExPIOLIM(PI 制御出力制限レジスタ)
17.3.3.28 VExCIDKG/VExCIQKG(PI 制御 d 軸/q 軸係数レンジ設定レジスタ)
17.3.3.29 VExVSLIM(電圧スカラ制限レジスタ)
17.3.3.30 VExVDQ(電圧スカラレジスタ)
17.3.3.31 VExDELTA(偏角レジスタ)
17.3.3.32 VExCPHI/VExCLD/VExCLQ/VExCR/VExCPHIG/VExCLG/VExCRG(モータ定数レジスタ)
17.3.3.33 VExVDE/VExVQE(非干渉制御 d 軸/q 軸電圧レジスタ)
17.3.3.34 VExDTC(デッドタイム補償レジスタ)
17.3.3.35 VExHYS(電流判定ヒステリシスレジスタ)
17.3.3.36 VExDTCS(デッドタイム補償制御/ステータスレジスタ)
17.3.3.37 VExPWMMAX/VExPWMMIN(PWM 出力制限レジスタ)
17.3.3.38 VExTHTCLP(位相クリップレジスタ)
17.4 動作説明....491
17.4.1 スケジュール管理....491
17.4.1.1 スケジュール制御
17.4.1.2 起動制御
17.4.1.3 割り込み制御
17.4.2 タスク概要....497
17.4.2.1 電流制御(タスク 5)
17.4.2.2 SIN/COS 演算(タスク 6)
17.4.2.3 出力電圧変換(座標軸変換/相変換)
17.4.2.4 出力制御
17.4.2.5 トリガ生成(タスク 1)
17.4.2.6 入力処理
17.4.2.7 入力電流変換(相変換/座標軸変換)
17.4.2.8 その他タスク
17.5 VE チャネルと PMD および ADC の組み合わせ....529

第 18 章 エンコーダ入力回路 (A-ENC)

18.1 概要....531
18.2 ブロック図....531
18.3 レジスタ....532
18.3.1 レジスタ一覧....532
18.3.2 ENxTNCR(ENC 制御レジスタ)....533
18.3.3 ENxRELOAD(RELOAD コンペアレジスタ)....536
18.3.4 ENxINT(INT コンペアレジスタ)....537
18.3.5 ENxCNT(カウンタレジスタ)....538
18.3.6 ENxMCMP(MCMP カウンタレジスタ)....539
18.3.7 ENxRATE(位相カウントレートレジスタ)....540
18.3.8 ENxSTS(ステータスレジスタ)....541
18.3.9 ENxINPCR(入力処理制御レジスタ)....542
18.3.10 ENxSMPDLY(サンプルディレイレジスタ)....543
18.3.11 ENxINPMON(入力モニタレジスタ)....544
18.3.12 ENxCLKCR(サンプルクロック制御レジスタ)....545

18.3.13 ENxINTCR(割り込み制御レジスタ)....546
18.3.14 ENxINTF(割り込みフラグレジスタ)....547
18.4 動作説明....548
18.4.1 エンコーダモード....548
18.4.2 センサモード....549
18.4.2.1 イベントカウント
18.4.2.2 タイマカウント
18.4.2.3 位相カウント
18.4.3 タイマモード....554
18.4.4 位相カウンタモード....555
18.4.4.1 位相差カウンタモード
18.5 回路別の機能概要....558
18.5.1 入力回路....558
18.5.1.1 サンプルクロック
18.5.1.2 サンプリングモード
18.5.1.3 ノイズキャンセル
18.5.2 デコーダ....561
18.5.2.1 回転エッジ検出と方向信号生成
18.5.2.2 Z 判定回路
18.5.2.3 スキップ判定と入力異常判定
18.5.2.4 エッジ検出エラー判定
18.5.2.5 バッファ更新制御
18.5.2.6 BEMF 検出制御
18.5.3 カウンタ....564
18.5.3.1 エンコーダモード、センサモード(イベントカウント)
18.5.3.2 センサモード(タイマカウント)、タイマモード
18.5.3.3 センサモード(位相カウント)、位相カウンタモード
18.5.4 割り込み要求制御....567
18.6 ブラシレス DC モータ制御例....568

第 19 章 パワーオンリセット回路(POR)

19.1 構成....569
19.2 機能....570

第 20 章 電圧検出回路(VLTD)

20.1 構成....571
20.2 レジスタ説明....572
20.2.1 レジスター覧....572
20.2.2 VDCR (電圧検出制御レジスタ)....572
20.3 動作説明....573
20.3.1 制御....573
20.3.2 機能....573
20.3.2.1 電圧検出動作の許可／禁止
20.3.2.2 検出電圧レベル選択

第 21 章 周波数検知回路(OFD)

21.1 構成....575
21.2 レジスタ説明....576
21.2.1 レジスター覧....576
21.2.1.1 OFDCR1(周波数検知回路制御レジスタ 1)
21.2.1.2 OFDCR2(周波数検知回路制御レジスタ 2)
21.2.1.3 OFDMNPLLOFF (検知周波数下限値設定レジスタ(PLL OFF 時))
21.2.1.4 OFDMNPLLON (検知周波数下限値設定レジスタ(PLL ON 時))
21.2.1.5 OFDMXPLLOFF (検知周波数上限値設定レジスタ(PLL OFF 時))
21.2.1.6 OFDMXPLLON (検知周波数上限値設定レジスタ(PLL ON 時))
21.3 動作説明....583

21.3.1 設定....583
21.3.2 使用可能な動作モード....583
21.3.3 動作....584
21.3.4 検知周波数....585
21.3.5 動作手順例....586

第 22 章 ウォッチドッグタイマ(WDT)

22.1 構成....587
22.2 レジスタ....588
22.2.1 レジスター覧....588
22.2.1.1 WDMOD(ウォッチドッグタイマモードレジスタ)
22.2.1.2 WDCR(ウォッチドッグタイマコントロールレジスタ)
22.3 動作説明....590
22.3.1 基本動作....590
22.3.2 動作モードと動作状態....590
22.3.3 暴走検出時の動作....590
22.3.3.1 INTWDT 割り込み発生の場合
22.3.3.2 内部リセット発生の場合
22.4 ウォッチドックタイマの制御....591
22.4.1 ディセーブル制御....591
22.4.2 イネーブル制御....591
22.4.3 ウォッチドッグタイマのクリア制御....591
22.4.4 ウォッチドッグタイマ検出時間の設定....591

第 23 章 フラッシュメモリ(FLASH)

23.1 フラッシュメモリの特長....593
23.1.1 メモリ容量と構成....593
23.1.2 機能....594
23.1.3 動作モード....594
23.1.3.1 モードの説明
23.1.3.2 モードの決定
23.1.4 メモリマップ....597
23.1.5 プロテクト/セキュリティ機能....599
23.1.5.1 プロテクト機能
23.1.5.2 セキュリティ機能
23.1.6 メモリスワップ機能....600
23.1.6.1 概要
23.1.6.2 動作説明
23.1.6.3 操作方法
23.1.7 レジスタ....602
23.1.7.1 レジスター覧
23.1.7.2 FCSECBIT(セキュリティビットレジスタ)
23.1.7.3 FCPSR0(プロテクトステータスレジスタ 0)
23.1.7.4 FCSR(ステータスレジスタ)
23.1.7.5 FCSWPSR(スワップステータスレジスタ)
23.1.7.6 FCAREASEL(エリア選択レジスタ)
23.1.7.7 FCCR(コントロールレジスタ)
23.1.7.8 FCSTSCLR(ステータスクリアレジスタ)
23.1.7.9 FCWCLKCR(WCLK 設定レジスタ)
23.1.7.10 FCPROGCR(Program 用カウンタ設定レジスタ)
23.1.7.11 FCERASECR(Erase 用カウンタ設定レジスタ)
23.2 フラッシュメモリ詳細....614
23.2.1 機能....614
23.2.2 フラッシュメモリの動作モード....614
23.2.3 コマンド実行方法....614
23.2.4 自動動作の中止....615
23.2.5 自動動作の完了検知....616
23.2.5.1 手順
23.2.6 コマンド説明....616
23.2.6.1 自動プログラム
23.2.6.2 自動チップ消去

23.2.6.3 自動エリア消去
23.2.6.4 自動ブロック消去
23.2.6.5 自動ページ消去
23.2.6.6 自動プロテクトビットプログラム
23.2.6.7 自動プロテクトビット消去
23.2.6.8 ID-Read
23.2.6.9 Read/リセットコマンド (ソフトウエアリセット)
23.2.6.10 自動メモリスワップ
23.2.7 コマンドシーケンス....621
23.2.7.1 コマンドシーケンス一覧
23.2.7.2 バスライトサイクル時のアドレスビット構成
23.2.7.3 エリアアドレス(AA)、ブロックアドレス(BA)
23.2.7.4 プロテクトビットの指定(PBA)
23.2.7.5 ID-Read のコード(IA, ID)
23.2.7.6 メモリスワップビットの指定(MSA)
23.2.7.7 コマンドシーケンス例
23.2.8 フローチャート....628
23.2.8.1 自動プログラム
23.2.8.2 自動消去
23.3 シングルブートモードによる書き替え方法....630
23.3.1 モード設定....630
23.3.2 インタフェース仕様....630
23.3.3 メモリの制約について....631
23.3.4 動作コマンド....631
23.3.4.1 RAM 転送
23.3.4.2 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
23.3.5 コマンドによらず共通の動作....632
23.3.5.1 シリアル動作モード判定
23.3.5.2 ACK 応答データ
23.3.5.3 パスワード判定
23.3.5.4 CHECK SUM の計算方法
23.3.6 シリアル動作モード判定の通信ルール....637
23.3.7 RAM 転送コマンドの通信ルール....638
23.3.8 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通信ルール....640
23.3.9 内蔵ブートプログラム全体フローチャート....641
23.3.10 内蔵 BOOT ROM の書き替えアルゴリズムを利用した書き替え手順....642
23.3.10.1 Step-1
23.3.10.2 Step-2
23.3.10.3 Step-3
23.3.10.4 Step-4
23.3.10.5 Step-5
23.3.10.6 Step-6
23.4 ユーザブートモードによる書き替え方法....645
23.4.1 (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....645
23.4.1.1 Step-1
23.4.1.2 Step-2
23.4.1.3 Step-3
23.4.1.4 Step-4
23.4.1.5 Step-5
23.4.1.6 Step-6
23.4.2 (1-B)書き替えルーチンを外部から転送する手順例....649
23.4.2.1 Step-1
23.4.2.2 Step-2
23.4.2.3 Step-3
23.4.2.4 Step-4
23.4.2.5 Step-5
23.4.2.6 Step-6
23.5 ユーザブートプログラムの書き替え方法....653
23.5.1 フラッシュ書き替えの手順例....653
23.5.1.1 Step-1
23.5.1.2 Step-2
23.5.1.3 Step-3
23.5.1.4 Step-4
23.5.1.5 Step-5
23.5.1.6 Step-6
23.5.1.7 Step-7
23.5.1.8 Step-8
23.5.1.9 Step-9

第 24 章　デバッグインタフェース

24.1　仕様概要.....657
24.2　SWJ-DP.....657
24.3　ETM.....657
24.4　ホールトモード中の周辺機能.....657
24.5　デバッグツールとの接続.....658
24.5.1　接続方法.....658
24.5.2　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.....658

第 25 章　ポート等価回路図

25.1　ポート.....659
25.2　アナログ端子.....660
25.3　制御端子.....660
25.4　クロック端子.....661

第 26 章　電気的特性

26.1　絶対最大定格.....663
26.2　DC 電気的特性 (1/2).....664
26.3　DC 電気的特性 (2/2).....665
26.4　12 ビット A/D コンバータ変換特性.....666
26.5　AC 電気的特性.....667
26.5.1　AC 測定条件.....667
26.5.2　シリアルチャネル (SIO/UART).....667
26.5.2.1　I/O インタフェースモード (VDD = 4.5 ~ 5.5V)
26.5.3　シリアルバスインタフェース (I2C).....669
26.5.3.1　I2C モード
26.5.4　16 ビットタイマ／カウンタ(TMRB).....671
26.5.4.1　イベントカウンタ
26.5.4.2　キャプチャ
26.5.5　外部割り込み.....671
26.5.6　デバッグ通信.....672
26.5.6.1　AC 測定条件
26.5.6.2　SWD インタフェース
26.5.6.3　JTAG インタフェース
26.5.7　ETM トレース.....673
26.5.8　Flash 特性.....674
26.5.9　内蔵発振回路特性.....674
26.5.10　外部発振子.....674
26.6　発振回路.....675
26.6.1　セラミック発振子.....675

第 27 章　パッケージ寸法図

CMOS 32 ビット マイクロコントローラ

TMPM470FDFG/FZFG/FYFG

TMPM470FDFG/FZFG/FYFG は、ARM®社 Cortex®-M4F コアを内蔵した 32 ビット RISC マイクロプロセッサです。

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®-M4F コア使用
   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 × 32 = 32 ビット)、積和演算(32 + 32 × 32 = 32 ビット)を 1 クロックで実行
      ・SIMD(Single Instruction Multiple Data)演算を 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い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 をハードウエアで自動的に実行

2. 単精度浮動小数点演算(FPU)
   ・ IEEE754 標準に準拠
   ・ 加算/減算/乗算は 1 クロックで実行、積和は 3 クロックで実行
   ・ CPU とは別に専用データレジスタによる並列処理が可能

3. 内蔵プログラムメモリ/データメモリ
   ・ 内蔵 RAM :
      TMPM470FDFG : 34Kbyte
      TMPM470FZFG : 34Kbyte
      TMPM470FYFG : 18Kbyte
   ・ 内蔵 FlashROM :
      TMPM470FDFG : 512Kbyte
      TMPM470FZFG : 384Kbyte
      TMPM470FYFG : 256Kbyte

4. μDMA コントローラ(μDMAC) : 32 チャネル/ 1 ユニット
   転送対象 : 内蔵メモリ、周辺機能

5. クロック制御(CG)
   - PLL 内蔵(8, 10, 12 逓倍)
   - クロックギア機能 : 高速クロックを 1/1, 1/2, 1/4, 1/8,1/16 に分周可能

6. 低消費電力機能
   IDLE, STOP

7. 入出力ポート(PORT) : 入出力端子 : 79 本

8. 16 ビットタイマ(TMRB) : 10 チャネル
   - 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
   - インプットキャプチャ機能
   - 16 ビット PPG 出力
   - 外部トリガ PPG 出力

9. ウォッチドッグタイマ(WDT) : 1 チャネル
   リセットまたはマスク不能割り込み(NMI)発生

10. シリアルチャネル(SIO/UART) : 4 チャネル
    - UART/クロック同期式モード選択可能
    - 送信 FIFO : 8 ビット幅 4 段、受信 FIFO : 8 ビット幅 4 段

11. シリアルバスインタフェース(I2C/SIO) : 1 チャネル
    通信速度 100kbps / 400kbps

12. パワーオンリセット回路(POR) : 1 ユニット

13. 電圧検出回路(VLTD) : 1 ユニット

14. 周波数検知回路(OFD) : 1 ユニット

15. ベクトルエンジン(A-VE) : 2 チャネル
    - モータ制御用演算機能
    - 2 モータ対応

16. モータ制御回路(PMD) : 2 チャネル
    - 3 相相補 PWM 出力
    - AD コンバータを連動させる同期トリガ生成
    - 緊急停止保護機能(EMG 端子)

17. エンコーダ入力回路(A-ENC) : 2 チャネル

- 位置検出機能

  インクリメンタル形エンコーダ対応(AB,ABZ 入力)

  ホール IC 対応(1 ~ 3 相入力)

  120°通電の 3 相 BLDC モータの誘起電圧ゼロクロス検出対応
- デジタルノイズフィルタ内蔵
- モータ制御回路(PMD)の PWM 信号に同期した入力サンプリング可能
- タイマカウンタ機能

  エンコーダ,ホール IC での位置検出による 16 ビットアップダウンカウンタ

  任意周期でカウント可能な 16 ビットカウンタ／キャプチャ

  32 ビットタイマカウンタ／キャプチャ

18. 12 ビット AD コンバータ(ADC) : 2 ユニット

  ユニット A アナログ入力 : 12 チャネル

  ユニット B アナログ入力 : 11 チャネル

  ユニット A 外部アナログ入力端子 : 9 本

  A/B 両ユニットへの外部アナログ入力端子 : 3 本

  ユニット B 外部アナログ入力端子 : 8 本
- トリガスタート機能 : TMRB 割り込み/ PMD トリガによるスタート可能
- 常時変換可能
- AD 監視機能 2ch
- 変換時間 1.0 μsec (ADC 変換クロック 120 MHz 時)

19. 割り込み機能 : 優先順位を 7 レベルで設定可能
- 内部 77 本 (ウォッチドッグタイマ割り込みを除く)
- 外部 16 本

20. デジタルノイズフィルタ(DNF) : 外部割り込み 16 本

21. エンディアン

  リトルエンディアン

22. 内蔵高速発振回路:10MHz

23. 内蔵高速発振調整機能(TRMOSC):1 ユニット

24. 最大動作周波数 : 120 MHz

25. 動作電圧範囲
- DVDD5 = 4.5 V~5.5 V @ fsys = 120 MHz

  全機能動作
- DVDD5 = 3.9 V~4.5 V @ fsys = 120 MHz

  除く、12 ビット AD コンバータと AC 電気的特性, Flash 書き込み

26. 温度範囲

−40°C ~ 85°C

27. パッケージ

LQFP100 (14mm×14mm, 0.5mm ピッチ)

1.2 ブロック図

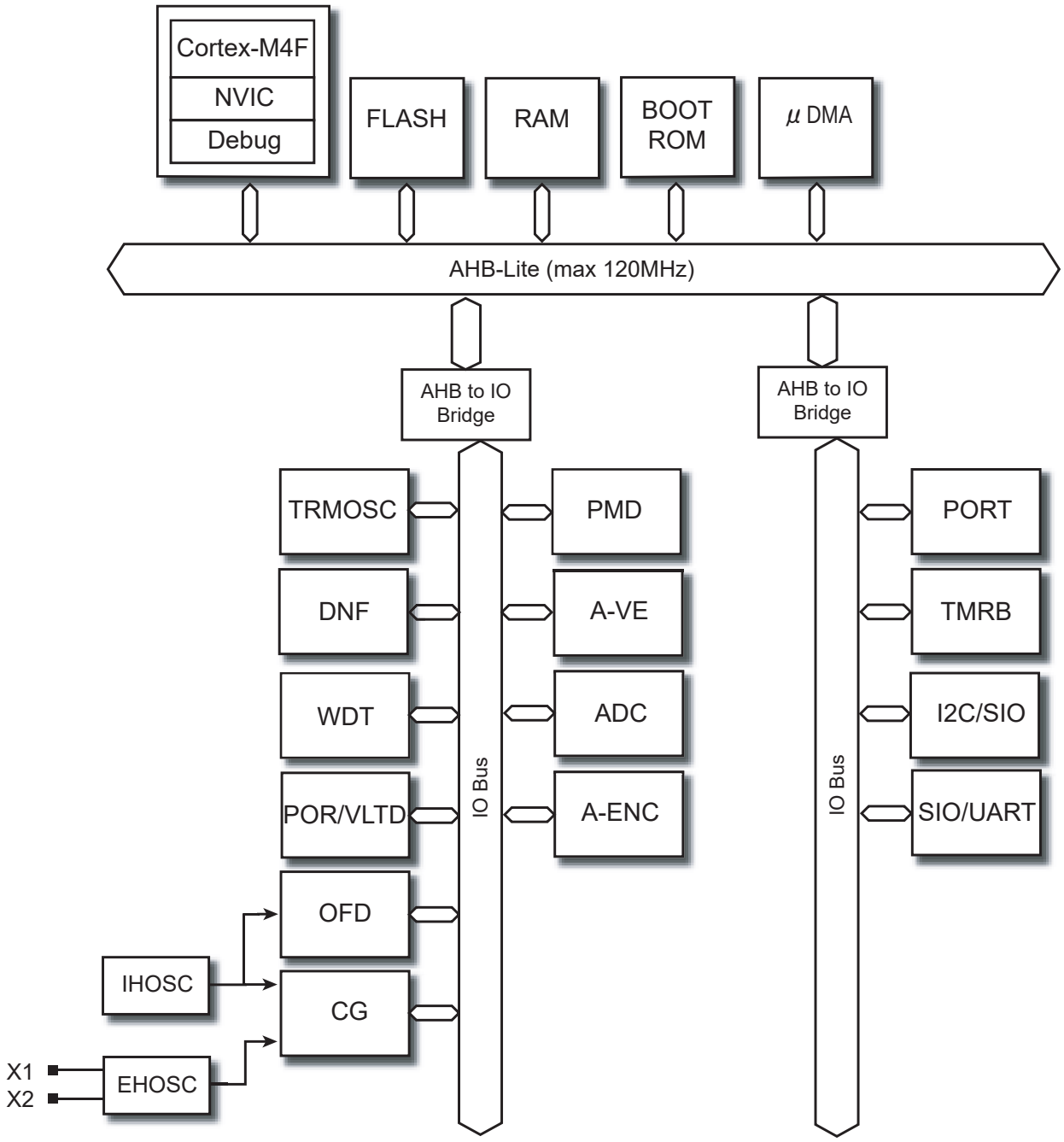


図 1-1 TMPM470FDFG/FZFG/FYFG ブロック図

1.3 ピン配置図(Top view)

TMPM470FDFG/FZFG/FYFG のピン配置図は、下図のとおりです。

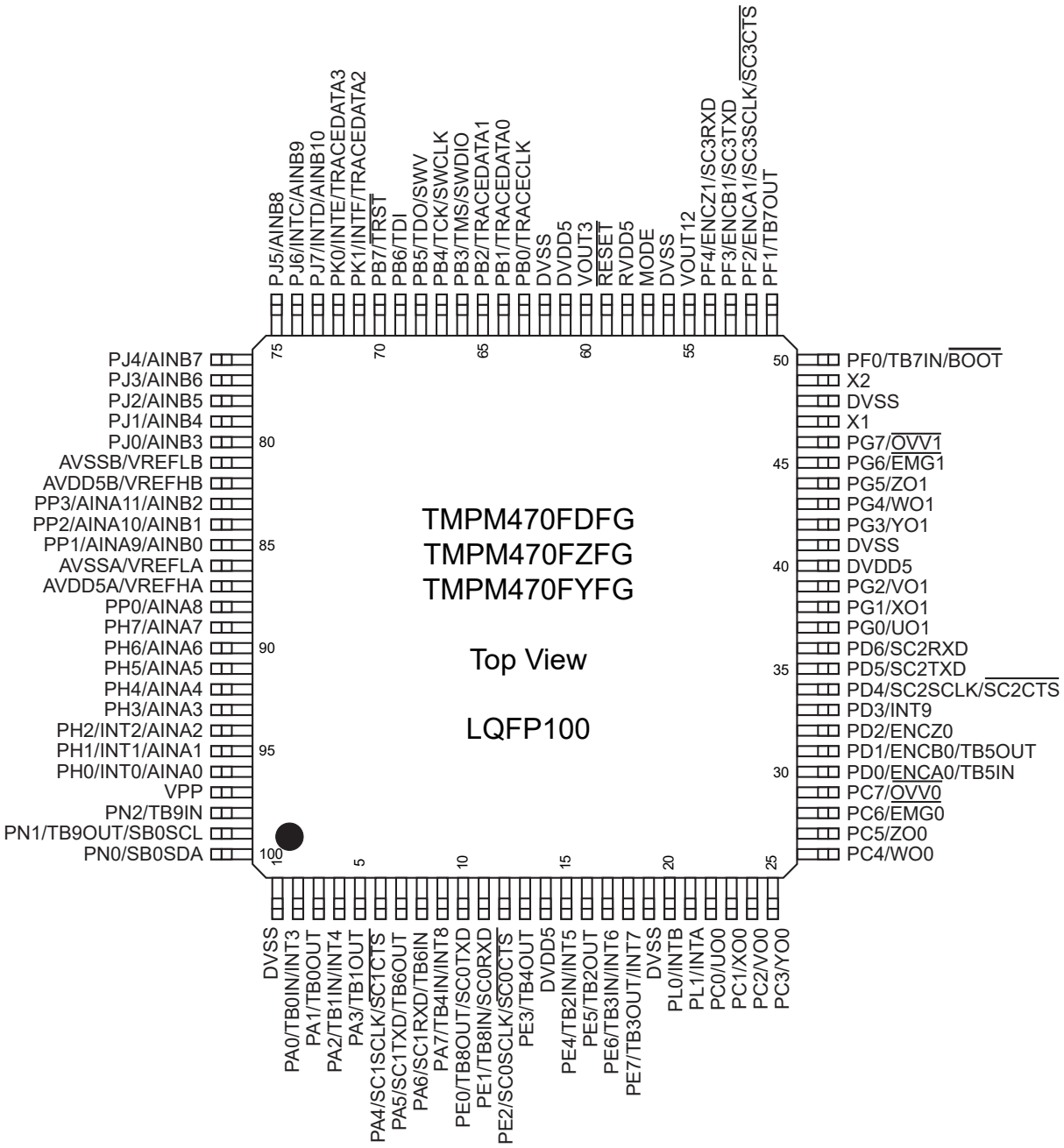


図 1-2 ピン配置図(LQFP100)

1.4 ピン名称と機能

1.4.1 機能端子名称と機能

1.4.1.1 周辺機能端子名称

表 1-1 各周辺機能の端子名称と機能

周辺機能	端子名称	Input or Output	機能
外部割り込み	INTx	Input	外部割り込み入力端子 x 外部割込み入力端子 x はノイズフィルタ(フィルタ幅 typ. 30ns)をもちます。
16 ビットタイマ/イベントカウンタ	TBxIN	Input	インプットキャプチャ入力端子
	TBxOUT	Output	出力端子
SIO/UART	SCxTXD	Output	データ出力端子
	SCxRXD	Input	データ入力端子
	SCxSCLK	I/O	クロック入出力端子
	$\overline{\text{SCxCTS}}$	Input	ハンドシェイク入力端子
I2C	SBxSDA	I/O	データ入出力端子
	SBxSCL	I/O	クロック入出力端子
アナログデジタルコンバータ	AINAx	Input	アナログ入力端子
	AINBx	Input	アナログ入力端子
エンコーダ	ENCAx	Input	エンコーダ入力端子
	ENCBx	Input	エンコーダ入力端子
	ENCZx	Input	エンコーダ入力端子
PMD	XOx	Output	X 相出力端子
	YOx	Output	Y 相出力端子
	ZOx	Output	Z 相出力端子
	UOx	Output	U 相出力端子
	VOx	Output	V 相出力端子
	WOx	Output	W 相出力端子
	$\overline{\text{EMGx}}$	Input	異常検出入力端子
	$\overline{\text{OVVx}}$	Input	過電圧入力端子

1.4.1.2 デバッグ端子名称

表 1-2 デバッグ端子名称と機能

デバッグ端子名称	Input or Output	機能
TMS	Input	JTAG テストモード選択入力端子
TCK	Input	JTAG シリアルクロック入力端子
TDO	Output	JTAG シリアルデータ出力端子
TDI	Input	JTAG シリアルデータ入力端子
$\overline{\text{TRST}}$	Input	JTAG テストリセット入力端子
SWDIO	I/O	シリアルワイヤデータ入出力端子
SWCLK	Input	シリアルワイヤクロック入力端子
SWV	Output	シリアルワイヤビューワ出力端子
TRACECLK	Output	トレースクロック出力端子
TRACEDATA0	Output	トレースデータ出力端子 0
TRACEDATA1	Output	トレースデータ出力端子 1
TRACEDATA2	Output	トレースデータ出力端子 2
TRACEDATA3	Output	トレースデータ出力端子 3

1.4.1.3 制御端子名称

表 1-3 制御端子名称と機能

制御端子名称	Input or Output	機能
X1	Input	高速発振子接続端子
X2	Output	高速発振子接続端子
MODE	Input	モード端子 必ず"Low"レベルに固定してください。
$\overline{\text{RESET}}$	Input	リセット信号入力端子
$\overline{\text{BOOT}}$	Input	BOOT モード制御用端子 リセット端子入力の立ち上がりで BOOT モード制御用端子がサンプリングされます。 BOOT モード制御用端子のレベルが"Low"の場合、シングルブートモードになります。"High"の場合、シングルチップモードになります。 詳細については、"フラッシュメモリ"章を参照してください。

1.4.1.4 電源端子名称

表 1-4 電源端子名称と機能

電源 端子名称	機能
VOUT12	レギュレータ用コンデンサ (3.3 to 4.7 μF) 接続端子
VOUT3	レギュレータ用コンデンサ (3.3 to 4.7 μF) 接続端子
VPP	(オープンで使用してください。)
RVDD5	レギュレータ用電源端子
DVDD5	デジタル用電源端子 DVDD5 は下記の端子に電源を供給しています。 PA, PB, PC, PD, PE, PF, PG, PL, PK, PN, X1, X2, MODE, $\overline{\text{RESET}}$
DVSS	デジタル用 GND 端子
AVDD5A AVDD5B	ADC 用電源端子 (注 1) AVDD5A は PH, PP0 の端子に電源を供給しています。 AVDD5B は PJ, PP1 to 3 の端子に電源を供給しています。
AVSSA AVSSB	ADC 用 GND 端子 (注 2)
VREFHA VREFHB	ADC 用基準電源端子
VREFLA VREFLB	ADC 用基準電源端子

注 1) 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。
注 2) 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。

1.4.1.5 電源間コンデンサ

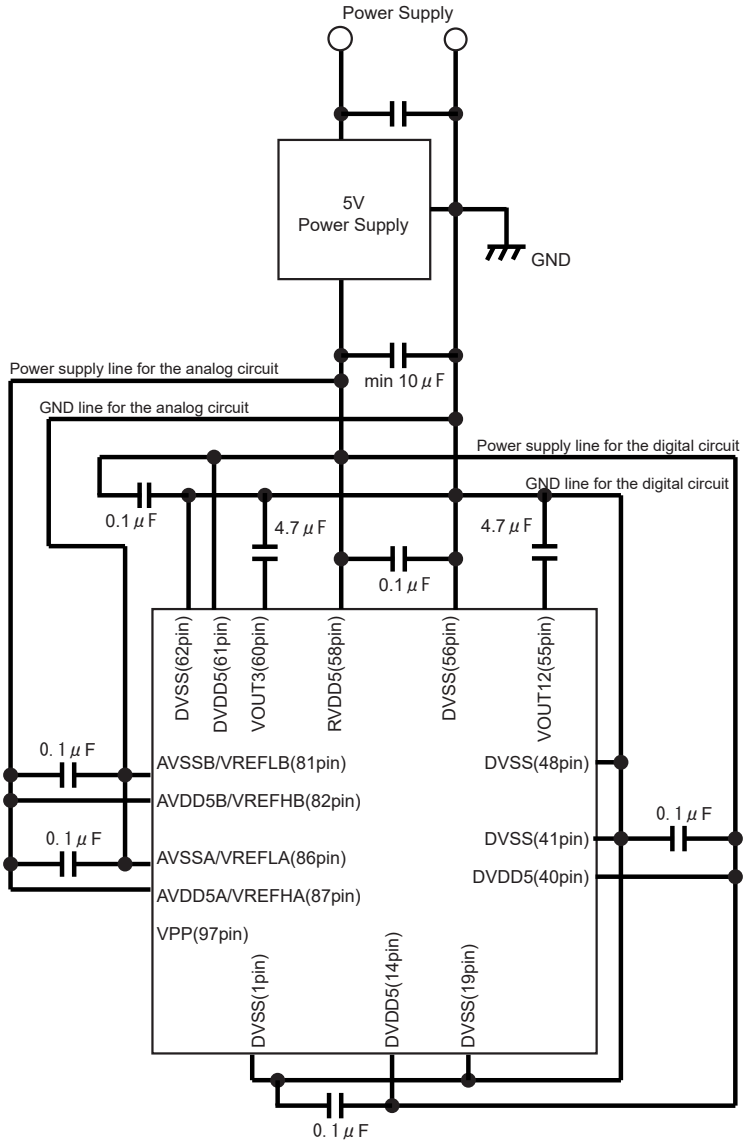


図 1-3 電源間コンデンサの接続図

注 1) 58 ピンのRVDD5, 56 ピンのDVSS からできるだけ短い距離に電源の出力コンデンサ (10 μF 以上)を配置してください。

注 2) 58 ピンのRVDD5, 56 ピンのDVSS と電源の出力コンデンサ (10 μF 以上)の間において、このコンデンサの近傍で、5V 電源ラインをアナログ電源ラインとデジタル電源ラインに分離してください。分離までの距離が長くなると、共通インピーダンスのためデジタル電源変動がアナログ電源に伝わり、アナログ回路のノイズになります。

注 3) 電源ラインと GND ラインを近づけて配線してください。離れていると電源回路のコンデンサを介して電源ラインと GND ラインで電源ループできてしまい高周波ノイズを受けるアンテナになります。

注 4) 内蔵電源 VOUT3 と VOUT12 のレギュレータ用のコンデンサは、同じ容量とし、56 ピンの DVSS 近傍で配置してください。

注 5) 14 ピンと 19 ピンの間、40 ピンと 41 ピンの間、58 ピンと 56 ピンの間、61 ピンと 62 ピンの間の各コンデンサを各端子の近傍で配置してください。

注 6) 82 ピンと 81 ピンの間、87 ピンと 86 ピンの間の各コンデンサを各端子の近傍で配置してください。

1.4.2 ピン名称と機能

1.4.2.1 表の見方

表中の記号の意味は下記の通りです。

1. 機能 A

   ファンクションレジスタ設定なしにポートに割り当てられる兼用機能が記載されています。

2. 機能 B

   ファンクションレジスタ設定によりポートに割り当てられる兼用機能が記載されています。機能 B の番号はファンクションレジスタの番号と対応しています。

3. 端子仕様

   記号の意味は下記のとおりです。

- SMT/CMOS : 入力ゲート

  SMT : Schmitt 入力

  CMOS : CMOS 入力

- 5V_T : 5V トレラント対応

  Yes : 対応

  N/A : 非対応

- OD : プログラマブル Open Drain 出力対応

  Yes : 対応

  N/A : 非対応

- PU/PD:プログラマブル Pull-Up/Pull-Down 対応

  PU : プログラマブル Pull-Up 選択可能

  PD : プログラマブル Pull-Down 選択可能

1.4.2.2 PORT / デバッグ端子

表 1-5 ピン番号と端子名称<PORT 順>

Pin No.	PORT	機能 A	機能 B					ポート仕様		
			1	2	3	4	5	PU/PD	OD	SMT/ CMOS
PORT A										
2	PA0	INT3	TB0IN					PU/PD	Yes	SMT
3	PA1		TB0OUT					PU/PD	Yes	SMT
4	PA2	INT4	TB1IN					PU/PD	Yes	SMT
5	PA3		TB1OUT					PU/PD	Yes	SMT
6	PA4		SC1SCLK	$\overline{\text{SC1CTST}}$				PU/PD	Yes	SMT
7	PA5		SC1TXD	TB6OUT				PU/PD	Yes	SMT
8	PA6		SC1RXD	TB6IN				PU/PD	Yes	SMT
9	PA7	INT8	TB4IN					PU/PD	Yes	SMT
PORTB										
63	PB0		TRACECLK					PU/PD	Yes	SMT
64	PB1		TRACEDATA0					PU/PD	Yes	SMT
65	PB2		TRACEDATA1					PU/PD	Yes	SMT
66	PB3		TMS/SWDIO					PU/PD	Yes	SMT
67	PB4		TCKSWCLK					PU/PD	Yes	SMT
68	PB5		TDO/SWV					PU/PD	Yes	SMT
69	PB6		TDI					PU/PD	Yes	SMT
70	PB7		$\overline{\text{TRST}}$					PU/PD	Yes	SMT
PORTC										
22	PC0		UO0					PU/PD	Yes	SMT
23	PC1		XO0					PU/PD	Yes	SMT
24	PC2		VO0					PU/PD	Yes	SMT
25	PC3		YO0					PU/PD	Yes	SMT
26	PC4		WO0					PU/PD	Yes	SMT
27	PC5		ZO0					PU/PD	Yes	SMT
28	PC6		$\overline{\text{EMG0}}$					PU/PD	Yes	SMT
29	PC7		$\overline{\text{OVV0}}$					PU/PD	Yes	SMT
PORTD										
30	PD0		ENCA0	TB5IN				PU/PD	Yes	SMT
31	PD1		ENCB0	TB5OUT				PU/PD	Yes	SMT
32	PD2		ENCZ0					PU/PD	Yes	SMT
33	PD3	INT9						PU/PD	Yes	SMT
34	PD4		SC2SCLK	$\overline{\text{SC2CTS}}$				PU/PD	Yes	SMT
35	PD5		SC2TXD					PU/PD	Yes	SMT
36	PD6		SC2RXD					PU/PD	Yes	SMT
PORTE										
10	PE0		SC0TXD	TB8OUT				PU/PD	Yes	SMT
11	PE1		SC0RXD	TB8IN				PU/PD	Yes	SMT
12	PE2		SC0SCLK	$\overline{\text{SC0CTS}}$				PU/PD	Yes	SMT
13	PE3		TB4OUT					PU/PD	Yes	SMT
15	PE4	INT5	TB2IN					PU/PD	Yes	SMT
16	PE5		TB2OUT					PU/PD	Yes	SMT
17	PE6	INT6	TB3IN					PU/PD	Yes	SMT

表 1-5 ピン番号と端子名称<PORT 順>

Pin No.	PORT	機能 A	機能 B					ポート仕様		
			1	2	3	4	5	PU/PD	OD	SMT/ CMOS
18	PE7	INT7	TB3OUT					PU/PD	Yes	SMT
PORTF										
50	PF0	$\overline{\text{BOOT}}$	TB7IN					PU/PD	Yes	SMT
51	PF1		TB7OUT					PU/PD	Yes	SMT
52	PF2		ENCA1	SC3SCLK	$\overline{\text{SC3CTS}}$			PU/PD	Yes	SMT
53	PF3		ENCB1	SC3TXD				PU/PD	Yes	SMT
54	PF4		ENCZ1	SC3RXD				PU/PD	Yes	SMT
PORTG										
37	PG0		UO1					PU/PD	Yes	SMT
38	PG1		XO1					PU/PD	Yes	SMT
39	PG2		VO1					PU/PD	Yes	SMT
42	PG3		YO1					PU/PD	Yes	SMT
43	PG4		WO1					PU/PD	Yes	SMT
44	PG5		ZO1					PU/PD	Yes	SMT
45	PG6		$\overline{\text{EMG1}}$					PU/PD	Yes	SMT
46	PG7		$\overline{\text{OVV1}}$					PU/PD	Yes	SMT
PORTH										
96	PH0	AINA0 INT0						PU/PD	Yes	SMT
95	PH1	AINA1 INT1						PU/PD	Yes	SMT
94	PH2	AINA2 INT2						PU/PD	Yes	SMT
93	PH3	AINA3						PU/PD	Yes	SMT
92	PH4	AINA4						PU/PD	Yes	SMT
91	PH5	AINA5						PU/PD	Yes	SMT
90	PH6	AINA6						PU/PD	Yes	SMT
89	PH7	AINA7						PU/PD	Yes	SMT
PORTJ										
80	PJ0	AINB3						PU/PD	Yes	SMT
79	PJ1	AINB4						PU/PD	Yes	SMT
78	PJ2	AINB5						PU/PD	Yes	SMT
77	PJ3	AINB6						PU/PD	Yes	SMT
76	PJ4	AINB7						PU/PD	Yes	SMT
75	PJ5	AINB8						PU/PD	Yes	SMT
74	PJ6	AINB9 INTC						PU/PD	Yes	SMT
73	PJ7	AINB10 INTD						PU/PD	Yes	SMT
PORTK										
72	PK0	INTE	TRACEDATA3					PU/PD	Yes	SMT
71	PK1	INTF	TRACEDATA2					PU/PD	Yes	SMT
PORTL										
20	PL0	INTB						PU/PD	Yes	SMT
21	PL1	INTA						PU/PD	Yes	SMT
PORTN										
100	PN0			SB0SDA				PU/PD	Yes	SMT

表 1-5 ピン番号と端子名称<PORT 順>

Pin No.	PORT	機能 A	機能 B					ポート仕様		
			1	2	3	4	5	PU/PD	OD	SMT/ CMOS
99	PN1		TB9OUT	SB0SCL				PU/PD	Yes	SMT
98	PN2		TB9IN					PU/PD	Yes	SMT
PORTP										
88	PP0	AINA8						PU/PD	Yes	SMT
85	PP1	AINA9 AINB0						PU/PD	Yes	SMT
84	PP2	AINA10 AINB1						PU/PD	Yes	SMT
83	PP3	AINA11 AINB2						PU/PD	Yes	SMT

1.4.2.3 制御端子

表 1-6 ピン番号と端子名称

Pin No.	制御 端子名称
47	X1
49	X2
57	MODE
59	$\overline{\text{RESET}}$
50	$\overline{\text{BOOT}}$

1.4.2.4 電源端子

表 1-7 ピン番号と端子名称

Pin No.	電源 端子名称
55	VOUT12
60	VOUT3
58	RVDD5
97	VPP
14, 40, 61	DVDD5
1, 19, 41, 48, 56, 62	DVSS
87	AVDD5A VREFHA
82	AVDD5B VREFHB
86	VREFLA AVSSA
81	AVREFLB AVSSB

第 2 章　製品情報

本章では、周辺機能に関し、チャネルまたはユニット数、端子情報、その他の製品固有機能の情報についてまとめます。周辺機能の章と合わせてご使用ください。

- 「2.1.1 DMA コントローラ(DMAC)」
- 「2.1.2 16 ビットタイマ/イベントカウンタ(TMRB)」
- 「2.1.3 シリアルチャネル(SIO/UART)」
- 「2.1.4 シリアルバスインタフェース(I2C)」
- 「2.1.5 ベクトルエンジン (A-VE)」
- 「2.1.6 モータ制御回路 (PMD)」
- 「2.1.7 エンコーダ (A-ENC)」
- 「2.1.8 アナログ/デジタルコンバータ(ADC)」
- 「2.1.9 ウォッチドッグタイマ(WDT)」
- 「2.1.10 デバッグインタフェース」

2.1　各周辺機能の情報

2.1.1　DMA コントローラ(DMAC)

TMPM470FDFG/FZFG/FYFG では DMA コントローラを 1 ユニット内蔵しています。

表 2-1 DMA 要求一覧

チャネル	バースト	シングル
0	SIO/UART0 受信	−
1	SIO/UART0 送信	−
2	SIO/UART1 受信	−
3	SIO/UART1 送信	−
4	SIO/UART2 受信	−
5	SIO/UART2 送信	−
6	SIO/UART3 受信	−
7	SIO/UART3 送信	−
8	タイマトリガ ADCA 変換終了	−
9	タイマトリガ ADCB 変換終了	−
10	ソフトスタート ADCA 終了	−
11	ソフトスタート ADCB 終了	−
12	PMD0 トリガ ADCA 終了	−
13	PMD0 トリガ ADCB 終了	−
14	PMD1 トリガ ADCA 終了	−
15	PMD1 トリガ ADCB 終了	−
16	PMD0 PWM 割り込み	−
17	PMD1 PWM 割り込み	−
18	−	−
19	−	−
20	TMRB0 インプットキャプチャ 0	−
21	TMRB1 インプットキャプチャ 0	−
22	TMRB2 インプットキャプチャ 0	−
23	TMRB3 インプットキャプチャ 0	−
24	VE 割り込み 0	−
25	VE 割り込み 1	−
26	TMRB0 コンペアー致(注)	−
27	TMRB1 コンペアー致(注)	−
28	TMRB2 コンペアー致(注)	−
29	TMRB3 コンペアー致(注)	−
30	VE タスク終了割り込み 0	−
31	VE タスク終了割り込み 1	−

注)　TMRB の DMA 転送要求は、TMRB 割り込みと同じ条件で発生します。TMRB 割り込みは、タイマレジスタ 0/1 との 一致およびオーバフローで発生しますので、必要に応じ不要な要因を割り込みマスクレジスタ TBxIM でマスクしてください。

2.1.2　16 ビットタイマ/イベントカウンタ(TMRB)

TMPM470FDFG/FZFG/FYFG では 10 チャネルの TMRB を内蔵しています。

表 2-2 端子仕様

チャネル	TBxOUT	TBxIN
TMRB0	PA1	PA0
TMRB1	PA3	PA2
TMRB2	PE5	PE4
TMRB3	PE7	PE6
TMRB4	PE3	PA7
TMRB5	PD1	PD0
TMRB6	PA5	PA6
TMRB7	PF1	PF0
TMRB8	PE0	PE1
TMRB9	PN1	PN2

表 2-3 割り込み

チャネル	キャプチャ割り込み	TMRB 割り込み
TMRB0	INTCAP00, INTCAP01	INTTB00, INTTB01
TMRB1	INTCAP10, INTCAP11	INTTB10, INTTB11
TMRB2	INTCAP20, INTCAP21	INTTB20, INTTB21
TMRB3	INTCAP30, INTCAP31	INTTB30, INTTB31
TMRB4	INTCAP40, INTCAP41	INTTB40, INTTB41
TMRB5	INTCAP50, INTCAP51	INTTB50, INTTB51
TMRB6	INTCAP60, INTCAP61	INTTB60, INTTB61
TMRB7	INTCAP70, INTCAP71	INTTB70, INTTB71
TMRB8	INTCAP80, INTCAP81	INTTB80, INTTB81
TMRB9	INTCAP90, INTCAP91	INTTB90, INTTB91

表 2-4 同期スタート仕様

マスタチャネル	スレーブチャネル
TMRB0	TMRB1, TMRB2, TMRB3
TMRB4	TMRB5, TMRB6
TMRB7	TMRB8, TMRB9

表 2-5 キャプチャトリガ仕様

トリガ入力チャネル	トリガ出力
TMRB0	ENC0 分周パルス
TMRB1	ENC1 分周パルス

2.1.3 シリアルチャネル(SIO/UART)

TMPM470FDFG/FZFG/FYFG では 4 チャネルの SIO/UART を内蔵しています。

表 2-6 端子仕様

チャネル	SCxTXD	SCxRXD	SCxSCLK	$\overline{\text{SCxCTS}}$
SC0	PE0	PE1	PE2	PE2
SC1	PA5	PA6	PA4	PA4
SC2	PD5	PD6	PD4	PD4
SC3	PF3	PF4	PF2	PF2

表 2-7 割り込み

チャネル	受信割り込み	送信割り込み
SC0	INTRX0	INTTX0
SC1	INTRX1	INTTX1
SC2	INTRX2	INTTX2
SC3	INTRX3	INTTX3

表 2-8 内部接続

チャネル	転送クロック入力
SC0	TMRB4
SC1	TMRB4
SC2	TMRB7
SC3	TMRB7

2.1.4　シリアルバスインタフェース(I2C)

TMPM470FDFG/FZFG/FYFG では 1 チャネルの I2C を内蔵し、SIO をサポートしていません。

表 2-9 端子仕様

チャネル	SBxSDA	SBxSCL
I2C0	PN0	PN1

表 2-10 割り込み

チャネル	割り込み
I2C0	INTSBI0

2.1.5　ベクトルエンジン (A-VE)

TMPM470FDFG/FZFG/FYFG では 2 チャネルの A-VE を内蔵しています。

表 2-11 割り込み

チャネル	割り込み
VE0	INTVCN0
VE1	INTVCN1

表 2-12 内部接続仕様

チャネル	PWM 割り込み信号入力	ADC 変換終了信号入力	ADC 変換結果入力
VE0	INTPMD0	INTADAPDA INTADBPDA	ADAREG0 to 3
VE1	INTPMD1	INTADAPDB INTADBPDB	ADBREG0 to 3

2.1.6 モータ制御回路 (PMD)

TMPM470FDFG/FZFG/FYFG では 2 チャネルの PMD を内蔵しています。

表 2-13 端子仕様

チャネル	$\overline{\text{OVVx}}$ 状態信号	$\overline{\text{EMGx}}$ 状態信号	ZOx Z 相出力	WOx W 相出力	YOx Y 相出力	VOx V 相出力	XOx X 相出力	UOx U 相出力
PMD0	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
PMD1	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0

表 2-14 割り込み

チャネル	OVV 割り込み	EMG 割り込み	PWM 割り込み
PMD0	INTOVV0	INTEMG0	INTPMD0
PMD1	INTOVV1	INTEMG1	INTPMD1

表 2-15 内部接続仕様 (1/2)

チャネル	EMG 復帰入力 (VE)	PWM コンペア入力 (VE)	出力制御入力 (VE)	トリガコンペア入力 (VE)	トリガ選択入力 (VE)	OVV 入力 (ADC)
PMD0	VEEMGRS0	VECMPU0, V0, W0	VEOUTCR0	VETRGCMP00, 01	VETRGSEL0	ADCA 監視機能比較 0, 1 出力
PMD1	VEEMGRS1	VECMPU1, V1, W1	VEOUTCR1	VETRGCMP10, 11	VETRGSEL1	ADCB 監視機能比較 0, 1 出力

表 2-16 内部接続仕様 (2/2)

チャネル	MDOUT 転送タイミング信号入力
PMD0	INTENC0, INTTB00, CTRGO(ENC0)
PMD1	INTENC1, INTTB10, CTRGO(ENC1)

2.1.7 エンコーダ (A-ENC)

TMPM470FDFG/FZFG/FYFG では 2 チャネルの A-ENC を内蔵しています。

表 2-17 端子仕様

チャネル	ENCAx	ENCBx	ENCZx
ENC0	PD0	PD1	PD2
ENC1	PF2	PF3	PF4

表 2-18 割り込み

ユニット	割り込み
ENC0	INTENC0
ENC1	INTENC1

表 2-19 内部接続仕様

チャネル	タイマパルス入力	PWM 信号入力
ENC0	TB0OUT	PWMON0
ENC1	TB1OUT	PWMON1

2.1.8　アナログ/デジタルコンバータ(ADC)

TMPM470FDFG/FZFG/FYFG では 2 ユニットの 12 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。モータ制御用ベクトルエンジン、および PMD 回路と連携してモータのベクトル制御を支援します。

表 2-20 端子仕様

ユニット	AINA0 to 7	AINA8	AINA9 to 11 AINB0 to 2	AINB3 to 10
ADCA	PH0 to 7	PP0	PP1 to 3	-
ADCB	-	-	PP1 to 3	PJ0 to 7

表 2-21 割り込み

ユニット	PMD トリガ同期変換終了割り込み	タイマ同期変換終了割り込み	ソフトスタート変換終了割り込み	監視機能割り込み
ADCA	INTADAPDB	INTADATMR	INTADASFT	INTADACPA INTADACPB
ADCB	INTADBPDB	INTADBTMR	INTADBSFT	INTADBCPA INTADBCPB

表 2-22 内部接続仕様

ユニット	PMD トリガ入力	TMRTRG
ADCA	PMD0TRG0 to 5	INTTB50
ADCB	PMD1TRG0 to 5	INTTB60

2.1.9　ウォッチドッグタイマ(WDT)

TMPM470FDFG/FZFG/FYFG では WDMOD レジスタの I2WDT ビットをサポートしていませんので、必ず、"0"をライトしてください。

2.1.10　デバッグインタフェース

TMPM470FDFG/FZFG/FYFG はシリアルワイヤデバッグポート、JTAG デバッグポートおよびトレース出力をサポートしています。

表 2-23 端子仕様

	TMS SWDIO	TCK SWCLK	TDO SWV	TDI	$\overline{\text{TRST}}$
JTAG シリアルワイヤ	PB3	PB4	PB5	PB6	PB7

	TRACECLK	TRACEDATA0	TRACEDATA1	TRACEDATA2	TRACEDATA3
トレース出力	PB0	PB1	PB2	PK1	PK0

第 3 章　プロセッサコア

TMPM470 シリーズには、高性能 32 ビットプロセッサコア(ARM 社 Cortex-M4F 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ドキュメンテーションセット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。

3.1　コアに関する情報

TMPM470FDFG/FZFG/FYFG で使用している Cortex-M4F 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、下記 URL より ARM 社の"Cortex-M4 プロセッサ用ドキュメンテーションセット"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
TMPM470FDFG/FZFG/FYFG	r0p1

3.2　構成可能なオプション

Cortex-M4F 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。

TMPM470FDFG/FZFG/FYFG での構成は以下のとおりです。

Configurable options	Implementation
MPU (Memory Protection Unit)	Absent
FPB (Flash Patch and Breakpoint)	Two literal comparators Six instruction comparators
DWT (Data Watchpoint and Trace)	Four comparators
ITM (Instrumentation Trace Macrocell)	Present
ETM (Embedded Trace Macrocell)	Present
AHB-AP (AHB Access Port)	Present
HTM Interface (AHB Trace Macrocell Interface)	Absent
TPIU (Trace Port Interface Unit)	Present
WIC (Wake-up Interrupt Controller)	Absent
Debug Port (Serial-Wire or JTAG Debug Port)	Present
FPU (Floating Point Unit)	Present
Bit banding	Present
Constant AHB control	Disable

3.3　例外/割り込み

例外/割り込みに関連する製品固有の情報をまとめます。

3.3.1　割り込み本数

Cortex-M4F 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。

TMPM470FDFG/FZFG/FYFG の割り込み本数は 93 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[3:0]>ビットに反映され、本製品では"0x02"が読み出されます。

3.3.2　割り込み優先度ビット数

Cortex-M4F 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。

TMPM470FDFG/FZFG/FYFG の割り込み優先度は 3 ビットです。このビット数は割り込み優先度レジスタとシステムハンドラ優先度レジスタのビット構成に反映されます。

3.3.3　SysTick

Cortex-M4F コアには SysTick と呼ばれるシステムタイマがあり、SysTick 例外を発生させることができます。

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

3.3.4　SYSRESETREQ

Cortex-M4F 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。

TMPM470FDFG/FZFG/FYFG 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。

3.3.5　LOCKUP

回復不能な例外が発生すると Cortex-M4F コアは LOCKUP 信号を出力し、ソフトウエアに重大な誤りのあることを示します。

TMPM470FDFG/FZFG/FYFG ではこの信号は未使用です。LOCKUP 状態から復帰する際にはマスク不能割り込み(NMI)またはリセットを使用する必要があります。

3.3.6　補助フォールトステータスレジスタ

Cortex-M4F コアにはソフトウエ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。

TMPM470FDFG/FZFG/FYFG ではこのレジスタに対して機能を定義していません。リードすると常に"0x0000_0000"が読み出されます。

3.4 イベント

Cortex-M4F 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。

TMPM470FDFG/FZFG/FYFG では、イベント出力信号、イベント入力信号とも未使用です。SEV 命令、WFE 命令は使用しないでください。

3.5 電力管理

Cortex-M4F 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。

TMPM470FDFG/FZFG/FYFG では、SLEEPDEEP 信号は使用していません。<SLEEPDEEP>ビットはセットしないでください。また、イベント信号も未使用のため、WFE 命令は使用しないでください。

電力管理については、「クロック／モード制御」の章を参照してください。

3.6 排他アクセス

Cortex-M4F コアの DCode バスおよびシステムバスは排他アクセスをサポートしていますが、TMPM470FDFG/FZFG/FYFG ではこの機能を使用していません。

3.7 浮動小数点演算装置(FPU)

本製品は、ARMv7M 浮動小数点拡張(FPv4-SP)の派生仕様の演算装置(FPU)を実装しており、IEEE 標準(ANSI/IEEE Std 754-2008)に準拠した単精度浮動小数点演算が可能です。

この FPU は、アドレスバスとデータバスを Cortex-M4F コアと共有し、協調して動作します。加算/減算/乗算を 1 クロックで、積和を 3 クロックで実行します。また、CPU とは別に専用データレジスタによる並列処理が可能です。

なお、このFPU は、ARM アーキテクチャリファレンスマニュアルに掲載された、全ての単精度データ演算命令とデータタイプをサポートします。

第 4 章　メモリマップ

4.1　メモリマップ

TMPM470FDFG/FZFG/FYFG のメモリマップは、ARM Cortex-M4F 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-M4F 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。

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、"ARM ドキュメンテーションセット Cortex-M4F 編"を参照してください。

"Fault"と記載された領域では、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、ベンダ固有領域にはアクセスしないでください。

TMPM470FDFG/FZFG/FYFG のメモリマップを以下に示します。

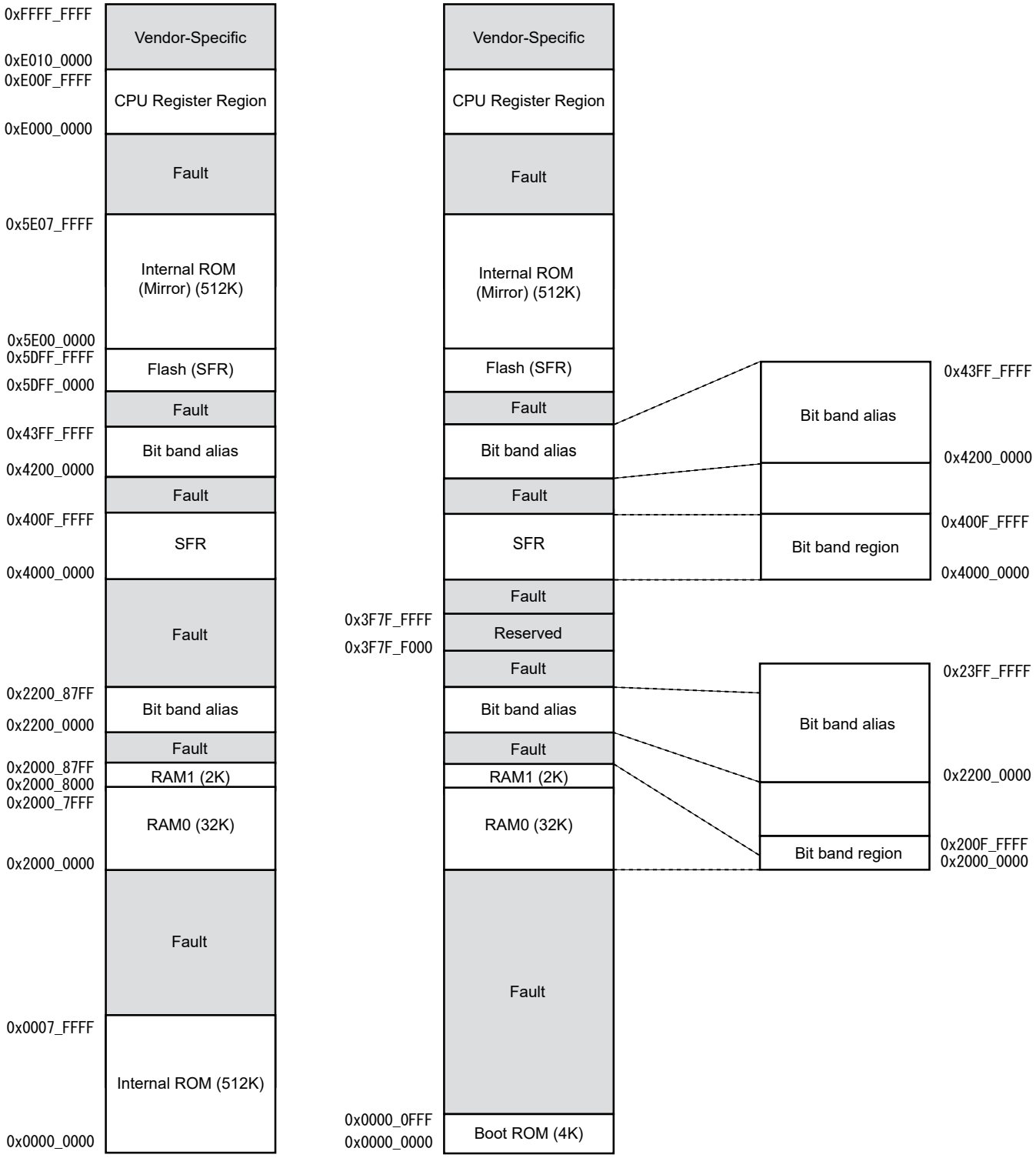


図 4-1 メモリマップ(512KB 製品）

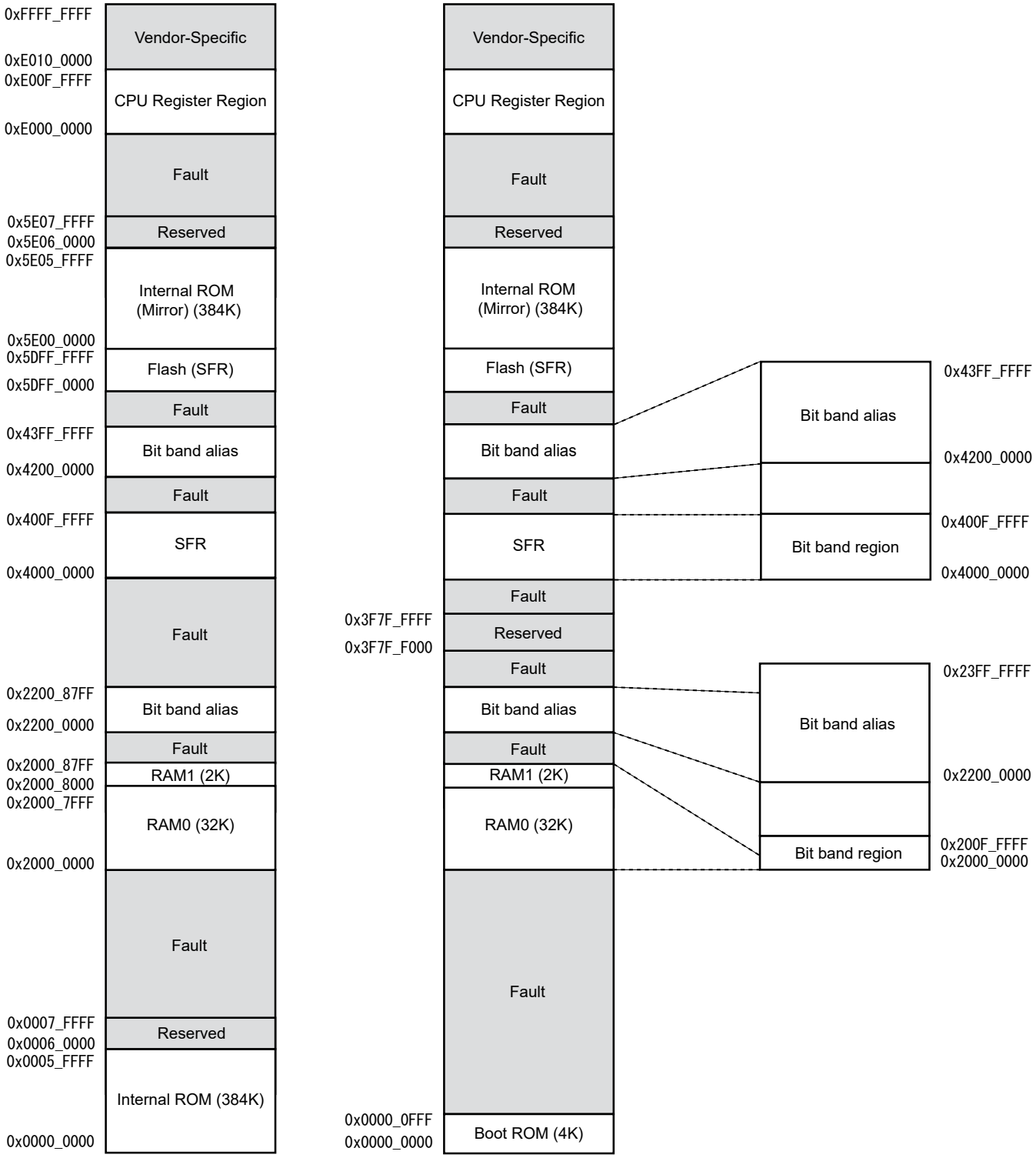


図 4-2 メモリマップ(384KB 製品)

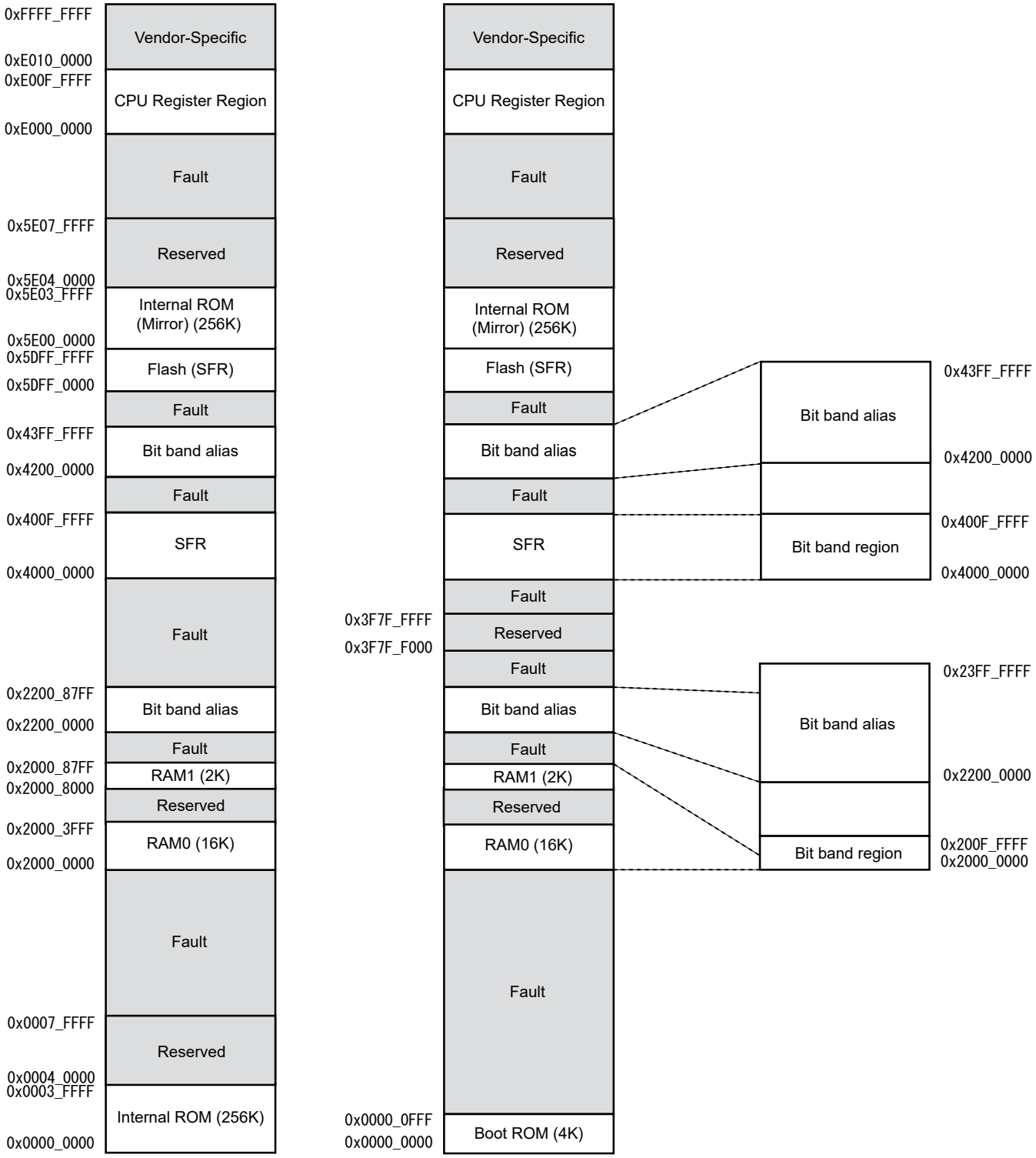


図 4-3 メモリマップ(256KB 製品)

4.2 バスマトリクス

本マイコンでは、CPU コア、μDMA コントローラ 2 種類のバスマスタが搭載されています。

バスマスタは、バスマトリクスのスレーブポート(S0~S5)に接続され、バスマトリクス内で、接続を示す記号(○,●)を経由して、マスタポート(M0~M15)から、周辺機能に接続されます。●は、ミラー領域への接続を示します。

バスマトリクス内の同一マスタライン上に、複数のスレーブが接続されている場合で、同一タイミングで複数のスレーブにアクセスが発生した場合は、スレーブ番号の小さいマスタのアクセスが優先されます。

4.2.1　構成

4.2.1.1　シングルチップモード

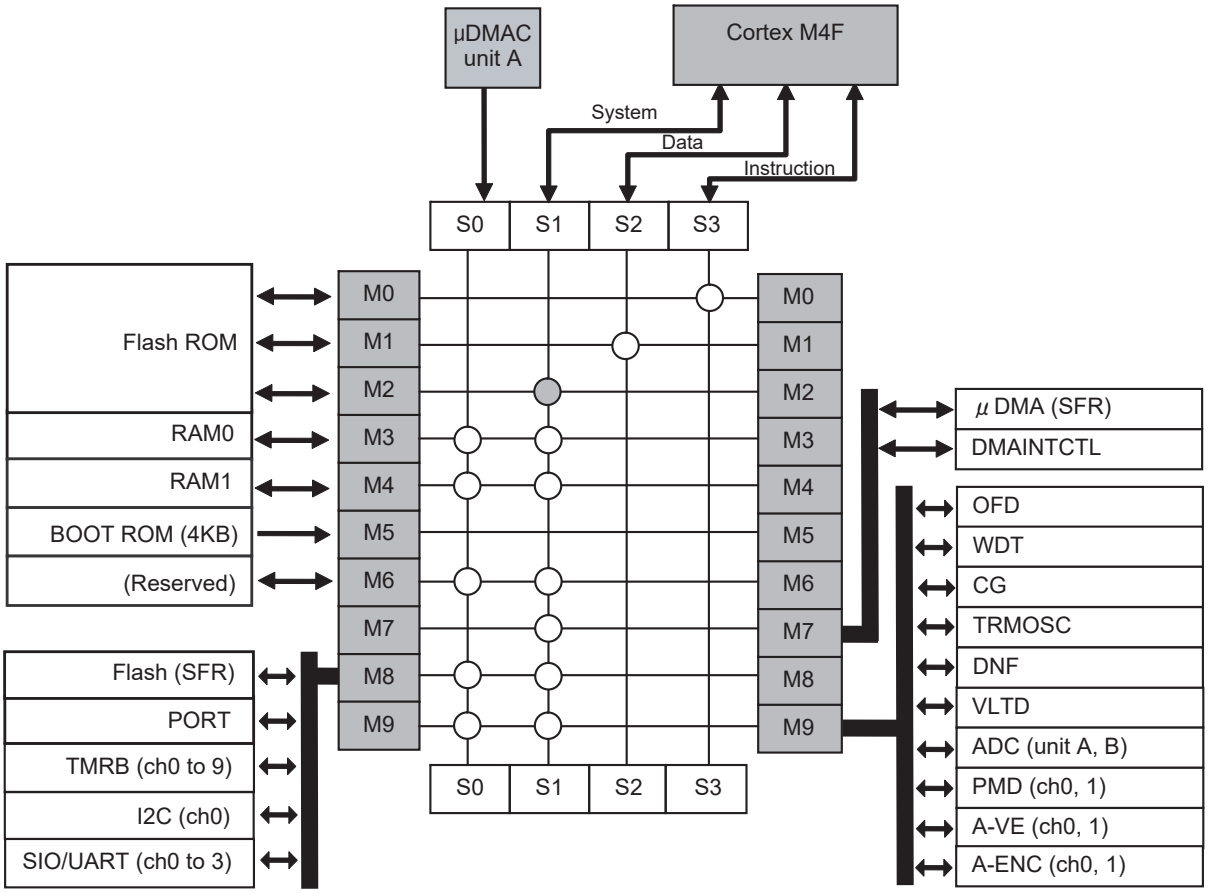


図 4-4 TMPM470FDFG/FZFG/FYFG のバス構成

注)　Reserved 記載のマスタにはアクセスしないで下さい。

4.2.1.2 シングルブートモード

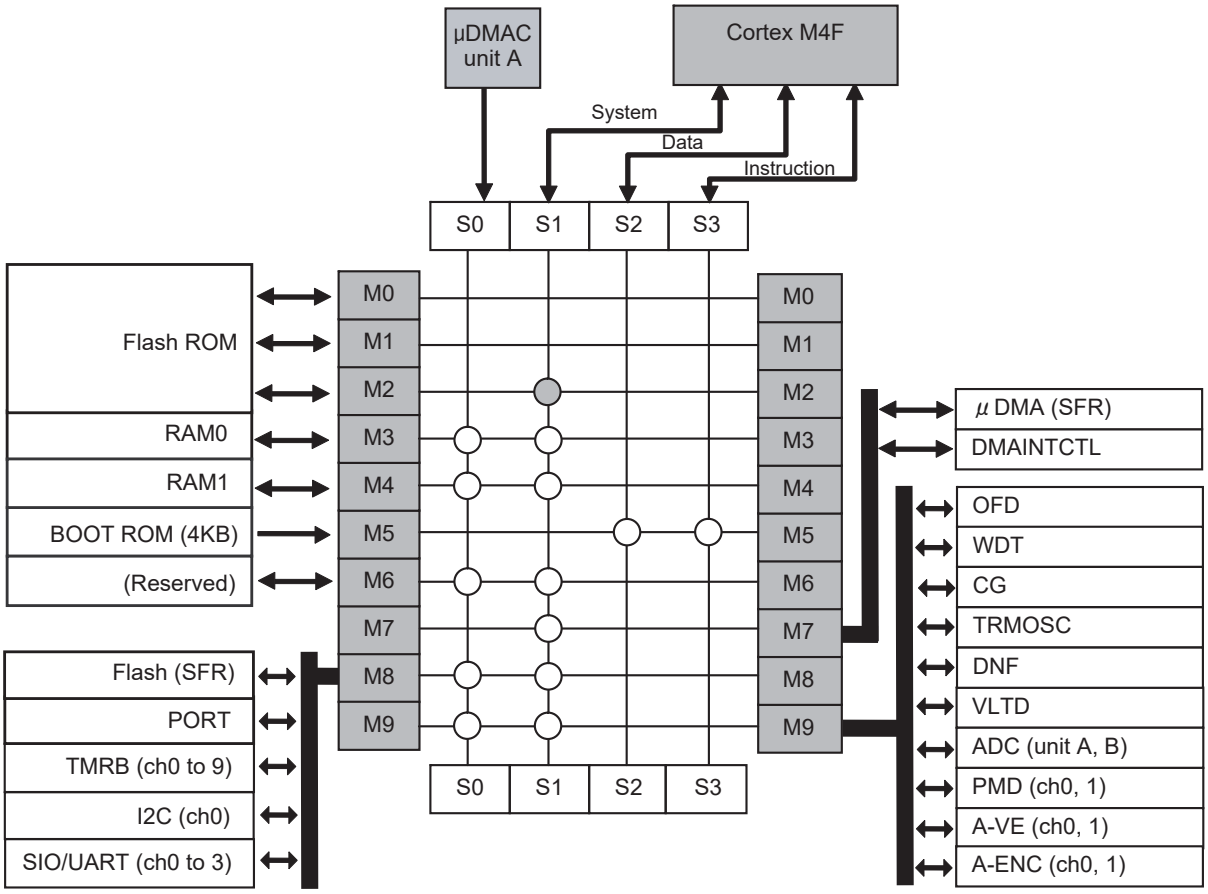


図 4-5 TMPM470FDFG/FZFG/FYFG のバス構成

注) Reserved 記載のマスタにはアクセスしないで下さい。

4.2.2　接続表

4.2.2.1　Code 領域/ SRAM 領域

(1)　シングルチップモード

Start Address	Slave	Master	μDMAC unitA	Core S-Bus	Core D-Bus	Core I-Bus
			S0	S1	S2	S3
0x0000_0000	Flash ROM	M0 M1	Fault	Fault	o	o
0x0010_0000	Fault	-	Fault	Fault	Fault	Fault
0x2000_0000	RAM0	M3	o	o	Fault	Fault
0x2000_8000	RAM1	M4	o	o	Fault	Fault
0x2000_8800	Fault	-	Fault	Fault	Fault	Fault
0x2200_0000	Bit band alias	-	Fault	o	Fault	Fault
0x2400_0000	Fault	-	Fault	Fault	Fault	Fault

(2)　シングルブートモード

Start Address	Slave	Master	μDMAC unitA	Core S-Bus	Core D-Bus	Core I-Bus
			S0	S1	S2	S3
0x0000_0000	Boot ROM	M5	Fault	Fault	o	o
0x0000_1000	Fault	-	Fault	Fault	Fault	Fault
0x2000_0000	RAM0	M3	o	o	Fault	Fault
0x2000_8000	RAM1	M4	o	o	Fault	Fault
0x2003_0400	Fault	-	Fault	Fault	Fault	Fault
0x2200_0000	Bit band alias	-	Fault	o	Fault	Fault
0x2400_0000	Fault	-	Fault	Fault	Fault	Fault
0x3F7F_F000	Reserved	-	Fault	Reserved	Fault	Fault
0x3F80_0000	Fault	-	Fault	Fault	Fault	Fault

注)　Reserved 記載のアドレス範囲にはアクセスしないで下さい。

4.2.2.2 Peripheral 領域

Start Address	Slave	Master	µDMAC unitA	Core S-Bus	Core D-Bus	Core I-Bus
			S0	S1	S2	S3
0x4000_0000	Fault	-	Fault	Fault	Fault	Fault
0x4004_C000	µDMAC unitA (SFR)	M7	-	o	Fault	Fault
0x4005_F000	DMAIF	M7	-	o	Fault	Fault
0x400C_0000	PORT	M8	o	o	Fault	Fault
0x400C_4000	TMRB (ch0 to -9)	M8	o	o	Fault	Fault
0x400E_0000	I2C (ch0)	M8	o	o	Fault	Fault
0x400E_1000	SIO/UART(ch0 to 3)	M8	o	o	Fault	Fault
0x400F_1000	OFD	M9	o	o	Fault	Fault
0x400F_2000	WDT	M9	o	o	Fault	Fault
0x400F_3000	CG	M9	o	o	Fault	Fault
0x400F_3200	TRMOSC	M9	o	o	Fault	Fault
0x400F_3400	DNF	M9	o	o	Fault	Fault
0x400F_4000	LVD	M9	o	o	Fault	Fault
0x400F_6000	PMD (ch0,1)	M9	o	o	Fault	Fault
0x400F_7000	A-ENC (ch0,1)	M9	o	o	Fault	Fault
0x400F_8000	A-VE (ch0, 1)	M9	o	o	Fault	Fault
0x400F_D000	ADC (unit A, B)	M9	o	o	Fault	Fault
0x4010_0000	Fault	-	Fault	Fault	Fault	Fault
0x4200_0000	Bit band alias	-	Fault	o	Fault	Fault
0x4400_0000	Fault	-	Fault	Fault	Fault	Fault
0x5DFF_0000	Flash (SFR)	M8	-	o	Fault	Fault
0x5E00_0000	Flash(Mirror)	M2	o	o	Fault	Fault
0x5E08_0000	Fault	-	Fault	Fault	Fault	Fault

4.2.3　周辺機能ベースアドレス一覧

周辺機能のベースアドレスを示します。周辺機能はすべて SFR0 領域です。

SFR0 領域のうち、制御レジスタ以外のアドレスにはアクセスしないでください。制御レジスタの詳細は、各周辺機能の章を参照してください。

周辺機能		ベースアドレス	領域
μDMA コントローラ(μDMAC)	unit A	0x4004_C000	SFR0
	DMAIF	0x4005_F000	SFR0
入出力ポート	PORTA	0x400C_0000	SFR0
	PORTB	0x400C_0100	SFR0
	PORTC	0x400C_0200	SFR0
	PORTD	0x400C_0300	SFR0
	PORTE	0x400C_0400	SFR0
	PORTF	0x400C_0500	SFR0
	PORTG	0x400C_0600	SFR0
	PORTH	0x400C_0700	SFR0
	PORTJ	0x400C_0800	SFR0
	PORTK	0x400C_0900	SFR0
	PORTL	0x400C_0A00	SFR0
	PORTN	0x400C_0C00	SFR0
	PORTP	0x400C_0D00	SFR0
16 ビットタイマ/ イベントカウンタ(TMRB)	ch0	0x400C_4000	SFR0
	ch1	0x400C_4100	SFR0
	ch2	0x400C_4200	SFR0
	ch3	0x400C_4300	SFR0
	ch4	0x400C_4400	SFR0
	ch5	0x400C_4500	SFR0
	ch6	0x400C_4600	SFR0
	ch7	0x400C_4700	SFR0
	ch8	0x400C_4800	SFR0
	ch9	0x400C_4900	SFR0
シリアルバスインタフェース(I2C)	ch0	0x400E_0000	SFR0
シリアルチャネル(SIO/UART)	ch0	0x400E_1000	SFR0
	ch1	0x400E_1100	SFR0
	ch2	0x400E_1200	SFR0
	ch3	0x400E_1300	SFR0
周波数検知回路(OFD)	unit 0	0x400F_1000	SFR0
ウォッチドッグタイマ(WDT)	ch0	0x400F_2000	SFR0
クロック/モード制御(CG)		0x400F_3000	SFR0
内蔵高速発振調整機能(TRMOSC)	unit 0	0x400F_3200	SFR0
デジタルノイズフィルタ回路(DNF)	unit 0	0x400F_3400	SFR0
電圧検出回路(VLTD)	unit 0	0x400F_4000	SFR0
モータ制御回路(PMD)	ch0	0x400F_6000	SFR0
	ch1	0x400F_6100	SFR0
エンコーダ入力回路(A-ENC)	ch0	0x400F_7000	SFR0
	ch1	0x400F_7100	SFR0

周辺機能		ベースアドレス	領域
ベクトルエンジン(A-VE)	ch0	0x400F_8000	SRF0
	ch1	0x400F_8400	SFR0
アナログ/デジタルコンバータ(ADC)	unit A	0x400F_D000	SFR0
	unit B	0x400F_D100	SFR0
フラッシュ/デバッグ(FC)	ch0	0x5DFF_0000	SFR0

第 5 章　リセット動作

リセットの種類として以下のものがあります。

- パワーオンリセット回路(POR)
- 電圧検出回路(VLTD)
- リセット端子($\overline{\text{RESET}}$)
- ウォッチドッグタイマ(WDT)
- 周波数検知回路(OFD)
- CPU 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>ビット

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。CGRSTFLG については、例外の章を参照してください。

パワーオンリセット回路、電圧検出回路、ウォッチドッグタイマ、周波数検知回路についてはそれぞれの章を参照してください。

<SYSRESETREQ>によるリセットについては、"ARM ドキュメンテーションセット"を参照してください。

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。

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、内蔵フラッシュメモリおよび内蔵高速発振の安定時間を考慮する必要があります。本製品では、これらの機能の安定のための時間を内部回路が自動的に挿入します。この回路は、電源電圧がパワーオンリセット解除電圧を超えると、パワーオンリセット回路(POR)の出力によってパワーオンカウンタが動作を開始し、tPWUP(fiosc が 10MHz の場合に約 0.9ms)の間、POR 引き伸ばし信号を有効にします。

また、本製品では、電源の電圧検出回路(VLTD)の動作が許可されています。

パワーオンリセット回路の動作については、「パワーオンリセット回路(POR)」の章を参照してください。電源検知回路の動作については、「電源検知回路(VLTD)」の章を参照してください。

5.1.1　POR によるリセット($\overline{\text{RESET}}$ 端子を使用しない場合)

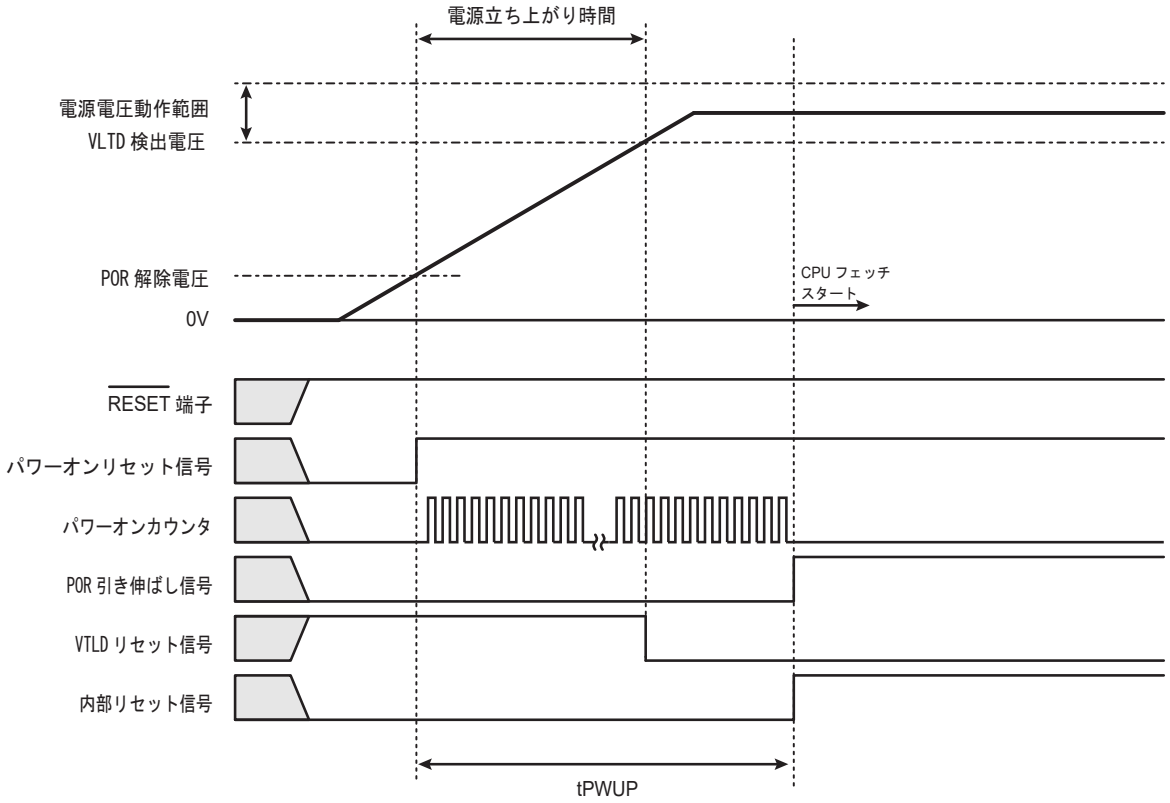


図 5-1 POR によるリセット動作($\overline{\text{RESET}}$ 端子を使用しない場合)

POR 引き伸ばし信号が"Low"のときに電源電圧を VLTD 検出電圧以上に立ち上げると、POR 引き伸ばし信号が"High"となるときに内部リセット信号が"High"となって、リセットが解除されます。

5.1.2 VLTD によるリセット(RESET 端子を使用しない場合)

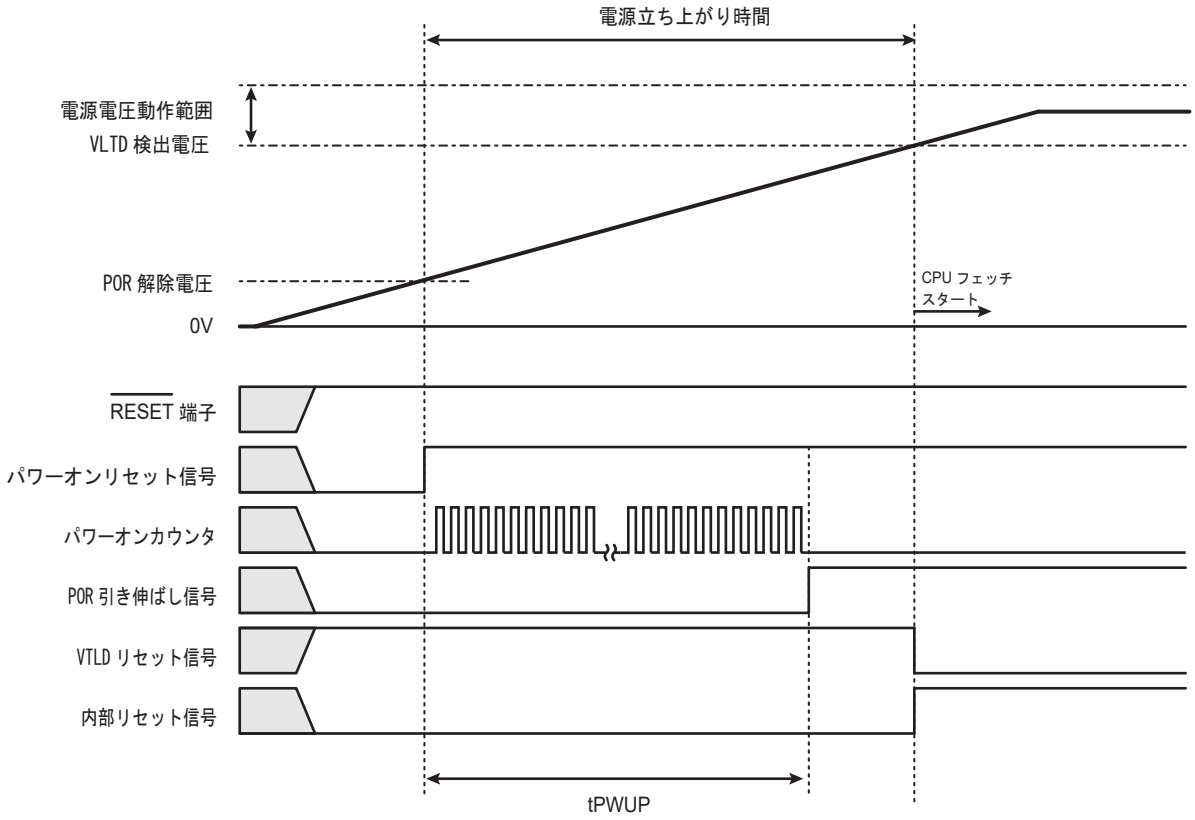


図 5-2 VLTD によるリセット動作(RESET 端子を使用しない場合)

POR 引き伸ばし信号が"High"のときに電源電圧を VLTD 検出電圧以上に立ち上げると、VLTD リセット信号が"Low"となるときに内部リセット信号が"High"となって、リセットが解除されます。

5.1.3　$\overline{\text{RESET}}$ 端子によるリセット(POR によるリセットが有効な場合)

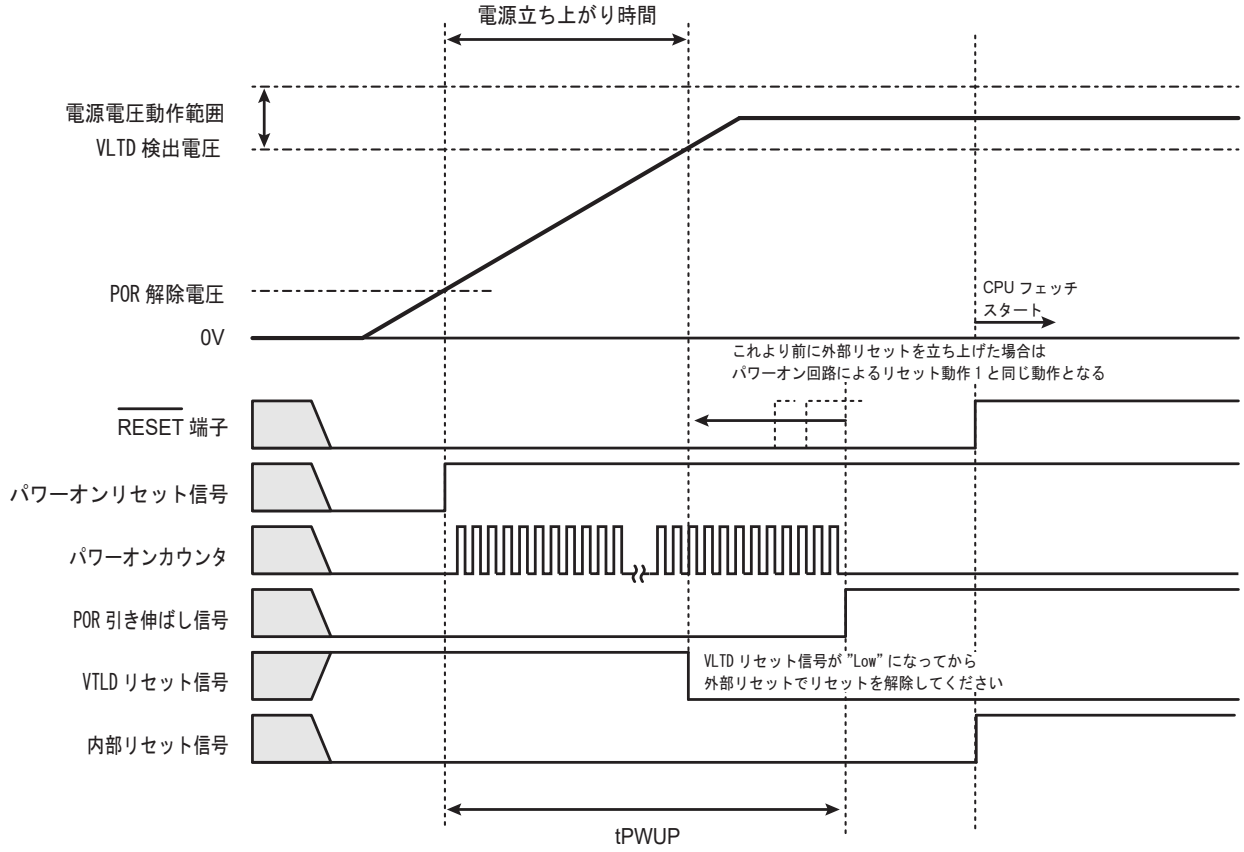


図 5-3 $\overline{\text{RESET}}$ 端子によるリセット動作(POR によるリセットが有効な場合)

POR によるリセットが有効な場合、POR 引き伸ばし信号が"High"になってから、$\overline{\text{RESET}}$ 端子を"High"とするときに内部リセット信号が"High"となって、リセットが解除されます。したがって、VLTD リセット信号が"Low"となる直後から POR 引き伸ばし信号が"High"となる直前まで間に $\overline{\text{RESET}}$ 端子を"High"にする場合は、1.1.1 の POR によるリセット動作と同じ動作になります。

なお、VLTD リセット信号が"High"のとき、すなわち、電源電圧が動作電圧未満のときには $\overline{\text{RESET}}$ 端子を"High"にしないでください。

5.1.4 $\overline{\text{RESET}}$ 端子によるリセット(VLTD によるリセットが有効な場合)

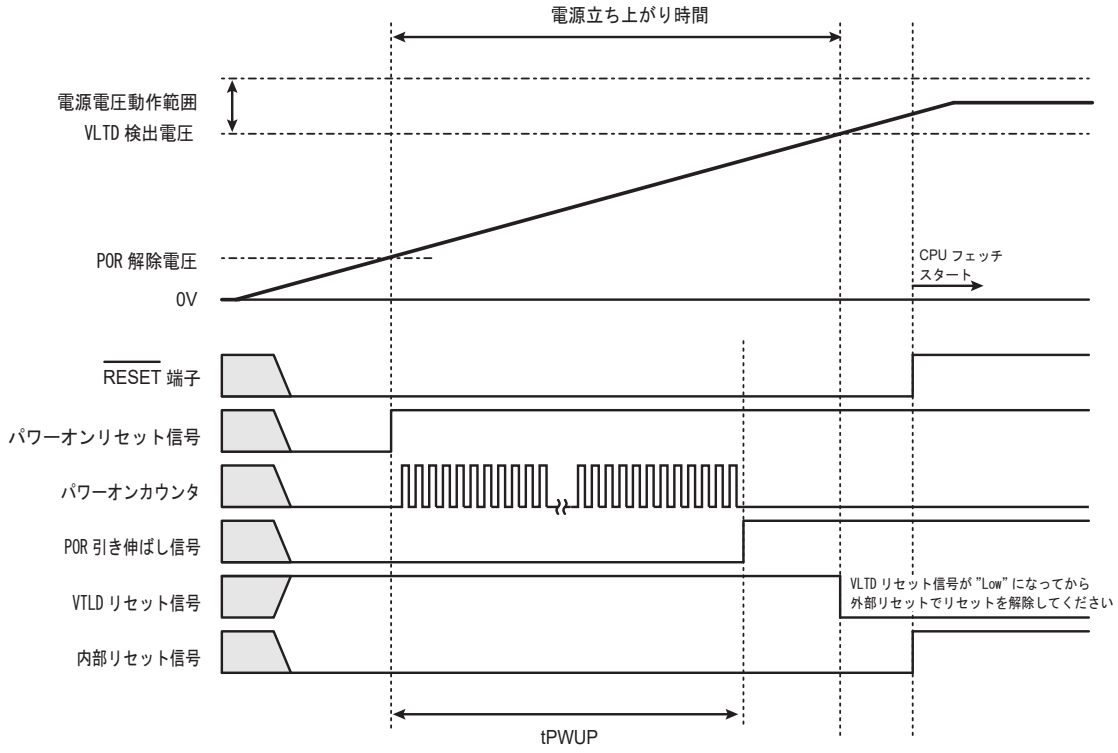


図 5-4 $\overline{\text{RESET}}$ 端子によるリセット動作(VLTD によるリセットが有効な場合)

VLTD によるリセットが有効な場合、VLTD リセット信号が"Low"となってから、$\overline{\text{RESET}}$ 端子を"High"とするときに内部リセット信号が"High"となって、リセットが解除されます。

なお、VLTD リセット信号が"High"のとき、すなわち、電源電圧が動作電圧未満のときには $\overline{\text{RESET}}$ 端子を"High"にしないでください。

5.2　ウォームリセット時

5.2.1　リセット期間

本製品にリセットをかけるには、電源電圧が動作範囲内であり、高速発振器の発振が安定した状態で、$\overline{\text{RESET}}$ 端子を少なくとも 12 システムクロック (fiosc が 10MHz の場合に最小 1.2 μs) 間"Low"にしてください。$\overline{\text{RESET}}$ 端子が"High"になってから内部リセットが解除されます。

5.3　リセット解除後

リセット解除後は、ほとんどの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 および FLASH 関連レジスタの FCSECBIT 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、内蔵高速発振器のクロックで動作を開始します。必要に応じて外部クロック、PLL 逓倍回路の設定を行ってください。

第 6 章　クロック/モード制御

6.1　特長

クロック/モード制御は、内部／外部発振、クロックギア、プリスケーラクロックの選択、PLL(逓倍回路)や発振器のウォーミングアップ等の制御を行います。

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。

6.2　レジスタ説明

6.2.1　レジスター覧

クロック/モード制御のレジスタとアドレスを以下に示します。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照してください。

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

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	-	-	-	-	-	-	-	-
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	−	R/W	"01"をライトしてください。
15-13	−	R	リードすると"0"が読めます。
12	FPSEL	R/W	fperiph のソースクロック選択 0: fgear 1: fc fperiph のソースクロックを選択します。 ソースクロックに fc を選択した場合、クロックギアの設定に関係なく、fperiph の周波数を固定することができます。
11	−	R	リードすると"0"が読めます。
10-8	PRCK[2:0]	R/W	プリスケーラクロック(φT0)の分周比選択 000: fperiph　100: fperiph/16 001: fperiph/2　101: fperiph/32 010: fperiph/4　110: Reserved 011: fperiph/8　111: Reserved 周辺機能に供給するプリスケーラクロック(φT0)の分周比を選択します。 周辺機能の動作中にプリスケーラクロックの分周比を切り替えないようにしてください。また、プリスケーラクロックの分周比を変更する場合、φT0 の周波数が fsys の周波数以下になるようにしてください。
7-3	−	R	リードすると"0"が読めます。
2-0	GEAR[2:0]	R/W	ギアクロック(fgear)の分周比選択 000: fc　100: fc/2 001: Reserved　101: fc/4 010: Reserved　110: fc/8 011: Reserved　111: fc/16 ギアクロックの分周比を選択します。 周辺機能の動作中にクロックギアの分周比を切り替えないようにしてください。また、クロックギアを使用する場合、φT0 の周波数が fsys の周波数以下になるようにしてください。

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	WUODR							
リセット後	1	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	WUODR				WUPSEL2	-	OSCSEL	XEN2
リセット後	0	0	0	0	0	0	0	1
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	XEN1
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	WUPSEL1	PLLON	WUEF	WUEON
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-20	WUODR[11:0]	R/W	高速発振器用ウォーミングアップタイマのカウンタ比較値設定 ウォーミングアップタイマのカウンタと比較する値を設定します。 ウォーミングアップ時間の計算値 16 ビットのうち、上位 12 ビットの値を設定します。
19	WUPSEL2	R/W	高速発振器用ウォーミングアップタイマのソースクロック選択(注 1) 0: 内部高速発振(fiosc) 1: 外部高速発振(feosc)
18	−	R/W	Reserved ("0"または"1"をライトしても動作に支障ありません。)
17	OSCSEL	R/W	高速発振器(fosc)のソースクロック選択 0: 内部高速発振(fiosc) 1: 外部高速発振(feosc) 高速発振器のソースクロックの選択を行います。レジスタ内容の更新は、切り替え先クロックの発振が安定している状態で行ってください。 また、設定を変更した場合、変更した値が<OSCSEL>に反映されていることを確認してください。
16	XEN2	R/W	内部高速発振器(IOSC)の動作選択 0: 禁止 1: 許可 内部高速発振器(fiosc)の制御を行います。 許可に設定した場合、ウォーミングアップタイマによる発振安定時間確保が必要です。詳細については、「6.3.5 ウォーミングアップ機能」を参照してください。
15-12	−	R/W	"0"をライトしてください。
11-10	−	R	リードすると"0"が読めます。
9	−	R/W	"0"をライトしてください。
8	XEN1	R/W	外部高速発振器(EOSC)の動作選択 0: 禁止 1:許可 外部高速発振器の動作を選択します。 許可に設定した場合、ウォーミングアップカウンタによる発振安定時間確保が必要です。詳細については、「6.3.5 ウォーミングアップ機能」を参照してください。
7-4	−	R	リードすると"0"が読めます。
3	WUPSEL1	R/W	ウォーミングアップタイマ(WUP)のクロック "0"をライトしてください。
2	PLLON	R/W	逓倍回路(PLL)の動作選択 0:停止 1:発振 逓倍回路(PLL)の動作を選択します。 許可にした場合、ウォーミングアップカウンタによる発振安定時間確保が必要です。詳細については、「6.3.5 ウォーミングアップ機能」を参照してください。

Bit	Bit Symbol	Type	機能
1	WUEF	R	高速発振器用ウォーミングアップタイマ動作ステータス 0: 動作終了 1: 動作中 ウォーミングアップタイマの状態を確認できます。
0	WUEON	W	高速発振器用ウォーミングアップタイマ制御 1: 動作スタート ウォーミングアップタイマの動作をスタート("1"を設定)させるまえに、必ず<WUEF>が"0"であることを確認してください。 <WUON>への"0"の書き込みは意味を持ちません。また、読み出すと"0"が読めます。

注) STOP モードを使用する場合、STOP モードに移行する前に<OSCSEL>で選択したクロックと同じクロックを設定してください。

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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	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"をライトしてください。
16-10	−	R	リードすると"0"が読めます。
9	−	R/W	"0"をライトしてください。
8	RXEN	R/W	STOP モード解除後の高速発振器の動作選択 "1"を設定してください。
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: Reserved 011: IDLE 100: Reserved 101: Reserved 110: Reserved 111: Reserved STOP モード起動時には内部高速発振器(IOSC)、または外部高速発振器(EOSC)のうち、システムクロックとして使用していない側の発振器は必ず停止させておいてください。

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	PLLSET							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PLLSET							PLLSEL
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。
15-1	PLLSET	R/W	PLL 逓倍数設定 (下記以外は設定禁止) 0x591E: 入力クロック 10MHz、出力クロック 80MHz 0x5926: 入力クロック 10MHz、出力クロック 100MHz 0x59AE: 入力クロック 10MHz、出力クロック 120MHz
0	PLLSEL	R/W	高速クロック(fc)のソースクロック選択 0: fosc 使用 1: f_{PLL} 使用 高速クロック(fc)のソースクロックの選択を行います。

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。

feosc　　: 外部高速発振器(EOSC)より出力されるクロック
fiosc　　: 内部高速発振器(IOSC)より出力されるクロック
fosc　　: 高速発振器から出力されるクロック
f_{PLL}　　: 逓倍回路(4 逓倍)から出力されるクロック
fc　　: fosc または f_{PLL} どちらか選択されたクロック(高速クロック)
fgear　　: 高速クロックを分周したクロック
fsys　　: fgear と同一クロック(システムクロック)
fperiph　　: CGSYSCR<FPSEL>で選択されたクロック
φT0　　: CGSYSCR<PRCK[2:0]>で選択されたクロック (プリスケーラクロック)

高速クロック fc と、プリスケーラクロック 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, fc/16
プリスケーラクロック　　: fperiph, fperiph/2, fperiph/4, fperiph/8, fperiph/16, fperiph/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。

外部高速発振器(EOSC)　　: 停止
内部高速発振器(IOSC)　　: 発振
逓倍回路(PLL)　　: 停止
高速クロックギア　　: fc (分周なし)

リセット動作により、すべてのクロックの設定が f_{OSC} と同じになります。

$f_C = f_{OSC}$
$f_{SYS} = f_C (= f_{OSC})$
$f_{periph} = f_C (= f_{OSC})$
$\phi T0 = f_{periph} (= f_{OSC})$

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。

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

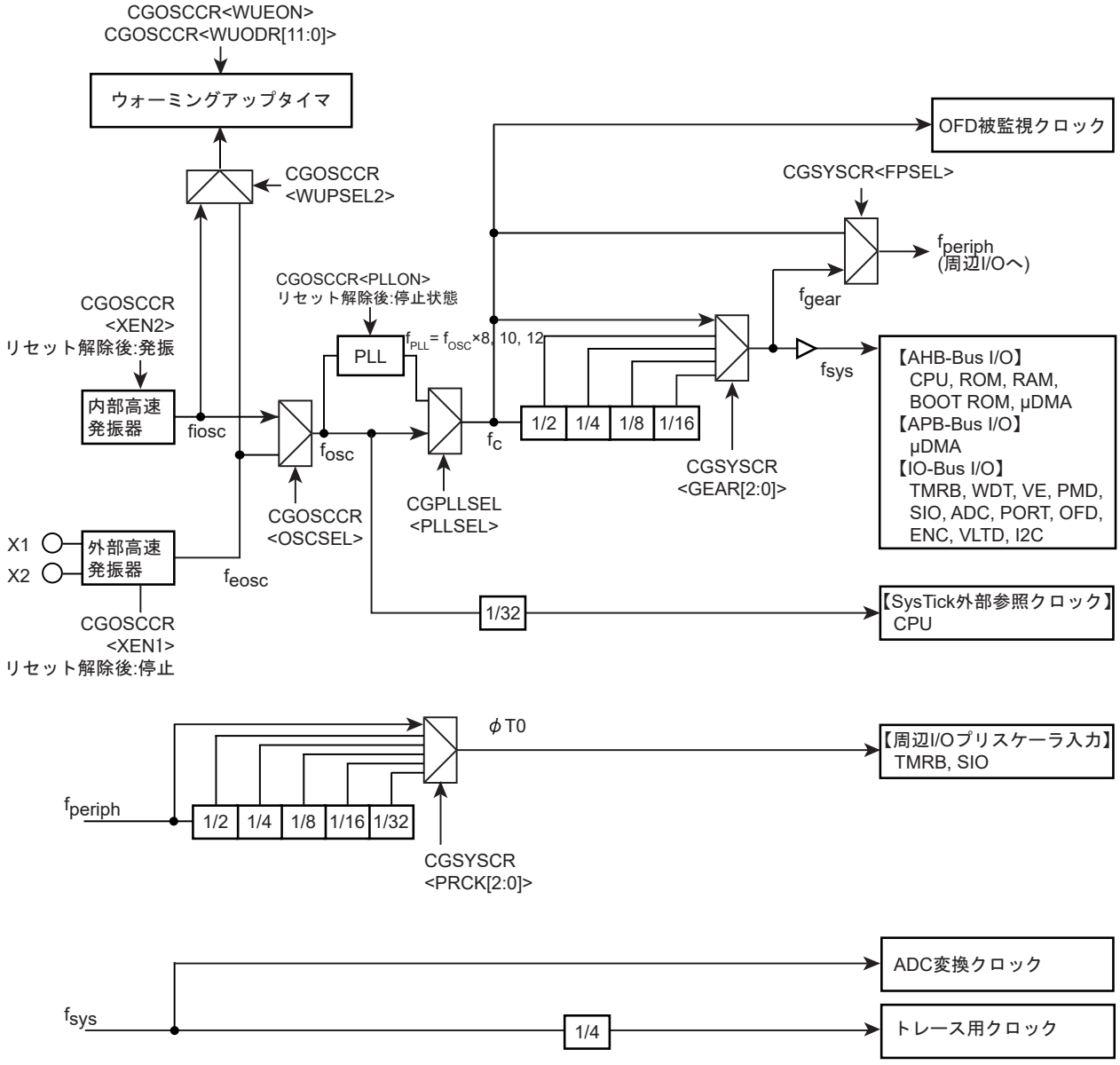


図 6-1 クロック系統図

6.3.4　クロック逓倍回路(PLL)

高速発振器の出力クロック fosc を 8, 10, 12 逓 倍したクロック(f_{PLL})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>を"1"に設定し、CGPLLSEL<PLLSEL>を"1"に設定することで fosc を 8, 10, 12 逓 倍したクロック(f_{PLL})を出力することができます。なお、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。

6.3.4.1　安定時間

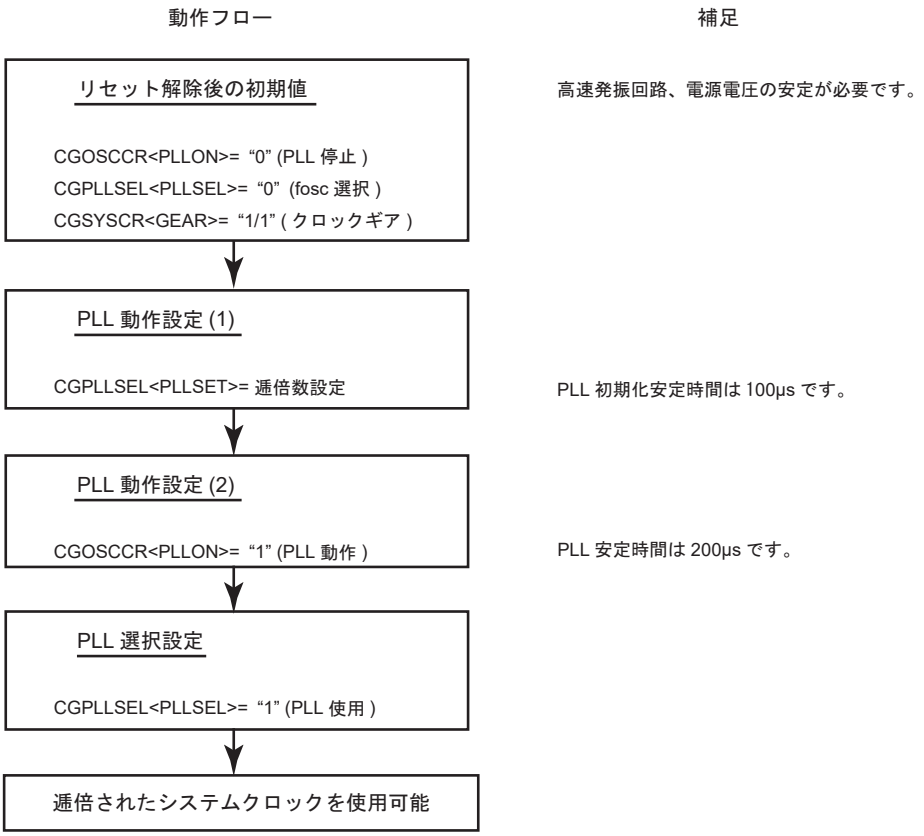
PLL 動作開始および、逓倍値の変更の際にはウォーミングアップ機能等を用いて安定時間を確保する必要があります。

動作を開始するときはロックアップ時間として約 200μs 必要です。

逓倍数の変更を行う場合、まず CGPLLSEL<PLLSEL> = "0"として逓倍クロックを使用しない設定に切り替えた上で<PLLON> を"0"として PLL を停止します。<PLLSET>の逓倍値を変更し、PLL の初期化時間として約 100μs 経過後に<PLLON>を"1"として PLL の動作を開始します。その後、ロックアップ時間(PLL 安定時間)を確保してください。

6.3.4.2 PLL 設定シーケンス

以下にリセット解除後の PLL 設定シーケンスを示します。



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。発振が安定している外部クロックなどを使用する場合にはウォーミングアップを行う必要はありません。

詳細機能については「6.6.6 ウォーミングアップ」にて説明します。

注)　ウォーミングアップタイマ動作中に、低消費電力モードへ遷移しないでください。

ウォーミングアップ機能は、STOP 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。

ウォーミングアップ機能の設定方法を説明します。

1. クロックの選択

   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<WUPSEL1>および<WUPSEL2>で選択します。(<WUPSEL1>は"0"を、<WUPSEL2>は"0"または"1"を設定してください。)

2. ウォーミングアップカウンタ設定値の算出

   ウォーミングアップ時間は CGOSCCR<WUODR[11:0]>により任意の値が設定可能です。CGOSCCR<WUODR[11:0]>の設定値は、以下の計算式から算出し、下位 4 ビットを切り捨てて上位 12 ビットを設定します。

   以下にウォーミングアップ時間の計算式と設定例を示します。

$$\text{ウォーミングアップサイクル数} = \frac{\text{ウォーミングアップ時間}}{\text{ウォームアップクロック周期}}$$

   <例>高速発振子 8MHz 使用時、ウォーミングアップ時間 5ms を設定する場合

$$\frac{\text{ウォーミングアップ時間}}{\text{ウォームアップクロック周期}} = \frac{5\text{ms}}{1/8\text{MHz}} = 40{,}000\text{サイクル} = 0\text{x}9\text{C}40$$

   下位 4 ビットを切り捨て、0x9C4 を CGOSCCR<WUODR[11:0]>に設定します。

3. ウォーミングアップの開始および終了確認

   ソフトウエア(命令)によりウォーミングアップタイマの開始および終了確認を行う場合、CGOSCCR<WUEON>に"1"を設定することでウォーミングアップを開始します。また、終了の確認は<WUEF>で行います。<WUEF>が"1"でウォーミングアップ中、"0"で終了を示します。

注) ウォーミングアップタイマ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。

ウォーミングアップ機能設定例 (外部発振の場合)

CGOSCCR<WUPSEL1> = "0"	:ウォーミングアップタイマクロック設定
CGOSCCR<WUPSEL2> = "1"	:ウォーミングアップタイマクロック設定(1:外部(feosc))
CGOSCCR<WUODR[11:0]> = "0x9C4"	:ウォーミングアップ時間設定
内部発振器から外部発振器への切り替えは 6.3.6 を参照してください	
CGOSCCR<WUEON>="1"	: ウォーミングアップタイマ(WUP)スタート
CGOSCCR<WUEF>リード	: "0"(WUP 終了)になるまでウエイト

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のみですが、内部発振器と外部発振器のどちらかを選択して使用します。リセット解除時は内部発振器が有効（発振）となっており、外部発振器は停止しています。高速クロックは分周することができます。

- X1, X2 入力周波数: 10MHz
- 内部発振入力周波数:10MHz
- クロックギア: 1/1, 1/2, 1/4, 1/8, 1/16 (リセット後は 1/1 分周)

表 6-1 高速クロック範囲(単位は MHz)

入力周波数 feosc fiosc	逓倍数	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	1/16	1/1	1/2	1/4	1/8	1/16
10	8	1.25	80	10	80	40	20	10	5	10	5	2.5	1.25	-
10	10	1.25	100	10	100	50	25	12.5	6.25	10	5	2.5	1.25	-
10	12	1.25	120	10	120	60	30	15	7.5	10	5	2.5	1.25	-

注 1)　PLL=ON / OFF は CGOSCCR<PLLON>で設定します。
注 2)　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 3)　PLL=OFF 時、クロックギアの 1/16 は設定しないでください。
注 4)　SysTick 使用時、クロックギアの 1/16 は設定しないでください。
注 5)　85°C ~ 105°C で使用する場合、最大動作周波数は 32MHz となります。PLL 使用時は分周機能を使うなどして最大動作周波数を超えないよう注意してください。

内部発振器から外部発振器への切り替え手順を以下に示します。

1. CGOSCCR<WUODR[11:0]> = "ウォーミングアップ時間"　:ウォーミングアップ時間設定
2. CGOSCCR<XEN1> = "1"　: 外部高速発振の発振
3. CGOSCCR<WUPSEL2> = "1"　:ウォーミングアップタイマクロック設定(1:外部(feosc))
4. CGOSCCR<WUEON>="1"　: ウォーミングアップタイマ(WUP)スタート
   CGOSCCR<WUEF>リード　: "0" (WUP 終了)になるまでウエイト
5. CGOSCCR<OSCSEL> = "1"　:高速発振器を外部発振器(feosc)に切り替え
6. CGOSCCR<OSCSEL>リード　: 外部発振選択を確認(<OSCSEL> = "1")
7. CGOSCCR<XEN2> = "0"　: 内部発振停止

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 φT0 は、CGSYSCR<FPSEL>から選択されたクロック fperiph 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 φT0 は fperiph/1 が選択されます。

注) クロックギアを使用する場合、周辺機能の各ブロックのプリスケーラ出力 φTn は、φTn < fsys を満足するように時間設定(φTn が fsys 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機能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。

6.4　動作モードとモード遷移

6.4.1　動作モード状態遷移

動作モードとしてシステムクロックに高速クロックを使用する NORMAL モードがあります。

また、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、STOP モードがあります。

図 6-2 にモード状態遷移図を示します。

WFI 命令、Sleep-on-exit については、"ARM ドキュメンテーションセット"を参照してください。

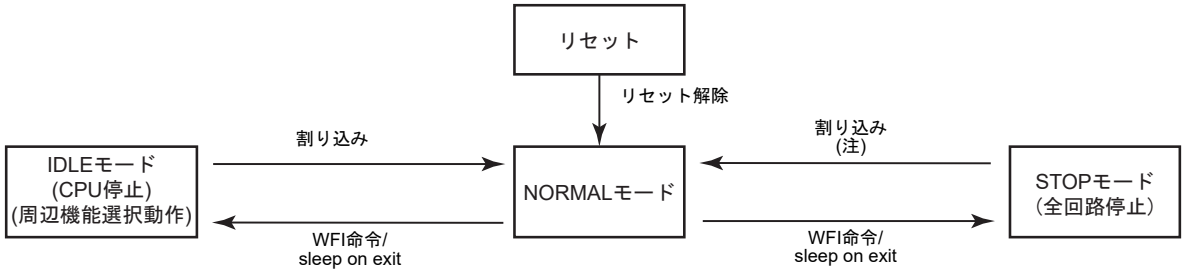


図 6-2 動作モード状態遷移図

注)　復帰時にウォーミングアップが必要となります。ウォーミングアップ時間の設定は STOP モードに入る前のモード(NORMAL)にて設定する必要があります。ウォーミングアップ時間に関しては、「6.6.6 ウォーミングアップ」を参照してください。

6.5　動作モード

6.5.1　NORMAL モード

CPU コアおよび周辺回路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。

6.6 低消費電力モード

低消費電力モードには、IDLE, 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-M0 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。

注 3) ウォーミングアップタイマ動作中に、低消費電力モードへ遷移しないでください。

IDLE, STOP モードの特長は次のとおりです。

6.6.1 IDLE モード

CPU のみが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)
- ベクトルエンジン(A-VE)

6.6.2　STOP モード

一部の回路を除き、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内蔵発振器が発振を開始し、NORMAL モードへ復帰します。

STOP モード中は、ポートレジスタの設定により端子の状態を保持することができます。STOP モード時の端子状態を表 6-2 に示します。

表 6-2 STOP モード時の端子状態

機能設定	端子名	入出力	STOP モード時(注)
制御端子	$\overline{\text{RESET}}$, MODE	入力	イネーブル
発振器	X1	入力	ディゼーブル
	X2	出力	"High"レベル出力
ポート	PAx ~ PPx	入力	PxIE[m]設定による
		出力	PxCR[m]設定による
デバッグ機能	SWCLK, SWDIO $\overline{\text{TRST}}$, TCK, TMS, TDI	入力	PxIE[m]設定による
	SWDIO, SWV TDO, TRACECLK TRACEDATA0/1/2/3	出力	PxCR[m]設定かつデータ有効な時にイネーブル
割り込み機能	INT0 ~ INTF	入力	PxIE[m]設定による
PMD	UO0, VO0, WO0, XO0, YO0, ZO0 UO1, VO1, WO1, XO1, YO1, ZO1	出力	PxCR[m]設定による
上記以外の機能	上記以外の端子	入力	PxIE[m]設定による
	上記以外の端子	出力	PxCR[m]設定による

注)「x」は該当ポート番号、「m」は該当ビットを示します。

6.6.3 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 6-3 に<STBY[2:0]>の設定より選択されるモードを示します。

表 6-3 低消費電力モードと設定

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

注) **上記以外の設定は行わないでください。**

6.6.4 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 6-4 に示します。

表 6-4 各動作モードにおける動作状態

Block	NORMAL	IDLE	STOP
Processor core	o	-	-
µDMAC	o	o	-
I/O port	o	o	- (注 1)
PMD	o	o	-
A-ENC	o	o	-
OFD	o	o	Δ
ADC	o	o	-
A-VE	o	・	-
SIO/UART	o	・	-
I2C	o	・	-
TMRB	o	・	-
WDT	o	-	Δ
VLTD	o	o	o (注 2)
POR	o	o	o (注 2)
DNF	o	o	-
CG	o	o	-
PLL	o	o	Δ
高速発振器(fc)	o	o	-

o : 対象のモード中に動作が可能

・ : 対象のモード中に動作が可能(チャネル毎に動作/停止選択可能)

- :対象のモードに移行すると自動的に Block へのクロックが停止

Δ: 対象のモードに移行する前にソフトウエアにてBlock を停止してください

注 1) ポートレジスタの設定によります。

注 2) 動作停止しませんが、クロックは供給されていません。

6.6.5 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、リセットによって行うことができます。使用できる低消費電力モード解除ソースは、低消費電力モードにより決まります。詳細を表 6-5 に示します。

表 6-5 解除ソースと解除可能なモード

低消費電力モード			IDLE	STOP
解除ソース	割り込み	INT0 ~ INTF (注 1)	o	o
		INTENC00, INTENC01 INTENC10, INTENC11	o	×
		INTVCN0, INTVCT0 INTVCN1, INTVCT1	o	×
		INTPWM0, INTEMG0, INTOVV0 INTPWM1, INTEMG1, INTOVV1	o	×
		INTRX0 ~ 3, INTTX0 ~ 3	o	×
		INTADACPA, INTADBCPA INTADACPB, INTADBCPB INTADATMR, INTADBTRM INTADASFT, INTADBSFT INTADAPDA, INTADBPDA INTADAPDB, INTADBPDB	o	×
		INTTB00 ~ INTTB90 INTCAP00 ~ INTCA90 INTTB01 ~ INTTB91 INTCAP01 ~ INTCAP91	o	×
		INTDMACATC, INTDMACAERR	o	×
	SysTick 割り込み		o	×
	リセット(OFD)		o	×
	リセット ($\overline{\text{RESET}}$ 端子, POR, VLTD)		o	o

o : 解除後、割り込み処理を開始します($\overline{\text{RESET}}$ は本製品を初期化します)。

× : 解除に使用できません。

注 1) IDLE、STOP モードから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。

注 3) 各モードからの復帰に必要なウォーミングアップについては「6.6.6 ウォーミングアップ」を参照してください。

・ 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。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、またデジタルノイズフィルタ回路を禁止にする必要があります。

・ リセットによる解除

$\overline{\text{RESET}}$ 端子、POR、VLTD 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。

OFD によるリセットでは、IDLE 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。

・ SysTick 割り込みによる解除

SysTick 割り込みは IDLE モードでのみ使用可能です。

割り込みの詳細に関しては、「例外」の章の「割り込み」をご参照ください。

6.6.6 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 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 モードに移行する命令を実行する前に CGOSCCR<WUPSEL1><WUPSEL2>でウォーミングアップする発振器の選択(注 1)および、<WUODR>でウォーミングアップ時間の設定(注 2)を行ってください。

注 1) TMPM470FDFG/FZFG/FYFG では、常に CGOSCCR<WUPSEL1>=0 を設定してください。

注 2) STOP モードでは PLL がディセーブルになるため、STOP モード解除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 μs です。

注 3) 自動ウォーミングアップする低消費電力モードからの復帰では、CGOSCCR<WUEON>の設定をする必要はありません。

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 6-6 に示します。

表 6-6 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 → STOP	不要
IDLE → NORMAL	不要
STOP → NORMAL	自動ウォーミングアップ

注) 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いません。発振器が安定するまでリセットを入力してください。

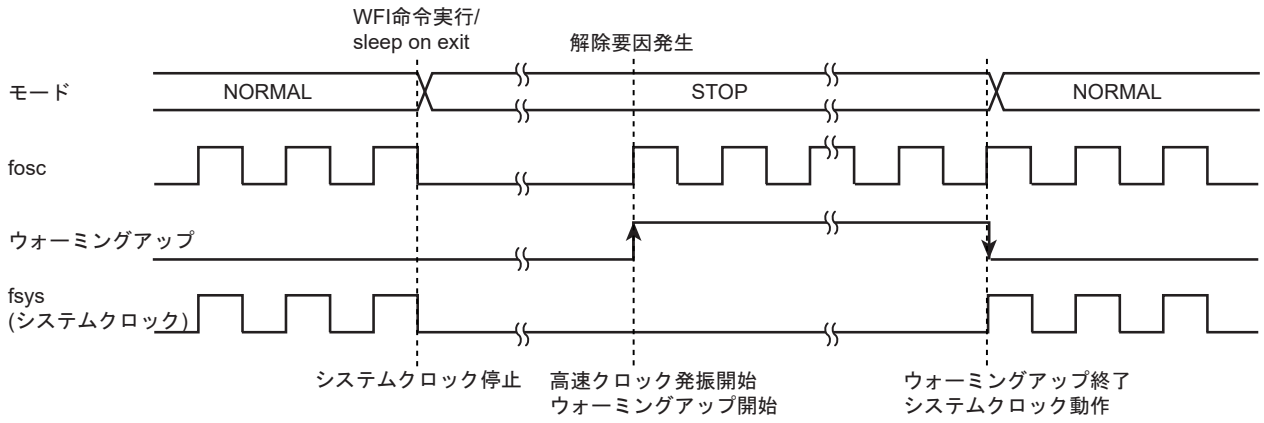
6.6.7　モード遷移によるクロック動作

モード遷移の際の、クロック動作について示します。

6.6.7.1　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。コールドリセット時と同じリセットを入力してください。



第 7 章　内蔵高速発振調整機能

TMPM470FDFG/FZFG/FYFG には、内蔵高速発振の周波数を調整する機能があります。

7.1　構成

内蔵高速発振調整機能は、16 ビットタイマ/ イベントカウンタ(TMRB) のパルス幅測定機能を使用して周波数の調整を行います。

図 7-1 に機能ブロック図を示します。

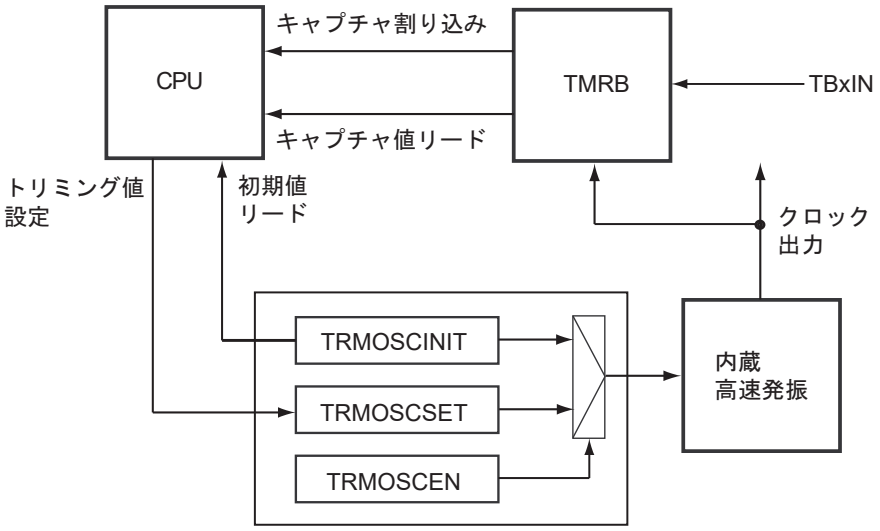


図 7-1 機能ブロック図

7.2　レジスタ説明

7.2.1　レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名(x=0~7)		Address(Base+)
プロテクトレジスタ	TRMOSCxPRO	0x0000
イネーブルレジスタ	TRMOSCxEN	0x0004
初期トリミング値モニタレジスタ	TRMOSCxINIT	0x0008
トリミング値設定レジスタ	TRMOSCxSET	0x000C

7.2.2　TRMOSCxPRO (プロテクトレジスタ)

	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	PROTECT							
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	PROTECT[7:0]	R/w	レジスタ書き込み制御 0xC1 : 許可 0xC1 以外 : 禁止 "0xC1" を設定すると、TRMOSCxEN, TRMOSCxINIT, TRMOSCxSET に書き込みができるようになります。

7.2.3 TRMOSCxEN (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	-	-	-	-	-	-	-	TRIMEN
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	TRIMEN	R/W	トリミング制御 0 : 禁止 1 : 許可 "1" を設定すると、内蔵発振器のトリミング値が、TRIMOSCxINIT で読み出される値から TRMOSCxSET に設定した値に切り替わります。

7.2.4　TRMOSCxINIT (初期トリミング値モニタレジスタ)

	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	-	-	TRIMINITC					
リセット後	0	0	不定					
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	TRIMINITF			
リセット後	0	0	0	0	不定			

Bit	Bit Symbol	Type	機能
31-14	−	R	リードすると"0"が読めます。
13-8	TRIMINITC [5:0]	R	初期粗トリミング値 出荷時の粗トリミング値が読めます。
7-4	−	R	リードすると"0"が読めます。
3-0	TRIMINITF[3:0]	R	初期微トリミング値 出荷時の微トリミング値が読めます。

粗トリミング、微トリミングの具体的な設定と調整値については、「7.3.2 調整範囲」を参照してください。

7.2.5 TRMOSCxSET (トリミング値設定レジスタ)

	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	-	-	TRIMSETC					
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	TRIMSETF			
リセット後	0	0	0	0	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。
13-8	TRIMSETC [5:0]	R/W	粗トリミング値設定 粗トリミング値を設定します。
7-4	−	R	リードすると"0" が読めます。
3-0	TRIMSETF[3:0]	R/W	微トリミング値設定 微トリミング値を設定します。

粗トリミング、微トリミングの具体的な設定と調整値については、「7.3.2 調整範囲」を参照してください。

7.3　動作説明

7.3.1　概要

周波数の調整は、粗トリミング値と微トリミング値で行います。

出荷時の設定値は、TRMOSCINIT<TRIMINITC> および<TRIMINITF> で確認できます。変更する設定値は、TRMOSCSET<TRIMSETC> および<TRIMSETF> に設定します。TRMOSCEN<TRIMEN> に"1" を設定することで内蔵高速発振器の設定値が切り替わります。

注)　リセット後、TRMOSCSET、TRMOSCEN への書き込みは禁止されています。書き込みを行うためには、TRMOSCPRO<PROTECT> に"0xC1" を設定する必要があります。

7.3.2　調整範囲

粗トリミングは平均 0.8% ステップで、およそ−19% ~ +32% の調整が可能です。

微トリミングは 0.1%ステップで、−0.8% ~ +0.7%の調整が可能です。

表 7-1 に調整範囲を示します。

注 1)　1 ステップの値は typ.条件のものであり、粗トリミングでは±0.6%、微トリミングでは±0.1%程度の誤差があります。
注 2)　各トリミングの分解能は、サンプル/条件により若干のバラつきを持ちます。

表 7-1 調整範囲

粗トリミング	
<TRIMSETC[5:0]>	周波数変化 (typ.)
011111	+32.0%
011110	+30.6%
011101	+29.2%
011100	+27.8%
011011	+26.5%
・	・
000011	+2.3%
000010	+1.5%
000001	+0.8%
000000	±0%
111111	−0.7%
111110	−1.5%
・	・
100100	−16.9%
100011	−17.4%
100010	−17.9%
100001	−18.3%
100000	−18.8%

微トリミング	
<TRIMSETF[3:0]>	周波数変化 (typ.)
0111	+0.7%
・	・
0001	+0.1%
0000	±0%
1111	−0.1%
1110	−0.2%
・	・
1000	−0.8%

第 8 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"ARM ドキュメンテーションセット Cortex-M4F 用"もご覧ください。

8.1　概要

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

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

8.1.1　種類

例外には以下のようなものがあります。

それぞれの例外の詳細な内容は、"ARM ドキュメンテーションセットCortex-M4F 用"をご覧ください。

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

8.1.2　処理の流れ

例外／割り込みの処理の流れの概略を以下に示します。以下の説明で、 はハードウエアによる処理を、 はソフトウエ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。	8.1.2.1 節
CPU が例外を処理	CPU が例外処理を行います。	8.1.2.2 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。	8.1.2.3 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。	8.1.2.4 節

8.1.2.1 例外要求と検出

(1) 例外要求の発生

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

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

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

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。低消費電力モード解除に使用する割り込みの場合、クロックジェネレータの設定も必要になります。詳細は「8.5 割り込み」の節で説明します。

(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 8-1 例外の種類と優先度

番号	例外	優先度	要因
1	リセット	−3 (最高)	リセット端子, WDT, POR, VLTD, OFD, SYSRESETREQ
2	マスク不能割り込み	−2	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	SVCall	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、「8.5.1.5 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>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。

TMPM470FDFG/FZFG/FYFG では、<PRI_n>は 3 ビット構成になっています。

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。

・　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 8-2 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 8-2 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。

8.1.2.2 例外の処理と割り込み処理ルーチンへの分岐(横取り)

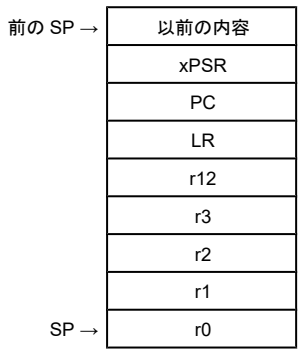
例外により、実行中の処理を中断して割り込み処理ルーチンへ分岐する動作を"横取り"と呼びます。

(1) レジスタの退避

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

1. プログラムカウンタ(PC)
2. プログラムステータスレジスタ(xPSR)
3. r0 ~ r3
4. r12
5. 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。

また、ベクタテーブルには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、マスク不能割り込み、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。

その他の例外については、割り込み処理ルーチンのアドレスを必要に応じて準備します。

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

8.1.2.3　割り込み処理ルーチンの発行

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「8.5 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。

8.1.2.4 例外からの復帰

(1) 割り込み処理ルーチンからの復帰先

割り込み処理ルーチン終了時の状態により復帰先が決まります。

- テールチェイン

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

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

- 処理が中断されている割り込み処理ルーチンへ復帰

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

- 元のプログラムへ復帰

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

(2) 復帰処理

CPU は、割り込み処理ルーチンから復帰する際に以下の処理を行います。

- レジスタの復帰

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

- 割り込み番号のロード

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

- SP の選択

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

8.2　リセット例外

リセット例外には、以下の要因があります。

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。

- 外部リセット端子によるリセット例外

  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。

- POR によるリセット例外

  POR にリセット例外を発生する機能があります。詳細は「パワーオンリセット回路」の章をご覧ください。

- VLTD によるリセット例外

  VLTD にリセットを発生する機能があります。詳細は「電源検出回路」の章をご覧ください。

- OFD によるリセット例外

  OFD にリセットを発生する機能があります。詳細は「周波数検知回路」の章をご覧ください。

- WDT によるリセット例外

  WDT にリセット例外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>をセットすることで、リセット例外を発生させることができます。

8.3 マスク不能割り込み(NMI)

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。

8.4 SysTick

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

SysTick リロード値レジスタに値を設定し、SysTick 制御およびステータスレジスタで機能をイネーブルにすると、リロード値レジスタに設定された値がカウンタへリロードされカウントダウンを開始します。カウンタが"0"になると SysTick 例外を発生します。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。

注) 本製品では外部参照クロックとして fosc(CGOSCCR<OSCSEL>で選択されるクロック)を 32 分周したクロックが使用されます。

8.5　割り込み

この節では、割り込み要求の伝わる経路、要因、 必要な設定について説明します。

割り込みは、割り込み要因ごとに割り込み要求信号により 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求は、クロックジェネレータを経由して CPU に要因が伝わるため、クロックジェネレータの設定も必要です。

8.5.1　要因

8.5.1.1　経路

割り込み要求の経路を図 8-1 に示します。

周辺機能からの割り込み要求のうち、低消費電力モード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要求(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)

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

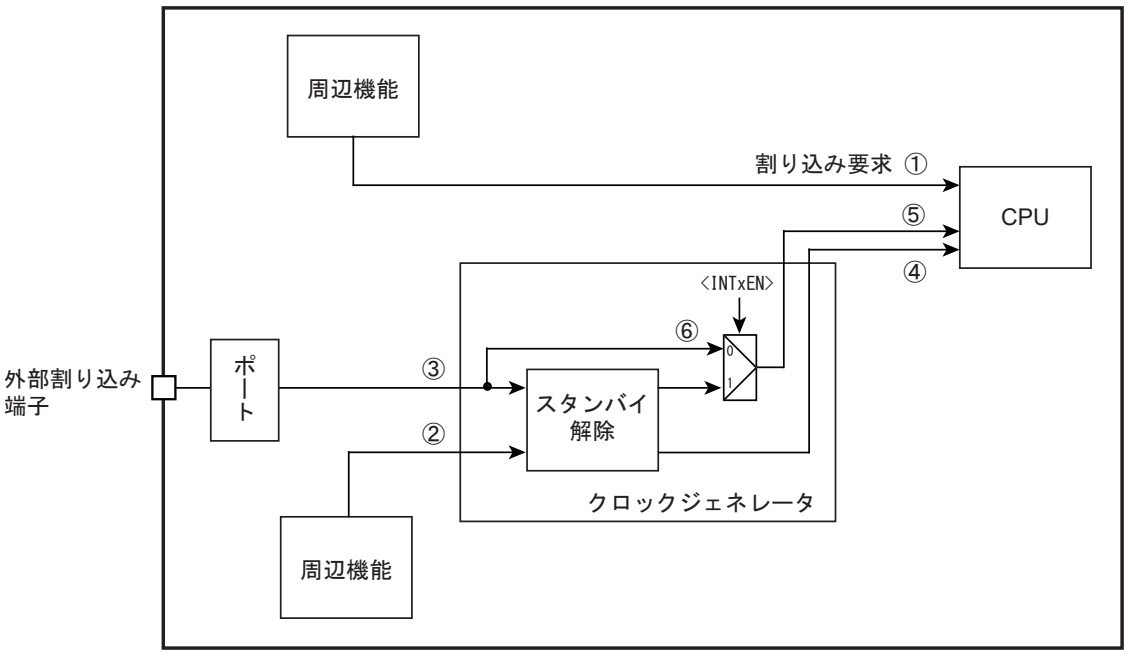


図 8-1 割り込み要因の経路

8.5.1.2 割り込み要求の発生

割り込み要求は、割り込み要因に割り当てられた外部割り込み端子、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。

- 外部割り込み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要求が発生されるよう設定する必要があります。

  設定の詳細については各章を参照ください。
- 割り込み要求の強制的な発生

  NVIC の割り込み保留セットレジスタの該当するビットをセットすることで、割り込み要求を強制的に発生させることができます。

CPU は、割り込み要求の"High"レベルを割り込みとして認識します。

8.5.1.3 割り込み要因の伝達

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

スタンバイ解除要因として使用できる割り込みは、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。

8.5.1.4 外部割り込み端子を使用する際の注意

外部割り込みを使用際には、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に入力信号は、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 8-1 割り込みの経路」の ⑥ の経路)、外部割り込み端子からの入力信号がそのまま CPU に伝わります。CPU は"High"入力を割り込みとして認識しますので、入力ディセーブルのまま CPU で該当する割り込みを有効にすると割り込みが発生します。

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

8.5.1.5　要因一覧

割り込みの要因一覧を表 8-3 に示します。

表 8-3 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	外部割込み端子 0	[H]/[L]レベルと [↑]/[↓]エッジ選択可	CGIMCGA
1	INT1	外部割込み端子 1		
2	INT2	外部割込み端子 2		
3	INT3	外部割込み端子 3		
4	INT4	外部割込み端子 4	[H]/[L]レベルと [↑]/[↓]エッジ選択可	CGIMCGB
5	INT5	外部割込み端子 5		
6	INTRX0	シリアルチャネル 0 受信割り込み		
7	INTTX0	シリアルチャネル 0 送信割り込み		
8	INTRX1	シリアルチャネル 1 受信割り込み		
9	INTTX1	シリアルチャネル 1 送信割り込み		
10	INTVCN0	ベクトルエンジン割り込み 0		
11	INTVCN1	ベクトルエンジン割り込み 1		
12	INTEMG0	PMD0 EMG 割り込み		
13	INTEMG1	PMD1 EMG 割り込み		
14	INTOVV0	PMD0 OVV 割り込み		
15	INTOVV1	PMD1 OVV 割り込み		
16	INTADAPDA	ADCA PMD トリガ同期変換終了 ADCA で INTADxPDA を選択した場合の割込み		
17	INTADBPDA	ADCB PMD トリガ同期変換終了 ADCB で INTADxPDA を選択した場合の割込み		
18	INTADAPDB	ADCA PMD トリガ同期変換終了 ADCA で INTADxPDB を選択した場合の割込み		
19	INTADBPDB	ADCB PMD トリガ同期変換終了 ADCB で INTADxPDB を選択した場合の割込み		
20	INTTB00	16bit TMRB0 コンペアー致 0/オーバフロー		
21	INTTB01	16bit TMRB0 コンペアー致 1		
22	INTTB10	16bit TMRB1 コンペアー致 0/オーバフロー		
23	INTTB11	16bit TMRB1 コンペアー致 1		
24	INTTB40	16bit TMRB4 コンペアー致 0/オーバフロー		
25	INTTB41	16bit TMRB4 コンペアー致 1		
26	INTTB50	16bit TMRB5 コンペアー致 0/オーバフロー		
27	INTTB51	16bit TMRB5 コンペアー致 1		
28	INTPMD0	PMD0 割り込み		
29	INTPMD1	PMD1 割り込み		
30	INTCAP00	16bit TMRB0 インプットキャプチャ 0		
31	INTCAP01	16bit TMRB0 インプットキャプチャ 1		
32	INTCAP10	16bit TMRB1 インプットキャプチャ 0		
33	INTCAP11	16bit TMRB1 インプットキャプチャ 1		
34	INTCAP40	16bit TMRB4 インプットキャプチャ 0		
35	INTCAP41	16bit TMRB4 インプットキャプチャ 1		
36	INTCAP50	16bit TMRB5 インプットキャプチャ 0		
37	INTCAP51	16bit TMRB5 インプットキャプチャ 1		
38	INT6	外部割込み端子 6	[H]/[L]レベルと [↑]/[↓]エッジ選択可	CGIMCGB
39	INT7	外部割込み端子 7		
40	INTRX2	シリアルチャネル 2 受信割り込み		

表 8-3 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
41	INTTX2	シリアルチャネル 2 送信割り込み		
42	INTADACPA	ADCA 監視割り込み A		
43	INTADBCPA	ADCB 監視割り込み A		
44	INTADACPB	ADCA 監視割り込み B		
45	INTADBCPB	ADCB 監視割り込み B		
46	INTTB20	16bit TMRB2 コンペアー致 0/オーバフロー		
47	INTTB21	16bit TMRB2 コンペアー致 1		
48	INTTB30	16bit TMRB3 コンペアー致 0/オーバフロー		
49	INTTB31	16bit TMRB3 コンペアー致 1		
50	INTCAP20	16bit TMRB2 インプットキャプチャ 0		
51	INTCAP21	16bit TMRB2 インプットキャプチャ 1		
52	INTCAP30	16bit TMRB3 インプットキャプチャ 0		
53	INTCAP31	16bit TMRB3 インプットキャプチャ 1		
54	INTADASFT	ADCA ソフトスタート変換終了		
55	INTADBSFT	ADCB ソフトスタート変換終了		
56	INTADATMR	ADCA タイマ同期変換終了		
57	INTADBTMR	ADCB タイマ同期変換終了		
58	INT8	外部割込み端子 8	[H]/[L]レベルと [↑]/[↓]エッジ選択可	CGIMCGC
59	INT9	外部割込み端子 9		
60	INTA	外部割込み端子 A		
61	INTB	外部割込み端子 B		
62	INTENC00	エンコーダ 0 割り込み 0		
63	INTENC01	エンコーダ 0 割り込み 1		
64	INTRX3	シリアルチャネル 3 受信割り込み		
65	INTTX3	シリアルチャネル 3 送信割り込み		
66	INTTB60	16bit TMRB6 コンペアー致 0/オーバフロー		
67	INTTB61	16bit TMRB6 コンペアー致 1		
68	INTTB70	16bit TMRB7 コンペアー致 0/オーバフロー		
69	INTTB71	16bit TMRB7 コンペアー致 1		
70	INTCAP60	16bit TMRB6 インプットキャプチャ 0		
71	INTCAP61	16bit TMRB6 インプットキャプチャ 1		
72	INTCAP70	16bit TMRB7 インプットキャプチャ 0		
73	INTCAP71	16bit TMRB7 インプットキャプチャ 1		
74	INTC	外部割込み端子 C	[H]/[L]レベルと [↑]/[↓]エッジ選択可	CGIMCGD
75	INTD	外部割込み端子 D	[H]/[L]レベルと [↑]/[↓]エッジ選択可	CGIMCGD
76	INTE	外部割込み端子 E		
77	INTF	外部割込み端子 F		
78	INTVCT0	ベクトルエンジンタスク終了割り込み 0		
79	INTVCT1	ベクトルエンジンタスク終了割り込み 1		
80	INTSBI0	I2C 割り込み		
81	–	–		
82	–	–		
83	–	–		
84	INTTB80	16bit TMRB8 コンペアー致 0/オーバフロー		
85	INTTB81	16bit TMRB8 コンペアー致 1		
86	INTTB90	16bit TMRB9 コンペアー致 0/オーバフロー		

表 8-3 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
87	INTTB91	16bit TMRB9 コンペア一致 1		
88	INTCAP80	16bit TMRB8 インプットキャプチャ 0		
89	INTCAP81	16bit TMRB8 インプットキャプチャ 1		
90	INTCAP90	16bit TMRB9 インプットキャプチャ 0		
91	INTCAP91	16bit TMRB9 インプットキャプチャ 1		
92	INTDMACATC	DMA 転送終了		
93	INTDMACAERR	DMA エラー割り込み		
94	INTENC10	エンコーダ 1 割り込み 0		
95	INTENC11	エンコーダ 1 割り込み 1		

8.5.1.6　アクティブレベル

アクティブレベルはどのような信号変化を割り込み要因とみ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<INT x EN>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 8-3 で指定されているとおりに設定してください。

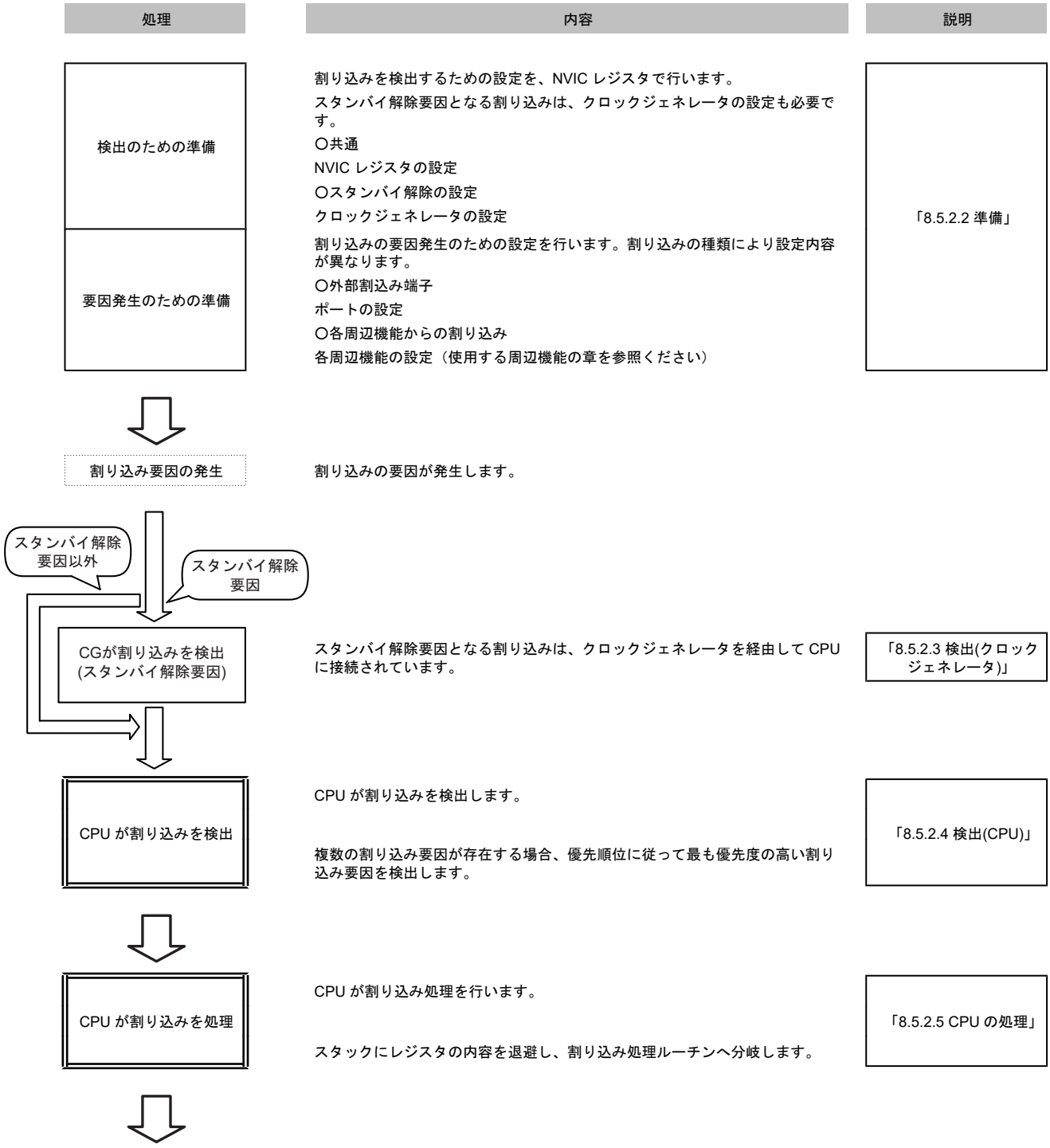
クロックジェネレータで検出された割り込みは、"High"レベル信号で CPU に通知されます。

8.5.2 処理詳細

8.5.2.1 処理の流れ

割り込みの処理の流れを以下に示します。

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



処理	内容	説明
割り込み サービスルーチン実行	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。	「8.5.2.6 割り込み処理ルーチンでの処理(要因の取り下げ)」
⇩		
元のプログラムへ復帰	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。	

8.5.2.2 準備

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

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

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

以下に設定の手順と、手順ごとの具体的な設定方法を示します。

1. CPU 割り込み禁止
2. CPU 割り込み設定
3. 要因の準備(1) (外部割込み端子)
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) (外部割込み端子)

外部割込み端子を使用する場合、該当する端子のポートの設定を行います。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		
PxIE<PxmIE>	←	"1"

注) 「x」は該当ポート番号、「m」は該当ビットを示します。
PxIE で入力イネーブル設定であれば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、「8.5.1.4 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		
<SETPEND[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、「8.6.3.2 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、「8.5.1.4 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		
<CLRPEND[m]>	←	"1"
<SETENA[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。

8.5.2.3 検出(クロックジェネレータ)

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

アクティブレベルが立ち上がりまたは立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ったとみなされるため、割り込み検出までレベルを保つ必要があります。

クロックジェネレータは割り込みを検出すると CG 割り込み要求クリアレジスタ(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。

8.5.2.4 検出(CPU)

CPU は優先順位に従って最も優先度の高い割り込み要因を検出します。

8.5.2.5　CPU の処理

割り込みが検出されると、CPU はスタックへ xPSR、PC、LR、R12、r3～r0 を退避し、検出した割り込みの割り込み処理ルーチンへ分岐します。

8.5.2.6　割り込み処理ルーチンでの処理(要因の取り下げ)

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

(1)　割り込み処理ルーチンでの処理

通常、割り込み処理ルーチンでは必要なレジスタの退避と割り込み処理を行います。Cortex-M4F コアは自動的に xPSR、PC、LR、R12、r3～r0 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。

8.6 例外/割り込み関連レジスタ

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

8.6.1 レジスター覧

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 較正値レジスタ	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セットレジスタ 3	0x0108
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
割り込みイネーブルクリアレジスタ 3	0x0188
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
割り込み保留セットレジスタ 3	0x0208
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
割り込み保留クリアレジスタ 3	0x0288
割り込み優先度レジスタ	0x0400 ~ 0x047F
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

周辺機能名：CG

レジスタ名		Address
CG 割り込み要求クリアレジスタ	CGICRCG	0x0014
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

8.6.2　NVIC レジスタ

8.6.2.1　SysTick 制御およびステータスレジスタ

	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(fosc/32) (注) 1: CPU クロック(fsys)
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。

注)　本製品では外部参照クロックとして fosc(CGOSCCR<OSCSEL>で選択されるクロック)を 32 分周したクロックが使用されます。

8.6.2.2 SysTick リロード値レジスタ

	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。

8.6.2.3 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。

8.6.2.4　SysTick 較正値レジスタ

	31	30	29	28	27	26	25	24
bit symbol	NOREF	SKEW	-	-	-	-	-	-
リセット後	0	1	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	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TENMS							
リセット後	0	0	0	0	0	0	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: 較正値は 10 ms 1: 較正値は 10 ms でない
29-24	−	R	リードすると"0"が読めます。
23-0	TENMS	R	較正値(注)

注)　本製品では較正用の値は準備していません。

8.6.2.5 割り込み制御用レジスタ

それぞれの割り込み要因について、割り込みイネーブルセットレジスタ、割り込みイネーブルクリアレジスタ、割り込み保留セットレジスタ、割り込みクリアレジスタがあります。

各ビットが指定された割り込みに対応しています。

(1) 割り込みイネーブルセットレジスタ

割り込みを許可したり、割り込みの許可/禁止状態が確認できます。

"1"をライトすることで該当する割り込みを許可します。

"0"の書き込みは意味を持ちません。

リードすると該当する割り込みの許可/禁止状態が確認できます。

このレジスタのビットをクリアするには、割り込みイネーブルクリアレジスタの対応するビットに"1"をセットします。

Bit symbol	Type	機能
SETENA	R/W	割り込み番号[95:0] [ライト] 1: 割り込みを許可状態にする [リード] 0: 割り込みが禁止状態 1: 割り込みが許可状態

(a) 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

注) 割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(b)　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	SETENA (割り込み 63)	SETENA (割り込み 62)	SETENA (割り込み 61)	SETENA (割り込み 60)	SETENA (割り込み 59)	SETENA (割り込み 58)	SETENA (割り込み 57)	SETENA (割り込み 56)
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 55)	SETENA (割り込み 54)	SETENA (割り込み 53)	SETENA (割り込み 52)	SETENA (割り込み 51)	SETENA (割り込み 50)	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

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(c)　割り込みイネーブルセットレジスタ 3

	31	30	29	28	27	26	25	24
bit symbol	SETENA (割り込み 95)	SETENA (割り込み 94)	SETENA (割り込み 93)	SETENA (割り込み 92)	SETENA (割り込み 91)	SETENA (割り込み 90)	SETENA (割り込み 89)	SETENA (割り込み 88)
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 87)	SETENA (割り込み 86)	SETENA (割り込み 85)	SETENA (割り込み 84)	–	–	–	SETENA (割り込み 80)
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 79)	SETENA (割り込み 78)	SETENA (割り込み 77)	SETENA (割り込み 76)	SETENA (割り込み 75)	SETENA (割り込み 74)	SETENA (割り込み 73)	SETENA (割り込み 72)
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 71)	SETENA (割り込み 70)	SETENA (割り込み 69)	SETENA (割り込み 68)	SETENA (割り込み 67)	SETENA (割り込み 66)	SETENA (割り込み 65)	SETENA (割り込み 64)
リセット後	0	0	0	0	0	0	0	0

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(2) 割り込みイネーブルクリアレジスタ

割り込みを禁止したり、割り込みの許可/禁止状態が確認できます。

"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。

リードすると該当する割り込みの許可/禁止状態が確認できます。

Bit symbol	Type	機能
CLRENA	R/W	割り込み番号[95:0] [ライト] 1: 割り込みを禁止状態にする [リード] 0: 割り込みが禁止状態 1: 割り込みが許可状態

(a) 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 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

注) 割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(b)　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	CLRENA (割り込み 63)	CLRENA (割り込み 62)	CLRENA (割り込み 61)	CLRENA (割り込み 60)	CLRENA (割り込み 59)	CLRENA (割り込み 58)	CLRENA (割り込み 57)	CLRENA (割り込み 56)
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 55)	CLRENA (割り込み 54)	CLRENA (割り込み 53)	CLRENA (割り込み 52)	CLRENA (割り込み 51)	CLRENA (割り込み 50)	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

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(c)　割り込みイネーブルクリアレジスタ 3

	31	30	29	28	27	26	25	24
bit symbol	CLRENA (割り込み 95)	CLRENA (割り込み 94)	CLRENA (割り込み 93)	CLRENA (割り込み 92)	CLRENA (割り込み 91)	CLRENA (割り込み 90)	CLRENA (割り込み 89)	CLRENA (割り込み 88)
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 87)	CLRENA (割り込み 86)	CLRENA (割り込み 85)	CLRENA (割り込み 84)	–	–	–	CLRENA (割り込み 80)
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 79)	CLRENA (割り込み 78)	CLRENA (割り込み 77)	CLRENA (割り込み 76)	CLRENA (割り込み 75)	CLRENA (割り込み 74)	CLRENA (割り込み 73)	CLRENA (割り込み 72)
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 71)	CLRENA (割り込み 70)	CLRENA (割り込み 69)	CLRENA (割り込み 68)	CLRENA (割り込み 67)	CLRENA (割り込み 66)	CLRENA (割り込み 65)	CLRENA (割り込み 64)
リセット後	0	0	0	0	0	0	0	0

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(3) 割り込み保留セットレジスタ

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。

Bit symbol	Type	機能
SETPEND	R/W	割り込み番号[95:0] [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり

(a) 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 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

注) 割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(b)　割り込み保留セットレジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	SETPEND (割り込み 63)	SETPEND (割り込み 62)	SETPEND (割り込み 61)	SETPEND (割り込み 60)	SETPEND (割り込み 59)	SETPEND (割り込み 58)	SETPEND (割り込み 57)	SETPEND (割り込み 56)
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 55)	SETPEND (割り込み 54)	SETPEND (割り込み 53)	SETPEND (割り込み 52)	SETPEND (割り込み 51)	SETPEND (割り込み 50)	SETPEND (割り込み 49)	SETPEND (割り込み 48)
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 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

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(c)　割り込み保留セットレジスタ 3

	31	30	29	28	27	26	25	24
bit symbol	SETPEND (割り込み 95)	SETPEND (割り込み 94)	SETPEND (割り込み 93)	SETPEND (割り込み 92)	SETPEND (割り込み 91)	SETPEND (割り込み 90)	SETPEND (割り込み 89)	SETPEND (割り込み 88)
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 87)	SETPEND (割り込み 86)	SETPEND (割り込み 85)	SETPEND (割り込み 84)	–	–	–	SETPEND (割り込み 80)
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 79)	SETPEND (割り込み 78)	SETPEND (割り込み 77)	SETPEND (割り込み 76)	SETPEND (割り込み 75)	SETPEND (割り込み 74)	SETPEND (割り込み 73)	SETPEND (割り込み 72)
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 71)	SETPEND (割り込み 70)	SETPEND (割り込み 69)	SETPEND (割り込み 68)	SETPEND (割り込み 67)	SETPEND (割り込み 66)	SETPEND (割り込み 65)	SETPEND (割り込み 64)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(4) 割り込み保留クリアレジスタ

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。

Bit symbol	Type	機能
CLRPEND	R/W	割り込み番号[95:0] [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

(a) 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 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

注) 割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(b)　割り込み保留クリアレジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	CLRPEND (割り込み 63)	CLRPEND (割り込み 62)	CLRPEND (割り込み 61)	CLRPEND (割り込み 60)	CLRPEND (割り込み 59)	CLRPEND (割り込み 58)	CLRPEND (割り込み 57)	CLRPEND (割り込み 56)
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 55)	CLRPEND (割り込み 54)	CLRPEND (割り込み 53)	CLRPEND (割り込み 52)	CLRPEND (割り込み 51)	CLRPEND (割り込み 50)	CLRPEND (割り込み 49)	CLRPEND (割り込み 48)
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 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

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

(c)　割り込み保留クリアレジスタ 3

	31	30	29	28	27	26	25	24
bit symbol	CLRPEND (割り込み 95)	CLRPEND (割り込み 94)	CLRPEND (割り込み 93)	CLRPEND (割り込み 92)	CLRPEND (割り込み 91)	CLRPEND (割り込み 90)	CLRPEND (割り込み 89)	CLRPEND (割り込み 88)
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 87)	CLRPEND (割り込み 86)	CLRPEND (割り込み 85)	CLRPEND (割り込み 84)	–	–	–	CLRPEND (割り込み 80)
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 79)	CLRPEND (割り込み 78)	CLRPEND (割り込み 77)	CLRPEND (割り込み 76)	CLRPEND (割り込み 75)	CLRPEND (割り込み 74)	CLRPEND (割り込み 73)	CLRPEND (割り込み 72)
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 71)	CLRPEND (割り込み 70)	CLRPEND (割り込み 69)	CLRPEND (割り込み 68)	CLRPEND (割り込み 67)	CLRPEND (割り込み 66)	CLRPEND (割り込み 65)	CLRPEND (割り込み 64)
リセット後	不定	不定	不定	不定	不定	不定	不定	不定

注)　割り込みの内容と割り込み番号については、「8.5.1.5 要因一覧」を参照してください。

8.6.2.6 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_51	PRI_50	PRI_49	PRI_48
0xE000_E434	PRI_55	PRI_54	PRI_53	PRI_52
0xE000_E438	PRI_59	PRI_58	PRI_57	PRI_56
0xE000_E43C	PRI_63	PRI_62	PRI_61	PRI_60
0xE000_E440	PRI_67	PRI_66	PRI_65	PRI_64
0xE000_E444	PRI_71	PRI_70	PRI_69	PRI_68
0xE000_E448	PRI_75	PRI_74	PRI_73	PRI_72
0xE000_E44C	PRI_79	PRI_78	PRI_77	PRI_76
0xE000_E450	–	–	–	PRI_80
0xE000_E454	PRI_87	PRI_86	PRI_85	PRI_84
0xE000_E458	PRI_91	PRI_90	PRI_89	PRI_88
0xE000_E45C	PRI_95	PRI_94	PRI_93	PRI_92

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。

8.6.2.7 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	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
31-7	TBLOFF	R/W	オフセット値 アドレス 0x0000_0000 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。

8.6.2.8　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がデフォルトで選択されます。

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。

8.6.2.9 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。

8.6.2.10　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。

8.6.3　クロックジェネレータレジスタ

8.6.3.1　CG 割り込みモードコントロールレジスタ

CG 割り込みモードコントロールレジスタは低消費電力モード解除に使用する割り込み要因のアクティブレベル設定、検出されたアクティブレベル、低消費電力モード解除の許可/禁止をおこないます。

Bit symbol	Type	機能
EMCGx[2:0]	R/W	低消費電力モード解除のアクティブレベルを設定 (下記のアクティブレベルのうち表 8-4 のアクティブレベルに記載されている解除要因から選択する) 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 上記以外:設定禁止
EMSTx[1:0]	R	検出されたアクティブレベル(EMCGx[2:0]="100"の場合のみ有効) 00: − 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
INTxEN	R/W	低消費電力モード解除 0:禁止 1: 許可

表 8-4 低消費電力モード解除のアクティブレベル設定

要因		アクティブレベル設定レジスタ	低消費電力モード解除のアクティブレベル				
			"Low" レベル	"HIGH" レベル	立ち上がりエッジ	立ち下がりエッジ	両エッジ
INT0	外部割込み端子 0	CGIMCGA <EMCG00[2:0]>	o	o	o	o	o
INT1	外部割込み端子 1	CGIMCGA <EMCG01[2:0]>	o	o	o	o	o
INT2	外部割込み端子 2	CGIMCGA <EMCG02[2:0]>	o	o	o	o	o
INT3	外部割込み端子 3	CGIMCGA <EMCG03[2:0]>	o	o	o	o	o
INT4	外部割込み端子 4	CGIMCGB <EMCG04[2:0]>	o	o	o	o	o
INT5	外部割込み端子 5	CGIMCGB <EMCG05[2:0]>	o	o	o	o	o
INT6	外部割込み端子 6	CGIMCGB <EMCG06[2:0]>	o	o	o	o	o
INT7	外部割込み端子 7	CGIMCGB <EMCG07[2:0]>	o	o	o	o	o
INT8	外部割込み端子 8	CGIMCGC <EMCG08[2:0]>	o	o	o	o	o
INT9	外部割込み端子 9	CGIMCGC <EMCG09[2:0]>	o	o	o	o	o
INTA	外部割込み端子 A	CGIMCGC <EMCG0A[2:0]>	o	o	o	o	o
INTB	外部割込み端子 B	CGIMCGC <EMCG0B[2:0]>	o	o	o	o	o
INTC	外部割込み端子 C	CGIMCGD <EMCG0C[2:0]>	o	o	o	o	o
INTD	外部割込み端子 D	CGIMCGD <EMCG0D[2:0]>	o	o	o	o	o
INTE	外部割込み端子 E	CGIMCGD <EMCG0E[2:0]>	o	o	o	o	o
INTF	外部割込み端子 F	CGIMCGD <EMCG0F[2:0]>	o	o	o	o	o

注) 低消費電力モード解除のアクティブレベルは"o"のついたものが選択できます。"×"のついたものは選択できません。

(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	-	EMCG03			EMST03		-	INT03EN
リセット後	0	0	1	0	0	0	不定	0
	23	22	21	20	19	18	17	16
bit symbol	-	EMCG02			EMST02		-	INT02EN
リセット後	0	0	1	0	0	0	不定	0
	15	14	13	12	11	10	9	8
bit symbol	-	EMCG01			EMST01		-	INT01EN
リセット後	0	0	1	0	0	0	不定	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCG00			EMST00		-	INT00EN
リセット後	0	0	1	0	0	0	不定	0

注 1)　<EMCGx[2:0]>に設定できるアクティブレベルは要因ごとにことなります。表 8-4 を参照の上、設定してください。

注 2)　<EMSTx>は、<EMCGx[2:0]>が"100"の両エッジ設定のときのみ有効です。それ以外のときは、不定となります。<EMSTx>を参照することにより、低消費電力モード解除を解除したアクティブレベル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 3)　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。

注 4)　ビット 31、23、15、7 は"0"が読み出されます。

注 5)　ビット 25、17、9、1 は不定が読み出されます。

(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	-	EMCG07			EMST07		-	INT07EN
リセット後	0	0	1	0	0	0	不定	0
	23	22	21	20	19	18	17	16
bit symbol	-	EMCG06			EMST06		-	INT06EN
リセット後	0	0	1	0	0	0	不定	0
	15	14	13	12	11	10	9	8
bit symbol	-	EMCG05			EMST05		-	INT05EN
リセット後	0	0	1	0	0	0	不定	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCG04			EMST04		-	INT04EN
リセット後	0	0	1	0	0	0	不定	0

注 1)　<EMCGx[2:0]>に設定できるアクティブレベルは要因ごとにことなります。表 8-4 を参照の上、設定してください。

注 2)　<EMSTx>は、<EMCGx[2:0]>が"100"の両エッジ設定のときのみ有効です。それ以外のときは、不定となります。<EMSTx>を参照することにより、低消費電力モード解除を解除したアクティブレベル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 3)　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。

注 4)　ビット 31、23、15、7 は"0"が読み出されます。

注 5)　ビット 25、17、9、1 は不定が読み出されます。

(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	-	EMCG0B			EMST0B		-	INT0BEN
リセット後	0	0	1	0	0	0	不定	0
	23	22	21	20	19	18	17	16
bit symbol	-	EMCG0A			EMST0A		-	INT0AEN
リセット後	0	0	1	0	0	0	不定	0
	15	14	13	12	11	10	9	8
bit symbol	-	EMCG09			EMST09		-	INT09EN
リセット後	0	0	1	0	0	0	不定	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCG08			EMST08		-	INT08EN
リセット後	0	0	1	0	0	0	不定	0

注 1) <EMCGx[2:0]>に設定できるアクティブレベルは要因ごとにことなります。表 8-4 を参照の上、設定してください。

注 2) <EMSTx>は、<EMCGx[2:0]>が"100"の両エッジ設定のときのみ有効です。それ以外のときは、不定となります。<EMSTx>を参照することにより、低消費電力モード解除を解除したアクティブレベル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 3) 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。

注 4) ビット 31、23、15、7 は"0"が読み出されます。

注 5) ビット 25、17、9、1 は不定が読み出されます。

(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	-	EMCG0F			EMST0F		-	INT0FEN
リセット後	0	0	1	0	0	0	不定	0
	23	22	21	20	19	18	17	16
bit symbol	-	EMCG0E			EMST0E		-	INT0EEN
リセット後	0	0	1	0	0	0	不定	0
	15	14	13	12	11	10	9	8
bit symbol	-	EMCG0D			EMST0D		-	INT0DEN
リセット後	0	0	1	0	0	0	不定	0
	7	6	5	4	3	2	1	0
bit symbol	-	EMCG0C			EMST0C		-	INT0CEN
リセット後	0	0	1	0	0	0	不定	0

注 1) <EMCGx[2:0]>に設定できるアクティブレベルは要因ごとにことなります。表 8-4 を参照の上、設定してください。

注 2) <EMSTx>は、<EMCGx[2:0]>が"100"の両エッジ設定のときのみ有効です。それ以外のときは、不定となります。<EMSTx>を参照することにより、低消費電力モード解除を解除したアクティブレベル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 3) 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。

注 4) ビット 31、23、15、7 は"0"が読み出されます。

注 5) ビット 25、17、9、1 は不定が読み出されます。

8.6.3.2　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_1000: INT8 0_0001: INT1　　0_1001: INT9 0_0010: INT2　　0_1010: INTA 0_0011: INT3　　0_1011: INTB 0_0100: INT4　　0_1100 : INTC 0_0101: INT5　　0_1101 : INTD 0_0110: INT6　　0_1110 : INTE 0_ 0111: INT7　　0_1111 : INTF 1_0000 to 1_1111: 設定禁止 リードすると"0"が読めます

8.6.3.3 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	-	-	OFDRSTF	DBGRSTF	VLTDRSTF	WDTRSTF	PINRSTF	PINRSTF
端子 リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-6	−	R	リードすると"0"が読めます。
5	OFDRSTF	R/W	OFD リセットフラグ 0: 0 ライト 1: OFD によるリセットフラグ
4	DBGRSTF	R/W	デバッグリセットフラグ(注 1) 0: 0 ライト 1: <SYSRESETREQ>によるリセットフラグ
3	VLTDRSTF	R/W	VLTD リセットフラグ 0: 0 ライト 1: VLTD によるリセットフラグ
2	WDTRSTF	R/W	WDT リセットフラグ 0: 0 ライト 1: WDT によるリセットフラグ
1	PINRSTF	R/W	$\overline{\text{RESET}}$ 端子フラグ 0: 0 ライト 1: $\overline{\text{RESET}}$ 端子によるリセットフラグ
0	PORRSTF	R/W	パワーオンフラグ(注 2) 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>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) 本製品はパワーオンリセット回路を内蔵しており、本レジスタはパワーオンリセットでのみ初期化されます。したがって、電源投入時の初期リセット後は<PONRSTF>がセットされますが、2 度目以降のリセットでは<PONRSTF>はセットされません。また、本レジスタは自動的にクリアされませんので"0"を書いてクリアしてください。

第 9 章　デジタルノイズフィルタ回路(DNF)

デジタル式のノイズキャンセラ回路により、外部割り込み端子に入力される信号を所定の幅でノイズを除去することができます。

9.1　構成

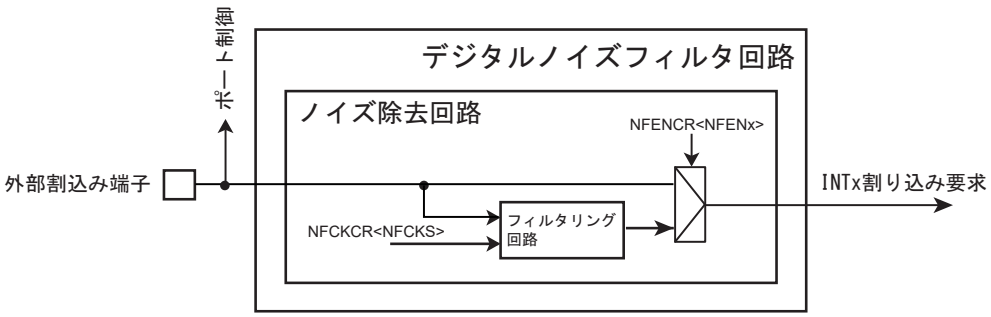


図 9-1 デジタルノイズフィルタ回路図

9.2　レジスタ説明

9.2.1　レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		Address(Base+)
ノイズフィルタ制御レジスタ	NFCKCR	0x0000
ノイズフィルタ許可レジスタ	NFENCR	0x0004

9.2.1.1　NFCKCR(ノイズフィルタ制御レジスタ)

	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	-	-	-	-	-	NFCKS		
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	NFCKS[2:0]	R/W	ノイズフィルタクロック選択 000: クロック制御回路停止 001: fsys/2 クロック出力 010: fsys/4 クロック出力 011: fsys/8 クロック出力 100: fsys/16 クロック出力 101: fsys/32 クロック出力 110: fsys/64 クロック出力 111: fsys/128 クロック出力

注 1)　NFCKCR<NFCKS>の設定は、NFENCR<NFEN[15:0]>がすべて"0"で行なってください。

注 2)　STOP モード解除に外部割込み端子を使用する場合、ノイズフィルタ回路は使用できません。必ず NFENCR レジスタのノイズフィルタ許可ビットを禁止設定にし、NFCKCR レジスタでクロックを停止してください。

9.2.1.2 NFENCR(ノイズフィルタ許可レジスタ)

	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	NFEN15	NFEN14	NFEN13	NFEN12	NFEN11	NFEN10	NFEN9	NFEN8
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	NFEN7	NFEN6	NFEN5	NFEN4	NFEN3	NFEN2	NFEN1	NFEN0
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	NFEN15	R/W	INTF ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
14	NFEN14	R/W	INTE ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
13	NFEN13	R/W	INTD ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
12	NFEN12	R/W	INTC ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
11	NFEN11	R/W	INTB ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
10	NFEN10	R/W	INTA ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
9	NFEN9	R/W	INT9 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
8	NFEN8	R/W	INT8 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
7	NFEN7	R/W	INT7 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
6	NFEN6	R/W	INT6 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
5	NFEN5	R/W	INT5 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）

Bit	Bit Symbol	Type	機能
4	NFEN4	R/W	INT4 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
3	NFEN3	R/W	INT3 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
2	NFEN2	R/W	INT2 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
1	NFEN1	R/W	INT1 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）
0	NFEN0	R/W	INT0 ノイズフィルタ許可 0: 禁止（ノイズ除去前信号出力およびノイズ除去回路カウンタクリア、STOP モード解除時） 1: 許可（ノイズ除去後信号出力）

注 1)　fsys より短いパルスに対してノイズ除去動作ができない場合があります。特に fsys の周波数が低い場合にノイズ除去動作が有効とならない場合があります。

注 2)　外部割込みを許可に設定する前に割り込み要因をクリアし、NFENCR レジスタの対象ビットを許可設定にしてください。

注 3)　STOP モード解除に外部割込み端子を使用する場合、ノイズフィルタ回路は使用できません。必ず NFENCR レジスタのノイズフィルタ許可ビットを禁止設定にし、NFCKCR レジスタでクロックを停止してください。

9.3 動作説明

9.3.1 構成

ノイズフィルタ回路は、ノイズ除去回路、割り込み要求信号発生回路から構成されています。

外部から入力された信号はノイズ除去回路によって High レベルまたは Low レベルのノイズを除去した後、各外部割り込みごとに CG で立ち上がり/立ち下がり/レベル検出を行ないます。

9.3.2 動作

外部割り込み信号 INTx の High レベル/Low レベル入力ともにノイズを除去します。

ノイズ除去時間は、レベル入力継続時間が NFCKCR<NFCKS>で設定したクロック周期の 7 クロック以下をノイズと判定し、継続時間がクロック周期の 8 クロック以上で有効な信号と判定します。

ただし、7、8 クロックの間の入力信号はエッジタイミングによって判定が異なる場合があります。

9.3.3 使用可能な動作モード

ノイズフィルタ回路は NORMAL モード、IDLE モードのときのみ使用可能です。

9.3.4 STOP モードを使用する場合の注意点

STOP モードを使用する場合、fsys クロックが停止するためノイズフィルタ回路は使用できません。STOP モード解除に外部割込みを使用する場合は割り込み許可ビットを禁止した後に NFENCR レジスタのノイズフィルタ許可/禁止ビットを禁止に設定し、NFCKCR レジスタのノイズフィルタクロックを停止させてください。

9.3.5 最小ノイズ除去時間

ノイズ除去回路は NFCKCR レジスタで設定したクロック周期の 8 クロック以上の期間、High レベル入力または Low レベル入力が入力されていた場合、レベル入力を判定し外部割込み信号を発生します。

表 9-1 最小ノイズ除去時間

NFCKCR<NFCKS>	fsys [MHz]				Unit
	20	40	80	100	
001	0.7	0.35	0.175	0.14	μs
010	1.4	0.7	0.35	0.28	
011	2.8	1.4	0.7	0.56	
100	5.6	2.8	1.4	1.12	
101	11.2	5.6	2.8	2.24	
110	22.4	11.2	5.6	4.48	
111	44.8	22.4	11.2	8.96	

第 10 章　μDMA コントローラ（μDMAC）

10.1　概要

10.1.1　機能一覧

1 ユニットあたりの主な機能を以下に説明します。

周辺機能による起動トリガの情報については、「製品情報」章を参照してください。

表 10-1 μDMA 概要(1 ユニットあたり)

項目	機能		
チャネル数	32ch		
DMA 起動トリガ	ハードウエアでスタート		周辺機能の DMA 要求で起動
	ソフトウエアでスタート		DMAxChnlSwRequest レジスタにて設定
プライオリティ	チャネル間	ch0(高優先度) > … > ch31(高優先度) > ch0(通常優先度) > … > ch31(通常優先度)	DMAxChnlPrioritySet レジスタにて高優先度設定が可能
転送データサイズ	8/16/32bit		
転送回数	1~1024 回		
アドレス	転送元アドレス	インクリメント/固定	転送元と転送先のアドレスは、固定かインクリメントするかを選択できます。
	転送先アドレス	インクリメント/固定	
エンディアン	リトルエンディアン		
割込み機能	転送終了割込み		ユニットごとに出力
	エラー割込み		
動作モード	基本モード 自動要求モード ピンポンモード メモリスキャッターギャザーモード 周辺スキャッターギャザーモード		

10.2　ブロック図

μDMA コントローラは以下の機能ブロックを内蔵しています。

- APB ブロック

  制御レジスタへのアクセスを制御します。
- AHB ブロック

  DMA 転送のバスサイクルを制御します。
- DMA 制御ブロック

  DMA 動作全体の制御を行います。
- 割り込み制御ブロック

  割り込み信号をまとめ、フラグレジスタを設定します。

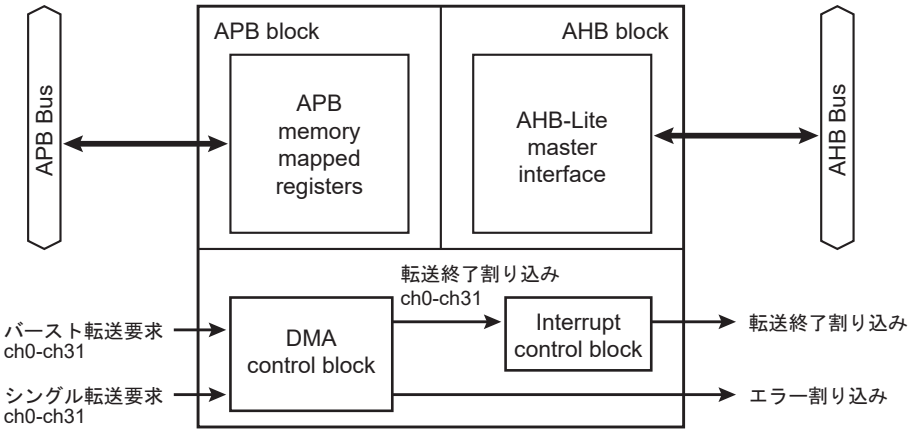


図 10-1 μDMA ブロック図(ユニット共通)

10.3 レジスタ説明

10.3.1 レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照下さい。

周辺機能名:DMA

レジスタ名		Address(Base+)
DMA status Register	DMAxStatus	0x0000
DMA configuration Register	DMAxCfg	0x0004
channel control data base pointer Register	DMAxCtrlBasePtr	0x0008
channel alternate control data base pointer Register	DMAxAltCtlBasePtr	0x000C
channel software request ststus Register	DMAxChnlSwRequest	0x0014
channel useburst set Register	DMAxChnlUseburstSet	0x0018
channel useburst clear Register	DMAxChnlUseburstClr	0x001C
channel request mask set Register	DMAxChnlReqMaskSet	0x0020
channel request mask clear Register	DMAxChnlReqMaskClr	0x0024
channel enable set Register	DMAxChnlEnableSet	0x0028
channel enable clear Register	DMAxChnlEnableClr	0x002C
channel primary-alternate set Register	DMAxChnlPriAltSet	0x0030
channel primary-alternate clear Register	DMAxChnlPriAltClr	0x0034
channel priority set Register	DMAxChnlPrioritySet	0x0038
channel priority clear Register	DMAxChnlPriorityClr	0x003C
Bus error clear Register	DMAxErrClr	0x004C

周辺機能名:DMAIF

レジスタ名		Address(Base+)
Flag Register A	DMAIFFLGA	0x0000

注) レジスタは必ずワード（32bit）アクセスしてください。

10.3.2　DMAxStatus (DMAC Status Register)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	1	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	1	1	1	1	1
	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	-	-	-	-	-	-	-	master_ enable
リセット後	不定	不定	不定	不定	0	0	0	0

Bit	Bit Symbol	Type	機能
31-29	-	R	"0"が読めます。
28	-	R	"1"が読めます。
27-21	-	R	"0"が読めます。
20-16	-	R	"1"が読めます。
15-8	-	R	"0"が読めます。
7-4	-	R	不定値が読めます。
3-1	-	R	"0"が読めます。
0	master_enable	R	DMA 動作 0: 禁止 1: 許可

10.3.3 DMAxCfg (DMAC Configuration Register)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
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	-	-	-	-	-	-	-	-
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	-	-	-	-	-	-	-	-
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	-	-	-	-	-	-	-	master_ enable
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
31-1	-	W	"0"をライトしてください。
0	master_enable	W	DMA 動作 0：禁止 1：許可

注) DMAxCfg = 0x00000001, DMAxChnlReqMaskSet = 0xFFFFFFFF, DMAxChnlEnableSet = 0xFFFFFFFF を設定した後、使用するチャネルをマスク解除(DMAxChnlReqMaskClr の該当ビットを"1"）に設定してください。

10.3.4　DMAxCtrlBasePtr (Channel control data base pointer Register)

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

Bit	Bit Symbol	Type	機能
31-10	ctrl_base_ptr	R/W	一次データベースポインタ 一次データのベースアドレスを指定します。
9-0	-	R	"0"が読めます。

10.3.5　DMAxAltCtrlBasePtr (Channel alternate control data base pointer Register)

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

Bit	Bit Symbol	Type	機能
31-0	alt_ctrl_base_pt	R	代替データベースポインタ 代替データのベースアドレスが読めます。

10.3.6 DMAxChnlSwRequest(Channel software request Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_sw_request (ch31)	chnl_sw_request (ch30)	chnl_sw_request (ch29)	chnl_sw_request (ch28)	chnl_sw_request (ch27)	chnl_sw_request (ch26)	chnl_sw_request (ch25)	chnl_sw_request (ch24)
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	chnl_sw_request (ch23)	chnl_sw_request (ch22)	chnl_sw_request (ch21)	chnl_sw_request (ch20)	chnl_sw_request (ch19)	chnl_sw_request (ch18)	chnl_sw_request (ch17)	chnl_sw_request (ch16)
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	chnl_sw_request (ch15)	chnl_sw_request (ch14)	chnl_sw_request (ch13)	chnl_sw_request (ch12q)	chnl_sw_request (ch11)	chnl_sw_request (ch10)	chnl_sw_request (ch9)	chnl_sw_request (ch8)
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	chnl_sw_request (ch7)	chnl_sw_request (ch6)	chnl_sw_request (ch5)	chnl_sw_request (ch4)	chnl_sw_request (ch3)	chnl_sw_request (ch2)	chnl_sw_request (ch1)	chnl_sw_request (ch0)
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	chnl_sw_request	W	DMA 要求 0: 転送要求しない 1: 転送要求する 各チャネルに対する転送要求を設定します。

10.3.7　DMAxChnlUseburstSet(Channel useburst set Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_useburst_set (ch31)	chnl_useburst_set (ch30)	chnl_useburst_set (ch29)	chnl_useburst_set (ch28)	chnl_useburst_set (ch27)	chnl_useburst_set (ch26)	chnl_useburst_set (ch25)	chnl_useburst_set (ch24)
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	chnl_useburst_set (ch23)	chnl_useburst_set (ch22)	chnl_useburst_set (ch21)	chnl_useburst_set (ch20)	chnl_useburst_set (ch19)	chnl_useburst_set (ch18)	chnl_useburst_set (ch17)	chnl_useburst_set (ch16)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	chnl_useburst_set (ch15)	chnl_useburst_set (ch14)	chnl_useburst_set (ch13)	chnl_useburst_set (ch12)	chnl_useburst_set (ch11)	chnl_useburst_set (ch10)	chnl_useburst_set (ch9)	chnl_useburst_set (ch8)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	chnl_useburst_set (ch7)	chnl_useburst_set (ch6)	chnl_useburst_set (ch5)	chnl_useburst_set (ch4)	chnl_useburst_set (ch3)	chnl_useburst_set (ch2)	chnl_useburst_set (ch1)	chnl_useburst_set (ch0)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	chnl_useburst_set	R/W	シングル転送禁止 [ライト] 1: シングル転送を禁止する [リード] 0: シングル転送許可 1: シングル転送禁止 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルのシングル転送が禁止され、バースト転送要求のみが有効になります。"0"の書き込みは意味を持ちません。シングル転送禁止を解除する際は、DMAxChnlUseburstClr レジスタで行います。 リードの場合、該当するチャネルのシングル転送の許可/禁止状態が確認できます。 以下の場合、自動的にビットが操作されます。 ・最後から 2 番目の 2^R 回転送("R"は制御データの channel_cfg<R_power>で設定)終了時に残りの転送回数が 2^R 回未満の場合、このビットは"0"にクリアされます。 ・周辺スキャッターギャザーモードで、制御データの channel_cfg<next_useburst>が"1"に設定されている場合、代替データによる DMA 転送終了後にこのビットに"1"が設定されます。

注)　転送回数が 2^R 回未満の設定で、バースト転送要求を使用しない場合はこのビットに"1"を設定しないでください。

10.3.8 DMAxChnlUseburstClr(Channel useburst clear Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_useburst_clr (ch31)	chnl_useburst_clr (ch30)	chnl_useburst_clr (ch29)	chnl_useburst_clr (ch28)	chnl_useburst_clr (ch27)	chnl_useburst_clr (ch26)	chnl_useburst_clr (ch25)	chnl_useburst_clr (ch24)
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	chnl_useburst_clr (ch23)	chnl_useburst_clr (ch22)	chnl_useburst_clr (ch21)	chnl_useburst_clr (ch20)	chnl_useburst_clr (ch19)	chnl_useburst_clr (ch18)	chnl_useburst_clr (ch17)	chnl_useburst_clr (ch16)
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	chnl_useburst_clr (ch15)	chnl_useburst_clr (ch14)	chnl_useburst_clr (ch13)	chnl_useburst_clr (ch12)	chnl_useburst_clr (ch11)	chnl_useburst_clr (ch10)	chnl_useburst_clr (ch9)	chnl_useburst_clr (ch8)
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	chnl_useburst_clr (ch7)	chnl_useburst_clr (ch6)	chnl_useburst_clr (ch5)	chnl_useburst_clr (ch4)	chnl_useburst_clr (ch3)	chnl_useburst_clr (ch2)	chnl_useburst_clr (ch1)	chnl_useburst_clr (ch0)
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	chnl_useburst_clr	W	シングル転送許可 1: シングル転送を許可する 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルのシングル転送を許可します。"0"の書き込みは意味を持ちません。 シングル転送の禁止および設定の確認は、DMAxChnlUseburstSet レジスタで行います。

10.3.9　DMAxChnlReqMaskSet(Channel request mask set Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_req_mask_set (ch31)	chnl_req_mask_set (ch30)	chnl_req_mask_set (ch29)	chnl_req_mask_set (ch28)	chnl_req_mask_set (ch27)	chnl_req_mask_set (ch26)	chnl_req_mask_set (ch25)	chnl_req_mask_set (ch24)
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	chnl_req_mask_set (ch23)	chnl_req_mask_set (ch22)	chnl_req_mask_set (ch21)	chnl_req_mask_set (ch20)	chnl_req_mask_set (ch19)	chnl_req_mask_set (ch18)	chnl_req_mask_set (ch17)	chnl_req_mask_set (ch16)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	chnl_req_mask_set (ch15)	chnl_req_mask_set (ch14)	chnl_req_mask_set (ch13)	chnl_req_mask_set (ch12)	chnl_req_mask_set (ch11)	chnl_req_mask_set (ch10)	chnl_req_mask_set (ch9)	chnl_req_mask_set (ch8)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	chnl_req_mask_set (ch7)	chnl_req_mask_set (ch6)	chnl_req_mask_set (ch5)	chnl_req_mask_set (ch4)	chnl_req_mask_set (ch3)	chnl_req_mask_set (ch2)	chnl_req_mask_set (ch1)	chnl_req_mask_set (ch0)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	chnl_req_mask_set	R/W	DMA 要求マスク [ライト] 1: 周辺回路からの DMA 要求をマスクする [リード] 0: DMA 外部要求は有効 1: DMA 外部要求は無効 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルに対する転送要求を無効にします。"0"の書き込みは意味を持ちません。マスクを無効にする際は、DMAxChnlReqMaskClr レジスタで行います。 リードの場合、該当するチャネルの DMA 外部要求マスク有効/無効の状態が確認できます。

10.3.10 DMAxChnlReqMaskClr(Channel request mask clear Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_req_mask_clr (ch31)	chnl_req_mask_clr (ch30)	chnl_req_mask_clr (ch29)	chnl_req_mask_clr (ch28)	chnl_req_mask_clr (ch27)	chnl_req_mask_clr (ch26)	chnl_req_mask_clr (ch25)	chnl_req_mask_clr (ch24)
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	chnl_req_mask_clr (ch23)	chnl_req_mask_clr (ch22)	chnl_req_mask_clr (ch21)	chnl_req_mask_clr (ch20)	chnl_req_mask_clr (ch19)	chnl_req_mask_clr (ch18)	chnl_req_mask_clr (ch17)	chnl_req_mask_clr (ch16)
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	chnl_req_mask_clr (ch15)	chnl_req_mask_clr (ch14)	chnl_req_mask_clr (ch13)	chnl_req_mask_clr (ch12)	chnl_req_mask_clr (ch11)	chnl_req_mask_clr (ch10)	chnl_req_mask_clr (ch9)	chnl_req_mask_clr (ch8)
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	chnl_req_mask_clr (ch7)	chnl_req_mask_clr (ch6)	chnl_req_mask_clr (ch5)	chnl_req_mask_clr (ch4)	chnl_req_mask_clr (ch3)	chnl_req_mask_clr (ch2)	chnl_req_mask_clr (ch1)	chnl_req_mask_clr (ch0)
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	chnl_req_mask_clr	W	DMA 要求マスクの解除 1: 該当するチャネルの DMA 要求マスクを解除する 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルの DMA 要求マスクを無効にします。"0"の書き込みは意味を持ちません。 有効の設定および設定の確認は、DMAxChnlReqMaskSet レジスタで行います。

10.3.11　DMAxChnlEnableSet(Channel enable set Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_enable_set (ch31)	chnl_enable_set (ch30)	chnl_enable_set (ch29)	chnl_enable_set (ch28)	chnl_enable_set (ch27)	chnl_enable_set (ch26)	chnl_enable_set (ch25)	chnl_enable_set (ch24)
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	chnl_enable_set (ch23)	chnl_enable_set (ch22)	chnl_enable_set (ch21)	chnl_enable_set (ch20)	chnl_enable_set (ch19)	chnl_enable_set (ch18)	chnl_enable_set (ch17)	chnl_enable_set (ch16)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	chnl_enable_set (ch15)	chnl_enable_set (ch14)	chnl_enable_set (ch13)	chnl_enable_set (ch12)	chnl_enable_set (ch11)	chnl_enable_set (ch10)	chnl_enable_set (ch9)	chnl_enable_set (ch8)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	chnl_enable_set (ch7)	chnl_enable_set (ch6)	chnl_enable_set (ch5)	chnl_enable_set (ch4)	chnl_enable_set (ch3)	chnl_enable_set (ch2)	chnl_enable_set (ch1)	chnl_enable_set (ch0)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	chnl_enable_set	R/W	DMA 動作 [ライト] 1: 該当チャネルを有効にする [リード] 0: 該当チャネルは無効 1: 該当チャネルは有効 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルを有効にします。"0"の書き込みは意味を持ちません。無効にする際は、DMAxChnlEnableClr レジスタで行います。 リードの場合、該当するチャネルの有効/無効の状態が確認できます。 また、以下の場合に自動的に無効になります。 ・DMA サイクル終了 ・channel_cfg<cycle_ctrl>が"000"の制御データをリードしたとき ・バスエラーが発生した時

10.3.12 DMAxChnlEnableClr(Channel enable clear Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_enable_clr (ch31)	chnl_enable_clr (ch30)	chnl_enable_clr (ch29)	chnl_enable_clr (ch28)	chnl_enable_clr (ch27)	chnl_enable_clr (ch26)	chnl_enable_clr (ch25)	chnl_enable_clr (ch24)
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	chnl_enable_clr (ch23)	chnl_enable_clr (ch22)	chnl_enable_clr (ch21)	chnl_enable_clr (ch20)	chnl_enable_clr (ch19)	chnl_enable_clr (ch18)	chnl_enable_clr (ch17)	chnl_enable_clr (ch16)
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	chnl_enable_clr (ch15)	chnl_enable_clr (ch14)	chnl_enable_clr (ch13)	chnl_enable_clr (ch12)	chnl_enable_clr (ch11)	chnl_enable_clr (ch10)	chnl_enable_clr (ch9)	chnl_enable_clr (ch8)
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	chnl_enable_clr (ch7)	chnl_enable_clr (ch6)	chnl_enable_clr (ch5)	chnl_enable_clr (ch4)	chnl_enable_clr (ch3)	chnl_enable_clr (ch2)	chnl_enable_clr (ch1)	chnl_enable_clr (ch0)
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	chnl_enable_clr	W	DMA 無効 1: 該当するチャネルを無効にする 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルを無効にします。"0"の書き込みは意味を持ちません。 有効の設定および設定の確認は、DMAxChnlEnableSet レジスタで行います。 また、以下の場合に自動的に無効になります。 ・DMA サイクル終了 ・channel_cfg<cycle_ctrl>が"000"の制御データをリードしたとき ・バスエラーが発生した時

10.3.13　DMAxChnlPriAltSet(Channel primary-alternate set Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_pri_alt_set (ch31)	chnl_pri_alt_set (ch30)	chnl_pri_alt_set (ch29)	chnl_pri_alt_set (ch28)	chnl_pri_alt_set (ch27)	chnl_pri_alt_set (ch26)	chnl_pri_alt_set (ch25)	chnl_pri_alt_set (ch24)
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	chnl_pri_alt_set (ch23)	chnl_pri_alt_set (ch22)	chnl_pri_alt_set (ch21)	chnl_pri_alt_set (ch20)	chnl_pri_alt_set (ch19)	chnl_pri_alt_set (ch18)	chnl_pri_alt_set (ch17)	chnl_pri_alt_set (ch16)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	chnl_pri_alt_set (ch15)	chnl_pri_alt_set (ch14)	chnl_pri_alt_set (ch13)	chnl_pri_alt_set (ch12)	chnl_pri_alt_set (ch11)	chnl_pri_alt_set (ch10)	chnl_pri_alt_set (ch9)	chnl_pri_alt_set (ch8)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	chnl_pri_alt_set (ch7)	chnl_pri_alt_set (ch6)	chnl_pri_alt_set (ch5)	chnl_pri_alt_set (ch4)	chnl_pri_alt_set (ch3)	chnl_pri_alt_set (ch2)	chnl_pri_alt_set (ch1)	chnl_pri_alt_set (ch0)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	chnl_pri_alt_set	R/W	一次データ/代替データ選択 [ライト] 1: 代替データを使用する [リード] 0: 一次データ 1: 代替データ 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルで最初に使用するデータを代替に設定します。"0"の書き込みは意味を持ちません。無効にする際は、DMAxChnlEnableClr レジスタで行います。最初のデータとして代替を指定できるのは、基本モード、自動要求モード、ピンポンモードです。 リードの場合、該当するチャネルのデータが一次か代替かを確認できます。 また、以下の場合に自動的に設定が切り替わります。 ・ピンポンモード、メモリスキャッターギャザーモードまたは周辺スキャッターギャザーモードで一次データによるデータ転送が終了したとき ・ピンポンモード、メモリスキャッターギャザーモードまたは周辺スキャッターギャザーモードで代替データによるデータ転送が終了したとき

10.3.14 DMAxChnlPriAltClr(Channel primary-alternate clear Register)

	31	30	29	28	27	26	25	24
bit symbol	chn_pri_alt_clr (ch31)	chn_pri_alt_clr (ch30)	chn_pri_alt_clr (ch29)	chn_pri_alt_clr (ch28)	chn_pri_alt_clr (ch27)	chn_pri_alt_clr (ch26)	chn_pri_alt_clr (ch25)	chn_pri_alt_clr (ch24)
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	chn_pri_alt_clr (ch23)	chn_pri_alt_clr (ch22)	chn_pri_alt_clr (ch21)	chn_pri_alt_clr (ch20)	chn_pri_alt_clr (ch19)	chn_pri_alt_clr (ch18)	chn_pri_alt_clr (ch17)	chn_pri_alt_clr (ch16)
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	chn_pri_alt_clr (ch15)	chn_pri_alt_clr (ch14)	chn_pri_alt_clr (ch13)	chn_pri_alt_clr (ch12)	chn_pri_alt_clr (ch11)	chn_pri_alt_clr (ch10)	chn_pri_alt_clr (ch9)	chn_pri_alt_clr (ch8)
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	chn_pri_alt_clr (ch7)	chn_pri_alt_clr (ch6)	chn_pri_alt_clr (ch5)	chn_pri_alt_clr (ch4)	chn_pri_alt_clr (ch3)	chn_pri_alt_clr (ch2)	chn_pri_alt_clr (ch1)	chn_pri_alt_clr (ch0)
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	chnl_pri_alt_clr	W	代替データ設定の解除 1: 一次データを使用する 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルのデータを一次に設定します。"0"の書き込みは意味を持ちません。代替の設定および設定の確認は、DMAxChnlPriAltSet レジスタで行います。 また、以下の場合に自動的に設定が切り替わります。 ・メモリスキャッターギャザーモードまたは周辺スキャッターギャザーモードで一次データによるデータ転送が終了したとき ・ピンポンモードで一次データによるデータ転送が終了したとき ・ピンポンモード、メモリスキャッターギャザーモード、周辺スキャッターギャザーモードで代替データによるデータ転送が終了したとき

10.3.15　DMAxChnlPrioritySet(Channel priority set Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_priority_set (ch31)	chnl_priority_set (ch30)	chnl_priority_set (ch29)	chnl_priority_set (ch28)	chnl_priority_set (ch27)	chnl_priority_set (ch26)	chnl_priority_set (ch25)	chnl_priority_set (ch24)
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	chnl_priority_set (ch23)	chnl_priority_set (ch22)	chnl_priority_set (ch21)	chnl_priority_set (ch20)	chnl_priority_set (ch19)	chnl_priority_set (ch18)	chnl_priority_set (ch17)	chnl_priority_set (ch16)
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	chnl_priority_set (ch15)	chnl_priority_set (ch14)	chnl_priority_set (ch13)	chnl_priority_set (ch12)	chnl_priority_set (ch11)	chnl_priority_set (ch10)	chnl_priority_set (ch9)	chnl_priority_set (ch8)
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	chnl_priority_set (ch7)	chnl_priority_set (ch6)	chnl_priority_set (ch5)	chnl_priority_set (ch4)	chnl_priority_set (ch3)	chnl_priority_set (ch2)	chnl_priority_set (ch1)	chnl_priority_set (ch0)
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	chnl_priority_set	R/W	優先度設定 [ライト] 1: 高優先度に設定する [リード] 0: 通常優先度 1: 高優先度 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルの優先度を高優先度に設定します。"0"の書き込みは意味を持ちません。通常優先度に戻す際は、DMAxChnlPriorityClr レジスタで行います。 リードの場合、該当するチャネルが高優先度か通常優先度かを確認できます。

10.3.16 DMAxChnlPriorityClr(Channel priority clear Register)

	31	30	29	28	27	26	25	24
bit symbol	chnl_priority_clr (ch31)	chnl_priority_clr (ch30)	chnl_priority_clr (ch29)	chnl_priority_clr (ch28)	chnl_priority_clr (ch27)	chnl_priority_clr (ch26)	chnl_priority_clr (ch25)	chnl_priority_clr (ch24)
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	chnl_priority_clr (ch23)	chnl_priority_clr (ch22)	chnl_priority_clr (ch21)	chnl_priority_clr (ch20)	chnl_priority_clr (ch19)	chnl_priority_clr (ch18)	chnl_priority_clr (ch17)	chnl_priority_clr (ch16)
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	chnl_priority_clr (ch15)	chnl_priority_clr (ch14)	chnl_priority_clr (ch13)	chnl_priority_clr (ch12)	chnl_priority_clr (ch11)	chnl_priority_clr (ch10)	chnl_priority_clr (ch9)	chnl_priority_clr (ch8)
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	chnl_priority_clr (ch7)	chnl_priority_clr (ch6)	chnl_priority_clr (ch5)	chnl_priority_clr (ch4)	chnl_priority_clr (ch3)	chnl_priority_clr (ch2)	chnl_priority_clr (ch1)	chnl_priority_clr (ch0)
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	chnl_priority_clr	W	高優先度設定の解除 [ライト] 1: 通常優先度に設定する 各ビットが指定された番号のチャネルに対応しています。 "1"をライトすることで該当するチャネルの優先度を通常に戻します。"0"の書き込みは意味を持ちません。高優先度の設定および設定の確認は、DMAxChnlPrioritySet レジスタで行います。

10.3.17　DMAxErrClr(Bus error clear Register)

	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	-	-	-	-	-	-	-	err_clr
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-1	-	R	"0"が読めます。
0	err_clr	R/W	バスエラー [ライト] 1: バスエラー解除 [リード] 0: バスエラーなし 1: バスエラー状態 リードするとバスエラーが発生しているかどうかを確認できます。 "1"をライトすることでバスエラーを解除することができます。"0"の書き込みは意味を持ちません。

10.3.18 DMAIFFLGx(DMA Flag Register)

	31	30	29	28	27	26	25	24
bit symbol	FLG31	FLG30	FLG29	FLG28	FLG27	FLG26	FLG25	FLG24
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	FLG23	FLG22	FLG21	FLG20	FLG19	FLG18	FLG17	FLG16
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	FLG15	FLG14	FLG13	FLG12	FLG11	FLG10	FLG9	FLG8
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	FLG7	FLG6	FLG5	FLG4	FLG3	FLG2	FLG1	FLG0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-1	FLG31 - FLG0	R	DMA 要因フラグ 0:終了割り込みは発生していない 1:終了割り込みが発生 ビット番号は DMA のチャネル番号に対応します。転送終了割り込みが発生すると、該当するビットが"1"にセットされます。 レジスタをリードすると"0"にクリアされます。

注) DMA 転送終了割り込みを許可する前に本レジスタをリードしてフラグをクリアしてください。

10.4　動作説明

本 DMA は、チャネル制御データによって制御されます。チャネル制御データはメモリ上に置かれたデータで、1 チャネルにつき 4 ワードのデータをチャネル数分連続した空間に配置します。

チャネル制御データには、一次データと代替データがあります。動作モードによってどちらかを設定レジスタで選択して使用する場合と、両方を使用する場合があります。

10.4.1　チャネル制御データメモリマップ

図 10-2 にチャネル制御データのメモリマップ例を示します。

一次データ用のスタートアドレスを DMAxCtrlBasePtr に、代替データ用のスタートアドレスを DMAxAltCtrlBasePtr に設定します。

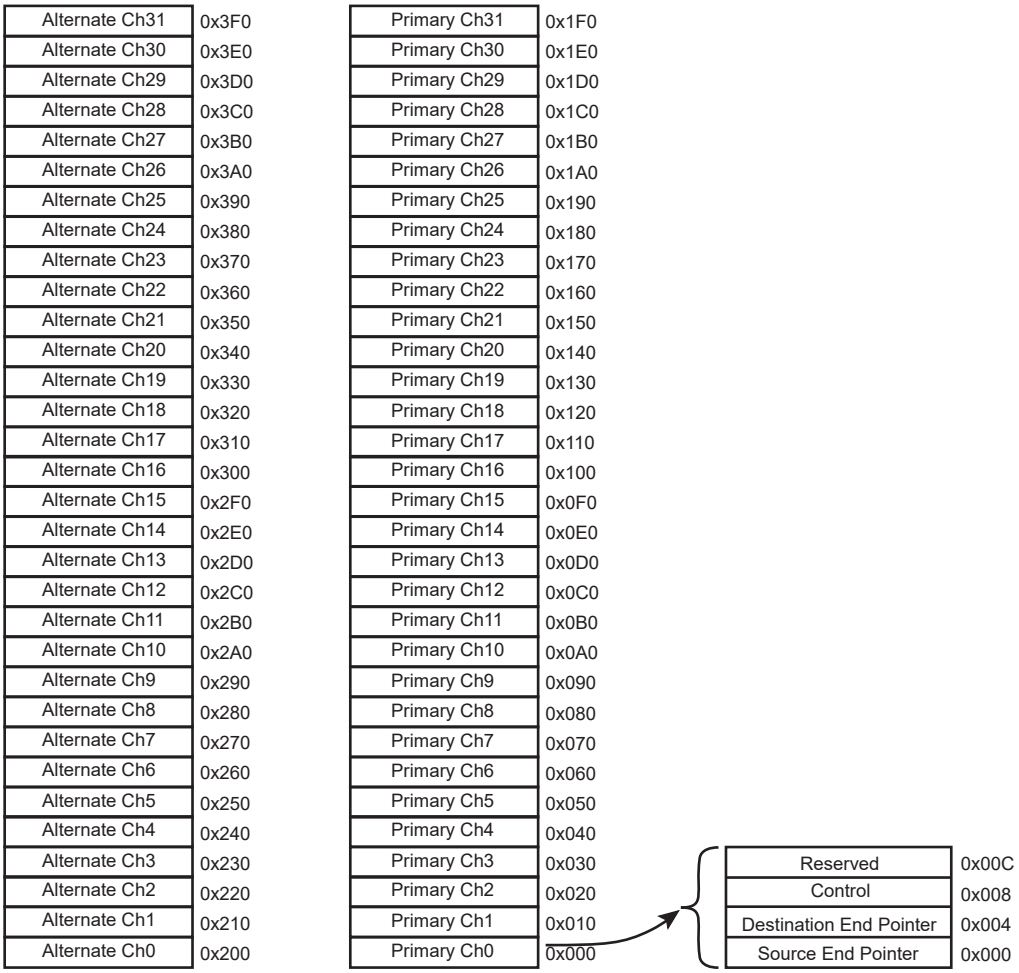


図 10-2 制御データのメモリマップ

図 10-2 は、32 チャネルすべてが使用できる場合のメモリマップです。使用できるチャネル数により必要となる領域は異なります。チャネル数とアドレスの関係を表 10-2 に示します。

表 10-2 チャネル制御データのアドレスビット設定

チャネル	アドレス							設定可能な ベースアドレス
	[9]	[8]	[7]	[6]	[5]	[4]	[3:0]	
0	-	-	-	-	-	A	チャネル制御 データ指定	0xXXXX_XX00, 0xXXXX_XX20, 0xXXXX_XX40, 0xXXXX_XX60, 0xXXXX_XX80, 0xXXXX_XXA0, 0xXXXX_XXC0, 0xXXXX_XXE0
0 ~ 1	-	-	-	-	A	C[0]		0xXXXX_XX00, 0xXXXX_XX40, 0xXXXX_XX80, 0xXXXX_XXC0
0 ~ 3	-	-	-	A	C[1:0]			0xXXXX_XX00, 0xXXXX_XX80
0 ~ 7	-	-	A	C[2:0]				0xXXXX_X000, 0xXXXX_X100, 0xXXXX_X200, 0xXXXX_X300, 0xXXXX_X400, 0xXXXX_X500, 0xXXXX_X600, 0xXXXX_X700, 0xXXXX_X800, 0xXXXX_X900, 0xXXXX_XA00, 0xXXXX_XB00, 0xXXXX_XC00, 0xXXXX_XD00, 0xXXXX_XE00, 0xXXXX_XF00
0 ~ 15	-	A	C[3:0]					0xXXXX_X000, 0xXXXX_X200, 0xXXXX_X400, 0xXXXX_X600, 0xXXXX_X800, 0xXXXX_XA00, 0xXXXX_XC00, 0xXXXX_XE00
0 ~ 31	A	C[4:0]						0xXXXX_X000, 0xXXXX_X400, 0xXXXX_X800, 0xXXXX_XC00

A : 一次/代替指定(0 : 一次、1 : 代替)
C[x:0] : チャネル番号の指定

10.4.2 チャネル制御データの構造

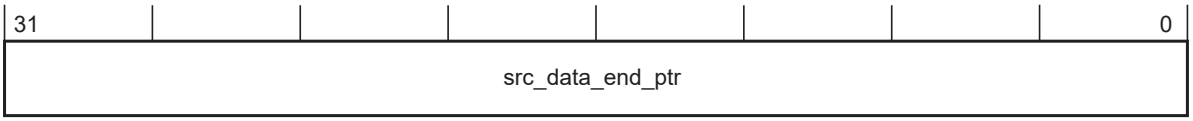
チャネル制御データは以下の 3 つのデータを含みます。

- 転送元データの最終アドレス
- 転送先の最終アドレス
- 制御データ

それぞれの内容について以下に説明します。

10.4.2.1 転送データ最終アドレス

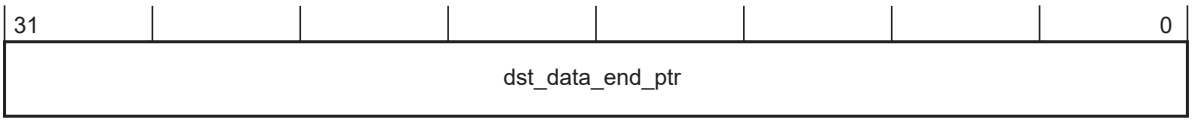
転送するデータの最終アドレスを設定します。アドレスのアライメントは、転送データサイズに合わせてください。このアドレスを元に DMA が転送元のスタートアドレスを計算します。



bit	bitsymbol	機能
[31:0]	src_data_end_ptr	転送元データの最終アドレス

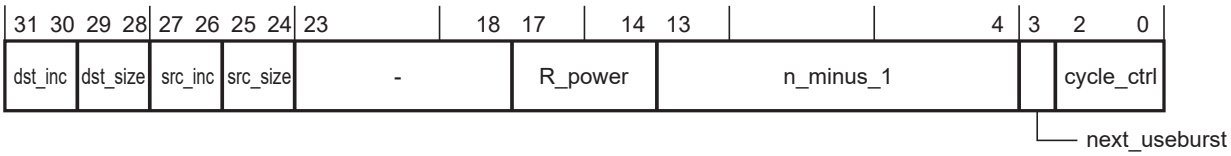
10.4.2.2　転送先の最終アドレス

転送先の最終アドレスを設定します。アドレスのアライメントは、転送データサイズに合わせてください。このアドレスを元に DMA が転送先のスタートアドレスを計算します。



bit	bitsymbol	機能
[31:0]	dst_data_end_ptr	転送先の最終アドレス

10.4.2.3　制御データ設定



bit	bit symbol	機能
[31:30]	dst_inc	転送先アドレスのインクリメント 注 2) 00: 1byte 01: 2byte 10: 4byte 11: インクリメントなし
[29:28]	dst_size	転送先データサイズ(注 1) 00: 1byte 01: 2byte 10: 4byte 11: Reserved
[27:26]	src_inc	転送元アドレスのインクリメント 注 2) 00: 1byte 01: 2byte 10: 4byte 11: インクリメントなし
[25:24]	src_size	転送元データサイズ(注 1) 00: 1byte 01: 2byte 10: 4byte 11: Reserved
[23:18]	-	"000000"を設定してください

bit	bit symbol	機能
[17:14]	R_power	アービトレーション 0000: 1 回転送後 0001: 2 回転送後 0010: 4 回転送後 0011: 8 回転送後 0100: 16 回転送後 0101: 32 回転送後 0110: 64 回転送後 0111: 128 回転送後 1000: 256 回転送後 1001: 512 回転送後 1010 - 1111: アービトレーションしない 設定した回数の転送後毎に、アービトレーションをします。
[13:4]	n_minus_1	転送回数 0x000: 1 回 0x001: 2 回 0x002: 3 回 : 0x3FF: 1024 回
[3]	next_useburst	シングル転送設定変更 0: <chnl_useburst_set>の値を変更しない 1: <chnl_useburst_set>に"1"を設定する 周辺スキャッターギャザーモードで代替データを用いた DMA 転送終了時に<chnl_useburst_set>ビットに"1"を設定するかどうかを指定します。 注)最後から 2 番目の 2^R 回転送("R"は<R_power>で設定)終了時に残りの転送回数が 2^R 回未満の場合、<chnl_useburst_set>は自動的に"0"にクリアされますが、このビットを"1"とすることで、<chnl_userburst_set>を"1"に設定できます。
[2:0]	cycle_ctrl	動作モード 000: 無効。DMA は動作を停止します。 001: 基本モード 010: 自動要求モード 011: ピンポンモード 100: メモリスキャッターギャザーモード(一次データ) 101: メモリスキャッターギャザーモード(代替データ) 110: 周辺スキャッターギャザーモード(一次データ) 111: 周辺スキャッターギャザーモード(代替データ)

注 1) <dst_size>は<src_size>と同じ値を設定してください。

注 2) <dst_size>と<src_size>の設定により、<dst_inc>と<src_inc>の設定は以下のように制限されます。

<src_inc>/<dst_inc>	<src_size>/<dst_size>		
	00 (1byte)	01 (2byte)	10 (4byte)
00(1byte)	o	-	-
01(2byte)	o	o	-
10(4byte)	o	o	o
インクリメントなし	o	o	o

10.4.3 動作モード

チャネル制御データの channel_cfg<cycle_ctrl>で設定する動作モードについて説明します。

10.4.3.1　無効

転送終了後に DMA は動作モードを無効に設定します。これにより、再度同じ転送が行われることを防ぎます。また、ピンポンモード、メモリスキャッターギャザーモード、周辺スキャッターギャザーモードの際に、無効設定のデータを読み込むと処理を終了します。

10.4.3.2　基本モード

基本モードでは、一次または代替のどちらのデータ構造を使用するか設定が可能です。

転送要求により転送を開始します。

<R_power>設定の転送ごとにアービトレーションを行い、より高い優先度の要求があればチャネルを切り替えます。動作中のチャネルの転送要求があると、転送を継続します。

<n_minus_1>に設定された回数の転送を行った後、転送終了割り込みを発生します。

10.4.3.3　自動要求モード

このモードでは 1 回の転送要求で転送を終了させることができます。一次または代替のどちらのデータ構造を使用するか設定が可能です。

転送要求により転送を開始します。

<R_power>設定の転送ごとに、より高い優先度の要求があればチャネルを切り替えます。なければ転送を継続します。

<n_minus_1>に設定された回数の転送を行った後、転送終了割り込みを発生します。

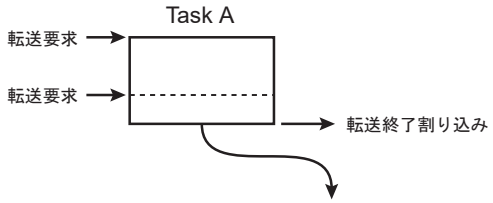
10.4.3.4　ピンポンモード

ピンポンモードでは、一次データと代替データを交互に使用しながら連続した DMA 転送を行います。<cycle_ctrl>に無効("000")が設定されたデータを読み込むか、チャネルが無効に設定されると転送を終了します。一次データおよび代替データによる転送が終了するごとに転送終了割り込みを発生します。

準備 :

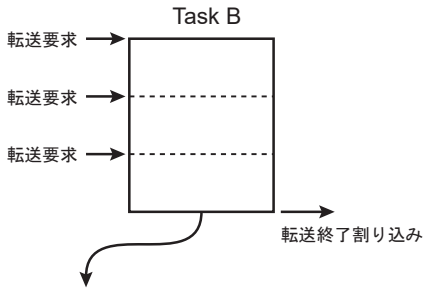
一次データと代替データを準備し、DMAxCfg<master_enable>および DMAxChnlEnableSet の該当チャネルのビットに"1"を設定します。

Task A: 一次データ
<cycle_ctrl[2:0]> = "011"
(ピンポンモード)
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0> = "0x005"
(6 回)



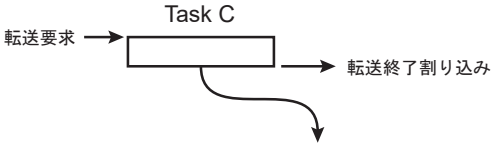
転送要求を受け、DMA は 4 回の転送を実施します。
アービトレーションを実施します。他の優先度の高い要求がない場合、該当チャネルへの転送要求により残りの 2 回の転送を実施します。
転送終了割り込み要求を発生し、アービトレーションを実施します。
Task A 終了後、Task C 用の一次データ設定が可能になります。

Task B: 代替データ
<cycle_ctrl[2:0]> = "011"
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x00B"
(12 回)



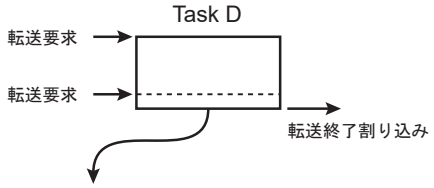
転送要求を受け、DMA は 4 回の転送を実施します。
アービトレーションを実施します。他の優先度の高い要求がない場合、該当チャネルへの転送要求により 4 回の転送を実施します。
アービトレーションを実施します。他の優先度の高い要求がない場合、該当チャネルへの転送要求により 4 回の転送を実施します。
転送終了割り込み要求を発生し、アービトレーションを実施します。
Task B 終了後、Task D 用の代替データ設定が可能になります。

Task C: 一次データ
<cycle_ctrl[2:0]> = "011"
<R_power[3:0]> = "0001"
(2 回)
<n_minus_1[9:0]> = "0x001"
(2 回)



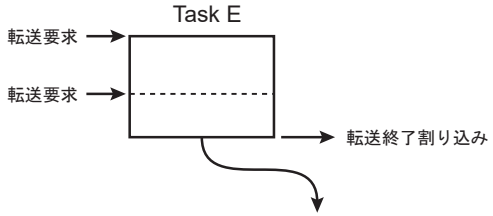
転送要求を受け、DMA は 2 回の転送を実施します。
転送終了割り込み要求を発生し、アービトレーションを実施します。
Task C 終了後、Task E 用の代替データ設定が可能になります。

Task D: 代替データ
<cycle_ctrl[2:0]> = "011"
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x004"
(5 回)



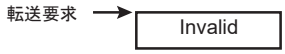
転送要求を受け、DMA は 4 回の転送を実施します。
アービトレーションを実施します。他の優先度の高い要求がない場合、該当チャネルへの転送要求により 1 回の転送を実施します。
転送終了割り込み要求を発生し、アービトレーションを実施します。

TaskE: 一次データ
<cycle_ctrl[2:0]> = "011"
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x006"
(7 回)



転送要求を受け、DMA は 4 回の転送を実施します。
アービトレーションを実施します。他の優先度の高い要求がない場合、該当チャネルへの転送要求により 3 回の転送を実施します。
転送終了割り込み要求を発生し、アービトレーションを実施します。

最終: 代替データ
<cycle_ctrl[2:0]> = "000"
(無効)



転送要求を受けますが、<cycle_ctrl>に無効が設定されているため、処理が終了します。
(Task E の<cycle_ctrl[2:0]>を"001"の通常モードに設定することによって処理を終了させることもできます。)

10.4.3.5　メモリスキャッターギャザーモード

メモリスキャッターギャザーモードでは、一次データは代替データ用のデータを転送するために使用します。

転送要求を受けると、一次データを用いて代替データの 4 つのデータを転送し、新たな転送要求なしに続けて代替データによるデータ転送を行います。その後、一次データによる代替データの転送と代替データによる転送を、<cycle_ctrl[2:0]>に無効("000")または基本モード("001)が設定されたデータを読み込むまで行います。この間、新たな転送要求必要ありません。転送終了後、割り込みを発生します。

このモードでは、一次データの channel_cfg の設定を以下のように設定する必要があります。

表 10-3 メモリスキャッターギャザーモード(一次データ)設定値

bit	bit symbol	設定値	説明
[31:30]	dst_inc	10	転送先アドレスのインクリメントとして 4byte を指定
[29:28]	dst_size	10	転送先サイズとして 4byte を指定
[27:26]	src_inc	10	転送元アドレスのインクリメントとして 4byte を指定
[25:24]	src_size	10	転送元サイズとして 4byte を指定
[17:14]	R_power	0010	アービトレーションサイクルとして 4 を指定
[13:4]	n_minus_1	N	準備する代替タスク数×4 を指定
[3]	next_useburst	0	メモリスキャッターギャザーモードでは"0"を指定
[2:0]	cycle_ctrl	100	メモリスキャッターギャザーモード(一次データ)を指定 注)

注)　<n_minus_1>に設定された回数の転送が終了すると、自動的に"000"の無効が設定されます。

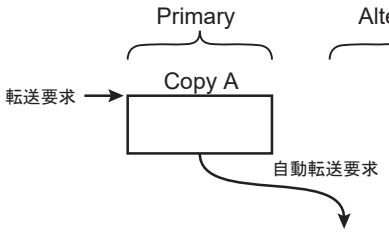
準備：

一次データを準備します。<cycle_ctrl>に"100"を設定し、転送回数<n_minus_1>には、4 つのタスク分の 4 × 4 = 16 を設定します。

タスク A,B,C,D 用の代替データを<src_data_end_ptr>に設定されたメモリ位置に準備します。

DMAxCfg<master_enable>および DMAxChnlEnableSet の該当チャネルのビットに"1"を設定します。

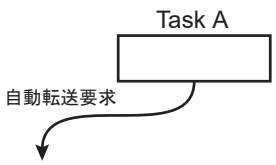
Copy A: 一次データ
<cycle_ctrl[2:0]> = "100"
(メモリスキャッターギャザー)
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x00F"
(16 回)



転送要求を受け、DMA は Task A の代替データ用の 4 回の転送を実施します。

転送終了後、自動的に転送要求が発生しアービトレーションを行います。

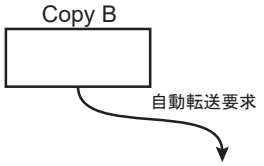
Task A: 代替データ
<cycle_ctrl[2:0]> = "100"
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x002"
(3 回)



DMA はタスク A を実行します。

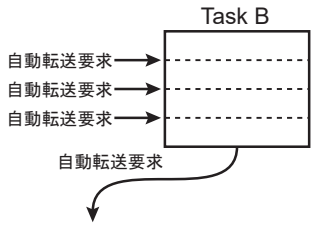
転送終了後、自動的に転送要求が発生しアービトレーションを行います。

Copy B: 一次データ



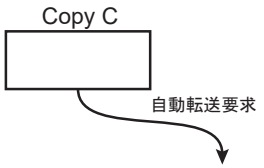
DMA は Task B の代替データ用の 4 回の転送を実施します。
転送終了後、自動的に転送要求が発生しアービトレーションを行います。

Task B: 代替データ
<cycle_ctrl[2:0]> = "100"
<R_power[3:0]> = "0001"
(2 回)
<n_minus_1[9:0]> = "0x007"
(8 回)



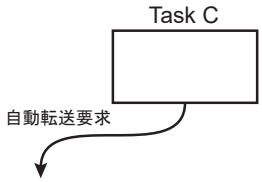
DMA はタスク B を実行します。
転送終了後、自動的に転送要求が発生しアービトレーションを行います。

Copy C: 一次データ



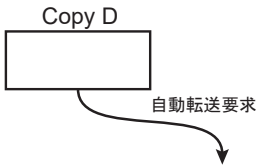
DMA は Task C の代替データ用の 4 回の転送を実施します。
転送終了後、自動的に転送要求が発生しアービトレーションを行います。

Task C: 代替データ
<cycle_ctrl[2:0]> = "100"
<R_power[3:0]> = "0011"
(8 回)
<n_minus_1[9:0]> = "0x004"
(5 回)



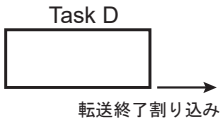
DMA はタスク C を実行します。
転送終了後、自動的に転送要求が発生しアービトレーションを行います。

Copy D: 一次データ



DMA は Task D の代替データ用の 4 回の転送を実施します。また、一次データの<cycle_ctrl[2:0]>に"000"を設定し次の一次データを無効とします。
自動的に転送要求が発生しアービトレーションを行います。

Task D: 代替データ
<cycle_ctrl[2:0]> = "001"
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x003"
(4 回)



DMA はタスク D を実行します。
<cycle_ctrl[2:0]>が"001"の基本モードに設定されているため、転送終了後に転送終了割り込み要求を発生し処理を終了ます。

10.4.3.6 周辺スキャッターギャザーモード

周辺スキャッターギャザーモードでは、一次データは代替データ用のデータを転送するために使用します。

転送要求を受けると、一次データを用いて代替データの 4 つのデータを転送し、続けて代替データによるデータ転送を行います。

その後、転送要求が発生すると一次データによる代替データの転送と代替データによる転送を、<cycle_ctrl[2:0]>に無効("000")または基本モード("001)が設定されたデータを読み込むまで行います。この間、新たな転送要求必要ありません。転送終了後、割り込みを発生します。

このモードでは、channel_cfg の設定を以下のようにする必要があります。

表 10-4 周辺スキャッターギャザーモード(一次データ)固定値

bit	bit symbol	設定値	説明
[31:30]	dst_inc	10	転送先アドレスのインクリメントとして 4byte を指定
[29:28]	dst_size	10	転送先サイズとして 4byte を指定
[27:26]	src_inc	10	転送元アドレスのインクリメントとして 4byte を指定
[25:24]	src_size	10	転送元サイズとして 4byte を指定
[17:14]	R_power	0010	アービトレーションサイクルとして 4 を指定
[13:4]	n_minus_1	N	準備する代替タスク数×4 を指定
[2:0]	cycle_ctrl	110	周辺スキャッターギャザーモード(一次データ)を指定

注)　<n_minus_1>に設定された回数の転送が終了すると、自動的に"000"の無効が設定されます。

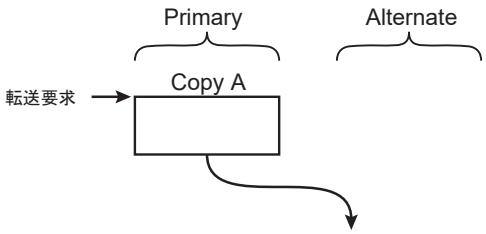
準備：

一次データを準備します。<cycle_ctrl>に"110"を設定し、転送回数<n_minus_1>には、4 つのタスク分の 4 × 4 = 16 を設定します。

タスク A,B,C,D 用の代替データを<src_data_end_ptr>に設定されたメモリ位置に準備します。

DMAxCfg<master_enable>および DMAxChnlEnableSet の該当チャネルのビットに"1"を設定します。

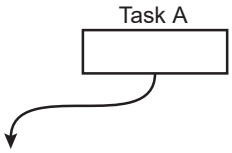
Copy A: 一次データ
<cycle_ctrl[2:0]> = "110"
(周辺スキャッターギャザー)
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x00F"
(16 回)



転送要求を受け、DMA は Task A の代替データ用の 4 回の転送を実施します。

転送終了後、自動的にタスク A の処理に移行します。

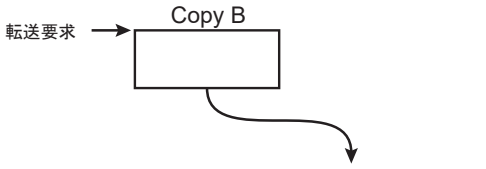
Task A: 代替データ
<cycle_ctrl[2:0]> = "111"
<R_power[3:0]> = "0010"
(4 回)
<n_minus_1[9:0]> = "0x002"
(3 回)



DMA はタスク A を実行します。

転送終了後、周辺機能からの転送要求があり、最高優先度の場合次の処理を開始します。

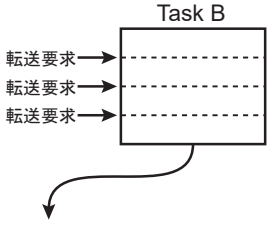
Copy B: 一次データ



DMA は Task B の代替データ用の 4 回の転送を実施します。

転送終了後、自動的にタスク B の処理に移行します。

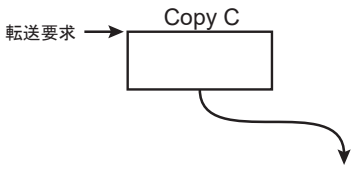
Task B: 代替データ
<cycle_ctrl[2:0]> = "111"
<R_power[3:0]> = "0001"
(2 回)
<n_minus_1[9:0]> = "0x007"
(8 回)



DMA はタスク B を実行します。2^R 回転送ごとにアービトレーションが発生するため、タスク B が終了するためには少なくとも 3 回の転送要求が必要です。

転送終了後、周辺機能からの転送要求があり、最高優先度の場合次の処理を開始します。

Copy C: 一次データ



DMA は Task C の代替データ用の 4 回の転送を実施します。

転送終了後、自動的にタスク C の処理に移行します。

Task C: 代替データ

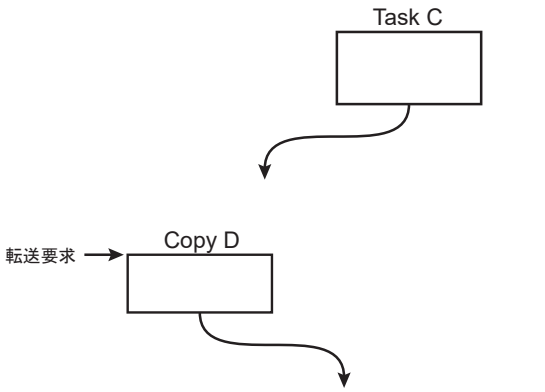
<cycle_ctrl[2:0]> = "111"

<R_power[3:0]> = "0011"

(8 回)

<n_minus_1[9:0]> = "0x004"

(5 回)



DMA はタスク C を実行します。

転送終了後、周辺機能からの転送要求があり、最高優先度の場合次の処理を開始します。

Copy D: 一次データ

DMA は Task D の代替データ用の 4 回の転送を実施します。また、一次データの<cycle_ctrl[2:0]> に"000"を設定し次の一次データを無効とします。

自動的にタスク D の処理に移行します。

Task D: 代替データ

<cycle_ctrl[2:0]> = "001"

<R_power[3:0]> = "0010"

(4 回)

<n_minus_1[9:0]> = "0x003"

(4 回)

DMA はタスク D を実行します。

<cycle_ctrl[2:0]>が"001"の基本モードに設定されているため、転送終了後に転送終了割り込み要求を発生し処理を終了ます。

10.5　使用上の注意

下記の周辺機能の DMA 転送要求を使用して転送を行う場合、使い方に注意が必要です。

- 4 バイト FIFO 付きシリアルチャネル(SIO/UART)
- 16 ビットタイマ/イベントカウンタ(TMRB)
- アナログ/デジタルコンバータ(ADC)

10.5.1　SIO/UART、TMRB、ADC を使用する場合

以下の点に注意して使用してください。

- 転送モードは基本モードを推奨します。
- アービトレーションは"1 回転送後"にしてください。

  制御データのアービトレーション<R_power>設定を"0000"としてください。
- SIO/UART の FIFO は使用しないでください。

  SIO/UART は、シングルバッファまたはダブルバッファの設定で使用してください。

転送開始が待たされて新たな要求が同一チャネルで発生した場合、転送は 1 回しか行われません。確実に転送が行われるよう、プログラム設計の際に考慮してください。

転送開始が待たされる状況として、以下のような場合が考えられます。

- 同一ユニット内の優先度の高い転送要求が発生した場合
- 他の優先度の高いバスマスタとアクセス対象が同じ場合

本 μDMA コントローラは、前処理/後処理に約 11 クロック、周辺機能と内蔵 RAM 間の 1 データの転送に約 5 クロックかかります。転送の待ち時間の目安としてください。

第 11 章　入出力ポート

本章では、ポート関連のレジスタとその設定および回路について説明します。

11.1　レジスタ説明

ポートを使用する際には以下のレジスタを設定する必要があります。

レジスタは全て 32 ビットですが、ポートのビット数、機能の割り当てにより構成が異なります。

以下の説明では"x"はポート名、"n"はファンクション番号を示します。

レジスタ名		設定値	
PxDATA	データレジスタ	0 または 1	ポートのデータ読み込み、データ書き込みを行います。
PxCR	出力コントロールレジスタ	0 : 出力禁止 1 : 出力許可	出力の制御を行います。
PxFRn	ファンクションレジスタ n	0 : PORT 1 : 機能	機能設定を行ないます。 "1" をセットすることにより割り当てられている機能を使用できるようになります。ファンクションレジスタはポートに割り当てられている機能ごとに存在します。複数の機能が割り当てられている場合、1 つの機能のみ有効になるように設定してください。
PxOD	オープンドレインコントロールレジスタ	0 : CMOS 1 : オープンドレイン	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	プルアップコントロールレジスタ	0 : プルアップ禁止 1 : プルアップ許可	プログラマブルプルアップを制御します。
PxPDN	プルダウンコントロールレジスタ	0 : プルダウン禁止 1 : プルダウン許可	プログラマブルプルダウンを制御します。
PxIE	入力コントロールレジスタ	0 : 入力禁止 1 : 入力許可	入力の制御を行ないます。 PxIE をイネーブルにしてから外部データが PxDATA に反映されるまで若干の時間が必要です。

11.1.1　レジスター覧

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

機能の存在しないビットをリードすると"0"が読めます。ライトは意味を持ちません。

レジスタ名	Address (Base+)	ポート A	ポート B	ポート C	ポート D	ポート E
データ レジスタ	0x0000	PADATA	PBDATA	PCDATA	PDDATA	PEDATA
出力コントロールレジスタ	0x0004	PACR	PBCR	PCCR	PDCR	PECR
ファンクションレジスタ 1	0x0008	PAFR1	PBFR1	PCFR1	PDFR1	PEFR1
ファンクションレジスタ 2	0x000C	PAFR2	−	−	PDFR2	PEFR2
オープンドレインコントロールレジスタ	0x0028	PAOD	PBOD	PCOD	PDOD	PEOD
プルアップコントロールレジスタ	0x002C	PAPUP	PBPUP	PCPUP	PDPUP	PEPUP
プルダウンコントロールレジスタ	0x0030	PAPDN	PBPDN	PCPDN	PDPDN	PEPDN
入力コントロールレジスタ	0x0038	PAIE	PBIE	PCIE	PDIE	PEIE

レジスタ名	Address (Base+)	ポート F	ポート G	ポート H	ポート J	ポート K
データ レジスタ	0x0000	PFDATA	PGDATA	PHDATA	PJDATA	PKDATA
出力コントロールレジスタ	0x0004	PFCR	PGCR	PHCR	PJCR	PKCR
ファンクションレジスタ 1	0x0008	PFFR1	PGFR1	–	–	PKFR1
ファンクションレジスタ 2	0x000C	PFFR2	–	–	–	–
ファンクションレジスタ 3	0x0010	PFFR3	–	–	–	–
オープンドレインコントロールレジスタ	0x0028	PFOD	PGOD	PHOD	PJOD	PKOD
プルアップコントロールレジスタ	0x002C	PFPUP	PGPUP	PHPUP	PJPUP	PKPUP
プルダウンコントロールレジスタ	0x0030	PFPDN	PGPDN	PHPDN	PJPDN	PKPDN
入力コントロールレジスタ	0x0038	PFIE	PGIE	PHIE	PJIE	PKIE

レジスタ名	Address (Base+)	ポート L	ポート N	ポート P
データ レジスタ	0x0000	PLDATA	PNDATA	PPDATA
出力コントロールレジスタ	0x0004	PLCR	PNCR	PPCR
ファンクションレジスタ 1	0x0008	–	PNFR1	–
ファンクションレジスタ 2	0x000C	–	PNFR2	–
オープンドレインコントロールレジスタ	0x0028	PLOD	PNOD	PPOD
プルアップコントロールレジスタ	0x002C	PLPUP	PNPUP	PPPUP
プルダウンコントロールレジスタ	0x0030	PLPDN	PNPDN	PPPDN
入力コントロールレジスタ	0x0038	PLIE	PNIE	PPIE

注)　"–"表記のアドレスにはアクセスしないでください。

11.1.2 ポート機能と設定一覧

以下に各ポートの機能とレジスタ設定についての一覧を示します。

- 「表 11-1 ポート A レジスタ設定」
- 「表 11-2 ポート B レジスタ設定」
- 「表 11-3 ポート C レジスタ設定」
- 「表 11-4 ポート D レジスタ設定」
- 「表 11-5 ポート E レジスタ設定」
- 「表 11-6 ポート F レジスタ設定」
- 「表 11-7 ポート G レジスタ設定」
- 「表 11-8 ポート H レジスタ設定」
- 「表 11-9 ポート J レジスタ設定」
- 「表 11-10 ポート K レジスタ設定」
- 「表 11-11 ポート L レジスタ設定」
- 「表 11-12 ポート N レジスタ設定」
- 「表 11-13 ポート P レジスタ設定」

PxFRn の欄は、設定の必要なファンクションレジスタを示します。このレジスタを"1"に設定するとその機能が有効となります。(x はポート名、n はファンクション番号)

表中の網掛けのビットはリードすると"0" が読め、ライトは意味を持ちません。

表中の"0"、"1"は設定値を示し、"0/1"は任意に設定可能であることを示します。

TMPM470FDFG/FZFG/FYFG では異なるポートに同一機能が割り振られている機能端子があります。同一機能の兼用機能端子はそれぞれのポートのファンクションレジスタの設定により、いずれか 1 つの端子で使用するよう排他的に切り替えを行って下さい。

Pxm は、ポート名"x"、そのレジスタのビット"m"を表します。例えば、PA0 とは、ポート名が A、ポート A のレジスタのビットが 0 を示します。

11.1.2.1　PORT A

表 11-1 ポート A レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PADATA	PACR	PAFRn	PAOD	PAPUP	PAPDN	PAIE
PA0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INT3	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TB0IN	Input	FT1	0/1	0	PAFR1	0/1	0/1	0/1	1
PA1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB0OUT	Output	FT1	0/1	1	PAFR1	0/1	0/1	0/1	0
PA2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INT4	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TB1IN	Input	FT1	0/1	0	PAFR1	0/1	0/1	0/1	1
PA3	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB1OUT	Output	FT1	0/1	1	PAFR1	0/1	0/1	0/1	0
PA4	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC1SCLK	Input	FT1	0/1	0	PAFR1	0/1	0/1	0/1	1
		Output		0/1	1	PAFR1	0/1	0/1	0/1	0
	$\overline{\text{SC1CTS}}$	Input	FT2	0/1	0	PAFR2	0/1	0/1	0/1	1
PA5	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC1TXD	Output	FT1	0/1	1	PAFR1	0/1	0/1	0/1	0
	TB6OUT	Output	FT1	0/1	1	PAFR2	0/1	0/1	0/1	0
PA6	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC1RXD	Input	FT1	0/1	0	PAFR1	0/1	0/1	0/1	1
	TB6IN	Input	FT1	0/1	0	PAFR2	0/1	0/1	0/1	1
PA7	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INT8	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TB4IN	Input	FT1	0/1	0	PAFR1	0/1	0/1	0/1	1

11.1.2.2 PORT B

表 11-2 ポート B レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PBDATA	PBCR	PBFRn	PBOD	PBPUP	PBPDN	PBIE
PB0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TRACECLK0	Output	FT1	0/1	1	PBFR1	0/1	0/1	0/1	0
PB1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TRACEDATA0	Output	FT1	0/1	1	PBFR1	0/1	0/1	0/1	0
PB2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TRACEDATA1	Ouput	FT1	0/1	1	PBFR1	0/1	0/1	0/1	0
PB3	リセット後 (TSM/SWDIO)			0	1	PBFR1	0	1	0	1
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TMS	I/O	FT3	0/1	1	PBFR1	0/1	0/1	0/1	1
	SWDIO	I/O	FT3	0/1	1	PBFR1	0/1	0/1	0/1	1
PB4	リセット後 (TCK/SWCLK)			0	0	PBFR1	0	0	1	1
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TCK	Input	FT3	0/1	0	PBFR1	0/1	0/1	0/1	1
	SWCLK	Input	FT3	0/1	0	PBFR1	0/1	0/1	0/1	1
PB5	リセット後 (TDO/SWV)			0	1	PBFR1	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TDO	Output	FT3	0/1	1	PBFR1	0/1	0/1	0/1	0
	SWV	Output	FT3	0/1	1	PBFR1	0/1	0/1	0/1	0
PB6	リセット後 (TDI)			0	0	PBFR1	0	1	0	1
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TDI	Input	FT3	0/1	0	PBFR1	0/1	0/1	0/1	1
PB7	リセット後 ($\overline{\text{TRST}}$)			0	0	PBFR1	0	1	0	1
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	$\overline{\text{TRST}}$	Input	FT3	0/1	0	PBFR1	0/1	0/1	0/1	1

11.1.2.3　PORT C

表 11-3 ポート C レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PCDATA	PCCR	PCFRn	PCOD	PCPUP	PCPDN	PCIE
PC0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	UO0	Output	FT2	0/1	1	PCFR1	0/1	0/1	0/1	0
PC1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	XO0	Output	FT2	0/1	1	PCFR1	0/1	0/1	0/1	0
PC2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	VO0	Output	FT2	0/1	1	PCFR1	0/1	0/1	0/1	0
PC3	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	YO0	Output	FT2	0/1	1	PCFR1	0/1	0/1	0/1	0
PC4	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	WO0	Output	FT2	0/1	1	PCFR1	0/1	0/1	0/1	0
PC5	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ZO0	Output	FT2	0/1	1	PCFR1	0/1	0/1	0/1	0
PC6	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	$\overline{\text{EMG0}}$	Input	FT1	0/1	0	PCFR1	0/1	0/1	0/1	1
PC7	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	$\overline{\text{OVV0}}$	Input	FT1	0/1	0	PCFR1	0/1	0/1	0/1	1

11.1.2.4 PORT D

表 11-4 ポート D レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PDDATA	PDCR	PDFRn	PDOD	PDPUP	PDPDN	PDIE
PD0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ENCA0	Input	FT1	0/1	0	PDFR1	0/1	0/1	0/1	1
	TB5IN	Input	FT1	0/1	0	PDFR2	0/1	0/1	0/1	1
PD1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ENCB0	Input	FT1	0/1	0	PDFR1	0/1	0/1	0/1	1
	TB5OUT	Output	FT1	0/1	1	PDFR2	0/1	0/1	0/1	0
PD2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ENCZ0	Input	FT1	0/1	0	PDFR1	0/1	0/1	0/1	1
PD3	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INT9	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
PD4	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC2SCLK	Input	FT1	0/1	0	PDFR1	0/1	0/1	0/1	1
		Output		0/1	1	PDFR1	0/1	0/1	0/1	0
	$\overline{\text{SC2CTS}}$	Input	FT1	0/1	0	PDFR2	0/1	0/1	0/1	1
PD5	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC2TXD	Output	FT1	0/1	1	PDFR1	0/1	0/1	0/1	0
PD6	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC2RXD	Input	FT1	0/1	0	PDFR1	0/1	0/1	0/1	1

11.1.2.5　PORT E

表 11-5 ポート E レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PEDATA	PECR	PEFRn	PEOD	PEPUP	PEPDN	PEIE
PE0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC0TXD	Output	FT1	0/1	1	PEFR1	0/1	0/1	0/1	0
	TB8OUT	Output	FT1	0/1	1	PEFR2	0/1	0/1	0/1	0
PE1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC0RXD	Input	FT1	0/1	0	PEFR1	0/1	0/1	0/1	1
	TB8IN	Input	FT1	0/1	0	PEFR2	0/1	0/1	0/1	1
PE2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SC0SCLK	Input	FT1	0/1	0	PEFR1	0/1	0/1	0/1	1
		Output		0/1	1	PEFR1	0/1	0/1	0/1	0
	$\overline{\text{SC0CTS}}$	Input	FT1	0/1	0	PEFR2	0/1	0/1	0/1	1
PE3	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB4OUT	Output	FT1	0/1	1	PEFR1	0/1	0/1	0/1	0
PE4	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INT5	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TB2IN	Input	FT1	0/1	0	PEFR1	0/1	0/1	0/1	1
PE5	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB2OUT	Output	FT1	0/1	1	PEFR1	0/1	0/1	0/1	0
PE6	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INT6	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TB3IN	Input	FT1	0/1	0	PEFR1	0/1	0/1	0/1	1
PE7	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INT7	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TB3OUT	Output	FT1	0/1	1	PEFR1	0/1	0/1	0/1	0

11.1.2.6 PORT F

表 11-6 ポート F レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PFDATA	PFCR	PFFRn	PFOD	PFPUP	PFPDN	PFIE
PF0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB7IN	Input	FT1	0/1	0	PFFR1	0/1	0/1	0/1	1
PF1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB7OUT	Output	FT1	0/1	1	PFFR1	0/1	0/1	0/1	0
PF2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ENCA1	Input	FT1	0/1	0	PFFR1	0/1	0/1	0/1	1
	SC3SCLK	Input	FT1	0/1	0	PFFR2	0/1	0/1	0/1	1
		Output		0/1	1	PFFR2	0/1	0/1	0/1	0
	$\overline{\text{SC3CTS}}$	Input	FT1	0/1	0	PFFR3	0/1	0/1	0/1	1
PF3	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ENCB1	Input	FT1	0/1	0	PFFR1	0/1	0/1	0/1	1
	SC3TXD	Output	FT1	0/1	1	PFFR2	0/1	0/1	0/1	0
PF4	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ENCZ1	Input	FT1	0/1	0	PFFR1	0/1	0/1	0/1	1
	SC3RXD	Input	FT1	0/1	0	PFFR2	0/1	0/1	0/1	1

注) PF0 はリセット信号 が"Low" の間 Pull-up と入力が許可になっており、$\overline{\text{BOOT}}$ 端子として機能します。
リセット信号の立ち上がりで PF0 が"1"の場合はシングルチップモード(内蔵フラッシュメモリから起動)、"0"の場合はシングルブートモード（内蔵 BOOT ROM から起動)となります。

11.1.2.7　PORT G

表 11-7 ポート G レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PGDATA	PGCR	PGFRn	PGOD	PGPUP	PGPDN	PGIE
PG0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	UO1	Output	FT2	0/1	1	PGFR1	0/1	0/1	0/1	0
PG1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	XO1	Output	FT2	0/1	1	PGFR1	0/1	0/1	0/1	0
PG2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	VO1	Output	FT2	0/1	1	PGFR1	0/1	0/1	0/1	0
PG3	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	YO1	Output	FT2	0/1	1	PGFR1	0/1	0/1	0/1	0
PG4	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	WO1	Output	FT2	0/1	1	PGFR1	0/1	0/1	0/1	0
PG5	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	ZO1	Output	FT2	0/1	1	PGFR1	0/1	0/1	0/1	0
PG6	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	$\overline{\text{EMG1}}$	Input	FT1	0/1	0	PGFR1	0/1	0/1	0/1	1
PG7	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	$\overline{\text{OVV1}}$	Input	FT1	0/1	0	PGFR1	0/1	0/1	0/1	1

11.1.2.8 PORT H

表 11-8 ポート H レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PHDATA	PHCR	PHFRn	PHOD	PHPUP	PHPDN	PHIE
PH0	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA0	Input	FT5	0/1	0		0/1	0	0	0
	INT0	Input	FT4	0/1	0		0/1	0	0/1	1
PH1	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA1	Input	FT5	0/1	0		0/1	0	0	0
	INT1	Input	FT4	0/1	0		0/1	0	0/1	1
PH2	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA2	Input	FT5	0/1	0		0/1	0	0	0
	INT2	Input	FT4	0/1	0		0/1	0	0/1	1
PH3	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA3	Input	FT5	0/1	0		0/1	0	0	0
PH4	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA4	Input	FT5	0/1	0		0/1	0	0	0
PH5	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA5	Input	FT5	0/1	0		0/1	0	0	0
PH6	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA6	Input	FT5	0/1	0		0/1	0	0	0
PH7	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINA7	Input	FT5	0/1	0		0/1	0	0	0

11.1.2.9　PORT J

表 11-9 ポート J レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PJDATA	PJCR	PJFRn	PJOD	PJPUP	PJPDN	PJIE
PJ0	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB3	Input	FT5	0/1	0		0/1	0	0	0
PJ1	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB4	Input	FT5	0/1	0		0/1	0	0	0
PJ2	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB5	Input	FT5	0/1	0		0/1	0	0	0
PJ3	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB6	Input	FT5	0/1	0		0/1	0	0	0
PJ4	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB7	Input	FT5	0/1	0		0/1	0	0	0
PJ5	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB8	Input	FT5	0/1	0		0/1	0	0	0
PJ6	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB9	Input	FT5	0/1	0		0/1	0	0	0
	INTC	Input	FT4	0/1	0		0/1	0/1	0/1	1
PJ7	リセット後			0	0		0	0	0	0
	Input Port	Input		0	0		0/1	0/1	0/1	1
	Output Port	Output		0	1		0/1	0/1	0/1	0
	AINB10	Input	FT5	0/1	0		0/1	0	0	0
	INTD	Input	FT4	0/1	0		0/1	0/1	0/1	1

11.1.2.10 PORT K

表 11-10 ポート K レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PKDATA	PKCR	PKFRn	PKOD	PKPUP	PKPDN	PKIE
PK0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INTE	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TRACEDATA3	Output	FT1	0/1	1	PKFR1	0/1	0/1	0/1	0
PK1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INTF	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
	TRACEDATA2	Output	FT1	0/1	1	PKFR1	0/1	0/1	0/1	0

11.1.2.11 PORT L

表 11-11 ポート L レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PLDATA	PLCR	PLFRn	PLOD	PLPUP	PLPDN	PLIE
PL0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INTB	Input	FT4	0/1	0	0	0/1	0/1	0/1	1
PL1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	INTA	Input	FT4	0/1	0	0	0/1	0/1	0/1	1

11.1.2.12　PORT N

表 11-12 ポート N レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PNDATA	PNCR	PNFRn	PNOD	PNPUP	PNPDN	PNIE
PN0	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	SB0SDA	I/O	FT1	0/1	1	PNFR2	1	0/1	0/1	1
PN1	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB9OUT	Output		0/1	1	PNFR1	0/1	0/1	0/1	0
	SB0SCL	I/O	FT1	0/1	1	PNFR2	1	0/1	0/1	1
PN2	リセット後			0	0	0	0	0	0	0
	Input Port	Input		0/1	0	0	0/1	0/1	0/1	1
	Output Port	Output		0/1	1	0	0/1	0/1	0/1	0
	TB9IN	Input	FT1	0/1	0	PNFR1	0/1	0/1	0/1	1

11.1.2.13　PORT P

表 11-13 ポート P レジスタ設定

PORT	リセット状態	Input/Output	PORT Type	制御レジスタ						
				PPDATA	PPCR	PPFRn	PPOD	PPPUP	PPPDN	PPIE
PP0	リセット後			0	0		0	0	0	0
	Input Port	Input		0/1	0		0/1	0/1	0/1	1
	Output Port	Output		0/1	1		0/1	0/1	0/1	0
	AINA8	Input	FT5	0/1	0		0/1	0	0	0
PP1	リセット後			0	0		0	0	0	0
	Input Port	Input		0/1	0		0/1	0/1	0/1	1
	Output Port	Output		0/1	1		0/1	0/1	0/1	0
	AINA9	Input	FT5	0/1	0		0/1	0	0	0
	AINB0	Input	FT5	0/1	0		0/1	0	0	0
PP2	リセット後			0	0		0	0	0	0
	Input Port	Input		0/1	0		0/1	0/1	0/1	1
	Output Port	Output		0/1	1		0/1	0/1	0/1	0
	AINA10	Input	FT5	0/1	0		0/1	0	0	0
	AINB1	Input	FT5	0/1	0		0/1	0	0	0
PP3	リセット後			0	0		0	0	0	0
	Input Port	Input		0/1	0		0/1	0/1	0/1	1
	Output Port	Output		0/1	1		0/1	0/1	0/1	0
	AINA11	Input	FT5	0/1	0		0/1	0	0	0
	AINB2	Input	FT5	0/1	0		0/1	0	0	0

11.2 ポート回路図

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。

回路図内の"ダイレクトリセット"は、コールドリセット時に有効になります。

11.2.1 タイプ FT1

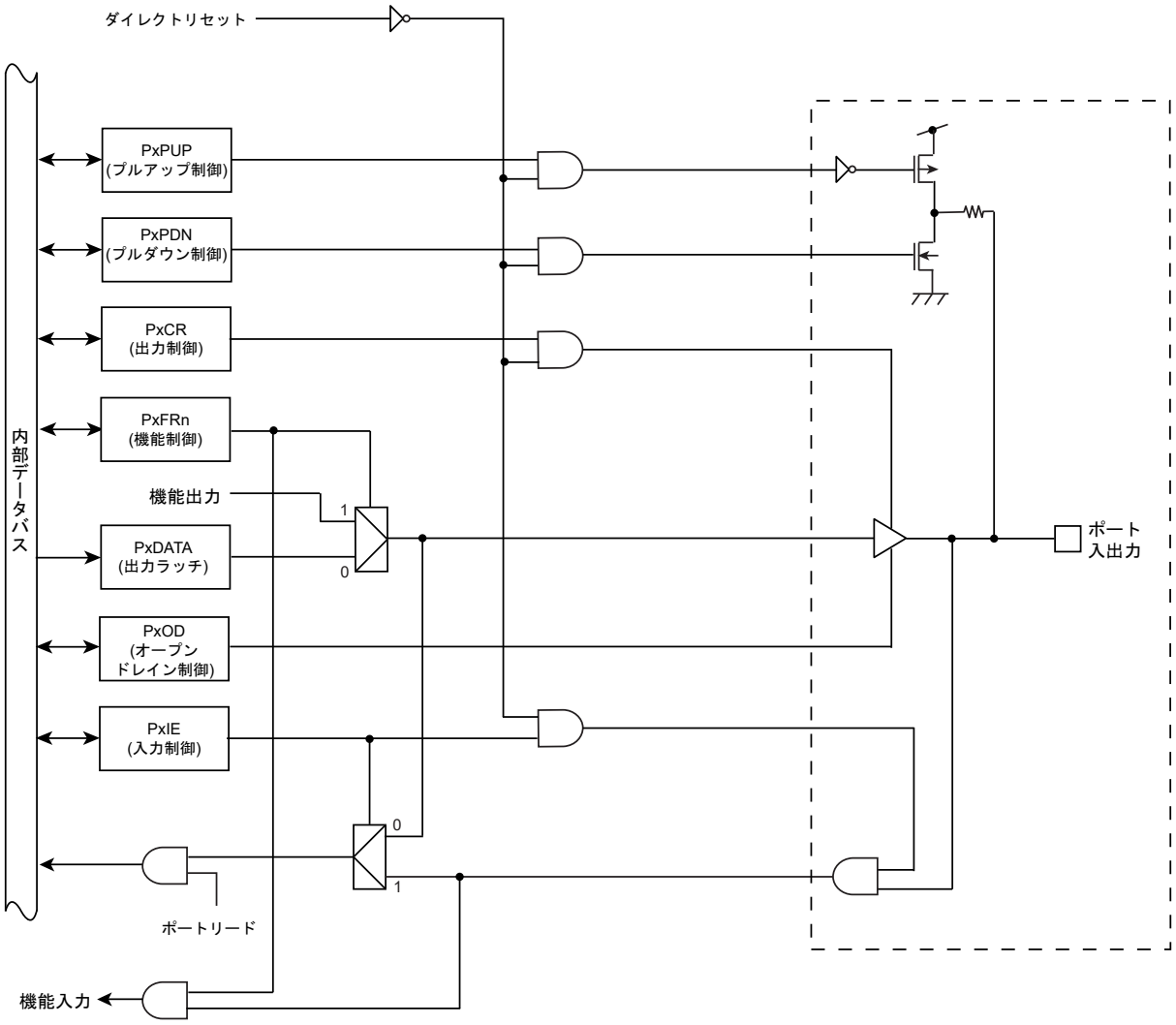


図 11-1 Port Type FT1

11.2.2　タイプ FT2

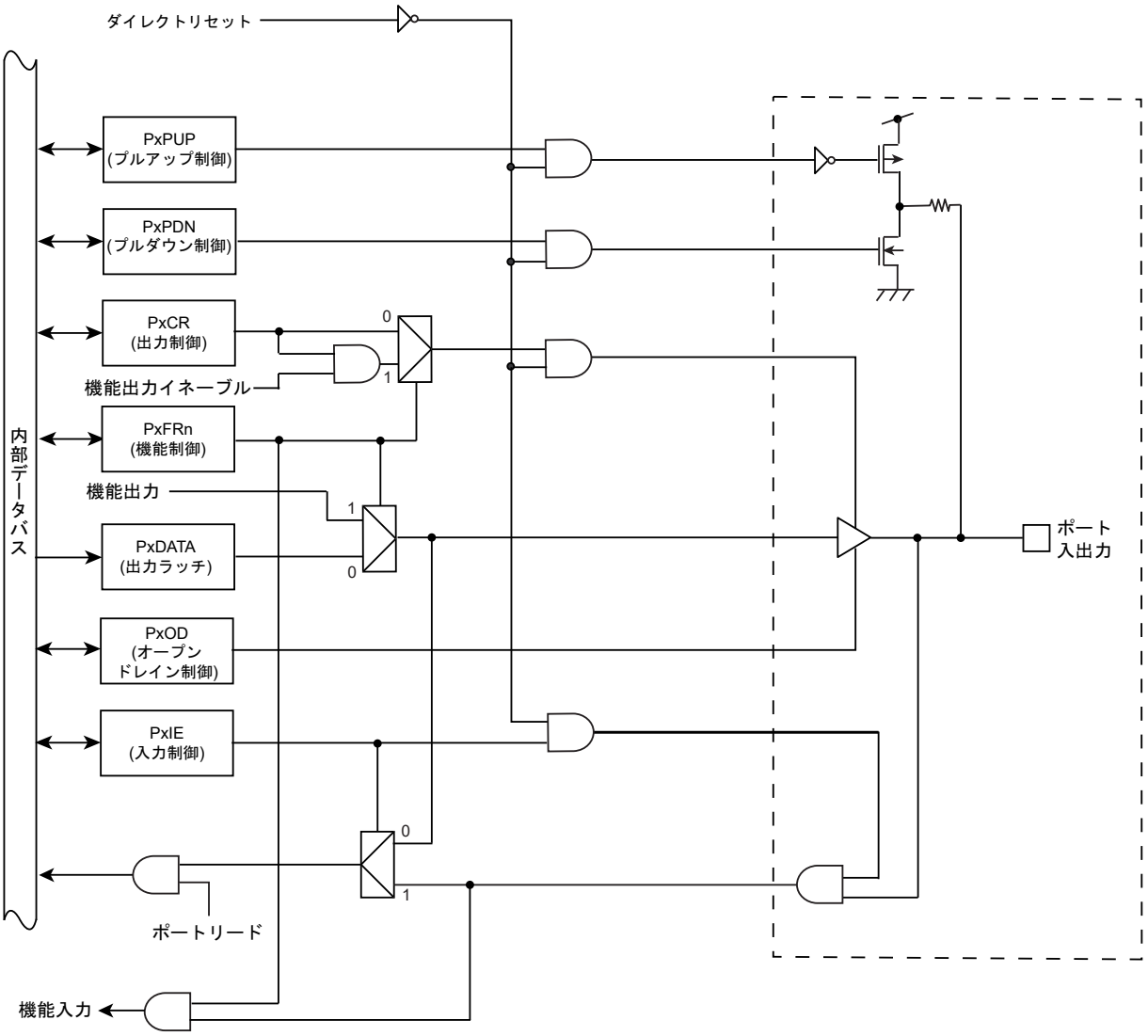


図 11-2 Port Type FT2

11.2.3 タイプ FT3

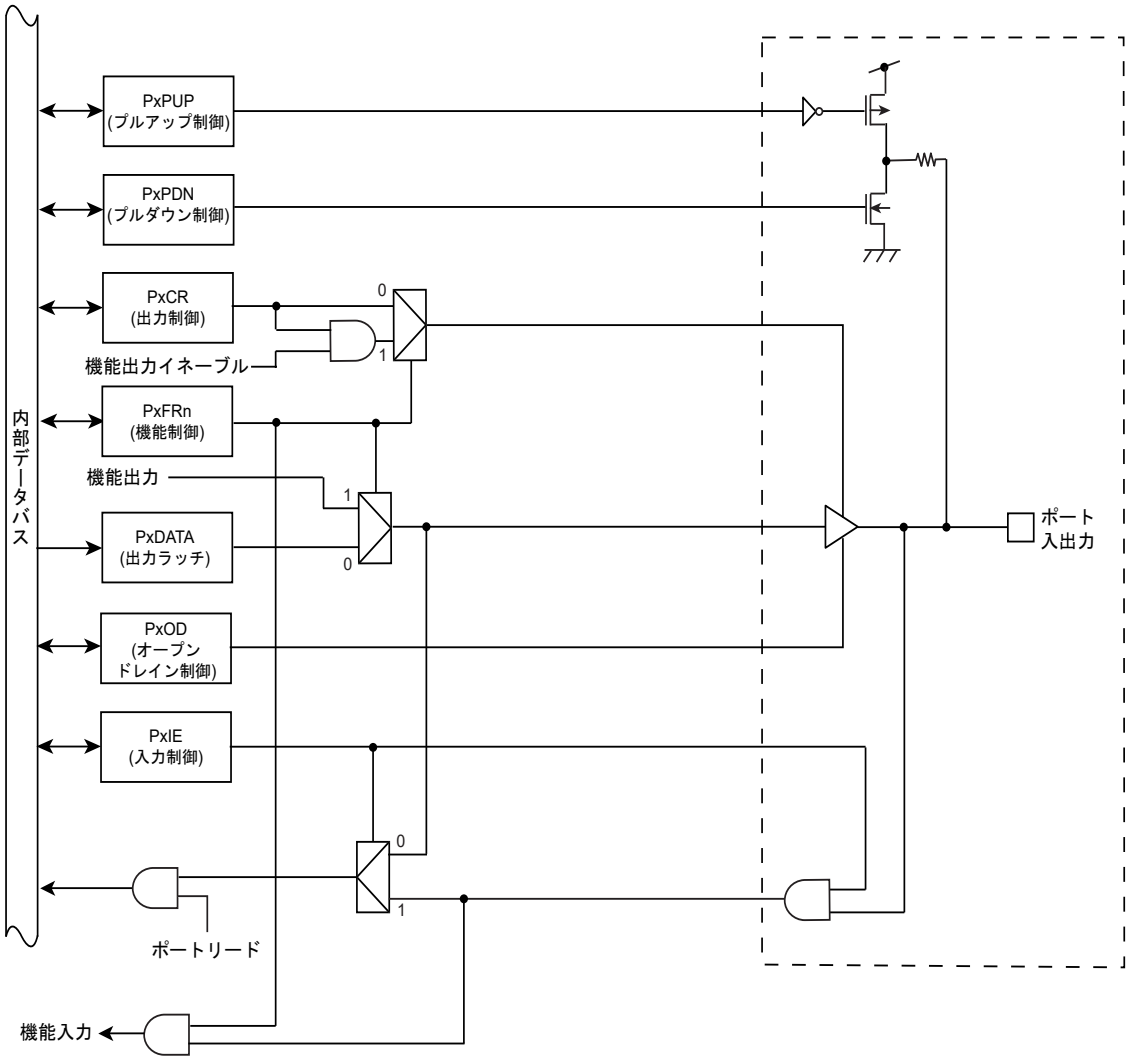


図 11-3 Port Type FT3

11.2.4　タイプ FT4

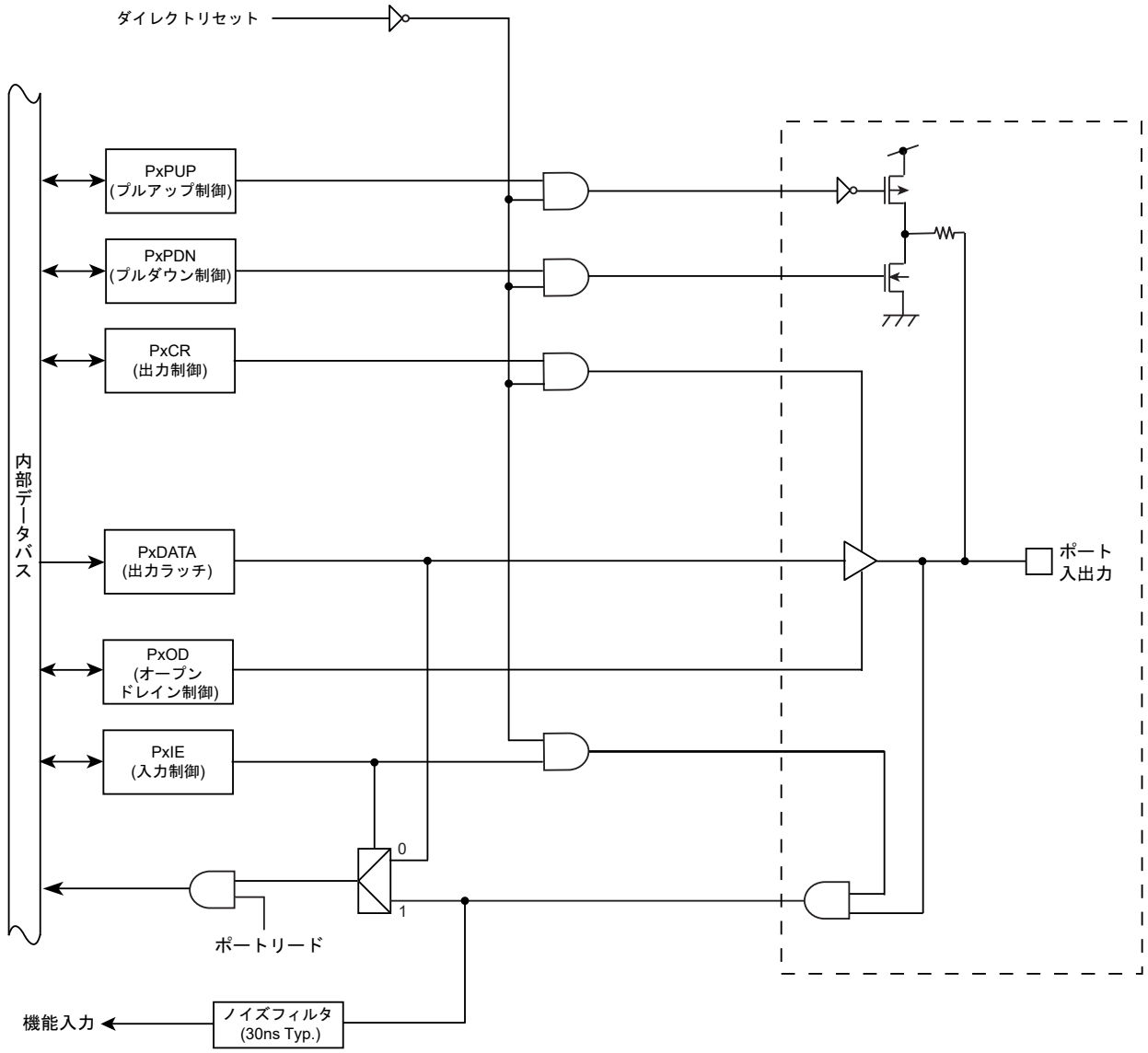


図 11-4 Port Type FT4

11.2.5 タイプ FT5

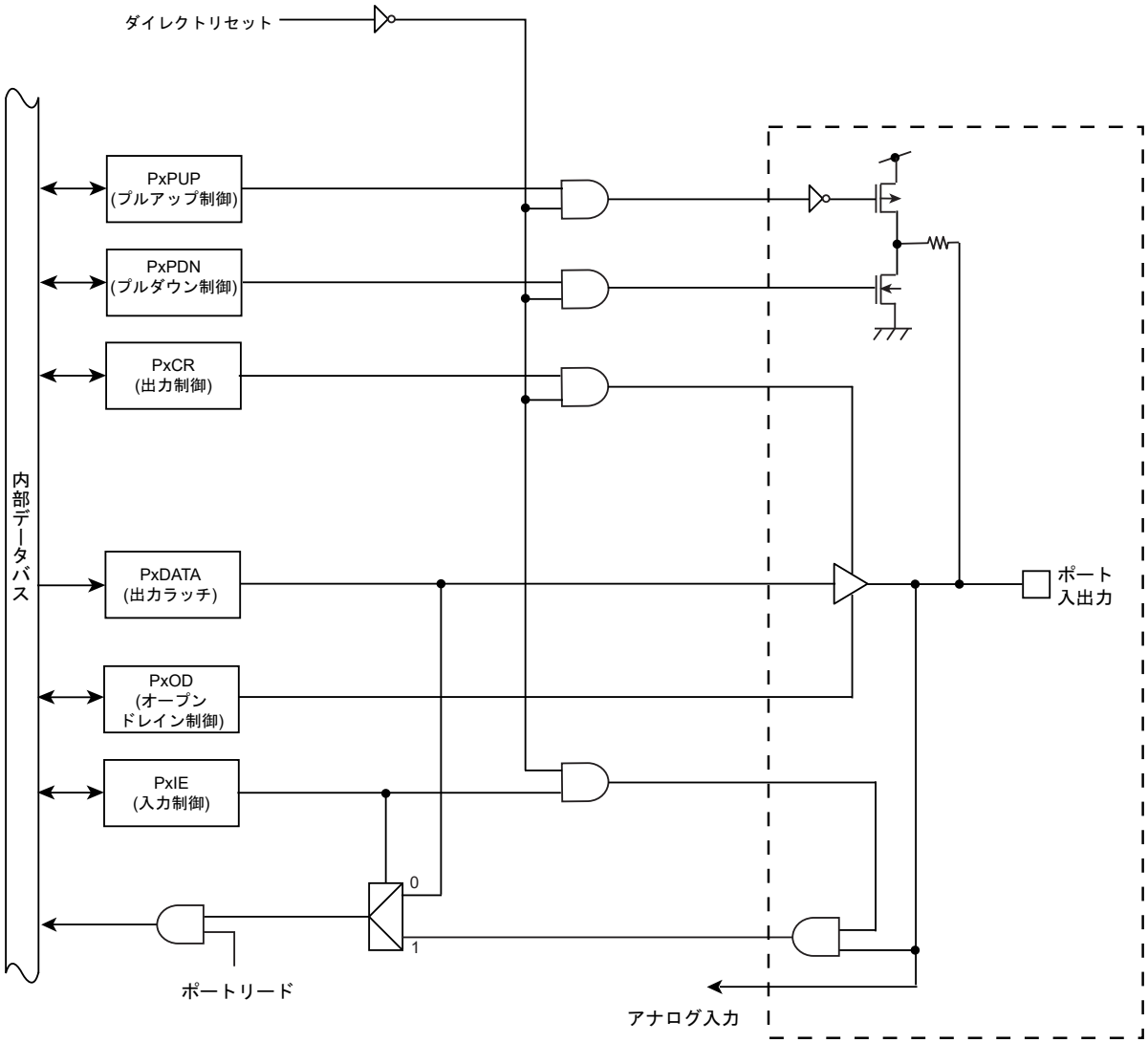


図 11-5 Port Type FT5

11.2.6　タイプ FT6

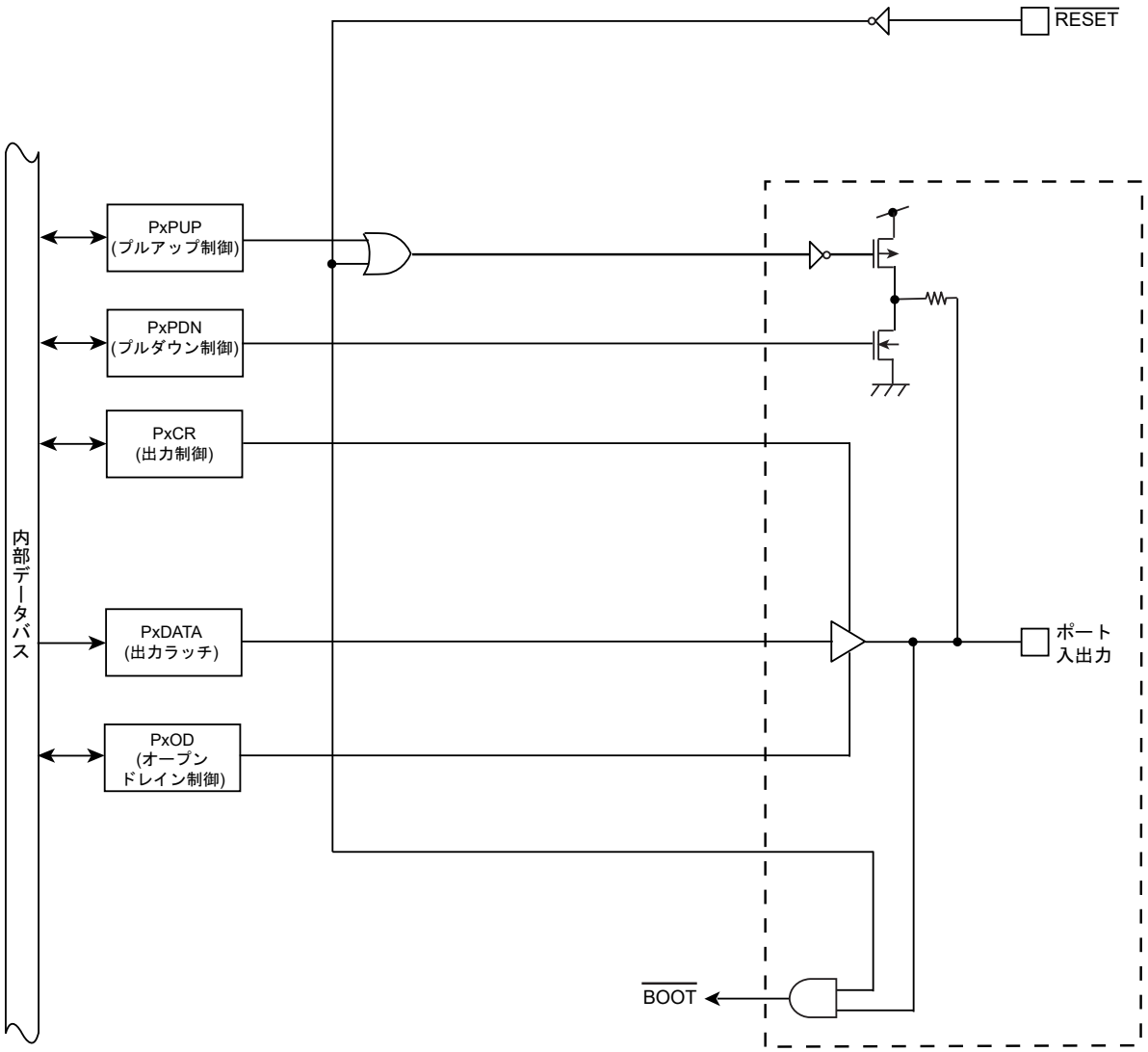


図 11-6 Port Type FT6

11.2.7 タイプ FT7

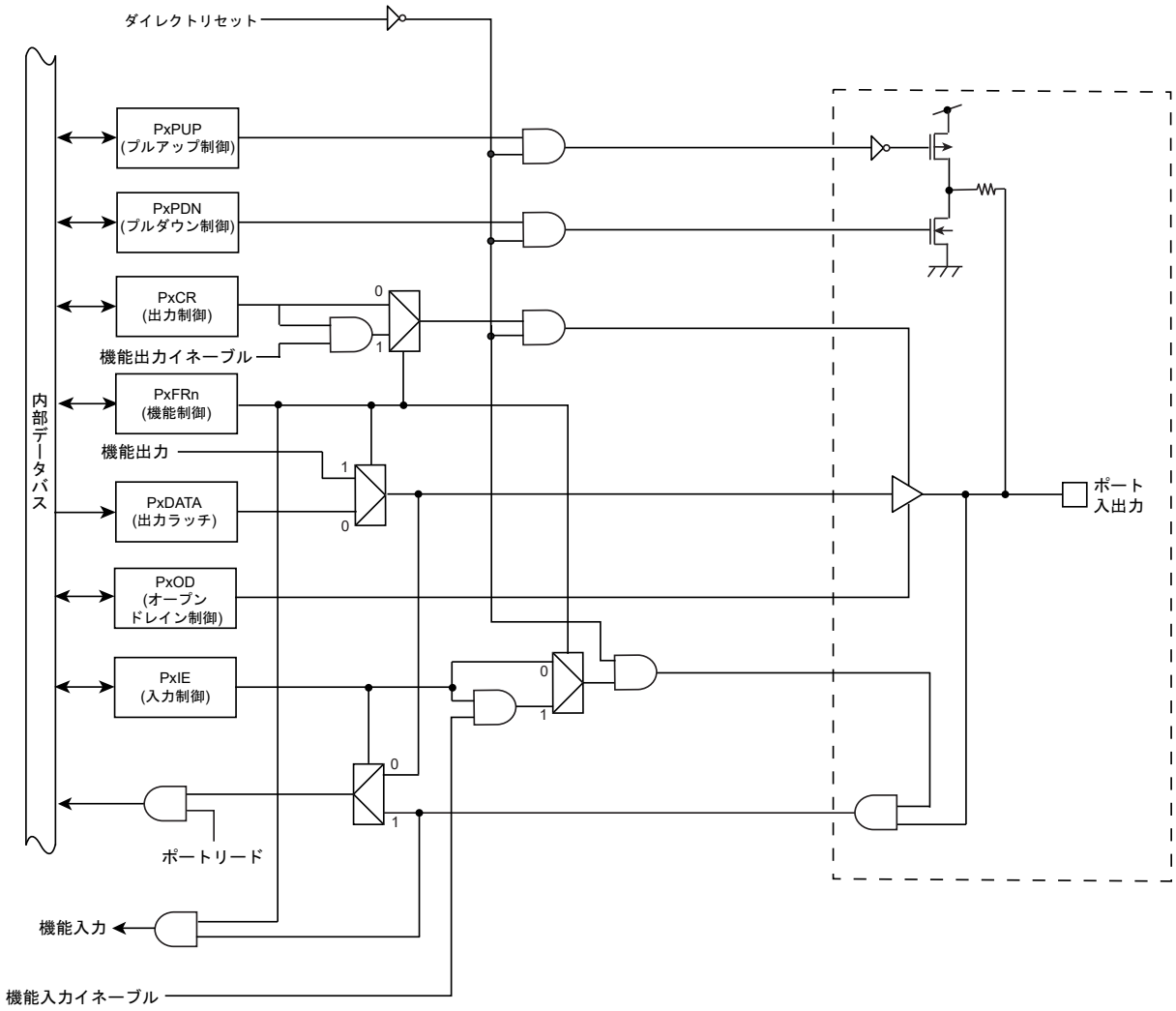


図 11-7 Port Type FT7

11.2.8　タイプ FT8

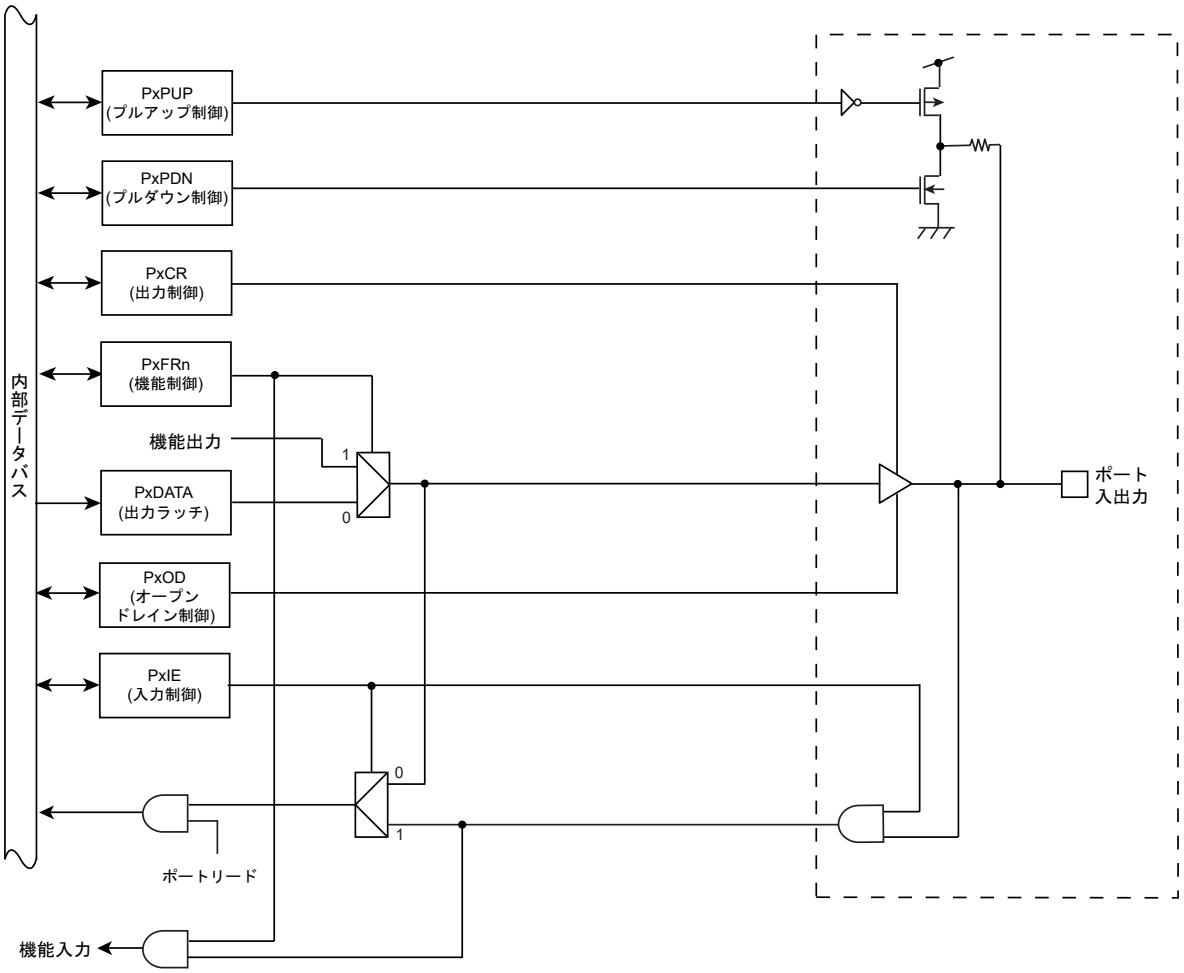


図 11-8 Port Type FT8

第 12 章　16 ビットタイマ/イベントカウンタ(TMRB)

12.1　概要

TMRB は、次の機能をもっています。

- 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
- 外部トリガプログラマブル矩形波出力 (PPG) 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。

- 外部トリガからのワンショットパルス出力
- パルス幅測定

以下の説明中、"x"はチャネル番号を表します。

12.2　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。

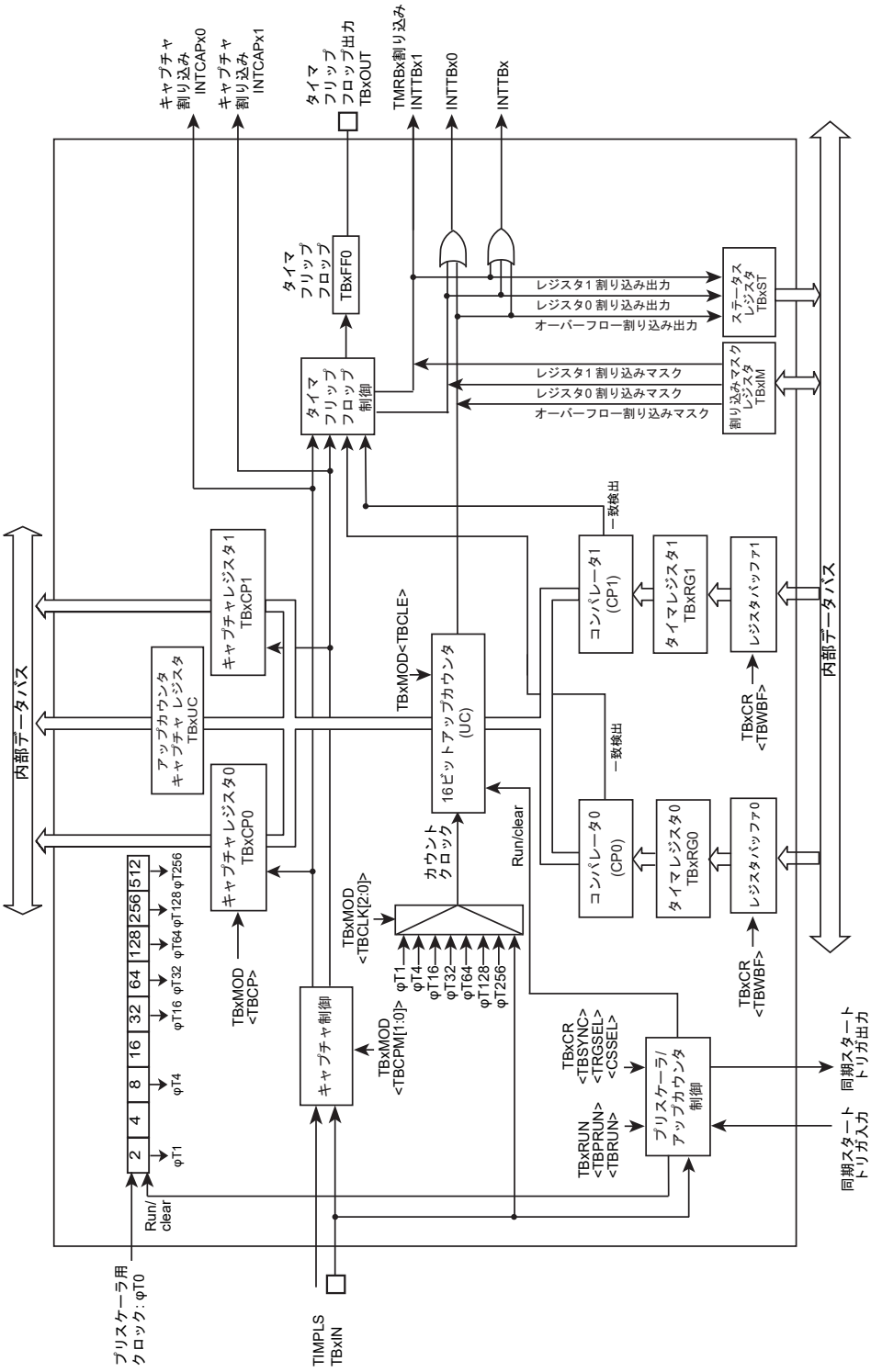


図 12-1 TMRBx ブロック図

12.3 レジスタ説明

12.3.1 レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		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

12.3.2　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	TBHALT	-	-	-	-	-	-
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	TMRBx 動作 0: 禁止 1: 許可 TMRBx の動作を指定します。動作禁止の状態では TMRBx 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)。 TMRBx を使用する場合は、TMRBx モジュールの各レジスタを設定する前に TMRBx 動作許可("1")にしてください。TMRBx 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。
6	TBHALT	R/W	ホールトモード時の制御 0: 動作 1: 停止 デバッグホールト中の動作を指定します。"0”を設定するとデバッグホールト中も動作を継続します。
5-0	−	R	リードすると"0"が読めます。

12.3.3 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	プリスケーラ動作 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

注 1) 外部トリガでカウントをスタートさせる場合は、必ず<TBRUN>=1 に設定してください。

注 2) カウンタ停止状態(<TBRUN>="0")でアップカウンタキャプチャレジスタの<TBUC[15:0]>をリードすると、カウンタ動作時に最後にキャプチャした値がリードされます。

12.3.4　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	-	TRGSEL	CSSEL
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"を書いて下さい。
5	TBSYNC	R/W	同期モード切替 0: 個別動作(チャネルごと) 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	−	R/W	"0"を書いて下さい。
1	TRGSEL	R/W	外部トリガ選択 0: 立ち上がりエッジ 1:立ち下がりエッジ 外部トリガ選択時（TBxIN 端子への信号）のエッジを選択します。
0	CSSEL	R/W	カウンタスタートの方法選択 0: ソフトスタート 1:外部トリガ

注 1)　TBxCR レジスタはタイマ動作中に変更しないでください。

注 2)　外部トリガによるカウントスタート機能を使用する場合は、<CSSEL>と<TRGSEL>を設定した後、<TBRUN>=<TBPRUN>=1 としてください。

12.3.5 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	1	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	TBRSWR	R/W	ダブルバッファ使用時のタイマレジスタ 0,1 への書き込みタイミング制御 0: タイマレジスタ 0、タイマレジスタ 1 への書き込み準備が片側のみしか出来ていない場合でも、 片方ずつタイマレジスタに書き込みを行うことができます。 1: タイマレジスタ 0、タイマレジスタ 1 への書き込み準備が両方とも出来ていない場合は、 タイマレジスタに書き込みを行うことができません。
6	TBCP	W	ソフトウエアキャプチャ制御 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。
5-4	TBCPM[1:0]	R/W	キャプチャタイミング 00: ディセーブル 01: TBxIN↑ TBxIN 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む 10: TBxIN↑ TBxIN↓ TBxIN 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、 TBxIN 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: TIMPLS↑ TIMPLS↓ TIMPLS 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、 TIMPLS 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
3	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。
2-0	TBCLK[2:0]	R/W	TMRBx のソースクロック選択 000: TBxIN 端子入力 001: φT1 010: φT4 011: φT16 100: φT32 101: φT64 110: φT128 111: φT256

注) TBxMOD レジスタはタイマ動作中に変更しないでください。

12.3.6　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マ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マ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)との一致時にタイマ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)との一致時にタイマ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。

注)　TBxFFCR レジスタはタイマ動作中に変更しないでください。

12.3.7 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) フラグは自動的にはクリアされません。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。

12.3.8　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。

注)　TBxIM を設定して割り込みをマスクしても TBxST はセットされます。

12.3.9 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。

注) カウンタ動作時に TBxUC をリードすると、前回リードした時にキャプチャされた値がリードされます。
また、現在のアップカウンタ値がキャプチャされます。

12.3.10　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。

12.3.11　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。

12.3.12 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	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBCP0							
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	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。

12.3.13 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	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBCP1							
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	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。

12.4　回路別の動作説明

12.4.1　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 φT0 は CG 部の CGSYSCR<PRCK[2:0]>で選択された fperiph/1, fperiph/2, fperiph/4, fperiph/8, fperiph/16, fperiph/32 のいずれかのクロックです。このペリフェラルクロック fperiph は CG 部の CGSYSCR<FPSEL>で選択したクロック fgear またはクロックギア分周前のクロック fc 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。

12.4.2　アップカウンタ(UC)

16 ビットのバイナリカウンタです。

12.4.2.1　ソースクロック

ソースクロックは TBxMOD<TBCLK[2:0]>で設定することができます。

プリスケーラ出力クロック φT1, φT4, φT16, φT32, φT64, φT128, φT256、または、TBxIN 入力のいずれかを選択できます。

12.4.2.2　動作の開始と停止

カウンタのスタート方法には、ソフトスタート、外部トリガスタート、同期スタートがあります。

1. ソフトスタート

   TBxRUN<TBRUN> に "1" を設定することでカウントを開始します。"0" でカウント停止と同時にアップカウンタのクリアを行います。

2. 外部トリガスタート

   外部トリガカウントスタートモードでは、外部信号でタイマのカウントスタートが可能となります。

   TBxCR<CSSEL> に"1" を設定することで外部トリガスタートモードとなります。この状態で、<TBRUN> に"1" を設定するとトリガ待ち状態となり、TBxIN0TBxIN の立ちあがりまたは立ち下がりでカウントを開始します。

   TBxCR<TRGSEL> ビットの設定により、外部トリガのエッジ切り替えを行います。

   - <TRGSEL> = "0"：TBxIN の立ち上がりエッジが選択されます。。
   - <TRGSEL> = "1"：TBxIN の立ち下がりエッジが選択されます。。

   TBxRUN<TBRUN> に "0" を設定することでカウント停止と同時にアップカウンタのクリアを行います。

3. 同期スタート

   タイマ同期モードでは、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 出力モードにてタイマ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。

製品によってマスタとなるチャネルとスレーブとなるチャネルの組み合わせは決まっています。本製品でのマスタとスレーブの組み合わせは「製品情報」の章を参照ください。

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。スレーブチャネルの<TBSYNC>ビットに "1"を設定するとマスタチャネルのソフトウエアまたは外部トリガによるスタートに同期してカウント開始および停止します。スレーブチャネルの TBxRUN <TBPRUN, TBRUN> ビットの設定は不要です。マスタチャネルの <TBSYNC> ビットは"0" を設定してください。

なお、外部トリガカウントモードとタイマ同期モードが同時に設定されている場合は、タイマ同期モードが優先されます。

12.4.2.3 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。

12.4.2.4 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 INTTBx0 が発生します。

12.4.3 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 へ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。

12.4.4 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<TBCPM[1:0]>で設定します。

また、ソフトウエ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<TBCP> 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。

12.4.5　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。

12.4.6　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。

12.4.7　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、INTTBx0 と INTTBx1 を発生します。

12.4.8　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。

12.4.9　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。

12.5 モード別動作説明

12.5.1 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し、INTTBx1 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	←	*	*	*	*	*	*	*	*	INTTBx1 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	TBxFF0 反転トリガをディセーブルします。
TBxMOD	←	0	1	0	0	1	*	*	*	入力クロックをプリスケー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。
		(*** = 001, 010, 011,100,101,110,111)								
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

12.5.2 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(TBxIN 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 TBxIN 端子入力の立ち上がりエッジでカウントアップします。ソフトウエ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
PORT 関連レジスタ設定										該当ポートを TBxIN になるように設定します。
TBxFFCR	←	X	X	0	0	0	0	1	1	TBxFF0 反転トリガをディセーブルします。
TBxMOD	←	0	1	0	0	0	0	0	0	入力クロックを TBxIN にします。
TBxRUN	←	*	*	*	*	*	1	X	1	TMRBx を起動します。
TBxMOD	←	0	0	0	0	0	0	0	0	ソフトウエアキャプチャを行います。

注) X; Don't care −; No change

12.5.3　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 (TBxFF0) 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 設定値 < TBxRG1 設定値

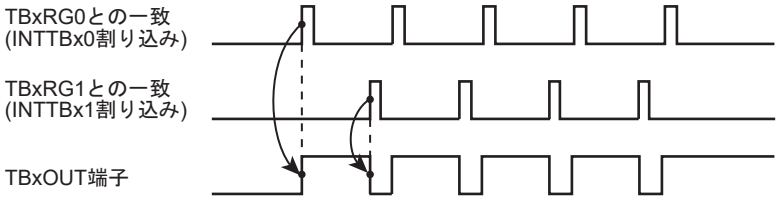


図 12-2 プログラマブル矩形波(PPG)出力波形例

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。

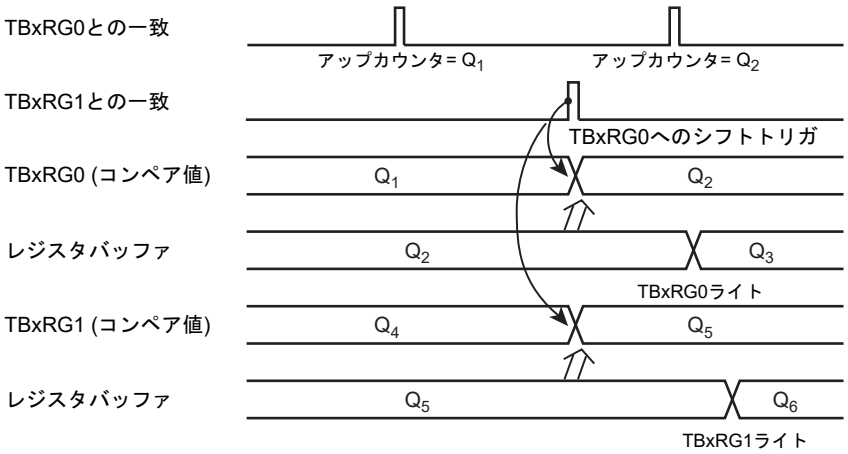


図 12-3 レジスタバッファの動作

このモードのブロック図を示します。

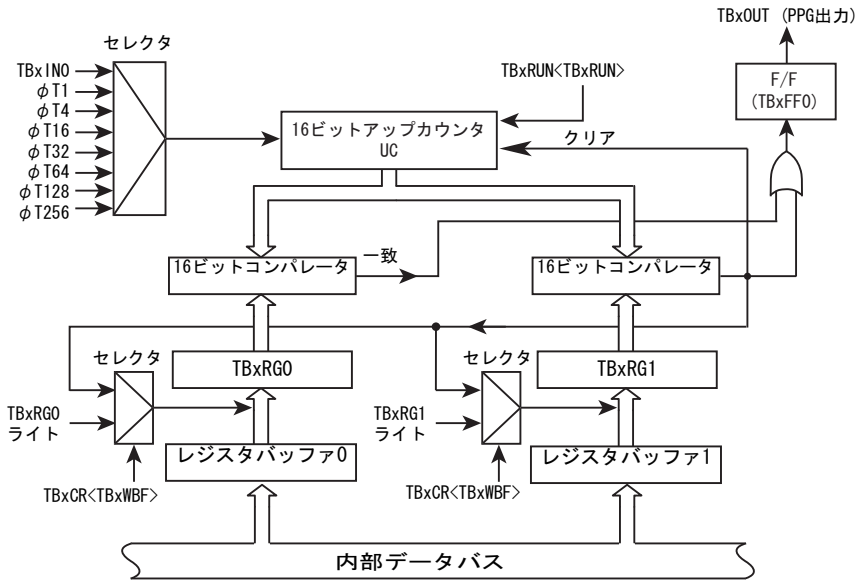


図 12-4 16 ビット PPG モードのブロック図

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	0	X	−	0	0	0	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	0	X	−	0	0	0	TBxRG0 のダブルバッファイネーブル (INTTBx0 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	←	0	1	0	0	1	*	*	*	入力クロックをプリスケー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。 (*** = 001, 010, 011,100,101,110,111) UC は TBxRG1 との一致でクリアされます。
PORT 関連レジスタ設定										該当ポートを TBxOUT になるように設定します。
TBxRUN	←	*	*	*	*	*	1	X	1	TMRBx を起動します。

注) X; Don't care
−; No change

12.5.4　外部トリガ PPG(プログラマブル矩形波)出力モード

外部トリガカウントスタートを使用すると、小さいディレイでのワンショットパルス出力が可能です。

外部トリガカウントスタートを使用したワンショットパルス出力(ディレイあり）の例を以下に示します。

16 ビットアップカウンタ(UC)が停止状態(TBxRUN<TBRUN>=0)で、TBxIN 端子の立ち上がりでカウントアップするように設定しておきます(TBxCR[1:0] = "01")。タイマレジスタ(TBxRG0）には、ディレイタイム(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幅(p)を加算した値(d+p)を設定します。

TBxFFCR<TBE1T1, TBE0T1>に"11"を設定し、UC と TBxRG0 との一致、および、TBxRG1 との一致より、タイマフリップフロップ(TBxFF0)が反転するようにトリガをイネーブルにします。

TBxRUN<TBRUN>を"1"にセットし、外部トリガパルスの立ち上がりで、カウントアップスタートを有効にします。

TBxIN 端子の立ち上がりでワンショットパルスが出力された後、INTTBx1 の割り込み処理でタイマフリップフロップ(TBxFF0)の反転をディセーブルにするか、TBxRUN<TBRUN>を"0"にクリアし、16 ビットアップカウンタの動作を停止します。

なお、文中の(d)、(p)は図 12-5 の d、p と対応しています。

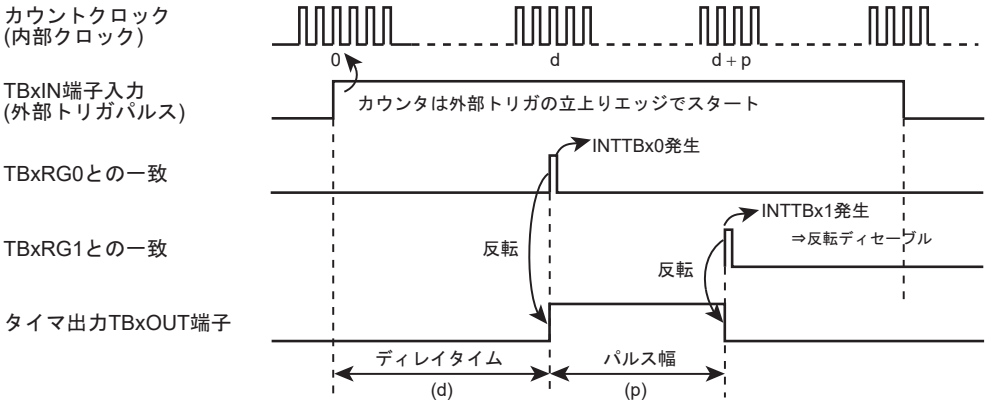


図 12-5 外部トリガカウントスタートを使用したワンショットパルス出力(ディレイあり)

12.6 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. パルス幅測定

12.6.1 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 をプリスケー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。TBxIN 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、UC 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、INTTBx0/INTTBx1 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 12-6 の c, d, p と対応しています。

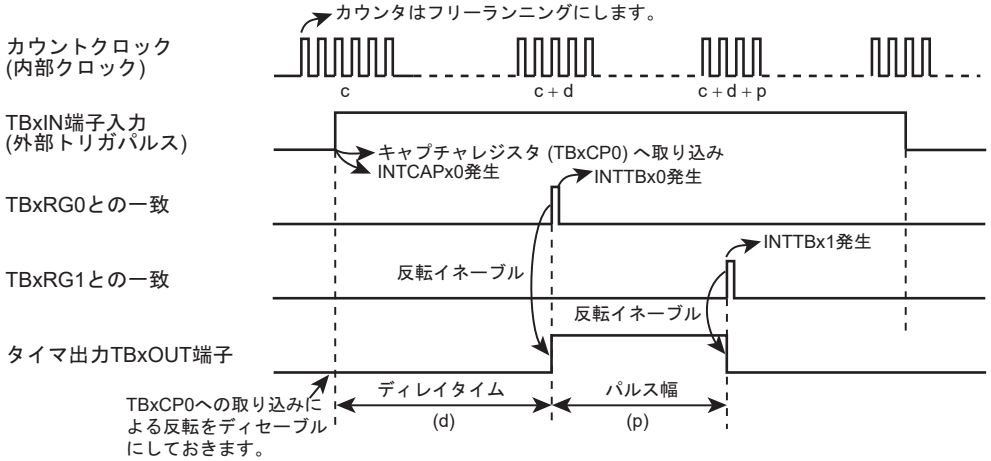


図 12-6 ワンショットパルス出力(ディレイあり)

TBxIN 入力の立ち上がりをトリガとして、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 ΦT1 を使用しています。

		7	6	5	4	3	2	1	0	
[メイン処理]　TBxIN でのキャプチャ設定										
PORT 関連レジスタ設定										該当ポートを TBxIN になるように設定します。
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	←	0	1	0	1	0	0	0	1	ソースクロックを ΦT1 にし、TBxIN 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	TBxFF0 反転トリガをクリアし、ディセーブルします。
PORT 関連レジスタ設定										該当ポートを TBxOUT になるように設定します。
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 + 3μσ/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	←	*	*	*	*	*	*	*	*	INTTBx1 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。
[INTTBx1 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	−	−	TBxFF0 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)　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、INTTBx1 割り込みでこれをディセーブルに戻します。

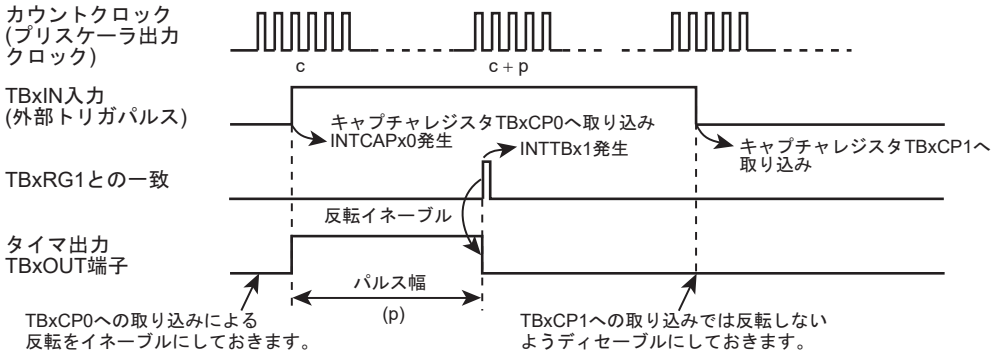


図 12-7 外部トリガパルスのワンショットパルス出力(ディレイなし)

12.6.2 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。TBxIN 端子より外部パルスを入力し、アップカウンタ (UC) をプリスケー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。TBxIN 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、プリスケーラ出力クロックの周期が 0.5 μs であれば、パルス幅は、100 × 0.5 μs = 50 μs 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、ソフトウエ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 12-8 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、プリスケー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。

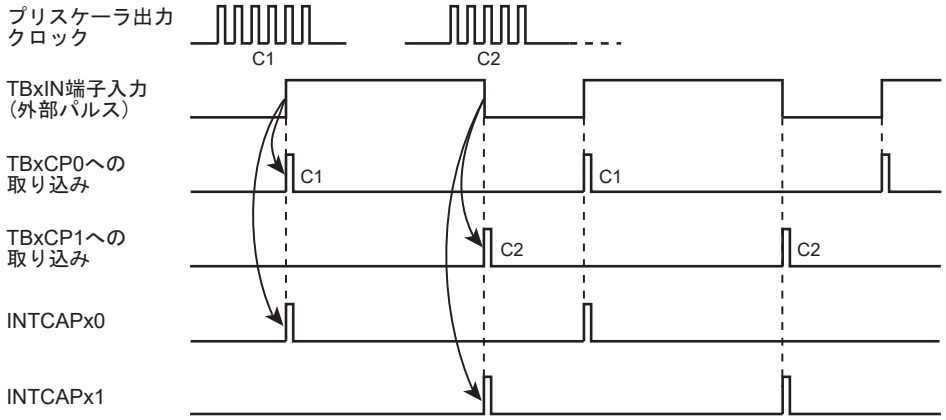


図 12-8 パルス幅測定

第 13 章　4 バイト FIFO 付きシリアルチャネル(SIO/UART)

13.1　概要

シリアルチャネル(SIO/UART)は次の動作モードを持っています。

- 同期通信モード(I/O インタフェースモード)
- 非同期通信モード(UART モード)

特長は以下のとおりです。

- 転送クロック
  - プリスケーラでペリフェラルクロック(φT0)を 1/1、1/2、1/4、1/8、1/16、1/32、1/64、1/128 分周
  - プリスケーラ出力クロックに対し、1~16 分周が可能
  - プリスケーラ出力クロックに対し、N + m/16 (N = 2~15, m = 1~15)分周が可能

    (UART モードのみ)
  - システムクロック(fsys)を使用可能(UART モードのみ)
- バッファ
  - ダブルバッファ構成で使用可能
  - 送信バッファのクリアが可能
- FIFO

  送受信合わせて 4 バイトの FIFO を使用可能
- I/O 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：立ち上がり/立ち下がりエッジ選択
  - 連続転送時のインタバル時間設定が可能
  - 最終ビット出力後の SCxTXD 端子の状態を下記から選択可能

    "High"保持/"Low"保持/最終ビット保持
  - クロック入力モード時、アンダーランエラーが発生したときの SCxTXD 端子の状態を下記から選択可能

    "High"保持/"Low"保持
  - クロック入力モード時、SCxTXD 端子の最終ビットホールド時間を設定可能
- 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
  - $\overline{\text{SCxCTS}}$ 端子を用いたハンドシェイク機能
  - SCxRXD 端子へのノイズキャンセラ付加

以下の説明中、"x"はチャネル番号をあらわします。

13.2　構成

下記にシリアルチャネルとシリアルクロック生成回路のブロック図を示します。

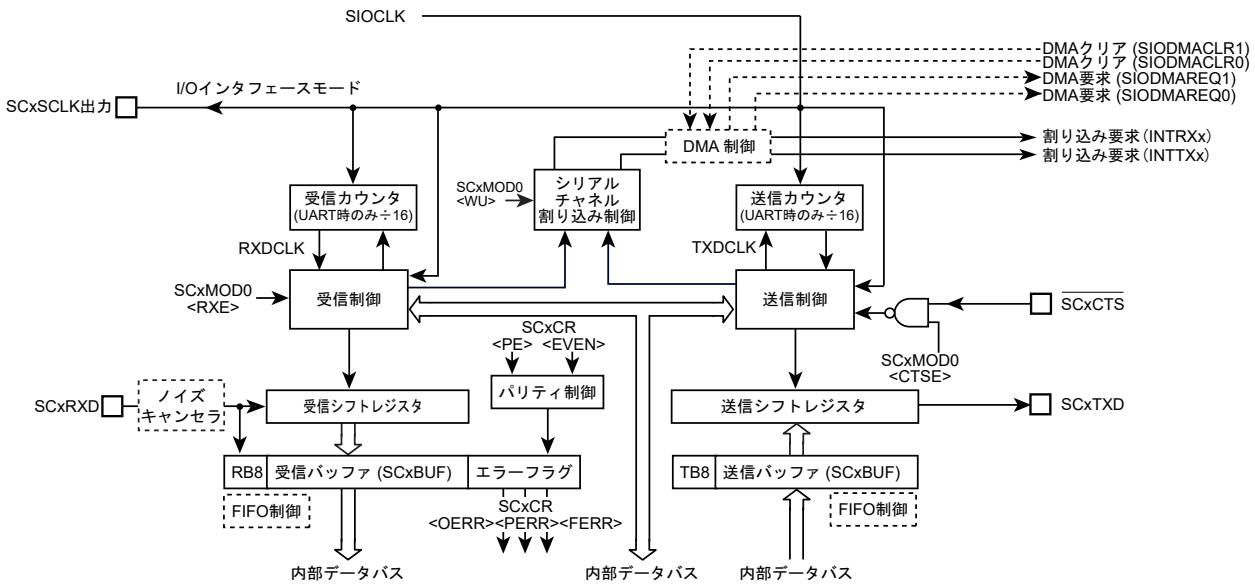


図 13-1 シリアルチャネルブロック図

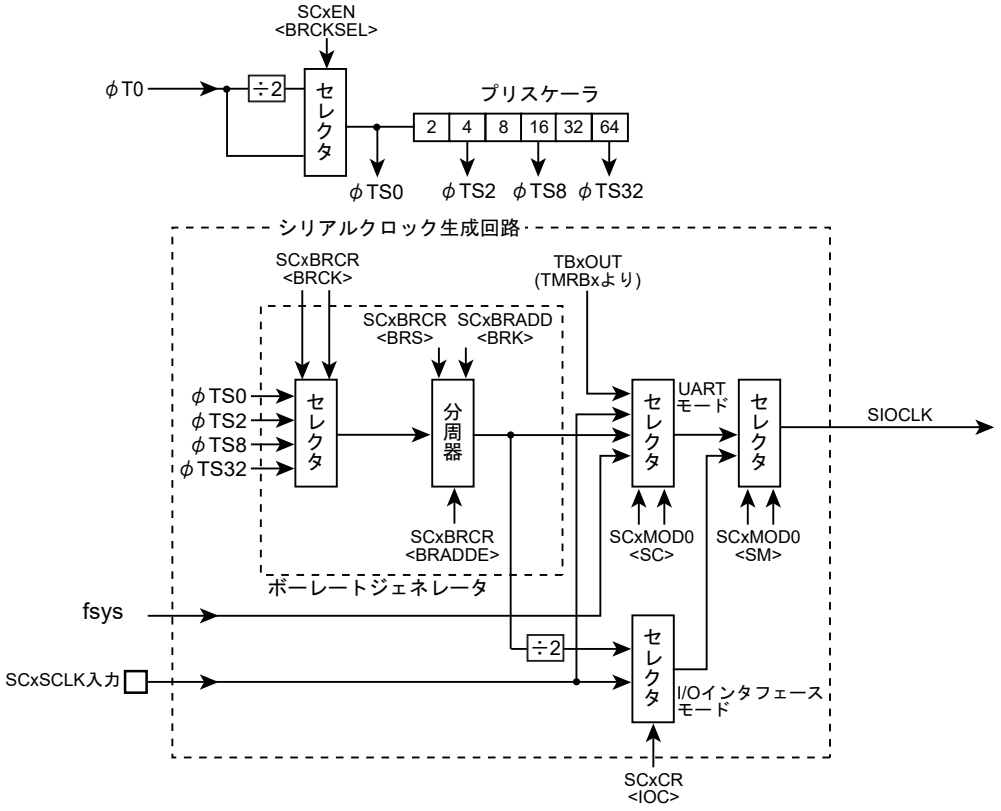


図 13-2 シリアルクロック生成回路ブロック図

13.3 レジスタ説明

13.3.1 レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		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、受信中にコントロールレジスタを書き換えないでください。

13.3.2　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	-	-	-	-	-	-	BRCKSEL	SIOE
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	BRCKSEL	R/W	プリスケーラの入力クロックを選択します。 0: φT0/2 1: φT0
0	SIOE	R/W	シリアルチャネル動作 0: 禁止 1: 動作 シリアルチャネルの動作を指定します。シリアルチャネルを使用する場合は、まず<SIOE>に"1"をセットしてください。 動作禁止の状態では、イネーブルレジスタを除くシリアルチャネル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。 シリアルチャネルを一旦動作させた後に動作禁止にした場合は、レジスタの設定は保持されます。

13.3.3 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：送信用バッファまたは FIFO [リード] RB：受信用バッファまたは FIFO

13.3.4　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	-	EHOLD			-	TXDEMP	TIDLE	
リセット後	0	0	0	0	0	1	1	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
31-15	−	R	リードすると"0"が読めます。
14-12	EHOLD[2:0]	R/W	クロック入力モードの SCxTXD 端子の最終ビットホールド時間の設定(I/O インタフェースモードのみ有効) 最終ビットのホールド時間 ≤ SCLK 周期/2 となるように、最終ビットのホールド時間、SCLK 周期を設定してください。 000: 2/fsys　100: 32/fsys 001: 4/fsys　101: 64/fsys 010: 8/fsys　110: 128/fsys 011: 16/fsys　111: Reserved
11	−	R	リードすると"0"が読めます。
10	TXDEMP	R/W	クロック入力モード時、アンダーランエラーが発生したときの SCxTXD 端子の状態選択(I/O インタフェースモードのみ有効) 0: "Low"出力 1: "High"出力
9-8	TIDLE[1:0]	R/W	最終ビット出力後の SCxTXD 端子の状態選択(I/O インタフェースモードのみ有効) <TIDLE[1:0]>="10"を設定したときには、<EHOLD[2:0]>="000"を設定してください。 00: "Low"出力保持 01: "High"出力保持 10: 最終ビット保持 11: Reserved
7	RB8	R	受信データビット 8 (UART モードのみ有効) 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 モードのみ有効) パリティの条件を設定します。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。 0: Odd 1: Even
5	PE	R/W	パリティ付加(UART モードのみ有効) 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

Bit	Bit Symbol	Type	機能
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	クロックエッジ選択(I/O インタフェース用) 0: SCxSCLK 端子の立ち下がりエッジで送信バッファのデータを 1bit ずつ SCxTXD 端子へ出力します。 SCxSCLK 端子の立ち上がりエッジで SCxRXD 端子のデータを 1bit ずつ受信バッファに取り込みます。 この時、SCxSCLK 端子は High レベルからスタートします(立ち上がりモード)。 1: SCxSCLK 端子の立ち上がりエッジで送信バッファのデータを 1bit ずつ SCxTXD 端子へ出力します。 SCxSCLK 端子の立ち下がりエッジで SCxRXD 端子のデータを 1bit ずつ受信バッファに取り込みます。 この時、SCxSCLK 端子は Low レベルからスタートします。(立ち下りモード)
0	IOC	R/W	クロック選択(I/O インタフェースモード用) 0: クロック出力モード (転送クロックは SCxSCLK 端子から出力されます) 1: クロック入力モード (転送クロックは SCxSCLK 端子から入力されます)

注) <OERR>, <PERR>, <FERR>は読み出すとクリアされます。

13.3.5　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 モードのみ) 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 モードのみ) 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。 "1"を設定すると $\overline{\text{SCxCTS}}$ 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 1)(注 2) 0: 禁止 1: 許可
4	WU	R/W	ウェイクアップ機能 (UART モードのみ) 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: I/O インタフェースモード 01: 7 ビット UART モード 10: 8 ビット UART モード 11: 9 ビット UART モード
1-0	SC[1:0]	R/W	シリアル転送クロック(UART モードのみ) 00: TMRB 出力 01: ボーレートジェネレータ 10: システムクロック(fsys) 11: 外部クロック(SCxSCLK 端子入力) (I/O インタフェースモード時の転送クロックは、SCxCR<IOC>で選択します。)

注 1)　<RXE>は、すべての設定が終わった後に許可してください。
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。

13.3.6 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 I/O インタフェースモード時の転送モードを設定します。 また FIFO が許可されている場合は、FIFO 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 1) (注 2) 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バル時間(I/O インタフェースモード用) 000: なし 001: 1 x SCLK 周期 010: 2 x SCLK 周期 011: 4 x SCLK 周期 100: 8 x SCLK 周期 101: 16 x SCLK 周期 110: 32 x SCLK 周期 111: 64 x SCLK 周期 I/O インタフェースモードでクロック出力モード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。 I/O 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(SCxMOD1<TXE>を"0"にクリア)しないでください。

13.3.7　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	RBFLL	TXRUN	SBLEN	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	RBFLL	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、受信バッファを読み出すと"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。

<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	SBLEN	R/W	送信 STOP ビット長(UART モード用) 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 STOP ビット長は 1 ビットであるとして動作します。
3	DRCHG	R/W	転送方向設定 0: LSB first 1: MSB first 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。
2	WBUF	R/W	ダブルバッファの許可 0: 不許可 1: 許可 I/O インタフェースモードの送信(クロック出力/入力モード), 受信(クロック出力モード), 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。 I/O インタフェースモードの受信(クロック入力モード), 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。

Bit	Bit Symbol	Type	機能
1-0	SWRST[1:0]	R/W	ソフトウエアリセット "10"→"01" の順に書き込むことでソフトウエアリセットが発生します。 ソフトウエアリセットにより、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)

レジスタ名	ビット
SCxMOD0	<RXE>
SCxMOD1	<TXE>
SCxMOD2	<TBEMP>, <RBFLL>, <TXRUN>
SCxCR	<OERR>, <PERR>, <FERR>

注 1) 転送動作中にソフトウエア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) ソフトウエアリセット動作が完了するのに、命令実行後 2 クロック必要です。

13.3.8　SCxBRCR (ボーレートジェネレータ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	-	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 モードのときのみ) 0: ディセーブル 1: イネーブル
5-4	BRCK[1:0]	R/W	ボーレートジェネレータ入力クロック選択 00: φTS0 01: φTS2 10: φTS8 11: φTS32
3-0	BRS[3:0]	R/W	分周値"N"の設定 0000 : N = 16 0001 : N = 1 0010 : N = 2 : 1111 : N = 15

注 1)　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 2)　I/O 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。

13.3.9 SCxBRADD (ボーレートジェネレータ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	-	-	-	-	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 モード用) 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 13-1 にまとめます。

表 13-1 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	
<BRK>の設定	設定不要	"K"値を設定 (注 2)
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) "K"値に"0"を設定することはできません。

13.3.10　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	RXTXCNT	CNFG
リセット後	0	0	0	0	0	0	0	0

<table>
<tr><th>Bit</th><th>Bit Symbol</th><th>Type</th><th>機能</th></tr>
<tr><td>31-8</td><td>-</td><td>R</td><td>リードすると"0"が読めます。</td></tr>
<tr><td>7-5</td><td>-</td><td>R/W</td><td>必ず"000"を書き込んでください。</td></tr>
<tr><td>4</td><td>RFST</td><td>R/W</td><td>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。</td></tr>
<tr><td>3</td><td>TFIE</td><td>R/W</td><td>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。</td></tr>
<tr><td>2</td><td>RFIE</td><td>R/W</td><td>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。</td></tr>
<tr><td>1</td><td>RXTXCNT</td><td>R/W</td><td>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>
<tr><td>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 に指定された有効バイト数分の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。</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(<TXE>)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></td></tr>
<tr><td>0</td><td>CNFG</td><td>R/W</td><td>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>
<tr><td>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></td></tr>
</table>

注 1) 送信 FIFO は常に構成されている FIFO の最大バイト数(<CNFG>の説明を参照)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。

13.3.11　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、SCxRST<RLVL>="000"となります。また、リードポインタも初期化されます。 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: FIFO fill レベル(SCxRST<RLVL[2:0]>) = 割り込み発生 fill レベル(<RIL[1:0]>) のとき 1: FIFO fill レベル(SCxRST<RLVL[2:0]>) ≥ 割り込み発生 fill レベル(<RIL[1:0]>) のとき 割り込み発生タイミングの詳細については、13.13.1.2 を参照してください。
5-2	−	R	リードすると"0"が読めます。
1-0	RIL[1:0]	R/W	受信割り込みが発生する受信 FIFO の fill レベル

	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ください。

13.3.12 SCxTFC (送信 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	-	-	-	-	-	-	-	TBCLR
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
31-9	–	R	リードすると"0"が読めます。
8	TBCLR	W	送信バッファクリア 0: Don't care 1: クリア "1"を書き込むと送信バッファがクリアされます。 リードすると"0"が読めます。
7	TFCS	W	送信 FIFO クリア(注 1) 0: Don't care 1: クリア "1"を書き込むと送信 FIFO がクリアされ、SCxTST<TLVL>="000"となります。また、ライトポインタも初期化されます。 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: FIFO fill レベル(SCxTST<TLVL[2:0]>) = 割り込み発生 fill レベル(<TIL[1:0]>) のとき 1: FIFO fill レベル(SCxTST<TLVL[2:0]>) ≤ 割り込み発生 fill レベル(<TIL[1:0]>) のとき 割り込み発生タイミングの詳細については、13.13.2.2 を参照してください。
5-2	–	R	リードすると"0"が読めます。
1-0	TIL[1:0]	R/W	送信割り込みが発生する FIFO の fill レベル

	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ください。

注 2) SCxEN<SIOE>=0(SIO/UART 動作禁止)、または SCxMOD1<I2SC>=0 で IDLE モードへの移行(IDLE モード中の動作停止) した場合は、必ず SCxTFC の再設定を行ってください。

13.3.13　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>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。

13.3.14 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	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	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>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。

13.4　動作モード

表 13-2 にモードをまとめます。

表 13-2 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 (I/O 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、I/O を拡張するために使用できます。SCLK クロックに同期してデータの送受信を行います。SCLK クロックはクロック入力/出力モードのいずれでも使用できます。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 のみ選択できます。

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。

13.5 データフォーマット

13.5.1 データフォーマット一覧

図 13-3 にデータフォーマットを示します。

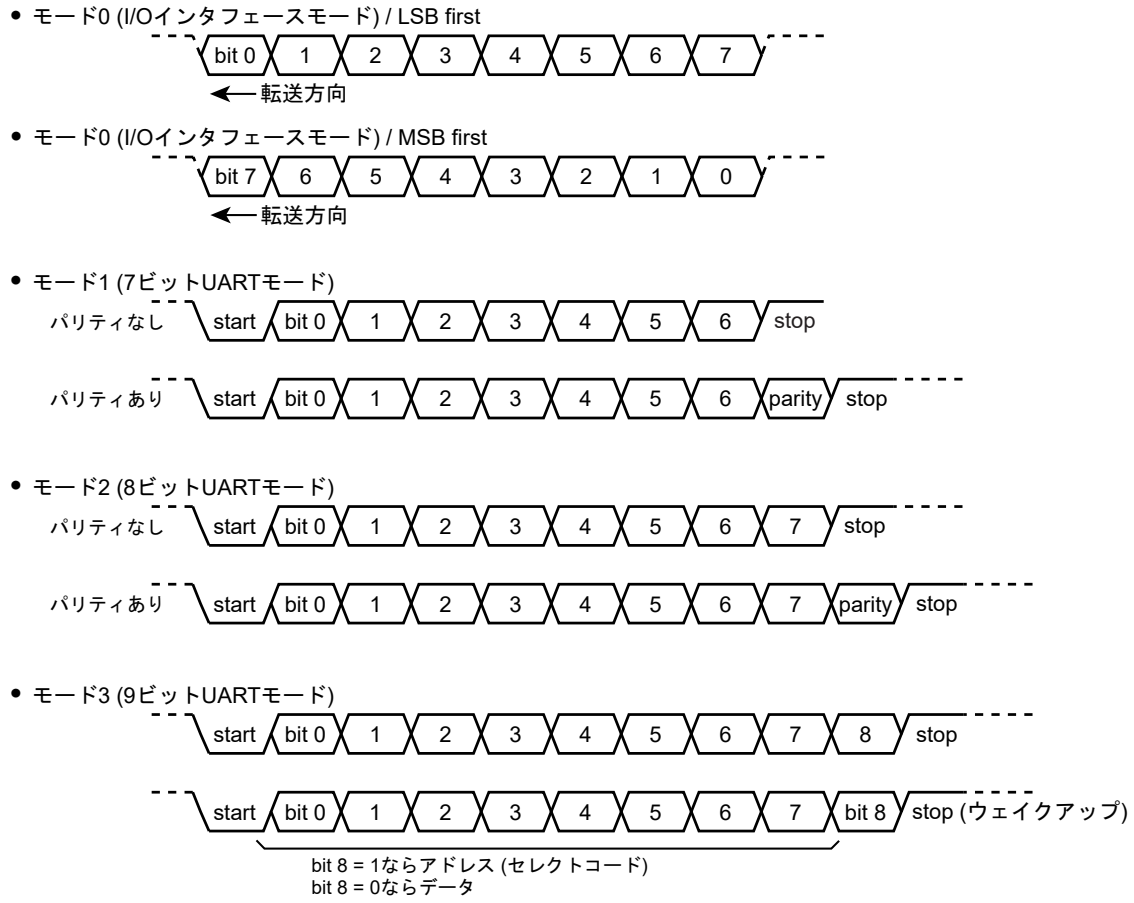


図 13-3 データフォーマット

13.5.2　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。また、受信データに付加されているパリティビットを生成されたパリティビットと比較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。

13.5.2.1　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 SCxMOD0<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。

13.5.2.2　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。

13.5.3　STOP ビット長

SCxMOD2<SBLEN>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。

13.6 クロック制御

13.6.1 プリスケーラ

7 ビットのプリスケーラを実装しており、φT0 の 1 / 2/ 4 / 8 / 16 / 32/ 64 / 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は、クロック/モード制御部の CGSYSCR と SCxEN<BRCKSEL>で選択します。

プリスケーラは、SCxMOD0<SC[1:0]>="01"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。

13.6.2 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。

13.6.2.1 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、プリスケーラ出力の 1 / 4 / 16 / 64 分周から選択します。入力クロックの選択は SCxEN<BRCKSEL>と SCxBRCR<BRCK>で行います。

SCxEN<BRCKSEL>	SCxBRCR<BRCK>	ボーレートジェネレータ入力 φTx
0	00	φT0/2
0	01	φT0/8
0	10	φT0/32
0	11	φT0/128
1	00	φT0
1	01	φT0/4
1	10	φT0/16
1	11	φT0/64

(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。

I/O 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[3:0]>	K 値 SCxBRADD<BRK[3:0]>
I/O 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。

ボーレートジェネレータ分周機へのクロック入力を φTx とした時、N 分周の場合と N + (16-K)/16 分周の場合のボーレートジェネレータ出力クロックは以下の計算式であらわされます。

・　N 分周

$$\text{ボーレートジェネレータ出力クロック} = \frac{\phi Tx}{N}$$

・　N + (16-K)/16 分周

$$\text{ボーレートジェネレータ出力クロック} = \frac{\phi Tx}{N + \frac{(16 - K)}{16}}$$

13.6.2.2 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[1:0]>で指定します。

I/O インタフェースモード時のクロックは、SCxCR<IOC><SCLKS>で設定します。

UART モード時のクロックは、SCxMOD0<SC[1:0]>で設定します。

(1) I/O インタフェースモードの転送クロック

表 13-3 に I/O インタフェースモードで可能なクロックを示します。

表 13-3 I/O インタフェースモードのクロック選択

モード SCxMOD0<SM[1:0]>	入出力選択 SCxCR<IOC>	クロックエッジ選択 SCxCR<SCLKS>	使用クロック
"00" (I/O インタフェース モード)	"0" (クロック出力 モード)	"0" (送信: 立ち下がり、 受信 : 立ち上がり)	ボーレートジェネレータ出力の 2 分周
		"1" (送信: 立ち上がり、 受信 : 立ち下がり)	ボーレートジェネレータ出力の 2 分周
	"1" (クロック入力 モード)	"0" (送信: 立ち下がり、 受信 : 立ち上がり)	SCxSCLK 端子入力
		"1" (送信: 立ち上がり、 受信 : 立ち下がり)	SCxSCLK 端子入力

SCxSCLK 端子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

- ダブルバッファ未使用の場合

  SCLK 周期 > 8/fsys

(2)　UART モードの転送クロック

表 13-4 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 13-4 UART モードのクロック選択

モード SCxMOD0<SM[1:0]>	クロック選択 SCxMOD0<SC[1:0]>
UART モード ("01", "10", "11")	"00" : タイマ出力
	"01" : ボーレートジェネレータ
	"10" : fsys
	"11" : SCxSCLK 端子入力

SCxSCLK 端子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/fsys

タイマの出力を使用する場合、カウンタと TBxRG1 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「TBxRG1 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

$$\text{転送レート} = \frac{\text{CGSYSCR<PRCK[1:0]>で選択されたクロック周波数}}{(\text{TBxRG1} \times 2) \times 2 \times 16}$$

(タイマプリスケーラクロック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)

13.7 送信/受信バッファと FIFO

13.7.1 構成

送信/受信バッファと FIFO の構成を図 13-4 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 13-4 バッファと FIFO の構成

13.7.2 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、I/O インタフェースモードでクロック入力モードの場合と 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 13-5 にモードとバッファ構成の関係をまとめます。

表 13-5 モードとバッファ構成

モード		SCxMOD2<WBUF>	
		"0"	"1"
UART モード	送信	シングル	ダブル
	受信	ダブル	ダブル
I/O インタフェースモード (クロック入力モード)	送信	シングル	ダブル
	受信	ダブル	ダブル
I/O インタフェースモード (クロック出力モード)	送信	シングル	ダブル
	受信	シングル	ダブル

13.7.3　送信バッファの初期化

送信バッファにデータが残っている状態で送信を中断した場合、送信バッファに新たにデータを書き込む前に送信バッファの初期化を行う必要があります。

送信バッファの初期化は、送信動作が停止している状態で行ってください。SCxMOD2<TXRUN>を読み出すことで、送信動作が停止しているかを判断できます。送信動作が停止していることが確認できたら、SCxTFC<TBCLR>に"1"を設定し、送信バッファの初期化を行います。

送信 FIFO が有効な場合、送信 FIFO の状態で動作が変わります。送信 FIFO にデータがある場合、送信 FIFO から送信バッファにデータが転送されます。送信 FIFO が空なら、SCxMOD2<TBEMP>が"1"にセットされます。

注)　I/O インタフェースモードのクロック入力モードではクロックが非同期に入力されます。送信動作が停止しているときにクロックを入力しないようにしてください。

13.7.4　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 13-6 にモードと FIFO 構成の関係をまとめます。

表 13-6 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

13.8 ステータスフラグ

SCxMOD2 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の時のみ意味を持ちます。

<RBFLL>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。

13.9 エラーフラグ

SCxCR 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 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 モード	オーバランエラー	パリティエラー	フレーミングエラー
I/O インタフェースモード (クロック入力モード)	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)	
I/O インタフェースモード (クロック出力モード)	不定	不定	"0"固定

13.9.1 OERR フラグ

UART モード、I/O 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 へデータ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。

I/O インタフェースモードのクロック出力モードの設定では、フラグのセットとともにクロック出力が停止します。

注) I/O インタフェースモードのクロック出力モードから他のモードへ切り替える際には、SCxCR を読み出し、オーバランフラグをクリアしてください。

13.9.2　PERR フラグ

UART モードではパリティエラーを、I/O 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ビットが異なる場合に"1"にセットされます。

アンダーランエラーは、I/O 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。

クロック入力モードの場合、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。

クロック出力モードの場合、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)　I/O インタフェースモードのクロック出力モードから他のモードへ切り替える際には、SCxCR を読み出し、アンダーランフラグをクリアしてください。

13.9.3　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<SBLEN>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。

I/O インタフェースモードではこのビットは"0"固定です。

13.10 受信

13.10.1 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、8 クロック目でデータをサンプリングします。

13.10.2 受信制御部

13.10.2.1 I/O インタフェースモードの場合

SCxCR<IOC>="0"のクロック出力モードのときは、SCxCR<SCLKS>の設定に従って、SCxSCLK 端子より出力されるクロックの立ち上がり/立ち下がりで SCxRXD 端子をサンプリングします。

SCxCR <IOC>="1"のクロック入力モードのときは、SCxCR<SCLKS>の設定に従って、SCxSCLK 端子の立ち上がり/立ち下がりエッジでシリアル受信データ SCxRXD 端子をサンプリングします。

13.10.2.2 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。

13.10.3 受信動作

13.10.3.1 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<RBFLL>) 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。

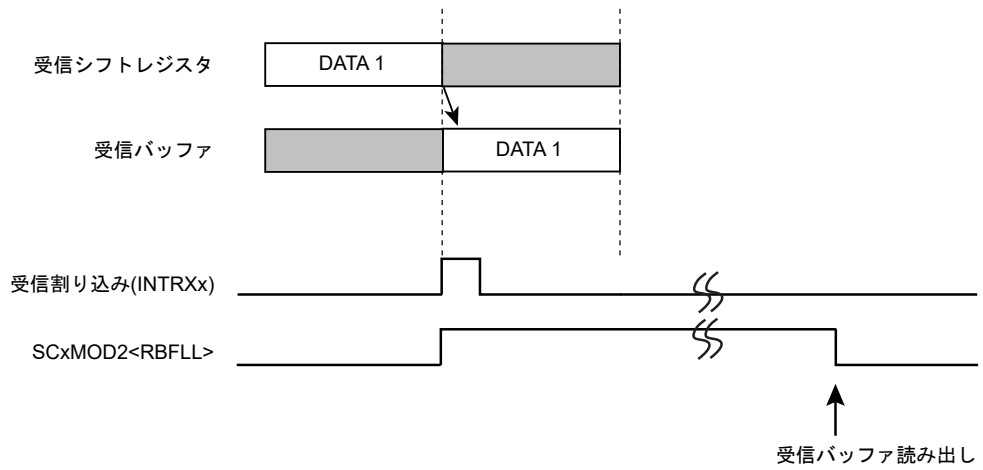


図 13-5 受信バッファの動作

13.10.3.2　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、パリティエラーフラグ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<FDPX[1:0]> = "01"	: 転送モードを半二重受信に設定
SCxFCNF<RFST><TFIE><RFIE>	: fill レベル到達後の継続受信自動禁止
<RXTCNT><CNFG> = "10111"	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<RIL[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<RFCS><RFIS> = "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。

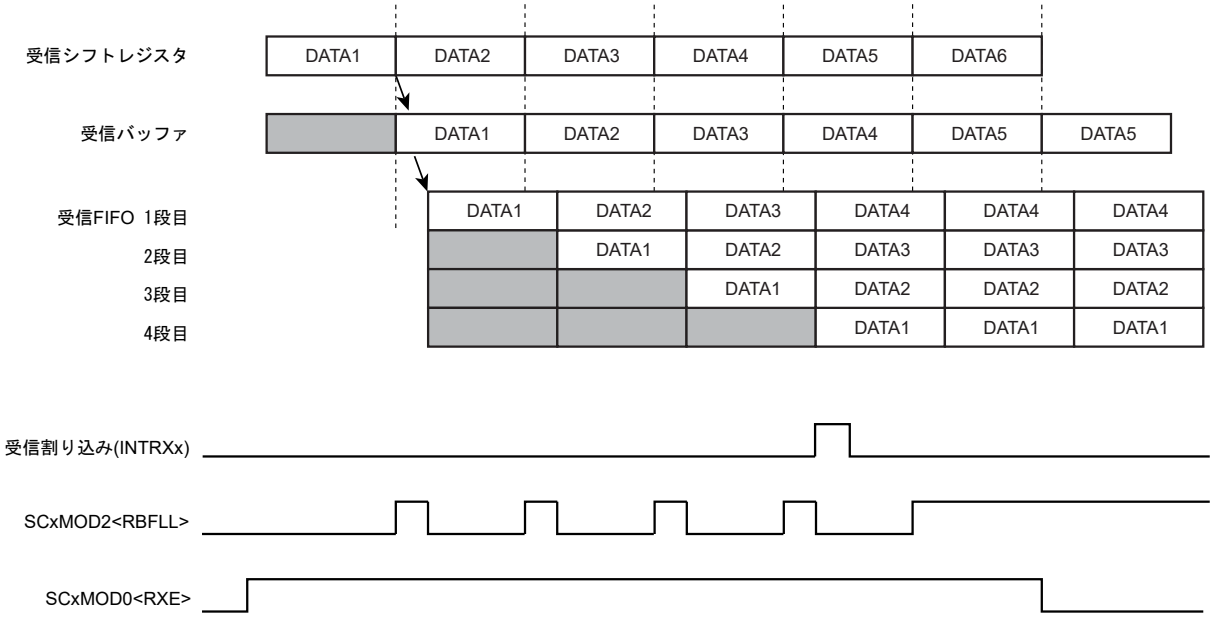


図 13-6 受信 FIFO の動作

13.10.3.3 I/O インタフェースモード、クロック出力モードでの受信

I/O インタフェースモードでクロック出力モード設定の場合、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クロック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。

クロック出力の停止/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 データ受信後にクロック出力を停止します。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クロック出力を再開します。

(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、クロック出力を停止します。1 データが読み出されるとクロック出力を再開します。

(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クロック出力を停止します。

1 データが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、クロック出力を再開します。

また、SCxFNCF<RXTXCNT>がセットされているとクロック出力停止とともに SCxMOD0<RXE>がクリアされ受信動作を停止します。

13.10.3.4　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<RBFLL>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、パリデ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。

13.10.3.5　ウエイクアップ機能

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、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。

13.10.3.6　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 のオーバーランが発生してオーバー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。

I/O インタフェースモードのクロック出力モードの設定では、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) I/O インタフェースのクロック出力モードから他のモードへ切り替える際には、SCxCR を読み出し、オーバランフラグをクリアしてください。

13.11 送信

13.11.1 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。

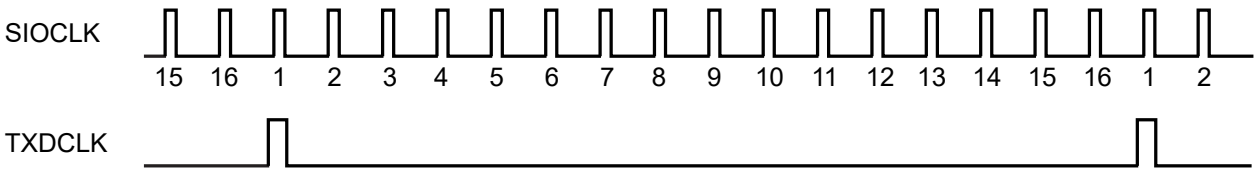


図 13-7 UART モード送信クロックの生成

13.11.2 送信制御部

13.11.2.1 I/O インタフェースモードの場合

SCxCR<IOC> = "0"のクロック出力モードのときは、SCxCR<SCLKS>の設定に従って、SCxSCLK 端子より出力されるクロックの立ち上がり/立ち下がりで送信バッファのデータを 1 ビットずつ SCxTXD 端子へ出力します。

SCxCR<IOC> = "1"のクロック入力モードのときは、SCxCR<SCLKS>の設定に従って、SCxSCLK 端子の立ち上がり/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 SCxTXD 端子へ出力します。

13.11.2.2 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。

13.11.3　送信動作

13.11.3.1　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 SC x MOD1<TXE>に"1"がセットされると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。

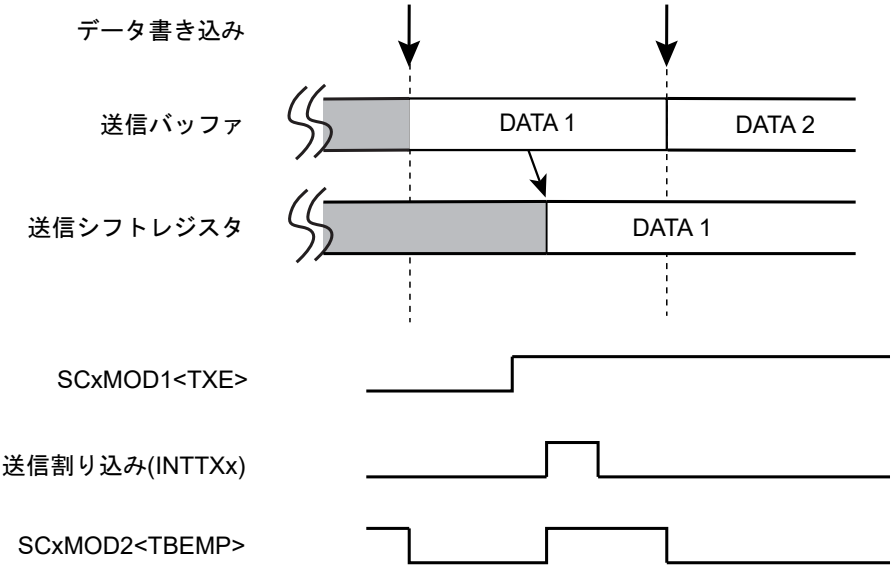


図 13-8 送信バッファの動作(ダブルバッファ有効時)

13.11.3.2 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。

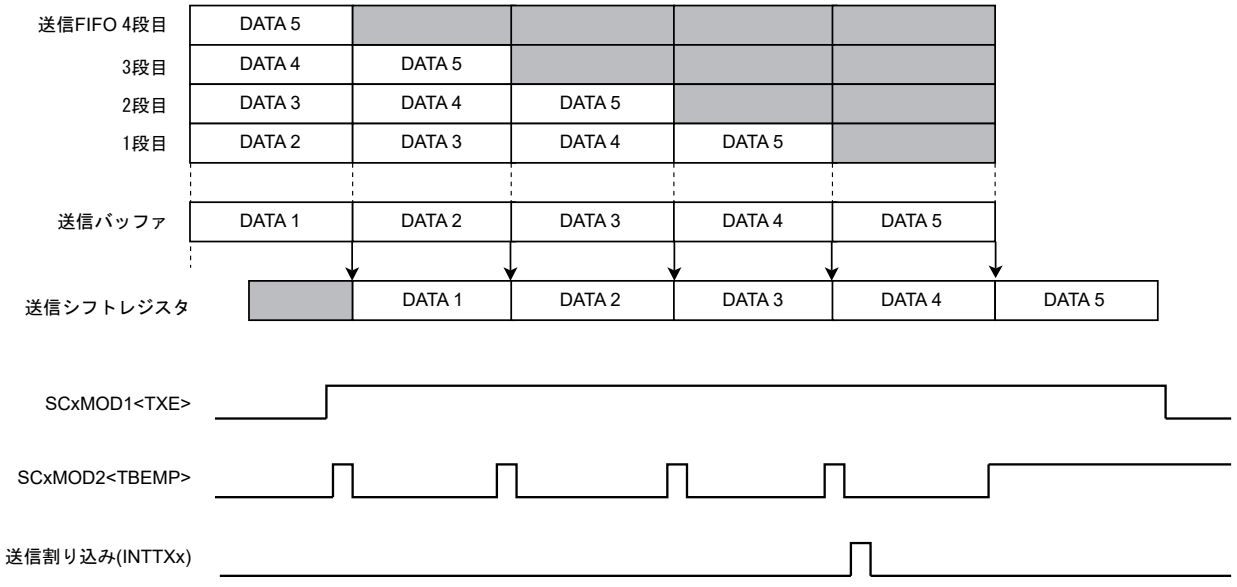
注) 送信 FIFO バッファ使用時は、SIO の転送モード設定(半二重/全二重)、FIFO 許可(SCxFCNF <CNFG>="1")の後、必ず送信 FIFO クリアを実行して下さい。

以下に、5 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<FDPX[1:0]> = "10"	: 転送モードを半二重送信に設定
SCxFCNF<RFST><TFIE><RFIE> <RXTXCNT><CNFG> = "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<TIL[1:0]> = "00"	: 割り込み発生の fill レベル を 0 に設定
SCxTFC<TFCS><TFIS> = "11"	: 送信 FIFO のクリアと割り込み発生条件の設定
SCxFCNF<CNFG> = "1"	: FIFO の許可

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。



13.11.3.3　I/O インタフェースモード、クロック出力モードでの送信

I/O インタフェースモードでクロック出力モードの場合、設定されたデータがすべて送信されるとクロック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 使用状況によってクロック出力の停止/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 データ送信後にクロック出力を停止します。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クロック出力を再開します。

(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クロック出力を停止します。バッファに次のデータが書き込まれるとクロック出力を再開します。

(3)　FIFO の場合

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

また、SCxFCNF<RXTXCNT>がセットされていると、クロック出力停止とともに SCxMOD0<TXE>がクリアされ、送信動作を停止します。

13.11.3.4　I/O インタフェースモード時の最終ビット出力後の SCxTXD 端子の状態

最終ビットを出力し、データのホールド時間が終了した後の SCxTXD 端子の状態を、SCxCR<TIDLE>で指定することができます。

SCxCR<TIDLE>="00"の場合、SCxTXD 端子は"Low"出力を保持します。SCxCR<TIDLE>="01"の場合、SCxTXD 端子は"High"出力を保持します。SCxCR<TIDLE>="10"の場合、SCxTXD 端子は最終ビットを保持します。

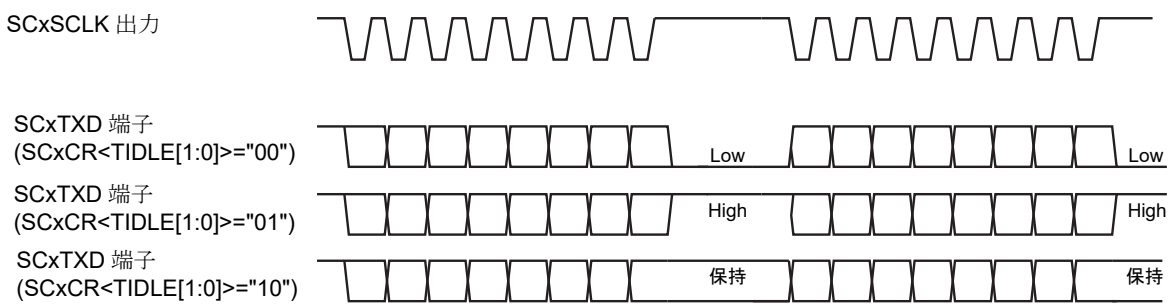


図 13-9 最終ビット出力後の SCxTXD 端子の状態

13.11.3.5 アンダーランエラー

I/O インタフェースモードのクロック入力モード時、送信 FIFO が空で、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へデータがセットされないときはアンダーランエラーになり、SCxCR<PERR>に"1"がセットされます。

アンダーランエラーが発生したときの SCxTXD 端子の状態を SCxCR<TXDEMP>で指定することができます。SCxCR<TXDEMP>が"0"の場合、SCxTXD 端子はデータ出力期間の間、"Low"出力を保持します。SCxCR<TXDEMP>が"1"の場合、SCxTXD 端子はデータ出力期間の間、"High"出力を保持します。

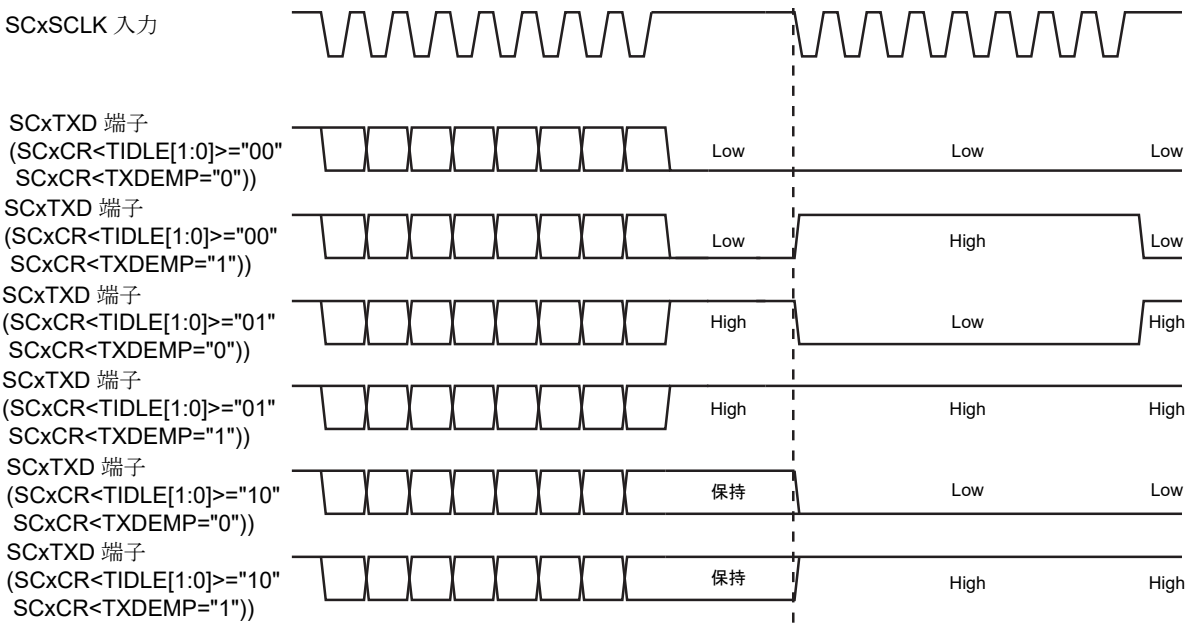


図 13-10 アンダーランが発生したときの SCxTXD 出力

I/O インタフェースモードのクロック出力モードの設定では、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) I/O インタフェースのクロック出力モードから他のモードへ切り替える際には、SCxCR を読み出し、アンダーランフラグをクリアしてください。

13.11.3.6 I/O インタフェースモード、クロック入力モードでのデータのホールド時間

I/O インタフェースモードでクロック入力モードの場合、SCxCR<EHOLD[2:0]>で最終ビットのホールド時間を設定できます。

最終ビットのホールド時間 ≤ SCLK 周期/2 を満たすように、最終ビットのホールド時間、SCLK 周期を設定してください。

13.12　ハンドシェイク機能

ハンドシェイク機能は $\overline{\text{SCxCTS}}$ (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。

$\overline{\text{SCxCTS}}$ 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、$\overline{\text{SCxCTS}}$ 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 $\overline{\text{SCxCTS}}$ 信号を立ち上げた場合、送信終了後に停止します。
注 2)　$\overline{\text{SCxCTS}}$ 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、$\overline{\text{RTS}}$ 端子はありませんが、任意のポートの 1 ビットを $\overline{\text{RTS}}$ 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。

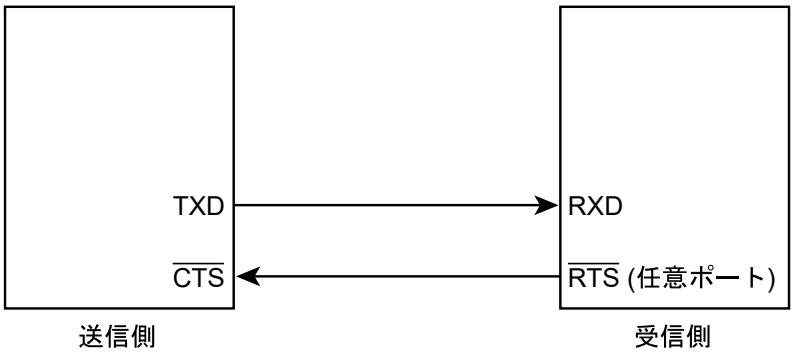


図 13-11 ハンドシェイク機能接続

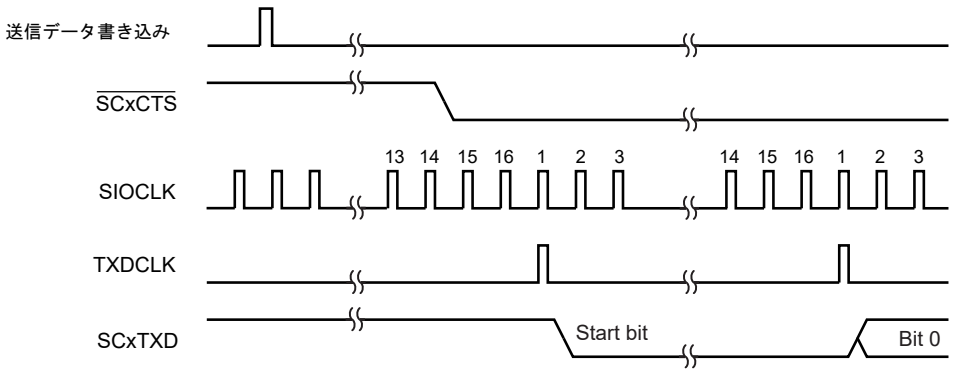


図 13-12 $\overline{\text{SCxCTS}}$ 信号のタイミング

13.13 割り込み/エラー発生タイミング

13.13.1 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 13-13 に示します。

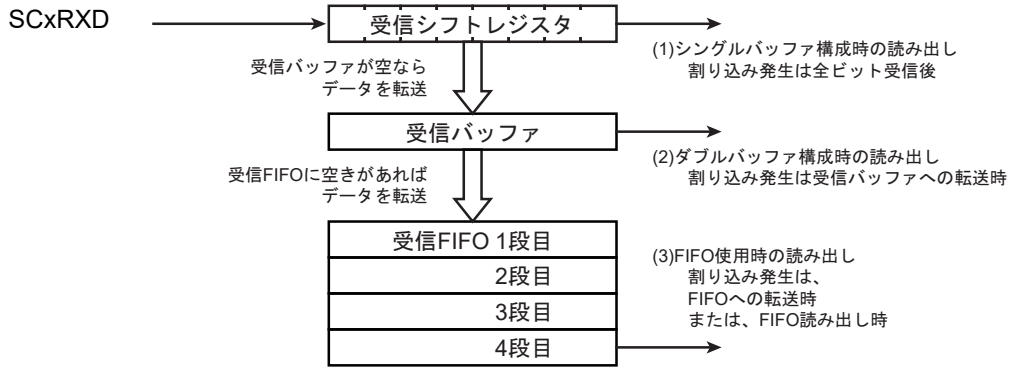


図 13-13 受信バッファ/FIFO 構成図

13.13.1.1 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。

表 13-7 シングルバッファ/ダブルバッファ構成の場合の受信割り込み発生条件

バッファ構成	UART モード	I/O インタフェースモード
シングルバッファ	−	・最終ビットの SCxSCLK 端子立ち上がり/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	受信シフトレジスタから受信バッファへ、データの移動が発生するとき。具体的には以下のタイミング。 ・受信バッファにデータがない場合、第 1 ストップビットの中央付近 ・受信シフトレジスタと受信バッファの両方にデータがある場合、バッファのリード時	受信シフトレジスタから受信バッファへ、データの移動が発生するとき。具体的には以下のタイミング。 ・受信バッファにデータがない場合、最終ビットの SCxSCLK 端子立ち上がり/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) ・受信シフトレジスタと受信バッファの両方にデータがある場合、バッファのリード時

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。

13.13.1.2 FIFO 使用の場合

FIFO 使用の場合の受信割り込みは、表 13-8 の割り込み発生タイミングに記載の動作が発生したときに、SCxRFC<RFIS>の設定で決まる条件を満たしていると発生します。

表 13-8 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	割り込み発生タイミング
"0"	FIFO fill レベル(SCxRST<RLVL[2:0]>) = 割り込み発生 fill レベル(<RIL[1:0]>) のとき	・受信バッファから受信FIFOへ受信データの転送がおこなわれるとき
"1"	FIFO fill レベル(SCxRST<RLVL[2:0]>) ≥ 割り込み発生 fill レベル(<RIL[1:0]>) のとき	・受信FIFOから受信データをリードしたとき ・受信バッファから受信FIFOへ受信データの転送がおこなわれるとき

13.13.2　送信割り込み

送信動作のデータの流れと書き込みの経路を図 13-14 に示します。

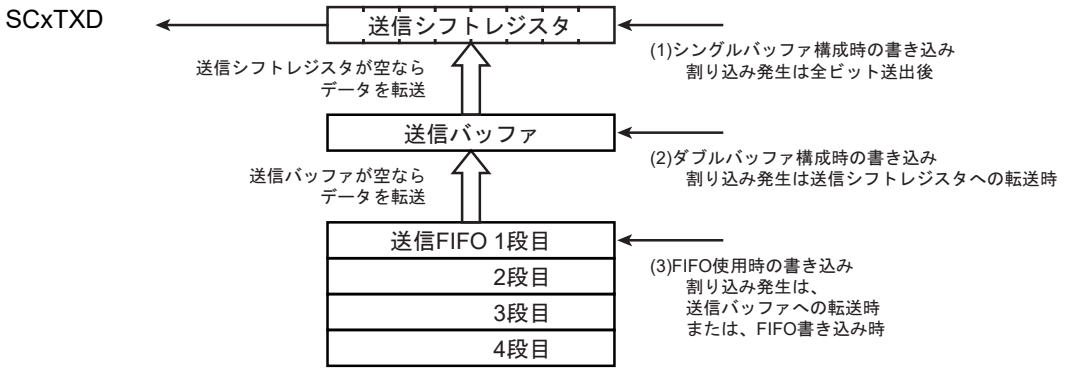


図 13-14 送信バッファ/FIFO 構成図

13.13.2.1　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。

表 13-9 シングルバッファ/ ダブルバッファ構成の場合の送信割り込み発生条件

バッファ構成	UART モード	I/O 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 SCxSCLK 端子立ち上がり/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に送信割り込みが発生します。 なお、SCxMOD1<TXE> = "1"で送信シフトレジスタが空の場合、送信バッファにデータを書いた直後に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るため、送信割り込みが発生します。	

13.13.2.2 FIFO 使用の場合

FIFO 使用の場合の送信割り込みは、表 13-10 の割り込み発生タイミングに記載の動作が発生したときに、SCxTFC<TFIS>の設定で決まる条件を満たした場合に発生します。

表 13-10 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	割り込み発生タイミング
"0"	FIFO fill レベル(SCxTST<TLVL[2:0]>) = 割り込み発生 fill レベル(<TIL[1:0]>) のとき	・送信FIFOから送信バッファへ送信データの転送が行われたとき
"1"	FIFO fill レベル(SCxTST<TLVL[2:0]>) ≤ 割り込み発生 fill レベル(<TIL[1:0]>) のとき	・送信FIFOへ送信データをライトしたとき ・送信FIF0から送信バッファへ送信データの転送が行われたとき

13.13.3 エラー発生

13.13.3.1 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	−	判定： パリティビットの中央付近 フラグ変化： ストップビットの中央付近

13.13.3.2 I/O インタフェースモード

オーバランエラー	最終ビットの SCxSCLK 端子立ち上がり/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	次回の SCxSCLK 端子の立ち上がり/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) クロック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。

13.14　DMA 要求

UART/SIO 割り込み(INTRXx,INTTXx)発生のタイミングで DMAC に対して DMA 要求を発行します。

本製品で DMA 要求に使用できるチャネルについては、「製品情報」の章を参照してください。

注 1)　SIO/UART の送信/受信割り込みを使用して DMA 転送を行う場合、SCxMOD<SWRST>でソフトウエアリセットを実施した後、DMAC の動作を許可させ、送信/受信の設定を行ってください。

注 2)　DMA 転送を使用する場合、FIFO は使用できません。

13.15 ソフトウエアリセット

SCxMOD2<SWRST[1:0]>に"10" → "01"の順でライトすることによりソフトウエアリセットが発生します。

これにより、SCxMOD0<RXE>、SCxMOD1<TXE>、SCxMOD2<TBEMP><RBFLL><TXRUN>、SCxCR<OERR> <PERR> <FERR>が初期化され、送受信回路は初期状態になります。

その他の状態は保持されます。

13.16　モード別動作説明

13.16.1　モード 0 (I/O インタフェースモード)

SCxMOD0 <SM[1: 0]>を"00"にセットすると I/O インタフェースモードになります。

このモードには、同期クロック SCLK を出力するクロック出力モードと、外部より同期クロック SCLK を入力するクロック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。

13.16.1.1　送信

(1)　クロック出力モード

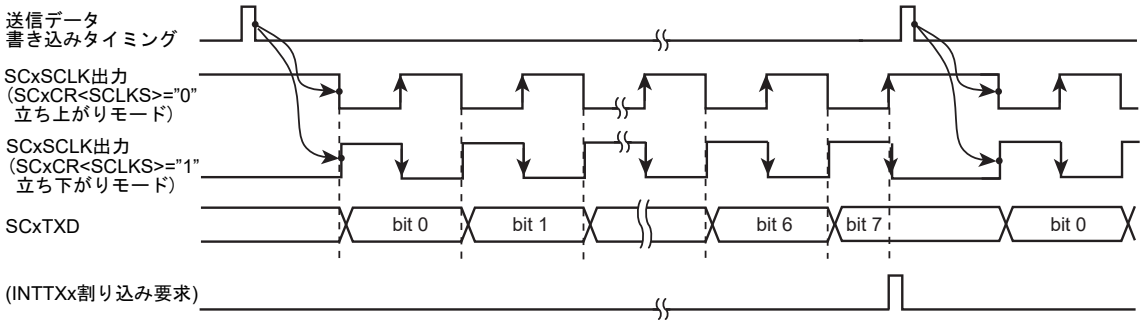
- ダブルバッファ不許可(SCxMOD2<WBUF> = "0")の場合

  送信バッファにデータを書き込むたびに、データが SCxTXD 端子から、クロックが SCxSCLK 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。

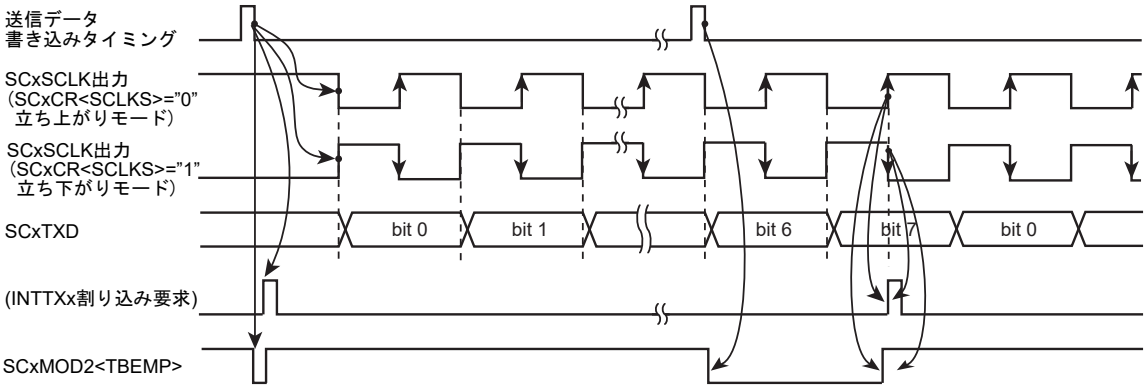
- ダブルバッファ許可(SCxMOD2<WBUF> = "1")の場合

  シフトレジスタが空の状態で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。

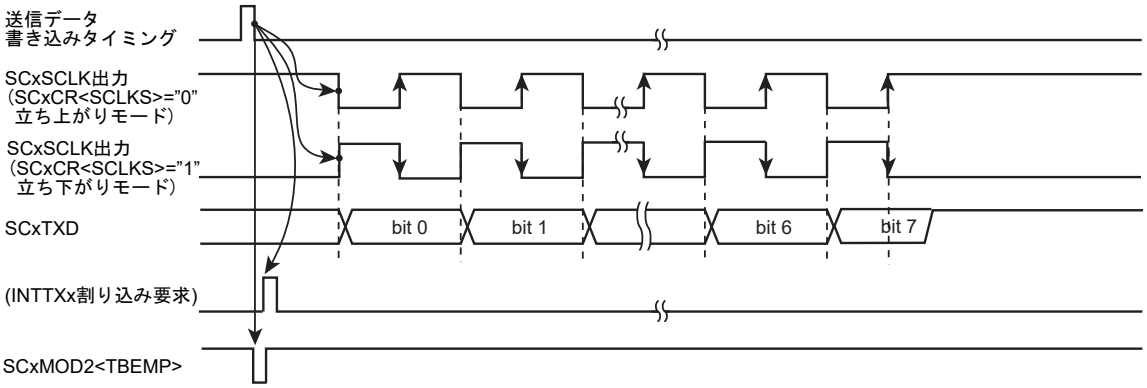
  シフトレジスタのデータ送出終了時に送信バッファにデータが存在しない場合は、割り込み(INTTXx)を発生せず、クロック出力も停止します。



<WBUF> = "0" (ダブルバッファ不許可)の場合 (SCxCR<TIDLE>="10")



<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) (SCxCR<TIDLE>="01")

図 13-15 I/O インタフェースモード送信動作(クロック出力モード)

(2)　クロック入力モード

・　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クロックが入力されると、8 ビットのデータが SCxTXD 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 13-16 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

クロック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、クロックが入力された場合、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、SCxCR<TXDEMP>で設定したレベルが SCxTXD 端子に出力されます。

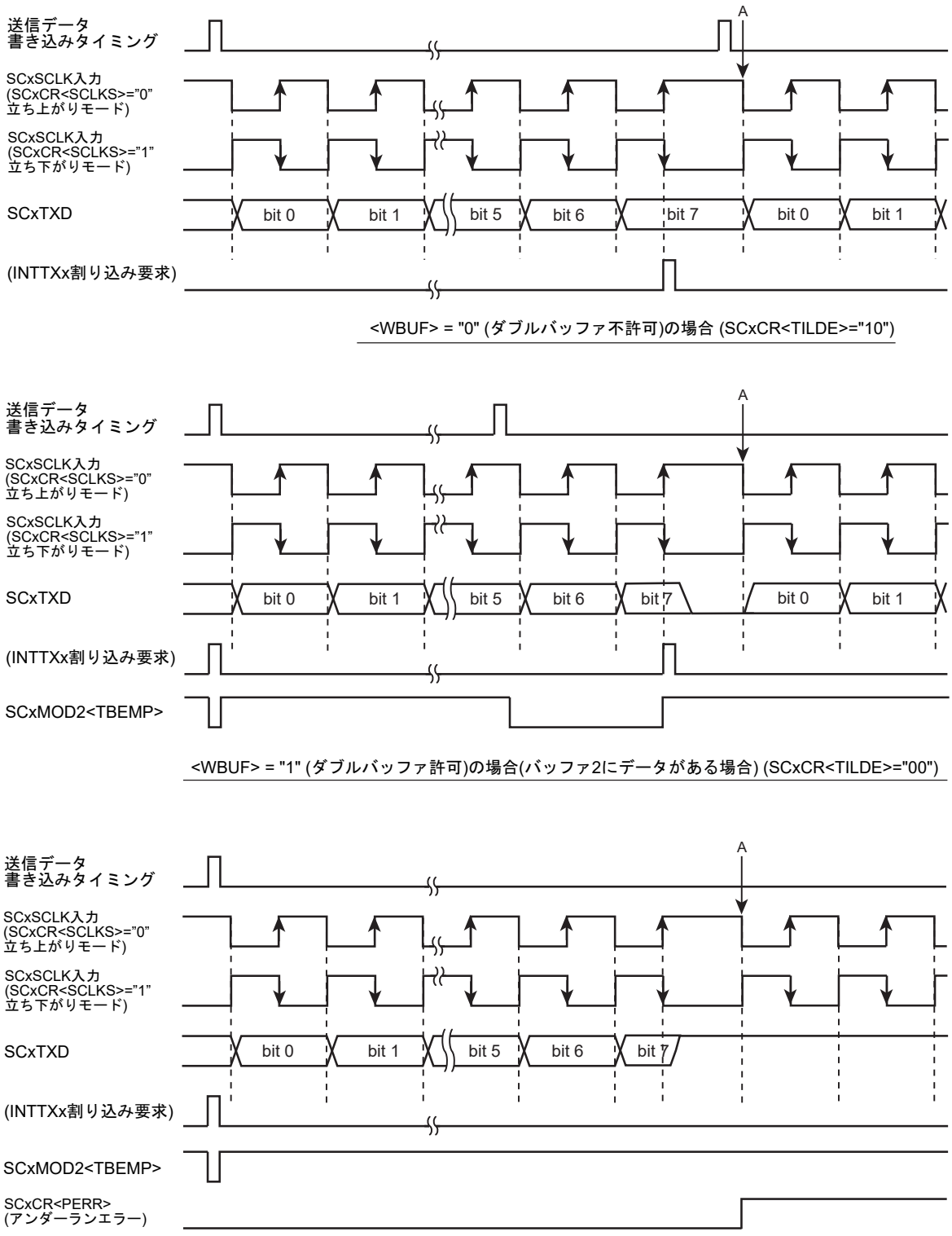


図 13-16 I/O インタフェースモード送信動作(クロック入力モード)

13.16.1.2　受信

(1)　クロック出力モード

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

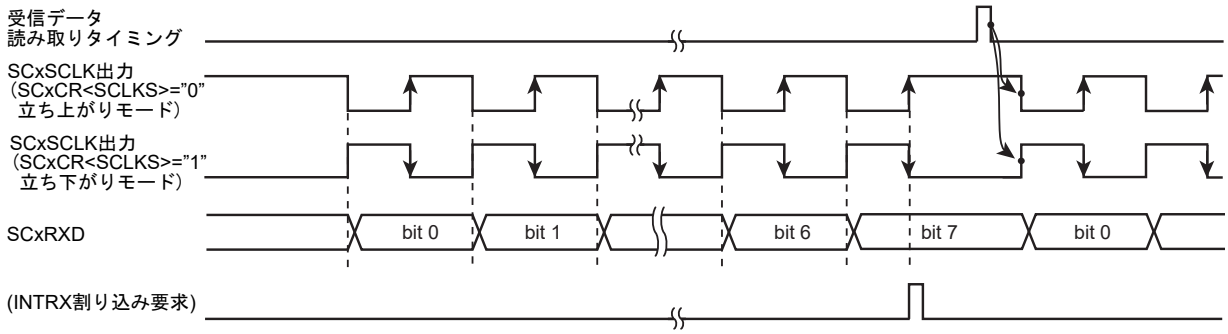
- ダブルバッファ不許可(SCxMOD2<WBUF> = "0")の場合

  受信データが読み出されるごとに、SCxSCLK 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。

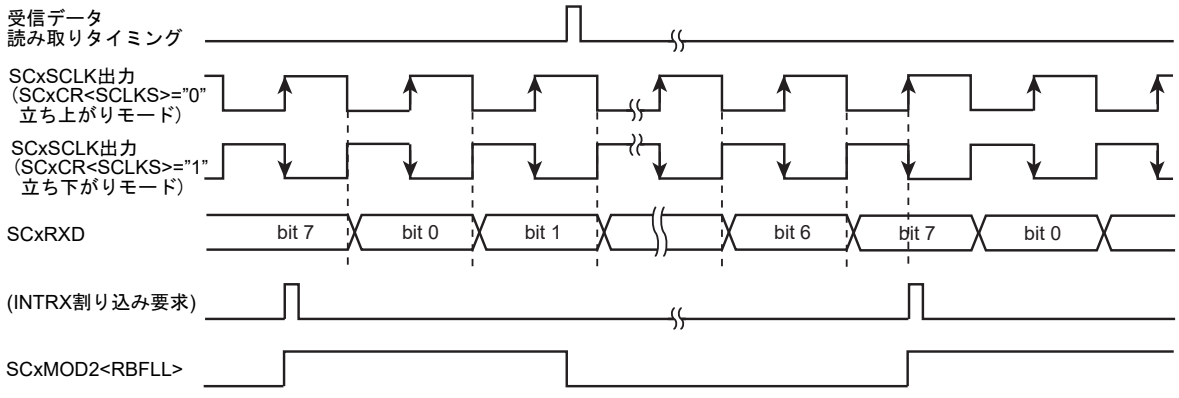
- ダブルバッファ許可(SCxMOD2<WBUF> = "1")の場合

  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<RBFLL>が"1"にセットされ、割り込み INTRXx が発生します。

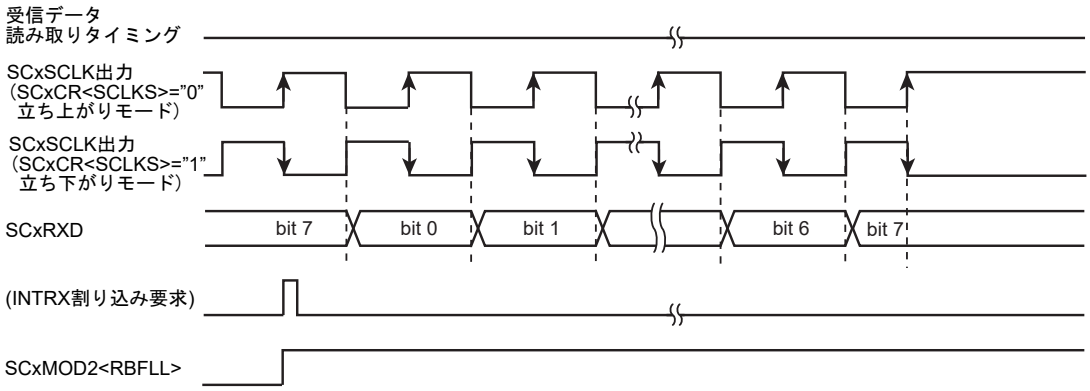
  受信バッファにデータが存在する状態で、次の 8 ビット分のデータを受信完了する前に受信バッファのデータが読み出されない場合、割り込み INTRXx は発生せず、クロック出力は停止します。この状態で受信バッファのデータを読み出すと、シフトレジスタのデータを受信バッファに移し、割り込み INTRXx を発生して受信を再開します。



<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(バッファのデータを読み出した場合)



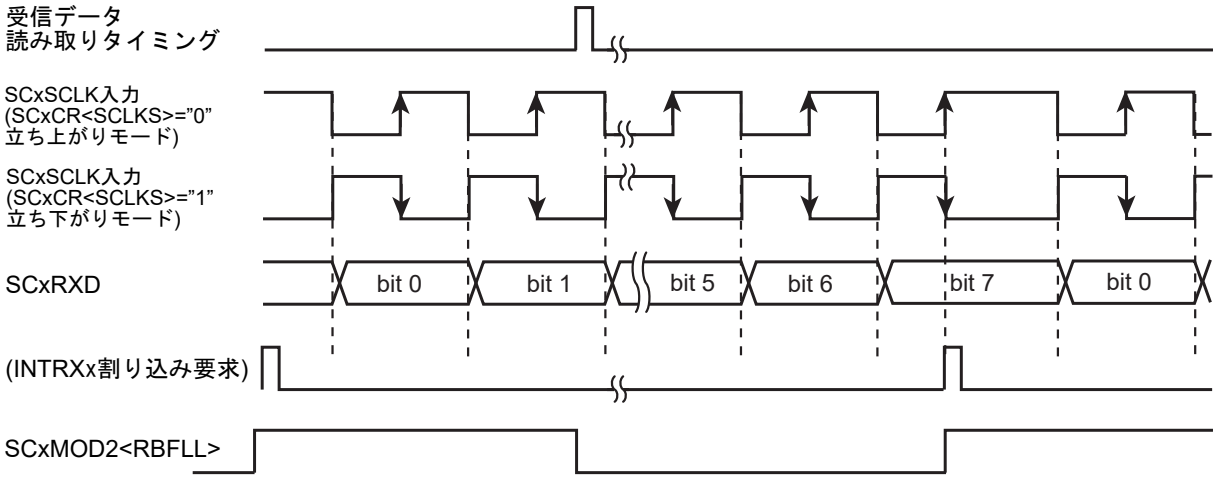
<WBUF> = "1" (ダブルバッファ許可)の場合(バッファのデータが読み出されない場合)

図 13-17 I/O インタフェースモード受信動作(クロック出力モード)

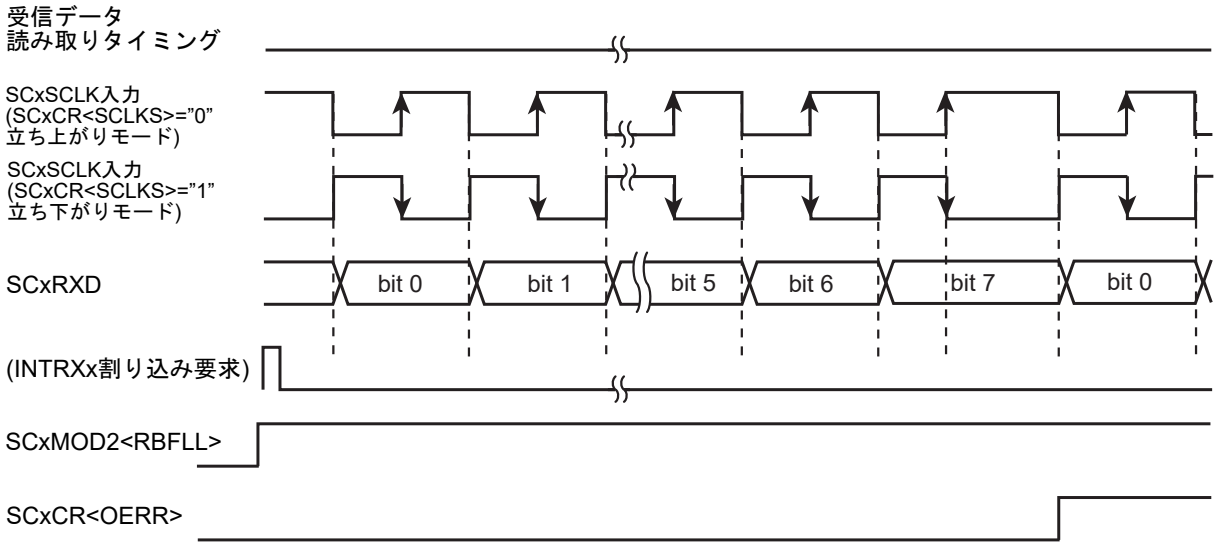
(2)　クロック入力モード

クロック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、受信したデータはシフトレジスタから受信バッファに移され、連続して次のデータ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。



バッファのデータを読み出した場合



バッファのデータが読み出されない場合

図 13-18 I/O インタフェースモード受信動作(クロック入力モード)

13.16.1.3 送受信(全二重)

(1) クロック出力モード

・ 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クロック出力を開始します。

クロック出力により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データが、SCxTXD 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。この状態でクロック出力は停止します。

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クロック出力を開始します。

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。受信と並行してデータが SCxTXD 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 <RBFLL> = "1") 場合はクロック出力が停止します。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、クロック出力が再開されて次の送受信が始まります。

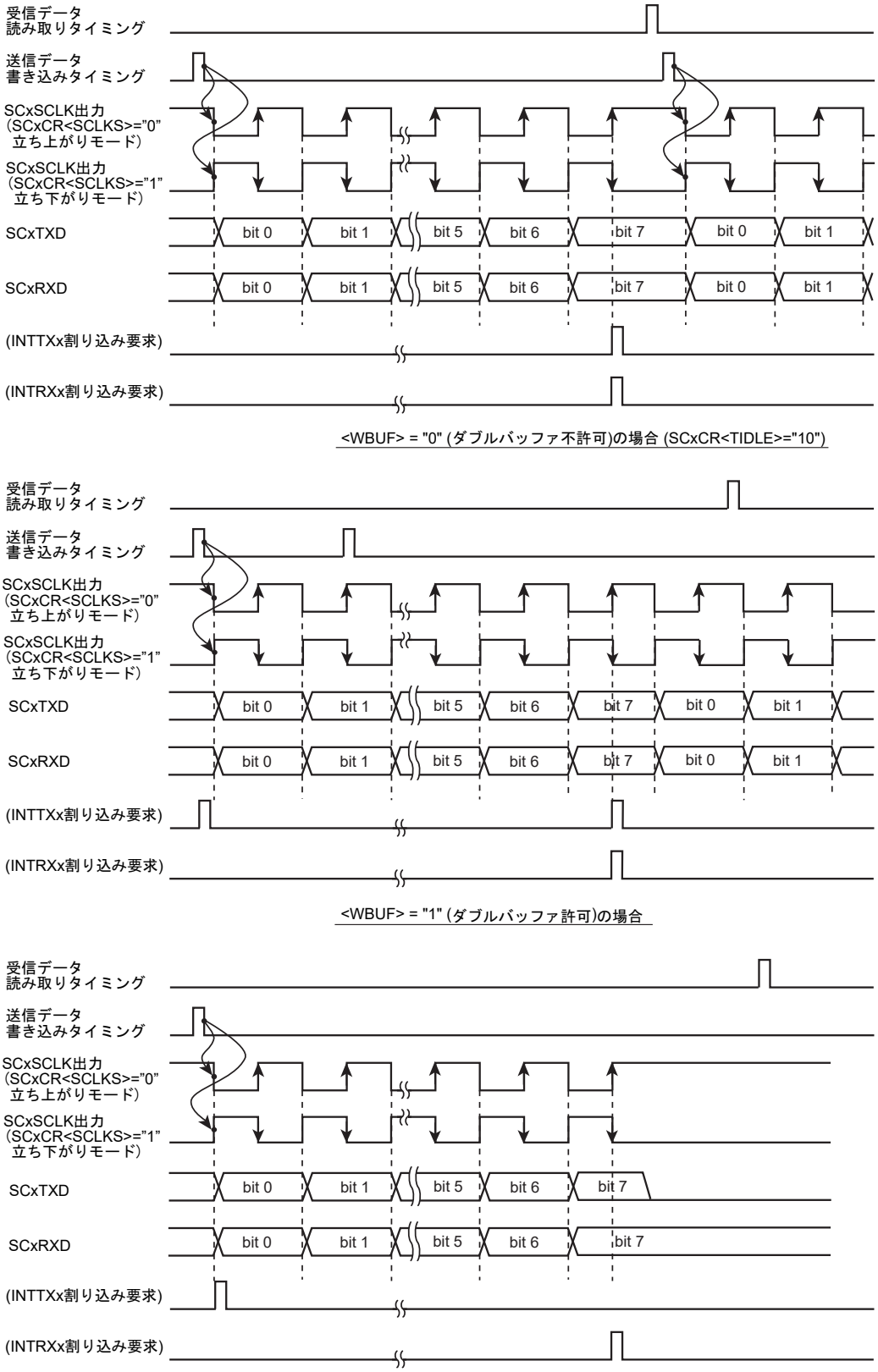


図 13-19 I/O インタフェースモード送受信動作(クロック出力モード)

(2) クロック入力モード

・ 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クロックが入力されると、8 ビットのデータが SCxTXD 端子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。

次のフレームのクロック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 13-20 に示す A 点までに書き込んでください)。受信データは、次のデータ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(INTTXx)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。

次のデータのためのクロック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 13-20 に示す A 点までに書き込んでください)。受信データは、次のデータの受信が終了する前に読み出してください。

続けて次のデータのためのクロックが入力されると、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。

データ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、次のデータのためのクロック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セットされ、SCxCR<TXDEMP>で設定したレベルが SCxTXD 端子に出力されます。

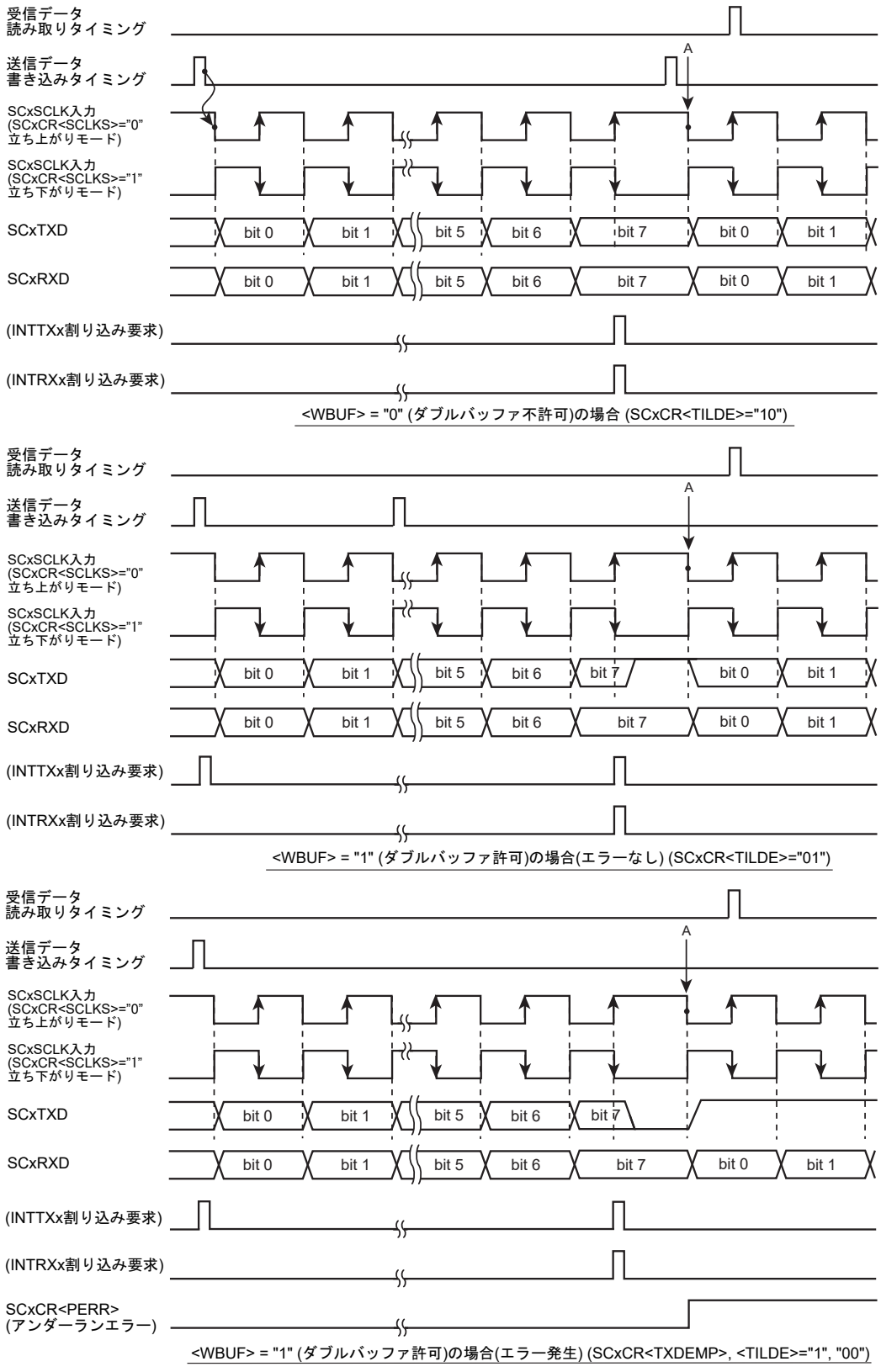


図 13-20 I/O インタフェースモード送受信動作(クロック入力モード)

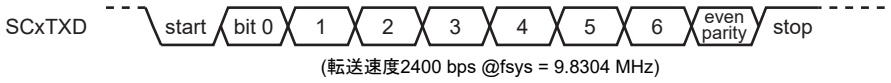
13.16.2 モード 1 (7 ビット UART モード)

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。



(転送速度2400 bps @fsys = 9.8304 MHz)

クロック条件	システムクロック:	高速 (fc)
	高速クロックギア:	1 倍 (fc)
	プリスケーラクロック:	fperiph/2 (fperiph = fsys)

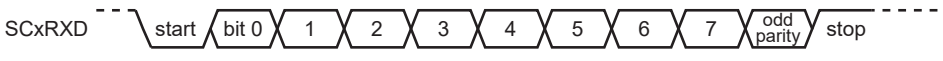
		7	6	5	4	3	2	1	0	
SCxMOD0	←	x	0	-	0	0	1	0	1	7 ビット UART モードに設定
SCxCR	←	x	1	1	x	x	x	0	0	偶数パリティイネーブルに設定
SCxBRCR	←	0	0	1	0	0	1	0	0	2400bps に設定
SCxBUF	←	*	*	*	*	*	*	*	*	送信データを設定

x : don't care - : no change

13.16.3 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。



(転送速度9600 bps @fsys = 9.8304 MHz)

クロック条件	システムクロック:	高速 (fc)
	高速クロックギア:	1 倍 (fc)
	プリスケーラクロック:	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	奇数パリティイネーブルに設定
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

13.16.4　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> に書き込みます。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。

13.16.4.1　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 SCxTXD 端子は、必ず PxOD 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。

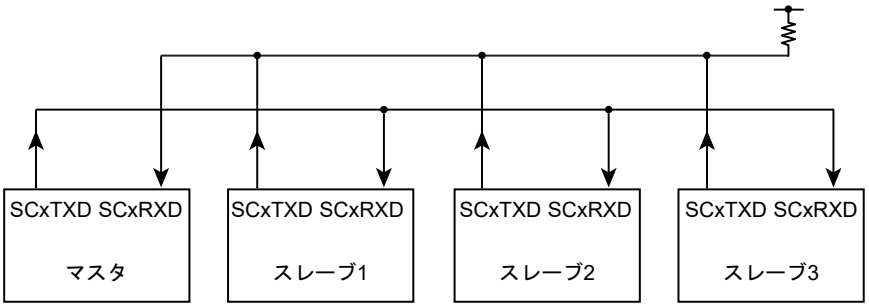
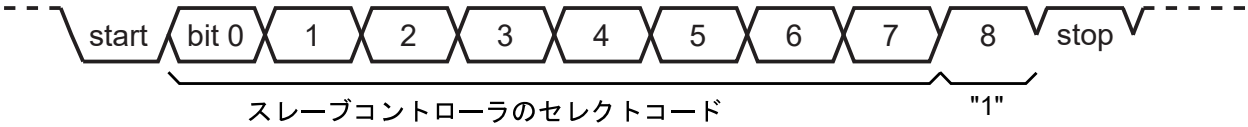


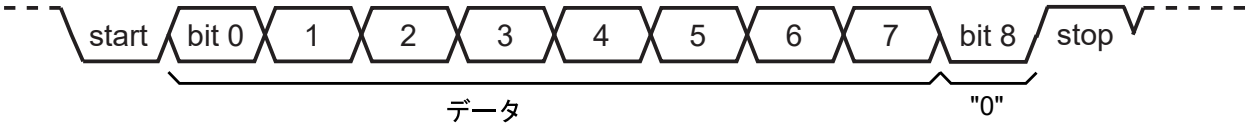
図 13-21 ウェイクアップ機能によるシリアルリンク

13.16.4.2 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 14 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

以下の説明中、"x"はチャネル番号を表します。

14.1　構成

図 14-1 にシリアルバスインタフェースのブロック図を示します。

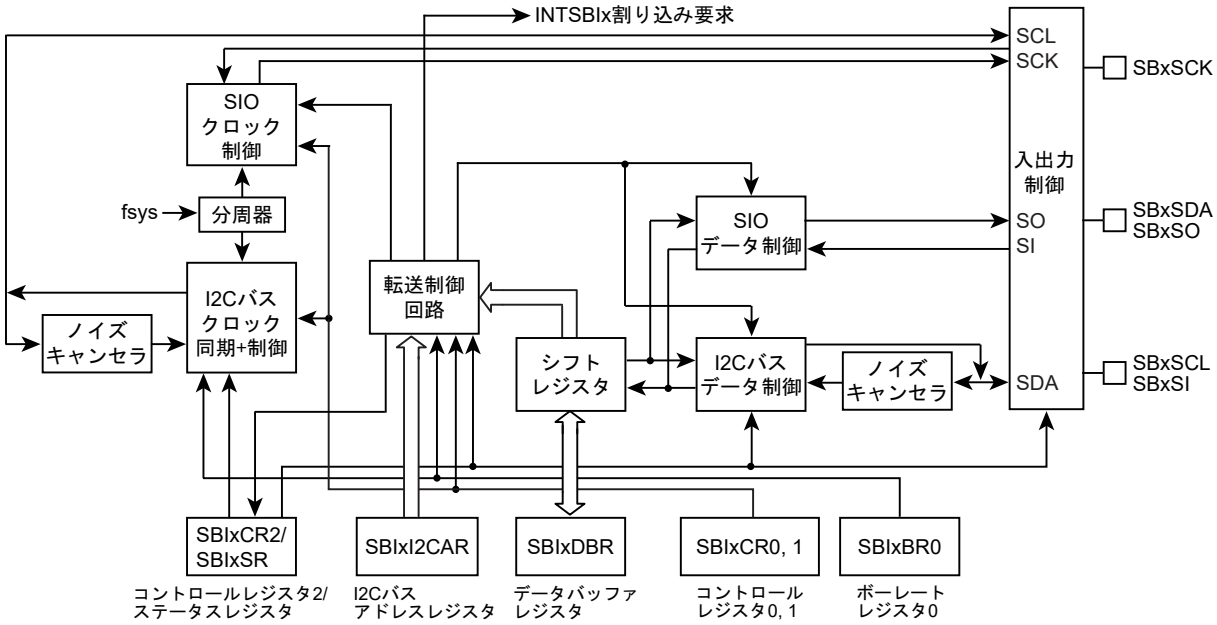


図 14-1 シリアルバスインタフェースブロック図

14.2　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「14.3.1 I2C バスモード時のコントロールレジスタ」および「14.4.1 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。

14.2.1　レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		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

14.3 I2C バスモード

14.3.1 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。

14.3.1.1 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、一番最初に<SBIEN>に"1"を設定してください。 <SBIEN>を"1"に設定することで SBI に関するレジスタのリード、ライトが可能になります。 動作禁止の場合、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。 シリアルバスインタフェース回路を一旦動作させた後に動作禁止にした場合、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。

14.3.1.2　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)
4	ACK	R/W	マスタモード 0: アクノリッジメントのためのクロックを発生しない 1: アクノリッジメントのためのクロックを発生する スレーブモード 0:アクノリッジメントのためのクロックをカウントしない 1:アクノリッジメントのための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	SBxSCL 出力クロックの周波数選択<SCK[2:0]>@ライト　(注 2)
0	SCK[0]	W	
	SWRMON	R	ソフトウエアリセット状態モニタ 0:ソフトウエアリセット中 1:ソフトウエアリセット中ではない

<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

000	n = 5	システムクロック: fsys クロックギア: fc/1 周波数 = $\frac{fsys}{2^n + 72}$ [Hz]
001	n = 6	
010	n = 7	
011	n = 8	
100	n = 9	
101	n = 10	
110	n = 11	
111	reserved	

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、「14.3.2.2 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。

注 5) マスタモードで<BC[2:0]>="001"で<ACK>="0"のとき、ストップコンディション発生後の SBxSCL の立ち下がりによって、SBxSCL が"L"に固定されることがあり、他のマスタデバイスがバスを使用できなくなります。複数のマスタデバイスがバスに接続されている場合、ストップコンディション発生前に、転送ビット数を"2"以上にしてください。

14.3.1.3　SBIxCR2(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	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	トランスミッタ/レシーバの選択 0: レシーバ 1: トランスミッタ
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 1) 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) (注 2) 01: SIO モード 10: I2C バスモード (注 3) 11: Reserved
1-0	SWRST[1:0]	W	ソフトウエアリセットの発生 最初に"10"、次に"01"をライトすると、ソフトウエアリセットが発生します。 詳細は、「14.3.2.16 ソフトウエアリセット」を参照してください。

注 1)　通信中はモードを切り替えないでください。

注 2)　ポートモードへの切り替えはバスフリーを確認してから行ってください。

注 3)　ポートモードから I2C バスモードへの切り替えは、SBxSDA 端子/SBxSCL 端子が"High"になっていることを確認してから行ってください

注 4)　SBIxCR2 は SBIxSR と同じアドレスに割り当てられています。従ってリードモディファイライトによる操作はできません。

14.3.1.4 SBIxSR(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	MST	TRX	BB	PIN	AL	AAS	AD0	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	アービトレー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 (ゼネラルコールアドレス検出時もセットされます。)
1	AD0	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"

14.3.1.5　SBIxBR0(ボーレートレジスタ 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。

14.3.1.6 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、リードモディファイライトによる操作はできません。

14.3.1.7　SBIxI2CAR(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 SBIxI2CAR を"0x00"に設定しないでください。("0x00"に設定した場合、スレーブモードで I2C バス規格の START バイト("0x01")を受信した時にスレーブアドレスが一致したと判断します。)

14.3.2 制御

14.3.2.1 動作モードの設定

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、<SBIM[1:0]>を"10"に設定してください。

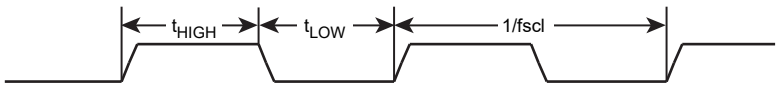
注 1) ポートモードへの切り替えは、SBxSDA 端子/SBxSCL 端子が"High"になっていることを確認してから行ってください。

注 2) ポートモードから I2C バスモードの切り替えは、SBxSDA 端子/SBxSCL 端子が"High"になっていることを確認してから行ってください。

14.3.2.2 シリアルクロック

(1) クロックソース

SBIxCR1 <SCK[2:0]>で、マスタモード時に SBxSCL 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。



$t_{LOW} = 2^{n-1}/fsys + 58/fsys$
$t_{HIGH} = 2^{n-1}/fsys + 14/fsys$
$fscl = 1/(t_{LOW} + t_{HIGH})$
$= \frac{fsys}{2^n + 72}$

SBIxCR1<SCK[2:0]>	n
000	5
001	6
010	7
011	8
100	9
101	10
110	11

図 14-2 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願います。

(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。

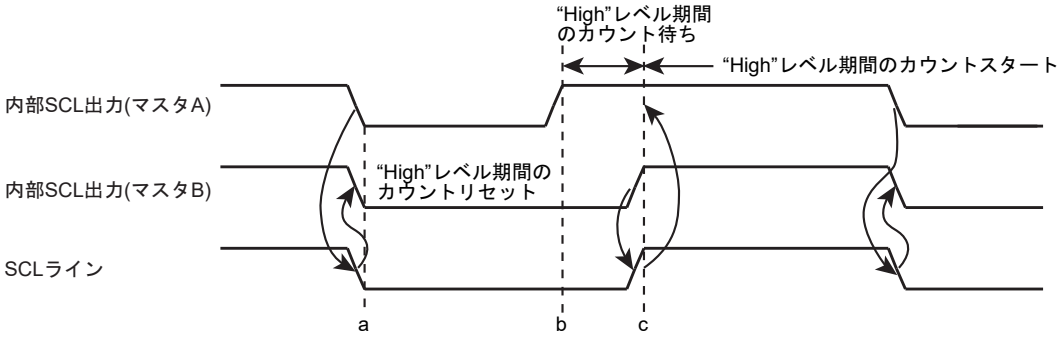


図 14-3 クロック同期化の例

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 が内部 SCL 出力を"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。

14.3.2.3　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 SBxSDA 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 SBxSDA 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アドレス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、SBxSDA 端子を"Low"レベルに引き、アクノリッジ信号を発生します。ゼネラルコールのセカンドバイトは内容によりアクノリッジ信号の発生有無を制御する必要がありますので、ソフトウエアによりアクノリッジ信号の制御を行なってください。

<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。

14.3.2.4　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。

14.3.2.5 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した場合、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。

14.3.2.6 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。

なお、<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、ハードウエアにより"0"にクリアされます。

14.3.2.7 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ます。

SBIxCR2<TRX>を"0"に設定すると、レシーバとして動作します。

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

アドレッシングフォーマットで使用するときには、<TRX>は下記のように設定されます。

(1) マスタモード時

マスタモード時は、スレーブアドレスと方向ビットを送信後、スレーブデバイスからアクノリッジが返ってくると、ハードウエアにより、下記のように<TRX>が設定されます。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。

- 送信した方向ビットが"1"の場合、<TRX>は"0"に設定されます。
- 送信した方向ビットが"0"の場合、<TRX>は"1"に設定されます。

(2) スレーブモード時

スレーブモード時は、アドレッシングフォーマットでデータ転送を行う場合に、下記の条件が成立するとマスタデバイスの送ってくる方向ビットにあわせ、<TRX>が設定されます。

- 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
- ゼネラルコールを受信したとき

<TRX>は下記のように設定されます。

- 受信した方向ビットが"1"の場合、<TRX>は"0"に設定されます。
- 受信した方向ビットが"0"の場合、<TRX>は"1"に設定されます。

14.3.2.8　バスビジーモニタ

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、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。

<BB>が"1"のときをバスビジー状態、<BB>が"0"のときをバスフリー状態と呼びます。

マスタデバイスは、バスフリーの状態でのみスタートコンディションを発生することができます。スタートコンディションを発生する前には、必ず<BB>が"0"であることを確認してください。

<BB>が"1"の状態でスタートコンディションを発生すると、スタートコンディションは発生せず、アービトレーションロストが発生します。

14.3.2.9　割り込みサービス要求と解除

割り込み要求(INTSBIx)が発生すると、SBIxCR2<PIN>が"0"にクリアされ、割り込みサービス要求状態になります。<PIN>が"0"の間、SBxSCL 端子を"Low"レベルにします。

<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。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。

<PIN>が"1"にセットされると SBxSCL 端子が開放されます。<PIN>が"1"にセットされてから SBxSCL 端子が開放されるまで t_{LOW} の時間がかかります。

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場合、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。)

14.3.2.10　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。

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。アービトレーションロストが発生したマスタ 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。

もし、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。

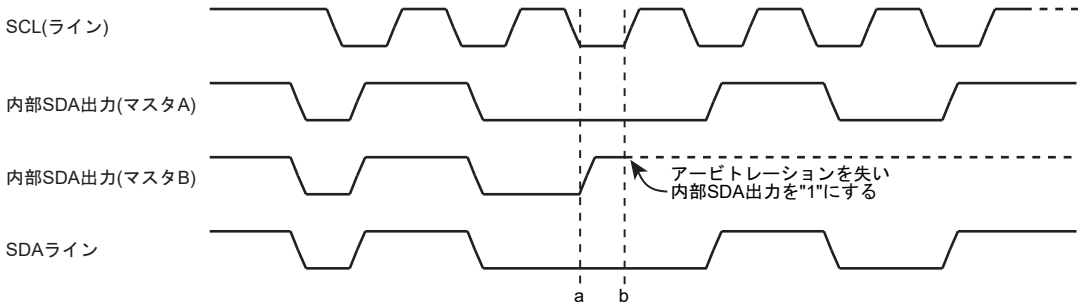


図 14-4 アービトレーションロスト

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。

アービトレーションロストが発生すると、SBIxSR<MST>と< TRX>は"0"にクリアされ、スレーブレシーバモードになり、SCL クロックの出力を停止します。

スレーブアドレス転送中にアービトレーションロストが発生したデバイスは、通常のスレーブデバイスのように、他のマスタが送信するスレーブアドレスを受信します。

受信したスレーブアドレスが SBIxI2CAR<SA>と一致した場合、<PIN>が"0"にクリアされ、INTSBIx が発生します。一致しなかった場合、<PIN>は"1"のままですが、INTSBIx が発生します。

<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。

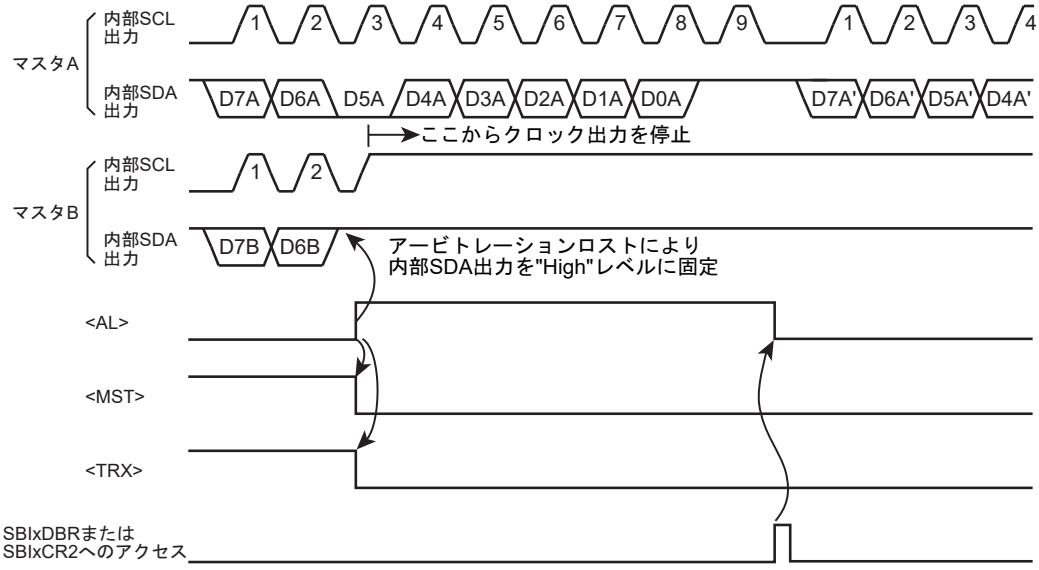


図 14-5 マスタ B の場合の例(D7A=D7B,D6A=D6B)

14.3.2.11 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、ゼネラルコールアドレスまたは SBIxI2CAR<SA>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。

14.3.2.12　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。

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。

14.3.2.13　最終受信ビットモニタ

SBIxSR<LRB>には、SBxSCL 端子の立ち上がりで取り込まれた SBxSDA 端子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。

14.3.2.14　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、SBIxDBR にスレーブアドレスと方向ビットを設定後、スタートコンディションを発生することで、スレーブデバイスに対し、スレーブアドレスと方向ビットを送信することができます。

14.3.2.15　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。

14.3.2.16　ソフトウエアリセット

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、ソフトウエ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。

I2C モードで SBIxCR2<SWRST[1:0]>へ、最初に"10"、次に"01"をライトすると、ソフトウエアリセットが発生します。ライトの際、SBIxCR2<MST><TRX><BB><PIN> は "0000"、SBIxCR2<SBIM[1:0]> は"10" (I2C バスモード)を指定してください。ソフトウエアリセットが発生すると、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"00"にクリアされます。

注)　ソフトウエ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。

14.3.3 データ転送手順

14.3.3.1 デバイスの初期化

最初に SBIxCR1<ACK>と<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には、<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", <SWRST[1:0]> に"00"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

14.3.3.2 スタートコンディション, スレーブアドレスの発生

スタートコンディションとスレーブアドレスを、次の手順で発生します。

はじめに、バスフリー状態(<BB> = "0")を確認します。次に、SBIxCR1<ACK>に"1"を書き込んで、アクノリッジメントモードに設定します。また、SBIxDBR に、送信するスレーブアドレスと方向ビットのデータを書き込みます。

<BB> = "0" の状態で、SBIxCR2<MST><TRX><BB><PIN>にそれぞれ"1"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、SBxSCL 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、SBxSDA 端子を解放し、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で、INTSBIx 割り込み要求が発生し、<PIN> = "0"にされます。

<PIN> = "0" の間、SBxSCL 端子を"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エ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.AND 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。
```

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

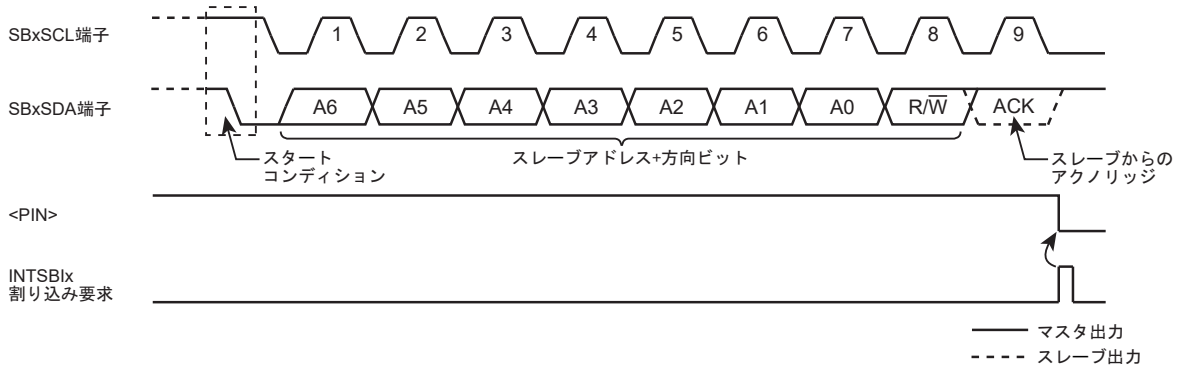


図 14-6 スタートコンディションとスレーブアドレスの発生

14.3.3.3　1 ワードのデータ転送

1 ワード転送終了の INTSBIx 割り込みの処理で<MST>をテストし、マスタモード/スレーブモードの判断をします。

(1)　マスタモードの場合(<MST> = "1")

<TRX>をテストし、トランスミッタ/レシーバの判断をします。

(a)　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 SBxSCL 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、SBxSDA 端子から 1 ワードのデータが転送されます。

転送終了後 INTSBIx 割り込み要求が発生し、<PIN>が"0"になり SBxSCL 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。

INTSBIx 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

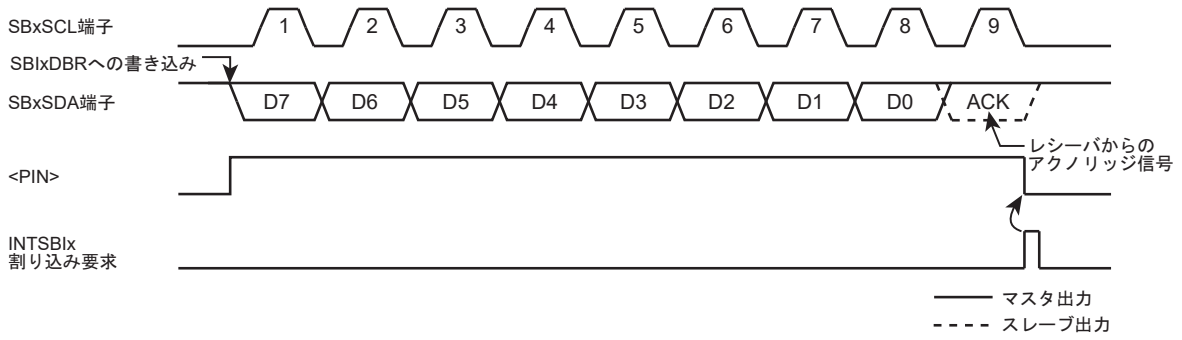


図 14-7 <BC[2:0]>="000",<ACK>="1"の場合（トランスミッタモード）

(b)　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]>を設定し、SBxSCL 端子を解放するために 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 SCL クロックを SBxSCL 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 SBxSDA 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 SBxSCL 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。

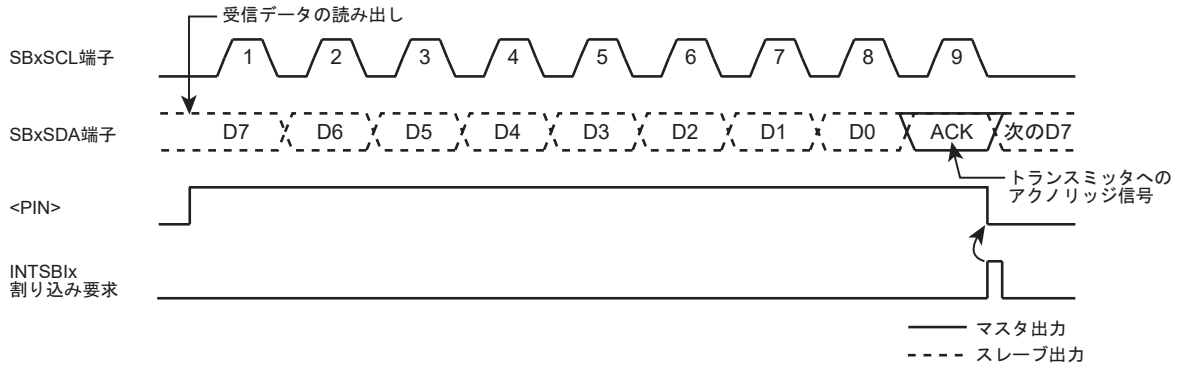


図 14-8 <BC[2:0]>="000",<ACK>="1"のときの例（レーシーバモード）

トランスミッタに対してデータの送信を終了させるときは、最後に受信したいデータの 1 ワード手前のデータを読み出す前に<ACK>を"0"にします。これにより、最終データのアクノリッジのためのクロックは発生されなくなります。転送終了の割り込み要求発生後の処理で、<BC[2:0]> = "001"に設定し、データを読み出すと、1 ビット転送のためのクロックを発生します。このときマスタはレシーバなのでバスの SDA ラインは"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。

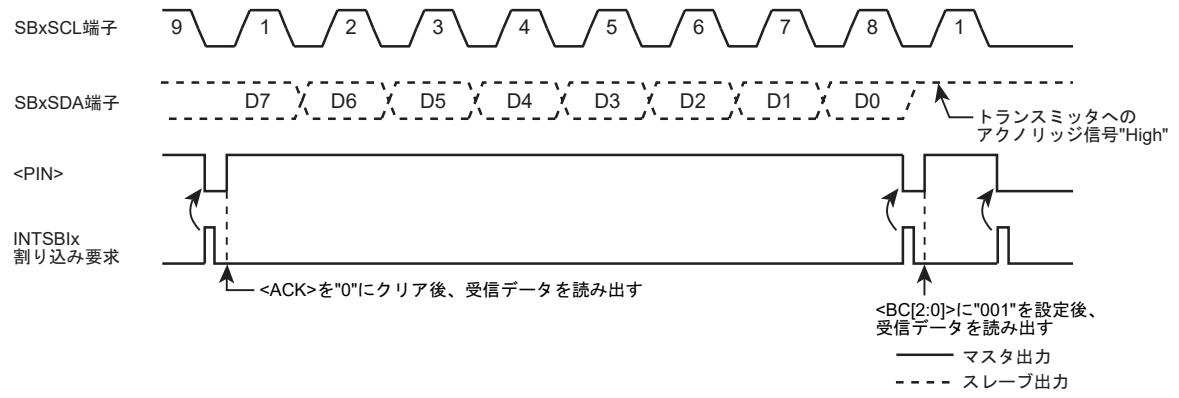


図 14-9 マスタレシーバモード時、データの送信を終了させるときの処理

例:データを N 回受信する場合

INTSBIx 割り込み(データ送信後)

		7	6	5	4	3	2	1	0	
SBIxCR1	←	X	X	X	X	0	X	X	X	受信データのビット数および ACK を設定します。
Reg.	←	SBIxDBR								ダミーデータを取り込みます。
割り込み終了										

INTSBIx 割り込み(データ受信 1~(N − 2)回目)

		7	6	5	4	3	2	1	0	
Reg.	←	SBIxDBR								1~(N − 2)回目のデータを取り込みます。
割り込み終了										

INTSBIx 割り込み(データ受信 (N − 1)回目)

		7	6	5	4	3	2	1	0	
SBIxCR1	←	X	X	X	0	0	X	X	X	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。
Reg.	←	SBIxDBR								(N − 1)回目のデータを取り込みます。
割り込み終了										

INTSBIx 割り込み(データ受信 N 回目)

		7	6	5	4	3	2	1	0	
SBIxCR1	←	0	0	1	0	0	X	X	X	1 ビット転送のためのクロックを発生します。
Reg.	←	SBIxDBR								N 回目のデータを取り込みます。
割り込み終了										

INTSBIx 割り込み(データ受信後)

ストップコンディションを発生する処理	データ転送を終了させます。
割り込み終了	

注) X; Don't care

(2) スレーブモードの場合(<MST> = "0")

スレーブモードのとき、マスタが送ったスレーブアドレス、またはゼネラルコールアドレスを受信したとき、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アドレス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、SBxSCL 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 SBxSCL 端子が t_{LOW} 後に開放されます。

なお、ゼネラルコールのセカンドバイトは内容によりアクノリッジ信号の発生有無を制御する必要がありますので、ソフトウエアによりアクノリッジ信号の制御を行なってください。

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 14-1 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  X  X  X  X     送信データをセットします。
```

注)　X; Don't care

表 14-1 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。
	0	1	0	スレーブレシーバモード時、マスタが送った方向ビットが"1"のスレーブアドレスを受信	
		0	0	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	1 ワードのビット数を<BC[2:0]>にセットし、受信データを SBIxDBR から読み出します。

14.3.3.4　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、SBxSDA 端子が立ち上がり、ストップコンディションが発生します。

```
                7  6  5  4  3  2  1  0
SBIxCR2     ←   1  1  0  1  1  0  0  0      ストップコンディションを発生させます。
```

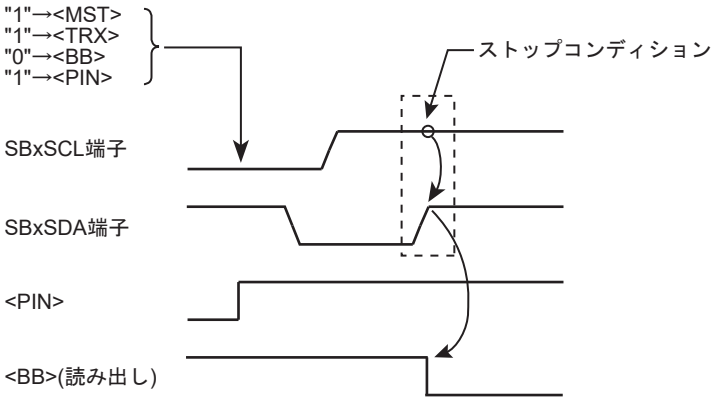


図 14-10 ストップコンディションの発生

14.3.3.5 再スタートの手順

再スタートはマスタデバイスがスレーブデバイスに対して、データ転送を終了させずに転送の方向を変化させるときに使用します。マスタモード時、再スタートを発生する場合の手順を以下に示します。

まず、SBIxCR2<MST>、<TRX>、<BB>に"0"、<PIN>に"1"を書き込み、バスを開放します。このとき SBxSDA 端子は"High"レベルを保ち、SBxSCL 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、SBxSCL 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「14.3.3.2 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)のソフトウエ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                           SBxSCL 端子の開放を確認します。
Then
if SBIxSR<LRB> ≠ 1                          他のデバイスの SBxSCL 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      アクノリッ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

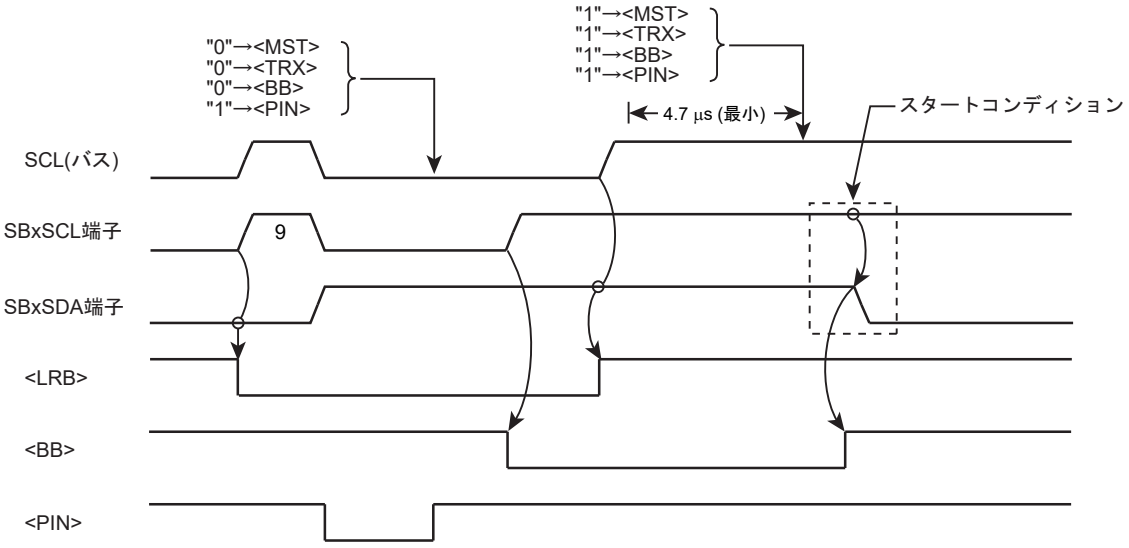
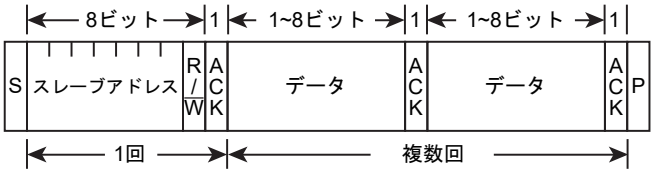


図 14-11 再スタートを発生する場合のタイミングチャート

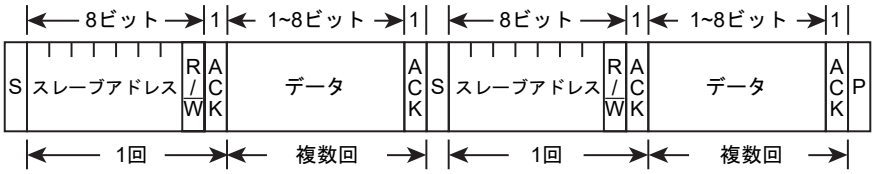
14.3.4 データフォーマット

I2C バスモード時のデータフォーマットを図 14-12 に示します。

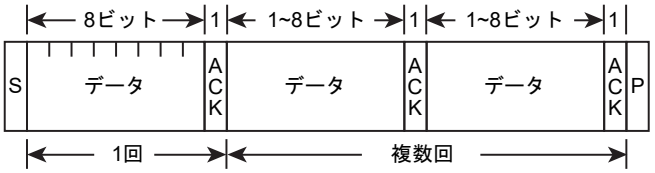
(a) アドレッシングフォーマット



(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 14-12 I2C バスモード時のデータフォーマット

14.3.5 マルチマスタで使用する際の注意点

マルチマスタで使用している際に通信がロックした場合に備え、ソフトウエアによるリカバリ対策を用意願います。

リカバリ処理例

1. 送信開始と共にタイムアウト検知用のタイマカウントを開始
2. 一定時間内にシリアルインタフェース割り込み（INTSBIx）が発生せず、タイムアウトとなった場合、通信がロックしたと判断
3. シリアルバスインタフェースのソフトウエアリセットを実行し通信ロックを解除
4. 送信タイミングの調整処理 (注)
5. 送信データを再送信

注) 再送信タイミングが重ならないよう、デバイス毎に送信タイミングを調整してください。

14.4　SIO モード

14.4.1　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。

14.4.1.1　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。

14.4.1.2 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)

000	n = 3	システムクロック: fsys クロックギア: fc/1 周波数 = $\frac{fsys/2}{2^n}$ [Hz]
001	n = 4	
010	n = 5	
011	n = 6	
100	n = 7	
101	n = 8	
110	n = 9	
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。

注 3) マスタモードで<BC[2:0]>="001"で<ACK>="0"のとき、ストップコンディション発生後の SCL ラインの立下りによって、SCL ラインが"L"に固定されることがあり、他のマスタデバイスがバスを使用できなくなります。複数のマスタデバイスがバスに接続されている場合、ストップコンディション発生前に、転送ビット数を"2"以上にしてください。

14.4.1.3　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。

14.4.1.4 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。

14.4.1.5　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)　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。

14.4.1.6 SBIxBR0(ボーレートレジスタ 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。

14.4.2　制御

14.4.2.1　シリアルクロック

(1)　クロックソース

SBIxCR1 <SCK[2:0]> により、次の選択ができます。

(a)　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 SBxSCK 端子より外部に出力されます。なお、転送開始時 SBxSCK 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随できないときには、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。

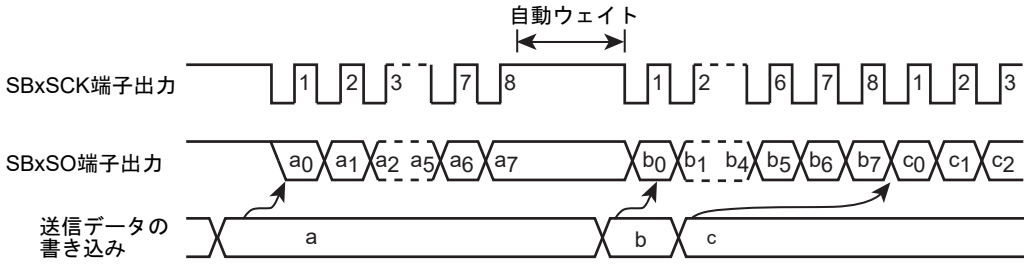
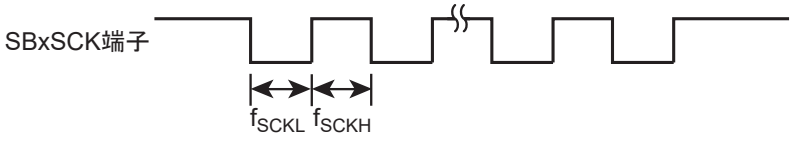


図 14-13 自動ウェイト機能

(b)　外部クロック(<SCK[2:0]> = "111")

外部から SBxSCK 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。



f_{SCKL}, f_{SCKH} > 4/fsys

図 14-14 外部クロック入力時の最大転送周波数

(2) シフトエッジ

送信は前縁シフト、受信は後縁シフトになります。

- 前縁シフト

  シリアルクロックの前縁(SBxSCK 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(SBxSCK 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。

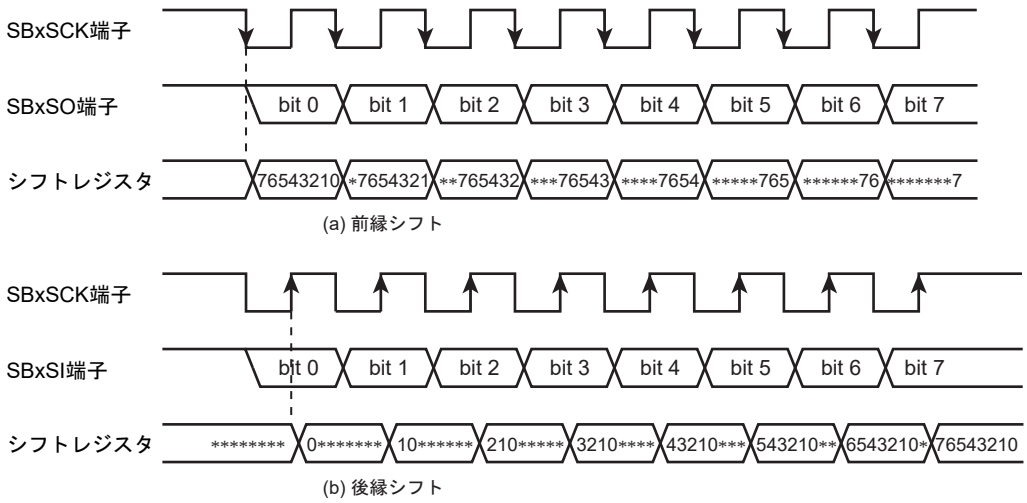


図 14-15 シフトエッジ

14.4.2.2　転送モード

SBIxCR1<SIOM[1:0]>で、送信/受信/送受信モードを選択します。

(1)　8 ビット送信モード

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 SBxSO 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 SBxSCK 端子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。

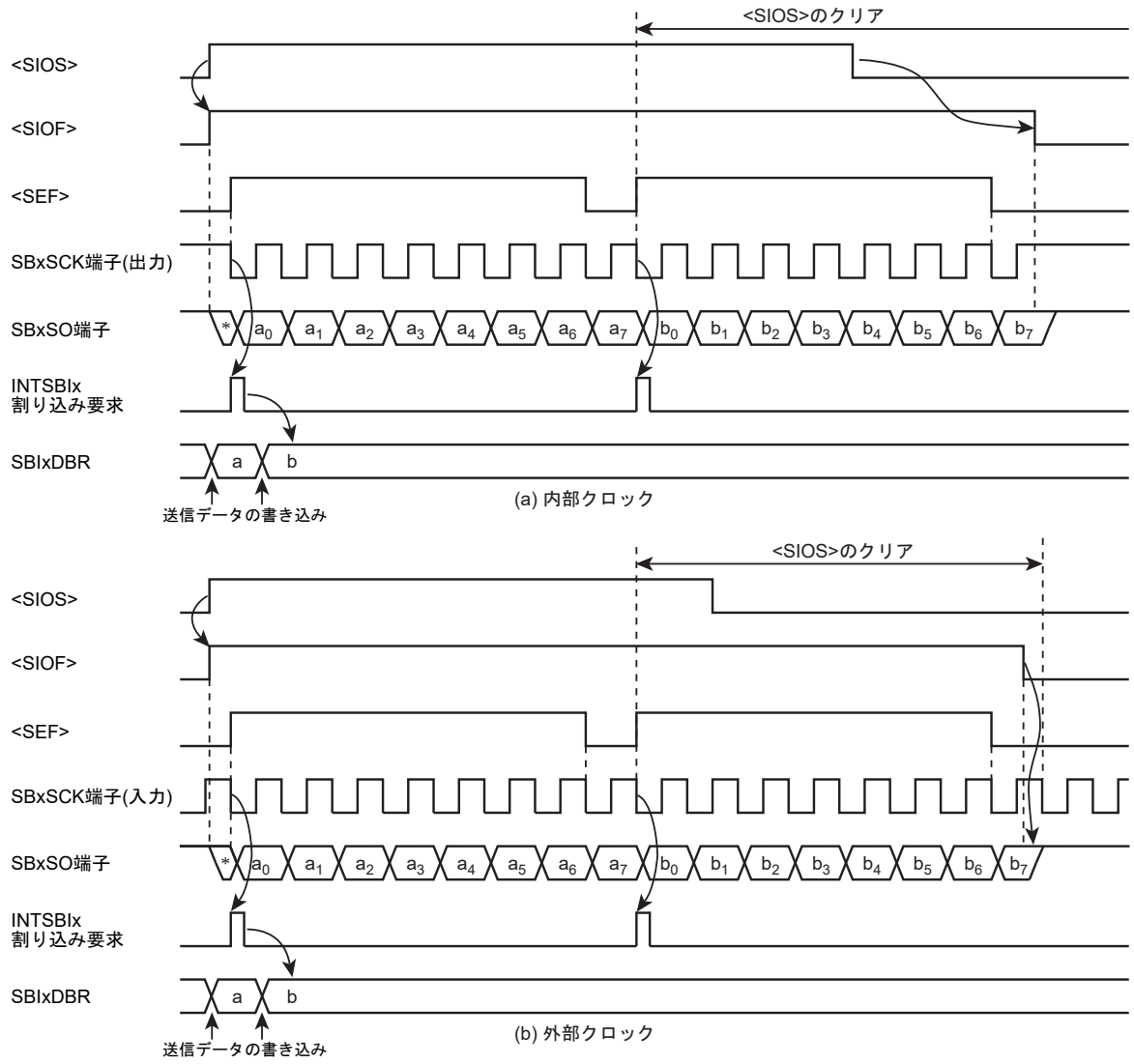


図 14-16 送信モード

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

(2)　8 ビット受信モード

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、SBxSI 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。

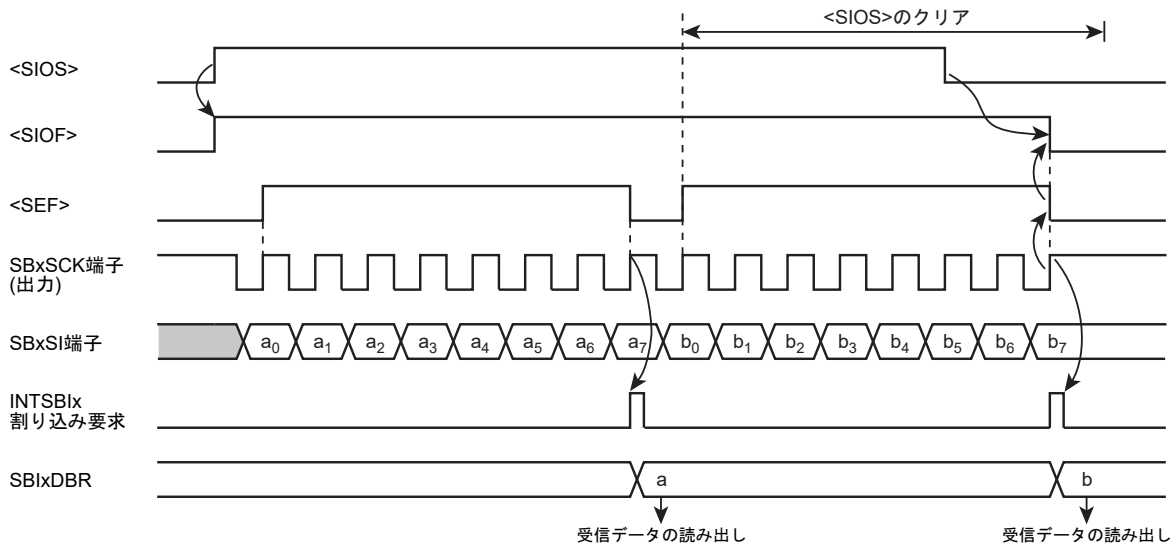


図 14-17 受信モード(例: 内部クロック)

(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 SBxSO 端子から出力され、立ち上がりで受信データが SBxSI 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 SBxSCK 端子の立ち下がりエッジまでの間、前回送信したデータの最後のビットと同じ値が出力されます。

送受信を終了させるには、INTSBIx 割り込みサービスプログラムで<SIOS> = "0"を書き込むか 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。

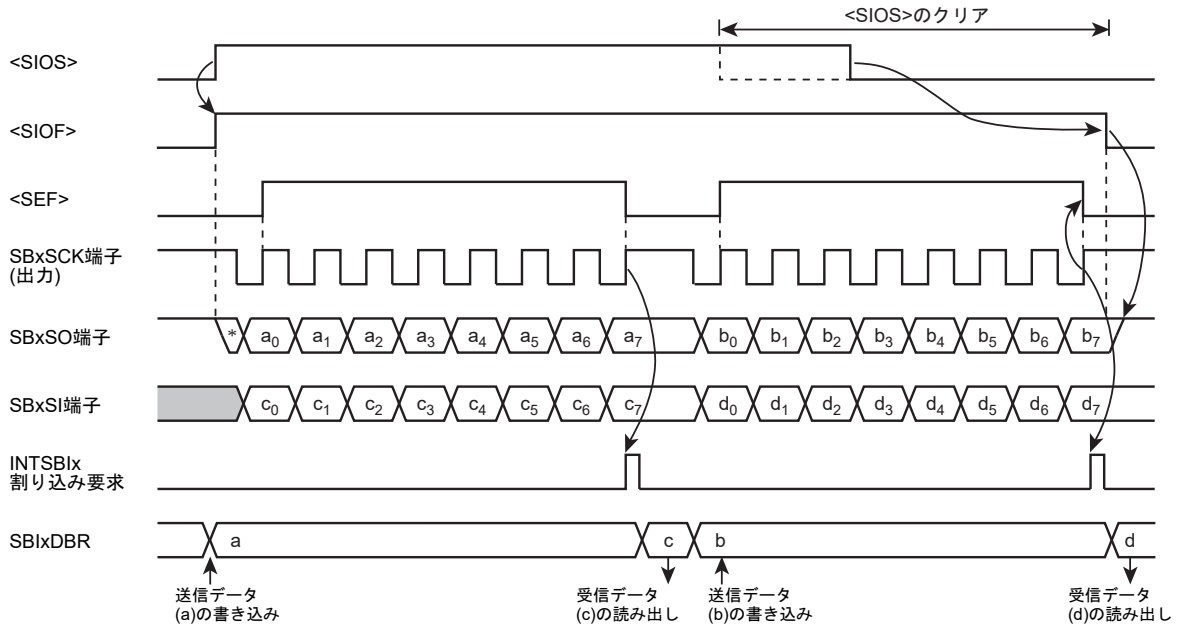


図 14-18 送受信モード(例: 内部クロック)

		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。

(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 SBxSCK 端子の立ち上がりに対するデータ保持時間は以下のようになります。送信モード、送受信モードとも同様です。

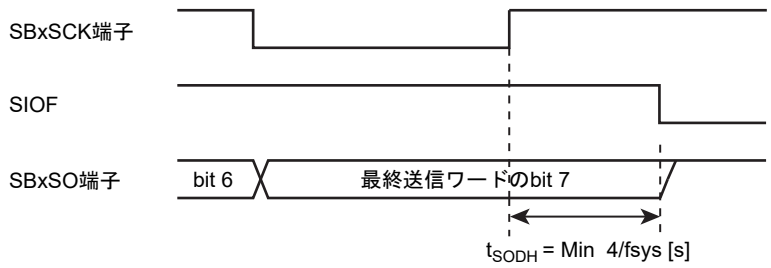


図 14-19 送信終了時の最終ビット保持時間

第 15 章　12 ビットアナログ/デジタルコンバータ

15.1　機能と特徴

1. PMD やタイマからのトリガ信号に同期して任意のアナログ入力を変換することができます。
2. ソフトウエア起動、常時起動において任意のアナログ入力を変換する事ができます。
3. AD 変換値レジスタが 12 個あります。
4. トリガ起動によるプログラム終了時に割り込みを発生できます。
5. ソフトウエア起動によるプログラム終了時に割り込みを発生できます。
6. AD 監視機能があります。有効時に比較条件と一致した場合は割り込みを発生します。

15.2　ブロック図

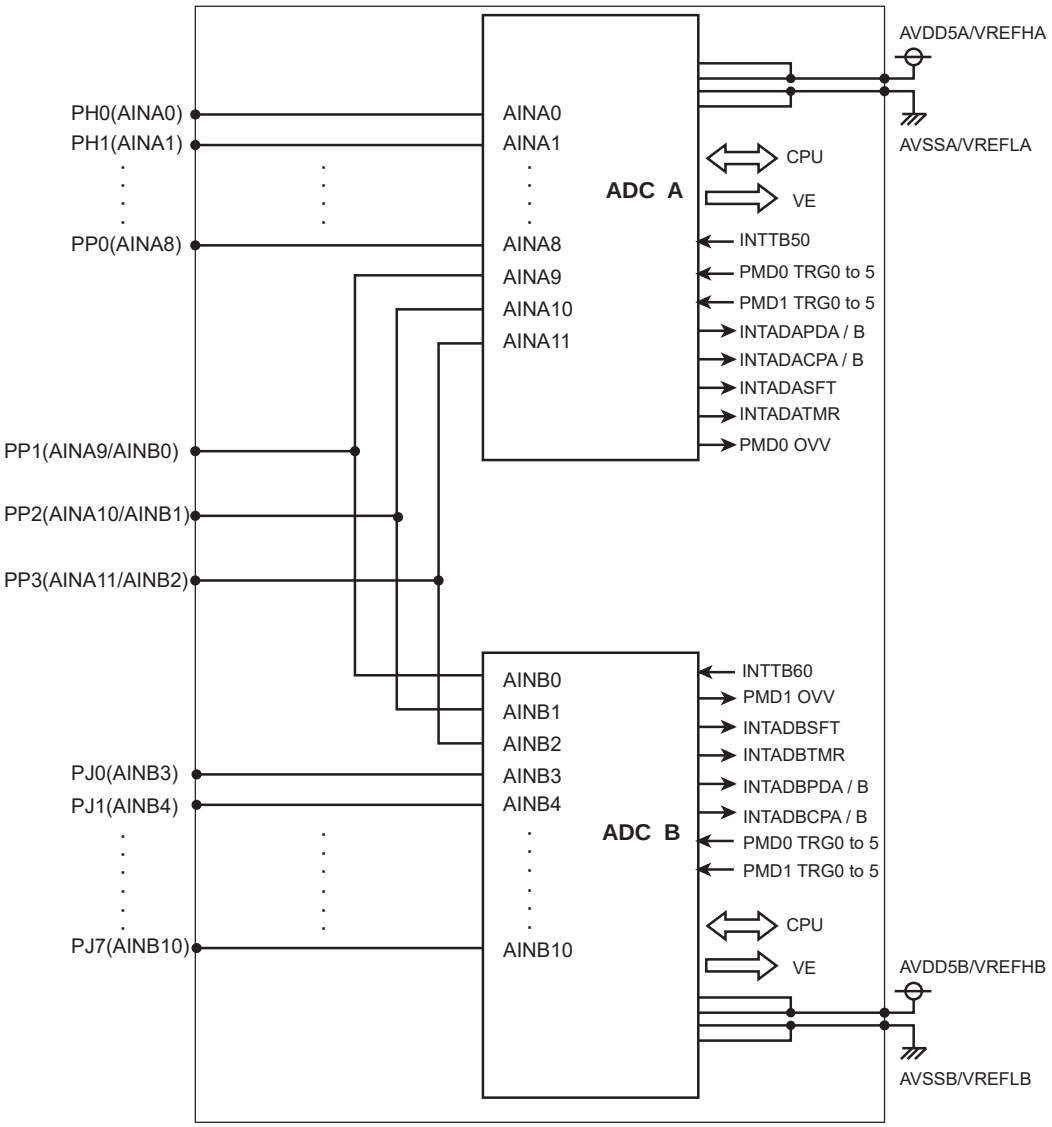


図 15-1 AD コンバータブロック図

15.3 レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名(x=A,B)		Address(Base+)
クロック設定レジスタ	ADxCLK	0x0000
モード設定レジスタ 0	ADxMOD0	0x0004
モード設定レジスタ 1	ADxMOD1	0x0008
モード設定レジスタ 2	ADxMOD2	0x000C
監視割り込み設定レジスタ 0	ADxCMPCR0	0x0010
監視割り込み設定レジスタ 1	ADxCMPCR1	0x0014
変換結果比較レジスタ 0	ADxCMP0	0x0018
変換結果比較レジスタ 1	ADxCMP1	0x001C
変換結果格納レジスタ 0	ADxREG0	0x0020
変換結果格納レジスタ 1	ADxREG1	0x0024
変換結果格納レジスタ 2	ADxREG2	0x0028
変換結果格納レジスタ 3	ADxREG3	0x002C
変換結果格納レジスタ 4	ADxREG4	0x0030
変換結果格納レジスタ 5	ADxREG5	0x0034
変換結果格納レジスタ 6	ADxREG6	0x0038
変換結果格納レジスタ 7	ADxREG7	0x003C
変換結果格納レジスタ 8	ADxREG8	0x0040
変換結果格納レジスタ 9	ADxREG9	0x0044
変換結果格納レジスタ 10	ADxREG10	0x0048
変換結果格納レジスタ 11	ADxREG11	0x004C
PMD トリガ用プログラム番号選択レジスタ 0	ADxPSEL0	0x0050
PMD トリガ用プログラム番号選択レジスタ 1	ADxPSEL1	0x0054
PMD トリガ用プログラム番号選択レジスタ 2	ADxPSEL2	0x0058
PMD トリガ用プログラム番号選択レジスタ 3	ADxPSEL3	0x005C
PMD トリガ用プログラム番号選択レジスタ 4	ADxPSEL4	0x0060
PMD トリガ用プログラム番号選択レジスタ 5	ADxPSEL5	0x0064
PMD トリガ用プログラム番号選択レジスタ 6	ADxPSEL6	0x0068
PMD トリガ用プログラム番号選択レジスタ 7	ADxPSEL7	0x006C
PMD トリガ用プログラム番号選択レジスタ 8	ADxPSEL8	0x0070
PMD トリガ用プログラム番号選択レジスタ 9	ADxPSEL9	0x0074
PMD トリガ用プログラム番号選択レジスタ 10	ADxPSEL10	0x0078
PMD トリガ用プログラム番号選択レジスタ 11	ADxPSEL11	0x007C
PMD トリガ用割り込み選択レジスタ 0	ADxPINTS0	0x0080
PMD トリガ用割り込み選択レジスタ 1	ADxPINTS1	0x0084
PMD トリガ用割り込み選択レジスタ 2	ADxPINTS2	0x0088
PMD トリガ用割り込み選択レジスタ 3	ADxPINTS3	0x008C
PMD トリガ用割り込み選択レジスタ 4	ADxPINTS4	0x0090
PMD トリガ用割り込み選択レジスタ 5	ADxPINTS5	0x0094
PMD トリガ用プログラム選択レジスタ 0	ADxPSET0	0x0098
PMD トリガ用プログラム選択レジスタ 1	ADxPSET1	0x009C
PMD トリガ用プログラム選択レジスタ 2	ADxPSET2	0x00A0
PMD トリガ用プログラム選択レジスタ 3	ADxPSET3	0x00A4

レジスタ名(x=A,B)		Address(Base+)
PMD トリガ用プログラム選択レジスタ 4	ADxPSET4	0x00A8
PMD トリガ用プログラム選択レジスタ 5	ADxPSET5	0x00AC
タイマトリガ用プログラムレジスタ 0 ~ 3	ADxTSET03	0x00B0
タイマトリガ用プログラムレジスタ 4 ~ 7	ADxTSET47	0x00B4
タイマトリガ用プログラムレジスタ 8 ~ 11	ADxTSET811	0x00B8
ソフトウエア トリガ用プログラムレジスタ0 ~ 3	ADxSSET03	0x00BC
ソフトウエア トリガ用プログラムレジスタ4 ~ 7	ADxSSET47	0x00C0
ソフトウエア トリガ用プログラムレジスタ8 ~ 11	ADxSSET811	0x00C4
常時変換用プログラムレジスタ 0 ~ 3	ADxASET03	0x00C8
常時変換用プログラムレジスタ 4 ~ 7	ADxASET47	0x00CC
常時変換用プログラムレジスタ 8 ~ 11	ADxASET811	0x00D0
モード設定レジスタ 3	ADxMOD3	0x00D4
モード設定レジスタ 4	ADxMOD4	0x00D8
モード設定レジスタ 5	ADxMOD5	0x00DC

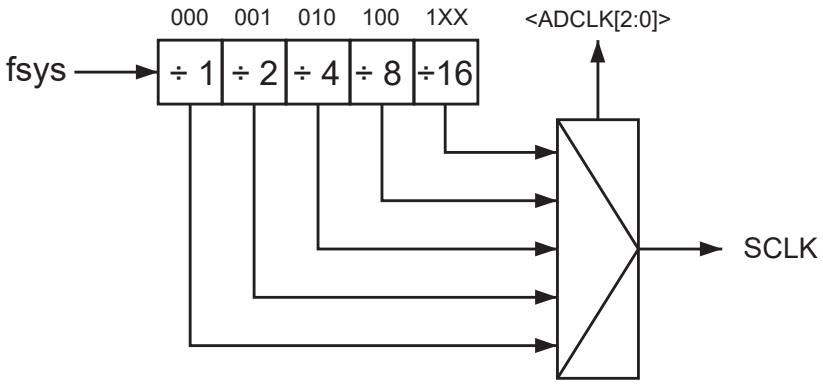
15.4 レジスタ詳細

AD 変換は AD コンバータクロック設定レジスタによって選択されたクロックで実行されます。

15.4.1 ADxCLK (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	-	EXAZ				VADCLK		
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。
6-3	EXAZ[3:0]	R/W	AIN サンプリング期間の引き伸ばし 0000: (SCLK 周期 ×34)×1 0001: (SCLK 周期 ×34)×2 0010: (SCLK 周期 ×34)×3 0011: (SCLK 周期 ×34)×4 0101: (SCLK 周期 ×34)×16 0111: (SCLK 周期 ×34)×64 1000: (SCLK 周期 ×34)×128 1001: (SCLK 周期 ×34)×256 1010: (SCLK 周期 ×34)×512 1011: (SCLK 周期 ×34)×1024 上記以外: Reserved
2-0	VADCLK[2:0]	R/W	AD プリスケーラ出力(SCLK)選択 000: fsys (注 1) 001: fsys/2 010: fsys/4 011: fsys/8 1xx: fsys/16



注 1)　SCLK は最大 120MHz です。

注 2)　AD 変換は上記レジスタで選択されたクロックで実行されますが、保証精度を満足する様に変換クロックを選択する必要があります。

注 3)　AD 変換中に、AD 変換クロック設定を変更しないで下さい。

15.4.2 ADxMOD0 (モード設定レジスタ 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	-	-	-	-	-	-	DACON	ADSS
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	DACON	R/W	DAC 制御 0: OFF 1: ON AD コンバータを使用する時には必ず<DACON>を"1"にセットしてください。
0	ADSS	W	ソフトウエア変換スタート 0: Don't care 1: 変換開始 ADxMOD1 レジスタの<ADEN>を"1"にセットして変換を許可し、<ADSS>を"1"にセットすると AD 変換を開始します。また、PMD トリガ，タイマ割り込み入力でも AD 変換を開始します。PMD トリガのタイミング設定，タイマ割り込みの設定はそれぞれ PMD,タイマの説明をご参照ください。

注) AD 変換をスタートさせる場合、ADxMOD0<DACON>を"1"に設定してから、ADxMOD1<ADEN>を"1"(AD 変換を許可)にした後、ソフトウエア変換スタートまたは、常時 AD 変換許可設定を行ってください。
なお、ADxMOD0<DACON>を"1"に設定した後、3μs の安定時間が必要です。

15.4.3　ADxMOD1 (モード設定レジスタ 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	ADEN	-	-	-	-	-	-	ADAS
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	ADEN	R/W	AD 変換許可/禁止 0: 禁止 1: 許可 <ADEN>のセットで AD を変換許可します。
6-1	−	R	リードすると"0"が読めます。
0	ADAS	R/W	常時 AD 変換許可 0: 変換禁止 1: 変換許可 AD 変換許可状態(<ADEN>=1)で、<ADAS>をセットすると常時 AD 変換を開始します。

15.4.4 ADxMOD2 (モード設定レジスタ 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	-	-	-	-	-	-	ADSFN	ADBFN
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	ADSFN	R	ソフトウエア変換フラグ 0: 終了 1: ソフト変換中 <ADSFN>はソフトウエア AD 変換 Busy フラグで、<ADSS>が"1"にセットされた後、AD 変換が実際に開始されると"1"にセットされ、ソフトウエアトリガによるすべての変換が終了すると"0" にクリアされます。
0	ADBFN	R	AD 変換 BUSY フラグ 0: 変換停止 1: 変換中 <ADBFN>は AD 変換 Busy フラグで、変換要因(PMD,タイマ,ソフトウエア,常時)に関係なく変換が開始されると"1"にセットされ、変換が終了すると"0"にクリアされます。

15.4.5　ADxMOD3 (モード設定レジスタ 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	-	PBSEL	-	BINMOD	BITS		-	RCUT
リセット後	0	0	0	0	0	0	0	1
	7	6	5	4	3	2	1	0
bit symbol	BIAS			-	-	-	-	-
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	−	R/W	"0"をライトしてください。
14	PBSEL	R/W	"0"をライトしてください。
13	−	R	リードすると"0"が読めます。
12	BINMOD	R/W	"0"をライトしてください。
11-10	BITS[1:0]	R/W	"00"をライトしてください。
9	−	R/W	"0"をライトしてください。
8	RCUT	R/W	低消費電力モード選択 0: 通常動作 1: Iref cut ADC を動作させる時はあらかじめ、"0"をライトしてください。 ADC が動作停止の間、"1"に設定することで電力消費を低減することができます。
7-5	BIAS[2:0]	R/W	"000"をライトしてください。
4-0	−	R	リードすると"0"が読めます。

15.4.6 ADxMOD4 (モード設定レジスタ 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	-	-	-	-	PHASEC			
リセット後	0	0	0	0	0	0	1	0
	7	6	5	4	3	2	1	0
bit symbol	PHASEB				-	PHASEA		
リセット後	0	0	0	1	0	0	1	0

Bit	Bit Symbol	Type	機能
31-12	−	R	リードすると"0"が読めます。
11-8	PHASEC[3:0]	R/W	"0001"をライトしてください。
7-4	PHASEB[3:0]	R/W	"0001"をライトしてください。
3	−	R	リードすると"0"が読めます。
2-0	PHASEA[2:0]	R/W	"001"をライトしてください。

15.4.7　ADxMOD5 (モード設定レジスタ 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	PHASEF			PHASEE				
リセット後	0	1	1	0	0	1	1	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	PHASED				
リセット後	0	0	0	0	1	0	0	1

Bit	Bit Symbol	Type	機能
31-16	−	R	リードすると"0"が読めます。
15-13	PHASEF[2:0]	R/W	"011"をライトしてください。
12-8	PHASEE[4:0]	R/W	"00110"をライトしてください。
7-5	−	R	リードすると"0"が読めます。
4-0	PHASED[4:0]	R/W	"01101"をライトしてください。

15.4.8 ADxCMPCR0(監視割り込み設定レジスタ 0)

判定が確定すると割り込み(INTADxCPn)を発生します。(n=A,B、A:監視 0、B:監視 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	-	-	-	-	CMPCNT0			
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMP0EN	-	-	ADBIG0	REGS0			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-12	-	R	リードすると"0"が読めます。
11-8	CMPCNT0[3:0]	R/W	判定確定するまでの比較回数を設定 0: 毎回有効 1: 2 回以上有効 ・ ・ 15: 16 回以上有効 AD 変換結果と ADCMP に設定した値との比較の許可禁止と比較する値の格納されたレジスタを選択します。
7	CMP0EN	R/W	AD 監視機能 0:禁止 1:許可
6-5	-	R	リードすると"0"が読めます。
4	ADBIG0	R/W	大小判定設定 0:比較 REG より大 1:比較 REG より小
3-0	REGS0[3:0]	R/W	比較する AD 変換結果格納レジスタ 0000: ADxREG0 / 0001: ADxREG1 / 0010: ADxREG2 / 0011: ADxREG3 0100: ADxREG4 / 0101: ADxREG5 / 0110: ADxREG6 / 0111: ADxREG7 1000: ADxREG8 / 1001: ADxREG9 / 1010: ADxREG10 / 1011: ADxREG11

15.4.9　ADxCMPCR1(監視割り込み設定レジスタ 1)

判定が確定すると割り込み(INTADxCPn)を発生します。(n=A,B、A:監視 0、B:監視 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	-	-	-	-	CMPCNT1			
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMP1EN	-	-	ADBIG1	REGS1			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-12	-	R	リードすると"0"が読めます。
11-8	CMPCNT1[3:0]	R/W	判定確定するまでの比較回数を設定 0: 毎回有効 1: 2 回以上有効 ・ ・ 15: 16 回以上有効 AD 変換結果と ADCMP に設定した値との比較の許可禁止と比較する値の格納されたレジスタを選択します。
7	CMP1EN	R/W	A/D 監視機能 0:禁止 1:許可
6-5	-	R	リードすると"0"が読めます。
4	ADBIG1	R/W	大小判定設定 0:比較 REG より大 1:比較 REG より小
3-0	REGS1[3:0]	R/W	比較する AD 変換結果格納レジスタ 0000: ADxREG0　0100: ADxREG4　1000: ADxREG8 0001: ADxREG1　0101: ADxREG5　1001: ADxREG9 0010: ADxREG2　0110: ADxREG6　1010: ADxREG10 0011: ADxREG3　0111: ADxREG7　1011: ADxREG11

15.4.10 ADxCMP0(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	AD0CMP0							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	AD0CMP0				-	-	-	-
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。
15-4	AD0CMP0[11:0]	R/W	AD 変換結果比較値格納 AD 変換結果と比較する値を設定します。
3-0	-	R	リードすると"0"が読めます。

15.4.11 ADxCMP1(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	AD0CMP1							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	AD0CMP1				-	-	-	-
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。
15-4	AD0CMP1[11:0]	R/W	AD 変換結果比較値格納 AD 変換結果と比較する値を設定します。
3-0	-	R	リードすると"0"が読めます。

15.4.12　ADxREG0(変換結果格納レジスタ 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	ADR0							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR0				-	-	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
15-4	ADR0[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR0	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG0 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG0 レジスタをリードすると"0"にクリアされます。
0	ADR0RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG0 レジスタをリードすると"0"にクリアされます。

15.4.13 ADxREG1(変換結果格納レジスタ 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	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
15-4	ADR1[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR1	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG1 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG1 レジスタをリードすると"0"にクリアされます。
0	ADR1RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG1 レジスタをリードすると"0"にクリアされます。

15.4.14　ADxREG2(変換結果格納レジスタ 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	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
15-4	ADR2[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR2	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG2 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG2 レジスタをリードすると"0"にクリアされます。
0	ADR2RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG2 レジスタをリードすると"0"にクリアされます。

15.4.15 ADxREG3(変換結果格納レジスタ 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	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
15-4	ADR3[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR3	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG3 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG3 レジスタをリードすると"0"にクリアされます。
0	ADR3RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG3 レジスタをリードすると"0"にクリアされます。

15.4.16　ADxREG4(変換結果格納レジスタ 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	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
15-4	ADR4[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR4	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG4 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG4 レジスタをリードすると"0"にクリアされます。
0	ADR4RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG4 レジスタをリードすると"0"にクリアされます。

15.4.17 ADxREG5(変換結果格納レジスタ 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	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
15-4	ADR5[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR5	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG5 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG5 レジスタをリードすると"0"にクリアされます。
0	ADR5RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG5 レジスタをリードすると"0"にクリアされます。

15.4.18　ADxREG6(変換結果格納レジスタ 6)

	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
15-4	ADR6[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR6	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG6 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG6 レジスタをリードすると"0"にクリアされます。
0	ADR6RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG6 レジスタをリードすると"0"にクリアされます。

15.4.19 ADxREG7(変換結果格納レジスタ 7)

	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
15-4	ADR7[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR7	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG7 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG7 レジスタをリードすると"0"にクリアされます。
0	ADR7RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG7 レジスタをリードすると"0"にクリアされます。

15.4.20　ADxREG8(変換結果格納レジスタ 8)

	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	ADR8							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR8				-	-	OVR8	ADR8RF
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
15-4	ADR8[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR8	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG8 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG8 レジスタをリードすると"0"にクリアされます。
0	ADR8RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG8 レジスタをリードすると"0"にクリアされます。

15.4.21 ADxREG9(変換結果格納レジスタ 9)

	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	ADR9							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR9				-	-	OVR9	ADR9RF
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
15-4	ADR9[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR9	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG9 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG9 レジスタをリードすると"0"にクリアされます。
0	ADR9RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG9 レジスタをリードすると"0"にクリアされます。

15.4.22　ADxREG10(変換結果格納レジスタ 10)

	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	ADR10							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR10				-	-	OVR10	ADR10RF
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
15-4	ADR10[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR10	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG10 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG10 レジスタをリードすると"0"にクリアされます。
0	ADR10RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG10 レジスタをリードすると"0"にクリアされます。

15.4.23 ADxREG11(変換結果格納レジスタ 11)

	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	ADR11							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ADR11				-	-	OVR11	ADR11RF
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
15-4	ADR11[11:0]	R	AD 変換結果値格納
3-2	-	R	リードすると"0"が読めます
1	OVR11	R	Over Run フラグ 0：発生なし 1：発生あり ADxREG11 レジスタを読み出す前に AD 変換結果が上書きされると"1"にセットされます。このフラグは ADxREG11 レジスタをリードすると"0"にクリアされます。
0	ADR11RF	R	AD 変換結果格納フラグ 0：変換結果なし 1：変換結果あり AD 変換値が格納されると"1"にセットされます。このフラグは ADxREG11 レジスタをリードすると"0"にクリアされます。

15.4.24　PMD トリガ用プログラムレジスタ

本 AD コンバータは PMD 回路が発生するトリガ信号によって AD 変換を開始する事ができます。

PMD トリガ用プログラムレジスタは PMD が発生する 12 種類のトリガによって起動するプログラム番号の設定、プログラム終了時に発生する割り込みの制御、AD 変換の AIN の設定などを行います。

PMD トリガ用プログラムレジスタは 3 種類のレジスタから構成されます。

（x=A,B : AD コンバータユニット）

・ PMD トリガ用プログラム番号選択レジスタ(ADxPSEL0 ~ ADxPSEL11)

PMD からの 12 本のトリガ信号(PMD0TRG0 ~ 5，PMD1TRG0 ~ 5)に対して、それぞれ起動するプログラム番号(0 ~ 5)を選択するレジスタです。

ADxPSEL0 ~ ADxPSEL5 が PMD0TRG0 ~ 5、ADxPSEL6 ~ ADxPSEL11 が PMD1TRG0 ~ 5 に対応しています。

・ PMD トリガ用割り込み選択レジスタ(ADxPINTS0 ~ ADxPINTS5)

それぞれのプログラム番号ごとにプログラム終了時に発生させる割り込みの有無や種類(INTADxPDA,INTADxPDB)を選択するレジスタです。

ADxPINTS0 がプログラム 0 に対応しており、ADxPINTS5(プログラム 5)まであります。

・ PMD トリガ用プログラム選択レジスタ(ADxPSET0 ~ ADxPSET5)

プログラム番号(0 ~ 5)に対して、AD 変換する AIN 番号とベクトルエンジンへの通知用に U/V/W の相を設定します。プログラム番号 1 つに対して PMD トリガ用プログラム選択レジスタは 4 組あり、変換結果は、それぞれ変換結果格納レジスタ 0 ~ 3(ADxREG0 ~ 3)に格納されます。

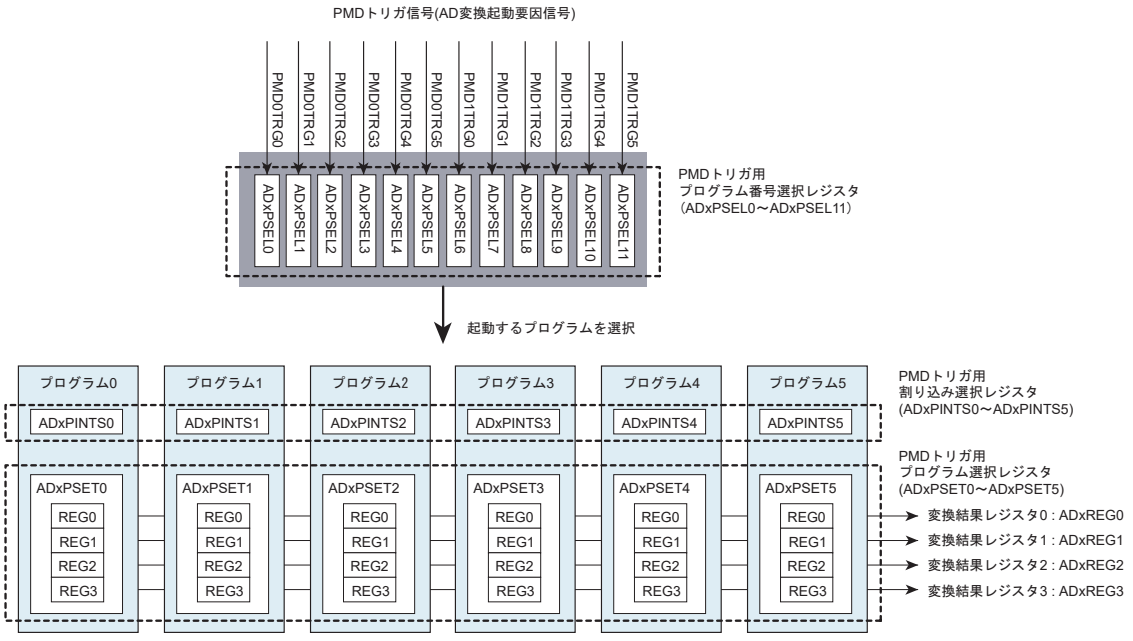


図 15-2 PMD トリガ用プログラムレジスタ

15.4.24.1 ADxPSEL0 ~ ADxPSEL11(PMD トリガ用プログラム番号選択レジスタ 0 ~ 11)

ADxPSEL0：PMD トリガ用プログラム番号選択レジスタ 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	PENS0	-	-	-	-	PMDS0		
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	PENS0	R/W	PMD0TRG0 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS0[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL1：PMD トリガ用プログラム番号選択レジスタ 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	PENS1	-	-	-	-	PMDS1		
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	PENS1	R/W	PMD0TRG1 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS1[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL2：PMD トリガ用プログラム番号選択レジスタ 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	PENS2	-	-	-	-	PMDS2		
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	PENS2	R/W	PMD0TRG2 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS2[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL3：PMD トリガ用プログラム番号選択レジスタ 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	PENS3	-	-	-	-	PMDS3		
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	PENS3	R/W	PMD0TRG3 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS3[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL4：PMD トリガ用プログラム番号選択レジスタ 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	PENS4	-	-	-	-	PMDS4		
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	PENS4	R/W	PMD0TRG4 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS4[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL5：PMD トリガ用プログラム番号選択レジスタ 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	PENS5	-	-	-	-	PMDS5		
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	PENS5	R/W	PMD0TRG5 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS5[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL6：PMD トリガ用プログラム番号選択レジスタ 6

	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	PENS6	-	-	-	-	PMDS6		
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	PENS6	R/W	PMD1TRG0 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS6[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL7：PMD トリガ用プログラム番号選択レジスタ 7

	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	PENS7	-	-	-	-	PMDS7		
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	PENS7	R/W	PMD1TRG1 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS7[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL8：PMD トリガ用プログラム番号選択レジスタ 8

	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	PENS8	-	-	-	-	PMDS8		
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	PENS8	R/W	PMD1TRG2 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS8[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL9：PMD トリガ用プログラム番号選択レジスタ 9

	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	PENS9	-	-	-	-	PMDS9		
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	PENS9	R/W	PMD1TRG3 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS9[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL10：PMD トリガ用プログラム番号選択レジスタ 10

	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	PENS10	-	-	-	-	PMDS10		
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	PENS10	R/W	PMD1TRG4 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS10[2:0]	R/W	プログラム番号選択(表 15-1 参照)

ADxPSEL11：PMD トリガ用プログラム番号選択レジスタ 11

	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	PENS11	-	-	-	-	PMDS11		
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	PENS11	R/W	PMD1TRG5 トリガ制御 0:ディセーブル 1:イネーブル
6-3	-	R	リードすると"0"が読めます。
2-0	PMDS11[2:0]	R/W	プログラム番号選択(表 15-1 参照)

表 15-1 プログラム番号選択

<PMDS0[2:0]>~ <PMDS11[2:0]>	
000	プログラム 0
001	プログラム 1
010	プログラム 2
011	プログラム 3
100	プログラム 4
101	プログラム 5
110	Reserved
111	Reserved

15.4.24.2　ADxPINTS0 ~ 5(PMD トリガ用割り込み選択レジスタ 0 ~ 5)

ADxPINTS0：PMD トリガ用割り込み選択レジスタ 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	-	-	-	-	-	-	INTSEL0	
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。
1-0	INTSEL0[1:0]	R/W	割込み選択 00:割込み出力なし 01:INTADxPDA 10:INTADxPDB 11: 割込み出力なし プログラム 0 に対して、起動する割り込みを選択します。

ADxPINTS1：PMD トリガ用割り込み選択レジスタ 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	-	-	-	-	-	-	INTSEL1	
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。
1-0	INTSEL1[1:0]	R/W	割込み選択 00:割込み出力なし 01:INTADxPDA 10:INTADxPDB 11: 割込み出力なし プログラム 1 に対して、起動する割り込みを選択します。

ADxPINTS2：PMD トリガ用割り込み選択レジスタ 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	-	-	-	-	-	-	INTSEL2	
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。
1-0	INTSEL2[1:0]	R/W	割込み選択 00:割込み出力なし 01:INTADxPDA 10:INTADxPDB 11: 割込み出力なし プログラム 2 に対して、起動する割り込みを選択します。

ADxPINTS3：PMD トリガ用割り込み選択レジスタ 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	-	-	-	-	-	-	INTSEL3	
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。
1-0	INTSEL3[1:0]	R/W	割込み選択 00:割込み出力なし 01:INTADxPDA 10:INTADxPDB 11: 割込み出力なし プログラム 3 に対して、起動する割り込みを選択します。

ADxPINTS4：PMD トリガ用割り込み選択レジスタ 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	-	-	-	-	-	-	INTSEL4	
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。
1-0	INTSEL4[1:0]	R/W	割込み選択 00:割込み出力なし 01:INTADxPDA 10:INTADxPDB 11: 割込み出力なし プログラム 4 に対して、起動する割り込みを選択します。

ADxPINTS5：PMD トリガ用割り込み選択レジスタ 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	-	-	-	-	-	-	INTSEL5	
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。
1-0	INTSEL5[1:0]	R/W	割込み選択 00:割込み出力なし 01:INTADxPDA 10:INTADxPDB 11: 割込み出力なし プログラム 5 に対して、起動する割り込みを選択します。

15.4.24.3　ADxPSET0 ~ 5(PMD トリガ用プログラム選択レジスタ 0 ~ 5)

各 ADxPSETn（n=0 ~ 5：プログラム番号）は、AD 変換入力端子の選択をする<AINSPnm [4:0]>とベクトルエンジンの相選択をする<UVWISnm[1:0]>および<ENSPnm>を 1 組とした 4 つのセットで構成されます。(m=0 ~ 3)（x ＝ A,B : AD コンバータユニット）

ADxREGm ADxPSETn	m=0	m=1	m=2	m=3
n=0	<ENSP00> <UVWIS00> <AINSP00>	<ENSP01> <UVWIS01> <AINSP01>	<ENSP02> <UVWIS02> <AINSP02>	<ENSP03> <UVWIS03> <AINSP03>
n=1	<ENSP10> <UVWIS10> <AINSP10>	<ENSP11> <UVWIS11> <AINSP11>	<ENSP12> <UVWIS12> <AINSP12>	<ENSP13> <UVWIS13> <AINSP13>
n=2	<ENSP20> <UVWIS20> <AINSP20>	<ENSP21> <UVWIS21> <AINSP21>	<ENSP22> <UVWIS22> <AINSP22>	<ENSP23> <UVWIS23> <AINSP23>
n=3	<ENSP30> <UVWIS30> <AINSP30>	<ENSP31> <UVWIS31> <AINSP31>	<ENSP32> <UVWIS32> <AINSP32>	<ENSP33> <UVWIS33> <AINSP33>
n=4	<ENSP40> <UVWIS40> <AINSP40>	<ENSP41> <UVWIS41> <AINSP41>	<ENSP42> <UVWIS42> <AINSP42>	<ENSP43> <UVWIS43> <AINSP43>
n=5	<ENSP50> <UVWIS50> <AINSP50>	<ENSP51> <UVWIS51> <AINSP51>	<ENSP52> <UVWIS52> <AINSP52>	<ENSP53> <UVWIS53> <AINSP53>

表 15-2 AD 変換入力端子の選択

<AINSP00 [4:0]> ~ <AINSP53 [4:0]>	AD コンバータユニット A	AD コンバータユニット B
0_0000	:AINA0	:AINB0
0_0001	:AINA1	:AINB1
0_0010	:AINA2	:AINB2
0_0011	:AINA3	:AINB3
0_0100	:AINA4	:AINB4
0_0101	:AINA5	:AINB5
0_0110	:AINA6	:AINB6
0_0111	:AINA7	:AINB7
0_1000	:AINA8	:AINB8
0_1001	:AINA9	:AINB9
0_1010	:AINA10	:AINB10
0_1011	:AINA11	:Reserved
0_1100 ~ 1_1111	:Reserved	

ADxPSET0：PMD トリガ用プログラム選択レジスタ 0

	31	30	29	28	27	26	25	24
bit symbol	ENSP03	UVWIS03		AINSP03				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSP02	UVWIS02		AINSP02				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSP01	UVWIS01		AINSP01				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSP00	UVWIS00		AINSP00				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSP03	R/W	ADxREG3 イネーブル 0:禁止 1:許可
30-29	UVWIS03[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
28-24	AINSP03[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
23	ENSP02	R/W	ADxREG2 イネーブル 0:禁止 1:許可
22-21	UVWIS02[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
20-16	AINSP02[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
15	ENSP01	R/W	ADxREG1 イネーブル 0:禁止 1:許可
14-13	UVWIS01[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
12-8	AINSP01[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
7	ENSP00	R/W	ADxREG0 イネーブル 0:禁止 1:許可
6-5	UVWIS00[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
4-0	AINSP00[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照

相選択

00	指定なし
01	U 相
10	V 相
11	W 相

ADxPSET1：PMD トリガ用プログラム選択レジスタ 1

	31	30	29	28	27	26	25	24
bit symbol	ENSP13	UVWIS13		AINSP13				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSP12	UVWIS12		AINSP12				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSP11	UVWIS11		AINSP11				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSP10	UVWIS10		AINSP10				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSP13	R/W	ADxREG3 イネーブル 0:禁止 1:許可
30-29	UVWIS13[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
28-24	AINSP13[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
23	ENSP12	R/W	ADxREG2 イネーブル 0:禁止 1:許可
22-21	UVWIS12[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
20-16	AINSP12[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
15	ENSP11	R/W	ADxREG1 イネーブル 0:禁止 1:許可
14-13	UVWIS11[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
12-8	AINSP11[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
7	ENSP10	R/W	ADxREG0 イネーブル 0:禁止 1:許可
6-5	UVWIS10[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
4-0	AINSP10[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照

相選択

00	指定なし
01	U 相
10	V 相
11	W 相

ADxPSET2：PMD トリガ用プログラム選択レジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	ENSP23	UVWIS23		AINSP23				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSP22	UVWIS22		AINSP22				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSP21	UVWIS21		AINSP21				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSP20	UVWIS20		AINSP20				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSP23	R/W	ADxREG3 イネーブル 0:禁止 1:許可
30-29	UVWIS23[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
28-24	AINSP23[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
23	ENSP22	R/W	ADxREG2 イネーブル 0:禁止 1:許可
22-21	UVWIS22[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
20-16	AINSP22[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
15	ENSP21	R/W	ADxREG1 イネーブル 0:禁止 1:許可
14-13	UVWIS21[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
12-8	AINSP21[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
7	ENSP20	R/W	ADxREG0 イネーブル 0:禁止 1:許可
6-5	UVWIS20[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
4-0	AINSP20[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照

相選択

00	指定なし
01	U 相
10	V 相
11	W 相

ADxPSET3：PMD トリガ用プログラム選択レジスタ 3

	31	30	29	28	27	26	25	24
bit symbol	ENSP33	UVWIS33		AINSP33				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSP32	UVWIS32		AINSP32				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSP31	UVWIS31		AINSP31				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSP30	UVWIS30		AINSP30				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSP33	R/W	ADxREG3 イネーブル 0:禁止 1:許可
30-29	UVWIS33[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
28-24	AINSP33[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
23	ENSP32	R/W	ADxREG2 イネーブル 0:禁止 1:許可
22-21	UVWIS32[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
20-16	AINSP32[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
15	ENSP31	R/W	ADxREG1 イネーブル 0:禁止 1:許可
14-13	UVWIS31[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
12-8	AINSP31[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
7	ENSP30	R/W	ADxREG0 イネーブル 0:禁止 1:許可
6-5	UVWIS30[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
4-0	AINSP30[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照

相選択

00	指定なし
01	U 相
10	V 相
11	W 相

ADxPSET4：PMD トリガ用プログラム選択レジスタ 4

	31	30	29	28	27	26	25	24
bit symbol	ENSP43	UVWIS43		AINSP43				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSP42	UVWIS42		AINSP42				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSP41	UVWIS41		AINSP41				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSP40	UVWIS40		AINSP40				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSP43	R/W	ADxEG3 イネーブル 0:禁止 1:許可
30-29	UVWIS43[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
28-24	AINSP43[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
23	ENSP42	R/W	ADxREG2 イネーブル 0:禁止 1:許可
22-21	UVWIS42[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
20-16	AINSP42[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
15	ENSP41	R/W	ADxREG1 イネーブル 0:禁止 1:許可
14-13	UVWIS41[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
12-8	AINSP41[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
7	ENSP40	R/W	ADxREG0 イネーブル 0:禁止 1:許可
6-5	UVWIS40[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
4-0	AINSP40[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照

相選択

00	指定なし
01	U 相
10	V 相
11	W 相

ADxPSET5：PMD トリガ用プログラム選択レジスタ 5

	31	30	29	28	27	26	25	24
bit symbol	ENSP53	UVWIS53		AINSP53				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSP52	UVWIS52		AINSP52				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSP51	UVWIS51		AINSP51				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSP50	UVWIS50		AINSP50				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSP53	R/W	ADxREG3 イネーブル 0:禁止 1:許可
30-29	UVWIS53[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
28-24	AINSP53[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
23	ENSP52	R/W	ADxREG2 イネーブル 0:禁止 1:許可
22-21	UVWIS52[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
20-16	AINSP52[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
15	ENSP51	R/W	ADxREG1 イネーブル 0:禁止 1:許可
14-13	UVWIS51[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
12-8	AINSP51[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照
7	ENSP50	R/W	ADxREG0 イネーブル 0:禁止 1:許可
6-5	UVWIS50[1:0]	R/W	相選択(ベクトルエンジン通知用) 下表参照
4-0	AINSP50[4:0]	R/W	AIN 選択 「表 15-2 AD 変換入力端子の選択」を参照

相選択

00	指定なし
01	U 相
10	V 相
11	W 相

15.4.25 ADxTSET03 / ADxTSET47 / ADxTSET811(タイマトリガ用プログラムレジスタ)

本 AD コンバータは、タイマ x のコンペア割り込みをトリガ信号にして、AD 変換を開始する事ができます。

タイマトリガ用プログラムレジスタは 12 個の設定レジスタから構成されます。<ENSTm>を 1 にセットすると ADxTSETm をイネーブルにします。<AINSTm [4:0]>は AIN を選択します。設定レジスタの番号は変換結果レジスタ番号(m=0 ~ 11)にそれぞれ対応しています。 ADxTSET0 は変換結果レジスタ 0 に対応し、ADxTSET11(変換値レジスタ 11)まで全部で 12 個あります。

タイマトリガによる AD 変換が終了すると割り込み(INTADxTMR)を発生します。(x=A,B : AD コンバータユニット)

表 15-3 AD 変換入力端子の選択

<AINST0 [4:0]> ~ <AINST11 [4:0]>	AD コンバータユニット A	AD コンバータユニット B
0_0000	:AINA0	:AINB0
0_0001	:AINA1	:AINB1
0_0010	:AINA2	:AINB2
0_0011	:AINA3	:AINB3
0_0100	:AINA4	:AINB4
0_0101	:AINA5	:AINB5
0_0110	:AINA6	:AINB6
0_0111	:AINA7	:AINB7
0_1000	:AINA8	:AINB8
0_1001	:AINA9	:AINB9
0_1010	:AINA10	:AINB10
0_1011	:AINA11	:Reserved
0_1100 ~ 1_1111	:Reserved	

ADxTSET03：タイマトリガ用プログラムレジスタ 03

	31	30	29	28	27	26	25	24
bit symbol	ENST3	-	-	AINST3				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENST2	-	-	AINST2				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENST1	-	-	AINST1				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENST0	-	-	AINST0				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENST3	R/W	ADxREG3 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINST3[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
23	ENST2	R/W	ADxREG2 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINST2[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
15	ENST1	R/W	ADxREG1 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINST1[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
7	ENST0	R/W	ADxREG0 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINST0[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照

ADxTSET47：タイマトリガ用プログラムレジスタ 47

	31	30	29	28	27	26	25	24
bit symbol	ENST7	-	-	AINST7				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENST6	-	-	AINST6				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENST5	-	-	AINST5				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENST4	-	-	AINST4				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENST7	R/W	ADxREG7 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINST7[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
23	ENST6	R/W	ADxREG6 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINST6[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
15	ENST5	R/W	ADxREG5 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINST5[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
7	ENST4	R/W	ADxREG4 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINST4[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照

ADxTSET811：タイマトリガ用プログラムレジスタ 811

	31	30	29	28	27	26	25	24
bit symbol	ENST11	-	-	AINST11				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENST10	-	-	AINST10				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENST9	-	-	AINST9				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENST8	-	-	AINST8				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENST11	R/W	ADxREG11 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINST11[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
23	ENST10	R/W	ADxREG10 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINST10[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
15	ENST9	R/W	ADxREG9 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINST9[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照
7	ENST8	R/W	ADxREG8 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINST8[4:0]	R/W	AIN 選択 「表 15-3 AD 変換入力端子の選択」を参照

15.4.26 ADxSSET03 / ADxSSET47 / ADxSSET811(ソフトウエアトリガ用プログラムレジスタ)

本 AD 変換はソフトウエアによって AD 変換を開始する事ができます。ソフトウエアトリガ用のプログラムレジスタは 12 個の設定レジスタから構成されます。

<ENSSm>を 1 にセットすると ADxSSETm をイネーブルにします。<AINSSm 4:0>は AIN を選択します。プログラム設定レジスタの番号(m=0 ~ 11)は変換結果レジスタ番号にそれぞれ対応しており、レジスタは全部で 12 個あります。

ソフトウエアトリガによるAD 変換が終了すると割り込み(INTADxSFT)を発生します。（x ＝ A,B : AD コンバータユニット）

表 15-4 AD 変換入力端子の選択

<AINSS0 [4:0]> ~ <AINSS11 [4:0]>	AD コンバータユニット A	AD コンバータユニット B
0_0000	:AINA0	:AINB0
0_0001	:AINA1	:AINB1
0_0010	:AINA2	:AINB2
0_0011	:AINA3	:AINB3
0_0100	:AINA4	:AINB4
0_0101	:AINA5	:AINB5
0_0110	:AINA6	:AINB6
0_0111	:AINA7	:AINB7
0_1000	:AINA8	:AINB8
0_1001	:AINA9	:AINB9
0_1010	:AINA10	:AINB10
0_1011	:AINA11	:Reserved
0_1100 ~ 1_1111	:Reserved	

ADxSSET03：ソフトウエアトリガ用プログラムレジスタ 03

	31	30	29	28	27	26	25	24
bit symbol	ENSS3	-	-	AINSS3				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSS2	-	-	AINSS2				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSS1	-	-	AINSS1				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSS0	-	-	AINSS0				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSS3	R/W	ADxREG3 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINSS3[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
23	ENSS2	R/W	ADxREG2 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINSS2[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
15	ENSS1	R/W	ADxREG1 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINSS1[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
7	ENSS0	R/W	ADxREG0 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINSS0[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照

ADxSSET47：ソフトウエアトリガ用プログラムレジスタ 47

	31	30	29	28	27	26	25	24
bit symbol	ENSS7	-	-	AINSS7				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSS6	-	-	AINSS6				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSS5	-	-	AINSS5				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSS4	-	-	AINSS4				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSS7	R/W	ADxREG7 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINSS7[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
23	ENSS6	R/W	ADxREG6 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINSS6[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
15	ENSS5	R/W	ADxREG5 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINSS5[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
7	ENSS4	R/W	ADxREG4 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINSS4[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照

ADxSSET811：ソフトウエアトリガ用プログラムレジスタ 811

	31	30	29	28	27	26	25	24
bit symbol	ENSS11	-	-	AINSS11				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSS10	-	-	AINSS10				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSS9	-	-	AINSS9				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSS8	-	-	AINSS8				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSS11	R/W	ADxREG11 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINSS11[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
23	ENSS10	R/W	ADxREG10 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINSS10[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
15	ENSS9	R/W	ADxREG9 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINSS9[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照
7	ENSS8	R/W	ADxREG8 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINSS8[4:0]	R/W	AIN 選択 「表 15-4 AD 変換入力端子の選択」を参照

15.4.27 ADxASET03 / ADxASET47 / ADxASET811(常時変換用プログラムレジスタ)

本 AD コンバータは常時変換状態に設定する事が出来ます。常時変換用のプログラムレジスタは 12 個の設定レジスタから構成されます。

設定レジスタの番号は変換結果レジスタ番号(m=0 ~ 11)にそれぞれ対応しています。常時変換用プログラムレジスタの<ENSAm>を 1 にセットすると ADxPSETm をイネーブルにします。<AINSAm [4:0]>は AIN を選択します。

レジスタは全部で 12 個あります。(x = A,B : AD コンバータユニット)

表 15-5 AD 変換入力端子の選択

<AINSA0[4:0]> ~ <AINSA11[4:0]>	AD コンバータ ユニット A	AD コンバータ ユニット B
0_0000	:AINA0	:AINB0
0_0001	:AINA1	:AINB1
0_0010	:AINA2	:AINB2
0_0011	:AINA3	:AINB3
0_0100	:AINA4	:AINB4
0_0101	:AINA5	:AINB5
0_0110	:AINA6	:AINB6
0_0111	:AINA7	:AINB7
0_1000	:AINA8	:AINB8
0_1001	:AINA9	:AINB9
0_1010	:AINA10	:AINB10
0_1011	:AINA11	:Reserved
0_1100 ~ 1_1111	:Reserved	

ADxASET03：常時トリガ用プログラムレジスタ 03

	31	30	29	28	27	26	25	24
bit symbol	ENSA3	-	-	AINSA3				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSA2	-	-	AINSA2				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSA1	-	-	AINSA1				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSA0	-	-	AINSA0				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSA3	R/W	ADxREG3 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINSA3[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
23	ENSA2	R/W	ADxREG2 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINSA2[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
15	ENSA1	R/W	ADxREG1 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINSA1[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
7	ENSA0	R/W	ADxREG0 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINSA0[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照

ADxASET47：常時変換用プログラムレジスタ 47

	31	30	29	28	27	26	25	24
bit symbol	ENSA7	-	-	AINSA7				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSA6	-	-	AINSA6				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSA5	-	-	AINSA5				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSA4	-	-	AINSA4				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSA7	R/W	ADxREG7 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINSA7[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
23	ENSA6	R/W	ADxREG6 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINSA6[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
15	ENSA5	R/W	ADxREG5 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINSA5[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
7	ENSA4	R/W	ADxREG4 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINSA4[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照

ADxASET811：常時変換用プログラムレジスタ 811

	31	30	29	28	27	26	25	24
bit symbol	ENSA11	-	-	AINSA11				
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	ENSA10	-	-	AINSA10				
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	ENSA9	-	-	AINSA9				
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ENSA8	-	-	AINSA8				
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31	ENSA11	R/W	ADxREG11 イネーブル 0:disable 1:enable
30-29	-	R	リードすると"0"が読めます。
28-24	AINSA11[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
23	ENSA10	R/W	ADxREG10 イネーブル 0:disable 1:enable
22-21	-	R	リードすると"0"が読めます。
20-16	AINSA10[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
15	ENSA9	R/W	ADxREG9 イネーブル 0:disable 1:enable
14-13	-	R	リードすると"0"が読めます。
12-8	AINSA9[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照
7	ENSA8	R/W	ADxREG8 イネーブル 0:disable 1:enable
6-5	-	R	リードすると"0"が読めます。
4-0	AINSA8[4:0]	R/W	AIN 選択 「表 15-5 AD 変換入力端子の選択」を参照

15.5 動作説明

15.5.1 アナログ基準電圧

アナログ基準電圧は AD コンバータ ユニット A の VREFHA, VREFLA 端子、AD コンバータ ユニット B の VREFHB、VREFLB 端子にそれぞれ High、Low のレベルを入力します。ADxMOD3<RCUT> に"1"をライトすることによって、VREFHx と VREFLx 間のスイッチオン状態をオフ状態に切り換えることができます。

15.5.2 AD 変換開始

AD コンバータはソフトウエアによる起動（ソフトウエアトリガ）または PMD/タイマからのトリガ信号により任意の AD が変換開始します。

これらの起動要因には優先順位があり、

PMD トリガ 0 > … > PMD トリガ 5 > タイマ > ソフトウエア > 常時

上位の起動要因(トリガ)が発生した時は、実行中のAD 変換を中止して直ちに上位のプログラムを実行します。PMDトリガによるAD変換実行中に上位要因が発生時は、実行中のAD 変換終了後に上位のプログラムに移行します。

起動要因発生から実際の AD 変換実行までには遅れがあります。トリガ要因による AD 変換タイミングと遅延時間を以下に示します。

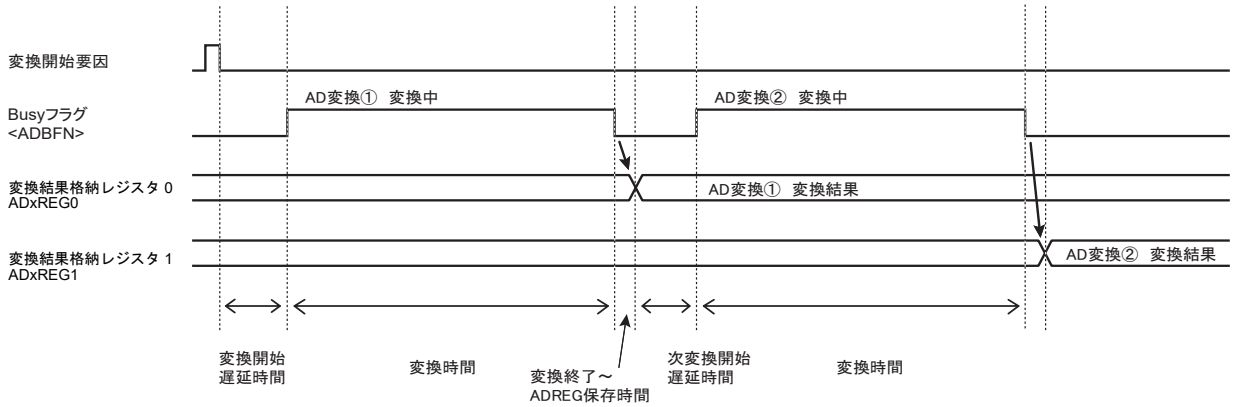


図 15-3 AD 変換開始動作タイミングチャート

表 15-6 AD 変換時間と変換開始遅延時間

変換開始要因	変換開始遅延時間(注 1)		変換時間	変換終了~ADREG 保存時間	次変換開始遅延時間(注 2)	
	Min	Max			Min	Max
PMD トリガ [1/SCLK]	5	8	120	2	3	5
TMRB トリガ[1/SCLK]		16				13
ソフトウエア[1/SCLK]	6	17				
常時変換[1/SCLK]						

注 1)　変換開始要因発生から変換開始までの時間
注 2)　同一変換開始要因で複数の変換を設定する場合の 2 番目以降の開始遅延時間

15.5.3　AD 監視機能

本 AD コンバータには AD 監視機能があり、監視機能有効時に比較条件と一致した場合、割り込みを発生します。

ADxCMPCR0<CMP0EN>または ADxCMPCR1<CMP1EN>を"1"に設定すると AD 監視機能が有効となり、<REGS0>/<REGS1>で指定された変換結果格納レジスタの内容が比較レジスタの値以上または以下になると AD 監視機能割り込みが発生します（<ADBIG0>/<ADBIG1>で以上、以下を指定)

比較動作は監視機能が設定された変換結果格納レジスタに対し、変換結果が格納されるごとにおこなわれ、条件が成立すると割り込みが発生します。

注 1)　AD 監視機能による比較動作では AD 変換結果格納フラグ<ADR0RF> ~ <ADR11RF>はクリアされません。
注 2)　AD 監視機能は通常の読み出し動作と異なりますので、ソフトによる変換結果読み出しなしに連続して AD 変換がおこなわれる場合は、Over Run フラグ<OVR0> ~ <OVR11>が"1"にセットされます。

15.6 AD 変換タイミングチャート

ソフトウエア AD 変換、常時 AD 変換、PMD トリガ／タイマトリガ受け付け時のタイミングチャートを以下に示します。

15.6.1 ソフトウエア AD 変換

ソフトウエア AD 変換では、ADxSSET03, ADxSSET47, ADxSSET811 でプログラムされた AD 変換が終了時に割り込みが発生します(図 15-4)。

ソフトウエア AD 変換中に ADxMOD1<ADEN>="0"とした場合、実行中の AD 変換を終了します。このとき、変換途中の結果はレジスタには格納されません(図 15-5)。

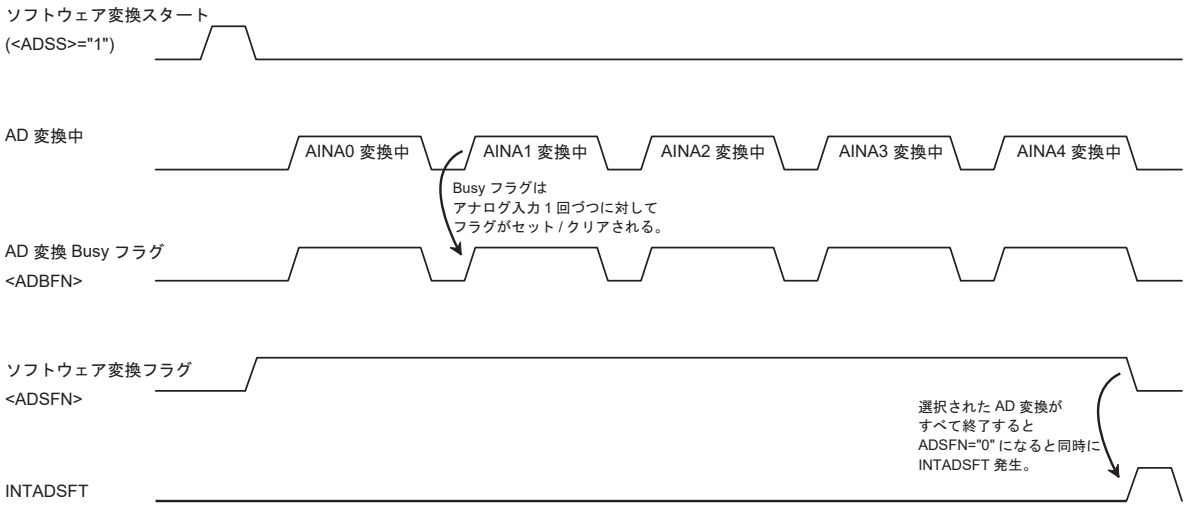


図 15-4 ソフトウエア AD 変換時のタイミングチャート

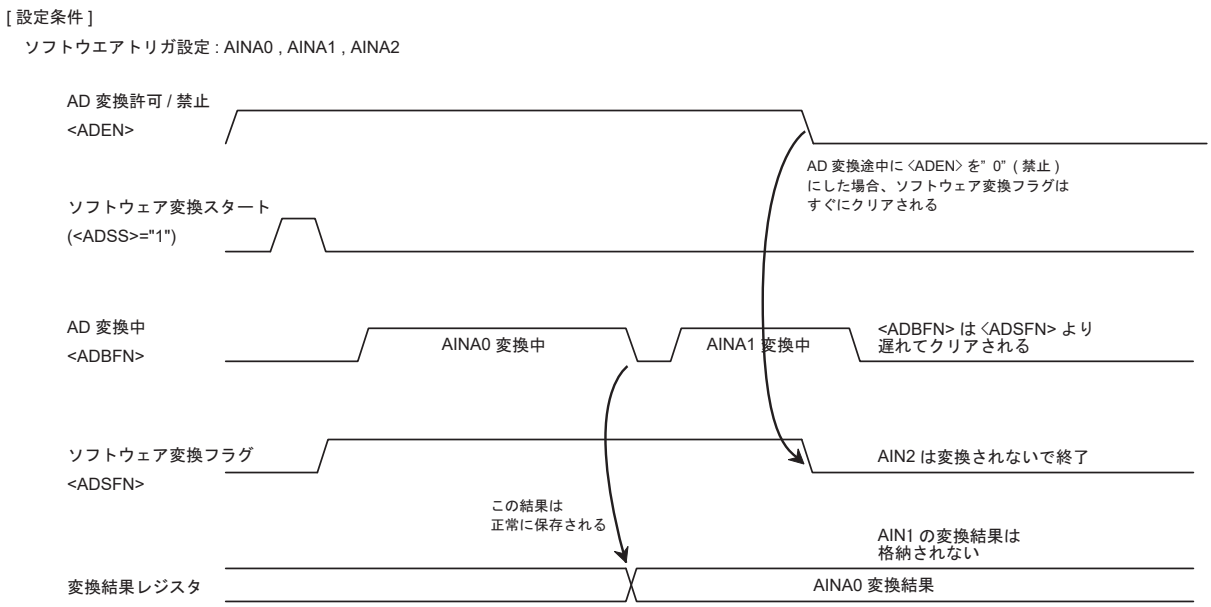


図 15-5 ソフトウエア AD 変換中に<ADEN>="0"書き込み

15.6.2　常時 AD 変換

常時 AD 変換では、変換結果を読み出す前に次の AD 変換が終了した場合、Over Run フラグが"1"にセットされます。このとき、変換結果レジスタは後続のデータによって上書きされます。OverRun フラグは変換結果を読み出すことにより"0"にクリアされます（図 15-6）。

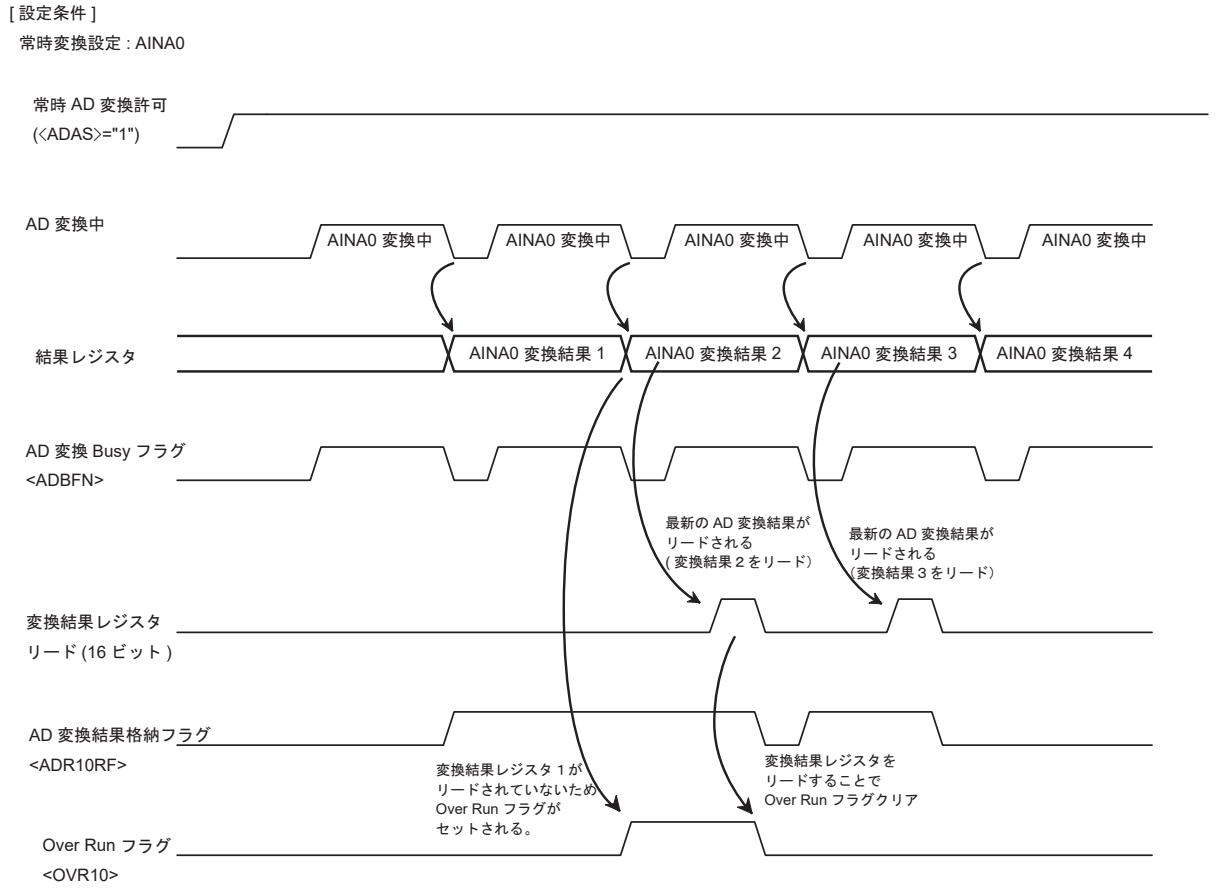


図 15-6 常時 AD 変換時のタイミングチャート

15.6.3 トリガによる AD 変換開始

ソフトウエアAD変換実行中にPMDトリガが発生した場合、実行中のソフトウエアAD変換は直ちに中断してPMDトリガによるAD変換を開始します(図15-7)。PMDトリガによるAD変換が終了後、ソフトウエアAD変換は設定されたプログラムの最初からAD変換を開始します。タイマトリガが発生した場合も同様です(図15-8)。

【設定条件】
ソフトウェアトリガ設定：AINA0 , AINA1 ， AINA2
PMD トリガ設定：AINA4

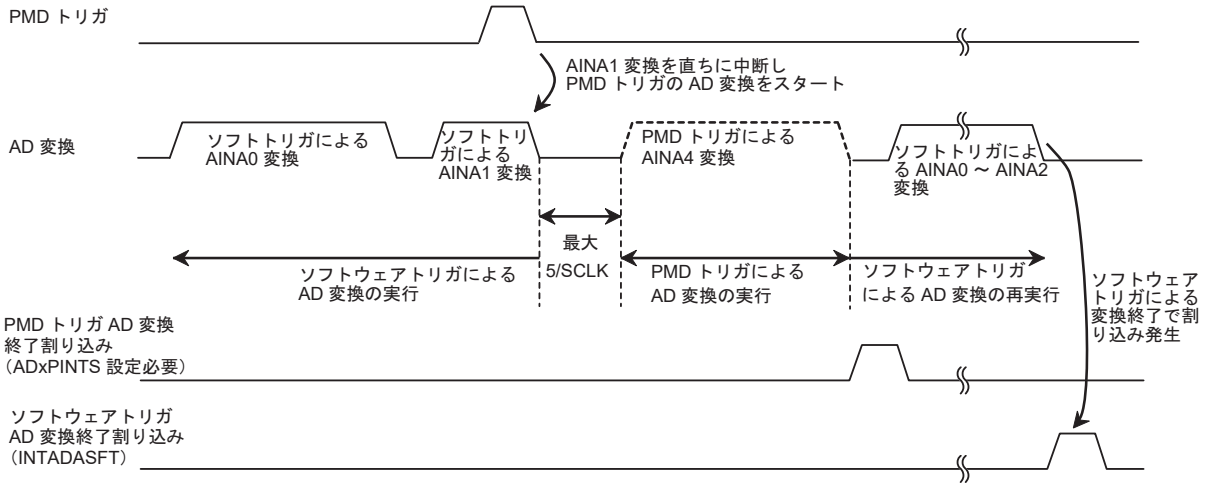


図 15-7 ソフトウエア AD 変換中の PMD トリガ発生

【条件設定】
ソフトウエアトリガ設定: AINB8、AINB9、AINB10
タイマトリガ設定: AINB3

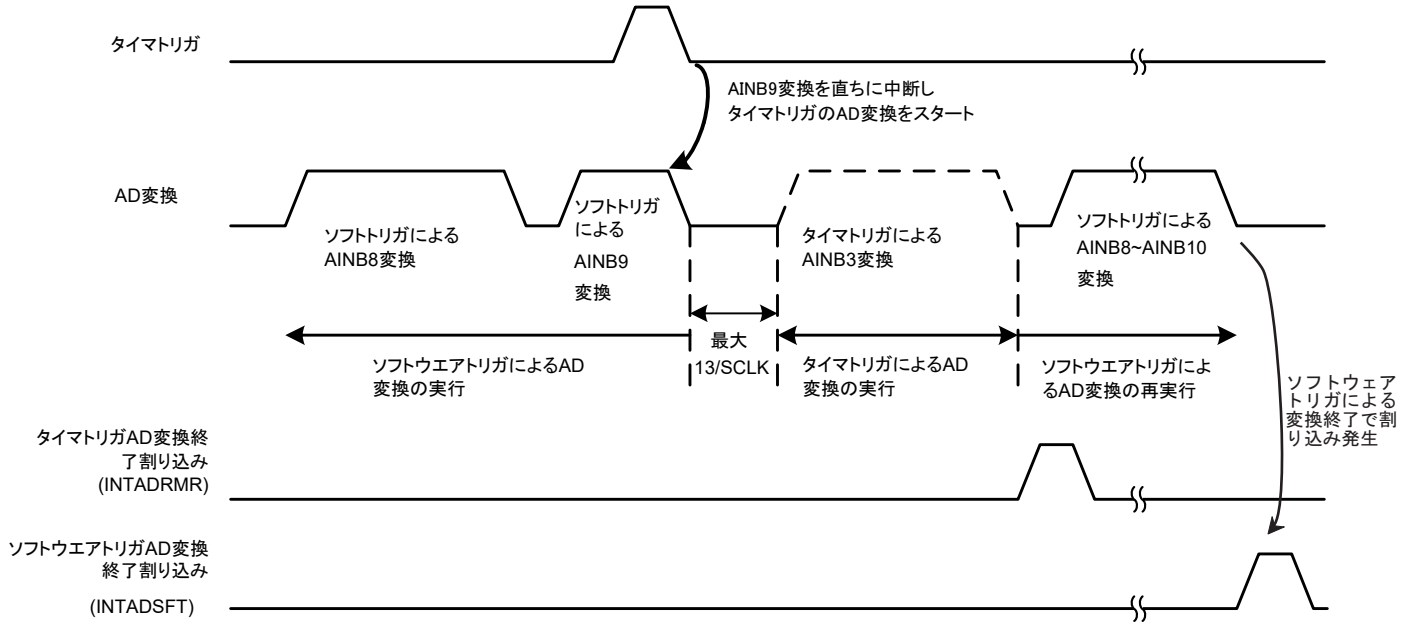


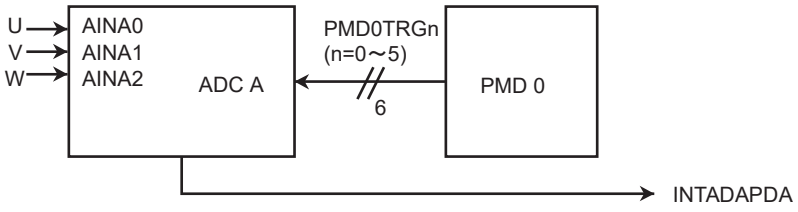
図 15-8 ソフトウエア AD 変換中のタイマ トリガ発生

注）タイマトリガを使用しない場合は、INTTB51を使用しないでください。TB5IM<TBIM1>=1に設定してください。

15.7 使用方法の例

15.7.1 PMD (3 シャント)、AD コンバータ × 1、順次変換方式

PMD0 を 3 シャントで、AD コンバータ（ユニット A）を 1 つ使用した場合の回路図を以下に示します。



この場合の AD コンバータの設定例を以下に示します。

AD コンバータユニット A

Program	0	1	2	3	4	5
reg0	U	V	W	V	W	U
reg1	V	W	U	U	V	W
INT	A	A	A	A	A	A

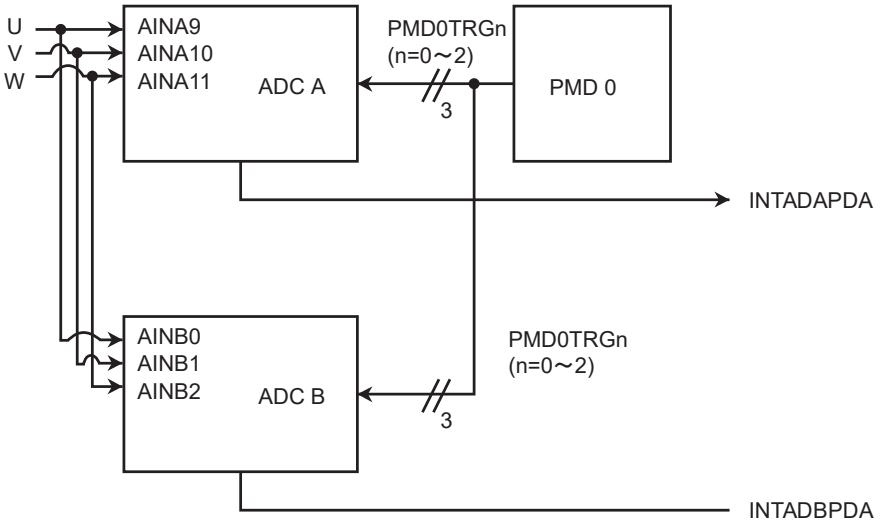
6 本のトリガ入力 PMD0TRG0～5 に対し、ADAPSEL0～5 でそれぞれ 0～5 のプログラム番号を割り付けます。

表中の reg0,1 は ADAPSETn[7:0]および ADAPSETn[15:8]を表します(n：プログラム番号)。表中の「U」、「V」、「W」はモータの位相を表しており、それぞれの位相が得られる AIN 入力を選択します。

トリガが入力されると reg0、reg1 の順に AD 変換が開始され、それぞれの変換結果が変換結果格納レジスタへ保存され、INTADAPDA 割り込みが出力されます。

15.7.2　PMD A(3 シャント)、AD コンバータ × 2、同時変換方式

PMD を 3 シャントで 1 つ、AD コンバータを 2 つ使用した場合の回路図を以下に示します。



この場合の AD コンバータの設定例を以下に示します。

AD コンバータユニット A

Program	0	1	2
reg0	U	V	W
INT	A	A	A

AD コンバータユニット B

Program	0	1	2
reg0	V	W	U
INT	A	A	A

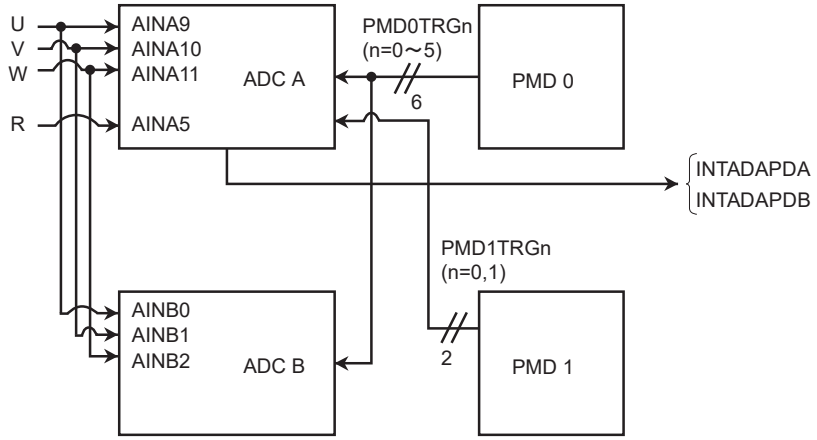
AD コンバータ ユニット A、AD コンバータ ユニット B に入力される 3 本のトリガ入力に、ADAPSEL0～2 および ADBPSEL0～2 でそれぞれ 0～2 のプログラム番号を割り付けます。

表中の reg0 は ADxPSETn[7:0]を表します(x : ADC ユニット , n : プログラム番号)。表中の「U」、「V」、「W」はモータの位相を表しており、それぞれの位相が得られる AIN 入力を選択します。

トリガが入力されると AD コンバータ ユニット A、AD コンバータ ユニット B が同時に AD 変換を開始し、それぞれの変換結果が ADxREG0 レジスタ 0 に保存されます。この時、AD コンバータ ユニット A，AD コンバータ ユニット B からそれぞれ割り込み要求 INTADAPDA と INTADBPDA が出力されます。

15.7.3 PMD 0(3 シャント)、PMD 1(1 シャント)、AD コンバータ × 2、順次変換方式

PMD を 3 シャント、1 シャントで 1 つずつ、AD コンバータを 2 つ使用した場合の回路図を以下に示します。



この場合の AD コンバータの設定例を以下に示します。

AD コンバータユニット A

トリガ	PMD0	PMD0	PMD0	PMD1	PMD1
	0,3	1,4	2,5	6	7
Program	0	1	2	3	4
reg0	U	V	W	-	-
reg1	-	-	-	R	-
reg2	-	-	-	-	R
INT	A	A	A	-	B

AD コンバータユニット B

トリガ	PMD0	PMD0	PMD0
	0,3	1,4	2,5
Program	0	1	2
reg0	V	W	U
INT	-	-	-

AD コンバータ ユニット A の設定は、PMD0、PMD1 から出力される合計 8 本のトリガ信号に対し、PMD0 からの 6 本を 0～2 のプログラム番号に、PMD1 からの 2 本を 3、4 のプログラム番号に割り付けます。

AD コンバータ ユニット B の設定は、PMD0 から出力される合計 6 本のトリガ信号に対し、0～2 のプログラム番号に割り付けます。

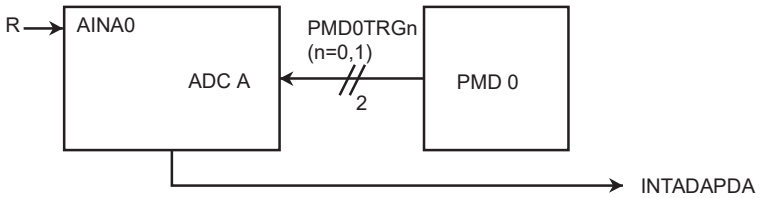
表中のreg0、1、2 は ADxPSETn[7:0]、ADxPSETn[15:8]、ADxPSETn[23:16]を表します。(x:ADC ユニット，n:プログラム番号)表中の「U」、「V」、「W」はモータの位相を表しており、それぞれの位相が得られる AIN 入力を選択します。「R」は抵抗を表しており、接続されている AIN を設定します。

トリガが入力されると AD コンバータ ユニット A、または AD コンバータ ユニット B の AD 変換が開始し、それぞれの変換結果が変換結果格納レジスタ 0 に保存されます。

AD コンバータ ユニット A に関しては PMD 0 からのトリガは INTADAPDA 割り込みが、PMD 1 からのトリガは INTADAPDB 割り込みが出力されます。AD コンバータ ユニット B はここでは割り込みを出力しない設定にしています。

15.7.4　PMD (1 シャント)、AD コンバータ × 1、順次変換方式

PMD を 1 シャントで 1 つ、AD コンバータ を 1 つ使用した場合の回路図を以下に示します。



この場合の AD コンバータの設定例を以下に示します。

AD コンバータユニット A

トリガ	PMD0	PMD0
	0	1
Program	0	1
reg0	R	-
reg1	-	R
INT	-	A

PMD0 から出力される 2 本のトリガ信号に対し、それぞれプログラム番号を割り付けています。

表中の reg0,1 は ADAPSETn[7:0]、ADAPSETn[15:8]を表します(n：プログラム番号)。表中の「R」は抵抗を表しており、接続されている AIN を設定します。

トリガが入力されると AD コンバータ ユニット A の AD 変換が開始し、変換結果が変換結果格納レジスタ 0 と 1 に保存されます。プログラム 0、1 の順で変換が実行され、終了すると INTADAPDA 割り込みを発生します。

15.8 AD コンバータ使用時の注意

AD 変換精度への影響を避けるために AD 動作(変換)中は出力ポートとして使用しているポート H/J/K の出力データを書き換えないようにしてください。

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。

また、AD 変換中に AD 入力が兼用となっている端子への入力及び出力の変化、出力ポートに設定している他の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の対策をして下さい。

第 16 章　モータ制御回路(PMD : Programmable Motor Driver)

本製品の PMD はベクトルエンジン(A-VE)やアナログ/デジタルコンバータ(ADC)と連携動作してベクトル制御などの 3 相モータ制御を実現します。パルス幅変調回路、通電制御および同期トリガ生成回路は VE からの指令で動作可能で、同期トリガ生成回路は ADC に変換開始指令ができます。

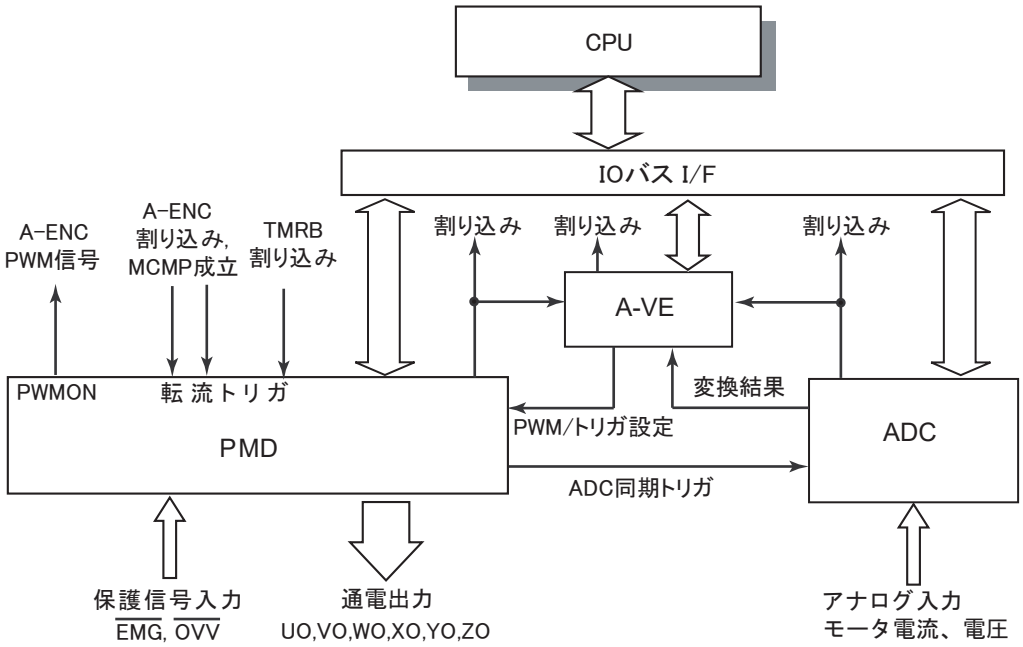


図 16-1 モータ制御関連機能のブロック図

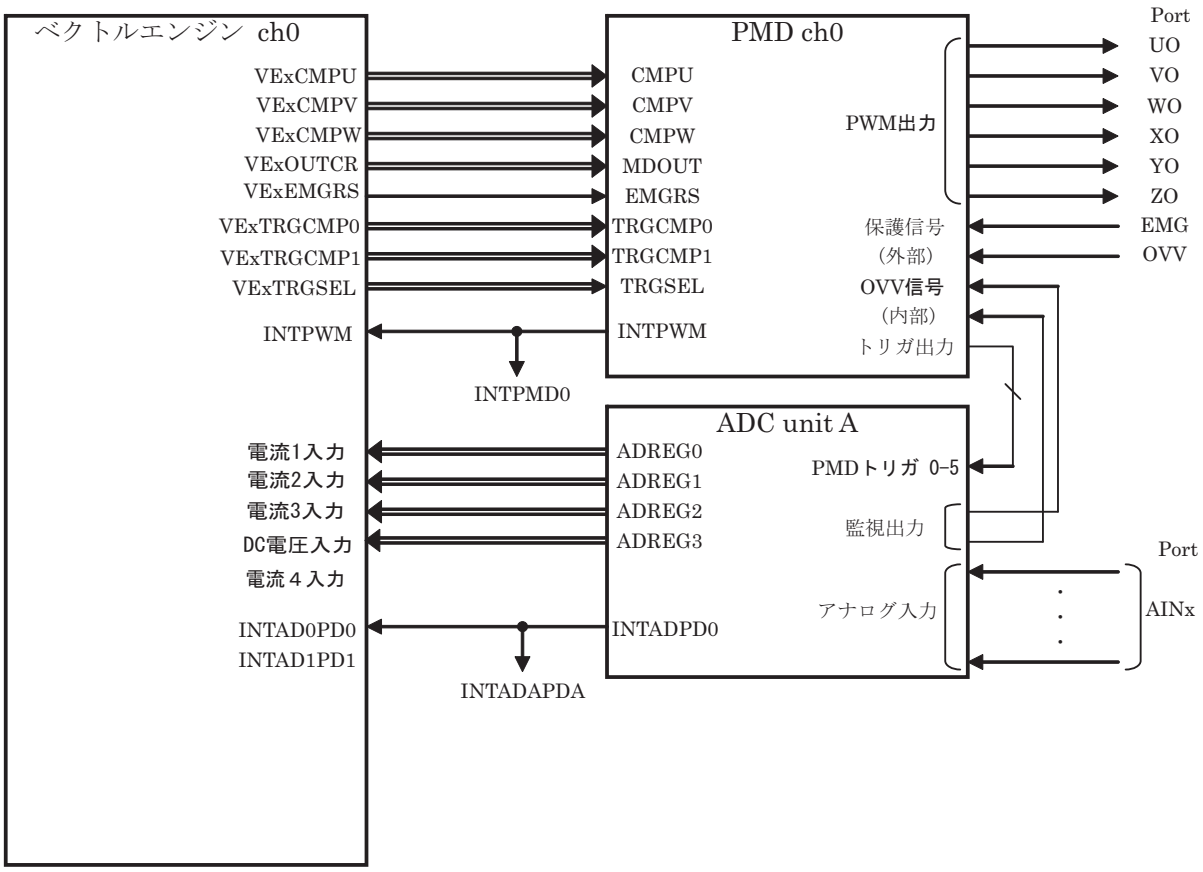


図 16-2 モータ制御回路とベクトルエンジンおよび A/D 変換器の関連図

16.1 PMD 回路構成

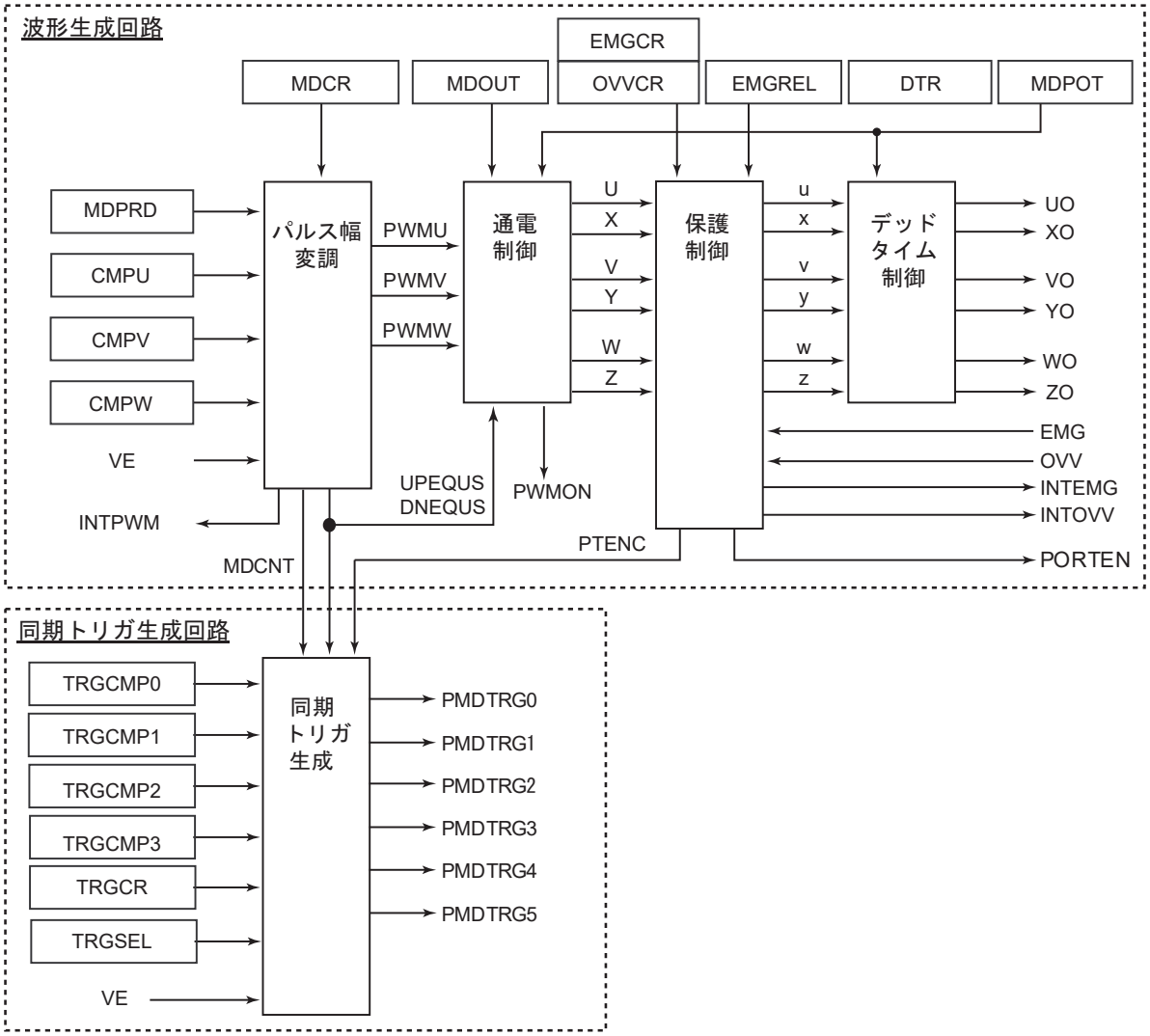


図 16-3 PMD 回路概略図

PMD(プログラマブルモータドライバ)回路は波形生成回路と同期トリガ生成回路の 2 ブロックから成り、波形生成回路はパルス幅変調回路、通電制御回路、保護制御回路、デッドタイム制御回路で構成されています。

- パルス幅変調回路は、PWM キャリアが共通で 3 相の独立した PWM 波形を生成します。
- 通電制御回路は U、V、W 相の各上下相の出力パターンを決定します。
- 保護回路では EMG 入力、OVV 入力による緊急出力停止を行ないます。
- デッドタイム制御回路では上下相の切り替え時の短絡を防止します。
- 同期トリガ生成回路では ADC への同期トリガ信号を生成します。

16.2　PMD レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		Address(Base+)
PMD イネーブルレジスタ	PMDxMDEN	0x0000
ポート出力モードレジスタ	PMDxPORTMD	0x0004
PMD コントロールレジスタ	PMDxMDCR	0x0008
PWM カウンタステータスレジスタ	PMDxCNTSTA	0x000C
PWM カウンタレジスタ	PMDxMDCNT	0x0010
PWM 周期レジスタ	PMDxMDPRD	0x0014
PWM コンペア U レジスタ	PMDxCMPU	0x0018
PWM コンペア V レジスタ	PMDxCMPV	0x001C
PWM コンペア W レジスタ	PMDxCMPW	0x0020
モード選択レジスタ	PMDxMODESEL	0x0024
PMD 通電制御レジスタ	PMDxMDOUT	0x0028
PMD 出力設定レジスタ	PMDxMDPOT	0x002C
EMG 解除レジスタ	PMDxEMGREL	0x0030
EMG コントロールレジスタ	PMDxEMGCR	0x0034
EMG ステータスレジスタ	PMDxEMGSTA	0x0038
OVV コントロールレジスタ	PMDxOVVCR	0x003C
OVV ステータスレジスタ	PMDxOVVSTA	0x0040
デッドタイムレジスタ	PMDxDTR	0x0044
トリガコンペア 0 レジスタ	PMDxTRGCMP0	0x0048
トリガコンペア 1 レジスタ	PMDxTRGCMP1	0x004C
トリガコンペア 2 レジスタ	PMDxTRGCMP2	0x0050
トリガコンペア 3 レジスタ	PMDxTRGCMP3	0x0054
トリガコントロールレジスタ	PMDxTRGCR	0x0058
トリガ出力モード設定レジスタ	PMDxTRGMD	0x005C
トリガ出力選択レジスタ	PMDxTRGSEL	0x0060
トリガ更新タイミング設定レジスタ	PMDxTRGSYNCR	0x0064

16.2.1 PMDxMDEN(PMD 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	-	-	-	-	-	-	-	PWMEN
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	PWMEN	R/W	波形合成機能の許可/禁止を制御。 0: 禁止 1: 許可 注)ポートを機能出力(PWM 出力)として設定している場合、<PWMEN> = "0"でポートは出力禁止(ハイインピーダンス)になります。 注)出力ポート極性等、<PWMEN>以外の初期設定を行った後に、<PWMEN> = "1"に設定して下さい。

16.2.2　PMDxPORTMD(ポート出力モードレジスタ)

	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	-	-	-	-	-	-	PORTMD	
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
1-0	PORTMD[1:0]	R/W	ツールブレーク時のポート制御の設定 00 : 上相 High-z / 下相 High-z 01 : 上相 High-z / 下相 PMD 出力 10 : 上相 PMD 出力 / 下相 High-z 11 : 上相 PMD 出力 / 下相 PMD 出力 ポートを機能出力(PMW 出力)で使用中にツールブレークが発生した場合の上相(UO/VO/WO)および下相(XO/YO/ZO)のポート出力を設定します。 "High-z"選択時にツールブレークが発生した場合、ポートを出力禁止(ハイインピーダンス)にします。それ以外の場合は PMD 出力に従います。

注 1)　<PWMEN> = "0" 時は PORTMD の設定に関わらず出力禁止(ハイインピーダンス)になります。
注 2)　ポート出力制御は PMDxEMGCR<EMGMD[1:0]> の設定により EMG 入力時にも行なわれます。

16.2.3 PMDxMODESEL (モード選択レジスタ)

	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	DCMPEN	-	-	-	MDSEL3	MDSEL2	MDSEL1	MDSEL0
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	DCMPEN	R/W	VE レジスタと PMD レジスタの自動切り替え許可 0: 2 レジスタ切り替え禁止 (<MDSEL0>設定のレジスタのみ使用) 1: 2 レジスタ切り替え許可 (PWM カウンタのアップ期間とダウン期間で切り替え) (注) <MDSEL0> = "1"の場合に有効 (注) 三角波キャリア選択(PMDxMDCR<PWMMD> = "1")時に有効
6-4	−	R	リードすると"0"が読めます。
3	MDSEL3	R/W	モード選択 3 0: バスモード(PMD レジスタ使用: PMDxTRGSEL) 1: VE モード(VE レジスタ使用: VExTRGSEL)
2	MDSEL2	R/W	モード選択 2 0: バスモード(PMD レジスタ使用: PMDxTRGCMP0 および PMDxTRGCMP1) 1: VE モード(VE レジスタ使用: VExTRGCMP0 および VExTRGCMP1)
1	MDSEL1	R/W	モード選択 1 0: バスモード(PMD レジスタ使用: PMDxMDOUT) 1: VE モード(VE レジスタ使用: VExOUTCR)
0	MDSEL0	R/W	モード選択レジスタ 0 0: バスモード(PMD レジスタ使用: PMDxCMPU,PMDxCMPV および PMDxCMPW) 1: VE モード(VE レジスタ使用: VExCMPU,VExCMPV,VExCMPW レジスタの選択および VExEMGRS レジスタ許可設定)

16.2.4　パルス幅変調回路

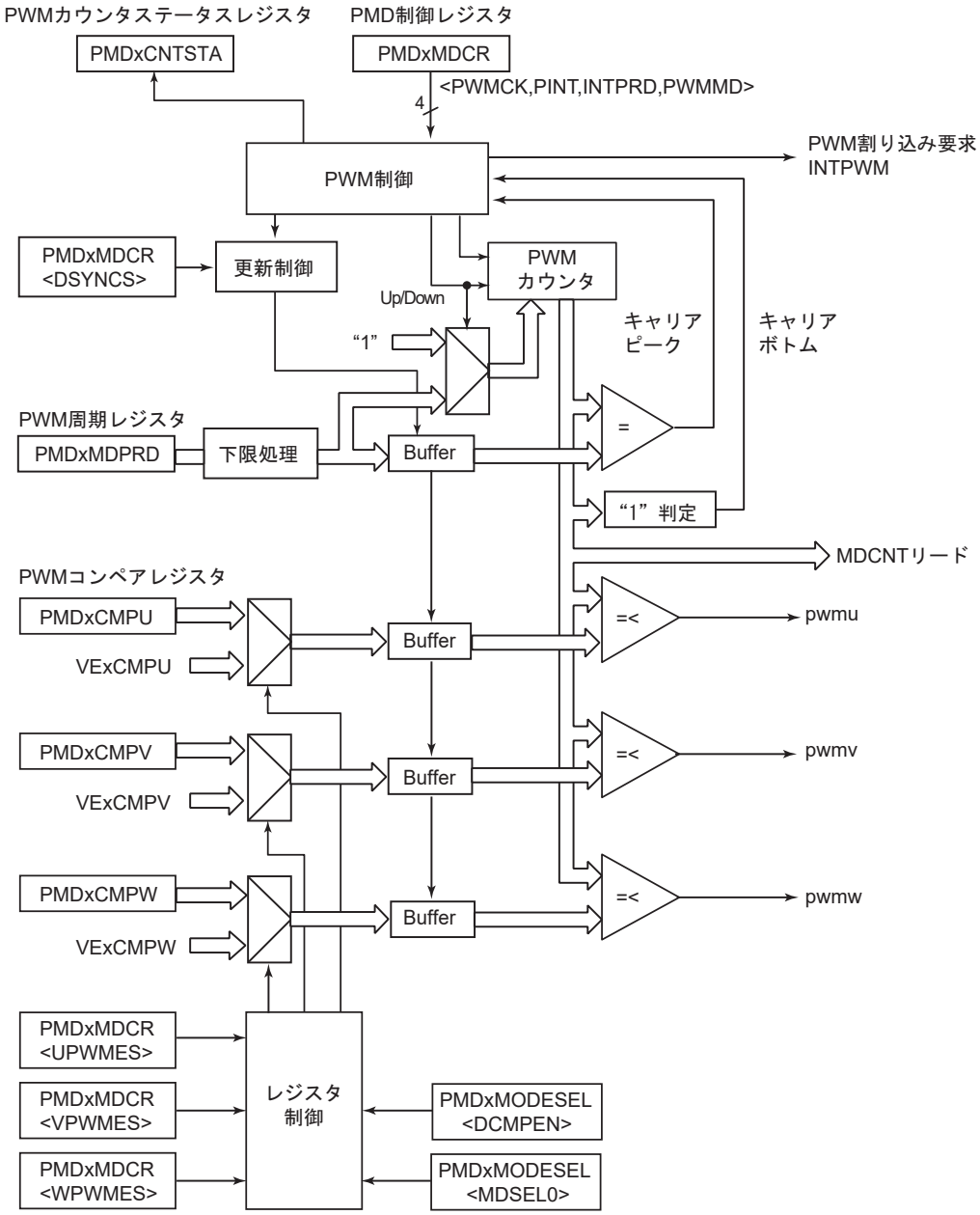


図 16-4 パルス幅変調回路概略図

パルス幅変調回路は、16bit のアップ/ダウンカウンタである PWM カウンタを持ち、1/fsys (8.33[ns] @120[MHz]) の分解能で PWM キャリアを生成します。PWM 周期延長モード(<PWMCK> = "1")にすることにより、PWM カウンタは 4/fsys (33.3[ns]@120[MHz])の分解能で PWM キャリアを生成します。

PWM キャリアの波形モードは PWM モード0 としてエッジ PWM(のこぎり波変調)、モード1 としてセンタ PWM(三角波変調)を選択可能です(「図 16-5 PWM 波形」参照)。また、三角波変調ではセンタPWM の他に立ち下がりエッジ固定と立ち上がりエッジ固定のPWM を相別に選択することができます。(「図 16-6 三角波キャリア PWM のエッジ固定波形」参照)。

1. PWM 周期設定

PMDxMDPRD により PWM 周期を決定します。PMDxMDPRD はダブルバッファ構成であり、後段バッファは PWM 周期で更新されます。PWM 半周期毎の更新も選択できます(「表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御」参照)。

$$\text{のこぎり波変調PWM：PMDxMDPRDレジスタ設定} = \frac{\text{システムクロックfsys[Hz]}}{\text{PWM周波数[Hz]}}$$

$$\text{三角波変調ＰWM：PMDxMDPRDレジスタ設定値} = \frac{\text{システムクロックfsys[Hz]}}{\text{PWM周波数[Hz]} \times 2}$$

2. コンペア機能

3 相の PWM コンペアレジスタ(PMDxCMPU/V/W)の値と、PWM カウンタ PMDxMDCNT <MDCNT[15:0]>により生成される PWM キャリアをコンパレータで大小比較して所望のデューティの PWM 波形を生成します。

各相の PWM コンペアレジスタはダブルバッファ構成となります。PWM コンペアレジスタの値は PWM 周期に同期して後段バッファにロードされます。PWM 半周期での更新(半周期毎ロード)も選択できます(「表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御」参照)。

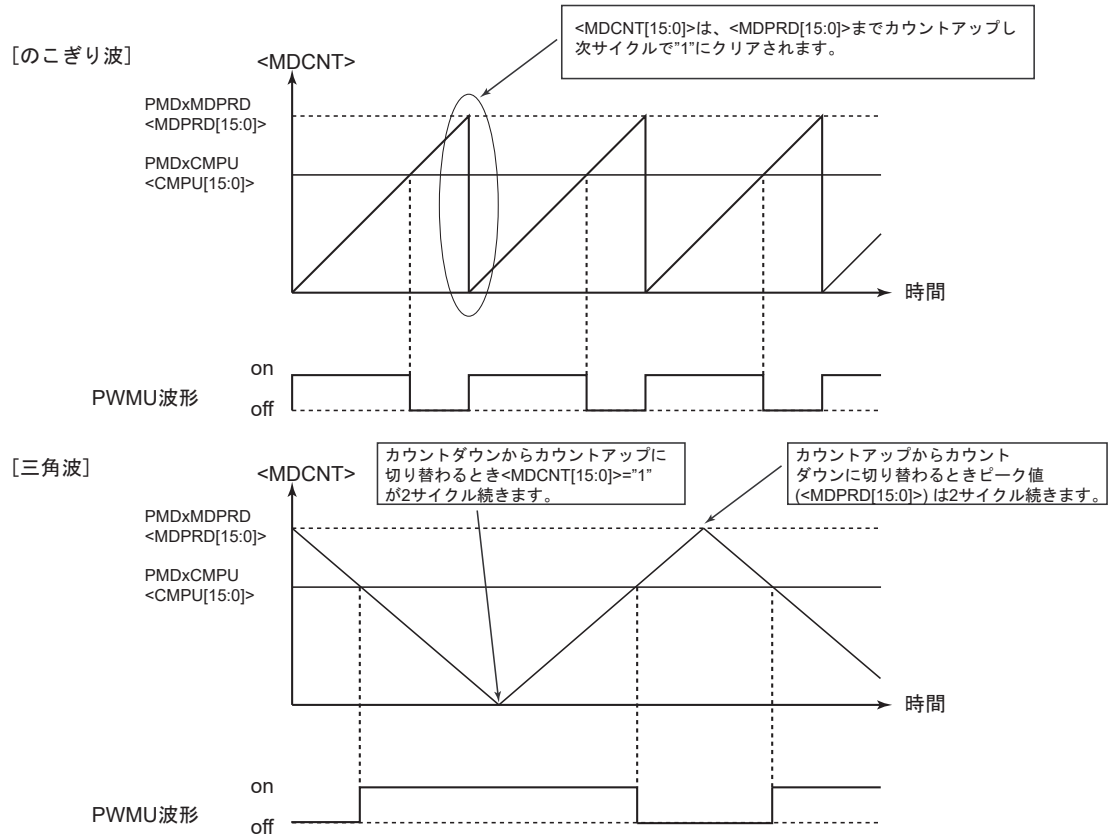


図 16-5 PWM 波形

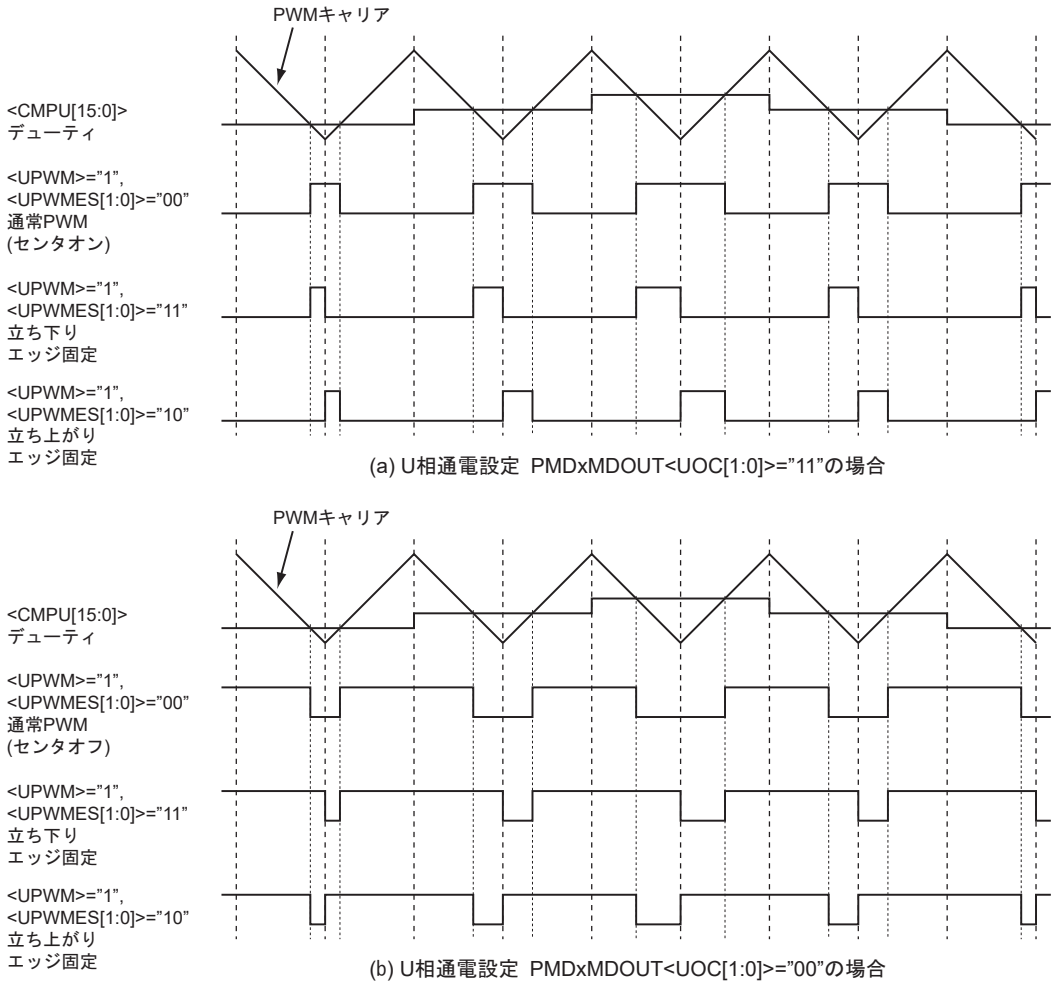


図 16-6 三角波キャリア PWM のエッジ固定波形

3. 波形モード

2 種類の 3 相 PWM の生成方法を選択できます。

1. 3 相独立 Duty モード: 3 相の PWM コンペアレジスタにそれぞれ独立した値を設定して、3 相の独立した PWM 波形を生成します。これは、正弦波等の任意の駆動波形生成に使用します。

2. 3 相共通 Duty モード: U 相の PWM コンペアレジスタだけに設定し、U 相の設定値で 3 相同一の PWM 波形を生成します。これは、DC モータの矩形波駆動に使用します。

4. 割り込み処理

パルス幅変調回路では PWM 波形に同期して PWM 割り込み要求を発生します。割り込み要求の発生タイミングは PWM キャリアピークと PWM キャリアボトムを選択できます。

PWM 割り込みの頻度は PWM 周期半周期に 1 回、1 周期に 1 回、2 周期に 1 回、4 周期に 1 回を選択することができます。

16.2.4.1 PMDxMDCR (PMD 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	WPWMES		VPWMES		UPWMES		DSYNCS	
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	DTCREN	PWMCK	SYNTMD	DTYMD	PINT	INTPRD		PWMMD
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。
15-14	WPWMES[1:0]	R/W	W 相 PWM エッジ設定 00:エッジ固定なし(センタ PWM) 01: Reserved 10: PWM 立ち上がりエッジ固定(キャリアボトムに固定) 11: PWM 立ち下がりエッジ固定(キャリアボトムに固定) 注) 三角波キャリア選択(<PWMMD> = "1")時に有効
13-12	VPWMES[1:0]	R/W	V 相 PWM エッジ設定 00:エッジ固定なし(センタ PWM) 01: Reserved 10: PWM 立ち上がりエッジ固定(キャリアボトムに固定) 11: PWM 立ち下がりエッジ固定(キャリアボトムに固定) 注) 三角波キャリア選択(<PWMMD> = "1")時に有効
11-10	UPWMES[1:0]	R/W	U 相 PWM エッジ設定 00:エッジ固定なし(センタ PWM) 01: Reserved 10: PWM 立ち上がりエッジ固定(キャリアボトムに固定) 11: PWM 立ち下がりエッジ固定(キャリアボトムに固定) 注) 三角波キャリア選択(<PWMMD> = "1")時に有効
9-8	DSYNCS[1:0]	R/W	Duty コンペアレジスタと PWM 周期レジスタのダブルバッファ更新タイミング 00: 割り込み周期設定(INTPRD)によります。(表 16-1 参照) 0.5 周期割り込み要求選択(<INTPRD> = "00")時はキャリアピークとボトムで更新、それ以外ではキャリアピーク更新 01: PWM キャリアボトムで更新 10: PWM キャリアピークで更新 11: PWM キャリアのピークとボトムで更新 注 1) ノコギリ波キャリア選択(<PWMMD> = "0")時は設定に関わらずキャリアピーク更新になります。 注 2) PMDxMDEN<PWMEN> = "0" 時は設定に関わらず非同期更新になります。
7	DTCREN	R/W	デッドタイム補正許可 0: 禁止 1: 許可
6	PWMCK	R/W	PWM 周期延長モードを指定 0: 通常周期 1: 4 倍周期 PWM カウンタのカウント周期を設定します。 通常周期設定:ノコギリ波 1/fsys (8.33[ns]@120[MHz]) / 三角波 2/fsys (16.7[ns]@120[MHz]) 4 倍周期設定:ノコギリ波 4/fsys (33.3[ns]@120[MHz]) / 三角波 8/fsys (66.7[ns]@120[MHz])
5	SYNTMD	R/W	ポート出力モード設定 <nOC>,<nPWM>,<POLH>,<POLL>,<SYNTMD>との組み合わせでポート出力を制御します。 (表 16-4)

Bit	Bit Symbol	Type	機能
4	DTYMD	R/W	DUTY モード選択 0: 3 相共通 1: 3 相独立 デューティの設定を PMDxCMPU/V/W の 3 相独立で行うか、PMDxCMPU レジスタの設定を 3 相共通で用いるかの選択を行います。
3	PINT	R/W	PWM 割り込み要求タイミング選択 0: PWM キャリアボトム(PMDxMDCNT<MDCNT[15:0]> = 0x0001)で割り込み要求発生 1: PWM キャリアピーク(PMDxMDCNT<MDCNT[15:0]> = PMDxMDPRD<MDPRD[15:0]>)で割り込み要求発生 注 1) PWM キャリアがノコギリ波(<PWMMD> = "0")の場合はキャリアピークになります。 注 2) 割り込み周期が 0.5 周期(<INTPRD> = "00")の場合はキャリアピークとキャリアボトムの両方になります。
2-1	INTPRD[1:0]	R/W	PWM 割り込み要求周期選択 00: PWM 0.5 周期毎に割り込み要求 注 1) PWM キャリアが三角波(<PWMMD> = "1")の場合のみ設定可能 注 2) コンペアレジスタ(PMDxCMPU/V/W)および周期レジスタ PMDxMDPRD のダブルバッファは PWM キャリアのピークとボトムで更新されます。 01: PWM 1 周期毎に割り込み要求 10: PWM 2 周期毎に割り込み要求 11: PWM 4 周期毎に割り込み要求 PWM 割り込み要求の発生する頻度を PWM 周期の 0.5 周期/1 周期/2 周期/4 周期に 1 回から選択します。
0	PWMMD	R/W	PWM キャリア波形選択 0: PWM モード 0(エッジ PWM、ノコギリ波) 1: PWM モード 1(センタ PWM、三角波)

表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御

設定		更新タイミング
<DSYNCS[1:0]>	<INTPRD[1:0]>	
00	1x	PWM キャリアピークで更新
	x1	PWM キャリアピークで更新
	00	PWM キャリアのピークとボトムで更新
01	xx	PWM キャリアボトムで更新
10	xx	PWM キャリアピークで更新
11	xx	PWM キャリアのピークとボトムで更新

x : Don’t care

表 16-2 PMDxCMPU/V/W および VExCMPU/V/W レジスタ切り替え制御

共通設定		相別設定	レジスタ選択信号
<DSYNCS[1:0]>	<INTPRD[1:0]>	<UPWMES[1]> <VPWMES[1]> <WPWMES[1]>	
01	xx	x	VE レジスタ
10	xx	x	VE レジスタ
11	xx	0	アップカウント中:PMD レジスタ ダウンカウント中:VE レジスタ
		1	VE レジスタ
00	00	0	アップカウント中:PMD レジスタ ダウンカウント中:VE レジスタ
		1	VE レジスタ

注) <MDSEL0> = "1", <DCMEN> = "1", <PWMMD> = "1" の場合に有効

x : Don’t care

16.2.4.2　PMDxCNTSTA (PWM カウンタ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	-	-	-	-	-	-	-	UPDWN
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	UPDWN	R	PWM カウンタフラグ 0: アップカウント中 1: ダウンカウント中 PWM カウンタがアップカウント中かダウンカウント中かを示します。 注) ノコギリ波キャリア(PMDxMDCR<PWMMD> = "0")は、常に"0"が読み出されます。

16.2.4.3 PMDxMDCNT(PWM カウンタレジスタ)

	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	MDCNT							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	MDCNT							
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	MDCNT[15:0]	R	PWM カウンタ PWM キャリア波形となるアップダウンカウンタを読み出せます。 カウンタ値分解能 1/fsys (8.33[ns]@120[MHz]) 注 1) 4 倍周期モード(PMDxMDCR<PWMCK> = "1")の場合、分解能は 4/fsys (33.3[ns]@120[MHz]) になります。 注 2) PMD 禁止(PMDxMDEN<PWMEN> = "0")の PWM カウンタ値は PMDxMDCR<PWMMD> 設定(キャリア波形選択)によって、以下のようになります。 <PWMMD> = "0"の場合: 0x0001 <PWMMD> = "1"の場合: PMDxMDPRD<MDPRD[15:0]>レジスタ 値

16.2.4.4　PMDxMDPRD(PWM 周期レジスタ)

	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	MDPRD							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	MDPRD							
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	MDPRD[15:0]	R/W	PWM 周期設定。 <MDPRD[15:0]> ≥ 0x010 PWM 周期は PWM 周期延長モード PMDxMDCR<PWMCK>と PWM キャリア波形<PWMMD>の組み合わせで以下の計算式で表されます。 <PWMCK> = "0"の場合、<PWMMD> = "0" : <MDPRD> × 1/fsys <PWMMD> = "1" : <MDPRD> × 2/fsys <PWMCK> = "1"の場合、 <PWMMD> = "0" : <MDPRD> × 4/fsys <PWMMD> = "1" : <MDPRD> × 8/fsys 注) <MDPRD[15:0]>に 0x010 未満の値を設定した場合、自動的に<MDPRD[15:0]> = 0x010 として動作します(レジスタには設定した値が入ります)。

注 1)　このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 2)　PMDxMDPRD レジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 3)　後段バッファの更新タイミングは「表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御」を参照してください。

注 4)　リード時は前段バッファの値(バスから設定したデータ)をリードします。

16.2.4.5 PMDxCMPU (U 相用 PWM コンペアレジスタ)

	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	CMPU							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMPU							
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	CMPU[15:0]	R/W	U 相 PWM パルス幅の設定。 0x0000 ~ 0xFFFF 注) <CMPU> > <MDPRD>の場合、デューティ 100%になります。 U 相 PWM のパルス幅を決定する比較レジスタで、ダブルバッファ構成となっています。後段のバッファと PWM カウンタとを大小比較しパルス幅を決定します。 パルス幅は PWM 周期延長モード PMDxMDCR<PWMCK>と PWM キャリア波形<PWMMD>の組み合わせで以下の計算式で表されます。 <PWMCK> = "0"の場合、<PWMMD> = "0" : <CMPU> × 1/fsys <PWMMD> = "1" : <CMPU> × 2/fsys <PWMCK> = "1"の場合、 <PWMMD> = "0" : <CMPU> × 4/fsys <PWMMD> = "1" : <CMPU> × 8/fsys

注 1) バスから更新したコンペアレジスタをダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL0>に"0" を書き込んでバスモード(デフォルト)にしてください。

注 2) このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 3) PMDxCMPU レジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 4) 後段バッファの更新タイミングは「表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御」を参照してください。

注 5) リード時は前段バッファの値(バスから設定したデータ)をリードします。

16.2.4.6　PMDxCMPV (V 相用 PWM コンペアレジスタ)

	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	CMPV							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMPV							
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	CMPV[15:0]	R/W	V 相 PWM パルス幅の設定。 0x0000 ~ 0xFFFF 注) <CMPV> > <MDPRD>の場合、デューティ 100%になります。 V 相 PWM のパルス幅を決定する比較レジスタで、ダブルバッファ構成となっています。後段のバッファと PWM カウンタとを大小比較しパルス幅を決定します。 パルス幅は PWM 周期延長モード PMDxMDCR<PWMCK>と PWM キャリア波形<PWMMD>の組み合わせで以下の計算式で表されます。 <PWMCK> = "0"の場合、<PWMMD> = "0" : <CMPV> × 1/fsys <PWMMD> = "1" : <CMPV> × 2/fsys <PWMCK> = "1"の場合、 <PWMMD> = "0" : <CMPV> × 4/fsys <PWMMD> = "1" : <CMPV> × 8/fsys

注 1)　バスから更新したコンペアレジスタをダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL0>に"0" を書き込んでバスモード(デフォルト)にしてください。

注 2)　このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 3)　PMDxCMPV レジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 4)　後段バッファの更新タイミングは「表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御」を参照してください。

注 5)　リード時は前段バッファの値(バスから設定したデータ)をリードします。

16.2.4.7 PMDxCMPW (W 相用 PWM コンペアレジスタ)

	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	CMPW							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMPW							
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	CMPW[15:0]	R/W	W 相 PWM パルス幅の設定。 0x0000 ~ 0xFFFF 注) <CMPW> > <MDPRD>の場合、デューティ 100%になります。 W 相 PWM のパルス幅を決定する比較レジスタで、ダブルバッファ構成となっています。後段のバッファと PWM カウンタとを大小比較しパルス幅を決定します。 パルス幅は PWM 周期延長モード PMDxMDCR<PWMCK>と PWM キャリア波形<PWMMD>の組み合わせで以下の計算式で表されます。 <PWMCK> = "0"の場合、<PWMMD> = "0" : <CMPW> × 1/fsys <PWMMD> = "1" : <CMPW> × 2/fsys <PWMCK> = "1"の場合、 <PWMMD> = "0" : <CMPW> × 4/fsys <PWMMD> = "1" : <CMPW> × 8/fsys

注 1) バスから更新したコンペアレジスタをダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL0>に"0" を書き込んでバスモード(デフォルト)にしてください。

注 2) このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 3) PMDxCMPW レジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 4) 後段バッファの更新タイミングは「表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御」を参照してください。

注 5) リード時は前段バッファの値(バスから設定したデータ)をリードします。

16.2.5　通電制御回路

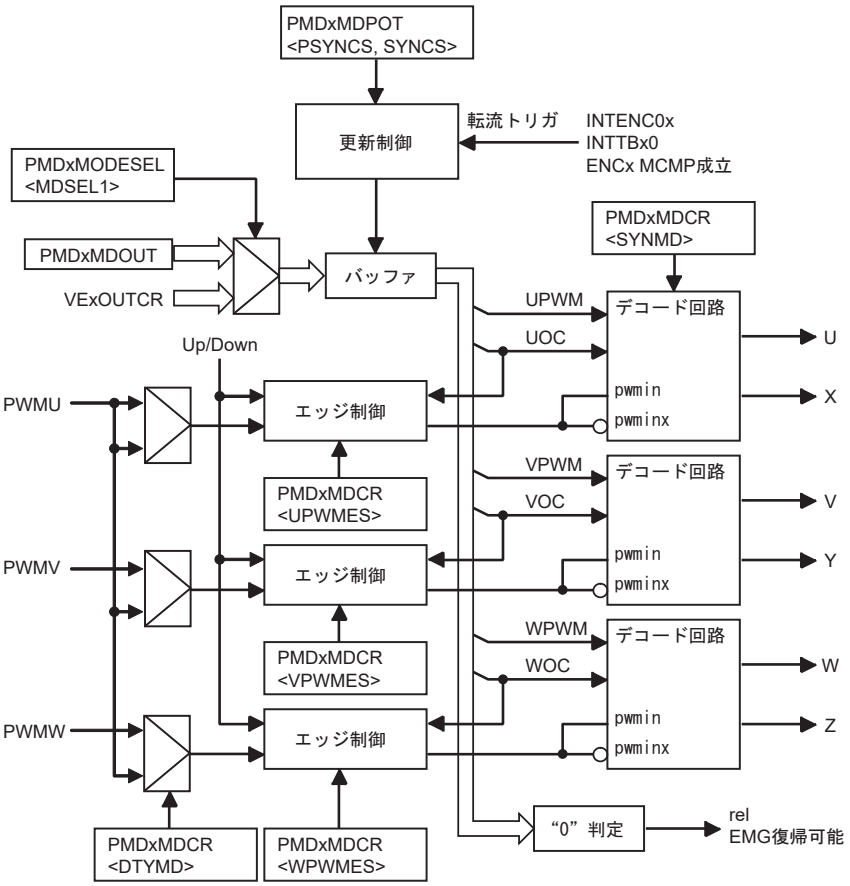


図 16-7 通電制御回路概略図

通電制御レジスタ PMDxMDOUT(VExOUTCR)と出力設定レジスタ PMDxMDPOT の設定により、出力ポートの制御を行います。PMDxMDOUT(VExOUTCR)レジスタはダブルバッファ構成であり、更新タイミングは PWM への同期更新と非同期更新を選択できます。また、トリガ入力に同期した更新を選択することも可能です。（更新タイミングは「表 16-3 PMDxMDOUT(VExOUTCR)バッファの更新タイミング」を参照してください。）

6 本のポートの出力設定は、上相出力(UO,VO,WO)と下相出力(XO,YO,ZO)のそれぞれにローアクティブ/ハイアクティブ設定を PMDxMDPOT の<POLH>,<POLL> により行うことができます。さらに、U,V,W それぞれに、PWM 出力と High/Low 出力との選択を PMDxMDOUT(VExOUTCR) の<WPWM>,<VPWM>,<UPWM>により設定します。PWM 出力を選択すると PWM 波形が、High/Low 出力を選択すると High 固定または Low 固定の出力が得られます。それぞれの出力を High にするか Low にするかは PMDxMDOUT(VExOUTCR)の<WOC>,<VOC>,<UOC>で選択します。PMDxMDOUT (VExOUTCR)と PMDxMDPOT によるポート出力設定と PMD 制御レジスタ(PMDxMDCR)のポート出力モード設定によって得られる端子出力の関係については「表 16-4 <UOC>,<VOC>,<WOC>および <UPWM>,<VPWM>,<WPWM>の各ビット設定によるポート出力」を参照してください。

通電制御回路はエンコーダ入力回路(A-ENC)で PWM 同期サンプリングするための PWM 信号(PWMON)を出力します。

表 16-3 PMDxMDOUT(VExOUTCR)バッファの更新タイミング

		PSYNCS 設定			
		00	01	10	11
SYNCS 設定	00	常時更新	PWM キャリアボトム	PWM キャリアピーク	PWM キャリアピークと PWM キャリアボトム
	01	INTENC0x 発生時	INTENC0x 発生毎に最初の PWM キャリアボトム	INTENC0x 発生毎に最初の PWM キャリアピーク	INTENC0x 発生毎に最初の PWM キャリアピークか PWM キャリアボトム
	10	INTTBx0 発生時	INTTBx0 発生毎に最初の PWM キャリアボトム	INTTBx0 発生毎に最初の PWM キャリアピーク	INTTBx0 発生毎に最初の PWM キャリアピークか PWM キャリアボトム
	11	ENCx CTRGO (MCMP コンペア成立) 発生時	ENCx CTRGO 発生毎に最初の PWM キャリアボトム	ENCx CTRGO 発生毎に最初の PWM キャリアピーク	ENCx CTRGO 発生毎に最初の PWM キャリアピークか PWM キャリアボトム

注) PMD 禁止(PMDxMDCR<PMWEN> = "0")にすると保持されているトリガ条件はクリアされます。

16.2.5.1　PMDxMDPOT (PMD 出力設定レジスタ)

	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	-	-	-	-	-	-	SYNCS	
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	POLH	POLL	PSYNCS	
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。
9-8	SYNCS[1:0]	R/W	PMDxMDOUT(VExOUTCR)設定転送タイミング選択(トリガ同期設定) 00：非同期 01：INTENC0x (ENCx 割り込み要求) 発生時 10：INTTBx0 (TMRBx 割り込み要求) 発生時 11：CTRGO(ENCx MCMP コンペア成立)発生時 通電制御レジスタのバッファ更新タイミングを選択します。 注 1) バッファ更新タイミングは、<PSYNC>設定と<SYNCS>設定の組み合わせで決まります。(「表 16-3 PMDxMDOUT(VExOUTCR)バッファの更新タイミング」参照) 注 2) PMD 禁止(PMDxMDEN<PWMEN> = "0") 時は設定に関わらず非同期になります。
7-4	−	R	リードすると"0"が読めます。
3	POLH	R/W	上相出力(UO,VO,WO)の出力極性選択。 0: ローアクティブ 1: ハイアクティブ
2	POLL	R/W	下相出力(XO,YO,ZO)の出力極性選択。 0: ローアクティブ 1: ハイアクティブ
1-0	PSYNCS[1:0]	R/W	PMDxMDOUT(VExOUTCR) 設定転送タイミング選択(PWM 同期設定)。 00: PWM 非同期 PMDxMDOUT/VExOUTCR レジスタ更新と同時にポート出力に反映されます。 01: キャリアボトム(<MDCNT[15:0]> = "1"の時) 10: キャリアピーク(<MDCNT[15:0]> = <MDPRD[15:0]>の時) 11: キャリアピークおよびキャリアボトム 通電制御レジスタのバッファ更新タイミングを選択します。 注 1) ノコギリ波キャリア(PMDxMDCR<PWMMD> = "0")の場合、 "00"以外ではキャリアピーク更新になります。 注 2) バッファ更新タイミングは、<PSYNC>設定と<SYNCS>設定の組み合わせで決まります。(「表 16-3 PMDxMDOUT(VExOUTCR)バッファの更新タイミング」参照) 注 3) PMD 禁止(PMDxMDEN<PWMEN> = "0")時は設定に関わらず PWM 非同期になります。

注)　PMDxMDEN<PWMEN> = "0"の状態で選択を行って下さい。

16.2.5.2 PMDxMDOUT(PMD 通電制御レジスタ)

	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	-	-	-	-	-	WPWM	VPWM	UPWM
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	WOC		VOC		UOC	
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-11	−	R	リードすると"0"が読めます。
10	WPWM	R/W	W 相 PWM 出力設定 0: H/L 出力 1: PWM 出力 <WOC>,<WPWM>,<POLH>,<POLL>,<SYNTMD>との組み合わせでポート出力を制御します。(表 16-4 参照)
9	VPWM	R/W	V 相 PWM 出力設定 0: H/L 出力 1: PWM 出力 <VOC>,<VPWM>,<POLH>,<POLL>,<SYNTMD>との組み合わせでポート出力を制御します。(表 16-4 参照)
8	UPWM	R/W	U 相 PWM 出力設定 0: H/L 出力 1: PWM 出力 <UOC>,<UPWM>,<POLH>,<POLL>,<SYNTMD>との組み合わせでポート出力を制御します。(表 16-4 参照)
7-6	−	R	リードすると"0"が読めます。
5-4	WOC[1:0]	R/W	W 相通電設定 <WOC>,<WPWM>,<POLH>,<POLL>,<SYNTMD>との組み合わせでポート出力を制御します。(表 16-4 参照)
3-2	VOC[1:0]	R/W	V 相通電設定 <VOC>,<VPWM>,<POLH>,<POLL>,<SYNTMD>との組み合わせでポート出力を制御します。(表 16-4 参照)
1-0	UOC[1:0]	R/W	U 相通電設定 <UOC>,<UPWM>,<POLH>,<POLL>,<SYNTMD>との組み合わせでポート出力を制御します。(表 16-4 参照)

注 1) バスから更新した PMDxMDOUT(VExOUTCR)レジスタをダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL0>に"0" を書き込んでバスモード(デフォルト)にしてください。

注 2) このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 3) 通電制御レジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 4) 後段バッファの更新タイミングは「表 16-3 PMDxMDOUT(VExOUTCR)バッファの更新タイミング」を参照してください。

注 5) リード時は前段バッファの値(バスから設定したデータ)をリードします。

表 16-4 <UOC>,<VOC>,<WOC>および<UPWM>,<VPWM>,<WPWM>の各ビット設定によるポート出力

PMDxMDCR<SYNTMD> = "0"

極性ハイアクティブ(PMDxMDPOT<POLH><POLL> = "11")

PMDxMDOUT 通電制御		<WPWM><VPWM><UPWM> 出力選択			
上相	下相				
<WOC[1]> <VOC[1]> <UOC[1]>	<WOC[0]> <VOC[0]> <UOC[0]>	0:H/L 出力		1: PWM 出力	
		上相出力	下相出力	上相出力	下相出力
0	0	L	L	$\overline{PWM}$	PWM
0	1	L	H	L	PWM
1	0	H	L	PWM	L
1	1	H	H	PWM	$\overline{PWM}$

PMDxMDCR<SYNTMD> = "0"

極性ローアクティブ(PMDxMDPOT<POLH><POLL> = "00")

PMDxMDOUT 通電制御		<WPWM><VPWM><UPWM> 出力選択			
上相	下相				
<WOC[1]> <VOC[1]> <UOC[1]>	<WOC[0]> <VOC[0]> <UOC[0]>	0:H/L 出力		1: PWM 出力	
		上相出力	下相出力	上相出力	下相出力
0	0	H	H	PWM	$\overline{PWM}$
0	1	H	L	H	$\overline{PWM}$
1	0	L	H	$\overline{PWM}$	H
1	1	L	L	$\overline{PWM}$	PWM

PMDxMDCR<SYNTMD> = "1"

極性ハイアクティブ(PMDxMDPOT<POLH><POLL> = "11")

PMDxMDOUT 通電制御		<WPWM><VPWM><UPWM> 出力選択			
上相	下相				
<WOC[1]> <VOC[1]> <UOC[1]>	<WOC[0]> <VOC[0]> <UOC[0]>	0:H/L 出力		1: PWM 出力	
		上相出力	下相出力	上相出力	下相出力
0	0	L	L	$\overline{PWM}$	PWM
0	1	L	H	L	$\overline{PWM}$
1	0	H	L	PWM	L
1	1	H	H	PWM	$\overline{PWM}$

PMDxMDCR<SYNTMD> = "1"

極性ローアクティブ(PMDxMDPOT<POLH><POLL> = "00")

PMDxMDOUT 通電制御		<WPWM><VPWM><UPWM> 出力選択			
上相	下相				
<WOC[1]> <VOC[1]> <UOC[1]>	<WOC[0]> <VOC[0]> <UOC[0]>	0:H/L 出力		1: PWM 出力	
		上相出力	下相出力	上相出力	下相出力
0	0	H	H	PWM	$\overline{PWM}$
0	1	H	L	H	PWM
1	0	L	H	$\overline{PWM}$	H
1	1	L	L	$\overline{PWM}$	PWM

16.2.6 保護制御回路

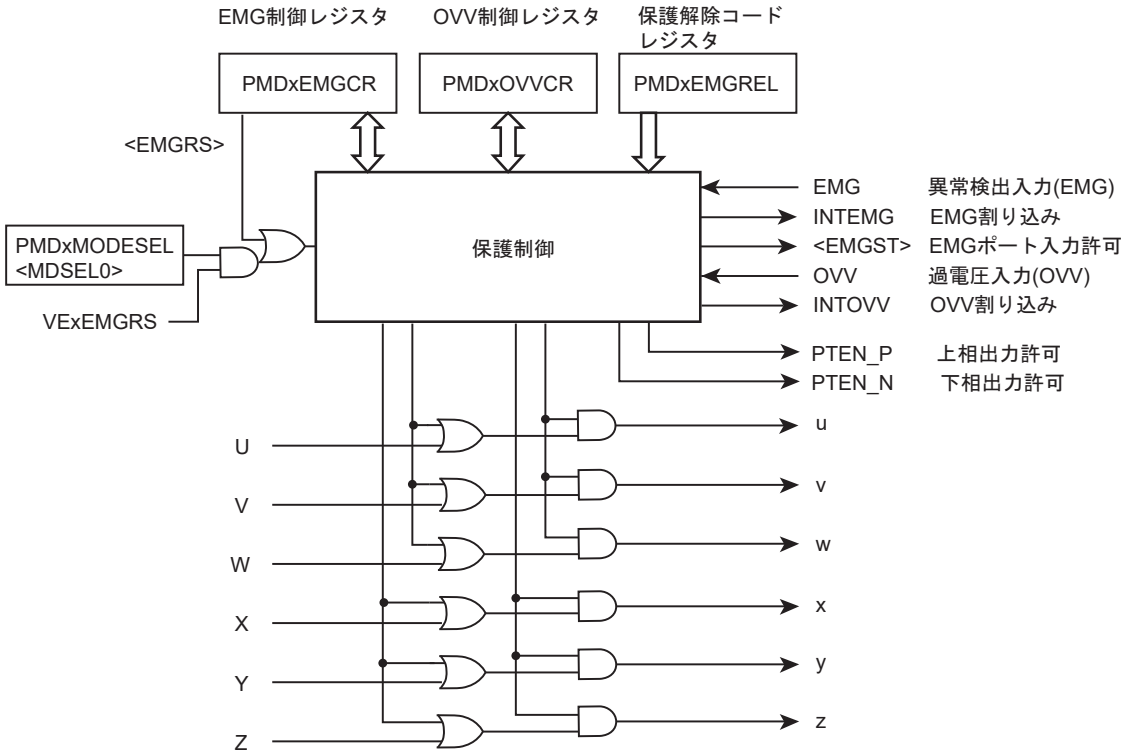


図 16-8 保護制御回路概略図

保護制御回路は、EMG 保護制御部と OVV 保護制御部から構成されます。

16.2.6.1　保護制御回路(EMG 入力部)

保護制御回路は EMG 保護制御とポート出力禁止部から構成され、EMG 入力が Low レベルになると動作します。

EMG 保護回路は緊急停止用の保護回路であり、EMG 入力があった場合(High → Low)、直ちに 6 本の PWM 出力を禁止し(PMDxEMGCR<EMGMD>の設定によります)、EMG 割り込み(INTEMG)を発生します。また<EMGMD>の設定により、外部出力ポートを"Hi-z"に設定する制御信号を出力します。

ツールブレークによる PMD 停止制御時も同様に 6 本の PWM 出力を禁止します、但し PMDxPORTMD<PORTMD>の設定によります。ツールブレーク時は PMDxPORTMD <PORTMD> の設定により、外部出力ポートの"Hi-z"制御を選択することができます。また、PMDxEMGSTA<EMGST>をリードした時、"1"の場合は EMG 保護状態であることを示します。

EMG 保護は EMG 制御レジスタ(PMDxEMGCR)で設定します。

EMG 保護状態の時は、ポート出力を全てインアクティブに設定(PMDxMDOUT(VExOUTCR)の<UPWM>,<VPWM>,<WPWM>,<UOC>,<VOC>,<WOC> を全て"0") 後 (注 1) 、 PMDxEMGCR (VExEMGRS)の<EMGRS>に"1"を設定することにより EMG 保護状態から復帰することができます。また、EMG 機能を禁止するには EMG 解除レジスタ(PMDxEMGREL)に 0x5A と 0xA5 を順番に設定後、PMDxEMGCR<EMGEN> に"0"を設定します(3 命令連続して行います)。ただし、EMG 保護入力が Low の期間は、復帰処理を行っても無視されます。PMDxEMGSTA<EMGI> をリードしポート入力が High になったことを確認してから復帰処理をしてください。

EMG 保護回路を禁止するには EMG 解除レジスタに所定のキーコード(0x5A、0xA5) を設定することではじめて可能になり、誤って EMG 保護回路を禁止することを防止します。

注 1)PMDxMDOUT(VExOUTCR)のダブルバッファまで反映されている必要があります。

注 2)リセット解除後の EMG の解除手順について

EMG 端子と兼用となっているポートは、リセット解除後はポート機能が選択されていますが、EMG 保護回路は初期状態で動作状態(許可)となっており、EMG 保護状態となっている場合がありますので、イニシャルシーケンスにて、以下の手順で EMG 保護状態を解除してください。

1: ポートのファンクションレジスタ(PxFR)にて EMG 機能を選択します。
2: PMDxEMGSTA<EMGI>をリードし、"1"であることを確認します。
3: PMDxMDOUT(VExOUTCR)の<UPWM>,<VPWM>,<WPWM>,<UOC>,<VOC>,<WOC>を全て"0"に設定し、ポート出力を全てインアクティブ出力にします。
4: PMDxEMGCR(VExEMGRS)の<EMGRS>を"1"に設定し、EMG 保護状態から復帰させます。

なお、EMG 保護を禁止とする場合は、継続して以下の手順で行ないます。

5: PMDxEMGREL に禁止コードをライトします。(0x5A → 0xA5 の順)
6: PMDxEMGCR<EMGEN>に"0"を設定し、EMG 保護回路を禁止にします。

16.2.6.2 PMDxEMGREL (EMG 解除レジスタ)

	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	EMGREL							
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	EMGREL[7:0]	W	EMG/OVV 禁止コード "0x5A"→"0xA5"に設定することにより EMG 機能と OVV 機能を禁止することができます。 禁止コード書き込み後、続けて PMDxEMGCR<EMGEN> = "0"または PMDxOVVCR<OVVEN> = "0"を設定してください。

注) EMG 禁止時と OVV 禁止時のそれぞれで禁止コードを書き込んでください。

16.2.6.3　PMDxEMGCR (EMG 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	-	-	-	-	EMGCNT			
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	INHEN	EMGMD		-	EMGRS	EMGEN
リセット後	0	0	1	1	1	0	0	1

Bit	Bit Symbol	Type	機能
31-12	−	R	リードすると"0"が読めます。
11-8	EMGCNT[3:0]	R/W	EMG 入力検出時間 0x0 ~ 0xF("0"設定時はノイズフィルタをスルーします。) 異常検出入力のノイズ除去時間を設定します。 ノイズ除去時間は以下の計算式で表されます。 <EMGCNT[3:0]> × 16/fsys (分解能 133[ns] @120[MHz])
7-6	−	R	リードすると"0"が読めます。
5	INHEN	R/W	ツールブレークの許可/禁止 0:禁止 1: 許可 ツールからの PMD 停止信号が入力された場合に PMD を停止させるかを選択します。 注) 初期状態で許可となっています。
4-3	EMGMD[1:0]	R/W	EMG 保護モード選択 00: 全相 High-z 01: 全上相オン/全下相 High-z 10: 全上相 High-z/全下相オン 11: 全相 High-z 注)オン: PWM 出力を継続します。 EMG 発生時の上相(UO,VO,WO)および下相(XO,YO,ZO)のポート出力を設定します。
2	−	R/W	常に"0"を書いてください。
1	EMGRS	W	EMG 保護状態からの復帰 0: − 1: 保護状態からの復帰 PMDxMDOUT(VExOUTCR)レジスタに"0x000"を設定後、<EMGRS>に"1"を設定することにより EMG 保護状態から復帰します。 注)リードすると常に"0"が読めます。 注) PMDxMDOUT(VExOUTCR)レジスタのダブルバッファが"0x000"に更新されていないと復帰できません。 注) PMDxEMGSTA<EMGI> が"1"になったことを確認後復帰させてください。
0	EMGEN	R/W	EMG 保護回路の許可/禁止を設定 0: 禁止 1: 許可 禁止するときは EMG 解除レジスタ PMDxEMGREL<EMGREL[7:0]>に"0x5A"→"0xA5"を順にライトした後、<EMGEN>に"0"を設定してください。(3 命令を連続して行います) 注)初期状態で許可となっています。

16.2.6.4 PMDxEMGSTA (EMG 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	-	-	-	-	-	-	EMGI	EMGST
リセット後	0	0	0	0	0	0	不定	0

Bit	Bit Symbol	Type	機能
31-2	−	R	リードすると"0"が読めます
1	EMGI	R	EMG 入力 EMG 入力 の状態 リードすることにより、EMG 入力の状態を知ることができます。
0	EMGST	R	EMG 保護状態 0: 通常動作中 1: 保護中 リードすることにより、EMG 保護の状態を知ることができます。

16.2.6.5　保護制御回路(OVV 入力部)

OVV 保護制御とポート出力禁止部から構成されます。OVV 入力ポートがアクティブになることで動作します。

OVV 保護回路は、一定期間(OVV カウントで設定)過電圧信号入力があった場合(High→Low)、通電制御部で 6 本のポート出力を High または Low に固定します。またその時、OVV 割り込み(INTOVV)を発生します。設定により、下相オフ、上相オフ、全相オフを選択することができます。

OVV 保護は OVV 制御レジスタ PMDxOVVCR で設定します。また、PMDxOVVSTA<OVVST>をリードした時、"1"の場合は OVV 保護回路が動作中であることを示します。

PMDxOVVCR<OVVRS> に"1"を設定することにより OVV 保護からの復帰可能状態となり、OVV 保護入力解除後、保護状態からの復帰タイミングで自動的に OVV 保護から復帰します。(OVV 保護入力が Low の期間は、保護状態からの復帰は行われません。ポート入力の状態は PMDxOVVSTA<OVVI> をリードすることで確認することができます。)

OVV から復帰するタイミングは、PWM 周期(PWM カウント PMDxMDCNT と PMDxMDPRD との一致時。ただし PWM0.5 周期割り込み設定時は PWM カウントが、"1"または PMDxMDPRD との一致時)となります。また、OVV 機能を禁止するには EMG 解除レジスタ PMDxEMGREL に 0x5A と 0xA5 を順番に設定後、PMDxOVVCR<OVVEN> に"0"を設定します。(3 命令連続して行ってください)

OVV 保護回路を禁止するには EMG 解除レジスタに所定のキーコード(0x5A、0xA5)を設定することではじめて可能になり、誤って OVV 保護回路を禁止することを防止します。

16.2.6.6 PMDxOVVCR (OVV 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	-	-	-	-	OVVCNT			
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	ADIN1EN	ADIN0EN	OVVMD		OVVISEL	OVVRS	OVVEN
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-12	−	R	リードすると"0"が読めます
11-8	OVVCNT[3:0]	R/W	OVV 入力検出時間 1 ~ 15 ("0" 設定時は"1"になります) OVV 入力のノイズ除去時間を設定します。 ノイズ除去時間は以下の計算式で表されます。 <OVVCNT[3:0]> × 16/fsys (分解能 133[ns] @120[MHz]) 注)<OVVCNT[3:0]> はポート入力(<OVVISEL> = "0")時のみ有効になります。
7	−	R	リードすると"0"が読めます
6	ADIN1EN	R/W	ADCx 監視機能 1 入力許可 0 : 入力禁止 1 : 入力許可 ADCx の監視機能 1 からの信号入力の許可/禁止を選択します。許可に設定し、かつ入力に ADC 監視信号(<OVVISEL> = "1")を選択時、ADCx 監視機能 1 の比較結果を OVV 入力として使用できます。(OVV 保護許可設定時) 注)ADCx 監視機能の詳細は、「12 ビットアナログ/デジタルコンバータ」の動作説明の"AD 監視機能"を参照してください。
5	ADIN0EN	R/W	ADCx 監視機能 0 入力許可 0 : 入力禁止 1 : 入力許可 ADCx の監視機能 0 からの信号入力の許可/禁止を選択します。許可に設定し、かつ入力に ADC 監視信号(<OVVISEL> = "1")を選択時、ADCx 監視機能 0 の比較結果を OVV 入力として使用できます。(OVV 保護許可設定時) 注)ADCx 監視機能の詳細は、「12 ビットアナログ/デジタルコンバータ」の動作説明の"AD 監視機能"を参照してください。
4-3	OVVMD	R/W	OVV 保護モード選択 00 : 出力制御なし 01 : 全上相オン、全下相オフ 10 : 全上相オフ、全下相オン 11 : 全相オフ OVV 発生時に、上相(UO,VO,WO)および下相(XO,YO,ZO)の出力をオン/オフさせます。 注)オンはアクティブ出力固定、オフはインアクティブ出力固定になります。アクティブ／インアクティブは<POLL>,<POLH>の設定で決まります。 注) OVV、EMG 同時発生時は<EMGMD[1:0]> での保護モード設定が優先されます。
2	OVVISEL	R/W	OVV 入力選択 0 : ポート入力 1 : ADC 監視信号 保護回路に入力させる OVV 信号をポートからの入力か、ADC からの監視信号かを選択します。 注) ADC 監視信号を選択した場合、OVV 入力検出時間<OVVCNT[3:0]> は無効となります(直接入力)。

Bit	Bit Symbol	Type	機能
1	OVVRS	R/W	OVV 保護状態からの復帰 0 : 保護状態からの自動復帰禁止 1 : 保護状態からの自動復帰許可 異常検出（OVV 入力 H → L)して OVV 保護状態へ移行しても、 自動復帰許可時は OVV 入力が High になった後の PWM 周期レジスタ(PMDxMDPRD)のバッファ更新タイミングで自動的に復帰します。(「表 16-1 PMDxMDPRD, PMDxCMPU/V/W および VExCMPU/V/W バッファ更新制御」参照)
0	OVVEN	R/W	OVV 保護回路の許可/禁止 0: 禁止 1: 許可 注) 禁止に変更する場合、EMG 解除レジスタ(PMDxEMGREL)に"0x5A" → "0xA5"を順に書き込み後、<OVVEN> = "0"に設定してください。(3 命令を連続して行います。)

16.2.6.7 PMDxOVVSTA (OVV 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	-	-	-	-	-	-	OVVI	OVVST
リセット後	0	0	0	0	0	0	不定	0

Bit	Bit Symbol	Type	機能
31-2	−	R	リードすると"0"が読めます
1	OVVI	R	OVVI 入力 OVVI の状態 リードすることにより、OVV 入力の状態(PMDxOVVCR<OVVISEL>)で選択された側)を知ることができます。
0	OVVST	R	OVV 保護状態 0: 通常動作中 1: 保護中 リードすることにより、OVV 保護の状態を知ることができます。

16.2.7　デッドタイム制御回路

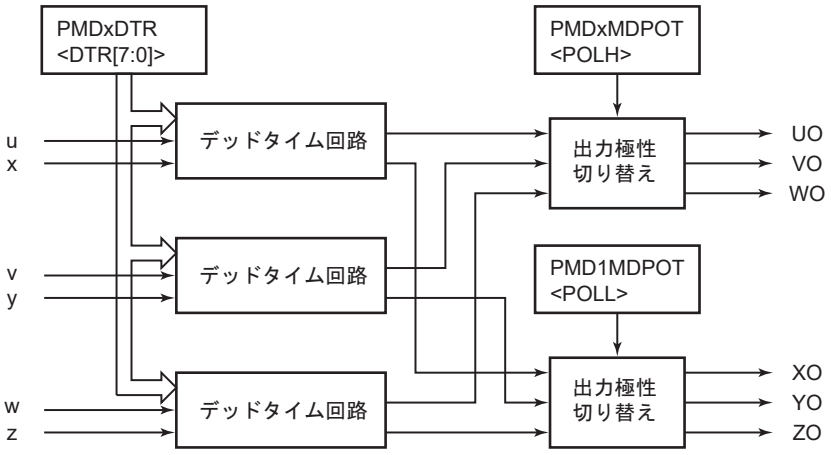


図 16-9 デッドタイム制御回路概略図

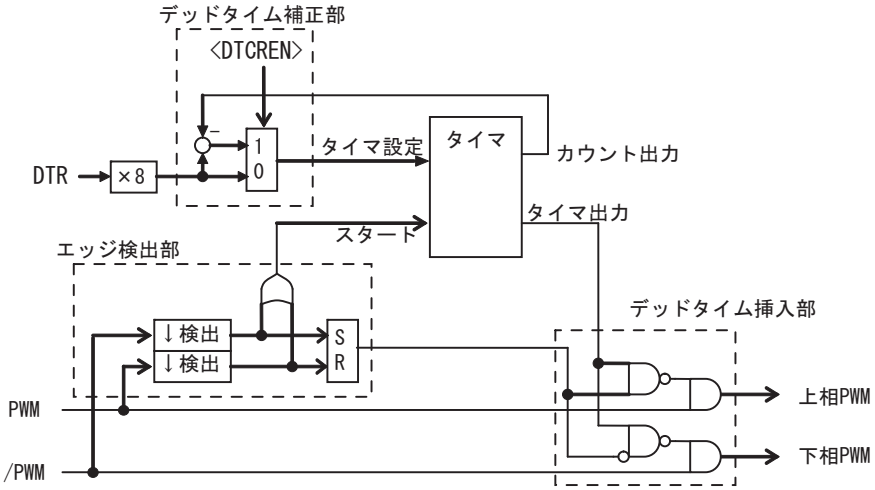


図 16-10 デッドタイム回路構成

デッドタイム制御回路はデッドタイム部と出力極性切替部から構成されます。デッドタイム回路は、エッジ検出部とタイマ部とデッドタイム挿入部およびデッドタイム補正部から構成されます。(「図 16-10 デッドタイム回路構成」参照)

U,V,W のそれぞれの相において、上下相が逆転する場合に上下相が短絡するのを回避するため、デッドタイム回路によりオン時間を遅延させます。遅延時間は、8bit 値により 8/fsys (66.7[ns]@120 [MHz]) の分解能で設定が可能です。遅延時間はデッドタイムレジスタ(PMDxDTR<DTR[7:0]>)に設定します。

出力極性切替回路は、上相出力(UO,VO,WO)と下相出力(XO,YO,ZO)のそれぞれに対してハイアクティブ/ローアクティブを PMD 出力設定レジスタ PMDxMDPOT の<POLH>,<POLL>により設定します。

デッドタイム補正部は、PMDxMDCR<DTCREN>を"1"に設定することで、上相 PWM または下相 PWM の一方のオン期間が 0 の場合にもう一方の PWM の遅延時間を短く補正します。補正方法は、デッドタイム期間の途中で PWM がオフに変わった場合に、デッドタイム期間の残り時間（デッドタイムレジスタ設定時間−オン時間）で反対相の遅延時間を短くします。上相 PWM がデッドタイム期間中にオフに変わった場合には下相の遅延時間を短く補正し、下相 PWM がデッドタイム期間中にオ

フに変わった場合には上相 PWM の遅延時間を短く補正します。遅延時間は「図 16-11 デッドタイム補正」に示すように上相 PWM のデューティ 100%付近と下相 PWM のデューティ 0%付近で補正されます。

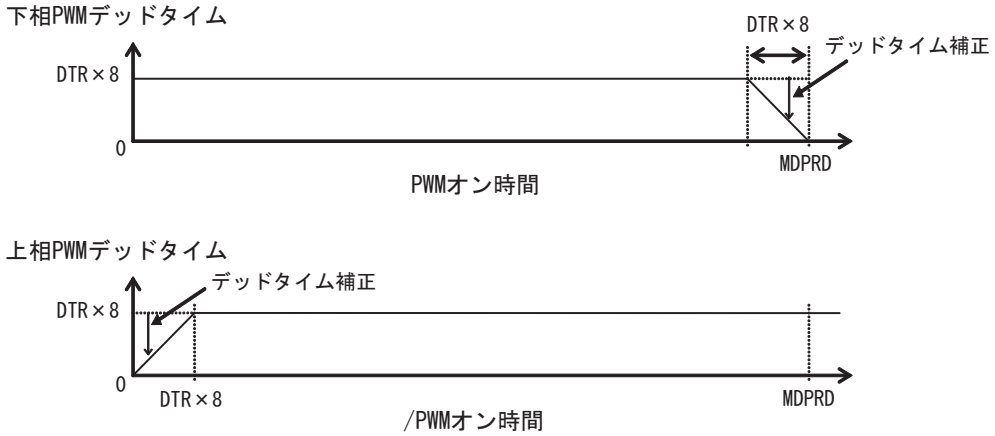


図 16-11 デッドタイム補正

16.2.7.1　PMDxDTR (デッドタイムレジスタ)

	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	DTR							
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	DTR[7:0]	R/W	デッドタイムを設定します。 0x00 ~ 0xFF デッドタイムは以下の計算式で表されます。 DTR 設定値 × 8/fsys (最大 17[μs]@120[MHz])

16.2.8 同期トリガ生成回路

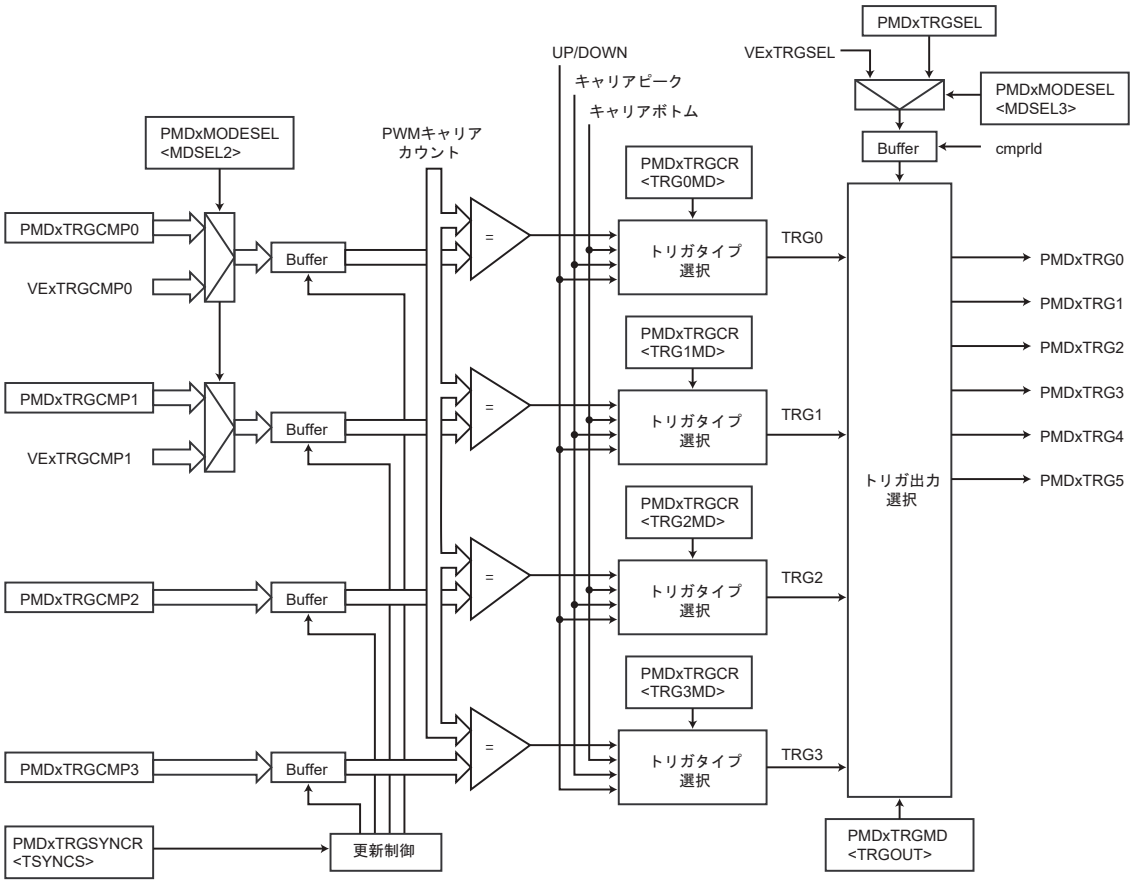


図 16-12 同期トリガ生成回路概略図

同期トリガ生成回路は PWM に同期して ADC のサンプリングを行うための 4 種類(TRG0~TRG3)のトリガ信号を生成します。PMDxMODESEL<MDSEL3>で VE モード選択すると、PMDxTRGCMP0 と PMDxTRGCMP1 は VE レジスタの VExTRGCMP0, VExTRGCMP1 に切り替わります。

トリガタイミングは以下の 6 種類が選択可能です。

1. アップカウント動作時のコンペア一致(注)
2. ダウンカウント動作時のコンペア一致(注)
3. アップ/ダウンカウント両動作でのコンペア一致(注)
4. PWM キャリアピーク
5. PWM キャリアボトム
6. PWM キャリアピークと PWM キャリアボトム

注) PWM カウンタ PMDxMDCNT<MDCNT[15:0]>と PMDxTRGCMPn<TRGCMPn[15:0]>が一致

トリガ選択出力モード PMDxTRGMD<TRGOUT> = "1" では、TRG0 信号(PMDxTRGCMP0 (VExTRGCMP0), PMDxTRGCR<TRG0MD>)がトリガ出力選択レジスタ PMDxTRGSEL (VExTRGSEL) で選択したトリガ(PMDxTRG0~5)信号として出力されます。

なお、エッジモード(ノコギリ波キャリアモード)選択時はコンペア一致はアップカウント時となります。また、EMG 保護動作時もトリガ出力許可する場合は PMDxTRGMD<EMGTGE>に"1"を設定します。

16.2.8.1 PMDxTRGCMP0 (トリガコンペアレジスタ 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	TRGCMP0							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TRGCMP0							
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	TRGCMP0 [15:0]	R/W	トリガ出力用コンペアレジスタ <TRGCMP0[15:0]>コンペアレジスタの設定範囲: 1 ~ [<MDPRD[15:0]>設定値 -1] PWM カウンタ<MDCNT[15:0]> と一致するとトリガ信号(TRG0)を出力します。 注) <TRGCMP0> = "0"および<TRGCMP0> ≥ [<MDPRD[15:0]>設定値] は設定禁止

注 1) バスから更新したコンペアレジスタをダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL2> に"0" を書き込んでバスモード(デフォルト)にしてください。

注 2) このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 3) トリガコンペアレジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 4) 後段バッファの更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照してください。

注 5) リード時は前段バッファの値(バスから設定したデータ)をリードします。

表 16-5 トリガコンペアレジスタのバッファ更新タイミング

<TSYNCS>設定	<TRGnMD>設定	TRGCMPn レジスタ バッファ更新タイミング
00	000	直ちに更新
	001	PWM キャリアピークで更新
	010	PWM キャリアボトムで更新
	011	PWM キャリアのピークとボトムで更新(注 1)
	1xx	直ちに更新
01	xxx	PWM キャリアボトムで更新
10	xxx	PWM キャリアピークで更新
11	xxx	PWM キャリアのピークとボトムで更新(注 1)

注) x : Don't care

注) PMDxMDEN<PWMEN> = "0"時は設定に関わらず直ちに更新されます。

注 1) PWM キャリアがノコギリ波(PMDxMDCR<PWMMD> = "0")の場合、キャリアピーク更新になります。

16.2.8.2　PMDxTRGCMP1 (トリガコンペアレジスタ 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	TRGCMP1							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TRGCMP1							
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	TRGCMP1 [15:0]	R/W	トリガ出力用コンペアレジスタ コンペアレジスタの設定範囲: 1 ~ [<MDPRD[15:0]> 設定値-1］ PWM カウンタ<MDCNT[15:0]> と一致するとトリガ信号(TRG1)を出力します。 注) <TRGCMP1> = "0"および<TRGCMP1> ≥ [<MDPRD[15:0]>設定値] は設定禁止

注 1)　バスから更新したコンペアレジスタをダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL2> に"0" を書き込んでバスモード(デフォルト)にしてください。

注 2)　このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 3)　トリガコンペアレジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 4)　後段バッファの更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照してください。

注 5)　リード時は前段バッファの値(バスから設定したデータ)をリードします。

16.2.8.3 PMDxTRGCMP2 (トリガコンペアレジスタ 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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TRGCMP2							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TRGCMP2							
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	TRGCMP2 [15:0]	R/W	トリガ出力用コンペアレジスタ コンペアレジスタの設定範囲: 1 ~ [<MDPRD[15:0]> 設定値-1］ PWM カウンタ<MDCNT[15:0]> と一致するとトリガ信号(TRG2)を出力します。 注) <TRGCMP2> = "0"および<TRGCMP2> ≥ [<MDPRD[15:0]>設定値] は設定禁止

注 1) このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 2) トリガコンペアレジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 3) 後段バッファの更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照してください。

注 4) リード時は前段バッファの値(バスから設定したデータ)をリードします。

16.2.8.4　PMDxTRGCMP3 (トリガコンペアレジスタ 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	TRGCMP3							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TRGCMP3							
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	TRGCMP3 [15:0]	R/W	トリガ出力用コンペアレジスタ コンペアレジスタの設定範囲: 1 ~ [<MDPRD[15:0]> 設定値-1］ PWM カウンタ<MDCNT[15:0]> と一致するとトリガ信号(TRG3)を出力します。 注) <TRGCMP3> = "0"および<TRGCMP3> ≥ [<MDPRD[15:0]>設定値] は設定禁止

注 1)　このレジスタへのバイト書き込み(上位 8bit([15:8]), 下位 8bit([7:0]を別々に書き込み)はしないでください。バイト書き込みした場合の動作は保証できません。

注 2)　トリガコンペアレジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 3)　後段バッファの更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照してください。

注 4)　リード時は前段バッファの値(バスから設定したデータ)をリードします。

16.2.8.5 PMDxTRGCR (トリガ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	TRG3BE	TRG3MD			TRG2BE	TRG2MD		
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TRG1BE	TRG1MD			TRG0BE	TRG0MD		
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	TRG3BE	R/W	PMDxTRGCMP3<TRGCMP3[15:0]>バッファの非同期更新許可 後段バッファの非同期更新を許可します。 0 : 同期更新 1 : 非同期更新(書き込み後、直ちに反映されます) 注) 更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照 注) PMDxMDEN<PWMEN> = "0"時は設定に関わらず非同期更新になります。
14-12	TRG3MD[2:0]	R/W	PMDxTRGCMP3<TRGCMP3[15:0]>のモード設定 <TRGCMP3[15:0]>トリガ出力の一致モードを選択します。 000 : トリガ出力禁止 001 : ダウンカウント時の一致でトリガ出力 010 : アップカウント時の一致でトリガ出力 011 : アップ/ダウンカウント時にトリガ出力 100 : PWM キャリアピークでトリガ出力 101 : PWM キャリアボトムでトリガ出力 110 : PWM キャリアピーク/ボトムでトリガ出力 111 : トリガ出力禁止 注) PMDxMDCR<PWMMD> = "0"(ノコギリ波)時は、"001"選択でもアップカウント一致に、"101"選択でもキャリアピークになります。 注) "011"選択時、PMDxTRGCMP3 = "0x0001" の場合、PMDxMDCR<PWMMD> = "1"(三角波)においてトリガ出力は 1 周期に 1 度となります。
11	TRG2BE	R/W	PMDxTRGCMP2<TRGCMP2[15:0]>バッファの非同期更新許可 後段バッファの非同期更新を許可します。 0 : 同期更新 1 : 非同期更新(書き込み後、直ちに反映されます) 注) 更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照 注) PMDxMDEN<PWMEN> = "0"時は設定に関わらず非同期更新になります。

Bit	Bit Symbol	Type	機能
10-8	TRG2MD[2:0]	R/W	PMDxTRGCMP2<TRGCMP2[15:0]>のモード設定 <TRGCMP2[15:0]>トリガ出力の一致モードを選択します。 000 : トリガ出力禁止 001 : ダウンカウント時の一致でトリガ出力 010 : アップカウント時の一致でトリガ出力 011 : アップ/ダウンカウント時にトリガ出力 100 : PWM キャリアピークでトリガ出力 101 : PWM キャリアボトムでトリガ出力 110 : PWM キャリアピーク/ボトムでトリガ出力 111 : トリガ出力禁止 注) PMDxMDCR<PWMMD> = "0"(ノコギリ波)時は、"001"選択でもアップカウント一致に、"101"選択でもキャリアピークになります。 注) "011"選択時、PMDxTRGCMP2 = "0x0001" の場合、PMDxMDCR<PWMMD> = "1"(三角波)においてトリガ出力は 1 周期に 1 度となります。
7	TRG1BE	R/W	PMDxTRGCMP1<TRGCMP1[15:0]>バッファの非同期更新許可 後段バッファの非同期更新を許可します。 0 : 同期更新 1 : 非同期更新(書き込み後、直ちに反映されます) 注) 更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照 注) PMDxMDEN<PWMEN> = "0"時は設定に関わらず非同期更新になります。
6-4	TRG1MD[2:0]	R/W	PMDxTRGCMP1<TRGCMP1[15:0]>のモード設定 <TRGCMP1[15:0]>トリガ出力の一致モードを選択します。 000 : トリガ出力禁止 001 : ダウンカウント時の一致でトリガ出力 010 : アップカウント時の一致でトリガ出力 011 : アップ/ダウンカウント時にトリガ出力 100 : PWM キャリアピークでトリガ出力 101 : PWM キャリアボトムでトリガ出力 110 : PWM キャリアピーク/ボトムでトリガ出力 111 : トリガ出力禁止 注) PMDxMDCR<PWMMD> = "0"(ノコギリ波)時は、"001"選択でもアップカウント一致に、"101"選択でもキャリアピークになります。 注) "011"選択時、PMDxTRGCMP1 = "0x0001" の場合、PMDxMDCR<PWMMD> = "1"(三角波)においてトリガ出力は 1 周期に 1 度となります。
3	TRG0BE	R/W	PMDxTRGCMP0<TRGCMP0[15:0]>バッファの非同期更新許可 後段バッファの非同期更新を許可します。 0 : 同期更新 1 : 非同期更新(書き込み後、直ちに反映されます) 注) 更新タイミングは「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照 注) PMDxMDEN<PWMEN> = "0"時は設定に関わらず非同期更新になります。
2-0	TRG0MD[2:0]	R/W	PMDxTRGCMP0<TRGCMP0[15:0]>のモード設定 <TRGCMP0[15:0]>トリガ出力の一致モードを選択します。 000 : トリガ出力禁止 001 : ダウンカウント時の一致でトリガ出力 010 : アップカウント時の一致でトリガ出力 011 : アップ/ダウンカウント時にトリガ出力 100 : PWM キャリアピークでトリガ出力 101 : PWM キャリアボトムでトリガ出力 110 : PWM キャリアピーク/ボトムでトリガ出力 111 : トリガ出力禁止 注) PMDxMDCR<PWMMD> = "0"(ノコギリ波)時は、"001"選択でもアップカウント一致に、"101"選択でもキャリアピークになります。 注) "011"選択時、PMDxTRGCMP0 = "0x0001" の場合、PMDxMDCR<PWMMD> = "1"(三角波)においてトリガ出力は 1 周期に 1 度となります。

16.2.8.6 PMDxTRGSYNCR (トリガ更新タイミング設定レジスタ)

	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	-	-	-	-	-	-	TSYNCS	
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。
1-0	TSYNCS	R/W	トリガコンペアレジスタのバッファ更新タイミング設定 00 : トリガ別に PMDxTRGCR<TRGxMD>設定で直ちに更新、PWM キャリアピークで更新、ＰＷＭキャリアボトムで更新、またはPWM キャリアピークとボトムで更新 01 : PWM キャリアボトムで更新 10 : PWM キャリアピークで更新 11 : PWM キャリアピークとボトムで更新 注)「表 16-5 トリガコンペアレジスタのバッファ更新タイミング」を参照 注) PMDxMDEN<PWMEN> = "0"時は設定に関わらず非同期になります。

16.2.8.7　PMDxTRGMD (トリガ出力モード設定レジスタ)

	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	-	-	-	-	-	-	TRGOUT	EMGTGE
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	TRGOUT	R/W	トリガ出力モード 0 :トリガ固定出力 1 :トリガ選択出力 トリガ固定出力では、トリガ出力 PMDxTRG0 ~ PMDxTRG3 はそれぞれ<TRGCMP0[15:0]> ~ <PMDCMP3[15:0]> で一致したトリガ信号を出力します。PMDxTRG4、PMDxTRG5 は出力されません。 トリガ選択出力では、<TRGCMP0[15:0]>の出力信号をトリガ出力 PMDxTRG0 ~ PMDxTRG5 のいづれかに出力します。トリガ出力はトリガ出力選択レジスタで選択します。 注)トリガ選択出力(<TRGOUT> = "1")時のトリガ出力パターンについては表 16-6 を参照してください。
0	EMGTGE	R/W	EMG 保護動作中の出力許可設定 0 : 保護動作時 トリガ出力禁止 1 : 保護動作時 トリガ出力許可 EMG 保護動作中のトリガ出力許可設定を選択します。

表 16-6 トリガ出力パターン

<TRGOUT> 設定	コンペアレジスタ	<TRGSEL[2:0]> 設定	トリガ出力
<TRGOUT> = "0"	PMDxTRGCMP0	×	PMDxTRG0
	PMDxTRGCMP1		PMDxTRG1
	PMDxTRGCMP2		PMDxTRG2
	PMDxTRGCMP3		PMDxTRG3
<TRGOUT> = "1"	PMDxTRGCMP0	0	PMDxTRG0
		1	PMDxTRG1
		2	PMDxTRG2
		3	PMDxTRG3
		4	PMDxTRG4
		5	PMDxTRG5
	PMDxTRGCMP1	×	トリガ出力されない
	PMDxTRGCMP2	×	トリガ出力されない
	PMDxTRGCMP3	×	トリガ出力されない

16.2.8.8 PMDxTRGSEL (トリガ出力選択レジスタ)

	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	-	-	-	-	-	TRGSEL		
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	TRGSEL[2:0]	R/W	トリガ出力ポート選択 000 : PMDxTRG0 より出力 001 : PMDxTRG1 より出力 010 : PMDxTRG2 より出力 011 : PMDxTRG3 より出力 100 : PMDxTRG4 より出力 101 : PMDxTRG5 より出力 110 : トリガ出力しない 111 : トリガ出力しない トリガ選択出力(<TRGOUT> = "1")時に有効となり、PMDxTRGCMP0 で設定した出力トリガを選択します。(「表 16-6 トリガ出力パターン」参照)

注 1) バスから更新したコンペアレジスタをダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL3>に"0"を書き込んでバスモード(デフォルト)にしてください。

注 2) トリガ出力選択レジスタはダブルバッファ構成のため PWM カウンタ動作中でも変更可能です。

注 3) 後段バッファの更新タイミングはコンペアレジスタ(PMDxCMPU/V/W)と同じです。

注 4) PMD 禁止(PMDxMDCR<PWMEN> = "0")時は非同期更新されます。

第 17 章　ベクトルエンジン(A-VE)

17.1　概要

17.1.1　特徴

1. ベクトル制御で実行される基本的な処理(座標軸変換,相変換,SIN/COS 演算)の組み込み
   演算処理は固定小数点形式データで行われます。
   →ソフトウエア処理では面倒な小数点位置管理不要

2. モータ制御回路(PMD)と A/D 変換器(ADC)を制御する I/F 処理(出力制御,トリガ生成,入力処理)の組み込み
   - 固定小数点形式の演算結果をモータ制御回路設定のデータ形式に変換
   - 連携動作タイミングデータの生成
   - A/D 変換結果を固定小数点形式のデータに変換

3. 電流,電圧,回転速度をそれぞれの最大値を基準に正規化した値で演算
   小数点数は固定小数点形式

4. 電流制御に PI 制御の組み込み

5. 回転速度を積分する位相補間の組み込み

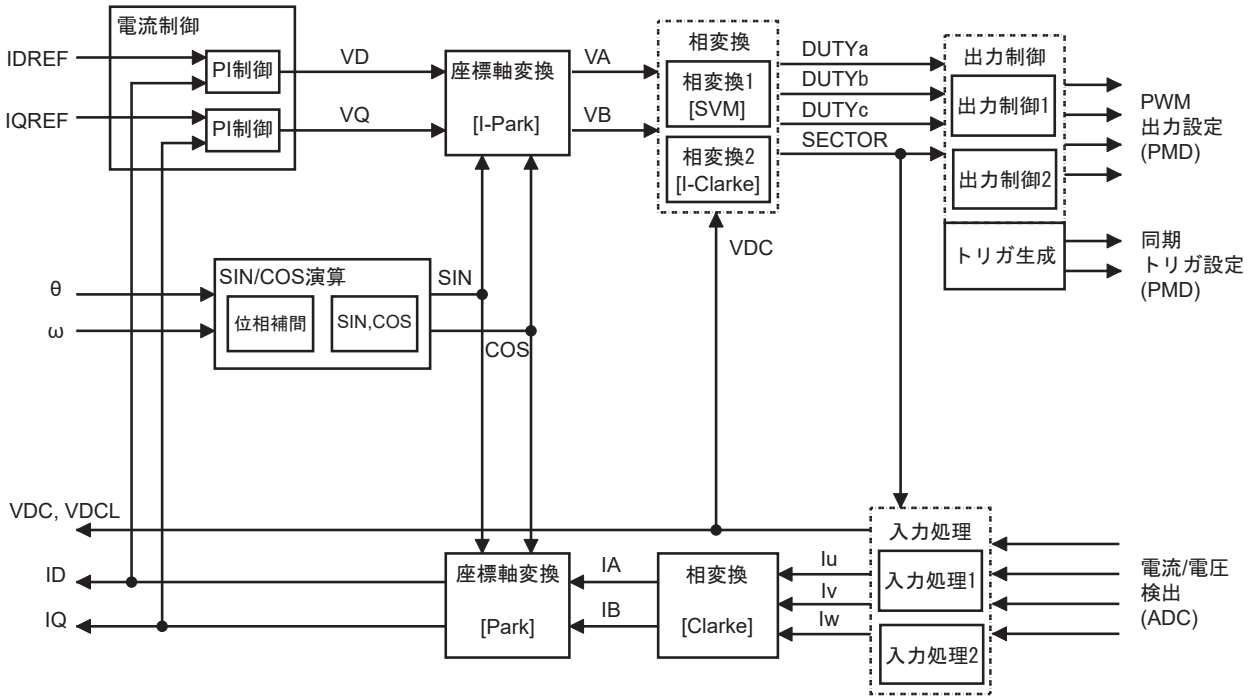


図 17-1 ベクトル制御の構成

17.1.2　主な機能

1. 2 相から 3 相への相変換には、空間ベクトル変換と逆クラーク変換に対応します。空間ベクトル変換は、2 相変調と 3 相変調に対応します。

2. トリガ生成では、センサレス電流検出方式に対応する A/D 変換器(ADC)のサンプリングタイミングを生成します。1 シャント方式の電流検出に対応するタイミングを生成できます。

3. 電流制御には、d 軸と q 軸に独立した PI 制御を採用。電流制御を使用せずに直接電圧指令することも可能です。

   PI 制御結果を制限機能

   d 軸と q 軸の非干渉制御可能

   d 軸,q 軸を合わせた電圧スカラ値による出力制限可能

4. SIN/COS 演算には、級数展開を使った近似式を採用。 位相指定は直接設定と回転速度から PWM 周期で積分する位相補間が可能です。また、位相補間時に指定位相でのクリッピング可能です。

   出力制御はデッドタイム補償および PWM 出力制限が可能です。

注)ベクトルエンジンを使用する場合はモータ制御回路および A/D 変換器の設定が必要です。

- モータ制御回路(PMD)はモード選択レジスタ(PMDxMODESEL)で VE モードを選択します。
- A/D 変換器(ADC)はモータ制御回路(PMD)からの同期トリガ毎のプログラム(トリガ許可,AIN 選択,結果レジスタ選択)を設定します。

17.2　構成

ベクトルエンジンの構成を図 17-2 に示します。

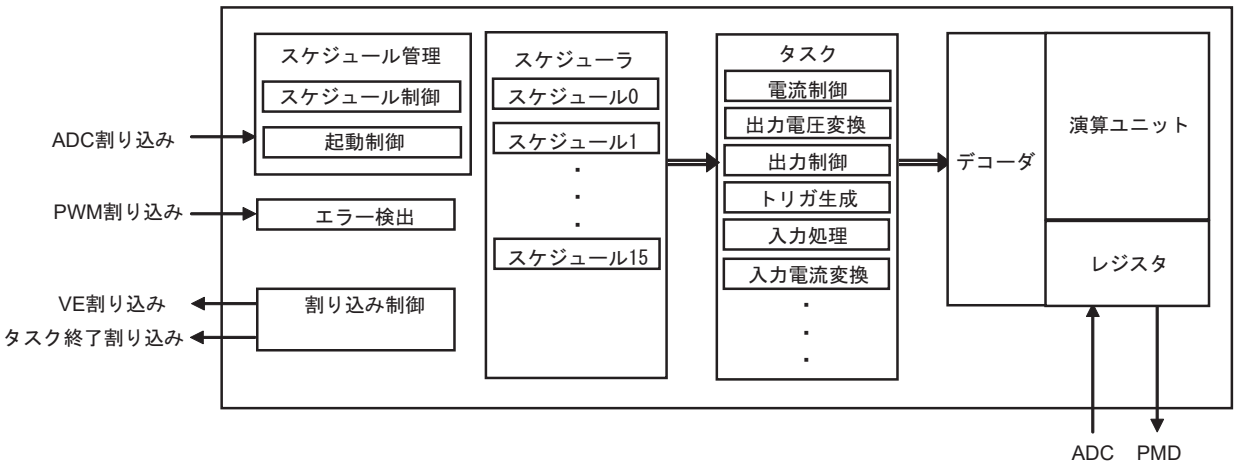


図 17-2 ベクトルエンジン構成

17.2.1 ベクトルエンジンとモータ制御回路および A/D 変換器の関連

ベクトルエンジンは図 17-3 、図 17-4 に示すようにモータ制御回路(PMD)および A/D 変換器(ADC)と直接データの受け渡しができます。

モータ制御回路(PMD)レジスタの PMDxCMPU, PMDxCMPV, PMDxCMPW, PMDxMDOUT, PMDxTRGCMP0, PMDxTRGCMP1, PMDxTRGSEL は PMDxMODESEL レジスタで VE モードに設定するとベクトルエンジンレジスタの VExCMPU, VExCMPV, VExCMPW, VExOUTCR, VExTRGCMP0, VExTRGCMP1, VExTRGSEL に切り替わります。この場合、CPU からモータ制御回路(PMD)の該当レジスタの書き込みによる制御はできず、ベクトルエンジンからの書き込みで制御します。その他の PMD のレジスタについては読み書き制限はありません。

ベクトルエンジンは A/D 変換器(ADC)の変換結果レジスタ（ADxREG0, ADxREG1, ADxREG2, ADxREG3） の値を入力処理タスクで読み込むことができます。また、2 ユニットの ADC から変換結果を読み込むことができ、2 相電流の同時サンプリングに対応します。変換結果を読み込む際には、PMD からの同期トリガ毎にプログラム設定されている相情報も ADC から読み込みます。

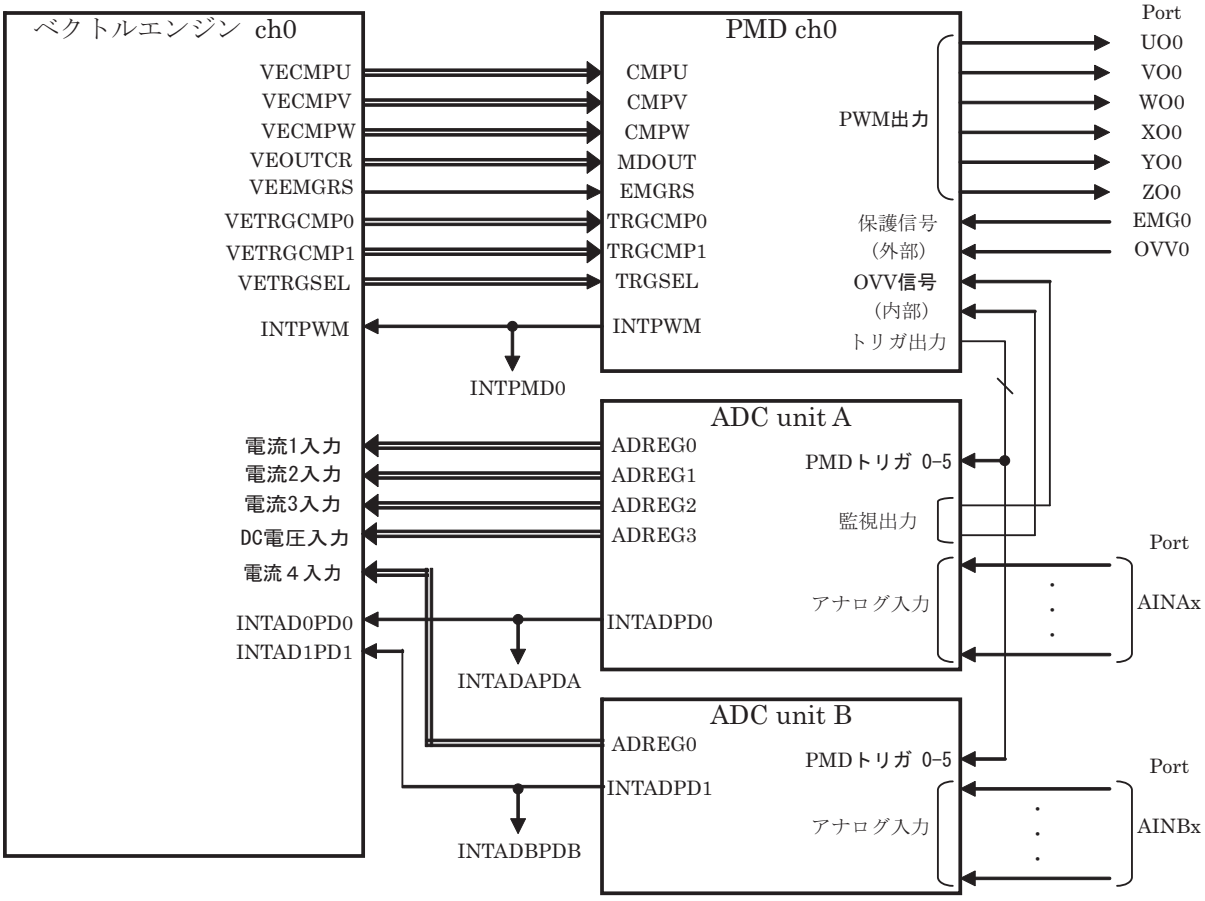


図 17-3 ベクトルエンジンとモータ制御回路および A/D 変換器の関連図(チャネル 0)

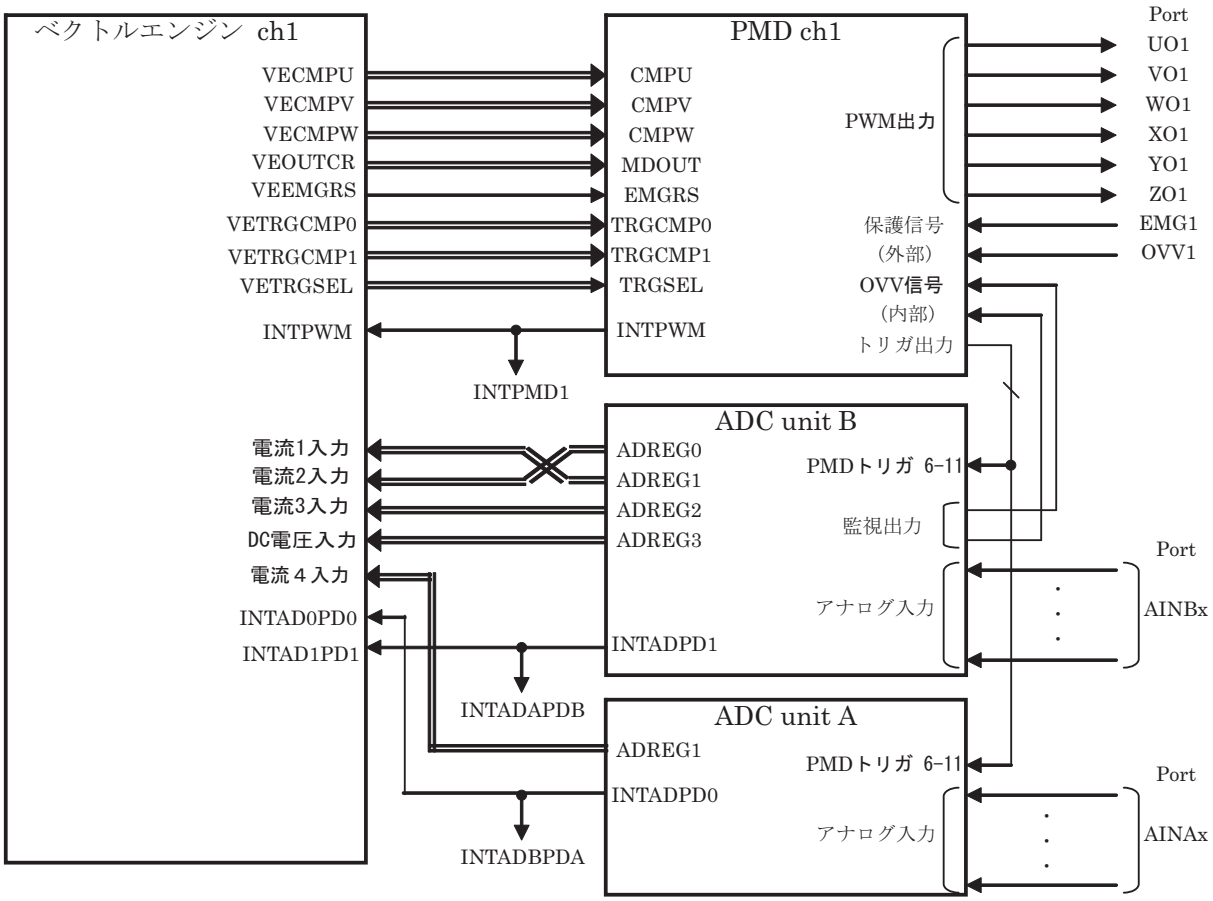


図 17-4 ベクトルエンジンとモータ制御回路および A/D 変換器の関連図(チャネル 1)

17.3 レジスタ説明

ベクトルエンジンは VE 制御レジスタと専用レジスタが存在します。

- VE 制御レジスタ
  ベクトルエンジン制御用レジスタおよびテンポラリレジスタ
- 専用レジスタ
  演算データおよび演算制御レジスタ

17.3.1 レジスター覧

VE 制御レジスタ

レジスタ名			Address
ベクトルエンジン動作許可/禁止	VExEN	R/W	0x0000
CPU 起動トリガ選択	VExCPURUNTRG	W	0x0004
タスク指定	VExTASKAPP	R/W	0x0008
動作スケジュール選択	VExACTSCH	R/W	0x000C
動作スケジュール繰り返し回数指定	VExREPTIME	R/W	0x0010
起動トリガモード設定	VExTRGMODE	R/W	0x0014
エラー割り込み許可／禁止設定	VExERRINTEN	R/W	0x0018
ベクトルエンジン強制終了	VExCOMPEND	W	0x001C
エラー検出	VExERRDET	R	0x0020
スケジュール実行中フラグ／実行中タスク	VExSCHTASKRUN	R	0x0024
Reserved	-	R	0x0028
テンポラリ 0	VExTMPREG0	R/W	0x002C
テンポラリ 1	VExTMPREG1	R/W	0x0030
テンポラリ 2	VExTMPREG2	R/W	0x0034
テンポラリ 3	VExTMPREG3	R/W	0x0038
テンポラリ 4	VExTMPREG4	R/W	0x003C
テンポラリ 5	VExTMPREG5	R/W	0x0040
Reserved	-	R	0x0214

注) "Reserved" 表記のアドレスにはアクセスしないでください。

専用レジスタ

レジスタ名			Address
異常/判定結果保持	VExMCTLF	R/W	0x0044
タスク制御モード	VExMODE	R/W	0x0048
フロー制御	VExFMODE	R/W	0x004C
PWM 周期レート(PWM 周期[s] ×最大速度(注 1) × 2^{16})設定	VExTPWM	R/W	0x0050
回転速度(速度[Hz] ÷最大速度(注 1) ×2^{15})設定	VExOMEGA	R/W	0x0054
モータ位相 (モータ位相 [deg]/360 × 2^{16})設定	VExTHETA	R/W	0x0058
d 軸基準電流値(電流[A] ÷最大電流(注 2) ×2^{15})設定	VExIDREF	R/W	0x005C
q 軸基準電流値(電流[A] ÷最大電流(注 2) ×2^{15})設定	VExIQREF	R/W	0x0060
d 軸電圧(電圧[V] ÷最大電圧(注 3) ×2^{31})設定	VExVD	R/W	0x0064
q 軸電圧(電圧[V] ÷最大電圧(注 3) ×2^{31})設定	VExVQ	R/W	0x0068
d 軸電流制御 PI 積分項係数設定	VExCIDKI	R/W	0x006C
d 軸電流制御 PI 比例項係数設定	VExCIDKP	R/W	0x0070

専用レジスタ

レジスタ名			Address
q 軸電流制御 PI 積分項係数設定	VExCIQKI	R/W	0x0074
q 軸電流制御 PI 比例項係数設定	VExCIQKP	R/W	0x0078
d 軸電圧積分成項 保持(VDI の上位 32bit)	VExVDIH	R/W	0x007C
d 軸電圧積分成項 保持(VDI の下位 32bit)	VExVDILH	R/W	0x0080
q 軸電圧積分成項 保持(VQI の上位 32bit)	VExVQIH	R/W	0x0084
q 軸電圧積分成項 保持(VQI の下位 32bit)	VExVQILH	R/W	0x0088
PWM 切り替え速度設定	VExFPWMCHG	R/W	0x008C
PWM 周期設定(PMD の PWM 周期設定と同じ値を設定)	VExMDPRD	R/W	0x0090
最小パルス幅設定	VExMINPLS	R/W	0x0094
同期トリガ補正量設定	VExTRGCRC	R/W	0x0098
DC 電源電圧(電圧[V] ÷最大電圧(注 3) ×2^{15})	VExVDCL	R/W	0x009C
THETA での余弦値 出力変換用 (Q15 データ)	VExCOS	R/W	0x00A0
THETA での正弦値 出力変換用 (Q15 データ)	VExSIN	R/W	0x00A4
前回の余弦値 入力処理用 (Q15 データ)	VExCOSM	R/W	0x00A8
前回の正弦値 入力処理用 (Q15 データ)	VExSINM	R/W	0x00AC
セクタ情報	VExSECTOR	R/W	0x00B0
前回セクタ情報	VExSECTORM	R/W	0x00B4
a 相ゼロ電流(注 4)	VExIAO	R/W	0x00B8
b 相ゼロ電流(注 4)	VExIBO	R/W	0x00BC
c 相ゼロ電流(注 4)	VExICO	R/W	0x00C0
a 相電流 ADC 変換結果(注 4)	VExIAADC	R/W	0x00C4
b 相電流 ADC 変換結果(注 4)	VExIBADC	R/W	0x00C8
c 相電流 ADC 変換結果(注 4)	VExICADC	R/W	0x00CC
電源電圧(電圧[V] ÷最大電圧(注 3) ×2^{15})	VExVDC	R/W	0x00D0
d 軸電流(電流[A] ÷最大電流(注 2) ×2^{31})	VExID	R/W	0x00D4
q 軸電流(電流[A] ÷最大電流(注 2) ×2^{31})	VExIQ	R/W	0x00D8
Reserved	-	R/W	0x0174
AD 変換時間設定	VExTADC	R/W	0x0178
PMD 制御: U 相 PWM パルス幅設定	VExCMPU	R/W	0x017C
PMD 制御: V 相 PWM パルス幅設定	VExCMPV	R/W	0x0180
PMD 制御: W 相 PWM パルス幅設定	VExCMP W	R/W	0x0184
PMD 制御: 6 相出力制御	VExOUTCR	R/W	0x0188
PMD 制御: トリガタイミング設定(TRGCMP0)	VExTRGCMP0	R/W	0x018C
PMD 制御: トリガタイミング設定(TRGCMP1)	VExTRGCMP1	R/W	0x0190
PMD 制御: 同期トリガ指定	VExTRGSEL	R/W	0x0194
PMD 制御: EMG 復帰設定	VExEMGRS	W	0x0198
PI 制御出力制限	VExPIOLIM	R/W	0x01BC
PI 制御 d 軸係数レンジ設定	VExCIDKG	R/W	0x01C0
PI 制御 q 軸係数レンジ設定	VExCIQKG	R/W	0x01C4
電圧スカラ制限	VExVSLIM	R/W	0x01C8
電圧スカラ	VExVDQ	R/W	0x01CC
偏角	VExDELTA	R/W	0x01D0
モータ鎖交磁束	VExCPHI	R/W	0x01D4
モータ q 軸インダクタンス	VExCLD	R/W	0x01D8
モータ d 軸インダクタンス	VExCLQ	R/W	0x01DC
モータ抵抗値	VExCR	R/W	0x01E0
モータ磁束レンジ設定	VExCPHIG	R/W	0x01E4
モータインダクタンスレンジ設定	VExCLG	R/W	0x01E8

専用レジスタ

レジスタ名			Address
モータ抵抗レンジ設定	VExCRG	R/W	0x01EC
非干渉制御 d 軸電圧	VExVDE	R/W	0x01F0
非干渉制御 q 軸電圧	VExVQE	R/W	0x01F4
デッドタイム補償量	VExDTC	R/W	0x01F8
電流判定ヒステリシス幅	VExHYS	R/W	0x01FC
デッドタイム補償制御／ステータス	VExDTCS	R/W	0x0200
PWM 上限設定	VExPWMMAX	R/W	0x0204
PWM 下限設定	VExPWMMIN	R/W	0x0208
位相クリップ	VExTHTCLP	R/W	0x020C

注 1)　"Reserved" 表記のアドレスにはアクセスしないでください。
注 2)　最大速度: 制御あるいは動作可能な最大回転数[Hz]
注 3)　最大電流: A/D 変換結果が 1LSB 変化する相電流の変化量[A] × 2^{11}
注 4)　最大電圧: A/D 変換結果が 1LSB 変化する電源電圧(VDC)の変化量[V] × 2^{12}
注 5)　A/D 変換結果は 16bit レジスタの上位 12bit に保存されます。

17.3.2　VE 制御レジスタ

17.3.2.1　VExEN(ベクトルエンジン動作許可／禁止レジスタ)

	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	-	-	-	-	-	-	VEIDLEN	VEEN
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	VEIDLEN	R/W	IDLE モード時のクロック動作制御 0: 停止 1: 動作
0	VEEN	R/W	ベクトルエンジン動作制御 0: 禁止 1: 許可

注)　ベクトルエンジン動作禁止(VExEN = "0")状態ではベクトルエンジンの他のレジスタにアクセスできません。

17.3.2.2 VExCPURUNTRG(CPU 起動トリガ選択レジスタ)

	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	-	-	-	-	-	-	-	VCPURT
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	VCPURT	W	ソフトウエアで起動 0: – 1: 動作開始 VExTASKAPP<VTASKA>で設定されたタスクから動作開始します。 動作開始前に VExTASKAPP, VExACTSCH および VExREPTIME を設定してください。

注 1) "1"を書き込んでも、次のサイクルでクリアされます。リードすると常に 0 が読み出されます。

注 2) スケジュール実行中に、スケジュールおよびタスクを再起動する場合、VExCOMPEND レジスタで強制終了してから、再度、VExCPURUNTRG レジスタにて動作を開始させてください。

17.3.2.3　VExTASKAPP(タスク指定レジスタ)

	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	-	-	-	-	VITASK			
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	VTASK			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-12	−	R	リードすると"0"が読めます。
11-8	VITASK[3:0]	R/W	タスク終了割り込み発生タスクを指定 0x0 : 出力制御 1 (タスク番号 0) 0x1 : トリガ生成 (タスク番号 1) 0x2 : 入力処理 1 (タスク番号 2) 0x3 : 入力相変換 (タスク番号 3) 0x4 : 入力座標軸変換 (タスク番号 4) 0x5 : 電流制御 (タスク番号 5) 0x6 : SIN/COS 演算 (タスク番号 6) 0x7 : 出力座標軸変換 (タスク番号 7) 0x8 : 出力相変換 1[SVM] (タスク番号 8) 0x9 : 出力制御 2 (タスク番号 9) 0xA : 入力処理 2 (タスク番号 10) 0xB : 出力相変換 2[I-Clarke] (タスク番号 11) 0xC : ATAN (タスク番号 12) 0xD : SQRT (タスク番号 13) 0xE,0xF : Reserved
7-4	−	R	リードすると"0"が読めます。
3-0	VTASK[3:0]	R/W	開始タスクを指定 0x0 : 出力制御 1 (タスク番号 0) 0x1 : トリガ生成 (タスク番号 1) 0x2 : 入力処理 1 (タスク番号 2) 0x3 : 入力相変換 (タスク番号 3) 0x4 : 入力座標軸変換 (タスク番号 4) 0x5 : 電流制御 (タスク番号 5) 0x6 : SIN/COS 演算 (タスク番号 6) 0x7 : 出力座標軸変換 (タスク番号 7) 0x8 : 出力相変換 1[SVM] (タスク番号 8) 0x9 : 出力制御 2 (タスク番号 9) 0xA : 入力処理 2 (タスク番号 10) 0xB : 出力相変換 2[I-Clarke] (タスク番号 11) 0xC : ATAN (タスク番号 12) 0xD : SQRT (タスク番号 13) 0xE,0xF : Reserved ソフトウエアで起動する時の開始タスクを指定します。

注)　動作スケジュールに含まれているタスク以外を指定しないでください。

17.3.2.4 VExACTSCH(動作スケジュール選択レジスタ)

	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	-	-	-	-	VACT			
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	VACT[3:0]	R/W	個別タスク実行または動作スケジュールを指定 0x0 : 個別タスク実行 0x1 : スケジュール 1 0x2 : スケジュール 2 0x3 : スケジュール 3 0x4 : スケジュール 4 0x5 : スケジュール 5 0x6 : スケジュール 6 0x7 : スケジュール 7 0x8 : スケジュール 8 0x9 : スケジュール 9 0xA : スケジュール 10 0xB : スケジュール 11 0xC : スケジュール 12 0xD : スケジュール 13 0xE : スケジュール 14 0xF : スケジュール 15 詳細は「表 17-4 スケジュール別の実行タスク」を参照。

17.3.2.5　VExREPTIME(動作スケジュール繰り返し回数指定レジスタ)

	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	-	-	-	-	VREP			
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	VREP[3:0]	R/W	動作スケジュールの繰り返し回数指定 0: スケジュール実行しない 1-15: 設定回数だけスケジュールを繰り返し実行します

注)　"0"設定時はスケジュール動作しないでください。

17.3.2.6 VExTRGMODE(起動トリガモード設定レジスタ)

	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	-	-	-	-	-	-	VTRG	
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。
1-0	VTRG[1:0]	R/W	AD 変換終了割り込みによる入力処理起動条件選択 00: 割り込み無視 01: INTADAPDA (ADCA PMD0 トリガ同期変換終了) 割り込みで起動 10: INTADBPDB (ADCB PMD1 トリガ同期変換終了) 割り込みで起動 11: Reserved

17.3.2.7　VExERRINTEN(エラー割り込み許可／禁止設定レジスタ)

	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	-	-	-	-	-	INTTEN	-	VERREN
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	INTTEN	R/W	タスク終了割り込み制御 0: 禁止 1: 許可
1	−	R	リードすると"0"が読めます。
0	VERREN	R/W	エラー検出時の割り込み制御 0: 禁止 1: 許可 動作スケジュールを実行中(起動トリガ待ちを含まない)に PWM 割り込みを検知するとエラーフラグに"1"がセットされます。

17.3.2.8 VExCOMPEND(VE 強制終了レジスタ)

	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	-	-	-	-	-	-	-	VCEND
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	VCEND	W	実行中のスケジュール強制終了 0: − 1: 停止 "1"を書き込んでも次のサイクルでクリアされます。リードすると常に"0"が読み出されます。

17.3.2.9　VExERRDET(エラー検出レジスタ)

	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	-	-	-	-	-	-	-	VERRD
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	VERRD	R	エラーフラグ 0: エラー未検出 1: エラー検出 動作スケジュールを実行中(起動トリガ待ちを含まない)に PWM 割り込みを検知すると"1"がセットされます。リードするとクリアされます。

17.3.2.10 VExSCHTASKRUN(スケジュール動作状態／実行中タスク番号レジスタ)

	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	-	-	-	VRTASK				VRSCH
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。
4-1	VRTASK[3:0]	R	実行中タスク番号 0x0: 出力制御 1 (タスク番号 0) 0x1: トリガ生成 (タスク番号 1) 0x2: 入力処理 1 (タスク番号 2) 0x3: 入力相変換 (タスク番号 3) 0x4: 入力座標軸変換 (タスク番号 4) 0x5: 電流制御 (タスク番号 5) 0x6: SIN/COS 演算 (タスク番号 6) 0x7: 出力座標軸変換 (タスク番号 7) 0x8: 出力相変換 1 [SVM] (タスク番号 8) 0x9: 出力制御 2 (タスク番号 9) 0xA: 入力処理 2 (タスク番号 10) 0xB: 出力相変換 2[I-Clarke] (タスク番号 11) 0xC: ATAN (タスク番号 12) 0xD: SQRT (タスク番号 13) 0xE: Reserved 0xF: Reserved
0	VRSCH	R	スケジュール動作状態 0: 停止 1: 実行中

17.3.2.11　VExTMPREG0 ~ 5(テンポラリレジスタ)

VExTMPREG0

	31	30	29	28	27	26	25	24
bit symbol	TMPREG0							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TMPREG0							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TMPREG0							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TMPREG0							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	TMPREG0[31:0]	R/W	テンポラリレジスタ 0

VExTMPREG1

	31	30	29	28	27	26	25	24
bit symbol	TMPREG1							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TMPREG1							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TMPREG1							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TMPREG1							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	TMPREG1[31:0]	R/W	テンポラリレジスタ 1

VExTMPREG2

	31	30	29	28	27	26	25	24
bit symbol	TMPREG2							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TMPREG2							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TMPREG2							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TMPREG2							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	TMPREG2[31:0]	R/W	テンポラリレジスタ 2

VExTMPREG3

	31	30	29	28	27	26	25	24
bit symbol	TMPREG3							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TMPREG3							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TMPREG3							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TMPREG3							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	TMPREG3[31:0]	R/W	テンポラリレジスタ 3

VExTMPREG4

	31	30	29	28	27	26	25	24
bit symbol	TMPREG4							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TMPREG4							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TMPREG4							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TMPREG4							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	TMPREG4[31:0]	R/W	テンポラリレジスタ 4

VExTMPREG5

	31	30	29	28	27	26	25	24
bit symbol	TMPREG5							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	TMPREG5							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TMPREG5							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TMPREG5							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-0	TMPREG5[31:0]	R/W	テンポラリレジスタ 5

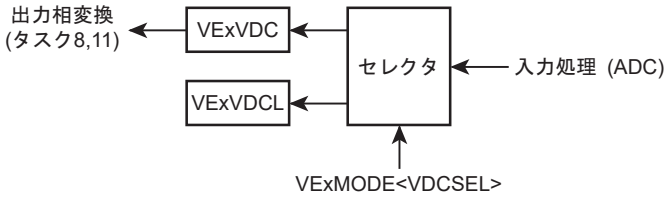
17.3.3 専用レジスタ

17.3.3.1 VExMODE(タスク制御モードレジスタ)

	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	IPDEN	PMDDTCEN	PWMFLEN	PWMBLEN	NICEN	T5ECEN	AWUMD	
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CLPEN	ATANMD		VDCSEL	OCRMD		ZIEN	PVIEN
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	IPDEN	R/W	タスク 2,タスク 10 の電流極性判定制御 0 : 判定禁止 1 : 判定許可 「17.4.2.6 入力処理」参照
14	PMDDTCEN	R/W	タスク 0,タスク 9 のデッドタイム補償時、PMD 回路のデッドタイム補正への対応制御 0 : PMD の補正が無効な場合の設定 1 : PMD の補正が有効な場合の設定 「17.4.2.4 出力制御」参
13	PWMFLEN	R/W	タスク 0,タスク 9 の PWM 上限設定時、デューティ 100%出力を許可 0 : 100%禁止 1 : 100%許可 「17.4.2.4 出力制御」参照
12	PWMBLEN	R/W	タスク 0,タスク 9 の PWM 下限設定時、デューティ 0%出力を許可 0 : 0%出力禁止 1 : 0%出力許可 「17.4.2.4 出力制御」参照
11	NICEN	R/W	タスク 5 の非干渉制御の許可 0 : 禁止 1 : 許可 「17.4.2.1 電流制御(タスク 5)」参照
10	T5ECEN	R/W	タスク 5 の拡張制御(非干渉制御,電圧スカラ制限)の許可 0 : 禁止 1 : 許可 「17.4.2.1 電流制御(タスク 5)」参照
9-8	AWUMD[1:0]	R/W	タスク 5 の PI 制御出力制限時のアンチワインドアップ(AWU)制御 00 : AWU 禁止 01 : 制限量 ÷ 4 を積分項に反映 10 : 制限量 ÷ 2 を積分項に反映 11 : 制限量を積分項に反映 「17.4.2.1 電流制御(タスク 5)」参照
7	CLPEN	R/W	タスク 6 の位相補間時の位相クリッピング制御 0 : クリッピング禁止 1 : クリッピング許可 「17.4.2.2 SIN/COS 演算(タスク 6)」参照

Bit	Bit Symbol	Type	機能
6-5	ATANMD[1:0]	R/W	タスク 4 の ATAN 演算制御 0x : 演算禁止 10 : 電流ベクトルの d-q 座標軸上の偏角算出 11 : 誘起電圧ベクトルの d-q 座標軸上の偏角算出 「17.4.2.7 入力電流変換(相変換/座標軸変換)」参照
4	VDCSEL	R/W	タスク 2、タスク 10 の電源電圧保存レジスタ選択 0 : VExVDC 保存 1 : VExVDCL 保存 「図 17-5 VExVDC/VExVDCL 保存レジスタ」参照
3-2	OCRMD[1:0]	R/W	タスク 0、タスク 9 出力制御 00 : 出力オフ 01 : 出力許可 10 : 短絡ブレーキ(出力は上相オフ、下相オン) 11 : EMG 復帰(出力オフ) 「17.4.2.4 出力制御」参照
1	ZIEN	R/W	タスク 2 のゼロ電流検出制御 0 : 通常電流検出 1 : ゼロ電流検出 「17.4.2.6 入力処理」参照
0	PVIEN	R/W	タスク 6 の位相補間制御 0 : 禁止 1 : 許可 「17.4.2.2 SIN/COS 演算(タスク 6)」参照



注) VExVDCレジスタで制御される電源電圧に補正した値を使用する場合、保存先にVExVDCLを選択し、補正値をVExVDCレジスタにセットしてください。

図 17-5 VExVDC/VExVDCL 保存レジスタ

17.3.3.2 VExFMODE(フロー制御レジスタ)

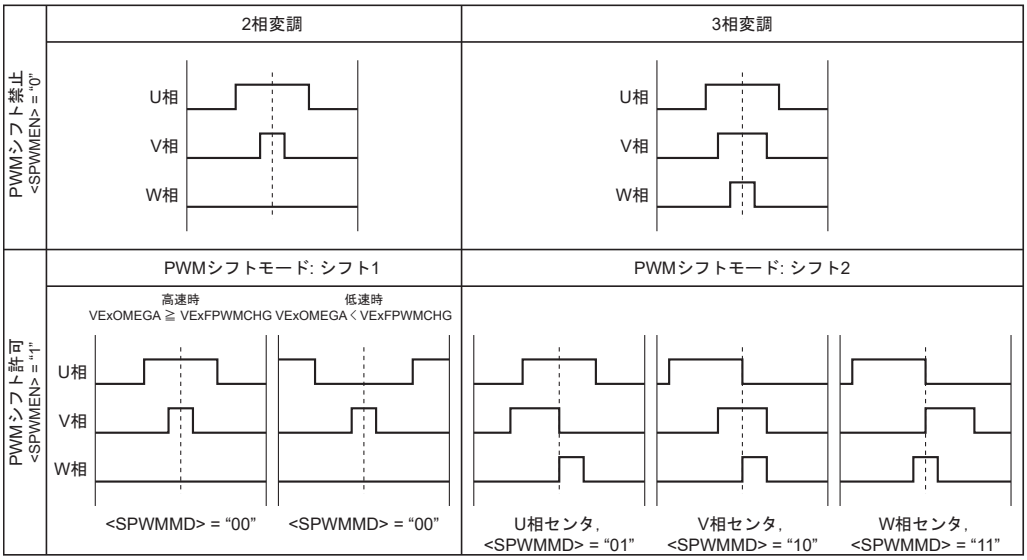
	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	SPWMMD		SADCEN	PHCVDIS	VSLIMMD		MREGDIS	CRCEN
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ICPLMD	IBPLMD	IAPLMD	-	IDMODE		SPWMEN	C2PEN
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。
15-14	SPWMMD[1:0]	R/W	PWM シフトモード選択 00 : シフト 1 01 : シフト 2 (U 相センタ) 10 : シフト 2 (V 相センタ) 11 : シフト 2 (W 相センタ) 詳細は「図 17-6 シフト設定と PWM 波形の関係」を参照。 注) 出力制御 1(タスク 0)実行時は、シフト 2 は選択できません(無効)
13	SADCEN	R/W	2 ユニットの ADC で電流の同時サンプリング制御 0 : 同時サンプリング禁止 1 : 同時サンプリング許可
12	PHCVDIS	R/W	相変換禁止 0 : 2-3 相変換許可 (3 相交流出力) 1 : 2-3 相変換禁止 (2 相交流出力) 注) 空間ベクトル変換(タスク 8)実行時は禁止できません(無効)
11-10	VSLIMMD [1:0]	R/W	タスク 5 の電圧スカラ制限制御 00 : スカラ制限禁止(各軸制限有効) 01 : d 軸方向に制限 10 : q 軸方向に制限 11 : dq 比例制限
9	MREGDIS	R/W	SIN/COS/SECTOR 前回値保持選択 0 : 有効 1 : 無効 無効時は、VExSINM = VExSIN、VExCOSM = VExCOS、VExSECTORM = VExSECTOR
8	CRCEN	R/W	トリガ補正許可 0 : 禁止 1 : 許可 注) トリガ生成(タスク 1)実行時、1 シャント電流検出モードのシフト禁止、またはシフト 1 選択時に有効
7	ICPLMD	R/W	タスク 10 の電流 Ic 検出方向設定 0 : シャントモード (Ic = VExICO − VExICADC) 1 : センサモード(Ic = VExICADC − VExICO)
6	IBPLMD	R/W	タスク 10 の電流 Ib 検出方向設定 0 : シャントモード (Ib = VExIBO − VExIBADC) 1 : センサモード (Ib = VExIBADC − VExIBO)
5	IAPLMD	R/W	タスク 10 の電流 Ia 検出方向設定 0 : シャントモード (Ia = VExIAO − VExIAADC) 1 : センサモード (Ia = VExIAADC − VExIAO)
4	−	R/W	"0"をライトしてください

Bit	Bit Symbol	Type	機能
3-2	IDMODE[1:0]	R/W	電流検出モード 00 : 3 シャント(注 1) 01 : 2 センサ(注 2) 10 : 1 シャント(PMD TRG アップカウンタ(注 3、注 4)) 11 : 1 シャント(PMD TRG ダウンカウンタ(注 3、注 4)) 注 1) 入力処理 2(タスク 10)では 3 相電流検出になります。 注 2) 入力処理 2(タスク 10)では 2 相電流検出になります。 注 4) 出力制御 2(タスク 9)および入力処理 2(タスク 10)を実行する場合は、PWM シフト 2 に設定してください。
1	SPWMEN	R/W	PWM シフト許可 0 : 禁止 1 : 制御 注)出力制御 1(タスク 0)および入力処理 1(タスク 2)はシフト 1 のみ対応。出力制御 2(タスク 9)および入力処理 2(タスク 10)はシフト 2 のみ対応。
0	C2PEN	R/W	変調モード選択 0 : 3 相変調 1 : 2 相変調

注 3) 1 シャントモード使用時の PMDTRG の設定

VExFMODE <IDMODE[1:0]>	PMDxTRGCR <TRG0MD[2:0]>	PMDxTRGCR <TRG1MD[2:0]>
10	010 (up-count)	010 (up-count)
10	101 (carrier bottom)	010 (up-count)
11	001 (down-count)	001 (down-count)
11	001 (down-count)	101 (carrier bottom)



注1) シフト1は2相変調時のみ選択可能です。
注2) シフト2は3相変調時のみ選択可能です。
注3) シフト2はPMDの設定が必要です。

図 17-6 シフト設定と PWM 波形の関係

17.3.3.3 VExTPWM(PWM 周期レート設定レジスタ)

	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	TPWM							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TPWM							
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	TPWM[15:0]	R/W	PWM 周期レート設定、位相補間時の積分単位を設定、16 ビット固定小数点データ 0.0 ~ 1.0 0x0000 ~ 0xFFFF (PWM 周期[s] × Max_Hz × 2^{16} を設定する) PWM 周波数と最大回転数との比を表します。 (Max_Hz : 最大回転数) SIN/COS 演算(タスク 6)で位相補間許可時に使用されます。

17.3.3.4　VExOMEGA(回転速度設定レジスタ)

	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	OMEGA							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	OMEGA							
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	OMEGA[15:0]	R/W	回転速度設定、16 ビット固定小数点データ−1.0 ~ 1.0 0x8000 ~ 0x7FFF (回転速度[Hz] ÷ Max_Hz × 2^{15} を設定します) (Max_Hz : 最大回転数) SIN/COS 演算(タスク 6)で位相補間許可時に使用されます。 出力制御 1(タスク 0)で 1 シャント電流検出の PWM シフト 1 選択時に使用されます。 電流制御(タスク 5),入力座標軸変換(タスク 4)でモータの電圧方程式に使用されます。

17.3.3.5 VExTHETA(モータ位相設定レジスタ)

	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	THETA							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	THETA							
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	THETA[15:0]	R/W	位相設定、16 ビット固定小数点データ 0.0 ~ 1.0 設定値 : 位相[deg] ÷ 360 × 2^{16} SIN/COS 演算(タスク 6)で使用されます。 位相補間許可で SIN/COS 演算(タスク 6)実行時に更新されます。

17.3.3.6　VExCOS/VExSIN/VExCOSM/VExSINM(SIN/COS レジスタ)

VExCOS

	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	COS							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	COS							
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	COS[15:0]	R/W	THETA 値での余弦値、16bit 固定小数点データ −1.0 ~ 1.0 0x8000 ~ 0x7FFF SIN/COS 演算(タスク 6)実行時に更新されます。 出力座標軸変換(タスク 7)で使用されます。

VExSIN

	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	SIN							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SIN							
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	SIN[15:0]	R/W	THETA 値での正弦値、16bit 固定小数点データ −1.0 ~ 1.0 0x8000 ~ 0x7FFF SIN/COS 演算(タスク 6)実行時に更新されます。 出力座標軸変換(タスク 7)で使用されます。

VExCOSM

	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	COSM							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	COSM							
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	COSM[15:0]	R/W	VExCOS レジスタの前回値保存 0x8000 ~ 0x7FFF SIN/COS 演算(タスク 6)実行時に更新されます。 入力座標軸変換(タスク 4)で使用されます。

VExSINM

	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	SINM							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	SINM							
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	SINM[15:0]	R/W	VExSIN レジスタの前回値保存 0x8000 ~ 0x7FFF SIN/COS 演算(タスク 6)実行時に更新されます。 入力座標軸変換(タスク 4)で使用されます。

17.3.3.7　VExIDREF/VExIQREF(d 軸/q 軸基準電流値設定レジスタ)

VExIDREF

	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	IDREF							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	IDREF							
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	IDREF[15:0]	R/W	d 軸電流指令値 16bit 固定小数点データ −1.0 ~ 1.0 0x8000 ~ 0x7FFF (d 軸電流指令[A] ÷ Max_I × 2^{15} を設定します) Max_I: (A/D 変換が 1LSB 変化する相電流の変化量[A]) × 2^{11} 電流制御(タスク 5)で使用されます。

VExIQREF

	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	IQREF							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	IQREF							
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	IQREF[15:0]	R/W	q 軸電流指令値 16bit 固定小数点データ −1.0 ~ 1.0 0x8000 ~ 0x7FFF (q 軸電流指令[A] ÷ Max_I × 2^{15} を設定します) Max_I: (A/D 変換が 1LSB 変化する相電流の変化量[A]) × 2^{11} 電流制御(タスク 5)で使用されます。

17.3.3.8 VExVD/VExVQ(d 軸/q 軸電圧設定レジスタ)

VExVD

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

Bit	Bit Symbol	Type	機能
31-0	VD[31:0]	R/W	d 軸電圧、32bit 固定小数点データ −1.0 ~ 1.0 0x8000_0000 ~ 0x7FFF_FFFF : d 軸電圧[V] ÷ Max_V × 2^{31} Max_V: (A/D 変換が 1LSB 変化する電源電圧の変化量[V]) × 2^{12} 電流制御(タスク 5)実行時に更新されます。 出力座標軸変換(タスク 7)で使用されます。 入力座標軸変換(タスク 4)でモータの電圧方程式に使用されます。

VExVQ

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

Bit	Bit Symbol	Type	機能
31-0	VQ[31:0]	R/W	q 軸電圧、32bit 固定小数点データ −1.0 ~ 1.0 0x8000_0000 ~ 0x7FFF_FFFF : q 軸電圧[V] ÷ Max_V × 2^{31} Max_V: (A/D 変換が 1LSB 変化する電源電圧の変化量[V]) × 2^{12} 電流制御(タスク 5)実行時に更新されます。 出力座標軸変換(タスク 7)で使用されます。 入力座標軸変換(タスク 4)でモータの電圧方程式に使用されます。

17.3.3.9　VExCIDKI/VExCIDKP/VExVCIQKI/VExCIQKP(PI 制御係数レジスタ)

VExCIDKI

	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	CIDKI							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CIDKI							
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	CIDKI[15:0]	R/W	d 軸 PI 制御積分係数 0x8000 ~ 0x7FFF

VExCIDKP

	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	CIDKP							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CIDKP							
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	CIDKP[15:0]	R/W	d 軸 PI 制御比例係数 0x8000 ~ 0x7FFF

VExVCIQKI

	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	CIQKI							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CIQKI							
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	CIQKI[15:0]	R/W	q 軸 PI 制御積分係数 0x8000 ~ 0x7FFF

VExCIQKP

	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	CIQKP							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CIQKP							
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	CIQKP[15:0]	R/W	q 軸 PI 制御比例係数 0x8000 ~ 0x7FFF

17.3.3.10　VExVDIH/VExVDILH/VExVQIH/VExVQILH(PI 制御積分項保持レジスタ)

VExVDIH

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

Bit	Bit Symbol	Type	機能
31-0	VDIH[31:0]	R/W	d 軸 PI 制御の積分項(VDI)の上位 32bit

VExVDILH

	31	30	29	28	27	26	25	24
bit symbol	VDILH							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	VDILH							
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	-							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	VDILH[15:0]	R/W	d 軸 PI 制御の積分項(VDI)の下位 16bit
15-0	−	R	リードすると"0"が読めます。

注 1)　VDI は 64bit 固定小数点データ(小数 63bit −1.0 ~ 1.0)
注 2)　VDI データは 48bit で構成されます。

VExVQIH

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

Bit	Bit Symbol	Type	機能
31-0	VQIH[31:0]	R/W	q 軸 PI 制御の積分項(VQI)の上位 32bit

VExVQILH

	31	30	29	28	27	26	25	24
bit symbol	VQILH							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	VQILH							
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	-							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	VQILH[15:0]	R/W	q 軸 PI 制御の積分項(VQI)の下位 16bit
15-0	−	R	リードすると"0"が読めます。

注 1) VQI は 64bit 固定小数点データ(小数 63bit −1.0 ~ 1.0)

注 2) VQI データは 48bit で構成されます。

17.3.3.11　VExMCTLF(異常/判定結果保持レジスタ)

	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	-	-	-	-	PWMOVF	VSOVF	PIQOVF	PIDOVF
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	PLSLFM	PLSLF	-	LVTF	LAVFM	LAVF
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。
15-12	−	R/W	"0"をライトしてください。
11	PWMOVF	R/W	PWM 出力制限超過フラグ 0 : 3 相 PWM 出力の全てが、PWMMIN 以上かつ PWMMAX 以下 1 : 3 相 PWM 出力の何れかが、PWMMIN 未満または PWMMAX 超 出力制御(タスク 0,9)実行時に更新されます。
10	VSOVF	R/W	電圧スカラ制限超過フラグ 0 : 電圧スカラ ≤ VSLIM 1 : 電圧スカラ > VSLIM 電流制御(タスク 5)実行時に更新されます。
9	PIQOVF	R/W	q 軸 PI 制御出力制限超過フラグ 0 : q 軸 PI 制御出力 ≤ PIOLIM 1 : q 軸 PI 制御出力 > PIOLIM 電流制御(タスク 5)実行時に更新されます。
8	PIDOVF	R/W	d 軸 PI 制御出力制限超過フラグ 0 : d 軸 PI 制御出力 ≤ PIOLIM 1 : d 軸 PI 制御出力 > PIOLIM 電流制御(タスク 5)実行時に更新されます。
7-6	−	R/W	"0"をライトしてください。
5	PLSLFM	R/W	<PLSLF>の前回値 出力制御(タスク 0, 9)実行時に更新されます。
4	PLSLF	R/W	パルス微小フラグ 出力制御 1(タスク 0)実行時、1 シャント電流検出設定の場合 最小パルス幅差　<　VExMINPLS<MINPLS>の場合に"1"をセット 出力制御 2(タスク 9)実行時、 最小オン幅または最小オフ幅　<　VExMINPLS<MINPLS>の場合に"1"をセット
3	−	R/W	"0"をライトしてください。
2	LVTF	R/W	電源電圧低下フラグ VExVDC<VDC>　≥　1/128 の場合"0" VExVDC<VDC>　<　1/128 の場合"1" 出力相変換(タスク 8, 11)実行時に更新されます。
1	LAVFM	R/W	<LAVF>の前回値 出力制御 1(タスク 0)実行時、1 シャント電流検出で PWM シフト許可時更新

Bit	Bit Symbol	Type	機能
0	LAVF	R/W	低速度フラグ 出力制御 1(タスク 0)実行時、1 シャント電流検出で PWM シフト許可時 0 : 高速 1 : 低速 VExOMEGA<OMEGA> ≥ VExFPWMCHG<FPWMCHG>の場合"0" VExOMEGA<OMEGA> < VExFPWMCHG<FPWMCHG>の場合"1"

17.3.3.12　VExFPWMCHG(PWM 切り替え速度設定レジスタ)

	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	FPWMCHG							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	FPWMCHG							
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	FPWMCHG[15:0]	R/W	PWM シフト許可時の PWM 切り替え速度設定 0x0000 ~ 0x7FFF (切り替え速度[Hz] ÷ Max_Hz × 2^{15} を設定します) (Max_Hz : 最大回転数[Hz]) タスク 0 で、1 シャント電流検出かつ PWM シフト 1 許可時に使用されます。

17.3.3.13　VExMDPRD(PWM 周期設定レジスタ)

	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	VMDPRD							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VMDPRD							
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	VMDPRD[15:0]	R/W	PWM 周期設定 PMD 回路の PMDxMDPRD レジスタと同じ値を設定します。 出力制御(タスク 0、9)およびトリガ生成(タスク 1)で使用されます。

17.3.3.14 VExMINPLS(最小パルス幅差設定レジスタ)

	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	MINPLS							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	MINPLS							
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	MINPLS[15:0]	R/W	出力制御 1(タスク 0)実行時、1 シャント電流検出で PWM シフト許可時の最小パルス幅差(3 相 PWM (VExCMPU,VExCMPV,VExCMPW)の Duty 差の最小値)基準値を設定します。 設定値は以下の計算式となります。 パルス幅差[s] ÷ PWM カウンタクロック周期[s] 出力制御 2(タスク 9)実行時、最小パルス幅(3 相 PWM(VExCMPU,VExCMPV,VExCMPW)の Duty の最小値)基準値を設定します。 設定値は以下の計算式となります。 パルス幅[s] ÷ PWM カウンタクロック周期[s]

17.3.3.15　VExSECTOR/VExSECTORM(セクタ情報レジスタ)

VExSECTOR

	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	-	-	-	-	SECTOR			
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	SECTOR[3:0]	R/W	セクタ情報 設定値 : 0x0 ~ 0xB 出力時の回転位置を 30 度毎の 12 エリアに分けてセクタで表します。 出力相変換(タスク 8、11)の実行時に更新されます。 出力制御 1(タスク 0)で使用されます。

VExSECTOR M

	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	-	-	-	-	SECTOR M			
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	SECTORM[3:0]	R/W	前回のセクタ情報 設定値 : 0x0 ~ 0xB 入力処理で使用。 出力相変換(タスク 8、11)の実行時に更新されます。 入力処理 1(タスク 2)で使用されます。

17.3.3.16 VExIAO/VExIBO/VExICO(ゼロ電流レジスタ)

VExIAO

	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	IAO							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	IAO							
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	IAO[15:0]	R/W	a 相ゼロ電流時 ADC 変換結果保存(停止時の a 相電流の ADC 変換結果を保存) ゼロ電流検出モード選択時、入力処理 1(タスク 2)で更新されます。 ADC 変換結果取り込み時は <IAO[15:4]> に保存され、<IAO[3:0]>は "0"が保存されます。

VExIBO

	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	IBO							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	IBO							
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	IBO[15:0]	R/W	b 相ゼロ電流時 ADC 変換結果保存(停止時の b 相電流の ADC 変換結果を保存) ゼロ電流検出モード選択時、入力処理 1(タスク 2)で更新されます。 ADC 変換結果取り込み時は <IBO[15:4]> に保存され、<IBO[3:0]>は "0"が保存されます。

VExICO

	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	ICO							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ICO							
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	ICO[15:0]	R/W	c 相ゼロ電流時 ADC 変換結果保存(停止時の c 相電流の ADC 変換結果を保存) ゼロ電流検出モード選択時、入力処理 1(タスク 2)で更新されます。 ADC 変換結果取り込み時は <ICO[15:4]> に保存され、<ICO[3:0]>は "0"が保存されます。

17.3.3.17 VExIAADC/VExIBADC/VExICADC(電流 ADC 結果レジスタ)

VExIAADC

	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	IAADC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	IAADC							
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	IAADC[15:0]	R/W	a 相電流 ADC 変換結果保存 0x0000 ~ 0xFFFF 入力処理(タスク 2、10)実行時に更新されます。 ADC 変換結果は<IAADC[15:4]> に保存され、<IAADC[3:0]>は "0"が保存されます。

VExIBADC

	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	IBADC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	IBADC							
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	IBADC[15:0]	R/W	b 相電流 ADC 変換結果保存 0x0000 ~ 0xFFFF 入力処理(タスク 2、10)実行時に更新されます。 ADC 変換結果は<IBADC[15:4]> に保存され、<IBADC[3:0]>は "0"が保存されます。

VExICADC

	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	ICADC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ICADC							
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	ICADC[15:0]	R/W	c 相電流 ADC 変換結果保存 0x0000 ~ 0xFFFF 入力処理(タスク 2、10)実行時に更新されます。 ADC 変換結果は<ICADC[15:4]> に保存され、<ICADC[3:0]>は "0"が保存されます。

17.3.3.18 VExVDC/VExVDCL(電源電圧レジスタ)

VEVDCx

	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	VDC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VDC							
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	VDC[15:0]	R/W	電源電圧,16bit 固定小数点データ: 0 ~ 1.0 設定値 : 0x0000 ~ 0x7FFF 実電圧値に変換するには VDC 値 × Max_V 値 ÷ 2^{15} (Max_V : (A/D 変換が 1LSB 変化する電源電圧の変化量[V]) × 2^{12}) VExMODE<VDCSEL> = "0" 設定時、入力処理(タスク 2、10)実行時に更新されます。 出力相変換(タスク 8、11)で使用されます。

VExVDCL

	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	VDCL							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VDCL							
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	VDCL[15:0]	R/W	電源電圧,16bit 固定小数点データ: 0 ~ 1.0 設定値 : 0x0000 ~ 0x7FFF 実電圧値に変換するには、VDCL 値 × Max_V 値 ÷ 2^{15} (Max_V : (A/D 変換が 1LSB 変化する電源電圧の変化量[V]) × 2^{12}) VExMODE<VDCSEL> = "1" 設定時、入力処理(タスク 2、10)実行時に更新されます。

17.3.3.19　VExID/VExIQ(d 軸/q 軸電流レジスタ)

VExID

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

Bit	Bit Symbol	Type	機能
31-0	ID[31:0]	R/W	d 軸電流、32bit 固定小数点データ: −1.0 ~ 1.0 設定値 : 0x8000_0000 ~ 0x7FFF_FFFF 実電流値に変換するには、 ID 値 × Max_I 値 ÷ 2^{31} (Max_I : (A/D 変換が 1LSB 変化する相電流の変化量[A]) × 2^{11}) 入力座標軸変換(タスク 4)実行時に更新されます。 電流制御(タスク 5)で使用されます。

VExIQ

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

Bit	Bit Symbol	Type	機能
31-0	IQ[31:0]	R/W	q 軸電流、32bit 固定小数点データ −1.0 ~ 1.0 設定値 : 0x8000_0000 ~ 0x7FFF_FFFF 実電流値に変換するには、 IQ 値 × Max_I 値 ÷ 2^{31} (Max_I : (A/D 変換が 1LSB 変化する相電流の変化量[A]) × 2^{11}) 入力座標軸変換(タスク 4)実行時に更新されます。 電流制御(タスク 5)で使用されます。

17.3.3.20 VExTADC(ADC 変換時間設定レジスタ)

	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	TADC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TADC							
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	TADC[15:0]	R/W	ADC 変換時間の設定(図 17-7 参照) 0x0000~0xFFFF : (ADC 変換時間[s]÷PWM カウンタクロック周期[s] を設定) 1 シャント電流検出モードで PWM シフト 1 出力(シフト 1 許可かつ低速フラグ" 1")の場合、キャリアピークでのトリガタイミングを前方補正します。 タスク 0 で使用されます。

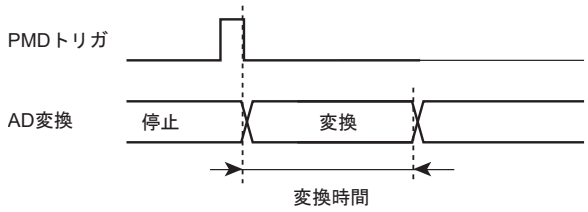


図 17-7 ADC 変換時間

17.3.3.21　VExCMPU/ VExCMPV/ VExCMPW(PWM DUTY レジスタ)

VExCMPU

	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	VCMPU							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VCMPU							
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	VCMPU[15:0]	R/W	U 相 PWM パルス幅設定 設定値 : 0x0000 ~ 0xFFFF 出力制御(タスク 0、9)実行時に更新されます。 トリガ生成(タスク 1)で使用されます。

VExCMPV

	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	VCMPV							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VCMPV							
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	VCMPV[15:0]	R/W	V 相 PWM パルス幅設定 設定値 : 0x0000 ~ 0xFFFF 出力制御(タスク 0、9)実行時に更新されます。 トリガ生成(タスク 1)で使用されます。

VExCMPW

	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	VCMPW							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VCMPW							
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	VCMPW[15:0]	R/W	W 相 PWM パルス幅設定 設定値 : 0x0000 ~ 0xFFFF 出力制御(タスク 0、9)実行時に更新されます。 トリガ生成(タスク 1)で使用されます。

17.3.3.22　VExOUTCR(6 相出力制御レジスタ)

	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	-	-	-	-	-	-	-	WPWM
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VPWM	UPWM	WOC		VOC		UOC	
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	WPWM	R/W	W 相 PWM 0 : オン/オフ出力 1 : PWM 出力
7	VPWM	R/W	V 相 PWM 0 : オン/オフ出力 1 : PWM 出力
6	UPWM	R/W	U 相 PWM 0 : オン/オフ出力 1 : PWM 出力
5-4	WOC[1:0]	R/W	W 相出力制御 00 : WO オフ,ZO オフ(注) 01 : WO オフ,ZO オン 10 : WO オン,ZO オフ 11 : WO オン,ZO オン (注)<WPWM> = 1 時は両方オン
3-2	VOC[1:0]	R/W	V 相出力制御 00 : VO オフ,YO オフ(注) 01 : VO オフ,YO オン 10 : VO オン,YO オフ 11 : VO オン,YO オン (注)<VPWM> = 1 時は両方オン
1-0	UOC[1:0]	R/W	U 相出力制御 00 : UO オフ,XO オフ(注) 01 : UO オフ,XO オン 10 : UO オン,XO オフ 11 : UO オン,XO オン (注)<UPWM> = 1 時は両方オン

注)　出力制御(タスク 0、9)実行時に更新されます。

PMD の U,V,W 相の出力制御を以下に示します。(VE で使用する組み合わせのみ表示)

表 17-1 <UPWM>,<UOC> PMD 設定: U 相(UO,XO) の出力制御

設定		出力	
<UPWM>	<UOC>	UO	XO
0	00	オフ出力	オフ出力
1	00	PWMU 反転出力	PWMU 出力
1	11	PWMU 出力	PWMU 反転出力

表 17-2 <VPWM>,<VOC> PMD 設定: V 相(VO,YO) の出力制御

設定		出力	
<VPWM>	<VOC>	VO	YO
0	00	オフ出力	オフ出力
1	00	PWMV 反転出力	PWMV 出力
1	11	PWMV 出力	PWMV 反転出力

表 17-3 <WPWM>,<WOC> PMD 設定: W 相(WO,ZO) の出力制御

設定		出力	
<WPWM>	<WOC>	WO	ZO
0	00	オフ出力	オフ出力
1	00	PWMW 反転出力	PWMW 出力
1	11	PWMW 出力	PWMW 反転出力

17.3.3.23　VExTRGCRC(同期トリガ補正量設定レジスタ)

	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	TRGCRC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TRGCRC							
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	TRGCRC[15:0]	R/W	同期トリガタイミングを補正 設定値 : 補正時間[s] ÷ PWM カウンタクロック周期[s] 1 シャント電流検出で PWM シフト禁止またはシフト 1 許可時のみ有効で、トリガタイミングを後方補正します。 トリガ生成(タスク 1)で使用されます。

17.3.3.24 VExTRGCMP0/VExTRGCMP1(トリガタイミング設定レジスタ)

VExTRGCMP0

	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	VTRGCMP0							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VTRGCMP0							
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	VTRGCMP0[15:0]	R/W	ADC を PMD と同期させてサンプリングさせるトリガタイミングの設定(PMD 設定) 0x0000 : 設定禁止 0x0001 ~ (VExMDPRD 値−1) : トリガタイミング VExMDPRD 値 ~ 0xFFFF : 設定禁止 PMD のトリガモードを次の何れかを選択時に有効。 ダウンカウント時の一致、アップカウント時の一致、アップおよびダウンカウント時の一致 トリガ生成(タスク 1)実行時、1 シャント電流検出で PWM シフト禁止またはシフト 1 許可時に更新されます。

VExTRGCMP1

	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	VTRGCMP1							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VTRGCMP1							
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	VTRGCMP1[15:0]	R/W	ADC を PMD と同期させてサンプリングさせるトリガタイミングの設定(PMD 設定) 0x0000 : 設定禁止 0x0001 ~ (VExMDPRD 値−1) : トリガタイミング VExMDPRD 値 ~ 0xFFFF : 設定禁止 PMD のトリガモードを次の何れかを選択時に有効。 ダウンカウント時の一致、アップカウント時の一致、アップおよびダウンカウント時の一致 PMD のトリガ出力モードにトリガ選択出力(PMDxTRGMD<TRGOUT> = "1")を選択時は無効。 トリガ生成(タスク 1)実行時、1 シャント電流検出で PWM シフト禁止またはシフト 1 許可時に更新されます。

17.3.3.25 VExTRGSEL(同期トリガ指定レジスタ)

	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	-	-	-	-	-	VTRGSEL		
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	VTRGSEL[2:0]	R/W	VExTRGCMP0 設定タイミングで出力する同期トリガ番号の指定(PMD 設定) 0 ~ 5 : 出力トリガ番号 6 ~ 7 : 使用禁止 PMD のトリガ出力モードにトリガ選択出力(PMDxTRGMD<TRGOUT> = "1")を選択時に有効。 トリガ生成(タスク 1)実行時、 VEx SECTOR 値 ÷ 2 に更新されます。

17.3.3.26　VExEMGRS(EMG 復帰設定レジスタ)

	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	-	-	-	-	-	-	-	EMGRS
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	EMGRS	W	EMG 復帰コマンド(PMD 設定) 0: - 1: EMG 復帰コマンド "1"を書き込んでも次のサイクルでクリアされます。リードすると常に"0"が読み出されます。 EMG 復帰モードで出力制御(タスク 0、9)実行時に "1" がセットされます。

17.3.3.27　VExPIOLIM(PI 制御出力制限レジスタ)

	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	PIOLIM							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PIOLIM							
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	PIOLIM[15:0]	R/W	PI 制御出力制限 設定値 : 0x0000 ~ 0x7FFF 電流制御(タスク５)で使用されます。

17.3.3.28 VExCIDKG/VExCIQKG(PI 制御 d 軸/q 軸係数レンジ設定レジスタ)

VExCIDKG

	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	CIDKG							
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	CIDKG[7:0]	R/W	PI 制御 d 軸係数レンジ選択 0x00 : 1/1 0x01 : $1/2^4$ 0x02 : $1/2^8$ 0x03 : $1/2^{12}$ 0x04 : $1/2^{16}$ 0x05 ~ 0xFF : Reserved 電流制御(タスク５)で使用されます。

VExCIQKG

	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	CIQKG							
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	CIQKG[7:0]	R/W	PI 制御 q 軸係数レンジ選択 0x00 : 1/1 0x01 : $1/2^4$ 0x02 : $1/2^8$ 0x03 : $1/2^{12}$ 0x04 : $1/2^{16}$ 0x05 ~ 0xFF : Reserved 電流制御(タスク５)で使用されます。

17.3.3.29　VExVSLIM(電圧スカラ制限レジスタ)

	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	VSLIM							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VSLIM							
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	VSLIM[15:0]	R/W	d 軸電圧(VExVD)、q 軸電圧(VExVQ)の電圧スカラの制限値設定 0x0000 ~ 0x7FFF : 電圧[V] ÷ Max_V × 2^{15} Max_V: (A/D 変換が 1LSB 変化する電源電圧の変化量[V]) × 2^{12} 0x0000 設定時、スカラ制限は実行されません。 電流制御(タスク５)で拡張制御が有効な場合に使用されます。

17.3.3.30 VExVDQ(電圧スカラレジスタ)

	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	VDQ							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VDQ							
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	VDQ[15:0]	R/W	d 軸電圧(VExVD)、q 軸電圧(VExVQ)の電圧スカラ値または軸方向スカラ制限時の軸制限値 0x0000 ~ 0x7FFF : 電圧[V] ÷ Max_V × 2^{15} Max_V: (A/D 変換が 1LSB 変化する電源電圧の変化量[V]) × 2^{12} 電流制御(タスク５)で拡張制御が有効な場合に更新されます。

17.3.3.31　VExDELTA(偏角レジスタ)

	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	DELTA							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	DELTA							
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	DELTA[15:0]	R/W	d-q 座標上の偏角 範囲 : 0x8000 ~ 0x7FFF(−180° ~ 180°) 偏角[deg] ÷ 360 × 2^{16} 座標軸変換（タスク４）で ATAN 演算、電流制御(タスク５)で電圧スカラ制限が有効な場合に更新されます。

17.3.3.32 VExCPHI/VExCLD/VExCLQ/VExCR/VExCPHIG/VExCLG/VExCRG(モータ定数レジスタ)

VExCPHI

	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	CPHI							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CPHI							
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	CPHI[15:0]	R/W	モータ定数の逆起電力定数 [V/rps] (鎖交磁束[Wb/s]) 設定値 : 0x0000 ~ 0x7FFF 逆起電力定数値[V/rps] ÷ Max_V × Max_Hz × 2^{11} ÷ [CPHIG 設定] 標軸変換（タスク４）で誘起電圧の偏角算出、電流制御(タスク５)で非干渉制御が有効な場合に使用されます。

VExCLD

	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	CLD							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CLD							
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	CLD[15:0]	R/W	モータ定数の d 軸インダクタンス 設定値 : 0x0000 ~ 0x7FFF インダクタンス値[H] × Max_I ÷ Max_V × Max_Hz × 2π × 2^{11} ÷ [CLG 設定] 標軸変換（タスク４）で誘起電圧の偏角算出、電流制御(タスク５)で非干渉制御が有効な場合に使用されます。

VExCLQ

	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	CLQ							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CLQ							
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	CLQ[15:0]	R/W	モータ定数の q 軸インダクタンス 設定値 : 0x0000 ~ 0x7FFF インダクタンス値[H] × Max_I ÷ Max_V × Max_Hz × 2π × 2^{11} ÷ [CLG 設定] 座標軸変換（タスク４）で誘起電圧の偏角算出、電流制御(タスク５)で非干渉制御が有効な場合に使用されます。

VExCR

	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	CR							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CR							
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	CR[15:0]	R/W	モータ定数の抵抗 設定値 : 0x0000 ~ 0x7FFF 抵抗値[Ω] × Max_I ÷ Max_V × 2^{11} ÷ [CRG 設定] 座標軸変換（タスク４）で誘起電圧の偏角算出、電流制御(タスク５)で非干渉制御が有効な場合に使用されます。

VExCPHIG

	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	CPHIG							
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	CPHIG[7:0]	R/W	モータ定数の磁束レンジ選択 0x00 : 1/1 0x01 : $1/2^4$ 0x02 : $1/2^8$ 0x03 : $1/2^{12}$ 0x04 : $1/2^{16}$ 0x05 ~ 0xFF : Reserved 座標軸変換（タスク４）で誘起電圧の偏角算出、電流制御(タスク５)で非干渉制御が有効な場合に使用されます。

VExCLG

	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	CLG							
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	CLG[7:0]	R/W	モータ定数のインダクタンスレンジ選択 0x00 : 1/1 0x01 : $1/2^4$ 0x02 : $1/2^8$ 0x03 : $1/2^{12}$ 0x04 : $1/2^{16}$ 0x05 ~ 0xFF : Reserved 座標軸変換（タスク４）で誘起電圧の偏角算出、電流制御(タスク５)で非干渉制御が有効な場合に使用されます。

VExCRG

	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	CRG							
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	CRG[7:0]	R/W	モータ定数の抵抗レンジ選択 0x00 : 1/1 (CR レジスタは Q27 フォーマットの[31:16]) 0x01 : $1/2^4$ 0x02 : $1/2^8$ 0x03 : $1/2^{12}$ 0x04 : $1/2^{16}$ 0x05 ~ 0xFF : Reserved 座標軸変換（タスク４）で誘起電圧の偏角算出、電流制御(タスク５)で非干渉制御が有効な場合に使用されます。

17.3.3.33　VExVDE/VExVQE(非干渉制御 d 軸/q 軸電圧レジスタ)

VExVDE

	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	VDE							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VDE							
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	VDE[15:0]	R/W	非干渉制御の d 軸演算値、１６ビット固定小数点データ −1.0 ~ 1.0 0x8000 ~ 0x7FFF : 電圧[V] ÷ Max_V × 2^{15} (Max_V : (A/D 変換が 1LSB 変化する電圧の変化量[V]) × 2^{12}) 電流制御(タスク５)で非干渉制御が有効な場合に更新されます。

VExVQE

	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	VQE							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VQE							
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	VQE[15:0]	R/W	非干渉制御の q 軸演算値、１６ビット固定小数点データ −1.0 ~ 1.0 0x8000 ~ 0x7FFF : 電圧[V] ÷ Max_V × 2^{15} (Max_V : (A/D 変換が 1LSB 変化する電圧の変化量[V]) × 2^{12}) 電流制御(タスク５)で非干渉制御が有効な場合に更新されます。

17.3.3.34 VExDTC(デッドタイム補償レジスタ)

	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	DTC							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	DTC							
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	DTC	R/W	デッドタイム補償制御の補償量を設定 0x0000 ~ 0xFFFF : デッドタイム[s] ÷ PWM 周期[s] × VExMDPRD 値 出力制御(タスク０，９)でデッドタイム補償が有効な場合に使用されます。

17.3.3.35 VExHYS(電流判定ヒステリシスレジスタ)

	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	HYS							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	HYS							
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	HYS[15:0]	R/W	電流極性判定時の電流のヒステリシス幅を設定、１６ビット固定小数点データ −1.0 ~ 1.0 0x8000 ~ 0x7FFF : ヒステリシス幅[A] ÷ Max_I × 2^{15} (Max_I : (A/D 変換が 1LSB 変化する相電流の変化量[A]) × 2^{11}) 入力処理(タスク２，１０)で電流極性判定許可時に使用されます。

17.3.3.36　VExDTCS(デッドタイム補償制御/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	-	-	-	-	-	ICSTS		
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	IBSTS			-	IASTS		
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。
15-11	−	R/W	"0"をライトしてください
10-8	ICSTS[1:0]	R/W	電流 Ic 極性判定状態および CMPW のデッドタイム補償制御 xx0 : 極性判定未確定、デッドタイム補償は行われません x01 : 正電流確定、デッドタイム補償時に DTC 値を加算 x11 : 負電流確定、デッドタイム補償時に DTC 値を減算
7	−	R/W	"0"をライトしてください
6-4	IBSTS[1:0]	R/W	電流 Ib 極性判定状態および CMPV のデッドタイム補償制御 xx0 : 極性判定未確定、デッドタイム補償は行われません x01 : 正電流確定、デッドタイム補償時に DTC 値を加算 x11 : 負電流確定、デッドタイム補償時に DTC 値を減算
3	−	R/W	"0"をライトしてください
2-0	IASTS[1:0]	R/W	電流 Ia 極性判定状態および CMPU のデッドタイム補償制御 xx0 : 極性判定未確定、デッドタイム補償は行われません x01 : 正電流確定、デッドタイム補償時に DTC 値を加算 x11 : 負電流確定、デッドタイム補償時に DTC 値を減算 入力処理(タスク２，１０)で電流極性判定許可時に更新されます。 出力制御(タスク０，９)でデッドタイム補償が有効な場合に使用されます。

17.3.3.37 VExPWMMAX/VExPWMMIN(PWM 出力制限レジスタ)

VExPWMMAX

	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	PWMMAX							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PWMMAX							
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	PWMMAX[15:0]	R/W	PWM 出力の上限値を設定 設定値 : 0x0000 ~ 0xFFFF 出力制御(タスク０、９)で使用されます。

VExPWMMIN

	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	PWMMIN							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PWMMIN							
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	PWMMIN[15:0]	R/W	PWM 出力の下限値を設定 設定値 : 0x0000 ~ 0xFFFF 出力制御(タスク０、９)で使用されます。

17.3.3.38　VExTHTCLP(位相クリップレジスタ)

	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	THTCLP							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	THTCLP							
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	THTCLP[15:0]	R/W	位相補間時に THETA をクリッピングする位相設定、16 ビット固定小数点データ 0.0 ~ 1.0 設定値 : 位相[deg] ÷ 360 × 2^{16} SIN/COS 演算(タスク５)で位相補間許可時に使用されます。

17.4 動作説明

17.4.1 スケジュール管理

モータ制御は図 17-8 のようなフローで実行されます。ベクトルエンジンは各動作状態をスケジュール設定(VExACTSCH)とモード設定(VExMODE)を切り替えることで遷移させます。

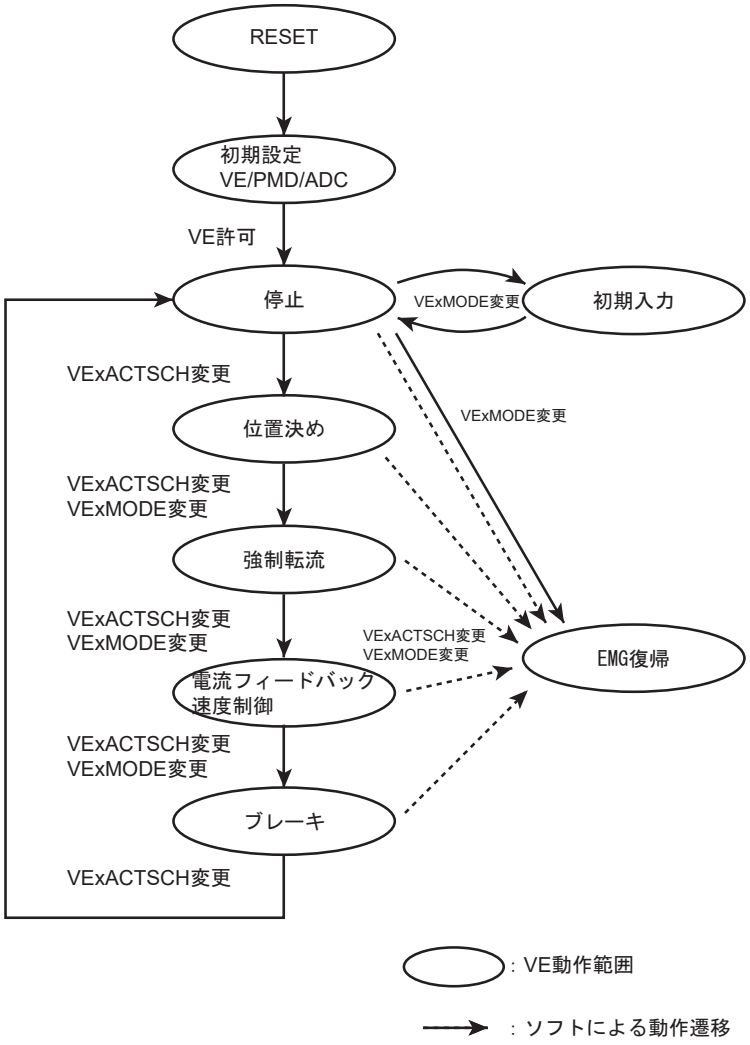


図 17-8 モータ制御の動作状態フロー例

RESET	:マイコンリセット
初期設定	:ユーザソフトによる初期設定
停止	:モータ停止
初期入力	:停止時のゼロ電流をサンプリングして保存
位置決め	:モータ起動時の位置決め制御
強制転流	:モータ起動 所定時間はフィードバック制御せずに設定速度で回転させる
電流フィードバック速度制御	:電流フィードバックによる制御
ブレーキ	:減速制御
EMG 復帰	: EMG 保護状態から復帰する

17.4.1.1 スケジュール制御

動作スケジュールは VExACTSCH レジスタで選択します。

スケジュールは出力処理を行う出力スケジュールと入力処理を行う入力スケジュールで構成されます。出力スケジュールは出力関連のタスク、入力スケジュールは入力関連のタスクで構成されます。スケジュールと動作するタスクの関連を 表 17-4 に示します。

モータ制御方法に応じて、専用レジスタでタスク動作を設定してください。

表 17-4 スケジュール別の実行タスク

スケジュール指定 VExACTSCH <VACT[3:0]>	出力スケジュール実行タスク								入力スケジュール実行タスク				個別実行のみ	
	電流制御	SIN/COS 演算	出力座標軸変換	出力相変換 1	出力相変換 2	出力制御 1	出力制御 2	トリガ生成	入力処理 1	入力処理 2	入力相変換	入力座標軸変換	ATAN 演算	平方根演算
	タスク 5	タスク 6	タスク 7	タスク 8	タスク 11	タスク 0	タスク 9	タスク 1	タスク 2	タスク 10	タスク 3	タスク 4	タスク 12	タスク 13
0 : 個別タスク実行	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)
1 : スケジュール 1	o	o	o	o	−	o	−	o	o	−	o	o	−	−
2 : スケジュール 2	o	o	o	o	−	−	o	o	−	o	o	o	−	−
3 : スケジュール 3	o	o	o	−	o	−	o	o	−	o	o	o	−	−
4 : スケジュール 4	−	o	o	o	−	o	−	o	o	−	o	o	−	−
5 : スケジュール 5	−	o	o	o	−	−	o	o	−	o	o	o	−	−
6 : スケジュール 6	−	o	o	−	o	−	o	o	−	o	o	o	−	−
7 : スケジュール 7	−	o	o	−	o	o	−	o	o	−	o	o	−	−
8 : スケジュール 8	o	o	o	−	o	o	−	o	o	−	o	o	−	−
9 : スケジュール 9	−	−	−	−	−	o	−	o	o	−	−	−	−	−
10: スケジュール 10	o	o	o	o	−	o	−	−	−	−	−	−	−	−
11: スケジュール 11	o	o	o	o	−	−	o	−	−	−	−	−	−	−
12: スケジュール 12	o	o	o	−	o	−	o	−	−	−	−	−	−	−
13: スケジュール 13	o	o	o	−	o	o	−	−	−	−	−	−	−	−
14: スケジュール 14	−	−	−	−	−	−	−	o	o	−	o	o	−	−
15:スケジュール 15	−	−	−	−	−	−	−	o	−	o	o	o	−	−

注 1) VExTASKAPP で指定したタスクのみ実行されます。

注 2) o: 実行するタスク、−: 実行しないタスク

表 17-5 代表動作フローでの設定例

設定 モータ制御フロー	スケジュール設定 VExACTSCH <VACT[3:0]>	タスク指定 VExTASKAPP <VTASK[3:0]>	位相補間許可 VExMODE <PVIEN>	出力制御動作 VExMODE <OCRMD[1:0]>	ゼロ電流検出 VExMODE <ZIEN>
停止	9	0	x	00	0
初期入力	9	0	x	00	1
位置決め	1	5	0	01	0
強制転流	1	5	1	01	0
電流フィードバック 速度制御	1	5	1	01	0
ブレーキ	4	6	0	01	0
EMG 復帰	9	0	x	11	0
短絡ブレーキ	4	6	x	10	0

出力スケジュールはコマンド(VExCPURUNTRG)で動作開始し、出力関連タスクが全て終了するとベクトルエンジンは待機状態に移行して起動トリガ(VExTRGMODE 設定)を待ちます。

入力スケジュールは起動トリガで動作開始し、入力関連タスクが全て終了すると CPU に割り込みを発生してベクトルエンジンは休止状態に移行します。ただし、スケジュールの繰り返し回数(VExREPTIME)を 2 回以上に設定している場合は、設定回数に達するまでは割り込みを発生せずに出力スケジュールを起動します。

注)　スケジュール 10 ~ 15 はリピートに対応してません。(VExREPTTIME ≥ 2 でも 1 回で終了します。)

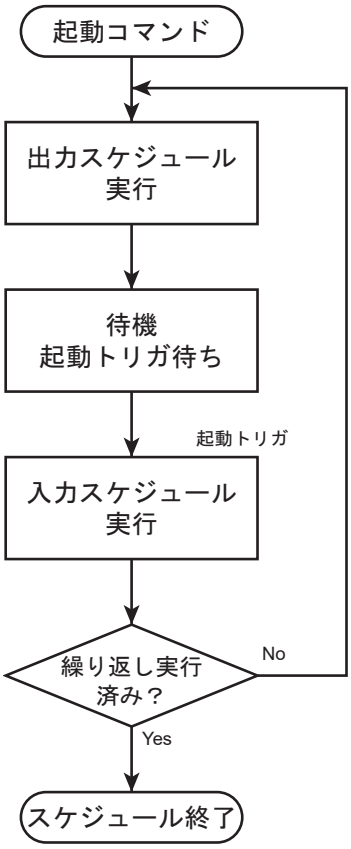


図 17-9 動作スケジュールの動作遷移

17.4.1.2 起動制御

初めに、ベクトルエンジン許可(<VEEN> = "1")して、動作スケジュール選択レジスタ(VExACTSCH)、タスク指定レジスタ(VExTASKAPP)、および動作スケジュール繰り返し回数指定レジスタ(VExREPTIME)を設定後、下記の手順でスケジュールを実行してください。

ベクトルエンジンの動作スケジュールは出力スケジュールと入力スケジュールに分かれており、基本的に出力スケジュール実行後の待機状態で起動トリガが発生すると入力スケジュールが実行されます。

出力スケジュールと入力スケジュールはそれぞれ次の条件で起動します。

- 出力スケジュールの起動

  1. コマンドスタート(VExCPURUNTRG)で指定タスク(VExTASKAPP)の開始
  2. 入力スケジュール終了後の繰り返しスタート(VExREPTIME ≥ 2)

- 入力スケジュールの起動

  1. 出力スケジュール終了後の待機状態から起動トリガ(VExTRGMODE で選択したトリガ入力)による入力処理タスクの開始
  2. コマンドスタート(VExCPURUNTRG)で指定タスク(VExTASKAPP)の開始

表 17-6 スケジュール関連レジスタ

レジスタ	機能	
VExACTSCH	動作スケジュール選択	0x0: VExTASKAPP 設定タスクだけ実行 0x1: スケジュール 1 実行 0x2: スケジュール 2 実行 0x3: スケジュール 3 実行 0x4: スケジュール 4 実行 0x5: スケジュール 5 実行 0x6: スケジュール 6 実行 0x7: スケジュール 7 実行 0x8: スケジュール 8 実行 0x9: スケジュール 9 実行 0xA: スケジュール 10 実行 0xB: スケジュール 11 実行 0xC: スケジュール 12 実行 0xD: スケジュール 13 実行 0xE:スケジュール 14 実行 0xF: スケジュール 15 実行
VExTASKAPP	起動タスク指定	選択した動作スケジュールで実行可能なタスク番号を指定してください。
VExREPTTIME	スケジュール繰り返し回数	1 ~ 15 を設定してください。 注) 1 回実行の場合も"1"を設定してください。"0" ではスケジュール実行できません。
VExTRGMD	起動トリガモード選択	入力スケジュールトリガ選択 ADC ユニット A または ADC ユニット B 割り込み。

17.4.1.3　割り込み制御

ベクトルエンジンはスケジュール終了時に発生するベクトルエンジン割り込み（INTVCNx）と指定タスク終了時に発生するタスク終了割り込み（INTVCTx）があります。

- ベクトルエンジン割り込み

  1. 動作スケジュール選択(VExACTSCH)でスケジュール指定しコマンドスタート(VExCPURUNTRG＝"1")します。

  2. 繰り返し回数指定(VExREPTIME)した回数だけの選択スケジュールの実行終了時にINTVCNx 割り込み発生します。

  3. エラー検出割り込み制御を許可（VExERRINTEN<VERREN>＝"1"）していると出力スケジュール実行中に PMD 回路の PWM 割り込みが発生すると INTVCNx 割り込みが発生し、エラーフラグ(VExERRDET<VERRD>)に"1" を設定します。

- タスク終了割り込み

  1. タスク終了割り込み発生タスクを指定（VExTASKAPP<VITASK>）してタスク終了割り込み制御を許可（VExERRINTEN<INTTEN>＝"1"）します。

  2. コマンドスタート(VExCPURUNTRG＝"1")でスケジュール開始し、<VITASK>設定のタスク終了時に INTVCTx 割り込み発生します。

17.4.2 タスク概要

スケジュールで動作する各タスクの概要を示します。

個別実行タスクや起動タスクの指定には表 17-7 のタスク番号を使用します。

表 17-7 タスク一覧

タスク		タスク機能	タスク番号
出力スケジュール	電流制御	d 軸/q 軸 PI 制御(PI 制御出力制限可能) d 軸/q 軸合わせた非干渉制御,電圧スカラ制限	5
	SIN/COS 演算	正弦/余弦演算 位相補間(クリッピング機能あり)	6
	出力座標軸変換	逆パーク変換	7
	出力相変換 1	2 相から 3 相に変換[SVM]	8
	出力相変換 2	2 相から 3 相に変換[逆クラーク変換]	11
	出力制御 1	PMD 設定形式へのデータ変換 PWM シフト 1 切り替え PWM 出力制限 デッドタイム補償制御	0
	出力制御 2	PMD 設定形式へのデータ変換 PWM シフト 2 切り替え PWM 出力制限 デッドタイム補償制御	9
	トリガ生成	同期トリガタイミング生成	1
入力スケジュール	入力処理 1	センサ,3 シャントおよび PWM シフト禁止/PWM シフト 1 許可の 1 シャント ADC 変換結果読み込み、固定小数点数変換 ヒステリシス幅指定して電流極性判定可能	2
	入力処理 2	センサ,3 シャントおよび PWM シフト 2 許可の 1 シャント ADC 変換結果読み込み、固定小数点数変換 ヒステリシス幅指定して電流極性判定可能	10
	入力相変換	3 相から 2 相に変換	3
	入力座標軸変換	パーク変換 d 軸/q 軸の電流ベクトルまたは誘起電圧の偏角算出可能	4
ATAN 演算		逆正接を算出	12
SQRT 演算		平方根を算出	13

17.4.2.1　電流制御(タスク 5)

電流制御タスクは、d 軸電流、q 軸電流それぞれに PI 制御演算を行います。PI 制御出力を制限することができます。

また、p 軸と q 軸を合わせて制御する、非干渉制御および電圧スカラ制限が可能です。

1. d 軸電流 PI 制御

<演算式>

[PI 制御]
CIDKP = VExCIDKP × [VExCIDKG 設定]　:比例係数
CIDKI = VExCIDKI × [VExCIDKG 設定]　:積分係数
Δ ID = VExIDREF − VExID　:電流指令値と電流フィードバックの差分
VDI0 = CIDKI × Δ ID + VExVDI　:積分項演算
VD0 = CIDKP × Δ ID + VExVDI0　:比例項をあわせて電圧算出
[PI 制御出力制限]
if (VD0 > VExPIOLIM)　: 上限値
　VExVD = VExPIOLIM
　VExMCTLF<PIDOVF> = 1
else if (VD0 < −VExPIOLIM)　: 下限値
　VExVD = −VExPIOLIM
　VExMCTLF<PIDOVF> = 1
else VExVD = VD0
[アンチワインドアップ(AWU)]
Δ VD = VExVD − VD0　: 制限による差分
VExVDI = VDI0 + Δ VD × [VExMODE<AWUMD>設定]　: 制限による差分を積分項に反映

	レジスタ名	機能	
入力	VExID	d 軸電流	32bit 固定小数点データ(小数点以下 31bit)
	VExIDREF	d 軸電流基準値	16bit 固定小数点データ(小数点以下 15bit)
	VExCIDKP	比例係数	16bit データ
	VExCIDKI	積分係数	16bit データ
	VExCIDKG	d 軸 PI 制御係数 レンジ設定	000: 1 倍、 001: $1/2^4$ 倍、 010: $1/2^8$ 倍、 011: $1/2^{12}$ 倍、 100: $1/2^{16}$ 倍、 101 ~ 111: Reserved
	VExPIOLIM	PI 制御出力制限値	16bit 固定小数点データ(小数点以下 15bit) 有効範囲 0x0 ~ 0x7FFF 注) VExPIOLIM = 0 の場合は出力制限禁止
	VExMODE[9:8]	出力制限時の アンチワインドアップ比設定	<AWUMD> 00: 禁止、 01: 1/4、 10: 1/2、 11: 1
出力	VExVD	d 軸電圧	32bit 固定小数点データ(小数点以下 31bit)
	VExMCTLF[8]	d 軸出力制限状態	<PIDOVF> 0: 制限なし、 1: 制限中
内部	VExVDI	d 軸電圧積分項保持	64bit 固定小数点データ(小数点以下 63bit)

注) VExVDI は 64 ビットデータで、上位が VExVDIH レジスタ、下位が VExVDILH レジスタの構成です。

2. q 軸電流 PI 制御

<演算式>

[PI 制御]

CIQKP = VExCIQKP × [VExCIQKG 設定] :比例係数

CIQKI = VExCIQKI × [VExCIQKG 設定] :積分係数

Δ IQ = VExIQREF − VExIQ :電流指令値と電流フィードバックの差分

VQI0 = CIQKI × Δ IQ + VExVQI :積分項演算

VQ0 = CIQKP × Δ IQ + VQI0 :比例項をあわせて電圧算出

[PI 制御出力制限]

if (VQ0 > VExPIOLIM) : 上限確認

VExVQ = VExPIOLIM

VExMCTLF<PIQOVF> = 1

else if (VQ0 < −VExPIOLIM) : 下限確認

VExVQ = −VExPIOLIM

VExMCTLF<PIQOVF> = 1

else VExVQ = VQ0

[アンチワインドアップ(AWU)]

Δ VQ = VExVQ − VQ0 : 制限による差分

VExVQI = VQI0 + Δ VQ × [VExMODE<AWUMD>設定] : 制限による差分を積分項に反映

	レジスタ名	機能	
入力	VExIQ	q 軸電流	32bit 固定小数点データ(小数点以下 31bit)
	VExIQREF	q 軸電流基準値	16bit 固定小数点データ(小数点以下 15bit)
	VExCIQKP	比例係数	16bit データ
	VExCIQKI	積分係数	16bit データ
	VExCIQKG	q 軸 PI 制御係数レンジ設定	000: 1 倍、 001: $1/2^4$ 倍、 010: $1/2^8$ 倍、 011: $1/2^{12}$ 倍、 100: $1/2^{16}$ 倍、 101 ~ 111: Reserved
	VExPIOLIM	PI 制御出力制限値	16bit 固定小数点データ(小数点以下 15bit) 有効範囲 0x0 ~ 0x7FFF 注) VExPIOLIM = 0 の場合は出力制限禁止
	VExMODE[9:8]	出力制限時の アンチワインドアップ比設定	<AWUMD> 00: 禁止、 01: 1/4、 10: 1/2、 11: 1
出力	VExVQ	q 軸電圧	32bit 固定小数点データ(小数点以下 31bit)
	VExMCTLF[9]	q 軸出力制限状態	<PIQOVF> 0: 制限なし、 1: 制限中
内部	VExVQI	q 軸電圧積分項保持	64bit 固定小数点データ(小数点以下 63bit)

注) VExVQI は 64 ビットデータで、上位が VExVQIH レジスタ、下位が VExVQILH レジスタの構成です。

3. 非干渉制御

モータの電圧方程式を使って d 軸,q 軸の干渉分を算出して PI 制御結果を補正します。

<演算式>

```
if (VExMODE[10] = 1 )                                          : 拡張制御有効
    LD = VExCLD × [VExCLG 設定]                                : d 軸インダクタンス
    LQ = VExCLQ × [VExCLG 設定]                                : q 軸インダクタンス
    PHI = VExCPHI × [VExCPHIG 設定]                            : 鎖交磁束
    VExVDE = −VExOMEGA × VExIQ × LQ                            : d 軸干渉分算出
    VExVQE = VExOMEGA × VExID × LD + VExOMEGA × PHI            : q 軸干渉分算出
    if (VExMODE[11] = 1 )                                      : 非干渉制御有効
        VExVD = VExVD + VExVDE
        VExVQ = VExVQ + VExVQE
```

	レジスタ名	機能	
入力	VExVD	d 軸電圧	32bit 固定小数点データ(小数点以下 31bit)
	VExVQ	q 軸電圧	
	VExID	d 軸電流	
	VExIQ	q 軸電流	
	VExCLD	d 軸インダクタンス	16bit 固定小数点データ(小数点以下 11bit)
	VExCLQ	q 軸インダクタンス	
	VExCPHI	鎖交磁束	
	VExCLG	インダクタンスレンジ設定	000: 1 倍、 001: $1/2^4$ 倍、 010: $1/2^8$ 倍、 011: $1/2^{12}$ 倍、 100: $1/2^{16}$ 倍、 101 ~ 111: Reserved
	VExCPHIG	鎖交磁束レンジ設定	
	VExOMEGA	回転速度	16bit 固定小数点データ(小数点以下 15bit)
	VExMODE[11]	非干渉制御許可	<NICEN> 0: 非干渉制御禁止、 1: 非干渉制御許可
	VExMODE[10]	拡張制御許可	<T5ECEN> 0: 拡張制御禁止(非干渉制御不可) 1: 拡張制御許可
出力	VExVDE	d 軸非干渉補正電圧	16bit 固定小数点データ(小数点以下 15bit)
	VExVQE	q 軸非干渉補正電圧	
	VExVD	d 軸電圧	32bit 固定小数点データ(小数点以下 31bit)
	VExVQ	q 軸電圧	

4. 電圧スカラ制限

d 軸,q 軸電圧の合成値($VD^2 + VQ^2$ の平方根)が制限値を超えないように、d 軸,q 軸電圧を制限します。

<演算式>

```
if (VExMODE[10] = 1 )                                                    : 拡張制御有効
    [VDQ 演算]
    if (VExVD² + VExVQ² > VExVSLIM² )                                    :超過確認
        if (VExFMODE[11:10] = 00)        VExVDQ = SQRT(VD² + VQ² )       :電圧スカラ制限なし
        else if (VExFMODE[11:10] = 01)   VExVDQ = SQRT(VExVSLIM² − VQ² ) :d 軸方向スカラ制限
        else if (VExFMODE[11:10] = 10)   VExVDQ = SQRT(VExVSLIM² − VD² ) :q 軸方向スカラ制限
        else if (VExFMODE[11:10] = 11)   VExVDQ = SQRT(VD² + VQ² )       :dq 比例スカラ制限
        注) SQRT :平方根演算
    [偏角算出]
    X = | VExVQ |
    Y = | VExVD |
    VExDELTA = ATAN2(X, Y)
        注) ATAN2 : 逆正接演算
    [各軸制限値算出]
    if (VExFMODE[11:10] = 00)                                            :電圧スカラ制限なし
        VDLIM = VExVSLIM
        VQLIM = VExVSLIM
    else if (VExFMODE[11:10] = 01)                                       :d 軸方向スカラ制限
        VDLIM = VExVDQ
        VQLIM = VExVSLIM
    else if (VExFMODE[11:10] = 10)                                       :q 軸方向スカラ制限
        VDLIM = VExVSLIM
        VQLIM = VExVDQ
    else if (VExFMODE[11:10] = 11)                                       :dq 比例スカラ制限
        VDLIM = VExVSLIM × SIN(VExDELTA)
        VQLIM = VExVSLIM × COS(VExDELTA)
    [制限処理]
    if (VExVD > VDLIM)               VExVD = VDLIM                       : d 軸上限処理
                                     VExMCTLF[10] = 1
    else if (VExVD < −VDLIM)         VExVD = −VDLIM                      : d 軸下限処理
                                     VExMCTLF[10] = 1
    if (VExVQ > VQLIM)               VExVQ = VQLIM                       : q 軸上限処理
                                     VExMCTLF[10] = 1
    else if (VExVQ < −VQLIM)         VExVQ = −VQLIM                      : q 軸下限処理
```

VExMCTLF[10] = 1

	レジスタ名	機能	
入力	VExVD	d 軸電圧	32bit 固定小数点データ(小数点以下 31bit)
	VExVQ	q 軸電圧	
	VExVSLIM	電圧スカラ制限値	16bit 固定小数点データ(小数点以下 15bit) 0x0 ~ 0x7FFF 注）VExVSLIM = 0x0 の場合は制限禁止
	VExMODE[10]	拡張制御許可	<T5ECEN> 0: 拡張制御禁止(スカラ制限不可) 1: 拡張制御許可
	VExFMODE[11:10]	制限モード設定	<VSLIMMD> 00: スカラ制限禁止（軸別制限有効） 01: スカラ制限許可、d 軸方向に制限 10: スカラ制限許可、q 軸方向に制限 11: スカラ制限許可、dq 比例制限
出力	VExVDQ	電圧スカラ値 または 軸方向制限値	16bit 固定小数点データ(小数点以下 15bit)
	VExDELTA	偏角	16bit データ 0x0000 ~ 0x4000 (0 ~ 90°)
	VExVD	d 軸電圧	32bit 固定小数点データ(小数点以下 31bit)
	VExVQ	q 軸電圧	
	VExMCTLF[10]	電圧スカラ制御超過フラグ	<VSOVF> 0:超過なし 1:超過

17.4.2.2 SIN/COS 演算(タスク 6)

SIN/COS 演算タスクは、位相補間演算と SIN/COS 演算を実行します。

位相補間は回転速度を PWM 周期で積分して計算し、位相補間許可時(VExMODE<PVIEN> = "1") のみ実行します。位相補間時は指定位相でのクリッピングが可能です。

1. 位相補間

<演算式>

```
THETA0 = VExOMEGA × VExTPWM + VExTHETA                          :位相補間値算出
THETA0 = THETA0 & 0x0000FFFF
if (VExMODE[7] = 1)                                             :クリッピング許可
    if ( VExOMEGA ≥ 0)                                          : 正回転時
        if ( VExTHETA ≤ VExTHTCLP ≤ THETA0 )      THETA0 = THTCLP
        else if ( THETA0 ≤ VExTHETA ≤ VExTHTCLP ) THETA0 = THTCLP
        else if ( VExTHTCLP ≤ THETA0 ≤ VExTHETA ) THETA0 = THTCLP
    else if ( VExOMEGA < 0)                                     :逆回転時
        if ( THETA0 ≤ VExTHTCLP ≤ VExTHETA )      THETA0 = THTCLP
        else if ( VExTHTCLP ≤ VExTHETA ≤ THETA0 ) THETA0 = THTCLP
        else if ( VExTHETA ≤ THETA0 ≤ VExTHTCLP ) THETA0 = THTCLP
if ( VExMODE[0] = 1 ) VExTHETA = THETA0                         :位相補間許可で THETA 更新
```

	レジスタ名	機能	
入力	VExTHETA	位相 θ	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 16bit)
	VExOMEGA	回転速度	16bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 15bit)
	VExTPWM	PWM 周期レート	16bit データ
	VExTHTCLP	クリップ値	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 16bit)
	VExMODE[0]	位相補間許可	<PVIEN> 0: 位相補間禁止、 1: 位相補間許可
	VExMODE[7]	位相クリッピング制御	<CLPEN> 0: クリップ禁止、 1: クリップ許可
出力	VExTHETA	位相 θ	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 16bit)

2. SIN/COS 演算

<演算式>

```
VExSINM = VExSIN                                  :前回値保存(入力処理用)
VExCOSM = VExCOS
VExSIN = SIN ( VExTHETA )                         : SIN,COS 値算出
VExCOS = SIN ( VExTHETA + 1/4 )
if ( VExFMODE[9] = 1)                             :前回値保持無効確認
    VExSINM = VExSIN
    VExCOSM = VExCOS
```

注) SIN : 正弦演算

	レジスタ名	機能	
入力	VExTHETA	位相 θ	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 16bit)
	VExFMODE[9]	SIN,COS 前回値保持選択	<MREGDIS> 0: 前回値保持有効, 1: 前回値保持無効
出力	VExSIN	θ での正弦値	16bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 15bit)
	VExCOS	θ での余弦値	
	VExSINM	前回の正弦値	
	VExCOSM	前回の余弦値	

17.4.2.3 出力電圧変換(座標軸変換/相変換)

出力電圧変換は座標軸変換と相変換の 2 段階で変換します。

相変換タスクには空間ベクトル変換と逆クラークの 2 種類あります。

1. 出力座標軸変換(タスク 7)

   出力座標軸タスクは d 軸電圧、q 軸電圧、sinθ、cosθ から α 軸電圧、β 軸電圧を算出します。

   <演算式>

   VExTMPREG3 = VExCOS × VExVD − VExSIN × VExVQ　　:Vα 算出

   VExTMPREG4 = VExSIN × VExVD + VExCOS × VExVQ　　:Vβ 算出

	レジスタ名	機能	
入力	VExVD	d 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExVQ	q 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExSIN	THETA での正弦値	16bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 15bit)
	VExCOS	THETA での余弦値	16bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 15bit)
出力	VExTMPREG3	α 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG4	β 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)

2. 出力相変換 1(空間ベクトル変換)(タスク 8)

   出力相変換 1 は α 軸電圧、β 軸電圧からセクタ判定し、セクタ別に空間ベクトル変換で a 相電圧、b 相電圧、c 相電圧の Duty を算出します。このタスクは変換方式に 2 相変調か 3 相変調かを選択できます。

a.　セクタ判定

<演算式>

```
VExSECTORM = VExSECTOR                                  : 前回セクタを保存
Vα = VExTMPREG3
Vβ = VExTMPREG4
if ( Vα ≥ 0 & Vβ ≥ 0 )
    if ( |Vα| ≥ |Vβ| ÷ sqrt(3) )
        if ( |Vα| ÷ sqrt(3) ≥ |Vβ| )      VExSECTOR = 0
        else                              VExSECTOR = 1
    else                                  VExSECTOR = 2
else if ( Vα < 0 & Vβ ≥ 0 )
    if ( |Vα| < |Vβ| ÷ sqrt(3) )          VExSECTOR = 3
    if ( |Vα| ÷ sqrt(3) < |Vβ| )          VExSECTOR = 4
    else                                  VExSECTOR = 5
else if ( Vα < 0 & Vβ < 0 )
    if ( |Vα| ≥ |Vβ| ÷ sqrt(3) )
        if ( |Vα| ÷ sqrt(3) ≥ |Vβ| )      VExSECTOR = 6
        else                              VExSECTOR = 7
    else                                  VExSECTOR = 8
else if ( Vα ≥ 0 & Vβ < 0 )
    if ( |Vα| < |Vβ| ÷ sqrt(3) )          VExSECTOR = 9
    else if ( |Vα| ÷ sqrt(3) < |Vβ| )     VExSECTOR = 10
    else                                  VExSECTOR = 11
if ( VExFMODE[9] = 1) VExSECTORM = VExSECTOR            :前回値保持無効確認
```

	レジスタ名	機能	
入力	VExTMPREG3	α 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG4	β 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExFMODE[9]	SECTOR 前回値保持選択	<MREGDIS> 0: 前回値保持有効, 1: 前回値保持無効
出力	VExSECTOR	セクタ	4bit データ
	VExSECTORM	前回のセクタ	4bit データ

b. 空間ベクトル変換(3 相変調で<SECTOR[3:0]> = "0", "1"の場合のみ記載)

<演算式>

```
if (SECTOR = 0,1)
    t1 = sqrt(3) ÷ VExVDC × (sqrt(3) ÷ 2 × Vα − 1 ÷ 2 × Vβ)      : t1 期間算出
    t2 = sqrt(3) ÷ VExVDC × Vβ)                                   : t2 期間算出
    t3 = 1 − t1 − t2                                              : ゼロベクトル期間算出
    if ( VExFMODE[0] = 0 )                                        : 3 相変調
        DUTYA = t1 + t2 + t3 ÷ 2
        DUTYB = t2 + t3 ÷ 2
        DUTYC = t3 ÷ 2
    else                                                          : 2 相変調
        DUTYA = t1 + t2
        DUTYB = t2
        DUTYC = 0

VExTMPREG0 = DUTYA
VExTMPREG1 = DUTYB
VExTMPREG2 = DUTYC
VExTMPREG5 = t3
```

	レジスタ名	機能	
入力	VExTMPREG3	α 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG4	β 軸電圧	
	VExVDC	電源電圧	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExSECTOR	セクタ	4bit データ
	VExFMODE[0]	変調モード	<C2PEN> 0: 3 相変調, 1: 2 相変調
出力	VExTMPREG0	a 相電圧 Duty	32bit 固定小数点データ(0..0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電圧 Duty	
	VExTMPREG2	c 相電圧 Duty	
	VExTMPREG5	ゼロベクトル Duty	

3. 出力相変換 2(逆クラーク変換)(タスク 11)

出力相変換 2 は α 軸電圧、β 軸電圧からセクタ判定し、逆クラーク変換で a 相電圧、b 相電圧、c 相電圧のデューティを算出するタスクです。このタスクは変換方式として 3 相変調だけに対応します。

また、このタスクは VExFMODE<PIGSEL> を"1"に設定することで 2 相電圧の Duty を算出します。

a. セクタ判定

<演算式>

```
VExSECTORM = VExSECTOR                                  : 前回セクタを保存
Vα = VExTMPREG3
Vβ = VExTMPREG4
if ( Vα ≥ 0 & Vβ ≥ 0 )
    if ( Vα| ≥ |Vβ| ÷ sqrt(3))
        if ( |Vα| ÷ sqrt(3) ≥ |Vβ| )       VExSECTOR = 0
        else                               VExSECTOR = 1
    else                                   VExSECTOR = 2
else if ( Vα < 0 & Vβ ≥ 0 )
    if ( |Vα| < |Vβ| ÷ sqrt(3) )           VExSECTOR = 3
    if ( |Vα| ÷ sqr(3) < |Vβ| )            VExSECTOR = 4
    else                                   VExSECTOR = 5
else if ( Vα < 0 & Vβ < 0 )
    if ( |Vα| ≥ |Vβ| ÷ sqrt(3) )
        if ( |Vα| ÷ sqrt(3) ≥ |Vβ| )       VExSECTOR = 6
        else                               VExSECTOR = 7
    else                                   VExSECTOR = 8
else if ( Vα ≥ 0 & Vβ < 0 )
    if ( |Vα| < |Vβ| ÷ sqrt(3) )           VExSECTOR = 9
    else if ( |Vα| ÷ sqrt(3) < |Vβ| )      VExSECTOR = 10
    else                                   VExSECTOR = 11
if ( VExFMODE[9] = 1) VExSECTORM = VExSECTOR            :前回値保持無効確認
```

	レジスタ名	機能	
入力	VExTMPREG3	α 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG4	β 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExFMODE[9]	SECTOR 前回値保持選択	<MREGDIS> 0: 前回値保持有効, 1: 前回値保持無効
出力	VExSECTOR	セクタ	4bit データ
	VExSECTORM	前回のセクタ	4bit データ

b. 逆クラーク変換

<演算式>

```
if ( VExFMODE[12] = 0 )                                                    : 3 相変換
    VExTMPREG0 = 1 ÷ VExVDC × Vα + 1/2                                     : Va Duty
    VExTMPREG1 = 1 ÷ VExVDC × ( −1 ÷ 2 × Vα + sqrt(3) ÷ 2 × Vβ) + 1/2      : Vb Duty
    VExTMPREG2 = 1 ÷ VExVDC × ( −1 ÷ 2 × Vα − sqrt(3) ÷ 2 × Vβ) + 1/2      : Vc Duty
else                                                                       : 相変換禁止
    VExTMPREG0 = 1 ÷ VExVDC × Vα + 1/2                                     : Va Duty
    VExTMPREG1 = 1 ÷ VExVDC × Vβ + 1/2                                     : Vb Duty
```

	レジスタ名	機能	
入力	VExTMPREG3	α 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG4	β 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExVDC	電源電圧	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExFMODE[12]	相変換禁止設定	<PHCVDIS> 0: 相変換許可, 1: 相変換禁止
出力	VExTMPREG0	a 相電圧 Duty	32bit 固定小数点データ(0..0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電圧 Duty	32bit 固定小数点データ(0..0 ~ 1.0、小数点以下 31bit)
	VExTMPREG2	c 相電圧 Duty	32bit 固定小数点データ(0..0 ~ 1.0、小数点以下 31bit)

17.4.2.4　出力制御

出力制御部は、3 相電圧デューティを PMD 設定形式に変換します。変換した結果を VExCMPU、VExCMPV、VExCMPW に設定し、出力制御動作設定に応じて VExOUTCR を設定します。また、デッドタイム補償制御および PWM 出力制限が可能です。

出力制御には出力制御 1 タスクと出力制御 2 タスクの 2 種類があり、それぞれ対応できる PWM 出力が違います。

1. 出力制御 1(タスク 0)

出力制御 1 タスクは通常 PWM 出力と PWM シフト 1 モードの PWM 出力に対応しています。

PWM シフト許可時に、回転速度(VExOMEGA)が PWM シフト切り替え基準(VExFPWMCHG)より小さい場合に PWM 出力が PWM シフト 1 に切り替わります。

注)PWM シフト 1 は 1 シャント電流検出モード時のみ選択できます。

- 出力変換

<演算式>

```
VExMCTLF[1]= VExMCTLF[0]                                  : 前回値フラグ更新
VExMCTLF[0]= 0                                            : 現フラグクリア
if ((FMODE[3]= 1) & (FMODE[0]= 1) & (FMODE[1]= 1))        : 1 シャント,2 相変調でシフト 1 許可
    if (|VExOMEGA| < VExFPWMCHG)    VExMCTLF[0] = 1       : 低速判定で低速フラグセット
DUTYA = VExTMPREG0
DUTYB = VExTMPREG1
DUTYC = VExTMPREG2
if (VExMCTLF[0] = 1)                                      : PWM シフト 1 で低速時
    if (VExSECTOR = 0,3,4,7,8,11)                         : セクタ判定
        DUTYA = DUTYA + VExTMPREG5                        : ゼロベクトル V7 変換
        DUTYB = DUTYB + VExTMPREG5
        DUTYC = DUTYC + VExTMPREG5
PWMA= VExTMPREG0 × VExMDPRD                               : PMD 設定値変換
PWMB= VExTMPREG1 × VExMDPRD
PWMC= VExTMPREG2 × VExMDPRD
```

	レジスタ名	機能	
入力	VExTMPREG0	a 相電圧	32bit 固定小数点データ(0.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電圧	32bit 固定小数点データ(0.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG2	c 相電圧	32bit 固定小数点データ(0.0 ~ 1.0、小数点以下 31bit)
	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
	VExSECTOR	セクタ	4bit データ
	VExOMEGA	回転速度	16bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 15bit)
	VExFPWMCHG	PWM シフト切り替え基準	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExFMODE[0]	変調モード	<C2PEN> 0: 3 相変調, 1: 2 相変調
	VExFMODE[1]	PWM シフト許可	<SPWMEN> 0: シフト禁止, 1: シフト許可
	VExFMODE[3:2]	電流検出モード	<IDMODE[1:0]> 00:3 シャント. 01: 2 センサ, 1x :1 シャント
出力	VExMCTLF[1:0]	低速度フラグ	<LAVFM>,<LAVF>

- PWM 出力制限

<演算式>

```
if (VExPWMMAX = 0)                          MAX = VExMDPRD
else                                        MAX = VExPWMMAX
if ((PWMA > MAX) & (VExMDPRD > MAX))                          : U 相 PWM 上限確認
    if ((VExMODE[13] = 0) | (PWMA < VExMDPRD))                : 100%出力制限確認
        PWMA = MAX
        VExMCTLF[11] = 1
MIN = VExPWMMIN
if ((PWMA < MIN) & (MIN > 0))                                 : U 相 PWM 下限確認
    if ((VExMODE[12] = 0) | (PWMA > 0))                       : 0%出力制限確認
        PWMA = MIN
        VExMCTLF[11] = 1
( 残りの 2 相も同様に演算)
```

	レジスタ名	機能	
入力	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
	VExPWMMAX	PWM 上限値設定	16bit データ(0 ~ VExMDPRD 設定値)
	VExPWMMIN	PWM 下限値設定	16bit データ(0 ~ VExMDPRD 設定値)
	VExMODE[12]	PWM 制限時の 0%出力許可	<PWMBLEN> 0: 禁止, 1: 許可
	VExMODE[13]	PWM 制限時の 100%出力許可	<PWMFLEN> 0: 禁止, 1: 許可
出力	VExMCTLF[11]	PWM 出力制限超過フラグ	<PWMOVF>

- デッドタイム補償

<演算式>

```
if (0 < PWMA < VExMDPRD)                    DT = VExDTC
else                                        DT = 0
if (VExDTCS<IASTS> = 01)                                                : 正電流時
    if (VExMODE[14] = 1)                                                : PMD のデッドタイム補正対応
        if (PWMA > (VExMDPRD − 2 × DT))     PWMA = (VExMDPRD + PWMA) ÷2
    else                                    PWMA = PWMA + DT
    if ((VExMODE[13] = 1) & (PWMMAX < VExMDPRD))                        : 100%出力制限確認
        if (PWMA > (VExMDPRD −1))           PWMA = VExMDPRD −1          : 補償後の出力制限
    else
        if (PWMA > VExMDPRD)                PWMA = VExMDPRD             : 補償後の出力制限
else if (VExDTCS<IASTS> = 11)                                           : 負電流時
    if (VExMODE[14] = 1)                                                : PMD のデッドタイム補正対応
        if (PWMA < (2 × DT))                PWMA = PWMA ÷2
    else                                    PWMA = PWMA − DT
    if ((VExMODE[12] = 1) & (PWMMIN > 0))                               : 0%出力制限確認
        if (PWMA < 1)                       PWMA = 1                    : 補償後の出力制限
    else
        if (PWMA < 0)                       PWMA = 0                    : 補償後の出力制限
( 残りの 2 相も同様に演算)
```

	レジスタ名	機能	
入力	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
	VExDTC	デッドタイム補償量	16bit データ(0 ~ VExMDPRD 設定値)
	VExMODE[12]	制限時の 0%出力許可	<PWMBLEN>
	VExMODE[13]	制限時の 100%出力許可	<PWMFLEN>
	VExMODE[14]	PMD 回路のデッドタイム補正対応制御	<PMDDTCEN> 0: PMD のデッドタイム補正禁止時に設定 1: PMD のデッドタイム補正許可時に設定
	VExDCTS	デッドタイム補償制御/ステータス	<ICSTS>, <IBSTS>, <IASTS> x0: 未確定, 01:正電流, 11: 負電流

2. 出力制御/PWM シフト 1 変換

<演算式>

```
OUTCR = 0x1FF                                                    : 全相センタオン
if (VExMCTLF[0] = 1)                                             : PWM シフト 1 で低速時
    if (VExSECTOR = 0,1,2,11)        PWMB = MDPRD − PWMB         : V 相センタオフ
                                     OUTCR = 0x1F3
    else if (VExSECTOR = 3,4,5,6)    PWMC = MDPRD − PWMC         : W 相センタオフ
                                     OUTCR = 0x1CF
    else if (VExSECTOR = 7,8,9,10)   PWMA = MDPRD − PWMA         : U 相センタオフ
                                     OUTCR = 0x1FC
if (VExMODE[3:2] = 00,11)            OUTCR = 0x000               : 出力オフ
else if (VExMODE[3:2] = 10)          OUTCR = 0x015               : 短絡ブレーキ
VExCMPU = PWMA
VExCMPV = PWMB
VExCMPW = PWMC
VExOUTCR = OUTCR
```

	レジスタ名	機能	
入力	VExSECTOR	セクタ	4bit データ
	VExMODE[3:2]	出力制御動作	<OCRMD[1:0]>
	VExMINPLS	最小パルス幅差	16bit データ
出力	VExCMPU	PMD U 相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExCMPV	PMD V 相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExCMPW	PMD W相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExOUTCR	PMD 出力制御設定	9bit 設定
	VExEMGRS	PMD EMG 復帰	1bit 設定
	VExMCTLF	パルス微小フラグ	<PLSLF>

3. 出力制御 2(タスク 9)

出力制御 2 タスクは通常 PWM 出力と PWM シフト 2 モードの PWM 出力に対応しています。

PWM シフト許可(VExFMODE<SPWMEN> = "1")かつ PWM シフトモード選択(VExFMODE<SPWMMD>)を "00"以外に設定することで PWM シフト 2 の PWM 出力になります。

注)PWM シフトは 1 シャント電流検出モード時のみ選択できます。

- 出力変換

<演算式>

```
VExMCTLF[1] = VExMCTLF[0]                          : 前回値フラグ更新
VExMCTLF[0] = 0                                    : 現フラグクリア
PWMA = VExTMPREG0 × VExMDPRD                       : PMD 設定値変換
PWMB = VExTMPREG1 × VExMDPRD
PWMC = VExTMPREG2 × VExMDPRD
```

	レジスタ名	機能	
入力	VExTMPREG0	a 相電圧	32bit 固定小数点データ(0.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電圧	32bit 固定小数点データ(0.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG2	c 相電圧	32bit 固定小数点データ(0.0 ~ 1.0、小数点以下 31bit)
	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
出力	VExMCTLF[1:0]	低速度フラグ	<LAVFM>,<LAVF>

- PWM 出力制限

<演算式>

```
if (VExPWMMAX = 0)                    MAX = VExMDPRD
else                                  MAX = VExPWMMAX
if ((PWMA > MAX) & (VExMDPRD > MAX))                  : U 相 PWM 上限確認
    if ((VExMODE[13] = 1) | (PWMA < VExMDPRD))        : 100%出力制限確認
        PWMA = MAX
        VExMCTLF[11] = 1
MIN = VExPWMMIN
if ((PWMA < MIN) & (MIN > 0))                         : U 相 PWM 下限確認
    if ((VExMODE[12] = 1) | (PWMA > 0))               : 0%出力制限確認
        PWMA = MIN
        VExMCTLF[11] = 1
( 残りの 2 相も同様に演算)
```

	レジスタ名	機能	
入力	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
	VExPWMMAX	PWM 上限値設定	16bit データ(0 ~ VExMDPRD 設定値)
	VExPWMMIN	PWM 下限値設定	16bit データ(0 ~ VExMDPRD 設定値)
	VExMODE	出力制御動作	<OCRMD[1:0]>
	VExFMODE[12]	PWM 制限時の 0%出力許可	<PWMBLEN> 0: 禁止, 1: 許可
	VExFMODE[13]	PWM 制限時の 100%出力許可	<PWMFLEN> 0: 禁止, 1: 許可
出力	VExMCTLF[11]	PWM 出力制限超過フラグ	<PWMOVF>

- デッドタイム補償

<演算式>

```
if (0 < PWMA < VExMDPRD)                          DT = VExDTC
else                                              DT = 0
if (VExDTCS<IASTS> = 01)                                                  : 正電流時
    if (VExMODE[14] = 1)                                                  : PMD のデッドタイム補正対応
        if (PWMA > (VExMDPRD − 2 × DT))           PWMA = (VExMDPRD + PWMA) ÷2
    else                                          PWMA = PWMA + DT
    if ((VExMODE[13] = 1) & (PWMMAX < VExMDPRD))                          : 100%出力制限確認
        if (PWMA > (VExMDPRD −1))                 PWMA = VExMDPRD −1      : 補償後の出力制限
    else
        if (PWMA > VExMDPRD)                      PWMA = VExMDPRD         : 補償後の出力制限
else if (VExDTCS<IASTS> = 11)                                             : 負電流時
    if (VExMODE[14] = 1)                                                  : PMD のデッドタイム補正対応
        if (PWMA < (2 × DT))                      PWMA = PWMA ÷2
    else                                          PWMA = PWMA − DT
    if ((VExMODE[12] = 1) & (PWMMIN > 0))                                 : 0%出力制限確認
        if (PWMA < 1)                             PWMA = 1                : 補償後の出力制限
    else
        if (PWMA < 0)                             PWMA = 0                : 補償後の出力制限
( 残りの 2 相も同様に演算)
```

	レジスタ名	機能	
入力	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
	VExDTC	デッドタイム補償量	16bit データ(0 ~ VExMDPRD 設定値)
	VExMODE[12]	制限時の 0%出力許可	<PWMBLEN]>
	VExMODE[13]	制限時の 100%出力許可	<PWMFLEN]>
	VExMODE[14]	PMD 回路のデッドタイム補正対応制御	<PMDDTCEN> 0: PMD のデッドタイム補正禁止時に設定 1: PMD のデッドタイム補正許可時に設定
	VExDCTS	デッドタイム補償制御/ステータス	<ICSTS>, <IBSTS>, <IASTS> x0: 未確定, 01:正電流, 11: 負電流

- 出力制御/PWM シフト 2 変換

<演算式>

```
OUTCR = 0x1FF                                                    : 全相センタオン
if ((VExFMODE[3] = 1) & (VExFMODE[1] = 1))                       : 1 シャントでシフト 2 許可
    if (VExFMODE[15:14] = 01)                                    : シフト 2,U 相基準
        if (PWMB > VExMDPRD ÷2)   PWMB = VExMDPRD − PWMB         : V 相 PWM センタオフ
                                  OUTCR = OUTCR & 0x1F3
        if (PWMC > VExMDPRD ÷2)   PWMC = VExMDPRD − PWMC         : W 相 PWM センタオフ
                                  OUTCR = OUTCR & 0x1CF
    else if (VExFMODE[15:14] = 10)                               : シフト 2,V 相基準
        if (PWMA > VExMDPRD ÷2)   PWMA = VExMDPRD − PWMA         : U 相 PWM センタオフ
                                  OUTCR = OUTCR & 0x1FC
        if (PWMC > VExMDPRD ÷2)   PWMC = VExMDPRD − PWMC         : W 相 PWM センタオフ
                                  OUTCR = OUTCR & 0x1CF
    else if (VExFMODE[15:14] = 11)                               : シフト 2,W 相基準
        if (PWMA > VExMDPRD ÷2)   PWMA = VExMDPRD − PWMA         : U 相 PWM センタオフ
                                  OUTCR = OUTCR & 0x1FC
        if (PWMB > VExMDPRD ÷2)   PWMB = VExMDPRD − PWMB         : V 相 PWM センタオフ
                                  OUTCR = OUTCR & 0x1F3
if (VExMODE[3:2] = 00,11)         OUTCR = 0x000                  : 出力オフ
else if (VExMODE[3:2] = 10)       OUTCR = 0x015                  : 短絡ブレーキ
VExCMPU = PWMA
VExCMPV = PWMB
VExCMPW = PWMC
VExOUTCR = OUTCR
```

	レジスタ名	機能	
入力	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
	VExFPWMCHG	PWM レベルシフト	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExMODE[3:2]	出力制御動作	<OCRMD>
	VExMINPLS	最小パルス幅	16bit データ
	VExFMODE[1]	PWM シフト許可	<SPWMEN>
	VExFMODE[3:2]	電流検出モード	<IDMODE>
	VExFMODE[15:14]	PWM シフトモード	<SPWMMD>
出力	VExCMPU	PMD U 相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExCMPV	PMD V 相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExCMPW	PMD W相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExOUTCR	PMD 出力制御設定	9bit 設定
	VExEMGRS	PMD EMG 復帰	1bit 設定
	VExMCTLF	パルス微小フラグ	<PLSLF>

17.4.2.5 トリガ生成(タスク 1)

トリガ生成部は、1 シャント電流検出時に PWM 設定値 VExCMPU、VExCMPV、VExCMPW から電流検出方式に応じたトリガタイミングを算出して VExTRGCMP0、VExTRGCMP1 に設定します。

注) 1 シャント電流検出以外では VExTRGCMP0、VExTRGCMP1 は更新されません。

注) PWM シフト 2 モード選択時は VExTRGCMP0、VExTRGCMP1 は更新されません。

	レジスタ名	機能	
入力	VExCMPU	PMD U 相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExCMPV	PMD V 相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExCMPW	PMD W 相 PWM 設定	16bit データ(0 ~ MDPRD 値)
	VExMDPRD	PWM 周期設定	16bit データ(PMD PWM 周期設定値)
	VExTADC	AD 変換時間	16bit データ(0 ~ MDPRD 値)
	VExTRGCRC	トリガ補正値	16bit データ(0 ~ MDPRD 値)
	VExSECTOR	セクタ	4bit データ
	VExMODE[0	零電流検出	<ZIEN>
	VExMODE[3:2	出力制御動作	<OCRMD>
	VExFMODE[0]	変調モード	<C2PEN> 0: 3 相変調, 1: 2 相変調
	VExFMODE[1]	PWM シフト許可	<SPWMEN>
	VExFMODE[3:2]	電流検出モード	<IDMODE>
	VExFMODE[8]	トリガ補正許可	<CRCEN>
	VExFMODE[15:14]	PWM シフトモード	<SPWMMD>
	VExMCTLF[0]	低速度フラグ	<LAVF>
出力	VExTRGCMP0	PMD トリガ 0 タイミング設定	16bit データ(0 ~ MDPRD 値)
	VExTRGCMP1	PMD トリガ 1 タイミング設定	16bit データ(0 ~ MDPRD 値)
	VExTRGSEL	PMD トリガ選択	3bit データ

17.4.2.6　入力処理

入力処理では A/D 変換器から変換結果と相情報を読み込みます。電流検出方式や PWM シフトモードなどの設定に応じて 3 相電流、電圧の変換結果を固定小数点データ変換して保存します。また、ゼロ電流検出モード時は電流検出結果をゼロ電流レジスタに保存します。

デッドタイム補償制御のために、ヒステリシス幅を指定して電流極性を判定できます。

入力処理には入力処理 1 タスクと入力処理 2 タスクの 2 種類があり、それぞれ対応できる電流検出方式が違います。

1. 入力処理 1(タスク 2)

入力処理 1 タスクは 3 シャント(2 相検出のみ(注 1))および 1 シャントの電流検出に対応します。ただし、PWM シフト 2 モードの 1 シャントには対応しません(注 2)。

(注 1)電流検出結果は 2 相のみ使用します。残りの 1 相は計算で求めます。

(注 2)PWM シフトは 1 シャント電流検出モード時のみ選択できます。

- 入力変換

<演算式>

```
[VDC 固定小数点数変換/保存]
if (VExMODE[4] = 0                          VExVDC = [DC 電圧] >>1
else                                        VExVDCL = [DC 電圧] >>1
[電流 1t 読み込み]
if (VExFMODE[3:2] = 10,11                                          : 1 シャント
    if (VExMCTLF[1] = 0                                            : 通常 PWM
        if (VExSECTORM = 4,5,6,7)           VExIAADC = [電流 1]
        else if (VExSECTORM = 8,9,10,11)    VExIBADC = [電流 1]
        else f (VExSECTORM = 0,1,2,3)       VExICADC = [電流 1]
    else if (VExMCTLF[1] = 1                                       : PWM シフト 1
        if (VExSECTORM = 1,2,7,8)           VExIAADC = [電流 1]
        else if (VExSECTORM = 0,5,6,11)     VExIBADC = [電流 1]
        else f (VExSECTORM = 3,4,9,10)      VExICADC = [電流 1]
else if (VExFMODE[3:2] = 00,01                                     : 3 シャント、2 センサ
    if ([電流 1 相情報] = 1)                VExIAADC = [電流 1]
    else if ([電流 1 相情報] = 2)           VExIBADC = [電流 1]
    else if ([電流 1 相情報] = 3)           VExICADC = [電流 1]
[電流 2 読み込み]
if (VExFMODE[3:2] = 10,11                                          : 1 シャント
    if (VExMCTLF[1] = 0                                            : 通常 PWM
        if (VExSECTORM = 0,1,10,11)         VExIAADC = [電流 2]
        else if (VExSECTORM = 2,3,4,5)      VExIBADC = [電流 2]
        else f (VExSECTORM = 6,7,8,9)       VExICADC = [電流 2]
    else if (VExMCTLF[1] = 1                                       : PWM シフト 1
        if (VExSECTORM = 3,4,9,10)          VExIAADC = [電流 2]
        else if (VExSECTORM = 1,2,7,8)      VExIBADC = [電流 2]
        else f (VExSECTORM = 0,5,6,11)      VExICADC = [電流 2]
else if (VExFMODE[3:2] = 00,01                                     : 3 シャント、2 センサ
```

```
    N = [電流 2 相情報]                                          : 1ADC で順次検出
    X = [電流 2]
    if (VExFMODE[13] =  1                  N = [電流 4 相情報]    : 同時サンプリング時は電流 4
                                           X = [電流 4]
        if ( N = 1)                        VExIAADC = X
        else if ( N = 2)                   VExIBADC = X
        else if ( N = 3)                   VExICADC = X]
[電流 3 読み込み]
if (VExFMODE[3] ≠ 1)                                             : 1 シャント以外
    if ( [電流 3 相情報] = 1)              VExIAADC = [電流 3]
    else if ( [電流 3 相情報] = 2)         VExIBADC = [電流 3]
    else if ( [電流 3 相情報] = 3)         VExICADC = [電流 3]
[電流固定小数点数変換]
IA = VExIAO − VExIAADC
IB = VExIBO − VExIBADC
IC = VExICO − VExICADC
if (VExFMODE[3:2] = 10,11                                        : 1 シャント
    if (VExMCTLF[1] = 0                                          : 通常 PWM
        if (VExSECTORM = 0,1,10,11)    IA = −IA
        else if (VExSECTORM = 6,7,8,9) IB = −IB
        else f (VExSECTORM = 2,3,4,5)  IC = −IC
    else if (VExMCTLF[1] = 1                                     : PWM シフト 1
        if (VExSECTORM = 1,2,5,6,9,10) IA = −IA
                                       IB = −IB
                                       IC = −IC
[電流 3 算出]
N = 6 − N − [電流 1 相情報]                                      : 電流 3 の相番号算出
if ( N = 1)                            IA = − IB − IC
else if ( N = 2)                       IB = − IC − IA
else if ( N = 3)                       IC = − IA − IB]
[電流保存]
VExTMPREG0 = IA
VExTMPREG1 = IB
VExTMPREG2 = IC
```

	レジスタ名	機能	
入力	(VDC1)	ADC 変換結果入力	16bit データ(上位 12bit が変換結果)
	(電流 1)		
	(電流 2)		
	(電流 3)		
	(電流 4)		
	VExSECTORM	セクタ情報	4bit データ
	VExMODE[1]	ゼロ電流検出	<ZIEN>
	VExMODE[4]	VDC 保存レジスタ	<VDCSEL>
	VExFMODE[3:2]	電流検出モード	<IDMODE>
	VExFMODE[13]	同時サンプリング制御	<SADCEN> 0: 同時サンプリング禁止, 1: 同時サンプリング許可
	VExMCTLF[0]	低速度フラグ	<LAVFM>

	レジスタ名	機能	
出力	VExVDC	DC 電源電圧	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExVDCL	DC 電源電圧	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExIAADC	a 相電流変換結果	16bit データ(上位 12bit に結果保持)
	VExIBADC	b 相電流変換結果	
	VExICADC	c 相電流変換結果	
	VExTMPREG0	a 相電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電流	
	VExTMPREG2	c 相電流	
内部	VExIAO	a 相ゼロ電流変換結果	16bit データ(上位 12bit に結果保持)
	VExIBO	b 相ゼロ電流変換結果	
	VExICO	c 相ゼロ電流変換結果	

- 電流極性判定

<演算式>

```
if ( VExMODE[15] = 1)                                          : 極性判定許可
    IA = VExTMPREG0
    if ( IASTS = xx0)                                          : 極性未確定
        if ( IA ≥ |HYS| )              IASTS = 001             : 正確定
        else if ( IA ≤ −|HYS| )        IASTS = 111             : 負確定
    else if ( IASTS = 001)                                     : 前回正
        if ( IA ≤ −|HYS| )             <IASTS> = 111           : 負変更
        else if ( IA < −HYS )          <IASTS> = 011           : 負変更(ヒステリシス領域)
    else if ( IASTS = 101)
        if ( IA ≥ |HYS| )              <IASTS> = 001           : 正(ヒステリシス領域外)
    else if ( IASTS = 111)                                     : 前回負
        if ( IA ≥ |HYS| )              <IASTS> = 001           : 正変更
        else if ( IA > HYS )           <IASTS> = 101           : 正変更(ヒステリシス領域)
    else if ( IASTS = 011)
        if ( IA ≤ −|HYS| )             <IASTS> = 111           : 負(ヒステリシス領域外)
( 残りの 2 相も同様に演算)
```

	レジスタ名	機能	
入力	VExHYS	電流極性判定ヒステリシス	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExDCTS	デッドタイム補償制御/ステータス	<ICSTS>, <IBSTS>, <IASTS> xx0: 未確定, x01:正電流, x11: 負電流
	VExMODE[15]	電流極性判定制御	<IPDEN> 0 : 判定禁止, 1 : 判定許可
	VExTMPREG0	a 相電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電流	
	VExTMPREG2	c 相電流	
出力	VExDCTS	デッドタイム補償制御/ ステータス	<ICSTS>, <IBSTS>, <IASTS> x0: 未確定, 01:正電流, 11: 負電流

2. 入力処理 2(タスク 10)

入力処理 2 タスクは 3 シャント(3 相検出、2 相検出)、2 センサの電流検出に対応します。また、PWM シフト 2 モードの PWM 出力時の 1 シャント電流検出に対応します。電流検出方向を相別に選択できます。

(注)入力処理 2 タスクはゼロ電流検出モードには対応していません。

(注)PWM シフト 2 は 1 シャント電流検出モード時のみ選択できます。

- 入力変換

<演算式>

```
[VDC 固定小数点数変換/保存]
if (VExMODE[4] = 0                    VExVDC = [DC 電圧] >>1
else                                  VExVDCL = [DC 電圧] >>1
[電流 1t 読み込み]
if ( [電流 1 相情報] = 1)              VExIAADC = [電流 1]
else if ( [電流 1 相情報] = 2)         VExIBADC = [電流 1]
else if ( [電流 1 相情報] = 3)         VExICADC = [電流 1]
[電流 2 読み込み]
N = [電流 2 相情報]                                          : 1ADC で順次検出
X = [電流 2]
if (VExFMODE[13] =  1                 N = [電流 4 相情報]    : 同時サンプリング時は電流 4
                                      X = [電流 4]
if ( N = 1)                           VExIAADC = X
else if ( N = 2)                      VExIBADC = X
else if ( N = 3)                      VExICADC = X]
[電流 3 読み込み]
if (VExFMODE[3:2] = 00)                                      : 3 相検出時のみ
    if ( [電流 3 相情報] = 1)          VExIAADC = [電流 3]
    else if ( [電流 3 相情報] = 2)     VExIBADC = [電流 3]
    else if ( [電流 3 相情報] = 3)     VExICADC = [電流 3]
else                                  N = 6 − N − [電流 1 相情報]  : 3 相検出以外は電流 3 を算出
[電流固定小数点数変換/保存]
IA = VExIAO − VExIAADC
if (VExFMODE[5] = 1)                  IA = −IA               : Ia 電流検出方向設定
IB = VExIBO − VExIBADC
if (VExFMODE[6] = 1)                  IB = −IB               : Ib 電流検出方向設定
IC = VExICO − VExICADC
if (VExFMODE[7] = 1)                  IC = −IC               : Ic 電流検出方向設定
if (VExFMODE[3:2] ≠ 00)                                      : 3 相検出以外は電流 3 を算出
    if ( N = 1)                       IA = − IB − IC
    else if ( N = 2)                  IB = − IC − IA
    else if ( N = 3)                  IC = − IA − IB
```

VExTMPREG0 = IA
VExTMPREG1 = IB
VExTMPREG2 = IC

	レジスタ名	機能	
入力	(DC 電圧)	ADC 変換結果入力	16bit データ(上位 12bit が変換結果)
	(電流 1)		
	(電流 2)		
	(電流 3)		
	(電流 4)		
	VExMODE[4]	VDC 保存レジスタ	<VDCSEL>
	VExFMODE[3:2]	電流検出モード	<IDMODE>
	VExFMODE[7:5]	電流検出極性	<ICPLMD>,<IBPLMD>,<IAPLMD> 0: シャントモード(In = VExInO − VExInADC) 1: センサモード(In = VExInADC − VExInO) 注) n = A,B,C
	VExFMODE[13]	同時サンプリング制御	<SADCEN> 0: 同時サンプリング禁止, 1: 同時サンプリング許可
出力	VExVDC	DC 電源電圧	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExVDCL	DC 電源電圧	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExIAADC	a 相電流変換結果	16bit データ(上位 12bit に結果保持)
	VExIBADC	b 相電流変換結果	
	VExICADC	c 相電流変換結果	
	VExTMPREG0	a 相電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電流	
	VExTMPREG2	c 相電流	
内部	VExIAO	a 相ゼロ電流変換結果	16bit データ(上位 12bit に結果保持)
	VExIBO	b 相ゼロ電流変換結果	
	VExICO	c 相ゼロ電流変換結果	

- 電流極性判定

<演算式>

```
if ( VExMODE[15] = 1)                                          : 極性判定許可
    IA = VExTMPREG0
    if ( IASTS = xx0)                                          : 極性未確定
        if ( IA ≥ |HYS| )              IASTS = 001             : 正確定
        else if ( IA ≤ −|HYS| )        IASTS = 111             : 負確定
    else if ( IASTS = 001)                                     : 前回正
        if ( IA ≤ −|HYS| )             <IASTS> = 111           : 負変更
        else if ( IA < −HYS )          <IASTS> = 011           : 負変更(ヒステリシス領域)
    else if ( IASTS = 101)
        if ( IA ≥ |HYS| )              <IASTS> = 001           : 正(ヒステリシス領域外)
    else if ( IASTS = 111)                                     : 前回負
        if ( IA ≥ |HYS| )              <IASTS> = 001           : 正変更
        else if ( IA > HYS )           <IASTS> = 101           : 正変更(ヒステリシス領域)
    else if ( IASTS = 011)
        if ( IA ≤ −|HYS| )             <IASTS> = 111           : 負(ヒステリシス領域外)
( 残りの 2 相も同様に演算)
```

	レジスタ名	機能	
入力	VExHYS	電流極性判定ヒステリシス	16bit 固定小数点データ(0.0 ~ 1.0、小数点以下 15bit)
	VExDCTS	デッドタイム補償制御/ステータス	<ICSTS>, <IBSTS>, <IASTS> x0: 未確定, 01:正電流, 11: 負電流
	VExMODE[15]	電流極性判定制御	<IPDEN> 0 : 判定禁止, 1 : 判定許可
	VExTMPREG0	a 相電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電流	
	VExTMPREG2	c 相電流	
出力	VExDCTS	デッドタイム補償制御/ ステータス	<ICSTS>, <IBSTS>, <IASTS> xx0: 未確定, x01:正電流, x11: 負電流

17.4.2.7 入力電流変換(相変換/座標軸変換)

入力電流変換は、相変換と座標軸変換の 2 つのタスクに分かれています。

1. 入力相変換(タスク 3)

入力相変換タスクは、Ia, Ib, Ic から Iα, Iβ を算出します。

<演算式>

```
if ( VExFMODE[12] = 0 )                                                    : 相変換許可
    VExTMPREG3 = VExTMPREG0                                                : Iα 算出
    VExTMPREG4 = 1 ÷ sqrt(3) × VExTMPREG1 − 1 ÷ sqrt(3) × VExTMPREG2       : Iβ 算出
else if ( VExFMODE[12] = 1 )                                               : 相変換禁止
    VExTMPREG3 = VExTMPREG0
    VExTMPREG4 = VExTMPREG1
```

	レジスタ名	機能	
入力	VExTMPREG0	a 相電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG1	b 相電流	
	VExTMPREG2	c 相電流	
	VExFMODE[12]	相変換禁止	<PHCVDIS>
出力	VExTMPREG3	α 軸電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG4	β 軸電流	

2. 入力座標軸変換(タスク 4)

入力座標軸変換タスクは Iα、Iβ、VExSINM、VExCOSM から Id、Iq を算出します。

a. 座標軸変換

<演算式>

```
if ( VExMCTLFM[5] = 0 )                                          : 微小パルスフラグ確認
    VExID = VExCOSM × VExTMPREG3 + VExSINM × VExTMPREG4          : Id 算出
    VExIQ = − VESINM × VExTMPREG3 + VExCOSM × VExTMPREG4         : Iq 算出
```

	レジスタ名	機能	
入力	VExTMPREG3	α 軸電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExTMPREG4	β 軸電流	
	VExSINM	θ での正弦値	16bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 15bit)
	VExCOSM	θ での余弦値	
	VExMCTLF[5]	微小パルスフラグ	<PLSLFM>
出力	VExID	d 軸電流	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExIQ	q 軸電流	

b. ATAN 演算

<ATANMD> = "10" の場合、d 軸,q 軸電流の偏角を算出します。

<ATANMD> = "11"の場合、モータの電圧方程式を使って d 軸,q 軸誘起電圧を算出して、誘起電圧の偏角を算出します。

<演算式>

```
if ( VExMODE[6] = 1 )                                                    : 偏角算出許可
    LD = VExCLD × [VExCLG 設定]                                          : d 軸インダクタンス
    LQ = VExCLQ × [VExCLG 設定]                                          : q 軸インダクタンス
    R = VExCR × [VExCRG 設定]                                            : 抵抗
    VDIV = VExVD − (VExID × R − VExOMEGA × VExIQ × LQ)                   : d 軸誘起電圧
    VQIV = VExVQ − (VExIQ × R + VExOMEGA × VExID × LD)                   : q 軸誘起電圧
    if ( VExMODE[5] = 1 )        VExDELTA = ATAN2(VQIV, VDIV)            : 誘起電圧偏角算出
    else                         VExDELTA = ATAN2(VExIQ, VExID)          : 電流偏角算出
```

	レジスタ名	機能	
入力	VExCLD	モータ d 軸インダクタンス	16bit 固定小数点データ(小数点以下 11bit)
	VExCLQ	モータ q 軸インダクタンス	
	VExCR	モータ抵抗値	
	VExCLG	インダクタンスレンジ設定	000: 1 倍、 001: $1/2^4$ 倍、 010: $1/2^8$ 倍、 011: $1/2^{12}$ 倍、 100: $1/2^{16}$ 倍、 101 ~ 111: Reserved
	VExCRG	抵抗レンジ設定	
	VExOMEGA	回転速度	16bit 固定小数点データ(小数点以下 15bit)
	VExVD	d 軸電圧	32bit 固定小数点データ(−1.0 ~ 1.0、小数点以下 31bit)
	VExVQ	q 軸電圧	
	VExMODE[6:5]	ATAN 動作モード設定	<ATANMD> 0x:演算禁止 10: Id,Iq の偏角算出 11: d 軸,q 軸の誘起電圧の偏角算出
出力	VExDELTA	偏角出力	16bit データ(−180 ~ 180, 0x8000 ~ 0x7FFF)

17.4.2.8 その他タスク

1. ATNA2(逆正接関数 2)(タスク 12)

ATAN2 タスクは、XY 平面上の原点と(X,Y)を結ぶ直線の X 軸から原点周りの角度を算出します。

<演算式>

X = VExTMPREG4
Y = VExTMPREG5
Z = ATAN (|Y| ÷ |X|) : 逆正接演算, 0 ~ 90°
if (X < 0 & Y ≥ 0) Z = 0x00008000 − Z : 第 2 象限(90 ~ 180°)
if (X < 0 & Y < 0) Z = 0xFFFF8000 + Z : 第 3 象限(−90 ~ −180°)
if (X ≥ 0 & Y < 0) Z = − Z : 第 4 象限(0 ~ −90°)
if (X = Y = 0) Z = 0x00000000 : 原点での 出力 (0°)
VExTMPREG5 = Z

	レジスタ名	機能	
入力	VExTMPREG4	入力 X	32bit 符号付きデータ
	VExTMPREG5	入力 Y	
出力	VExTMPREG5	位相値	32bit データ(0xFFFF8000 ~ 0x00008000 (−180~ 180°))

2. SQRT(平方根関数)(タスク 13)

SQRT タスクは 0.0～4.0 の入力値から平方根を計算して 0.0～2.0 を出力します。

<演算式>

VExTMPREG5 = SQRT(VExTMPREG5)　　: 平方根算出
X = VExTMPREG5　　: 入力 (0 ~ 4.0)
N = 0
if (X < 0x2000) N = 1
if (X < 0x0800) N = 2
if (X < 0x0200) N = 3
if (X < 0x0080) N = 4
if (X < 0x0020) N = 5
if (X < 0x0008) N = 6
if (X < 0x0002) N = 7
if (X ≥ 0x8000) N = −1
$X = X \times 2^{2N}$　　: 正規化 (0.25 ~ 1.0)
if (X > 0x7FFF) X = 0x7FFF
Z = SQRT(X)　　: 平方根演算,出力 0.5 ~ 1.0
if (X = 0) Z = 0
$Z = Z \div 2^{N}$　　: 逆変換 (0 ~ 2.0)
VExTMPREG5 = Z

	レジスタ名	機能	
入力	VExTMPREG5	入力値	32bit 固定小数点データ(0.0 ~ 4.0、小数点以下 15bit) 0x0000_0000 ~ 0x0001_FFFF
出力	VExTMPREG5	平方根値	32bit 固定小数点データ(0.0 ~ 2.0、小数点以下 15bit) 0x0000_0000 ~ 0x0000_FFFF

17.5 VE チャネルと PMD および ADC の組み合わせ

ベクトルエンジンの使用チャネルにより、使用できる PMD および ADC の組み合わせに制限があります。

また、電流検出選択と使用 ADC ユニット選択によっても使用する組み合わせは変わります。

3 シャントおよび 2 センサの電流検出では、2 つの ADC を使って電流を同時サンプリングすることができます。この場合に使用する電流入力は電流 1 と電流 4 になります。

表 17-8 ベクトルエンジンと PMD の組み合わせ

ベクトルエンジン		PMD チャネル 0	PMD チャネル 1
	チャネル 0	o	–
	チャネル 1	–	o

表 17-9 ベクトルエンジンと ADC の組み合わせ（同時サンプリングなし）

ベクトルエンジン		ADC ユニット A				ADC ユニット B			
チャネル	電流検出 VExMODE <IDMODE[1:0]>	ADAREG0	ADAREG1	ADAREG2	ADAREG3	ADBREG0	ADBREG1	ADBREG2	ADBREG3
0	00	電流 1	電流 2	電流 3	DC 電圧	–	–	–	–
	01	電流 1	電流 2	–	DC 電圧	–	–	–	–
	1x	電流 1	電流 2	–	DC 電圧	–	–	–	–
1	00	–	–	–	–	電流 2	電流 1	電流 3	DC 電圧
	01	–	–	–	–	電流 2	電流 1	–	DC 電圧
	1x	–	–	–	–	電流 2	電流 1	–	DC 電圧

表 17-10 ベクトルエンジンと ADC の組み合わせ（同時サンプリングあり）

ベクトルエンジン		ADC ユニット A				ADC ユニット B			
チャネル	電流検出 VExMODE <IDMODE[1:0]>	ADAREG0	ADAREG1	ADAREG2	ADAREG3	ADBREG0	ADBREG1	ADBREG2	ADBREG3
0	00	電流 1	–	電流 3	DC 電圧	電流 4	–	–	–
	01	電流 1	–	–	DC 電圧	電流 4	–	–	–
	1x	–	–	–	–	–	–	–	–
1	00	–	電流 4	–	–	–	電流 1	電流 3	DC 電圧
	01	–	電流 4	–	–	–	電流 1	–	DC 電圧
	1x	–	–	–	–	–	–	–	–

第 18 章　エンコーダ入力回路 (A-ENC)

18.1　概要

エンコーダ入力回路は、エンコーダモード、センサモード (3 種類)、タイマモードおよび位相カウンタモードの 6 つの動作モードに対応しています。

以下の機能を有しています。

- インクリメンタル形エンコーダ(Z 信号対応)およびホール IC センサ対応
- 回転エッジ検出(2 相入力：4 逓倍、3 相入力：6 逓倍)
- 回転方向判定
- カウンタ機能(イベントカウント、タイマカウント、位相カウント)
- 2 種類の割り込み出力
- 割り込み要求出力許可／禁止および要因フラグ
- 入力回路にデジタルノイズフィルタ内蔵
- PMD 回路の PWM 信号に同期したサンプリング可能
- 矩形波駆動の BLDC モータの BEMF(誘起電圧)ゼロクロス検出対応(PMD 回路使用時)
- 汎用 32 ビットタイマ／キャプチャとして使用可能

18.2　ブロック図

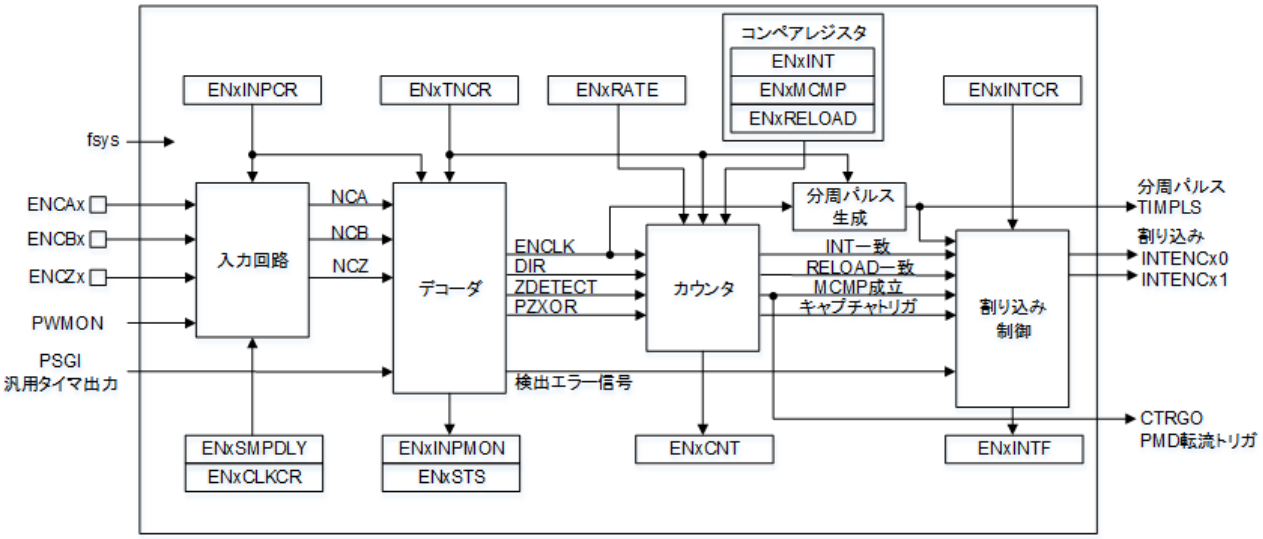


図 18-1 エンコーダ入力回路ブロック図

18.3　レジスタ

18.3.1　レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		address(Base+)
ENC 制御レジスタ	ENxTNCR	0x0000
RELOAD コンペアレジスタ	ENxRELOAD	0x0004
INT コンペアレジスタ	ENxINT	0x0008
カウンタ/キャプチャレジスタ	ENxCNT	0x000C
MCMP コンペアレジスタ	ENxMCMP	0x0010
位相カウントレートレジスタ	ENxRATE	0x0014
ステータスレジスタ	ENxSTS	0x0018
入力処理制御レジスタ	ENxINPCR	0x001C
サンプルディレイレジスタ	ENxSMPDLY	0x0020
入力モニタレジスタ	ENxINPMON	0x0024
サンプルクロック制御レジスタ	ENxCLKCR	0x0028
割り込み要求制御レジスタ	ENxINTCR	0x002C
割り込み要因フラグレジスタ	ENxINTF	0x0030

18.3.2 ENxTNCR(ENC 制御レジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	CMPSEL	UDMD		TOVMD	MCMPMD
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	DECMD		SDTEN	-	MODE			P3EN
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	TRGCAPMD	SFTCAP	ENCLR	ZESEL	
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	ZEN	ENRUN	-	-	-	ENDEV		
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-29	−	R	リードすると"0"が読めます。
28	CMPSEL	R/W	タイマモードのカウンタクリア条件 0：ENxINT レジスター致 1：ENxRELOAD レジスター致
27-26	UDMD	R/W	センサモード(位相カウント)、位相カウントモードのアップ/ダウンカウント制御 00：アップカウント 01：ダウンカウント 1x：ENxRATE レジスタでアップ・ダウン制御 "1x"を設定すると、ENxRATE<RATE> < 0 の場合ダウンカウント、<RATE> ≥ 0 の場合アップカウントします。
25	TOVMD	R/W	RELOAD 一致時のカウンタ動作(タイムオーバー動作設定) [センサモード(タイマカウント)] 0：カウント継続 1：カウント停止 カウント停止の場合、停止状態からカウント再開するにはソフトクリアで一致状態を解除します。 [タイマモード、センサモード(位相カウント)、位相カウンタモード] 0：カウンタクリアし、カウント継続 1：カウント停止 カウント停止の場合、停止状態からカウント再開するにはソフトクリアで一致状態を解除します。 [エンコーダモード] TOVMD 設定に関係なく、 CW 方向の場合、カウンタクリアしてカウント継続 CCW 方向の場合、カウント継続 [センサモード(イベントカウント)] RELOAD 一致は使用できません。
24	MCMPMD	R/W	[センサモード(タイマカウント)、タイマモード] ENxMCMP レジスタの比較モード 0：一致比較(ENxMCMP<MCMP> = カウンタ値) 1：大小比較(ENxMCMP<MCMP> ≤ カウンタ値) センサモード(タイマカウント)、タイマモード以外では、<MCMPMD> = 0 にしてください。

Bit	Bit Symbol	Type	機能
23-22	DECMD	R/W	[エンコーダモード、センサモード] デコーダ検出方向選択 00：　CW、CCW エッジ検出 入力信号(ENCAx、ENCBx、ENCZx)の変化を検出 01：　CW エッジ検出 前回回転エッジ検出状態からの入力信号の変化検出(検出結果保持) 10：　CCW エッジ検出 前回回転エッジ検出状態からの入力信号の変化検出(検出結果保持) 11：　CW、CCW エッジ検出 前回回転エッジ検出状態からの入力信号の変化検出(検出結果保持) タイマモード、位相カウントモードでは、<DECMD> = 00 にしてください。
21	SDTEN	R/W	センサモードで 3 相デコード選択時のスキップ検出 0：検出禁止 1：検出許可 スキップ検出するとエラーフラグ(ENxSTS<SKPDT>)がセットされさます。
20	–	R	リードすると"0"が読めます。
19-17	MODE[2:0]	R/W	動作モード設定 000：エンコーダモード 001：センサモード(イベントカウント) 010：センサモード(タイマカウント) 011：タイマモード 100：Reserved 101：Reserved 110：センサモード(位相カウント) 111：位相カウンタモード
16	P3EN	R/W	[センサモード](注 1) デコードモード設定(2 相 / 3 相入力選択) 0：2 相デコード 1：3 相デコード
15-13	–	R	リードすると"0"が読めます。
12	TRGCAPMD	R/W	[センサモード(タイマカウント、位相カウント)、タイマモード、位相カウンタモード] トリガキャプチャ動作選択 0：キャプチャ＆カウンタクリア 1：キャプチャのみ センサモード(タイマカウント、位相カウント)の回転エッジ検出時および、タイマモード、位相カウンタモードで Z 入力許可時のトリガキャプチャ動作を選択します。 ソフトキャプチャの場合、カウンタクリアしません。
11	SFTCAP	W	[センサモード(タイマカウント、位相カウント)、タイマモード、位相カウンタモード] ソフトキャプチャ実行 1：カウント値をキャプチャします "1"を書き込むとカウント値をキャプチャします。キャプチャ値の取り出しは ENxCNT レジスタをリードします。 "0"の書き込みは意味を持ちません。また、リードすると"0"が読めます。
10	ENCLR	W	カウンタクリア 1：クリア "1"を書き込むと、カウンタは"0"にクリアされます。クリアされた後は、再びカウントを始めます。"0" の書き込みは意味を持ちません。また、リードすると"0"が読めます。
9-8	ZESEL	R/W	[タイマモード、位相カウンタモード] Z 相入力許可時の検出エッジを選択します。(Z 相入力/PSGI 入力) 00：Reserved 01：立ち上がりエッジ検出 10：立ち下がりエッジ検出 11：両エッジ検出

Bit	Bit Symbol	Type	機能
7	ZEN	R/W	[エンコーダモード、タイマモード、位相カウンタモード](注 2) Z 相入力許可 0：ENCZ 入力無効 1：ENCZ 入力有効
6	ENRUN	R/W	エンコーダ入力回路の許可 0：禁止 1：許可 <ENRUN> = "1" で、<ZDET> を"0"クリアするとともにエンコーダ入力回路をイネーブルします。 <ENRUN> = "0" で、エンコーダ入力回路をディセーブルにします。
5-3	−	R	リードすると"0"が読めます。
2-0	ENDEV[2:0]	R/W	[エンコーダモード、センサモード(イベントカウンタ)] 回転エッジパルスの分周出力(TIMPLS)の分周比 この設定に従って回転エッジパルスが分周され、割り込み要因として使用されます。 000：1 分周　100：16 分周 001：2 分周　101：32 分周 010：4 分周　110：64 分周 011：8 分周　111：128 分周

注 1)　センサモード以外では<P3EN> = 0 に設定してください。
注 2)　エンコーダモード、タイマモード、位相カウンタモード以外では、<ZEN> = 0 に設定してください。

動作モードは <MODE[2:0]>、<P3EN>、 <ZEN>により決定し、全部で 12 種類の設定があります。
動作モードの設定表を以下に示します。

<MODE[2:0]>	<ZEN>	<P3EN>	入力端子	モード
000	0	0	A, B	エンコーダモード
	1		A, B, Z	エンコーダモード (Z 使用)
001	0	0	A, B	センサモード (イベントカウント)
		1	A, B, Z	センサモード (イベントカウント , Z 使用)
010	0	0	A, B	センサモード (タイマカウント , 2 相入力)
		1	A, B, Z	センサモード (タイマカウント , 3 相入力)
011	0	0	-	タイマモード
	1		Z	タイマモード (Z 使用)
110	0	0	A, B	センサモード (位相カウント , 2 相入力)
		1	A, B, Z	センサモード (位相カウント , 3 相入力)
111	0	0	-	位相カウンタモード
	1		Z	位相カウンタモード (Z 使用)

18.3.3　ENxRELOAD(RELOAD コンペアレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	RELOADH[15:8]							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	RELOADH[7:0]							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	RELOADL[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	RELOADL[7:0]							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	RELOADH	R/W	[エンコーダモード] カウンタの最大値を設定します。 1 回転分の入力パルス数 × 4 - 1 を設定します。 [センサモード(位相カウント)、位相カウンタモード] カウンタの最大値(1 回転分のカウント範囲)を設定します。 [センサモード(タイマカウント)、タイマモード] カウンタとの比較レジスタとして使用でき、一致で割り込みを発生します。 32 ビット比較の上位ビットを設定します。 [センサモード(イベントカウント)] 使用しません。
15-0	RELOADL	R/W	[センサモード(タイマカウント)、タイマモード] 32 ビット比較の下位 16 ビットを設定します。 [上記以外] 使用しません。

18.3.4 ENxINT(INT コンペアレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	INTH[15:8]							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	INTH[7:0]							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	INTL[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	INTL[7:0]							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	INTH	R/W	[エンコーダモード、センサモード(イベントカウント)] 16 ビットにてカウンタとの比較値を設定します。 カウンタと<INTH>との一致により INT 一致信号を発生します。割り込みを発生させることもできます。 [センサモード(タイマカウント)、タイマモード] 32 ビットにてカウンタとの比較値を設定します。<INTH>に 32 ビット比較の上位 16 ビットを設定します。 カウンタとの一致により INT 一致信号を発生します。割り込みを発生させることもできます。また、PWM 同期サンプリング許可の場合、位置検出開始信号になります。 [センサモード(位相カウント)、位相カウンタモード] 16 ビットにてカウンタとの比較値を設定します。 カウンタと<INTH>との一致により INT 一致信号を発生します。割り込みを発生させることもできます。
15-0	INTL	R/W	[センサモード(タイマカウント)、タイマモード] 32 ビット比較の下位 16 ビットを設定します。 [上記以外] 使用しません。

18.3.5　ENxCNT(カウンタレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	CNTH[15:8]							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	CNTH[7:0]							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	CNTL[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	CNTL[7:0]							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	CNTH	R	[エンコーダモード、センサモード(イベントカウント)] 回転エッジパルスのカウント値が読めます。 [センサモード(タイマカウント、位相カウント)] 回転エッジ検出/ENCZx 入力エッジ検出時のキャプチャ値、もしくはソフトキャプチャ値が読めます。 [タイマモード、位相カウンタモード] ENCZ 入力エッジ検出時のキャプチャ値もしくは、ソフトキャプチャ値が読めます。 [センサモード(タイマカウント)、タイマモード] 32 ビットキャプチャ値の上位 16 ビットが読めます。
15-0	CNTL	R	[エンコーダモード、センサモード(イベントカウント、位相カウント)、位相カウンタモード] 使用しません。 [センサモード(タイマカウント)、タイマモード] 32 ビットキャプチャ値の下位 16 ビットが読めます。

18.3.6 ENxMCMP(MCMP カウンタレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	MCMPH[15:8]							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	MCMPH[7:0]							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	MCMPL[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	MCMPL[7:0]							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	MCMPH	R/W	[センサモード(タイマカウント)、タイマモード] 32 ビットにてカウンタとの比較値を設定します。<MCMPH>に 32 ビット比較の上位 16 ビットを設定します。 割り込み要求の発生が可能です。 大小比較モード(<MCMPMD> = 1)の場合 ENxMCMP ≤ カウンタ値成立時に MCMP 成立信号を出力します。 このモードでは、レジスタ書き込みごとに 1 度だけ成立信号を出力します。 一致比較モード(<MCMPMD> = 0)の場合 ENxMCMP = カウンタ値時に MCMP 成立信号を出力します。 [センサモード(タイマカウント)、タイマモード以外] 16 ビットにてカウンタとの比較値を設定します。割り込み要求の発生が可能です。 ENxMCMP<MCMPH> = カウンタ値時に MCMP 成立信号を出力します。 ENxINTCR<MCMPIE>が許可されている場合、PMD 回路への転流トリガ信号を出力します。
15-0	MCMPL	R/W	[センサモード(タイマカウント)、タイマモード] 32 ビット比較の下位 16 ビット [上記以外] 使用しません

18.3.7　ENxRATE(位相カウントレートレジスタ)

	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	RATE[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	RATE[7:0]							
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。
15-0	RATE	R/W	[センサモード(位相カウント)、位相カウンタモード] カウンタのカウント周波数を設定します。 生成クロック周波数：fsys × <RATE>/2^{16} ENxTNCR<UDMD>の設定により、<RATE>設定値の符号の有無を選択できます。負数の場合、カウンタはダウンカウントします。 <UDMD> = 0x：符号なし、0 以上/1.0 未満(0x0000 ~ 0xFFFF) <UDMD> = 1x：符号あり、-0.5 以上/0.5 未満(0x8000 ~ 0x7FFF) <UDMD> = 1x の場合、<RATE>には 2 の補数を設定します。

18.3.8 ENxSTS(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	-	REVERR	UD	ZDET	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	SKPDT	PDERR	INERR
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	REVERR	R	[センサモード(タイマカウント、位相カウント)] 両方向検出時の<UD>の反転フラグ(注 1)(注 2) 0：- 1：<UD>反転発生 ENxTNCR<ENRUN> = "0" のときは常に "0" がセットされます。
13	UD	R	[エンコーダモード、センサモード(イベントカウント、タイマカウント、位相カウント)] 回転方向判定状態 0：CCW方向(反時計回り) 1：CW方向(時計回り) モータが CW 方向に回転しているときは "1" 、CCW 方向に回転しているときは "0" がセットされます。 また、ENxTNCR<ENRUN> = "0" のときは常に "0" がセットされます。
12	ZDET	R	ENCZ 入力通過検出 0：エンコーダ入力許可後、Z 入力未検出 1：Z 入力検出済み ENxTNCR<ENRUN> = "0"でクリアされます。
11-3	−	R	リードすると"0"が読めます。
2	SKPDT	R	スキップ検出許可時のスキップ検出フラグ(注 1) 0：未検出 1：スキップ検出
1	PDERR	R	[エンコーダモード、センサモード(イベントカウント、タイマカウント、位相カウント)] デコード検出エラーフラグ(注 1) 0：未検出 1：エラー検出
0	INERR	R	[センサモード(イベントカウント、タイマカウント、位相カウント)] 入力異常検出(注 1) 0：異常未検出 1：異常検出 3 相デコード時に 3 相入力すべてが Low またはすべて High が検出されると"1"がセットされます。

注 1) 本レジスタを読むとフラグがクリアされます。
注 2) モードを変更した後は、必ず最初にフラグをリードし、"0"にクリアしてください。

18.3.9　ENxINPCR(入力処理制御レジスタ)

	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	-	NCT						
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PDSTP	PDSTT	-	-	-	SYNCNCZEN	SYNCSPLMD	SYNCSPLEN
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	NCT	R/W	ノイズキャンセル時間 設定範囲：0~127(0x00~0x7F) 除去時間：設定値 × サンプルクロック周期(<SPLCKS>設定による) "0"設定時は、ノイズキャンセルしません。 PWM オフエッジサンプルモード時のサンプリングクロックは PWM 信号になります。
7	PDSTP	W	[センサモード(タイマカウント、位相カウント)] PWM 同期サンプリング時の位置検出停止コマンド(BFMF 検出制御) 1：位置検出停止 "1"を書き込むと位置検出を停止します。"0"の書き込みは意味を持ちません。 リードすると"0"が読めます。
6	PDSTT	W	センサモード(タイマカウント、位相カウント) PWM 同期サンプリング時の位置検出開始コマンド(BEMF 検出制御) 1：位置検出開始 "1"を書き込むと位置検出を開始します。"0"の書き込みは意味を持ちません。 リードすると"0"が読めます。
5-3	−	R	リードすると"0"が読めます。
2	SYNCNCZEN	R/W	PWM オン期間サンプリング時のノイズキャンセルカウンタ制御 0：PWM オフ期間カウンタ停止 1：PWM オフ期間カウンタ停止およびクリア PWM 同期サンプリング許可(<SYNCSPLEN> = 1)、PWM オン期間サンプリング選択(<SYNCSPLMD> = 0) に有効です。
1	SYNCSPLMD	R/W	PWM 同期サンプリング選択 0：PWM オン期間サンプリング 1：PWM オフエッジサンプリング PWM 同期サンプリング許可時(<SYNCSPLEN> = 1)に有効です。
0	SYNCSPLEN	R/W	PWM 同期サンプリング許可 0：連続サンプリング 1：PWM 同期サンプリング PMD 回路の PWM 信号に同期してサンプリングします。 センサモード(タイマカウント、位相カウント)で<SYNCSPLEN> = 1 に設定すると、デコード動作は BEMF 検出制御が有効になります。

18.3.10 ENxSMPDLY(サンプルディレイレジスタ)

	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	SMPDLY							
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	SMPDLY	R/W	サンプリング禁止期間 設定範囲：0~255(0x00~0xFF) 禁止時間：<SMPDLY>設定値×サンプリング周期(SPLCKS 設定による) PWM オン期間サンプリング時(ENxINPCR<SYNCSPLEN> = 1、ENxINPCR<SYNCSPLMD> = 0)の PWM オン後のサンプリング禁止期間を設定します。

18.3.11　ENxINPMON(入力モニタレジスタ)

	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	-	DETMONZ	DETMONB	DETMONA	-	SPLMONZ	SPLMONB	SPLMONA
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	DETMONZ	R	NCZ の位置検出状態モニタ 位置検出時の NCZ の値を保存しています。
5	DETMONB	R	NCB の位置検出状態モニタ 位置検出時の NCB の値を保持しています。
4	DETMONA	R	NCA の位置検出状態モニタ 位置検出時の NCA の値を保持しています。
3	−	R	リードすると"0"が読めます。
2	SPLMONZ	R	ENCZx のノイズキャンセル後の状態 ENCZx 入力のノイズキャンセル後の信号(NCZ)の状態
1	SPLMONB	R	ENCBx のノイズキャンセル後の状態 ENCBx 入力のノイズキャンセル後の信号(NCB)の状態
0	SPLMONA	R	ENCAx のノイズキャンセル後の状態 ENCAx 入力のノイズキャンセル後の信号(NCA)の状態

18.3.12 ENxCLKCR(サンプルクロック制御レジスタ)

	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	-	-	-	-	-	-	SPLCKS	
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。
1-0	SPLCKS	R/W	サンプリング周波数 00：fsys 01：fsys/2 10：fsys/4 11：fsys/8 ENCAx、ENCBx、ENCZx 入力のサンプリング周波数を選択します。 PWM 同期サンプリングでオフエッジサンプリング(ENxINPCR<SYNCSPLEN> = 1、ENxINPCR <SYNCSPLMD> = 1)の場合は無効です。

18.3.13　ENxINTCR(割り込み制御レジスタ)

	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	-	-	MCMPIE	RLDIE	CMPIE	ERRIE	CAPIE	TPLSIE
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	MCMPIE	R/W	MCMP 成立割り込み許可 0：禁止 1：許可 "1"を設定すると、MCMP 成立により INTENCx1 が発生します。
4	RLDIE	R/W	RELOAD 一致割り込み許可 0：禁止 1：許可 "1"を設定すると、RELOAD 一致により INTENCx1 が発生します。 エンコーダモード、センサモード(イベントカウント)では割り込み要求は発生しません。
3	CMPIE	R/W	INT 一致割り込み許可 0：禁止 1：許可 "1"を設定すると、INT 一致により INTENCx1 が発生します。
2	ERRIE	R/W	検出エラー割り込み許可 0：禁止 1：許可 "1"を設定すると、エッジ検出エラー(PDERR)またはスキップ検出(SKPDT)により INTENCx0 が発生します。 タイマモード、位相カウンタモードでは割り込み要求は発生しません。
1	CAPIE	R/W	キャプチャトリガ割り込み許可 0：禁止 1：許可 "1"を設定すると、外部トリガ(ENCZx 入力)または回転エッジパルス(ENCLK)でのキャプチャで INTENCx0 が発生します。 エンコーダモード、センサモード(イベントカウント)では割り込み要求は発生しません。
0	TPLSIE	R/W	回転エッジ分周割り込み許可 0：禁止 1：許可 "1"を設定すると、回転エッジの分周パルスにより INTENCx0 が発生します。 エンコーダモード、センサモード(イベントカウント)以外では割り込み要求は発生しません。

18.3.14 ENxINTF(割り込みフラグレジスタ)

	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	-	-	MCMPF	RLDCPF	INTCPF	ERRF	CAPF	TPLSF
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	MCMPF	R	MCMP 比較成立フラグ 0：発生なし 1：発生
4	RLDCPF	R	RELOAD 一致フラグ 0：発生なし 1：発生 エンコーダモード、センサモード(イベントカウント)ではセットされません。
3	INTCPF	R	INT 一致フラグ 0：発生なし 1：発生
2	ERRF	R	検出エラーフラグ 0：発生なし 1：発生 タイマモード、位相カウンタモードではセットされません。
1	CAPF	R	キャプチャフラグ 0：発生なし 1：発生 ソフトキャプチャでは、セットされません。 エンコーダモード、センサモード(イベントカウント)ではセットされません。
0	TPLSF	R	回転エッジ分周パルスフラグ 0：発生なし 1：発生 エンコーダモード、センサモード(イベントカウント)の場合に有効です。

注) 各フラグは許可された要因の発生でセットされ、ENxINTF レジスタをリードするとクリアされます。

18.4　動作説明

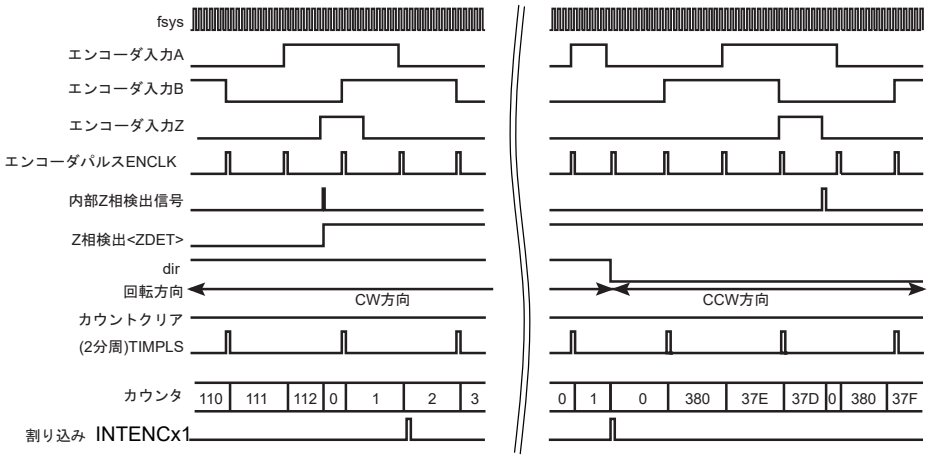
18.4.1　エンコーダモード

高速位置センサ対応 (位相判定) で、インクリメンタル形エンコーダ入力(AB および ABZ)に対応しています。

- 回転エッジ検出を行い、分周パルスの出力および割り込み要求が可能
- 回転エッジパルスカウント、任意カウント値で割り込み要求発生が可能
- 回転方向判定
- アップダウンカウント(回転方向判定により制御)
- カウント数設定可能
- 検出回転方向の指定可能
- 異常検出フラグ

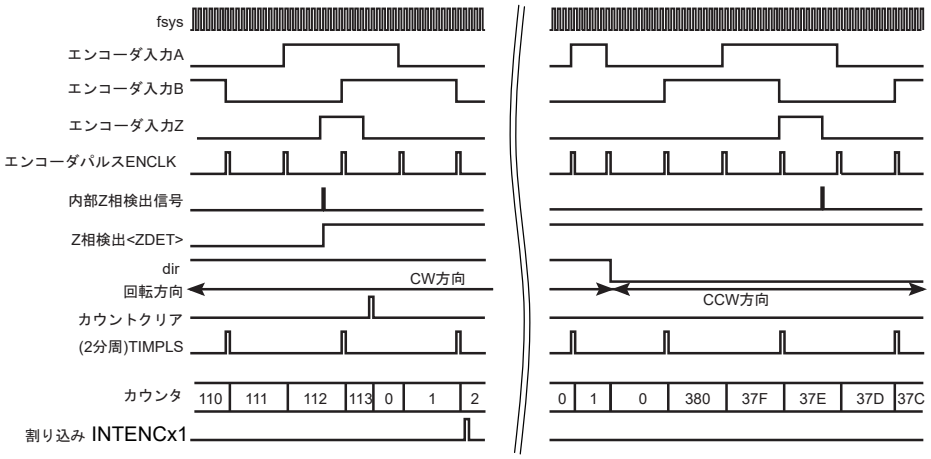
1. ENCZ 入力有効(ENxTNCR<ZEN> = 1)

   ENxRELOAD<RELOADH[15:0]> = 0x0380、ENxINT<INTH[15:0]> = 0x0002 の場合



2. ENCZ 入力無効(ENxTNCR<ZEN> = 0)

   ENxRELOAD<RELOADH[15:0]> = 0x0380、ENxINT<INTH[15:0]> = 0x0002 の場合



エンコーダモードは、インクリメンタル形エンコーダ信号を ENCA、ENCB、ENCZ 端子に接続します。ENCA、ENCB 信号を 4 逓倍し て、エンコーダパルスをカウントします。

CW 方向 (A 相が B 相に対して 90 度進んでいる状態) に回転しているときはアップカウン トを行い、カウンタ値 が <RELOADH> と等しくなったとき、次の ENCLK でカウンタ が"0"クリアされます。

CCW 方向 (A 相が B 相に対して 90 度遅れている状態) に回転しているときはダウンカ ウントを行い、カウンタ値 が"0x0000"と等しくなったとき、次の ENCLK でカウンタ に <RELOADH> の値がセットされます。

さらに、<ZEN> = "1"の場合は、CW 方向に回転しているときは Z 相の立ち上がりエッ ジでカウンタが"0"にクリアされます。CCW 方向に回転しているときは Z 相の立ち下 がりエッジでカウンタが"0"にクリアされます。ENCLK と Z 相検出のタイミングが同時になったときは、カウント動作を行わず、"0"にクリアされます。

ENxTNCR<ENCLR> に "1" が書き込まれると、カウンタは"0"にクリアされます。

ENxSTS<UD> は CW 方向に回転していることを検出しているときは "1"、 CCW 方向のときは "0" がセットされます。

ENxTNCR<DECMD[1:0]>設定で検出方向を CW 方向のみ、CCW 方向のみ指定することができます。また、<DECMD> = "00"以外の場合、前回のエッジ検出時に保存した入力状態 ENxINPMON <DETMONA><DETMONB><DETMONZ>と現在の入力値を比較し、回転エッジの検出をします。

ENCLK を分周した信号 (TIMPLS) が出力されます。

ENxINTCR<CMPIE> = "1" のとき、<ENINTH> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

ENxINTCR<MCMPIE> = "1" のとき、ENxMCMP<MCMPH[15:0]> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

ただし、<ZEN> = "1" の場合、ENxSTS<ZDET> = "0" の期間の一致では割り込みを発生しません。<ZDET>はエンコーダ入力許可後、最初の Z 信号を検出すると "1" になります。

<ZDET>、ENxSTS<UD> は、ENxTNCR<ENRUN> = "0" のときは"0"にクリアされます。

18.4.2 センサモード

低速位置センサ対応 (ゼロクロス判定) で、2 相ホールセンサ入力および 3 相ホールセンサ入力に対応しています。イベントカウントモードとタイマカウントモード および位相カウント モードの 3 種類があります。

タイマカウントモードおよび位相カウントモードは、PMD 回路からブラシレス DC モータの矩形波駆動している場合、PWM 同期サンプリングにすることで誘起電圧のゼロクロス検出に対応できます。

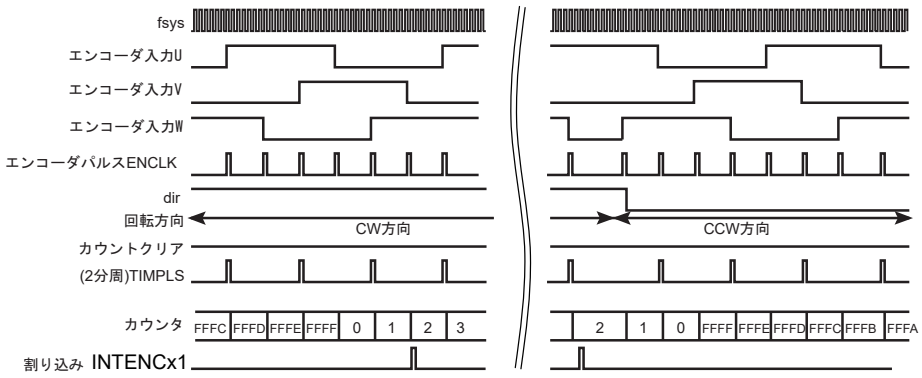
18.4.2.1　イベントカウント

回転エッジ検出によりカウントします。

- 回転エッジ検出を行い、分周パルスの出力および割り込み要求が可能
- 回転エッジパルスカウント、任意カウント値で割り込み要求発生が可能
- 回転方向判定
- アップダウンカウント(回転方向判定により制御)
- 検出回転方向の指定可能
- 異常検出フラグ

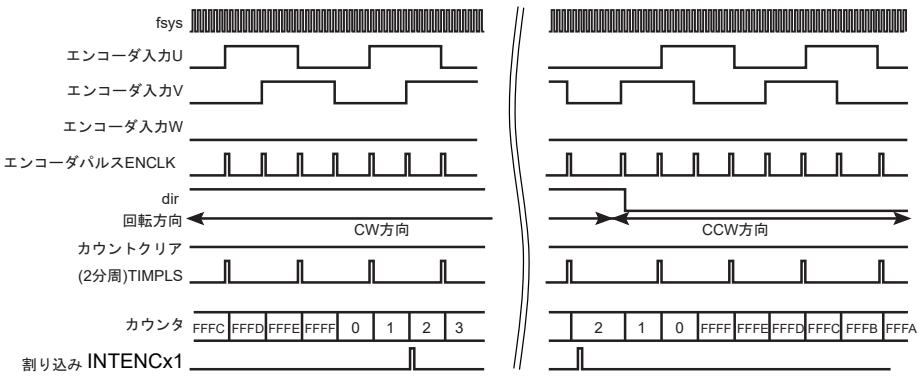
1. 3 相デコード(ENxTNCR<P3EN> = 1)

   ENxINT<INTH[15:0]> = 0x0002)の場合



2. 2 相デコード(ENxTNCR<P3EN> = 0)

   ENxINT<INTH[15:0]> = 0x0002)



ホールセンサ入力(U,V,W)を ENCAx、ENCBx、ENCZx に接続します。<P3EN> = "0"の場合は、2 相入力(ENCAx、ENCBx)を 4 逓倍、<P3EN> = "1"の場合は、3 相入力(ENCAx、ENCBx、ENCZx)を 6 逓倍して、ホールセンサパルスを カウントします。

CW 方向 (ENCA が ENCB 対して 90 度進んでいる状態) に回転していると きはアップカウントを行い、カウント値 が"0xFFFF"と等しくなったとき、次の ENCLK でカウンタが"0"にクリアされます。

CCW 方向 (ENCA が ENCB に対して 90 度遅れている状態) に回転しているとき はダウンカウントを行い、カウント値 が"0x0000"と等しくなったとき、次の ENCLK でカウンタに"0xFFFF"がセットされます。

ENxTNCR<ENCLR> に "1" が書き込まれると、カウンタは"0"にクリアされます。

ENxSTS<UD> は CW 方向に回転していることを検出しているときは "1"、 CCW 方向のときは "0" がセットされます。<UD> は ENxTNCR<ENRUN> = "0" のときは"0"にクリアされます。

ENxTNCR<DECMD[1:0]>設定で検出方向を CW 方向のみ、CCW 方向のみ指定することができます。また、<DECMD> = "00"以外の場合、前回のエッジ検出時に保存した入力状態 ENxINPMON <DETMONA><DETMONB><DETMONZ>と現在の入力値を比較し、回転エッジの検出をします。

ENCLK を分周した信号 (TIMPLS) が出力されます。

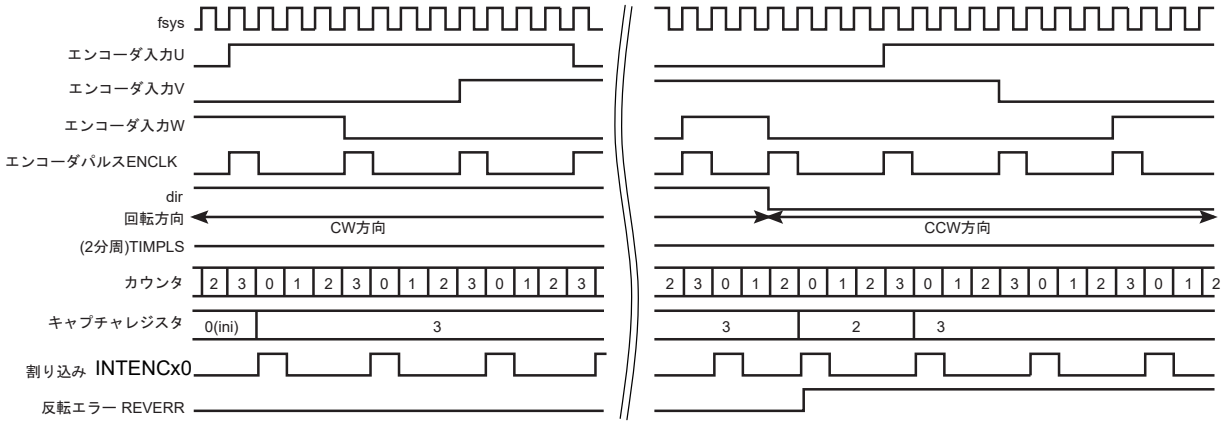
ENxINTCR<CMPIE> = "1" のとき、<ENINTH> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

ENxINTCR<MCMPIE> = "1" のとき、<ENMCMPH> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

18.4.2.2 タイマカウント

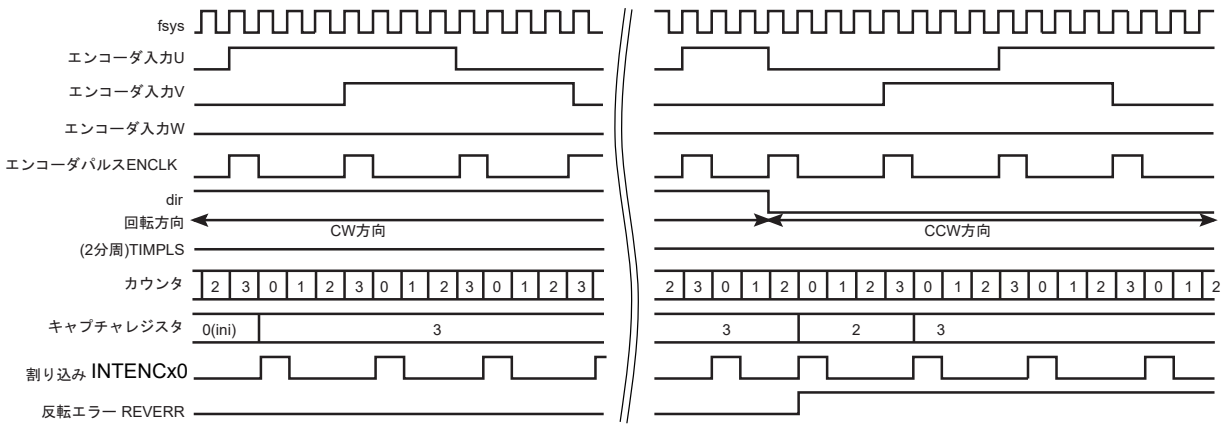
1. 3 相デコード(ENxTNCR<P3EN> = 1)

   ENxINT<INTH[15:0]><INTL[15:0]> = 0x00000002 の場合



2. 2 相デコード(ENxTNCR<P3EN> = 0)

   ENxINT<INTH[15:0]><INTL[15:0]> = 0x00000002 の場合



ホールセンサ入力(U,V,W)を ENCAx、ENCBx、ENCZx に接続します。<P3EN> = "0" の場合は、2 相入力(ENCA、ENCB)から 4 逓倍、<P3EN> = "1" の場合は、3 相入力(ENCA、ENCB、ENCZ)から 6 逓倍の回転エッジパルス(ENCLK)を生成します。

カウンタは常にアップカウントを行い、ENCLK でカウンタが"0x00000000"にクリアされます。

ENxTNCR<ENCLR> に "1" が書き込まれると、カウンタは"0x00000000"にクリアされます。

ENCLK により、カウンタ値がキャプチャされます。キャプチャした値は ENxCNT レジスタから読み出すことができます。

ENxTNCR<SFTCAP> に "1" が書き込まれると、カウンタ値がキャプチャされます。キャプチャは任意のタイミングで行うことができます。キャプチャした値は ENxCNT レジスタから読み出すことができます。

ENxCNT レジスタの値 (キャプチャ値) は、ENxTNCR<ENRUN> の値にかかわらず保持されます。

ENxSTS<UD> は CW 方向に回転していることを検出しているときは "1"、 CCW 方向のときは "0" がセットされます。<UD> は <ENRUN> = "0" のときは"0"にクリアされます。回転方向が変化した場合は ENxSTS<REVERR> = "1" にセットされます。フラグは読み出すことでクリアされます。

ENxTNCR<DECMD[1:0]>設定で検出方向を CW 方向のみ、CCW 方向のみ指定することができます。また、<DECMD> = "00"以外の場合、前回のエッジ検出時に保存した入力状態 ENxINPMON <DETMONA><DETMONB><DETMONZ>と現在の入力値を比較し、回転エッジを検出します。

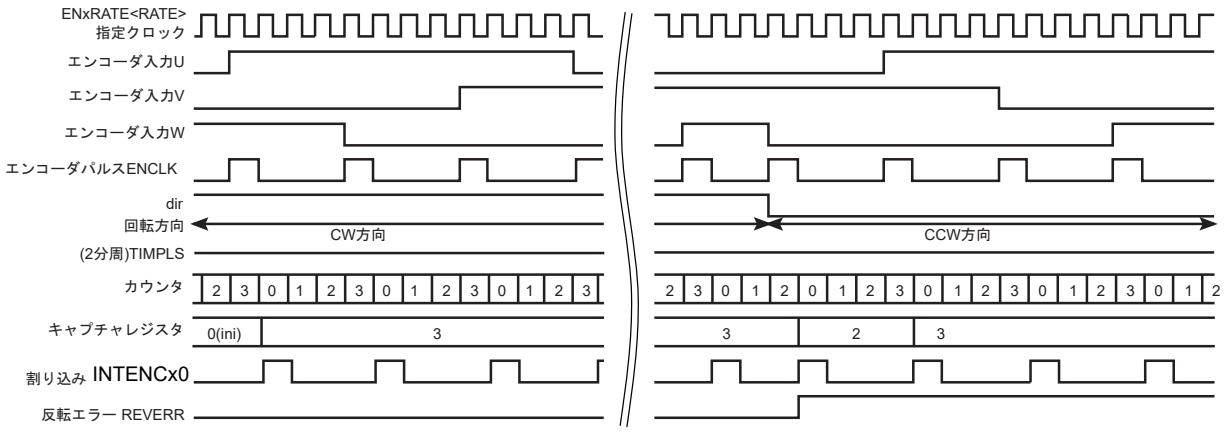
ENxINTCR<RLDIE> = "1"のとき、ENxRELOAD<RELOADH[15:0]><RELOADL[15:0]>の値とカウンタ値が等しくな ったときに INTENCx1 割り込みを発生させることができます。

ENxINTCR<CMPIE> = "1" のとき、ENxINT<INTH[15:0]><INTL[15:0]> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

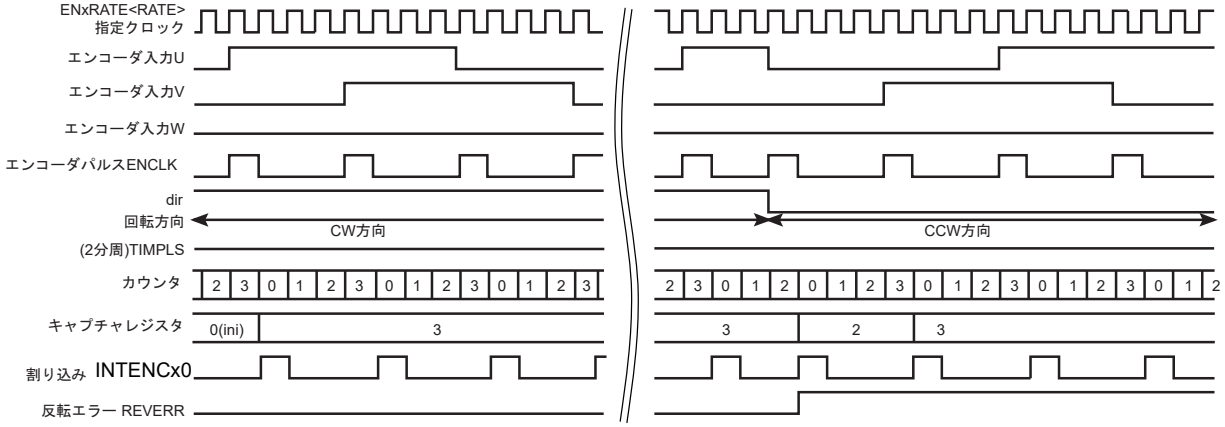
ENxINTCR<MCMPIE> = "1" のとき、ENxMCMP<MCMPH[15:0]><MCMPL[15:0]> の値とカウンタ値が等しくなったときにINTENCx1割り込みを発生させることができます。
また、ENxINTCR<MCMPMD> = "1" に設定すると、カウンタ値が <MCMPH> 値以上になったときに INTENCx1 割り込みを発生させることができます。

18.4.2.3　位相カウント

1. 3 相デコード(ENxTNCR<P3EN> = 1)



2. 2 相デコード(ENxTNCR<P3EN> = 0)



ホールセンサ入力(U,V,W)を ENCAx、ENCBx、ENCZx に接続します。<P3EN> = "0" の場合は、2 相入力(ENCAx、ENCBx)から 4 逓倍、<P3EN> = "1" の場合は、3 相入力(ENCAx、ENCBx、ENCZx)から 6 逓倍の回転エッジパルス(ENCLK)を生成します。

カウンタは<UDMD>設定および RATE レジスタ設定で任意周波数でアップダウンカウントを制御できます。アップカウント時に カウント値が <RELOADH> と等しくなるとカウンタ が"0"クリアされます。ダウンカ ウント時に カウント値が"0x0000"と等しくなるとカウンタに <RELOADH> 値がセットされます。

<ENCLR> に "1" が書き込まれると、カウンタは"0"にクリアされます。

<TOVMD> = "1" の場合、カウンタは <RELOADH> 値で停止します。

ENCLK により、カウンタ値がキャプチャされます。キャプチャした値 は ENxCNT レジスタから読み出すことができます。

<SFTCAP> に "1" が書き込まれると、カウンタ値がキャプチャされます。キャ プチャは任意のタイミングで行うことができます。キャプチャした値は ENxCNT レジス タから読み出すことができます。

ENxCNT レジスタの値 (キャプチャ値) は、<ENRUN> の値にかかわらず保持されます。

<UD> は CW 方向に回転していることを検出しているときは "1"、 CCW 方向のときは "0" がセットされます。<UD> は <ENRUN> = "0" のときは"0"にクリアされます。回転方向が変化した場合は <REVERR> = "1" にセットされます。フラグは読み出すこ とでクリアされます。

ENxTNCR<DECMD[1:0]>設定で検出方向を CW 方向のみ、CCW 方向のみ指定することができます。また、<DECMD> = "00" 以外の場合、前回のエッジ検出時に保存した入力状態 ENxINPMON <DETMONA> <DETMONB> <DETMONZ>と現在の入力値を比較し、回転エッジの検出をします。

ENxINTCR<CMPIE> = "1" のとき、<ENxINT> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

ENxINTCR<MCMPIE> = "1" のとき、<ENxMCMP> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

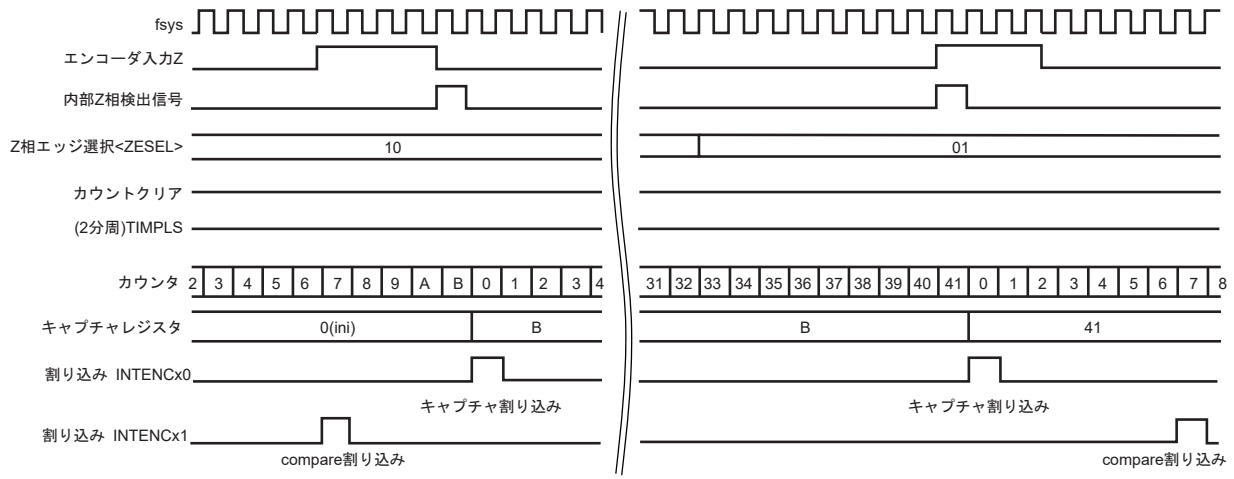
18.4.3　タイマモード

汎用 32 ビットタイマとして使用できます。

- 32 ビットアップカウンタ(fsys クロックでカウント)
- カウンタクリア制御(ソフトクリア、コンペア一致クリア、外部トリガ)
- コンペア機能により 一致割り込み発生
- キャプチャ機能： 外部トリガキャプチャ(割り込み発生可能)、ソフトキャプチャ

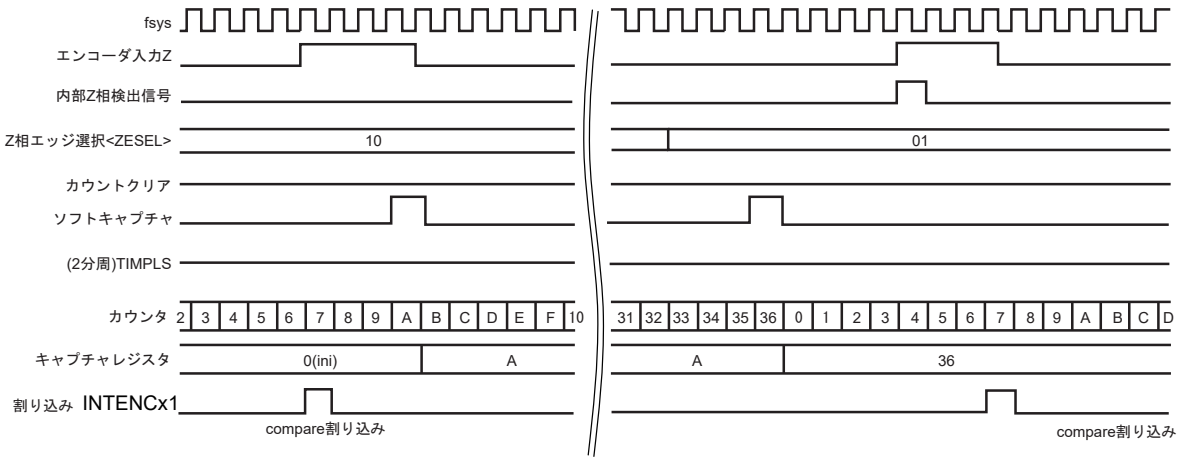
1. ENCZ 入力有効(ENxTNCR<ZEN> = 1)

   ENxINT<INTH[15:0]><INTL[15:0]> = 0x0000_0006



2. ENCZ 入力無効(ENxTNCR<ZEN> = 0)

   ENxINT<INTH[15:0]><INTL[15:0]> = 0x0000_0006)



<ZEN> = "1" のとき、Z 入力を外部トリガとして使います。<ZEN> = "0" のとき、外部トリガは使用しません。

カウンタは常にアップカウントを行います。

ENxTNCR<ENCLR> に "1" が書き込まれると、カウンタは"0"にクリアされます。

<ZEN> = "1" の場合、ENxTNCR<ZESEL> = "01" のとき は Z 相の立ち上がりエッジでカウンタが"0"にクリアされ、<ZESEL> = "10" のときは Z 相の立ち下がりエッジでカウンタが"0"にクリアされ、<ZESEL> = "11"のときは Z 相の両エッジでカウンタが"0"にクリアされます。

Z 相のエッジ検出により、カウンタ値がキャプチャされます。キャプチャした値は ENxCNT レジスタから読み出すことができます。

ENxTNCR<SFTCAP> に "1" が書き込まれると、カウンタ値がキャプチャされます。キャプチャは任意のタイミングで行うことができます。キャプチャした値は ENxCNT レジスタから読み出すことができます。

ENxCNT レジスタの値 (キャプチャ値)は、ENxTNCR<ENRUN>の値にかかわらず保持されます。キャプチャ値のクリア要因はリセットのみです。

ENxINTCR<RLDIE> = "1"のとき、ENxRELOAD<RELOADH[15:0]><RELOADL[15:0]>の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

ENxINTCR<CMPIE> = "1" のとき、ENxINT<INTH[15:0]><INTL[15:0]> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

ENxINTCR<MCMPIE> = "1" のとき、ENxMCMP<MCMPH[15:0]><MCMPL[15:0]> の値とカウンタ値が等しくなったときに INTENCx1 割り込みを発生させることができます。

また、ENxTNCR<MCMPMD> = "1" に設定すると、カウンタ値が <MCMPH[15:0]><MCMPL[15:0]> 値以上になったときに INTENCx1 割り込みを発生させることができます。

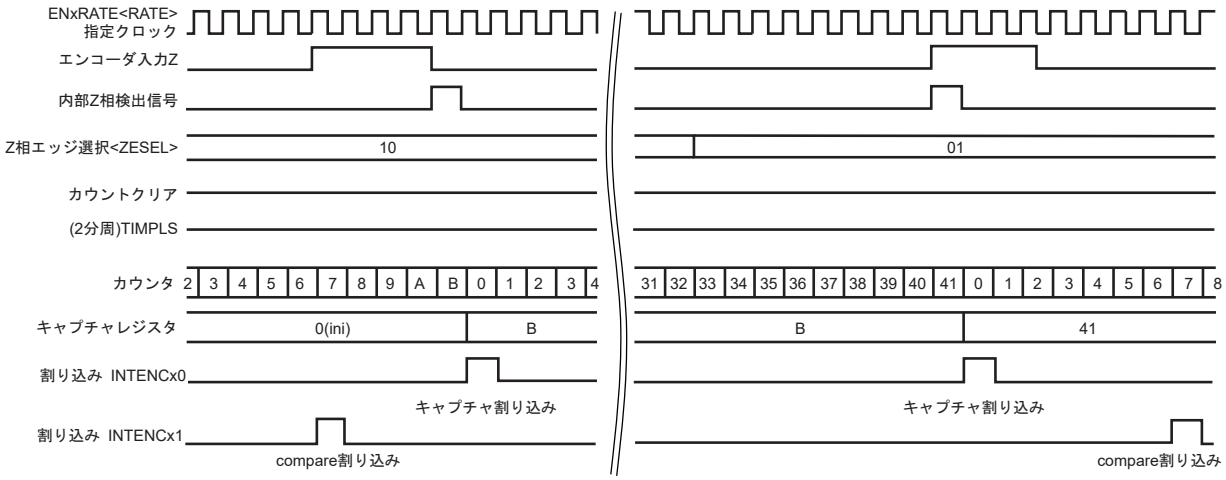
18.4.4 位相カウンタモード

任意周波数で制御できる 16 ビットカウンタです。

- アップダウン制御可能
- コンペア機能 、一致割り込み要求発生可能
- ENCZ 入力でキャプチャおよびカウンタクリア可能、割り込み要求発生可能

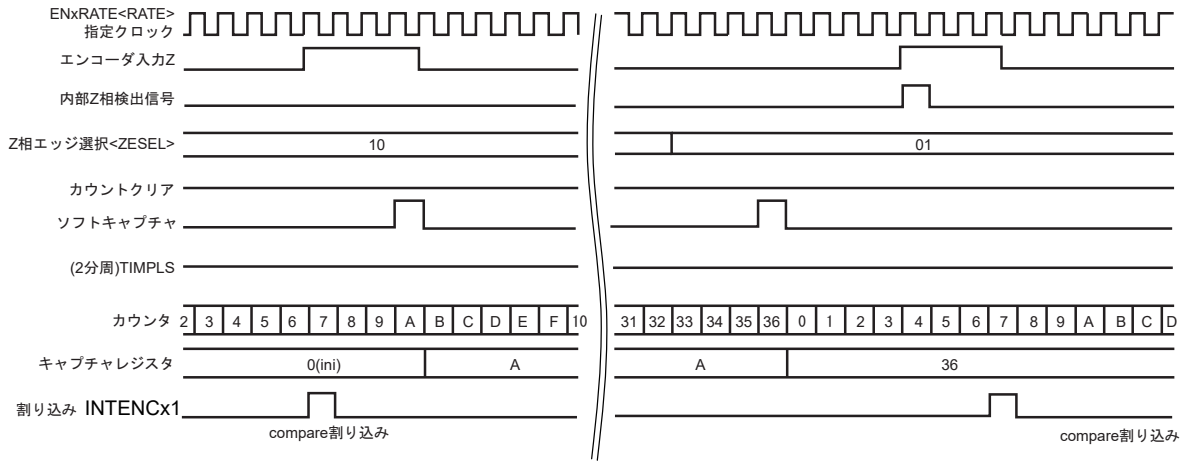
1. ENCZ 入力有効(ENxTNCR<ZEN> = 1)

   ENxINT<INTH[15:0]> = 0x0006



2. ENCZ 入力無効(ENxTNCR<ZEN> = 0)

   ENxINT<INTH[15:0]> = 0x0006)



<ZEN> = "1" のとき、Z 入力端子を外部トリガとして使います。<ZEN> = "0" のとき、外部トリガは使用しません。

カウンタは ENxTNCR<UDMD>設定および ENxRATE レジスタ設定で任意周波数でアップダウンカウントを制御できます。

アップカウント時に カウント値 が ENxRELOAD<RELOADH[15:0]> と等しくなるとカウンタが"0"クリアされます。

ダウンカ ウント時に カウント値 が"0x0000"と等しくなるとカウンタに <RELOADH> 値がセットされます。

ENxTNCR<TOVMD> = "1" の場合、カウンタは <RELOADH> 値で停止します。

ENxTNCR<ENCLR> に "1" が書き込まれると、カウンタは"0"にクリアされます。

<ZEN> = "1" の場合、ENxTNCR<ZESEL> = "01" のとき はZ相の立ち上がりエッジでカウンタが"0"にクリアされ、<ZESEL> = "10" のときはZ相の立ち下がりエッジでカウンタが"0"にクリアされ、<ZESEL> = "11"のときはZ相の両エッジでカウンタが"0"にクリアされます。

Z相のエッジ検出により、カウンタ値がキャプチャされます。キャプチャした値はENxCNTレジスタから読み出すことができます。

ENxTNCR<SFTCAP>に"1"が書き込まれると、カウンタ値がキャプチャされます。キャプチャは任意のタイミングで行うことができます。キャプチャした値はENxCNTレジスタから読み出すことができます。

ENxCNTレジスタの値(キャプチャ値)　は、ENxTNCR<ENRUN>の値にかかわらず保持されます。キャプチャ値のクリア要因はリセットのみです。

ENxINTCR<CMPIE> = "1" のとき、ENxINT<INTH> の値とカウンタ値が等しくなったときにINTENCx1割り込みを発生させることができます。

ENxINTCR<MCMPIE> = "1" のとき、ENxMCMP<MCMPH> の値とカウンタ値が等しくなったときにINTENCx1割り込みを発生させることができます。

18.4.4.1　位相差カウンタモード

位相カウンタモードで <P3EN> = <ZEN> = "1" に設定すると位相差カウンタモードになります。アップダウンカウンタをTMRB出力とENCZ入力で制御します。

- TMRB出力とENCZ入力が同じ場合はアップカウントし、異なる場合はダウンカウントします。
- TMRB出力エッジでキャプチャおよびカウントクリア可能、割り込み要求発生可能

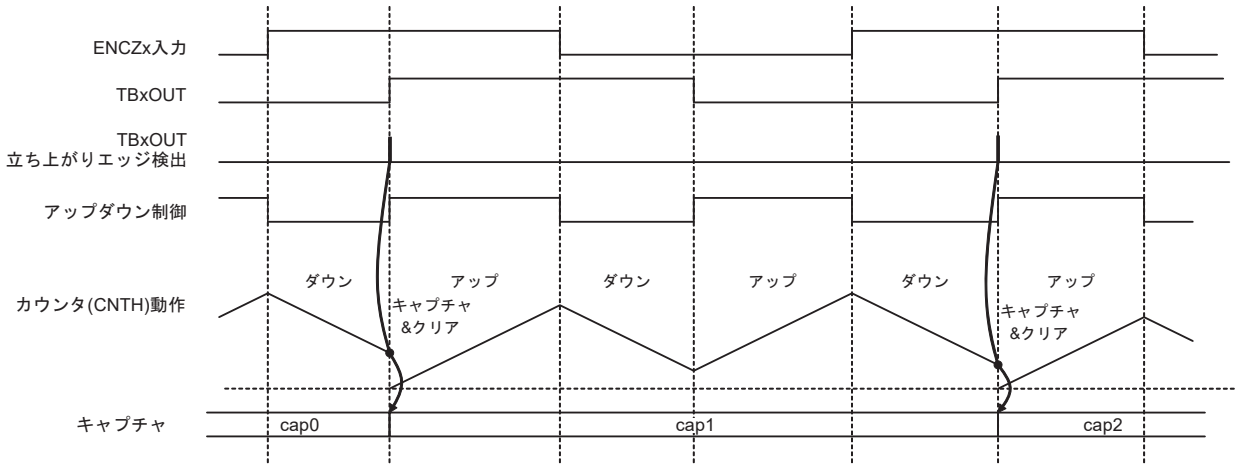


図 18-2 位相カウンタモード(位相差)の動作

TBxOUT信号のエッジ検出し、カウンタキャプチャおよびカウンタクリアします。この際の検出エッジはENxTNCR<ZESEL>で選択します。

キャプチャ値はENCZ入力とTBxOUT信号との位相差を表します。ENCZ入力とTBxOUT信号との位相差が1/4周期の場合が基準（キャプチャ値が0）になります。

18.5　回路別の機能概要

18.5.1　入力回路

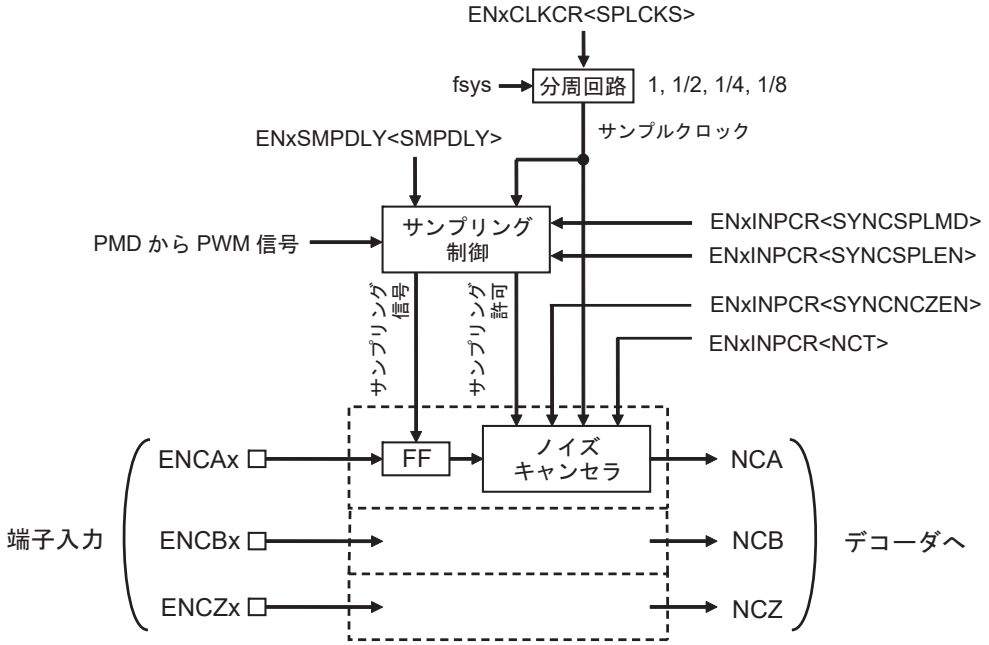


図 18-3 入力回路構成

入力部回路は、端子入力（ENA,ENB,ENZ）を所定のサンプリング信号でサンプリングし、デジタルノイズフィルタによるノイズ除去を行います。

18.5.1.1　サンプルクロック

<SPLCKS>により fsys、fsys/2、fsys/4、fsys/8 を選択できます。

18.5.1.2　サンプリングモード

1. 連続サンプリング(ENxINPCR<SYNCSPLEN> = "0")

   サンプルクロックで入力信号をサンプリングします。

2. PWM 同期サンプリング(ENxINPCR<SYNCSPLEN> = "1")

   PMD 回路からの PWM 信号に同期したサンプリングを行います。

   - PWM オン期間サンプリング(ENxINPCR<SYNSPLMD> = "0")。

     PWM 信号がオン期間中だけ ENxCLKCR<SPLCKS>で選択されたサンプルクロックでサンプリングします。
   - PWM オフエッジサンプリング(ENxINPCR<SYNSPLMD> = "1")

     PWM 信号がサンプリング信号になり、PWM 信号のオフエッジでサンプリングします。
   - PWM オンディレイ設定

ENxSMPLDSY<SMPDLY> で PWM オン期間サンプリング時にオンディレイを設定できます。

ディレイ時間： <SMPDLY> × サンプルクロック周期

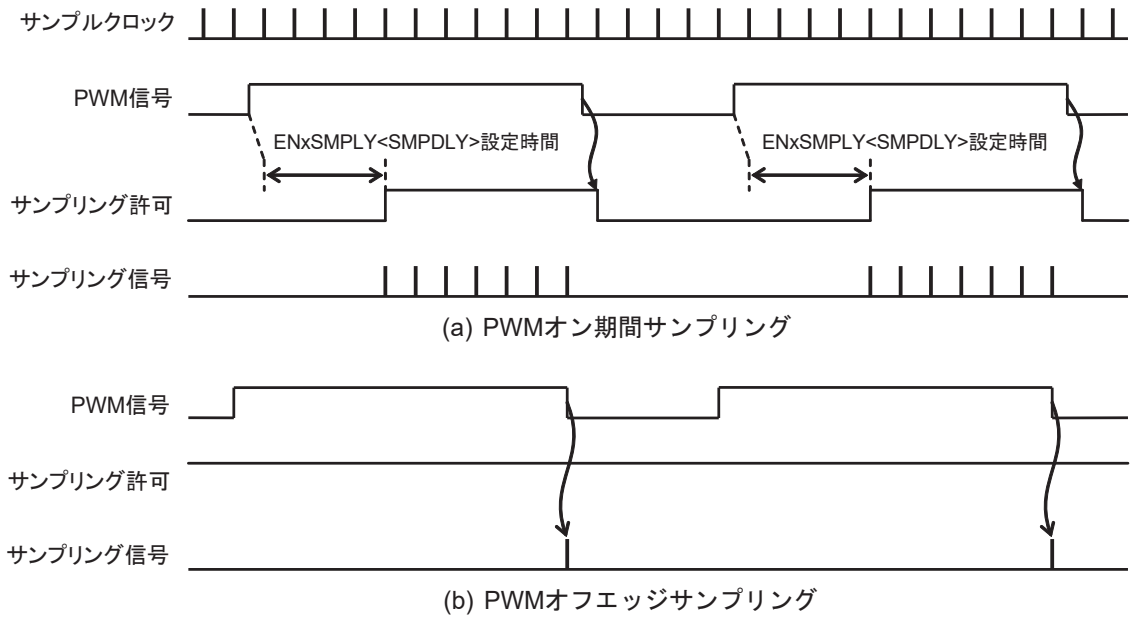


図 18-4 PWN 同期サンプリング

18.5.1.3 ノイズキャンセル

1. ノイズキャンセル時間

ENxINPCR<NCT[6:0]>でノイズキャンセル時間を設定します。ノイズキャンセル時間は以下の計算式で求められます。

ノイズキャンセル時間：<NCT>×サンプルクロック周期

注) <NCT>に"0"を設定するとノイズキャンセルは無効になります。

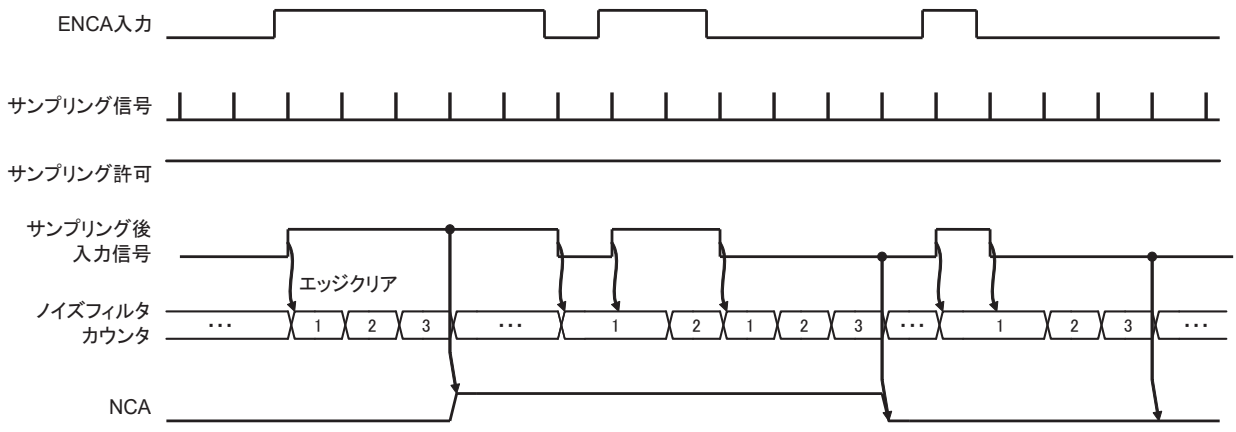


図 18-5 ノイズキャンセル(連続、<NCT> = 3)

2. PWM オン期間サンプリング時のノイズキャンセル

・　PWM オフ期間にノイズ除去タイマを停止（ENxINPCR<SYNCNCZEN > = "0"）
・　PWM オフ期間にノイズ除去タイマをクリア（ENxINPCR<SYNCNCZEN> = "1"）

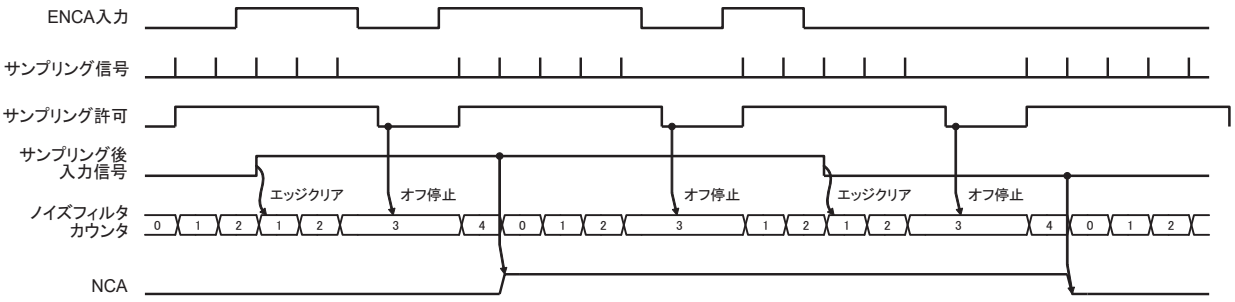


図 18-6 ノイズキャンセル(PWM オン期間サンプリング、PWM オフ停止、<NCT> = 4)

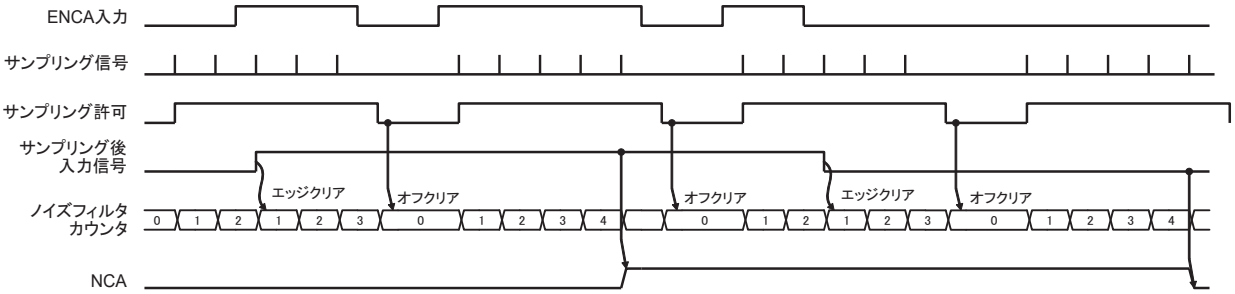


図 18-7 ノイズキャンセル(PWM オン期間サンプリング、PWM オフクリア、<NCT> = 4)

18.5.2 デコーダ

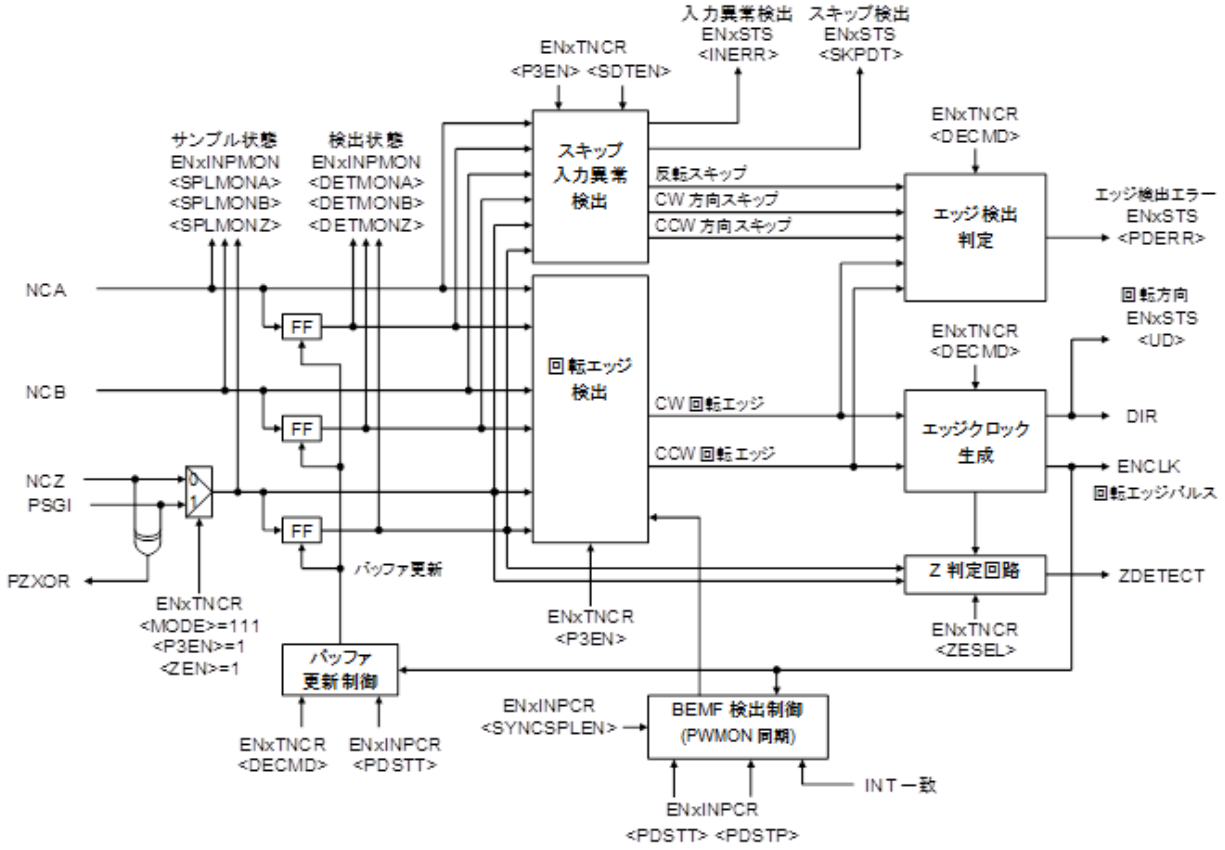


図 18-8 デコーダ構成

デコーダは、ノイズ除去後の 2 相または 3 相入力信号から、回転エッジ検出および回転方向判定を行います。また、インクリメンタル形エンコーダ接続時の Z 相検出、単相入力時の Z 信号のエッジ検出を行います。

18.5.2.1 回転エッジ検出と方向信号生成

1. 2 相デコード（ENxTNCR<P3EN> = "0"）

   エンコーダモードとセンサモード（<P3EN> = "0"設定）で対応します。

   2 相デコードでは 4 つの入力パターンの変化（回転エッジ）を検出します。

   CW 方向入力の場合、①→②、②→③、③→④、④→①の回転エッジを検出し、これらの回転エッジ検出でステータス ENxSTS<UD> は "1" になります。

   CCW 方向入力の場合、④→③、③→②、②→①、①→④の回転エッジを検出し、これらの回転エッジ検出でステータス ENxSTS<UD> は "0" になります。

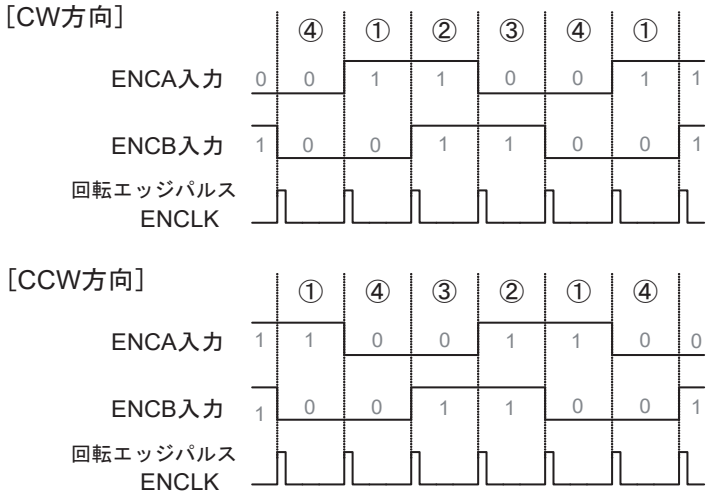


図 18-9 2 相デコーダ波形

2. 3 相デコード （ ENxTNCR<P3EN> = "1"）

センサモード のみ対応します。

3 相デコードでは 6 つの入力パターンの変化（回転エッジ）を検出します。

CW 方向の場合、①→②、②→③、③→④、④→⑤、⑤→⑥、⑥→①の回転エッジを検出し、これらの回転エッジ検出でステータス ENxSTS<UD> に "1" を設定します。

CCW 方向の場合、⑥→⑤、⑤→④、④→③、③→②、②→①、①→⑥の回転エッジを検出し、これらの回転エッジ検出でステータス ENxSTS<UD> に "0" を設定します。

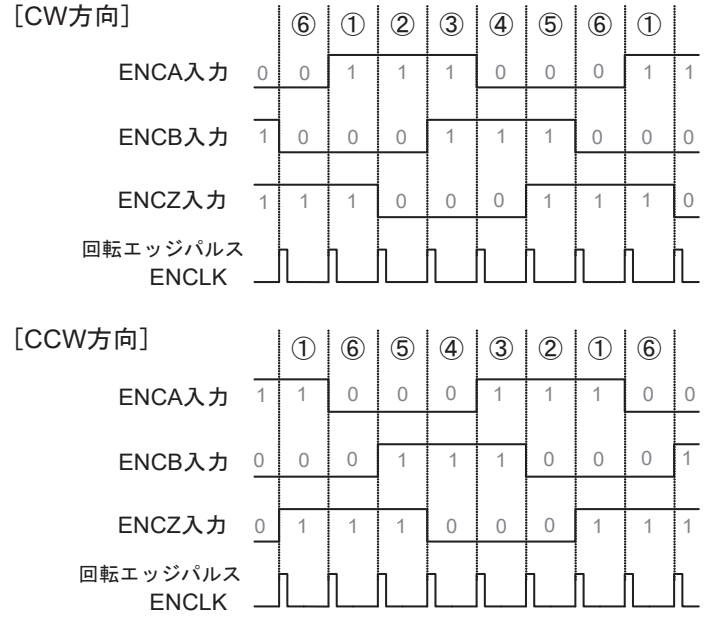


図 18-10 3 相デコーダ波形

18.5.2.2　Z 判定回路

Z 入力のエッジを検出します。

1. エンコーダモード

   CW 方向では立ち上がりエッジを検出し、CCW 方向では立ち下がりエッジを検出します。

2. タイマモード,位相カウンタモード

   <ZESEL> で 立ち上がりエッジ検出、立ち下がりエッジ検出および両エッジ検出を選択できます。

18.5.2.3 スキップ判定と入力異常判定

1. スキップ判定

   ENxTNCR<SDTEN> = "1" で有効になります。

   スキップ検出すると <SKPDT> が "1" にセットされます。

   - 2 相デコード （ENxSTS<P3EN> = "0"）のスキップ検出の組み合わせ

     反転スキップ検出：①→③、②→④、③→①、④→②
   - 3 相デコード （ENxSTS<P3EN> = "1"）のスキップ検出の組み合わせ

     CW 方向スキップ検出 ：①→③、②→④、③→⑤、④→⑥、⑤→①、⑥→②

     CCW 方向スキップ検出：①→⑤、②→⑥、③→①、④→②、⑤→③、⑥→④

     反転スキップ検出　　：①→④、④→①、②→⑤、⑤→②、③→⑥、⑥→③

2. 入力異常判定

   センサモード(イベントカウント,タイマカウント,位相カウント)で 3 相デコード時に 3 入力が全て"0" または 全て"1" に変わるエッジ検出で入力異常と判定します。入力異常と判定すると ENxSTS<INERR> が "1" にセットされます。

18.5.2.4 エッジ検出エラー判定

ENxTNCR<DECMD> 設定で方向指定時に、指定外の方向検出でエラーと判定します。エラー判定は割り込み要因となります。

- スキップ検出禁止 (ENxTNCR<SDTEN> = "0")

  CW 回転検出時(ENxTNCR<DECMD> = "01") ：CCW 回転エッジでエラー発生

  CCW 回転検出時(ENxTNCR<DECMD> = "10") ：CW 回転エッジでエラー発生
- スキップ検出許可 (ENxTNCR<SDETEN> = "1")

  CW 回転検出時(ENxTNCR<DECMD> = "01") ：CCW 方向スキップ,反転スキップおよび CCW 回転エッジでエラー発生

  CCW 回転検出時(ENxTNCR<DECMD> = "10") ：CW 方向スキップ,反転スキップおよび CW 回転エッジでエラー発生

18.5.2.5 バッファ更新制御

ENxTNCR<DECMD> = "00" に設定すると、バッファは常に有効となります。この場合、回転エッジ判定とスキップ判定は入力信号の変化で判定します。

<DECMD> を"00"以外に設定すると、バッファ更新は回転エッジ検出時のみとなります。このため、エッジ判定とスキップ判定はバッファが保持している、前の回転エッジ検出時の状態(ENxINPMON<DETMONA><DETMONB><DETMONZ>) と現在の入力状態 (ENxINPMON <SPLMONA><SPLMONB><SPLMONZ>)とで判定します。

18.5.2.6　BEMF 検出制御

センサモード(タイマカウント,位相カウント)の場合に PWM 同期サンプリング(ENxINPCR<SYNCSPLEN> = "1")にすると有効になり、回転エッジ判定検出を停止(中断)したり開始(再開)したりすることができます。

この制御は、モータ制御回路(PMD)を使って矩形波駆動しているブラシレス DC モータの誘起電圧(BEMF)から位置検出(位置センサレス) する場合に使用します。

1. 回転エッジ検出開始
   - コマンド動作：ENxINPCR<PDSTT> = "1" 書き込み
   - イベント動作：カウンタ回路からの INT コンペア一致

2. 回転エッジ検出停止
   - コマンド動作：ENxINPCR<PDSTP> = "1" 書き込み
   - イベント動作：回転エッジ検出

18.5.3　カウンタ

カウンタ回路はクロック生成部、カウンタ部、コンペア機能およびキャプチャ機能などで構成され、動作モードによって使用できる機能が異なります。

18.5.3.1　エンコーダモード、センサモード(イベントカウント)

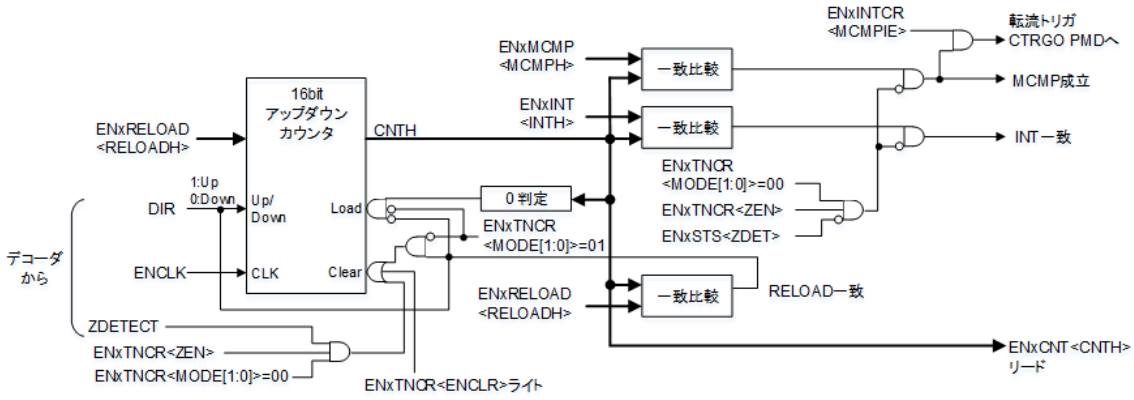


図 18-11 カウンタ構成(エンコーダモード、センサモード(イベントカウント))

デコーダからの回転エッジパルス(enclk)と回転方向信号(dir)で動作する 16 ビットアップダウンカウンタ、3 種類（<RELOADH>,<INTH>,<MCMPH>）のコンペア機能で構成されます。

エンコーダモードの場合、CW 回転検出時は RELOAD コンペアでカウンタクリアし、CCW 回転検出時はカウンタの"0"判定で <RELOADH> の値をカウンタにロードします。

エンコーダモードでZ検出許可（ENxTNCR<ZEN> = "1" ）の場合、エンコーダ入力許可（ENxTNCR<ENRUN> = "1"）してからZエッジ検出するまでのINTコンペアとMCMPコンペアの一致信号は無視されます。

カウンタレジスタ（ENxCNT）をリードするとアップダウンカウンタの値を読み出せます。ENxINTCR<MCMPIE> = "1" に設定するとMCMPコンペア一致信号をPMD回路の転流トリガに使用できます。

18.5.3.2 センサモード(タイマカウント)、タイマモード

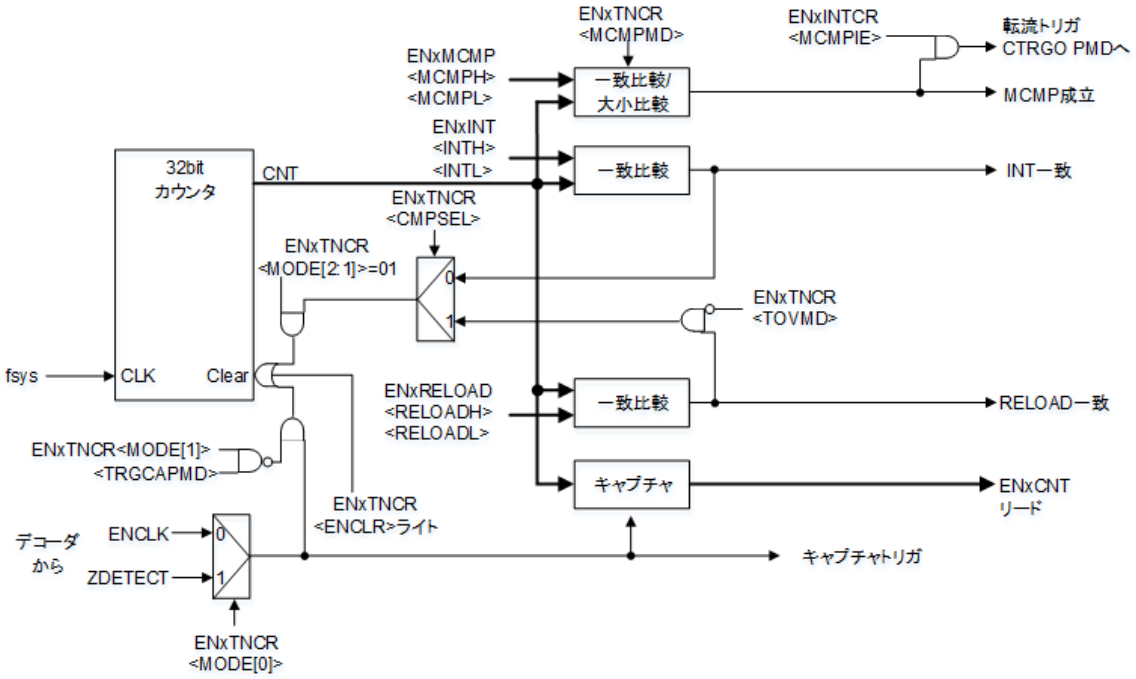


図 18-12 カウンタ構成(センサモード(タイマカウント)、タイマモード)

システムクロック(fsys)で動作する32ビットカウンタ、3種類（RELOAD,INT,MCMP）のコンペア機能、キャプチャ機能で構成されます。

MCMPコンペア機能は一致比較と大小比較を選択できます。大小比較（ENxTNCR <MCMPMD> = "1"）の場合は、ENxMCMPレジスタ設定することで比較を開始し、条件成立によりMCMP成立信号を出力して比較を終了します。

タイマモードでは、INTコンペアまたはRELOADコンペアでカウンタクリアできます。

センサモード（タイマカウント）の場合は回転エッジ検出（ENCLK）でカウンタキャプチャおよびカウンタクリアし、タイマモードの場合はZエッジ検出（ZDETECT）でカウンタキャプチャおよびカウンタクリアできます。カウンタレジスタ（ENxCNT）をリードするとキャプチャ値を読み出せます。

ENxINTCR<MCMPIE> = "1" に設定するとMCMPコンペア一致信号をPMD回路の転流トリガに使用できます。

18.5.3.3　センサモード(位相カウント)、位相カウンタモード

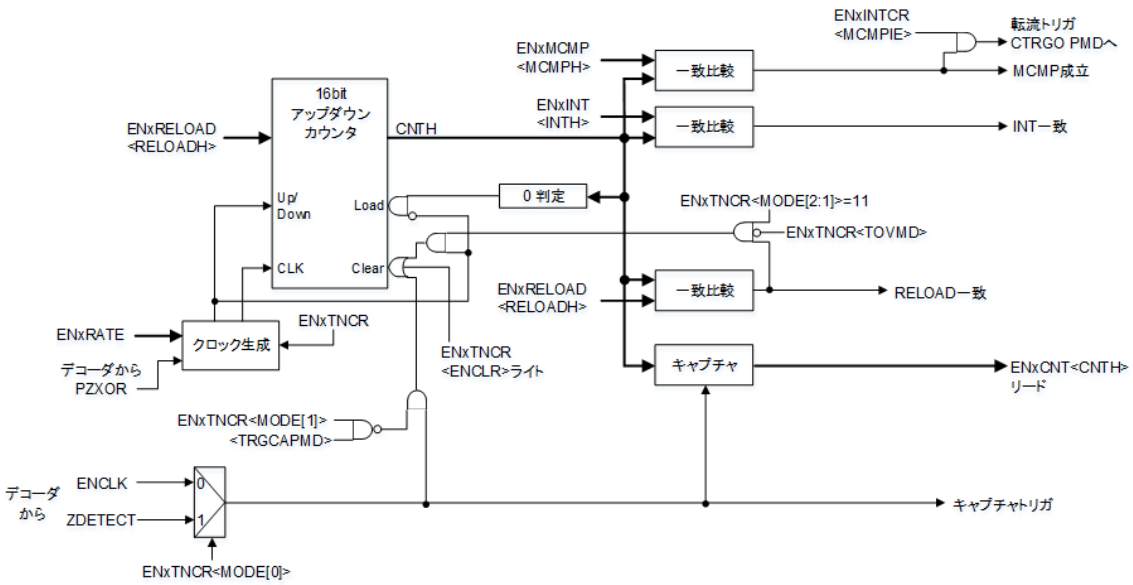


図 18-13 カウンタ構成(センサモード(位相カウント)、位相カウンタモード)

ENxRATE<RATE>の設定によりカウンタクロックを生成するクロック生成回路、クロック生成回路からのクロック信号と方向信号で動作する 16 ビットアップダウンカウンタ、<RELOADH>、<INTH>、<MCMPH> との 3 種類の一致比較器、キャプチャ機能で構成されます。

カウンタクロックは ENxRATE レジスタで任意に設定します。

アップダウンカウントは ENxTNCR<UDMD> で設定します。

アップカウント設定時は RELOAD コンペア一致でカウンタクリアし、ダウンカウント設定時は"0"一致で <RELOADH> の値をカウンタにロードします。

センサモード（位相カウント）の場合は回転エッジ検出（ENCLK）でカウンタキャプチャおよびカウンタクリアし、位相カウンタモードの場合はＺエッジ検出（ZDETECT）でカウンタキャプチャおよびカウンタクリアできます。カウンタレジスタ（ENxCNT）をリードするとキャプチャ値を読み出せます。

ENxINTCR<MCMPEN> = "1" に設定すると MCMP コンペア一致信号を PMD 回路の転流トリガに使用できます。

18.5.4 割り込み要求制御

6 種類の割り込み要因から 2 種類の割り込み要求を出力します。割り込み要求は割り込み制御レジスタ(ENxINTCR)で要因別に出力許可し、発生要因は割り込みフラグ(ENxINTF)で確認できます。

割り込みフラグレジスタ(ENxINTF)は割り込み要因の発生でセットされ、レジスタリードでクリアされます。

表 18-1 割り込み要因一覧

割り込み要因	説明	モード	割り込み要求出力許可 ENxINTCR	要因別フラグ ENxINTF	割り込み要求出力
分周パルス	回転エッジパルスを、ENxTNCR <ENDEV> の設定により 1 分周 ~ 128 分周して、通知します。	エンコーダモード センサモード(イベントカウント)	<TPLSIE>	<TPLSF>	INTENCx0
キャプチャ	外部トリガ(ENCZ 入力)でキャプチャが行われたことを通知します。	センサモード(位相カウント) 位相カウンタモード	<CAPIE>	<CAPF>	INTENCx0
	回転エッジパルス(ENCLK)でキャプチャが行われたことを通知します。	センサモード (タイマカウント) タイマモード			
検出エラー	エッジ検出エラー(PDERR)またはスキップ検出(SKPDT)の発生を通知します。	エンコーダモード センサモード	<ERRIE>	<ERRF>	INTENCx0
INT 一致	ENxINT レジスタとカウンタ値が一致したことを通知します。	全てのモード	<CMPIE>	<INTCPF>	INTENCx1
RELOAD 一致	ENxRELOAD レジスタとカウント値が一致したことを通知します。	センサモード(タイマカウント、位相カウント) タイマモード 位相カウンタモード	<RLDIE>	<RLDCPF>	INTENCx1
MCMP 成立	ENxTNCR<MCMPMD> = "0"の場合、ENxMCMP レジスタとカウンタ値が一致したことを通知し、<MCMPMD> = "1"の場合、カウント値が ENxMCMP レジスタ値以上になったことを通知します。	センサモード(タイマカウント) タイマモード	<MCMPIE>	<MCMPF>	INTENCx1
	ENxMCMP レジスタとカウンタ値が一致したことを通知します。	エンコーダモード センサモード(イベントカウント、位相カウント) 位相カウンタモード			

18.6　ブラシレス DC モータ制御例

PMD からの矩形波にて駆動しているブラシレス DC モータの、BEMF ゼロクロス検出する場合の波形を示します。

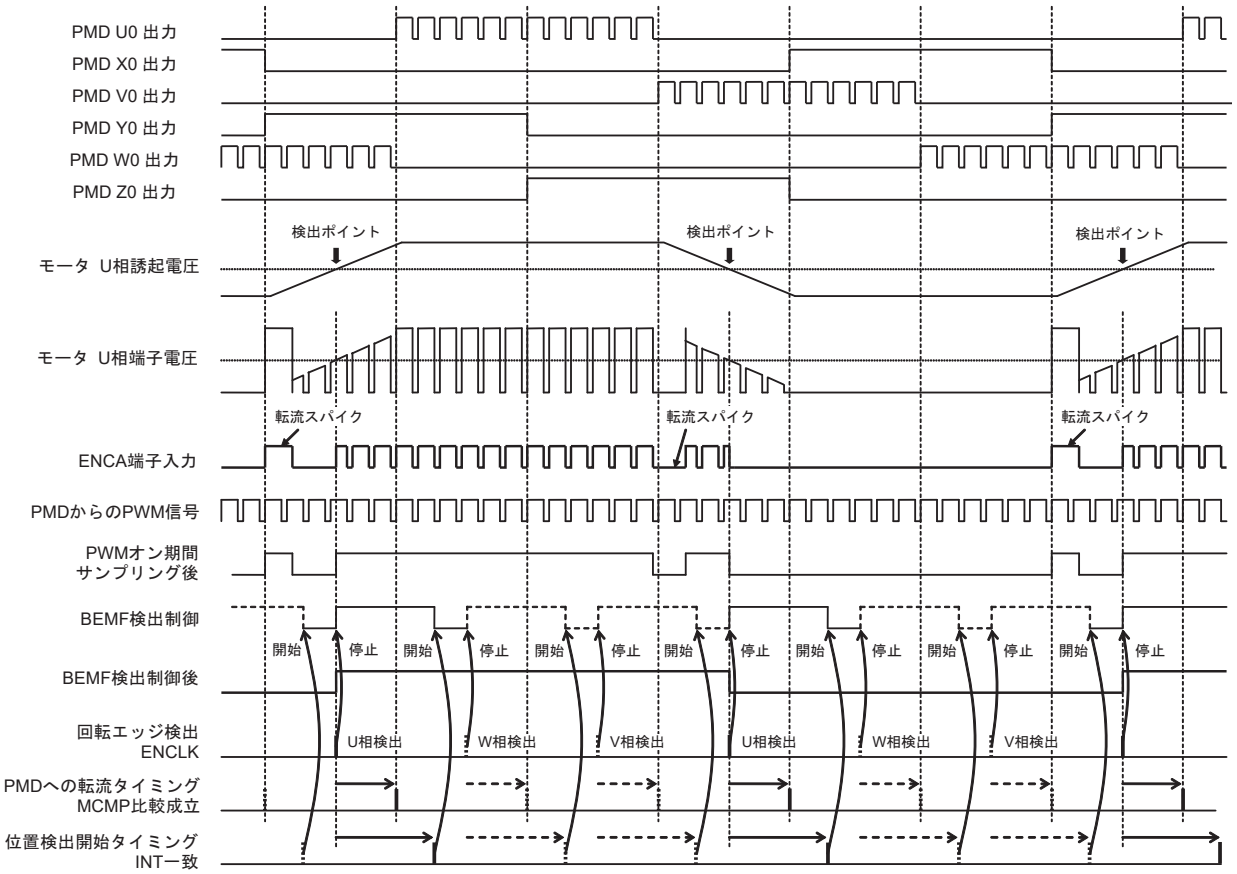


図 18-14 120 度通電時のセンサレス制御タイミング波形

センサモード(タイマカウント,位相カウント)で PWM 同期サンプリングを許可(ENxINPCR <SYNCSPLEN> = "1")すると PMD で矩形波駆動しているブラシレスＤＣモータの誘起電圧(BEMF)のゼロクロス検出に対応できます。この場合、入力回路は PWM 同期サンプリングで入力信号から PWM 成分を除去します。また、デコーダ回路は BEMF 検出制御で転流スパイクを回避できます。これらにより、入力信号から BEMF のゼロクロスを回転エッジパルス(ENCLK)として検出できます。

BEMF 検出制御は、回転エッジパルスでエッジ検出を停止(中断)し、INT コンペア成立でエッジ検出を開始(再開)します。また、コマンドストップ(ENxINPCR<PDSTP> = "1")、コマンドスタート(ENxINPCR<PDSTT> = "1")も可能です。

BEMF ゼロクロス検出は、カウンタのキャプチャ機能および３種類のコンペア機能を次のように使用して PMD 回路と連携動作します。PMD と連携動作するために、転流タイミング出力を有効(ENxINTCR<MCMPIE> = "1")にしてください。

- キャプチャ：回転エッジパルス(ENCLK)でキャプチャ(ゼロクロス検出間隔の測定)
- MCMP コンペア： PMD の転流タイミング(設定例：キャプチャ値×0.5)
- INT コンペア：位置検出開始タイミング(設定例：キャプチャ値×0.75)
- RELOAD コンペア：位置検出タイムアウト(設定例：キャプチャ値×2.0)

第 19 章　パワーオンリセット回路(POR)

パワーオンリセット回路(POR)は、電源投入時にパワーオンリセット信号を発生します。

電源電圧とは DVDD5 を、内部電源電圧とは VOUT12 を指しています。

注)　電源電圧の変動によりパワーオンリセット回路が完全な動作をしないことがあります。機器設計的には電気的特性を参照の上十分な考慮をしてください。

19.1　構成

パワーオンリセット回路は、基準電圧発生回路、コンパレータで構成されます。

電源電圧および内部電源電圧をラダー抵抗によって分圧した電圧それぞれを、基準電圧発生回路が発生した基準電圧とコンパレータで比較します。

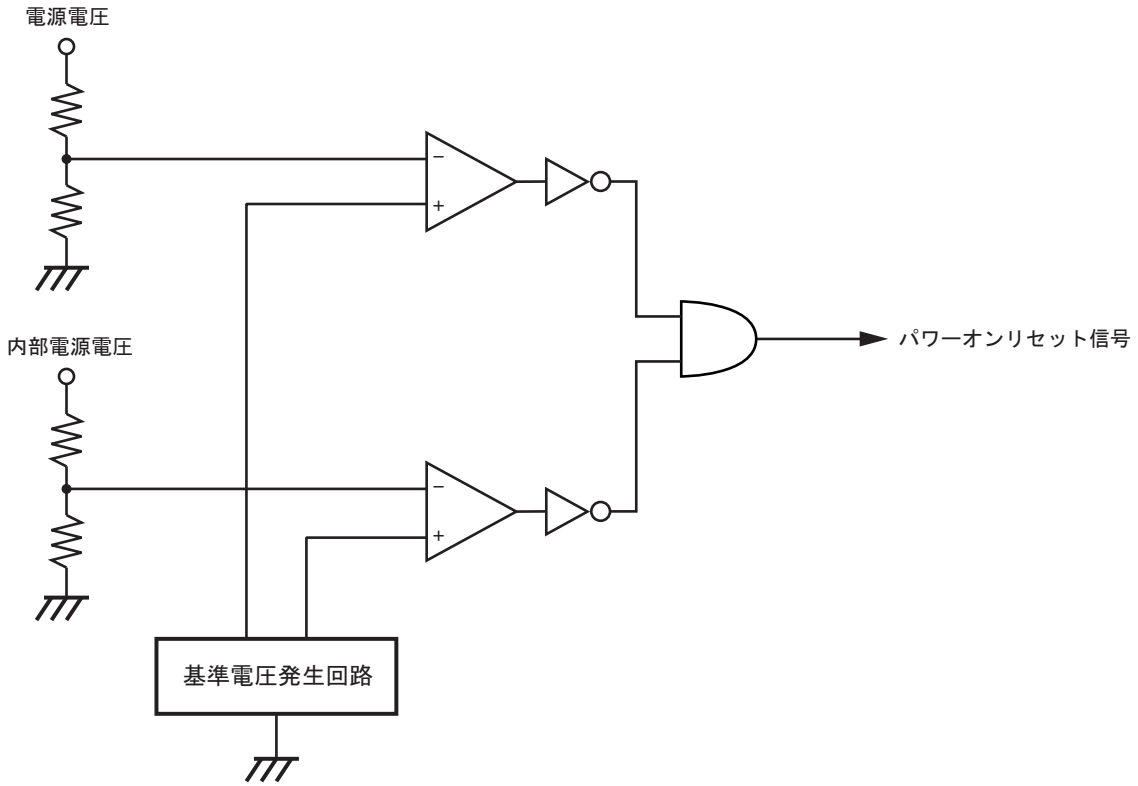


図 19-1 パワーオンリセット回路

19.2　機能

電源投入時、電源電圧が POR 解除電圧以下の間、パワーオンリセット信号は有効です。パワーオンリセット信号が無効となるのは、電源電圧が 3.0 ± 0.2 V を超えるタイミングです。

電源下降時、電源電圧が POR 検出電圧以上の間、パワーオンリセット信号は無効です。パワーオンリセット信号が有効となるのは、電源電圧が 2.8 ± 0.2 V を下回るタイミングです。

パワーオンリセット信号が有効の間、CPU および周辺機能はリセットされます。

リセットについては、リセット動作の章を参照してください。

また、動作タイミングの詳細については電源検出回路(VLTD)も参照してください。

注)　POR 解除電圧と POR 検出電圧は、相対的に変動するためこれらの電圧レベルが逆転することはありません。

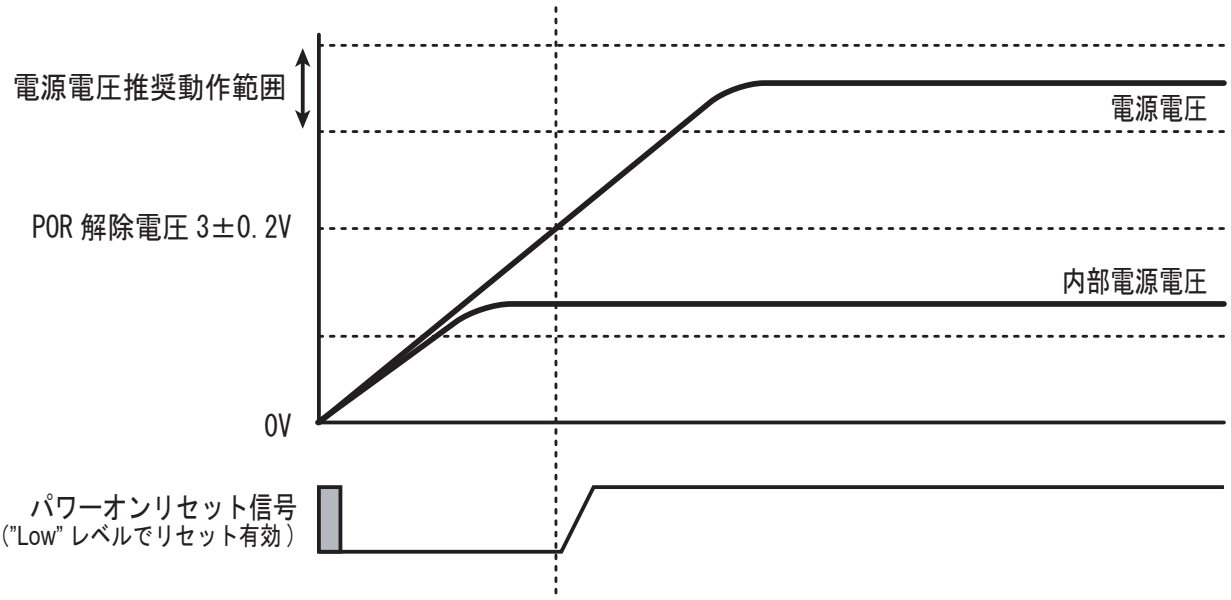


図 19-2 電源投入時のパワーオン動作タイミング

第 20 章　電圧検出回路(VLTD)

電圧検出回路(VLTD)は、電源電圧の低下を検出し、電圧検出リセット信号を発生します。

電源電圧とは、DVDD5 を指しています。

注)　電源電圧の変動によっては電圧検出回路が完全な動作をしないことがありますので、機器設計時には電気的特性を参照の上、十分な考慮が必要です。

20.1　構成

電圧検出回路は、基準電圧発生回路、検出電圧レベル選択回路、コンパレータ、制御レジスタから構成されます。

電源電圧 はラダー抵抗で分圧され、検出電圧選択回路に入力されます。検出電圧選択回路で検出電圧(VDLVL) に応じた電圧が選択され、コンパレータで基準電圧と比較されます。

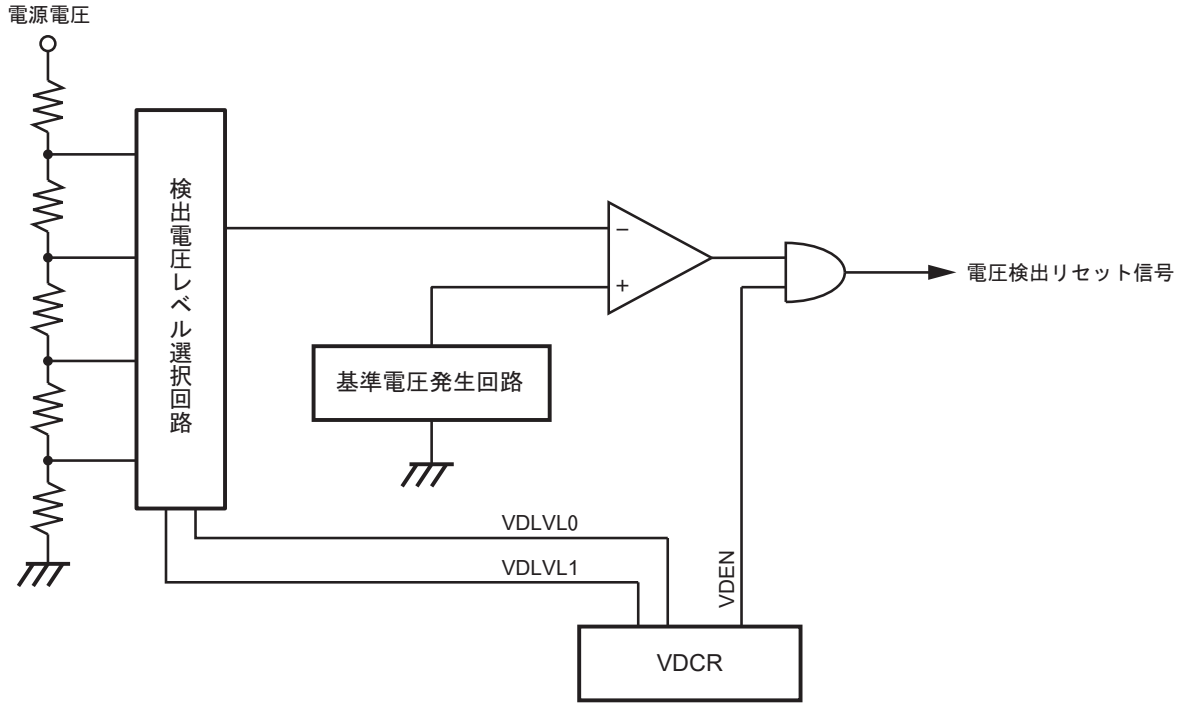


図 20-1 電圧検出回路

20.2　レジスタ説明

20.2.1　レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		Address(Base+)
電圧検出制御レジスタ	VDCR	0x0000

20.2.2　VDCR (電圧検出制御レジスタ)

	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	–	–	–	–	–	VDLVL		VDEN
リセット後	0	0	0	0	0	0	1	1

Bit	Bit Symbol	Type	機能
31-3	–	R	リードすると"0"が読めます。
2-1	VDLVL[1:0]	R/W	検出電圧レベル選択 00: Reserved 01: 4.1± 0.2V 10: 4.4 ± 0.2V 11: 4.6 ± 0.2V
0	VDEN	R/W	電圧検出の許可／禁止 0: 禁止 1: 許可

注)　VDCR はパワーオンリセット、外部リセット入力で初期化されます。

20.3 動作説明

20.3.1 制御

電圧検出回路は、電圧検出制御レジスタで制御されます。

20.3.2 機能

電圧検出回路は、検出電圧レベル選択ビット VDCR<VDLVL[1:0]> 、電圧検出の許可／禁止 VDCR<VDEN>で設定します。電圧検出の許可／禁止を設定し、電源電圧 が検出電圧 (VDLVL[1:0]) を下回ったとき、電圧検出リセット信号が有効になります。

リセットについては、リセット動作の章を参照してください。

20.3.2.1 電圧検出動作の許可／禁止

VDCR<VDEN> はパワーオンリセット、外部リセット解除後 、"1" にセットされて電圧検出動作が許可されます。

注) 電源電圧が 検出電圧レベル以下の状態で VDCR<VDEN> = "0" (禁止) から"1" (許可) に設定すると、設定した時点でリセット信号が発生します。

20.3.2.2 検出電圧レベル選択

VDCR<VDLVL[1:0]> で検出電圧を選択します。

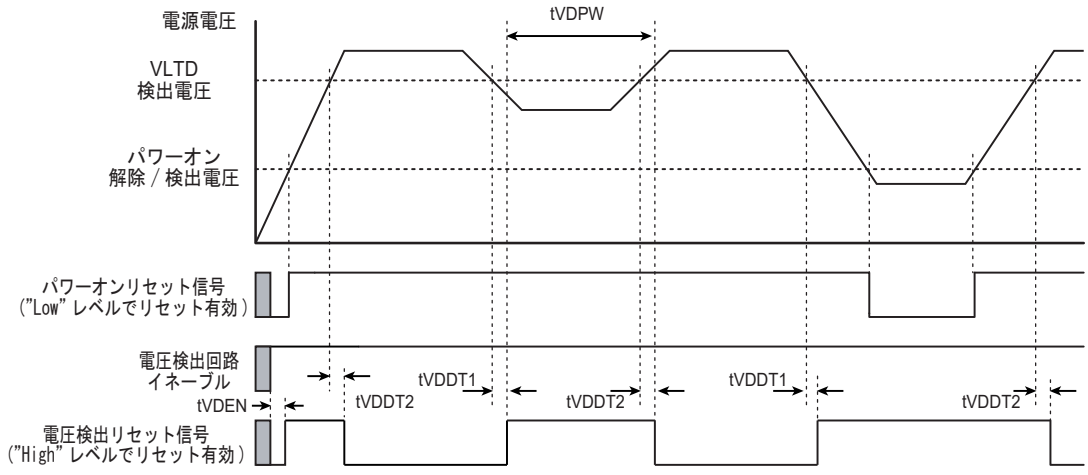


図 20-2 電圧検出タイミング

注) 内部リセットによって VLTD が許可となった後に、電源電圧が VLTD 解除電圧に達する場合です。

記号	項目	Min	Typ.	Max	単位
tVDEN	電圧検出回路が許可になるまでの時間	-	40	-	μs
tVDDT1	電圧検出回路のリセット有効応答時間	-	40	-	
tVDDT2	電圧検出回路のリセット無効応答時間	-	40	-	
tVDPW	電圧検出回路の検出最小パルス幅	45	-	-	

第 21 章　周波数検知回路(OFD)

周波数検知回路(OFD ; Oscillation Frequency Detector)はクロック周波数の異常を検出する回路です。OFD を使用することで、高調波、低調波、停止といったクロックの異常を検出することができます。

OFD は、基準となるクロックを用いて対象のクロックを観測し、異常を検出するとリセットを発生します。本製品の基準クロックは内蔵高速発振クロック(fiosc)、検出対象クロックは外部高速発振器クロック(feosc)です。

注)　いかなる異常も検知できるわけではありません。また、クロックの誤差を測定するための回路ではありません。

21.1　構成

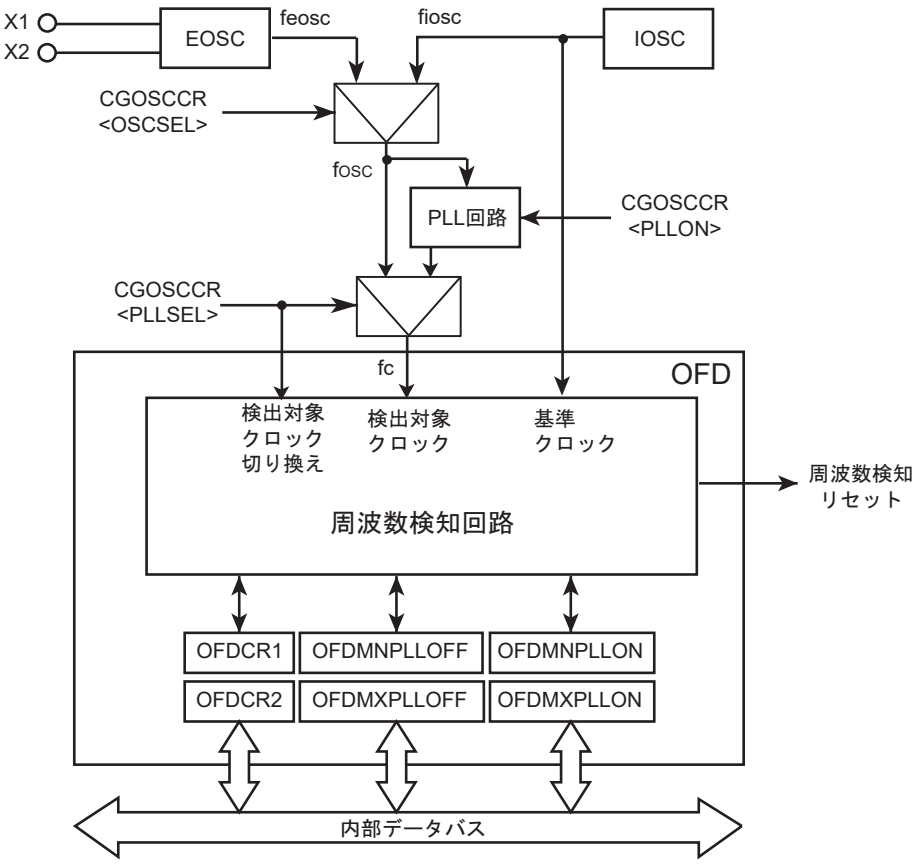


図 21-1 周波数検知回路ブロック図

21.2　レジスタ説明

21.2.1　レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

レジスタ名		Address(Base+)
周波数検知回路制御レジスタ 1	OFDCR1	0x0000
周波数検知回路制御レジスタ 2	OFDCR2	0x0004
検知周波数下限値レジスタ（PLL OFF 時）	OFDMNPLLOFF	0x0008
検知周波数下限値レジスタ（PLL ON 時）	OFDMNPLLON	0x000C
検知周波数上限値レジスタ（PLL OFF 時）	OFDMXPLLOFF	0x0010
検知周波数上限値レジスタ（PLL ON 時）	OFDMXPLLON	0x0014

21.2.1.1 OFDCR1(周波数検知回路制御レジスタ 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	OFDWEN							
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	OFDWEN[7:0]	R/W	レジスタ書き込み制御 0x06: 禁止 0xF9: 許可 0xF9 を設定すると、OFDCR1 以外のレジスタへの書き込みができるようになります。 0x06、0xF9 以外の値を書いた場合、0x06 が書かれます。 書き込みが禁止されていても、各レジスタを読み出すことは可能です。

注) OFDCR1 は外部リセット($\overline{\text{RESET}}$ 端子への"L"入力)、パワーオンリセット、または VLTD リセットで初期化されます。

21.2.1.2　OFDCR2(周波数検知回路制御レジスタ 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	OFDEN							
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	OFDEN[7:0]	R/W	周波数検知動作制御 0x00: 禁止 0xE4: 許可 0x00、0xE4 以外の値の書き込みは無効で値は変化しません。

注)　OFDCR2 は外部リセット($\overline{\text{RESET}}$ 端子への"L"入力)、パワーオンリセット、または VLTD リセットで初期化されます。

21.2.1.3 OFDMNPLLOFF (検知周波数下限値設定レジスタ(PLL OFF 時))

	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	-	-	-	-	-	-	-	OFDMNPLLOFF
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	OFDMNPLLOFF							
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。
8-0	OFDMNPLLOFF[8:0]	R/W	検知周波数の下限値を設定します。

注 1) 本レジスタは、周波数検知動作が許可されているときは書き込みできません。

注 2) OFDMNPLLOFF は外部リセット($\overline{\text{RESET}}$ 端子への"L"入力)、パワーオンリセット、または VLTD リセットで初期化されます。

21.2.1.4　OFDMNPLLON (検知周波数下限値設定レジスタ(PLL ON 時))

	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	-	-	-	-	-	-	-	OFDMNPLLON
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	OFDMNPLLON							
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。
8-0	OFDMNPLLON [8:0]	R/W	検知周波数の下限値を設定します。

注 1)　本レジスタは、周波数検知動作が許可されているときは書き込みできません。

注 2)　OFDMNPLLON は外部リセット($\overline{\text{RESET}}$ 端子への"L"入力)、パワーオンリセット、または VLTD リセットで初期化されます。

21.2.1.5 OFDMXPLLOFF (検知周波数上限値設定レジスタ(PLL OFF 時))

	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	-	-	-	-	-	-	-	OFDMXPLLOFF
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	OFDMXPLLOFF							
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。
8-0	OFDMXPLLOFF[8:0]	R/W	検知周波数の上限値を設定します。

注 1) 本レジスタは、周波数検知動作が許可されているときは書き込みできません。

注 2) OFDMXPLLOFF は外部リセット($\overline{\text{RESET}}$ 端子への"L"入力)、パワーオンリセット、または VLTD リセットで初期化されます。

21.2.1.6　OFDMXPLLON (検知周波数上限値設定レジスタ(PLL ON 時))

	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	-	-	-	-	-	-	-	OFDMXPLLON
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	OFDMXPLLON							
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。
8-0	OFDMXPLLON [8:0]	R/W	検知周波数の上限値を設定します。

注 1)　本レジスタは、周波数検知動作が許可されているときは書き込みできません。

注 2)　OFDMXPLLON は外部リセット($\overline{\text{RESET}}$ 端子への"L"入力)、パワーオンリセット、または VLTD リセットで初期化されます。

21.3 動作説明

21.3.1 設定

外部リセット入力($\overline{\text{RESET}}$ 端子への"L"入力)、パワーオンリセットまたは VLTD リセットにより周波数検知回路(OFD)の制御レジスタは初期化され、OFDCR1 以外のレジスタには書き込みができない状態になります。OFDCR1 に"0xF9"を書き込むことにより、OFDCR1 以外のレジスタに書き込みができるようになります。

検知する周波数の範囲は、対象クロックごとに OFDMNPLLON と OFDMXPLLON 並びに OFDMNPLLOFF と OFDMXPLLOFF で設定します。これらは CGPLLSEL<PLLSEL> 設定により自動的に切り換わります。なお、PLL オンの状態 CGPLLSEL<PLLSEL> = "1"で周波数検知リセットが発生した場合、PLL がオフ状態 CGPLLSEL<PLLSEL> = "0"となり、検知周波数設定レジスタは OFDMNPLLON と OFDMXPLLON から OFDMNPLLOFF と OFDMXPLLOFF に自動的に切り換ります。

OFDCR1 に書き込み許可コード "0xF9" が設定された状態で OFDCR2 に "0xE4" を書き込むと OFD はイネーブルとなり動作します。

誤書き込み防止のため、すべてのレジスタの設定後、OFDCR1 に"0x06"を設定し書込み禁止にしてください。また、設定の変更は動作が停止した状態で行ってください。

周波数検知動作がイネーブル状態(OFDCR2 = "0xE4")で STOP モードを起動した場合、OFD は自動的にディセーブルとなります。この状態で STOP モードが解除されると OFD は STOP モード解除後のウォーミングアップ期間が終了してからイネーブルとなります。

注) CGPLLSEL レジスタにて PLL を設定する場合、および CGOSCCR<OSCSEL>レジスタで内部 fiosc または外部 feosc にシステムクロックを切り換える場合、OFD は必ずディセーブルの状態で行なってください。

21.3.2 使用可能な動作モード

周波数検知回路(OFD)は NORMAL モードと IDLE モードのときのみ使用可能です。詳細は表 21-1 を参照してください。

内部高周波発振に遷移する際は、OFD を停止させてください。

表 21-1 各動作モードにおける周波数検知回路の状態

動作モードまたは状態	周波数検知回路動作 (OFDCR2 = "0xE4" 設定時)	周波数検知リセットによる端子状態 (電源、$\overline{\text{RESET}}$、MODE、X1,X2 端子を除く)
NORMAL	動作	ハイインピーダンス
IDLE	動作	ハイインピーダンス
STOP (ウォーミングアップ期間含む)	周波数検知回路は自動的にディセーブルとなります。	
周波数検知によるリセット	動作	ハイインピーダンス
ウォッチドッグタイマリセット SYSRESETREQ リセット	動作	ハイインピーダンス
外部リセット入力 ($\overline{\text{RESET}}$ 端子への"L"入力) パワーオンリセット VLTD リセット	停止	-

21.3.3　動作

周波数検知回路(OFD)では、動作開始から検知開始まで検知周期 2 周期分の時間が必要で、検知周期は 128/基準クロック周波数です。

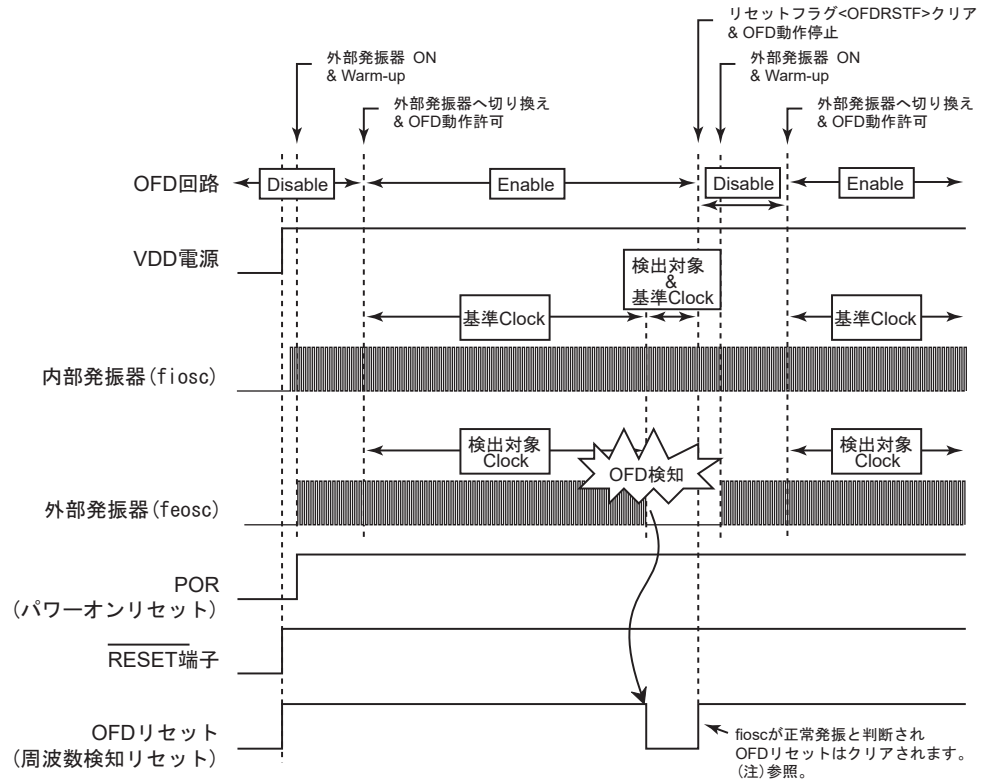
検出対象クロックが OFDMNPLLON、OFDMNPLLOFF と OFDMXPLLON、OFDMXPLLOFF で設定した周波数範囲を超えると OFD はリセットを発生します。OFD が異常を検知してからリセットを発生するまで検知周期 1 周期分の時間が必要です。OFD が発生するリセットでは OFD 自身はリセットされず検知動作を継続します。

OFD が発生するリセットにより fosc は内部高速発振クロック fiosc に初期化され、検知対象クロック fc は PLL OFF 時の fiosc に切り換わるため、検知対象クロックと基準クロックが同じになって周波数範囲が設定値以内に納まるので、周波数検知リセットは解除されます。

なお、システムクロック fc として内部高速発振クロック fiosc を選択した場合は、OFD は機能しません。

注 1)　リセットの要因は複数あります。クロックジェネレータレジスタの CGRSTFLG で要因を確認することができます。CGRSTFLG については例外の章を参照してください。

注 2)　検出対象クロックの設定値（OFDMNPLLOFF、OFDMXPLLOFF）を 10MHz 以外、例えば 8MHz の設定値にして OFD を動作させ周波数検知リセットが掛かった場合、対象クロックが異常と判断し、周波数検知リセットを発生し続ける場合があります。



(注) 検出対象クロックの設定値（OFDMNPLLOFF、OFDMXPLLOFF）を10MHz 以外、例えば8MHz の設定値にして検知回路を動作させ周波数検知リセットが掛かった場合、対象クロックが異常と判断し、周波数検知リセットを発生し続ける場合があります。

図 21-2 周波数検知回路の動作例

21.3.4 検知周波数

検知周波数には、基準クロックの発振精度によって、検出範囲と非検出範囲が存在します。検出範囲と非検出範囲の間の周波数では、検出されるかどうかは未確定です。

検知周波数上限値、下限値は、検出対象クロックと基準クロックの最大誤差より計算します。

OFDMNPLLON、OFDMNPLLOFF と OFDMXPLLON、OFDMXPLLOFF を決める時の計算結果の丸め方により、検出範囲と非検出範囲の上限値、下限値が下記のように変わります。検出対象となるクロックのばらつきに応じて選択してください。

- OFDMXPLLON/OFDMXPLLOFF を切上げ、OFDMNPLLON/OFDMNPLLOFF を切捨てた場合

  検出対象クロックに対して、非検出範囲の上限が高くなり、非検出範囲の下限が低くなります。
- OFDMXPLLON/OFDMXPLLOFF を切捨て、OFDMNPLLON/OFDMNPLLOFF を切上げた場合

  検出対象クロックに対して、非検出範囲の上限が低くなり、非検出範囲の下限が高くなります。

基準クロック誤差を±5%、検出対象クロック誤差±5%(非検出範囲)を許容する場合の OFDMXPLLOFF/OFDMNPLLOFF の設定値の算出方法を以下に示します。この例では、OFDMXPLLOFF を切上げ、OFDMNPLLOFF を切り捨てます。(①～⑧は、「図 21-3 検出周波数範囲例(10MHz の場合)」に対応)

検出対象クロック	10MHz ± 5%	Max 10.5MHz	-------- ③
		Min 9.5MHz	-------- ②
基準クロック	9.7MHz ± 5%	Max 10.185MHz	-------- ⑥
		Min 9.215MHz	-------- ⑤

OFDMXPLLOFF = ③ ÷ ⑤ × 32 = 36.46...= 37 (小数点以下切上げ) = 0x25

OFDMNPLLOFF = ② ÷ ⑥ × 32 = 29.85... = 29 (小数点以下切捨て) = 0x1D

このときの検出範囲は以下のように求められます。

① = ⑤ × OFDMNPLLOFF ÷ 32 = 8.35

④ = ⑥ × OFDMXPLLOFF ÷ 32 = 11.78

また、このときの非検出範囲は以下のようになります。

⑦ = ⑤ × OFDMXPLLOFF ÷ 32 = 10.65

⑧ = ⑥ × OFDMNPLLOFF ÷ 32 = 9.23

よって、レジスタ OFDMXPLLOFF に 0x25、OFDMNPLLOFF に 0x1D を設定すると、11.78MHz 以上と 8.35MHz 以下の発振周波数を検知した場合、周波数検知リセットを発生します。

また、9.23MHz から 10.65MHz の発振周波数を検知した場合、周波数検知リセットは発生しません。

図 21-3 に、このときの検出範囲/非検出範囲を示します。

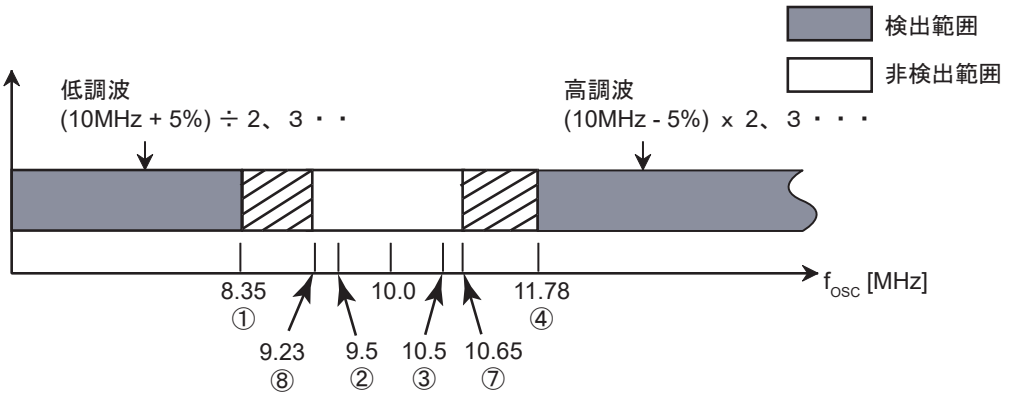


図 21-3 検出周波数範囲例(10MHz の場合)

21.3.5　動作手順例

周波数検知回路(OFD)を使用する場合の動作手順例を示します。

リセット発生後、CGRSTFLG でリセットの要因を確認します。要因が周波数検知リセットでなければ、外部発振を有効にし、OFD を使用するためのレジスタ設定を行い動作を許可します。

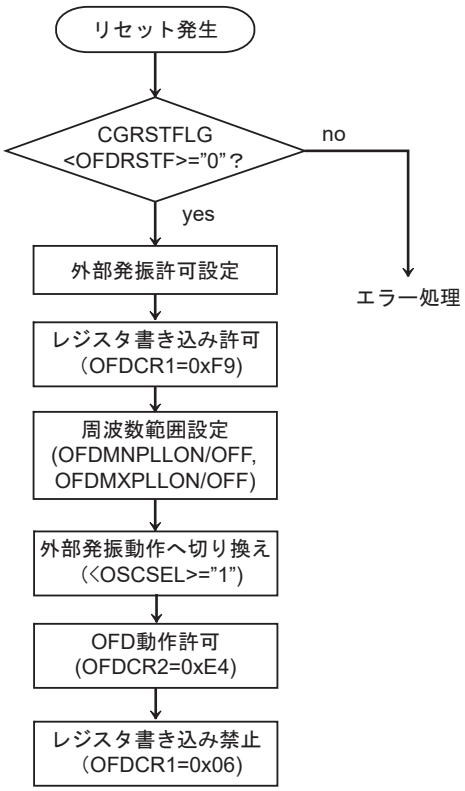


図 21-4 動作手順例

第 22 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($\overline{\text{WDTOUT}}$)より"Low"を出力して暴走の検出を知らせます。

注)　TMPM470FDFG/FZFG/FYFG にはウォッチドッグタイマアウト端子($\overline{\text{WDTOUT}}$)はありません。

22.1　構成

図 22-1 にウォッチドッグタイマのブロック図を示します。

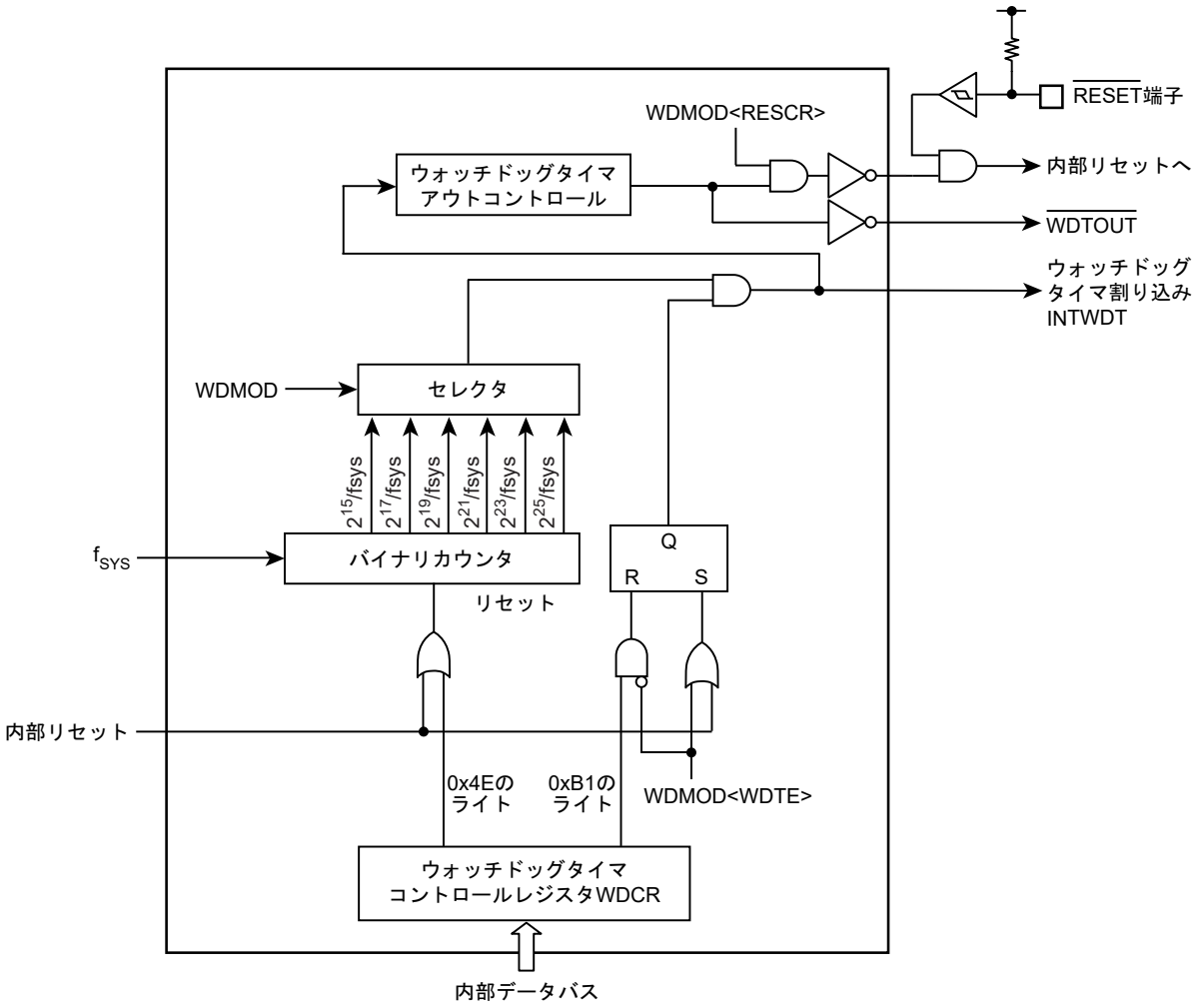


図 22-1 ウォッチドッグタイマのブロック図

22.2　レジスタ

22.2.1　レジスター覧

制御レジスタとアドレスは以下のとおりです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照ください。

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

22.2.1.1　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 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。
6-4	WDTP[2:0]	R/W	検出時間の選択 000: $2^{15}/f_{SYS}$　100: $2^{23}/f_{SYS}$ 001: $2^{17}/f_{SYS}$　101: $2^{25}/f_{SYS}$ 010: $2^{19}/f_{SYS}$　110: 設定禁止 011: $2^{21}/f_{SYS}$　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。

22.2.1.2 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

22.3　動作説明

22.3.1　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($\overline{\text{WDTOUT}}$)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、ソフトウエア(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)　TMPM470FDFG/FZFG/FYFG にはウォッチドッグタイマアウト端子($\overline{\text{WDTOUT}}$)はありません。

22.3.2　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では使用できません。以下に示すモードに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。

- STOP mode

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。

22.3.3　暴走検出時の動作

22.3.3.1　INTWDT 割り込み発生の場合

図 22-2 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。

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)　TMPM470FDFG/FZFG/FYFG にはウォッチドッグタイマアウト端子($\overline{\text{WDTOUT}}$)はありません。

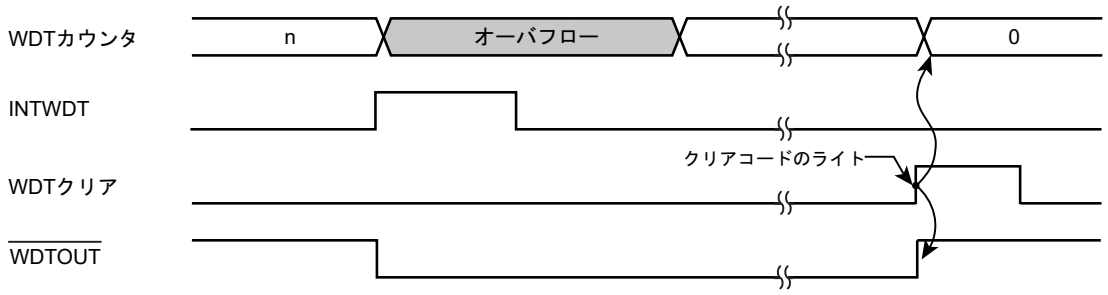


図 22-2 INTWDT 割り込み発生

22.3.3.2 内部リセット発生の場合

図 22-3 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。

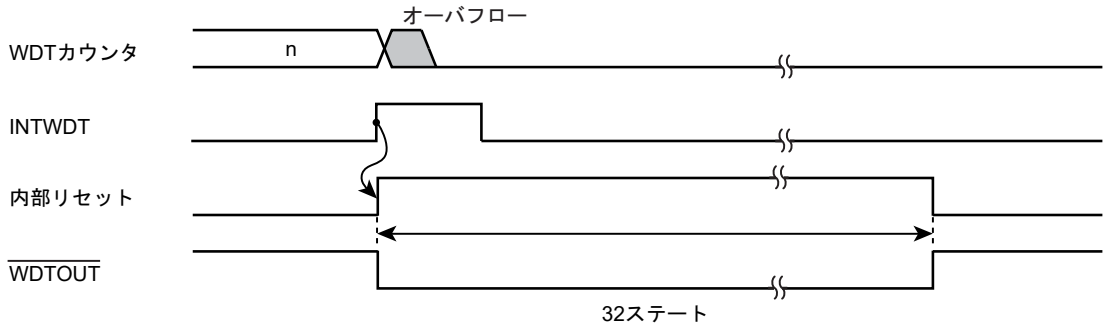


図 22-3 内部リセット発生

22.4 ウォッチドックタイマの制御

22.4.1 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。

22.4.2 イネーブル制御

WDMOD<WDTE>に"1"を設定します。

22.4.3 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。

22.4.4 ウォッチドッグタイマ検出時間の設定

検出時間を WDMOD<WDTP[2:0]>に設定します。

例えば、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。

第 23 章　フラッシュメモリ(FLASH)

フラッシュメモリについて、構成およびその動作を説明します。本文中の「1 ワード」は、32 ビットをあらわします。

23.1　フラッシュメモリの特長

23.1.1　メモリ容量と構成

本製品の内蔵するフラッシュメモリの容量と構成は、表 23-1 と表 23-2 の通りです。

表 23-1 メモリ容量と構成

製品	容量 (KB)	エリア情報		ブロック情報		ページ情報		書き込み時間(s) (注)	消去時間(ms) (注)			
		サイズ (KB)	個数	サイズ (KB)	個数	サイズ (Byte)	個数		1 ページ	1 ブロック	1 エリア	チップ
TMPM470FDFG	512	512	1	32	16	4096	128	8.0	115	920	115	115
TMPM470FZFG	384	384	1		12		96	6.0				
TMPM470FYFG	256	256	1		8		64	4.0				

注)　上記の時間は各レジスタがリセット後の初期値の場合を表しており、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の書き替え方法により異なります。

表 23-2 ブロック構成

エリア番号	Block 番号	アドレス (シングルチップモード)	アドレス (シングルブートモード) (シングルチップモード(ミラー))	サイズ (KByte)	ページ数	使用可能な領域		
						TMPM470FDFG	TMPM470FZFG	TMPM470FYFG
0	0	0x0000_0000 ~ 0x0000_7FFF	0x5E00_0000 ~ 0x5E00_7FFF	32	8	*	*	*
	1	0x0000_8000 ~ 0x0000_FFFF	0x5E00_8000 ~ 0x5E00_FFFF	32	8	*	*	*
	2	0x0001_0000 ~ 0x0001_7FFF	0x5E01_0000 ~ 0x5E01_7FFF	32	8	*	*	*
	3	0x0001_8000 ~ 0x0001_FFFF	0x5E01_8000 ~ 0x5E01_FFFF	32	8	*	*	*
	4	0x0002_0000 ~ 0x0002_7FFF	0x5E02_0000 ~ 0x5E02_7FFF	32	8	*	*	*
	5	0x0002_8000 ~ 0x0002_FFFF	0x5E02_8000 ~ 0x5E02_FFFF	32	8	*	*	*
	6	0x0003_0000 ~ 0x0003_7FFF	0x5E03_0000 ~ 0x5E03_7FFF	32	8	*	*	*
	7	0x0003_8000 ~ 0x0003_FFFF	0x5E03_8000 ~ 0x5E03_FFFF	32	8	*	*	*
	8	0x0004_0000 ~ 0x0004_7FFF	0x5E04_0000 ~ 0x5E04_7FFF	32	8	*	*	
	9	0x0004_8000 ~ 0x0004_FFFF	0x5E04_8000 ~ 0x5E04_FFFF	32	8	*	*	
	10	0x0005_0000 ~ 0x0005_7FFF	0x5E05_0000 ~ 0x5E05_7FFF	32	8	*	*	
	11	0x0005_8000 ~ 0x0005_FFFF	0x5E05_8000 ~ 0x5E05_FFFF	32	8	*	*	
	12	0x0006_0000 ~ 0x0006_7FFF	0x5E06_0000 ~ 0x5E06_7FFF	32	8	*		
	13	0x0006_8000 ~ 0x0006_FFFF	0x5E06_8000 ~ 0x5E06_FFFF	32	8	*		
	14	0x0007_0000 ~ 0x0007_7FFF	0x5E07_0000 ~ 0x5E07_7FFF	32	8	*		
	15	0x0007_8000 ~ 0x0007_FFFF	0x5E07_8000 ~ 0x5E07_FFFF	32	8	*		

フラッシュメモリ構成の単位として、「エリア」、「ブロック」、「ページ」があります。

- ページ

  消去機能、プロテクト機能で使用します。

  1 ページは 4096 バイト固定です。
- ブロック

  消去機能、プロテクト機能で使用します。

  1 ブロックは 32K バイト固定です。
- エリア

  消去機能で使用します。

  1 エリアは 512K バイトです。

書き込みは 16 バイト単位(4 バイト x4 回)で行います。16 バイトあたりの書き込み時間は 163μs (Typ.)です。

消去はページ単位、ブロック単位、エリア単位またはフラッシュメモリ全体で行います。消去時間は使用するコマンドによって異なります。自動ブロック消去コマンドを使用した場合は 1 ブロックあたり 920 ms (Typ.)、それ以外のコマンドを使用した場合は 115 ms (Typ.)です。

プロテクトの設定は、ページ 0～7 はページ単位で行い、残りのブロックはブロック単位で行います。プロテクト設定の消去は、一括消去で行います。プロテクト機能については「23.1.5 プロテクト/セキュリティ機能」を参照してください。

23.1.2　機能

本製品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、本製品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、 消去動作を容易に実現できます。

JEDEC 準拠の機能	変更, 追加, 削除した機能
・自動プログラム ・自動チップ消去 ・自動ブロック消去 ・データポーリング/トグルビット	<追加>自動エリア消去、自動ページ消去、自動メモリスワップ <変更>ライト/消去プロテクト(ソフトウエアプロテクトのみサポート) <削除>消去レジューム/サスペンド機能

23.1.3　動作モード

フラッシュメモリの書き込み中と消去中は、電源オフやリセットを禁止します。

23.1.3.1　モードの説明

本製品には、シングルチップモードとシングルブートモードがあり、シングルチップモードにはノーマルモード、ユーザブートモードがあります。モード遷移図を図 23-1 に示します。

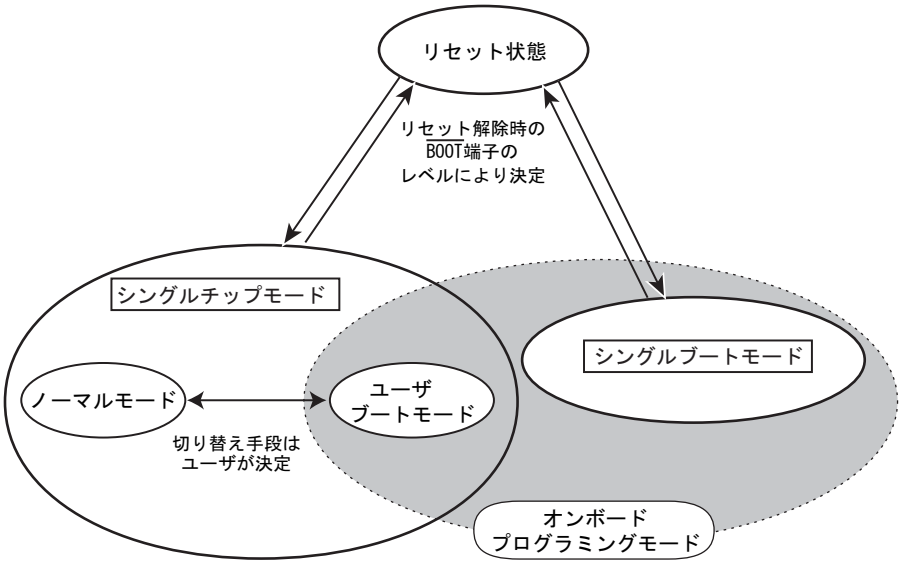


図 23-1 モード遷移図

(1) シングルチップモード

リセット解除後、フラッシュメモリから起動するモードで、以下の 3 つのモードがあります。

- ノーマルモード

  ユーザのアプリケーションプログラムを実行するモードです。

- ユーザブートモード

  ユーザのセット上でフラッシュメモリ外に置くフラッシュメモリの書き替えプログラムにてフラッシュメモリの書き替えを実行するモードです。

  フラッシュメモリの書き替え方法は「23.4 ユーザブートモードによる書き替え方法」を参照してください。

各モードの切り替えはユーザが独自に設定できます。例えばポートA の PA0 が"1" のときノーマルモード、"0" のときにユーザ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。

(2) 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) から起動するモードです。

内蔵メモリなど、フラッシュメモリ外に置くフラッシュメモリの書き替えプログラムにてフラッシュメモリの書き替えを実行することができます。フラッシュメモリの書き替え方法は「23.3 シングルブートモードによる書き替え方法」を参照してください。

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のセット上で書き換えを行う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フラッシュメモリの書き替えが実行できます。

(3)　オンボードプログラミングモード

ユーザのセット上でフラッシュメモリの書き替えが可能なモードは、ユーザブートモードとシングルブートモードです。これらをオンボードプログラミングモードと定義します。

23.1.3.2　モードの決定

シングルチップ、シングルブートの各動作モードは、リセットを解除するときの $\overline{\text{BOOT}}$ 端子の状態により決定されます。

リセットの動作については、「リセットの動作」の章を参照してください。

(1)　ウォームリセット時のモードの決定

表 23-3 ウォームリセット時の動作モード設定表

動作モード	端子	
	$\overline{\text{RESET}}$	$\overline{\text{BOOT}}$
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

(2)　コールドリセット時のモードの決定

表 23-4 コールドリセット時の動作モード設定表($\overline{\text{RESET}}$ 端子を使用しない場合)

動作モード	POR 引き伸ばし信号	端子
		$\overline{\text{BOOT}}$
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

- パワーオンリセット回路によるモード決定($\overline{\text{RESET}}$ 端子を使用しない場合)

  パワーオンカウンタによる POR 引き伸ばし信号が"0" から"1" になるまで、$\overline{\text{BOOT}}$ 端子の設定を継続してください。

- $\overline{\text{RESET}}$ 端子によるモード決定

  POR 引き伸ばし信号が"1" の時は $\overline{\text{RESET}}$ 端子によるリセットが可能となるので、この場合、表 23-3 と同じ設定を実施してください。

  また、POR 引き伸ばし信号が"0"の時は表 23-4 と同じ設定を実施してください。この場合、$\overline{\text{RESET}}$ 端子よりも POR 引き伸ばし信号によるリセットが優先となります。

23.1.4 メモリマップ

図 23-2 および図 23-3 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、フラッシュメモリは 0x5E00_0000 番地からマッピングされます。また、シングルブートモードでは 0x0000_0000 番地から 0x0000_0FFF 番地には BOOT ROM がマッピングされます。

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	RAM アドレス
TMPM470FDFG	512KB	34KB	0x0000_0000 ~ 0x0007_FFFF(シングルチップモード) 0x5E00_0000 ~ 0x5E07_FFFF(シングルチップモード(ミラー)) 0x5E00_0000 ~ 0x5E07_FFFF(シングルブートモード)	0x2000_0000 ~ 0x2000_87FF
TMPM470FZFG	384KB	34KB	0x0000_0000 ~ 0x0005_FFFF(シングルチップモード) 0x5E00_0000 ~ 0x5E05_FFFF(シングルチップモード(ミラー)) 0x5E00_0000 ~ 0x5E05_FFFF(シングルブートモード)	0x2000_0000 ~ 0x2000_87FF
TMPM470FYFG	256KB	18KB	0x0000_0000 ~ 0x0003_FFFF(シングルチップモード) 0x5E00_0000 ~ 0x5E03_FFFF(シングルチップモード(ミラー)) 0x5E00_0000 ~ 0x5E03_FFFF(シングルブートモード)	0x2000_0000 ~ 0x2000_3FFF, 0x2000_8000 ~ 0x2000_87FF

注) TMPM470FZFG、TMPM470FYFG には、ID やパスワード用共通エリア(0x5E07_FFF0 ~ 0x5E07_FFFF)が存在します。

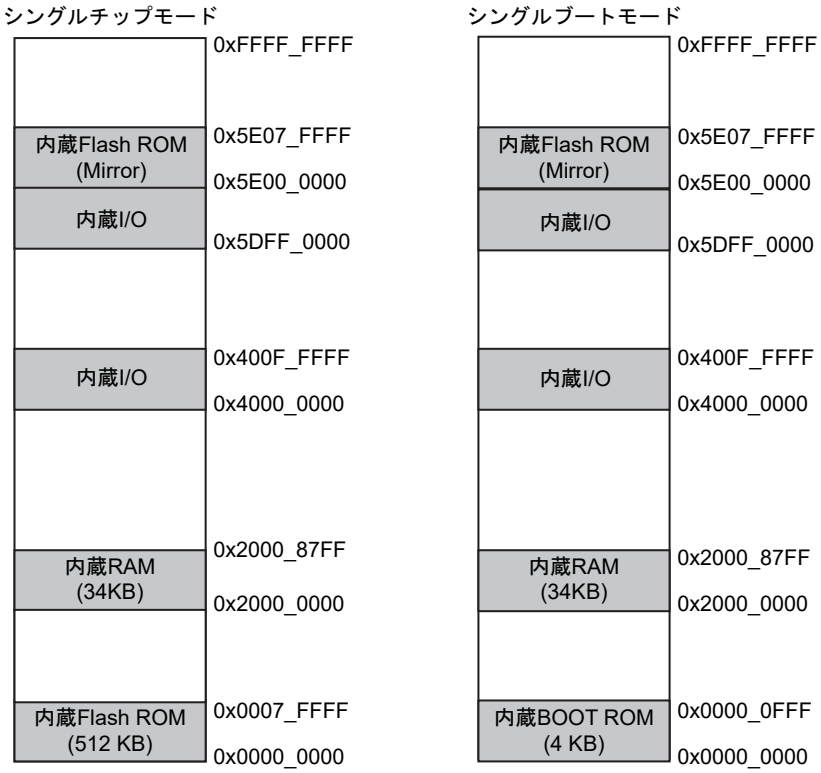


図 23-2 メモリマップの比較(TMPM470FDFG)

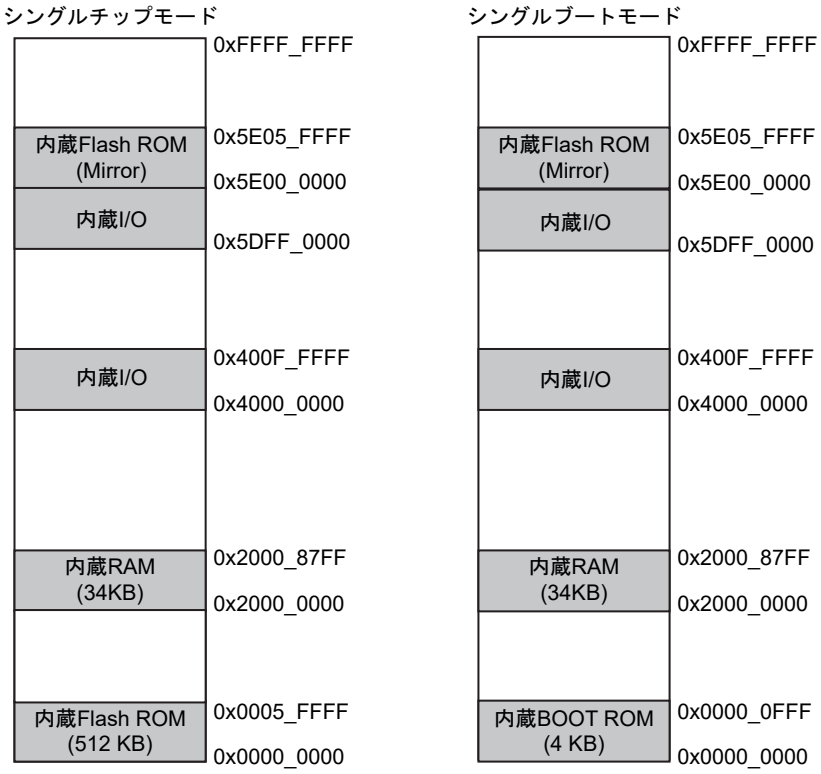


図 23-3 メモリマップの比較(TMPM470FZFG)

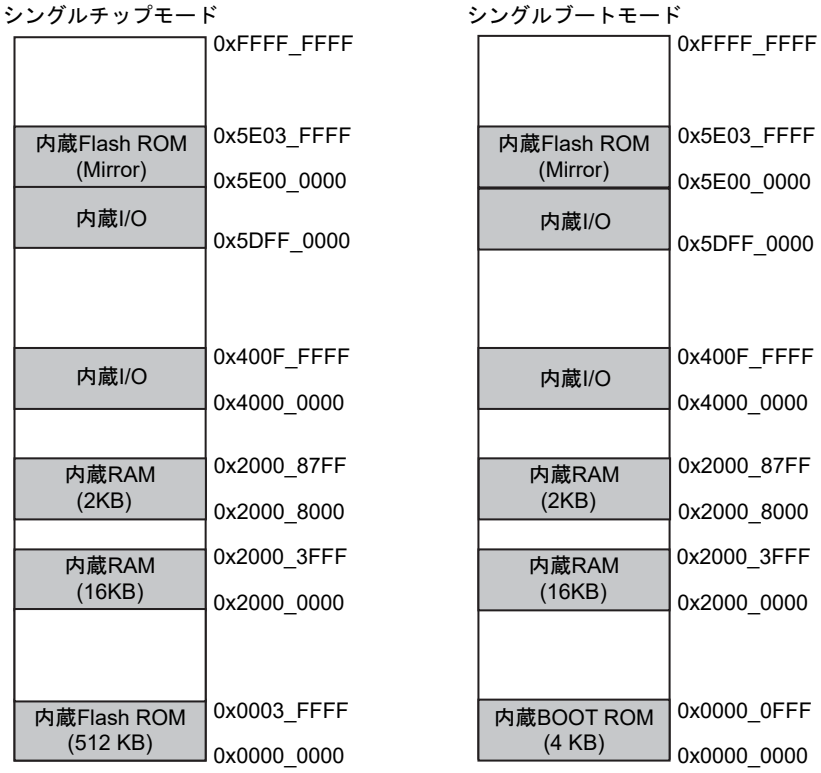


図 23-4 メモリマップの比較(TMPM470FYFG)

23.1.5 プロテクト/セキュリティ機能

本製品は、フラッシュメモリに対して、プロテクトとセキュリティの 2 つの機能を持っています。

1. プロテクト機能

   フラッシュメモリへの書き込み、消去を禁止

2. セキュリティ機能

   フラッシュライタによるフラッシュメモリの読み出しの禁止

   デバッグ機能の使用制限

23.1.5.1 プロテクト機能

Block0 はページ単位とブロック単位で、Block1 から最終ブロックまではブロック単位でプロテクト機能を使用することができます。

プロテクト機能を有効にするためには、自動プロテクトビットプログラムコマンドを用いて対応するプロテクトビットを"1"にします。プロテクトビット消去コマンドによりすべてのプロテクトビットを"0"にし、すべてのプロテクトは解除されます。プロテクトビットは、各 FCPSR レジスタでそれぞれモニタすることができます。

プロテクトビットのプログラムと消去の方法については、「23.2.6 コマンド説明」の章を参照してください。

23.1.5.2 セキュリティ機能

セキュリティ機能が有効な場合の動作を、表 23-5 に示します。

表 23-5 セキュリティ機能が有効な場合の動作

項目	内容
フラッシュメモリの読み出し	CPU からの読み出しは可能です。
デバッグポート	シリアルワイヤ、トレースの通信ができなくなります。
フラッシュメモリ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。

セキュリティ機能が有効になる条件は以下のとおりです。

1. FCSECBIT<SECBIT>が"1"にセットされている。
2. すべてのプロテクトビット(各 FCPSR レジスタのすべてのビット)が"1"にセットされている。

FCSECBIT<SECBIT>はコールドリセット 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。

注) 以下の 1., 2.の書き込みは 32bit 転送命令で行ってください。

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) FCSECBIT<SECBIT>="0" を実行すると、プロテクトビットが "0" にマスクされ、書き込みが可能となります。なお、プロテクトビットそのものに変更はありません。(セキュリティ＆プロテクトが解除された状態)

表 23-6 SECBIT とプロテクトビット

	<SECBIT>=1	<SECBIT>=0
プロテクトビット <All>=1	セキュリティ状態	TOOL 接続 /Read/Write 可
プロテクトビット <any>=1	TOOL 接続可 ブロックライトプロテクト状態	TOOL 接続 /Read/Write 可
プロテクトビット <All>=0	TOOL 接続 /Read/Write 可	TOOL 接続 /Read/Write 可

23.1.6　メモリスワップ機能

23.1.6.1　概要

フラッシュメモリの書き替え操作の途中で電源が OFF になった場合、例えばプログラム消去後の電源が OFF になり、書き込みができなくなるケースが考えられます。このようなケースを回避するために、本機能を利用して書き込みプログラムを残すことができます。

23.1.6.2　動作説明

スワップ領域は 0 番地で始まる領域と次の領域で、スワップサイズは FCSWPSR<SIZE>で決まります。このサイズを変更するには、自動メモリスワップコマンドにて FCSWPSR<SIZE>のビットを"1"にセットします。

メモリスワップを行うには、自動メモリスワップコマンドにて FCSWPSR[0]に"1"を設定します。スワップ状態を解除するには、自動メモリスワップコマンドにて FCSWPSR[1]に"1"を設定します。スワップ状態は FCSWPSR<SWP>にて確認することができます。

自動メモリスワップコマンドの詳細は「23.2.6 コマンド説明」の章を参照してください。

23.1.6.3　操作方法

メモリスワップ操作の基本的な流れを以下に示します。メモリスワップ操作の具体例は「23.5 ユーザブートプログラムの書き替え方法」を参照してください。

1. セキュリティ機能が有効の場合は、セキュリティを解除してください。

   セキュリティの解除方法は「23.1.5.2 セキュリティ機能」を参照してください。

   セキュリティを解除しない場合、手順におけるコマンド実行にてフラッシュメモリが消去されます。

2. プロテクト機能が有効の場合は、プロテクトビットを消去してください。

   プロテクトビットの消去方法は「23.1.5.1 プロテクト機能」を参照してください。

   プロテクトビットを消去しない場合、手順におけるコマンド実行が行われません。

3. 0 番地で始まる領域の次の領域がブランク状態であることを確認します。(以後、0 番地で始まる領域を Page0、次の領域を Page1 として説明します。)ブランク状態でなければ消去してください。

   Page0：旧オリジナルデータ

   Page1：ブランク

4. 0 番地で始まる領域のオリジナルデータを次の領域にも書き込みます。(両方の領域のデータを同じにします)

   Page0：旧オリジナルデータ

   Page1：コピーデータ(旧オリジナルデータ)

5. メモリスワップを行います。

   Page0：コピーデータ(旧オリジナルデータ)

   Page1：旧オリジナルデータ

6. 旧オリジナルデータを消去して、ブランク状態にします。
   Page0：コピーデータ(旧オリジナルデータ)
   Page1：ブランク

7. ブランク領域に新しいデータを書き込みます。
   Page0：コピーデータ(旧オリジナルデータ)
   Page1：新オリジナルデータ

8. スワップ状態を解除します。
   Page0：新オリジナルデータ
   Page1：コピーデータ(旧オリジナルデータ)

9. 自動プロテクトビット消去コマンドを実行します。

10. 必要により以下を行ってください。
    - コピーデータ(旧オリジナルデータ)消去。
    - スワップ領域以外のフラッシュメモリのデータ書き換え。
    - プロテクト機能の有効化
    - セキュリティ機能の有効化

手順		3	4	5	6	7	8
内蔵 RAM		消去ルーチン	書き替えルーチン	スワップルーチン	消去ルーチン	書き替えルーチン	スワップルーチン
フラッシュメモリ	Page0	旧オリジナル	旧オリジナル	旧オリジナルのコピー	旧オリジナルのコピー	旧オリジナルのコピー	新オリジナル
	Page1	ブランク	旧オリジナルのコピー	旧オリジナル	ブランク	新オリジナル	旧オリジナルのコピー

消去ルーチン: フラッシュメモリの消去を行うためのプログラム
書き替えルーチン: フラッシュメモリの書き替えを行うためのプログラム
スワップルーチン: フラッシュメモリのスワップを行うためのプログラム

23.1.7　レジスタ

23.1.7.1　レジスター覧

制御レジスタとアドレスは以下の通りです。

ベースアドレスは、「メモリマップ」章の「周辺機能ベースアドレス一覧」を参照下さい。

周辺機能:FC

レジスタ名		Address(Base+)
セキュリティビットレジスタ	FCSECBIT	0x0010
プロテクトステータスレジスタ 0	FCPSR0	0x0020
ステータスレジスタ	FCSR	0x0100
スワップステータスレジスタ	FCSWPSR	0x0104
エリア選択レジスタ	FCAREASEL	0x0140
コントロールレジスタ	FCCR	0x0148
ステータスクリアレジスタ	FCSTSCLR	0x014C
WCLK 設定レジスタ	FCWCLKCR	0x0150
Program 用カウンタ設定レジスタ	FCPROGCR	0x0154
Erase 用カウンタ設定レジスタ	FCERASECR	0x0158

23.1.7.2 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。

23.1.7.3　FCPSR0(プロテクトステータスレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	BLK15	BLK14	BLK13	BLK12	BLK11	BLK10	BLK9	BLK8
リセット後	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)
	23	22	21	20	19	18	17	16
bit symbol	BLK7	BLK6	BLK5	BLK4	BLK3	BLK2	BLK1	-
リセット後	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	0
	15	14	13	12	11	10	9	8
bit symbol	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
リセット後	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)	(注 1)
	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
31-17	BLK15 ~ BLK1	R	Block1 ~ 15 のプロテクト状態 0: プロテクト状態ではない 1: プロテクト状態 プロテクト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。
16	-	R	リードすると"0"が読めます。
15-8	PG7 ~ PG0	R	Page0 ~ 7 のプロテクト状態 1: プロテクト状態 0: プロテクト状態ではない プロテクトビット値は各ページのプロテクト状態に対応します。該当ビットが"1"の時は対応するページがプロテクト状態であることを示します。プロテクト状態のページは書き換えはできません。
7-1	-	R	リードすると"0"が読めます。
0	RDY_BSY	R	自動プログラムまたは自動チップ消去コマンド実行時の Ready/Busy (注 2) 0: 自動動作中 1:自動動作終了 自動動作の状態を認識できます。フラッシュメモリが自動動作中は "0" になり、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。

注 1)　プロテクト状態に応じた値になります。
注 2)　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。

23.1.7.4 FCSR(ステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	WEABORT
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	-	-	-	-	-	-	-	-
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	WEABORT	R	自動動作の中止が行われると"1"がセットされます。 詳細は「23.2.4 自動動作の中止」を参照してください。
23-0	−	R	リードすると"0x000001"が読めます。

23.1.7.5　FCSWPSR(スワップ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	-	-	-	-	-	SIZE		
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	FLG						SWP	
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-11	–	R	リードすると"0"が読めます。
10-8	SIZE[2:0]	R	スワップサイズ 000: 4K バイト(ページ 0　⇔ページ 1) 001: 8K バイト(ページ 0-1　⇔ページ 2-3) 010: 16K バイト(ページ 0-3　⇔ページ 4-7) 011: 32K バイト(ブロック 0　⇔ブロック 1) 上記以外: 設定禁止
7-2	FLG	R	ソフトウエアの管理用フラグ(「23.5 ユーザブートプログラムの書き替え方法」を参考にして下さい)
1-0	SWP[1:0]	R	スワップの状態 11: スワップ解除 10: 設定禁止 01:スワップ中 00: スワップ解除(初期化状態)

注)　本レジスタは、自動プロテクトビット消去コマンドで初期化されます。

23.1.7.6 FCAREASEL(エリア選択レジスタ)

	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	-	-	-	-	-	AREA0		
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	AREA0	R/W	フラッシュメモリ操作コマンドにより実行の対象となるフラッシュメモリの"エリア"を指定します。 111: エリア 0 を選択 上記以外: エリア 0 を非選択

注 1) 本レジスタに値を設定する場合は、本レジスタに値をライトし、その後ライトした値がリードできることを確認してください。

注 2) エリア選択ビットが"111(0x7)"以外のままフラッシュメモリ操作コマンドを実行するとコマンド実行がキャンセルされます。

注 3) 自動チップ消去コマンドを実行する場合はすべてのエリア選択ビットに"111(0x7)"を設定してください。

23.1.7.7　FCCR(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	-	-	-	-	-	WEABORT		
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	WEABORT	R/W	自動動作コマンドの中止 111:中止する リードすると設定値が読めます。 詳細は「23.2.4 自動動作の中止」を参照してください。

23.1.7.8 FCSTSCLR(ステータスクリアレジスタ)

	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	-	-	-	-	-	WEABORT		
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	WEABORT	R/W	FCSR<WEABORT>の"0"クリア。 111:クリアする リードすると設定値が読めます。 詳細は「23.2.4 自動動作の中止」を参照してください。

23.1.7.9　FCWCLKCR(WCLK 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	-	-	-	DIV				
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	DIV	R/W	自動動作中のクロック(WCLK : fsys/(DIV+1))が 8 ~ 12MHz となる分周比 00000: 1 分周 00001: 2 分周 : 11110: 31 分周 11111: 32 分周

注 1)　<DIV>を設定する前にギア切り替え処理が終了していることを確認してください。

注 2)　ギアを切り替えて Flash 操作を行う場合、WCLK が 8 ~ 12MHz の範囲内となるよう動作周波数(fsys)に応じて再設定を行ってください。表 23-6 に主な動作周波数(fsys)と<DIV>値の例を示します。

表 23-6 動作周波数(fsys)と<DIV>に対する主な自動動作中のクロック

	fsys	10MHz	20MHz	25MHz	30MHz	40MHz	80MHz	100MHz	120MHz
<DIV>	分周値	自動動作中のクロック(MHz)							
00000	1 分周	10	-	-	-	-	-	-	-
00001	2 分周	-	10	-	-	-	-	-	-
00010	3 分周	-	-	8.3	10	-	-	-	-
00011	4 分周	-	-	-	-	10	-	-	-
00111	8 分周	-	-	-	-	-	10	-	-
01001	10 分周	-	-	-	-	-	8	10	12
01011	12 分周	-	-	-	-	-	-	8.3	10
01110	15 分周	-	-	-	-	-	-	-	8

-: 設定できません

23.1.7.10 FCPROGCR(Program 用カウンタ設定レジスタ)

	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	-	-	-	-	-	-	CNT	
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。
1-0	CNT	R/W	自動プログラム実行コマンドによる書き込み時間(CNT/WCLK)が 20 ~ 40μs となるカウント数 00:カウント数 250 01: カウント数 300 上記以外:カウント数 350

注) WCLKCR<DIV>を再設定する場合、自動プログラム実行コマンドによる書き込み時間が 20 ~ 40μs の範囲内となるよう必要に応じて再設定を行ってください。表 23-7 に主な WCLK 値と<CNT>値の例を示します。

表 23-7 WCLK と<CNT>に対する主な書き込み時間

	WCLK	8.33MHz	10MHz	12MHz
<CNT>	カウント数	書き込み時間(μs)		
00	250 回	30.0	25.0	20.8
01	300 回	36.0	30.0	25.0
上記以外	350 回	-	35.0	29.2

- : 設定できません

23.1.7.11　FCERASECR(Erase 用カウンタ設定レジスタ)

	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	-	-	-	-	CNT			
リセット後	0	0	0	0	0	1	1	0

Bit	Bit Symbol	Type	機能
31- 4	-	R	リードすると"0"が読めます。
3-0	CNT	R/W	各自動消去コマンド実行による消去時間(CNT/WCLK)が 100 ~ 130ms となるカウント数 0000:カウント数 850000 0001: カウント数 900000 0010: カウント数 950000 0011: カウント数 1000000 0100: カウント数 1050000 0101: カウント数 1100000 0110: カウント数 1150000 0111: カウント数 1200000 1000: カウント数 1250000 1001: カウント数 1300000 1010: カウント数 1350000 上記以外: カウント数 1400000

注)　WCLKCR<DIV>を再設定する場合、各自動消去コマンドによる消去時間が 100 ~ 130ms の範囲内となるよう必要に応じて再設定を行ってください。表 23-8 に主な WCLK 値と<CNT>値の例を示します。

表 23-8 WCLK と<CNT>に対する主な消去時間

	WCLK	8.33MHz	10MHz	12MHz
<CNT>	カウント数	消去時間(ms)		
0000	850000 回	102.0	-	-
0001	900000 回	108.0	-	-
0010	950000 回	114.0	-	-
0011	1000000 回	120.0	-	-
0100	1050000 回	126.1	105.0	-
0101	1100000 回	-	110.0	-
0110	1150000 回	-	115.0	-
0111	1200000 回	-	120.0	-
1000	1250000 回	-	125.0	104.2
1001	1300000 回	-	-	108.3
1010	1350000 回	-	-	112.5
上記以外	1400000 回	-	-	116.7

-: 設定できません

23.2　フラッシュメモリ詳細

オンボードプログラミングモードでは、制御プログラム中で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があらかじめ用意しておきます。

またエリア 0 のフラッシュメモリ上でプログラムを実行中に、命令実行を行っていない他の"エリア"(例えばエリア 1)のフラッシュメモリを書き込み/消去できます(逆も可能です)。

23.2.1　機能

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が、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(1 ワード)のストア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 23-9 フラッシュメモリの機能

主な機能	説明
自動プログラム	4 ワード(16 バイト)でデータ書き込みを自動で行います。
自動チップ消去	フラッシュメモリの全領域の一括消去を自動で行います。
自動エリア消去	エリア単位での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。
自動ページ消去	ページ単位での消去を自動で行います。
ライト/消去プロテクト	書き込みおよび消去を禁止することができます。
自動メモリスワップ	フラッシュメモリ領域のスワップ/スワップ解除/スワップサイズ指定を自動で行います。

23.2.2　フラッシュメモリの動作モード

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)

電源投入後、リセット解除後、および自動動作の終了時にフラッシュメモリはリードモードになります。フラッシュメモリに書かれた命令の実行、およびデータの読み出しはリードモードで行います。

リードモード中にコマンドを入力すると自動動作へ移り、コマンドの処理が終了するとリードモードに戻ります(ID-Read コマンドを除く)。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。コマンドの実行方法については「23.2.3 コマンド実行方法」を参照してください。

23.2.3　コマンド実行方法

コマンド実行は、ストア命令を用いてフラッシュメモリに対してコマンドシーケンスを書き込むことで行います。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。コマンド実行の詳細は、「23.2.6 コマンド説明」を参照してください。

フラッシュメモリに対するストア命令の実行を"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ライトサイクルで構成されています。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実行された時はコマンドの自動動作を実施します。規定の順番で実行されなかった場合、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。

コマンドシーケンスの途中で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。Read/リセットコマンドをソフトウエアリセットと呼びます。

コマンドシーケンスの書き込みが終了すると自動動作を開始し、FCPSR0<RDY_BSY> = "0"になります。自動動作が終了すると FCPSR0<RDY_BSY> = "1" となり、リードモードに復帰します。

自動動作中は、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「23.2.4 自動動作の中止」を参照してください。

コマンドを実行する際には以下の事項に留意してください。

1. 自動動作中は以下の操作を行わないでください。
   - すべての例外発生

2. コマンドシーケンサがコマンドを認識するために、コマンド開始前の状態がリードモードである必要があります。各コマンドシーケンスの第 1 バスライトサイクル前に FCPSR0<RDY_BSY> ="1"であることを確認してください。続いて Read/リセットコマンドを実行することを推奨します。

3. コマンドシーケンスは、内蔵 RAM 上で実行してください。

4. 各コマンドを実行する前に FCAREASEL レジスタのエリア選択ビットを設定(<AREAn> に”111”(0x07)をライト）してください。

5. 各バスライトサイクルは連続して、1 ワード(32 ビット)のデータ転送命令で行います。

6. 各コマンドシーケンスの実行中に、実行対象となるフラッシュメモリへのアクセスを行うとバスフォールトを発生します。

7. 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、必ずソフトウエアリセットを発行して、リードモードに戻してください。

8. 各コマンド実行の終了確認手順は以下の通りです。

   1)最終バスライトサイクルを実行します。

   2)FCPSR0<RDY_BSY>="0"(Busy)となるまでポーリングします。

   3)FCPSR0<RDY_BSY>="1"(Ready)となるまでポーリングします。

9. フラッシュメモリからデータをリードする場合は、FCAREASEL レジスタのエリア選択ビットをクリア(<AREAn>に"000"(0x0)）してください。

23.2.4 自動動作の中止

自動動作を中止する手順を以下に示します。

1. FCPSR0<RDY_BSY>をリードします。

2. 手順 1 のリード結果が"1"(Ready)の場合は、手順 9 で終了してください。"0"(Busy)の場合は、手順 3 へ進みます。

3. FCCR<WEABORT>に"0x7"をライトします。

4. FCCR<WEABORT>に"0x0"をライトします。
5. FCPSR0<RDY_BSY>="1"(Ready)となるまでポーリングします。
6. FCSR<WEABORT>をリードします。
7. Read/リセットコマンドを実行します。
8. 手順 6 のリード結果が"0"の場合(注)は、手順 9 で終了してください。"1"の場合は、以下の操作を行い、このフラグをクリアします。
   (1) FCSTSCLR<WEABORT>に"0x7"をライトします。
   (2) FCSTSCLR<WEABORT>に"0x0"をライトします。
   (3) FCSR<WEABORT>="0"となるまでポーリングします。
9. 終了

注)　手順 3 より先に FCPSR0<RDY_BSY>="1"となった場合です。

23.2.5　自動動作の完了検知

Flash の書込み/消去動作等の完了を検知する割り込み機能があります。

23.2.5.1　手順

自動動作の完了検知割り込みを使用する手順は以下の通りです。

割り込み処理の詳細については、例外の章の”割り込み”を参照して下さい。

1. Flash に対し書込み/消去コマンドを発行後、FCPSR0<RDY_BSY>で、自動動作中(BUSY 状態）を確認します。自動動作中を確認したら、CPU 割り込みの許可の設定をします。
2. Flash の自動動作終了後、INTFLRDY 割り込みが発生
3. INTFLRDY 割り込み処理ルーチンの中で、CPU 割り込みの禁止をしてください。

23.2.6　コマンド説明

各コマンドの内容について説明します。具体的なコマンドシーケンスは「23.2.7 コマンドシーケンス」を参照してください。

23.2.6.1　自動プログラム

(1)　動作内容

自動プログラムコマンドシーケンスにより、4 ワード(16 バイト)単位で書き込みができます。16 バイトを跨ってデータを書き込むことはできません。

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。

自動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 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、自動ページ消去、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プログラムを実行しなおす必要があります。

自動プログラム中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、該当となるデータの書き込みは正常に行われていない可能性があるため、消去動作後に改めて自動プログラムを実行する必要があります。

注 1) 消去動作を伴わない同一ページへの 2 回以上プログラム実施はデータ破損の可能性があります。
注 2) プロテクトされたブロックへの書き込みはできません。

(2) 実行方法

第 1～第 3 バスライトサイクルが自動プログラムのコマンドシーケンスです。

第 4 バスライトサイクルでページの先頭アドレスとデータを書き込みます。第 5 バスライトサイクル以降は 4 ワードの内の残りのデータを書き込みます。データは 1 ワード(32 ビット)単位で書き込んでください。

16 バイトの一部に書き込みを行う場合、書き込みが不要なアドレスのデータを"0xFFFFFFFF"として 16 バイト分の書き込みを行ってください。

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。

23.2.6.2 自動チップ消去

(1) 動作内容

自動チップ消去は、全アドレスのメモリセルに対して消去動作を行います。プロテクトされているページまたはブロックがあ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入力後にリードモードに戻ります。

自動消去中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、データの消去は正常に行われていない可能性があるため、改めて自動消去を実行する必要があります。

(2) 実行方法

第 1～第 6 バスライトサイクルが自動チップ消去のコマンドシーケンスです。コマンドシーケンス入力後、自動チップ消去動作を行います。

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。

23.2.6.3 自動エリア消去

(1) 動作内容

自動エリア消去コマンドは、指定された"エリア"に対して消去動作を行います。プロテクトされているページまたはブロックがある場合は自動エリア消去を実行せず、コマンドシーケンスの入力後にリードモードに戻ります。

自動消去中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、データの消去は正常に行われていない可能性があるため、改めて自動消去を実行する必要があります。

(2)　実行方法

第 1～第 5 バスライトサイクルが自動エリア消去のコマンドシーケンスです。第 6 バスライトサイクルで消去する"エリア"を指定します。コマンドシーケンス入力後、自動エリア消去動作を行います。

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。

23.2.6.4　自動ブロック消去

(1)　動作内容

自動ブロック消去コマンドは、指定されたブロックに対する消去動作を行います。プロテクトされているページや指定されたブロックがプロテクトされている場合は自動ブロック消去を実行せず、コマンドシーケンスの入力後にリードモードに戻ります。

自動消去中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、データの消去は正常に行われていない可能性があるため、改めて自動消去を実行する必要があります。

(2)　実行方法

第 1～第 5 バスライトサイクルが自動ブロック消去のコマンドシーケンスです。第 6 バスライトサイクルで消去するブロックを指定します。コマンドシーケンス入力後、自動ブロック消去動作を行います。

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。

23.2.6.5　自動ページ消去

(1)　動作内容

自動ページ消去コマンドは、指定されたページに対する消去動作を行います。指定されたページがプロテクトされている場合は消去を実行せず、コマンドシーケンスの入力後にリードモードに戻ります。

自動消去中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、データの消去は正常に行われていない可能性があるため、改めて自動消去を実行する必要があります。

(2)　実行方法

第 1～第 5 バスライトサイクルが自動ページ消去のコマンドシーケンスです。第 6 バスライトサイクルで消去するページを指定します。コマンドシーケンス入力後、自動ページ消去動作を行います。

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。

23.2.6.6 自動プロテクトビットプログラム

(1) 動作内容

自動プロテクトビットプログラムは、プロテクトビットにビット単位で"1"を書き込みます。プロテクトビットを"0"にするためには自動プロテクトビット消去コマンドを使用します。

プロテクトの機能については「23.1.5 プロテクト/セキュリティ機能」を参照してください。

自動プロテクトビットプログラム中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、プロテクトの設定は正常に行われていない可能性があるため、改めて自動プロテクトビットプログラムを実行する必要があります。

(2) 実行方法

第 1～第 3 バスライトサイクルが自動プロテクトビットプログラムのコマンドシーケンスになります。第 4 バスライトサイクルで書き込むプロテクトビットを指定します。コマンドシーケンス入力後、自動プロテクトビットプログラム動作を行います。正しく書き込みができたかどうか、各 FCPSR レジスタの各ビットを確認してください。

23.2.6.7 自動プロテクトビット消去

(1) 動作内容

自動プロテクトビット消去コマンドは、実行する際のセキュリティの状態によって動作内容が異なります。

- セキュリティ状態でない場合

  すべてのプロテクトビットを"0"にクリアします。
- セキュリティ状態の場合

  フラッシュメモリの全アドレスのデータを消去した後、すべてのプロテクトビットを消去します。

プロテクトの機能については、「23.1.5 プロテクト/セキュリティ機能」を参照してください。

自動プロテクトビット消去中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、プロテクトの消去は正常に行われていない可能性があるため、改めて自動プロテクトビット消去を実行する必要があります。

(2) 実行方法

第 1～第 3 バスライトサイクルが自動プロテクトビット消去のコマンドシーケンスになります。第 4 バスライトサイクルで 0x5E000000 を指定します。コマンドシーケンス入力後、自動動作を行います。

セキュリティ状態でない場合、すべてのプロテクトビットが消去されます。正常に消去が行われたか、各 FCPSR レジスタの各ビットを確認してください。

セキュリティ状態の場合、フラッシュメモリの全アドレスのデータ消去およびすべてのプロテクトビットが消去されます。データの消去およびプロテクトビットの消去が正常に行われたか読み出して確認し、必要に応じて再度自動プロテクトビット消去、自動チップ消去あるいは自動ブロック消去を実行してください。

23.2.6.8　ID-Read

(1)　動作内容

ID-Read コマンドは、フラッシュメモリのタイプ等の情報を読み出すことができます。読み出せる内容は、メーカーコード、デバイスコード、マクロコードの 3 種類です。

(2)　実行方法

第 1 ~ 第 3 バスライトサイクルが ID-Read のコマンドシーケンスになります。第 4 バスサイクルで読み出すコードを指定します。第 4 バスライトサイクル以降、任意のフラッシュ領域からのリード動作でコードが得られます。

ID-Read コマンドは連続実行が可能です。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。

ID-Read コマンドは自動的にリードモードに戻りません。リードモードへの復帰は Read/リセットコマンドで行います。

23.2.6.9　Read/リセットコマンド (ソフトウエアリセット)

(1)　動作内容

フラッシュメモリをリードモードに戻すコマンドです。

ID-Read コマンドを実行した場合、Flash メモリは自動的にはリードモードに復帰せず、その状態で停止します。このような状態からリード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する場合にも使用します。

(2)　実行方法

Read/リセットコマンドでは第 1 バスライトサイクルがコマンドシーケンスになります。コマンドシーケンス実行後、フラッシュメモリはリードモードになります。

23.2.6.10　自動メモリスワップ

(1)　動作内容

自動メモリスワップは、FCSWPSR[10:0]の各ビットにビット単位で"1"を書き込むコマンドです。各ビットを"0"にすることはできず、自動プロテクトビット消去コマンドを使用してすべてのビットを"0"クリアします。

自動メモリスワップ動作中は、新たにコマンドシーケンスを受け付けません。この動作を中止するには「23.2.4 自動動作の中止」を参考に中止してください。この場合、この動作は正常に行われていない可能性があるため、改めて自動メモリスワップを実行する必要があります。

(2) 実行方法

第 1～第 4 バスライトサイクルが自動メモリスワップのコマンドシーケンスです。コマンドシーケンス入力後、FCSWPSR レジスタの指定ビットに"1"が書き込まれます。

23.2.7 コマンドシーケンス

23.2.7.1 コマンドシーケンス一覧

表 23-10 に各コマンドのバスライトサイクルのアドレスとデータを示します。

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(1 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 23-11 を参照してください。表 23-11 で「コマンド」と記載された、Addr [15:9]に下記値を使用します。

注 1) 各コマンドのアドレスは、Flash 領域(Mirror）に設定してください

注 2) アドレスビット[20:19]は対象のエリアにより以下の値を設定してください。
エリア 0 : "00"

表 23-10 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/リセット	0xXXXXXX	–	–	–	–	–	–
	0xF0	–	–	–	–	–	–
ID-Read	0xXX54XX	0xXXAAXX	0xXX54XX	IA	0xXX	–	–
	0xAA	0x55	0x90	0x00	ID	–	–
自動プログラム	0xXX54XX	0xXXAAXX	0xXX54XX	PA	PA	PA	PA
	0xAA	0x55	0xA0	PD0	PD1	PD2	PD3
自動チップ消去	0xXX54XX	0xXXAAXX	0xXX54XX	0xXX54XX	0xXXAAXX	0xXX54XX	–
	0xAA	0x55	0x80	0xAA	0x55	0x10	–
自動エリア消去	0xXX54XX	0xXXAAXX	0xXX54XX	0xXX54XX	0xXXAAXX	AA	–
	0xAA	0x55	0x80	0xAA	0x55	0x20	–
自動ブロック消去	0xXX54XX	0xXXAAXX	0xXX54XX	0xXX54XX	0xXXAAXX	BA	–
	0xAA	0x55	0x80	0xAA	0x55	0x30	–
自動ページ消去	0xXX54XX	0xXXAAXX	0xXX54XX	0xXX54XX	0xXXAAXX	PGA	–
	0xAA	0x55	0x80	0xAA	0x55	0x40	–
自動プロテクトビット プログラム	0xXX54XX	0xXXAAXX	0xXX54XX	PBA	–	–	–
	0xAA	0x55	0x9A	0x9A	–	–	–
自動プロテクトビット 消去	0xXX54XX	0xXXAAXX	0xXX54XX	0x0000XX	–	–	–
	0xAA	0x55	0x6A	0x6A	–	–	–
自動メモリスワップ	0xXX54XX	0xXXAAXX	0xXX54XX	MSA	–	–	–
	0xAA	0x55	0x9A	0x9A	–	–	–

補足説明

- IA: ID アドレス
- ID: ID データ
- PA: プログラムアドレス
- 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 16 バイト分をアドレス順にデータ入力
- AA: エリアアドレス(表 23-2 参照)
- BA: ブロックアドレス(表 23-2 参照)
- PGA: ページアドレス
- PBA: プロテクトビットアドレス(表 23-12 参照)
- MSA: メモリスワップアドレス(表 23-14 参照)

23.2.7.2 バスライトサイクル時のアドレスビット構成

表 23-11 は「表 23-10 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。

表 23-11 バスライトサイクル時のアドレスビット構成

[通常のコマンド]

アドレス	Addr [31:24]	Addr [23:21]	Addr [20:19]	Addr [18:16]	Addr [15:9]	Addr [8:0]
通常コマンド	通常コマンドのバスライトサイクルアドレス設定					
	0x5E	"000" 固定	注	"0"推奨	コマンド	"0"推奨

[Read/リセット、ID-READ]

アドレス	Addr [31:24]	Addr [23:21]	Addr [20:19]	Addr [18:16]	Addr [15:14]	Addr [13:0]
Read /リセット	Read/リセットの第 1 バスライトサイクルアドレス設定					
	0x5E	"000" 固定	"00" 固定	"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)					
	0x5E	"000" 固定	"00" 固定	"0"推奨	ID アドレス (表 23-13)	"0"推奨

[自動エリア消去]

アドレス	Addr [31:24]	Addr [23:21]	Addr [20:19]	Addr [18:0]
エリア消去	AA: エリアアドレス(エリア消去の第 6 バスライトサイクルアドレス設定)			
	0x5E	"000" 固定	注	"0"推奨

[自動ブロック消去]

アドレス	Addr [31:24]	Addr [23:21]	Addr [20:19]	Addr [18:13]	Addr [12:0]
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)				
	0x5E	"000" 固定	注	ブロックアドレス(表 23-2)	"0"推奨

[自動ページ消去]

アドレス	Addr [31:24]	Addr [23:21]	Addr [20:19]	Addr [18:12]	Addr [11:0]
ページ消去	PGA: ページアドレス(ページ消去の第 6 バスライトサイクルアドレス設定)				
	0x5E	"000" 固定	注	ページアドレス	"0"推奨

[自動プログラム]

アドレス	Addr [31:24]	Addr [23:21]	Addr [20:19]	Addr [18:3]	Addr [2:0]
プログラム	PA: プログラムアドレス(プログラムの第 4 バスライトサイクルアドレス設定)				
	0x5E	"000" 固定	注	プログラムアドレス	"0"推奨

[プロテクトビットプログラム、メモリスワップ]

アドレス	Addr [31:24]	Addr [23:21]	Addr [20:19]	Addr [18:12]	Addr [11:8]	Addr [7:4]	Addr [3:0]
プロテクトビット消去	プロテクトビット消去の第 4 バスライトサイクルアドレス設定						
	0x5E	"000" 固定	"0"固定			"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 4 バスライトサイクルアドレス設定)						
	0x5E	"000" 固定	"00" 固定	"0" 推奨		プロテクトビット選択 (表 23-12)	"0"推奨
メモリスワップ	MSA: メモリスワップアドレス(メモリスワップの第 4 バスライトサイクルアドレス設定)						
	0x5E	"000" 固定	"00" 固定	"0" 推奨		メモリスワップビット選択 (表 23-14)	"0"推奨

注)　アドレスビット[20:19]は対象のエリアにより以下の値を設定してください。
エリア 0 : "00"

23.2.7.3　エリアアドレス(AA)、ブロックアドレス(BA)

表 23-2 にエリアアドレスとブロックアドレスを示します。自動エリア消去コマンドと自動ブロック消去コマンドの第 6 バスライトサイクルで、消去する"エリア"またはブロックに含まれる任意のアドレスを指定します。シングルチップモードでは、ミラー領域のアドレスを指定してください。

23.2.7.4　プロテクトビットの指定(PBA)

プロテクトビットは、1 ビット単位の操作になります。

自動プロテクトビットプログラムのプロテクトビット選択表を表 23-12 に示します。

表 23-12 プロテクトビットプログラムアドレス表

Block	Page	レジスタ	プロテクトビット	PBA[11:4]							アドレス例 [31:0]
				Addr [11:10]	Addr [9]	Addr [8]	Addr [7]	Addr [6]	Addr [5]	Addr [4]	
0	0	FCPSR0	<PG0>	0	0	0	0	0	0	0	0x5E00_0000
	1		<PG1>	0	0	0	0	0	0	1	0x5E00_0010
	2		<PG2>	0	0	0	0	0	1	0	0x5E00_0020
	3		<PG3>	0	0	0	0	0	1	1	0x5E00_0030
	4		<PG4>	0	0	0	0	1	0	0	0x5E00_0040
	5		<PG5>	0	0	0	0	1	0	1	0x5E00_0050
	6		<PG6>	0	0	0	0	1	1	0	0x5E00_0060
	7		<PG7>	0	0	0	0	1	1	1	0x5E00_0070
1	8 ~ 15		<BLK1>	0	0	0	1	0	0	0	0x5E00_0080
2	16 ~ 23		<BLK2>	0	0	0	1	0	0	1	0x5E00_0090
3	24 ~ 31		<BLK3>	0	0	0	1	0	1	0	0x5E00_00A0
4	32 ~ 39		<BLK4>	0	0	0	1	0	1	1	0x5E00_00B0
5	40 ~ 47		<BLK5>	0	0	0	1	1	0	0	0x5E00_00C0
6	48 ~ 55		<BLK6>	0	0	0	1	1	0	1	0x5E00_00D0
7	56 ~ 63		<BLK7>	0	0	0	1	1	1	0	0x5E00_00E0
8	64 ~ 71		<BLK8>	0	0	0	1	1	1	1	0x5E00_00F0
9	72 ~ 79		<BLK9>	0	0	1	0	0	0	0	0x5E00_0100
10	80 ~ 87		<BLK10>	0	0	1	0	0	0	1	0x5E00_0110
11	88 ~ 95		<BLK11>	0	0	1	0	0	1	0	0x5E00_0120
12	96 ~ 103		<BLK12>	0	0	1	0	0	1	1	0x5E00_0130
13	104 ~ 111		<BLK13>	0	0	1	0	1	0	0	0x5E00_0140
14	112 ~ 119		<BLK14>	0	0	1	0	1	0	1	0x5E00_0150
15	120 ~ 127		<BLK15>	0	0	1	0	1	1	0	0x5E00_0160

23.2.7.5　ID-Read のコード(IA, ID)

ID-Read コマンドでのコード指定方法と読み出される内容を表 23-13 に示します。

表 23-13 ID-Read コマンドのコード指定とコードの内容

Code	ID[7:0]	IA[15:14]	アドレス例[31:0]
メーカーコード	0x0098	00	0x5E00_0000
デバイスコード	0x005A	01	0x5E00_4000
−	Reserved	10	−
マクロコード	0x012F	11	0x5E00_C000

23.2.7.6　メモリスワップビットの指定(MSA)

自動メモリスワップコマンドの第 4 バスライトサイクルで指定する FCSWPSR[10:0]への設定値を表 23-14 に示します。

表 23-14 メモリスワップコマンドによる FCSWPSR[10:0]への設定値とアドレス例

FCSWPSR[10:0]	MSA[11:4]							アドレス例 [31:0]
	アドレス [11]	アドレス [10:9]	アドレス [8]	アドレス [7]	アドレス [6]	アドレス [5]	アドレス [4]	
FCSWPSR[0]	1	"0"固定	0	1	0	0	0	0x5E00_0880
FCSWPSR[1]	1	"0"固定	0	1	0	0	1	0x5E00_0890
FCSWPSR[2]	1	"0"固定	0	1	0	1	0	0x5E00_08A0
FCSWPSR[3]	1	"0"固定	0	1	0	1	1	0x5E00_08B0
FCSWPSR[4]	1	"0"固定	0	1	1	0	0	0x5E00_08C0
FCSWPSR[5]	1	"0"固定	0	1	1	0	1	0x5E00_08D0
FCSWPSR[6]	1	"0"固定	0	1	1	1	0	0x5E00_08E0
FCSWPSR[7]	1	"0"固定	0	1	1	1	1	0x5E00_08F0
FCSWPSR[8]	1	"0"固定	1	0	0	0	0	0x5E00_0900
FCSWPSR[9]	1	"0"固定	1	0	0	0	1	0x5E00_0910
FCSWPSR[10]	1	"0"固定	1	0	0	1	0	0x5E00_0920

23.2.7.7 コマンドシーケンス例

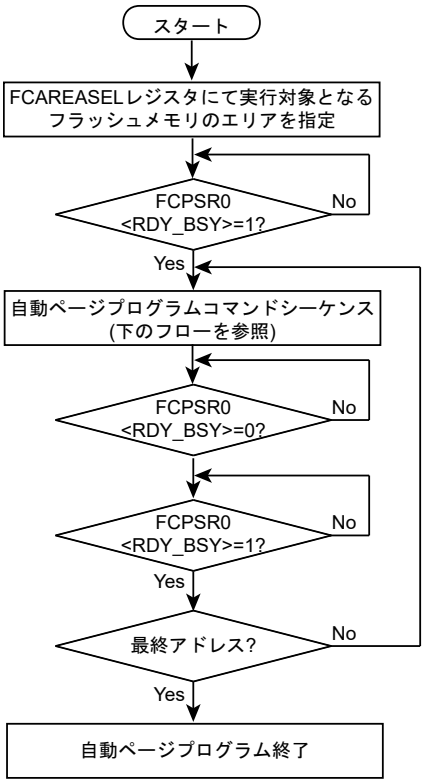
コマンドシーケンス例

コマンド		バスサイクル						
		1	2	3	4	5	6	7
Read/リセット	アドレス	0x5E00_0000	–	–	–	–	–	–
	データ	0x0000_00F0	–	–	–	–	–	–
ID-Read	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	IA	0x5E00_0000	–	–
	データ	0x0000_00AA	0x0000_0055	0x0000_0090	0x0000_0000	ID	–	–
自動チップ消去	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	0x5E00_5400	0x5E00_AA00	0x5E00_5400	–
	データ	0x0000_00AA	0x0000_0055	0x0000_0080	0x0000_00AA	0x0000_0055	0x0000_0010	–
自動プロテクトビットプログラム	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	PBA	–	–	–
	データ	0x0000_00AA	0x0000_0055	0x0000_009A	0x0000_009A	–	–	–
自動プロテクトビット消去	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	0x5E00_0000	–	–	–
	データ	0x0000_00AA	0x0000_0055	0x0000_006A	0x0000_006A	–	–	–
自動メモリスワップ	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	MSA	–	–	–
	データ	0x0000_00AA	0x0000_0055	0x0000_006A	0x0000_009A	–	–	–

コマンド		バスサイクル						
		1	2	3	4	5	6	7
自動プログラム	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	PA	以降、連続して 16 バイト分のデータを書き込み		
	データ	0x0000_00AA	0x0000_0055	0x0000_00A0	PD			
自動エリア消去	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	0x5E00_5400	0x5E00_AA00	0x5E00_0000	–
	データ	0x0000_00AA	0x0000_0055	0x0000_0080	0x0000_00AA	0x0000_0055	0x0000_0020	–
自動ブロック消去	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	0x5E00_5400	0x5E00_AA00	BA	–
	データ	0x0000_00AA	0x0000_0055	0x0000_0080	0x0000_00AA	0x0000_0055	0x0000_0030	–
自動ページ消去	アドレス	0x5E00_5400	0x5E00_AA00	0x5E00_5400	0x5E00_5400	0x5E00_AA00	PGA	–
	データ	0x0000_00AA	0x0000_0055	0x0000_0080	0x0000_00AA	0x0000_0055	0x0000_0040	–

23.2.8　フローチャート

23.2.8.1　自動プログラム



自動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)

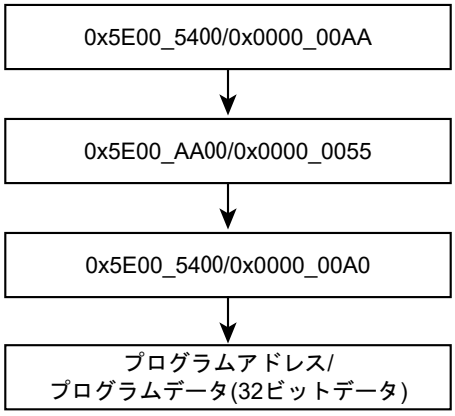


図 23-5 自動プログラムフローチャート

23.2.8.2 自動消去

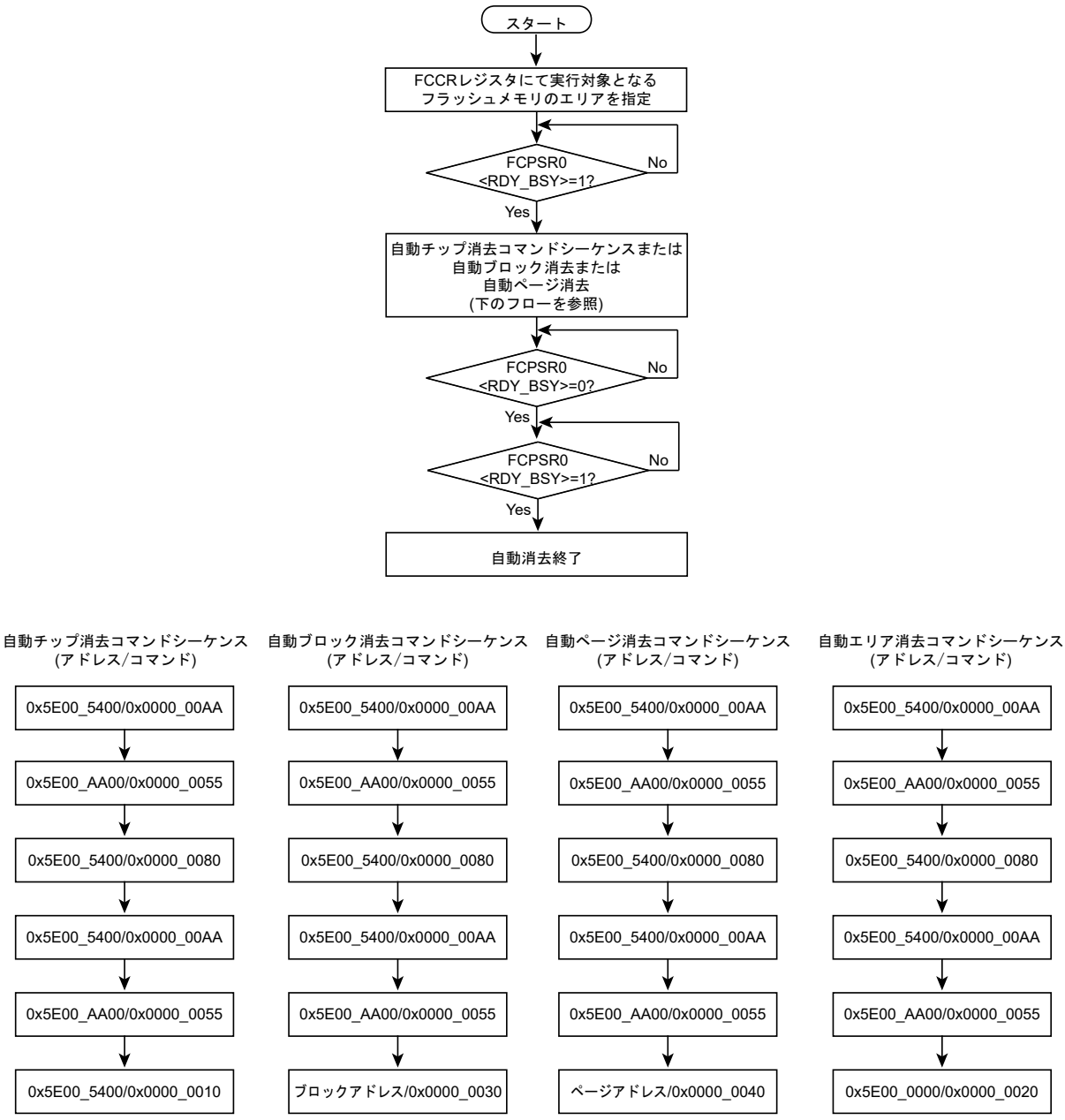


図 23-6 自動消去フローチャート

23.3　シングルブートモードによる書き替え方法

内蔵ブート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、BOOT 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、フラッシュメモリは BOOT ROM 領域とは別のアドレス空間にマッピングされます。

シングルブートモードでは、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。本デバイスのシリアルチャネル(SIO/UART) と外部ホストを接続し、外部ホスト側から内蔵 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。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。

シングルブートモード中は、すべての例外発生を禁止してください。

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。

23.3.1　モード設定

オンボードプログラミングを実行するためには、本デバイスをシングルブートモードで立ち上げます。シングルブートモードで立ち上がるための設定を以下に示します。

$\overline{\text{BOOT}}$ = 0
$\overline{\text{RESET}}$ = 0 → 1

$\overline{\text{RESET}}$ 入力端子を"0"の状態にして、$\overline{\text{BOOT}}$ 端子をあらかじめ上記条件に設定します。その後 RESET 解除を行うとシングルブートモードで起動します。

23.3.2　インタフェース仕様

シングルブートモードでのシリアル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) 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。

通信チャネル:	チャネル 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

内蔵ブートプログラムは、クロック/モード制御ブロックの設定は初期状態のままで動作します。クロック設定の初期状態は、「クロック/モード制御」の章を参照してください。

ボーレートは、「23.3.5.1 シリアル動作モード判定」で説明しているように 16 ビットタイマ(TMRB) を用いて判定します。判定時のボーレートは所望のボーレートの 1/16 で通信するため、このボーレートがタイマで計測可能な範囲である必要があります。タイマのカウントクロックは ΦT1(fc/2)で動作します。

内蔵ブートプログラムで使用する端子を表 23-15 にまとめます。これ以外の端子は内蔵ブートプログラムでは操作しません。

表 23-15 使用端子

モード設定端子	$\overline{\text{BOOT}}$(PF0)
リセット端子	$\overline{\text{RESET}}$
通信端子	SC0TXD (PE0)
	SC0RXD (PE1)

23.3.3 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, 内蔵フラッシュメモリに対して表 23-16 のような制約がありますのでご注意ください。

表 23-16 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 プログラムのワークエリアになります。 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。
内蔵フラッシュメモリ	以下の番地はソフトなどの ID 情報やパスワード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。 0x5E07_FFF0 ~ 0x5E07_FFFF

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。

23.3.4 動作コマンド

内蔵ブートプログラムには、以下の動作コマンドが準備されています。

表 23-17 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
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

23.3.4.1 RAM 転送

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プログラムの実行を開始します。ユーザプログラム領域として、内蔵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、ユーザ独自のオンボートプログラミング制御を行うことができます。ユーザ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、23.2.7 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。

23.3.4.2 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。

23.3.5　コマンドによらず共通の動作

内蔵ブートプログラム実行において、共通に行われる動作について説明します。

23.3.5.1　シリアル動作モード判定

コントローラは、所望のボーレートで 1 バイト目を 0x86 にして送信してください。図 23-7 に波形を示します。

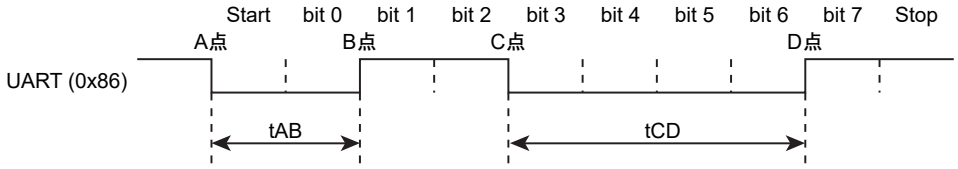


図 23-7 シリアル動作モード判定データ

内蔵ブートプログラムは図 23-8 に示すフローチャートで、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)を、16 ビットタイマ(TMRB)を用いて図 23-7 の tAB, tAC と、tAD の時間から求めています。図 23-8 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、tAB, tAC と、tAD 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。

図 23-9 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。tAB ≤ tCD 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 tAD の時間から判定します。tAB > tCD の場合、UART と判定しません。なお、先に述べたように、tAB, tAC, tAD 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、UART と判定されないことがあります。このようなことを考慮して、コントローラは 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。

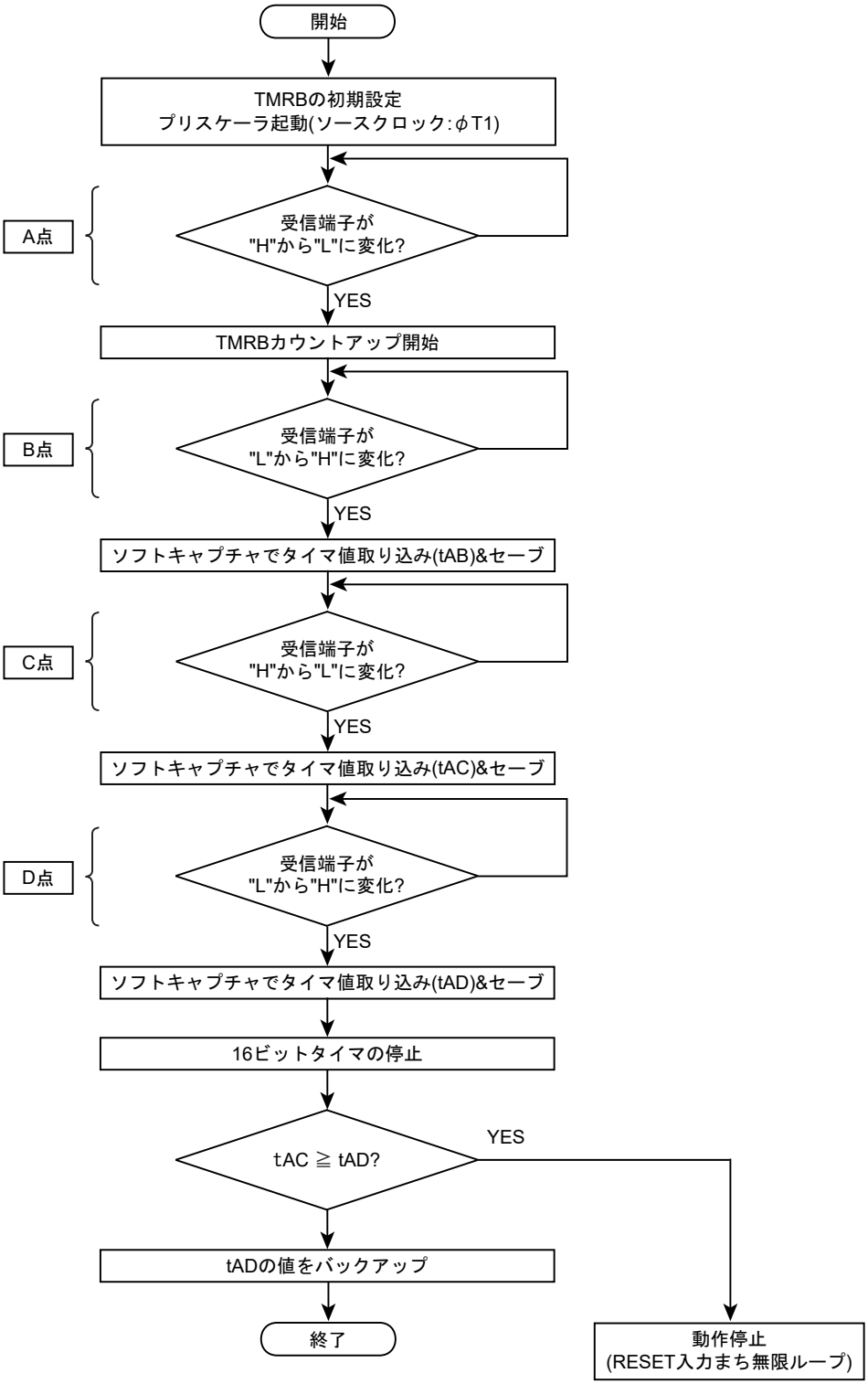


図 23-8 シリアル動作モード受信フローチャート

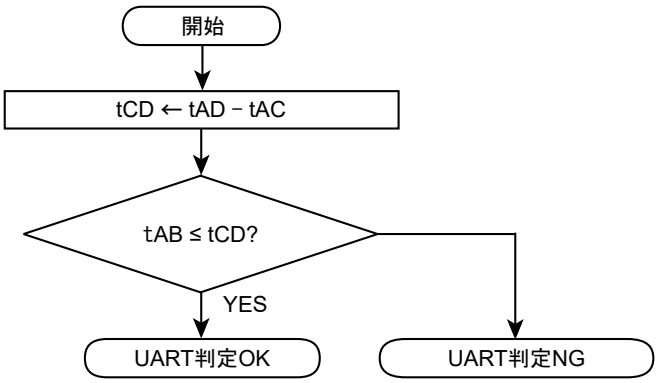


図 23-9 シリアル動作モード判定フローチャート

23.3.5.2　ACK 応答データ

内蔵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 23-18 から表 23-21 に各受信データに対する ACK 応答データを示します。

表 23-19 から表 23-21 に示す 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。

表 23-18 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)

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 23-19 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。
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 23-20 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。

表 23-21 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
0x47	フラッシュ操作コマンドを中止した。

23.3.5.3 パスワード判定

内蔵ブートプログラムでは、以下の領域をパスワード要否判定およびパスワードのデータ領域として使用します。

領域	アドレス
パスワード要否判定	0x5E07_FFF0 (1byte)
パスワード	0x5E07_FFF4 ~ 0x5E07_FFFF (12byte)

RAM 転送コマンドでは、パスワード要否判定にかかわらずパスワード判定を行います。

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では、パスワード要否判定データが「要」の場合のみパスワード判定を行います。

パスワード要否選択	データ
パスワード要	0xFF 以外
パスワード否	0xFF

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。

(1) RAM 転送コマンドでのパスワード判定

「RAM 転送コマンドの通信ルール」における No.5 のパスワード判定について説明します。

図 23-10 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、ACK 対応は 0x11 を送信します。

次に、受信データ(パスワードデータ)の照合を行います。12 バイト分すべてがフラッシュメモリ内のパスワードと一致しないと、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、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。

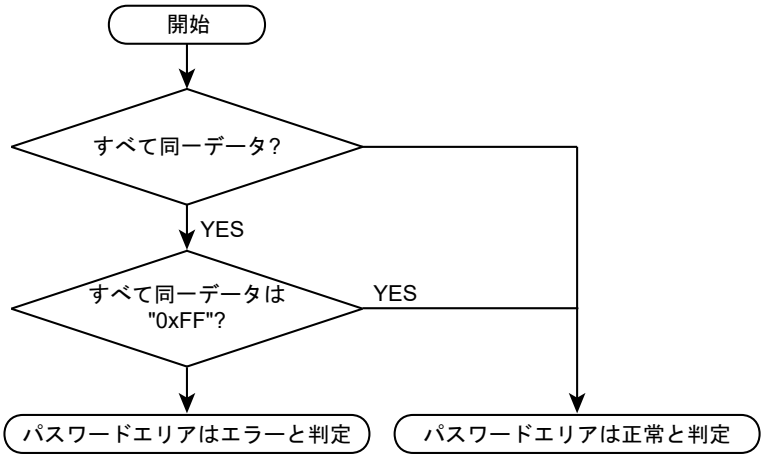


図 23-10 パスワードエリアチェックフローチャート

(2)　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でのパスワード判定

「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通信ルール」における No.5 のパスワード判定について説明します。

図 23-11 に示すように、消去パスワード要否の選択エリアがパスワード有効の場合は、パスワードエリアのデータが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、CHECK SUM 値に対する ACK 対応は 0x41 を送信します。

次に、受信データ (パスワードデータ) の照合を行います。12 バイト分すべてがフラッシュメモリ内のパスワードと一致しないと、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、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。

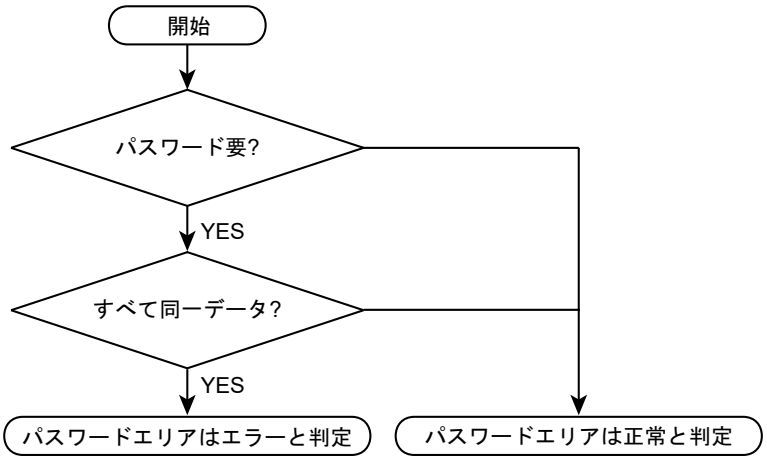


図 23-11 パスワードエリアチェックフローチャート

23.3.5.4 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。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)CHECK SUM 計算例

2 バイトのデータ 0xE5、0xF6 の CHECK SUM 値を求める場合、まず符号なし 8 ビット加算を行います。

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

23.3.6 シリアル動作モード判定の通信ルール

シリアル動作モード判定の通信ルールを示します。表中の転送方向の表記が示す意味は以下のとおりです。

転送方向「C→T」: コントローラから TMPM470 へ

転送方向「C←T」: TMPM470 からコントローラへ

シリアル動作モード判定の通信ルール

No	転送方向	転送データ	内容
1	C→T	シリアル動作モード 、 ボーレート設定	コントローラ側はシリアル動作モードを判定するデータを送信します。ターゲット側のモード判定の詳細は「23.3.5.1 シリアル動作モード判定」を参照してください。
		0x86	コントローラ側は 0x86 を送信してください。 ターゲット側が UART モード OK 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。
2	C←T	シリアル動作モードに対する ACK 応答	コントローラ側の受信データは、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。 ターゲット側は設定が可能と判定した場合、UART の設定を行います。受信を許可するタイミングは、送信バッファにデータを書き込む前に行っています。
		正常の場合: 0x86	ターゲット側が設定が可能と判定した場合 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。 コントローラ側は、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)を設けてください。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。
3	−	−	コントローラ側は動作コマンドデータを送信してください。 各動作コマンドの転送フォーマットは「23.3.7 RAM 転送コマンドの通信ルール」または「23.3.8 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通信ルール」を参照してください。

23.3.7　RAM 転送コマンドの通信ルール

RAM 転送コマンドの通信ルールを示します。表中の転送方向の表記が示す意味は以下のとおりです。

転送方向「C→T」: コントローラから TMPM470 へ

転送方向「C←T」: TMPM470 からコントローラへ

RAM 転送コマンドの通信ルール

No	転送方向	転送データ	内容
1	C→T	動作コマンドデータ(0x10)	コントローラ側は RAM 転送コマンドデータ(0x10)を送信してください。
2	C←T	動作コマンドに対する ACK 応答 正常の場合: 0x10 異常の場合: 0x11 通信異常の場合: 0x18	ターゲット側は、受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ 0x18 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、表 23-17 に記載の動作コマンドデータとの照合を行います。 照合に失敗した場合、異常の ACK 応答データ 0x11 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 照合に成功した場合、正常の ACK 応答データ 0x10 を返信して、次の送信データを待ちます。
3	C→T	パスワードデータ(12 バイト)	コントローラ側はフラッシュメモリのパスワードのデータ領域と同じデータを送信してください。 パスワードのデータ領域は「23.3.5.3 パスワード判定」を参照してください。
4	C→T	送信データ(No.3)の CHECK SUM 値	コントローラ側は送信データ(No.3)の CHECK SUM 値を送信してください。 CHECK SUM の計算方法は「23.3.5.4 CHECK SUM の計算方法」を参照してください。
5	C←T	CHECK SUM 値に対する ACK 応答 正常の場合: 0x10 異常の場合: 0x11 通信異常の場合: 0x18	ターゲット側は受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ 0x18 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、CHECK SUM の確認とパスワードの照合を行います。パスワードの照合の詳細は「23.3.5.3 パスワード判定」してください。 照合に失敗した場合、異常の ACK 応答データ 0x11 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 照合に成功した場合、正常の ACK 応答データ 0x10 を返信して、次の送信データを待ちます。
6	C→T	RAM 格納開始アドレス 31 ~ 24	コントローラ側は次の送信データとして RAM 格納データの格納先の RAM の開始アドレスを 4 回に分けて送信してください。送信順番は、1 番目がアドレスの 31 ビット ~ 24 ビットに対応し、4 番目が 7 ビット ~ 0 ビットに対応します。RAM のアドレス 0x2000_0400 から RAM の最終番地に収まるように指定してください。 ターゲット側は受信データをチェックします。 受信エラーがある場合は通信異常の ACK 応答データ 0x18 を返信して初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合は何も ACK 応答データは返信せずに次の送信データを待ちます。
7	C→T	RAM 格納開始アドレス 23 ~ 16	
8	C→T	RAM 格納開始アドレス 15 ~ 8	
9	C→T	RAM 格納開始アドレス 7 ~ 0	
10	C→T	RAM 格納バイト数 15 ~ 8	コントローラ側はブロック転送するバイト数を送信してください。送信順番は、1 番目が転送バイト数の 15 ビット ~ 8 ビット目に対応し、2 番目が 7 ビット ~ 0 ビット目に対応します。RAM のアドレス 0x2000_0400 から RAM の最終番地に収まるように指定してください。 ターゲット側は受信データをチェックします。 受信エラーがある場合は通信異常の ACK 応答データ 0x18 を返信して初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合は何も ACK 応答データは返信せずに次の送信データを待ちます。
11	C→T	RAM 格納バイト数 7 ~ 0	
12	C→T	送信データ(No.6 ~ 11)の CHECK SUM 値	コントローラ側は送信データ(No.6 ~ 11)の CHECK SUM 値を送信してください。

No	転送方向	転送データ	内容
13	C←T	CHECK SUM 値に対する ACK 応答 正常の場合: 0x10 異常の場合: 0x11 通信異常の場合: 0x18	ターゲット側は、受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ 0x18 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、CHECK SUM の確認を行います。 確認に失敗した場合、異常の ACK 応答データ 0x11 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 確認に成功した場合、正常の ACK 応答データ 0x10 を返信して、次の送信データを待ちます。
14	C→T	RAM 格納データ	コントローラ側は RAM に格納するデータを送信してください。 ターゲット側は RAM 格納バイト数分のデータを受信します。
15	C→T	送信データ(No.14)の CHECK SUM 値	コントローラ側は送信データ(No.14)の CHECK SUM 値を送信してください。
16	C←T	CHECK SUM 値に対する ACK 応答 正常の場合:0x10 異常の場合: 0x11 通信異常の場合: 0x18	ターゲット側は、受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ 0x18 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、CHECK SUM の確認を行います。 確認に失敗した場合、異常の ACK 応答データ 0x11 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 確認に成功した場合、正常の ACK 応答データ 0x10 を返信して、RAM 格納開始アドレス(No.6 ~ 9)を分岐先アドレスとしてジャンプします。

23.3.8　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通信ルール

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の通信ルールを示します。表中の転送方向の表記が示す意味は以下のとおりです。

転送方向「C→T」: コントローラ→TMPM470

転送方向「C←T」: コントローラ←TMPM470

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通信ルール

No	転送方向	転送データ	内容
1	C→T	動作コマンドデータ(0x40)	コントローラ側は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(0x40)を送信してください。
2	C←T	動作コマンドに対する ACK 応答 正常の場合: 0x40 異常の場合: 0x41 通信異常の場合: 0x48	ターゲット側は、受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ 0x48 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、表 23-17 に記載の動作コマンドデータとの照合を行います。 照合に失敗した場合、異常の ACK 応答データ 0x41 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 照合に成功した場合、正常の ACK 応答データ 0x40 を返信して、次の送信データを待ちます。
3	C→T	パスワードデータ(12 バイト)	コントローラ側はフラッシュメモリのパスワードのデータ領域と同じデータを送信してください。 但し、フラッシュメモリのパスワード要否選択が「否」(データ:0xFF)の場合、ターゲット側はパスワード認証を行いませんので、パスワードデータはダミーデータで構いません。 パスワードのデータ領域は「23.3.5.3 パスワード判定」を参照してください。
4	C→T	送信データ(No.3)の CHECK SUM 値	コントローラ側は送信データ(No.3)の CHECK SUM 値を送信してください。 CHECK SUM の計算方法は「23.3.5.4 CHECK SUM の計算方法」を参照してください。
5	C←T	CHECK SUM 値に対する ACK 応答 正常の場合: 0x40 異常の場合: 0x41 通信異常の場合: 0x48	ターゲット側は受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ 0x48 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 受信エラーがない場合、CHECK SUM の確認を行います。 確認に失敗した場合、異常の ACK 応答データ 0x41 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 確認に成功した場合、パスワードの照合を行います。 パスワード要否選択が「否」の場合、正常 ACK 応答データ 0x40 を送信します。 パスワード要否選択が「要」の場合、パスワードの照合を行います。 照合に失敗した場合、異常の ACK 応答データ 0x41 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 照合に成功した場合、正常の ACK 応答データ 0x40 を返信して、次の送信データを待ちます。
6	C→T	消去イネーブルコマンドデータ(0x54)	コントローラ側は消去イネーブルコマンドデータ(0x54)を送信してください。
7	C←T	消去イネーブルコマンドに対する ACK 応答 正常の場合: 0x54 異常の場合: 0x51 通信異常の場合: 0x58	ターゲット側は受信データをチェックして ACK 応答データを返信します。 受信エラーがある場合、通信異常の ACK 応答データ 0x58 を返信して、初めの動作コマンドデータ待ち状態状態に戻ります。 受信エラーがない場合、消去イネーブルコマンド(0x54)であることを確認します。 確認に失敗した場合、異常の ACK 応答データ 0x51 を返信して、初めの動作コマンドデータ待ち状態に戻ります。 確認に成功した場合、正常の ACK 応答データ 0x54 を返信して、チップ消去処理を行います。
8	C←T	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	ターゲット側はチップ消去処理の結果を返信します。 問題なく消去できた場合、正常の応答データ(0x4F)を返信します。 ブランクチェックエラーが起きた場合は、異常の応答データ(0x4C)を返します。 自動チップ消去コマンドを中止した場合は、中止の応答データ(0x47)を返します。 その後、初めの動作コマンドデータ待ち状態に戻ります。

23.3.9 内蔵ブートプログラム全体フローチャート

内蔵ブートプログラム全体フローチャートを示します。

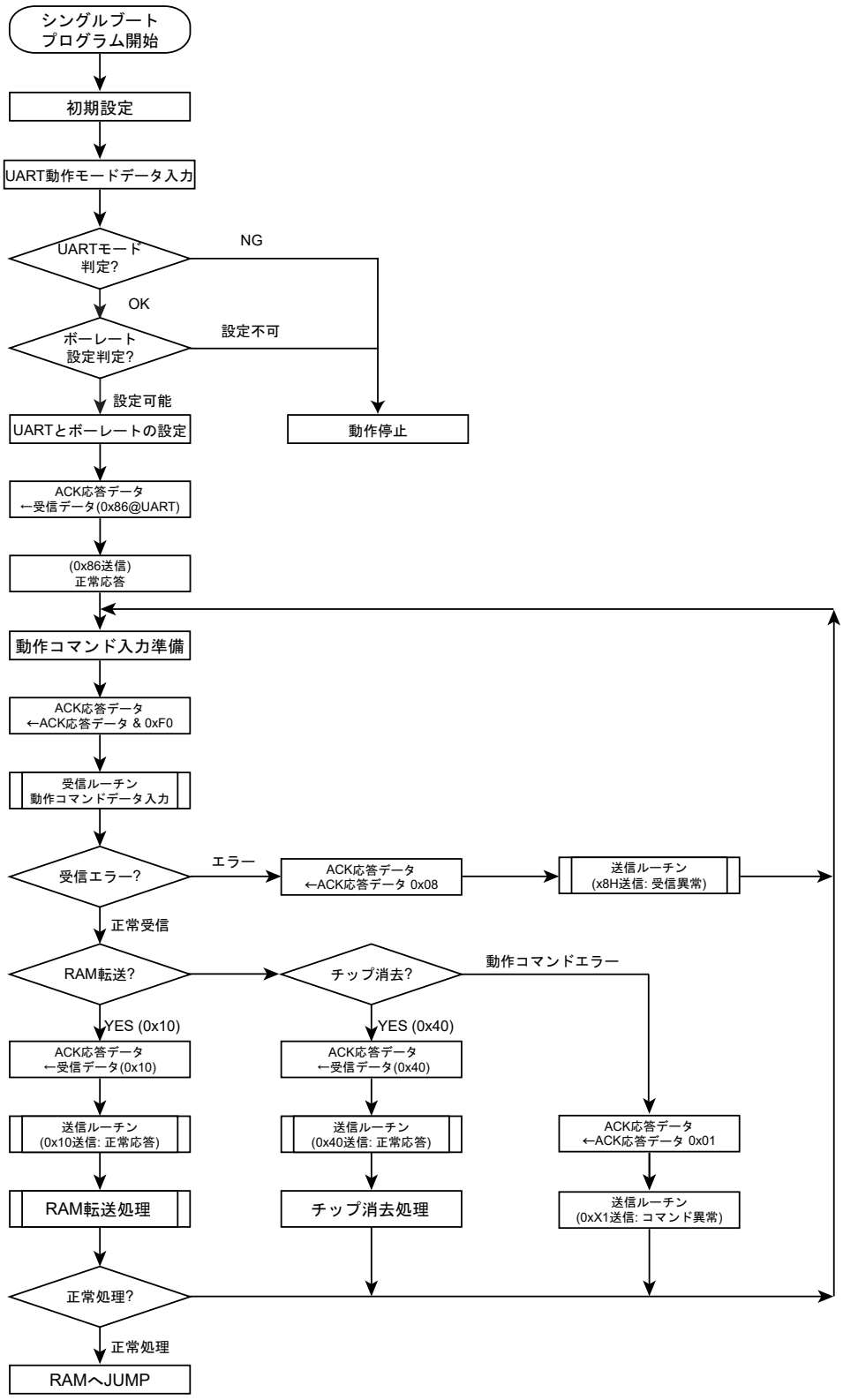


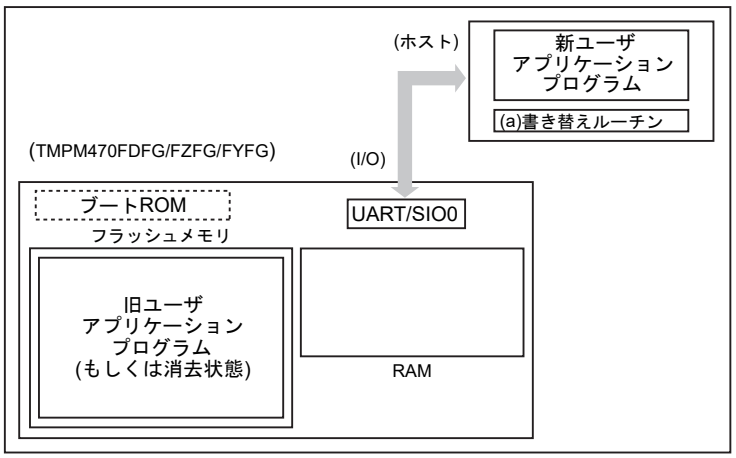
図 23-12 内蔵ブートプログラム全体フローチャート

23.3.10　内蔵 BOOT ROM の書き替えアルゴリズムを利用した書き替え手順

内蔵ブートプログラムを利用した書き替え手順を示します。

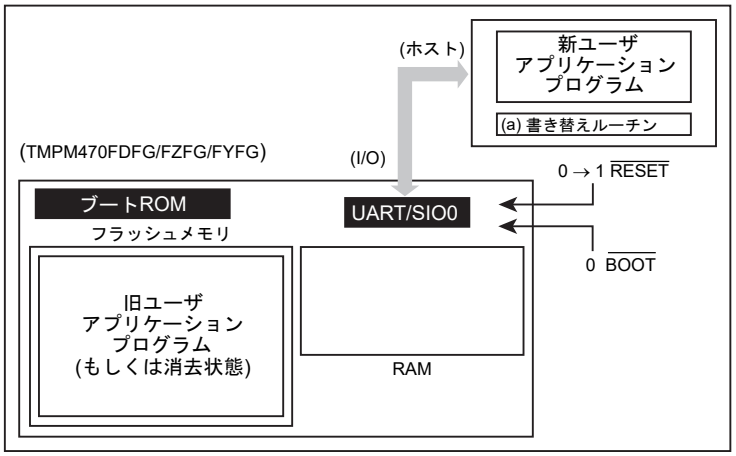
23.3.10.1　Step-1

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。



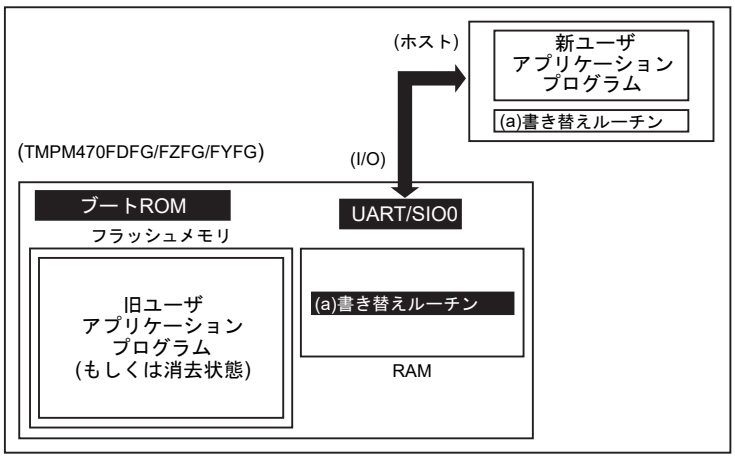
23.3.10.2　Step-2

シングルブートモードの端子条件設定でリセットを解除し、BOOT ROM で起動します。シングルブートモードの手順に従い、SIO0　を経由して転送元(ホスト)より(a)書き替えルーチンの転送を行いますが、最初にユーザ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。)



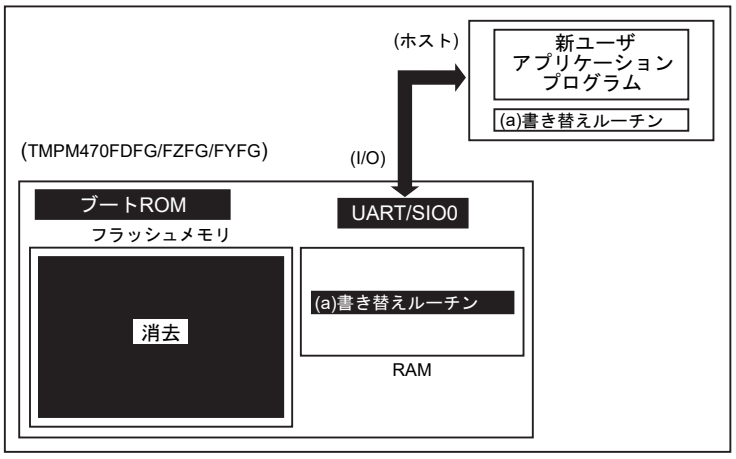
23.3.10.3 Step-3

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。BOOT 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。



23.3.10.4 Step-4

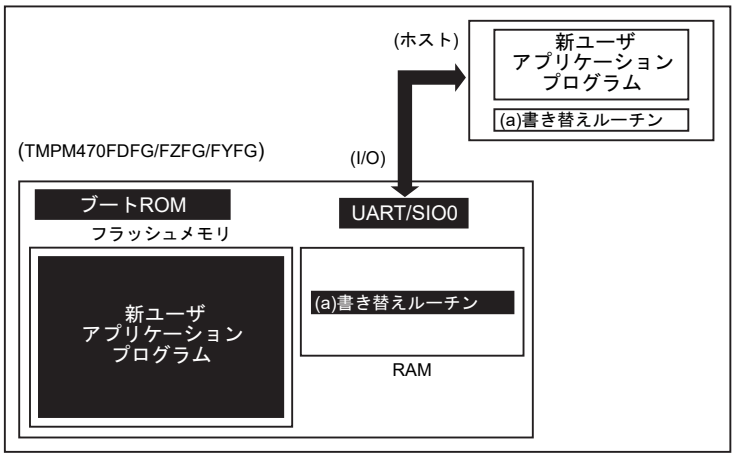
RAM 上の(a)書き替えルーチンへジャンプし、旧ユーザアプリケーションプログラム領域の消去を行います。(任意の消去単位)



23.3.10.5　Step-5

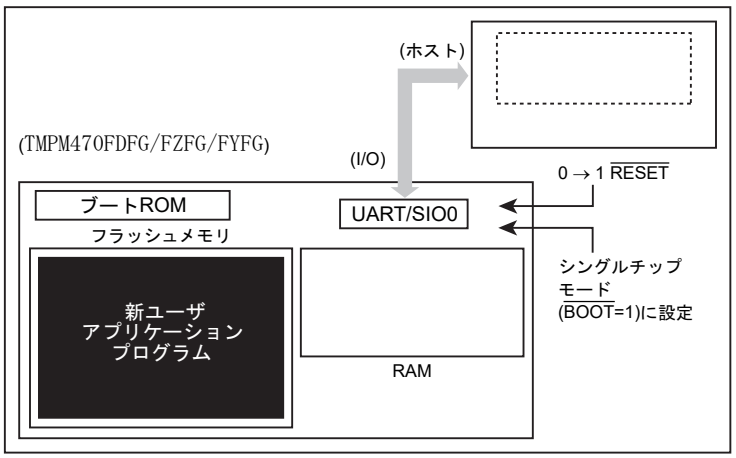
さらに、RAM 上の(a)書き替えルーチンを実行して、転送元(ホスト)より新ユーザアプリケーションプログラムのデータをロードし、フラッシュメモリの消去した領域に書き込みを行います。書き込みが完了したら、ユーザプログラム領域のライト/消去プロテクトをオンにします。

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。



23.3.10.6　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、新しいユーザアプリケーションプログラムを実行します。



23.4 ユーザブートモードによる書き替え方法

ユーザのセット上で内蔵 RAM に置くフラッシュメモリの書き替えプログラムにてフラッシュメモリの書き替えを実行する方法です。ユーザ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、ユーザ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、ユーザブートモード移行後に使用するユーザ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。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。ユーザブートモード中は、すべての例外発生を禁止してください。

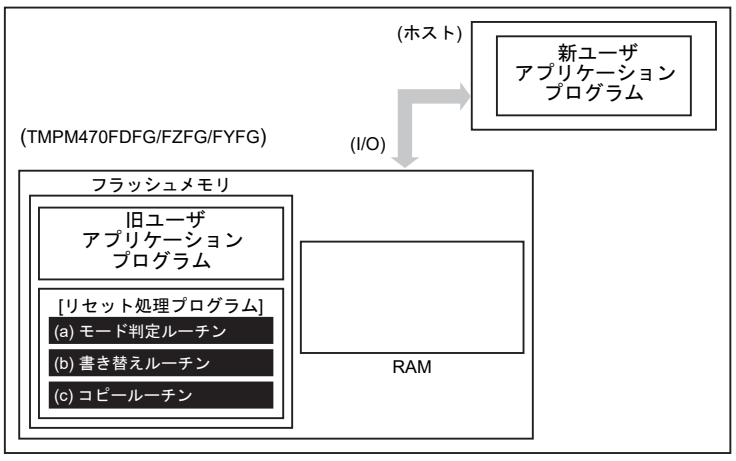
書き替えルーチンをフラッシュメモリに置く場合と、外部から転送する場合の 2 ケースを例に、以下 (1-A), (1-B)にその手順を説明します。フラッシュメモリへの書き込み/消去 方法の詳細は、「23.2 フラッシュメモリ詳細」を参照してください。

23.4.1 (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

23.4.1.1 Step-1

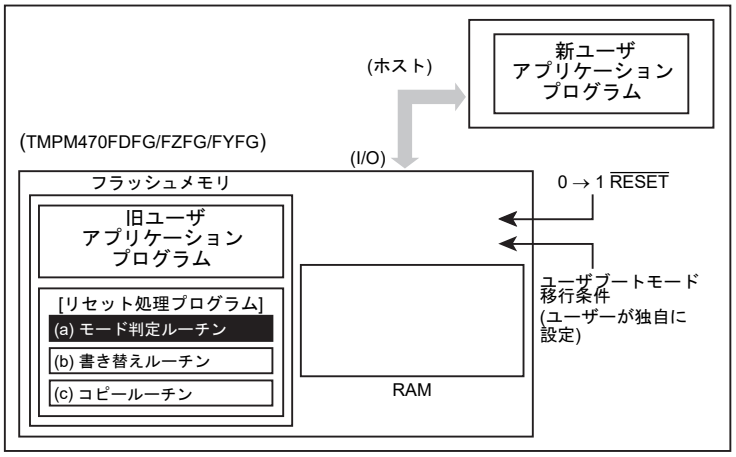
ユーザは、あらかじめ どのような条件 (例えば端子状態) に設定されたらユーザブートモードに移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合った回路の設計、プログラムの作成を行います。ユーザは本デバイスをボードに組み込む前に、あらかじめフラッシュメモリ上の任意のブロックにライタなどを使用して以下に示す 3 つのプログラムを書き込んでおきます。

(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 にコピーするためのプログラム



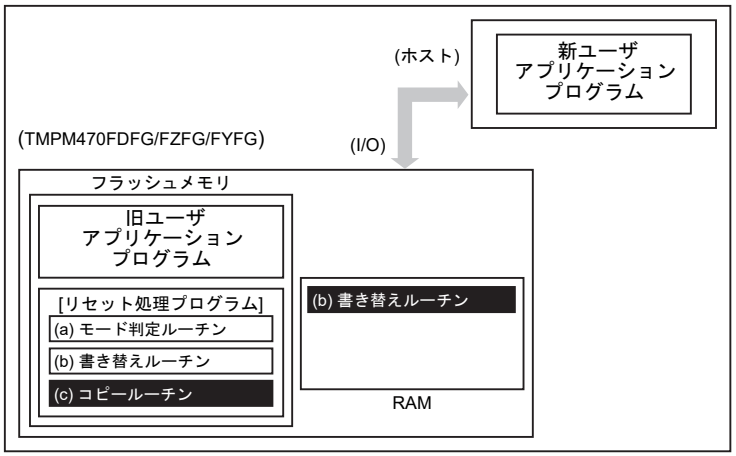
23.4.1.2　Step-2

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、リセット処理プログラムにおいてユーザ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ブートモードに移ります。(ユーザブートモードに移行した場合は、これ以降例外を発生させないでください)



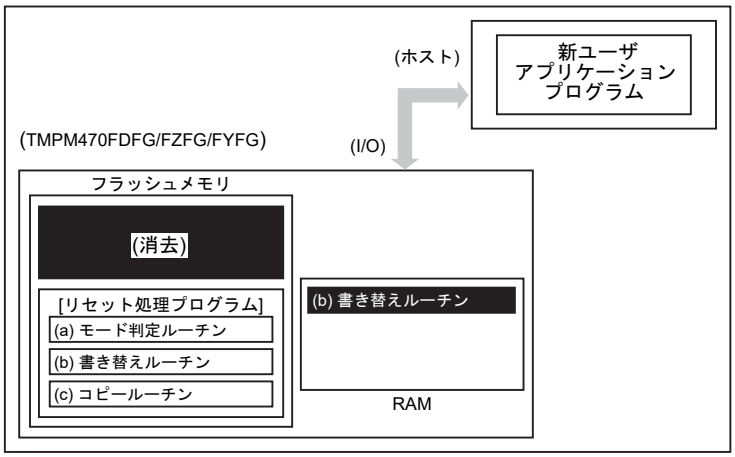
23.4.1.3　Step-3

ユーザ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蔵 RAM にコピーします。



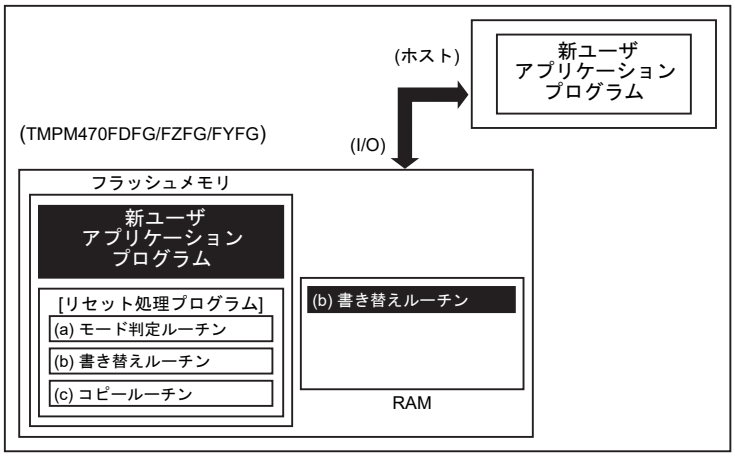
23.4.1.4 Step-4

RAM 上の書き替えルーチンへジャンプし、旧ユーザプログラム領域のライト/消去プロテクトを解除して、消去(任意の消去単位)を行います。



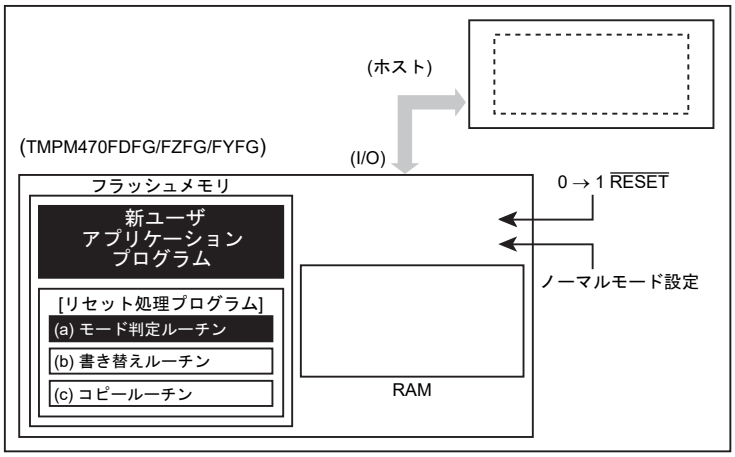
23.4.1.5 Step-5

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領域に書き込みを行います。書き込みが完了したら、ユーザプログラム領域のライト/消去プロテクトをオンにします。



23.4.1.6　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の設定にします。リセット解除後、新ユーザアプリケーションプログラムで動作を開始します。



23.4.2 (1-B)書き替えルーチンを外部から転送する手順例

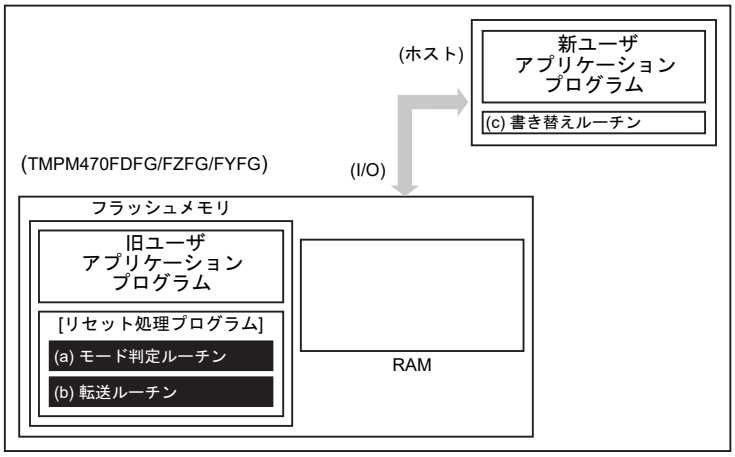
23.4.2.1 Step-1

ユーザは、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

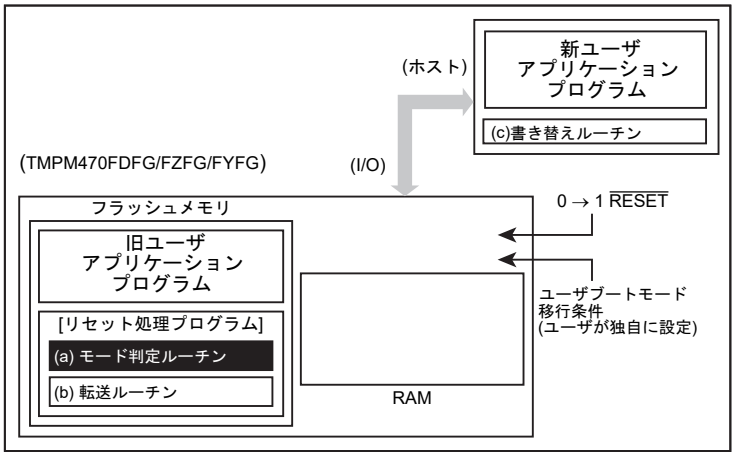
また、下記に示すプログラムはホスト上に用意します。

(c) 書き替えルーチン: 書き替えを行うためのプログラム



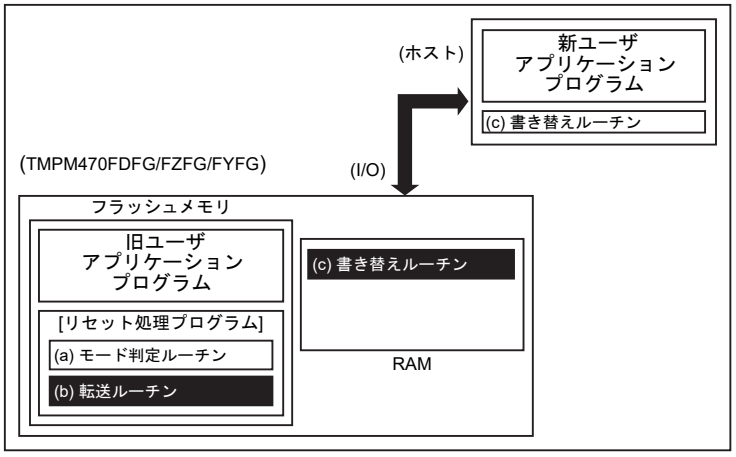
23.4.2.2　Step-2

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ブートモ ードに移ります。(ユーザブートモードに移行した場合は、これ以降例外を発生させないでください)



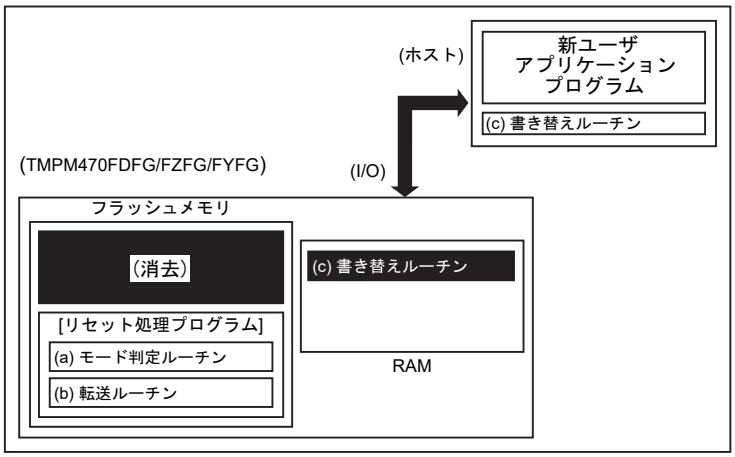
23.4.2.3　Step-3

ユーザ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蔵 RAM にロードします。



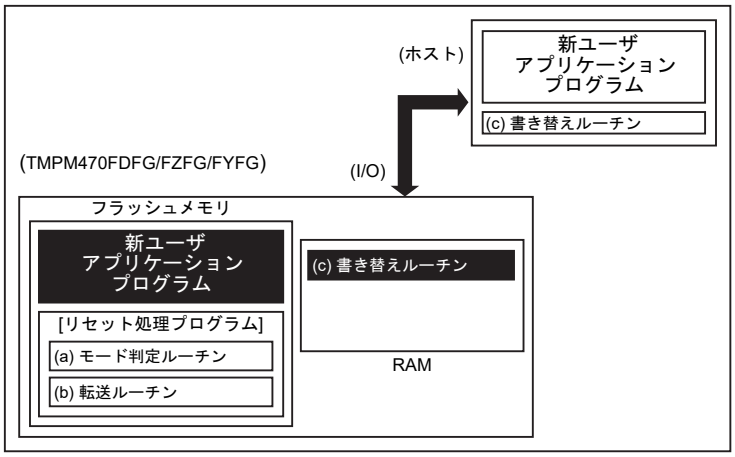
23.4.2.4 Step-4

RAM 上の書き替えルーチンへジャンプし、旧ユーザプログラム領域のライト/消去プロテクトを解除して、消去(任意の消去単位)を行います。



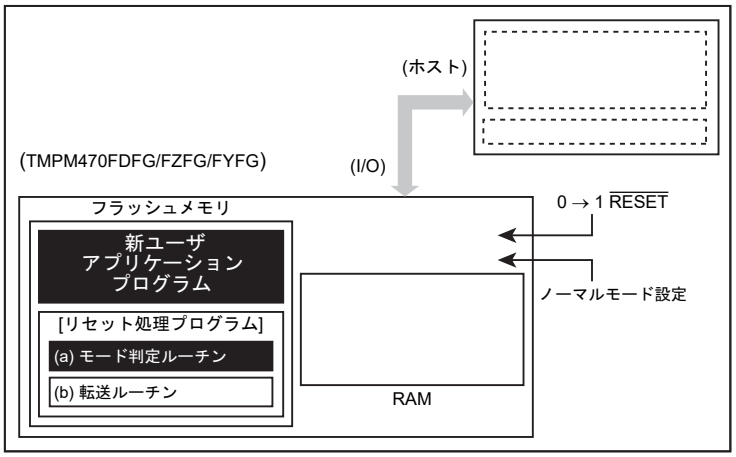
23.4.2.5 Step-5

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)より新ユーザアプリケーションプログラムのデータをロードし、消去した領域に書き込みを行います。書き込みが完了したら、ユーザプログラム領域のライト/消去プロテクトをオンにします。



23.4.2.6　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、新ユーザアプリケーションプログラムで動作を開始します。



23.5 ユーザブートプログラムの書き替え方法

メモリスワップ機能を利用して、ユーザブートプログラムが残るように Page0 と Page1 の領域を交換させてフラッシュメモリの書き替えを実行する方法です。

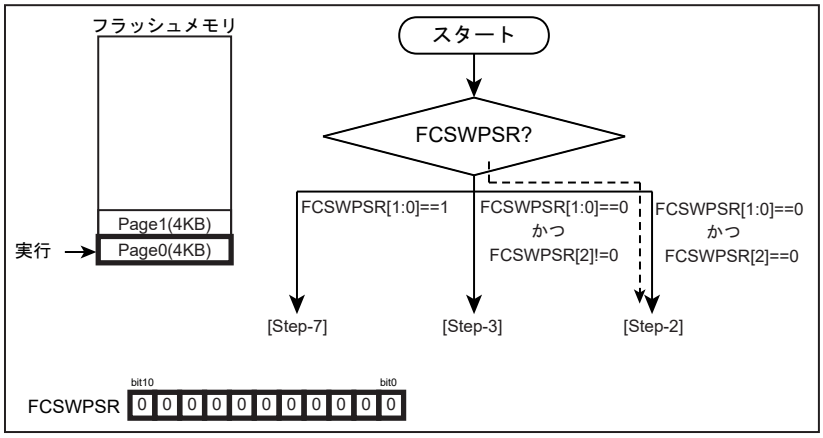
ユーザブートプログラムの書き替え手順の参考例を以下に示します。

(以下では、スワップサイズは 4K バイト、Page1 のプログラムは、Page0 からコピーするものとして、記載します)

23.5.1 フラッシュ書き替えの手順例

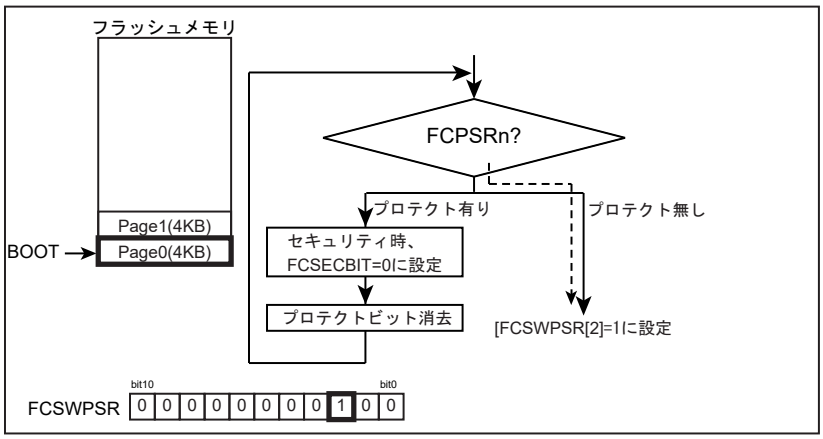
23.5.1.1 Step-1

FCSWPSR[2:0]から 0x0 が読み出せることを確認します。



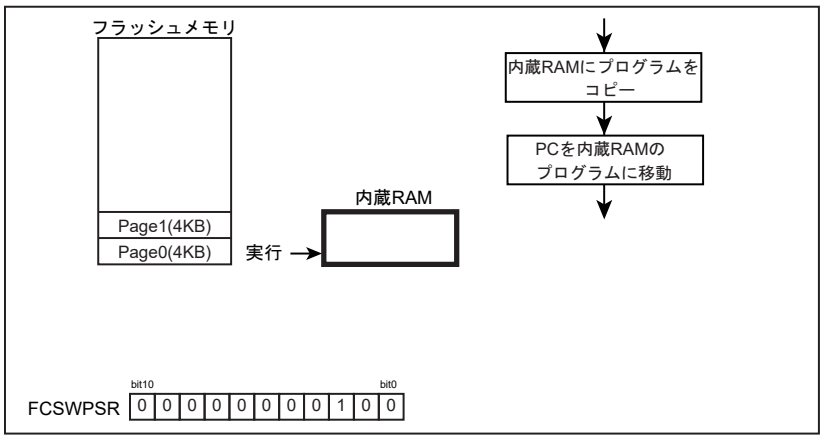
23.5.1.2 Step-2

各 FCPSR レジスタの各ビットにてプロテクト状態をチェックし、プロテクトが無いことを確認します。その後、自動メモリスワップコマンドにて FCSWPSR[2]に"1"をセットします。



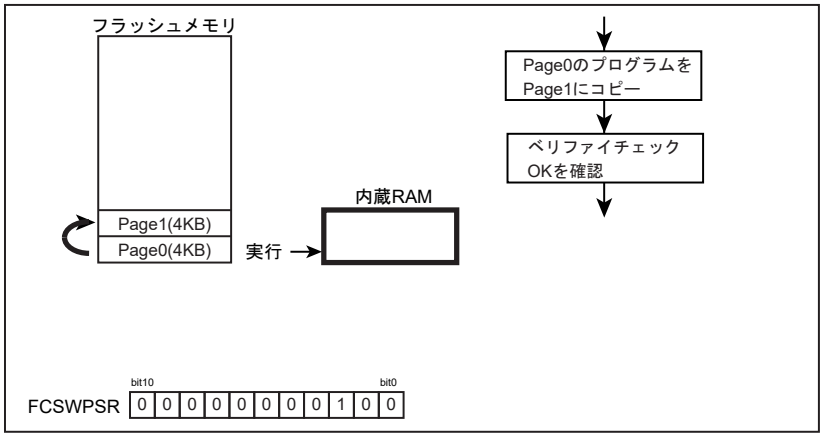
23.5.1.3 Step-3

内蔵 RAM に書き替えルーチンを転送し、PC(プログラムカウンタ)を転送したプログラムに移動します。



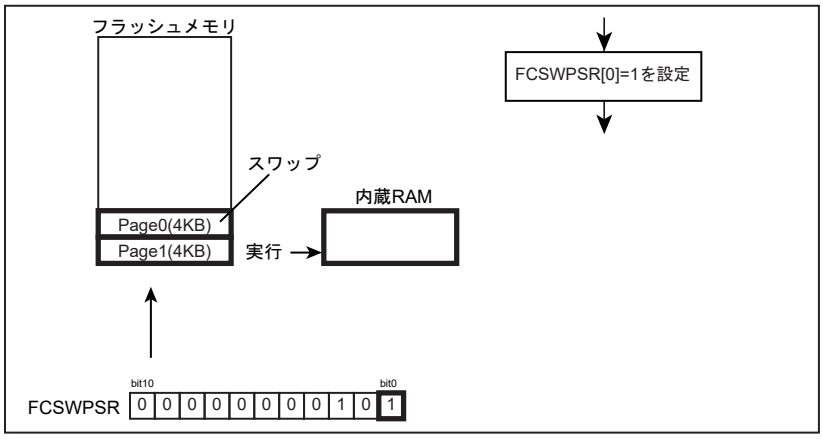
23.5.1.4　Step-4

Page1 を消去し、その後 Page0 のプログラムを Page1 に書き込みます。



23.5.1.5　Step-5

自動メモリスワップコマンドにて FCSWPSR[0]に"1"をセットし、Page0 と Page1 をスワップします。

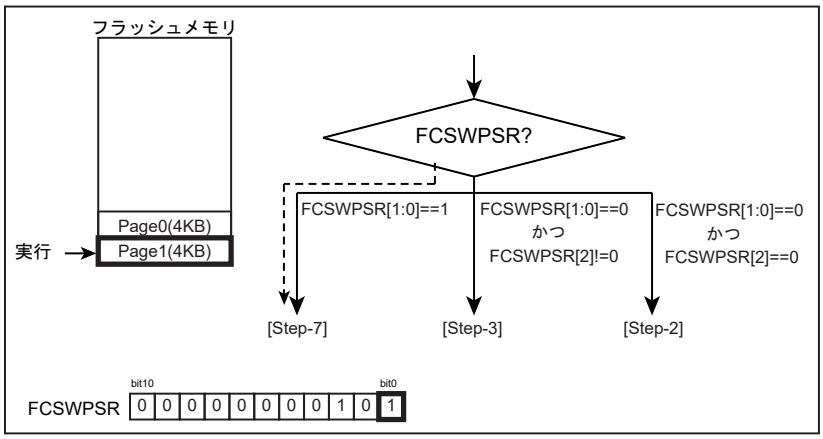


23.5.1.6 Step-6

リセット&リセット解除を行います。

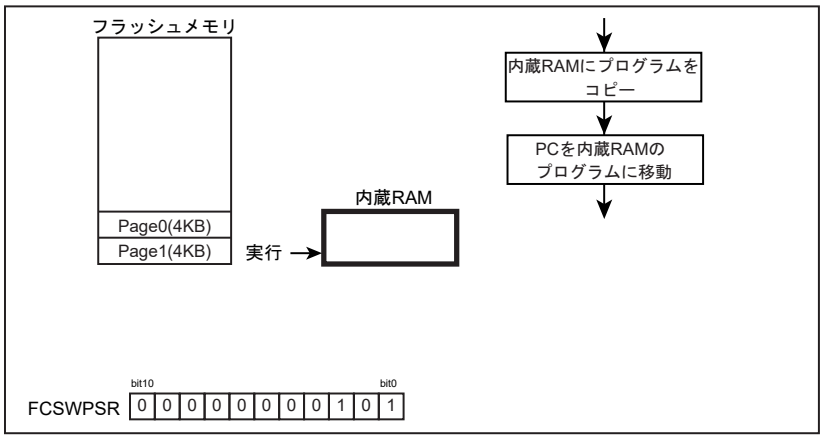
Page1 が 0 番地に割り付けられ、Page1 から起動します。

プログラムは、FCSWPSR[1:0]が"1"の条件用ルーチンへ分岐します。([Step-7]へ)



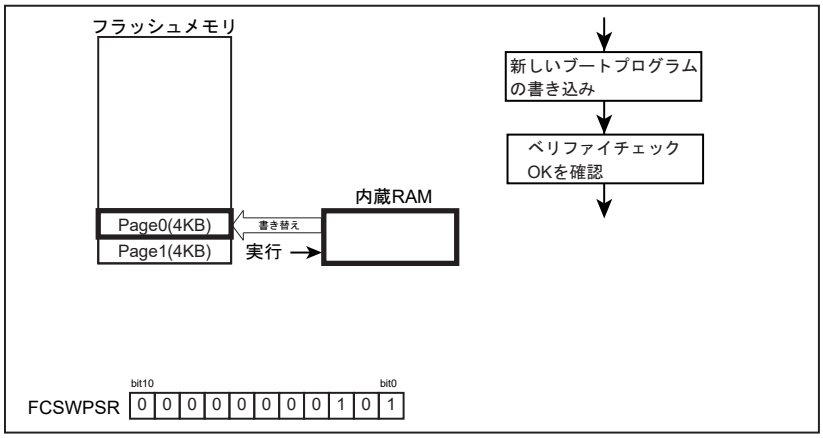
23.5.1.7 Step-7

内蔵 RAM に書き替えルーチンを転送し、PC(プログラムカウンタ)を転送したプログラムに移動します。



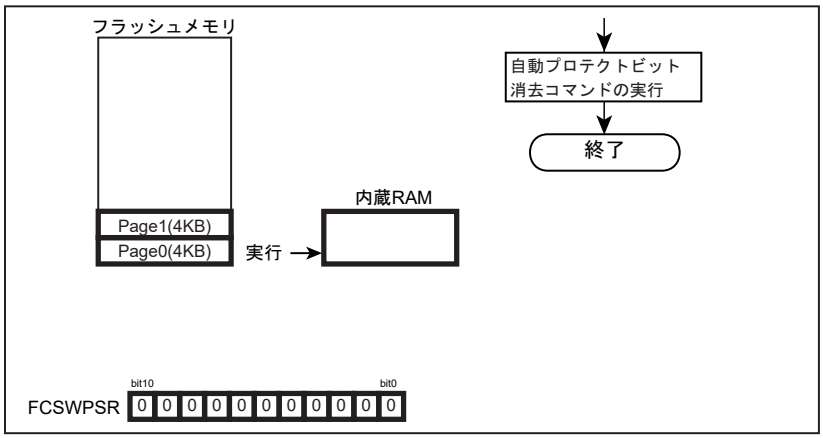
23.5.1.8 Step-8

新しいブートプログラムを Page0 に書き込みます。



23.5.1.9　Step-9

自動プロテクトビット消去コマンドを実行します。



第 24 章　デバッグインタフェース

24.1　仕様概要

TMPM470FDFG/FZFG/FYFG 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(TRACEDATA[3:0], SWV)に出力されます。

SWJ-DP, ETM, TPIU の詳細に関しましては ARM 社からリリースされる"ARM ドキュメンテーションセット Cortex-M4F 用"を参照してください。

24.2　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, $\overline{\text{TRST}}$)をサポートしています。

Pin name	Function	Description	I/O
TMS	JTAG	JTAG Test Mode Selection	Input
SWDIO	SW	Serial Wire Data Input/Output	I/O
TCK	JTAG	JTAG Test Clock	Input
SWCLK	SW	Serial Wire Clock	Input
TDO	JTAG	JTAG Test Data Output	Output
SWV	SW	(Serial Wire Viewer Output)	(Output)(注)
TDI	JTAG	JTAG Test Data Input	Input
$\overline{\text{TRST}}$	JTAG	JTAG Test $\overline{\text{RESET}}$	Input

注)　SWV 機能を許可した場合

24.3　ETM

データ信号 4pin (TRACEDATA[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。

24.4　ホールトモード中の周辺機能

Cortex-M4F 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。また、16 ビットタイマ(TMRB および TMR16A）はホールドモード時に動作するかどうかを指定することができます。その他の周辺機能は動作を続けます。

24.5　デバッグツールとの接続

24.5.1　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。

注)　デバッグツールを接続すると、STOP モード時の消費電流は増加します。

24.5.2　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

デバッグインタフェース端子は汎用ポートと兼用です。

リセット解除後、デバッグ端子となるもの以外は汎用ポート機能となります。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。

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 24-1 デバッグインタフェース端子の使用例

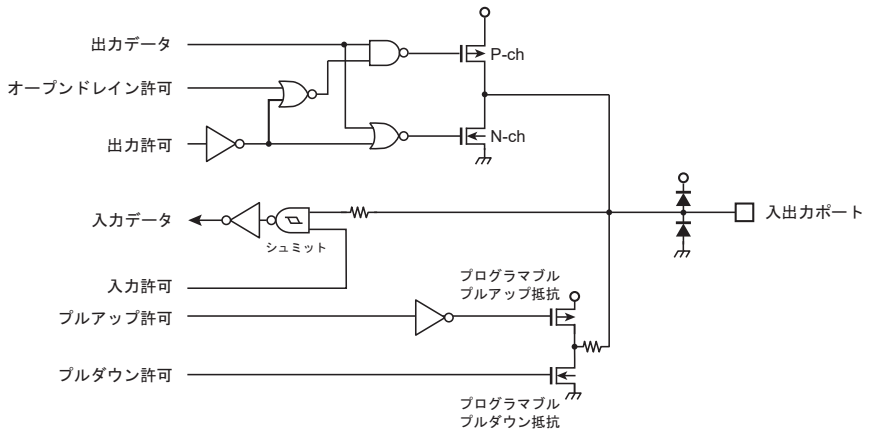
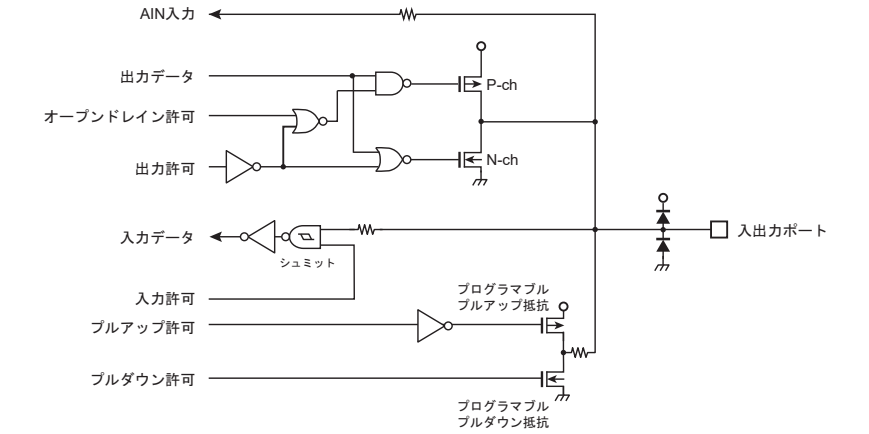
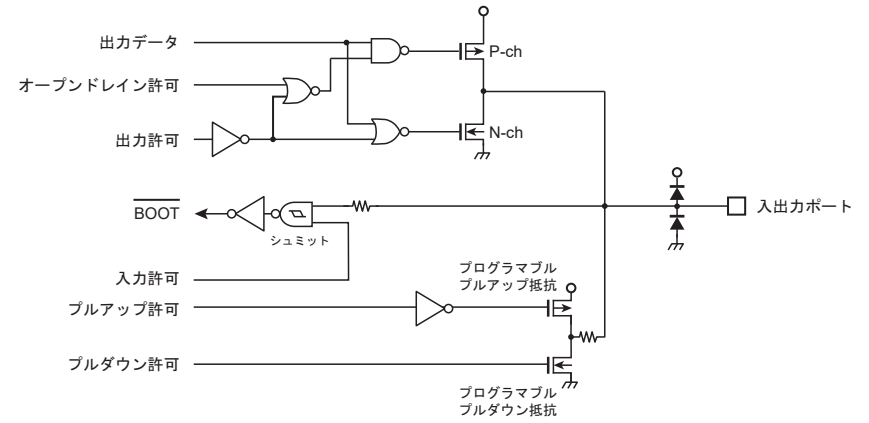
	使用するデバッグインタフェース端子						
	$\overline{\text{TRST}}$	TDI	TDO / SWV	TCK / SWCLK	TMS / SWDIO	TRACE DATA[3:0]	TRACE CLK
JTAG+SW (リセット解除時)	o	o	o	o	o	×	×
JTAG+SW ($\overline{\text{TRST}}$ なし)	× (注)	o	o	o	o	×	×
JTAG+TRACE	o	o	o	o	o	o	o
SW	×	×	×	o	o	×	×
SW+SWV	×	×	o	o	o	×	×
デバッグ機能ディセーブル	×	×	×	×	×	×	×

o：イネーブル ×：ディセーブル(汎用ポートとして使用可能)
注) $\overline{\text{TRST}}$が割り当てられている端子は、$\overline{\text{TRST}}$を選択して、オープンにするか"High"レベルを入力してください。

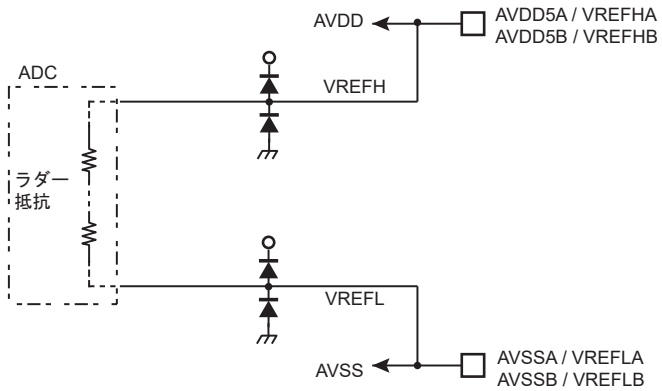
第 25 章　ポート等価回路図

ポート等価回路図は、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。

25.1　ポート

PA0~PA7, PB0~PB7, PC0~PC7, PD0~PD6, PE0~PE7, PF1~PF4, PG0~PG7, PK0~PK1, PL0~PL1, PN0~PN2	(プログラマブル Open Drain 出力、プログラマブル Pull-Up / Pull-Down、Schmitt 入力) 
PH0~PH7, PJ0~PJ7, PP0~PP3	(アナログ入力、プログラマブル Open Drain 出力、プログラマブル Pull-Up / Pull-Down、Schmitt 入力) 
PF0	(Boot 入力、プログラマブル Open Drain 出力、プログラマブル Pull-Up / Pull-Down、Schmitt 入力) 

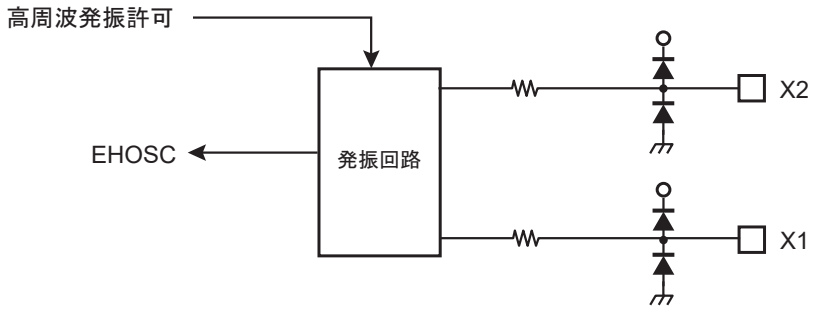
25.2　アナログ端子

VREFHA, VREFLA, VREFHB, VREFLB	

25.3　制御端子

MODE	 MODE 端子は必ず GND に接続して下さい。
$\overline{\text{RESET}}$	

25.4 クロック端子

X1, X2	

第 26 章　電気的特性

26.1　絶対最大定格

項目		記号	定格	単位
電源電圧		DVDD5	−0.3 to 6	V
		RVDD5	−0.3 to 6	
		AVDD5A/B	−0.3 to 6	
電圧保持用キャパシタ端子電圧		VOUT12	−0.3 to 3	V
		VOUT3	−0.3 to 3.9	
入力電圧		V_{IN}	−0.3 ~ VDD + 0.3 (注 2)	V
低レベル出力電流	1 端子	I_{OL}	5	mA
	合計	ΣI_{OL}	50	
高レベル出力電流	1 端子	I_{OH}	−5	
	合計	ΣI_{OH}	−50	
消費電力 (Ta = 85 °C)		PD	600	mW
はんだ付け温度(10 s)		T_{SOLDER}	260	°C
保存温度		T_{STG}	−55 ~ 125	°C
動作温度		T_{OPR}	−40 ~ 85	°C

注 1) **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。**

注 2) **VDD = DVDD5 = RVDD5 = AVDD5A = AVDD5B**

26.2　DC 電気的特性 (1/2)

DVSS = AVSSA = AVSSB = 0V, Ta = −40 ~ 85 °C

項目		記号	条件		Min	Typ. (注1)	Max	単位
電源電圧 (注 2)	DVDD5 RVDD5 AVDD5A AVDD5B	VDD	f_{OSC} = 8 ~ 10 MHz fsys = 1 ~ 120 MHz		4.5	−	5.5	V
電源電圧 (FLASH W/E 時) (注 2)	DVDD5 RVDD5 AVDD5A AVDD5B	VDD	f_{OSC} = 8 ~ 10 MHz fsys = 1 ~ 120 MHz (Ta (°C) = 0 ~ 70)		4.5	−	5.5	V
電源電圧 (電源上昇、下降時) (注 3)	DVDD5 RVDD5 AVDD5A AVDD5B	VDD	f_{OSC} = 8 ~ 10 MHz fsys = 1 ~ 120 MHz		3.9	−	5.5	V
低レベル入力電圧	シュミット入力	V_{IL1}	VDD = 4.5V ~ 5.5V (注 4)		−0.3	−	0.25 VDD	V
高レベル入力電圧	シュミット入力	V_{IH1}	VDD = 4.5V ~ 5.5V (注 4)		0.75VDD		VDD+0.3	V
電圧保持用キャパシタ容量(注 5)		C_{out}	RVDD5 = 4.5V ~ 5.5V VOUT12, VOUT3		3.3	−	4.7	μF
低レベル出力電圧		V_{OL}	I_{OL} = 1.6 mA	VDD ≥ 4.5V (注 4)	−	−	0.4	V
高レベル出力電圧		V_{OH}	I_{OH} = −1.6 mA	VDD ≥ 4.5V (注 4)	4.1	−	−	V
入力リーク電流		I_{LI1}	0.0 ≤ V_{IN} ≤ VDD (注 4)		−	0.02	±5	μA
出力リーク電流		I_{LO}	0.2 ≤ V_{IN} ≤ VDD -0.2 (注 4)		−	0.05	±10	
リセットプルアップ抵抗		R_{RST}	4.5 ≤ VDD ≤ 5.5 (注 4)		−	50	150	kΩ
プログラマブルプルアップ/ダウン抵抗		P_{KH}	4.5 ≤ VDD ≤ 5.5 (注 4)		−	50	150	kΩ
シュミット入力幅		VTH	4.5 ≤ VDD ≤ 5.5 (注 4)		0.3	0.6	−	V
Pin 容量(電源端子を除く)		C_{IO}	fc = 1 MHz		−	−	10	pF

注 1)　Typ. 値は特に指定のない限り Ta = 25 °C, DVDD5 = AVDD5A = AVDD5B = RVDD5 = 5V の値です。

注 2)　DVDD5、AVDD5A、AVDD5B、RVDD5 は同電圧で使用してください。

注 3)　電源上昇時（パワーオン時）と電圧検出回路（VLTD)イネーブルでの電源電圧下降時における電圧範囲となります。
なお、3.9V ≤ VDD < 4.5V の範囲では 12 ビット A/D コンバータ変換特性および AC 電気的特性は保証外となります。

注 4)　VDD = DVDD5 = RVDD5 = AVDD5A = AVDD5B

注 5)　VOUT12 端子と VOUT3 端子は同値の保持用キャパシタを介して GND に接続してください。VOUT12 端子と VOUT3 端子から IC 外部への電源供給はできません。

26.3 DC 電気的特性 (2/2)

DVDD5 =RVDD5 = AVDD5A = AVDD5B = 4.5 V ~ 5.5 V, Ta = −40 ~ 85 °C

項目	記号	条件		Min	Typ. (注 1)	Max	単位
		システムクロック (fsys)	動作条件				
NORMAL	I_{DD}	120 MHz	表 26-1 と表 26-2 を参照してください。	−	45	100	mA
IDLE				−	13	50	
NORMAL		80 MHz		−	34	75	
IDLE				−	9.8	38	
STOP		停止		−	1.6	25	

注 1) Typ. 値は、特に指定のない限り Ta=25°C、DVDD5 = AVDD5A = AVDD5B = RVDD5 = 5V です。
注 2) 80MHz 時の IDD は参考値です。

表 26-1 I_{DD} 測定条件 (端子設定、発振回路)

		NORMAL	IDLE	STOP
端子設定	DVDD5 = AVDD5A = AVDD5B = RVDD5	4.5 ~ 5.5 V		
	x1, X2 端子	発振子接続		
	入力端子	固定		
	出力端子	開放		
動作条件 (発振回路)	外部高速発振器(EOSC)	発振		停止
	内部高速発振器 (IOSC)	停止		
	fsys 用 PLL	動作(12 逓倍)		停止

表 26-2 I_{DD} 測定条件 (CPU、周辺回路)

回路	搭載回路数	NORMAL	IDLE	STOP
CPU	1	動作 (ドライストン Ver. 2.1)	停止	停止
A-VE	2	動作	停止	停止
µDMAC	1	停止	停止	停止
ADC	2	動作	停止	停止
TMRB	10	動作	停止	停止
WDT	1	停止	停止	停止
SIO/UART	4	動作	停止	停止
I2C	1	停止	停止	停止
PMD	2	動作	停止	停止
A-ENC	2	動作	停止	停止
I/O port	-	動作	停止	停止
VLTD	1	動作	動作	停止
OFD	1	動作	停止	停止

26.4　12 ビット A/D コンバータ変換特性

DVDD5 = RVDD5 = AVDD5A / VREFHA = AVDD5B / VREFHB = 4.5 V ~ 5.5 V
DVSS = AVSSA / VREFLA = AVSSB / VREFLB = 0V, Ta = −40 ~ 85 °C

項目	記号	条件	Min	Typ.	Max	単位
アナログ基準電圧(+)	VREFHA VREFHB	−	−	AVDD	−	V
アナログ入力電圧	VAIN	−	AVSS	−	AVDD	V
アナログ基準電圧電源電流 (注 1)	IREF	DVSS = AVSS	−	3.5	5.0	mA
消費電流(注 1)　A/D 変換時	−	IREF を除く	−	−	6.0	mA
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 ≥ 1.00 μs 変換クロック周波数=120MHz (注 4)	−	−	± 6	LSB
微分非直線性誤差			−	−	± 5	
オフセット誤差			−	−	± 5	
フルスケール誤差			−	−	± 6	
総合誤差			−	−	−10 ~ +6	

注 1)　AD コンバータ 1 ユニットの電流です。
注 2)　1LSB = (AVDD − AVSS)/4096 [V]
注 3)　AVDD = AVDD5A = AVDD5B、AVSS = AVSSA = AVSSB
注 4)　2 ユニット AD コンバータのみ動作時の特性です。

26.5 AC 電気的特性

26.5.1 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 × VDD、Low = 0.2 × VDD
- 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

注) VDD = DVDD5 = AVDD5A = AVDD5B

26.5.2 シリアルチャネル (SIO/UART)

26.5.2.1 I/O インタフェースモード (VDD = 4.5 ~ 5.5V)

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 (Ta = − 40 ~ 85°C)

[データ入力]

項目	記号	計算式		80 MHz		120 MHz		単位
		Min	Max	Min	Max	Min	Max	
SCLK クロック High 幅(入力)	t_{SCH}	4x	−	50	−	33.3	−	ns
SCLK クロック Low 幅(入力)	t_{SCL}	4x	−	50	−	33.3	−	
SCLK 周期	t_{SCY}	8x	−	100	−	66.6	−	
有効 Data 入力 ← SCLK 立ち上がり/立ち下がり (注 1)	t_{SRD}	30	−	30	−	30	−	
SCLK 立ち上がり/立ち下がり(注 1) → Input Data 保持	t_{HSR}	x + 30	−	42.5	−	38.3	−	

[データ出力]

項目	記号	計算式		80 MHz		120 MHz		単位
		Min	Max	Min	Max	Min	Max	
SCLK クロック High 幅(入力)	t_{SCH}	4x	−	82.5 (注 3)	−	70.0 (注 3)	−	ns
SCLK クロック Low 幅(入力)	t_{SCL}	4x	−	82.5 (注 3)	−	70.0 (注 3)	−	
SCLK 周期	t_{SCY}	8x	−	165	−	140	−	
Output Data ← SCLK 立ち上がり/立ち下がり (注 1)	t_{OSS}	t_{SCY} / 2 - 3x - 45 (注 2)	−	0 (注 2)	−	0 (注 2)	−	
SCLK 立ち上がり/立ち下がり(注 1) → Output Data 保持	t_{OHS}	t_{SCY} / 2	−	82.5	−	70.0	−	

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 周期で使用して下さい。

注 3) t_{OSS} がマイナスにならない最小値を示しています。計算式による値ではありません。

(2)　SCLK 出力モード (Ta = − 40 ~ 85°C)

[データ入出力]

項目	記号	計算式		80 to 120 MHz		単位
		Min	Max	Min	Max	
SCLK 周期(プログラマブル) (注 3)	t_{SCY}	4x	−	50	−	ns
Output Data ← SCLK 立ち上がり/立ち下がり	t_{OSS}	t_{SCY}/2 − 25 （注 1）	−	0 (注 2)	−	
SCLK 立ち上がり/立ち下がり→ Output Data 保持	t_{OHS}	t_{SCY}/2 − 25 （注 1）	−	0 (注 2)	−	
有効 Data 入力 ← SCLK 立ち上がり/立ち下がり	t_{SRD}	45	−	45	−	
SCLK 立ち上がり/立ち下がり→ input Data 保持	t_{HSR}	0	−	0	−	

注 1)　計算値がマイナスにならない範囲の SCLK 周期で使用して下さい。
注 2)　マイナスにならない最小値を示しています。計算値ではありません。
注 3)　SCLK の周期 ≥ 50ns となるように、SCLK を調整してください。

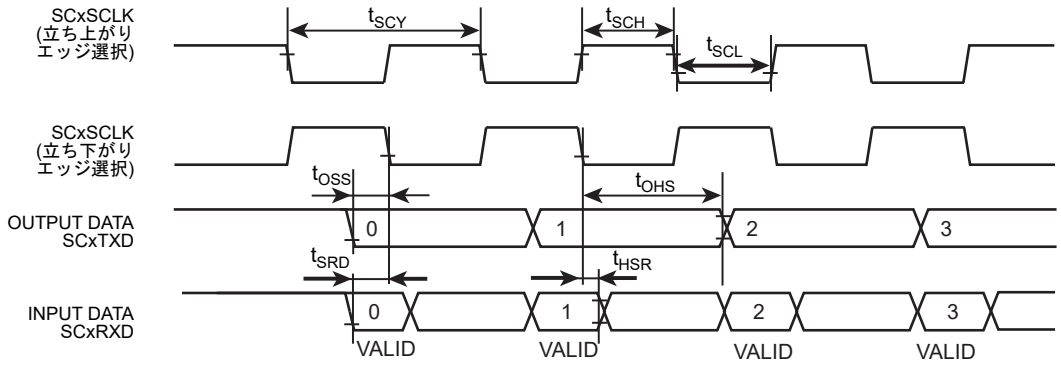


図 26-1 シリアルチャネルのタイミング (SIO)

26.5.3 シリアルバスインタフェース (I2C)

26.5.3.1 I2C モード

表中の 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_{SCL}	0	−	0	100	0	400	kHz
スタートコンディション保持	$t_{HD;\ STA}$	−	−	4.0	−	0.6	−	μs
SCL クロック Low 幅(入力) (注 1)	t_{LOW}	−	−	4.7	−	1.3	−	μs
SCL クロック High 幅(入力) (注 2)	t_{HIGH}	−	−	4.0	−	0.6	−	μs
再スタートコンディション セットアップ時間	$t_{SU;\ STA}$	(注 5)	−	4.7	−	0.6	−	μs
データ保持時間(入力) (注 3, 4)	$t_{HD;\ DAT}$	−	−	0.0	−	0.0	−	μs
データセットアップ時間	$t_{SU;\ DAT}$	−	−	250	−	100	−	ns
ストップコンディションセットアップ時間	$t_{SU;\ STO}$	−	−	4.0	−	0.6	−	μs
ストップコンディションとスタート コンディション間のバスフリー時間	t_{BUF}	(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): $(2^{n-1} + 14)/x$

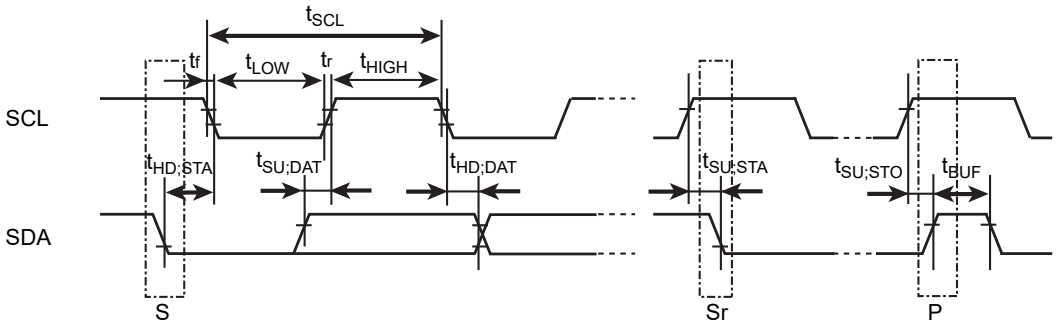
通信規格上、標準モード/高速モードの最高速は 100 kHz/400 kHz です。内部 SCL クロックの周波数の設定は、使用される fsys と上記注 1,注 2 の計算式にて設定されますのでご注意願います。

注 3) データ保持時間(出力)は内部 SCL から 4x の時間です。

注 4) フィリップススペックでは内部で、SDA 入力時にデータホールド 300nsec を確保して SCL 立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 のエッジスロープ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) ソフトウエアに依存します。

注 6) フィリップススペックでは、ファーストモードデバイスへの電源供給が切れた場合、バスラインを妨げることがないように SDA および SCL 信号の I/O ピンと電源との接続が外れるようにする必要がありますが、本デバイスでは対応していません。



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

図 26-2 シリアルバスのタイミング(I2C)

26.5.4 16 ビットタイマ／カウンタ(TMRB)

26.5.4.1 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 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。

項目	記号	計算式		80 MHz		120 MHz		単位
		Min	Max	Min	Max	Min	Max	
クロック低レベルパルス幅	t_{VCKL}	2x + 100	−	125	−	117	−	ns
クロック高レベルパルス幅	t_{VCKH}	2x + 100	−	125	−	117	−	ns

26.5.4.2 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 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。

項目	記号	計算式		80 MHz		120 MHz		単位
		Min	Max	Min	Max	Min	Max	
低レベルパルス幅	t_{CPL}	2x + 100	−	125	−	117	−	ns
高レベルパルス幅	t_{CPH}	2x + 100	−	125	−	117	−	ns

26.5.5 外部割り込み

表中の x はシステムクロック fsys 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		80 MHz		120 MHz		単位
		Min	Max	Min	Max	Min	Max	
INT0 ~ F 低レベルパルス幅	t_{INTAL}	x + 100	−	112.5	−	108.3	−	ns
INT0 ~ F 高レベルパルス幅	t_{INTAH}	x + 100	−	112.5	−	108.3	−	ns

2. STOP 解除割り込み

項目	記号	計算式		80 MHz		120 MHz		単位
		Min	Max	Min	Max	Min	Max	
INT0 ~ F 低レベルパルス幅	t_{INTBL}	100	−	100	−	100	−	ns
INT0 ~ F 高レベルパルス幅	t_{INTBH}	100	−	100	−	100	−	ns

26.5.6　デバッグ通信

26.5.6.1　AC 測定条件

- 出力レベル: High = 0.7 × DVDD5, Low = 0.3 × DVDD5
- 負荷容量: TRACECLK CL = 25pF、TRACEDATA CL = 20pF

26.5.6.2　SWD インタフェース

項目	記号	Min	Max	単位
CLK 周期	T_{dck}	100	–	ns
CLK 立上がり → 出力データ保持	T_{d1}	4	–	
CLK 立上がり → 出力データ有効	T_{d2}	–	30	
入力データ有効 → CLK 立上がり	T_{ds}	20	–	
CLK 立上がり → 入力データ保持	T_{dh}	15	–	

26.5.6.3　JTAG インタフェース

項目	記号	Min	Max	単位
CLK 周期	T_{dck}	100	–	ns
CLK 立下がり → 出力データ保持	T_{d3}	4	–	
CLK 立下がり → 出力データ有効	T_{d4}	–	50	
入力データ有効 → CLK 立上がり	T_{ds}	20	–	
CLK 立上がり → 入力データ保持	T_{dh}	15	–	

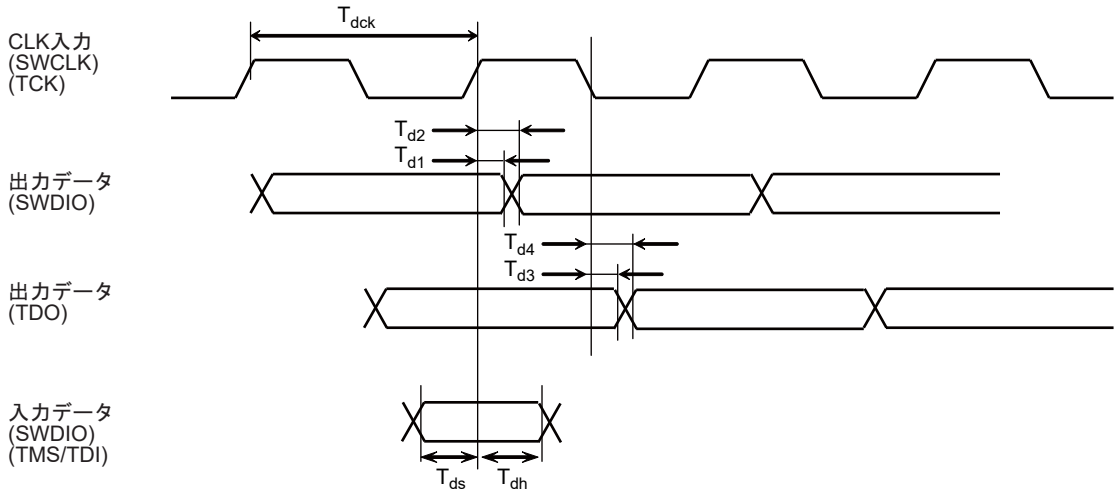


図 26-3 SWD/JTAG インタフェースのタイミング

26.5.7 ETM トレース

AC 測定条件は以下の通りです。

- 出力レベル: High = 0.7 × DVDD5, Low = 0.3 × DVDD5
- 負荷容量: TRACECLK CL = 25pF、TRACEDATA CL = 20pF

項目	記号	Min	Max	単位
TRACECLK 周期	t_{tclk}	33.3	−	ns
TRACEDATA 有効 ← TRACECLK 立上がり	t_{setupr}	2	−	
TRACECLK 立上がり → TRACEDATA 保持	t_{holdr}	1	−	
TRACEDATA 有効 ← TRACECLK 立下がり	t_{setupf}	2	−	
TRACECLK 立下がり → TRACEDATA 保持	t_{holdf}	1	−	

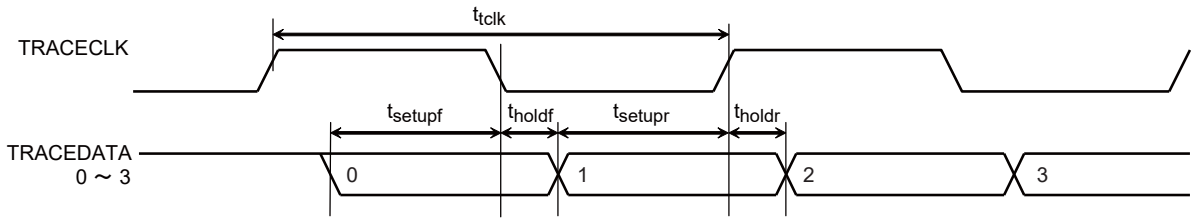


図 26-4 EMT トレースタイミング

26.5.8　Flash 特性

項目	条件	Min	Typ.	Max	単位
フラッシュメモリ 消去/書き込み回数	Ta = -40 to 85°C DVDD5 = RVDD5 = AVDD5A = AVDD5B = 4.5 ~ 5.5V	−	−	1000	回

26.5.9　内蔵発振回路特性

項目	記号	条件	Min	Typ.	Max	単位
発振周波数 (注 1)	fiosc	Ta = -40 to 85°C	−	−	10	MHz
発振周波数精度 (注 1), (注 2)	−	Ta = 0 to 50°C	−	−	± 1	%
		Ta = -40 to 0°C, Ta = 50 to 85°C	−	−	± 2	%

注 1)　弊社出荷テスト時の値です。リフロー実装直後とそれ以降において、記載の発振精度が必要な場合には、12 逓倍 PLL 設定時の最大動作周波数 120MHz を超えないように、発振周波数をその都度トリミングしてください。

注 2)　DVDD5 = RVDD5 = AVDD5A = AVDD5B =5.0V, Ta=25℃でのトリミング実施後の発振周波数に対する誤差です。

26.5.10　外部発振子

項目	記号	条件	Min	Typ.	Max	単位
高周波発振	feosc	Ta = -40 to 85°C	−	10	−	MHz

26.6 発振回路

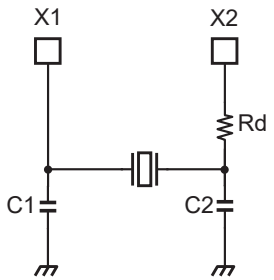


図 26-5 高周波発振回路例

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。

26.6.1 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。

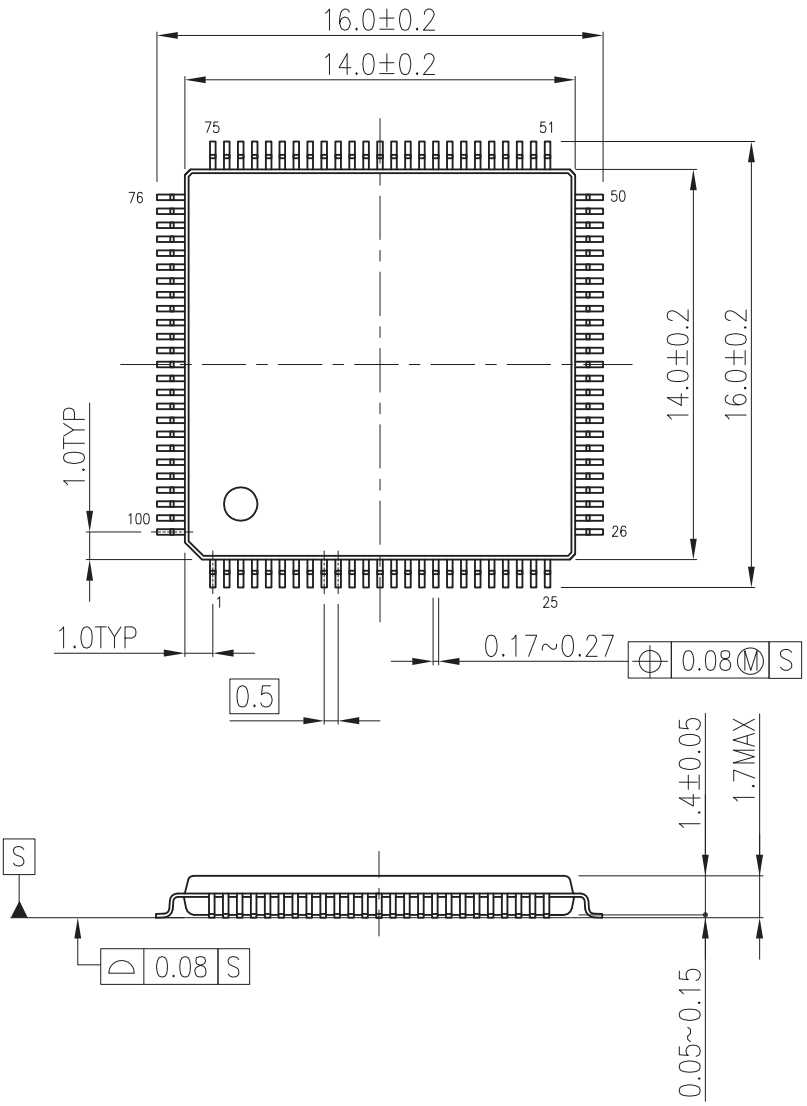
(株)村田製作所の製品詳細につきましては、同社ホームページを参照してください。

第 27 章　パッケージ寸法図

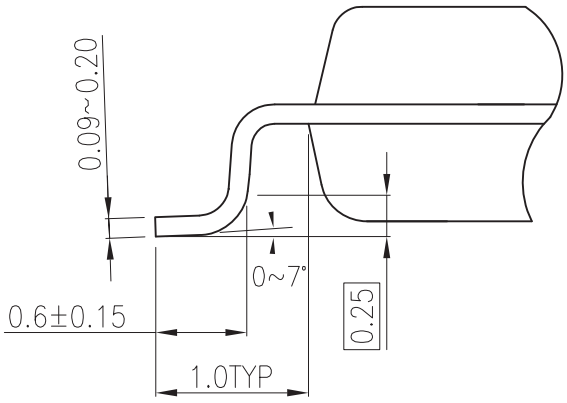
パッケージ型名：P-LQFP100-1414-0.50-002

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

株式会社東芝およびその子会社ならびに関係会社を以下「当社」といいます。
本資料に掲載されているハードウエア、ソフトウエアおよびシステムを以下「本製品」といいます。

- 本製品に関する情報等、本資料の掲載内容は、技術の進歩などにより予告なしに変更されることがあります。
- 文書による当社の事前の承諾なしに本資料の転載複製を禁じます。また、文書による当社の事前の承諾を得て本資料を転載複製する場合でも、記載内容に一切変更を加えたり、削除したりしないでください。
- 当社は品質、信頼性の向上に努めていますが、半導体・ストレージ製品は一般に誤作動または故障する場合があります。本製品をご使用頂く場合は、本製品の誤作動や故障により生命・身体・財産が侵害されることのないように、お客様の責任において、お客様のハードウエア・ソフトウエア・システムに必要な安全設計を行うことをお願いします。なお、設計および使用に際しては、本製品に関する最新の情報（本資料、仕様書、データシート、アプリケーションノート、半導体信頼性ハンドブックなど）および本製品が使用される機器の取扱説明書、操作説明書などをご確認の上、これに従ってください。また、上記資料などに記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を使用する場合は、お客様の製品単独およびシステム全体で十分に評価し、お客様の責任において適用可否を判断してください。
- **本製品は、特別に高い品質・信頼性が要求され、またはその故障や誤作動が生命・身体に危害を及ぼす恐れ、膨大な財産損害を引き起こす恐れ、もしくは社会に深刻な影響を及ぼす恐れのある機器（以下“特定用途”という）に使用されることは意図されていませんし、保証もされていません。特定用途には原子力関連機器、航空・宇宙機器、医療機器（生命直結機器）、車載・輸送機器、防衛関連機器などが含まれますが、本資料に個別に記載する用途は除きます。特定用途に使用された場合には、当社は一切の責任を負いません。なお、詳細は当社営業窓口まで、または当社 Web サイトのお問い合わせフォームからお問い合わせください。**
- 本製品を分解、解析、リバースエンジニアリング、改造、改変、翻案、複製等しないでください。
- 本製品を、国内外の法令、規則及び命令により、製造、使用、販売を禁止されている製品に使用することはできません。
- 本資料に掲載してある技術情報は、製品の代表的動作・応用を説明するためのもので、その使用に際して当社及び第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。
- 別途、書面による契約またはお客様と当社が合意した仕様書がない限り、当社は、本製品および技術情報に関して、明示的にも黙示的にも一切の保証（機能動作の保証、商品性の保証、特定目的への合致の保証、情報の正確性の保証、第三者の権利の非侵害保証を含むがこれに限らない。）をしておりません。
- 本製品、または本資料に掲載されている技術情報を、大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」、「米国輸出管理規則」等、適用ある輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
- 本製品のRoHS適合性など、詳細につきましては製品個別に必ず当社営業窓口までお問い合わせください。本製品のご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用ある環境関連法令を十分調査の上、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関して、当社は一切の責任を負いかねます。