

**TOSHIBA**

32 ビット RISC マイクロコントローラ  
TX03 シリーズ

TMPM376FDDFG/TMPM376FDFG

株式会社 **東芝**  
セミコンダクター & ストレージ社



## 改訂履歴

日付	版	改訂内容
2011/12/28	1	First Release
2013/04/08	2	Contents Revised
2021/10/15	3	Contents Revised
2022/03/31	4	Contents Revised
2022/06/01	5	Contents Revised
2023/07/21	6	Contents Revised
2023/07/31	7	Contents Revised



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



# TMPM376FDDFG/FDFG

TMPM376FDDFG/FDFGは、ARM社Cortex™-M3コアを内蔵した32ビットRISCマイクロプロセッサです。

製品名	ROM (FLASH)	RAM	パッケージ
TMPM376FDDFG	512 Kbyte	32 Kbyte	QFP100-P-1420-0.65Q
TMPM376FDFG	512 Kbyte	32 Kbyte	LQFP100-P-1414-0.50H

機能概要と特長は次のとおりです。

## 1.1 機能概要

### 1. ARM 社製 Cortex-M3 コアを使用

- a. Thumb®-2 命令で、コード効率の向上を実現
  - ・プログラムフロー改善のための新しい 16 ビット命令
  - ・性能とコードサイズ向上のための新しい 32 ビット命令
  - ・32 ビット /16 ビット混在の命令セットでコード効率を向上
- b. 高性能化と低消費電力化を同時に実現
  - 【高性能化】
    - ・32 ビット乗算 ( $32 \times 32 = 32$  ビット) を 1 クロックで実行
    - ・除算を 2~12 クロックで実行
  - 【低消費電力化】
    - ・低消費電力ライブラリを使用した最適化設計
    - ・プロセッサコアの動作を停止させるスタンバイ機能
- c. リアルタイム制御に向けた高速割り込み応答
  - ・実行時間の長い命令は割り込みで中断可能
  - ・スタックへの PUSH をハードウェアで自動的に実行

### 2. 内蔵プログラムメモリ / データメモリ

- ・内蔵 RAM : 32Kbyte
- ・内蔵 FlashROM : 512Kbyte

### 3. 16 ビットタイマ (TMRB) : 8 チャンネル

- ・16 ビットインタバルタイマモード
- ・16 ビットイベントカウンタモード
- ・インプットキャプチャ機能
- ・外部トリガ PPG 出力

### 4. ウォッチドッグタイマ (WDT) : 1 チャンネル

リセットまたはマスク不能割り込み (NMI) 発生

5. パワーオンリセット回路 (POR)
6. 電圧検出回路 (VLTD)
7. 周波数検知回路 (OFD)
8. ベクトルエンジン (VE) : 1 ユニット
  - モータ制御用演算機能
  - 2 モータ対応
9. モータ制御回路 (PMD) : 2 チャンネル
  - 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)
10. エンコーダ入力回路 (ENC) : 2 チャンネル
  - インクリメンタル形エンコーダ対応 (AB 信号 / ABZ 信号)
  - 回転方向検出回路
  - 絶対位置検出カウンタ
  - 位置コンペア回路
  - ノイズフィルタ内蔵
  - 3 相センサ入力対応
11. 汎用シリアルインターフェース (SIO/UART) : 4 チャンネル
  - UART/ クロック同期式モード選択可能 (4byte FIFO 内蔵)
12. シリアルバスインタフェース (I2C/SIO) : 1 チャンネル
  - I2C バスモード / クロック同期式モード選択可能
13. 12 ビット AD コンバータ (ADC) : 2 ユニット (アナログ入力 : 22 チャンネル)
  - 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 2  $\mu$ sec (ADC 変換クロック 40 MHz 時)
14. 入出力ポート (PORT) : 82 端子
  - 入出力端子 : 80 本
  - 入力端子 : 2 本
15. 割り込み機能
  - 内部 63 本 : 7 レベルの優先順位設定可能 (ウォッチドッグタイマ割り込みを除く)



- 外部 16 本 : 7 レベルの優先順位設定可能

#### 16. スタンバイ機能

スタンバイモード : IDLE, STOP

#### 17. クロックジェネレータ (CG)

- PLL 内蔵 (8 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

#### 18. エンディアン

リトルエンディアン

#### 19. 内蔵高速発振器

#### 20. 最大動作周波数 : 80 MHz

#### 21. 動作電圧範囲

4.5 V~5.5 V (内蔵レギュレータ使用)

#### 22. 温度範囲

- -40°C ~ 85°C (Flash W/E およびデバッグ時以外)
- 0°C ~ 70°C (Flash W/E 時)

#### 23. パッケージ

- QFP100-P-1420-0.65Q (14 mm × 20 mm, 0.65 mm ピッチ)
- LQFP100-P-1414-0.50H (14 mm × 14 mm, 0.5 mm ピッチ)

## 1.2 ブロック図

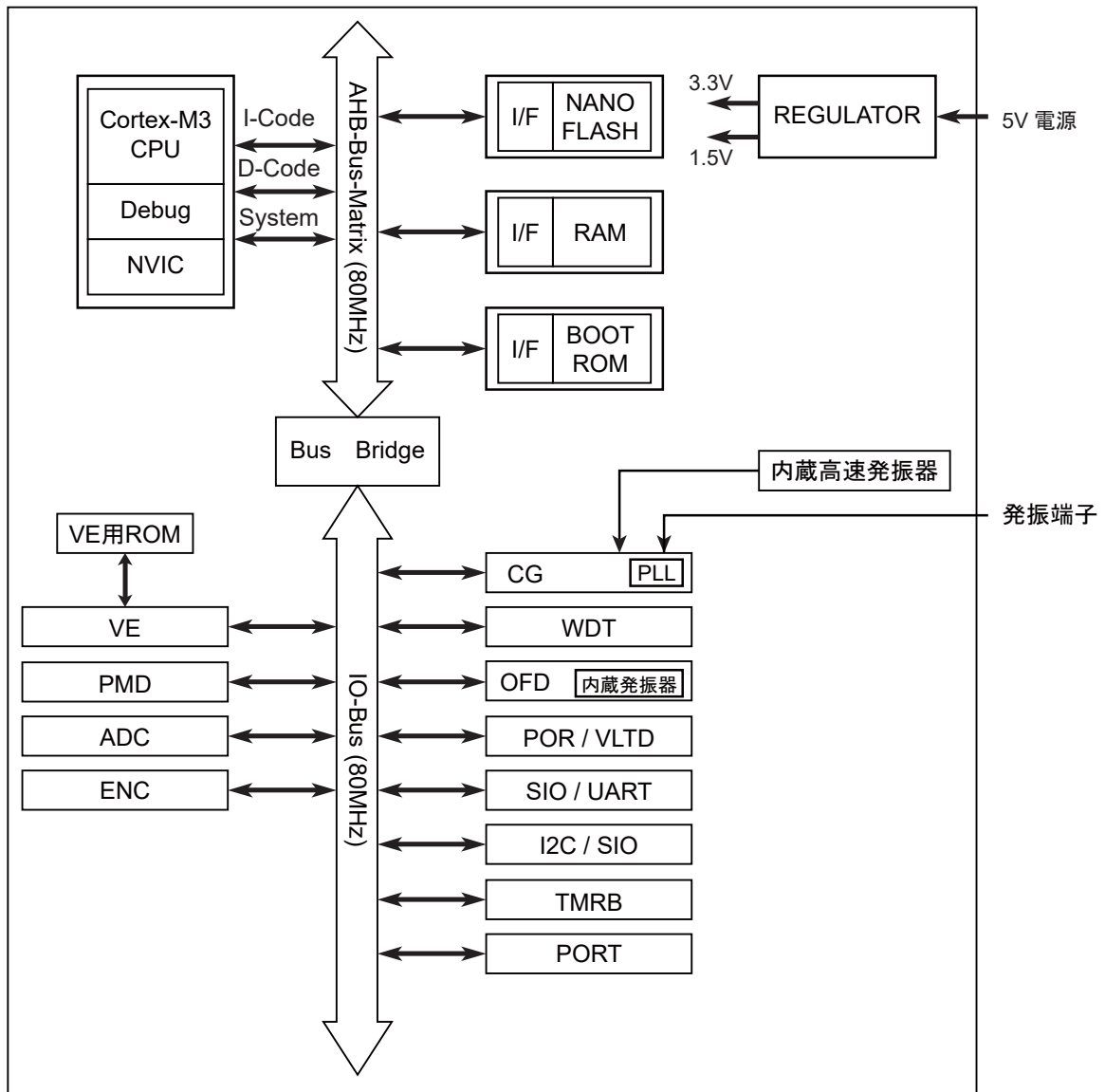


図 1-1 TMPM376FDDFG/FDFG ブロック図

### 1.3 ピン配置図 (Top view)

TMPM376FDDFG/FDFG のピン配置図は、下図のとおりです。

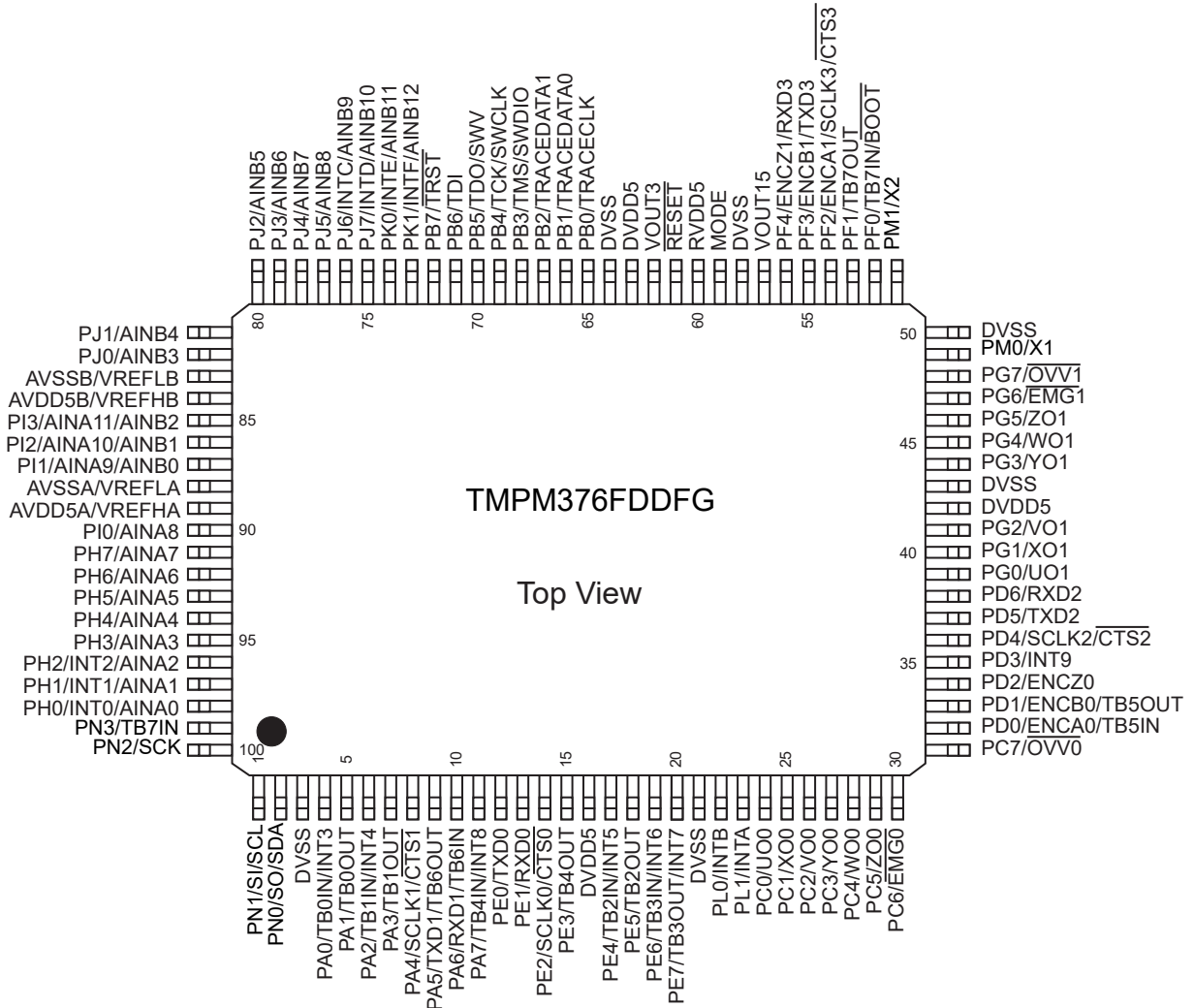


図 1-2 ピン配置図 (QFP100)

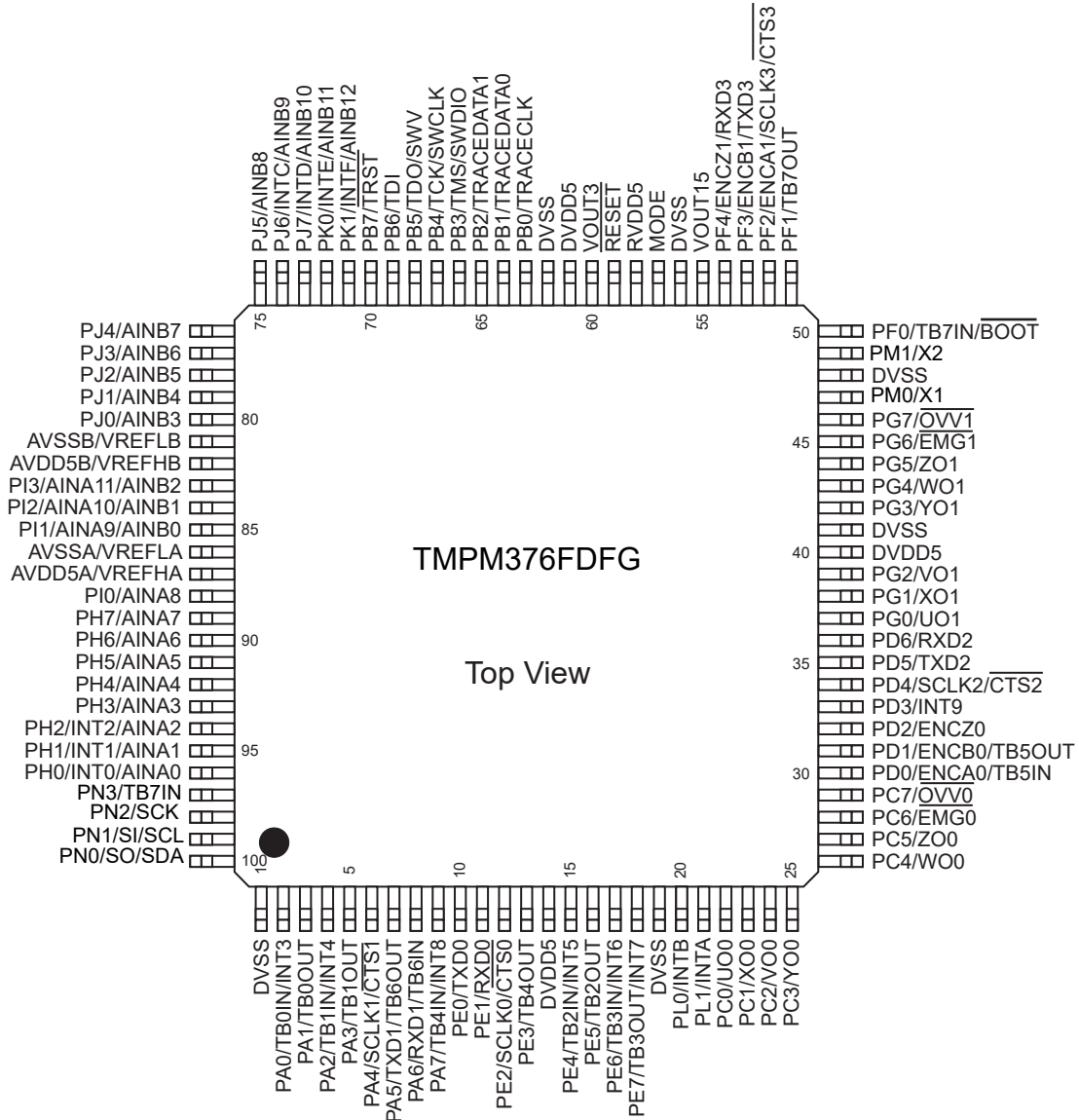


図 1-3 ピン配置図 (LQFP100)

## 1.4 ピン名称と機能

TMPM376FDDFG/FDFG の入出力ピン名称と機能は、表 1-1 のとおりです。

### 1.4.1 ポート順

表 1-1 ピン名称と機能 < ポート順 > ( 1 / 5 )

PORT	分類	ピン番号 (DFG/FG)	記号	入出力	機能
PORT A	機能	4 / 2	PA0 TB0IN INT3	入出力 入力 入力	入出力ポート タイマ B インพุットキャプチャ端子 外部割り込み端子
PORT A	機能	5 / 3	PA1 TB0OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT A	機能	6 / 4	PA2 TB1IN INT4	入出力 入力 入力	入出力ポート タイマ B インพุットキャプチャ端子 外部割り込み端子
PORT A	機能	7 / 5	PA3 TB1OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT A	機能	8 / 6	PA4 SCLK1 CTS1	入出力 入出力 入力	入出力ポート SIO クロック端子 SIO ハンドシェイク用端子
PORT A	機能	9 / 7	PA5 TXD1 TB6OUT	入出力 出力 出力	入出力ポート SIO 送信端子 タイマ B 出力端子
PORT A	機能	10 / 8	PA6 RXD1 TB6IN	入出力 入力 入力	入出力ポート SIO 受信端子 タイマ B インพุットキャプチャ端子
PORT A	機能	11 / 9	PA7 TB4IN INT8	入出力 入力 入力	入出力ポート タイマ B インพุットキャプチャ端子 外部割り込み端子
PORT B	機能 / デバッグ	65 / 63	PB0 TRACECLK	入出力 出力	入出力ポート デバッグ用端子
PORT B	機能 / デバッグ	66 / 64	PB1 TRACEDATA0	入出力 出力	入出力ポート デバッグ用端子
PORT B	機能 / デバッグ	67 / 65	PB2 TRACEDATA1	入出力 出力	入出力ポート デバッグ用端子
PORT B	機能 / デバッグ	68 / 66	PB3 TMS/SWDIO	入出力 入出力	入出力ポート デバッグ用端子
PORT B	機能 / デバッグ	69 / 67	PB4 TCK/SWCLK	入出力 入力	入出力ポート デバッグ用端子
PORT B	機能 / デバッグ	70 / 68	PB5 TDO/SWV	入出力 出力	入出力ポート デバッグ用端子
PORT B	機能 / デバッグ	71 / 69	PB6 TDI	入出力 入力	入出力ポート デバッグ用端子
PORT B	機能 / デバッグ	72 / 70	PB7 TRST	入出力 入力	入出力ポート デバッグ用端子
PORT C	機能	24 / 22	PC0 UO0	入出力 出力	入出力ポート U 相出力端子

表 1-1 ピン名称と機能 &lt; ポート順 &gt; ( 2 / 5 )

PORT	分類	ピン番号 (DFG/FG)	記号	入出力	機能
PORT C	機能	25 / 23	PC1 XO0	入出力 出力	入出力ポート X 相出力端子
PORT C	機能	26 / 24	PC2 VO0	入出力 出力	入出力ポート V 相出力端子
PORT C	機能	27 / 25	PC3 YO0	入出力 出力	入出力ポート Y 相出力端子
PORT C	機能	28 / 26	PC4 WO0	入出力 出力	入出力ポート W 相出力端子
PORT C	機能	29 / 27	PC5 ZO0	入出力 出力	入出力ポート Z 相出力端子
PORT C	機能	30 / 28	PC6 EMG0	入出力 入力	入出力ポート 異常検出入力
PORT C	機能	31 / 29	PC7 OVV0	入出力 入力	入出力ポート 過電圧入力
PORT D	機能	32 / 30	PD0 ENCA0 TB5IN	入出力 入力 入力	入出力ポート A 相入力端子 タイマ B インพุットキャプチャ端子
PORT D	機能	33 / 31	PD1 ENCB0 TB5OUT	入出力 入力 出力	入出力ポート B 相入力端子 タイマ B 出力端子
PORT D	機能	34 / 32	PD2 ENCZ0	入出力 入力	入出力ポート Z 相入力端子
PORT D	機能	35 / 33	PD3 INT9	入出力 入力	入出力ポート 外部割り込み端子
PORT D	機能	36 / 34	PD4 SCLK2 CTS2	入出力 入出力 入力	入出力ポート SIO クロック端子 SIO ハンドシェイク用端子
PORT D	機能	37 / 35	PD5 TXD2	入出力 出力	入出力ポート SIO 送信端子
PORT D	機能	38 / 36	PD6 RXD2	入出力 入力	入出力ポート SIO 受信端子
PORT E	機能	12 / 10	PE0 TXD0	入出力 出力	入出力ポート SIO 送信端子
PORT E	機能	13 / 11	PE1 RXD0	入出力 入力	入出力ポート SIO 受信端子
PORT E	機能	14 / 12	PE2 SCLK0 CTS0	入出力 入出力 入力	入出力ポート SIO クロック端子 SIO ハンドシェイク用端子
PORT E	機能	15 / 13	PE3 TB4OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT E	機能	17 / 15	PE4 TB2IN INT5	入出力 入力 入力	入出力ポート タイマ B インพุットキャプチャ端子 外部割り込み端子
PORT E	機能	18 / 16	PE5 TB2OUT	入出力 出力	入出力ポート タイマ B 出力端子

表 1-1 ピン名称と機能 &lt; ポート順 &gt; ( 3 / 5 )

PORT	分類	ピン番号 (DFG/FG)	記号	入出力	機能
PORT E	機能	19 / 17	PE6 TB3IN INT6	入出力 入力 入力	入出力ポート タイマ B インพุットキャプチャ端子 外部割り込み端子
PORTE	機能	20 / 18	PE7 TB3OUT INT7	入出力 出力 入力	入出力ポート タイマ B 出力端子 外部割り込み端子
PORT F	機能 / 制御	52 / 50	PF0 TB7IN <u>BOOT</u>	入出力 入力 入力	入出力ポート タイマ B インพุットキャプチャ端子 (注 3) BOOT モード端子 (注)RESET 信号の立ち上がりで "Low" をサンプリングしてシングルブートモードになります。
PORT F	機能	53 / 51	PF1 TB7OUT	入出力 出力	入出力ポート タイマ B 出力端子
PORT F	機能	54 / 52	PF2 ENCA1 SCLK3 <u>CTS3</u>	入出力 入力 入出力 入力	入出力ポート エンコーダ入力 SIO クロック端子 SIO ハンドシェイク用端子
PORT F	機能	55 / 53	PF3 ENCB1 TXD3	入出力 入力 出力	入出力ポート エンコーダ入力 SIO 送信端子
PORT F	機能	56 / 54	PF4 ENCZ1 RXD3	入出力 入力 入力	入出力ポート エンコーダ入力 SIO 受信端子
PORT G	機能	39 / 37	PG0 UO1	入出力 出力	入出力ポート U 相出力端子
PORT G	機能	40 / 38	PG1 XO1	入出力 出力	入出力ポート X 相出力端子
PORT G	機能	41 / 39	PG2 VO1	入出力 出力	入出力ポート V 相出力端子
PORT G	機能	44 / 42	PG3 YO1	入出力 出力	入出力ポート Y 相出力端子
PORT G	機能	45 / 43	PG4 WO1	入出力 出力	入出力ポート W 相出力端子
PORT G	機能	46 / 44	PG5 ZO1	入出力 出力	入出力ポート Z 相出力端子
PORT G	機能	47 / 45	PG6 <u>EMG1</u>	入出力 入力	入出力ポート 異常検出入力
PORT G	機能	48 / 46	PG7 <u>OVV1</u>	入出力 入力	入出力ポート 過電圧入力
PORT H	機能	98 / 96	PH0 INT0 AINA0	入出力 入力 入力	入出力ポート 外部割り込み端子 アナログ入力端子
PORT H	機能	97 / 95	PH1 INT1 AINA1	入出力 入力 入力	入出力ポート 外部割り込み端子 アナログ入力端子
PORT H	機能	96 / 94	PH2 INT2 AINA2	入出力 入力 入力	入出力ポート 外部割り込み端子 アナログ入力端子
PORT H	機能	95 / 93	PH3 AINA3	入出力 入力	入出力ポート アナログ入力端子

表 1-1 ピン名称と機能 &lt; ポート順 &gt; ( 4 / 5 )

PORT	分類	ピン番号 (DFG/FG)	記号	入出力	機能
PORT H	機能	94 / 92	PH4 AINA4	入出力 入力	入出力ポート アナログ入力端子
PORT H	機能	93 / 91	PH5 AINA5	入出力 入力	入出力ポート アナログ入力端子
PORT H	機能	92 / 90	PH6 AINA6	入出力 入力	入出力ポート アナログ入力端子
PORT H	機能	91 / 89	PH7 AINA7	入出力 入力	入出力ポート アナログ入力端子
PORT I	機能	90 / 88	PI0 AINA8	入出力 入力	入出力ポート アナログ入力端子
PORT I	機能	87 / 85	PI1 AINA9/AINB0	入出力 入力	入出力ポート アナログ入力端子
PORT I	機能	86 / 84	PI2 AINA10/AINB1	入出力 入力	入出力ポート アナログ入力端子
PORT I	機能	85 / 83	PI3 AINA11/AINB2	入出力 入力	入出力ポート アナログ入力端子
PORT J	機能	82 / 80	PJ0 AINB3	入出力 入力	入出力ポート アナログ入力端子
PORT J	機能	81 / 79	PJ1 AINB4	入出力 入力	入出力ポート アナログ入力端子
PORT J	機能	80 / 78	PJ2 AINB5	入出力 入力	入出力ポート アナログ入力端子
PORT J	機能	79 / 77	PJ3 AINB6	入出力 入力	入出力ポート アナログ入力端子
PORT J	機能	78 / 76	PJ4 AINB7	入出力 入力	入出力ポート アナログ入力端子
PORT J	機能	77 / 75	PJ5 AINB8	入出力 入力	入出力ポート アナログ入力端子
PORT J	機能	76 / 74	PJ6 INTC AINB9	入出力 入力 入力	入出力ポート 外部割り込み端子 アナログ入力端子
PORT J	機能	75 / 73	PJ7 INTD AINB10	入出力 入力 入力	入出力ポート 外部割り込み端子 アナログ入力端子
PORT K	機能	74 / 72	PK0 INTE AINB11	入出力 入力 入力	入出力ポート 外部割り込み端子 アナログ入力端子
PORT K	機能	73 / 71	PK1 INTF AINB12	入出力 入力 入力	入出力ポート 外部割り込み端子 アナログ入力端子
PORT L	機能	22 / 20	PL0 INTB	入力 入力	入力ポート 外部割り込み端子
PORT L	機能	23 / 21	PL1 INTA	入力 入力	入力ポート 外部割り込み端子
PORT M	機能 /クロック	49 / 47	PM0 X1	入出力 入力	入出力ポート 高速発振子接続端子



表 1-1 ピン名称と機能 &lt; ポート順 &gt; ( 5 / 5 )

PORT	分類	ピン番号 (DFG/FG)	記号	入出力	機能
PORT M	機能 / クロック	51 / 49	PM1 X2	入出力 出力	入出力ポート 高速発振子接続端子
PORT N	機能	2 / 100	PN0 SO / SDA	入出力 入出力	入出力ポート SIO モード送信端子、I2C モード送受信
PORT N	機能	1 / 99	PN1 SI / SCL	入出力 入出力	入出力ポート SIO モード受信端子、I2C モードクロック
PORT N	機能	100 / 98	PN2 SCK	入出力 入出力	入出力ポート SIO モードクロック端子
PORT N	機能	99 / 97	PN3 TB7IN	入出力 入力	入出力ポート タイマ B インพุットキャプチャ端子 (注 3)
-	制御	59 / 57	MODE	入力	モード端子 (注) 必ず GND に接続してください。
-	機能	61 / 59	$\overline{\text{RESET}}$	入力	リセット入力端子 (注) プルアップおよびノイズフィルタ (Typ. 条件で約 30ns) 付きです。
-	電源	3 / 1	DVSS	-	GND 端子
-	電源	21 / 19	DVSS	-	GND 端子
-	電源	43 / 41	DVSS	-	GND 端子
-	電源	50 / 48	DVSS	-	GND 端子
-	電源	58 / 56	DVSS	-	GND 端子
-	電源	64 / 62	DVSS	-	GND 端子
-	電源	16 / 14	DVDD5	-	電源端子
-	電源	42 / 40	DVDD5	-	電源端子
-	電源	63 / 61	DVDD5	-	電源端子
-	電源	60 / 58	RVDD5	-	電源端子
-	電源	57 / 55	VOUT15	-	電源端子
-	電源	62 / 60	VOUT3	-	電源端子
-	電源	83 / 81	AVSSB VREFLB	-	AD コンバータ用 GND 端子 (注 1) AD コンバータ用基準電源端子
-	電源	84 / 82	AVDD5B VREFHB	-	AD コンバータ用電源端子 (注 2) AD コンバータ用基準電源端子
-	電源	88 / 86	AVSSA VREFLA	-	AD コンバータ用 GND 端子 (注 1) AD コンバータ用基準電源端子
-	電源	89 / 87	AVDD5A VREFHA	-	AD コンバータ用電源端子 (注 2) AD コンバータ用基準電源端子

注 1) AD コンバータを使用しない場合でも GND 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電源に接続してください。

注 3) TB7IN は同時に使用することはできません。

## 1.5 電源の種類と供給端子

表 1-2 電源の種類と供給端子

電源の種類	電圧範囲	ピン番号 (DFG / FG)	電源供給端子
DVDD5	4.5 ~ 5.5V	16 / 14 , 42 / 40 , 63 / 61	PA,PB,PC,PD,PE,PF,PG,PL,PM PN,RESET,MODE
AVDD5A		89 / 87	PH, PI
AVDD5B		84 / 82	PJ, PK
RVDD5		60 / 58	-
VOUT15	1.35 ~ 1.65V	57 / 55	内部電源の安定化コンデンサ接続端子です。外部回路への電源供給はできません。DVSS との間にコンデンサ 3.3 $\mu$ F ~ 4.7 $\mu$ F を接続してください。
VOUT3	2.7 ~ 3.6V	62 / 60	内部電源の安定化コンデンサ接続端子です。外部回路への電源供給はできません。DVSS との間にコンデンサ 3.3 $\mu$ F ~ 4.7 $\mu$ F を接続してください。

注) VOUT15,VOUT3 は内蔵レギュレータ出力の安定化のためのコンデンサ接続端子です。

## 第2章 プロセッサコア

TX03 シリーズには、高性能 32 ビットプロセッサコア (ARM 社 Cortex-M3 コア) が内蔵されています。プロセッサコアの動作については、ARM 社からリリースされる "Cortex-M3 テクニカルリファレンスマニュアル" を参照してください。ここでは、製品固有の情報について説明します。

### 2.1 コアに関する情報

TMPM376FDDDFG/FDFG で使用している Cortex-M3 コアのリビジョンは以下のとおりです。

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

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

製品名	コアリビジョン
TMPM376FDDDFG/ FDFG	r2p0

### 2.2 構成可能なオプション

Cortex-M3 コアは、一部のブロックについて実装するかどうかを選択することができます。TMPM376FDDDFG/FDFG での構成は以下のとおりです。

構成可能なオプション	実装
FPB	リテラルコンパレータ: 2 本 命令コンパレータ: 6 本
DWT	コンパレータ: 4 本
ITM	あり
MPU	なし
ETM	あり
AHB-AP	あり
AHB トレースマクロセル インタフェース	あり
TPIU	あり
WIC	なし

## 2.3 例外 / 割り込み

例外 / 割り込みに関連する製品固有の情報をまとめます。

### 2.3.1 割り込み本数

Cortex-M3 コアは割り込み本数を 1 ~ 240 本の間で任意に構成することができます。

TMPM376FDDFG/FDFG の割り込み本数は 79 本です。割り込み本数は NVIC レジスタの割り込みコントローラタイプレジスタの <INTLINESNUM[4:0]> ビットに反映され、本製品では "0x00" が読み出されます。

### 2.3.2 割り込み優先度ビット数

Cortex-M3 コアは割り込み優先度ビット数を 3 ~ 8 ビットの間で任意に構成することができます。

TMPM376FDDFG/FDFG の割り込み優先度は 3 ビットです。このビット数は割り込み優先度レジスタとシステムハンドラ優先度レジスタのビット構成に反映されます。

### 2.3.3 SysTick

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

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。

### 2.3.4 SYSRESETREQ

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

TMPM376FDDFG/FDFG 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。

### 2.3.5 LOCKUP

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

TMPM376FDDFG/FDFG ではこの信号は未使用です。LOCKUP 状態から復帰する際にはマスク不能割り込み (NMI) またはリセットを使用する必要があります。

### 2.3.6 補助フォールトステータスレジスタ

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

TMPM376FDDFG/FDFG ではこのレジスタに対して機能を定義していません。リードすると常に "0x0000\_0000" が読み出されます。

## 2.4 イベント

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

TMPM376FDDDFG/FDFG では、イベント出力信号、イベント入力信号とも未使用です。SEV 命令、WFE 命令は使用しないでください。

## 2.5 電力管理

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

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

TMPM376FDDDFG/FDFG では、SLEEPDEEP 信号は使用していません。<SLEEPDEEP> ビットはセットしないでください。また、イベント信号も未使用のため、WFE 命令は使用しないでください。

電力管理については、「クロック/モード制御」の章を参照してください。

## 2.6 排他アクセス

Cortex-M3 コアは DCode バスおよびシステムバスの排他アクセスをサポートしていますが、TMPM376FDDDFG/FDFG ではこの機能を使用していません。



## 第3章 メモリマップ

### 3.1 メモリマップ

TMPM376FDDDFG/FDFG のメモリマップは、ARM Cortex-M3 コアのメモリマップに沿って作られており、内蔵 ROM 領域は Cortex-M3 コアメモリマップの Code 領域、内蔵 RAM 領域は SRAM 領域、特殊機能レジスタ (SFR) 領域は Peripheral 領域に割り付けられています。特殊機能レジスタ (SFR: Special function register) とは、入出力ポートおよび周辺機能のコントロールレジスタを示します。SRAM 領域、SFR 領域はすべてビットバンド領域に含まれています。

CPU 内レジスタ領域はコア内部のレジスタ領域です。

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

"Fault" と記載された領域では、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。

#### 3.1.1 TMPM376FDDDFG/FDFG メモリマップ

TMPM376FDDDFG/FDFG のメモリマップを以下に示します。

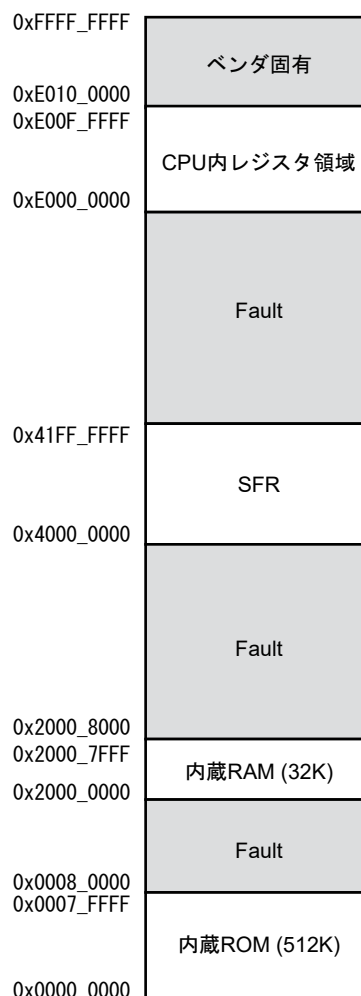


図 3-1 メモリマップ

## 3.2 SFR 領域詳細

SFR 領域の詳細を表 3-1 に示します。

表 3-1 の Reserved ( 予約領域 ) にはアクセスしないでください。また、周辺機能によってはアクセスを制限するアドレスを有する場合があります。詳細は各周辺機能の章を参照してください。

表 3-1 SFR 詳細

Start Address	End Address	Peripheral
0x4000_0000	0x4000_037F	PORT
0x4000_0380	0x4000_FFFF	Reserved
0x4001_0000	0x4001_01FF	TMRB
0x4001_0200	0x4001_03FF	Reserved
0x4001_0400	0x4001_053F	ENC
0x4001_0540	0x4001_FFFF	Reserved
0x4002_0000	0x4002_007F	I2C/SIO
0x4002_0080	0x4002_017F	SIO/UART
0x4002_0180	0x4002_FFFF	Reserved
0x4003_0000	0x4003_02FF	ADC
0x4003_0300	0x4003_FFFF	Reserved
0x4004_0000	0x4004_003F	WDT
0x4004_0040	0x4004_01FF	Reserved
0x4004_0200	0x4004_023F	CG
0x4004_0240	0x4004_07FF	Reserved
0x4004_0800	0x4004_083F	OFD
0x4004_0840	0x4004_08FF	Reserved
0x4004_0900	0x4004_093F	VLTD
0x4004_0940	0x4004_FFFF	Reserved
0x4005_0000	0x4005_023F	VE
0x4005_0240	0x4005_03FF	Reserved
0x4005_0400	0x4005_04FF	PMD
0x4005_0500	0x4007_FFFF	Reserved
0x4008_0000	0x41FF_EFFF	Hard fault
0x41FF_F000	0x41FF_F03F	FLASH
0x41FF_F040	0x41FF_FFFF	Reserved



## 第 4 章 リセット動作

### 4.1 リセット入力前状態

電源投入時は、製品の状態は不定です。全ての電源に電圧が印加され、リセット端子にローレベルが入力されるまでの期間、内部回路は不確定であり、レジスタの設定や各端子の状態は不定となります。

### 4.2 リセット期間

本デバイスはパワーオンリセットを内蔵しており、電源投入時にパワーオンリセット信号を発生します。

また、外部からリセットをかける場合は、電源電圧が動作範囲内であり、かつ、高周波発振器の発振が安定した状態で、RESET 端子入力を少なくとも 12 システムクロック間 (1.2 $\mu$ s@10MHz 発振接続時) ローレベル "L" にしてください。

### 4.3 リセット解除後

リセット解除後は、Cortex-M3 コアの制御レジスタや内蔵 I/O レジスタは初期化されます。なお、リセット解除後は PLL 逡倍回路が停止しているため、PLL 逡倍回路を使用する場合は CGOSCCR および CGPLLSEL にて設定が必要です。

リセット例外処理をおこなった後、プログラムはリセットの割り込みサービスルーチンへ分岐します。リセットの割り込みサービスルーチン先頭アドレスは、0x0000\_0004 番地に格納します。

注 1) RESET 端子を "L" にした状態で電源投入することも可能です。

注 2) リセット動作により、内蔵 RAM のデータは保証できなくなります。



## 第5章 クロック / モード制御

### 5.1 特長

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

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

本章では、クロックの制御および動作モードとモード遷移について説明します。

クロックに関連する機能としては以下のようなものがあります。

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

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

## 5.2 レジスタ説明

### 5.2.1 レジスタ一覧

CG 関連のレジスタとアドレスを以下に示します。

Base Address = 0x4004 \_ 0200

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

5.2.2 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 選択 0: fgear 1: fc fperiph のソースクロックを選択します。
11	-	R	リードすると "0" が読めます。
10-8	PRCK[2:0]	R/W	プリスケラック選択 000: fperiph 001: fperiph/2 010: fperiph/4 011: fperiph/8 100: fperiph/16 101: fperiph/32 110: Reserved 111: Reserved 周辺 I/O に供給するプリスケラックを選択します。
7-3	-	R	リードすると "0" が読めます。
2-0	GEAR[2:0]	R/W	高速クロック (fc) のギア選択 000: fc 001: Reserved 010: Reserved 011: Reserved 100: fc/2 101: fc/4 110: fc/8 111: fc/16

5.2.3 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	HOSCON	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	発振器用ウォーミングアップタイマの時間を設定
19	WUPSEL2	R/W	ウォーミングアップタイマ (WUP) のクロック 0: 内部 (OSC2) 1: 外部 (OSC1) ウォーミングアップタイマのクロックソースを選択します。
18	HOSCON	R/W	ポート M/ 外部高速発振器選択 (注) 0: ポート M 1: 発振器 (X1 / X2) 外部高速発振器 (X1 / X2) とポート M のどちらを使用するか選択します。 外部発振器として選択する場合は、ポート M について、PMCR/PMPUP/PMPDN/PMIE=disable に設定してください (リセット後は、全て disable)。
17	OSCSEL	R/W	高速発振器の切り替え 0: 内部 (OSC2) 1: 外部 (OSC1) 高速発振を外部 (OSC1) へ切り替えます。切り替え後は高速発振器が外部 (OSC1) へ切り替わった事 (<OSCSEL> = "1") を確認し、内蔵発振器は速やかに停止してください。また、外部発振器に切り替え後は本ビットの書き替えによる内蔵発振器に切り替えを行わないでください。
16	XEN2	R/W	高速発振器 2 (内部 : OSC2) 0: 停止 1: 発振 高速発振器 2(OSC2) の動作を選択します。
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	高速発振器 1 (外部 : OSC1) 0: 停止 1: 発振 高速発振器 1(OSC1) の動作を選択します。
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" を設定してください。
2	PLLON	R/W	PLL 動作 0: 停止 1: 発振 PLL (通倍回路) の動作を選択します。 リセット解除後は停止状態です。PLL を使用する場合はこのビットをセットします。
1	WUEF	R	ウォーミングアップタイマ (WUP) ステータス 0: ウォーミングアップ終了 1: ウォーミングアップ中 このビットを読むことでウォーミングアップタイマの動作状態を確認できます。
0	WUEON	W	ウォーミングアップタイマ (WUP) 制御 0: don't care 1: ウォーミングアップスタート このビットをセットすることでウォーミングアップタイマがスタートします。  (注) 自動ウォーミングアップする低消費電力モード (表 5-6) からの復帰では、本レジスタの設定をする必要はありません。ソフトにて WUP スタートした場合にはウォーミングアップが終了していること (<WUEF> = "0") を確認してから、モード遷移するようにしてください。

注) <HOSCON>="1" 設定時、ポート M のすべてのレジスタにアクセスできなくなります。(読み出すと常に "0" が読めます) また、PMDATA, PMOD を除くポート M のレジスタのいずれかが "0" でない場合、<HOSCON> を "1" に設定することはできません。

5.2.4 CGSTBYCR(スタンバイコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	DRVE
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	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	DRVE	R/W	STOPモード中の端子状態制御 0: STOPモード中端子をドライブしません 1: STOPモード中も端子をドライブします
15-10	-	R	リードすると"0"が読めます。
9	-	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蔵発振器(OSC2)または外部発振器(OSC1)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。



## 5.2.5 CGPLLSEL(PLL セレクトレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	1	0	1	0	0	0	0	1
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	PLLSEL
リセット後	0	0	1	1	1	1	1	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	"1010" をライトしてください。
11	-	R	リードすると "0" が読めます。
10-1	-	R/W	"00_1001_1111" をライトしてください。
0	PLLSEL	R/W	PLL 選択 0: fosc 1: PLL 使用  PLL で逡倍されたクロック ( $f_{PLL}$ ) をシステムクロック : fc へのクロックソースとするか選択します。 リセット解除後は "fosc" が選択されているので、PLL を使用する場合は設定が必要です

## 5.3 クロック制御

### 5.3.1 クロックの種類

クロックの一覧を以下に示します。

fosc1	: X1, X2 端子より入力されるクロック
fosc2	: 内蔵発振器より入力されるクロック
fosc	: fosc1 または fosc2 どちらか選択されたシステムクロック
f <sub>PLL</sub>	: PLL により逡倍 (8 逡倍) されたクロック
fc	: CGPLLSEL<PLLSEL> で選択されたクロック (高速クロック)
fgear	: CGSYSCR<GEAR[2:0]> で選択されたクロック
f <sub>sys</sub>	: fgear と同一クロック (システムクロック)
f <sub>periph</sub>	: CGSYSCR<FPSEL> で選択されたクロック
φT0	: CGSYSCR<PRCK[2:0]> で選択されたクロック (プリスケラクロック)

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

高速クロック	: fc, fc/2, fc/4, fc/8, fc/16
プリスケラクロック	: f <sub>periph</sub> , f <sub>periph</sub> /2, f <sub>periph</sub> /4, f <sub>periph</sub> /8, f <sub>periph</sub> /16, f <sub>periph</sub> /32

### 5.3.2 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 (OSC1)	: 停止 (X1,X2)
内部高速発振器 (OSC2)	: 発振
PLL (逡倍回路)	: 停止
高速クロックギア	: fc (分周なし)

リセット動作により、すべてのクロックの設定が f<sub>OSC2</sub> と同じになります。

f <sub>c</sub>	= f <sub>OSC2</sub>
f <sub>sys</sub>	= f <sub>c</sub> (= f <sub>OSC2</sub> )
f <sub>periph</sub>	= f <sub>c</sub> (= f <sub>OSC2</sub> )
φT0	= f <sub>periph</sub> (= f <sub>OSC2</sub> )

### 5.3.3 クロック系統図

クロック系統図を図 5-1 に示します。

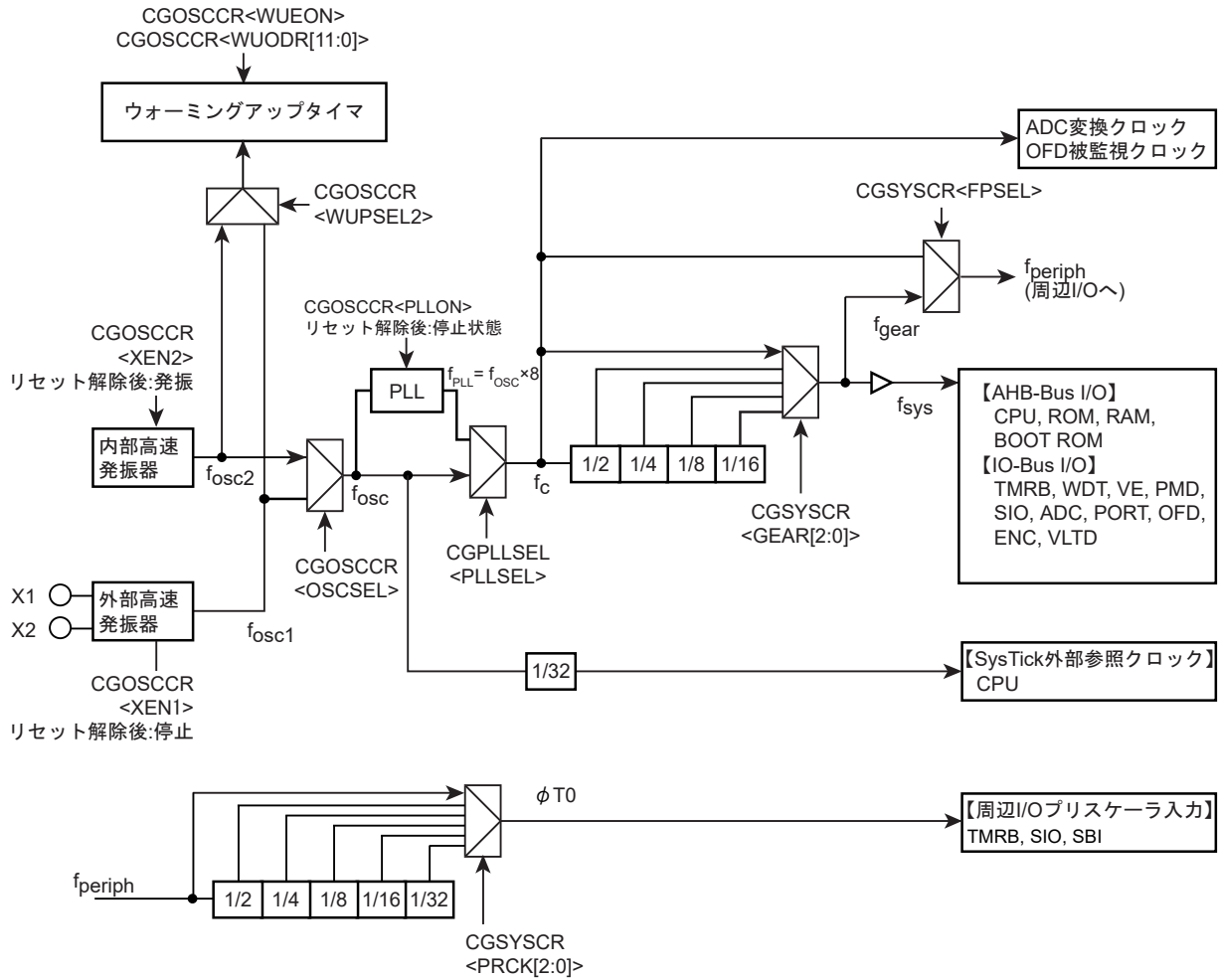


図 5-1 クロック系統図

セレクタに入力されるクロックのうち、矢印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。

### 5.3.4 クロック逡倍回路 (PLL)

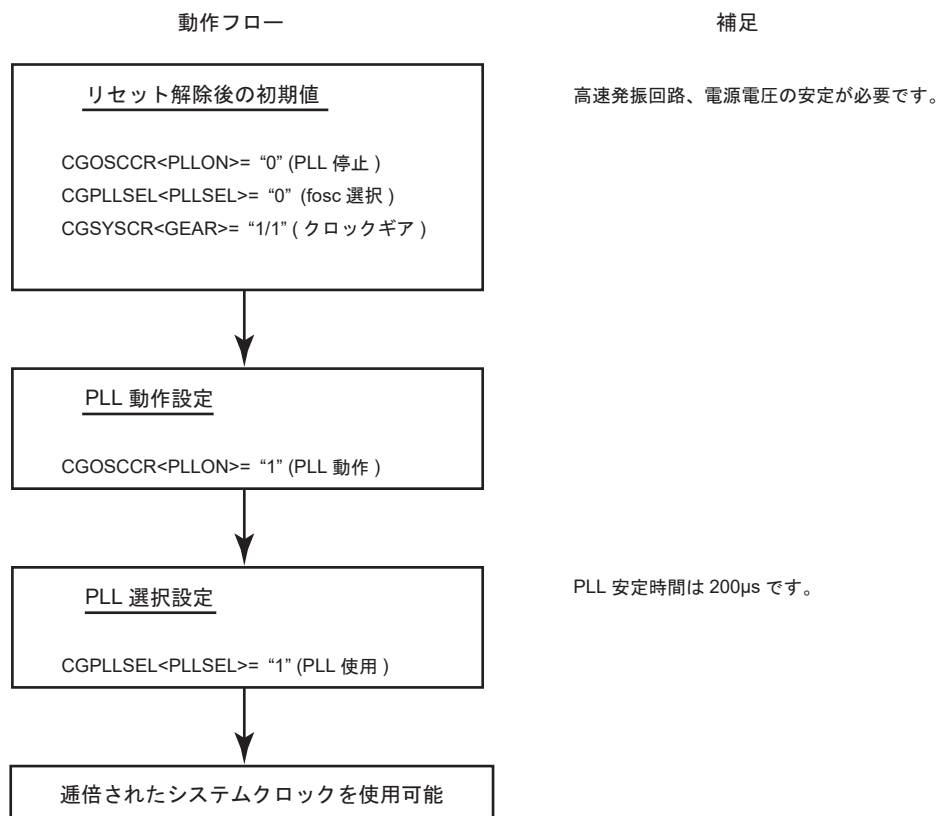
高速発振器の出力クロック  $f_{osc}$  を 8 逡倍した  $f_{PLL}$  クロックを出力する回路です。これにより、発振器への入力周波数は低く内部クロックは高速にすることが可能です。

PLL はリセット解除後、ディセーブル状態です。PLL を有効にするためには、CGOSCCR<PLLON> を "1" に設定し、CGPLLSEL<PLLSEL> を "1" に設定することで  $f_{osc}$  を 8 逡倍した  $f_{PLL}$  クロック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。

注) PLL 動作開始の PLL 安定時間は、約 200  $\mu$ s です。

#### 5.3.4.1 PLL 設定シーケンス

以下にリセット解除後の PLL 設定シーケンスを示します。



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

### 5.3.5 ウォーミングアップ機能

ウォーミングアップ機能は、STOP モード解除時に、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。詳細機能については「5.6.6 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。

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

ウォーミングアップ機能の設定方法を説明します。

#### 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。

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

$$\frac{\text{ウォーミングアップ時間}}{\text{ウォームアップクロック周期}} = \frac{5\text{ms}}{1/8\text{MHz}} = 40,000\text{サイクル} = 0x9C40$$

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

#### 3. ウォーミングアップの開始および終了確認

ウォーミングアップタイマのスタートおよび終了確認は、CGOSCCR<WUEON><WUEF> を使用してソフトウェア (命令) により行います。

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 (OSC1))
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 5.3.6 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

### 5.3.6 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 : 8 MHz ~ 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 5-1 高速クロック範囲 (単位は MHz)

入力周波数	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	
OSC1	8	1	80	8	64	32	16	8	4	8	4	2	1	-
	10			10	80	40	20	10	5	10	5	2.5	1.25	-
OSC2	10			10	80	40	20	10	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。

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. PMCR / PMPUP / PMPDN / PMIE の該当ビット = "0" : PM0, PM1 について禁止 (リセット後は、全て禁止)
2. 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
3. CGOSCCR<HOSCON> = "1" : 外部発振使用 : PM0, PM1 X1, X2
4. CGOSCCR<XEN1> = "1" : 高速発振器 (OSC) の発振
5. CGOSCCR<WUPSEL2> = "1" : ウォーミングアップタイマクロック設定 (1: 外部 (OSC1))
6. 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) になるまでウェイト
7. CGOSCCR<OSCSEL> = "1" : 高速発振器を外部発振器 (OSC1) に切り替え
8. CGOSCCR<OSCSEL> リード : 外部発振選択を確認 (<OSCSEL> = "1")
9. CGOSCCR<XEN2> = "0" : 内部発振停止

なお、CGOSCCR<HOSCON> = "1" と設定したあとはポート M の全てのレジスタのアクセスが禁止されるため、ポート M の設定値を変更する事はできません。

### 5.3.7 プリスケーラクロック

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

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



## 5.4 モードとモード遷移

### 5.4.1 モード状態遷移

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

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

図 5-2 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、"Cortex-M3 テクニカルリファレンスマニュアル" を参照してください。

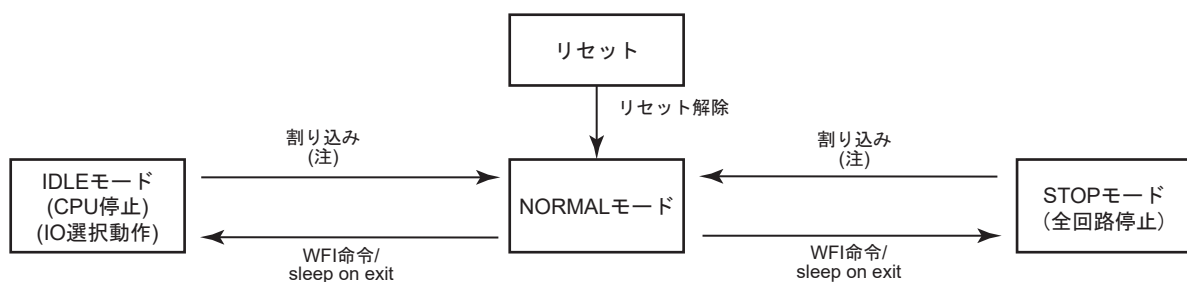


図 5-2 モード状態遷移図

注) 復帰時にウォーミングアップが必要となります。ウォーミングアップ時間の設定は 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、「5.6.6 ウォーミングアップ」を参照してください。

## 5.5 動作モード

動作モードには NORMAL モードがあります。NORMAL モードの特長は次のとおりです。

### 5.5.1 NORMAL モード

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

## 5.6 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-M3 コアの SLEEPDEEP による低消費電力モードはサポートしていません。システム制御レジスタの <SLEEPDEEP> ビットは設定しないでください。

IDLE, STOP モードの特長は次のとおりです。

### 5.6.1 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/SIO)
- ウォッチドッグタイマ (WDT)
- ベクトルエンジン (VE)

注) ウォッチドッグタイマは IDLE モードへ遷移する前に停止してください。

5.6.2 STOP モード

STOP モードでは、内部発振器も含めてすべての内部回路が停止します。

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

STOP モードでは、CGSTBYCR<DRVE> の設定により端子の状態を設定することができます。STOP モード時の端子状態を表 5-2 に示します。

表 5-2 STOP モード時の端子状態

	機能	入出力	<DRVE> = 0	<DRVE> = 1
ポート 以外	RESET, MODE	入力専用	○	
ポート	X1	入力専用	×	
	X2	出力専用	"High" レベル出力	
	TMS TCK TDI TRST	入力	○	
	TDO	出力	データ有効時はイネーブル データ無効時はディセーブル	
	SWCLK	入力	○	
	SWDIO	入力	○	
		出力	データ有効時はイネーブル データ無効時はディセーブル	
	TRACECLK TRACEDATA0 TRACEDATA1 SWV	出力	○	
	U00,1 VO0,1 WO0,1 XO0,1 YO0,1 ZO0,1	出力	×	○
	INT0, INT1, INT2 INT3, INT4, INT5 INT6, INT7, INT8 INT9, INTA, INTB INTC, INTD, INTE INTF	入力	○	
上記以外の機能端子または汎用入出力として使用	入力	×	○	
	出力	×	○	

○：入力または出力がイネーブルであることを示します。

×：入力または出力がディセーブルであることを示します。

### 5.6.3 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 5-3 に <STBY[2:0]> の設定より選択されるモードを示します。

表 5-3 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。

## 5.6.4 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 5-4 に示します。

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

表 5-4 各動作モードにおける動作状態

ブロック	NORMAL	IDLE	STOP
プロセッサコア	○	×	×
I/O ポート	○	○	* (注 1)
PMD	○	○	×
ENC	○	○	×
OFD	○	○	×
ADC	○	○	×
VE	○		×
SIO	○	モジュールごとに動作 / 停止 選択可能	×
SBI	○		×
TMRB	○		×
WDT	○		×
VLTD	○	○	○ (注 2)
POR	○	○	○ (注 2)
CG	○	○	×
PLL	○	○	×
高速発振器 (fc)	○	○	×

○: 対象のモード中に動作が可能

×: 対象のモードに移行すると自動的にモジュールへのクロックが停止

注 1) CGSTBYCR<DRVE> の設定によります。

注 2) クロックは供給されていませんが、動作停止しません。

### 5.6.5 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。使用できるスタンバイ解除ソースは、低消費電力モードにより決まります。詳細を表 5-5 に示します。

表 5-5 解除ソースと解除可能なモード

低消費電力モード		IDLE (プログラマブル)	STOP
解除 ソース	割り 込み	INT0~F (注 1)	○
		INTRX0~3, INTTX0~3	○
		INTVCNA, INTVCNB	○
		INTEMG0~1	○
		INTOVV0~1	○
		INTADAPDA, INTADBPDA, INTADAPDB, INTADBPDB	○
		INTTB00, 10, 20, 30, 40, 50, 60, 70 INTTB01, 11, 21, 31, 41, 51, 61, 71	○
		INTPMD0, 1	○
		INTCAP00, 10, 20, 30, 40, 50, 60, 70 INTCAP01, 11, 21, 31, 41, 51, 61, 71	○
		INTADACPA, INTADBCPA, INTADACPB, INTADBCPB	○
		INTADASFT, INTADBSFT	○
		INTADATMR, INTADBTMR	○
		INTENC0, INTENC1	○
		INTSBI	○
		SysTick 割り込み	○
マスク不能割り込み (INTWDT)	○		
RESET (RESET 端子)	○		

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

×: 解除に使用できません。

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

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

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「5.6.6 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す。

#### • マスク不能割り込み (NMI) による解除

マスク不能割り込みの要因には、WDT 割り込み (INTWDT) があります。INTWDT は IDLE モードでのみ使用可能です。

- リセットによる解除

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

- 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。

### 5.6.6 ウォーミングアップ

モード遷移の際には内部発振器 (OSC2) 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、CGOSCCR<WUODR> でウォーミングアップ時間の設定を行ってください。

注 1) TMPM376FDDDFG/FDFG 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、PLL の安定時間も考慮してウォーミングアップ時間を設定してください。PLL の安定時間は、約 200  $\mu$ s です。

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 5-6 に示します。

表 5-6 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



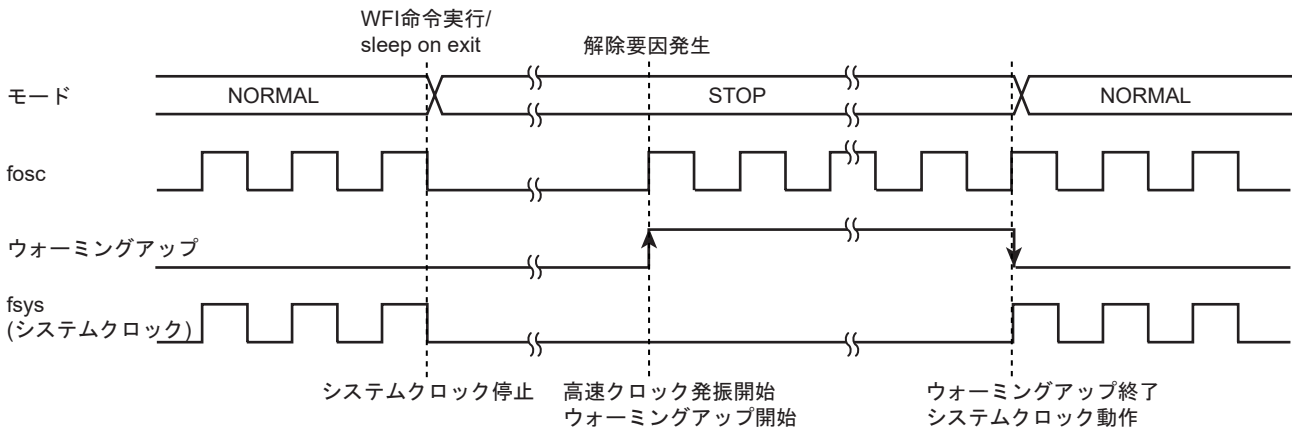
### 5.6.7 モード遷移によるクロック動作

モード遷移の際の、クロック動作について示します。

#### 5.6.7.1 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ので、発振動作が安定するまでリセット信号を有効に保ってください。





## 第6章 例外

この章では、例外の特長、種類、処理について概略を説明します。

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

### 6.1 概要

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

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

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



#### 6.1.1 種類

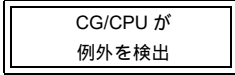
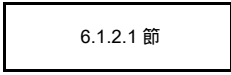

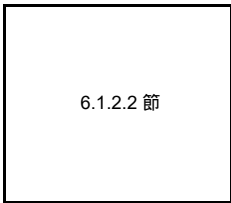
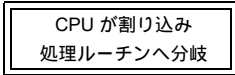
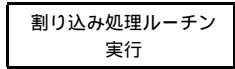
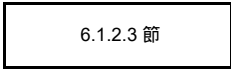
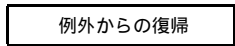
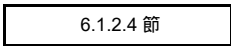
例外には以下のようなものがあります。

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

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

### 6.1.2 処理の流れ

例外 / 割り込みの処理の流れの概略を以下に示します。以下の説明で、 はハードウェアによる処理を、 はソフトウェアによる処理を示しています。それぞれの処理の内容について、後続の節で説明します。

処理	内容	説明
 CG/CPU が例外を検出	CG/CPU が例外要求を検出します。	 6.1.2.1 節
↓		
 CPU が例外を処理	CPU が例外処理を行います。	 6.1.2.2 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。	 6.1.2.3 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。	 6.1.2.4 節

## 6.1.2.1 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、例外が発生する命令の実行や、命令実行中の異常が要因で発生します。

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

割り込みの要求は、外部割り込み端子からの信号入力や周辺機能から発生します。スタンバイ解除に使用する割り込みの場合、クロックジェネレータの設定も必要になります。詳細は「6.5 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 6-1 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、「6.5.1.5 要因一覧」を参照してください。

(3) 優先度の設定

・優先度レベル

外部割り込みの優先度は、割り込み優先度レジスタで、それ以外の例外はシステムハンドラ優先度レジスタで該当する <PRI\_n> ビットに設定します。

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

8 ビットの構成の場合、優先度は 0 ~ 255 のレベルを設定できます。最も高い優先度は "0" です。複数の要因を同じ優先度に設定した場合、番号の小さい例外の優先度が高くなります。

注) 本製品では、<PRI\_n> ビットは 3 ビットの構成になっています。

・優先度のグループ化

優先度をグループ化することもできます。アプリケーション割り込みおよびリセット制御レジスタの <PRIGROUP> を設定することで、<PRI\_n> を横取り優先度とサブ優先度に分割することができます。

優先度はまず横取り優先度で判定され、横取り優先度が同じ場合サブ優先度で判定されます。サブ優先度も同じ場合は例外番号の小さいほうが優先度が高くなります。

表 6-2 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 6-2 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。

### 6.1.2.2 例外の処理と割り込み処理ルーチンへの分岐 (横取り)

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

#### (1) レジスタの退避

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

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

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



#### (2) 割り込み処理ルーチンのフェッチ

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

各例外の割り込み処理ルーチンの先頭番地をベクタテーブルに準備しておきます。ベクタテーブルはリセット後、コード領域の 0x0000\_0000 番地に置かれます。ベクタテーブルは、ベクタテーブルオフセットレジスタを設定することでコード空間または SRAM 空間の任意のアドレスに置くことができます。

ベクタテーブルにはまた、メインスタックの初期値を設定します。

#### (3) 後着

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

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

#### (4) ベクタテーブルの構成

ベクタテーブルの構成は以下のとおりです。

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

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

#### 6.1.2.3 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「6.5 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。



#### 6.1.2.4 例外からの復帰

##### (1) 割り込み処理ルーチンからの復帰先

割り込み処理ルーチン終了時の状態により復帰先が決まります。

- テールチェイン

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

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

- 処理が中断されている割り込み処理ルーチンへ復帰

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

- 元のプログラムへ復帰

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

##### (2) 復帰処理

CPU は、割り込み処理ルーチンから復帰する際に以下の処理を行います。

- レジスタの復帰

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

- 割り込み番号のロード

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

- SP の選択

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

## 6.2 リセット例外

リセット例外には、以下の6種類の要因があります。

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

- 外部リセット端子

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

- POR によるリセット例外

POR にリセットが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 にリセットが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 にリセットが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 にリセットが発生する機能があります。詳細は「ウォッチドッグタイマ」の章をご覧ください。

- SYSRESETREQ によるリセット例外

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

## 6.3 マスク不能割り込み (NMI)

WDT にマスク不能割り込みが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。

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

## 6.4 SysTick

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

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

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

注) 本製品では外部参照クロックとして fosc( CGOSCCR<OSCESEL> 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。

## 6.5 割り込み

この節では、割り込みの伝わる経路、要因、必要な設定について説明します。

割り込みは、割り込み要因ごとの信号により CPU へ通知されます。

CPU は、優先順位付けを行い最も優先度の高い割り込みを発生します。

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

### 6.5.1 要因

#### 6.5.1.1 経路

割り込み要求の経路を図 6-1 に示します。

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

スタンバイ解除に使用される周辺機能割り込み(経路 ②)および、外部割り込み端子からの割り込み(経路 ③)はクロックジェネレータに入力され、スタンバイ解除のロジックを経由して CPU に入力されます。(経路 ④)

外部割り込み端子からの割り込みは、スタンバイ解除に使用しない場合スタンバイ解除ロジックを経由せずに CPU に入力されます。(経路 ⑤)

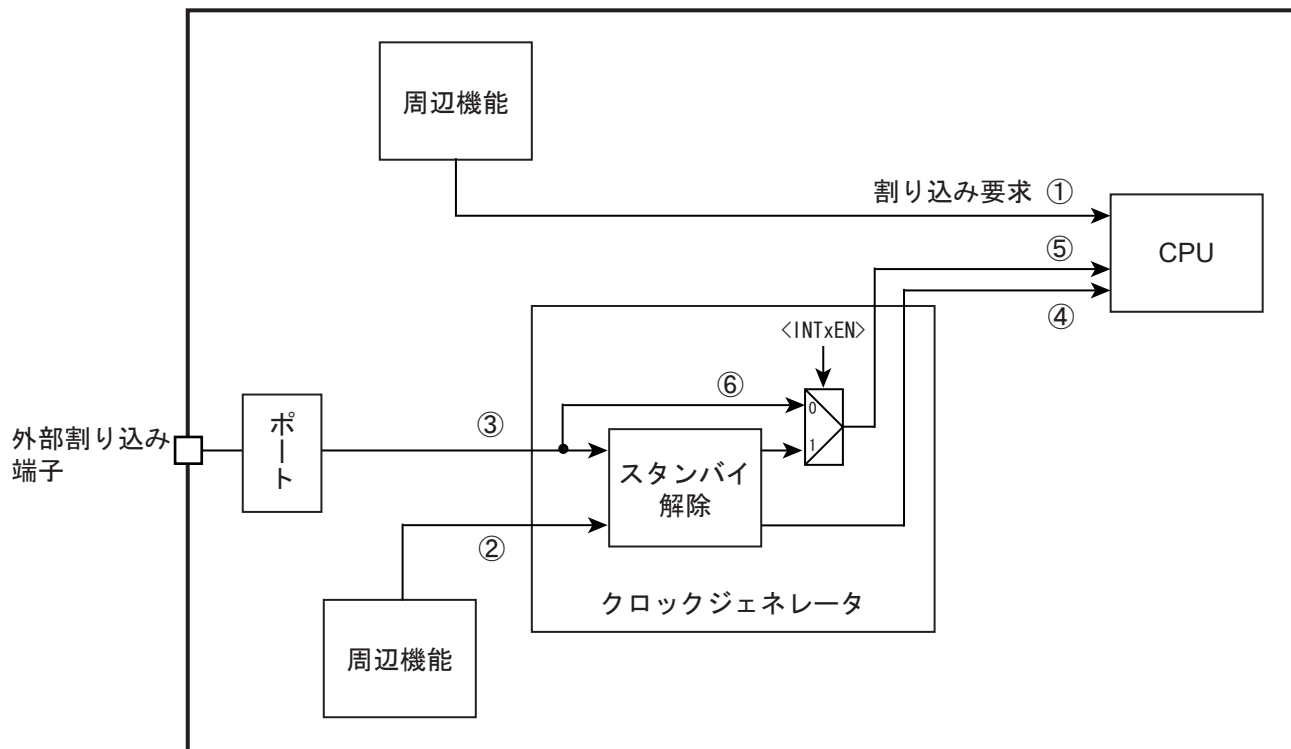


図 6-1 割り込みの経路

### 6.5.1.2 割り込み要因の発生

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

- 外部端子からの割り込み  
外部割り込み端子を使用する場合、ポートの制御レジスタで端子を割り込み機能に設定します。
- 周辺機能の割り込み  
周辺機能の割り込みを使用する場合、使用する周辺機能で割り込みが出力されるよう設定する必要があります。  
詳細は各章をご覧ください。
- 割り込みの強制的な保留  
割り込み保留セットレジスタの該当する割り込みのビットをセットすることで、割り込み要因を発生させることができます。

### 6.5.1.3 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。

### 6.5.1.4 外部割り込み端子を使用する際の注意

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

外部割り込み端子からの入力信号は、入力ディセーブル (PxIE<PxmIE>="0") の場合 "High" となります。また、外部割り込みをスタンバイ解除要因として使用しない場合 (「図 6-1 割り込みの経路」の 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。

6.5.1.5 要因一覧

割り込みの要因一覧を表 6-3 に示します。

表 6-3 割り込み要因一覧

番号	要因		アクティブレベル (スタンバイ解除)	CG 割り込みモード コントロールレジスタ
0	INT0	割り込み端子	[H]/[L] レベル [↑]/[↓] エッジ 選択可	CGIMCGA
1	INT1	割り込み端子		
2	INT2	割り込み端子		
3	INT3	割り込み端子		
4	INT4	割り込み端子	[H]/[L] レベル [↑]/[↓] エッジ選択可	CGIMCGB
5	INT5	割り込み端子		
6	INTRX0	シリアル受信 (channel0)		
7	INTTX0	シリアル送信 (channel0)		
8	INTRX1	シリアル受信 (channel1)		
9	INTTX1	シリアル送信 (channel1)		
10	INTVCNA	ベクトルエンジン割り込み A		
11	INTVCNB	ベクトルエンジン割り込み B		
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 PMD0 トリガ同期変換終了		
17	INTADBPDA	ADCB PMD0 トリガ同期変換終了		
18	INTADAPDB	ADCA PMD1 トリガ同期変換終了		
19	INTADBPDB	ADCB PMD1 トリガ同期変換終了		
20	INTTB00	16bitTMRB0 コンペア一致 0/ オーバーフロー		
21	INTTB01	16bitTMRB0 コンペア一致 1		
22	INTTB10	16bitTMRB1 コンペア一致 0/ オーバーフロー		
23	INTTB11	16bitTMRB1 コンペア一致 1		
24	INTTB40	16bitTMRB4 コンペア一致 0/ オーバーフロー		
25	INTTB41	16bitTMRB4 コンペア一致 1		
26	INTTB50	16bitTMRB5 コンペア一致 0/ オーバーフロー		
27	INTTB51	16bitTMRB5 コンペア一致 1		
28	INTPMD0	PMD0 PWM 割り込み		
29	INTPMD1	PMD1 PWM 割り込み		
30	INTCAP00	16-bit TMRB0 インพุットキャプチャ 0		
31	INTCAP01	16-bit TMRB0 インพุットキャプチャ 1		
32	INTCAP10	16-bit TMRB1 インพุットキャプチャ 0		
33	INTCAP11	16-bit TMRB1 インพุットキャプチャ 1		
34	INTCAP40	16-bit TMRB4 インพุットキャプチャ 0		

表 6-3 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		
35	INTCAP41	16-bit TMRB4 インพุットキャプチャ 1	[H]/[L] レベル [↑]/[↓] エッジ選択可	CGIMCGB		
36	INTCAP50	16-bit TMRB5 インพุットキャプチャ 0				
37	INTCAP51	16-bit TMRB5 インพุットキャプチャ 1				
38	INT6	割り込み端子				
39	INT7	割り込み端子				
40	INTRX2	シリアル受信 (channel2)				
41	INTTX2	シリアル送信 (channel2)				
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	16bitTMRB2 コンペア一致 0/ オーバーフロー				
47	INTTB21	16bitTMRB2 コンペア一致 1				
48	INTTB30	16bitTMRB3 コンペア一致 0/ オーバーフロー				
49	INTTB31	16bitTMRB3 コンペア一致 1				
50	INTCAP20	16-bit TMRB2 インพุットキャプチャ 0				
51	INTCAP21	16-bit TMRB2 インพุットキャプチャ 1				
52	INTCAP30	16-bit TMRB3 インพุットキャプチャ 0				
53	INTCAP31	16-bit TMRB3 インพุ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				

表 6-3 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
58	INT8	割り込み端子	[H]/[L] レベル [↑]/[↓] エッジ選択可	CGIMCGC
59	INT9	割り込み端子		
60	INTA	割り込み端子		
61	INTB	割り込み端子		
62	INTENC0	エンコーダ入力 0 割り込み		
63	INTENC1	エンコーダ入力 1 割り込み		
64	INTRX3	シリアル受信 (channel3)		
65	INTTX3	シリアル送信 (channel3)		
66	INTTB60	16bitTMRB6 コンペアー致 0/ オーバーフロー		
67	INTTB61	16bitTMRB6 コンペアー致 1		
68	INTTB70	16bitTMRB7 コンペアー致 0/ オーバーフロー		
69	INTTB71	16bitTMRB7 コンペアー致 1		
70	INTCAP60	16-bit TMRB6 インプットキャプチャ 0		
71	INTCAP61	16-bit TMRB6 インプットキャプチャ 1		
72	INTCAP70	16-bit TMRB7 インプットキャプチャ 0		
73	INTCAP71	16-bit TMRB7 インプットキャプチャ 1		
74	INTC	割り込み端子	[H]/[L] レベル [↑]/[↓] エッジ選択可	CGIMCGD
75	INTD	割り込み端子		
76	INTE	割り込み端子		
77	INTF	割り込み端子		
78	INTSBI	シリアルバスインタフェース		

### 6.5.1.6 アクティブレベル

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

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

スタンバイ解除要因となる割り込みを使用するときにはクロックジェネレータレジスタの CGIMCGx<INTxEN> を有効にし、CGIMCGx<EMCGx> にアクティブレベルを設定します。周辺機能からの割り込み要求のアクティブレベルは表 6-3 で指定されているとおりに設定してください。

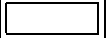

クロックジェネレータで検出された割り込みは、"High" レベル信号で CPU に通知されます。



## 6.5.2 処理詳細

### 6.5.2.1 処理の流れ

割り込みの処理の流れを以下に示します。

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

処理	内容	説明
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px; text-align: center;">検出のための準備</div> <div style="border: 1px solid black; padding: 5px; text-align: center;">要因発生のための準備</div>	<p>割り込みを検出するための設定を、NVIC レジスタで行います。スタンバイ解除要因となる割り込みは、クロックジェネレータの設定も必要です。</p> <p>共通 NVIC レジスタの設定 スタンバイ解除の設定 クロックジェネレータの設定</p> <p>割り込みの要因発生のための設定を行います。割り込みの種類により設定内容が異なります。</p> <p>外部割り込み ポートの設定 各周辺機能からの割り込み 各周辺機能の設定（使用する周辺機能の章を参照ください）</p>	「6.5.2.2 準備」
↓		
割り込み要因の発生	割り込みの要因が発生します。	
	スタンバイ解除要因となる割り込みは、クロックジェネレータを経由してCPUに接続されています。	「6.5.2.3 検出 (クロックジェネレータ)」
↓	CPU が割り込みを検出します。	「6.5.2.4 検出 (CPU)」
↓	CPU が割り込み処理を行います。	「6.5.2.5 CPU の処理」
↓	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。	「6.5.2.6 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から通常の処理プログラムに復帰します。	
元のプログラムへ復帰		

### 6.5.2.2 準備

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

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

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

以下に設定の手順と、手順ごとの具体的な設定方法を示します。

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

#### (1) CPU 割り込み禁止

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

このレジスタをセットするためには "MSR" 命令を使用します。

割り込みマスクレジスタ		
PRIMASK	←	"1"(割り込み禁止)

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

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

#### (2) CPU 割り込み設定

NVIC レジスタの割り込み優先度 レジスタで <PRI\_n> に優先度の設定を行います。

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

グループ優先度を設定する場合にはアプリケーション割り込みおよびリセット制御レジスタの <PRIGROUP> も設定します。

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

注) 「n」は該当する例外 / 割り込みの番号を示します。  
本製品では割り込み優先度レジスタの優先度設定領域は3ビットの構成になっています。

### (3) 要因の準備 (1) (外部割り込み)

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

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

注) 「x」は該当ポート番号、「m」は該当ビット、「n」はファンクションレジスタ番号を示します。  
STOP 以外のモードでは、PxIE で入力イネーブル設定であれば PxFR の設定によらず割り込みの入力が有効になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。また、「6.5.1.4 外部割り込み端子を使用する際の注意」の記載事項に注意してください。

### (4) 要因の準備 (2) (周辺機能からの割り込み)

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

### (5) 要因の準備 (3) (割り込み保留セットレジスタ)

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

NVIC レジスタ		
割り込み保留セット [m]	←	"1"

注) 「m」は該当ビットを示します。

### (6) クロックジェネレータの設定

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

割り込み許可の前に、不要な割り込み発生を防止するため割り込み要求のクリアをCGICRCGレジスタで行います。CGICRCGレジスタは、要因に対応した値を書き込むことで保持されていた割り込み要求をクリアすることができます。具体的な値は、「6.6.3.5 CGICRCG(CG割り込み要求クリアレジスタ)」を参照してください。

割り込み端子からの割り込み要求をスタンバイ解除要因として使用しない場合、クロックジェネレータの設定を行わずに使用することもできます。ただし、CPUが割り込み要因として検出するためには、「High」パルスまたは「High」レベルの信号を入力する必要があります。また、「6.5.1.4 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が発生する場合、割り込み保留クリアを行うと割り込み要因そのものが失われるため、この操作は不要です。

最後に、PRIMASKレジスタを"0"にクリアします。

NVIC レジスタ		
割り込み保留クリア [m]	←	"1"
割り込みイネーブルセット [m]	←	"1"
割り込みマスクレジスタ		
PRIMASK	←	"0"

注 1) 「m」は該当ビットを示します。

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

#### 6.5.2.3 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。

#### 6.5.2.4 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。

#### 6.5.2.5 CPU の処理

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

#### 6.5.2.6 割り込み処理ルーチンでの処理 (要因の取り下げ)

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

##### (1) 割り込み処理ルーチンでの処理

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

その他のレジスタについては必要に応じて退避します。

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

##### (2) 割り込み要因の取り下げ

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

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

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

## 6.6 例外 / 割り込み関連レジスタ

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

### 6.6.1 レジスタ一覧

NVIC レジスタ		Base Address = 0xE000_E000
レジスタ名	Address	
SysTick 制御およびステータスレジスタ	0x0010	
SysTick リロード値レジスタ	0x0014	
SysTick 現在値レジスタ	0x0018	
SysTick 較正值レジスタ	0x001C	
割り込みイネーブルセットレジスタ 1	0x0100	
割り込みイネーブルセットレジスタ 2	0x0104	
割り込みイネーブルセットレジスタ 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 ~ 0x0460	
ベクタテーブルオフセットレジスタ	0x0D08	
アプリケーション割り込みおよびリセット制御レジスタ	0x0D0C	
システムハンドラ優先度レジスタ	0x0D18, 0x0D1C, 0x0D20	
システムハンドラ制御および状態レジスタ	0x0D24	



クロックジェネレータレジスタ

Base Address = 0x4004\_0200

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

注) "Reserved" 表記のアドレスにはアクセスしないでください。

## 6.6.2 NVIC レジスタ

### 6.6.2.1 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。

6.6.2.2 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。

6.6.2.3 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。

6.6.2.4 SysTick 較正值レジスタ

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

Bit	Bit Symbol	Type	機能
31	NOREF	R	0: 参照クロックあり 1: 参照クロックなし
30	SKEW	R	0: 較正值は 10 ms 1: 較正值は 10 ms でない
29-24	-	R	リードすると "0" が読めます。
23-0	TENMS	R	較正值 外部参照クロックで 10 ms をカウントするために使用するリロード値 (0xC35) です。(注)

注) マルチショットで使用する場合、この値を -1 して使用してください。

6.6.2.5 割り込みイネーブルセットレジスタ 1

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

Bit	Bit Symbol	Type	機能
31-0	SETENA	R/W	割り込み番号 [31:0] [ライト] 1: 許可 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しています。 "1" をライトすることで該当する割り込みを許可します。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みの許可 / 禁止状態が確認できます。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.6 割り込みイネーブルセットレジスタ 2

	31	30	29	28	27	26	25	24
bit symbol	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

Bit	Bit Symbol	Type	機能
31-0	SETENA	R/W	割り込み番号 [63:32] [ライト] 1: 許可 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しています。 "1" をライトすることで該当する割り込みを許可します。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みの許可 / 禁止状態が確認できます。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

## 6.6.2.7 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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
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 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

Bit	Bit Symbol	Type	機能
31-15	-	R	リードすると"0"が読めます。
1413-0	SETENA	R/W	割り込み番号 [78:64] [ライト] 1: 許可 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しています。 "1"をライトすることで該当する割り込みを許可します。"0"の書き込みは意味を持ちません。 リードの場合、該当する割り込みの許可 / 禁止状態が確認できます。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.8 割り込みイネーブルクリアレジスタ 1

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

Bit	Bit Symbol	Type	機能
31-0	CLRENA	R/W	割り込み番号 [31:0] [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しており、割り込みを禁止したり、禁止されているかどうかを確認できます。 "1" をライトすることで該当する割り込みを禁止します。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みの許可 / 禁止の状態を確認できます。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。



6.6.2.9 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

Bit	Bit Symbol	Type	機能
31-0	CLRENA	R/W	割り込み番号 [63:32] [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しており、割り込みを禁止したり、禁止されているかどうかを確認できます。 "1" をライトすることで該当する割り込みを禁止します。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みの許可 / 禁止の状態を確認できます。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.10 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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
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 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

Bit	Bit Symbol	Type	機能
31-15	-	R	リードすると "0" が読めます。
14-0	CLRENA	R/W	割り込み番号 [78:64] [ライト] 1: 禁止 [リード] 0: 禁止 1: 許可 各ビットが指定された番号の割り込みに対応しており、割り込みを禁止したり、禁止されているかどうかを確認できます。 "1" をライトすることで該当する割り込みを禁止します。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みの許可 / 禁止の状態を確認できます。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.11 割り込み保留セットレジスタ 1

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

Bit	Bit Symbol	Type	機能
31-0	SETPEND	R/W	割り込み番号 [31:0] [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり 各ビットが指定された番号の割り込みに対応しており、割り込みを強制的に保留したり、保留されているかどうかを確認できます。 "1"をライトすることで該当する割り込みを保留します。ただし、すでに保留されている割り込みおよび禁止されている割り込みに対しては無効です。"0"の書き込みは意味を持ちません。 リードの場合、該当する割り込みが保留されているかどうかを示します。 このレジスタのビットをクリアするには、割り込み保留クリアレジスタの対応するビットに "1" をセットします。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.12 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

Bit	Bit Symbol	Type	機能
31-0	SETPEND	R/W	割り込み番号 [63:32] [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり 各ビットが指定された番号の割り込みに対応しており、割り込みを強制的に保留したり、保留されているかどうかを確認できます。 "1"をライトすることで該当する割り込みを保留します。ただし、すでに保留されている割り込みおよび禁止されている割り込みに対しては無効です。"0"の書き込みは意味を持ちません。 リードの場合、該当する割り込みが保留されているかどうかを示します。 このレジスタのビットをクリアするには、割り込み保留クリアレジスタの対応するビットに "1" をセットします。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.13 割り込み保留セットレジスタ 3

	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	-	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

Bit	Bit Symbol	Type	機能
31-15	-	R	リードすると "0" が読めます。
14-0	SETPEND	R/W	割り込み番号 [78:64] [ライト] 1: 保留する [リード] 0: 保留なし 1: 保留あり 各ビットが指定された番号の割り込みに対応しており、割り込みを強制的に保留したり、保留されているかどうかを確認できます。 "1" をライトすることで該当する割り込みを保留します。ただし、すでに保留されている割り込みおよび禁止されている割り込みに対しては無効です。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みが保留されているかどうかを示します。 このレジスタのビットをクリアするには、割り込み保留クリアレジスタの対応するビットに "1" をセットします。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.14 割り込み保留クリアレジスタ 1

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

Bit	Bit Symbol	Type	機能
31-0	CLRPEND	R/W	割り込み番号 [31:0] [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり 各ビットが指定された番号の割り込みに対応しており、保留された割り込みをクリアしたり、保留されているかどうかを確認できます。 "1"をライトすることで該当する保留された割り込みをクリアします。ただし、すでに処理が開始されている割り込みに対しては無効です。"0"の書き込みは意味を持ちません。 リードの場合、該当する割り込みが保留されているかどうかを示します。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.15 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

Bit	Bit Symbol	Type	機能
31-0	CLRPEND	R/W	割り込み番号 [63:32] [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり 各ビットが指定された番号の割り込みに対応しており、保留された割り込みをクリアしたり、保留されているかどうかを確認できます。 "1" をライトすることで該当する保留された割り込みをクリアします。ただし、すでに処理が開始されている割り込みに対しては無効です。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みが保留されているかどうかを示します。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。

6.6.2.16 割り込み保留クリアレジスタ 3

	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	-	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

Bit	Bit Symbol	Type	機能
31-15	-	R	リードすると "0" が読めます。
14-0	CLRPEND	R/W	割り込み番号 [78:64] [ライト] 1: 保留をクリアする [リード] 0: 保留なし 1: 保留あり 各ビットが指定された番号の割り込みに対応しており、保留された割り込みをクリアしたり、保留されているかどうかを確認できます。 "1" をライトすることで該当する保留された割り込みをクリアします。ただし、すでに処理が開始されている割り込みに対しては無効です。"0" の書き込みは意味を持ちません。 リードの場合、該当する割り込みが保留されているかどうかを示します。

注) 割り込みの内容と割り込み番号については、「6.5.1.5 要因一覧」を参照してください。



## 6.6.2.17 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_78		PRI_77		PRI_76

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。

6.6.2.18 ベクタテーブルオフセットレジスタ

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

Bit	Bit Symbol	Type	機能
31-30	-	R	リードすると "0" が読めます。
29	TBLBASE	R/W	テーブルベース ベクタテーブルを置くメモリ空間を指定します。 0: コード空間 1: SRAM 空間
28-7	TBLOFF	R/W	オフセット値 TBLBASE で指定した領域の先頭からのオフセット値を設定します。 オフセットはテーブルにある例外の数に基づいてアラインされる必要があります。16 個までの割り込みが使える、最小のアライメントは 32 ワードになります。割り込みの数がより多い場合は、次の 2 のべき乗まで切り上げて、アライメントを調整する必要があります。
6-0	-	R	リードすると "0" が読めます。

6.6.2.19 アプリケーション割り込みおよびリセット制御レジスタ

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

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

注 1) 本製品はリトルエンディアン固定です。

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

6.6.2.20 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。

## 6.6.2.21 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	BUSFAULT ENA	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
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。

### 6.6.3 クロックジェネレータレジスタ

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

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



Bit	Bit Symbol	Type	機能
31	–	R	リードすると"0"が読めます。
30-28	EMCG3[2:0]	R/W	INT3 スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
27-26	EMST3[1:0]	R	INT3 スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
25	–	R	リードすると不定値が読まれます。
24	INT3EN	R/W	INT3 解除入力 0: ディセーブル 1: イネーブル
23	–	R	リードすると"0"が読めます。
22-20	EMCG2[2:0]	R/W	INT2 スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
19-18	EMST2[1:0]	R	INT2 スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
17	–	R	リードすると不定値が読まれます。
16	INT2EN	R/W	INT2 解除入力 0: ディセーブル 1: イネーブル
15	–	R	リードすると"0"が読めます。
14-12	EMCG1[2:0]	R/W	INT1 スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
11-10	EMST1[1:0]	R	INT1 スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
9	–	R	リードすると不定値が読まれます。
8	INT1EN	R/W	INT1 解除入力 0: ディセーブル 1: イネーブル
7	–	R	リードすると"0"が読めます。
6-4	EMCG0[2:0]	R/W	INT0 スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ

Bit	Bit Symbol	Type	機能
3-2	EMST0[1:0]	R	INT0 スタンバイ解除要求のアクティブ状態 00: - 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
1	-	R	リードすると不定値が読めます。
0	INT0EN	R/W	INT0 解除入力 0: ディセーブル 1: イネーブル

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

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

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

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



Bit	Bit Symbol	Type	機能
31	–	R	リードすると"0"が読めます。
30-28	EMCG7[2:0]	R/W	INT7スタンバイ解除要求のアクティブ状態を設定(101~111:設定禁止) 000:"Low"レベル 001:"High"レベル 010:立ち下がりエッジ 011:立ち上がりエッジ 100:両エッジ
27-26	EMST7[1:0]	R	INT7スタンバイ解除要求のアクティブ状態 00:– 01:立ち上がりエッジ 10:立ち下がりエッジ 11:両エッジ
25	–	R	リードすると不定値が読まれます。
24	INT7EN	R/W	INT7解除入力 0:ディセーブル 1:イネーブル
23	–	R	リードすると"0"が読めます。
22-20	EMCG6[2:0]	R/W	INT6スタンバイ解除要求のアクティブ状態を設定(101~111:設定禁止) 000:"Low"レベル 001:"High"レベル 010:立ち下がりエッジ 011:立ち上がりエッジ 100:両エッジ
19-18	EMST6[1:0]	R	INT6スタンバイ解除要求のアクティブ状態 00:– 01:立ち上がりエッジ 10:立ち下がりエッジ 11:両エッジ
17	–	R	リードすると不定値が読まれます。
16	INT6EN	R/W	INT6解除入力 0:ディセーブル 1:イネーブル
15	–	R	リードすると"0"が読めます。
14-12	EMCG5[2:0]	R/W	INT5スタンバイ解除要求のアクティブ状態を設定(101~111:設定禁止) 000:"Low"レベル 001:"High"レベル 010:立ち下がりエッジ 011:立ち上がりエッジ 100:両エッジ
11-10	EMST5[1:0]	R	INT5スタンバイ解除要求のアクティブ状態 00:– 01:立ち上がりエッジ 10:立ち下がりエッジ 11:両エッジ
9	–	R	リードすると不定値が読まれます。
8	INT5EN	R/W	INT5解除入力 0:ディセーブル 1:イネーブル
7	–	R	リードすると"0"が読めます。
6-4	EMCG4[2:0]	R/W	INT4スタンバイ解除要求のアクティブ状態を設定(101~111:設定禁止) 000:"Low"レベル 001:"High"レベル 010:立ち下がりエッジ 011:立ち上がりエッジ 100:両エッジ

Bit	Bit Symbol	Type	機能
3-2	EMST4[1:0]	R	INT4 スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
1	–	R	リードすると不定値が読めます。
0	INT4EN	R/W	INT4 解除入力 0: ディセーブル 1: イネーブル

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

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

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

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





Bit	Bit Symbol	Type	機能
31	–	R	リードすると"0"が読めます。
30-28	EMCGB[2:0]	R/W	INTB スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
27-26	EMSTB[1:0]	R	INTB スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
25	–	R	リードすると不定値が読まれます。
24	INTBEN	R/W	INTB 解除入力 0: ディセーブル 1: イネーブル
23	–	R	リードすると"0"が読めます。
22-20	EMCGA[2:0]	R/W	INTA スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
19-18	EMSTA[1:0]	R	INTA スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
17	–	R	リードすると不定値が読まれます。
16	INTAEN	R/W	INTA 解除入力 0: ディセーブル 1: イネーブル
15	–	R	リードすると"0"が読めます。
14-12	EMCG9[2:0]	R/W	INT9 スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
11-10	EMST9[1:0]	R	INT9 スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
9	–	R	リードすると不定値が読まれます。
8	INT9EN	R/W	INT9 解除入力 0: ディセーブル 1: イネーブル
7	–	R	リードすると"0"が読めます。
6-4	EMCG8[2:0]	R/W	INT8 スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ

Bit	Bit Symbol	Type	機能
3-2	EMST8[1:0]	R	INT8 スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
1	–	R	リードすると不定値が読めます。
0	INT8EN	R/W	INT8 解除入力 0: ディセーブル 1: イネーブル

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

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

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

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



Bit	Bit Symbol	Type	機能
31	–	R	リードすると"0"が読めます。
30-28	EMCGF[2:0]	R/W	INTF スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
27-26	EMSTF[1:0]	R	INTF スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
25	–	R	リードすると不定値が読まれます。
24	INTFEN	R/W	INTF 解除入力 0: ディセーブル 1: イネーブル
23	–	R	リードすると"0"が読めます。
22-20	EMCGE[2:0]	R/W	INTE スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
19-18	EMSTE[1:0]	R	INTE スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
17	–	R	リードすると不定値が読まれます。
16	INTEEN	R/W	INTE 解除入力 0: ディセーブル 1: イネーブル
15	–	R	リードすると"0"が読めます。
14-12	EMCGD[2:0]	R/W	INTD スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ
11-10	EMSTD[1:0]	R	INTD スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
9	–	R	リードすると不定値が読まれます。
8	INTDEN	R/W	INTD 解除入力 0: ディセーブル 1: イネーブル
7	–	R	リードすると"0"が読めます。
6-4	EMCGC[2:0]	R/W	INTC スタンバイ解除要求のアクティブ状態を設定 (101~111: 設定禁止) 000: "Low" レベル 001: "High" レベル 010: 立ち下がりエッジ 011: 立ち上がりエッジ 100: 両エッジ

Bit	Bit Symbol	Type	機能
3-2	EMSTC[1:0]	R	INTC スタンバイ解除要求のアクティブ状態 00: – 01: 立ち上がりエッジ 10: 立ち下がりエッジ 11: 両エッジ
1	–	R	リードすると不定値が読まれます。
0	INTCEN	R/W	INTC 解除入力 0: ディセーブル 1: イネーブル

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

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

## 6.6.3.5 CGICRCG(CG 割り込み要求クリアレジスタ)

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

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると"0"が読めます。
4-0	ICRCG[4:0]	W	割り込み要求をクリア 0_0000:INT0            0_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 ~ 1_1111: 設定禁止 リードすると"0"が読めます

6.6.3.6 CGNMIFLG(NMI フラグレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	NMIFLG0
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	NMIFLG0	R	NMI 起動要因フラグ 0: 要因なし 1: WDT による NMI 発生

注) <NMIFLG> は読み出すと "0" にクリアされます。



## 6.6.3.7 CGRSTFLG(リセットフラグレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
パワーオンリセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
パワーオンリセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
パワーオンリセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	OFDRSTF	DBGRSTF	VLDRSTF	WDTRSTF	PINRSTF	PONRSTF
パワーオンリセット後	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 によるリセットによるリセットフラグ
3	VLDRSTF	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	RESET 端子フラグ 0: 0 ライト 1: RESET 端子によるリセットフラグ
0	PONRSTF	R/W	パワーオンフラグ 0: 0 ライト 1: 電源投入時のリセットによるリセットフラグ

注1) CPU の NVIC 内にある、アプリケーション割り込みおよびリセット制御レジスタの <SYSRESETREQ> ビットのセットにより発生したリセットであることを示します。

注2) 本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7章 入出力ポート

### 7.1 ポート機能

#### 7.1.1 機能一覧

TMPM376FDDFG/FDFGには82本のポートがあり、ポート機能のほかに内蔵する周辺機能に対する入出力端子としても使用されます。

表7-1にポート機能の一覧を示します。

表7-1 ポート機能一覧

ポート	端子名	入出力	Pull-up Pull-down	Schmitt 入力	ノイズ フィルタ	プログラ マブル オープン ドレイン	機能端子名
ポートA							
	PA0	入出力	Pull-up / Pull-down	○	○(INT3のみ)	○	TB0IN, INT3
	PA1	入出力	Pull-up / Pull-down	○	-	○	TB0OUT
	PA2	入出力	Pull-up / Pull-down	○	○(INT4のみ)	○	TB1IN, INT4
	PA3	入出力	Pull-up / Pull-down	○	-	○	TB1OUT
	PA4	入出力	Pull-up / Pull-down	○	-	○	SCLK1, $\overline{CTS1}$
	PA5	入出力	Pull-up / Pull-down	○	-	○	TXD1, TB6OUT
	PA6	入出力	Pull-up / Pull-down	○	-	○	RXD1, TB6IN
	PA7	入出力	Pull-up / Pull-down	○	○(INT8のみ)	○	TB4IN, INT8
ポートB							
	PB0	入出力	Pull-up / Pull-down	○	-	○	TRACECLK
	PB1	入出力	Pull-up / Pull-down	○	-	○	TRACEDATA0
	PB2	入出力	Pull-up / Pull-down	○	-	○	TRACEDATA1
	PB3	入出力	Pull-up / Pull-down	○	-	○	TMS / SWDIO
	PB4	入出力	Pull-up / Pull-down	○	-	○	TCK / SWCLK
	PB5	入出力	Pull-up / Pull-down	○	-	○	TDO / SWV
	PB6	入出力	Pull-up / Pull-down	○	-	○	TDI
	PB7	入出力	Pull-up / Pull-down	○	○	○	$\overline{TRST}$
ポートC							
	PC0	入出力	Pull-up / Pull-down	○	-	○	UO0
	PC1	入出力	Pull-up / Pull-down	○	-	○	XO0
	PC2	入出力	Pull-up / Pull-down	○	-	○	VO0
	PC3	入出力	Pull-up / Pull-down	○	-	○	YO0
	PC4	入出力	Pull-up / Pull-down	○	-	○	WO0
	PC5	入出力	Pull-up / Pull-down	○	-	○	ZO0
	PC6	入出力	Pull-up / Pull-down	○	-	○	$\overline{EMG0}$
	PC7	入出力	Pull-up / Pull-down	○	-	○	$\overline{OVV0}$

○: あり

-: なし

表 7-1 ポート機能一覧

ポート	端子名	入出力	Pull-up Pull-down	Schmitt 入力	ノイズ フィルタ	プログラ マブル オープン ドレイン	機能端子名
ポート D							
	PD0	入出力	Pull-up / Pull-down	○	—	○	ENCA0 , TB5IN
	PD1	入出力	Pull-up / Pull-down	○	—	○	ENCB0 , TB5OUT
	PD2	入出力	Pull-up / Pull-down	○	—	○	ENCZ0
	PD3	入出力	Pull-up / Pull-down	○	○	○	INT9
	PD4	入出力	Pull-up / Pull-down	○	—	○	SCLK2 , $\overline{CTS2}$
	PD5	入出力	Pull-up / Pull-down	○	—	○	TXD2
	PD6	入出力	Pull-up / Pull-down	○	—	○	RXD2
ポート E							
	PE0	入出力	Pull-up / Pull-down	○	—	○	TXD0
	PE1	入出力	Pull-up / Pull-down	○	—	○	RXD0
	PE2	入出力	Pull-up / Pull-down	○	—	○	SCLK0 , $\overline{CTS0}$
	PE3	入出力	Pull-up / Pull-down	○	—	○	TB4OUT
	PE4	入出力	Pull-up / Pull-down	○	○(INT5のみ)	○	TB2IN , INT5
	PE5	入出力	Pull-up / Pull-down	○	—	○	TB2OUT
	PE6	入出力	Pull-up / Pull-down	○	○(INT6のみ)	○	TB3IN , INT6
	PE7	入出力	Pull-up / Pull-down	○	○(INT7のみ)	○	TB3OUT , INT7
ポート F							
	PF0	入出力	Pull-up / Pull-down	○	—	○	TB7IN , $\overline{BOOT}$
	PF1	入出力	Pull-up / Pull-down	○	—	○	TB7OUT
	PF2	入出力	Pull-up / Pull-down	○	—	○	ENCA1 , SCLK3 , $\overline{CTS3}$
	PF3	入出力	Pull-up / Pull-down	○	—	○	ENCB1 , TXD3
	PF4	入出力	Pull-up / Pull-down	○	—	○	ENCZ1 , RXD3
ポート G							
	PG0	入出力	Pull-up / Pull-down	○	—	○	UO1
	PG1	入出力	Pull-up / Pull-down	○	—	○	XO1
	PG2	入出力	Pull-up / Pull-down	○	—	○	VO1
	PG3	入出力	Pull-up / Pull-down	○	—	○	YO1
	PG4	入出力	Pull-up / Pull-down	○	—	○	WO1
	PG5	入出力	Pull-up / Pull-down	○	—	○	ZO1
	PG6	入出力	Pull-up / Pull-down	○	—	○	$\overline{EMG1}$
	PG7	入出力	Pull-up / Pull-down	○	—	○	$\overline{OVV1}$
ポート H							
	PH0	入出力	Pull-up / Pull-down	○	○(INT0のみ)	○	INT0 , AINA0
	PH1	入出力	Pull-up / Pull-down	○	○(INT1のみ)	○	INT1 , AINA1
	PH2	入出力	Pull-up / Pull-down	○	○(INT2のみ)	○	INT2 , AINA2
	PH3	入出力	Pull-up / Pull-down	○	—	○	AINA3

○: あり  
—: なし

表 7-1 ポート機能一覧

ポート	端子名	入出力	Pull-up Pull-down	Schmitt 入力	ノイズ フィルタ	プログラ マブル オープン ドレイン	機能端子名
	PH4	入出力	Pull-up / Pull-down	o	-	o	AINA4
	PH5	入出力	Pull-up / Pull-down	o	-	o	AINA5
	PH6	入出力	Pull-up / Pull-down	o	-	o	AINA6
	PH7	入出力	Pull-up / Pull-down	o	-	o	AINA7
ポート I							
	PI0	入出力	Pull-up / Pull-down	o	-	o	AINA8
	PI1	入出力	Pull-up / Pull-down	o	-	o	AINA9 / AINB0
	PI2	入出力	Pull-up / Pull-down	o	-	o	AINA10 / AINB1
	PI3	入出力	Pull-up / Pull-down	o	-	o	AINA11 / AINB2
ポート J							
	PJ0	入出力	Pull-up / Pull-down	o	-	o	AINB3
	PJ1	入出力	Pull-up / Pull-down	o	-	o	AINB4
	PJ2	入出力	Pull-up / Pull-down	o	-	o	AINB5
	PJ3	入出力	Pull-up / Pull-down	o	-	o	AINB6
	PJ4	入出力	Pull-up / Pull-down	o	-	o	AINB7
	PJ5	入出力	Pull-up / Pull-down	o	-	o	AINB8
	PJ6	入出力	Pull-up / Pull-down	o	o(INTCのみ)	o	INTC , AINB9
	PJ7	入出力	Pull-up / Pull-down	o	o(INTDのみ)	o	INTD , AINB10
ポート K							
	PK0	入出力	Pull-up / Pull-down	o	o(INTEのみ)	o	INTE , AINB11
	PK1	入出力	Pull-up / Pull-down	o	o(INTFのみ)	o	INTF , AINB12
ポート L							
	PL0	入力	-	o	o	-	INTB
	PL1	入力	-	o	o	-	INTA
ポート M							
	PM0	入出力	Pull-up / Pull-down	o	-	o	X1
	PM1	入出力	Pull-up / Pull-down	o	-	o	X2
ポート N							
	PN0	入出力	Pull-up / Pull-down	o	-	o	SO / SDA
	PN1	入出力	Pull-up / Pull-down	o	-	o	SI / SCL
	PN2	入出力	Pull-up / Pull-down	o	-	o	SCK
	PN3	入出力	Pull-up / Pull-down	o	-	o	TB7IN

o: あり

-: なし

注) ノイズフィルタのノイズ除去幅は、Typ. 条件で約 30ns です。

## 7.1.2 ポートレジスタ概略説明

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。

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

7.1.3 STOP モード中のポート状態

STOP モード中の入力と出力の状態を、クロック / モード制御部の CGSTBYCR<DRVE> で制御することができます。

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

STOP モード時の端子状態を表 7-2 に示します。

表 7-2 STOP モード時の端子状態

	機能	入出力	<DRVE> = 0	<DRVE> = 1
ポート 以外	RESET, MODE	入力専用	o	
ポート	X1	入力専用	x	
	X2	出力専用	"High" レベル出力	
	TMS TCK TDI TRST	入力	o	
	TDO	出力	データ有効時はイネーブル データ無効時はディセーブル	
	SWCLK	入力	o	
	SWDIO	入力	o	
		出力	データ有効時はイネーブル データ無効時はディセーブル	
	TRACECLK TRACEDATA0 TRACEDATA1 SWV	出力	o	
	U00,1 VO0,1 WO0,1 XO0,1 YO0,1 ZO0,1	出力	データ有効時はイネーブル データ無効時はディセーブル	
	INT0, INT1, INT2 INT3, INT4, INT5 INT6, INT7, INT8 INT9, INTA, INTB INTC, INTD, INTE INTF	入力	o	
上記以外の機能端子または汎用入出力として使用	入力	x	o	
	出力	x	o	

o: 入力または出力がイネーブルであることを示します。

x: 入力または出力がディセーブルであることを示します。

## 7.2 ポート機能詳細

本章では、各ポートのレジスタの詳細について説明します。

回路構成については、本章では「回路タイプ」のみ記載しています。具体的な回路図は「7.3 ポート回路図」に記載していますので、そちらを参照してください。

### 7.2.1 ポート A (PA0 ~ PA7)

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

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

ポート A には複数のファンクションレジスタがあります。汎用ポート機能として使用する場合は、全てのファンクションレジスタの該当するビットに "0" を設定してください。汎用ポート機能以外で使用する場合には、ファンクションレジスタの該当ビットに "1" を設定してください。複数のファンクションレジスタを同時に "1" に設定しないでください。

割り込み入力を STOP モード解除に使用する場合、PAFR で機能設定にし、PAIE で入力設定をしてください。この設定では、クロック / モード制御部の CGSTBYCR<DRVE> で STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

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

#### 7.2.1.1 ポート A 回路タイプ

	7	6	5	4	3	2	1	0
Type	T12	T11	T13	T9	T2	T12	T2	T12

#### 7.2.1.2 ポート A レジスタ一覧

		Base Address = 0x4000_0000
レジスタ名		Address(Base+)
ポート A データ レジスタ	PADATA	0x0000
ポート A 出力コントロールレジスタ	PACR	0x0004
ポート A ファンクションレジスタ 1	PAFR1	0x0008
ポート A ファンクションレジスタ 2	PAFR2	0x000C
ポート A オープンドレインコントロールレジスタ	PAOD	0x0028
ポート A プルアップコントロールレジスタ	PAPUP	0x002C
ポート A プルダウンコントロールレジスタ	PAPDN	0x0030
ポート A 入力コントロールレジスタ	PAIE	0x0038



## 7.2.1.3 PADATA (ポート A データレジスタ)

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

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

## 7.2.1.4 PACR (ポート A 出力コントロールレジスタ)

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

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

7.2.1.5 PAFR1 (ポート A ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7F1	PA6F1	PA5F1	PA4F1	PA3F1	PA2F1	PA1F1-	PA0F1
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	PA7F1	R/W	0: PORT 1: TB4IN
6	PA6F1	R/W	0: PORT 1: RXD1
5	PA5F1	R/W	0: PORT 1: TXD1
4	PA4F1	R/W	0: PORT 1: SCLK1
3	PA3F1	R/W	0: PORT 1: TB1OUT
2	PA2F1	R/W	0: PORT 1: TB1IN
1	PA1F1	R/W	0: PORT 1: TB0OUT
0	PA0F1	R/W	0: PORT 1: TB0IN

## 7.2.1.6 PAFR2 (ポート A ファンクションレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7F2	PA6F2	PA5F2	PA4F2	-	PA2F2	-	PA0F2
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	PA7F2	R/W	0: PORT 1: INT8
6	PA6F2	R/W	0: PORT 1: TB6IN
5	PA5F2	R/W	0: PORT 1: TB6OUT
4	PA4F2	R/W	0: PORT 1: CTS1
3	-	R	リードすると "0" が読めます。
2	PA2F2	R/W	0: PORT 1: INT4
1	-	R	リードすると "0" が読めます。
0	PA0F2	R/W	0: PORT 1: INT3

7.2.1.7 PAOD (ポート A オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7OD	PA6OD	PA5OD	PA4OD	PA3OD	PA2OD	PA1OD	PA0OD
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	PA7OD-PA0OD	R/W	0: CMOS 1: オープンドレイン

7.2.1.8 PAPUP (ポート A ブルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7UP	PA6UP	PA5UP	PA4UP	PA3UP	PA2UP	PA1UP	PA0UP
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	PA7UP-PA0UP	R/W	ブルアップ 0: 禁止 1: 許可

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

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PA7DN	PA6DN	PA5DN	PA4DN	PA3DN	PA2DN	PA1DN	PA0DN
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	PA7DN-PA0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.1.10 PAIE (ポート A 入力コントロールレジスタ)

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

## 7.2.2 ポート B (PB0 ~ PB7)

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

リセット後 PB3、PB4、PB5、PB6、PB7 はデバッグ通信機能になります。

PB3 は TMS または SWDIO 機能で入力・出力・プルアップイネーブル、PB4 は TCK または SWCLK 機能で入力・プルダウンイネーブルとなります。

PB5 は TDO または SWV 機能で出力イネーブル、PB6 は TDI 機能で入力・プルアップイネーブル、PB7 は TRST 機能で入力・プルアップイネーブルとなります。

その他のビットは、リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

注) PB3, PB5 が機能設定の場合、CGSTBYCR<DRVE> の設定によらず、STOP モード中でも出力が有効な状態のまま保持されます。

### 7.2.2.1 ポート B 回路タイプ

	7	6	5	4	3	2	1	0
Type	T7	T7	T19	T8	T6	T18	T18	T18

### 7.2.2.2 ポート B レジスタ一覧

Base Address = 0x4000\_0040

レジスタ名		Address(Base+)
ポート B データ レジスタ	PBDATA	0x0000
ポート B 出力コントロールレジスタ	PBCR	0x0004
ポート B ファンクションレジスタ 1	PBFR1	0x0008
ポート B オープンドレインコントロールレジスタ	PBOD	0x0028
ポート B プルアップコントロールレジスタ	PBPUP	0x002C
ポート B プルダウンコントロールレジスタ	PBPDN	0x0030
ポート B 入力コントロールレジスタ	PBIE	0x0038

## 7.2.2.3 PBDATA (ポート B データレジスタ)

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

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

## 7.2.2.4 PBCR (ポート B 出力コントロールレジスタ)

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

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

7.2.2.5 PBFR1 (ポート B ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PB7F1	PB6F1	PB5F1	PB4F1	PB3F1	PB2F1	PB1F1	PB0F1
リセット後	1	1	1	1	1	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	PB7F1	R/W	0: PORT 1: $\overline{\text{TRST}}$
6	PB6F1	R/W	0: PORT 1: TDI
5	PB5F1	R/W	0: PORT 1: TDO / SWV
4	PB4F1	R/W	0: PORT 1: TCK / SWCLK
3	PB3F1	R/W	0: PORT 1: TMS / SWDIO
2	PB2F1	R/W	0: PORT 1: TRACEDATA1
1	PB1F1	R/W	0: PORT 1: TRACEDATA0
0	PB0F1	R/W	0: PORT 1: TRACECLK



## 7.2.2.6 PBOD (ポート B オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PB7OD	PB6OD	PB5OD	PB4OD	PB3OD	PB2OD	PB1OD	PB0OD
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	PB7OD- PB0OD	R/W	0: CMOS 1: オープンドレイン

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

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

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

7.2.2.8 PBPDN (ポート B プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PB7DN	PB6DN	PB5DN	PB4DN	PB3DN	PB2DN	PB1DN	PB0DN
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。
7-0	PB7DN- PB0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.2.9 PBIE (ポート B 入力コントロールレジスタ)

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

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

### 7.2.3 ポート C (PC0 ~ PC7)

ポート C はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用入出力ポート機能以外に三相モータ制御のための入出力端子機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

#### 7.2.3.1 ポート C 回路タイプ

	7	6	5	4	3	2	1	0
Type	T3	T3	T1	T1	T1	T1	T1	T1

#### 7.2.3.2 ポート C レジスタ一覧

Base Address = 0x4000\_0080

レジスタ名		Address(Base+)
ポート C データ レジスタ	PCDATA	0x0000
ポート C 出力コントロールレジスタ	PCCR	0x0004
ポート C ファンクションレジスタ 1	PCFR1	0x0008
ポート C オープンドレインコントロールレジスタ	PCOD	0x0028
ポート C プルアップコントロールレジスタ	PCPUP	0x002C
ポート C プルダウンコントロールレジスタ	PCPDN	0x0030
ポート C 入力コントロールレジスタ	PCIE	0x0038

## 7.2.3.3 PCDATA (ポート C データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
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	PC7-PC0	R/W	ポート C データレジスタ

## 7.2.3.4 PCCR (ポート C 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PC7C	PC6C	PC5C	PC4C	PC3C	PC2C	PC1C	PC0C
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	PC7C-PC0C	R/W	出力 0: 禁止 1: 許可

7.2.3.5 PCFR1 (ポートC ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PC7F1	PC6F1	PC5F1	PC4F1	PC3F1	PC2F1	PC1F1	PC0F1
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	PC7F1	R/W	0: PORT 1: $\overline{OVV0}$
6	PC6F1	R/W	0: PORT 1: $\overline{EMG0}$
5	PC5F1	R/W	0: PORT 1: Z00
4	PC4F1	R/W	0: PORT 1: W00
3	PC3F1	R/W	0: PORT 1: Y00
2	PC2F1	R/W	0: PORT 1: V00
1	PC1F1	R/W	0: PORT 1: X00
0	PC0F1	R/W	0: PORT 1: U00

## 7.2.3.6 PCOD (ポート C オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PC7OD	PC6OD	PC5OD	PC4OD	PC3OD	PC2OD	PC1OD	PC0OD
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	PC7OD- PC0OD	R/W	0: CMOS 1: オープンドレイン

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

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PC7UP	PC6UP	PC5UP	PC4UP	PC3UP	PC2UP	PC1UP	PC0UP
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	PC7UP- PC0UP	R/W	プルアップ 0: 禁止 1: 許可

### 7.2.3.8 PCPDN (ポート C プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PC7DN	PC6DN	PC5DN	PC4DN	PC3DN	PC2DN	PC1DN	PC0DN
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	PC7DN-PC0DN	R/W	プルダウン 0: 禁止 1: 許可

### 7.2.3.9 PCIE (ポート C 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PC7IE	PC6IE	PC5IE	PC4IE	PC3IE	PC2IE	PC1IE	PC0IE
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	PC7IE-PC0IE	R/W	入力 0: 禁止 1: 許可



### 7.2.4 ポート D (PD0 ~ PD6)

ポート D はビット単位で入出力の指定ができる 7 ビットの汎用入出力ポートです。汎用入出力ポート機能以外にシリアルインタフェース機能 (SIO / UART)、外部割り込み入力機能、16 ビットタイマ入出力機能、エンコーダ入力機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

ポート D には複数のファンクションレジスタがあります。汎用ポート機能として使用する場合は、全てのファンクションレジスタの該当するビットに "0" を設定してください。汎用ポート機能以外で使用する場合には、ファンクションレジスタの該当ビットに "1" を設定してください。複数のファンクションレジスタを同時に "1" に設定しないでください。

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

割り込み入力を STOP モード解除に使用する場合、PDFR1 で機能設定にし、PDIE で入力設定をしてください。この設定では、クロック / モード制御部の CGSTBYCR<DRVE> で STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

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

#### 7.2.4.1 ポート D 回路タイプ

	7	6	5	4	3	2	1	0
Type	-	T3	T2	T9	T4	T3	T10	T11

#### 7.2.4.2 ポート D レジスタ一覧

Base Address = 0x4000\_00C0

レジスタ名		Address(Base+)
ポート D データ レジスタ	PDDATA	0x0000
ポート D 出力コントロールレジスタ	PDCR	0x0004
ポート D ファンクションレジスタ 1	PDFR1	0x0008
ポート D ファンクションレジスタ 2	PDFR2	0x000C
ポート D オープンドレインコントロールレジスタ	PDOD	0x0028
ポート D プルアップコントロールレジスタ	PDPUP	0x002C
ポート D プルダウンコントロールレジスタ	PDPDN	0x0030
ポート D 入力コントロールレジスタ	PDIE	0x0038

## 7.2.4.3 PDDATA (ポートD データレジスタ)

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

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6-0	PD6-PD0	R/W	ポートD データレジスタ

## 7.2.4.4 PDCR (ポートD 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PD6C	PD5C	PD4C	PD3C	PD2C	PD1C	PD0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると"0"が読めます。
6-0	PD6C-PD0C	R/W	出力 0: 禁止 1: 許可

## 7.2.4.5 PDFR1 (ポートD ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PD6F1	PD5F1	PD4F1	PD3F1	PD2F1	PD1F1	PD0F1
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	PD6F1	R/W	0: PORT 1: RXD2
5	PD5F1	R/W	0: PORT 1: TXD2
4	PD4F1	R/W	0: PORT 1: SCLK2
3	PD3F1	R/W	0: PORT 1: INT9
2	PD2F1	R/W	0: PORT 1: ENCZ0
1	PD1F1	R/W	0: PORT 1: ENCB0
0	PD0F1	R/W	0: PORT 1: ENCA0

7.2.4.6 PDFR2 (ポートD ファンクションレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	PD4F2	-	-	PD1F2	PD0F2
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	PD4F2	R/W	0: PORT 1: CTS2
3-2	-	R	リードすると"0"が読めます。
1	PD1F2	R/W	0: PORT 1: TB5OUT
0	PD0F2	R/W	0: PORT 1: TB5IN

## 7.2.4.7 PDOD (ポート D オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PD6OD	PD5OD	PD4OD	PD3OD	PD2OD	PD1OD	PD0OD
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると "0" が読めます。
6-0	PD6OD- PD0OD	R/W	0: CMOS 1: オープンドレイン

7.2.4.8 PDPUP (ポート D プルアップコントロールレジスタ)

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

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると "0" が読めます。
6-0	PD6UP- PD0UP	R/W	プルアップ 0: 禁止 1: 許可

7.2.4.9 PDPDN (ポート D プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	PD6DN	PD5DN	PD4DN	PD3DN	PD2DN	PD1DN	PD0DN
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると "0" が読めます。
6-0	PD6DN- PD0DN	R/W	プルダウン 0: 禁止 1: 許可

7.2.4.10 PDIE (ポート D 入力コントロールレジスタ)

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

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると "0" が読めます。
6-0	PD6IE-PD0IE	R/W	入力 0: 禁止 1: 許可

### 7.2.5 ポート E (PE0 ~ PE7)

ポート E はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用入出力ポート機能以外にシリアルインタフェース機能 (SIO / UART)、16 ビットタイマ入出力機能、外部割り込み入力機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

ポート E には複数のファンクションレジスタがあります。汎用ポート機能として使用する場合は、全てのファンクションレジスタの該当するビットに "0" を設定してください。汎用ポート機能以外で使用する場合には、ファンクションレジスタの該当ビットに "1" を設定してください。複数のファンクションレジスタを同時に "1" に設定しないでください。

割り込み入力を STOP モード解除に使用する場合、PEFR2 で機能設定にし、PEIE で入力設定をしてください。この設定では、クロック / モード制御部の CGSTBYCR<DRVE> で STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

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

#### 7.2.5.1 ポート E 回路タイプ

	7	6	5	4	3	2	1	0
Type	T14	T12	T2	T12	T2	T9	T3	T2

#### 7.2.5.2 ポート E レジスタ一覧

Base Address = 0x4000\_0100

レジスタ名		Address(Base+)
ポート E データ レジスタ	PEDATA	0x0000
ポート E 出力コントロールレジスタ	PECR	0x0004
ポート E ファンクションレジスタ 1	PEFR1	0x0008
ポート E ファンクションレジスタ 2	PEFR2	0x000C
ポート E オープンドレインコントロールレジスタ	PEOD	0x0028
ポート E プルアップコントロールレジスタ	PEPUP	0x002C
ポート E プルダウンコントロールレジスタ	PEPDN	0x0030
ポート E 入力コントロールレジスタ	PEIE	0x0038



## 7.2.5.3 PEDATA (ポートE データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PE7-PE0	R/W	ポートE データレジスタ

## 7.2.5.4 PECCR (ポートE 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7C	PE6C	PE5C	PE4C	PE3C	PE2C	PE1C	PE0C
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7-0	PE7C-PE0C	R/W	出力 0: 禁止 1: 許可

7.2.5.5 PEFR1 (ポートE ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7F1	PE6F1	PE5F1	PE4F1	PE3F1	PE2F1	PE1F1	PE0F1
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	PE7F1	R/W	0: PORT 1: TB3OUT
6	PE6F1	R/W	0: PORT 1: TB3IN
5	PE5F1	R/W	0: PORT 1: TB2OUT
4	PE4F1	R/W	0: PORT 1: TB2IN
3	PE3F1	R/W	0: PORT 1: TB4OUT
2	PE2F1	R/W	0: PORT 1: SCLK0
1	PE1F1	R/W	0: PORT 1: RXD0
0	PE0F1	R/W	0: PORT 1: TXD0

## 7.2.5.6 PEFR2 (ポート E ファンクションレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7F2	PE6F2	-	PE4F2	-	PE2F2	-	-
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	PE7F2	R/W	0: PORT 1: INT7
6	PE6F2	R/W	0: PORT 1: INT6
5	-	R	リードすると "0" が読めます。
4	PE4F2	R/W	0: PORT 1: INT5
3	-	R	リードすると "0" が読めます。
2	PE2F2	R/W	0: PORT 1: CTS0
1-0	-	R	リードすると "0" が読めます。

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

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7OD	PE6OD	PE5OD	PE4OD	PE3OD	PE2OD	PE1OD	PE0OD
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7-0	PE7OD-PE0OD	R/W	0: CMOS 1: オープンドレイン

7.2.5.8 PEPUP (ポート E ブルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7UP	PE6UP	PE5UP	PE4UP	PE3UP	PE2UP	PE1UP	PE0UP
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7-0	PE7UP-PE0UP	R/W	ブルアップ 0: 禁止 1: 許可

## 7.2.5.9 PEPDN (ポート E プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7DN	PE6DN	PE5DN	PE4DN	PE3DN	PE2DN	PE1DN	PE0DN
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	PE7DN- PE0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.5.10 PEIE (ポート E 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PE7IE	PE6IE	PE5IE	PE4IE	PE3IE	PE2IE	PE1IE	PE0IE
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7-0	PE7IE-PE0IE	R/W	入力 0: 禁止 1: 許可

## 7.2.6 ポート F (PF0 ~ PF4)

ポート F はビット単位で入出力の指定ができる 5 ビットの汎用入出力ポートです。汎用入出力ポート機能以外にシリアルインタフェース機能 (SIO / UART)、16 ビットタイマ入出力機能、エンコーダ入力機能、動作モード設定機能があります。

リセット端子に "0" が入力されている期間、PF0 は入力とプルアップがイネーブルになっており、リセット信号の立ち上がりで PF0 が "1" の場合、シングルチップモードになって内蔵 Flash メモリから起動し、PF0 が "0" の場合、シングルブートモードとなって内蔵ブートプログラムから起動します。シングルブートモードの説明は、「Flash 動作説明」の章を参照してください。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

なお、リセット期間が終了すると PF0 の入力とプルアップは PFIE と PFPUP の初期値に従いディセーブルとなります。

PF0 以外の端子は、リセットにより汎用ポートとなり、入力・出力・プルダウンはディセーブル状態になります。

### 7.2.6.1 ポート F 回路タイプ

	7	6	5	4	3	2	1	0
Type	-	-	-	T11	T10	T15	T2	T20

### 7.2.6.2 ポート F レジスタ一覧

Base Address = 0x4000\_0140

レジスタ名		Address(Base+)
ポート F データ レジスタ	PFDATA	0x0000
ポート F 出力コントロールレジスタ	PF0CR	0x0004
ポート F ファンクションレジスタ 1	PFFR1	0x0008
ポート F ファンクションレジスタ 2	PFFR2	0x000C
ポート F ファンクションレジスタ 3	PFFR3	0x0010
ポート F オープンドレインコントロールレジスタ	PFOD	0x0028
ポート F プルアップコントロールレジスタ	PFPUP	0x002C
ポート F プルダウンコントロールレジスタ	PFPDN	0x0030
ポート F 入力コントロールレジスタ	PFIE	0x0038

## 7.2.6.3 PFDATA (ポートF データレジスタ)

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

Bit	Bit Symbol	TypF	機能
31-5	-	R	リードすると"0"が読めます。
4-0	PF4-PF0	R/W	ポートF データレジスタ

## 7.2.6.4 PFCR (ポートF 出力コントロールレジスタ)

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

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると"0"が読めます。
4-0	PF4C-PF0C	R/W	出力 0: 禁止 1: 許可

7.2.6.5 PFFR1 (ポート F ファンクションレジスタ 1)

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

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると "0" が読めます。
4	PF4F1	R/W	0: PORT 1: ENCZ1
3	PF3F1	R/W	0: PORT 1: ENCB1
2	PF2F1	R/W	0: PORT 1: ENCA1
1	PF1F1	R/W	0: PORT 1: TB7OUT
0	PF0F1	R/W	0: PORT 1: TB7IN (注)

注) TB7IN は PF0 と PN3 の兼用機能として割り当てられています。各ポートに設定する場合は以下のよう  
に設定してください。

TB7IN	<PF0F1>	<PN3F1>
PF0	1	0
PN3	0	1
<PF0F1> と <PN3F1> の両方に "1" を設定した場合には PF0 の入力 が有効となります。		



7.2.6.6 PFFR2 (ポート F ファンクションレジスタ 2)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	PF4F2	PF3F2	PF2F2	-	-
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	PF4F2	R/W	0: PORT 1: RXD3
3	PF3F2	R/W	0: PORT 1: TXD3
2	PF2F2	R/W	0: PORT 1: SCLK3
1-0	-	R	リードすると "0" が読めます。

7.2.6.7 PFFR3 (ポート F ファンクションレジスタ 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	-	-	-	-	-	PF2F3	-	-
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	PF2F3	R/W	0: PORT 1: CTS3
1-0	-	R	リードすると "0" が読めます。

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

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

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると "0" が読めます。
4-0	PF4OD- PF0OD	R/W	0: CMOS 1: オープンドレイン

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

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

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると "0" が読めます。
4-0	PF4UP-PF0UP	R/W	プルアップ 0: 禁止 1: 許可

## 7.2.6.10 PFPDN (ポート F プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	PF4DN	PF3DN	PF2DN	PF1DN	PF0DN
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	PF4DN-PF0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.6.11 PFIE (ポート F 入力コントロールレジスタ)

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

Bit	Bit Symbol	Type	機能
31-5	-	R	リードすると "0" が読めます。
4-0	PF4IE-PF0IE	R/W	入力 0: 禁止 1: 許可

## 7.2.7 ポート G (PG0 ~ PG7)

ポート G はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用入出力ポート機能以外に三相モータ制御のための入出力端子機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

### 7.2.7.1 ポート G 回路タイプ

	7	6	5	4	3	2	1	0
Type	T3	T3	T1	T1	T1	T1	T1	T1

### 7.2.7.2 ポート G レジスタ一覧

Base Address = 0x4000\_0180

レジスタ名		Address(Base+)
ポート G データ レジスタ	PGDATA	0x0000
ポート G 出力コントロールレジスタ	PGCR	0x0004
ポート G ファンクションレジスタ 1	PGFR1	0x0008
ポート G オープンドレインコントロールレジスタ	PGOD	0x0028
ポート G プルアップコントロールレジスタ	PGPUP	0x002C
ポート G プルダウンコントロールレジスタ	PGPDN	0x0030
ポート G 入力コントロールレジスタ	PGIE	0x0038

## 7.2.7.3 PGDATA (ポート G データレジスタ)

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

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

## 7.2.7.4 PGCR (ポート G 出力コントロールレジスタ)

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

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

7.2.7.5 PGFR1 (ポート G ファンクションレジスタ 1)

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

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	PG7F1	R/W	0: PORT 1: OV1
6	PG6F1	R/W	0: PORT 1: EMG1
5	PG5F1	R/W	0: PORT 1: ZO1
4	PG4F1	R/W	0: PORT 1: WO1
3	PG3F1	R/W	0: PORT 1: YO1
2	PG2F1	R/W	0: PORT 1: VO1
1	PG1F1	R/W	0: PORT 1: XO1
0	PG0F1	R/W	0: PORT 1: UO1

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

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

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

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

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

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

7.2.7.8 PGPDN (ポート G プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PG7DN	PG6DN	PG5DN	PG4DN	PG3DN	PG2DN	PG1DN	PG0DN
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	PG7DN- PG0DN	R/W	プルダウン 0: 禁止 1: 許可

7.2.7.9 PGIE (ポート G 入力コントロールレジスタ)

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

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



## 7.2.8 ポート H (PH0 ~ PH7)

ポート H はビット単位で入出力の指定ができる 8 ビットの汎用入出力ポートです。汎用入出力ポート機能以外に AD コンバータのアナログ入力端子機能、外部割込み入力機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

割り込み入力を STOP モード解除に使用する場合、PHFR1 で機能設定にし、PHIE で入力設定をしてください。この設定では、クロック / モード制御部の CGSTBYCR<DRVE> で STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

- 注 1) ポート H を、アナログ入力とそれ以外の機能で混在して使用した場合、AD 変換精度が悪化する場合があります。必ずご使用になるシステム上で問題のないことを確認してください。
- 注 2) STOP 以外のモードでは、PxIE で入力許可設定であれば PxFR の設定によらず割り込みの入力が可能になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。

### 7.2.8.1 ポート H 回路タイプ

	7	6	5	4	3	2	1	0
Type	T16	T16	T16	T16	T16	T17	T17	T17

### 7.2.8.2 ポート H レジスタ一覧

Base Address = 0x4000\_01C0

レジスタ名		Address(Base+)
ポート H データ レジスタ	PHDATA	0x0000
ポート H 出力コントロールレジスタ	PHCR	0x0004
ポート H ファンクションレジスタ 1	PHFR1	0x0008
ポート H オープンドレインコントロールレジスタ	PHOD	0x0028
ポート H プルアップコントロールレジスタ	PHPUP	0x002C
ポート H プルダウンコントロールレジスタ	PHPDN	0x0030
ポート H 入力コントロールレジスタ	PHIE	0x0038

### 7.2.8.3 PHDATA (ポートH データレジスタ)

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

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

### 7.2.8.4 PHCR (ポートH 出力コントロールレジスタ)

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

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

## 7.2.8.5 PHFR1 (ポートHファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	PH2F1	PH1F1	PH0F1
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	PH2F1	R/W	0: PORT 1: INT2
1	PH1F1	R/W	0: PORT 1: INT1
0	PH0F1	R/W	0: PORT 1: INT0

7.2.8.6 PHOD (ポートH オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PH7OD	PH6OD	PH5OD	PH4OD	PH3OD	PH2OD	PH1OD	PH0OD
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	PH7OD- PH0OD	R/W	0: CMOS 1: オープンドレイン

7.2.8.7 PHPUP (ポートH プルアップコントロールレジスタ)

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

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

## 7.2.8.8 PHPDN (ポート H プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PH7DN	PH6DN	PH5DN	PH4DN	PH3DN	PH2DN	PH1DN	PH0DN
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	PH7DN- PH0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.8.9 PHIE (ポート H 入力コントロールレジスタ)

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

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

## 7.2.9 ポート I (PI0 ~ PI3)

ポート I はビット単位で入出力の指定ができる 4 ビットの汎用入出力ポートです。汎用入出力ポート機能以外に AD コンバータのアナログ入力端子機能があります。

リセットによってすべてのビットは汎用ポートとなり、入力・出力・プルアップ・プルダウンはディセーブル状態になります。

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

### 7.2.9.1 ポート I 回路タイプ

	7	6	5	4	3	2	1	0
Type	-	-	-	-	T16	T16	T16	T16

### 7.2.9.2 ポート I レジスタ一覧

Base Address = 0x4000\_0200

レジスタ名		Address(Base+)
ポート I データ レジスタ	PIDATA	0x0000
ポート I 出力コントロールレジスタ	PICR	0x0004
ポート I オープンドレインコントロールレジスタ	PIOD	0x0028
ポート I プルアップコントロールレジスタ	PIPUP	0x002C
ポート I プルダウンコントロールレジスタ	PIPDN	0x0030
ポート I 入力コントロールレジスタ	PIIE	0x0038

## 7.2.9.3 PIDATA (ポートIデータレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PI3	PI2	PI1	PI0
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	PI3-PI0	R/W	ポートIデータレジスタ

## 7.2.9.4 PICR (ポートI出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PI3C	PI2C	PI1C	PI0C
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	PI3C-PI0C	R/W	出力 0: 禁止 1: 許可

7.2.9.5 PIOD (ポートIオープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PI3OD	PI2OD	PI1OD	PI0OD
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	PI3OD-PI0OD	R/W	0: CMOS 1: オープンドレイン

7.2.9.6 PIPUP (ポートIプルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PI3UP	PI2UP	PI1UP	PI0UP
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	PI3UP-PI0UP	R/W	プルアップ 0: 禁止 1: 許可



## 7.2.9.7 PIPDN (ポートIプルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PI3DN	PI2DN	PI1DN	PI0DN
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	PI3DN-PI0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.9.8 PIIIE (ポートI入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PI3IE	PI2IE	PI1IE	PI0IE
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	PI3IE-PI0IE	R/W	入力 0: 禁止 1: 許可

### 7.2.10 ポート J (PJ0 ~ PJ7)

ポート J はビット単位で入出力の指定ができる 8 ビットの汎用入出力です。汎用入出力ポート機能以外に AD コンバータのアナログ入力端子機能、外部割り込み入力機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

割り込み入力を STOP モード解除に使用する場合、PJFR1 で機能設定にし、PJIE で入力設定をしてください。この設定では、クロック / モード制御部の CGSTBYCR<DRVE> で STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

- 注 1) ポート J を、アナログ入力とそれ以外の機能で混在して使用した場合、AD 変換精度が悪化する場合があります。必ずご使用になるシステム上で問題のないことを確認してください。
- 注 2) STOP 以外のモードでは、PxIE で入力許可設定であれば PxFR の設定によらず割り込みの入力が可能になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。

#### 7.2.10.1 ポート J 回路タイプ

	7	6	5	4	3	2	1	0
Type	T17	T17	T16	T16	T16	T16	T16	T16

#### 7.2.10.2 ポート J レジスタ一覧

Base Address = 0x4000\_0240

レジスタ名		Address(Base+)
ポート J データレジスタ	PJDATA	0x0000
ポート J 出力コントロールレジスタ	PJCR	0x0004
ポート J ファンクションレジスタ 1	PJFR1	0x0008
ポート J オープンドレインコントロールレジスタ	PJOD	0x0028
ポート J プルアップコントロールレジスタ	PJPUP	0x002C
ポート J プルダウンコントロールレジスタ	PJPDN	0x0030
ポート J 入力コントロールレジスタ	PJIE	0x0038

## 7.2.10.3 PJDATA (ポート J データレジスタ)

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

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

## 7.2.10.4 PJCR (ポート J 出力コントロールレジスタ)

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

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

7.2.10.5 PJFR1 (ポートJファンクションレジスタ 1)

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

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると"0"が読めます。
7	PJ7F1	R/W	0: PORT 1: INTD
6	PJ6F1	R/W	0: PORT 1: INTC
5-0	-	R	リードすると"0"が読めます。

7.2.10.6 PJOD (ポートJオープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PJ7OD	PJ6OD	PJ5OD	PJ4OD	PJ3OD	PJ2OD	PJ1OD	PJ0OD
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	PJ7OD-PJ0OD	R/W	0: CMOS 1: オープンドレイン

## 7.2.10.7 PJPUP (ポートJプルアップコントロールレジスタ)

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

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

## 7.2.10.8 PJPDN (ポートJプルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	PJ7DN	PJ6DN	PJ5DN	PJ4DN	PJ3DN	PJ2DN	PJ1DN	PJ0DN
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	PJ7DN-PJ0DN	R/W	プルダウン 0: 禁止 1: 許可

7.2.10.9 PJIE (ポートJ入力コントロールレジスタ)

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

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

### 7.2.11 ポート K (PK0 ~ PK1)

ポート K はビット単位で入出力の指定ができる 2 ビットの汎用入出力ポートです。汎用入出力ポート機能以外に AD コンバータのアナログ入力端子機能、外部割り込み入力機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

割り込み入力を STOP モード解除に使用する場合、PKFR1 で機能設定にし、PKIE で入力設定をしてください。この設定では、クロック / モード制御部の CGSTBYCR<DRVE> で STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

- 注 1) ポート K を、アナログ入力とそれ以外の機能で混在して使用した場合、AD 変換精度が悪化する場合があります。必ずご使用になるシステム上で問題のないことを確認してください。
- 注 2) STOP 以外のモードでは、PxIE で入力許可設定であれば PxFR の設定によらず割り込みの入力が可能になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。

#### 7.2.11.1 ポート K 回路タイプ

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

#### 7.2.11.2 ポート K レジスタ一覧

Base Address = 0x4000 \_ 0280

レジスタ名		Address(Base+)
ポート K データレジスタ	PKDATA	0x0000
ポート K 出力コントロールレジスタ	PKCR	0x0004
ポート K ファンクションレジスタ 1	PKFR1	0x0008
ポート K オープンドレインコントロールレジスタ	PKOD	0x0028
ポート K プルアップコントロールレジスタ	PKPUP	0x002C
ポート K プルダウンコントロールレジスタ	PKPDN	0x0030
ポート K 入力コントロールレジスタ	PKIE	0x0038

### 7.2.11.3 PKDATA (ポートK データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PK1	PK0
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	PK1-PK0	R/W	ポートK データレジスタ

### 7.2.11.4 PKCR (ポートK 出力コントロールレジスタ)

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



## 7.2.11.5 PKFR1 (ポートKファンクションレジスタ 1)

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

## 7.2.11.6 PKOD (ポートKオープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PK1OD	PK0OD
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	PK1OD- PK0OD	R/W	0: CMOS 1: オープンドレイン

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

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PK1UP	PK0UP
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	PK1UP-PK0UP	R/W	プルアップ 0: 禁止 1: 許可

## 7.2.11.8 PKPDN (ポート K プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PK1DN	PK0DN
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	PK1DN-PK0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.11.9 PKIE (ポート K 入力コントロールレジスタ)

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

### 7.2.12 ポート L (PL0 ~ PL1)

ポート L はビット単位で入力の指定ができる 2 ビットの汎用入力ポートです。汎用入力ポート機能以外に外部割り込み入力機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力は禁止となります。

割り込み入力を STOP モード解除に使用する場合、PLFR1 で機能設定にし、PLIE で入力設定をしてください。この設定では、クロック / モード制御部の CGSTBYCR<DRVE> で STOP モード中端子をドライブしない設定を行っていても割り込み入力可能です。

- 注 1) STOP 以外のモードでは、PxIE で入力許可設定であれば PxFR の設定によらず割り込みの入力が可能になります。割り込みの設定を行う際に、未使用の割り込みをイネーブルにしないようご注意ください。
- 注 2) ポート L は電源投入時、リセット期間中を含め一定時間 "Low" レベルに保持してください。詳細は『電気的特性』の『電源投入時のご注意』を参照してください。

#### 7.2.12.1 ポート L 回路タイプ

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

#### 7.2.12.2 ポート L レジスタ一覧

Base Address = 0x4000\_02C0

レジスタ名		Address(Base+)
ポート L データ レジスタ	PLDATA	0x0000
ポート L ファンクションレジスタ 1	PLFR1	0x0008
ポート L 入力コントロールレジスタ	PLIE	0x0038

7.2.12.3 PLDATA (ポートL データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PL1	PL0
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	PL1-PL0	R/W	ポートL データレジスタ

7.2.12.4 PLFR1 (ポートL ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PL1F1	PL0F1
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	PL1F1	R/W	0: PORT 1: INTA
0	PL0F1	R/W	0: PORT 1: INTB

7.2.12.5 PLIE (ポートL入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PL1IE	PL0IE
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	PL1IE-PL0IE	R/W	入力 0: 禁止 1: 許可

### 7.2.13 ポート M (PM0~PM1)

ポート M はビット単位で入出力の指定ができる 2 ビットの汎用入出力ポートです。汎用入出力ポート機能以外に CGOSCCR<HOSCON>=1 で外部高速発振子接続用端子 (X1,X2) になります。

CGOSCCR<HOSCON>=1 となっている間はポート M の各レジスタの書換えはできません。外部高速発振子接続端子として使用する場合は「システムクロック」の章をご参照ください(注 1)。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。(注 2)

- 注 1) PMDATA、PMOD を除く、ポート M のレジスタのいずれかが "0" でない場合、外部高速発振子接続端子への切り替え (CGOSCCR<HOSCON>=1) ができません。
- 注 2) リセット解除後に選択される高速クロックは内蔵高速クロックです。従って、初期状態ではポート M になります。

#### 7.2.13.1 ポート M 回路タイプ

	7	6	5	4	3	2	1	0
Type	-	-	-	-	-	-	T21	T21

#### 7.2.13.2 ポート M レジスタ一覧

Base Address = 0x4000\_0300

レジスタ名		Address(Base+)
ポート M データ レジスタ	PMDATA	0x0000
ポート M 出力コントロールレジスタ	PMCR	0x0004
ポート M オープンドレインコントロールレジスタ	PMOD	0x0028
ポート M プルアップコントロールレジスタ	PMPUP	0x002C
ポート M プルダウンコントロールレジスタ	PMPDN	0x0030
ポート M 入力コントロールレジスタ	PMIE	0x0038

### 7.2.13.3 PMDATA (ポート M データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PM1	PM0
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	PM1-PM0	R/W	ポート M データレジスタ

### 7.2.13.4 PMCR (ポート M 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PM1C	PM0C
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	PM1C-PM0C	R/W	出力 0: 禁止 1: 許可



## 7.2.13.5 PMOD (ポート M オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PM1OD	PM0OD
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	PM1OD- PM0OD	R/W	0: CMOS 1: オープンドレイン

7.2.13.6 PMPUP (ポート M プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PM1UP	PM0UP
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	PM1UP- PM0UP	R/W	プルアップ 0: 禁止 1: 許可

7.2.13.7 PMPDN (ポート M プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PM1DN	PM0DN
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	PM1DN- PM0DN	R/W	プルダウン 0: 禁止 1: 許可

## 7.2.13.8 PMIE (ポート M 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	PM1IE	PM0IE
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	PM1IE-PM0IE	R/W	入力 0: 禁止 1: 許可

### 7.2.14 ポート N (PN0~PN3)

ポートNはビット単位で入出力の指定ができる4ビットの汎用入出力ポートです。汎用入出力ポート機能以外にシリアルバスインタフェース機能 (I2C / SIO)、16 ビットタイマ入力機能があります。

リセット後ファンクションレジスタは汎用ポートとなり、入力・出力・プルアップ・プルダウンは禁止となります。

#### 7.2.14.1 ポート N 回路タイプ

	7	6	5	4	3	2	1	0
Type	-	-	-	-	T3	T22	T22	T22

#### 7.2.14.2 ポート N レジスタ一覧

Base Address = 0x4000\_0340

レジスタ名		Address(Base+)
ポート N データ レジスタ	PNDATA	0x0000
ポート N 出力コントロールレジスタ	PNCR	0x0004
ポート N ファンクションレジスタ 1	PNFR1	0x0008
ポート N オープンドレインコントロールレジスタ	PNOD	0x0028
ポート N プルアップコントロールレジスタ	PNPUP	0x002C
ポート N プルダウンコントロールレジスタ	PNPDN	0x0030
ポート N 入力コントロールレジスタ	PNIE	0x0038

## 7.2.14.3 PNDATA (ポート N データレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PN3	PN2	PN1	PN0
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	PN3-PN0	R/W	ポート N データレジスタ

## 7.2.14.4 PNCR (ポート N 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PN3C	PN2C	PN1C	PN0C
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	PN3C-PN0C	R/W	出力 0: 禁止 1: 許可

7.2.14.5 PNFR1 (ポートN ファンクションレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PN3F1	PN2F1	PN1F1	PN0F1
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	PN3F1	R/W	0: PORT 1: TB7IN (注)
2	PN2F1	R/W	0: PORT 1: SCK
1	PN1F1	R/W	0: PORT 1: SI / SCL
0	PN0F1	R/W	0: PORT 1: SO / SDA

注) TB7IN を選択する場合 (<PN3F1> =1 に設定) は、<PN0F1> に 0 を設定してください。ただし、<PN3F1> と <PN0F1> の両方に "1" を設定した場合には PF0 の入力の有効となります。

## 7.2.14.6 PNOD (ポート N オープンドレインコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PN3OD	PN2OD	PN1OD	PN0OD
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	PN3OD- PN0OD	R/W	0: CMOS 1: オープンドレイン

## 7.2.14.7 PNPUP (ポート N プルアップコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PN3UP	PN2UP	PN1UP	PN0UP
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	PN3UP- PN0UP	R/W	プルアップ 0: 禁止 1: 許可

7.2.14.8 PNPDN (ポート N プルダウンコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PN3DN	PN2DN	PN1DN	PN0DN
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	PN3DN- PN0DN	R/W	プルダウン 0: 禁止 1: 許可

7.2.14.9 PNIE (ポート N 入力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	PN3IE	PN2IE	PN1IE	PN0IE
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	PN3IE-PN0IE	R/W	入力 0: 禁止 1: 許可



## 7.3 ポート回路図

### 7.3.1 ポートタイプ一覧

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

図中の点線は「ポート部等価回路図」で記されている等価回路の範囲を示します。

表 7-3 機能一覧

Type	汎用ポート	機能 1	機能 2	機能 3	アナログ	Pull-up	Pull-dn	プログラマブルオープンドレイン	備考
T1	入出力	出力	-	-	-	R	R	o	機能出力にイネーブル付き
T2	入出力	出力	-	-	-	R	R	o	
T3	入出力	入力	-	-	-	R	R	o	
T4	入出力	入力 (int)	-	-	-	R	R	o	
T5	入力	入力 (int)	-	-	-	-	-	-	
T6	入出力	入出力	-	-	-	NoR	-	-	機能出力にイネーブル付き
T7	入出力	入力	-	-	-	NoR	-	-	
T8	入出力	入力	-	-	-	-	NoR	-	
T9	入出力	入出力	入力	-	-	R	R	o	
T10	入出力	入力	出力	-	-	R	R	o	
T11	入出力	入力	入力	-	-	R	R	o	
T12	入出力	入力	入力 (int)	-	-	R	R	o	
T13	入出力	出力	出力	-	-	R	R	o	
T14	入出力	出力	入出力	-	-	R	R	o	
T15	入出力	入力	入出力	入力	-	R	R	o	
T16	入出力	-	-	-	o	R	R	o	
T17	入出力	入力 (int)	-	-	o	R	R	o	
T18	入出力	出力	-	-	-	R	-	-	
T19	入出力	出力	-	-	-	NoR	-	-	機能出力にイネーブル付き
T20	入出力	入力	-	-	-	NoR	NoR	o	リセット中 $\overline{\text{BOOT}}$ 入力許可
T21	入出力	-(OSC1)	-	-	-	R	R	o	外部発振器
T22	入出力	入出力	-	-	-	R	R	o	

int : 割り込み入力

- : なし

o : 有り

R: リセット中は強制的に禁止

NoR: リセットでは制御されない

7.3.2 タイプ T1

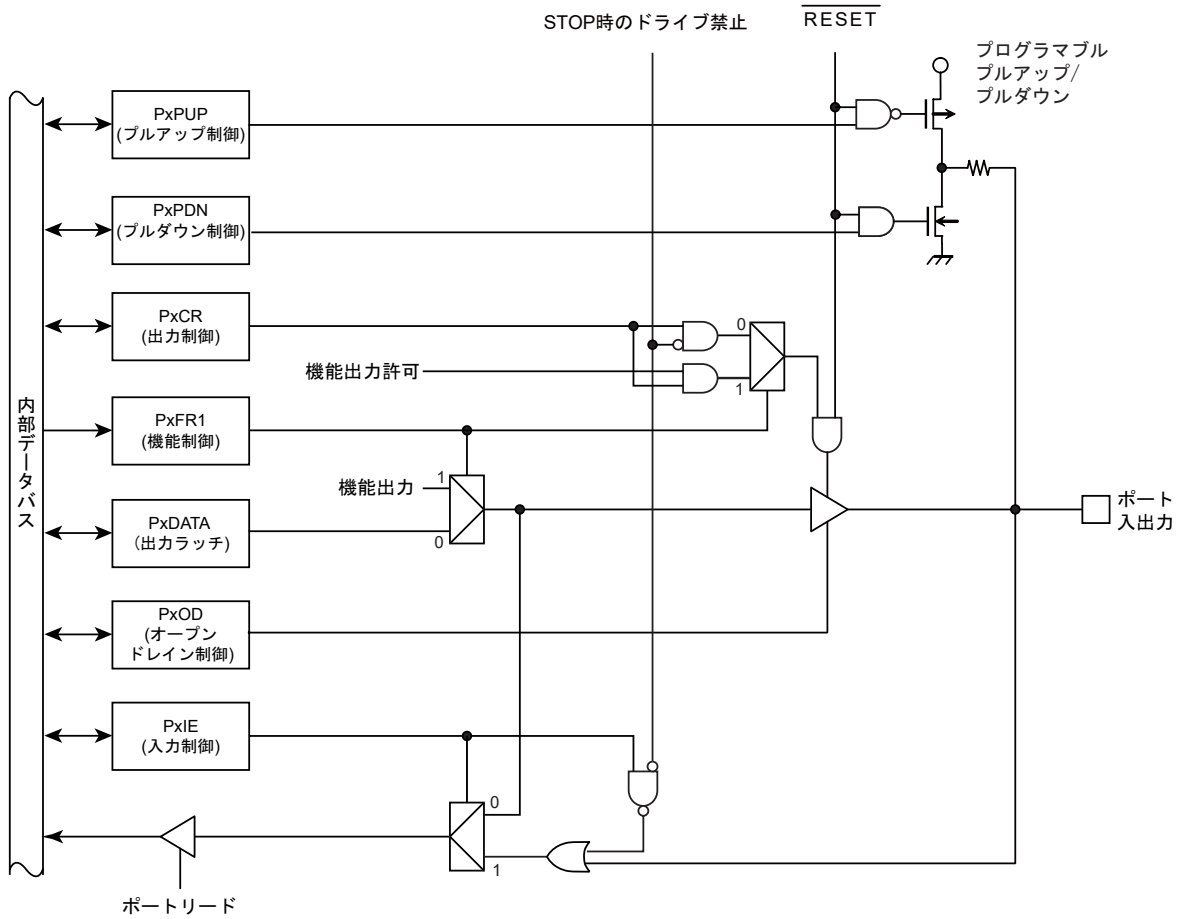


図 7-1 ポートタイプ T1

7.3.3 タイプ T2

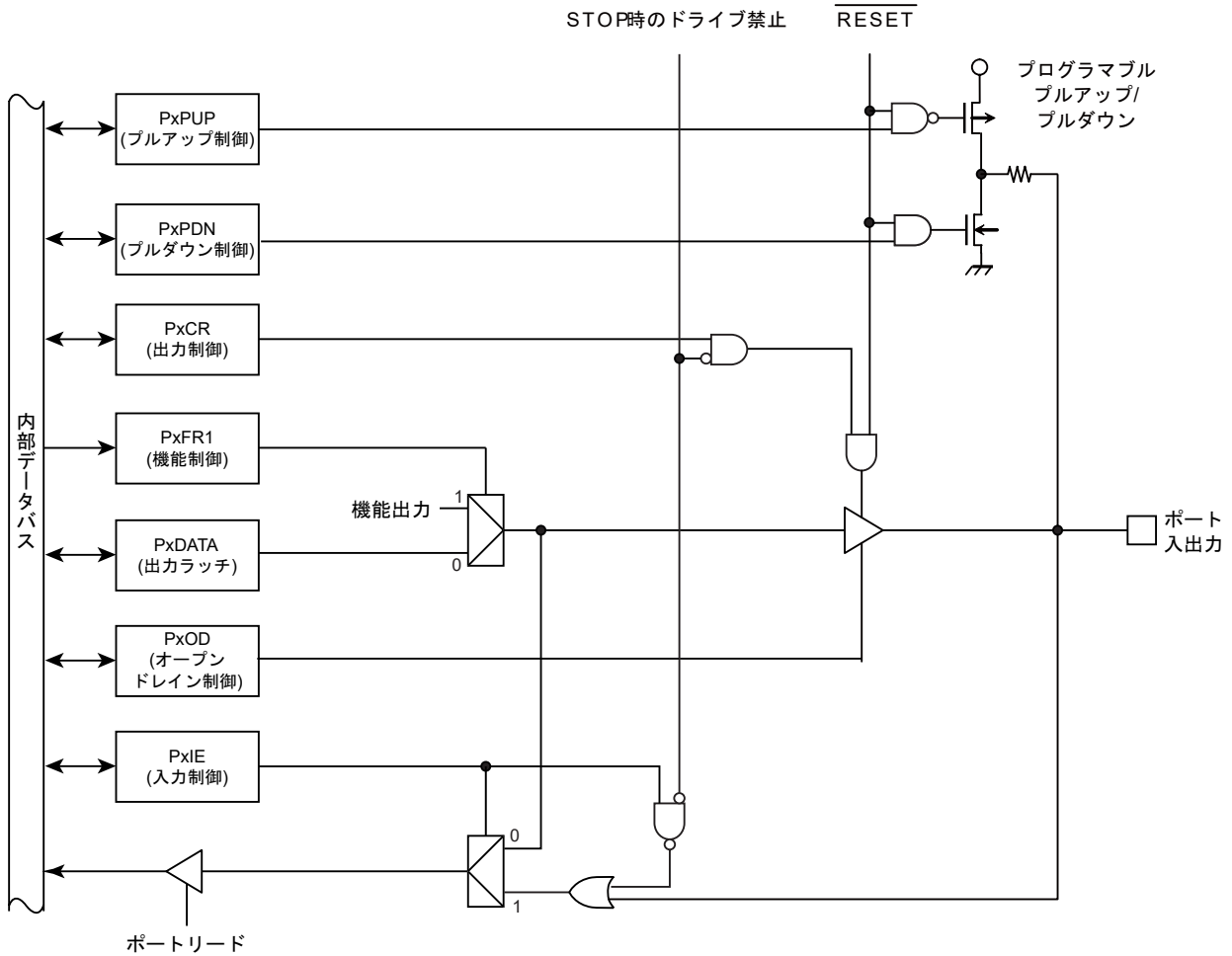


図 7-2 ポートタイプ T2

7.3.4 タイプ T3

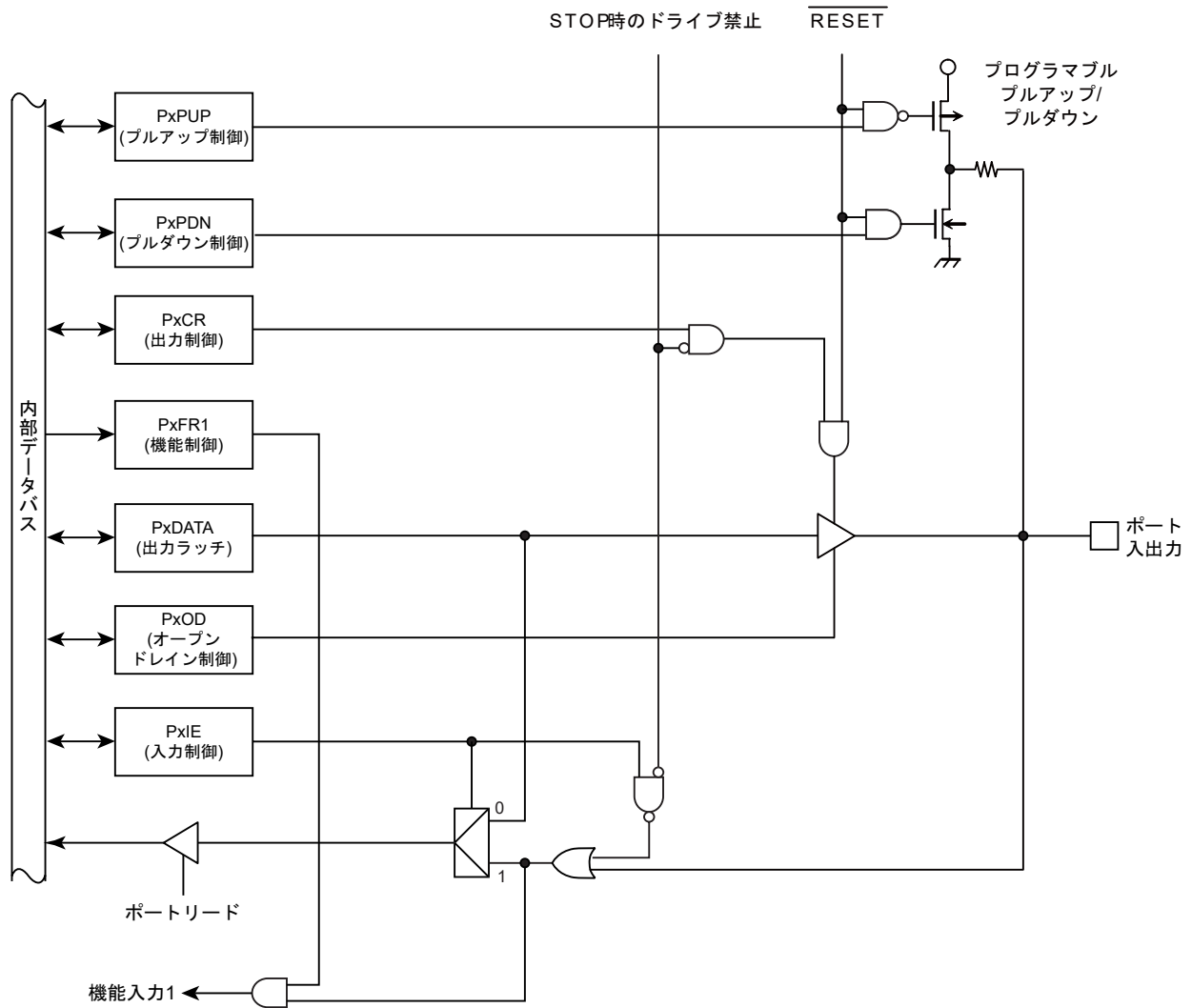


図 7-3 ポートタイプ T3

7.3.5 タイプ T4

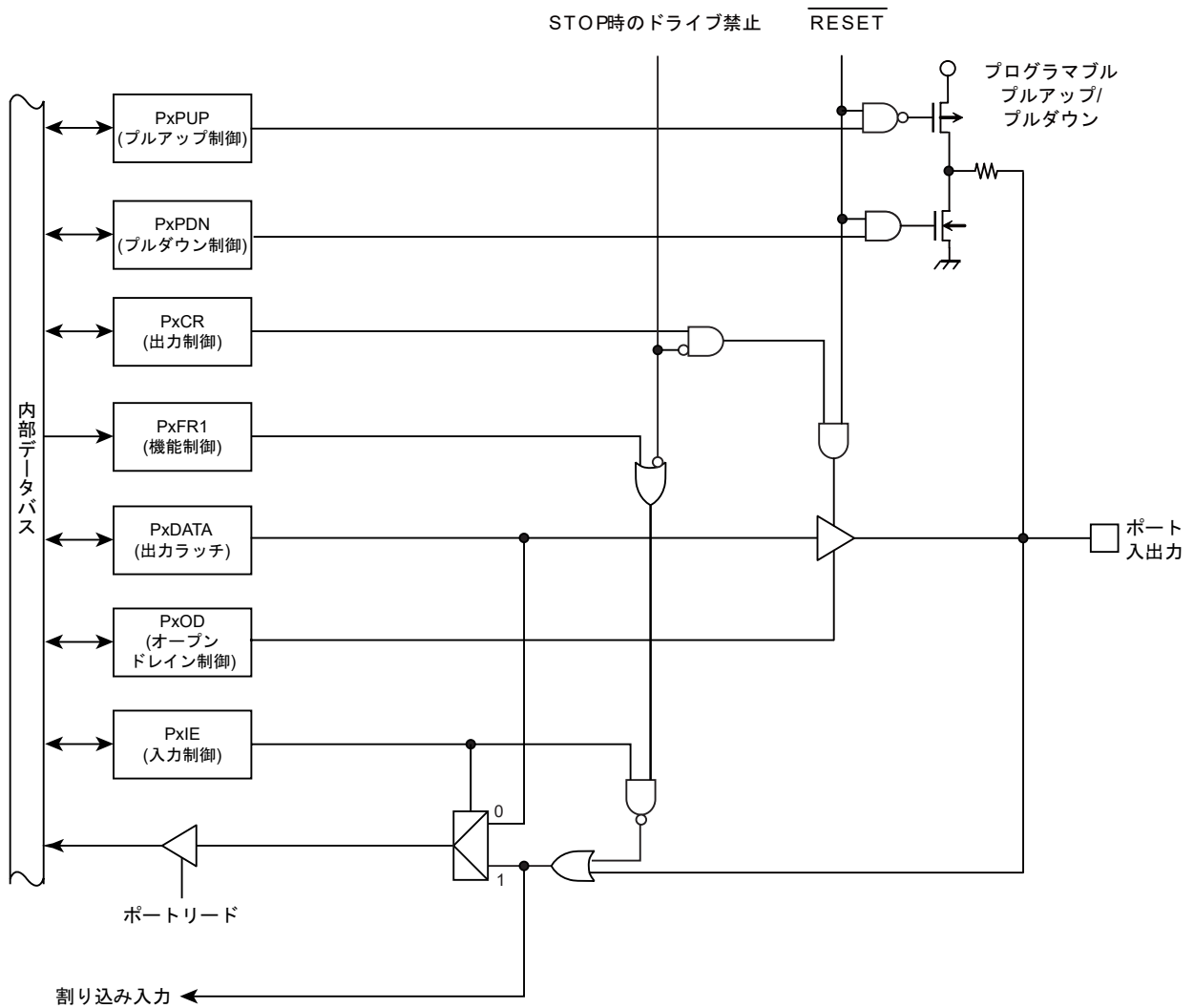


図 7-4 ポートタイプ T4

7.3.6 タイプ T5

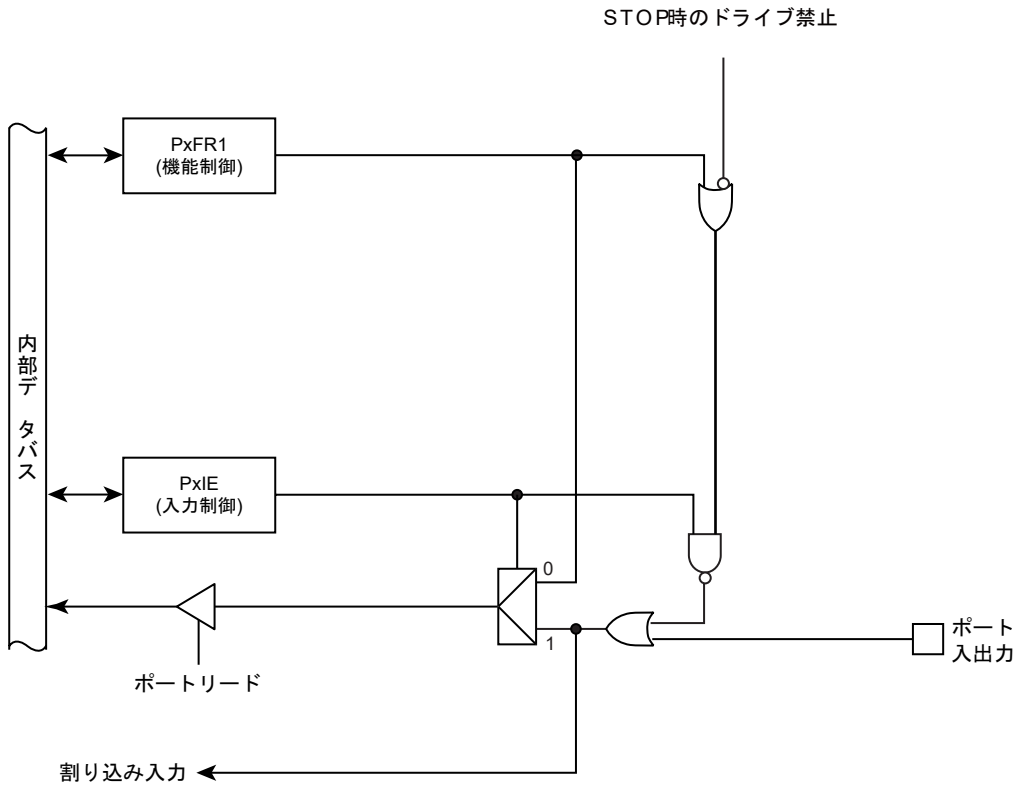


図 7-5 ポートタイプ T5

7.3.7 タイプ T6

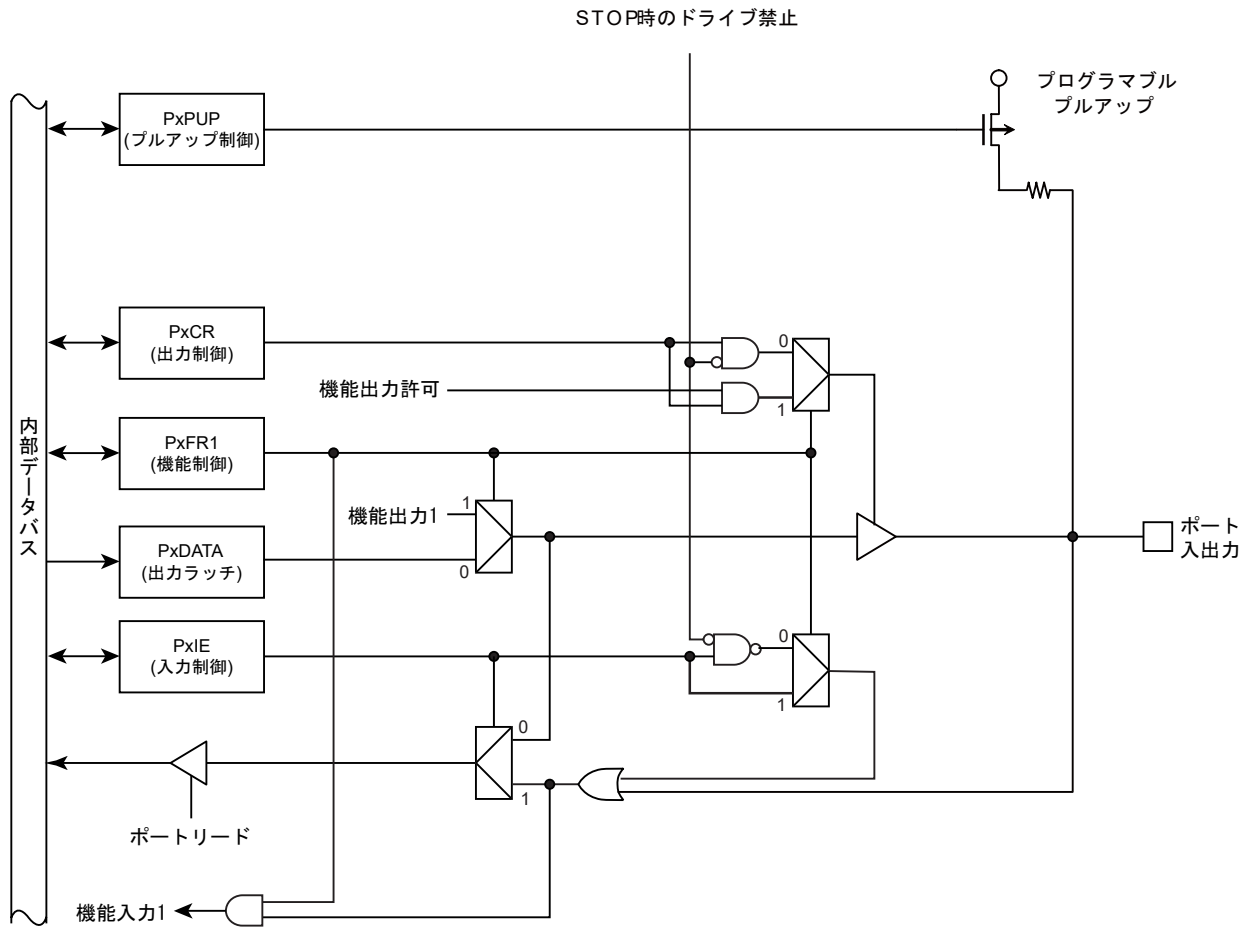


図 7-6 ポートタイプ T6

### 7.3.8 タイプ T7

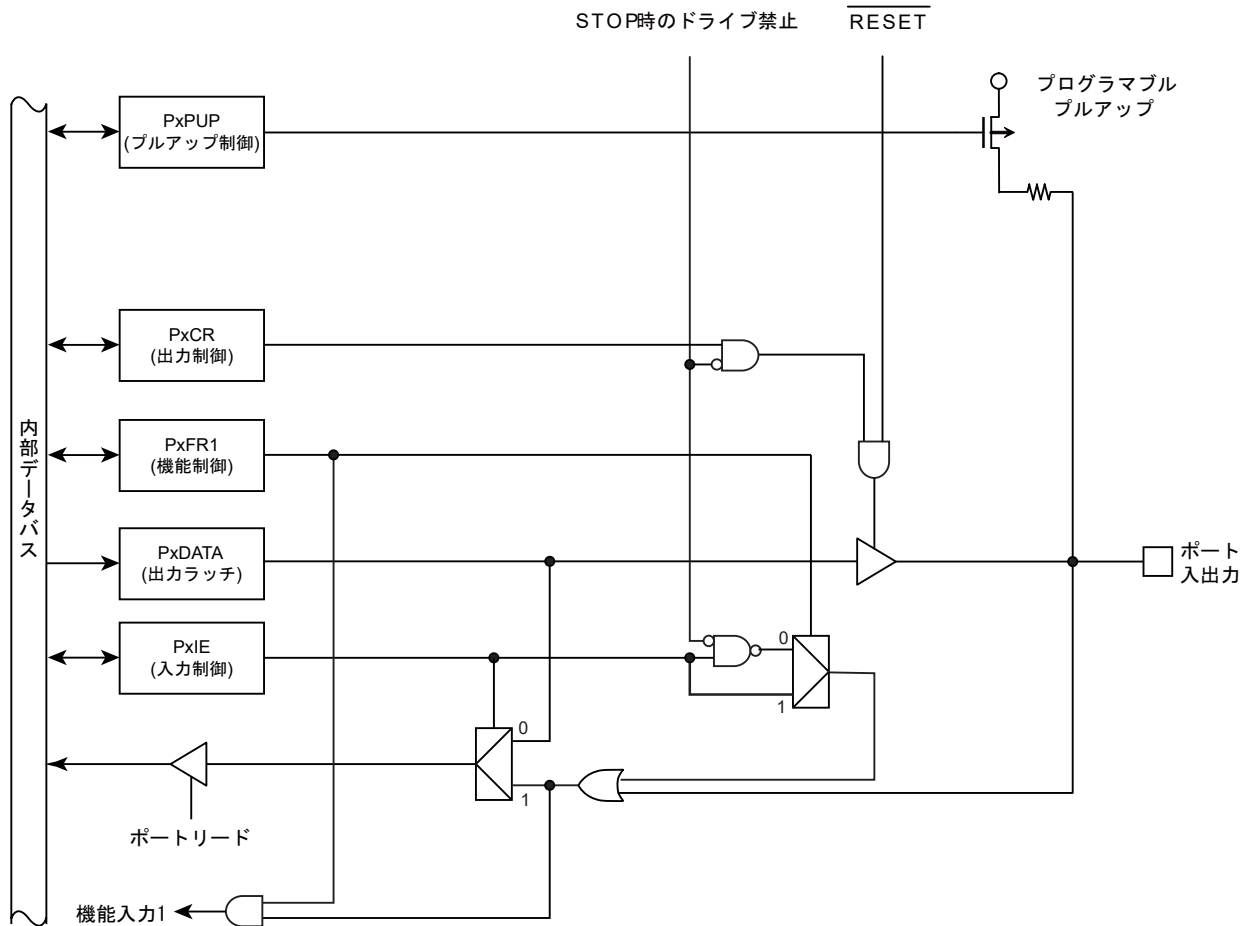


図 7-7 ポートタイプ T7

注)  $\overline{\text{TRST}}$ には、ノイズフィルタ(30ns Typ.)が入ります。



7.3.9 タイプ T8

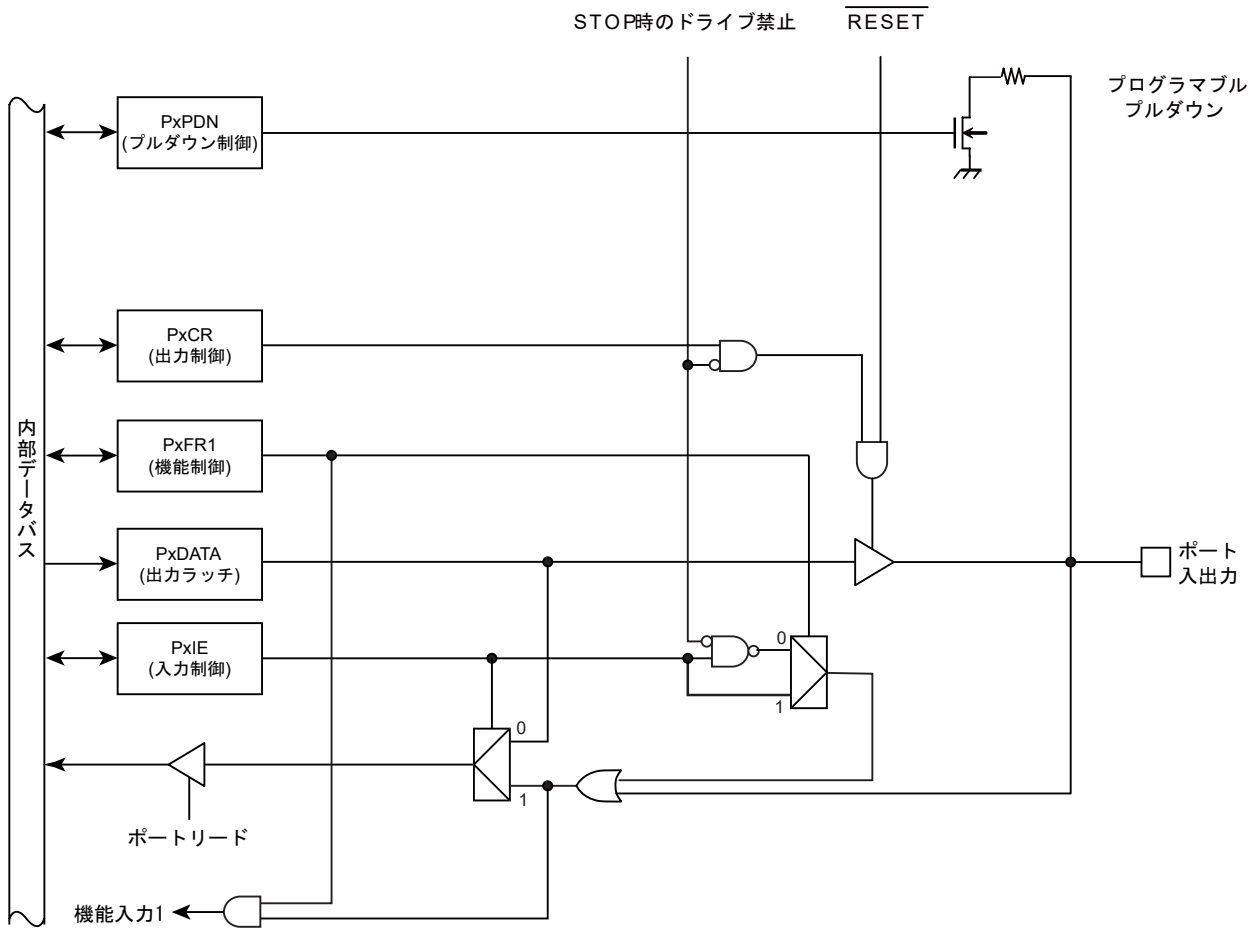


図 7-8 ポートタイプ T8

7.3.10 タイプ T9

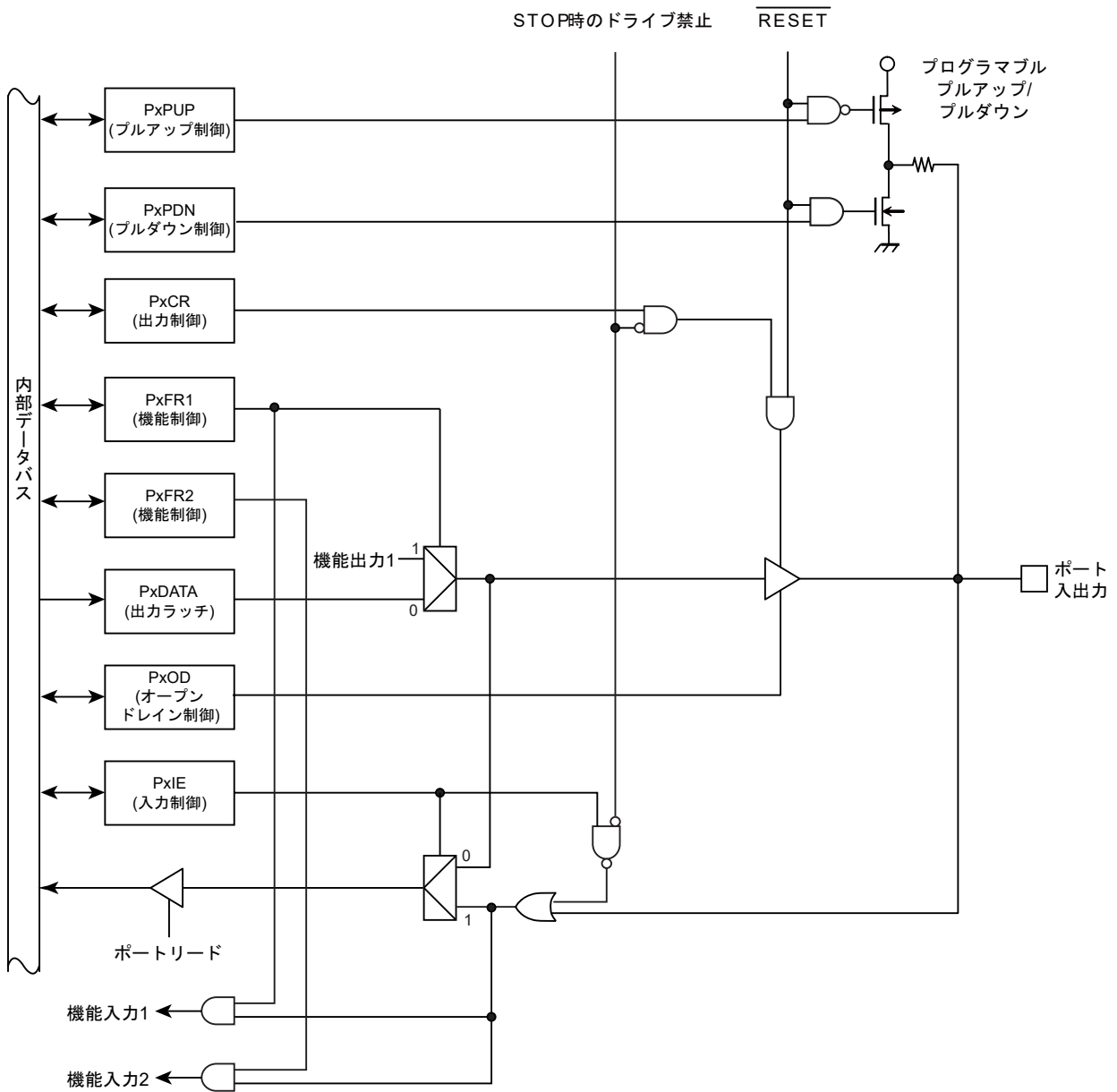


図 7-9 ポートタイプ T9

7.3.11 タイプ T10

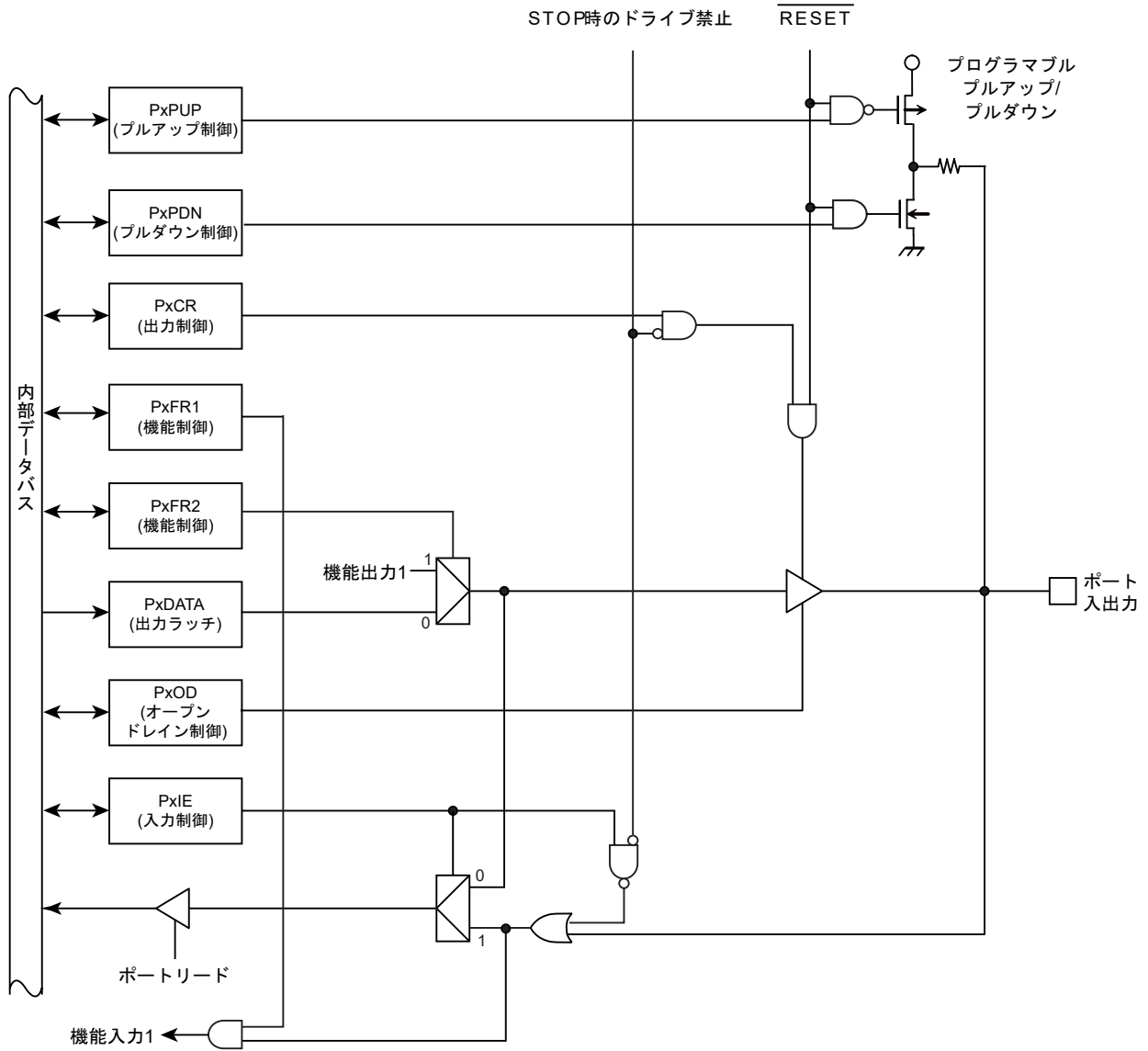


図 7-10 ポートタイプ T10

7.3.12 タイプ T11

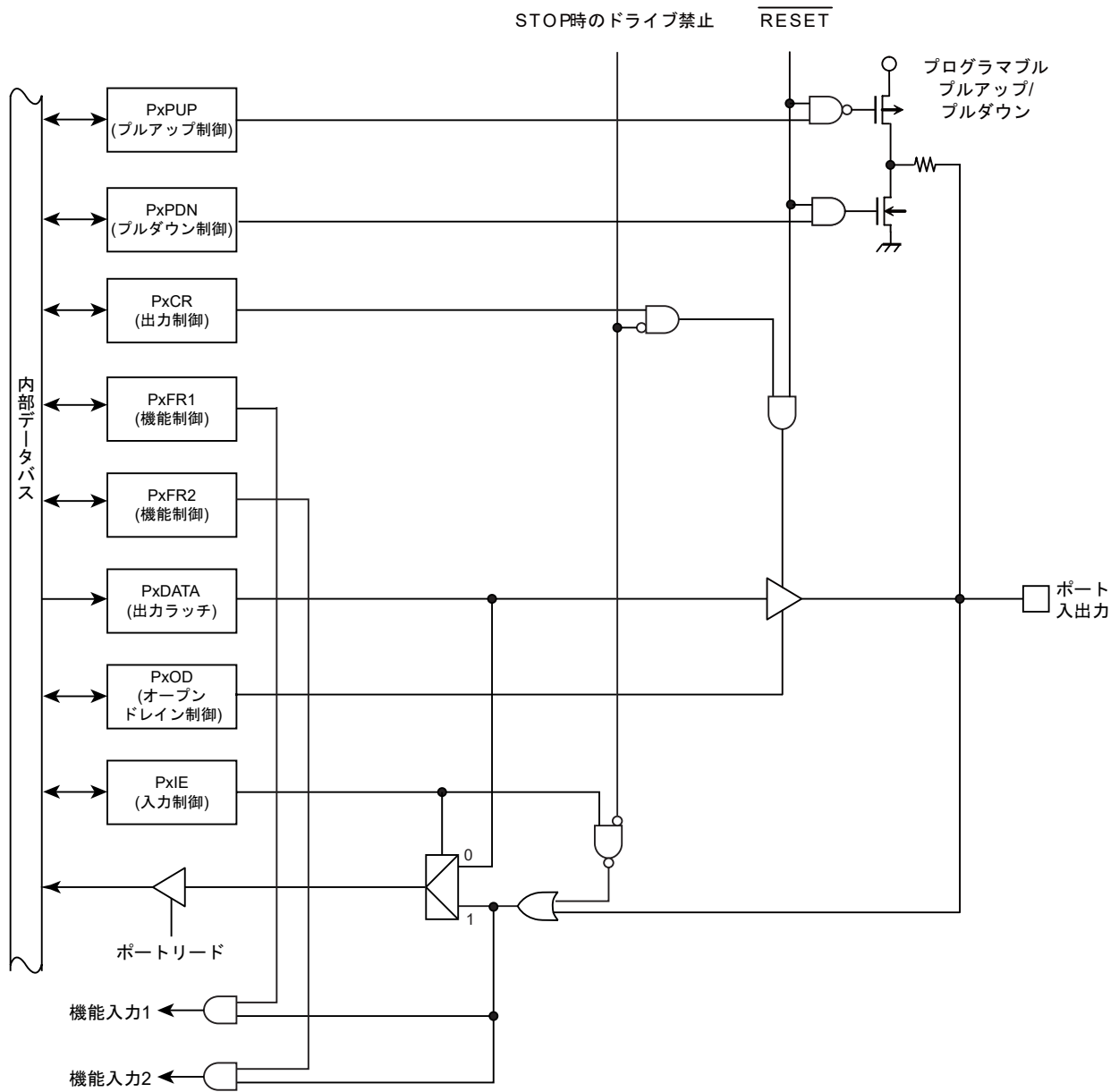


図 7-11 ポートタイプ T11

7.3.13 タイプ T12

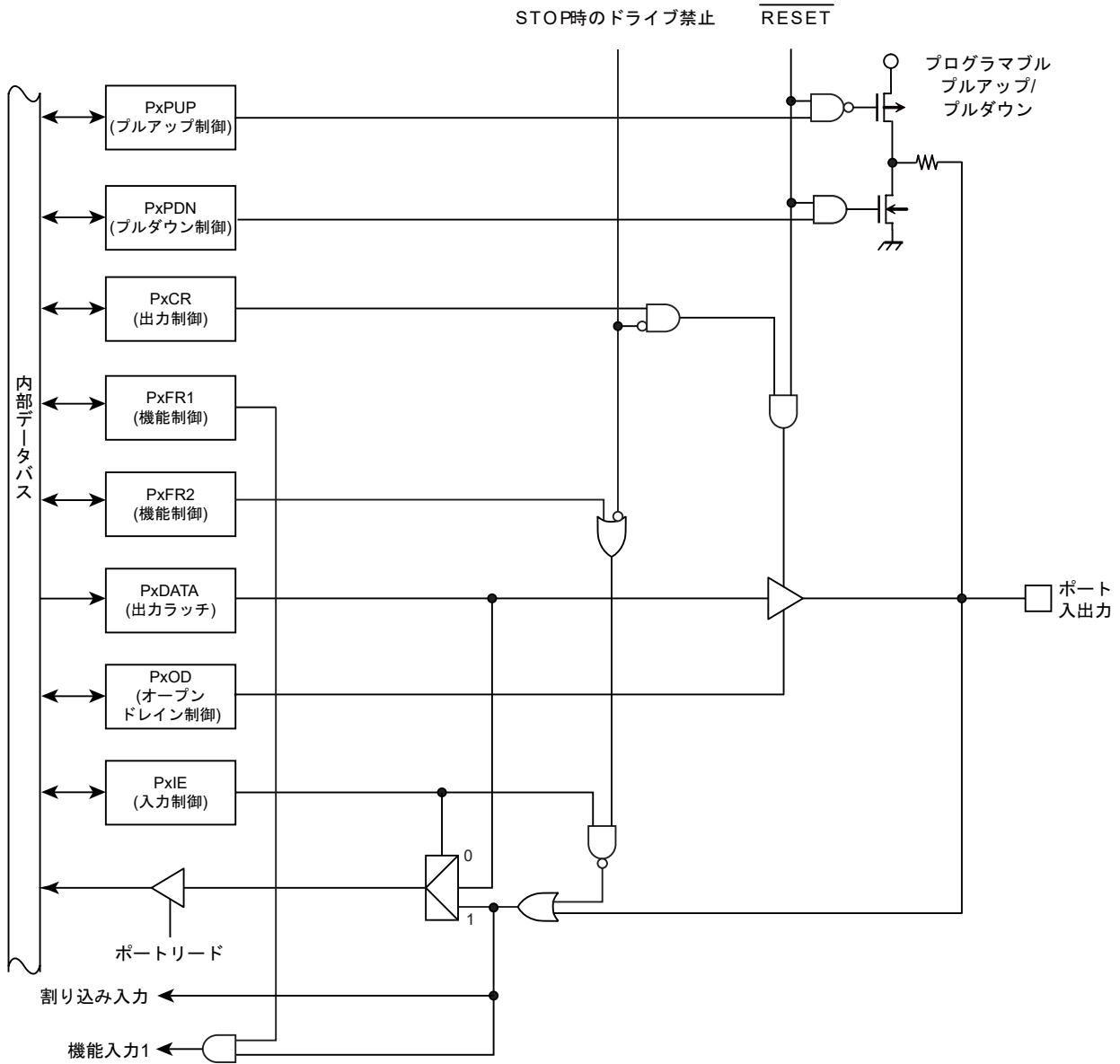


図 7-12 ポートタイプ T12

7.3.14 タイプ T13

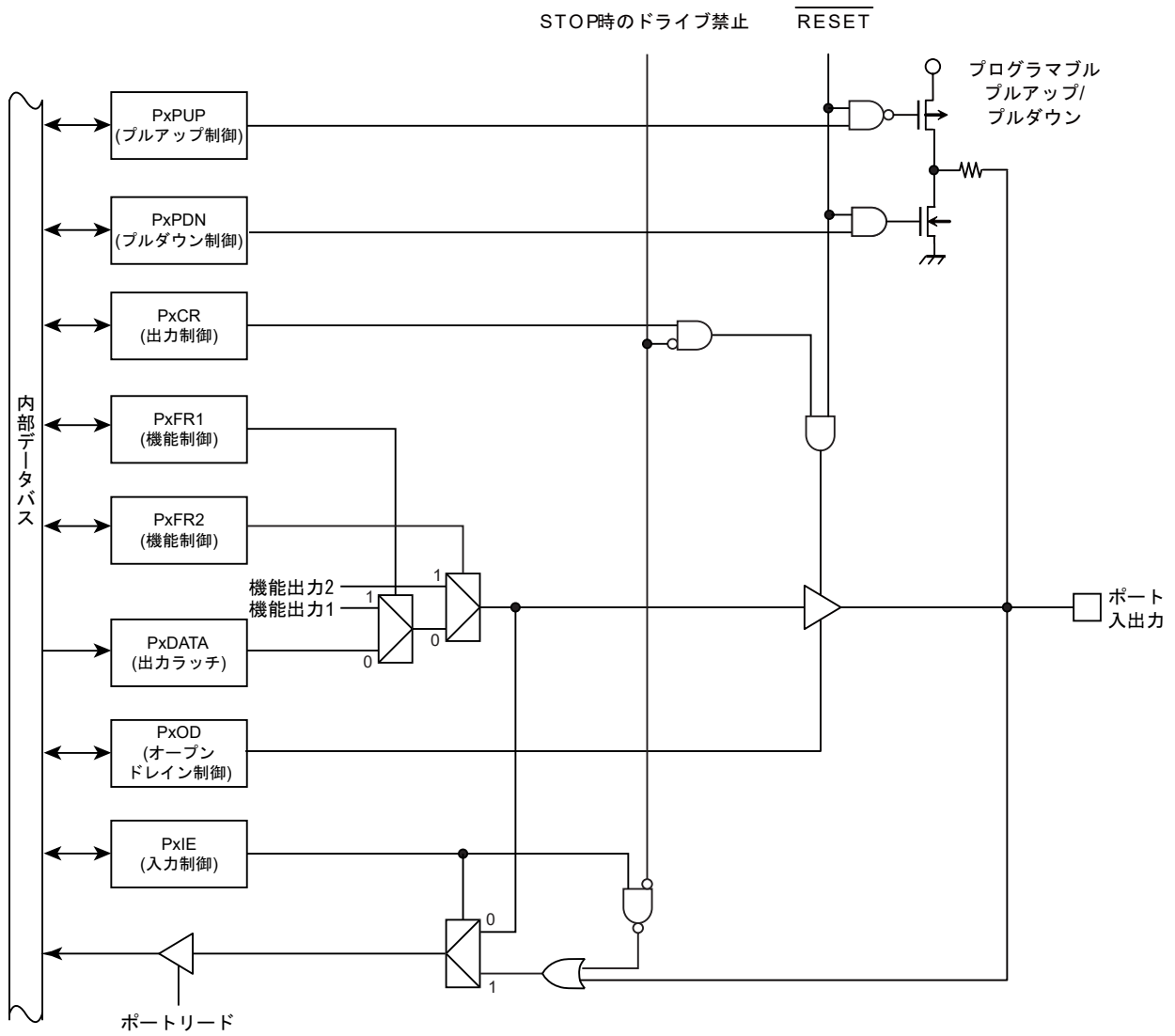


図 7-13 ポートタイプ T13

7.3.15 タイプ T14

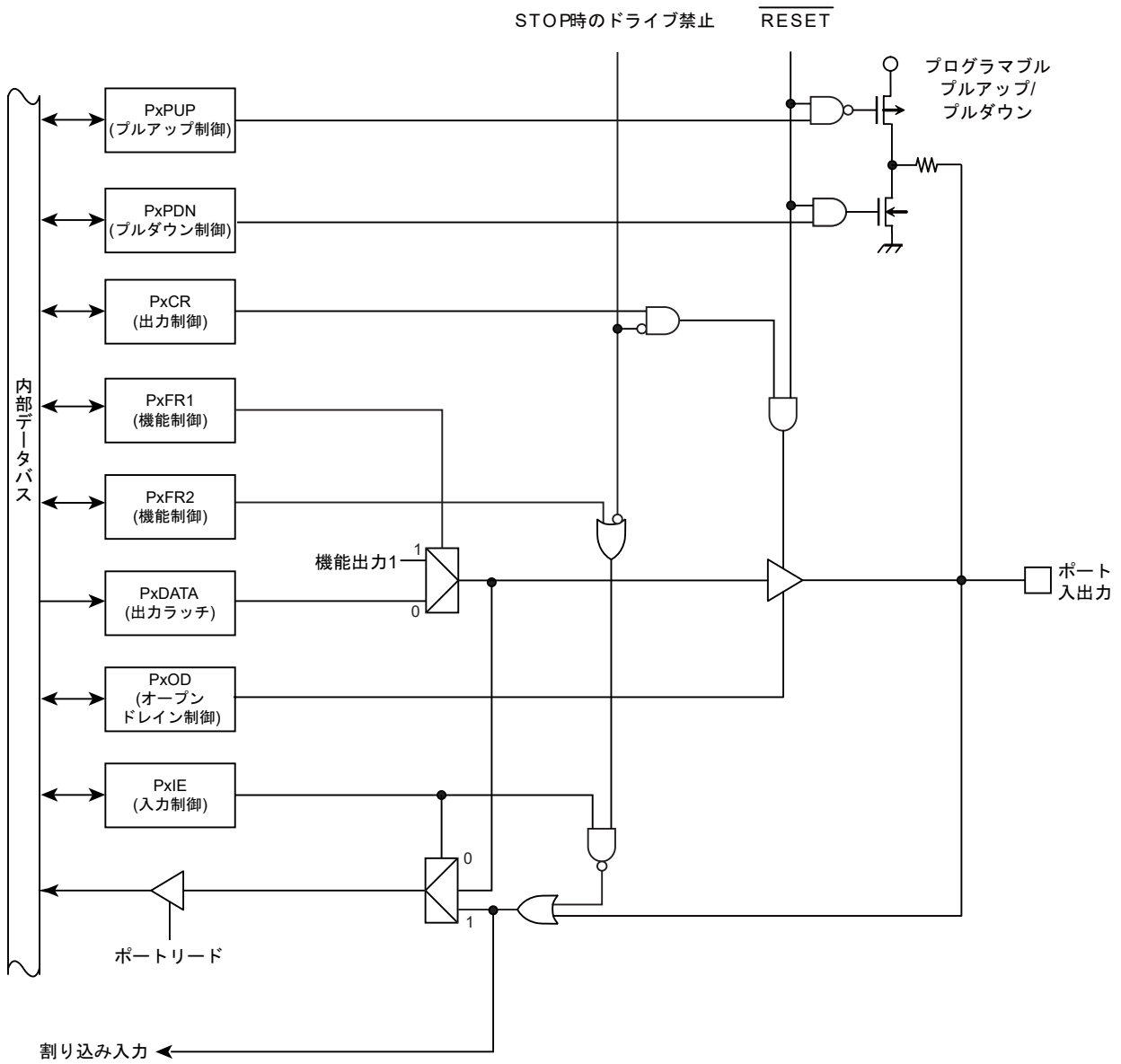


図 7-14 ポートタイプ T14

7.3.16 タイプ T15

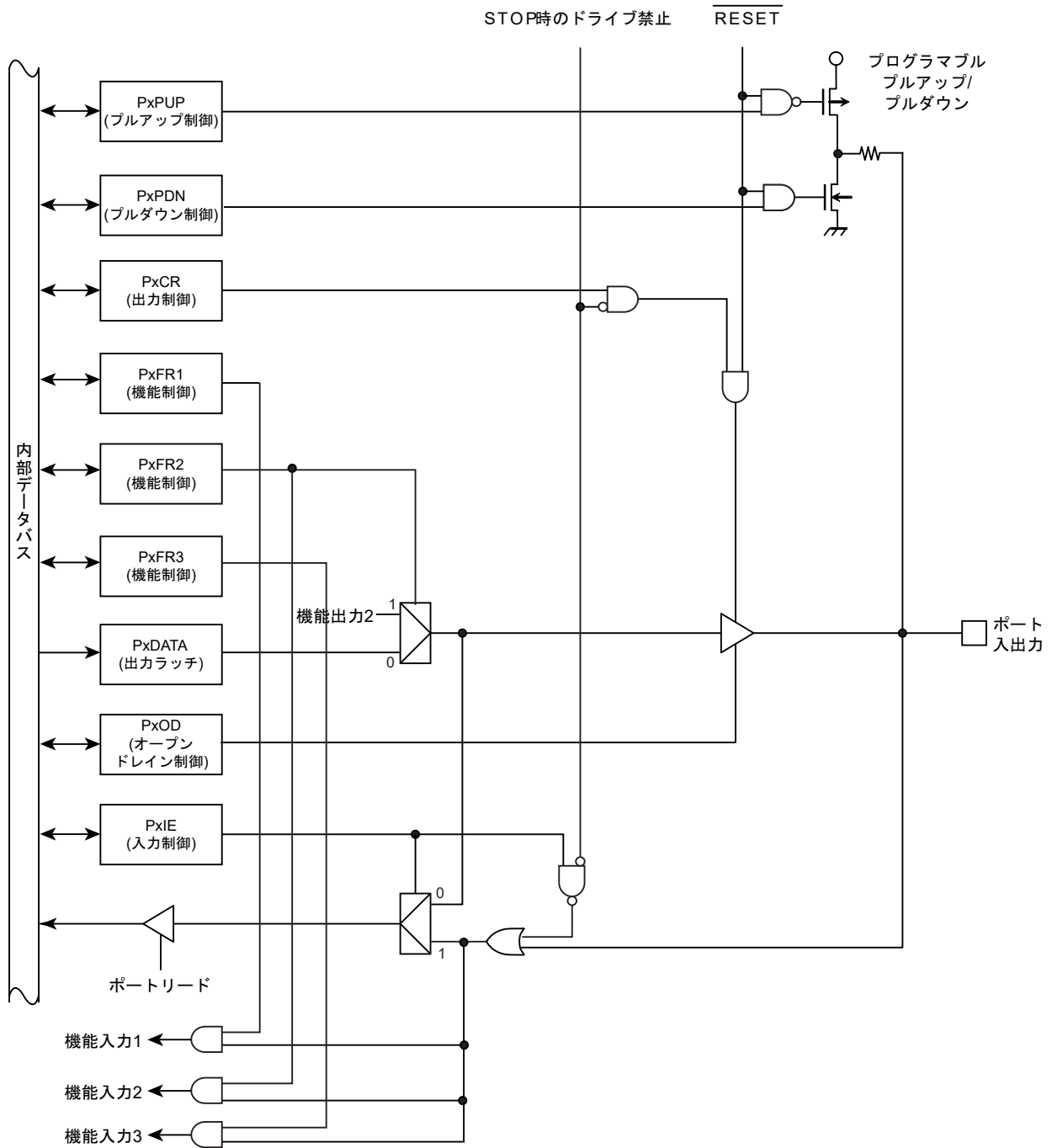


図 7-15 ポートタイプ T15



7.3.17 タイプ T16

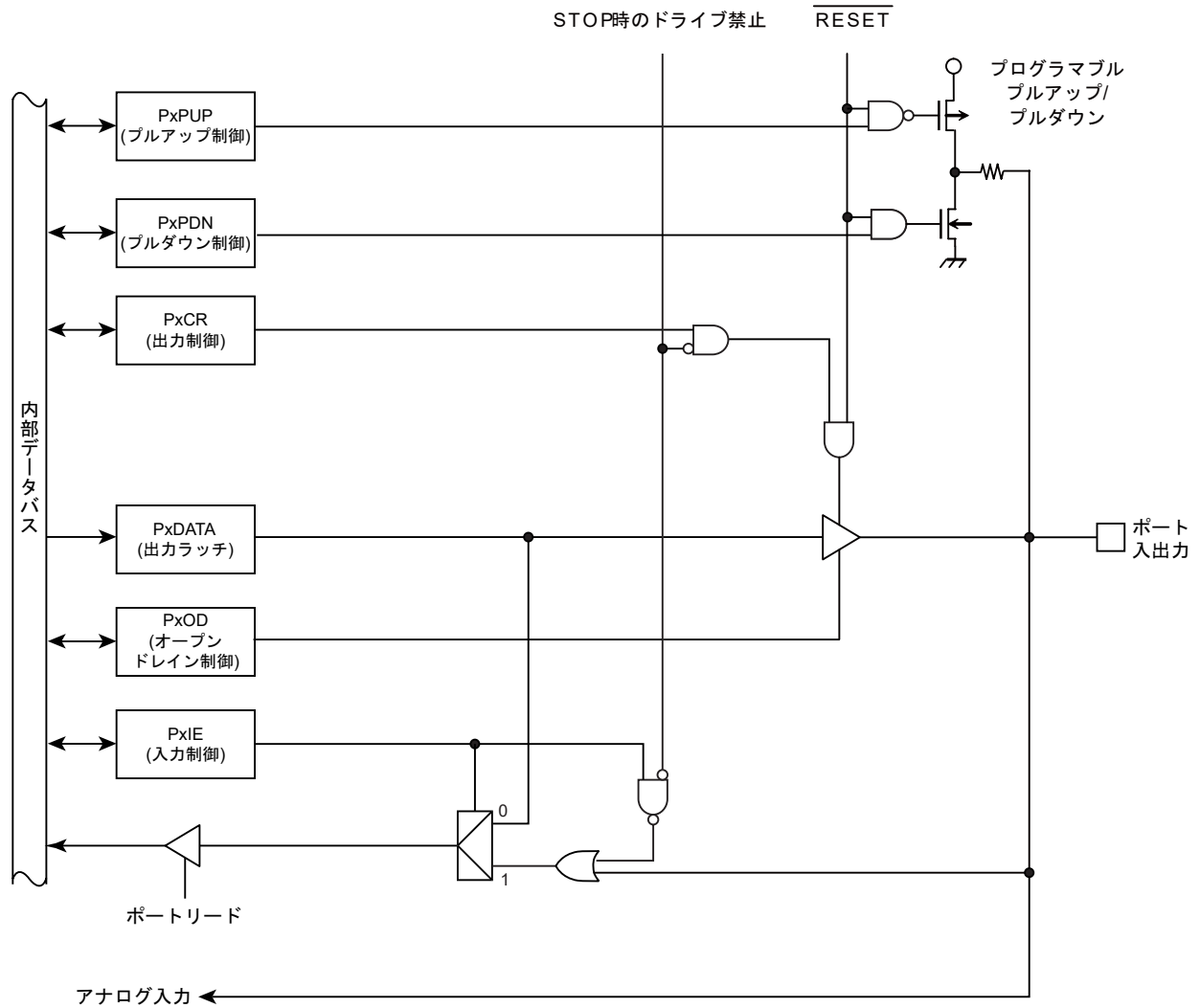


図 7-16 ポートタイプ T16

7.3.18 タイプ T17

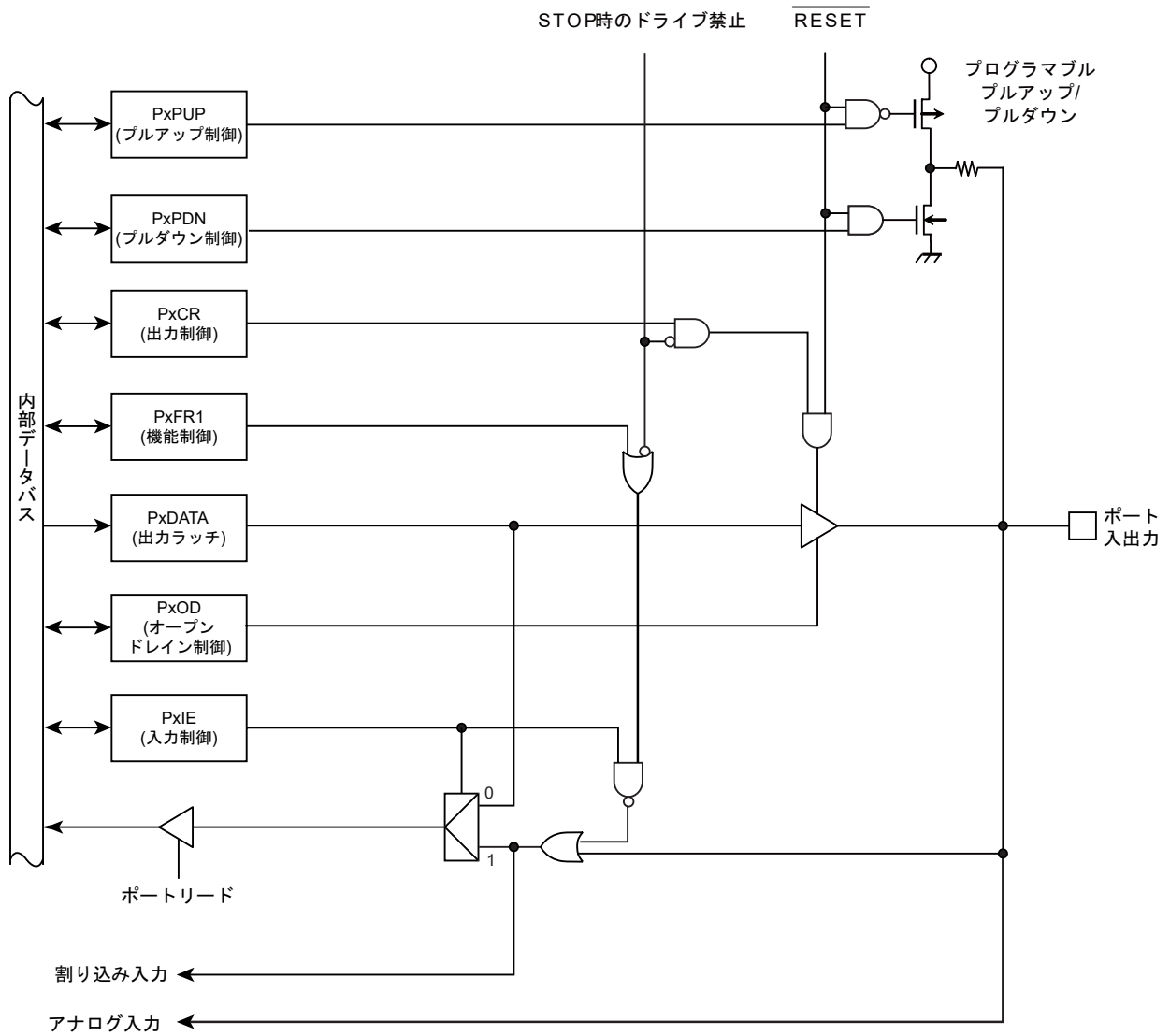


図 7-17 ポートタイプ T17

7.3.19 タイプ T18

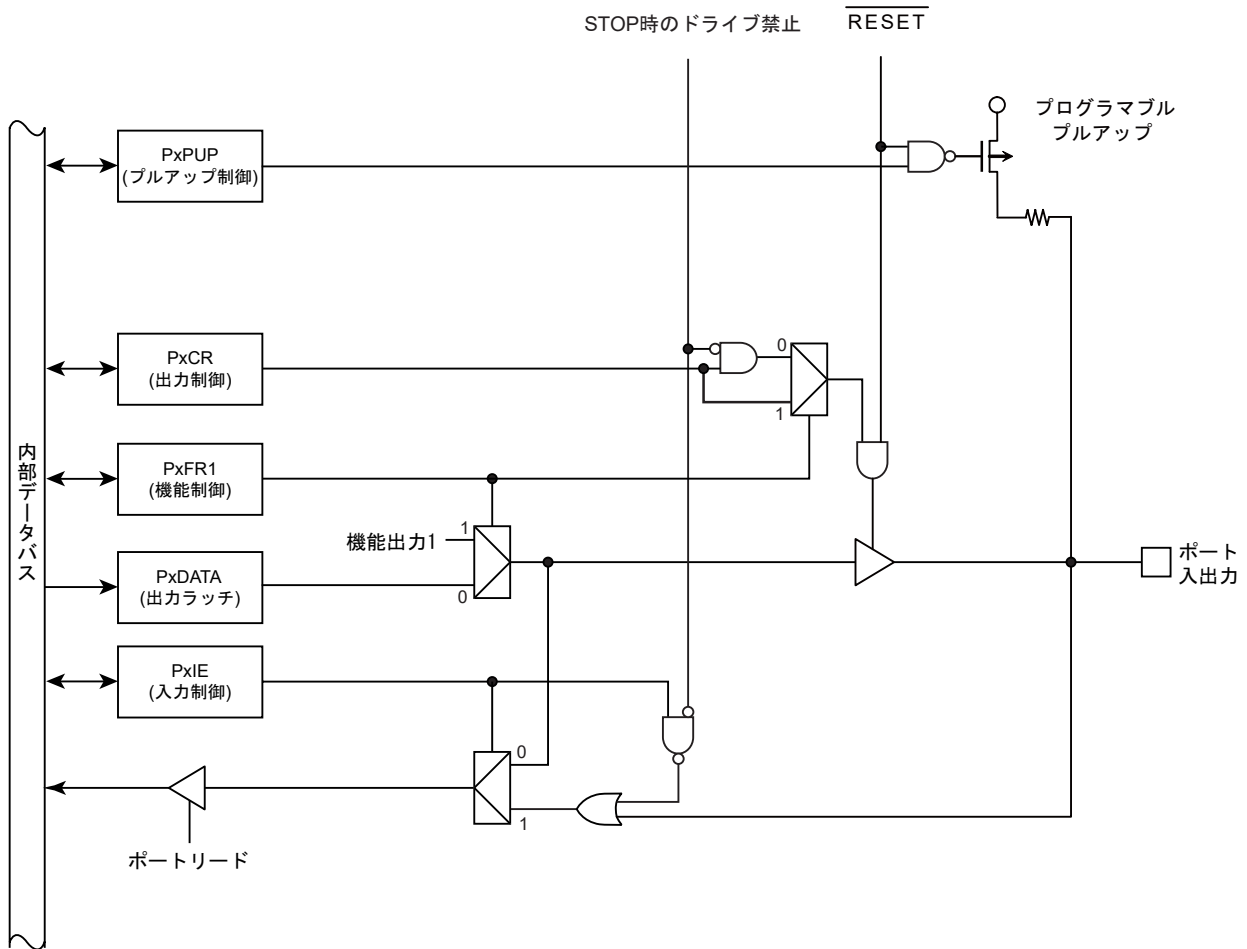


図 7-18 ポートタイプ T18

7.3.20 タイプ T19

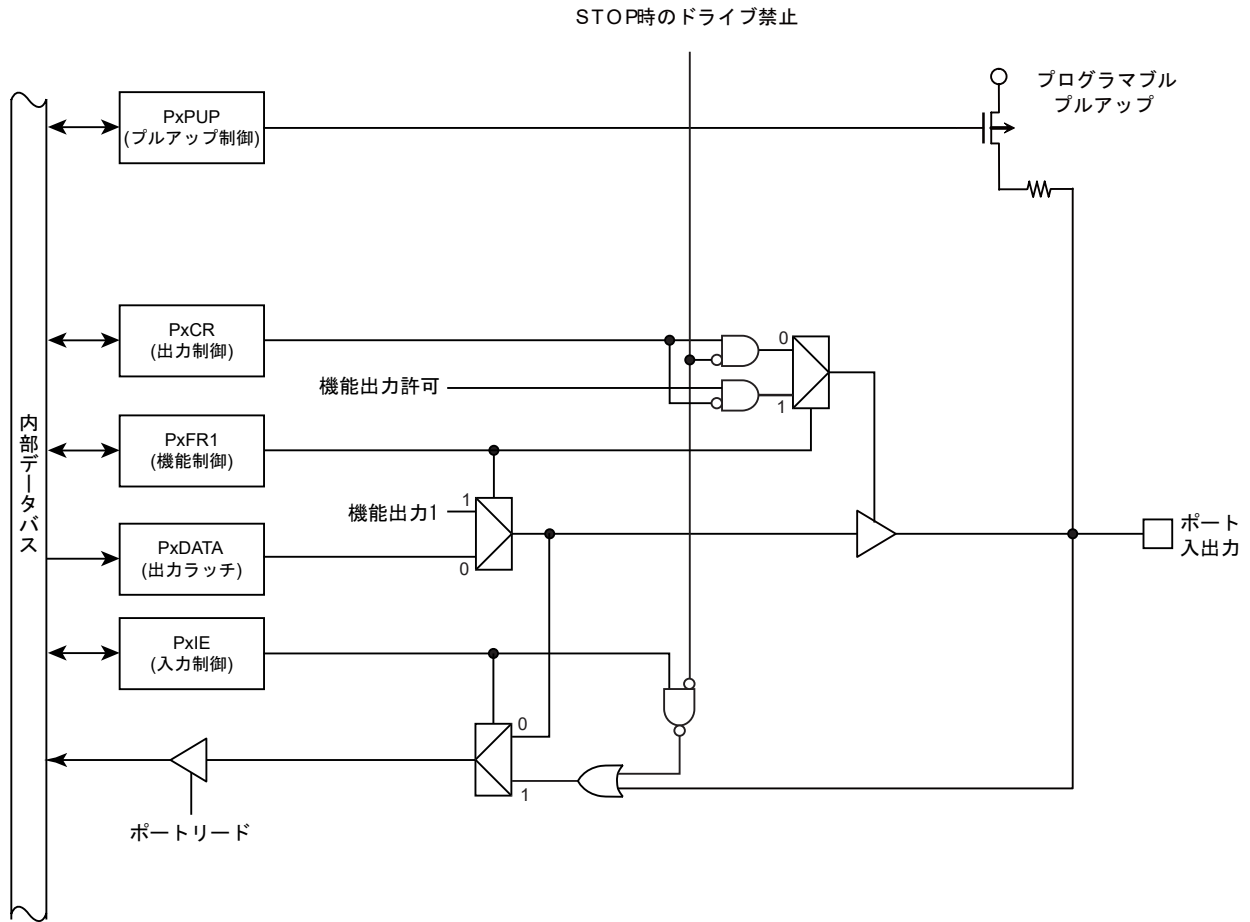


図 7-19 ポートタイプ T19

7.3.21 タイプ T20

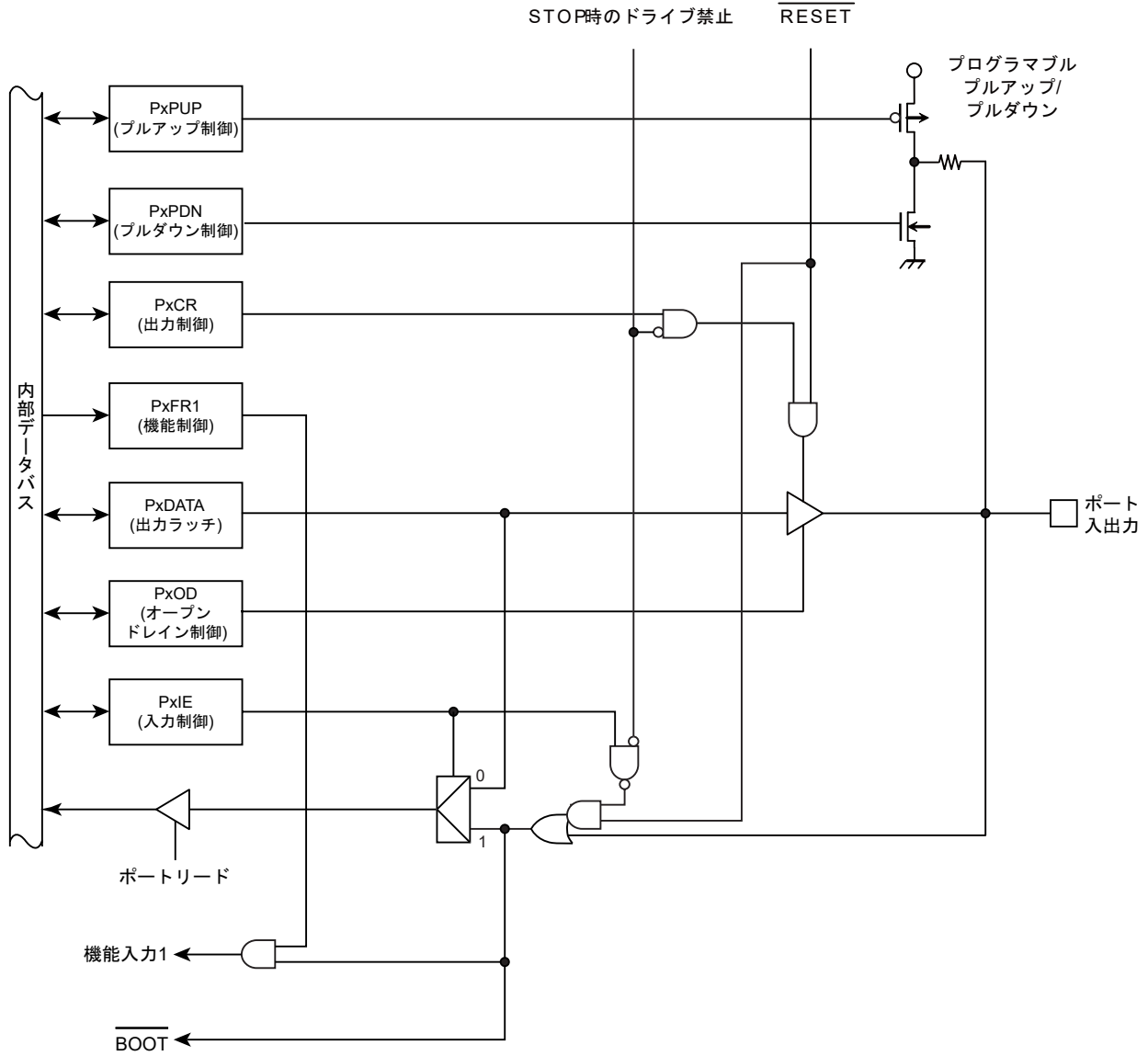


図 7-20 ポートタイプ T20

7.3.22 タイプ T21

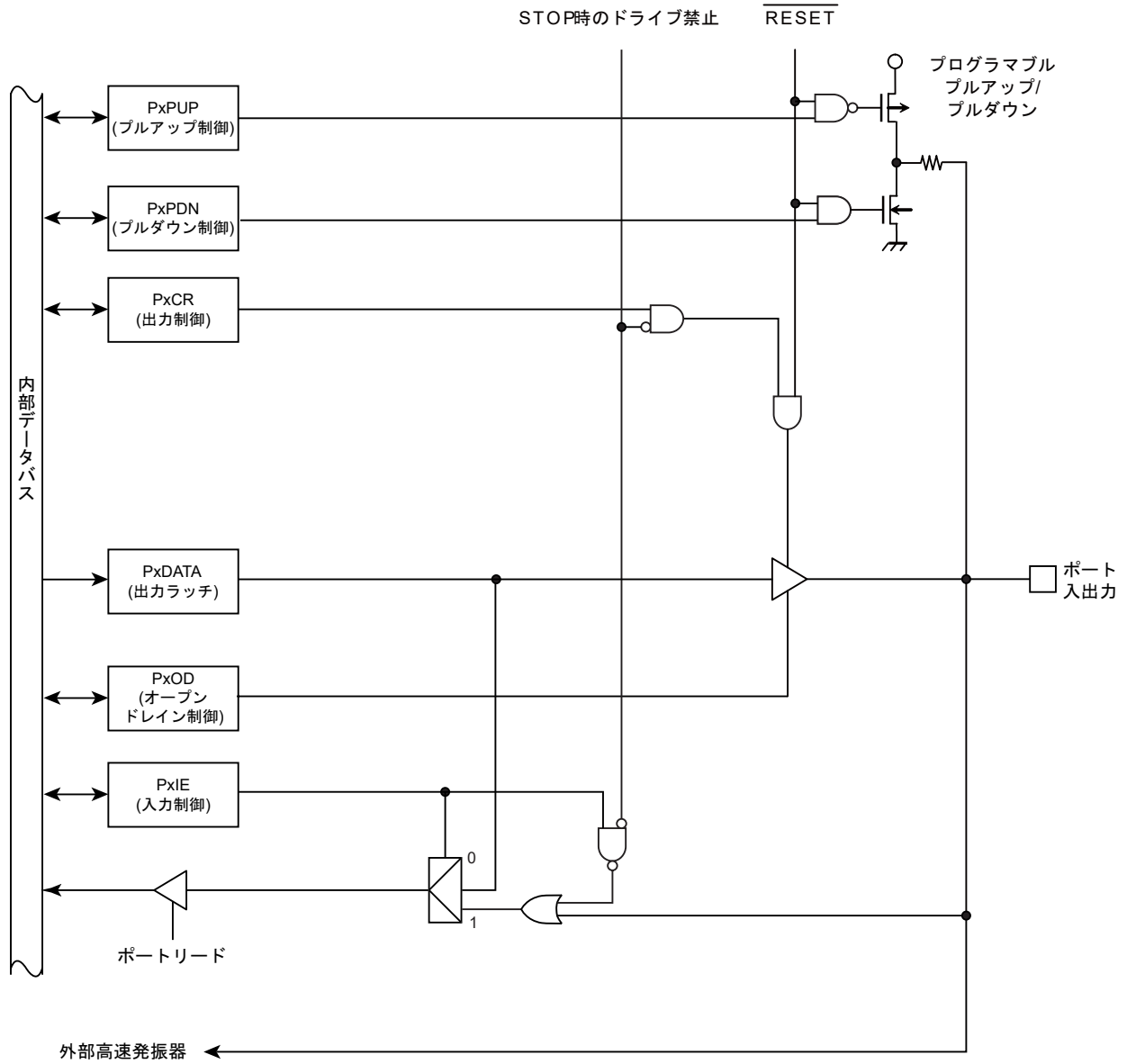


図 7-21 ポートタイプ T21

7.3.23 タイプ T22

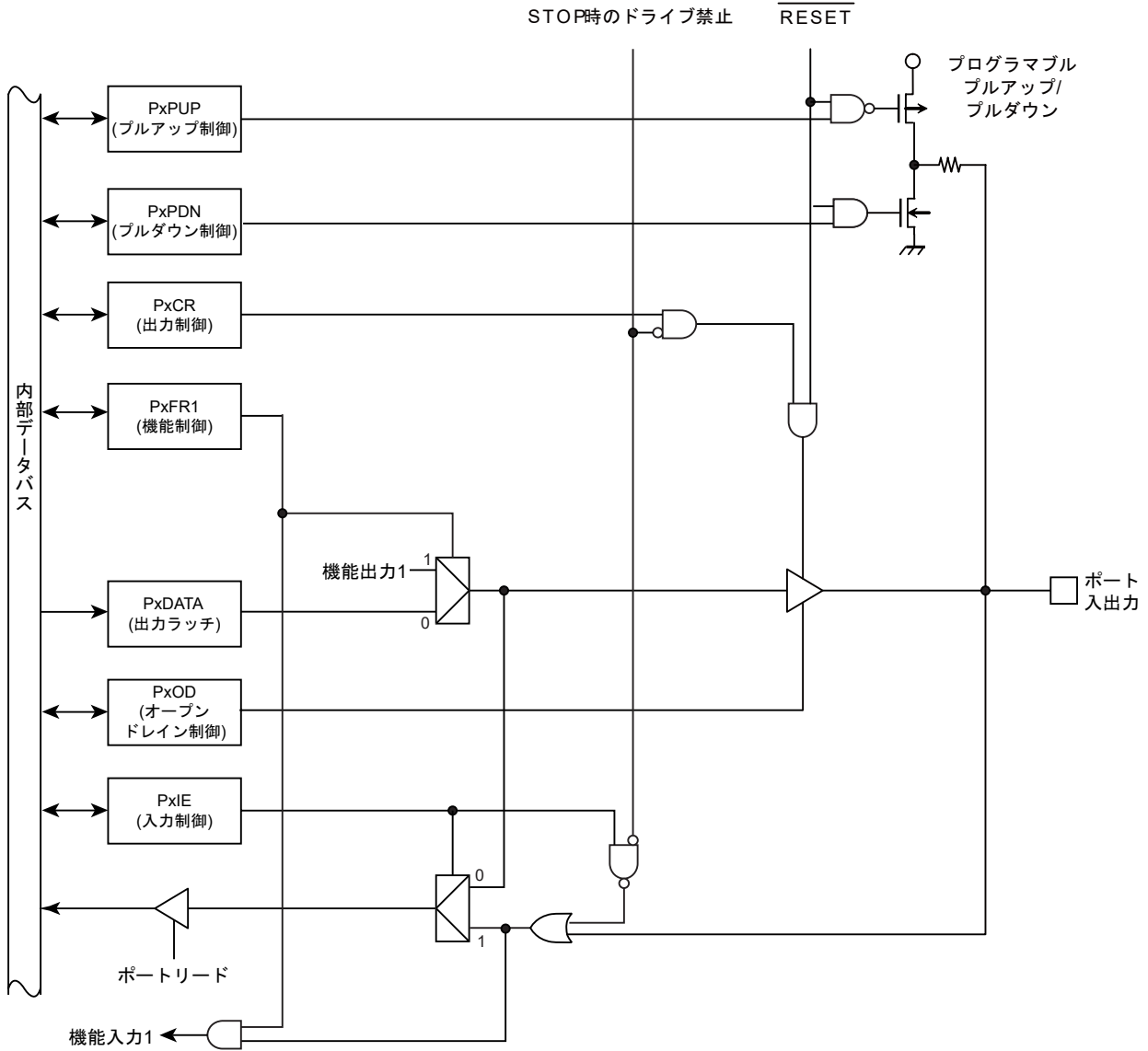


図 7-22 ポートタイプ T22

## 7.4 付録 (ポート設定一覧)

機能ごとのレジスタ設定一覧を以下に示します。

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

"x" のビット設定は任意に行ってください。

### 7.4.1 ポート A 設定

表 7-4 ポート設定一覧 (ポート A)

端子名	ポートタイプ	機能	初期設定	PACR	PAFR1	PAFR2	PAOD	PAPUP	PAPDN	PAIE
PA0	T12	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		TB0IN (入力)		0	1	0	x	x	x	1
		INT3 (入力)		0	0	1	x	x	x	1
PA1	T2	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		TB0OUT (出力)		1	1	-	x	x	x	0
PA2	T12	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		TB1IN (入力)		0	1	0	x	x	x	1
		INT4 (入力)		0	0	1	x	x	x	1
PA3	T2	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		TB1OUT (出力)		1	1	-	x	x	x	0
PA4	T9	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		SCLK1 (入出力)		1	1	0	x	x	x	1
		CTS1 (入力)		0	0	1	x	x	x	1
PA5	T13	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		TXD1 (出力)		1	1	0	x	x	x	0
		TB6OUT (出力)		1	0	1	x	x	x	0
PA6	T11	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		RXD1 (入力)		0	1	0	x	x	x	1
		TB6IN (入力)		0	0	1	x	x	x	1
PA7	T12	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		TB4IN (入力)		0	1	0	x	x	x	1
		INT8 (入力)		0	0	1	x	x	x	1



## 7.4.2 ポート B 設定

表 7-5 ポート設定一覧 (ポート B)

端子名	ポート タイプ	機能	初期 設定	PBCR	PBFR1	PBOD	PBPUP	PBPDN	PBIE
PB0	T18	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TRACECLK (出力)		1	1	0	0	0	0
PB1	T18	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TRACEDATA0 (出力)		1	1	0	0	0	0
PB2	T18	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TRACEDATA1 (出力)		1	1	0	0	0	0
PB3	T6	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TMS / SWDIO (入出力)		1	1	0	1	0	1
PB4	T8	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TCK / SWCLK (入力)		0	1	0	0	1	1
PB5	T19	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TDO / SWV (出力)		1	1	0	0	0	0
PB6	T7	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TDI (入力)		0	1	0	1	0	1
PB7	T7	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		$\overline{\text{TRST}}$ (入力)		0	1	0	1	1	1

### 7.4.3 ポート C 設定

表 7-6 ポート設定一覧 (ポート C)

端子名	ポートタイプ	機能	初期設定	PCCR	PCFR1	PCOD	PCPUP	PCPDN	PCIE
PC0	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		UO0 (出力)		1	1	x	x	x	0
PC1	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		XO0 (出力)		1	1	x	x	x	0
PC2	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		VO0 (出力)		1	1	x	x	x	0
PC3	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		YO0 (出力)		1	1	x	x	x	0
PC4	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		WO0 (出力)		1	1	x	x	x	0
PC5	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		ZO0 (出力)		1	1	x	x	x	0
PC6	T3	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		$\overline{\text{EMG0}}$ (入力)		0	1	x	x	x	1
PC7	T3	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		$\overline{\text{OVV0}}$ (入力)		0	1	x	x	x	1

## 7.4.4 ポート D 設定

表 7-7 ポート設定一覧 (ポート D)

端子名	ポート タイプ	機能	初期 設定	PDCR	PDFR1	PDFR2	PDOD	PDPUP	PDPDN	PDIE
PD0	T11	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		ENCA0 (入力)		0	1	0	x	x	x	1
		TB5IN (入力)		0	0	1	x	x	x	1
PD1	T10	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		ENCB0 (入力)		0	1	0	x	x	x	1
		TB5OUT (出力)		1	0	1	x	x	x	0
PD2	T3	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		ENCZ0 (入力)		0	1	-	x	x	x	1
PD3	T4	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		INT9 (入力)		0	1	-	x	x	x	1
PD4	T9	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		SCLK2 (入出力)		1	1	0	x	x	x	1
		$\overline{\text{CTS2}}$ (入力)		0	0	1	x	x	x	1
PD5	T2	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		TXD2 (出力)		1	1	-	x	x	x	0
PD6	T3	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		RXD2 (入力)		0	1	-	x	x	x	1

### 7.4.5 ポート E 設定

表 7-8 ポート設定一覧 (ポート E)

端子名	ポート タイプ	機能	初期 設定	PECR	PEFR1	PEFR2	PEOD	PEPUP	PEPDN	PEIE
PE0	T2	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		TXD0 (出力)		1	1	-	x	x	x	0
PE1	T3	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		RXD0 (入力)		0	1	-	x	x	x	1
PE2	T9	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		SCLK0 (入出力)		1	1	0	x	x	x	1
		$\overline{CTS0}$ (入力)		0	0	1	x	x	x	1
PE3	T2	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		TB4OUT (出力)		1	1	-	x	x	x	0
PE4	T12	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		TB2IN (入力)		0	1	0	x	x	x	1
		INT5 (入力)		0	0	1	x	x	x	1
PE5	T2	入力ポート		0	0	-	x	x	x	1
		出力ポート		1	0	-	x	x	x	0
		TB2OUT (出力)		1	1	-	x	x	x	0
PE6	T12	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		TB3IN (入力)		0	1	0	x	x	x	1
		INT6 (入力)		0	0	1	x	x	x	1
PE7	T14	入力ポート		0	0	0	x	x	x	1
		出力ポート		1	0	0	x	x	x	0
		TB3OUT (出力)		1	1	0	x	x	x	0
		INT7 (入力)		0	0	1	x	x	x	1

## 7.4.6 ポート F 設定

表 7-9 ポート設定一覧 (ポート F)

端子名	ポート タイプ	機能	初期 設定	PF0CR	PFFR1	PFFR2	PFFR3	PF0D	PFPUP	PFPDN	PFIE
PF0	T20	入力ポート		0	0	-	-	x	x	x	1
		出力ポート		1	0	-	-	x	x	x	0
		TB7IN (入力)		0	1	-	-	x	x	x	1
PF1	T2	入力ポート		0	0	-	-	x	x	x	1
		出力ポート		1	0	-	-	x	x	x	0
		TB7OUT (出力)		1	1	-	-	x	x	x	0
PF2	T15	入力ポート		0	0	0	0	x	x	x	1
		出力ポート		1	0	0	0	x	x	x	0
		ENCA1 (入力)		0	1	0	0	x	x	x	1
		SCLK3 (入出力)		1	0	1	0	x	x	x	1
		$\overline{\text{CTS3}}$ (入力)		0	0	0	1	x	x	x	1
PF3	T10	入力ポート		0	0	0	-	x	x	x	1
		出力ポート		1	0	0	-	x	x	x	0
		ENCB1 (入力)		0	1	0	-	x	x	x	1
		TXD3 (出力)		1	0	1	-	x	x	x	0
PF4	T11	入力ポート		0	0	0	-	x	x	x	1
		出力ポート		1	0	0	-	x	x	x	0
		ENCZ1 (入力)		0	1	0	-	x	x	x	1
		RXD3 (入力)		0	0	1	-	x	x	x	1

注) PF0 は  $\overline{\text{RESET}}$  が "Low" の間 Pull-up と入力が可能になっており、 $\overline{\text{BOOT}}$  入力端子として機能します。

7.4.7 ポート G 設定

表 7-10 ポート設定一覧 (ポート G)

端子名	ポート タイプ	機能	初期 設定	PGCR	PGFR1	PGOD	PGPUP	PGPDN	PGIE
PG0	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		UO1 (出力)		1	1	x	x	x	0
PG1	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		XO1 (出力)		1	1	x	x	x	0
PG2	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		VO1 (出力)		1	1	x	x	x	0
PG3	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		YO1 (出力)		1	1	x	x	x	0
PG4	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		WO1 (出力)		1	1	x	x	x	0
PG5	T1	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		ZO1 (出力)		1	1	x	x	x	0
PG6	T3	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		$\overline{\text{EMG1}}$ (入力)		0	1	x	x	x	1
PG7	T3	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		$\overline{\text{OVV1}}$ (入力)		0	1	x	x	x	1

## 7.4.8 ポート H 設定

表 7-11 ポート設定一覧 (ポート H)

端子名	ポート タイプ	機能	初期 設定	PHCR	PHFR1	PHOD	PHPUP	PHPDN	PHIE
PH0	T17	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		アナログ入力		0	0	0	0	0	0
		INT0 (入力)		0	1	x	x	x	1
PH1	T17	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		アナログ入力		0	0	0	0	0	0
		INT1 (入力)		0	1	x	x	x	1
PH2	T17	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		アナログ入力		0	0	0	0	0	0
		INT2 (入力)		0	1	x	x	x	1
PH3	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PH4	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PH5	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PH6	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PH7	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0

### 7.4.9 ポート I 設定

表 7-12 ポート設定一覧 (ポート I)

端子名	ポート タイプ	機能	初期 設定	PICR	PIOD	PIPUP	PIPDN	PIIE
PI0	T16	入力ポート		0	x	x	x	1
		出力ポート		1	x	x	x	0
		アナログ入力		0	0	0	0	0
PI1	T16	入力ポート		0	x	x	x	1
		出力ポート		1	x	x	x	0
		アナログ入力		0	0	0	0	0
PI2	T16	入力ポート		0	x	x	x	1
		出力ポート		1	x	x	x	0
		アナログ入力		0	0	0	0	0
PI3	T16	入力ポート		0	x	x	x	1
		出力ポート		1	x	x	x	0
		アナログ入力		0	0	0	0	0



## 7.4.10 ポート J 設定

表 7-13 ポート設定一覧 (ポート J)

端子名	ポート タイプ	機能	初期 設定	PJCR	PJFR1	PJOD	PJPUP	PJPDN	PJIE
PJ0	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PJ1	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PJ2	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PJ3	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PJ4	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PJ5	T16	入力ポート		0	-	x	x	x	1
		出力ポート		1	-	x	x	x	0
		アナログ入力		0	-	0	0	0	0
PJ6	T17	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		アナログ入力		0	0	0	0	0	0
		INTC (入力)		0	1	x	x	x	1
PJ7	T17	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		アナログ入力		0	0	0	0	0	0
		INTD (入力)		0	1	x	x	x	1

### 7.4.11 ポート K 設定

表 7-14 ポート設定一覧 (ポート K)

端子名	ポートタイプ	機能	初期設定	PKCR	PKFR1	PKOD	PKPUP	PKPDN	PKIE
PK0	T17	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		アナログ入力		0	0	0	0	0	0
		INTE (入力)		0	1	x	x	x	1
PK1	T17	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		アナログ入力		0	0	0	0	0	0
		INTF (入力)		0	1	x	x	x	1

### 7.4.12 ポート L 設定

表 7-15 ポート設定一覧 (ポート L)

端子名	ポートタイプ	機能	初期設定	PLFR1	PLIE
PL0	T5	入力ポート		0	1
		出力ポート		0	0
		INTB (入力)		1	1
PL1	T5	入力ポート		0	1
		出力ポート		0	0
		INTA (入力)		1	1

### 7.4.13 ポート M 設定

表 7-16 ポート設定一覧 (ポート M)

端子名	ポートタイプ	機能	初期設定	PMCR	PMOD	PMPUP	PMPDN	PMIE
PM0	T21	入力ポート		0	x	x	x	1
		出力ポート		1	x	x	x	0
PM1	T21	入力ポート		0	x	x	x	1
		出力ポート		1	x	x	x	0

X1,X2 あり

## 7.4.14 ポート N 設定

表 7-17 ポート設定一覧 (ポート N)

端子名	ポート タイプ	機能	初期 設定	PNCR	PNFR1	PNOD	PNPUP	PNPDN	PNIE
PN0	T22	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		SO (出力)		1	1	x	x	x	0
		SDA (入出力)		1	1	1	x	x	1
PN1	T22	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		SI (入力)		0	1	x	x	x	1
		SCL (入出力)		1	1	1	x	x	1
PN2	T22	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		SCK (入出力)		1	1	x	x	x	1
PN3	T3	入力ポート		0	0	x	x	x	1
		出力ポート		1	0	x	x	x	0
		TB7IN (入力)		0	1	x	x	x	1

注) TB7IN を選択する場合は <PN3F1> を 1 に設定する場合、<PF0F1> に 0 を設定してください。ただし、<PN3F1> と <PF0F1> の両方に "1" を設定した場合には PF0 の入力が有効となります。



## 第 8 章 16 ビットタイマ / イベントカウンタ (TMRB)

### 8.1 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" はチャンネル番号を表します。

## 8.2 チャンネル別仕様相違点

TMPM376FDDFG/FDFG は、8チャンネルの TMRB を内蔵しています。

各チャンネルはそれぞれ独立に動作します。いずれのチャンネルも表 8-1 に示される仕様相違点を除いて同一の動作をします。

表 8-1 TMRB のチャンネル別仕様相違点

仕様 チャンネル	外部端子		割り込み		内部接続	
	外部クロック / キャプチャトリガ 入力端子	タイマ フリップフロップ 出力端子	キャプチャ 割り込み	TMRB 割り込み	ADC 変換開始	タイマフリップフ ロップ出力 TBxOUT から SIO/ UART(TXTRG: 転送 クロック)
	信号名	信号名				
TMRB0	TB0IN	TB0OUT	INTCAP00 INTCAP01	INTTB00 INTTB01		
TMRB1	TB1IN	TB1OUT	INTCAP10 INTCAP11	INTTB10 INTTB11		
TMRB2	TB2IN	TB2OUT	INTCAP20 INTCAP21	INTTB20 INTTB21		
TMRB3	TB3IN	TB3OUT	INTCAP30 INTCAP31	INTTB30 INTTB31		
TMRB4	TB4IN	TB4OUT	INTCAP40 INTCAP41	INTTB40 INTTB41		SIO0,SIO1
TMRB5	TB5IN	TB5OUT-	INTCAP50 INTCAP51	INTTB50 INTTB51	INTTB51	
TMRB6	TB6IN	TB6OUT	INTCAP60 INTCAP61	INTTB60 INTTB61		
TMRB7	TB7IN	TB7OUT	INTCAP70 INTCAP71	INTTB70 INTTB71		SIO2,SIO3

### 8.3 構成

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

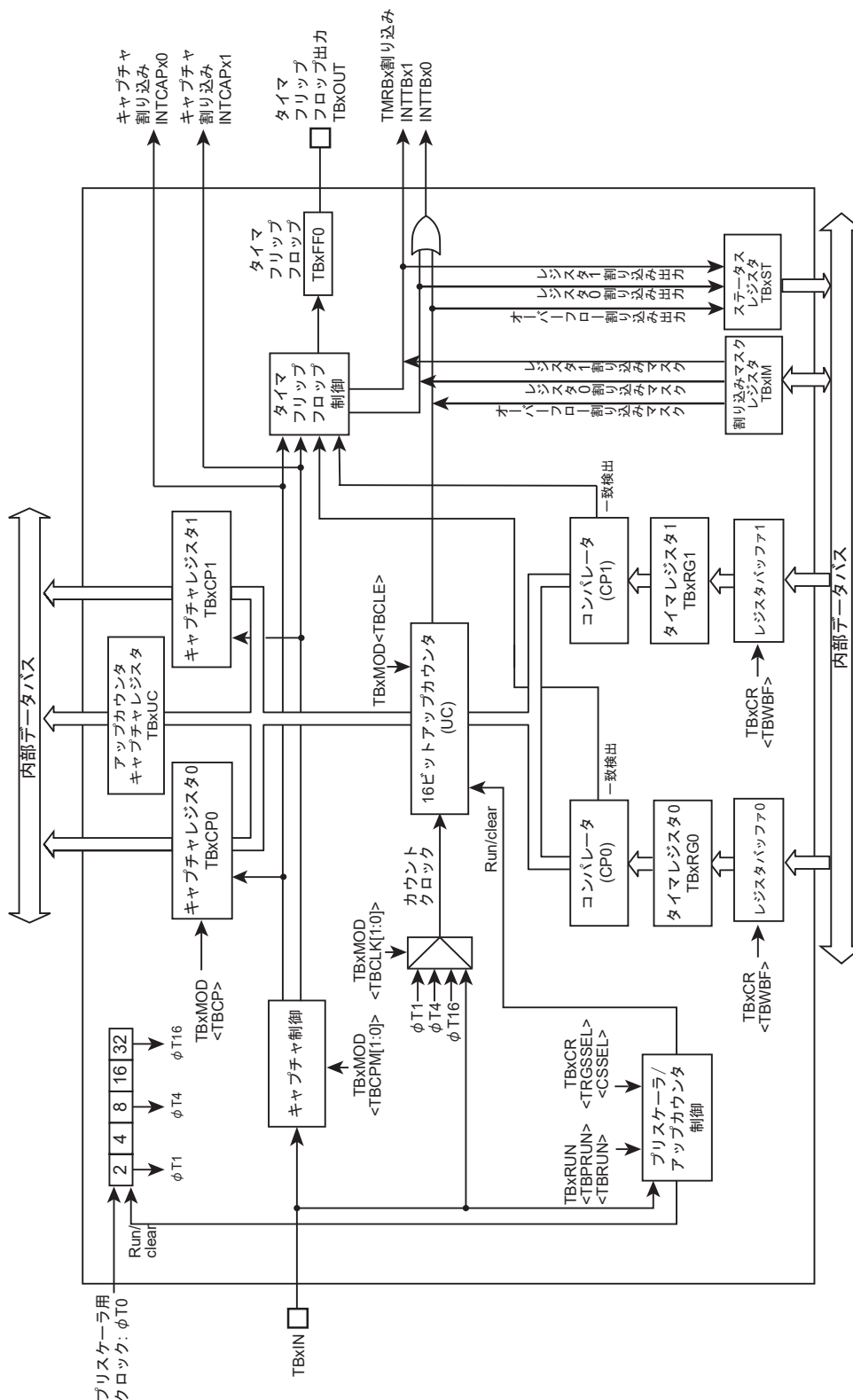


図 8-1 TMRBx ブロック図 (x= 0~7)

## 8.4 レジスタ説明

### 8.4.1 チャンネル別レジスタ一覧

各チャンネルのレジスタとアドレスを以下に示します。

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

レジスタ名 (x=0~7)		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



## 8.4.2 TBxEN( イネーブルレジスタ )

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

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	TBEN	R/W	<p>TMRBx 動作</p> <p>0: 禁止</p> <p>1: 許可</p> <p>TMRB の動作を指定します。動作禁止の状態では TMRB モジュールの他のレジスタへクロックが供給されませんので消費電力の低減が可能です (この状態では、TBxEN レジスタ以外のレジスタへのリード、ライトはできません)。</p> <p>TMRB を使用する場合は、TMRB モジュールの各レジスタを設定する前に TMRB 動作許可 ("1") にしてください。TMRB をいったん動作させた後に、動作禁止した場合は各レジスタの設定は保持されます。</p>
6-0	-	R	リードすると "0" が読めます。

8.4.3 TBxRUN(RUN レジスタ)

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

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると"0"が読めます。
2	TBPRUN	R/W	プリスケラ動作 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 <TBxUC[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。

## 8.4.4 TBxCR( コントロールレジスタ )

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TBWBFB	-	-	-	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	TBWBFB	R/W	ダブルバッファ 0: 禁止 1: 許可
6-5	-	R/W	"0" をライトしてください。
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	リードすると "0" が読めます。
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
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タ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。

8.4.5 TBxMOD( モードレジスタ )

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

Bit	Bit Symbol	Type	機能
31-7	-	R	リードすると "0" が読めます。
6	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。
5	TBCP	W	ソフトウェアキャプチャ制御 0: ソフトキャプチャ 1: Don't care "0" を書き込むとキャプチャレジスタ 0 (TBxCP0) にカウント値を取り込みます。 リードすると "1" が読めます。
4-3	TBCPM[1:0]	R/W	キャプチャタイミング 00: ディセーブル 01: 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: ディセーブル
2	TBCLE	R/W	アップカウンタ制御 0: クリアディセーブル 1: クリアイネーブル アップカウンタのクリア制御を行います。 "0" でクリア禁止、"1" でタイマレジスタ 1 (TBxRG1) との一致時にクリアします。
1-0	TBCLK[1:0]	R/W	TMRBx のソースクロック選択 00: TBxIN 端子入力 01: φT1 10: φT4 11: φT16

注) TBxMOD レジスタはタイマ動作中に変更しないでください。

8.4.6 TBxFFCR( フリップフロップコントロールレジスタ )

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

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

注) TBxFFCR レジスタはタイム動作中に変更しないでください。

8.4.7 TBxST( ステータスレジスタ )

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

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

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

注 2) フラグは自動的にクリアされません。クリアするためには本レジスタをリードしてください。

## 8.4.8 TBxIM( 割り込みマスクレジスタ )

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

Bit	Bit Symbol	Type	機能
31-3	-	R	リードすると "0" が読めます。
2	TBIMOF	R/W	オーバー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ー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。

## 8.4.9 TBxUC( アップカウンタキャプチャレジスタ )

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

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると "0" が読めます。
15-0	TBUC[15:0]	R	アップカウンタ読み出しによりキャプチャした値 <TBxUC[15:0]> 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 1) カウンタ動作時に <TBxUC[15:0]> をリードすると、リード時のアップカウンタの値をキャプチャしリードすることが出来ます。
- 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 <TBxUC[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。



## 8.4.10 TBxRG0( タイマレジスタ 0)

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

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

## 8.4.11 TBxRG1( タイマレジスタ 1)

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

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

8.4.12 TBxCP0( キャプチャレジスタ 0)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBCP0							
リセット後	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。

8.4.13 TBxCP1( キャプチャレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	TBCP1							
リセット後	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。

## 8.5 回路別の動作説明

各チャンネルは表 8-1 に示される仕様相違点を除いて同一の動作をします。

### 8.5.1 プリスケータ

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タです。

プリスケータへの入力クロック  $\phi T0$  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。

プリスケータは TBxRUN<TBPRUN> により動作 / 停止の設定をします。"1" をライトするとカウント開始し "0" をライトするとクリアされ停止します。プリスケータ出力クロックの分解能を、表 8-2 に示します。

表 8-2 プリスケーラ出力クロック分解能 (fc = 80MHz)

ペリフェラル クロック選択 CGSYSCR <FPSEL>	クロックギア値 CGSYSCR <GEAR[2:0]>	プリスケーラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケーラ出力クロック機能		
			$\phi T1$	$\phi T4$	$\phi T16$
0 (fgear)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.025 $\mu s$ )	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )
		001 (fperiph/2)	$fc/2^2$ (0.05 $\mu s$ )	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )
		010 (fperiph/4)	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
	100 (fc/2)	000 (fperiph/1)	$fc/2^2$ (0.05 $\mu s$ )	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )
		001 (fperiph/2)	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )
		010 (fperiph/4)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		011 (fperiph/8)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		100 (fperiph/16)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
		101 (fperiph/32)	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )	$fc/2^{11}$ (25.6 $\mu s$ )
	101 (fc/4)	000 (fperiph/1)	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )
		001 (fperiph/2)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		010 (fperiph/4)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		011 (fperiph/8)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
		100 (fperiph/16)	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )	$fc/2^{11}$ (25.6 $\mu s$ )
		101 (fperiph/32)	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )	$fc/2^{12}$ (51.2 $\mu s$ )
	110 (fc/8)	000 (fperiph/1)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		001 (fperiph/2)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		010 (fperiph/4)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
		011 (fperiph/8)	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )	$fc/2^{11}$ (25.6 $\mu s$ )
		100 (fperiph/16)	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )	$fc/2^{12}$ (51.2 $\mu s$ )
		101 (fperiph/32)	$fc/2^9$ (6.4 $\mu s$ )	$fc/2^{11}$ (25.6 $\mu s$ )	$fc/2^{13}$ (102.4 $\mu s$ )
111 (fc/16)	000 (fperiph/1)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )	
	001 (fperiph/2)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )	
	010 (fperiph/4)	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )	$fc/2^{11}$ (25.6 $\mu s$ )	
	011 (fperiph/8)	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )	$fc/2^{12}$ (51.2 $\mu s$ )	
	100 (fperiph/16)	$fc/2^9$ (6.4 $\mu s$ )	$fc/2^{11}$ (25.6 $\mu s$ )	$fc/2^{13}$ (102.4 $\mu s$ )	
	101 (fperiph/32)	$fc/2^{10}$ (12.8 $\mu s$ )	$fc/2^{12}$ (51.2 $\mu s$ )	$fc/2^{14}$ (204.8 $\mu s$ )	

表 8-2 プリスケーラ出力クロック分解能 (fc = 80MHz)

ペリフェラル クロック選択 CGSYSCR <FPSEL>	クロックギア値 CGSYSCR <GEAR[2:0]>	プリスケーラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケーラ出力クロック機能		
			$\phi T1$	$\phi T4$	$\phi T16$
1 (fc)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.025 $\mu s$ )	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )
		001 (fperiph/2)	$fc/2^2$ (0.05 $\mu s$ )	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )
		010 (fperiph/4)	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
	100 (fc/2)	000 (fperiph/1)	-	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )
		001 (fperiph/2)	$fc/2^2$ (0.05 $\mu s$ )	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )
		010 (fperiph/4)	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
	101 (fc/4)	000 (fperiph/1)	-	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )
		010 (fperiph/4)	$fc/2^3$ (0.1 $\mu s$ )	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
	110 (fc/8)	000 (fperiph/1)	-	-	$fc/2^5$ (0.4 $\mu s$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )
		010 (fperiph/4)	-	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu s$ )	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )
111 (fc/16)	000 (fperiph/1)	-	-	$fc/2^5$ (0.4 $\mu s$ )	
	001 (fperiph/2)	-	-	$fc/2^6$ (0.8 $\mu s$ )	
	010 (fperiph/4)	-	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	
	011 (fperiph/8)	-	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	
	100 (fperiph/16)	$fc/2^5$ (0.4 $\mu s$ )	$fc/2^7$ (1.6 $\mu s$ )	$fc/2^9$ (6.4 $\mu s$ )	
	101 (fperiph/32)	$fc/2^6$ (0.8 $\mu s$ )	$fc/2^8$ (3.2 $\mu s$ )	$fc/2^{10}$ (12.8 $\mu s$ )	

注 1) プリスケーラ出力クロック  $\phi Tn$  は、必ず  $\phi Tn < fsys$  を満足するように ( $\phi Tn$  が  $fsys$  よりも遅くなるように) 選択してください。

注 2) タイマ動作中はクロックギアの切り替えは行わないでください。

注 3) 表中 "-" は設定禁止です。

## 8.5.2 アップカウンタ (UC)

16ビットのバイナリカウンタです。

- ソースクロック

ソースクロックは  $TB_xMOD\langle TBCLK[1:0]\rangle$  で設定することができます。

プリスケアラ出力クロック  $\phi T1, \phi T4, \phi T16$ 、または、 $TB_xIN$  入力のいずれかを選択できます。

- カウンタ動作の開始と停止

カウンタ動作は  $TB_xRUN\langle TBRUN\rangle$  で行います。 $\langle TBRUN\rangle = "1"$  でカウントを開始し、 $"0"$  でカウント停止と同時にカウンタのクリアを行います。

- カウンタクリアのタイミング

1. コンペア一致時

$TB_xMOD\langle TBCLE\rangle = "1"$  に設定することで、 $TB_xRG1$  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。 $TB_xMOD\langle TBCLE\rangle = "0"$  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

$TB_xRUN\langle TBRUN\rangle = "0"$  に設定すると、カウンタが停止するとともにクリアされます。

- カウンタのオーバフロー

アップカウンタUCがオーバフローすると、オーバフロー割り込みINTTBx0が発生します。

## 8.5.3 タイマレジスタ (TBxRG0, TBxRG1)

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

$TB_xRG0/1$  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  $TB_xCR\langle TBWBF\rangle$  によって行います。 $\langle TBWBF\rangle = "0"$  のときディセーブル、 $\langle TBWBF\rangle = "1"$  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  $TB_xRG1$  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  $TB_xRG0/1$  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、 $TB_xRG0/1$  に直接値を書き込むことができます。

#### 8.5.4 キャプチャ制御

アップカウンタ UC の値をキャプチャレジスタ TBxCP0, TBxCP1 にラッチするタイミングを制御する回路です。キャプチャレジスタのラッチタイミングは、TBxMOD<TBCPM[1:0]> で設定します。

また、ソフトウェアによってもアップカウンタ UC の値をキャプチャレジスタへ取り込むことができ、TBxMOD<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。

#### 8.5.5 キャプチャレジスタ (TBxCP0, TBxCP1)

アップカウンタ UC の値をキャプチャするレジスタです。

#### 8.5.6 アップカウンタキャプチャレジスタ (TBxUC)

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

#### 8.5.7 コンパレータ (CP0, CP1)

アップカウンタ UC と、タイマレジスタ TBxRG0, TBxRG1 への設定値とを比較し、一致を検出します。一致すると、INTTBx0 と INTTBx1 を発生します。

#### 8.5.8 タイマフリップフロップ (TBxFF0)

タイマフリップフロップ (TBxFF0) は、コンパレータからの一致信号、キャプチャレジスタへのラッチ信号によって反転するフリップフロップです。反転のディセーブル / イネーブルは、TBxFFCR<TBCIT1, TBCOT1, TBEIT1, TBEOT1> 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<TBFFOC[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。

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

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

## 8.6 モード別動作説明

### 8.6.1 16ビットインタバルタイマモード

一定周期の割り込みを発生させる場合、タイマレジスタ TBxRG0 にインタバル時間を設定し、INTTBx0 割り込みを発生します。同様にタイマレジスタ TBxRG1 にインタバル時間を設定することで 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	← X	0	1	0	0	1	*	*	入力クロックをプリスケアラ出力クロックにし、キャプチャ機能ディセーブルにします。
						(** = 01, 10, 11)			
TBxRG1	← *	*	*	*	*	*	*	*	インタバル時間を設定します。(16ビット)
	← *	*	*	*	*	*	*	*	
TBxRUN	← *	*	*	*	*	1	X	1	TMRBx を起動します。

注) X; Don't care  
-; No change

### 8.6.2 16ビットイベントカウンタモード

入力クロックを外部クロック (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。ソフトウェアキャプチャを行い、キャプチャ値をリードすることでカウント値を読むことができます。

	7	6	5	4	3	2	1	0	
TBxEN	← 1	X	X	X	X	X	X	X	TMRBx モジュールを起動します。
TBxRUN	← X	X	X	X	X	0	X	0	TMRBx を停止します
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	← X	0	1	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	← X	0	0	0	0	0	0	0	ソフトウェアキャプチャを行います。

注) X; Don't care  
-; No change



### 8.6.3 16ビットPPG (プログラマブル矩形波) 出力モード

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

アップカウンタ (UC) とタイマレジスタ (TBxRG0, TBxRG1) への設定値との一致によりタイマフリップフロップ (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

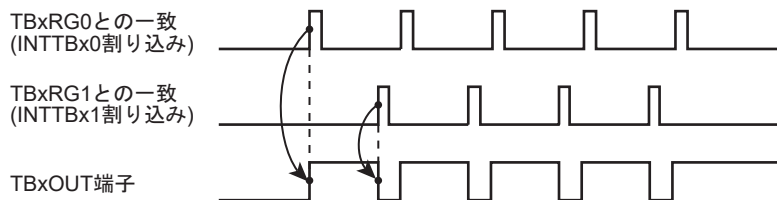


図 8-2 プログラマブル矩形波 (PPG) 出力波形例

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。

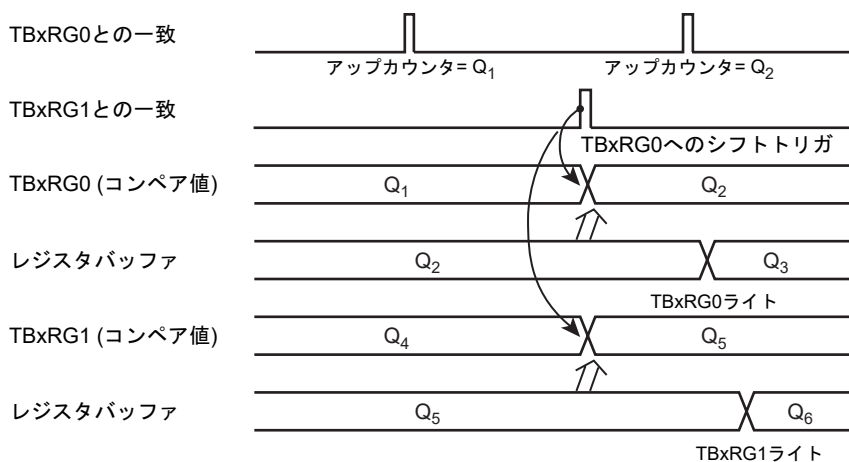


図 8-3 レジスタバッファの動作

このモードのブロック図を示します。

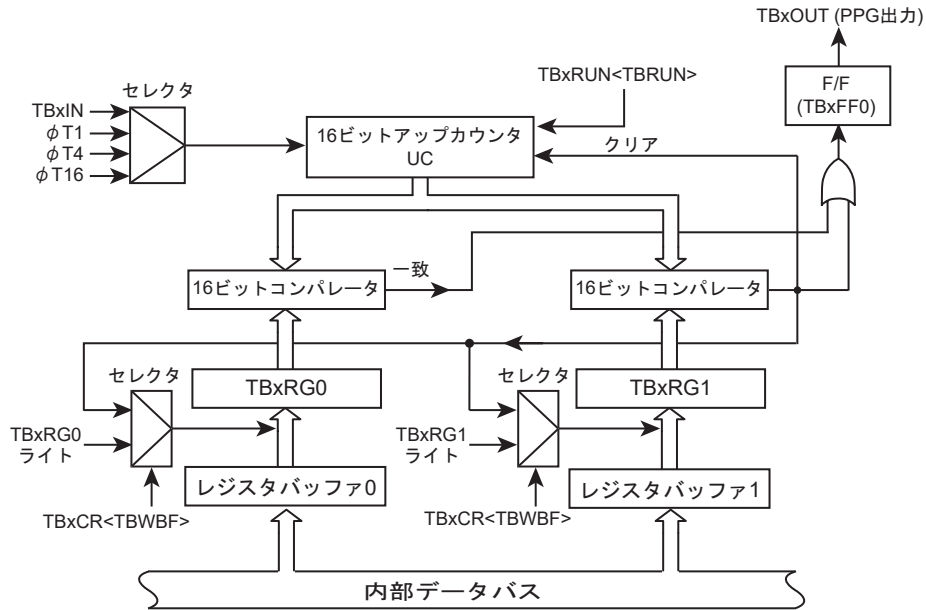


図 8-4 16 ビット PPG モードのブロック図

16 ビット PPG 出力モード時の各レジスタは、次のように設定します。

	7	6	5	4	3	2	1	0	
TBxEN	← 1	X	X	X	X	X	X	X	TMRBx モジュールを起動します。
TBxRUN	← X	X	X	X	X	0	X	0	TMRBx を停止します
TBxCR	← 0	0	-	X	-	X	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	-	X	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	← X	0	1	0	0	1	*	*	入力クロックをプリスケアラ出力クロックにし、キャプチャ機能ディセーブルにします。 UC は TBxRG1 との一致でクリアされます。
PORT 関連レジスタ設定									
TBxRUN	← *	*	*	*	*	1	X	1	該当ポートを TBxOUT になるように設定します。 TMRBx を起動します。

注) X; Don't care  
-; No change

### 8.6.4 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。

$TBxFFCR<TBEIT1, TBE0T1>$  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 8-5 の  $d$ 、 $p$  と対応しています。

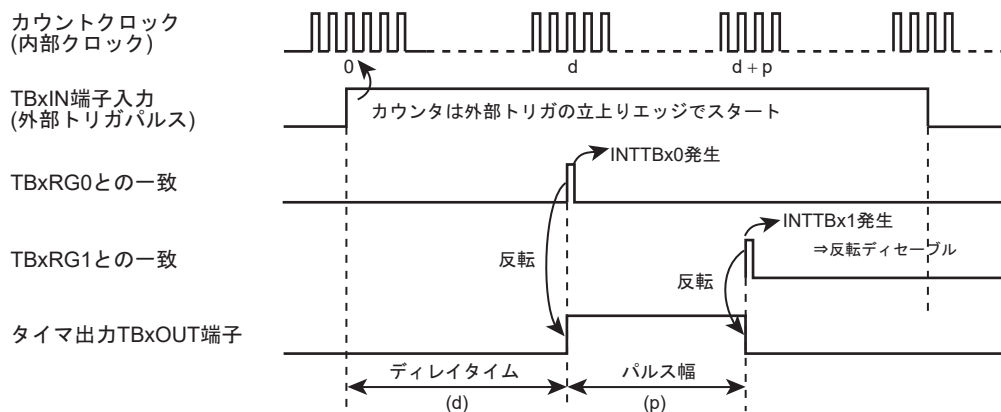


図 8-5 外部トリガカウントスタートを使用したワンショットパルス出力 (ディレイあり)

## 8.7 キャプチャ機能を利用した応用例

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

1. 外部トリガパルスからのワンショットパルス出力
2. パルス幅測定

### 8.7.1 外部トリガパルスからのワンショットパルス出力

外部トリガパルスからのワンショットパルス出力は、次のように行います。

16 ビットアップカウンタ UC をプリスケ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<TBEIT1, 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 8-6 の c, d, p と対応しています。

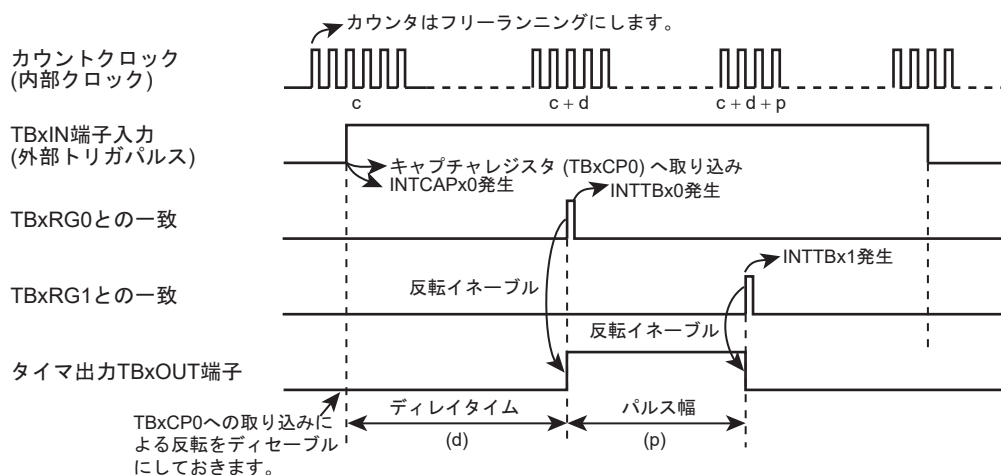


図 8-6 ワンショットパルス出力 (ディレイあり)

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  $\phi T1$  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									
TBxEN	← 1	X	X	X	X	X	X	X	該当ポートを TBxIN になるように設定します。
TBxRUN	← X	X	X	X	X	0	X	0	TMRBx モジュールを起動します。
TBxMOD	← X	0	1	0	1	0	0	1	TMRBx を停止します
TBxFFCR	← X	X	0	0	0	0	1	0	ソースクロックを $\phi T1$ にし、TBxIN 立ち上がりで TBxCP0 へカウント値を取り込みます。
PORT 関連レジスタ設定									
割り込みイネーブル セットレジスタ	← *	*	*	*	*	*	*	*	INTCAPx0 割り込みに対応するビットを "1" にし、割り込みを許可します。
TBxRUN	← *	*	*	*	*	1	X	1	TMRBx を起動します。
[INTCAPx0 割り込みサービスルーチンでの処理] パルス出力設定									
TBxRG0	← *	*	*	*	*	*	*	*	カウント値を設定します。(TBxCAP0 + 3ms/ $\phi T1$ )
TBxRG1	← *	*	*	*	*	*	*	*	カウント値を設定します。(TBxCAP0 + (3+2)ms/ $\phi T1$ )
TBxFFCR	← X	X	-	-	1	1	-	-	TBxRG0, TBxRG1 との一致で TBxFF0 を反転します。
TBxIM	← X	X	X	X	X	1	0	1	TBxRG1 との一致割り込み以外をマスクします。
割り込みイネーブル セットレジスタ	← *	*	*	*	*	*	*	*	INTTBx 割り込みに対応するビットを "1" にし、割り込みを許可します。
[INTTBx 割り込みサービスルーチンでの処理] 出力ディセーブル									
TBxFFCR	← X	X	-	-	0	0	-	-	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。

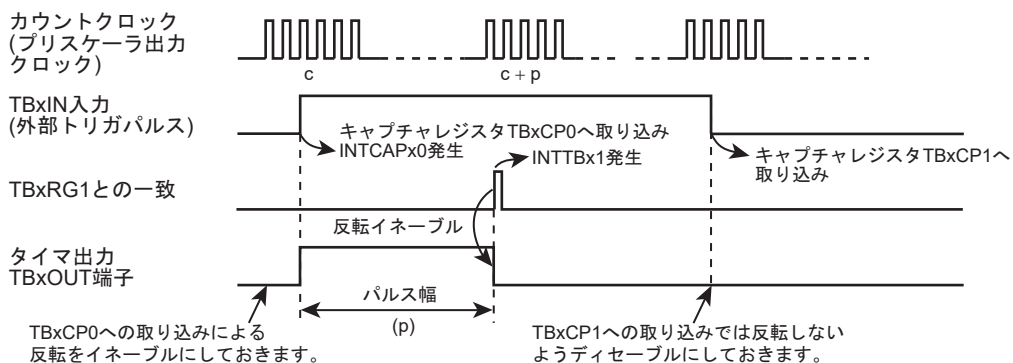


図 8-7 外部トリガパルスのワンショットパルス出力 (ディレイなし)

## 8.7.2 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) をプリスケアラ出力クロックにてフリーランニングでカウントアップさせておきます。キャプチャ機能を用いて、外部パルスの立ち上がり / 立ち下がり、それぞれのエッジでトリガをかけ、このときのアップカウンタ値をキャプチャレジスタ (TBxCP0, TBxCP1) に取り込みます。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、プリスケアラ出力クロックの周期が  $0.5 \mu\text{s}$  であれば、パルス幅は、 $100 \times 0.5 \mu\text{s} = 50 \mu\text{s}$  となります。

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

また、外部パルスの "Low" レベル幅を測定することもできます。この場合、図 8-8 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、プリスケアラ出力クロックの周期をかけることにより、求めることができます。

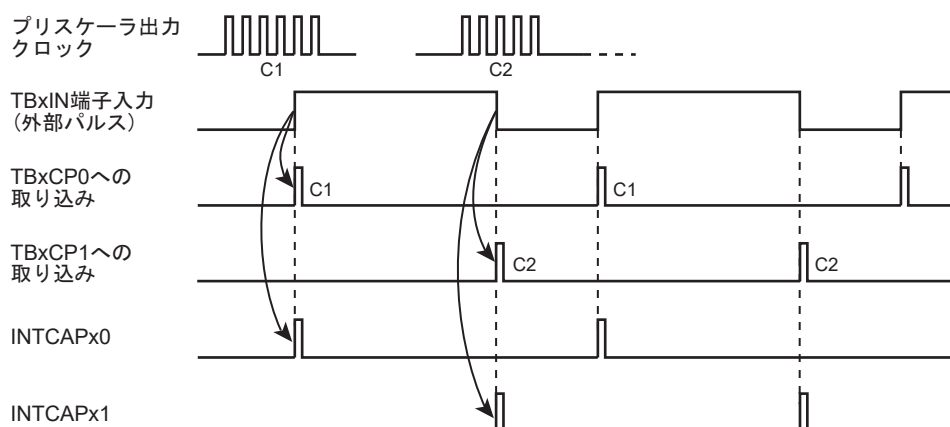


図 8-8 パルス幅測定

## 第9章 シリアルチャネル (SIO/UART)

### 9.1 概要

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

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

以下の説明中、"x" はチャネル番号をあらわします。

### 9.2 チャネル別仕様相違点

TMPM376FDDFG/FDFG は 4 チャネルの SIO を内蔵しています。

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

表 9-1 SIO のチャネル別仕様相違点

	端子名			割り込み		シリアルクロック生成タイム
	TXD	RXD	$\overline{\text{CTSx}}$ / SCLKx	受信割り込み	送信割り込み	
チャネル 0	PE0	PE1	PE2	INTRX0	INTTX0	TB4OUT
チャネル 1	PA5	PA6	PA4	INTRX1	INTTX1	TB4OUT
チャネル 2	PD5	PD6	PD4	INTRX2	INTTX2	TB7OUT
チャネル 3	PF3	PF4	PF2	INTRX3	INTTX3	TB7OUT

### 9.3 構成

図 9-1 に SIO のブロック図を示します。

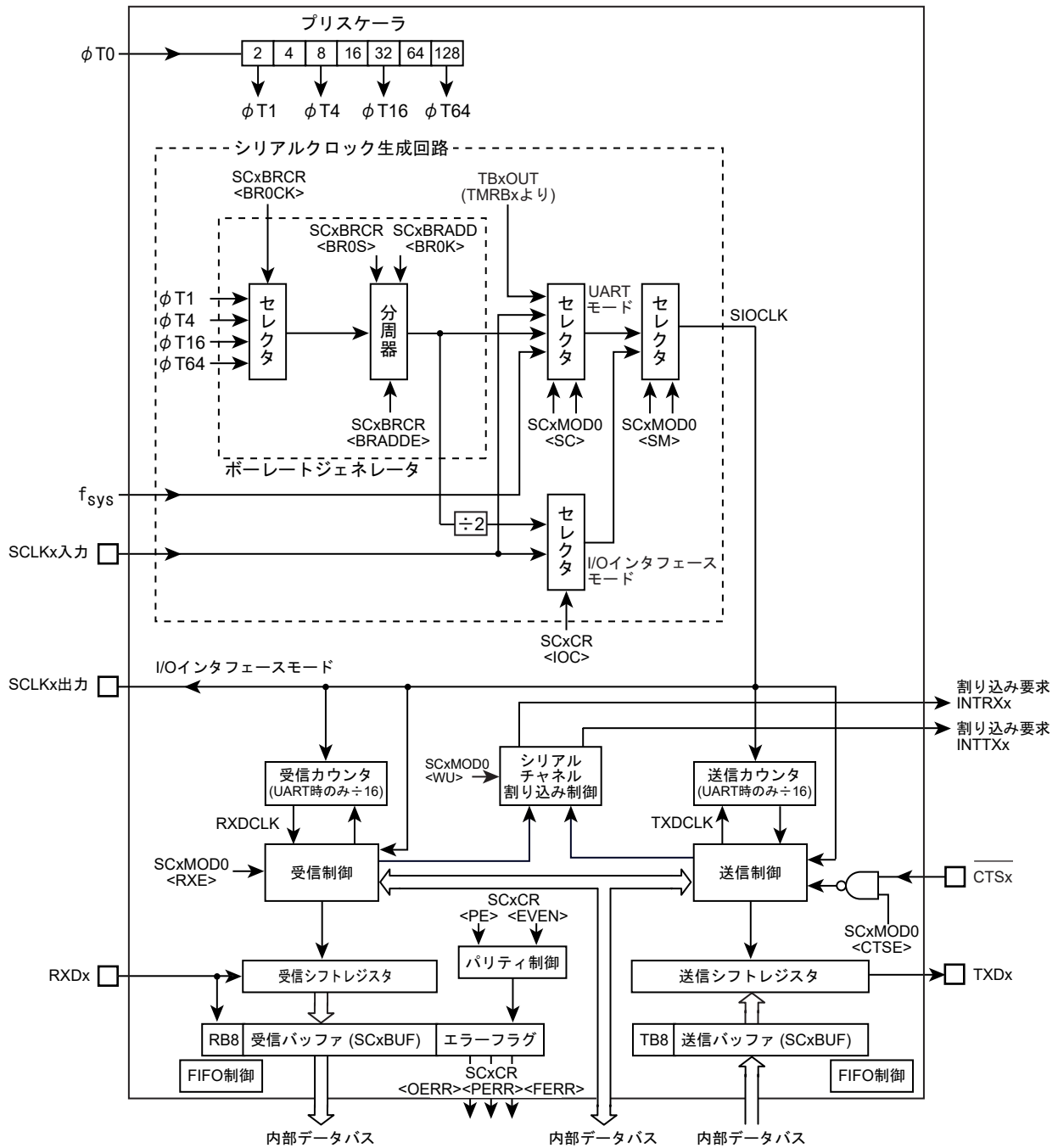


図 9-1 SIO ブロック図



## 9.4 レジスタ説明

### 9.4.1 チャネル別レジスタ一覧

各チャネルのレジスタとアドレスを示します。

Channel x	Base Address
Channel0	0x4002_0080
Channel1	0x4002_00C0
Channel2	0x4002_0100
Channel3	0x4002_0140

レジスタ名 (x=0,1,2,3,)		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
ポーレートジェネレータコントロールレジスタ	SCxBRCR	0x0010
ポ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 1) 送信中、受信中に制御レジスタを書き換えないでください。

9.4.2 SCxEN ( イネーブルレジスタ )

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

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると "0" が読めます。
0	SIOE	R/W	<p>SIO 動作</p> <p>0: 禁止</p> <p>1: 動作</p> <p>SIO の動作を指定します。SIO を使用する場合は、まず &lt;SIOE&gt; に "1" をセットしてください。</p> <p>動作禁止の状態では、イネーブルレジスタを除く SIO のすべてのクロックが停止しますので消費電力の低減が可能です。</p> <p>SIO を一旦動作させた後に動作禁止にした場合は、SCxTFC&lt;TIL[1:0]&gt; を除くレジスタの設定は保持されます。</p>

注) SCxEN<SIOE>=0(SIO 動作禁止)、または SCxMOD1<I2S0>=0 で IDLE モードへの移行 (IDLE モード中の動作禁止) した場合は、必ず SCxTFC の再設定を行ってください。

## 9.4.3 SCxBUF (バッファレジスタ)

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

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

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

9.4.4 SCxCR (コントロールレジスタ)

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

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	RB8	R	受信データビット 8 (UART 用) 9 ビット UART モード時の 9 ビット目の受信データです。
6	EVEN	R/W	パリティ (UART 用) 0: Odd 1: Even パリティの条件を設定します。 "0" で奇数 (Odd) パリティ、"1" で偶数 (Even) パリティです。 パリティは、7 ビット UART モード、8 ビット UART モードで使用可能です。
5	PE	R/W	パリティ付加 (UART 用) 0: 禁止 1: 許可 パリティ許可 / 禁止を制御するビットです。 パリティは、7 ビット UART モード、8 ビット UART モードで使用可能です。
4	OERR	R	オーバランエラー (注) 0: エラーではない 1: エラー
3	PERR	R	パリティ / アンダーランエラー (注) 0: エラーではない 1: エラー
2	FERR	R	フレーミングエラー (注) 0: エラーではない 1: エラー
1	SCLKS	R/W	入力クロックエッジ選択 (IO インタフェース用) 0: SCLKx の立ち下がりエッジで送信バッファのデータを 1bit ずつ TXDx 端子へ出力します。 SCLKx の立ち上がりエッジで RXDx 端子のデータを 1bit ずつ受信バッファに取り込みます。 この時、SCLKx は High レベルからスタートします。 1: SCLKx の立ち上がりエッジで送信バッファのデータを 1bit ずつ TXDx 端子へ出力します。 SCLKx の立ち下がりエッジで RXDx 端子のデータを 1bit ずつ受信バッファに取り込みます。 この時、SCLKx は Low レベルからスタートします。  クロック出力モード時は、"0" を設定してください。
0	IOC	R/W	クロック選択 (IO インタフェース用) 0: ボーレートジェネレータ 1: SCLK 端子入力

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

9.4.5 SCxMOD0 (モードコントロールレジスタ 0)

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

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	TB8	R/W	送信データビット 8 (UART 用) 9 ビット UART モード時の 9 ビット目の送信データをセットします。
6	CTSE	R/W	ハンドシェイク機能制御 (UART 用) 0: CTS ディセーブル 1: CTS イネーブル ハンドシェイク機能の制御を行います。 イネーブルにすると CTS 端子を用いたハンドシェイク機能が使用可能になります。
5	RXE	R/W	受信制御 (注 1)(注 2) 0: 禁止 1: 許可
4	WU	R/W	ウェイクアップ機能 (UART 用) 0: ディセーブル 1: イネーブル 9 ビット UART モードの場合のみ有効で、その他のモードでは無視されます。 イネーブルの場合、受信データ 9 ビット目が "1" のときのみ割り込みが発生します。
3-2	SM[1:0]	R/W	シリアル転送モード 00: IO インタフェースモード 01: 7 ビット長 UART モード 10: 8 ビット長 UART モード 11: 9 ビット長 UART モード
1-0	SC[1:0]	R/W	シリアル転送クロック (UART 用) 00: タイマ TBxOUT (表 9-1 を参照) 01: ボーレートジェネレータ 10: 内部クロック fsys 11: 外部クロック (SCLK 入力) (IO インタフェースモード時の転送クロックは、コントロールレジスタ (SCxCR) で選択します。)

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

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

9.4.6 SCxMOD1 (モードコントロールレジスタ 1)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	I2S0	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	I2S0	R/W	IDLE モード時の動作 0: 停止 1: 動作 IDLE モード時の動作を指定します。
6-5	FDPX[1:0]	R/W	転送モード設定 00: 転送禁止 01: 半二重 (受信) 10: 半二重 (送信) 11: 全二重 IO インタフェースモード時の転送モードを設定します。また FIFO が許可されている場合は FIFO の構成を指定します。 UART モードの場合は FIFO 構成の指定のみ行われます。
4	TXE	R/W	送信制御 (注 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ーバル時間 (IO インタフェース用) 000: なし 001: 1SCLK 010: 2SCLK 011: 4SCLK 100: 8SCLK 101: 16SCLK 110: 32SCLK 111: 64SCLK IO インタフェースモードで SCLK 出力の場合に有効なビットです。その他のモードでは意味を持ちません。 IO インタフェースモードで、ダブルバッファまたは FIFO が許可されている時に連続転送のインターバル時間を指定します。
0	-	R/W	"0" をライトしてください。

注 1) <TXE> ビットは、すべての設定を行った後に許可してください。

注 2) 送信中に動作を停止 (SCxMOD1<TXE> を "0" にクリア) しないでください。

注 3) SCxEN<SIOE>=0 (SIO 動作禁止)、または SCxMOD1<I2S0>=0 で IDLE モードへの移行 (IDLE モード中の動作禁止) した場合は、必ず SCxTFC の再設定を行ってください。

9.4.7 SCxMOD2 (モードコントロールレジスタ 2)

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

Bit	Bit Symbol	Type	機能											
31-8	–	R	リードすると "0" が読めます。											
7	TBEMP	R	送信バッファエンプティフラグ 0: Full 1: Empty ダブルバッファ不許可の場合はこのフラグは意味を持ちません。 送信ダブルバッファのデータ empty を示すフラグです。送信ダブルバッファのデータが送信シフトレジスタに移され empty になると "1" になり、送信データが書き込まれると "0" になります。											
6	RBFL	R	受信バッファ full フラグ 0: Empty 1: Full ダブルバッファ不許可の場合はこのフラグは意味を持ちません。 受信ダブルバッファのデータ full を示すフラグです。受信動作が終了して、受信シフトレジスタから受信ダブルバッファヘデータが格納されると "1" になり、読み出すと "0" になります。											
5	TXRUN	R	送信動作中フラグ 0: 停止 1: 動作 送信シフト動作中を示すステータスフラグです。 <TXRUN> と <TBEMP> ビットで以下のような状態を示します。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>&lt;TXRUN&gt;</th> <th>&lt;TBEMP&gt;</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>–</td> <td>送信動作中</td> </tr> <tr> <td rowspan="2">0</td> <td>1</td> <td>送信が完全に終了</td> </tr> <tr> <td>0</td> <td>送信バッファに次のデータがあり送信待ち</td> </tr> </tbody> </table>	<TXRUN>	<TBEMP>	状態	1	–	送信動作中	0	1	送信が完全に終了	0	送信バッファに次のデータがあり送信待ち
<TXRUN>	<TBEMP>	状態												
1	–	送信動作中												
0	1	送信が完全に終了												
	0	送信バッファに次のデータがあり送信待ち												
4	SBLEN	R/W	送信 STOP ビット長 (UART 用) 0: 1 ビット 1: 2 ビット UART モード時の送信 STOP ビットの長さを指定します。 受信の場合は設定に関わらず 1 ビットの STOP ビットで認識します。											
3	DRCHG	R/W	転送方向設定 0: LSB first 1: MSB first IO インタフェースモード時の転送方向を指定します。 UART モード時は LSB first に設定してください。											
2	WBUF	R/W	ダブルバッファの許可 0: 不許可 1: 許可 IO インタフェースモードの送信 (SCLK 出力 / 入力), 受信 (SCLK 出力), UART モードの送信時に、送信、受信のダブルバッファの許可 / 不許可を指定します。 IO インタフェースモードの受信 (SCLK 入力), UART モードの受信時は設定に関わらず、常にダブルバッファは許可されます。											
1-0	SWRST[1:0]	R/W	ソフトリセット "10"→"01" の順に書き込むことでソフトウェアリセットが発生します。 ソフトウェアリセットにより、以下のビットが初期化されます。また、送受信回路と FIFO は初期状態になります。(注1)(注2) <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>レジスタ名</th> <th>ビット</th> </tr> </thead> <tbody> <tr> <td>SCxMOD0</td> <td>&lt;RXE&gt;</td> </tr> <tr> <td>SCxMOD1</td> <td>&lt;TXE&gt;</td> </tr> <tr> <td>SCxMOD2</td> <td>&lt;TBEMP&gt;, &lt;RBFL&gt;, &lt;TXRUN&gt;</td> </tr> <tr> <td>SCxCR</td> <td>&lt;OERR&gt;, &lt;PERR&gt;, &lt;FERR&gt;</td> </tr> </tbody> </table>	レジスタ名	ビット	SCxMOD0	<RXE>	SCxMOD1	<TXE>	SCxMOD2	<TBEMP>, <RBFL>, <TXRUN>	SCxCR	<OERR>, <PERR>, <FERR>	
レジスタ名	ビット													
SCxMOD0	<RXE>													
SCxMOD1	<TXE>													
SCxMOD2	<TBEMP>, <RBFL>, <TXRUN>													
SCxCR	<OERR>, <PERR>, <FERR>													

注 1) 転送動作中にソフトリセットを実施する場合は 2 回連続して実行してください。

注 2) ソフトウェアリセット動作が完了するのに、命令実行後 2 クロック必要です。



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

ボーレートジェネレータの分周値は、下記の 2 つのレジスタで設定します。

SCxBRCR

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	BRADDE	BR0CK		BR0S			
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	-	R/W	"0" をライトしてください。
6	BRADDE	R/W	N + (16 - K) / 16 分周機能 (UART 用) 0: ディセーブル 1: イネーブル この機能は、UART モードのときのみ使用可能です。
5-4	BR0CK[1:0]	R/W	ボーレートジェネレータ入力クロック選択 00: φT1 01: φT4 10: φT16 11: φT64
3-0	BR0S[3:0]	R/W	分周値 "N" の設定 0000: 16 分周 0001: 1 分周 0010: 2 分周 : 1111: 15 分周

SCxBRADD

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	BR0K			
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	BR0K[3:0]	R/W	N + (16 - K) / 16 分周の K 値の設定 (UART 用) 0000: 設定禁止 0001: K = 1 0010: K = 2 : 1111: K = 15

ボーレートジェネレータ分周値の設定方法を表 9-2 にまとめます。

表 9-2 分周値の設定方法

	<BRADDE> = "0" のとき	<BRADDE> = "1" のとき (注 1) (UART モードのみ使用可能)
<BR0S> の設定	分周値 "N" を設定 (注 2) (注 3)	
<BR0K> の設定	設定不要	"K" 値を設定 (注 4)
分周値	N 分周	$N + \frac{(16 - K)}{16}$ 分周

注 1) N + (16 - K)/16 分周機能を使用する場合、必ず <BR0K> に "K" 値を設定後に <BRADDE> = "1" を設定してください。この機能は、UART モードのときのみ使用可能です。

注 2) UART モードで N + (16 - K)/16 分周機能を使用する場合、分周値 "N" に 1 分周 ("0001") と 16 分周 ("0000") は設定できません。

注 3) IO インタフェースモードの場合、分周値 "N" に 1 分周 ("0001") を設定できるのはダブルバッファを使用する場合のみです。

注 4) "K" 値に "0" を設定することはできません。

## 9.4.9 SCxFCNF (FIFO コンフィグレジスタ)

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

Bit	Bit Symbol	Type	機能						
31-8	-	R	リードすると "0" が読めます。						
7-5	-	R/W	必ず "000" を書き込んでください。						
4	RFST	R/W	受信 FIFO 使用バイト数 0: 最大 1: 受信 FIFO の FILL レベルに同じ 受信 FIFO 使用バイト数の設定ビットです。(注 1) "0" の場合、構成されている FIFO の最大のバイト数 (<CNFG> ビットの説明を参照。) が使用可能です。 "1" の場合、SCxRFC<RIL[1:0]> で指定された FILL レベルのバイト数になります。						
3	TFIE	R/W	送信 FIFO 使用時の送信割り込み許可 0: 禁止 1: 許可 送信 FIFO が有効にされている時の送信割り込みの禁止 / 許可を切り替えます。						
2	RFIE	R/W	受信 FIFO 使用時の受信割り込み許可 0: 禁止 1: 許可 受信 FIFO が有効にされている時の受信割り込みの禁止 / 許可を切り替えます。						
1	RXTXCNT	R/W	<RXE>/<TXE> の自動禁止 0: なし 1: 自動禁止 送信 / 受信の自動禁止機能の制御ビットです。 "1" に設定した場合、設定された通信方式により以下のように動作します。 <table border="1" style="margin-left: 20px;"> <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 &lt;RXE&gt;) が "0" となり、受信が禁止される。</td> </tr> <tr> <td>半二重送信</td> <td>送信 FIFO、送信バッファおよび送信シフトレジスタのすべてのデータ送信が終了すると自動的に送信許可ビット (SCxMOD1 &lt;TXE&gt;) が "0" となり、送信が禁止される。</td> </tr> <tr> <td>全二重</td> <td>上記の半二重受信または半二重送信のどちらかの条件が成立すると自動的に送信許可ビット、受信許可ビットとも "0" となり、送受信が禁止される。</td> </tr> </table>	半二重受信	受信シフトレジスタ、受信バッファおよび受信 FIFO 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。	半二重送信	送信 FIFO、送信バッファおよび送信シフトレジスタのすべてのデータ送信が終了すると自動的に送信許可ビット (SCxMOD1 <TXE>) が "0" となり、送信が禁止される。	全二重	上記の半二重受信または半二重送信のどちらかの条件が成立すると自動的に送信許可ビット、受信許可ビットとも "0" となり、送受信が禁止される。
半二重受信	受信シフトレジスタ、受信バッファおよび受信 FIFO 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。								
半二重送信	送信 FIFO、送信バッファおよび送信シフトレジスタのすべてのデータ送信が終了すると自動的に送信許可ビット (SCxMOD1 <TXE>) が "0" となり、送信が禁止される。								
全二重	上記の半二重受信または半二重送信のどちらかの条件が成立すると自動的に送信許可ビット、受信許可ビットとも "0" となり、送受信が禁止される。								
0	CNFG	R/W	FIFO の許可 0: 禁止 1: 許可 FIFO 使用の許可ビットです。(注 2) "1" に設定すると FIFO が使用可能となります。設定された通信方式により FIFO の構成は以下のようになります。 <table border="1" style="margin-left: 20px;"> <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>	半二重受信	受信 FIFO 4 バイト	半二重送信	送信 FIFO 4 バイト	全二重	受信 FIFO 2 バイト + 送信 FIFO 2 バイト
半二重受信	受信 FIFO 4 バイト								
半二重送信	送信 FIFO 4 バイト								
全二重	受信 FIFO 2 バイト + 送信 FIFO 2 バイト								

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

注 2) 9 ビット UART モードでは FIFO は使用できません。

9.4.10 SCxRFC (受信 FIFO コンフィグレジスタ)

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

Bit	Bit Symbol	Type	機能															
31-8	-	R	リードすると "0" が読めます。															
7	RFCS	W	受信 FIFO クリア (注) 1: クリア "1" を書き込むと受信 FIFO がクリアされ、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: fill レベル到達 1: fill レベル到達および新規データ読み出し時に fill レベルを超えている															
5-2	-	R	リードすると "0" が読めます。															
1-0	RIL[1:0]	R/W	受信割り込みが発生する FIFO の fill レベル <table border="1" style="margin-left: 20px;"> <thead> <tr> <th></th> <th>半二重</th> <th>全二重</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>4 バイト</td> <td>2 バイト</td> </tr> <tr> <td>01</td> <td>1 バイト</td> <td>1 バイト</td> </tr> <tr> <td>10</td> <td>2 バイト</td> <td>2 バイト</td> </tr> <tr> <td>11</td> <td>3 バイト</td> <td>1 バイト</td> </tr> </tbody> </table>		半二重	全二重	00	4 バイト	2 バイト	01	1 バイト	1 バイト	10	2 バイト	2 バイト	11	3 バイト	1 バイト
	半二重	全二重																
00	4 バイト	2 バイト																
01	1 バイト	1 バイト																
10	2 バイト	2 バイト																
11	3 バイト	1 バイト																

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

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

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

Bit	Bit Symbol	Type	機能															
31-8	-	R	リードすると "0" が読めます。															
7	TFCS	W	送信 FIFO クリア (注 1) 1: クリア "1" を書き込むと送信 FIFO がクリアされ、SCxTST<TLVL>="000" となります。また、ライトポイントも初期化されます。 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: fill レベル到達 1: fill レベル到達および新規データ書き込み時に fill レベルに達していない															
5-2	-	R	リードすると "0" が読めます。															
1-0	TIL[1:0]	R/W	送信割り込みが発生する FIFO の fill レベル <table border="1" style="margin-left: 20px;"> <thead> <tr> <th></th> <th>半二重</th> <th>全二重</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>Empty</td> <td>Empty</td> </tr> <tr> <td>01</td> <td>1 バイト</td> <td>1 バイト</td> </tr> <tr> <td>10</td> <td>2 バイト</td> <td>Empty</td> </tr> <tr> <td>11</td> <td>3 バイト</td> <td>1 バイト</td> </tr> </tbody> </table>		半二重	全二重	00	Empty	Empty	01	1 バイト	1 バイト	10	2 バイト	Empty	11	3 バイト	1 バイト
	半二重	全二重																
00	Empty	Empty																
01	1 バイト	1 バイト																
10	2 バイト	Empty																
11	3 バイト	1 バイト																

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

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

- ・ SCxEN<SIOE>="0" (SIO 動作停止)
- ・ SCxMOD1<I2S0>="0" (IDLE モード時動作禁止) 設定で、WFI 命令による低消費電力モードへの遷移からの復帰後

## 9.4.12 SCxRST (受信 FIFO ステータスレジスタ)

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

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

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

9.4.13 SCxTST (送信 FIFO ステータスレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	TUR	-	-	-	-	TLVL		
リセット後	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> ビットはバッファレジスタ (SCxBUF) に書き込みを行うと "0" にクリアされます。



## 9.5 動作モード

表 9-3 にモードとデータフォーマットをまとめます。

表 9-3 モードとデータフォーマット

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

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

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

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

モード 1 とモード 2 はパリティビットの付加が可能です。モード 3 は、マスタコントローラが、シリアルリンク (マルチコントローラシステム) でスレーブコントローラを起動させるためのウエイクアップ機能を有しています。

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

## 9.6 データフォーマット

### 9.6.1 データフォーマット一覧

図 9-2 にデータフォーマットを示します。

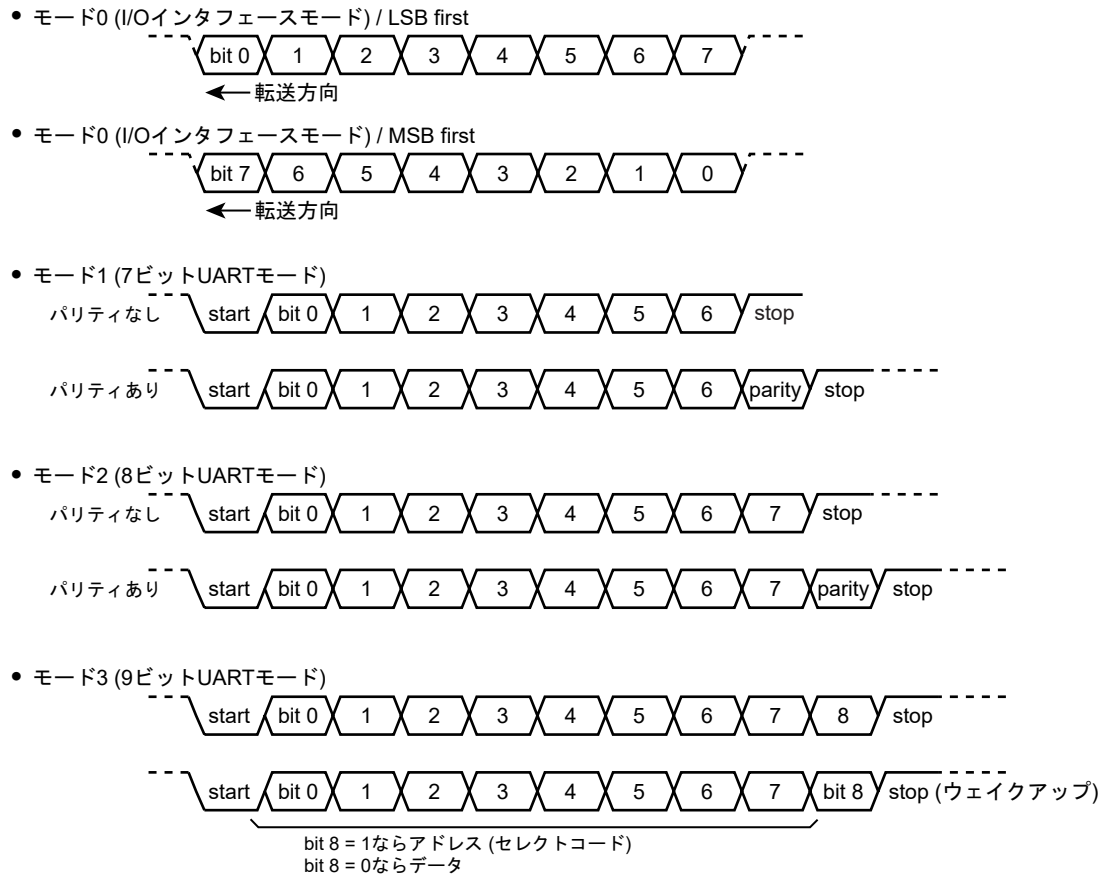


図 9-2 データフォーマット

## 9.6.2 パリティ制御

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

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

### 9.6.2.1 送信

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

### 9.6.2.2 受信

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

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

## 9.6.3 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。

---

## 9.7 クロック制御

### 9.7.1 プリスケーラ

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

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

プリスケーラは、 $SCxMOD0\langle SC[1:0]\rangle = "01"$ でポ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。

ポーレートジェネレータへの入力クロック分解能を、下表に示します。

表 9-4 ボーレートジェネレータへの入力クロック分解能  $f_c = 80 \text{ MHz}$

ペリフェラル クロック選択 CGSYSCR <FPSEL>	クロック ギア値 CGSYSCR <GEAR[2:0]>	プリスケアラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケアラ出力クロック 分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
0 (fgear)	000 (fc)	000 (fperiph/1)	$f_c/2^1$ (0.025 $\mu\text{s}$ )	$f_c/2^3$ (0.1 $\mu\text{s}$ )	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )
		001 (fperiph/2)	$f_c/2^2$ (0.05 $\mu\text{s}$ )	$f_c/2^4$ (0.2 $\mu\text{s}$ )	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )
		010 (fperiph/4)	$f_c/2^3$ (0.1 $\mu\text{s}$ )	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )
		011 (fperiph/8)	$f_c/2^4$ (0.2 $\mu\text{s}$ )	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )
		100 (fperiph/16)	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )
		101 (fperiph/32)	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )
	100 (fc/2)	000 (fperiph/1)	$f_c/2^2$ (0.05 $\mu\text{s}$ )	$f_c/2^4$ (0.2 $\mu\text{s}$ )	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )
		001 (fperiph/2)	$f_c/2^3$ (0.1 $\mu\text{s}$ )	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )
		010 (fperiph/4)	$f_c/2^4$ (0.2 $\mu\text{s}$ )	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )
		011 (fperiph/8)	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )
		100 (fperiph/16)	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )
		101 (fperiph/32)	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )	$f_c/2^{13}$ (102.4 $\mu\text{s}$ )
	101 (fc/4)	000 (fperiph/1)	$f_c/2^3$ (0.1 $\mu\text{s}$ )	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )
		001 (fperiph/2)	$f_c/2^4$ (0.2 $\mu\text{s}$ )	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )
		010 (fperiph/4)	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )
		011 (fperiph/8)	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )
		100 (fperiph/16)	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )	$f_c/2^{13}$ (102.4 $\mu\text{s}$ )
		101 (fperiph/32)	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )	$f_c/2^{14}$ (204.8 $\mu\text{s}$ )
	110 (fc/8)	000 (fperiph/1)	$f_c/2^4$ (0.2 $\mu\text{s}$ )	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )
		001 (fperiph/2)	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )
		010 (fperiph/4)	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )
		011 (fperiph/8)	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )	$f_c/2^{13}$ (102.4 $\mu\text{s}$ )
		100 (fperiph/16)	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )	$f_c/2^{14}$ (204.8 $\mu\text{s}$ )
		101 (fperiph/32)	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )	$f_c/2^{13}$ (102.4 $\mu\text{s}$ )	$f_c/2^{15}$ (409.6 $\mu\text{s}$ )
111 (fc/16)	000 (fperiph/1)	$f_c/2^5$ (0.4 $\mu\text{s}$ )	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )	
	001 (fperiph/2)	$f_c/2^6$ (0.8 $\mu\text{s}$ )	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )	
	010 (fperiph/4)	$f_c/2^7$ (1.6 $\mu\text{s}$ )	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )	$f_c/2^{13}$ (102.4 $\mu\text{s}$ )	
	011 (fperiph/8)	$f_c/2^8$ (3.2 $\mu\text{s}$ )	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )	$f_c/2^{14}$ (204.8 $\mu\text{s}$ )	
	100 (fperiph/16)	$f_c/2^9$ (6.4 $\mu\text{s}$ )	$f_c/2^{11}$ (25.6 $\mu\text{s}$ )	$f_c/2^{13}$ (102.4 $\mu\text{s}$ )	$f_c/2^{15}$ (409.6 $\mu\text{s}$ )	
	101 (fperiph/32)	$f_c/2^{10}$ (12.8 $\mu\text{s}$ )	$f_c/2^{12}$ (51.2 $\mu\text{s}$ )	$f_c/2^{14}$ (204.8 $\mu\text{s}$ )	$f_c/2^{16}$ (819.2 $\mu\text{s}$ )	

表 9-4 ポーレートジェネレータへの入力クロック分解能  $f_c = 80 \text{ MHz}$

ペリフェラル クロック選択 CGSYSCR <FPSEL>	クロック ギア値 CGSYSCR <GEAR[2:0]>	プリスケアラ クロック選択 CGSYSCR <PRCK[2:0]>	プリスケアラ出力クロック 分解能			
			$\phi T1$	$\phi T4$	$\phi T16$	$\phi T64$
1 (fc)	000 (fc)	000 (fperiph/1)	$fc/2^1$ (0.025 $\mu\text{s}$ )	$fc/2^3$ (0.1 $\mu\text{s}$ )	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.05 $\mu\text{s}$ )	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.1 $\mu\text{s}$ )	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )	$fc/2^{11}$ (25.6 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )	$fc/2^{12}$ (51.2 $\mu\text{s}$ )
	100 (fc/2)	000 (fperiph/1)	-	$fc/2^3$ (0.1 $\mu\text{s}$ )	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )
		001 (fperiph/2)	$fc/2^2$ (0.05 $\mu\text{s}$ )	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.1 $\mu\text{s}$ )	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )	$fc/2^{11}$ (25.6 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )	$fc/2^{12}$ (51.2 $\mu\text{s}$ )
	101 (fc/4)	000 (fperiph/1)	-	$fc/2^3$ (0.1 $\mu\text{s}$ )	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )
		010 (fperiph/4)	$fc/2^3$ (0.1 $\mu\text{s}$ )	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )	$fc/2^{11}$ (25.6 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )	$fc/2^{12}$ (51.2 $\mu\text{s}$ )
	110 (fc/8)	000 (fperiph/1)	-	-	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )
		001 (fperiph/2)	-	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )
		010 (fperiph/4)	-	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )
		011 (fperiph/8)	$fc/2^4$ (0.2 $\mu\text{s}$ )	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )
		100 (fperiph/16)	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )	$fc/2^{11}$ (25.6 $\mu\text{s}$ )
		101 (fperiph/32)	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )	$fc/2^{12}$ (51.2 $\mu\text{s}$ )
111 (fc/16)	000 (fperiph/1)	-	-	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	
	001 (fperiph/2)	-	-	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	
	010 (fperiph/4)	-	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )	
	011 (fperiph/8)	-	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )	
	100 (fperiph/16)	$fc/2^5$ (0.4 $\mu\text{s}$ )	$fc/2^7$ (1.6 $\mu\text{s}$ )	$fc/2^9$ (6.4 $\mu\text{s}$ )	$fc/2^{11}$ (25.6 $\mu\text{s}$ )	
	101 (fperiph/32)	$fc/2^6$ (0.8 $\mu\text{s}$ )	$fc/2^8$ (3.2 $\mu\text{s}$ )	$fc/2^{10}$ (12.8 $\mu\text{s}$ )	$fc/2^{12}$ (51.2 $\mu\text{s}$ )	

注 1) プリスケアラ出力クロック  $\phi Tn$  は、必ず  $\phi Tn \leq fsys/2$  を満足するように ( $\phi Tn$  が  $fsys$  よりも遅くなるように) 選択してください。

注 2) SIO 動作中はクロックギアの切り替えは行わないでください。

注 3) 表中 "-" は設定禁止、"\*" は Don't Care です。

## 9.7.2 シリアルクロック生成回路

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

### 9.7.2.1 ボーレートジェネレータ

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

#### (1) ボーレートジェネレータ入力クロック

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

#### (2) ボーレートジェネレータ出力クロック

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

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

以下に設定可能な分周値をまとめます。

モード	分周機能設定 SCxBRCR<BRADDE>	N 分周値 SCxBRCR<BR0S>	K 値 SCxBRADD<BR0K>
IO インタフェース	N 分周	1 ~ 16 (注)	-
UART	N 分周	1 ~ 16	-
	$N + (16-K)/16$ 分周	2 ~ 15	1 ~ 15

注) 1 分周は、ダブルバッファ許可時のみ使用できます。

## 9.7.2.2 クロック選択回路

モードとレジスタ設定により、クロックが選択されます。

モードは SCxMOD0<SM> で指定します。

IO インタフェースモード時のクロックは、SCxCR で設定します。

UART モード時のクロックは、SCxMOD0<SC> で設定します。

## (1) IO インタフェースモードの転送クロック

表 9-5 に IO インタフェースモードで可能なクロックを示します。

表 9-5 IO インタフェースモードのクロック選択

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

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

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

- クロック / モード制御部の設定

- fc = 80MHz
- fgear = 80MHz (CGSYSCR<GEAR[2:0]> = "000" : fc 選択)
- $\Phi T0$  = 80MHz (CGSYSCR<PRCK[2:0]> = "000" : 1 分周)

- SIO の設定 (ダブルバッファ使用の場合)

- クロック選択 (SCxBRCR<BRCK[1:0]> = "00" :  $\Phi T1$  選択) = 40MHz
- 分周値 (SCxBRCR<BRS[3:0]> = "0001" : 1 分周) = 40MHz

ダブルバッファ使用の場合、1 分周が選択できます。ポーレートは 40MHz が 2 分周され、20Mbps となります。

- SIO の設定 (ダブルバッファ未使用の場合)

- クロック選択 (SCxBRCR<BRCK[1:0]> = "00" :  $\Phi T1$  選択) = 40MHz
- 分周値 (SCxBRCR<BRS[3:0]> = "0010" : 2 分周) = 20MHz

ダブルバッファ未使用の場合は、2 分周が最速になります。ポーレートは 20MHz が 2 分周され、10Mbps となります。



SCLK 入力を使用する場合、以下の条件を満足する必要があります。

- ダブルバッファ使用の場合

- SCLK 周期 > 6/fsys

最高ボーレートは、 $80 \div 6 = 13.3$  Mbps 未満となります。

- ダブルバッファ未使用の場合

- SCLK 周期 > 8/fsys

最高ボーレートは、 $80 \div 8 = 10$  Mbps 未満となります。

## (2) UART モードの転送クロック

表 9-6 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 9-6 UART モードのクロック選択

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

それぞれのクロックでのボーレート例を示します。

- ボーレートジェネレータを使用する場合

- fc = 80MHz

- fgear = 80MHz (CGSYSCR<GEAR[2:0]> = "000" : fc 選択)

- $\Phi T0 = 80$ MHz (CGSYSCR<PRCK[2:0]> = "000" : 1 分周)

- クロック選択 =  $\Phi T1 = 40$ MHz (SCxBRCR<BRCK[1:0]> = "00" :  $\Phi T1$  選択)

最高ボーレートは 40MHz が 16 分周され、2.5Mbps となります。

表 9-7 に以下のクロック設定でボーレートジェネレータを使用する場合のボーレート例を示します。

- fc = 9.8304MHz

- fgear = 9.8304MHz (CGSYSCR<GEAR[2:0]> = "000" : fc 選択)

- $\Phi T0 = 4.9152$ MHz (CGSYSCR<PRCK[2:0]> = "001" : 2 分周)

表 9-7 UART モードのボーレート例 (ボーレートジェネレータ使用)

fc [MHz]	分周値 N (SCxBRCR<BRS[3:0]>)	φT1 (fc/4)	φT4 (fc/16)	φT16 (fc/64)	φT64 (fc/256)
9.830400	2	76.800	19.200	4.800	1.200
	4	38.400	9.600	2.400	0.600
	8	19.200	4.800	1.200	0.300
	16	9.600	2.400	0.600	0.150

単位 : kbps

• SCLK 入力を使用する場合

SCLK 入力を使用する場合、以下の条件を満足する必要があります。

- SCLK 周期 > 2/fsys

最高ボーレートは、 $80 \div 2 \div 16 = 2.5$  Mbps 未満にする必要があります。

• fsys を使用する場合

fsys の最高が 80MHz ですので、最高ボーレートは、 $80 \div 16 = 5$  Mbps となります。

• タイマ出力を使用する場合

タイマの出力を使用する場合、カウンタと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

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

↑
↑
(タイマプリスケークロックφT1(2分周)を選択した場合)  
(タイマフリップフロップ反転2回で1クロック周期となる)

表 9-8 に以下のクロック設定でタイマ出力を使用する場合のボーレート例を示します。

- fc = 80MHz / 9.8304MHz / 8MHz
- fgear = 80MHz / 9.8304MHz / 8MHz (CGSYSCR<GEAR[2:0]> = "000" : fc 選択)
- φT0 = 40MHz / 4.9152MHz / 4MHz (CGSYSCR<PRCK[2:0]> = "001" : 2 分周)
- タイマカウントクロック  
= 4MHz / 1.2287MHz / 1MHz (TBxMOD<TBCLK[1:0]> = "01" : φT1 選択)

表 9-8 UART モードのボーレート例 ( タイマ出力使用 )

TBxRG 設定	fc		
	80MHz	9.8304MHz	8MHz
0x0001	625.0	76.8	62.5
0x0002	312.5	38.4	31.25
0x0003	-	25.6	-
0x0004	156.25	19.2	15.625
0x0005	125.0	15.36	12.5
0x0006	-	12.8	-
0x0008	78.125	9.6	-
0x000A	62.5	7.68	6.25
0x0010	39.025	4.8	-
0x0014	31.25	3.84	3.125

単位 : kbps

## 9.8 送信 / 受信バッファと FIFO

### 9.8.1 構成

送信 / 受信バッファと FIFO の構成を図 9-3 に示します。

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

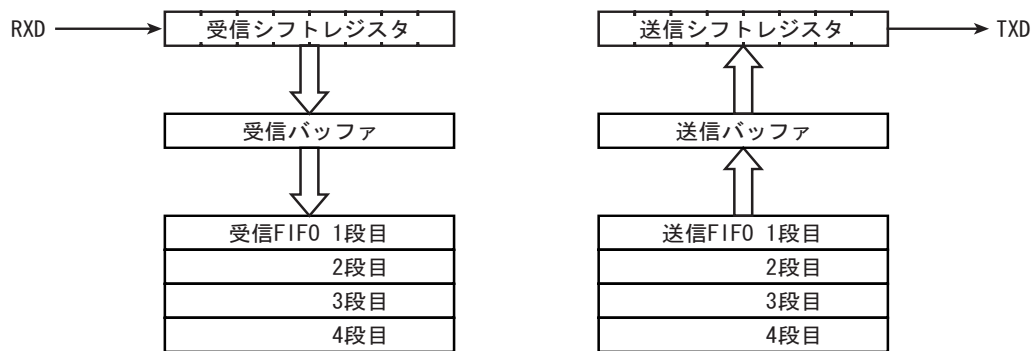


図 9-3 バッファと FIFO の構成

### 9.8.2 送信 / 受信バッファ

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

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

表 9-9 にモードとバッファ構成の関係をまとめます。

表 9-9 モードとバッファ構成

モード		SCxMOD2<WBUF>	
		"0"	"1"
UART	送信	シングル	ダブル
	受信	ダブル	ダブル
IO インタフェース (SCLK 入力)	送信	シングル	ダブル
	受信	ダブル	ダブル
IO インタフェース (SCLK 出力)	送信	シングル	ダブル
	受信	シングル	ダブル

### 9.8.3 FIFO

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

FIFO を有効にするには SCxMOD2<WBUF> を "1" としてダブルバッファをイネーブルにし、SCxFCNF<CNFG> に "1" をセットします。FIFO バッファの構成は SCxMOD1<FDPX[1:0]> 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 9-10 にモードと FIFO 構成の関係をまとめます。

表 9-10 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

## 9.9 ステータスフラグ

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

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

<TBEMP> は、送信バッファ empty を示すフラグです。送信バッファから送信シフトレジスタへデータが移されると、"1" がセットされます。送信バッファにデータをセットすると "0" にクリアされます。

## 9.10 エラーフラグ

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

これらのフラグは、SCxCR レジスタを読み出すと "0" にクリアされます。

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

### 9.10.1 OERR フラグ

UART モード、IO インタフェースモード共に、受信バッファのデータを読み出す前に次のフレームの受信が終了すると "1" にセットされます。受信 FIFO を有効にしている場合は、受信 FIFO ヘ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。

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

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

### 9.10.2 PERR フラグ

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

パリティエラーは UART モードで受信したデータから生成されたパリティと受信したパリティが異なる場合に "1" にセットされます。

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

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

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

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

### 9.10.3 FERR フラグ

フレーミングエラーはUARTモードの受信データのストップビットを中央付近でサンプリングし、結果が "0" の場合に発生します。SCxMOD2<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。

IO インタフェースモードではこのビットは "0" 固定です。

## 9.11 受信

### 9.11.1 受信カウンタ

受信カウンタは4ビットのバイナリカウンタで、SIOCLKでカウントアップされます。

UARTモードでは、データ1ビットの受信にSIOCLKが16クロック用いられ、7、8、9発目でデータをサンプリングします。3度のデータサンプリングによる多数決論理により受信データを判断しています。

### 9.11.2 受信制御部

#### 9.11.2.1 IOインタフェースモードの場合

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

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

#### 9.11.2.2 UARTモードの場合

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

### 9.11.3 受信動作

#### 9.11.3.1 受信バッファの動作

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

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



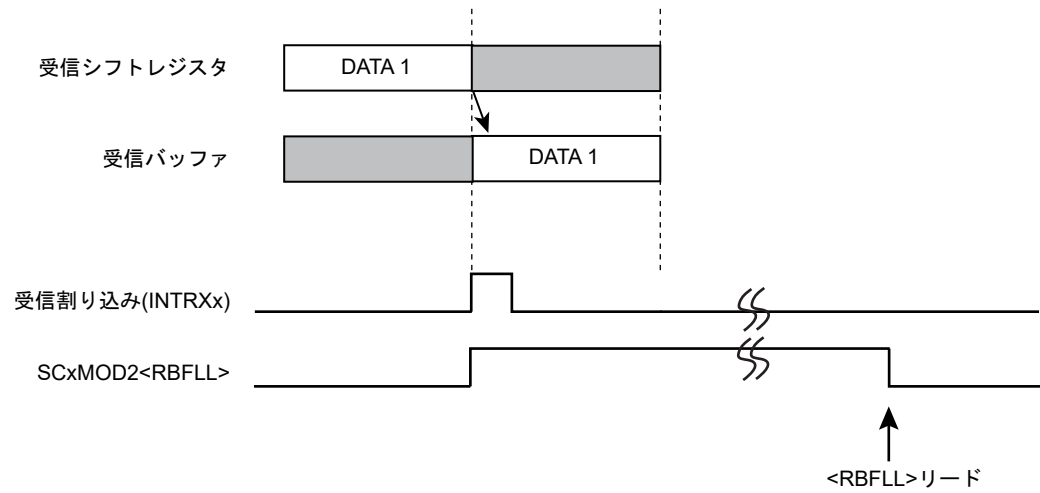


図 9-4 受信バッファの動作

### 9.11.3.2 受信 FIFO の動作

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

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

以下に、半二重受信の設定と動作を示します。

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

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

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

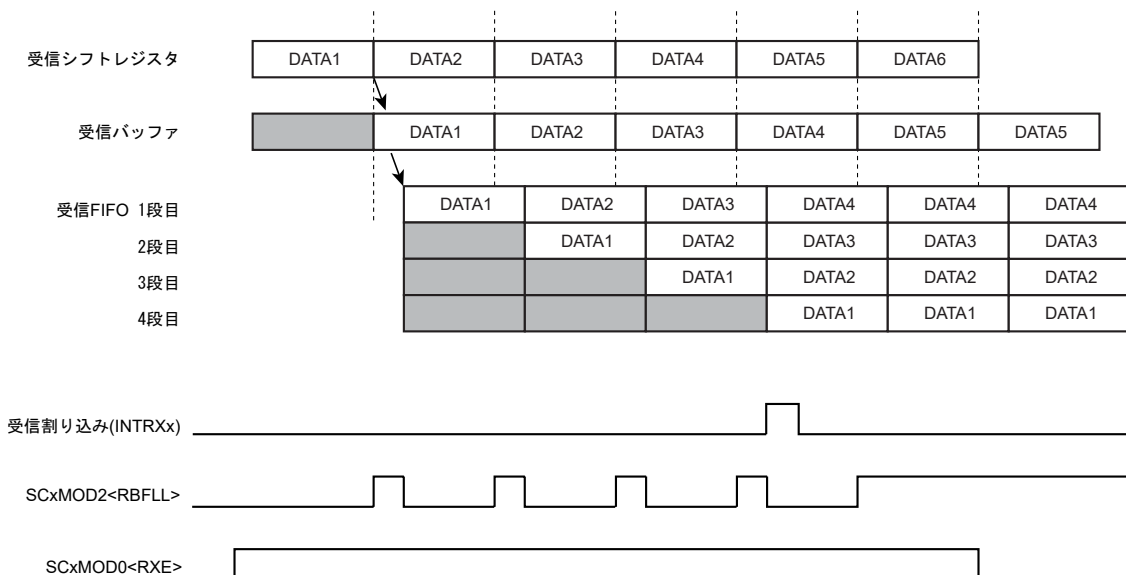


図 9-5 受信 FIFO の動作

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

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

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

#### (1) シングルバッファの場合

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

#### (2) ダブルバッファの場合

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

#### (3) FIFO の場合

受信シフトレジスタ、受信バッファ、FIFO すべてにデータが格納されると SCLK 出力を停止します。1 データ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、SCLK 出力を再開します。

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

### 9.11.3.4 受信データの読み出し

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

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

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

### 9.11.3.5 ウェイクアップ機能

9 ビット UART モードの場合、ウェイクアップ機能 SCxMOD0<WU> を "1" にすることによって、スレーブコントローラのウェイクアップ動作が可能で、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。

### 9.11.3.6 オーバランエラー

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

受信 FIFO が有効にされている場合は受信 FIFO が full になり、受信バッファに次のデータが移される前に受信 FIFO を読み出さないと、受信 FIFO のオーバランが発生してオーバランフラグがセットされます。この場合でも受信 FIFO のデータは保存されます。

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

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

## 9.12 送信

### 9.12.1 送信カウンタ

送信カウンタは4ビットのバイナリカウンタで、受信カウンタ同様 SIOCLK でカウントされます。UART モードでは、16クロックごとに送信クロック (TXDCLK) を生成します。

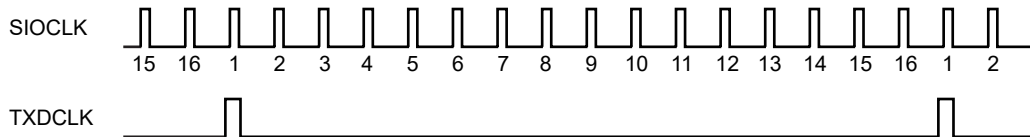


図 9-6 UART モード送信クロックの生成

### 9.12.2 送信制御部

#### 9.12.2.1 IO インタフェースモードの場合

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

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

#### 9.12.2.2 UART モードの場合

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

### 9.12.3 送信動作

#### 9.12.3.1 送信バッファの動作

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

ダブルバッファ有効の場合（送信 FIFO が有効な場合も含む）、送信バッファへ書き込まれたデータは送信シフトレジスタに転送されます。同時に送信割り込み INTTX<sub>x</sub> が発生し、送信バッファエンプティフラグ (SC<sub>x</sub>MOD2<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。

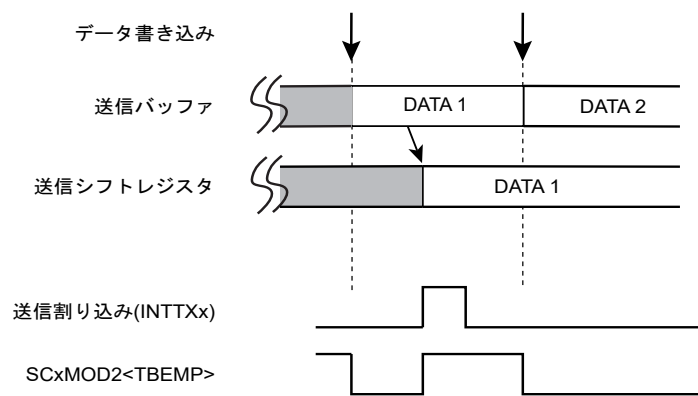


図 9-7 送信バッファの動作 (ダブルバッファ有効時)

#### 9.12.3.2 送信 FIFO の動作

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

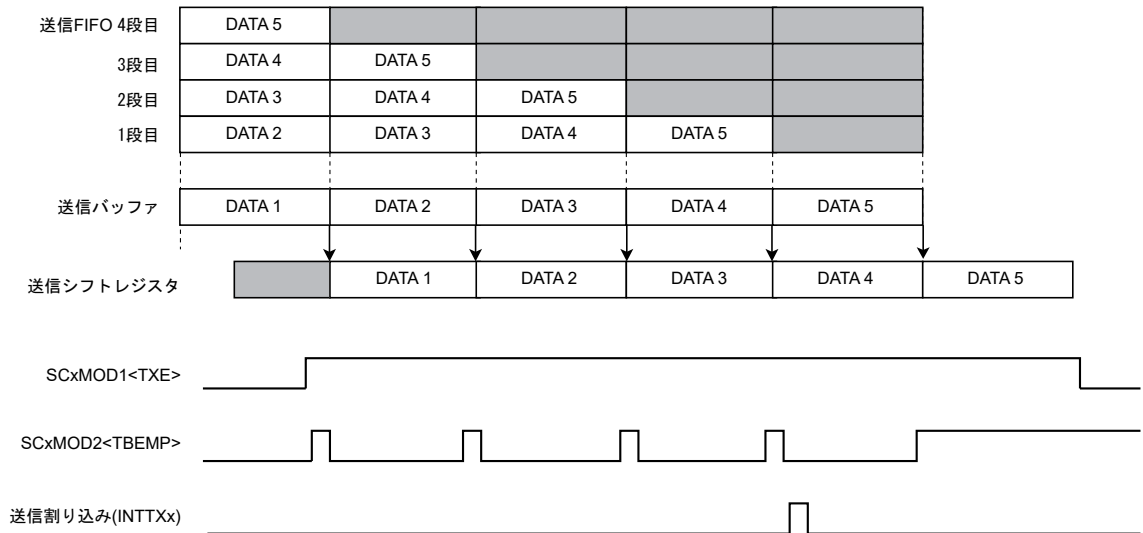
注) 送信 FIFO バッファ使用時は、SIO の転送モード設定 (半二重 / 全二重)、FIFO 許可 (SC<sub>x</sub>FCNF <CNFG>="1") の後、必ず送信 FIFO クリアを実行して下さい。

以下に、4 バイトのデータを半二重送信する場合の設定と動作を示します。

SC <sub>x</sub> MOD1[6:5] = 10	: 転送モードを半二重送信に設定
SC <sub>x</sub> FCNF[4:0] = 11011	: FIFO が空になると送信を自動的に禁止 受信 FIFO の使用バイト数は割り込み発生 fill レベルと同じ
SC <sub>x</sub> TFC[1:0] = 00	: 割り込み発生時の fill レベル を 0 に設定
SC <sub>x</sub> TFC[7:6] = 11	: 送信 FIFO のクリアと割り込み発生条件の設定
SC <sub>x</sub> FCNF[0] = 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。



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

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

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

#### (1) シングルバッファの場合

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

#### (2) ダブルバッファの場合

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

#### (3) FIFO の場合

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

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

#### 9.12.3.4 アンダーランエラー

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

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

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



## 9.13 ハンドシェイク機能

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

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

- 注 1) 送信中に  $\overline{\text{CTS}}$  信号を立ち上げた場合、送信終了後に停止します。(図 9-9 "a")  
 注 2)  $\overline{\text{CTS}}$  信号立下り後の最初の TXDCLK クロックから送信を開始します。(図 9-9 "b")

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。

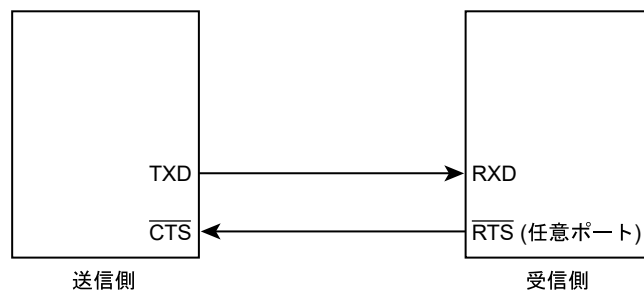


図 9-8 ハンドシェイク機能接続

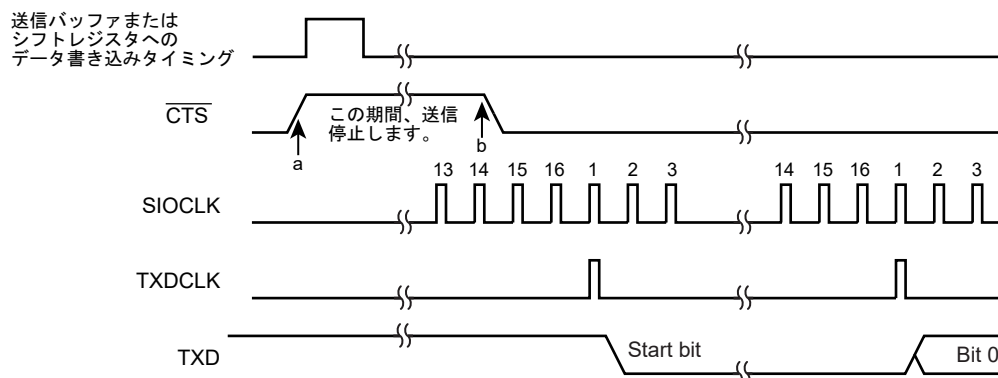


図 9-9  $\overline{\text{CTS}}$  信号のタイミング

## 9.14 割り込み / エラー発生タイミング

### 9.14.1 受信割り込み

受信動作のデータの流と読み出しの経路を図 9-10 に示します。

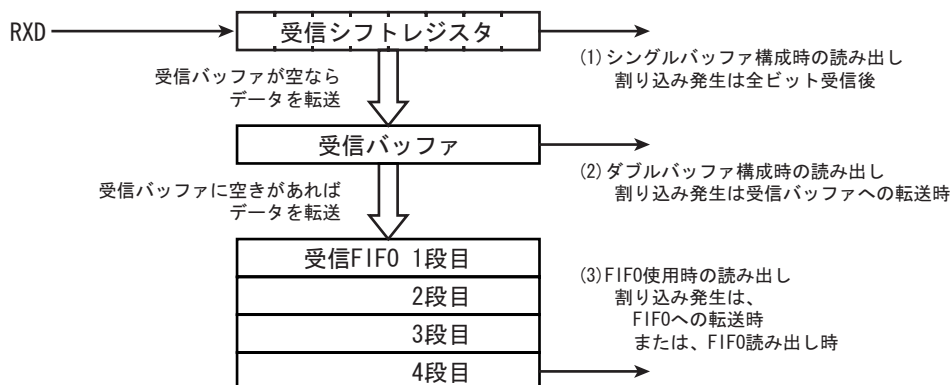


図 9-10 受信バッファ / FIFO 構成図

#### 9.14.1.1 シングルバッファ / ダブルバッファ構成の場合

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

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

注) オーバーランエラー成立時は割り込みは発生しません。

#### 9.14.1.2 FIFO 使用の場合

FIFO 使用の場合の受信割り込みは、表9-11 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 9-11 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

9.14.2 送信割り込み

送信動作のデータの流と読み出しの経路を図 9-11 に示します。

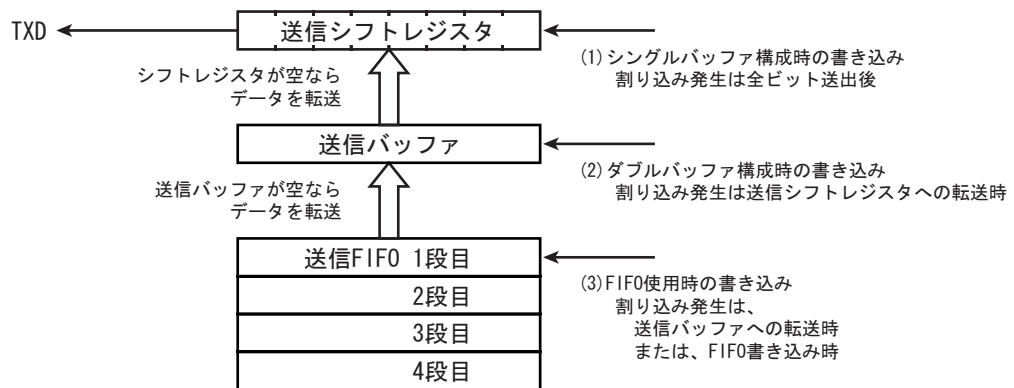


図 9-11 送信バッファ /FIFO 構成図

9.14.2.1 シングルバッファ / ダブルバッファ構成の場合

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

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

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

9.14.2.2 FIFO 使用の場合

FIFO 使用の場合の送信割り込みは、表 9-12 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 9-12 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

### 9.14.3 エラー発生

#### 9.14.3.1 UART モード

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

#### 9.14.3.2 IO インタフェースモード

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

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

## 9.15 ソフトリセット

SCxMOD2<SWRST[1:0]> に "10" → "01" の順でライトすることによりソフトウェアリセットが発生します。これにより、SCxMOD0<RXE>、SCxMOD1<TXE>、SCxMOD2<TBEMP><RBFL><TXRUN>、SCxCR<OERR> <PERR> <FERR> が初期化され、送受信回路と FIFO は初期状態になります。その他の状態は保持されます。

## 9.16 モード別動作説明

### 9.16.1 モード 0 (IO インタフェースモード)

このモードには、同期クロック SCLK を出力する SCLK 出力モードと、外部より同期クロック SCLK を入力する SCLK 入力モードがあります。以下に FIFO の使用が禁止されている状態での各動作の説明を行います。FIFO の動作については、前述の受信 FIFO 動作および送信 FIFO 動作の項を参照してください。

#### 9.16.1.1 送信

##### (1) SCLK 出力モード

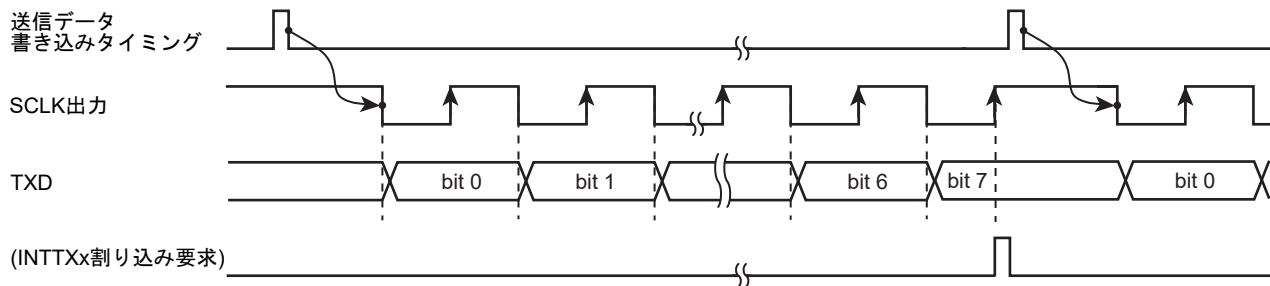
- ダブルバッファ不許可 (SCxMOD2<WBUF> = "0") の場合

送信バッファにデータを書き込むたびに、データが TXD 端子から、クロックが SCLK 端子より出力されます。データがすべて出力されると割り込み (INTTX<sub>x</sub>) が発生します。

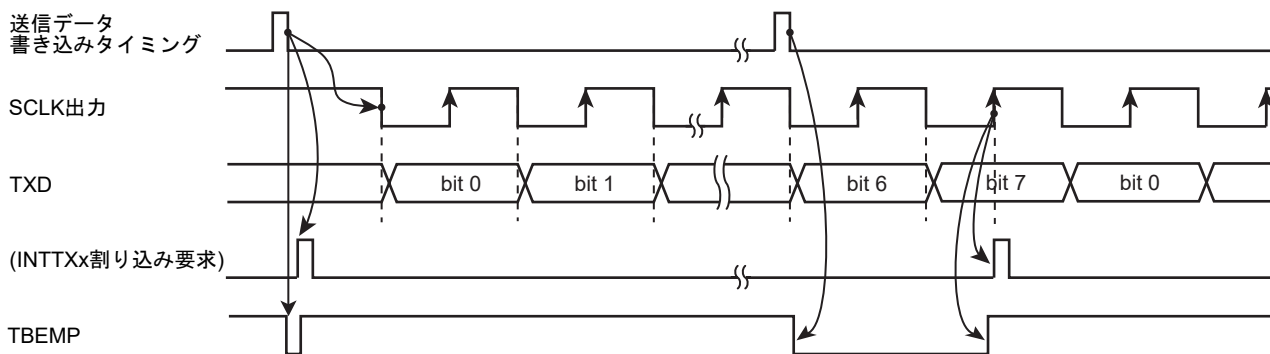
- ダブルバッファ許可 (SCxMOD2<WBUF> = "1") の場合

送信停止の状態ですべて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 (INTTX<sub>x</sub>) が発生します。

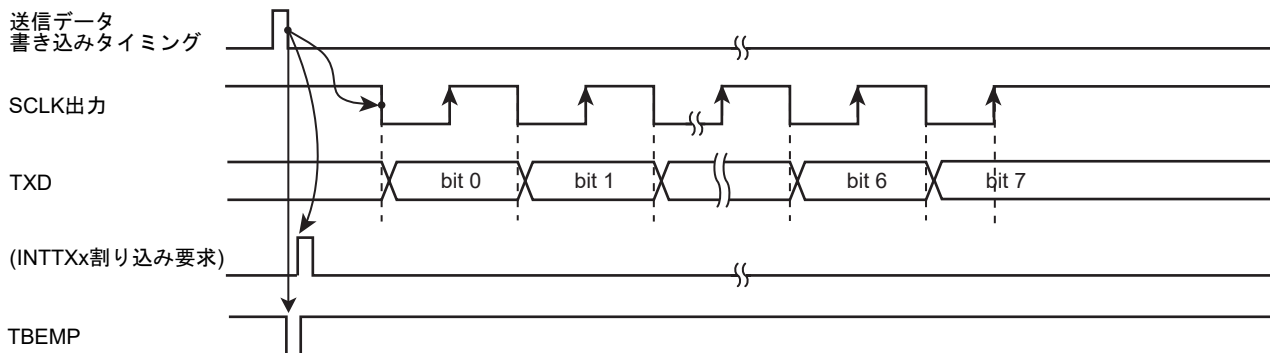
シフトレジスタのデータ送出終了時に送信バッファにデータが存在しない場合は、割り込み (INTTX<sub>x</sub>) を発生せず、SCLK 出力も停止します。



<WBUF> = "0" (ダブルバッファ不許可)の場合



<WBUF> = "1" (ダブルバッファ許可)の場合(バッファデータがある場合)



<WBUF> = "1" (ダブルバッファ許可)の場合(バッファにデータがない場合)

図 9-12 IO インタフェースモード送信動作 (SCLK 出力モード)

## (2) SCLK 入力モード

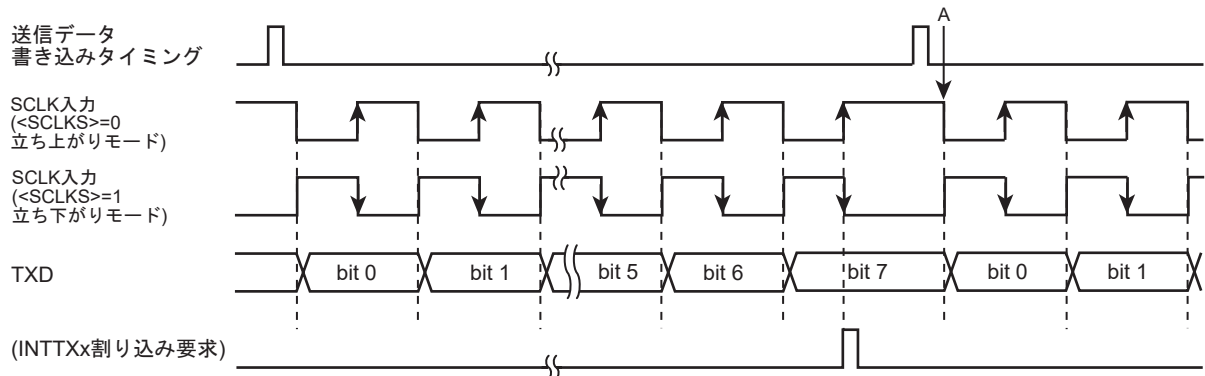
- ダブルバッファ不許可 (SCxMOD2<WBUF> = "0") の場合

送信バッファにデータが書き込まれている状態で SCLK が入力されると、8 ビットのデータが TXD 端子より出力されます。データがすべて出力されると割り込み INTTXx が発生します。次の送信データは図 9-13 に示す A 点までに書き込んでください。

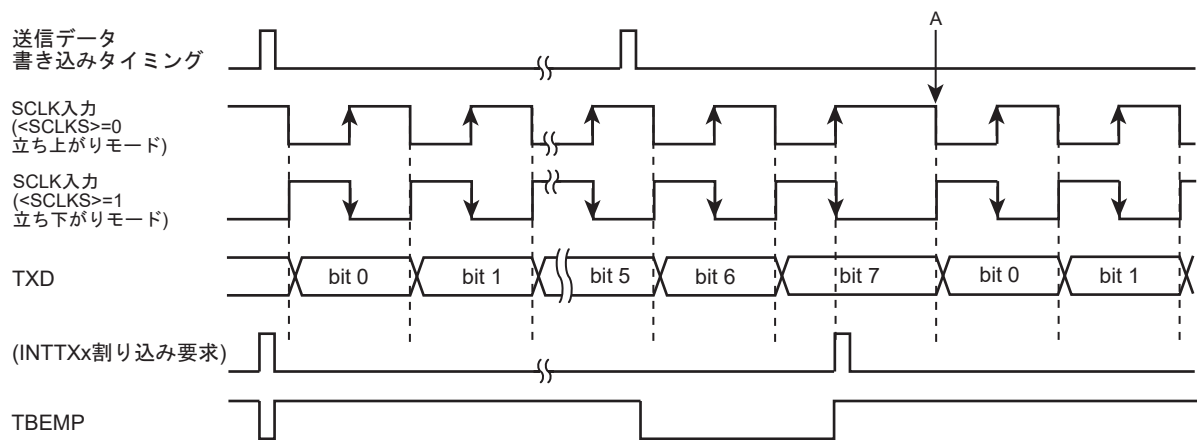
- ダブルバッファ許可 (SCxMOD2<WBUF> = "1") の場合

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

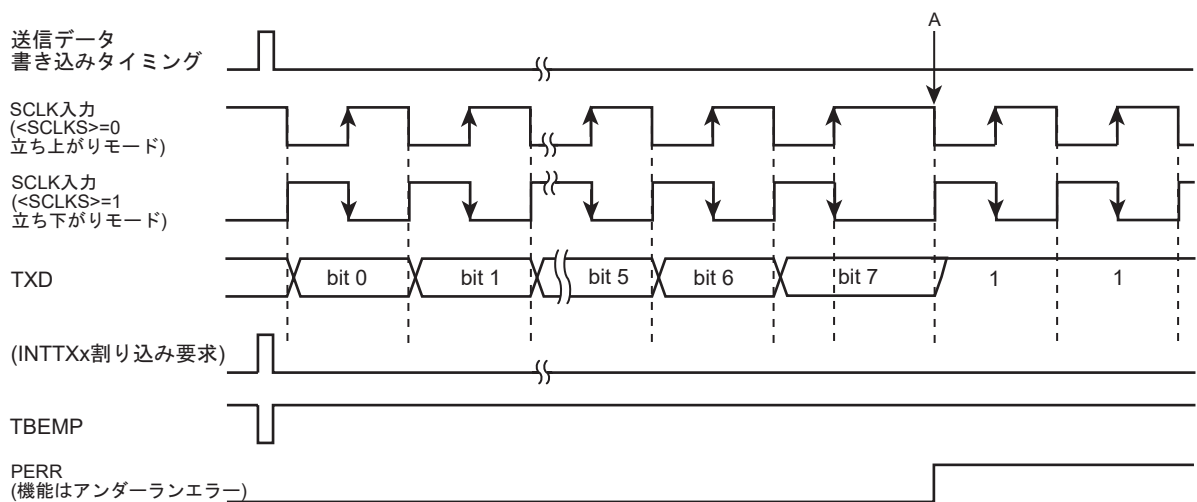
送信バッファにデータが書き込まれていない状態で、SCLK が入力された場合、内部ビット数カウンタはカウントを開始しますが、アンダーランエラーがセットされ、8 ビット分のダミーデータ (0xFF) を送出します。



<WBUF> = "0" (ダブルバッファ不許可)の場合



<WBUF> = "1" (ダブルバッファ許可)の場合(バッファにデータがある場合)



<WBUF> = "1" (ダブルバッファ許可)の場合(バッファにデータがない場合)

図 9-13 IO インタフェースモード送信動作 (SCLK 入力モード)



### 9.16.1.2 受信

#### (1) SCLK 出力モード

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

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

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

- ダブルバッファ許可 (SCxMOD2<WBUF> = "1") の場合

シフトレジスタに格納されたデータは受信バッファに移され、続けて次のフレームを受信することができます。シフトレジスタから受信バッファにデータが移されると、受信バッファ full フラグ SCxMOD2<RBFL> が "1" にセットされ、割り込み INTRX<sub>x</sub> 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 INTRX<sub>x</sub> は発生せず、SCLK 出力は停止します。この状態で受信バッファのデータを読み出すと、シフトレジスタのデータを受信バッファに移し、割り込み INTRX<sub>x</sub> を発生して受信を再開します。

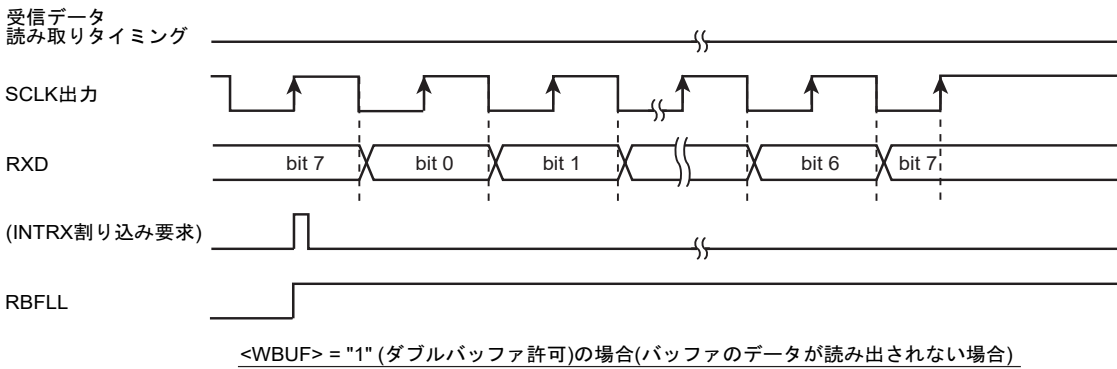
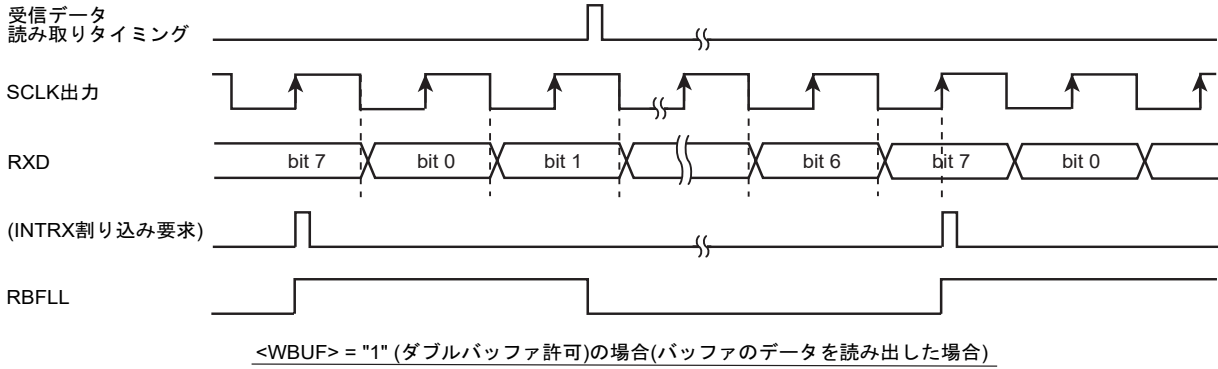
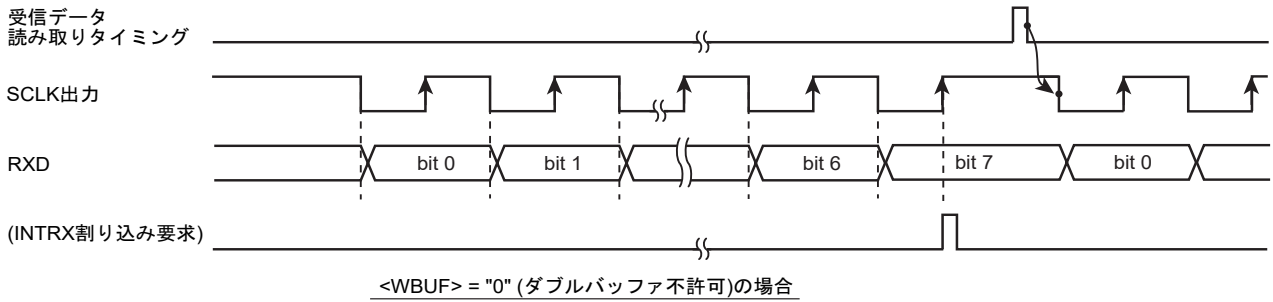


図 9-14 IO インタフェースモード受信動作 (SCLK 出力モード)

(2) SCLK 入力モード

SCLK 入力モードでは常に受信ダブルバッファが許可されており、受信したフレームはシフトレジスタから受信バッファに移され、連続して次のフレームを受信することができます。

受信データが受信バッファへ移されるごとに受信割り込み INTRX<sub>x</sub> が発生します。

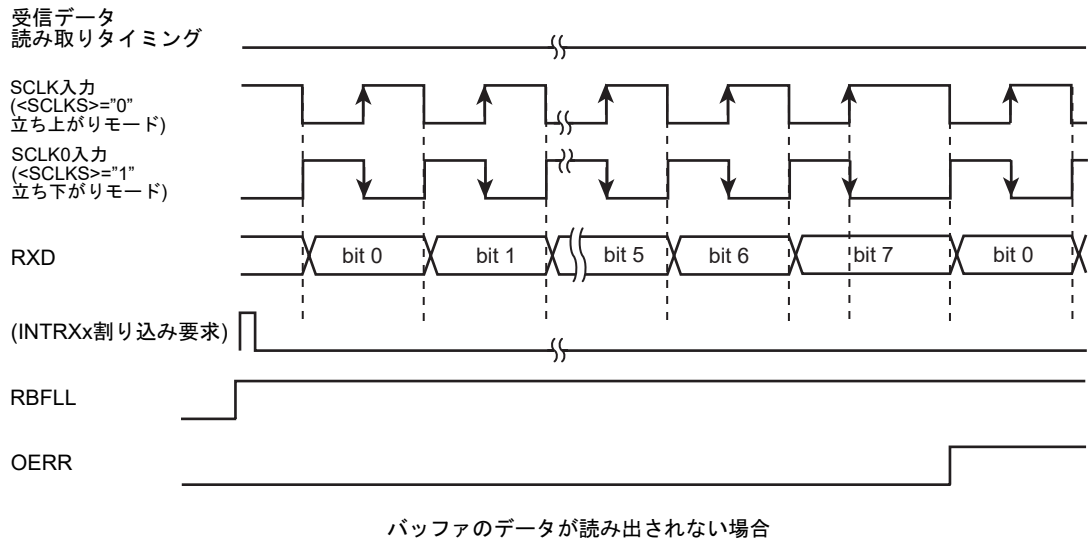
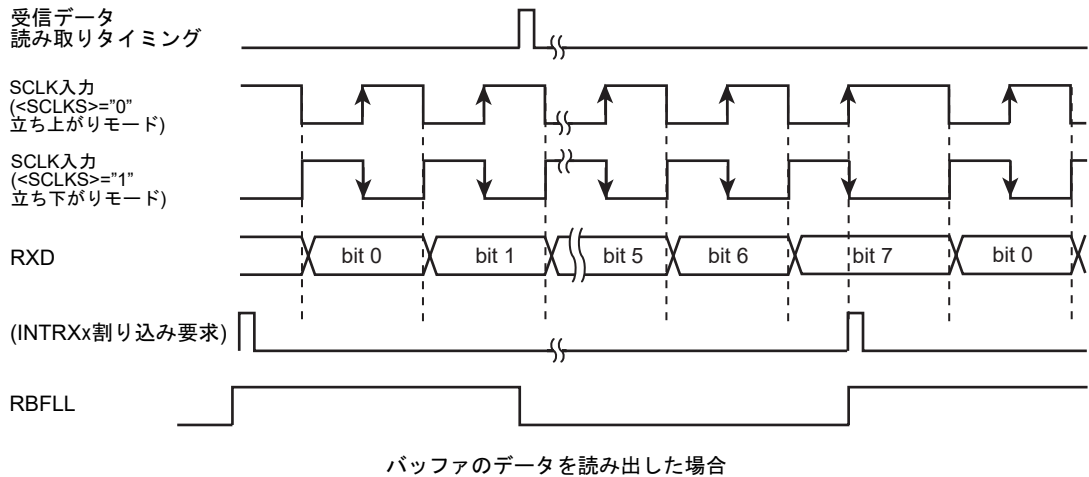


図 9-15 IO インタフェースモード受信動作 (SCLK 入力モード)

## 9.16.1.3 送受信 (全二重)

## (1) SCLK 出力モード

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

送信バッファにデータを書き込むと SCLK 出力を開始します。

SCLK の出力により 8 ビットデータが受信シフトレジスタにシフトインされ、受信割り込み (INTRX<sub>x</sub>) が発生します。それと並行して送信バッファに書き込まれた 8 ビットデータが、TXD 端子より出力され、すべてのデータが送信されると送信割り込み (INTTX<sub>x</sub>) が発生します。この状態で SCLK の出力は停止します。

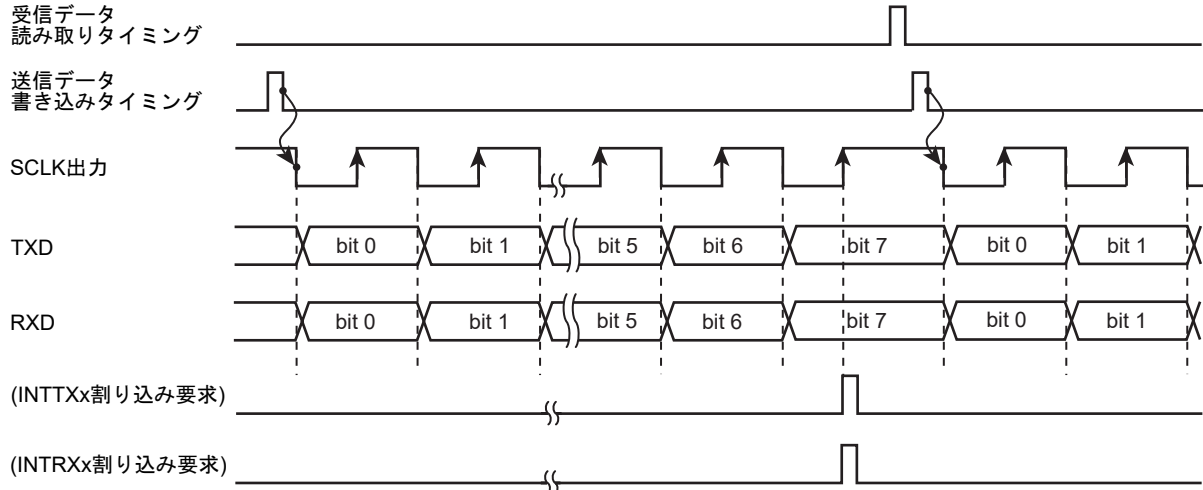
受信バッファの読み出しと送信バッファへのデータ書き込みを行うと次の送受信が開始されます。受信バッファの読み出しと送信バッファの書き込み順番は任意です。両方の条件が成立した場合に再開されます。

- ダブルバッファ許可 (SCxMOD2<WBUF> = "1") の場合

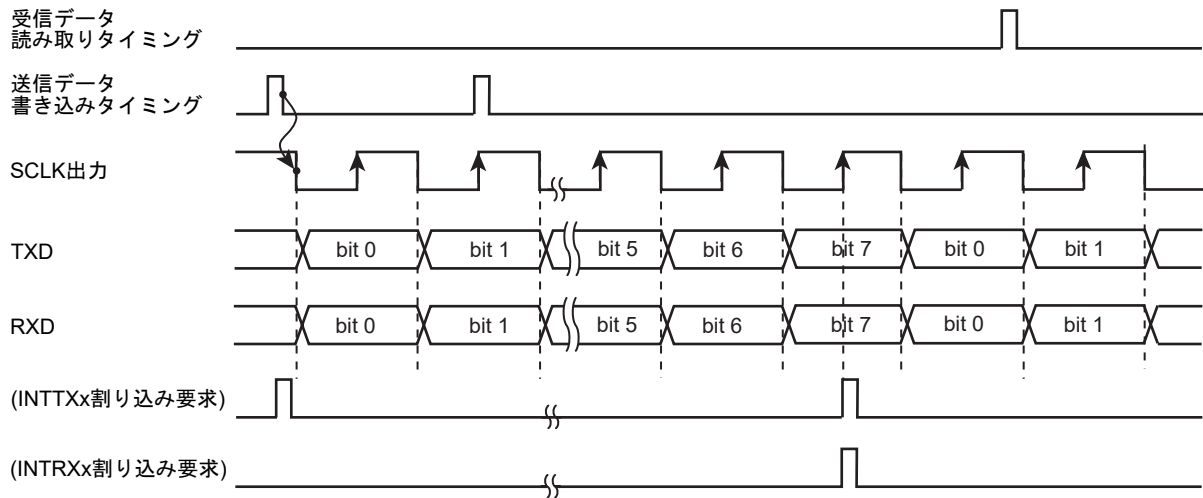
送信バッファにデータを書き込むと SCLK 出力を開始します。

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

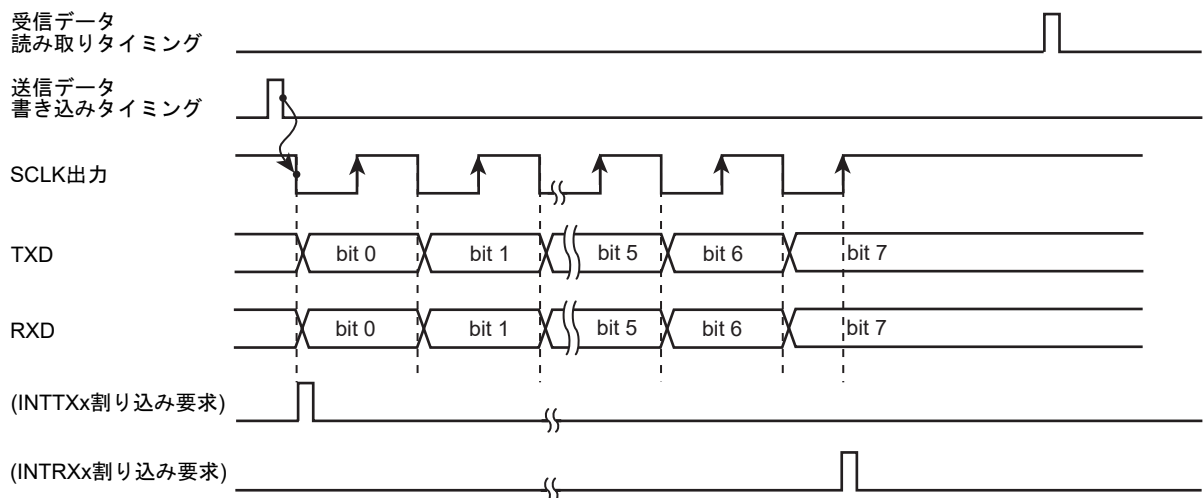
この時、送信バッファに移すデータが存在しない (SCxMOD2 <TBEMP> = "1") または受信バッファにデータが存在している (SCxMOD2 <RBFL> = "1") 場合は SCLK 出力が停止します。その後は受信データの読み出しと送信データの書き込みの両方の条件が成立すると再度 SCLK の出力が開始されて次の送受信が始まります。



<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



<WBUF> = "1" (ダブルバッファ許可)の場合

図 9-16 IO インタフェースモード送受信動作 (SCLK 出力モード)

## (2) SCLK 入力モード

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

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

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

次のフレームの SCLK が入力される前にデータを送信バッファへ書き込むようにしてください。(図 9-17 に示す A 点までに書き込んでください)。受信データは、次のフレームの受信が終了する前に読み出してください。

- ダブルバッファ許可 (SCxMOD2<WBUF> = "1") の場合

送信シフトレジスタのデータの送信が終了すると、送信バッファのデータが送信シフトレジスタへ移されると同時に割り込み (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。

次のフレームの SCLK が入力される前に送信データを送信バッファへ書き込むようにしてください。(図 9-17 に示す A 点までに書き込んでください)。受信データは、次のフレームの受信が終了する前に読み出してください。

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

フレームの最終ビットの受信までに受信バッファのデータが読み出されていない場合はオーバーランエラーが発生します。また、次のフレームの SCLK 入力までに送信バッファへ転送データが書き込まれていない場合はアンダーランエラーが発生します。

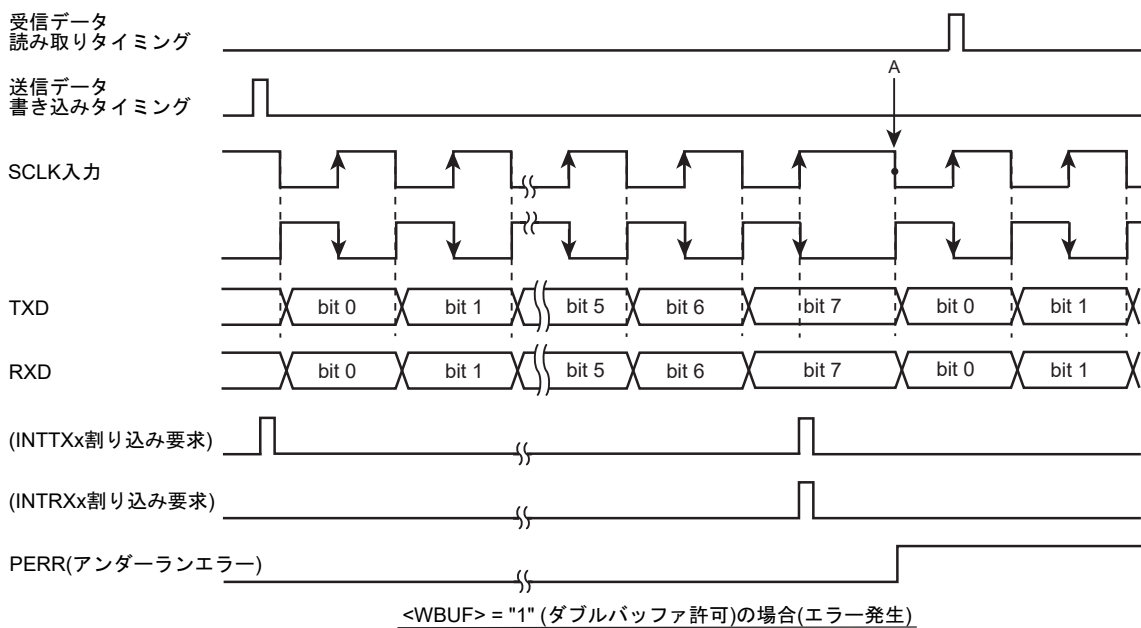
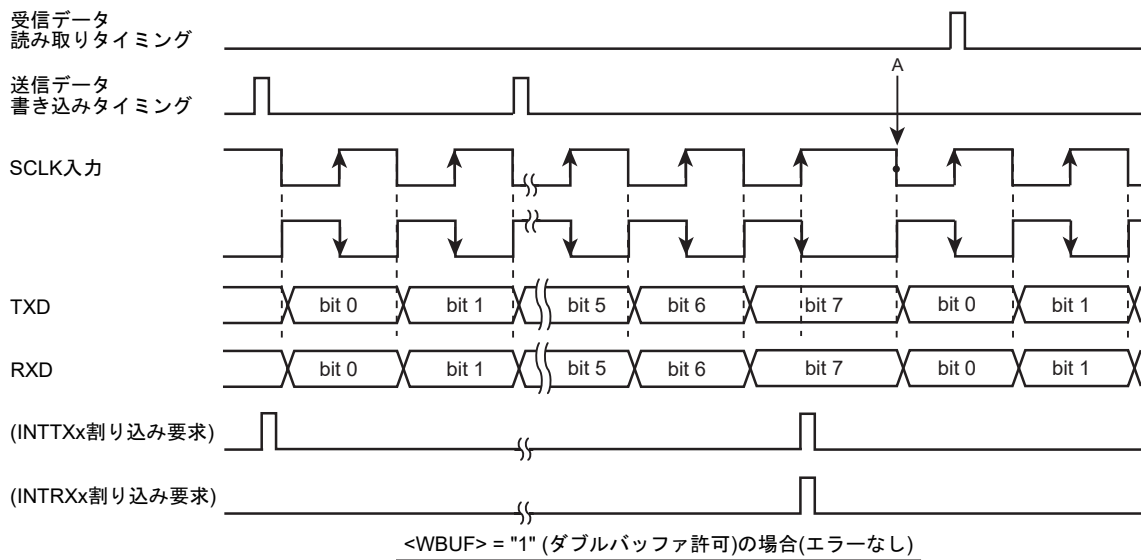
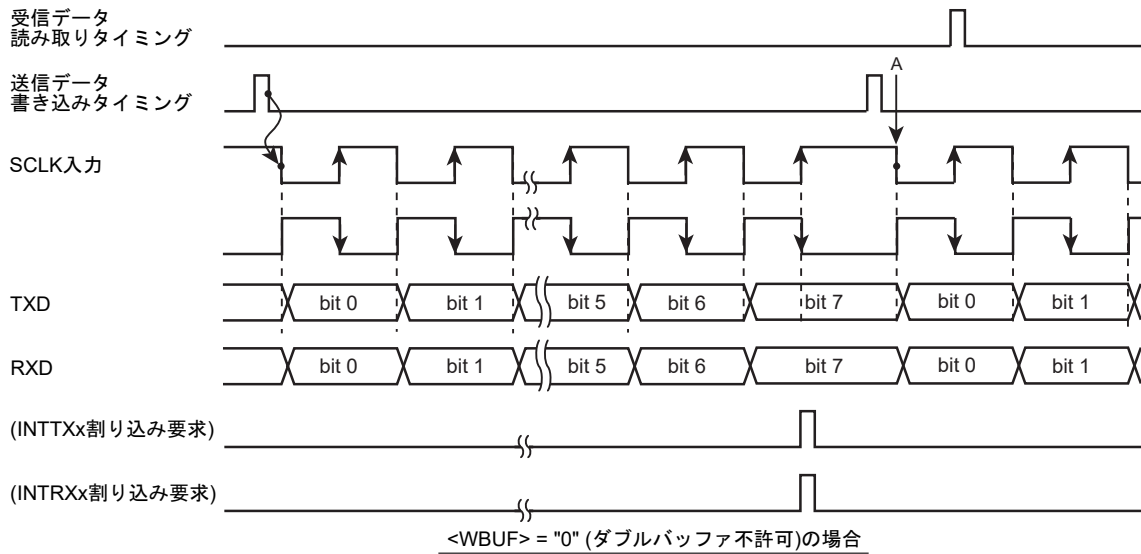


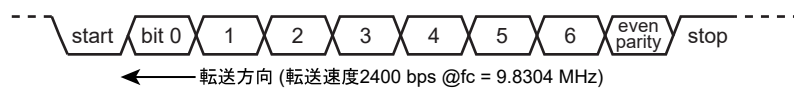
図 9-17 IO インタフェースモード送受信動作 (SCLK 入力モード)

### 9.16.2 モード 1 (7 ビット UART モード)

モードコントロールレジスタ (SCxMOD0 <SM[1: 0]>) を "01" にセットすると 7 ビット UART モードになります。

このモードではパリティビットの付加が可能で、コントロールレジスタ (SCxCR <PE>) でパリティビット付加のイネーブル / ディセーブルを制御しています。<PE> = "1" (イネーブル) のときは、SCxCR<EVEN> で偶数パリティ / 奇数パリティを選択できます。STOP ビットの長さは SCxMOD2<SBLEN> で指定することができます。

下記フォーマットのデータを送信する場合の各コントロールレジスタの設定を示します。



クロック条件	システムクロック :		高速 (fc)						
	高速クロックギア :		1 倍 (fc)						
	プリスケールクロック :		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

### 9.16.3 モード 2 (8 ビット UART モード)

SCxMOD0 <SM[1:0]> を "10" にセットすると 8 ビット UART モードになります。このモードでは、パリティビットの付加が可能で SCxCR <PE> でパリティビット付加のイネーブル / ディセーブルを制御できます。<PE> = "1" (イネーブル) のとき、SCxCR <EVEN> で偶数パリティ / 奇数パリティの選択も可能です。

下記のフォーマットのデータを受信する場合の各コントロールレジスタの設定を示します。



クロック条件	システムクロック :		高速 (fc)						
	高速クロックギア :		1 倍 (fc)						
	プリスケールクロック :		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

### 9.16.4 モード3 (9ビットUARTモード)

SCxMOD0 <SM[1:0]> を "11" にセットすると9ビットUARTモードになります。このモードでは、パリティビットの付加を禁止 (SCxCR<PE> = "0") してください。

最上位ビット (9ビット目) は、送信の場合 SCxMOD0 <TB8> に書き込み、受信の場合 SCxCR <RB8> に格納されます。また、バッファに対する書き込み、読み出しは必ず最上位ビットの方を先に行い、SCxBUFの方を後にします。STOPビットの長さは SCxMOD2 <SBLLEN> で指定することができます。

#### 9.16.4.1 ウェイクアップ機能

9ビットUARTモードでは、ウェイクアップ機能制御ビット SCxMOD0 <WU> を "1" にすることによって、スレーブコントローラのウェイクアップ動作が可能で、SCxCR <RB8> = "1" のときのみ割り込み (INTRXx) が発生します。

注) スレーブコントローラの TXD 端子は、必ず PxOD レジスタを設定してオープンドレイン出力モードにしてください。

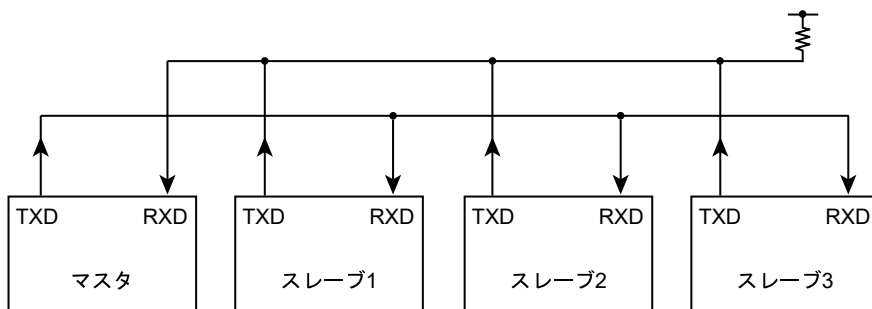
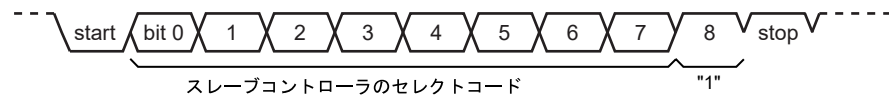


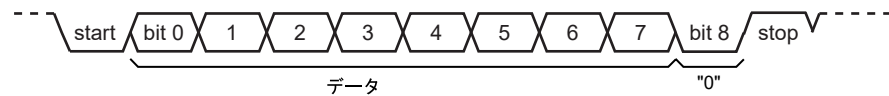
図 9-18 ウェイクアップ機能によるシリアルリンク

## 9.16.4.2 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  $SC_xMOD0<WU>$  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 (INTRX<sub>x</sub>) 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 10 章 シリアルバスインタフェース (I2C/SIO)

シリアルバスインタフェースを 1 チャンネル内蔵しています。シリアルバスインタフェースは、下記の 2 つの動作モードを持っています。

- I2C バスモード (マルチマスタ)
- クロック同期式 8 ビット SIO モード

I2C バスモードでは、SCL および SDA を通して外部デバイスと接続されます。

SIO モードでは、SCK, SI, SO を通して外部デバイスと接続されます。

チャンネルごとの使用端子とポートの設定は、下記のとおりです。

表 10-1 シリアルバスインタフェース使用時のポート設定

チャンネル	動作モード	端子	ポート ファンクション レジスタ設定	ポート 出力コントロール レジスタ設定	ポート 入力コントロール レジスタ設定	ポート オープンドレイン コントロール レジスタ設定
SBI	I2C バスモード	SCL :PN1 SDA :PN0	PNFR1[1:0] = 11	PNCR[1:0] = 11	PNIE[1:0] = 11	PNOD[1:0] = 11
	SIO モード	SCK :PN2 SI :PN1 SO :PN0	PNFR1[2:0] = 111	PNCR[2:0] = 101(SCK0 出力) PNCR[2:0] = 001(SCK0 入力)	PNIE[2:0] = 010(SCK0 出力) PNIE[2:0] = 110(SCK0 入力)	PNOD[2:0] = xxx

注) x: Don't care

## 10.1 構成

構成を図 10-1 に示します。

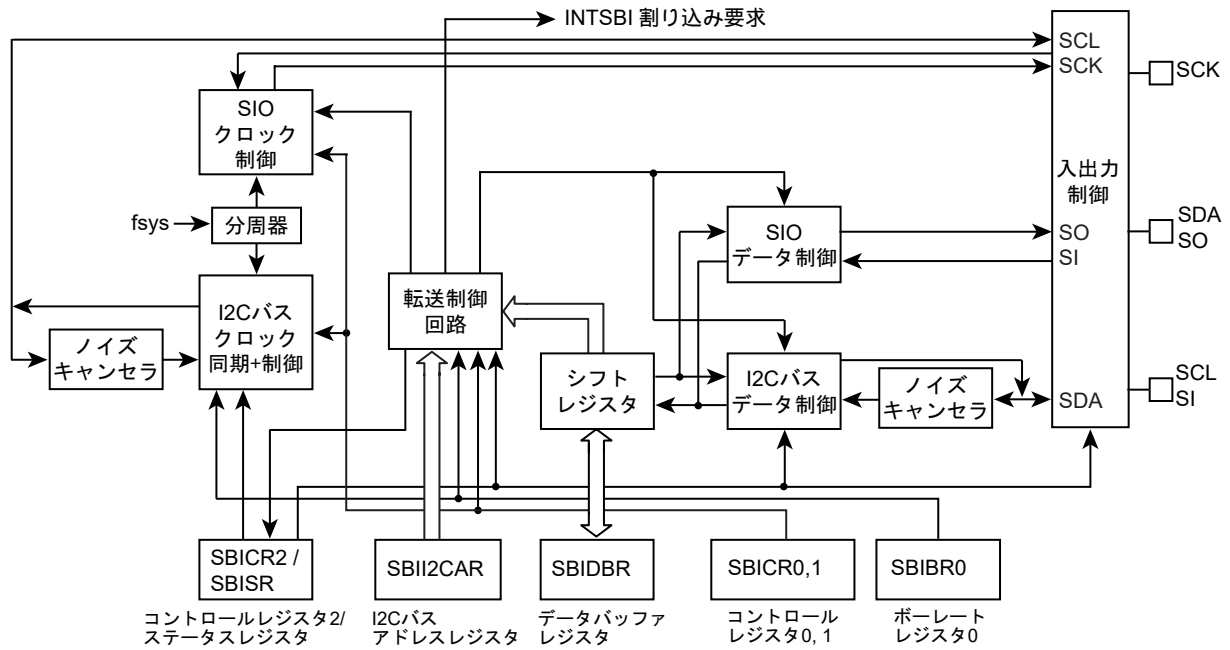


図 10-1 シリアルバスインタフェースブロック図

## 10.2 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「10.4 I2C バスモード時のコントロールレジスタ」および「10.7 SIO モード時のコントロールレジスタ」を参照してください。

### 10.2.1 チャンネル別レジスタ一覧

各チャンネルのレジスタとアドレスを示します。

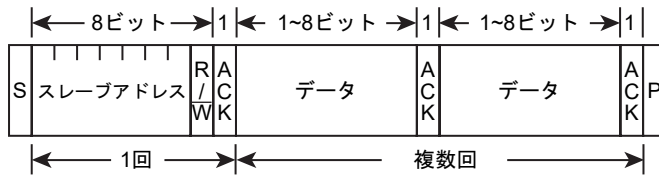
Base Address = 0x4002\_0000

レジスタ名		Address(Base+)
コントロールレジスタ 0	SBICR0	0x0000
コントロールレジスタ 1	SBICR1	0x0004
データバッファレジスタ	SBIDBR	0x0008
I2C バスアドレスレジスタ	SBII2CAR	0x000C
コントロールレジスタ 2	SBICR2(ライト時)	0x0010
ステータスレジスタ	SBISR(リード時)	
ポーレートレジスタ 0	SBIBR0	0x0014

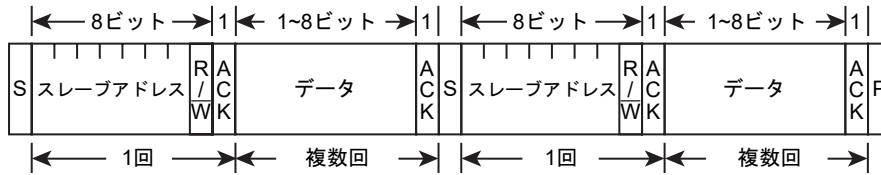
### 10.3 I2C バスモード時のデータフォーマット

I2C バスモード時のデータフォーマットを図 10-2 に示します。

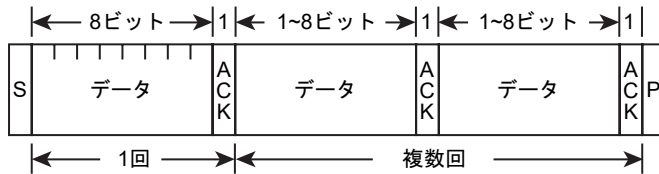
(a) アドレッシングフォーマット



(b) アドレッシングフォーマット (再スタートあり)



(c) フリーデータフォーマット (マスタデバイスからスレーブデバイスへデータを転送する転送フォーマット)



注) S: スタートコンディション  
 R/W: 方向ビット  
 ACK: アクノリッジビット  
 P: ストップコンディション

図 10-2 I2C バスモード時のデータフォーマット

## 10.4 I2C バスモード時のコントロールレジスタ

シリアルバスインタフェースを I2C バスモードで使用するときの制御、および動作状態のモニタは以下のレジスタで行います。

### 10.4.1 SBICR0( 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。 許可の設定をして初めて、SBI に関連するレジスタのリード、ライトが可能になります。 禁止の場合、SBICR0 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。

注) 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。

10.4.2 SBICR1( コントロールレジスタ 1)

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

Bit	Bit Symbol	Type	機能																																																	
31-8	-	R	リードすると "0" が読めます。																																																	
7-5	BC[2:0]	R/W	転送ビット数の選択 (注 1) <table border="1" style="margin-left: 20px;"> <thead> <tr> <th rowspan="2">&lt;BC&gt;</th> <th colspan="2">&lt;ACK&gt; = 0 のとき</th> <th colspan="2">&lt;ACK&gt; = 1 のとき</th> </tr> <tr> <th>クロック数</th> <th>データ長</th> <th>クロック数</th> <th>データ長</th> </tr> </thead> <tbody> <tr><td>000</td><td>8</td><td>8</td><td>9</td><td>8</td></tr> <tr><td>001</td><td>1</td><td>1</td><td>2</td><td>1</td></tr> <tr><td>010</td><td>2</td><td>2</td><td>3</td><td>2</td></tr> <tr><td>011</td><td>3</td><td>3</td><td>4</td><td>3</td></tr> <tr><td>100</td><td>4</td><td>4</td><td>5</td><td>4</td></tr> <tr><td>101</td><td>5</td><td>5</td><td>6</td><td>5</td></tr> <tr><td>110</td><td>6</td><td>6</td><td>7</td><td>6</td></tr> <tr><td>111</td><td>7</td><td>7</td><td>8</td><td>7</td></tr> </tbody> </table>	<BC>	<ACK> = 0 のとき		<ACK> = 1 のとき		クロック数	データ長	クロック数	データ長	000	8	8	9	8	001	1	1	2	1	010	2	2	3	2	011	3	3	4	3	100	4	4	5	4	101	5	5	6	5	110	6	6	7	6	111	7	7	8	7
<BC>	<ACK> = 0 のとき		<ACK> = 1 のとき																																																	
	クロック数	データ長	クロック数	データ長																																																
000	8	8	9	8																																																
001	1	1	2	1																																																
010	2	2	3	2																																																
011	3	3	4	3																																																
100	4	4	5	4																																																
101	5	5	6	5																																																
110	6	6	7	6																																																
111	7	7	8	7																																																
4	ACK	R/W	マスターモード 0: アクノリッジのためのメントクロックを発生しない 1: アクノリッジのためのメントクロックを発生する スレーブモード 0: アクノリッジのためのメントクロックをカウントしない 1: アクノリッジのためのメントクロックをカウントする																																																	
3	-	R	リードすると "1" が読めます。																																																	
2-1	SCK[2:1]	R/W	内部 SCL 出力クロックの周波数選択 <SCK[2:0]>@ ライト (注 2)																																																	
0	SCK[0]	W	<table border="1" style="margin-left: 20px;"> <tbody> <tr><td>000</td><td>n = 5</td><td>769 kHz</td></tr> <tr><td>001</td><td>n = 6</td><td>588 kHz</td></tr> <tr><td>010</td><td>n = 7</td><td>400 kHz</td></tr> <tr><td>011</td><td>n = 8</td><td>244 kHz</td></tr> <tr><td>100</td><td>n = 9</td><td>137 kHz</td></tr> <tr><td>101</td><td>n = 10</td><td>73 kHz</td></tr> <tr><td>110</td><td>n = 11</td><td>38 kHz</td></tr> <tr><td>111</td><td></td><td>reserved</td></tr> </tbody> </table> <div style="margin-left: 40px;">                     システムクロック: fsys (= 80MHz)                      クロックギア: fc/1  <math display="block">\left. \begin{matrix} \text{周波数} = \frac{f_{\text{sys}}}{2^n + 72} \text{ [Hz]} \end{matrix} \right\}</math> </div>	000	n = 5	769 kHz	001	n = 6	588 kHz	010	n = 7	400 kHz	011	n = 8	244 kHz	100	n = 9	137 kHz	101	n = 10	73 kHz	110	n = 11	38 kHz	111		reserved																									
000	n = 5	769 kHz																																																		
001	n = 6	588 kHz																																																		
010	n = 7	400 kHz																																																		
011	n = 8	244 kHz																																																		
100	n = 9	137 kHz																																																		
101	n = 10	73 kHz																																																		
110	n = 11	38 kHz																																																		
111		reserved																																																		
	SWRMON	R	ソフトウェアリセット状態モニタ <SWRMON>@ リード 0: ソフトウェアリセット中 1: ソフトウェアリセット中ではない																																																	



- 注 1) SIO モードに切り替える前に <BC[2:0]> を "000" にクリアしてください。
- 注 2) SCL ラインクロックの周波数については、「10.5.1 シリアルクロック」を参照してください。
- 注 3) <SCK[0]/SWRMON> は、リセット後 "1" が読み出されますが、SBICR2 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) **マスターモードで <BC[2:0]>="001" で <ACK>="0" のとき、ストップコンディション発生後の SCL ラインの立下りによって、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。**

### 10.4.3 SBICR2( コントロールレジスタ 2)

このレジスタをリードすると、SBISR レジスタとして機能します。

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

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	MST	W	マスタモード / スレープモードの選択 0: スレープモード 1: マスタモード
6	TRX	W	送信 / 受信の選択 0: 受信 1: 送信
5	BB	W	スタート / ストップ状態の発生 0: ストップ状態発生 1: スタート状態発生
4	PIN	W	INTSBI 割り込み要求解除 0: - 1: 割り込み要求の解除
3-2	SBIM[1:0]	W	シリアルバスインタフェースの動作モード選択 (注) 00: ポートモード (シリアルバスインタフェースの出力禁止) 01: SIO モード 10: I2C バスモード 11: Reserved
1-0	SWRST[1:0]	W	ソフトウェアリセットの発生 最初に "10"、次に "01" をライトすると、ソフトウェアリセットが発生します。 ライトの際、<SBIM[1:0]> は "10" の I2C バスモードを指定してください。

注) 通信中はモードを切り替えないでください。  
ポートモードへの切り替えはバスフリーを確認してから行ってください。  
また、ポートモードから I2C バスモードへの切り替えは、ポートの状態が "High" になっていることを確認してから行ってください。

## 10.4.4 SBISR( ステータスレジスタ )

このレジスタをライトすると、SBICR2 として機能します。

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

Bit	Bit Symbol	Type	機能
31-8	-	R	リードすると "0" が読めます。
7	MST	R	マスタ / スレーブ選択状態モニタ 0: スレーブモード 1: マスタモード
6	TRX	R	トランスミッタ / レシーバ選択状態モニタ 0: レシーバ 1: トランスミッタ
5	BB	R	I2C バス状態モニタ 0: バスフリー 1: バスビジー
4	PIN	R	INTSBI 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 (ゼネラルコール検出時もセットされます。)
1	ADO	R	ゼネラルコール検出モニタ 0: - 1: 検出
0	LRB	R	最終受信ビットモニタ 0: 最終受信ビット "0" 1: 最終受信ビット "1"

10.4.5 SBIBR0( 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。

## 10.4.6 SBIDBR( 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) SBIDBR 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。

10.4.7 SBII2CAR(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)

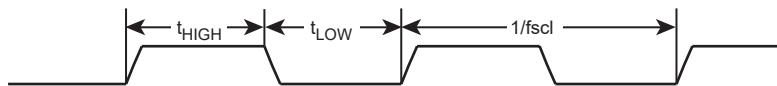
- 注 1) <ALS> はフリーデータフォーマット使用時以外は必ず "0" に設定してください。"1" に設定した場合にはフリーデータフォーマットとして動作し、マスタ時は送信に、スレーブ時は受信に転送方向が固定されません。
- 注 2) スレーブモード時 SBII2CAR を "0x00" に設定しないでください。("0x00" に設定した場合、スレーブモードで I2C バス規格の START バイト ("0x01") を受信した時にスレーブアドレスが一致したと判断します。)

## 10.5 I2C バスモード時の制御

### 10.5.1 シリアルクロック

#### 10.5.1.1 クロックソース

SBICR1 <SCK[2:0]> で、マスタモード時に SCL 端子から出力されるシリアルクロックの最大転送周波数を選択します。



$$t_{LOW} = 2^{n-1}/f_{sys} + 58/f_{sys}$$

$$t_{HIGH} = 2^{n-1}/f_{sys} + 14/f_{sys}$$

$$f_{SCL} = 1/(t_{LOW} + t_{HIGH})$$

$$= \frac{f_{sys}}{2^n + 72}$$

SBICR1<SCK[2:0]>	n
000	5
001	6
010	7
011	8
100	9
101	10
110	11

図 10-3 クロックソース

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

#### 10.5.1.2 クロック同期化

I2C バスでは、端子の構造上バスをワイヤードアンドで駆動させるため、クロックラインを最初に "Low" レベルに引いたマスタが、"High" レベルを出力しているマスタのクロックを無効にします。このため、"High" レベルを出力しているマスタは、これを検出し対応する必要があります。

シリアルバスインタフェース回路はクロック同期化機能をもっており、バス上に複数のマスタが存在する場合でも、正常に転送が行われます。

クロック同期の手順を、バス上に 2 つのマスタが同時に存在した場合を例に挙げて以下に示します。

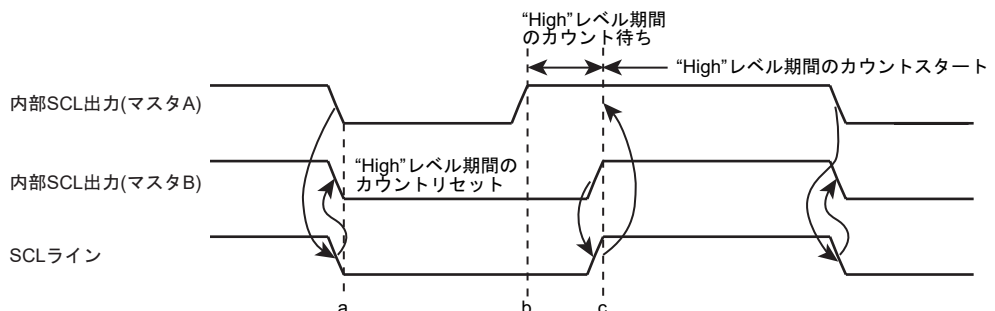


図 10-4 クロック同期化の例

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

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

以上のようにバス上のクロックは、バスに接続されているマスタの中で最も短い "High" レベル期間をもつマスタと最も長い "Low" レベル期間をもつマスタによって決定されます。

### 10.5.2 アクノリッジメントモードの指定

SBICR1<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 SDA 端子を開放し、レシーバからのアクノリッジ信号を受信できる状態にします。レシーバモードのときはクロック期間中 SDA 端子を "Low" レベルに引き、更に、スレーブモードのときにゼネラルコールを受信した場合にもアクノリッジのためのクロック期間中、SDA 端子を "Low" レベルに引き、アクノリッジ信号を発生します。

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

### 10.5.3 転送ビット数の選択

SBICR1<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。

### 10.5.4 スレーブアドレスとアドレス認識モードの設定

スレーブアドレスを認識するアドレッシングフォーマットで動作させるときは、SBII2CAR<ALS> に "0" を設定し、SBII2CAR <SA[6:0]> にスレーブアドレスを設定します。

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



### 10.5.5 動作モード

SBICR2<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、シリアルバスインタフェース端子の状態が "High" になっていることを確認後、<SBIM[1:0]> を "10" に設定してください。また、ポートモードへの切り替えは、バスフリーであることを確認してから行ってください。

### 10.5.6 トランスミッタ/レシーバの選択

SBICR2<TRX> を "1" に設定すると、トランスミッタとして動作し、<TRX> を "0" に設定すると、レシーバとして動作します。

スレーブモード時は、

- アドレッシングフォーマットのデータ転送を行うとき
- 受信したスレーブアドレスが SBII2CAR にセットした値と同じとき
- ゼネラルコール (スタートコンディション後の 8 ビットのデータがすべて "0") を受信したとき

ハードウェアによりマスタデバイスから送られてくる方向ビット (R/W) が "1" の場合、<TRX> は "1" にセットされ、"0" の場合、<TRX> は "0" にクリアされます。

マスタモード時は、スレーブデバイスからアクノリッジが返ってくると、ハードウェアにより、送信した方向ビットが "1" の場合、<TRX> は "0" に、方向ビットが "0" の場合、<TRX> は "1" に変化します。アクノリッジが返ってこないときは、以前の状態を保ちます。

<TRX> はバス上のストップコンディションの検出、またはアービトレーションロストの検出で、ハードウェアにより "0" にクリアされます。

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

### 10.5.7 マスタ/スレーブの選択

SBICR2<MST> を "1" に設定すると、マスタデバイスとして動作します。

<MST> を "0" に設定すると、スレーブデバイスとして動作します。<MST> はバス上のストップコンディションの検出、またはアービトレーションロストの検出で、ハードウェアにより "0" にクリアされます。

### 10.5.8 スタート / ストップコンディションの発生

SBISR<BB> が "0" のときに、SBICR2<MST, TRX, BB, PIN> に "1" を書き込むと、バス上にスタートコンディションと、あらかじめデータバッファレジスタに書き込んだスレーブアドレスと方向ビットが出力されます。あらかじめ、<ACK> に "1" を設定しておいてください。

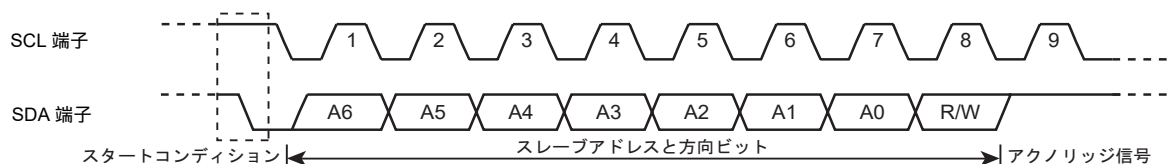


図 10-5 スタートコンディションの発生とスレーブアドレスの発生

<BB> = "1" のときに、<MST, TRX, PIN> に "1"、<BB> に "0" を書き込むと、バス上にストップコンディション出力のシーケンスが開始されます。バス上にストップコンディションが発生するまで、<MST, TRX, BB, PIN> の内容を書き替えないでください。

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

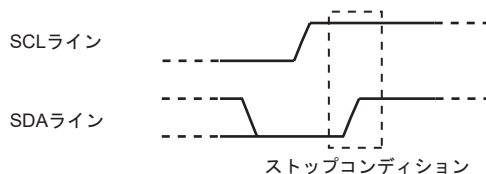


図 10-6 ストップコンディションの発生

また、SBISR<BB> を読み出すことで、バスの状態を知ることができます。<BB> は、バス上のスタートコンディションを検出すると "1" にセットされ (バスビジー状態)、ストップコンディションを検出すると "0" にクリアされます (バスフリー状態)。

### 10.5.9 割り込みサービス要求と解除

マスタモードの時、<BC> と <ACK> によって設定されたクロック数の転送が終了すると、シリアルバスインタフェース割り込み要求 (INTSBI) が発生します。

スレーブモードの場合は、以下のときに INTSBI が発生します。

- 受信したスレーブアドレスが SBII2CAR<SA[6:0]> に設定されたスレーブアドレスとアクノリッジ信号出力後
- ゼネラルコールを受信した時のアクノリッジ信号出力後
- スレーブアドレス一致、またはゼネラルコール受信後におけるデータ転送終了時

アドレス認識モード (<ALS> = "0") では、受信したスレーブアドレスが SBII2CAR にセットした値と同じとき、またはゼネラルコール (スタートコンディション後の 8 ビットのデータがすべて "0") を受信したときに INTSBI が発生します。

割り込み要求 (INTSBI) が発生すると、SBICR2<PIN> が "0" にクリアされます。<PIN> が "0" の間、SCL ラインを "Low" レベルに引きます。

<PIN> は SBIDBR にデータを書き込むか、SBIDBR からデータを読み出すと "1" にセットされます。<PIN> が "1" にセットされてから、SCL ラインが開放されるまで、 $t_{LOW}$  の時間がかかります。プログラムで <PIN> に "1" を書き込むと "1" にセットされますが、"0" を書き込んでも "0" にクリアされません。

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

### 10.5.10 アービトレーションロスト検出モニタ

I2C バスではマルチマスタ (1 つのバス上で同時に 2 つ以上のマスタが存在する) が可能なため、転送されるデータの内容を保証するためにバスのアービトレーション手順が必要となります。

バスビジーの状態のときにスタートコンディションを出力しようとした場合、SCL, SDA ラインには出力されずにアービトレーションロストが発生します。I2C バスではバスのアービトレーションに SDA ラインのデータを使用します。

アービトレーションの手順を、バス上に 2 つのマスタが同時に存在した場合を例に挙げて以下に示します。

a 点のビットまでマスタ A, マスタ B と同じデータを出力し、a 点でマスタ A が "Low" レベルを出力、マスタ B が "High" レベルを出力すると、バスの SDA ラインはワイヤードアンドで駆動されるためにマスタ A によって "Low" レベルに引かれます。b 点でバスの SCL ラインが立ち上がると、スレーブデバイスは SDA ラインデータ、すなわち、マスタ A のデータを取り込みます。このとき、マスタ B の出力したデータは無効になります。マスタ B のこの状態を "アービトレーションロスト" と呼びます。マスタ B は SDA 端子を開放し、他のマスタの出力するデータに影響を及ぼさないようにします。また、複数のマスタが 1 ワード目でまったく等しいデータを送信した場合、アービトレーションの手順は 2 ワード目以降も継続されます。

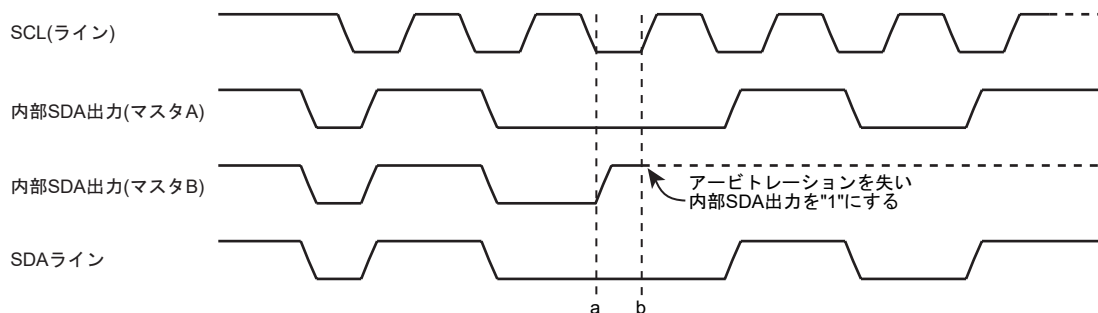


図 10-7 アービトレーションロスト

バスの SDA ラインのレベルと内部 SDA 出力のレベルの比較は、SCL ラインの立ち上がりで行います。この比較結果が不一致の場合アービトレーションロストになり、SBISR<AL> が "1" にセットされます。

<AL> が "1" にセットされると SBISR<MST, TRX> は "0" にクリアされ、スレーブレシーバモードになります。そのため、<AL> が "1" にセットされた後のデータ転送ではシリアルバスインタフェース回路はクロック出力を停止します。

<AL> は、SBIDBR にデータを書き込むか、SBIDBR からデータを読み込む、または SBICR2 にデータを書き込むと "0" にクリアされます。

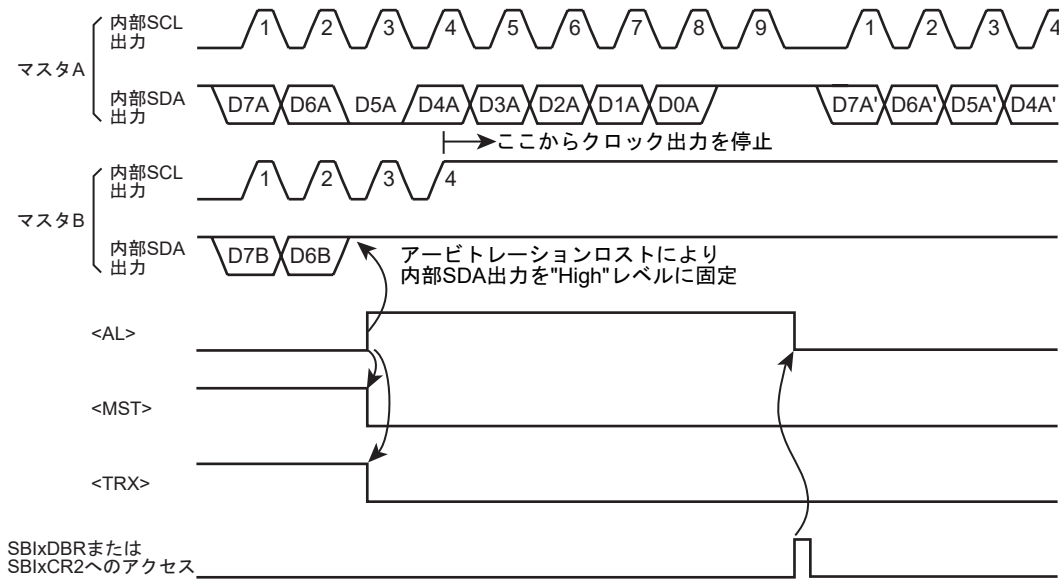


図 10-8 マスタ B の場合の例 (D7A=D7B, D6A=D6B)

### 10.5.11 スレーブアドレス一致検出モニタ

SBISR<AAS> は、スレーブモード時、アドレス認識モード (SBII2CAR<ALS> = "0") のとき、ゼネラルコールまたは SBII2CAR 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 SBIDBR にデータを書き込むか、SBIDBR からデータを読み出すと "0" にクリアされます。

### 10.5.12 ゼネラルコール検出モニタ

SBISR<ADO> は、スレーブモード時、ゼネラルコール (スタートコンディション後の 8 ビットのデータがすべて "0") を受信したとき "1" にセットされます。また、バス上のスタートコンディション、またはストップコンディションが検出されると "0" にクリアされます。

### 10.5.13 最終受信ビットモニタ

SBISR<LRB>には、SCL ラインの立ち上がりで取り込まれた SDA ラインの値がセットされます。アクノリジメントモードのとき、INTSBI 割り込み要求発生直後に SBISR<LRB>を読み出すと、ACK 信号が読み出されます。

### 10.5.14 データバッファレジスタ (SBIDBR)

SBIDBR をリード/ライトすることで、受信データの読み出し/送信データの書き込みを行います。また、マスタモード時において、このレジスタにスレーブアドレスと方向ビットを設定後、スタートコンディションが発生します。

### 10.5.15 ボーレートレジスタ (SBIBR0)

SBIBR0<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。

### 10.5.16 ソフトウェアリセット

シリアルバスインタフェース回路が、外部からのノイズによりロックした場合、ソフトウェアリセット機能を使い、シリアルバスインタフェース回路を初期化することができます。

SBICR2<SWRST[1:0]>へ、最初に"10"、次に"01"をライトすると、ソフトウェアリセットが発生します。ライトの際、SBICR2<SBIM[1:0]>は"10"のI2Cバスモードを指定してください。ソフトウェ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"0"にクリアされます。

注) ソフトウェアリセットをかけると動作モード選択もリセットされ、I2CモードからPORTモードになります。

## 10.6 I2C バスモード時のデータ転送手順

### 10.6.1 デバイスの初期化

最初に SBICR1<ACK, SCK[2:0]> を設定します。SBICR1[7:5] には、"0" を書き込んでください。

次に SBII2CAR にスレーブアドレス <SA[6:0]> と <ALS> ( アドレッシングフォーマット時、<ALS> = "0") を設定します。

それから、シリアルバスインタフェース端子の状態が "High" になっていることを確認し、SBICR2<MST, TRX, BB> に "0", <PIN> に "1", <SBIM[1:0]> に "10", ビット 1, 0 に "0" を書き込み、初期状態をスレーブレシーバモードにします。

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

	7	6	5	4	3	2	1	0	
SBICR1	← 0	0	0	X	0	X	X	X	ACK および SCL クロックの設定をします。
SBII2CAR	← 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。
SBICR2	← 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

### 10.6.2 スタートコンディション, スレーブアドレスの発生

#### 10.6.2.1 マスタモードの場合

マスタモード時は、スタートコンディションとスレーブアドレスを、次の手順で発生します。

はじめに、バスフリー状態 (<BB> = "0") を確認します。次に、SBICR1<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、SBIDBR 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、SBICR2<MST, TRX, BB, PIN> に "1111" を書き込むと、バス上にスタートコンディションが発生します。スタートコンディションの発生に次いで、SCL 端子から 9 クロックを出力します。最初の 8 クロックで、SBIDBR に設定したスレーブアドレスと方向ビットを出力します。9 クロック目で、SDA ラインを解放し、スレーブデバイスからのアクノリッジ信号を受信します。

9 クロック目の立ち下がり、INTSBI 割り込み要求が発生し、<PIN> = "0" にされます。マスタモード時は、<PIN> = "0" の間 SCL ラインを "Low" レベルにひきます。また、スレーブデバイスからのアクノリッジ信号が返ってきたときのみ、INTSBI 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 SBIDBR に書き込む時は事前にソフトウェアによってバスフリーを検出してから行ってください。この制約が守られない場合、現在出力中のバス上のデータが破壊されることがあります。

メインルーチンでの設定

		7	6	5	4	3	2	1	0	
Reg.	←	SBISR								
Reg.	←	Reg. e 0x20								
if Reg.	≠	0x00								バスがフリー状態になるまで確認します。
Then										
SBICR1	←	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。
SBIDBR	←	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。
SBICR2	←	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。

INTSBI 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

10.6.2.2 スレーブモードの場合

スレーブモードの場合は、スタートコンディションとスレーブアドレスを受信します。

マスタデバイスからのスタートコンディションを受信した後、SCL ラインの最初の 8 クロックで、マスタデバイスからのスレーブアドレスと方向ビットを受信します。ゼネラルコール、または SBII2CAR に設定されたスレーブアドレスと同一のアドレスを受信したとき、9 クロック目で SDA ラインを "Low" レベルに引き、アクノリッジ信号を出力します。

9 クロック目の立ち下がり、INTSBI 割り込み要求が発生し、<PIN> = "0" にされます。スレーブモード時は、<PIN> = "0" の間 SCL ラインを "Low" レベルに引きます。

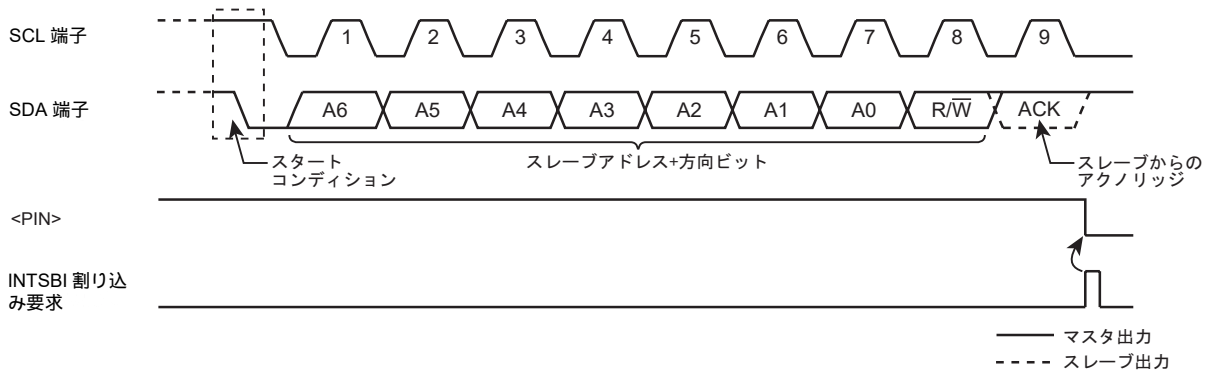


図 10-9 スタートコンディションとスレーブアドレスの発生

### 10.6.3 1ワードのデータ転送

1ワード転送終了の INTSBI 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。

#### 10.6.3.1 マスタモードの場合 (<MST> = "1")

<TRX> をテストし、トランスミッタ / レシーバの判断をします。

##### (1) トランスミッタモードの場合 (<TRX> = "1")

<LRB> をテストします。<LRB> = "1" のとき、レシーバはデータを要求していないのでストップコンディションを発生する処理 (後記参照) を行ってデータ転送を終了します。

<LRB> = "0" のときレシーバが次のデータを要求しています。次に転送するデータのビット数が 8 ビットするとき SBIDBR 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 SBIDBR に書き込みます。データを書き込むと <PIN> が "1" になり SCL 端子から次の 1ワードデータ転送用のシリアルクロックが発生され、SDA 端子から 1ワードのデータが転送されます。転送終了後 INTSBI 割り込み要求が発生し、<PIN> が "0" になり SCL 端子を "Low" レベルに引きます。複数ワードの転送が必要な場合は上記 <LRB> のテストから繰り返します。

#### INTSBI 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
SBICR1    ←  X  X  X  X  0  X  X  X      転送ビット数および ACK を設定します。
SBIDBR    ←  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

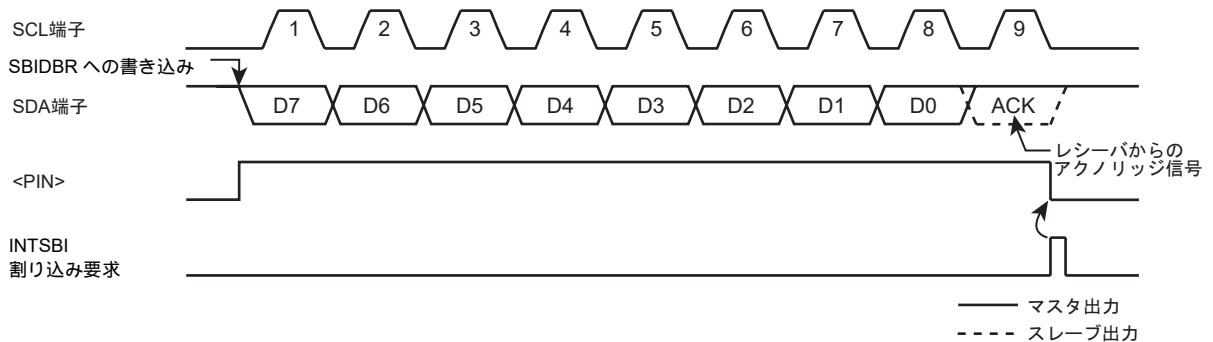


図 10-10 <BC[2:0]>="000", <ACK>="1" の場合 (トランスミッタモード)



(2) レシーバモードの場合 (<TRX> = "0")

次に転送するデータのビット数が 8 ビットの場合は SBIDBR に転送データを書き込みます。8ビット以外の場合は<BC[2:0]>, <ACK>を設定し、SCLラインを解放するためにSBIDBRから受信データを読み出します（スレーブアドレス送信直後のリードデータは不定です）。データを読み出すと<PIN>は"1"になり、次の1ワードデータ転送用のシリアルクロックをSCL端子に出力します。最後のビットでアクノリッジ信号の"Low"レベルのタイミングで"0"をSDA端子に出力します。

その後、INTSBI 割り込み要求が発生し、<PIN>が"0"になりSCL端子を"Low"レベルに引きます。SBIDBR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。

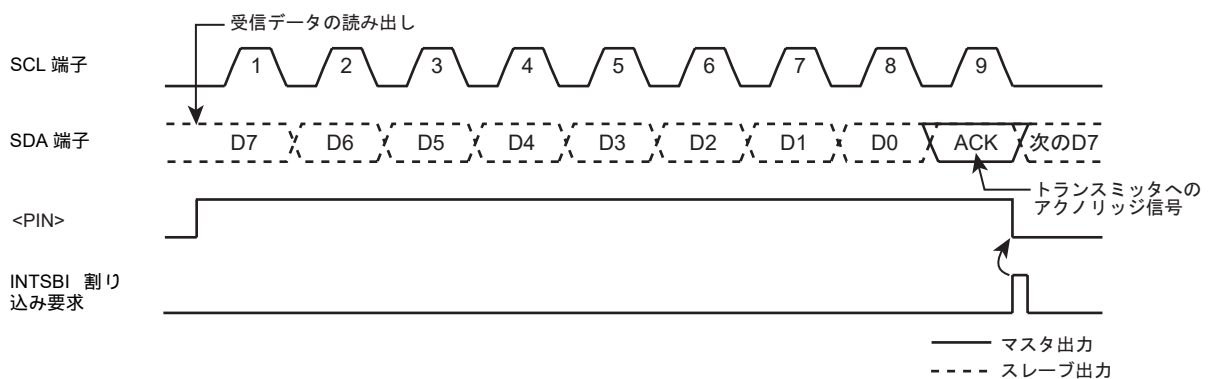


図 10-11 <BC[2:0]>="000", <ACK>="1" のときの例 (レシーバモード)

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

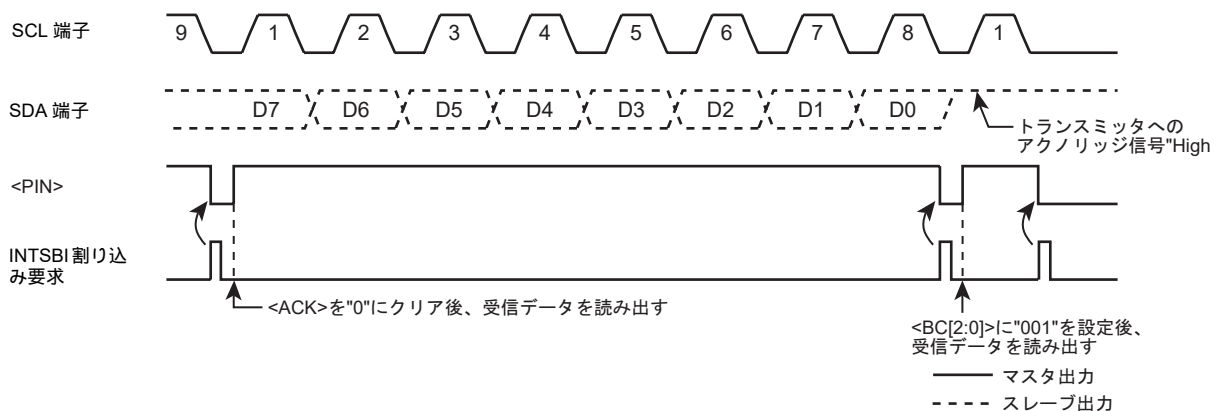


図 10-12 マスタレシーバモード時、データの送信を終了させるときの処理

例：データを N 回受信する場合

INTSBI 割り込み (データ送信後)

```

          7   6   5   4   3   2   1   0
SBICR1  ←  X   X   X   X   0   X   X   X
Reg.     ←  SBIDBR
割り込み終了
    
```

受信データのビット数および ACK を設定します。  
ダミーデータを取り込みます。

INTSBI 割り込み (データ受信 1~(N-2) 回目)

```

          7   6   5   4   3   2   1   0
Reg.     ←  SBIDBR
割り込み終了
    
```

1~(N-2) 回目のデータを取り込みます。

INTSBI 割り込み (データ受信 (N-1) 回目)

```

          7   6   5   4   3   2   1   0
SBICR1  ←  X   X   X   0   0   X   X   X
Reg.     ←  SBIDBR
割り込み終了
    
```

アクノリッジ信号のクロックを発生しないようにし  
ます。  
(N-1) 回目のデータを取り込みます。

INTSBI 割り込み (データ受信 N 回目)

```

          7   6   5   4   3   2   1   0
SBICR1  ←  0   0   1   0   0   X   X   X
Reg.     ←  SBIDBR
割り込み終了
    
```

1 ビット転送のためのクロックを発生します。  
N 回目のデータを取り込みます。

INTSBI 割り込み (データ受信後)

ストップコンディションを発生する処理  
割り込み終了

データ転送を終了させます。

注) X; Don't care

### 10.6.3.2 スレーブモードの場合 (<MST> = "0")

スレーブモードのとき、マスタが送ったスレーブアドレス、またはゼネラルコールを受信したとき、もしくは、受信したスレーブアドレスが一致した後、またはゼネラルコールを受信した後のデータ転送終了時に INTSBI 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 INTSBI 割り込み要求が発生します。INTSBI 割り込み要求が発生すると <PIN> が "0" にされ、SCL 端子を "Low" レベルに引きます。SBIDBR にデータを書き込む、SBIDBR からデータを読み出す、または <PIN> に "1" を設定すると SCL 端子が  $t_{LOW}$  後に開放されます。

スレーブモード時は、通常のスレーブモードとしての処理、またはアービトレーションロストを検出し、スレーブモードになったときの処理を行います。

SBISR<AL>, <TRX>, <AAS>, <ADO> をテストし、場合分けを行います。「表 10-2 スレーブモード時の処理」にスレーブモード時の状態と必要な処理を示します。

例：スレーブレシーバモード時スレーブアドレスが一致し、方向ビットが "1" の場合

#### INTSBI 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
SBICR1    ←  X  X  X  1  0  X  X  X      送信ビット数を設定します。
SBIDBR    ←  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 10-2 スレーブモード時の処理

<TRX>	<AL>	<AAS>	<ADO>	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 SBIDBR に書き込みます。
	0	1	0	スレーブレシーバモード時、マスタが送った方向ビットが "1" のスレーブアドレスを受信	
		0	0	0	スレーブトランスマッタモード時、1 ワードのデータの送信が終了
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 SBIDBR 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	

### 10.6.4 ストップコンディションの発生

SBISR<BB> = "1" のときに、SBICR2<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、SDA 端子が立ち上がり、ストップコンディションが発生します。

```

          7   6   5   4   3   2   1   0
SBICR2 ← 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。

```

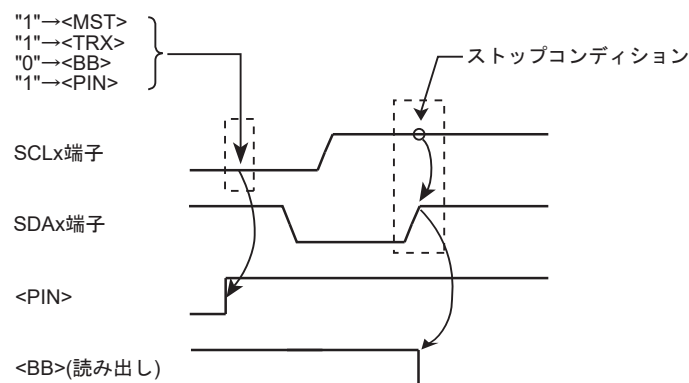


図 10-13 ストップコンディションの発生

### 10.6.5 再スタートの手順

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

まず、SBICR2<MST, TRX, BB> に "0"、<PIN> に "1" を書き込み、バスを開放します。このとき SDA 端子は "High" レベルを保ち、SCL 端子が開放され、バス上にストップコンディションが発生されないため、他のデバイスからみるとバスはビジー状態のままです。このあと、SBISR<BB> をテストして "0" になるまで待ち、SCL 端子が開放されたことを確認します。次に <LRB> をテストして "1" になるまで待ち、他のデバイスがバスの SCL ラインを "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「10.6.2 スタートコンディション、スレーブアドレスの発生」の手順で、スタートコンディションの発生を行います。

なお、再スタート時のセットアップタイムを満たすために、バスフリーの確認からスタートの発生まで最低 4.7  $\mu$ s (標準モード時) のソフトウェアによる待ち時間が必要です。

注 1) <MST> = "0" の状態の時に <MST> = "0" をライトしないでください (再スタートできません)。

注 2) マスタデバイスがレシーバのとき、再スタートを発生させる前にトランスミッタとなっているスレーブデバイスからのデータ送信を終了させる必要があります。データ送信を終了させるために、"High" レベルのアクノリッジ信号をスレーブデバイスに受信させます。このため、再スタート発生前の <LRB> は "1" となり、再スタートの手順で <LRB> = "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		
→	SBICR2	←	0	0	0	1	1	0	0	0	バスを開放します。
→	if SBISR<BB> ≠ 0										SCL 端子の開放を確認します。
→	Then										
	if SBISR<LRB> ≠ 1										他のデバイスの SCL 端子 "Low" レベルの確認を行います。
	Then										
	4.7 μs Wait										
	SBICR1	←	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。
	SBIDBR	←	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。
	SBICR2	←	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

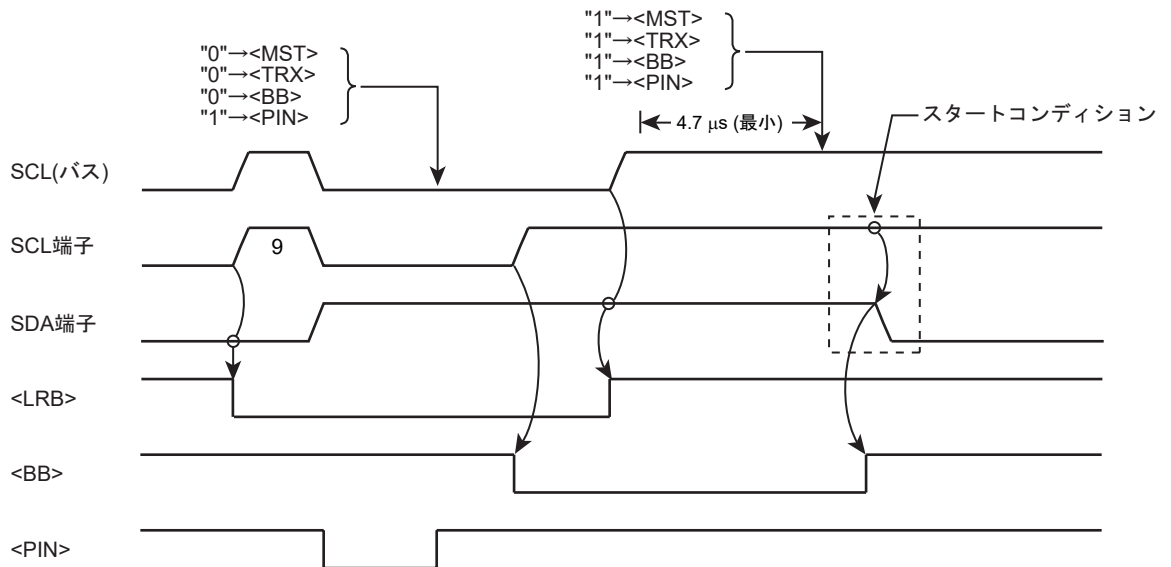


図 10-14 再スタートを発生する場合のタイミングチャート

## 10.7 SIO モード時のコントロールレジスタ

シリアルバスインタフェースをクロック同期式 8 ビット SIO モードで使用するときの制御、および動作状態のモニタは、以下のレジスタで行います。

### 10.7.1 SBICR0( 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、SBICR0 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。

10.7.2 SBICR1( コントロールレジスタ 1)

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

Bit	Bit Symbol	Type	機能																								
31-8	-	R	リードすると "0" が読めます。																								
7	SIOS	R/W	転送の開始 / 終了 0: 終了 1: 開始																								
6	SIOINH	R/W	転送の強制停止 0: 転送継続 1: 強制停止																								
5-4	SIOM[1:0]	R/W	転送モードの設定 00: 送信モード 01: Reserved 10: 送受信モード 11: 受信モード																								
3	-	R	リードすると "1" が読めます。																								
2-0	SCK[2:0]	R/W	シリアルクロックの周波数選択 <SCK[2:0]>@ ライト (注 1) <table border="1" style="margin-left: 20px;"> <tr> <td>000</td> <td>n = 3</td> <td>5 MHz</td> </tr> <tr> <td>001</td> <td>n = 4</td> <td>2.5 MHz</td> </tr> <tr> <td>010</td> <td>n = 5</td> <td>1.25 MHz</td> </tr> <tr> <td>011</td> <td>n = 6</td> <td>625 kHz</td> </tr> <tr> <td>100</td> <td>n = 7</td> <td>313 kHz</td> </tr> <tr> <td>101</td> <td>n = 8</td> <td>156 kHz</td> </tr> <tr> <td>110</td> <td>n = 9</td> <td>78 kHz</td> </tr> <tr> <td>111</td> <td>-</td> <td>外部クロック</td> </tr> </table> <div style="margin-left: 40px;"> <math display="block">\left. \begin{array}{l} \text{システムクロック: } f_{\text{sys}} \\ \text{クロックギア: } fc/1 \\ \text{周波数} = \frac{f_{\text{sys}}/2^n}{2^n} \text{ [Hz]} \end{array} \right\} \begin{array}{l} (= 80\text{MHz}) \end{array}</math> </div>	000	n = 3	5 MHz	001	n = 4	2.5 MHz	010	n = 5	1.25 MHz	011	n = 6	625 kHz	100	n = 7	313 kHz	101	n = 8	156 kHz	110	n = 9	78 kHz	111	-	外部クロック
000	n = 3	5 MHz																									
001	n = 4	2.5 MHz																									
010	n = 5	1.25 MHz																									
011	n = 6	625 kHz																									
100	n = 7	313 kHz																									
101	n = 8	156 kHz																									
110	n = 9	78 kHz																									
111	-	外部クロック																									

注 1) <SCK[0]> ビットは、リセット後 "1" が読み出されますが、SBICR2 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、SBICR2 レジスタ、SBISR 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) マスターモードで <BC[2:0]>="001" で <ACK>="1" 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。



## 10.7.3 SBIDBR( 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) SBIDBR 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。

### 10.7.4 SBICR2( コントロールレジスタ 2)

このレジスタをリードすると、SBISR 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。

## 10.7.5 SBISR( ステータスレジスタ )

このレジスタをライトすると、SBICR2 として機能します。

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

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

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

10.7.6 SBIBR0( 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。

## 10.8 SIO モード時の制御

### 10.8.1 シリアルクロック

#### 10.8.1.1 クロックソース

SBICR1 <SCK[2:0]> により、次の選択ができます。

##### (1) 内部クロック

内部クロックモードでは7種類の周波数が選択できます。シリアルクロックはSCK端子より外部に出力されます。なお、転送開始時SCK端子出力は"High"レベルになります。

プログラムでデータの書き込み(送信時)またはデータの読み出し(受信時)がこのシリアルクロックレートに追従できないときには、自動的にシリアルクロックを停止し、それらの処理が終了するまで次のシフト動作を待機させる自動ウェイト機能を有しています。

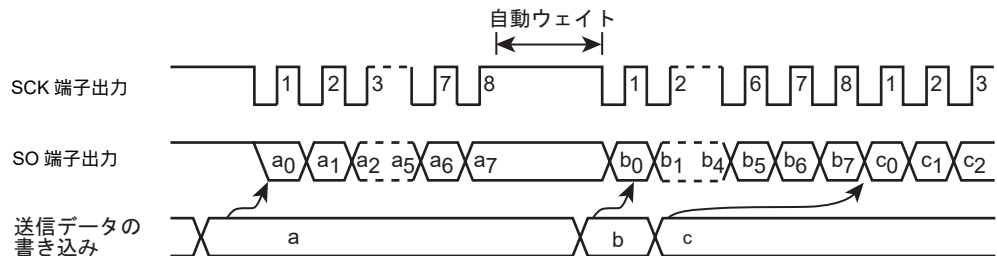


図 10-15 自動ウェイト機能

##### (2) 外部クロック (<SCK[2:0]> = "111")

外部からSCK端子に供給されるクロックをシリアルクロックとして用います。なお、シフト動作を確実に行うためには、シリアルクロックの"High"レベル、"Low"レベル幅は下記に示すパルス幅が必要です。

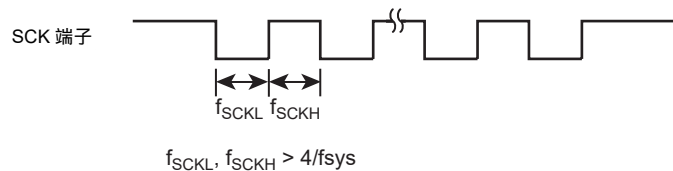


図 10-16 外部クロック入力時の最大転送周波数

### 10.8.1.2 シフトエッジ

送信は前縁シフト、受信は後縁シフトになります。

- 前縁シフト

シリアルクロックの前縁 (SCK 端子入出力の立ち下がりエッジ) でデータをシフトします。

- 後縁シフト

シリアルクロックの後縁 (SCK 端子入出力の立ち上がりエッジ) でデータをシフトします。

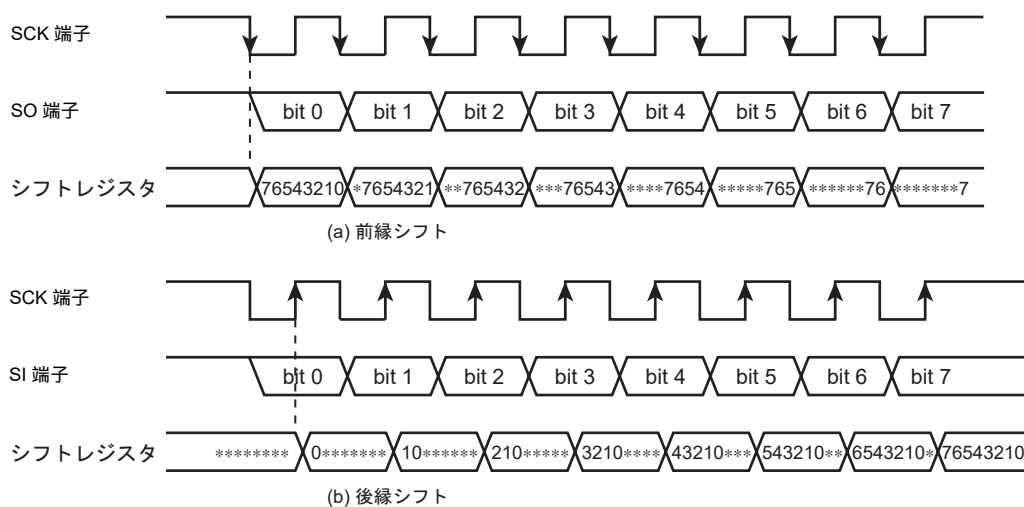


図 10-17 シフトエッジ

## 10.8.2 転送モード

SBICR1<SIOM[1:0]> で、送信 / 受信 / 送受信モードを選択します。

### 10.8.2.1 8ビット送信モード

コントロールレジスタに送信モードをセットした後、送信データを SBIDBR に書き込みます。

送信データの書き込み後、SBICR1<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、SBIDBR からシフトレジスタに移され、シリアルクロックに同期して最下位ビット (LSB) 側から SO 端子に出力されます。送信データがシフトレジスタに移されると、SBIDBR が空になりますので、次の送信データを要求する INTSBI(バッファEMPTY) 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、SBIDBR 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、SBIDBR 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、SBISR<SIOF> が "1" となってから SCK の立ち上がりエッジまでの間、前回送信したデータの最後のビットと同じ値が出力されます。

送信を終了させるには、INTSBI 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 SBISR<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、データのデータの送信後、停止します。

		7	6	5	4	3	2	1	0	
SBICR1	←	0	1	0	0	0	X	X	X	送信モードをセットします。
SBIDBR	←	X	X	X	X	X	X	X	X	送信データを書き込みます。
SBICR1	←	1	0	0	0	0	X	X	X	送信を開始します。

#### INTSBI 割り込み

SBIDBR	←	X	X	X	X	X	X	X	X	送信データを書き込みます。
--------	---	---	---	---	---	---	---	---	---	---------------

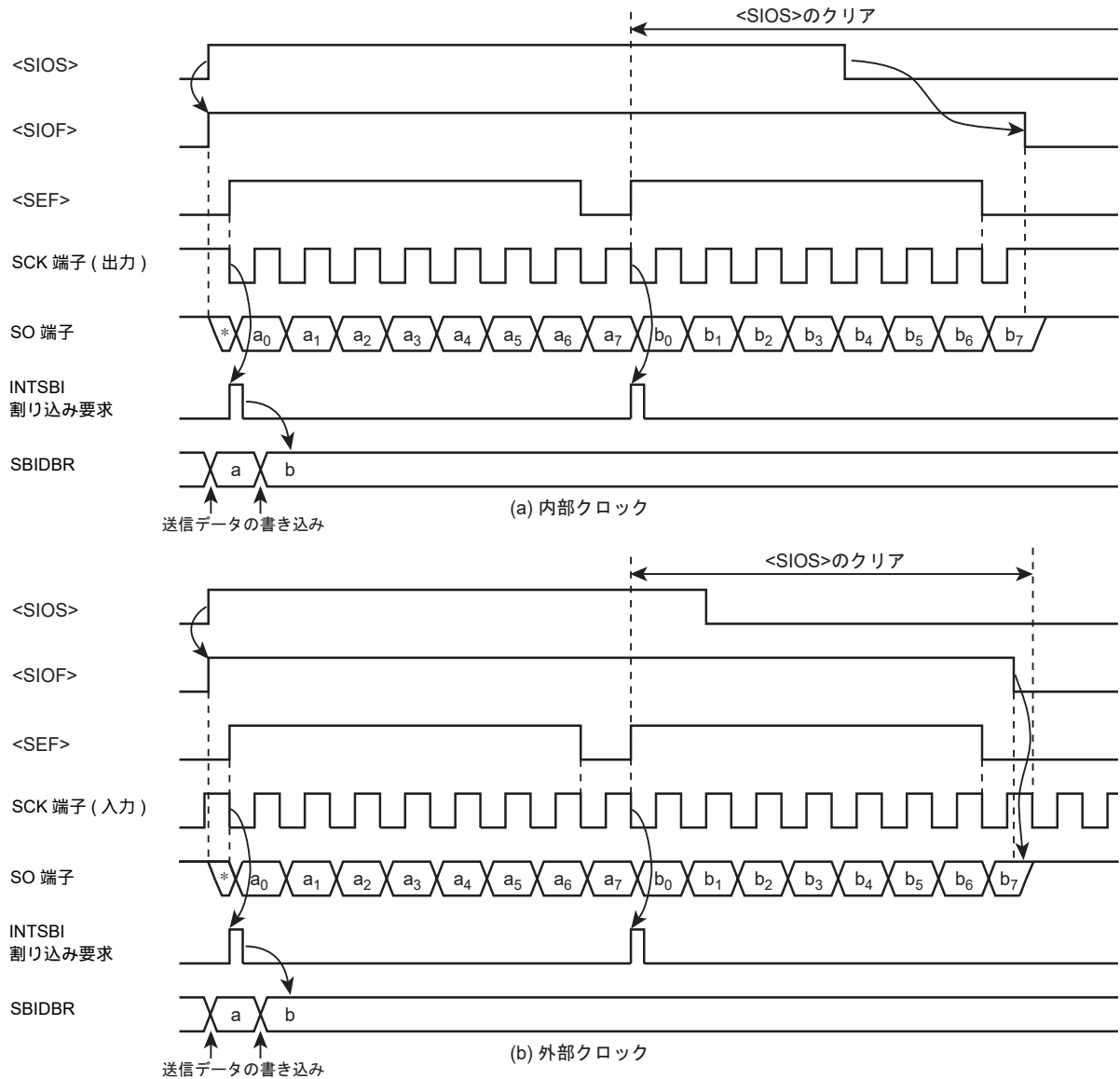


図 10-18 送信モード

例 : <SIO> の送信終了指示 (外部クロックの場合) のプログラム例

```

    7 6 5 4 3 2 1 0
    if SBISR<SIOF> ≠ 0
    Then
    if SCK ≠ 1
    Then
    SBICR1 ← 0 0 0 0 0 1 1 1

```

転送の終了を確認します。

ポートをモニタし、SCK 端子が "1" になったことを確認します。

<SIO> = 0 を設定し送信を終了します。



10.8.2.2 8ビット受信モード

コントロールレジスタに受信モードをセットした後、SBICR1<SIOS> = "1" を書き込むことにより受信可能となります。シリアルクロックに同期して、SI 端子より最下位ビット側からシフトレジスタへデータを取り込みます。8ビットのデータが取り込まれるとシフトレジスタから SBIDBR 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 INTSBI (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 SBIDBR 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 SBIDBR 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 INTSBI 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、SBIDBR 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、SBISR<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 SBIDBR 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	
SBICR1	←	0	1	1	1	0	X	X	X	受信モードをセットします。
SBICR1	←	1	0	1	1	0	X	X	X	受信を開始します。

INTSBI 割り込み

Reg.	←	SBIDBR	受信データを取り込みます。
------	---	--------	---------------

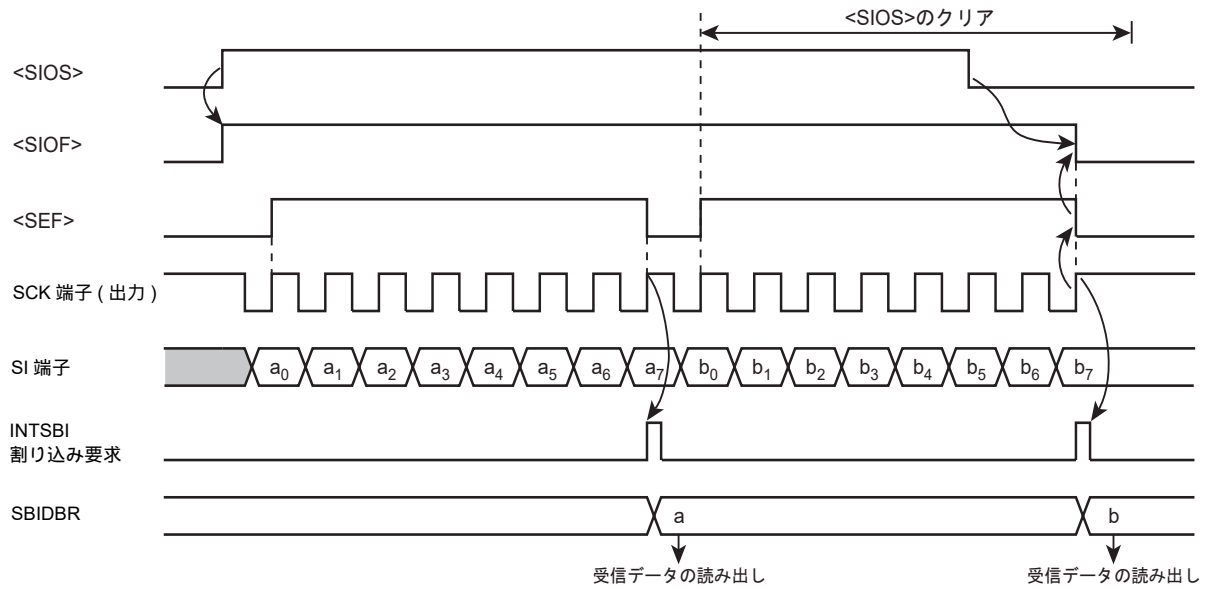


図 10-19 受信モード (例：内部クロック)

## 10.8.2.3 8ビット送受信モード

コントロールレジスタに送受信モードをセットした後、送信データを SBIDBR に書き込みます。その後、SBICR1<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上がりで送信データが SO<sub>x</sub> 端子から出力され、立ち上がりで受信データが SI 端子から取り込まれます。8ビットのデータが取り込まれると、シフトレジスタから SBIDBR へ受信データが転送され、INTSBI 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。SBIDBR 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 SCK の立ち上がりエッジまでの間、前回送信したデータの最後のビットと同じ値が出力されます。

送受信を終了させるには、INTSBI 割り込みサービスプログラムで <SIOS> = "0" を書き込むか SBICR1 <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、SBIDBR 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、SBISR<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 SBIDBR 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。

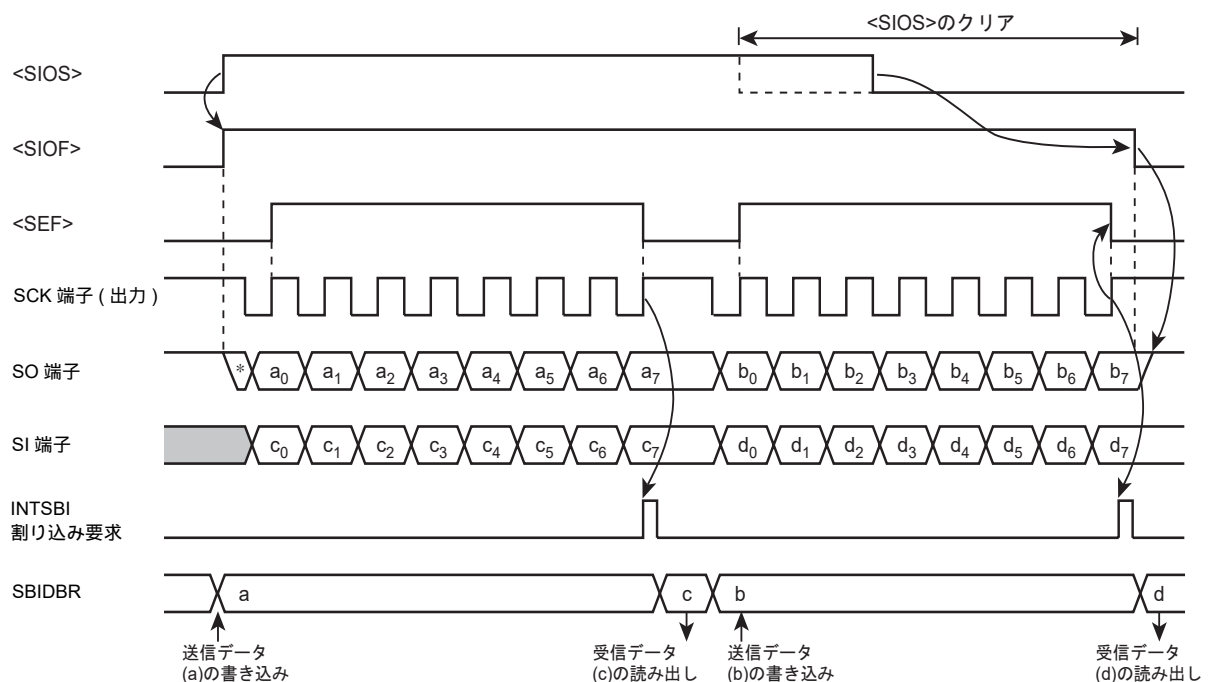


図 10-20 送受信モード (例: 内部クロック)

		7	6	5	4	3	2	1	0	
SBICR1	←	0	1	1	0	0	X	X	X	送受信モードをセットします。
SBIDBR	←	X	X	X	X	X	X	X	X	送信データを書き込みます。
SBICR1	←	1	0	1	0	0	X	X	X	送受信を開始します。

INTSBI 割り込み

Reg.	←	SBIDBR								受信データを取り込みます。
SBIDBR	←	X	X	X	X	X	X	X	X	送信データを書き込みます。

10.8.2.4 送信終了時の最終ビット保持時間

SBICR1<SIOS>="0" の状態では、送信データの最終ビットの SCK 立ち上がりに対するデータ保持時間は以下ようになります。送信モード、送受信モードとも同様です。

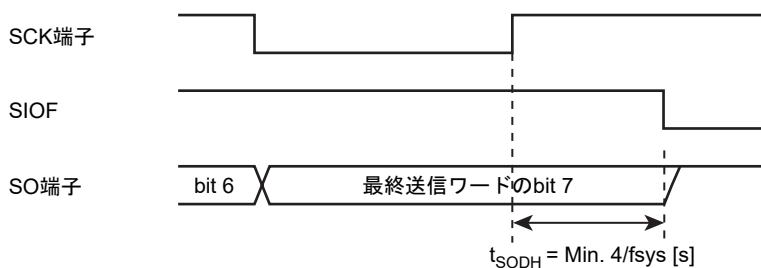


図 10-21 送信終了時の最終ビット保持時間

## 第 11 章 12 ビットアナログ / デジタルコンバータ

TMPM376FDDFG/FDFG は、2 つの 12 ビット逐次変換方式アナログ / デジタルコンバータ (AD コンバータ) を内蔵しています。モーター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ーのベクトル制御を支援します。

AD コンバータユニット A は 12 本のアナログ入力を持っています。3 本はモータ 0 の測定用に使用可能です。外部から入力可能な AD は 12 本です。

AD コンバータユニット B は 13 本のアナログ入力を持っています。3 本はモータ 0 の測定用、および 1 本はモータ 1 の測定用に使用可能です。外部から入力可能な AD は 13 本です。

22 本の外部アナログ入力端子 (AINA0 ~ AINA8、AINA9/AINB0、AINA10/AINB1、AINA11/AINB2、AINB3 ~ AINB12) は、入出力専用ポートと兼用です。

### 11.1 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. ソフトウェア起動、常時起動において任意のアナログ入力を変換することができます。
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. ソフトウェア起動、常時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。

## 11.2 ブロック図

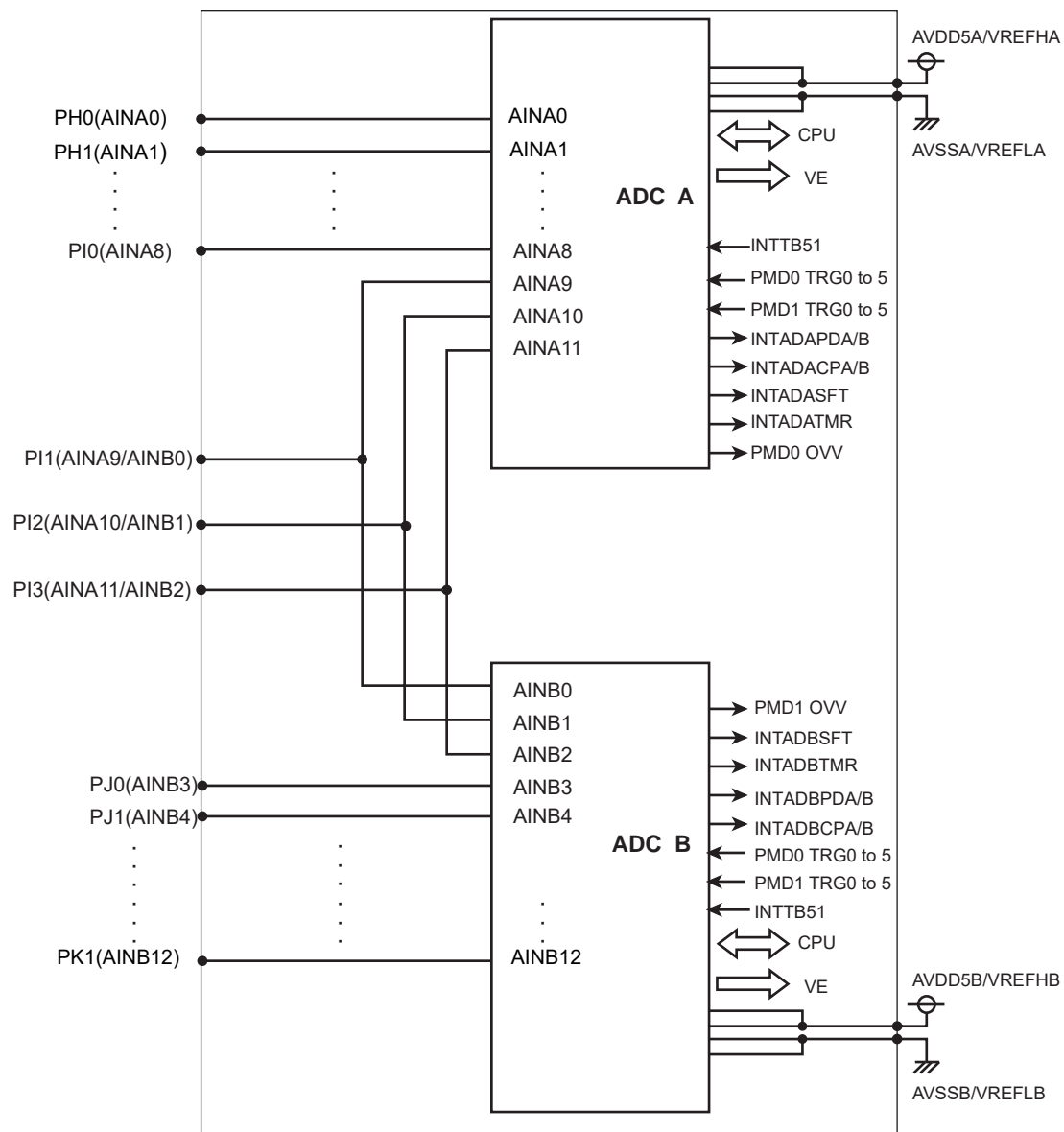


図 11-1 AD コンバータブロック図

## 11.3 レジスタ一覧

ユニット	Base Address
ユニット A	0x4003_0000
ユニット B	0x4003_0200

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

レジスタ名 (x=A,B)		Address(Base+)
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
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
ソフトウェア トリガ用プログラムレジスタ 0 ~ 3	ADxSSET03	0x00BC
ソフトウェア トリガ用プログラムレジスタ 4 ~ 7	ADxSSET47	0x00C0
ソフトウェア 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



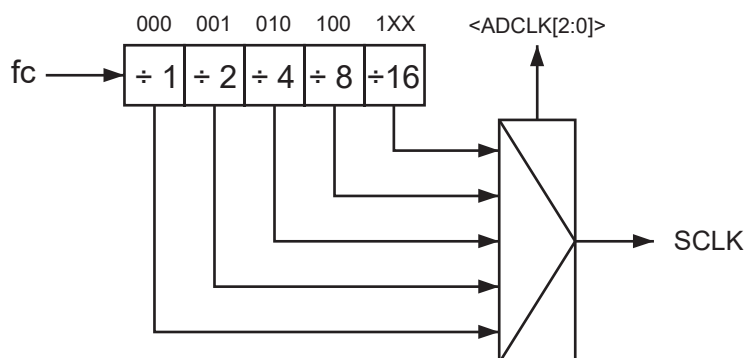
## 11.4 レジスタ詳細

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。

### 11.4.1 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	-	TSH				ADCLK			
リセット後	0	1	0	1	1	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	TSH[3:0]	R/W	"1001" をライトしてください。
2-0	ADCLK[2:0]	R/W	AD 変換クロック選択 000: fc (注 1) 001: fc/2 010: fc/4 011: fc/8 1xx: fc/16



- 注 1) SCLK は最大 40MHz です。したがって、fc が 40MHz を超える場合は <ADCLK[2:0]>="000" を選択しないでください。
- 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。

11.4.2 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	ソフトウェア変換スタート 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。

## 11.4.3 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。この状態の時に <ADAS> の常時 AD 変換許可のセットで AD 変換を開始します。
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

11.4.4 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	ソフトウェア変換フラグ 0: 終了 1: ソフト変換中
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、ソフトウェア変換が開始されると、"1" にセットされ、変換が終了するとフラグが "0" にクリアされます。

## 11.4.5 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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	1	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	PMODE			-	-	-
リセット後	0	1	1	0	0	0	0	0

Bit	Bit Symbol	Type	機能
31-16	-	R	リードすると"0"が読めず。
15-11	-	R/W	"0"をライトしてください。
10	-	R/W	"1"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	-	R/W	"0"をライトしてください。
7	-	R/W	"0"をライトしてください。
6	-	R/W	"1"をライトしてください。
5-3	PMODE[2:0]	R/W	"100"をライトしてください。
2-0	-	R/W	"0"をライトしてください。

注) ADxMOD3<PMODE[2:0]> は必ず "100" に設定してください。その他の bit は初期値の値を変更しないでください。

### 11.4.6 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	CMPOEN	-	-	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	CMPOEN	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 <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>0000: ADxREG0</td> <td>0100: ADxREG4</td> <td>1000: ADxREG8</td> </tr> <tr> <td>0001: ADxREG1</td> <td>0101: ADxREG5</td> <td>1001: ADxREG9</td> </tr> <tr> <td>0010: ADxREG2</td> <td>0110: ADxREG6</td> <td>1010: ADxREG10</td> </tr> <tr> <td>0011: ADxREG3</td> <td>0111: ADxREG7</td> <td>1011: ADxREG11</td> </tr> </table>	0000: ADxREG0	0100: ADxREG4	1000: ADxREG8	0001: ADxREG1	0101: ADxREG5	1001: ADxREG9	0010: ADxREG2	0110: ADxREG6	1010: ADxREG10	0011: ADxREG3	0111: ADxREG7	1011: ADxREG11
0000: ADxREG0	0100: ADxREG4	1000: ADxREG8													
0001: ADxREG1	0101: ADxREG5	1001: ADxREG9													
0010: ADxREG2	0110: ADxREG6	1010: ADxREG10													
0011: ADxREG3	0111: ADxREG7	1011: ADxREG11													

11.4.7 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 <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>0000: ADxREG0</td> <td>0100: ADxREG4</td> <td>1000: ADxREG8</td> </tr> <tr> <td>0001: ADxREG1</td> <td>0101: ADxREG5</td> <td>1001: ADxREG9</td> </tr> <tr> <td>0010: ADxREG2</td> <td>0110: ADxREG6</td> <td>1010: ADxREG10</td> </tr> <tr> <td>0011: ADxREG3</td> <td>0111: ADxREG7</td> <td>1011: ADxREG11</td> </tr> </table>	0000: ADxREG0	0100: ADxREG4	1000: ADxREG8	0001: ADxREG1	0101: ADxREG5	1001: ADxREG9	0010: ADxREG2	0110: ADxREG6	1010: ADxREG10	0011: ADxREG3	0111: ADxREG7	1011: ADxREG11
0000: ADxREG0	0100: ADxREG4	1000: ADxREG8													
0001: ADxREG1	0101: ADxREG5	1001: ADxREG9													
0010: ADxREG2	0110: ADxREG6	1010: ADxREG10													
0011: ADxREG3	0111: ADxREG7	1011: ADxREG11													

11.4.8 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。

11.4.9 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。



## 11.4.10 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。

11.4.11 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。

## 11.4.12 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。

11.4.13 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。

## 11.4.14 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。

11.4.15 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。

## 11.4.16 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。

11.4.17 ADxBREG7( 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。



## 11.4.18 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。

11.4.19 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。

## 11.4.20 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。

11.4.21 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。

### 11.4.22 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。

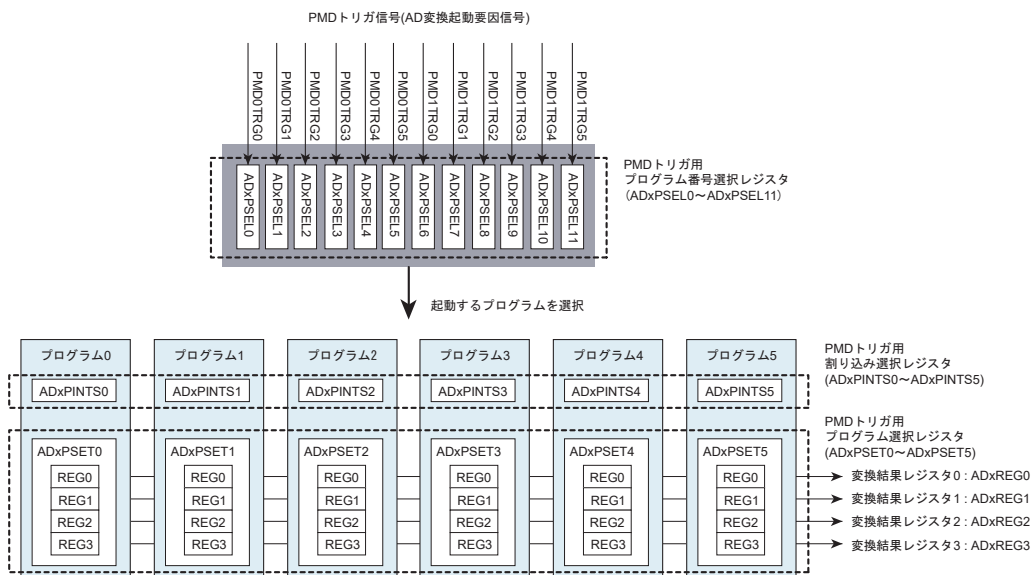


図 11-2 PMD トリガ用プログラムレジスタ

11.4.22.1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 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 11-1 参照)

表 11-1 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

## 11.4.22.2 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	割り込み選択 00: 割り込み出力なし 01: INTADxPDA 10: INTADxPDB 11: 割り込み出力なし  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	割り込み選択 00: 割り込み出力なし 01: INTADxPDA 10: INTADxPDB 11: 割り込み出力なし  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	割り込み選択 00: 割り込み出力なし 01: INTADxPDA 10: INTADxPDB 11: 割り込み出力なし  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	割り込み選択 00: 割り込み出力なし 01: INTADxPDA 10: INTADxPDB 11: 割り込み出力なし  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	割り込み選択 00: 割り込み出力なし 01: INTADxPDA 10: INTADxPDB 11: 割り込み出力なし  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	割り込み選択 00: 割り込み出力なし 01: INTADxPDA 10: INTADxPDB 11: 割り込み出力なし  プログラム 5 に対して、起動する割り込みを選択します。

11.4.22.3 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	m=0	m=1	m=2	m=3
ADxPSETn				
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 11-2 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	:AINA10
0_1011	:AINA11	:AINB11
0_1100	:Reserved	:AINB12
0_1101 ~ 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 AD 変換入力端子の選択」を参照
15	ENSP11	R/W	ADxBREG1 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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	ADxREG3 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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 11-2 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

## 11.4.23 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 5(TMRB5) の INTTB51 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。

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 11-3 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	:AINA10
0_1011	:AINA11	:AINB11
0_1100	:Reserved	:AINB12
0_1101 ~ 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 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 11-3 AD 変換入力端子の選択」を参照

#### 11.4.24 ADxSSET03 / ADxSSET47 / ADxSSET811( ソフトウェアトリガ用プログラムレジスタ )

本 AD 変換はソフトウェアによって AD 変換を開始する事ができます。ソフトウェ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。

ソフトウェアトリガによる 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 11-4 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	:AINA10
0_1011	:AINA11	:AINB11
0_1100	:Reserved	:AINB12
0_1101 ~ 1_1111	:Reserved	

ADxSSET03 : ソフトウェ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 11-4 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 11-4 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 11-4 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 11-4 AD 変換入力端子の選択」を参照

ADxSSET47 : ソフトウェ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 11-4 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 11-4 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 11-4 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 11-4 AD 変換入力端子の選択」を参照

ADxSSET811 : ソフトウェ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 11-4 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 11-4 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 11-4 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 11-4 AD 変換入力端子の選択」を参照

## 11.4.25 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 <ENSA<sub>m</sub>> を 1 にセットすると AD<sub>x</sub>PSET<sub>m</sub> をイネーブルにします。<AINSA<sub>m</sub>[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 11-5 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	:AINA10
0_1011	:AINA11	:AINB11
0_1100	:Reserved	:AINB12
0_1101 ~ 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 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 11-5 AD 変換入力端子の選択」を参照

## 11.5 動作説明

### 11.5.1 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。また VREFHA と VREFLA 間（または VREFHB と VREFLB 間）に流れる電流を制御するレジスタは無く、一定の電流が流れ続けます。

- 注 1) 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/I/J/K の出力データを書き換えないようにしてください。
- 注 2) 電源電圧の変動や周囲のノイズの影響によって AD 変換結果がばらつくことがあります。AD 変換中に AD 入力に兼用となっている端子への入力および端子出力を変化させる、または AD 変換中に出力ポートに設定している端子の出力電流が変動すると AD 変換精度が低下することがあります。プログラムで複数回の平均値をとるなどして対策してください。

### 11.5.2 AD 変換開始

AD コンバータはソフトウェアによる起動 ( ソフトウェアトリガ ) 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 > ソフトウェア > 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した時は、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。

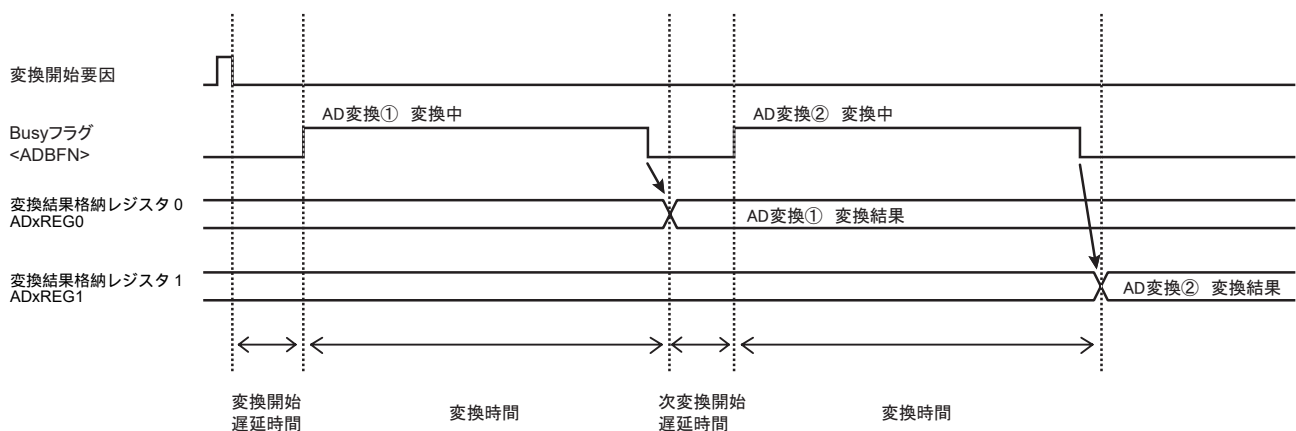


図 11-3 AD 変換開始動作タイミングチャート

表 11-6 AD 変換時間と変換開始遅延時間 (SCLK = 40MHz)

	変換開始要因	fsys = 80MHz		fsys = 40MHz	
		MIN	MAX	MIN	MAX
開始遅延時間 [μs] (注 1)	PMD	0.125	0.163	0.225	0.3
	TMRB	0.125	0.263	0.225	0.5
	ソフトウェア、 常時変換	0.138	0.275	0.25	0.525
AD 変換時間 [μs]	-	1.85		1.85	
次変換開始遅延時間 [μs](注 2)	PMD	0.1	0.125	0.175	0.225
	TMRB、ソフトウェア、 常時変換	0.1	0.238	0.175	0.425

注 1) 変換開始要因発生から AD 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

### 11.5.3 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。

AD<sub>x</sub>BCMPCR0<CMP0EN> または AD<sub>x</sub>CMPCR1<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ことにおこなわれ、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。

## 11.6 AD 変換タイミングチャート

ソフトウェア 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。

### 11.6.1 ソフトウェア AD 変換

ソフトウェア AD 変換では、AD<sub>x</sub>SSET03, AD<sub>x</sub>SSET47, AD<sub>x</sub>SSET811 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 11-4)。

ソフトウェア AD 変換中に AD<sub>x</sub>MOD1<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 11-5)。

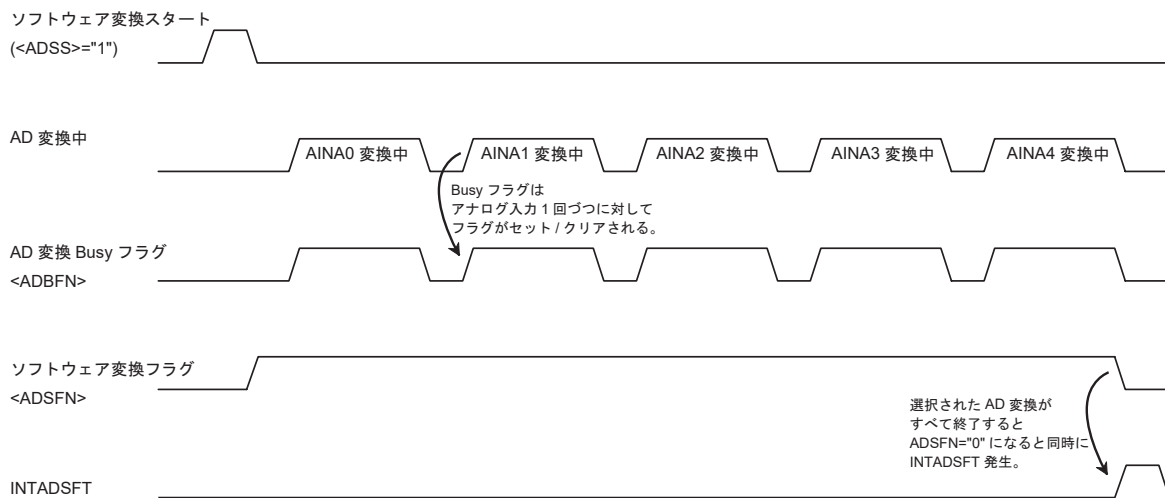
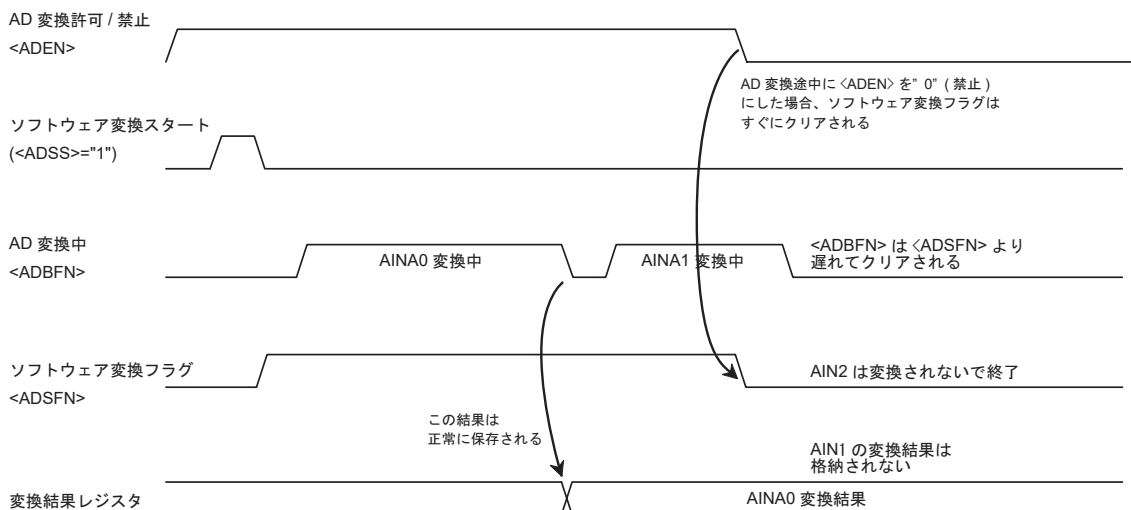


図 11-4 ソフトウェア AD 変換時のタイミングチャート

[設定条件]

ソフトウェアトリガ設定: AINA0, AINA1, AINA2



[ 設定条件 ]

ソフトウェアトリガ設定 : AINB10 , AINB11 , AINB12

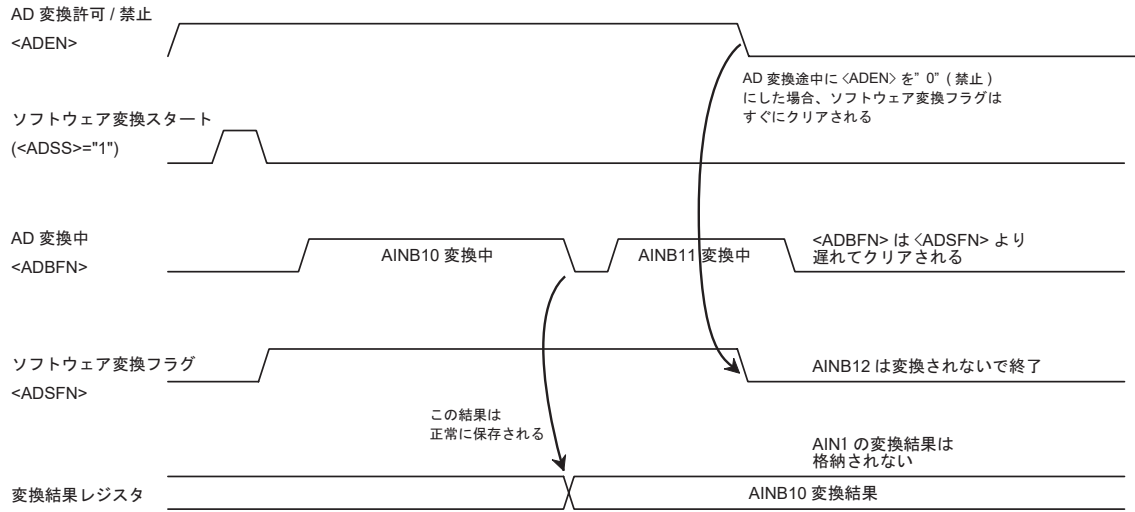


図 11-5 ソフトウェア AD 変換中に <ADEN> = "0" 書き込み

### 11.6.2 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 11-6 )。

[ 設定条件 ]

常時変換設定 : AINA0

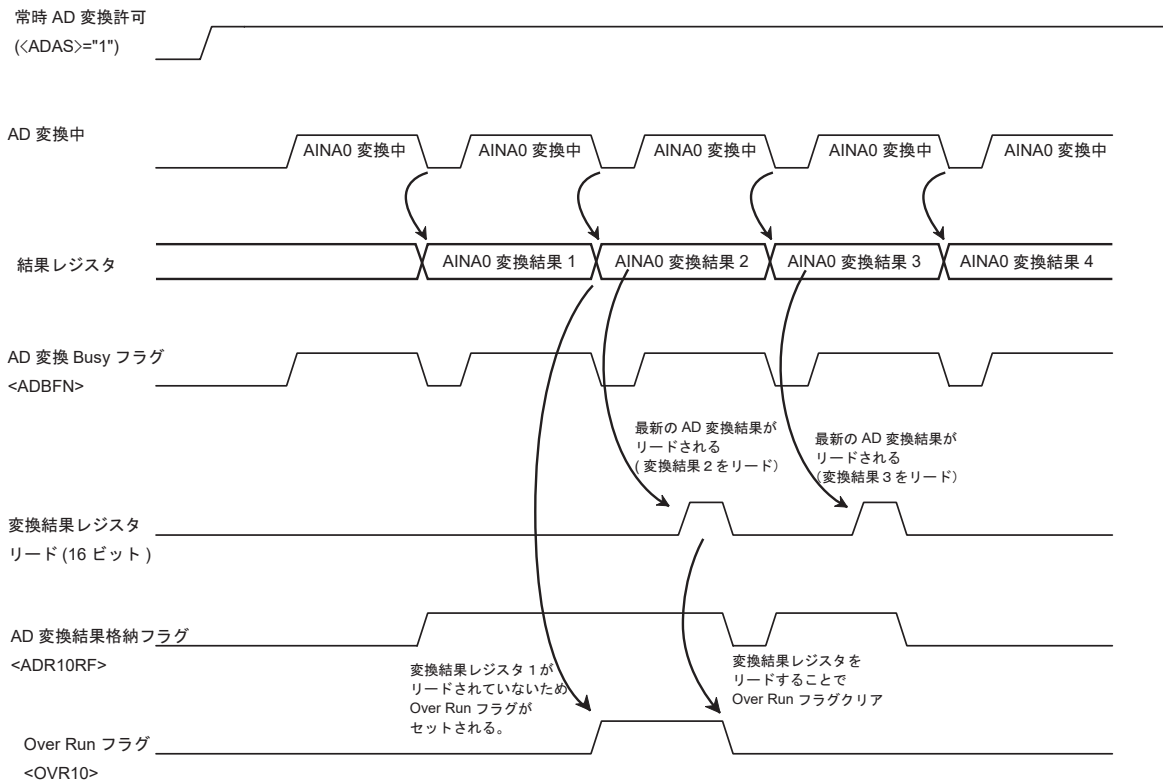


図 11-6 常時 AD 変換時のタイミングチャート

### 11.6.3 トリガによる AD 変換開始

ソフトウェアAD変換実行中にPMDトリガが発生した場合、実行中のソフトウェア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11-7)。PMDトリガによるAD変換が終了後、ソフトウェア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11-8)。

**【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

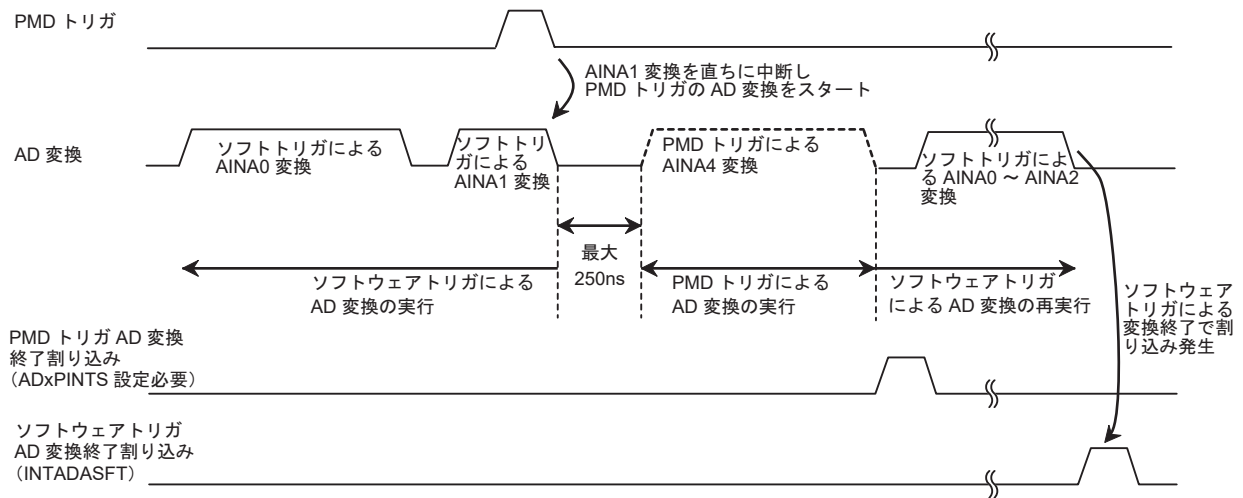


図 11-7 ソフトウェア AD 変換中の PMD トリガ発生



【条件設定】  
 ソフトウェアトリガ設定: AINB10、AINB11、AINB12  
 タイマトリガ設定: AINB2

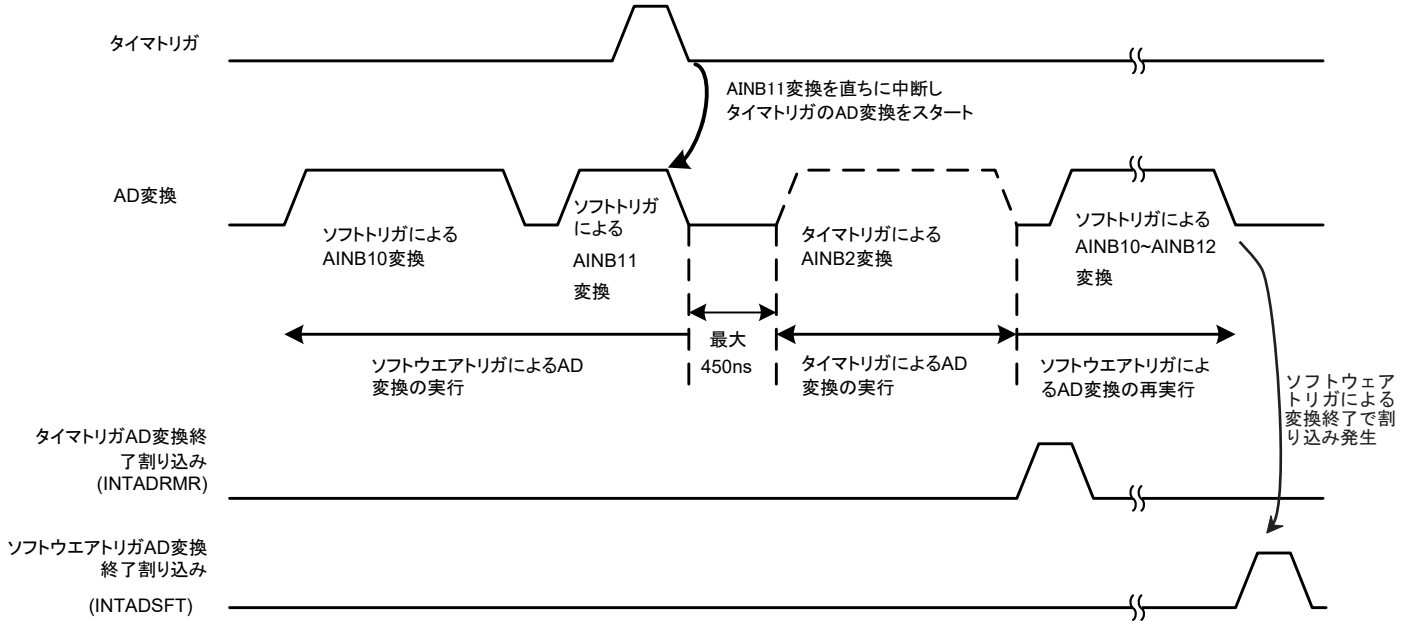


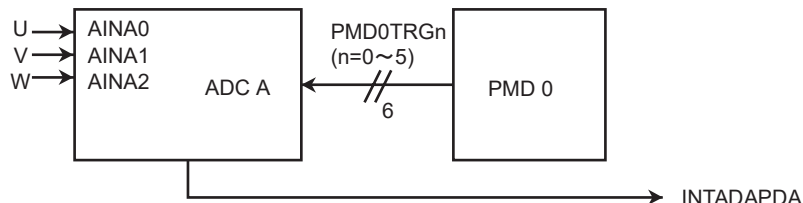
図 11-8 ソフトウェア AD 変換中のタイマトリガ発生

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。

## 11.7 使用方法の例

### 11.7.1 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

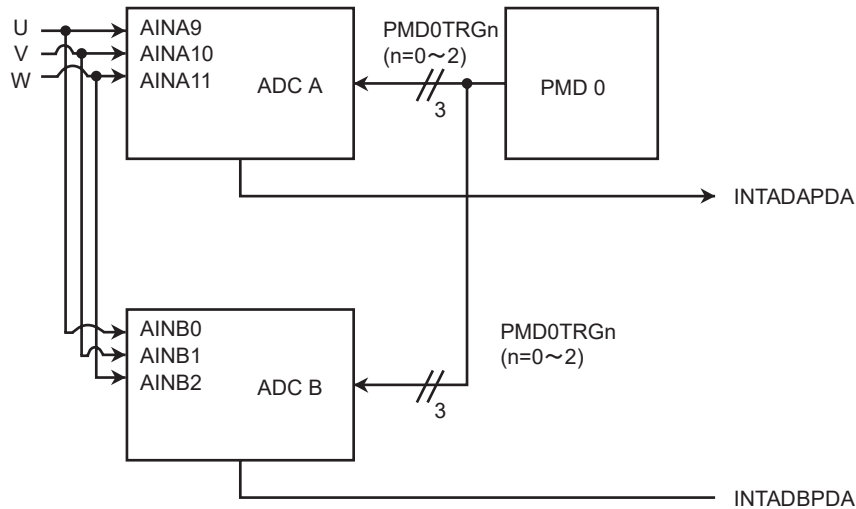
6 本のトリガ入力 PMD0TRG0 ~ 5 に対し、ADAPSEL0 ~ 5 でそれぞれ 0 ~ 5 のプログラム番号を割り付けます。

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」はモーター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。

11.7.2 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	U	V	W
INT	A	A	A

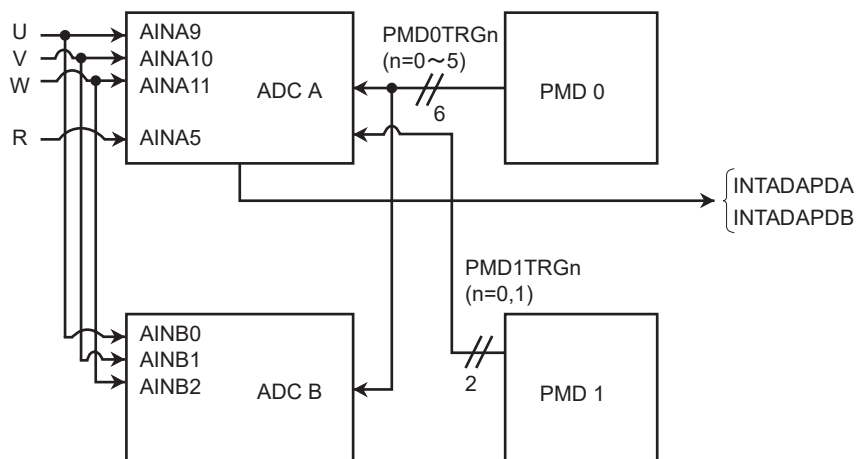
ADコンバータ ユニットA、ADコンバータ ユニットBに入力される3本のトリガ入力に、ADAPSEL0 ~ 2 および ADBPSEL0 ~ 2 でそれぞれ 0 ~ 2 のプログラム番号を割り付けます。

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」はモーター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。

### 11.7.3 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	PMD0 V	PMD0 W	PMD0 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。

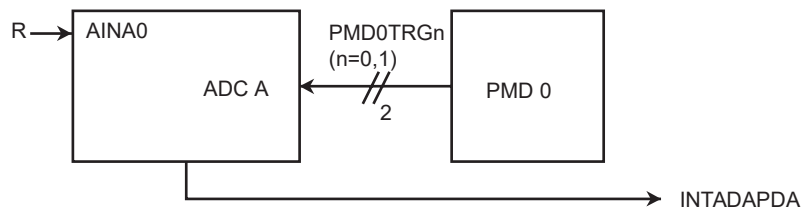
表中の reg0、1、2 は ADxPSETn[7:0]、ADxPSETn[15:8]、ADxPSETn[23:16] を表します。(x:ADC ユニット, n: プログラム番号) 表中の「U」、「V」、「W」はモーターの位相を表しており、それぞれの位相が得られる AIN 入力を選択します。「R」は抵抗を表しており、接続されている AIN を設定します。

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。

### 11.7.4 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
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 割り込みが発生します。



## 第 12 章 モータ制御回路 (PMD : Programmable Motor Driver)

TMPM376FDDFG/FDFG はモータ制御回路 (PMD) を 2 チャンネル内蔵しています。

本製品の PMD は 1 シャントセンサレスモータ制御を実現する為に通電出力制御や、DC 過電圧検出入力を追加し、ADC を連携させたモータ制御を可能としています。

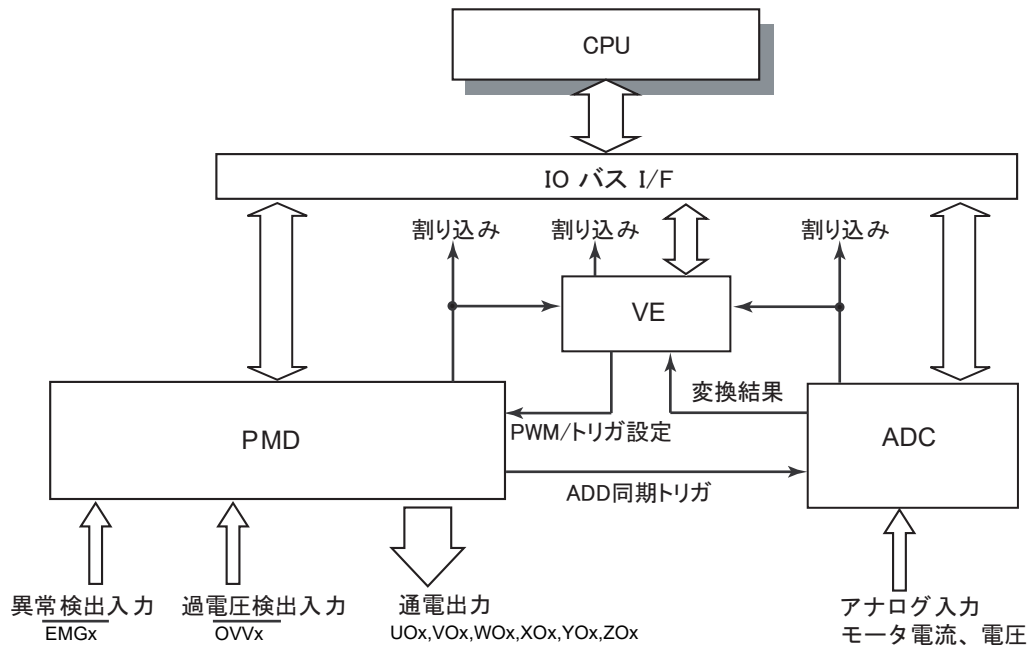


図 12-1 モータ制御関連ブロック図

## 12.1 PMD 回路への入出力信号の動作説明

モータ制御回路のチャンネル別の入力信号及び出力信号は以下の通りです。

表 12-1 入出力信号一覧

CH	端子名	PMD 信号名	機能
PMD0	PC7/ $\overline{\text{OVV0}}$	OVV 0	OVV 状態出力
	PC6/ $\overline{\text{EMG0}}$	EMG 0	EMG 状態信号
	PC0/UO0	UO 0	U 相出力
	PC1/XO0	XO 0	X 相出力
	PC2/VO0	VO 0	V 相出力
	PC3/YO0	YO 0	Y 相出力
	PC4/WO0	WO 0	W 相出力
	PC5/ZO0	ZO 0	Z 相出力
PMD1	PG7/ $\overline{\text{OVV1}}$	OVV 1	OVV 状態出力
	PG6/ $\overline{\text{EMG1}}$	EMG 1	EMG 状態信号
	PG0/UO1	UO 1	U 相出力
	PG1/XO1	XO 1	X 相出力
	PG2/VO1	VO 1	V 相出力
	PG3/YO1	YO 1	Y 相出力
	PG4/WO1	WO 1	W 相出力
	PG5/ZO1	ZO 1	Z 相出力



## 12.2 PMD 回路

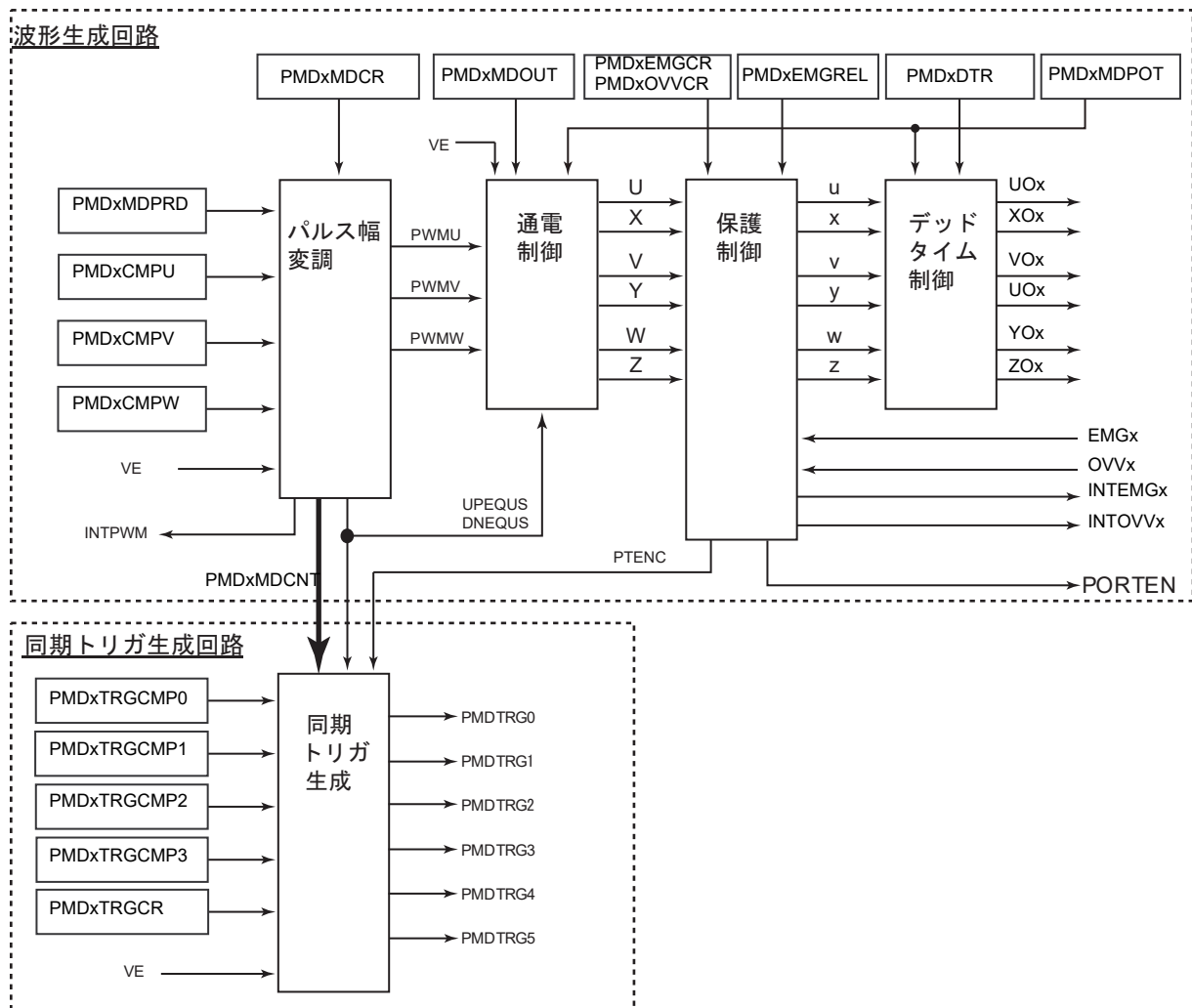


図 12-2 PMD 回路概略図

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周波数が等しい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いません。
- 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。

## 12.3 PMD レジスタ一覧

各チャンネルのレジスタとアドレスを示します。

Channel x	Base Address
Channel 0	0x4005_0400
Channel 1	0x4005_0480

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 出力コントロールレジスタ	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
Reserved	-	0x007C

注) Reserved にはアクセスしないでください。

## 12.3.1 PMDxDEN(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	<p>波形生成機能の許可 / 禁止を制御。</p> <p>0: 禁止 1: 許可</p> <p>ポートを機能出力 (PWM 出力) として設定している場合、&lt;PWMEN&gt;="0"( 禁止 ) にすると出力ポートは Hi-z になります。</p> <p>出力ポート極性等、&lt;PWMEN&gt; 以外の初期設定を行った後に、&lt;PWMEN&gt;="1"( 許可 ) を設定して下さい。</p>

12.3.2 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	PORT 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  <PORTMD[1:0]> の設定により外部 PORT に対して上相 (U/V/W 相) および下相 (X/Y/Z 相) のポート出力制御信号を出力します。"High-z" 選択時にツールブレイクが発生した場合、外部出力ポートの上下相を Hi-z に設定します。それ以外の場合は PMD 出力に従います。

注 1) <PWMEN>=0 時は出力ポートの設定によらず出力ポートを Hi-z に設定します。

注 2) 外部ポート出力制御は PMDxEMGCR<EMGMD[1:0]> の設定により EMG 入力時にも行なわれます。

## 12.3.3 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	-	-	-	-	-	-	-	MDESEL
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	MDESEL	R/W	<p>モード選択レジスタ</p> <p>0: バスモード</p> <p>1: VE モード</p> <p>&lt;MDESEL&gt; は、ダブルバッファ後段へ入力するデータを、バスから設定したレジスタ値を使用する (バスモード)、またはベクトルエンジン制御部 (VE) からの値を使用する (VE モード) を選択します。尚、コンペアレジスタ (PMDxCMPU、PMDxCMPV、PMDxCMPW)、トリガコンペアレジスタ (PMDxTRGCMP0、PMDxTRGCMP1)、PMDxMDOUT レジスタはダブルバッファ構成となっており、PMD 内部の更新タイミングでダブルバッファ後段へ入力データがサンプリングされます。</p>

### 12.3.4 パルス幅変調回路

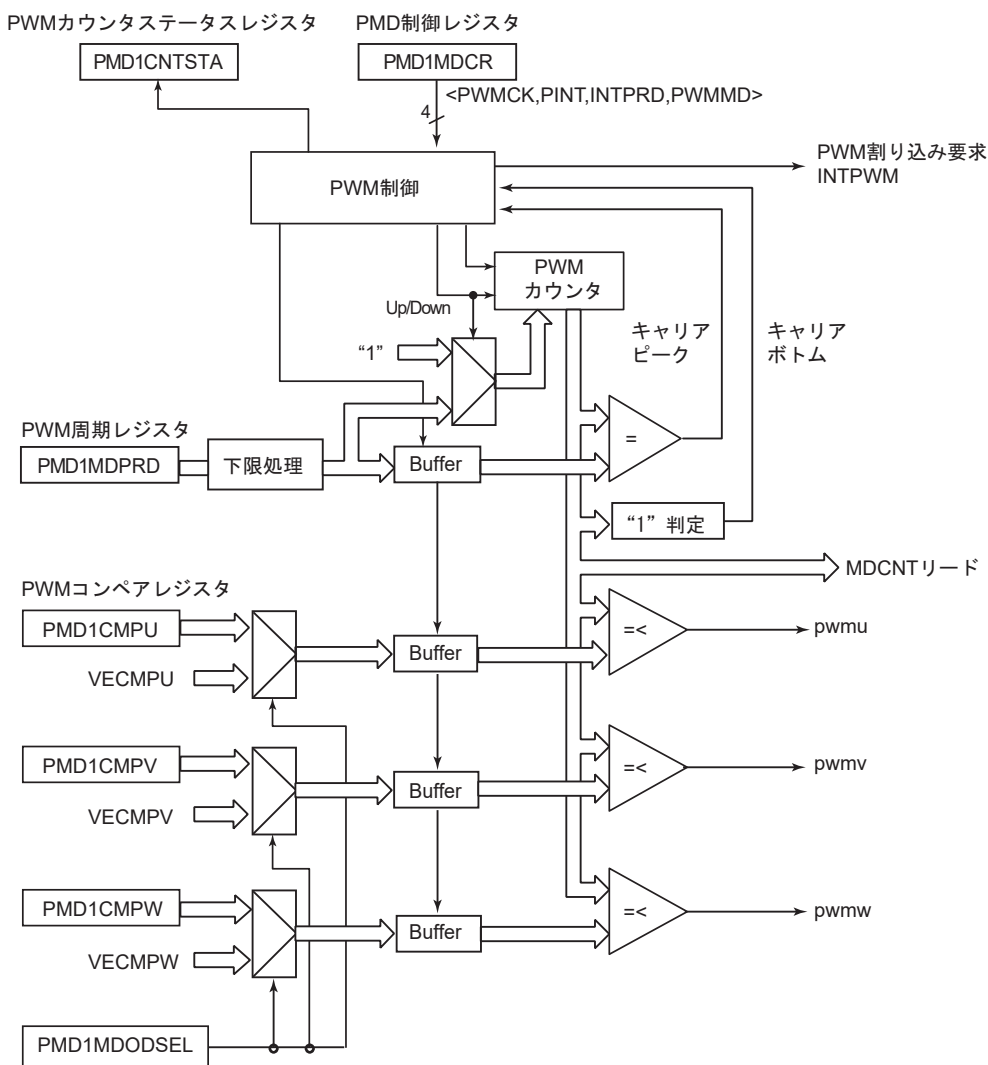


図 12-3 パルス幅変調回路概略図

パルス幅変調回路は、16bit のアップ/ダウンカウンタであるPMDカウンタを持ち、12.5nsec@80MHz 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 としてセンター PWM( 三角波変調) を選択可能です。

また、PWM 周期延長モード (<PWMCK>=1) にすることにより、PWM カウンタは 50nsec の分解能で PWM キャリアを生成します。

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、コンパレータ入力は 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。

$$\text{のこぎり波 PWM : PMDxMDPRD レジスタ設定} = \frac{\text{発振周波数[Hz]}}{\text{PWM周波数[Hz]}}$$

$$\text{三角波変調 PWM : PMDxMDPRD レジスタ設定値} = \frac{\text{発振周波数[Hz]}}{\text{PWM周波数[Hz]} \times 2}$$

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、PMD カウンタ <MDCNT[15:0]> が生成する搬送波と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 PMD コンペアレジスタは比較レジスタを持ち、ダブルバッファ構成となります。PMD コンペアレジスタの値は PWM 周期に同期して (内部カウンタ値が <MDPRD[15:0]> と一致時) 比較レジスタにロードされます。

PWM 半周期での更新 (半周期毎ロード) も選択できます。

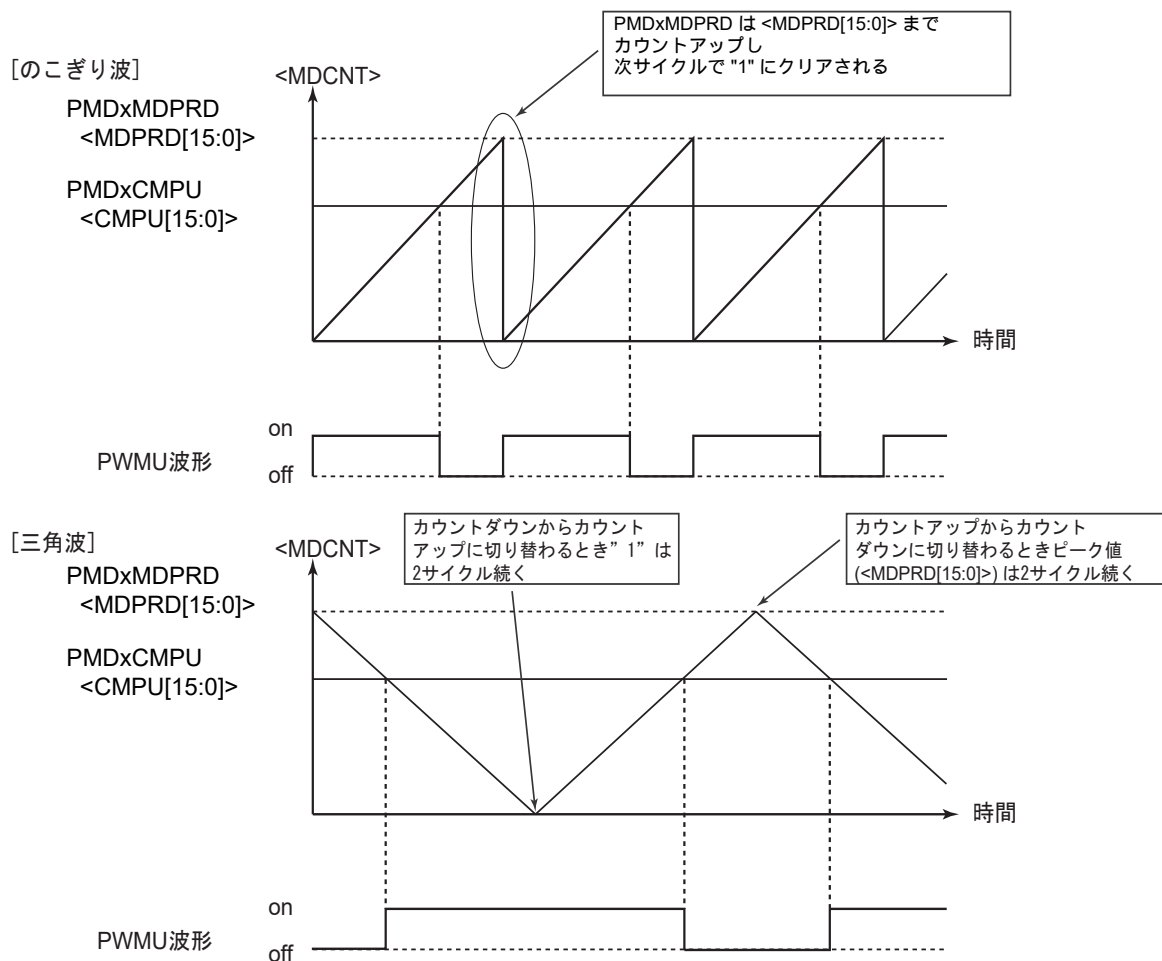


図 12-4 PWM 波形

### 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 PMD 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 PMD 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。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 回を選択する事ができます。



12.3.4.1 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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	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
31-7	-	R	リードすると "0" が読めます。
6	PWMCK	R/W	<p>PWM 周期延長モードを指定。</p> <p>0: 通常周期 1: 4 倍周期</p> <p>通常設定時、PWM カウンタは分解能 12.5ns @fsys=80MHz で動作します。 ノコギリ波 12.5ns、三角波 25ns</p> <p>4 倍周期設定時、PWM カウンタは分解能 50ns@2bit カウンタ (fsys=80MHz 動作) で動作します。 ノコギリ波 50ns、三角波 100ns</p>
5	SYNTMD	R/W	<p>ポート出力モード設定。</p> <p>U,V,W 相のポート出力設定を行います。 注) 表 12-2 を参照してください。</p>
4	DTYMD	R/W	<p>DUTY モード選択。</p> <p>0: 3 相共通 1: 3 相独立</p> <p>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。</p>
3	PINT	R/W	<p>PWM 割り込みタイミング選択。</p> <p>0: PWM カウンタ PMDxMDCNT&lt;MDCNT[15:0]&gt; = 0x0001 のとき (最小) 割り込み要求 1: PWM カウンタ PMDxMDCNT&lt;MDCNT[15:0]&gt; = PMDxMDPRD&lt;MDPRD[15:0]&gt; のとき割り込み要求 (&lt;PWMMD&gt;="0" のときは &lt;MDPRD[15:0]&gt; のとき、&lt;PWMMD&gt;="1" 選択時は 1、&lt;MDPRD[15:0]&gt; 両方時。)</p> <p>割り込み発生タイミングを PWM カウンタが MIN 時か MAX 時かを選択可能。エッジモード選択時は &lt;MDPRD[15:0]&gt; のとき、0.5 周期選択時は 1、&lt;MDPRD[15:0]&gt; 両方時となります。</p>
2-1	INTPRD[1:0]	R/W	<p>PWM 割り込み周期選択。</p> <p>00: PWM 0.5 周期毎に割り込み (&lt;PWMMD&gt; = "1" (三角波) の場合のみ設定可能です) 01: PWM 1 周期毎に割り込み 10: PWM 2 周期毎に割り込み 11: PWM 4 周期毎に割り込み</p> <p>PWM 割り込みの発生する頻度を PWM 周期 0.5 周期 /1 周期 /2 周期 /4 周期に 1 回から選択します。 注) "00" 設定時、コンペアレジスタ (PMDxCMPU/V/W)、周期レジスタ (PMDxMDPRD) は内部カウンタが 1 または PMDxMDPRD 一致時にダブルバッファ更新されます。</p>
0	PWMMD	R/W	<p>PWM キャリア波形選択。</p> <p>0: PWM モード 0 (エッジ PWM、ノコギリ波) 1: PWM モード 1 (センター PWM、三角波)</p> <p>PWM のモードを選択します。PWM モード 0 はエッジ PWM、PWM モード 1 はセンター PWM になります。</p>

12.3.4.2 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。 カウントアップ中は、"0" が読み出されます。 エッジ PWM モードを選択した場合、常に "0" が読み出されます。

12.3.4.3 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 周期カウント。 PWM 周期をカウントする 16bit のレジスタで読み出し専用です。 PMD カウンタ値 12.5ns 分解能 @fsys=80MHz ノコギリ波 12.5ns、三角波 25ns  PMDxMDCR<PWMCK> を 1 に設定した場合、50ns 分解能となります。  PMD ディセーブル時 (<PWMEN> = 0) の PWM カウンタ値は <PWMMMD> 設定 (キャリア波形選択) によって、以下ようになります。 PMDxMDCR<PWMMMD>=0 の場合 :0x0001 PMDxMDCR<PWMMMD>=1 の場合 :PMDxMDPRD<MDPRD[15:0]> 値

12.3.4.4 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 <MDPRD[15:0]> は PWM 周期を決定するレジスタで、ダブルバッファ構成となっています。したがって、PMD カウンタの動作中でも変更する事が出来ます。バッファへは PWM 周期毎にロードされます。(内部カウンタが <MDPRD[15:0]> と一致した時にロード。半周期選択時 (<INTPRD[1:0]>="00" 設定時) は "1"、<MDPRD[15:0]> 時にロード。最下位ビットは 0 を設定して下さい。) <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) 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障できません。

12.3.4.5 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	CMPUx							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMPUx							
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	CMPUx[15:0]	R/W	<p>PWM パルス幅の設定。                      コンペアレジスタ 12.5ns 分解能 @fsys=80MHz。                      ノコギリ波 12.5ns、三角波 25ns</p> <p>&lt;PWMCK&gt; を 1 に設定した場合、50ns 分解能となります。</p> <p>&lt;CMPUx [15:0]&gt; は、U 相の出力する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。バッファと PMD カウンタとを大小比較しパルス幅を決定します。( 内部カウンタが &lt;MDPRD[15:0]&gt; と一致時にロード。半周期選択時 (&lt;INTPRD[1:0]&gt;="00" 設定時) は "1", &lt;MDPRD[15:0]&gt; 時にロード。)                      リード時はダブルバッファ前段の値 ( バスから設定したデータ ) をリードします。</p>

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 の bit0 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障できません。

12.3.4.6 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	CMPVx							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMPVx							
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	CMPVx[15:0]	R/W	<p>PWM パルス幅の設定。                      コンペアレジスタ 12.5ns 分解能 @fsys=80MHz。                      ノコギリ波 12.5ns、三角波 25ns</p> <p>&lt;PWMCK&gt; を 1 に設定した場合、50ns 分解能となります。</p> <p>&lt;CMPVx [15:0]&gt; は、V 相の出力する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。バッファと PMD カウンタとを大小比較しパルス幅を決定します。( 内部カウンタが &lt;MDPRD[15:0]&gt; と一致時にロード。半周期選択時 (&lt;INTPRD[1:0]&gt;="00" 設定時) は 1, &lt;MDPRD[15:0]&gt; 時にロード。) リード時はダブルバッファ前段の値 (バスから設定したデータ) をリードします。。                      リード時はバッファの値 (バスから設定したデータ) をリードします。</p>

- 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 の bit0 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障できません。

12.3.4.7 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	CMPWx							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	CMPWx							
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	CMPWx[15:0]	R/W	<p>PWM パルス幅の設定をします。                      コンペアレジスタ 12.5ns 分解能 @fsys=80MHz。                      ノコギリ波 12.5ns、三角波 25ns</p> <p>&lt;PWMCK&gt; を 1 に設定した場合、50ns 分解能となります。</p> <p>&lt;CMPWx[15:0]&gt; は、W 相の出力する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。バッファと PMD カウンタとを大小比較しパルス幅を決定します。(内部カウンタが &lt;MDPRD[15:0]&gt; と一致時にロード。半周期選択時 (&lt;INTPRD[15:0]&gt;="00" 設定時) は 1, &lt;MDPRD[15:0]&gt; 時にロード。) リード時はダブルバッファ前段の値 (バスから設定したデータ) をリードします。。                      リード時はバッファの値 (バスから設定したデータ) をリードします。</p>

- 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 の bit0 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障できません。

## 12.3.5 通電制御回路

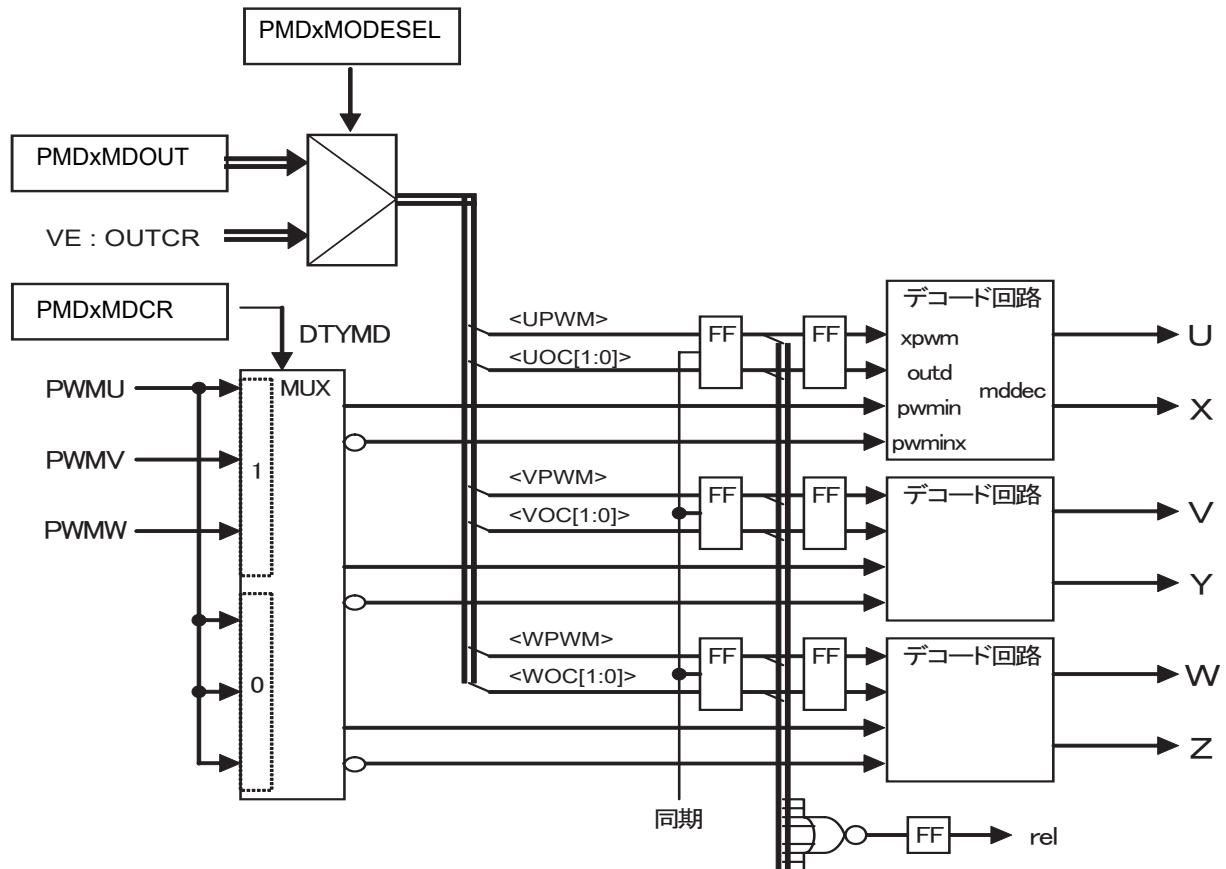


図 12-5 通電制御回路概略図

"PMDxMDOUT" に設定された内容により、出力ポートの制御を行います。設定内容は、ポート出力時の同期信号の選択とポート出力設定に分けられます。ポート出力設定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。

6 本のポートの出力設定は、それぞれ独立にアクティブ / インアクティブの設定を PMDxMDPOT <POLH><POLL> により行います。さらに、U,V,W の 3 相それぞれに、PWM 出力と High/Low 出力との選択を PMDxMDOUT<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。PMDxMDOUT によるポート出力設定と PMD 制御レジスタ (PMDxMDCR) の極性設定によって得られる端子出力の関係については、表 12-2 を参照してください。

12.3.5.1 PMD<sub>x</sub>MDPOT (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	-	-	-	-	-	-	-	-
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
31-4	-	R	リードすると "0" が読めます。
3	POLH	R/W	上相の出力ポート極性選択。(注) 0: ロー・アクティブ 1: ハイ・アクティブ
2	POLL	R/W	下相の出力ポート極性選択。(注) 0: ロー・アクティブ 1: ハイ・アクティブ
1-0	PSYNCS[1:0]	R/W	MDO <sub>UT</sub> 設定転送タイミング選択。(注) 00: PWM 非同期 01: PWM カウンタ <MDCNT> = 1 の時リロード 10: PWM カウンタ <MDCNT> = PMD <sub>x</sub> MDPRD<MDPRD[15:0]> の時リロード 11: PWM カウンタ <MDCNT> = 1 および PMD <sub>x</sub> MDPRD<MDPRD[15:0]> の時リロード  U,V,W 相出力設定のポート出力反映時のタイミングを選択します。 "00" (PWM 非同期) を選択した場合、PMD <sub>x</sub> MDO <sub>UT</sub> の更新と同時にポート出力が反映されます。なお、ベクトルエンジンからの VEOUTCR <sub>x</sub> に対しても有効です。(MDCNT のピーク / ボトム同期又は非同期を選択)

注) PMD<sub>x</sub>MDEN<PWMEN>=0 の状態で選択を行って下さい。



12.3.5.2 PMD<sub>x</sub>MDOUT(PMD 出力コントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	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	UVW 相出力制御。 0: H/L 出力 1: PWM 出力 詳細は表 12-2 を参照してください。
9	VPWM	R/W	
8	UPWM	R/W	
7-6	-	R	リードすると"0"が読めます。
5-4	WOC[1:0]	R/W	UVW 相出力制御。 詳細は表 12-2 を参照してください。
3-2	VOC[1:0]	R/W	
1-0	UOC[1:0]	R/W	

- 注 1) バスから更新した PMD<sub>x</sub>MDOUT をダブルバッファ後段にロードさせる場合は、PMD<sub>x</sub>MODESEL の bit0 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障できません。

表 12-2 <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						PMDxMDCR<SYNTMD>=0					
極性ハイアクティブ (PMDxMDPOT<POLH><POLL>="11")						極性ローアクティブ (PMDxMDPOT<POLH><POLL>="00")					
PMDxMDOUT 出力制御		<WPWM><VPWM><UPWM> 出力選択				PMDxMDOUT 出力制御		<WPWM><VPWM><UPWM> 出力選択			
上相	下相	0:H/L 出力		1: PWM 出力		上相	下相	0:H/L 出力		1: PWM 出力	
<WOC[1]> <VOC[1]> <UOC[1]>	<WOC[0]> <VOC[0]> <UOC[0]>	上相 出力	下相 出力	上相 出力	下相 出力	上相 出力	下相 出力	上相 出力	下相 出力	上相 出力	下相 出力
0	0	L	L	$\overline{\text{PWM}}$	PWM	0	0	H	H	PWM	$\overline{\text{PWM}}$
0	1	L	H	L	PWM	0	1	H	L	H	$\overline{\text{PWM}}$
1	0	H	L	PWM	L	1	0	L	H	$\overline{\text{PWM}}$	H
1	1	H	H	PWM	$\overline{\text{PWM}}$	1	1	L	L	$\overline{\text{PWM}}$	PWM

PMDxMDCR<SYNTMD>=1						PMDxMDCR<SYNTMD>=1					
極性ハイアクティブ (PMDxMDPOT<POLH><POLL>="11")						極性ローアクティブ (PMDxMDPOT<POLH><POLL>="00")					
PMDxMDOUT 出力制御		<WPWM><VPWM><UPWM> 出力選択				PMDxMDOUT 出力制御		<WPWM><VPWM><UPWM> 出力選択			
上相	下相	0:H/L 出力		1: PWM 出力		上相	下相	0:H/L 出力		1: PWM 出力	
<WOC[1]> <VOC[1]> <UOC[1]>	<WOC[0]> <VOC[0]> <UOC[0]>	上相 出力	下相 出力	上相 出力	下相 出力	上相 出力	下相 出力	上相 出力	下相 出力	上相 出力	下相 出力
0	0	L	L	$\overline{\text{PWM}}$	PWM	0	0	H	H	PWM	$\overline{\text{PWM}}$
0	1	L	H	L	$\overline{\text{PWM}}$	0	1	H	L	H	PWM
1	0	H	L	PWM	L	1	0	L	H	$\overline{\text{PWM}}$	H
1	1	H	H	PWM	$\overline{\text{PWM}}$	1	1	L	L	$\overline{\text{PWM}}$	PWM

1 シャント電流検出対応の出力設定

1 シャント電流の検出は下記のように設定することで対応可能です。

表 12-3 1 シャント電流の検出設定一覧

	通常	U 相 PWM シフト	V 相 PWM シフト	W 相 PWM シフト
CMPU	duty_U	<MDPRD[15:0]>-duty_U	duty_U	duty_U
CMPV	duty_V	duty_V	<MDPRD[15:0]>-duty_V	duty_V
CMPW	duty_W	duty_W	duty_W	<MDPRD[15:0]>-duty_W
<UOC[1:0]>	11	00	11	11
<VOC[1:0]>	11	11	00	11
<WOC[1:0]>	11	11	11	00

12.3.6 保護制御回路

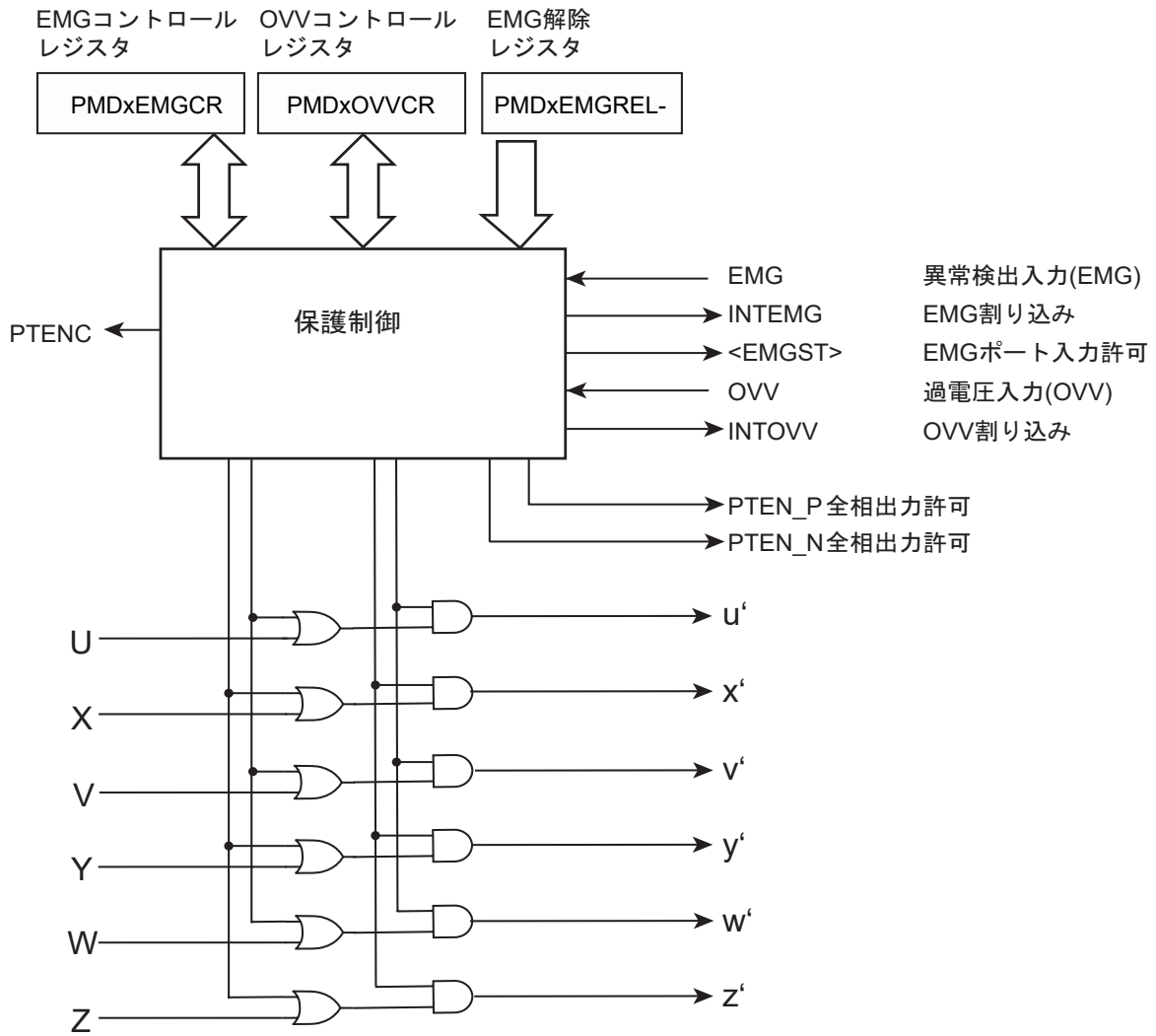


図 12-6 保護制御回路概略図

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。

### 12.3.6.1 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 入力が高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。

ツールブレイクによる PMD 停止制御時も同様に 6 本の PWM 出力を禁止します、但し <PORTM> の設定によります。ツールブレイク時は 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出来ます。

EMG 保護は EMG 制御レジスタ (PMDxEMGCR) で設定します。

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 保護状態の時は、ポート出力を全てインアクティブに設定 (PMDxMDOUT<[10:8]>,<[5:0]>:0) 後、PMDxEMGCR<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 保護入力が高レベルに落ちている間は、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高レベル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 <EMGREL[7:0]> に所定のキーコード 0x5A、0xA5 を設定する事ではじめて可能になり、誤って EMG 保護回路を禁止する事を防止します。

#### 注) 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 機能を選択する。
- 2: PMDxEMGSTA<EMGI> をリードし、"1" であることを確認する。
- 3: PMDxMDOUT<[10:8]>、<[5:0]> を "0" に設定し、ポート出力を全てインアクティブ ("Low" 出力) とする。
- 4: PMDxEMGCR<EMGRS> を "1" に設定し、EMG 保護状態から復帰させる。

なお、EMG 保護を禁止とする場合は、継続して以下の手順でおこなう。

- 5: PMDxEMGREL に禁止コードをライトする (0x5A → 0xA5 の順)
- 6: PMDxEMGCR<EMGEN> に "0" を設定し、EMG 保護回路を禁止にする。

## 12.3.6.2 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 禁止コードを書き込みます。 <EMGREL[7:0]> に "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出来ます。 この場合、<EMGEN> = "0", <OVVEN> = "0" に設定してください。 * EMG, OVV 共通

12.3.6.3 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 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 200[nsec] @80MHz) <EMGCNT[3:0]> = 0 ~ 15(0 設定時はノイズフィルタをスルーする)
7-6	-	R	リードすると "0" が読めます。
5	INHEN	R/W	ツールブレイクの許可 / 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。初期状態では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: PWM 出力制御なし / PORT 出力 全相 High-z 01: 全上相オン、全下相オフ / PORT 出力 下相 High-z 10: 全上相オフ、全下相オン / PORT 出力 上相 High-z 11: 全相オフ / PORT 出力 全相 High-z  オン = PWM 出力 (出力制御なし) オフ = Low (ハイアクティブ (POLL/H=1) 時)  EMG 発生時に、上相および下相の PWM 出力をオン / オフさせます。 また、EMG 発生時の PORT 出力制御を行いません。
2	-	R/W	常に "0" を書いてください。
1	EMGRS	R/W	EMG 保護状態からの復帰 0: - 1: 保護状態からの復帰 PMDxMDOUT レジスタに 0x0 を設定後、<EMGRS> に "1" を設定する事により EMG 保護状態から復帰します。 この bit はリードすると常に 0 が読まれます。 注) PMDxMDOUT レジスタには、必ず上位 bit[10-8], 下位 bit[5-0] 両方に 0 を書き込んでください。 注) PMDxEMGSTA<EMGI> が High 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 1 を設定する事により EMG 保護回路が動作状態となります。初期状態では許可となっています。 禁止するときは禁止コードを PMDxEMGREL<EMGREL[7:0]> に "0x5A" → "0xA5" を順にライトした後、<EMGEN> に "0" を設定します。(3 命令を連続して行います。)

## 12.3.6.4 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	-	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 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 保護の状態モニタ 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。

## 12.3.6.5 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出来ます。

OVV 保護は 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 保護回路動作後、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 Low に落ちている間は、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<OVVLI> をリードすることで確認することが出来ます。)

OVV から復帰するタイミングは、PWM 周期 (PWM カウントと <MDPRD[15:0]> との一致時。ただし PWM0.5 周期割り込み設定時は 1 または <MDPRD[15:0]> と一致時) となります。また、OVV 機能を禁止するには <EMGREL[7:0]> に 0x5A と 0xA5 を順番に設定後、PMDxOVVCR<OVVEN> に "0" を設定します。(3 命令連続して行う。)

OVV 保護回路を禁止するには <EMGREL[7:0]> に所定のキーコード 0x5A、0xA5 を設定する事ではじめて可能になり、誤って OVV 保護回路を禁止する事を防止します。

12.3.6.6 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 <OVVCNT[3:0]> = 1 ~ 15 (0 設定時は 1 になる) <OVVCNT[3:0]> × 16/fsys (分解能 200[nsec] @80MHz)  注) <OVVCNT[3:0]> はポート入力 <OVVISEL>=1 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	ADC 監視機能 1 入力許可 0: 入力禁止 1: 入力許可 ADC 監視機能 1 からの監視信号を許可/禁止にするか選択します。許可に設定し、かつ入力にADC監視信号(<OVVISEL>="1")を選択時、ADC の監視機能 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) ADC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	ADC 監視機能 0 入力許可 0: 入力禁止 1: 入力許可 ADC 監視機能 0 からの監視信号を許可/禁止にするか選択します。許可に設定し、かつ入力にADC監視信号(<OVVISEL>="1")を選択時、ADC の監視機能 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) ADC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 (オン =High オフ =Low(ハイアクティブ (POLL/H=1) 時)) OVV 発生時に、上相および下相の出力をオン / オフさせます。  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) 1: ADC 監視信号を選択した場合、<OVVCNT[3:0]> は無効となります (直接入力)。
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 電圧異常検出信号の入力 (High → Low) による OVV 保護状態へ移行した場合、あらかじめ 1 を設定する事により、電圧異常検出信号の入力が High になった後、PWM カウンタと <MDPRD[15:0]> が一致したタイミングで自動的に復帰します。  注) PWM0.5 周期割り込み設定 (<INTPRD[1:0]>="00") 時は 1 または <MDPRD[15:0]> と一致時となります。
0	OVVEN	R/W	OVV 保護回路の許可 / 禁止 0: 禁止 1: 許可 1 を設定する事により OVV 保護回路が動作状態となります。初期状態では禁止となっています。許可後に禁止する場合、禁止コードレジスタ <EMGREL[7:0]> に 0x5A → 0xA5 を順にライトし、<OVVEN> を "0" に設定します。(3 命令を連続して行います。)

12.3.6.7 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	-	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 (<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 リードする事により、OVV 保護の状態を知る事が出来ます。

## 12.3.7 デッドタイム回路

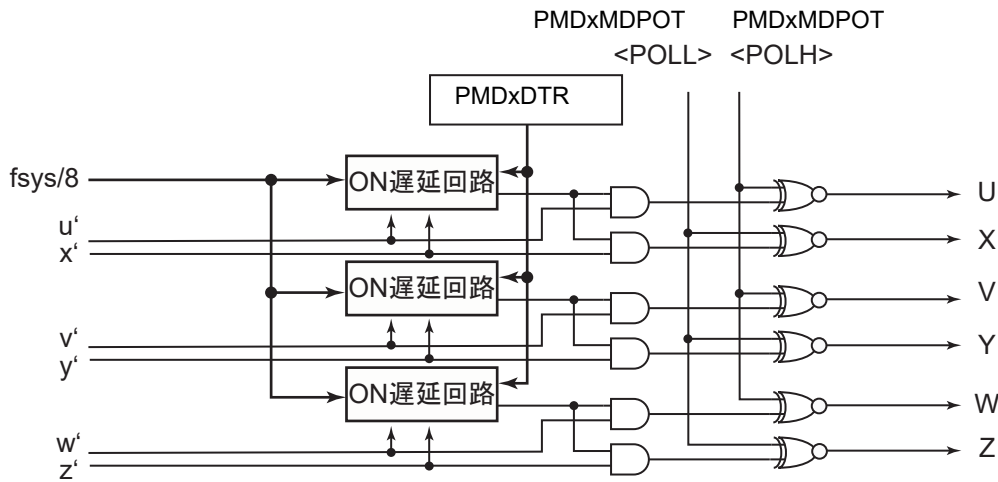


図 12-7 デッドタイム回路概略図

デッドタイム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。

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、デッドタイムカウンタによりオン時間を遅延させます。遅延時間は、8bit 値により 100ns @ 80MHz 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・下相をそれぞれに対してハイアクティブ / ローアクティブを PMDxMDPOT<POLH><POLL> により設定します。

12.3.7.1 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。 デッドタイムは以下の計算式で表されます。 $100\text{nsec} \times \text{<DTR[7:0]>}$ (最大 25.5 $\mu\text{sec}$ 、 $f_{\text{sys}}=80\text{MHz}$ )

注) <DTR[7:0]> は PMD $\times$ MDEN<PWMEN>=1 の状態では変更しないでください。

## 12.3.8 同期トリガ生成回路

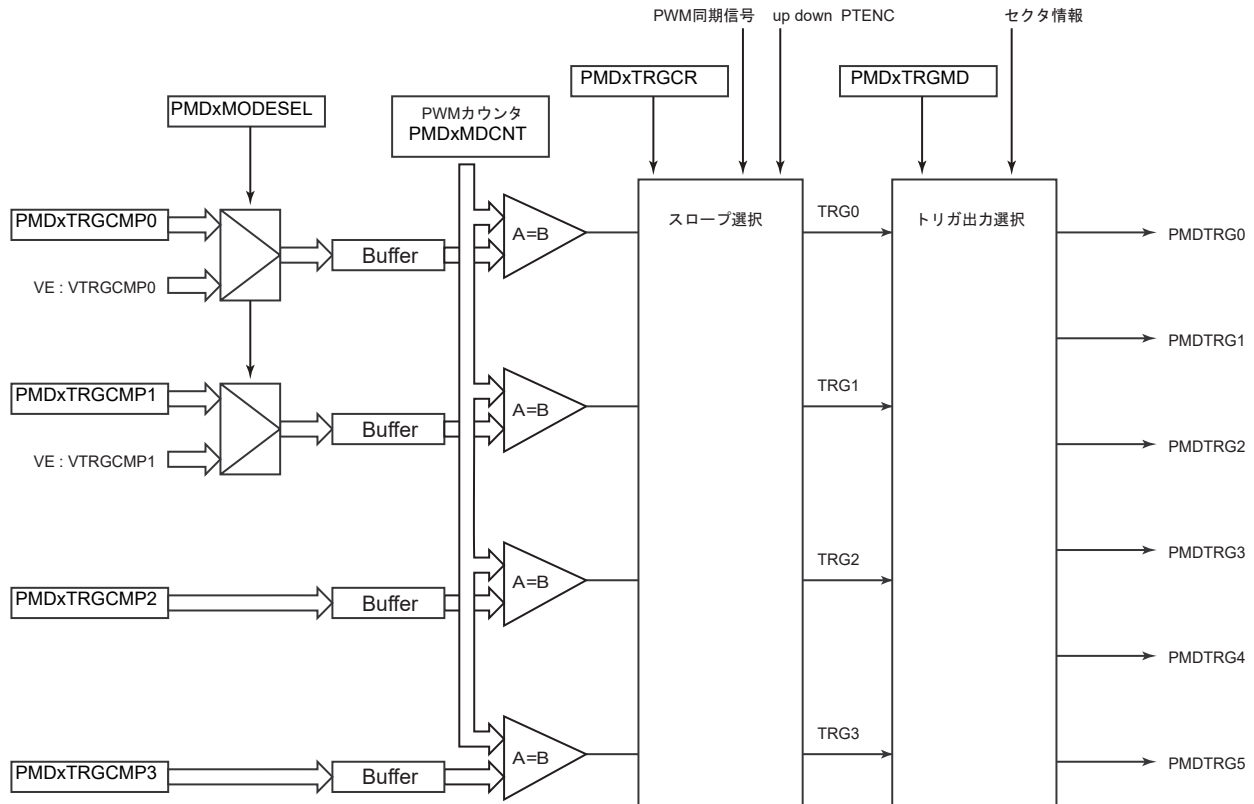


図 12-8 同期トリガ生成回路概略図

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為のトリガ信号を生成します。動作は  $\langle \text{MDCNT}[15:0] \rangle$  と  $\langle \text{TRGCMP}_n[15:0] \rangle$  とが一致する時に ADC トリガ信号 PMDTRG を発生します。発生タイミングはアップカウント動作時の一致、ダウンカウント動作時の一致、アップ / ダウンカウント両動作での一致を選択できます。エッジモード選択時はアップ時となります。PWM 出力禁止時 ( $\text{PMDxMDEN} \langle \text{PWMEN} \rangle = 0$ ) トリガは出力されません。

トリガ選択出力モードに設定した場合、TRGCMP0 との一致によるトリガ信号が、トリガ出力選択情報または  $\text{PMDxTRGSEL} \langle \text{TRGSEL} \rangle$  の設定によりトリガ出力ポートを切り替えます。

12.3.8.1 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 PMD カウンタ <MDCNT[15:0]> と一致すると PMDTRG を出力します。 リード時はダブルバッファ前段の値 (バスから設定したデータ) をリードします。  コンペアレジスタの設定範囲: 1 ~ [ <MDPRD[15:0]> 設定値 -1 ] <TRGCMP0> = 0 及び <TRGCMP0> [ <MDPRD[15:0]> 設定値 ] は設定禁止

- 注 1) バスから更新したトリガコンペアレジスタ 0 及びトリガコンペアレジスタ 1 をダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL> 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障できません。
- 注 3) <TRGCMP0> に 0x0001 を設定した場合、<PMMMD>=0(ノコギリ波) 時において PWM 開始 (<PWMEN>=1) 後、最初の 1 周期のみトリガ出力されません。

## 12.3.8.2 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 PMD カウンタ <MDCNT[15:0]> と一致すると PMDTRG を出力します。 リード時はダブルバッファ前段の値 (バスから設定したデータ) をリードします。  コンペアレジスタの設定範囲: 1 ~ [ <MDPRD[15:0]> 設定値 -1 ] <TRGCMP1> = 0 及び <TRGCMP1> [ <MDPRD[15:0]> 設定値 ] は設定禁止

- 注 1) バスから更新したトリガコンペアレジスタ 0 及びトリガコンペアレジスタ 1 をダブルバッファ後段にロードさせる場合は、PMDxMODESEL<MDSEL> 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障できません。
- 注 3) <TRGCMP1> に 0x0001 を設定した場合、<PWMMD>=0(ノコギリ波) 時において PWM 開始 (<PWMMEN>=1) 後、最初の 1 周期のみトリガ出力されません。

12.3.8.3 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 PMD カウンタ <MDCNT[15:0]> と一致すると PMDTRG を出力します。 リード時はダブルバッファ前段の値 (バスから設定したデータ) をリードします。  コンペアレジスタの設定範囲: 1 ~ [ <MDPRD[15:0]> 設定値 -1 ] <TRGCMP2> = 0 及び <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障できません。
- 注 2) <TRGCMP2> に 0x0001 を設定した場合、<PWMMD>=0( ノコギリ波 ) 時において PWM 開始 (<PWMEN>=1) 後、最初の 1 周期のみトリガ出力されません。



12.3.8.4 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 PMD カウンタ <MDCNT[15:0]> と一致すると PMDTRG を出力します。 リード時はダブルバッファ前段の値 (バスから設定したデータ) をリードします。  コンペアレジスタの設定範囲: 1 ~ [ <MDPRD[15:0]> 設定値 - 1 ] <TRGCMP3> = 0 及び <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障できません。

注 2) <TRGCMP3> に 0x0001 を設定した場合、<PWMMD>=0( ノコギリ波 ) 時において PWM 開始 (<PWMDEN>=1) 後、最初の 1 周期のみトリガ出力されません。

トリガコンペアレジスタ (TRGCMPx) の更新タイミング

トリガコンペア x レジスタ (TRGCMPx) はダブルバッファ構造となっており、内部バッファの更新タイミングはモード設定 <TRGxMD[2:0]> によってタイミングが異なります。<TRGxBE> に "1" を設定するとトリガモードに関係なく常に更新されます。

表 12-4 トリガ出力モード設定とトリガコンペアレジスタの内部バッファ更新タイミング

<TRGxMD[2:0]> 設定	TBUFx 更新タイミング
000: トリガ禁止	常に更新
001: ダウンカウント時の一致	PWM キャリアピーク (<MDPRD[15:0]> との一致) でバッファ更新
010: アップカウント時の一致	PWM キャリアボトム (1 との一致) でバッファ更新
011: アップ及びダウンカウント時の一致	PWM キャリアのピークとボトムで更新
100::PWM キャリアピーク	常に更新
101:PWM キャリアボトム	
110:PWM キャリアピーク及びボトム	
111: トリガ禁止	

12.3.8.5 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	PMDTRG3 のバッファ更新タイミング設定 0: 同期更新 1: 非同期更新 (書き込み後、直ちに反映される) バッファの非同期更新を許可します。
14-12	TRG3MD[2:0]	R/W	PMDTRG3 のモード設定 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 トリガ出力の一致モードを選択します。 PMDxMDCR<PMDMD> にてエッジモード選択時はダウンカウント/キャリアボトムを選択しても、アップカウント/キャリアピークでの出力となります。  "011" 選択時、PMDxTRGCMP3= "0x0001" の場合、<PWMMMD>="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	PMDTRG2 のバッファ更新タイミング設定 0: 同期更新 1: 非同期更新 (書き込み後、直ちに反映される) バッファの非同期更新を許可します。
10-8	TRG2MD[2:0]	R/W	PMDTRG2 のモード設定 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 トリガ出力の一致モードを選択します。 PMDxMDCR<PMDMD> にてエッジモード選択時はダウンカウント/キャリアボトムを選択しても、アップカウント/キャリアピークでの出力となります。  "011" 選択時、PMDxTRGCMP2="0x0001" の場合、<PWMMMD>="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	PMDTRG1 のバッファ更新タイミング設定 0: 同期更新 1: 非同期更新 (書き込み後、直ちに反映される) バッファの非同期更新を許可します。
6-4	TRG1MD[2:0]	R/W	PMDTRG1 のモード設定 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 トリガ出力の一致モードを選択します。 PMDxMDCR<PMDMD> にてエッジモード選択時はダウンカウント/キャリアボトムを選択しても、アップカウント/キャリアピークでの出力となります。  "011" 選択時、PMDxTRGCMP1= "0x0001" の場合、<PWMMMD>="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	PMDTRG0 のバッファ更新タイミング設定 0: 同期更新 1: 非同期更新 (書き込み後、直ちに反映される) バッファの非同期更新を許可します。

Bit	Bit Symbol	Type	機能
2-0	TRG0MD[2:0]	R/W	<p>PMDTRG0 のモード設定</p> <p>000 : トリガ出力禁止</p> <p>001 : ダウンカウント時の一致でトリガ出力</p> <p>010 : アップカウント時の一致でトリガ出力</p> <p>011 : アップ / ダウンカウント時にトリガ出力</p> <p>100 : PWM キャリアピークでトリガ出力</p> <p>101 : PWM キャリアボトムでトリガ出力</p> <p>110 : PWM キャリアピーク / ボトムでトリガ出力</p> <p>111 : トリガ出力禁止</p> <p>トリガ出力の一致モードを選択します。</p> <p>PMDxMDCR&lt;PMDMD&gt; にてエッジモード選択時はダウンカウント / キャリアボトムを選択しても、アップカウント / キャリアピークでの出力となります。</p> <p>"011" 選択時、PMDxTRGCMP0= "0x0001" の場合、&lt;PWWMD&gt;="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。</p>

12.3.8.6 PMDxTRGM (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 "0" 設定時、トリガ出力ポート <PMDTRG0> ~ <PMDTRG3> はそれぞれ <TRGCMP0> ~ <PMDCMP3> で一致したトリガ信号を出力します。VPM_TRG4,5 は Low 固定となります。 "1" 設定時、<TRGSEL> の設定またはベクトルエンジンからのセクタ情報により、PMDxTRGCMP0 に対応したトリガ出力を切り替えます。"1" 設定時のトリガ出力パターンについては表 12-5 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 12-5 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	TRGCMP0 によるトリガ出力
<TRGOUT>=0	PMDxTRGCMP0	×	PMDTRG0
	PMDxTRGCMP1		PMDTRG1
	PMDxTRGCMP2		PMDTRG2
	PMDxTRGCMP3		PMDTRG3
<TRGOUT>=1	PMDxTRGCMP0	0	PMDTRG0
		1	PMDTRG1
		2	PMDTRG2
		3	PMDTRG3
		4	PMDTRG4
	5	PMDTRG5	
	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	

12.3.8.7 PMDxTRGSEL (トリガ出力選択)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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
31-2	-	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 : PMDTRG0 より出力 001 : PMDTRG1 より出力 010 : PMDTRG2 より出力 011 : PMDTRG3 より出力 100 : PMDTRG4 より出力 101 : PMDTRG5 より出力 110 : トリガ出力しない 111 : トリガ出力しない トリガ選択出力モード時 (<TRGOUT>= "1") に有効となり、PMDxTRGCMP0 で設定した出力トリガを選択します。(表 12-5 参照)

# 第 13 章 ベクトルエンジン (VE)

## 13.1 概要

### 13.1.1 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。  
 → ソフトウェ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 (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 変換結果を固定少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  
 この場合、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

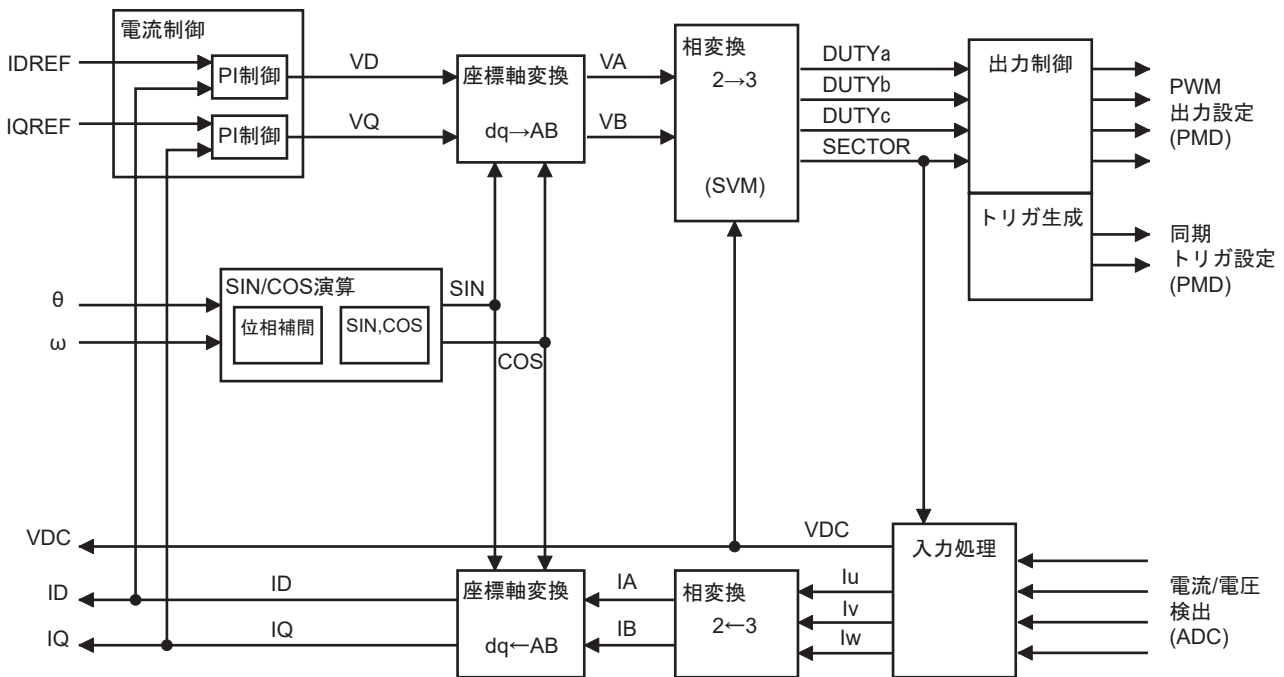


図 13-1 ベクトル制御の構成

### 13.1.2 主な機能

1. 2 相から 3 相への相変換には、空間ベクトル変換を採用。変換方式は、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可能、電流検出は 1 シャント方式, 3 シャント方式および 2 センサ方式に対応します。
3. 電流制御には、d 軸と q 軸に独立した PI 制御を採用。電流制御処理を使用せずに直接電圧指令することも可能です。
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。

注 1) 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。

注 2) 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。

## 13.2 構成

ベクトルエンジンの構成を図 13-2 に示します。

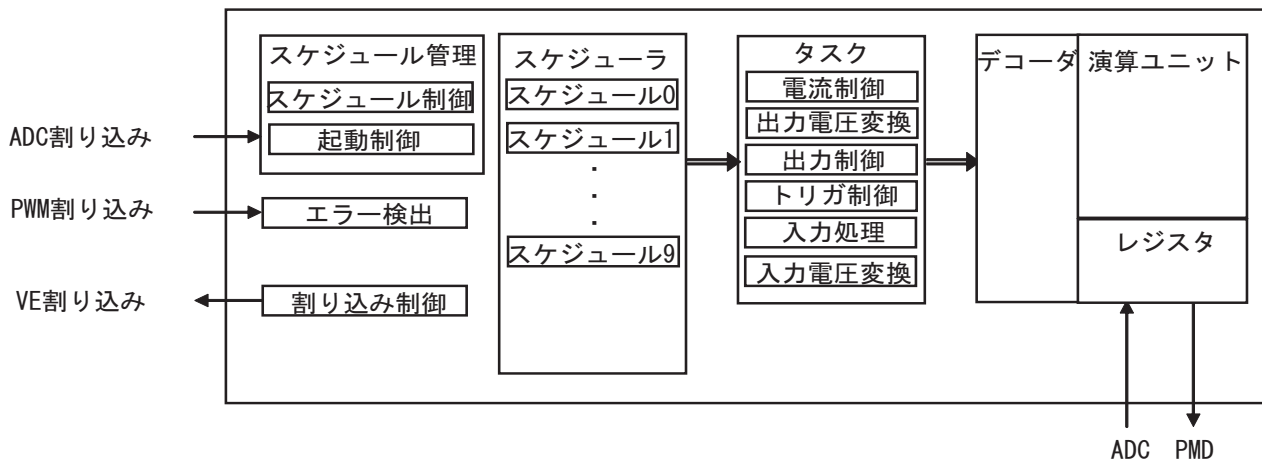


図 13-2 ベクトルエンジン構成



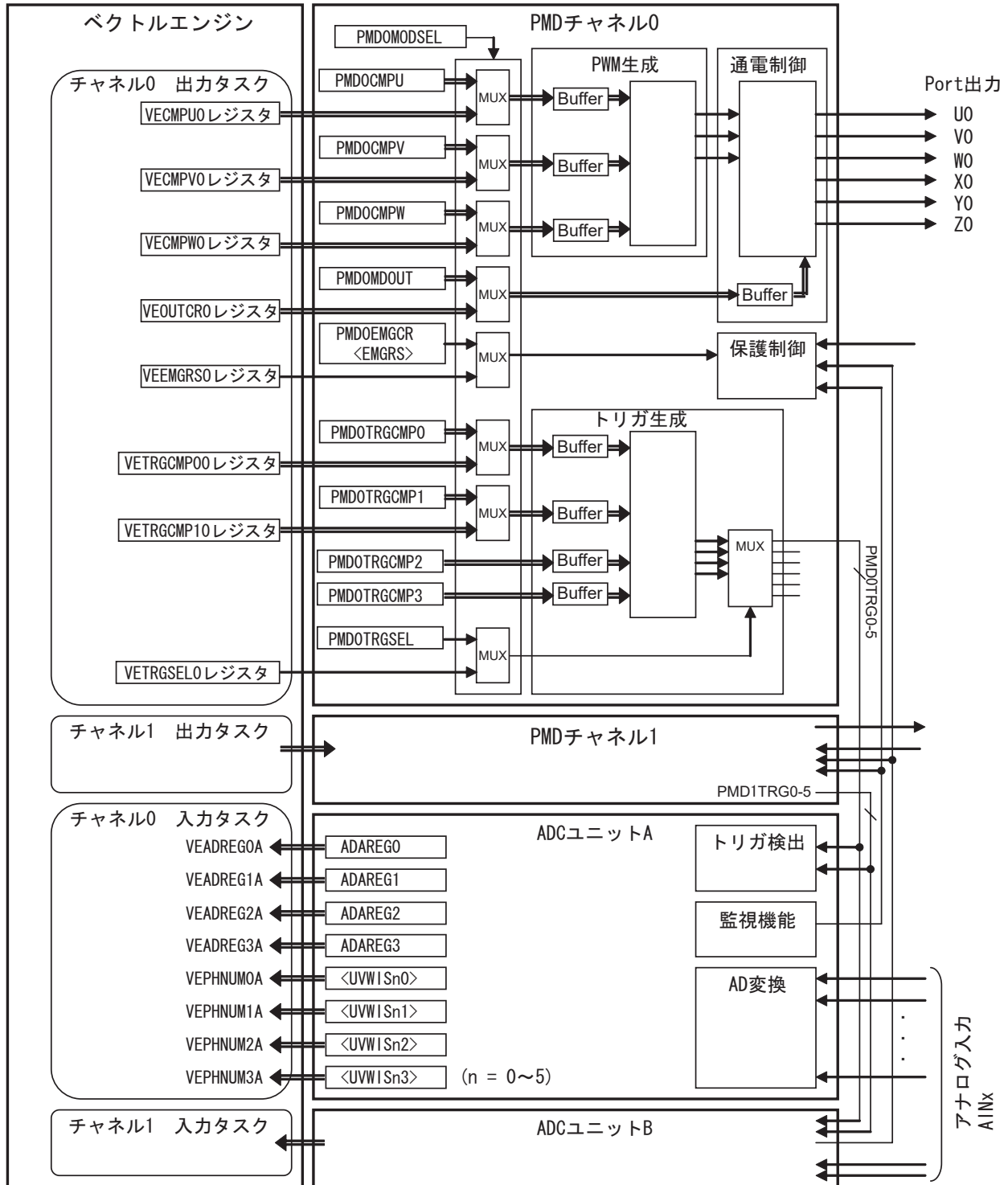
### 13.2.1 ベクトルエンジンとモータ制御回路および A/D 変換器の関連

ベクトルエンジンは 2 個のモータを制御可能で、2 つのモータ制御回路 (PMD) と 2 つの A/D 変換器 (ADC) と連携動作します。ベクトルエンジンのチャンネル 0 では PMD チャンネル 0 を制御し、チャンネル 1 では PMD チャンネル 1 を制御します。

ベクトルエンジンは図 13-3 に示すようにモータ制御回路 (PMD) 及び 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) チャンネル 0 レジスタの PMD0CMPU, PMD0CMPV, PMD0CMPW, PMD0MDOUT, PMD0TRGCMP0, PMD0TRGCMP1, PMD0TRGSEL は PMD0MODESEL で VE モードに設定すると、それぞれベクトルエンジンレジスタの VECMPU0, VECMPV0, VECMPW0, VEOUTCR0, VETRGCMP00, VETRGCMP10, VETRGSSEL0 に切り替わります。同様に、モータ制御回路 (PMD) チャンネル 1 レジスタはベクトルエンジンレジスタの VECMPU1, VECMPV1, VECMPW1, VEOUTCR1, VETRGCMP01, VETRGCMP11, VETRGSSEL1 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。

A/D 変換器 (ADC) ユニット A レジスタの ADAREG0, ADAREG1, ADAREG2, ADAREG3, ADAPSET<sub>n</sub><UVWISn0[1:0]>, <UVWISn1[[1:0]>, <UVWISn2[1:0]>, <UVWISn3[1:0]> はそれぞれベクトルエンジンレジスタの VEADREG0A, VEADREG1A, VEADREG2A, VEADREG3A, VEPHNUM0A, VEPHNUM1A, VEPHNUM2A, VEPHNUM3A (これらは CPU からアクセスできない専用レジスタ) に読み込まれます。同様に A/D 変換器 (ADC) ユニット B レジスタはベクトルエンジンレジスタの VEADREG0B, VEADREG1B, VEADREG2B, VEADREG3B, VEPHNUM0B, VEPHNUM1B, VEPHNUM2B, VEPHNUM3B (これらは CPU からアクセスできない専用レジスタ) に読み込まれます。CPU から A/D 変換器のレジスタへの読み書きの制限はありません。



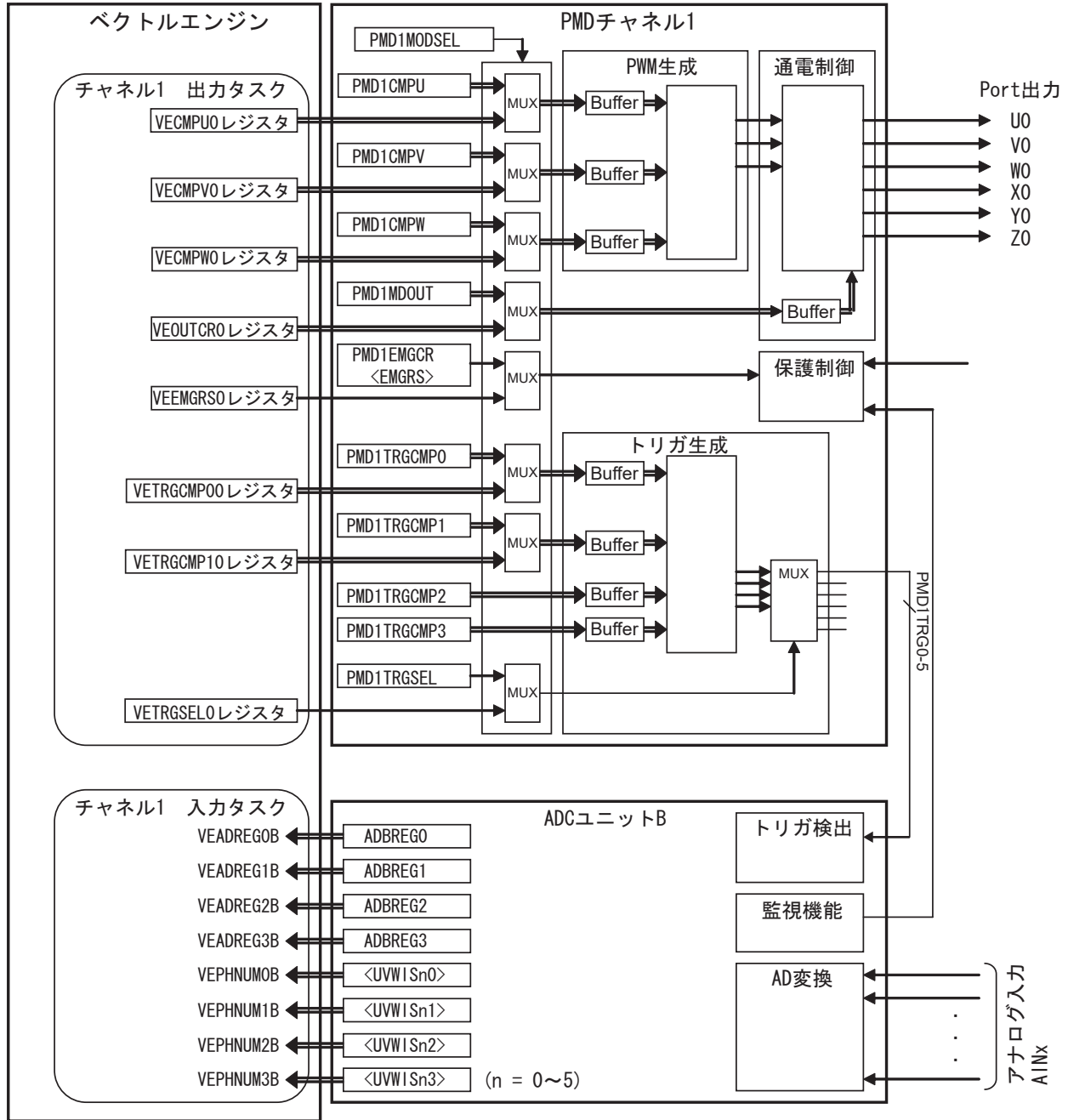


図 13-3 ベクトルエンジンとモータ制御回路及び A/D 変換器の関連図

## 13.3 レジスタ説明

ベクトルエンジンは特殊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
- 共通レジスタ
  - チャンネルで共通に使用するレジスタ
- 専用レジスタ
  - チャンネル毎の演算データ及び演算制御レジスタ

### 13.3.1 レジスタ一覧

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	VEEN	R/W	0x4005_0000
CPU 起動トリガ選択	VECPURUNTRG	W	0x4005_0004
タスク指定	VETASKAPP	R/W	0x4005_0008
動作スケジュール選択	VEACTSCH	R/W	0x4005_000C
動作スケジュール繰り返し回数指定	VEREPTIME	R/W	0x4005_0010
起動トリガモード設定	VETRGMODE	R/W	0x4005_0014
エラー割り込み許可 / 禁止設定	VEERRINTEN	R/W	0x4005_0018
ベクトルエンジン強制終了	VECOMPEND	W	0x4005_001C
エラー検出	VEERRDET	R	0x4005_0020
スケジュール実行中フラグ / 実行中タスク	VESCHTASKRUN	R	0x4005_0024
Reserved	-	R	0x4005_0028
テンポラリ 0	VETMPREG0	R/W	0x4005_002C
テンポラリ 1	VETMPREG1	R/W	0x4005_0030
テンポラリ 2	VETMPREG2	R/W	0x4005_0034
テンポラリ 3	VETMPREG3	R/W	0x4005_0038
テンポラリ 4	VETMPREG4	R/W	0x4005_003C
テンポラリ 5	VETMPREG5	R/W	0x4005_0040
Reserved	-	R	0x4005_01BC

共通レジスタ

レジスタ名			Address
Reserved	-	R/W	0x4005_0174
ADC 変換時間設定 (PWM クロック基準)	VETADC	R/W	0x4005_0178

## チャンネル0専用レジスタ

レジスタ名		Address
異常 / 判定結果保持	VEMCTLF0	R/W 0x4005_0044
タスク制御モード	VEMODE0	R/W 0x4005_0048
フロー制御	VEFMODE0	R/W 0x4005_004C
PWM 周期レート (PWM 周期 [s]× 最大速度 (注 1)×2 <sup>16</sup> ) 設定	VETPWM0	R/W 0x4005_0050
回転速度 (速度 [Hz]÷ 最大速度 (注 1)×2 <sup>15</sup> ) 設定	VEOMEGA0	R/W 0x4005_0054
モータ位相 (モータ位相 [deg]/360×2 <sup>16</sup> ) 設定	VETHETA0	R/W 0x4005_0058
d 軸基準電流値 (電流 [A]÷ 最大電流 (注 2)×2 <sup>15</sup> ) 設定	VEIDREF0	R/W 0x4005_005C
q 軸基準電流値 (電流 [A]÷ 最大電流 (注 2)×2 <sup>15</sup> ) 設定	VEIQREF0	R/W 0x4005_0060
d 軸電圧 (電圧 [V]÷ 最大電圧 (注 3)×2 <sup>31</sup> ) 設定	VEVD0	R/W 0x4005_0064
q 軸電圧 (電圧 [V]÷ 最大電圧 (注 3)×2 <sup>31</sup> ) 設定	VEVQ0	R/W 0x4005_0068
d 軸電流制御 PI 積分項係数設定	VECIDK10	R/W 0x4005_006C
d 軸電流制御 PI 比例項係数設定	VECIDKP0	R/W 0x4005_0070
q 軸電流制御 PI 積分項係数設定	VECIQK10	R/W 0x4005_0074
q 軸電流制御 PI 比例項係数設定	VECIQKP0	R/W 0x4005_0078
d 軸電圧積分成項 保持 (VDI の上位 32bit)	VEVDIH0	R/W 0x4005_007C
d 軸電圧積分成項 保持 (VDI の下位 32bit)	VEVDILH0	R/W 0x4005_0080
q 軸電圧積分成項 保持 (VQI の上位 32bit)	VEVQIH0	R/W 0x4005_0084
q 軸電圧積分成項 保持 (VQI の下位 32bit)	VEVQILH0	R/W 0x4005_0088
2 相変調でシフト PWM 許可時の切り替え速度	VEFPWMCHG0	R/W 0x4005_008C
PWM 周期設定 (PMD の PWM 周期設定と同じ値を設定)	VEMDPRD0	R/W 0x4005_0090
最小パルス幅設定	VEMINPLS0	R/W 0x4005_0094
同期トリガ補正量設定	VETRGCR0	R/W 0x4005_0098
Reserved	-	R/W 0x4005_009C
THETA での余弦値 出力変換用 (Q15 データ)	VECOS0	R/W 0x4005_00A0
THETA での正弦値 出力変換用 (Q15 データ)	VESIN0	R/W 0x4005_00A4
前回の余弦値 入力処理用 (Q15 データ)	VECOSM0	R/W 0x4005_00A8
前回の正弦値 入力処理用 (Q15 データ)	VESINM0	R/W 0x4005_00AC
セクタ情報	VESECTOR0	R/W 0x4005_00B0
前回セクタ情報	VESECTORM0	R/W 0x4005_00B4
a 相ゼロ電流 (注 4)	VEIA00	R/W 0x4005_00B8
b 相ゼロ電流 (注 4)	VEIB00	R/W 0x4005_00BC
c 相ゼロ電流 (注 4)	VEIC00	R/W 0x4005_00C0
a 相電流 ADC 変換結果 (注 4)	VEIAADC0	R/W 0x4005_00C4
b 相電流 ADC 変換結果 (注 4)	VEIBADC0	R/W 0x4005_00C8
c 相電流 ADC 変換結果 (注 4)	VEICADC0	R/W 0x4005_00CC
電源電圧 (電圧 [V]÷ 最大電圧 (注 3)×2 <sup>15</sup> )	VEVDC0	R/W 0x4005_00D0
d 軸電流 (電流 [A]÷ 最大電流 (注 2)×2 <sup>31</sup> )	VEID0	R/W 0x4005_00D4
q 軸電流 (電流 [A]÷ 最大電流 (注 2)×2 <sup>31</sup> )	VEIQ0	R/W 0x4005_00D8
PMD 制御 : U 相 PWM パルス幅設定	VECMPU0	R/W 0x4005_017C
PMD 制御 : V 相 PWM パルス幅設定	VECMPV0	R/W 0x4005_0180
PMD 制御 : W 相 PWM パルス幅設定	VECMPW0	R/W 0x4005_0184
PMD 制御 : 6 相出力制御	VEOUTCR0	R/W 0x4005_0188
PMD 制御 : トリガタイミング設定 (TRGCMP0)	VETRGCMP00	R/W 0x4005_018C
PMD 制御 : トリガタイミング設定 (TRGCMP1)	VETRGCMP10	R/W 0x4005_0190
PMD 制御 : 同期トリガ指定	VETRGSEL0	R/W 0x4005_0194
PMD 制御 : EMG 復帰設定	VEEMGRS0	W 0x4005_0198

チャンネル 1 専用レジスタ

レジスタ名		Address
異常 / 判定結果保持	VEMCTLF1	R/W 0x4005_00DC
タスク制御モード	VEMODE1	R/W 0x4005_00E0
フロー制御	VEFMODE1	R/w 0x4005_00E4
PWM 周期レート (PWM 周期 [s]× 最大速度 (注 1)×2 <sup>16</sup> ) 設定	VETPWM1	R/W 0x4005_00E8
回転速度 (速度 [Hz]÷ 最大速度 (注 1)×2 <sup>15</sup> ) 設定	VEOMEGA1	R/w 0x4005_00EC
モータ位相 (モータ位相 [deg]/360×2 <sup>16</sup> ) 設定	VETHETA1	R/W 0x4005_00F0
d 軸基準電流値 (電流 [A]÷ 最大電流 (注 2)×2 <sup>15</sup> ) 設定	VEIDREF1	R/w 0x4005_00F4
q 軸基準電流値 (電流 [A]÷ 最大電流 (注 2)×2 <sup>15</sup> ) 設定	VEIQREF1	R/W 0x4005_00F8
d 軸電圧 (電圧 [V]÷ 最大電圧 (注 3)×2 <sup>31</sup> ) 設定	VEVD1	R/w 0x4005_00FC
q 軸電圧 (電圧 [V]÷ 最大電圧 (注 3)×2 <sup>31</sup> ) 設定	VEVQ1	R/W 0x4005_0100
d 軸電流制御 PI 積分項係数設定	VECIDK11	R/w 0x4005_0104
d 軸電流制御 PI 比例項係数設定	VECIDKP1	R/W 0x4005_0108
q 軸電流制御 PI 積分項係数設定	VECIQK11	R/w 0x4005_010C
q 軸電流制御 PI 比例項係数設定	VECIQKP1	R/W 0x4005_0110
d 軸電圧積分成項 保持 (VDI の上位 32bit)	VEVDIH1	R/w 0x4005_0114
d 軸電圧積分成項 保持 (VDI の下位 32bit)	VEVDILH1	R/W 0x4005_0118
q 軸電圧積分成項 保持 (VQI の上位 32bit)	VEVQIH1	R/w 0x4005_011C
q 軸電圧積分成項 保持 (VQI の下位 32bit)	VEVQILH1	R/W 0x4005_0120
2 相変調でシフト PWM 許可時の切り替え速度	VEFPWMCHG1	R/w 0x4005_0124
PWM 周期設定 (PMD の PWM 周期設定と同じ値を設定)	VEMDPRD1	R/W 0x4005_0128
最小パルス幅設定	VEMINPLS1	R/W 0x4005_012C
同期トリガ補正量設定	VETRGCRC1	R/w 0x4005_0130
Reserved	-	R/W 0x4005_0134
THETA での余弦値 出力変換用 (Q15 データ)	VECOS1	R/w 0x4005_0138
THETA での正弦値 出力変換用 (Q15 データ)	VESIN1	R/W 0x4005_013C
前回の余弦値 入力処理用 (Q15 データ)	VECOSM1	R/w 0x4005_0140
前回の正弦値 入力処理用 (Q15 データ)	VESINM1	R/W 0x4005_0144
セクタ情報	VESECTOR1	R/w 0x4005_0148
前回セクタ情報	VESECTORM1	R/W 0x4005_014C
a 相ゼロ電流 (注 4)	VEIAO1	R/w 0x4005_0150
b 相ゼロ電流 (注 4)	VEIBO1	R/W 0x4005_0154
c 相ゼロ電流 (注 4)	VEICO1	R/w 0x4005_0158
a 相電流 ADC 変換結果 (注 4)	VEIAADC1	R/W 0x4005_015C
b 相電流 ADC 変換結果 (注 4)	VEIBADC1	R/w 0x4005_0160
c 相電流 ADC 変換結果 (注 4)	VEICADC1	R/W 0x4005_0164
電源電圧 (電圧 [V]÷ 最大電圧 (注 3)×2 <sup>15</sup> )	VEVDC1	R/w 0x4005_0168
d 軸電流 (電流 [A]÷ 最大電流 (注 2)×2 <sup>31</sup> )	VEID1	R/W 0x4005_016C
q 軸電流 (電流 [A]÷ 最大電流 (注 2)×2 <sup>31</sup> )	VEIQ1	R/w 0x4005_0170
PMD 制御 : U 相 PWM パルス幅設定	VECMU1	R/W 0x4005_019C
PMD 制御 : V 相 PWM パルス幅設定	VECMV1	R/w 0x4005_01A0
PMD 制御 : W 相 PWM パルス幅設定	VECMW1	R/W 0x4005_01A4
PMD 制御 : 6 相出力制御	VEOUTCR1	R/w 0x4005_01A8
PMD 制御 : トリガタイミング設定 (TRGCMP0)	VETRGCMP01	R/W 0x4005_01AC
PMD 制御 : トリガタイミング設定 (TRGCMP1)	VETRGCMP11	R/w 0x4005_01B0
PMD 制御 : 同期トリガ指定	VETRGSSEL1	R/W 0x4005_01B4
PMD 制御 : EMG 復帰設定	VEEMGRS1	W 0x4005_01B8

- 注 1) 最大速度：制御あるいは動作可能な最大回転数 [Hz]
- 注 2) 最大電流：相電流の A/D 変換結果をゼロレベル補正した後に 0x7FF0 になる場合の電流値 [A]
- 注 3) 最大電圧：電源電圧 (VDC) の A/D 変換結果が 0xFFF0 になる場合の電圧値 [V]
- 注 4) A/D 変換結果は 16bit レジスタの上位 12bit に保存されます

### 13.3.2 VE 制御レジスタ

#### 13.3.2.1 VEEN(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



## 13.3.2.2 VECPURUNTRG(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	-	-	-	-	-	-	VCPURTB	VCPURTA
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	VCPURTB	W	チャンネル 1 をソフトウェアで起動 0: - 1: 動作開始
0	VCPURTA	W	チャンネル 0 をソフトウェアで起動 0: - 1: 動作開始  VETASKAPP<VTASKA> と VEACTSCH<VACTA> で設定されたタスクの動作を開始します。

- 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) 動作開始されるタスクは VETASKAPP と VEACTSCH の設定によります。
- 注 3) スケジュール実行中に、同じチャンネルのタスクを再起動する場合、VECOMPEND レジスタで対応するチャンネルを強制終了してから、再度、VECPURUNTRG レジスタにて動作を開始させてください。

13.3.2.3 VETASKAPP( 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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VTASKB				VTASKA			
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。
7-4	VTASKB[3:0]	R/W	チャンネル 1 の開始タスクを指定  0x0 : 出力制御 0x1 : トリガ制御 0x2 : 入力処理 0x3 : 入力相変換 0x4 : 入力座標軸変換 0x5 : 電流制御 0x6 : SIN/COS 演算 0x7 : 出力座標軸変換 0x8 : 出力相変換 0x9 ~ 0xF : Reserved  チャンネル 1 をソフトウェアで起動する時の開始タスクを指定します。
3-0	VTASKA[3:0]	R/W	チャンネル 0 の開始タスクを指定  0x0 : 出力制御 0x1 : トリガ制御 0x2 : 入力処理 0x3 : 入力相変換 0x4 : 入力座標軸変換 0x5 : 電流制御 0x6 : SIN/COS 演算 0x7 : 出力座標軸変換 0x8 : 出力相変換 0x9 ~ 0xF : Reserved  チャンネル 0 をソフトウェ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。

## 13.3.2.4 VECTSCH( 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	VACTB				VACTA			
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。
7-4	VACTB[3:0]	R/W	チャンネル 1 の動作スケジュールを指定 0x0 : タスク実行 0x1 : スケジュール 1 0x4 : スケジュール 4 0x9 : スケジュール 9 その他 : Reserved
3-0	VACTA[3:0]	R/W	チャンネル 0 の動作スケジュールを指定 0x0 : タスク実行 0x1 : スケジュール 1 0x4 : スケジュール 4 0x9 : スケジュール 9 その他 : Reserved

13.3.2.5 VEREPTIME(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	VREPB				VREPA			
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。
7-4	VREPB[3:0]	R/W	チャンネル 1 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実行する
3-0	VREPA[3:0]	R/W	チャンネル 0 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実行する

注) "0" 設定時はスケジュール動作しないでください。

## 13.3.2.6 VETRGMODE( 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	-	-	-	-	VTRGB		VTRGA	
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。
3-2	VTRGB[1:0]	R/W	チャンネル 1 での AD 変換終了割り込みによる入力処理起動条件選択 00 : INTB0( ユニット A),INTB1( ユニット B) とも無視 01 : INTB0( ユニット A) 割り込みで起動 10 : INTB1( ユニット B) 割り込みで起動 11 : INTB0( ユニット A),INTB1( ユニット B) 両方が揃うと起動
1-0	VTRGA[1:0]	R/W	チャンネル 0 での AD 変換終了割り込みによる入力処理起動条件選択 00 : INTA0( ユニット A),INTA1( ユニット B) とも無視 01 : INTA0( ユニット A) 割り込みで起動 10 : INTA1( ユニット B) 割り込みで起動 11 : INTA0( ユニット A),INTA1( ユニット B) 両方が揃うと起動

13.3.2.7 VEERRINTEN(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	-	-	-	-	-	-	VERRENB	VERRENA
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	VERRENB	R/W	チャンネル 1 エラー検出時の割り込み制御 0: 禁止 1: 許可
0	VERRENA	R/W	チャンネル 0 のエラー検出時の割り込み制御 0: 禁止 1: 許可

## 13.3.2.8 VECOMPEND(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	-	-	-	-	-	-	VCENDB	VCENDA
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	VCENDB	W	チャンネル1 実行中のスケジュール強制終了 0: - 1: 停止
0	VCENDA	W	チャンネル0 実行中のスケジュール強制終了 0: - 1: 停止

注) "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。

13.3.2.9 VEERRDET( 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	-	-	-	-	-	-	VERRDB	VERRDA
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	VERRDB	R	チャンネル 1 エラーフラグ 0 : エラー未検出 1 : エラー検出
0	VERRDA	R	チャンネル 0 エラーフラグ 0 : エラー未検出 1 : エラー検出

注 1) 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。

注 2) エラーフラグはレジスタをリードするとクリアされます。



13.3.2.10 VESCHTASKRUN( 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	-	-	-	-	-	-	VRTASKB	
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	VRTASKB		VRSCHB	VRTASKA				VRSCHA
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。
9-6	VRTASKB[3:0]	R	チャンネル 1 の実行中タスク番号 0x0 : 出力制御 0x1 : トリガ生成 0x2 : 入力処理 0x3 : 入力相変換 0x4 : 入力座標軸変換 0x5 : 電流制御 0x6 : SIN/COS 演算 0x7 : 出力座標軸変換 0x8 : 出力相変換 0x9 ~ 0xF : Reserved
5	VRSCHB	R	チャンネル 1 スケジュール動作状態 0 : 停止 1 : 実行中
4-1	VRTASKA[3:0]	R	チャンネル 0 の実行中タスク番号 0x0 : 出力制御 0x1 : トリガ生成 0x2 : 入力処理 0x3 : 入力相変換 0x4 : 入力座標軸変換 0x5 : 電流制御 0x6 : SIN/COS 演算 0x7 : 出力座標軸変換 0x8 : 出力相変換 0x9 ~ 0xF : Reserved
0	VRSCHA	R	チャンネル 0 スケジュール動作状態 0 : 停止 1 : 実行中

13.3.2.11 VETMPREG0( テンポラリレジスタ 0 )

	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

13.3.2.12 VETMPREG1( テンポラリレジスタ 1 )

	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

## 13.3.2.13 VETMPREG2( テンポラリレジスタ 2 )

	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

## 13.3.2.14 VETMPREG3( テンポラリレジスタ 3 )

	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

13.3.2.15 VETMPREG4( テンポラリレジスタ 4 )

	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

13.3.2.16 VETMPREG5( テンポラリレジスタ 5 )

	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

## 13.3.3 共通レジスタ

## 13.3.3.1 VETADC(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 0x0000-0xFFFF : (ADC 変換時間 [s]=PWM カウンタクロック周期 [s] を設定) 注)1 シャント電流検出方式で PWM シフト許可時に有効

13.3.4 専用レジスタ (x=0 ~ 1)

13.3.4.1 VEMODEx( 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	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	OCRMD		ZIEN	PVIEN
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。
7-4	-	R/W	"0" をライトしてください。
3-2	OCRMD[1:0]	R/W	出力制御動作 00 : 出力オフ 01 : 出力許可 10 : Reserved 11 : EMG 復帰
1	ZIEN	R/W	ゼロ電流検出 0 : 禁止 1 : 許可
0	PVIEN	R/W	位相補間 0 : 禁止 1 : 許可

13.3.4.2 VEFMODEx( 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	-	-	-	-	-	-	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	ADCSEL		-	PMDSEL	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。						
15-10	-	R/W	"0" を書いてください。						
9	MREGDIS	R/W	SIN/COS/SECTOR 前回値保持選択 0: 有効 1: 無効 無効時は、VESINMx=VESINx、VECOSMx=VECOSx、VESECTORMx=VESECTORx						
8	CRCEN	R/W	トリガ補正許可 0: 禁止 1: 許可						
7-6	ADCSEL[1:0]	R/W	ADC ユニット選択 00: ユニット A 01: ユニット B 10: ユニット A,B 11: ユニット A,B  ベクトルエンジンのチャンネルにより以下に設定してください <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>VE</th> <th>使用 ADC</th> </tr> </thead> <tbody> <tr> <td>チャンネル 0</td> <td>ユニット A または ユニット A,B</td> </tr> <tr> <td>チャンネル 1</td> <td>ユニット B または ユニット A,B</td> </tr> </tbody> </table>	VE	使用 ADC	チャンネル 0	ユニット A または ユニット A,B	チャンネル 1	ユニット B または ユニット A,B
VE	使用 ADC								
チャンネル 0	ユニット A または ユニット A,B								
チャンネル 1	ユニット B または ユニット A,B								
5	-	R/W	"0" を書いてください						
4	PMDSEL	R/W	PMD チャンネル選択 0: チャンネル 0 1: チャンネル 1  ベクトルエンジンのチャンネルにより以下に設定してください <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>VE</th> <th>使用 PMD</th> </tr> </thead> <tbody> <tr> <td>チャンネル 0</td> <td>チャンネル 0</td> </tr> <tr> <td>チャンネル 1</td> <td>チャンネル 1</td> </tr> </tbody> </table>	VE	使用 PMD	チャンネル 0	チャンネル 0	チャンネル 1	チャンネル 1
VE	使用 PMD								
チャンネル 0	チャンネル 0								
チャンネル 1	チャンネル 1								
3-2	IDMODE	R/W	電流検出モード 00: 3 シャント 01: 2 センサ 10: 1 シャント (PMD TRG アップカウンタ (注)) 11: 1 シャント (PMD TRG ダウンカウンタ (注))						
1	SPWMEN	R/W	PWM シフト許可 0: 禁止 1: 許可						
0	C2PEN	R/W	変調モード選択 0: 3 相変調 1: 2 相変調						

注 ) シャントモード使用時の PMDTRG の設定値

VEFMODE0 <IDMODE>	VEFMODE1 <IDMODE>	PMD0TRGCR <TRG0MD>	PMD0TRGCR <TRG1MD>	PMD1TRGCR <TRG0MD>	PMD1TRGCR <TRG1MD>
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)	-	-
-	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)



13.3.4.3 VETPWMx(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 $0x0000 \sim 0xFFFF : PWM \text{ 周期 [s]} \times \text{Max\_Hz} \times 2^{16}$ 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)

13.3.4.4 VEOMEGAx( 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 $0x0000 \sim 0xFFFF : \text{回転速度 [Hz]} \div \text{Max\_Hz} \times 2^{15}$ (Max_Hz : 最大回転数)

13.3.4.5 VETHETAx( モーター位相設定レジスタ )

	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 計算式 : 位相 [deg] ÷ 360 × 2 <sup>16</sup>

13.3.4.6 VECOSx/VESINx/VECOSMx/VESINMx

VECOSx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF

## VESINx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF

## VECOSMx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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	VECOSx レジスタの前回値保存 0x0000 ~ 0xFFFF

VESINMx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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	VESINx レジスタの前回値保存 0x0000 ~ 0xFFFF

13.3.4.7 VEIDREFx/VEIQREFx(d 軸 /q 軸基準電流値設定レジスタ)

VEIDREFx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF(d 軸電流指令 [A]=Max <sub>I</sub> ×2 <sup>15</sup> を設定する  Max <sub>I</sub> : (A/D 変換が 1LSB 変化する相電流の変化量 [A])×2 <sup>11</sup>

VEIQREFx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF(q 軸電流指令 [A]=Max <sub>I</sub> ×2 <sup>15</sup> を設定する )  Max <sub>I</sub> : (A/D 変換が 1LSB 変化する相電流の変化量 [A])×2 <sup>11</sup>

13.3.4.8 VEVDx/VEVQx(d 軸 /q 軸電圧設定レジスタ)

VEVDx

	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 0x0000-0000 ~ 0xFFFF-FFFF(d 軸電圧 $\pm$ Max_V $\times 2^{31}$ を設定する )  Max_V : (A/D 変換が 1LSB 変化する電源電圧の変化量 [V]) $\times 2^{12}$

VEVQx

	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 0x0000-0000 ~ 0xFFFF-FFFF(q 軸電圧 $\pm$ Max_V $\times 2^{31}$ を設定する )  Max_V : (A/D 変換が 1LSB 変化する電源電圧の変化量 [V]) $\times 2^{12}$

13.3.4.9 VECIDKix/VECIDKPx/VEVCIQKix/VEVCIQKPx(PI 制御係数レジスタ)

VECIDKix

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF

VECIDKPx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF

VEVCIQKIx

	31	30	29	28	27	26	25	24
bit symbol	-							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF

VECIQKPx

	31	30	29	28	27	26	25	24
bit symbol	-							
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 0x0000 ~ 0xFFFF



13.3.4.10 VEVDIHx/VEVDILHx/VEVQIHx/VEVQILHx(PI 制御積分項保持レジスタ)

VEVDIHx

	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

VEVDILHx

	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 32bit、ただし下位 16bit は 0x0000 固定
15-0	-	R	リードすると "0" が読めます。

注) VDI は 64bit 固定小数点データ (小数 63bit -1.0 ~ 1.0)

VEVQIHx

	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

VEVQILHx

	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 32bit、ただし下位 16bit は 0x0000 固定
15-0	-	R	リードすると "0" が読めます。

注) VQI は 64bit 固定小数点データ (小数 63bit -1.0 ~ 1.0)

## 13.3.4.11 VEMCTLFx( 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	-	-	-	-	-	-	-	-
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
31-8	-	R	リードすると "0" が読めます。
7-6	-	R/W	"0" をライトしてください。
5	PLSLFM	R/W	<PLSLF> の前回値
4	PLSLF	R/W	パルス微小フラグ 最小パルス幅差 < VEMINPLSx<MINPLS> の場合 "1"
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 VEVDCx<VDC> ≥ 1/128 の場合 "0" VEVDCx<VDC> < 1/128 の場合 "1"
1	LAVFM	R/W	<LAVF> の前回値
0	LAVF	R/W	低速度フラグ 0 : 高速 1 : 低速 VEOMEGAx<OMEGA> ≥ VEFPWMCHGx<FPWMCHG> の場合 "0" VEOMEGAx<OMEGA> < VEFPWMCHGx<FPWMCHG> の場合 "1"

13.3.4.12 VEFPWMCHGx(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 切り替え速度 [Hz] ÷ Max _ Hz × 2 <sup>15</sup> を設定  (Max _ Hz : 最大回転数 [Hz])

13.3.4.13 VEMDPRDx(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 機能の PMDxMDPRD レジスタ値を設定します。

## 13.3.4.14 VEMINPLSx( 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	最小パルス幅差 (3相PWM(VECMPUx,VECMPxV,VECMPWx)のDuty差の最小値) 設定値 : パルス幅差 [s] ÷ PWM カウンタクロック周期 [s]

13.3.4.15 VESECTORx/VESECTORMx( セクタ情報レジスタ )

VESECTORx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 ~ 0xF 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す。

VESECTORMx

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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	-	-	-	-	SECTORM			
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 ~ 0xF 入力処理で使用。

13.3.4.16 VEIAOx/VEIBOx/VEICOx( ゼロ電流レジスタ )

VEIAOx

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

VEIBOx

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

VEICOx

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

注 1) ゼロ電流検出許可時に ADC 変換結果が自動で保存されます。

注 2) ADC 変換結果取り込み時は Bit 15-4 に保存され、Bit 3-0 は "0"



13.3.4.17 VEIAADCx/VEIBADCx/VEICADCx( 電流 ADC 結果レジスタ )

VEIAADCx

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

VEIBADCx

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

VEICADCx

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

注) ADC 変換結果は上位 12bit に保存されます。下位 4bit は常に "0" です。

## 13.3.4.18 VEVDcX( 電源電圧レジスタ )

	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	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 : 0x0000-0xFFFF  実電圧値に変換するには、VDC 値 ×Max_V 値 ÷2 <sup>15</sup> (Max_V : (A/D 変換が 1LSB 変化する電源電圧の変化量 [V]) × 2 <sup>12</sup> )

13.3.4.19 VEIDx/VEIQx(d 軸 /q 軸電流レジスタ )

VEIDx

	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 : 0x0000_0000 ~ 0xFFFF_FFFF  実電流値に変換するには、In 値 × Max_I 値 ÷ 2 <sup>31</sup> (Max_I : (A/D 変換が 1LSB 変化する相電流の変化量 [A]) × 2 <sup>11</sup> )

VEIQx

	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 : 0x0000_0000 ~ 0xFFFF_FFFF  実電流値に変換するには、In 値 × Max_I 値 ÷ 2 <sup>31</sup> (Max_I : (A/D 変換が 1LSB 変化する相電流の変化量 [A]) × 2 <sup>11</sup> )

13.3.4.20 VECMPUx/ VECMPVx/ VECMPWx(PWM DUTY レジスタ)

VECMPUx

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

VECMPVx

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

VECMPWx

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

13.3.4.21 VEOUTCRx(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

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)

<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

<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

<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

13.3.4.22 VETRGCRcx( 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]



## 13.3.4.23 VETRGCMP0x/VETRGCMP1x(トリガタイミング設定レジスタ)

VETRGCMP0x

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 ~ (<MDPRD[15:0]> 値 -1): トリガタイミング <MDPRD[15:0]> 値 ~ 0xFFFF: 設定禁止

VETRGCMP1x

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit 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 ~ (<MDPRD[15:0]> 値 -1): トリガタイミング <MDPRD[15:0]> 値 ~ 0xFFFF: 設定禁止

注 1) 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、アップ及びダウンカウント時の一致

注 2) 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

13.3.4.24 VETRGSELx( 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	<VTRGCMP0[15:0]> 設定タイミングで出力する同期トリガ番号の指定 0 ~ 5 : 出力トリガ番号 6 ~ 7 : 使用禁止 注 ) 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

## 13.3.4.25 VEEMGRSx(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	R/W	EMG 復帰コマンド (PMD 設定) 0 : Nop 1 : EMG 復帰コマンド

## 13.4 動作説明

### 13.4.1 スケジュール管理

モータ制御は図 13-4 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と MODE 設定を切り替えることで遷移させます。

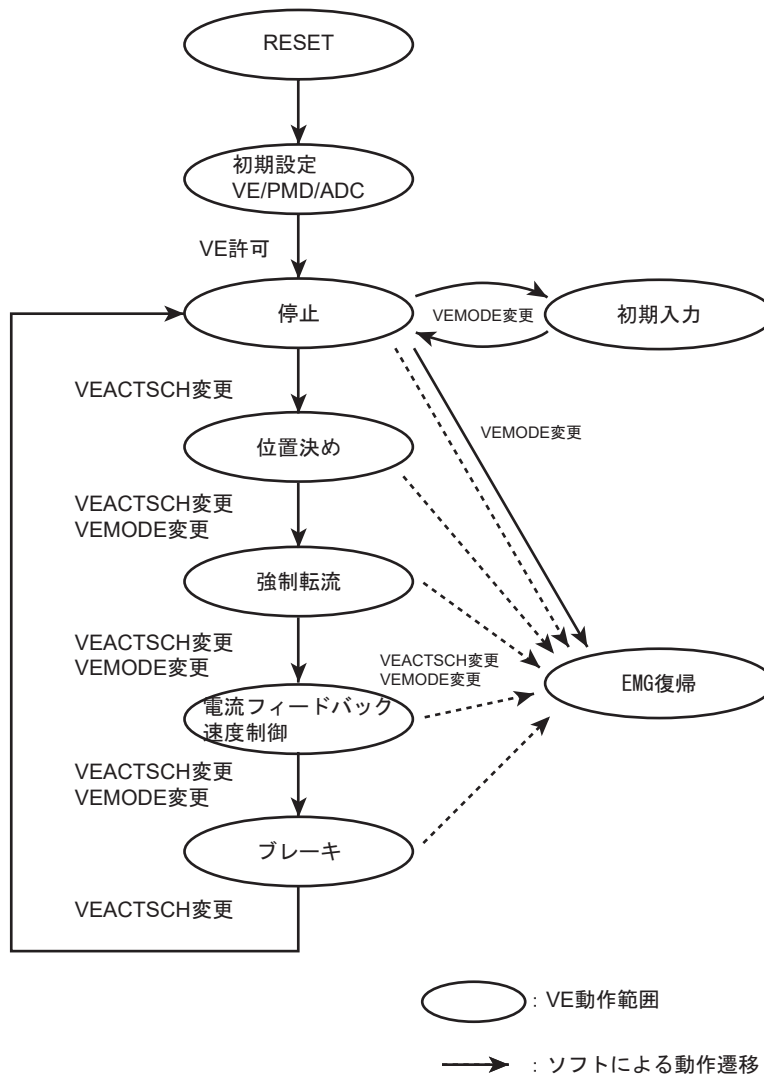


図 13-4 モータ制御の動作状態フロー例

RESET	: マイコンリセット
初期設定	: ユーザー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

13.4.1.1 スケジュール制御

動作スケジュールは VECTSCH 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 13-1 に示します。

また、VEMODE レジスタ設定で位相補間許可 / 出力制御動作 / ゼロ電流検出をモータ制御フローに応じて設定を切り替えます。(表 13-2)

表 13-1 スケジュール別の動作タスク

スケジュール選択 VEACTSCH	出力スケジュール						入力スケジュール		
	電流制御	SIN/COS 演算	出力 座標軸 変換	出力 相変換	出力制御	トリガ 生成	入力処理	入力 相変換	入力 座標軸 変換
0: 個別動作	(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(注 2)	o	o	o(注 3)	o	o(注 4)	o	o
4: スケジュール 4	-	o(注 2)	o	o	o(注 3)	o	o(注 4)	o	o
9: スケジュール 9	-	-	-	-	o(注 3)	o	o(注 4)	-	-

- 注 1) 指定タスクのみ実行
- 注 2) 位相補間設定
- 注 3) 出力オフ設定 : <EMGRS>
- 注 4) ゼロ電流検出でタスク動作を切り替え

表 13-2 代表動作フローでの設定例

設定	スケジュール設定 VEACTSCH	タスク指定 VETASKAPP	位相補間許可 VEMODE	出力制御動作 VEMODE	ゼロ電流検出 VEMODE
モータ制御フロー	<VACTn[3:0]>	<VTASKn[3:0]>	<PVIEN>	<OCRMD[1:0]>	<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

出力スケジュールはコマンド (VECPURUNTRG) 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を待ちます。この時、他チャンネルのスケジュールを実行することができます。

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 (VEREPTIME) 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。

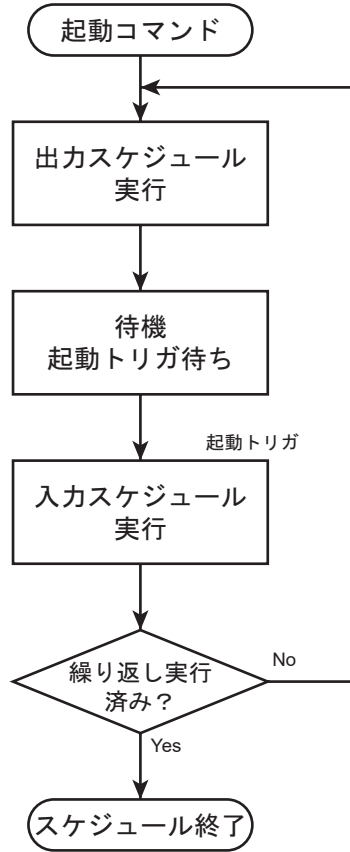


図 13-5 動作スケジュールの動作遷移

### 13.4.1.2 起動制御

ベクトルエンジン許可 (<VEEN> =1) して、動作スケジュール指定 (VEACTSCH)、タスク指定 (VETASKAPP)、繰り返し回数 (VEREPTIME) を設定後

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、基本的には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 (VECPURUNTRG) で指定タスク (VETASKAPP) の開始
2. 入力スケジュール終了後の繰り返しスタート (VEREPTIME ≥ 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起動トリガ (VETRGMODE 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 (VECPURUNTRG) で指定タスク (VETASKAPP) の開始

### 13.4.2 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。

個別実行または起動タスクを指定する場合は表のタスク番号を指定します。

表 13-3 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	dq 電流制御	5
	SIN/COS 演算	正弦 / 余弦演算, 位相補間	6
	出力座標軸変換	dq 座標軸から αβ 座標軸に変換	7
	出力相変換	2 相から 3 相に変換	8
	出力制御	PMD 設定形式へのデータ変換 PWM シフト切り替え	0
	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	ADC 変換結果取り込み 固定小数点数へのデータ変換	2
	入力相変換	3 相から 2 相に変換	3
	入力座標軸変換	αβ 座標軸から dq 座標軸に変換	4

### 13.4.2.1 電流制御

電流制御部は、d 軸 ,q 軸それぞれの PI 制御部で構成されます。

本電流制御で d 軸 ,q 軸電圧を算出します。

#### 1. d 軸電流 PI 制御

##### < 演算式 >

$$\Delta ID = VEIDREFx - \langle ID[31:0] \rangle \quad : \text{電流指令値と電流フィードバックの差分}$$

$$VDIx = VECIDKix \times \Delta ID + VDIx \quad : \text{積分項演算}$$

$$VEVDx = VECIDKPx \times \Delta ID + VDIx \quad : \text{比例項をあわせて電圧算出}$$

	レジスタ名	機能	
入力	VEIDx	d 軸電流	32bit 固定小数点データ (小数点以下 31bit)
	VEIDREFx	d 軸電流基準値	16bit 固定小数点データ (小数点以下 15bit)
	VECIDKPx	比例係数	16bit データ
	VECIDKix	積分係数	16bit データ
出力	VEVDx	d 軸電圧	32bit 固定小数点データ (小数点以下 31bit)
内部	VDIx	d 軸電圧積分項保持	64bit 固定小数点データ (小数点以下 63bit)

#### 2. q 軸電流 PI 制御

##### < 演算式 >

$$\Delta IQ = VEIQREFx - \langle IQ[31:0] \rangle \quad : \text{電流指令値と電流フィードバックの差分}$$

$$VQIx = VECIQKix \times \Delta IQ + VQIx \quad : \text{積分項演算}$$

$$VEVQx = VECIQKPx \times \Delta IQ + VQIx \quad : \text{比例項をあわせて電圧算出}$$

	レジスタ名	機能	
入力	VEIQx	q 軸電流	32bit 固定小数点データ (小数点以下 31bit)
	VEIQREFx	q 軸電流基準値	16bit 固定小数点データ (小数点以下 15bit)
	VECIQKPx	比例係数	16bit データ
	VECIQKix	積分係数	16bit データ
出力	VEVQx	q 軸電圧	32bit 固定小数点データ (小数点以下 31bit)
内部	VQIx	q 軸電圧積分項保持	64bit 固定小数点データ (小数点以下 63bit)



13.4.2.2 SIN/COS 演算

SIN/COS 演算部は、位相補間部と SIN/COS 算出部で構成されます。

位相補間は回転速度を PWM 周期で積分して算出する、位相補間許可時のみ実行します。

1. 位相補間

< 演算式 >

$VETHETAx = VEOMEGAx \times VETPWMx + VETHETAx$  : 速度積分、位相補間許可時のみ

	レジスタ名	機能	
入力	VETHETAx	位相 $\theta$	16bit 固定小数点データ (0.0 ~ 1.0、小数点以下 16bit)
	VEOMEGAx	回転速度	16bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 15bit)
	VETPWMx	PWM 周期レート	16bit 固定小数点データ (0.0 ~ 1.0、小数点以下 16bit)
	VEMODEx	位相補間許可	設定
出力	VETHETAx	位相 $\theta$	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 >

$VESINM = VESIN$  : 前回値保存 (入力処理用)

$VECOSM = VECOS$  : 前回値保存 (入力処理用)

$VESIN = \sin ( VETHETA \times \pi )$  : SIN/COS 演算

$VECOS = \sin ( ( VETHETA + 1/4 ) \times \pi )$  : SIN/COS 演算

	レジスタ名	機能	
入力	VETHETAx	位相 $\theta$	16bit 固定小数点データ (0.0 ~ 1.0、小数点以下 16bit)
出力	VESINx	$\theta$ での正弦値	16bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 15bit)
	VECOSx	$\theta$ での余弦値	
	VESINMx	前回の正弦値	
	VECOSMx	前回の余弦値	

13.4.2.3 出力電圧変換 (座標軸変換 / 相変換)

出力電圧変換は、dq-αβ 座標軸変換と 2-3 相変換で構成されます。

dq-αβ 座標軸変換は Vd, Vq, VESIN, VECOS から Vα, Vβ を算出します。

2-3 相変換は Vα, Vβ からセクタ判定し、判別したセクタに応じた空間ベクトル変換を行い Va, Vb, Vc を算出します。また、2-3 相変換では変換方式に 2 相変調か 3 相変調を選択できます。

1. dq - αβ 座標軸変換

<演算式>

$$VETMPREG3 = VECOSx \times VEVDx - VESINx \times VEVQx \quad : V\alpha \text{ 算出}$$

$$VETMPREG4 = VESINx \times VEVDx + VECOSx \times VEVQx \quad : V\beta \text{ 算出}$$

	レジスタ名	機能	
入力	VEVDx	d 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VEVQx	q 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VESINx	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)
	VECOSx	θ 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	VETMPREG3	α 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG4	β 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)

2. 2 – 3 相変換 (空間ベクトル変換)

a. セクタ判定

<演算式>

```

VESECTORMx = VESECTORx                                : 前回セクタを保存
if (Vα ≥ 0 & Vβ ≥ 0)                                    : Vβ 算出
    if (|Vα| ≥ |Vβ| ÷ 3)
        if (|Vα| ÷ 3 ≥ |Vβ|)                            SECTOR=0
        else                                            SECTOR=1
    else                                                SECTOR=2
else if (Vα < 0 & Vβ ≥ 0)
    if (|Vα| < |Vβ| ÷ 3)                                SECTOR=3
    else if (|Vα| ÷ 3 < |Vβ|)                           SECTOR=4
    else                                                SECTOR=5
else if (Vα < 0 & Vβ < 0)
    if (|Vα| ≥ |Vβ| ÷ 3)
        if (|Vα| ÷ 3 ≥ |Vβ|)                            SECTOR=6
        else                                            SECTOR=7
    else                                                SECTOR=8
else if (Vα ≥ 0 & Vβ < 0)
    if (|Vα| < |Vβ| ÷ 3)                                SECTOR=9
    else if (|Vα| ÷ 3 < |Vβ|)                           SECTOR=10
    else                                                SECTOR=11
    
```

	レジスタ名	機能	
入力	VETMPREG3	α 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG4	β 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
出力	VESECTORx	セクタ	4bit データ
	VESECTORMx	前回のセクタ	4bit データ

b. 3 相電圧算出 (<SECTOR[3:0]> = 0 の場合)

<演算式>

```

if (VESECTORx<SECTOR[3:0]> = 0)
    t1 = (√3) ÷ (VEVDCx) × ((√3) ÷ 2 × Va - 1 ÷ 2 × Vβ)           : V1 期間算出
    t2 = (√3) ÷ (VEVDCx) × (Vβ)                                   : V2 期間算出
    t3 = 1 - t1 - t2                                             : V0 + V7 期間算出
if (VEFMODEx<C2PEN> = 0)                                       : 3相変調
    VETMPREG0 = t1 + t2 + t3 ÷ 2                                 : Va 算出
    VETMPREG1 = t2 + t3 ÷ 2                                     : Vb 算出
    VETMPREG2 = t3 ÷ 2                                          : Vc 算出
else                                                             : 2相変調
    VETMPREG0 = t1 + t2                                         : Va 算出
    VETMPREG1 = t2                                              : Vb 算出
    VETMPREG2 = 0                                              : Vc 算出
    
```

	レジスタ名	機能	
入力	VETMPREG3	α 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG4	β 軸電圧	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VEVDCx	電源電圧	16bit 固定小数点データ (0.0 ~ 1.0、小数点以下 15bit)
	VESECTORx	セクタ	4bit データ
	VEFMODEx	変調モード	設定
出力	VETMPREG0	a 相電圧	32bit 固定小数点データ (0.0 ~ 1.0、小数点以下 31bit)
	VETMPREG1	b 相電圧	32bit 固定小数点データ (0.0 ~ 1.0、小数点以下 31bit)
	VETMPREG2	c 相電圧	32bit 固定小数点データ (0.0 ~ 1.0、小数点以下 31bit)

## 13.4.2.4 出力制御

出力制御部は、3相電圧値をPWM設定形式のVECMPU<sub>x</sub>、VECMV<sub>x</sub>、VECPW<sub>x</sub>に変換して設定し、動作モードに応じてVEOUTCR<sub>x</sub>を設定します。

1 シャント電流検出かつ2相変調でシフトPWM許可している場合、回転速度がPWMシフト切り替え基準より小さいと出力はシフトPWM出力に切り替わります。

	レジスタ名	機能	
入力	VETMPREG0	a相電圧	32bit 固定小数点データ (0.0 ~ 1.0、小数点以下 31bit)
	VETMPREG1	b相電圧	32bit 固定小数点データ (0.0 ~ 1.0、小数点以下 31bit)
	VETMPREG2	c相電圧	32bit 固定小数点データ (0.0 ~ 1.0、小数点以下 31bit)
	VEMDPRD <sub>x</sub>	PWM 周期設定	16bit データ (PMD PWM 周期設定値)
	VESECTOR <sub>x</sub>	セクタ	4bit データ
	VEOMEGA <sub>x</sub>	回転速度	16bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 15bit)
	VEFPWMCHG <sub>x</sub>	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)
	VEMODE <sub>x</sub>	出力制御動作	設定
	VEFMODE <sub>x</sub>	PMD チャンネル / シフト許可 / 変調モード / 検出モード	設定
出力	VECMPU <sub>x</sub>	PMD U 相 PWM 設定	16 ビットデータ (0-MDPRD 値)
	VECMV <sub>x</sub>	PMD V 相 PWM 設定	16 ビットデータ (0-MDPRD 値)
	VECPW <sub>x</sub>	PMD W 相 PWM 設定	16 ビットデータ (0-MDPRD 値)
	VEOUTCR <sub>x</sub>	PMD 出力制御設定	9bit 設定
	VEEMGRS <sub>x</sub>	PMD EMG 復帰	1bit 設定
	VEMCTLF <sub>x</sub>	シフト切り替えフラグ	ステータス

13.4.2.5 トリガ生成

トリガ生成部は、PWM 設定値 VECMPU<sub>x</sub>, VECMPV<sub>x</sub>, VECMPW<sub>x</sub> 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 VETRGCMP0<sub>x</sub>, VETRGCMP1<sub>x</sub> に設定します。

	レジスタ名	機能	
入力	VECMPU <sub>x</sub>	PMD U 相 PWM 設定	16 ビットデータ (0-MDPRD 値)
	VECMPV <sub>x</sub>	PMD V 相 PWM 設定	16 ビットデータ (0-MDPRD 値)
	VECMPW <sub>x</sub>	PMD W 相 PWM 設定	16 ビットデータ (0-MDPRD 値)
	VEMDPRD <sub>x</sub>	PWM 周期設定	16 ビットデータ (PMD PWM 周期設定値)
	VETADC	AD 変換時間	16 ビットデータ (0-MDPRD 値)
	VETRGCRC <sub>x</sub>	トリガ補正值	16 ビットデータ (0-MDPRD 値)
	VESECTOR <sub>x</sub>	セクタ	4bit データ
	VEMODE <sub>x</sub>	出力制御動作	設定
	VEFMODE <sub>x</sub>	PMD チャンネル/ シフト許可/ 変調モード/ 検出モード/ トリガ補正許可	設定
	VEMCTLF <sub>x</sub>	シフト切り替えフラグ	ステータス
出力	VETRGCMP0	PMD トリガ 0 タイミング設定	16 ビットデータ (0-MDPRD 値)
	VETRGCMP1	PMD トリガ 1 タイミング設定	16 ビットデータ (0-MDPRD 値)
	VETRGSSEL <sub>x</sub>	PMD トリガ選択	3bit データ

## 13.4.2.6 入力処理

入力処理タスクでは電流変換結果を 3 相成分判定して保存し、電流および電圧の変換結果を固定小数点データに変換します。また、初期入力動作時にゼロ電流変換結果を保存します。

	レジスタ名	機能	
入力	VEADREG0A	ADC ユニット A 変換結果 0	16bit データ (上位 12bit に結果保持)
	VEADREG1A	ADC ユニット A 変換結果 1	
	VEADREG2A	ADC ユニット A 変換結果 2	
	VEADREG3A	ADC ユニット A 変換結果 3	
	VEADREG0B	ADC ユニット B 変換結果 0	16bit データ (上位 12bit に結果保持)
	VEADREG1B	ADC ユニット B 変換結果 1	
	VEADREG2B	ADC ユニット B 変換結果 2	
	VEADREG3B	ADC ユニット B 変換結果 3	
	VEPHNUM0A	ADREG0A 検出相情報	2bit データ
	VEPHNUM1A	ADREG1A 検出相情報	
	VEPHNUM2A	ADREG2A 検出相情報	
	VEPHNUM3A	ADREG3A 検出相情報	
	VEPHNUM0B	ADREG0B 検出相情報	2bit データ
	VEPHNUM1B	ADREG1B 検出相情報	
	VEPHNUM2B	ADREG2B 検出相情報	
	VEPHNUM3B	ADREG3B 検出相情報	
	VESECTORMx	セクタ情報	4bit データ
	VEMODEx	ゼロ電流検出	設定
	VEFMODEx	PMD チャネル / 電流検出モード / ADC ユニット / シフト許可	設定
	VEMCTLFx	シフト切り替えフラグ	ステータス
出力	VEVDCx	電源電圧	16bit 固定小数点データ (0.0 ~ 1.0、小数点以下 15bit)
	VETMPREG0	a 相電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG1	b 相電流	
	VETMPREG2	c 相電流	
内部	VEIAOx	a 相ゼロ電流変換結果	16bit データ (上位 12bit に結果保持)
	VEIBOx	b 相ゼロ電流変換結果	
	VEICOx	c 相ゼロ電流変換結果	
	VEIAADCx	a 相電流変換結果	16bit データ (上位 12bit に結果保持)
	VEIBADCx	b 相電流変換結果	
	VEICADCx	c 相電流変換結果	

13.4.2.7 入力電流変換 (相変換 / 座標軸変換)

入力電流変換は、3-2 相変換と  $\alpha\beta$ -dq 座標軸変換で構成されます。

3-2 相変換は  $I_a, I_b, I_c$  から  $I_\alpha, I_\beta$  を算出します。

$\alpha\beta$ -dq 座標軸変換は  $I_\alpha, I_\beta, VESINM, VECOSM$  から  $I_d, I_q$  を算出します。

1. 2 - 3 相変換

< 演算式 >

$$\begin{aligned} \text{VETMPREG3} &= \text{VETMPREG0} && : I_\alpha \text{ 算出} \\ \text{VETMPREG4} &= 1 \div 3 \times \text{VETMPREG1} - 1 \div 3 \times \text{VETMPREG2} && : I_\beta \text{ 算出} \end{aligned}$$

	レジスタ名	機能	
入力	VETMPREG0	a 相電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG1	b 相電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG2	c 相電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
出力	VETMPREG3	$\alpha$ 軸電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG4	$\beta$ 軸電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)

2. dq -  $\alpha\beta$  座標軸変換

< 演算式 >

$$\begin{aligned} \text{VEIDx} &= \text{VECOSMx} \times \text{VETMPREG3} + \text{VESINMx} \times \text{VETMPREG4} && : I_d \text{ 算出} \\ \text{VEIQx} &= -\text{VESINMx} \times \text{VETMPREG3} + \text{VECOSMx} \times \text{VETMPREG4} && : I_q \text{ 算出} \end{aligned}$$

	レジスタ名	機能	
入力	VETMPREG3	$\alpha$ 軸電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VETMPREG4	$\beta$ 軸電流	
	VESINMx	$\theta$ での正弦値	16bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 15bit)
	VECOSMx	$\theta$ での余弦値	
出力	VEIDx	d 軸電流	32bit 固定小数点データ (-1.0 ~ 1.0、小数点以下 31bit)
	VEIQx	q 軸電流	



## 13.5 VE チャネルと PMD 及び ADC の組み合わせ

ベクトルエンジンの使用チャンネルにより、使用できる PMD 及び 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。

表 13-4 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
チャンネル 0	チャンネル 0
チャンネル 1	チャンネル 1

表 13-5 ベクトルエンジンと ADC の組み合わせ

ベクトルエンジン			ADC Unit A				ADC Unit B			
チャンネル	VEFMODE (注 2)		ADREG0	ADREG1	ADREG2	ADREG3	ADREG0	ADREG1	ADREG2	ADREG3
	電流検出 <IDMODE[1:0]>	ADC 選択 <ADCSEL[1:0]>								
0	0x	00	電流検出 1	電流検出 2	(注 1)	VDC 検出	-	-	-	-
		1x	電流検出 1	-	(注 1)	VDC 検出	電流検出 2	-	-	-
	1x	00	電流検出 1	電流検出 2	-	VDC 検出	-	-	-	-
1	0x	01	-	-	-	-	電流検出 1	電流検出 2	(注 1)	VDC 検出
		1x	-	電流検出 2	-	-	-	電流検出 1	(注 1)	VDC 検出
	1x	01	-	-	-	-	電流検出 1	電流検出 2	-	VDC 検出

注 1) ADREG2 の変換結果は不要ですが、相情報の設定が必要です。「12 ビット アナログ / デジタルコンバータ」を参照してください。

注 2) 電流検出と ADC 選択は表以外の組み合わせにしないでください。



## 第 14 章 エンコーダ入力回路 (ENC)

### 14.1 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 (2 種類)、タイマモードの 4 つの動作モードに対応しています。また、以下の機能を有しています。

- ・ インクリメンタルエンコーダおよびホール IC センサ対応 (センサ信号を直接入力可能)
- ・ 汎用 24 ビットタイマ機能
- ・ 4 通倍 (6 通倍) 回路内蔵
- ・ 回転方向検出回路内蔵
- ・ カウンタ (24 ビット) 内蔵
- ・ コンペア許可 / 禁止設定可能
- ・ 割り込み出力 1 本
- ・ 入力信号についてデジタルノイズフィルタ内蔵

### 14.2 チャンネル別相違点

TMPM376FDDFG/FDFG は、エンコーダ入力回路を内蔵しています。インクリメンタルエンコーダの信号を直接入力し、モータの絶対位置を容易に得ることができます。

いづれのチャンネルもに示される使用相違点を除いて同一の動作をします。

表 14-1 チャンネル別仕様相違点

チャンネル	入力端子			エンコーダ入力 割り込み
	A 相	B 相	Z 相	
チャンネル 0	PD0 / ENCA0	PD1 / ENCB0	PD2 / ENCZ0	INTENC0
チャンネル 1	PF2 / ENCA1	PF3 / ENCB1	PF4 / ENCZ1	INTENC1

### 14.3 ブロック図

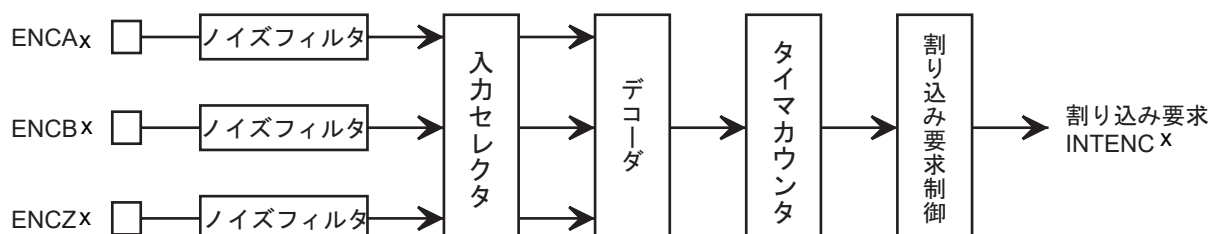


図 14-1 エンコーダ入力回路ブロック図

## 14.4 レジスタ説明

### 14.4.1 レジスタ一覧

エンコーダ入力回路の制御レジスタとアドレスは以下のとおりです。

Channel x	Base Address
Channel0	0x4001_0400
Channel1	0x4001_0500

レジスタ名 (x=0,1)		Address(Base+)
入力制御レジスタ	ENxTNCR	0x0000
カウンタリロードレジスタ	ENxRELOAD	0x0004
比較レジスタ	ENxINT	0x0008
カウンタレジスタ	ENxCNT	0x000C

## 14.4.2 ENxTNCR( 入力制御レジスタ )

	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	-	-	-	-	-	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	CMP	REVERR	UD	ZDET	SFTCAP	ENCLR	ZESEL	CMPEN
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	NR		INTEN	ENDEV		
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。
18-17	MODE[1:0]	R/W	エンコーダ入力モード設定 00 : エンコーダモード 01 : センサモード (イベントカウント) 10 : センサモード (タイマカウント) 11 : タイマモード
16	P3EN	R/W	2相 / 3相入力選択 (センサモード時) (注 1) 0 : 2相入力 1 : 3相入力  入力信号の数を設定します
15	CMP	R	コンペア発生フラグ 0 : - 1 : コンペア発生 (RD でクリア)  コンペアが実施されると <CMP> が "1" にセットされます。 値を読み出すとフラグがクリアされます。また、<ENRUN> = "0" のときは常に "0" がセットされます。 このビットに値を書き込んでも何も起こりません。
14	REVERR	R	反転エラーフラグ (センサモード (タイマカウント) 時) (注 2) 0 : - 1 : エラー発生 (RD でクリア)  センサモード (タイマカウント) 時、反転エラーが発生すると <REVERR> が "1" にセットされます。 値を読み出すとフラグがクリアされます。また、<ENRUN> = "0" のときは常に "0" がセットされます。 このビットに値を書き込んでも何も起こりません。 エンコーダモード、センサモード (イベントカウント) およびタイマモードの場合、このビットは意味を持ちません。
13	UD	R	エンコーダ回転方向 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、<ENRUN> = "0" のときは常に "0" がセットされます。
12	ZDET	R	Z相通過検出 0 : Z非検出 1 : Z検出  <ENRUN> = "0" → "1" 変化後の Z相入力 (ENCZ) の最初の検出エッジ (インクリメンタル型エンコーダ信号の Z相の立ち上がりエッジ (CW 方向)、もしくは立ち下がりエッジ (CCW 方向)) のタイミングで "1" がセットされます。また、<ENRUN> = "0" のときは常に "0" がセットされます。 このビットの動作は <ZEN> の値とは無関係です。  センサモード (イベントカウント) およびセンサモード (タイマカウント) の場合、このビットは常に "0" です。
11	SFTCAP	W	ソフトキャプチャ実行 (タイマモード / センサモード (タイマカウント) 時) 0 : - 1 : ソフトキャプチャ  <SFTCAP> に "1" が書き込まれると、エンコーダカウンタの値をキャプチャし、ENxCNT レジスタに格納します。 "0" を書き込んでも何も起こりません。また、リードすると "0" が読めます。 エンコーダモードおよびセンサモード (イベントカウント) の場合、このビットは意味を持たず、"1" を書き込んでも何も起こりません。
10	ENCLR	W	エンコーダパルスカウンタクリア 0 : - 1 : クリア  <ENCLR> に "1" が書き込まれると、エンコーダカウンタは "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" を書き込んでも何も起こりません。また、リードすると "0" が読めます。

Bit	Bit Symbol	Type	機能								
9	ZESEL	R/W	<p>ENCZ の使用エッジ選択 ( タイマモード時 )</p> <p>0 : 立ち上がりエッジ 1 : 立ち下がりエッジ</p> <p>タイマモード時、外部トリガとして使用する ENCZ 入力のエッジを選択します。 他のモードの場合、このビットは意味を持ちません</p>								
8	COMPEN	R/W	<p>コンペアイネーブル</p> <p>0 : コンペア実行しない 1 : コンペア実行する</p> <p>&lt;COMPEN&gt; に "1" が書き込まれると、エンコーダカウンタのカウンタ値と ENINT レジスタ値のコンペアを実施します。&lt;COMPEN&gt; に "0" が書き込まれると、コンペアを実施しません。</p>								
7	ZEN	R/W	<p>Z相イネーブル ( エンコーダモード / タイマモード時 )</p> <p>0 : 禁止 1 : 許可</p> <p>他のモードの場合、このビットは意味を持ちません。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; padding: 5px;"> <p>&lt;エンコーダモード時&gt; ENCZ 入力によるエンコーダカウンタのクリアを設定</p> </td> <td style="width: 50%; padding: 5px;"> <p>&lt;ZEN&gt; = "1" のとき、CW 方向に回転しているときは ENCZ の立ち上がりエッジを検出してエンコーダカウンタが "0" にクリアされ、CCW 方向に回転しているときは ENCZ の立ち下がりエッジを検出してエンコーダカウンタが "0" にクリアされます。ENCLK (A 相、B 相信号を 4 通倍したクロック) タイミングとこのエッジ検出のタイミングが同時となった場合には、エンコーダカウンタはカウント動作を行わず "0" にクリアされます。(クリア優先)</p> </td> </tr> <tr> <td style="width: 50%; padding: 5px;"> <p>&lt;タイマモード時&gt; ENCZ 入力を外部トリガとして使用するかどうかを設定</p> </td> <td style="width: 50%; padding: 5px;"> <p>&lt;ZEN&gt; = "1" のとき、&lt;ZESEL&gt; で指定された ENCZ 入力のエッジ検出で、エンコーダカウンタの値をキャプチャ、およびエンコーダカウンタを "0" にクリアします。</p> </td> </tr> </table>	<p>&lt;エンコーダモード時&gt; ENCZ 入力によるエンコーダカウンタのクリアを設定</p>	<p>&lt;ZEN&gt; = "1" のとき、CW 方向に回転しているときは ENCZ の立ち上がりエッジを検出してエンコーダカウンタが "0" にクリアされ、CCW 方向に回転しているときは ENCZ の立ち下がりエッジを検出してエンコーダカウンタが "0" にクリアされます。ENCLK (A 相、B 相信号を 4 通倍したクロック) タイミングとこのエッジ検出のタイミングが同時となった場合には、エンコーダカウンタはカウント動作を行わず "0" にクリアされます。(クリア優先)</p>	<p>&lt;タイマモード時&gt; ENCZ 入力を外部トリガとして使用するかどうかを設定</p>	<p>&lt;ZEN&gt; = "1" のとき、&lt;ZESEL&gt; で指定された ENCZ 入力のエッジ検出で、エンコーダカウンタの値をキャプチャ、およびエンコーダカウンタを "0" にクリアします。</p>				
<p>&lt;エンコーダモード時&gt; ENCZ 入力によるエンコーダカウンタのクリアを設定</p>	<p>&lt;ZEN&gt; = "1" のとき、CW 方向に回転しているときは ENCZ の立ち上がりエッジを検出してエンコーダカウンタが "0" にクリアされ、CCW 方向に回転しているときは ENCZ の立ち下がりエッジを検出してエンコーダカウンタが "0" にクリアされます。ENCLK (A 相、B 相信号を 4 通倍したクロック) タイミングとこのエッジ検出のタイミングが同時となった場合には、エンコーダカウンタはカウント動作を行わず "0" にクリアされます。(クリア優先)</p>										
<p>&lt;タイマモード時&gt; ENCZ 入力を外部トリガとして使用するかどうかを設定</p>	<p>&lt;ZEN&gt; = "1" のとき、&lt;ZESEL&gt; で指定された ENCZ 入力のエッジ検出で、エンコーダカウンタの値をキャプチャ、およびエンコーダカウンタを "0" にクリアします。</p>										
6	ENRUN	R/W	<p>エンコーダ動作イネーブル</p> <p>0 : 禁止 1 : 許可</p> <p>&lt;ENRUN&gt; = "1" で、&lt;ZDET&gt; を "0" クリアするとともにエンコーダ動作をイネーブルします。 &lt;ENRUN&gt; = "0" で、エンコーダ動作をディセーブルにします。 エンコーダの動作を停止した際 (&lt;ENRUN&gt; ビットを "1" "0")、クリアされるカウンタおよびフラグと、クリアされないカウンタおよびフラグが存在します。</p>								
5-4	NR[1:0]	R/W	<p>ノイズフィルタ</p> <p>00 : ノイズフィルタなし 01 : 31/fsys 未満のパルスはノイズとして除去 (387.5ns@80MHz) 10 : 63/fsys 未満のパルスはノイズとして除去 (787.5ns@80MHz) 11 : 127/fsys 未満のパルスはノイズとして除去 (1587ns@80MHz)</p> <p>デジタルノイズフィルタでノイズとして除去するパルスの幅を設定します。</p>								
3	INTEN	R/W	<p>エンコーダ割り込みイネーブル</p> <p>0 : 禁止 1 : 許可</p> <p>&lt;INTEN&gt; = "1" で割り込み信号の発生をイネーブルにし、&lt;INTEN&gt; = "0" で割り込み信号の発生をディセーブルにします。</p>								
2-0	ENDEV[2:0]	R/W	<p>エンコーダパルス分周比</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%;">000 : 1 分周</td> <td style="width: 50%;">100 : 16 分周</td> </tr> <tr> <td>001 : 2 分周</td> <td>101 : 32 分周</td> </tr> <tr> <td>010 : 4 分周</td> <td>110 : 64 分周</td> </tr> <tr> <td>011 : 8 分周</td> <td>111 : 128 分周</td> </tr> </table> <p>エンコーダパルスの分周比を設定します。 この設定に従ってエンコーダパルスが分周され、イベント割り込みの発生周期として使用されます。</p>	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
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) エンコーダモード、タイマモードの場合は、必ず <P3EN> = "0" に設定してください。

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。

動作モードは <MODE[1:0]>、<P3EN>、<ZEN> により決定し、全部で 8 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[1:0]>	<ZEN>	<P3EN>	入力端子	モード
00	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)
01	0	0	U,V	センサモード (イベントカウント, 2 相入力)
		1	U,V,W	センサモード (イベントカウント, 3 相入力)
10	0	0	U,V	センサモード (タイマカウント, 2 相入力)
		1	U,V,W	センサモード (タイマカウント, 3 相入力)
11	0	0	-	タイマモード
	1		Z	タイマモード (Z 使用)

<ENRUN> と各信号の状態を以下に示します。

カウンタ/フラグ	<ENRUN> = 0 時 (リセット解除後)	<ENRUN> = 1 時 (動作中)	<ENRUN> = 0 時 (停止中)	<ENRUN> = 0 時 対象フラグ/カウンタの クリア手段
エンコーダカウンタ	0x000000	カウント動作	停止時の値を保持	ソフトクリア (<ENCLR> = 1 WR)
ノイズフィルタ カウンタ	0y0000000	カウントアップ動作	カウントアップ動作 (常時フィルタリング)	リセットのみ
エンコーダパルス 分周カウンタ	0x00	カウントダウン動作	停止してクリア	<ENRUN> = 0 時はクリア
コンペアフラグ <CMP>	0	コンペア時 "1" セット リードでクリア	クリア	<ENRUN> = 0 時はクリア
反転エラーフラグ <REVERR>	0	エラー発生で "1" セット リードでクリア	クリア	<ENRUN> = 0 時はクリア
Z 検出フラグ <ZDET>	0	Z 検出で "1" セット	クリア	<ENRUN> = 0 時はクリア
回転方向ビット <UD>	0	方向検出で "0" / "1" セット	クリア	<ENRUN> = 0 時はクリア



14.4.3 ENxRELOAD( カウンタリロードレジスタ )

	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	RELOAD							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	RELOAD							
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	RELOAD[15:0]	R/W	<p>エンコーダカウンタの周期 ( 4 通倍 (6 通倍) 後 ) 設定 0x0000 ~ 0xFFFF</p> <p>Z 相使用する場合 : 1 回転分のカウントパルス数を設定 Z 相使用しない場合 : 1 回転分のカウントパルス数 - 1 を設定</p> <p>エンコーダカウンタの周期 (4 通倍後) を設定します。 エンコーダカウンタが UP カウントをおこなっていた場合、カウンタの値が &lt;RELOAD[15:0]&gt; の値と等しくなったら、次の ENCLK タイミングで "0" にクリアされます。DOWN カウントをおこなっていた場合、カウンタの値が "0" になった次の ENCLK のタイミングで、&lt;RELOAD[15:0]&gt; の値がエンコーダカウンタにロードされます。</p>

注) エンコーダモード時のみ使用

14.4.4 ENxINT( 比較レジスタ )

	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	INT							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	INT							
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	INT							
リセット後	0	0	0	0	0	0	0	0

Bit	Bit Symbol	Type	機能																								
31-24	-	R	リードすると "0" が読めます。																								
23-0	INT[23:0]	R/W	<p>カウンタ比較値設定</p> <table border="1"> <tr> <td>エンコーダモード時</td> <td>エンコーダのパルス位置割り込み発生位置設定</td> <td>0x0000 ~ 0xFFFF</td> </tr> <tr> <td colspan="3">                     &lt;CMPEN&gt; = "1" のときエンコーダカウンタと &lt;INT[15:0]&gt; の値が一致すると、&lt;CMP&gt; が "1" にセットされます。さらに &lt;INTEN&gt; = "1" であれば割り込み要求 (INTENC0) が発生します。                      ただし、&lt;ZEN&gt; = "1" のときは、&lt;ZDET&gt; = "1" になるまでの間の一致については割り込み要求は発生しません。                 </td> </tr> <tr> <td>センサモード時 (イベントカウント)</td> <td>エンコーダのパルス位置割り込み発生位置設定</td> <td>0x0000 ~ 0xFFFF</td> </tr> <tr> <td colspan="3">                     &lt;CMPEN&gt; = "1" のとき、エンコーダカウンタと &lt;INT[15:0]&gt; の値が一致すると、&lt;CMP&gt; が "1" にセットされます。さらに &lt;INTEN&gt; = "1" であれば割り込み要求 (INTENC0) が発生します。                      &lt;ZEN&gt; の値には影響しません。                 </td> </tr> <tr> <td>センサモード時 (タイマカウント)</td> <td>パルス未検出時間異常判定割り込み発生位置設定</td> <td>0x000000 ~ 0xFFFFF</td> </tr> <tr> <td colspan="3">                     &lt;CMPEN&gt; = "1" のとき、内部カウンタ値が &lt;INT[23:0]&gt; の値と一致すると、パルス未検出時間異常エラーと判定され、&lt;CMP&gt; が "1" にセットされます。さらに &lt;INTEN&gt; = "1" であれば割り込み要求 (INTENC0) が発生します。                      &lt;ZEN&gt; の値には影響しません。                 </td> </tr> <tr> <td>タイマモード時</td> <td>タイマコンペア割り込み発生位置設定</td> <td>0x000000 ~ 0xFFFFF</td> </tr> <tr> <td colspan="3">                     &lt;CMPEN&gt; = "1" のとき、内部カウンタ値が &lt;INT[23:0]&gt; の値と一致すると、&lt;CMP&gt; が "1" にセットされます。さらに &lt;INTEN&gt; = "1" であればタイマコンペア割り込み要求 (INTENC0) が発生します。                      &lt;ZEN&gt; の値には影響しません。                 </td> </tr> </table>	エンコーダモード時	エンコーダのパルス位置割り込み発生位置設定	0x0000 ~ 0xFFFF	<CMPEN> = "1" のときエンコーダカウンタと <INT[15:0]> の値が一致すると、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であれば割り込み要求 (INTENC0) が発生します。 ただし、<ZEN> = "1" のときは、<ZDET> = "1" になるまでの間の一致については割り込み要求は発生しません。			センサモード時 (イベントカウント)	エンコーダのパルス位置割り込み発生位置設定	0x0000 ~ 0xFFFF	<CMPEN> = "1" のとき、エンコーダカウンタと <INT[15:0]> の値が一致すると、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であれば割り込み要求 (INTENC0) が発生します。 <ZEN> の値には影響しません。			センサモード時 (タイマカウント)	パルス未検出時間異常判定割り込み発生位置設定	0x000000 ~ 0xFFFFF	<CMPEN> = "1" のとき、内部カウンタ値が <INT[23:0]> の値と一致すると、パルス未検出時間異常エラーと判定され、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であれば割り込み要求 (INTENC0) が発生します。 <ZEN> の値には影響しません。			タイマモード時	タイマコンペア割り込み発生位置設定	0x000000 ~ 0xFFFFF	<CMPEN> = "1" のとき、内部カウンタ値が <INT[23:0]> の値と一致すると、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であればタイマコンペア割り込み要求 (INTENC0) が発生します。 <ZEN> の値には影響しません。		
エンコーダモード時	エンコーダのパルス位置割り込み発生位置設定	0x0000 ~ 0xFFFF																									
<CMPEN> = "1" のときエンコーダカウンタと <INT[15:0]> の値が一致すると、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であれば割り込み要求 (INTENC0) が発生します。 ただし、<ZEN> = "1" のときは、<ZDET> = "1" になるまでの間の一致については割り込み要求は発生しません。																											
センサモード時 (イベントカウント)	エンコーダのパルス位置割り込み発生位置設定	0x0000 ~ 0xFFFF																									
<CMPEN> = "1" のとき、エンコーダカウンタと <INT[15:0]> の値が一致すると、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であれば割り込み要求 (INTENC0) が発生します。 <ZEN> の値には影響しません。																											
センサモード時 (タイマカウント)	パルス未検出時間異常判定割り込み発生位置設定	0x000000 ~ 0xFFFFF																									
<CMPEN> = "1" のとき、内部カウンタ値が <INT[23:0]> の値と一致すると、パルス未検出時間異常エラーと判定され、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であれば割り込み要求 (INTENC0) が発生します。 <ZEN> の値には影響しません。																											
タイマモード時	タイマコンペア割り込み発生位置設定	0x000000 ~ 0xFFFFF																									
<CMPEN> = "1" のとき、内部カウンタ値が <INT[23:0]> の値と一致すると、<CMP> が "1" にセットされます。さらに <INTEN> = "1" であればタイマコンペア割り込み要求 (INTENC0) が発生します。 <ZEN> の値には影響しません。																											

注) <INT[23:16]> はセンサモード (タイマカウント)、タイマモード時のみ使用

## 14.4.5 ENxCNT(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	CNT							
リセット後	0	0	0	0	0	0	0	0
	15	14	13	12	11	10	9	8
bit symbol	CNT							
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		
31-24	–	R	リードすると "0" が読めます。		
23-0	CNT[23:0]	R/W	エンコーダカウンタ / キャプチャ値		
			エンコーダモード時	エンコーダパルスのカウント値	0x0000 ~ 0xFFFF
			エンコーダパルスのカウント値を読み出すことができます。エンコーダモード時、エンコーダカウンタは、エンコーダパルス (ENCLK) を UP もしくは DOWN カウントします。CW 方向に回転しているときは UP カウントを行い、カウント値が <RELOAD[15:0]> と等しくなったとき、次の ENCLK タイミングで "0" にクリアされます。CCW 方向に回転しているときは DOWN カウントを行い、カウント値が "0" のとき、次の ENCLK タイミングで <RELOAD[15:0]> の値がカウンタにロードされます。		
			センサモード時 (イベントカウント)	エンコーダパルスのカウント値	0x0000 ~ 0xFFFF
エンコーダパルスのカウント値を読み出すことができます。センサモード (イベントカウント) 時、エンコーダカウンタは、エンコーダパルス (ENCLK) を UP もしくは DOWN カウントします。CW 方向に回転しているときは UP カウントを行い、カウント値が "0xFFFF" までカウントすると、次の ENCLK タイミングで "0" にクリアされます。CCW 方向に回転しているときは DOWN カウントを行い、カウント値が "0" のとき、次の ENCLK タイミングで "0xFFFF" がカウンタにロードされます。					
センサモード時 (タイマカウント)	パルス検出時間の値 もしくは ソフトキャプチャ値	0x000000 ~ 0xFFFFFF			
エンコーダパルス (ENCLK) によりエンコーダカウンタをキャプチャした値、もしくは、<SFTCAP> に "1" を書きこむことでエンコーダカウンタをソフトキャプチャした値を読み出すことができます。キャプチャした値はシステムリセットで "0" にクリアされます。また、<ENCLR> = "1" を書き込んでカウンタをクリアした後、ソフトキャプチャすることによりクリアすることも可能です。センサモード (タイマカウント) 時、エンコーダカウンタは、f <sub>sys</sub> で UP カウントし、フリーランで動作します。エンコーダパルス (ENCLK) を検出すると "0" にクリアされます。"0xFFFFFF" までカウントすると自動的に "0" にクリアされます。					
タイマモード時	内部カウンタのキャプチャ値もしくは ソフトキャプチャ値	0x000000 ~ 0xFFFFFF			
<SFTCAP> に "1" を書きこむことでエンコーダカウンタをソフトキャプチャした値を読み出すことができます。<ZEN> = "1" のときは、<ZESEL> で指定された Z 相の検出エッジでもキャプチャされます。キャプチャした値はリセットで "0" にクリアされます。<ENCLR> = "1" を書き込んでカウンタをクリアした後にソフトキャプチャすることにより、クリアすることも可能です。タイマモード時、エンコーダカウンタは、f <sub>sys</sub> で UP カウントし、フリーランで動作します。"0xFFFFFF" までカウントすると自動的に "0" にクリアされます。					

注) <CNT[23:16]> はセンサモード (タイマカウント)、タイマモード時のみ使用。(エンコーダモード、センサモード (イベントカウント) 時は常に "0" が READ されます。)

## 14.5 動作説明

### 14.5.1 エンコーダモード

高速位置センサ対応（位相判定）で、AB エンコーダ入力および ABZ エンコーダ入力に対応しています。

- イベント検出（回転パルス）→ 割り込み発生
- イベントカウント → 一致割り込み発生（移動量測定）
- 回転方向検出
- アップダウンカウント（動作中随時変更可能）
- カウンタ周期設定可能

### 14.5.2 センサモード

低速位置センサ対応（零クロス判定）で、UV ホールセンサ入力および UVW 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（fsys でカウント）の 2 種類があります。

#### 14.5.2.1 イベントカウントモード

- イベント検出（回転パルス）→ 割り込み発生
- イベントカウント → 一致割り込み発生（移動量測定）
- 回転方向検出

#### 14.5.2.2 タイマカウントモード

- イベント検出（回転パルス）→ 割り込み発生
- タイマカウント
- 回転方向検出
- キャプチャ機能 → イベントキャプチャ（イベント間隔測定）→ 割り込み発生  
ソフトキャプチャ
- 未検出時間によるエラー（タイマコンペア）→ 一致割り込み発生
- 反転検出エラー → 回転方向変化によるエラーフラグ

### 14.5.3 タイマモード

汎用 24 ビットタイマとして使用できます。

- 24 ビットアップカウンタ
- カウンタクリア制御（ソフトクリア、タイマクリア、外部トリガ、フリーランカウント）
- コンペア機能 → 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

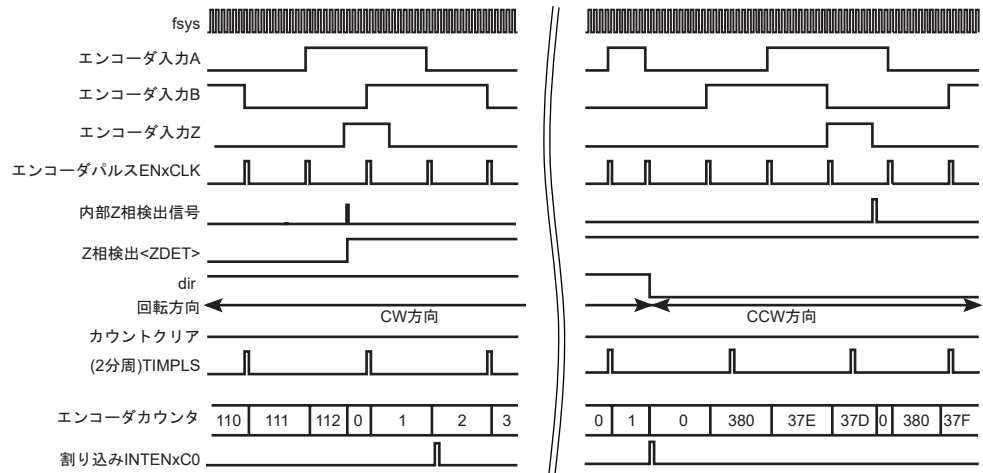
ソフトキャプチャ

## 14.6 機能

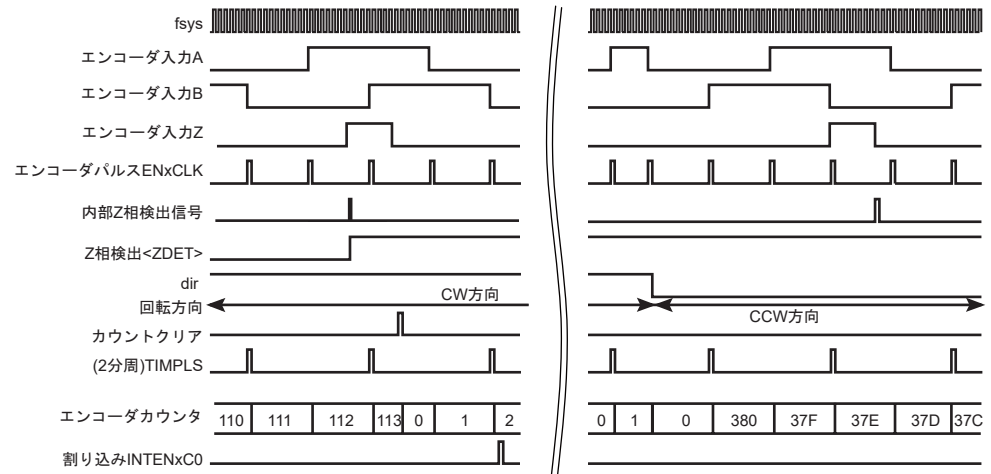
### 14.6.1 モード動作概要

#### 14.6.1.1 エンコーダモード

1. <ZEN> = 1 のとき (<RELOAD> = 0x0380、<EN0INT> = 0x0002)



2. <ZEN> = 0 のとき (<RELOAD> = 0x0380、<EN0INT> = 0x0002)

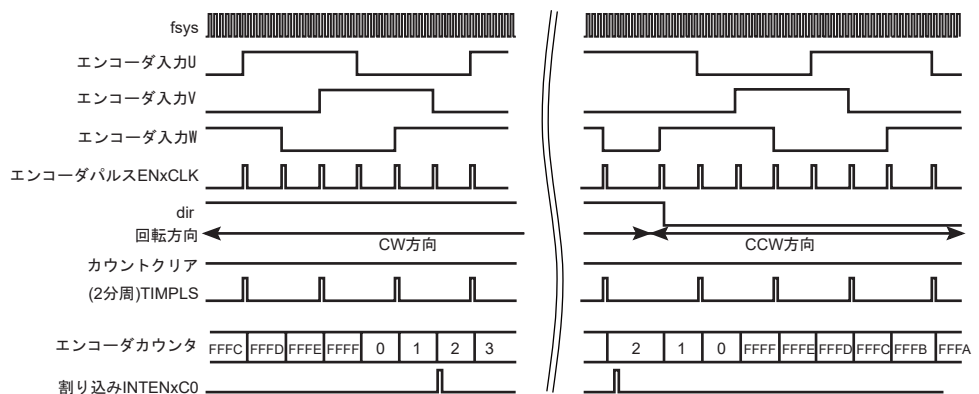


- インクリメンタルエンコーダ入力を A、B、Z 相に接続します。A、B 信号を 4 通り倍して、エンコーダパルス数のカウントを行います。
- CW 方向 (A 相が B 相に対して 90 度進んでいる状態) に回転しているときは Up カウントを行い、カウンタ値が <RELOAD> と等しくなったとき、次の ENCLK でカウンタが "0" クリアされます。
- CCW 方向 (A 相が B 相に対して 90 度遅れている状態) に回転しているときは Down カウントを行い、カウンタ値が "0x0000" と等しくなったとき、次の ENCLK でカウンタに <RELOAD> の値がセットされます。
- 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。

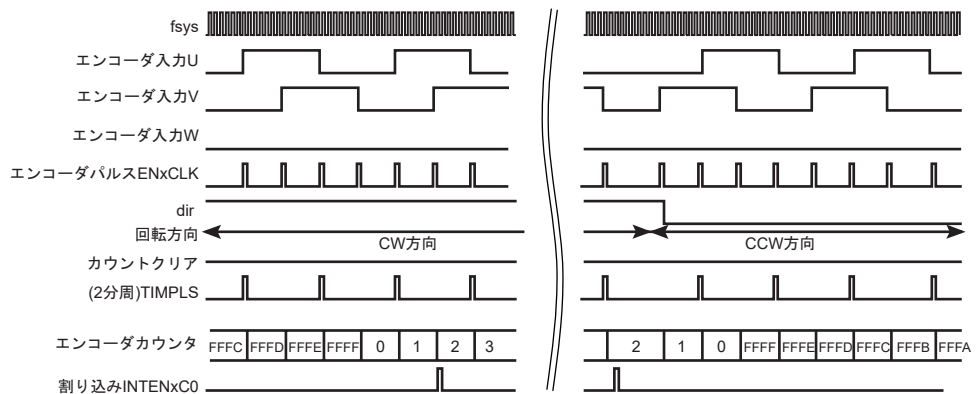
- <ENCLR> に "1" が書き込まれると、カウンタは "0" にクリアされます。
- <UD> は CW 方向に回転していることを検出しているときは "1"、CCW 方向のときは "0" がセットされます。
- ENCLK を分周した信号 (TIMPLS) を取り出すことができます。
- <CMPEN> = "1" のとき、<EN0INT> の値とエンコーダカウンタ値が等しくなったときに割り込みを発生させることができます。ただし、<ZEN> = "1" の場合、<ZDET> = "0" の期間の一致では割り込みを発生しません。
- <ZDET>、<UD> は、<ENRUN> = "0" のときは "0" にクリアされます。

14.6.1.2 センサモード ( イベントカウント )

1. <P3EN> = 1 のとき (<EN0INT> = 0x0002)



2. <P3EN> = 0 のとき (<EN0INT> = 0x0002)

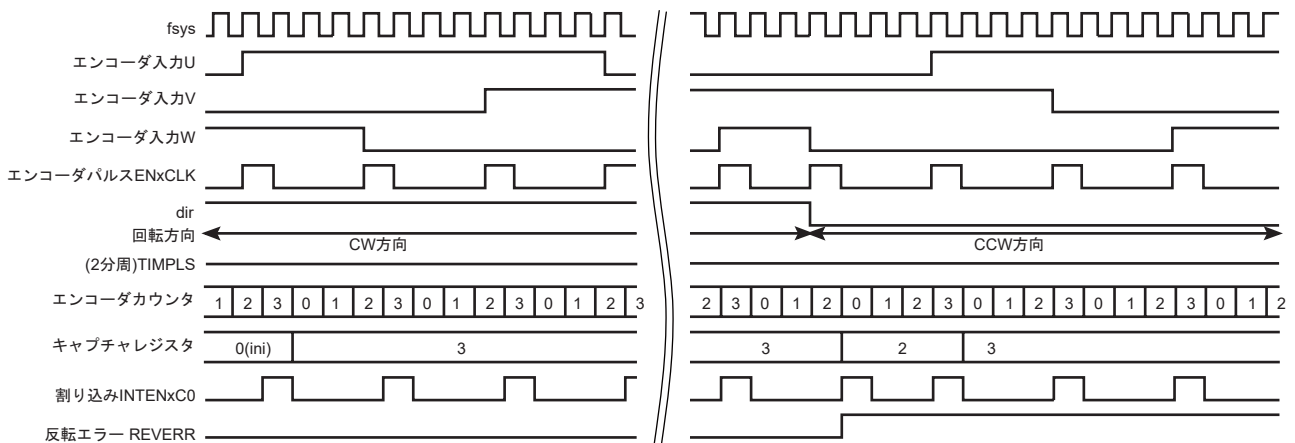


- ホールセンサ入力を U、V、W 相に接続します。<P3EN> = "0" の場合は U、V 信号を 4 通倍、<P3EN> = "1" の場合は U、V、W 信号を 6 通倍して、ホールセンサパルス数のカウントを行います。
- CW 方向 (U 相が V 相、V 相が W 相に対して 90 度進んでいる状態) に回転しているときは Up カウントを行い、カウンタ値が "0xFFFF" と等しくなったとき、次の ENCLK でカウンタが "0" にクリアされます。
- CCW 方向 (U 相が V 相、V 相が W 相に対して 90 度遅れている状態) に回転しているときは Down カウントを行い、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 "0xFFFF" 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。
- <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。

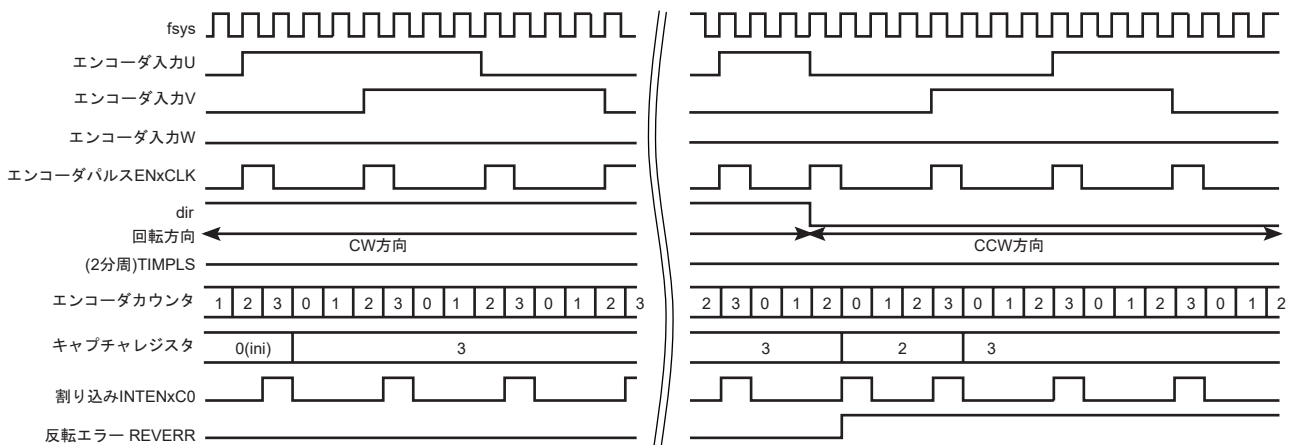
- ENCLK を分周した信号 (TIMPLS) を取り出すことができます。
- <CMPEN> = "1" のとき、<EN0INT> の値とカウンタ値が等しくなったときに割り込みを発生させることができます。
- <UD> は <ENRUN> = "0" のときは "0" にクリアされます。

### 14.6.1.3 センサモード (タイマカウント)

#### 1. <P3EN> = 1 のとき (<EN0INT> = 0x0002)



#### 2. <P3EN> = 0 のとき (<EN0INT> = 0x0002)



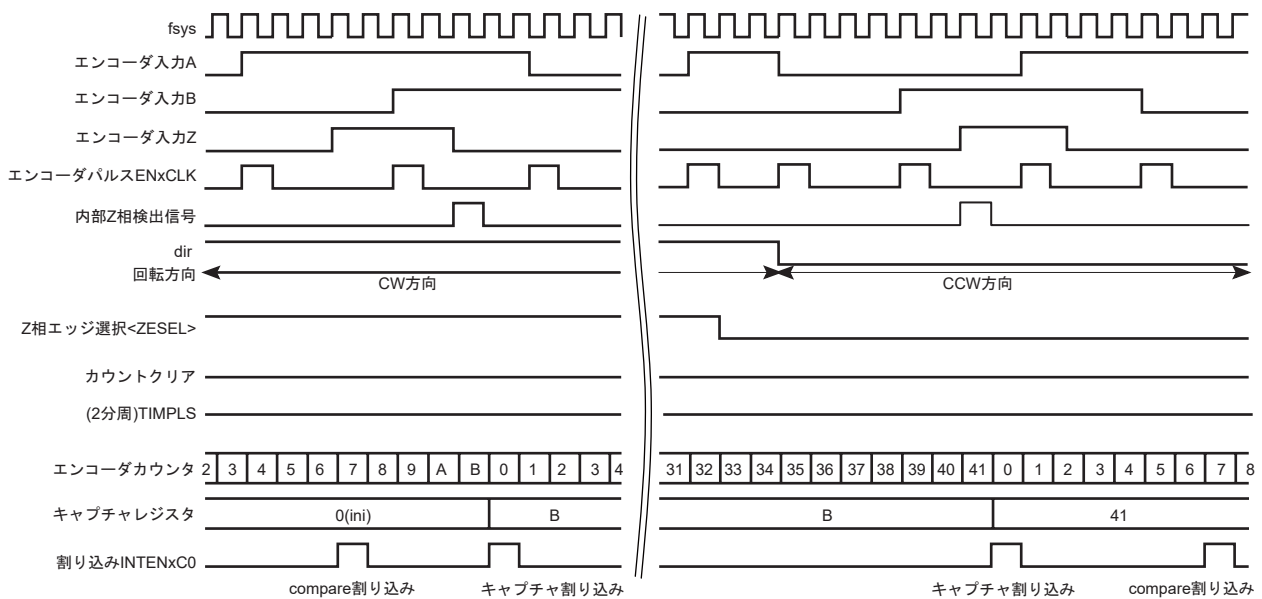
- ホールセンサ入力を U、V、W 相に接続します。<P3EN> = "0" の場合は U、V 信号を 4 通倍、<P3EN> = "1" の場合は U、V、W 信号を 6 通倍して、ホールセンサパルス幅のカウントを行います。
- カウンタは常に Up カウントを行い、ENCLK でカウンタが "0" にクリアされます。また、カウンタ値が "0xFFFFF" と等しくなったとき、カウンタが "0" にクリアされます。
- <ENCLR> に "1" が書き込まれると、カウンタは "0" にクリアされます。
- ENCLK により、発生時のカウンタ値がキャプチャされます。キャプチャした値は ENCNT 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、発生時の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 ENCNT レジスタから読み出すことができます。
- <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。



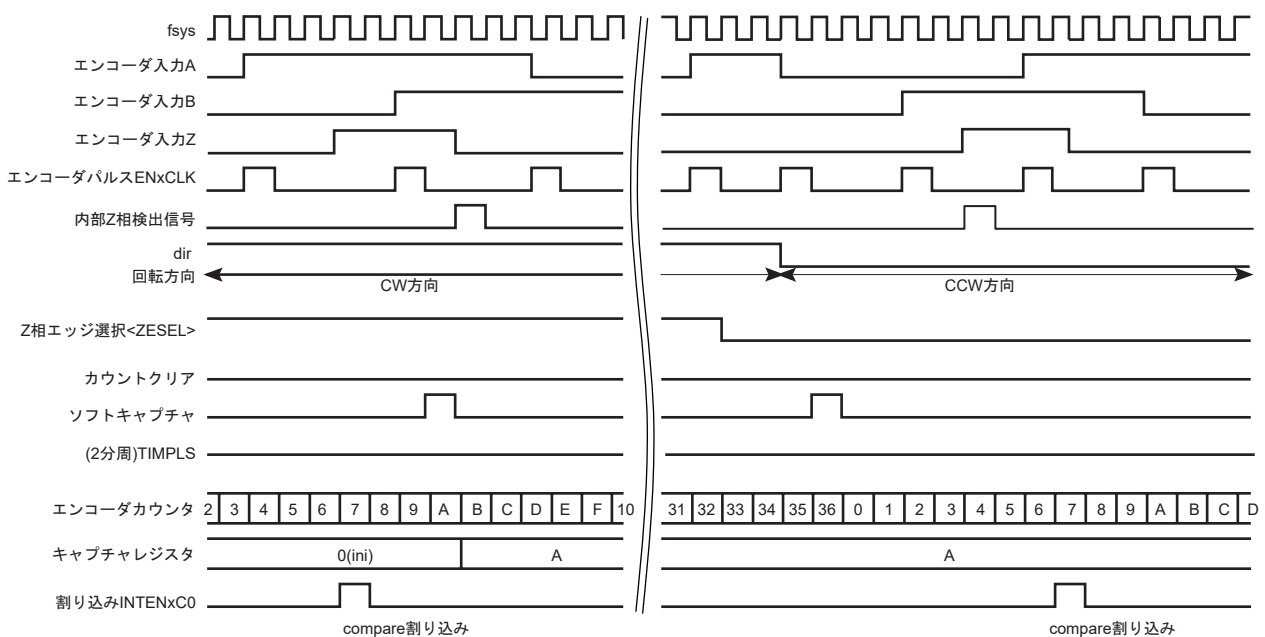
- <CMPEN> = "1" のとき、<EN0INT> の値とカウンタ値が等しくなったときに割り込みを発生させることができます。
- <UD> は <ENRUN> = "0" のときは "0" にクリアされます。
- 回転方向が変化した場合は <REVERR> = "1" にセットされます。フラグは読み出すことでクリアされます。
- ENCNT レジスタの値 (キャプチャ値) は、<ENRUN> の値にかかわらず保持されます。ENCNT レジスタのクリア要因はリセットのみです。

14.6.1.4 タイマモード

1. <ZEN> = 1 のとき (<EN0INT> = 0x0006)



2. <ZEN> = 0 のとき (<EN0INT> = 0x0006)

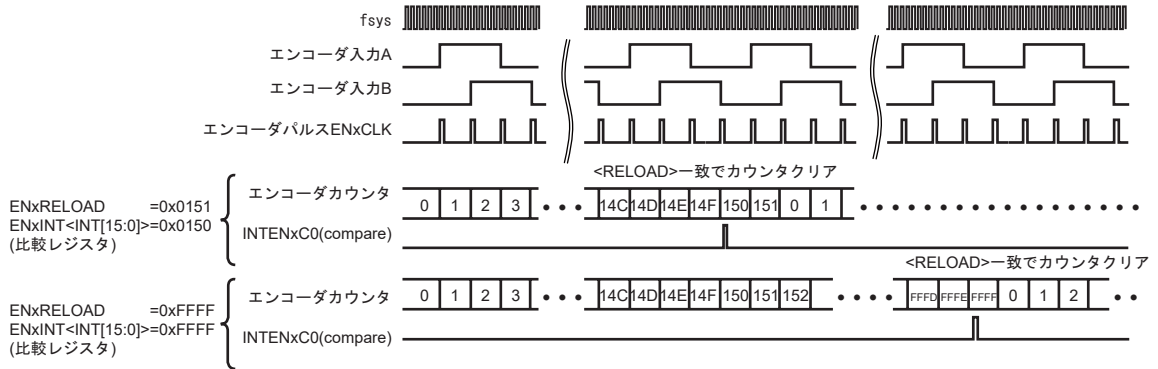


- <ZEN> = "1" のとき、Z 入力端子を外部トリガとして使います。<ZEN> = "0" のとき、外部入力を使用しません。

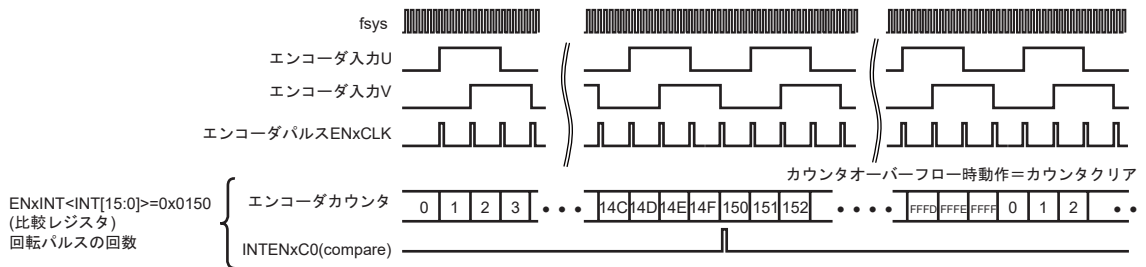
- カウンタは常に Up カウントを行います。〈ZEN〉 = "1" の場合、〈ZESEL〉 = "0" のときは Z 相の立ち上がりエッジでカウンタが "0" にクリアされ、〈ZESEL〉 = "1" のときは Z 相の立ち下がりエッジでカウンタが "0" にクリアされます。また、エンコーダカウンタ値が "0xFFFFFFFF" と等しくなったとき、カウンタが "0" にクリアされます。
- 〈ENCLR〉 に "1" が書き込まれると、カウンタは "0" にクリアされます。
- Z 相検出により、発生時のカウンタ値がキャプチャされます。キャプチャした値は ENCNT レジスタから読み出すことができます。
- 〈SFTCAP〉 に "1" が書き込まれると、発生時のエンコーダ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 ENCNT レジスタから読み出すことができます。
- 〈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。
- 〈CMPEN〉 = "1" のとき、〈EN0INT〉 の値とエンコーダカウンタ値が等しくなったときに割り込みを発生させることができます。
- 〈UD〉 は 〈ENRUN〉 = "0" のときは "0" にクリアされます。
- ENCNT 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。ENCNT レジスタのクリア要因はリセットのみです。

14.6.2 カウンタおよび割り込み発生動作 <CMPEN> = 1 のとき

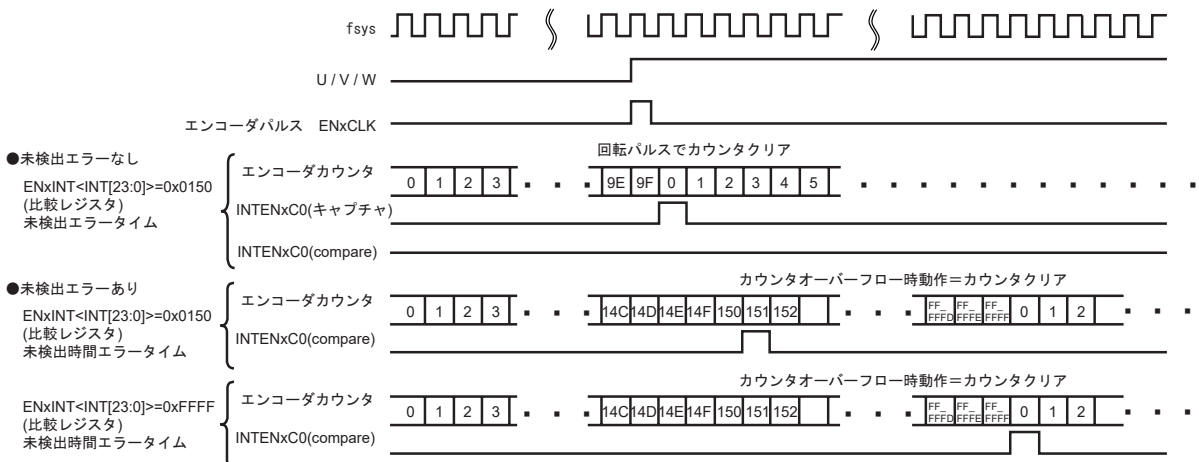
14.6.2.1 エンコーダモード



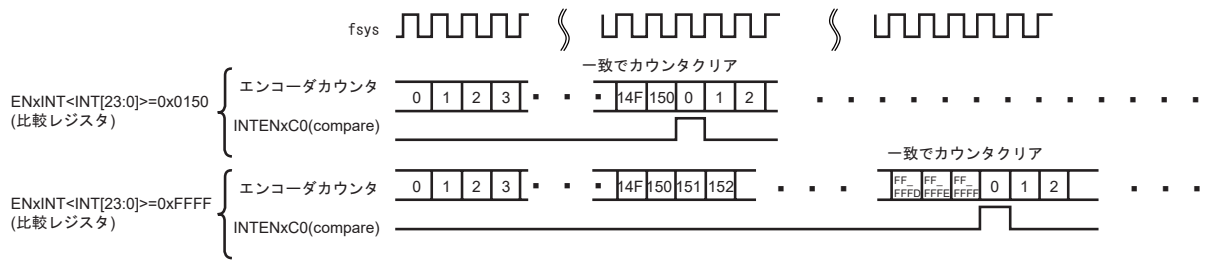
14.6.2.2 センサモード ( イベントカウント )



14.6.2.3 センサモード ( タイマカウント )



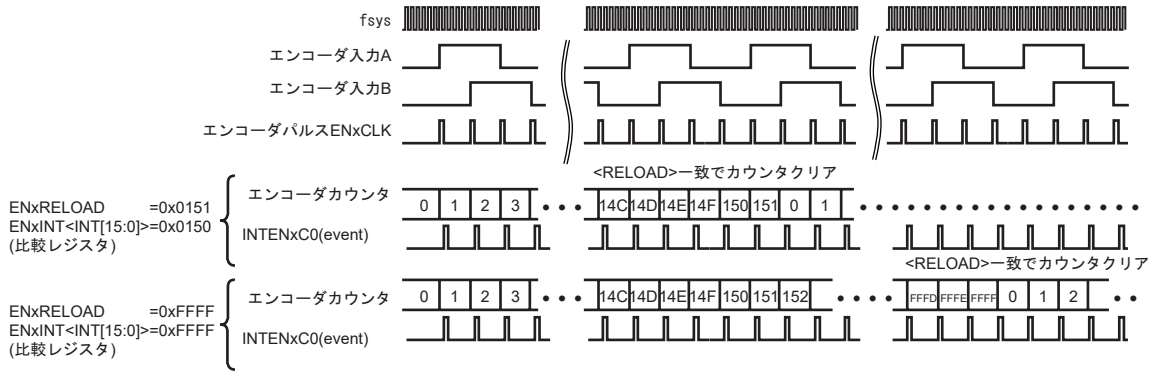
14.6.2.4 タイマモード



14.6.3 カウンタおよび割り込み発生動作 <CMPEN> = 0 のとき

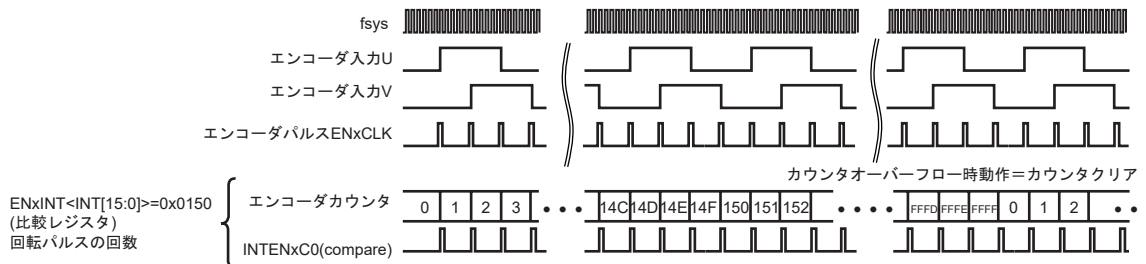
14.6.3.1 エンコーダモード

<ENDEV>="000"

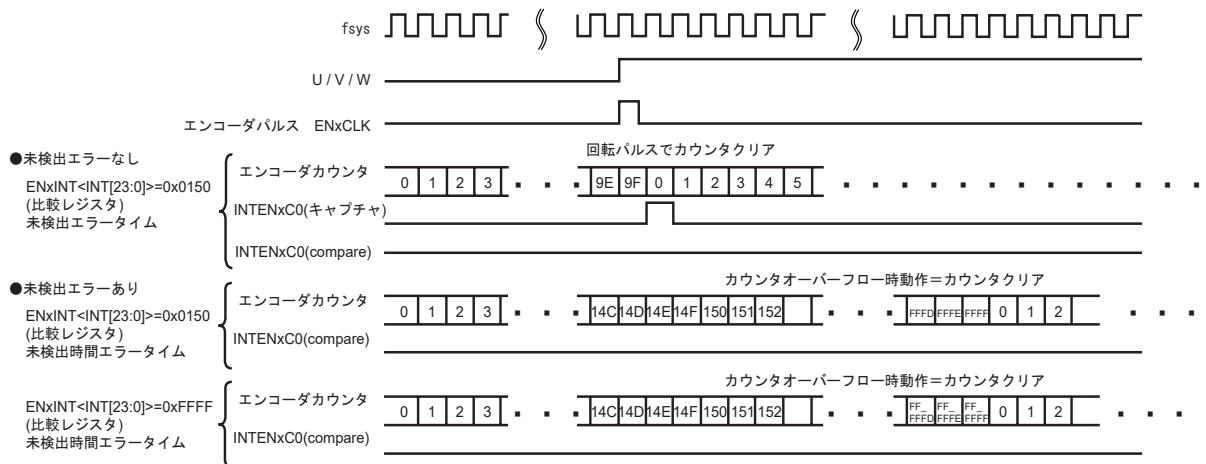


14.6.3.2 センサモード ( イベントカウント )

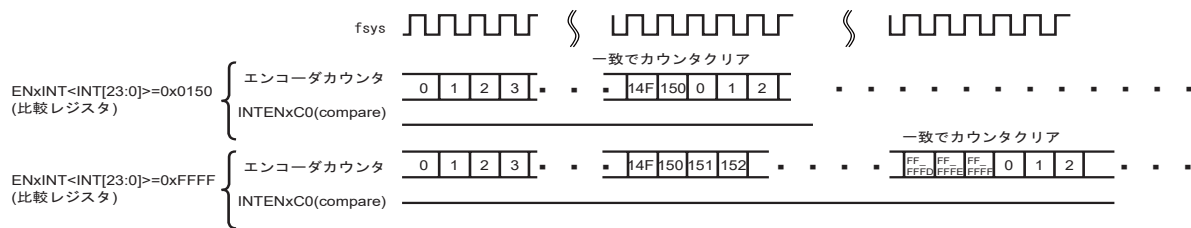
<ENDEV>="000"



14.6.3.3 センサモード ( タイマカウント )



14.6.3.4 タイマモード



## 14.6.4 エンコーダ回転方向

A,B,Z 相の位相を判定します。

この回路は 2 相 (A,B) / 3 相 (A,B,Z) 入力兼用で使用します。3 相入力時は <P3EN> = "1" に設定します。

	2 相入力時	3 相入力時
CW 方向	<p>A 0 1 1 0 0 1 B 0 0 1 1 0 0</p>	<p>A 0 1 1 1 0 0 0 1 1 B 0 0 0 1 1 1 0 0 0 Z 1 1 0 0 0 1 1 1 0</p>
CCW 方向	<p>A 0 0 1 1 0 0 B 0 1 1 0 0 1</p>	<p>A 1 1 0 0 0 1 1 1 0 B 0 0 0 1 1 1 0 0 0 Z 0 1 1 1 0 0 0 1 1</p>

### 14.6.5 カウンタ回路

カウンタ回路は、24bit のアップダウンカウンタを内蔵し、カウンタの制御を行います。

#### 14.6.5.1 動作概要

動作モードにより、カウント動作、クリア、リロードを制御します。カウンタの制御を表 14-2 に示します。

表 14-2 カウンタの制御

モード <MODE[1:0]>	<ZEN>	<P3EN>	入力 端子	カウント	動作	カウンタ クリア条件	カウンタ リロード条件	カウンタ動作 可能範囲 (リロード値)		
エンコーダモード 00	0	0	A,B	エンコー ダパルス (ENCLK)	UP	[1]<ENCLR> = 1 WR [2]<RELOAD> と一致	-	0x0000~<REL OAD>		
			DOWN		[1]<ENCLR> = 1 WR	[1]0x0000 と一致				
	1		A,B,Z		UP	[1]<ENCLR> = 1 WR [2]<RELOAD> と一致 [3]Z トリガ	-			
			DOWN		[1]<ENCLR> = 1 WR	[1]0x0000 と一致				
センサモード (イベントカウント) 01	0	0	U,V	fsys	UP	[1]<ENCLR> = 1 WR [2]0xFFFF と一致	-	0x0000~0xFFFF F		
			DOWN		[1]<ENCLR> = 1 WR	[1]0x0000 と一致				
		1	U,V,W		UP	[1]<ENCLR> = 1 WR [2]0xFFFF と一致	-			
			DOWN		[1]<ENCLR> = 1 WR	[1]0x0000 と一致				
センサモード (タイムカウント) 10	0	0	U,V	fsys	UP	[1]<ENCLR> = 1 WR [2]0xFFFFF と一致	-	0x000000~0xF FFFFF		
		1	U,V,W		UP	[3] エンコーダパルス (ENCLK)	-			
タイマモード 11	0	×	-		fsys	UP	[1]<ENCLR> = 1 WR [2] 0xFFFFF と一致 [3]<EN0INT> と一致		-	0x000000 ~ 0xFFFFF
			Z			UP	[1]<ENCLR> = 1 WR [2] 0xFFFFF と一致 [3] <EN0INT> と一致 [4]Z トリガ		-	

注) カウンタの値は、ENRUN = "0" を書き込んででもクリアされません。また、再度 ENRUN = 1 とすると、停止時のカウンタ値からカウントを再開します。カウンタ値を "0" にクリアする場合は、<ENCLR> = "1" を書き込むことでソフトクリアしてください。



## 14.6.6 割り込み

割り込みには、イベント(分周パルス、キャプチャ)割り込み、未検出時間異常割り込み、タイマコンペア割り込み、キャプチャ割り込みがあります。

### 14.6.6.1 動作概要

<INTEN> = "1" のとき、カウンタ値およびエンコーダパルスにより割り込みを発生します。

割り込みの要因は、動作モードおよび <CMPEN>、<ZEN> の設定により以下の 6 種類があります。表 14-3 に割り込み要因を示します。

表 14-3 割り込み要因

	割り込み要因	説明	モード	割り込み出力	Status フラグ
1	イベントカウント割り込み	<CMPEN> = "1" のとき、イベント(回転パルス)の発生をカウントするカウンタを使用し、設定回数(= <EN0INT>) カウントされたことを通知します。	エンコーダモード および センサモード (イベントカウント)	<INTEN> = "1" かつ <CMPEN> = "1" 時	あり <CMP>
2	イベント割り込み (分周パルス)	イベント(エンコーダパルス)の発生を、<ENDEV> の設定により 1 分周 ~ 128 分周して、通知します。		<INTEN> = "1" 時	なし
3	イベント割り込み (キャプチャ割り込み)	イベント(エンコーダパルス)の発生およびイベント(回転パルス)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)	<INTEN> = "1" 時	なし
4	未検出時間異常割り込み	<CMPEN> = "1" のとき、fsys でカウントしイベント(エンコーダパルス)でクリアするカウンタを使用し、イベントがある一定時間(= <EN0INT>) 以上発生しないことを通知します。		<INTEN> = "1" かつ <CMPEN> = "1" 時	あり <CMP>
5	タイマコンペア割り込み	<CMPEN> = "1" のとき、タイマで設定時間(= <EN0INT>) カウントしたことを通知します。	タイマモード	<INTEN> = "1" かつ <CMPEN> = "1" 時	あり <CMP>
6	キャプチャ割り込み	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。		<INTEN> = "1" 時	なし

センサモード(タイマカウント)およびタイマモード時は、エンコーダカウンタのキャプチャ動作が可能です。

キャプチャした値は、ENCNT レジスタから読み出すことができます。

センサモード(タイマカウント)時は、イベント発生(エンコーダパルス)により、カウンタの値がキャプチャされます。<SFTCAP> に "1" を書き込むことにより、ソフトキャプチャも可能です。

タイマモード時は、<SFTCAP> に "1" を書き込むことにより、ソフトキャプチャが可能です。<ZEN> = "1" 設定時は、ENCZ 入力を使用して、<ZESEL> に従ったエッジでの外部トリガキャプチャも可能です。



## 15 パワーオンリセット回路

パワーオンリセット回路は、電源投入時にパワーオンリセット信号を発生します。また、電源電圧がパワーオンリセット回路の検出電圧以下の時、パワーオンリセット信号を発生します。

### 15.1 構成

パワーオンリセット回路は、基準電圧発生回路とコンパレータおよびパワーオンカウンタから構成されます。

電源電圧をラダー抵抗によって分圧した電圧と基準電圧発生回路が発生した基準電圧をコンパレータで比較します。

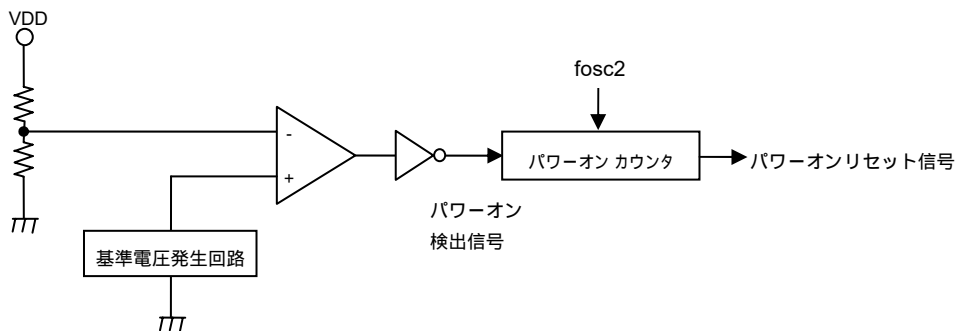


図 15-1 パワーオンリセット回路

### 15.2 機能

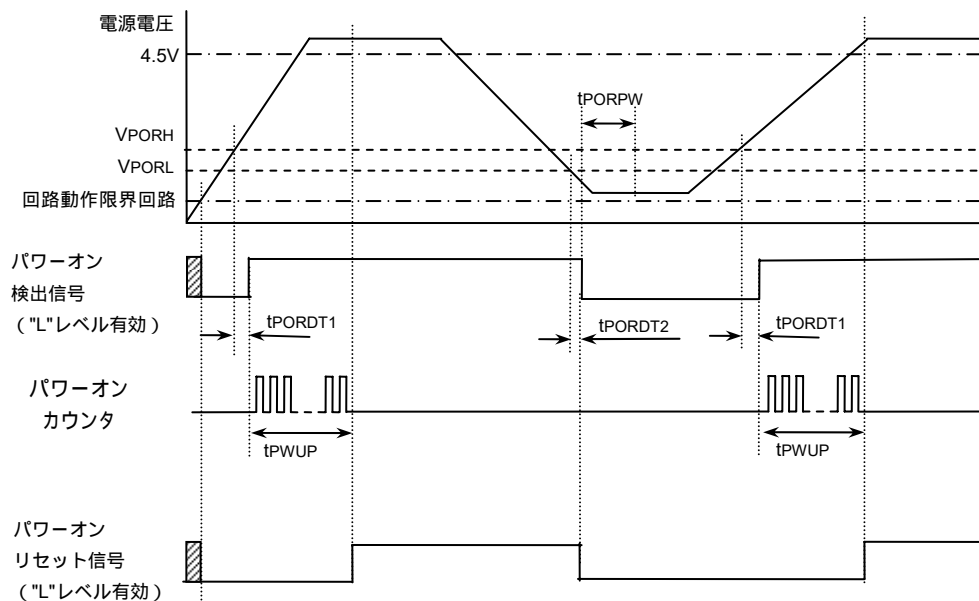
電源投入時、電源電圧がパワーオンリセットの解除電圧以下の間、パワーオンリセット信号が発生されます。電源電圧が解除電圧を超えるとパワーオンカウンタ回路が動作し、 $2^{15}/f_{osc2}$  (s)後にパワーオンリセット信号が解除されます。

電源遮断時、電源電圧がパワーオンリセットの検出電圧以下になるとパワーオンリセット信号が発生されます。

パワーオンリセット信号が発生されている間、パワーオンカウンタ回路、CPU、周辺回路はリセットされます。

外部リセット入力を使用しないでパワーオンリセット回路のみを使用する場合は、パワーオンリセットの解除電圧検出から 3ms 以内に電源電圧を推奨動作範囲(注)まで上昇させてください。電源電圧が推奨動作範囲に到達しない場合、TMPM376 は正常に動作することができません。

(注) 電源立ち上がり時、DVDD5 端子が推奨動作電圧範囲(4.5 ~ 5.5V)に到達し 200 $\mu$ s 経過するまでの期間、ポート L(PL0 および PL1 端子)はオープンまたは 0.5V 以下となるようにしてください。



注 1) 電源電圧 (DVDD5) 変動によっては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ので、機器設計的には電気的特性を参照の上、十分な考慮が必要です。

注 2) 回路動作限界 (基準電圧発生回路が動作できない電圧) 以下ではパワーオンリセット信号は不定になります。

図 15-2 パワーオンリセットの動作タイミング

記号	項目	Min	Typ.	Max	単位
VPORH	パワーオンリセット解除電圧	2.8	3	3.2	V
VPORL	パワーオンリセット検出電圧	2.6	2.8	3.0	V
tPORDT1	パワーオンリセット解除応答時間		30.		μs
tPORDT2	パワーオン検出応答時間		30		μs
tPORPW	パワーオンリセット最小パルス幅	45			μs

注 1) パワーオンリセット解除電圧とパワーオンリセット検出電圧は、相対的に変動するため検出電圧が逆転することはありません。

電源投入シーケンスの詳細は『電気的特性』の章を参照してください。

外部リセット入力を使用する場合の詳細は『例外』の章の『リセット例外』を参照してください。

## 16 電圧検出回路 (VLTD)

電圧検出回路は、電源電圧の低下を検出し、リセット信号を発生します。

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。

### 16.1 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 (VDD) 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。電源電圧 (DVDD5) が検出電圧 (VDLVL) を下回ると、リセット信号を発生します。

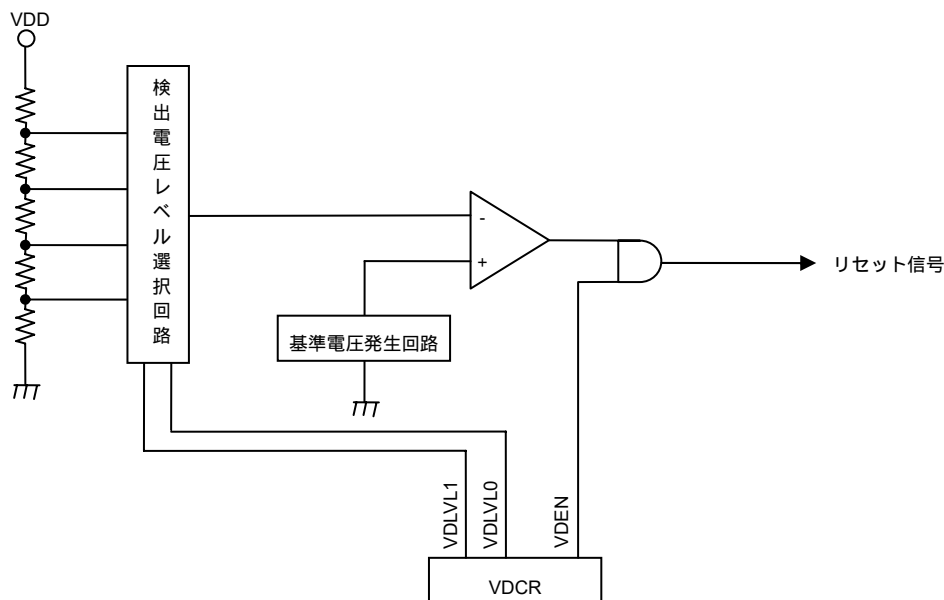


図 16-1 電圧検出回路

## 16.2 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。

### 電圧検出制御レジスタ

VDCR (0x4004_0900)		7	6	5	4	3	2	1	0
Bit Symbol	-	-	-	-	-	-	VDLVL1	VDLVL0	VDEN
Read/Write	R	R	R	R	R	R	R/W		R/W
リセット後	0	0	0	0	0	0	00		0

VDLVL[1:0]	検出電圧レベル選択	00 : Reserved 01 : 4.1 ± 0.2 V 10 : 4.4 ± 0.2 V 11 : 4.6 ± 0.2 V
VDEN	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。

## 16.3 機能

電圧検出回路は、検出電圧レベル選択ビット (VDLVL[1:0])、電圧検出の許可 / 禁止 (VDEN) で設定できます。電圧検出の許可 / 禁止を設定し、電源電圧 (DVDD5) が検出電圧 (VDLVL[1:0]) を下回ったとき、リセット信号を発生します。

### 16.3.1 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、"0" にクリアされ禁止されます。"1" にセットすると電圧検出動作が許可されます。

注) 電源電圧 (VDD) < 検出電圧 (VDLVL) の状態で、VDCR<VDEN> = "1" (禁止) から "0" (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。

16.3.2 検出電圧レベル選択

VDCR<VDLVL[1:0]> で検出電圧を選択します。

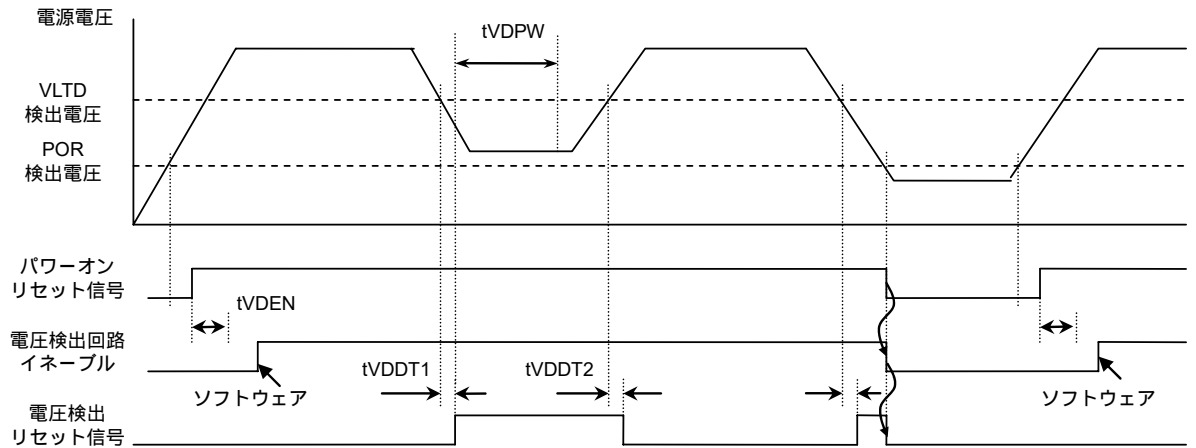


図 16-2 電圧検出タイミング

記号	項目	Min	Typ.	Max	単位
tVDEN	電圧検出回路が有効になる時間		40		μs
tVDDT1	電圧検出回路検出応答時間		40		μs
tVDDT2	電圧検出回路検出解除時間		40		μs
tVDPW	電圧検出回路検出最小パルス幅	45			μs





## 17 周波数検知回路 ( Oscillation Frequency Detector )

### 17.1 構成

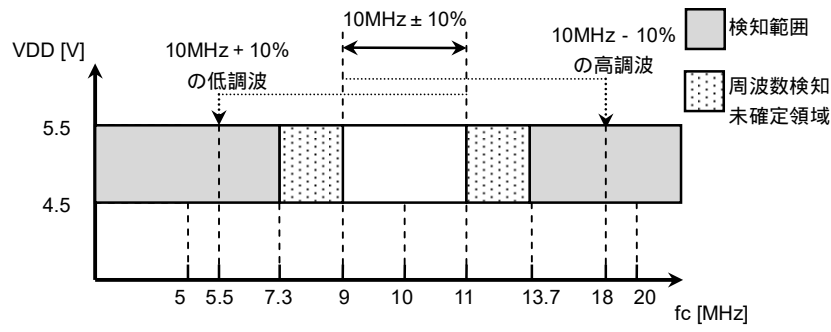
周波数検知回路 (Oscillation Frequency Detector) は、CPU クロック用高周波の周波数が検知周波数設定レジスタによって設定された周波数範囲を超えた場合にリセットを発生する回路です。

検知する周波数の上限と下限は OFDMXPLLOFF と OFDMNPLLOFF レジスタで設定します。TMPM376FD の検出周波数の初期値は図 17-1 を参照してください。OFDMXPLLOFF, OFDMXPLLON, OFDMNPLLOFF 及び OFDMNPLLON レジスタは周波数検知動作中は書き換えできませんので、検知周波数を変更する場合は周波数検知が停止中におこなってください。また、OFDMXPLLOFF, OFDMXPLLON, OFDMNPLLOFF 及び OFDMNPLLON レジスタに書き込みをおこなうためには、周波数検知回路制御レジスタ 1 (OFDCR1) に書き込み許可コード "0xF9" を設定する必要があります。周波数検知回路は RESET 端子への外部リセット入力、パワーオンリセットおよび VLTD リセットによりディセーブルとなります。検知動作をイネーブルにするためには、OFDCR1 に書き込み許可コード "0xF9" を書き込んだ後、周波数検知回路制御レジスタ 2 (OFDCR2) に "0xE4" を設定します。

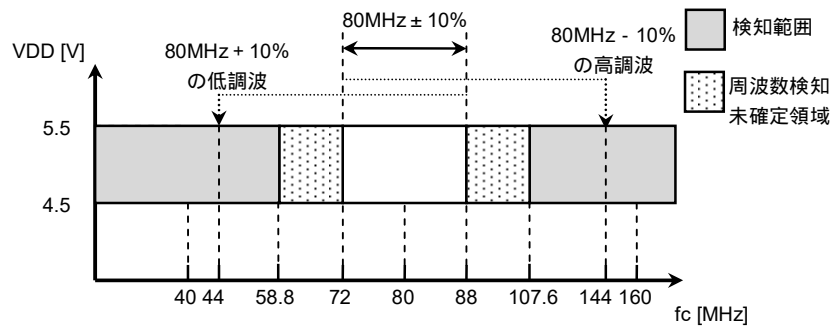
OFDMNPLLOFF, OFDMNPLLON, OFDMXPLLOFF 及び OFDMXPLLON レジスタで設定した周波数範囲を超えた場合、TMPM376FD は周波数検知リセットを発生し、電源端子、RESET 端子、X1 および X2 端子を除く全ての入出力ポートはハイインピーダンス状態となります。CPU などの内部回路はクロックに同期したリセットによって初期化されますので、高周波クロックの停止により周波数検知リセットが発生した場合、高周波クロックの発振が再開するまで内部回路は初期化されません。

周波数検知回路の全てのレジスタ (OFDCR1, OFDCR2, OFDMNPLLOFF, OFDMNPLLON, OFDMXPLLOFF, OFDMXPLLON) は、周波数検知リセットでは初期化されず、周波数検知リセット発生後も検知動作は継続します。したがって、外部高速発振クロック (fosc1) の異常により、周波数検知リセットが発生した場合、その後周波数検知リセットの解除により内部高速発振クロック (fosc2) をシステムクロックとして起動します。

- 注 1) 周波数検知回路は NORMAL モードおよび IDLE モードのときのみ有効となります。STOP モードのときは、周波数検知回路は自動的にディセーブルとなります。
- 注 2) CGPLLSEL レジスタにて PLL を設定する場合、周波数検知回路 (OFD) は必ずディセーブルの状態でおこなってください。なお、PLL オンの状態で OFD リセットが発生した場合、検知周波数設定レジスタは、OFDMNPLLON/OFDMXPLLON から OFDMNPLLOFF/OFDMXPLLOFF に自動的に切り替わります。

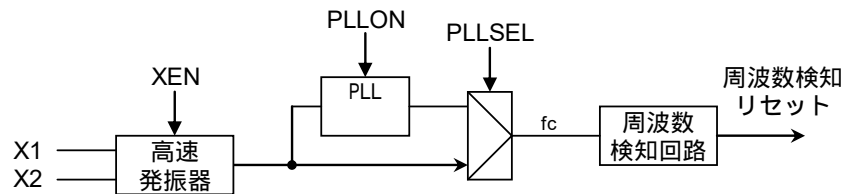


(a) PLL OFF 時



(b) PLL ON 時

図 17-1 検出周波数範囲 例



注)CGPLLSEL レジスタにて PLL の設定をおこなう場合、必ず OFD がディセーブルの状態でおこなってください。

図 17-2 周波数検知回路

17.2 制御

周波数検知回路の動作と停止は、周波数検知回路制御レジスタ 2 (OFDCR2) で制御されます。検知周波数の上限値と下限値は OFDMNPLLOFF, OFDMNPLLON, OFDMXPLLOFF 及び OFDMXPLLON レジスタによって設定します。OFDCR2, OFDMNPLLOFF, OFDMNPLLON, OFDMXPLLOFF 及び OFDMXPLLON レジスタへの書き込みは周波数検知回路制御レジスタ 1 (OFDCR1) によって制御されます。

周波数検知回路制御レジスタ 1

OFDCR1 (0x4004_0800)	31-8								
	Bit Symbol	-							
	Read/Write	R							
	リセット後	0							
		7	6	5	4	3	2	1	0
	Bit Symbol	OFDWEN7	OFDWEN6	OFDWEN5	OFDWEN4	OFDWEN3	OFDWEN2	OFDWEN1	OFDWEN0
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	リセット後	0	0	0	0	0	1	1	0
	機能	0x06: OFDCR2/OFDMNPLLOFF/OFDMNPLLON/OFDMXPLLOFF/OFDMXPLLON レジスタへの書き込み禁止 (書き込み禁止コード) 0xF9: OFDCR2/OFDMNPLLOFF/OFDMNPLLON/OFDMXPLLOFF/OFDMXPLLON レジスタへの書き込み許可 (書き込み許可コード) その他: Reserved (注 1)							

注 1) OFDCR1 への書き込みデータは "0x06" と "0xF9" のみ有効です。"0x06" と "0xF9" 以外のデータを OFDCR1 に書き込もうとした場合、自動的に "0x06" が書かれます。

注 2) OFDCR1 は外部リセット (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。

周波数検知回路制御レジスタ 2

OFDCR2 (0x4004_0804)	31-8								
	Bit Symbol	-							
	Read/Write	R							
	リセット後	0							
		7	6	5	4	3	2	1	0
	Bit Symbol	OFDEN7	OFDEN6	OFDEN5	OFDEN4	OFDEN3	OFDEN2	OFDEN1	OFDEN0
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	リセット後	0	0	0	0	0	0	0	0
	機能	0x00: 周波数検知動作停止 0xE4: 周波数検知動作許可 その他: Reserved (注 1)							

注 1) OFDCR2 への書き込みデータは "0x00" と "0xE4" のみ有効です。"0x00" と "0xE4" 以外のデータを書こうとした場合は無効となります (レジスタは変化しません)。

注 2) OFDCR1 に書き込み禁止コード "0x06" を書き込むと、OFDCR2 への書き込みは禁止となり、OFDCR2 に書こうとした全てのデータは無視されます。なお、書き込み禁止の場合でも OFDCR2 の読み出しは可能です。

注 3) OFDCR2 は外部リセット (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。

検知周波数下限値設定レジスタ(PLL OFF 時)

OFDMNPLLOFF (0x4004_0808)	31 - 9							8		
	Bit Symbol	-							OFDMNPLLOFF	
	Read/Write	R							R/W	
	リセット後	0							0	
		7	6	5	4	3	2	1	0	
	Bit Symbol	OFDMNPLLOFF								
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	リセット後	0	0	0	1	1	0	1	0	

検知周波数下限値設定レジスタ(PLL ON 時)

OFDMNPLLON (0x4004_080C)	31 - 9							8		
	Bit Symbol	-							OFDMNPLLON	
	Read/Write	R							R/W	
	リセット後	0							0	
		7	6	5	4	3	2	1	0	
	Bit Symbol	OFDMNPLLON								
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	リセット後	1	1	0	1	0	0	0	1	

検知周波数上限値設定レジスタ(PLL OFF 時)

OFDMXPLLOFF (0x4004_0810)	31 - 9							8		
	Bit Symbol	-							OFDMXPLLOFF	
	Read/Write	R							R/W	
	リセット後	0							0	
		7	6	5	4	3	2	1	0	
	Bit Symbol	OFDMXPLLOFF								
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	リセット後	0	0	1	0	1	0	0	0	

検知周波数上限値設定レジスタ(PLL ON 時)

OFDMXPLLON (0x4004_0814)	31 - 9							8		
	Bit Symbol	-							OFDMXPLLON	
	Read/Write	R							R/W	
	リセット後	0							1	
		7	6	5	4	3	2	1	0	
	Bit Symbol	OFDMXPLLON								
	Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	リセット後	0	0	1	1	1	0	0	1	

- 注 1) 検知周波数設定レジスタの初期値は暫定の値です。  
(OFDCR2 = "0xE4") またはレジスタへの書き込み禁止状態 (OFDCR1 = "0x06") のときは書き換えできません。
- 注 2) OFDMNPLLOFF, OFDMNPLLON, OFDMXPLLOFF 及び OFDMXPLLON レジスタは周波数検知動作が許可状態 (OFDCR2 = "0xE4") またはレジスタへの書き込み禁止状態 (OFDCR1 = "0x06") のときは書き換えできません。
- 注 3) OFDMN PLLOFF, OFDMNPLLON, OFDMXPLLOFF 及び OFDMXPLLON レジスタへの書き込みは OFDCR1 への "0x06" 書き込みでプロテクトされますが、読み出しについては OFDCR1 の設定にかかわらず可能です。
- 注 4) OFDMNPLLOFF と OFDMXPLLOFF は OFDMNPLLOFF < OFDMXPLLOFF の条件下で、使用するクロック周波数に合わせて適切な値を設定してください。設定値は、「17.3.2 検知周波数の設定」参照してください。
- 注 5) OFDMNPLLON と OFDMXPLLON は OFDMNPLLON < OFDMXPLLON の条件下で、使用するクロックを 8 番倍した周波数に合わせて適切な値を設定してください。設定値は、「17.3.2 検知周波数の設定」を参照してください。
- 注 6) OFDMNPLLOFF, OFDMNPLLON, OFDMXPLLOFF 及び OFDMXPLLON は外部リセット (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。
- 注 7) OFDMNPLLOFF/OFDMXPLLOFF と OFDMNPLLON/OFDMXPLLON は PLLON 設定により自動的に切り替わりません。

## 17.3 機能

### 17.3.1 周波数検知回路の動作制御

OFDCR1 に書き込み許可コード "0xF9" が設定された状態で OFDCR2 に "0xE4" を書き込むと周波数検知回路はイネーブルとなり、OFDCR1 に "0xF9" が設定された状態で OFDCR2 に "0x00" を書き込むと周波数検知回路はディセーブルとなります。

外部リセット入力 (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 OFD の制御レジスタは初期化されます

上記リセットにより OFDCR1 は "0x06" に初期化され、OFDCR2 は "0x00" に初期化されますので、リセット解除後は周波数検知動作停止、レジスタ書き込み禁止状態となります。なお、OFDCR1 の設定に関係なく OFDCR2 の読み出しは可能です。

注) OFDCR2 を設定した場合、誤書き込み防止のため OFDCR1 に書き込み禁止コード "0x06" を設定してください。

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、周波数検知回路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、周波数検知回路は 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。

周波数検知回路は NORMAL モードと IDLE モードのときのみ使用可能です。詳細は表 17-1 を参照してください。

表 17-1 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、RESET、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 (RESET 端子への "L" 入力) パワーオンリセット VLTD リセット	停止	-

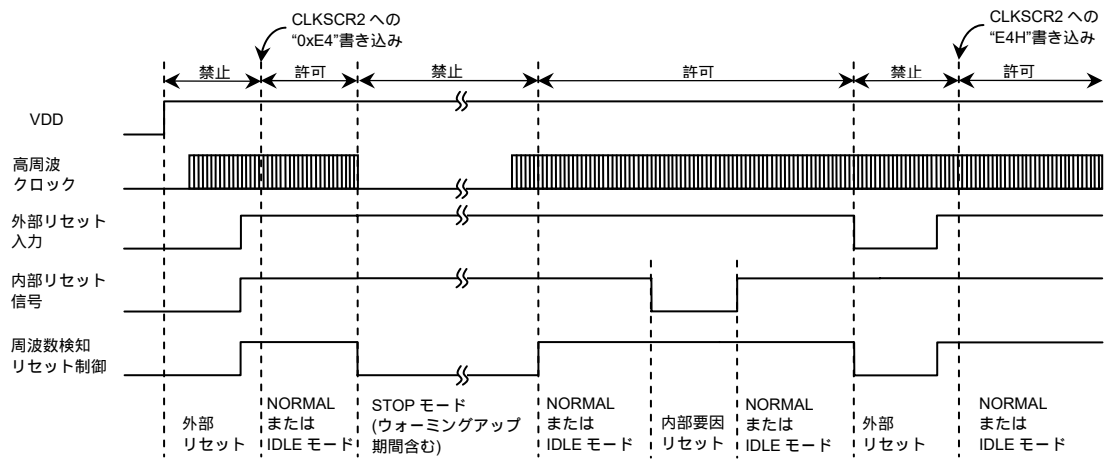


図 17-3 各モードにおける周波数検知回路動作

### 17.3.2 検知周波数の設定

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。基準クロックの周波数は 9.5MHz で、10%の誤差を持ちます。

(1)	検出対象クロック	最大
(2)		最小
(3)	基準クロック	最大(10.5MHz)
(4)		最小(8.5MHz)

算出方法は以下の通りです。

検知周波数上限値 = (1) ÷ (4) × 32 (小数点以下切り捨て)

検知周波数下限値 = (2) ÷ (3) × 32 (小数点以下切り上げ)

### 17.3.3 周波数検知リセット

OFDMNPLLON/OFDMNPLLOFF で設定された値より低い周波数、または OFDMXPLLON/OFDMXPLLOFF で設定された値より高い周波数を検出した場合、TMPM376FD は全ての入出力ポートを初期化する周波数検知リセットを発生します。

検出対象クロックが異常となった場合

周波数検知回路(OFD)は、基準クロックを元に作られる一定期間(Tofd)にカウントされる検出対象クロックの数が検知周波数上限値と下限値レジスタの設定値から外れている場合に異常と判断し、周波数検知リセットを発生します。

#### a. 外部高速発振クロック(fosc1)の異常

fosc1 の異常により周波数検知リセットが発生した場合、内部回路の初期化後、内部高速発振クロック(fosc2)をシステムクロックとして起動し、OFD は fosc2 を検出対象クロックとして検知動作が継続します。

#### b. 内部高速発振クロック(fosc2)の異常

fosc2 の異常により周波数検知リセットが発生した場合、内部回路の初期化後も fosc2 を検出対象クロックとして検知動作が継続しますので Tofd の間 fosc2 が正常周波数に回復するまで周波数検知リセットを発生し続けます。





## 第 18 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) 本製品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。

### 18.1 構成

図 18-1 にウォッチドッグタイマのブロック図を示します。

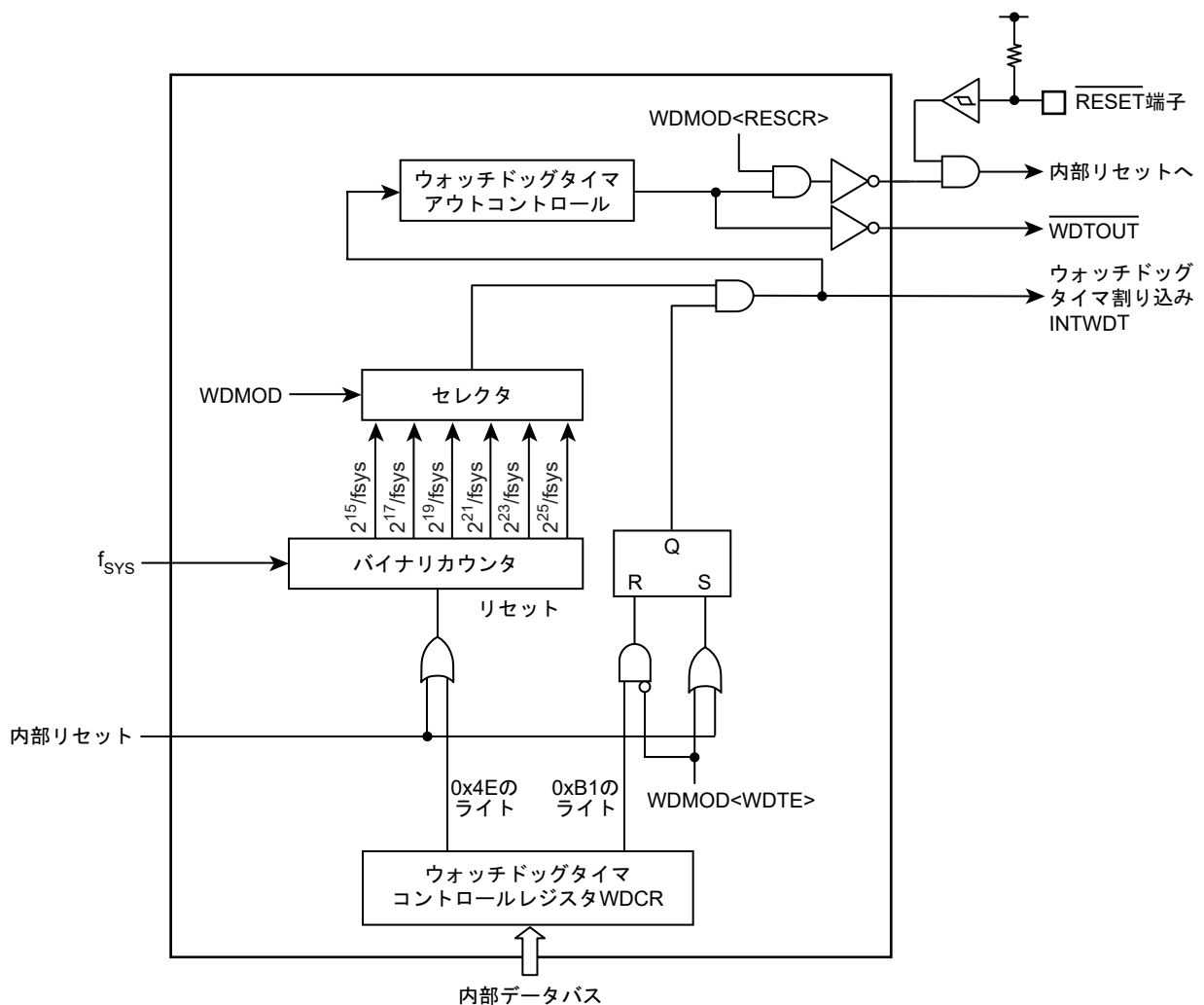


図 18-1 ウォッチドッグタイマのブロック図



表 18-1 ウォッチドッグタイマの検出時間 (fc = 80MHz)

クロックギア値 CGSYSR<GEAR[2:0]>	WDMOD<WDTP[2:0]>					
	000	001	010	011	100	101
000 (fc)	0.41 ms	1.64 ms	6.55 ms	26.21 ms	104.86 ms	419.43 ms
100 (fc/2)	0.82 ms	3.28 ms	13.11 ms	52.43 ms	209.72 ms	838.86 ms
101 (fc/4)	1.64 ms	6.55 ms	26.21 ms	104.86 ms	419.43 ms	1.68 s
110 (fc/8)	3.28 ms	13.11 ms	52.43 ms	209.72 ms	838.86 ms	3.36 s
111 (fc/16)	6.55 ms	26.21 ms	104.86 ms	419.43 ms	1.68 s	6.71 s

## 18.2.2 WDCR(ウォッチドッグタイ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	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

## 18.3 動作説明

### 18.3.1 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、ソフトウェア (命令) でウォッチドッグタイマ用のバイナリカウンタを INTWDT 割り込みが発生する前にクリアするように設定しておきます。クリアが行われなかった場合、INTWDT によってマスク不能割り込みが発生しますので、CPU は暴走を検知し、暴走対策プログラムにより正常な状態に戻すことができます。

また、ウォッチドッグタイマアウト端子を外部周辺装置のリセットなどへ接続することにより、CPU の誤動作 (暴走) に対処することができます。

**注) 本製品にはウォッチドッグタイマアウト端子 ( $\overline{\text{WDTOUT}}$ ) はありません。**

### 18.3.2 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。

## 18.4 暴走検出時の動作

### 18.4.1 INTWDT 割り込み発生の場合

図 18-2 に INTWDT 割り込み発生 (WDMOD<RESCR>="0") の場合の動作を示します。

バイナリカウンタのオーバーフローにより INTWDT 割り込みが発生します。INTWDT 割り込みはマスク不能割り込み (NMI) の要因であるため、CPU はマスク不能割り込みを検出し処理を行います。

マスク不能割り込み要因は複数あり、CGNMIFLG レジスタでマスク不能割り込み要因を識別できます。INTWDT 割り込みの場合、CGNMIFLG<NMIFLG0> がセットされます。

INTWDT 割り込み発生と同時にウォッチドッグタイマアウト ( $\overline{\text{WDTOUT}}$ ) より "Low" を出力します。 $\overline{\text{WDTOUT}}$  は、ウォッチドッグタイマのクリア (WDCR レジスタにクリアコード 0x4E をライト) により "High" に戻ります。

注) 本製品にはウォッチドッグタイマアウトの外部出力端子はありません。

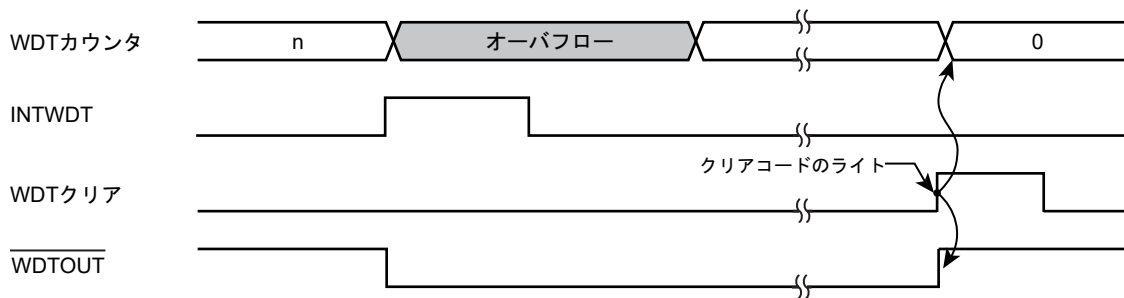


図 18-2 INTWDT 割り込み発生

### 18.4.2 内部リセット発生の場合

図 18-3 に内部リセット発生 (WDMOD<RESCR>="1") の場合の動作を示します。

バイナリカウンタのオーバーフローによりマイコンをリセットします。この場合、32 ステートの期間、リセットを行います。クロックの設定も初期化され、入力クロック  $f_{SYS}$  と高速発振器のクロック  $f_{OSC}$  の関係は、 $f_{SYS} = f_{OSC}$  となります。

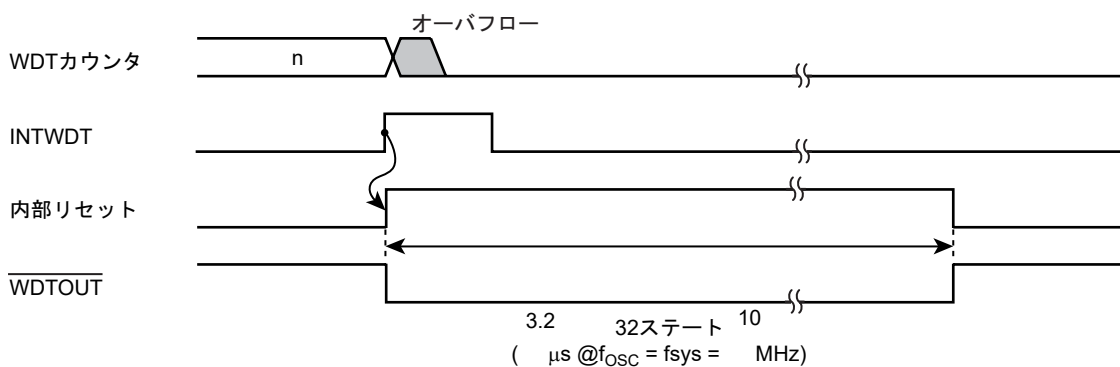


図 18-3 内部リセット発生

## 18.5 コントロールレジスタ

ウォッチドッグタイマ (WDT) は、2つのコントロールレジスタ (WDMOD, WDCR) によって制御されています。

### 18.5.1 ウォッチドッグタイマモードレジスタ (WDMOD)

#### 1. ウォッチドッグタイマ検出時間の設定 <WDTP[2:0]>

ウォッチドッグタイマ検出時間を設定します。リセット時 WDMOD<WDTP[2:0]> = "000" に初期化されます。

#### 2. ウォッチドッグタイマのイネーブル/ディセーブル制御 <WDTE>

リセット時 WDMOD<WDTE> = "1" に初期化されますので、ウォッチドッグタイマはイネーブルになっています。

暴走による誤書き込みを防止するため、ディセーブルにするには、このビットを "0" にした後で、WDCR にディセーブルコード (0xB1) を書き込む必要があります。

ディセーブル状態からイネーブル状態に戻す場合は、WDMOD<WDTE> を "1" に設定します。

#### 3. ウォッチドッグタイマアウトのリセット接続 <RESCR>

WDTOUTを内部リセットとして使用するか割り込みとして使用するかを設定するレジスタです。リセット時 WDMOD<RESCR> = "1" に初期化されますので、バイナリカウンタのオーバフローにより内部リセットが発生します。

### 18.5.2 ウォッチドッグタイマコントロールレジスタ (WDCR)

ウォッチドッグタイマ機能のディセーブルおよびバイナリカウンタのクリアを制御するレジスタです。

### 18.5.3 設定例

#### 18.5.3.1 ディセーブル制御

WDMOD<WDTE> に "0" を設定したあと、WDCR レジスタにディセーブルコード (0xB1) を書き込むとウォッチドッグタイマはディセーブルになり、バイナリカウンタはクリアされます。

		7	6	5	4	3	2	1	0	
WDMOD	←	0	-	-	-	-	-	-	-	<WDTE> に "0" を設定します。
WDCR	←	1	0	1	1	0	0	0	1	ディセーブルコード (0xB1) を書き込みます。

#### 18.5.3.2 イネーブル制御

WDMOD<WDTE> に "1" を設定します。

		7	6	5	4	3	2	1	0	
WDMOD	←	1	-	-	-	-	-	-	-	<WDTE> に "1" を設定します。

#### 18.5.3.3 ウォッチドッグタイマのクリア制御

WDCR レジスタにクリアコード (0x4E) を書き込むと、バイナリカウンタはクリアされ、再カウントします。

		7	6	5	4	3	2	1	0	
WDCR	←	0	1	0	0	1	1	1	0	クリアコード (0x4E) を書き込みます。

#### 18.5.3.4 ウォッチドッグタイマ検出時間の設定

検出時間を  $2^{21}/f_{SYS}$  に設定する場合、WDMOD<WDTP[2:0]> に "011" を設定します。

		7	6	5	4	3	2	1	0	
WDMOD	←	1	0	1	1	-	-	-	-	



## 第 19 章 Flash 動作説明

Flash 機能について、ハードウェアの構成およびその動作を説明します。

### 19.1 フラッシュメモリ

#### 19.1.1 特長

##### 1. メモリ容量

TMPM376FDDFG/FDFG はフラッシュメモリを搭載しています。メモリ容量と構成は下記の表のとおりです。

各ブロック個別に書き込みを行うことができます。CPU から内蔵フラッシュメモリをアクセスする場合、データバス幅は 32 ビットとなります。

##### 2. 書き込み / 消去時間

書き込みはページ単位で行います。1 ページは 128 ワードです。

1 ページあたりの書き込み時間は、ワード数にかかわらず 1.25ms (Typ.) です。

消去時間は 1 ブロックあたり 0.1 sec (Typ.) です。

1 チップあたりの書き込み時間、消去時間は以下のようになります。

製品	メモリ容量	ブロック構成				ワード数	書き込み時間	消去時間
		128 KB	64 KB	32 KB	16 KB			
TMPM376FDDFG / FDFG	512 KB	3	1	2	0	128	1.28 sec	0.4 sec

注) 上記の値は理論時間を表しており、データ転送時間などは含まれていません。チップ当たりの時間はユーザーの書き替え方法により異なります。

##### 3. プログラミング方法

ユーザーのボード上で書き替えが可能なオンボードプログラミングモードには以下の 2 種類のモードがあります。

###### a. ユーザーブートモード

ユーザー独自の書き替え方法をサポート

###### b. シングルブートモード

シリアル転送 (当社オリジナル) での書き替え方法をサポート

#### 4. 書き替え方式

本デバイス内蔵のフラッシュメモリは、一部の機能を除き JEDEC 標準機能に準拠しています。このため、外部メモリとしてフラッシュメモリをご使用になられている場合でも、本 LSI への移行が容易です。また、フラッシュメモリ内に書き込み、チップ消去など自動で行う回路を内蔵していますので、書き込み、消去動作自身に係わる複雑なフローをユーザーがプログラムで組む必要がありません。

JEDEC 準拠の機能	変更, 追加, 削除した機能
<ul style="list-style-type: none"> <li>• 自動プログラム</li> <li>• 自動チップ消去</li> <li>• 自動ブロック消去</li> <li>• データボーリング / トグルビット</li> </ul>	<p>&lt;変更&gt; ブロック単位でのライト / 消去プロテクト (ソフトウェアプロテクトのみサポート)</p> <p>&lt;削除&gt; 消去レジューム / サスペンド機能</p>

#### 5. プロテクト / セキュリティ機能

本デバイスでは、ライターでのフラッシュデータの読み出しを禁止する、セキュリティ機能を追加しています。一方、書き替え禁止を設定するライト / 消去プロテクトは、コマンド (ソフトウェア) による対応のみで 12 V 電圧を印加して設定する方式 (ハードウェア) には対応できません。プロテクト、セキュリティ機能の詳細については、「プロテクト / セキュリティ機能」の章を参照してください。

注) パスワードが消去データ (0xFF) の場合、容易にパスワードの照合が可能になり、セキュリティの確保が難しくなります。シングルブートモードを使用しない場合も固有の値を置くことを推奨します。

19.1.2 フラッシュ部ブロック図

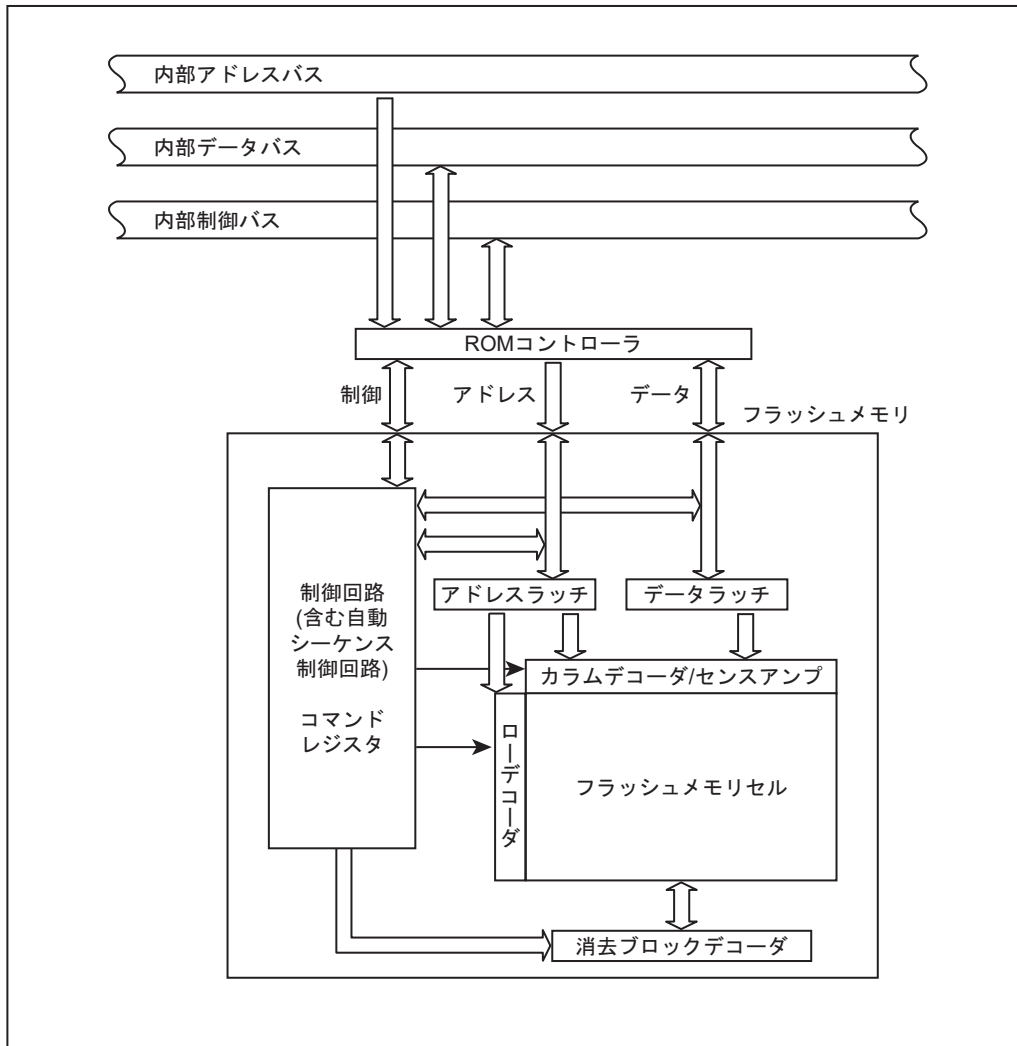


図 19-1 フラッシュ部ブロック図

## 19.2 動作モード

本デバイスは内蔵フラッシュメモリを使用しない場合を含めて、3通りの動作状態（モード）が存在します。

表 19-1 動作モード説明

動作モード名	動作の内容
シングルチップモード	リセット解除後、内蔵のフラッシュメモリから起動します。
ノーマルモード	本動作モードの中で、ユーザーのアプリケーションプログラムを実行するモードと、ユーザーのセット上でフラッシュメモリの書き替えを実行するモードとに分けて定義します。前者を「ノーマルモード」、後者を「ユーザーブートモード」と呼びます。
ユーザーブートモード	この両者の切り替えはユーザーが独自に設定できます。 例えばポート A0 が "1" のときノーマルモード、"0" のときにユーザーブートモードというように自由に設計することが可能です。 ユーザーはアプリケーションプログラムの一部に切り替えを判定するためのルーチンを準備してください。
シングルブートモード	リセット解除後、内蔵する Boot ROM (Mask ROM) から起動します。Boot ROM には、本デバイスのシリアルポートを経由してユーザーのセット上で書き替えを行うことができるアルゴリズムがプログラムされています。シリアルポートにより外部ホストと接続し、規定されたプロトコルでデータの転送を行うことで内蔵フラッシュの書き替えが実行できます。

上記表 19-1 でプログラムが可能なフラッシュメモリの動作モードはユーザーブートモード、シングルブートモードの 2 つです。ユーザーのセット上で内蔵フラッシュメモリの書き替えが可能なモードは、ユーザーブートモードとシングルブートモードで、この 2 つをオンボードプログラミングモードと定義します。

シングルチップ、シングルブートの各動作モードは、リセット状態で  $\overline{\text{BOOT}}$  (PF0) 端子のレベルを外部で設定することにより決定されます。

表 19-2 動作モード設定表

動作モード	端子	
	$\overline{\text{RESET}}$	$\overline{\text{BOOT}}$ (PF0)
シングルチップモード	0 → 1	1
シングルブートモード	0 → 1	0

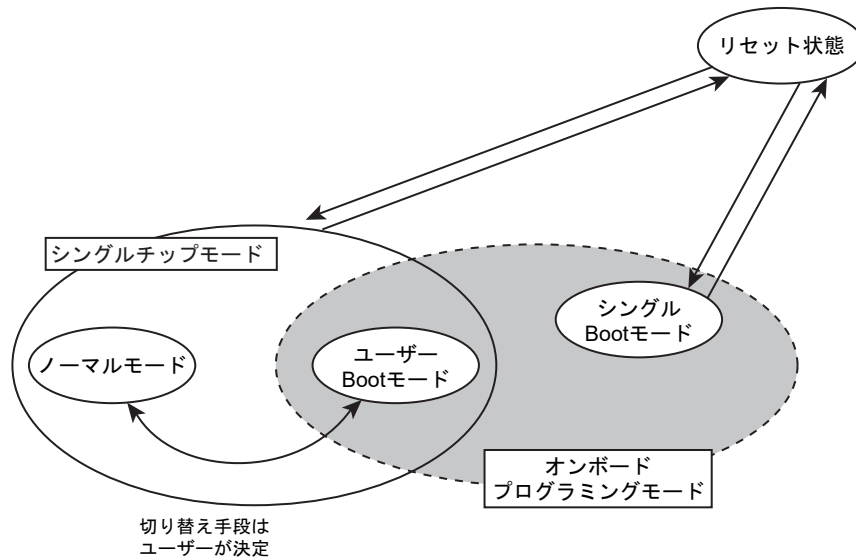


図 19-2 モード遷移図

### 19.2.1 リセット動作

本デバイスにリセットをかけるには、電源電圧が動作電圧範囲内で、かつ内部発振器の発振が安定した状態で、少なくとも 12 システムクロック間 (80MHz 動作で  $0.15 \mu\text{s}$  (リセット後は、クロックギア 1/1 モード)) RESET 入力を "0" にしてください。

- 注 1) 電源投入後は、電源電圧が安定した状態から  $700 \mu\text{s}$  以上経過してからリセット解除させてください。
- 注 2) 内蔵フラッシュの自動プログラム / 消去動作中にハードウェアリセットを行う場合は、システムクロックによらず  $0.5 \mu\text{s}$  以上のリセット期間が必要となります。またこの場合、リセット解除後読み出しが可能になるまで 2 ms 程度の時間がかかります。

### 19.2.2 ユーザーブートモード (シングルチップモード)

ユーザーブートモードは、ユーザー独自のフラッシュメモリ書き替えルーチンを使う方法です。ユーザーアプリケーション上で用意されているフラッシュメモリ書き替え用のプログラムで用いる、データ転送バスがシリアル I/O と異なる場合に使用します。動作はシングルチップモード上で行います。このため、シングルチップモードにおいて通常のユーザーアプリケーションプログラムが動作しているノーマルモードから、フラッシュを書き替えるためのユーザーブートモードに移行する必要があります。従って、条件判定を行うプログラムをユーザーアプリケーションの中で、リセット処理プログラムの中に組み込んでください。

このモード切り替えの条件設定は、本デバイスの I/O を使用してユーザーのシステムセット条件に合わせて独自に構築してください。また、ユーザーブートモード移行後に使用するユーザー独自のフラッシュメモリ書き替えルーチンも同様にユーザーアプリケーションの中にあらかじめ組み込んでおき、ユーザーブートモード移行後にこれらのルーチンを使用して書き替えを行ってください。なお、内蔵フラッシュメモリは消去 / 書き込み動作モード中はフラッシュのデータを読み出せません。このため、書き替えルーチンはフラッシュメモリエリア外に格納して実行させる必要があります。また、シングルチップモード (通常動作モード) 中に誤ってフラッシュの内容を書き替えないように、書き替え処理が完了した後、必要なブロックにライト / 消去プロテクトをかけておくことを推奨します。あわせて、ユーザーブートモード中は、ノンマスカブルを含めたすべての割り込み発生を禁止してください。

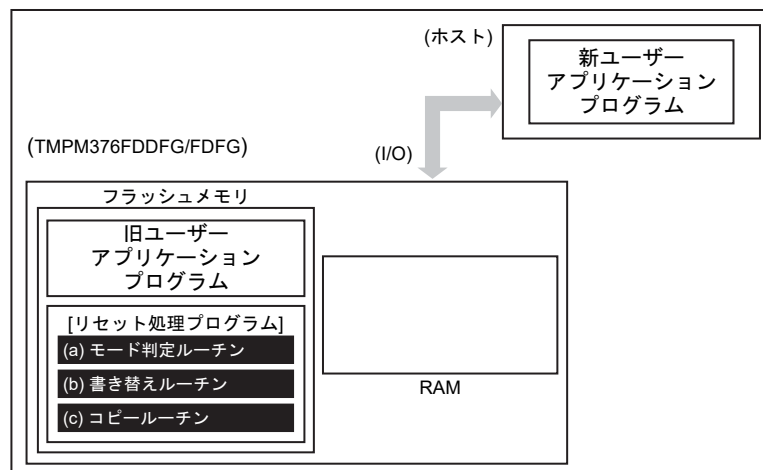
書き換えルーチンを内蔵フラッシュメモリに置く場合と、外部から転送する場合の 2 ケースを例に、以下 (1-A), (1-B) にその手順を説明します。フラッシュメモリへの書き込み / 消去 方法の詳細は、「19.3 オンボードプログラミングでのフラッシュメモリ書き込み / 消去」を参照してください。

### 19.2.2.1 (1-A) 書き換えルーチンをフラッシュメモリに内蔵する場合の手順例

#### (1) Step-1

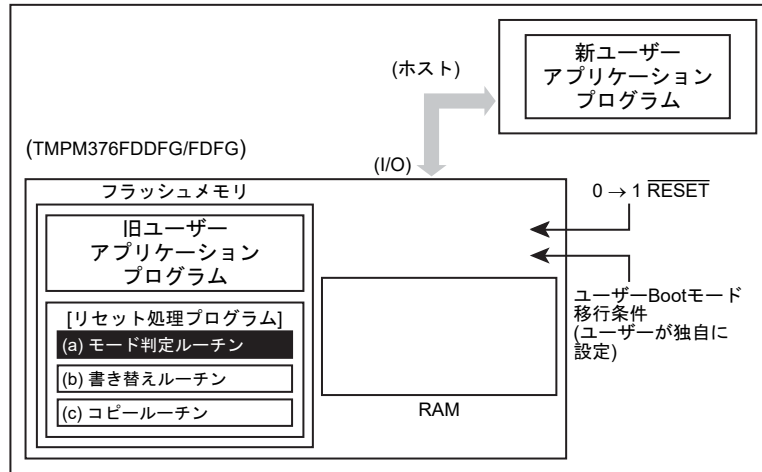
ユーザーは、あらかじめどのような条件（例えば端子状態）に設定されたらユーザーブートモードに移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合った回路の設計、プログラムの作成を行います。ユーザーは本デバイスをボードに組み込む前に、あらかじめフラッシュメモリ上の任意のブロックにライタなどを使用して以下に示す 3 つのプログラムを書き込んでおきます。

- (a) モード判定ルーチン：           書き換え動作に移るためのプログラム
- (b) フラッシュ書き換えルーチン：   書き換えデータを外部から取り込み、フラッシュメモリを書き替えるためのプログラム
- (c) コピールーチン：               上記 (b) を内蔵 RAM または外部メモリにコピーするためのプログラム



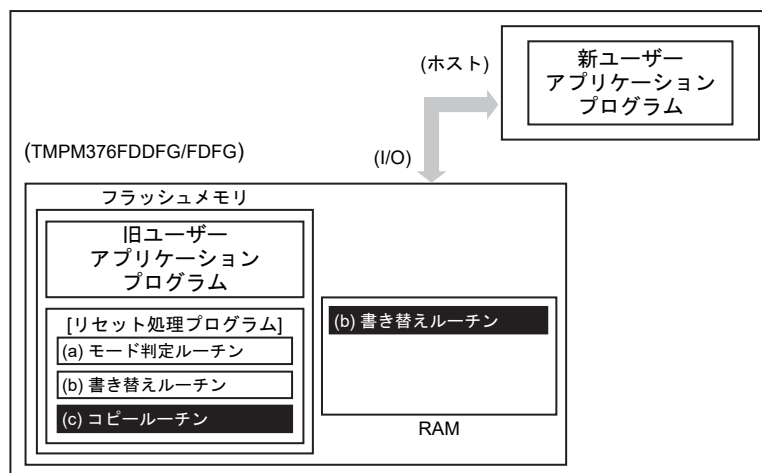
(2) Step-2

以下リセット処理プログラム内にこれらのルーチンを組み込んだ場合について説明します。まず、リセット解除後のリセット処理プログラムにおいてユーザーブートモードへの移行を判定します。このとき、移行条件が整っていれば、プログラムは書き替えのためのユーザーブートモードに移ります。(ユーザーブートモードに移行した場合は、これ以降割り込みを使わないでください。)



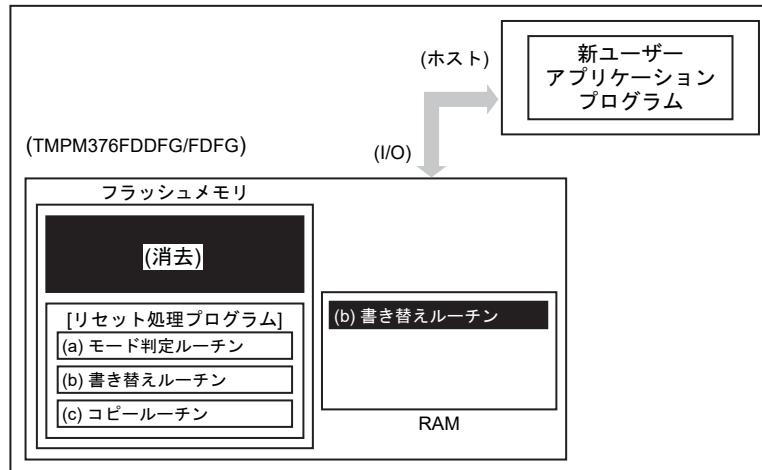
(3) Step-3

ユーザーブートモードに移ると、(c) コピールーチンを使用して、(b) 書き替えルーチンを内部 RAM にコピーします。



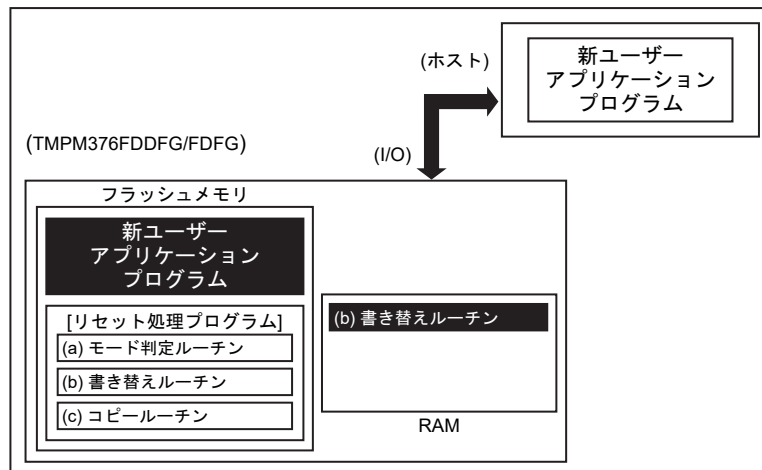
(4) Step-4

RAM 上の書き替えルーチンへジャンプし、旧ユーザープログラムエリアのライト / 消去プロテクトを解除して、消去 (ブロック単位) を行います。



(5) Step-5

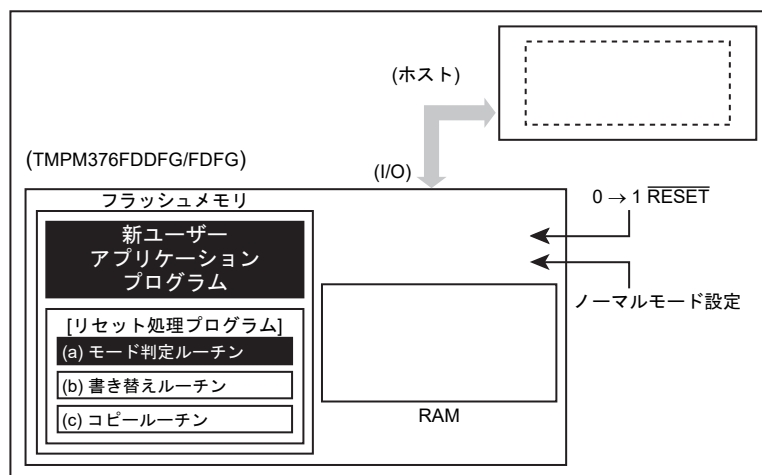
さらに、RAM 上の書き替えルーチンを実行して、転送元 (ホスト) より新ユーザーアプリケーションプログラムのデータをロードし、フラッシュメモリの消去したエリアに書き込みを行います。書き込みが完了したら、ユーザープログラムエリアのライト / 消去プロテクトをオンにします。





## (6) Step-6

RESET 入力端子を "0" にしてリセットを行い、設定条件をノーマルモードの設定にします。リセット解除後、新ユーザーアプリケーションプログラムで動作を開始します。



### 19.2.2.2 (1-B) 書き替えルーチンを外部から転送する手順例

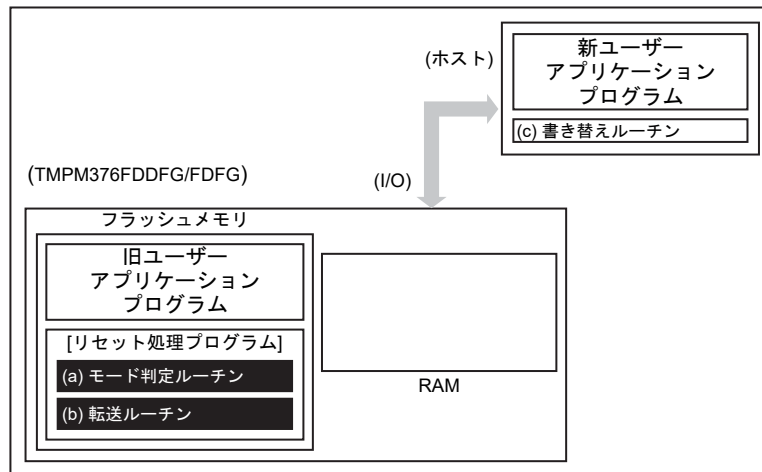
#### (1) Step-1

ユーザーは、あらかじめどのような条件（例えば端子状態）に設定されたらユーザーブートモードに移行するか、どの I/O バスを使用してデータ転送を行うかを決め、それに合った回路の設計、プログラムの作成を行います。ユーザーは本デバイスをボードに組み込む前に、あらかじめフラッシュメモリ上の任意のブロックにライタなどを使用して以下に示す 2 つのプログラムを書き込んでおきます。

- (a) モード判定ルーチン： 書き替え動作に移るためのプログラム
- (b) 転送ルーチン： 書き替えプログラムを外部から取り込むためのプログラム

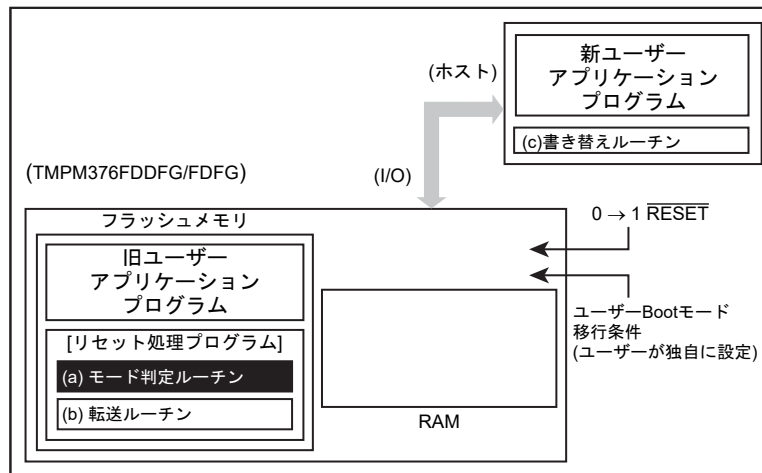
また、下記に示すプログラムはホスト上に用意します。

- (c) 書き替えルーチン： 書き替えを行うためのプログラム



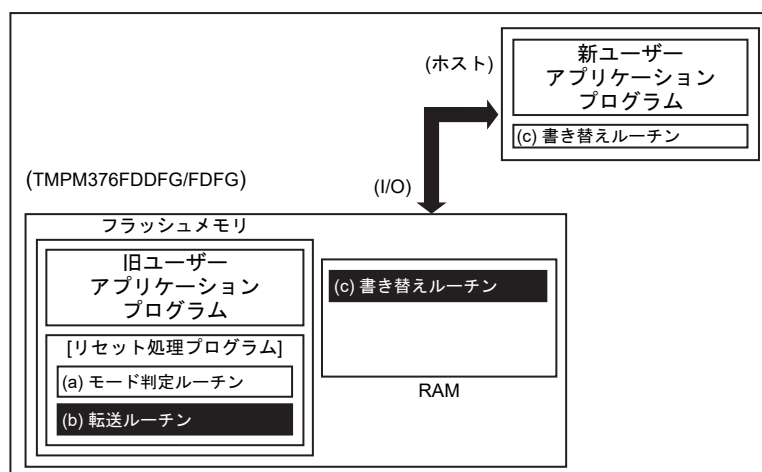
(2) Step-2

以下、リセット処理プログラム内にこれらのルーチンを組み込んだ場合について説明します。まず、リセット解除後のリセット処理プログラムにおいてユーザーブートモードへの移行を判定します。このとき、移行条件が整っていれば、プログラムは書き替えのためのユーザーブートモードに移ります。(ユーザーブートモードに移りし場合は、これ以降割り込みを使わないでください。)



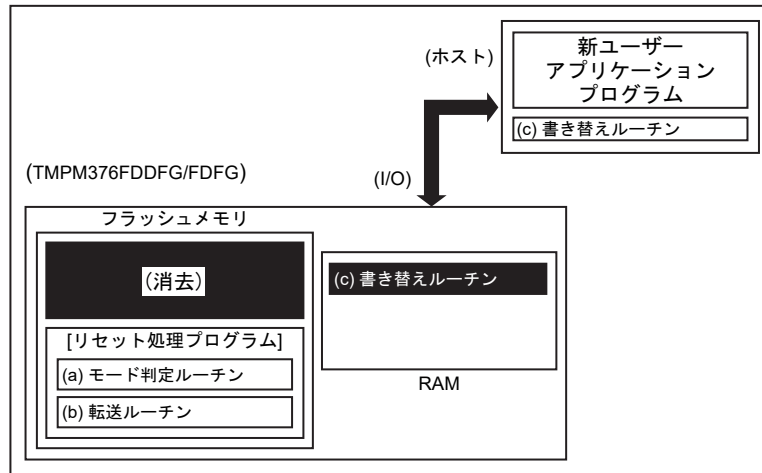
(3) Step-3

ユーザーブートモードに移ると、(b) 転送ルーチンを使用して、転送元 (ホスト) より (c) 書き替えルーチンを内部 RAM にロードします。



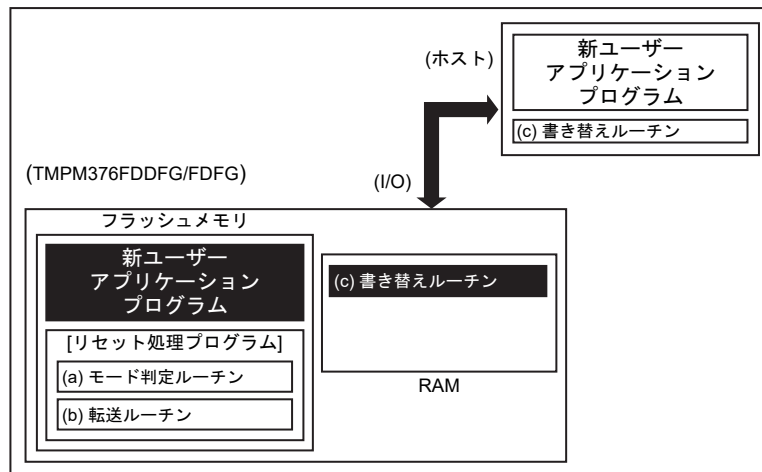
(4) Step-4

RAM 上の書き替えルーチンへジャンプし、旧ユーザープログラムエリアのライト / 消去プロテクトを解除して、消去 (ブロック単位) を行います。



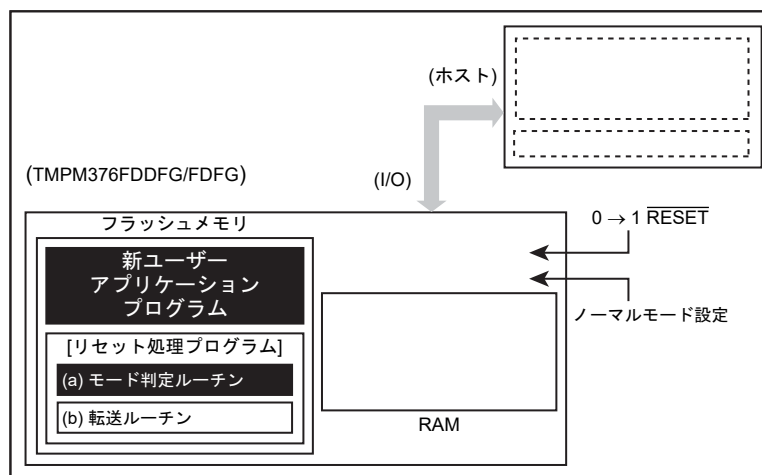
(5) Step-5

さらに、RAM 上の (c) 書き替えルーチンを実行して、転送元 (ホスト) より新ユーザーアプリケーションプログラムのデータをロードし、消去したエリアに書き込みを行います。書き込みが完了したら、ユーザープログラムエリアのライト / 消去プロテクトをオンにします。



## (6) Step-6

RESET 入力端子を "0" にしてリセットを行い、設定条件をノーマルモードに設定します。  
リセット解除後、新ユーザーアプリケーションプログラムで動作を開始します。



### 19.2.3 シングルブートモード

内蔵ブート ROM (マスク ROM) を起動して、ブート ROM のプログラムを利用してフラッシュメモリを書き替える方法です。このモードでは、内蔵ブート ROM が割り込みベクタテーブルを含む領域にマッピングされ、ブート ROM プログラムが実行されます。また、フラッシュメモリはブート ROM 領域とは別のアドレス空間にマッピングされます。

ブートモードでは、コマンドおよびデータをシリアル転送してフラッシュメモリの書き替えを行います。本デバイスの SIO (SIO0) と外部ホストを接続し、外部ホスト側から本デバイスの内蔵 RAM へ書き替えプログラムをコピーし、RAM 上の書き替えルーチンを実行してフラッシュメモリの書き替えを行います。書き替えルーチンは、ホスト側からコマンドおよび書き替えデータを送出することにより実行します。ホスト側との通信の詳細は後述のプロトコルに従ってください。

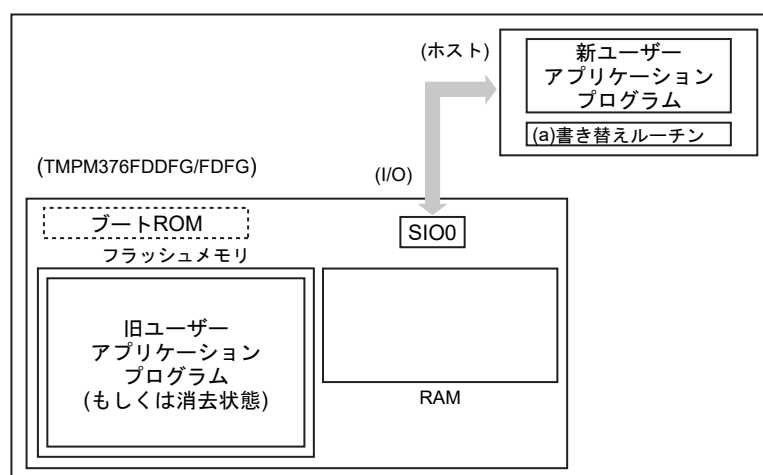
RAM へのプログラム転送は、ユーザーの ROM データ、セキュリティ確保のため、実行に先立ちユーザーパスワードの照合を行います。パスワードが一致しない場合は、RAM 転送そのものが実行されません。なお、シングルブートモードでもユーザーブートモードと同様、割り込み禁止状態で行います。シングルブートモード時、ブート ROM プログラムは NORMAL モードで動作します。

シングルチップモード (通常動作モード) 中に誤ってフラッシュメモリの内容を書き替えないよう、書き替え処理が完了したら必要なブロックにライト / 消去プロテクトをかけておくことを推奨します。

#### 19.2.3.1 (2-A) 内蔵ブート ROM の書き替えアルゴリズムを利用する場合

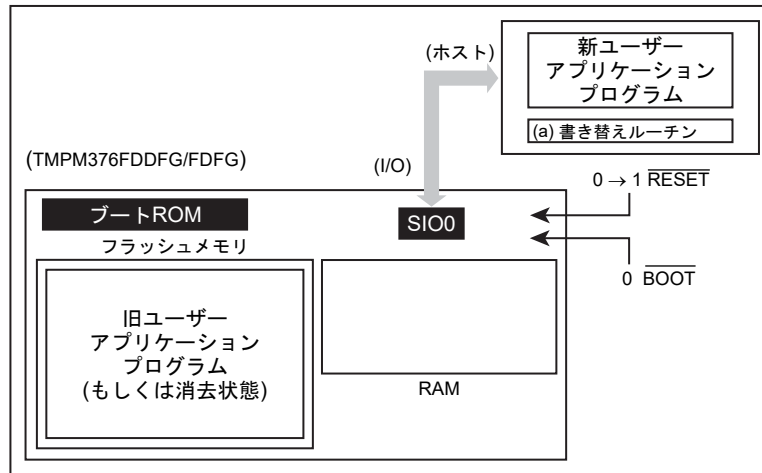
##### (1) Step-1

フラッシュメモリの状態は旧バージョンのユーザープログラムが書かれた状態でも、消去されている状態でも構いません。書き替えルーチン、書き替えデータなどの転送は SIO (SIO0) を経由して行いますので、ボード上で本デバイスの SIO (SIO0) と外部ホストとをつなげます。書き替えを行うための (a) 書き替えルーチンはホスト上に用意します。



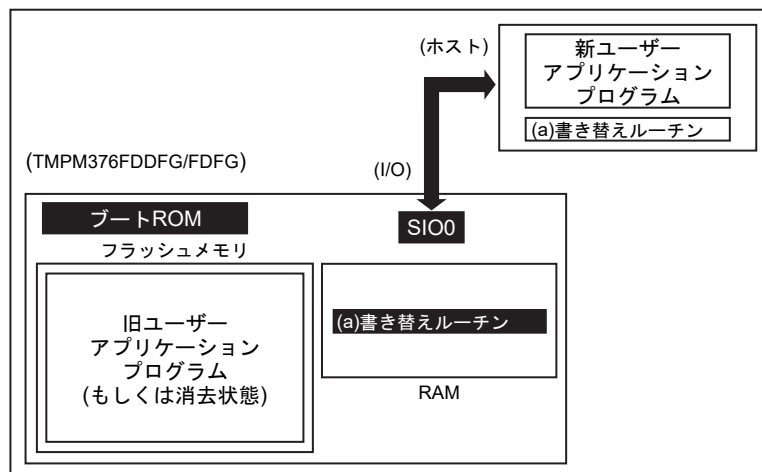
(2) Step-2

ブートモードの端子条件設定でリセットを解除し、ブート ROM で起動します。ブートモードの手順に従い、SIO0 を経由して転送元 (ホスト) より (a) 書き替えルーチンの転送を行います。最初にユーザーアプリケーションプログラム上に記録されているパスワードとの照合を行います。(フラッシュメモリが消去されている状態でも、消去データ (0xFF) をパスワードとして照合を行います。)



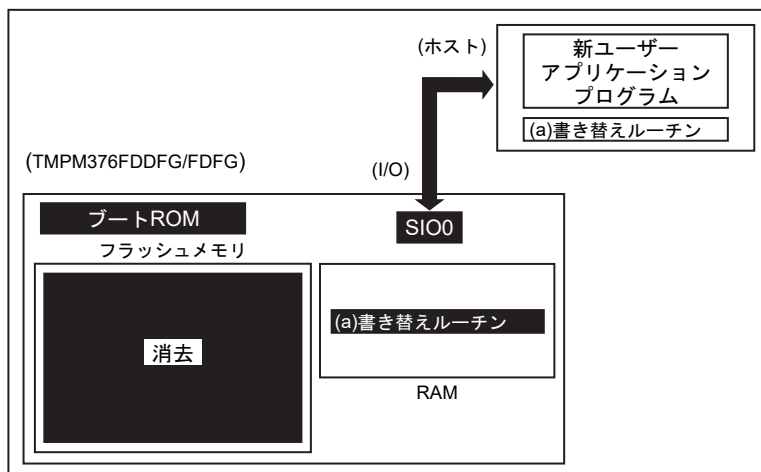
(3) Step-3

パスワードの照合が終了すると、転送元 (ホスト) から (a) 書き替えルーチンを転送します。ブート ROM はそのルーチンを内部 RAM にロードします。ただし、RAM 上のアドレス 0x2000\_0400 から RAM の最終番地の範囲に格納してください。



(4) Step-4

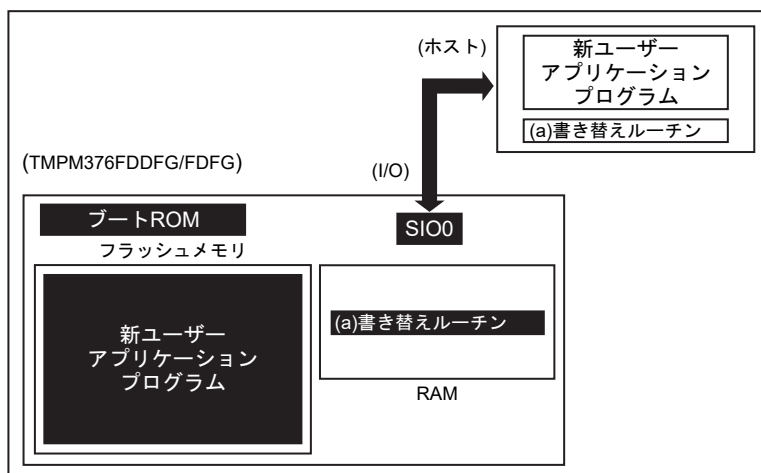
RAM 上の (a) 書き替えルーチンへジャンプし、旧ユーザーアプリケーションプログラムエリアの消去を行います。(ブロック単位もしくは一括)



(5) Step-5

さらに、RAM 上の (a) 書き替えルーチンを実行して、転送元 (ホスト) より新ユーザーアプリケーションプログラムのデータをロードし、フラッシュメモリの消去したエリアに書き込みを行います。書き込みが完了したら、ユーザープログラムエリアのライト / 消去プロテクトをオンにします。

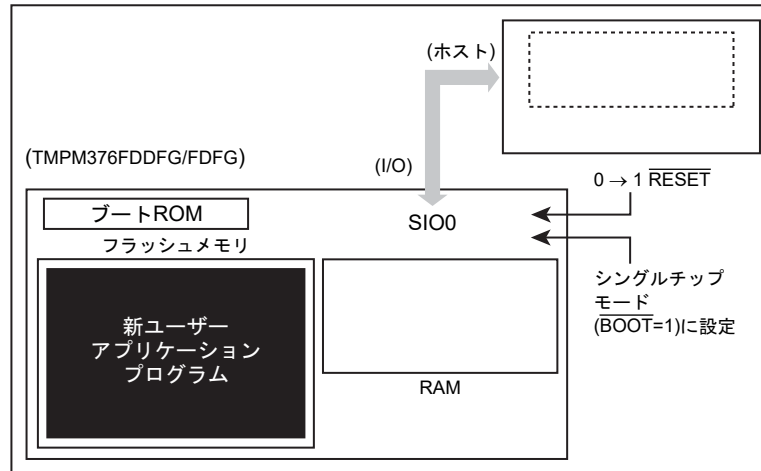
下の例の場合、書き替えルーチンを転送したときと同じホストおよび SIO0 経由で書き替えデータも転送されていますが、RAM 上で動作を開始した以降では、ユーザー独自にデータバスおよび転送元を設定することもできます。方法に応じて、ボードのハードおよび書き替えルーチンを組み立ててください。





## (6) Step-6

書き込みが完了したら、一度ボードの電源を落とし、ホストと接続していたケーブルをはずします。この後、再度電源を入れ直し、シングルチップモード（ノーマルモード）起動し、新しいユーザーアプリケーションプログラムを実行します。



## 19.2.4 モード設定

オンボードプログラミングを実行するためには、本デバイスをシングルブートモードで立ち上げます。シングルブートモードで立ち上がるための設定を以下に示します。

$$\overline{\text{BOOT}} (\text{PF0}) = 0$$

$$\overline{\text{RESET}} = 0 \rightarrow 1$$

$\overline{\text{RESET}}$  入力端子を "0" の状態にして、 $\overline{\text{BOOT}}$  (PF0) 端子をあらかじめ上記条件に設定します。その後リセット解除を行うとシングルブートモードで起動します。

### 19.2.5 メモリマップ

図 19-3 にシングルチップモードとシングルブートモードのメモリマップの比較を示します。図のように、シングルブートモードでは、内蔵フラッシュメモリは 0x3F80\_0000 番地からマッピングされます。また、0x0000\_0000 番地から 0x0000\_0FFF 番地にはブート ROM (マスク ROM) がマッピングされます。

内蔵フラッシュメモリと RAM のマッピングは以下のとおりです。

製品	Flash サイズ	RAM サイズ	Flash アドレス (シングルチップ/シングルブートモード)	RAM アドレス
TMPM376FDDFG/ FDFG	512 KB	32 KB	0x0000_0000 ~ 0x0007_FFFF 0x3F80_0000 ~ 0x3F87_FFFF	0x2000_0000 ~ 0x2000_7FFF

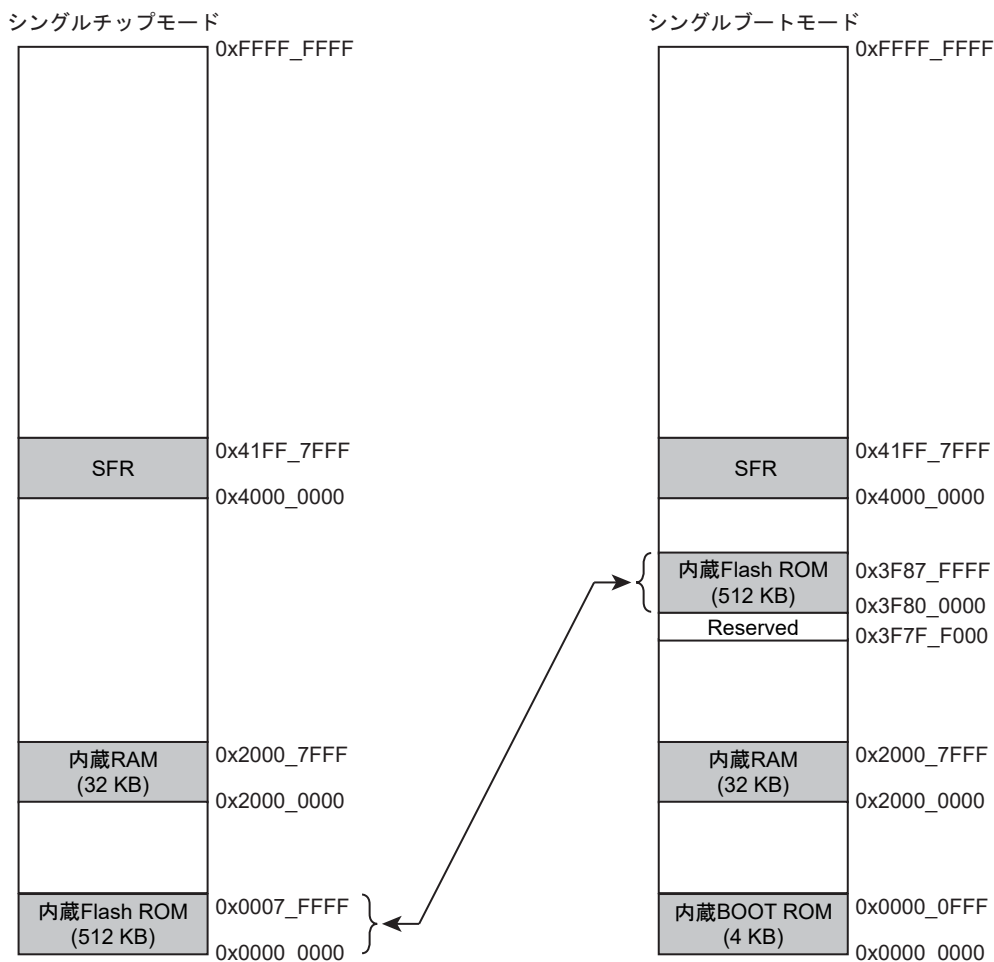


図 19-3 メモリマップの比較 (TMPM376FDDFG/FDFG)

### 19.2.6 インタフェース仕様

シングルブートモードでの SIO 通信フォーマットを以下に示します。シリアル動作のモードは、UART (非同期通信) と I/O インタフェースモード両方に対応しています。オンボードプログラミングを実行するためには、書き込みコントローラ側の通信フォーマットも同様に設定する必要があります。

- UART で通信する場合

通信チャンネル: SIO0

シリアル転送モード: UART (非同期通信) モード, 半二重通信, LSB ファスト

データ長: 8 ビット

パリティビット: なし

STOP ビット: 1 ビット

ボーレート: 任意のボーレート

- I/O インタフェースモードで通信する場合

通信チャンネル: SIO0

シリアル転送モード: I/O インタフェースモード, 全二重通信, LSB ファスト

同期信号 (SCLK0): 入力モード

ハンドシェイク端子: 出力モード PE4

ボーレート: 任意のボーレート

表 19-3 端子の接続

端子		インタフェース	
		UART	I/O インタフェースモード
電源系端子	DVDD5	○	○
	DVSS	○	○
	AVDD5A	○	○
	AVSSA	○	○
	AVDD5B	○	○
	AVSSB	○	○
	VOUT3	○	○
	VOUT15	○	○
	RVDD5	○	○
モード設定端子	$\overline{\text{BOOT}}$ (PF0)	○	○
リセット端子	$\overline{\text{RESET}}$	○	○
通信端子	TXD0 (PE0)	○	○
	RXD0 (PE1)	○	○
	SCLK0 (PE2)	×	○ (入力モード)
	PE4	×	○ (出力モード)

### 19.2.7 データ転送フォーマット

動作コマンド、および各動作モード時のデータ転送フォーマットをそれぞれ表 19-4, 表 19-6 ~ 表 19-7 に示します。「19.2.10 ブートプログラム動作説明」とあわせてお読みください。

表 19-4 動作コマンドデータ

動作コマンドデータ	動作モード
0x10	RAM 転送
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

### 19.2.8 メモリの制約について

シングルブートモードでは、内蔵 RAM, 内蔵 Flash ROM に対して表 19-5 のように制約が付きまます。

表 19-5 シングルブート時のメモリの制約

メモリ	制約内容
内蔵 RAM	0x2000_0000 ~ 0x2000_03FF 番地は BOOT_ROM のワークエリアになります。 RAM 転送のプログラムは 0x2000_0400 から RAM の最終番地に格納してください。
内蔵 ROM	以下の番地はソフトなどの ID 情報や password の格納エリアとなりますので、なるべく プログラムエリアとしての使用はさけてください。 0x3F87_FFF0 ~ 0x3F87_FFFF

### 19.2.9 ブートプログラムの転送フォーマット

各コマンドのブートプログラムの転送フォーマットを示します。「19.2.10 ブートプログラム動作説明」とあわせて参照してください。

19.2.9.1 RAM 転送

表 19-6 ブートプログラムの転送フォーマット [RAM 転送の場合]

	転送バイト数	コントローラ → 本デバイスの転送データ	ボーレート	本デバイス → コントローラへの転送データ
ブート ROM	1 バイト目	シリアル動作モード & ボーレート設定 UART の場合 : 0x86 I/O インタフェースの場合 : 0x30	指定された ボーレート (注 1)	-
	2 バイト目	-		シリアル動作モードに対する ACK 応答 •UART の場合 正常 (設定可能) の場合 : 0x86 (ボーレートの設定が不可能と判断した場合は動作停止) •I/O インタフェースの場合 正常の場合 : 0x30
	3 バイト目	動作コマンドデータ (0x10)		-
	4 バイト目	-		動作コマンドに対する ACK 応答 (注 2) 正常の場合 : 0x10 異常の場合 : 0xX1 通信異常の場合 : 0xX8
	5 バイト目 ~ 16 バイト目	PASS WORD データ (12 バイト) 0x3F87_FFF4 ~ 0x3F87_FFFF		-
	17 バイト目	5 ~ 16 バイト目の CHECK SUM 値		-
	18 バイト目	-		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 : 0x10 異常の場合 : 0xX1 通信異常の場合 : 0xX8
	19 バイト目	RAM 格納開始アドレス 31 ~ 24		-
	20 バイト目	RAM 格納開始アドレス 23 ~ 16		-
	21 バイト目	RAM 格納開始アドレス 15 ~ 8		-
	22 バイト目	RAM 格納開始アドレス 7 ~ 0		-
	23 バイト目	RAM 格納バイト数 15 ~ 8		-
	24 バイト目	RAM 格納バイト数 7 ~ 0		-
	25 バイト目	19 ~ 24 バイト目の CHECK SUM 値		-
	26 バイト目	-		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 : 0x10 異常の場合 : 0xX1 通信異常の場合 : 0xX8
	27 バイト目 ~ m バイト目	RAM 格納データ		-
	m + 1 バイト目	27 ~ m バイト値の CHECK SUM 値		-
	m + 2 バイト目	-		CHECK SUM 値に対する ACK 応答 (注 2) 正常の場合 : 0x10 異常の場合 : 0xX1 通信異常の場合 : 0xX8
RAM	m + 3 バイト目	-	JUMP RAM 格納開始アドレス	

注 1) I/O インタフェースモードの場合、1 バイト目と 2 バイト目のボーレートは、指定されたボーレート ÷ 16 で行ってください。

注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。I/O インタフェースモードの場合は、通信異常の場合は発生しません。

注 3) 19 バイト目 ~ 25 バイト目のデータは、RAM 上のアドレス 0x2000\_0400 から RAM の最終番地の領域内に納まるようにプログラムしてください。

19.2.9.2 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

表 19-7 ブートプログラムの転送フォーマット  
[フラッシュメモリチップ消去およびプロテクトビット消去の場合]

	転送バイト数	コントローラ → 本デバイスの転送データ	ボーレート	本デバイス → コントローラへの転送データ
ブート ROM	1 バイト目	シリアル動作モード & ボーレート設定 UART の場合 : 0x86 I/O インタフェースの場合 : 0x30	指定された ボーレート (注 1)	-
	2 バイト目	-		シリアル動作モードに対する ACK 応答 正常 (設定可能) の場合 ●UART の場合 : 0x86 ●I/O インタフェースの場合 : 0x30 (ボーレートの設定が不可能と判断した場合は 動作停止)
	3 バイト目	動作コマンドデータ (0x40)		-
	4 バイト目	-		動作コマンドに対する ACK 応答 (注 2) 正常の場合 : 0x40 異常の場合 : 0xX1 通信異常の場合 : 0xX8
	5 バイト目	消去イネーブルコマンドデータ (0x54)		-
	6 バイト目	-		動作コマンドに対する ACK 応答 (注 2) 正常の場合 : 0x54 異常の場合 : 0xX1 通信異常の場合 : 0xX8
	7 バイト目	-		消去コマンドに対する ACK 応答 正常の場合 : 0x4F 異常の場合 : 0x4C
	8 バイト目	(次の動作コマンドデータ待ち)		-

注 1) I/O インタフェースモードの場合、1 バイト目と 2 バイト目のボーレートは、指定されたボーレート ÷ 16 で行ってください。

注 2) 異常応答後は、動作コマンド (3 バイト目) 待ちになります。I/O インタフェースモードの場合は、通信異常の場合は発生しません。

## 19.2.10 ブートプログラム動作説明

シングルブートモードで立ち上げるとブートプログラムが起動し、以下の機能を提供します。詳細は、1. RAM 転送コマンドと 2. フラッシュメモリチップ消去コマンドに記載してありますので、参照してください。

### 1. RAM 転送コマンド

RAM 転送は、コントローラから送られてくるデータを内蔵 RAM へ格納します。転送が正常に終了するとユーザープログラムの実行を開始します。ユーザープログラム領域として、ブートプログラムで使用する領域 (0x2000\_0000 ~ 0x2000\_03FF) を除く、0x2000\_0400 以降を使用可能です。実行開始アドレスは、RAM 格納開始アドレスになります。

この RAM 転送機能により、ユーザー独自のオンボードプログラミング制御を行うことができます。ユーザープログラムでオンボードプログラミングを実行するためには、19.3 で説明するフラッシュメモリコマンドシーケンスを使う必要があります。RAM 転送コマンドは、実行に先立ちパスワードの照合結果をチェックします。パスワードが一致していない場合、実行されません。

注) パスワードが消去データ (0xFF) の場合、容易にパスワードの照合が可能になり、セキュリティの確保が難しくなります。シングルブートモードを使用しない場合も固有の値を置くことを推奨します。

### 2. フラッシュメモリ SUM コマンド

フラッシュメモリ全領域の SUM を計算しその結果を返します。ブートプログラムではフラッシュメモリの全エリアのデータを読み出す動作コマンドはサポートしていません。その代わりに、このフラッシュメモリ SUM コマンドがあります。SUM を読み出すことで、アプリケーションプログラムのレビジョンを管理することができます。

### 3. 製品情報読み出しコマンド

本デバイスの製品名やメモリ情報などを返します。デバイス情報読み出しコマンドでは、フラッシュメモリの一部エリアのデータを返します。フラッシュメモリ SUM コマンド以外に、このデータを用いることで、アプリケーションプログラムのレビジョンを管理することができます。製品ごとの対象エリアは以下のとおりです。

製品	対象エリア
TMPM376FDDFG/ FDFG	0x3F87_ FFF0 ~ 0x3F87_ FFF3

### 4. フラッシュメモリチップ消去および プロテクトビット消去

このコマンドはすべてのブロックのフラッシュメモリを消去します。ライト / 消去プロテクトおよび、セキュリティ状態にかかわらず、メモリセルのすべてのブロックを消去し、すべてのブロックのライト / 消去プロテクトを消去します。コマンド終了後、FCSECBIT <SECBIT> は "1" になります。このコマンドは、パスワードを忘れた場合のブートプログラムの操作を回復する機能も兼用しているため、パスワード比較は行っていません。

#### 19.2.10.1 RAM 転送コマンド

データ転送フォーマットは表 19-6 を参照してください。

1. 1 バイト目のデータは、シリアル動作モードを判定するデータになります。シリアルの動作モードを認める方法は、後述の「シリアル動作モード判定」を参照してください。シリアルの動作モードで UART と判定した場合は、ボーレートの設定が可能かどうかを判定します。1 バイト目のデータは、受信を禁止した状態 ( $SC0MOD0<RXE> = 0$ ) にしています。

- UART で通信を行いたい場合

コントローラからターゲットボードへは、UART の設定で、所望のボーレートでデータを  $0x86$  にして送信してください。シリアルの動作モードの判定で UART と判定した場合、ボーレートの設定が可能かどうかを判定します。設定が不可能と判断した場合は動作を停止するため、通信が行えなくなります。ボーレートの設定が可能かどうかを判定する方法については、後述の「ボーレートの設定方法」を参照してください。

- I/O インタフェースで通信を行いたい場合

コントローラからターゲットボードへは、同期式の設定で、所望のボーレート  $\div 16$  でデータを  $0x30$  にして送信してください。2 バイト目も同様に、所望のボーレート  $\div 16$  にしてください。所望のボーレートで転送するのは、3 バイト目 (動作コマンドデータ) からにしてください。

I/O インタフェースの場合、CPU が受信端子を入力ポートとして見ており、その入力ポートのレベルの変化をモニタしています。従って、ボーレートが早い場合や動作周波数が多い場合は、CPU はレベルの変化を判別できないことがあります。これを防ぐために I/O インタフェースの場合、ボーレートは所望のボーレート  $\div 16$  で指定します。I/O インタフェースと判定した場合、SCLK 入力モードになります。コントローラは、AC タイミングを満足するボーレートで送信を行ってください。I/O インタフェースの場合、受信エラーフラグのチェックは行いません。従って、ACK 応答データの通信異常 ACK (bit 3) ( $0xX8$ ) はありません。

2. 2 バイト目の送信データは、1 バイト目のシリアル動作モード設定データに対する ACK 応答データになります。1 バイト目のデータが、UART と判定されボーレートの設定が可能な場合  $0x86$  を I/O インタフェースと判定された場合  $0x30$  を送信します。

- UART と判定された場合

ボーレートの設定が可能かどうかを判定します。設定が可能と判定した場合、 $SC0BRCR$  の値を書き替え、 $0x86$  を送信し、設定が不可能と判定した場合、動作を停止するため何も送信しません。コントローラは、1 バイト目のデータの送信が終了した後、タイムアウト時間 (5 秒) を設けます。タイムアウト時間内に、データ ( $0x86$ ) を正常受信できなければ、通信不能と判断してください。受信を許可 ( $SC0MOD0<RXE> = 1$ ) するタイミングは、送信バッファにデータ ( $0x86$ ) を書き込む前に行っています。

- I/O インタフェースと判定された場合

I/O インタフェースの設定になるように  $SC0MOD0$ ,  $SC0CR$  の値を書き替え、 $SC0BUF$  に  $0x30$  を書き込み、SCLK0 クロックを待ちます。コントローラは、1 バイト目のデータ送信が終了した後、アイドル時間 (数 ms) 後、SCLK クロックを出力してください。このときのボーレートは、所望のボーレート  $\div 16$  で行い、受信データが  $0x30$  なら、通信可能と判断してください。3 バイト目からは所望のボーレートで行ってください。受信を許可 ( $SC0MOD0<RXE> = 1$ ) するタイミングは、送信バッファにデータ ( $0x30$ ) を書き込む前に行っています。



3. 3 バイト目の受信データは、動作コマンドデータになります。この場合は、RAM 転送コマンドデータ (0x10) になります。
4. 4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。最初に、3 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 0xX8 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。)なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、3 バイト目の受信データが、表 19-4 の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、0x10 をエコーバック送信して RAM 転送処理ルーチンに分岐します。このルーチンに分岐後、パスワードエリアのデータをチェックします。パスワードエリアのデータのチェック方法は、後述の「パスワードについて」を参照してください。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) 0xX1 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、不定値になります。(直前の動作コマンドデータの上位 4 ビットになります。)

5. 5 バイト目 ~ 16 バイト目の受信データは、パスワードデータ (12 バイト) になります。5 バイト目の受信データから順に、フラッシュメモリの以下の表のアドレスと照合します。一致していない場合、パスワードエラーフラグをセットします。

製品	パスワード領域
TMPM376FDDFG/ FDFG	0x3F87_ FFF4 ~ 0x3F87_ FFFF

6. 17 バイト目の受信データは、CHECK SUM データになります。5 バイト目から 16 バイト目の送信データを符号なしの 8 ビット加算 (オーバーフローを無視) して得られた下位 8 ビット値の 2 の補数をコントローラから送信してください。CHECK SUM データの計算方法は、後述の「CHECK SUM の計算方法」を参照してください。
7. 18 バイト目の送信データは、5 バイト目 ~ 17 バイト目のデータに対する ACK 応答データ (CHECK SUM 値に対する ACK 応答) になります。最初に、5 バイト目 ~ 17 バイト目の受信データに受信エラーがあるかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 0x18 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので、"1" になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、17 バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、5 バイト目 ~ 16 バイト目までの受信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた値の下位 8 ビットが、0x00 かどうかをチェックしています。0x00 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 0x11 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

最後に、パスワードの照合結果をチェックします。次の場合、パスワードエラーの ACK 応答データ (bit 0) 0x11 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

- 5 バイト目 ~ 16 バイト目のパスワードデータの照合結果に関わらず、パスワードエリアの 12 バイトのデータが、0xFF 以外の同一データの場合。

- 5 バイト目 ~ 16 バイト目のパスワードデータの照合がすべて一致しない場合。

上記のチェックを終えて、すべて正常なら、正常 ACK 応答データ 0x10 を送信します。

8. 19 バイト目 ~ 22 バイト目までの受信データは、ブロック転送における格納先の RAM の開始アドレスを表します。19 バイト目がアドレスの 31 ビット ~ 24 ビットに対応し、22 バイト目が 7 ビット ~ 0 ビットに対応します。

格納先の RAM の開始アドレスは偶数アドレスで無ければいけません。

9. 23 バイト目、24 バイト目の受信データは、ブロック転送するバイト数を表します。23 バイト目が転送バイト数の 15 ビット ~ 8 ビット目に対応し、24 バイト目が 7 ビット ~ 0 ビット目に対応します。

10. 25 バイト目の受信データは、CHECK SUM データになります。19 バイト目から 24 バイト目の送信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた下位 8 ビット値の 2 の補数値をコントローラから送信してください。CHECK SUM データ計算方法は、後述の「CHECK SUM の計算方法」を参照してください。

11. 26 バイト目の送信データは、19 バイト目 ~ 25 バイト目のデータに対する ACK 応答データ (CHECK SUM 値に対する ACK 応答) になります。最初に、19 バイト目 ~ 25 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 0x18 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマンドデータの上位 4 ビットになるので "1" になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、25 バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、19 バイト目 ~ 24 バイト目までの受信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた値の下位 8 ビットが、0x00 かどうかをチェックしています。0x00 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 0x11 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。

- 19 バイト目 ~ 25 バイト目のデータは RAM 上のアドレス 0x2000\_0400 から RAM の最終番地の領域に納まるようにプログラムしてください。

上記のチェックを終えてすべて正常なら、正常 ACK 応答データ 0x10 を送信します。

12. 27 バイト目 ~ m バイト目の受信データは、RAM へ格納するデータになります。RAM に格納するデータを、19 バイト目から 22 バイト目で指定されたアドレスから書き込み、23 バイト目から 24 バイト目に指定されたバイト数分だけ書き込みます。

13. m + 1 バイト目の受信データは、CHECK SUM データになります。27 バイト目 ~ m バイト目の送信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた下位 8 ビット値の 2 の補数をコントローラから送信してください。CHECK SUM データの計算方法は、後述の「CHECK SUM の計算方法」を参照してください。

14. m + 2 バイト目の送信データは、27 バイト目 ~ m + 1 バイト目のデータに対する ACK 応答データ (CHECK SUM に対する ACK 応答) になります。最初に 27 バイト目 ~ m + 1 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 0x18 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは、直前の動作コマ

ンドデータの上位 4 ビットになるので "1" になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

次に、 $m + 1$  バイト目の CHECK SUM データをチェックします。CHECK SUM データのチェック方法は、27 バイト目 ~  $m$  バイト目までの受信データを符号なし 8 ビット加算 (オーバーフローを無視) して得られた値の下位 8 ビットが、0x00 かどうかをチェックしています。0x00 以外の場合、CHECK SUM エラーの ACK 応答データ (bit 0) 0x11 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。上記のチェックを終えてすべて正常なら、正常 ACK 応答データ 0x10 を送信します。

15.  $m + 2$  バイト目の ACK 応答データが正常 ACK 応答データの場合、正常 ACK 応答データ 0x10 を送信後、19 バイト目 ~ 22 バイト目で指定されたアドレスに分岐します。

### 19.2.10.2 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

データ転送フォーマットは表 19-7 を参照してください。

1. 1 バイト目 ~ 2 バイト目までの送受信データは RAM 転送コマンドの場合と同一になります。

2. コントローラ → デバイス

3 バイト目の受信データは動作コマンドデータになります。この場合は、フラッシュメモリチップ消去コマンドデータ (0x40) になります。

3. デバイス → コントローラ

4 バイト目の送信データは、3 バイト目の動作コマンドデータに対する ACK 応答データになります。

最初に、3 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 0xX8 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります (直前の動作コマンドデータの上位 4 ビットになります)。

次に、3 バイト目の受信データが、表 19-4 の動作コマンドデータのいずれかに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、0x40 をエコーバック送信します。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) 0xX1 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります (直前の動作コマンドデータの上位 4 ビットになります)。

4. コントローラ → デバイス

5 バイト目の受信データは消去イネーブルコマンドデータ (0x54) になります。

5. デバイス → コントローラ

6 バイト目の送信データは、5 バイト目の消去イネーブルコマンドデータに対する ACK 応答データになります。

最初に、5 バイト目の受信データに受信エラーがあるかどうかをチェックします。受信エラーがある場合、通信異常の ACK 応答データ (bit 3) 0xX8 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります (直前の動作コマンドデータの上位 4 ビットになります)。

次に、5 バイト目の受信データが、消去イネーブルコマンドデータに該当する場合は、受信データをエコーバック送信 (正常 ACK 応答データ) します。この場合、0x54 をエコーバック送信して、フラッシュメモリチップ消去処理ルーチンに分岐します。該当しない場合は、動作コマンドエラーの ACK 応答データ (bit 0) 0xX1 を送信して、次の動作コマンド (3 バイト目) データ待ちになります。送信データの上位 4 ビットは不定値になります (直前の動作コマンドデータの上位 4 ビットになります)。

6. デバイス → コントローラ

7 バイト目の送信データが正常に終了したかどうかを示します。

正常に終了した時は、終了コード (0x4F) を返します。

消去 Error が起きた場合は、エラーコード (0x4C) を返します。

7. 8 バイト目の受信データは、次の動作コマンドデータになります。

### 19.2.10.3 ACK 応答データ

ブートプログラムは処理状況を各種コードによってコントローラに送信します。表 19-8 から表 19-11 に各受信データに対する ACK 応答データを示します。ACK 応答データの上位 4 ビットは、動作コマンドデータの上位 4 ビットになります。また 3 ビット目は受信エラーを表し、0 ビット目は動作コマンドエラー、CHECK SUM エラー、パスワードエラーの状態を表します。1 ビット目と 2 ビット目は常に 0 になります。なお、I/O インタフェースの場合、受信エラーのチェックは行いません。

表 19-8 シリアル動作判定データに対する ACK 応答データ

送信データ	送信データの意味
0x86	UART での通信が可能と判定した。(注)
0x30	I/O インタフェースでの通信が可能と判定した。

注) UART の場合、ボーレートの設定が不可能と判定したら、何も送信しないで動作を停止します。

表 19-9 動作コマンドデータに対する ACK 応答データ

送信データ	送信データの意味
0x78 (注)	動作コマンドデータに受信エラーが発生した。
0x71 (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 19-10 CHECK SUM データに対する ACK 応答データ

送信データ	送信データの意味
0xN8 (注)	受信エラーが発生していた。
0xN1 (注)	CHECK SUM エラーが発生した。あるいは、パスワードエラーが発生した。
0xN0 (注)	CHECK SUM 値は正常な値と判定した。

注) 上位 4 ビットは動作コマンドデータの上位 4 ビットになります。例えば、パスワードエラー発生時は 1 (N = RAM 転送コマンドデータ [7:4]) となります。

表 19-11 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。

## 19.2.10.4 シリアル動作モード判定

コントローラは、UART で通信したい場合、所望のボーレートで 1 バイト目を 0x86 にし、I/O インタフェースで通信したい場合、所望のボーレート ÷ 16 で 1 バイト目を 0x30 にして送信してください。図 19-4 にそれぞれの場合の波形を示します。

注) 図 19-4 の A、B、C、D の各点間を、 $t_{AB}$ 、 $t_{AC}$ 、 $t_{AD}$ 、 $t_{CD}$  と表現します。

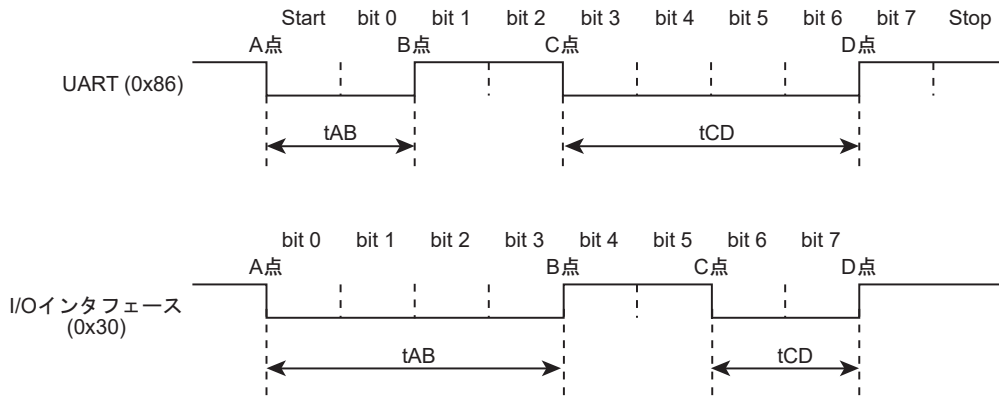


図 19-4 シリアル動作モード判定データ

ブートプログラムは、リセット解除後の 1 バイト目のシリアル動作モード判定データ (0x86, 0x30) を受信禁止状態にして、図 19-5 に示すフローチャートで、図 19-4 の  $t_{AB}$ 、 $t_{AC}$  と、 $t_{AD}$  の時間を求めています。図 19-5 のフローチャートに示すように、CPU が受信端子のレベルをモニタしてレベルの変化があると、そのときのタイマ値を取り込みます。このため、 $t_{AB}$ 、 $t_{AC}$  と、 $t_{AD}$  のタイマ値には誤差が生じます。また、ボーレートが速いときには、CPU は受信端子のレベルの変化を判断できない場合がありますので注意してください。特に、I/O インタフェースは UART に比べボーレートが速いため、このような場合が発生しやすくなります。このようなことが起こらないようにするために、I/O インタフェースの場合、コントローラのボーレートは所望ボーレート ÷ 16 にして送信してください。

図 19-5 のフローチャートに示すように、シリアル動作モードの判定は、受信端子が "L" レベルのときの時間幅の大小関係で判定しています。 $t_{AB} \leq t_{CD}$  の場合 UART と判定し、ボーレートの自動設定が可能かどうかを  $t_{AD}$  の時間から判定します。 $t_{AB} > t_{CD}$  の場合、I/O インタフェースと判定します。なお、先に述べたように、 $t_{AB}$ 、 $t_{AC}$ 、 $t_{AD}$  のタイマ値には誤差が生じているため、ボーレートが速く、動作周波数が低い場合、各タイマ値が小さくなり、意図しない判断を行うことがありますので注意してください (書き換えルーチン内で UART の再設定を行ってください)。

例えば、コントローラは UART で通信したいのに、I/O インタフェースと判定してしまうことがあります。このようなことを考慮して、コントローラは UART で通信したい場合、1 バイト目のデータを送信後、タイムアウト時間内にデータ 0x86 を正常受信できなければ通信不可能と判断してください。I/O インタフェースで通信したい場合は 1 バイト目のデータを送信後、アイドル時間後に SCLK クロックを出力してデータを受信し、受信データが 0x30 でなければ通信不可能と判断してください。

I/O インタフェースで通信したい場合は上記のとおり、 $t_{AB} > t_{CD}$  であれば 1 バイト目のデータは 0x30 でなくても構いません。A 点と C 点の立ち下がり、B 点と D 点の立ち上がりを判定できるように 0x91, 0xA1 あるいは 0xB1 を 1 バイト目のデータとして送信できます。 $t_{AB} > t_{CD}$  が成立しており、動作モード判定結果 SIO が選択された場合、(1 バイト目の送信データが 0x30 でない場合でも) 2 バイト目のデータは 0x30 となります (以下、I/O インタフェース判定用の 1 バイト目のデータは 0x30 を表記しています)。

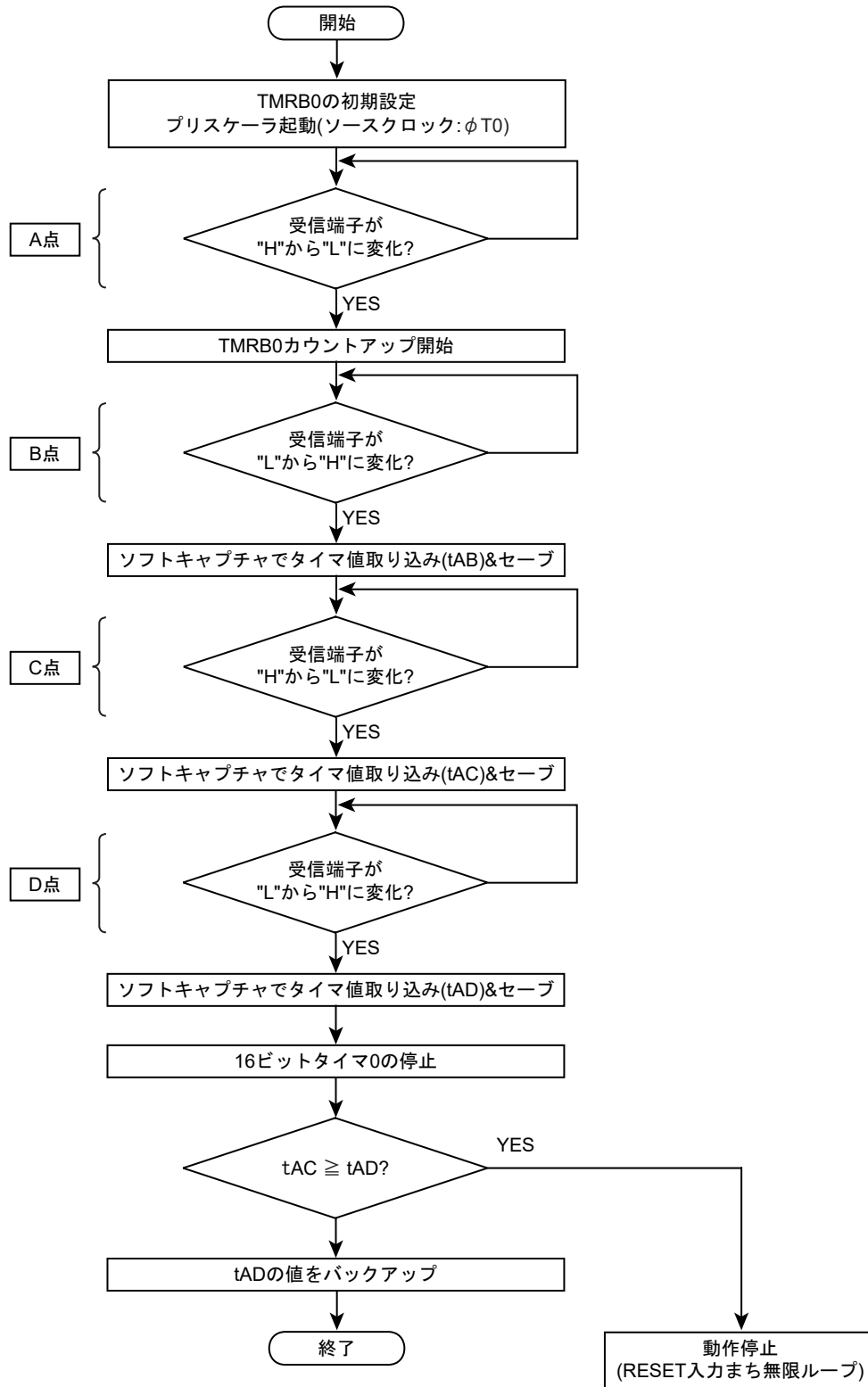


図 19-5 シリアル動作モード受信フローチャート



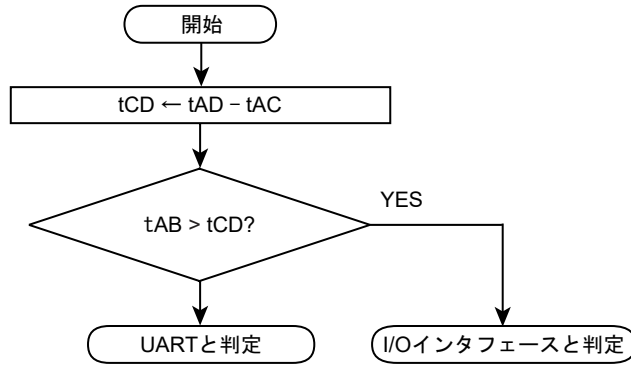


図 19-6 シリアル動作モード判定フローチャート

19.2.10.5 パスワードについて

動作コマンドによりパスワードの確認方法が異なります。パスワード領域はコマンドによらず共通で、以下の通りです。セキュリティ機能が有効な状態でもパスワードの参照を行います。

製品	パスワード領域
TMPM376FDDFG/ FDFG	0x3F87_FFF4 ~ 0x3F87_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。

図 19-7 に示すようにパスワードエリアのデータが、0xFF 以外の同一データになっていた場合、パスワードエリアエラーと判定します。パスワードエリアエラーと判定された場合、パスワードデータの照合結果に関わらず、17 バイト目の CHECK SUM 値に対する ACK 対応は 0x11 を送信します。

次に、5 バイト目 ~ 16 バイト目の受信データ (パスワードデータ) の照合を行います。12 バイト分すべてが一致しないと、パスワードエラーになります。パスワードエラーと判定された場合、17 バイト目の CHECK SUM 値に対する ACK 応答は、パスワードエラーとなります。

セキュリティ機能が有効な状態でもパスワードの参照を行います。

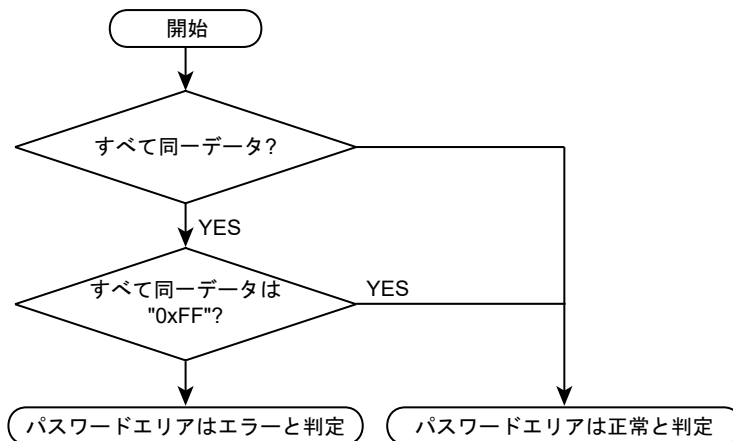


図 19-7 パスワードエリアチェックフローチャート

### 19.2.10.6 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 (オーバーフローを無視) して得られた下位 8 ビット値の 2 の補数値を求めています。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 値を求めます。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。

$$0 - 0xDB = 0x25$$

19.2.11 ブートプログラム全体フローチャート

ブートプログラム全体フローチャートを示します。

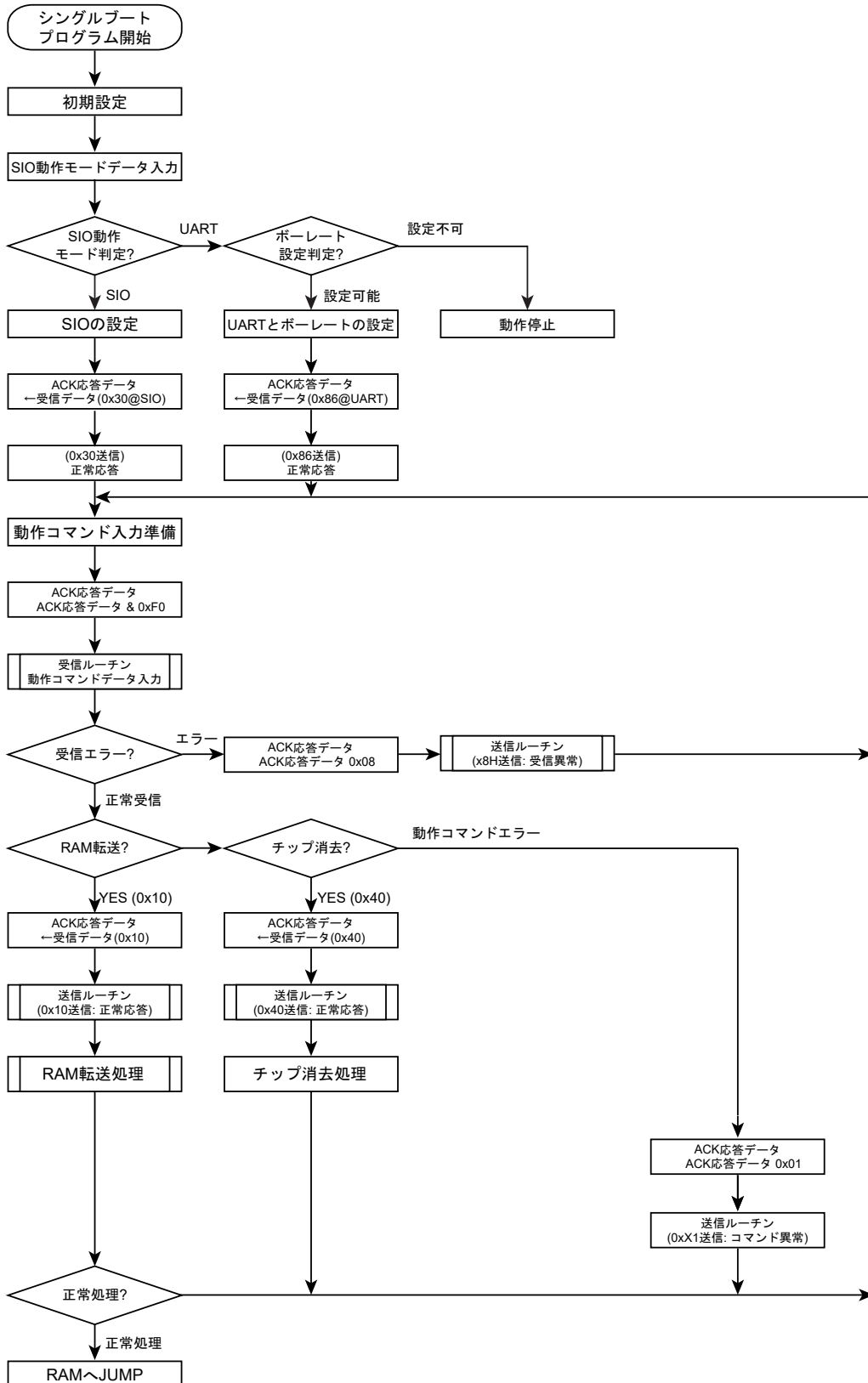


図 19-8 ブートプログラム全体フローチャート

## 19.3 オンボードプログラミングでのフラッシュメモリ書き込み / 消去

オンボードプログラミングでは、CPU によりソフトウェア的にコマンドを実行することで、フラッシュの書き込み / 消去を行います。この書き込み / 消去の制御プログラムはユーザーがあらかじめ用意しておきます。フラッシュメモリの書き込み / 消去を行っている間は、フラッシュメモリ自身の読み出しはできませんので、ユーザーブートモードに移行後、書き込み / 消去制御プログラムは内蔵 RAM 上で実行してください。

### 19.3.1 フラッシュメモリ

一部の機能を除き、フラッシュメモリの書き込みおよび消去などは JEDEC 標準コマンドに準拠しています。CPU とのインタフェースの関係上、動作コマンドのアドレス指定が標準コマンドとは異なります。

書き込み、消去を行う場合、32 ビット (ワード) のデータ転送命令を用いてフラッシュメモリへコマンドを入力します。コマンド入力後、書き込みおよび消去は内部で自動的に行われます。

表 19-12 フラッシュメモリの機能

主な機能	説明
自動ページプログラム	データ書き込みを自動で行います。
自動チップ消去	フラッシュメモリの全エリアの一括消去を自動で行います。
自動ブロック消去	ブロック単位での消去を自動で行います。
ライト / 消去プロテクト	ブロック単位ごとに書き込みおよび消去を禁止することができます。

#### 19.3.1.1 ブロック構成

##### (1) TMPM376FDDFG / FDFG

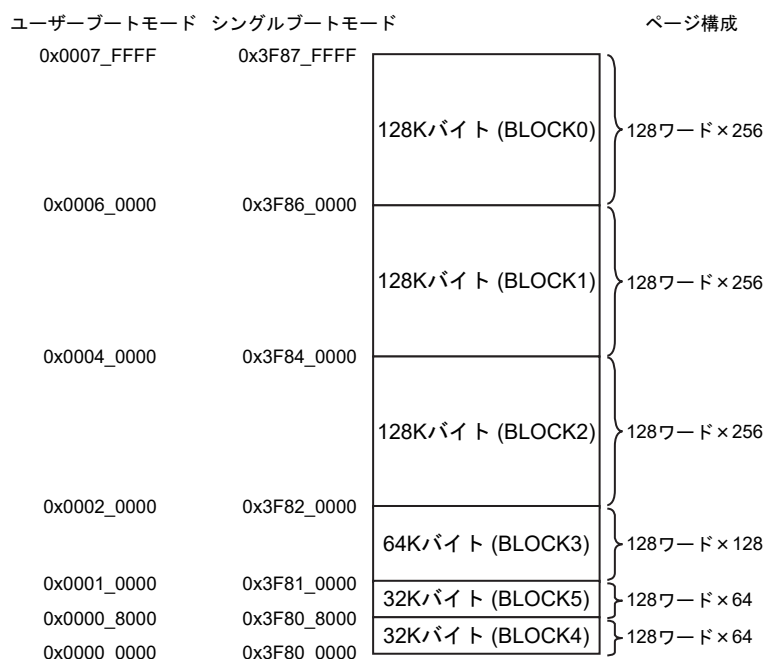


図 19-9 ブロック構成 (TMPM376FDDFG / FDFG)

### 19.3.1.2 基本動作

このフラッシュメモリには、大きく分けて以下の2種類の動作モードがあります。

- メモリデータを読み出すモード (リードモード)
- メモリデータを自動的に消去 / 書き替えるモード (自動動作)

リードモード中にコマンドシーケンスを実行することで、自動動作に移ることができます。自動動作中は、フラッシュメモリデータの読み出しとフラッシュメモリ上の命令の実行ができません。自動動作中はハードウェアリセットを除いて例外が発生した場合、リードモードに移りません。自動動作中はデバッグポート接続時のデバッグ例外とリセットを除いて、すべての例外を発生させないでください。ハードウェアリセットを除いて例外が発生した場合、リードモードに移りません。

#### (1) リード

データを読み出す場合、フラッシュメモリをリードモードにします。電源投入直後、CPUリセット解除後および自動動作の正常終了時に、フラッシュメモリはリードモードになります。自動動作の異常終了時や、他のモードからリードモードに復帰させるには、後述する Read/リセットコマンド (ソフトウェアリセット) もしくはハードウェアリセットを用います。フラッシュメモリに書かれた命令を実行する場合もリードモードでなければなりません。

- Read/リセットコマンド および Read コマンド (ソフトウェアリセット)

ID-Read コマンドを実行した場合、マクロは自動的に Read モードに復帰せず、その状態で停止します。このような状態から Read モードに復帰させるために、Read/リセットコマンドを使用します。また、途中まで入力したコマンドをキャンセルしたい場合にも、Read/リセットコマンドを使用します。Read コマンドは、フラッシュメモリの任意のアドレスに 0x0000\_00F0 データを 32 ビット (ワード) のデータ転送命令を実行してリードモードに復帰するコマンドです。

Read/リセットコマンドは第3バスライトサイクル終了後にリードモードになります。

#### (2) コマンドライト

このフラッシュメモリは、コマンドコントロール方式を用いています。コマンド実行は、フラッシュメモリに対してコマンドシーケンスを実行することで行います。フラッシュメモリは、入力されたアドレスとデータの組み合わせによって各自動動作コマンドを実行します (コマンドシーケンス参照)。

コマンドシーケンスの途中でコマンドライトをキャンセルしたい場合や、間違ったコマンドシーケンスを入力した場合は、Read/リセットコマンドを実行します。フラッシュメモリはコマンド実行を中止してリードモードになります。

フラッシュメモリに対する 32 ビット (ワード) のデータ転送命令を "バスライトサイクル" と呼びます。各コマンドはいくつかのバスサイクルで構成されています。各バスライトサイクルには順番があり、フラッシュメモリはバスライトサイクルのアドレスとデータが規定の順番でコマンドライトされた時は自動動作を実施します。規定の順番でコマンドライトされなかったバスライトサイクルがあった場合にフラッシュメモリはコマンドの実行を中止してリードモードになります。

注1) 各コマンドシーケンスは、フラッシュメモリ外のエリアから実施します。

- 注 2) 各バスライトサイクルは連続して、32 ビット (ワード) のデータ転送命令で行って下さい。各コマンドシーケンスの実行中に、フラッシュメモリへのアクセスは実施しないで下さい。また、すべての割り込み (デバックポート接続時は、デバッグ例外を除く) を発生させないでください。
- 各バスライトサイクルおよび、各コマンドシーケンスの実行中にフラッシュメモリに対して予期せぬリードアクセスが生じることになり、コマンドシーケンスがコマンドを正常に認識できない恐れがあります。各コマンドシーケンスは正常終了しない恐れがあると同時に、誤ったコマンドライトとして認識してしまう可能性があります。
- 注 3) コマンドシーケンスがコマンドを認識するために、コマンド開始前の状態がリードモードである必要があります。各コマンドシーケンスの第 1 バスライトサイクル前に FCFLCS<RDY/BSY> = 1 であることを確認してください。続いて Read コマンドを実行することを推奨します。
- 注 4) コマンド発行時、誤ったアドレスやデータをライトした場合は、必ずソフトウェアリセットを発行して、一度リードモードに戻して下さい。

### 19.3.1.3 リセット (ハードウェアリセット)

ハードウェアリセットは、自動プログラム / 消去動作の強制的な実行中止や、自動動作が異常終了したとき、コマンドライトにより設定した動作モードの解除に使用します。

本フラッシュメモリはメモリブロックとしてリセット入力を持ち、この入力は CPU のリセット信号とつながっています。このため、本デバイスの  $\overline{\text{RESET}}$  入力端子が "Low" となるか、ウォッチドッグタイマのオーバフローなどにより CPU のリセットがかかると、フラッシュメモリは自動動作の実行中であってもその動作を中止し、リードモードに戻ります。なお、自動動作の実行中にハードウェアリセットが入った場合は、データの書き替えが正常に行えませんが注意が必要です。再度、書き替えを行う処置をしてください。

CPU のリセット動作については、「19.2.1 リセット動作」を参照してください。所定のリセット入力後、CPU はフラッシュメモリよりリセットベクタデータをリードし、リセット解除後の動作を開始します。

### 19.3.1.4 コマンド説明

#### (1) 自動ページプログラム

フラッシュメモリへの書き込みは、"1" データセルを "0" データにすることです。"0" データセルを "1" データにすることはできません。"0" データセルを "1" データにするには消去動作を行う必要があります。

本デバイスの自動ページプログラムは、ページごとの書き込みとなります。1 ページは、128 ワードです。1 ページ 128 ワードの場合アドレス [31:9] が同じで、先頭アドレス [8:0] = 0、最後のアドレス [8:0] = 0x1FF のグループです。以降はページプログラムの単位をページと呼びます。

データセルへの書き込みは、内部シーケンサで自動的に行われ、CPU による外部からの制御を必要としません。自動ページプログラムの状態 (書き込み動作中であるか) は FCFLCS<RDY/BSY> にて確認できます。

また、自動ページプログラム中は、新たにコマンドシーケンスを受け付けません。自動ページプログラム動作を中止したい場合は、ハードウェアリセットを用います。これにより動作を中止させた場合、該当のページに対するデータの書き込みは正常に行われていないため、消去動作後に改めて自動ページプログラムを実行する必要があります。

自動ページプログラムは消去後のページに対して 1 回のみ可能で、"1" データセルであっても "0" データセルであってもページに対して 2 回以上の実行はできません。一度書き込み動作を行ったページに対して再度書き込みを行う場合は、自動ブロック消去または自動チップ消去コマンドを行った後に自動ページプログラムを実行しなおす必要がありますのでご注意ください。消去動作を伴わない同一ページへの 2 回以上ページプログラム実施はデバイス破損の可能性があります。

本デバイス内部で自動的なベリファイ動作は行いません。正常に書き込みができたか、実行後に読み出しをして確認してください。

自動ページプログラムは、コマンドサイクルの第 3 バスライトサイクル終了から開始します。第 5 バスライトサイクル以降は、第 4 バスライトサイクルで指定した次のアドレス (第 4 バスライトサイクルではページの先頭アドレスをコマンドライトします) から順番に書き込みを行います (データ入力は 32 ビット単位で行います)。第 4 バスライトサイクル以降のコマンドライトは必ず 32 ビット (ワード) のデータ転送命令を使用してください。このとき 32 ビット (ワード) のデータ転送命令はワード境界をまたいだ位置へ実施しないでください。第 5 バスライトサイクル以降は同一ページエリアに対してデータをコマンドライトします。また、ページの一部に書き込みをしたい場合でもページ単位で自動ページプログラムする必要があります。この場合も第 4 バスライトサイクルのアドレス入力はページ先頭アドレスにしてください。この時 "0" データセルにしたくない箇所は入力データを "1" にしてコマンドライトします。例えば、あるページ先頭アドレスの書き込みをしない場合、第 4 バスライトサイクルのデータ入力を 0xFFFFFFFF としてコマンドライトします。

第 3 バスライトサイクルを実行すると自動プログラム動作中となります。このことは FCFLCS<RDY/BSY> をモニタすることで確認できます。自動プログラム動作中は、新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。動作を中止させた場合は、データの書き込みは正常に行えませんので注意してください。1 ページのデータをコマンドライト後、ページ自動書き込みが正常終了した時に FCFLCS<RDY/BSY> = "1" となり、リードモードに復帰します。

複数のページに対してデータの書き込みを行うときは、ページごとにページプログラムコマンドを実行する必要があります (1 回の自動ページプログラムコマンドで書き込めるサイズは 1 ページです)。ページを跨ったデータ入力の自動ページプログラムはできません。

ライト / 消去プロテクトされたブロックへの書き込みはできません。自動プログラムが正常終了すると、自動的にリードモードに復帰します。このことは FCFLCS<RDY/BSY> をモニタすることで確認できます。自動プログラム動作が不良となった場合は、フラッシュメモリはこのモードのままロックされ、リードモードには復帰しません。リード状態に復帰させるにはハードウェアリセットでデバイスをリセットする必要があります。この場合、このアドレスへの書き込みが不良になっていますので、デバイスの使用を停止するか、以後このアドレスを含むブロックを使用しないことを推奨します。

注) 自動ページプログラム第 4 バスライトサイクル以降のバスライトサイクルでは、ソフトウェアリセットが無効になります。

## (2) 自動チップ消去

自動チップ消去動作は、コマンドサイクルの第 6 バスライトサイクル終了から開始します。

自動チップ消去動作は、FCFLCS<RDY/BSY> をモニタすることで確認できます。本デバイス内部で自動的なベリファイ動作は行いませんので、正常に消去ができたか、実行後に読み出しをして確認してください。自動チップ消去動作中は、新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。動作を中止した場合、データの消去は正常に行えないので、再度自動チップ消去を行う必要があります。

また、ライト / 消去プロテクトされているブロックがある場合、そのブロックの消去は行いません。すべてのブロックがライト / 消去プロテクトされている場合は、自動チップ消去を実行せず、コマンドシーケンスの第 6 バスライトサイクルの完了後にリードモードになります。自動チップ消去が正常終了すると、自動的にリードモードに復帰します。自動チップ消去動作が不良となった場合は、フラッシュメモリはこのモードのままロックされ、リードモードには復帰しません。

リードモードに復帰させるにはハードウェアリセットでデバイスをリセットする必要があります。この場合は、不良が発生したブロックの検知はできません。デバイスの使用を停止するか、ブロック消去機能を利用して不良ブロックを特定し、以降不良ブロックを使用しないことを推奨します。

### (3) 自動ブロック消去 (Block 単位)

自動ブロック消去は、コマンドサイクルの第 6 バスライトサイクル終了から開始します。

自動ブロック消去動作の状態は、FCFLCS<RDY/BSY> をモニタすることで確認できます。本デバイス内部で自動的なベリファイ動作は行いませんので、正常に消去ができたか、実行後に読み出しをして確認してください。自動ブロック消去中は新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。この場合、データの消去は正常に行えませんが、再度自動ブロック消去を行う必要があります。

また、ライト / 消去プロテクトされているブロックがある場合、そのブロックの消去を行いません。自動ブロック消去動作が不良になった場合、フラッシュメモリはこのモードのままロックされ、リードモードには復帰しません。ハードウェアリセットを用いてデバイスをリセットしてください。

### (4) 自動プロテクトビットプログラム (Block 単位)

本デバイスはプロテクトビットを内蔵し Block 単位で設定することができます。Block とプロテクトビットの関係は表 19-16 にあります。自動プロテクトビットプログラムは 1 ビット単位で実行します。ビットの指定は第 7 バスライトサイクルの PBA で行います。自動プロテクトビットプログラムにより、ブロックごとに書き込みと消去の動作を禁止 (プロテクト) することができます。各ブロックのプロテクトの状態は FCFLCS<BLPRO> で確認できます。自動プロテクトビットプログラム動作の状態は、FCFLCS<RDY/BSY> をモニタすることで確認できます。自動プロテクトビットプログラム動作中は新たなコマンドシーケンスを受け付けません。動作を中止する場合は、ハードウェアリセットを用います。この場合、プロテクトの設定は正常に行えない可能性がありますので、再度ブロックプロテクトの動作をやり直す必要があります。すべてのプロテクトビットをプログラムした後は、FCFLCS レジスタのすべての <BLPRO> ビットが "1" になっています。これ以降はすべての Block に対し、ライト / 消去はできません。

注) 自動プロテクトビットプログラム第 7 バスライトサイクルでは、ソフトウェアリセットが無効になります。FCFLCS<RDY/BSY> は、第 7 バスライトサイクル入力後から、FCFLCS<RDY/BSY> = "0" となります。

### (5) 自動プロテクトビット消去

セキュリティビットとプロテクトビットの状態によって、自動プロテクトビット消去コマンドの実行結果が異なります。FCSECBIT<SECBIT> = 1 の場合、FCFLCS レジスタのすべての <BLPRO> が "1" が、それ以外の値かで動作が決まります。自動プロテクトビット消去コマンド実施前に必ず FCFLCS<BLPRO> の値を確認してください。セキュリティの詳細については「プロテクト / セキュリティ機能」の章を参照してください。

- FCFLCS<BLPRO> = all "1" (すべてのプロテクトビットがプログラムされている) の場合



自動プロテクトビット消去コマンドをコマンドライトすると、自動的に本デバイス内部でフラッシュメモリの初期化を行います。第 7 バスライトサイクル終了後、フラッシュメモリ全エリアのデータセルの消去を行い、引き続きプロテクトビットの消去を行います。この動作に関しては FCFLCS<RDY/BSY> をモニタすることで確認できます。自動プロテクトビット消去動作が正常に終了した場合 FCFLCS = 0x00000001 となります。本デバイス内部で自動的なベリファイ動作は行いませんので、正常に消去ができたか、実行後に読み出しをして確認してください。第 7 バスライトサイクル以降の自動動作中にリードモードに復帰させるにはハードウェアリセットでデバイスをリセットする必要があります。この場合リードモードに復帰後、FCFLCS<BLPRO> にてプロテクトビットの状態を確認して、必要に応じて再度自動プロテクトビット消去または、自動チップ消去あるいは自動ブロック消去を実行する必要があります。

- FCFLCS<BLPRO> ≠ all "1" (すべてのプロテクトビットがプログラムされていない) の場合

プロテクトビットを消去することで、プロテクトの状態を解除することができます。本デバイスでは、表 19-16 のように Block 単位でプロテクトビットをプログラムするのに対して消去は 4 ビットをまとめて行います。消去を行いたいプロテクトビットの指定は第 7 バスライトサイクルで行います。各ブロックのプロテクトビットの状態は後で説明する FCFLCS<BLPRO> で確認できます。自動プロテクトビットプログラム動作の状態は、FCFLCS<RDY/BSY> をモニタすることで確認できます。自動プロテクトビット消去動作が正常に終了した場合、FCFLCS<BLPRO> の消去選択したプロテクトビットの値が "0" となります。

いずれの場合も、自動プロテクトビット消去動作中は新たなコマンドシーケンスを受け付けません。動作を中止する場合はハードウェアリセットを用います。自動プロテクトビット消去動作が正常に終了した場合はリードモードに復帰します。

注) FCFLCS<RDY/BSY> ビットは自動動作中 "0"、自動動作終了後 "1" になります。

## (6) ID-Read

ID-Read コマンドを使用すると、本デバイスに内蔵しているフラッシュメモリのタイプ等の情報を知ることができます。第 4 バスライトサイクル以降でのアドレス [15:14] の値によりロードされるデータが異なります (データ入力値は 0x00 推奨)。第 5 バスライトサイクル以降で任意のフラッシュメモリエリアの読み出しを行うと ID の値が読み出されます。ID-Read コマンド第 4 バスライトサイクル以降は自動的にリードモードに復帰しません。第 4 バスライトサイクルと ID の値の読み出しは繰り返し実行できます。リードモードへの復帰は Read/リセットコマンドまたはハードウェアリセットで行います。

### 19.3.1.5 フラッシュコントロール/ステータスレジスタ

		Base Address = 0x41FF_F000
レジスタ名		Address(Base+)
Reserved	-	0x0000, 0x0004
セキュリティビットレジスタ	FCSECBIT	0x0010
Reserved	-	0x0014
フラッシュコントロールレジスタ	FCFLCS	0x0020
Reserved	-	0x0024 ~ 0x0FFF

注) "Reserved" 表記のアドレスにはアクセスしないでください。

( 1 ) FCFLCS( フラッシュコントロールレジスタ )

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	BLPRO5	BLPRO4	BLPRO3	BLPRO2	BLPRO1	BLPRO0
リセット後	0	0	(注 2)	(注 2)	(注 2)	(注 2)	(注 2)	(注 2)
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	RDY_BSY
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-22	-	R	リードすると "0" が読めます。
21-16	BLPRO5- BLPRO0	R	Block5 ~ 0 のプロテクト状態 0: プロテクト状態ではない 1: プロテクト状態 プロテクトビット値は各ブロックのプロテクト状態に対応します。該当ビットが "1" の時は対応するブロックがプロテクト状態であることを示します。プロテクト状態のブロックは書き換えはできません。
15-1	-	R	リードすると "0" が読めます。
0	RDY/BSY	R	Ready/Busy (注 1) 0: 自動動作中 1: 自動動作終了 自動動作の状態を認識する方法として、RDY/BSY 出力を備えています。本ビットはこの機能を CPU からモニタするための機能ビットです。フラッシュメモリが自動動作中は "0" を出力し、ビジー状態であることを示します。自動動作が終了するとレディ状態となり "1" を出力し、次のコマンドを受け付けます。自動動作の結果が不良であった場合、本ビットは "0" 出力を継続します。ハードウェアリセットにより "1" に復帰します。

- 注 1) コマンド発行は、必ずレディ状態であることを確認してから発行してください。ビジー中にコマンド発行を行なった場合、正常なコマンドが送られないだけでなく、それ以降のコマンドを入力できなくなる可能性があります。その際は、ハードウェアリセットで復帰してください。ハードウェアリセットを行う場合は、システムクロックによらず 0.5 μs 以上のリセット期間が必要となります。またこの場合、リセット解除後読み出しが可能になるまで 2 ms 程度の時間がかかります。
- 注 2) プロテクト状態に対応した値になります。

## (2) FCSECBIT(セキュリティビットレジスタ)

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

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	SECBIT	R/W	セキュリティビット 0: セキュリティ機能設定不可 1: セキュリティビット設定可能

注) 本レジスタは、コールドリセットで初期化されます。

## 19.3.1.6 コマンドシーケンス一覧

各コマンドの、アドレスとデータを表 19-13 に示します。

Read コマンドの第 2 バスサイクル, Read/リセットコマンドの第 4 バスサイクル, ID-Read コマンドの第 5 バスサイクル以外はすべて "バスライトサイクル" です。バスライトサイクルは 32 ビット (ワード) のデータ転送命令で実施します。(表では、データの下位 8 ビットのデータのみ示しています。)

アドレスの詳細は、表 19-14 を参照してください。「コマンド」と記載された、Addr[15:8] に下記の値を使用します。

注) 全バスサイクル, アドレスビット [1:0] へは常に "0" を設定して下さい。

表 19-13 内部 CPU によるフラッシュメモリアクセス

コマンド シーケンス	第 1 バス サイクル	第 2 バス サイクル	第 3 バス サイクル	第 4 バス サイクル	第 5 バス サイクル	第 6 バス サイクル	第 7 バス サイクル
	Addr.	Addr.	Addr.	Addr.	Addr.	Addr.	Addr.
	Data	Data	Data	Data	Data	Data	Data
Read	0xXX	–	–	–	–	–	–
	0xF0	–	–	–	–	–	–
Read/ リセット	0x54XX	0xAAXX	0x54XX	RA	–	–	–
	0xAA	0x55	0xF0	RD	–	–	–
ID-Read	0x54XX	0xAAXX	0x54XX	IA	0xXX	–	–
	0xAA	0x55	0x90	0x00	ID	–	–
自動ページ プログラム	0x54XX	0xAAXX	0x54XX	PA	PA	PA	PA
	0xAA	0x55	0xA0	PD0	PD1	PD2	PD3
自動チップ消去	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	0x54XX	–
	0xAA	0x55	0x80	0xAA	0x55	0x10	–
自動 ブロック消去	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	BA	–
	0xAA	0x55	0x80	0xAA	0x55	0x30	–
自動プロテクトビット プログラム	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	0x54XX	PBA
	0xAA	0x55	0x9A	0xAA	0x55	0x9A	0x9A
自動プロテクトビット 消去	0x54XX	0xAAXX	0x54XX	0x54XX	0xAAXX	0x54XX	PBA
	0xAA	0x55	0x6A	0xAA	0x55	0x6A	0x6A

## 補足説明

- RA: リードアドレス
- RD: リードデータ
- IA: ID アドレス
- ID: ID データ
- PA: プログラム ページアドレス
- PD: プログラムデータ (32 ビットデータ)

第 4 バスサイクル以降 1 ページ分をアドレス順にデータ入力

- BA: ブロックアドレス
- PBA: プロテクトビットアドレス

19.3.2 バスライトサイクル時のアドレスビット構成

表 19-14 は「表 19-13 内部 CPU によるフラッシュメモリアクセス」と併せてご使用願います。

第 1 バスサイクルから「通常のバスライトサイクルアドレス設定」に従い、アドレス設定を行ってください。「0" 推奨」は適宜変更可能です。

アドレス	Addr [31:19]	Addr [18]	Addr [17]	Addr [16]	Addr [15]	Addr [14]	Addr [13:11]	Addr [10]	Addr [9]	Addr [8]	Addr [7:0]
通常 コマンド	通常のバスライトサイクルアドレス設定										
	フラッシュ領域	"0" 推奨			コマンド				Addr[1:0] = "0" 固定、他ビットは "0" 推奨		
ID-READ	IA: ID アドレス (ID-READ の第 4 バスライトサイクルアドレス設定)										
	フラッシュ領域	"0" 推奨		ID アドレス	Addr[1:0] = "0" 固定、他ビットは "0" 推奨						
ブロック 消去	BA: ブロックアドレス (ブロック消去の第 6 バスライトサイクルアドレス設定)										
	ブロックアドレス (表 19-14)				Addr[1:0] = "0" 固定、他ビットは "0" 推奨						
Auto ページ プログラム	PA: プログラムページアドレス (ページプログラムの第 4 バスライトサイクルアドレス設定)										
	ページアドレス									Addr[1:0] = "0" 固定、他ビットは "0" 推奨	
プロテクト ビットプロ グラム	PBA: プロテクトビットアドレス (プロテクトビットプログラムの第 7 バスライトサイクルアドレス設定)										
	フラッシュ領域	プロテクト ビット選択 (表 19-15)		"0" 固定			プロテクト ビット選択 (表 19-15)		Addr[1:0] = "0" 固定、他ビットは "0" 推奨		
プロテクト ビット消去	PBA: プロテクトビットアドレス (プロテクトビット消去の第 7 バスライトサイクルアドレス設定)										
	フラッシュ領域	プロテクト ビット選択 (表 19-16)		"0" 固定				Addr[1:0] = "0" 固定、他ビットは "0" 推奨			

ブロックアドレスには、消去するブロックに含まれる任意のアドレスを指定します。

ブロック構成については「19.3.1.1 ブロック構成」を参照してください。

表 19-14 ブロックアドレス表

Block	アドレス (ユーザーブートモード)	アドレス (シングルブートモード)	サイズ (Kbyte)
4	0x0000_0000 ~ 0x0000_7FFF	0x3F80_0000 ~ 0x3F80_7FFF	32
5	0x0000_8000 ~ 0x0000_FFFF	0x3F80_8000 ~ 0x3F80_FFFF	32
3	0x0001_0000 ~ 0x0001_FFFF	0x3F81_0000 ~ 0x3F81_FFFF	64
2	0x0002_0000 ~ 0x0003_FFFF	0x3F82_0000 ~ 0x3F83_FFFF	128
1	0x0004_0000 ~ 0x0005_FFFF	0x3F84_0000 ~ 0x3F85_FFFF	128
0	0x0006_0000 ~ 0x0007_FFFF	0x3F86_0000 ~ 0x3F87_FFFF	128

注) 第 1 バスサイクルから第 5 バスサイクルまで上位側のアドレスは消去するブロックのアドレスを指定してください。

表 19-15 プロテクトビットプログラムアドレス表

Block	プロテクトビット	第 7 バスライトサイクルのアドレス						
		アドレス [18]	アドレス [17]	アドレス [16]	アドレス [15:11]	アドレス [10]	アドレス [9]	アドレス [8]
Block0	<BLPRO[0]>	0	0	0	"0" 固定	0	0	"0" 推奨
Block1	<BLPRO[1]>	0	0	0		0	1	
Block2	<BLPRO[2]>	0	0	0		1	0	
Block3	<BLPRO[3]>	0	0	0		1	1	
Block4	<BLPRO[4]>	0	1	0		0	0	
Block5	<BLPRO[5]>	0	1	0		0	1	

表 19-16 プロテクトビット消去アドレス表

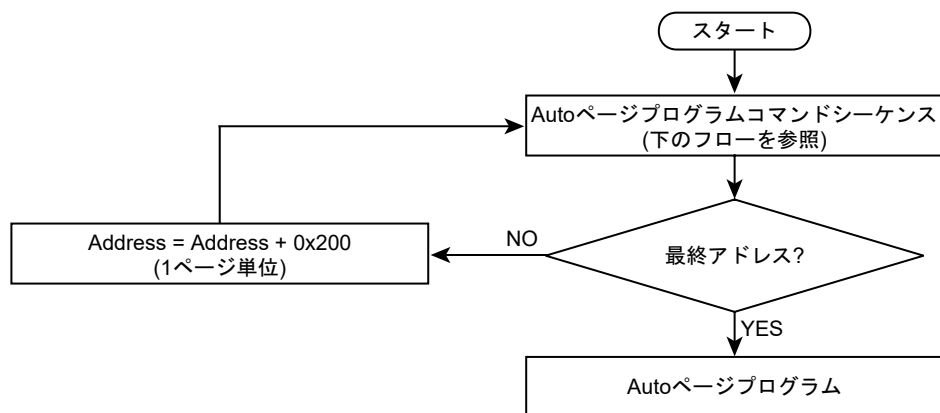
Block	プロテクトビット	第 7 バスライトサイクルのアドレス [18:17]	
		アドレス [18]	アドレス [17]
Block0 ~ 3	<BLPRO[3:0]>	0	0
Block4 ~ 5	<BLPRO[5:4]>	0	1

注) プロテクトビット消去コマンドは、プロテクトビット単位での消去はできません。

表 19-17 ID-Read コマンド第 4 バスライトサイクルの ID アドレス (IA) とその後の 32 ビット転送命令で読み出せるデータ (ID)

IA[15:14]	ID[7:0]	Code
0y00	0x98	メーカーコード
0y01	0x5A	デバイスコード
0y10	Reserved	-
0y11	0x12	マクロコード

## 19.3.2.1 フローチャート



Autoページプログラムコマンドシーケンス(アドレス/コマンド)

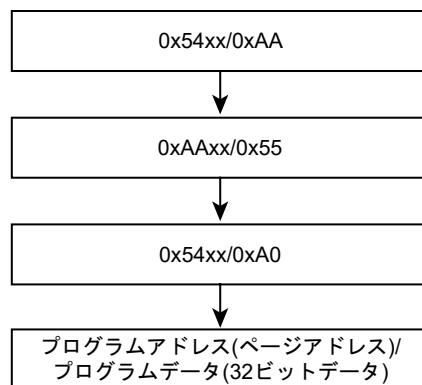


図 19-10 自動プログラム

注) 0x54xx は 0x55xx でもコマンドシーケンスを実行します。

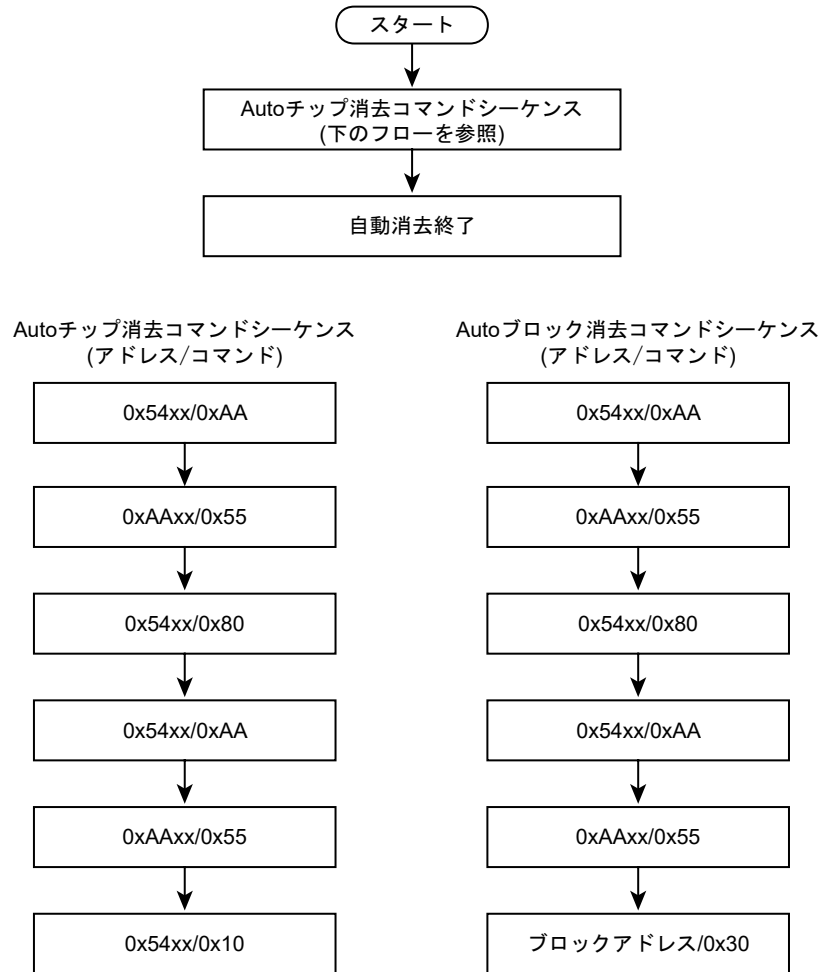


図 19-11 自動消去

注) 0x54xx は 0x55xx でもコマンドシーケンスを実行します。



## 第 20 章 プロテクト / セキュリティ機能

### 20.1 概要

本製品は内蔵 ROM (Flash) のライト / 消去をプロテクトする機能、およびライターでの内蔵 ROM (Flash) 領域を読み出し禁止に設定できるセキュリティ機能を内蔵しています。セキュリティ機能はデバッグ機能の使用制限も行います。プロテクト / セキュリティ機能として、次の 2 つの機能をもっています。

- 内蔵 ROM (Flash) のライト / 消去プロテクト
- セキュリティ機能

### 20.2 特長

#### 20.2.1 内蔵 ROM (Flash) のライト / 消去プロテクト

内蔵フラッシュは、ブロック単位で書き込みと消去の動作を禁止することができます。この機能をライト / 消去プロテクトと呼びます。

ライト / 消去プロテクト機能を有効にするためには、プロテクトをかけたいブロックに対応するプロテクトビットを "1" にします。プロテクトビットを "0" にすることによりブロックプロテクトを解除することができます。(プログラム方法については、「Flash 動作説明」の章をご覧ください。)

プロテクトビットは、FCFLCS<BLPRO[5:0]> でモニタすることができます。

#### 20.2.2 セキュリティ機能

内蔵フラッシュに対してのデータの読み出しの制限および、デバッグ機能を制限することができます。この機能をセキュリティ機能と呼びます。

セキュリティ機能が有効になる条件を、以下に示します。

1. FCSECBIT<SECBIT> が "1" にセットされている。
2. ライト / 消去プロテクト用のすべてのプロテクトビット (FCFLCS<BLPRO>) が "1" にセットされている。

注) FCSECBIT<SECBIT> はコールドリセットで "1" にセットされます。

セキュリティ機能が有効な状態の制限内容を、表 20-1 に示します。

表 20-1 セキュリティ機能有効時の制限内容

項目	内容
1) ROM 領域のデータの読み出し	CPU からの読み出しは可能です。
2) デバッグポート	JTAG/SW, トレースの通信ができなくなります。
3) フラッシュに対するコマンドの実行	フラッシュに対してのコマンドライトが受け付けられません。また、ライト / 消去プロテクト用のプロテクトビットを消去しようとすると、チップ消去が行われ、すべてのプロテクトビットも消去されます。

## 20.3 レジスタ

### 20.3.1 レジスタ一覧

Base Address = 0x41FF\_F000

レジスタ名		Address(Base+)
Reserved	-	0x0000, 0x0004
セキュリティビットレジスタ	FCSECBIT	0x0010
Reserved	-	0x0014
フラッシュコントロールレジスタ	FCFLCS	0x0020
Reserved	-	0x0024 ~ 0x0FFF

注) "Reserved" 表記のアドレスにはアクセスしないでください。

20.3.2 FCFLCS(フラッシュコントロールレジスタ)

	31	30	29	28	27	26	25	24
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	23	22	21	20	19	18	17	16
bit symbol	-	-	BLPRO5	BLPRO4	BLPRO3	BLPRO2	BLPRO1	BLPRO0
リセット後	0	0	(注2)	(注2)	(注2)	(注2)	(注2)	(注2)
	15	14	13	12	11	10	9	8
bit symbol	-	-	-	-	-	-	-	-
リセット後	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
bit symbol	-	-	-	-	-	-	-	RDY_BSY
リセット後	0	0	0	0	0	0	0	1

Bit	Bit Symbol	Type	機能
31-22	-	R	リードすると"0"が読めます。
21-16	BLPRO5 - BLPRO0	R	Block5 ~ 0のプロテクト状態(注2) 0: プロテクト状態ではない 1: プロテクト状態 プロテクトビット値は各ブロックのプロテクト状態に対応します。該当ビットが"1"の時は対応するブロックがプロテクト状態であることを示します。プロテクト状態のブロックは書き換えはできません。
15-1	-	R	リードすると"0"が読めます。
0	RDY/BSY	R	Ready/Busy フラグ(注1) 0: 自動動作中 1: 自動動作終了 自動動作の状態を認識する方法として、RDY/BSY出力を備えています。本ビットはこの機能をCPUからモニタするための機能ビットです。フラッシュメモリが自動動作中は"0"を出力し、ビジー状態であることを示します。自動動作が終了するとレディ状態となり"1"を出力し、次のコマンドを受け付けます。自動動作の結果が不良であった場合、本ビットは"0"出力を継続します。ハードウェアリセットにより"1"に復帰します。

- 注 1) コマンド発行は、必ずレディ状態であることを確認してから発行してください。ビジー中にコマンド発行を行なった場合、正常なコマンドが送られないだけでなく、それ以降のコマンドを入力できなくなる可能性があります。その際は、ハードウェアリセットで復帰してください。  
ハードウェアリセットを行う場合は、システムクロックによらず 0.5 μs 以上のリセット期間が必要となります。またこの場合、リセット解除後読み出しが可能になるまで 2 ms 程度の時間がかかります。
- 注 2) プロテクト状態に対応した値が読めます。

20.3.3 FCSECBIT(セキュリティビットレジスタ)

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

Bit	Bit Symbol	Type	機能
31-1	-	R	リードすると"0"が読めます。
0	SECBIT	R/W	セキュリティビット 0: セキュリティ機能設定不可 1: セキュリティ機能設定可能

注) 本レジスタは、コールドリセットで初期化されます。

## 20.4 設定 / 解除方法

### 20.4.1 内蔵 ROM (Flash) のライト / 消去プロテクト

プロテクトビットの書き込みと消去はコマンドシーケンスを用いて行います。

プロテクトビットの書き込みは1ブロックごと、消去はBlock0~3とBlock4~5の2つの単位で行います。

全Blockすべてのプロテクトビットが"1"でかつ、FCSECBIT<SECBIT>が"1"の場合、セキュリティ機能が有効になっているので注意が必要です。この状態でプロテクトビットの消去を行うと、チップ消去を行った上でプロテクトビットすべてを消去します。このため、FCSECBIT<SECBIT>を"0"にしてからプロテクトビットの消去を行う必要があります。

コマンドシーケンスの詳細は「Flash動作説明」の章を参照してください。

### 20.4.2 セキュリティビット

セキュリティ機能を有効にするFCSECBIT<SECBIT>は電源投入時のリセットで"1"にセットされます。FCSECBIT<SECBIT>の書き替えは以下の手順で行います。

1. FCSECBITに対して特定のコード(0xa74a9d23)を書き込む。
2. 1.の書き込みから16クロック以内にデータを書き込む。

注) 上記1., 2.の書き込みは32bit転送命令で行ってください。



## 第 21 章 デバッグインタフェース

### 21.1 仕様概要

TMPM376FDDFG/FDFG 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、および内部プログラムをトレース出力するための ETM™(Embedded Trace Macrocell) ユニー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 (TRACEDATA0~1,SWV) へ出力されます。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 " Cortex-M3 テクニカルリファレンスマニュアル " を参照してください。

### 21.2 SWJ-DP の特長

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

### 21.3 ETM の特長

データ信号 2pin(TRACEDATA0~1) とクロック信号 1pin(TRACECLK) および、シリアルワイヤビューア信号 1pin(SWV) による SWV トレース出力をサポートしています。

## 21.4 端子情報

デバッグインタフェース端子は汎用ポートと兼用しています。

デバッグインタフェース端子と兼用の汎用ポートのうち、PB3/PB4 は JTAG デバッグポート機能とシリアルワイヤデバッグポート機能、PB5 は JTAG デバッグポート機能および SWV トレース出力機能と兼用しています。

表 21-1 SWJ-DP,ETM デバッグ機能

SWJ-DP 端子名	汎用 ポート名	JTAG デバッグ機能		SW デバッグ機能	
		I/O	説明	I/O	説明
TMS/SWDIO	PB3	入力	JTAG Test Mode Selection	入出力	Serial Wire Data Input/Output
TCK/SWCLK	PB4	入力	JTAG Test Check	入力	Serial Wire Clock
TDO/SWV	PB5	出力	JTAG Test Data Output	(出力) (注 1)	(Serial Wire Viewer Output)
TDI	PB6	入力	JTAG Test Data Input	-	-
$\overline{\text{TRST}}$	PB7	入力	JTAG Test RESET	-	-
TRACECLK	PB0	出力	TRACE Clock Output		
TRACEDATA0	PB1	出力	TRACE DATA Output0		
TRACEDATA1	PB2	出力	TRACEDATA Output1		

注 1) SWV 機能を許可した場合

リセット解除後、PB3/PB4/PB5/PB6/PB7 はデバッグインタフェース端子となりますが、その他のデバッグインタフェース端子は汎用ポート設定です。必要に応じてデバッグインタフェースを使用する設定をおこなってください。また、使用しないデバッグインタフェース端子は汎用ポートとして使用可能です。

表 21-2 にデバッグインタフェースの端子情報とリセット解除後のポート設定を示します。

表 21-2 デバッグインタフェース / 汎用ポート兼用端子のリセット解除後設定値

初期設定	ポート (ビット名)	デバッグ機能	ポート制御レジスタの設定値					
			機能 (PBFRR)	入力 (PBIE)	出力 (PBCR)	オープン ドレイン (PBOD)	プル アップ (PBPUP)	プル ダウン (PBPDN)
ポート	PB0	TRACECLK	0	0	0	0	0	0
ポート	PB1	TRACEDATA0	0	0	0	0	0	0
ポート	PB2	TRACEDATA1	0	0	0	0	0	0
デバッグ	PB3	TMS/SWDIO	1	1	1	0	1	0
デバッグ	PB4	TCK/SWCLK	1	1	0	0	0	1
デバッグ	PB5	TDO/SWV	1	0	1	0	0	0
デバッグ	PB6	TDI	1	1	0	0	1	0
デバッグ	PB7	$\overline{\text{TRST}}$	1	1	0	0	1	0

低消費電力モードを使用する場合には、以下の注意事項に留意してください。

- 注 1) PB3/PB5 がデバッグインタフェース設定の場合、CGSTBYCR<DRVE> の設定によらず、STOP モード中も出力が有効な状態で保持されます。
- 注 2) PB4 がデバッグインタフェース設定の場合、STOP モードで十分な低消費電力効果が得られません。デバッグインタフェースとして使用しない場合は、PB4 をポート設定にしてください。



## 21.5 デバッグツールとの接続

### 21.5.1 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ていますので、外部にプルアップ / プルダウン抵抗を接続する場合には注意してください。

### 21.5.2 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場合

デバッグ時に、リセット解除後ユーザプログラムでデバッグインタフェース端子を汎用ポートに設定すると、それ以降はデバッグツールからの制御が受け付けられず、デバッグツールを使用してデバッグ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 21-3 デバッグインタフェース端子の使用例

デバッグインタフェースの 組み合わせ (例)	使用するデバッグインタフェース端子							
	TRST	TDI	TDO/ SWV	TCK/ SWCLK	TMS/ SWDIO	TRACE DATA1	TRACE DATA0	TRACE CLK
JTAG+SW(リセット解除時)						×	×	×
JTAG+SW(TRST無し)	×(注)					×	×	×
JTAG+TRACE								
SW	×	×	×			×	×	×
SW+SWV	×	×				×	×	×
デバッグ機能ディセーブル	×	×	×	×	×	×	×	×

;イネーブル、×;ディセーブル(汎用ポートとして使用)

注) TRST が割り当てられている端子は、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。

## 21.6 ホールトモード中の周辺機能(プログラム実行の一時停止時)

Cortex-M3 コアがホールトモードに入ると、ウォッチドッグタイマ(WDT)のカウントが自動的に停止します。また、16ビットタイマ(TMRB)はホールトモード時に動作するかどうか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。



## 第 22 章 電気的特性

### 22.1 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		VOOUT15	-0.3 to 3	V
		VOOUT3	-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	$\Sigma I_{OL}$	50	
高レベル 出力電流	1 端子	$I_{OH}$	-5	
	合計	$\Sigma I_{OH}$	-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	Flash W/E 時を除く	$T_{OPR}$	-40 ~ 85	°C
	Flash W/E 時		0 ~ 70	

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

## 22.2 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 $f_{OSC} = 8 \sim 10 \text{ MHz}$ $f_{SYS} = 1 \sim 80 \text{ MHz}$	4.5	-	5.5	V
電源電圧 (FLASH W/E 時) (注 2)	DVDD5 RVDD5 AVDD5A AVDD5B	VDD $f_{OSC} = 8 \sim 10 \text{ MHz}$ $f_{SYS} = 1 \sim 80 \text{ MHz}$ (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 $f_{OSC} = 8 \sim 10 \text{ MHz}$ $f_{SYS} = 1 \sim 80 \text{ MHz}$	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 3)	$C_{out}$	RVDD5 = 4.5V ~ 5.5V VOUT15, VOUT3	3.3	-	4.7	μF
低レベル出力電圧	$V_{OL}$	$I_{OL} = 1.6 \text{ mA}$ VDD ≥ 4.5V (注 4)	-	-	0.4	V
高レベル出力電圧	$V_{OH}$	$I_{OH} = -1.6 \text{ mA}$ VDD ≥ 4.5V (注 4)	4.1	-	-	V
入力リーク電流	$I_{L11}$	$0.0 \leq V_{IN} \leq VDD$ (注 4)	-	0.02	±5	μA
出力リーク電流	$I_{LO}$	$0.2 \leq V_{IN} \leq VDD - 0.2$ (注 4)	-	0.05	±10	
リセットプルアップ抵抗	$R_{RST}$	$4.5 \leq VDD \leq 5.5$ (注 4)	-	50	150	kΩ
プログラブルプルアップ/ダウン抵抗	$P_{KH}$	$4.5 \leq VDD \leq 5.5$ (注 4)	-	50	150	kΩ
シュミット入力幅	VTH	$4.5 \leq VDD \leq 5.5$ (注 4)	0.3	0.6	-	V
Pin 容量 (電源端子を除く)	$C_{IO}$	$f_c = 1 \text{ MHz}$	-	-	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、DVDD5A、DVDD5B、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、 $3.9V \leq VDD < 4.5V$  の範囲では 12 ビット A/D コンバータ変換特性および AC 電気的特性は保障外となります。詳細は「図 22-2 電源投入シーケンス (外部リセット IC を使用しない場合)」を参照してください。

注 4) VOUT15 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。VOUT15 端子と VOUT3 端子から IC 外部への電源供給はできません。

注 5) VDD = DVDD5 = RVDD5 = AVDD5A = AVDD5B

## 22.3 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
NORMAL (注 2) ギア比 1/1	I <sub>DD</sub>	f <sub>sys</sub> = 80 MHz	-	70	80	mA
IDLE (注 4) ギア比 1/1			-	21	30	
STOP		-	-	7	11	mA

注 1) Typ 値は、特に指定のない限り Ta=25°C、DVDD5 = AVDD5A/B = RVDD5 = 5V です。

注 2) IDD NORMAL の測定条件：  
A/D を除く内蔵周辺機能はすべて動作

注 3) AD のアナログ基準電圧はオフできません。

注 4) IDD IDLE の測定条件：  
内蔵周辺機能はすべて停止

## 22.4 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)、(注 2)	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 ≥ 2 μs	-	-	± 6	LSB
微分直線性誤差			-	-	± 5	
オフセット誤差			-	-	± 5	
フルスケール誤差			-	-	± 5	
総合誤差			-	-	-10 ~ +5	

注 1) AD コンバータ 1 ユニットの電流です。

注 2) 本マイコンではアナログ基準電源電流は常に流れます。

注 3) 1LSB = (AVDD - AVSS)/4096 [V]

注 4) AVDD = AVDD5A = AVDD5B、AVSS = AVSSA = AVSSB

注 5) AD コンバータのみ動作時の特性です。

## 22.5 AC 電気的特性

### 22.5.1 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 =  $0.8 \times VDD$ 、Low =  $0.2 \times VDD$
- 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

### 22.5.2 シリアルチャネル (SIO/UART)

#### 22.5.2.1 I/O インターフェースモード (VDD = 4.5V ~ 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		単位
		Min.	Max	Min.	Max	
SCLK クロック High 幅 (入力)	t <sub>SCH</sub>	4x	-	50	-	ns
SCLK クロック Low 幅 (入力)	t <sub>SCL</sub>	4x	-	50	-	
SCLK 周期	t <sub>SCY</sub>	8x	-	100	-	
有効 Data 入力 ← SCLK 立ち上がり / 立ち下がり (注 1)	t <sub>SRD</sub>	30	-	30	-	
SCLK 立ち上がり → Input Data 保持 / 立ち下がり (注 1)	t <sub>HSR</sub>	x + 30	-	42.5	-	

[データ出力]

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
SCLK クロック High 幅 (入力)	t <sub>SCH</sub>	4x	-	82.5 (注 3)	-	ns
SCLK クロック Low 幅 (入力)	t <sub>SCL</sub>	4x	-	82.5 (注 3)	-	
SCLK 周期	t <sub>SCY</sub>	8x	-	165	-	
OutputData ← SCLK 立ち上がり / 立ち下がり (注 1)	t <sub>OSS</sub>	t <sub>SCY</sub> / 2 - 3x - 45 (注 2)	-	0 (注 2)	-	
SCLK 立ち上がり → Output Data 保持 / 立ち下がり (注 1)	t <sub>OHS</sub>	t <sub>SCY</sub> / 2	-	82.5	-	

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<sub>OSS</sub> がマイナスにならない最小値を示しています。

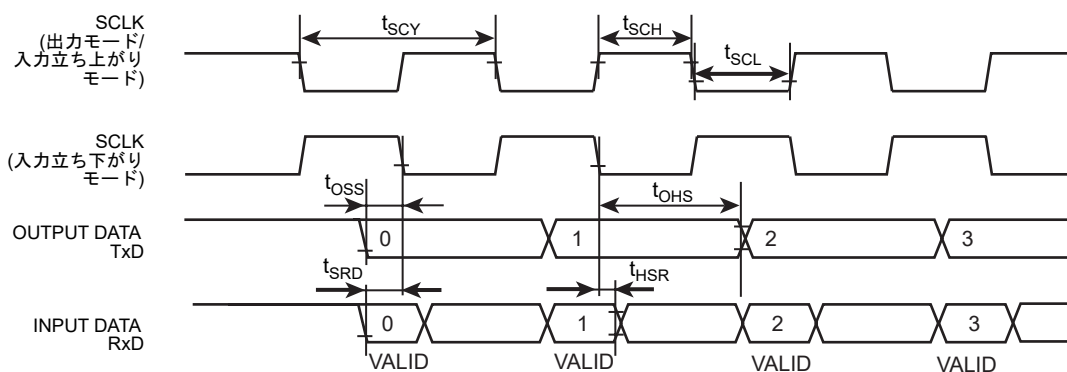
(2) SCLK 出力モード (Ta = -40 ~ 85°C)

[出力]

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
SCLK 周期 (プログラマブル)	$t_{SCY}$	4x	-	60	-	ns
Output Data ← SCLK 立ち上がり	$t_{OSS}$	$t_{SCY}/2 - 30$ (注 1)	-	0 (注 2)	-	
SCLK 立ち上がり → Output Data 保持	$t_{OHS}$	$t_{SCY}/2 - 30$ (注 1)	-	0 (注 2)	-	
有効 Data 入力 ← SCLK 立ち上がり	$t_{SRD}$	45	-	45	-	
SCLK 立ち上がり → Input Data 保持	$t_{HSR}$	0	-	0	-	

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)  $t_{OSS}$  がマイナスにならない最小値を示しています。





22.5.3 シリアルバスインターフェース (I2C/SIO)

22.5.3.1 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  $f_{sys}$  と同じ周期です。この周期は、クロックギアの設定に依存します。

$n$  は 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	-	$\mu s$
SCL クロック Low 幅 (入力) (注 1)	$t_{LOW}$	-	-	4.7	-	1.3	-	$\mu s$
SCL クロック High 幅 (入力) (注 2)	$t_{HIGH}$	-	-	4.0	-	0.6	-	$\mu s$
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	-	$\mu s$
データ保持時間 (入力) (注 3, 4)	$t_{HD; DAT}$	-	-	0.0	-	0.0	-	$\mu s$
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	-	$\mu s$
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	-	$\mu s$

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):  $(2^{n-1} + 14)/x$

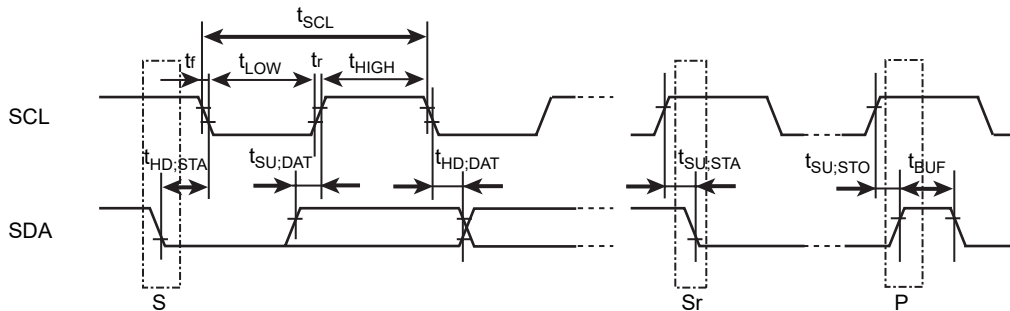
通信規格上、標準モード / 高速モードの最高速は 100 kHz/400 kHz です。内部 SCL クロックの周波数の設定は、使用される  $f_{sys}$  と上記注 1, 注 2 の計算式にて設定されますのでご注意願います。

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) フィリップス仕様では内部で、SDA 入力時にデータホールド 300nsec を確保して SCL 立ち下がり時の不安定状態を回避することになっていますが、本製品では対応していません。また SCL のエッジスローブコントロール機能を持っていません。従って、SCL / SDA の  $tr/tf$  を含めて BUS 上で上表のデータ保持時間 (入力) を守るように設計してください。

注 5) ソフトウェアに依存します。

注 6) フィリップス仕様では、ファーストモードデバイスへの電源供給が切れた場合、バスラインを妨げることがないように SDA および SCL 信号の I/O ピンと電源との接続が外れるようにする必要がありますが、本デバイスでは対応していません。



S: スタートコンディション  
 Sr: 再スタートコンディション  
 P: ストップコンディション

### 22.5.3.2 クロック同期式 8 ビット SIO モード

x は I2C の動作クロックの周期を表します。I2C の動作クロックは、システムクロック  $f_{sys}$  と同じ周期です。この周期は、クロックギアの設定に依存します。

#### (1) SCK 入力モード (SCK デューティ 50% の場合)

[ データ入力 ]

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
SCK クロック High 幅 (入力)	$t_{SCH}$	4x	–	82.5	–	ns
SCK クロック Low 幅 (入力)	$t_{SCL}$	4x	–	82.5	–	
SCK 周期	$t_{SCY}$	$8xt_{SCH} + t_{SCL}$	–	165	–	
有効 Data 入力 ←SCK 立ち上がり	$t_{SRD}$	$30 - x$	–	17.5	–	
SCK 立ち上がり →Input Data 保持	$t_{HSR}$	30	–	30	–	

[ データ出力 ]

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
SCK クロック High 幅 (入力)	$t_{SCH}$	4x	–	82.5	–	ns
SCK クロック Low 幅 (入力)	$t_{SCL}$	4x	–	82.5	–	
SCK 周期	$t_{SCY}$	$8xt_{SCH} + t_{SCL}$	–	165	–	
Output Data ←SCK 立ち上がり	$t_{OSS}$	$t_{SCY}/2 - 3x - 45$	–	0 (注)	–	
SCK 立ち上がり →Output Data 保持	$t_{OHS}$	$t_{SCY}/2 + x$	–	95	–	

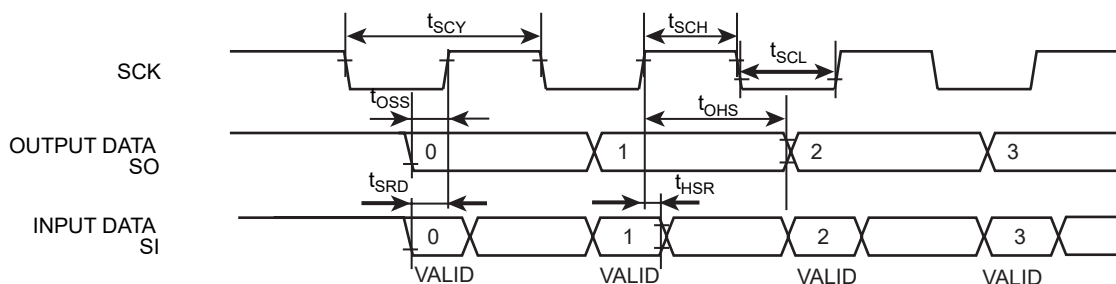
注) 計算式によりマイナス値となる場合は、シリアルクロックを落としてください。

#### (2) SCK 出力モード (SCK デューティ 50% の場合)

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
SCK 周期 (プログラマブル)	$t_{SCY}$	16x	–	200	–	ns
Output Data ←SCK 立ち上がり	$t_{OSS}$	$t_{SCY}/2 - 20$	–	80	–	
SCK 立ち上がり →Output Data 保持	$t_{OHS}$	$t_{SCY}/2 - 20$	–	80	–	
有効データ入力 ←SCK 立ち上がり	$t_{SRD}$	45	–	45	–	
SCK 立ち上がり →Input Data 保持	$t_{HSR}$	0	–	0	–	

注 1) 自動 wait 後の SCK 周期が 14x になります。

注 2) 自動 wait 後の  $t_{OSS}$  データ出力が、 $t_{SCY}/2 - x - 20$  になる場合があります。



## 22.5.4 イベントカウンタ

x は TMRB の動作クロックの周期を表します。TMRB の動作クロックは、システムクロック  $f_{sys}$  と同じ周期です。この周期は、クロックギアの設定に依存します。

$T_a = -40 \sim 85^\circ\text{C}$  (1 ~ 80MHz)

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
クロック低レベルパルス幅	$t_{VCKL}$	$2x + 100$	–	125	–	ns
クロック高レベルパルス幅	$t_{VCKH}$	$2x + 100$	–	125	–	ns

## 22.5.5 キャプチャ

x は TMRB の動作クロックの周期を表します。TMRB の動作クロックは、システムクロック  $f_{sys}$  と同じ周期です。この周期は、クロックギアの設定に依存します。

$T_a = -40 \sim 85^\circ\text{C}$  (1 ~ 80MHz)

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
低レベルパルス幅	$t_{CPL}$	$2x + 100$	–	125	–	ns
高レベルパルス幅	$t_{CPH}$	$2x + 100$	–	125	–	ns

## 22.5.6 外部割り込み

表中の x はシステムクロック  $f_{sys}$  の周期を表します。

$T_a = -40 \sim 85^\circ\text{C}$  (1 ~ 80MHz)

## 1. STOP 解除割り込み以外

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
INT0 ~ F 低レベルパルス幅	$t_{INTAL}$	$x + 100$	–	112.5	–	ns
INT0 ~ F 高レベルパルス幅	$t_{INTAH}$	$x + 100$	–	112.5	–	ns

## 2. STOP 解除割り込み

項目	記号	計算式		80 MHz		単位
		Min.	Max	Min.	Max	
INT0 ~ F 低レベルパルス幅	$t_{INTBL}$	100	–	100	–	ns
INT0 ~ F 高レベルパルス幅	$t_{INTBH}$	100	–	100	–	ns

## 22.5.7 デバッグ通信

### 22.5.7.1 AC 測定条件

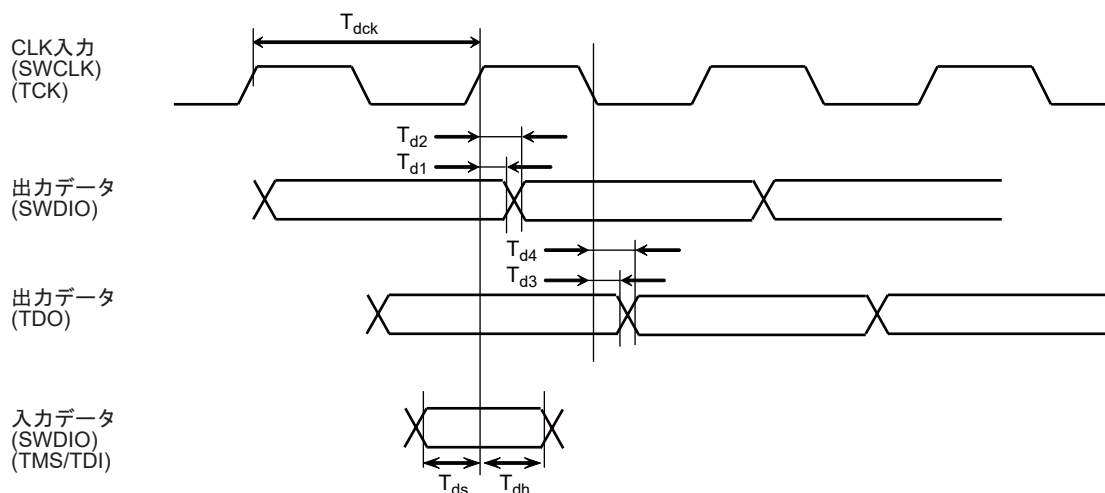
- 出力レベル : High =  $0.7 \times DVDD5$ , Low =  $0.3 \times DVDD5$
- 負荷容量 : TRACECLK CL = 25pF、TRACEDATA CL = 20pF

### 22.5.7.2 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	$T_{d2}$	–	37	
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	–	

### 22.5.7.3 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	$T_{d4}$	–	37	
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	–	

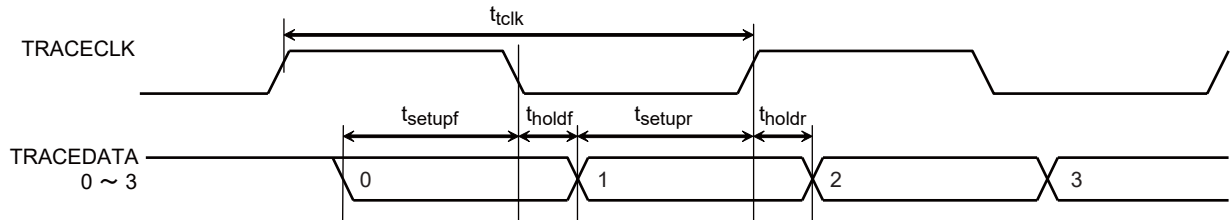


### 22.5.8 ETM トレース

AC 測定条件は以下の通りです。

- 出力レベル : High =  $0.7 \times DVDD5$ , Low =  $0.3 \times DVDD5$
- 負荷容量 : TRACECLK CL = 25pF、TRACEDATA CL = 20pF

項目	記号	Min.	Max	単位
TRACECLK 周期	$t_{clk}$	25	-	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	-	



### 22.5.9 Flash 特性

項目	条件	Min.	Typ.	Max	単位
フラッシュメモリ書き換え保障回数	Ta = 0 to 70°C DVDD5 = RVDD5 = AVDD5A = AVDD5B = 4.5 ~ 5.5V	-	-	100	回

### 22.5.10 内蔵発振回路特性

項目	記号	条件	Min.	Typ.	Max	単位
発振周波数	fosc2	Ta = -40 to 85°C	9.0	9.5	10	MHz

### 22.5.11 外部発振子

項目	記号	条件	Min.	Typ.	Max	単位
高周波発振	fosc1	Ta = -40 to 85°C	8	-	10	MHz

## 22.6 発振回路

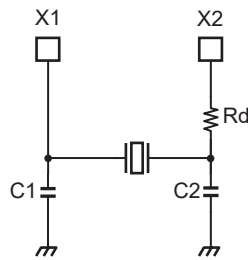


図 22-1 高周波発振回路例

注) 発振の安定には、発振子の位置、負荷容量を適切にする必要があります。これらは基板パターンにより大きな影響を受けます。安定した発振を得るため、ご使用される基板での評価をされるようお願いいたします。

本製品は、下記のメーカーの発振子を用いて評価しています。発振回路設計時に発振子の選択に活用願います。

### 22.6.1 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。

## 22.7 電源投入時のご注意

ポート L(PL0、PL1 端子) に対する注意事項

電源立ち上がり時、VDD 端子が推奨動作電圧範囲 (4.5 ~ 5.5V) に到達し 200  $\mu$ s 経過するまでの期間、ポート L(PL0、PL1 端子) はオープンまたは 0.5V 以下となるようにしてください。

動作時に電源電圧が降下してパワーオンリセット回路がリセット信号を発生し、再度電源立ち上がりの場合も同様の注意が必要です。

注) VDD = DVDD5 = RVDD5 = DVDD5A = DVDD5B

### 22.7.1 外部リセット IC を使用しない場合

注 1) 内蔵のパワーオンリセットを使用して電源を立ち上げる場合、DVDD5 と RVDD5 端子は 3ms 以内に推奨動作電圧範囲 (3.9 ~ 5.5V) に到達するように電源を立ち上げてください。

注 2) マイコンの動作開始後、電圧検出回路 (VLTD) で任意の検出レベルを選択し、動作をイネーブルにしてください。

記号	項目	Min	Typ.	Max	単位
$t_{PWUP}$	リセット解除後のウォーミングアップ時間	-	-	3.7	ms
$t_{DVDD}$	電源立ち上がり時間	-	-	3	
$V_{VLTD}$	電圧検出回路検出電圧 (VDLVL=01 時)	3.9	4.1	4.3	V
$V_{PORH}$	パワーオンリセット解除電圧 (パワーオンカウンタが動作開始する電圧)	2.8	3	3.2	
$V_{PORL}$	パワーオンリセット検知電圧	2.6	2.8	3.0	

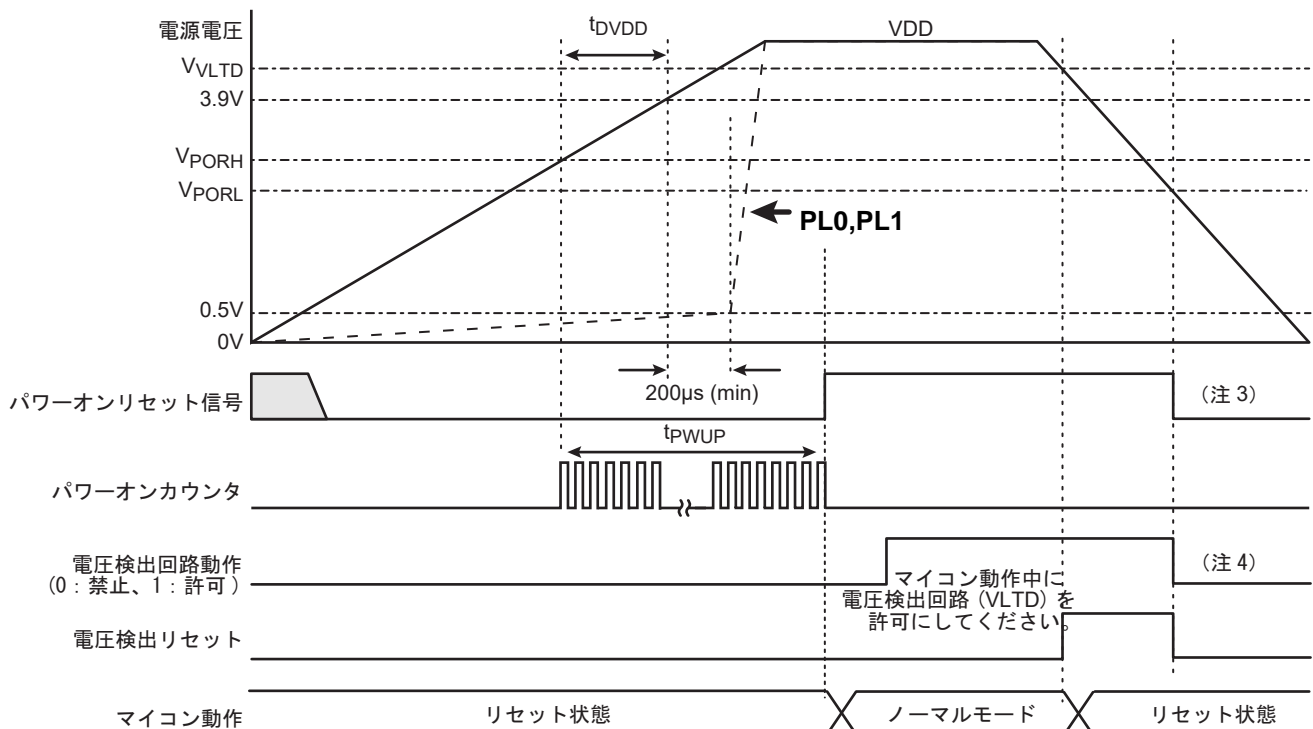


図 22-2 電源投入シーケンス (外部リセット IC を使用しない場合)

- 注 1)  $VDD = DVDD5 = RVDD5 = AVDD5A = AVDD5B$
- 注 2) パワーオンリセット解除電圧 ( $V_{PORH}$ ) とパワーオンリセット検出電圧 ( $V_{PORL}$ ) は、相対的に変動するため検出電圧が逆転する事はありません。
- 注 3) 電源電圧が  $V_{PORL}$  以下となるとパワーオンリセットがかかります。
- 注 4) 電圧検出回路 (VLTD) は、パワーオンリセット発生により初期化 (動作禁止) となります。



22.7.2 外部リセットを使用する場合

22.7.2.1 外部リセットの時間が POR 時間よりも短い場合

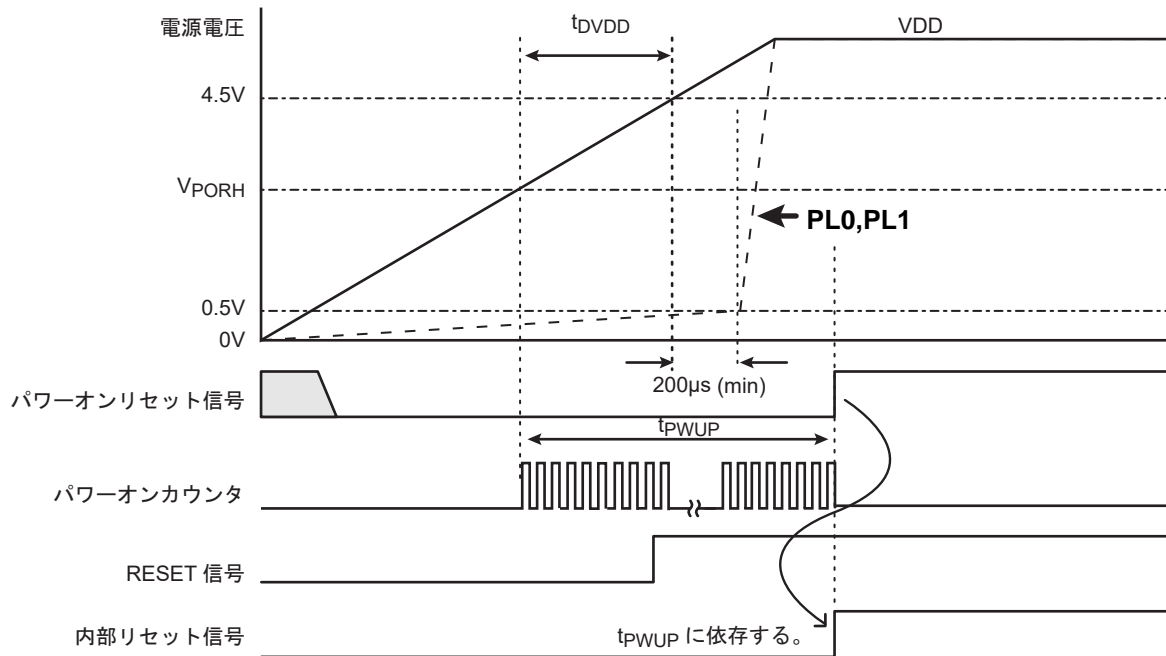


図 22-3 電源投入シーケンス (POR+ 外部リセット)(1)

注) VDD = DVDD5 = RVDD5 = AVDD5A = AVDD5B

22.7.2.2 外部リセットの時間が POR 時間よりも長い場合

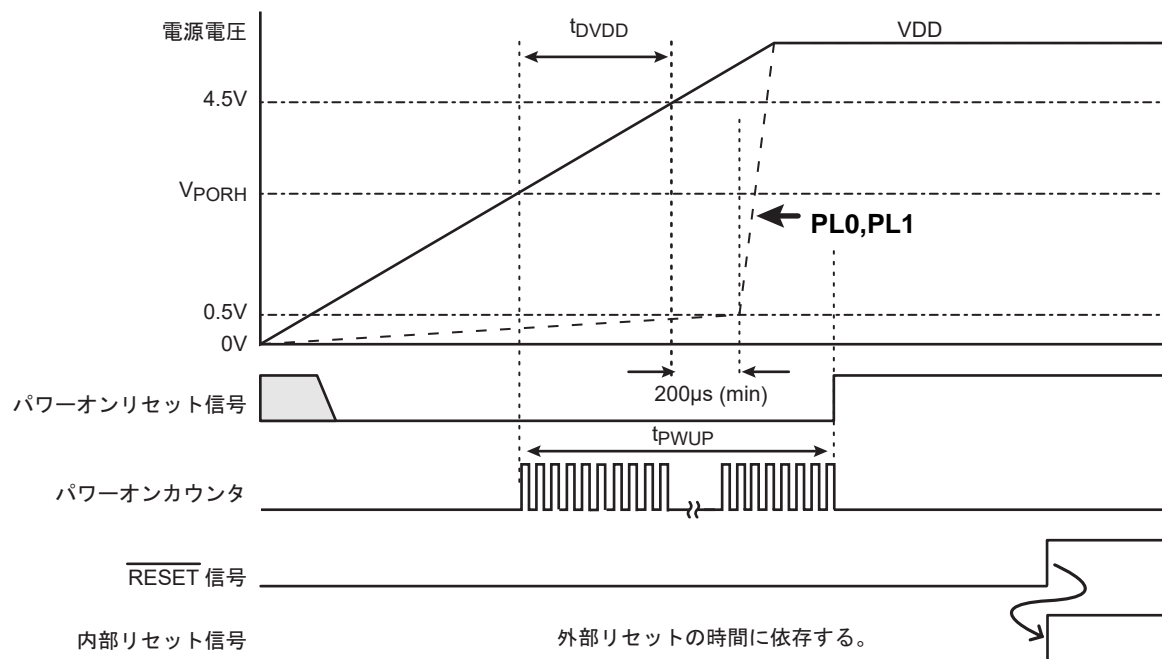


図 22-4 電源投入シーケンス (POR+ 外部リセット)(2)

注) VDD = DVDD5 = RVDD5 = AVDD5A = AVDD5B

22.7.2.3 電源の立ち上がり時間が $t_{PWUP}$ よりも長い場合

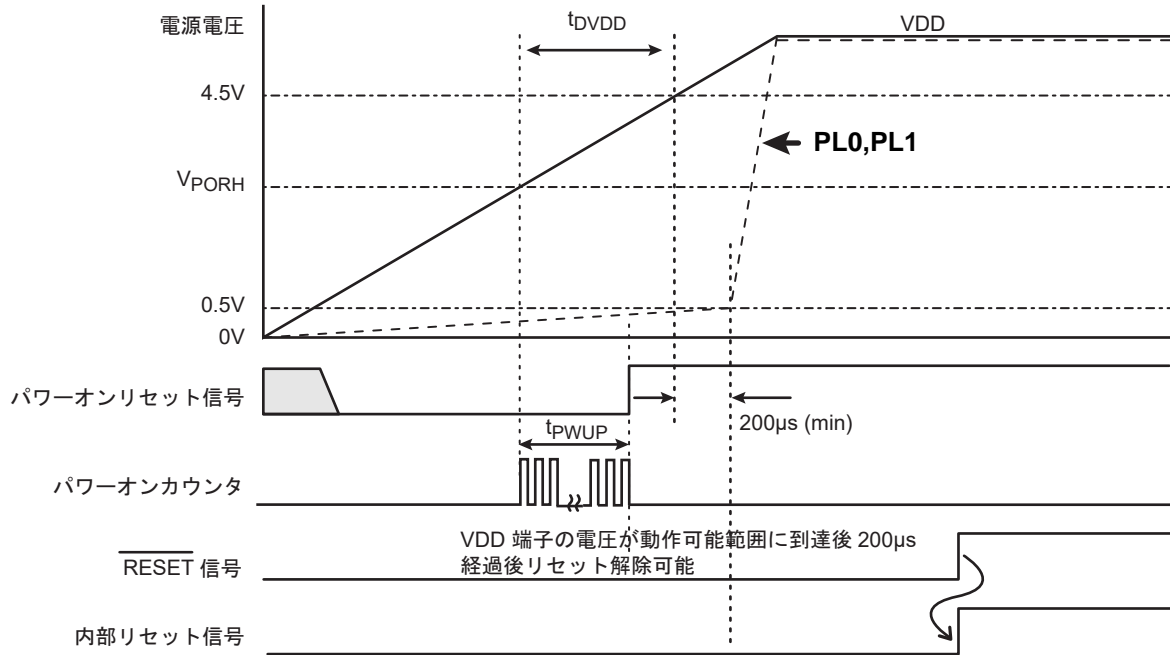


図 22-5 電源投入シーケンス ( $t_{DVDD} > t_{PWUP}$ )

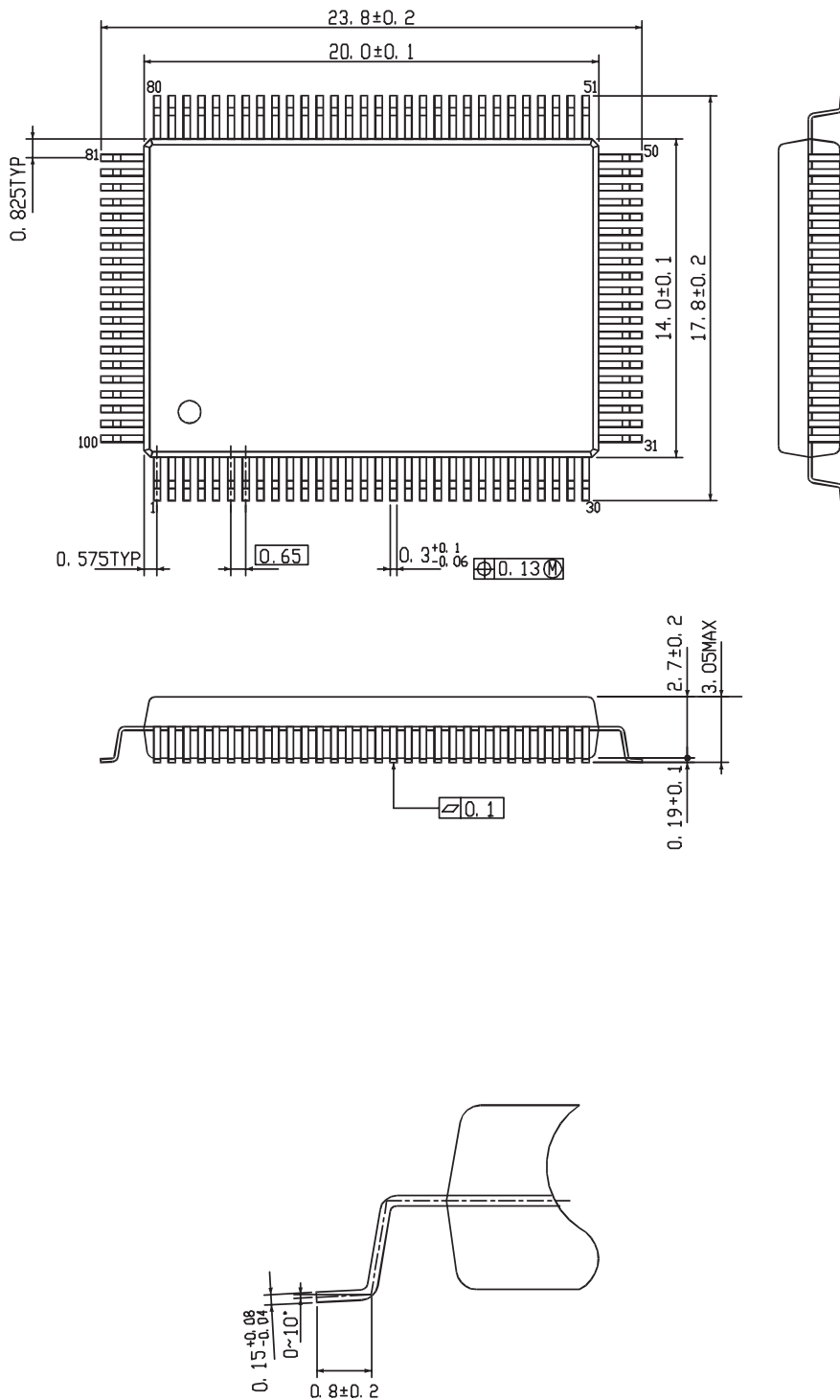
- 注 1)  $VDD = DVDD5 = RVDD5 = AVDD5A = AVDD5B$
- 注 2) 必ず RESET 端子から外部リセットをかけてください。



# 第 23 章 パッケージ寸法図

## 23.1 パッケージ型名 : QFP100-P-1420-0.65Q

Unit: mm





## 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ェア、ソフトウェ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、お客様のハードウェア・ソフトウェア・システムに必要な安全設計を行うことをお願いします。なお、設計および使用に際しては、本製品に関する最新の情報（本資料、仕様書、データシート、アプリケーションノート、半導体信頼性ハンドブックなど）および本製品が使用される機器の取扱説明書、操作説明書などをご確認の上、これに従ってください。また、上記資料などに記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を使用する場合は、お客様の製品単独およびシステム全体で十分に評価し、お客様の責任において適用可否を判断してください。
- 本製品は、特別に高い品質・信頼性が要求され、またはその故障や誤作動が生命・身体に危害を及ぼす恐れ、膨大な財産損害を引き起こす恐れ、もしくは社会に深刻な影響を及ぼす恐れのある機器（以下“特定用途”という）に使用されることは意図されていませんし、保証もされていません。特定用途には原子力関連機器、航空・宇宙機器、医療機器（ヘルスケア除く）、車載・輸送機器、列車・船舶機器、交通信号機器、燃焼・爆発制御機器、各種安全関連機器、昇降機器、発電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。