

ADJ-602-268

# H8/3068 F-ZTAT™ ハードウェアマニュアル

H8/3068 F-ZTAT™ HD64F3068

---

H8/3068 F-ZTAT™ ハードウェアマニュアル

発行年月日 2001年3月 第1版

発行 株式会社 日立製作所  
半導体グループ電子統括営業本部

編集 株式会社 日立小平セミコン  
技術ドキュメントグループ

©株式会社 日立製作所 2001

---

## ご注意

1. 本書に記載の製品及び技術のうち「外国為替及び外国貿易法」に基づき安全保障貿易管理関連貨物・技術に該当するものを輸出する場合、または国外に持ち出す場合は日本国政府の許可が必要です。
2. 本書に記載された情報の使用に際して、弊社もしくは第三者の特許権、著作権、商標権、その他の知的所有権等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。また本書に記載された情報を使用した事により第三者の知的所有権等の権利に関わる問題が生じた場合、弊社はその責を負いませんので予めご了承ください。
3. 製品及び製品仕様は予告無く変更する場合がありますので、最終的な設計、ご購入、ご使用に際しましては、事前に最新の製品規格または仕様書をお求めになりご確認ください。
4. 弊社は品質・信頼性の向上に努めておりますが、宇宙、航空、原子力、燃焼制御、運輸、交通、各種安全装置、ライフサポート関連の医療機器等のように、特別な品質・信頼性が要求され、その故障や誤動作が直接人命を脅かしたり、人体に危害を及ぼす恐れのある用途にご使用をお考えのお客様は、事前に弊社営業担当迄ご相談をお願い致します。
5. 設計に際しては、特に最大定格、動作電源電圧範囲、放熱特性、実装条件及びその他諸条件につきましては、弊社保証範囲内でご使用いただきますようお願い致します。  
保証値を越えてご使用された場合の故障及び事故につきましては、弊社はその責を負いません。  
また保証値内のご使用であっても半導体製品について通常予測される故障発生率、故障モードをご考慮の上、弊社製品の動作が原因でご使用機器が人身事故、火災事故、その他の拡大損害を生じないようにフェールセーフ等のシステム上の対策を講じて頂きますようお願い致します。
6. 本製品は耐放射線設計をしておりません。
7. 本書の一部または全部を弊社の文書による承認なしに転載または複製することを堅くお断り致します。
8. 本書をはじめ弊社半導体についてのお問い合わせ、ご相談は弊社営業担当迄お願い致します。

---

# はじめに

---

H8/3068F は、内部 32 ビット構成の H8/300H CPU を核にして、システム構成に必要な周辺機能を集積した高性能シングルチップマイクロコンピュータです。

H8/300H CPU は、内部 32 ビット構成で 16 ビット×16 本の汎用レジスタと高速動作を指向した簡潔で最適化された命令セットを備えており、16M バイトのリニアなアドレス空間を扱うことができます。

周辺機能として、ROM、RAM、16 ビットタイマ、8 ビットタイマ、プログラマブルタイミングパターンコントローラ (TPC)、ウォッチドッグタイマ (WDT)、シリアルコミュニケーションインタフェース (SCI)、A/D 変換器、D/A 変換器、I/O ポート、DMA コントローラ (DMAC) などを内蔵しています。3 チャンネルの SCI は、ISO/IEC7816-3 に準拠したスマートカードインタフェースを拡張機能としてサポートしています。また、電池駆動時の消費電力を低減するため、モジュール単位のスタンバイ機能やチップに供給するシステムクロックの分周比をプログラマブルに変更する機能を追加しています。

アドレス空間は 8 つのエリアに分割されており、エリアごとにデータバス幅、アクセスステートを選択でき、各種のメモリを容易に接続することができます。MCU 動作モードは、モード 1~7 があり、データバス幅の初期値とアドレス空間を選択することができます。

このため、H8/3068F を用いることにより高性能かつ小型のシステムを容易に実現することができます。

H8/3068F は、フラッシュメモリを内蔵した F-ZTAT™\* 版で、基板実装後のプログラム書換えを可能にしています。変化の激しい市場ニーズに即応し、フレキシブルな製品開発が実現できます。

本マニュアルは、H8/3068F のハードウェアについて説明します。命令の詳細については、「H8/300H シリーズ プログラミングマニュアル」をあわせてご覧ください。

【注】 \* F-ZTAT™は (株) 日立製作所の商標です。

---

# 目次

---

## 第1章 概要

1.1	概要	3
1.2	内部ブロック図	8
1.3	端子説明	9
1.3.1	ピン配置図	9
1.3.2	端子機能	10
1.3.3	モード別ピン配置一覧	17

## 第2章 CPU

2.1	概要	25
2.1.1	特長	25
2.1.2	H8/300CPU との相違点	26
2.2	CPU 動作モード	27
2.3	アドレス空間	28
2.4	レジスタ構成	29
2.4.1	概要	29
2.4.2	汎用レジスタ	30
2.4.3	コントロールレジスタ	31
2.4.4	CPU 内部レジスタの初期値	33
2.5	データ構成	34
2.5.1	汎用レジスタのデータ構成	34
2.5.2	メモリ上でのデータ構成	35
2.6	命令セット	37
2.6.1	命令セットの概要	37
2.6.2	命令とアドレッシングモードの組み合わせ	37
2.6.3	命令の機能別一覧	39
2.6.4	命令の基本フォーマット	50

	2.6.5	ビット操作命令使用上の注意.....	51
2.7		アドレッシングモードと実効アドレスの計算方法.....	53
	2.7.1	アドレッシングモード.....	53
	2.7.2	実効アドレスの計算方法.....	56
2.8		処理状態.....	60
	2.8.1	概要.....	60
	2.8.2	プログラム実行状態.....	60
	2.8.3	例外処理状態.....	60
	2.8.4	例外処理の動作.....	63
	2.8.5	バス権解放状態.....	64
	2.8.6	リセット状態.....	64
	2.8.7	低消費電力状態.....	64
2.9		基本動作タイミング.....	66
	2.9.1	概要.....	66
	2.9.2	内蔵メモリアクセスタイミング.....	66
	2.9.3	内蔵周辺モジュールアクセスタイミング.....	67
	2.9.4	外部アドレス空間アクセスタイミング.....	68

### 第3章 MCU動作モード

3.1		概要.....	71
	3.1.1	動作モードの種類を選択.....	71
	3.1.2	レジスタ構成.....	72
3.2		モードコントロールレジスタ (MDCR).....	73
3.3		システムコントロールレジスタ (SYSCR).....	74
3.4		各動作モードの説明.....	77
	3.4.1	モード 1.....	77
	3.4.2	モード 2.....	77
	3.4.3	モード 3.....	77
	3.4.4	モード 4.....	77
	3.4.5	モード 5.....	77
	3.4.6	モード 6.....	78
	3.4.7	モード 7.....	78
3.5		各動作モードにおける端子機能.....	79
3.6		各動作モードのメモリマップ.....	80
	3.6.1	リザーブ領域について.....	80

## 第4章 例外処理

4.1	概要 .....	87
4.1.1	例外処理の種類と優先度 .....	87
4.1.2	例外処理の動作.....	87
4.1.3	例外処理要因とベクタテーブル .....	87
4.2	リセット.....	89
4.2.1	概要 .....	89
4.2.2	リセットシーケンス.....	89
4.2.3	リセット直後の割込み.....	91
4.3	割込み.....	92
4.4	トラップ命令.....	93
4.5	例外処理後のスタックの状態.....	94
4.6	スタック使用上の注意.....	95

## 第5章 割込みコントローラ

5.1	概要 .....	99
5.1.1	特長 .....	99
5.1.2	ブロック図 .....	100
5.1.3	端子構成.....	101
5.1.4	レジスタ構成.....	101
5.2	各レジスタの説明 .....	102
5.2.1	システムコントロールレジスタ (SYSCR) .....	102
5.2.2	インタラプトプライオリティレジスタ A、B (IPRA、IPRB) .....	103
5.2.3	IRQ ステータスレジスタ (ISR) .....	110
5.2.4	IRQ イネーブルレジスタ (IER) .....	111
5.2.5	IRQ センスコントロールレジスタ (ISCR) .....	112
5.3	割込み要因 .....	113
5.3.1	外部割込み .....	113
5.3.2	内部割込み .....	114
5.3.3	割込み例外処理ベクタテーブル .....	114
5.4	割込み動作 .....	117
5.4.1	割込み動作の流れ .....	117
5.4.2	割込み例外処理シーケンス.....	122
5.4.3	割込み応答時間.....	123
5.5	使用上の注意.....	124
5.5.1	割込みの発生とディスエーブルとの競合 .....	124
5.5.2	割込みの受け付けを禁止している命令.....	125

5.5.3	EEPMOV 命令実行中の割込み.....	125
-------	-----------------------	-----

## 第6章 バスコントローラ

6.1	概要 .....	129
6.1.1	特長 .....	129
6.1.2	ブロック図 .....	131
6.1.3	端子構成 .....	132
6.1.4	レジスタ構成 .....	133
6.2	各レジスタの説明 .....	134
6.2.1	バス幅コントロールレジスタ (ABWCR) .....	134
6.2.2	アクセスステートコントロールレジスタ (ASTCR) .....	135
6.2.3	ウェイトコントロールレジスタ H、L (WCRH、WCRL) .....	136
6.2.4	バスリリースコントロールレジスタ (BRCR) .....	140
6.2.5	バスコントロールレジスタ (BCR) .....	143
6.2.6	チップセレクトコントロールレジスタ (CSCR) .....	146
6.2.7	DRAM コントロールレジスタ A (DRCRA) .....	147
6.2.8	DRAM コントロールレジスタ B (DRCRB) .....	149
6.2.9	リフレッシュタイマコントロール/ステータスレジスタ (RTMCSR) .....	152
6.2.10	リフレッシュタイマカウンタ (RTCNT) .....	154
6.2.11	リフレッシュタイムコンスタントレジスタ (RTCOR) .....	154
6.2.12	アドレスコントロールレジスタ (ADRCCR) .....	155
6.3	動作説明 .....	156
6.3.1	エリア分割 .....	156
6.3.2	バス仕様 .....	158
6.3.3	メモリアンタフェース .....	159
6.3.4	チップセレクト信号 .....	160
6.3.5	アドレス出力方式 .....	161
6.4	基本バスインタフェース .....	163
6.4.1	概要 .....	163
6.4.2	データサイズとデータアライメント .....	163
6.4.3	有効ストロブ .....	165
6.4.4	各エリアの説明 .....	166
6.4.5	基本バス制御信号タイミング .....	168
6.4.6	ウェイト制御 .....	176
6.5	DRAM インタフェース .....	178
6.5.1	概要 .....	178
6.5.2	DRAM 空間と RAS 出力端子の設定 .....	178

6.5.3	アドレスマルチプレクス .....	179
6.5.4	データバス .....	180
6.5.5	DRAM インタフェース使用端子 .....	180
6.5.6	基本タイミング .....	181
6.5.7	プリチャージステート制御 .....	182
6.5.8	ウェイト制御 .....	183
6.5.9	バイトアクセス制御と CAS 出力端子 .....	185
6.5.10	バースト動作 .....	187
6.5.11	リフレッシュ制御 .....	192
6.5.12	使用例 .....	196
6.5.13	使用上の注意 .....	201
6.6	インターバルタイム .....	203
6.6.1	動作説明 .....	203
6.7	割込み要因 .....	209
6.8	バースト ROM インタフェース .....	209
6.8.1	概要 .....	209
6.8.2	基本タイミング .....	209
6.8.3	ウェイト制御 .....	210
6.9	アイドルサイクル .....	211
6.9.1	動作説明 .....	211
6.9.2	アイドルサイクルでの端子状態 .....	214
6.10	バスアービタ .....	215
6.10.1	動作説明 .....	215
6.11	レジスタと端子入力のタイミング .....	218
6.11.1	レジスタライトタイミング .....	218
6.11.2	BREQ 端子の入力タイミング .....	219

## 第7章 DMA コントローラ

7.1	概要 .....	223
7.1.1	特長 .....	223
7.1.2	ブロック図 .....	224
7.1.3	機能概要 .....	225
7.1.4	端子構成 .....	227
7.1.5	レジスタ構成 .....	227
7.2	各レジスタの説明 (1) (ショートアドレスモード) .....	229
7.2.1	メモリアドレスレジスタ (MAR) .....	230
7.2.2	I/O アドレスレジスタ (IOAR) .....	231



	7.2.3	転送カウントレジスタ (ETCR) .....	231
	7.2.4	データトランスファコントロールレジスタ (DTCR) .....	233
7.3		各レジスタの説明 (2) (フルアドレスモード) .....	237
	7.3.1	メモリアドレスレジスタ (MAR) .....	237
	7.3.2	I/O アドレスレジスタ (IOAR) .....	237
	7.3.3	転送カウントレジスタ (ETCR) .....	238
	7.3.4	データトランスファコントロールレジスタ (DTCR) .....	240
7.4		動作説明.....	248
	7.4.1	概要.....	248
	7.4.2	I/O モード.....	250
	7.4.3	アイドルモード.....	253
	7.4.4	リピートモード.....	255
	7.4.5	ノーマルモード.....	260
	7.4.6	ブロック転送モード.....	263
	7.4.7	DMAC の起動要因.....	268
	7.4.8	DMAC のバスサイクル.....	270
	7.4.9	DMAC 複数チャネルの動作 .....	274
	7.4.10	外部バス権要求、DRAM インタフェースと DMAC の関係.....	275
	7.4.11	NMI 割込みと DMAC.....	276
	7.4.12	DMAC 動作の強制終了.....	277
	7.4.13	フルアドレスモードの解除.....	277
	7.4.14	リセット、スタンバイモード、スリープモードの DMAC の状態 .....	278
7.5		割込み.....	279
7.6		使用上の注意.....	280
	7.6.1	ワードデータ転送時の注意.....	280
	7.6.2	DMAC による DMAC 自体のアクセス.....	280
	7.6.3	MAR のロングワードアクセス.....	280
	7.6.4	フルアドレスモード設定時の注意.....	280
	7.6.5	内部割込みで DMAC を起動する場合の注意 .....	281
	7.6.6	NMI 割込みとブロック転送モード.....	282
	7.6.7	MAR、IOAR のアドレス指定.....	283
	7.6.8	転送中断時のバスサイクル.....	283
	7.6.9	A/D 変換器による転送要求.....	284

## 第 8 章 I/O ポート

8.1	概要 .....	287
8.2	ポート 1.....	292

	8.2.1	概要 .....	292
	8.2.2	レジスタ構成 .....	292
8.3	ポート 2 .....		295
	8.3.1	概要 .....	295
	8.3.2	レジスタ構成 .....	296
8.4	ポート 3 .....		299
	8.4.1	概要 .....	299
	8.4.2	レジスタ構成 .....	299
8.5	ポート 4 .....		302
	8.5.1	概要 .....	302
	8.5.2	レジスタ構成 .....	303
8.6	ポート 5 .....		306
	8.6.1	概要 .....	306
	8.6.2	レジスタ構成 .....	306
8.7	ポート 6 .....		310
	8.7.1	概要 .....	310
	8.7.2	レジスタ構成 .....	310
8.8	ポート 7 .....		315
	8.8.1	概要 .....	315
	8.8.2	レジスタ構成 .....	315
8.9	ポート 8 .....		317
	8.9.1	概要 .....	317
	8.9.2	レジスタ構成 .....	318
8.10	ポート 9 .....		324
	8.10.1	概要 .....	324
	8.10.2	レジスタ構成 .....	325
8.11	ポート A .....		329
	8.11.1	概要 .....	329
	8.11.2	レジスタ構成 .....	331
8.12	ポート B .....		341
	8.12.1	概要 .....	341
	8.12.2	レジスタ構成 .....	343

## 第9章 16ビットタイマ

9.1	概要 .....	355	
	9.1.1	特長 .....	355
	9.1.2	ブロック図 .....	357

	9.1.3	端子構成.....	360
	9.1.4	レジスタ構成.....	361
9.2		各レジスタの説明.....	362
	9.2.1	タイマスタートレジスタ ( TSTR ) .....	362
	9.2.2	タイマシンクロレジスタ ( TSNC ) .....	363
	9.2.3	タイマモードレジスタ ( TMDR ) .....	365
	9.2.4	タイマインタラプトステータスレジスタ A ( TISRA ) .....	368
	9.2.5	タイマインタラプトステータスレジスタ B ( TISRB ) .....	371
	9.2.6	タイマインタラプトステータスレジスタ C ( TISRC ) .....	374
	9.2.7	タイマカウンタ ( 16TCNT ) .....	377
	9.2.8	ジェネラルレジスタ A、B ( GRA、GRB ) .....	378
	9.2.9	タイマコントロールレジスタ ( 16TCR ) .....	379
	9.2.10	タイマ I/O コントロールレジスタ ( TIOR ) .....	382
	9.2.11	タイマアウトプットレベルセットレジスタ ( TOLR ) .....	385
9.3		CPU とのインタフェース.....	387
	9.3.1	16 ビットアクセス可能なレジスタ.....	387
	9.3.2	8 ビットアクセスのレジスタ.....	389
9.4		動作説明.....	390
	9.4.1	概要.....	390
	9.4.2	基本機能.....	391
	9.4.3	同期動作.....	401
	9.4.4	PWM モード.....	403
	9.4.5	位相計数モード.....	407
	9.4.6	16 ビットタイマ出力初期値の設定.....	409
9.5		割込み.....	410
	9.5.1	ステータスフラグのセットタイミング.....	410
	9.5.2	ステータスフラグのクリアタイミング.....	412
	9.5.3	割込み要因と DMA コントローラの起動.....	413
9.6		使用上の注意.....	414

## 第 10 章 8 ビットタイマ

10.1		概要.....	429
	10.1.1	特長.....	429
	10.1.2	ブロック図.....	430
	10.1.3	端子構成.....	431
	10.1.4	レジスタ構成.....	432
10.2		各レジスタの説明.....	433

10.2.1	タイマカウンタ (8TCNT) .....	433
10.2.2	タイムコンスタントレジスタ A (TCORA) .....	434
10.2.3	タイムコンスタントレジスタ B (TCORB) .....	435
10.2.4	タイマコントロールレジスタ (8TCR) .....	436
10.2.5	タイマコントロール/ステータスレジスタ (8TCSR) .....	439
10.3	CPU とのインタフェース.....	444
10.3.1	8ビットレジスタ .....	444
10.4	動作説明.....	446
10.4.1	8TCNT のカウントタイミング.....	446
10.4.2	コンペアマッチタイミング.....	447
10.4.3	インプットキャプチャ信号タイミング .....	449
10.4.4	ステータスフラグのセットタイミング .....	450
10.4.5	カスケード接続時の動作 .....	451
10.4.6	インプットキャプチャの設定.....	454
10.5	割込み.....	456
10.5.1	割込み要因 .....	456
10.5.2	A/D 変換器の起動.....	456
10.6	8ビットタイマの使用例 .....	457
10.7	使用上の注意.....	458
10.7.1	8TCNT のライトとクリアの競合 .....	458
10.7.2	8TCNT のライトとカウントアップの競合 .....	459
10.7.3	TCOR のライトとコンペアマッチの競合 .....	460
10.7.4	TCOR のリードとインプットキャプチャの競合 .....	461
10.7.5	インプットキャプチャによるカウンタクリアとカウントアップの競合.....	461
10.7.6	TCOR のライトとインプットキャプチャの競合 .....	462
10.7.7	16ビットカウントモード (カスケード接続時) の8TCNT のバイトライトと カウントアップの競合.....	463
10.7.8	コンペアマッチ A、B の競合.....	463
10.7.9	内部クロックの切り替えと8TCNT の動作 .....	464

## 第 11 章 プログラマブルタイミングパターンコントローラ (TPC)

11.1	概要 .....	469
11.1.1	特長 .....	469
11.1.2	ブロック図 .....	470
11.1.3	端子構成.....	471
11.1.4	レジスタ構成.....	472
11.2	各レジスタの説明 .....	473

11.2.1	ポート A データディレクションレジスタ (PADDR) .....	473
11.2.2	ポート A データレジスタ (PADR) .....	473
11.2.3	ポート B データディレクションレジスタ (PBDDR) .....	474
11.2.4	ポート B データレジスタ (PBDR) .....	474
11.2.5	ネクストデータレジスタ A (NDRA) .....	475
11.2.6	ネクストデータレジスタ B (NDRB) .....	477
11.2.7	ネクストデータイネーブルレジスタ A (NDERA) .....	479
11.2.8	ネクストデータイネーブルレジスタ B (NDERB) .....	480
11.2.9	TPC 出力コントロールレジスタ (TPCR) .....	481
11.2.10	TPC 出力モードレジスタ (TPMR) .....	484
11.3	動作説明.....	487
11.3.1	概要.....	487
11.3.2	出力タイミング.....	488
11.3.3	TPC 出力通常動作.....	489
11.3.4	TPC 出力ノンオーバーラップ動作.....	491
11.3.5	インプットキャプチャによる TPC 出力 .....	493
11.4	使用上の注意.....	494
11.4.1	TPC 出力端子の動作.....	494
11.4.2	ノンオーバーラップ動作時の注意 .....	494

## 第 12 章 ウォッチドッグタイマ

12.1	概要 .....	499
12.1.1	特長 .....	499
12.1.2	ブロック図 .....	500
12.1.3	レジスタ構成.....	500
12.2	各レジスタの説明 .....	501
12.2.1	タイマカウンタ (TCNT) .....	501
12.2.2	タイマコントロール/ステータスレジスタ (TCSR) .....	501
12.2.3	リセットコントロール/ステータスレジスタ (RSTCSR) .....	504
12.2.4	レジスタ書換え時の注意 .....	505
12.3	動作説明.....	507
12.3.1	ウォッチドッグタイマ時の動作 .....	507
12.3.2	インターバルタイマ時の動作.....	508
12.3.3	オーバフローフラグ (OVF) セットタイミング.....	508
12.3.4	ウォッチドッグタイマリセット (WRST) のセットタイミング .....	509
12.4	割込み.....	510
12.5	使用上の注意.....	510

## 第13章 SCI

13.1	概要 .....	513
	13.1.1 特長 .....	513
	13.1.2 ブロック図 .....	515
	13.1.3 端子構成 .....	516
	13.1.4 レジスタ構成 .....	517
13.2	各レジスタの説明 .....	518
	13.2.1 レシーブシフトレジスタ (RSR) .....	518
	13.2.2 レシーブデータレジスタ (RDR) .....	518
	13.2.3 トランスミットシフトレジスタ (TSR) .....	519
	13.2.4 トランスミットデータレジスタ (TDR) .....	519
	13.2.5 シリアルモードレジスタ (SMR) .....	520
	13.2.6 シリアルコントロールレジスタ (SCR) .....	524
	13.2.7 シリアルステータスレジスタ (SSR) .....	529
	13.2.8 ビットレートレジスタ (BRR) .....	536
13.3	動作説明 .....	544
	13.3.1 概要 .....	544
	13.3.2 調歩同期式モード時の動作 .....	547
	13.3.3 マルチプロセッサ通信機能 .....	557
	13.3.4 クロック同期式モード時の動作 .....	563
13.4	SCI 割込み .....	572
13.5	使用上の注意 .....	573
	13.5.1 SCI を使用する際の注意 .....	573

## 第14章 スマートカードインタフェース

14.1	概要 .....	581
	14.1.1 特長 .....	581
	14.1.2 ブロック図 .....	582
	14.1.3 端子構成 .....	582
	14.1.4 レジスタ構成 .....	583
14.2	各レジスタの説明 .....	584
	14.2.1 スマートカードモードレジスタ (SCMR) .....	584
	14.2.2 シリアルステータスレジスタ (SSR) .....	586
	14.2.3 シリアルモードレジスタ (SMR) .....	587
	14.2.4 シリアルコントロールレジスタ (SCR) .....	589
14.3	動作説明 .....	590
	14.3.1 概要 .....	590

14.3.2	端子接続.....	590
14.3.3	データフォーマット.....	591
14.3.4	レジスタの設定.....	593
14.3.5	クロック.....	595
14.3.6	データの送信 / 受信動作.....	597
14.4	使用上の注意.....	606

## 第 15 章 A/D 変換器

15.1	概要.....	613
15.1.1	特長.....	613
15.1.2	ブロック図.....	614
15.1.3	端子構成.....	615
15.1.4	レジスタ構成.....	616
15.2	各レジスタの説明.....	617
15.2.1	A/D データレジスタ A ~ D ( ADDRA ~ D ) .....	617
15.2.2	A/D コントロール / ステータスレジスタ ( ADCSR ) .....	618
15.2.3	A/D コントロールレジスタ ( ADCR ) .....	621
15.3	CPU とのインタフェース.....	622
15.4	動作説明.....	623
15.4.1	単一モード ( SCAN = 0 ) .....	623
15.4.2	スキャンモード ( SCAN = 1 ) .....	625
15.4.3	入力サンプリングと A/D 変換時間.....	627
15.4.4	外部トリガ入力タイミング.....	628
15.5	割込み.....	629
15.6	使用上の注意.....	629

## 第 16 章 D/A 変換器

16.1	概要.....	637
16.1.1	特長.....	637
16.1.2	ブロック図.....	637
16.1.3	端子構成.....	638
16.1.4	レジスタ構成.....	638
16.2	各レジスタの説明.....	639
16.2.1	D/A データレジスタ 0, 1 ( DADR0, 1 ) .....	639
16.2.2	D/A コントロールレジスタ ( DACR ) .....	640
16.2.3	D/A スタンバイコントロールレジスタ ( DASTCR ) .....	642

16.3	動作説明.....	643
16.4	D/A 出力制御.....	644

## 第 17 章 RAM

17.1	概要.....	647
	17.1.1    ブロック図.....	647
	17.1.2    レジスタ構成.....	648
17.2	システムコントロールレジスタ (SYSCR).....	649
17.3	動作説明.....	650

## 第 18 章 フラッシュメモリ

18.1	概要.....	653
18.2	特長.....	654
	18.2.1    ブロック図.....	655
	18.2.2    端子構成.....	655
	18.2.3    レジスタ構成.....	656
18.3	レジスタの説明.....	657
	18.3.1    フラッシュメモリコントロールレジスタ 1 (FLMCR1).....	657
	18.3.2    フラッシュメモリコントロールレジスタ 2 (FLMCR2).....	660
	18.3.3    消去ブロック指定レジスタ 1 (EBR1).....	661
	18.3.4    消去ブロック指定レジスタ 2 (EBR2).....	662
	18.3.5    RAM コントロールレジスタ (RAMCR).....	663
18.4	動作の概要.....	665
	18.4.1    モード遷移図.....	665
	18.4.2    オンボードプログラミングモード.....	666
	18.4.3    RAM によるフラッシュメモリのエミュレーション.....	668
	18.4.4    ブロック分割法.....	669
18.5	オンボードプログラミングモード.....	670
	18.5.1    ブートモード.....	671
	18.5.2    ユーザプログラムモード.....	676
18.6	フラッシュメモリの書き込み / 消去.....	678
	18.6.1    プログラムモード.....	679
	18.6.2    プログラムベリファイモード.....	680
	18.6.3    イレースモード.....	685
	18.6.4    イレースベリファイモード.....	685
18.7	プロテクト.....	687



18.7.1	ハードウェアプロテクト	687
18.7.2	ソフトウェアプロテクト	688
18.7.3	エラープロテクト	689
18.8	RAMによるフラッシュメモリのエミュレーション	691
18.9	NMI 入力禁止の条件	694
18.10	フラッシュメモリの PROM モード	695
18.10.1	ソケットアダプタとメモリマップ	695
18.10.2	PROM モード使用時の注意事項	695
18.11	フラッシュメモリの書き込み / 消去時の注意	697

## 第 19 章 クロック発振器

19.1	概要	705
19.1.1	ブロック図	705
19.2	発振器	706
19.2.1	水晶発振子を接続する方法	706
19.2.2	外部クロックを入力する方法	708
19.3	デューティ補正回路	711
19.4	プリスケアラ	711
19.5	分周器	711
19.5.1	レジスタ構成	711
19.5.2	分周比コントロールレジスタ (DIVCR)	711
19.5.3	使用上の注意	712

## 第 20 章 低消費電力状態

20.1	概要	715
20.2	レジスタ構成	717
20.2.1	システムコントロールレジスタ (SYSCR)	717
20.2.2	モジュールスタンバイコントロールレジスタ H (MSTCRH)	720
20.2.3	モジュールスタンバイコントロールレジスタ L (MSTCRL)	722
20.3	スリープモード	724
20.3.1	スリープモードへの遷移	724
20.3.2	スリープモードの解除	724
20.4	ソフトウェアスタンバイモード	725
20.4.1	ソフトウェアスタンバイモードへの遷移	725
20.4.2	ソフトウェアスタンバイモードの解除	725
20.4.3	ソフトウェアスタンバイモード解除後の発振安定待機時間の設定	726

20.4.4	ソフトウェアスタンバイモードの応用例	728
20.4.5	使用上の注意	728
20.5	ハードウェアスタンバイモード	729
20.5.1	ハードウェアスタンバイモードへの遷移	729
20.5.2	ハードウェアスタンバイモードの解除	729
20.5.3	ハードウェアスタンバイモードのタイミング	730
20.6	モジュールスタンバイ機能	731
20.6.1	モジュールスタンバイタイミング	731
20.6.2	モジュールスタンバイ中のリード/ライト	731
20.6.3	使用上の注意	731
20.7	クロック出力禁止機能	733

## 第 21 章 電気的特性（暫定仕様）

21.1	H8/3068F-ZTAT 電気的特性	737
21.1.1	絶対最大定格	737
21.1.2	DC 特性	738
21.1.3	AC 特性	743
21.1.4	A/D 変換特性	748
21.1.5	D/A 変換特性	749
21.1.6	フラッシュメモリ特性	750
21.2	動作タイミング	754
21.2.1	クロックタイミング	754
21.2.2	制御信号タイミング	755
21.2.3	バスタイミング	757
21.2.4	DRAM インタフェースバスタイミング	760
21.2.5	TPC、I/O ポートタイミング	763
21.2.6	タイマ入出力タイミング	764
21.2.7	SCI 入出力タイミング	765
21.2.8	DMAC タイミング	766

## 付録

A.	命令	769
A.1	命令一覧	769
A.2	オペレーションコードマップ	784
A.3	命令実行ステート数	787
B.	内部 I/O レジスタ一覧	798

	B.1	アドレス一覧 (EMC ビット=0 のとき) .....	798
	B.2	アドレス一覧 (EMC ビット=1 のとき) .....	809
	B.3	機能一覧.....	834
C.	I/O ポートブロック図.....		926
	C.1	ポート 1 ブロック図 .....	926
	C.2	ポート 2 ブロック図 .....	927
	C.3	ポート 3 ブロック図 .....	928
	C.4	ポート 4 ブロック図 .....	929
	C.5	ポート 5 ブロック図 .....	930
	C.6	ポート 6 ブロック図 .....	931
	C.7	ポート 7 ブロック図 .....	938
	C.8	ポート 8 ブロック図 .....	939
	C.9	ポート 9 ブロック図 .....	944
	C.10	ポート A ブロック図.....	950
	C.11	ポート B ブロック図.....	953
D.	端子状態.....		961
	D.1	各処理状態におけるポートの状態.....	961
	D.2	リセット時の端子状態.....	968
E.	ハードウェアスタンバイモード遷移 / 復帰時のタイミングについて.....		972
F.	型名一覧.....		973
G.	外形寸法図 .....		974
H	H8/300H シリーズ製品仕様比較.....		976
	H.1	H8/3068F と H8/3067、H8/3062 シリーズ、H8/3048 シリーズの相異点.....	976
	H.2	100 ピンパッケージ品の端子機能比較 (FP-100B、TFP-100B の場合) .....	978

---

# 1. 概要

---

## 第1章 目次

1.1 概要.....	3
1.2 内部ブロック図.....	8
1.3 端子説明.....	9
1.3.1 ピン配置図.....	9
1.3.2 端子機能.....	10
1.3.3 モード別ピン配置一覧.....	17



## 1.1 概要

H8/3068F は、日立オリジナルアーキテクチャを採用した H8/300H CPU を核にして、システム構成に必要な周辺機能を集積したシングルチップマイクロコンピュータ (MCU) です。

H8/300H CPU は、内部 32 ビット構成で 16 ビット×16 本の汎用レジスタと高速動作を指向した簡潔で最適化された命令セットを備えており、16M バイトのリニアなアドレス空間を扱うことができます。また、H8/300CPU の命令に対しオブジェクトレベルで上位互換を保っていますので、H8/300 シリーズから容易に移行することができます。

システム構成に必要な周辺機能としては、ROM、RAM、16 ビットタイマ、8 ビットタイマ、プログラマブルタイミングパターンコントローラ (TPC)、ウォッチドッグタイマ (WDT)、シリアルコミュニケーションインタフェース (SCI)、A/D 変換器、D/A 変換器、I/O ポート、DMA コントローラ (DMAC) などを内蔵しています。

H8/3068F には、384k バイトフラッシュメモリと 16k バイト RAM が内蔵されています。

MCU 動作モードは、モード 1~7 (シングルチップモード 2 種類、拡張モード 5 種類) があり、データバス幅とアドレス空間を選択することができます。

H8/3068F は、ユーザサイドで自由にプログラムの書換えができるフラッシュメモリを内蔵した F-ZTAT™\*版です。仕様流動性の高い応用機器、さらに量産初期から本格的量産など、ユーザの状況に応じて迅速かつ柔軟な対応が可能です。

H8/3068F の特長を表 1.1 に示します。

【注】 \* F-ZTAT™ は (株) 日立製作所の商標です。

表 1.1 特長

項目	仕様
CPU	<p>H8/300CPU に対してオブジェクトレベルで上位互換</p> <p>汎用レジスタマシン</p> <ul style="list-style-type: none"> <li>・ 汎用レジスタ : 16 ビット × 16 本 ( 8 ビット × 16 本 + 16 ビット × 8 本、32 ビット × 8 本としても使用可能 )</li> </ul> <p>高速動作</p> <ul style="list-style-type: none"> <li>・ 最大動作周波数 : 25MHz</li> <li>・ 加減算 : 80ns</li> <li>・ 乗除算 : 560ns</li> </ul> <p>アドレス空間 16M バイト</p> <p>特長ある命令</p> <ul style="list-style-type: none"> <li>・ 8 / 16 / 32 ビット転送 ・ 演算命令</li> <li>・ 符号なし / 符号付き乗算命令 ( 8 ビット × 8 ビット、16 ビット × 16 ビット )</li> <li>・ 符号なし / 符号付き除算命令 ( 16 ビット ÷ 8 ビット、32 ビット ÷ 16 ビット )</li> <li>・ ビットアキュムレータ機能</li> <li>・ レジスタ間接指定によりビット番号を指定可能なビット操作命令</li> </ul>
メモリ	<p>H8/3068F</p> <ul style="list-style-type: none"> <li>・ ROM : 384k バイト</li> <li>・ RAM : 16k バイト</li> </ul>
割込みコントローラ	<ul style="list-style-type: none"> <li>・ 外部割込み端子 7 本 : NMI、<math>\overline{IRQ}_0 \sim \overline{IRQ}_5</math></li> <li>・ 内部割込み 36 要因</li> <li>・ 3 レベルの割込み優先順位が設定可能</li> </ul>

項目	仕様
バスコントローラ	<ul style="list-style-type: none"> <li>・ アドレス空間を 8 エリアに分割し、エリアごとに独立してバス仕様を設定可能</li> <li>・ エリア 0~7 に対してそれぞれチップセレクト出力可能</li> <li>・ エリアごとに 8 ビットアクセス空間 / 16 ビットアクセス空間を設定可能</li> <li>・ エリアごとに 2 ステートアクセス空間 / 3 ステートアクセス空間を設定可能</li> <li>・ 2 種類のウェイトモードを設定可能</li> <li>・ エリアごとにプログラムウェイトのステート数を設定可能</li> <li>・ バースト ROM を直接接続可能</li> <li>・ 最大 8M バイトの DRAM を直接接続可能(またはインターバルタイマとして使用可能)</li> <li>・ バス権調停機能</li> </ul>
DMA コントローラ (DMAC)	<p>ショートアドレスモード</p> <ul style="list-style-type: none"> <li>・ 最大 4 チャンネルを使用可能</li> <li>・ I/O モード / アイドルモード / リピートモードの選択可能</li> <li>・ 起動要因： 16 ビットタイマチャンネル 0~2 のコンペアマッチ / インพุットキャプチャ A 割込み、A/D 変換器の変換終了割込み、SCI の送信データエンプティ / 受信データフル割込み、外部リクエスト</li> </ul> <p>フルアドレスモード</p> <ul style="list-style-type: none"> <li>・ 最大 2 チャンネルを使用可能</li> <li>・ ノーマルモード / ブロック転送モードの選択可能</li> <li>・ 起動要因： 16 ビットタイマチャンネル 0~2 のコンペアマッチ / インพุットキャプチャ A 割込み、A/D 変換器の変換終了割込み、外部リクエスト、オートリクエスト</li> </ul>



1. 概要

項目	仕様
16ビットタイマ× 3チャンネル	<ul style="list-style-type: none"> <li>・ 16ビットタイマ3チャンネルを内蔵。最大6端子のパルス出力、または最大6種類のパルスの入力処理が可能</li> <li>・ 16ビットタイマカウンタ×1(チャンネル0~2)</li> <li>・ アウトプットコンペア出力/インプットキャプチャ入力(兼用端子)×2(チャンネル0~2)</li> <li>・ 同期動作可能(チャンネル0~2)</li> <li>・ PWMモード設定可能(チャンネル0~2)</li> <li>・ 位相計数モード設定可能(チャンネル2)</li> <li>・ コンペアマッチ/インプットキャプチャAの割込みによりDMAC起動可能(チャンネル0~2)</li> </ul>
8ビットタイマ× 4チャンネル	<ul style="list-style-type: none"> <li>・ 8ビットアップカウンタ(外部イベントカウント可能)</li> <li>・ タイムコンスタントレジスタ×2</li> <li>・ 2チャンネルの接続が可能</li> </ul>
プログラマブル タイミングパターン コントローラ (TPC)	<ul style="list-style-type: none"> <li>・ 16ビットタイマをタイムベースとした最大16ビットのパルス出力が可能</li> <li>・ 最大4ビット×4系統のパルス出力が可能(16ビット×1系統、8ビット×2系統などの設定も可能)</li> <li>・ ノンオーバーラップモード設定可能</li> <li>・ DMACによる出力データの転送可能</li> </ul>
ウォッチドッグタイ マ(WDT)×1チャ ネル	<ul style="list-style-type: none"> <li>・ オーバフローによりリセット信号を発生可能</li> <li>・ リセット信号の外部出力可能(ただし、F-ZTAT版は不可)、</li> <li>・ インターバルタイマとして使用可能</li> </ul>
シリアルコミュニケ ーションインタフェ ース(SCI)×3チャ ネル)	<ul style="list-style-type: none"> <li>・ 調歩同期/クロック同期式モードの選択可能</li> <li>・ 送受信同時動作(全二重動作)可能</li> <li>・ 専用のボーレートジェネレータ内蔵</li> <li>・ スマートカードインタフェース拡張機能内蔵</li> </ul>
A/D変換器	<ul style="list-style-type: none"> <li>・ 分解能:10ビット</li> <li>・ 8チャンネル:単一モード/スキャンモード選択可能</li> <li>・ アナログ変換電圧範囲の設定が可能</li> <li>・ サンプル&amp;ホールド機能付き</li> <li>・ 外部トリガまたは8ビットタイマのコンペアマッチによるA/D変換開始可能</li> <li>・ A/D変換終了割込みによるDMAC起動可能</li> </ul>

項目	仕様																																								
D/A 変換器	<ul style="list-style-type: none"> <li>・ 分解能 : 8 ビット</li> <li>・ 2 チャンネル</li> <li>・ ソフトウェアスタンバイモード時 D/A 出力保持可能</li> </ul>																																								
I/O ポート	<ul style="list-style-type: none"> <li>・ 入出力端子 70 本</li> <li>・ 入力端子 9 本</li> </ul>																																								
動作モード	<p>7 種類の MCU 動作モード</p> <table border="1"> <thead> <tr> <th>モード</th> <th>アドレス空間</th> <th>アドレス端子</th> <th>バス幅初期値</th> <th>バス幅最大値</th> </tr> </thead> <tbody> <tr> <td>モード 1</td> <td>1M バイト</td> <td>A<sub>19</sub> ~ A<sub>0</sub></td> <td>8 ビット</td> <td>16 ビット</td> </tr> <tr> <td>モード 2</td> <td>1M バイト</td> <td>A<sub>19</sub> ~ A<sub>0</sub></td> <td>16 ビット</td> <td>16 ビット</td> </tr> <tr> <td>モード 3</td> <td>16M バイト</td> <td>A<sub>23</sub> ~ A<sub>0</sub></td> <td>8 ビット</td> <td>16 ビット</td> </tr> <tr> <td>モード 4</td> <td>16M バイト</td> <td>A<sub>23</sub> ~ A<sub>0</sub></td> <td>16 ビット</td> <td>16 ビット</td> </tr> <tr> <td>モード 5</td> <td>16M バイト</td> <td>A<sub>23</sub> ~ A<sub>0</sub></td> <td>8 ビット</td> <td>16 ビット</td> </tr> <tr> <td>モード 6</td> <td>64K バイト</td> <td>-</td> <td>-</td> <td>-</td> </tr> <tr> <td>モード 7</td> <td>1M バイト</td> <td>-</td> <td>-</td> <td>-</td> </tr> </tbody> </table> <ul style="list-style-type: none"> <li>・ モード 1 ~ 4 では内蔵 ROM は無効となります。</li> </ul>	モード	アドレス空間	アドレス端子	バス幅初期値	バス幅最大値	モード 1	1M バイト	A <sub>19</sub> ~ A <sub>0</sub>	8 ビット	16 ビット	モード 2	1M バイト	A <sub>19</sub> ~ A <sub>0</sub>	16 ビット	16 ビット	モード 3	16M バイト	A <sub>23</sub> ~ A <sub>0</sub>	8 ビット	16 ビット	モード 4	16M バイト	A <sub>23</sub> ~ A <sub>0</sub>	16 ビット	16 ビット	モード 5	16M バイト	A <sub>23</sub> ~ A <sub>0</sub>	8 ビット	16 ビット	モード 6	64K バイト	-	-	-	モード 7	1M バイト	-	-	-
モード	アドレス空間	アドレス端子	バス幅初期値	バス幅最大値																																					
モード 1	1M バイト	A <sub>19</sub> ~ A <sub>0</sub>	8 ビット	16 ビット																																					
モード 2	1M バイト	A <sub>19</sub> ~ A <sub>0</sub>	16 ビット	16 ビット																																					
モード 3	16M バイト	A <sub>23</sub> ~ A <sub>0</sub>	8 ビット	16 ビット																																					
モード 4	16M バイト	A <sub>23</sub> ~ A <sub>0</sub>	16 ビット	16 ビット																																					
モード 5	16M バイト	A <sub>23</sub> ~ A <sub>0</sub>	8 ビット	16 ビット																																					
モード 6	64K バイト	-	-	-																																					
モード 7	1M バイト	-	-	-																																					
低消費電力状態	<ul style="list-style-type: none"> <li>・ スリープモード</li> <li>・ ソフトウェアスタンバイモード</li> <li>・ ハードウェアスタンバイモード</li> <li>・ モジュール別スタンバイ機能あり</li> <li>・ システムクロック分周比可変</li> </ul>																																								
その他	<ul style="list-style-type: none"> <li>・ クロック発振器内蔵</li> </ul>																																								
製品ラインアップ	<table border="1"> <thead> <tr> <th colspan="2">製品分類</th> <th>製品型名</th> <th>パッケージ (日立パッケージコード)</th> </tr> </thead> <tbody> <tr> <td rowspan="2">H8/3068F-ZTAT</td> <td>5V動作</td> <td>HD64F3068F HD64F3068TE</td> <td>100ピンQFP (FP-100B) 100ピンTQFP (TFP-100B)</td> </tr> <tr> <td>3V動作</td> <td>HD64F3068VF HD64F3068VTE</td> <td>100ピンQFP (FP-100B) 100ピンTQFP (TFP-100B)</td> </tr> </tbody> </table>	製品分類		製品型名	パッケージ (日立パッケージコード)	H8/3068F-ZTAT	5V動作	HD64F3068F HD64F3068TE	100ピンQFP (FP-100B) 100ピンTQFP (TFP-100B)	3V動作	HD64F3068VF HD64F3068VTE	100ピンQFP (FP-100B) 100ピンTQFP (TFP-100B)																													
製品分類		製品型名	パッケージ (日立パッケージコード)																																						
H8/3068F-ZTAT	5V動作	HD64F3068F HD64F3068TE	100ピンQFP (FP-100B) 100ピンTQFP (TFP-100B)																																						
	3V動作	HD64F3068VF HD64F3068VTE	100ピンQFP (FP-100B) 100ピンTQFP (TFP-100B)																																						

## 1.2 内部ブロック図

内部ブロック図を図 1.1 に示します。

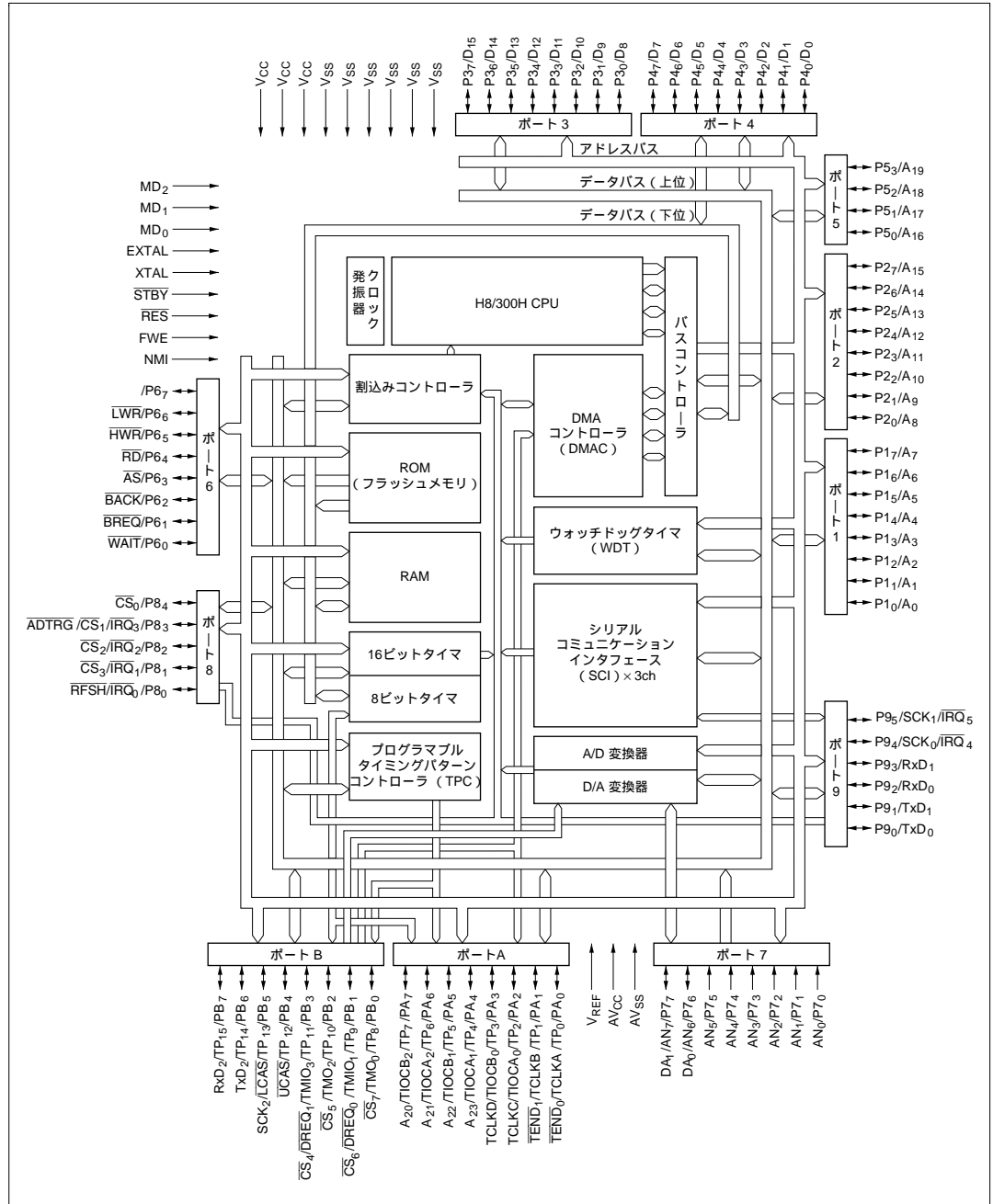


図 1.1 内部ブロック図

## 1.3 端子説明

### 1.3.1 ピン配置図

H8/3068F のピン配置図 FP-100B、TFP-100B を図 1.2 に示します。

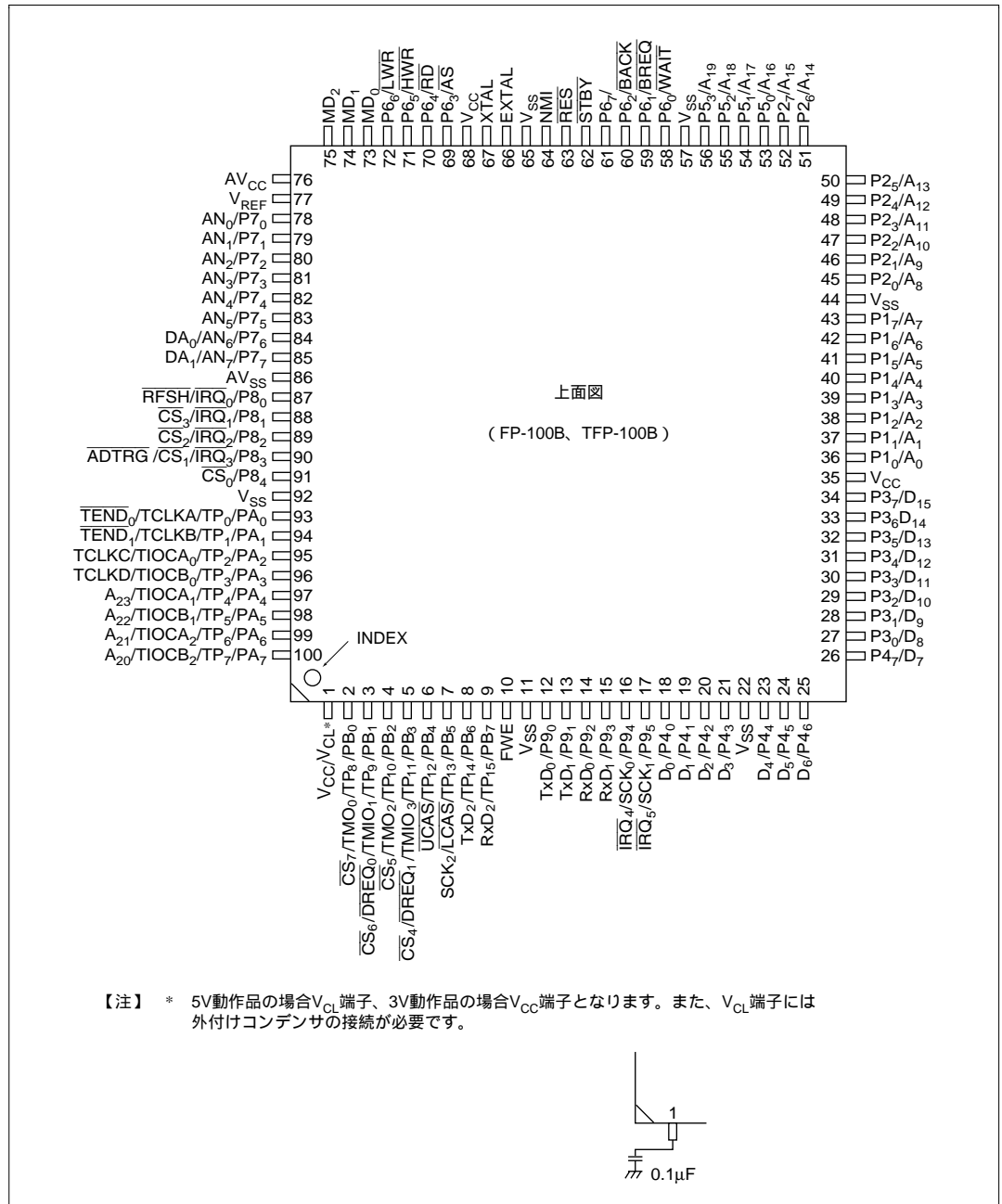
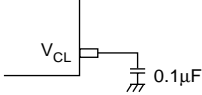


図 1.2 ピン配置図 (FP-100B、TFP-100B : 上面図)

## 1.3.2 端子機能

各端子の機能について表 1.2 に示します。

表 1.2 端子機能

分類	記号	ピン番号	入出力	名称および機能
		FP-100B TFP-100B		
電源	Vcc	1* <sup>1</sup> 、35、68	入力	<b>電源</b> 電源に接続します。 Vcc 端子は、全端子をシステムの電源に接続してください。
	Vss	11、22、44、57、 65、92	入力	<b>グラウンド</b> 電源 (0V) に接続します。 Vss 端子は、全端子をシステムの電源 (0V) に接続してください。
内部降圧端子	V <sub>CL</sub>	1* <sup>2</sup>	出力	本端子と GND (0V) との間に外付けコンデンサを接続します。 <u>Vcc と接続しないでください</u> 
クロック	XTAL	67	入力	水晶発振子を接続します。 水晶発振子を接続する場合、および外部クロック入力の場合の接続例については、「第 19 章 クロック発振器」を参照してください。
	EXTAL	66	入力	水晶発振子を接続します。また、EXTAL 端子は外部クロックを入力することもできます。 水晶発振子を接続する場合、および外部クロック入力の場合の接続例については、「第 19 章 クロック発振器」を参照してください。
		61	出力	<b>システムクロック</b> 外部デバイスにシステムクロックを供給します。

分類	記号	ピン番号	入出力	名称および機能																																				
		FP-100B TFP-100B																																						
動作モード コントロール	MD <sub>2</sub> ~ MD <sub>0</sub>	75 ~ 73	入力	<p><u>モード端子</u></p> <p>動作モードを設定します。</p> <p>MD<sub>2</sub> ~ MD<sub>0</sub> 端子と動作モードの関係は次のとおりです。これらの端子は動作中には変化させないでください。</p> <table border="1"> <thead> <tr> <th>MD<sub>2</sub></th> <th>MD<sub>1</sub></th> <th>MD<sub>0</sub></th> <th>動作モード</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>-</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>モード1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>モード2</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>モード3</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>モード4</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>モード5</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>モード6</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>モード7</td> </tr> </tbody> </table>	MD <sub>2</sub>	MD <sub>1</sub>	MD <sub>0</sub>	動作モード	0	0	0	-	0	0	1	モード1	0	1	0	モード2	0	1	1	モード3	1	0	0	モード4	1	0	1	モード5	1	1	0	モード6	1	1	1	モード7
				MD <sub>2</sub>	MD <sub>1</sub>	MD <sub>0</sub>	動作モード																																	
0	0	0	-																																					
0	0	1	モード1																																					
0	1	0	モード2																																					
0	1	1	モード3																																					
1	0	0	モード4																																					
1	0	1	モード5																																					
1	1	0	モード6																																					
1	1	1	モード7																																					
システム 制御	$\overline{\text{RES}}$	63	入力	<p><u>リセット入力</u></p> <p>この端子が Low レベルになると、リセット状態となります。</p>																																				
	FWE	10	入力	<p><u>書き込みイネーブル信号</u></p> <p>フラッシュメモリの書き込み制御信号です。</p>																																				
	$\overline{\text{STBY}}$	62	入力	<p><u>スタンバイ</u></p> <p>この端子が Low レベルになると、ハードウェアスタンバイモードに遷移します。</p>																																				
	$\overline{\text{BREQ}}$	59	入力	<p><u>バス権要求</u></p> <p>本 LSI に対し、外部バスマスタがバス権を要求します。</p>																																				
	BACK	60	出力	<p><u>バス権要求アクノリッジ</u></p> <p>バス権を外部バスマスタに解放したことを示します。</p>																																				

1. 概要

分類	記号	ピン番号	入出力	名称および機能
		FP-100B TFP-100B		
割込み	NMI	64	入力	<u>ノンマスクブル割込み</u> マスク不可能な割込みを要求します。
	$\overline{\text{IRQ}}_5 \sim \overline{\text{IRQ}}_0$	17、16、 90～87	入力	<u>割込み要求 5～0</u> マスク可能な割込みを要求します。
アドレス バス	$A_{23} \sim A_0$	100～97、 56～45、 43～36	出力	<u>アドレスバス</u> アドレスを出力します。
データバス	$D_{15} \sim D_0$	34～23、 21～18	入出力	<u>データバス</u> 双方向データバスです。
バス制御	$\overline{\text{CS}}_7 \sim \overline{\text{CS}}_0$	2～5、 88～91、	出力	<u>チップセレクト</u> エリア 7～0 の選択信号です。
	$\overline{\text{AS}}$	69	出力	<u>アドレスストrobe</u> この端子が Low レベルのとき、アドレスバス上のアドレス出力が有効であることを示します。
	$\overline{\text{RD}}$	70	出力	<u>リード</u> この端子が Low レベルのとき、外部アドレス空間のリード状態であることを示します。
	$\overline{\text{HWR}}$	71	出力	<u>ハイライト</u> この端子が Low レベルのとき、外部アドレス空間のライト状態であり、データバスの上位側 ( $D_{15} \sim D_8$ ) が有効であることを示します。
	$\overline{\text{LWR}}$	72	出力	<u>ロウライト</u> この端子が Low レベルのとき、外部アドレス空間のライト状態であり、データバスの下位側 ( $D_7 \sim D_0$ ) が有効であることを示します。
	$\overline{\text{WAIT}}$	58	入力	<u>ウェイト</u> 外部アドレス空間をアクセスするときに、バスサイクルにウェイトステートの挿入を要求します。

分類	記号	ピン番号	入出力	名称および機能
		FP-100B TFP-100B		
DRAM イン タフェース	$\overline{\text{RFSH}}$	87	出力	<u>リフレッシュ</u> リフレッシュサイクルを示します。
	$\overline{\text{CS}}_2 \sim \overline{\text{CS}}_5$	89、88、5、 4	出力	<u>ロウアドレスストロープ (<math>\overline{\text{RAS}}</math>)</u> DRAMのロウアドレスストロープ信号です。
	$\overline{\text{RD}}$	70	出力	<u>ライトイネーブル (<math>\overline{\text{WE}}</math>)</u> DRAMのライトイネーブル信号です。
	$\overline{\text{HWR}}$ $\overline{\text{UCAS}}$	71 6	出力	<u>アッパーカラムアドレスストロープ (<math>\overline{\text{UCAS}}</math>)</u> DRAMのカラムアドレスストロープ信号です。
	$\overline{\text{LWR}}$ $\overline{\text{LCAS}}$	72 7	出力	<u>ロウアーカラムアドレスストロープ (<math>\overline{\text{LCAS}}</math>)</u> DRAMのカラムアドレスストロープ信号です。
DMA コント ローラ (DMAC)	$\overline{\text{DREQ}}_1$ 、 $\overline{\text{DREQ}}_0$	5、3	入力	<u>DMA 要求 1、0</u> DMAC の起動を要求します。
	$\overline{\text{TEND}}_1$ 、 $\overline{\text{TEND}}_0$	94、93	出力	<u>DMA 終了 1、0</u> DMAC のデータ転送終了を示します。
16 ビット タイマ	TCLKD ~ TCLKA	96 ~ 93	入力	<u>クロック入力 D~A</u> 外部クロックを入力します。
	TIOCA <sub>2</sub> ~ TIOCA <sub>0</sub>	99、97、95	入出力	<u>インプットキャプチャ /</u> <u>アウトプットコンペア A2 ~ A0</u> GRA2 ~ A0 のアウトプットコンペア出力 / イン プットキャプチャ入力 / PWM 出力端子です。
	TIOCB <sub>2</sub> ~ TIOCB <sub>0</sub>	100、98、 96	入出力	<u>インプットキャプチャ /</u> <u>アウトプットコンペア B2 ~ B0</u> GRB2 ~ B0 のアウトプットコンペア出力 / イン プットキャプチャ入力 / PWM 出力端子です。
8 ビット タイマ	TMO <sub>0</sub> 、TMO <sub>2</sub>	2、4	出力	<u>コンペアマッチ出力</u> コンペアマッチ出力端子です。
	TMIO <sub>1</sub> 、 TMIO <sub>3</sub>	3、5	入出力	<u>インプットキャプチャ入力 / コンペアマッチ出力</u> <u>インプットキャプチャ入力 / コンペアマッチ出力</u> 端子です。
	TCLKD ~ TCLKA	96 ~ 93	入力	<u>カウンタ外部クロック入力</u> カウンタに入力する外部クロックの入力端子で す。



1. 概要

分類	記号	ピン番号	入出力	名称および機能
		FP-100B TFP-100B		
プログラマブルタイミングパターンコントローラ (TPC)	TP <sub>15</sub> ~ TP <sub>0</sub>	9 ~ 2、 100 ~ 93	出力	TPC 出力 15 ~ 0 パルス出力端子です。
シリアルコミュニケーションインタフェース (SCI)	TxD <sub>2</sub> ~ TxD <sub>0</sub>	8、13、12	出力	トランスミットデータ (チャンネル 0、1、2) SCI のデータ出力端子です。
	RxD <sub>2</sub> ~ RxD <sub>0</sub>	9、15、14	入力	レシーブデータ (チャンネル 0、1、2) SCI のデータ入力端子です。
	SCK <sub>2</sub> ~ SCK <sub>0</sub>	7、17、16	入出力	シリアルクロック (チャンネル 0、1、2) SCI のクロック入出力端子です。
A/D 変換器	AN <sub>7</sub> ~ AN <sub>0</sub>	85 ~ 78	入力	アナログ 7 ~ 0 アナログ入力端子です。
	ADTRG	90	入力	A/D 変換外部トリガ入力 A/D 変換開始のための外部トリガ入力端子です。
D/A 変換器	DA <sub>1</sub> 、DA <sub>0</sub>	85、84	出力	アナログ出力 D/A 変換器のアナログ出力端子です。
A/D 変換器、 D/A 変換器	AVcc	76	入力	A/D 変換器および D/A 変換器の電源端子です。 A/D 変換器および D/A 変換器を使用しない場合はシステム電源に接続してください。
	AVss	86	入力	A/D 変換器および D/A 変換器のグランド端子です。 システムの電源 (0V) に接続してください。
	V <sub>REF</sub>	77	入力	A/D 変換器および D/A 変換器の基準電圧入力端子です。 A/D 変換器および D/A 変換器を使用しない場合はシステムの電源に接続してください。

分類	記号	ピン番号	入出力	名称および機能
		FP-100B TFP-100B		
I/O ポート	P1 <sub>7</sub> ~ P1 <sub>0</sub>	43 ~ 36	入出力	<u>ポート 1</u> 8 ビットの入出力端子です。 ポート 1 データディレクションレジスタ (P1DDR) によって、1 ビットごとに入出力を指定できます。
	P2 <sub>7</sub> ~ P2 <sub>0</sub>	52 ~ 45	入出力	<u>ポート 2</u> 8 ビットの入出力端子です。 ポート 2 データディレクションレジスタ (P2DDR) によって、1 ビットごとに入出力を指定できます。
	P3 <sub>7</sub> ~ P3 <sub>0</sub>	34 ~ 27	入出力	<u>ポート 3</u> 8 ビットの入出力端子です。 ポート 3 データディレクションレジスタ (P3DDR) によって、1 ビットごとに入出力を指定できます。
	P4 <sub>7</sub> ~ P4 <sub>0</sub>	26 ~ 23、 21 ~ 18	入出力	<u>ポート 4</u> 8 ビットの入出力端子です。 ポート 4 データディレクションレジスタ (P4DDR) によって、1 ビットごとに入出力を指定できます。
	P5 <sub>3</sub> ~ P5 <sub>0</sub>	56 ~ 53	入出力	<u>ポート 5</u> 4 ビットの入出力端子です。 ポート 5 データディレクションレジスタ (P5DDR) によって、1 ビットごとに入出力を指定できます。
	P6 <sub>7</sub> ~ P6 <sub>0</sub>	61、 72 ~ 69、 60 ~ 58	入出力	<u>ポート 6</u> 8 ビットの入出力端子です。 ポート 6 データディレクションレジスタ (P6DDR) によって、1 ビットごとに入出力を指定できます。
	P7 <sub>7</sub> ~ P7 <sub>0</sub>	85 ~ 78	入力	<u>ポート 7</u> 8 ビットの入力端子です。

## 1. 概要

分類	記号	ピン番号	入出力	名称および機能
		FP-100B TFP-100B		
I/O ポート	P8 <sub>4</sub> ~ P8 <sub>0</sub>	91 ~ 87	入出力	ポート 8 5 ビットの入出力端子です。 ポート 8 データディレクションレジスタ (P8DDR) によって、1 ビットごとに入出力を指定できます。
	P9 <sub>5</sub> ~ P9 <sub>0</sub>	17 ~ 12	入出力	ポート 9 6 ビットの入出力端子です。 ポート 9 データディレクションレジスタ (P9DDR) によって、1 ビットごとに入出力を指定できます。
	PA <sub>7</sub> ~ PA <sub>0</sub>	100 ~ 93	入出力	ポート A 8 ビットの入出力端子です。 ポート A データディレクションレジスタ (PADDDR) によって、1 ビットごとに入出力を指定できます。
	PB <sub>7</sub> ~ PB <sub>0</sub>	9 ~ 2	入出力	ポート B 8 ビットの入出力端子です。 ポート B データディレクションレジスタ (PBDDR) によって、1 ビットごとに入出力を指定できます。

【注】 \*1 3V 動作品の場合

\*2 5V 動作品の場合

## 1.3.3 モード別ピン配置一覧

モード別ピン配置一覧を表 1.3 に示します。

表 1.3 モード別ピン配置一覧 (FP-100B、TFP-100B)

ピン番号	端子名						
	モード 1	モード 2	モード 3	モード 4	モード 5	モード 6	モード 7
FP-100B TFP-100B							
1	Vcc/V <sub>CL</sub> *1	Vcc/V <sub>CL</sub> *1	Vcc/V <sub>CL</sub> *1	Vcc/V <sub>CL</sub> *1	Vcc/V <sub>CL</sub> *1	Vcc/V <sub>CL</sub> *1	Vcc/V <sub>CL</sub> *1
2	PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub> /CS <sub>7</sub>	PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub> /CS <sub>7</sub>	PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub> /CS <sub>7</sub>	PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub> /CS <sub>7</sub>	PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub> /CS <sub>7</sub>	PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub>	PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub>
3	PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub> /CS <sub>6</sub>	PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub> /CS <sub>6</sub>	PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub> /CS <sub>6</sub>	PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub> /CS <sub>6</sub>	PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub> /CS <sub>6</sub>	PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub>	PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub>
4	PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub> /CS <sub>5</sub>	PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub> /CS <sub>5</sub>	PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub> /CS <sub>5</sub>	PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub> /CS <sub>5</sub>	PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub> /CS <sub>5</sub>	PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub>	PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub>
5	PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub> /CS <sub>4</sub>	PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub> /CS <sub>4</sub>	PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub> /CS <sub>4</sub>	PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub> /CS <sub>4</sub>	PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub> /CS <sub>4</sub>	PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub>	PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub>
6	PB <sub>4</sub> /TP <sub>12</sub> / UCAS	PB <sub>4</sub> /TP <sub>12</sub> / UCAS	PB <sub>4</sub> /TP <sub>12</sub> / UCAS	PB <sub>4</sub> /TP <sub>12</sub> / UCAS	PB <sub>4</sub> /TP <sub>12</sub> / UCAS	PB <sub>4</sub> /TP <sub>12</sub>	PB <sub>4</sub> /TP <sub>12</sub>
7	PB <sub>5</sub> /TP <sub>13</sub> / LCAS/ SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub> / LCAS/ SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub> / LCAS/ SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub> / LCAS/ SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub> / LCAS/ SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub> / SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub> / SCK <sub>2</sub>
8	PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>
9	PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>
10	FWE	FWE	FWE	FWE	FWE	FWE	FWE
11	Vss	Vss	Vss	Vss	Vss	Vss	Vss
12	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>
13	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>

1. 概要

ピン番号	端子名						
	モード 1	モード 2	モード 3	モード 4	モード 5	モード 6	モード 7
FP-100B TFP-100B							
14	P9 <sub>2</sub> /RXD <sub>0</sub>	P9 <sub>2</sub> /RXD <sub>0</sub>	P9 <sub>2</sub> /RXD <sub>0</sub>	P9 <sub>2</sub> /RXD <sub>0</sub>	P9 <sub>2</sub> /RXD <sub>0</sub>	P9 <sub>2</sub> /RXD <sub>0</sub>	P9 <sub>2</sub> /RXD <sub>0</sub>
15	P9 <sub>3</sub> /RXD <sub>1</sub>	P9 <sub>3</sub> /RXD <sub>1</sub>	P9 <sub>3</sub> /RXD <sub>1</sub>	P9 <sub>3</sub> /RXD <sub>1</sub>	P9 <sub>3</sub> /RXD <sub>1</sub>	P9 <sub>3</sub> /RXD <sub>1</sub>	P9 <sub>3</sub> /RXD <sub>1</sub>
16	P9 <sub>4</sub> /IRQ <sub>4</sub> / SCK <sub>0</sub>	P9 <sub>4</sub> /IRQ <sub>4</sub> / SCK <sub>0</sub>	P9 <sub>4</sub> /IRQ <sub>4</sub> / SCK <sub>0</sub>	P9 <sub>4</sub> /IRQ <sub>4</sub> / SCK <sub>0</sub>	P9 <sub>4</sub> /IRQ <sub>4</sub> / SCK <sub>0</sub>	P9 <sub>4</sub> /IRQ <sub>4</sub> / SCK <sub>0</sub>	P9 <sub>4</sub> /IRQ <sub>4</sub> / SCK <sub>0</sub>
17	P9 <sub>5</sub> /IRQ <sub>5</sub> / SCK <sub>1</sub>	P9 <sub>5</sub> /IRQ <sub>5</sub> / SCK <sub>1</sub>	P9 <sub>5</sub> /IRQ <sub>5</sub> / SCK <sub>1</sub>	P9 <sub>5</sub> /IRQ <sub>5</sub> / SCK <sub>1</sub>	P9 <sub>5</sub> /IRQ <sub>5</sub> / SCK <sub>1</sub>	P9 <sub>5</sub> /IRQ <sub>5</sub> / SCK <sub>1</sub>	P9 <sub>5</sub> /IRQ <sub>5</sub> / SCK <sub>1</sub>
18	P4 <sub>0</sub> /D <sub>0</sub> * <sup>2</sup>	P4 <sub>0</sub> /D <sub>0</sub> * <sup>3</sup>	P4 <sub>0</sub> /D <sub>0</sub> * <sup>2</sup>	P4 <sub>0</sub> /D <sub>0</sub> * <sup>3</sup>	P4 <sub>0</sub> /D <sub>0</sub> * <sup>2</sup>	P4 <sub>0</sub>	P4 <sub>0</sub>
19	P4 <sub>1</sub> /D <sub>1</sub> * <sup>2</sup>	P4 <sub>1</sub> /D <sub>1</sub> * <sup>3</sup>	P4 <sub>1</sub> /D <sub>1</sub> * <sup>2</sup>	P4 <sub>1</sub> /D <sub>1</sub> * <sup>3</sup>	P4 <sub>1</sub> /D <sub>1</sub> * <sup>2</sup>	P4 <sub>1</sub>	P4 <sub>1</sub>
20	P4 <sub>2</sub> /D <sub>2</sub> * <sup>2</sup>	P4 <sub>2</sub> /D <sub>2</sub> * <sup>3</sup>	P4 <sub>2</sub> /D <sub>2</sub> * <sup>2</sup>	P4 <sub>2</sub> /D <sub>2</sub> * <sup>3</sup>	P4 <sub>2</sub> /D <sub>2</sub> * <sup>2</sup>	P4 <sub>2</sub>	P4 <sub>2</sub>
21	P4 <sub>3</sub> /D <sub>3</sub> * <sup>2</sup>	P4 <sub>3</sub> /D <sub>3</sub> * <sup>3</sup>	P4 <sub>3</sub> /D <sub>3</sub> * <sup>2</sup>	P4 <sub>3</sub> /D <sub>3</sub> * <sup>3</sup>	P4 <sub>3</sub> /D <sub>3</sub> * <sup>2</sup>	P4 <sub>3</sub>	P4 <sub>3</sub>
22	Vss	Vss	Vss	Vss	Vss	Vss	Vss
23	P4 <sub>4</sub> /D <sub>4</sub> * <sup>2</sup>	P4 <sub>4</sub> /D <sub>4</sub> * <sup>3</sup>	P4 <sub>4</sub> /D <sub>4</sub> * <sup>2</sup>	P4 <sub>4</sub> /D <sub>4</sub> * <sup>3</sup>	P4 <sub>4</sub> /D <sub>4</sub> * <sup>2</sup>	P4 <sub>4</sub>	P4 <sub>4</sub>
24	P4 <sub>5</sub> /D <sub>5</sub> * <sup>2</sup>	P4 <sub>5</sub> /D <sub>5</sub> * <sup>3</sup>	P4 <sub>5</sub> /D <sub>5</sub> * <sup>2</sup>	P4 <sub>5</sub> /D <sub>5</sub> * <sup>3</sup>	P4 <sub>5</sub> /D <sub>5</sub> * <sup>2</sup>	P4 <sub>5</sub>	P4 <sub>5</sub>
25	P4 <sub>6</sub> /D <sub>6</sub> * <sup>2</sup>	P4 <sub>6</sub> /D <sub>6</sub> * <sup>3</sup>	P4 <sub>6</sub> /D <sub>6</sub> * <sup>2</sup>	P4 <sub>6</sub> /D <sub>6</sub> * <sup>3</sup>	P4 <sub>6</sub> /D <sub>6</sub> * <sup>2</sup>	P4 <sub>6</sub>	P4 <sub>6</sub>
26	P4 <sub>7</sub> /D <sub>7</sub> * <sup>2</sup>	P4 <sub>7</sub> /D <sub>7</sub> * <sup>3</sup>	P4 <sub>7</sub> /D <sub>7</sub> * <sup>2</sup>	P4 <sub>7</sub> /D <sub>7</sub> * <sup>3</sup>	P4 <sub>7</sub> /D <sub>7</sub> * <sup>2</sup>	P4 <sub>7</sub>	P4 <sub>7</sub>
27	D <sub>8</sub>	D <sub>8</sub>	D <sub>8</sub>	D <sub>8</sub>	D <sub>8</sub>	P3 <sub>0</sub>	P3 <sub>0</sub>
28	D <sub>9</sub>	D <sub>9</sub>	D <sub>9</sub>	D <sub>9</sub>	D <sub>9</sub>	P3 <sub>1</sub>	P3 <sub>1</sub>
29	D <sub>10</sub>	D <sub>10</sub>	D <sub>10</sub>	D <sub>10</sub>	D <sub>10</sub>	P3 <sub>2</sub>	P3 <sub>2</sub>
30	D <sub>11</sub>	D <sub>11</sub>	D <sub>11</sub>	D <sub>11</sub>	D <sub>11</sub>	P3 <sub>3</sub>	P3 <sub>3</sub>
31	D <sub>12</sub>	D <sub>12</sub>	D <sub>12</sub>	D <sub>12</sub>	D <sub>12</sub>	P3 <sub>4</sub>	P3 <sub>4</sub>
32	D <sub>13</sub>	D <sub>13</sub>	D <sub>13</sub>	D <sub>13</sub>	D <sub>13</sub>	P3 <sub>5</sub>	P3 <sub>5</sub>
33	D <sub>14</sub>	D <sub>14</sub>	D <sub>14</sub>	D <sub>14</sub>	D <sub>14</sub>	P3 <sub>6</sub>	P3 <sub>6</sub>
34	D <sub>15</sub>	D <sub>15</sub>	D <sub>15</sub>	D <sub>15</sub>	D <sub>15</sub>	P3 <sub>7</sub>	P3 <sub>7</sub>
35	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc
36	A <sub>0</sub>	A <sub>0</sub>	A <sub>0</sub>	A <sub>0</sub>	P1 <sub>0</sub> /A <sub>0</sub>	P1 <sub>0</sub>	P1 <sub>0</sub>
37	A <sub>1</sub>	A <sub>1</sub>	A <sub>1</sub>	A <sub>1</sub>	P1 <sub>1</sub> /A <sub>1</sub>	P1 <sub>1</sub>	P1 <sub>1</sub>
38	A <sub>2</sub>	A <sub>2</sub>	A <sub>2</sub>	A <sub>2</sub>	P1 <sub>2</sub> /A <sub>2</sub>	P1 <sub>2</sub>	P1 <sub>2</sub>
39	A <sub>3</sub>	A <sub>3</sub>	A <sub>3</sub>	A <sub>3</sub>	P1 <sub>3</sub> /A <sub>3</sub>	P1 <sub>3</sub>	P1 <sub>3</sub>
40	A <sub>4</sub>	A <sub>4</sub>	A <sub>4</sub>	A <sub>4</sub>	P1 <sub>4</sub> /A <sub>4</sub>	P1 <sub>4</sub>	P1 <sub>4</sub>

ピン番号	端子名						
	モード1	モード2	モード3	モード4	モード5	モード6	モード7
FP-100B TFP-100B							
41	A <sub>5</sub>	A <sub>5</sub>	A <sub>5</sub>	A <sub>5</sub>	P1 <sub>5</sub> /A <sub>5</sub>	P1 <sub>5</sub>	P1 <sub>5</sub>
42	A <sub>6</sub>	A <sub>6</sub>	A <sub>6</sub>	A <sub>6</sub>	P1 <sub>6</sub> /A <sub>6</sub>	P1 <sub>6</sub>	P1 <sub>6</sub>
43	A <sub>7</sub>	A <sub>7</sub>	A <sub>7</sub>	A <sub>7</sub>	P1 <sub>7</sub> /A <sub>7</sub>	P1 <sub>7</sub>	P1 <sub>7</sub>
44	Vss	Vss	Vss	Vss	Vss	Vss	Vss
45	A <sub>8</sub>	A <sub>8</sub>	A <sub>8</sub>	A <sub>8</sub>	P2 <sub>0</sub> /A <sub>8</sub>	P2 <sub>0</sub>	P2 <sub>0</sub>
46	A <sub>9</sub>	A <sub>9</sub>	A <sub>9</sub>	A <sub>9</sub>	P2 <sub>1</sub> /A <sub>9</sub>	P2 <sub>1</sub>	P2 <sub>1</sub>
47	A <sub>10</sub>	A <sub>10</sub>	A <sub>10</sub>	A <sub>10</sub>	P2 <sub>2</sub> /A <sub>10</sub>	P2 <sub>2</sub>	P2 <sub>2</sub>
48	A <sub>11</sub>	A <sub>11</sub>	A <sub>11</sub>	A <sub>11</sub>	P2 <sub>3</sub> /A <sub>11</sub>	P2 <sub>3</sub>	P2 <sub>3</sub>
49	A <sub>12</sub>	A <sub>12</sub>	A <sub>12</sub>	A <sub>12</sub>	P2 <sub>4</sub> /A <sub>12</sub>	P2 <sub>4</sub>	P2 <sub>4</sub>
50	A <sub>13</sub>	A <sub>13</sub>	A <sub>13</sub>	A <sub>13</sub>	P2 <sub>5</sub> /A <sub>13</sub>	P2 <sub>5</sub>	P2 <sub>5</sub>
51	A <sub>14</sub>	A <sub>14</sub>	A <sub>14</sub>	A <sub>14</sub>	P2 <sub>6</sub> /A <sub>14</sub>	P2 <sub>6</sub>	P2 <sub>6</sub>
52	A <sub>15</sub>	A <sub>15</sub>	A <sub>15</sub>	A <sub>15</sub>	P2 <sub>7</sub> /A <sub>15</sub>	P2 <sub>7</sub>	P2 <sub>7</sub>
53	A <sub>16</sub>	A <sub>16</sub>	A <sub>16</sub>	A <sub>16</sub>	P5 <sub>0</sub> /A <sub>16</sub>	P5 <sub>0</sub>	P5 <sub>0</sub>
54	A <sub>17</sub>	A <sub>17</sub>	A <sub>17</sub>	A <sub>17</sub>	P5 <sub>1</sub> /A <sub>17</sub>	P5 <sub>1</sub>	P5 <sub>1</sub>
55	A <sub>18</sub>	A <sub>18</sub>	A <sub>18</sub>	A <sub>18</sub>	P5 <sub>2</sub> /A <sub>18</sub>	P5 <sub>2</sub>	P5 <sub>2</sub>
56	A <sub>19</sub>	A <sub>19</sub>	A <sub>19</sub>	A <sub>19</sub>	P5 <sub>3</sub> /A <sub>19</sub>	P5 <sub>3</sub>	P5 <sub>3</sub>
57	Vss	Vss	Vss	Vss	Vss	Vss	Vss
58	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub>	P6 <sub>0</sub>
59	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub>	P6 <sub>1</sub>
60	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK	P6 <sub>2</sub>	P6 <sub>2</sub>
61					P6 <sub>7</sub> /	P6 <sub>7</sub> /	P6 <sub>7</sub> /
62	STBY	STBY	STBY	STBY	STBY	STBY	STBY
63	RES	RES	RES	RES	RES	RES	RES
64	NMI	NMI	NMI	NMI	NMI	NMI	NMI
65	Vss	Vss	Vss	Vss	Vss	Vss	Vss
66	EXTAL	EXTAL	EXTAL	EXTAL	EXTAL	EXTAL	EXTAL
67	XTAL	XTAL	XTAL	XTAL	XTAL	XTAL	XTAL
68	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc
69	AS	AS	AS	AS	AS	P6 <sub>3</sub>	P6 <sub>3</sub>
70	RD	RD	RD	RD	RD	P6 <sub>4</sub>	P6 <sub>4</sub>
71	HWR	HWR	HWR	HWR	HWR	P6 <sub>5</sub>	P6 <sub>5</sub>

1. 概要

ピン番号	端子名						
	モード 1	モード 2	モード 3	モード 4	モード 5	モード 6	モード 7
FP-100B TFP-100B							
72	$\overline{LWR}$	$\overline{LWR}$	$\overline{LWR}$	$\overline{LWR}$	$\overline{LWR}$	P6 <sub>6</sub>	P6 <sub>6</sub>
73	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>
74	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>
75	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>
76	AVcc	AVcc	AVcc	AVcc	AVcc	AVcc	AVcc
77	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>
78	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>
79	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>
80	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>
81	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>
82	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>
83	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>
84	P7 <sub>6</sub> /AN <sub>6</sub> / DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> / DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> / DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> / DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> / DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> / DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> / DA <sub>0</sub>
85	P7 <sub>7</sub> /AN <sub>7</sub> / DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> / DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> / DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> / DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> / DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> / DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> / DA <sub>1</sub>
86	AVss	AVss	AVss	AVss	AVss	AVss	AVss
87	P8 <sub>0</sub> /IRQ <sub>0</sub> / RFSH	P8 <sub>0</sub> /IRQ <sub>0</sub> / RFSH	P8 <sub>0</sub> /IRQ <sub>0</sub> / RFSH	P8 <sub>0</sub> /IRQ <sub>0</sub> / RFSH	P8 <sub>0</sub> /IRQ <sub>0</sub> / RFSH	P8 <sub>0</sub> /IRQ <sub>0</sub>	P8 <sub>0</sub> /IRQ <sub>0</sub>
88	P8 <sub>1</sub> /IRQ <sub>1</sub> / CS <sub>3</sub>	P8 <sub>1</sub> /IRQ <sub>1</sub> / CS <sub>3</sub>	P8 <sub>1</sub> /IRQ <sub>1</sub> / CS <sub>3</sub>	P8 <sub>1</sub> /IRQ <sub>1</sub> / CS <sub>3</sub>	P8 <sub>1</sub> /IRQ <sub>1</sub> / CS <sub>3</sub>	P8 <sub>1</sub> /IRQ <sub>1</sub>	P8 <sub>1</sub> /IRQ <sub>1</sub>
89	P8 <sub>2</sub> /IRQ <sub>2</sub> / CS <sub>2</sub>	P8 <sub>2</sub> /IRQ <sub>2</sub> / CS <sub>2</sub>	P8 <sub>2</sub> /IRQ <sub>2</sub> / CS <sub>2</sub>	P8 <sub>2</sub> /IRQ <sub>2</sub> / CS <sub>2</sub>	P8 <sub>2</sub> /IRQ <sub>2</sub> / CS <sub>2</sub>	P8 <sub>2</sub> /IRQ <sub>2</sub>	P8 <sub>2</sub> /IRQ <sub>2</sub>
90	P8 <sub>3</sub> /IRQ <sub>3</sub> / CS <sub>1</sub> / ADTRG	P8 <sub>3</sub> /IRQ <sub>3</sub> / CS <sub>1</sub> / ADTRG	P8 <sub>3</sub> /IRQ <sub>3</sub> / CS <sub>1</sub> / ADTRG	P8 <sub>3</sub> /IRQ <sub>3</sub> / CS <sub>1</sub> / ADTRG	P8 <sub>3</sub> /IRQ <sub>3</sub> / CS <sub>1</sub> / ADTRG	P8 <sub>3</sub> /IRQ <sub>3</sub> / ADTRG	P8 <sub>3</sub> /IRQ <sub>3</sub> / ADTRG
91	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub>	P8 <sub>4</sub>
92	Vss	Vss	Vss	Vss	Vss	Vss	Vss
93	PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ TEND <sub>0</sub>	PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ TEND <sub>0</sub>	PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ TEND <sub>0</sub>	PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ TEND <sub>0</sub>	PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ TEND <sub>0</sub>	PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ TEND <sub>0</sub>	PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ TEND <sub>0</sub>

ピン番号	端子名						
	モード1	モード2	モード3	モード4	モード5	モード6	モード7
FP-100B TFP-100B							
94	PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{\text{TEND}}_1$	PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{\text{TEND}}_1$	PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{\text{TEND}}_1$	PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{\text{TEND}}_1$	PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{\text{TEND}}_1$	PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{\text{TEND}}_1$	PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{\text{TEND}}_1$
95	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC
96	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD
97	PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub>	PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub>	PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub> /A <sub>23</sub>	PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub> /A <sub>23</sub>	PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub> /A <sub>23</sub>	PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub>	PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub>
98	PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub>	PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub>	PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub> /A <sub>22</sub>	PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub> /A <sub>22</sub>	PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub> /A <sub>22</sub>	PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub>	PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub>
99	PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub>	PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub>	PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub> /A <sub>21</sub>	PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub> /A <sub>21</sub>	PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub> /A <sub>21</sub>	PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub>	PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub>
100	PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub>	PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub>	A <sub>20</sub>	A <sub>20</sub>	PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub> /A <sub>20</sub>	PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub>	PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub>

【注】 \*1 5V 動作品は V<sub>CL</sub> 端子、3V 動作品は V<sub>CC</sub> 端子となります。

\*2 モード 1、3、5 では、リセット直後、P4<sub>0</sub>/D<sub>0</sub> ~ P4<sub>7</sub>/D<sub>7</sub> 端子は P4<sub>0</sub> ~ P4<sub>7</sub> 端子となります（プログラムで変更できます）。

\*3 モード 2、4 では、リセット直後、P4<sub>0</sub>/D<sub>0</sub> ~ P4<sub>7</sub>/D<sub>7</sub> 端子は D<sub>0</sub> ~ D<sub>7</sub> 端子となっています（プログラムで変更できます）。





---

# 2. CPU

---

## 第2章 目次

2.1	概要	25
2.1.1	特長	25
2.1.2	H8/300CPU との相違点	26
2.2	CPU 動作モード	27
2.3	アドレス空間	28
2.4	レジスタ構成	29
2.4.1	概要	29
2.4.2	汎用レジスタ	30
2.4.3	コントロールレジスタ	31
2.4.4	CPU 内部レジスタの初期値	33
2.5	データ構成	34
2.5.1	汎用レジスタのデータ構成	34
2.5.2	メモリ上でのデータ構成	35
2.6	命令セット	37
2.6.1	命令セットの概要	37
2.6.2	命令とアドレッシングモードの組み合わせ	37
2.6.3	命令の機能別一覧	39
2.6.4	命令の基本フォーマット	50
2.6.5	ビット操作命令使用上の注意	51
2.7	アドレッシングモードと実効アドレスの計算方法	53
2.7.1	アドレッシングモード	53
2.7.2	実効アドレスの計算方法	56
2.8	処理状態	60
2.8.1	概要	60
2.8.2	プログラム実行状態	60
2.8.3	例外処理状態	60
2.8.4	例外処理の動作	63

2.8.5	バス権解放状態 .....	64
2.8.6	リセット状態.....	64
2.8.7	低消費電力状態 .....	64
2.9	基本動作タイミング.....	66
2.9.1	概要.....	66
2.9.2	内蔵メモリアクセスタイミング.....	66
2.9.3	内蔵周辺モジュールアクセスタイミング.....	67
2.9.4	外部アドレス空間アクセスタイミング.....	68

## 2.1 概要

H8/300H CPU は、H8/300CPU の上位互換のアーキテクチャを持つ内部 32 ビット構成の高速 CPU です。H8/300H CPU は、16 ビット×16 本の汎用レジスタを持ち、16M バイトのリニアなアドレス空間を取り扱うことができ、リアルタイム制御に最適です。

### 2.1.1 特長

H8/300H CPU には、次の特長があります。

#### H8/300CPU 上位互換

H8/300 シリーズのオブジェクトプログラムを実行可能

#### 汎用レジスタ方式

16 ビット×16 本 (8 ビット×16 本、32 ビット×8 本としても使用可能)

#### 62 種類の基本命令

- ・ 8/16/32 ビット転送、演算命令
- ・ 乗除算命令
- ・ 強力なビット操作命令

#### 8 種類のアドレッシングモード

- ・ レジスタ直接 (Rn)
- ・ レジスタ間接 (@ERn)
- ・ ディスプレースメント付レジスタ間接 (@ (d:16,ERn) , @ (d:24, ERn) )
- ・ ポストインクリメント / プリデクリメントレジスタ間接 (@ERn + /@ - ERn)
- ・ 絶対アドレス (@aa:8, @aa:16, @aa:24)
- ・ イミディエイト (#xx:8, #xx:16, #xx:32)
- ・ プログラムカウンタ相対 (@ (d:8,PC) , @ (d:16 PC) )
- ・ メモリ間接 (@@aa:8)

#### 16M バイトのリニアアドレス空間

#### 高速動作

- ・ 頻出命令をすべて 2~4 ステートで実行
- ・ 最高動作周波数 : 25MHz
- ・ 8/16/32 ビットレジスタ間加減算      80ns
- ・ 8×8 ビットレジスタ間乗算            560ns
- ・ 16÷8 ビットレジスタ間除算          560ns
- ・ 16×16 ビットレジスタ間乗算        880ns
- ・ 32÷16 ビットレジスタ間除算        880ns

#### 2 種類の CPU 動作モード

- ・ ノーマルモード
- ・ アドバンスモード

低消費電力動作

SLEEP 命令により低消費電力状態に遷移

## 2.1.2 H8/300CPU との相違点

H8/300H CPU は、H8/300CPU に対して、次の点が強化、拡張されています。

汎用レジスタを拡張

16 ビット×8 本の拡張レジスタを追加

アドレス空間を拡張

- ・アドバンスモードのとき、最大 16M バイトのアドレス空間を使用可能
- ・ノーマルモードのとき、H8/300CPU と同一の 64k バイトのアドレス空間を使用可能

アドレッシングモードを強化

16M バイトのアドレス空間を有効に使用可能

命令強化

- ・32 ビット転送、演算命令を追加
- ・符号付き乗除算命令などを追加

## 2.2 CPU 動作モード

H8/300H CPU は、ノーマルモードおよびアドバンストモードの2つのCPU動作モードをもっています。サポートするアドレス空間は、ノーマルモードの場合最大64kバイト、アドバンストモードの場合最大16Mバイトとなります。

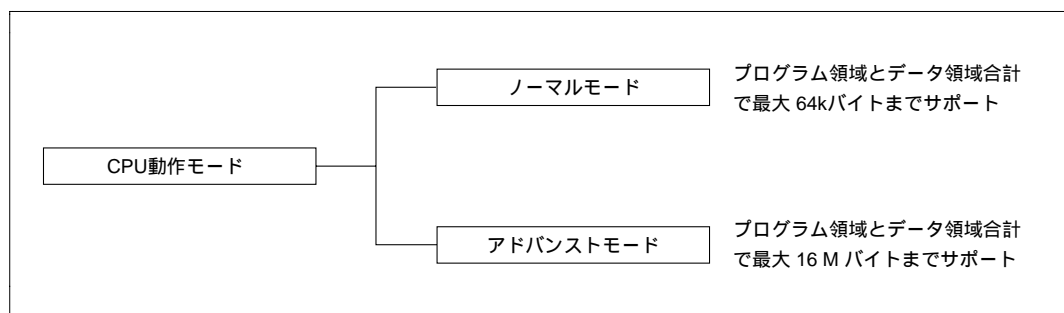


図 2.1 CPU 動作モード

## 2.3 アドレス空間

本LSIでのメモリマップの概要を図2.2に示します。H8/300H CPUはノーマルモードのとき最大64Kバイト、またアドバンスモードのとき最大16Mバイトのアドレス空間をリニアに使用することができます。詳細は「3.6 各動作モードのメモリマップ」を参照してください。

アドレス空間が1Mバイトモードの場合、実効アドレスの上位4ビットは無視され、20ビットのアドレスとなります。

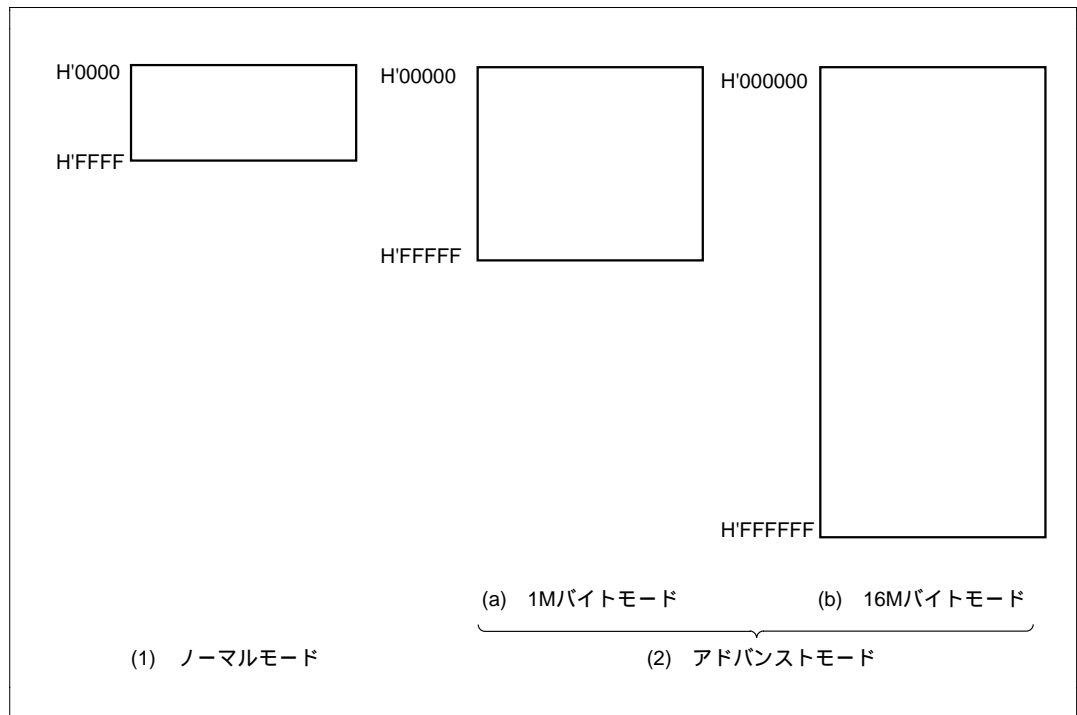


図2.2 メモリマップ

## 2.4 レジスタ構成

### 2.4.1 概要

H8/300H CPU の内部レジスタ構成を図 2.3 に示します。これらのレジスタは、汎用レジスタとコントロールレジスタの 2 つに分類することができます。

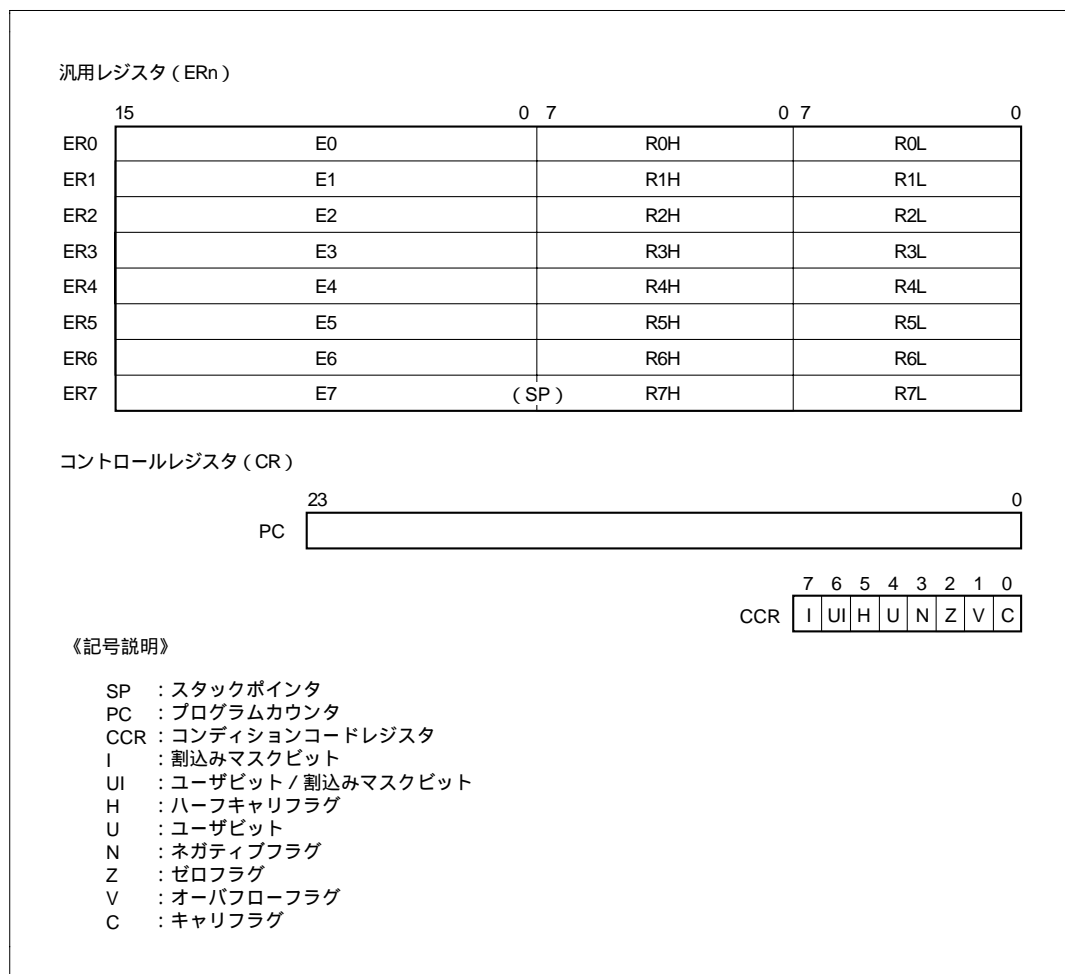


図 2.3 CPU 内部レジスタ構成



## 2.4.2 汎用レジスタ

H8/300H CPUは32ビット長の汎用レジスタ8本を持っています。汎用レジスタは、すべて同じ機能を持っており、アドレスレジスタとしてもデータレジスタとしても使用することができます。

データレジスタとしては32ビット、16ビットまたは8ビットレジスタとして使用できます。

アドレスレジスタおよび32ビットレジスタとしては、一括して汎用レジスタER (ER0~ER7)として使用します。

16ビットレジスタとしては、汎用レジスタERを分割して汎用レジスタE (E0~E7)、汎用レジスタR (R0~R7)として使用します。これらは同等の機能を持っており、16ビットレジスタを最大16本を使用することができます。なお、汎用レジスタE (E0~E7)を、特に拡張レジスタと呼ぶ場合があります。

8ビットレジスタとしては、汎用レジスタRを分割して汎用レジスタRH (R0H~R7H)、汎用レジスタRL (R0L~R7L)として使用します。これらは同等の機能を持っており、8ビットレジスタを最大16本を使用することができます。

汎用レジスタの使用方法を図2.4に示します。各レジスタを独立に使用方法を選択することができます。

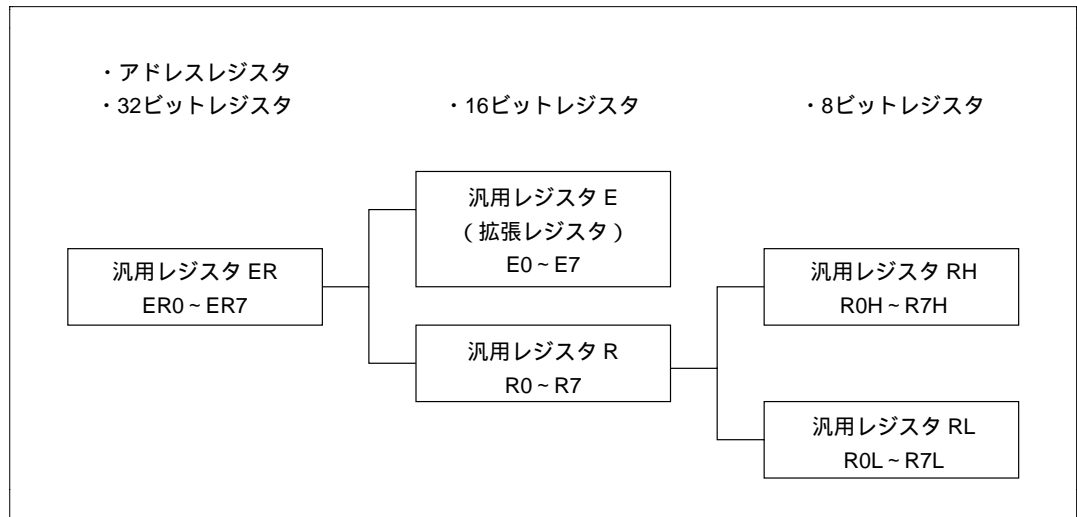


図2.4 汎用レジスタの使用方法

汎用レジスタER7には、汎用レジスタとしての機能に加えて、スタックポインタ (SP)としての機能が割り当てられており、例外処理やサブルーチン分岐などで暗黙的に使用されます。スタックの状態を図2.5に示します。

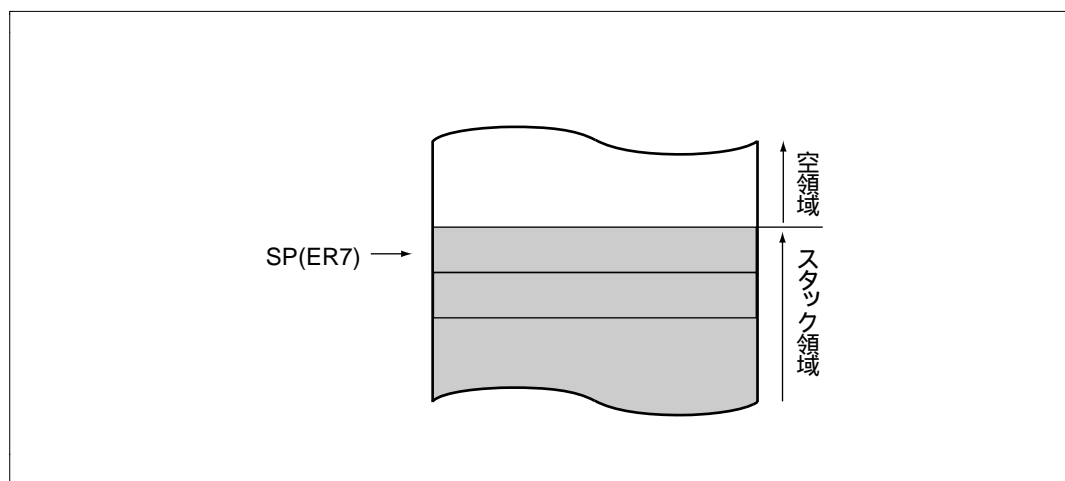


図 2.5 スタックの状態

### 2.4.3 コントロールレジスタ

コントロールレジスタには、24 ビットのプログラムカウンタ (PC) と 8 ビットのコンディションコードレジスタ (CCR) があります。

#### (1) プログラムカウンタ (PC)

24 ビットのカウンタで、CPU が次に実行する命令のアドレスを示しています。CPU の命令は、すべて 2 バイト (ワード) を単位としているため、最下位ビットは無効です (命令コードのリード時には最下位ビットは 0 とみなされます)。

#### (2) コンディションコードレジスタ (CCR)

8 ビットのレジスタで、CPU の内部状態を示しています。割込みマスクビット (I) とハーフキャリ (H)、ネガティブ (N)、ゼロ (Z)、オーバフロー (V)、キャリ (C) の各フラグを含む 8 ビットで構成されています。

##### ビット 7 : 割込みマスクビット (I)

本ビットが 1 にセットされると、割込みがマスクされます。ただし、NMI は I ビットに関係なく受け付けられます。例外処理の実行が開始されたときに 1 にセットされます。

##### ビット 6 : ユーザビット / 割込みマスクビット (UI)

ソフトウェア (LDC、STC、ANDC、ORC、XORC 命令) でリード / ライトできます。割込みマスクビットとしても使用可能です。詳細は「第 5 章 割込みコントローラ」を参照してください。

ビット5：ハーフキャリフラグ (H)

ADD.B、ADDX.B、SUB.B、SUBX.B、CMP.B、NEG.B 命令の実行により、ビット3にキャリまたはボローが生じたとき1にセットされ、生じなかったとき0にクリアされます。ADD.W、SUB.W、CMP.W、NEG.W 命令の実行によりビット11にキャリまたはボローが生じたとき、またはADD.L、SUB.L、CMP.L、NEG.L 命令の実行によりビット27にキャリまたはボローが生じたとき1にセットされ、生じなかったとき0にクリアされます。

ビット4：ユーザビット (U)

ソフトウェア (LDC、STC、ANDC、ORC、XORC 命令) でリード/ライトできます。

ビット3：ネガティブフラグ (N)

データの最上位ビットを符号ビットとみなし、最上位ビットの値を格納します。

ビット2：ゼロフラグ (Z)

データがゼロのとき1にセットされ、ゼロ以外のとき0にクリアされます。

ビット1：オーバフローフラグ (V)

算術演算命令の実行により、オーバフローが生じたとき1にセットされます。それ以外のとき0にクリアされます。

ビット0：キャリフラグ (C)

演算の実行により、キャリが生じたとき1にセットされ、生じなかったとき0にクリアされます。キャリには次の種類があります。

- (a) 加算結果のキャリ
- (b) 減算結果のボロー
- (c) シフト/ローテートのキャリ

また、キャリフラグには、ビットアキュムレータ機能があり、ビット操作命令で使用されます。

なお、命令によってはフラグが変化しない場合があります。CCR は、LDC、STC、ANDC、ORC、XORC 命令で操作することができます。また、N、Z、V、C の各フラグは、条件分岐命令 (Bcc) で使用されます。

各命令ごとのフラグの変化については、「付録 A.1 命令一覧」を参照してください。また I、UI ビットについては、「第5章 割込みコントローラ」を参照してください。

#### 2.4.4 CPU 内部レジスタの初期値

リセット例外処理によって、CPU 内部レジスタのうち、PC はベクタからロードすることにより初期化され、CCR の I ビットは 1 にセットされますが、汎用レジスタと CCR の他のビットは初期化されません。SP (ER7) の初期値も不定です。したがって、リセット直後に、MOV.L 命令を使用して SP (ER7) の初期化を行ってください。

## 2.5 データ構成

H8/300H CPU は、1ビット、4ビットBCD、8ビット（バイト）、16ビット（ワード）、および32ビット（ロングワード）のデータを扱うことができます。

1ビットデータはビット操作命令で扱われ、オペランドデータ（バイト）の第  $n$  ビット ( $n=0, 1, 2, \dots, 7$ ) という形式でアクセスされます。

なお、DAA および DAS の10進補正命令では、バイトデータは2桁の4ビットBCDデータとなります。

### 2.5.1 汎用レジスタのデータ構成

汎用レジスタのデータ構成を図 2.6、図 2.7 に示します。

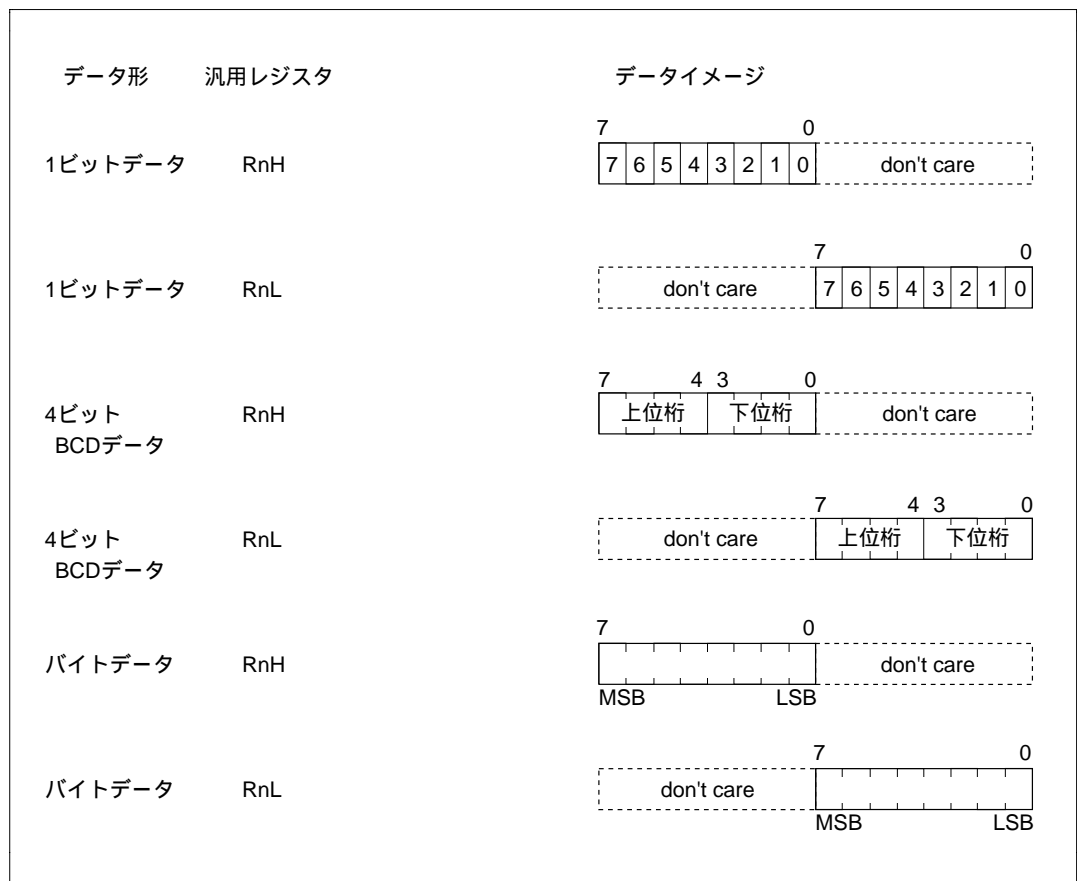


図 2.6 汎用レジスタのデータ構成 (1)

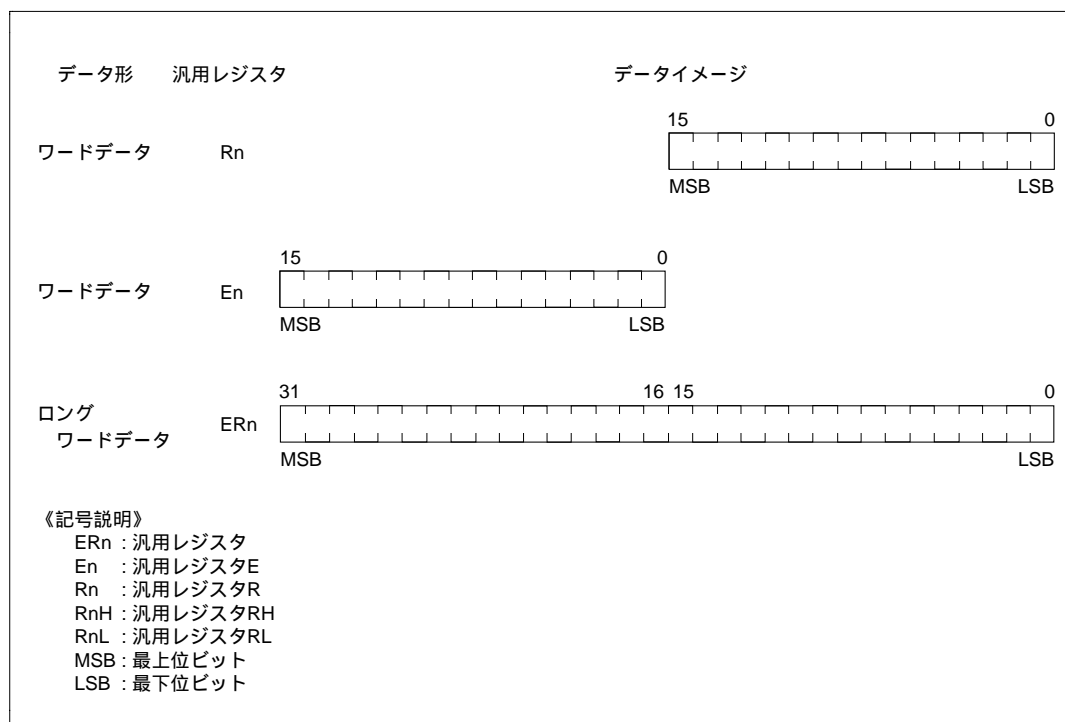


図 2.7 汎用レジスタのデータ構成 (2)

## 2.5.2 メモリ上でのデータ構成

メモリ上でのデータ構成を図 2.8 に示します。

H8/300H CPU は、メモリ上のワードデータ/ロングワードデータをアクセスすることができます。これらは、偶数番地から始まるデータに限定されます。奇数番地から始まるワードデータ/ロングワードデータをアクセスした場合、アドレスの最下位ビットは 0 とみなされ、1 番地前から始まるデータをアクセスします。この場合、アドレスエラーは発生しません。命令コードについても同様です。

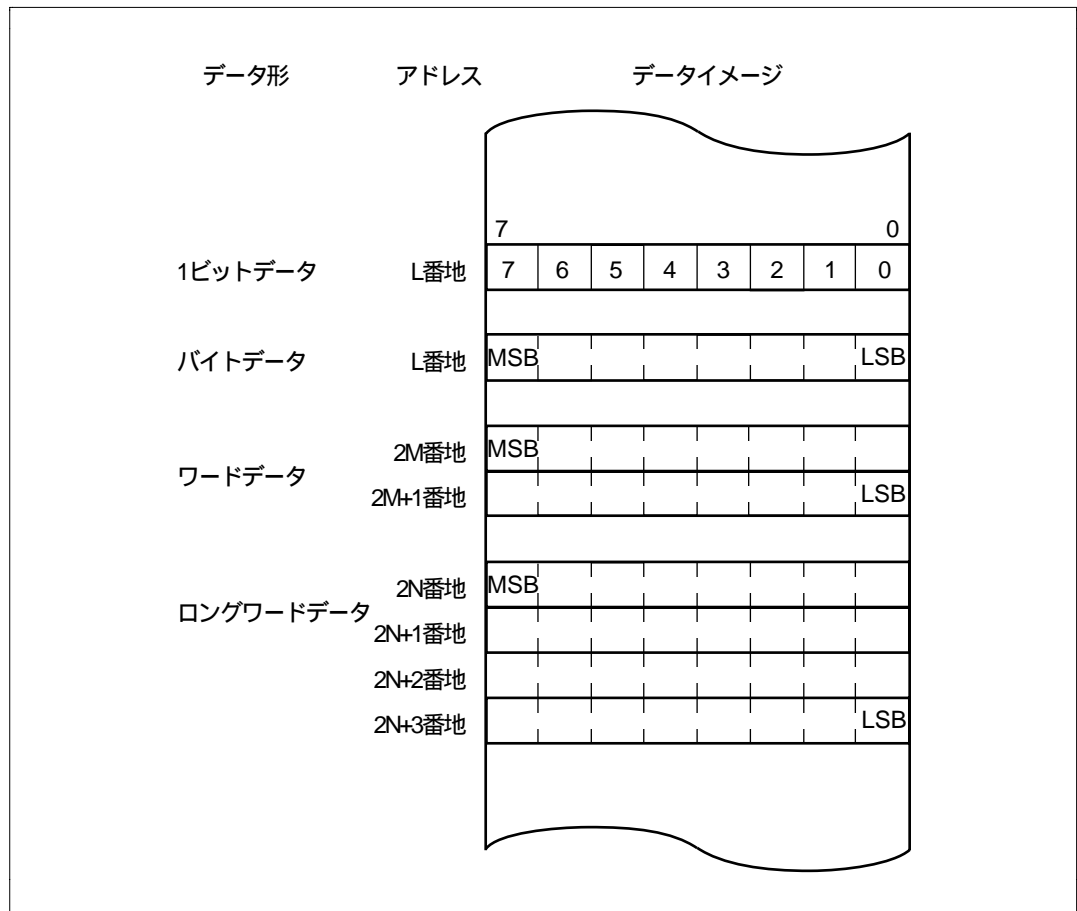


図 2.8 メモリ上でのデータ構成

なお、ER7 (SP) をアドレスレジスタとしてスタックをアクセスするときは、必ずワードサイズまたはロングワードサイズでアクセスしてください。

## 2.6 命令セット

### 2.6.1 命令セットの概要

H8/300H CPUの命令は合計 62 種類あり、各命令の機能によって、表 2.1 に示すように分類されます。

表 2.1 命令の分類

機能	命令	種類
データ転送命令	MOV、PUSH* <sup>1</sup> 、POP* <sup>1</sup> 、MOVTPE* <sup>2</sup> 、MOVFPE* <sup>2</sup>	3
算術演算命令	ADD、SUB、ADDX、SUBX、INC、DEC、ADDS、SUBS、DAA、DAS、MULXU、MULXS、DIVXU、DIVXS、CMP、NEG、EXTS、EXTU	18
論理演算命令	AND、OR、XOR、NOT	4
シフト命令	SHAL、SHAR、SHLL、SHLR、ROTL、ROTR、ROTXL、ROTXR	8
ビット操作命令	BSET、BCLR、BNOT、BTST、BAND、BIAND、BOR、BIOR、BXOR、BIXOR、BLD、BILD、BST、BIST	14
分岐命令	Bcc* <sup>3</sup> 、JMP、BSR、JSR、RTS	5
システム制御命令	TRAPA、RTE、SLEEP、LDC、STC、ANDC、ORC、XORC、NOP	9
ブロック転送命令	EEPMOV	1

合計 62 種類

【注】 \*1 POP.W Rn、PUSH.W Rn は、それぞれ MOV.W @SP+ , Rn、MOV.W Rn , @ - SP と同一です。

また、POP.L ERn、PUSH.L ERn はそれぞれ MOV.L @SP+ , Rn、MOV.L Rn , @ - SP と同一です。

\*2 本 LSI では使用できません。

\*3 Bcc は条件分岐命令の総称です。

### 2.6.2 命令とアドレッシングモードの組み合わせ

H8/300H CPU で使用可能な命令を表 2.2 に示します。



表 2.2 命令とアドレッシングモードの組み合わせ

機能	命令	アドレッシングモード															
		#xx	Rn	@ERn	@(d:16.ERn)	@(d:24.ERn)	@ERn+/@-ERn	@aa:8	@aa:16	@aa:24	@(d:8.PC)	@(d:16.PC)	@(d:8.PC)	@(d:16.PC)	@aa:8	---	
子一タ 伝送命令	MOV	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	-	
	POP, PUSH	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WL	
	MOVFP, E,	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	MOVTP, E	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	ADD, CMP	BWL	BWL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	SUB	WL	BWL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	ADDX, SUBX	B	B	-	-	-	-	-	-	-	-	-	-	-	-	-	
	ADD, SUBS	-	L	-	-	-	-	-	-	-	-	-	-	-	-	-	
	INC, DEC	-	BWL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	DAA, DAS	-	B	-	-	-	-	-	-	-	-	-	-	-	-	-	
算 術 演 算 命 令	MULXU,	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	MULXS,	-	BW	-	-	-	-	-	-	-	-	-	-	-	-	-	
	DIVXU,	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	DIVXS,	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	NEG	-	BWL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	EXTU, EXTS	-	WL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	AND, OR, XOR	-	BWL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	NOT	-	BWL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	シフト命令	-	BWL	-	-	-	-	-	-	-	-	-	-	-	-	-	
	ビット操作命令	-	B	B	-	-	-	-	-	-	-	-	-	-	B	-	
分 岐 命 令	BCC, BSR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	JMP, JSR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	RTS	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	TRAPA	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	RTE	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
シ ス テ ム 制 御 命 令	SLEEP	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	LDC	B	B	W	W	W	W	W	W	W	W	W	W	W	W	-	
	STC	-	B	W	W	W	W	W	W	W	W	W	W	W	W	-	
	ANDC, ORC	B	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	XORC	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
NOP	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-		
ブロック転送命令	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BW		

### 2.6.3 命令の機能別一覧

各命令の機能について表 2.3 ~ 表 2.10 に示します。各表で使用しているオペレーションの記号の意味は次のとおりです。

#### 《オペレーションの記号》

Rd	汎用レジスタ (デスティネーション側)*
Rs	汎用レジスタ (ソース側)*
Rn	汎用レジスタ*
ERn	汎用レジスタ (32 ビットレジスタ / アドレスレジスタ)
(EAd)	デスティネーションオペランド
(EAs)	ソースオペランド
CCR	コンディションコードレジスタ
N	CCR の N (ネガティブ) フラグ
Z	CCR の Z (ゼロ) フラグ
V	CCR の V (オーバフロー) フラグ
C	CCR の C (キャリ) フラグ
PC	プログラムカウンタ
SP	スタックポインタ
#IMM	イミディエイトデータ
disp	ディスプレースメント
+	加算
-	減算
x	乗算
÷	除算
	論理積
	論理和
⊕	排他的論理和
	転送
~	反転論理 (論理的補数)
: 3/ : 8/ : 16/ : 24	3/8/16/24 ビット長

【注】 \* 汎用レジスタは、8 ビット (R0H ~ R7H, R0L ~ R7L)、16 ビット (R0 ~ R7, E0 ~ E7)、または 32 ビットレジスタ / アドレスレジスタ (ER0 ~ ER7) です。

表 2.3 データ転送命令

命令	サイズ*	機能
MOV	B/W/L	(EAs) $\longrightarrow$ Rd, Rs $\longrightarrow$ (EAd) 汎用レジスタと汎用レジスタまたは汎用レジスタとメモリ間でデータ転送します。また、イミディエイトデータを汎用レジスタに転送します。
MOVFPPE	B	(EAs) $\longrightarrow$ Rd 本 LSI では使用できません。
MOVTPPE	B	Rs $\longrightarrow$ (EAs) 本 LSI では使用できません。
POP	W/L	@SP+ $\longrightarrow$ Rn スタックから汎用レジスタへデータを復帰します。 POP.W Rn は MOV.W @SP+ Rn と、また POP.L ERn は MOV.L @SP+ , ERn と同一です。
PUSH	W/L	Rn $\longrightarrow$ @ - SP 汎用レジスタの内容をスタックに退避します。 PUSH.W Rn は MOV.W Rn , @ - SP と、また PUSH.L ERn は MOV.L ERn , @ - SP と同一です。

【注】 \* サイズはオペランドサイズを示します。

B : バイト

W : ワード

L : ロングワード

表 2.4 算術演算命令

命令	サイズ*	機能
ADD SUB	B/W/L	$Rd \pm Rs \longrightarrow Rd$ 、 $Rd \pm \#IMM \longrightarrow Rd$ 汎用レジスタと汎用レジスタ、または汎用レジスタとイミディエイトデータ間の加減算を行います（バイトサイズでの汎用レジスタとイミディエイトデータ間の減算はできません。SUBX 命令または ADD 命令を使用してください）。
ADDX SUBX	B	$Rd \pm Rs \pm C \longrightarrow Rd$ 、 $Rd \pm \#IMM \pm C \longrightarrow Rd$ 汎用レジスタと汎用レジスタ、または汎用レジスタとイミディエイトデータ間のキャリ付き加減算を行います。
INC DEC	B/W/L	$Rd \pm 1 \longrightarrow Rd$ 、 $Rd \pm 2 \longrightarrow Rd$ 汎用レジスタに 1 または 2 を加減算します（バイトサイズの演算では 1 の加減算のみ可能です）。
ADDS SUBS	L	$Rd \pm 1 \longrightarrow Rd$ 、 $Rd \pm 2 \longrightarrow Rd$ 、 $Rd \pm 4 \longrightarrow Rd$ 32 ビットレジスタに 1、2 または 4 を加減算します。
DAA DAS	B	$Rd (10 \text{ 進補正}) \longrightarrow Rd$ 汎用レジスタ上の加減算結果を CCR を参照して 4 ビット BCD データに補正します。
MULXU	B/W	$Rd \times Rs \longrightarrow Rd$ 汎用レジスタと汎用レジスタ間の符号なし乗算を行います。 8 ビット×8 ビット 16 ビット、 16 ビット×16 ビット 32 ビットの乗算が可能です。
MULXS	B/W	$Rd \times Rs \longrightarrow Rd$ 汎用レジスタと汎用レジスタ間の符号付き乗算を行います。 8 ビット×8 ビット 16 ビット、 16 ビット×16 ビット 32 ビットの乗算が可能です。
DIVXU	B/W	$Rd \div Rs \longrightarrow Rd$ 汎用レジスタと汎用レジスタ間の符号なし除算を行います。 16 ビット÷8 ビット 商 8 ビット 余り 8 ビット、 32 ビット÷16 ビット 商 16 ビット 余り 16 ビットの除算が可能です。
DIVXS	B/W	$Rd \div Rs \longrightarrow Rd$ 汎用レジスタと汎用レジスタ間の符号付き除算を行います。 16 ビット÷8 ビット 商 8 ビット 余り 8 ビット、 32 ビット÷16 ビット 商 16 ビット 余り 16 ビットの除算が可能です。

命令	サイズ*	機能
CMP	B/W/L	Rd - Rs、Rd - #IMM 汎用レジスタと汎用レジスタ、または汎用レジスタとイミディエイトデータ間の比較を行い、その結果をCCRに反映します。
NEG	B/W/L	0 - Rd → Rd 汎用レジスタの内容の2の補数（算術的補数）をとります。
EXTS	W/L	Rd（符号拡張） → Rd 16ビットレジスタの下位8ビットをワードサイズに符号拡張します。または、32ビットレジスタの下位16ビットをロングワードサイズに符号拡張します。
EXTU	W/L	Rd（ゼロ拡張） → Rd 16ビットレジスタの下位8ビットをワードサイズにゼロ拡張します。または、32ビットレジスタの下位16ビットをロングワードサイズにゼロ拡張します。

【注】 \* サイズはオペランドサイズを示します。

B：バイト

W：ワード

L：ロングワード

表 2.5 論理演算命令

命令	サイズ*	機能
AND	B/W/L	$Rd \ Rs \longrightarrow Rd$ 、 $Rd \ #IMM \longrightarrow Rd$ 汎用レジスタと汎用レジスタ、または汎用レジスタとイミディエイトデータ間の論理積をとります。
OR	B/W/L	$Rd \ Rs \longrightarrow Rd$ 、 $Rd \ #IMM \longrightarrow Rd$ 汎用レジスタと汎用レジスタ、または汎用レジスタとイミディエイトデータ間の論理和をとります。
XOR	B/W/L	$Rd \oplus Rs \longrightarrow Rd$ 、 $Rd \oplus \#IMM \longrightarrow Rd$ 汎用レジスタ間の排他的論理和、または汎用レジスタとイミディエイトデータの排他的論理和をとります。
NOT	B/W/L	$\sim Rd \longrightarrow Rd$ 汎用レジスタの内容の1の補数（論理的補数）をとります。

【注】 \* サイズはオペランドサイズを示します。

B：バイト

W：ワード

L：ロングワード

表 2.6 シフト命令

命令	サイズ*	機能
SHAL SHAR	B/W/L	$Rd \text{ (シフト処理)} \longrightarrow Rd$ 汎用レジスタの内容を算術的にシフトします。
SHLL SHLR	B/W/L	$Rd \text{ (シフト処理)} \longrightarrow Rd$ 汎用レジスタの内容を論理的にシフトします。
ROTL ROTR	B/W/L	$Rd \text{ (ローテート処理)} \longrightarrow Rd$ 汎用レジスタの内容をローテートします。
ROTXL ROTXR	B/W/L	$Rd \text{ (ローテート処理)} \longrightarrow Rd$ 汎用レジスタの内容をキャリフラグを含めてローテートします。

【注】 \* サイズはオペランドサイズを示します。

B：バイト

W：ワード

L：ロングワード

表 2.7 ビット操作命令

命令	サイズ*	機能
BSET	B	$1 \longrightarrow (\text{ビット番号} \text{ of } \text{EAd})$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットを 1 にセットします。ビット番号は、3 ビットのイミディエイトデータまたは汎用レジスタの内容下位 3 ビットで指定します。
BCLR	B	$0 \longrightarrow (\text{ビット番号} \text{ of } \text{EAd})$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットを 0 にクリアします。ビット番号は、3 ビットのイミディエイトデータまたは汎用レジスタの内容下位 3 ビットで指定します。
BNOT	B	$\sim (\text{ビット番号} \text{ of } \text{EAd})$ $\longrightarrow (\text{ビット番号} \text{ of } \text{EAd})$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットを反転します。ビット番号は、3 ビットのイミディエイトデータまたは汎用レジスタの内容下位 3 ビットで指定します。
BTST	B	$\sim (\text{ビット番号} \text{ of } \text{EAd}) \longrightarrow Z$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットをテストし、ゼロフラグに反映します。ビット番号は、3 ビットのイミディエイトデータまたは汎用レジスタの内容下位 3 ビットで指定します。
BAND	B	$C (\text{ビット番号} \text{ of } \text{EAd}) \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットとキャリフラグとの論理積をとり、キャリフラグに結果を格納します。
BIAND	B	$C [\sim (\text{ビット番号} \text{ of } \text{EAd})] \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットを反転し、キャリフラグとの論理積をとり、キャリフラグに結果を格納します。  ビット番号は、3 ビットのイミディエイトデータで指定します。

表 2.7 ビット操作命令 (2)

命令	サイズ*	機能
BOR	B	$C (< \text{ビット番号} > \text{ of } < \text{EAd} > ) \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットとキャリフラグとの論理和をとり、キャリフラグに結果を格納します。
BIOR	B	$C [ \sim ( < \text{ビット番号} > \text{ of } < \text{EAd} > ) ] \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットを反転し、キャリフラグとの論理和をとり、キャリフラグに結果を格納します。  ビット番号は、3 ビットのイミディエイトデータで指定します。
BXOR	B	$C \oplus ( < \text{ビット番号} > \text{ of } < \text{EAd} > ) \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットとキャリフラグとの排他的論理和をとり、キャリフラグに結果を格納します。
BIXOR	B	$C \oplus [ \sim ( < \text{ビット番号} > \text{ of } < \text{EAd} > ) ] \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットを反転し、キャリフラグとの排他的論理和をとり、キャリフラグに結果を格納します。  ビット番号は、3 ビットのイミディエイトデータで指定します。
BLD	B	$( < \text{ビット番号} > \text{ of } < \text{EAd} > ) \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットをキャリフラグに転送します。
BILD	B	$\sim ( < \text{ビット番号} > \text{ of } < \text{EAd} > ) \longrightarrow C$ 汎用レジスタまたはメモリのオペランドの指定された 1 ビットを反転し、キャリフラグに転送します。  ビット番号は、3 ビットのイミディエイトデータで指定します。



命令	サイズ*	機能
BST	B	C → ( <ビット番号> of <EAd> ) 汎用レジスタまたはメモリのオペランドの指定された1ビットにキャリフラグの内容を転送します。
BIST	B	C → ~ ( <ビット番号> of <EAd> ) 汎用レジスタまたはメモリのオペランドの指定された1ビットに、反転されたキャリフラグの内容を転送します。 ビット番号は、3ビットのイミディエイトデータで指定されます。

【注】 \* サイズはオペランドサイズを示します。

B : バイト

表 2.8 分岐命令

命令	サイズ	機能																																																			
Bcc	-	指定した条件が成立しているとき、指定されたアドレスへ分岐します。分岐条件を下表に示します。																																																			
		<table border="1"> <thead> <tr> <th>ニーモニック</th> <th>説明</th> <th>分岐条件</th> </tr> </thead> <tbody> <tr> <td>BRA (BT)</td> <td>Always (True)</td> <td>Always</td> </tr> <tr> <td>BRN (BF)</td> <td>Never (False)</td> <td>Never</td> </tr> <tr> <td>BHI</td> <td>High</td> <td>C Z = 0</td> </tr> <tr> <td>BLS</td> <td>Low or Same</td> <td>C Z = 1</td> </tr> <tr> <td>Bcc (BHS)</td> <td>Carry Clear (High or Same)</td> <td>C = 0</td> </tr> <tr> <td>BCS (BLO)</td> <td>Carry Set (LOW)</td> <td>C = 1</td> </tr> <tr> <td>BNE</td> <td>Not Equal</td> <td>Z = 0</td> </tr> <tr> <td>BEQ</td> <td>Equal</td> <td>Z = 1</td> </tr> <tr> <td>BVC</td> <td>oVerflow Clear</td> <td>V = 0</td> </tr> <tr> <td>BVS</td> <td>oVerflow Set</td> <td>V = 1</td> </tr> <tr> <td>BPL</td> <td>PLus</td> <td>N = 0</td> </tr> <tr> <td>BMI</td> <td>MInus</td> <td>N = 1</td> </tr> <tr> <td>BGE</td> <td>Greater or Equal</td> <td><math>N \oplus V = 0</math></td> </tr> <tr> <td>BLT</td> <td>Less Than</td> <td><math>N \oplus V = 1</math></td> </tr> <tr> <td>BGT</td> <td>Greater Than</td> <td><math>Z (N \oplus V) = 0</math></td> </tr> <tr> <td>BLE</td> <td>Less or Equal</td> <td><math>Z (N \oplus V) = 1</math></td> </tr> </tbody> </table>	ニーモニック	説明	分岐条件	BRA (BT)	Always (True)	Always	BRN (BF)	Never (False)	Never	BHI	High	C Z = 0	BLS	Low or Same	C Z = 1	Bcc (BHS)	Carry Clear (High or Same)	C = 0	BCS (BLO)	Carry Set (LOW)	C = 1	BNE	Not Equal	Z = 0	BEQ	Equal	Z = 1	BVC	oVerflow Clear	V = 0	BVS	oVerflow Set	V = 1	BPL	PLus	N = 0	BMI	MInus	N = 1	BGE	Greater or Equal	$N \oplus V = 0$	BLT	Less Than	$N \oplus V = 1$	BGT	Greater Than	$Z (N \oplus V) = 0$	BLE	Less or Equal	$Z (N \oplus V) = 1$
		ニーモニック	説明	分岐条件																																																	
		BRA (BT)	Always (True)	Always																																																	
		BRN (BF)	Never (False)	Never																																																	
		BHI	High	C Z = 0																																																	
		BLS	Low or Same	C Z = 1																																																	
		Bcc (BHS)	Carry Clear (High or Same)	C = 0																																																	
		BCS (BLO)	Carry Set (LOW)	C = 1																																																	
		BNE	Not Equal	Z = 0																																																	
		BEQ	Equal	Z = 1																																																	
		BVC	oVerflow Clear	V = 0																																																	
		BVS	oVerflow Set	V = 1																																																	
		BPL	PLus	N = 0																																																	
		BMI	MInus	N = 1																																																	
		BGE	Greater or Equal	$N \oplus V = 0$																																																	
		BLT	Less Than	$N \oplus V = 1$																																																	
BGT	Greater Than	$Z (N \oplus V) = 0$																																																			
BLE	Less or Equal	$Z (N \oplus V) = 1$																																																			
JMP	-	指定されたアドレスへ無条件に分岐します。																																																			
BSR	-	指定されたアドレスへサブルーチン分岐します。																																																			
JSR	-	指定されたアドレスへサブルーチン分岐します。																																																			
RTS	-	サブルーチンから復帰します。																																																			

表 2.9 システム制御命令

命令	サイズ*	機能
TRAPA	-	命令トラップ例外処理を行います。
RTE	-	例外処理ルーチンから復帰します。
SLEEP	-	低消費電力状態に移ります。
LDC	B/W	(EAs) → CCR ソースオペランドを CCR に転送します。CCR はバイトサイズですが、メモリからの転送のときデータのリードはワードサイズで行われます。
STC	B/W	CCR → (EAd) CCR の内容をデスティネーションのロケーションに転送します。CCR はバイトサイズですが、メモリへの転送のときデータのライトはワードサイズで行われます。
ANDC	B	CCR #IMM → CCR CCR とイミディエイトデータの論理積をとります。
ORC	B	CCR #IMM → CCR CCR とイミディエイトデータの論理和をとります。
XORC	B	CCR ⊕ #IMM → CCR CCR とイミディエイトデータの排他的論理和をとります。
NOP	-	PC + 2 → PC PC のインクリメントだけを行います。

【注】 \* サイズはオペランドサイズを示します。

B : バイト

W : ワード

表 2.10 ブロック転送命令

命令	サイズ	機能
EEPMOV.B	-	<pre> if R4L 0 then   Repeat @ER5 + <math>\longrightarrow</math> @ER6 +、R4L - 1 <math>\longrightarrow</math> R4L   Until R4L = 0 else next; </pre>
EEPMOV.W	-	<pre> if R4 0 then   Repeat @ER5 + <math>\longrightarrow</math> @ER6 +、R4 - 1 <math>\longrightarrow</math> R4   Until R4 = 0 else next; </pre> <p>ブロック転送命令です。ER5 で示されるアドレスから始まり、R4LまたはR4 で指定されるバイト数のデータを、ER6 で示されるアドレスのロケーションへ転送します。転送終了後、次の命令を実行します。</p>

## 2.6.4 命令の基本フォーマット

H8/300H CPUの命令は、2 バイト（ワード）を単位にしています。各命令はオペレーションフィールド（OP）、レジスタフィールド（r）、EA 拡張部（EA）およびコンディションフィールド（cc）から構成されています。

### （1） オペレーションフィールド

命令の機能を表し、アドレッシングモードの指定、オペランドの処理内容を指定します。命令の先頭 4 ビットを必ず含みます。2 つのオペレーションフィールドを持つ場合もあります。

### （2） レジスタフィールド

汎用レジスタを指定します。アドレスレジスタのとき 3 ビット、データレジスタのとき 3 ビットまたは 4 ビットです。2 つのレジスタフィールドを持つ場合、またはレジスタフィールドを持たない場合もあります。

### （3） EA 拡張部

イミディエイトデータ、絶対アドレスまたはディスプレースメントを指定します。8 ビット、16 ビット、32 ビットです。24 ビットアドレスおよびディスプレースメントは上位 8 ビットをすべて 0（H'00）とした 32 ビットデータとして扱われます。

### （4） コンディションフィールド

Bcc 命令の分岐条件を指定します。

図 2.9 に命令フォーマットの例を示します。

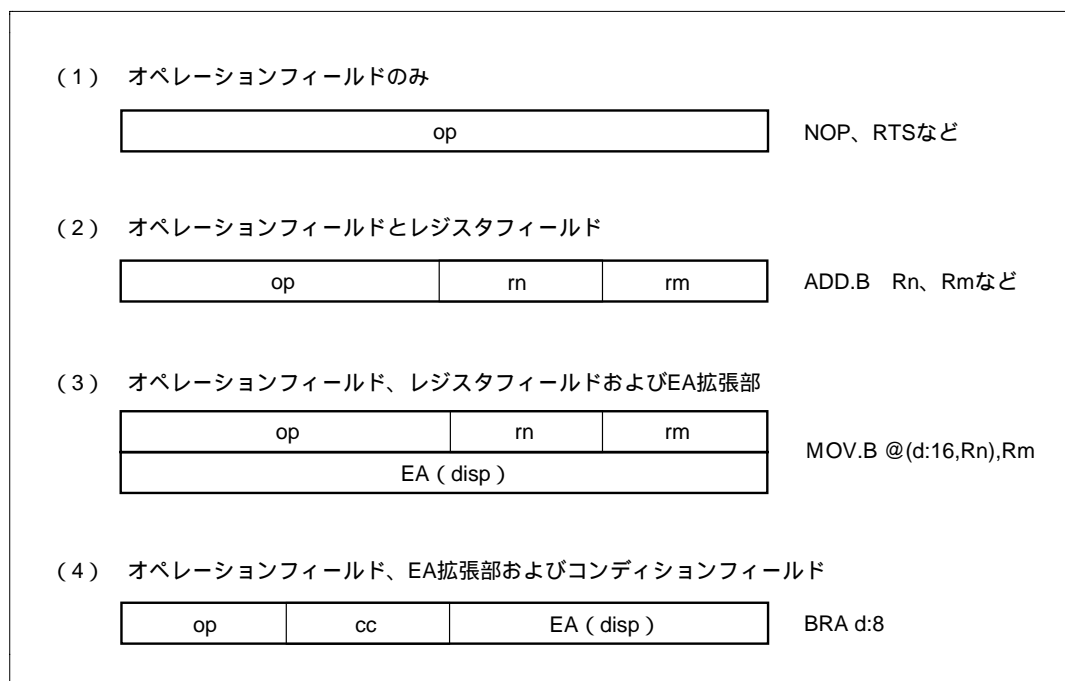


図 2.9 命令フォーマット

## 2.6.5 ビット操作命令使用上の注意

BSET、BCLR、BNOT、BST、BIST の各命令は、バイト単位でデータをリードし、ビット操作後に再びバイト単位でデータをライトします。したがって、ライト専用ビットを含むレジスタ、またはポートに対してこれらの命令を使用する場合には注意が必要です。

動作順序		動作内容
1	リード	指定したアドレスのデータ (バイト単位) をリードします。
2	ビット操作	リードしたデータの指定された 1 ビットを操作します。
3	ライト	指定したアドレスに操作したデータ (バイト単位) をライトします。

ポート 4 の DDR に、BCLR 命令を実行した例を示します。

P4<sub>7</sub>、P4<sub>6</sub>は入力ポートに設定され、P4<sub>5</sub>～P4<sub>0</sub>は出力ポートに設定されているとします。

ここで、BCLR 命令で、P4<sub>0</sub>を入力ポートにする例を示します。

## (1) BCLR 命令を実行前

	P4 <sub>7</sub>	P4 <sub>6</sub>	P4 <sub>5</sub>	P4 <sub>4</sub>	P4 <sub>3</sub>	P4 <sub>2</sub>	P4 <sub>1</sub>	P4 <sub>0</sub>
入出力	入力	入力	出力	出力	出力	出力	出力	出力
DDR	0	0	1	1	1	1	1	1

## (2) BCLR 命令を実行

`BCLR #0, @P4DDR` DDR に対して BCLR 命令を実行します。

## (3) BCLR 命令を実行後

	P4 <sub>7</sub>	P4 <sub>6</sub>	P4 <sub>5</sub>	P4 <sub>4</sub>	P4 <sub>3</sub>	P4 <sub>2</sub>	P4 <sub>1</sub>	P4 <sub>0</sub>
入出力	出力	出力	出力	出力	出力	出力	出力	入力
DDR	1	1	1	1	1	1	1	0

## (4) BCLR 命令の動作説明

BCLR 命令を実行すると、CPU は、最初に P4DDR をリードします。

P4DDR はライト専用レジスタですので、CPU は H'FF をリードします。

したがって、この例では、DDR は H'3F ですが、CPU がリードしたデータは H'FF となります。

つぎに、CPU は、リードしたデータのビット 0 を 0 にクリアして、データを H'FE に変更します。

最後に、このデータ (H'FE) を DDR に書き込んで、BCLR 命令を終了します。

その結果、P4<sub>0</sub> は、DDR が 0 になり、入力ポートになります。しかし、入力ポートであったビット 7、6 の DDR が 1 になって、出力ポートに変化してしまいます。

また、内部 I/O レジスタのフラグを 0 にクリアするために、BCLR 命令を使用することができます。この場合、割り込み処理ルーチンなどで当該フラグが 1 にセットされていることが明らかであれば、事前に当該フラグをリードする必要はありません。

## 2.7 アドレッシングモードと実効アドレスの計算方法

### 2.7.1 アドレッシングモード

H8/300H CPUは、表 2.11 に示すように、8 種類のアドレッシングモードをサポートしています。命令ごとに、使用できるアドレッシングモードは異なります。

演算命令では、レジスタ直接、およびイミディエイトが使用できます。

転送命令では、プログラムカウンタ相対とメモリ間接を除くすべてのアドレッシングモードが使用できます。

また、ビット操作命令では、オペランドの指定にレジスタ直接、レジスタ間接、および絶対アドレス (@aa:8) が使用できます。さらに、オペランド中のビット番号を指定するためにレジスタ直接 (BSET、BCLR、BNOT、BTST の各命令)、およびイミディエイト (3 ビット) が独立して使用できます。

表 2.11 アドレッシングモード一覧表

No	アドレッシングモード	記号
(1)	レジスタ直接	Rn
(2)	レジスタ間接	@ERn
(3)	ディスプレースメント付きレジスタ間接	@(d:16, ERn) / @(d:24, ERn)
(4)	ポストインクリメントレジスタ間接 プリデクリメントレジスタ間接	@ERn + @ - ERn
(5)	絶対アドレス	@aa:8 / @aa:16 / @aa:24
(6)	イミディエイト	#xx:8 / #xx:16 / #xx:32
(7)	プログラムカウンタ相対	@(d:8,PC) / @(d:16, PC)
(8)	メモリ間接	@@aa:8

#### (1) レジスタ直接 Rn

命令コードのレジスタフィールドで指定されるレジスタ(8ビット、16ビットまたは32ビット)がオペランドとなります。

8ビットレジスタとしてはR0H～R7H、R0L～R7Lを指定可能です。

16ビットレジスタとしてはR0～R7、E0～E7を指定可能です。

32ビットレジスタとしてはER0～ER7を指定可能です。

#### (2) レジスタ間接 @ERn

命令コードのレジスタフィールドで指定されるアドレスレジスタ (ERn) の内容の下位 24 ビットをアドレスとしてメモリ上のオペランドを指定します。



## (3) ディスプレースメント付きレジスタ間接 @ (d:16, ERn) / @ (d:24, ERn)

命令コードのレジスタフィールドで指定されるアドレスレジスタ (ERn) の内容に、命令コード中に含まれる 16 ビットディスプレースメントまたは 24 ビットディスプレースメントを加算した内容の下位 24 ビットをアドレスとしてメモリ上のオペランドを指定します。加算に際して、16 ビットディスプレースメントは符号拡張されます。

## (4) ポストインクリメントレジスタ間接@ERn+ / プリデクリメントレジスタ間接 @ - ERn

## ・ポストインクリメントレジスタ間接 @ERn+

命令コードのレジスタフィールドで指定されるアドレスレジスタ (ERn) の内容の下位 24 ビットをアドレスとしてメモリ上のオペランドを指定します。

その後、アドレスレジスタの内容 (32 ビット) に 1、2 または 4 が加算され、加算結果がアドレスレジスタに格納されます。バイトサイズでは 1、ワードサイズでは 2、ロングワードサイズでは 4 がそれぞれ加算されます。ワードサイズ / ロングワードサイズの場合、レジスタの内容が偶数となるようにしてください。

## ・プリデクリメントレジスタ間接 @ - ERn

命令コードのレジスタフィールドで指定されるアドレスレジスタ (ERn) の内容から 1、2 または 4 を減算した内容の下位 24 ビットをアドレスとして、メモリ上のオペランドを指定します。

その後、減算結果がアドレスレジスタに格納されます。バイトサイズでは 1、ワードサイズでは 2、ロングワードサイズでは 4 がそれぞれ減算されます。ワードサイズ、ロングワードサイズの場合、アドレスレジスタの内容が偶数となるようにしてください。

## (5) 絶対アドレス @aa:8 / @aa:16 / @aa:24

命令コード中に含まれる絶対アドレスで、メモリ上のオペランドを指定します。

絶対アドレスは 8 ビット (@aa:8)、16 ビット (@aa:16)、または 24 ビット (@aa:24) です。

8 ビット絶対アドレスの場合、上位 16 ビットはすべて 1 (H'FFFF) となります。

16 ビット絶対アドレスの場合、上位 8 ビットは符号拡張されます。

24 ビット絶対アドレスの場合、全アドレス空間をアクセスできます。

絶対アドレスのアクセス範囲を表 2.12 に示します。

表 2.12 絶対アドレスのアクセス範囲

絶対アドレス	1M バイトモード	16M バイトモード
8 ビット (@aa:8)	H'FFF00 ~ H'FFFFFF (1048320 ~ 1048575)	H'FFFF00 ~ H'FFFFFF (16776960 ~ 16777215)
16 ビット (@aa:16)	H'00000 ~ H'07FFF, H'F8000 ~ H'FFFFFF (0 ~ 32767, 1015808 ~ 1048575)	H'000000 ~ H'007FFF, H'FF8000 ~ H'FFFFFF (0 ~ 32767, 16744448 ~ 16777215)
24 ビット (@aa:24)	H'00000 ~ H'FFFFFF (0 ~ 1048575)	H'000000 ~ H'FFFFFF (0 ~ 16777215)

## (6) イミディエイト #xx:8 / #xx:16 / #xx:32

命令コードの中に含まれる 8 ビット (#xx:8)、16 ビット (#xx:16)、または 32 ビット (#xx:32) のデータを直接オペランドとして使用します。

なお、ADDS、SUBS、INC、DEC 命令では、イミディエイトデータが命令コード中に暗黙的に含まれます。ビット操作命令では、ビット番号を指定するための 3 ビットのイミディエイトデータが、命令コード中に含まれる場合があります。また、TRAPA 命令ではベクタアドレスを指定するための 2 ビットのイミディエイトデータが、命令コード中に含まれます。

## (7) プログラムカウンタ相対 @(d:8,PC) / @(d:16,PC)

Bcc、BSR 命令で使用されます。

PC の内容で指定される 24 ビットのアドレスに、命令コード中に含まれる 8 ビット、または 16 ビットディスプレースメントを加算して、24 ビットの分岐アドレスを生成します。加算に際して、ディスプレースメントは 24 ビットに符号拡張されます。また加算される PC の内容は次の命令の先頭アドレスとなっていますので、分岐可能範囲は分岐命令に対して -126 ~ +128 バイト (-63 ~ +64 ワード) または -32766 ~ +32768 バイト (-16383 ~ +16384 ワード) です。このとき、加算結果が偶数となるようにしてください。

## (8) メモリ間接 @@aa:8

JMP、JSR 命令で使用されます。

命令コードの中に含まれる 8 ビット絶対アドレスでメモリ上のオペランドを指定し、この内容を分岐アドレスとして分岐します。メモリ上のオペランドはロングワードサイズで指定します。このうち先頭 1 バイトは無視され、24 ビット長の分岐アドレスを生成します。図 2.10 にメモリ間接による分岐アドレスの指定方法を示します。

8 ビット絶対アドレスの上位のビットはすべて 0 (H'0000) となりますので、分岐アドレスを格納できるのは 0 ~ 255 (H'000000 ~ H'0000FF) 番地です。

ただし、この内の先頭領域は例外処理ベクタ領域と共通になっていますから注意してください。

詳細は「第 5 章 割込みコントローラ」を参照してください。

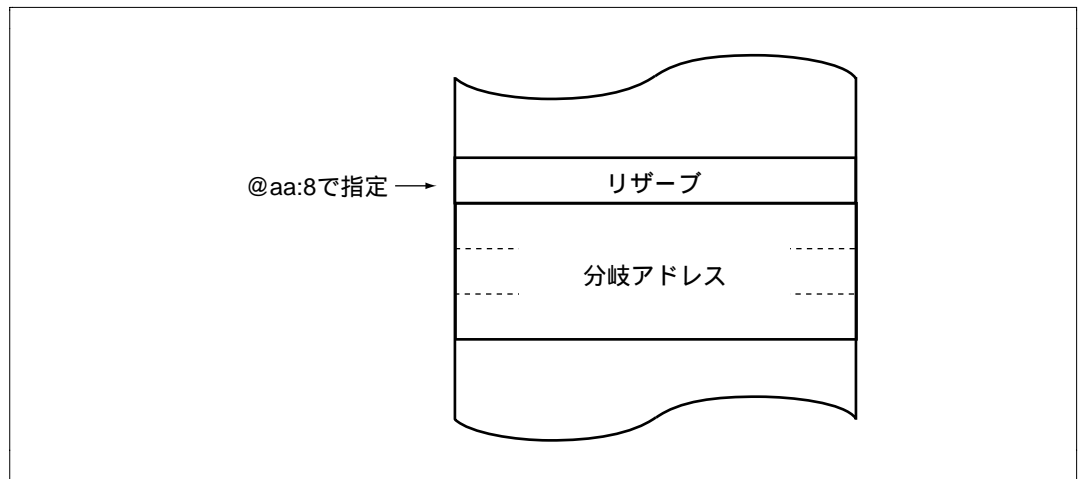


図 2.10 メモリ間接による分岐アドレスの指定





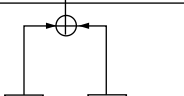
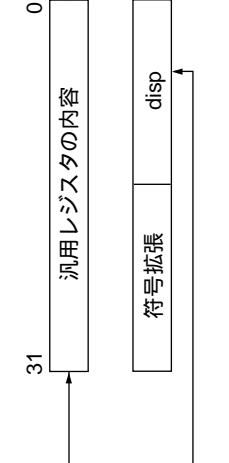
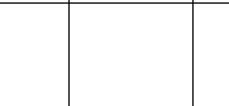


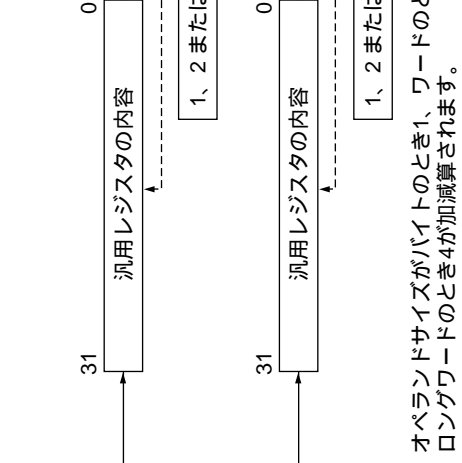
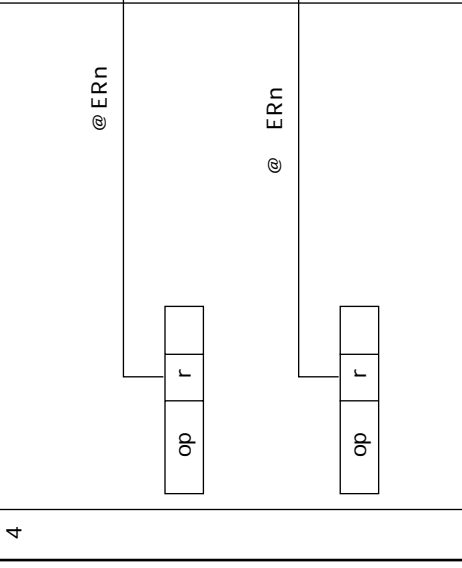
ワードサイズ、またはロングワードサイズでメモリを指定する場合、および分岐アドレスを指定する場合に奇数アドレスを指定すると、最下位ビットは0とみなされ、1番地前から始まるデータまたは命令コードをアクセスします（「2.5.2 メモリ上でのデータ構成」を参照してください）。







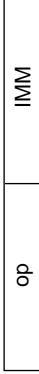


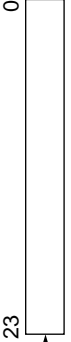
## 2.7.2 実効アドレスの計算方法

各アドレッシングモードにおける実効アドレス (EA: Effective Address) の計算方法を表 2.13 に示します。

1M バイトモードの場合、計算結果の上位4ビットは無視され、20ビットの実効アドレスを生成します。

表 2.13 実行アドレス計算方法

NO.	アドレッシングモード・命令フォーマット	実行アドレス計算方法	実行アドレス (EA)
(1)	レジスタ直接 (Rn) 		オペランドは、汎用レジスタの内容です。
(2)	レジスタ間接 (@ERn) 		
(3)	ディスプレースメント付レジスタ間接 @ (d : 16, ERn) / @ (d : 24, ERn) 		
(4)	ポストインクリメントレジスタ間接 / プリデクリメントレジスタ間接 ・ポストインクリメントレジスタ間接@ERn +  ・プリデクリメントレジスタ間接@ - ERn 	 	オペランドは、汎用レジスタの内容です。

NO.	アドレッシングモード・命令フォーマット	実効アドレス計算方法	実効アドレス (EA)
(5)	<p>絶対アドレス</p> <p>@ aa : 8</p>  <p>@ aa : 16</p>  <p>@ aa : 24</p> 		  
(6)	<p>イミディエイト # xx : 8 / # xx : 16 / # xx : 32</p> 		<p>オペランドは、イミディエイトデータです。</p>
(7)	<p>プログラムカウンタ相対</p> <p>@(d : 8, PC) / @(d : 16, PC)</p> 		

No.	アドレッシングモード・命令フォーマット	実効アドレス計算方法	実効アドレス (EA)
(8)	メモリ間接 @@ aa : 8 ・ノーマルモード op abs		
	・アドバンストモード op abs		

## 記号説明

r, rm, m : レジスタフィールド  
 OP : オペレーションフィールド  
 disp : ディスプレースメント  
 IMM : イミディエイトデータ  
 abs : 絶対アドレス

## 2.8 処理状態

### 2.8.1 概要

H8/300H CPUの処理状態には、プログラム実行状態、例外処理状態、低消費電力状態、リセット状態、およびバス権解放状態の5種類があります。さらに、低消費電力状態には、スリープモード、ソフトウェアスタンバイモード、およびハードウェアスタンバイモードがあります。処理状態の分類を図2.11に、各状態間の遷移を図2.13に示します。

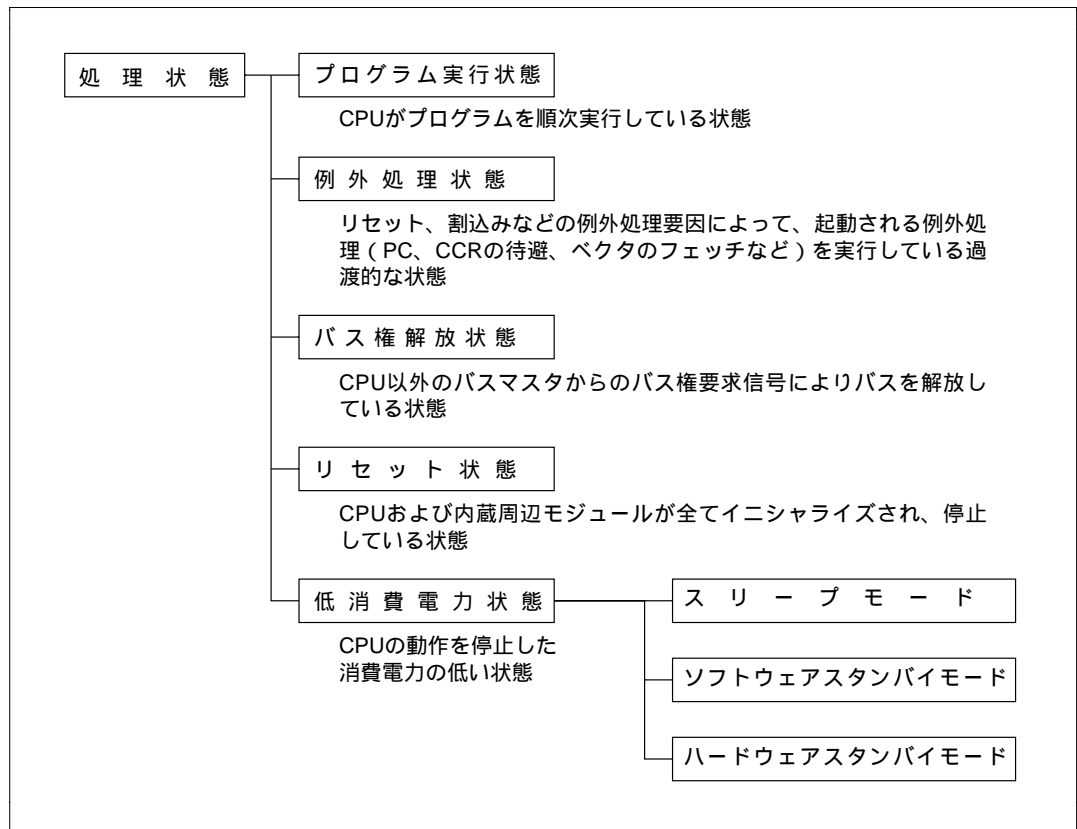


図 2.11 処理状態の分類

### 2.8.2 プログラム実行状態

CPUがプログラムを順次実行している状態です。

### 2.8.3 例外処理状態

リセット、割り込み、またはトラップ命令の例外処理要因によって起動され、CPUが通常の処理状態の流れを変え、例外処理ベクタテーブルからスタートアドレスを取り出し、その番地に分岐する過渡的な状態です。割り込みおよびトラップ命令例外処理では、SP(ER7)を参照して、PCおよびCCRの退避を行います。

### (1) 例外処理の種類と優先度

例外処理には、リセット、割込み、およびトラップ命令があります。表 2.14 に、例外処理の種類と優先度を示します。トラップ命令例外処理は、プログラム実行状態で常に受け付けられます。

表 2.14 例外処理の種類と優先度

優先度	例外処理要因	例外処理検出タイミング	例外処理開始タイミング
高 ↑	リセット	クロック同期	$\overline{RES}$ 端子が Low レベルから High レベルに変化すると、ただちに例外処理を開始します。
	割込み	命令の実行終了時または例外処理終了時*	割込み要求が発生すると、命令の実行終了時または例外処理終了時に例外処理を開始します。
低	トラップ命令	TRAPA 命令実行時	トラップ (TRAPA) 命令を実行すると、例外処理を開始します。

【注】 \* ANDC、ORC、XORC、LDC 命令の実行終了時点、またはリセット例外処理の終了時点では、割込み要因の検出を行いません。

例外処理要因は、図 2.12 に示すように分類されます。

例外処理要因とベクタ番号ならびにベクタアドレスの詳細は「第 4 章 例外処理」および「第 5 章 割込みコントローラ」を参照してください。

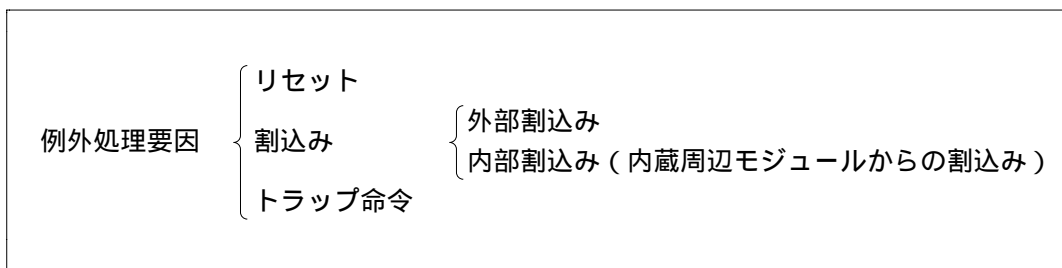


図 2.12 例外処理要因の分類



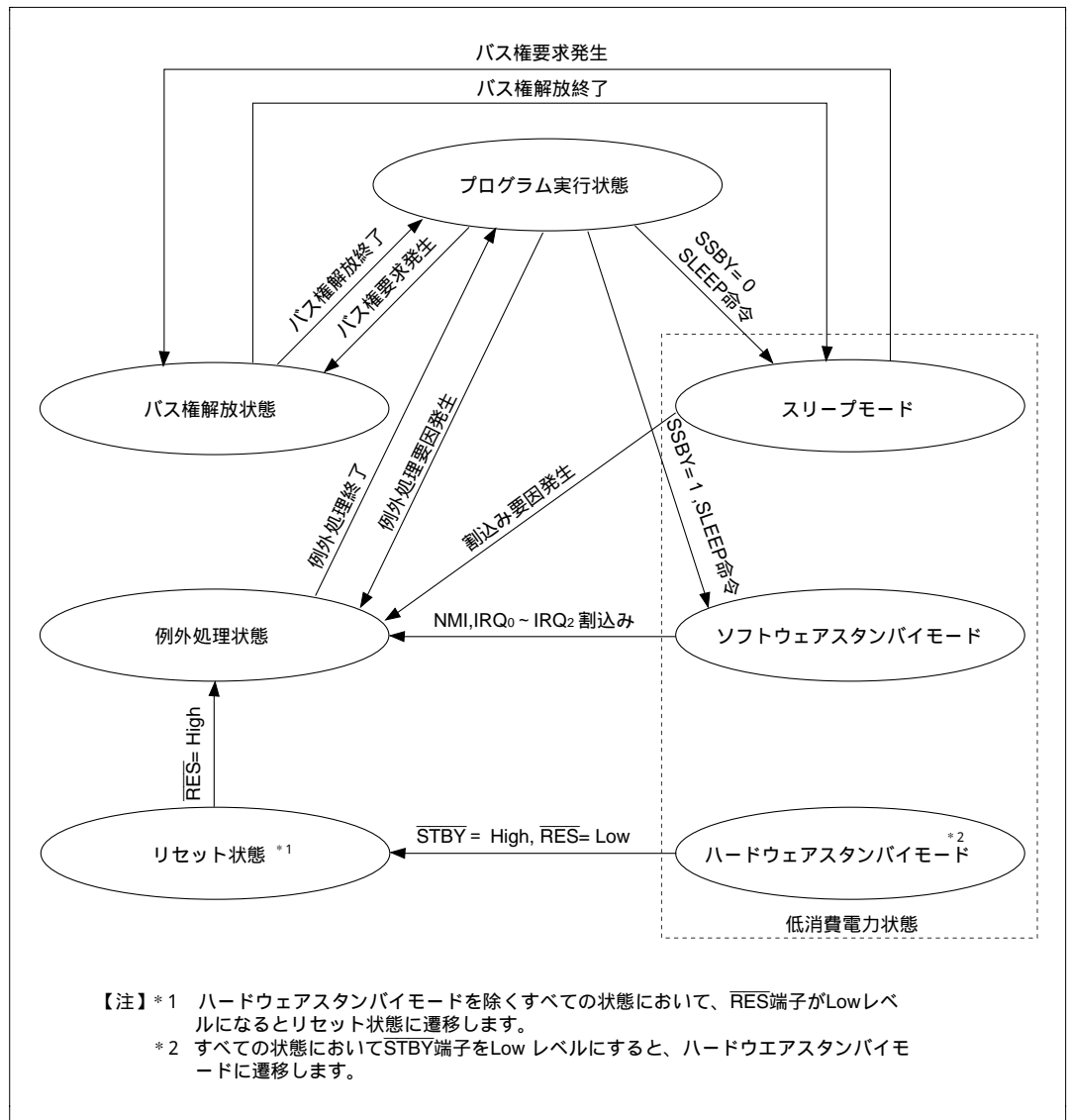


図 2.13 状態遷移図

## 2.8.4 例外処理の動作

### (1) リセット例外処理の動作

リセット例外処理は、最も優先度の高い例外処理です。 $\overline{\text{RES}}$  端子を Low レベルにしてリセット状態にした後、 $\overline{\text{RES}}$  端子を High レベルにすると、リセット例外処理が起動されます。リセット例外処理が起動されると、CPU は、例外処理ベクタテーブルからスタートアドレスを取り出し、その番地からプログラムの実行を開始します。リセット例外処理実行中、および終了後は、NMI を含めたすべての割り込みが禁止されます。

### (2) 割り込み例外処理およびトラップ命令例外処理の動作

これらの例外処理が起動されると、CPU は SP (ER7) を参照して PC と CCR をスタックに退避します。次に、SYSCR の UE ビットが 1 のときは CCR の I ビットが 1 にセットされ、UE ビットが 0 のときは CCR の I ビット、UI ビットがいずれも 1 にセットされます。

その後、例外処理ベクタテーブルからスタートアドレスを取り出して分岐します。例外処理終了後のスタックの構造を図 2.14 に示します。

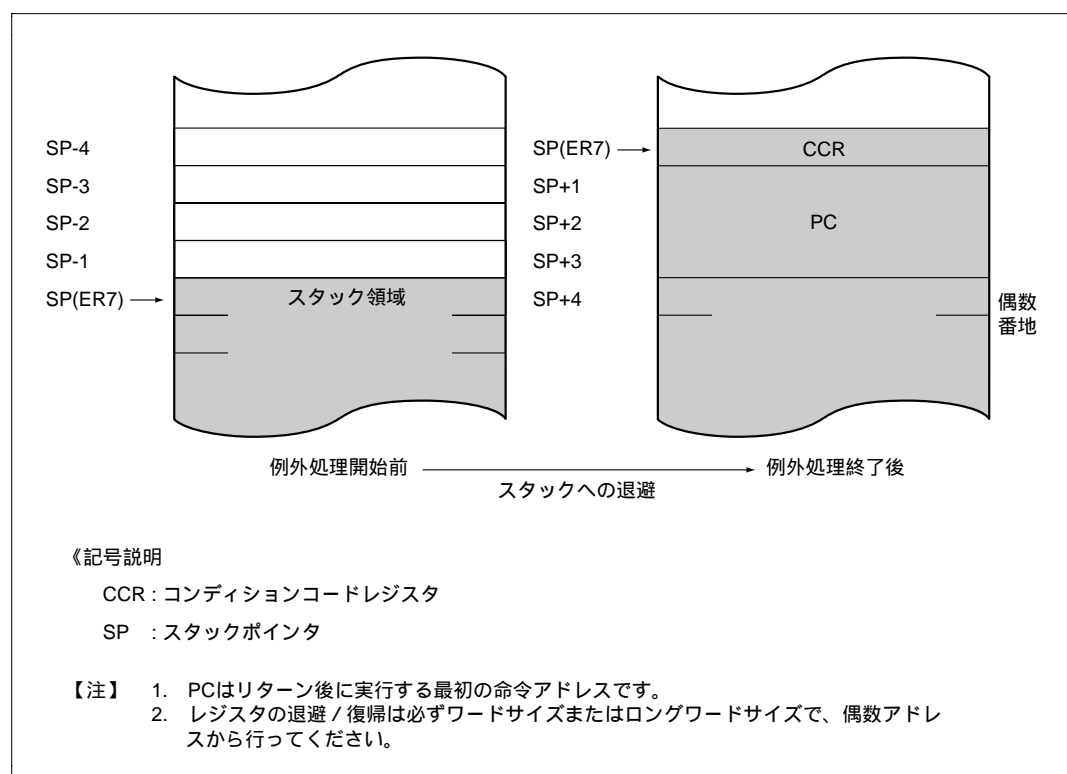


図 2.14 例外処理終了後のスタック状態

### 2.8.5 バス権解放状態

CPU 以外のバスマスタによるバス権要求に対して、バス権を解放した状態です。CPU 以外のバスマスタには DMA コントローラ、DRAM インタフェース、および外部バスマスタがあります。

バス権解放状態では、CPU は内部動作を除き、停止します。また、割込みも受け付けられません。詳細は「6.11 バスアービタ」を参照してください。

### 2.8.6 リセット状態

$\overline{\text{RES}}$  端子が Low レベルになると、実行中の処理はすべて中止され、CPU はリセット状態になります。リセットによって CCR の I ビットが 1 にセットされます。リセット状態ではすべての割込みが禁止されます。

$\overline{\text{RES}}$  端子を Low レベルから High レベルにすると、リセット例外処理が開始されます。

ウォッチドッグタイマのオーバフローによって、リセット状態とすることもできます。詳細は「第 12 章 ウォッチドッグタイマ」を参照してください。

### 2.8.7 低消費電力状態

低消費電力状態は CPU の動作を停止して、消費電力を下げる状態です。スリープモード、ソフトウェアスタンバイモード、ハードウェアスタンバイモードがあります。

#### (1) スリープモード

スリープモードは、SYSCR の SSBY ビットを 0 にクリアした状態で、SLEEP 命令を実行することによって遷移するモードです。CPU の動作は、SLEEP 命令実行直後で停止します。CPU の内部レジスタの内容は保持されます。

#### (2) ソフトウェアスタンバイモード

ソフトウェアスタンバイモードは、SYSCR の SSBY ビットを 1 にセットした状態で、SLEEP 命令を実行することによって遷移するモードです。

CPU およびクロックをはじめ内蔵周辺モジュールのすべての動作が停止します。内蔵周辺モジュールはリセット状態になりますが、規定の電圧が与えられている限り CPU の内部レジスタの内容および内蔵 RAM の内容は保持されます。また、I/O ポートの状態も保持されます。

### (3) ハードウェアスタンバイモード

ハードウェアスタンバイモードは、 $\overline{\text{STBY}}$  端子を Low レベルにすることによって遷移するモードです。ソフトウェアスタンバイモードと同様に、CPU およびすべてのクロックは停止し、内蔵周辺モジュールはリセット状態になりますが、規定の電圧が与えられている限り、内蔵 RAM の内容は保持されます。

低消費電力状態についての詳細は、「第 20 章 低消費電力状態」を参照してください。

## 2.9 基本動作タイミング

### 2.9.1 概要

H8/300H CPUは、クロック（ ）を基準に動作しています。 の立上がりから次の立上がりまでの1単位をステートと呼びます。メモリサイクルまたはバスサイクルは、2または3ステートで構成され、内蔵メモリ、内蔵周辺モジュール、または外部アドレス空間によってそれぞれ異なるアクセスを行います。外部アドレス空間のアクセスについては、バスコントローラで設定することができます。

### 2.9.2 内蔵メモリアクセスタイミング

内蔵メモリのアクセスは、2ステートアクセスを行います。このとき、データバス幅は16ビットで、バイトおよびワードサイズのアクセスが可能です。内蔵メモリアクセスサイクルを図2.15に、端子状態を図2.16に示します。

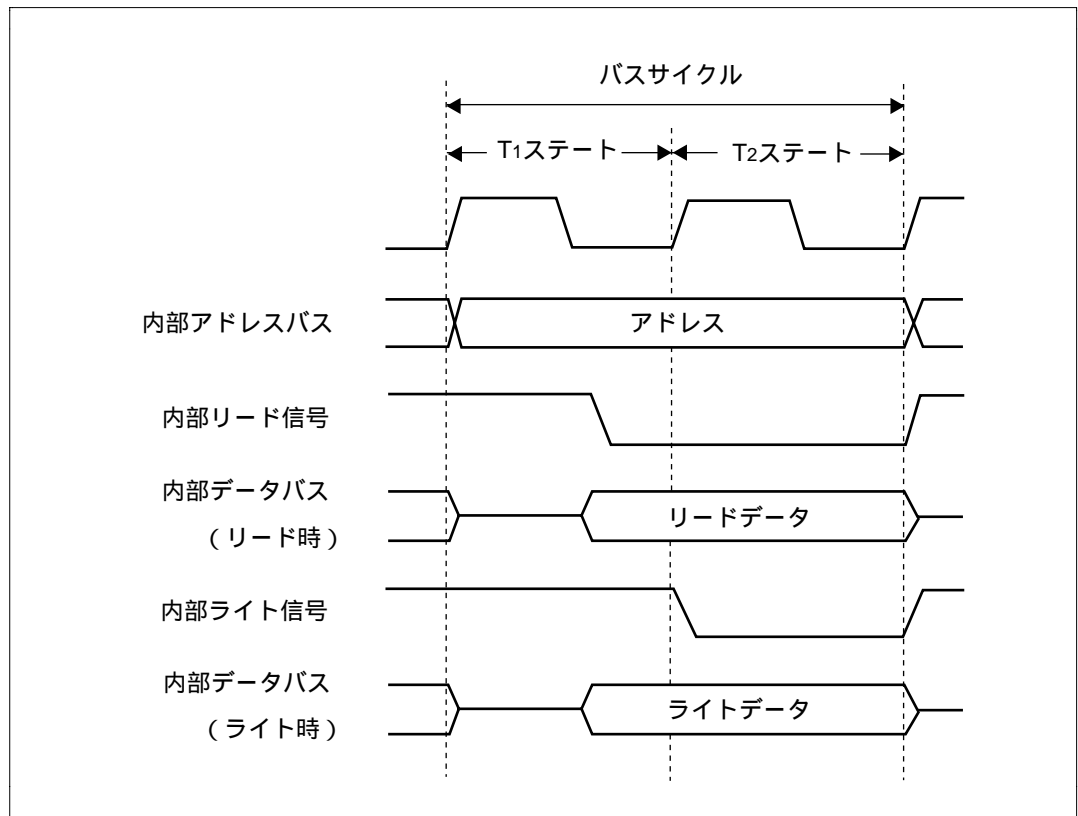


図 2.15 内蔵メモリアクセスサイクル

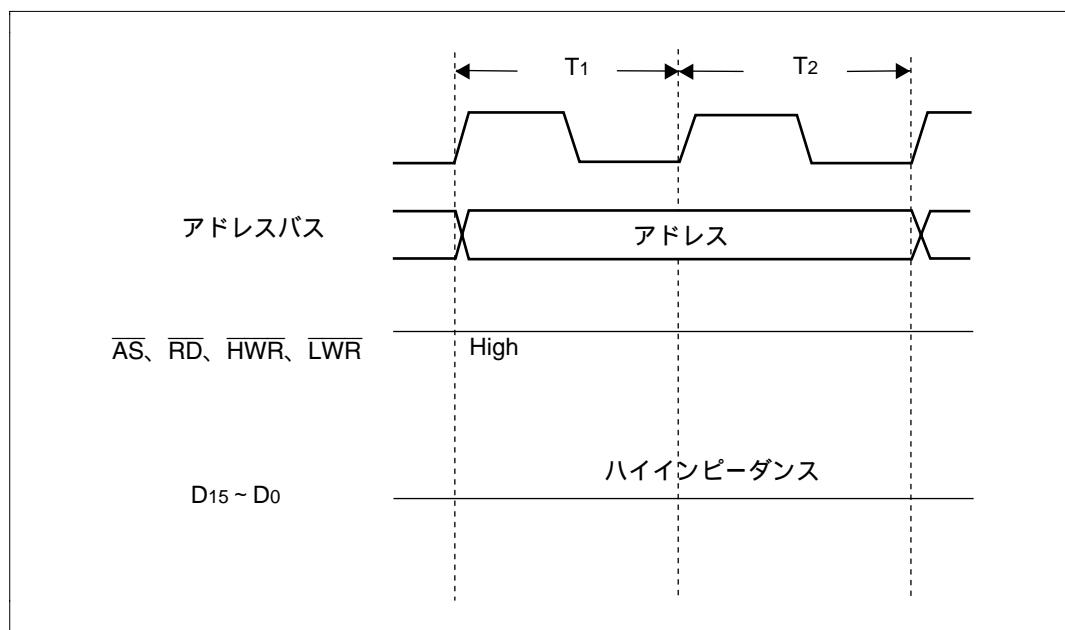


図 2.16 内蔵メモリアクセス時の端子状態

### 2.9.3 内蔵周辺モジュールアクセスタイミング

内蔵周辺モジュールのアクセスは3ステートで行われます。このとき、データバス幅は8ビットまたは16ビットであり、内部I/Oレジスタにより異なります。内蔵周辺モジュールアクセスタイミングを図2.17に、端子状態を図2.18に示します。

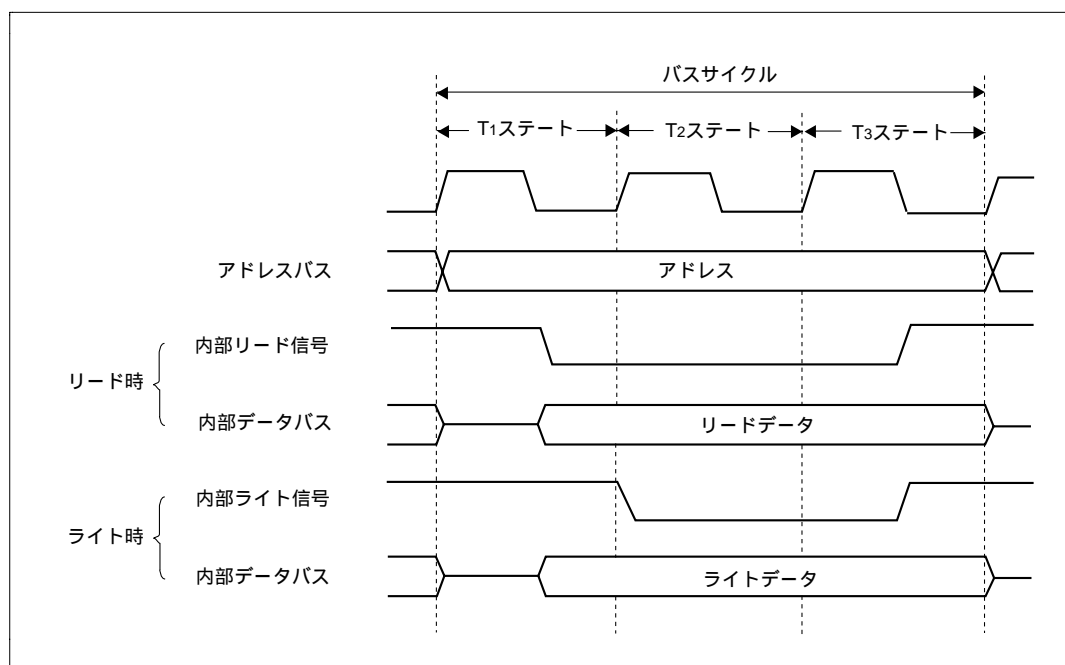


図 2.17 内蔵周辺モジュールアクセスサイクル

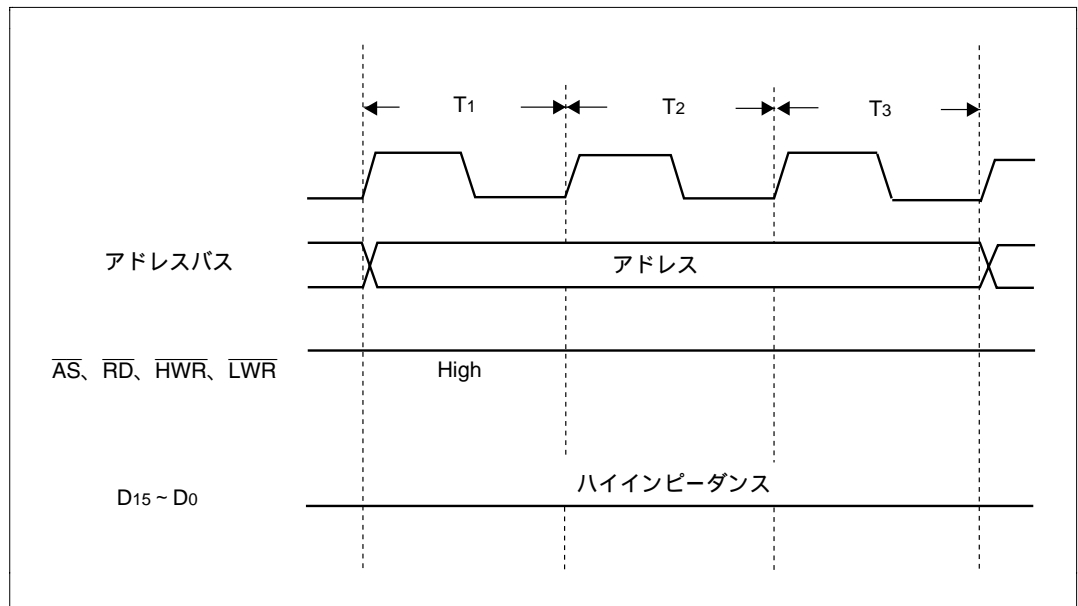


図 2.18 内蔵周辺モジュールアクセス時の端子状態

#### 2.9.4 外部アドレス空間アクセスタイミング

外部アドレス空間は8つのエリア(エリア0~7)に分割されており、バスコントローラの設定により、各エリアごとにデータバス幅(8ビットまたは16ビット)とアクセスステート(2ステートまたは3ステート)の選択ができます。

詳細は「第6章 バスコントローラ」を参照してください。

---

# 3. MCU 動作モード

---

## 第3章 目次

3.1	概要.....	71
	3.1.1 動作モードの種類を選択.....	71
	3.1.2 レジスタ構成.....	72
3.2	モードコントロールレジスタ (MDCR) .....	73
3.3	システムコントロールレジスタ (SYSCR) .....	74
3.4	各動作モードの説明.....	77
	3.4.1 モード1 .....	77
	3.4.2 モード2 .....	77
	3.4.3 モード3 .....	77
	3.4.4 モード4 .....	77
	3.4.5 モード5 .....	77
	3.4.6 モード6 .....	78
	3.4.7 モード7 .....	78
3.5	各動作モードにおける端子機能.....	79
3.6	各動作モードのメモリマップ .....	80
	3.6.1 リザーブ領域について .....	80





## 3.1 概要

### 3.1.1 動作モードの種類を選択

本LSIには、7種類の動作モード（モード1~7）があります。これらのモードは、モード端子（MD<sub>2</sub>~MD<sub>0</sub>）を表3.1のように設定することによってバスモードの初期状態とアドレス空間を選択することができます。

表 3.1 動作モードの種類を選択

動作モード	端子設定			内容			
	MD <sub>2</sub>	MD <sub>1</sub>	MD <sub>0</sub>	アドレス空間	バスモード 初期状態* <sup>1</sup>	内蔵ROM	内蔵RAM
-	0	0	0	-	-	-	-
モード1	0	0	1	拡張モード	8ビット	無効	有効* <sup>2</sup>
モード2	0	1	0	拡張モード	16ビット	無効	有効* <sup>2</sup>
モード3	0	1	1	拡張モード	8ビット	無効	有効* <sup>2</sup>
モード4	1	0	0	拡張モード	16ビット	無効	有効* <sup>2</sup>
モード5	1	0	1	拡張モード	8ビット	有効	有効* <sup>2</sup>
モード6	1	1	0	シングルチップ ノーマルモード	-	有効	有効
モード7	1	1	1	シングルチップ アドバンスモード	-	有効	有効

【注】 \*1 モード1~5において、バス幅コントロールレジスタ（ABWCR）を設定することによりデータバス幅をエリアごとに8ビットデータバスまたは16ビットデータバスにすることができます。

詳細は、「第6章 バスコントローラ」を参照してください。

\*2 SYSCRのRAMEビットを0にクリアすると外部アドレス空間に切り換わります。

アドレス空間は、64kバイト/1Mバイト/16Mバイトのいずれかを選択することができます。外部データバスのバス幅はABWCRにより、8ビット/16ビットバスモードのいずれかになります。すべてのエリアを8ビットアクセス空間に設定した場合、8ビットバスモードとなります。詳細は「第6章 バスコントローラ」を参照してください。

モード1~4は、外部メモリおよび周辺デバイスをアクセスすることができる内蔵ROM無効拡張モードです。

モード1、2でサポートするアドレス空間は、最大1Mバイトです。また、モード3、4でサポートするアドレス空間は、最大16Mバイトです。

モード5は、外部メモリおよび周辺デバイスをアクセスすることができる内蔵ROM有効拡張モードです。モード5でサポートするアドレス空間は、最大16Mバイトです。

モード6、7は、内蔵ROMとRAM、内部I/Oレジスタで動作するシングルチップモードです。すべてのポートを使用することができます。モード6でサポートするアドレス空間は最大64kバイトです。モード7でサポートするアドレス空間は最大1Mバイトです。

モード1~7以外は、本LSIでは使用できません。したがって、モード端子は必ずモード1~7になるように設定してください。

モード端子は、動作中に変化させないでください。

### 3.1.2 レジスタ構成

本LSIにはモード端子(MD<sub>2</sub>~MD<sub>0</sub>)の状態が反映されるMDCRと、動作を制御するSYSCRがあります。レジスタ構成を表3.2に示します。

表3.2 レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H'EE011	モードコントロールレジスタ	MDCR	R	不定
H'EE012	システムコントロールレジスタ	SYSCR	R/W	H'09

【注】 \* アドバンスモード時のアドレス下位20ビットを示しています。

## 3.2 モードコントロールレジスタ (MDCR)

MDCRは8ビットのリード専用のレジスタで、本LSIの現在の動作モードをモニタするのに使います。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	—	MDS2	MDS1	MDS0
初期値:	1	1	0	0	0	— *	— *	— *
R/W :	—	—	—	—	—	R	R	R

リザーブビット
リザーブビット
モードセレクト2~0  
現在の動作モード  
を示すビットです。

【注】 \* MD<sub>2</sub>~MD<sub>0</sub>端子により決定されます。

### ビット7、6：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

### ビット5~3：リザーブビット

リザーブビットです。リードすると常に0が読み出されます。ライトは無効です。

### ビット2~0：モードセレクト2~0 (MDS2~0)

これらのビットは、モード端子 (MD<sub>2</sub>~MD<sub>0</sub>) のレベルを反映した値 (現在の動作モード) を示しています。MDS2~MDS0ビットはMD<sub>2</sub>~MD<sub>0</sub>端子にそれぞれ対応します。これらのビットは、リード専用でライトは無効です。MDCRをリードすると、モード端子 (MD<sub>2</sub>~MD<sub>0</sub>) のレベルがこれらのビットにラッチされます。

### 3.3 システムコントロールレジスタ (SYSCR)

SYSCR は8ビットのレジスタで本 LSI の動作を制御します。

ビット:	7	6	5	4	3	2	1	0
	SSBY	STS2	STS1	STS0	UE	NMIEG	SSOE	RAME
初期値:	0	0	0	0	1	0	0	1
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

RAMイネーブル  
内蔵RAMの有効/  
無効を選択するビットです。

ソフトウェアスタンバイ  
出力ポートイネーブル  
ソフトウェアスタンバイ  
モード時にアドレスバス、  
バス制御信号の出力の状態  
を選択するビットです。

NMIエッジセレクト  
NMI端子の入力エッジを選択するビットです。

ユーザビットイネーブル  
CCRのUIビットをユーザビットとして使用する  
か、割り込みマスクビットとして使用するか選  
択するビットです。

スタンバイタイムセレクト2~0  
ソフトウェアスタンバイモードから復帰する場合の待機時間  
を選択するビットです。

ソフトウェアスタンバイ  
ソフトウェアスタンバイモードへの遷移を指定するビットです。

#### ビット7: ソフトウェアスタンバイ (SSBY)

ソフトウェアスタンバイモードへの遷移を指定します(ソフトウェアスタンバイモードについては「第20章 低消費電力状態」を参照してください)。

なお、外部割込みによりソフトウェアスタンバイモードが解除され、通常動作に遷移したとき、このビットは1にセットされたままです。クリアする場合は、0をライトしてください。

ビット7	説明
SSBY	
0	SLEEP 命令実行後、スリープモードに遷移 (初期値)
1	SLEEP 命令実行後、ソフトウェアスタンバイモードに遷移

## ビット6～4：スタンバイタイムセレクト2～0（STS2～0）

外部割込みによって、ソフトウェアスタンバイモードを解除する場合に、内部クロックが安定するまでCPUと内蔵周辺モジュールが待機する時間を指定します。

水晶発振の場合、動作周波数に応じて待機時間が7ms以上となるように指定してください。

待機時間の設定については、「20.4.3 ソフトウェアスタンバイモード解除後の発振安定待機時間の設定」を参照してください。

ビット6	ビット5	ビット4	説 明
STS2	STS1	STS0	
0	0	0	待機時間 = 8192 ステート (初期値)
0	0	1	待機時間 = 16384 ステート
0	1	0	待機時間 = 32768 ステート
0	1	1	待機時間 = 65536 ステート
1	0	0	待機時間 = 131072 ステート
1	0	1	待機時間 = 262144 ステート
1	1	0	待機時間 = 1024 ステート
1	1	1	使用禁止

## ビット3：ユーザビットイネーブル（UE）

CCRのUIビットをユーザビットとして使用するか、割込みマスクビットとして使用するかを選択します。

ビット3	説 明
UE	
0	CCRのUIビットを、割込みマスクビットとして使用
1	CCRのUIビットを、ユーザビットとして使用 (初期値)

## ビット2：NMIエッジセレクト（NMIEG）

NMI端子の入力エッジ選択を行います。

ビット2	説 明
NMIEG	
0	NMI入力の立下がりエッジで割込み要求を発生 (初期値)
1	NMI入力の立上がりエッジで割込み要求を発生

ビット1：ソフトウェアスタンバイ出力ポートイネーブル（SSOE）

ソフトウェアスタンバイモード時にアドレスバス、バス制御信号（ $\overline{CS}_0 \sim \overline{CS}_7$ 、 $\overline{AS}$ 、 $\overline{RD}$ 、 $\overline{HWR}$ 、 $\overline{LWR}$ 、 $\overline{UCAS}$ 、 $\overline{LCAS}$ 、 $\overline{RFSH}$ ）の出力を保持またはHigh固定するか、ハイインピーダンスにするかを指定します。

ビット1	説明
SSOE	
0	ソフトウェアスタンバイモード時、アドレスバス、バス制御信号はすべてハイインピーダンス (初期値)
1	ソフトウェアスタンバイモード時、 アドレスバス：出力状態を保持 バス制御信号：High 固定

ビット0：RAMイネーブル（RAME）

内蔵RAMの有効/無効を選択します。RAMEビットは、 $\overline{RES}$ 端子の立上がりエッジでイニシャライズされます。ソフトウェアスタンバイモードでは、イニシャライズされません。

ビット0	説明
RAME	
0	内蔵RAMの無効
1	内蔵RAMの有効 (初期値)

## 3.4 各動作モードの説明

### 3.4.1 モード1

ポート1、2、5の機能がアドレス端子  $A_{19} \sim A_0$  となり、最大 1M バイトのアドレス空間をアクセスできます。リセット直後は8ビットバスモードとなり、すべてのエリアは8ビットアクセス空間となります。ただし、ABWCRにより少なくとも1つのエリアを16ビットアクセス空間に設定した場合には、16ビットバスモードとなります。

### 3.4.2 モード2

ポート1、2、5の機能がアドレス端子  $A_{19} \sim A_0$  となり、最大 1M バイトのアドレス空間をアクセスできます。リセット直後は16ビットバスモードとなり、すべてのエリアは16ビットアクセス空間となります。ただし、ABWCRによりすべてのエリアを8ビットアクセス空間に設定した場合には、8ビットバスモードとなります。

### 3.4.3 モード3

ポート1、2、5およびポートAの一部の機能がアドレス端子  $A_{23} \sim A_0$  となり、最大 16M バイトのアドレス空間をアクセスできます。バスモードはリセット直後に8ビットバスモードとなり、すべてのエリアは8ビットアクセス空間となります。ただし、ABWCRにより少なくとも1つのエリアを16ビットアクセス空間に設定した場合には、16ビットバスモードとなります。 $A_{23} \sim A_{21}$  は、バスリリースコントロールレジスタ (BRCR) のビット7~5に0をライトすると有効になります(本モードでは  $A_{20}$  は常に出力となります)。

### 3.4.4 モード4

ポート1、2、5およびポートAの一部の機能がアドレス端子  $A_{23} \sim A_0$  となり、最大 16M バイトのアドレス空間をアクセスできます。バスモードはリセット直後に16ビットバスモードとなり、すべてのエリアは16ビットアクセス空間となります。ただし、ABWCRによりすべてのエリアを8ビットアクセス空間に設定した場合には、8ビットバスモードとなります。 $A_{23} \sim A_{21}$  は、BRCRのビット7~5に0をライトすると有効になります(本モードでは  $A_{20}$  は常に出力となります)。

### 3.4.5 モード5

ポート1、2、5およびポートAの一部の機能がアドレス端子  $A_{23} \sim A_0$  となり、最大 16M バイトのアドレス空間をアクセスできます。リセット直後は入力ポートになっています。したがってポート1、2、5をアドレスバスとして使用する場合は、各々の対応するデータディレクションレジスタ (P1DDR、P2DDR、P5DDR) を1にセットして、ポート1、2、5を出力に設定してください。また、 $A_{23} \sim A_{20}$  を出力する場合には、BRCRのビット7~4に0をライトしてください。



バスモードはリセット直後に、8ビットバスモードとなり、すべてのエリアは8ビットアクセス空間となります。ただし、ABWCRにより、少なくとも1つのエリアを16ビットアクセス空間に設定した場合には、16ビットバスモードとなります。

#### 3.4.6 モード6

内蔵ROMとRAM、内部I/Oレジスタで動作するモードです。すべてのポートを使用することができます。

モード6はアドレス空間が64kバイトとなります。

#### 3.4.7 モード7

内蔵ROMとRAM、内部I/Oレジスタで動作するモードです。すべてのポートを使用することができます。

モード7はアドレス空間が1Mバイトとなります。

### 3.5 各動作モードにおける端子機能

動作モードによりポート1~5、およびポートAの端子機能が切り換わります。各動作モードにおける端子機能の一覧を表3.3に示します。

表3.3 各動作モードにおけるポート1~5、およびポートAの機能

ポート	モード1	モード2	モード3	モード4	モード5	モード6	モード7
ポート1	A <sub>7</sub> ~A <sub>0</sub>	A <sub>7</sub> ~A <sub>0</sub>	A <sub>7</sub> ~A <sub>0</sub>	A <sub>7</sub> ~A <sub>0</sub>	P1 <sub>7</sub> ~P1 <sub>0</sub> * <sup>2</sup>	P1 <sub>7</sub> ~P1 <sub>0</sub>	P1 <sub>7</sub> ~P1 <sub>0</sub>
ポート2	A <sub>15</sub> ~A <sub>8</sub>	A <sub>15</sub> ~A <sub>8</sub>	A <sub>15</sub> ~A <sub>8</sub>	A <sub>15</sub> ~A <sub>8</sub>	P2 <sub>7</sub> ~P2 <sub>0</sub> * <sup>2</sup>	P2 <sub>7</sub> ~P2 <sub>0</sub>	P2 <sub>7</sub> ~P2 <sub>0</sub>
ポート3	D <sub>15</sub> ~D <sub>8</sub>	D <sub>15</sub> ~D <sub>8</sub>	D <sub>15</sub> ~D <sub>8</sub>	D <sub>15</sub> ~D <sub>8</sub>	D <sub>15</sub> ~D <sub>8</sub>	P3 <sub>7</sub> ~P3 <sub>0</sub>	P3 <sub>7</sub> ~P3 <sub>0</sub>
ポート4	P4 <sub>7</sub> ~P4 <sub>0</sub> * <sup>1</sup>	D <sub>7</sub> ~D <sub>0</sub> * <sup>1</sup>	P4 <sub>7</sub> ~P4 <sub>0</sub> * <sup>1</sup>	D <sub>7</sub> ~D <sub>0</sub> * <sup>1</sup>	P4 <sub>7</sub> ~P4 <sub>0</sub> * <sup>1</sup>	P4 <sub>7</sub> ~P4 <sub>0</sub>	P4 <sub>7</sub> ~P4 <sub>0</sub>
ポート5	A <sub>19</sub> ~A <sub>16</sub>	A <sub>19</sub> ~A <sub>16</sub>	A <sub>19</sub> ~A <sub>16</sub>	A <sub>19</sub> ~A <sub>16</sub>	P5 <sub>3</sub> ~P5 <sub>0</sub> * <sup>2</sup>	P5 <sub>3</sub> ~P5 <sub>0</sub>	P5 <sub>3</sub> ~P5 <sub>0</sub>
ポートA	PA <sub>7</sub> ~PA <sub>4</sub>	PA <sub>7</sub> ~PA <sub>4</sub>	PA <sub>6</sub> ~PA <sub>4</sub> , A <sub>20</sub> * <sup>3</sup>	PA <sub>6</sub> ~PA <sub>4</sub> , A <sub>20</sub> * <sup>3</sup>	PA <sub>7</sub> ~PA <sub>4</sub> * <sup>4</sup>	PA <sub>7</sub> ~PA <sub>4</sub>	PA <sub>7</sub> ~PA <sub>4</sub>

- 【注】 \*1 初期状態を示しています。ABWCRの設定により、バスモードを切り替えることができます。8ビットモード時にはP4<sub>7</sub>~P4<sub>0</sub>に、16ビットバスモード時にはD<sub>7</sub>~D<sub>0</sub>となります。
- \*2 初期状態を示しています。各々対応するデータディレクションレジスタ(P1DDR、P2DDR、P5DDR)を1に設定することにより、アドレスバスとなります。
- \*3 初期状態を示しています。A<sub>20</sub>は常にアドレス出力です。PA<sub>6</sub>~PA<sub>4</sub>は、BRCCRのビット7~5に0をライトすることによりA<sub>23</sub>~A<sub>21</sub>出力になります。
- \*4 初期状態を示しています。PA<sub>7</sub>~PA<sub>4</sub>は、BRCCRのビット7~4に0をライトすることによりA<sub>23</sub>~A<sub>20</sub>出力になります。

## 3.6 各動作モードのメモリマップ

本 LSI のメモリマップを図 3.1 ~ 図 3.3 に示します。アドレス空間は 8 エリアに分割されています。

BCR の EMC ビットを書き換えることにより、2 種類のメモリマップを選択可能です。詳細は「6.2.5 バスコントロールレジスタ」を参照してください。

モード 1 とモード 2、モード 3 とモード 4 ではそれぞれバスモードの初期状態が異なります。

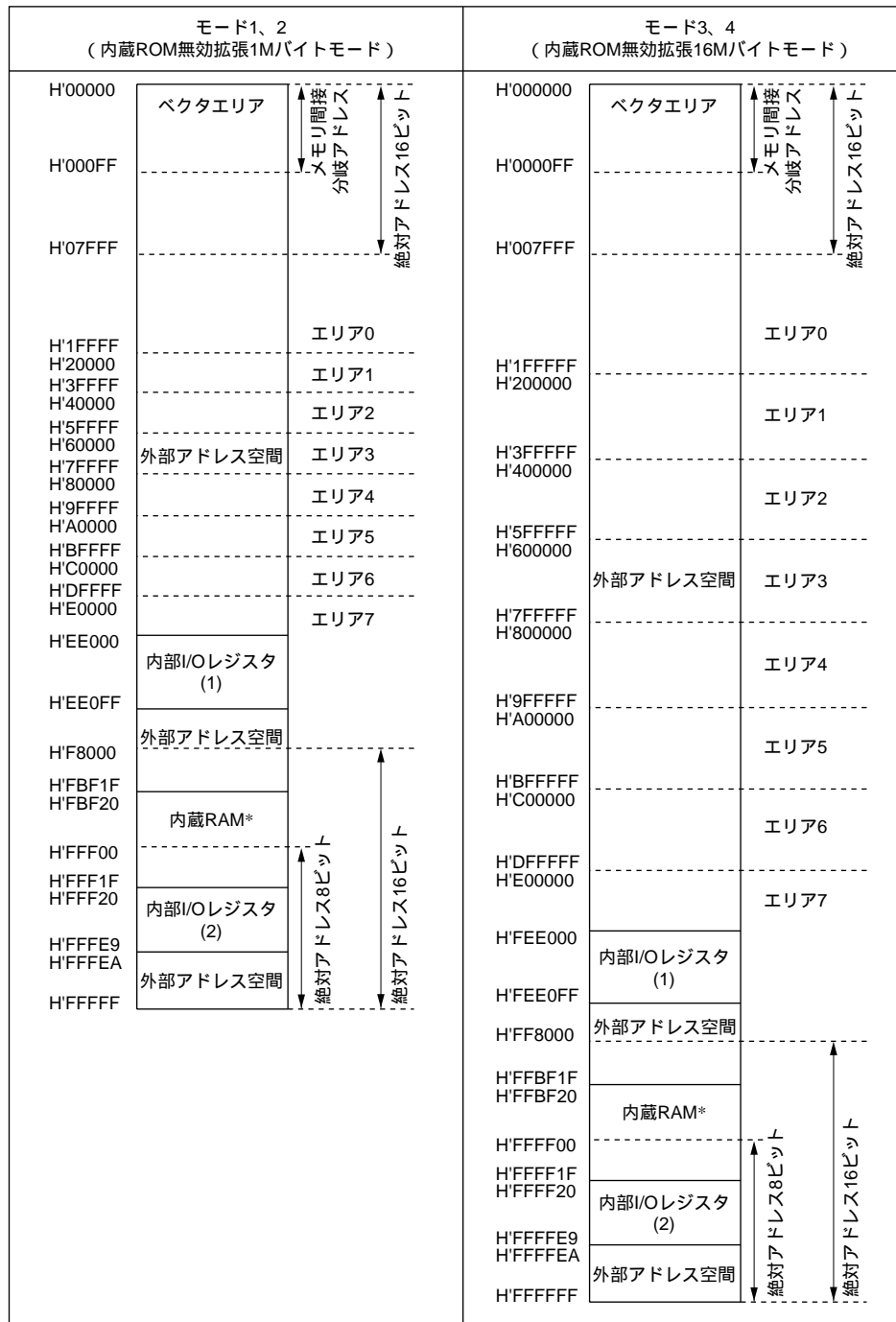
また、モード 6 (64k バイトモード) とモード 1、2、7 (1M バイトモード) とモード 3、4、5 (16M バイトモード) で、内蔵 RAM および内部 I/O レジスタの配置が異なります。また、CPU のアドレッシングモードのうち、絶対アドレス 8 ビット / 16 ビット (@aa:8/@aa:16) で指定できる範囲が異なります。

### 3.6.1 リザーブ領域について

本 LSI のメモリマップ上にはリード / ライトアクセスが禁止されているリザーブ領域があります。下記のリザーブ領域にアクセスした場合、正常動作が保証されないので注意してください。

#### (1) 内部 I/O レジスタ空間のリザーブ領域

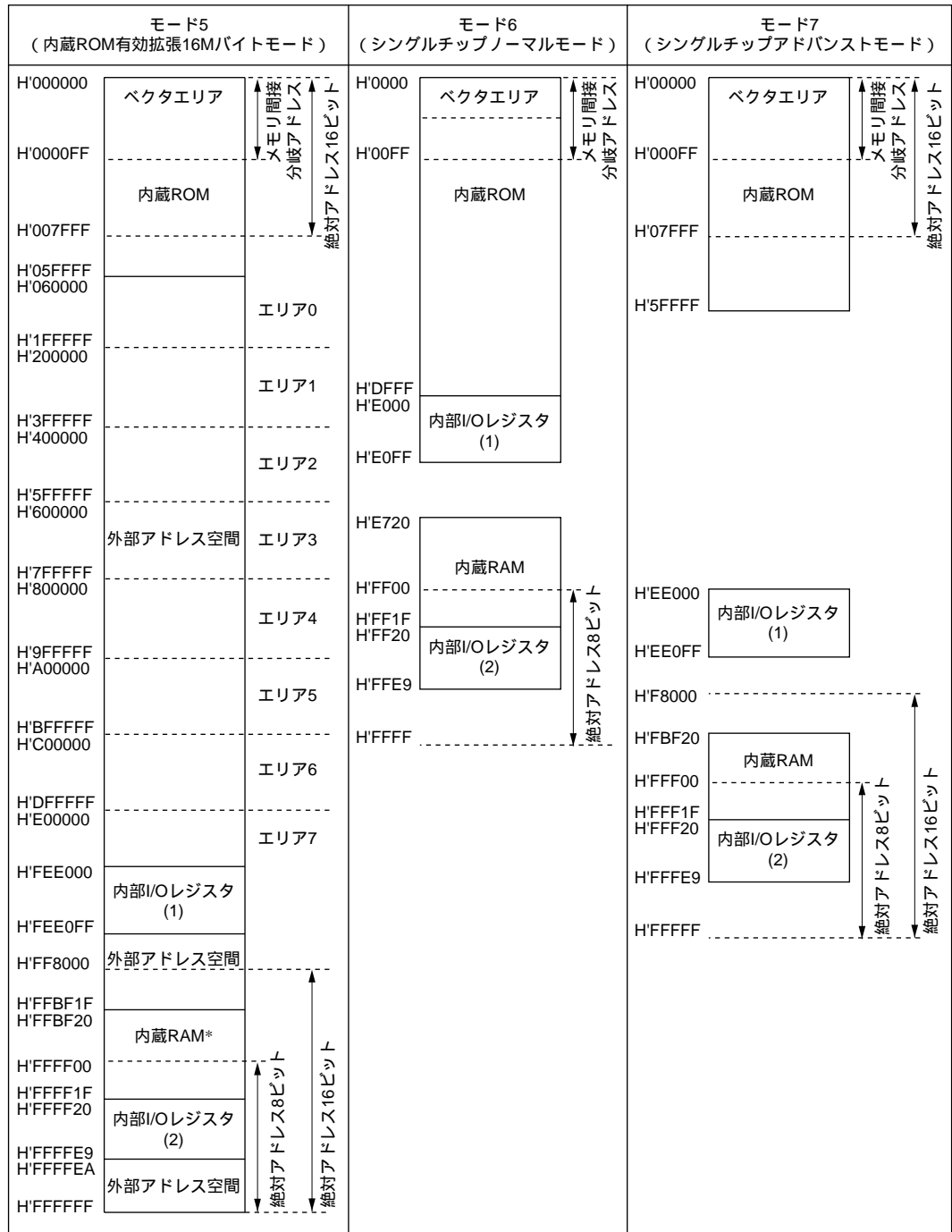
H8/3068F の内部 I/O レジスタ空間にはアクセス禁止のリザーブ領域があります。詳細は、「付録 B 内部 I/O レジスタ一覧」を参照してください。



【注】\* 内蔵RAMをディスエーブルにすると外部アドレス空間になります。

図 3.1 H8/3068F の各動作モードにおけるメモリマップ (EMC ビット=1 のとき) (1)

3. MCU動作モード



【注】\* 内蔵RAMをディスエーブルにすると外部アドレス空間になります。

図 3.1 H8/3068F の各動作モードにおけるメモリマップ (EMC ビット=1 のとき) (2)

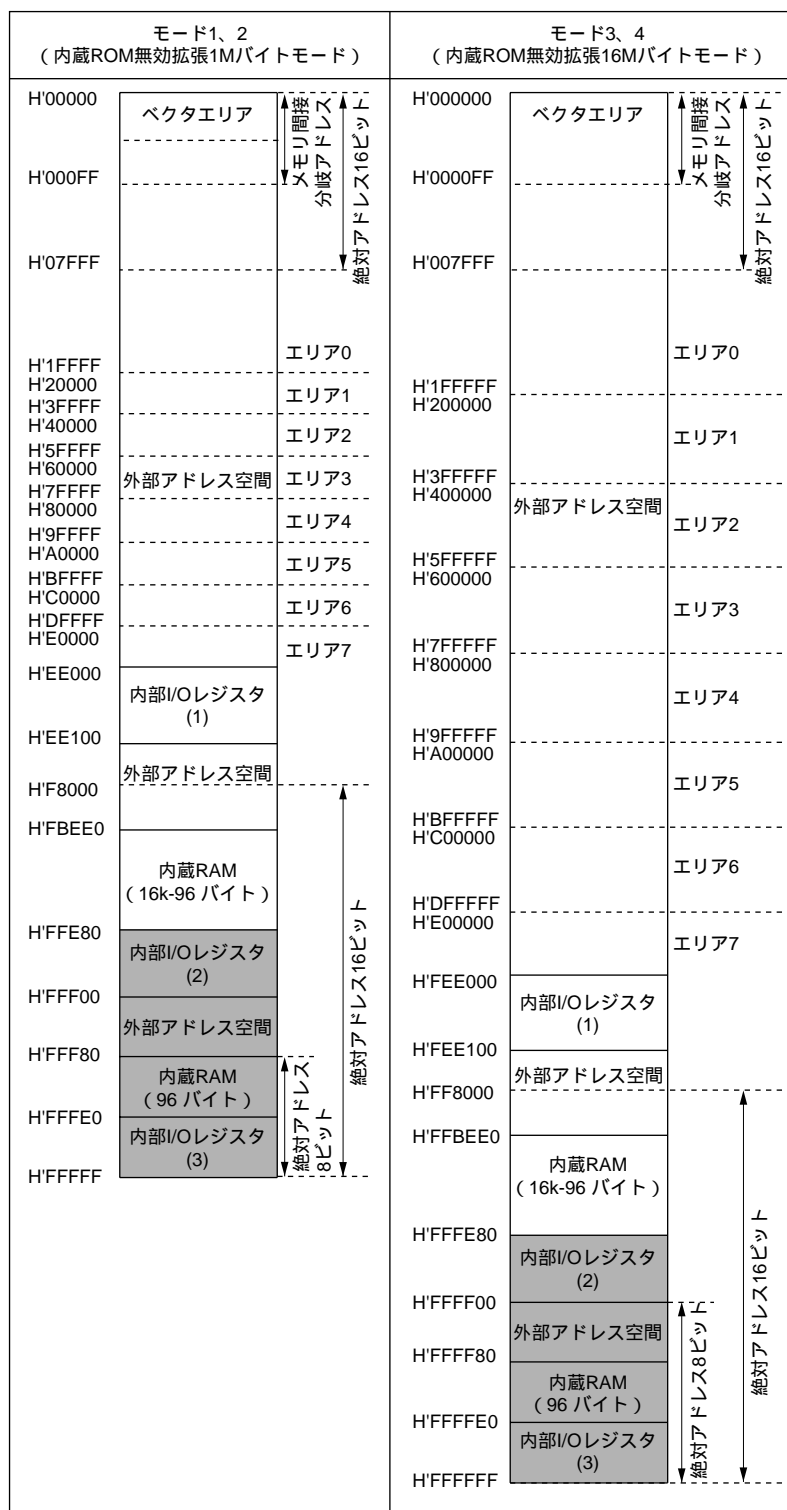


図 3.2 H8/3068F の各動作モードにおけるメモリマップ (EMC ビット=0 のとき) (1)

3. MCU動作モード

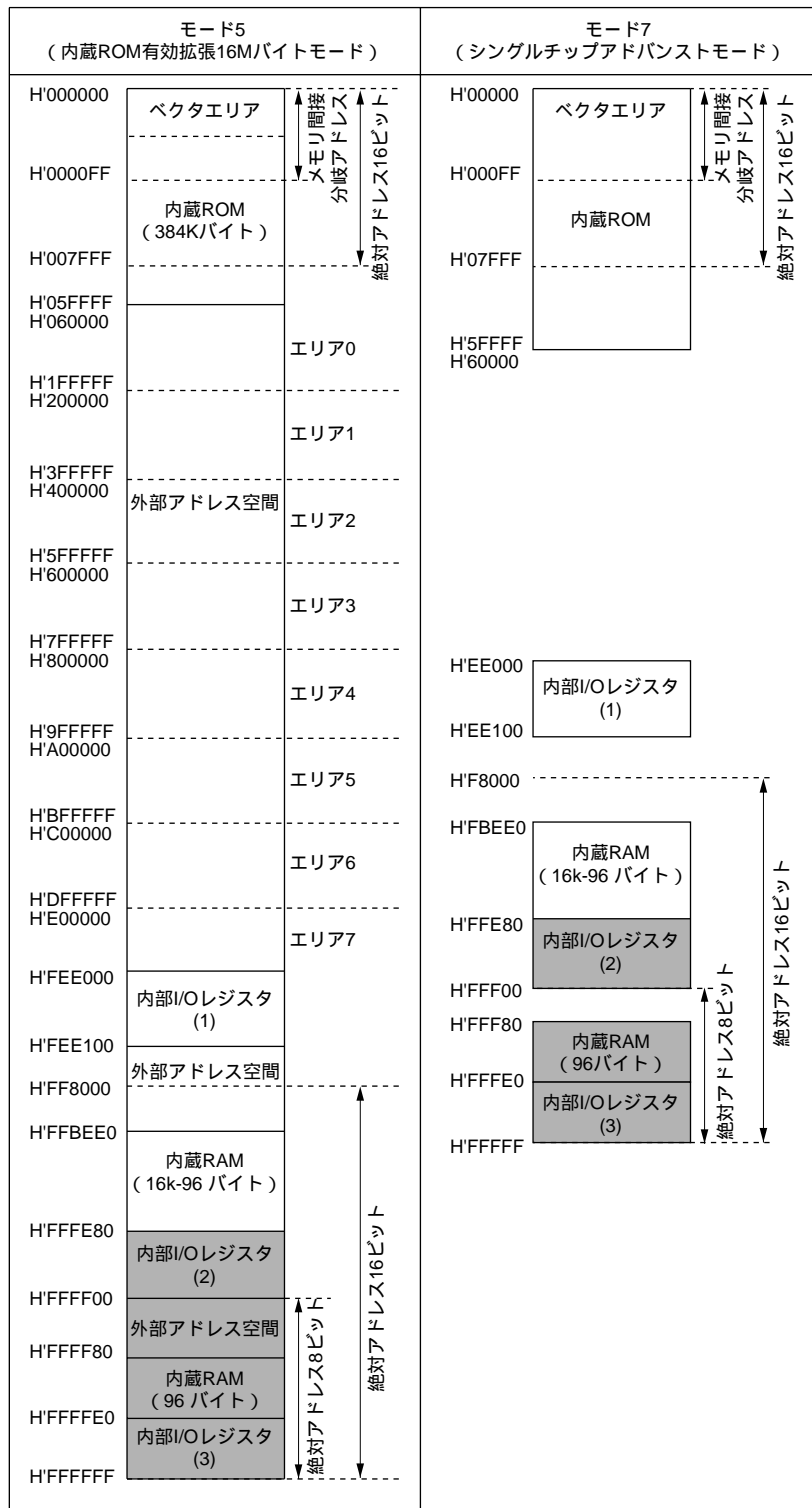


図 3.2 H8/3068F の各動作モードにおけるメモリマップ (EMC ビット=0 のとき) (2)

---

# 4. 例外処理

---

## 第4章 目次

4.1	概要	87
4.1.1	例外処理の種類と優先度	87
4.1.2	例外処理の動作	87
4.1.3	例外処理要因とベクタテーブル	87
4.2	リセット	89
4.2.1	概要	89
4.2.2	リセットシーケンス	89
4.2.3	リセット直後の割込み	91
4.3	割込み	92
4.4	トラップ命令	93
4.5	例外処理後のスタックの状態	94
4.6	スタック使用上の注意	95





## 4.1 概要

### 4.1.1 例外処理の種類と優先度

例外処理には、表 4.1 に示すように、リセット、トラップ命令、および割込みによるものがあります。これらの例外処理には表 4.1 に示すように優先度が設けられており、複数の例外処理が同時に発生した場合は、この優先度に従って受け付けられ処理されます。トラップ命令例外処理は、プログラム実行状態で常に受け付けられます。

表 4.1 例外処理の種類と優先度

優先度	例外処理の種類	例外処理開始タイミング
高 ↑ 低	リセット	RES 端子が Low レベルから High レベルに変化すると、ただちに開始します。
	割込み	割込み要求が発生すると、命令の実行終了時または例外処理終了時に開始します。
	トラップ命令 (TRAPA)	トラップ (TRAPA) 命令の実行により開始します。

### 4.1.2 例外処理の動作

例外処理は、各例外処理要因により起動されます。

トラップ命令および割込み例外処理は、次のように動作します。

- (1) プログラムカウンタ (PC) とコンディションコードレジスタ (CCR) をスタックに退避します。
- (2) CCR の割込みマスクビットを 1 にセットします。
- (3) 起動要因に対応するベクタアドレスを生成し、そのベクタアドレスの内容が示す番地からプログラムの実行が開始されます。

リセット例外処理の場合は上記 (2)、(3) の動作を行います。

### 4.1.3 例外処理要因とベクタテーブル

各例外処理要因は、図 4.1 に示すように分類されます。

これらの例外処理要因には、それぞれ異なるベクタアドレスが割り当てられています。

これらの要因とベクタアドレスとの対応を表 4.2 に示します。

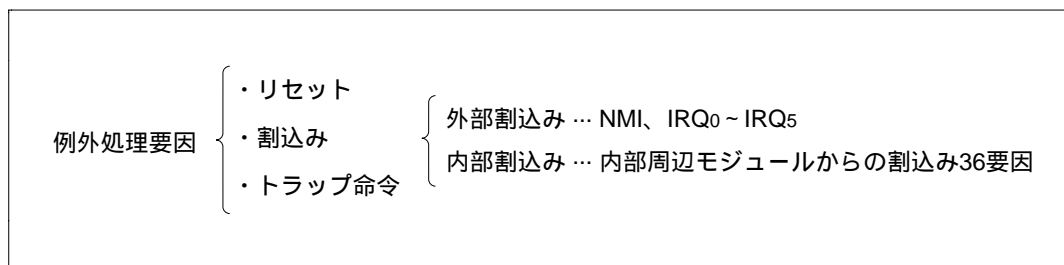


図 4.1 例外処理要因

表 4.2 例外処理ベクタテーブル

			ベクタアドレス* <sup>1</sup>	
			アドバンストモード	ノーマルモード
リセット		0	H'0000 ~ H'0003	H'0000 ~ H'0001
システム予約		1	H'0004 ~ H'0007	H'0002 ~ H'0003
		2	H'0008 ~ H'000B	H'0004 ~ H'0005
		3	H'000C ~ H'000F	H'0006 ~ H'0007
		4	H'0010 ~ H'0013	H'0008 ~ H'0009
		5	H'0014 ~ H'0017	H'000A ~ H'000B
		6	H'0018 ~ H'001B	H'000C ~ H'000D
外部割り込み NMI		7	H'001C ~ H'001F	H'000E ~ H'000F
トラップ命令 (4 要因)		8	H'0020 ~ H'0023	H'0010 ~ H'0011
		9	H'0024 ~ H'0027	H'0012 ~ H'0013
		10	H'0028 ~ H'002B	H'0014 ~ H'0015
		11	H'002C ~ H'002F	H'0016 ~ H'0017
外部割り込み	IRQ <sub>0</sub>	12	H'0030 ~ H'0033	H'0018 ~ H'0019
	IRQ <sub>1</sub>	13	H'0034 ~ H'0037	H'001A ~ H'001B
	IRQ <sub>2</sub>	14	H'0038 ~ H'003B	H'001C ~ H'001D
	IRQ <sub>3</sub>	15	H'003C ~ H'003F	H'001E ~ H'001F
	IRQ <sub>4</sub>	16	H'0040 ~ H'0043	H'0020 ~ H'0021
	IRQ <sub>5</sub>	17	H'0044 ~ H'0047	H'0022 ~ H'0023
システム予約		18	H'0048 ~ H'004B	H'0024 ~ H'0025
		19	H'004C ~ H'004F	H'0026 ~ H'0027
内部割り込み* <sup>2</sup>		20	H'0050 ~ H'0053	H'0028 ~ H'0029
		63	H'00FC ~ H'00FF	H'007E ~ H'007F

【注】 \*1 アドレスの下位 16 ビットを示しています。

\*2 内部割り込みのベクタテーブルは、「5.3.3 割り込み例外処理ベクタテーブル」を参照してください。

## 4.2 リセット

### 4.2.1 概要

リセットは、最も優先順位の高い例外処理です。

$\overline{\text{RES}}$  端子が Low レベルになると、実行中の処理はすべて打ち切れ、本 LSI はリセット状態になります。リセットによって、CPU の内部状態と内蔵周辺モジュールの各レジスタがイニシャライズされます。

$\overline{\text{RES}}$  端子が Low レベルから High レベルになるとリセット例外処理が開始されます。

ウォッチドッグタイマのオーバフローによって、リセット状態とすることもできます。

詳細は、「第 12 章 ウォッチドッグタイマ」を参照してください。

### 4.2.2 リセットシーケンス

$\overline{\text{RES}}$  端子が Low レベルになると本 LSI はリセット状態になります。

本 LSI を確実にリセットするために、電源投入時は最低 20ms の間、 $\overline{\text{RES}}$  端子を Low レベルに保持してください。また、動作中は最低 10 システムクロック ( ) サイクルの間 Low レベルにしてください。フラッシュメモリ版およびフラッシュメモリ R 版使用時は、最低 20 システムクロック以上 Low レベルにしてください。リセット状態の各端子の状態は「付録 D.2 リセット時の端子状態」を参照してください。

$\overline{\text{RES}}$  端子が一定期間 Low レベルの後、High レベルになるとリセット例外処理が開始され、本 LSI は次のように動作します。

- (1) CPU の内部状態と内蔵周辺モジュールの各レジスタがイニシャライズされ、CCR の I ビットが 1 にセットされます。
- (2) リセット例外処理ベクタアドレス ( アドバンスモード時は H'0000 ~ H'0003、ノーマルモード時は H'0000 ~ H'0001 ) をリードして、そのベクタアドレスの内容が示すアドレスからプログラムの実行を開始します。

モード 1、3 のリセットシーケンスを図 4.2 に、モード 2、4 のリセットシーケンスを図 4.3 に、モード 6 のリセットシーケンスを図 4.4 に示します。

#### 4. 例外処理

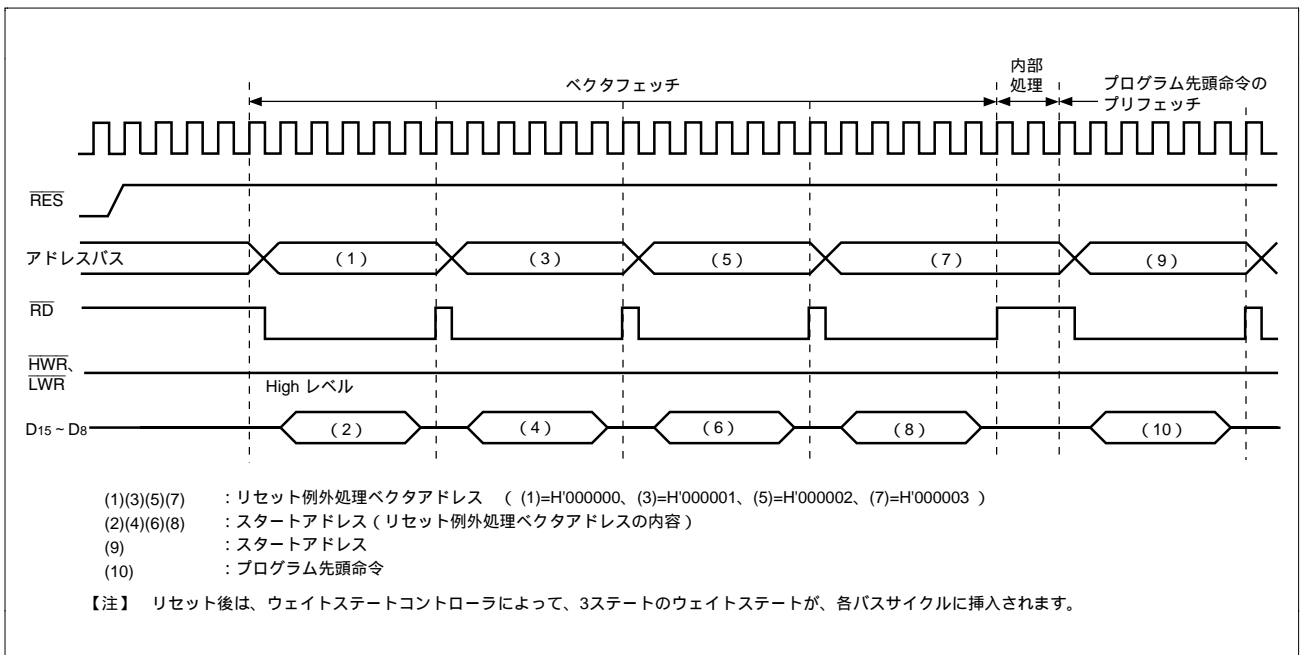


図 4.2 リセットシーケンス (モード 1、3)

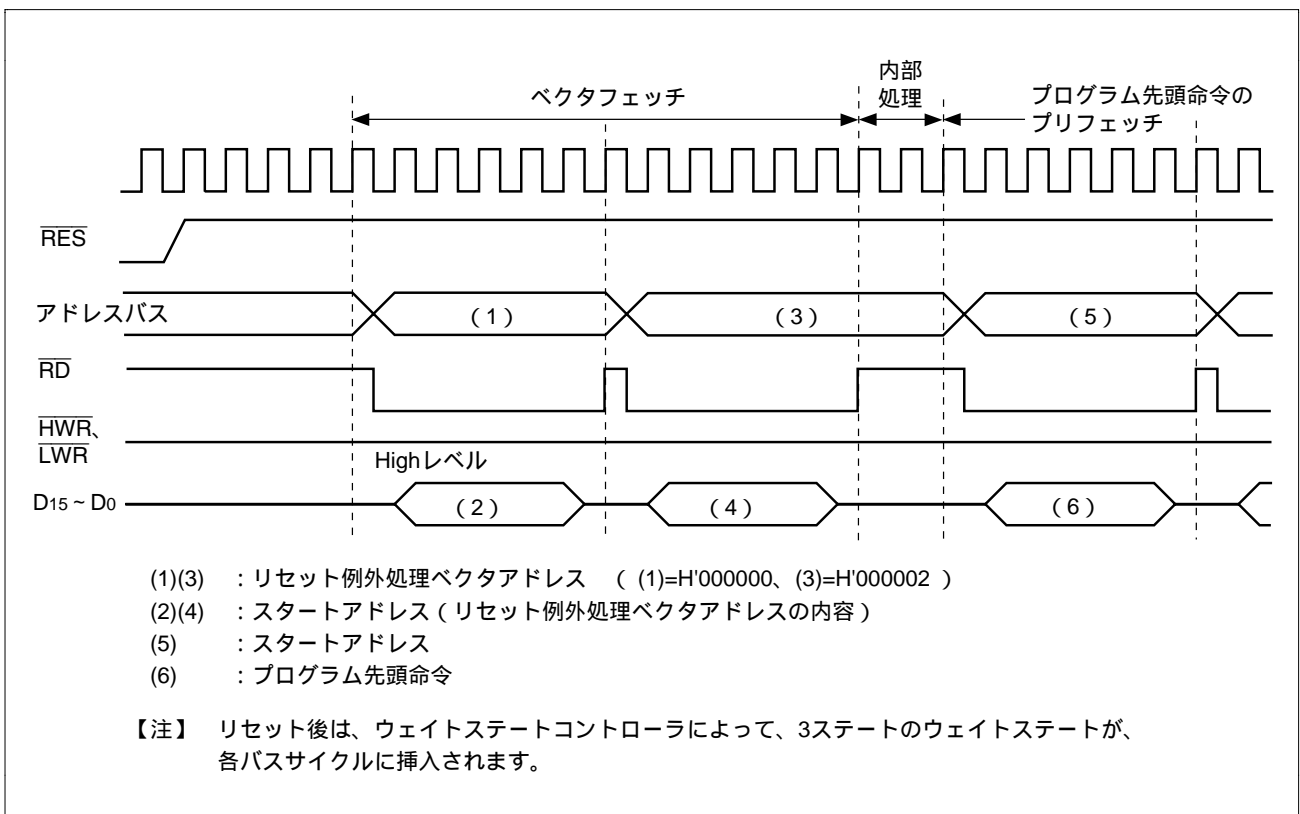


図 4.3 リセットシーケンス (モード 2、4)

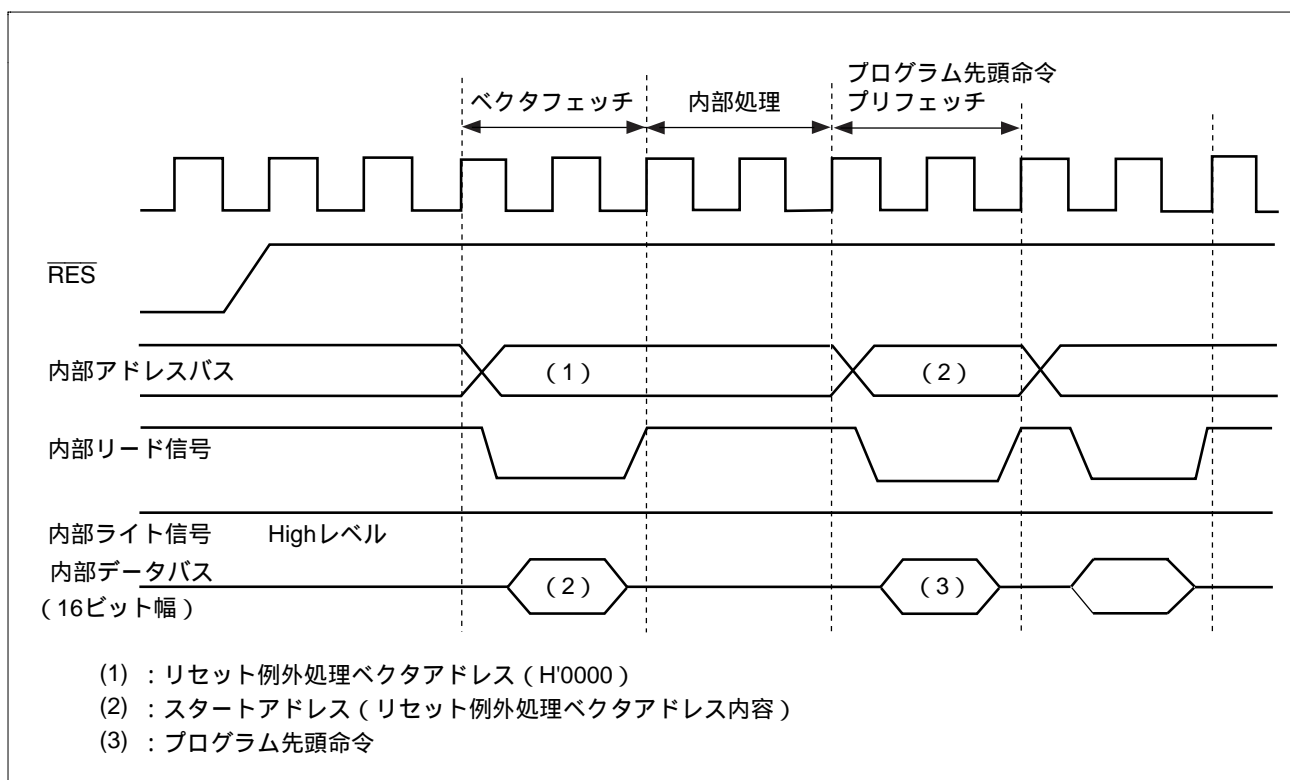


図 4.4 リセットシーケンス (モード6)

### 4.2.3 リセット直後の割込み

リセット直後、スタックポインタ (SP) をイニシャライズする前に割込みを受け付けると、PC と CCR の退避が正常に行われなため、プログラムの暴走につながります。これを防ぐため、リセット例外処理が実行された直後は、NMI を含めたすべての割込み要求が禁止されます。すなわち、リセット直後はプログラムの先頭 1 命令が必ず実行されますので、プログラム先頭命令は SP をイニシャライズする命令としてください (例: `MOV L #xx : 32, SP`)。

## 4.3 割込み

割込み例外処理を開始させる要因には、7つの外部割込み（NMI、IRQ<sub>0</sub>～IRQ<sub>5</sub>）と、内蔵周辺モジュールからの要求による36の内部要因があります。割込み要因と要因数を図4.5に示します。

割込み要因を要求する内蔵周辺モジュールには、ウォッチドッグタイマ（WDT）、DRAMインタフェース、16ビットタイマ、8ビットタイマ、DMAコントローラ（DMAC）、シリアルコミュニケーションインタフェース（SCI）、およびA/D変換器があります。割込みベクタアドレスは、各要因別に割り当てられています。

NMIは最優先の割込みで、常に受け付けられます\*。割込みは、割込みコントローラによって制御されます。割込みコントローラは、NMI以外の割込みを2レベルの優先順位を設定して、多重割込みの制御を行うことができます。割込みの優先順位は、割込みコントローラのインタラプトプライオリティレジスタA、B（IPRA、B）に設定します。

【注】\* フラッシュメモリ版およびフラッシュメモリR版にはNMI入力が禁止される場合があります。詳細は「18.6.4 NMI入力の禁止条件」を参照してください。

割込みについての詳細は「第5章 割込みコントローラ」を参照してください。

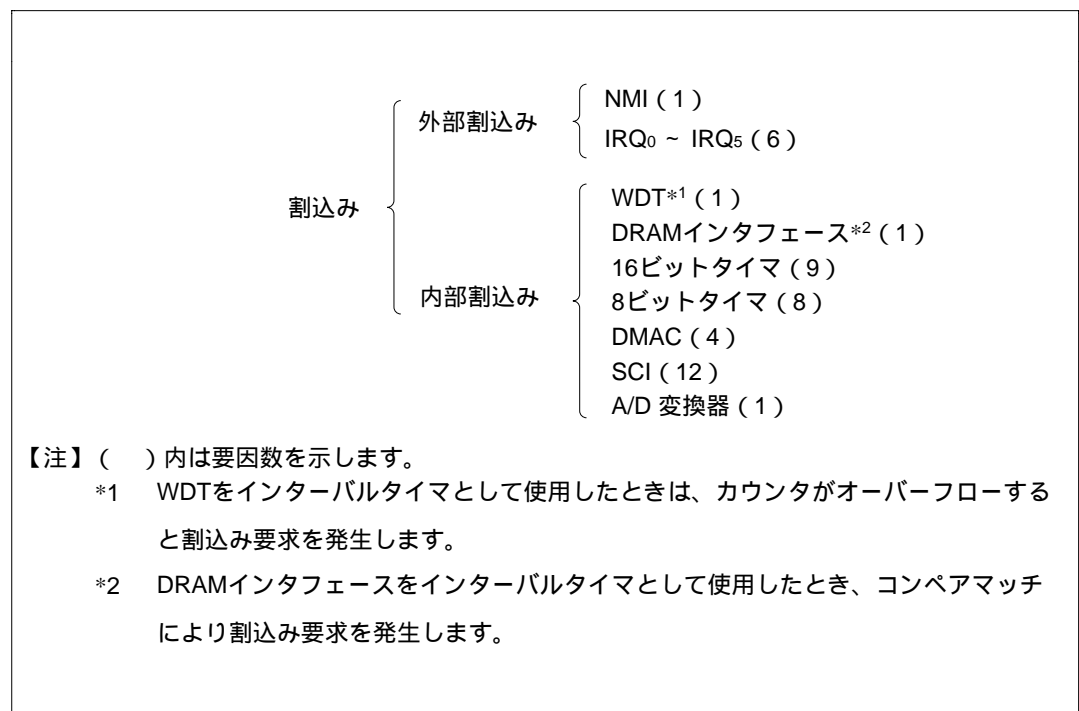


図 4.5 割込み要因と要因数

## 4.4 トラップ命令

トラップ命令例外処理は、TRAPA 命令を実行すると例外処理を開始します。

例外処理によって、SYSCR の UE ビットが 1 のときは CCR の I ビットが 1 にセットされ、UE ビットが 0 のときには CCR の I ビット、UI ビットがそれぞれ 1 にセットされます。

TRAPA 命令は、命令コード中で指定した 0~3 のベクタ番号に対応するベクタテーブルからスタートアドレスを取り出します。



## 4.5 例外処理後のスタックの状態

トラップ命令例外処理および割り込み例外処理後のスタックの状態を図 4.6 に示します。

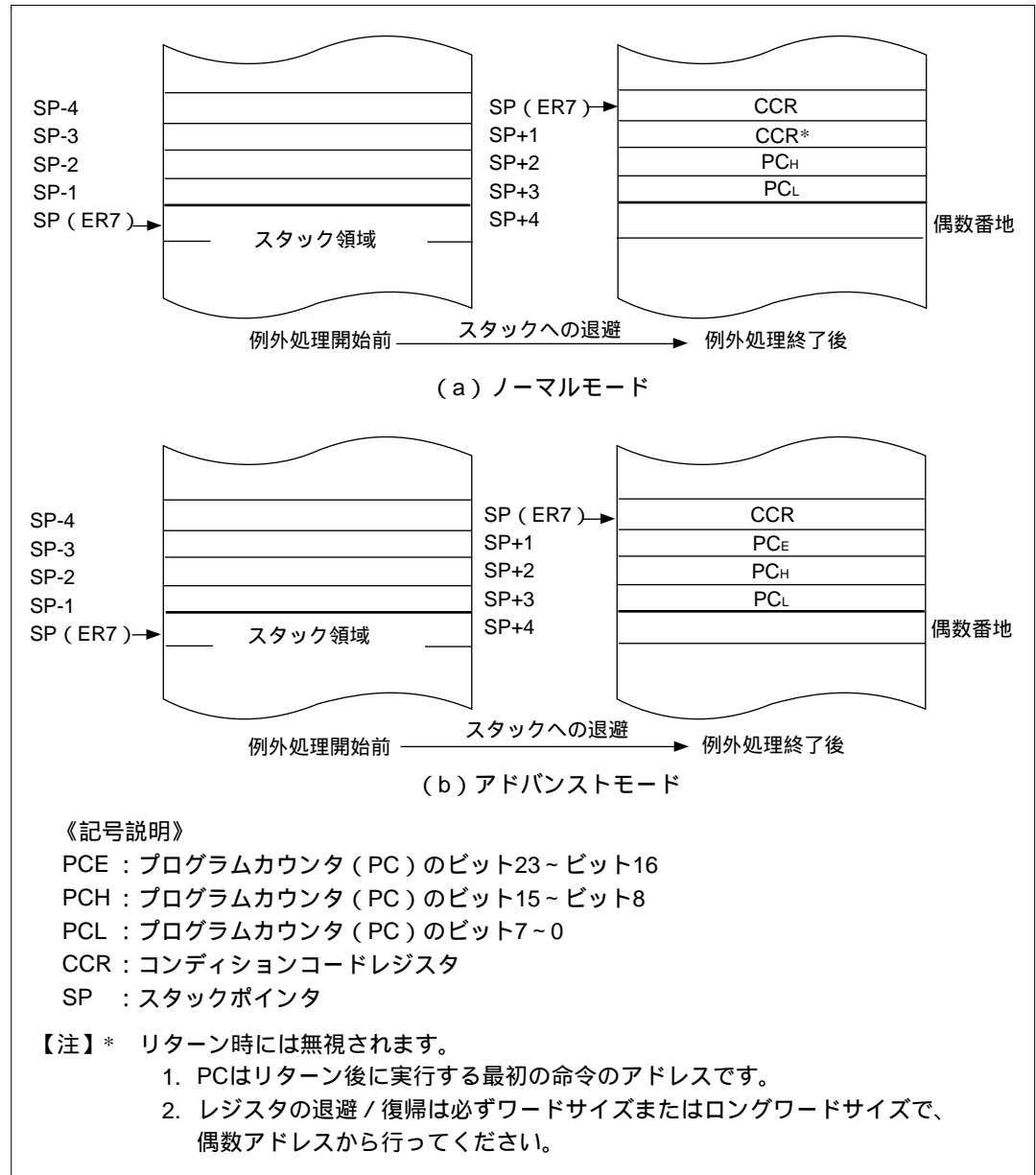


図 4.6 例外処理終了後のスタックの状態

## 4.6 スタック使用上の注意

本LSIでは、ワードデータまたはロングワードデータをアクセスする場合は、アドレスの最下位ビットは0とみなされます。スタック領域に対するアクセスは、常にワードサイズまたはロングワードサイズで行い、スタックポインタ (SP : ER7) の内容は奇数にしないでください。

すなわち、レジスタの退避は、

```
PUSH . W Rn (MOV . W Rn , @ - SP)
```

```
PUSH . L ERn (MOV . L ERn , @ - SP)
```

また、レジスタの復帰は、

```
POP . W Rn (MOV . W @SP+ , Rn)
```

```
POP . L ERn (MOV . L @SP+ , ERn)
```

を使用してください。

SPを奇数に設定すると、誤動作の原因となります。SPを奇数に設定した場合の動作例を図4.7に示します。

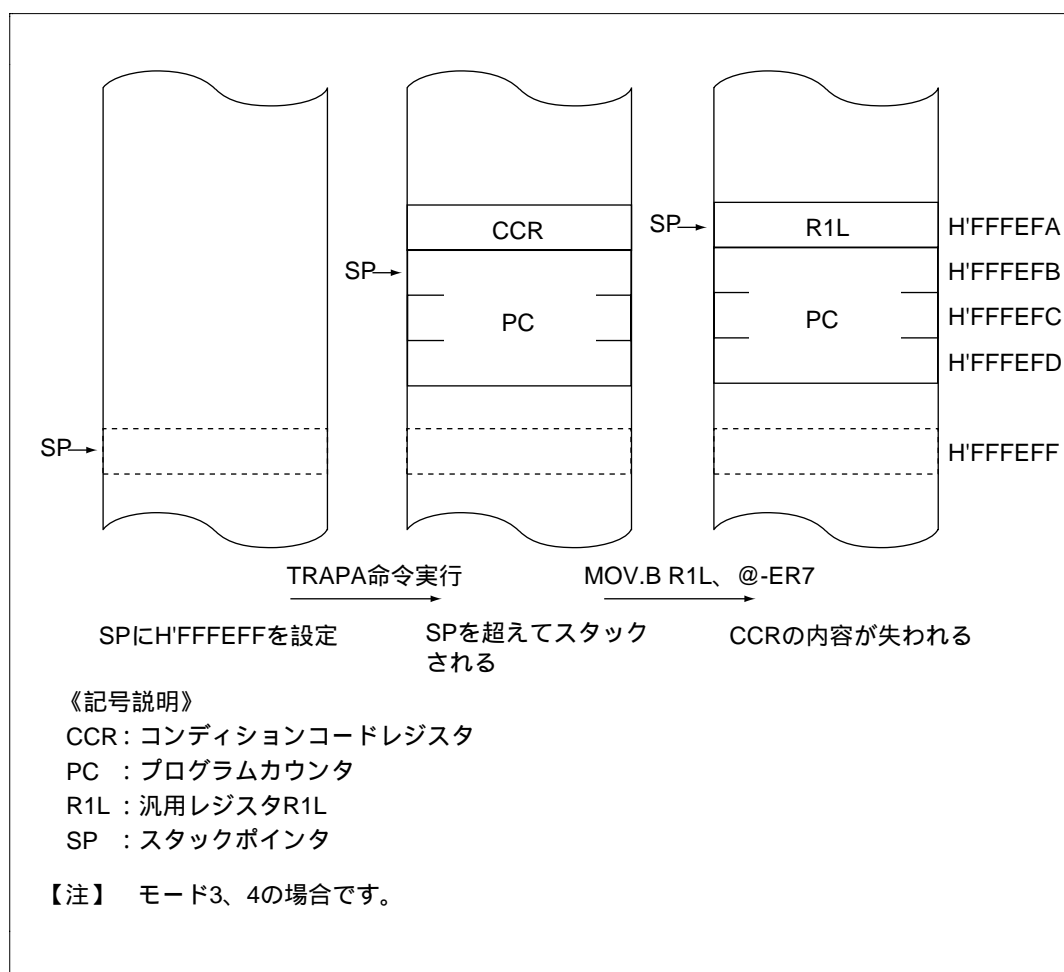


図 4.7 SP を奇数に設定したときの動作



---

# 5. 割り込みコントローラ

---

## 第5章 目次

5.1	概要	99
5.1.1	特長	99
5.1.2	ブロック図	100
5.1.3	端子構成	101
5.1.4	レジスタ構成	101
5.2	各レジスタの説明	102
5.2.1	システムコントロールレジスタ (SYSCR)	102
5.2.2	インタラプトプライオリティレジスタ A、B (IPRA、IPRB)	103
5.2.3	IRQ ステータスレジスタ (ISR)	110
5.2.4	IRQ イネーブルレジスタ (IER)	111
5.2.5	IRQ センスコントロールレジスタ (ISCR)	112
5.3	割り込み要因	113
5.3.1	外部割り込み	113
5.3.2	内部割り込み	114
5.3.3	割り込み例外処理ベクタテーブル	114
5.4	割り込み動作	117
5.4.1	割り込み動作の流れ	117
5.4.2	割り込み例外処理シーケンス	122
5.4.3	割り込み応答時間	123
5.5	使用上の注意	124
5.5.1	割り込みの発生とディスエーブルとの競合	124
5.5.2	割り込みの受け付けを禁止している命令	125
5.5.3	EEPMOV 命令実行中の割り込み	125



## 5.1 概要

### 5.1.1 特長

割り込みコントローラには、次の特長があります。

IPR により、優先順位を設定可能

割り込み優先順位を設定するインタラプトプライオリティレジスタ A、B (IPRA、B) を備えており、NMI 以外の割り込みを要因ごとまたはモジュールごとに 2 レベルの優先順位を設定できます。

CPU のコンディションコードレジスタ (CCR) の I、UI ビットにより、3 レベルの許可 / 禁止状態を設定可能。

7 本の外部割り込み端子

NMI は最優先の割り込みで常に受け付けられます\*。NMI は立上がりエッジ / 立下がりエッジを選択できます。また IRQ<sub>0</sub> ~ IRQ<sub>5</sub> は立下がりエッジ / レベルセンスを独立に選択できます。

【注】 \* フラッシュメモリ版およびフラッシュメモリ R 版には NMI 入力禁止される場合があります。詳細は「18.6.4 NMI 入力の禁止条件」を参照してください。

### 5.1.2 ブロック図

割り込みコントローラのブロック図を図 5.1 に示します。

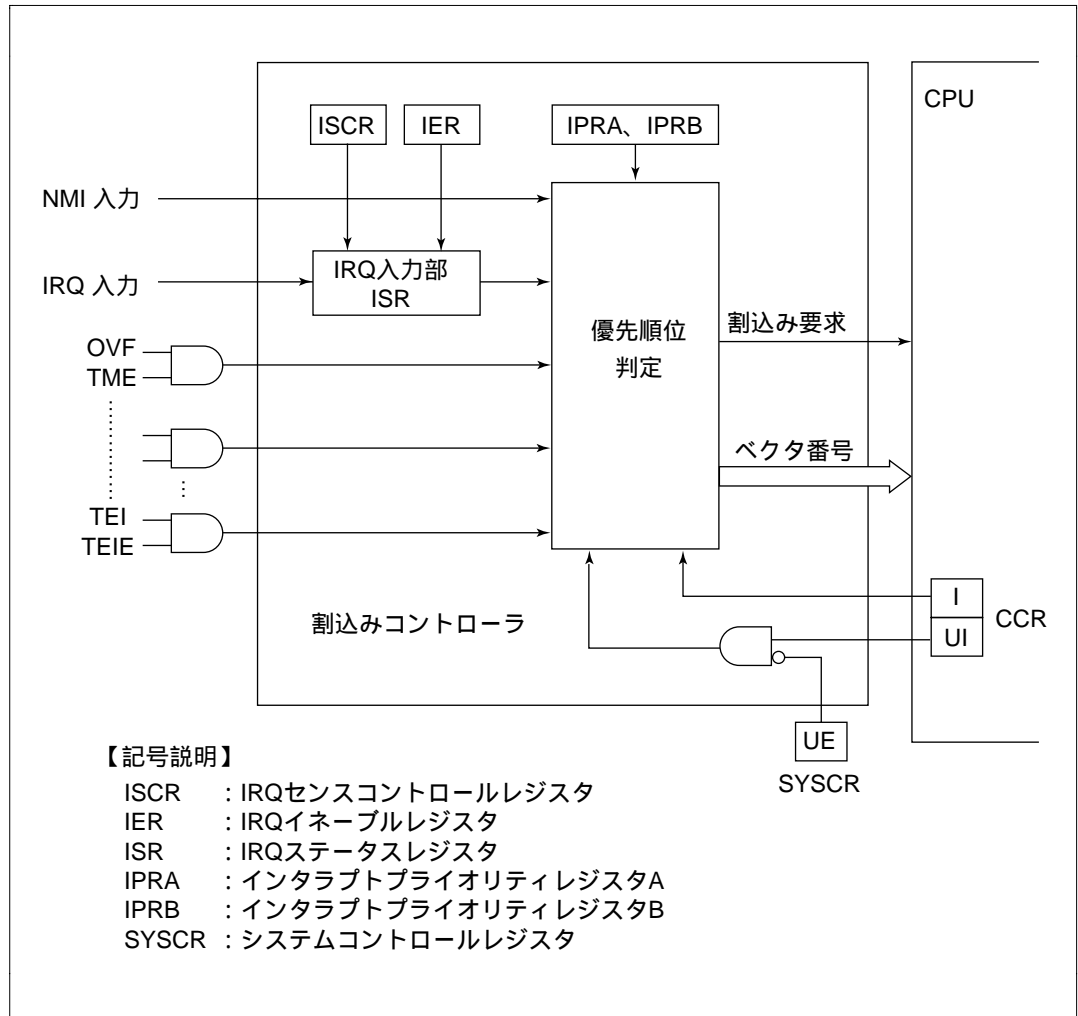


図 5.1 割り込みコントローラのブロック図

### 5.1.3 端子構成

割込みコントローラの端子構成を表 5.1 に示します。

表 5.1 端子構成

名称	略称	入出力	機能
ノンマスクابل割込み	NMI	入力	マスク不可能な外部割込み*、立上がりエッジ / 立下がりエッジ選択可能
外部割込み要求 5~0	$\overline{IRQ}_5 \sim \overline{IRQ}_0$	入力	マスク可能な外部割込み、立下がりエッジ / レベルセンス選択可能

【注】 \* NMI 入力が禁止される場合があります。詳細は「18.6.4 NMI 入力の禁止条件」を参照してください。

### 5.1.4 レジスタ構成

割込みコントローラのレジスタ構成を表 5.2 に示します。

表 5.2 レジスタ構成

アドレス* <sup>1</sup>	名称	略称	R/W	初期値
H'EE012	システムコントロールレジスタ	SYSCR	R/W	H'09
H'EE014	IRQ センスコントロールレジスタ	ISCR	R/W	H'00
H'EE015	IRQ イネーブルレジスタ	IER	R/W	H'00
H'EE016	IRQ ステータスレジスタ	ISR	R/(W) * <sup>2</sup>	H'00
H'EE018	インタラプトプライオリティレジスタ A	IPRA	R/W	H'00
H'EE019	インタラプトプライオリティレジスタ B	IPRB	R/W	H'00

【注】 \*<sup>1</sup> アドバンスモード時のアドレス下位 20 ビットを示しています。

\*<sup>2</sup> フラグをクリアするための 0 ライトのみ可能です。



## 5.2 各レジスタの説明

### 5.2.1 システムコントロールレジスタ (SYSCR)

SYSCR は 8 ビットのリード/ライト可能なレジスタで、ソフトウェアスタンバイモードの制御、CCR の UI ビットの動作の選択、NMI の検出エッジの選択、および内蔵 RAM 有効/無効の選択を行います。

ここでは、ビット 3、2 についてのみ説明します。なお、その他のビットの詳細については「3.3 システムコントロールレジスタ (SYSCR)」を参照してください。

SYSCR はリセット、またはハードウェアスタンバイモード時に H'09 にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

ビット:	7	6	5	4	3	2	1	0
	SSBY	STS2	STS1	STS0	UE	NMIEG	SSOE	RAME
初期値:	0	0	0	0	1	0	0	1
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

RAMイネーブル

ソフトウェアスタンバイ  
出力ポートイネーブル

NMIエッジセレクト  
NMI端子の入力エッジを  
選択するビットです。

ユーザビットイネーブル  
CCRのUIビットをユーザビットとして  
使用するか、割込みマスクビットとし  
て使用するかを選択するビットです。

スタンバイタイムセレクト2~0

ソフトウェアスタンバイ

## ビット3：ユーザビットイネーブル（UE）

CCRのUIビットをユーザビットとして使用するか、割込みマスクビットとして使用するかを選択します。

ビット3	説明
UE	
0	CCRのUIビットを割込みマスクビットとして使用
1	CCRのUIビットをユーザビットとして使用 (初期値)

## ビット2：NMIエッジセレクト（NMIEG）

NMI端子の入力エッジ選択を行います。

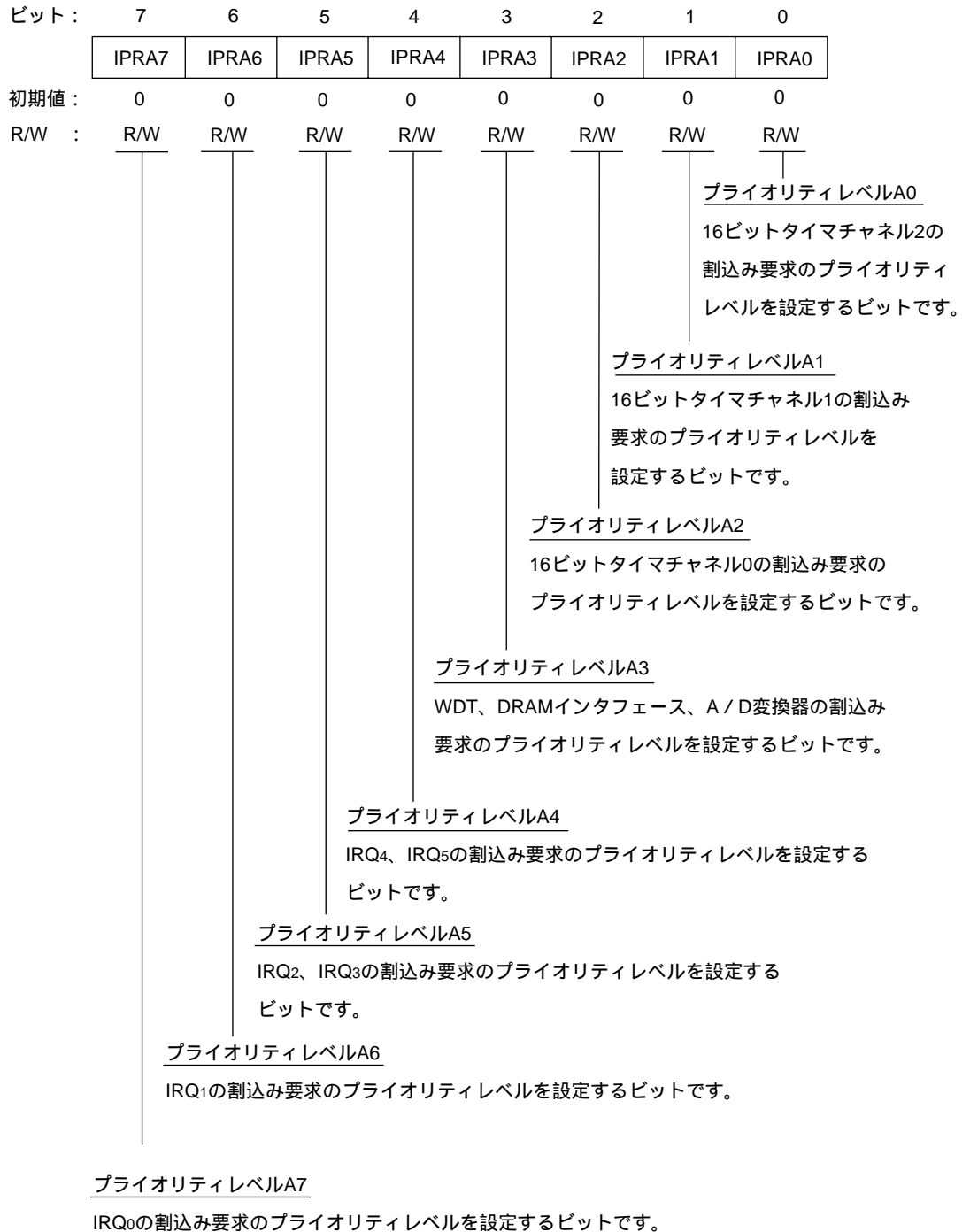
ビット2	説明
NMIEG	
0	NMI入力の立下がりエッジで割込み要求を発生 (初期値)
1	NMI入力の立上がりエッジで割込み要求を発生

## 5.2.2 インタラプトプライオリティレジスタ A、B（IPRA、IPRB）

IPRA、IPRBは各々8ビットのリード/ライト可能なレジスタで割込みの優先順位を制御します。

(1) インタラプトプライオリティレジスタ A (IPRA)

IPRA は 8 ビットのリード/ライト可能なレジスタで、プライオリティレベルを設定できます。



IPRA はリセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。

## ビット7：プライオリティレベルA7 (IPRA7)

IRQ<sub>0</sub>の割り込み要求のプライオリティレベルを設定します。

ビット7	説明	
IPRA7		
0	IRQ <sub>0</sub> の割り込み要求はプライオリティレベル0 (非優先)	(初期値)
1	IRQ <sub>0</sub> の割り込み要求はプライオリティレベル1 (優先)	

## ビット6：プライオリティレベルA6 (IPRA6)

IRQ<sub>1</sub>の割り込み要求のプライオリティレベルを設定します。

ビット6	説明	
IPRA6		
0	IRQ <sub>1</sub> の割り込み要求はプライオリティレベル0 (非優先)	(初期値)
1	IRQ <sub>1</sub> の割り込み要求はプライオリティレベル1 (優先)	

## ビット5：プライオリティレベルA5 (IPRA5)

IRQ<sub>2</sub>、IRQ<sub>3</sub>の割り込み要求のプライオリティレベルを設定します。

ビット5	説明	
IPRA5		
0	IRQ <sub>2</sub> 、IRQ <sub>3</sub> の割り込み要求はプライオリティレベル0 (非優先)	(初期値)
1	IRQ <sub>2</sub> 、IRQ <sub>3</sub> の割り込み要求はプライオリティレベル1 (優先)	

## ビット4：プライオリティレベルA4 (IPRA4)

IRQ<sub>4</sub>、IRQ<sub>5</sub>の割り込み要求のプライオリティレベルを設定します。

ビット4	説明	
IPRA4		
0	IRQ <sub>4</sub> 、IRQ <sub>5</sub> の割り込み要求はプライオリティレベル0 (非優先)	(初期値)
1	IRQ <sub>4</sub> 、IRQ <sub>5</sub> の割り込み要求はプライオリティレベル1 (優先)	

## ビット3：プライオリティレベルA3 (IPRA3)

WDT、DRAM インタフェース、A/D 変換器の割込み要求のプライオリティレベルを設定します。

ビット3	説明
IPRA3	
0	WDT、DRAM インタフェース、A/D 変換器の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	WDT、DRAM インタフェース、A/D 変換器の割込み要求はプライオリティレベル1 (優先)

## ビット2：プライオリティレベルA2 (IPRA2)

16 ビットタイマチャンネル0 割込み要求のプライオリティレベルを設定します。

ビット2	説明
IPRA2	
0	16 ビットタイマチャンネル0 の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	16 ビットタイマチャンネル0 の割込み要求はプライオリティレベル1 (優先)

## ビット1：プライオリティレベルA1 (IPRA1)

16 ビットタイマチャンネル1 の割込み要求のプライオリティレベルを設定します。

ビット1	説明
IPRA1	
0	16 ビットタイマチャンネル1 の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	16 ビットタイマチャンネル1 の割込み要求はプライオリティレベル1 (優先)

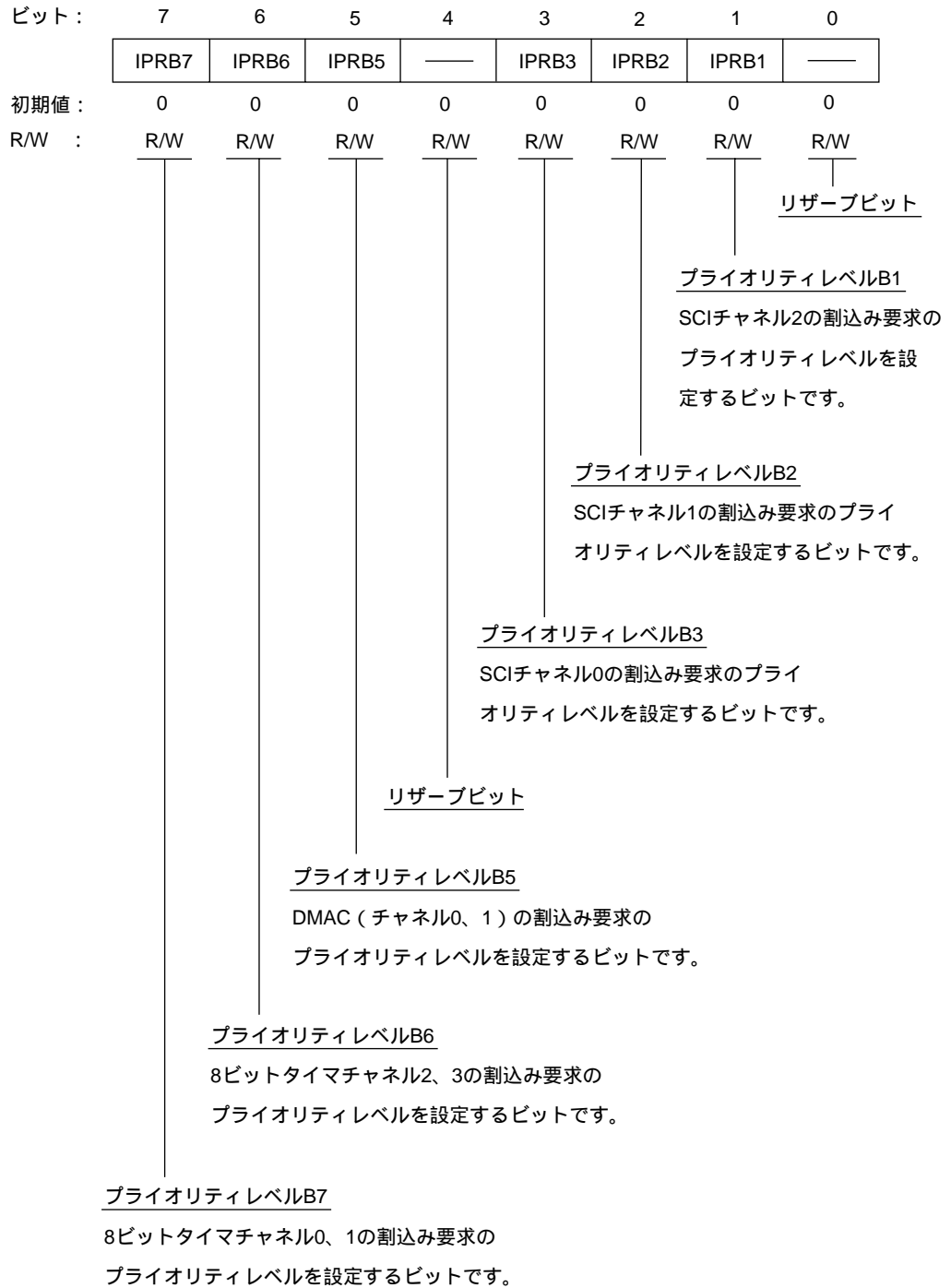
## ビット0：プライオリティレベルA0 (IPRA0)

16 ビットタイマチャンネル2 の割込み要求のプライオリティレベルを設定します。

ビット0	説明
IPRA0	
0	16 ビットタイマチャンネル2 の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	16 ビットタイマチャンネル2 の割込み要求はプライオリティレベル1 (優先)

## (2) インタラプトプライオリティレジスタ B (IPRB)

IPRB は 8 ビットのリード/ライト可能なレジスタで、プライオリティレベルを設定できます。



IPRB はリセット、ハードウェアスタンバイモード時に、H'00 にイニシャライズされま  
す。

ビット7：プライオリティレベルB7 (IPRB7)

8ビットタイマチャンネル0、1の割込み要求のプライオリティレベルを設定します。

ビット7	説明
IPRB7	
0	8ビットタイマチャンネル0、1の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	8ビットタイマチャンネル0、1の割込み要求はプライオリティレベル1 (優先)

ビット6：プライオリティレベルB6 (IPRB6)

8ビットタイマチャンネル2、3の割込み要求のプライオリティレベルを設定します。

ビット6	説明
IPRB6	
0	8ビットタイマチャンネル2、3の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	8ビットタイマチャンネル2、3の割込み要求はプライオリティレベル1 (優先)

ビット5：プライオリティレベルB5 (IPRB5)

DMAC (チャンネル0、1)の割込み要求のプライオリティレベルを設定します。

ビット5	説明
IPRB5	
0	DMAC (チャンネル0、1)の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	DMAC (チャンネル0、1)の割込み要求はプライオリティレベル1 (優先)

ビット4：リザーブビット

リザーブビットです。リード/ライト可能ですが、優先順位には関係ありません。

## ビット3：プライオリティレベルB3 (IPRB3)

SCI チャンネル0の割込み要求のプライオリティレベルを設定します。

ビット3	説明
IPRB3	
0	SCI チャンネル0の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	SCI チャンネル0の割込み要求はプライオリティレベル1 (優先)

## ビット2：プライオリティレベルB2 (IPRB2)

SCI チャンネル1の割込み要求のプライオリティレベルを設定します。

ビット2	説明
IPRB2	
0	SCI チャンネル1の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	SCI チャンネル1の割込み要求はプライオリティレベル1 (優先)

## ビット1：プライオリティレベルB1 (IPRB1)

SCI チャンネル2の割込み要求のプライオリティレベルを設定します。

ビット1	説明
IPRB1	
0	SCI チャンネル2の割込み要求はプライオリティレベル0 (非優先) (初期値)
1	SCI チャンネル2の割込み要求はプライオリティレベル1 (優先)

## ビット0：リザーブビット

リザーブビットです。リード/ライト可能ですが、優先順位には関係ありません。



### 5.2.3 IRQ ステータスレジスタ (ISR)

ISR は8ビットのリード/ライト可能なレジスタで、IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割込み要求のステータスの表示を行います。

ビット:	7	6	5	4	3	2	1	0
	—	—	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
初期値:	0	0	0	0	0	0	0	0
R/W :	—	—	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

リザーブビット
IRQ<sub>5</sub> ~ IRQ<sub>0</sub>フラグ  
IRQ<sub>5</sub> ~ IRQ<sub>0</sub>フラグ割込み要求の  
ステータスを表示するビットです。

【注】 \* フラグをクリアするための0ライトのみ可能です。

ISR はリセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。

ビット7、6：リザーブビット

リザーブビットです。リードすると常に0が読み出されます。ライトは無効です。

ビット5～0：IRQ<sub>5</sub>～IRQ<sub>0</sub>フラグ (IRQ5F～IRQ0F)

IRQ<sub>5</sub>～IRQ<sub>0</sub> 割込み要求のステータスの表示を行います。

ビット5～0	説明
IRQ5F～IRQ0F	
0	【クリア条件】 <span style="float: right;">(初期値)</span> (1) IRQ <sub>n</sub> F = 1 の状態で IRQ <sub>n</sub> F フラグをリードした後、IRQ <sub>n</sub> F フラグに0をライトしたとき (2) IRQ <sub>n</sub> SC = 0、 $\overline{\text{IRQ}}_n$ 入力が High レベルの状態での割込み例外処理を実行したとき (3) IRQ <sub>n</sub> SC = 1 の状態で IRQ <sub>n</sub> 割込み例外処理を実行したとき
1	【セット条件】 (1) IRQ <sub>n</sub> SC = 0 の状態で $\overline{\text{IRQ}}_n$ 入力が Low レベルになったとき (2) IRQ <sub>n</sub> SC = 1 の状態で $\overline{\text{IRQ}}_n$ 入力が立上がりエッジが発生したとき

(n = 5 ~ 0)

## 5.2.4 IRQ イネーブルレジスタ (IER)

IER は 8 ビットのリード/ライト可能なレジスタで、IRQ<sub>5</sub> ~ IRQ<sub>0</sub> 割り込み要求の許可/禁止を制御します。

ビット:	7	6	5	4	3	2	1	0
	—	—	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

リザーブビット
IRQ<sub>5</sub> ~ IRQ<sub>0</sub>イネーブル  
IRQ<sub>5</sub> ~ IRQ<sub>0</sub>割り込みを許可/禁止するかを  
選択するビットです。

IER はリセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。

### ビット 7、6 : リザーブビット

リザーブビットです。リード/ライト可能ですが、割り込み要求の許可/禁止には関係ありません。

### ビット 5 ~ 0 : IRQ<sub>5</sub> ~ IRQ<sub>0</sub> イネーブル (IRQ5E ~ IRQ0E)

IRQ<sub>5</sub> ~ IRQ<sub>0</sub> 割り込みを許可/禁止するかを選択します。

ビット 5 ~ 0	説明
IRQ5E ~ IRQ0E	
0	IRQ <sub>5</sub> ~ IRQ <sub>0</sub> 割り込みを禁止 (初期値)
1	IRQ <sub>5</sub> ~ IRQ <sub>0</sub> 割り込みを許可

### 5.2.5 IRQ センスコントロールレジスタ (ISCR)

ISCR は 8 ビットのリード/ライト可能なレジスタで、 $\overline{\text{IRQ}}_5 \sim \overline{\text{IRQ}}_0$  端子の入力のレベルセンスまたは立下がりエッジを選択します。

ビット:	7	6	5	4	3	2	1	0
	—	—	IRQ5SC	IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

リザーブビット
IRQ5 ~ IRQ0 センスコントロール  
 IRQ5 ~ IRQ0 割込みのレベルセンスまたは  
 立下がりエッジを選択するビットです。

ISCR はリセットまたは、ハードウェアスタンバイモード時に、H'00 にイニシャライズされます。

#### ビット 7、6 : リザーブビット

リザーブビットです。リード/ライト可能ですが、レベルセンスまたは立下がりエッジの選択には関係ありません。

#### ビット 5 ~ 0 : IRQ<sub>5</sub> ~ IRQ<sub>0</sub> センスコントロール (IRQ5SC ~ IRQ0SC)

IRQ<sub>5</sub> ~ IRQ<sub>0</sub> 割込みを  $\overline{\text{IRQ}}_5 \sim \overline{\text{IRQ}}_0$  端子のレベルセンスで要求するか、立下がりエッジで要求するかを選択します。

ビット 5 ~ 0	説明
IRQ5SC ~ IRQ0SC	
0	$\overline{\text{IRQ}}_5 \sim \overline{\text{IRQ}}_0$ 入力の Low レベルで割込み要求を発生 (初期値)
1	$\overline{\text{IRQ}}_5 \sim \overline{\text{IRQ}}_0$ 入力の立下がりエッジで割込み要求を発生

## 5.3 割り込み要因

割り込み要因には、外部割り込み (NMI、IRQ<sub>0</sub> ~ IRQ<sub>5</sub>) と内部割り込み (36 要因) があります。

### 5.3.1 外部割り込み

外部割り込みには、NMI、IRQ<sub>0</sub> ~ IRQ<sub>5</sub> の 7 要因があります。このうち、NMI、IRQ<sub>0</sub> ~ IRQ<sub>2</sub> はソフトウェアスタンバイモードからの復帰に使用できます。

#### (1) NMI 割り込み

NMI は最優先の割り込みで、CCR の I ビット、UI ビットの状態にかかわらず常に受け付けられます\*。NMI 端子の立上がりエッジまたは立下がりエッジのいずれかで割り込みを要求するか、SYSCR の NMIEG ビットで選択できます。

NMI 割り込み例外処理のベクタ番号は 7 です。

【注】 \* NMI 入力が禁止される場合があります。詳細は「18.9 NMI 入力禁止の条件」を参照してください。

#### (2) IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込み

IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込みは  $\overline{\text{IRQ}}_0 \sim \overline{\text{IRQ}}_5$  端子の入力信号により要求されます。IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込みには次の特長があります。

- $\overline{\text{IRQ}}_0 \sim \overline{\text{IRQ}}_5$  端子の Low レベルまたは立下がりエッジのどちらで割り込みを要求するか、ISCR で選択できます。
- IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込み要求を許可するか禁止するかを、IER で選択できます。また、IPRA の IPRA7 ~ IPRA4 ビットにより割り込みプライオリティレベルを設定できます。
- IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込み要求のステータスは、ISR に表示されます。ISR のフラグはソフトウェアで 0 にクリアすることができます。

IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込みのブロック図を図 5.2 に示します。

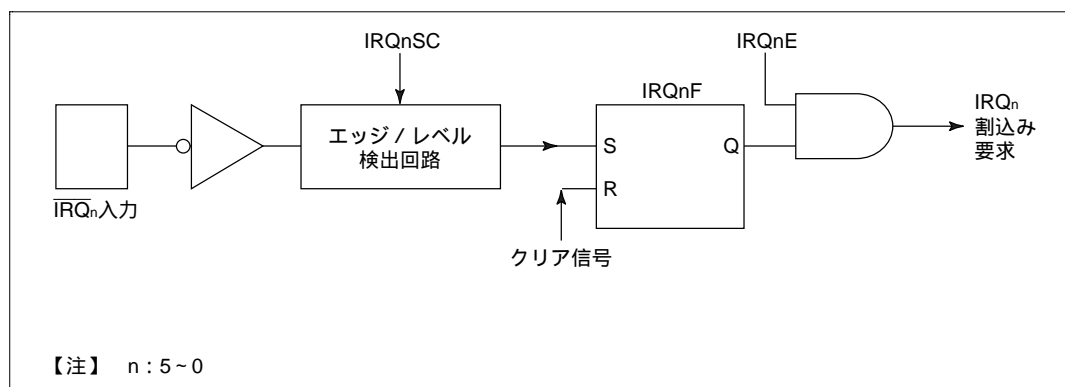


図 5.2 IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込みのブロック図

IRQnF のセットタイミングを図 5.3 に示します。

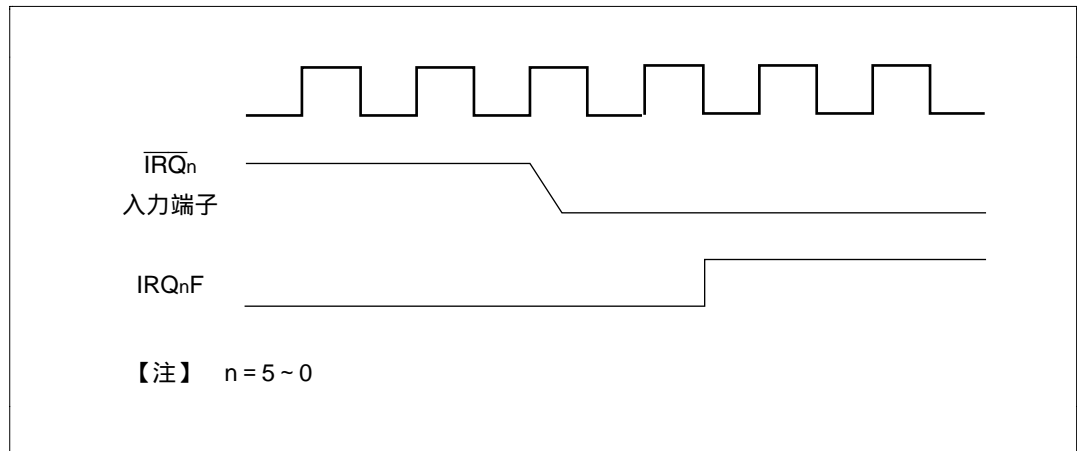


図 5.3 IRQnF セットタイミング

IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割込み例外処理のベクタ番号は 12 ~ 17 です。

IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割込みの検出は、当該の端子が入力に設定されているか、出力に設定されているかに依存しません。したがって、外部割込み入力端子として使用する場合は、対応する DDR を 0 にクリアし、チップセレクト出力端子、リフレッシュ出力端子、SCI の入出力端子、A/D 外部トリガ入力端子としては使用しないでください。

### 5.3.2 内部割込み

内蔵周辺モジュールからの割込みによる内部割込みは 36 要因あります。

- (1) 各内蔵周辺モジュールには割込み要求のステータスを表示するフラグと、これらの割込みを許可するか禁止するかを選択するイネーブルビットがあります。
- (2) IPRA、B によって割込みプライオリティレベルを設定できます。
- (3) 16 ビットタイマ、SCI、A/D 変換器の割込み要求で DMAC の起動ができます。この場合、割込みコントローラに対して割込みは要求されません。このときは I、UI ビットの影響を受けません。

### 5.3.3 割込み例外処理ベクタテーブル

表 5.3 に割込み例外処理要因とベクタアドレスおよび割込み優先順位の一覧を示します。デフォルトの優先順位はベクタ番号の小さいものほど高くなりますが、IPRA、B により NMI 以外の割込みの優先順位を変更することができます。

リセット後の割込み優先順位は表 5.3 に示されるデフォルトの順位となります。

表 5.3 割込み要因とベクタアドレスおよび割込み優先順位一覧 (1)

割込み要因	要因発生元	ベクタ 番号	ベクタアドレス*		IPR	優先 順位
			アドバンスモード	ノーマルモード		
NMI	外部端子	7	H'001C ~ H'001F	H'000E ~ H'000F	——	高 ↑
IRQ0		12	H'0030 ~ H'0033	H'0018 ~ H'0019	IPRA7	
IRQ1		13	H'0034 ~ H'0037	H'001A ~ H'001B	IPRA6	
IRQ2		14	H'0038 ~ H'003B	H'001C ~ H'001D	IPRA5	
IRQ3		15	H'003C ~ H'003F	H'001E ~ H'001F		
IRQ4		16	H'0040 ~ H'0043	H'0020 ~ H'0021	IPRA4	
IRQ5		17	H'0044 ~ H'0047	H'0022 ~ H'0023		
リザーブ	——	18	H'0048 ~ H'004B	H'0024 ~ H'0025	IPRA3	
		19	H'004C ~ H'004F	H'0026 ~ H'0027		
WOVI (インターバルタイマ)	ウォッチ ドッグタイマ	20	H'0050 ~ H'0053	H'0028 ~ H'0029	IPRA3	
CMI (コンペアマッチ)	DRAM インタ フェース	21	H'0054 ~ H'0057	H'002A ~ H'002B	IPRA2	
リザーブ	——	22	H'0058 ~ H'005B	H'002C ~ H'002D		
ADI (A/D エンド)	A/D	23	H'005C ~ H'005F	H'002E ~ H'002F	IPRA2	
IMIA0 (コンペアマッチ/インプットキャプチャ A0)	16 ビット タイマ チャンネル 0	24	H'0060 ~ H'0063	H'0030 ~ H'0031		
IMIB0 (コンペアマッチ/インプットキャプチャ B0)		25	H'0064 ~ H'0067	H'0032 ~ H'0033		
OVI0 (オーバフロー-0)		26	H'0068 ~ H'006B	H'0034 ~ H'0035		
リザーブ	——	27	H'006C ~ H'006F	H'0036 ~ H'0037		
IMIA1 (コンペアマッチ/インプットキャプチャ A1)	16 ビット タイマ チャンネル 1	28	H'0070 ~ H'0073	H'0038 ~ H'0039	IPRA1	
IMIB1 (コンペアマッチ/インプットキャプチャ B1)		29	H'0074 ~ H'0077	H'003A ~ H'003B		
OVI1 (オーバフロー-1)		30	H'0078 ~ H'007B	H'003C ~ H'003D		
リザーブ	——	31	H'007C ~ H'007F	H'003E ~ H'003F		
IMIA2 (コンペアマッチ/インプットキャプチャ A2)	16 ビット タイマ チャンネル 2	32	H'0080 ~ H'0083	H'0040 ~ H'0041	IPRA0	
IMIB2 (コンペアマッチ/インプットキャプチャ B2)		33	H'0084 ~ H'0087	H'0042 ~ H'0043		
OVI2 (オーバフロー-2)		34	H'0088 ~ H'008B	H'0044 ~ H'0045		
リザーブ	——	35	H'008C ~ H'008F	H'0046 ~ H'0047		
CMIA0 (コンペアマッチ A0)	8 ビット タイマ チャンネル 0/1	36	H'0090 ~ H'0093	H'0048 ~ H'0049	IPRB7	
CMIB0 (コンペアマッチ B0)		37	H'0094 ~ H'0097	H'004A ~ H'004B		
CMIA1/CMIB1 (コンペアマッチ A1/B1)		38	H'0098 ~ H'009B	H'004C ~ H'004D		
TOVI0/TOVI1 (オーバフロー-0/1)		39	H'009C ~ H'009F	H'004E ~ H'004F		
CMIA2 (コンペアマッチ A2)	8 ビット タイマ チャンネル 2/3	40	H'00A0 ~ H'00A3	H'0050 ~ H'0051	IPRB6	
CMIB2 (コンペアマッチ B2)		41	H'00A4 ~ H'00A7	H'0052 ~ H'0053		
CMIA3/CMIB3 (コンペアマッチ A3/B3)		42	H'00A8 ~ H'00AB	H'0054 ~ H'0055		
TOVI2/TOVI3 (オーバフロー-2/3)		43	H'00AC ~ H'00AF	H'0056 ~ H'0057		

【注】 \* アドレスの下位 16 ビットを示しています。

5. 割り込みコントローラ

表 5.3 割り込み要因とベクタアドレスおよび割り込み優先順位一覧(2)

割り込み要因	要因発生元	ベクタ 番号	ベクタアドレス*		IPR	優先 順位
			アドバンスモード	ノーマルモード		
DEND0A	DMAC	44	H'00B0 ~ H'00B3	H'0058 ~ H'0059	IPRB5	↑
DEND0B		45	H'00B4 ~ H'00B7	H'005A ~ H'005B		
DEND1A		46	H'00B8 ~ H'00BB	H'005C ~ H'005D		
DEND1B		47	H'00BC ~ H'00BF	H'005E ~ H'005F		
リザーブ	—	48	H'00C0 ~ H'00C3	H'0060 ~ H'0061	—	↑
		49	H'00C4 ~ H'00C7	H'0062 ~ H'0063		
		50	H'00C8 ~ H'00CB	H'0064 ~ H'0065		
		51	H'00CC ~ H'00CF	H'0066 ~ H'0067		
ERI0 (受信エラー0)	SCI チャンネル0	52	H'00D0 ~ H'00D3	H'0068 ~ H'0069	IPRB3	↑
RX10 (受信完了0)		53	H'00D4 ~ H'00D7	H'006A ~ H'006B		
TX10 (送信データエンプティ0)		54	H'00D8 ~ H'00DB	H'006C ~ H'006D		
TEI0 (送信終了0)		55	H'00DC ~ H'00DF	H'006E ~ H'006F		
ERI1 (受信エラー1)	SCI チャンネル1	56	H'00E0 ~ H'00E3	H'0070 ~ H'0071	IPRB2	↑
RX11 (受信完了1)		57	H'00E4 ~ H'00E7	H'0072 ~ H'0073		
TX11 (送信データエンプティ1)		58	H'00E8 ~ H'00EB	H'0074 ~ H'0075		
TEI1 (送信終了1)		59	H'00EC ~ H'00EF	H'0076 ~ H'0077		
ERI2 (受信エラー2)	SCI チャンネル2	60	H'00F0 ~ H'00F3	H'0078 ~ H'0079	IPRB1	↑
RX12 (受信完了2)		61	H'00F4 ~ H'00F7	H'007A ~ H'007B		
TX12 (送信データエンプティ2)		62	H'00F8 ~ H'00FB	H'007C ~ H'007D		
TEI2 (送信終了2)		63	H'00FC ~ H'00FF	H'007E ~ H'007F		

【注】 \* アドレスの下位 16 ビットを示しています。

## 5.4 割り込み動作

### 5.4.1 割り込み動作の流れ

本 LSI では、割り込みの動作は UE ビットの状態によって異なります。UE=1 のときは I ビットで割り込みの制御が行われます。UE=0 のときは、I、UI ビットの組み合わせで割り込みの制御が行われます。表 5.4 に UE、I、UI ビットの各組み合わせのときの割り込みの状態を示します。

NMI 割り込みはリセット状態ハードウェアスタンバイ状態を除き常に受け付けられます\*。IRQ 割り込みおよび内蔵周辺モジュールの割り込みは、それぞれの割り込みに対応したイネーブルビットがあります。このイネーブルビットを 0 にクリアすると、その割り込み要求は無視されます。

【注】 \* NMI 入力が禁止される場合があります。詳細は「18.9 NMI 入力の禁止条件」を参照してください。

表 5.4 UE、I、UI ビットの組み合わせによる割り込みの状態

SYSCR	CCR		状態
	I	UI	
1	0	-	すべての割り込みを受け付けます。プライオリティレベル 1 の割り込み要因の優先順位が高くなります。
	1	-	NMI 以外の割り込みを受け付けません。
0	0	-	すべての割り込みを受け付けます。プライオリティレベル 1 の割り込み要因の優先順位が高くなります。
		0	NMI およびプライオリティレベル 1 の割り込み要因のみを受け付けます。
	1	1	NMI 以外の割り込みを受け付けません。

#### (1) UE ビット = 1 の場合

IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割り込みおよび内蔵周辺モジュールの割り込みは CPU の CCR の I ビットにより一括して、許可 / 禁止を設定できます。I ビットが 0 にクリアされているときは許可状態、1 にセットされているときは禁止状態です。プライオリティレベル 1 の割り込み要因の優先順位は高くなります。

この場合の割り込み受け付けの動作フローチャートを図 5.4 に示します。



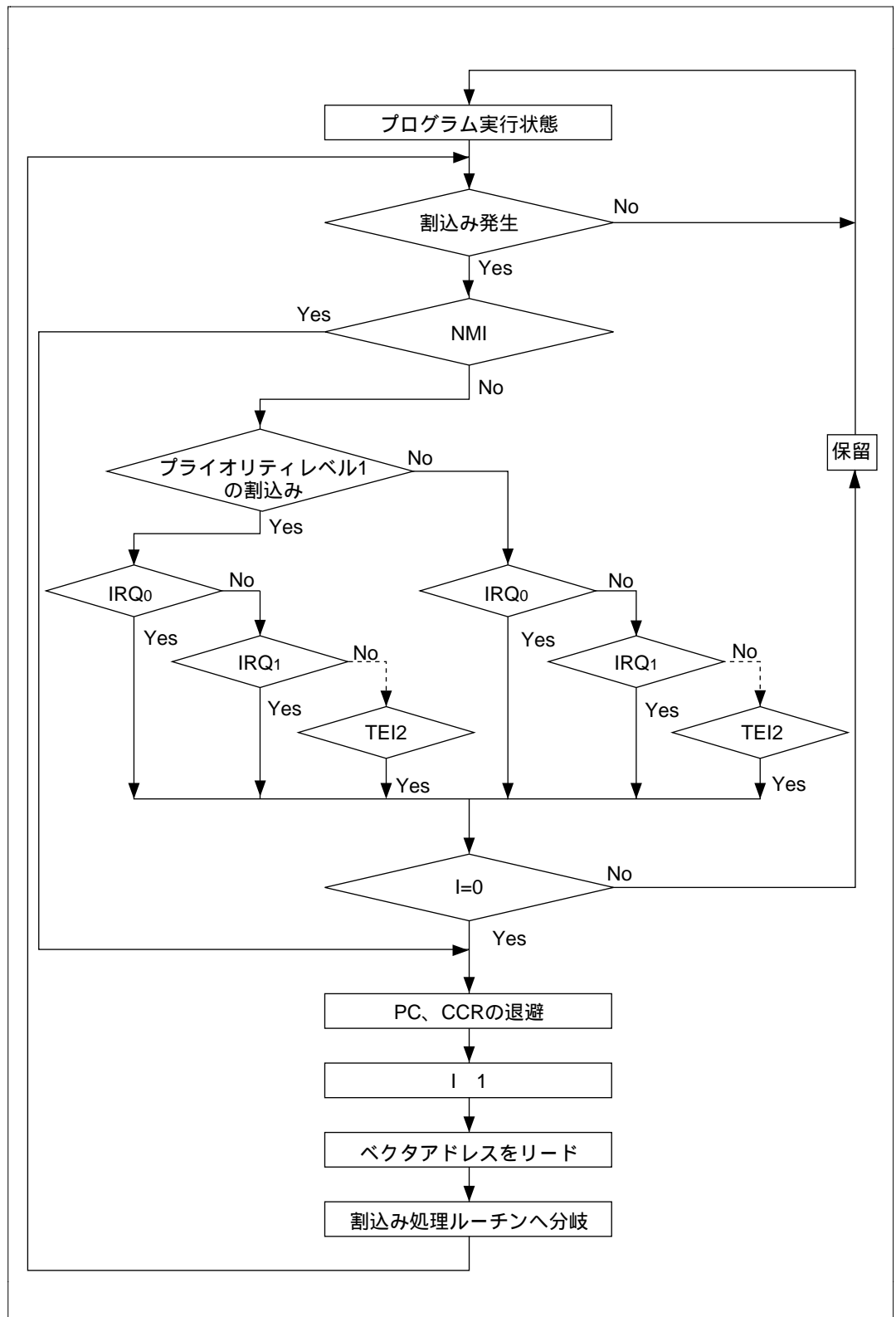


図 5.4 UE = 1 の場合の割り込み受けまでのフロー

- [1] 対応する割込みイネーブルビットが1にセットされている状態で割込み要因が発生したとき、割込みコントローラに対して、割込み要求が送られます。
- [2] 割込みコントローラに対して割込み要求が送られると、IPR に設定された割込み優先順位に従って優先順位が最も高い割込み要求が選択され、その他は保留となります。IPR の設定が同一の割込み要求が同時に発生したときは、表 5.3 に示す優先順位に従って、優先順位の最も高い割込み要求が選択されます。
- [3] Iビットを参照します。Iビットが0にクリアされているときは、割込み要求が受け付けられます。Iビットが1にセットされているときは、NMI 割込みのみ受け付けられ、その他の割込み要求は保留されます。
- [4] 割込み要求が受け付けられると、そのとき実行中の命令の処理が終了した後、割込み例外処理を起動します。
- [5] 割込み例外処理によって、PC と CCR がスタック領域に退避されます。退避される PC は、リターン後に実行する最初の命令のアドレスを示しています。
- [6] 次に CCR の Iビットが1にセットされます。これにより、NMIを除く割込みはマスクされます。
- [7] 受け付けた割込み要求に対応するベクタアドレスを生成し、そのベクタアドレスの内容によって示されるアドレスから、割込み処理ルーチンの実行が開始されます。

## (2) UE ビット = 0 の場合

IRQ<sub>0</sub> ~ IRQ<sub>5</sub> 割込みおよび内蔵周辺モジュールの割込みは CPU の CCR の I、UI ビット、IPR によって 3 レベルの許可 / 禁止状態を実現できます。

- (a) プライオリティレベル 0 の割込み要求は、Iビットが0にクリアされているとき許可状態、1にセットされているとき禁止状態となります。
- (b) プライオリティレベル 1 の割込み要求は、Iビットまたは UI ビットが0にクリアされているとき許可状態、Iビットおよび UI ビットがいずれも 1 にセットされているとき禁止状態となります。

例えば、各割込み要求の対応する割込みイネーブルビットを 1 にセット、IPRA、IPRB をそれぞれ H'20、H'00 に設定した場合 (IRQ<sub>2</sub>、IRQ<sub>3</sub> 割込み要求の優先順位を他の割込みより高くした場合)、次のようになります。

- (a) I = 0 のとき、すべての割込みを許可  
(優先順位 : NMI > IRQ<sub>2</sub> > IRQ<sub>3</sub> > IRQ<sub>0</sub>...)
- (b) I = 1、UI = 0 のとき、NMI、IRQ<sub>2</sub>、IRQ<sub>3</sub> 割込みのみを許可
- (c) I = 1、UI = 1 のとき、NMI 以外の割込みを禁止

また、このときの状態遷移を図 5.5 に示します。

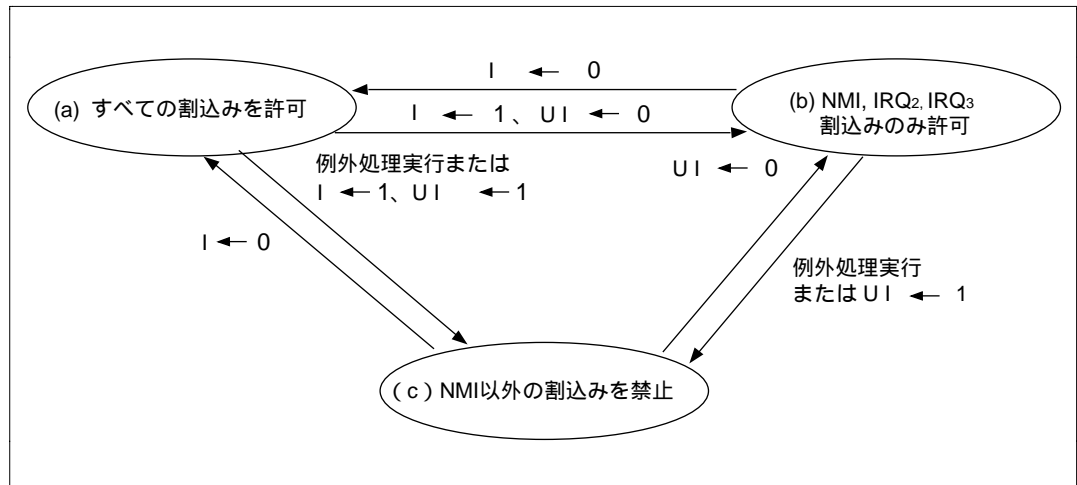


図 5.5 割り込み許可 / 禁止状態の遷移例

UE ビット = 0 のときの割り込み受け付けの動作フローチャートを図 5.6 に示します。

- [ 1 ] 対応する割り込みイネーブルビットが 1 にセットされている状態で割り込み要因が発生したとき、割り込みコントローラに対して割り込み要求が送られます。
- [ 2 ] 割り込みコントローラに対して割り込み要求が送られると、IPR に設定された割り込み優先順位に従って優先順位が最も高い割り込み要求が選択され、その他は保留となります。このとき、IPR の設定が同一の割り込み要求が同時に発生したときは、表 5.3 に示す優先度に従って、優先順位の最も高い割り込み要求が選択されます。
- [ 3 ] I ビットを参照します。I ビットが 0 にクリアされているときは、IPR に関係なく割り込み要求が受け付けられます。このときは UI ビットの影響を受けません。I ビットが 1 にセットされ、UI ビットが 0 にクリアされているときは、プライオリティレベル 1 の割り込み要求のみが受け付けられ、プライオリティレベル 0 の割り込み要求は保留となります。I、UI ビットがいずれも 1 にセットされているときは、割り込み要求は保留となります。
- [ 4 ] 割り込み要求が受け付けられると、そのとき実行中の命令の処理が終了した後、割り込み例外処理を起動します。
- [ 5 ] 割り込み例外処理によって PC と CCR がスタック領域に退避されます。退避される PC は、リターン後に実行する最初の命令のアドレスを示しています。
- [ 6 ] CCR の I、UI ビットが 1 にセットされます。これにより、NMI を除く割り込みはマスクされます。
- [ 7 ] 受け付けた割り込み要求に対応するベクタアドレスを生成し、そのベクタアドレスの内容によって示されるアドレスから、割り込み処理ルーチンの実行が開始されます。

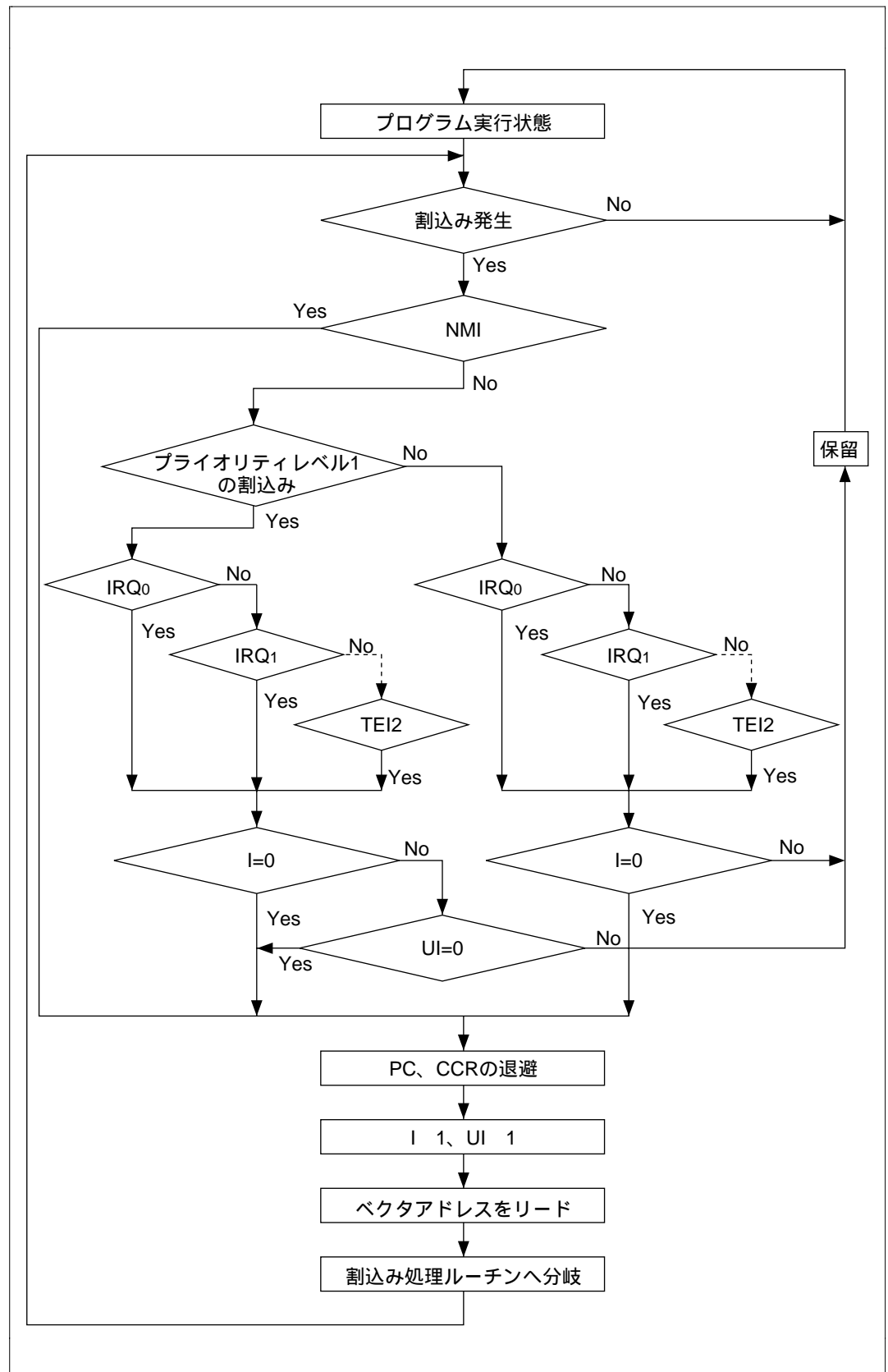


図 5.6 UE = 0 の場合の割り込み受けまでのフロー

### 5.4.2 割り込み例外処理シーケンス

モード2で、プログラム領域とスタック領域を外部メモリ16ビット2ステートアクセス空間にとった場合の割り込みシーケンスを図5.7に示します。

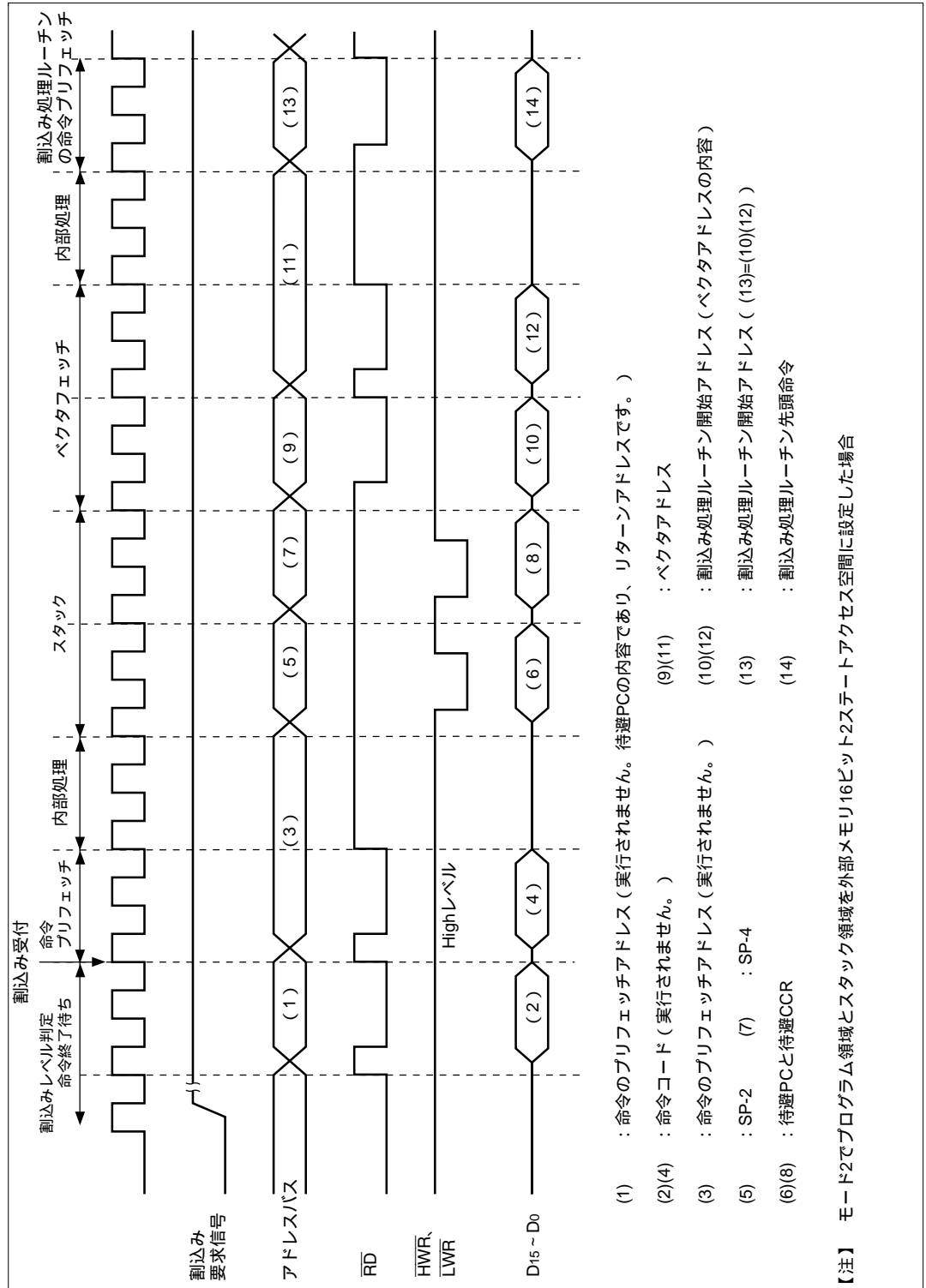


図 5.7 割り込み例外処理シーケンス

### 5.4.3 割り込み応答時間

割り込み要求発生後、割り込み処理ルーチンの先頭命令を実行するまでの、割り込み応答時間を表 5.5 に示します。

表 5.5 割り込み応答時間

No.	項目	内蔵メモリ	外部メモリ			
			8ビットバス		16ビットバス	
			2状態	3状態	2状態	3状態
1	割り込み優先順位判定		2 <sup>*1</sup>			
2	実行中の命令が終了するまでの最大待ちステート数	1~23	1~27	1~31 <sup>*4</sup>	1~23	1~25 <sup>*4</sup>
3	PC、CCRのスタック	4	8	12 <sup>*4</sup>	4	6 <sup>*4</sup>
4	ベクタフェッチ	4	8	12 <sup>*4</sup>	4	6 <sup>*4</sup>
5	命令フェッチ <sup>*2</sup>	4	8	12 <sup>*4</sup>	4	6 <sup>*4</sup>
6	内部処理 <sup>*3</sup>	4	4	4	4	4
合計		19~41	31~57	43~73	19~41	25~49

【注】 \*1 内部割り込みの場合 1 ステートとなります。

\*2 割り込み受け後のプリフェッチおよび割り込み処理ルーチンのプリフェッチ

\*3 割り込み受け後の内部処理およびベクタフェッチ後の内部処理

\*4 外部メモリアクセス時にウェイトが挿入される場合には、ステート数が増加します。

## 5.5 使用上の注意

### 5.5.1 割り込みの発生とディスエーブルとの競合

割り込みイネーブルビットを0にクリアして割り込みを禁止する場合、割り込みの禁止はその命令実行終了後有効になります。すなわち、BCLR 命令、MOV 命令などで割り込みイネーブルビットを0にクリアする場合、命令実行中にその割り込みが発生すると、命令実行終了時点では当該割り込み許可状態にあるため命令実行終了後にその割り込み例外処理を実行します。ただし、その割り込みより優先順位の高い割り込み要求がある場合には優先順位の高い割り込み例外処理を実行し、その割り込みは無視されます。割り込み要因フラグを0にクリアする場合も同様です。

16ビットタイマのTISRAのIMIEAビットを0にクリアする場合の例を図5.8に示します。

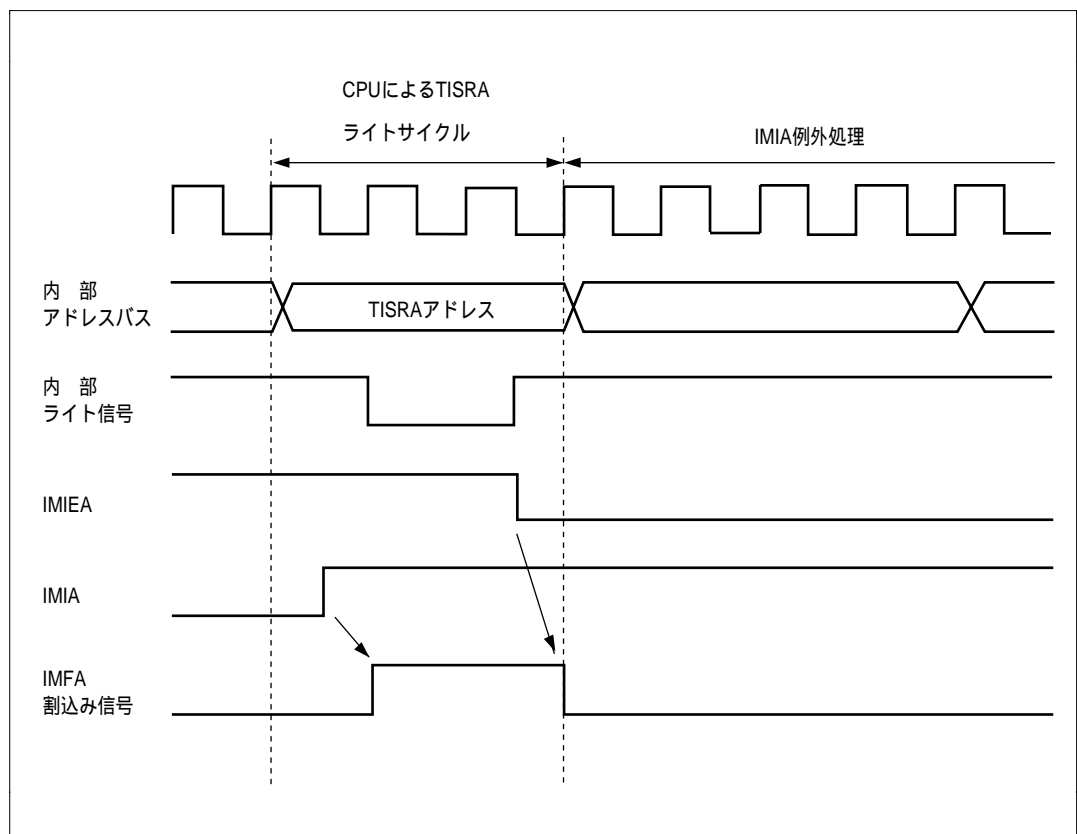


図 5.8 割り込みの発生とディスエーブルの競合

なお、割り込みをマスクした状態でイネーブルビットまたは割り込み要因フラグを0にクリアすれば、上記の競合は発生しません。

### 5.5.2 割り込みの受け付けを禁止している命令

割り込みを禁止している命令には、LDC、ANDC、ORC、XORC 命令があります。

割り込み要求が発生すると、割り込みコントローラが優先順位を判定した後、CPU に対して割り込みを要求します。そのとき、CPU が割り込みを禁止している命令を実行している場合は、その命令の実行を終了した後、必ず次の命令を実行します。

### 5.5.3 EEPMOV 命令実行中の割り込み

EEPMOV 命令は、EEPMOV . B 命令と EEPMOV . W 命令では、割り込み動作が異なります。

EEPMOV . B 命令は、転送中に NMI を含めた割り込み要求があっても転送終了まで割り込みを受け付けません。

EEPMOV . W 命令のときは、転送中に NMI 以外の割り込み要求があっても転送終了まで割り込みを受け付けません。NMI 割り込み要求の場合は、転送サイクルの切れ目で NMI 例外処理が開始されます。このときスタックされる PC の値は次命令のアドレスとなります。

このため、EEPMOV . W 命令実行中に NMI 割り込みが発生する場合には、以下のプログラムとしてください。

```
L1 : EEPMOV . W  
MOV . W R4 , R4  
BNE L1
```





---

# 6. バスコントローラ

---

## 第6章 目次

6.1	概要	129
6.1.1	特長	129
6.1.2	ブロック図	131
6.1.3	端子構成	132
6.1.4	レジスタ構成	133
6.2	各レジスタの説明	134
6.2.1	バス幅コントロールレジスタ (ABWCR)	134
6.2.2	アクセスステートコントロールレジスタ (ASTCR)	135
6.2.3	ウェイトコントロールレジスタ H、L (WCRH、WCRL)	136
6.2.4	バスリリースコントロールレジスタ (BRCR)	140
6.2.5	バスコントロールレジスタ (BCR)	143
6.2.6	チップセレクトコントロールレジスタ (CSCR)	146
6.2.7	DRAM コントロールレジスタ A (DRCRA)	147
6.2.8	DRAM コントロールレジスタ B (DRCRB)	149
6.2.9	リフレッシュタイマコントロール/ステータスレジスタ (RTMCSR)	152
6.2.10	リフレッシュタイマカウンタ (RTCNT)	154
6.2.11	リフレッシュタイムコンスタントレジスタ (RTCOR)	154
6.2.12	アドレスコントロールレジスタ (ADRCR)	155
6.3	動作説明	156
6.3.1	エリア分割	156
6.3.2	バス仕様	158
6.3.3	メモリアインタフェース	159
6.3.4	チップセレクト信号	160
6.3.5	アドレス出力方式	161
6.4	基本バスインタフェース	163
6.4.1	概要	163
6.4.2	データサイズとデータアライメント	163

6.4.3	有効ストロープ .....	165
6.4.4	各エリアの説明 .....	166
6.4.5	基本バス制御信号タイミング .....	168
6.4.6	ウェイト制御.....	176
6.5	DRAM インタフェース.....	178
6.5.1	概要.....	178
6.5.2	DRAM 空間と $\overline{\text{RAS}}$ 出力端子の設定 .....	178
6.5.3	アドレスマルチプレクス.....	179
6.5.4	データバス.....	180
6.5.5	DRAM インタフェース使用端子 .....	180
6.5.6	基本タイミング .....	181
6.5.7	プリチャージステート制御.....	182
6.5.8	ウェイト制御.....	183
6.5.9	バイトアクセス制御と $\overline{\text{CAS}}$ 出力端子.....	185
6.5.10	バースト動作.....	187
6.5.11	リフレッシュ制御.....	192
6.5.12	使用例.....	196
6.5.13	使用上の注意.....	201
6.6	インターバルタイマ.....	203
6.6.1	動作説明 .....	203
6.7	割込み要因 .....	209
6.8	バースト ROM インタフェース.....	209
6.8.1	概要.....	209
6.8.2	基本タイミング .....	209
6.8.3	ウェイト制御.....	210
6.9	アイドルサイクル.....	211
6.9.1	動作説明 .....	211
6.9.2	アイドルサイクルでの端子状態.....	214
6.10	バスアービタ.....	215
6.10.1	動作説明 .....	215
6.11	レジスタと端子入力のタイミング.....	218
6.11.1	レジスタライトタイミング.....	218
6.11.2	$\overline{\text{BREQ}}$ 端子の入力タイミング .....	219

## 6.1 概要

本 LSI はバスコントローラ (BSC) を内蔵しており、外部アドレス空間を 8 つのエリアに分割して管理します。各エリアでは、バス幅、アクセスステート数などのバス仕様を独立に設定することが可能であり、複数のメモリを容易に接続することができます。

また、バスコントローラはバス調停機能をもっており、内部バスマスタである CPU、DMA コントローラ (DMAC) および DRAM インタフェースの動作を制御するとともに、外部にバス権を解放することができます。

### 6.1.1 特長

バスコントローラの特長を次に示します。

#### 外部アドレス空間をエリア単位で管理

- ・外部空間を、1M バイトモードでは 128k バイト単位、16M バイトモードでは 2M バイト単位の 8 エリア (エリア 0 ~ 7) に分割して管理
- ・エリアごとにバス仕様を設定可能
- ・DRAM / バースト ROM インタフェースを設定可能

#### 基本バスインタフェース

- ・エリア 0 ~ 7 に対してチップセレクト ( $\overline{CS}_0 \sim \overline{CS}_7$ ) を出力可能
- ・エリアごとに、8 ビットアクセス空間 / 16 ビットアクセス空間を選択可能
- ・エリアごとに、2 ステートアクセス空間 / 3 ステートアクセス空間を選択可能
- ・エリアごとに、プログラムウェイトステートを挿入可能
- ・端子ウェイトを挿入可能

#### DRAM インタフェース

- ・エリア 2 ~ 5 に対して DRAM インタフェースを設定可能
- ・ロウアドレス / カラムアドレスのマルチプレクス出力 (8 / 9 / 10 ビット)
- ・バイトアクセス方式は 2CAS 方式
- ・バースト動作 (高速ページモード)
- ・RAS プリチャージタイム確保のための  $T_p$  サイクル挿入
- ・CAS ビフォ RAS リフレッシュとセルフリフレッシュを選択可能

#### バースト ROM インタフェース

- ・エリア 0 に対してバースト ROM インタフェースを設定可能
- ・バーストアクセスの 2 または 3 ステートを選択可能

#### アイドルサイクル挿入

- ・異なるエリア間の外部リードサイクル時、アイドルサイクルを挿入可能
- ・外部リードサイクルの直後の外部ライトサイクル時、アイドルサイクルを挿入可能

バス権調停機能（バスアービトレーション）

- ・バスアービタを内蔵し、CPU、DMAC、DRAM インタフェースおよび外部バスマスタのバス権を調停

その他

- ・リフレッシュ用カウンタ（リフレッシュタイマ）をインターバルタイマとして使用可能
- ・2つのアドレス更新モードを選択可能

## 6.1.2 ブロック図

バスコントローラのブロック図を図6.1に示します。

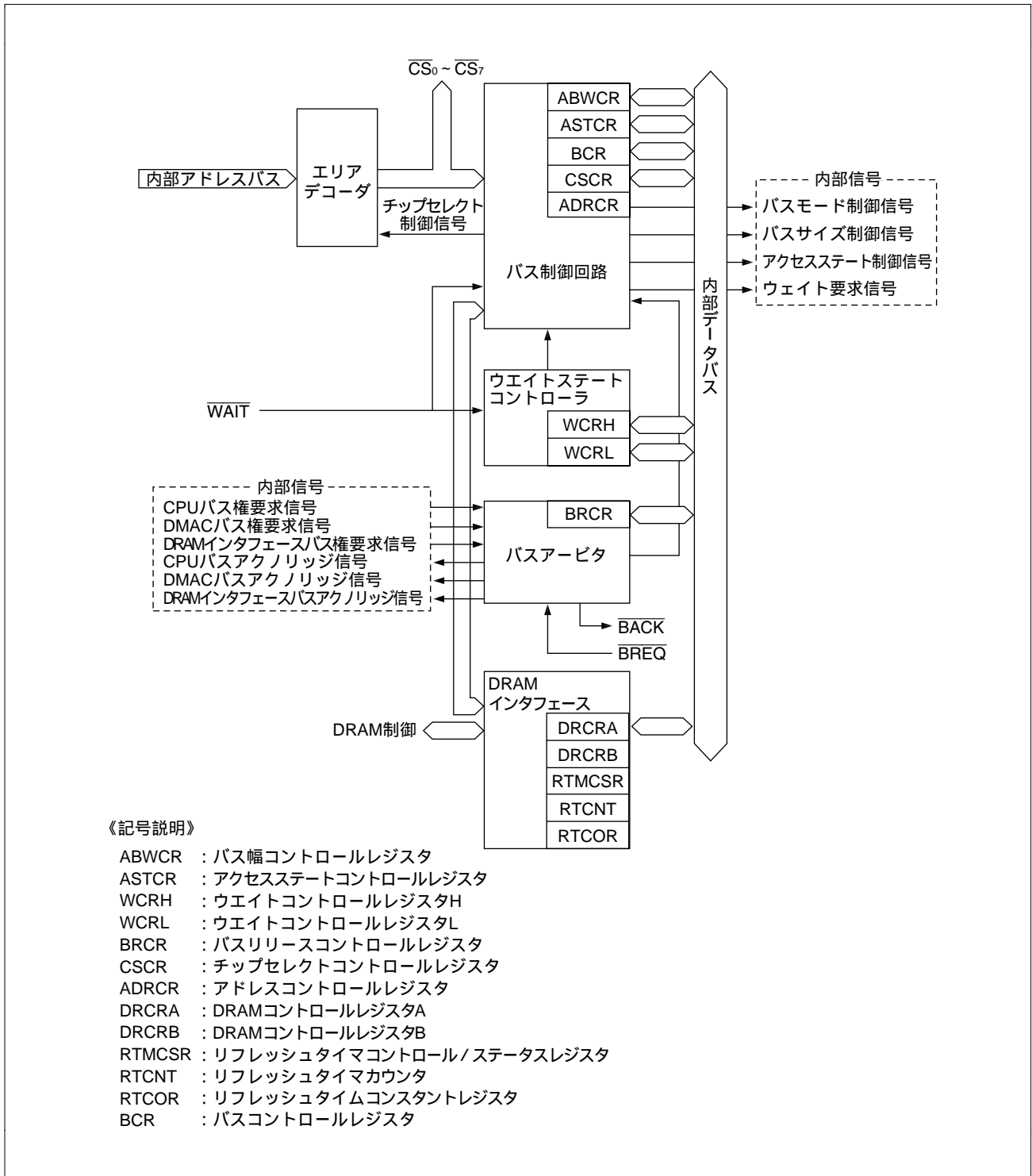


図 6.1 バスコントローラのブロック図

## 6.1.3 端子構成

バスコントローラの入出力端子を表 6.1 に示します。

表 6.1 端子構成

名称	略称	入出力	機能
チップセレクト 0~7	$\overline{CS}_0 \sim \overline{CS}_7$	出力	エリア 0~7 が選択されていることを示すストロープ信号
アドレス ストロープ	$\overline{AS}$	出力	アドレスバス上のアドレス出力が有効であることを示すストロープ信号
リード	$\overline{RD}$	出力	外部アドレス空間をリードしていることを示すストロープ信号
ハイライト	$\overline{HWR}$	出力	外部アドレス空間をライトし、データバスの上位側 ( $D_{15} \sim D_8$ ) が有効であることを示すストロープ信号
ロウライト	$\overline{LWR}$	出力	外部アドレス空間をライトし、データバスの下位側 ( $D_7 \sim D_0$ ) が有効であることを示すストロープ信号
ウェイト	$\overline{WAIT}$	入力	外部 3 ステートアクセス空間をアクセスするときのウェイト要求信号
バス権要求	$\overline{BREQ}$	入力	バス権を外部に解放する要求信号
バス権要求 アクノリッジ	$\overline{BACK}$	出力	バス権を外部に解放したことを示すアクノリッジ信号

### 6.1.4 レジスタ構成

バスコントローラのレジスタ構成を表 6.2 に示します。

表 6.2 レジスタ構成

アドレス*1	名称	略称	R/W	初期値
H'EE020	バス幅コントロールレジスタ	ABWCR	R/W	H'FF*2
H'EE021	アクセスステートコントロールレジスタ	ASTCR	R/W	H'FF
H'EE022	ウェイトコントロールレジスタ H	WCRH	R/W	H'FF
H'EE023	ウェイトコントロールレジスタ L	WCRL	R/W	H'FF
H'EE013	バスリリースコントロールレジスタ	BRCR	R/W	H'FE*3
H'EE01F	チップセレクトコントロールレジスタ	CSCR	R/W	H'0F
H'EE01E	アドレスコントロールレジスタ	ADRCR	R/W	H'FF
H'EE024	バスコントロールレジスタ	BCR	R/W	H'C6
H'EE026	DRAM コントロールレジスタ A	DRCRA	R/W	H'10
H'EE027	DRAM コントロールレジスタ B	DRCRB	R/W	H'08
H'EE028	リフレッシュタイムコントロール/ステータスレジスタ	RTMCSR	R/(W)*4	H'07
H'EE029	リフレッシュタイムカウンタ	RTCNT	R/W	H'00
H'EE02A	リフレッシュタイムコンスタントレジスタ	RTCOR	R/W	H'FF

【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。

\*2 モード 2、4 のときは、初期値 H'00 になります。

\*3 モード 3、4 のときは、初期値 H'EE になります。

\*4 ビット 7 は、フラグをクリアするための 0 ライトのみ可能です。



## 6.2 各レジスタの説明

### 6.2.1 バス幅コントロールレジスタ (ABWCR)

ABWCR は 8 ビットのリード/ライト可能なレジスタで、各エリアを 8 ビットアクセス空間または 16 ビットアクセス空間のいずれかに設定します。

ビット :	7	6	5	4	3	2	1	0
	ABW7	ABW6	ABW5	ABW4	ABW3	ABW2	ABW1	ABW0
初期値 : (モード1、3、5、6、7)	1	1	1	1	1	1	1	1
(モード2、4)	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ABWCR の内容が H'FF (全エリア 8 ビットアクセス空間) の場合、8 ビットバスモードとなり、データバスは上位側 ( $D_{15} \sim D_8$ ) が有効となります。このときポート 4 は入出力ポートとなります。ABWCR の少なくとも 1 ビットを 0 にクリアした場合には、16 ビットバスモードとなり、データバスは 16 ビット ( $D_{15} \sim D_0$ ) となります。ABWCR はリセット、またはハードウェアスタンバイモード時にモード 1、3、5、6、7 では H'FF に、モード 2、4 では H'00 にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

ビット 7~0 : エリア 7~0 バス幅コントロール (ABW7~ABW0)

対応するエリアを 8 ビットアクセス空間とするか 16 ビットアクセス空間とするかを選択します。

ビット 7~0	説明
ABW7 ~ ABW0	
0	エリア 7~0 を 16 ビットアクセス空間に設定
1	エリア 7~0 を 8 ビットアクセス空間に設定

ABWCR は、外部メモリ空間のデータバス幅を指定します。内蔵メモリ、内部 I/O レジスタのデータバス幅は ABWCR の設定値にかかわらず固定です。したがって、シングルチップモード (モード 6、7) では設定値には意味がありません。

## 6.2.2 アクセスステートコントロールレジスタ (ASTCR)

ASTCR は 8 ビットのリード/ライト可能なレジスタで、各エリアを 2 ステートアクセス空間または 3 ステートアクセス空間のいずれかに設定します。

ビット:	7	6	5	4	3	2	1	0
	AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0
初期値:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

各エリアのアクセスステート数を選択するビットです。

ASTCR はリセット、またはハードウェアスタンバイモード時に、H'FF にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

ビット 7~0: エリア 7~0 アクセスステートコントロール (AST7~AST0)

対応するエリアを 2 ステートアクセス空間とするか 3 ステートアクセス空間とするかを選択します。

ビット 7~0	説明
AST7~AST0	
0	エリア 7~0 を 2 ステートアクセス空間に設定
1	エリア 7~0 を 3 ステートアクセス空間に設定 (初期値)

ASTCR は、外部メモリ空間のアクセスステート数を設定します。内蔵メモリ、内部 I/O レジスタに対するアクセスステート数は ASTCR の設定値にかかわらず固定です。したがって、シングルチップモード (モード 6、7) では設定値は無効です。

なお、対応するエリアが DRAM コントロールレジスタ A (DRCRA) の DRAS2~DRAS0 ビットにより DRAM 空間に設定されている場合、アクセスステート数は AST ビットの設定値に依存しません。また、AST ビットの値が 0 の場合は、プログラマブルウェイトは挿入されません。

### 6.2.3 ウェイトコントロールレジスタ H、L (WCRH、WCRL)

WCRH、WCRL は、それぞれ 8 ビットのリード/ライト可能なレジスタで、各エリアのプログラムウェイトステート数を選択します。

内蔵メモリおよび内部 I/O レジスタに対するアクセスステート数は WCRH、WCRL の設定値にかかわらず固定です。

WCRH、WCRL は、リセットおよびハードウェアスタンバイモード時に、H'FF にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

#### (1) WCRH

ビット:	7	6	5	4	3	2	1	0
	W71	W70	W61	W60	W51	W50	W41	W40
初期値:	1	1	1	1	1	1	1	1
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

#### ビット 7、6 : エリア 7 ウェイトコントロール 1、0 (W71、W70)

ASTCR の AST7 ビットが 1 にセットされた状態でエリア 7 の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット 7	ビット 6	説明
W71	W70	
0	0	エリア 7 の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア 7 の外部空間アクセス時、プログラムウェイトを 1 ステート挿入
1	0	エリア 7 の外部空間アクセス時、プログラムウェイトを 2 ステート挿入
	1	エリア 7 の外部空間アクセス時、プログラムウェイトを 3 ステート挿入 (初期値)

ビット5、4：エリア6 ウェイトコントロール1、0 (W61、W60)

ASTCRのAST6ビットが1にセットされた状態でエリア6の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット5	ビット4	説明
W61	W60	
0	0	エリア6の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア6の外部空間アクセス時、プログラムウェイトを1ステート挿入
1	0	エリア6の外部空間アクセス時、プログラムウェイトを2ステート挿入
	1	エリア6の外部空間アクセス時、プログラムウェイトを3ステート挿入 (初期値)

ビット3、2：エリア5 ウェイトコントロール1、0 (W51、W50)

ASTCRのAST5ビットが1にセットされた状態でエリア5の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット3	ビット2	説明
W51	W50	
0	0	エリア5の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア5の外部空間アクセス時、プログラムウェイトを1ステート挿入
1	0	エリア5の外部空間アクセス時、プログラムウェイトを2ステート挿入
	1	エリア5の外部空間アクセス時、プログラムウェイトを3ステート挿入 (初期値)

ビット1、0：エリア4 ウェイトコントロール1、0 (W41、W40)

ASTCRのAST4ビットが1にセットされた状態でエリア4の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット1	ビット0	説明
W41	W40	
0	0	エリア4の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア4の外部空間アクセス時、プログラムウェイトを1ステート挿入
1	0	エリア4の外部空間アクセス時、プログラムウェイトを2ステート挿入
	1	エリア4の外部空間アクセス時、プログラムウェイトを3ステート挿入 (初期値)

## (2) WCRL

ビット:	7	6	5	4	3	2	1	0
	W31	W30	W21	W20	W11	W10	W01	W00
初期値:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ビット7、6：エリア3ウェイトコントロール1、0（W31、W30）

ASTCRのAST3ビットが1にセットされた状態でエリア3の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット7	ビット6	説明
W31	W30	
0	0	エリア3の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア3の外部空間アクセス時、プログラムウェイトを1ステート挿入
1	0	エリア3の外部空間アクセス時、プログラムウェイトを2ステート挿入
	1	エリア3の外部空間アクセス時、プログラムウェイトを3ステート挿入 (初期値)

ビット5、4：エリア2ウェイトコントロール（W21、W20）

ASTCRのAST2ビットが1にセットされた状態でエリア2の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット5	ビット4	説明
W21	W20	
0	0	エリア2の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア2の外部空間アクセス時、プログラムウェイトを1ステート挿入
1	0	エリア2の外部空間アクセス時、プログラムウェイトを2ステート挿入
	1	エリア2の外部空間アクセス時、プログラムウェイトを3ステート挿入 (初期値)

ビット3、2：エリア1ウェイトコントロール1、0（W11、W10）

ASTCRのAST1ビットが1にセットされた状態でエリア1の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット3	ビット2	説明
W11	W10	
0	0	エリア1の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア1の外部空間アクセス時、プログラムウェイトを1ステート挿入
1	0	エリア1の外部空間アクセス時、プログラムウェイトを2ステート挿入
	1	エリア1の外部空間アクセス時、プログラムウェイトを3ステート挿入 (初期値)

ビット1、0：エリア0ウェイトコントロール1、0（W01、W00）

ASTCRのAST0ビットが1にセットされた状態でエリア0の外部空間をアクセスするときの、プログラムウェイトステート数を選択します。

ビット1	ビット0	説明
W01	W00	
0	0	エリア0の外部空間アクセス時、プログラムウェイトを挿入しない
	1	エリア0の外部空間アクセス時、プログラムウェイトを1ステート挿入
1	0	エリア0の外部空間アクセス時、プログラムウェイトを2ステート挿入
	1	エリア0の外部空間アクセス時、プログラムウェイトを3ステート挿入 (初期値)

## 6.2.4 バスリリースコントロールレジスタ (BRCR)

BRCR は 8 ビットのリード/ライト可能なレジスタで、アドレスバス ( $A_{23} \sim A_{20}$ ) 出力の選択、バス権の外部に対する解放を許可/禁止します。

ビット:	7	6	5	4	3	2	1	0
	A23E	A22E	A21E	A20E	—	—	—	BRLE
モード1、2、6、7 初期値 :	1	1	1	1	1	1	1	0
R/W :	—	—	—	—	—	—	—	R/W
モード3、4 初期値 :	1	1	1	0	1	1	1	0
R/W :	R/W	R/W	R/W	—	—	—	—	R/W
モード5 初期値 :	1	1	1	1	1	1	1	0
R/W :	R/W	R/W	R/W	R/W	—	—	—	R/W

リザーブビット

アドレス23~20イネーブル  
PA<sub>7</sub>~PA<sub>4</sub>をアドレス出力端子  
A<sub>23</sub>~A<sub>20</sub>として使用するた  
めのビットです。

バスリリースイネーブル  
バス権の外部に対する解放を  
許可/禁止するビットです。

BRCR はリセット、またはハードウェアスタンバイモード時にモード 1、2、5~7 の場合は H'FE に、モード 3、4 の場合は H'EE にイニシャライズされます。ソフトウェアスタンバイモード時にはイニシャライズされません。

### ビット7 : アドレス 23 イネーブル (A23E)

PA<sub>4</sub> をアドレス出力端子 A<sub>23</sub> として使用するためのビットです。0 をライトするとアドレス出力 A<sub>23</sub> となります。モード 3、4、5 以外ではこのビットはライトできず、通常のポート機能となります。

ビット7	説明
A23E	
0	PA <sub>4</sub> はアドレス出力端子 A <sub>23</sub>
1	PA <sub>4</sub> は入出力端子 (初期値)

## ビット6：アドレス22イネーブル（A22E）

PA<sub>5</sub>をアドレス出力端子A<sub>22</sub>として使用するためのビットです。0をライトするとアドレス出力A<sub>22</sub>となります。モード3、4、5以外ではこのビットはライトできず、通常のポート機能となります。

ビット6	説明
A22E	
0	PA <sub>5</sub> はアドレス出力端子A <sub>22</sub>
1	PA <sub>5</sub> は入出力端子 (初期値)

## ビット5：アドレス21イネーブル（A21E）

PA<sub>6</sub>をアドレス出力端子A<sub>21</sub>として使用するためのビットです。0をライトするとアドレス出力A<sub>21</sub>となります。モード3、4、5以外ではこのビットはライトできず、通常のポート機能となります。

ビット5	説明
A21E	
0	PA <sub>6</sub> はアドレス出力端子A <sub>21</sub>
1	PA <sub>6</sub> は入出力端子 (初期値)

## ビット4：アドレス20イネーブル（A20E）

PA<sub>7</sub>をアドレス出力端子A<sub>20</sub>として使用するためのビットです。0をライトするとアドレス出力A<sub>20</sub>となります。モード5以外ではこのビットはライトできません。

ビット4	説明
A20E	
0	PA <sub>7</sub> はアドレス出力端子A <sub>20</sub> (モード3、4のときの初期値)
1	PA <sub>7</sub> は入出力端子 (モード1、2、5~7のときの初期値)

## ビット3~1：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。



ビット0 : バスリリースイネーブル (BRLE)

バス権の外部に対する解放を許可 / 禁止します。

ビット0	説明
BRLE	
0	バス権の外部に対する解放を禁止し、 $\overline{\text{BREQ}}$ 、 $\overline{\text{BACK}}$ 端子は入出力端子として使用可 (初期値)
1	バス権の外部に対する解放を許可

## 6.2.5 バスコントロールレジスタ (BCR)

ビット:	7	6	5	4	3	2	1	0
	ICIS1	ICIS0	BROME	BRSTS1	BRSTS0	EMC	RDEA	WAITE
初期値:	1	1	0	0	0	1	1	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

BCRは8ビットのリード/ライト可能なレジスタで、アイドルサイクル挿入の許可または禁止、アドレスマップの選択、エリアの分割単位の選択、 $\overline{\text{WAIT}}$ 端子入力の許可または禁止を行います。

BCRは、リセットおよびハードウェアスタンバイモード時に、H'C6にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

### ビット7: アイドルサイクル挿入1 (ICIS1)

異なるエリアの外部リードサイクルが連続する場合、バスサイクルの間にアイドルサイクルを1ステート挿入するか、挿入しないかを選択します。

ビット7	説明
ICIS1	
0	異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入しない
1	異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入する (初期値)

### ビット6: アイドルサイクル挿入0 (ICIS0)

外部リードサイクルと外部ライトサイクルが連続する場合、バスサイクルの間にアイドルサイクルを1ステート挿入するか、挿入しないかを選択します。

ビット6	説明
ICIS0	
0	外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入しない
1	外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入する (初期値)

## ビット5：バーストROMイネーブル (BROME)

エリア0をバーストROMインタフェースとするかを選択します。

ビット5	説明
BROME	
0	エリア0は基本バスインタフェース (初期値)
1	エリア0はバーストROMインタフェース

## ビット4：バーストサイクルセレクト1 (BRSTS1)

バーストROMインタフェースのバーストサイクル数を選択します。

ビット4	説明
BRSTS1	
0	バーストアクセスのサイクル数は2ステート (初期値)
1	バーストアクセスのサイクル数は3ステート

## ビット3：バーストサイクルセレクト0 (BRSTS0)

バーストROMインタフェースのバーストアクセス可能なワード数を選択します。

ビット3	説明
BRSTS0	
0	バーストアクセスは最大4ワード (A3より上位アドレス一致でバーストアクセス) (初期値)
1	バーストアクセスは最大8ワード (A4より上位アドレス一致でバーストアクセス)

## ビット2：拡張メモリマップコントロール (EMC)

2種類のメモリマップを選択できます。

ビット2	説明
EMC	
0	3章MCU動作モード、3.6各動作モードのメモリマップの図3.2のメモリマップとなります
1	3章MCU動作モード、3.6各動作モードのメモリマップの図3.1のメモリマップとなります (初期値)

なを本ビットを0にクリアすると内部I/Oレジスタのアドレスが一部、移動しますので注意してください。移動後のアドレスは「付録B.2アドレス一覧(EMCビット=0のとき)」を参照してください。

本ビットはモード6のときは使用できません。モード6のときは0にクリアしないでく

ださい。また、RDEA ビットが0のとき、本ビットを0にクリアしないでください。

#### ビット1：エリア分割単位選択（RDEA）

メモリアップのエリアの分割単位を選択します。

本ビットはモード3、4、5で有効です。モード1、2、6、7では、無効です。

また、EMC ビットが0のとき本ビットを0にしないでください。

ビット1	説明	
RDEA		
0	エリア分割は、エリア0：2MB エリア1：2MB エリア2：8MB エリア3：2MB	エリア4：1.93MB エリア5：4kB エリア6：23.75kB エリア7：22B
1	エリア分割は、エリア0～エリア7まで等分割（2MB） （初期値）	

#### ビット0：WAIT 端子イネーブル（WAITE）

$\overline{\text{WAIT}}$  端子によるウェイト入力の許可または禁止を選択します。

ビット0	説明
WAITE	
0	$\overline{\text{WAIT}}$ 端子によるウェイト入力を禁止。 $\overline{\text{WAIT}}$ 端子は入出力ポートとして使用可 （初期値）
1	$\overline{\text{WAIT}}$ 端子によるウェイト入力を許可

## 6.2.6 チップセレクトコントロールレジスタ (CSCR)

CSCR は 8 ビットのリード/ライト可能なレジスタで、チップセレクト信号 ( $\overline{CS}_7 \sim \overline{CS}_4$ ) の出力を許可/禁止します。

本レジスタでチップセレクト信号 ( $\overline{CS}_7 \sim \overline{CS}_4$ ) 出力を選択すると、端子機能は他の機能に優先してチップセレクト信号 ( $\overline{CS}_7 \sim \overline{CS}_4$ ) 出力となります。なお、本レジスタはシングルチップモードではライトできません。

ビット:	7	6	5	4	3	2	1	0
	CS7E	CS6E	CS5E	CS4E	—	—	—	—
初期値:	0	0	0	0	1	1	1	1
R/W :	R/W	R/W	R/W	R/W	—	—	—	—

リザーブビット

チップセレクト7~4イネーブル  
チップセレクト信号の出力を許可/  
禁止するビットです。

CSCR はリセット、またはハードウェアスタンバイモード時に H'0F にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

ビット 7~4: チップセレクト 7~4 イネーブル (CS7E ~ CS4E)

対応するチップセレクト信号の出力を許可/禁止します。

ビット n	説明
CSnE	
0	チップセレクト信号 ( $\overline{CS}_n$ ) の出力を禁止 <span style="float: right;">(初期値)</span>
1	チップセレクト信号 ( $\overline{CS}_n$ ) の出力を許可

【注】 n = 7~4

ビット 3~0: リザーブビット

リザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。

## 6.2.7 DRAM コントロールレジスタ A (DRCRA)

ビット:	7	6	5	4	3	2	1	0
	DRAS2	DRAS1	DRAS0	—	BE	RDM	SRFMD	RFSHE
初期値:	0	0	0	1	0	0	0	0
R/W :	R/W	R/W	R/W	—	R/W	R/W	R/W	R/W

DRCRA は 8 ビットのリード/ライト可能なレジスタで、DRAM インタフェースの機能するエリアの選択、アクセスモード、セルフリフレッシュの許可/禁止およびリフレッシュ端子出力の許可/禁止の選択を行います。

DRCRA は、リセットおよびハードウェアスタンバイモード時に、H'10 にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

## ビット 7~5 : DRAM エリアセレクト (DRAS2~DRAS0)

拡張モードのときに、エリア 2~5 から DRAM インタフェースとして機能するエリア (DRAM 空間) を選択します。また、同時に各 DRAM 空間に対応する  $\overline{RAS}$  出力端子を選択します。

ビット 7	ビット 6	ビット 5	説明			
DRAS2	DRAS1	DRAS0	エリア 5	エリア 4	エリア 3	エリア 2
0	0	0	通常	通常	通常	通常
		1	通常	通常	通常	DRAM 空間 ( $\overline{CS}_2$ )
	1	0	通常	通常	DRAM 空間 ( $\overline{CS}_3$ )	DRAM 空間 ( $\overline{CS}_2$ )
		1	通常	通常	DRAM 空間 ( $\overline{CS}_2$ ) *	
1	0	0	通常	DRAM 空間 ( $\overline{CS}_4$ )	DRAM 空間 ( $\overline{CS}_3$ )	DRAM 空間 ( $\overline{CS}_2$ )
		1	DRAM 空間 ( $\overline{CS}_5$ )	DRAM 空間 ( $\overline{CS}_4$ )	DRAM 空間 ( $\overline{CS}_3$ )	DRAM 空間 ( $\overline{CS}_2$ )
	1	0	DRAM 空間 ( $\overline{CS}_4$ ) *		DRAM 空間 ( $\overline{CS}_2$ ) *	
		1	DRAM 空間 ( $\overline{CS}_2$ ) *			

【注】 \* 1 本の  $\overline{CS}_n$  端子が複数エリアに共通の  $\overline{RAS}$  出力端子となります。このとき不要となる  $\overline{CS}_n$  端子は入出力ポートとして使用可能になります。

拡張モードのときに DRAS2~0 のいずれかが 1 にセットされていると、DRCRB、RTMCSR、RTCNT および RTCOR へのライトはできません。ただし、RTMCSR の CMF フラグについては、フラグをクリアするための 0 ライトのみ可能です。

なお、DRAS2～DRAS0 に任意の値が設定されている場合、000 以外の異なる値のライトは行わないでください。

#### ビット4：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

#### ビット3：バーストアクセスイネーブル（BE）

DRAM 空間に対するバーストアクセスの許可または禁止を選択します。DRAM 空間のバーストアクセスは高速ページモードになります。

ビット3	説明
BE	
0	バースト禁止（常にフルアクセス） <span style="float: right;">（初期値）</span>
1	DRAM空間アクセス時 高速ページモードでアクセス

#### ビット2：RAS ダウンモード（RDM）

DRAM 空間に対するバーストアクセスが許可されている場合（BE=1）、DRAM に対するアクセスが途切れたときに、 $\overline{\text{RAS}}$  信号を Low レベルに保持したままで次の DRAM へのアクセスを待つか（RAS ダウンモード）、 $\overline{\text{RAS}}$  信号を High レベルに戻すか（RAS アップモード）を選択します。

なお、 $\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$  の出力端子として  $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$  を使用する場合、注意が必要です。

詳細は「6.5.10 バースト動作」の（2）RAS ダウンモードと RAS アップモードを参照してください。

ビット2	説明
RDM	
0	DRAM インタフェース：RAS アップモードを選択 <span style="float: right;">（初期値）</span>
1	DRAM インタフェース：RAS ダウンモードを選択

#### ビット1：セルフリフレッシュモード（SRFMD）

ソフトウェアスタンバイモード時、DRAM のセルフリフレッシュを指定します。

エリア 2～5 のいずれかが DRAM 空間に設定されているとき、SRFMD ビットを 1 にセットした後に、ソフトウェアスタンバイモードに遷移すると、DRAM のセルフリフレッシュが可能となります。

SRFMD の設定にかかわらず、ソフトウェアスタンバイモードの解除により、通常のアクセス状態に戻ります。

ビット1	説明
SRFMD	
0	ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュを禁止（初期値）
1	ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュが可能

ビット0：リフレッシュ端子イネーブル（RFSHE）

$\overline{\text{RFSH}}$  端子のリフレッシュ信号出力を許可 / 禁止します。エリア 2~5 を DRAM 空間に設定しない場合は、本ビットを 1 に設定しないでください。

ビット0	説明
RFSHE	
0	$\overline{\text{RFSH}}$ 端子のリフレッシュ信号出力を禁止（初期値） （ $\overline{\text{RFSH}}$ 端子は入出力ポートとして使用可）
1	$\overline{\text{RFSH}}$ 端子のリフレッシュ信号出力を許可

## 6.2.8 DRAM コントロールレジスタ B（DRCRB）

ビット：	7	6	5	4	3	2	1	0
	MXC1	MXC0	CSEL	RCYCE	—	TPC	RCW	RLW
初期値：	0	0	0	0	1	0	0	0
R/W：	R/W	R/W	R/W	R/W	—	R/W	R/W	R/W

DRCRB は 8 ビットのリード / ライト可能なレジスタで、DRAM インタフェースのアドレスマルチプレクスのカラムアドレスビット数、カラムアドレスストロブの出力端子、リフレッシュサイクル挿入の許可 / 禁止、プリチャージサイクル数、 $\overline{\text{RAS}} \cdot \overline{\text{CAS}}$  間のウェイトステート挿入の許可 / 禁止、およびリフレッシュサイクルのウェイトステート挿入の許可 / 禁止を選択します。

DRCRB は、リセットおよびハードウェアスタンバイモード時に、H'08 にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

なお、DRCRA の DRAS2 ~ DRAS0 が全て 0 の時、本レジスタの設定は無効となります。



ビット7、6：マルチプレクスコントロール1、0（MXC1、MXC0）

DRAM インタフェースの、ロウアドレス/カラムアドレスのマルチプレクスの方法を選択します。バースト動作時に比較の対象となるロウアドレスは、本ビットの設定と ABWCR で設定された各エリアのバス幅で決まります。

ビット7	ビット6	説明
MXC1	MXC0	
0	0	カラムアドレス：8ビット 比較対象アドレス： モード1、2      8ビットアクセス空間 $A_{19} \sim A_8$ 16ビットアクセス空間 $A_{19} \sim A_9$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_8$ 16ビットアクセス空間 $A_{23} \sim A_9$
	1	カラムアドレス：9ビット 比較対象アドレス： モード1、2      8ビットアクセス空間 $A_{19} \sim A_9$ 16ビットアクセス空間 $A_{19} \sim A_{10}$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_9$ 16ビットアクセス空間 $A_{23} \sim A_{10}$
1	0	カラムアドレス：10ビット 比較対象アドレス： モード1、2      8ビットアクセス空間 $A_{19} \sim A_{10}$ 16ビットアクセス空間 $A_{19} \sim A_{11}$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_{10}$ 16ビットアクセス空間 $A_{23} \sim A_{11}$
	1	設定禁止

ビット5： $\overline{\text{CAS}}$  出力端子選択（CSEL）

エリア2～5をDRAM空間に設定した場合の $\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$ の出力端子を選択します。

ビット5	説明
CSEL	
0	$\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$ の出力端子：PB4、PB5を選択 <span style="float: right;">（初期値）</span>
1	$\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$ の出力端子： $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$ を選択

## ビット4：リフレッシュサイクルイネーブル（RCYCE）

CAS ピフォ RAS リフレッシュサイクルの挿入を許可または禁止します。エリア 2~5 のいずれも DRAM 空間に設定されていないときは、本ビットの設定によらずリフレッシュサイクルは挿入されません。

ビット4	説明
RCYCE	
0	リフレッシュサイクルを禁止 (初期値)
1	DRAM に対するリフレッシュサイクルを許可

## ビット3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

## ビット2：TP サイクルコントロール（TPC）

DRAM のリード/ライトサイクルおよび CAS ピフォ RAS リフレッシュサイクルのプリチャージサイクル ( $T_p$ ) を 1 ステートにするか、2 ステートにするかを選択します。なお、本ビットの設定はセルフリフレッシュ機能には影響を与えません。

ビット2	説明
TPC	
0	プリチャージサイクルを 1 ステート挿入 (初期値)
1	プリチャージサイクルを 2 ステート挿入

ビット1： $\overline{\text{RAS}} \cdot \overline{\text{CAS}}$  間ウェイト（RCW）

DRAM のリード/ライトサイクルの  $T_r$ - $T_{Cl}$  間ウェイトステート ( $T_{rw}$ ) の挿入を制御します。なお、本ビットの設定はリフレッシュサイクルに対し影響を与えません。

ビット1	説明
RCW	
0	ウェイトステート ( $T_{rw}$ ) の挿入を禁止 (初期値)
1	ウェイトステート ( $T_{rw}$ ) を 1 ステート挿入

## ビット0：リフレッシュサイクルウェイトコントロール（RLW）

CAS ピフォ RAS リフレッシュサイクルに対するウェイトステート ( $T_{RW}$ ) の挿入を制御します。なお、本ビットの設定は DRAM のリード/ライトサイクルに対し影響を与えません。

ビット0	説明
RLW	
0	ウェイトステート ( $T_{RW}$ ) の挿入を禁止 (初期値)
1	ウェイトステート ( $T_{RW}$ ) を 1 ステート挿入

## 6.2.9 リフレッシュタイムコントロール/ステータスレジスタ (RTMCSR)

ビット:	7	6	5	4	3	2	1	0
	CMF	CMIE	CKS2	CKS1	CKS0	—	—	—
初期値:	0	0	0	0	0	1	1	1
R/W :	R/(W)*	R/W	R/W	R/W	R/W	—	—	—

RTMCSR は 8 ビットのリード/ライト可能なレジスタで、リフレッシュタイムカウンタのクロックの選択を行います。また、インターバルタイムとして使用する場合は、割込要求の許可/禁止も行います。RTMCSR のビット 7、6 はリセットおよびスタンバイモード時に 0 にイニシャライズされます。ビット 5~3 はリセットおよびハードウェアスタンバイモード時に 0 にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

【注】 \* フラグをクリアするための 0 ライトのみ可能です。

## ビット 7 : コンペアマッチフラグ (CMF)

RTCNT と RTCOR の値が一致したことを示すステータスフラグです。

ビット 7	説明
CMF	
0	[ クリア条件 ] (1) リセットまたはスタンバイモード時 (2) CMF=1 の状態で、CMF フラグをリードした後、CMF フラグに 0 をライトしたとき (初期値)
1	[ セット条件 ] RTCNT = RTCOR になったとき

## ビット 6 : コンペアマッチ割込みイネーブル (CMIE)

RTMCSR の CMF フラグが 1 にセットされたとき、CMF フラグによる割込み要求 (CMF) を許可または禁止します。

エリア 2~5 のいずれかが DRAM 空間に設定されている場合、CMIE ビットは常に 0 にクリアされています。

ビット 6	説明
CMIE	
0	CMF フラグによる割込み要求 (CMI) を禁止 (初期値)
1	CMF フラグによる割込み要求 (CMI) を許可

## ビット5~3：リフレッシュカウンタクロックセレクト（CKS2~CKS0）

システムクロック（ ）を分周して得られる7種類の内部クロックから RTCNTにを入力するクロックを選択します。CKS2~CKS0 ビットで入力クロックを選択すると、RTCNTがカウントアップを開始します。

ビット5	ビット4	ビット3	説明
CKS2	CKS1	CKS0	
0	0	0	カウント動作停止 (初期値)
		1	/2でカウント
	1	0	/8でカウント
		1	/32でカウント
1	0	0	/128でカウント
		1	/512でカウント
	1	0	/2048でカウント
		1	/4096でカウント

## ビット2~0：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

### 6.2.10 リフレッシュタイマカウンタ (RTCNT)

ビット:	7	6	5	4	3	2	1	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

RTCNT は 8 ビットのリード/ライト可能なアップカウンタです。

RTCNT は、RTMCSR の CKS2~CKS0 ビットで選択された内部クロックにより、カウントアップします。

RTCNT が RTCOR に一致 (コンペアマッチ) すると、RTMCSR の CMF フラグが 1 にセットされ、RTCNT は H'00 にクリアされます。このとき、DRCRB の RCYCE ビットが 1 にセットされていると、リフレッシュサイクルが起動されます。また、RTMCSR の CMIE ビットが 1 にセットされているとき、コンペアマッチ割込み (CMI) が発生します。

RTCNT は、リセットおよびスタンバイモード時 H'00 にイニシャライズされます。

### 6.2.11 リフレッシュタイムコンスタントレジスタ (RTCOR)

ビット:	7	6	5	4	3	2	1	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
初期値:	1	1	1	1	1	1	1	1
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

RTCOR は 8 ビットのリード/ライト可能なレジスタで、RTCNT とのコンペアマッチ周期を設定します。

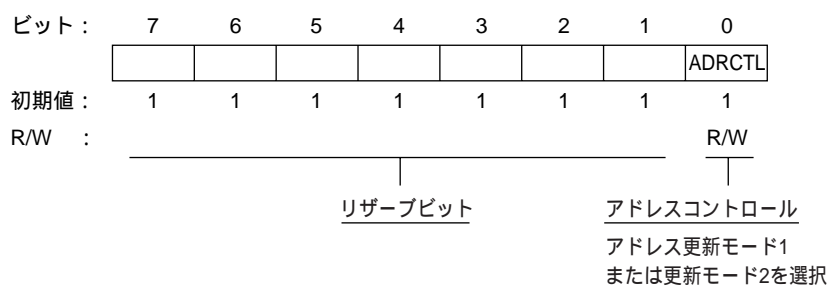
RTCOR と RTCNT の値は常に比較されており、両方の値が一致すると、RTMCSR の CMF フラグが 1 にセットされ、RTCNT は H'00 にクリアされます。

RTCOR は、リセットおよびハードウェアスタンバイモード時 H'FF にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

【注】 本レジスタへのアクセスは、かならずバイトアクセスとしてください。

## 6.2.12 アドレスコントロールレジスタ (ADRCR)

ADRCR は 8 ビットのリード/ライト可能なレジスタで、アドレスの出力方式をアドレス更新モード 1 またはアドレス更新モード 2 のいずれかに設定します。



ADRCR は、リセット、またはハードウェアスタンバイモード時に、H'FF にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

ビット 7~1: リザーブビット

ビット 7~1 をリードすると、常に 1 が読み出されます。ライトは無効です。

ビット 0: アドレスコントロールビット (ADRCTL)

アドレス出力方式を選択します。

ビット 0	説明
ADRCTL	
0	アドレス更新モード 2 を選択
1	アドレス更新モード 1 を選択 <span style="float: right;">(初期値)</span>

## 6.3 動作説明

### 6.3.1 エリア分割

外部アドレス空間は、1M バイトモードのとき 128k バイト、16M バイトモードのとき 2M バイトごとのエリア 0～7 に分割されています。メモリマップの概要を図 6.2 に示します。

H' 00000	エリア0 (128k バイト)	H' 000000	エリア0 (2M バイト)
H' 1FFFF		H' 1FFFFFFF	
H' 20000	エリア1 (128k バイト)	H' 200000	エリア1 (2M バイト)
H' 3FFFF		H' 3FFFFFFF	
H' 40000	エリア2 (128k バイト)	H' 400000	エリア2 (2M バイト)
H' 5FFFF		H' 5FFFFFFF	
H' 60000	エリア3 (128k バイト)	H' 600000	エリア3 (2M バイト)
H' 7FFFF		H' 7FFFFFFF	
H' 80000	エリア4 (128k バイト)	H' 800000	エリア4 (2M バイト)
H' 9FFFF		H' 9FFFFFFF	
H' A0000	エリア5 (128k バイト)	H' A00000	エリア5 (2M バイト)
H' BFFFF		H' BFFFFFFF	
H' C0000	エリア6 (128k バイト)	H' C00000	エリア6 (2M バイト)
H' DFFFF		H' DFFFFFFF	
H' E0000	エリア7 (128k バイト)	H' E00000	エリア7 (2M バイト)
H' FFFFF		H' FFFFFFFF	

(a) 1M バイトモード (モード1、2)

(b) 16M バイトモード (モード3、4、5)

図 6.2 各動作モードにおけるアクセスエリアマップ

各エリアごとに選択信号 ( $\overline{CS}_0 \sim \overline{CS}_7$ ) を出力することができます。また、各エリアのバス仕様は、ABWCR、ASTCR、WCRH、WCRL で指定されます。

また、16M バイトモードのとき BCR の RDEA ビットにより、エリア分割の単位を選択することができます。

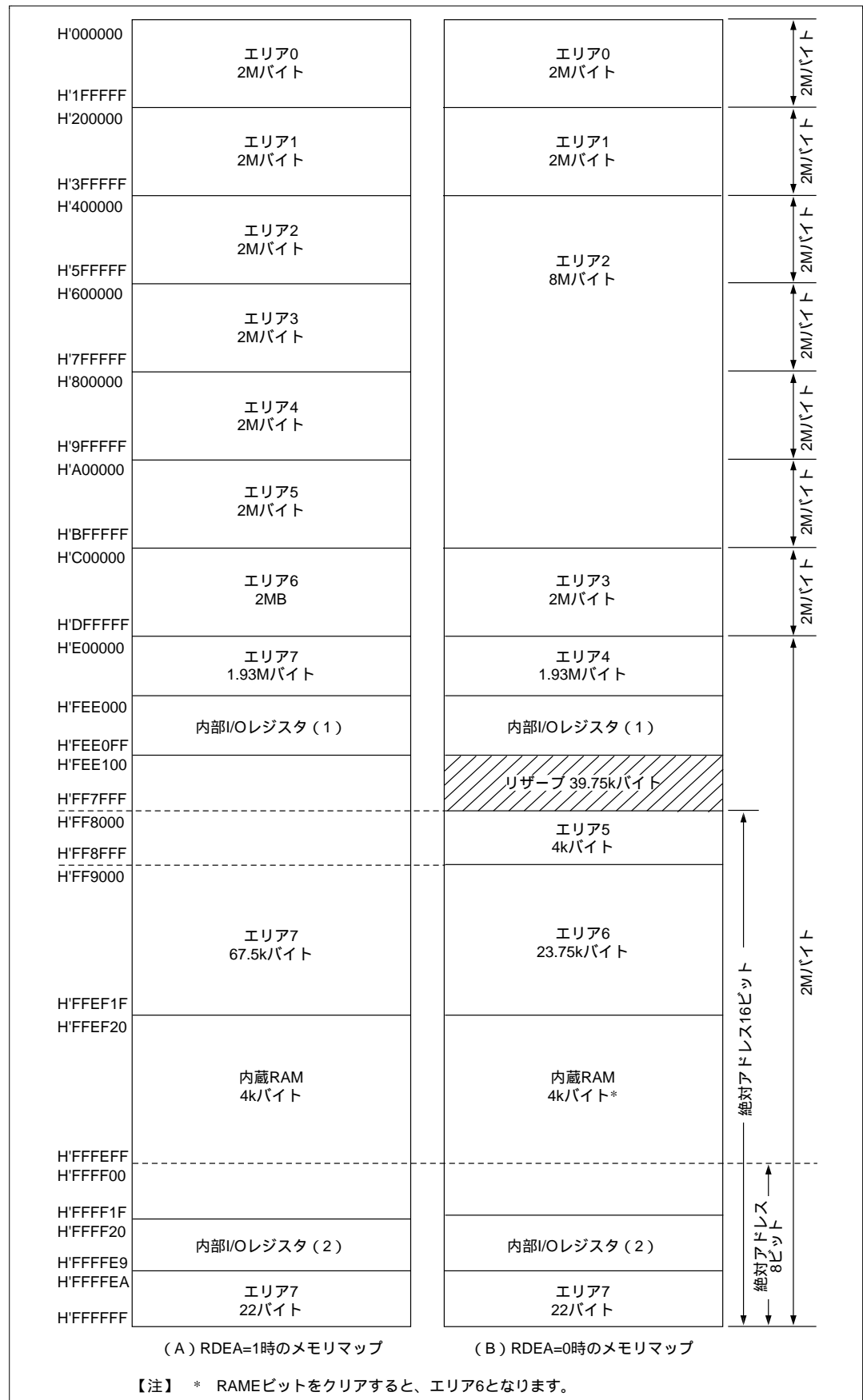


図 6.3 16Mバイトモード時のメモリマップ



### 6.3.2 バス仕様

外部空間のバス仕様は、(1)バス幅、(2)アクセスステート数、(3)プログラムウェイトステート数、の3つの要素で構成されます。

なお、内蔵メモリ、内蔵I/Oレジスタは、バス幅・アクセスステート数は固定で、バスコントローラの影響を受けません。

#### (1) バス幅

バス幅はABWCRにより、8ビットまたは16ビットを選択します。8ビットバスを選択したエリアが8ビットアクセス空間、16ビットバスを選択したエリアが16ビットアクセス空間です。

全てのエリアを8ビットアクセス空間に設定すると8ビットバスモードに、いずれかのエリアを16ビットアクセス空間に設定すると16ビットバスモードになります。

#### (2) アクセスステート数

アクセスステート数は、ASTCRにより、2ステートまたは3ステートを選択します。2ステートアクセスを選択したエリアが2ステートアクセス空間、3ステートアクセスを選択したエリアが3ステートアクセス空間です。

なお、DRAM空間ではASTCRによらず4ステートアクセスとなります。

2ステートアクセス空間に設定すると、ウェイトの挿入が禁止されます。

#### (3) プログラムウェイトステート数

ASTCRによって3ステートアクセス空間に設定したとき、WCRH、WCRLにより、自動的に挿入するプログラムウェイトステート数を選択します。プログラムウェイトは0~3ステートを選択可能です。

DRAM空間でASTCRを0にクリアした場合、プログラムウェイト( $T_{C1}$ - $T_{C2}$ 間ウェイト)は挿入されません。バーストROM空間のバーストサイクルもプログラムウェイトは挿入されません。

基本バスインタフェースの各エリアのバス仕様を表6.3に示します。

表 6.3 各エリアのバス仕様 (基本バスインタフェース)

ABWCR	ASTCR	WCRH / WCRL		バス仕様 (基本バスインタフェース)		
ABWn	ASTn	Wn1	Wn0	バス幅	アクセス ステート数	プログラムウェイト ステート数
0	0			16	2	0
	1	0	0		3	0
			1		1	
		1	0		2	
			1		3	
1	0			8	2	0
	1	0	0		3	0
			1			1
		1	0			2
			1			3

【注】 n = 7 ~ 0

### 6.3.3 メモリインタフェース

本 LSI のメモリインタフェースは、ROM、SRAM などの直結が可能な基本バスインタフェース、DRAM の直結が可能な DRAM インタフェース、およびバースト ROM の直結が可能なバースト ROM インタフェースがあり、エリア単位で選択することができます。

基本バスインタフェースを設定したエリアが通常空間です。また、DRAM インタフェースを設定したエリアが DRAM 空間、バースト ROM インタフェースを設定したエリア 0 がバースト ROM 空間です。

### 6.3.4 チップセレクト信号

本 LSI は、エリア 0~7 に対してそれぞれチップセレクト信号 ( $\overline{CS}_0 \sim \overline{CS}_7$ ) を出力することができ、拡張モードで当該エリアが選択されたとき、Low レベルを出力します。

図 6.4 に  $\overline{CS}_n$  信号出力タイミングを示します。

#### (1) $\overline{CS}_0 \sim \overline{CS}_3$ 出力方法

$\overline{CS}_0 \sim \overline{CS}_3$  の出力の許可または禁止は各  $\overline{CS}_n$  端子に対応するポートのデータディレクションレジスタ (DDR) を設定することにより行います。

内蔵 ROM 無効拡張モード時、 $\overline{CS}_0$  端子はリセット直後に出力状態となっています。 $\overline{CS}_1 \sim \overline{CS}_3$  はリセット直後に入力状態となっていますので、 $\overline{CS}_1 \sim \overline{CS}_3$  を出力する場合には対応する DDR を 1 にセットしてください。内蔵 ROM 有効拡張モード時、 $\overline{CS}_0 \sim \overline{CS}_3$  端子はリセット直後に入力状態になっていますので、 $\overline{CS}_0 \sim \overline{CS}_3$  を出力する場合には、対応する DDR を 1 にセットしてください。

詳細は「第 8 章 I/O ポート」を参照してください。

#### (2) $\overline{CS}_4 \sim \overline{CS}_7$ 出力方法

$\overline{CS}_4 \sim \overline{CS}_7$  の出力許可または禁止はチップセレクトコントロールレジスタ (CSCR) を設定することにより行います。 $\overline{CS}_4 \sim \overline{CS}_7$  端子はリセット直後に入力状態となっていますので、 $\overline{CS}_4 \sim \overline{CS}_7$  端子を出力する場合には CSCR の対応するビットを 1 にセットしてください。詳細は「第 8 章 I/O ポート」を参照してください。

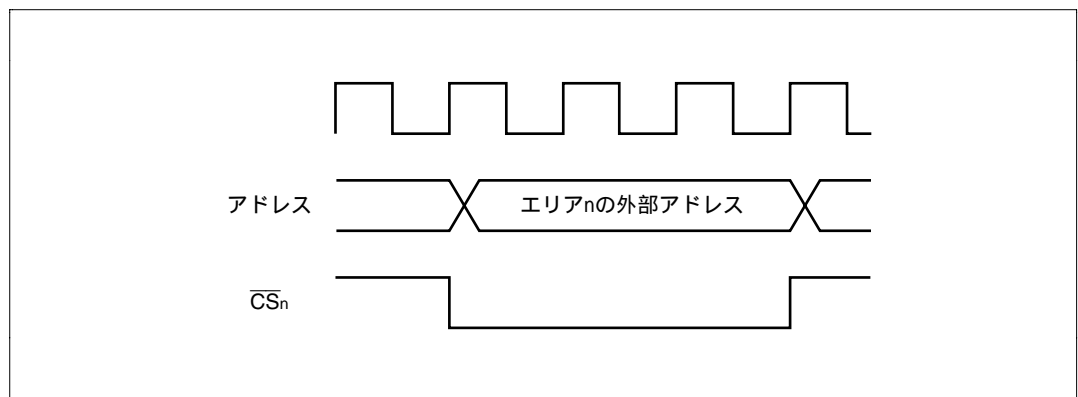


図 6.4  $\overline{CS}_n$  信号の出力タイミング (n=0~7)

内蔵 ROM、内蔵 RAM および内部 I/O レジスタを選択した場合、 $\overline{CS}_0 \sim \overline{CS}_7$  端子は High レベルです。 $\overline{CS}_n$  信号はアドレスをデコードした信号で、SRAM などのチップセレクト信号として使用することができます。

### 6.3.5 アドレス出力方式

本 LSI では、アドレス出力方式について、従来 H8/300H シリーズと同じ更新方式（アドレス更新モード 1）または、アドレス更新を外部空間アクセス時またはセルフリフレッシュサイクル時に限定する方式（アドレス更新モード 2）いずれかに設定することが可能です。

図 6.5 に 2 つの更新モードのアドレス出力例を示します。

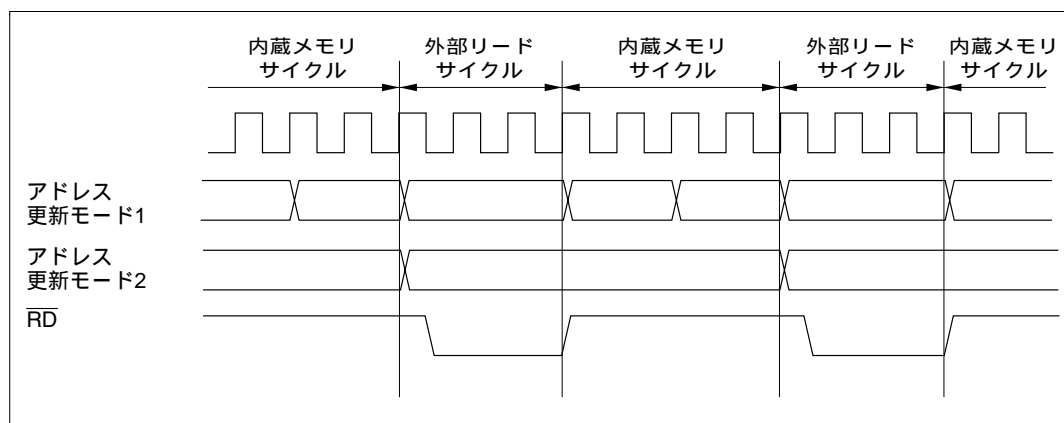


図 6.5 各アドレス更新モードにおけるアドレス出力例  
（基本バスインタフェース、3 ステート空間の場合）

#### (1) アドレス更新モード 1

アドレス更新モード 1 は、従来の H8/300H シリーズと互換性のあるアドレス更新モードです。アドレスは各バスサイクルの切れ目で常に更新されます。

#### (2) アドレス更新モード 2

アドレス更新モード 2 は、アドレスの更新が外部空間アクセスまたはセルフリフレッシュサイクル時に限定されます。

アドレス更新モード 2 では、プログラムを内蔵メモリに配置することにより、外部空間リードサイクルと命令フェッチサイクル（内蔵メモリ）の間でアドレスを保持することができます。従って  $\overline{RD}$  ストロープの立ち上がりに対して、アドレスのホールドを要求するデバイスを接続する場合に、アドレス更新モード 2 が有効です。

アドレス更新モード 1 とモード 2 の切り替えは、ADRCR の ADRCCTL ビットによって選択します。ADRCR の初期値はアドレス更新モード 1 となっており、従来 H8/300H シリーズと互換性があります。

(3) 注意事項

本アドレス更新モードを使用する際、以下の点に注意してください。

- ・ アドレス更新モード2を選択した場合、内部空間（内蔵メモリ、内部I/O）に対するアクセスサイクルのアドレスは外部に出力されません。
- ・  $\overline{RD}$  の立ち上がりに対するアドレスのホールド確保を目的として、アドレス更新モード2を使用する場合、外部空間リードアクセスは、必ず単一のアクセスサイクルで終了してください。たとえば、8ビットアクセス空間に対するワードアクセスは図6.6のようにバスサイクルが2回に分割され、単一のアクセスサイクルとはなりません。このようなケースでは、1回目のアクセスサイクル（偶数アドレス）と2回目のアクセスサイクル（奇数アドレス）の間の  $\overline{RD}$  の立ち上がりで、アドレスのホールドが保証されません。（図中の楕円で囲んだ部分）。

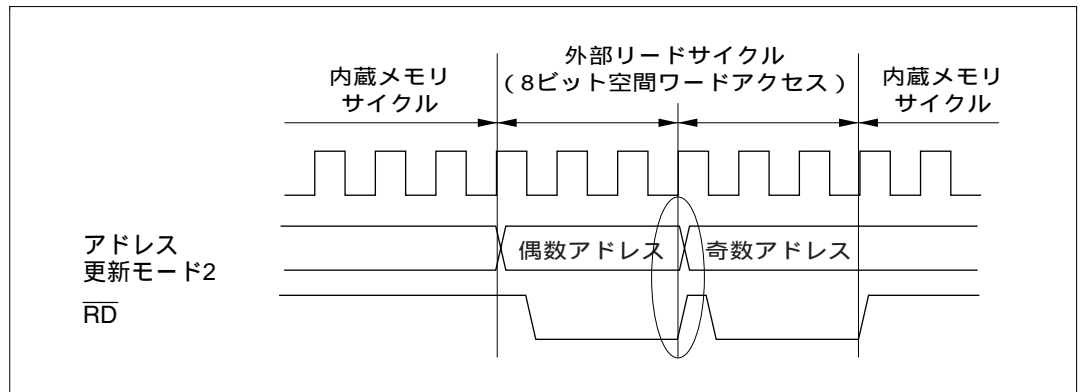


図 6.6 アドレス更新モード2での外部空間連続アクセス例

- ・ アドレス更新モード2を選択した場合、DRAM 空間に対する CAS ビフォ RAS (CBR) リフレッシュサイクル時には、直前のアドレスを保持します。（エリア2の先頭アドレスは出力されません。）

## 6.4 基本バスインタフェース

### 6.4.1 概要

基本バスインタフェースは、ROM、SRAMなどの直結が可能です。

ABWCR、ASTCR、WCRH、WCRLによってバス仕様を選択できます。表6.3を参照してください。

### 6.4.2 データサイズとデータアライメント

CPUおよびそのほかの内部バスマスタのデータサイズには、バイト、ワード、ロングワードがあります。バスコントローラは、データアライメント機能を持っており、外部空間をアクセスするとき、上位側データバス ( $D_{15} \sim D_8$ ) を使用するか、下位側データバス ( $D_7 \sim D_0$ ) を使用するかを、アクセスするエリアのバス仕様 (8ビットアクセス空間か16ビットアクセス空間) とデータサイズによって制御します。

#### (1) 8ビットアクセス空間

図6.7に、8ビットアクセス空間のデータアライメント制御を示します。8ビットアクセス空間では、常に上位側データバス ( $D_{15} \sim D_8$ ) を使ってアクセスを行います。一回にアクセスできるデータ量は1バイトで、ワードアクセスでは2回、ロングワードアクセスは4回のバイトアクセスを実行します。

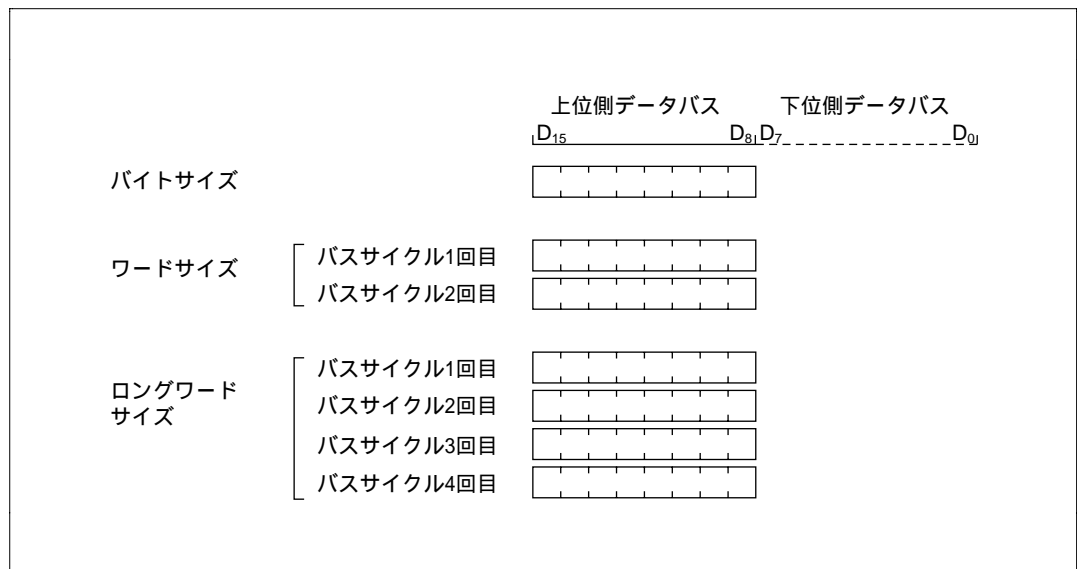


図6.7 アクセスサイズとデータアライメント制御 (8ビットアクセス空間)

(2) 16 ビットアクセス空間

図 6.8 に、16 ビットアクセス空間の、データアライメント制御を示します。16 ビットアクセス空間では、上位側データバス ( $D_{15} \sim D_8$ ) および下位側データバス ( $D_7 \sim D_0$ ) を使ってアクセスを行います。一回にアクセスできるデータ量は 1 バイトまたは 1 ワードで、ロングワードアクセスは、ワードアクセスを 2 回実行します。

バイトアクセスのとき、上位側データバスを使用するか、下位側データバスを使用するかは、アドレスの偶数 / 奇数で決まります。偶数アドレスに対するバイトアクセスは上位側データバスを使用し、奇数アドレスに対するバイトアクセスは下位側データバスを使用します。

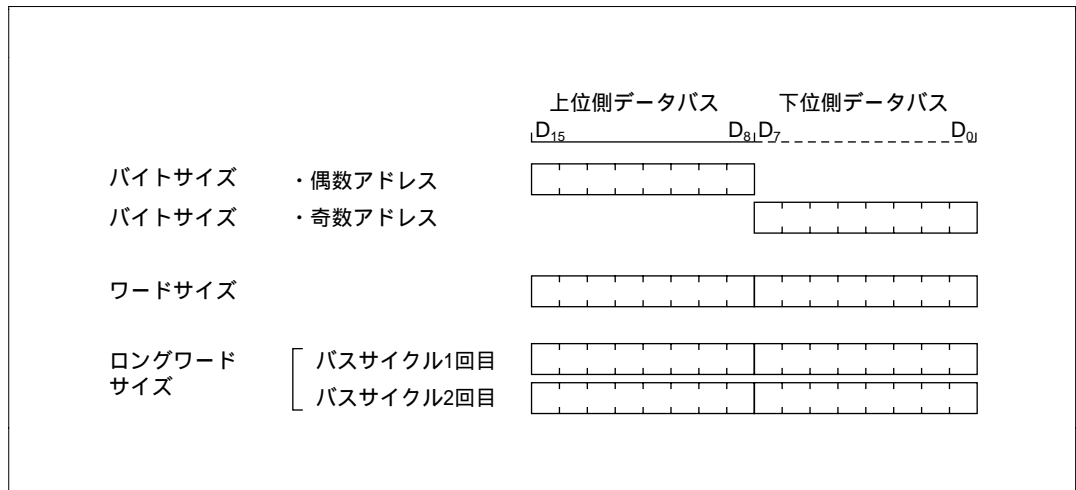


図 6.8 アクセスサイズとデータアライメント制御 (16 ビットアクセス空間)

### 6.4.3 有効ストロープ

表 6.4 にアクセス空間と、使用するデータバスおよび有効なストロープを示します。

リード時には、データバスの上位側、下位側の区別なく、 $\overline{RD}$  信号が有効です。

ライト時には、データバスの上位側に対して  $\overline{HWR}$  信号が、下位側に対して  $\overline{LWR}$  信号が有効です。

表 6.4 使用するデータバスと有効ストロープ

エリア	アクセス サイズ	リード/ ライト	アドレス	有効な ストロープ	データバス上位 ( $D_{15} \sim D_8$ )	データバス下位 ( $D_7 \sim D_0$ )
8ビット アクセス 空間	バイト	リード	-	$\overline{RD}$	有効	無効
		ライト	-	$\overline{HWR}$		不定
16ビット アクセス 空間	バイト	リード	偶数	$\overline{RD}$	有効	無効
			奇数		無効	有効
		ライト	偶数	$\overline{HWR}$	有効	不定
			奇数	$\overline{LWR}$	不定	有効
	ワード	リード	-	$\overline{RD}$	有効	有効
		ライト	-	$\overline{HWR}$ , $\overline{LWR}$	有効	有効

【注】 不定：不定データが出力されます。

無効：入力状態であり、入力値は無視されます。



#### 6.4.4 各エリアの説明

各エリアの初期状態は、基本バスインタフェース、かつ3ステートアクセス空間になっています。バス幅の初期状態は動作モードで選択します。ここで説明しているバス仕様に関しては、基本的なことについてのみ記載していますので、必ず「6.4 基本バスインタフェース」「6.5 DRAM インタフェース」「6.8 バースト ROM インタフェース」を確認してください。

##### (1) エリア 0

エリア 0 は内蔵 ROM を含んでおり、ROM 無効拡張モードでは、エリア 0 の全ての空間が外部空間となります。ROM 有効拡張モードでは、内蔵 ROM を除いた空間が外部空間となります。

エリア 0 の外部空間をアクセスするとき、 $\overline{CS}_0$  信号を出力することができます。

エリア 0 は、基本バスインタフェースまたはバースト ROM インタフェースを選択することができます。

エリア 0 の大きさはモード 1、2 のときは 128k バイト、モード 3、4、5 のときは 2M バイトとなります。

##### (2) エリア 1、6

エリア 1、6 は、外部拡張モードのとき、エリアの全ての空間が外部空間となります。

エリア 1、6 の外部空間をアクセスするとき、それぞれ、 $\overline{CS}_1$ 、 $\overline{CS}_6$  端子信号を出力することができます。

エリア 1、6 は、基本バスインタフェースのみを使用することができます。

エリア 1、6 の大きさはモード 1、2 のときは 128k バイト、モード 3、4、5 のときは 2M バイトとなります。

##### (3) エリア 2~5

エリア 2~5 は外部拡張モードのとき、エリア 2~5 の全ての空間が外部空間となります。

エリア 2~5 の外部空間をアクセスするとき、 $\overline{CS}_2 \sim \overline{CS}_5$  信号を出力することができます。

エリア 2~5 は、基本バスインタフェース、DRAM インタフェースを選択することができます。DRAM インタフェースでは、 $\overline{CS}_2 \sim \overline{CS}_5$  信号は  $\overline{RAS}$  信号として使用されます。

エリア 2~5 の大きさはモード 1、2 のときは 128k バイト、モード 3、4、5 のときは 2M バイトとなります。

#### (4) エリア7

エリア7は内蔵RAMおよび内部I/Oレジスタを含んでおり、外部拡張モードのとき、内蔵RAMおよび内蔵I/Oレジスタ空間を除いた空間が外部空間となります。なお、内蔵RAMはシステムコントロールレジスタ(SYSCR)のRAMEビットを1にセットしたとき有効で、RAMEビットを0にクリアすると、内蔵RAMは無効になり、対応するアドレスは外部空間になります。

エリア7の外部空間をアクセスするとき、 $\overline{CS}_7$ 信号を出力することができます。

エリア7のメモリインタフェースには、基本バスインタフェースのみを使用することができます。

エリア7の大きさはモード1、2のときは128kバイト、モード3、4、5のときは2Mバイトとなります。

## 6.4.5 基本バス制御信号タイミング

## (1) 8 ビット 3 ステートアクセス空間

図 6.9 に 8 ビット 3 ステートアクセス空間のバス制御信号タイミングを示します。8 ビットアクセス空間をアクセスする場合、データバスの上位側 ( $D_{15} \sim D_8$ ) を使用します。

$\overline{LWR}$  端子は常に High レベルとなっています。ウェイトステートを挿入することができます。

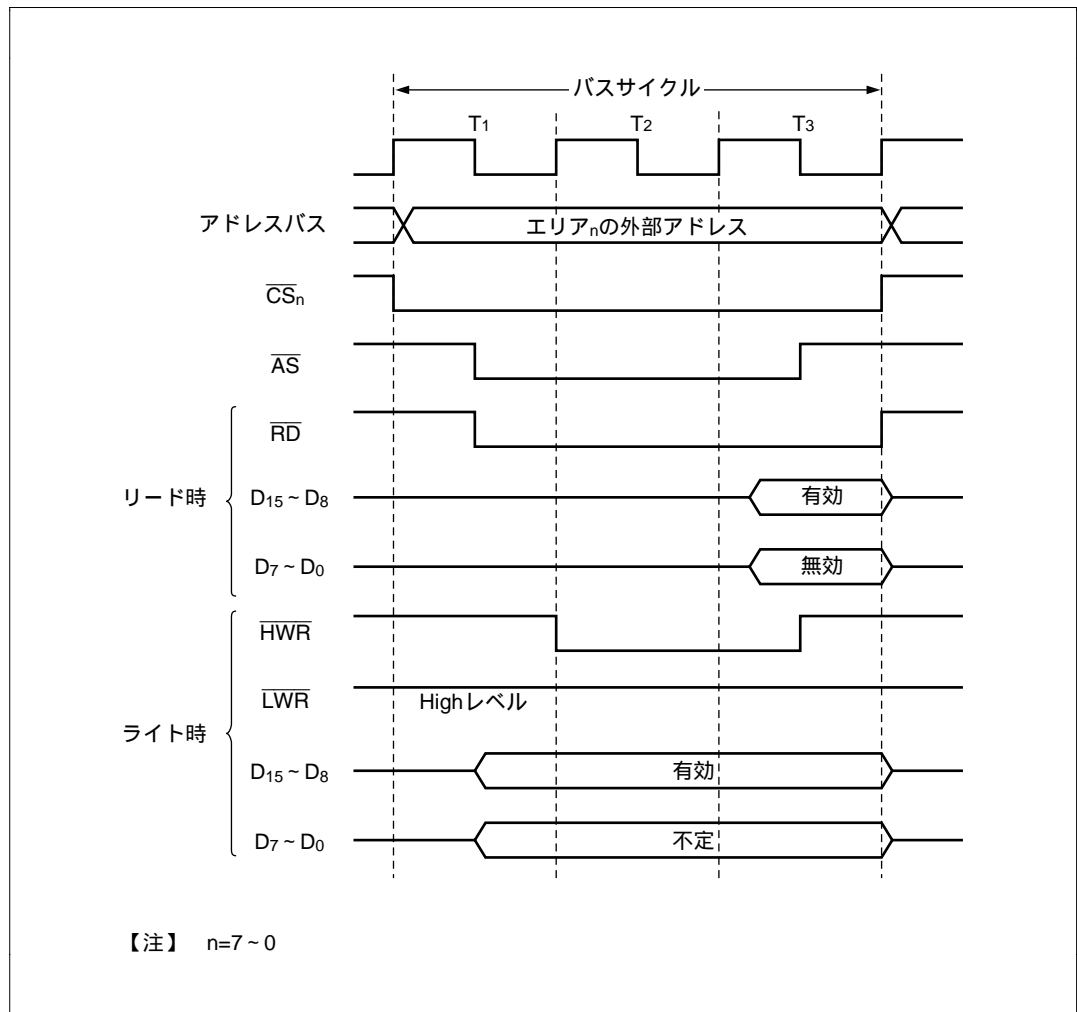


図 6.9 8 ビット 3 ステートアクセス空間のバス制御信号タイミング

## (2) 8 ビット 2 ステートアクセス空間

図 6.10 に 8 ビット 2 ステート空間のバス制御信号タイミングを示します。8 ビットアクセス空間をアクセスする場合、データバスの上位側 ( $D_{15} \sim D_8$ ) を使用します。

$\overline{LWR}$  端子は常に High レベルとなっています。ウェイトステートを挿入することはできません。

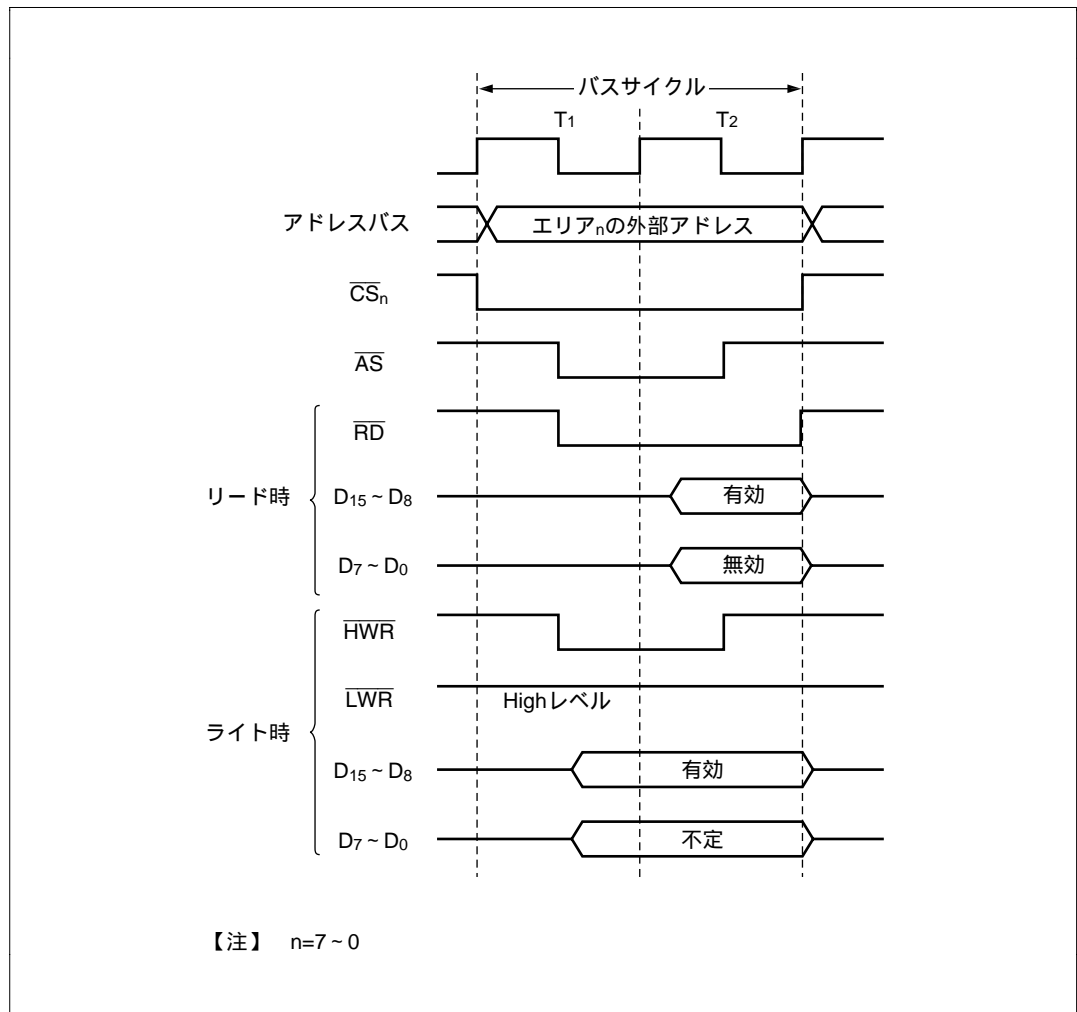


図 6.10 8 ビット 2 ステートアクセス空間のバス制御信号タイミング

(3) 16 ビット 3 ステートアクセス空間

図 6.11 ~ 図 6.13 に 16 ビット 3 ステートアクセス空間のバス制御信号タイミングを示します。16 ビットアクセス空間をアクセスする場合、偶数アドレスに対してはデータバスの上位側 ( $D_{15} \sim D_8$ ) を使用し、奇数アドレスに対してはデータバスの下位側 ( $D_7 \sim D_0$ ) を使用します。ウェイトステートを挿入することができます。

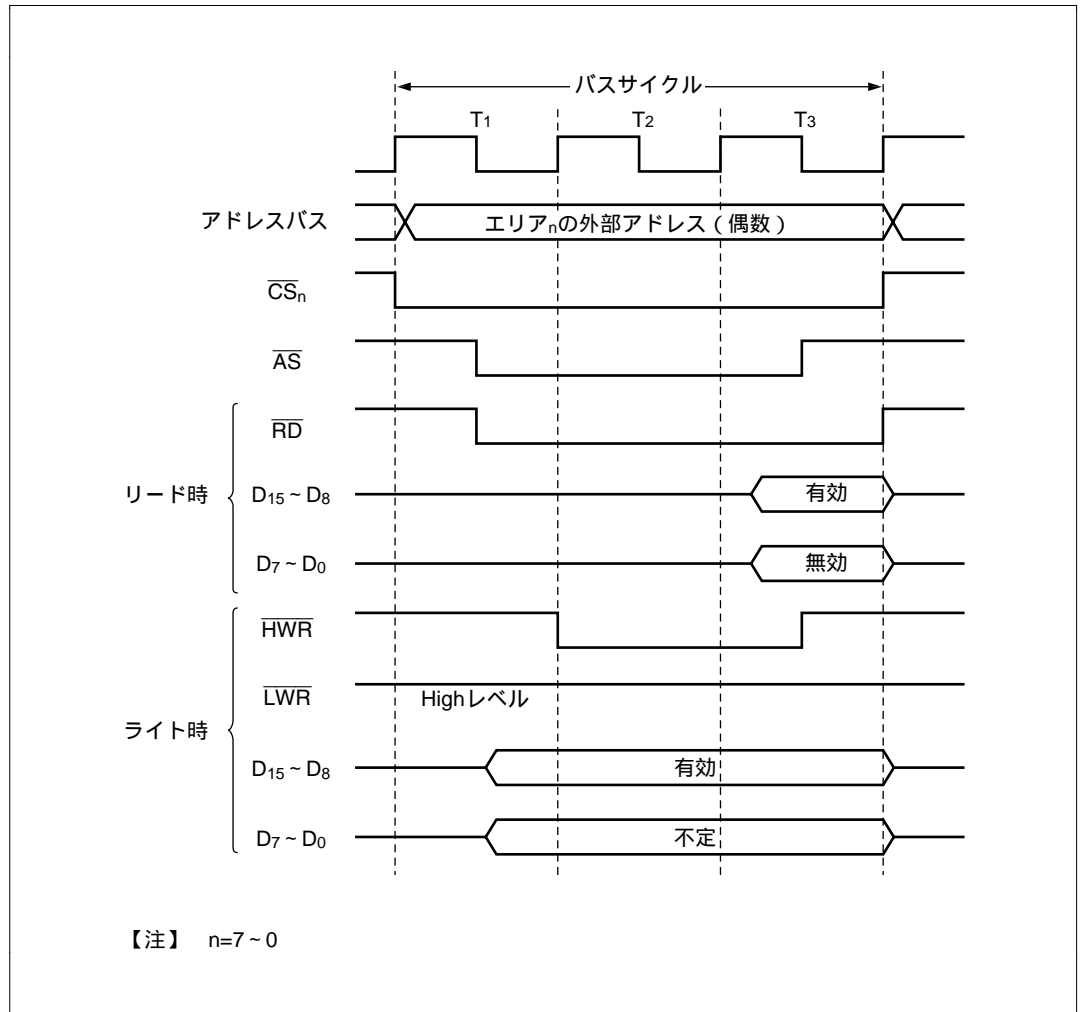


図 6.11 16 ビット 3 ステートアクセス空間のバス制御信号タイミング  
(偶数アドレスバイトアクセス)

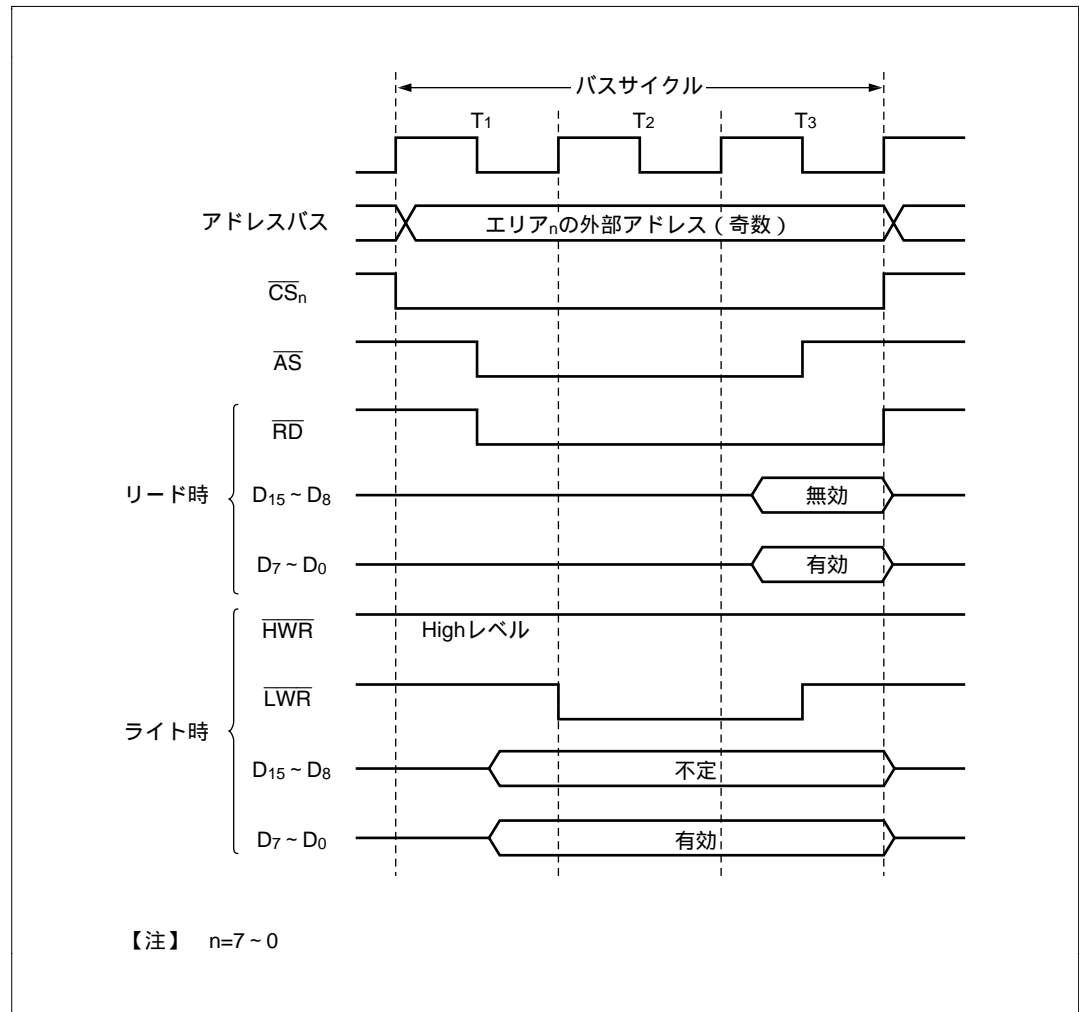


図 6.12 16ビット3ステートアクセス空間のバス制御信号タイミング  
(奇数アドレスバイトアクセス)

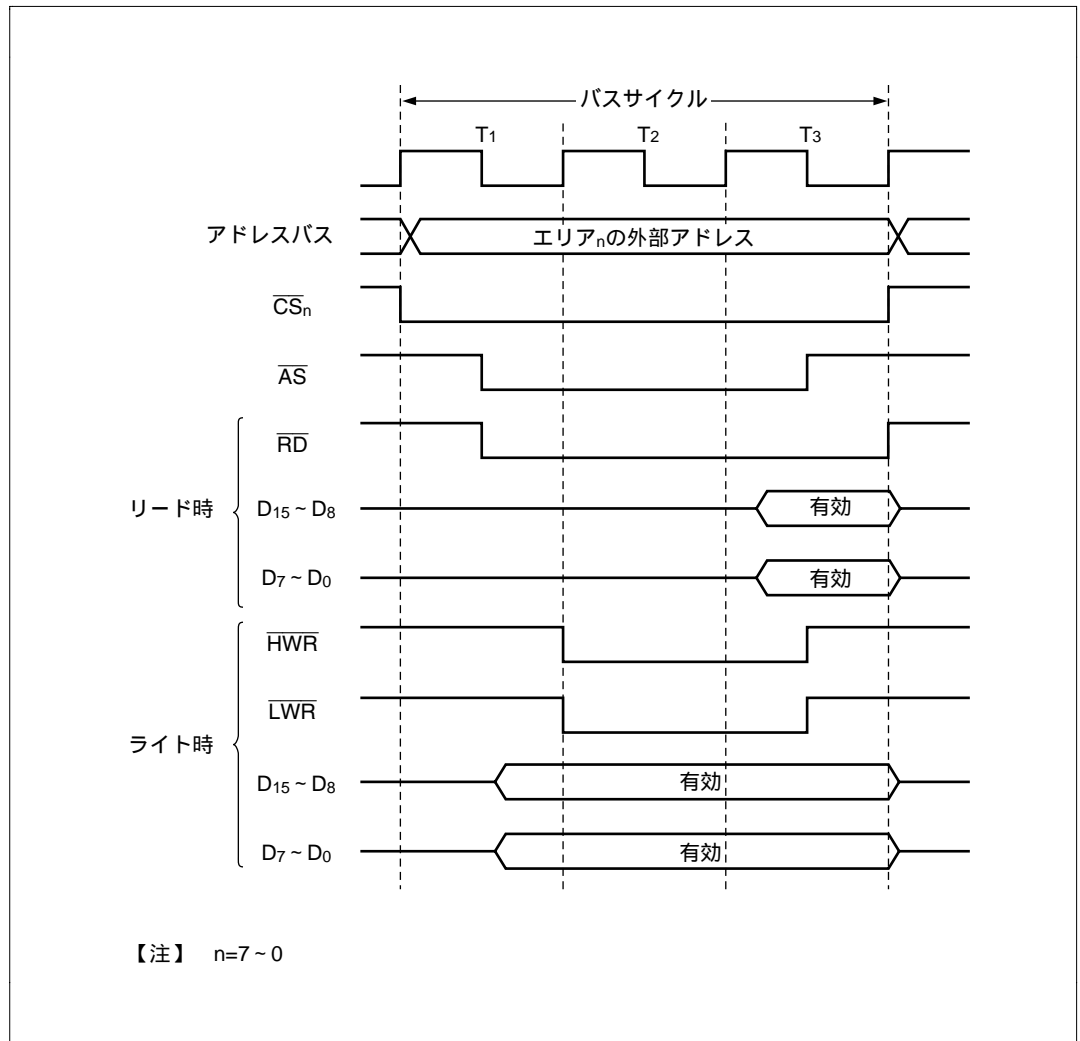


図 6.13 16ビット3ステートアクセス空間のバス制御信号タイミング  
(ワードアクセス)

## (4) 16 ビット 2 ステートアクセス空間

図 6.14 ~ 図 6.16 に 16 ビット 2 ステートアクセス空間のバス制御信号タイミングを示します。16 ビットアクセス空間をアクセスする場合、偶数アドレスに対してはデータバスの上位側 ( $D_{15} \sim D_8$ ) を使用し、奇数アドレスに対してはデータバスの下位側 ( $D_7 \sim D_0$ ) を使用します。ウェイトステートを挿入することはできません。

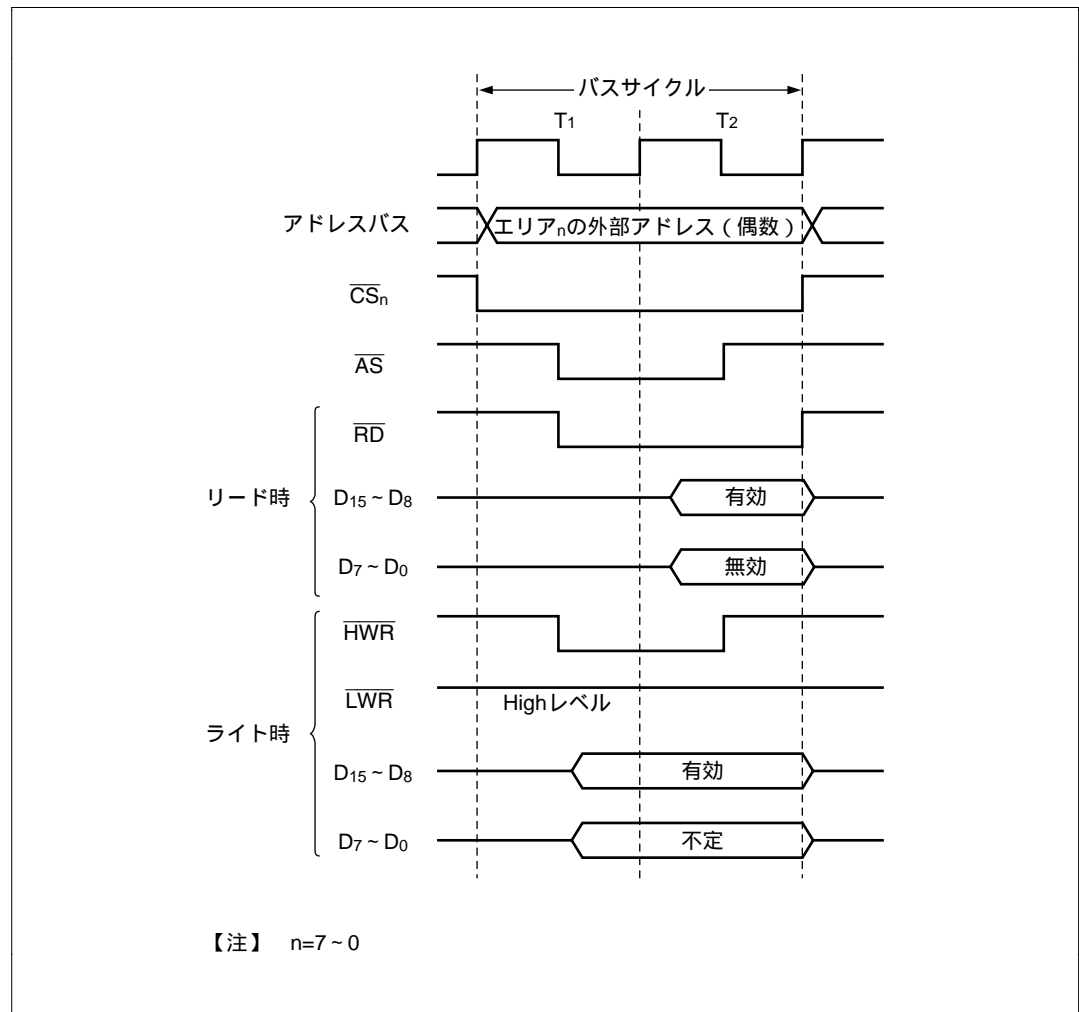


図 6.14 16 ビット 2 ステートアクセス空間のバス制御信号タイミング  
(偶数アドレスバイトアクセス)



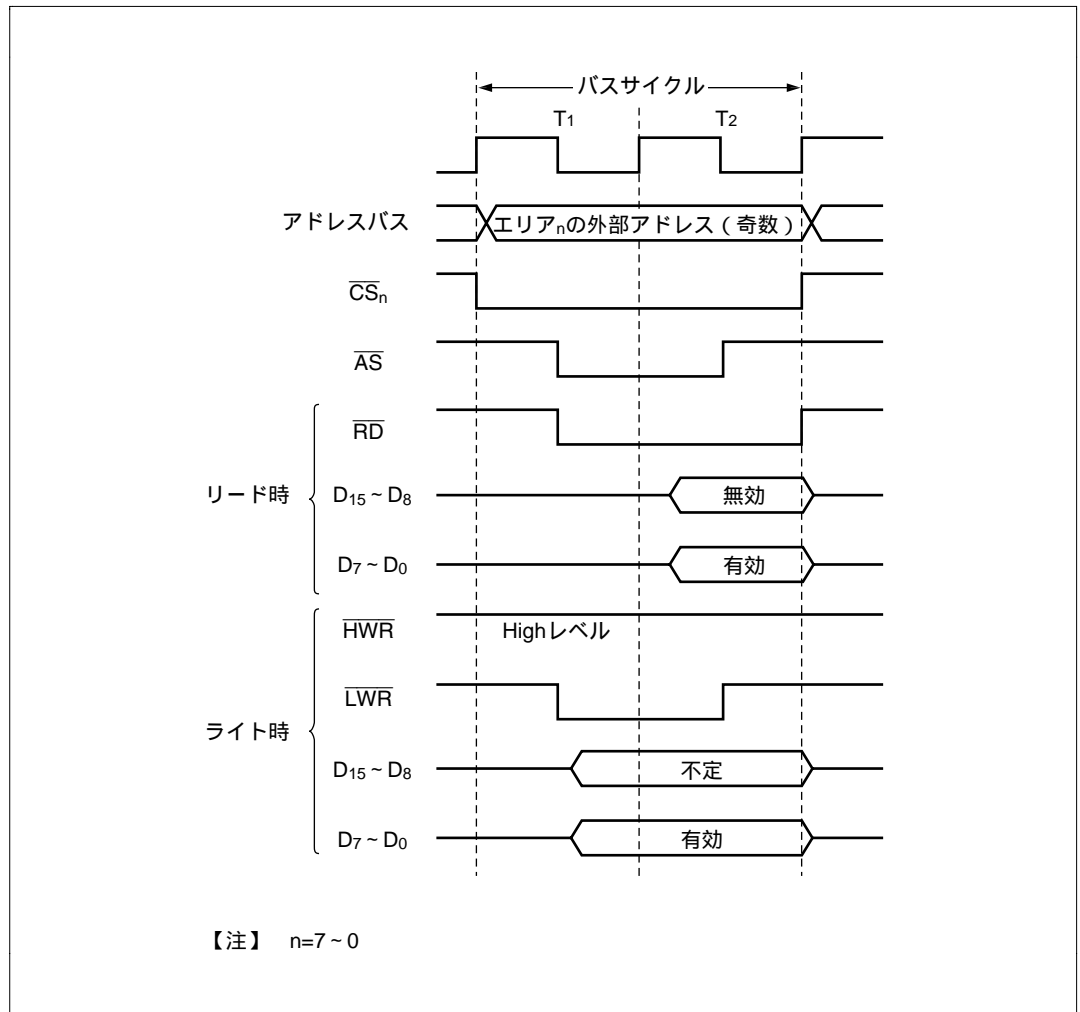


図 6.15 16ビット2ステートアクセス空間のバス制御信号タイミング  
(奇数アドレスバイトアクセス)

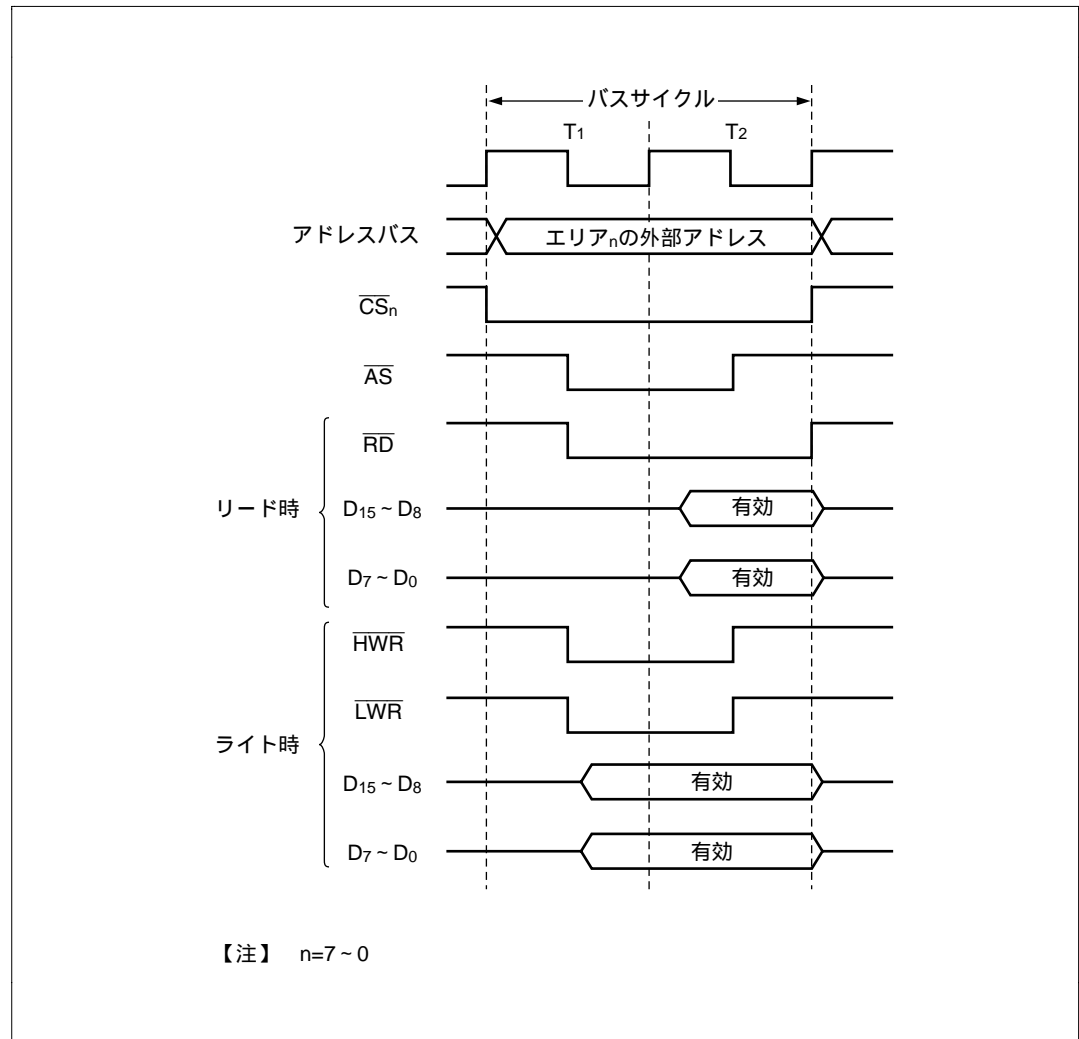


図 6.16 16ビット2ステータアクセス空間のバス制御信号タイミング  
(ワードアクセス)

### 6.4.6 ウェイト制御

本 LSI は、外部空間をアクセスするとき、ウェイトステート ( $T_w$ ) を挿入してバスサイクルを引き伸ばすことができます。ウェイトステートを挿入する方法には、(1) プログラムウェイトの挿入、(2)  $\overline{\text{WAIT}}$  端子による端子ウェイトの挿入、があります。

#### (1) プログラムウェイトの挿入

WCRH、WCRL の設定により、3 ステートアクセス空間に対して、エリア単位で 0~3 ステートのウェイトステートを自動的に  $T_2$  ステートと  $T_3$  ステートの間に挿入することができます。

#### (2) 端子ウェイトの挿入

BKR の WAITE ビットを 1 にセットすると、 $\overline{\text{WAIT}}$  端子によるウェイト入力が有効になります。この状態で、外部空間をアクセスすると、まずプログラムウェイトが挿入されます。 $T_2$  または  $T_w$  の最後のステートの の立ち下がりのタイミングで、 $\overline{\text{WAIT}}$  端子が Low レベルであると、更に  $T_w$  が挿入されます。 $\overline{\text{WAIT}}$  端子が Low レベルに保持されると、 $\overline{\text{WAIT}}$  端子が High レベルになるまで  $T_w$  が挿入されます。

4 ステート以上の  $T_w$  を挿入する場合や、外部デバイス毎に挿入する  $T_w$  数を変える場合などに有効です。

WAITE ビットは全てのエリアに対して共通です。

なお、DRAM 空間には端子ウェイトは挿入されません。

図 6.17 に 3 ステート空間プログラムウェイト 1 ステートのウェイトステート挿入のタイミング例を示します。

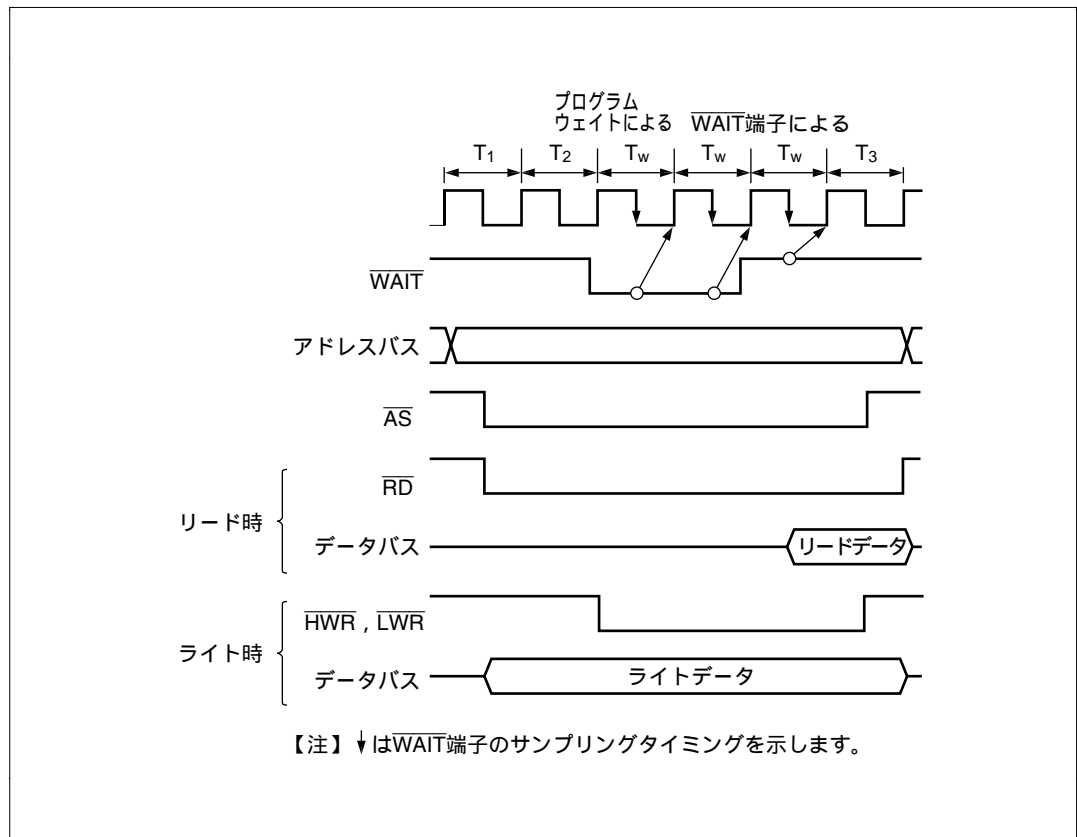


図 6.17 ウェイトステート挿入タイミング例

## 6.5 DRAM インタフェース

### 6.5.1 概要

本 LSI は、DRAM 制御信号 ( $\overline{RAS}$ 、 $\overline{UCAS}$ 、 $\overline{LCAS}$ 、 $\overline{WE}$ ) の出力機能、アドレスマルチプレクス機能、およびリフレッシュ機能をもった DRAM インタフェースを備えることにより DRAM を直結することができます。拡張モードのとき、エリア 2~5 の外部アドレス空間を、DRAM インタフェースが機能する DRAM 空間として割り付けることができます。DRAM 空間のデータバス幅は ABWCR の設定により 8 ビットと 16 ビットから選択できます。データバス幅として 16 ビットを選択した場合のバイトアクセス制御は CAS を用います。したがって ×16 ビット構成の DRAM では 2CAS 方式のものが接続可能となります。アクセスモードとしては通常のリード、ライトアクセスに加え、高速ページモードをサポートします。

### 6.5.2 DRAM 空間と $\overline{RAS}$ 出力端子の設定

エリア 2~5 に対する DRAM 空間の割付けと DRAM 空間に設定された各エリアに対応する  $\overline{RAS}$  出力端子の選択は、DRCRA の DRAS2~DRAS0 ビットを設定することにより行います。DRAS2~DRAS0 ビットの設定値と選択される DRAM 空間および  $\overline{RAS}$  出力端子の関係を表 6.5 に示します。なお、DRAS2~DRAS0 に任意の値が設定されている場合、000 以外の異なる値のライトは行わないでください。

表 6.5 DRAS2~DRAS0 の設定値と DRAM 空間 ( $\overline{RAS}$  出力端子) の関係

DRAS2	DRAS1	DRAS0	エリア 5	エリア 4	エリア 3	エリア 2
0	0	0	通常空間	通常空間	通常空間	通常空間
		1	通常空間	通常空間	通常空間	DRAM 空間 ( $\overline{CS}_2$ )
	1	0	通常空間	通常空間	DRAM 空間 ( $\overline{CS}_3$ )	DRAM 空間 ( $\overline{CS}_2$ )
		1	通常空間	通常空間	DRAM 空間 ( $\overline{CS}_2$ ) *	
1	0	0	通常空間	DRAM 空間 ( $\overline{CS}_4$ )	DRAM 空間 ( $\overline{CS}_3$ )	DRAM 空間 ( $\overline{CS}_2$ )
		1	DRAM 空間 ( $\overline{CS}_5$ )	DRAM 空間 ( $\overline{CS}_4$ )	DRAM 空間 ( $\overline{CS}_3$ )	DRAM 空間 ( $\overline{CS}_2$ )
	1	0	DRAM 空間 ( $\overline{CS}_4$ ) *		DRAM 空間 ( $\overline{CS}_2$ ) *	
		1	DRAM 空間 ( $\overline{CS}_2$ ) *			

【注】 \* 一本の  $\overline{CS}_n$  端子が複数エリアに共通の  $\overline{RAS}$  出力端子となります。このとき不要となる  $\overline{CS}_n$  端子は入出力ポートとして使用可能となります。

### 6.5.3 アドレスマルチプレクス

DRAM空間をアクセスする場合、ロウアドレスとカラムアドレスがマルチプレクスされます。アドレスマルチプレクスの方法は、DRAMのカラムアドレスのビット数に合せDRCRBのMXC1、MXC0ビットにより選択します。表6.6にMXC1、MXC0の設定値とアドレスマルチプレクスの関係を示します。

表6.6 MXC1、MXC0とアドレスマルチプレクスの関係

	DRCRB		カラム アドレス	アドレス端子													
	MXC1	MXC0	ビット数	A <sub>23</sub> ~ A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>	A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>
ロウアドレス	0	0	8ビット	A <sub>23</sub> ~ A <sub>13</sub>	A <sub>20</sub> *	A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>
		1	9ビット	A <sub>23</sub> ~ A <sub>13</sub>	A <sub>12</sub>	A <sub>20</sub> *	A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>
	1	0	10ビット	A <sub>23</sub> ~ A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>20</sub> *	A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>
		1	設定禁止	-	-	-	-	-	-	-	-	-	-	-	-	-	-
カラムアドレス	-	-	-	A <sub>23</sub> ~ A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>	A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>

【注】 \* 動作モードが1Mバイトモードの時、ロウアドレスのA<sub>20</sub>はマルチプレクスされません。

### 6.5.4 データバス

DRAM 空間に設定したエリアに対応する ABWCR のビットを 1 にセットすると、当該エリアは 8 ビット DRAM 空間となり、0 にクリアすると 16 ビット DRAM 空間となります。16 ビット DRAM 空間では ×16 ビット構成の DRAM を直結することができます。

8 ビット DRAM 空間では  $D_{15} \sim D_8$  の上位側データバスが有効となり、16 ビット DRAM 空間では  $D_{15} \sim D_0$  の上位側、下位側データバスが有効になります。

アクセスサイズとデータアライメントは基本バスインタフェースと同様ですので、「6.4.2 データサイズとデータアライメント」を参照してください。

### 6.5.5 DRAM インタフェース使用端子

表 6.7 に DRAM インタフェースで使用する端子と機能を示します。

表 6.7 DRAM インタフェース端子構成

端子	DRAM 設定時	名称	入出力	機能
PB4	$\overline{UCAS}$	アッパーカラムアドレスストロープ	出力	DRAM 空間アクセス時のアッパーカラムアドレスストロープ (DRCRB の CSEL=0 のとき)
PB5	$\overline{LCAS}$	ロウアーカラムアドレスストロープ	出力	DRAM 空間アクセス時のロウアーカラムアドレスストロープ (DRCRB の CSEL=0 のとき)
$\overline{HWR}$	$\overline{UCAS}$	アッパーカラムアドレスストロープ	出力	DRAM 空間アクセス時のアッパーカラムアドレスストロープ (DRCRB の CSEL=1 のとき)
$\overline{LWR}$	$\overline{LCAS}$	ロウアーカラムアドレスストロープ	出力	DRAM 空間アクセス時のロウアーカラムアドレスストロープ (DRCRB の CSEL=1 のとき)
$\overline{CS}_2$	$\overline{RAS}_2$	ロウアドレスストロープ 2	出力	DRAM 空間アクセス時のロウアドレスストロープ
$\overline{CS}_3$	$\overline{RAS}_3$	ロウアドレスストロープ 3	出力	DRAM 空間アクセス時のロウアドレスストロープ
$\overline{CS}_4$	$\overline{RAS}_4$	ロウアドレスストロープ 4	出力	DRAM 空間アクセス時のロウアドレスストロープ
$\overline{CS}_5$	$\overline{RAS}_5$	ロウアドレスストロープ 5	出力	DRAM 空間アクセス時のロウアドレスストロープ
$\overline{RD}$	$\overline{WE}$	ライトイネーブル	出力	DRAM 空間ライトアクセス時のライトイネーブル*
P80	$\overline{RFSH}$	リフレッシュ	出力	リフレッシュサイクル時 Low レベルとなります
$A_{12} \sim A_0$	$A_{12} \sim A_0$	アドレス	出力	ロウアドレス / カラムアドレスのマルチプレクス出力
$D_{15} \sim D_0$	$D_{15} \sim D_0$	データ	入出力	データ入出力端子

【注】 \* リードアクセス時は High レベル固定となります。

### 6.5.6 基本タイミング

DRAM空間の基本アクセスタイミングを図6.18に示します。DRAMの基本タイミングは4状態で、プリチャージサイクル( $T_p$ )1ステート、ロウアドレス出力サイクル( $T_r$ )1ステート、カラムアドレス出力サイクル( $T_{c1}$ 、 $T_{c2}$ )2ステートで構成されています。基本バスインタフェースとは異なり、ASTCRの対応するビットは $T_{c1}$ - $T_{c2}$ 間ウェイトステートの挿入の許可または禁止のみを制御し、アクセスステート数に影響は与えません。ASTCRの当該ビットを0にクリアしてある場合、DRAMアクセスサイクルの $T_{c1}$ - $T_{c2}$ 間ウェイトステートの挿入を行うことができません。

なお、 $\overline{UCAS}$ 、 $\overline{LCAS}$ の出力端子として $\overline{HWR}$ 、 $\overline{LWR}$ を選択した場合、DRAMのリード/ライトサイクルに続きDRAM空間以外の外部エリアに対するアクセスサイクルが連続すると、DRAMアクセスサイクルの直後に無条件にアイドルサイクル( $T_i$ )が挿入されます。詳細は「6.9 アイドルサイクル」を参照してください。

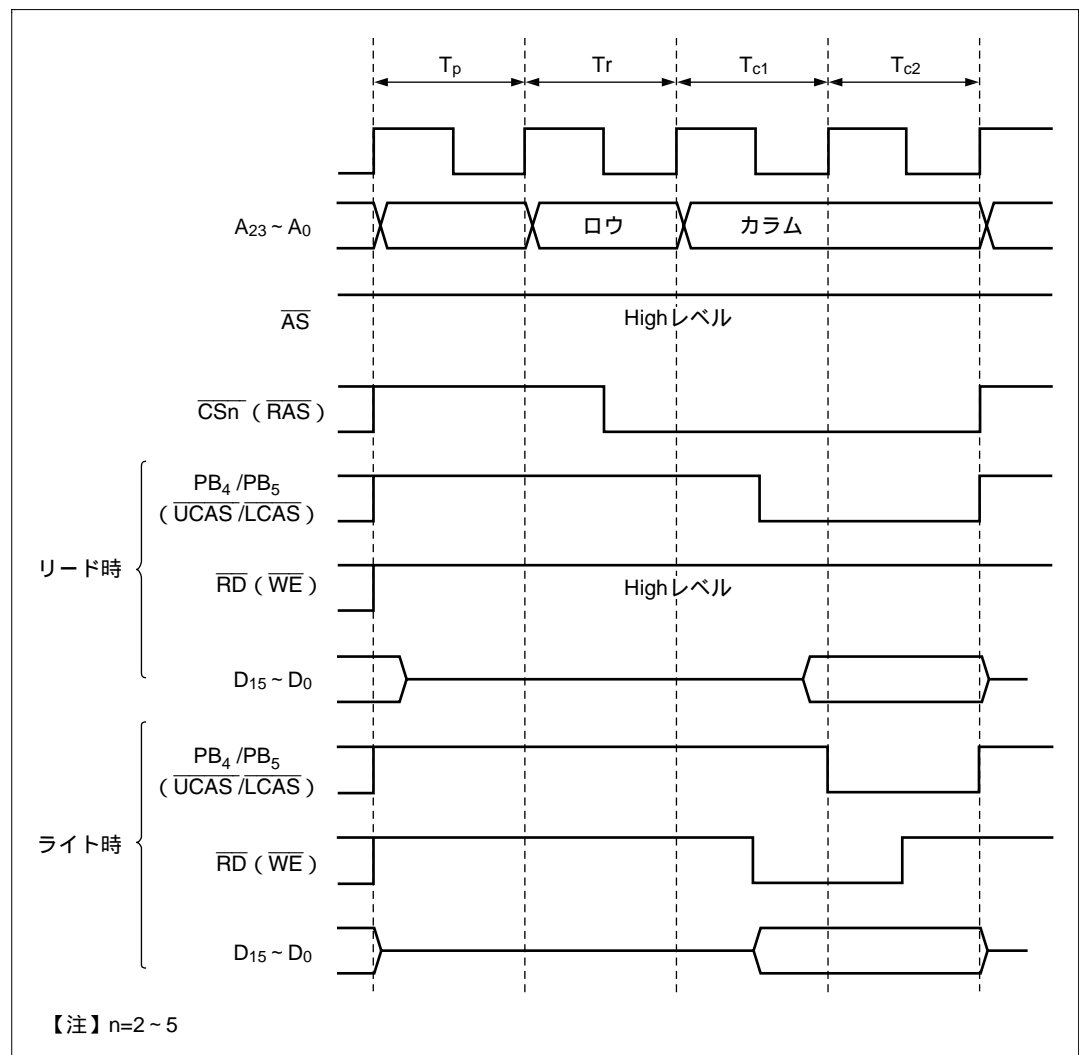


図 6.18 基本アクセスタイミング (DRCRB の CSEL=0 の場合)



## 6.5.7 プリチャージステート制御

本LSIでは、DRAMのRASプリチャージ時間を考慮し、DRAM空間をアクセスするとき、RASプリチャージステート ( $T_p$ ) を必ず1ステート挿入します。さらに、DRCRBのTPCビットを1にセットすることにより、 $T_p$ を1ステートから2ステートへ変更することができます。接続するDRAMと本LSIの動作周波数に応じて最適な $T_p$ サイクル数を設定してください。図6.19に $T_p$ を2ステートとしたときのタイミングを示します。

TPCビットを1にセットすると、CASビフォアRASリフレッシュサイクルの $T_p$ も2ステートとなります。

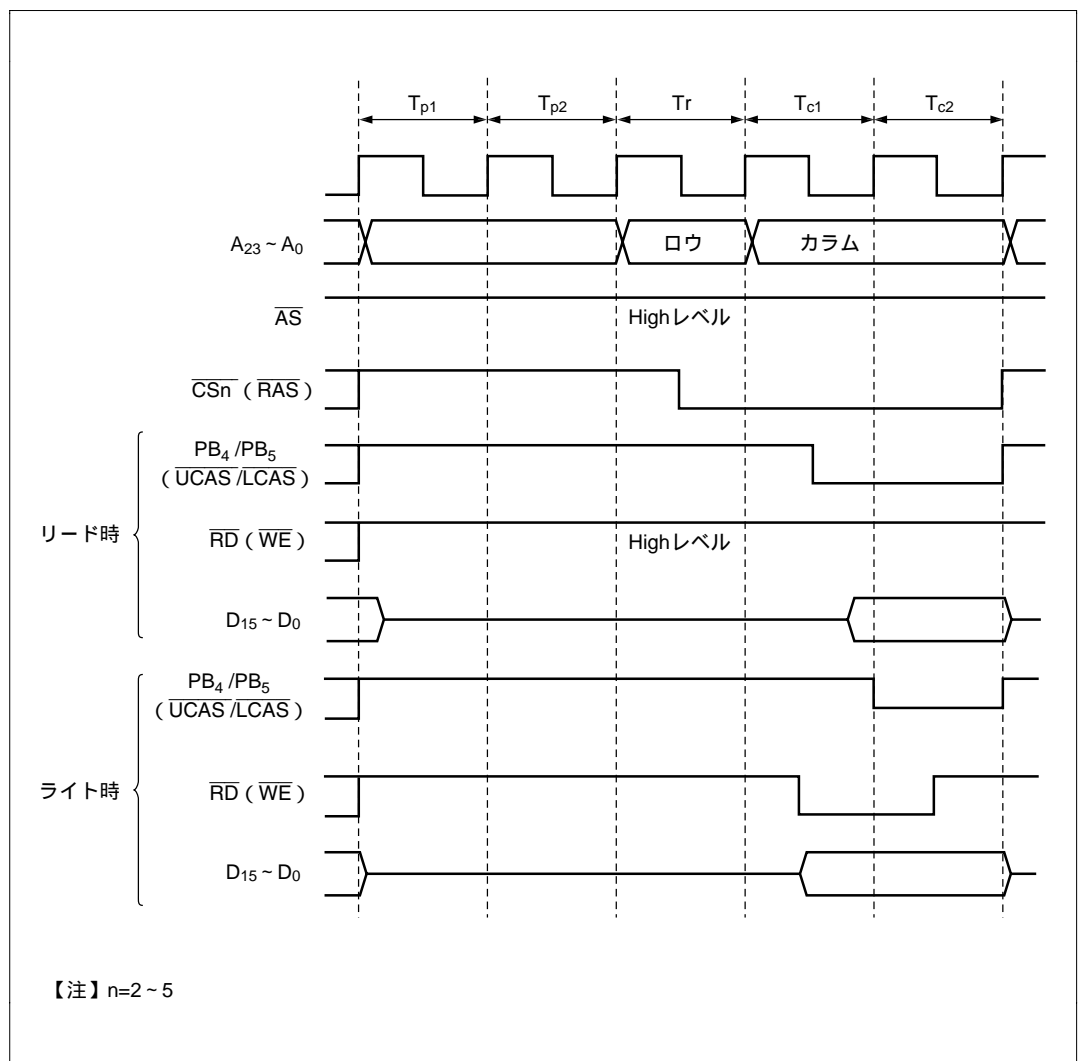


図 6.19 プリチャージサイクル2ステート時のタイミング  
(DRCRBのCSEL=0の場合)

## 6.5.8 ウェイト制御

DRAM アクセスサイクルの (1)  $T_r$  ステートと  $T_{c1}$  ステートの間および (2)  $T_{c1}$  ステートと  $T_{c2}$  ステートの間にウェイトステートを挿入することができます。

### (1) $T_r - T_{c1}$ 間ウェイトステート ( $T_{rw}$ ) の挿入

DRCRB の RCW ビットを 1 に設定することにより  $T_{rw}$  を  $T_r$  ステートと  $T_{c1}$  ステートの間に 1 ステート挿入することができます。

### (2) $T_{c1} - T_{c2}$ 間ウェイトステート ( $T_w$ ) の挿入

DRAM 空間に設定されたエリアに対応する ASTCR のビットが 1 にセットされているとき、WCRH、WCRL の設定により、0~3 ステートの  $T_w$  を、 $T_{c1}$  ステートと  $T_{c2}$  ステートの間に挿入することができます。

図 6.20 にウェイトステート挿入のタイミング例を示します。

なお、DRCRB の RCW ビットの設定および ASTCR、WCRH、WCRL の設定はリフレッシュサイクルに対し影響を与えません。また、DRAM 空間のアクセスサイクルには  $\overline{\text{WAIT}}$  端子によるウェイトステートの挿入はできません。

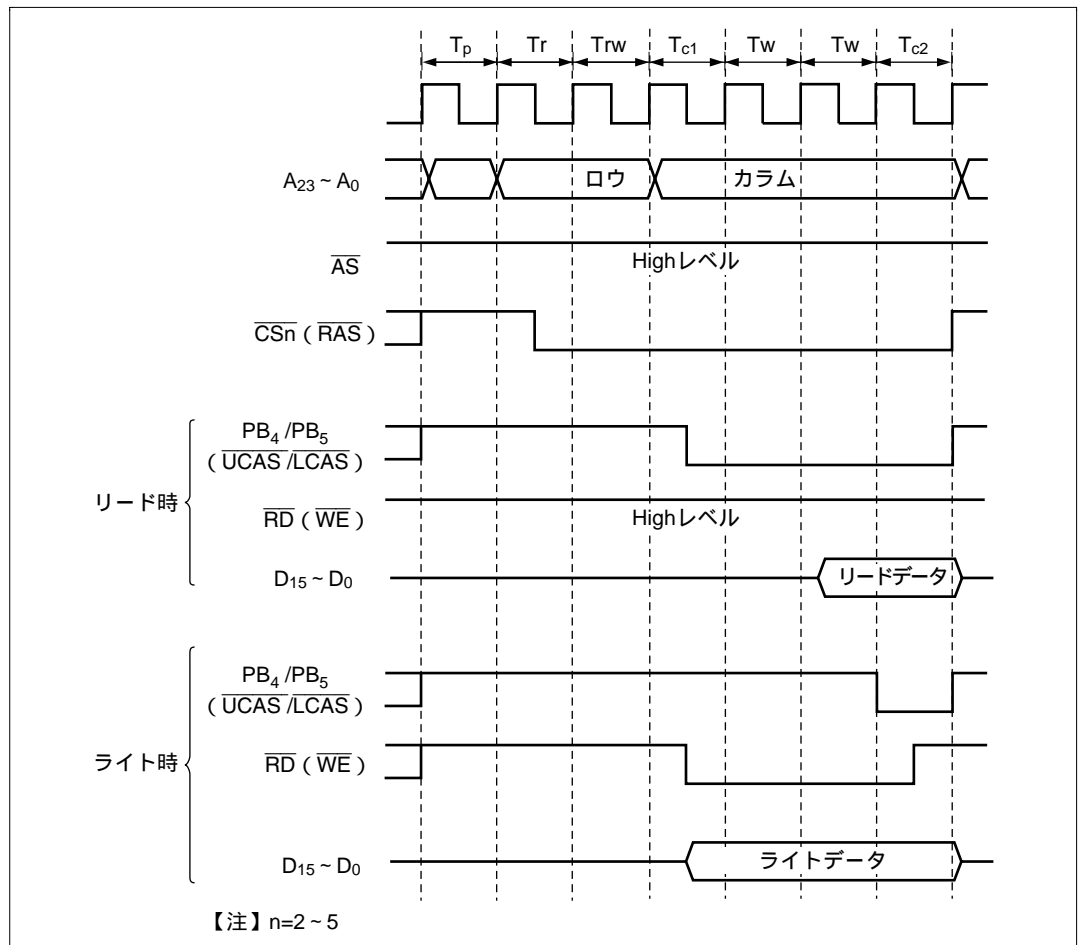


図 6.20 ウェイトステート挿入タイミング例 (CSEL=0 の場合)

### 6.5.9 バイトアクセス制御と $\overline{\text{CAS}}$ 出力端子

ABWCR により 16 ビットアクセス空間に指定された DRAM 空間をアクセスする場合、外部データバスの上位側、下位側にそれぞれ対応するカラムアドレスストロープ ( $\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$ ) を出力します。×16 ビット構成の DRAM では 2CAS 方式のものが接続できます。

$\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$  の出力端子として PB4、PB5 および  $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$  があり DRCRB の CSEL ビットの設定により選択することができます。表 6.8 に CSEL の設定値と  $\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$  の出力端子の関係を示します。

ABWCR により 8 ビットアクセス空間に設定された DRAM 空間をアクセスする場合、 $\overline{\text{UCAS}}$  のみ出力します。全ての DRAM 空間が 8 ビットアクセス空間に設定されており、CSEL=0 の場合、PB5 は入出力ポートとして使用することができます。

なお、外部空間に DRAM 以外のデバイスを接続し、 $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$  をライトストロープとして使用する場合、 $\overline{\text{RAS}}$  ダウンモードは使用できませんので注意してください。また、この場合 DRAM 空間アクセス後に DRAM 空間以外の外部アクセスが発生すると、必ずアイドルサイクル ( $T_i$ ) が挿入されます。詳細は「6.9 アイドルサイクル」を参照してください。

表 6.8 CSEL の設定値と  $\overline{\text{UCAS}}$ 、 $\overline{\text{LCAS}}$  の出力端子の関係

CSEL	$\overline{\text{UCAS}}$	$\overline{\text{LCAS}}$
0	PB4	PB5
1	$\overline{\text{HWR}}$	$\overline{\text{LWR}}$

図 6.21 に制御タイミングを示します。

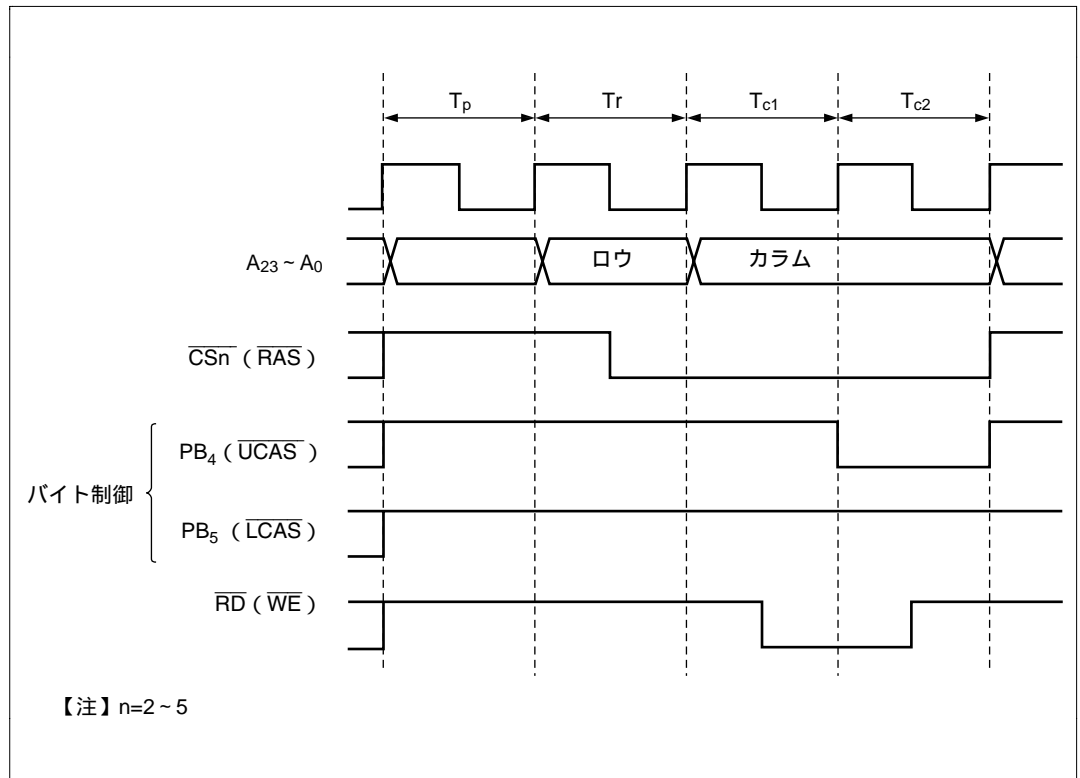


図 6.21 制御タイミング (CSEL=0 の場合の上位バイトライトアクセス時)

### 6.5.10 バースト動作

DRAM には、アクセスのたびにロウアドレスを出力してデータをアクセスするフルアクセス（ノーマルアクセス）の他に、同一のロウアドレスに対するアクセスが連続するとき、ロウアドレスを出力した後、カラムアドレスを変更するだけでデータを高速にアクセス（バーストアクセス）できる高速ページモードを備えているものがあります。DRCRA の BE ビットを 1 にセットすることにより、バーストアクセスを選択することができます。

#### (1) バーストアクセス（高速ページモード）の動作タイミング

図 6.22 にバーストアクセスの動作タイミングを示します。DRAM 空間へのアクセスサイクルが連続したとき、各アクセスサイクルのロウアドレスが一致している間、カラムアドレスと  $\overline{\text{CAS}}$  信号の出力サイクル（2 ステート）が連続して行われます。バーストアクセスにおいても  $T_{c1}$ - $T_{c2}$  間ウェイトステートを挿入してバスサイクルを引き伸ばすことができます。ウェイトステートの挿入方法、タイミングはフルアクセスのときと同様です。詳細は、「6.5.8 ウェイト制御」を参照してください。

比較対象となるロウアドレスは DRCRB の MXC1、MXC0 ビットと ABWCR で設定された各エリアのバス幅で決まります。表 6.9 に MXC1、MXC0 ビットおよび ABWCR の設定値と比較対象となるロウアドレスの関係を示します。

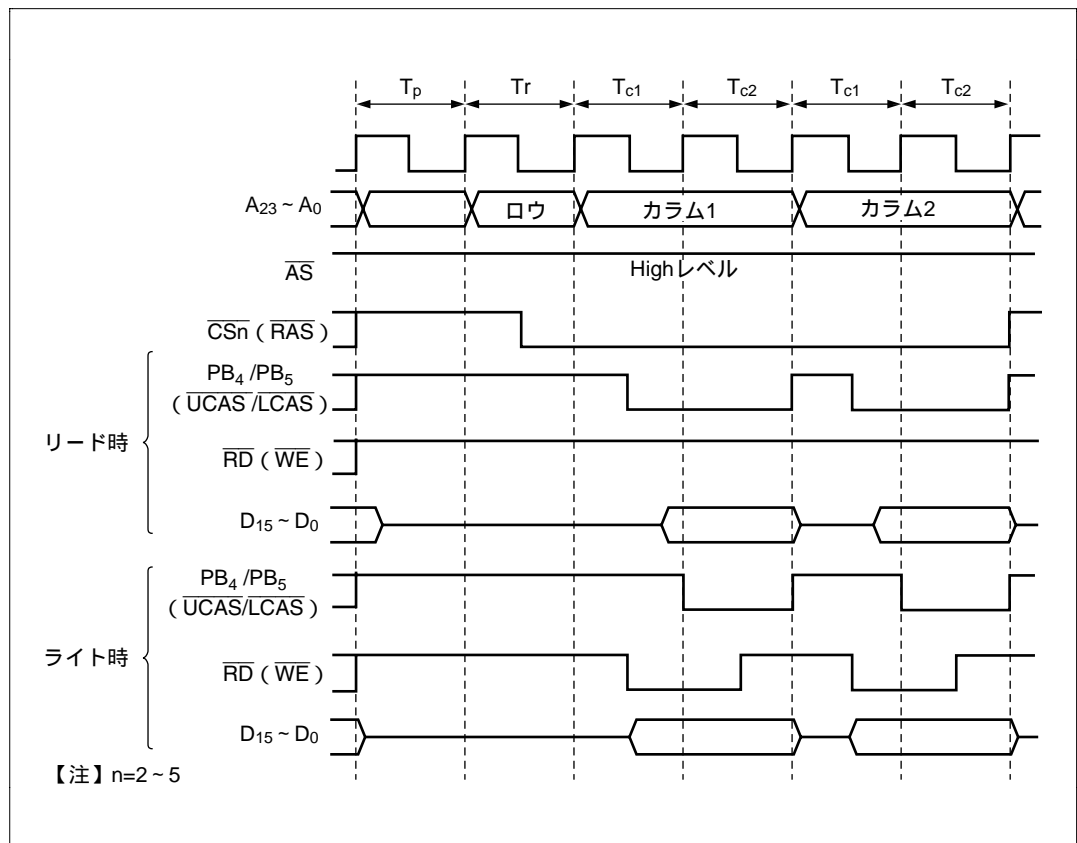


図 6.22 高速ページモードの動作タイミング

表 6.9 MXC1、MXC0 および ABWCR の設定値とバーストアクセス時に比較対象となるロウアドレスの関係

動作モード	DRCRB		ABWCR	バス幅	比較対象の ロウアドレス
	MXC1	MXC0	ABWn		
モード 1、2 (1M バイト)	0	0	0	16 ビット	A19 ~ A9
			1	8 ビット	A19 ~ A8
		1	0	16 ビット	A19 ~ A10
			1	8 ビット	A19 ~ A9
	1	0	0	16 ビット	A19 ~ A11
			1	8 ビット	A19 ~ A10
		1			
モード 3、4、5 (16M バイト)	0	0	0	16 ビット	A23 ~ A9
			1	8 ビット	A23 ~ A8
		1	0	16 ビット	A23 ~ A10
			1	8 ビット	A23 ~ A9
	1	0	0	16 ビット	A23 ~ A11
			1	8 ビット	A23 ~ A10
		1			

【注】 n=2~5

## (2) RAS ダウンモードと RAS アップモード

高速ページモードを備えた DRAM では、アクセスが連続していない場合でも同一ロウアドレスへのアクセスならば、 $\overline{\text{RAS}}$  信号を Low レベルに保持しておくことにより、バースト動作を断続的に続けることができます。

## (a) RAS ダウンモード

RAS ダウンモードを選択するときは、DRCRA の BE、RDM ビットを 1 にセットしてください。DRAM 空間に対するアクセスが途切れて他空間をアクセスしている間、 $\overline{\text{RAS}}$  信号を Low レベルに保持し、次の DRAM 空間アクセスのロウアドレスと前の DRAM 空間アクセスのロウアドレスが一致した場合、バーストアクセスが行われます。図 6.23 に RAS ダウンモードのタイミング例を示します。

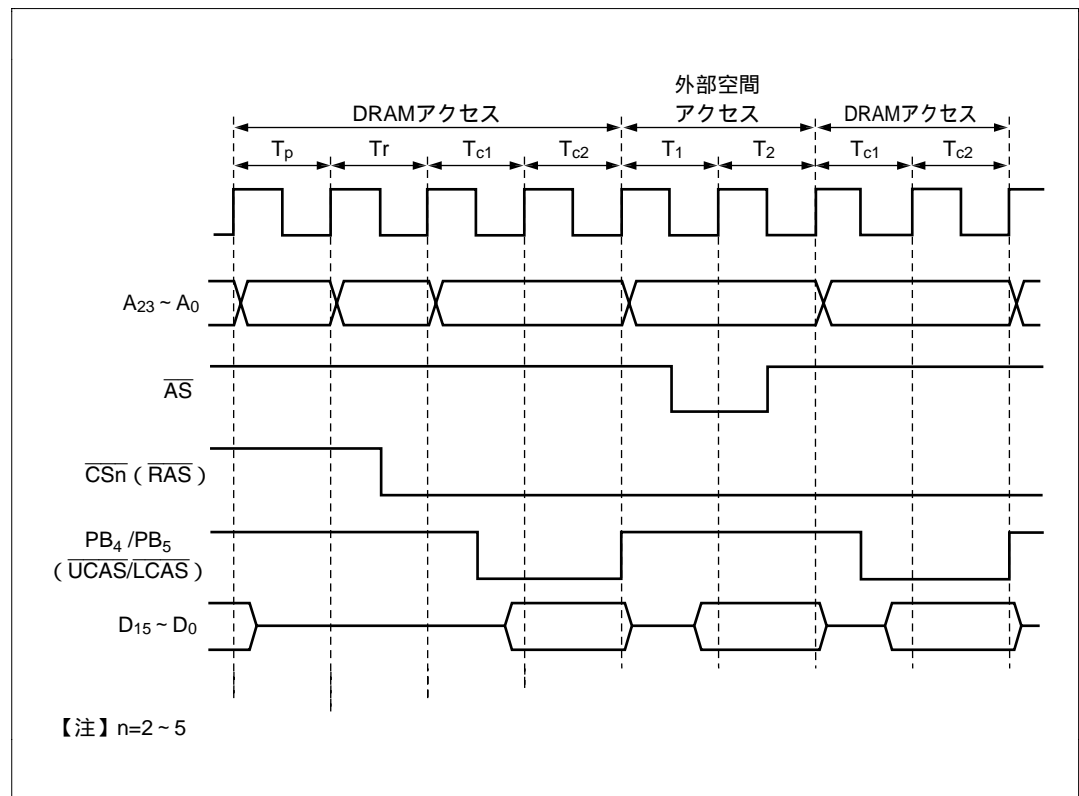


図 6.23 RAS ダウンモードの動作タイミング例  
(CSEL=0 の場合)



RAS ダウンモードを選択している場合、いったんアサートされた  $\overline{\text{RASn}}$  信号が High レベルに戻る条件は以下の通りです。その時のタイミング図を図 6.24 に示します。

- (a) 異なるロウアドレスの DRAM 空間をアクセスする場合
- (b) CAS ビフォ RAS リフレッシュサイクルの直前
- (c) DRCRA の BE ビットまたは RDM ビットへの 0 セット
- (d) 外部バス権解放の直前

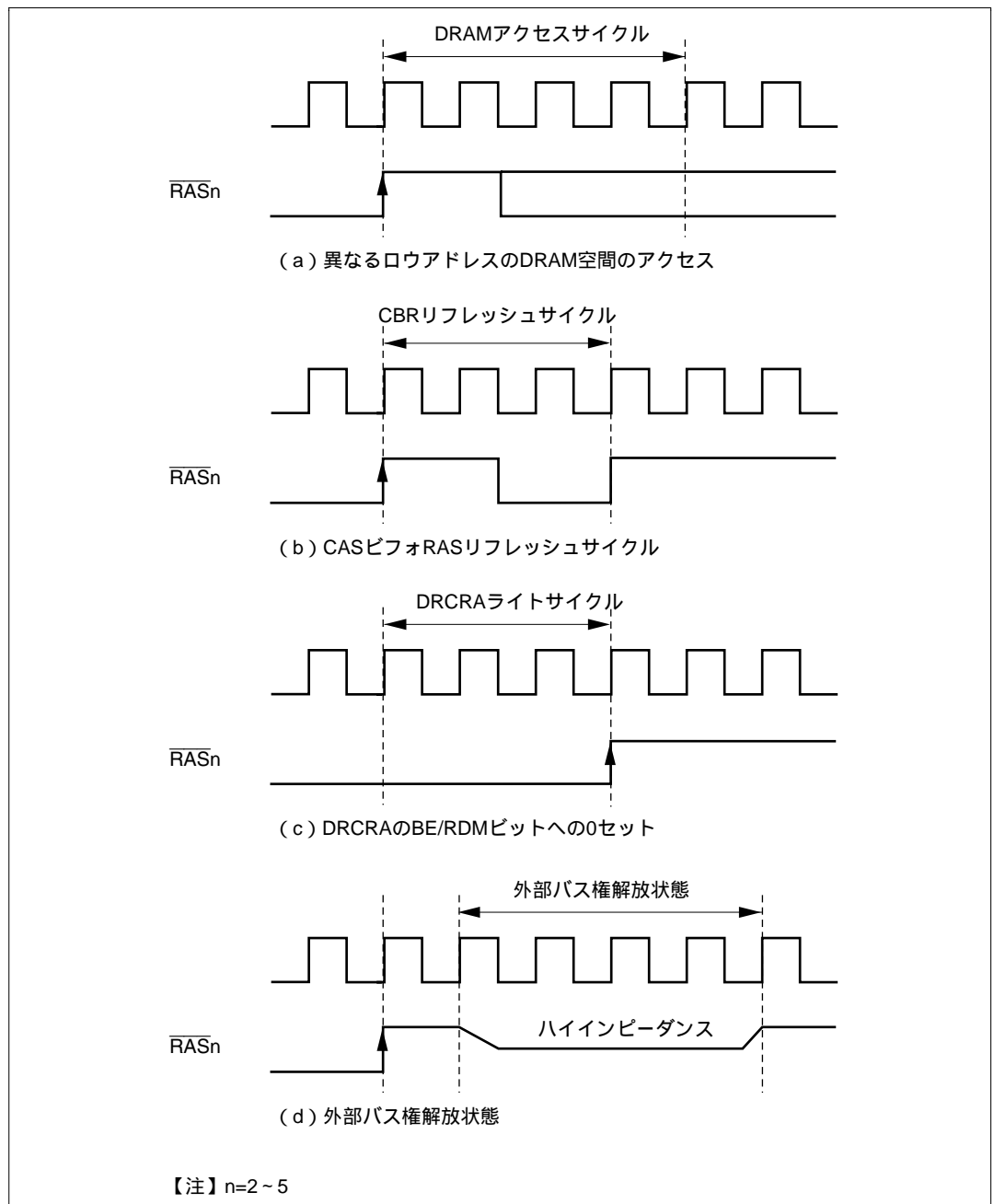


図 6.24 RAS ダウンモード選択時の  $\overline{\text{RASn}}$  のネゲートタイミング

RAS ダウンモードを選択する場合には、DRAM のリフレッシュの方法として必ず、本 DRAM インタフェースの CAS ビフォ RAS リフレッシュ機能を使用してください。リフレッシュ動作を行う時、 $\overline{\text{RAS}}$  信号はその直前に High レベルになります。DRAM の  $\overline{\text{RAS}}$  パルス幅の最大規定を満たすように、リフレッシュ間隔の設定を行ってください。

また、セルフリフレッシュ機能を使用する場合には、ソフトウェアスタンバイモードに遷移する為の SLEEP 命令を実行する前に必ず RDM ビットを 0 にクリアし、RAS アップモードを選択してください。RAS ダウンモードの選択はソフトウェアスタンバイ解除後、あらためて行ってください。

なお、 $\overline{\text{UCAS}}/\overline{\text{LCAS}}$  として  $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$  を選択し、かつ外部空間に DRAM 以外のデバイスを接続し、 $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$  をライトストロープとして使用する場合、RAS ダウンモードは使用できませんので注意してください。

#### (b) RAS アップモード

RAS アップモードを選択するときは、DRCRA の RDM ビットを 0 にクリアしてください。DRAM 空間へのアクセスが途切れて他空間をアクセスするたびに、 $\overline{\text{RAS}}$  信号を High レベルに戻します。DRAM 空間が連続している場合だけ、バースト動作が行われます。

図 6.25 に RAS アップモードのタイミング例を示します。

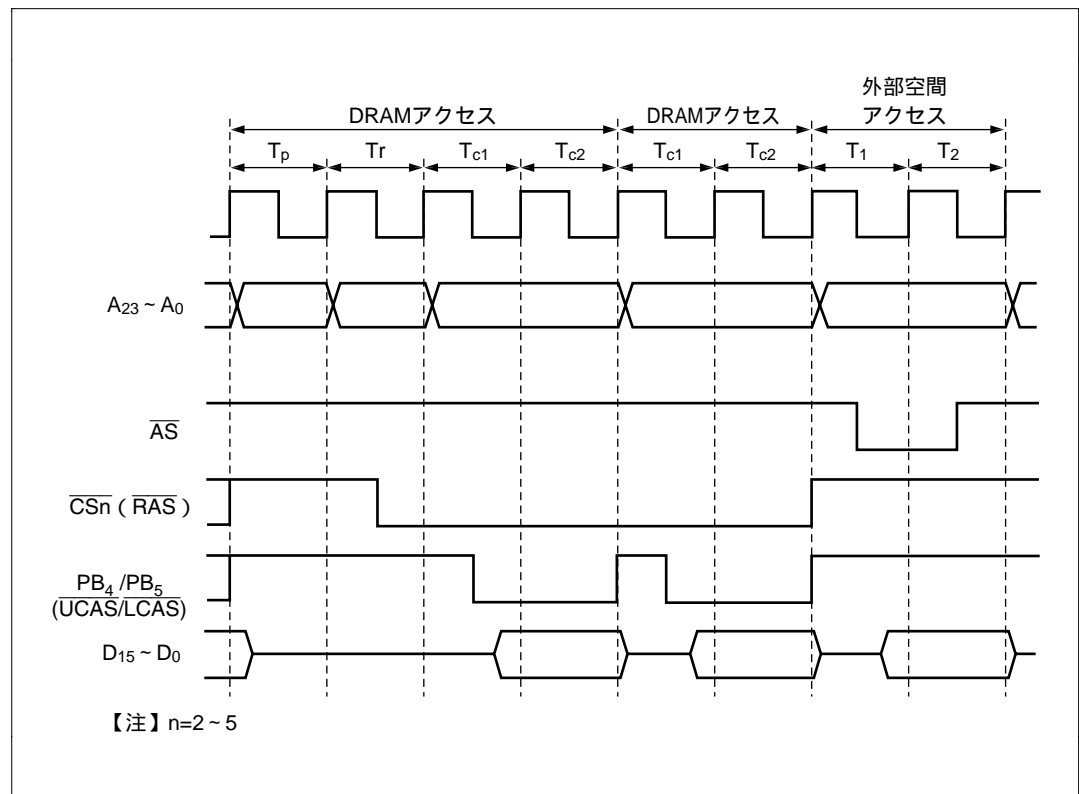


図 6.25 RAS アップモードの動作タイミング例

### 6.5.11 リフレッシュ制御

本 LSI は、DRAM のリフレッシュ制御機能として、CAS ビフォ RAS(CBR)リフレッシュ機能、およびセルフリフレッシュ機能を備えています。

#### (1) CAS ビフォ RAS(CBR)リフレッシュ

CBR リフレッシュを選択するためには、DRCRB の RCYCE ビットを 1 にセットしてください。

CBR リフレッシュでは、RTMCSR の CKS2~CKS0 ビットで選択した入力クロックにより、RTCNT がカウントアップされ、RTCOR に設定した値と一致 (コンペアマッチ) すると、リフレッシュ要求が発生します。同時に RTCNT はリセットされ、H'00 からカウントアップを再開します。すなわち、リフレッシュ要求は RTCOR と CKS2~CKS0 ビットで決まる一定間隔で繰り返し、発生します。リフレッシュサイクルの実行は、このリフレッシュ要求を受け、DRAM インタフェースがバス権を獲得した後に行われます。使用する DRAM のリフレッシュ間隔規定を満たすように、RTCOR と CKS2~CKS0 ビットの値を設定してください。また、RAS ダウンモードを使用する場合には、 $\overline{\text{RAS}}$  パルス幅の最大規定を満たすようにリフレッシュ間隔の設定を行ってください。

RTCNT のカウントアップは、CKS2~CKS0 ビットの設定を行うことにより開始されません。したがって、CKS2~CKS0 ビットの設定を行う前に、RTCNT および RTCOR の設定を行ってください。

なお、バス権要求中に発生したリフレッシュの再要求、あるいはリフレッシュサイクル実行中のリフレッシュ要求は無視されるので注意してください。

図 6.26 に RTCNT の動作を、図 6.27 にコンペアマッチのタイミングを、図 6.28、図 6.29 に CBR リフレッシュのタイミングを示します。

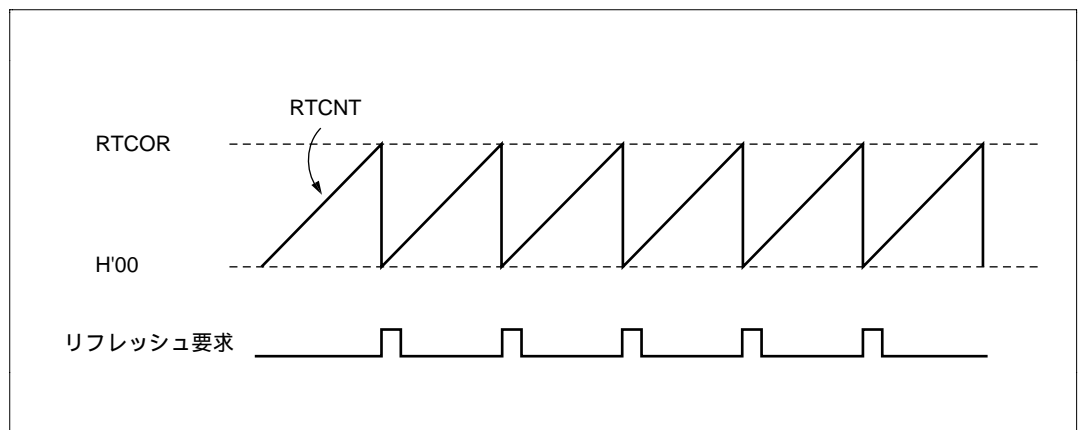


図 6.26 RTCNT の動作

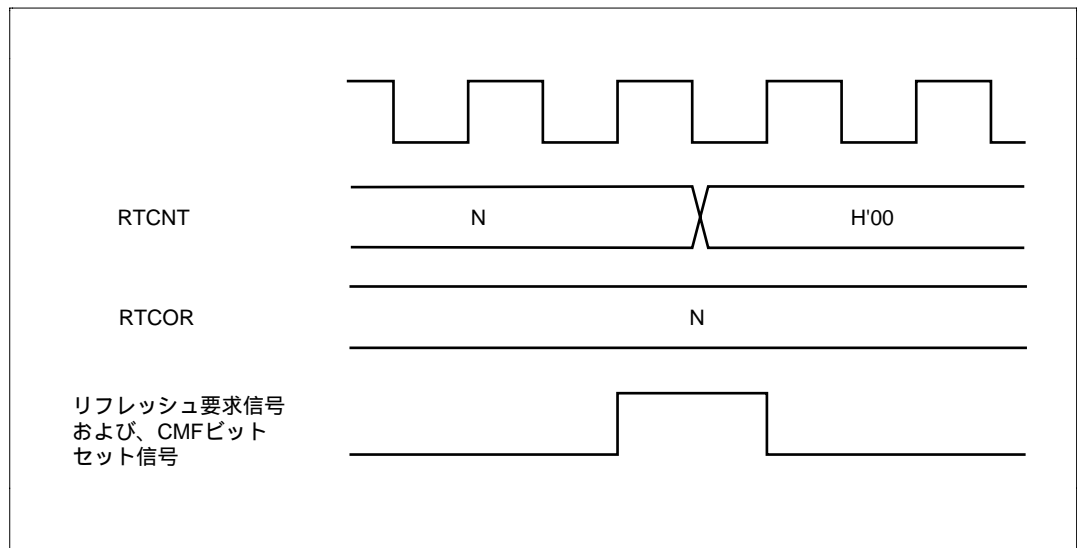
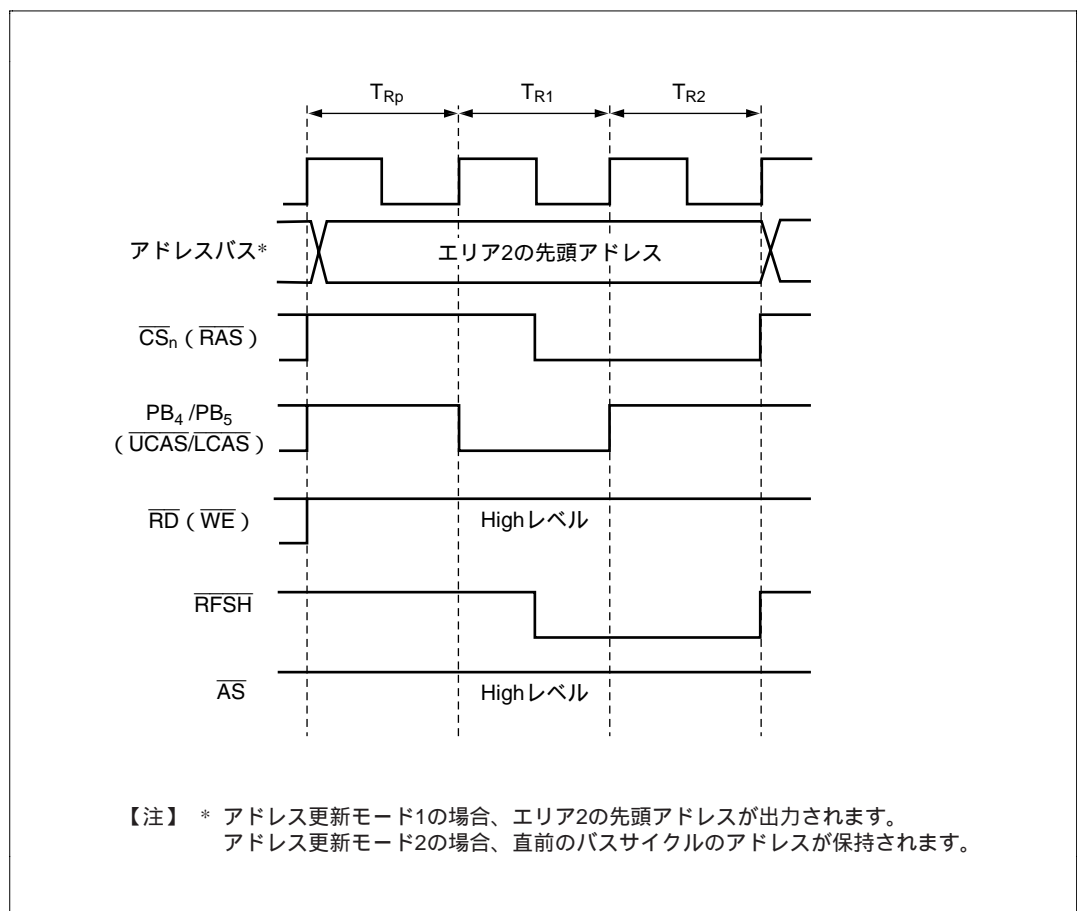


図 6.27 コンペアマッチのタイミング



【注】 \* アドレス更新モード1の場合、エリア2の先頭アドレスが出力されます。  
 アドレス更新モード2の場合、直前のバスサイクルのアドレスが保持されます。

図 6.28 CBR リフレッシュタイミング (CSEL=0,TPC=0,RLW=0 の場合)

CBS リフレッシュサイクルの基本タイミングは、3 ステートで、RAS プリチャージサイクル ( $T_{RP}$ ) 1 ステート、RAS 出力サイクル ( $T_{R1}$ 、 $T_{R2}$ ) 2 ステートで構成されます。RAS プリチャージサイクルのステート数は1ステートまたは2ステートから選択することができます。DRCRB の TPC ビットを1に設定した場合、 $\overline{RAS}$  信号が1サイクル遅れて出力されます。このとき、 $\overline{UCAS}$ 、 $\overline{LCAS}$  の出力タイミングは変化しません。

$\overline{RAS}$  信号幅は、DRCRB の RLW ビットで調整してください。RLW ビットを1に設定することにより  $T_{R1}$  ステートと  $T_{R2}$  ステートの間にリフレッシュウェイトステート ( $T_{RW}$ ) を1ステート挿入することができます。

なお、RLW ビットの設定は、CBR リフレッシュサイクルに対してのみ有効になり、DRAM のリード/ライトサイクルに対しては影響を与えません。また、ASTCR、WCRH、WCRL の設定値および  $\overline{WAIT}$  端子の状態は、CBR リフレッシュサイクルのステート数には影響を与えません。

図 6.29 に TPC ビットおよび RLW ビットにそれぞれ1を設定したときのタイミングを示します。

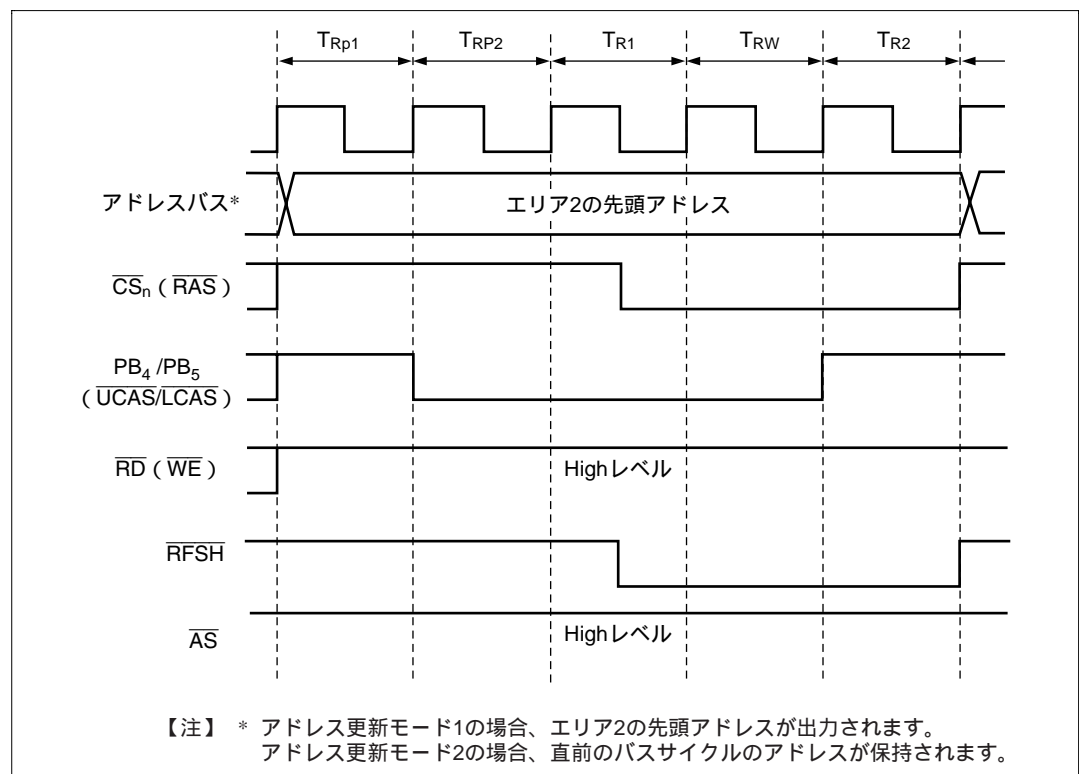


図 6.29 CBR リフレッシュタイミング (CSEL=0、TPC=1、RLW=1 の場合)

DRAM は、電源投入直後、内部状態を安定させるためにリフレッシュを必要とします。したがって、本 LSI の CAS ビフォ RAS リフレッシュ機能を使用する場合には DRCRA の DRAS2~0 の設定直後、他のタイマモジュールによる割込み、あるいは RTMCSR のビット 7 (CMF) がセットされる回数を数えるなどして、DRAM の安定期間を確保してください。

## (2) セルフリフレッシュ

DRAM には、スタンバイモードの一種として、DRAM 内部でリフレッシュタイミングとリフレッシュアドレスを生成するセルフリフレッシュモード(バッテリーバックアップモード)を備えているものがあります。本LSIは、ソフトウェアスタンバイモード時に、DRAM をセルフリフレッシュモードに遷移させる機能を備えています。

セルフリフレッシュ機能を使用するためには、DRCRA の SRFMD ビットを 1 にセットしてください。その後、ソフトウェアスタンバイモードに遷移するための SLEEP 命令を実行すると、図 6.30 に示すように  $\overline{\text{CAS}}$  信号と  $\overline{\text{RAS}}$  信号が出力され DRAM はセルフリフレッシュモードに遷移します。

ソフトウェアスタンバイモードが解除されると、 $\overline{\text{CAS}}$ 、 $\overline{\text{RAS}}$  は High レベル出力となります。

なお、セルフリフレッシュ機能を使用する場合、次の条件を守ってください。

- (1) バーストアクセスを選択している場合、ソフトウェアスタンバイモードに遷移するための SLEEP 命令の実行は、必ず RAS アップモードを選択した状態で行ってください。したがって、RAS ダウンモードを選択している場合には、SLEEP 命令を実行する前に必ず DRCRA の RDM ビットを 0 にクリアし RAS アップモードを選択してください。この場合、RAS ダウンモードの選択はソフトウェアスタンバイモード解除後に改めて行ってください。
- (2) SLEEP 命令の直後の命令を DRAM 空間に設定されたエリアに置かないでください。

以上の条件が満たされない場合、セルフリフレッシュ機能は正しく動作しませんので注意してください。

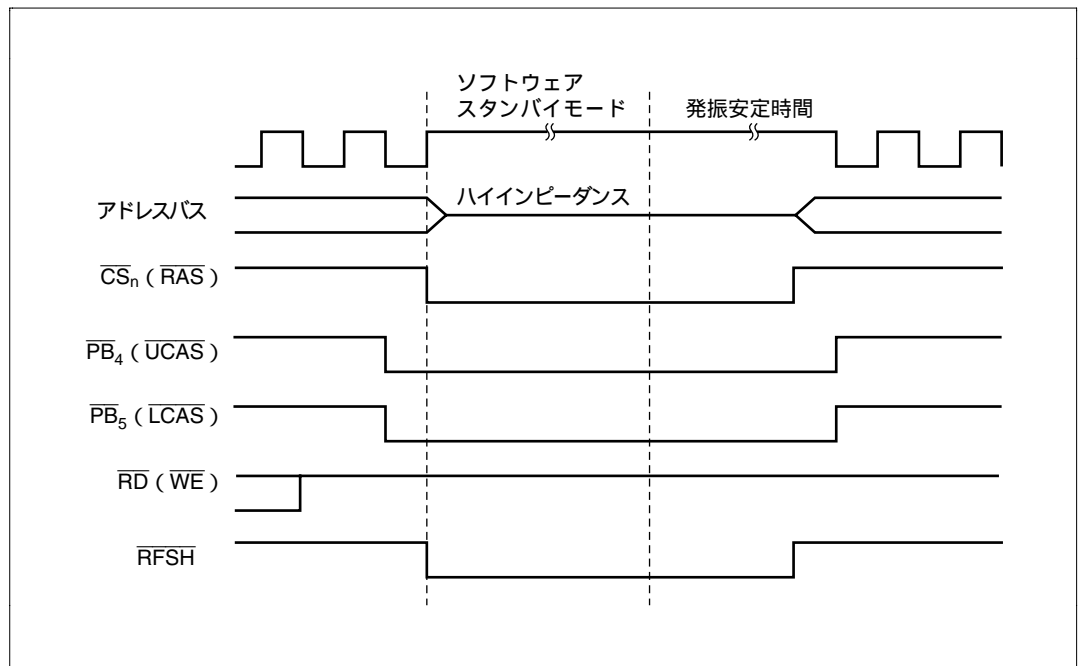


図 6.30 セルフリフレッシュタイミング (CSEL=0 の場合)

### (3) リフレッシュ信号 ( $\overline{\text{RFSH}}$ )

DRCRA の RFSHE ビットを 1 にセットすることで、リフレッシュサイクルを外部に伝えるリフレッシュ信号 ( $\overline{\text{RFSH}}$ ) を出力することができます。  $\overline{\text{RFSH}}$  の出力タイミングは図 6.28、図 6.29、図 6.30 を参照してください。

## 6.5.12 使用例

以下に DRAM の接続例とプログラム設定手順例を示します。DRAM インタフェースを使用する場合には、DRAM デバイス特性を確認の上、そのデバイスに適合する使い方をしてください。

### (1) 接続例

- (1) 図 6.31 に  $\times 16$  ビット構成 (2CAS 方式) の 16M ビット DRAM を 2 個使用する場合の接続例とそのアドレスマップを示します。本例では 10 ビットロウアドレス  $\times$  10 ビットコラムアドレスのタイプのものを使用しています。エリア 2~5 を DRAM 空間に設定することにより最大 4 個の DRAM を接続できます。

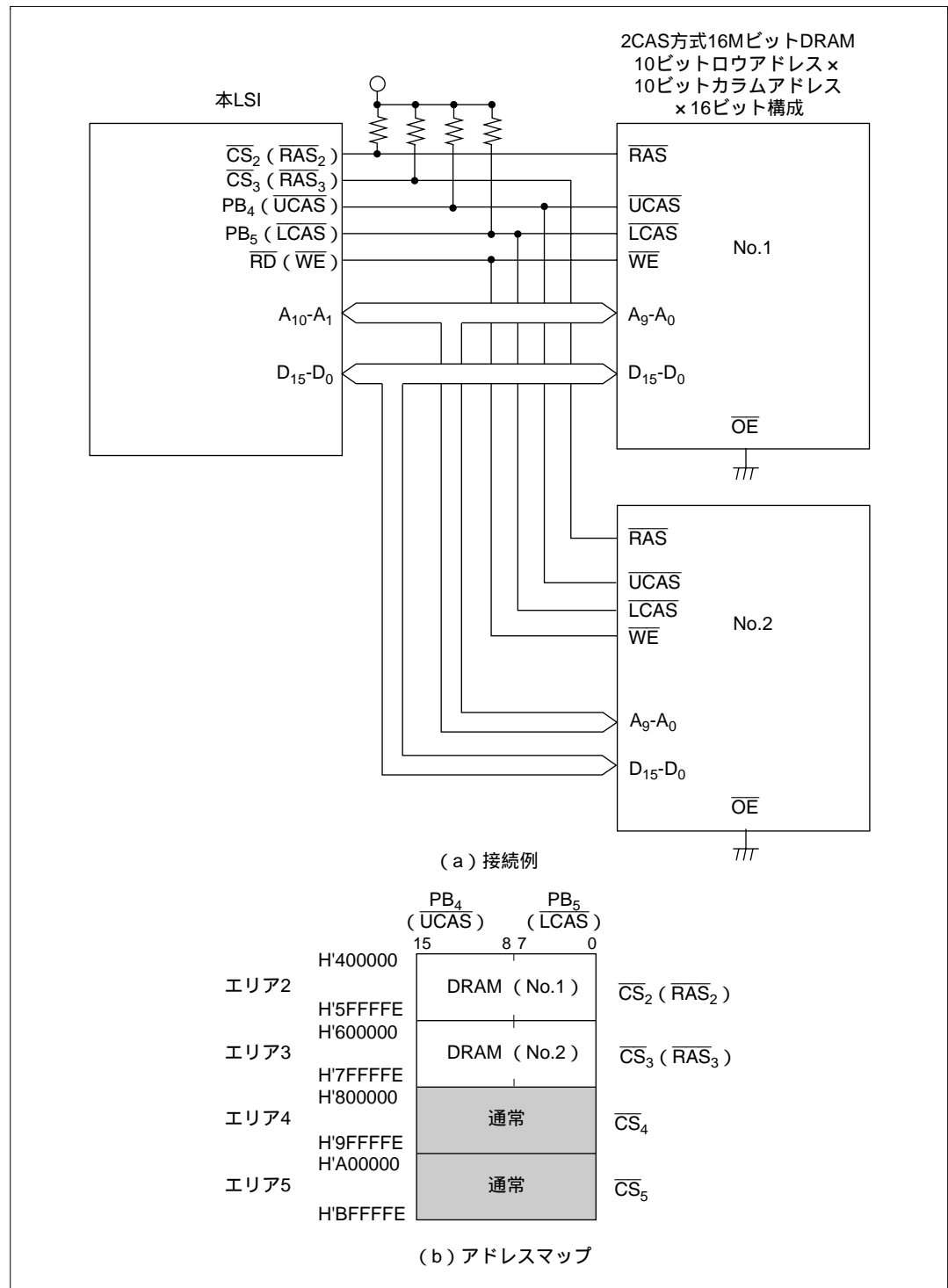


図 6.31 ×16 ビット構成(2CAS 方式)の 16M ビット DRAM の接続例とアドレスマップ

(2) 図 6.32 に×8 ビット構成の 16M ビット DRAM を 2 個使用する場合の接続例とそのアドレスマップを示します。本例では 11 ビットロウアドレス×10 ビットカラムアドレスのタイプのものを使用しています。 $\overline{CS}_2$  端子をエリア 2 およびエリア 3 に共通の  $\overline{RAS}$  出力端子として使用しています。本例のように DRAM のアドレス空間が複数エリアにまたがって連続する場合、DRAS2~0 の設定により、一本の  $\overline{CS}$  端子が複数エリアに共



通常の  $\overline{\text{RAS}}$  出力端子として使用でき、アドレス空間が最大 4 エリアにまたがる大容量 DRAM を直接接続することができます。この場合不要となる  $\overline{\text{CS}}$  端子（本例では  $\overline{\text{CS}}_3$  端子）は入出力ポートとして使用することができます。

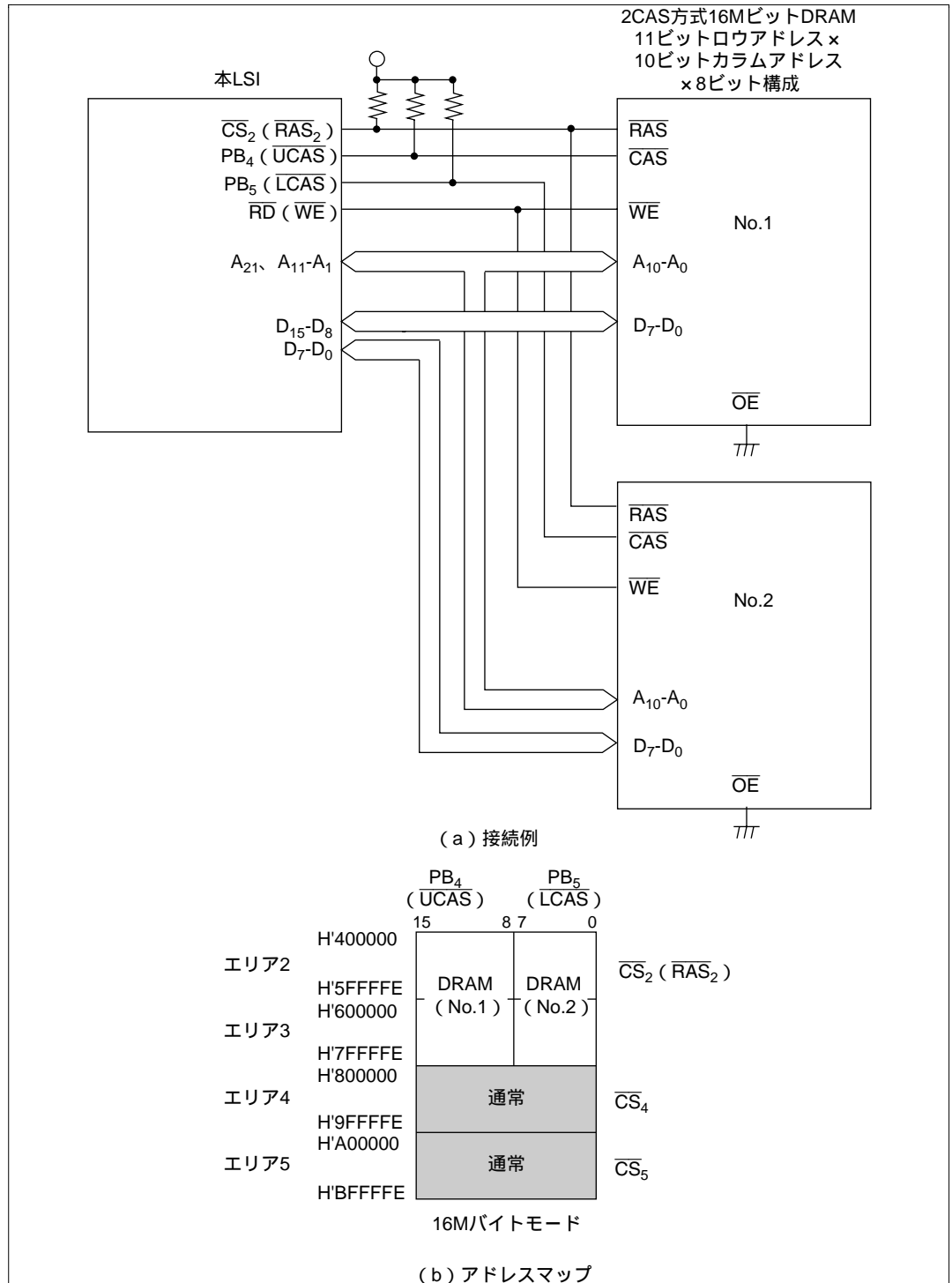


図 6.32 ×8 ビット構成の 16M ビット DRAM の接続例とアドレスマップ

(3) 図 6.33 に 4M ビット DRAM を 2 個使用する場合の接続例とそのアドレスマップを示します。9 ビットロウアドレス × 9 ビットカラムアドレスのタイプのものを使用しています。本例では上位アドレスをデコードすることにより、複数の DRAM を一つのエリアに接続しています。この場合、すべての DRAM を同時にリフレッシュするため、 $\overline{\text{RFSH}}$  端子を使用します。

ただし本接続例では、RAS ダウンモードは使用できませんので注意してください。

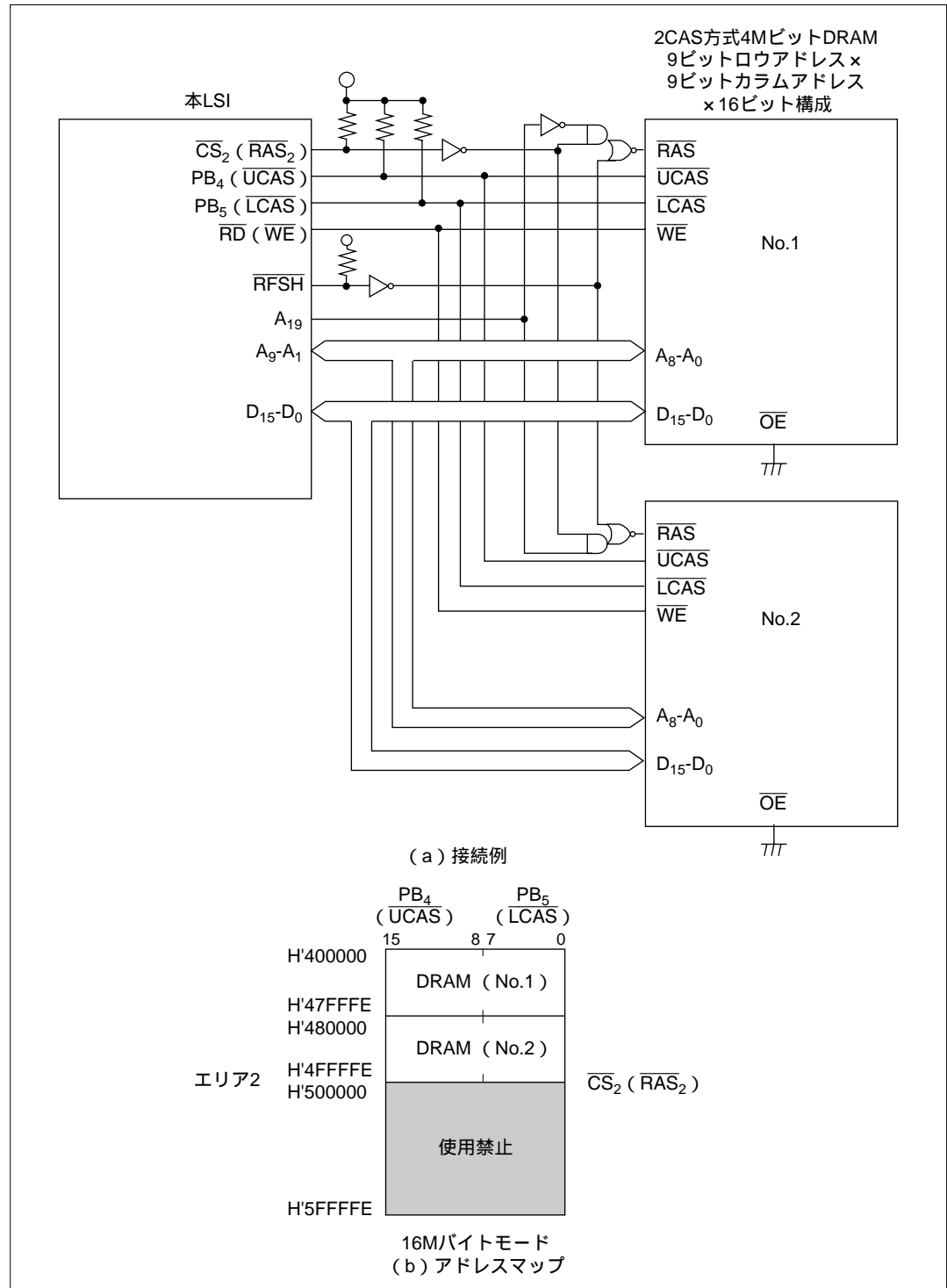


図 6.33 × 16 ビット構成 (2CAS 方式) の 4M ビット DRAM の接続例とアドレスマップ

(2) プログラムの設定順序例

図 6.34 にプログラムの設定順序例を示します。



図 6.34 DRAM インタフェースの使用時の設定手順例

### 6.5.13 使用上の注意

DRAM リフレッシュ機能の使用に際して以下の点に注意してください。

- (1) 外部バス権解放状態、ソフトウェアスタンバイモード、またはバスサイクルがウェイトステートの挿入により長く続く場合、リフレッシュサイクルは実行されません。したがって、これらの状態では、別の方法でリフレッシュを行う必要があります。
- (2) 外部バス権解放状態中に内部でリフレッシュ要求が発生した場合、最初の要求が保持され、バス権解放状態解除後、リフレッシュサイクルが1回だけ実行されます。図 6.35 にその場合のバスサイクルを示します。
- (3) バスサイクルがウェイトステートの挿入により長く続く場合も、外部バス権解放状態と同様に、最初の要求が保持されます。
- (4) ソフトウェアスタンバイモードへ遷移するとき、外部バスマスタからのバス権要求が競合すると、ソフトウェアスタンバイモードへの遷移時に  $\overline{\text{BACK}}$  とストロープの状態が不確定になる場合があります (図 6.36 参照)。ソフトウェアスタンバイモードを使用するときは、SLEEP 命令を実行する前に、BRCR の BRLE ビットを 0 にクリアしてください。また、セルフリフレッシュモードへの遷移でも同様の競合により、ストロープ波形出力が保証されないことがあります。これも BRCR の BRLE ビットを 0 にクリアすることにより防止できます。
- (5) セルフリフレッシュ解除直後、CPU サイクルが始まるまでの一定期間、外部バス権解放が可能です。セルフリフレッシュ直後の  $\overline{\text{RAS}}$  プリチャージ時間の規定を充たすよう  $\overline{\text{RAS}}$  の状態に注意してください。

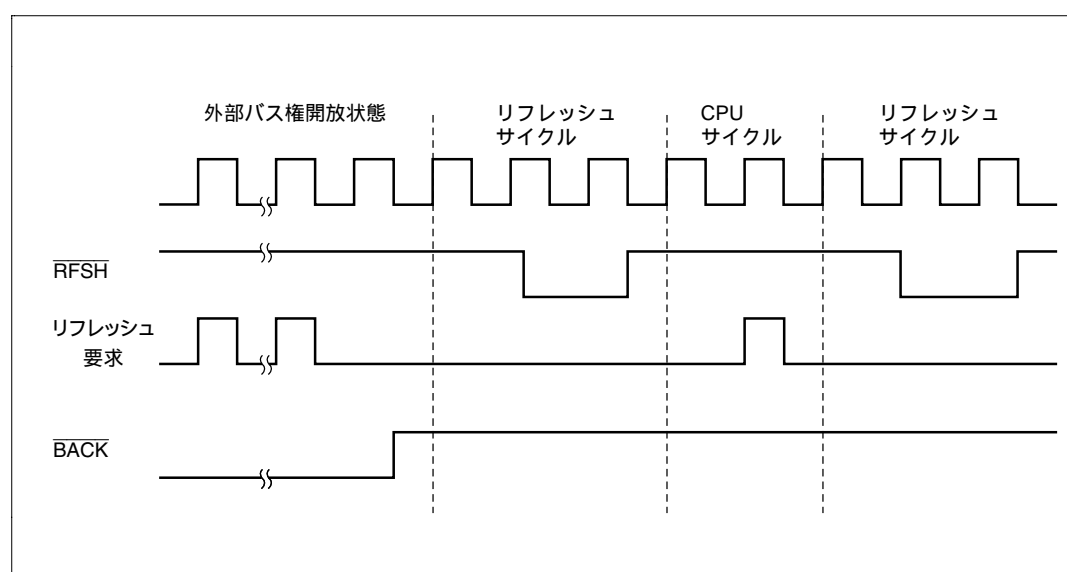


図 6.35 バス解放状態時のリフレッシュサイクル

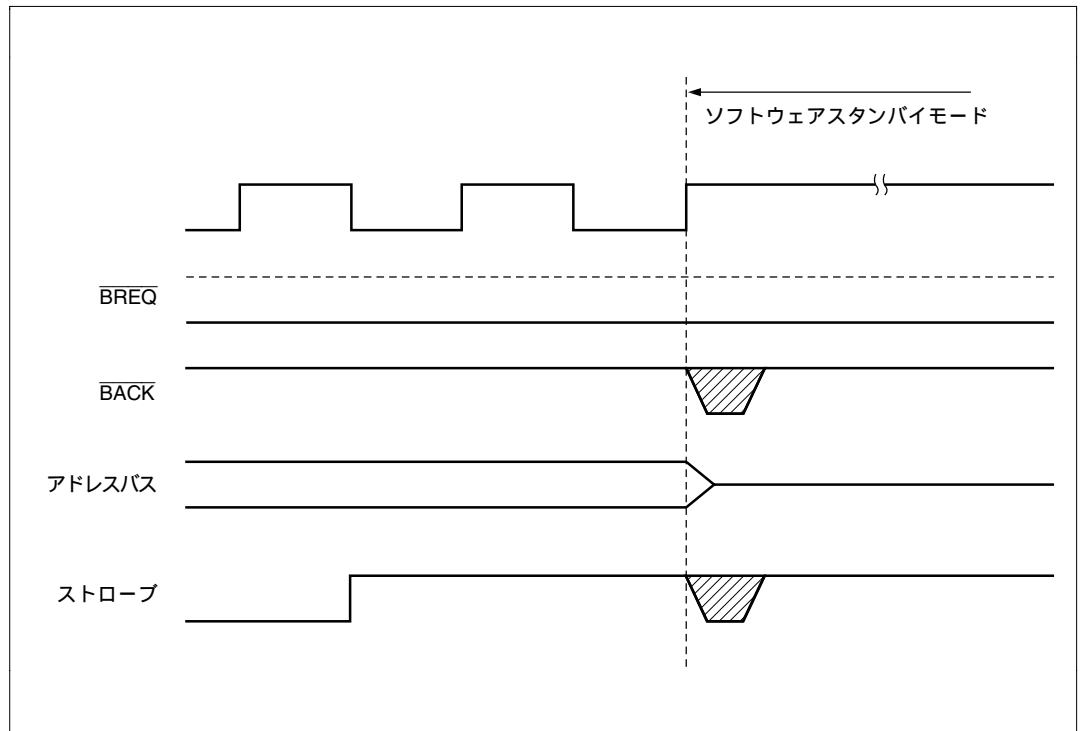


図 6.36 バス解放状態とソフトウェアスタンバイモード

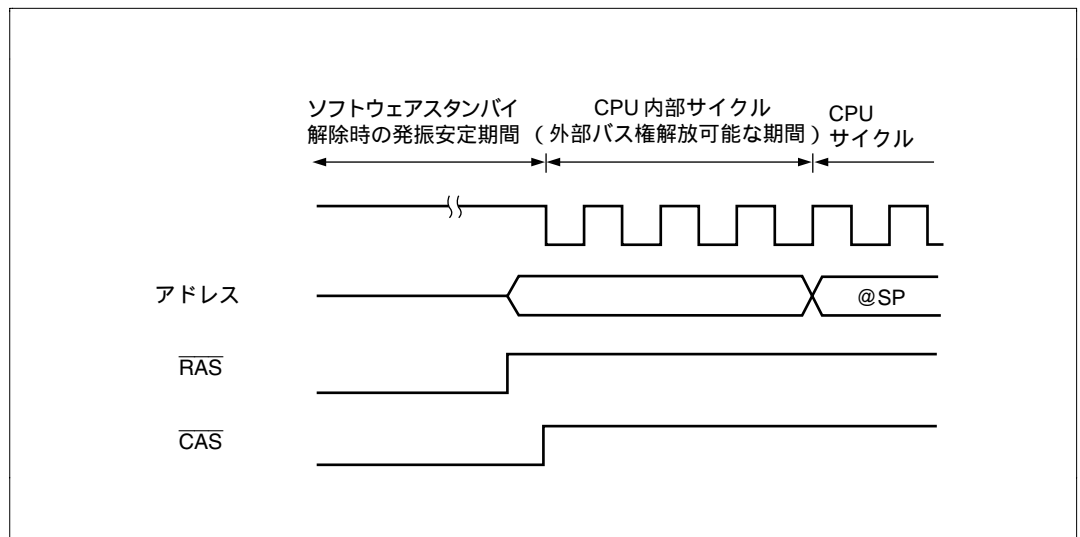


図 6.37 セルフリフレッシュの解除

## 6.6 インターバルタイマ

### 6.6.1 動作説明

本 LSI に DRAM を接続しない場合、リフレッシュタイマをインターバルタイマとして使用することができます。リフレッシュタイマをインターバルタイマとして使用する場合、DRCRA の DRAS2~0 ビットを 0 にクリアします。RTCOR を設定後、RTMCSR の CKS2~CKS0 ビットにより入力クロックを選択し、CMIE ビットを 1 にセットします。

#### (1) コンペアマッチフラグのセットタイミングとコンペアマッチによるクリア

RTMCSR の CMF フラグは、RTCOR と RTCNT の値が一致したときに出力されるコンペアマッチ信号により 1 にセットされます。コンペアマッチ信号は一致した最後のステート (RTCNT が一致したカウント値を更新するタイミング) で発生します。

したがって、RTCNT と RTCOR が一致した後、カウントアップクロックが発生するまでコンペアマッチ信号は発生しません。このタイミングを図 6.38 に示します。

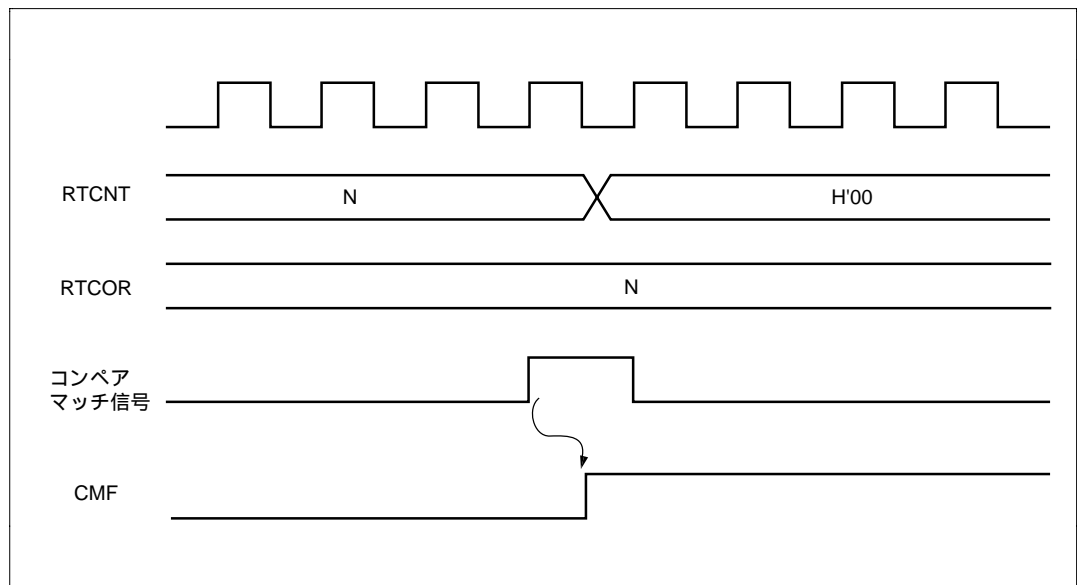


図 6.38 CMF フラグセットタイミング

#### (2) 低消費電力状態時の動作

インターバルタイマは、スリープモードでも動作します。また、ハードウェアスタンバイモード時は動作を停止します。ソフトウェアスタンバイモード時、RTCNT と RTMCSR のビット 7、6 はイニシャライズされますが、RTMCSR のビット 5~3、および RTCOR はソフトウェアスタンバイモードに遷移する直前の状態を保持します。

## (3) RTCNT のライトとカウンタクリアの競合

RTCNT のライトサイクル中の  $T_3$  ステートで、カウンタクリアが発生するとカウンタへのライトは行われず、クリアが優先されます。

図 6.39 にこのタイミングを示します。

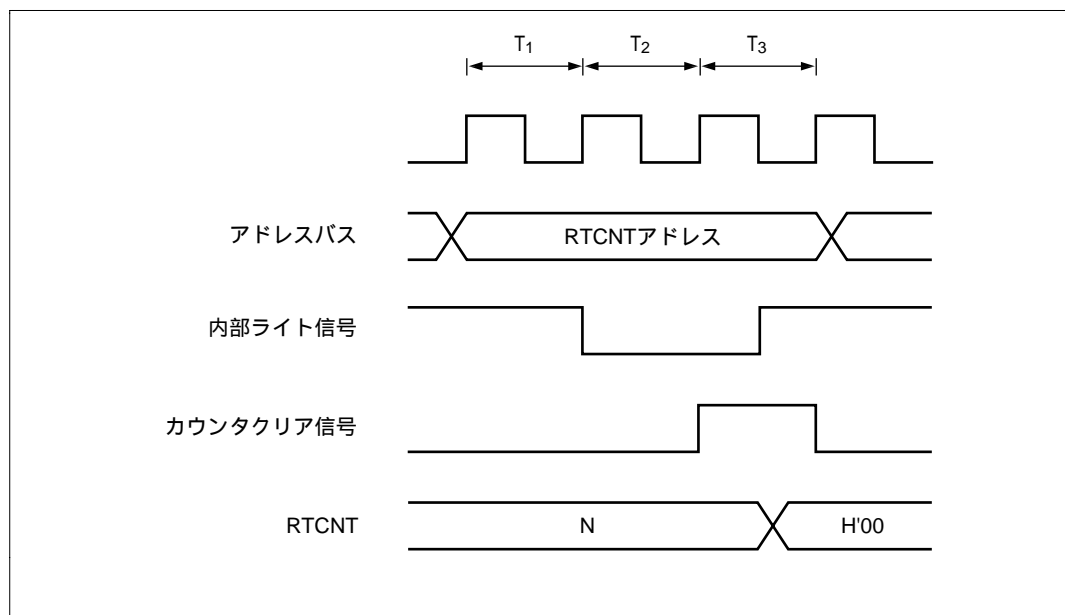


図 6.39 RTCNT のライトとクリアの競合

#### (4) RTCNT のライトとカウントアップの競合

RTCNT のライトサイクル中の  $T_3$  ステートでカウントアップが発生しても、カウントアップされずに、カウンタライトが優先されます。

図 6.40 にこのタイミングを示します。

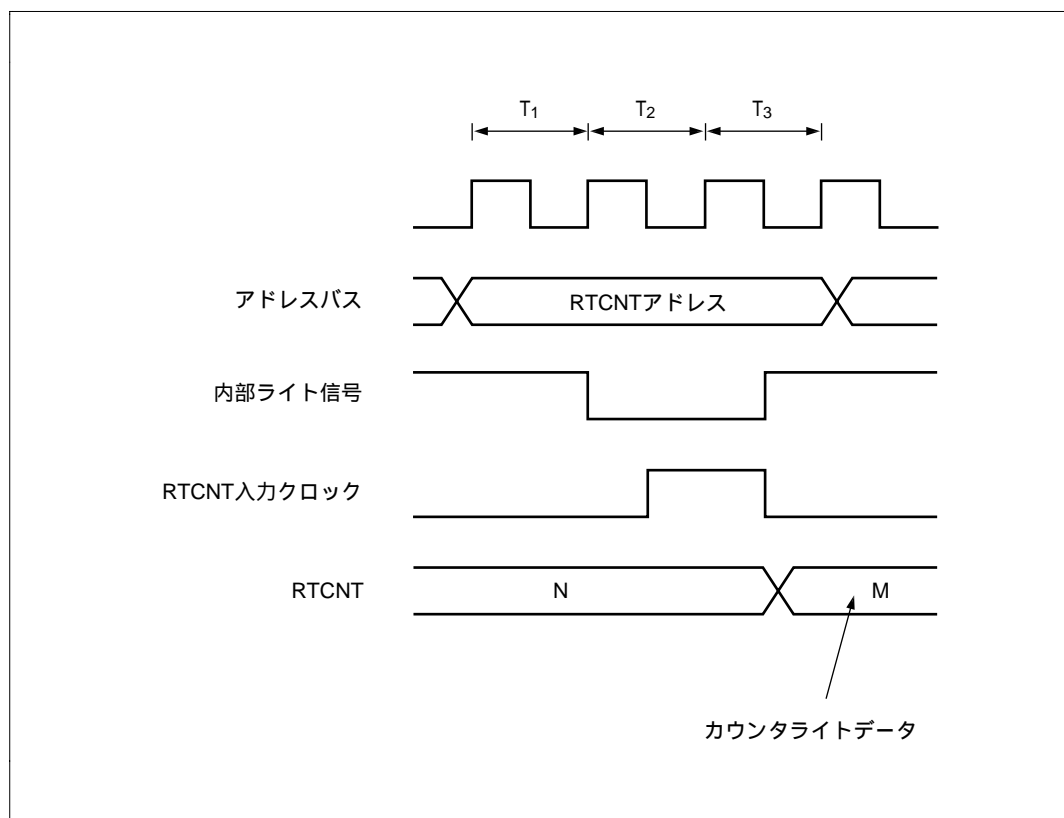


図 6.40 RTCNT のライトとカウントアップの競合



## (5) RTCOR のライトとコンペアマッチの競合

RTCORのライトサイクル中の $T_3$ 状態でコンペアマッチが発生しても、図 6.41 のように RTCOR のライトが優先され、コンペアマッチ信号は禁止されます。

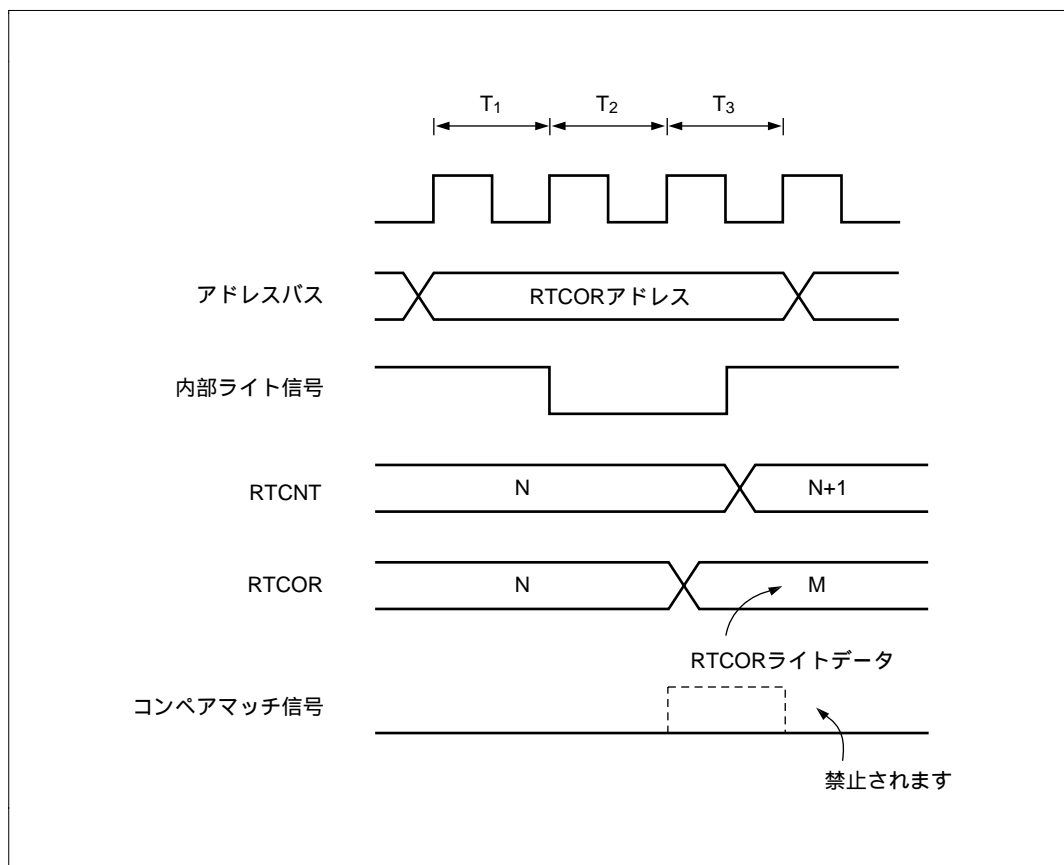


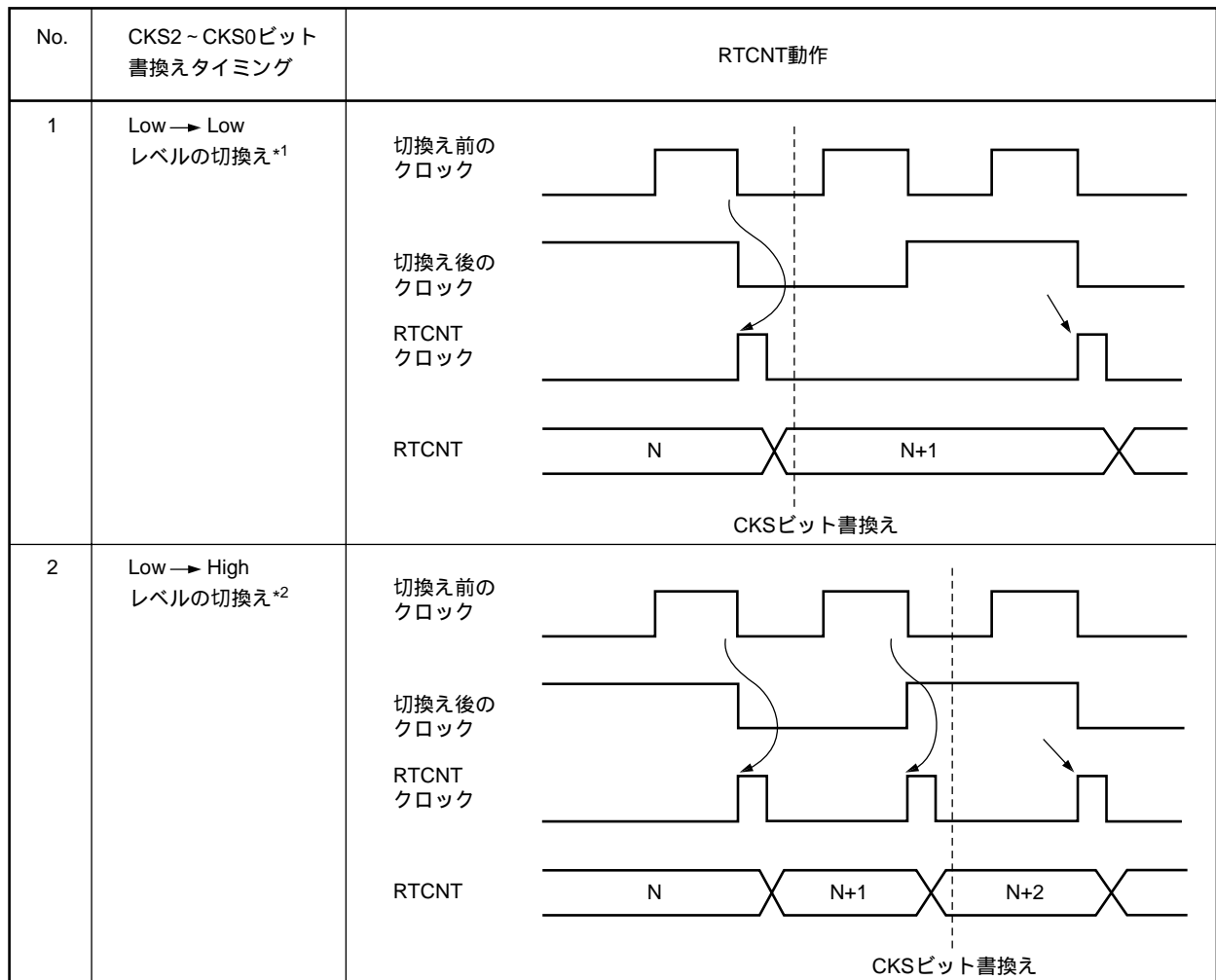
図 6.41 RTCOR のライトとコンペアマッチの競合

## (6) 内部クロックの切換えと RTCNT の動作

内部クロックを切換えるタイミングによっては、RTCNT がカウントアップされてしまう場合があります。内部クロックの切換えタイミング (CKS2 ~ CKS0 ビットの書換え) と RTCNT 動作の関係を表 6.10 に示します。

内部クロックから RTCNT クロックを生成する場合、内部クロックの立下がりエッジで検出しています。そのため、表 6.10 の No.3 のように High Low になるようなクロックの切換えを行うと、切換えタイミングを立下がりエッジとみなして RTCNT クロックが発生し、RTCNT がカウントアップされてしまいます。

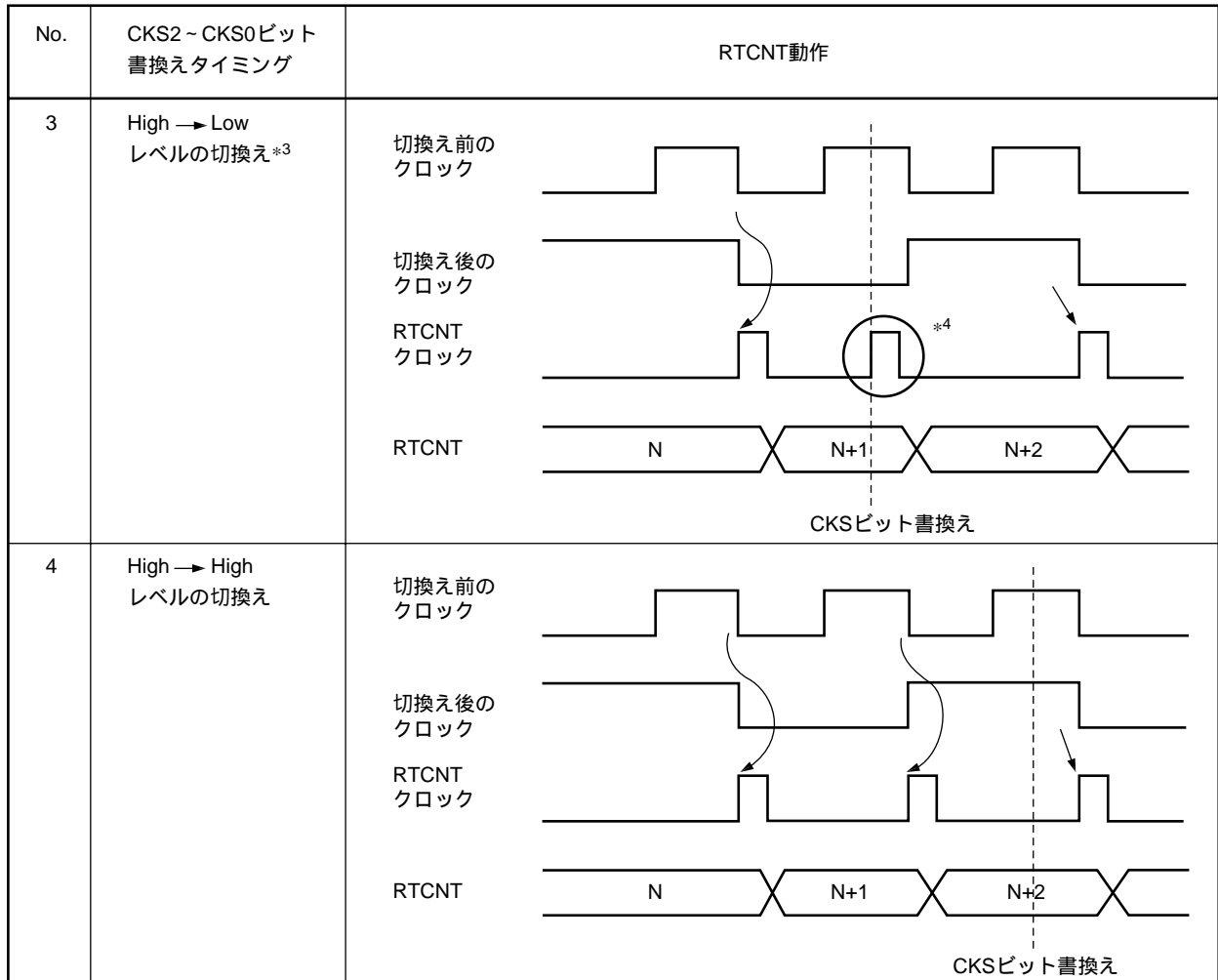
表 6.10 内部クロックの切換えと RTCNT 動作 (1)



【注】\*1 Lowレベル → 停止、および停止 → Lowレベルの場合を含みます。

\*2 停止 → Highレベルの場合を含みます。

表 6.10 内部クロックの切換えと RTCNT 動作 (2)



【注】\*3 Highレベル → 停止の場合を含みます。

\*4 切換えのタイミングを立下がりエッジとみなすために発生し、RTCNTはカウントアップしてしまいます。

## 6.7 割込み要因

リフレッシュタイマをインターバルタイマとして使用する場合、コンペアマッチ割込み (CMI) 要求を発生します。コンペアマッチ割込み要求は RTMCSR の CMIE ビットで許可または禁止することができます。

## 6.8 バースト ROM インタフェース

### 6.8.1 概要

本 LSI は、エリア 0 の外部空間を、バースト ROM 空間に設定し、バースト ROM インタフェースを行うことができます。バースト ROM インタフェースでは、バーストアクセス可能な ROM を高速にアクセスすることができます。BCR の BROME ビットにより、エリア 0 をバースト ROM 空間に設定します。

エリアの外部空間を最大 4 ワード / 8 ワードの連続バーストアクセスを行うことができます。バーストアクセスは 2 ステート / 3 ステート選択可能です。

### 6.8.2 基本タイミング

バースト ROM インタフェースの、イニシャルサイクル (フルアクセス) およびバーストサイクルのアクセスステート数は ASTCR の AST0 ビットの設定に従います。また、AST0 ビットを 1 にセットした場合は、イニシャルサイクルにウェイトステートを挿入することもできます。バーストサイクルは、ウェイトステートは挿入できません。

また、BCR の BRSTS0 ビットを 0 にクリアすると最大 4 ワードのバーストアクセスを行います。BRSTS0 ビットを 1 にセットすると最大 8 ワードのバーストアクセスを行います。バーストアクセスのステート数は、BRSTS1 ビットを 0 にクリアすると、2 ステートになります。BRSTS1 ビットを 1 にセットすると 3 ステートとなります。

バースト ROM 空間の基本アクセスタイミングを図 6.42 に示します

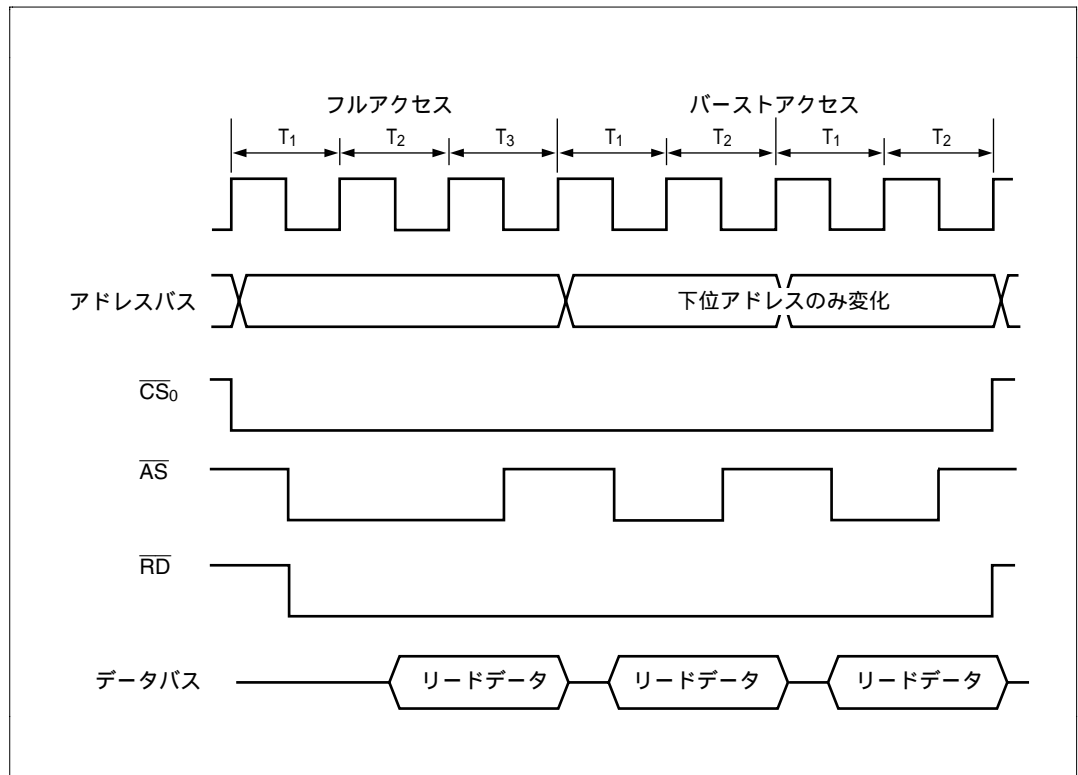


図 6.42 バースト ROM アクセスタイミング例

### 6.8.3 ウェイト制御

バーストROMインタフェースのインisialサイクル(フルアクセス)には、基本バスインタフェースと同様に、(1)プログラムウェイトの挿入、(2)  $\overline{\text{WAIT}}$  端子による端子ウェイトの挿入、が可能です。

バーストサイクルにはウェイトステートを挿入することはできません。

## 6.9 アイドルサイクル

### 6.9.1 動作説明

本 LSI は外部空間をアクセスするとき、(1)異なるエリア間でリードアクセスが連続して発生したとき、(2)リードサイクルの直後にライトサイクルが発生したとき、(3) DRAM 空間アクセスの直後に、DRAM 空間以外の外部アドレス空間アクセスが発生したときの3つの場合に、バスサイクルとバスサイクルの間にアイドルサイクル ( $T_i$ ) を1ステート挿入することができます。アイドルサイクルを挿入することにより、例えば出力フローティング時間の大きい ROM などと、高速メモリ、I/O インタフェースなどとのデータ衝突を防ぐことができます。

BCR の ICIS1、ICIS0 ビットとも初期値は 1 ですので、初期状態でアイドルサイクルは挿入されます。データ衝突がない場合に、ICIS ビットのクリアが可能です。

#### (1) 異なるエリア間での連続リード

BCR の ICIS1 ビットを 1 にセットした状態で、異なるエリア間の連続リードが発生すると、2 回目のリードサイクルの先頭に、アイドルサイクルが挿入されます。

図 6.43 に動作例を示します。バスサイクル A は、出力フローティング時間の大きい ROM からのリードサイクル、バスサイクル B は SRAM からのリードサイクルで、それぞれ異なるエリアに配置した場合の例です。(a) はアイドルサイクルを挿入しない場合で、バスサイクル B で、ROM からのリードデータと SRAM からのリードデータの衝突が発生しています。これに対し (b) ではアイドルサイクルを挿入し、データの衝突を回避しています。

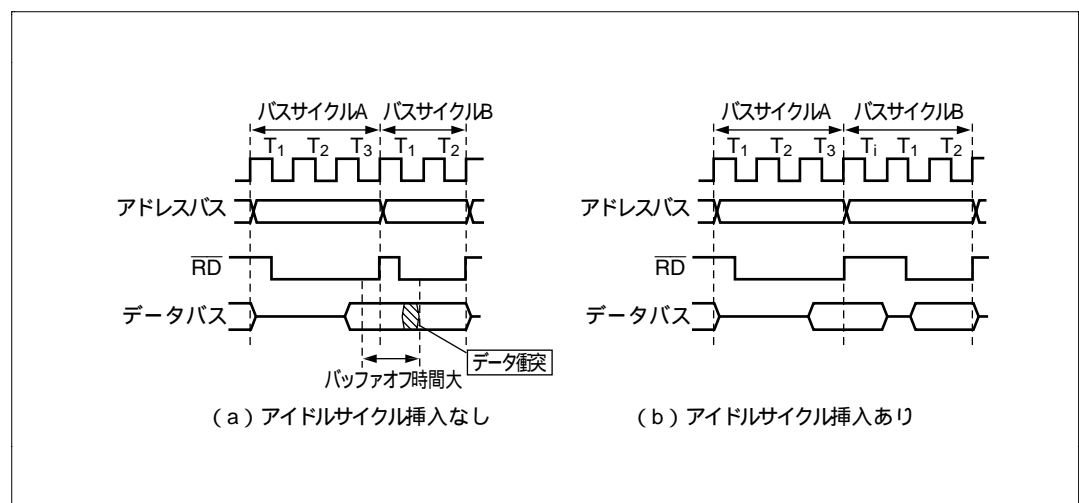


図 6.43 アイドルサイクル動作例 (1) (ICIS1=1 のとき)

## (2) リード後のライト

BCRのICIS0ビットを1にセットした状態で外部リード後に、外部ライトが発生するとライトサイクルの先頭に、アイドルサイクルが挿入されます。

図6.44に動作例を示します。バスサイクルAは、出力フローティング時間の大きいROMからのリードサイクル、バスサイクルBはCPUのライトサイクルの場合の例です。

(a)はアイドルサイクルを挿入しない場合で、バスサイクルBで、ROMからのリードデータとCPUのライトデータの衝突が発生しています。これに対し(b)ではアイドルサイクルを挿入し、データの衝突を回避しています。

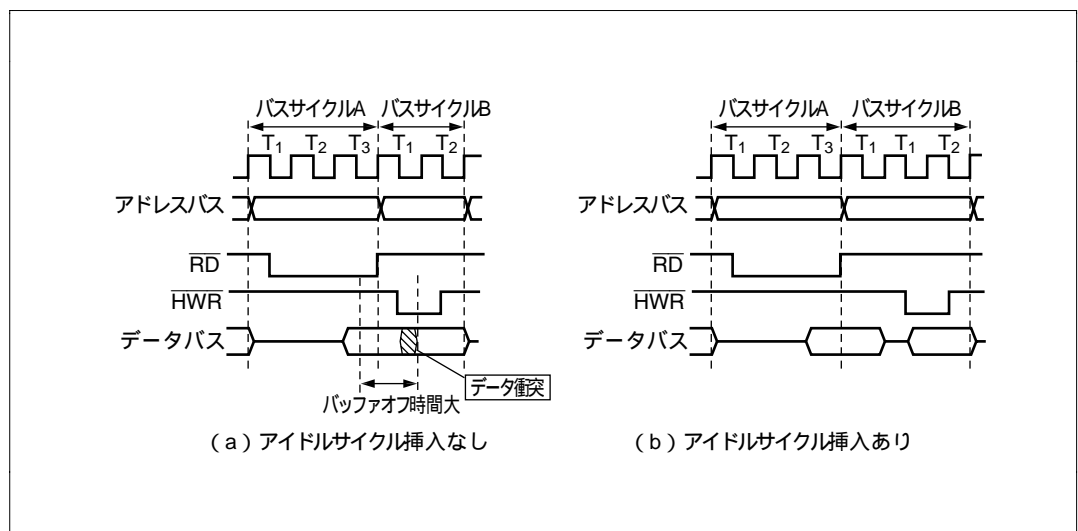


図 6.44 アイドルサイクル動作例 (2) (ICIS0=1 のとき)

## (3) DRAM 空間アクセス直後の外部アドレス空間アクセス

DRCRBのCSELビットにより $\overline{UCAS}$ , $\overline{LCAS}$ の出力端子を $\overline{HWR}$ , $\overline{LWR}$ に選択した場合、DRAM空間アクセス後にDRAM以外の外部アクセスが発生するとBCRのICIS0,ICIS1ビットの値にかかわらずT<sub>i</sub>サイクルが挿入されます。図6.45に動作例を示します。

これはDRAM空間で $\overline{UCAS}$ , $\overline{LCAS}$ として使用している $\overline{HWR}$ , $\overline{LWR}$ と次サイクルの空間の $\overline{CSn}$ が同じタイミングで変化するのを避け、次サイクルの外部デバイスへ誤って書込むのを回避するためです。

なお、 $\overline{UCAS}$ , $\overline{LCAS}$ の出力端子をPB4,PB5に選択した場合はT<sub>i</sub>サイクルは挿入されません。

DRAM空間アクセスのプリチャージサイクル(T<sub>p</sub>)が連続する場合には、ICIS0,1ビットの設定は無効です。例えば、異なるエリア間での連続リードの場合、2回目のリードがDRAMアクセスのとき、T<sub>p</sub>サイクルのみが挿入され、T<sub>i</sub>サイクルは挿入されません。

このタイミングを図6.44に示します。

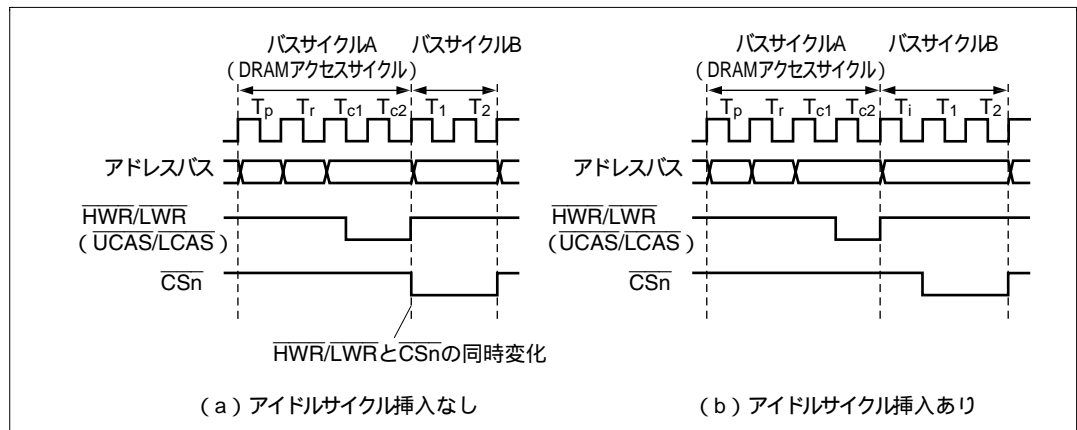
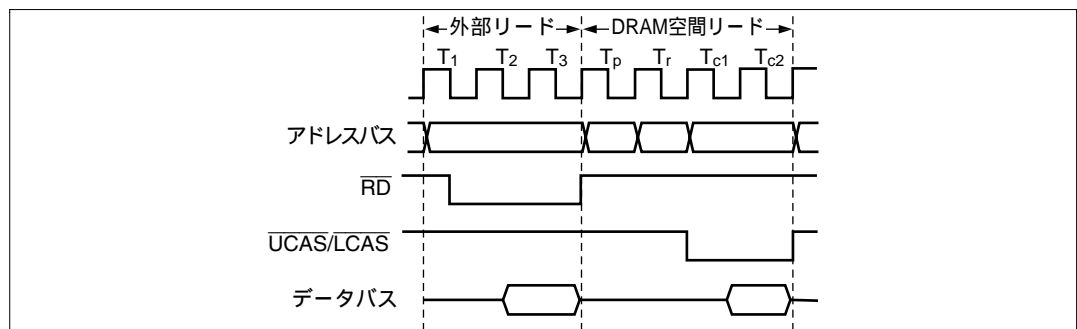
図 6.45 アイドルサイクル動作例(3) ( $\overline{\text{UCAS/LCAS}}$ として $\overline{\text{HWR/LWR}}$ を使用した場合)

図 6.46 アイドルサイクル動作例(4) (プリチャージサイクルが連続する場合)

#### (4) 使用上の注意

アイドルサイクルの挿入を行わない設定とした場合、 $\overline{\text{RD}}$ の立ち上がり(ネゲート)と $\overline{\text{CSn}}$ の立ち下がり(アサート)が同じタイミングで発生する場合があります。図 6.47 に動作例を示します。

BCRのICIS1ビットを0にセットした状態で、異なる外部エリア間での連続リードが発生した場合、またはICIS0ビットを0にセットした状態で、外部リード後に異なる外部エリアに対するライトサイクルが発生した場合、はじめのリードサイクルにおける $\overline{\text{RD}}$ のネゲートと、連続するバスサイクルの $\overline{\text{CSn}}$ のアサートが同じタイミングで発生します。したがって各信号の出力遅延時間に依存し、前のリードサイクルの $\overline{\text{RD}}$ のLow出力と、連続するバスサイクルの $\overline{\text{CSn}}$ のLow出力がオーバーラップする可能性があります。

$\overline{\text{RD}}$ と $\overline{\text{CSn}}$ の同時変化が発生しない場合、または発生しても問題にならない場合に限り、アイドルサイクルの挿入を行わない設定とすることができます。



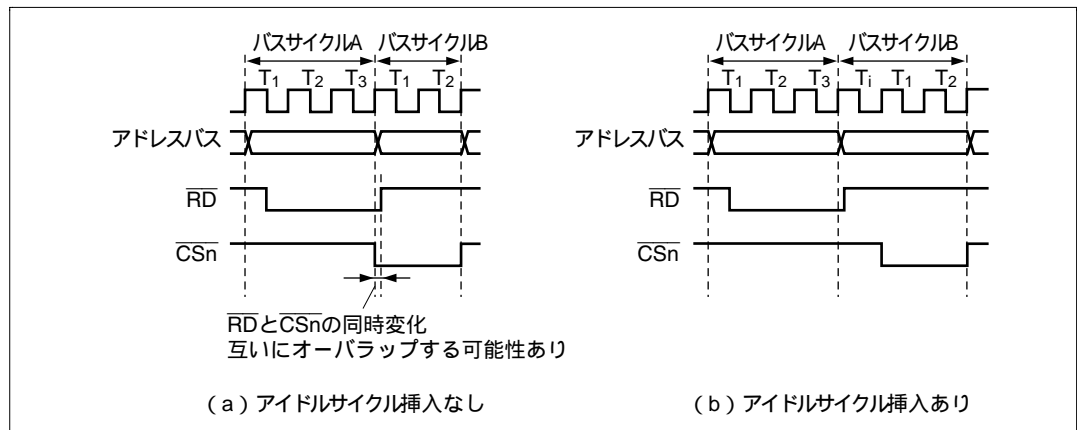


図 6.47 アイドルサイクル動作例 (5)

### 6.9.2 アイドルサイクルでの端子状態

アイドルサイクルでの端子状態を表 6.11 に示します。

表 6.11 アイドルサイクルでの端子状態

端子名	端子の状態
A <sub>23</sub> ~ A <sub>0</sub>	次サイクルのアドレスの値
D <sub>15</sub> ~ D <sub>0</sub>	ハイインピーダンス
$\overline{CS}_n$	High レベル* <sup>1</sup>
$\overline{UCAS}, \overline{LCAS}$	High レベル
$\overline{AS}$	High レベル
$\overline{RD}$	High レベル
$\overline{HWR}$	High レベル
$\overline{LWR}$	High レベル

【注】 \*<sup>1</sup> DRAM 空間での RAS ダウンモードでは Low レベルを保持します。

## 6.10 バスアービタ

バスコントローラは、バスマスタの動作を調停（バスアービトレーション）するバスアービタを内蔵しています。

バスマスタは、CPU、DMAC、DRAM インタフェース、外部バスマスタの4つがあり、バス権を占有した状態でリード/ライトやリフレッシュ動作を行います。各バスマスタはバス権要求信号によりバス権を要求します。バスアービタは所定のタイミングで優先順位を判定し、バス権要求アクノリッジ信号によりバスの使用を許可します。バスマスタはバス権を獲得して動作します。

バスアービタは、バスマスタのバス権要求信号がアクティブになっているかどうかを調べ、アクティブになっていれば、そのバスマスタにバス権要求アクノリッジ信号を返します。複数のバスマスタからバス権要求があれば、最も優先順位の高いものにバス権要求アクノリッジ信号を返します。バス権要求アクノリッジ信号を受け取ったバスマスタは、以後この信号が取り消されるまでバスを使用します。

バスマスタの優先順位は、

(高) 外部バスマスタ > DRAM インタフェース > DMAC > CPU (低)

の順となっています。バスアービタは常にバス権要求信号をサンプリングし、優先順位を判定していますが、現在のバスマスタよりも優先順位の高いバスマスタからバス権要求があったとしても、すぐにバス権を放棄するとは限りません。

各バスマスタには、それぞれ自分より優先順位の高いバスマスタにバス権を譲ることができるタイミングがあります。

### 6.10.1 動作説明

#### (1) CPU

CPUは最も優先順位の低いバスマスタです。CPUがバスマスタの場合にDMAC、DRAMインタフェース、または外部バスマスタからのバス権要求が発生するとバスアービタはバス権を要求があったバスマスタに移行します。バス権が移行するタイミングは次のとおりです。

- (1) バスサイクルの切れ目でバス権を移行します。ワードデータアクセスを2回のバイトアクセスに分割して行う場合には、これら2つのバイトアクセスの間ではバス権は移行しません。
- (2) CPUが乗除算命令などの内部動作を行っている場合、他のバスマスタからバス権要求が発生するとただちに、バス権が移行します。CPUの内部動作は継続されます。

(3) CPUがスリープモードの場合、他のバスマスタからバス権要求が発生するとただちにバス権が移行します。

## (2) DMAC

DMACは、起動要求が発生するとバスアービタに対してバス権を要求します。

DMACがバスマスタの場合にリフレッシュコントローラ、または外部バスマスタからのバス権要求が発生すると、バスアービタはバス権の要求があったバスマスタに移行します。バス権が移行するタイミングは次のとおりです。

DMACの1バイトまたは1ワードの転送が終了したとき、バス権を移行します。DMACの転送サイクルはリードサイクルとライトサイクルで構成され、これらのリードサイクルと次のライトサイクルの間ではバス権は移行しません。

なお、DMACの各チャンネルには優先順位が設けられています。詳細については、「7.4.9 DMAC複数チャンネルの動作」を参照してください。

## (3) DRAM インタフェース

DRAMインタフェースは、リフレッシュサイクル要求が発生するとバスマスタに対してバス権を要求します。リフレッシュサイクルが終了すると、バス権を放棄します。詳細は、「6.5 DRAMインタフェース」を参照してください。

#### (4) 外部バスマスタ

BRCR の BRLE ビットを 1 にセットすると外部バスマスタにバス権を解放することができます。外部バスマスタは最も優先順位の高いバスマスタであり、 $\overline{\text{BREQ}}$  端子を Low レベルにすることにより、バスアービタに対してバス権を要求します。外部バスマスタがいったんバス権を獲得すると  $\overline{\text{BREQ}}$  を Low レベルにしている間、バス権を保持し続けます。本 LSI は、外部バス権解放状態になると、アドレスバス、データバス、バス制御信号 ( $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$ )、チップセレクト信号 ( $\overline{\text{CSn}}$ :  $n=7\sim 0$ ) がハイインピーダンスとなります。また外部バス権解放状態では、 $\overline{\text{BACK}}$  端子が Low レベル出力となります。

バスアービタは、の立上がりで  $\overline{\text{BREQ}}$  端子をサンプリングします。 $\overline{\text{BREQ}}$  端子の Low レベルをサンプリングすると所定のタイミングで外部バス権解放状態となります。 $\overline{\text{BACK}}$  端子が Low レベルになるまで  $\overline{\text{BREQ}}$  端子を Low レベルに保持してください。

外部バス権解放で、 $\overline{\text{BREQ}}$  端子の High レベルを 2 回連続してサンプリングすると、 $\overline{\text{BACK}}$  端子を High レベルにしてバス権解放サイクルを終了します。図 6.48 に 2 ステートアクセス空間のリードサイクル中に外部バスマスタからバス権を要求された場合の動作タイミングを示します。 $\overline{\text{BREQ}}$  端子を Low レベルとしてから外部バス権解放状態となるまで最小 3 ステートがかかります。

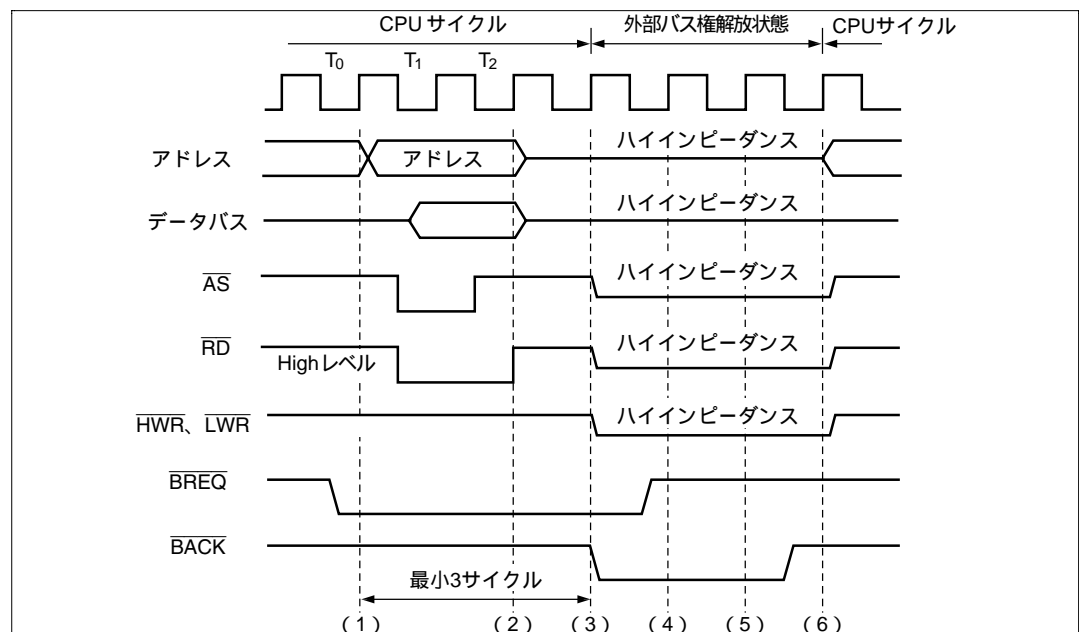


図 6.48 外部バスマスタ動作例

なお、ソフトウェアスタンバイモードへ遷移するとき、外部バスマスタからのバス権要求が競合すると、ソフトウェアスタンバイモードへの遷移時に  $\overline{\text{BACK}}$  とストロープの状態が不確定になる場合があります (図 6.36 参照)。

ソフトウェアスタンバイモードを使用するときは、SLEEP 命令を実行する前に、BRCR の BRLE ビットを 0 にクリアしてください。

## 6.11 レジスタと端子入力のタイミング

### 6.11.1 レジスタライトタイミング

#### (1) ABWCR、ASTCR、WCRH および WCRL のライトタイミング

ABWCR、ASTCR、WCRH および WCRL をライトした場合、ライトデータは次のバスサイクルから有効となります。

このタイミングを図 6.49 に示します。

エリア 0 上の命令でエリア 0 を 3 ステートアクセス空間から 2 ステートアクセス空間へ変更した場合の例です。

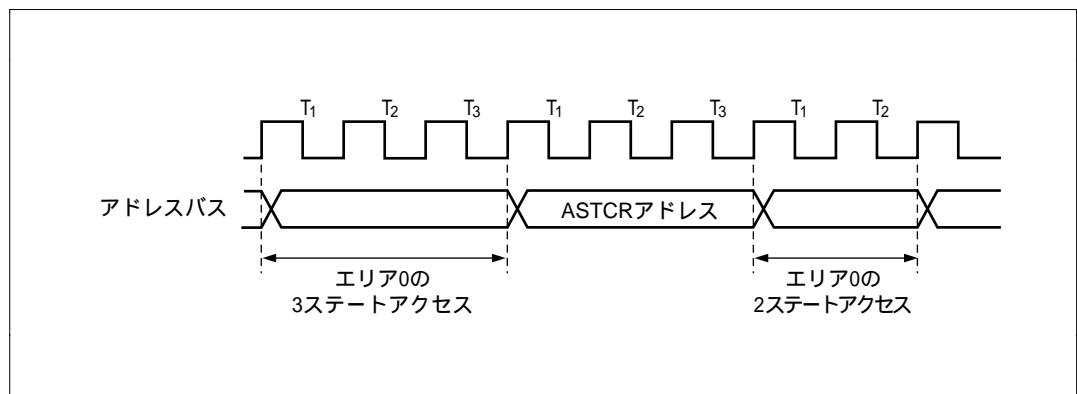


図 6.49 ASTCR ライトタイミング

#### (2) DDR および CSCR のライトタイミング

$\overline{CS}_n$  端子に対応するポートの DDR または CSCR をライトし、 $\overline{CS}_n$  出力と入力ポートを切り換える場合、ライトデータは DDR ライトサイクルの  $T_3$  から有効になります。このタイミングを図 6.50 に示します。 $\overline{CS}_1$  端子を出力とする場合の例です。

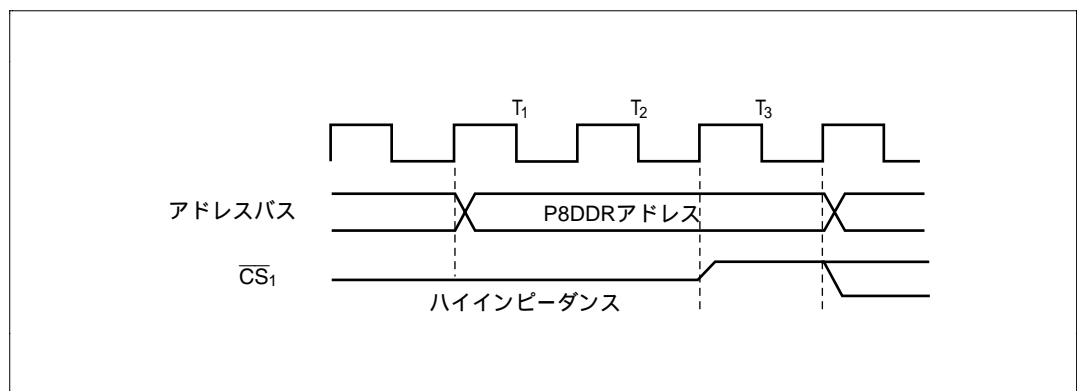


図 6.50 DDR ライトタイミング

### (3) BRCR ライトタイミング

BRCR にライトし、 $A_{23} \sim A_{20}$  出力と入出力ポートを切り換える場合、ライトデータは BRCR ライトサイクルの  $T_3$  から有効になります。このタイミングを図 6.51 に示します。入力ポートを  $A_{23} \sim A_{20}$  出力とする場合の例です。

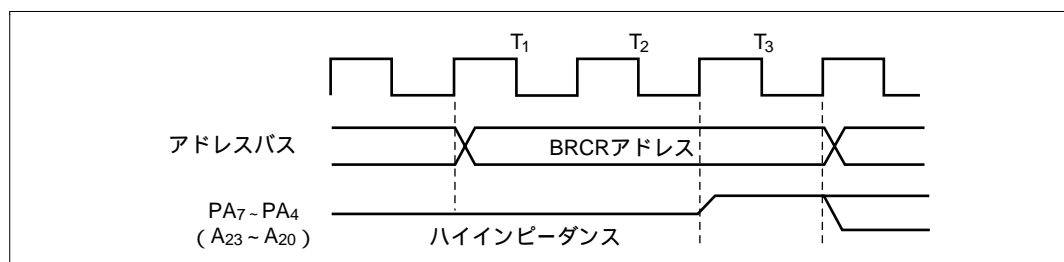


図 6.51 BRCR ライトタイミング

### 6.11.2 $\overline{\text{BREQ}}$ 端子の入力タイミング

$\overline{\text{BREQ}}$  端子を Low レベルにした後、 $\overline{\text{BACK}}$  端子が Low レベルになるまで Low レベルを保持してください。 $\overline{\text{BACK}}$  端子が Low レベルになる前に  $\overline{\text{BREQ}}$  端子を High レベルに戻すと、バスアービタの動作が不確定となりますので注意してください。

また、外部バス権解放状態を終了するときには  $\overline{\text{BREQ}}$  端子を 3 ステート以上 High レベルにしてください。 $\overline{\text{BREQ}}$  端子の High レベルの期間が短いとバスアービタの動作が不確定となりますので注意してください。



---

# 7. DMA コントローラ

---

## 第7章 目次

7.1	概要	223
7.1.1	特長	223
7.1.2	ブロック図	224
7.1.3	機能概要	225
7.1.4	端子構成	227
7.1.5	レジスタ構成	227
7.2	各レジスタの説明(1)(ショートアドレスモード)	229
7.2.1	メモリアドレスレジスタ(MAR)	230
7.2.2	I/Oアドレスレジスタ(IOAR)	231
7.2.3	転送カウントレジスタ(ETCR)	231
7.2.4	データトランスファコントロールレジスタ(DTCR)	233
7.3	各レジスタの説明(2)(フルアドレスモード)	237
7.3.1	メモリアドレスレジスタ(MAR)	237
7.3.2	I/Oアドレスレジスタ(IOAR)	237
7.3.3	転送カウントレジスタ(ETCR)	238
7.3.4	データトランスファコントロールレジスタ(DTCR)	240
7.4	動作説明	248
7.4.1	概要	248
7.4.2	I/Oモード	250
7.4.3	アイドルモード	253
7.4.4	リピートモード	255
7.4.5	ノーマルモード	260
7.4.6	ブロック転送モード	263
7.4.7	DMACの起動要因	268
7.4.8	DMACのバスサイクル	270
7.4.9	DMAC複数チャンネルの動作	274
7.4.10	外部バス権要求、DRAMインタフェースとDMACの関係	275



---

7.4.11	NMI 割込みと DMAC.....	276
7.4.12	DMAC 動作の強制終了.....	277
7.4.13	フルアドレスモードの解除.....	277
7.4.14	リセット、スタンバイモード、スリープモードの DMAC の状態.....	278
7.5	割込み.....	279
7.6	使用上の注意.....	280
7.6.1	ワードデータ転送時の注意.....	280
7.6.2	DMAC による DMAC 自体のアクセス .....	280
7.6.3	MAR のロングワードアクセス.....	280
7.6.4	フルアドレスモード設定時の注意.....	280
7.6.5	内部割込みで DMAC を起動する場合の注意 .....	281
7.6.6	NMI 割込みとブロック転送モード .....	282
7.6.7	MAR、IOAR のアドレス指定.....	283
7.6.8	転送中断時のバスサイクル.....	283
7.6.9	A/D 変換器による転送要求.....	284

## 7.1 概要

本 LSI は、DMA コントローラ (DMAC) を内蔵しています。DMAC は最大 4 チャンネルのデータ転送を行うことができます。

消費電流低減のため DMA コントローラを使用しない場合には、DMA コントローラを単独で停止することができます。詳細は「20.6 モジュールスタンバイ機能」を参照してください。

### 7.1.1 特長

DMAC には次の特長があります。

ショートアドレスモードとフルアドレスモードを選択可能

#### (1) ショートアドレスモード

- ・転送元、転送先アドレスの一方を 24 ビット、他方を 8 ビットで指定
- ・最大 4 チャンネルを使用可能
- ・I/O モード / アイドルモード / リピートモードの選択が可能

#### (2) フルアドレスモード

- ・転送元、転送先アドレスを 24 ビットで指定
- ・最大 2 チャンネルを使用可能
- ・ノーマルモード / ブロック転送モードの選択が可能

16M バイトのアドレス空間を直接指定可能

転送単位をバイト / ワードに設定可能

起動要因は、内部割込み、外部リクエスト、オートリクエスト (転送モードに依存)

- ・16 ビットタイマのコンペアマッチ / インพุットキャプチャ割込み × 3
- ・シリアルコミュニケーションインタフェース (SCI チャンネル 0) の送信データエンブティ割込み、受信データフル割込み
- ・外部リクエスト
- ・オートリクエスト
- ・A/D 変換器の変換終了割込み

### 7.1.2 ブロック図

DMAC のブロック図を図 7.1 に示します。

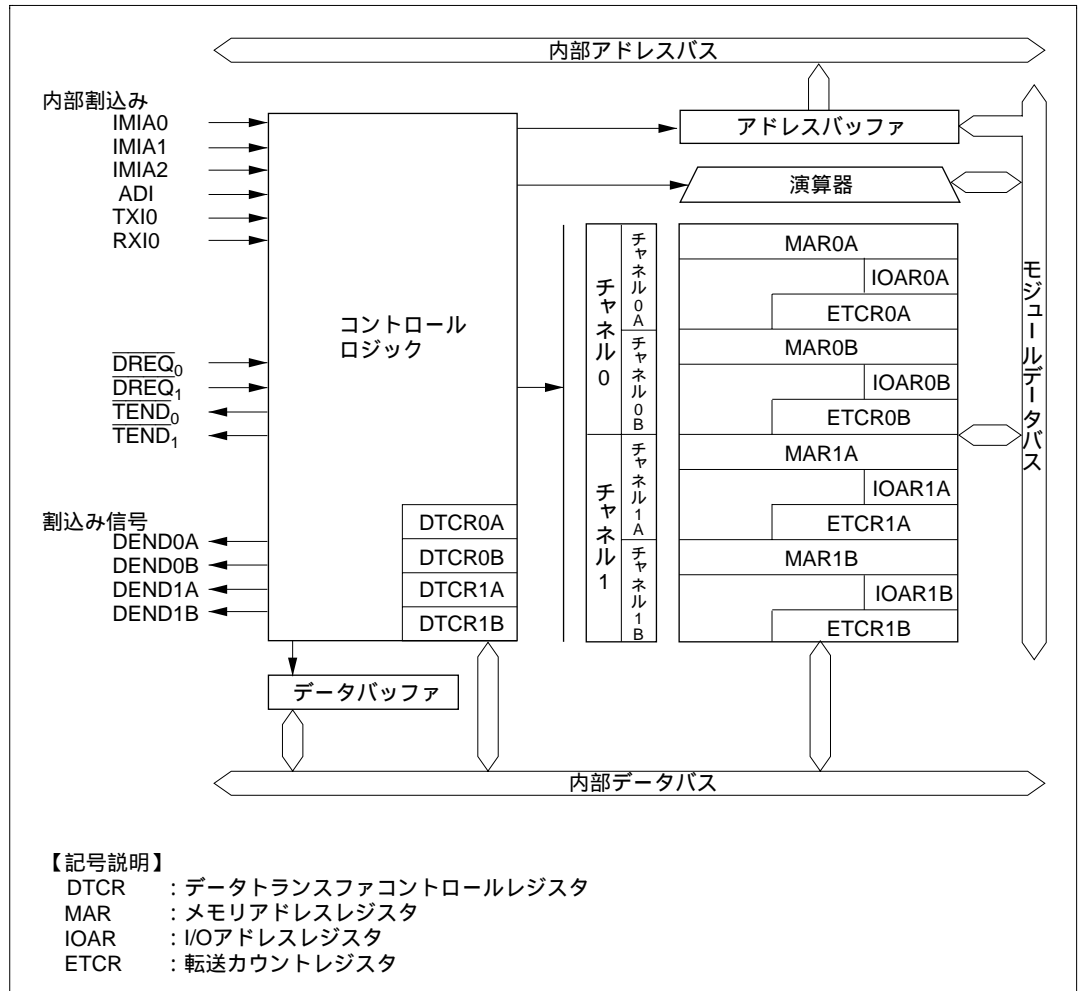


図 7.1 DMAC のブロック図

## 7.1.3 機能概要

DMAC の機能概要を表 7.1 (1)、(2) にそれぞれ示します。

表 7.1 (1) DMAC の機能概要

転送モード		転送要因	アドレスレジスタビット長	
			ソース	デスティネーション
ショート アドレス モード	(1) I/O モード <ul style="list-style-type: none"> <li>・ 1 回の転送要求で 1 バイト または 1 ワードの転送を実行</li> <li>・ メモリアドレスを 1 または 2 増減</li> <li>・ 転送回数は 1 ~ 65536</li> </ul>	<ul style="list-style-type: none"> <li>・ 16 ビットタイマチャ ネル 0 ~ 2 のコンパ アマッチ / インプ トキャプチャ A 割込 み</li> <li>・ SCI チャネル 0 の送信 データエンpty割 込み</li> </ul>	24	8
	(2) アイドルモード <ul style="list-style-type: none"> <li>・ 1 回の転送要求で 1 バイト または 1 ワードの転送を実行</li> <li>・ メモリアドレスは固定</li> <li>・ 転送回数は 1 ~ 65536</li> </ul>			
	(3) リピートモード <ul style="list-style-type: none"> <li>・ 1 回の転送要求で 1 バイト または 1 ワードの転送を 実行</li> <li>・ メモリアドレスを 1 または 2 増減</li> <li>・ 指定回数 (1 ~ 255) 転送後 初期状態を回復して動作を 継続</li> </ul>	<ul style="list-style-type: none"> <li>・ 外部リクエスト</li> </ul>	24	8

表 7.1 (2) DMAC の機能概要

転送モード		転送要因	アドレスレジスタビット長	
			ソース	デスティネーション
フルアドレスモード	(1) ノーマルモード (a) オートリクエスト ・ 転送要求を内部保持 ・ 指定回数 (1 ~ 65536) 継続して転送 ・ バーストモード / サイクルスチールモードを選択可能 (b) 外部リクエスト ・ 1 回の転送要求で 1 バイトまたは 1 ワードの転送を実行 ・ 転送回数は 1 ~ 65536	・ オートリクエスト ・ 外部リクエスト	24	24
	(2) ブロック転送モード ・ 1 回の転送要求で指定したブロックサイズの転送 ・ 転送回数は 1 ~ 65536 ・ ソースまたはデスティネーションのいずれかをブロックエリアに指定可能 ・ ブロックサイズ 1 ~ 255 バイトまたはワード	・ 16 ビットタイマチャネル 0 ~ 2 のコンペアマッチ / インพุットキャプチャ A 割込み ・ 外部リクエスト ・ A/D 変換器の変換終了割込み	24	24

### 7.1.4 端子構成

DMAC の端子構成を表 7.2 に示します。

表 7.2 端子構成

チャンネル	名 称	略 称	入出力	機 能
0	DMA 要求 0	$\overline{DREQ}_0$	入力	DMAC チャンネル 0 の外部リクエスト
	DMA 終了 0	$\overline{TEND}_0$	出力	DMAC チャンネル 0 の転送終了
1	DMA 要求 1	$\overline{DREQ}_1$	入力	DMAC チャンネル 1 の外部リクエスト
	DMA 終了 1	$\overline{TEND}_1$	出力	DMAC チャンネル 1 の転送終了

【注】 ショートアドレスモードでは、チャンネル A に対する外部リクエストは行えません。

### 7.1.5 レジスタ構成

DMAC のレジスタ構成を表 7.3 に示します。

表 7.3 レジスタ構成 (1)

チャンネル	アドレス*	名 称	略 称	R/W	初期値
0	H'FFF20	メモリアドレスレジスタ 0AR	MAR0AR	R/W	不定
	H'FFF21	メモリアドレスレジスタ 0AE	MAR0AE	R/W	不定
	H'FFF22	メモリアドレスレジスタ 0AH	MAR0AH	R/W	不定
	H'FFF23	メモリアドレスレジスタ 0AL	MAR0AL	R/W	不定
	H'FFF26	I/O アドレスレジスタ 0A	IOAR0A	R/W	不定
	H'FFF24	転送カウントレジスタ 0AH	ETCR0AH	R/W	不定
	H'FFF25	転送カウントレジスタ 0AL	ETCR0AL	R/W	不定
	H'FFF27	データトランスファコントロールレジスタ 0A	DTCR0A	R/W	H'00
	H'FFF28	メモリアドレスレジスタ 0BR	MAR0BR	R/W	不定
	H'FFF29	メモリアドレスレジスタ 0BE	MAR0BE	R/W	不定
	H'FFF2A	メモリアドレスレジスタ 0BH	MAR0BH	R/W	不定
	H'FFF2B	メモリアドレスレジスタ 0BL	MAR0BL	R/W	不定
	H'FFF2E	I/O アドレスレジスタ 0B	IOAR0B	R/W	不定
	H'FFF2C	転送カウントレジスタ 0BH	ETCR0BH	R/W	不定
	H'FFF2D	転送カウントレジスタ 0BL	ETCR0BL	R/W	不定
	H'FFF2F	データトランスファコントロールレジスタ 0B	DTCR0B	R/W	H'00

【注】 \* アドレスは下位 20 ビットを示しています。

表 7.3 レジスタ構成 (2)

チャンネル	アドレス*	名 称	略 称	R/W	初期値
1	H'FFF30	メモリアドレスレジスタ 1AR	MAR1AR	R/W	不定
	H'FFF31	メモリアドレスレジスタ 1AE	MAR1AE	R/W	不定
	H'FFF32	メモリアドレスレジスタ 1AH	MAR1AH	R/W	不定
	H'FFF33	メモリアドレスレジスタ 1AL	MAR1AL	R/W	不定
	H'FFF36	I/O アドレスレジスタ 1A	IOAR1A	R/W	不定
	H'FFF34	転送カウントレジスタ 1AH	ETCR1AH	R/W	不定
	H'FFF35	転送カウントレジスタ 1AL	ETCR1AL	R/W	不定
	H'FFF37	データトランスファコントロールレジスタ 1A	DTCR1A	R/W	H'00
	H'FFF38	メモリアドレスレジスタ 1BR	MAR1BR	R/W	不定
	H'FFF39	メモリアドレスレジスタ 1BE	MAR1BE	R/W	不定
	H'FFF3A	メモリアドレスレジスタ 1BH	MAR1BH	R/W	不定
	H'FFF3B	メモリアドレスレジスタ 1BL	MAR1BL	R/W	不定
	H'FFF3E	I/O アドレスレジスタ 1B	IOAR1B	R/W	不定
	H'FFF3C	転送カウントレジスタ 1BH	ETCR1BH	R/W	不定
	H'FFF3D	転送カウントレジスタ 1BL	ETCR1BL	R/W	不定
	H'FFF3F	データトランスファコントロールレジスタ 1B	DTCR1B	R/W	H'00

【注】 \* アドレスは下位 20 ビットを示しています。

## 7.2 各レジスタの説明(1) (ショートアドレスモード)

ショートアドレスモード転送は、チャンネル A、B 独立に行うことができます。

表 7.4 に示すように DTCRA の DTS2A、DTS1A ビットにより各チャンネルのショートアドレスモード転送を指定します。

表 7.4 ショートアドレスモード、フルアドレスモードの設定

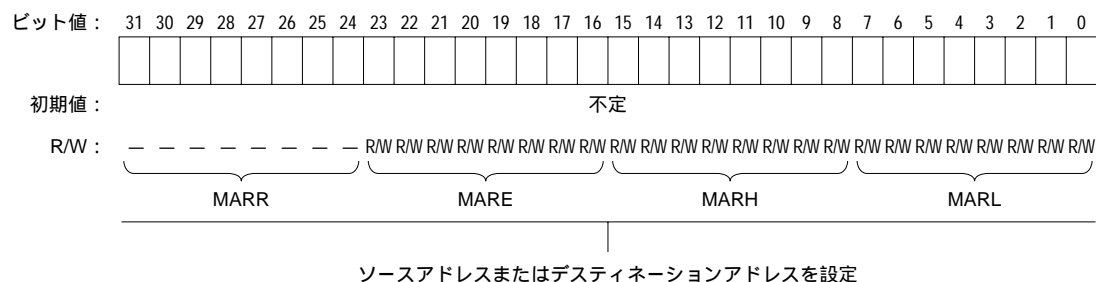
チャンネル	ビット 2	ビット 1	説 明
	DTS2A	DTS1A	
0	1	1	DMAC チャンネル 0 は、1 チャンネルのフルアドレスモード転送
	上記以外		DMAC チャンネル 0A、チャンネル 0B は、各々独立動作で 2 チャンネルのショートアドレスモード転送
1	1	1	DMAC チャンネル 1 は、1 チャンネルのフルアドレスモード転送
	上記以外		DMAC チャンネル 1A、チャンネル 1B は、各々独立動作で 2 チャンネルのショートアドレスモード転送



## 7.2.1 メモリアドレスレジスタ (MAR)

MAR は 32 ビットのリード/ライト可能なレジスタで、転送のソースアドレスまたはデスティネーションアドレスを指定します。転送方向は起動要因により自動的に決定されます。

MAR は 4 本の 8 ビットレジスタ MARR、MARE、MARH、および MARL により構成されています。MARR は全ビットリザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。



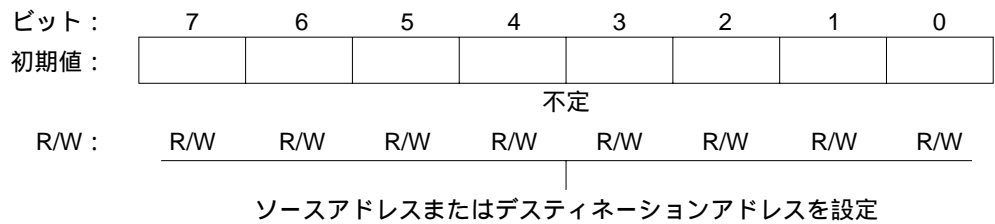
MAR がソースアドレスレジスタとして機能するか、デスティネーションアドレスレジスタとして機能するかは、起動要因によって自動的に決定されます。起動要因が SCI チャンネル 0 の受信完了割込み、および A/D 変換器の変換終了割込みの場合はデスティネーションアドレスレジスタとして、それ以外の場合にはソースアドレスレジスタとして機能します。

MAR は 1 回のバイト転送またはワード転送のたびにインクリメント/デクリメントされ、ソースまたはデスティネーションのメモリアドレスを自動的に更新していきます。詳細は、「7.3.4 データトランスファコントロールレジスタ (DTCR)」を参照してください。

MAR は、リセット、またはスタンバイモード時にイニシャライズされません。

## 7.2.2 I/O アドレスレジスタ (IOAR)

IOAR は 8 ビットのリード/ライト可能なレジスタで、転送のソースアドレスまたはデスティネーションアドレスを指定します。IOAR はアドレスの下位 8 ビットを指定し、上位 16 ビットは、すべて 1 (H'FFFF) となります。



IOAR がソースアドレスレジスタとして機能するか、デスティネーションアドレスレジスタとして機能するかは、起動要因によって自動的に決定されます。起動要因が SCI チャネル 0 の受信完了割り込み、および A/D 変換器の変換終了割り込みの場合はソースアドレスレジスタとして、それ以外の場合にはデスティネーションアドレスレジスタとして機能します。

IOAR は転送によってインクリメント/デクリメントされず、固定されます。

IOAR はリセット、またはスタンバイモード時にイニシャライズされません。

## 7.2.3 転送カウンタレジスタ (ETCR)

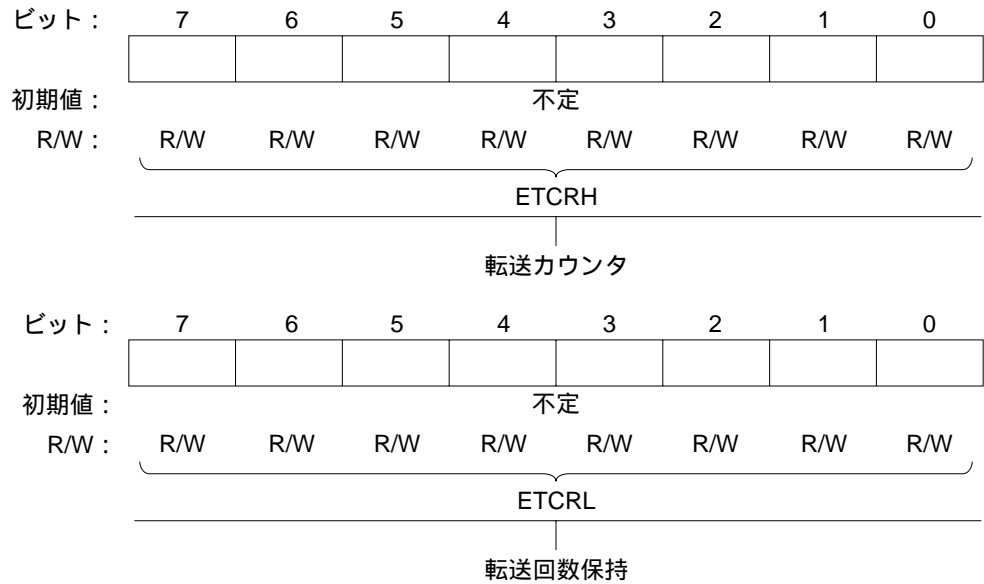
ETCR は 16 ビットのリード/ライト可能なレジスタで、転送回数の指定に使用します。このレジスタは、I/O モードおよびアイドルモードと、リピートモードとでは機能が異なります。

### (1) I/O モードおよびアイドルモード



I/O モードとアイドルモードでは、ETCR は 16 ビットの転送カウンタとして機能します。1 回の転送を行うたびに、1 だけデクリメントされカウンタ値が H'0000 になると転送を終了します。

## (2) リピートモード

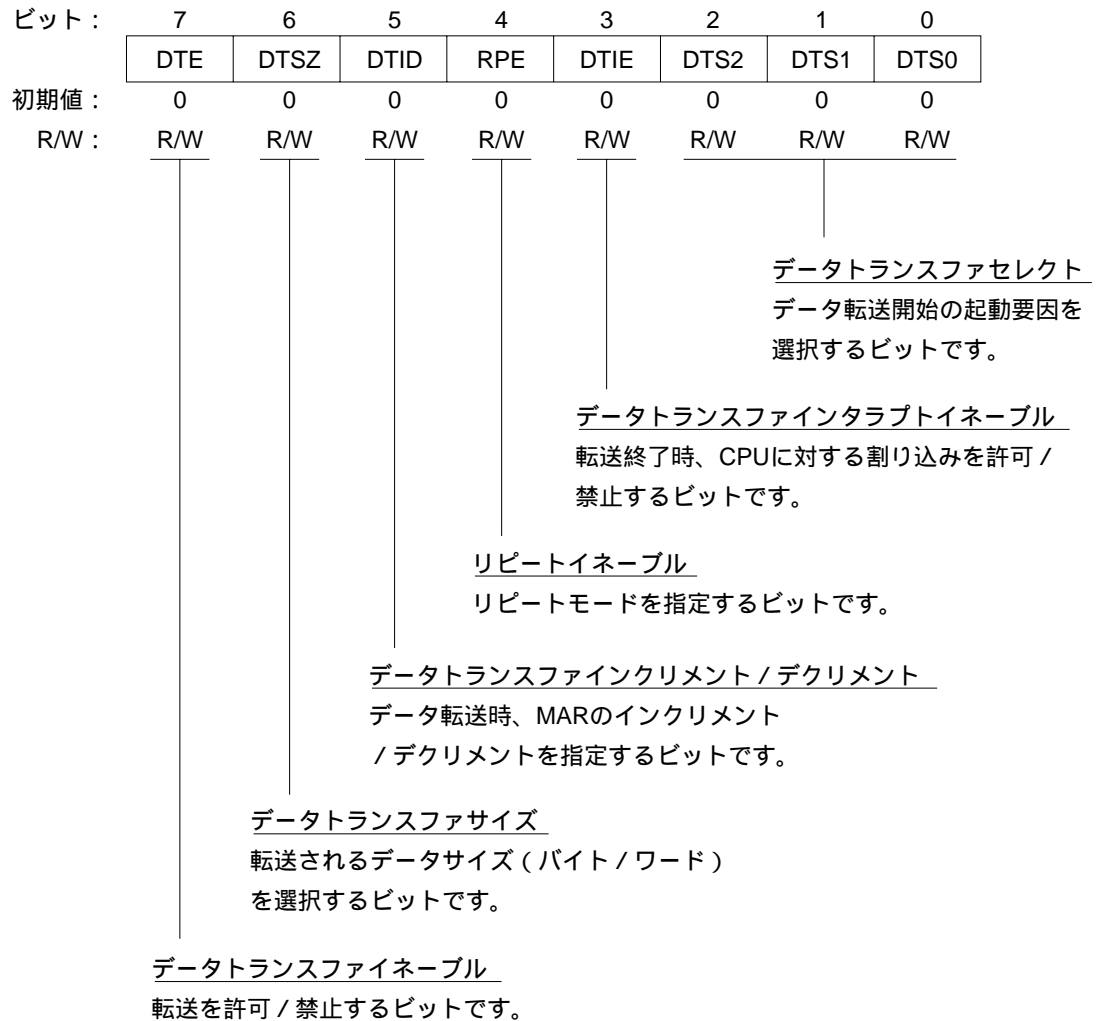


リピートモードでは、ETCRH は 8 ビットの転送カウンタとして機能し、ETCRL は転送回数を保持します。ETCRH は 1 回の転送を行うたびに 1 だけデクリメントされ、H'00 になると ETCRL の内容が転送されます。以降この動作を繰り返して転送が行われます。

ETCR は、リセット、またはスタンバイモード時にイニシャライズされません。

## 7.2.4 データトランスファコントロールレジスタ (DTCR)

DTCR は 8 ビットのリード/ライト可能なレジスタで、DMAC の各チャネルの動作を制御します。



DTCR はリセット、またはスタンバイモード時に H'00 にイニシャライズされます。

## ビット7：データトランスファイネーブル (DTE)

当該チャンネルのデータ転送を許可/禁止します。DTE ビットを1にセットすると、そのチャンネルは転送要求待ち状態となり、DTS2～DTS0 ビットで指定された起動要因によりデータ転送が行われます。本ビットが0のとき、当該チャンネルは停止状態となり転送要求を受け付けません。DTE ビットは、DTE=0の状態をリードした後、1をライトしたとき1にセットされます。

ビット7	説明
DTE	
0	データ転送禁止。I/O モードとアイドルモードでは、指定された回数の転送を終了したとき、0にクリア (初期値)
1	データ転送許可

DTIE=1の状態、本ビットが0にクリアされるとCPUに割り込みを要求します。

## ビット6：データトランスファサイズ (DTSZ)

1回に転送されるデータサイズを選択します。

ビット6	説明
DTSZ	
0	バイトサイズ転送 (初期値)
1	ワードサイズ転送

## ビット5：データトランスファインクリメント/デクリメント (DTID)

I/O モードまたはリピートモードの場合、データ転送後のMARのインクリメント/デクリメントを選択します。

ビット5	説明
DTID	
0	データ転送後MARをインクリメント (1) DTSZ=0のとき、転送後MARを+1 (2) DTSZ=1のとき、転送後MARを+2
1	データ転送後MARをデクリメント (1) DTSZ=0のとき、転送後MARを-1 (2) DTSZ=1のとき、転送後MARを-2

アイドルモードの場合、MARはインクリメントもデクリメントもされません。

## ビット4：リポートイネーブル（RPE）

データ転送をI/Oモード、アイドルモード、またはリピートモードの動作で行うかを選択します。

ビット4	ビット3	説明
RPE	DTIE	
0	0	I/Oモードで転送 (初期値)
	1	
1	0	リピートモードで転送
	1	アイドルモードで転送

I/Oモード、アイドルモード、およびリピートモードの動作については、「7.4.2 I/Oモード」、「7.4.3 アイドルモード」、「7.4.4 リピートモード」を参照してください。

## ビット3：データトランスファインタラプトイネーブル（DTIE）

DTEビットが0にクリアされたとき、DTEビットによる割込み（DEND）要求を許可/禁止します。

ビット3	説明
DTIE	
0	DTEによる割込み（DEND）を要求を禁止 (初期値)
1	DTEによる割込み（DEND）を要求を許可

## ビット2~0：データトランスファセレクト（DTS2~DTS0）

データ転送の起動要因を選択します。チャンネルAとチャンネルBでは一部指定内容が異なります。

ビット2	ビット1	ビット0	説明
DTS2	DTS1	DTS0	
0	0	0	16ビットタイマチャンネル0のコンペアマッチ / インพุットキャプチャA割込みで起動（初期値）
		1	16ビットタイマチャンネル1のコンペアマッチ / インพุットキャプチャA割込みで起動
	1	0	16ビットタイマチャンネル2のコンペアマッチ / インพุットキャプチャA割込みで起動
		1	A/D変換器の変換終了割込みで起動
1	0	0	SCIチャンネル0の送信データエンpty割込みで起動
		1	SCIチャンネル0の受信データフル割込みで起動
	1	0	$\overline{\text{DREQ}}$ 端子の立下がりエッジ入力で起動（チャンネルBの場合）
			フルアドレスモード転送を指定（チャンネルAの場合）
	1	1	$\overline{\text{DREQ}}$ 端子のLowレベル入力で起動（チャンネルBの場合）
			フルアドレスモード転送を指定（チャンネルAの場合）

【注】 \* 「7.3.4 データトランスファコントロールレジスタ（DTCR）」を参照してください。

内部割込みによる起動では、複数のチャンネル間で同一の起動要因を指定することが可能です。この場合、チャンネル間の優先順位に従い優先順位の高いチャンネルから起動されます。優先順位については、「7.4.9 DMAC複数チャンネルの動作」を参照してください。

転送許可の状態（DTE=1）では、DMACの起動要因に選択された割込みは、CPUに対して割込みを要求しません。

## 7.3 各レジスタの説明(2) (フルアドレスモード)

フルアドレスモード転送は、チャンネルAとチャンネルBを組み合わせて行います。フルアドレスモード転送の設定については、表7.4を参照してください。

### 7.3.1 メモリアドレスレジスタ (MAR)

MARは32ビットのリード/ライト可能なレジスタで、MARAは転送のソースアドレスレジスタとして、MARBはデスティネーションアドレスレジスタとして機能します。

MARは4本の8ビットレジスタMARR、MARE、MARH、およびMARLにより構成されています。MARRはすべてリザーブビットです。リードすると常に1が読み出されます。ライトは無効です。



MARは1回のバイト転送またはワード転送のたびにインクリメント/デクリメントされ、ソースまたはデスティネーションのメモリアドレスを自動的に更新することができます。詳細は、「7.3.4 データトランスファコントロールレジスタ(DTCR)」を参照してください。

MARは、リセット、またはスタンバイモード時にイニシャライズされません。

### 7.3.2 I/O アドレスレジスタ (IOAR)

IOARはフルアドレスモード転送では使用しません。



### 7.3.3 転送カウントレジスタ (ETCR)

ETCR は 16 ビットのリード/ライト可能なレジスタで、転送回数の指定に使用します。  
このレジスタは、ノーマルモードとブロック転送モードとでは機能が異なります。

#### (1) ノーマルモード

##### (a) ETCRA



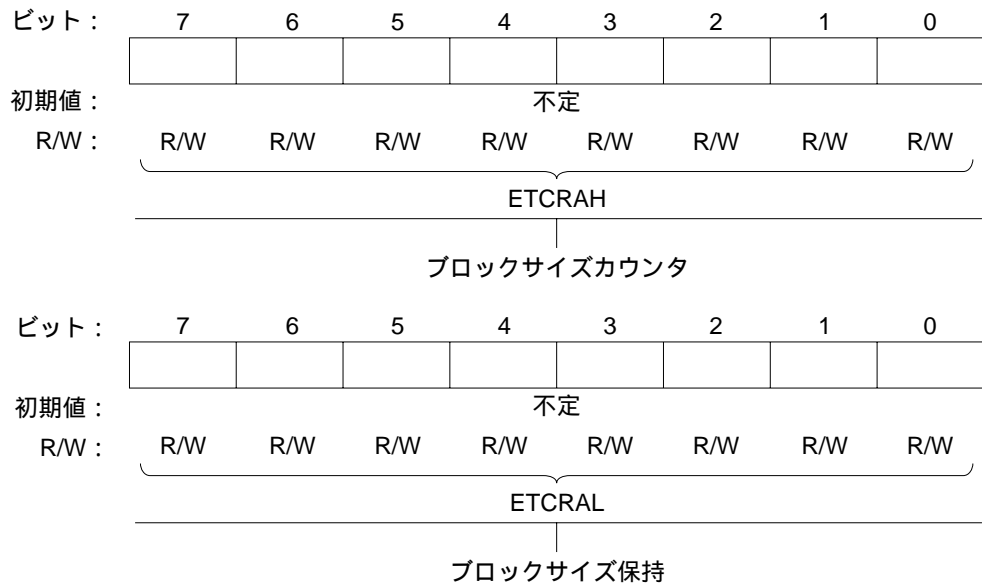
##### (b) ETCRB

ETCRB はノーマルモードでは使用しません。

ノーマルモードでは、ETCRA は 16 ビットの転送カウンタとして機能します。1 回の転送を行うたびに 1 だけデクリメントされ、カウンタ値が H'0000 になると転送を終了します。このとき、ETCRB は使用されません。

## (2) ブロック転送モード

## (a) ETCRA



## (b) ETCRB



ブロック転送モードでは、ETCRAHは8ビットのブロックサイズカウンタとして機能し、ETCRALはブロックサイズを保持します。ETCRAHは、1バイトまたは1ワードの転送を行うたびに1だけデクリメントされ、H'00になるとETCRALの内容が転送されます。したがって、ETCRAHとETCRALにブロックサイズを初期設定することにより、任意のバイト数またはワード数のブロック転送を繰り返し行うことができます。

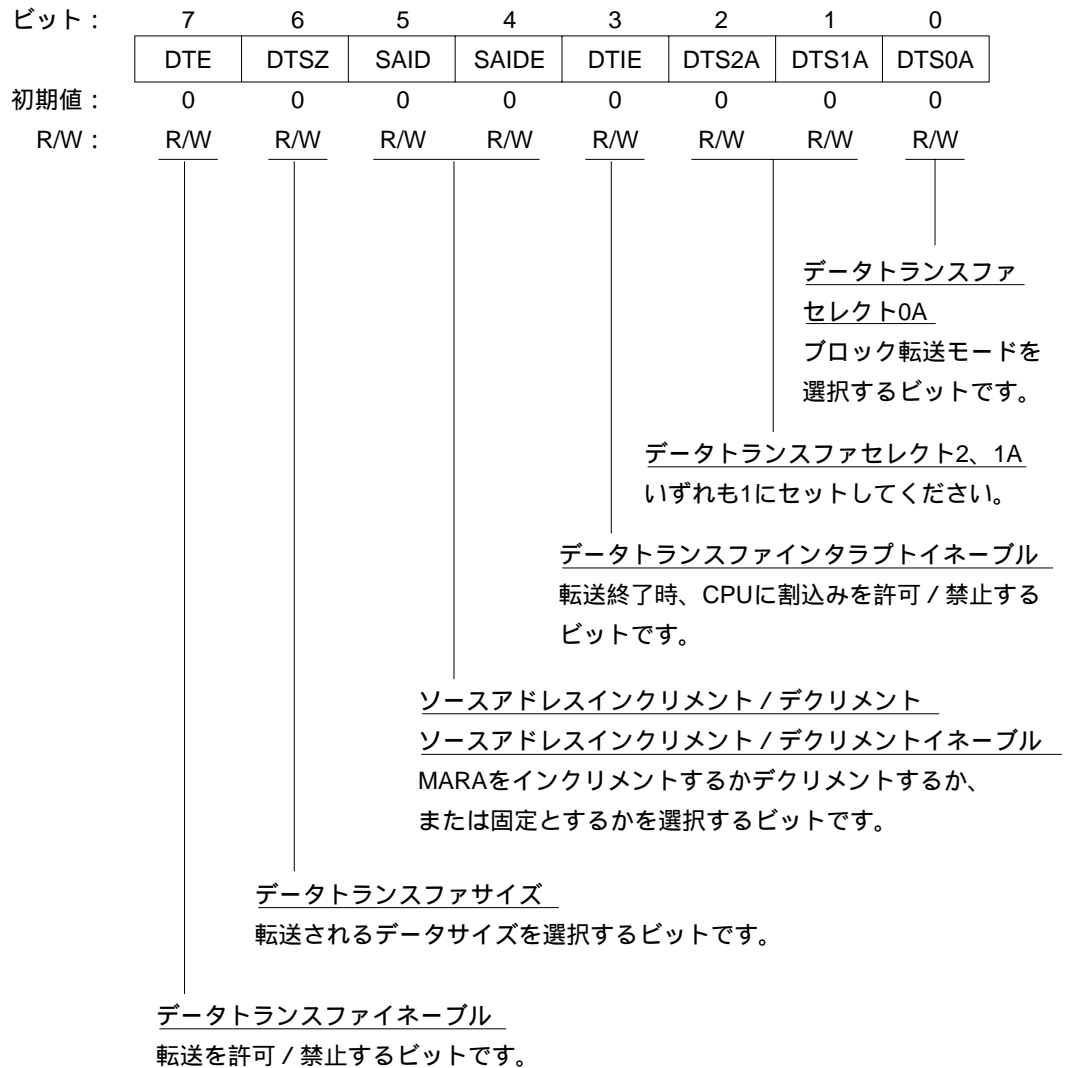
また、ブロック転送モードではETCRBは16ビットのブロック転送カウンタとして機能します。1回のブロック転送を行うたびに1だけデクリメントされ、カウンタ値がH'0000になると転送を終了します。

ETCRは、リセット、またはスタンバイモード時にはイニシャライズされません。

### 7.3.4 データトランスファコントロールレジスタ (DTCR)

DTCR は 8 ビットのリード/ライト可能なレジスタで、DMAC の各チャンネルの動作を制御します。DTCRA の DTS2A、DTS1A ビットをいずれも 1 にセットすると当該チャンネルはフルアドレスモードとなります。フルアドレスモードでは DTCRA と DTCRB では機能が異なります。

#### (1) DTCRA



DTCRA はリセット、またはスタンバイモード時に H'00 にイニシャライズされます。

## ビット7：データトランスファイネブル（DTE）

DTCRBのDTMEビットとともに当該チャネルのデータ転送を許可/禁止を制御します。DTMEビットとDTEビットをいずれも1にセットすると、そのチャネルは転送許可状態となります。オートリクエストを指定したときはただちにデータ転送を開始し、その他のときは転送要求待ち状態となります。指定された回数の転送を終了するとDTEビットは自動的に0にクリアされます。本ビットが0にクリアされているとき、当該チャネルは停止状態となり転送要求を受け付けません。DTEビットは、0の状態をリードした後、1をライトしたとき1にセットされます。

ビット7	説明
DTE	
0	データ転送禁止（指定された回数の転送を終了したとき、0にクリア） (初期値)
1	データ転送許可

DTIE=1の状態では、本ビットが0にクリアされるとCPUに割り込みを要求します。

## ビット6：データトランスファサイズ（DTSZ）

1回に転送されるデータサイズを選択します。

ビット6	説明
DTSZ	
0	バイトサイズ転送 (初期値)
1	ワードサイズ転送

ビット5：ソースアドレスインクリメント/デクリメント (SAID)

ビット4：ソースアドレスインクリメント/デクリメントイネーブル (SAIDE)

データ転送時、ソースアドレスレジスタ MARA をインクリメントするかデクリメントするか、または固定とするかを指定します。

ビット5	ビット4	説明
SAID	SAIDE	
0	0	MARA 固定 (初期値)
	1	データ転送後 MARA をインクリメント (1) DTSZ=0 のとき、データ転送後 MARA を +1 (2) DTSZ=1 のとき、データ転送後 MARA を +2
1	0	MARA 固定
	1	データ転送後 MARA をデクリメント (1) DTSZ=0 のとき、転送後 MARA を -1 (2) DTSZ=1 のとき、転送後 MARA を -2

ビット3：データトランスファインタラプトイネーブル (DTIE)

DTE ビットが0にクリアされたとき、DTE による割込み (DEND) 要求を許可/禁止します。

ビット3	説明
DTIE	
0	DTE による割込み (DEND) を要求を禁止 (初期値)
1	DTE による割込み (DEND) を要求を許可

ビット2、1：データトランスファセレクト2、1A (DTS2A、DTS1A)

DTS2A、DTS1A ビットはいずれも1にセットしたとき、当該チャネルはフルアドレスモードとなります。

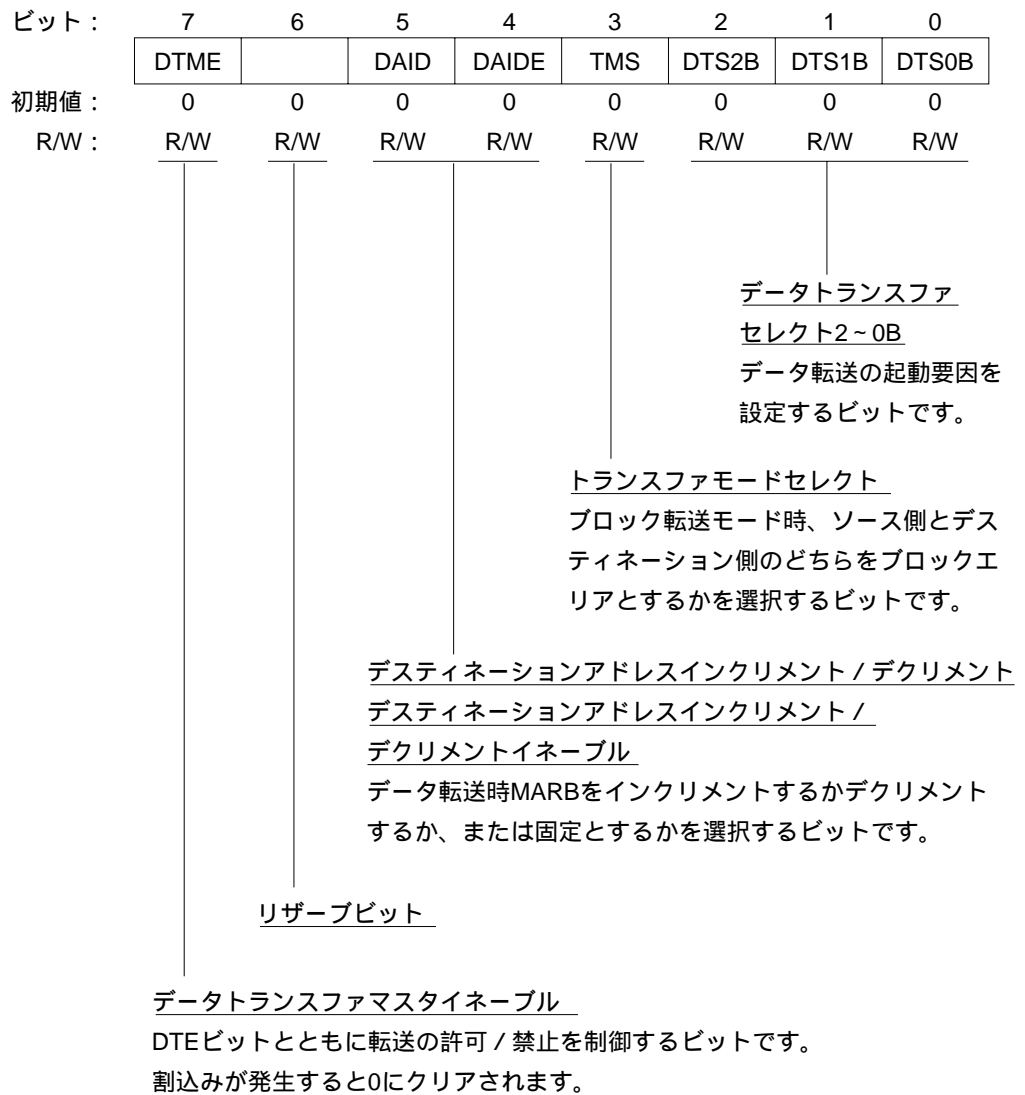
## ビット0 : データトランスファセレクト0A (DTS0A)

DMAC をノーマルモードで動作させるか、ブロック転送モードで動作させるか選択します。

ビット0	説明
DTS0A	
0	ノーマルモードの動作 (初期値)
1	ブロック転送モードで動作

ノーマルモード、ブロック転送モードの動作については、「7.4.5 ノーマルモード」、「7.4.6 ブロック転送モード」を参照してください。

(2) DTCRB



DTCRB はリセット、またはスタンバイモード時に H'00 にイニシャライズされます。

## ビット7：データトランスファマスタイネーブル (DTME)

DTCRA のDTEビットとともに当該チャンネルのデータ転送の許可 / 禁止を制御します。DTME ビットと DTE ビットをいずれも 1 にセットすると、そのチャンネルは転送許可状態となります。NMI 割込みが発生したとき DTME ビットは 0 にクリアされ、転送を中断して CPU にバス権を移します。その後、本ビットを 1 にセットすると中断された転送が再開されます。ただし、ブロック転送モード時の動作については「7.6.6 NMI 割込みとブロック転送モード」を参照してください。

DTME ビットは、DTME=0 の状態をリードした後、1 をライトしたとき 1 にセットされます。

ビット7	説明
DTME	
0	データ転送禁止。NMI 割込みが発生したとき、0 にクリア (初期値)
1	データ転送許可

## ビット6：リザーブビット

リザーブビットです。リード / ライト可能です。

## ビット5：デスティネーションアドレスインクリメント / デクリメント (DAID)

## ビット4：デスティネーションアドレスインクリメント / デクリメントイネーブル (DAIDE)

データ転送時、MARB をインクリメントするかデクリメントするか、または固定とするかを指定します。

ビット5	ビット4	説明
DAID	DAIDE	
0	0	MARB 固定 (初期値)
	1	データ転送後 MARB をインクリメント (1) DTSZ=0 のとき、データ転送後 MARB を +1 (2) DTSZ=1 のとき、データ転送後 MARB を +2
1	0	MARB 固定
	1	データ転送後 MARB をデクリメント (1) DTSZ=0 のとき、転送後 MARB を -1 (2) DTSZ=1 のとき、転送後 MARB を -2



## ビット3：トランスファモードセレクト (TMS)

ブロック転送モード時、ソース側とデスティネーション側のどちらかをブロックエリアとして転送するかを選択します。。

ビット3	説明
TMS	
0	ブロック転送モード時、デスティネーション側をブロックエリアとして転送 (初期値)
1	ブロック転送モード時、ソース側をブロックエリアとして転送

## ビット2~0：データトランスファセレクト2~0B (DTS2B~DTS0B)

データ転送の起動要因を選択します。ノーマルモードとブロック転送モードでは指定できる起動要因が異なります。

## (ノーマルモード)

ビット2	ビット1	ビット0	説明
DTS2B	DTS1B	DTS0B	
0	0	0	オートリクエスト (バーストモード) (初期値)
		1	使用できません。
	1	0	オートリクエスト (サイクルスチールモード)
		1	使用できません。
1	0	0	使用できません。
		1	使用できません。
	1	0	$\overline{\text{DREQ}}$ 端子の立下がりエッジ入力で起動
		1	$\overline{\text{DREQ}}$ 端子の Low レベル入力で起動

## (ブロック転送モード)

ビット2	ビット1	ビット0	説明
DTS2B	DTS1B	DTS0B	
0	0	0	16ビットタイマチャンネル0のコンペアマッチ/インプットキャプチャA割込みで起動 (初期値)
		1	16ビットタイマチャンネル1のコンペアマッチ/インプットキャプチャA割込みで起動
	1	0	16ビットタイマチャンネル2のコンペアマッチ/インプットキャプチャA割込みで起動
		1	A/D変換器の変換終了割込みで起動
1	0	0	使用できません。
		1	使用できません。
	1	0	$\overline{\text{DREQ}}$ 端子の立下がりエッジ入力で起動
		1	使用できません。

内部割込みによる起動では、複数のチャンネル間で同一の起動要因を指定することが可能です。この場合、チャンネル間の優先順位に従い優先順位の高いチャンネルから起動されます。優先順位については「7.4.9 DMAC複数チャンネルの動作」を参照してください。

## 7.4 動作説明

### 7.4.1 概要

DMAC のモード一覧を表 7.5 に示します。

表 7.5 モード一覧

転送モード		起動要因	備考
ショート アドレス モード	(1) I/O モード (2) アイドルモード (3) リピートモード	16 ビットタイマチャンネル 0~2 のコンペアマッチ / インพุットキ ャプチャ A 割込み	<ul style="list-style-type: none"> <li>・最大 4 チャンネルを独立に動作可能</li> <li>・外部リクエストはチャンネル B のみ可能</li> </ul>
		SCI チャンネル 0 の送信データエン プティ / 受信データフル割込み	
		A/D 変換器の変換終了割込み	
		外部リクエスト	
フル アドレス モード	(4) ノーマルモード (5) ブロック転送 モード	オートリクエスト	<ul style="list-style-type: none"> <li>・チャンネル A、B を組み合わせて最大 2 チャンネル動作可能</li> <li>・オートリクエストではバーストモード 転送 / サイクルスチールモード転送の 選択可能</li> </ul>
		外部リクエスト	
		16 ビットタイマチャンネル 0~2 の コンペアマッチ / インพุットキャ プチャ A 割込み	
		A/D 変換器の変換終了割込み	
		外部リクエスト	

各モードの動作概要を以下に示します。

#### (1) I/O モード

一回の転送要求に対して 1 バイトまたは 1 ワードずつ、指定された回数だけ転送を行います。指定された回数の転送が終了すると CPU に割込みを要求することができます。アドレスの一方は 24 ビット、他方は 8 ビットで指定します。転送方向は起動要因により自動的に決定されます。

#### (2) アイドルモード

一回の転送要求に対して 1 バイトまたは 1 ワードずつ、指定された回数だけ転送を行います。指定された回数の転送が終了すると CPU に割込みを要求することができます。アドレスの一方は 24 ビット、他方は 8 ビットで指定します。アドレスは固定になっています。転送方向は起動要因によって自動的に決定されます。

### (3) リピートモード

一回の転送要求に対して1バイトまたは1ワードずつ、指定された回数だけ転送を行います。指定された回数の転送が終了するとアドレスと転送カウンタを設定値に戻し、動作を続けます。CPU に対して割り込みは要求しません。アドレスの一方は24ビット、他方は8ビットで指定します。転送方向は起動要因により自動的に決定されます。

### (4) ノーマルモード

#### (a) オートリクエスト

レジスタ設定のみで DMAC を起動し、指定された回数の転送が完了するまで転送を続けます。転送が完了すると CPU に割り込みを要求することができます。アドレスはいずれも24ビットで指定します。

- ・サイクルスチールモード

1バイトまたはワード転送ごとにバスをいったん他のバスマスタに解放します。

- ・バーストモード

他の優先順位の高いバスマスタからのバス権要求がなければ、指定された転送が完了するまでバスを専有して転送を行います。

#### (b) 外部リクエスト

一回の転送要求に対して1バイトまたは1ワードずつ、指定された回数だけ転送を行います。指定された回数の転送が終了すると CPU に割り込みを要求することができます。アドレスはいずれも24ビットで指定します。

### (5) ブロック転送モード

一回の転送要求に対して指定されたブロックサイズのブロック転送を行い、これを転送要求のあるごとに指定された回数だけ繰り返します。一回のブロック転送が終了するたびに一方のアドレスは設定値に戻ります。指定された回数のブロック転送が終了すると CPU に割り込みを要求することができます。アドレスはいずれも24ビットで指定します。

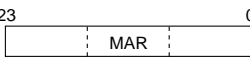
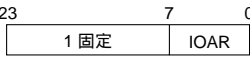

## 7.4.2 I/O モード

I/O モードは、各チャンネル独立に設定可能です。

I/O モードでは、一回の転送要求に対して 1 バイトまたは 1 ワードずつ転送を行い、これを指定された回数だけ実行します。アドレスの一方は MAR、他方は IOAR で指定します。転送方向は起動要因によって自動的に決定され、SCI チャンネル 0 の受信データフル割込みで起動される場合は IOAR で指定されるアドレスから MAR で指定されるアドレスへ、それ以外の場合は MAR で指定されるアドレスから IOAR で指定されるアドレスへ転送されます。

I/O モード時のレジスタの機能を表 7.6 に示します。

表 7.6 I/O モード時のレジスタの機能

対象レジスタ	機能		初期設定値	動作
	SCI0 受信データフル割込みによる起動	その他の起動		
	デスティネーションアドレスレジスタ	ソースアドレスレジスタ	転送先または転送元の先頭アドレス	一回の転送ごとにインクリメント/デクリメント
	ソースアドレスレジスタ	デスティネーションアドレスレジスタ	転送元または転送先のアドレス	固定
	転送カウンタ		転送回数	一回の転送ごとにデクリメント。H'0000 になると、転送終了

### 【記号説明】

MAR : メモリアドレスレジスタ

IOAR : I/O アドレスレジスタ

ETCR : 転送カウントレジスタ

転送元および転送先アドレスは、MAR と IOAR によって指定します。MAR には転送元または転送先の先頭アドレスを 24 ビットで指定します。MAR は 1 回のバイト転送またはワード転送のたびにインクリメント/デクリメントします。IOAR はアドレスの下位 8 ビットを指定し、上位 16 ビットは 1 となります。IOAR はインクリメントもデクリメントもされません。

図 7.2 に I/O モードの動作を示します。

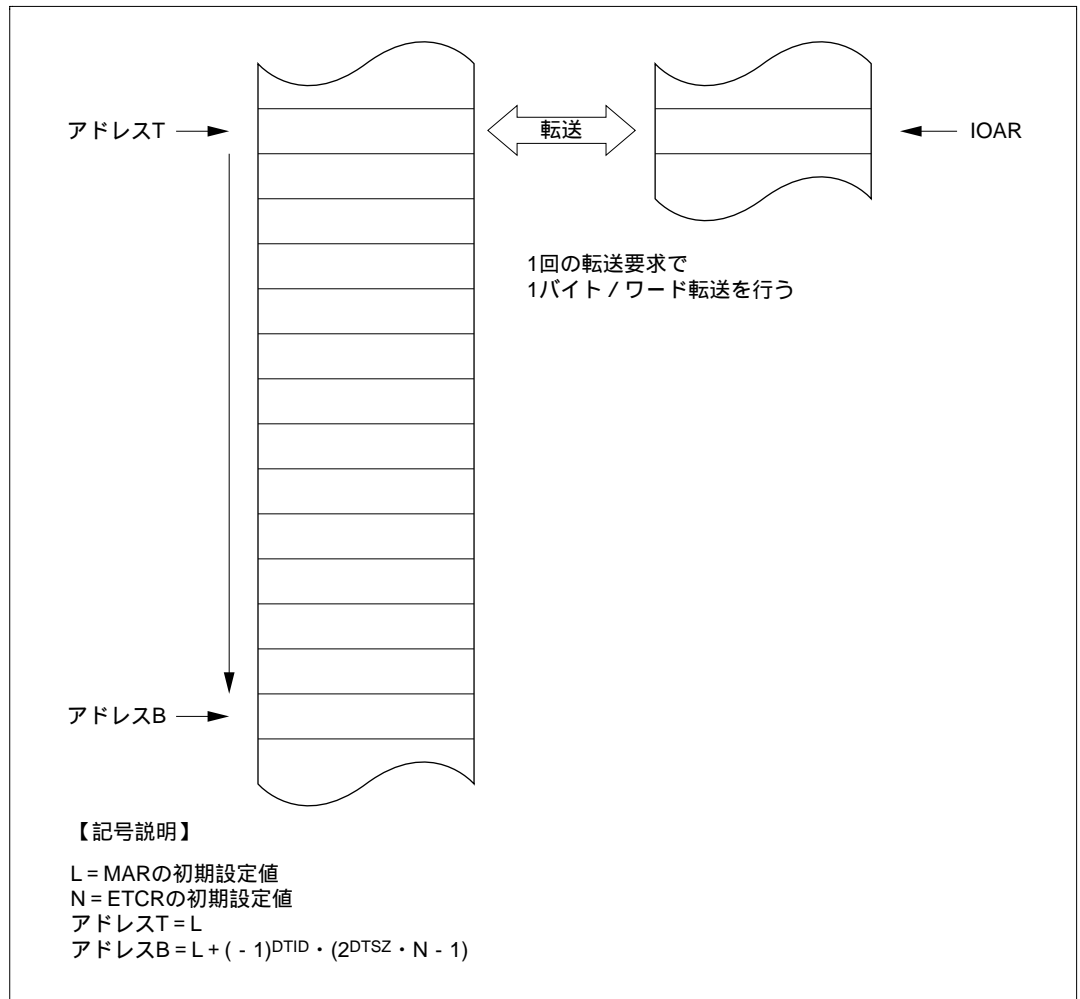


図 7.2 I/O モードの動作

転送回数はETCRによって16ビットで指定します。ETCRは一回の転送を行うたびに1だけデクリメントされ、H'0000になったときにDTEビットをクリアして転送を終了します。このとき、DTIEビットが1にセットされているとCPUに割り込みを要求します。

なお、転送回数の最大値はETCRにH'0000を設定したときで、65536となります。

転送要求（起動要因）には、16ビットタイマチャネル0~2のコンペアマッチ/インプットキャプチャA割り込み、SCIチャネル0の送信データエンプティ、受信データフル割り込み、A/D変換器の変換終了割り込み、および外部リクエストがあります。

設定の参照は「7.2.4 データトランスファコントロールレジスタ（DTCR）」を参照してください。

I/Oモードの設定手順例を図7.3に示します。

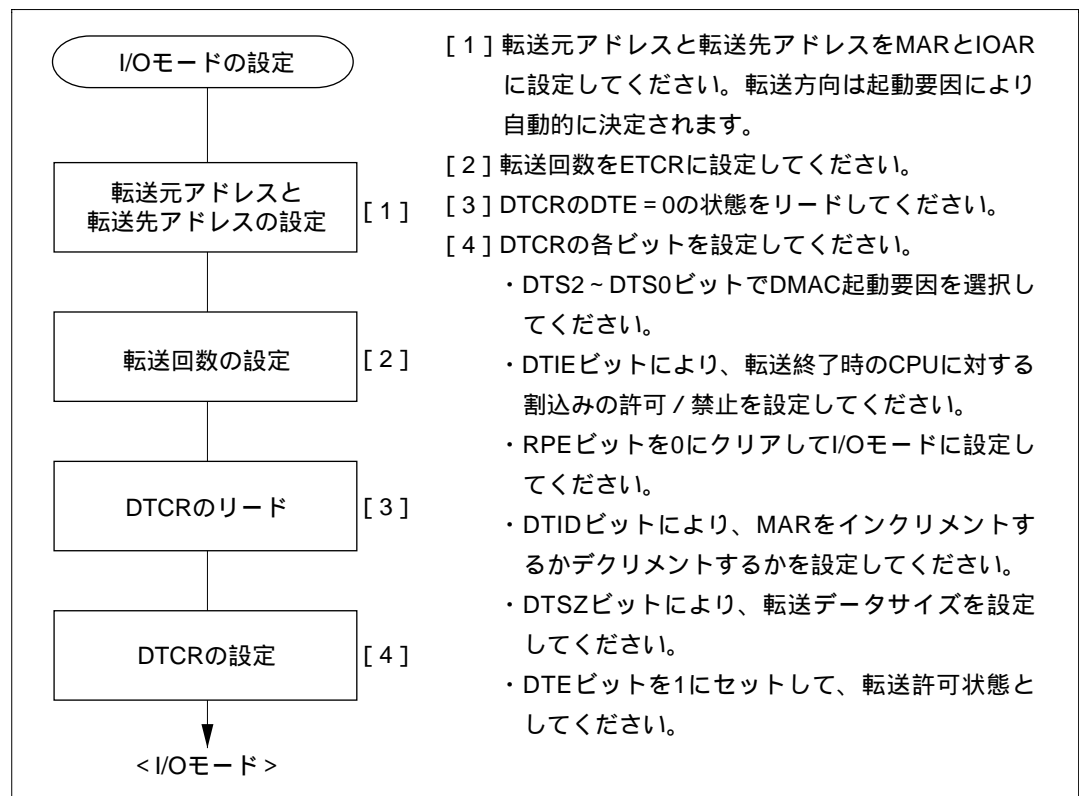


図 7.3 I/Oモードの設定手順例

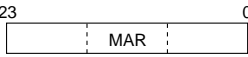
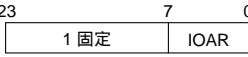
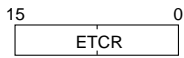
### 7.4.3 アイドルモード

アイドルモードは、各チャンネル独立に設定可能です。

アイドルモードでは、一回の転送要求に対して1バイトまたは1ワードずつ転送を行い、これを指定した回数だけ実行します。アドレスの一方はMAR、他方はIOARで指定します。転送方向は起動要因によって自動的に決定され、SCIチャンネル0の受信データフル割込みで起動される場合はIOARで指定されるアドレスからMARで指定されるアドレスへ、それ以外の場合はMARで指定されるアドレスからIOARで指定されるアドレスへ転送されます。

アイドルモード時のレジスタの機能を表7.7に示します。

表 7.7 アイドルモード時のレジスタの機能

対象レジスタ	機 能		初期設定値	動 作
	SCI0の受信データフル割込みによる起動	その他の起動		
	デスティネーションアドレスレジスタ	ソースアドレスレジスタ	転送先または転送元のアドレス	固定
	ソースアドレスレジスタ	デスティネーションアドレスレジスタ	転送元または転送先のアドレス	固定
	転送カウンタ		転送回数	一回の転送ごとにデクリメント。H'0000になると、転送終了

【記号説明】

MAR : メモリアドレスレジスタ

IOAR : I/O アドレスレジスタ

ETCR : 転送カウントレジスタ

転送元および転送先アドレスは、MARとIOARによって指定できます。MARには転送元または転送先のアドレスを24ビットで指定します。IOARはアドレス下位8ビットを指定し、上位16ビットは1となります。MAR、IOARはインクリメントもデクリメントもされません。



アイドルモードの動作を図 7.4 に示します。

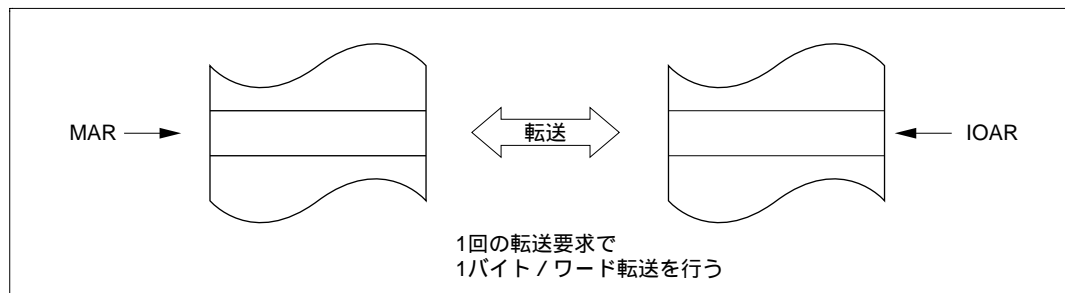


図 7.4 アイドルモードの動作

転送回数は ETCR によって 16 ビットで指定します。ETCR は一回の転送を行うたびに 1 だけデクリメントされ、H'0000 になったときに DTE ビットをクリアして転送を終了します。このとき、CPU に割り込みを要求します。

なお、転送回数の最大値は ETCR に H'0000 を設定したときで、65536 となります。

転送要求 (起動要因) には、16 ビットタイマチャンネル 0~2 のコンペアマッチ/インプットキャプチャ A 割り込み、SCI チャンネル 0 の送信データエンプティ、受信データフル割り込み、A/D 変換器の変換終了割り込み、および外部リクエストがあります。

設定の詳細は「7.3.4 データトランスファコントロールレジスタ (DTCR)」を参照してください。

アイドルモードの設定手順例を図 7.5 に示します。

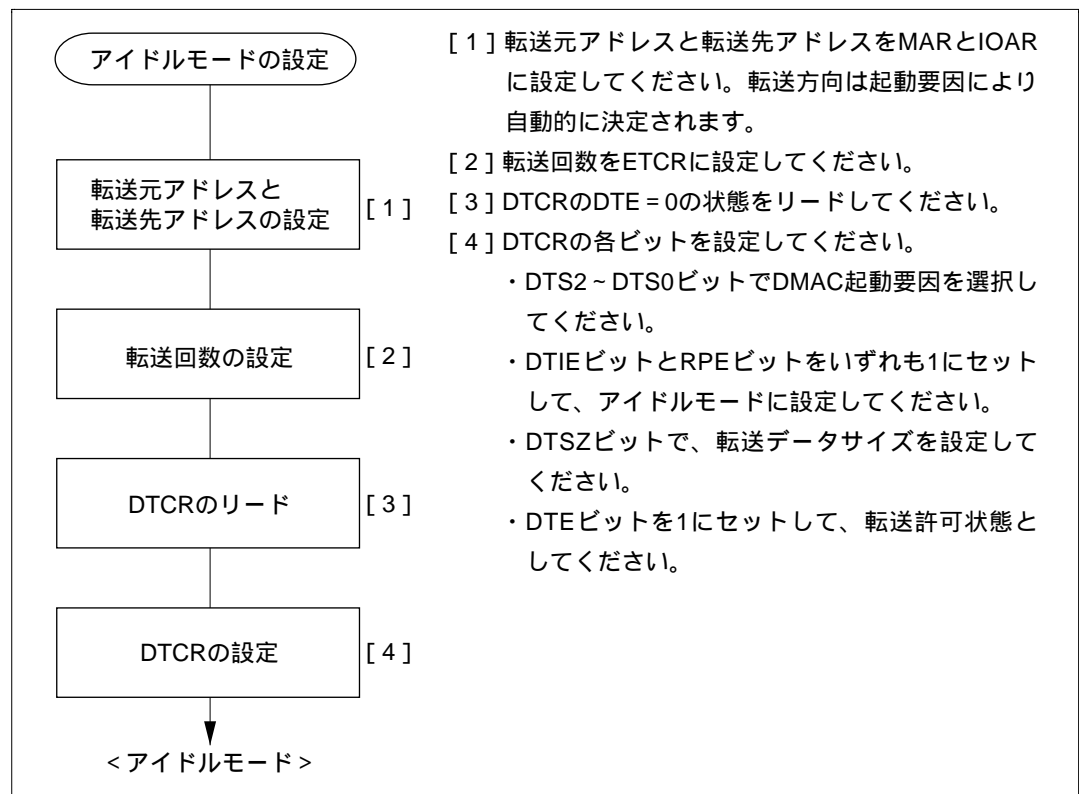


図 7.5 アイドルモードの設定手順例

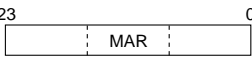
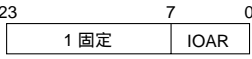
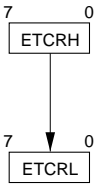
#### 7.4.4 リピートモード

リピートモードは、16ビットタイマのコンペアマッチなどに同期して、テーブル上のデータをプログラマブルタイミングパターンコントローラ（TPC）に対して繰り返し転送するのに便利なモードです。各チャンネル独立に設定可能です。

リピートモードでは、I/Oモードと同様に一回の転送要求に対して1バイトまたは1ワードずつ転送を行い、これを指定した回数だけ実行します。アドレスの一方はMAR、他方はIOARで指定します。指定された回数の転送終了時、MAR、およびETCRHの内容が初期設定値となり、さらに動作を継続します。転送方向は起動要因によって自動的に決定され、SCIチャンネル0の受信データフル割込みで起動される場合はIOARで指定されるアドレスからMARで指定されるアドレスへ、それ以外の場合はMARで指定されるアドレスからIOARで指定されるアドレスへ転送されます。

リピートモード時のレジスタの機能を表7.8に示します。

表 7.8 リピートモード時のレジスタの機能

対象レジスタ	機 能		初期設定値	動 作
	SCI0 の受信データフル割込みによる起動	その他の起動		
	デスティネーションアドレスレジスタ	ソースアドレスレジスタ	転送先または転送元の先頭アドレス	一回の転送ごとにインクリメント/デクリメント ETCRH が H'0000 になると、初期設定値を回復
	ソースアドレスレジスタ	デスティネーションアドレスレジスタ	転送元または転送先のアドレス	固定
	転送カウンタ		転送回数	一回の転送ごとにデクリメント H'0000 になると ETCRL の内容を格納
	転送回数保持		転送回数	固定

【記号説明】

MAR : メモリアドレスレジスタ

IOAR : I/O アドレスレジスタ

ETCR : 転送カウンタレジスタ

リピートモードでは ETCRH を転送カウンタとし、ETCRL は転送回数保持に使用します。ETCRH は 1 回の転送を行うたびに 1 だけデクリメントされ、H'00 になると ETCRL の値が格納されます。また、MAR は DTCR の DTSZ ビットおよび DTID ビットの値に応じて初期設定値を回復します。このときの MAR の動作は次のようになります。

$$\text{MAR} = \text{MAR} - (-1)^{\text{DTID}} \cdot 2^{\text{DTSZ}} \cdot \text{ETCRL}$$

ETCRH と ETCRL には同じ値を初期設定してください。

リピートモードでは、CPU が DTE ビットを 0 にクリアするまで転送を繰り返します。DTE ビットを 0 にクリアした後、CPU が DTE ビットを 1 にセットすると、クリアした時点の状態から転送を再開します。CPU に対して割込み要求は発生しません。

転送元および転送先アドレスは、I/O モードと同様、MAR と IOAR によって指定します。MAR には転送元または転送先アドレスを 24 ビットで指定します。IOAR にはアドレスの下位 8 ビットを指定し、上位 16 ビットは 1 となります。IOAR は転送によりインクリメントもデクリメントもされません。

図 7.6 にリピートモードの動作を示します。

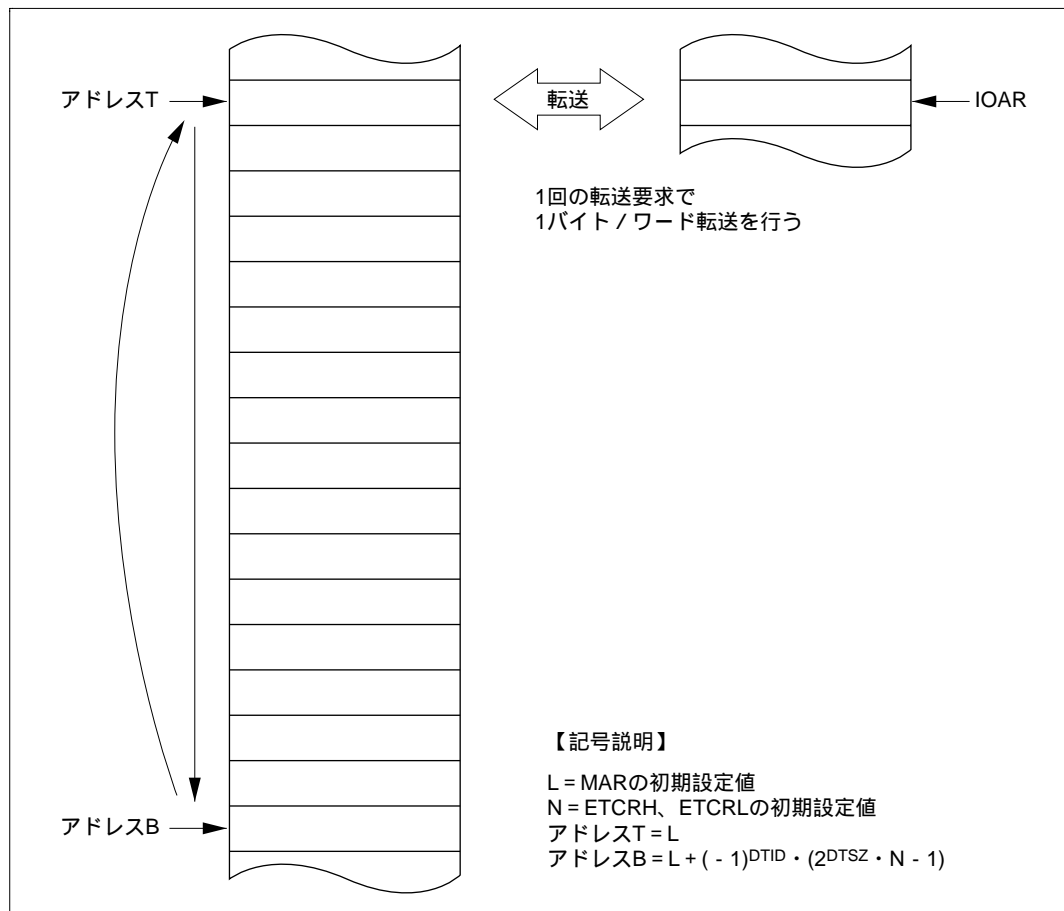


図 7.6 リピートモードの動作

転送回数は ETCRH、ETDRL に 8 ビットで指定します。転送回数の最大値は ETCRH、ETDRL にそれぞれ H'FF を設定したときで、255 となります。

転送要求（起動要因）には、16 ビットタイマチャネル 0~2 のコンペアマッチ/インプットキャプチャ A 割込み、SCI チャネル 0 の送信データエンプティ、受信データフル割込み、A/D 変換器の変換終了割込み、および 外部リクエストがあります。

設定の詳細は「7.2.4 データトランスファコントロールレジスタ (DTCR)」を参照してください。

リピートモードの設定手順例を図 7.7 に示します。

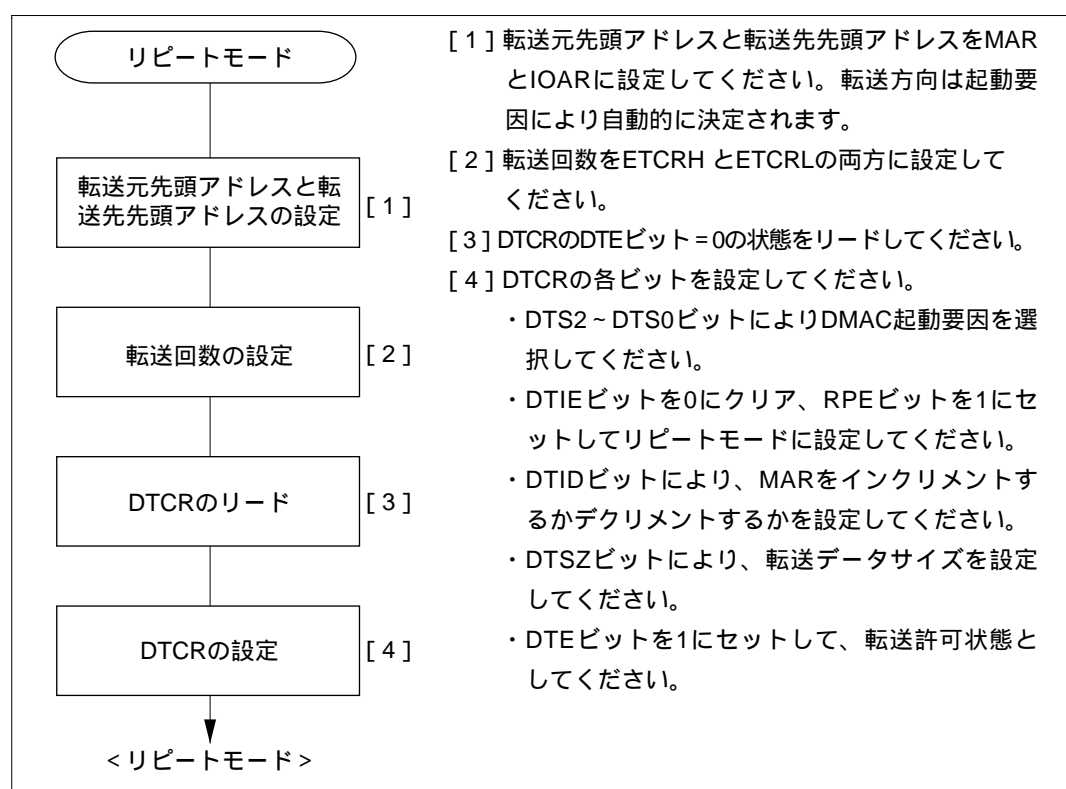


図 7.7 リピートモードの設定手順例

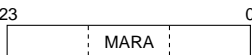
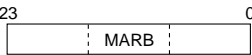
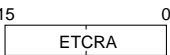
### 7.4.5 ノーマルモード

ノーマルモードは、チャンネル A、B を組み合わせて転送を行います。

ノーマルモードでは、一回の転送要求に対して 1 バイトまたは 1 ワードずつ転送を行い、これを指定された回数だけ実行します。アドレスは MARA、MARB で指定します。

ノーマルモード時のレジスタの機能を表 7.9 に示します。

表 7.9 ノーマルモード時のレジスタの機能

対象レジスタ	機能	初期設定値	動作
	ソースアドレスレジスタ	転送元先頭アドレス	一回の転送ごとにインクリメント/デクリメントまたは固定
	デスティネーションアドレスレジスタ	転送先先頭アドレス	一回の転送ごとにインクリメント/デクリメントまたは固定
	転送カウンタ	転送回数	一回の転送ごとにデクリメント

**【記号説明】**

MARA : メモリアドレスレジスタ A

MARB : メモリアドレスレジスタ B

ETCRA : 転送カウンタレジスタ A

転送元および転送先アドレスはともに 24 ビットで指定し、MARA がソースアドレスレジスタ、MARB がデスティネーションアドレスレジスタとなります。転送によるアドレスのインクリメント、デクリメントまたは固定の制御は MARA、MARB 独立に行うことができます。

転送回数は ETCRA によって 16 ビットで指定します。転送を行うたびに 1 だけデクリメントされ、H'0000 になったときに DTE ビットをクリアして転送を終了します。このとき、DTIE ビットが 1 にセットされていると CPU に割り込みを要求します。

なお、転送回数の最大値は ETCRA に H'0000 を設定したときで、65536 となります。

図 7.8 にノーマルモードの動作を示します。

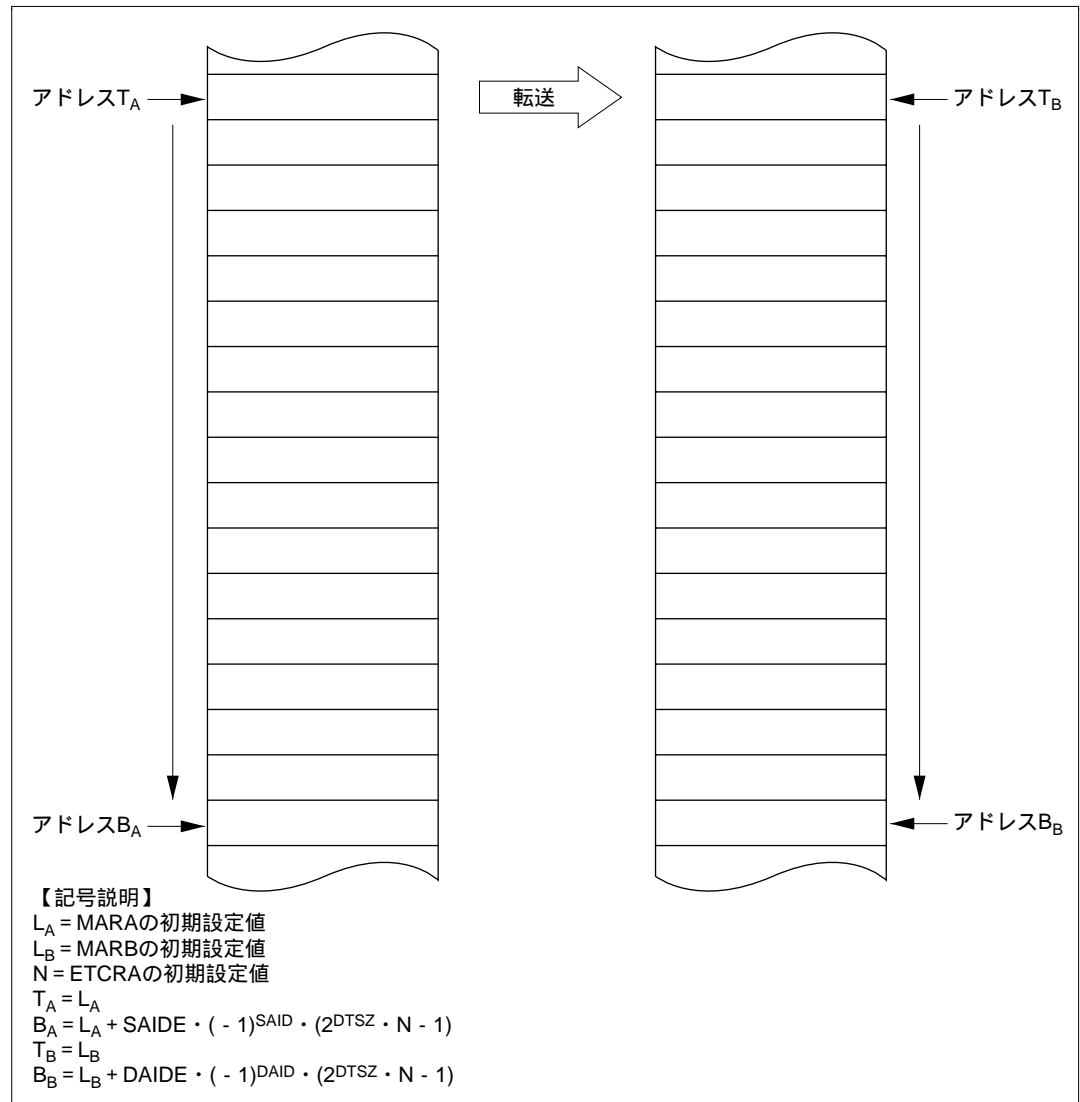


図 7.8 ノーマルモードの動作

転送要求（起動要因）には、外部リクエストとオートリクエストがあります。オートリクエストはレジスタの設定のみで起動され、指定された回数の転送を自動的に行います。オートリクエスト時にはサイクルスチールモードとバーストモードを選択できます。サイクルスチールモードでは DMAC は 1 回の転送を行うたびにバスをいったん解放します。バーストモードでは、より優先順位の高いバスマスタからバス権要求がないかぎり転送終了までバスを占有し続けます。

設定の詳細は「7.3.4 データトランスファコントロールレジスタ (DTCR)」を参照してください。



ノーマルモードの設定手順例を図 7.9 に示します。

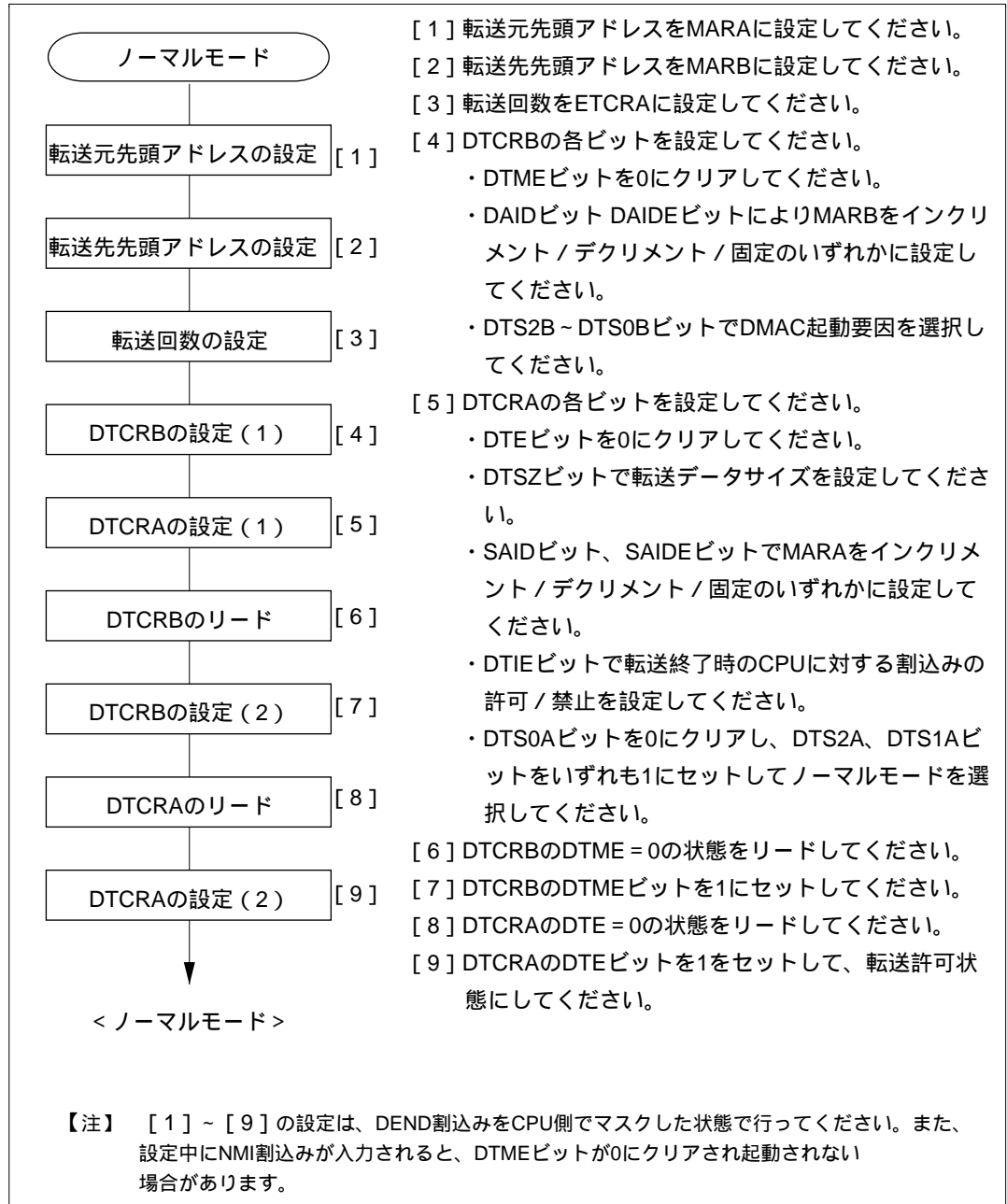


図 7.9 ノーマルモードの設定手順例

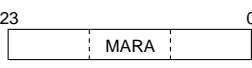
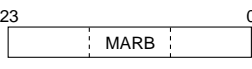


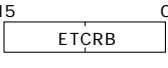
## 7.4.6 ブロック転送モード

ブロック転送モードは、チャンネル A、B を組み合わせて転送を行います。

ブロック転送モードでは、一回の転送要求に対して、指定されたブロックサイズの転送を行い、これを指定された回数だけ実行します。アドレスは MARA、MARB で指定します。ブロックエリア側のアドレスを固定とするか、連続したアドレスとするかを選択できます。

ブロック転送モード時のレジスタの機能を表 7.10 に示します。

表 7.10 ブロック転送モード時のレジスタの機能

対象レジスタ	機能	初期設定値	動作
	ソースアドレスレジスタ	転送元先頭アドレス	一回の転送ごとにインクリメント/デクリメントまたは固定
	デスティネーションアドレスレジスタ	転送先先頭アドレス	一回の転送ごとにインクリメント/デクリメントまたは固定
	ブロックサイズカウンタ	ブロックサイズ	一回の転送ごとにデクリメント、H'00 になると ETCRL の値を格納
	ブロックサイズ保持	ブロックサイズ	固定
	ブロック転送カウンタ	ブロック転送回数	ブロック転送ごとにデクリメント H'0000 になると転送終了

### 【記号説明】

MARA : メモリアドレスレジスタ A

MARB : メモリアドレスレジスタ B

ETCRA : 転送カウンタレジスタ A

ETCRB : 転送カウンタレジスタ B

転送元および転送先アドレスはともに 24 ビットで指定し、MARA ソースアドレスレジスタ、MARB がデスティネーションアドレスレジスタとなります。転送によるアドレスのインクリメント、デクリメントまたは固定の制御は、MARA、MARB 独立に行うことができます。ブロックエリアを指定する MAR は、インクリメント/デクリメントを指定した場合でも一回のブロック転送を終了するたびに初期設定値に戻ります。ソースアドレスとデスティネーションアドレスのどちらをブロックエリアとみなすかは DTCRB の TMS ビットにより指定します。

一回の転送要求で転送するブロックサイズを M (M=1 ~ 255) とし、N 回 (N=1 ~ 65,536) の転送を行うとき、ETCRAH と ETCRAL にそれぞれ M を、ETCRB に N を設定します。

図 7.10 にブロック転送モードの動作を示します。TMS ビットを 0 にクリアして、デスティネーションアドレスをブロックエリアとした場合の例です。

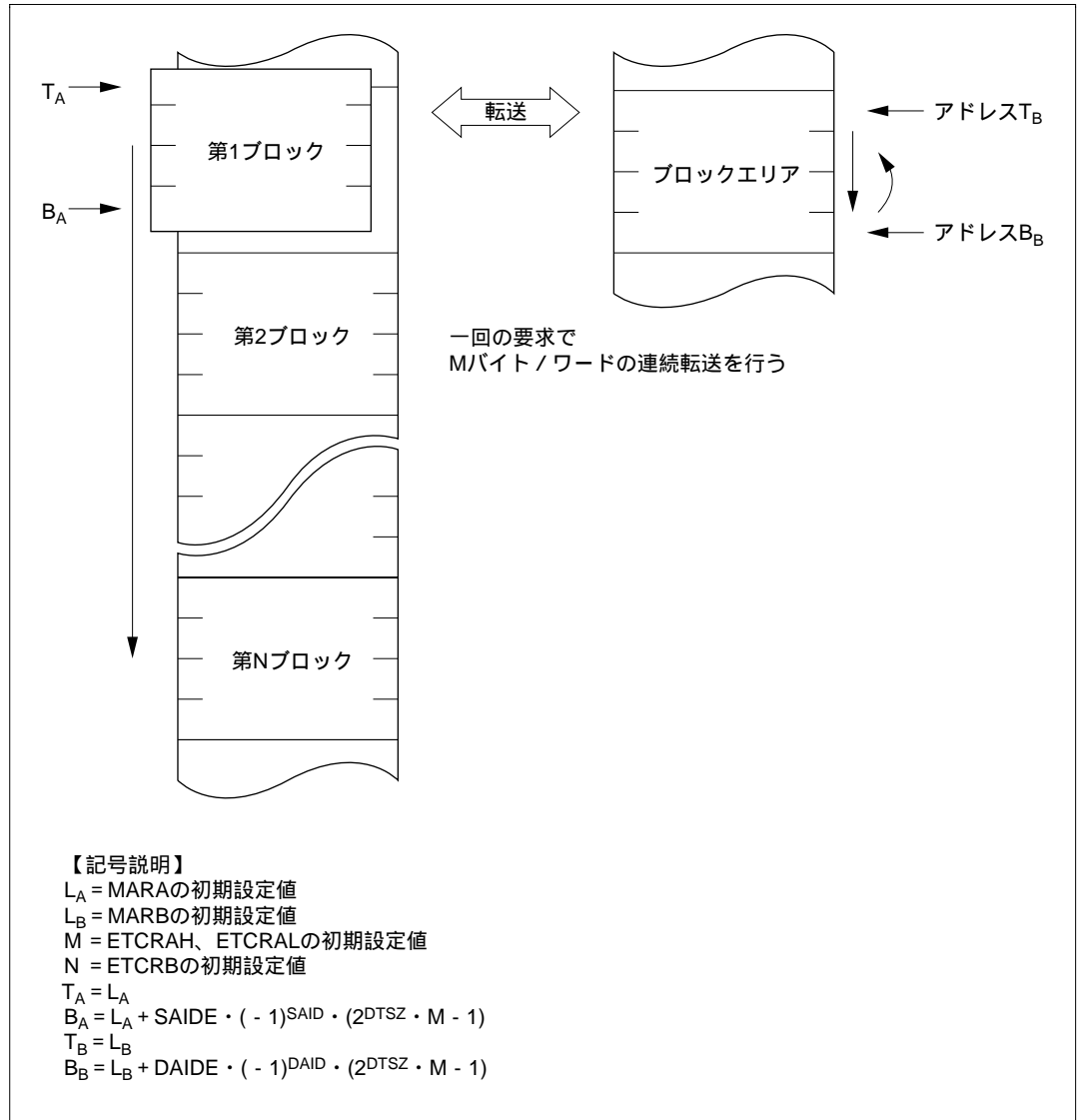


図 7.10 ブロック転送モードの動作

転送要求によって DMAC が起動されるとバースト転送を行います。この間、MARA、MARB とも DTCR の設定に従い更新され、ETCRAH をデクリメントします。ETCRAH が H'00 になると、ETCRAH は ETCRAL の値が格納され初期設定値に戻ります。同時にブロックエリア側の MAR も初期設定値に戻り、ETCRB をデクリメントして H'0000 でなければ次の転送要求待ちとなります。ETCRAH と ETCRAL には同じ値を初期設定にしてください。

この動作を繰り返して ETCRB の値が H'0000 になったとき、DTE ビットを 0 にクリアして転送を終了します。このとき、DTIE ビットが 1 にセットされていると CPU に対して割り込みを要求します。

デスティネーションアドレスをブロックエリアとしてバイト単位でブロック転送する場合の DMAC の動作フロー例を図 7.11 に示します。(a) はブロックエリアのアドレスが連続する場合、(b) はブロックエリアのアドレス固定の場合を示します。

転送要求 (起動要因) には、16 ビットタイマチャネル 0~2 コンペアマッチ/インプットキャプチャ A 割り込み、A/D 変換器の変換終了割り込み、および外部リクエストがあります。

設定の詳細は「7.3.4 データトランスファコントロールレジスタ (DTCR)」を参照してください。

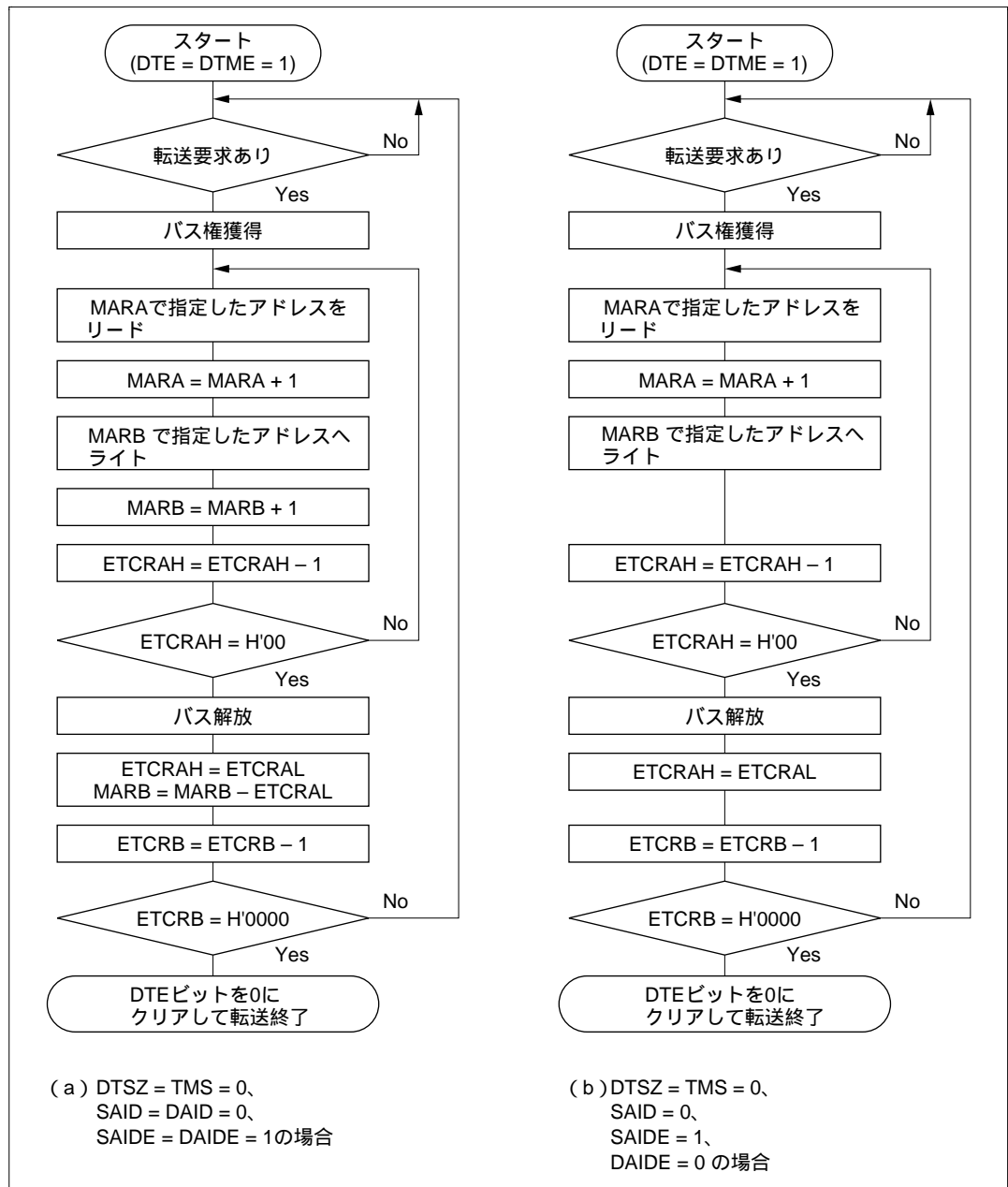


図 7.11 ブロック転送モードの動作フロー

ブロック転送モードの設定手順例を図 7.12 に示します。

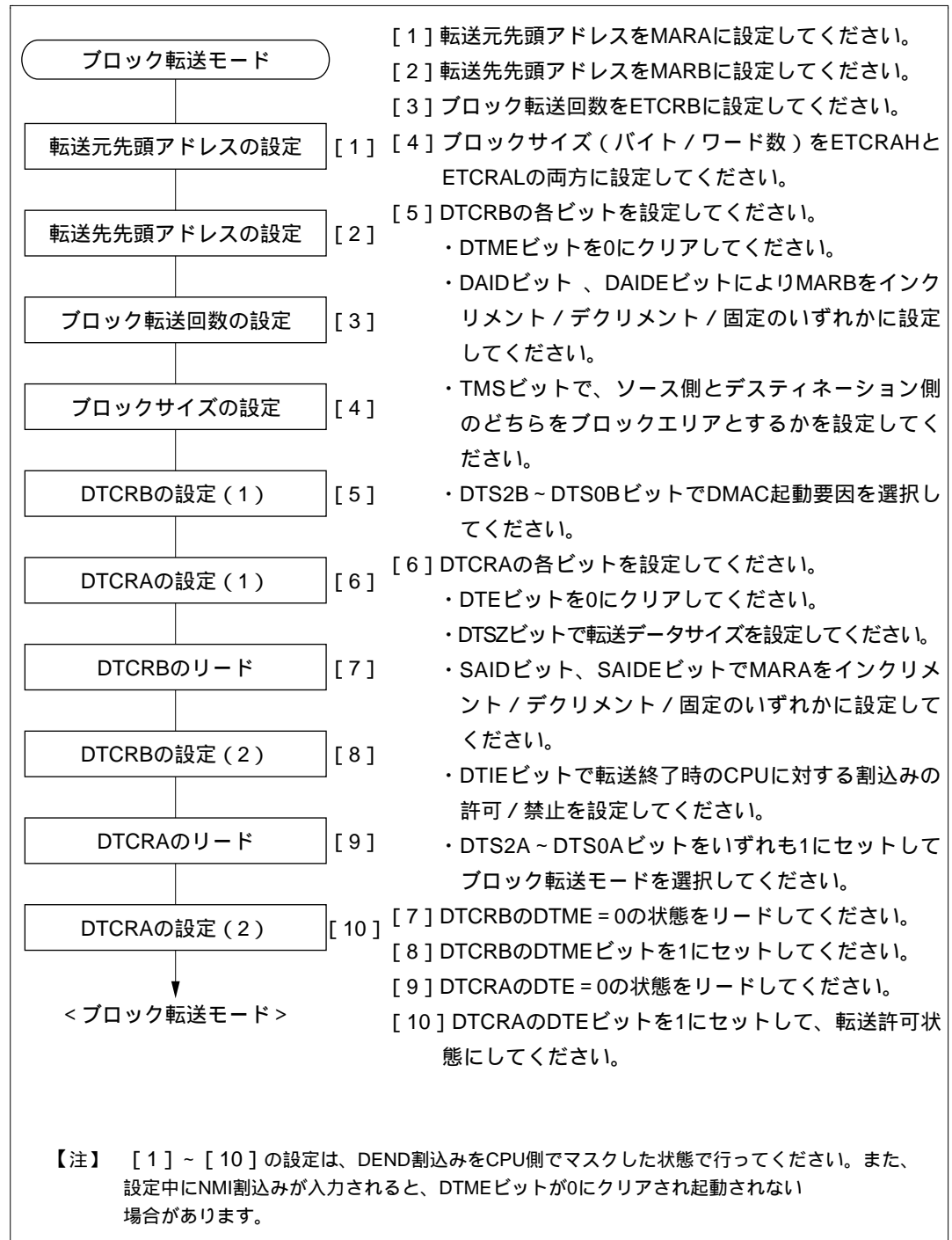


図 7.12 ブロック転送モードの設定手順例

### 7.4.7 DMAC の起動要因

DMAC の起動要因には、内部割込み、外部リクエスト、およびオートリクエストがあります。転送モードおよびチャンネルにより指定できる要因が表 7.11 に示すように異なります。

表 7.11 DMAC の起動要因

起動要因		ショートアドレスモード		フルアドレスモード	
		チャンネル 0A、1A	チャンネル 0B、1B	ノーマル	ブロック
内部 割込み	IMIA0			×	
	IMIA1			×	
	IMIA2			×	
	ADI			×	
	TXI0			×	×
	RXI0			×	×
外部 リクエスト	$\overline{\text{DREQ}}$ 端子の立下がり	×			
	$\overline{\text{DREQ}}$ 端子の Low レベル入力	×			×
オートリクエスト			×		×

#### (1) 内部割込みによる起動

DMAC の起動要因として選択された割込み要求は、DTE=1 の状態では CPU に対しては要求されません。したがって、起動要因として使用している割込みで同時に CPU に割込みを発生させることはできません。

割込み要求により DMAC が起動されると、割込み要求フラグは自動的にクリアされます。複数のチャンネルで同一の割込みを起動要因として指定した場合、最初に最も優先順位の高いチャンネルが起動された時点で割込み要求フラグがクリアされます。その他のチャンネルの転送要求は DMAC 内部で保持されて、優先順位に従って起動されます。

## (2) 外部リクエストによる起動

起動要因として外部リクエスト ( $\overline{\text{DREQ}}$  端子) を指定した場合は、該当する  $\overline{\text{DREQ}}$  端子と  $\overline{\text{TEND}}$  端子が対応するポートのデータディレクションレジスタ (DDR) の設定にかかわらず、それぞれ入力端子、出力端子になります。

$\overline{\text{DREQ}}$  端子入力にはレベルセンスとエッジセンスがあります。

ショートアドレスモードとノーマルモード時の外部リクエスト動作は次のようになります。

エッジセンスを選択した場合は  $\overline{\text{DREQ}}$  端子入力の High レベルから Low レベルへの変化を検出するたびに、1 バイトまたは 1 ワードの転送を行います。転送を完了前に次のエッジが入力された場合は次の転送が行われない場合があります。

レベルセンスを選択した場合は  $\overline{\text{DREQ}}$  端子が Low レベルに保持されている間は、転送終了まで転送を継続します。ただし、1 バイトまたは 1 ワードの転送を行うたびにいったんバスを解放します。転送の途中で  $\overline{\text{DREQ}}$  端子が High レベルとなった場合、転送中の 1 バイトまたは 1 ワードの転送した時点で転送を中断します。なお、 $\overline{\text{DREQ}}$  端子が Low レベルにすると、起動要因は 1 バイトまたは 1 ワードの転送が行われるまで内部で保持されています。

$\overline{\text{TEND}}$  端子は最後の転送のライトサイクル中 Low レベルとなります。

ブロック転送モード時の外部リクエスト動作は次のようになります。

ブロック転送モード時はエッジセンスの転送要求のみ可能です。 $\overline{\text{DREQ}}$  端子入力の High レベルから Low レベルへの変化を検出するたびに、指定された 1 ブロックを転送します。

$\overline{\text{TEND}}$  端子は 1 ブロック転送の最後のライトサイクル中 Low レベルとなります。

## (3) オートリクエストによる起動

オートリクエストはレジスタ設定のみで起動され、転送終了まで継続して転送を行います。

サイクルスチールモードとバーストモードが選択できます。

サイクルスチールモードでは、DMAC は 1 バイトまたは 1 ワードの転送を行うたびにバスをいったん解放しますので、通常、DMAC サイクルと CPU サイクルが交互に繰り返されます。

バーストモードでは、より優先順位の高いバス要求権がないかぎり転送終了までバスを占有し続けます。優先順位の高いバス要求があった場合は、転送中の 1 バイトまたは 1 ワードを転送した時点でバスを解放します。



## 7.4.8 DMAC のバスサイクル

DMAC の基本的なバスサイクルのタイミング例を図 7.13 に示します。この例はワードサイズで 16 ビット 2 ステートアクセス空間から 8 ビット 3 ステートアクセス空間へ転送する場合の例です。CPU から DMAC にバス権が移ると、1 サイクル ( $T_d$ ) の後、ソースアドレスのリード、デスティネーションアドレスのライトを行います。このリード、ライト動作の間に、他のバス権要求などによってバスを解放することはありません。DMAC サイクルは CPU サイクルと同様、バスコントローラの設定にしがいます。

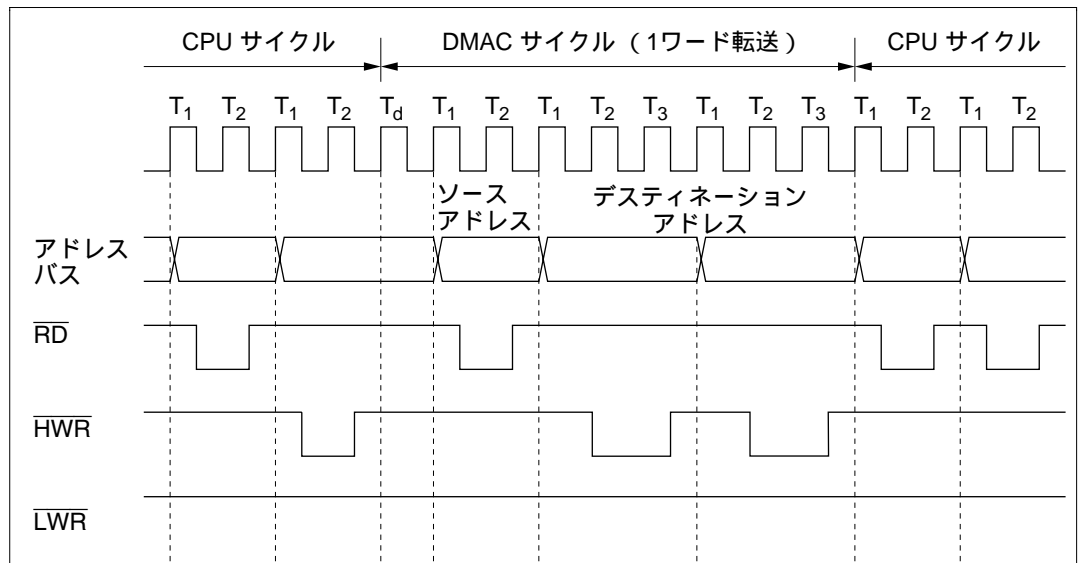


図 7.13 DMA 転送バスタイミング例

$\overline{\text{DREQ}}$  端子 Low レベルで DMAC を起動した場合のタイミングを図 7.14 に示します。ワードサイズで 16 ビット 2 ステートアクセス空間から 16 ビット 2 ステートアクセス空間へ転送する場合の例です。 $\overline{\text{DREQ}}$  端子が Low レベルに保持されている間、DMAC は転送を継続します。

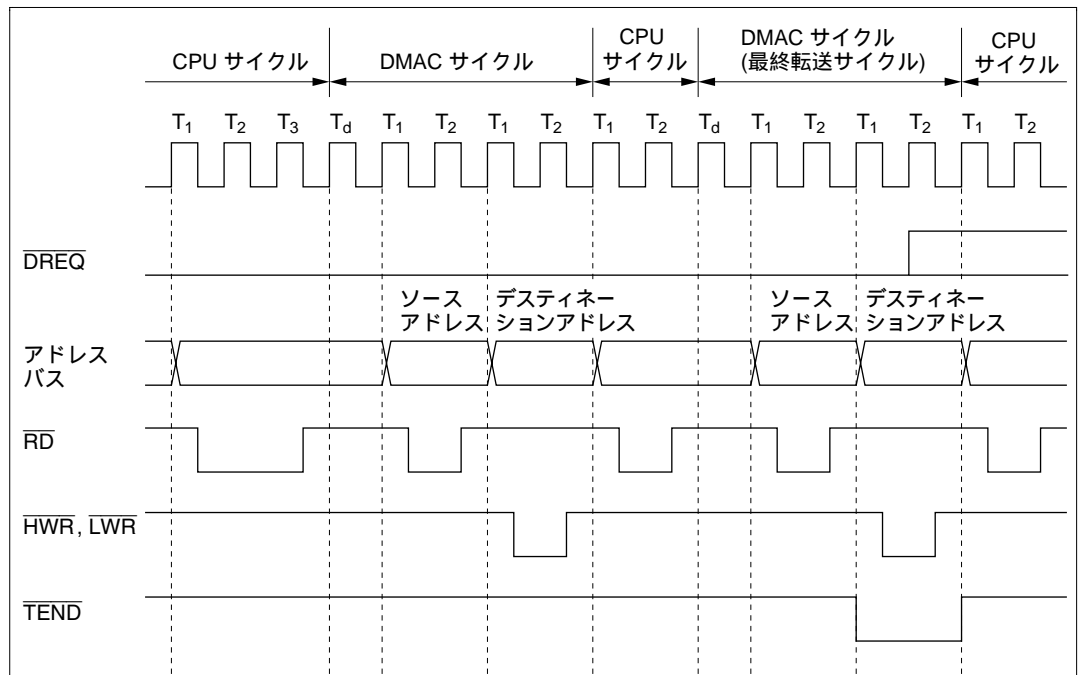


図 7.14  $\overline{\text{DREQ}}$  端子 Low レベル入力選択時の DMAC 転送バスタイミング

オートリクエストバーストモードの場合のタイミングを図 7.15 に示します。ワードサイズで 16 ビット 2 ステートアクセス空間から 16 ビット 2 ステートアクセス空間、3 ワード転送する場合の例です。

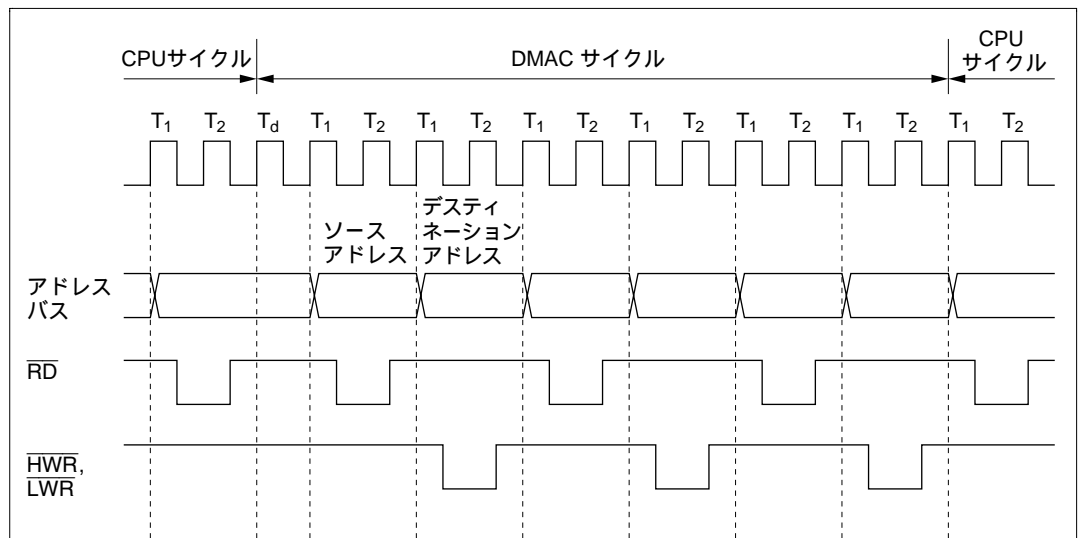


図 7.15 バーストモード DMA 転送バスタイミング

$\overline{\text{DREQ}}$  端子で DMAC を起動する場合、転送要求が発生してから DMAC が動作を開始するまでの期間は最短で 4 ステートです。

転送要求が発生後、DMAC が動作を開始し転送を行うまで、 $\overline{\text{DREQ}}$  端子のサンプリングは行われません。次のサンプリングは、ショートアドレスモードとノーマルモードの場合、リードサイクル終了後から行い、ブロック転送モードの場合、1 ブロックの転送終了後から行います。

ノーマルモード時、 $\overline{\text{DREQ}}$  端子の立下がりエッジで DMAC を起動する場合のタイミングを図 7.16 に示します。

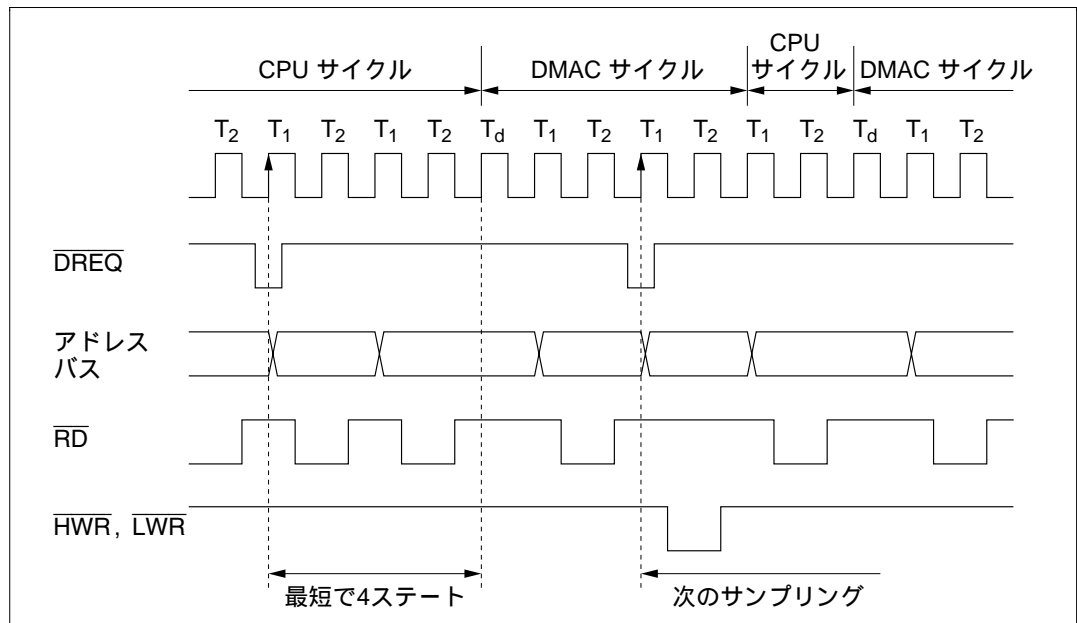


図 7.16 ノーマルモード時の  $\overline{\text{DREQ}}$  端子の立下がりエッジで DMAC 起動タイミング

ノーマルモード時、 $\overline{\text{DREQ}}$  端子の Low レベルで DMAC を起動する場合のタイミングを図 7.17 に示します。

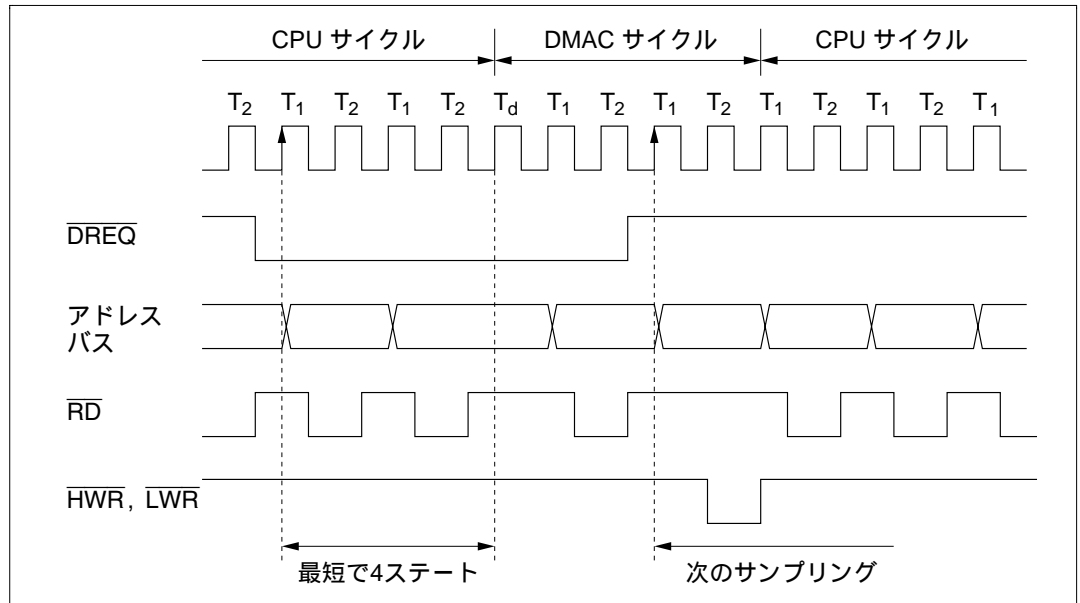


図 7.17 ノーマルモード時の  $\overline{\text{DREQ}}$  端子の Low レベルによる DMAC 起動タイミング

ブロック転送モード時、 $\overline{\text{DREQ}}$  端子立下がりエッジで DMAC を起動する場合のタイミングを図 7.18 に示します。

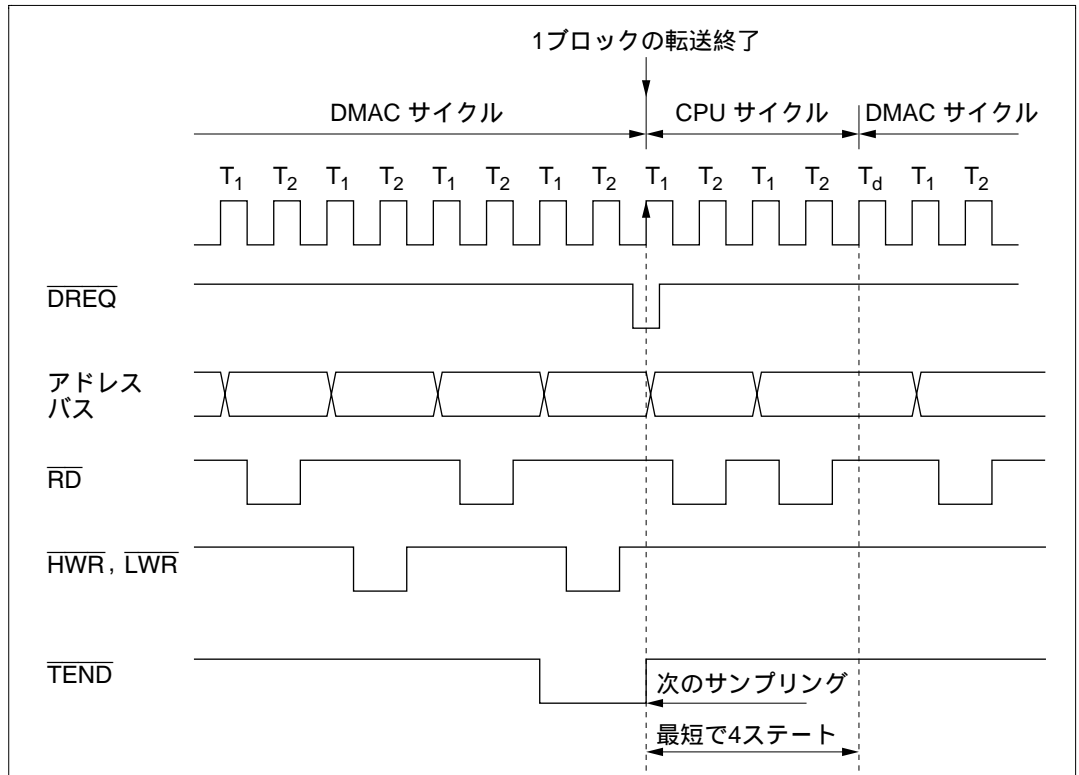


図 7.18 ブロック転送モード時の  $\overline{\text{DREQ}}$  端子の立下がりエッジによる DMAC 起動タイミング

### 7.4.9 DMAC 複数チャネルの動作

DMAC のチャネル間優先順位はチャンネル 0 > チャンネル 1、また、チャンネル A > チャンネル B の順に優先順位が高くなっています。表 7.12 に DMAC のチャネル間優先順位を示します。

表 7.12 チャネル間優先順位

ショートアドレスモード	フルアドレスモード	優先順位
チャンネル 0A	チャンネル 0	高 ↑
チャンネル 0B		
チャンネル 1A	チャンネル 1	低 ↓
チャンネル 1B		

複数のチャネルに対して同時に転送要求が発生した場合、または転送中に他のチャネルの転送要求が発生した場合、DMAC は以下のように動作します。

- (1) 転送要求が発生するとバス権を要求し、DMAC がバス権を獲得する時点で最も優先順位の高いチャンネルの転送が起動されます。
- (2) 1つのチャンネルが起動されると、そのチャンネルがバス権を解放するまで他のチャンネルは保留となります。
- (3) ショートアドレスモードおよびノーマルモードの外部リクエスト、サイクルスチールモードの場合、1回の転送を行った後、バスを解放して(1)に戻ります。バスを解放した後、他のチャンネルの転送要求が存在すると、再度バス権を要求します。
- (4) バーストの場合は転送終了後、ブロック転送モードの場合は1ブロックの転送後、バスを解放して(1)に戻ります。ただし、優先順位の高いチャンネルの転送要求または優先順位の高いバスマスタのバス権要求が存在すると、転送中の1バイトまたは1ワードの転送を終了した時点でバスを解放します。バスを解放した後、他のチャンネルの転送要求が存在すると、再度バス権を要求します。

チャンネル0AをI/Oモード、チャンネル1をバーストモードとし、チャンネル1が動作中、チャンネル0Aの転送要求が発生した場合のタイミングを図7.19に示します。

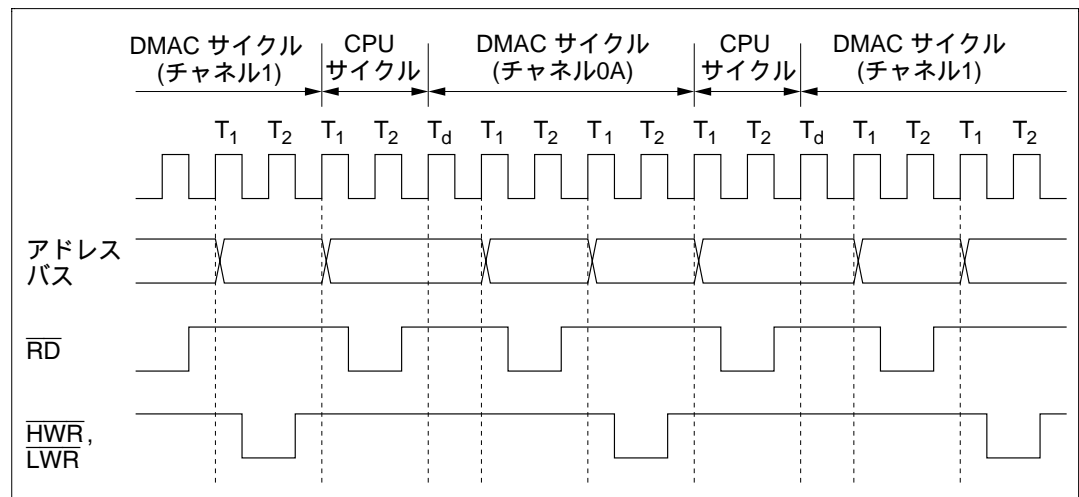


図 7.19 複数チャンネルの動作タイミング

#### 7.4.10 外部バス権要求、DRAM インタフェースと DMAC の関係

DMAC 動作中に、 $\overline{\text{BREQ}}$  端子による外部バス権要求、DRAM インタフェースによるバス権要求 (リフレッシュサイクル) があった場合、DMAC は転送中の1バイトまたは1ワードの転送を終了した時点でバスを解放します。この時点で転送要求が存在する場合、DMAC は再度バス権を要求します。

チャンネル0でバーストモード転送中にリフレッシュサイクルが挿入される場合のタイミングを図7.20に示します。

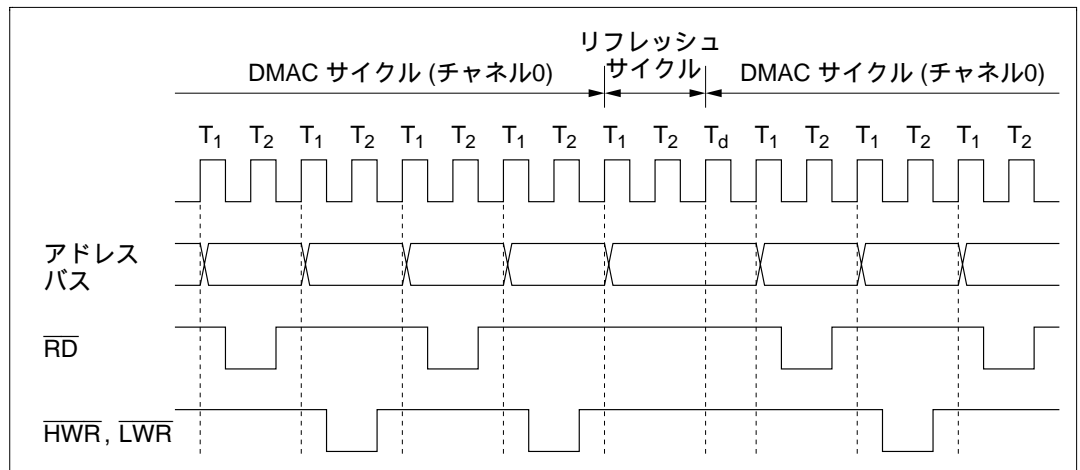


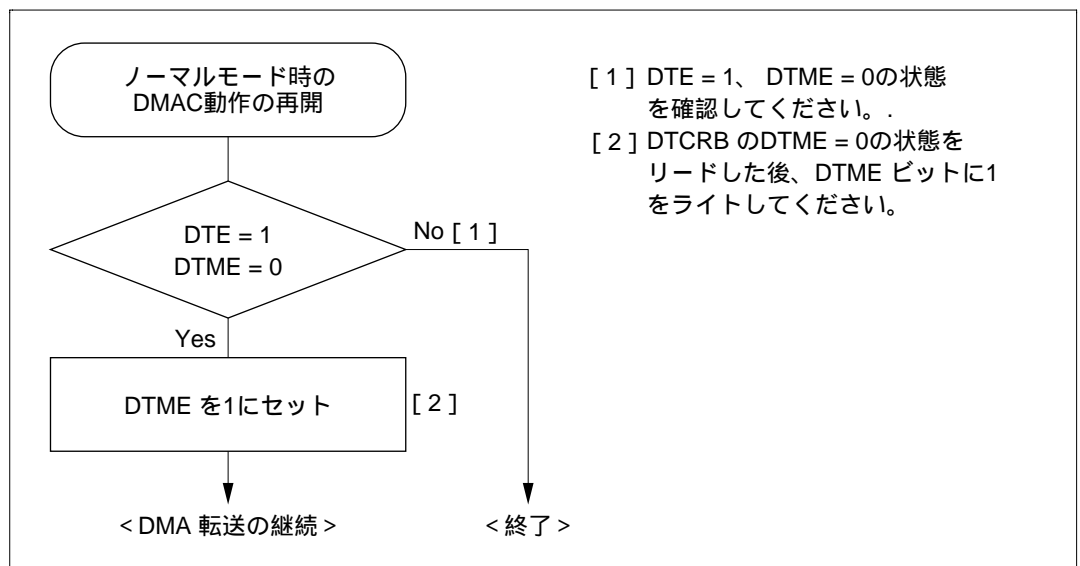
図 7.20 DRAM インタフェースと DMAC の動作タイミング

### 7.4.11 NMI 割込みと DMAC

ショートアドレスモードでは、NMI 割込みは DMAC の動作に影響を与えません。

フルアドレスモードでは、NMI 割込みが発生すると、DMAC は動作を中断します。フルアドレスモードでは、DTE ビットと DTME ビットがいずれも 1 にセットされているとき、そのチャンネルが転送許可状態となります。NMI 割込みが発生すると DTME ビットが 0 にクリアされ、DMAC は転送中の 1 バイトまたは 1 ワードの転送を完了した時点でバスを解放し、CPU にバス権が移ります。ノーマルモードのときは、その後 CPU が DTME ビットを 1 にセットすると中断した動作を再開します。この場合、事前に DTE ビットが 1 にセットされ、DTME ビットが 0 にクリアされていることを確認してください。

チャンネル 0 をノーマルモードとしたときに、NMI 割込みにより DMAC 動作が停止したとき、動作を再開する手順を図 7.21 に示します。



- [ 1 ] DTE = 1、DTME = 0の状態を確認してください。
- [ 2 ] DTCRB のDTME = 0の状態をリードした後、DTME ビットに1をライトしてください。

図 7.21 NMI 割込みにより停止した DMAC 動作再開手順例

ブロック転送モード時の NMI 割込みについては「7.6.6 NMI 割込みとブロック転送モ

ード」を参照してください。

#### 7.4.12 DMAC 動作の強制終了

動作中のチャンネルの DTE ビットを 0 にクリアすると、転送中の 1 バイトまたは 1 ワードの転送を終了した時点で DMAC は停止します。この後、DTE ビットを 1 にセットすると DMAC は動作を再開します。

フルアドレスモードの場合、DTME ビットを使用しても同様です。

DMAC をソフトウェアで強制終了させる場合の手順を図 7.22 に示します。

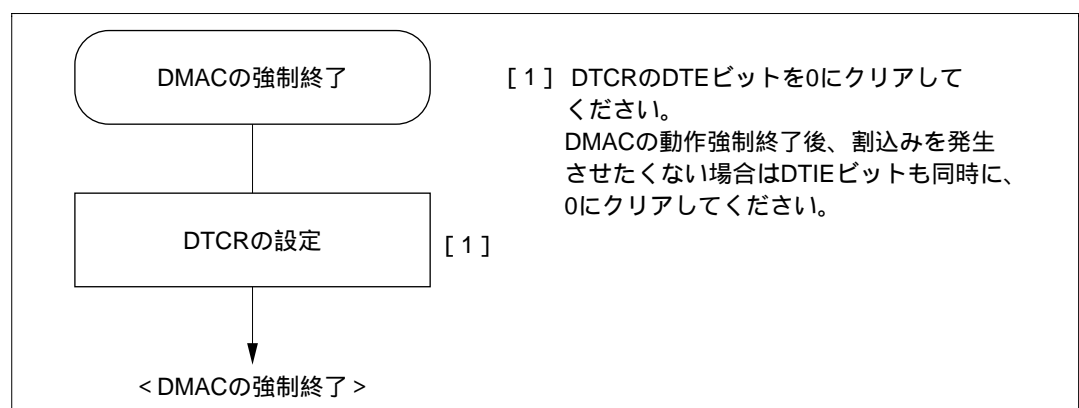


図 7.22 DMAC 動作の強制終了手順

#### 7.4.13 フルアドレスモードの解除

フルアドレスモードに設定したチャンネルを解除し、初期化する場合の手順を図 7.23 に示します。解除後に再設定する場合には各転送モードの設定手順に従ってください。

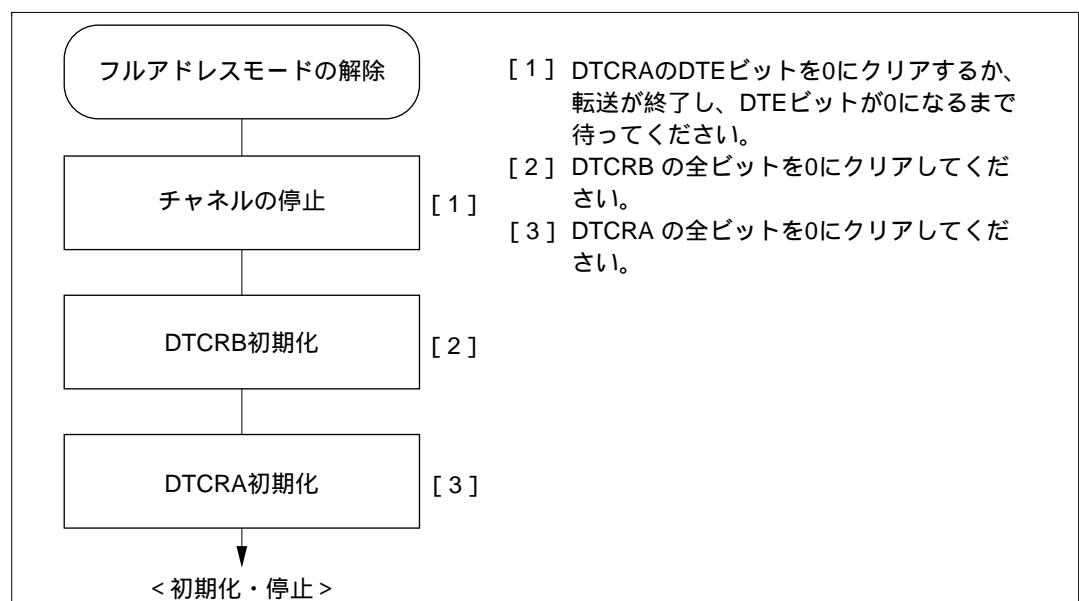


図 7.23 フルアドレスモードの解除手順例



### 7.4.14 リセット、スタンバイモード、スリープモードのDMACの状態

リセット、ハードウェアスタンバイモード、またはソフトウェアスタンバイモード時、DMACはイニシャライズされ、停止します。

スリープモード中はDMACは動作を継続します。

スリープモード中のサイクルスチールモードのタイミングを図7.24に示します。

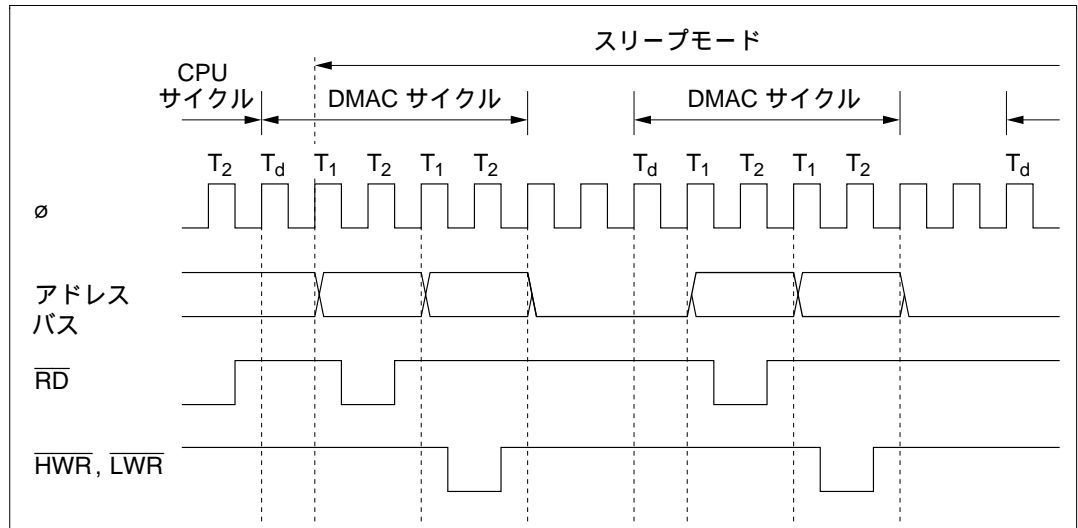


図7.24 スリープモード中のサイクルスチールモードのタイミング

## 7.5 割り込み

DMAC の割り込み要因は転送終了のみです。表 7.13 に割り込み要因と優先度を示します。

表 7.13 DMAC の割り込み要因

割り込み要因	割り込み要因		割り込み 優先順位
	ショートアドレスモード	フルアドレスモード	
DEND0A	チャンネル 0A の転送終了による割り込み	チャンネル 0 の転送終了による割り込み	高 ↑ 低
DEND0B	チャンネル 0B の転送終了による割り込み		
DEND1A	チャンネル 1A の転送終了による割り込み	チャンネル 1 の転送終了による割り込み	
DEND1B	チャンネル 1B の転送終了による割り込み		

各割り込み要因は、対応する DTICR の DTIE ビットにより許可 / 禁止が設定されており、それぞれ独立に割り込みコントローラに送られます。

チャンネル間の割り込みの優先順位は、チャンネル 0 > チャンネル 1、またチャンネル A > チャンネル B のように優先順位が高くなっています。

転送終了の割り込みブロック図を図 7.25 に示します。

DTE = 0 の状態で DTIE ビットを 1 に設定すると、常に割り込みが発生します。

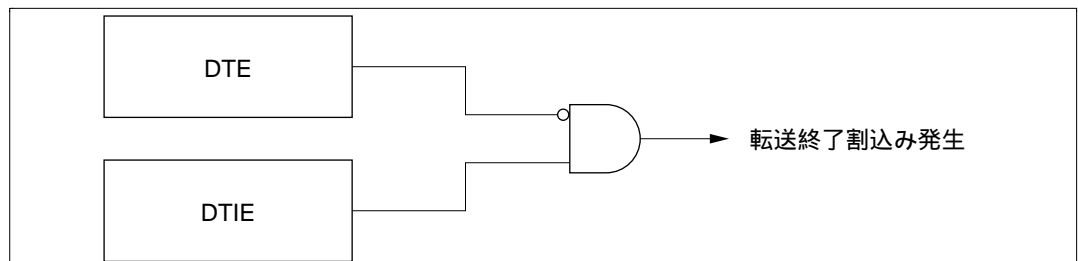


図 7.25 転送終了割り込みブロック図

フルアドレスモードでは、チャンネル B の転送終了割り込み (DENDB) は使用できません。また、DTME ビットは割り込み動作に影響を与えません。

## 7.6 使用上の注意

### 7.6.1 ワードデータ転送時の注意

奇数アドレスから始まるワードデータはアクセスできません。転送データをワードにした場合、MAR および IOAR は偶数値としてください。

### 7.6.2 DMAC による DMAC 自体のアクセス

DMAC サイクル中は DMAC 自体へのアクセスはできません。したがって、DMAC のレジスタをソースまたはデスティネーションとして転送することはできません。

### 7.6.3 MAR のロングワードアクセス

MAR は MARR から始まるロングワードデータとしてアクセスすることができます。

(例)

```
MOV. L    #LBL, ER0
```

```
MOV. L    ER0, @MARR
```

このとき、バイトデータアクセスが4回行われます。第2バイト(MARE)と第3バイト(MARH)アクセス間に、CPUがバスを解放する場合がありますので注意してください。

MAR のリード/ライトは DMAC 停止中に行ってください。

### 7.6.4 フルアドレスモード設定時の注意

フルアドレスモードは、2つのレジスタ DTCRA、DTCRB によって制御されます。これらレジスタの設定時には、チャンネル B がショートアドレスモードで動作しないように注意してください。許可ビット(DTE、DTME)は、最後に1にセットしてください。

### 7.6.5 内部割込みで DMAC を起動する場合の注意

(1) 内部割込みで DMAC を起動する場合、起動要因を選択してから DMAC を転送許可状態にするまでの期間に選択した起動要因が発生しないようにしてください。すなわち、DMAC を転送許可状態にした後、起動要因となる内蔵周辺モジュールを動作させてください。内蔵周辺モジュールの動作中に DMAC を許可状態にする場合、図 7.26 の手順で行ってください。

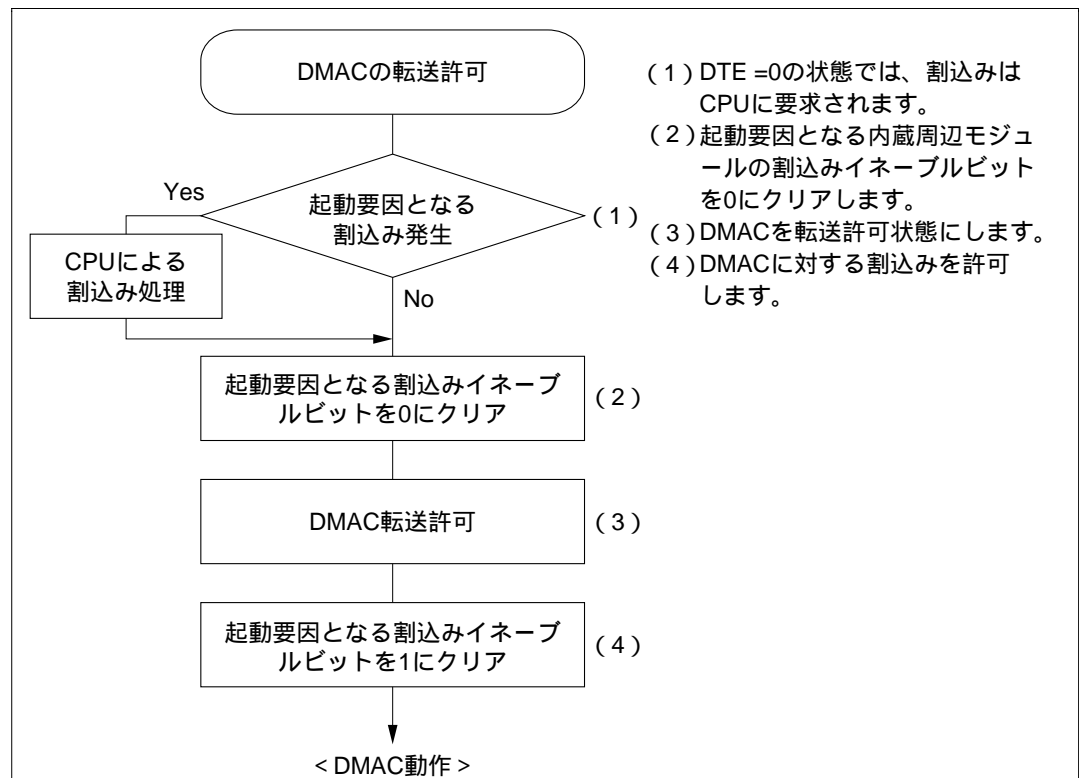


図 7.26 内蔵周辺モジュールが動作中に DMAC を転送許可状態にする場合の手順例

なお、DTE=1 なおかつ DTME=0 の状態では、DMAC は停止状態であり、また CPU にも起動要因となる割込みは要求されません。例えば、NMI 割込みにより、DMAC を停止状態にした場合は、起動要因となる割込みは CPU には要求はされません。このとき、DMAC 動作を打ち切る場合には、DTE ビットを 0 にクリアし、CPU に割込みを要求させてください。DMAC 動作を継続する場合には、DTME ビットを 1 にセットする前後で、図 7.26 の (2)、(4) の操作を行ってください。

- (2) 16 ビットタイマの割込み要求で DMAC を起動する場合、割込みによって起動される DMA 転送が終了するまで、次の割込みが発生しないようにしてください。1 つの 16 ビットタイマの割込み要求で複数チャンネルを起動する場合には、起動されるすべての DMA 転送が終了するまで、次の割込みが発生しないようにしてください。転送が終了するまでに次の割込みが発生すると、その割込みを選択しているチャンネルが以降の起動要求を受け入れなくなる場合があります。

### 7.6.6 NMI 割込みとブロック転送モード

ブロック転送モード中に NMI 割込みが発生すると DMAC は以下のように動作します。

- (1) NMI 割込みが発生すると DMAC は転送中の 1 バイトまたは 1 ワードの転送終了後、DTME ビットを 0 にクリアして停止します。したがって、1 つのブロックの転送途中停止する場合があります。
- ブロックの転送途中で停止したことは、ブロックサイズカウンタで判定できます。ブロックサイズカウンタが初期設定値以外の場合はブロックの転送途中で停止したことになります。
- (2) ブロックの転送途中で停止した場合、起動要因となる割込みフラグは 0 にクリアされています。起動要因の内部保持は行っていません。
- (3) DTE ビットが 1 にセットされ、DTME ビットを 0 にクリアされた状態では、DMAC は停止中であり、DMAC は起動要因となる割込み要求を受付けません。この状態で起動要因となる割込みが発生すると DMAC は動作せず、転送要求の内部保持も行いません。また、CPU にも割込みは要求されません。
- このため、DTME ビットを 1 にセットする前に起動要因となる割込みのイネーブルビットを 0 にクリアし、次に DTME ビットを 1 にセットし、その後、割込みイネーブルビットを 1 にセットしてください。「7.6.5 内部割込みで DMAC を起動する場合の注意」を参照してください。
- (4) DTME ビットを 1 にセットすると、DMAC は次の転送要求を待ちます。ブロックの転送途中で停止した場合、次の転送要求が発生するとブロックの残りを転送します。それ以外の場合、転送要求が発生すると次のブロックの転送を行います。

### 7.6.7 MAR、IOAR のアドレス指定

MAR、IOAR で指定できるアドレスの範囲を表 7.14 に示します。

表 7.14 MAR、IOAR で指定できるアドレスの範囲

	1Mバイトモード	16Mバイトモード
MAR	H'00000 ~ H'FFFFFF (0 ~ 1048575)	H'000000 ~ H'FFFFFF (0 ~ 16777215)
IOAR	H'FFF00 ~ H'FFFFFF (1048320 ~ 1048575)	H'FFFF00 ~ H'FFFFFF (16776960 ~ 16777215)

1Mバイトモードのとき、MAR のビット 23 ~ ビット 20 は無視されます。

### 7.6.8 転送中断時のバスサイクル

DTE ビットによる強制終了や、NMI 割込みによる DTME ビットクリアの転送停止により、DMAC 内部で、すでに要求を保持しているチャンネルを停止させるとデッドサイクルが発生することがあります。このデッドサイクルにより中断したチャンネルのアドレスレジスタおよびカウンタの値が更新されることはありません。チャンネル 0 でオートリクエストサイクルスチール転送中に、チャンネル 0 の DTE ビットをクリアした場合のタイミングを図 7.27 に示します。

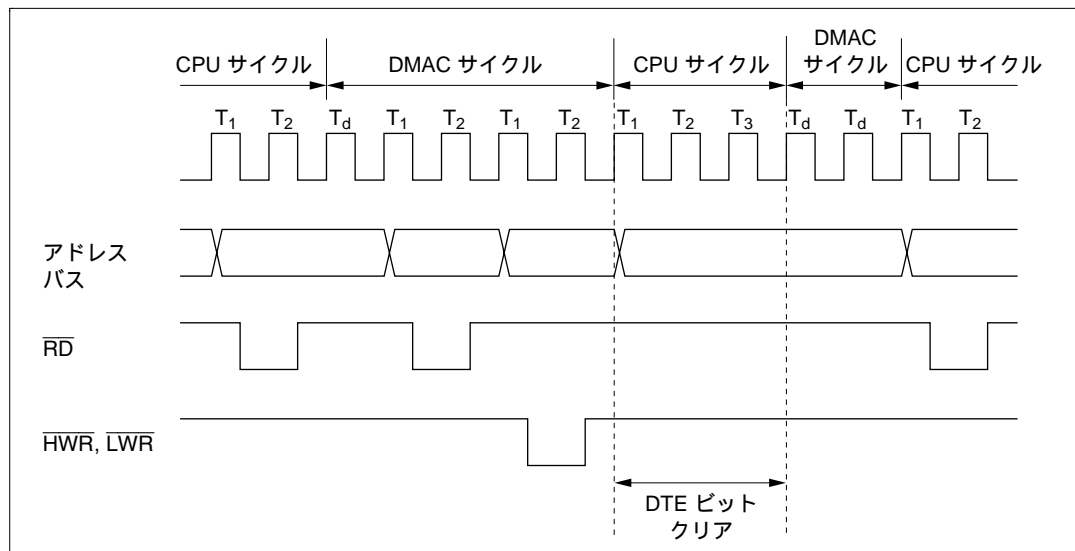


図 7.27 サイクルスチールモード DMA 転送の強制終了バスタイミング

### 7.6.9 A/D 変換器による転送要求

A/D 変換器をスキャンモードに設定し、複数チャンネルを変換している場合、A/D 変換器はすべての変換が終了した時点で転送要求を発生します。この時、変換されたデータは複数の ADDR に格納されています。したがって、すべての変換結果を一度に転送する場合は、フルアドレスモードによるブロック転送モードを使用してください。

---

# 8. I/O ポート

---

## 第8章 目次

8.1	概要	287
8.2	ポート1	292
	8.2.1 概要	292
	8.2.2 レジスタ構成	292
8.3	ポート2	295
	8.3.1 概要	295
	8.3.2 レジスタ構成	296
8.4	ポート3	299
	8.4.1 概要	299
	8.4.2 レジスタ構成	299
8.5	ポート4	302
	8.5.1 概要	302
	8.5.2 レジスタ構成	303
8.6	ポート5	306
	8.6.1 概要	306
	8.6.2 レジスタ構成	306
8.7	ポート6	310
	8.7.1 概要	310
	8.7.2 レジスタ構成	310
8.8	ポート7	315
	8.8.1 概要	315
	8.8.2 レジスタ構成	315
8.9	ポート8	317
	8.9.1 概要	317
	8.9.2 レジスタ構成	318
8.10	ポート9	324
	8.10.1 概要	324



## 8. I/Oポート

---

	8.10.2	レジスタ構成.....	325
8.11	ポートA.....		329
	8.11.1	概要.....	329
	8.11.2	レジスタ構成.....	331
8.12	ポートB.....		341
	8.12.1	概要.....	341
	8.12.2	レジスタ構成.....	343

## 8.1 概要

本LSIには、10本の入出力ポート（ポート1、2、3、4、5、6、8、9、A、B）と1本の入力専用ポート（ポート7）があります。

ポート機能一覧を表8.1に示します。表8.1に示すように、各ポートは兼用端子になっています。

各ポートは、入出力を制御するデータディレクションレジスタ（DDR）と出力データを格納するデータレジスタ（DR）から構成されています。

DDRとDRのほかに、ポート2、4、5には入力プルアップMOSコントロールレジスタ（PCR）があり、プルアップMOSのオン/オフを制御できます。

ポート1~6、8は1個のTTL負荷と90pFの容量負荷を駆動することができ、ポート9~Bは1個のTTL負荷と30pFの容量負荷を駆動することができます。

また、ポート1~6、8~Bはダーリントントランジスタを駆動することができます。ポート1、2、5はLEDを駆動（シンク電流10mA）することができます。ポートP8<sub>2</sub>~P8<sub>0</sub>、およびPA<sub>7</sub>~PA<sub>0</sub>はシュミット入力となっています。

各ポートのブロック図は「付録C I/Oポートブロック図」を参照してください。

表8.1 動作モード別ポート機能一覧

ポート	概要	端子	拡張モード					シングルチップモード	
			モード1	モード2	モード3	モード4	モード5	モード6	モード7
ポート1	<ul style="list-style-type: none"> <li>・8ビットの入出力ポート</li> <li>・LED駆動可能</li> </ul>	P1 <sub>7</sub> ~P1 <sub>0</sub> /A <sub>7</sub> ~A <sub>0</sub>	アドレス出力端子（A <sub>7</sub> ~A <sub>0</sub> ）					アドレス出力端子（A <sub>7</sub> ~A <sub>0</sub> ）と入力ポートの兼用	入出力ポート
							DDR=0のとき入力ポート	DDR=1のときアドレス出力端子	

## 8. I/O ポート

ポート	概要	端子	拡張モード					シングルチップモード	
			モード1	モード2	モード3	モード4	モード5	モード6	モード7
ポート2	<ul style="list-style-type: none"> <li>・ 8ビットの入出力ポート</li> <li>・ 入力プルアップ MOS 内蔵</li> <li>・ LED 駆動可能</li> </ul>	P2 <sub>7</sub> ~ P2 <sub>0</sub> / A <sub>15</sub> ~ A <sub>8</sub>	アドレス出力端子 (A <sub>15</sub> ~ A <sub>8</sub> )				アドレス出力端子 (A <sub>15</sub> ~ A <sub>8</sub> ) と入力ポートの兼用 DDR=0 のとき入力ポート DDR=1 のときアドレス出力端子	入出力ポート	
ポート3	<ul style="list-style-type: none"> <li>・ 8ビットの入出力ポート</li> </ul>	P3 <sub>7</sub> ~ P3 <sub>0</sub> / D <sub>15</sub> ~ D <sub>8</sub>	データ入出力端子 (D <sub>15</sub> ~ D <sub>8</sub> )					入出力ポート	
ポート4	<ul style="list-style-type: none"> <li>・ 8ビットの入出力ポート</li> <li>・ 入力プルアップ MOS 内蔵</li> </ul>	P4 <sub>7</sub> ~ P4 <sub>0</sub> /D <sub>7</sub> ~ D <sub>0</sub>	データ入出力端子 (D <sub>7</sub> ~ D <sub>0</sub> ) と 8ビットの入出力ポートの兼用 8ビットバスモードのとき入出力ポート 16ビットバスモードのときデータ入出力端子					入出力ポート	
ポート5	<ul style="list-style-type: none"> <li>・ 4ビットの入出力ポート</li> <li>・ 入力プルアップ MOS 内蔵</li> <li>・ LED 駆動可能</li> </ul>	P5 <sub>3</sub> ~ P5 <sub>0</sub> / A <sub>19</sub> ~ A <sub>16</sub>	アドレス出力端子 (A <sub>19</sub> ~ A <sub>16</sub> )				アドレス出力端子 (A <sub>19</sub> ~ A <sub>16</sub> ) と 4ビットの入出力ポートの兼用 DDR=0 のとき入力ポート DDR=1 のときアドレス出力端子	入出力ポート	

ポート	概要	端子	拡張モード					シングルチップモード	
			モード1	モード2	モード3	モード4	モード5	モード6	モード7
ポート6	・8ビットの入出力ポート	P6 <sub>7</sub> /	クロック出力端子 ( ) と入力ポートの兼用						
		P6 <sub>6</sub> /LWR P6 <sub>5</sub> /HWR P6 <sub>4</sub> /RD P6 <sub>3</sub> /AS	バス制御信号出力端子 ( LWR, HWR, RD, AS )					入出力ポート	
		P6 <sub>2</sub> /BACK P6 <sub>1</sub> /BREQ P6 <sub>0</sub> /WAIT	バス制御信号入出力端子 ( BACK, BREQ, WAIT ) と3ビットの入出力ポートの兼用						
		ポート7	・8ビットの入出力ポート	P7 <sub>7</sub> /AN <sub>7</sub> /DA <sub>1</sub> P7 <sub>6</sub> /AN <sub>6</sub> /DA <sub>0</sub>	A/D変換器のアナログ入力端子 ( AN <sub>7</sub> , AN <sub>6</sub> ) および D/A変換器のアナログ出力端子 ( DA <sub>1</sub> , DA <sub>0</sub> ) と入力ポートの兼用				
ポート8	P7 <sub>5</sub> ~ P7 <sub>0</sub> / AN <sub>5</sub> ~ AN <sub>0</sub>	A/D変換器のアナログ入力端子 ( AN <sub>5</sub> ~ AN <sub>0</sub> ) と入力ポートの兼用							
ポート8	・5ビットの入出力ポート ・P8 <sub>2</sub> ~ P8 <sub>0</sub> はシユミット入力	P8 <sub>4</sub> /CS <sub>0</sub>	DDR=0のとき入力ポート DDR=1のとき (リセット後) CS <sub>0</sub> 出力端子			DDR=0のとき (リセット後) 入力ポート DDR=1のとき CS <sub>0</sub> 出力端子		入出力ポート	
		P8 <sub>3</sub> /IRQ <sub>3</sub> /CS <sub>1</sub> / ADTRG	IRQ <sub>3</sub> 入力端子、CS <sub>1</sub> 出力端子、A/D変換器の外部トリガ入力端子 ( ADTRG ) と入力ポートの兼用、DDR=0のとき (リセット後) 入力ポート、DDR=1のとき CS <sub>1</sub> 出力端子					IRQ <sub>3</sub> 入力端子、A/D変換器の外部トリガ入力端子 ( ADTRG ) と入出力ポートの兼用	
		P8 <sub>2</sub> /IRQ <sub>2</sub> /CS <sub>2</sub> P8 <sub>1</sub> /IRQ <sub>1</sub> /CS <sub>3</sub>	IRQ <sub>2</sub> , IRQ <sub>1</sub> 入力端子、CS <sub>2</sub> , CS <sub>3</sub> 出力端子と入力ポートの兼用*、DDR=0のとき (リセット後) 入力ポート、DDR=1のとき CS <sub>2</sub> , CS <sub>3</sub> 出力端子 【注】* P8 <sub>1</sub> はDRCRAの設定により出力ポートとして使用できません。					IRQ <sub>2</sub> , IRQ <sub>1</sub> 入力端子と入出力ポートの兼用	
		P8 <sub>0</sub> /IRQ <sub>0</sub> /RFSH	IRQ <sub>0</sub> 入力端子、RFSH出力端子と入出力ポートの兼用					IRQ <sub>0</sub> 入力端子と入出力ポートの兼用	

8. I/O ポート

ポート	概要	端子	拡張モード					シングルチップモード	
			モード 1	モード 2	モード 3	モード 4	モード 5	モード 6	モード 7
ポート 9	・ 6 ビットの入出力ポート	P9 <sub>5</sub> / $\overline{\text{IRQ}}_5$ /SCK <sub>1</sub> P9 <sub>4</sub> / $\overline{\text{IRQ}}_4$ /SCK <sub>0</sub> P9 <sub>3</sub> /RxD <sub>1</sub> P9 <sub>2</sub> /RxD <sub>0</sub> P9 <sub>1</sub> /TxD <sub>1</sub> P9 <sub>0</sub> /TxD <sub>0</sub>	シリアルコミュニケーションインタフェースチャンネル 0, 1( SCI0,1)の入出力端子( SCK <sub>1</sub> , SCK <sub>0</sub> , RxD <sub>1</sub> , RxD <sub>0</sub> , TxD <sub>1</sub> , TxD <sub>0</sub> ) ,および $\overline{\text{IRQ}}_5$ , $\overline{\text{IRQ}}_4$ 入力端子と 6 ビットの入出力ポートの兼用						
ポート A	・ 8 ビットの入出力ポート ・ シュミット入力	PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub> /A <sub>20</sub>	プログラマブルタイミ ングパターンコント ローラ (TPC) 出力端子 (TP <sub>7</sub> )、16 ビット タイマの入出力端子 (TIOCB <sub>2</sub> ) と入出力 ポートの兼用	アドレス出力端子 (A <sub>20</sub> )	アドレス出力端子 (A <sub>20</sub> )、プログラマ ブルタイミングパ ターンコントローラ (TPC) 出力端子 (TP <sub>7</sub> )、16 ビット タイマの入出力端子 (TIOCB <sub>2</sub> ) と入出力 ポートの兼用	TPC 出力端子 (TP <sub>7</sub> )、 16 ビットタイマの入出 力端子 (TIOCB <sub>2</sub> ) と 入出力ポートの兼用			
		PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub> /A <sub>21</sub> PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub> /A <sub>22</sub> PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub> /A <sub>23</sub>	TPC 出力端子 (TP <sub>6</sub> ~ TP <sub>4</sub> )、16 ビット タイマの入出力端子 (TIOCA <sub>2</sub> , TIOCB <sub>1</sub> , TIOCA <sub>1</sub> ) と入出力 ポートの兼用	TPC 出力端子 (TP <sub>6</sub> ~ TP <sub>4</sub> )、16 ビット タイマの入出力端子 (TIOCA <sub>2</sub> , TIOCB <sub>1</sub> , TIOCA <sub>1</sub> )、アドレス出力端子 (A <sub>23</sub> ~ A <sub>21</sub> ) と入出力ポートの兼用	TPC 出力端子 (TP <sub>6</sub> ~ TP <sub>4</sub> )、16 ビット タイマの入出力端子 (TIOCA <sub>2</sub> , TIOCB <sub>1</sub> , TIOCA <sub>1</sub> ) と 入出力ポートの兼用				
		PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> /TCLKD PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> /TCLKC PA <sub>1</sub> /TP <sub>1</sub> /TCLKB $\overline{\text{TEND}}_1$ PA <sub>0</sub> /TP <sub>0</sub> /TCLKA $\overline{\text{TEND}}_0$	TPC 出力端子 (TP <sub>3</sub> ~ TP <sub>0</sub> )、16 ビット タイマの入出力端子 (TIOCB <sub>0</sub> , TIOCA <sub>0</sub> , TCLKD, TCLKC, TCLKB, TCLKA)、 8 ビットタイマの 入力端子 (TCLKD, TCLKC, TCLKB, TCLKA)、DMA コントローラ (DMAC) の出力端子 ( $\overline{\text{TEND}}_1$ , $\overline{\text{TEND}}_0$ )、と入出力 ポートの兼用	TPC 出力端子 (TP <sub>3</sub> ~ TP <sub>0</sub> )、16 ビット タイマの入出力端子 (TIOCB <sub>0</sub> , TIOCA <sub>0</sub> , TCLKD, TCLKC, TCLKB, TCLKA)、 8 ビットタイマの 入力端子 (TCLKD, TCLKC, TCLKB, TCLKA)、DMA コントローラ (DMAC) の出力端子 ( $\overline{\text{TEND}}_1$ , $\overline{\text{TEND}}_0$ )、と入出力 ポートの兼用	TPC 出力端子 (TP <sub>3</sub> ~ TP <sub>0</sub> )、16 ビット タイマの入出力端子 (TIOCB <sub>0</sub> , TIOCA <sub>0</sub> , TCLKD, TCLKC, TCLKB, TCLKA)、 8 ビットタイマの 入力端子 (TCLKD, TCLKC, TCLKB, TCLKA)、DMA コントローラ (DMAC) の出力端子 ( $\overline{\text{TEND}}_1$ , $\overline{\text{TEND}}_0$ )、と入出力 ポートの兼用				

ポート	概要	端子	拡張モード					シングルチップモード		
			モード1	モード2	モード3	モード4	モード5	モード6	モード7	
ポート B	・ 8 ビットの入出力ポート	PB <sub>7</sub> /TP <sub>15</sub> /RXD <sub>2</sub> PB <sub>6</sub> /TP <sub>14</sub> /TXD <sub>2</sub> PB <sub>5</sub> /TP <sub>13</sub> /SCK <sub>2</sub> / $\overline{\text{LCAS}}$ PB <sub>4</sub> /TP <sub>12</sub> / $\overline{\text{UCAS}}$	TPC 出力端子 ( TP <sub>15</sub> ~ TP <sub>12</sub> )、SCI2 の入出力端子 ( SCK <sub>2</sub> , RxD <sub>2</sub> , TxD <sub>2</sub> )、DRAM インタフェースの出力端子 ( $\overline{\text{LCAS}}$ , $\overline{\text{UCAS}}$ ) と入出力ポートの兼用						TPC 出力端子 ( TP <sub>15</sub> ~ TP <sub>12</sub> )、SCI2 の入出力端子 ( SCK <sub>2</sub> , RxD <sub>2</sub> , TxD <sub>2</sub> ) と入出力ポートの兼用	
		PB <sub>3</sub> /TP <sub>11</sub> /TMIO <sub>3</sub> / $\overline{\text{DREQ}}_1/\overline{\text{CS}}_4$ PB <sub>2</sub> /TP <sub>10</sub> /TMO <sub>2</sub> / $\overline{\text{CS}}_5$ PB <sub>1</sub> /TP <sub>9</sub> /TMIO <sub>1</sub> / $\overline{\text{DREQ}}_0/\overline{\text{CS}}_6$ PB <sub>0</sub> /TP <sub>8</sub> /TMO <sub>0</sub> / $\overline{\text{CS}}_7$	TPC 出力端子 ( TP <sub>11</sub> ~ TP <sub>8</sub> )、8 ビットタイマの入出力端子 ( TMIO <sub>3</sub> , TMO <sub>2</sub> , TMIO <sub>1</sub> , TMO <sub>0</sub> )、DMA コントローラ ( DMAC ) の入力端子 ( $\overline{\text{DREQ}}_1$ , $\overline{\text{DREQ}}_0$ )、 $\overline{\text{CS}}_7$ ~ $\overline{\text{CS}}_4$ 出力端子と入出力ポートの兼用						TPC 出力端子 ( TP <sub>11</sub> ~ TP <sub>8</sub> )、8 ビットタイマの入出力端子 ( TMIO <sub>3</sub> , TMO <sub>2</sub> , TMIO <sub>1</sub> , TMO <sub>0</sub> )、DMA コントローラ ( DMAC ) の入力端子 ( $\overline{\text{DREQ}}_1$ , $\overline{\text{DREQ}}_0$ ) と入出力ポートの兼用	

## 8.2 ポート 1

### 8.2.1 概要

ポート 1 は、アドレス出力兼用の 8 ビットの入出力ポートです。ポート 1 の各端子は、図 8.1 に示す構成となっており、動作モードにより端子機能が異なります。モード 1~4 (内蔵 ROM 無効拡張モード) のときは、アドレスバス ( $A_7 \sim A_0$ ) 出力端子となります。

モード 5 (内蔵 ROM 有効拡張モード) のときは、ポート 1 データディレクションレジスタ (P1DDR) の設定によりアドレスバス ( $A_7 \sim A_0$ ) 出力端子、または入力ポートとなります。

モード 6、7 (シングルチップモード) のときは、入出力ポートとなります。

エリア 2、3、4、5 に DRAM を接続する場合には、リード/ライトサイクルで  $A_7 \sim A_0$  がロウ/カラムアドレス出力となります。詳細は「6.5 DRAM インタフェース」を参照してください。

また、ポート 1 は、1 個の TTL 負荷と 90pF の容量を駆動することができます。また、LED、ダーリントントランジスタを駆動することもできます。



図 8.1 ポート 1 の端子構成

### 8.2.2 レジスタ構成

表 8.2 にポート 1 のレジスタ構成を示します。

表 8.2 ポート 1 レジスタ構成

アドレス*	名称	略称	R/W	初期値	
				モード 1~4	モード 5~7
H'EE000	ポート 1 データディレクションレジスタ	P1DDR	W	H'FF	H'00
H'FFFD0	ポート 1 データレジスタ	P1DR	R/W	H'00	

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

## (1) ポート1 データディレクションレジスタ (P1DDR)

P1DDRは、8ビットのライト専用のレジスタで、ポート1各端子の入出力をビットごとに指定することができます。

ビット:		7	6	5	4	3	2	1	0
		P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR
モード1~4	初期値:	1	1	1	1	1	1	1	1
	R/W:	—	—	—	—	—	—	—	—
モード5~7	初期値:	0	0	0	0	0	0	0	0
	R/W:	W	W	W	W	W	W	W	W

ポート1データディレクション7~0  
ポート1の各端子の入出力を選択するビットです。

## (a) モード1~4 (内蔵ROM無効拡張モード)

P1DDRは1に固定され、ポート1はアドレスバスとして機能します。

## (b) モード5 (内蔵ROM有効拡張モード)

ポート1はリセット直後は入力ポートとなっています。

P1DDRに1をセットすると対応するポート1の端子はアドレス出力となり、0にクリアすると入力ポートになります。

## (c) モード6、7 (シングルチップモード)

ポート1は入出力ポートとして機能します。P1DDRに1をセットすると対応するポート1の端子は出力ポートとなり、0にクリアすると入力ポートになります。

モード1~4ではP1DDRは、リードすると常に1が読み出されます。ライトは無効です。

モード5~7ではP1DDRは、ライト専用のレジスタで、リードは無効です。リードすると1が読み出されます。

P1DDRは、リセット、またはハードウェアスタンバイモード時にモード1~4の場合はH'FFに、モード5~7の場合はH'00にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。そのため、ポート1が入出力ポートとして機能しているとき、P1DDRが1にセットされた状態でソフトウェアスタンバイモードに移行するとその端子は出力状態のままとなっています。



## (2) ポート 1 データレジスタ (P1DR)

P1DR は、8 ビットのリード/ライト可能なレジスタで、ポート 1 の出力データを格納します。ポート 1 が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、P1DDR が 0 のビットは端子のロジックレベルが読み出され、1 のビットは P1DR の値が読み出されます。

ビット:	7	6	5	4	3	2	1	0
	P17	P16	P15	P14	P13	P12	P11	P10
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート1データ7~0  
ポート1の各端子のデータを格納するビットです。

P1DR は、リセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

## 8.3 ポート 2

### 8.3.1 概要

ポート 2 は、アドレス出力兼用の 8 ビットの入出力ポートです。ポート 2 の各端子は、図 8.2 に示す構成となっており、動作モードにより端子機能が異なります。

モード 1~4 (内蔵 ROM 無効拡張モード) のときは、アドレスバス ( $A_{15} \sim A_8$ ) 出力端子となります。モード 5 (内蔵 ROM 有効拡張モード) のときは、ポート 2 データディレクションレジスタ (P2DDR) の設定によりアドレスバス ( $A_{15} \sim A_8$ ) または入力ポートとなります。

モード 6、7 (シングルチップモード) のときは、入出力ポートとなります。

エリア 2、3、4、5 に DRAM を接続する場合には、リード/ライトサイクルで  $A_{12} \sim A_8$  が ROW/カラムアドレス出力となります。詳細は「6.5 DRAM インタフェース」を参照してください。

ポート 2 は、プログラムで制御可能なプルアップ MOS が内蔵されています。また、1 個の TTL 負荷と 90pF の容量を駆動することや、LED、ダーリントントランジスタを駆動することができます。

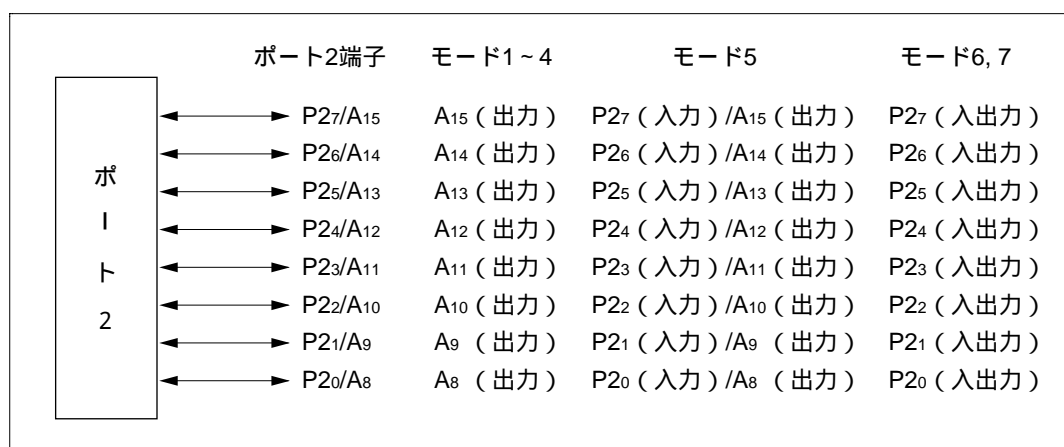


図 8.2 ポート 2 の端子構成

## 8.3.2 レジスタ構成

表 8.3 にポート 2 のレジスタ構成を示します。

表 8.3 ポート 2 レジスタ構成

アドレス*	名 称	略 称	R/W	初期値	
				モード 1~4	モード 5~7
H'EE001	ポート 2 データディレクションレジスタ	P2DDR	W	H'FF	H'00
H'FFFD1	ポート 2 データレジスタ	P2DR	R/W	H'00	
H'EE03C	ポート 2 入力プルアップ MOS コントロ ールレジスタ	P2PCR	R/W	H'00	

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

## (1) ポート 2 データディレクションレジスタ (P2DDR)

P2DDR は、8 ビットのライト専用のレジスタで、ポート 2 の各端子の入出力をビットごとに指定することができます。

ビット:		7	6	5	4	3	2	1	0
		P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P20DDR
モード 1~4	初期値:	1	1	1	1	1	1	1	1
	R/W:	—	—	—	—	—	—	—	—
モード 5~7	初期値:	0	0	0	0	0	0	0	0
	R/W:	W	W	W	W	W	W	W	W

ポート 2 データディレクション 7~0  
ポート 2 の各端子の入出力を選択するビットです。

## (a) モード 1~4 (内蔵 ROM 無効拡張モード)

P2DDR は 1 に固定され、ポート 2 はアドレスバスとして機能します。

## (b) モード 5 (内蔵 ROM 有効拡張モード)

ポート 2 はリセット直後は入力ポートとなっています。

P2DDR に 1 をセットすると対応するポート 2 の端子はアドレス出力端子となり、0 にクリアすると入力ポートになります。

## (c) モード6、7 (シングルチップモード)

ポート2は入出力ポートとして機能します。P2DDRに1をセットすると対応するポート2の端子は出力ポートとなり、0にクリアすると入力ポートとなります。

モード1~4ではP2DDRは、リードすると常に1が読み出されます。ライトは無効です。モード5~7ではP2DDRは、ライト専用のレジスタで、リードは無効です。リードすると1が読み出されます。

P2DDRは、リセット、またはハードウェアスタンバイモード時にモード1~4の場合はH'FFに、モード5~7の場合はH'00にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。そのため、ポート2が入出力ポートとして機能しているとき、P2DDRが1にセットされた状態でソフトウェアスタンバイモードに遷移するとその端子は出力状態のままとなっています。

## (2) ポート2 データレジスタ (P2DR)

P2DRは、8ビットのリード/ライト可能なレジスタで、ポート2の出力データを格納します。ポート2が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、P2DDRが0のビットは端子のロジックレベルが読み出され、1のビットはP2DRの値が読み出されます。

ビット:	7	6	5	4	3	2	1	0
	P27	P26	P25	P24	P23	P22	P21	P20
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート2データ7~0

ポート2の各端子のデータを格納するビットです。

P2DRは、リセット、またはハードウェアスタンバイモード時に、H'00にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

## (3) ポート2 入力プルアップ MOS コントロールレジスタ (P2PCR)

P2PCRは8ビットのリード/ライト可能なレジスタで、ポート2に内蔵した入力プルアップMOSをビットごとに制御します。

ビット:	7	6	5	4	3	2	1	0
	P27PCR	P26PCR	P25PCR	P24PCR	P23PCR	P22PCR	P21PCR	P20PCR
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート2入力プルアップMOSコントロール7~0  
ポート2に内蔵した入力プルアップMOSを制御するビットです。

モード5~7のとき、P2DDRを0にクリアした(入力ポートの)状態でP2PCRを1にセットすると対応するビットの入力プルアップMOSはONします。

P2PCRは、リセット、またはハードウェアスタンバイモード時に、H'00にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

表8.4 入力プルアップMOSの状態(ポート2)

モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	その他の動作時
1				
2		OFF		OFF
3				
4				
5				
6		OFF		ON / OFF
7				

《記号説明》

OFF : 入力プルアップMOSは、常にOFF状態です。

ON / OFF : P2PCR = 1かつP2DDR = 0のときON状態、その他のときはOFF状態です。

## 8.4 ポート3

### 8.4.1 概要

ポート3は、データバス兼用の8ビットの入出力ポートです。ポート3の各端子は、図8.3に示す構成となっており、モード1~5(拡張モード)のときはデータバスとなり、モード6、7(シングルチップモード)のときは、入出力ポートとなります。

ポート3は、1個のTTL負荷と90pFの容量負荷を駆動することができます。また、ダーリントントランジスタを駆動することもできます。

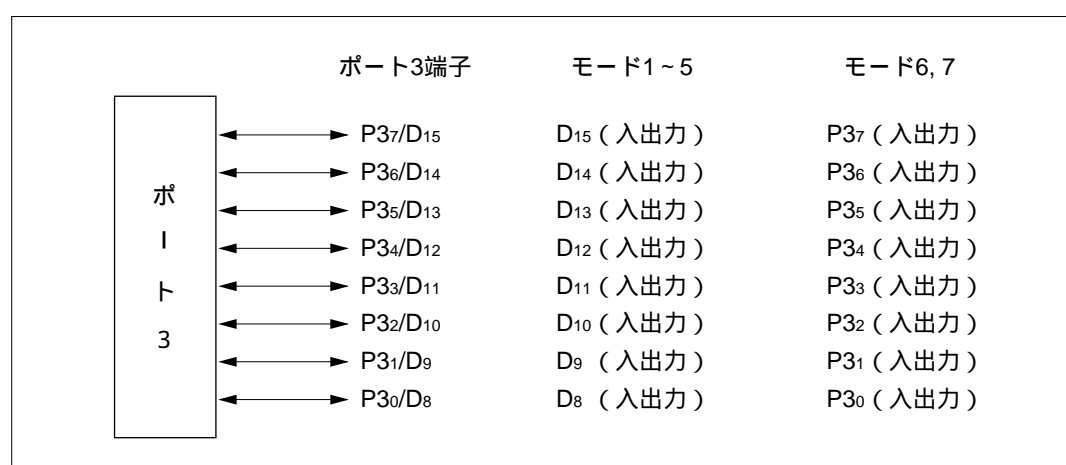


図 8.3 ポート3の端子構成

### 8.4.2 レジスタ構成

表 8.5 にポート3のレジスタ構成を示します。

表 8.5 ポート3レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H'EE002	ポート3データディレクションレジスタ	P3DDR	W	H'00
H'FFFD2	ポート3データレジスタ	P3DR	R/W	H'00

【注】 \* アドバンスモード時のアドレス下位20ビットを示しています。

## (1) ポート3 データディレクションレジスタ (P3DDR)

P3DDR は、8 ビットのライト専用のレジスタで、ポート3 各端子の入出力をビットごとに指定することができます。

ビット :	7	6	5	4	3	2	1	0
	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR
初期値 :	0	0	0	0	0	0	0	0
R/W :	W	W	W	W	W	W	W	W

ポート3データディレクション7~0  
ポート3の各端子の入出力を選択するビットです。

## (a) モード1~5 (拡張モード)

P3DDR の設定に関わらず、ポート3 はデータバスとして機能します。

## (b) モード6、7 (シングルチップモード)

ポート3 は入出力ポートとして機能します。

P3DDR に1 をセットすると対応するポート3 の端子は出力ポートとなり、0 にクリアすると入力ポートになります。

P3DDR は、ライト専用で、リードは無効です。リードすると、1 が読み出されます。

P3DDR は、リセット、またはハードウェアスタンバイモード時に H'00 にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。そのため、ポート3 が入出力ポートとして機能しているとき、P3DDR が1 にセットされた状態でソフトウェアスタンバイモードに遷移するとその端子は出力状態のままとなっています。

## (2) ポート3データレジスタ (P3DR)

P3DRは、8ビットのリード/ライト可能なレジスタで、ポート3の出力データを格納します。ポート3が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、P3DDRが0のビットは端子のロジックレベルが読み出され、1のビットはP3DRの値が読み出されます。

ビット:	7	6	5	4	3	2	1	0
	P37	P36	P35	P34	P33	P32	P31	P30
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート3データ7~0  
ポート3の各端子のデータを格納するビットです。

P3DRは、リセット、またはハードウェアスタンバイモード時に、H'00にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。



## 8.5 ポート4

### 8.5.1 概要

ポート4は、データバス兼用の8ビットの入出力ポートです。ポート4の各端子は、図8.4に示す構成となっており、動作モードにより端子機能が異なります。

モード1~5(拡張モード)のときバス幅コントロールレジスタ(ABWCR)により、エリア0~7のすべてを8ビットアクセス空間に設定すると、8ビットバスモードとなり、ポート4は入出力ポートとなります。また、エリア0~7のうち少なくとも1つのエリアを16ビットアクセス空間に設定すると、16ビットバスモードとなり、ポート4はデータバスとなります。

モード6、7(シングルチップモード)のとき、ポート4は、入出力ポートとなります。

ポート4は、プログラムで制御可能なプルアップMOSが内蔵されています。

ポート4は、1個のTTL負荷と90pFの容量負荷を駆動することができます。また、ダーリントントランジスタを駆動することもできます。

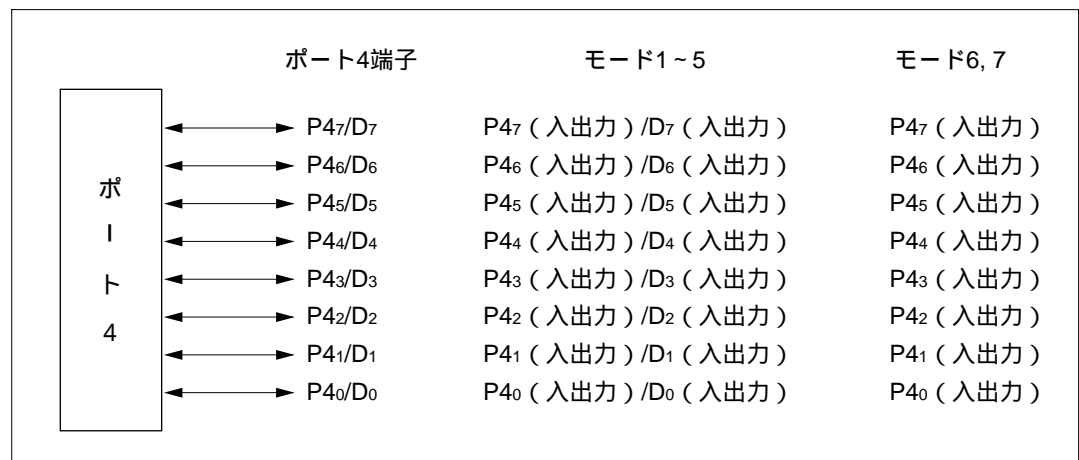


図 8.4 ポート4の端子構成

## 8.5.2 レジスタ構成

表 8.6 にポート 4 のレジスタ構成を示します。

表 8.6 ポート 4 レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H' EE003	ポート 4 データディレクションレジスタ	P4DDR	W	H'00
H' FFFD3	ポート 4 データレジスタ	P4DR	R/W	H'00
H' EE03E	ポート 4 入力プルアップ MOS コントロール レジスタ	P4PCR	R/W	H'00

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

### (1) ポート 4 データディレクションレジスタ (P4DDR)

P4DDR は、8 ビットのライト専用のレジスタで、ポート 4 各端子の入出力をビットごとに指定することができます。

ビット:	7	6	5	4	3	2	1	0
	P47DDR	P46DDR	P45DDR	P44DDR	P43DDR	P42DDR	P41DDR	P40DDR
初期値:	0	0	0	0	0	0	0	0
R/W :	W	W	W	W	W	W	W	W

ポート4データディレクション7~0  
ポート4の各端子の入出力を選択するビットです。

#### (a) モード 1~5 (拡張モード)

バスコントローラのバス幅コントロールレジスタ (ABWCR) により全エリアを 8 ビットアクセス空間に設定し、8 ビットバスモードとしたときポート 4 は入出力ポートとして機能します。このとき P4DDR に 1 をセットすると対応するポート 4 の端子は出力ポートとなり、0 にクリアすると入力ポートになります。

また少なくとも 1 つのエリアを 16 ビットアクセス空間に設定し、16 ビットバスモードとしたとき、P4DDR の設定値に関わらずポート 4 はデータバスとして機能します。

#### (b) モード 6、7 (シングルチップモード)

ポート 4 は入出力ポートとして機能します。P4DDR に 1 をセットすると対応するポート 4 の端子は出力ポートとなり、0 にクリアすると入力ポートになります。

P4DDR は、ライト専用レジスタで、リードは無効です。リードすると、1 が読み出され

ます。

P4DDR は、リセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。ソフトウェアスタンバイモード時には直前の状態を保持します。

ABWCR および P4DDR はソフトウェアスタンバイモード時にはイニシャライズされません。そのため、ポート 4 が入出力ポートとして機能しているとき、P4DDR が 1 にセットされた状態でソフトウェアスタンバイモードに遷移すると、その端子は出力状態のままとなっています。

## (2) ポート 4 データレジスタ (P4DR)

P4DR は、8 ビットのリード/ライト可能なレジスタで、ポート 4 の出力データを格納します。ポート 4 が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、P4DDR が 0 のビットは端子のロジックレベルが読み出され、1 のビットは、P4DR の値が読み出されます。

ビット:	7	6	5	4	3	2	1	0
	P47	P46	P45	P44	P43	P42	P41	P40
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート4データ7~0  
 ポート4の各端子のデータを格納するビットです。

P4DR は、リセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

### (3) ポート 4 入力プルアップ MOS コントロールレジスタ (P4PCR)

P4PCR は 8 ビットのリード/ライト可能なレジスタで、ポート 4 に内蔵した入力プルアップ MOS をビットごとに制御します。

ビット:	7	6	5	4	3	2	1	0
	P47PCR	P46PCR	P45PCR	P44PCR	P43PCR	P42PCR	P41PCR	P40PCR
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート4入力プルアップMOSコントロール7~0  
ポート4に内蔵した入力プルアップMOSを制御するビットです。

モード 1~5 (拡張モード) の 8 ビットバスモード時とモード 6、7 (シングルチップモード) 時、P4DDR を 0 にクリアした (入力ポートの) 状態で、P4PCR を 1 にセットすると入力プルアップ MOS は ON します。

P4PCR は、リセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

各動作モードでの、入力プルアップ MOS の状態を表 8.7 に示します。

表 8.7 入力プルアップ MOS の状態 (ポート 4)

モード		リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	その他の動作時
1~5	8 ビットバス モード	OFF		ON / OFF	
	16 ビットバス モード			OFF	
6、7					

《記号説明》

OFF : 入力プルアップ MOS は、常に OFF 状態です。

ON / OFF : P4PCR = 1 かつ P4DDR = 0 のとき ON 状態、その他のときは OFF 状態です。

## 8.6 ポート5

### 8.6.1 概要

ポート5は、アドレス出力兼用の4ビットの入出力ポートです。ポート5の各端子は、図8.5に示す構成となっており、動作モードにより端子機能が異なります。

モード1~4(内蔵ROM無効拡張モード)に設定したとき、ポート5の各端子はアドレス(A<sub>19</sub>~A<sub>16</sub>)出力として機能します。モード5(内蔵ROM有効拡張モード)に設定したときは、ポート5データディレクションレジスタ(P5DDR)の設定によりアドレスバス(A<sub>19</sub>~A<sub>16</sub>)または入力ポートとなります。

モード6、7(シングルチップモード)のときは、入出力ポートとなります。

ポート5は、プログラムで制御可能なプルアップMOSが内蔵されています。

ポート5は、1個のTTL負荷と90pFの容量負荷を駆動することができます。また、LED、ダーリントントランジスタを駆動することも可能です。

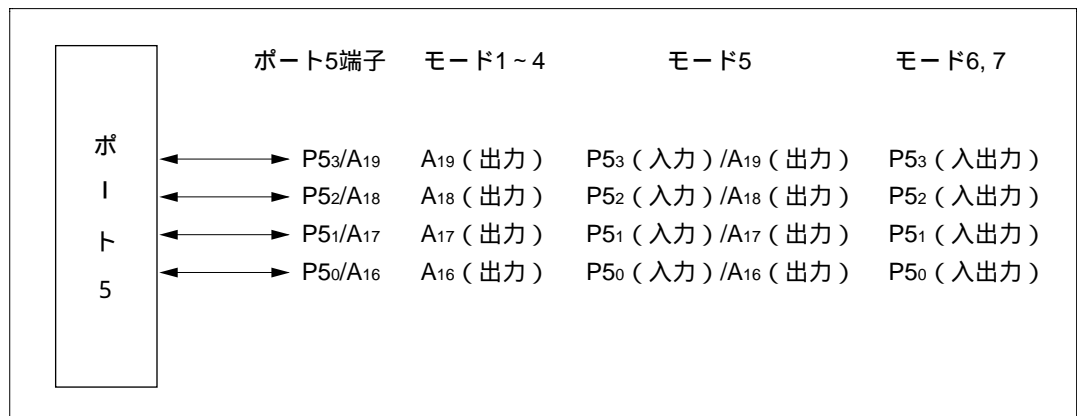


図8.5 ポート5の端子構成

### 8.6.2 レジスタ構成

表8.8にポート5のレジスタ構成を示します。

表8.8 ポート5レジスタ構成

アドレス*	名称	略称	R/W	初期値	
				モード1~4	モード5~7
H'EE004	ポート5データディレクションレジスタ	P5DDR	W	H'FF	H'F0
H'FFFD4	ポート5データレジスタ	P5DR	R/W	H'F0	
H'EE03F	ポート5入力プルアップMOSコントロールレジスタ	P5PCR	R/W	H'F0	

【注】 \* アドバンスモード時のアドレス下位20ビットを示しています。

## (1) ポート5 データディレクションレジスタ (P5DDR)

P5DDRは、8ビットのライト専用のレジスタで、ポート5各端子の入出力をビットごとに指定することができます。

ビット7~ビット4はリザーブビットで1に固定されています。ライトは無効です。

ビット:		7	6	5	4	3	2	1	0
		—	—	—	—	P5 <sub>3</sub> DDR	P5 <sub>2</sub> DDR	P5 <sub>1</sub> DDR	P5 <sub>0</sub> DDR
モード1~4	初期値:	1	1	1	1	1	1	1	1
	R/W:	—	—	—	—	—	—	—	—
モード5~7	初期値:	1	1	1	1	0	0	0	0
	R/W:	—	—	—	—	W	W	W	W

リザーブビット

ポート5データディレクション3~0  
ポート5の各端子の入出力を選択する  
ビットです。

## (a) モード1~4 (内蔵ROM無効拡張モード)

P5DDRは1に固定され、ポート5はアドレス出力として機能します。

## (b) モード5 (内蔵ROM有効拡張モード)

ポート5はリセット直後は入力ポートとなっています。P5DDRに1をセットすると対応するポート5の端子がアドレス出力端子になり、0にクリアすると入力ポートになります。

## (c) モード6、7 (シングルチップモード)

ポート5は、入出力ポートとして機能します。P5DDRに1をセットすると対応する端子は出力ポートになり、0にクリアすると入力ポートになります。

モード1~4ではP5DDRは、リードすると常に1が読み出されます。ライトは無効です。

モード5~7ではP5DDRは、ライト専用レジスタで、リードは無効です。リードすると1が読み出されます。

P5DDRは、リセット、またはハードウェアスタンバイモード時に、モード1~4の場合はH'FFに、モード5~7の場合はH'F0にイニシャライズされます。ソフトウェアスタンバイモード時には直前の状態を保持します。そのため、ポート5が入出力ポートとして機能しているときP5DDRが1にセットされた状態でソフトウェアスタンバイモードに移行すると、その端子は出力状態のままとなっています。

## (2) ポート5データレジスタ (P5DR)

P5DRは、8ビットのリード/ライト可能なレジスタで、ポート5の出力データを格納します。ポート5が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、P5DDRの値が0のビットは端子のロジックレベルが読み出され、1のビットは、P5DRの値が読み出されます。

ビット7~ビット4はリザーブビットで1に固定されています。ライトは無効です。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	P5 <sub>3</sub>	P5 <sub>2</sub>	P5 <sub>1</sub>	P5 <sub>0</sub>
初期値:	1	1	1	1	0	0	0	0
R/W:	—	—	—	—	R/W	R/W	R/W	R/W

リザーブビット

ポート5データ3~0  
ポート5の各端子のデータを格納するビットです。

P5DRは、リセット、またはハードウェアスタンバイモード時に、H'F0にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

## (3) ポート5入力プルアップMOSコントロールレジスタ (P5PCR)

P5PCRは8ビットのリード/ライト可能なレジスタで、ポート5に内蔵した入力プルアップMOSをビットごとに制御します。

ビット7~ビット4はリザーブビットで1に固定されています。ライトは無効です。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	P5 <sub>3</sub> PCR	P5 <sub>2</sub> PCR	P5 <sub>1</sub> PCR	P5 <sub>0</sub> PCR
初期値:	1	1	1	1	0	0	0	0
R/W:	—	—	—	—	R/W	R/W	R/W	R/W

リザーブビット

ポート5入力プルアップMOSコントロール3~0  
ポート5に内蔵した入力プルアップMOSを制御するビットです。

モード5~7のとき、P5DDRを0にクリアした(入力ポート)状態でP5PCRを1にセットすると入力プルアップMOSはONします。

P5PCRは、リセット、またはハードウェアスタンバイモード時に、H'F0にイニシャラ

イズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。  
各動作モードでの、入力プルアップMOSの状態を表8.9に示します。

表8.9 入力プルアップMOSの状態（ポート5）

モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	その他の動作時
1	OFF	OFF	OFF	OFF
2				
3				
4				
5	OFF	OFF	ON / OFF	ON / OFF
6				
7				

《記号説明》

OFF : 入力プルアップMOSは、常にOFF状態です。

ON / OFF : P5PCR = 1 かつ P5DDR = 0 のとき ON 状態、その他のときは OFF 状態です。



## 8.7 ポート6

### 8.7.1 概要

ポート6は、8ビットの入出力ポートです。ポート6はバス制御入出力端子( $\overline{LWR}$ 、 $\overline{HWR}$ 、 $\overline{RD}$ 、 $\overline{AS}$ 、 $\overline{BACK}$ 、 $\overline{BREQ}$ 、 $\overline{WAIT}$ )、クロック端子( )と兼用になっています。

ポート6の端子構成を図8.6に示します。

モード1~5(拡張モード)時には、P6<sub>7</sub>(入力ポート) /  $\overline{LWR}$ 、 $\overline{HWR}$ 、 $\overline{RD}$ 、 $\overline{AS}$ 、P6<sub>2</sub>/ $\overline{BACK}$ 、P6<sub>1</sub>/ $\overline{BREQ}$ 、P6<sub>0</sub>/ $\overline{WAIT}$ として機能します。端子機能の選択方法については表8.11を参照してください。モード6、7(シングルチップモード)時には、P6<sub>7</sub>(入力ポート) / P6<sub>6</sub>~P6<sub>0</sub>(入出力ポート)となります。

エリア2、3、4、5にDRAMを接続する場合には、 $\overline{LWR}$ 、 $\overline{HWR}$ 、 $\overline{RD}$ がそれぞれ $\overline{LCAS}$ 、 $\overline{UCAS}$ 、 $\overline{WE}$ と兼用になります。詳細は「6.5 DRAMインタフェース」を参照してください。

ポート6は、1個のTTL負荷と90pFの容量を負荷駆動することができます。また、ダーリントントランジスタを駆動することができます。



図8.6 ポート6の端子構成

### 8.7.2 レジスタ構成

表8.10にポート6のレジスタ構成を示します。

表8.10 ポート6レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H'EE005	ポート6データディレクションレジスタ	P6DDR	W	H'80
H'FFFD5	ポート6データレジスタ	P6DR	R/W	H'80

【注】 \* アドバンスモード時のアドレス下位20ビットを示しています。

## (1) ポート6データディレクションレジスタ (P6DDR)

P6DDRは、8ビットのライト専用のレジスタで、ポート6の各端子の入出力をビットごとに指定することができます。

ビット7はリザーブビットで1に固定されています。ライトは無効です。

ビット:	7	6	5	4	3	2	1	0
	—	P6 <sub>6</sub> DDR	P6 <sub>5</sub> DDR	P6 <sub>4</sub> DDR	P6 <sub>3</sub> DDR	P6 <sub>2</sub> DDR	P6 <sub>1</sub> DDR	P6 <sub>0</sub> DDR
初期値:	1	0	0	0	0	0	0	0
R/W:	—	W	W	W	W	W	W	W

リザーブビット
ポート6データディレクション6~0  
ポート6の各端子の入出力を選択するビットです。

## (a) モード1~5 (拡張モード)

ポートP6<sub>7</sub>はクロック出力端子( ) / 入力ポートとして機能します。

MSTCRHのPSTOPビットを0にクリア(初期状態)するとクロック出力端子( )となり、1にセットすると入力ポートになります。

ポートP6<sub>6</sub>~P6<sub>3</sub>はP6<sub>6</sub>DDR~P6<sub>3</sub>DDRの設定に関わらずバス制御出力端子( $\overline{LWR}$ 、 $\overline{HWR}$ 、 $\overline{RD}$ 、 $\overline{AS}$ )として機能します。

P6<sub>2</sub>~P6<sub>0</sub>は、バス制御入出力端子( $\overline{BACK}$ 、 $\overline{BREQ}$ 、 $\overline{WAIT}$ ) / 入出力ポートとして機能します。端子機能の選択方法については、表8.11を参照してください。

ポートP6<sub>2</sub>~P6<sub>0</sub>が入出力ポートとして機能する場合、P6DDRを1にセットすると対応するポート6の端子は出力ポートとなり、0にクリアすると入力ポートになります。

## (b) モード6、7 (シングルチップモード)

ポートP6<sub>7</sub>はクロック出力端子( ) / 入力ポートとして機能します。P6<sub>6</sub>~P6<sub>0</sub>は入出力ポートとして機能します。

P6<sub>7</sub>は、MSTCRHのPSTOPビットを0にクリア(初期状態)するとクロック出力端子( )となり、1にセットすると入力ポートになります。

P6<sub>6</sub>~P6<sub>0</sub>は、P6<sub>6</sub>DDR~P6<sub>0</sub>DDRに1をセットすると対応するポート6の端子は出力ポートとなり、0にクリアすると入力ポートになります。

P6DDRは、ライト専用で、リードは無効です。リードすると1が読み出されます。

P6DDRは、リセット、またはハードウェアスタンバイモード時に、H'80にイニシャライズされます。ソフトウェアスタンバイモード時には直前の状態を保持します。このため、ポート6が入出力ポートとして機能しているとき、P6DDRが1にセットされた状態でソフ

トウェアスタンバイモードに遷移すると、その端子は出力状態のままとなっています。

## (2) ポート6データレジスタ (P6DR)

P6DRは、8ビットのリード/ライト可能なレジスタで、ポート6の各端子のデータを格納します。ポート6が出力ポートとして機能する場合、本レジスタの値が出力されます。

ビット7は、MSTCRHのPSTOPビットが0のときリードすると1が読み出され、1のときリードするとP6<sub>7</sub>端子のロジックレベルが読み出されます。ライトは無効です。ビット6~0は、P6DDRの対応するビットが0のときリードすると、端子のロジックレベルが読み出され、1のときリードするとP6DRの値が読み出されます。

ビット:	7	6	5	4	3	2	1	0
	P6 <sub>7</sub>	P6 <sub>6</sub>	P6 <sub>5</sub>	P6 <sub>4</sub>	P6 <sub>3</sub>	P6 <sub>2</sub>	P6 <sub>1</sub>	P6 <sub>0</sub>
初期値:	1	0	0	0	0	0	0	0
R/W :	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポート6データ7~0  
ポート6の各端子のデータを格納するビットです。

P6DRは、リセット、またはハードウェアスタンバイモード時に、H'80にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

表 8.11 モード1～5の端子機能（ポート6）

端子	選択方法と端子機能		
P6 <sub>7</sub> /	MSTCRHのPSTOPビットにより、次のように切り換わります。		
	PSTOP	0	1
	端子機能	出力端子	P6 <sub>7</sub> 入力端子
$\overline{\text{LWR}}$	P6 <sub>6</sub> DDRビットの設定に関わらず、 $\overline{\text{LWR}}$ になります。		
	P6 <sub>6</sub> DDR	0	1
	端子機能	$\overline{\text{LWR}}$ 出力端子 *	
	【注】* DRCRAのDRAS2～DRAS0ビットのいずれかが1であり、DRCRBのCSELビットが1であるとき、 $\overline{\text{LCAS}}$ として $\overline{\text{LWR}}$ 出力します。		
$\overline{\text{HWR}}$	P6 <sub>5</sub> DDRビットの設定に関わらず、 $\overline{\text{HWR}}$ になります。		
	P6 <sub>5</sub> DDR	0	1
	端子機能	$\overline{\text{HWR}}$ 出力端子 *	
	【注】* DRCRAのDRAS2～DRAS0ビットのいずれかが1であり、DRCRBのCSELビットが1であるとき、 $\overline{\text{UCAS}}$ として $\overline{\text{HWR}}$ 出力します。		
$\overline{\text{RD}}$	P6 <sub>4</sub> DDRビットの設定に関わらず、 $\overline{\text{RD}}$ になります。		
	P6 <sub>4</sub> DDR	0	1
	端子機能	$\overline{\text{RD}}$ 出力端子 *	
	【注】* DRCRAのDRAS2～DRAS0ビットのいずれかが1であるとき、 $\overline{\text{WE}}$ として $\overline{\text{RD}}$ 出力します。		
$\overline{\text{AS}}$	P6 <sub>3</sub> DDRビットの設定に関わらず、 $\overline{\text{AS}}$ になります。		
	P6 <sub>3</sub> DDR	0	1
	端子機能	$\overline{\text{AS}}$ 出力端子	
P6 <sub>2</sub> / $\overline{\text{BACK}}$	BRCRのBRLEビットとP6 <sub>2</sub> DDRビットの組み合わせにより、次のように切り換わります。		
	BRLE	0	1
	P6 <sub>2</sub> DDR	0	1
	端子機能	P6 <sub>2</sub> 入力端子	P6 <sub>2</sub> 出力端子

端子	選択方法と端子機能			
P6 <sub>1</sub> / $\overline{\text{BREQ}}$	BRCR の BRLE ビットと P6 <sub>1</sub> DDR ビットの組み合わせにより、次のように切り換わります。			
	BRLE	0		1
	P6 <sub>1</sub> DDR	0	1	-
	端子機能	P6 <sub>1</sub> 入力端子	P6 <sub>1</sub> 出力端子	$\overline{\text{BREQ}}$ 入力端子
P6 <sub>0</sub> / $\overline{\text{WAIT}}$	BCR の WAITE ビットと P6 <sub>0</sub> DDR ビットの組合せにより、次のように切り換わります。			
	WAITE	0		1
	P6 <sub>0</sub> DDR	0	1	0*
	端子機能	P6 <sub>0</sub> 入力端子	P6 <sub>0</sub> 出力端子	$\overline{\text{WAIT}}$ 入力端子
【注】 *P6 <sub>0</sub> DDR は 1 にセットしないでください。				

## 8.8 ポート7

### 8.8.1 概要

ポート7は8ビットの入力専用ポートです。ポート7は、A/D変換器のアナログ入力端子とD/A変換器のアナログ出力端子と兼用になっています。これらの端子機能はいずれの動作モードでも共通です。ポート7の端子構成を図8.7に示します。

A/D変換器のアナログ入力端子については「第15章 A/D変換器」を参照してください。

D/A変換器のアナログ出力端子については「第16章 D/A変換器」を参照してください。

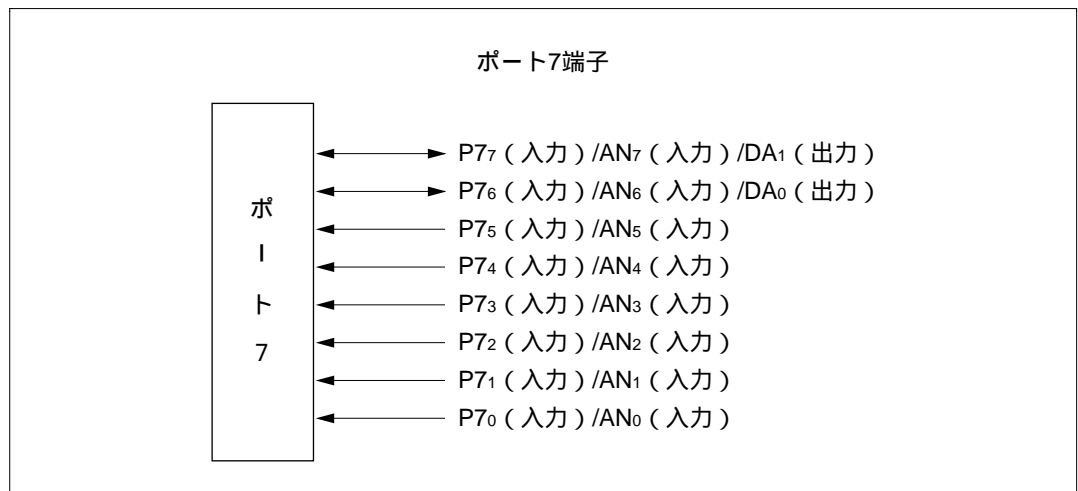


図8.7 ポート7の端子構成

### 8.8.2 レジスタ構成

表8.12にポート7のレジスタ構成を示します。ポート7は入力専用ポートであり、データディレクションレジスタはありません。

表8.12 ポート7レジスタ構成

アドレス*	名称	略称	R/W	初期値
H' FFFD6	ポート7データレジスタ	P7DR	R	不定

【注】 \* アドバンスモード時のアドレス下位20ビットを示しています。

## (1) ポート7 データレジスタ (P7DR)

ビット:	7	6	5	4	3	2	1	0
	P77	P76	P75	P74	P73	P72	P71	P70
初期値:	—	*	—	*	—	*	—	*
R/W :	R	R	R	R	R	R	R	R

【注】 \* P7<sub>7</sub>~P7<sub>0</sub>端子により決定されます。

P7DR のリードを行うと、常に端子のロジックレベルが読み出されます。ライトは無効です。

## 8.9 ポート 8

### 8.9.1 概要

ポート 8 は、5 ビットの入出力ポートです。ポート 8 は、 $\overline{CS}_3 \sim \overline{CS}_0$  出力端子、 $\overline{RFSH}$  出力端子、 $\overline{IRQ}_3 \sim \overline{IRQ}_0$  入力端子、A/D 変換器の  $\overline{ADTRG}$  入力端子と兼用になっています。ポート 8 の端子構成を図 8.8 に示します。

モード 1~5 (拡張モード) 時には、ポート 8 は、 $\overline{CS}_3 \sim \overline{CS}_0$  出力端子、 $\overline{RFSH}$  出力端子、 $\overline{IRQ}_3 \sim \overline{IRQ}_0$  入力端子、 $\overline{ADTRG}$  入力端子と兼用になります。拡張モードでの端子機能の選択方法については表 8.14 を参照してください。

モード 6、7 (シングルチップモード) 時には、ポート 8 は、 $\overline{IRQ}_3 \sim \overline{IRQ}_0$  入力端子、 $\overline{ADTRG}$  入力端子と兼用になります。シングルチップモードでの端子機能の選択方法については表 8.15 を参照してください。

A/D 変換器の  $\overline{ADTRG}$  入力端子については「第 15 章 A/D 変換器」を参照してください。 $\overline{IRQ}_3 \sim \overline{IRQ}_0$  の機能は入出力にかかわらず IER をセットすることにより選択されます。そのため注意が必要です。詳細は「5.3.1 外部割込み」を参照してください。

エリア 2、3、4、5 に DRAM を接続する場合には、 $\overline{CS}_3$ 、 $\overline{CS}_2$  出力端子が各エリアに対応した  $\overline{RAS}$  の出力端子となります。詳細は「6.5 DRAM インタフェース」を参照してください。

ポート 8 は、1 個の TTL 負荷と 90pF の容量負荷を駆動することができます。また、ダーリントントランジスタを駆動することができます。

P8<sub>2</sub> ~ P8<sub>0</sub> はシュミットトリガ入力です。



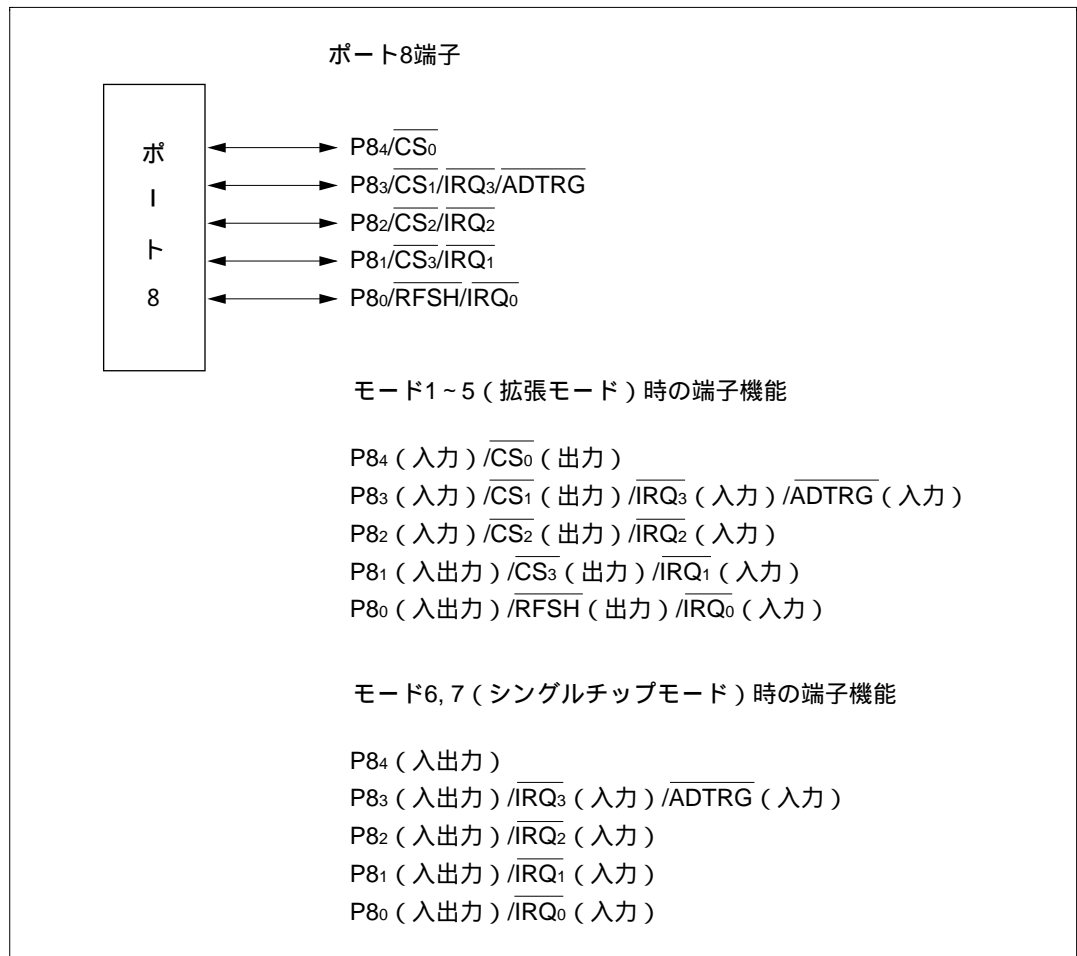


図 8.8 ポート 8 の端子構成

### 8.9.2 レジスタ構成

表 8.13 にポート 8 のレジスタ構成を示します。

表 8.13 ポート 8 レジスタ構成

アドレス*	名 称	略 称	R/W	初期値	
				モード 1~4	モード 5~7
H'EE007	ポート 8 データディレクションレジスタ	P8DDR	W	H'F0	H'E0
H'FFFD7	ポート 8 データレジスタ	P8DR	R/W	H'E0	

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

## (1) ポート 8 データディレクションレジスタ (P8DDR)

P8DDR は、8 ビットのライト専用のレジスタで、ポート 8 各端子の入出力をビットごとに指定することができます。

ビット 7~ビット 5 はリザーブビットで 1 に固定されています。ライトは無効です。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	P8 <sub>4</sub> DDR	P8 <sub>3</sub> DDR	P8 <sub>2</sub> DDR	P8 <sub>1</sub> DDR	P8 <sub>0</sub> DDR
モード 1~4	初期値:	1	1	1	1	0	0	0
	R/W:	—	—	—	W	W	W	W
モード 5~7	初期値:	1	1	1	0	0	0	0
	R/W:	—	—	—	W	W	W	W

リザーブビット
ポート 8 データディレクション 4~0  
ポート 8 の各端子の入出力を選択するビットです。

## (a) モード 1~5 (拡張モード)

P8<sub>4</sub>~P8<sub>1</sub>はP8DDRの対応するビットが1のとき $\overline{CS}_0$ ~ $\overline{CS}_3$ 出力端子となり、0のとき入力ポートとなります。ただしP8<sub>1</sub>はDRAMコントロールレジスタA(DRCRA)のDRAS2~DRAS0ビットの設定によっては出力ポートとしても使用できます。詳細は「6.5.2 DRAM空間とRAS出力端子の設定」を参照してください。

モード 1~4 (内蔵 ROM 無効拡張モード) ではリセット直後は P8<sub>4</sub>のみ $\overline{CS}_0$ 出力となり、 $\overline{CS}_1$ ~ $\overline{CS}_3$ は入力ポートとなります。モード 5 (内蔵 ROM 有効拡張モード) ではリセット直後 $\overline{CS}_0$ ~ $\overline{CS}_3$ 共入力ポートになります。

DRCRAのリフレッシュ端子イネーブル(RFSHE)を1にセットするとP8<sub>0</sub>は $\overline{RFSH}$ 出力となり、0にクリアすると入出力ポートとなってP8DDRの設定値に従います。詳細は表 8.14 を参照してください。

## (b) モード 6、7 (シングルチップモード)

入出力ポートとして機能します。P8DDRを1にセットすると対応する端子は出力ポートとなり、0にクリアすると入力ポートとなります。

P8DDRは、ライト専用レジスタで、リードは無効です。リードすると1が読み出されます。

P8DDRは、リセット、またはハードウェアスタンバイモード時に、モード 1~4 の場合はH'F0に、モード 5~7 の場合はH'E0にイニシャライズされます。P8DDRは、ソフトウェアスタンバイモード時には直前の状態を保持します。そのため、ポート 8 が入出力ポートとして機能しているときP8DDRが1にセットされた状態でソフトウェアスタンバイモ

ードに遷移すると、その端子は出力状態のままとなっています。

## (2) ポート 8 データレジスタ (P8DR)

P8DR は、8 ビットのリード/ライト可能なレジスタで、ポート 8 の出力データを格納します。ポート 8 が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、P8DDR が 0 のビットは端子のロジックレベルが読み出され、1 のビットは P8DR の値が読み出されます。

ビット 7~ビット 5 はリザーブビットで 1 に固定されています。ライトは無効です。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	P84	P83	P82	P81	P80
初期値:	1	1	1	0	0	0	0	0
R/W :	—	—	—	R/W	R/W	R/W	R/W	R/W

リザーブビット
ポート8データ4 ~ 0  
ポート8の各端子のデータを格納する  
ビットです。

P8DR は、リセット、またはハードウェアスタンバイモード時に、H'E0 にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

表 8.14 モード 1～5 の端子機能 (ポート 8)

端子	選択方法と端子機能							
P8 <sub>4</sub> / $\overline{CS}_0$	P8 <sub>4</sub> DDR ビットにより、次のように切り換わります。							
	P8 <sub>4</sub> DDR	0		1				
	端子機能	P8 <sub>4</sub> 入力端子		$\overline{CS}_0$ 出力端子				
P8 <sub>3</sub> / $\overline{CS}_1$ / $\overline{IRQ}_3$ / $\overline{ADTRG}$	P8 <sub>3</sub> DDR ビットにより、次のように切り換わります。							
	P8 <sub>3</sub> DDR	0		1				
	端子機能	P8 <sub>3</sub> 入力端子		$\overline{CS}_1$ 出力端子				
		$\overline{IRQ}_3$ 入力端子						
		$\overline{ADTRG}$ 入力端子						
P8 <sub>2</sub> / $\overline{CS}_2$ / $\overline{IRQ}_2$	DRCRA の DRAS2、DRAS1、DRAS0 ビットによる DRAM インタフェースの設定、および P8 <sub>2</sub> DDR ビットにより、次のように切り換わります。							
	DRAM インタフェースの設定	下表 (1)		下表 (2)				
		P8 <sub>2</sub> DDR	0	1	-			
	端子機能	P8 <sub>2</sub> 入力端子	$\overline{CS}_2$ 出力端子	$\overline{CS}_2$ 出力端子*				
		$\overline{IRQ}_3$ 入力端子						
	【注】* $\overline{RAS}_2$ として $\overline{CS}_2$ 出力します。							
DRAM インタフェースの設定	(1)	(2)						
	DRAS2	0				1		
DRAS1	0		1		0		1	
DRAS1	0	1	0	1	0	1	0	1

端子	選択方法と端子機能						
P8 <sub>1</sub> / $\overline{CS}_3$ / $\overline{IRQ}_1$	DRCRAのDRAS2、DRAS1、DRAS0ビットによるDRAMインタフェースの設定、およびP8 <sub>1</sub> DDRビットにより、次のように切り換わります。						
	DRAMインタフェースの設定	下表(1)		下表(2)		下表(3)	
	P8 <sub>1</sub> DDR	0	1	0	1		
	端子機能	P8 <sub>1</sub> 入力端子	$\overline{CS}_3$ 出力端子	P8 <sub>1</sub> 入力端子	P8 <sub>1</sub> 出力端子	$\overline{CS}_3$ 出力端子*	
		$\overline{IRQ}_1$ 入力端子					
	【注】* $\overline{RAS}_3$ として $\overline{CS}_3$ 出力します。						
DRAMインタフェースの設定	(1)	(3)	(2)	(3)	(2)		
DRAS2	0			1			
DRAS1	0	1	0	1			
DRAS0	0	1	0	1	0	1	
P8 <sub>0</sub> / $\overline{RFSH}$ / $\overline{IRQ}_0$	DRCRAのRFSHEビットとP8 <sub>0</sub> DDRビットの組み合わせにより、次のように切り換わります。						
	RFSHE	0		1*			
	P8 <sub>0</sub> DDR	0	1	-			
	端子機能	P8 <sub>0</sub> 入力端子	P8 <sub>0</sub> 出力端子	$\overline{RFSH}$ 出力端子			
		$\overline{IRQ}_0$ 入力端子					
【注】* エリア2~5をDRAM空間に設定しない場合は、1にセットしないでください。							

表 8.15 モード 6、7 の端子機能 (ポート 8)

端子	選択方法と端子機能		
P8 <sub>4</sub>	P8 <sub>4</sub> DDR ビットにより、次のように切り換わります。		
	P8 <sub>4</sub> DDR	0	1
	端子機能	P8 <sub>4</sub> 入力端子	P8 <sub>4</sub> 出力端子
P8 <sub>3</sub> / $\overline{\text{IRQ}}_3$ / $\overline{\text{ADTRG}}$	P8 <sub>3</sub> DDR ビットにより、次のように切り換わります。		
	P8 <sub>3</sub> DDR	0	1
	端子機能	P8 <sub>3</sub> 入力端子	P8 <sub>3</sub> 出力端子
		$\overline{\text{IRQ}}_3$ 入力端子	
		$\overline{\text{ADTRG}}$ 入力端子	
P8 <sub>2</sub> / $\overline{\text{IRQ}}_2$	P8 <sub>2</sub> DDR ビットにより、次のように切り換わります。		
	P8 <sub>2</sub> DDR	0	1
	端子機能	P8 <sub>2</sub> 入力端子	P8 <sub>2</sub> 出力端子
$\overline{\text{IRQ}}_2$ 入力端子			
P8 <sub>1</sub> / $\overline{\text{IRQ}}_1$	P8 <sub>1</sub> DDR ビットにより、次のように切り換わります。		
	P8 <sub>1</sub> DDR	0	1
	端子機能	P8 <sub>1</sub> 入力端子	P8 <sub>1</sub> 出力端子
$\overline{\text{IRQ}}_1$ 入力端子			
P8 <sub>0</sub> / $\overline{\text{IRQ}}_0$	P8 <sub>0</sub> DDR ビットにより、次のように切り換わります。		
	P8 <sub>0</sub> DDR	0	1
	端子機能	P8 <sub>0</sub> 入力端子	P8 <sub>0</sub> 出力端子
$\overline{\text{IRQ}}_0$ 入力端子			

## 8.10 ポート9

### 8.10.1 概要

ポート9は、6ビットの入出力ポートです。ポート9はシリアルコミュニケーションインタフェースチャンネル0、1 (SCI0、SCI1) の入出力端子 (TxD<sub>0</sub>、TxD<sub>1</sub>、RxD<sub>0</sub>、RxD<sub>1</sub>、SCK<sub>0</sub>、SCK<sub>1</sub>)、 $\overline{\text{IRQ}}_5$ 、 $\overline{\text{IRQ}}_4$  入力端子と兼用になっています。

端子機能の選択方法については表 8.17 を参照してください。

$\overline{\text{IRQ}}_5 \sim \overline{\text{IRQ}}_4$  の機能は入出力にかかわらず IER をセットすることにより選択されます。そのため注意が必要です。

詳細は「5.3.1 外部割込み」を参照してください。

ポート9の端子機能はいずれの動作モードでも共通です。ポート9の端子構成を図 8.9 に示します。

ポート9は、1個のTTL負荷と30pFの容量負荷を駆動することができます。また、ダーリントントランジスタを駆動することができます。

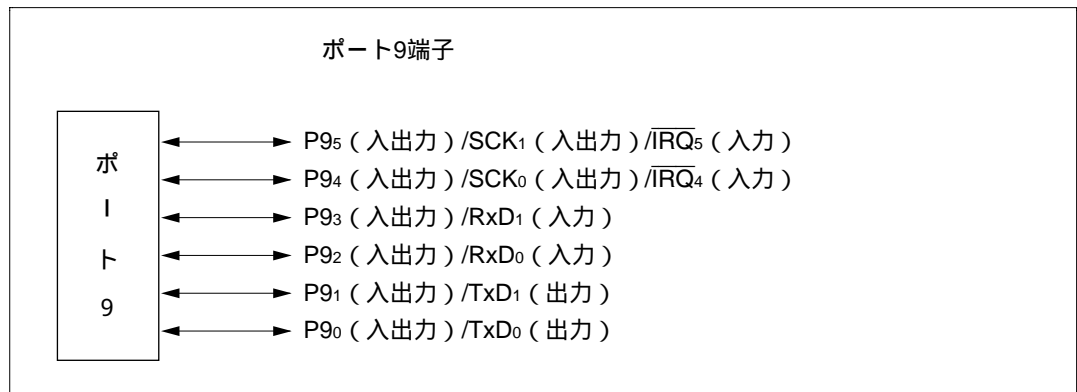


図 8.9 ポート9の端子構成

## 8.10.2 レジスタ構成

表 8.16 にポート 9 のレジスタ構成を示します。

表 8.16 ポート 9 レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H'EE008	ポート 9 データディレクションレジスタ	P9DDR	W	H'C0
H'FFFD8	ポート 9 データレジスタ	P9DR	R/W	H'C0

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

### (1) ポート 9 データディレクションレジスタ (P9DDR)

P9DDR は、8 ビットのライト専用のレジスタで、ポート 9 各端子の入出力をビットごとに指定することができます。

ビット 7、ビット 6 はリザーブビットで 1 に固定されています。ライトは無効です。

ビット:	7	6	5	4	3	2	1	0
	—	—	P9 <sub>5</sub> DDR	P9 <sub>4</sub> DDR	P9 <sub>3</sub> DDR	P9 <sub>2</sub> DDR	P9 <sub>1</sub> DDR	P9 <sub>0</sub> DDR
初期値:	1	1	0	0	0	0	0	0
R/W :	—	—	W	W	W	W	W	W
	リザーブビット		ポート 9 データディレクション 5~0 ポート 9 の各端子の入出力を選択するビットです。					

ポート 9 が入出力ポートとして機能している場合、P9DDR を 1 にセットすると対応するポート 9 の各端子は出力ポートとなり、0 にクリアすると入力ポートになります。端子機能の選択方法については、表 8.17 を参照してください。

P9DDR は、ライト専用レジスタで、リードは無効です。リードすると 1 が読み出されます。

P9DDR は、リセット、またはハードウェアスタンバイモード時に、H'C0 にイニシャライズされます。ソフトウェアスタンバイモード時には直前の状態を保持します。そのため、ポート 9 が入出力ポートとして機能しているとき、P9DDR が、1 にセットされた状態でソフトウェアスタンバイモードに遷移すると、その端子は出力状態のままとなっています。



## (2) ポート9データレジスタ (P9DR)

P9DRは、8ビットのリード/ライト可能なレジスタで、ポート9の出力データを格納します。ポート9が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、P9DDRが0のビットは端子のロジックレベルが読み出され、1のビットはP9DRの値が読み出されます。

ビット7、6はリザーブビットで1に固定されています。ライトは無効です。

ビット:	7	6	5	4	3	2	1	0
	—	—	P95	P94	P93	P92	P91	P90
初期値:	1	1	0	0	0	0	0	0
R/W :	—	—	R/W	R/W	R/W	R/W	R/W	R/W

リザーブビット

ポート9データ5~0  
ポート9の各端子のデータを格納するビットです。

P9DRは、リセット、またはハードウェアスタンバイモード時に、H'COにイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

表 8.17 ポート 9 の端子機能

端子	選択方法と端子機能					
P9 <sub>5</sub> /SCK <sub>1</sub> / $\overline{\text{IRQ}}_5$	SCI1 の SMR の C/ $\overline{\text{A}}$ ビット、SCR の CKE0、CKE1 ビットと P9 <sub>5</sub> DDR ビットの組合せにより、次のように切り換わります。					
	CKE1	0			1	
	C/ $\overline{\text{A}}$	0		1	-	
	CKE0	0		1	-	-
	P9 <sub>5</sub> DDR	0	1	-	-	-
	端子機能	P9 <sub>5</sub> 入力 端子	P9 <sub>5</sub> 出力 端子	SCK <sub>1</sub> 出力 端子	SCK <sub>1</sub> 出力端子	SCK <sub>1</sub> 入力 端子
$\overline{\text{IRQ}}_5$ 入力端子						
P9 <sub>4</sub> /SCK <sub>0</sub> / $\overline{\text{IRQ}}_4$	SCI0 の SMR の C/ $\overline{\text{A}}$ ビット、SCR の CKE0、CKE1 ビットと P9 <sub>4</sub> DDR ビットの組合せにより、次のように切り換わります。					
	CKE1	0			1	
	C/ $\overline{\text{A}}$	0		1	-	
	CKE0	0		1	-	-
	P9 <sub>4</sub> DDR	0	1	-	-	-
	端子機能	P9 <sub>4</sub> 入力 端子	P9 <sub>4</sub> 出力 端子	SCK <sub>0</sub> 出力 端子	SCK <sub>0</sub> 出力端子	SCK <sub>0</sub> 入力 端子
$\overline{\text{IRQ}}_4$ 入力端子						
P9 <sub>3</sub> /RxD <sub>1</sub>	SCI1 の SCR の RE ビットと SCMR の SMIF ビットと P9 <sub>3</sub> DDR ビットの組合せにより、次のように切り換わります。					
	SMIF	0			1	
	RE	0		1	-	
	P9 <sub>3</sub> DDR	0	1	-	-	
	端子機能	P9 <sub>3</sub> 入力端子	P9 <sub>3</sub> 出力端子	RxD <sub>1</sub> 入力端子	RxD <sub>1</sub> 入力端子	
P9 <sub>2</sub> /RxD <sub>0</sub>	SCI0 の SCR の RE ビットと SCMR の SMIF ビットと P9 <sub>2</sub> DDR ビットの組合せにより、次のように切り換わります。					
	SMIF	0			1	
	RE	0		1	-	
	P9 <sub>2</sub> DDR	0	1	-	-	
	端子機能	P9 <sub>2</sub> 入力端子	P9 <sub>2</sub> 出力端子	RxD <sub>0</sub> 入力端子	RxD <sub>0</sub> 入力端子	

端子	選択方法と端子機能			
P9 <sub>1</sub> /TxD <sub>1</sub>	SCI1 の SCR の TE ビットと、SCMR の SMIF ビット、および P9 <sub>1</sub> DDR ビットの組合せにより、次のように切り換わります。			
	SMIF	0		1
	TE	0	1	-
	P9 <sub>1</sub> DDR	0	1	-
	端子機能	P9 <sub>1</sub> 入力端子	P9 <sub>1</sub> 出力端子	TxD <sub>1</sub> 出力端子
【注】* TxD <sub>1</sub> 出力端子として機能します。ただしハイインピーダンス状態と端子ドライブ状態の2種類の状態があります。				
P9 <sub>0</sub> /TxD <sub>0</sub>	SCI0 の SCR の TE ビット、SCMR の SMIF ビット、および P9 <sub>0</sub> DDR ビットの組合せにより、次のように切り換わります。			
	SMIF	0		1
	TE	0	1	-
	P9 <sub>0</sub> DDR	0	1	-
	端子機能	P9 <sub>0</sub> 入力端子	P9 <sub>0</sub> 出力端子	TxD <sub>0</sub> 出力端子
【注】* TxD <sub>0</sub> 出力端子として機能します。ただし、ハイインピーダンス状態と端子ドライブ状態の2種類の状態があります。				

## 8.11 ポート A

### 8.11.1 概要

ポート A は、8 ビットの入出力ポートです。ポート A はプログラマブルタイミングパターンコントローラ (TPC) の出力端子 (TP<sub>7</sub> ~ TP<sub>0</sub>)、16 ビットタイマの入出力端子 (TIOCB<sub>2</sub>、TIOCA<sub>2</sub>、TIOCB<sub>1</sub>、TIOCA<sub>1</sub>、TIOCB<sub>0</sub>、TIOCA<sub>0</sub>、TCLKD、TCLKC、TCLKB、TCLKA)、8 ビットタイマのクロック入力端子 (TCLKD、TCLKC、TCLKB、TCLKA)、DMA コントローラ (DMAC) の出力端子 ( $\overline{\text{TEND}}_1$ 、 $\overline{\text{TEND}}_0$ )、アドレスバス (A<sub>23</sub> ~ A<sub>20</sub>) と兼用になっています。ポート A は、動作モード 3、4 で A<sub>20</sub> が強制的に出力となることを除き、リセットおよびハードウェアスタンバイモードで入力ポートとなっています。端子機能の選択方法については表 8.19 ~ 表 8.21 を参照してください。

TPC、16 ビットタイマ、8 ビットタイマ、DMAC の入出力端子として使用する端子については、それぞれのモジュールの説明を参照してください。モード 3、4、5 で A<sub>23</sub> ~ A<sub>20</sub> を出力する場合は、「6.2.4 バスリリースコントロールレジスタ」を参照してください。これらいずれの機能も割当てられない端子は入出力ポートとして使用できます。ポート A の端子構成を図 8.10 に示します

ポート A は、1 個の TTL 負荷と 30pF の容量負荷を駆動することができます。また、ダーリントントランジスタを駆動することができます。ポート A は、シュミットトリガ入力です。

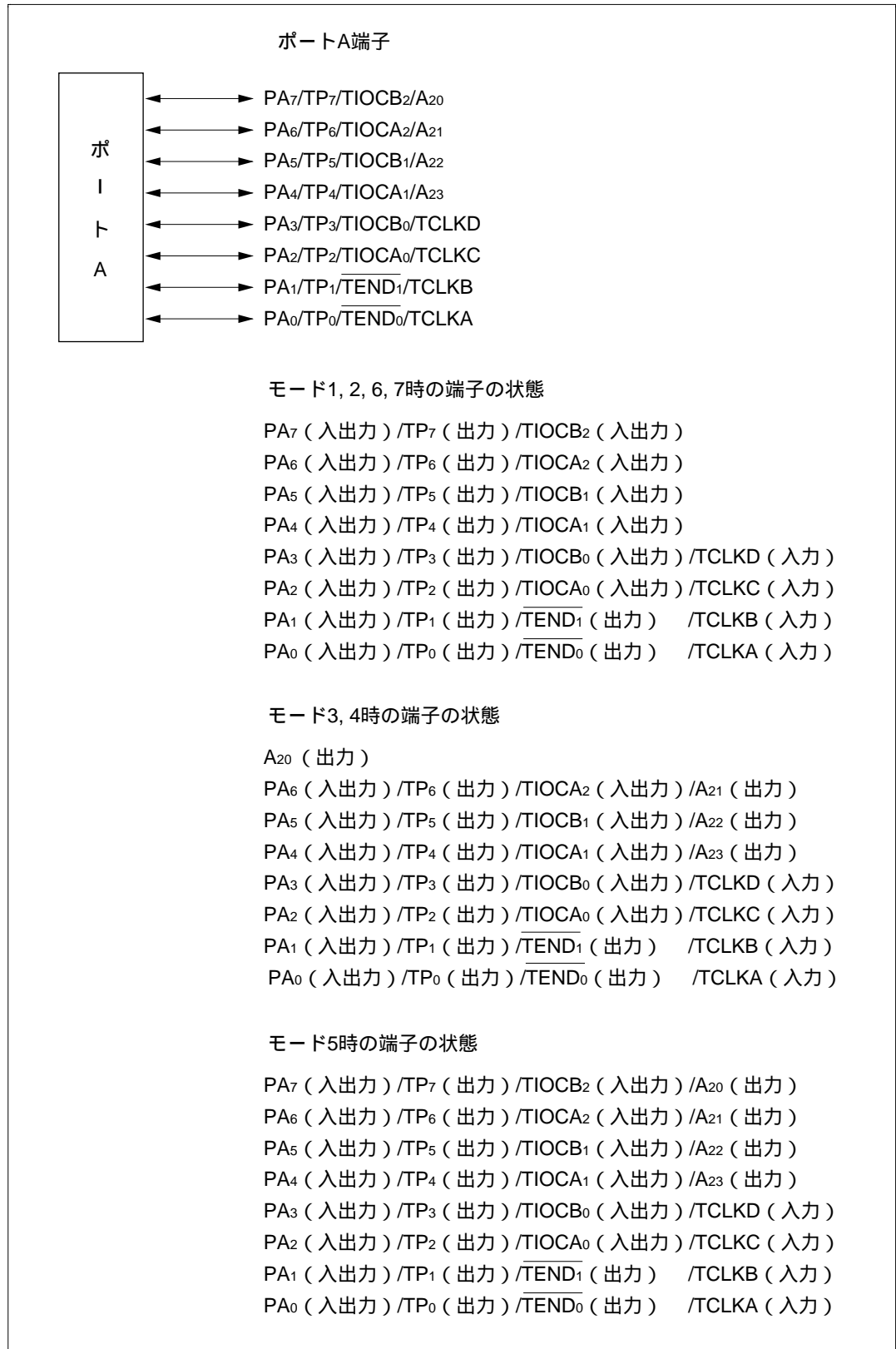


図 8.10 ポート A の端子構成

## 8.11.2 レジスタ構成

表 8.18 にポート A のレジスタ構成を示します。

表 8.18 ポート A レジスタ構成

アドレス*	名 称	略 称	R/W	初期値	
				モード 1、2、5、6、 7	モード 3、4
H'EE009	ポート A データディレクションレジスタ	PADDR	W	H'00	H'80
H'FFFD9	ポート A データレジスタ	PADR	R/W	H'00	

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

### (1) ポート A データディレクションレジスタ (PADDR)

PADDR は、8 ビットのライト専用のレジスタで、ポート A 各端子の入出力をビットごとに指定することができます。TPC の出力端子として使用する場合も PADDR の対応するビットをセットしてください。

ビット:		7	6	5	4	3	2	1	0
		PA <sub>7</sub> DDR	PA <sub>6</sub> DDR	PA <sub>5</sub> DDR	PA <sub>4</sub> DDR	PA <sub>3</sub> DDR	PA <sub>2</sub> DDR	PA <sub>1</sub> DDR	PA <sub>0</sub> DDR
モード3,4	初期値:	1	0	0	0	0	0	0	0
	R/W:	—	W	W	W	W	W	W	W
モード1,2,5,6,7	初期値:	0	0	0	0	0	0	0	0
	R/W:	W	W	W	W	W	W	W	W

ポートAデータディレクション7~0  
ポートAの各端子の入出力を選択するビットです。

PA<sub>7</sub>~PA<sub>4</sub>はモード 1、2、6、7 の場合と、モード 3~5 の場合で、選択できる端子機能が異なります。端子機能の選択方法については、表 8.19 および表 8.20 を参照してください。

PA<sub>3</sub>~PA<sub>0</sub>については、選択できる端子機能はモード 1~7 で共通です。端子機能の選択方法については表 8.21 を参照してください。

ポート A が入出力ポートとして機能している場合、PADDR を 1 にセットすると対応するポート A の各端子は出力ポートとなり、0 にクリアすると入力ポートになります。ただし、モード 3、4 では PA<sub>7</sub>DDR は 1 に固定され、PA<sub>7</sub> はアドレス A<sub>20</sub> 出力として機能します。

PADDR は、ライト専用レジスタで、リードは無効です。リードすると 1 が読み出されます。

PADDR は、リセット、またはハードウェアスタンバイモード時に、モード 1、2、5、6、

7ではH'00に、モード3、4ではH'80にイニシャライズされます。ソフトウェアスタンバイモード時には直前の状態を保持します。そのため、ポートAが入出力ポートとして機能しているとき、PADDRが1にセットされた状態でソフトウェアスタンバイモードに遷移すると、その端子は出力状態のままとなっています。

## (2) ポートAデータレジスタ(PADR)

PADRは、8ビットのリード/ライト可能なレジスタで、ポートAの出力データを格納します。ポートAが出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、PADDRが0のビットは端子のロジックレベルが読み出され、1のビットはPADRの値が読み出されます。

ビット:	7	6	5	4	3	2	1	0
	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポートAデータ7~0  
ポートAの各端子のデータを格納するビットです。

PADRは、リセット、またはハードウェアスタンバイモード時に、H'00にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

表 8.19 ポート A (モード 1、2、6、7) の端子機能

端子	選択方法と端子機能				
PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub>	TMDR の PWM2 ビット、TIOB2 の IOB2 ~ IOB0 ビット、NDERA の NDER7 ビットと PA <sub>7</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	16 ビットタイマ チャンネル 2 の設定	下表 (1)		下表 (2)	
	PA <sub>7</sub> DDR	-	0	1	1
	NDER7	-	-	0	1
	端子機能	TIOCB <sub>2</sub> 出力端子	PA <sub>7</sub> 入力端子	PA <sub>7</sub> 出力端子	TP <sub>7</sub> 出力端子
	TIOCB <sub>2</sub> 入力端子*				
	【注】* IOB2 = 1、かつ PWM2 = 0 のとき、TIOCB <sub>2</sub> 入力端子となります。				
	16 ビットタイマ チャンネル 2 の設定	(2)	(1)		(2)
	IOB2	0			1
	IOB1	0	0	1	-
IOB0	0	1	-	-	
PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub>	TMDR の PWM2 ビット、TIOB2 の IOA2 ~ IOA0 ビット、NDERA の NDER6 ビットと PA <sub>6</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	16 ビットタイマ チャンネル 2 の設定	下表 (1)		下表 (2)	
	PA <sub>6</sub> DDR	-	0	1	1
	NDER6	-	-	0	1
	端子機能	TIOCA <sub>2</sub> 出力端子	PA <sub>6</sub> 入力端子	PA <sub>6</sub> 出力端子	TP <sub>6</sub> 出力端子
	TIOCA <sub>2</sub> 入力端子*				
	【注】* IOA2 = 1 のとき、TIOCA <sub>2</sub> 入力端子となります。				
	16 ビットタイマ チャンネル 2 の設定	(2)	(1)	(2)	(1)
	PWM2	0			1
	IOA2	0		1	-
IOA1	0	0	1	-	
IOA0	0	1	-	-	



端子	選択方法と端子機能				
PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub>	TMDR の PWM1 ビット、TIOR1 の IOB2 ~ IOB0 ビット、NDERA の NDER5 ビットと PA <sub>5</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	16 ビットタイマ チャンネル 1 の設定	下表 (1)		下表 (2)	
	PA <sub>5</sub> DDR	-	0	1	1
	NDER5	-	-	0	1
	端子機能	TIOCB <sub>1</sub> 出力端子	PA <sub>5</sub> 入力端子	PA <sub>5</sub> 出力端子	TP <sub>5</sub> 出力端子
	TIOCB <sub>1</sub> 入力端子*				
	【注】* IOB2 = 1、かつ PWM1 = 0 のとき、TIOCB <sub>1</sub> 入力端子となります。				
	16 ビットタイマ チャンネル 1 の設定	(2)	(1)		(2)
	IOB2	0			1
	IOB1	0	0	1	-
IOB0	0	1	-	-	
PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub>	TMDR の PWM1 ビット、TIOR1 の IOA2 ~ IOA0 ビット、NDERA の NDER4 ビットと PA <sub>4</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	16 ビットタイマ チャンネル 1 の設定	下表 (1)		下表 (2)	
	PA <sub>4</sub> DDR	-	0	1	1
	NDER4	-	-	0	1
	端子機能	TIOCA <sub>1</sub> 出力端子	PA <sub>4</sub> 入力端子	PA <sub>4</sub> 出力端子	TP <sub>4</sub> 出力端子
	TIOCA <sub>1</sub> 入力端子*				
	【注】* IOA2 = 1 のとき、TIOCA <sub>1</sub> 入力端子となります。				
	16 ビットタイマ チャンネル 1 の設定	(2)	(1)	(2)	(1)
	PWM1	0			1
	IOA2	0		1	-
IOA1	0	0	1	-	
IOA0	0	1	-	-	

表 8.20 ポート A (モード 3、4、5) の端子機能

端子	選択方法と端子機能					
PA <sub>7</sub> /TP <sub>7</sub> / TIOCB <sub>2</sub> /A <sub>20</sub>	モード 3、4 の場合： A <sub>20</sub> が強制的に出力されます。					
	端子機能	A <sub>20</sub> 出力端子				
	モード 5 の場合： TMDR の PWM2 ビット、TIOB2 の IOB2 ~ IOB0 ビット、NDERA の NDER7 ビット、BRCCR の A20E ビットと PA <sub>7</sub> DDR ビットの組み合わせにより、次のように切り換わります。					
	A20E	1			0	
	16 ビットタイマ チャンネル 2 の設定	下表 (1)	下表 (2)			-
	PA <sub>7</sub> DDR	-	0	1	1	-
	NDER7	-	-	0	1	-
	端子機能	TIOCB <sub>2</sub> 出力端子	PA <sub>7</sub> 入力端子	PA <sub>7</sub> 出力端子	TP <sub>7</sub> 出力端子	A <sub>20</sub> 出力端子
	TIOCB <sub>2</sub> 入力端子*					
	【注】* IOB2=1 かつ PWM2=0 のとき TIOCB <sub>2</sub> 入力端子となります					
16 ビットタイマチャンネル 2 の設定	(2)	(1)		(2)		
IOB2	0			1		
IOB1	0	0	1	-		
IOB0	0	1	-	-		
PA <sub>6</sub> /TP <sub>6</sub> / TIOCA <sub>2</sub> /A <sub>21</sub>	TMDR の PWM2 ビット、TIOB2 の IOA2 ~ IOA0 ビット、NDERA の NDER6 ビット、BRCCR の A21E ビットと PA <sub>6</sub> DDR ビットの組み合わせにより、次のように切り換わります。					
	A21E	1			0	
	16 ビットタイマ チャンネル 2 の設定	下表 (1)	下表 (2)			-
	PA <sub>6</sub> DDR	-	0	1	1	-
	NDER6	-	-	0	1	-
	端子機能	TIOCA <sub>2</sub> 出力端子	PA <sub>6</sub> 入力端子	PA <sub>6</sub> 出力端子	TP <sub>6</sub> 入力端子	A <sub>21</sub> 出力端子
	TIOCA <sub>2</sub> 入力端子*					
	【注】* IOA2=1 のとき、TIOCA <sub>2</sub> 入力端子となります。					
	16 ビットタイマチャンネル 2 の設定	(2)	(1)		(2)	(1)
	PWM2	0			1	
IOA2	0		1		-	
IOA1	0	0	1	-		
IOA0	0	1	-	-		

端子	選択方法と端子機能					
PA <sub>5</sub> /TP <sub>5</sub> / TIOCB <sub>1</sub> /A <sub>22</sub>	TMDR の PWM1 ビット、TIOR1 の IOB2 ~ IOB0 ビット、NDERA の NDER5 ビット、BRCR の A22E ビットと PA <sub>5</sub> DDR ビットの組合わせにより、次のように切り換わります。					
	A22E	1			0	
	16 ビットタイマ チャンネル 1 の設定	下表 (1)	下表 (2)			-
	PA <sub>5</sub> DDR	-	0	1	1	-
	NDER5	-	-	0	1	-
	端子機能	TIOCB <sub>1</sub> 出力端子	PA <sub>5</sub> 入力端子	PA <sub>5</sub> 出力端子	TP <sub>5</sub> 出力端子	A <sub>22</sub> 出力端子
	TIOCB <sub>1</sub> 入力端子*					
	【注】* IOB2 = 1、かつ PWM1 = 0 のとき、TIOCB <sub>1</sub> 入力端子となります。					
	16 ビットタイマ チャンネル 1 の設定	(2)	(1)		(2)	
	IOB2	0			1	
IOB1	0	0	1	-		
IOB0	0	1	-	-		
PA <sub>4</sub> /TP <sub>4</sub> / TIOCA <sub>1</sub> /A <sub>23</sub>	TMDR の PWM1 ビット、TIOR1 の IOA2 ~ IOA0 ビット、NDERA の NDER4 ビット、BRCR の A23E ビットと PA <sub>4</sub> DDR ビットの組合わせにより、次のように切り換わります。					
	A23E	1			0	
	16 ビットタイマ チャンネル 1 の設定	下表 (1)	下表 (2)			-
	PA <sub>4</sub> DDR	-	0	1	1	-
	NDER4	-	-	0	1	-
	端子機能	TIOCA <sub>1</sub> 出力端子	PA <sub>4</sub> 入力端子	PA <sub>4</sub> 出力端子	TP <sub>4</sub> 出力端子	A <sub>23</sub> 出力端子
	TIOCA <sub>1</sub> 入力端子*					
	【注】* IOA2 = 1 のとき、TIOCA <sub>1</sub> 入力端子となります。					
	16 ビットタイマ チャンネル 1 の設定	(2)	(1)	(2)	(1)	
	PWM1	0			1	
IOA2	0		1	-		
IOA1	0	0	1	-	-	
IOA0	0	1	-	-	-	

表 8.21 ポート A (モード 1~7) の端子機能

端子	選択方法と端子機能			
PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> / TCLKD	TMDR の PWM0 ビット、TIOR0 の IOB2 ~ IOB0 ビット、16 ビットタイマの 16TCR2 ~ 16TCR0 の TPSC2 ~ TPSC0 ビット、8 ビットタイマの 8TCR2 の CKS2 ~ CKS0 ビット、NDERA の NDER3 ビットと PA <sub>3</sub> DDR ビットの組合わせにより、次のように切り換わります。			
	16 ビットタイマ チャンネル 0 の設定	下表 (1)	下表 (2)	
	PA <sub>3</sub> DDR	-	0	1
	NDER3	-	-	0
	端子機能	TIOCB <sub>0</sub> 出力端子	PA <sub>3</sub> 入力端子	PA <sub>3</sub> 出力端子
			TP <sub>3</sub> 出力端子	
			TIOCB <sub>0</sub> 入力端子* <sup>1</sup>	
			TCLKD 入力端子* <sup>2</sup>	
	【注】*1 IOB2 = 1、かつ PWM0 = 0 のとき、TIOCB <sub>0</sub> 入力端子となります。			
	*2 16TCR2 ~ 16TCR0 のいずれかの設定が TPSC2 = TPSC1 = TPSC0 = 1 または、8TCR2 の CKS2 ~ CKS0 ビットが下表 (3) のとき TCLKD 入力端子となります。			
	16 ビットタイマ チャンネル 0 の設定	(2)	(1)	(2)
	IOB2	0		1
	IOB1	0	0	1
	IOB0	0	1	-
	8 ビットタイマ チャンネル 0 の設定	(4)	(3)	
	CKS2	0	1	
	CKS1	-	0	1
	CKS0	-	0	1

端子	選択方法と端子機能					
PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> / TCLKC	TMDR の PWM0 ビット、TIOR0 の IOA2 ~ IOA0 ビット、16 ビットタイマの 16TCR2 ~ 16TCR0 の TPSC2 ~ TPSC0 ビット、8 ビットタイマの 8TCR0 の CKS2 ~ CKS0 ビット、NDERA の NDER2 ビットと PA <sub>2</sub> DDR ビットの組合わせにより、次のように切り換わります。					
	16 ビットタイマ チャンネル0の設定	下表 (1)		下表 (2)		
	PA <sub>2</sub> DDR	-	0	1	1	
	NDER2	-	-	0	1	
	端子機能	TIOCA <sub>0</sub> 出力端子	PA <sub>2</sub> 入力端子	PA <sub>2</sub> 出力端子	TP <sub>2</sub> 出力端子	
		TIOCA <sub>0</sub> 入力端子* <sup>1</sup>				
		TCLKC入力端子* <sup>2</sup>				
<p>【注】*1 IOA2 = 1 のとき、TIOCA<sub>0</sub>入力端子となります。</p> <p>*2 16TCR2 ~ 16TCR0 のいずれかの設定が TPSC2 = TPSC1 = 1、TPSC0 = 0、または、8TCR0 の CKS2 ~ CKS0 ビットが下表 (3) のとき TCLKC 入力端子となります。</p>						
	16 ビットタイマ チャンネル0の設定	(2)	(1)		(2)	
	PWM0	0			1	
	IOA2	0		1	-	
	IOA1	0	0	1	-	
	IOA0	0	1	-	-	
	8 ビットタイマ チャンネル0の設定	(4)			(3)	
	CKS2	0	1			
	CKS1	-	0		1	
	CKS0	-	0	1	-	

端子	選択方法と端子機能			
PA <sub>1</sub> /TP <sub>1</sub> / TCLKB/ $\overline{TEND}_1$	TMDR の MDF ビット、16 ビットタイマの 16TCR2 ~ 16TCR0 の TPSC2 ~ TPSC0 ビット、8 ビットタイマの 8TCR3 の CKS2 ~ CKS0 ビット、NDERA の NDER1 ビットと PA <sub>1</sub> DDR ビットの組み合わせにより、次のように切り換わります。			
	PA <sub>1</sub> DDR	0	1	1
	NDER1	-	0	1
端子機能	PA <sub>1</sub> 入力端子	PA <sub>1</sub> 出力端子		TP <sub>1</sub> 出力端子
	TCLKB 出力端子* <sup>1</sup>			
	$\overline{TEND}_1$ 出力端子* <sup>2</sup>			
<p>【注】*<sup>1</sup> TMDR の MDF = 1、または 16TCR2 ~ 16TCR0 のいずれかの設定が TPSC2 = 1、TPSC1 = 0、TPSC0 = 1、または、8TCR3 の CKS2 ~ CKS0 ビットが下表 (1) のとき TCLKB 入力端子となります。</p> <p>*<sup>2</sup> DMAC の起動要因として外部リクエストを指定した場合は、PA<sub>1</sub>DDR ビット、NDER1 ビットとは無関係に <math>\overline{TEND}_1</math> 出力端子となります。</p>				
8 ビットタイマ チャンネル 3 の設定	(2)	(1)		
CKS2	0	1		
CKS1	-	0	1	
CKS0	-	0	1	-

端子	選択方法と端子機能			
PA <sub>0</sub> /TP <sub>0</sub> / TCLKA/ $\overline{TEND}_0$	TMDR の MDF ビット、16 ビットタイマの 16TCR2 ~ 16TCR0 の TPSC2 ~ TPSC0 ビット、8 ビットタイマの 8TCR1 の CKS2 ~ CKS0 ビット、NDERA の NDER0 ビットと PA <sub>0</sub> DDR ビットの組合わせにより、次のように切り換わります。			
	PA <sub>0</sub> DDR	0	1	
	NDER0	-	0	1
端子機能	PA <sub>0</sub> 入力端子	PA <sub>0</sub> 出力端子		TP <sub>0</sub> 出力端子
	TCLKA 出力端子* <sup>1</sup>			
	$\overline{TEND}_0$ 出力端子* <sup>2</sup>			
	<p>【注】 *1 TMDR の MDF = 1、または 16TCR2 ~ 16TCR0 のいずれかの設定が TPSC2 = 1、TPSC1 = 0、TPSC0 = 0 または、8TCR1 の CKS2 ~ CKS0 ビットが下表 (1) のとき TCLKA 入力端子となります。</p> <p>*2 DMAC の起動要因として外部リクエストを指定した場合は、PA<sub>0</sub>DDR ビット、NDER0 ビットとは無関係に <math>\overline{TEND}_0</math> 出力端子となります。</p>			
	8 ビットタイマ チャンネル 1 の設定	(2)		(1)
	CKS2	0	1	
	CKS1	-	0	1
	CKS0	-	0	1
			1	-

## 8.12 ポート B

### 8.12.1 概要

ポート B は、8 ビットの入出力ポートです。ポート B はプログラマブルタイミングパターンコントローラ (TPC) の出力端子 (TP<sub>15</sub> ~ TP<sub>8</sub>)、8 ビットタイマの入出力端子 (TMIO<sub>3</sub>、TMO<sub>2</sub>、TMIO<sub>1</sub>、TMO<sub>0</sub>)、 $\overline{CS}_7$  ~  $\overline{CS}_4$  出力端子、DMA コントローラ (DMAC) の入力端子 ( $\overline{DREQ}_1$ 、 $\overline{DREQ}_0$ )、シリアルコミュニケーションインタフェースチャンネル 2 (SCI2) の入出力端子 (TxD<sub>2</sub>、RxD<sub>2</sub>、SCK<sub>2</sub>)、DRAM インタフェースの出力端子 ( $\overline{UCAS}$ 、 $\overline{LCAS}$ ) と兼用になっています。端子機能の選択方法については表 8.23 ~ 表 8.24 を参照してください。

ポート B はリセットおよびハードウェアスタンバイモードで入力ポートになっています。モード 1 ~ 5 で  $\overline{CS}_7$  ~  $\overline{CS}_4$  を出力する場合は、「6.3.4 チップセレクト信号」を参照してください。これらのいずれの機能も割当てられない端子は入出力ポートとして使用できます。ポート B の端子構成を図 8.11 に示します。

エリア 2、3、4、5 に DRAM を接続する場合には、 $\overline{CS}_4$ 、 $\overline{CS}_5$  出力が各エリアに対応した  $\overline{RAS}$  の出力端子となります。詳細は「6.5 DRAM インタフェース」を参照してください。

ポート B は、1 個の TTL 負荷と 30pF の容量負荷を駆動することができます。また、ダーリントントランジスタを駆動することができます。



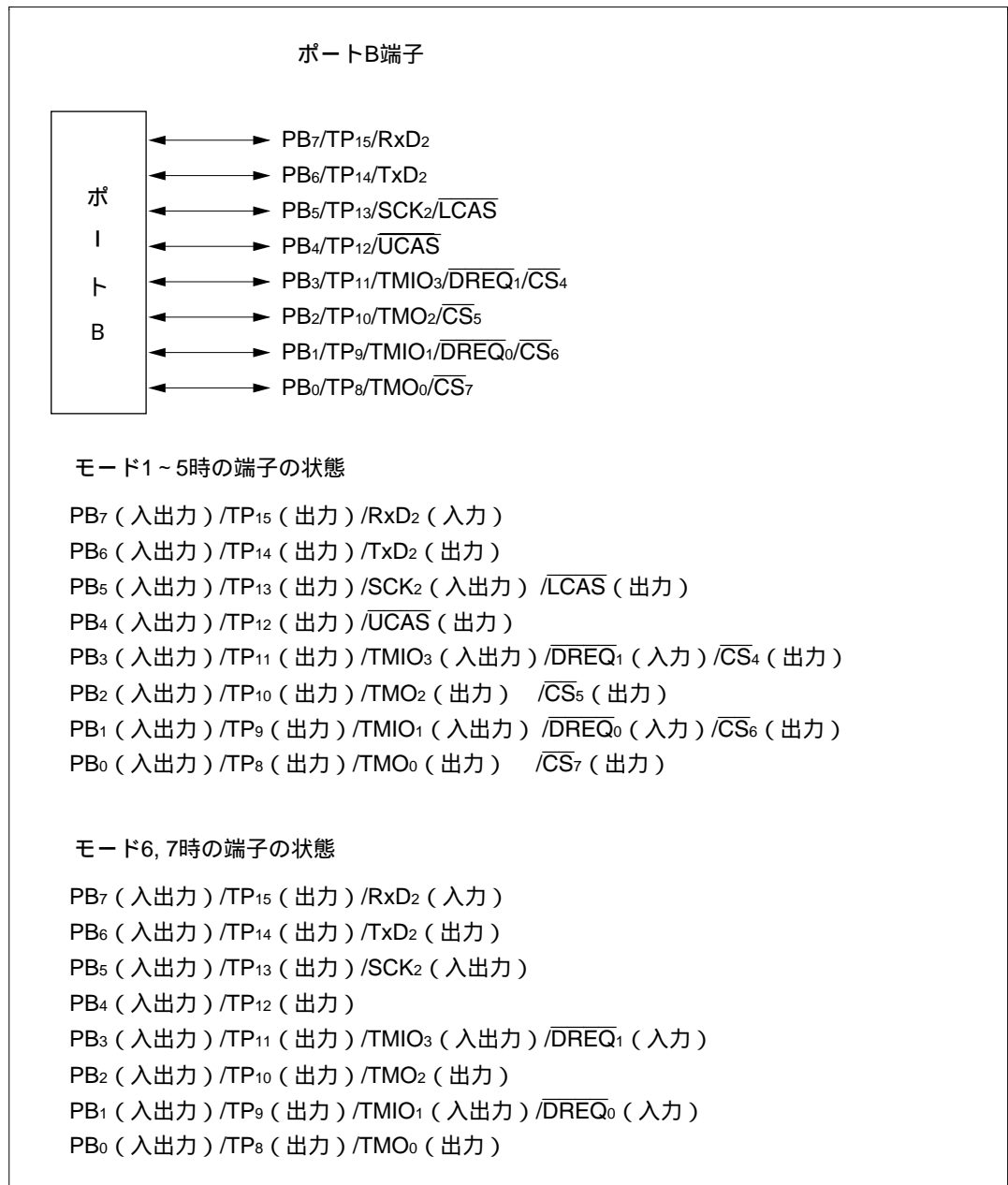


図 8.11 ポートBの端子構成

## 8.12.2 レジスタ構成

表 8.22 にポート B のレジスタ構成を示します。

表 8.22 ポート B レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H'EE00A	ポート B データディレクションレジスタ	PBDDR	W	H'00
H'FFFDA	ポート B データレジスタ	PBDR	R/W	H'00

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

### (1) ポート B データディレクションレジスタ (PBDDR)

PBDDR は、8 ビットのライト専用のレジスタで、ポート B 各端子の入出力をビットごとに指定することができます。TPC の出力端子として使用する場合も PBDDR の対応するビットをセットしてください。

ビット:	7	6	5	4	3	2	1	0
	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR	PB0DDR
初期値:	0	0	0	0	0	0	0	0
R/W :	W	W	W	W	W	W	W	W

ポート B データディレクション 7~0  
ポート B の各端子の入出力を選択するビットです。

ポート B はモード 1~5 の場合と、モード 6、7 の場合で選択できる端子機能が異なります。端子機能の選択方法については表 8.23 および表 8.24 を参照してください。

ポート B が入出力ポートとして機能している場合、PBDDR を 1 にセットすると対応するポート B の各端子は出力ポートとなり、0 にクリアすると入力ポートになります。

PBDDR は、ライト専用レジスタで、リードは無効です。リードすると、1 が読み出されます。

PBDDR は、リセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。ソフトウェアスタンバイモード時には直前の状態を保持します。そのため、ポート B が入出力ポートとして機能しているとき、PBDDR が 1 にセットされた状態でソフトウェアスタンバイモードに遷移すると、その端子は出力状態のままとなっています。

## (2) ポート B データレジスタ (PBDR)

PBDR は、8 ビットのリード/ライト可能なレジスタで、ポート B の出力データを格納します。ポート B が出力ポートとして機能する場合、本レジスタの値が出力されます。また、このレジスタをリードすると、PBDDR が 0 のビットは端子のロジックレベルが読み出され、1 のビットは PBDR の値が読み出されます。

ビット :	7	6	5	4	3	2	1	0
	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ポートBデータ7~0  
ポートBの各端子のデータを格納するビットです。

PBDR は、リセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。ソフトウェアスタンバイモード時には、直前の状態を保持します。

表 8.23 ポート B (モード 1~5) の端子機能

端子	選択方法と端子機能						
PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	SCI2 の SCR の RE ビット、SCMR の SMIF ビット、NDERB の NDER15 ビットと PB <sub>7</sub> DDR ビットの組合わせにより、次のように切り換わります。						
	SMIF	0				1	
	RE	0		1	1	-	
	PB <sub>7</sub> DDR	0	1	1	-	-	
	NDER15	-	0	1	-	-	
	端子機能	PB <sub>7</sub> 入力 端子	PB <sub>7</sub> 出力 端子	TP <sub>15</sub> 出力 端子	RxD <sub>2</sub> 入力 端子	RxD <sub>2</sub> 入力 端子	
PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	SCI2 の SCR の TE ビット、SCMR の SMIF ビット、NDERB の NDER14 ビットと PB <sub>6</sub> DDR ビットの組合わせにより、次のように切り換わります。						
	SMIF	0				1	
	TE	0		1	1	-	
	PB <sub>6</sub> DDR	0	1	1	-	-	
	NDER14	-	0	1	-	-	
	端子機能	PB <sub>6</sub> 入力 端子	PB <sub>6</sub> 出力 端子	TP <sub>14</sub> 出力 端子	TxD <sub>2</sub> 出力 端子	TxD <sub>2</sub> 出力 端子*	
【注】* TxD <sub>2</sub> 出力端子として機能します。ただし、ハイインピーダンス状態と端子ドライブ状態の 2 種類の状態があります。							
PB <sub>5</sub> / TP <sub>13</sub> / SCK <sub>2</sub> /LCAS	SCI2 の SMR の C/A ビット、SCR の CKE0, CKE1 ビット、NDERB の NDER13 ビットと PB <sub>5</sub> DDR ビットの組合わせにより、次のように切り換わります。						
	CKE1	0				1	
	C/A	0		1	1	-	
	CKE0	0		1	-	-	
	PB <sub>5</sub> DDR	0	1	1	-	-	
	NDER13	-	0	1	-	-	
端子機能	PB <sub>5</sub> 入力 端子	PB <sub>5</sub> 出力 端子	TP <sub>13</sub> 出 力 端子	SCK <sub>2</sub> 出 力 端子	SCK <sub>2</sub> 出 力 端子	SCK <sub>2</sub> 入 力 端子	
LCAS 出力端子*							
【注】* DRCRA の DRAS2~0 ビットと DRCRB の CSEL ビットに依存し、C/A ビット、CKE0, 1 ビット、NDER13 ビット、PB <sub>5</sub> DDR ビットとは無関係に LCAS 出力となります。詳細は「第 6 章 パスコントローラ」を参照してください。							

端子	選択方法と端子機能								
PB <sub>4</sub> /TP <sub>12</sub> / UCAS	NDERBの NDER12 ビットと PB <sub>4</sub> DDR ビットの組合わせにより、次のように切り換わります。								
	PB <sub>4</sub> DDR	0		1		1			
	NDER12	-		0		1			
	端子機能	PB <sub>4</sub> 入力端子		PB <sub>4</sub> 出力端子		TP <sub>12</sub> 出力端子		UCAS 出力端子*	
	【注】* DRCRA の DRAS2~0 ビットと DRCRB の CSEL ビットに依存し、NDER12 ビット、PB <sub>4</sub> DDR ビットとは無関係に UCAS 出力となります。詳細は「第6章 パスコントローラ」を参照してください。								
PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> / DREQ <sub>1</sub> / $\overline{CS}_4$	DRCRA の DRAS2、DRAS1、DRAS0 ビットによる DRAM インタフェースの設定、8TCSR3 の OIS3、OIS2、OS1、OS0 ビット、8TCR3 の CCLR1、CCLR0 ビット、CSCR の CS4E ビット、NDERB の NDER11 ビットと PB <sub>3</sub> DDR ビットの組合わせにより、次のように切り換わります。								
	DRAM インタフェースの設定	下表(1)					下表(2)		
	OIS3, OIS2, OS1, OS0	すべてが0				いずれかが1	-		
	CS4E	0		1		-	-		
	PB <sub>3</sub> DDR	0	1	1	-	-	-		
	NDER11	-	0	1	-	-	-		
	端子機能	PB <sub>3</sub> 入力端子	PB <sub>3</sub> 出力端子	TP <sub>11</sub> 出力端子	$\overline{CS}_4$ 出力端子	TMIO <sub>3</sub> 出力端子	$\overline{CS}_4$ 出力端子* <sup>3</sup>		
		TMIO <sub>3</sub> 入力端子* <sup>1</sup>							
		$\overline{DREQ}_1$ 入力端子* <sup>2</sup>							
		【注】*1 CCLR1、CCLR0=1 のとき、TMIO <sub>3</sub> 入力端子となります。 *2 DMAC の起動要因を外部リクエストに指定した場合は、OIS3、OIS2 ビット、OS1、OS0 ビット、CCLR1、CCLR0 ビット、CS4E ビット、NDER11 ビット、PB <sub>3</sub> DDR ビットとは無関係に $\overline{DREQ}_1$ 入力端子となります。 *3 $\overline{RAS}_4$ として $\overline{CS}_4$ 出力します。							
DRAM インタフェースの設定	(1)			(2)			(1)		
DRAS2	0			1					
DRAS1	0	1	0	1	0	1			
DRAS0	0	1	0	1	0	1	0	1	

端子	選択方法と端子機能						
PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub> / $\overline{CS}_5$	DRCRA の DRAS2、DRAS1、DRAS0 ビットによる DRAM インタフェースの設定、8TCSR2 の OIS3、OSI2、OS1、OS0 ビット、CSCR の CS5E ビット、NDERB の NDER10 ビットと PB <sub>2</sub> DDR ビットの組合わせにより、次のように切り換わります。						
DRAM インタフェースの設定	下表 (1)					下表 (2)	
OIS3, OIS2, OS1, OS0	すべてが 0				いずれかが 1	-	
CS5E	0		1		-	-	
PB <sub>2</sub> DDR	0	1	1	-	-	-	
NDER10	-	0	1	-	-	-	
端子機能	PB <sub>2</sub> 入力端子	PB <sub>2</sub> 出力端子	TP <sub>10</sub> 出力端子	$\overline{CS}_5$ 出力端子	TMO <sub>2</sub> 出力端子	$\overline{CS}_5$ 出力端子*	
【注】* $\overline{RAS}_5$ として $\overline{CS}_5$ 出力します。							
DRAM インタフェースの設定	(1)				(2)	(1)	
DRAS2	0			1			
DRAS1	0		1		0		1
DRAS0	0	1	0	1	0	1	0 1

端子	選択方法と端子機能						
PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / DREQ <sub>0</sub> / $\overline{CS}_6$	8TCSR1 の OIS3、OIS2、OS1、OS0 ビット、8TCR0 の CCLR1、CCLR0 ビット、CSCR の CS6E ビット、NDERB の NDER9 ビットと PB <sub>1</sub> DDR ビットの組合わせにより、次のように切り換わります。						
	OIS3, OIS2, OS1, OS0	すべてが 0				いずれかが 1	
	CS6E	0			1	-	
	PB <sub>1</sub> DDR	0	1	1	-	-	
	NDER9	-	0	1	-	-	
	端子機能	PB <sub>1</sub> 入力端子	PB <sub>1</sub> 出力端子	TP <sub>9</sub> 出力端子	$\overline{CS}_6$ 出力端子	TMIO <sub>1</sub> 出力端子	
		TMIO <sub>1</sub> 入力端子* <sup>1</sup>					
DREQ <sub>0</sub> 入力端子* <sup>2</sup>							
<p>【注】 *1 CCLR1~0=1 のとき、TMIO<sub>1</sub> 入力端子となります。</p> <p>*2 DMAC の起動要因を外部リクエストに指定した場合は、OIS3、OIS2、OS1、OS0 ビット、CCLR1、CCLR0 ビット、CS6E ビット、NDER9 ビット、PB<sub>1</sub>DDR ビットとは無関係に <math>\overline{DREQ}_0</math> 入力端子となります。</p>							
PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub> / $\overline{CS}_7$	8TCSR0 の OIS3、OIS2、OS1、OS0 ビット、CSCR の CS7E ビット、NDERB の NDER8 ビットと PB <sub>0</sub> DDR ビットの組合わせにより、次のように切り換わります。						
	OIS3, OIS2, OS1, OS0	すべてが 0				いずれかが 1	
	CS7E	0			1	-	
	PB <sub>0</sub> DDR	0	1	1	-	-	
	NDER8	-	0	1	-	-	
	端子機能	PB <sub>0</sub> 入力端子	PB <sub>0</sub> 出力端子	TP <sub>8</sub> 出力端子	$\overline{CS}_7$ 出力端子	TMO <sub>0</sub> 出力端子	

表 8.24 ポート B (モード 6、7) の端子機能

端子	選択方法と端子機能					
PB <sub>7</sub> /TP <sub>15</sub> / RxD <sub>2</sub>	SCI2 の SCR の RE ビット、SCMR の SMIF ビット、NDERB の NDER15 ビットと PB <sub>7</sub> DDR ビットの組合わせにより、次のように切り換わります。					
	SMIF		0			1
	RE		0	1	-	-
	PB <sub>7</sub> DDR	0	1	1	-	-
	NDER15		-	0	1	-
端子機能	PB <sub>7</sub> 入力端子	PB <sub>7</sub> 出力端子	TP <sub>15</sub> 出力端子	RxD <sub>2</sub> 入力端子	RxD <sub>2</sub> 入力端子	
PB <sub>6</sub> /TP <sub>14</sub> / TxD <sub>2</sub>	SCI2 の SCR の TE ビット、SCMR の SMIF ビット、NDERB の NDER14 ビットと PB <sub>6</sub> DDR ビットの組合わせにより、次のように切り換わります。					
	SMIF		0			1
	TE		0	1	-	-
	PB <sub>6</sub> DDR	0	1	1	-	-
	NDER14		-	0	1	-
端子機能	PB <sub>6</sub> 入力端子	PB <sub>6</sub> 出力端子	TP <sub>14</sub> 出力端子	TxD <sub>2</sub> 出力端子	TxD <sub>2</sub> 出力端子*	
【注】* TxD <sub>2</sub> 出力端子として機能します。ただし、ハイインピーダンス状態と端子ドライブ状態の 2 種類の状態があります。						
PB <sub>5</sub> /TP <sub>13</sub> / SCK <sub>2</sub>	SCI2 の SMR の C/ $\bar{A}$ ビット、SCR の CKE0, CKE1 ビット、NDERB の NDER13 ビットと PB <sub>5</sub> DDR ビットの組合わせにより、次のように切り換わります。					
	CKE1		0			1
	C/ $\bar{A}$		0			1
	CKE0		0	1	-	-
	PB <sub>5</sub> DDR	0	1	1	-	-
	NDER13		-	0	1	-
端子機能	PB <sub>5</sub> 入力端子	PB <sub>5</sub> 出力端子	TP <sub>13</sub> 出力端子	SCK <sub>2</sub> 出力端子	SCK <sub>2</sub> 出力端子	SCK <sub>2</sub> 入力端子
PB <sub>4</sub> /TP <sub>12</sub>	NDERB の NDER12 ビットと PB <sub>4</sub> DDR ビットの組合わせにより、次のように切り換わります。					
	PB <sub>4</sub> DDR		0	1	1	
	NDER12		-	0	1	
端子機能	PB <sub>4</sub> 入力端子		PB <sub>4</sub> 出力端子		TP <sub>12</sub> 出力端子	



端子	選択方法と端子機能				
PB <sub>3</sub> /TP <sub>11</sub> / TMIO <sub>3</sub> /  DREQ <sub>1</sub>	TCSR3の OIS3、OSI2、OS1、OS0 ビット、TCR3の CCLR1、CCLR0 ビット、NDERBの NDER11 ビットと PB <sub>3</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	OIS3, OSI2, OS1, OS0	すべてが 0			いずれかが 1
	PB <sub>3</sub> DDR	0	1	1	-
	NDER11	-	0	1	-
	端子機能	PB <sub>3</sub> 入力端子	PB <sub>3</sub> 出力端子	TP <sub>11</sub> 出力端子	TMIO <sub>3</sub> 出力端子
	TMIO <sub>3</sub> 入力端子* <sup>1</sup>				
	DREQ <sub>1</sub> 入力端子* <sup>2</sup>				
	<p>【注】*1 CCLR1、CCLR0=1 のとき、TMIO<sub>3</sub>入力端子となります。</p> <p>*2 DMAC の起動要因を外部リクエストに指定した場合は、OIS3、OSI2、OS1、OS0 ビット、NDER11 ビット、PB<sub>3</sub>DDR ビットとは無関係に DREQ<sub>1</sub> 入力端子となります。</p>				
PB <sub>2</sub> /TP <sub>10</sub> / TMO <sub>2</sub>	TCSR2の OIS3、OSI2、OS1、OS0 ビット、NDERBの NDER10 ビットと PB <sub>2</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	OIS3, OSI2, OS1, OS0	すべてが 0			いずれかが 1
	PB <sub>2</sub> DDR	0	1	1	-
	NDER10	-	0	1	-
	端子機能	PB <sub>2</sub> 入力端子	PB <sub>2</sub> 出力端子	TP <sub>10</sub> 出力端子	TMO <sub>2</sub> 出力端子

端子	選択方法と端子機能				
PB <sub>1</sub> /TP <sub>9</sub> / TMIO <sub>1</sub> / $\overline{\text{DREQ}}_0$	TCSR1の OIS3、OIS2、OS1、OS0 ビット、TCR1の CCLR1、CCLR0 ビット、NDERBの NDER9 ビットと PB <sub>1</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	OIS3, OIS2, OS1, OS0	すべてが 0			いずれかが 1
	PB <sub>1</sub> DDR	0	1	1	-
	NDER9	-	0	1	-
	端子機能	PB <sub>1</sub> 入力端子	PB <sub>1</sub> 出力端子	TP <sub>9</sub> 出力端子	TMIO <sub>1</sub> 出力端子
		TMIO <sub>1</sub> 入力端子* <sup>1</sup>			
DREQ <sub>0</sub> 入力端子* <sup>2</sup>					
<p>【注】*1 CCLR1、CCLR0=1 のとき、TMIO<sub>1</sub>入力端子となります。</p> <p>*2 DMACの起動要因に外部リクエストを指定した場合は、OIS3、OIS2、OS1、OS0、NDER9 ビット、PB<sub>1</sub>DDR ビットとは無関係に <math>\overline{\text{DREQ}}_0</math> 入力端子となります。</p>					
PB <sub>0</sub> /TP <sub>8</sub> / TMO <sub>0</sub>	TCSR0の OIS3、OIS2、OS1、OS0 ビット、NDERBの NDER8 ビットと PB <sub>0</sub> DDR ビットの組合わせにより、次のように切り換わります。				
	OIS3, OIS2, OS1, OS0	すべてが 0			いずれかが 1
	PB <sub>0</sub> DDR	0	1	1	-
	NDER8	-	0	1	-
	端子機能	PB <sub>0</sub> 入力端子	PB <sub>0</sub> 出力端子	TP <sub>8</sub> 出力端子	TMO <sub>0</sub> 出力端子



---

# 9. 16ビットタイマ

---

## 第9章 目次

9.1 概要	355
9.1.1 特長	355
9.1.2 ブロック図	357
9.1.3 端子構成	360
9.1.4 レジスタ構成	361
9.2 各レジスタの説明	362
9.2.1 タイマスタートレジスタ (TSTR)	362
9.2.2 タイマシンクロレジスタ (TSNC)	363
9.2.3 タイマモードレジスタ (TMDR)	365
9.2.4 タイマインタラプトステータスレジスタ A (TISRA)	368
9.2.5 タイマインタラプトステータスレジスタ B (TISRB)	371
9.2.6 タイマインタラプトステータスレジスタ C (TISRC)	374
9.2.7 タイマカウンタ (16TCNT)	377
9.2.8 ジェネラルレジスタ A、B (GRA、GRB)	378
9.2.9 タイマコントロールレジスタ (16TCR)	379
9.2.10 タイマ I/O コントロールレジスタ (TIOR)	382
9.2.11 タイマアウトプットレベルセットレジスタ (TOLR)	385
9.3 CPU とのインタフェース	387
9.3.1 16ビットアクセス可能なレジスタ	387
9.3.2 8ビットアクセスのレジスタ	389
9.4 動作説明	390
9.4.1 概要	390
9.4.2 基本機能	391
9.4.3 同期動作	401
9.4.4 PWM モード	403
9.4.5 位相計数モード	407
9.4.6 16ビットタイマ出力初期値の設定	409

9.5	割込み.....	410
9.5.1	ステータスフラグのセットタイミング .....	410
9.5.2	ステータスフラグのクリアタイミング .....	412
9.5.3	割込み要因とDMA コントローラの起動.....	413
9.6	使用上の注意.....	414

## 9.1 概要

本 LSI は、3 チャンネルの 16 ビットカウンタにより構成される 16 ビットタイマを内蔵しています。

### 9.1.1 特長

16 ビットタイマの特長を以下に示します。

最大 6 種類のパルス出力、または最大 6 種類のパルス入力処理が可能

各チャンネル 2 本、合計 6 本のジェネラルレジスタ (GR) を持ち、各レジスタ独立にアウトプットコンペアマッチ/インプットキャプチャの機能設定が可能

各チャンネルとも 8 種類のカウンタ入力クロックを選択可能

内部クロック： 、 /2、 /4、 /8

外部クロック：TCLKA、TCLKB、TCLKC、TCLKD

各チャンネルとも次の動作モードを設定可能

- ・コンペアマッチによる波形出力：0 出力 / 1 出力 / トグル出力が選択可能 (チャンネル 2 は 0 出力 / 1 出力が可能)
- ・インプットキャプチャ機能：立上がりエッジ / 立下がりエッジ / 両エッジ検出が選択可能
- ・カウンタクリア機能：コンペアマッチ / インプットキャプチャによるカウンタクリアが可能
- ・同期動作：複数のタイマカウンタ (16TCNT) への同時書き込みが可能  
コンペアマッチ / インプットキャプチャによる同時クリアが可能  
カウンタの同期動作による各レジスタの同期入出力が可能
- ・PWM モード：任意デューティの PWM 出力が可能  
同期動作と組み合わせることにより、最大 3 相の PWM 出力が可能

チャンネル 2 は位相計数モードを設定可能

2 相エンコーダのカウント数の自動計測が可能

内部 16 ビットバスによる高速アクセス

16TCNT、GR の 16 ビットレジスタに対して、16 ビットバスによる高速アクセスが可能

タイマ出力初期値を任意に設定可能

9 種類の割込み要因

各チャンネルともコンペアマッチ / インプットキャプチャ兼用割込み × 2 要因、オーバーフロー割込み × 1 要因があり、それぞれ独立に要求可能

プログラマブルパターンコントローラ (TPC) の出力トリガが生成可能

チャンネル 0~2 のコンペアマッチ / インプットキャプチャ信号を TPC の出力トリガとして使用可能

16ビットタイマの機能一覧を表9.1に示します。

表9.1 16ビットタイマの機能一覧

項目	チャンネル0	チャンネル1	チャンネル2
カウントクロック	内部クロック： 、 /2、 /4、 /8 外部クロック：TCLKA、TCLKB、TCLKC、TCLKDから独立に選択可能		
ジェネラルレジスタ (アウトプットコンペア/インプットキャプチャ兼用レジスタ)	GRA0、GRB0	GRA1、GRB1	GRA2、GRB2
入出力端子	TIOCA <sub>0</sub> 、TIOCB <sub>0</sub>	TIOCA <sub>1</sub> 、TIOCB <sub>1</sub>	TIOCA <sub>2</sub> 、TIOCB <sub>2</sub>
カウンタクリア機能	GRA0/GRB0のコンペアマッチまたはインプットキャプチャ	GRA1/GRB1のコンペアマッチまたはインプットキャプチャ	GRA2/GRB2のコンペアマッチまたはインプットキャプチャ
出力初期値設定機能			
コンペア	0出力		
マッチ	1出力		
出力	トグル出力		
インプットキャプチャ機能			
同期動作			
PWMモード			
位相計数モード			
割込み要因	3要因 ・コンペアマッチ/インプットキャプチャA0 ・コンペアマッチ/インプットキャプチャB0 ・オーバーフロー	3要因 ・コンペアマッチ/インプットキャプチャA1 ・コンペアマッチ/インプットキャプチャB1 ・オーバーフロー	3要因 ・コンペアマッチ/インプットキャプチャA2 ・コンペアマッチ/インプットキャプチャB2 ・オーバーフロー

《記号説明》

：可能

-：不可

## 9.1.2 ブロック図

## (1) 16ビットタイマのブロック図(全体図)

16ビットタイマのブロック図(全体図)を図9.1に示します。

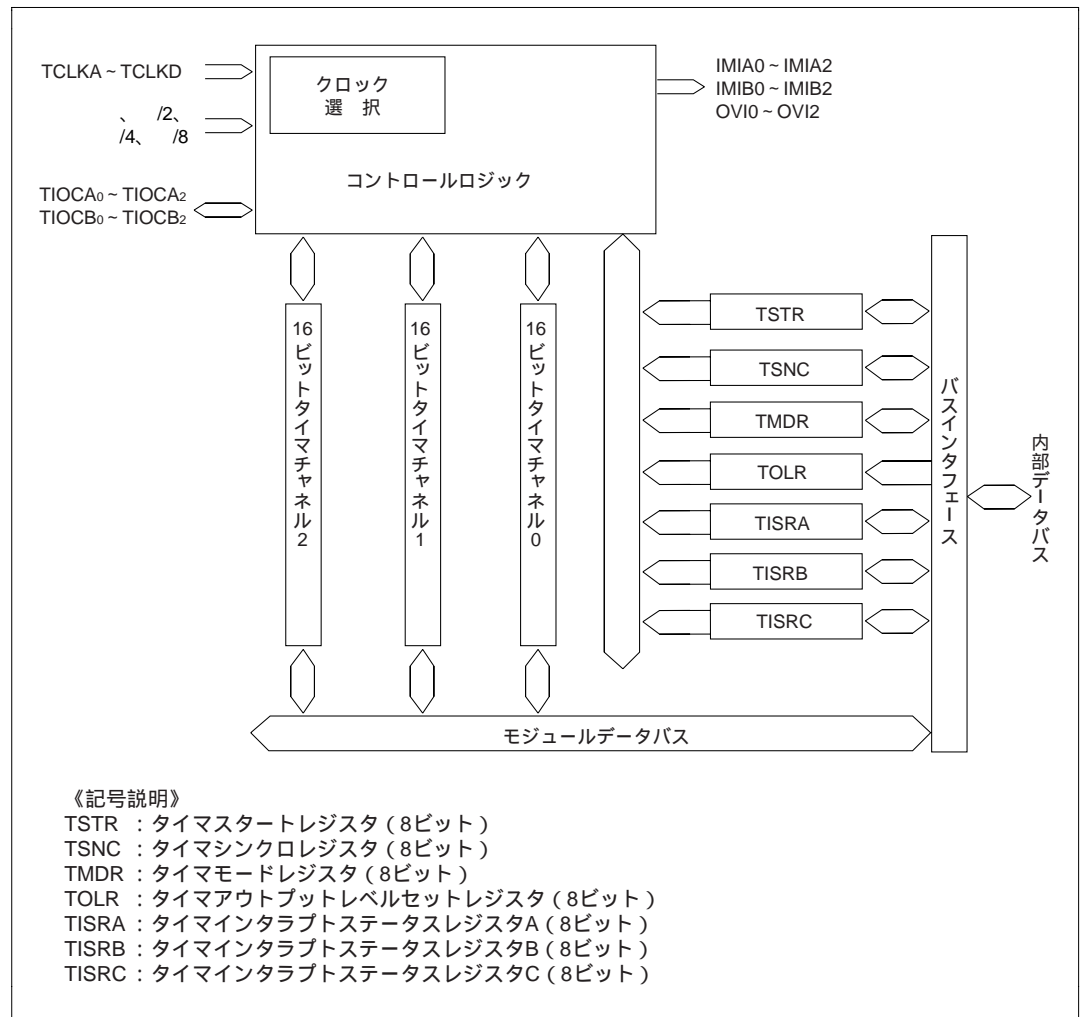


図9.1 16ビットタイマのブロック図(全体図)



(2) チャンネル0、1のブロック図

16ビットタイマのチャンネル0、1は同一の機能を持っています。チャンネル0、1のブロック図を図9.2に示します。

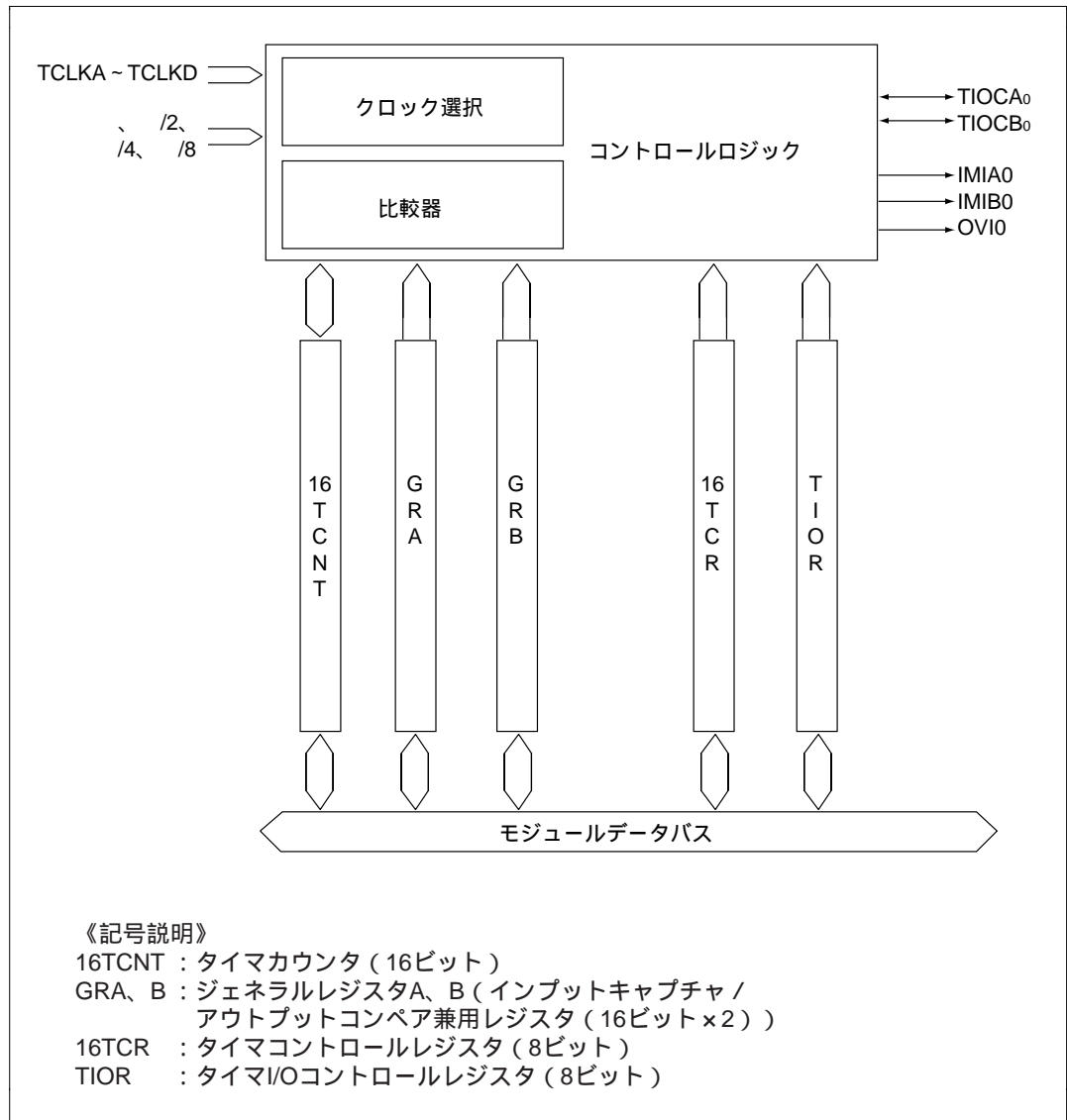


図9.2 チャンネル0、1のブロック図

## (3) チャンネル2のブロック図

チャンネル2のブロック図を図9.3に示します。

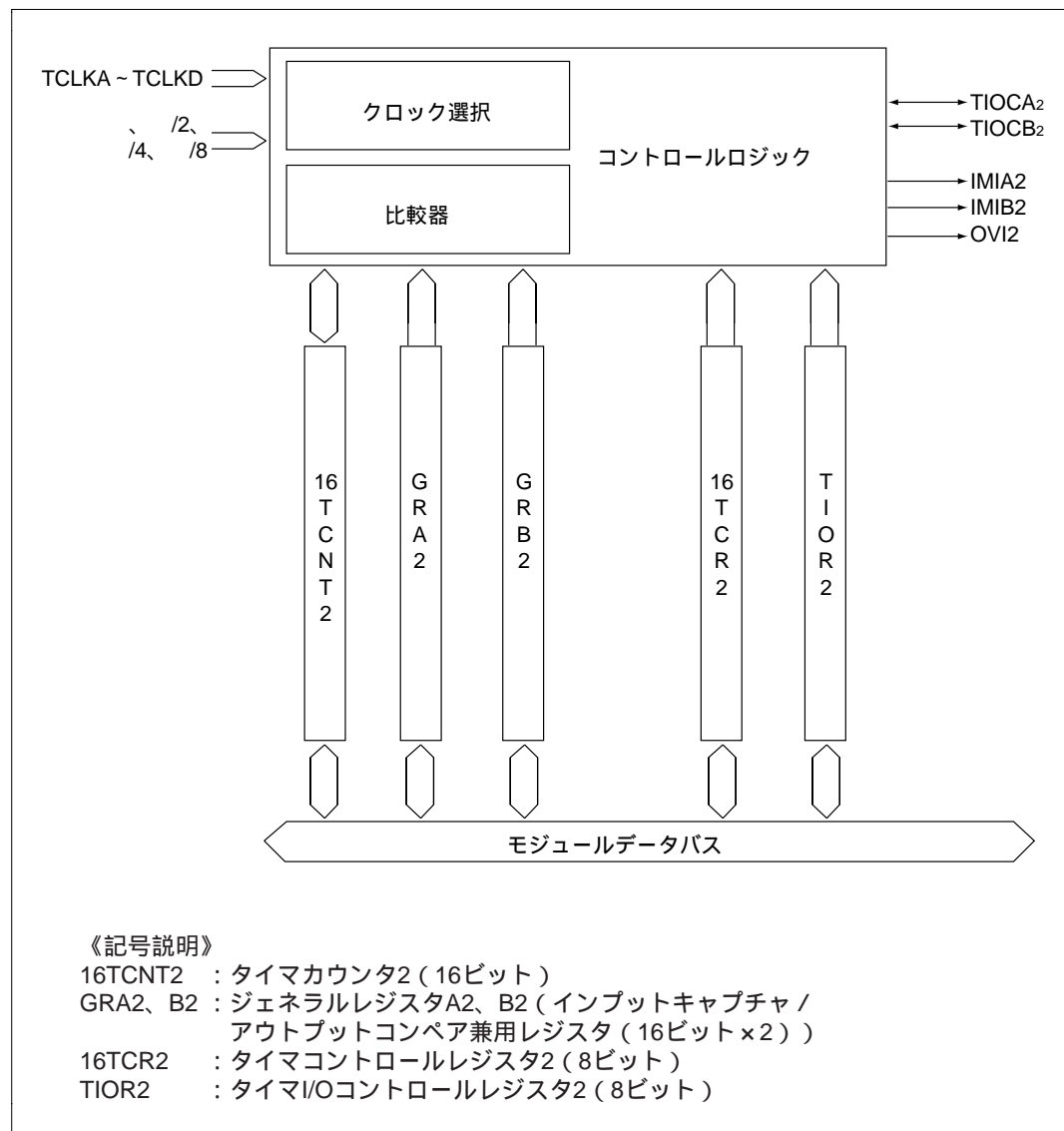


図9.3 チャンネル2のブロック図

## 9.1.3 端子構成

16ビットタイマの端子構成を表9.2に示します。

表9.2 端子構成

チャンネル	名称	略称	入出力	機能
共通	クロック入力 A	TCLKA	入力	外部クロック A 入力端子 (位相計数モード時 A 相入力端子)
	クロック入力 B	TCLKB	入力	外部クロック B 入力端子 (位相計数モード時 B 相入力端子)
	クロック入力 C	TCLKC	入力	外部クロック C 入力端子
	クロック入力 D	TCLKD	入力	外部クロック D 入力端子
0	インプットキャプチャ / アウトプットコンペア A0	TIOCA <sub>0</sub>	入出力	GRA0 アウトプットコンペア出力 / GRA0 インプットキャプチャ入力 / PWM 出力端子 (PWM モード時)
	インプットキャプチャ / アウトプットコンペア B0	TIOCB <sub>0</sub>	入出力	GRB0 アウトプットコンペア出力 / GRB0 インプットキャプチャ入力端子
1	インプットキャプチャ / アウトプットコンペア A1	TIOCA <sub>1</sub>	入出力	GRA1 アウトプットコンペア出力 / GRA1 インプットキャプチャ入力 / PWM 出力端子 (PWM モード時)
	インプットキャプチャ / アウトプットコンペア B1	TIOCB <sub>1</sub>	入出力	GRB1 アウトプットコンペア出力 / GRB1 インプットキャプチャ入力端子
2	インプットキャプチャ / アウトプットコンペア A2	TIOCA <sub>2</sub>	入出力	GRA2 アウトプットコンペア出力 / GRA2 インプットキャプチャ入力 / PWM 出力端子 (PWM モード時)
	インプットキャプチャ / アウトプットコンペア B2	TIOCB <sub>2</sub>	入出力	GRB2 アウトプットコンペア出力 / GRB2 インプットキャプチャ入力端子

## 9.1.4 レジスタ構成

16ビットタイマのレジスタ構成を表9.3に示します。

表9.3 レジスタ構成

チャンネル	アドレス* <sup>1</sup>	名称	略称	R/W	初期値
共通	H'FFF60	タイマスタートレジスタ	TSTR	R/W	H'F8
	H'FFF61	タイマシンクロレジスタ	TSNC	R/W	H'F8
	H'FFF62	タイマモードレジスタ	TMDR	R/W	H'98
	H'FFF63	タイマアウトプットレベルセットレジスタ	TOLR	W	H'C0
	H'FFF64	タイマインタラプトステータスレジスタ A	TISRA	R/(W)* <sup>2</sup>	H'88
	H'FFF65	タイマインタラプトステータスレジスタ B	TISRB	R/(W)* <sup>2</sup>	H'88
	H'FFF66	タイマインタラプトステータスレジスタ C	TISRC	R/(W)* <sup>2</sup>	H'88
0	H'FFF68	タイマコントロールレジスタ 0	16TCR0	R/W	H'80
	H'FFF69	タイマ I/O コントロールレジスタ 0	TIOR0	R/W	H'88
	H'FFF6A	タイマカウンタ 0H	16TCNT0H	R/W	H'00
	H'FFF6B	タイマカウンタ 0L	16TCNT0L	R/W	H'00
	H'FFF6C	ジェネラルレジスタ A0H	GRA0H	R/W	H'FF
	H'FFF6D	ジェネラルレジスタ A0L	GRA0L	R/W	H'FF
	H'FFF6E	ジェネラルレジスタ B0H	GRB0H	R/W	H'FF
	H'FFF6F	ジェネラルレジスタ B0L	GRB0L	R/W	H'FF
1	H'FFF70	タイマコントロールレジスタ 1	16TCR1	R/W	H'80
	H'FFF71	タイマ I/O コントロールレジスタ 1	TIOR1	R/W	H'88
	H'FFF72	タイマカウンタ 1H	16TCNT1H	R/W	H'00
	H'FFF73	タイマカウンタ 1L	16TCNT1L	R/W	H'00
	H'FFF74	ジェネラルレジスタ A1H	GRA1H	R/W	H'FF
	H'FFF75	ジェネラルレジスタ A1L	GRA1L	R/W	H'FF
	H'FFF76	ジェネラルレジスタ B1H	GRB1H	R/W	H'FF
	H'FFF77	ジェネラルレジスタ B1L	GRB1L	R/W	H'FF
2	H'FFF78	タイマコントロールレジスタ 2	16TCR2	R/W	H'80
	H'FFF79	タイマ I/O コントロールレジスタ 2	TIOR2	R/W	H'88
	H'FFF7A	タイマカウンタ 2H	16TCNT2H	R/W	H'00
	H'FFF7B	タイマカウンタ 2L	16TCNT2L	R/W	H'00
	H'FFF7C	ジェネラルレジスタ A2H	GRA2H	R/W	H'FF
	H'FFF7D	ジェネラルレジスタ A2L	GRA2L	R/W	H'FF
	H'FFF7E	ジェネラルレジスタ B2H	GRB2H	R/W	H'FF
	H'FFF7F	ジェネラルレジスタ B2L	GRB2L	R/W	H'FF

【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。

\*2 ビット 3~0 はフラグをクリアするための 0 ライトのみ可能です。

## 9.2 各レジスタの説明

### 9.2.1 タイマスタートレジスタ (TSTR)

TSTRは8ビットのリード/ライト可能なレジスタで、チャンネル0~2の16TCNTの動作/停止を選択します。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	—	STR2	STR1	STR0
初期値:	1	1	1	1	1	0	0	0
R/W:	—	—	—	—	—	R/W	R/W	R/W

リザーブビット

カウンタスタート2~0  
16TCNT2~16TCNT0の動作/停止を  
選択するビットです。

TSTRはリセット、またはスタンバイモード時に、HF8にイニシャライズされます。

ビット7~3: リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット2: カウンタスタート2 (STR2)

タイマカウンタ2 (16TCNT2) の動作/停止を選択します。

ビット2	
STR2	説 明
0	16TCNT2のカウンタ動作は停止 (初期値)
1	16TCNT2はカウンタ動作

ビット1: カウンタスタート1 (STR1)

タイマカウンタ1 (16TCNT1) の動作/停止を選択します。

ビット1	
STR1	説 明
0	16TCNT1のカウンタ動作は停止 (初期値)
1	16TCNT1はカウンタ動作

ビット0：カウンタスタート0 (STR0)

タイマカウンタ0 (16TCNT0) の動作 / 停止を選択します。

ビット0	説明	
STR0		
0	16TCNT0 のカウント動作は停止	(初期値)
1	16TCNT0 はカウント動作	

## 9.2.2 タイマシンクロレジスタ (TSNC)

TSNC は8ビットのリード / ライト可能なレジスタで、チャンネル0~2の独立動作 / 同期動作を選択します。対応するビットを1にセットしたチャンネルが同期動作を行います。

ビット：	7	6	5	4	3	2	1	0
	—	—	—	—	—	SYNC2	SYNC1	SYNC0
初期値：	1	1	1	1	1	0	0	0
R/W：	—	—	—	—	—	R/W	R/W	R/W

リザーブビット

タイマ同期2~0  
チャンネル2~0の同期動作を設定するビットです。

TSNC はリセット、またはスタンバイモード時に、HF8 にイニシャライズされます。

ビット7~3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット2：タイマ同期2 (SYNC2)

チャンネル2の独立動作 / 同期動作を選択します。

ビット2	説明	
SYNC2		
0	チャンネル2のタイマカウンタ (16TCNT2) は独立動作 (16TCNT2のプリセット / クリアは他チャンネルと無関係)	(初期値)
1	チャンネル2は同期動作 16TCNT2の同期プリセット / 同期クリアが可能	

## ビット1：タイマ同期1 (SYNC1)

チャンネル1の独立動作/同期動作を選択します。

ビット1	
SYNC1	説明
0	チャンネル1のタイマカウンタ(16TCNT1)は独立動作(16TCNT1のプリセット/クリアは他チャンネルと無関係) (初期値)
1	チャンネル1は同期動作 16TCNT1は同期プリセット/同期クリアが可能

## ビット0：タイマ同期0 (SYNC0)

チャンネル0の独立動作/同期動作を選択します。

ビット0	
SYNC0	説明
0	チャンネル0のタイマカウンタ(16TCNT0)は独立動作(16TCNT0のプリセット/クリアは他チャンネルと無関係) (初期値)
1	チャンネル0は同期動作 16TCNT0は同期プリセット/同期クリアが可能

### 9.2.3 タイマモードレジスタ (TMDR)

TMDR は 8 ビットのリード/ライト可能なレジスタで、チャンネル 0~2 の PWM モードの設定、チャンネル 2 の位相計数モードの設定およびオーバーフローフラグ (OVF) のセット条件の設定を行います。

ビット:	7	6	5	4	3	2	1	0
	—	MDF	FDIR	—	—	PWM2	PWM1	PWM0
初期値:	1	0	0	1	1	0	0	0
R/W:	—	R/W	R/W	—	—	R/W	R/W	R/W

ビット 7: リザーブビット  
 ビット 6: 位相計数モード  
 チャンネル 2 を位相計数モードに設定するビットです。  
 ビット 5: フラグディレクション  
 TISRC の OVF フラグセット条件を設定するビットです。  
 ビット 4: リザーブビット  
 ビット 3: リザーブビット  
 ビット 2~0: PWM モード 2~0  
 チャンネル 2~0 を PWM モードに設定するビットです。

TMDR はリセット、またはスタンバイモード時に、H98 にイニシャライズされます。

#### ビット 7: リザーブビット

リザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。

#### ビット 6: 位相計数モード (MDF)

チャンネル 2 を通常動作させるか、位相計数モードで動作させるかを選択します。

ビット 6	説明
MDF	
0	チャンネル 2 は通常動作 (初期値)
1	チャンネル 2 は位相計数モード

MDF ビットを 1 にセットして位相計数モードにすると、16TCNT2 はアップ/ダウンカウンタ、TCLKA、TCLKB 端子がカウントクロック入力端子となります。16TCNT2 は TCLKA、TCLKB 端子の立上がり ( ) 立下がり ( ) の両エッジでカウントされ、カウントアップ/ダウン方向は次のようになります。



カウント方向	カウントダウン				カウントアップ			
TCLKA端子		High		Low	Low		High	
TCLKB端子	Low		High			High		Low

位相計数モードでは、16TCR2のCKEG1、CKEG0ビットによる外部クロックエッジの選択、およびTPSC2～TPSC0ビットによるカウンタクロックの選択は無効となり、上記の位相計数モードの動作が優先されます。

ただし、16TCR2のCCLR1、CCLR0ビットによるカウンタクリア条件の設定、TIOR2、TISRA、TISRB、TISRCのコンペアマッチ/インプットキャプチャ機能と割込みの設定は位相計数モードでも有効です。

ビット5：フラグディレクション (FDIR)

TISRCのOVFフラグのセット条件を設定します。本ビットの設定は、チャンネル2がいずれのモードで動作していても有効となります。

ビット5	説明
FDIR	
0	TISRCのOVFフラグは、16TCNT2がオーバーフローまたはアンダフローしたときに1にセット (初期値)
1	TISRCのOVFフラグは、16TCNT2がオーバーフローしたときに1にセット

ビット4、3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット2：PWMモード2 (PWM2)

チャンネル2を通常動作させるか、PWMモードで動作させるかを選択します。

ビット2	説明
PWM2	
0	チャンネル2は通常動作 (初期値)
1	チャンネル2はPWMモード

PWM2を1にセットしてPWMモードにすると、TIOCA<sub>2</sub>端子はPWM出力端子となり、GRA2のコンペアマッチで1出力、GRB2のコンペアマッチで0出力となります。

## ビット1：PWMモード1（PWM1）

チャンネル1を通常動作させるか、PWMモードで動作させるかを選択します。

ビット1	説明	
PWM1	説明	
0	チャンネル1は通常動作	(初期値)
1	チャンネル1はPWMモード	

PWM1を1にセットしてPWMモードにすると、TIOCA<sub>1</sub>端子はPWM出力端子となり、GRA1のコンペアマッチで1出力、GRB1のコンペアマッチで0出力となります。

## ビット0：PWMモード0（PWM0）

チャンネル0を通常動作させるか、PWMモードで動作させるかを選択します。

ビット0	説明	
PWM0	説明	
0	チャンネル0は通常動作	(初期値)
1	チャンネル0はPWMモード	

PWM0を1にセットしてPWMモードにすると、TIOCA<sub>0</sub>端子はPWM出力端子となり、GRA0のコンペアマッチで1出力、GRB0のコンペアマッチで0出力となります。

### 9.2.4 タイマインタラプトステータスレジスタ A (TISRA)

TISRA は 8 ビットのリード/ライト可能なレジスタで、GRA のコンペアマッチ/インプットキャプチャの発生を示し、GRA のコンペアマッチ/インプットキャプチャ割り込み要求の許可/禁止を制御します。

ビット:	7	6	5	4	3	2	1	0
	—	IMIEA2	IMIEA1	IMIEA0	—	IMFA2	IMFA1	IMFA0
初期値:	1	0	0	0	1	0	0	0
R/W:	—	R/W	R/W	R/W	—	R/(W)*	R/(W)*	R/(W)*

リザーブビット

インプットキャプチャ/コンペアマッチインタラプトイネーブルA2~0  
IMFAフラグによる割り込みを許可/禁止します。

リザーブビット

インプットキャプチャ/コンペアマッチ/ GRAによるコンペアマッチ/インプットキャプチャの発生を示すステータスフラグです。

【注】\* フラグをクリアするための0ライトのみ可能です。

TISRA はリセット、またはスタンバイモード時に、H'88 にイニシャライズされます。

ビット7: リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット6: インプットキャプチャ/コンペアマッチインタラプトイネーブルA2 (IMIEA2)

IMFA2フラグが1にセットされたとき、IMFA2による割り込み要求を許可/禁止します。

ビット6	説明
IMIEA2	
0	IMFA2フラグによる割り込み (IMIA2) 要求を禁止 (初期値)
1	IMFA2フラグによる割り込み (IMIA2) 要求を許可

ビット5：インプットキャプチャ/コンペアマッチインタラプトイネーブルA1(IMIEA1)

IMFA1フラグが1にセットされたとき、IMFA1による割込み要求を許可/禁止します。

ビット5		
IMIEA1	説明	
0	IMFA1フラグによる割込み(IMIA1)要求を禁止	(初期値)
1	IMFA1フラグによる割込み(IMIA1)要求を許可	

ビット4：インプットキャプチャ/コンペアマッチインタラプトイネーブルA0(IMIEA0)

IMFA0フラグが1にセットされたとき、IMFA0による割込み要求を許可/禁止します。

ビット4		
IMIEA0	説明	
0	IMFA0フラグによる割込み(IMIA0)要求を禁止	(初期値)
1	IMFA0フラグによる割込み(IMIA0)要求を許可	

ビット3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット2：インプットキャプチャ/コンペアマッチフラグA2(IMFA2)

GRA2のコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット2		
IMFA2	説明	
0	[クリア条件] (1) IMFA2=1の状態、IMFA2フラグをリードした後、IMFA2フラグに0をライトしたとき (2) IMIA2割込みによりDMACが起動されたとき	(初期値)
1	[セット条件] (1) GRA2がアウトプットコンペアレジスタとして機能している場合、16TCNT2=GRA2になったとき (2) GRA2がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号により16TCNT2の値がGRA2に転送されたとき	

## ビット1：インプットキャプチャ/コンペアマッチフラグ A1 (IMFA1)

GRA1 のコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット1	
IMFA1	説明
0	<p>[クリア条件] (初期値)</p> <p>(1) IMFA1=1 の状態で、IMFA1 フラグをリードした後、IMFA1 フラグに0をライトしたとき</p> <p>(2) IMIA1 割込みにより DMAC が起動されたとき</p>
1	<p>[セット条件]</p> <p>(1) GRA1 がアウトプットコンペアレジスタとして機能している場合、16TCNT1 = GRA1 になったとき</p> <p>(2) GRA1 がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号により 16TCNT1 の値が GRA1 に転送されたとき</p>

## ビット0：インプットキャプチャ/コンペアマッチフラグ A0 (IMFA0)

GRA0 のコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット0	
IMFA0	説明
0	<p>[クリア条件] (初期値)</p> <p>(1) IMFA0=1 の状態で、IMFA0 フラグをリードした後、IMFA0 フラグに0をライトしたとき</p> <p>(2) IMIA0 割込みにより DMAC が起動されたとき</p>
1	<p>[セット条件]</p> <p>(1) GRA0 がアウトプットコンペアレジスタとして機能している場合、16TCNT0 = GRA0 になったとき</p> <p>(2) GRA0 がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号により 16TCNT0 の値が GRA0 に転送されたとき</p>

### 9.2.5 タイマインタラプトステータスレジスタ B (TISR B)

TISR B は 8 ビットのリード/ライト可能なレジスタで、GRB のコンペアマッチ/インプットキャプチャの発生を示し、GRB のコンペアマッチ/インプットキャプチャ割り込み要求の許可/禁止を制御します。

ビット:	7	6	5	4	3	2	1	0
	—	IMIEB2	IMIEB1	IMIEB0	—	IMFB2	IMFB1	IMFB0
初期値:	1	0	0	0	1	0	0	0
R/W:	—	R/W	R/W	R/W	—	R/(W)*	R/(W)*	R/(W)*

リザーブビット  
 インพุットキャプチャ/コンペアマッチインタラプトイネーブル B2~0  
 IMFB フラグによる割り込みを許可/禁止します。

リザーブビット  
 インพุットキャプチャ/コンペアマッチインタラプトイネーブル B2~0  
 IMFB フラグによる割り込みを許可/禁止します。

リザーブビット  
 インพุットキャプチャ/コンペアマッチフラグ B2~0  
 GRB によるコンペアマッチ/インพุットキャプチャの発生を示すステータスフラグです。

【注】\* フラグをクリアするための 0 ライトのみ可能です。

TISR B はリセット、またはスタンバイモード時に、H'88 にイニシャライズされます。

ビット 7: リザーブビット

リザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。

ビット 6: インพุットキャプチャ/コンペアマッチインタラプトイネーブル B2 (IMIEB2)

IMFB2 フラグが 1 にセットされたとき、IMFB2 による割り込み要求を許可/禁止します。

ビット 6		
IMIEB2	説明	
0	IMFB2 フラグによる割り込み (IMIEB2) 要求を禁止	(初期値)
1	IMFB2 フラグによる割り込み (IMIEB2) 要求を許可	

ビット5：インプットキャプチャ/コンペアマッチインタラプトイネーブルB1(IMIEB1)  
IMFB1フラグが1にセットされたとき、IMFB1による割込み要求を許可/禁止します。

ビット5		
IMIEB1	説明	
0	IMFB1フラグによる割込み(IMIB1)要求を禁止	(初期値)
1	IMFB1フラグによる割込み(IMIB1)要求を許可	

ビット4：インプットキャプチャ/コンペアマッチインタラプトイネーブルB0(IMIEB0)  
IMFB0フラグが1にセットされたとき、IMFB0による割込み要求を許可/禁止します。

ビット4		
IMIEB0	説明	
0	IMFB0フラグによる割込み(IMIB0)要求を禁止	(初期値)
1	IMFB0フラグによる割込み(IMIB0)要求を許可	

ビット3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット2：インプットキャプチャ/コンペアマッチフラグB2(IMFB2)

GRB2のコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット2		
IMFB2	説明	
0	[クリア条件] IMFB2=1の状態、IMFB2フラグをリードした後、IMFB2フラグに0をライトしたとき	(初期値)
1	[セット条件] (1) GRB2がアウトプットコンペアレジスタとして機能している場合、16TCNT2 = GRB2になったとき (2) GRB2がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号により16TCNT2の値がGRB2に転送されたとき	

## ビット1：インプットキャプチャ/コンペアマッチフラグ B1 (IMFB1)

GRB1のコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット1	
IMFB1	説明
0	[クリア条件] (初期値) IMFB1=1の状態、IMFB1フラグをリードした後、IMFB1フラグに0をライトしたとき
1	[セット条件] (1) GRB1がアウトプットコンペアレジスタとして機能している場合、16TCNT1=GRB1になったとき (2) GRB1がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号により16TCNT1の値がGRB1に転送されたとき

## ビット0：インプットキャプチャ/コンペアマッチフラグ B0 (IMFB0)

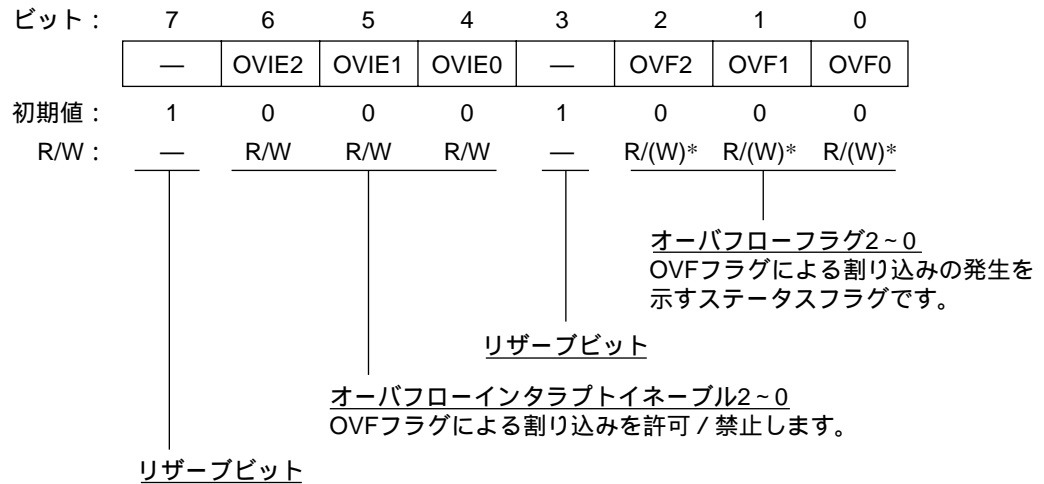
GRB0のコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット0	
IMFB0	説明
0	[クリア条件] (初期値) IMFB0=1の状態、IMFB0フラグをリードした後、IMFB0フラグに0をライトしたとき
1	[セット条件] (1) GRB0がアウトプットコンペアレジスタとして機能している場合、16TCNT0=GRB0になったとき (2) GRB0がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号により16TCNT0の値がGRB0に転送されたとき



### 9.2.6 タイマインタラプトステータスレジスタ C (TISRC)

TISRCは8ビットのリード/ライト可能なレジスタで、16TCNTのオーバーフロー/アンダフローの発生を示し、オーバーフロー割込み要求の許可/禁止を制御します。



【注】\* フラグをクリアするための0ライトのみ可能です。

TISRCはリセット、またはスタンバイモード時に、H'88にイニシャライズされます。

ビット7: リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット6: オーバフローインタラプトイネーブル2 (OVIE2)

OVF2フラグが1にセットされたとき、OVF2による割込み要求を許可/禁止します。

ビット6	説明	
OVIE2		
0	OVF2フラグによる割込み(OVIE2)要求を禁止	(初期値)
1	OVF2フラグによる割込み(OVIE2)要求を許可	

## ビット5：オーバフローインタラプトイネーブル1 (OVIE1)

OVF1 フラグが1にセットされたとき、OVF1 による割込み要求を許可 / 禁止します。

ビット5		
OVIE1	説明	
0	OVF1 フラグによる割込み (OVI1) 要求を禁止	(初期値)
1	OVF1 フラグによる割込み (OVI1) 要求を許可	

## ビット4：オーバフローインタラプトイネーブル0 (OVIE0)

OVF0 フラグが1にセットされたとき、OVF0 による割込み要求を許可 / 禁止します。

ビット4		
OVIE0	説明	
0	OVF0 フラグによる割込み (OVI0) 要求を禁止	(初期値)
1	OVF0 フラグによる割込み (OVI0) 要求を許可	

## ビット3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

## ビット2：オーバフローフラグ2 (OVF2)

16TCNT2のオーバフローの発生を示すステータスフラグです。

ビット2		
OVF2	説明	
0	[クリア条件] OVF2 = 1 の状態で、OVF2 フラグをリードした後、OVF2 フラグに0をライトしたとき	(初期値)
1	[セット条件] 16TCNT2の値がオーバフロー (H'FFFF H'0000) またはアンダフロー (H'0000 H'FFFF) したとき	

【注】 16TCNT のアンダフローは、16TCNT がアップ / ダウンカウントとして機能している場合に発生します。したがって、次の場合にのみアンダフローが発生することがあります。  
チャンネル2が位相計数モードに設定されているとき (TMDR の MDF = 1)

## ビット1：オーバフローフラグ1 (OVF1)

16TCNT1のオーバフローの発生を示すステータスフラグです。

ビット1	
OVF1	説明
0	[クリア条件] (初期値) OVF1 = 1の状態、OVF1フラグをリードした後、OVF1フラグに0をライトしたとき
1	[セット条件] 16TCNT1の値がオーバフロー (H'FFFF H'0000) したとき

## ビット0：オーバフローフラグ0 (OVF0)

16TCNT0のオーバフローの発生を示すステータスフラグです。

ビット0	
OVF0	説明
0	[クリア条件] (初期値) OVF0 = 1の状態、OVF0フラグをリードした後、OVF0フラグに0をライトしたとき
1	[セット条件] 16TCNT0の値がオーバフロー (H'FFFF H'0000) したとき

## 9.2.7 タイマカウンタ (16TCNT)

16TCNTは16ビットのカウンタです。16ビットタイマには、各チャンネル1本、計3本の16TCNTがあります。

チャンネル	略称	機能
0	16TCNT0	アップカウンタ
1	16TCNT1	
2	16TCNT2	位相計数モード：アップ/ダウンカウンタ 上記以外：アップカウンタ

ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

初期値： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

R/W： R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W

16TCNTは16ビットのリード/ライト可能なレジスタで、入力したクロックによりカウント動作を行います。入力するクロックは、16TCRのTPSC2～TPSC0ビットにより選択します。

16TCNT0、16TCNT1はアップカウント動作を行います。

16TCNT2は位相計数モード時、アップ/ダウンカウント動作を行い、それ以外の場合はアップカウント動作します。

16TCNTは、対応するGRA、GRBとのコンペアマッチ、またはGRA、GRBへのインプットキャプチャによりH'0000にクリアすることができます(カウンタクリア機能)。

16TCNTがオーバフロー(H'FFFF H'0000)すると、TISRCの対応するチャンネルのOVFフラグが1にセットされます。

16TCNTがアンダフロー(H'0000 H'FFFF)すると、TISRCの対応するチャンネルのOVFフラグが1にセットされます。

16TCNTはCPUと内部16ビットバスで接続されており、ワード/バイト単位のリード/ライトが可能です。

16TCNTはリセット、またはスタンバイモード時にH'0000にイニシャライズされます。

### 9.2.8 ジェネラルレジスタ A、B (GRA、GRB)

GR は、16 ビットのレジスタです。16 ビットタイマには、各チャンネル2本、計6本のジェネラルレジスタがあります。

チャンネル	略 称	機 能
0	GRA0、GRB0	アウトプットコンペア/インプットキャプチャ兼用レジスタ
1	GRA1、GRB1	
2	GRA2、GRB2	

ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

初期値： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

R/W： R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W

GR は16ビットのリード/ライト可能なレジスタで、アウトプットコンペアレジスタとインプットキャプチャレジスタの両方の機能をもっています。機能の切替えは、TIOR により行います。

アウトプットコンペアレジスタとして使用しているときは、GRA/GRB の値と 16TCNT の値は常に比較されています。両者の値が一致 (コンペアマッチ) すると、TISRA/TISRB の IMFA/IMFB フラグが1にセットされます。TIOR によりコンペアマッチ出力を設定することができます。

インプットキャプチャレジスタとして使用しているときは、外部からのインプットキャプチャ信号を検出して、16TCNT の値を格納します。このとき対応する TISRA/TISRB の IMFA/IMFB フラグが1にセットされます。インプットキャプチャ信号の検出エッジは TIOR により行います。

PWM モードに設定されている場合には、TIOR の設定は無視されます。

GR は CPU と内部 16 ビットバスで接続されており、ワード/バイト単位のリード/ライトが可能です。

GR はリセット、またはスタンバイモード時にアウトプットコンペアレジスタ (端子出力なし) に設定され、H'FFFF にイニシャライズされます。

### 9.2.9 タイマコントロールレジスタ (16TCR)

16TCRは8ビットのレジスタです。16ビットタイマには、各チャンネル1本、計3本の16TCRがあります。

チャンネル	略称	機能
0	16TCR0	16TCRは16TCNTの制御を行います。
1	16TCR1	各チャンネルの16TCRは同一の機能をもっています。
2	16TCR2	チャンネル2を位相計数モードに設定したとき、16TCR2のCKEG1、CKEG0ビットおよびTPSC2~TPSC0ビットの設定は無効となります。

ビット:	7	6	5	4	3	2	1	0
	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
初期値:	1	0	0	0	0	0	0	0
R/W:	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W

リザーブビット  
 カウンタクリア1、0  
 カウンタクリア要因を選択するビットです。

クロックエッジ1、0  
 クロックの検出エッジを選択するビットです。

タイムプリスケラ2~0  
 16TCNTのカウントクロック  
 を選択するビットです。

16TCRは8ビットのリード/ライト可能なレジスタで、16TCNTのカウントクロックの選択、外部クロック選択時のエッジの選択、およびカウンタクリア要因の選択を行います。

16TCRはリセット、またはスタンバイモード時にH'80にイニシャライズされます。

ビット7: リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

## ビット6、5：カウンタクリア1、0（CCLR1、CCLR0）

16TCNTのカウンタクリア要因を選択します。

ビット6	ビット5	説明
CCLR1	CCLR0	
0	0	16TCNTのクリア禁止（初期値）
	1	GRAのコンペアマッチ/インプットキャプチャ*1で16TCNTをクリア
1	0	GRBのコンペアマッチ/インプットキャプチャ*1で16TCNTをクリア
	1	同期クリア。同期動作*2をしている他のタイマのカウンタクリアに同期して16TCNTをクリア

【注】 \*1 GRがアウトプットコンペアレジスタとして機能しているとき、コンペアマッチにより、クリアされます。GRがインプットキャプチャレジスタとして機能しているとき、インプットキャプチャによりクリアされます。

\*2 同期動作の設定はTSNCにより行います。

## ビット4、3：クロックエッジ1、0（CKEG1、CKEG0）

外部クロック選択時に、外部クロックの入力エッジを選択します。

ビット4	ビット3	説明
CKEG1	CKEG0	
0	0	立上がりエッジでカウント（初期値）
0	1	立下がりエッジでカウント
1		立上がり/立下がりエッジの両エッジでカウント

チャンネル2が位相計数モードに設定されているとき、16TCR2のCKEG1、CKEG0ビットの設定は無効になり、位相計数モードの動作が優先されます。

ビット2~0：タイマプリスケラ2~0 (TPSC2~TPSC0)

16TCNTのカウントクロックを選択します。

ビット2	ビット1	ビット0	説 明
TPSC2	TPSC1	TPSC0	
0	0	0	内部クロック： でカウント (初期値)
		1	内部クロック： /2 でカウント
	1	0	内部クロック： /4 でカウント
		1	内部クロック： /8 でカウント
1	0	0	外部クロック A：TCLKA 端子入力でカウント
		1	外部クロック B：TCLKB 端子入力でカウント
	1	0	外部クロック C：TCLKC 端子入力でカウント
		1	外部クロック D：TCLKD 端子入力でカウント

TPSC2 ビットを0にクリアして内部クロックを選択した場合、クロックの立下がりエッジでカウントされます。また、TPSC2 ビットを1にセットして外部クロックを選択した場合、カウントエッジはCKEG1、CKEG0 ビットの設定に従います。

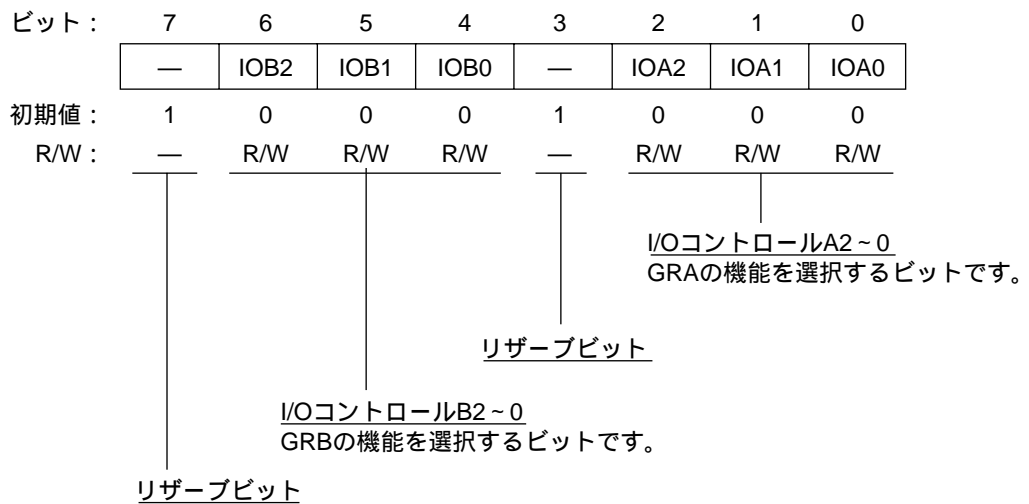
チャンネル2が位相計数モードに設定されているとき(TMDRのMDFビット=1) 16TCR2のTPSC2~TPSC0のビットの設定は無効となり、位相計数モードの動作が優先されます。



### 9.2.10 タイマ I/O コントロールレジスタ (TIOR)

TIOR は 8 ビットのレジスタです。16 ビットタイマには、各チャンネル 1 本、計 3 本の TIOR があります。

チャンネル	略 称	機 能
0	TIOR0	TIOR は GR の制御を行います。
1	TIOR1	PWM モード時、一部機能が異なります。
2	TIOR2	



TIOR は 8 ビットのリード/ライト可能なレジスタで、GRA、GRB をアウトプットコンペアレジスタとして使用するか、インプットキャプチャレジスタとして使用するかを選択します。また TIOA、TIOB 端子の機能を選択します。アウトプットコンペアレジスタを選択した場合は出力設定を選択し、インプットキャプチャレジスタを選択した場合はインプットキャプチャ信号の入力エッジを選択します。

TIOR はリセット、またはスタンバイモード時に、H'88 にイニシャライズされます。

#### ビット 7：リザーブビット

リザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。

## ビット6～4：I/OコントロールB2～0（IOB2～IOB0）

GRBの機能を選択します。

ビット6	ビット5	ビット4	説明	
IOB2	IOB1	IOB0		
0	0	0	GRBはアウトプットコンペアレジスタ	コンペアマッチによる端子出力禁止（初期値）
		1		GRBのコンペアマッチで0出力* <sup>1</sup>
	1	0		GRBのコンペアマッチで1出力* <sup>1</sup>
		1		GRBのコンペアマッチでトグル出力* <sup>1</sup> * <sup>2</sup> （チャンネル2のみ1出力）
1	0	0	GRBはインプットキャプチャレジスタ	立上がりエッジでGRBヘインプットキャプチャ
		1		立下がりエッジでGRBヘインプットキャプチャ
	1	0		立上がり/立下がりの両エッジでインプットキャプチャ
		1		

【注】 \*1 リセット後、最初のコンペアマッチが発生するまでの出力値は TOLR の設定に従います。

\*2 チャンネル2はコンペアマッチによるトグル出力機能がありません。この設定にすると自動的に1出力が選択されます。

## ビット3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット2～0：I/OコントロールA2～0（IOA2～IOA0）

GRAの機能を選択します。

ビット2	ビット1	ビット0	説明	
IOA2	IOA1	IOA0		
0	0	0	GRAはアウトプットコンペアレジスタ	コンペアマッチによる端子出力禁止（初期値）
		1		GRAのコンペアマッチで0出力*1
	1	0		GRAのコンペアマッチで1出力*1
		1		GRAのコンペアマッチでトグル出力*1*2（チャンネル2のみ1出力）
1	0	0	GRAはインプットキャプチャレジスタ	立上がりエッジでGRAへインプットキャプチャ
		1		立下がりエッジでGRAへインプットキャプチャ
	1	0		立上がり/立下がりの両エッジでインプットキャプチャ
		1		

【注】 \*1 リセット後、最初のコンペアマッチが発生するまでの出力値は TOLR の設定に従います。

\*2 チャンネル2はコンペアマッチによるトグル出力機能がありません。この設定にすると自動的に1出力が選択されます。

### 9.2.11 タイマアウトプットレベルセットレジスタ (TOLR)

TOLRは8ビットのライト専用のレジスタで、チャンネル0~2のタイマ出力レベルの設定を行います。

ビット:	7	6	5	4	3	2	1	0
	—	—	TOB2	TOA2	TOB1	TOA1	TOB0	TOA0
初期値:	1	1	0	0	0	0	0	0
R/W:	—	—	W	W	W	W	W	W

リザーブビット

出力レベルセットA2~0、B2~0  
タイマ出力 (TIOCA<sub>2~0</sub>、TIOCB<sub>2~0</sub>)  
のレベルを設定するビットです。

TOLRはTSTRの対応するビットが0の時のみ設定が可能です。

TOLRレジスタはライト専用レジスタで、リードは無効です。リードすると、1が読み出されます。

TOLRはリセット、またはスタンバイモード時に、H'COにイニシャライズされます。

ビット7~6: リザーブビット

リード/ライトは無効です。

ビット5: 出力レベルセット B2 (TOB2)

タイマ出力 (TIOCB<sub>2</sub>) の値を設定します。

ビット5	
TOB2	説 明
0	TIOCB <sub>2</sub> は0出力 (初期値)
1	TIOCB <sub>2</sub> は1出力

ビット4: 出力レベルセット A2 (TOA2)

タイマ出力 (TIOCA<sub>2</sub>) の値を設定します。

ビット4	
TOA2	説 明
0	TIOCA <sub>2</sub> は0出力 (初期値)
1	TIOCA <sub>2</sub> は1出力

## ビット3：出力レベルセット B1 (TOB1)

タイマ出力 (TIOCB<sub>1</sub>) の値を設定します。

ビット3		
TOB1	説明	
0	TIOCB <sub>1</sub> は 0 出力	(初期値)
1	TIOCB <sub>1</sub> は 1 出力	

## ビット2：出力レベルセット A1 (TOA1)

タイマ出力 (TIOCA<sub>1</sub>) の値を設定します。

ビット2		
TOA1	説明	
0	TIOCA <sub>1</sub> は 0 出力	(初期値)
1	TIOCA <sub>1</sub> は 1 出力	

## ビット1：出力レベルセット B0 (TOB0)

タイマ出力 (TIOCB<sub>0</sub>) の値を設定します。

ビット1		
TOB0	説明	
0	TIOCB <sub>0</sub> は 0 出力	(初期値)
1	TIOCB <sub>0</sub> は 1 出力	

## ビット0：出力レベルセット A0 (TOA0)

タイマ出力 (TIOCA<sub>0</sub>) の値を設定します。

ビット0		
TOA0	説明	
0	TIOCA <sub>0</sub> は 0 出力	(初期値)
1	TIOCA <sub>0</sub> は 1 出力	

## 9.3 CPU とのインタフェース

### 9.3.1 16ビットアクセス可能なレジスタ

16TCNT、GRA、GRB は 16ビットのレジスタです。これらのレジスタは、CPU と内部 16ビットデータバスで接続されており、ワード単位のリード/ライトが可能です。また、バイト単位のリード/ライトもできます。

16TCNT に対してワード単位のリード/ライトを行った場合の動作を図 9.4、図 9.5 に示します。

また、16TCNTH、16TCNTL に対してワード単位のリード/ライトを行った場合の動作を図 9.6、図 9.7、図 9.8、図 9.9 に示します。

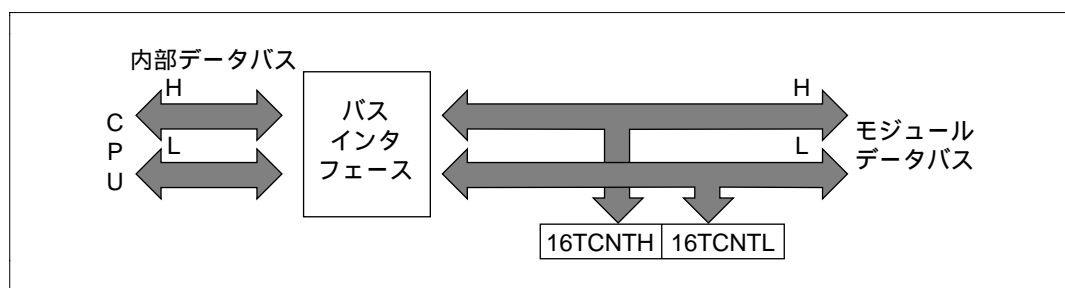


図 9.4 16TCNT のアクセス動作 [ CPU 16TCNT (ワード) ]

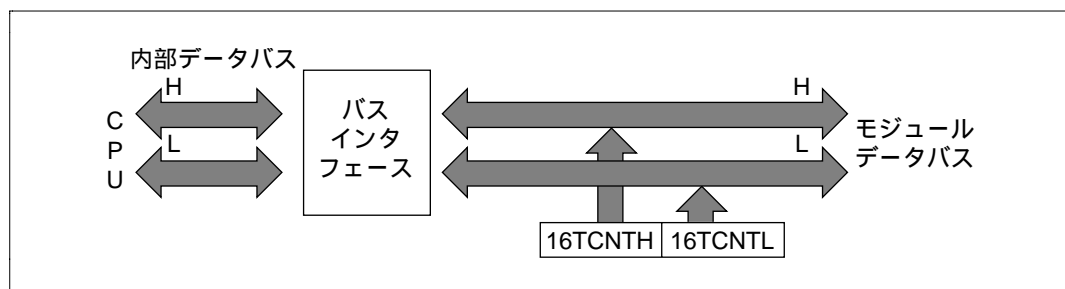


図 9.5 16TCNT のアクセス動作 [ 16TCNT CPU (ワード) ]

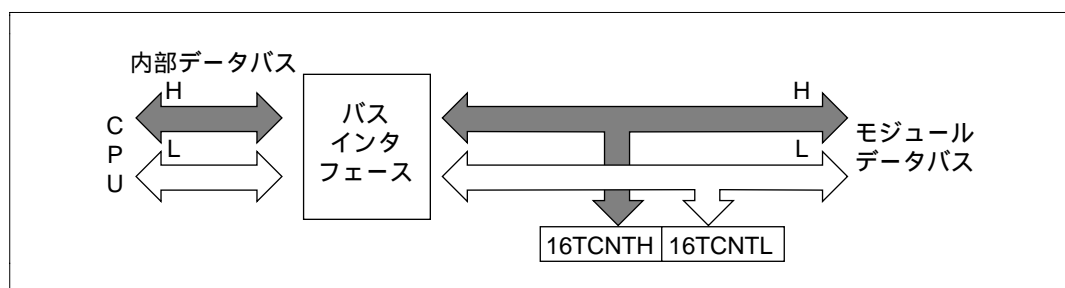


図 9.6 16TCNTH のアクセス動作 [ CPU 16TCNTH (上位バイト) ]

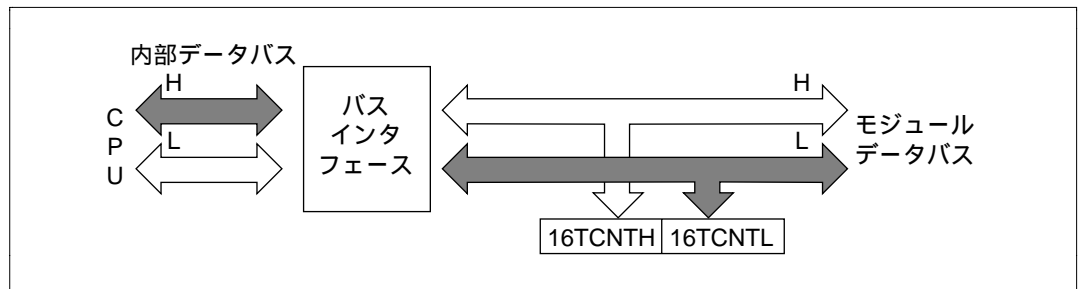


図 9.7 16TCNTLのアクセス動作 [CPU 16TCNTL (下位バイト)]

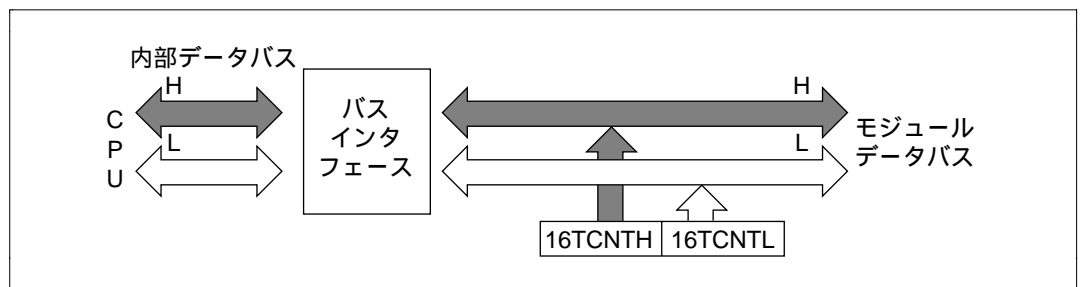


図 9.8 16TCNTHのアクセス動作 [16TCNTH CPU (上位バイト)]

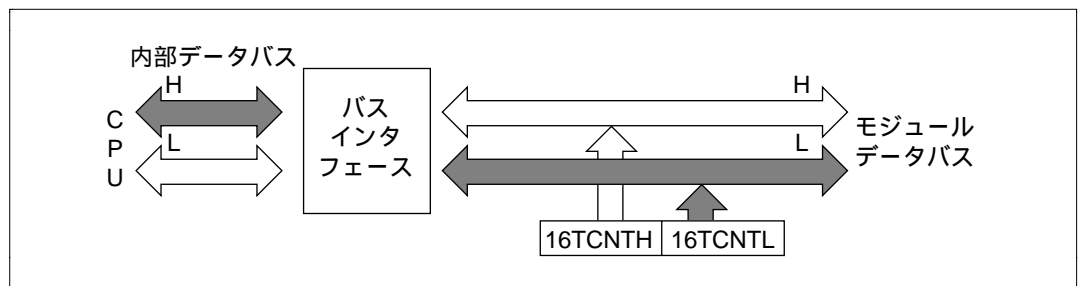


図 9.9 16TCNTLのアクセス動作 [16TCNTL CPU (下位バイト)]

### 9.3.2 8ビットアクセスのレジスタ

16TCNT、GR 以外のレジスタは 8 ビットのレジスタです。これらのレジスタは、CPU と内部 8 ビットデータバスで接続されています。

16TCR に対してバイト単位のリード/ライトを行った場合の動作を図 9.10、図 9.11 に示します。

なお、ワードサイズの転送命令を実行するとバイト単位 2 回の転送が行われます。

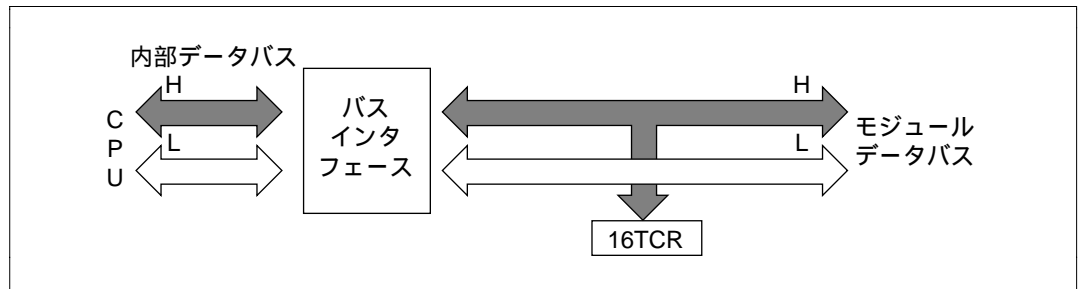


図 9.10 16TCR のアクセス動作 [ CPU 16TCR ]

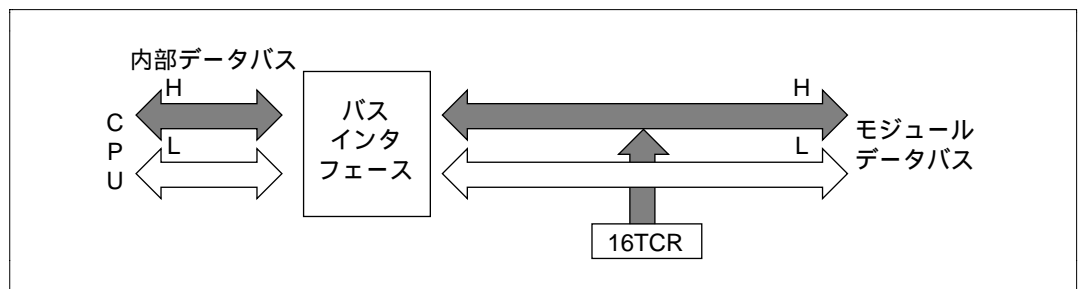


図 9.11 16TCR のアクセス動作 [ 16TCR CPU ]



## 9.4 動作説明

### 9.4.1 概要

以下に各モードの動作概要を示します。

#### (1) 通常動作

各チャンネルには、16TCNTとGRがあります。16TCNTは、アップカウント動作を行い、フリーランニング動作、周期カウント動作、または外部イベントカウント動作が可能です。

GRA、GRBは、それぞれインプットキャプチャレジスタまたはアウトプットコンペアレジスタとして使用することができます。

#### (2) 同期動作

同期動作を設定したチャンネルの16TCNTは、同期プリセット動作を行います。すなわち、同期動作に設定されたチャンネルのうち任意の16TCNTを書き換えると他のチャンネルの16TCNTも同時に書き換えられます。また、同期動作に設定された複数のチャンネルの16TCRのCCLR1、CCLR0ビットの設定により、16TCNTの同期クリアが可能です。

#### (3) PWMモード

TIOCA端子からPWM波形を出力するモードです。コンペアマッチAにより1出力、コンペアマッチBにより0出力となります。GRA、GRBの設定により、デューティ0~100%のPWM波形を出力できます。PWMモードに設定すると当該チャンネルのGRA、GRBは自動的にアウトプットコンペアレジスタとして機能します。

#### (4) 位相計数モード

TCLKA、TCLKB端子から入力される2つのクロックの位相差を検出して、16TCNT2をアップ/ダウンカウント動作させるモードです。位相計数モードに設定すると、TCLKA、TCLKB端子はクロック入力となり、また16TCNT2はアップ/ダウンカウント動作を行います。

## 9.4.2 基本機能

### (1) カウンタの動作

タイマスタートレジスタ (TSTR) の STR0~STR2 ビットを 1 にセットすると、対応するチャンネルの 16TCNT はカウント動作を開始します。フリーランニングカウンタ動作、周期カウンタ動作などが可能です。

#### (a) カウント動作の設定手順例

カウント動作の設定手順例を図 9.12 に示します。

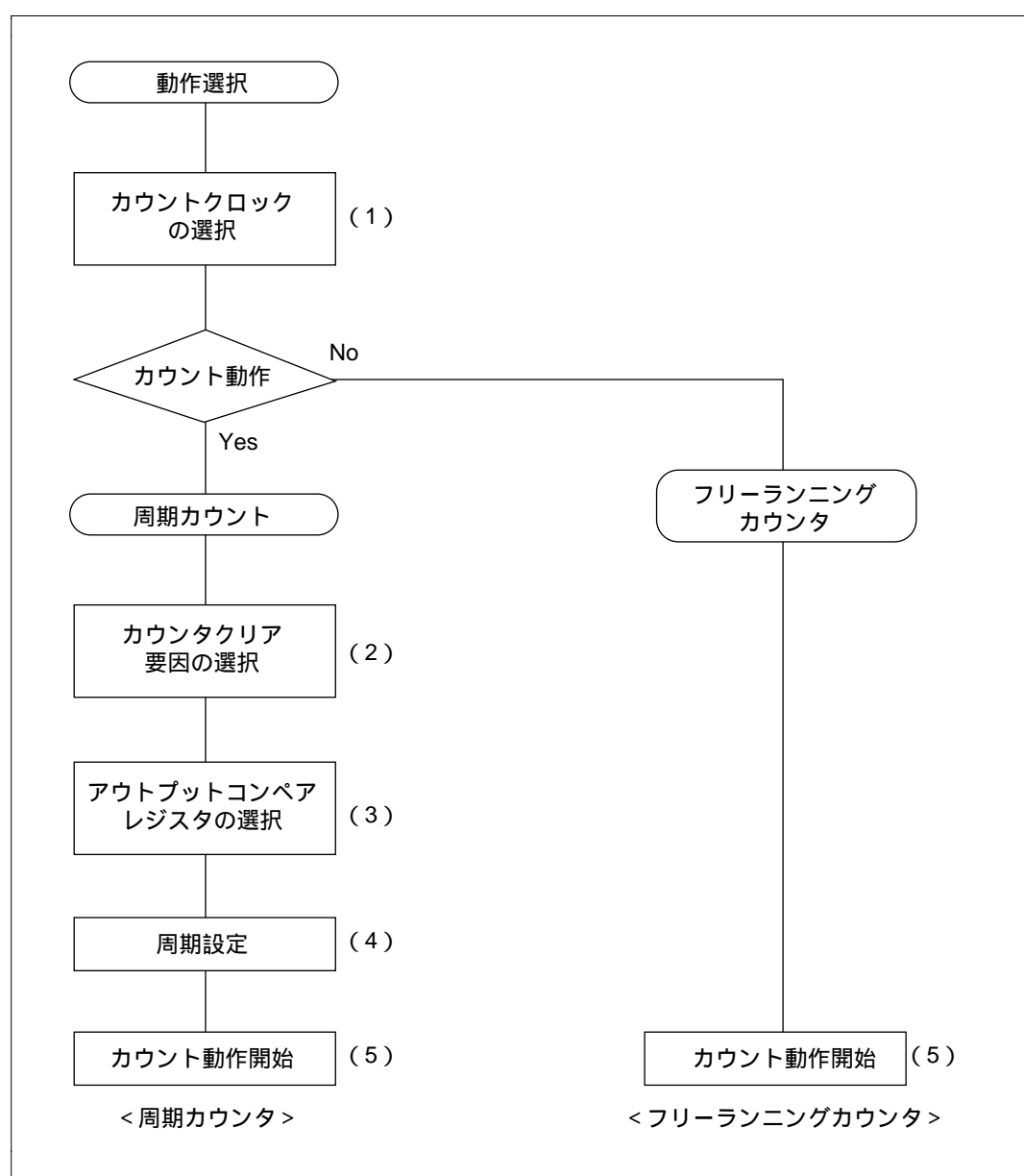


図 9.12 カウント動作設定手順例

- (1) 16TCR の TPSC2 ~ TPSC0 ビットでカウンタクロックを選択してください。外部クロックを選択した場合は、16TCR の CKEG1、CKEG0 ビットで外部クロックのエッジを選択してください。
- (2) 周期カウント動作の場合、16TCR の CCLR1、CCLR0 ビットで 16TCNT を GRA のコンペアマッチでクリアするか、GRB のコンペアマッチでクリアするかを選択してください。
- (3) (2) で選択した GRA または GRB を、TIOR によりアウトプットコンペアレジスタに設定してください。
- (4) (2) で選択した GRA または GRB に周期を設定してください。
- (5) TSTR の STR ビットを 1 にセットしてカウント動作を開始してください。

(b) フリーランニングカウント動作と周期カウント動作

チャンネル0~2のカウンタ(16TCNT)はリセット直後は、すべてフリーランニングカウンタの設定となっており、TSTRの対応するビットを1にセットするとフリーランニングカウンタとしてアップカウント動作を開始します。16TCNTがオーバーフロー(H'FFFF H'0000)するとTISRCのOVFフラグが1にセットされます。16TCNTはオーバーフロー後、H'0000から再びアップカウント動作を継続します。

フリーランニングカウンタの動作を図9.13に示します。

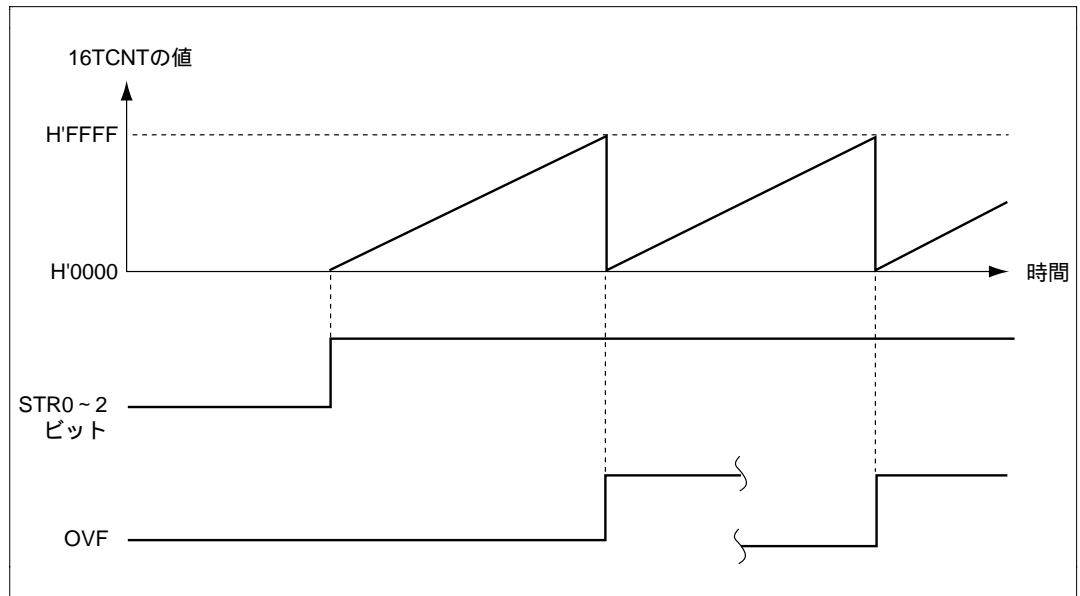


図 9.13 フリーランニングカウンタの動作

16TCNTのクリア要因にコンペアマッチを選択したときは、当該チャネルの16TCNTは周期カウント動作を行います(周期設定用のGRAまたはGRBをアウトプットコンペアレジスタに設定し、16TCRのCCLR1、CCLR0ビットにより、コンペアマッチによるカウンタクリアを選択します)。設定後、TSTRの対応するビットを1にセットすると周期カウンタとしてアップカウント動作を開始します。カウント値がGRA/GRBの値と一致するとTISRA/TISRBのIMFA/IMFBフラグが1にセットされ、16TCNTはH'0000にクリアされます。

このとき、対応するTISRA/TISRBのIMIEA/IMIEBビットが1ならば、CPUに割込みを要求します。16TCNTはコンペアマッチ後、H'0000から再びアップカウント動作を続けます。

周期カウンタの動作を図9.14に示します。

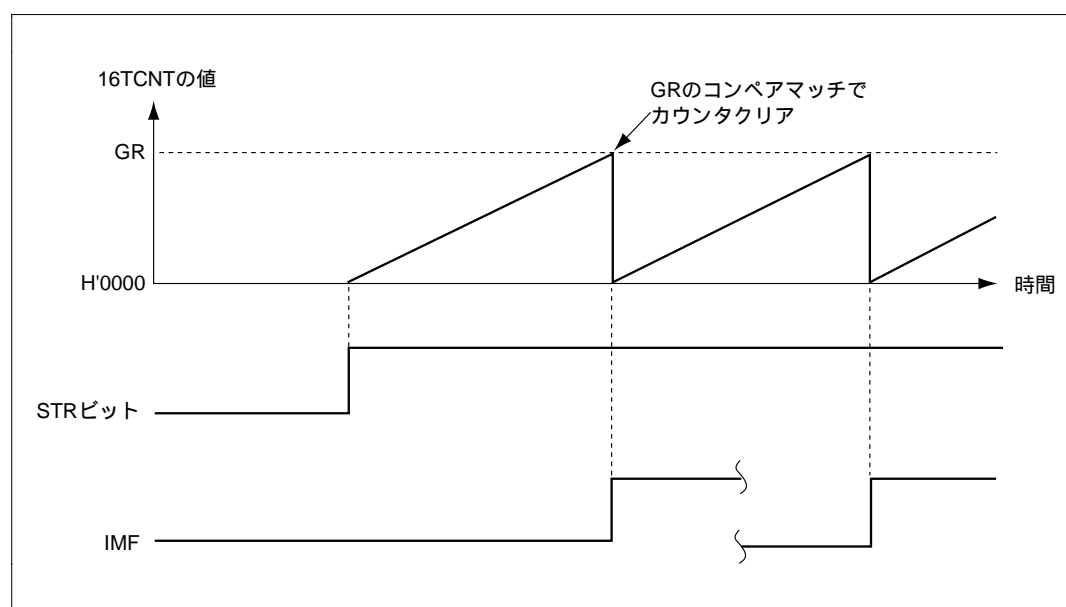


図 9.14 周期カウンタの動作

(c) 16TCNT のカウントタイミング

(1) 内部クロック動作の場合

16TCR の TPSC2 ~ TPSC0 ビットにより、システムクロック ( ) またはシステムクロックを分周した3種類のクロック ( /2、 /4、 /8 ) が選択できます。

このときのタイミングを図9.15に示します。

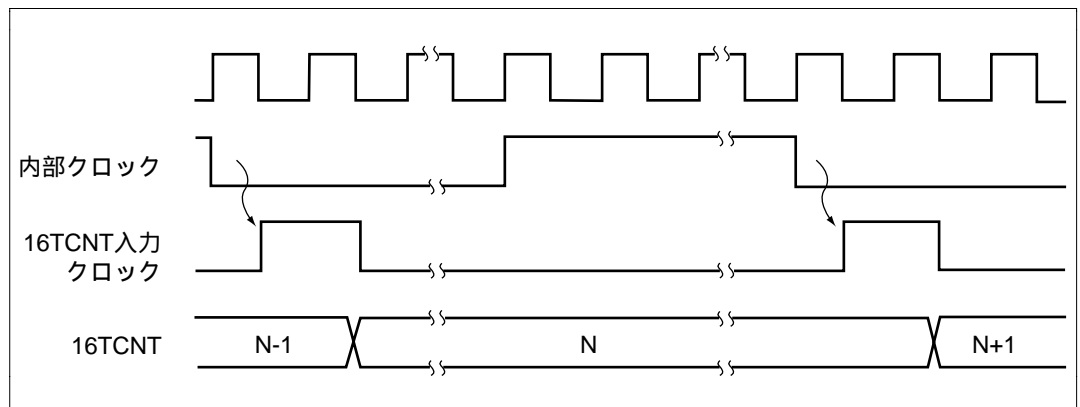


図 9.15 内部クロック動作時のカウントタイミング

(2) 外部クロック動作の場合

16TCR の TPSC2 ~ TPSC0 ビットにより、外部クロック入力端子 (TCLKA ~ TCLKD) を、また CKEG1、CKEG0 ビットにより検出エッジを選択できます。外部クロックの検出は立上がりエッジ / 立下がりエッジ / 両エッジの選択が可能です。

なお、外部クロックのパルス幅は、単エッジの場合は1.5システムクロック以上、両エッジの場合は2.5システムクロック以上必要です。これ以下のパルス幅では正しく動作しませんので注意してください。

立上がり / 立下がりの両エッジ検出時のタイミングを図9.16に示します。

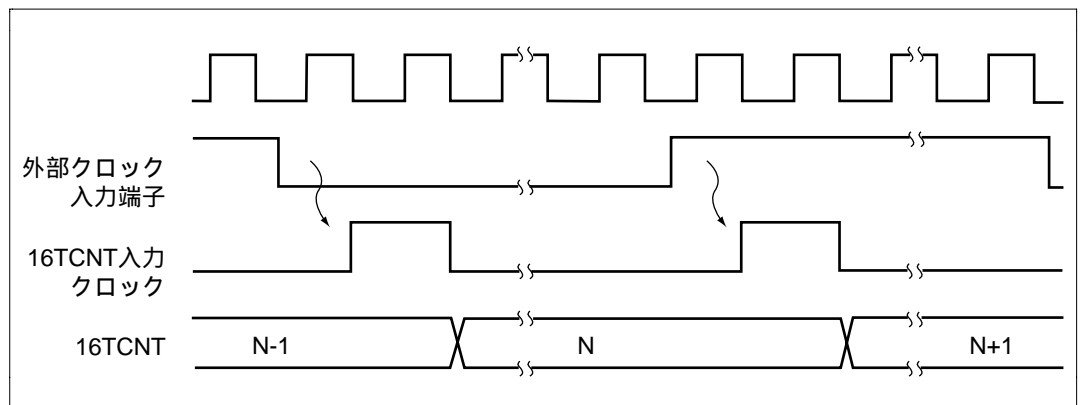


図 9.16 外部クロック動作時のカウントタイミング (両エッジ検出の場合)

## (2) コンペアマッチによる波形出力機能

16ビットタイマチャンネル0、1は、コンペアマッチA、Bにより対応するTIOCA、TIOCB端子から0出力/1出力/トグル出力を行うことができます。

チャンネル2は0出力/1出力のみ可能です。

### (a) コンペアマッチによる波形出力動作の設定手順例

コンペアマッチによる波形出力動作の設定手順例を図9.17に示します。

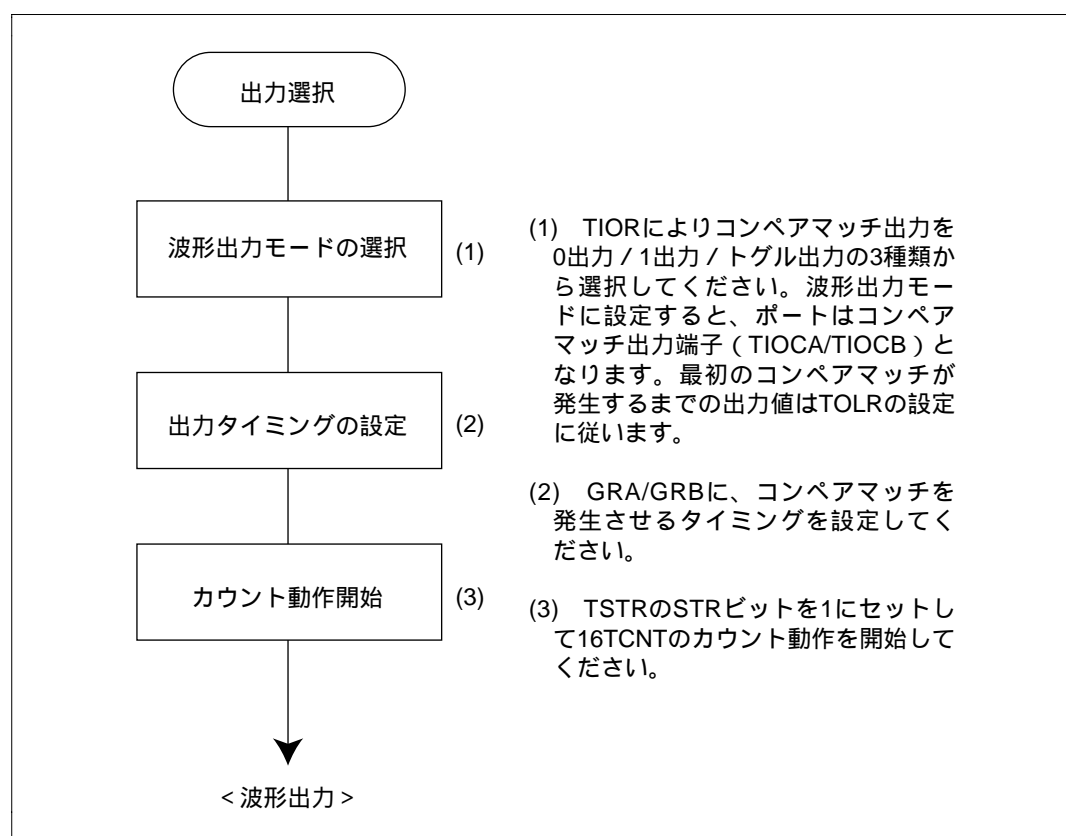


図9.17 コンペアマッチによる波形出力動作例

(b) 波形出力動作例

0出力 / 1出力の例を図9.18に示します。

16TCNTフリーランニングカウント動作、またコンペアマッチAにより0出力、コンペアマッチBにより1出力となるように設定した場合の例です。設定したレベルと端子のレベルが一致した場合には、端子のレベルは変化しません。

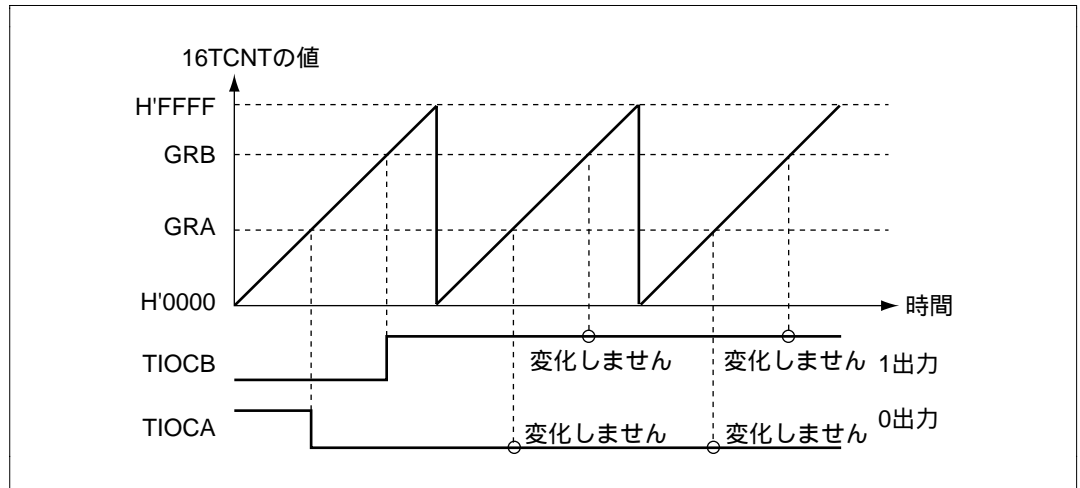


図 9.18 0出力、1出力の動作例 (TOA=1、TOB=0の場合)

トグル出力の例を図9.19に示します。

16TCNTを周期カウント動作 (コンペアマッチBでカウンタクリア) に、コンペアマッチA、Bともトグル出力となるように設定した場合の例です。

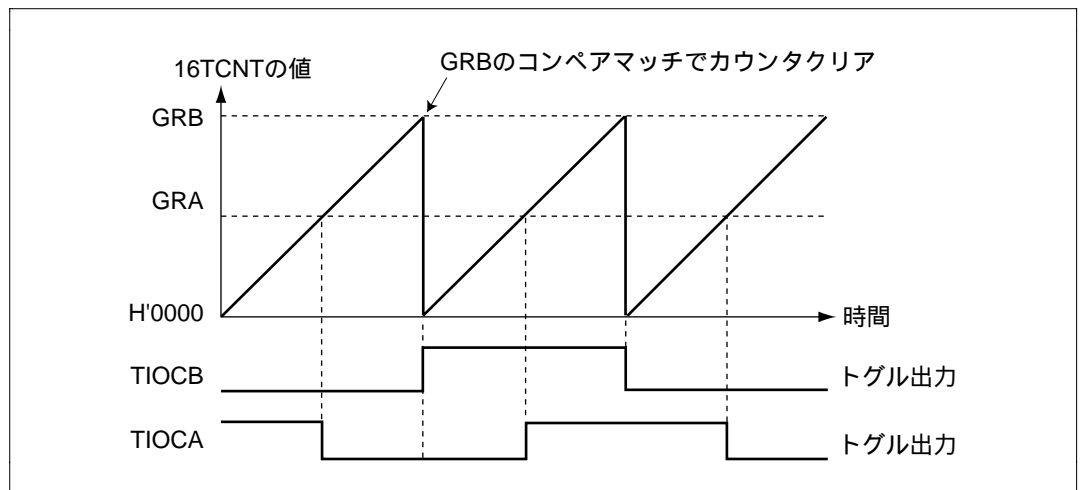


図 9.19 トグル出力の動作例 (TOA=1、TOB=0の場合)

## (c) アウトプットコンペア出力タイミング

コンペアマッチ信号は、16TCNT と GR が一致した最後のステート（16TCNT が一致したカウント値を更新するタイミング）で発生します。コンペアマッチ信号が発生したとき、TIOCR で設定される出力値がアウトプットコンペア出力端子（TIOCA、TIOCB）に出力されます。16TCNT と GR が一致した後、16TCNT 入力クロックが発生するまでコンペアマッチ信号は発生しません。

アウトプットコンペア出力タイミングを図 9.20 に示します。

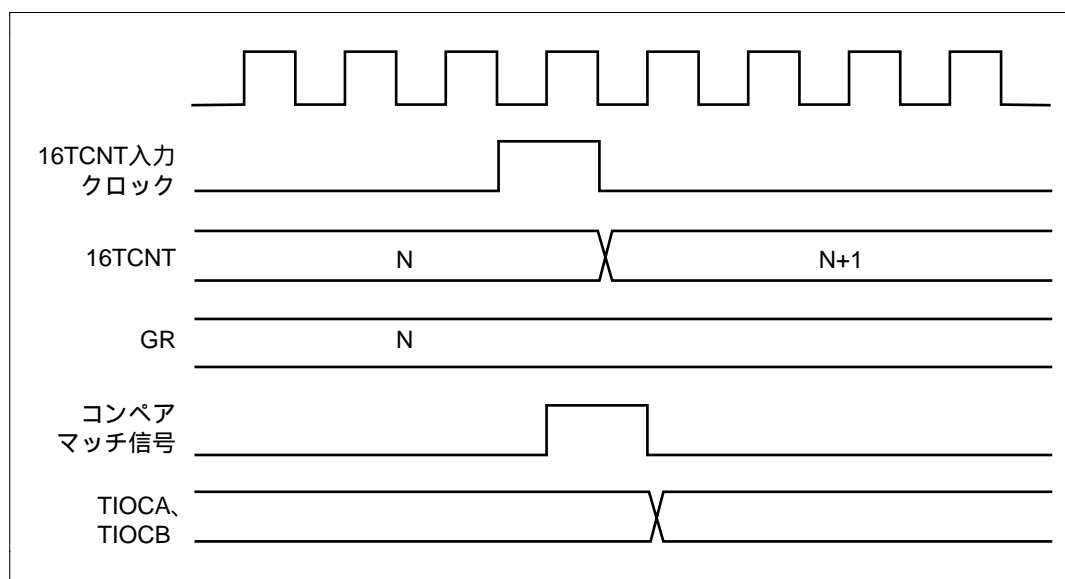


図 9.20 アウトプットコンペア出力タイミング



### (3) インพุットキャプチャ機能

インพุットキャプチャ/アウトプットコンペア端子 (TIOCA、TIOCB) の入力エッジを検出して 16TCNT の値を GR に転送することができます。検出エッジは立上がりエッジ/立下がりエッジ/両エッジから選択できます。

インพุットキャプチャ機能を利用することで、パルス幅や周期の測定を行うことができます。

#### (a) インพุットキャプチャ動作の設定手順例

インพุットキャプチャ動作の設定手順例を図 9.21 に示します。

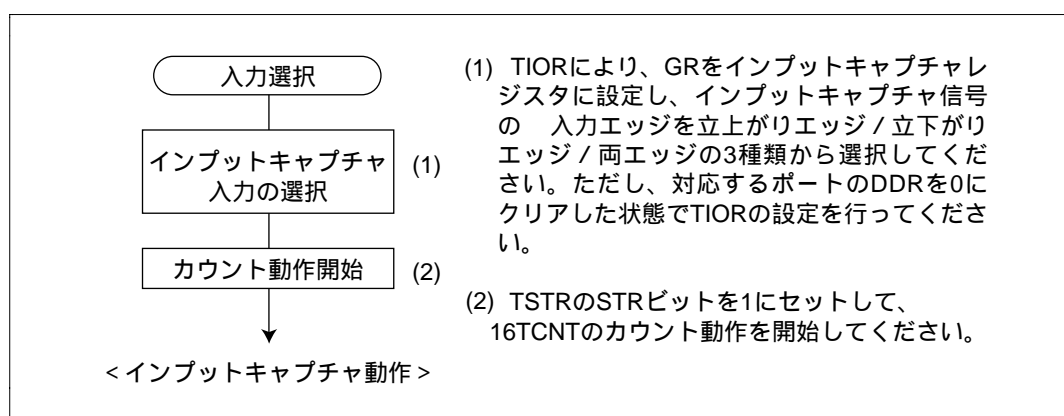


図 9.21 インพุットキャプチャ動作の設定手順例

## (b) インพุットキャプチャ動作例

インพุットキャプチャ動作例を図9.22に示します。

TIOCA 端子のインพุットキャプチャ入力エッジは立上がり / 立下がりの両エッジ、また TIOCB 端子のインพุットキャプチャ入力エッジは立下がりエッジを選択し、16TCNT は GRB のインพุットキャプチャでカウンタクリアされるように設定した場合の例です。

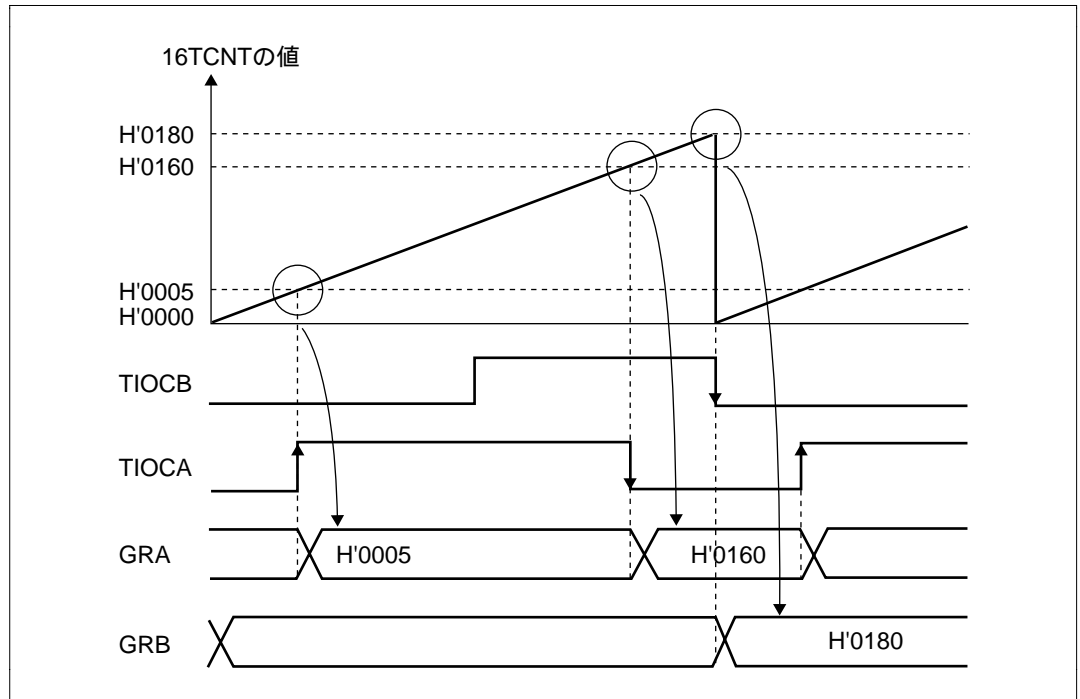


図 9.22 インพุットキャプチャ動作例

## (c) インพุットキャプチャ信号タイミング

インพุットキャプチャ入力は、TIOR の設定により立上がりエッジ / 立下がりエッジ / 両エッジの選択ができます。

立上がりエッジを選択した場合のタイミングを図 9.23 に示します。

インพุットキャプチャ入力信号のパルス幅は、単エッジの場合は 1.5 システムクロック以上、両エッジの場合は 2.5 システムクロック以上必要です。

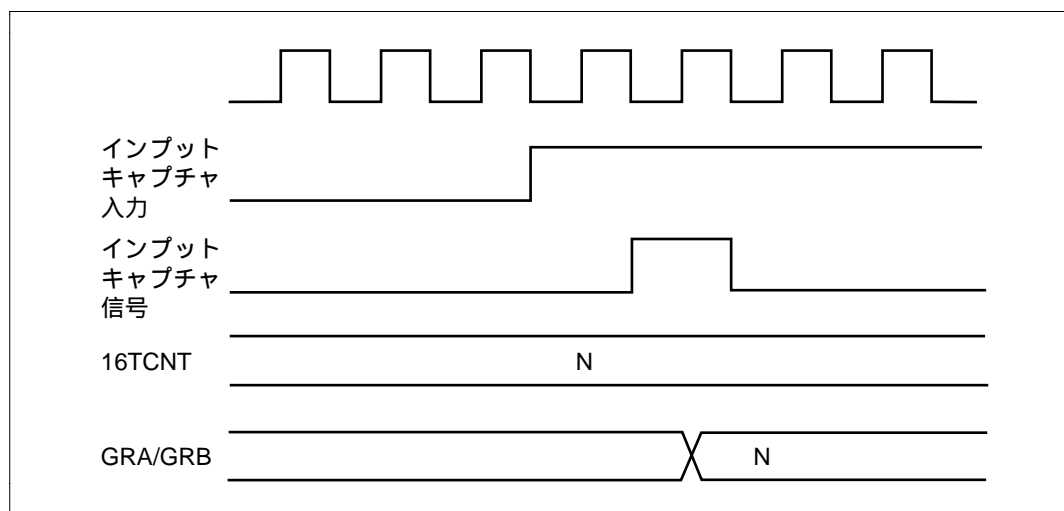


図 9.23 インพุットキャプチャ入力信号タイミング

### 9.4.3 同期動作

同期動作は、複数の 16TCNT の値を同時に書き換えることができます（同期プリセット）。また、16TCR の設定により複数の 16TCNT を同時にクリアすることができます（同期クリア）。

同期動作により、1つのタイムベースに対してジェネラルレジスタを増加することができます。

チャンネル0~2はすべて同期動作の設定が可能です。

#### (1) 同期動作の設定手順例

同期動作の設定手順例を図9.24に示します。

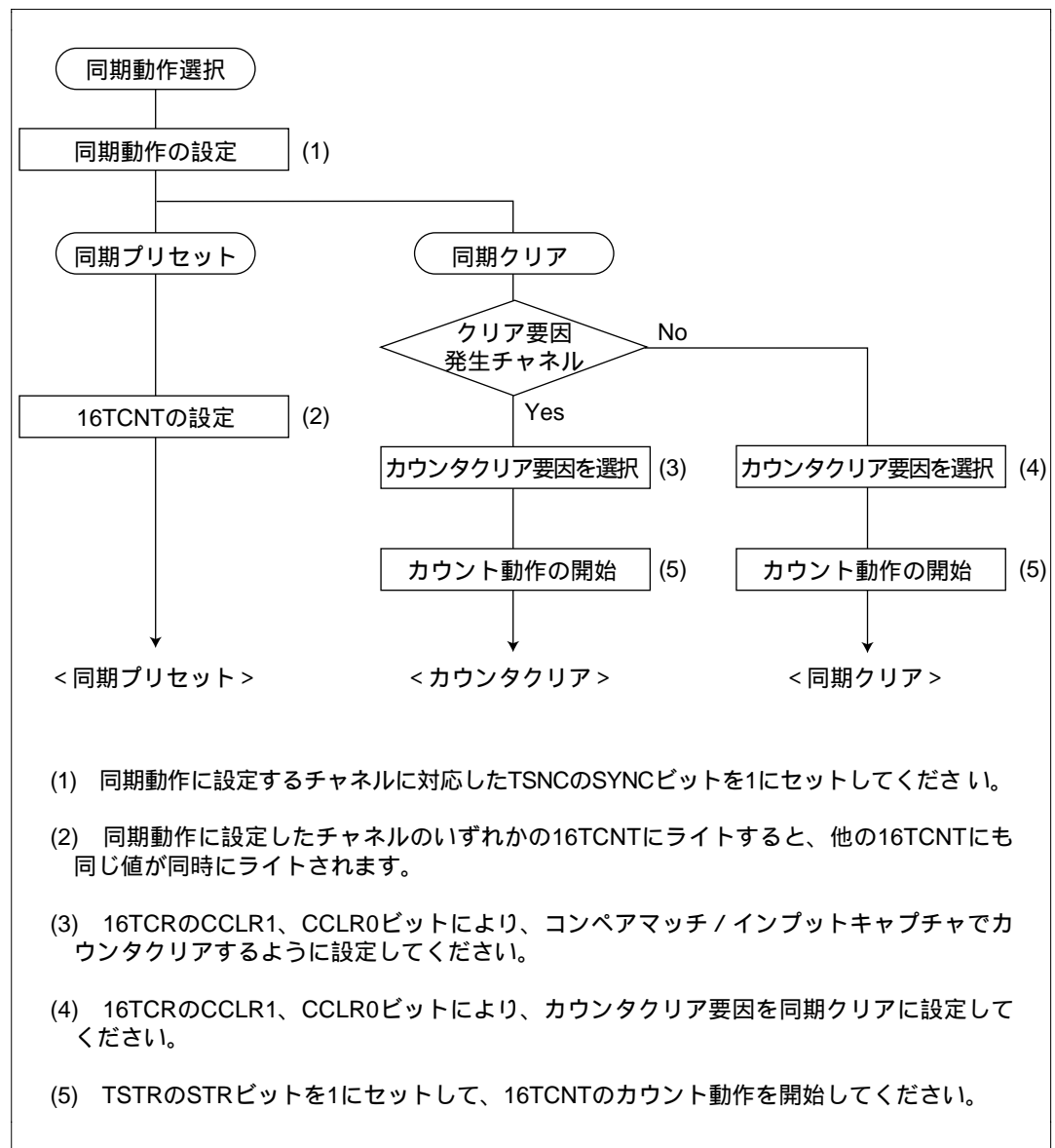


図 9.24 同期モード設定手順例

## (2) 同期動作例

同期動作例を図9.25に示します。

チャンネル0~2を同期動作かつPWMモードに設定し、チャンネル0のカウンタクリア要因をGRB0のコンペアマッチ、またはチャンネル1、2のカウンタクリア要因を同期クリアに設定した場合の例です。

このとき、チャンネル0~2の16TCNTは同期プリセット、GRB0のコンペアマッチによる同期クリア動作を行い、3相のPWM波形をTIOCA<sub>0</sub>、TIOCA<sub>1</sub>、TIOCA<sub>2</sub>端子から出力します。

PWMモードについては「9.4.4 PWMモード」を参照してください。

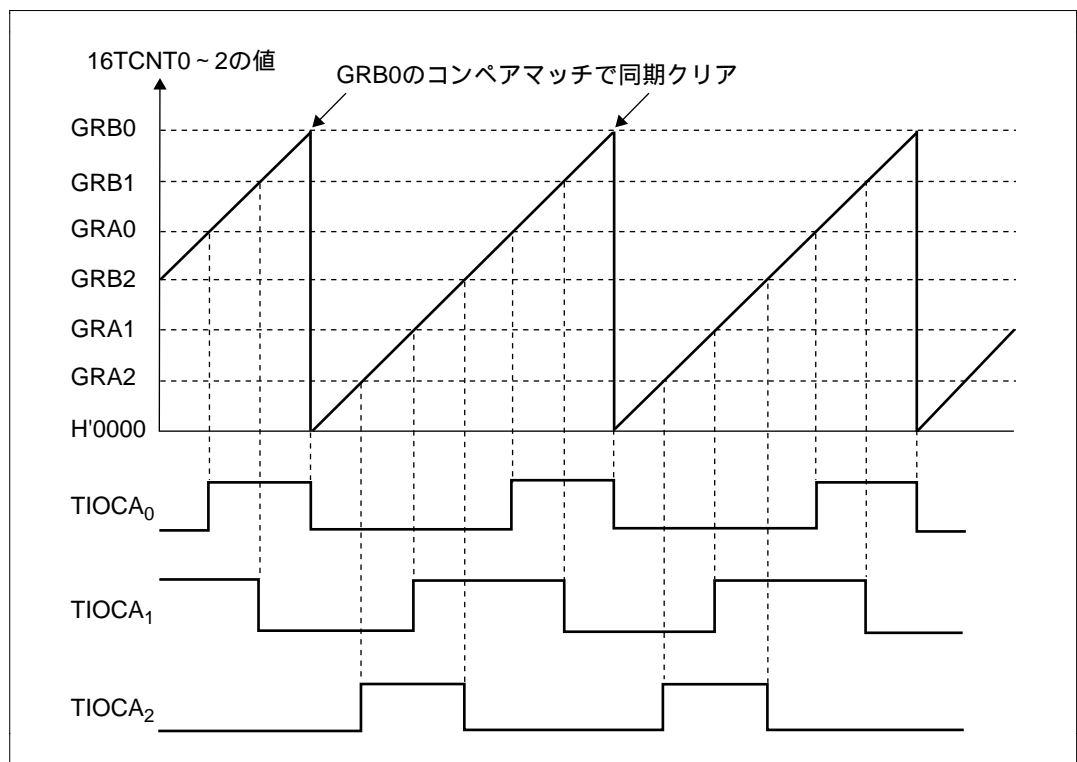


図9.25 同期動作例

#### 9.4.4 PWM モード

PWM モードは GRA と GRB をペアで使用し、TIOCA 出力端子より PWM 波形を出力します。GRA には PWM 波形の 1 出力タイミングを設定し、GRB には PWM 波形の 0 出力タイミングを設定します。

GRA と GRB のいずれかのコンペアマッチを 16TCNT のカウンタクリア要因とすることにより、デューティ 0 ~ 100% の PWM 波形を TIOCA 端子より出力することができます。チャンネル 0 ~ 2 はすべて PWM モードの設定が可能です。

PWM 出力端子とレジスタの対応を表 9.4 に示します。GRA と GRB の設定値が同一の場合、コンペアマッチが発生しても出力値は変化しません。

表 9.4 PWM 出力端子とレジスタの組み合わせ

チャンネル	出力端子	1 出力	0 出力
0	TIOCA <sub>0</sub>	GRA0	GRB0
1	TIOCA <sub>1</sub>	GRA1	GRB1
2	TIOCA <sub>2</sub>	GRA2	GRB2

(1) PWMモードの設定手順例

PWMモードの設定手順例を図9.26に示します。

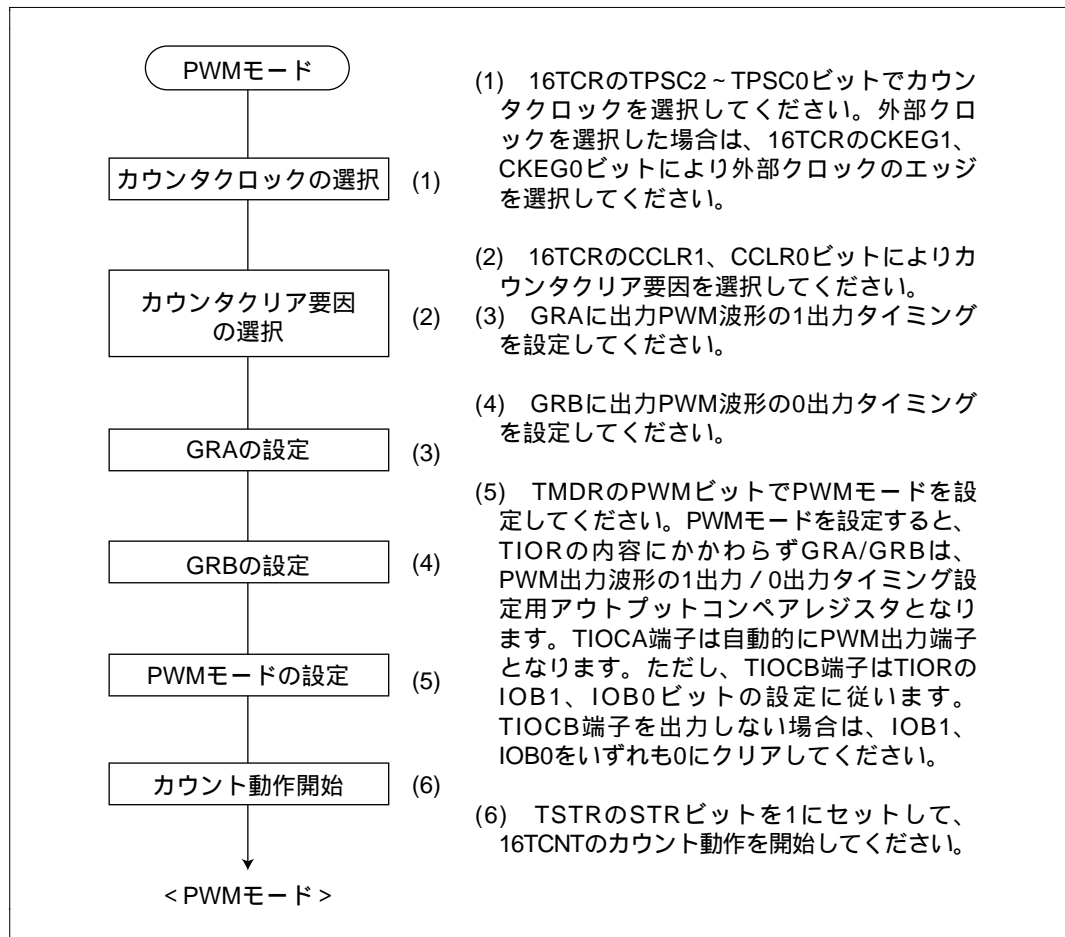


図 9.26 PWMモードの設定手順

## (2) PWM モードの動作例

PWM モードの動作例を図 9.27 に示します。

PWM モードに設定すると TIOCA 端子は出力端子となり、GRA のコンペアマッチで 1 出力、GRB のコンペアマッチで 0 出力となります。

16TCNT のカウンタクリア要因を、GRA、GRB のコンペアマッチとした場合の例です。同期動作またはフリーランニングカウンタ動作も使用できます。

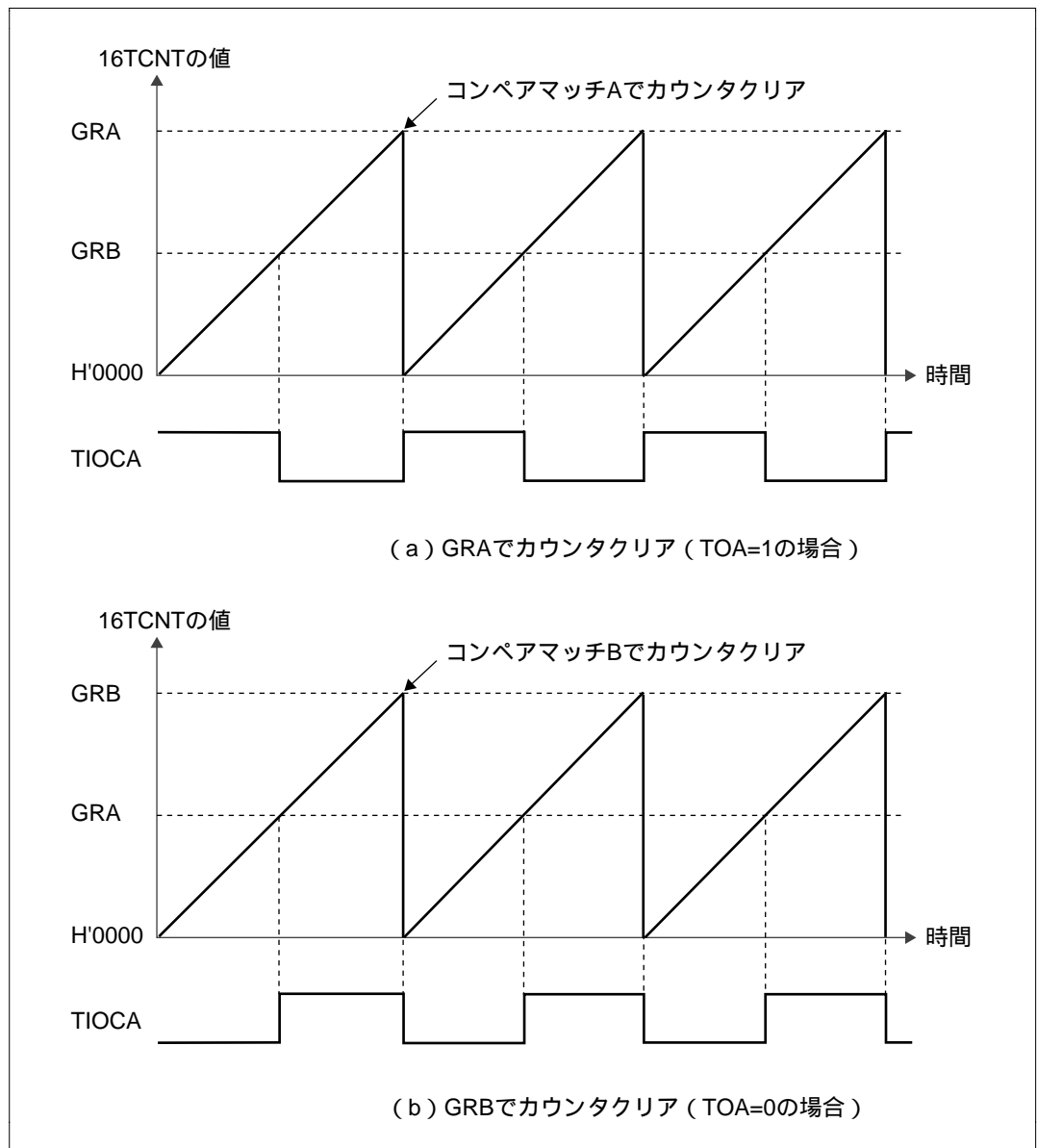


図 9.27 PWM モードの動作例 (1)



PWMモードで、デューティ0%、デューティ100%のPWM波形を出力する例を図9.28に示します。

カウンタクリア要因をGRBのコンペアマッチに設定し、GRAの設定値 > GRBの設定値としたとき、PWM波形はデューティ0%となります。また、カウンタクリア要因をGRAのコンペアマッチに設定し、GRBの設定値 > GRAの設定値としたときPWM波形はデューティ100%となります。

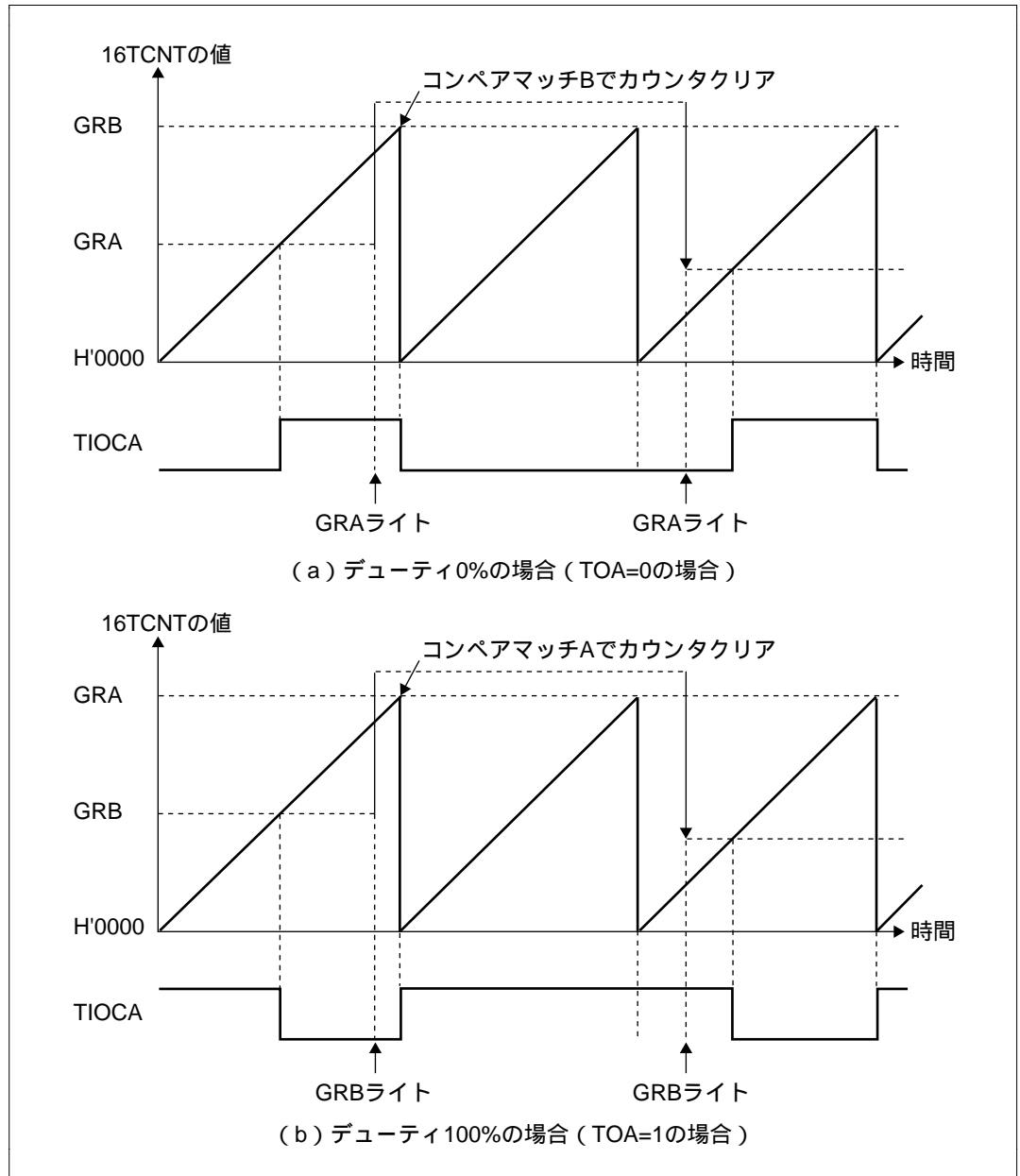


図9.28 PWMモードの動作例(2)

### 9.4.5 位相計数モード

位相計数モードは、2本の外部クロック入力（TCLKA、TCLKB 端子）の位相差を検出し、16TCNT2をアップ/ダウンカウントします。

位相計数モードに設定すると、16TCR2のTPSC2～TPSC0ビット、CKEG1、CKEG0ビットの設定に関わらずTCLKA、TCLKB端子は自動的に外部クロック入力端子として機能し、また16TCNT2はアップ/ダウンカウンタとなります。ただし、16TCR2のCCLR1、CCLR0ビット、TIOR2、TISRA、TISRB、TISRC、TSTRのSTR2ビット、GRA2、GRB2は有効ですので、インプットキャプチャ/アウトプットコンペア機能や割込み要因は使用することができます。

位相計数モードはチャンネル2のみがもつ機能です。

#### (1) 位相計数モードの設定手順例

位相計数モードの設定手順例を図9.29に示します。

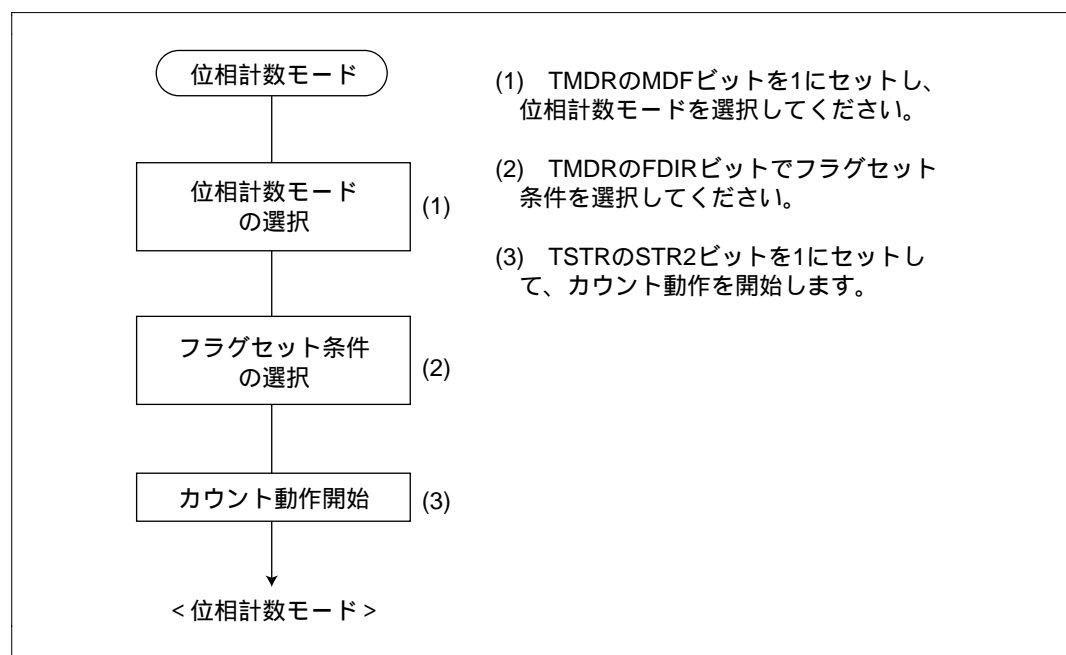


図 9.29 位相計数モードの設定手順例

#### (2) 位相計数モードの動作例

位相計数モードの動作例を図9.30に、16TCNT2のアップ/ダウンカウント条件を表9.5にそれぞれ示します。

位相計数モードでは、TCLKA、TCLKB端子の立上がり（ ） / 立下がり（ ）の両エッジでカウントされます。このとき、TCLKA、TCLKBの位相差およびオーバーラップはそれぞれ1.5ステート以上、パルス幅は2.5ステート以上必要です。

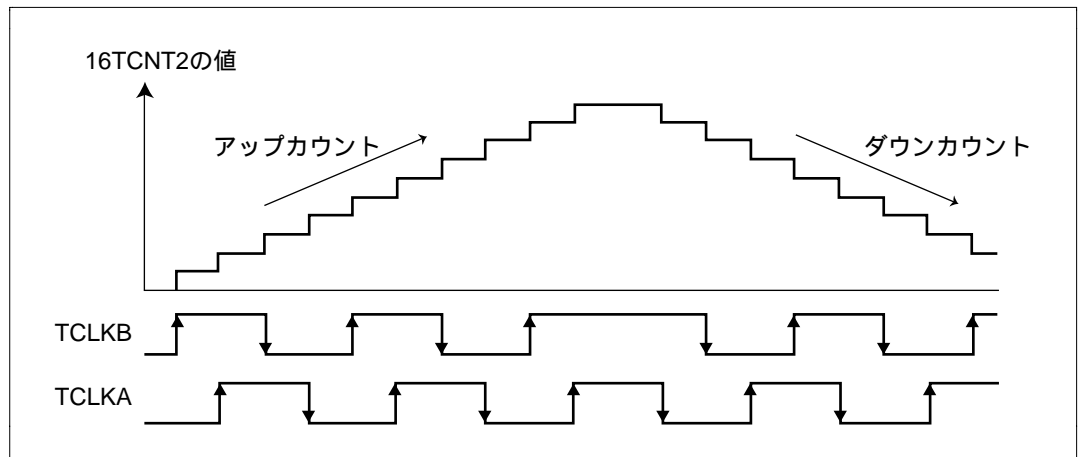


図 9.30 位相計数モードの動作例

表 9.5 アップ/ダウンカウント条件

カウント方向	アップカウント				ダウンカウント			
TCLKB		High		Low	High		Low	
TCLKA	Low		High			Low		High

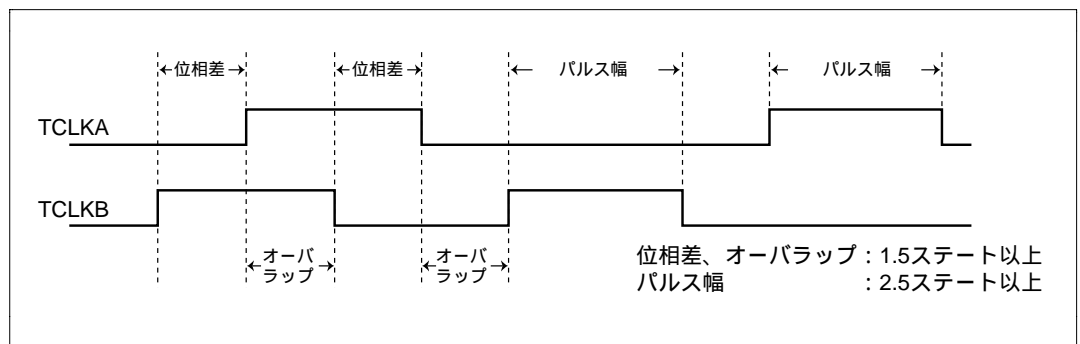


図 9.31 位相計数モード時の位相差、オーバーラップおよびパルス幅

### 9.4.6 16ビットタイマ出力初期値の設定

16ビットタイマ出力は、TOLRの設定により、タイマカウント動作起動時の出力の初期値を任意に設定することができます。

図9.32にTOLRによる出力初期値設定タイミングを示します。

なお、TOLRへのライトは必ず対応するTSTRのビットが0の時に行ってください。

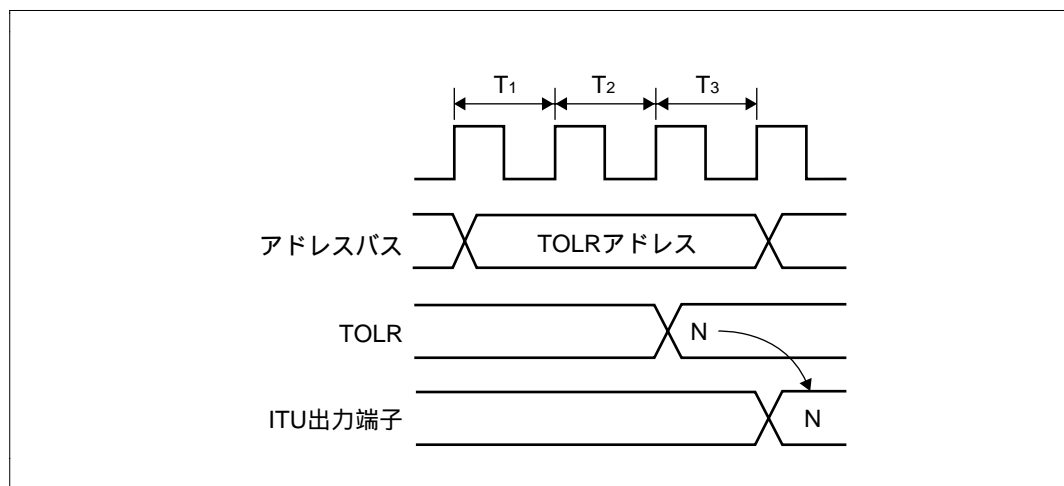


図 9.32 TOLR へのライトによる 16 ビットタイマ出力初期値設定タイミング

## 9.5 割込み

16ビットタイマの割込み要因には、インプットキャプチャ/コンペアマッチ割込み、オーバーフロー割込みの2種類があります。

### 9.5.1 ステータスフラグのセットタイミング

#### (1) コンペアマッチ時の IMFA、IMFB フラグのセットタイミング

IMF フラグは、GR と 16TCNT が一致したときに発生するコンペアマッチ信号により 1 にセットされます。コンペアマッチ信号は、一致した最後のステート (16TCNT が一致したカウント値を更新するタイミング) で発生します。したがって、16TCNT と GR が一致した後、16TCNT 入力クロックが発生するまでコンペアマッチ信号は発生しません。

図 9.33 に IMF フラグのセットタイミングを示します。

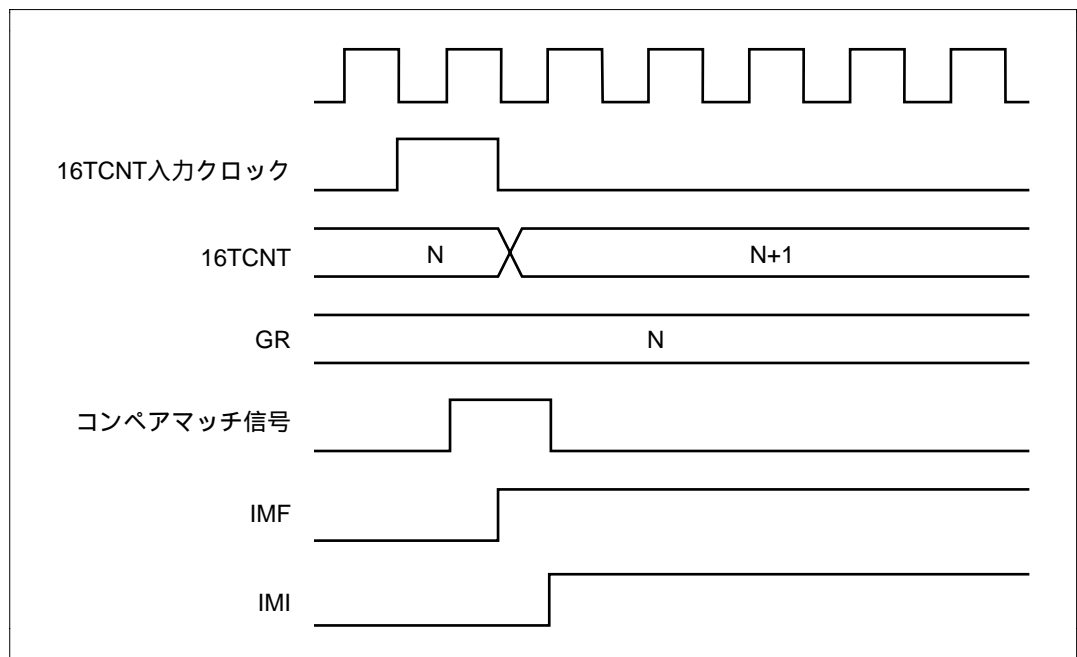


図 9.33 コンペアマッチ時の IMFA、IMFB フラグのセットタイミング

## (2) インพุットキャプチャ時の IMFA、IMFB フラグのセットタイミング

インพุットキャプチャ信号の発生により IMF フラグは1にセットされ、同時に 16TCNT の値が対応する GR に転送されます。

このタイミングを図 9.34 に示します。

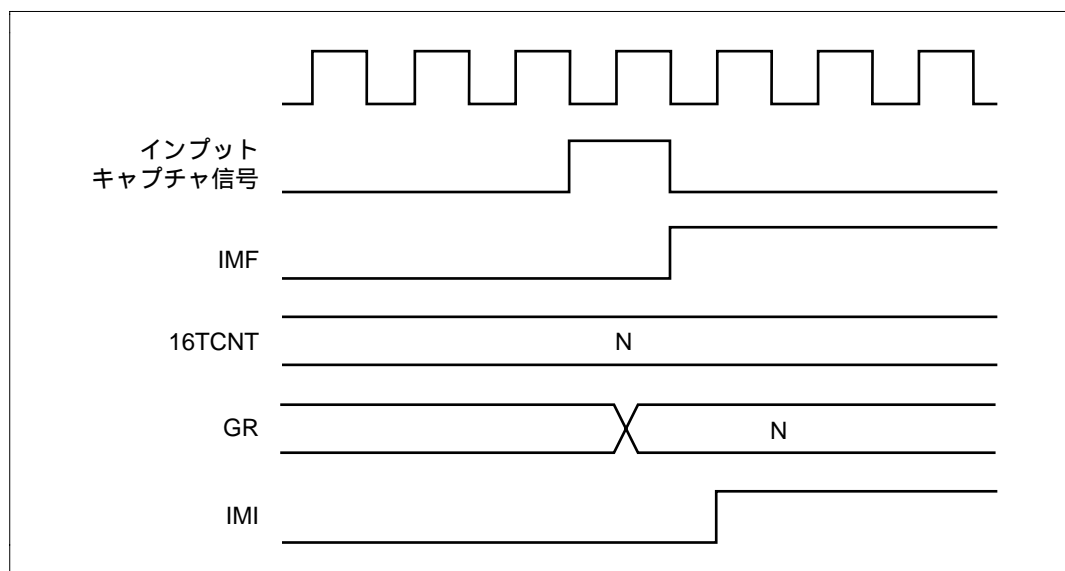


図 9.34 インพุットキャプチャ時の IMFA、IMFB フラグのセットタイミング

## (3) オーバフローフラグ (OVF) のセットタイミング

OVF フラグは、16TCNT がオーバフロー (H'FFFF H'0000) したとき、またはアンダフロー (H'0000 H'FFFF) したときに 1 にセットされます。

このタイミングを図 9.35 に示します。

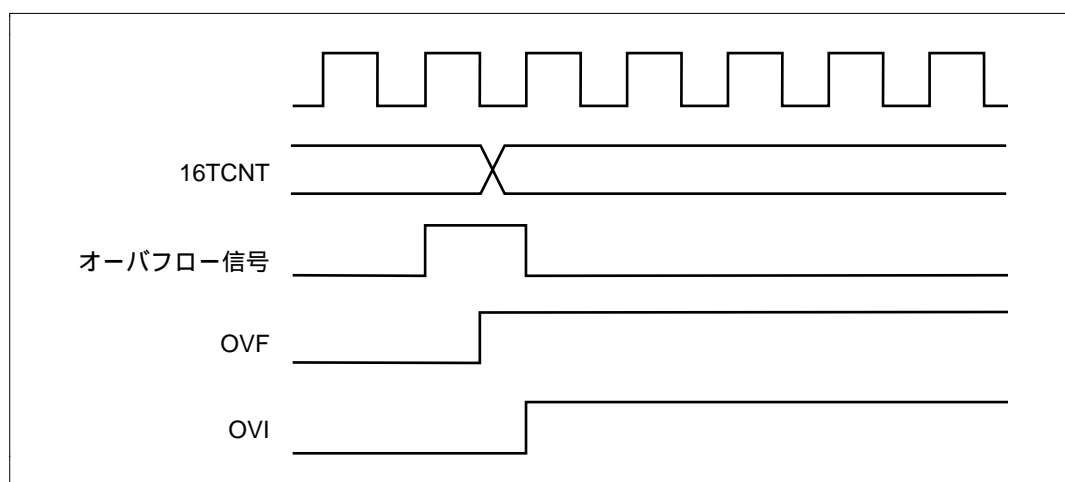


図 9.35 OVF のセットタイミング

### 9.5.2 ステータスフラグのクリアタイミング

ステータスフラグはCPUが1の状態をリードした後0をライトするとクリアされます。  
このタイミングを図9.36に示します。

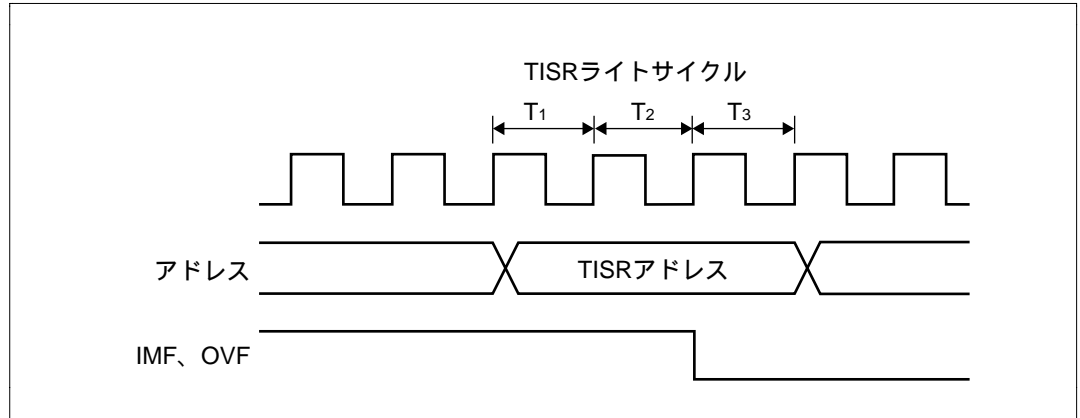


図 9.36 ステータスフラグのクリアタイミング

### 9.5.3 割込み要因とDMAコントローラの起動

16ビットタイマは各チャンネルごとにコンペアマッチ/インプットキャプチャA割込み、コンペアマッチ/インプットキャプチャB割込み、およびオーバーフロー割込みをもっています。これら3種類の割込み計9本の割込みは、それぞれ独立のベクタアドレスが割り付けられています。割込み要求フラグが1にセットされているとき、当該割込みが要求されます。

チャンネル間の優先順位は、IPRAにより変更可能です。詳細は「第5章 割込みコントローラ」を参照してください。

チャンネル0~2のコンペアマッチ/インプットキャプチャA割込みはDMACの起動要因とすることができます。このときは、CPUへは当該割込みは要求されません。

16ビットタイマの割込み要因を表9.6に示します。

表9.6 16ビットタイマ割込み要因

チャンネル	割込み要因	内 容	DMACの起動	優先順位*
0	IMIA0	コンペアマッチ/インプットキャプチャA0	可	高 ▲ ↑ ↓ 低
	IMIB0	コンペアマッチ/インプットキャプチャB0	不可	
	OVI0	オーバーフロー0	不可	
1	IMIA1	コンペアマッチ/インプットキャプチャA1	可	
	IMIB1	コンペアマッチ/インプットキャプチャB1	不可	
	OVI1	オーバーフロー1	不可	
2	IMIA2	コンペアマッチ/インプットキャプチャA2	可	
	IMIB2	コンペアマッチ/インプットキャプチャB2	不可	
	OVI2	オーバーフロー2	不可	

【注】\* リセット直後の初期状態について示しています。チャンネル間の優先順位はIPRAにより変更可能です。



## 9.6 使用上の注意

16ビットタイマの動作中、次のような競合や動作がおこりますので、注意してください。

### (1) 16TCNTのライトとクリアの競合

16TCNTのライトサイクル中の $T_3$ 状態で、カウンタクリア信号が発生すると、16TCNTへの書込みサイクルは行われず16TCNTのクリアが優先されます。

このタイミングを図9.37に示します。

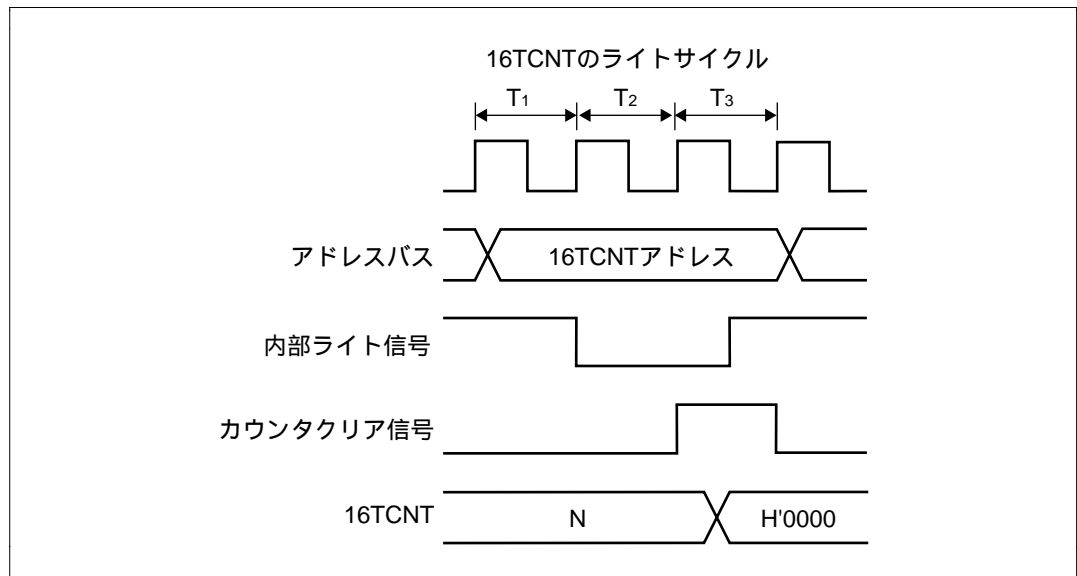


図 9.37 16TCNT のライトとクリアの競合

## (2) 16TCNTのワードライトとカウントアップの競合

16TCNTのワードライトサイクル中の $T_3$ 状態でカウントアップが発生しても、カウントアップされずカウンタライトが優先されます。

このタイミングを図9.38に示します。

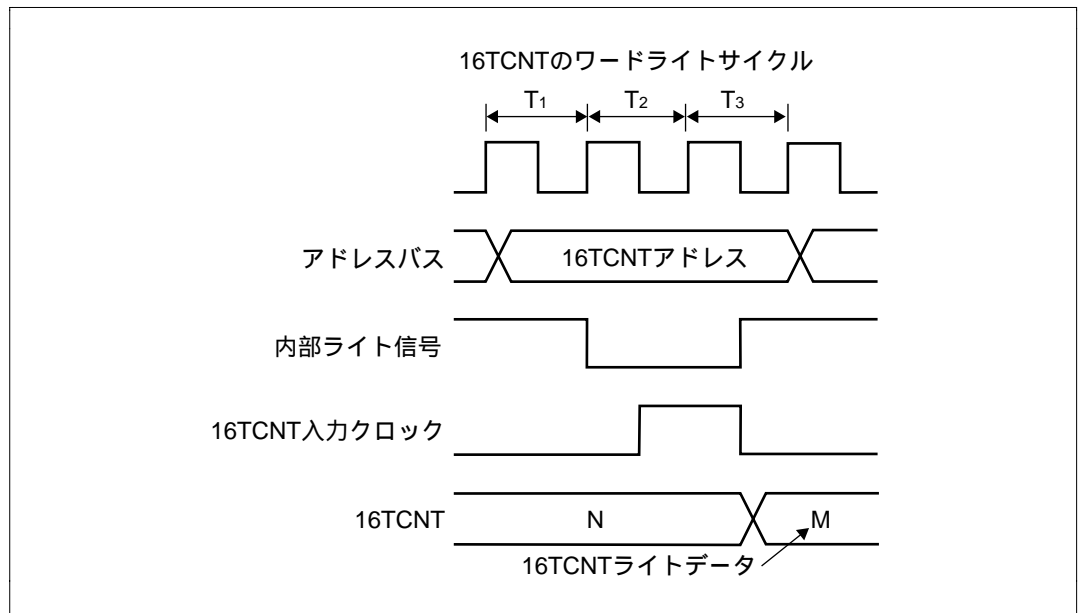


図9.38 16TCNTのワードライトとカウントアップの競合

(3) 16TCNTのバイトライトとカウントアップの競合

16TCNTのバイトライトサイクル中の $T_2$ ステートまたは $T_3$ ステートでカウントアップが発生しても、ライトを行った側のバイトデータはカウントアップされず、カウンタライトが優先されます。ライトを行わなかった側のバイトデータもカウントアップされずライトする前の内容となります。

このタイミングを図9.39に示します。

16TCNTHのバイトライトサイクル中の $T_2$ ステートでカウントアップが発生した場合の例です。

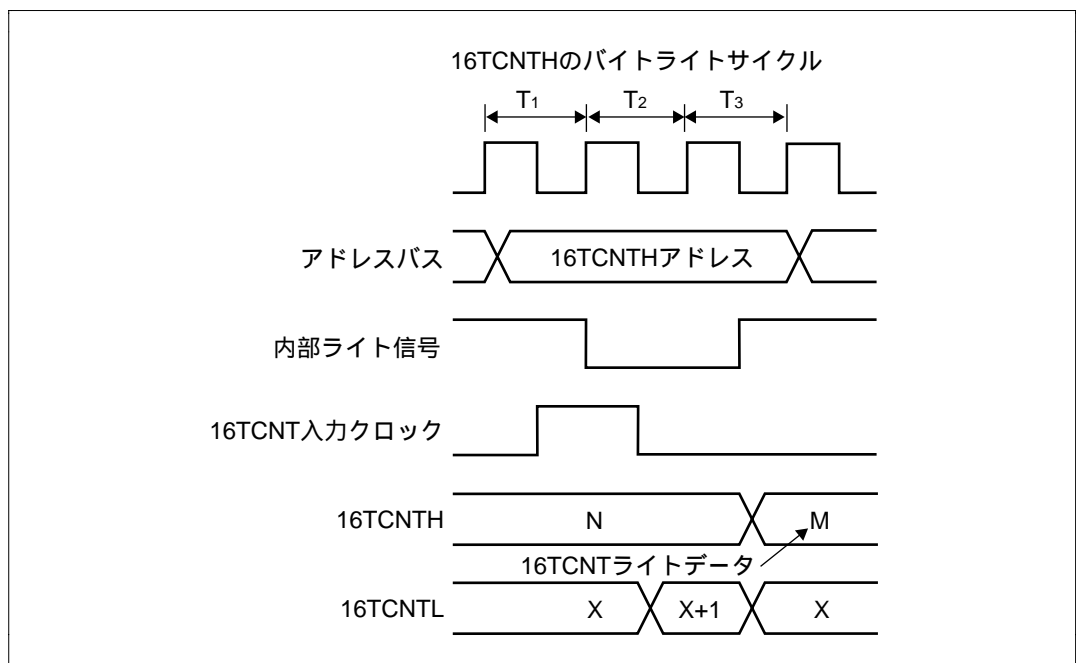


図 9.39 16TCNTのバイトライトとカウントアップの競合

## (4) GR のライトとコンペアマッチの競合

GR のライトサイクル中の  $T_3$  ステートでコンペアマッチが発生しても、GR のライトが優先され、コンペアマッチ信号は禁止されます。

このタイミングを図 9.40 に示します。

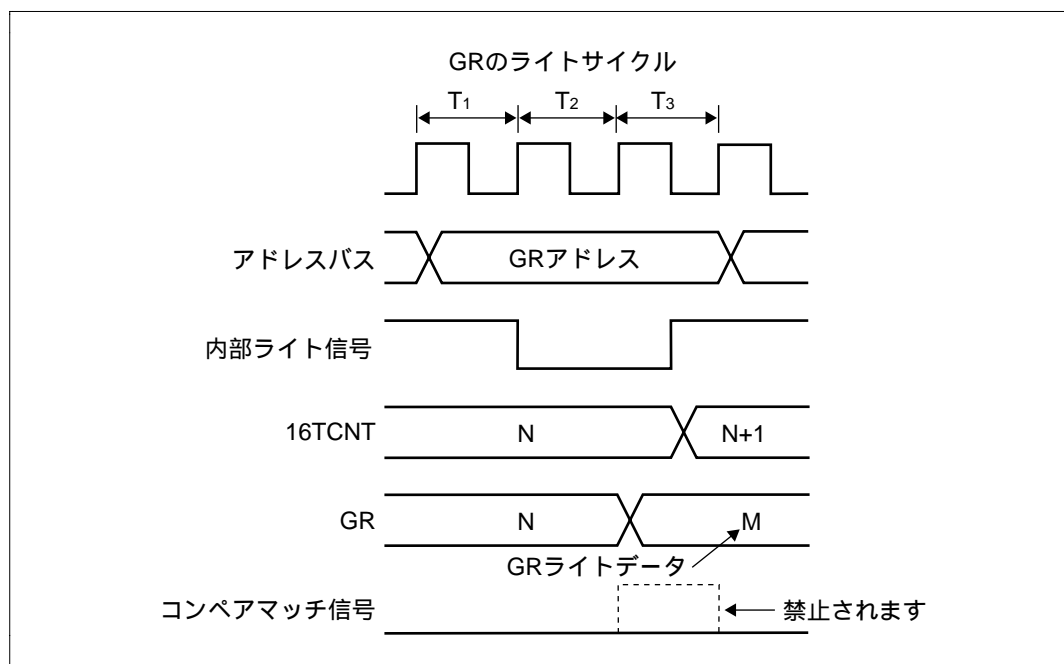


図 9.40 GR のライトとコンペアマッチの競合

(5) 16TCNTのライトとオーバーフロー/アンダフローとの競合

16TCNTのライトサイクル中の $T_3$ 状態でオーバーフローが発生した場合、カウントアップされずカウンタライトが優先されます。このときOVFフラグは1にセットされます。アンダフローの場合も同様です。

このタイミングを図9.41に示します。

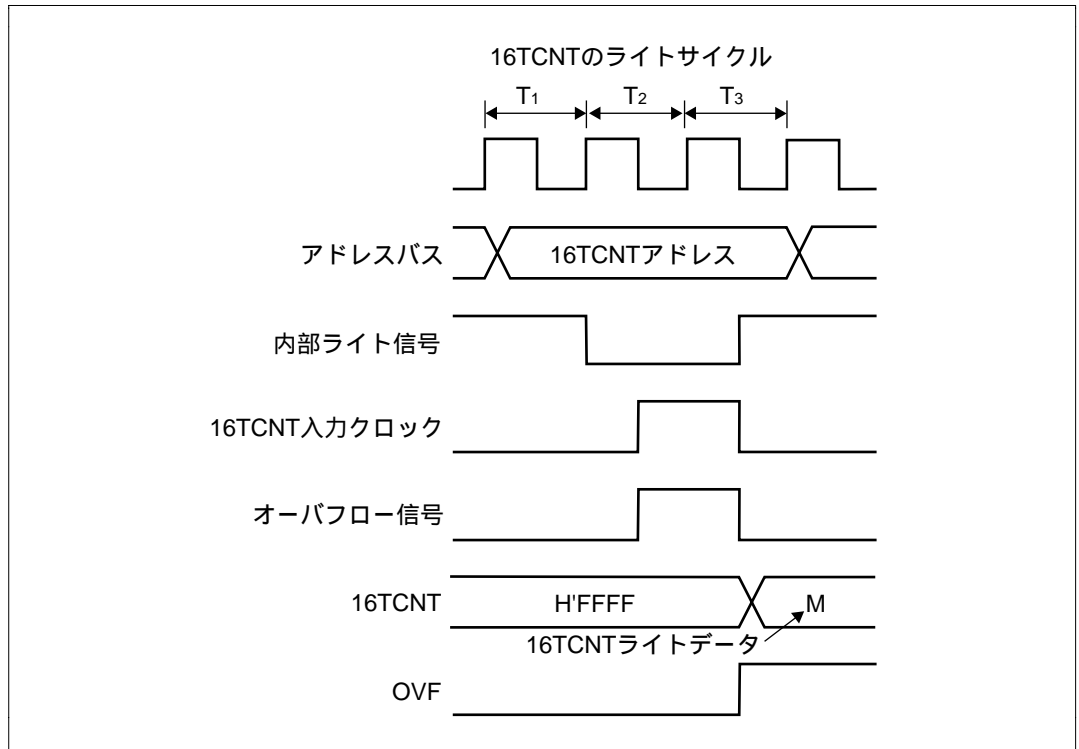


図 9.41 16TCNTのライトとオーバーフローの競合

## (6) GR のリードとインプットキャプチャの競合

GRのリードサイクル中のT3ステートで、インプットキャプチャ信号が発生すると、リードされるデータはインプットキャプチャ転送前のデータです。

このタイミングを図9.42に示します。

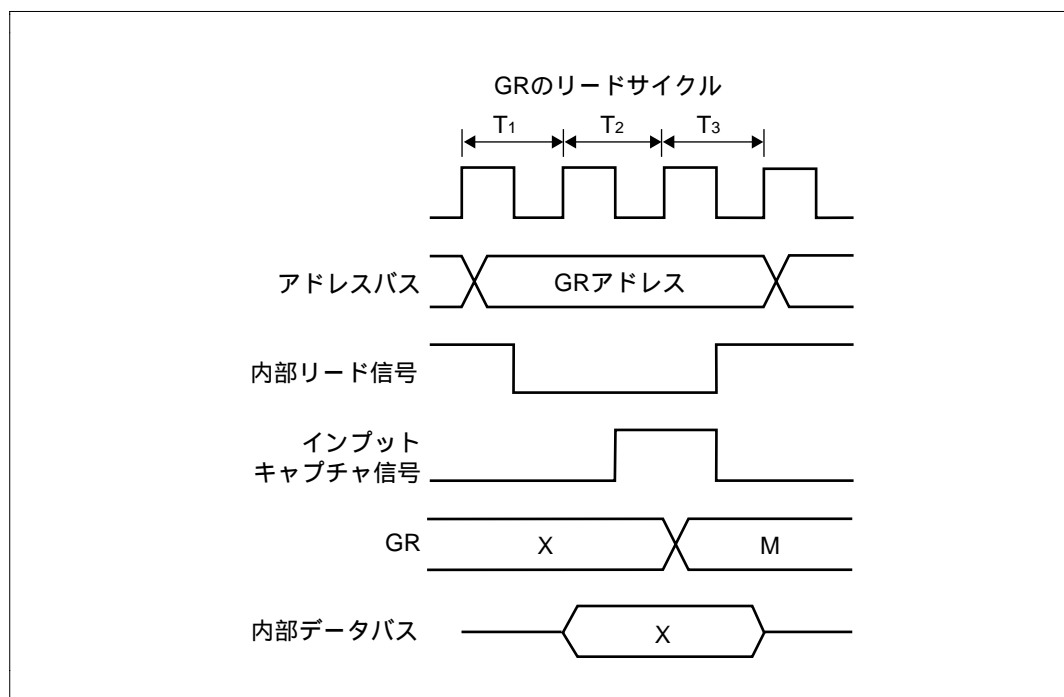


図 9.42 GR のリードとインプットキャプチャの競合

## (7) インพุットキャプチャによるカウンタクリアとカウントアップの競合

インพุットキャプチャ信号とカウントアップ信号が同時に発生すると、カウントアップされずインพุットキャプチャによるカウンタクリアが優先されます。GR にはカウンタクリア前の 16TCNT の内容が転送されます。

このタイミングを図 9.43 に示します。

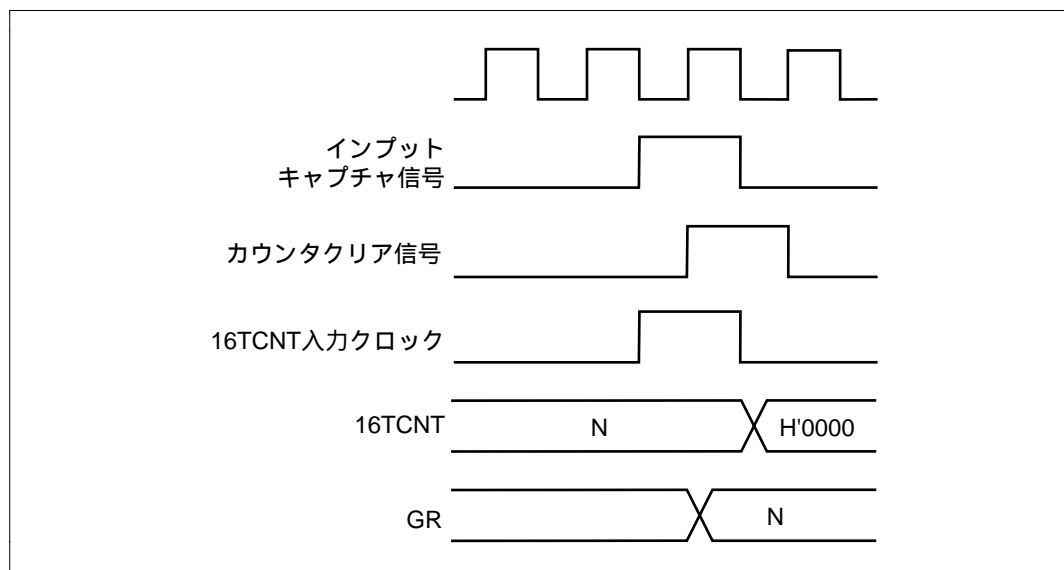


図 9.43 インพุットキャプチャによるカウンタクリアとカウントアップの競合

## (8) GRのライトと入力キャプチャの競合

GRのライトサイクル中の $T_3$ 状態で、入力キャプチャ信号が発生すると、GRへの書込みは行われず、入力キャプチャが優先されます。

このタイミングを図9.44に示します。

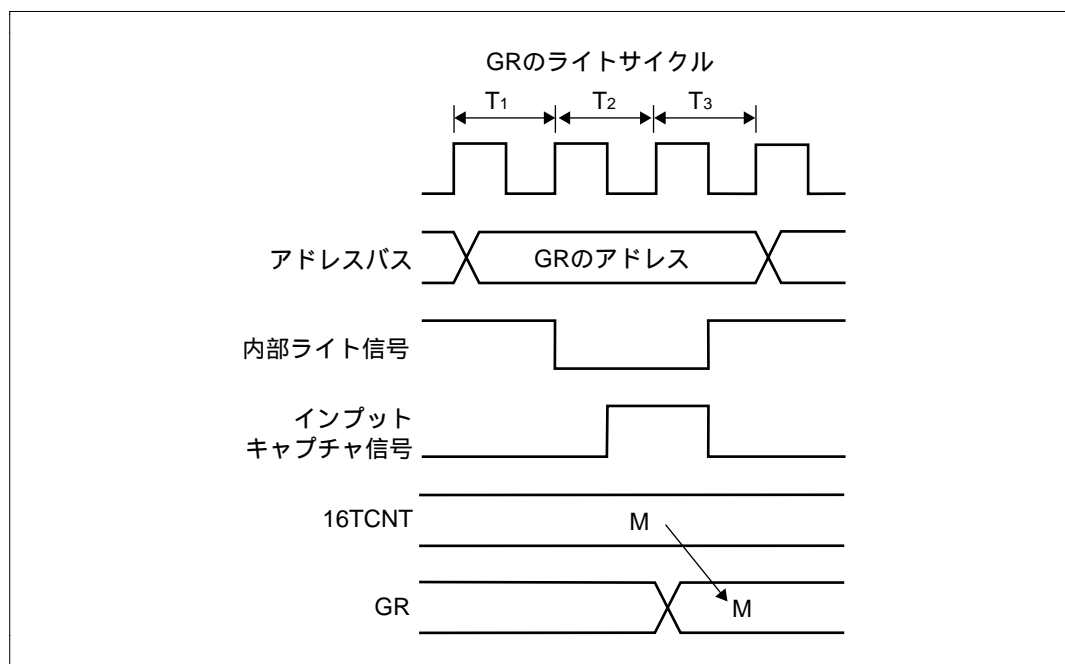


図 9.44 GRのライトと入力キャプチャの競合

## (9) 周期設定上の注意事項

コンペアマッチによるカウンタクリアを設定した場合、16TCNTはGRの値と一致した最後のステート(16TCNTが一致したカウント値を更新するタイミング)でクリアされず。このため、実際のカウンタ周波数は次の式のようにになります。

$$f = \frac{f_{clk}}{(N+1)}$$

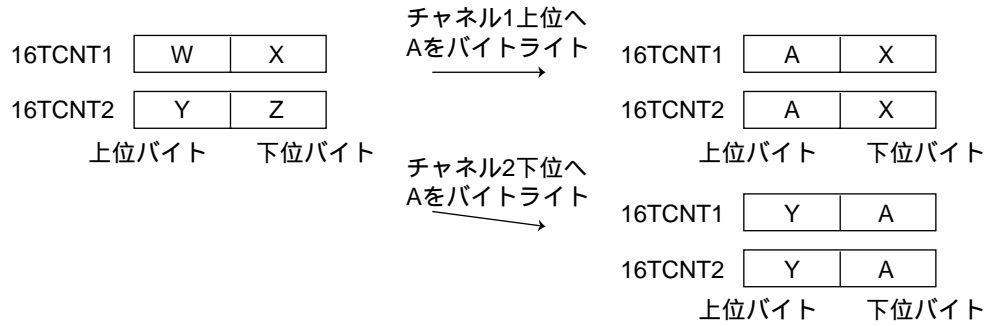
( $f$ : カウンタ周波数、 $f_{clk}$ : 動作周波数、 $N$ : GRの設定値)



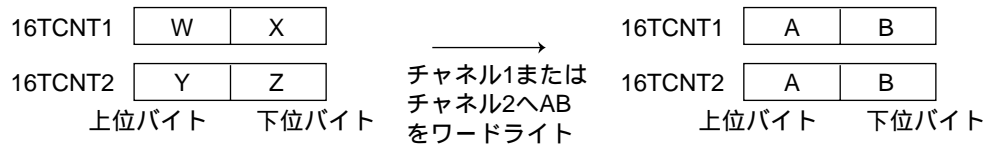
(10) 同期動作時のライト動作に関する注意事項

同期動作を設定した状態で、16TCNTのバイトライトを行った場合、同期しているすべてのカウンタはアドレスで指定した16TCNTと、16ビットすべて同じ値となります。

(例) チャンネル1、2を同期モードで指定した場合  
 ・チャンネル1 / チャンネル2へのバイトライト



・チャンネル1 / チャンネル2へのワードライト



(11) 16ビットタイマの動作モード一覧

表 9.7 (a) 16ビットタイマの動作モード (チャンネル0)

動作モード	レジスタ設定									
	TNSC			TMDR			TIOR0		16TCR0	
	同期動作	MDF	FDIR	PWM	IOA	IOB	クリア選択	クロック選択		
同期プリセット	SYNC0=1	—	—	○	○	○	○	○	○	
PWMモード	○	—	—	PWM0=1	—	○*	○	○	○	
アウトプット コンペアA機能	○	—	—	PWM0=0	IOA2=0 他任意	○	○	○	○	
アウトプット コンペアB機能	○	—	—	○	○	IOB2=0 他任意	○	○	○	
インプット キャプチャA機能	○	—	—	PWM0=0	IOA2=1 他任意	○	○	○	○	
インプット キャプチャB機能	○	—	—	PWM0=0	○	IOB2=1 他任意	○	○	○	
カウンタ クリア機能	○	—	—	○	○	○	CCLR1=0 CCLR0=1	○	○	
同期クリア	SYNC0=1	—	—	○	○	○	CCLR1=1 CCLR0=0	○	○	

< 記号説明 >

○ : 設定可能 (有効) です。 — : 設定は当該動作モードに影響しません。

【注】 \* PWMモードでは、インプットキャプチャ機能は使用できません。また、コンペアマッチAとコンペアマッチBが同時に発生した場合、コンペアマッチ信号は禁止されます。

表 9.7 (b) 16ビットタイマの動作モード(チャンネル1)

動作モード	レジスタ設定										
	TSNC			TMDR			TIOR1			16TCR1	
	同期動作	MDF	FDIR	PWM	IOA	IOB	クリア選択	クロック選択			
同期プリセット	SYNC1=1	—	—	○	○	○	○	○	○	○	○
PWMモード	○	—	—	PWM1=1	—	○*	○	○	○	○	○
アウトプットコンペアA機能	○	—	—	PWM1=0	IOA2=0 他任意	○	○	○	○	○	○
アウトプットコンペアB機能	○	—	—	○	○	IOB2=0 他任意	○	○	○	○	○
インプットキャプチャA機能	○	—	—	PWM1=0	IOA2=1 他任意	○	○	○	○	○	○
インプットキャプチャB機能	○	—	—	PWM1=0	○	IOB2=1 他任意	○	○	○	○	○
コンペアマッチ/インプットキャプチャAでクリア	○	—	—	○	○	○	○	○	CCLR1=0 CCLR0=1	○	○
コンペアマッチ/インプットキャプチャBでクリア	○	—	—	○	○	○	○	○	CCLR1=1 CCLR0=0	○	○
同期クリア機能	SYNC1=1	—	—	○	○	○	○	○	CCLR1=1 CCLR0=1	○	○

< 記号説明 >

○ : 設定可能(有効)です。 — : 設定は当該動作モードに影響しません。

【注】\* PWMモードでは、インプットキャプチャ機能は使用できません。また、コンペアマッチAとコンペアマッチBが同時に発生した場合、コンペアマッチ信号は禁止されます。

表 9.7 (c) 16ビットタイマの動作モード (チャンネル2)

動作モード	レジスタ設定									
	TSNC		TMDR				TIOR2		16TCR2	
	同期動作	MDF	FDIR	PWM	IOA	IOB	クリア選択	クロック選択		
同期プリセット	SYNC2=1	○	—	○	○	○	○	○	○	○
PWMモード	○	○	—	PWM2=1	—	○*	○	○	○	○
アウトプットコンペアA機能	○	○	—	PWM2=0	IOA2=0 他任意	○	○	○	○	○
アウトプットコンペアB機能	○	○	—	○	○	IOB2=0 他任意	○	○	○	○
インプットキャプチャA機能	○	○	—	PWM2=0	IOA2=1 他任意	○	○	○	○	○
インプットキャプチャB機能	○	○	—	PWM2=0	○	IOB2=1 他任意	○	○	○	○
カウンタクリア機能	○	○	—	○	○	○	○	CCLR1=0 CCLR0=1	○	○
	○	○	—	○	○	○	○	CCLR1=1 CCLR0=0	○	○
	同期クリア	SYNC2=1	○	—	○	○	○	CCLR1=1 CCLR0=1	○	○
位相計数モード	○	MDF=1	○	○	○	○	○	○	○	—

< 記号説明 >

○ : 設定可能 (有効) です。 — : 設定は当該動作モードに影響しません。

【注】 \* PWMモードでは、インプットキャプチャ機能は使用できません。また、コンペアマッチAとコンペアマッチBが同時に発生した場合、コンペアマッチ信号は禁止されます。



---

# 10. 8ビットタイマ

---

## 第10章 目次

10.1	概要	429
10.1.1	特長	429
10.1.2	ブロック図	430
10.1.3	端子構成	431
10.1.4	レジスタ構成	432
10.2	各レジスタの説明	433
10.2.1	タイマカウンタ (8TCNT)	433
10.2.2	タイムコンスタントレジスタ A (TCORA)	434
10.2.3	タイムコンスタントレジスタ B (TCORB)	435
10.2.4	タイマコントロールレジスタ (8TCR)	436
10.2.5	タイマコントロール/ステータスレジスタ (8TCSR)	439
10.3	CPUとのインタフェース	444
10.3.1	8ビットレジスタ	444
10.4	動作説明	446
10.4.1	8TCNTのカウントタイミング	446
10.4.2	コンペアマッチタイミング	447
10.4.3	インプットキャプチャ信号タイミング	449
10.4.4	ステータスフラグのセットタイミング	450
10.4.5	カスケード接続時の動作	451
10.4.6	インプットキャプチャの設定	454
10.5	割込み	456
10.5.1	割込み要因	456
10.5.2	A/D変換器の起動	456
10.6	8ビットタイマの使用例	457
10.7	使用上の注意	458
10.7.1	8TCNTのライトとクリアの競合	458
10.7.2	8TCNTのライトとカウントアップの競合	459

10.7.3	TCOR のライトとコンペアマッチの競合 .....	460
10.7.4	TCOR のリードとインプットキャプチャの競合 .....	461
10.7.5	インプットキャプチャによるカウンタクリアとカウントアップの競合 .....	461
10.7.6	TCOR のライトとインプットキャプチャの競合 .....	462
10.7.7	16 ビットカウントモード (カスケード接続時) の 8TCNT のバイトライトと カウントアップの競合 .....	463
10.7.8	コンペアマッチ A、B の競合 .....	463
10.7.9	内部クロックの切り替えと 8TCNT の動作 .....	464

## 10.1 概要

本 LSI は、8ビットのカウンタをベースにした4チャンネルの8ビットタイマ (TMR0、TMR1、TMR2、TMR3) を内蔵しています。4チャンネルの8ビットタイマには、それぞれタイマカウンタ (8TCNT) のほかに8ビットのタイムコンスタントレジスタ A、B (TCORA、TCORB) があり、8TCNT と TCOR の値の一致によるコンペアマッチ信号により、任意のデューティ比のパルス出力など、多機能タイマとして種々の応用が可能です。

### 10.1.1 特長

4種類のカウンタ入力クロックを選択可能

- ・3種類の内部クロック ( /8、 /64、 /8192) と、外部クロックのうちから選択できます (外部イベントのカウントが可能)。

カウンタのクリア指定が可能

- ・コンペアマッチ A、B、またはインプットキャプチャ B のうちから選択できます。

2つのコンペアマッチ信号の組み合わせでタイマ出力を制御

- ・独立に動作可能な2つのコンペアマッチ信号の組み合わせによって、任意のデューティのパルス出力や PWM 出力などの種々の応用が可能です。

コンペアマッチによる A/D 変換器の起動が可能です。

2チャンネルのカスケード接続が可能

- ・チャンネル0を上位、チャンネル1を下位とする16ビットタイマとして動作可能です (16ビットカウントモード)。
- ・チャンネル2を上位、チャンネル3を下位とする16ビットタイマとして動作可能です (16ビットカウントモード)。
- ・チャンネル1はチャンネル0のコンペアマッチをカウント可能です (コンペアマッチカウントモード)。
- ・チャンネル3はチャンネル2のコンペアマッチをカウント可能です (コンペアマッチカウントモード)。

インプットキャプチャ機能を設定可能

- ・8ビット / 16ビットのインプットキャプチャ動作が可能です。

12種類の割り込み要因

- ・コンペアマッチ×4要因、コンペアマッチ / インプットキャプチャ×4要因、オーバフロー×4要因の計12要因があります。

コンペアマッチ割り込みのうち2要因とコンペアマッチ / インプットキャプチャ兼用割り込みのうち2要因は、独立した割り込みベクタを持っています。残りのコンペアマッチ割り込み、コンペアマッチ / インプットキャプチャ兼用割り込み、オーバフロー割り込みは、2要因で1つの割り込みベクタを持っています。



### 10.1.2 ブロック図

8ビットタイマのブロック図を図10.1に示します。8ビットタイマは2チャンネルのグループ0(チャンネル0、チャンネル1)およびグループ1(チャンネル2、チャンネル3)に分割されています。

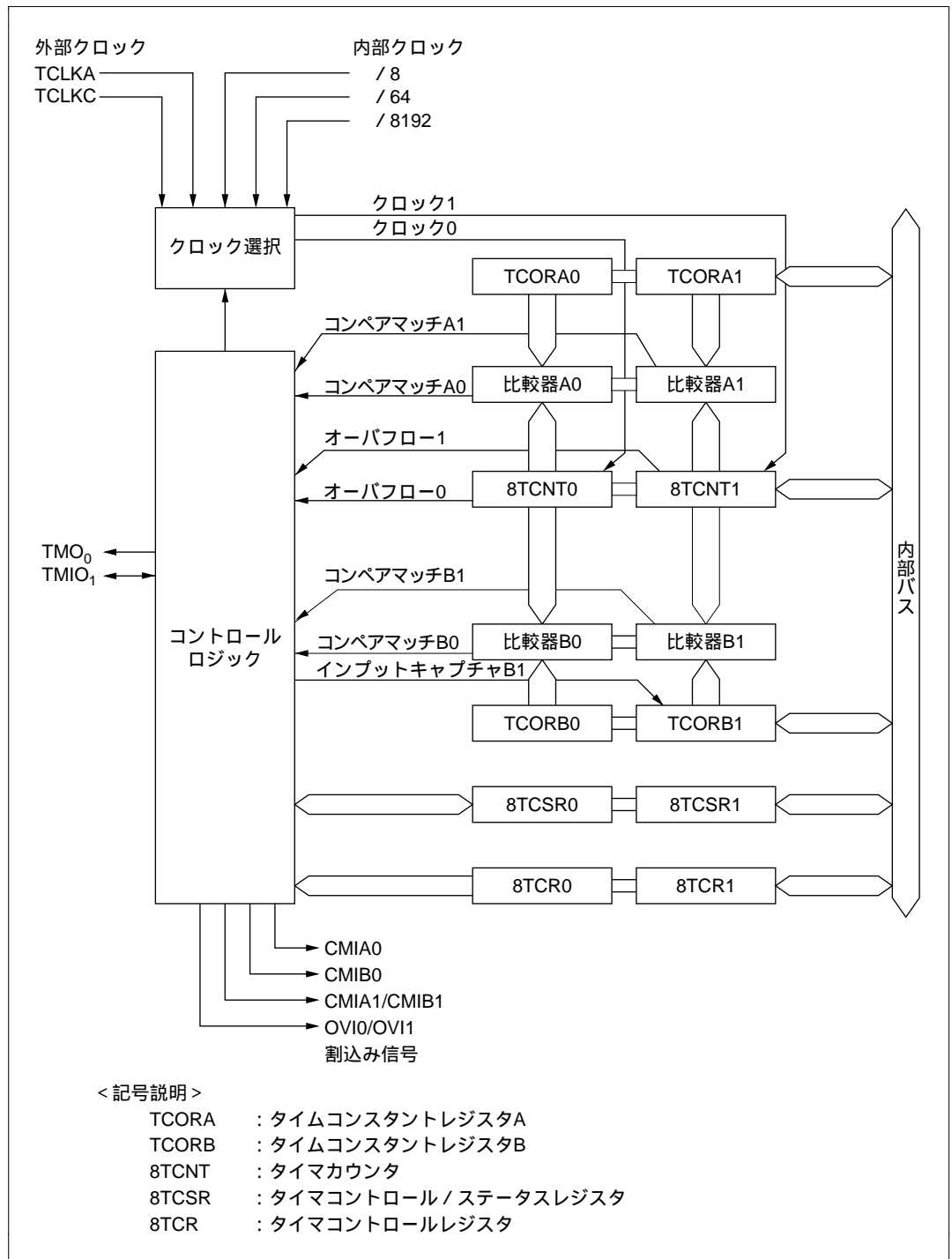


図10.1 8ビットタイマのブロック図(2チャンネル分:グループ0)

### 10.1.3 端子構成

8ビットタイマの入出力端子を表 10.1 に示します。

表 10.1 端子構成

グループ	チャンネル	名称	略称	入出力	機能
0	0	タイマ出力端子	TMO <sub>0</sub>	出力	コンペアマッチ出力
		タイマクロック入力端子	TCLKC	入力	カウンタ外部クロック入力
	1	タイマ入出力端子	TMIO <sub>1</sub>	入出力	コンペアマッチ出力 / インブットキャプチャ入力
		タイマクロック入力端子	TCLKA	入力	カウンタ外部クロック入力
1	2	タイマ出力端子	TMO <sub>2</sub>	出力	コンペアマッチ出力
		タイマクロック入力端子	TCLKD	入力	カウンタ外部クロック入力
	3	タイマ入出力端子	TMIO <sub>3</sub>	入出力	コンペアマッチ出力 / インブットキャプチャ入力
		タイマクロック入力端子	TCLKB	入力	カウンタ外部クロック入力

## 10.1.4 レジスタ構成

8ビットタイマのレジスタ構成を表 10.2 に示します。

表 10.2 レジスタ構成

チャンネル	アドレス*1	名 称	略称	R/W	初期値
0	H'FFF80	タイマコントロールレジスタ 0	8TCR0	R/W	H'00
	H'FFF82	タイマコントロール/ステータスレジスタ 0	8TCSR0	R/(W)*2	H'00
	H'FFF84	タイムコンスタントレジスタ A 0	TCORA0	R/W	H'FF
	H'FFF86	タイムコンスタントレジスタ B 0	TCORB0	R/W	H'FF
	H'FFF88	タイマカウンタ 0	8TCNT0	R/W	H'00
1	H'FFF81	タイマコントロールレジスタ 1	8TCR1	R/W	H'00
	H'FFF83	タイマコントロール/ステータスレジスタ 1	8TCSR1	R/(W)*2	H'00
	H'FFF85	タイムコンスタントレジスタ A 1	TCORA1	R/W	H'FF
	H'FFF87	タイムコンスタントレジスタ B1	TCORB1	R/W	H'FF
	H'FFF89	タイマカウンタ 1	8TCNT1	R/W	H'00
2	H'FFF90	タイマコントロールレジスタ 2	8TCR2	R/W	H'00
	H'FFF92	タイマコントロール/ステータスレジスタ 2	8TCSR2	R/(W)*2	H'10
	H'FFF94	タイムコンスタントレジスタ A 2	TCORA2	R/W	H'FF
	H'FFF96	タイムコンスタントレジスタ B 2	TCORB2	R/W	H'FF
	H'FFF98	タイマカウンタ 2	8TCNT2	R/W	H'00
3	H'FFF91	タイマコントロールレジスタ 3	8TCR3	R/W	H'00
	H'FFF93	タイマコントロール/ステータスレジスタ 3	8TCSR3	R/(W)*2	H'00
	H'FFF95	タイムコンスタントレジスタ A 3	TCORA3	R/W	H'FF
	H'FFF97	タイムコンスタントレジスタ B3	TCORB3	R/W	H'FF
	H'FFF99	タイマカウンタ 3	8TCNT3	R/W	H'00

【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。

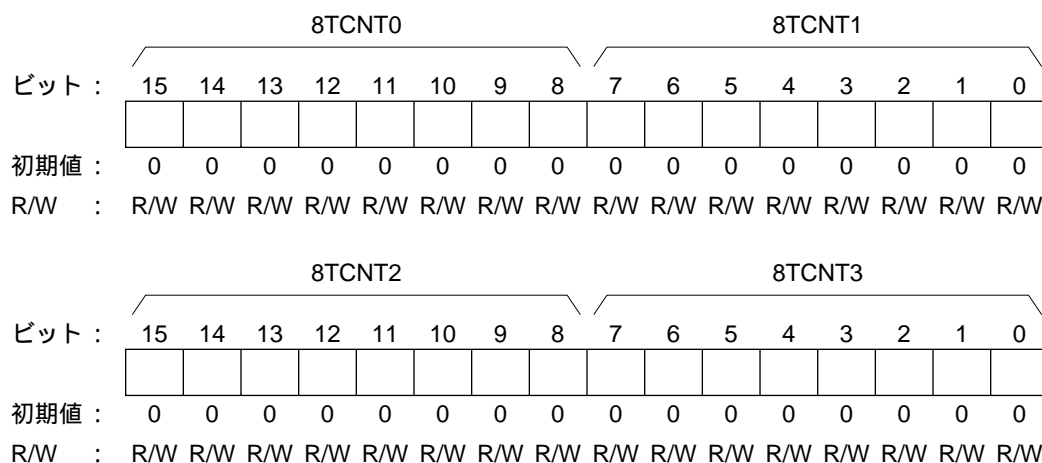
\*2 ビット 7~5 は、フラグをクリアするための 0 ライトのみ可能です。

チャンネル 0 とチャンネル 1 の対応するレジスタは、チャンネル 0 を上位、チャンネル 1 を下位とする 16 ビットレジスタとして、ワードアクセスすることができます。

また、同様にチャンネル 2 とチャンネル 3 の対応するレジスタは、チャンネル 2 を上位、チャンネル 3 を下位とする 16 ビットレジスタとして、ワードアクセスすることができます。

## 10.2 各レジスタの説明

### 10.2.1 タイマカウンタ (8TCNT)



8TCNT はそれぞれ 8 ビットのリード/ライト可能なアップカウンタで、入力する内部または外部クロックによってカウントアップされます。入力するクロックは、8TCR の CKS2~CKS0 ビットで選択します。8TCNT の値は、CPU から常にリード/ライト可能です。

8TCNT0、8TCNT1 および 8TCNT2、8TCNT3 を 1 本の 16 ビットレジスタとしてワードアクセスすることも可能です。

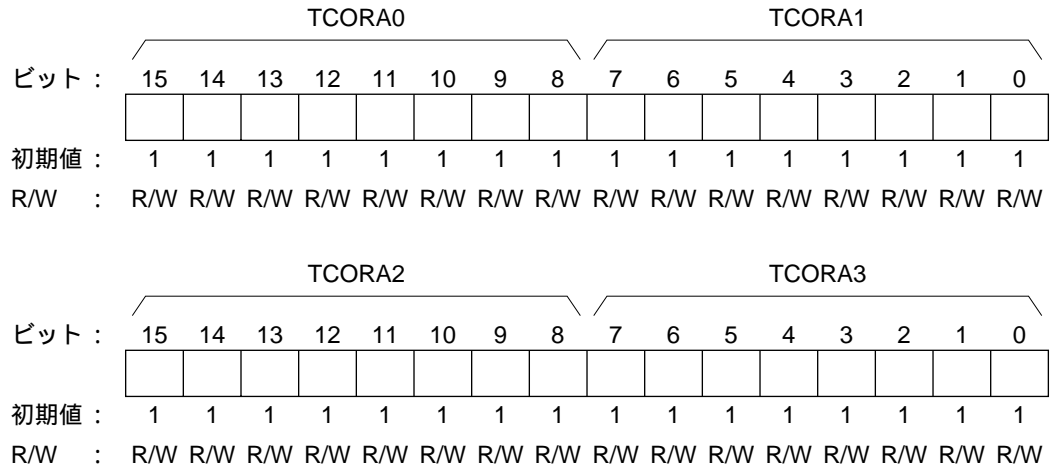
8TCNT は、インプットキャプチャ信号またはコンペアマッチ信号によりクリアすることができます。いずれの信号でクリアするかは、8TCR の CCLR1、CCLR0 ビットで選択します。

また、8TCNT がオーバーフロー (H'FF H'00) すると、8TCSR の OVF が 1 にセットされます。

8TCNT は、リセットまたはスタンバイモード時に H'00 にイニシャライズされます。

## 10.2.2 タイムコンスタントレジスタ A (TCORA)

TCORA はそれぞれ 8 ビットのリード/ライト可能なレジスタです。



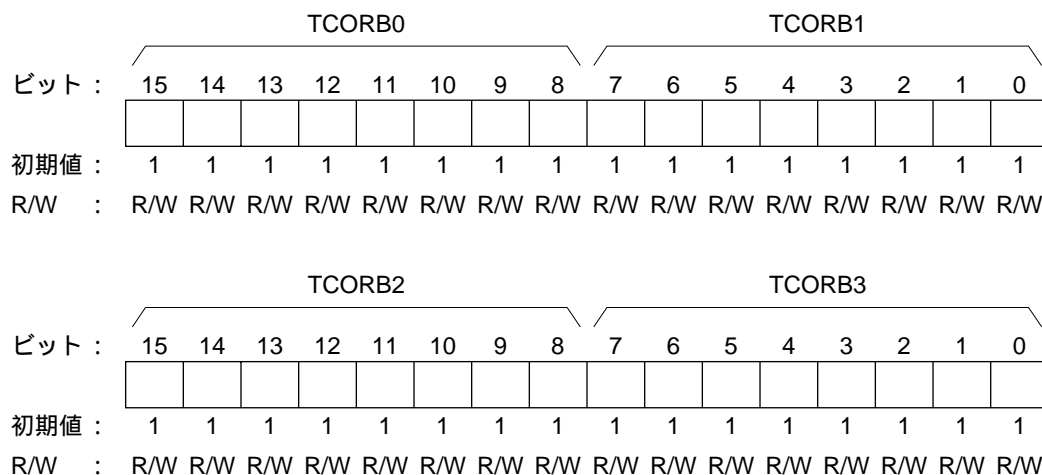
TCORA0、TCORA1 および TCORA2、TCORA3 を 1 本の 16 ビットレジスタとしてワードアクセスすることも可能です。

TCORA と 8TCNT の値は常に比較されており、両者の値が一致すると 8TCSR の CMFA が 1 にセットされます。

また、この一致による信号 (コンペアマッチ) と 8TCSR の OS1、OS0 ビットの設定により、タイマ出力を自由に制御することができます。

TCORA は、リセットまたはスタンバイモード時に H'FF にイニシャライズされます。

## 10.2.3 タイムコンスタントレジスタ B (TCORB)



TCORB はそれぞれ 8 ビットのリード/ライト可能なレジスタです。TCORB0、TCORB1 および TCORB2、TCORB3 を 1 本の 16 ビットレジスタとしてワードアクセスすることも可能です。

TCORB と 8TCNT の値は常に比較されており、両者の値が一致すると 8TCSR の CMFB が 1 にセットされます。

また、この一致による信号 (コンペアマッチ) と 8TCSR のアウトプット/インプットキャプチャエッジセレクト OIS3、OIS2 ビットの設定により、タイマ出力を自由に制御することができます。

インプットキャプチャとして使用しているときは、外部からのインプットキャプチャ信号を検出して、8TCNT の値を格納します。このとき対応する 8TCSR の CMFB フラグが "1" にセットされます。インプットキャプチャ信号の検出エッジは 8TCSR により行います。

TCORB は、リセットまたはスタンバイモード時に H'FF にイニシャライズされます。

**【注】** チャンネル 1 およびチャンネル 3 を TCORB インプットキャプチャに設定した場合、チャンネル 0 およびチャンネル 2 のコンペアマッチ B による CMFB フラグのセットは起こりません。

## 10.2.4 タイマコントロールレジスタ (8TCR)

ビット：	7	6	5	4	3	2	1	0
	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
初期値：	0	0	0	0	0	0	0	0
R/W：	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

8TCRはそれぞれ8ビットのリード/ライト可能なレジスタで、8TCNTの入力クロックの選択、8TCNTのクリア指定、および各割り込み要求の許可を制御します。

8TCRは、リセットまたはスタンバイモード時に、H'00にイニシャライズされます。なお、タイミングについては、「10.4 動作説明」を参照してください。

## ビット7：コンペアマッチインタラプトイネーブルB (CMIEB)

8TCSRのCMFBが1にセットされたとき、CMFBによる割り込み要求 (CMIB) の許可または禁止を選択します。

ビット7	説明
CMIEB	
0	CMFBによる割り込み要求 (CMIB) を禁止 (初期値)
1	CMFBによる割り込み要求 (CMIB) を許可

## ビット6：コンペアマッチインタラプトイネーブルA (CMIEA)

8TCSRのCMFAが1にセットされたとき、CMFAによる割り込み要求 (CMIA) の許可または禁止を選択します。

ビット6	説明
CMIEA	
0	CMFAによる割り込み要求 (CMIA) を禁止 (初期値)
1	CMFAによる割り込み要求 (CMIA) を許可

## ビット5：タイマオーバフローインタラプトイネーブル (OVIE)

8TCSRのOVFが1にセットされたとき、OVFによる割り込み要求 (OVI) の許可または禁止を選択します。

ビット5	説明
OVIE	
0	OVFによる割り込み要求 (OVI) を禁止 (初期値)
1	OVFによる割り込み要求 (OVI) を許可

## ビット4、3：カウンタクリア1、0（CCLR1、CCLR0）

8TCNTのクリア要因を指定します。クリア要因は、コンペアマッチA、BまたはインプットキャプチャBから選択します。

ビット4	ビット3	説明
CCLR1	CCLR0	
0	0	クリアを禁止 (初期値)
	1	コンペアマッチAによりクリア
1	0	コンペアマッチB/インプットキャプチャBによりクリア
	1	インプットキャプチャBによりクリア

【注】 8TCNT1および8TCNT3のカウンタクリア要因を、インプットキャプチャBに設定した場合、8TCNT0および8TCNT2はコンペアマッチBによりクリアされません。

## ビット2～0：クロックセレクト2～0（CKS2～CKS0）

8TCNTに入力するクロックを、内部クロックまたは外部クロックから選択します。

内部クロックは、システムクロック（ ）を分周した3種類のクロック（ /8、 /64、 /8192）から選択できます。これら内部クロックは、立ち上がりエッジでカウントします。

外部クロックのとき、クロック入力は立ち上がり、立ち下がり、または立ち上がり/立ち下がり両エッジのカウントの3種類から選択できます。

CKS2、CKS1、CKS0 = 1、0、0の設定の場合、チャンネル0と1およびチャンネル2と3でカスケード接続になります。

8TCR0と8TCR2に設定した場合と、8TCR1と8TCR3に設定した場合は、カウントアップのクロックソースが異なります。



ビット2	ビット1	ビット0	説明
CKS2	CKS1	CKS0	
0	0	0	クロック入力を禁止 (初期値)
		1	内部クロック : /8 立ち上がりエッジでカウント
	1	0	内部クロック : /64 立ち上がりエッジでカウント
		1	内部クロック : /8192 立ち上がりエッジでカウント
1	0	0	チャンネル0の場合(16ビットカウントモード) : 8TCNT1のオーバーフロー信号でカウント* <sup>1</sup> チャンネル1の場合(コンペアマッチカウントモード) : 8TCNT0のコンペアマッチAでカウント* <sup>1</sup> チャンネル2の場合(16ビットカウントモード) : 8TCNT3のオーバーフロー信号でカウント* <sup>2</sup> チャンネル3の場合(コンペアマッチカウントモード) : 8TCNT2のコンペアマッチAでカウント* <sup>2</sup>
		1	外部クロック : 立ち上がりエッジでカウント
		1	外部クロック : 立ち下がりエッジでカウント
	1	0	外部クロック : 立ち下がりエッジでカウント
		1	外部クロック : 立ち上がり/立ち下がり両エッジでカウント
		1	外部クロック : 立ち上がり/立ち下がり両エッジでカウント

【注】 \*1 チャンネル0のクロック入力を8TCNT1のオーバーフロー信号とし、チャンネル1のクロック入力を8TCNT0のコンペアマッチ信号とすると、カウントアップクロックが発生しません。この設定は行わないでください。

\*2 チャンネル2のクロック入力を8TCNT3のオーバーフロー信号とし、チャンネル3のクロック入力を8TCNT2のコンペアマッチ信号とすると、カウントアップクロックが発生しません。この設定は行わないでください。

## 10.2.5 タイマコントロール/ステータスレジスタ (8TCSR)

8TCSR0

ビット:	7	6	5	4	3	2	1	0
	CMFB	CMFA	OVF	ADTE	OIS3	OIS2	OS1	OS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W	R/W

8TCSR2

ビット:	7	6	5	4	3	2	1	0
	CMFB	CMFA	OVF	-	OIS3	OIS2	OS1	OS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	-	R/W	R/W	R/W	R/W

8TCSR1、8TCSR3

ビット:	7	6	5	4	3	2	1	0
	CMFB	CMFA	OVF	ICE	OIS3	OIS2	OS1	OS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W	R/W

【注】\* ビット7~5は、フラグをクリアするための0ライトのみ可能です。

8TCSRは8ビットのレジスタで、コンペアマッチ/インプットキャプチャやタイマオーバフローのステータスの表示、およびコンペアマッチ出力/インプットキャプチャのエッジの選択の制御を行います。

リセットまたはスタンバイモード時に、8TCSRはH'00にイニシャライズされます。

ビット7: コンペアマッチ/インプットキャプチャフラグB (CMFB)

TCORBのコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット7	説明
CMFB	
0	[クリア条件] (初期値) CMFB=1の状態、CMFBをリードした後、CMFBに0ライトしたとき
1	[セット条件] (1) 8TCNT = TCORBになったとき* (2) TCORBがインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号により8TCNTの値がTCORBに転送されたとき

【注】\* 8TCSR1、8TCSR3のICEビットが1のとき、8TCNT0 = TCORB0、8TCNT2 = TCORB2となってもCMFBフラグはセットされません。

ビット6：コンペアマッチフラグA（CMFA）

TCORAのコンペアマッチの発生を示すステータスフラグです。

ビット6	説明
CMFA	
0	[クリア条件] (初期値) CMFA=1の状態、CMFAをリードした後、CMFAに0をライトしたとき
1	[セット条件] 8TCNT = TCORAになったとき

ビット5：タイマオーバフローフラグ（OVF）

8TCNTがオーバフロー（H'FF H'00）したことを示すステータスフラグです。

ビット5	説明
OVF	
0	[クリア条件] (初期値) OVF=1の状態、OVFをリードした後、OVFに0をライトしたとき
1	[セット条件] 8TCNTがH'FF H'00になったとき

ビット4：A/Dトリガイネーブル（ADTE）（8TCSR0）

ADTEは、ADCRのTRGEとの組み合わせにより、コンペアマッチAまたは外部トリガによるA/D変換開始要求の許可または禁止を選択します。

8TCSR2のビット4はリザーブビットです。リード/ライトは可能です。

TRGE*	ビット4	説明
	ADTE	
0	0	コンペアマッチAまたは外部トリガ端子（ $\overline{ADTRG}$ ）入力によるA/D変換開始要求を禁止 (初期値)
	1	コンペアマッチAまたは外部トリガ端子（ $\overline{ADTRG}$ ）入力によるA/D変換開始要求を禁止
1	0	外部トリガ端子（ $\overline{ADTRG}$ ）入力によるA/D変換開始要求を許可、およびコンペアマッチAによるA/D変換開始要求を禁止
	1	コンペアマッチAによるA/D変換開始要求を許可、および外部トリガ端子（ $\overline{ADTRG}$ ）入力によるA/D変換開始要求を禁止

【注】 \* TRGEはA/Dコントロールレジスタ（ADCR）のビット7です。

ビット4：リザーブビット（8TCSR1の場合）

リザーブビットです。リード/ライトは可能です。

ビット4：インプットキャプチャイネーブル（ICE）（8TCSR1、8TCSR3）

TCORB1、TCORB3の機能を選択します。

ビット4	説明
ICE	
0	TCORB1、TCORB3はコンペアマッチレジスタ（初期値）
1	TCORB1、TCORB3はインプットキャプチャレジスタ

8TCSR1 および8TCSR3のICEビットを1にセットしたときの、チャンネル0～3のTCORA、TCORB各レジスタの動作を表10.3、表10.4にまとめます。

表10.3 8TCSR1レジスタのICEビット=1に設定した場合のチャンネル0とチャンネル1の動作

レジスタ	レジスタ機能	ステータスフラグの変化	タイマ出力 キャプチャ入力	割込み要求
TCORA0	コンペアマッチ動作	コンペアマッチにより8TCSR0のCMFA=0 1にセット	TMO <sub>0</sub> から出力制御可能	コンペアマッチによりCMIA0割込み要求発生
TCORB0	コンペアマッチ動作	コンペアマッチが発生しても8TCSR0のCMFB=0 1にセットされない	TMO <sub>0</sub> から出力しない	コンペアマッチが発生してもCMIB0割込み要求発生しない
TCORA1	コンペアマッチ動作	コンペアマッチにより8TCSR1のCMFA=0 1にセット	TMIO <sub>1</sub> はインプットキャプチャ専用端子	コンペアマッチによりCMIA1割込み要求発生
TCORB1	インプットキャプチャ動作	インプットキャプチャにより8TCSR1のCMFB=0 1にセット	TMIO <sub>1</sub> はインプットキャプチャ専用端子	インプットキャプチャによりCMIB1割込み要求発生

表 10.4 8TCSR3 レジスタの ICE ビット = 1 に設定した場合のチャンネル 2 とチャンネル 3 の動作

レジスタ	レジスタ機能	ステータスフラグの変化	タイマ出力 キャプチャ入力	割込み要求
TCORA2	コンペアマッチ動作	コンペアマッチにより 8TCSR2 の CMFA=0 1 にセット	TMIO <sub>2</sub> から出力制御可能	コンペアマッチにより CMIA2 割込み要求発生
TCORB2	コンペアマッチ動作	コンペアマッチが発生しても 8TCSR2 の CMFB=0 1 にセットされない	TMIO <sub>2</sub> から出力しない	コンペアマッチが発生しても CMIB2 割込み要求発生しない
TCORA3	コンペアマッチ動作	コンペアマッチにより 8TCSR3 の CMFA=0 1 にセット	TMIO <sub>3</sub> はインプットキャプチャ専用端子	コンペアマッチにより CMIA3 割込み要求発生
TCORB3	インプットキャプチャ動作	インプットキャプチャにより 8TCSR3 の CMFB=0 1 にセット	TMIO <sub>3</sub> はインプットキャプチャ専用端子	インプットキャプチャにより CMIB3 割込み要求発生

ビット 3、2 : アウトプット / インプットキャプチャエッジセレクト B3、2 ( OIS3、OIS2 )

OIS3、OIS2 は 8TCSR1 ( 8TCSR3 ) の ICE との組み合わせにより、コンペアマッチ B による出力レベルの選択またはインプットキャプチャ入力の検出エッジの選択をします。

8TCSR1 ( 8TCSR3 ) のビット 4 の設定により TCORB1 ( TCORB3 ) の機能が変わります。TCORB0、TCORB2 は 8TCSR1 ( 8TCSR3 ) のビット 4 の設定にかかわらずコンペアマッチレジスタとして機能します。

8TCSR1 ( 8TCSR3 ) の ICE ビット	ビット 3	ビット 2	説 明
	OIS3	OIS2	
0	0	0	コンペアマッチ B で変化しない ( 初期値 )
		1	コンペアマッチ B で 0 出力
	1	0	コンペアマッチ B で 1 出力
		1	コンペアマッチ B ごとに反転出力 ( トグル出力 )
1	0	0	立上がりエッジで TCORB インプットキャプチャ
		1	立下がりエッジで TCORB インプットキャプチャ
	1	0	立上がり / 立下がりの両エッジで TCORB インプット
		1	キャプチャ

- ・コンペアマッチレジスタとして機能している場合、タイマ出力はトグル出力 > 1 出力 > 0 出力の順で優先順位が高くなるように設定してあります。
- ・コンペアマッチが A、B 同時に発生した場合は、優先順位が高い方のコンペアマッチに従って出力が変化します。
- ・OIS3、OIS2、OS1、OS0 ビットがすべて 0 の場合にはタイマ出力は禁止されます。

ビット 1、0 : アウトプットセレクト A1、0 (OS1、OS0)

コンペアマッチ A による出力レベルを選択します。

ビット 1	ビット 0	説 明
OS1	OS0	
0	0	コンペアマッチ A で変化しない (初期値)
	1	コンペアマッチ A で 0 出力
1	0	コンペアマッチ A で 1 出力
	1	コンペアマッチ A ごとに反転出力 (トグル出力)

- ・コンペアマッチレジスタとして機能している場合、タイマ出力はトグル出力 > 1 出力 > 0 出力の順で優先順位が高くなるように設定してあります。
- ・コンペアマッチが A、B 同時に発生した場合は、優先順位が高い方のコンペアマッチに従って出力が変化します。
- ・OIS3、OIS2、OS1、OS0 ビットがすべて 0 の場合にはタイマ出力は禁止されます。

## 10.3 CPUとのインタフェース

### 10.3.1 8ビットレジスタ

8TCNT、TCORA、TCORB、8TCR、8TCSR は 8 ビットのレジスタです。これらのレジスタは、CPU と内部 16 ビットデータバスで接続されており、ワード単位のリード/ライトが可能です。また、バイト単位のリード/ライトもできます。

8TCNT に対してワード単位のリード/ライトを行った場合の動作を図 10.2、図 10.3 に示します。

また、8TCNT0、8TCNT1 に対してワード単位のリード/ライトを行った場合の動作を図 10.4、図 10.5、図 10.6、図 10.7 に示します。

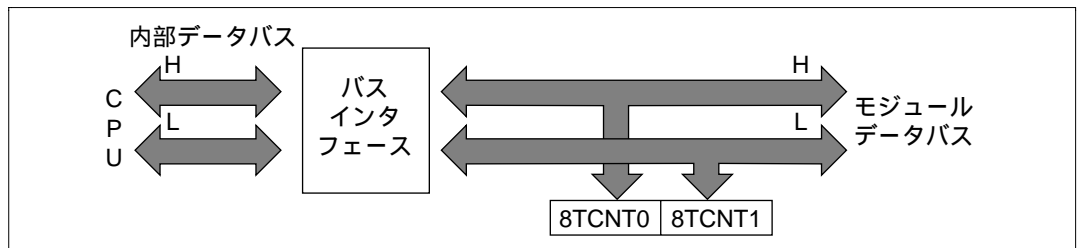


図 10.2 8TCNT のアクセス動作 [ CPU 8TCNT (ワード) ]

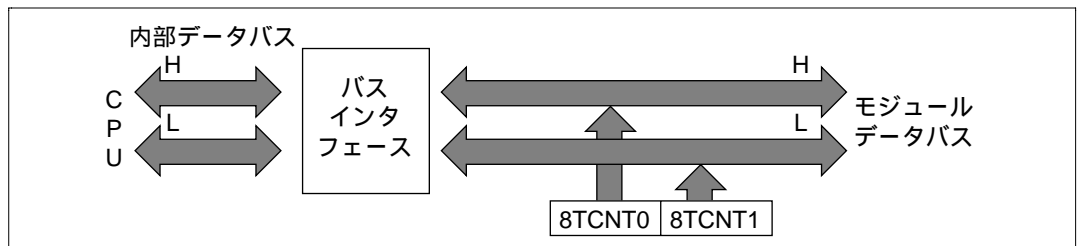


図 10.3 8TCNT のアクセス動作 [ 8TCNT CPU (ワード) ]

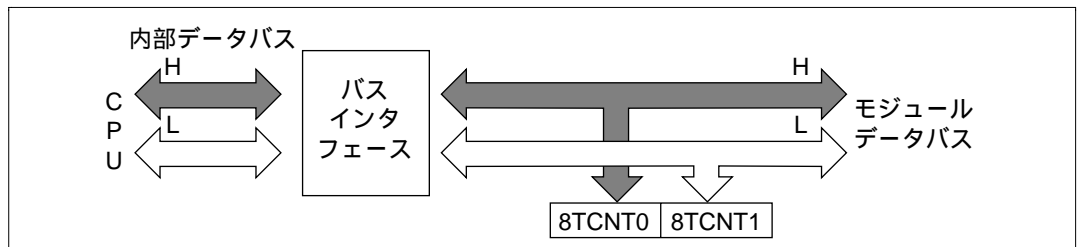


図 10.4 8TCNTH のアクセス動作 [ CPU 8TCNTH (上位バイト) ]

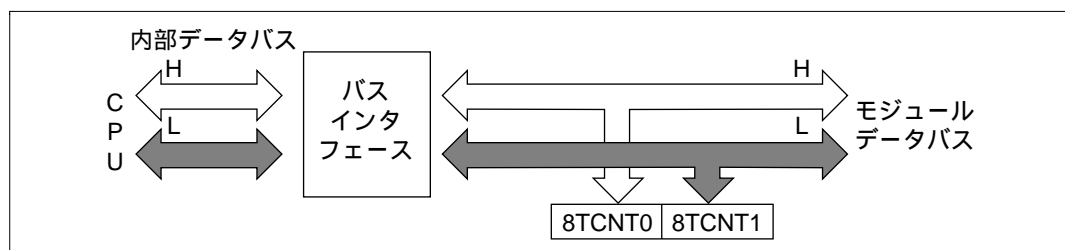


図 10.5 8TCNT1のアクセス動作 [CPU 8TCNT1 (下位バイト)]

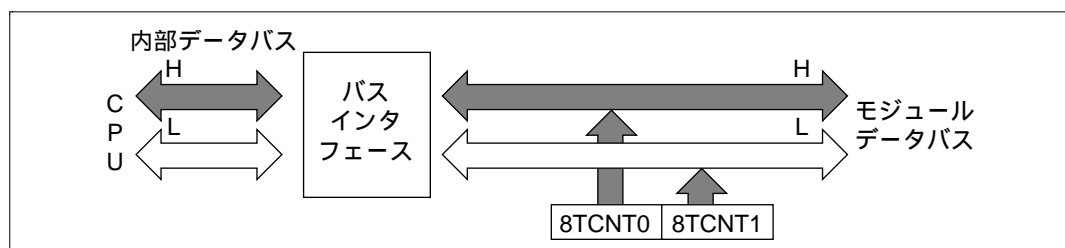


図 10.6 8TCNT0のアクセス動作 [8TCNT0 CPU (上位バイト)]

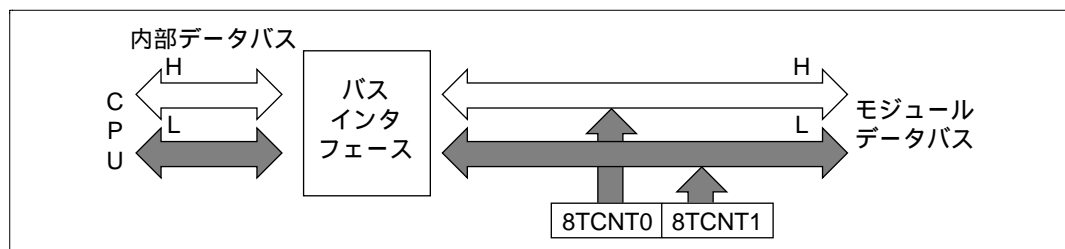


図 10.7 8TCNT1のアクセス動作 [8TCNT1 CPU (下位バイト)]



## 10.4 動作説明

### 10.4.1 8TCNT のカウントタイミング

8TCNT は、入力されたクロック（内部クロックまたは外部クロック）によりカウントアップされます。

#### (1) 内部クロック動作の場合

8TCR の CKS2 ~ CKS0 ビットの設定により、システムクロック（ ）を分周して作られる 3 種類の内部クロック（ /8、 /64、 /8192）が選択されます。このタイミングを図 10.8 に示します。

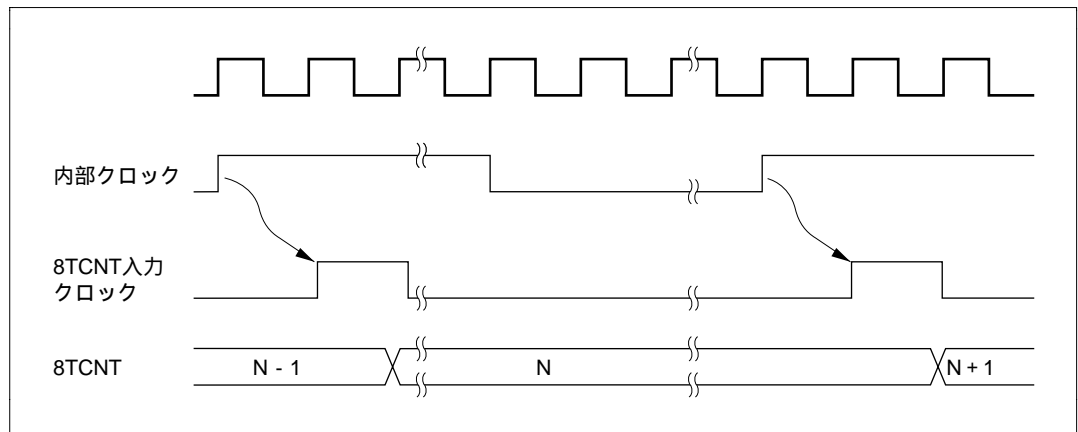


図 10.8 内部クロック動作時のカウントタイミング

【注】 16ビットタイマと8ビットタイマで同じ内部クロックを選択した場合でもカウントアップさせるエッジが異なるため、同じ動作をしませんので注意してください。

#### (2) 外部クロック動作の場合

8TCR の CKS2 ~ CKS0 ビットの設定により、外部クロックの立ち上がりエッジ、立ち下がりエッジ、立ち上がり / 立ち下がり両エッジのいずれかによるカウントアップが選択されます。

なお、外部クロックのパルス幅は、単エッジの場合は 1.5 ステート以上、両エッジの場合は 2.5 ステート以上必要です。これ以下のパルス幅では、正しく動作しませんので注意してください。

図 10.9 に、外部クロックとして、立ち上がり / 立ち下がり両エッジの場合のタイミングを示します。

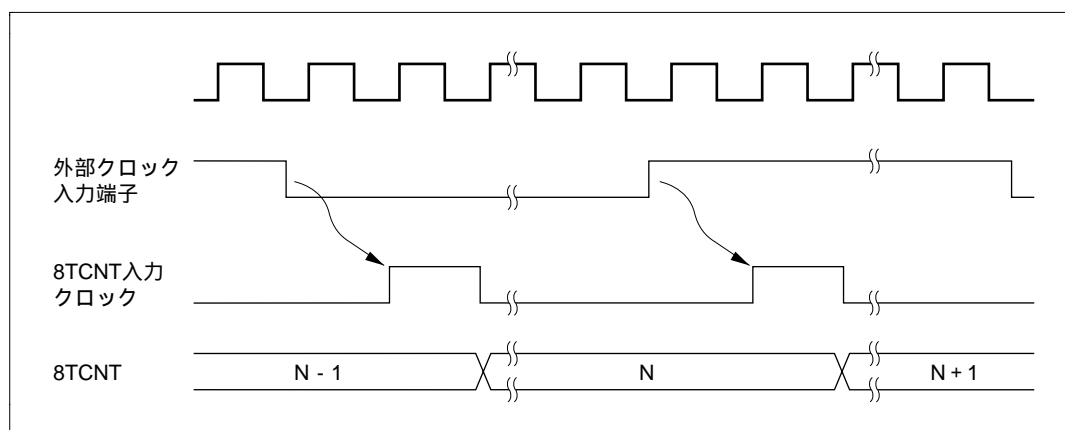


図 10.9 外部クロック動作時のカウントタイミング (両エッジ検出の場合)

## 10.4.2 コンペアマッチタイミング

### (1) タイマ出力タイミング

タイマ出力はコンペアマッチ A、B が発生したとき、8TCSR の OIS3,2、OS1,0 ビットで選択された状態 (変化しない、0 出力、1 出力、トグル出力) で出力されます。

図 10.10 にコンペアマッチ A 信号によるトグル出力の場合の出力タイミングを示します。

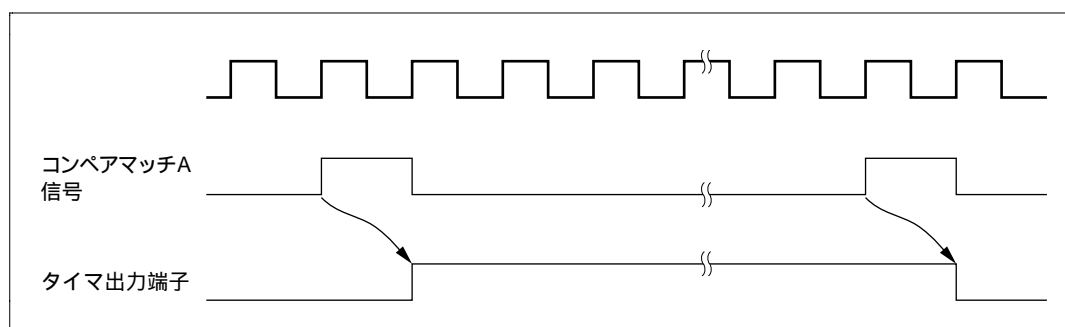


図 10.10 タイマ出力タイミング

(2) コンペアマッチによるクリア

8TCNTは、8TCRのCCLR1、CCLR0ビットの選択によりコンペアマッチAまたはコンペアマッチBでクリアされます。このクリアされるタイミングを図10.11に示します。

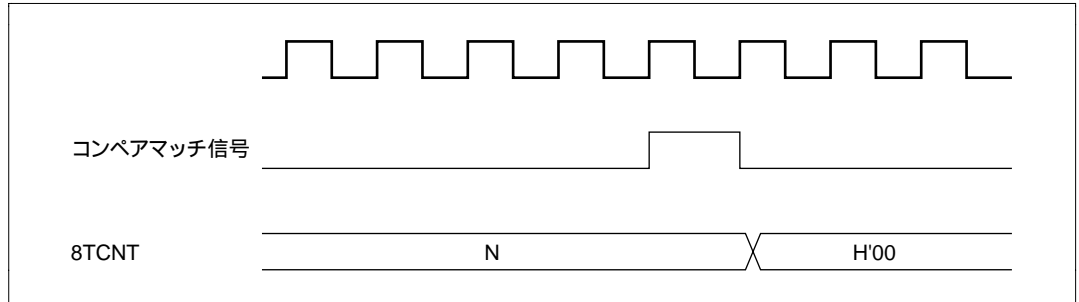


図 10.11 コンペアマッチによるクリアタイミング

(3) インพุットキャプチャによるクリア

8TCNTは、8TCRのCCLR1、CCLR0ビットの選択によりインพุットキャプチャBでクリアされます。このタイミングを図10.12に示します。

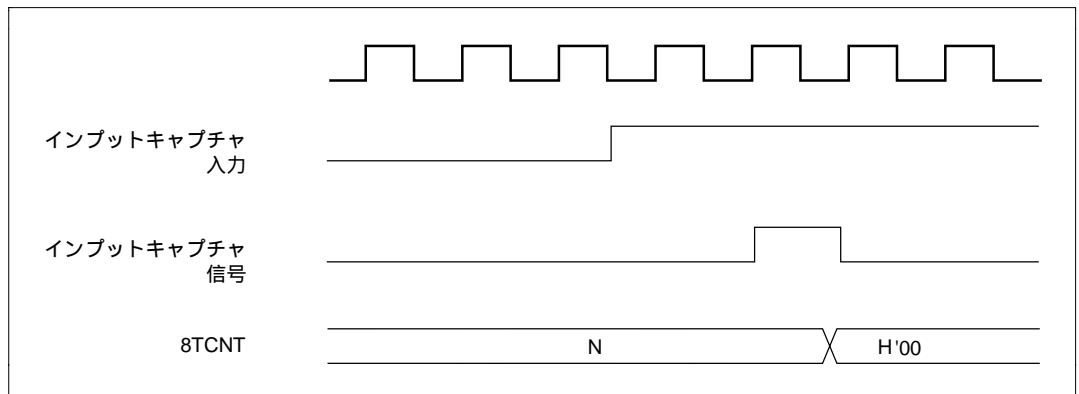


図 10.12 インพุットキャプチャによるクリアタイミング

### 10.4.3 インพุットキャプチャ信号タイミング

インพุットキャプチャ入力、8TCSR の設定により立上がりエッジ / 立下がりエッジ / 両エッジの選択ができます。

立上がりエッジを選択した場合のタイミングを図 10.13 に示します。

インพุットキャプチャ入力信号のパルス幅は、単エッジの場合は 1.5 システムクロック以上、両エッジの場合は 2.5 システムクロック以上必要です。

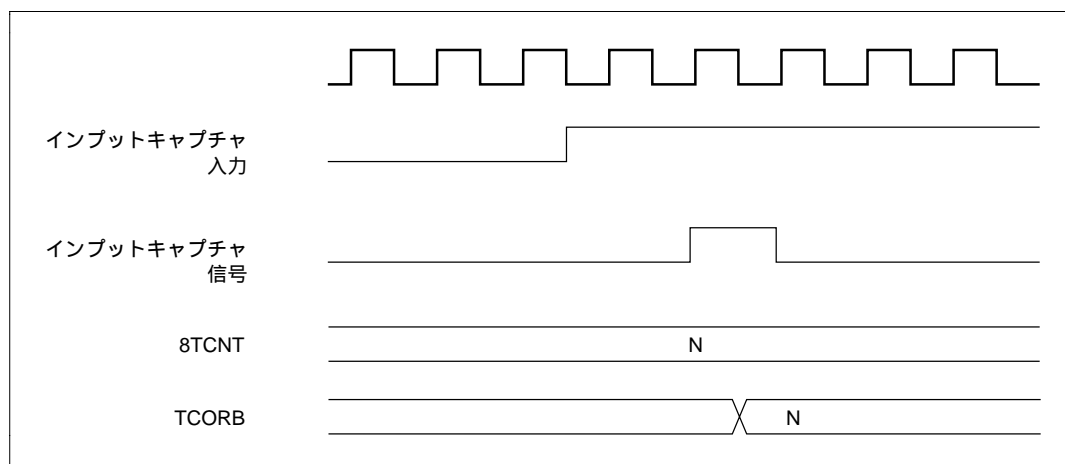


図 10.13 インพุットキャプチャ入力信号タイミング

#### 10.4.4 ステータスフラグのセットタイミング

##### (1) コンペアマッチ時の CMFA、CMFB フラグのセットタイミング

8TCSR の CMFA、CMFB は、TCOR と 8TCNT の値が一致したとき出力されるコンペアマッチ信号により 1 にセットされます。コンペアマッチ信号は一致した最後のステート（8TCNT が一致したカウント値を更新するタイミング）で発生します。

したがって、8TCNT と TCOR が一致した後、カウントアップクロックが発生するまでコンペアマッチ信号は発生しません。このタイミングを図 10.14 に示します。

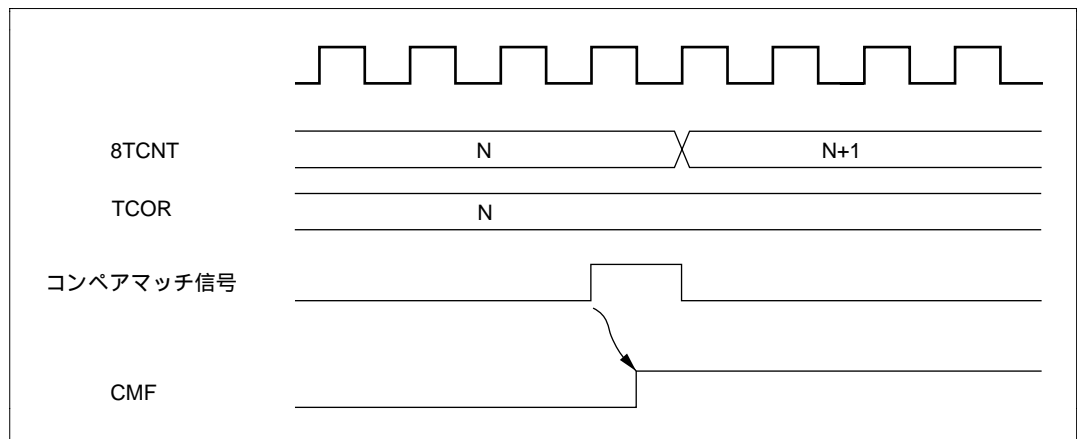


図 10.14 コンペアマッチ時の CMF フラグセットタイミング

##### (2) インพุットキャプチャ時の CMFB フラグのセットタイミング

インพุットキャプチャ信号の発生により CMFB フラグは 1 にセットされ、同時に 8TCNT の値が対応する TCORB に転送されます。

このタイミングを図 10.15 に示します。

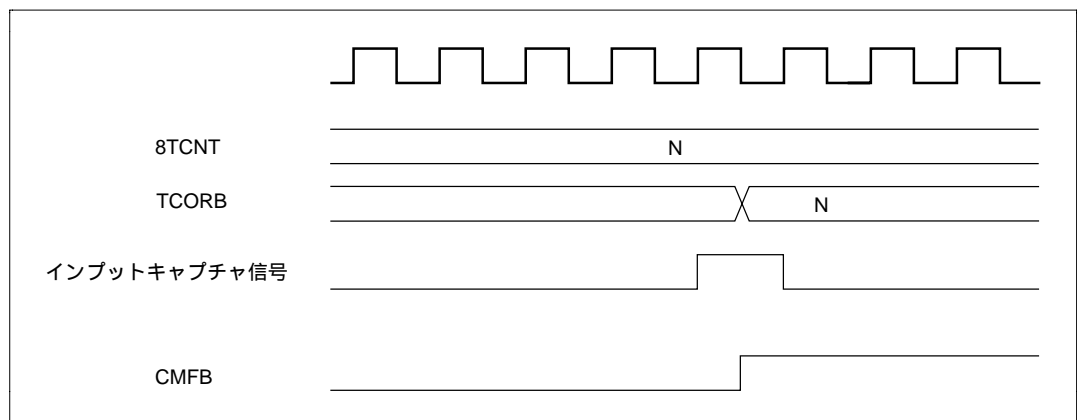


図 10.15 インพุットキャプチャ時の CMFB フラグセットタイミング

### (3) オーバフローフラグ (OVF) のセットタイミング

8TCSR の OVF は、オーバフロー (H'FF H'00) したとき出力されるオーバフロー信号により 1 にセットされます。

このタイミングを図 10.16 に示します。

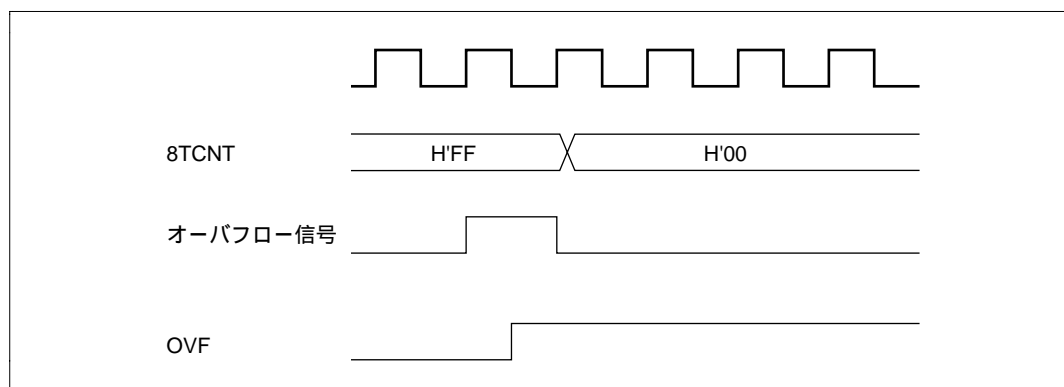


図 10.16 OVF のセットタイミング

## 10.4.5 カスケード接続時の動作

8TCR0、8TCR1 のいずれか一方の CKS2~CKS0 ビットを (100) に設定すると、チャンネル 0 とチャンネル 1 の 8 ビットタイマはカスケード接続されます。この場合、1 本の 16 ビットタイマとして使用する (16 ビットタイマモード) か、またはチャンネル 0 の 8 ビットタイマのコンペアマッチをチャンネル 1 でカウントする (コンペアマッチカウントモード) ことができます。なおチャンネル 2 とチャンネル 3 も同様に 8TCR2、8TCR3 のいずれか一方の CKS2~CKS0 ビットを (100) に設定すると、チャンネル 2 とチャンネル 3 の 8 ビットタイマはカスケード接続されます。この場合、1 本の 16 ビットタイマとして使用する (16 ビットタイマモード) か、またはチャンネル 2 の 8 ビットタイマのコンペアマッチをチャンネル 3 でカウントする (コンペアマッチカウントモード) ことができます。このとき、本タイマは以下のように動作します。

### (1) 16 ビットカウントモード

チャンネル 0、1 の場合：

8TCR0 の CKS2~CKS0 ビットが (100) のとき、本タイマはチャンネル 0 を上位 8 ビット、チャンネル 1 を下位 8 ビットとする 1 チャンネルの 16 ビットタイマとして動作します。

#### (a) コンペアマッチ時の設定

- ・ 8TCR0 の CMF フラグは、16 ビットのコンペアマッチが発生したとき 1 にセットされます。
- ・ 8TCR1 の CMF フラグは、下位 8 ビットのコンペアマッチが発生したとき 1 にセットされます。

- ・8TCSR0のOIS3、OIS2、OS1、OS0ビットによるTMO0端子の出力制御は16ビットのコンペアマッチ条件に従います。
  - ・8TCSR1のOIS3、OIS2、OS1、OS0ビットによるTMIO<sub>1</sub>端子の出力制御は下位8ビットのコンペアマッチ条件に従います。
- (b) インพุットキャプチャ時の設定
- ・8TCR0、8TCR1のCMFBフラグは、8TCSR1のICEビットが1でインพุットキャプチャが発生したとき1にセットされます。
  - ・8TCSR0のOIS3、OIS2ビットによりTMIO<sub>1</sub>端子のインพุットキャプチャ入力信号の検出エッジを選択します。
- (c) カウンタクリアの指定
- ・8TCR0のCCLR1、CCLR0ビットでコンペアマッチまたはインพุットキャプチャによるカウンタクリアをそれぞれ設定した場合、16ビットカウンタ(8TCNT0、8TCNT1の両方)がクリアされます。
  - ・8TCR1のCCLR1、CCLR0ビットの設定は無効になります。下位8ビットのみのカウンタクリアはできません。
- (d) OVFフラグの動作
- ・8TCSR0のOVFフラグは、16ビットのカウンタ(8TCNT0、8TCNT1)がオーバーフロー(H'FFFF H'0000)したとき1にセットされます。
  - ・8TCSR1のOVFフラグは、8ビットのカウンタ(8TCNT1)がオーバーフロー(H'FF H'00)したとき1にセットされます。

チャンネル2、3の場合：

8TCR2のCKS2~CKS0ビットが(100)のとき、本タイマはチャンネル2を上位8ビット、チャンネル3を下位8ビットとする1チャンネルの16ビットタイマとして動作します。

- (a) コンペアマッチ時の設定
- ・8TCR2のCMFフラグは、16ビットのコンペアマッチが発生したとき1にセットされます。
  - ・8TCR3のCMFフラグは、下位8ビットのコンペアマッチが発生したとき1にセットされます。
  - ・8TCSR2のOIS3、OIS2、OS1、OS0ビットによるTMO<sub>2</sub>端子の出力制御は16ビットのコンペアマッチ条件に従います。
  - ・8TCSR3のOIS3、OIS2、OS1、OS0ビットによるTMIO<sub>3</sub>端子の出力制御は下位8ビットのコンペアマッチ条件に従います。
- (b) インพุットキャプチャ時の設定
- ・8TCR2、8TCR3のCMFBフラグは、8TCSR3のICEビットが"1"でインพุットキャプチャが発生したとき1にセットされます。
  - ・8TCSR2のOIS3、OIS2ビットによりTMIO<sub>3</sub>端子のインพุットキャプチャ入力信号の検出エッジを選択します。

## (c) カウンタクリア指定

- ・8TCR2のCCLR1、CCLR0ビットでコンペアマッチによるカウンタクリアをそれぞれ設定した場合、16ビットカウンタ(8TCNT2、8TCNT3の両方)がクリアされません。
- ・8TCR3のCCLR1、CCLR0ビットの設定は無効になります。下位8ビットのみのカウンタクリアはできません。

## (d) OVF フラグの動作

- ・8TCR2のOVFフラグは16ビットのカウンタ(8TCNT2、8TCNT3)がオーバーフロー(H'FFFF H'0000)したとき1にセットされます。
- ・8TCR3のOVFフラグは、8ビットのカウンタ(8TCNT3)がオーバーフロー(H'FF H'00)したとき1にセットされます。

## (2) コンペアマッチカウントモード

チャンネル0、1の場合：

8TCR1のCKS2～CKS0ビットが(100)のとき、8TCNT1はチャンネル0のコンペアマッチAをカウントします。

チャンネル0、1の制御はそれぞれ独立に行われます。CMFフラグのセット、割込みの発生、TMO端子の出力、カウンタクリアなどは、各チャンネルの設定に従います。

チャンネル2、3の場合：

8TCR3のCKS2～CKS0ビットが(100)のとき、8TCNT3はチャンネル2のコンペアマッチAをカウントします。

チャンネル2、3の制御はそれぞれ独立に行われます。CMFフラグのセット、割込みの発生、TMO端子の出力、カウンタクリアなどは、各チャンネルの設定に従います。

## (3) 使用上の注意

同一グループ内で16ビットカウンタモードとコンペアマッチカウントモードを同時に設定した場合、8TCNTの入カクロックが発生しなくなるため、カウンタが停止して動作しません。この設定は行わないでください。



## 10.4.6 インพุットキャプチャの設定

インพุットキャプチャ/アウトプットコンペア端子 (TMIO<sub>1</sub>、TMIO<sub>3</sub>) の入力エッジを検出して 8TCNT の値を TCORB に転送することができます。検出エッジは立上がりエッジ/立下がりエッジ/両エッジから選択できます。また、16ビットカウントモードのときは、16ビットのインพุットキャプチャとして使用できます。

### (1) 8ビットタイマ (通常動作) 時のインพุットキャプチャ動作の設定手順例

チャンネル1の場合：

- (1) 8TCSR1 の ICE ビットにより TCORB1 を 8ビットのインพุットキャプチャレジスタに設定します。
- (2) 8TCSR1 の OIS3、OIS2 ビットによって、インพุットキャプチャ信号 (TMIO<sub>1</sub>) の入力エッジを立上がりエッジ/立下がりエッジ/両エッジから選択します。
- (3) 8TCR1 の CKS2~CKS0 ビットにより入力クロックを選択し、8TCNT のカウント動作を開始してください。

チャンネル3の場合：

- (1) 8TCSR3 の ICE ビットにより TCORB3 を 8ビットのインพุットキャプチャレジスタに設定します。
- (2) 8TCSR3 の OIS3、OIS2 ビットによって、インพุットキャプチャ信号 (TMIO<sub>3</sub>) の入力エッジを立上がりエッジ/立下がりエッジ/両エッジから選択します。
- (3) 8TCR3 の CKS2~CKS0 ビットにより入力クロックを選択し、8TCNT のカウント動作を開始してください。

**【注】** チャンネル1のTCORB1をインพุットキャプチャとして使用する場合、チャンネル0のTCORB0レジスタはコンペアマッチレジスタとして使用できません。  
またチャンネル3のTCORB3をインพุットキャプチャとして使用する場合、チャンネル2のTCORB2レジスタはコンペアマッチレジスタとして使用できません。

## (2) 16ビットカウントモード時のインプットキャプチャ動作の設定手順

チャンネル0、1の場合：

- (1) 16ビットカウントモード時、8TCSR1のICEビットを1に設定すると、TCORB0、TCORB1は16ビットのインプットキャプチャレジスタになります。
- (2) 8TCSR0のOIS3、OIS2ビットによって、インプットキャプチャ信号(TMIO<sub>1</sub>)の入力エッジを立上がりエッジ/立下がりエッジ/両エッジから選択します。(16ビットカウントモードのときは8TCSR1のOIS3、OIS2ビットの設定は無効になります。)
- (3) 8TCR1のCKS2~CKS0ビットにより入力クロックを選択し、8TCNTのカウント動作を開始してください。

チャンネル2、3の場合：

- (1) 16ビットカウントモード時、8TCSR3のICEビットを1に設定すると、TCORB2、TCORB3は16ビットのインプットキャプチャレジスタになります。
- (2) 8TCSR2のOIS3、OIS2ビットによって、インプットキャプチャ信号(TMIO<sub>3</sub>)の入力エッジを立上がりエッジ/立下がりエッジ/両エッジから選択します。(16ビットカウントモードのときは8TCSR3のOIS3、OIS2ビットの設定は無効になります。)
- (3) 8TCR3のCKS2~CKS0ビットにより入力クロックを選択し、8TCNTのカウント動作を開始してください。

## 10.5 割込み

### 10.5.1 割込み要因

8ビットタイマの割込み要因は、CMIA、CMIB、OVIの3種類があります。表10.5に各割込み要因と優先順位を示します。各割込み要因は、8TCRの各割込みイネーブルビットにより許可または禁止が設定され、それぞれ独立に割込みコントローラに送られます。

表 10.5 8ビットタイマ割込み要因の優先順位

割込み要因	内 容	優先順位
CMIA	CMFAによる割込み	高 ↑ 低
CMIB	CMFBによる割込み	
TOVI	OVIによる割込み	

なお、コンペアマッチ割り込み（CMIA1/CMIB1、CMIA3/CMIB3）およびオーバーフロー割込み（TOVI0/TOVI1、TOVI2/TOVI3）は、2つの割り込みでベクタを兼用しています。

表10.6に割り込み要因一覧を示します。

表 10.6 8ビットタイマ割込み要因一覧

チャンネル	割込み要因	内 容
0	CMIA0	TCORA0のコンペアマッチ
	CMIB0	TCORB0のコンペアマッチ / インพุットキャプチャ
1	CMIA1/CMIB1	TCORA1のコンペアマッチ、または TCORB1のコンペアマッチ / インพุットキャプチャ
0、1	TOVI0/TOVI1	カウンタ0、またはカウンタ1のオーバーフロー
2	CMIA2	TCORA2のコンペアマッチ
	CMIB2	TCORB2のコンペアマッチ / インพุットキャプチャ
3	CMIA3/CMIB3	TCORA3のコンペアマッチ、または TCORB3のコンペアマッチ / インพุットキャプチャ
2、3	TOVI2/TOVI3	カウンタ2、またはカウンタ3のオーバーフロー

### 10.5.2 A/D変換器の起動

チャンネル0のコンペアマッチAのみ、A/D変換器を起動することができます。

チャンネル0のコンペアマッチAの発生により、8TCSR0のCMFAフラグが1にセットされたとき、ADTEビットが1にセットされていると、A/D変換器に対して、A/D変換の開始を要求します。このとき、ADCRのTRGEビットが1にセットされていると、A/D変換が開始されます。8TCSR0のADTEビットが1にセットされている場合、A/D変換器の外部トリガ端子（ $\overline{\text{ADTRG}}$ ）入力は無効となります。

## 10.6 8ビットタイマの使用例

任意のデューティパルスを出力させた例を図 10.17 に示します。これは次に示すように設定します。

- (1) TCORA のコンペアマッチにより 8TCNT がクリアされるように、8TCR の CCLR1 ビットを 0 にクリア、CCLR0 ビットを 1 にセットします。
- (2) TCORA のコンペアマッチにより 1 出力、TCORB のコンペアマッチにより 0 出力になるように 8TCSR の OIS3,2、OS1,0 ビットを (0110) に設定します。

以上の設定により周期が TCORA、パルス幅が TCORB の波形をソフトウェアの介入なしに出力できます。

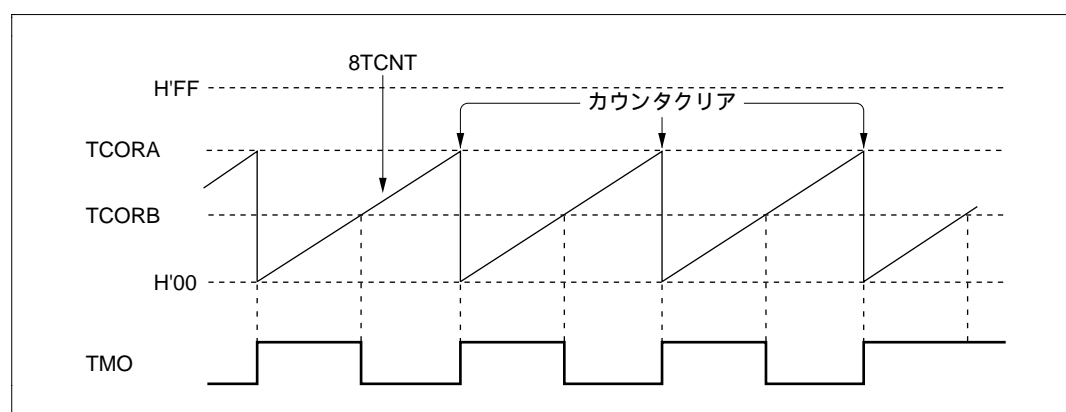


図 10.17 パルス出力例

## 10.7 使用上の注意

8ビットタイマの動作中、次のような競合が発生した場合、以下のような動作が起こりますので注意してください。

### 10.7.1 8TCNTのライトとクリアの競合

8TCNTのライトサイクル中の $T_3$ 状態で、カウンタクリア信号が発生すると、8TCNTへの書込みサイクルは行われず、8TCNTのクリアが優先されます。

このタイミングを図10.18に示します。

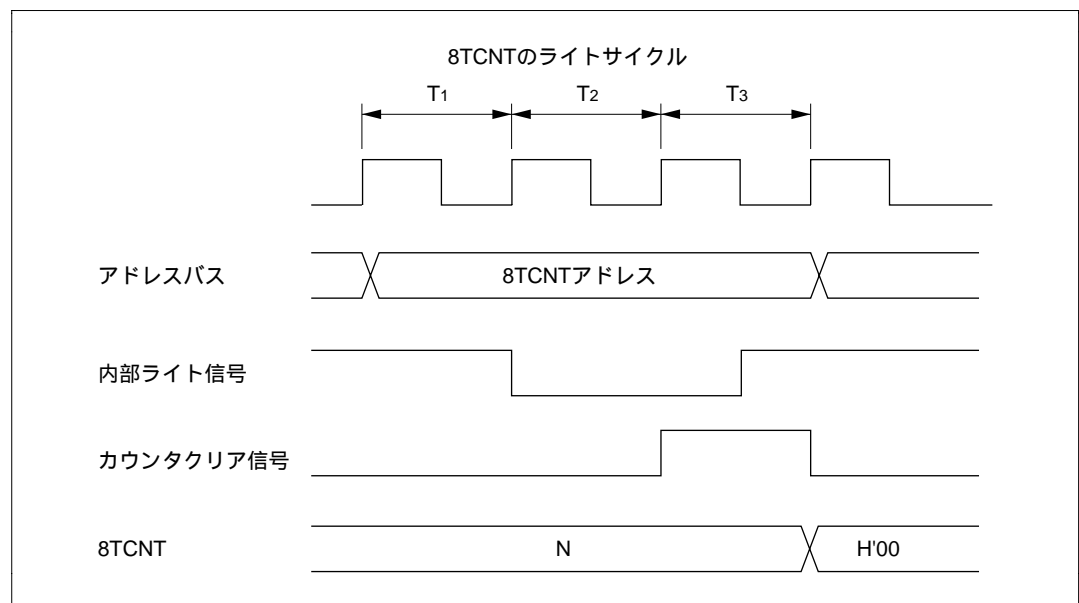


図 10.18 8TCNTのライトとクリアの競合

## 10.7.2 8TCNTのライトとカウントアップの競合

8TCNTのライトサイクル中の $T_3$ 状態でカウントアップが発生しても、カウントアップされずカウンタライトが優先されます。

このタイミングを図10.19に示します。

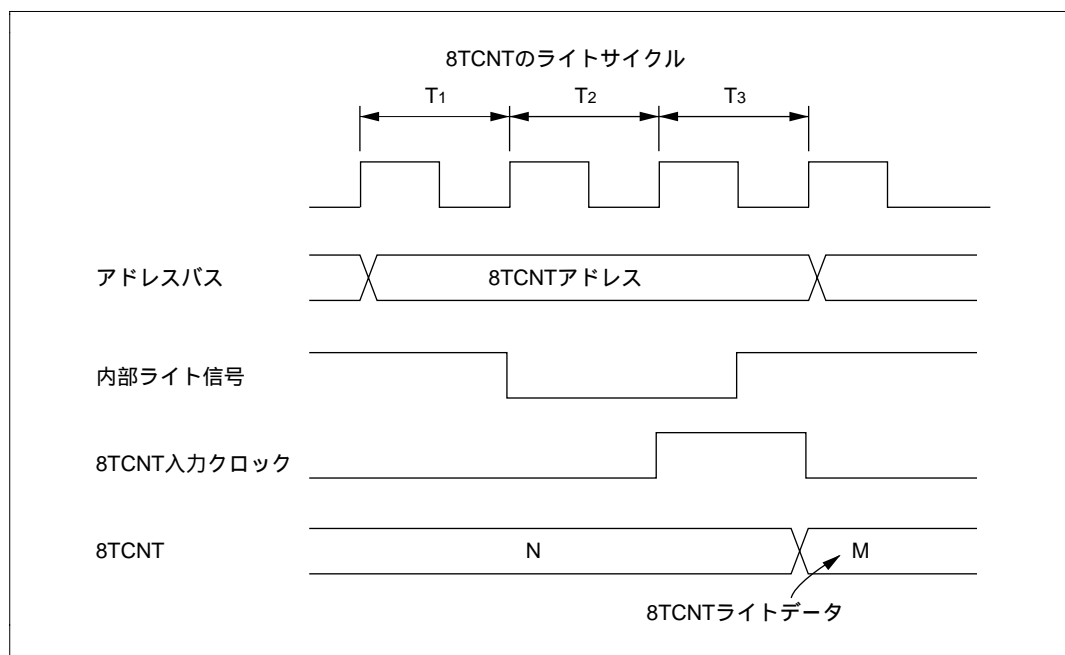


図 10.19 8TCNTのライトとカウントアップの競合

### 10.7.3 TCOR のライトとコンペアマッチの競合

TCOR のライトサイクル中の  $T_3$  ステートで、コンペアマッチが発生しても、TCOR のライトが優先され、コンペアマッチ信号は禁止されます。

このタイミングを図 10.20 に示します。

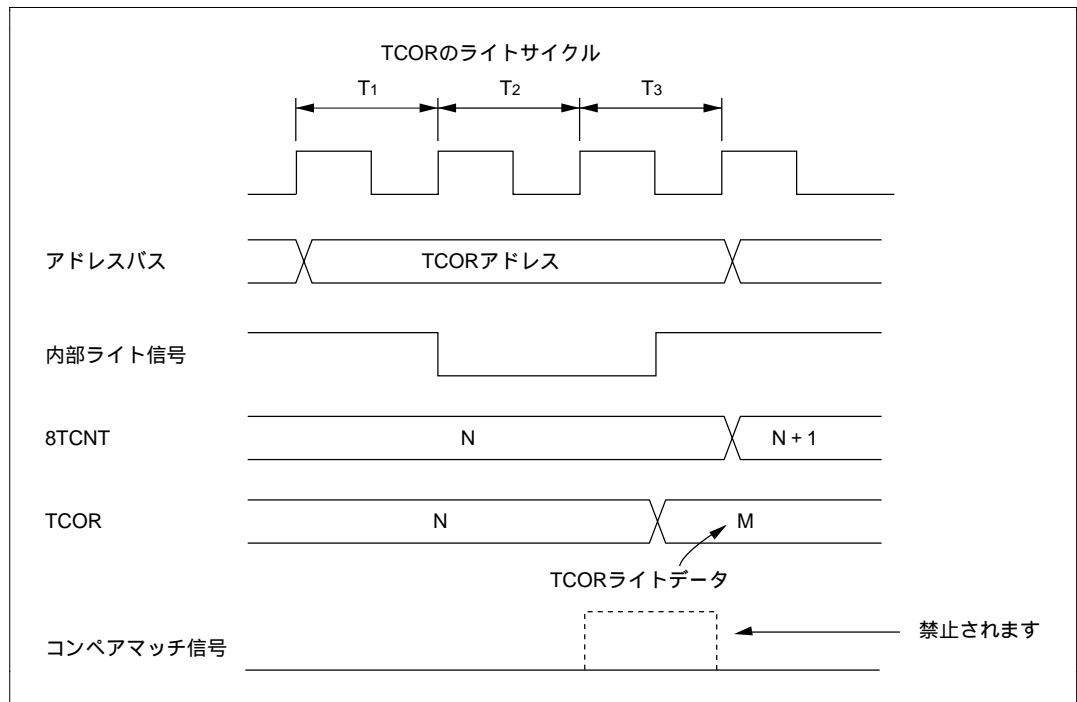


図 10.20 TCOR のライトとコンペアマッチの競合

### 10.7.4 TCOR のリードとインプットキャプチャの競合

TCOR のリードサイクル中の  $T_3$  ステートで、インプットキャプチャ信号が発生すると、リードされるデータはインプットキャプチャ転送前のデータです。

このタイミング図を図 10.21 に示します。

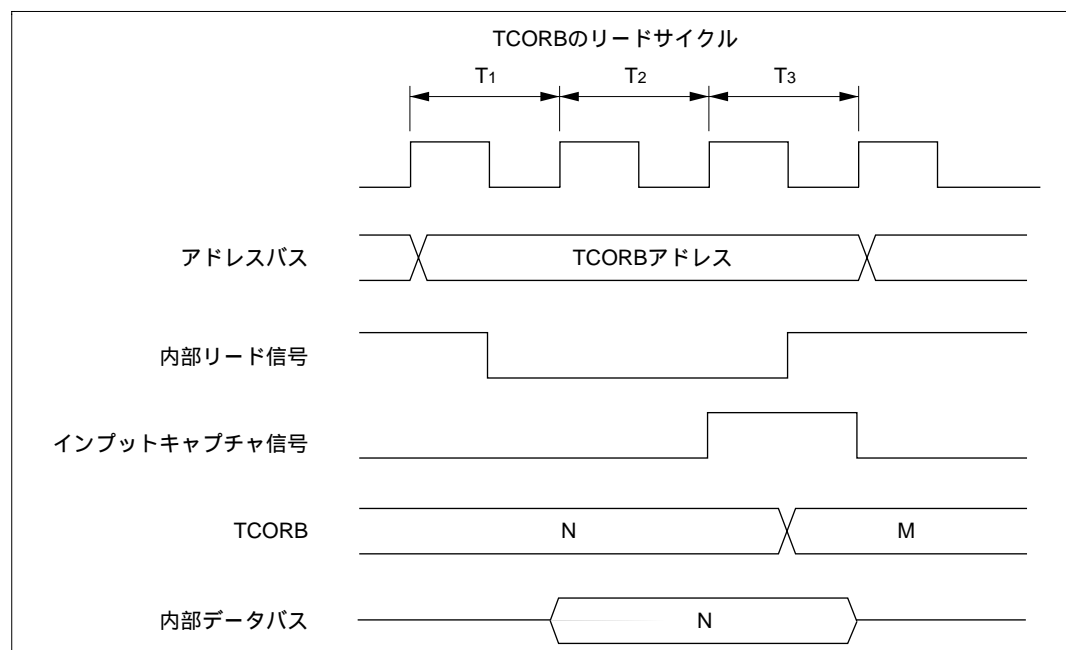


図 10.21 TCORB のリードとインプットキャプチャの競合

### 10.7.5 インプットキャプチャによるカウンタクリアとカウントアップの競合

インプットキャプチャ信号とカウントアップ信号が同時に発生すると、カウントアップされずインプットキャプチャによるカウンタクリアが優先されます。TCORB にはカウンタクリア前の 8TCNT の内容が転送されます。

このタイミング図を図 10.22 に示します。



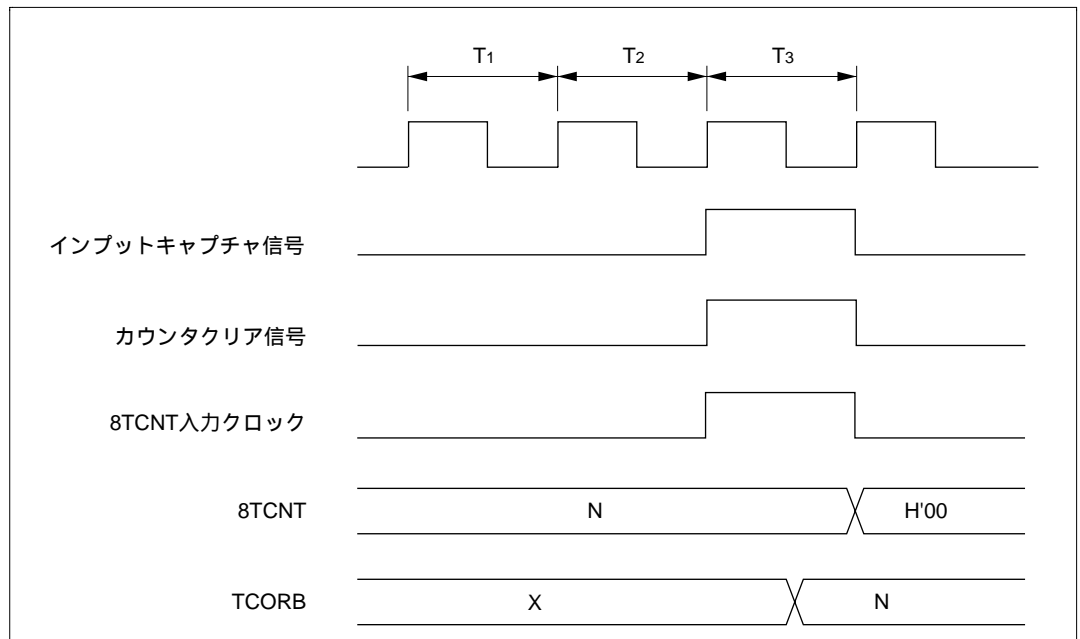


図 10.22 インพุットキャプチャによるカウンタクリアとカウントアップの競合

### 10.7.6 TCOR のライトとインพุットキャプチャの競合

TCOR のライトサイクル中の  $T_3$  ステートで、インพุットキャプチャ信号が発生すると、TCOR への書込みは行われず、インพุットキャプチャが優先されます。

このタイミング図を図 10.23 に示します。

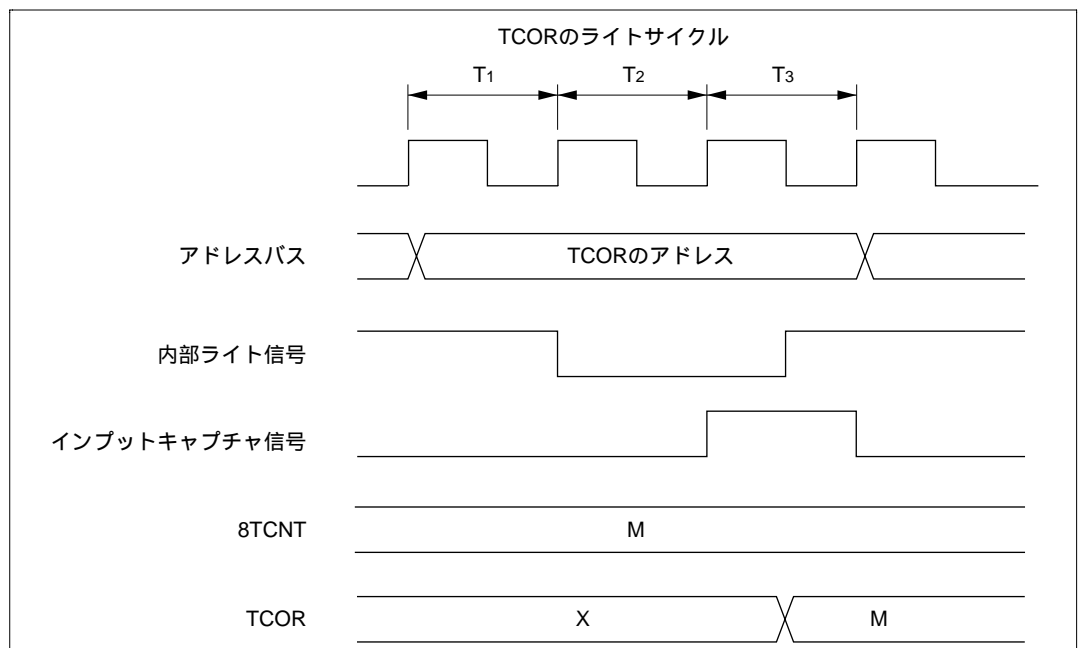


図 10.23 TCOR とインพุットキャプチャの競合

### 10.7.7 16ビットカウントモード（カスケード接続時）の8TCNTのバイトライトとカウントアップの競合

16ビットカウントモードで8TCNTのバイトライトサイクル中の $T_2$ ステートまたは $T_3$ ステートでカウントアップが発生しても、ライトを行った側のバイトデータはカウントアップされず、カウンタライトが優先されます。ライトを行わなかった側のバイトデータはカウントアップされます。

このタイミングを図10.24に示します。

8TCNTHのバイトライトサイクル中の $T_2$ ステートでカウントアップが発生した場合の例です。

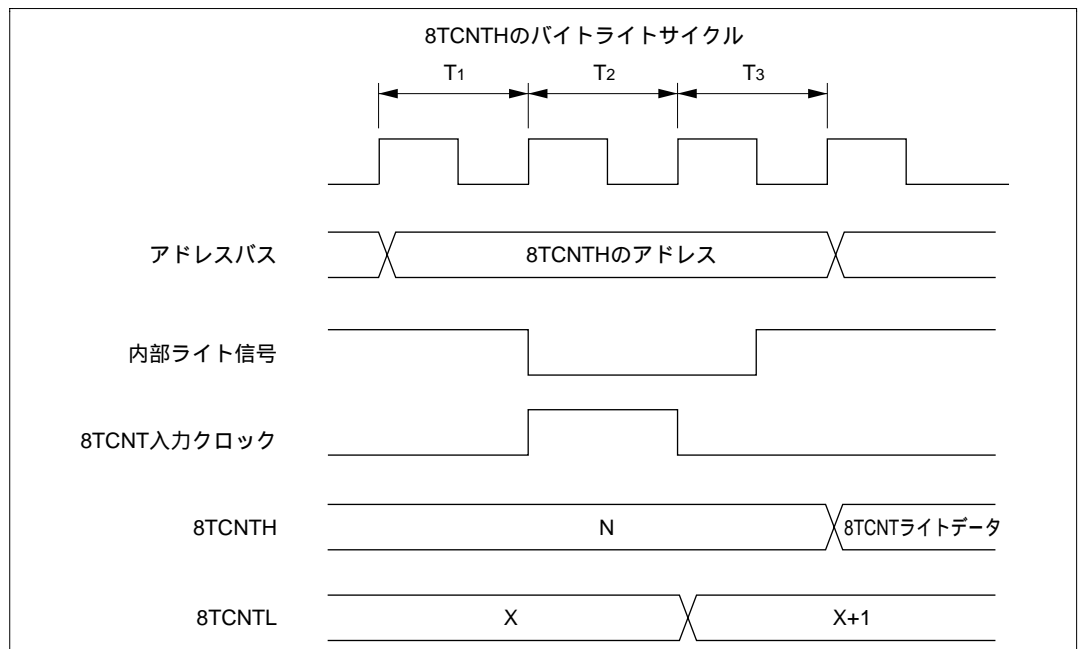


図 10.24 16ビットカウントモード時の8TCNTのバイトライトとカウントアップの競合

### 10.7.8 コンペアマッチ A、B の競合

コンペアマッチ A、B が同時に発生すると、コンペアマッチ A に対して設定されている出力状態と、コンペアマッチ B に対して設定されている出力状態のうち、表 10.7 に示すタイマ出力の優先順位に従って動作します。

表 10.7 タイマ出力の優先順位

出力設定	優先順位
トグル出力	高 ↑ 低
1 出力	
0 出力	
変化しない	

### 10.7.9 内部クロックの切り替えと8TCNTの動作

内部クロックを切り替えるタイミングによっては、8TCNT がカウントアップされてしまう場合があります。内部クロックの切り替えタイミング（CKS1、CKS0ビットの書き換え）と8TCNT動作の関係を表10.8に示します。

内部クロックから8TCNTクロックを生成する場合、内部クロックの立ち上がりエッジで検出しています。そのため表10.8のNo.3のように、Low Highレベルになるようなクロックの切り替えを行うと、切り替えタイミングを立ち下がりエッジとみなして8TCNTクロックが発生し、8TCNTがカウントアップされてしまいます。

また、内部クロックと外部クロックを切り替えるときも、8TCNT がカウントアップされることがあります。

表 10.8 内部クロックの切り替えと8TCNTの動作

No.	CKS1、CKS0 ビット 書き換えタイミング	8TCNTクロックの動作
1	High High レベル* <sup>1</sup> の切り替え	<p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>8TCNTクロック</p> <p>8TCNT</p> <p>CKSビット書き換え</p>
2	High Low レベル* <sup>2</sup> の切り替え	<p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>8TCNTクロック</p> <p>8TCNT</p> <p>CKSビット書き換え</p>

No.	CKS1、CKS0 ビット 書き換えタイミング	8TCNT クロックの動作
3	Low High レベル* <sup>3</sup> の切り替え	<p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>8TCNT クロック</p> <p>8TCNT</p> <p>CKSビット書き換え</p>
4	Low Low レベル の切り替え	<p>切り替え前のクロック</p> <p>切り替え後のクロック</p> <p>8TCNT クロック</p> <p>8TCNT</p> <p>CKSビット書き換え</p>

【注】 \*1 High レベル 停止、および停止 High レベルの場合を含みます。

\*2 停止 Low レベルの場合を含みます。

\*3 Low レベル 停止を含みます。

\*4 切り替えのタイミングを立ち上がりエッジとみなすために発生し、8TCNT はカウントアップされてしまいます。



---

# 11. プログラマブル タイミングパターン コントローラ (TPC)

---

## 第 11 章 目次

11.1	概要	469
11.1.1	特長	469
11.1.2	ブロック図	470
11.1.3	端子構成	471
11.1.4	レジスタ構成	472
11.2	各レジスタの説明	473
11.2.1	ポート A データディレクションレジスタ (PADDDR)	473
11.2.2	ポート A データレジスタ (PADR)	473
11.2.3	ポート B データディレクションレジスタ (PBDDR)	474
11.2.4	ポート B データレジスタ (PBDR)	474
11.2.5	ネクストデータレジスタ A (NDRA)	475
11.2.6	ネクストデータレジスタ B (NDRB)	477
11.2.7	ネクストデータイネーブルレジスタ A (NDERA)	479
11.2.8	ネクストデータイネーブルレジスタ B (NDERB)	480
11.2.9	TPC 出力コントロールレジスタ (TPCR)	481
11.2.10	TPC 出力モードレジスタ (TPMR)	484
11.3	動作説明	487
11.3.1	概要	487
11.3.2	出力タイミング	488
11.3.3	TPC 出力通常動作	489
11.3.4	TPC 出力ノンオーバーラップ動作	491
11.3.5	インプットキャプチャによる TPC 出力	493
11.4	使用上の注意	494

## 11. プログラマブルタイミングパターンコントローラ (TPC)

---

11.4.1	TPC 出力端子の動作.....	494
11.4.2	ノンオーバーラップ動作時の注意.....	494

## 11.1 概要

本 LSI は、16 ビットタイマをタイムベースとしてパルス出力を行うプログラマブルタイミングパターンコントローラ (TPC) を内蔵しています。TPC は 4 ビット単位の TPC 出力グループ 3~0 から構成されており、これらは同時に動作させることも、独立に動作させることも可能です。

### 11.1.1 特長

TPC の特長を以下に示します。

#### 出力データ 16 ビット

最大 16 ビットのデータ出力が可能で、TPC 出力をビット単位に許可することができます。

#### 4 系統の出力可能

4 ビット単位のグループで出力トリガ信号が選択可能で、最大 4 ビット × 4 系統の出力を行うことができます。

#### 出力トリガ信号を選択可能

16 ビットタイマの 3 チャンネルのコンペアマッチ信号の中から、グループごとに出カトリガ信号を選択することができます。

#### ノンオーバーラップ動作

複数のパルス出力の間のノンオーバーラップ期間を設定することができます。

#### DMA コントローラ (DMAC) との連携動作可能

出力トリガ信号に選択したコンペアマッチ信号で DMAC を起動することにより、CPU の介在なくデータを順次出力することができます。



### 11.1.2 ブロック図

TPCのブロック図を図11.1に示します。

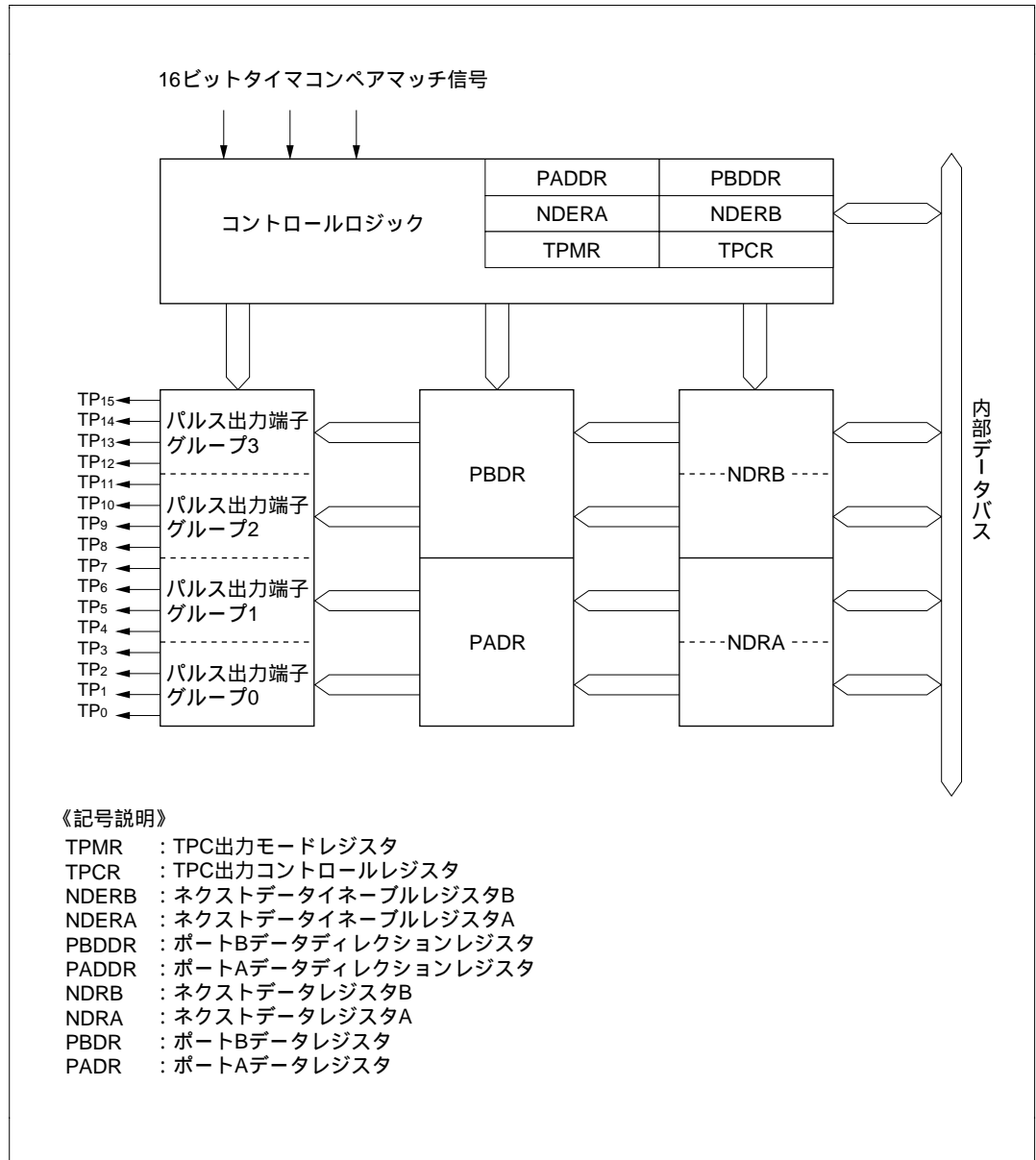


図 11.1 TPCのブロック図

## 11.1.3 端子構成

TPCの端子構成を表11.1に示します。

表 11.1 端子構成

名称	略称	入出力	機能
TPC 出力 0	TP <sub>0</sub>	出力	グループ 0 のパルス出力
TPC 出力 1	TP <sub>1</sub>	出力	
TPC 出力 2	TP <sub>2</sub>	出力	
TPC 出力 3	TP <sub>3</sub>	出力	
TPC 出力 4	TP <sub>4</sub>	出力	グループ 1 のパルス出力
TPC 出力 5	TP <sub>5</sub>	出力	
TPC 出力 6	TP <sub>6</sub>	出力	
TPC 出力 7	TP <sub>7</sub>	出力	
TPC 出力 8	TP <sub>8</sub>	出力	グループ 2 のパルス出力
TPC 出力 9	TP <sub>9</sub>	出力	
TPC 出力 10	TP <sub>10</sub>	出力	
TPC 出力 11	TP <sub>11</sub>	出力	
TPC 出力 12	TP <sub>12</sub>	出力	グループ 3 のパルス出力
TPC 出力 13	TP <sub>13</sub>	出力	
TPC 出力 14	TP <sub>14</sub>	出力	
TPC 出力 15	TP <sub>15</sub>	出力	

## 11.1.4 レジスタ構成

TPC のレジスタ構成を表 11.2 に示します。

表 11.2 レジスタ構成

アドレス* <sup>1</sup>	名称	略称	R/W	初期値
H'EE009	ポート A データディレクションレジスタ	PADDR	W	H'00
H'FFFD9	ポート A データレジスタ	PADR	R/(W)* <sup>2</sup>	H'00
H'EE00A	ポート B データディレクションレジスタ	PBDDR	W	H'00
H'FFFDA	ポート B データレジスタ	PBDR	R/(W)* <sup>2</sup>	H'00
H'FFFA0	TPC 出力モードレジスタ	TPMR	R/W	H'F0
H'FFFA1	TPC 出力コントロールレジスタ	TPCR	R/W	H'FF
H'FFFA2	ネクストデータインーブルレジスタ B	NDERB	R/W	H'00
H'FFFA3	ネクストデータインーブルレジスタ A	NDERA	R/W	H'00
H'FFFA5/ H'FFFA7* <sup>3</sup>	ネクストデータレジスタ A	NDRA	R/W	H'00
H'FFFA4/ H'FFFA6* <sup>3</sup>	ネクストデータレジスタ B	NDRB	R/W	H'00

【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。

\*2 TPC 出力として使用しているビットは、ライトできません。

\*3 TPCR の設定により TPC 出力グループ 0 と TPC 出力グループ 1 の出力トリガが同一の場合は NDRA のアドレスは H'FFFA5 となり、出力トリガが異なる場合はグループ 0 に対応する NDRA のアドレスは H'FFFA7、グループ 1 に対応する NDRA のアドレスは H'FFFA5 となります。

同様に、TPCR の設定により TPC 出力グループ 2 と TPC 出力グループ 3 の出力トリガが同一の場合は NDRB のアドレスは H'FFFA4 となり出力トリガが異なる場合はグループ 2 に対応する NDRB のアドレスは H'FFFA6、グループ 3 に対応する NDRB のアドレスは H'FFFA4 となります。

## 11.2 各レジスタの説明

### 11.2.1 ポート A データディレクションレジスタ (PADDR)

PADDR は 8 ビットのライト専用のレジスタで、ポート A の各端子の入出力方向をビット単位に設定します。

ビット :	7	6	5	4	3	2	1	0
	PA7DDR	PA6DDR	PA5DDR	PA4DDR	PA3DDR	PA2DDR	PA1DDR	PA0DDR
初期値 :	0	0	0	0	0	0	0	0
R/W :	W	W	W	W	W	W	W	W

ポートAデータディレクション7~0

ポートAの各端子の入出力を選択するビットです。

ポート A は TP<sub>7</sub> ~ TP<sub>0</sub> 端子との兼用端子となっています。TPC 出力を行う端子に対応するビットは 1 にセットしてください。

PADDR の詳細は、「8.11 ポート A」を参照してください。

### 11.2.2 ポート A データレジスタ (PADR)

PADR は 8 ビットのリード/ライト可能なレジスタで、TPC 出力を使用する場合、グループ 0、1 の出力データを格納します。

ビット :	7	6	5	4	3	2	1	0
	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

ポートAデータ7~0

TPC出力グループ0、1の出力データを格納するビットです。

【注】\* NDERA により、TPC 出力に設定されたビットはリード専用となります。

PADR の詳細は、「8.11 ポート A」を参照してください。

### 11.2.3 ポート B データディレクションレジスタ (PBDDR)

PBDDR は 8 ビットのライト専用のレジスタで、ポート B の各端子の入出力方向をビット単位に設定します。

ビット :	7	6	5	4	3	2	1	0
	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR	PB0DDR
初期値 :	0	0	0	0	0	0	0	0
R/W :	W	W	W	W	W	W	W	W

ポートBデータディレクション7~0

ポートBの各端子の入出力を選択するビットです。

ポート B は TP<sub>15</sub> ~ TP<sub>8</sub> 端子との兼用端子となっています。TPC 出力を行う端子に対応するビットは 1 にセットしてください。

PBDDR の詳細は、「8.12 ポート B」を参照してください。

### 11.2.4 ポート B データレジスタ (PBDR)

PBDR は 8 ビットのリード/ライト可能なレジスタで、TPC 出力を使用する場合、PBDR はグループ 2、3 の出力データを格納します。

ビット :	7	6	5	4	3	2	1	0
	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

ポートBデータ7~0

TPC出力グループ2、3の出力データを格納するビットです。

【注】\* NDERB により、TPC 出力に設定されたビットはリード専用となります。

PBDR の詳細は、「8.12 ポート B」を参照してください。

### 11.2.5 ネクストデータレジスタ A (NDRA)

NDRA は 8 ビットのリード/ライト可能なレジスタで、TPC 出力グループ 1、0 (TP<sub>7</sub> ~ TP<sub>0</sub> 端子) の次の出力データを格納します。TPC 出力を行う場合、TPCR で指定した 16 ビットタイマのコンペアマッチが発生したときに、NDRA の内容が PADR の対応するビットに転送されます。

NDRA のアドレスは、TPC 出力グループ 0、1 の出力トリガを同一に設定した場合と異なるように設定した場合とで異なります。

NDRA はリセット、またはハードウェアスタンバイモード時に H'00 にイニシャライズされます。

ソフトウェアスタンバイモード時にはイニシャライズされません。

#### (1) TPC 出力グループ 0、1 の出力トリガが同一の場合

TPC 出力グループ 0、1 の出力トリガとなるコンペアマッチを同一にすると、NDRA のアドレスは H'FFFA5 となります。グループ 1、0 はそれぞれ上位 4 ビット、下位 4 ビットになります。このとき、アドレス H'FFFA7 はすべてリザーブビットとなります。リザーブビットはリードすると常に 1 が読み出され、ライトは無効です。

##### (a) アドレス : H'FFFA5

ビット :	7	6	5	4	3	2	1	0
	NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ネクストデータ7~4	ネクストデータ3~0
TPC出力グループ1の次の	TPC出力グループ0の次の
出力データを格納するビットです。	出力データを格納するビットです。

##### (b) アドレス : H'FFFA7

ビット :	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-
初期値 :	1	1	1	1	1	1	1	1
R/W :	-	-	-	-	-	-	-	-

リザーブビット

(2) TPC 出力グループ 0、1 の出力トリガが異なる場合

TPC 出力グループ 0、1 の出力トリガとなるコンペアマッチを別にする、NDRA の上位 4 ビット (グループ 1) のアドレスは H'FFFA5、NDRA の下位 4 ビット (グループ 0) のアドレスは H'FFFA7 となります。このとき、アドレス H'FFFA5 のビット 3~0、アドレス H'FFFA7 のビット 7~4 はリザーブビットとなります。リザーブビットはリードすると常に 1 が読み出され、ライトは無効です。

(a) アドレス : H'FFFA5

ビット :	7	6	5	4	3	2	1	0
	NDR7	NDR6	NDR5	NDR4	-	-	-	-
初期値 :	0	0	0	0	1	1	1	1
R/W :	R/W	R/W	R/W	R/W	-	-	-	-

ネクストデータ7~4
リザーブビット

TPC出力グループ1の次の  
 出力データを格納するビットです。

(b) アドレス : H'FFFA7

ビット :	7	6	5	4	3	2	1	0
	-	-	-	-	NDR3	NDR2	NDR1	NDR0
初期値 :	1	1	1	1	0	0	0	0
R/W :	-	-	-	-	R/W	R/W	R/W	R/W

リザーブビット
ネクストデータ3~0

TPC出力グループ0の次の  
 出力データを格納するビットです。

### 11.2.6 ネクストデータレジスタ B (NDRB)

NDRB は 8 ビットのリード/ライト可能なレジスタで、TPC 出力グループ 3、2 (TP<sub>15</sub> ~ TP<sub>8</sub> 端子) の次の出力データを格納します。TPC 出力を行う場合、TPCR で指定した 16 ビットタイマのコンペアマッチが発生したときに、NDRB の内容が PBDR の対応するビットに転送されます。NDRB のアドレスは、TPC 出力グループ 2、3 の出力トリガを同一に設定した場合と異なるように設定した場合とで異なります。

NDRB はリセット、またはハードウェアスタンバイモード時に H'00 にイニシャライズされます。

ソフトウェアスタンバイモード時にはイニシャライズされません。

#### (1) TPC 出力グループ 2、3 の出力トリガが同一の場合

TPC 出力グループ 2、3 の出力トリガとなるコンペアマッチを同一にすると、NDRB のアドレスは H'FFFA4 となります。グループ 3、2 はそれぞれ下位 4 ビット、上位 4 ビットになります。このとき、アドレス H'FFFA6 はすべてリザーブビットとなります。リザーブビットはリードすると常に 1 が読み出され、ライトは無効です。

(a) アドレス : H'FFFA4

ビット :	7	6	5	4	3	2	1	0
	NDR15	NDR14	NDR13	NDR12	NDR11	NDR10	NDR9	NDR8
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ネクストデータ15~12 TPC出力グループ3の次の 出力データを格納するビットです。	ネクストデータ11~8 TPC出力グループ2の次の 出力データを格納するビットです。
---	--

(b) アドレス : H'FFFA6

ビット :	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-
初期値 :	1	1	1	1	1	1	1	1
R/W :	-	-	-	-	-	-	-	-

リザーブビット
---------



(2) TPC 出力グループ 2、3 の出力トリガが異なる場合

TPC 出力グループ 2、3 の出力トリガとなるコンペアマッチを別にする、NDRB の上位 4 ビット (グループ 3) のアドレスは H'FFFA4、NDRB の下位 4 ビット (グループ 2) のアドレスは H'FFFA6 となります。このとき、アドレス H'FFFA4 のビット 3~0、アドレス H'FFFA6 のビット 7~4 はリザーブビットとなります。リザーブビットはリードすると常に 1 が読み出され、ライトは無効です。

(a) アドレス : H'FFFA4

ビット :	7	6	5	4	3	2	1	0
	NDR15	NDR14	NDR13	NDR12	-	-	-	-
初期値 :	0	0	0	0	1	1	1	1
R/W :	R/W	R/W	R/W	R/W	-	-	-	-

ネクストデータ15~12	リザーブビット
TPC出力グループ3の次の出力データを格納するビットです。	

(b) アドレス : H'FFFA6

ビット :	7	6	5	4	3	2	1	0
	-	-	-	-	NDR11	NDR10	NDR9	NDR8
初期値 :	1	1	1	1	0	0	0	0
R/W :	-	-	-	-	R/W	R/W	R/W	R/W

リザーブビット	ネクストデータ11~8
TPC出力グループ2の次の出力データを格納するビットです。	

### 11.2.7 ネクストデータイネーブルレジスタ A (NDERA)

NDERA は 8 ビットのリード/ライト可能なレジスタで、TPC 出力グループ 1、0 (TP<sub>7</sub> ~ TP<sub>0</sub> 端子) の許可/禁止をビット単位で選択します。

ビット :	7	6	5	4	3	2	1	0
	NDER7	NDER6	NDER5	NDER4	NDER3	NDER2	NDER1	NDER0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ネクストデータイネーブル7~0  
 TPC出力グループ1、0を許可/禁止を  
 選択するビットです。

NDERA により TPC 出力が許可されたビットは、TPCR で選択された 16 ビットタイマのコンペアマッチが発生すると、NDRA の値が PADR の当該ビットに自動転送され出力値が更新されます。TPC 出力を禁止されているビットについては、NDRA から PADR への転送は行われず出力値も変化しません。

NDERA はリセット、またはハードウェアスタンバイモード時に H'00 にイニシャライズされます。ソフトウェアスタンバイモード時にはイニシャライズされません。

ビット7~0 : ネクストデータイネーブル7~0 (NDER7 ~ NDER0)

TPC 出力グループ 1、0 (TP<sub>7</sub> ~ TP<sub>0</sub> 端子) の許可/禁止をビット単位で選択します。

ビット7~0	説明
NDER7 ~ NDER0	
0	TPC 出力 TP <sub>7</sub> ~ TP <sub>0</sub> を禁止 (NDR7 ~ NDR0 から PA <sub>7</sub> ~ PA <sub>0</sub> への転送禁止) (初期値)
1	TPC 出力 TP <sub>7</sub> ~ TP <sub>0</sub> を許可 (NDR7 ~ NDR0 から PA <sub>7</sub> ~ PA <sub>0</sub> への転送許可)

### 11.2.8 ネクストデータイネーブルレジスタ B (NDERB)

NDERB は 8 ビットのリード/ライト可能なレジスタで、TPC 出力グループ 3、2 (TP<sub>15</sub> ~ TP<sub>8</sub> 端子) の許可/禁止をビット単位で選択します。

ビット :	7	6	5	4	3	2	1	0
	NDER15	NDER14	NDER13	NDER12	NDER11	NDER10	NDER9	NDER8
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ネクストデータイネーブル15~8  
 TPC出力グループ3、2を許可/禁止を  
 選択するビットです。

NDERB により TPC 出力が許可されたビットは、TPCR で選択された 16 ビットタイマのコンペアマッチが発生すると、NDERB の値が PBDR の当該ビットに自動転送され出力値が更新されます。 TPC 出力を禁止されているビットについては、NDERB から PBDR への転送は行われず出力値も変化しません。

NDERB はリセット、またはハードウェアスタンバイモード時に H'00 にイニシャライズされます。ソフトウェアスタンバイモード時にはイニシャライズされません。

ビット 7~0 : ネクストデータイネーブル 15~8 (NDER15 ~ NDER8)

TPC 出力グループ 3、2 (TP<sub>15</sub> ~ TP<sub>8</sub> 端子) の許可/禁止をビット単位で選択します。

ビット 7~0	説明
NDER15 ~ NDER8	
0	TPC 出力 TP <sub>15</sub> ~ TP <sub>8</sub> を禁止 (NDER15 ~ NDR8 から PB <sub>7</sub> ~ PB <sub>0</sub> への転送禁止) (初期値)
1	TPC 出力 TP <sub>15</sub> ~ TP <sub>8</sub> を許可 (NDER15 ~ NDR8 から PB <sub>7</sub> ~ PB <sub>0</sub> への転送許可)

## 11.2.9 TPC 出力コントロールレジスタ (TPCR)

TPCR は 8 ビットのリード/ライト可能なレジスタで、TPC 出力の出力トリガ信号をグループ単位で選択します。

ビット:	7	6	5	4	3	2	1	0
	G3CMS1	G3CMS0	G2CMS1	G2CMS0	G1CMS1	G1CMS0	G0CMS1	G0CMS0
初期値:	1	1	1	1	1	1	1	1
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

グループ0コンペアマッチセレクト1、0  
TPC出力グループ0 (TP<sub>3</sub>-TP<sub>0</sub>) の出力トリガとなるコンペアマッチを選択するビットです。

グループ1コンペアマッチセレクト1、0  
TPC出力グループ1 (TP<sub>7</sub>-TP<sub>4</sub>) の出力トリガとなるコンペアマッチを選択するビットです。

グループ2コンペアマッチセレクト1、0  
TPC出力グループ2 (TP<sub>11</sub>-TP<sub>8</sub>) の出力トリガとなるコンペアマッチを選択するビットです。

グループ3コンペアマッチセレクト1、0  
TPC出力グループ3 (TP<sub>15</sub>-TP<sub>12</sub>) の出力トリガとなるコンペアマッチを選択するビットです。

TPCR は、リセットまたはハードウェアスタンバイモード時に H'FF にイニシャライズされます。ソフトウェアスタンバイモード時にはイニシャライズされません。

ビット7、6：グループ3 コンペアマッチセレクト 1、0 (G3CMS1、G3CMS0)

TPC 出力グループ 3 (TP<sub>15</sub> ~ TP<sub>12</sub> 端子) の出力トリガとなるコンペアマッチを選択します。

ビット7	ビット6	説明
G3CMS1	G3CMS0	
0	0	TPC 出力グループ 3 (TP <sub>15</sub> ~ TP <sub>12</sub> 端子) の出力トリガは、16 ビットタイマチャネル 0 のコンペアマッチ
	1	TPC 出力グループ 3 (TP <sub>15</sub> ~ TP <sub>12</sub> 端子) の出力トリガは、16 ビットタイマチャネル 1 のコンペアマッチ
1	0	TPC 出力グループ 3 (TP <sub>15</sub> ~ TP <sub>12</sub> 端子) の出力トリガは、16 ビットタイマチャネル 2 のコンペアマッチ
	1	TPC 出力グループ 3 (TP <sub>15</sub> ~ TP <sub>12</sub> 端子) の出力トリガは、16 ビットタイマチャネル 2 のコンペアマッチ (初期値)

ビット5、4：グループ2 コンペアマッチセレクト 1、0 (G2CMS1、G2CMS0)

TPC 出力グループ 2 (TP<sub>11</sub> ~ TP<sub>8</sub> 端子) の出力トリガとなるコンペアマッチを選択します。

ビット5	ビット4	説明
G2CMS1	G2CMS0	
0	0	TPC 出力グループ 2 (TP <sub>11</sub> ~ TP <sub>8</sub> 端子) の出力トリガは、16 ビットタイマチャネル 0 のコンペアマッチ
	1	TPC 出力グループ 2 (TP <sub>11</sub> ~ TP <sub>8</sub> 端子) の出力トリガは、16 ビットタイマチャネル 1 のコンペアマッチ
1	0	TPC 出力グループ 2 (TP <sub>11</sub> ~ TP <sub>8</sub> 端子) の出力トリガは、16 ビットタイマチャネル 2 のコンペアマッチ
	1	TPC 出力グループ 2 (TP <sub>11</sub> ~ TP <sub>8</sub> 端子) の出力トリガは、16 ビットタイマチャネル 2 のコンペアマッチ (初期値)

ビット3、2：グループ1コンペアマッチセレクト1、0 (G1CMS1、G1CMS0)

TPC 出力グループ 1 (TP<sub>7</sub> ~ TP<sub>4</sub> 端子) の出力トリガとなるコンペアマッチを選択します。

ビット3	ビット2	説明
G1CMS1	G1CMS0	
0	0	TPC 出力グループ 1 (TP <sub>7</sub> ~ TP <sub>4</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 0 のコンペアマッチ
	1	TPC 出力グループ 1 (TP <sub>7</sub> ~ TP <sub>4</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 1 のコンペアマッチ
1	0	TPC 出力グループ 1 (TP <sub>7</sub> ~ TP <sub>4</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 2 のコンペアマッチ
	1	TPC 出力グループ 1 (TP <sub>7</sub> ~ TP <sub>4</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 2 のコンペアマッチ (初期値)

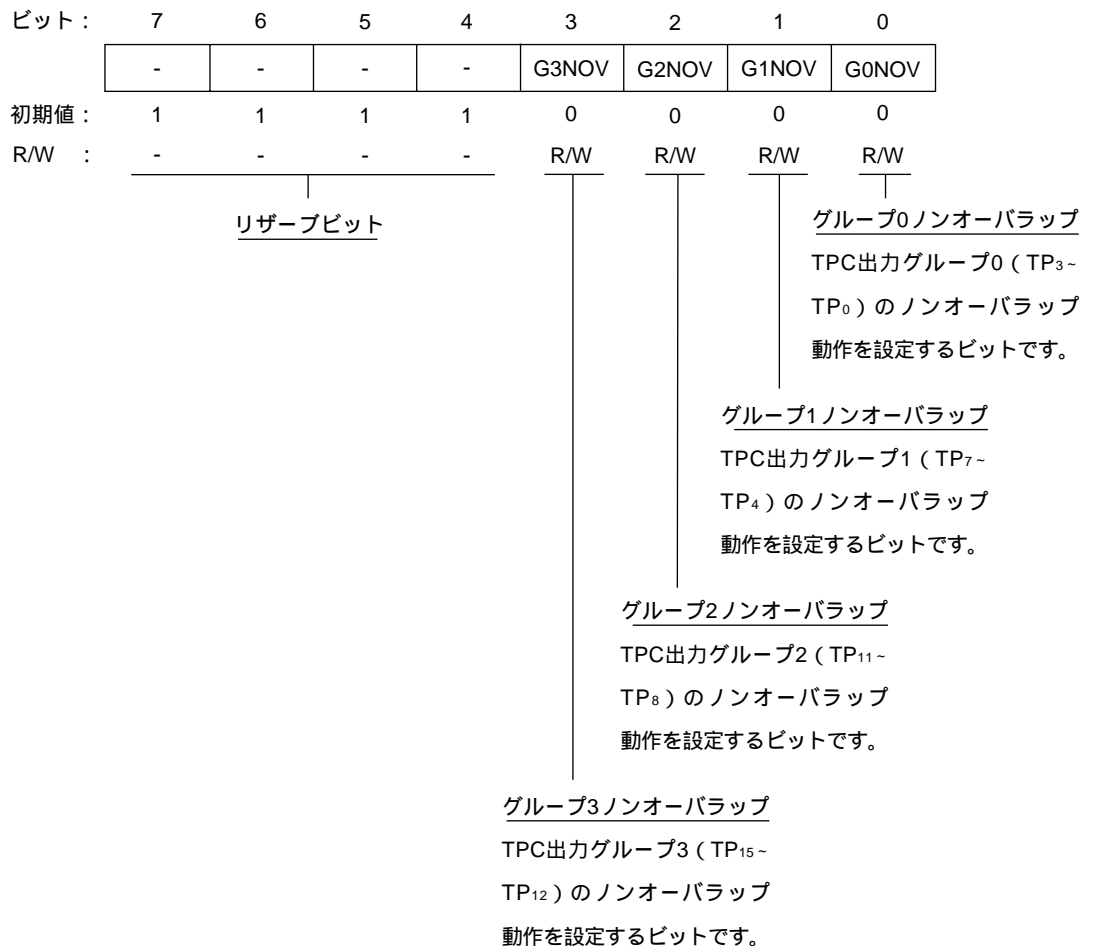
ビット1、0：グループ0コンペアマッチセレクト1、0 (G0CMS1、G0CMS0)

TPC 出力グループ 0 (TP<sub>3</sub> ~ TP<sub>0</sub> 端子) の出力トリガとなるコンペアマッチを選択します。

ビット1	ビット0	説明
G0CMS1	G0CMS0	
0	0	TPC 出力グループ 0 (TP <sub>3</sub> ~ TP <sub>0</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 0 のコンペアマッチ
	1	TPC 出力グループ 0 (TP <sub>3</sub> ~ TP <sub>0</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 1 のコンペアマッチ
1	0	TPC 出力グループ 0 (TP <sub>3</sub> ~ TP <sub>0</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 2 のコンペアマッチ
	1	TPC 出力グループ 0 (TP <sub>3</sub> ~ TP <sub>0</sub> 端子) の出力トリガは、16 ビットタイマチャンネル 2 のコンペアマッチ (初期値)

### 11.2.10 TPC 出力モードレジスタ (TPMR)

TPMR は 8 ビットのリード/ライト可能なレジスタで、TPC 出力を通常動作で行うか、ノンオーバーラップ動作で行うかをグループ単位で指定します。



ノンオーバーラップ動作の TPC 出力は、出力トリガとなる 16 ビットタイマの GRB に出  
力波形の周期を、また GRA にノンオーバーラップ期間を設定し、コンペアマッチ A、B で  
出力値を変化させます。

詳細は、「11.3.4 TPC 出力ノンオーバーラップ動作」を参照してください。

TPMR はリセット、またはハードウェアスタンバイモード時に H'F0 にイニシャライズ  
されます。

ソフトウェアスタンバイモード時にはイニシャライズされません。

#### ビット 7~4 : リザーブビット

リザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。

## ビット3 : グループ3 ノンオーバーラップ (G3NOV)

TPC 出力グループ 3 (TP<sub>15</sub> ~ TP<sub>12</sub> 端子) を通常動作させるか、ノンオーバーラップ動作させるかを選択します。

ビット3	説明
G3NOV	
0	TPC 出力グループ 3 は、通常動作 (選択された 16 ビットタイマのコンペアマッチ A で出力値を更新します。) (初期値)
1	TPC 出力グループ 3 は、ノンオーバーラップ動作 (選択された 16 ビットタイマのコンペアマッチ A、B により、1 出力、0 出力を独立に行うことができます。)

## ビット2 : グループ2 ノンオーバーラップ (G2NOV)

TPC 出力グループ 2 (TP<sub>11</sub> ~ TP<sub>8</sub> 端子) を通常動作させるか、ノンオーバーラップ動作させるかを選択します。

ビット2	説明
G2NOV	
0	TPC 出力グループ 2 は、通常動作 (選択された 16 ビットタイマのコンペアマッチ A で出力値を更新します。) (初期値)
1	TPC 出力グループ 2 は、ノンオーバーラップ動作 (選択された 16 ビットタイマのコンペアマッチ A、B により、1 出力、0 出力を独立に行うことができます。)

## ビット1 : グループ1 ノンオーバーラップ (G1NOV)

TPC 出力グループ 1 (TP<sub>7</sub> ~ TP<sub>4</sub> 端子) を通常動作させるか、ノンオーバーラップ動作させるかを選択します。

ビット1	説明
G1NOV	
0	TPC 出力グループ 1 は、通常動作 (選択された 16 ビットタイマのコンペアマッチ A で出力値を更新します。) (初期値)
1	TPC 出力グループ 1 は、ノンオーバーラップ動作 (選択された 16 ビットタイマのコンペアマッチ A、B により、1 出力、0 出力を独立に行うことができます。)



ビット0 : グループ0 ノンオーバーラップ (G0NOV)

TPC 出力グループ 0 (TP<sub>3</sub> ~ TP<sub>0</sub> 端子) を通常動作させるか、ノンオーバーラップ動作させるかを選択します。

ビット0	説明
G0NOV	
0	TPC 出力グループ 0 は、通常動作 (選択された 16 ビットタイマのコンペアマッチ A で出力値を更新します。) (初期値)
1	TPC 出力グループ 0 は、ノンオーバーラップ動作 (選択された 16 ビットタイマのコンペアマッチ A、B により、1 出力、0 出力を独立に行うことができます。)

## 11.3 動作説明

### 11.3.1 概要

TPC 出力は、PADDR、PBDDR と NDERA、NDERB の対応するビットをそれぞれ 1 にセットすることにより許可状態となります。この状態では、対応する PADR、PBDR の内容が出力されます。

その後、TPCR で指定したコンペアマッチが発生すると、ビットに対応する NDRA および NDRB の内容がそれぞれ PADR および PBDR に転送され、出力値が更新されます。

TPC 出力動作を図 11.2 に示します。また、TPC 動作条件を表 11.3 に示します。

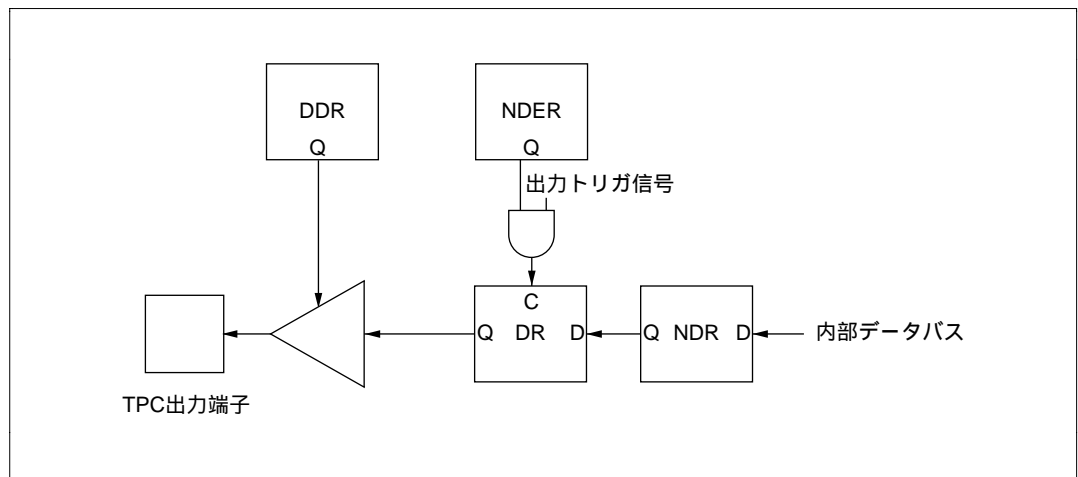


図 11.2 TPC 出力動作

表 11.3 TPC 動作条件

NDER	DDR	端子機能
0	0	入力ポート
	1	出力ポート
1	0	入力ポート (ただし、コンペアマッチ時に NDR から DR の転送を行い、DR へのライトはできません)
	1	TPC パルス出力

次のコンペアマッチが発生するまでに NDRA および NDRB に出力データを書き込むことにより、コンペアマッチごとに最大 16 ビットのデータを順次出力することができます。

ノンオーバーラップ動作については、「11.3.4 TPC 出力ノンオーバーラップ動作」を参照してください。

### 11.3.2 出力タイミング

TPC 出力許可状態で指定されたコンペアマッチが発生すると、NDRA/NDRB の内容が PADR/PBDR に転送され、出力されます。

このタイミングを図 11.3 に示します。

コンペアマッチ A により、グループ 2、3 で通常出力を行った場合の例です。

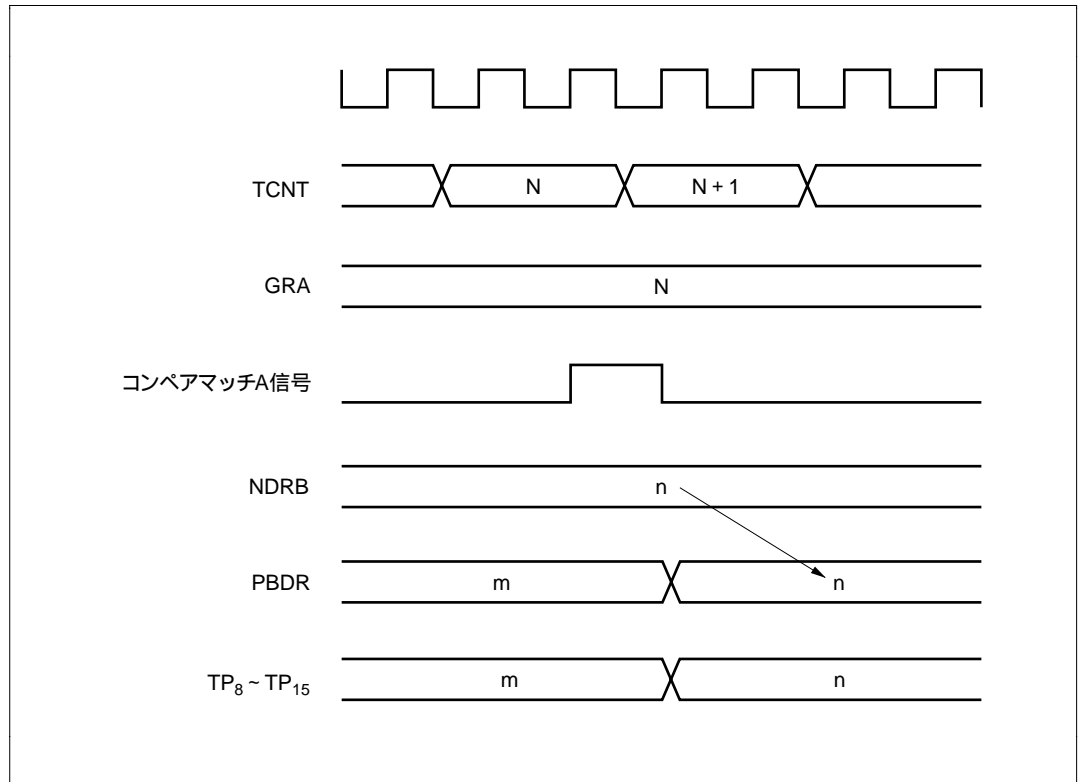


図 11.3 NDR の内容が転送・出力されるタイミング (例)

### 11.3.3 TPC 出力通常動作

#### (1) TPC 出力通常動作の設定手順例

TPC 出力通常動作の設定手順例を図 11.4 に示します。

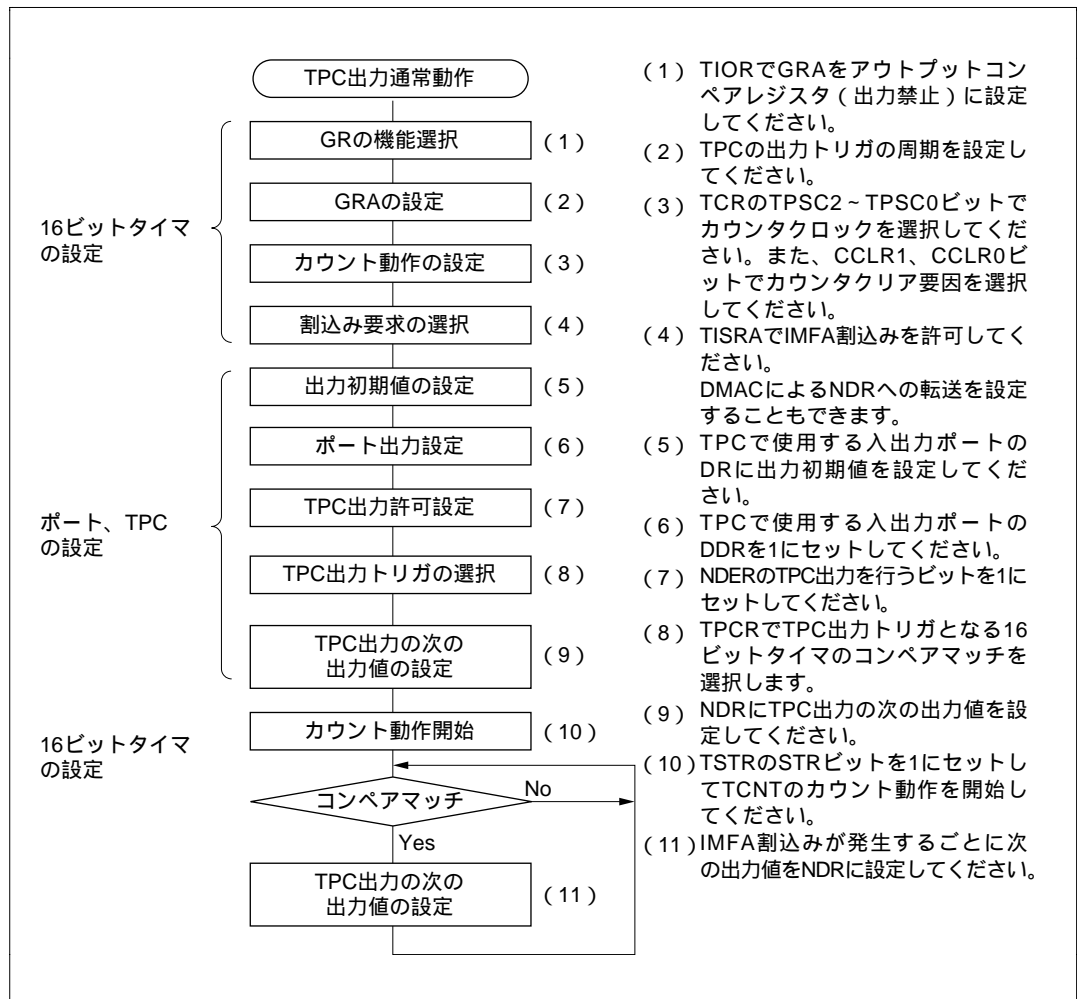
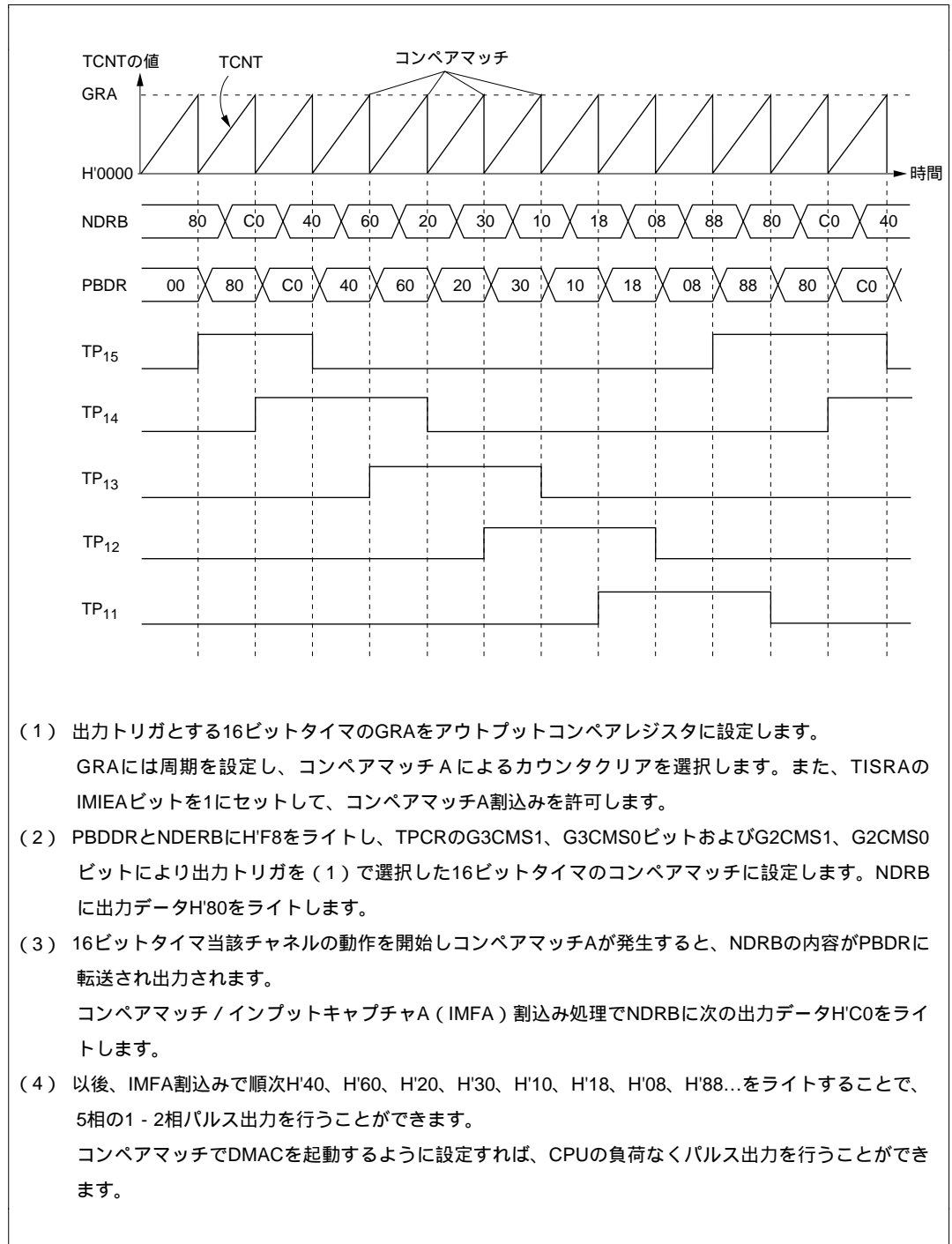


図 11.4 TPC 出力通常動作の設定手順例

(2) TPC 出力通常動作例 (5 相パルス出力例)

TPC 出力を使用して一定周期で 5 相パルスを出力させた例を図 11.5 に示します。



- (1) 出力トリガとする16ビットタイマのGRAをアウトプットコンペアレジスタに設定します。  
GRAには周期を設定し、コンペアマッチAによるカウンタクリアを選択します。また、TISRAのIMIEAビットを1にセットして、コンペアマッチA割込みを許可します。
- (2) PBDDRとNDRBにH'F8をライトし、TPCRのG3CMS1、G3CMS0ビットおよびG2CMS1、G2CMS0ビットにより出力トリガを(1)で選択した16ビットタイマのコンペアマッチに設定します。NDRBに出力データH'80をライトします。
- (3) 16ビットタイマ当該チャンネルの動作を開始しコンペアマッチAが発生すると、NDRBの内容がPBDRに転送され出力されます。  
コンペアマッチ/インプットキャプチャA(IMFA)割込み処理でNDRBに次の出力データH'C0をライトします。
- (4) 以後、IMFA割込みで順次H'40、H'60、H'20、H'30、H'10、H'18、H'08、H'88...をライトすることで、5相の1-2相パルス出力を行うことができます。  
コンペアマッチでDMACを起動するように設定すれば、CPUの負荷なくパルス出力を行うことができます。

図 11.5 TPC 出力通常動作例 (5 相パルス出力例)

### 11.3.4 TPC 出力ノンオーバーラップ動作

#### (1) TPC 出力ノンオーバーラップ動作の設定手順例

TPC 出力ノンオーバーラップ動作の設定手順例を図 11.6 に示します。

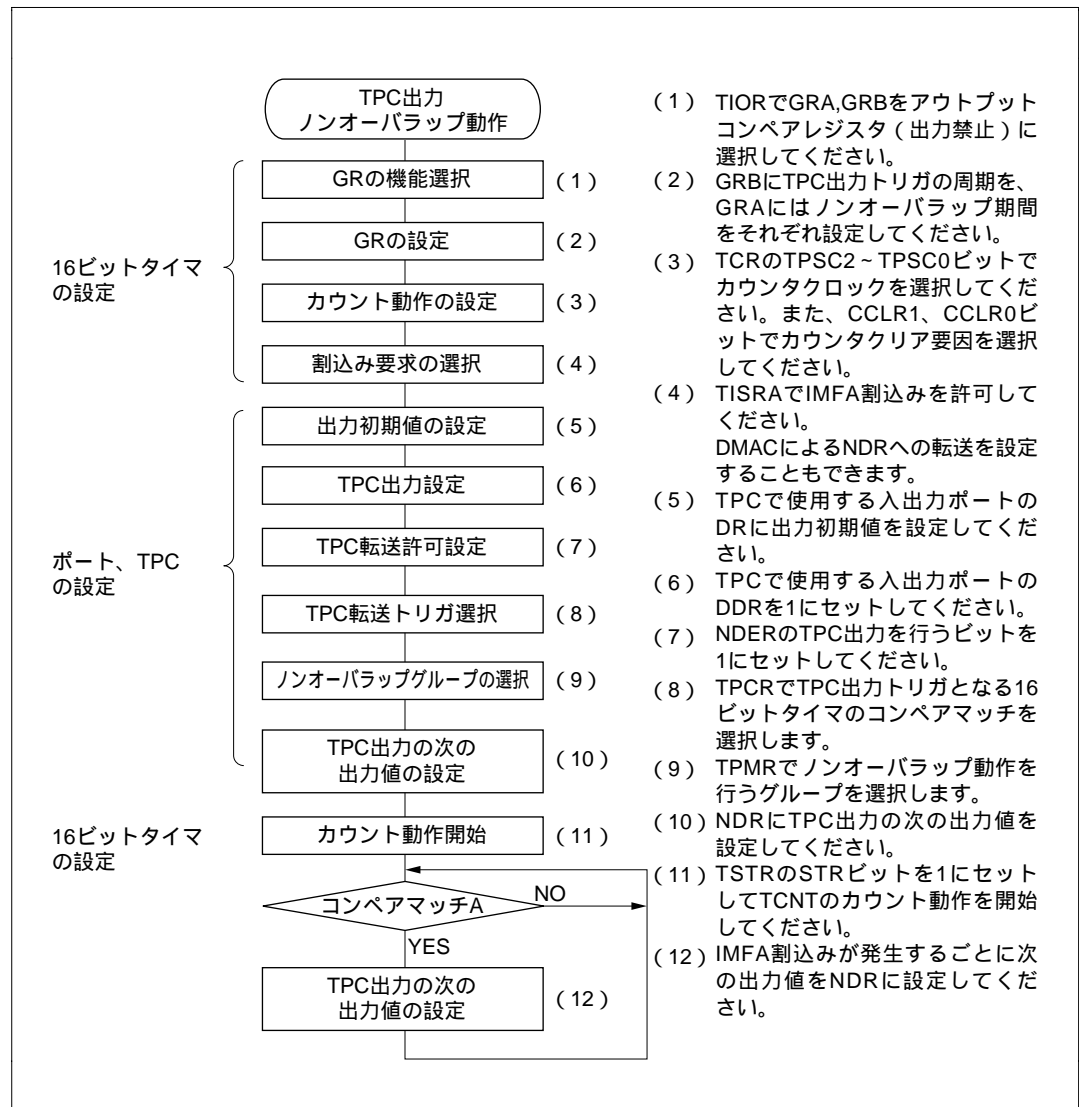


図 11.6 TPC 出力ノンオーバーラップ動作の設定手順例

(2) TPC 出力ノンオーバーラップ動作例 (4 相の相補ノンオーバーラップ出力例)

TPC 出力を使用して 4 相の相補ノンオーバーラップのパルスを出力させた例を図 11.7 に示します。

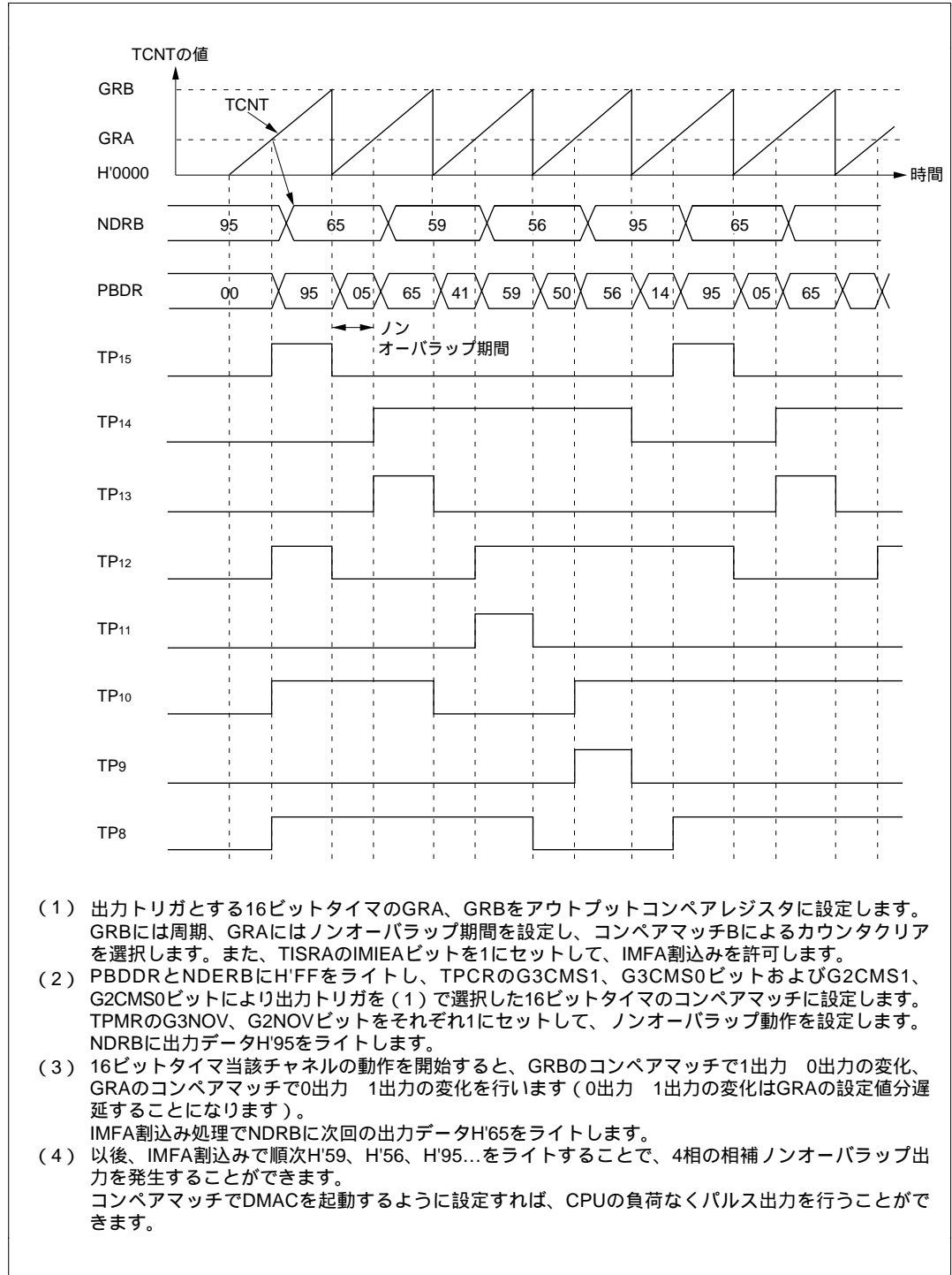


図 11.7 TPC 出力ノンオーバーラップ動作例 (4 相の相補ノンオーバーラップ出力例)

### 11.3.5 インพุットキャプチャによる TPC 出力

TPC 出力は、16 ビットタイマのコンペアマッチだけでなく、インพุットキャプチャによっても可能です。

TPCR によって選択された 16 ビットタイマの GRA がインพุットキャプチャレジスタとして機能しているとき、インพุットキャプチャ信号により TPC 出力を行います。

このタイミングを図 11.8 に示します。

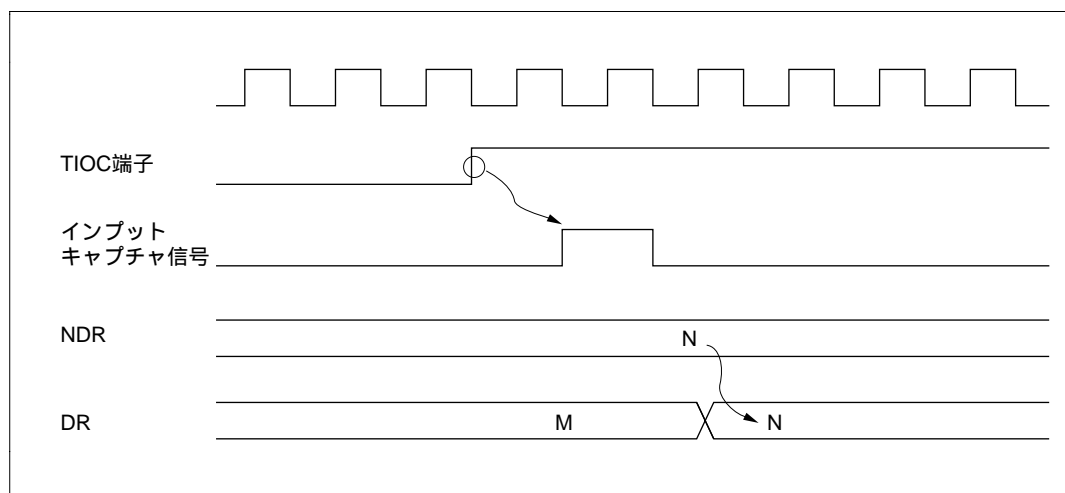


図 11.8 インพุットキャプチャによる TPC 出力例



## 11.4 使用上の注意

### 11.4.1 TPC 出力端子の動作

TP<sub>0</sub> ~ TP<sub>15</sub> は 16 ビットタイマ、DMAC、アドレスバスなどの端子と兼用になっています。これらの端子は、16 ビットタイマ、DMAC、アドレスバスが出力許可状態になっているときには、TPC 出力を行うことができません。ただし、NDR から DR への転送は、端子の状態にかかわらず常に行うことが可能です。

端子機能の変更は、出力トリガが発生しない状態で行ってください。

### 11.4.2 ノンオーバーラップ動作時の注意

ノンオーバーラップ動作時の NDR から DR の転送は以下のようになっています。

- (1) コンペアマッチ A では NDR の内容を常に DR へ転送します。
- (2) コンペアマッチ B では NDR の転送するビットの内容が 0 のときのみ転送を行います。1 のときは転送を行いません。

ノンオーバーラップ時の TPC 出力動作を図 11.9 に示します。

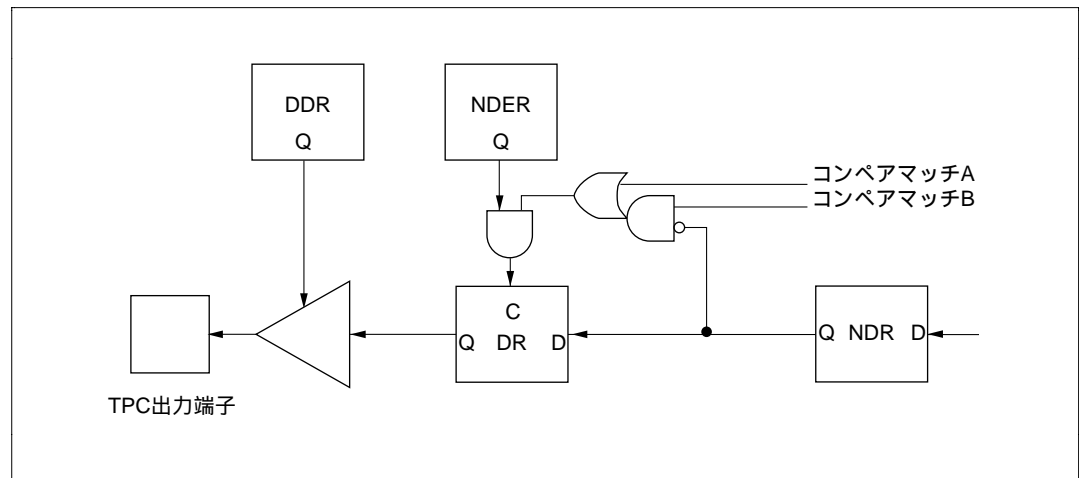


図 11.9 TPC 出力ノンオーバーラップ動作

したがって、コンペアマッチ B を、コンペアマッチ A よりも先に発生させることにより、0 データの転送を 1 データの転送に先だって行うことが可能です。

この場合、コンペアマッチ B が発生した後、コンペアマッチ A が発生するまで (ノンオーバーラップ期間) の間、NDR の内容を変更しないようにしてください。

これは IMFA 割込みの割込み処理ルーチンで NDR に次のデータをライトすることによって実現できます。また、IMFA 割込みで DMAC を起動することもできます。ただし、このライトは次のコンペアマッチ B が発生する前に行ってください。

このタイミングを図 11.10 に示します。

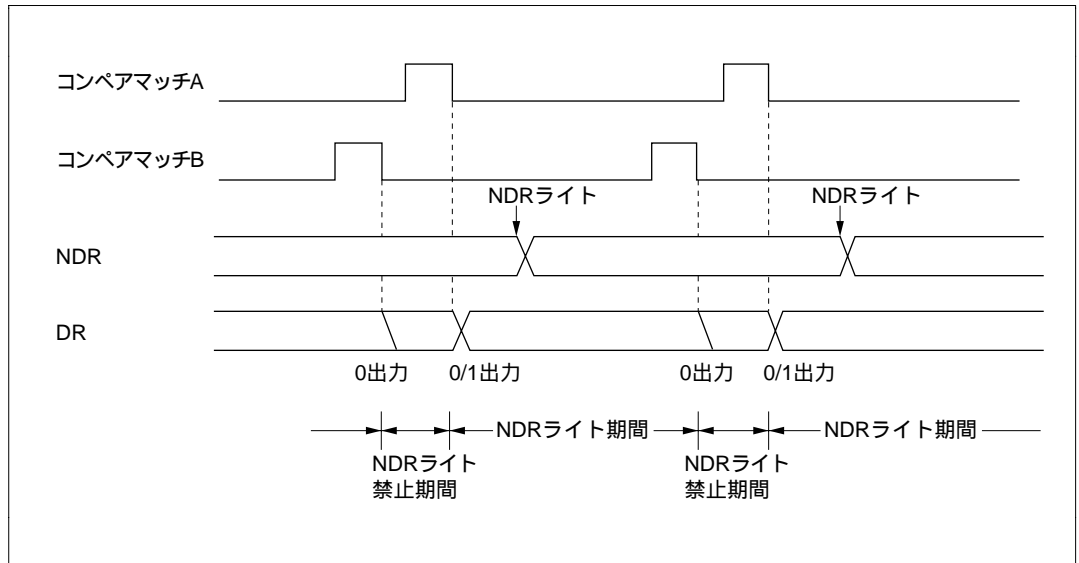


図 11.10 ノンオーバーラップ動作と NDR ライトタイミング



---

# 12. ウォッチドッグタイマ

---

## 第12章 目次

12.1	概要	499
12.1.1	特長	499
12.1.2	ブロック図	500
12.1.3	レジスタ構成	500
12.2	各レジスタの説明	501
12.2.1	タイマカウンタ (TCNT)	501
12.2.2	タイマコントロール/ステータスレジスタ (TCSR)	501
12.2.3	リセットコントロール/ステータスレジスタ (RSTCSR)	504
12.2.4	レジスタ書換え時の注意	505
12.3	動作説明	507
12.3.1	ウォッチドッグタイマ時の動作	507
12.3.2	インターバルタイマ時の動作	508
12.3.3	オーバフローフラグ (OVF) セットタイミング	508
12.3.4	ウォッチドッグタイマリセット (WRST) のセットタイミング	509
12.4	割込み	510
12.5	使用上の注意	510



## 12.1 概要

本 LSI は、ウォッチドッグタイマ (WDT) を内蔵しています。WDT には、システムの監視を行うウォッチドッグタイマとインターバルタイマの 2 つの機能があり、いずれかを選択することができます。

ウォッチドッグタイマはシステムの暴走などにより、タイマカウンタ (TCNT) の値が書き換えられずオーバーフローすると、本 LSI に対してリセット信号を発生します。

また、インターバルタイマは、TCNT がオーバーフローするごとにインターバルタイマ割込みを発生することができます。

### 12.1.1 特長

WDT の特長を以下に示します。

8 種類のカウンタ入力クロックを選択可能

/ 2、 / 32、 / 64、 / 128、 / 256、 / 512、 / 2048、 / 4096

インターバルタイマとして使用可能

TCNT がオーバーフローするとリセット信号または割込みを発生

ウォッチドッグタイマ時にはリセット信号、インターバルタイマ時にはインターバルタイマ割込みを発生します。

ウォッチドッグタイマの発生したリセット信号により、本 LSI 全体を内部リセット可能

ウォッチドッグタイマ時に TCNT のオーバーフローによってリセット信号を発生すると、本 LSI 全体は内部リセットされます。

### 12.1.2 ブロック図

図 12.1 に WDT のブロック図を示します。

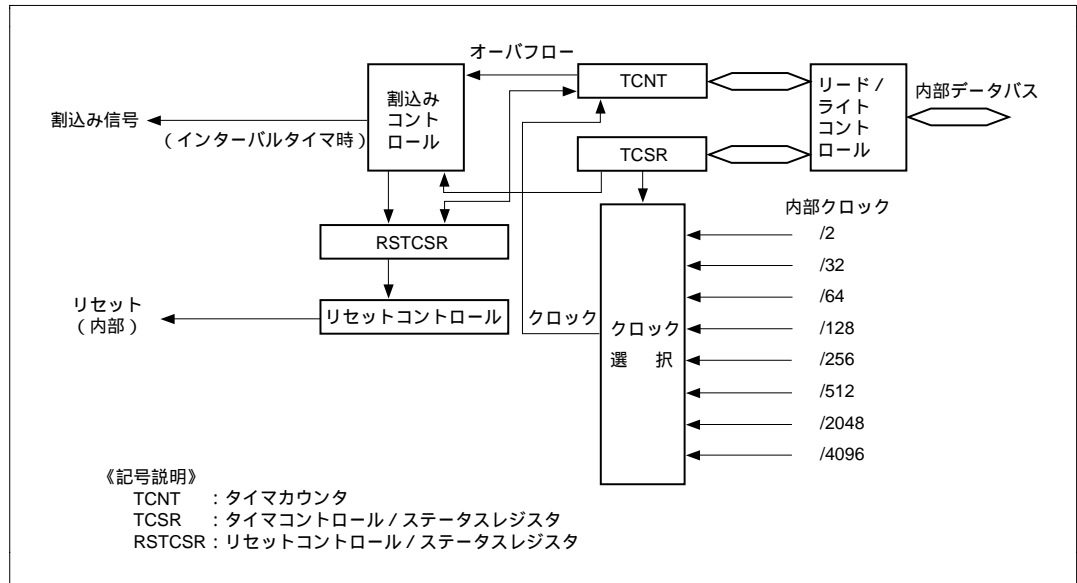


図 12.1 WDT のブロック図

### 12.1.3 レジスタ構成

表 12.1 に WDT のレジスタ構成を示します。

表 12.1 レジスタ構成

アドレス*1		名 称	略 称	R/W	初期値
ライト時*2	リード時				
H'FFF8C	H'FFF8C	タイマコントロール / ステータスレジスタ	TCSR	R/(W)*3	H'18
	H'FFF8D	タイマカウンタ	TCNT	R/W	H'00
H'FFF8E	H'FFF8F	リセットコントロール / ステータスレジスタ	RSTCSR	R/(W)	H'3F

- 【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。  
 \*2 このアドレスから始まるワードデータとしてライトしてください。  
 \*3 ビット 7 は、フラグをクリアするための 0 ライトのみ可能です。

## 12.2 各レジスタの説明

### 12.2.1 タイマカウンタ (TCNT)

TCNTは、8ビットのリード/ライト可能なアップカウンタです。

ビット:	7	6	5	4	3	2	1	0
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 TCNTは、容易に書き換えられないようにライト方法が一般のレジスタと異なります。詳細は「12.2.4 レジスタ書換え時の注意」を参照してください。

TCSRのTMEビットを1にセットすると、TCSRのCKS2~CKS0ビットで選択された内部クロックにより、カウントアップを開始します。また、TCNTの値がオーバーフロー(H'FF H'00)すると、TCSRのOVFフラグが1にセットされます。

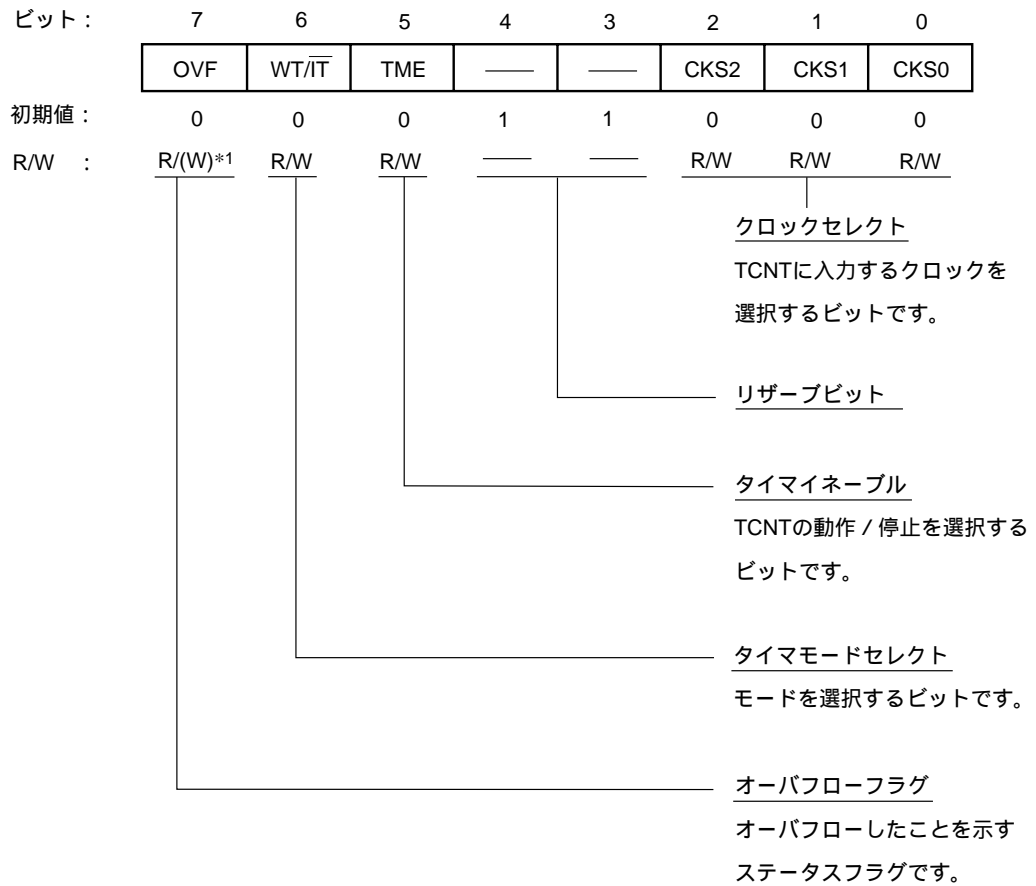
また、TCNTはリセット、またはTME=0のときH'00にイニシャライズされます。

### 12.2.2 タイマコントロール/ステータスレジスタ (TCSR)

TCSRは、8ビットのリード/ライト可能なレジスタで、TCNTに入力するクロックの選択、およびモードの選択などを行います。



## 12. ウォッチドッグタイマ



【注】 TCSRは、容易に書き換えられないようにライト方法が一般のレジスタと異なります。詳細は「12.2.4 レジスタ書き換え時の注意」を参照してください。  
 \*1 フラグをクリアするための0ライトのみ可能です。

ビット7~5 はリセット、またはスタンバイモード時に各ビットとも0 にイニシャライズされます。ビット2~0 は、リセット時に各ビットとも0 にイニシャライズされます。なお、ビット2~0 はソフトウェアスタンバイモード時には、イニシャライズされずにソフトウェアスタンバイモードに遷移する直前の値を保持します。

### ビット7：オーバフローフラグ (OVF)

TCNT がオーバフロー (H'FF H'00) したことを示すステータスフラグです。

ビット7	説明
OVF	
0	〔クリア条件〕 OVF = 1 の状態で、OVF フラグをリード後、OVF フラグに0 をライトしたとき (初期値)
1	〔セット条件〕 TCNT が H'FF H'00 に変化したとき

ビット6：タイマモードセレクト (WT/ $\overline{IT}$ )

WDT をウォッチドッグタイマとして使用するか、インターバルタイマとして使用するかを選択するビットです。インターバルタイマ時はTCNTのオーバフローでインターバルタイマ割込み要求を発生します。また、ウォッチドッグタイマ時はTCNTのオーバフローでリセット信号を発生します。

ビット6	説明
WT/ $\overline{IT}$	
0	インターバルタイマを選択：インターバルタイマ割込み要求 (初期値)
1	ウォッチドッグタイマを選択：リセット信号を発生

## ビット5：タイマイネーブル (TME)

TCNTの動作/停止を選択します。WT/ $\overline{IT}$ =1の場合、SYSCRのソフトウェアスタンバイビット (SSBY) を0クリアしてから、TMEをセットしてください。また、SSBYを1にセットするときは、TMEを0クリアしてください。

ビット5	説明
TME	
0	TCNTをH'00にイニシャライズし、カウント動作は停止 (初期値)
1	TCNTはカウント動作

## ビット4、3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

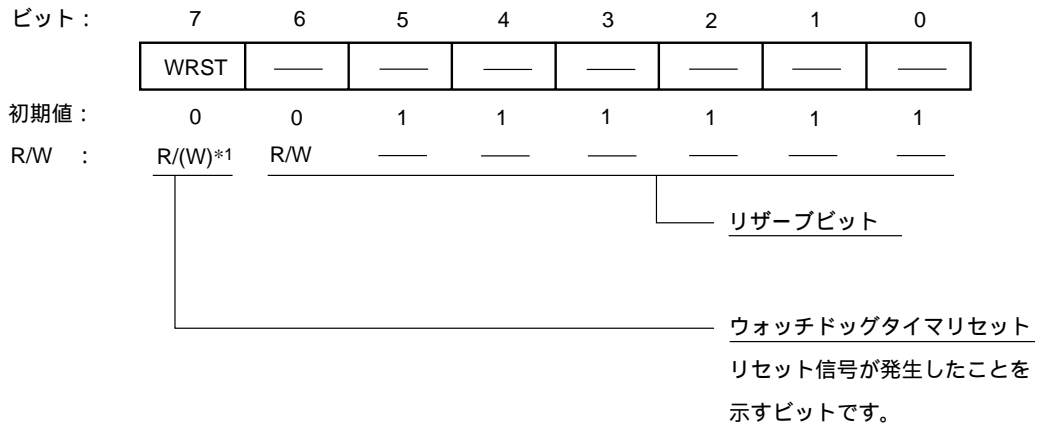
## ビット2~0：クロックセレクト2~0 (CKS2~0)

システムクロック ( ) を分周して得られる8種類の内部クロックからTCNTに入力するクロックを選択するビットです。

ビット2	ビット1	ビット0	説明
CKS2	CKS1	CKS0	
0	0	0	/2 (初期値)
0	0	1	/32
0	1	0	/64
0	1	1	/128
1	0	0	/256
1	0	1	/512
1	1	0	/2048
1	1	1	/4096

### 12.2.3 リセットコントロール/ステータスレジスタ (RSTCSR)

RSTCSR は 8 ビットのリード/ライト可能なレジスタで、ウォッチドッグタイマのオーバーフローによるリセット信号の発生状態をモニタします。



【注】 RSTCSRは、容易に書き換えられないように、ライト方法が一般のレジスタと異なります。詳細は「12.2.4 レジスタ書き換え時の注意」を参照してください。  
\*1 ビット7は、フラグをクリアするための0ライトのみ可能です。

ビット7、6は、RES 端子によるリセット信号でイニシャライズされます。ウォッチドッグタイマのオーバーフローによるリセット信号ではイニシャライズされません。

#### ビット7: ウォッチドッグタイマリセット (WRST)

ウォッチドッグタイマ時に TCNT がオーバーフローし、リセット信号が発生したことを示すビットです。

オーバーフローで発生したリセット信号により、本 LSI 全体が内部リセットされます。

ビット7	説 明
WRST	
0	〔クリア条件〕 (初期値) (1) RES 端子によるリセット信号 (2) WRST = 1 の状態で、WRST フラグをリード後、WRST フラグに 0 をライトしたとき
1	〔セット条件〕 ウォッチドッグタイマ時に、TCNT がオーバーフローし、リセット信号が発生したとき

#### ビット6: リザーブビット

リザーブビットです。

#### ビット5~0: リザーブビット

リザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。

## 12.2.4 レジスタ書換え時の注意

WDTのTCNT、TCSR、RSTCSRは、容易に書き換えられないように、ライト方法が一般レジスタと異なります。リード/ライトの方法を以下に示します。

### (1) TCNT、TCSR へのライト

TCNT、TCSR へライトする場合、必ずワード転送命令を使用してください。バイト命令では、ライトすることができません。図 12.2 に TCNT、TCSR へのライトデータを示します。

ライト時のアドレスは、TCNT、TCSR とも同一アドレスになっています。そのため、TCNT、TCSR へライトするときは、下位バイトをライトデータに、上位バイトを H'5A (TCNT のとき) または H'A5 (TCSR のとき) にしてワード転送を行います。

これにより、下位バイトのデータが TCNT、または TCSR へライトされます。

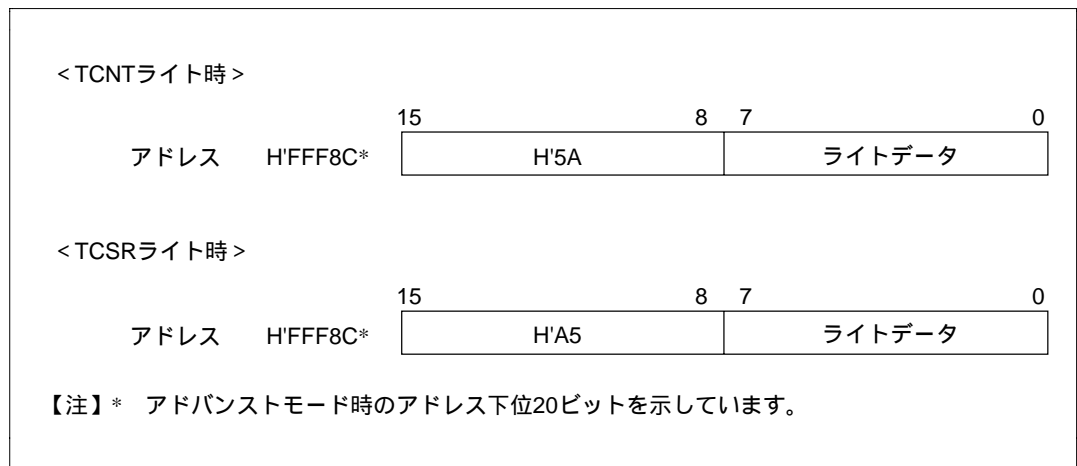


図 12.2 TCNT、TCSR へのライトデータ

## (2) RSTCSR へのライト

RSTCSR へライトする場合、必ずワード転送命令を使用してください。バイト転送命令では、ライトすることができません。図 12.3 に RSTCSR のライトデータを示します。

WRST ビットへ 0 をライトする場合、上位バイトを H'A5、下位バイトを H'00 としてワード転送を行います。これにより、下位バイトのデータ (H'00) が RSTCSR の WRST ビットへライトされ、WRST ビットが 0 にクリアされます。

RSTOE ビットへライトする場合、上位バイトを H'5A、下位バイトをライトデータとしてワード転送を行います。

これにより、下位バイトのデータが RSTOE ビットへライトされます。

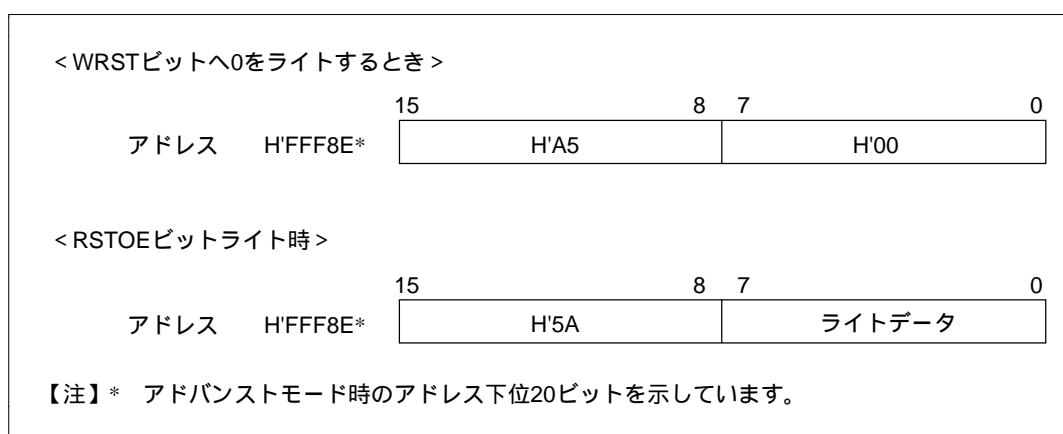


図 12.3 RSTCSR へのライトデータ

## (3) TCNT、TCSR、RSTCSR のリード

TCNT、TCSR、RSTCSR をリードする場合、アドレス H'FFF8C に TCSR、H'FFF8D に TCNT、H'FFF8F に RSTCSR が割り当てられています。

したがって、一般のレジスタと同様にリードしてください。リードの場合は、バイト転送命令を使用することができます。表 12.3 に TCNT、TCSR、RSTCSR のリードを示します。

表 12.3 TCNT、TCSR、RSTCSR のリード

アドレス*	レジスタ
H'FFF8C	TCSR
H'FFF8D	TCNT
H'FFF8F	RSTCSR

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

## 12.3 動作説明

以下にウォッチドッグタイマ時、インターバルタイマ時の WDT の動作について説明します。

### 12.3.1 ウォッチドッグタイマ時の動作

図 12.4 にウォッチドッグタイマ時の動作を示します。

ウォッチドッグタイマとして使用する場合は、TCSR の  $\overline{WTIT}$  ビット、TME ビットをそれぞれ 1 にセットします。

プログラムでは TCNT がオーバーフローする前に、ソフトウェアで TCNT の値を書き換えて (通常は H'00 をライト)、常にオーバーフローが発生しないようにします。システムの暴走などにより、TCNT の値が書き換えられず、オーバーフローすると、518 ステート期間、本 LSI 内部をリセットします。

WDT によるリセットと  $\overline{RES}$  端子によるリセットは、同一ベクタです。そのため、 $\overline{RES}$  端子によるリセットか、WDT によるリセットかは、RSTCSR の WRST ビットをチェックすることによって判別してください。

また、 $\overline{RES}$  端子によるリセットと WDT のオーバーフローによるリセットが同時に発生した場合は、 $\overline{RES}$  端子によるリセットが優先されます。

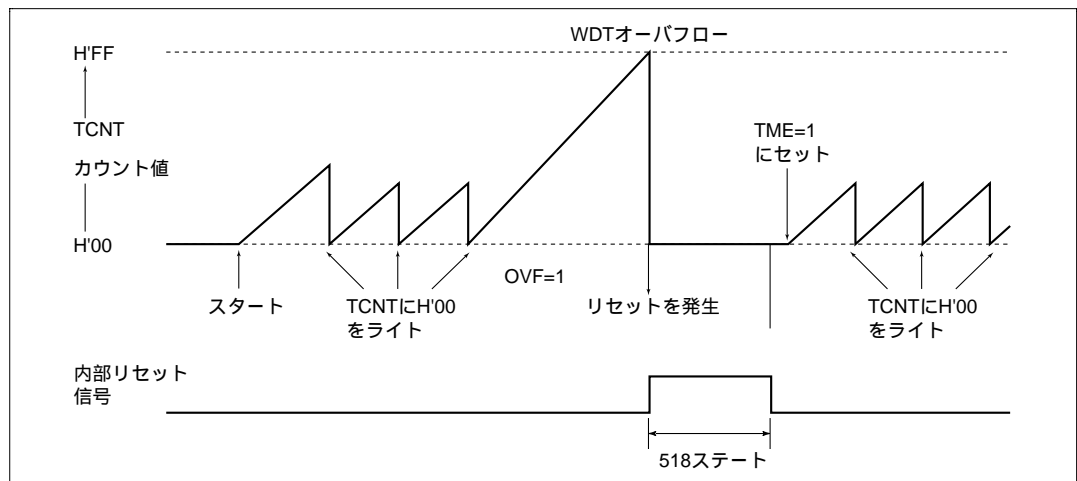


図 12.4 ウォッチドッグタイマモード時の動作

### 12.3.2 インターバルタイマ時の動作

図 12.5 にインターバルタイマ時の動作を示します。

インターバルタイマとして使用するには、TCSR の WT/IT̄ ビットを 0 にクリアし、TME ビットを 1 にセットします。

インターバルタイマとして動作しているときは、TCNT がオーバーフローするごとに、インターバルタイマ割込み要求が発生します。これにより、一定時間ごとにインターバルタイマ割込みを発生させることができます。

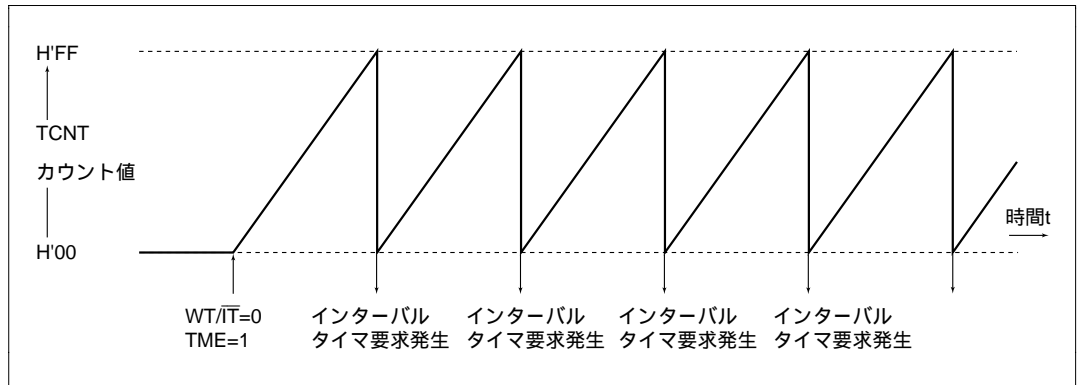


図 12.5 インターバルタイマ時の動作

### 12.3.3 オーバフローフラグ (OVF) セットタイミング

図 12.6 に OVF フラグのセットタイミングを示します。

TCSR の OVF フラグは、TCNT がオーバーフローすると 1 にセットされます。このとき同時にウォッチドッグタイマ時にはリセット信号出力、インターバルタイマ時にはインターバルタイマ割込みが発生します。

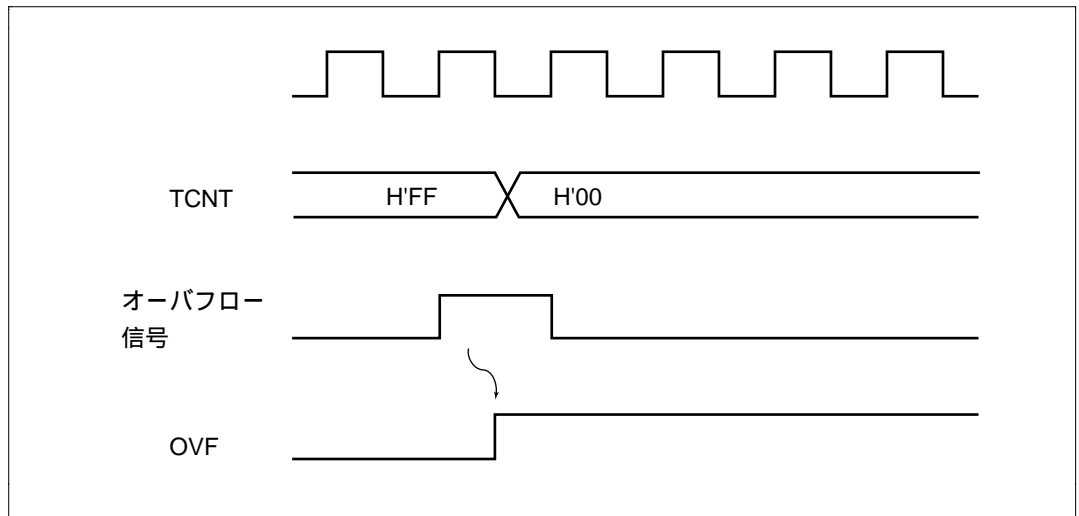


図 12.6 OVF フラグのセットタイミング

### 12.3.4 ウォッチドッグタイマリセット (WRST) のセットタイミング

RSTCSR の WRST ビットは、TCSR の WT/ $\overline{IT}$  ビット、TME ビットをそれぞれ 1 にセットしたとき有効になります。

図 12.7 に WRST ビットのセット、および内部リセットタイミングを示します。

TCNT がオーバーフローして、OVF フラグが "1" にセットされたとき、WRST ビットは 1 にセットされます。このとき同時に、本 LSI 全体に対して内部リセット信号を発生します。この内部リセット信号で OVF フラグは 0 にクリアされますが、WRST ビットは 1 にセットされたままです。したがって、リセット処理ルーチンの中で、必ず WRST ビットのクリアを行ってください。

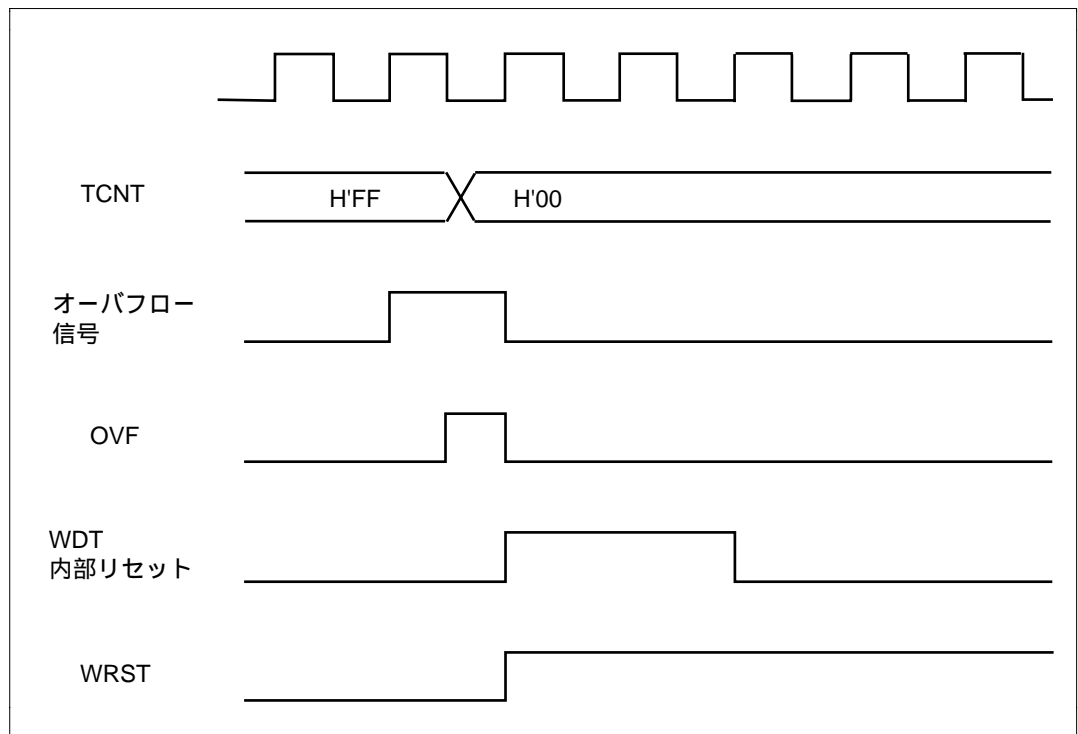


図 12.7 WRST ビットのセットおよび内部リセットタイミング



## 12.4 割込み

インターバルタイマ時、オーバフローによりインターバルタイマ割込み (WOVI) を発生します。インターバルタイマ割込みは TCSR の OVF フラグが 1 にセットされると常に要求されます。

## 12.5 使用上の注意

### (1) TCNT のライトとカウントアップの競合

図 12.8 に TCNT のライトとカウントアップの競合を示します。

TCNT のライトサイクル中の  $T_3$  ステートでカウントアップが発生しても、カウントアップされずカウンタライトが優先されます。

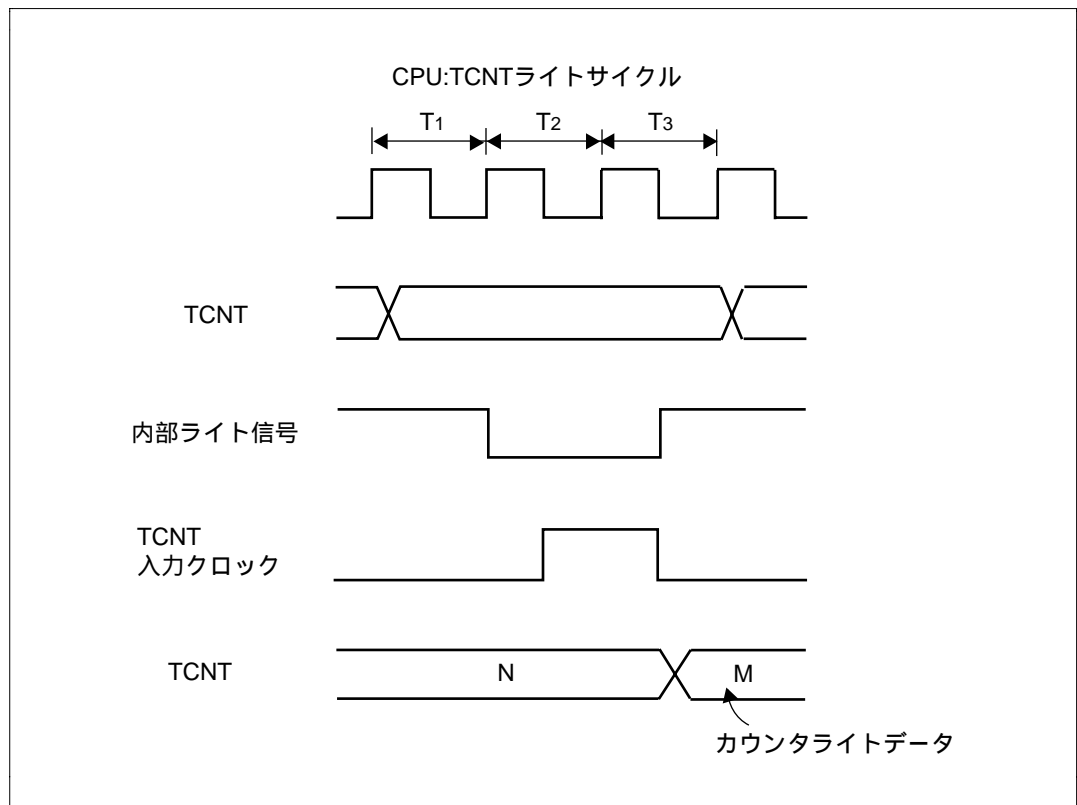


図 12.8 TCNT のライトとカウントアップの競合

### (2) CKS2 ~ CKS0 ビットの切り換え

CKS2 ~ CKS0 ビットを切り換えるときは、TCSR の TME ビットを 0 にクリアし、TCNT を停止させてから行ってください。

---

# 13. SCI

---

## 第13章 目次

13.1	概要	513
13.1.1	特長	513
13.1.2	ブロック図	515
13.1.3	端子構成	516
13.1.4	レジスタ構成	517
13.2	各レジスタの説明	518
13.2.1	レシーブシフトレジスタ (RSR)	518
13.2.2	レシーブデータレジスタ (RDR)	518
13.2.3	トランスミットシフトレジスタ (TSR)	519
13.2.4	トランスミットデータレジスタ (TDR)	519
13.2.5	シリアルモードレジスタ (SMR)	520
13.2.6	シリアルコントロールレジスタ (SCR)	524
13.2.7	シリアルステータスレジスタ (SSR)	529
13.2.8	ビットレートレジスタ (BRR)	536
13.3	動作説明	544
13.3.1	概要	544
13.3.2	調歩同期式モード時の動作	547
13.3.3	マルチプロセッサ通信機能	557
13.3.4	クロック同期式モード時の動作	563
13.4	SCI 割込み	572
13.5	使用上の注意	573
13.5.1	SCI を使用する際の注意	573



## 13.1 概要

本 LSI は、独立した 3 チャンネルのシリアルコミュニケーションインタフェース (SCI : Serial Communication Interface) を備えています。3 チャンネルは、同一の機能を持っています。SCI は、調歩同期式通信とクロック同期式通信の 2 方式でシリアル通信ができます。また、複数のプロセッサ間のシリアル通信機能 (マルチプロセッサ通信機能) を備えています。

消費電流低減のため SCI を使用しない場合には、SCI 各チャンネル単独に停止することができます。詳細は「20.6 モジュールスタンバイ機能」を参照してください。

また、SCI は "ISO/IEC7816-3 (Identification Card)" に準拠した IC カードインタフェース用シリアル通信機能としてスマートカードインタフェースをサポートしています。通常のシリアルコミュニケーションインタフェースとスマートカードインタフェースの切り替えはレジスタの設定で行います。

### 13.1.1 特長

通常のシリアルコミュニケーションインタフェースの特長を以下に示します。

シリアル通信モードを調歩同期式モード / クロック同期式モードから選択可能

#### (a) 調歩同期式モード

キャラクタ単位で同期をとる調歩同期式方式でシリアルデータの通信を行います。Universal Asynchronous Receiver/Transmitter (UART) や Asynchronous Communication Interface Adapter (ACIA) など標準の調歩同期式通信用 LSI とのシリアルデータ通信が可能です。また、複数のプロセッサとシリアルデータ通信ができるマルチプロセッサ間通信機能を備えています。

シリアルデータ通信フォーマットを 12 種類のフォーマットから選択できます。

- ・データ長 : 7 ビット / 8 ビット
- ・ストップビット長 : 1 ビット / 2 ビット
- ・パリティ : 偶数パリティ / 奇数パリティ / パリティなし
- ・マルチプロセッサビット : 1 / 0
- ・受信エラーの検出 : パリティエラー、オーバランエラー、フレーミングエラーを検出
- ・ブレークの検出 : フレーミングエラー発生時に RxD 端子のレベルを直接リードすることによりブレークを検出できます。

#### (b) クロック同期式モード

クロックに同期してシリアルデータ通信を行います。クロック同期式通信機能を持つ他の LSI とのシリアルデータ通信が可能です。

シリアルデータ通信フォーマットは1種類です。

- ・データ長                               : 8ビット
- ・受信エラーの検出                   : オーバランエラーを検出

全二重通信が可能

独立した送信部と受信部を備えているので、送信と受信を同時に行うことができます。また、送信部、および受信部ともにダブルバッファ構造になっていますのでシリアルデータの連続送信、連続受信ができます。

送受信するシリアルデータに対して、

- ・LSB ファースト / MSB ファースト
- ・データのロジックレベルの反転

を設定することができます。

内蔵ボーレートジェネレータにより任意のビットレートを選択可能

送受信クロックソースを、ボーレートジェネレータからの内部クロック、または SCK 端子からの外部クロックから選択可能

4種類の割込み要因

送信データエンプティ、送信終了、受信データフル、受信エラーの4種類の割込み要因があり、それぞれ独立に要求することができます。また、SCI0 については送信データエンプティ割込みと受信データフル割込みにより DMA コントローラ (DMAC) を起動させてデータの転送を行うことができます。

スマートカードインタフェースの特長を以下に示します。

調歩同期式モード

- ・データ長                               : 8ビット
- ・パリティビットの生成およびチェック
- ・受信モードにおけるエラーシグナル (パリティエラー) の送出
- ・送信モードにおけるエラーシグナルの検出とデータの自動再送信
- ・ダイレクトコンベンション / インバースコンベンションの両方をサポート

内蔵ボーレートジェネレータにより任意のビットレートを選択可能

## 3種類の割り込み要因

送信データエンプティ、受信データフル、送受信エラーの3種類の割り込み要因があり、それぞれ独立に要求することができます。また、送信データエンプティ割り込みと受信データフル割り込みによりDMAコントローラ(DMAC)を起動させてデータの転送を行うことができます。

## 13.1.2 ブロック図

図 13.1 に SCI のブロック図を示します。

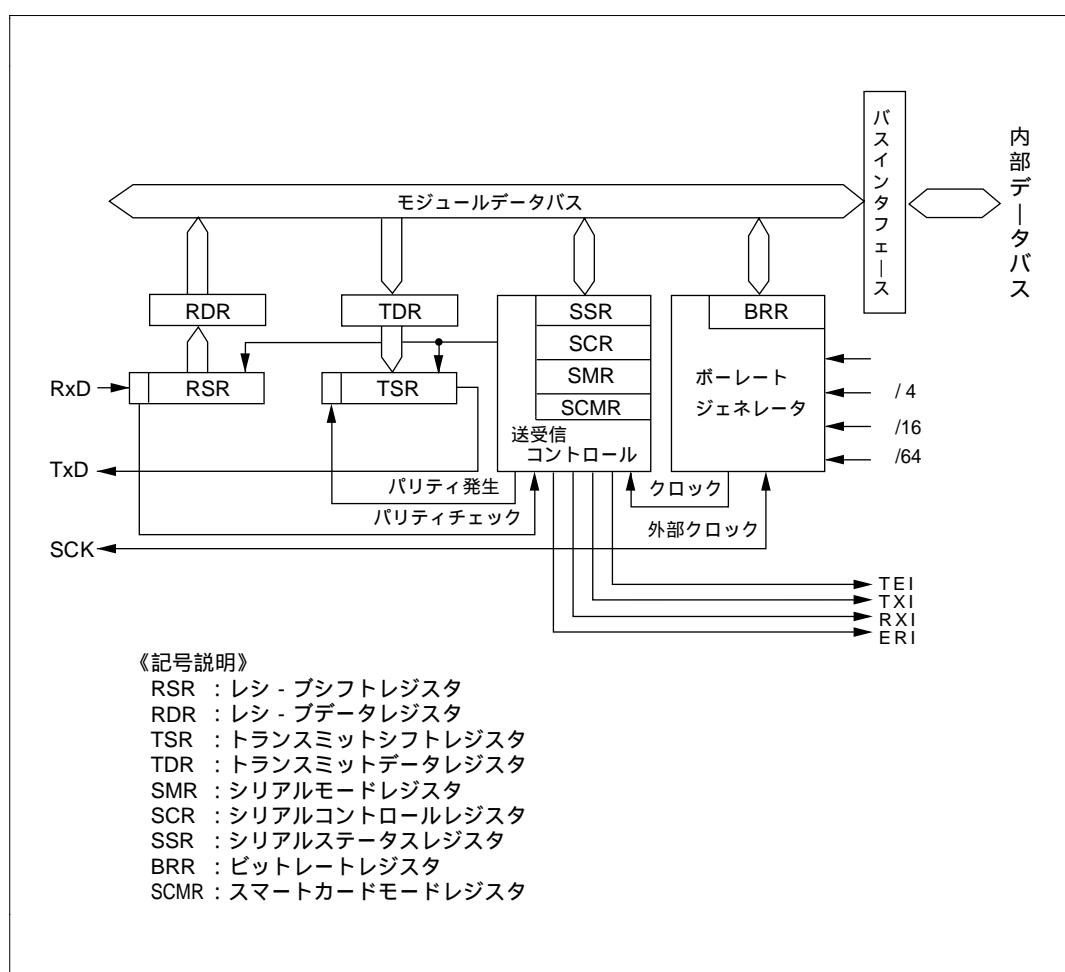


図 13.1 SCI のブロック図

### 13.1.3 端子構成

SCIは、チャンネルごとに表 13.1 に示すシリアル端子を持っています。

表 13.1 端子構成

チャンネル	名称	略称	入出力	機能
0	シリアルクロック端子	SCK <sub>0</sub>	入出力	SCI <sub>0</sub> のクロック入出力
	レシーブデータ端子	RxD <sub>0</sub>	入力	SCI <sub>0</sub> の受信データ入力
	トランスミットデータ端子	TxD <sub>0</sub>	出力	SCI <sub>0</sub> の送信データ出力
1	シリアルクロック端子	SCK <sub>1</sub>	入出力	SCI <sub>1</sub> のクロック入出力
	レシーブデータ端子	RxD <sub>1</sub>	入力	SCI <sub>1</sub> の受信データ入力
	トランスミットデータ端子	TxD <sub>1</sub>	出力	SCI <sub>1</sub> の送信データ出力
2	シリアルクロック端子	SCK <sub>2</sub>	入出力	SCI <sub>2</sub> のクロック入出力
	レシーブデータ端子	RxD <sub>2</sub>	入力	SCI <sub>2</sub> の受信データ入力
	トランスミットデータ端子	TxD <sub>2</sub>	出力	SCI <sub>2</sub> の送信データ出力

### 13.1.4 レジスタ構成

SCI には、表 13.2 に示す内部レジスタがあります。これらのレジスタにより調歩同期式モード/クロック同期式モードの指定、データフォーマットの指定、ビットレートの指定、送信部/受信部の制御、およびシリアルコミュニケーションインタフェースとスマートカードインタフェースの切り替えの指定を行うことができます。

表 13.2 レジスタ構成

チャンネル	アドレス* <sup>1</sup>	名称	略称	R/W	初期値
0	H'FFFB0	シリアルモードレジスタ	SMR	R/W	H'00
	H'FFFB1	ビットレートレジスタ	BRR	R/W	H'FF
	H'FFFB2	シリアルコントロールレジスタ	SCR	R/W	H'00
	H'FFFB3	トランスミットデータレジスタ	TDR	R/W	H'FF
	H'FFFB4	シリアルステータスレジスタ	SSR	R/(W)* <sup>2</sup>	H'84
	H'FFFB5	レシーブデータレジスタ	RDR	R	H'00
	H'FFFB6	スマートカードモードレジスタ	SCMR	R/W	H'F2
1	H'FFFB8	シリアルモードレジスタ	SMR	R/W	H'00
	H'FFFB9	ビットレートレジスタ	BRR	R/W	H'FF
	H'FFFB A	シリアルコントロールレジスタ	SCR	R/W	H'00
	H'FFFB B	トランスミットデータレジスタ	TDR	R/W	H'FF
	H'FFFB C	シリアルステータスレジスタ	SSR	R/(W)* <sup>2</sup>	H'84
	H'FFFB D	レシーブデータレジスタ	RDR	R	H'00
	H'FFFB E	スマートカードモードレジスタ	SCMR	R/W	H'F2
2	H'FFFC0	シリアルモードレジスタ	SMR	R/W	H'00
	H'FFFC1	ビットレートレジスタ	BRR	R/W	H'FF
	H'FFFC2	シリアルコントロールレジスタ	SCR	R/W	H'00
	H'FFFC3	トランスミットデータレジスタ	TDR	R/W	H'FF
	H'FFFC4	シリアルステータスレジスタ	SSR	R/(W)* <sup>2</sup>	H'84
	H'FFFC5	レシーブデータレジスタ	RDR	R	H'00
	H'FFFC6	スマートカードモードレジスタ	SCMR	R/W	H'F2

【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。

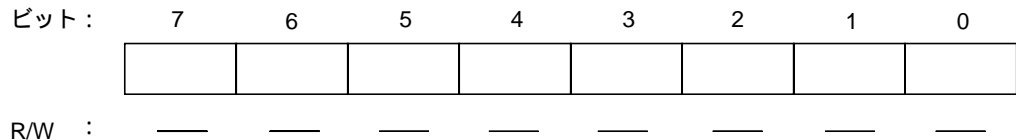
\*2 フラグをクリアするための 0 ライトのみ可能です。



## 13.2 各レジスタの説明

### 13.2.1 レシーブシフトレジスタ (RSR)

RSR は、シリアルデータを受信するためのレジスタです。

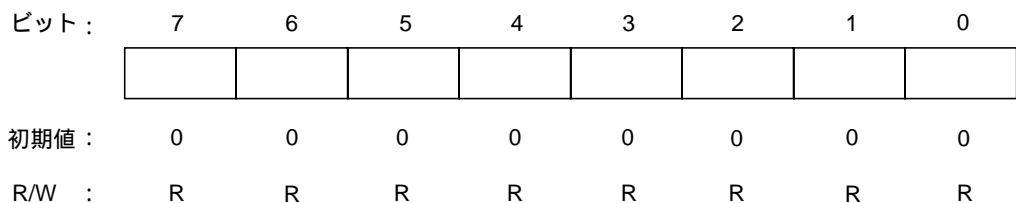


SCI は、RSR に RxD 端子から入力されたシリアルデータを LSB (ビット 0) から受信した順にセットし、パラレルデータに変換します。1 バイトのデータ受信を終了すると、データは自動的に RDR へ転送されます。

CPU から RSR を直接リード/ライトすることはできません。

### 13.2.2 レシーブデータレジスタ (RDR)

RDR は、受信したシリアルデータを格納するレジスタです。



SCI は、1 バイトのシリアルデータの受信が終了すると、RSR から RDR へ受信したシリアルデータを転送して格納し、受信動作を完了します。この後、RSR は受信可能になります。

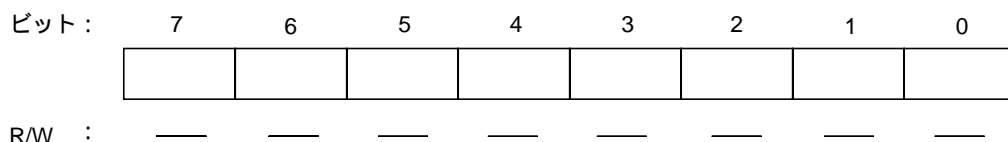
このように、RSR と RDR はダブルバッファになっているため連続した受信動作が可能です。

RDR は、リード専用レジスタですので CPU からライトすることはできません。

RDR は、リセット、またはスタンバイモード時に H'00 にイニシャライズされます。

### 13.2.3 トランスミットシフトレジスタ (TSR)

TSR は、シリアルデータを送信するためのレジスタです。



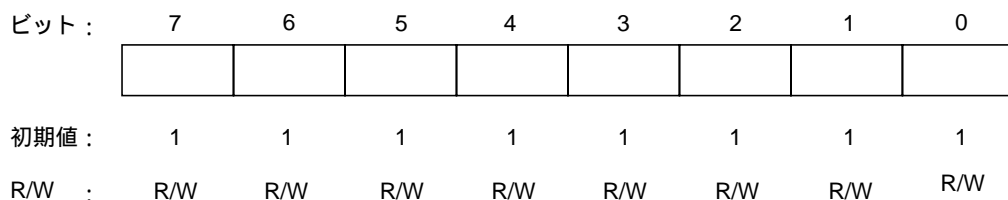
SCI は、TDR から送信データをいったん TSR に転送し、LSB (ビット 0) から順に TxD 端子に送り出すことでシリアルデータ送信を行います。

1 バイトのデータ送信を終了すると自動的に TDR から TSR へ次の送信データを転送し、送信を開始します。ただし SSR の TDRE ビットが 1 にセットされている場合には、TDR から TSR へのデータ転送は行いません。

CPU から、直接 TSR をリード/ライトすることはできません。

### 13.2.4 トランスミットデータレジスタ (TDR)

TDR は、シリアル送信するデータを格納する 8 ビットのレジスタです。



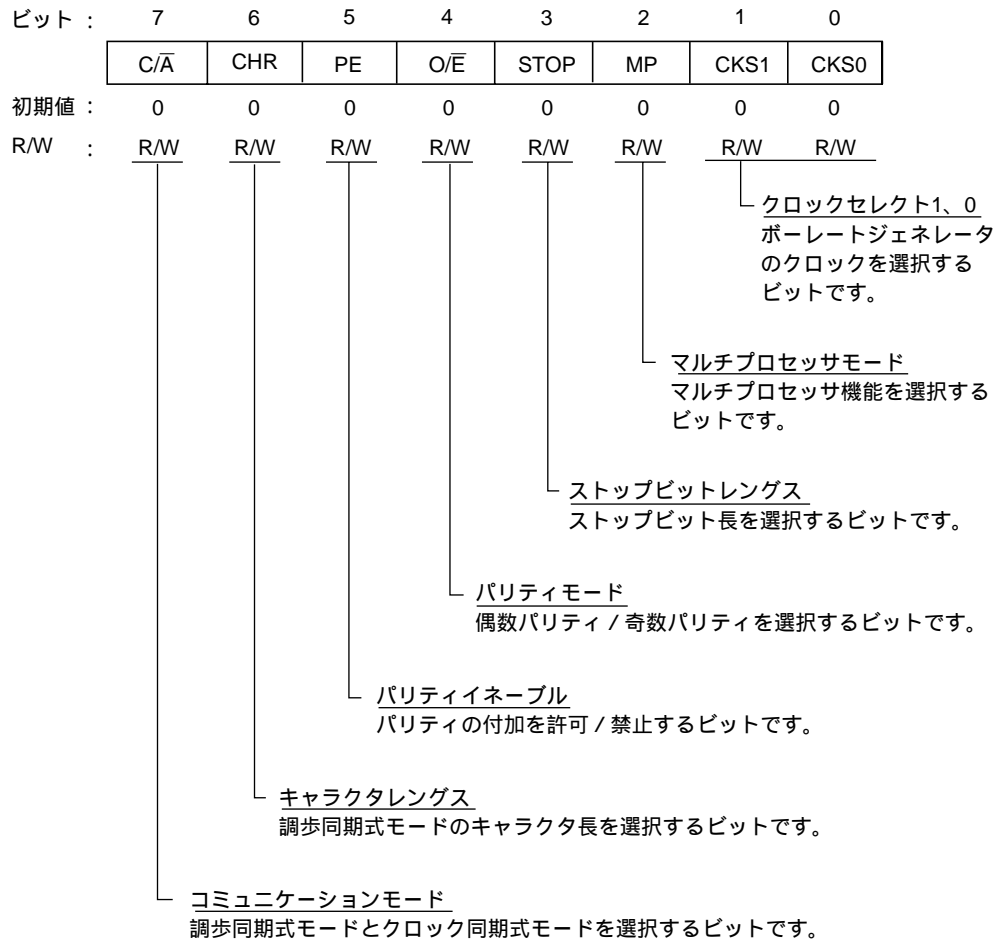
SCI は、TSR の空を検出すると、TDR にライトされた送信データを TSR に転送してシリアル送信を開始します。TSR のシリアルデータ送信中に TDR に次の送信データをライトしておくと、連続シリアル送信ができます。

TDR は、常に CPU によるリード/ライトが可能です。

TDR は、リセット、またはスタンバイモード時に H'FF にイニシャルライズされます。

### 13.2.5 シリアルモードレジスタ (SMR)

SMR は、SCI のシリアル通信フォーマットの設定と、ボーレートジェネレータのクロックソースを選択するための 8 ビットのレジスタです。



SMR は、常に CPU によるリード / ライトが可能です。

SMR は、リセット、またはスタンバイモード時に H'00 にイニシャライズされます。

ビット7 : コミュニケーションモード (C/ $\bar{A}$ ) / GSM モード (GM)

通常のシリアルコミュニケーションインタフェースとスマートカードインタフェースでは、機能が異なります。SCMR の SMIF ビットによって切り替わります。

(a) シリアルコミュニケーションインタフェースの時 (SCMR の SMIF ビットが 0)

SCI の動作モードを調歩同期式モード / クロック同期式モードのいずれかから選択します。

ビット 7	説明	
C/A		
0	調歩同期式モード	(初期値)
1	クロック同期式モード	

(b) スマートカードインタフェースの時 (SCMR の SMIF ビットが 1)

スマートカードインタフェースの時 GSM モードを選択します。

ビット 7	説明	
GM		
0	TEND フラグがスタートビットから 12.5 etu 後に発生します。	(初期値)
1	TEND フラグがスタートビットから 11.0 etu 後に発生します。	

【注】 etu (Elementary Time Unit) : 1 ビットの転送期間

ビット 6 : キャラクタレングス (CHR)

調歩同期式モードのデータ長を 7 ビット / 8 ビットデータのいずれかから選択します。クロック同期式モードでは CHR の設定にかかわらず、データ長は 8 ビットデータ固定です。

ビット 6	説明	
CHR		
0	8 ビットデータ	(初期値)
1	7 ビットデータ*	

【注】 \* 7 ビットデータを選択した場合、TDR の MSB (ビット 7) は送信されません。

ビット 5 : パリティイネーブル (PE)

調歩同期式モードでは、送信時にパリティビットの付加を、受信時にパリティビットのチェックを行うかどうかを選択します。クロック同期式モードでは、PE ビットの設定にかかわらずパリティビットの付加、およびチェックは行いません。

ビット 5	説明	
PE		
0	パリティビットの付加、およびチェックを禁止	(初期値)
1	パリティビットの付加、およびチェックを許可*	

【注】\* PE ビットに1をセットすると送信時には、O $\bar{E}$  ビットで指定した偶数、または奇数パリティを送信データに付加して送信します。受信時には、受信したパリティビットがO $\bar{E}$  ビットで指定した偶数、または奇数パリティになっているかどうかをチェックします。

#### ビット4：パリティモード（O $\bar{E}$ ）

パリティの付加やチェックを偶数パリティ / 奇数パリティのいずれで行うかを選択します。O $\bar{E}$  ビットの設定は、調歩同期式モードで PE ビットに1を設定しパリティビットの付加やチェックを許可したときのみ有効になります。クロック同期式モードや、調歩同期式モードでパリティの付加やチェックを禁止している場合には、O $\bar{E}$  ビットの設定は無効です。

ビット4	説明
O $\bar{E}$	
0	偶数パリティ* <sup>1</sup> (初期値)
1	奇数パリティ* <sup>2</sup>

【注】 \*1 偶数パリティに設定すると送信時には、パリティビットと送信キャラクタをあわせて、その中の1の数の合計が偶数になるようにパリティビットを付加して送信します。受信時には、パリティビットと受信キャラクタをあわせて、その中の1の数の合計が偶数であるかどうかをチェックします。

\*2 奇数パリティに設定すると送信時には、パリティビットと送信キャラクタをあわせて、その中の1の数の合計が奇数になるようにパリティビットを付加して送信します。受信時には、パリティビットと受信キャラクタをあわせて、その中の1の数の合計が奇数であるかどうかをチェックします。

#### ビット3：ストップビットレングス（STOP）

調歩同期式モードでのストップビットの長さを1ビット / 2ビットのいずれかから選択します。STOP ビットの設定は調歩同期式モードでのみ有効になります。クロック同期式モードに設定した場合にはストップビットは付加されませんので、このビットの設定は無効です。

ビット3	説明
STOP	
0	1ストップビット* <sup>1</sup> (初期値)
1	2ストップビット* <sup>2</sup>

【注】 \*1 送信時には、送信キャラクタの最終尾に1ビットの1（ストップビット）を付加して送信します。

\*2 送信時には、送信キャラクタの最終尾に2ビットの1（ストップビット）を付加して送信します。

なお、受信時にはSTOPビットの設定にかかわらず、受信したストップビットの1ビット目のみをチェックします。ストップビットの2ビット目が1の場合は、ストップビットとして扱いますが、0の場合は、次の送信キャラクタのスタートビットとして扱います。

#### ビット2：マルチプロセッサモード（MP）

マルチプロセッサフォーマットを選択します。マルチプロセッサフォーマットを選択した場合、PEビット、および $O\bar{E}$ ビットにおけるパリティの設定は無効になります。また、MPビットの設定は、調歩同期式モードのときのみ有効です。クロック同期式モードでは、MPビットの設定は無効です。

マルチプロセッサ通信機能については、「13.3.3 マルチプロセッサ通信機能」を参照してください。

ビット2	説明
MP	
0	マルチプロセッサ機能の禁止 (初期値)
1	マルチプロセッサフォーマットを選択

#### ビット1、0：クロックセレクト1、0（CKS1、0）

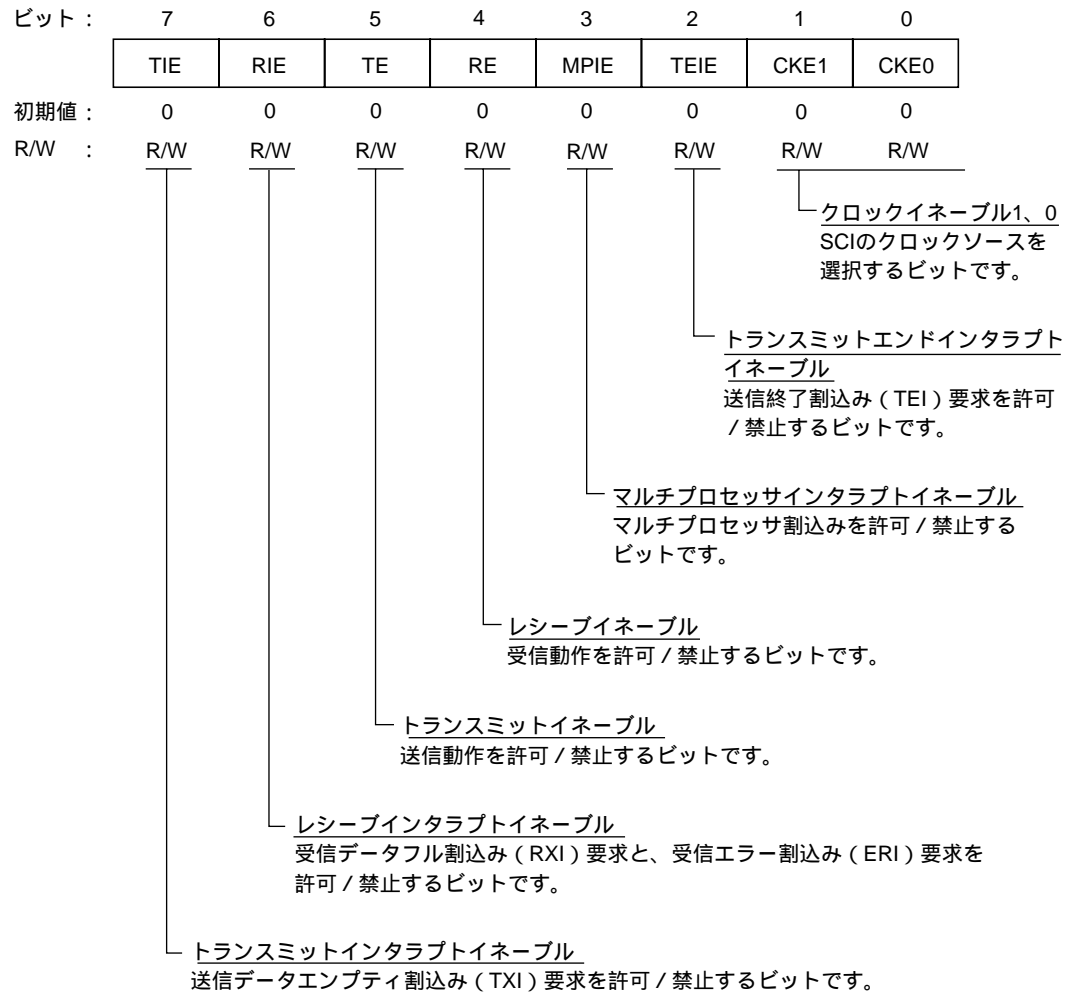
内蔵ボーレートジェネレータのクロックソースを選択します。CKS1、0ビットの設定により、 $/4$ 、 $/16$ 、 $/64$ の4種類からクロックソースを選択できます。

クロックソースと、ビットレートレジスタの設定値、およびボーレートの関係については、「13.2.8 ビットレートレジスタ」を参照してください。

ビット1	ビット0	説明
CKS1	CKS0	
0	0	クロック (初期値)
0	1	$/4$ クロック
1	0	$/16$ クロック
1	1	$/64$ クロック

### 13.2.6 シリアルコントロールレジスタ (SCR)

SCR は、SCI の送信 / 受信動作、調歩同期式モードでのシリアルクロック出力、割り込み要求の許可 / 禁止、および送信 / 受信クロックソースの選択を行うレジスタです。



SCR は、常に CPU によるリード / ライトが可能です。

SCR は、リセット、またはスタンバイモード時に H'00 にイニシャライズされます。

### ビット7：トランスミットインタラプトイネーブル（TIE）

TDR から TSR ヘシリアル送信データが転送され SSR の TDRE フラグが 1 にセットされたときに、送信データエンプティ割込み（TXI）要求の発生を許可 / 禁止します。

ビット7	説明
TIE	
0	送信データエンプティ割込み（TXI）要求の禁止* (初期値)
1	送信データエンプティ割込み（TXI）要求の許可

【注】\* TXI 割込み要求の解除は、TDRE フラグから 1 をリードした後、0 にクリアするか、または TIE ビットを 0 にクリアすることで行うことができます。

### ビット6：レシーブインタラプトイネーブル（RIE）

シリアル受信データが RSR から RDR へ転送されて SSR の RDRF フラグが 1 にセットされたとき、受信データフル割込み（RXI）要求、および受信エラー割込み（ERI）要求の発生を許可 / 禁止します。

ビット6	説明
RIE	
0	受信データフル割込み（RXI）要求、および受信エラー割込み（ERI）要求を禁止* (初期値)
1	受信データフル割込み（RXI）要求、および受信エラー割込み（ERI）要求を許可

【注】\* RXI、および ERI 割込み要求の解除は、RDRF、または FER、PER、ORER の各フラグから 1 をリードした後、0 にクリアするか、RIE ビットを 0 にクリアすることで行えます。

### ビット5：トランスミットイネーブル（TE）

SCI のシリアル送信動作の開始を許可 / 禁止します。

ビット5	説明
TE	
0	送信動作を禁止* <sup>1</sup> (初期値)
1	送信動作を許可* <sup>2</sup>

【注】 \*<sup>1</sup> SSR の TDRE フラグは 1 に固定されます。

\*<sup>2</sup> この状態で、TDR に送信データをライトして、SSR の TDRE フラグを 0 にクリアするとシリアル送信を開始します。

なお、TE ビットを 1 にセットする前に必ず SMR の設定を行い送信フォーマットを決定してください。



## ビット4：レシーブイネーブル（RE）

SCIのシリアル受信動作の開始を許可／禁止します。

ビット4	説明
RE	
0	受信動作を禁止* <sup>1</sup> (初期値)
1	受信動作を許可* <sup>2</sup>

【注】 \*1 RE ビットを0にクリアしてもRDRF、FER、PER、ORERの各フラグは影響を受けず、状態を保持しますので注意してください。

\*2 この状態で調歩同期式モードの場合はスタートビットを、クロック同期式モードの場合は同期クロック入力をそれぞれ検出すると、シリアル受信を開始します。なお、RE ビットを1にセットする前に必ずSMRの設定を行い、受信フォーマットを決定してください。

## ビット3：マルチプロセッサインタラプトイネーブル（MPIE）

マルチプロセッサ割込みを許可／禁止します。MPIE ビットの設定は、調歩同期式モードで、かつ、SMRのMPビットが1に設定されている受信時のみ有効です。

クロック同期式モードのとき、あるいはMPビットが0のときにはMPIEビットの設定は無効です。

ビット3	説明
MPIE	
0	マルチプロセッサ割込み禁止状態（通常の受信動作をします）（初期値） 〔クリア条件〕 (1) MPIE ビットを0にクリア (2) MPB=1のデータを受信したとき
1	マルチプロセッサ割込み許可状態* マルチプロセッサビットが1のデータを受け取るまで受信割込み（RXI）要求、受信エラー割込み（ERI）要求、およびSSRのRDRF、FER、ORERの各フラグのセットを禁止します。

【注】\* RSRからRDRへの受信データの転送、および受信エラーの検出とSSRのRDRF、FER、ORERの各フラグのセットは行いません。MPB=1を含む受信データを受信すると、SSRのMPBビットを1にセットし、MPIEビットを自動的に0にクリアし、RXI、ERI割込み要求の発生（SCRのTIE、RIEビットが1にセットされている場合）とFER、ORERフラグのセットが許可されます。

### ビット2：トランスミットエンドインタラプトイネーブル (TEIE)

MSB データ送出時に有効な送信データが TDR がないとき、送信終了割込み (TEI) 要求の発生を許可 / 禁止します。

ビット2	説明
TEIE	
0	送信終了割込み (TEI) 要求を禁止* (初期値)
1	送信終了割込み (TEI) 要求を許可*

【注】\* TEIの解除は、SSRのTDREフラグから1をリードした後、0にクリアしてTENDフラグを0にクリアするか、TEIEビットを0にクリアすることで行うことができます。

### ビット1、0：クロックイネーブル1、0 (CKE1、0)

通常のシリアルコミュニケーションインタフェースとスマートカードインタフェースでは、機能が異なります。また、SCMRのSMIFビットによって切り替えます。

#### (a) シリアルコミュニケーションインタフェースのとき (SCMRのSMIFビットが0)

SCIのクロックソースの選択、およびSCK端子からのクロック出力の許可 / 禁止を設定します。CKE1ビットとCKE0ビットの組み合わせによってSCK端子を入出力ポートにするか、シリアルクロック出力端子にするか、またはシリアルクロック入力端子にするかが決まります。

ただし、CKE0ビットの設定は調歩同期式モードで内部クロック動作 (CKE1=0) 時のみ有効です。クロック同期式モードのとき、および外部クロック動作 (CKE1=1) の場合はCKE0ビットの設定は無効です。また、SMRでSCIの動作モードを設定する前にCKE1、CKE0ビットを設定してください。

SCIのクロックソースの選択についての詳細は「13.3 動作説明」の表13.9を参照してください。

ビット1	ビット0	説明	
CKE1	CKE0		
0	0	調歩同期式モード	内部クロック / SCK 端子は入出力ポート* <sup>1</sup>
		クロック同期式モード	内部クロック / SCK 端子は同期クロック出力* <sup>1</sup>
0	1	調歩同期式モード	内部クロック / SCK 端子はクロック出力* <sup>2</sup>
		クロック同期式モード	内部クロック / SCK 端子は同期クロック出力
1	0	調歩同期式モード	外部クロック / SCK 端子はクロック入力* <sup>3</sup>
		クロック同期式モード	外部クロック / SCK 端子は同期クロック入力
1	1	調歩同期式モード	外部クロック / SCK 端子はクロック入力* <sup>3</sup>
		クロック同期式モード	外部クロック / SCK 端子は同期クロック入力

【注】 \*1 初期値

\*2 ビットレートと同じ周波数のクロックを出力

\*3 ビットレートの 16 倍の周波数のクロックを入力

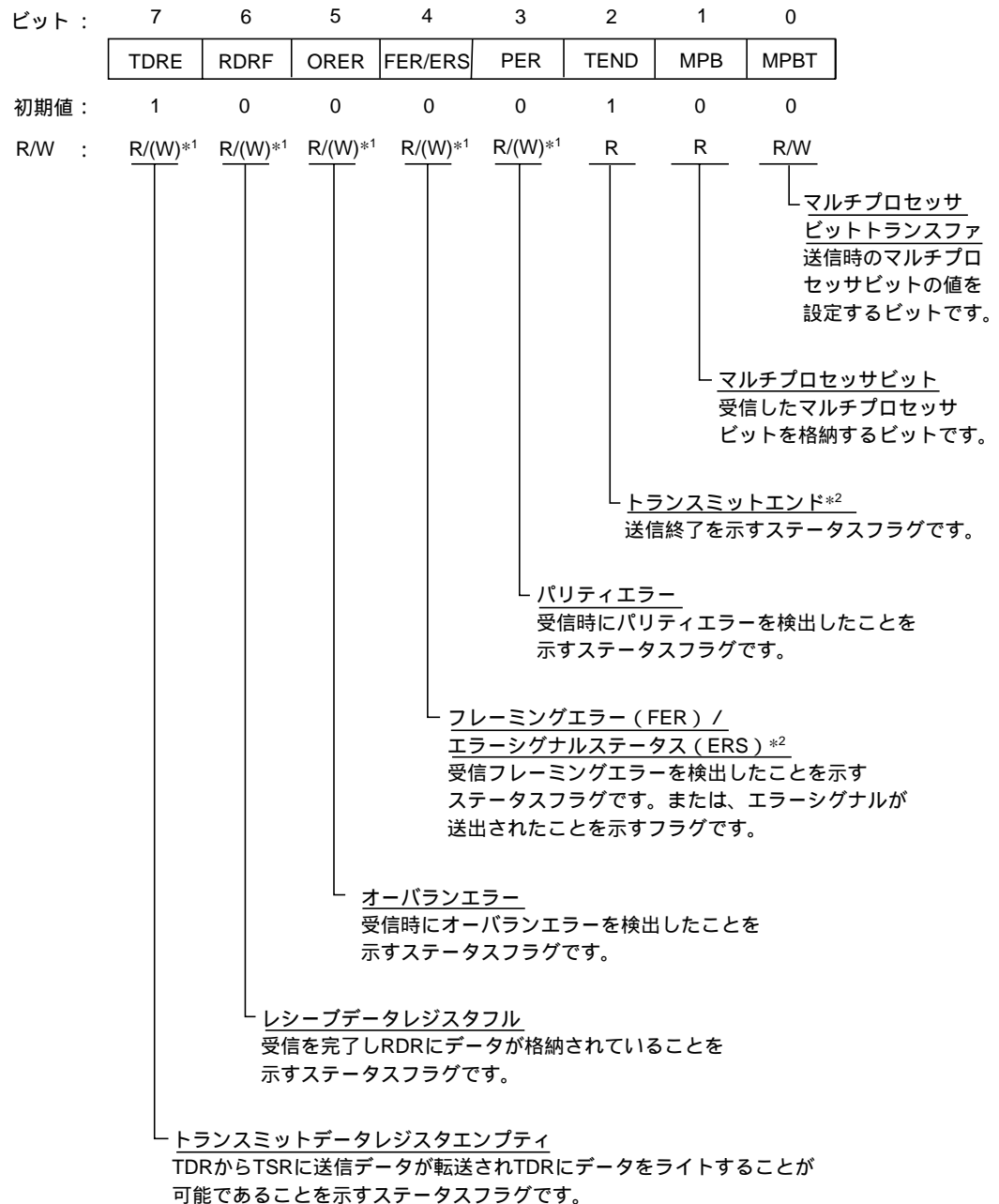
(b) スマートカードインタフェースのとき (SCMR の SMIF ビットが 1)

CKE1 ビットと CKE0 ビットおよび SMR の GM ビットの組み合わせによって SCK 端子を入出力ポートにするか、シリアルクロック出力端子にするかが決まります。

SMR	ビット1	ビット0	説明
GM	CKE1	CKE0	
0	0	0	SCK 端子は入出力ポート (初期値)
0	0	1	SCK 端子はクロック出力
1	0	0	SCK 端子は Low 出力固定
1	0	1	SCK 端子はクロック出力
1	1	0	SCK 端子は High 出力固定
1	1	1	SCK 端子はクロック出力

## 13.2.7 シリアルステータスレジスタ (SSR)

SCIの動作状態を示すステータスフラグと、マルチプロセッサビットを内蔵した8ビットのレジスタです。



【注】\*<sup>1</sup> フラグをクリアするための0ライトのみ可能です。

\*<sup>2</sup> 通常のシリアルコミュニケーションインタフェースとスマートカードインタフェースでは機能が異なります。

SSR は常に CPU からリード/ライトできます。ただし、TDRE、RDRF、ORER、PER、FER の各フラグへ 1 をライトすることはできません。また、これらを 0 にクリアするためには、あらかじめ 1 をリードしておく必要があります。また、TEND フラグ、および MPB フラグはリード専用であり、ライトすることはできません。

SSR は、リセット、またはスタンバイモード時に H'84 にイニシャライズされます。

#### ビット 7：トランスミットデータレジスタエンプティ (TDRE)

TDR から TSR にデータ転送が行われ TDR に次のシリアル送信データをライトすることが可能になったことを示します。

ビット 7	説明
TDRE	
0	TDR に有効な送信データがライトされていることを表示 〔クリア条件〕 (1) TDRE=1 の状態をリードした後、0 をライトしたとき (2) DMAC で TDR ヘデータをライトしたとき
1	TDR に有効な送信データがないことを表示 (初期値) 〔セット条件〕 (1) リセット、またはスタンバイモード時 (2) SCR の TE ビットが 0 のとき (3) TDR から TSR にデータ転送が行われて TDR にデータライトが可能になったとき

## ビット6：レシーブデータレジスタフル（RDRF）

受信したデータがRDRに格納されていることを示します。

ビット6	説明
RDRF	
0	RDRに受信データが格納されていないことを表示 〔クリア条件〕 (1) リセット、またはスタンバイモード時 (2) RDRF=1の状態をリードした後、0をライトしたとき (3) DMACでRDRのデータをリードしたとき (初期値)
1	RDRに受信データが格納されていることを表示 〔セット条件〕 シリアル受信が正常終了し、RSRからRDRへ受信データが転送されたとき

【注】 受信時にエラーを検出したとき、およびSCRのREビットを0にクリアしたときにはRDRおよびRDRFフラグは影響を受けず以前の状態を保持します。

RDRFフラグが1にセットされたまま次のデータを受信完了するとオーバランエラーを発生し、受信データが失われますので注意してください。

## ビット5：オーバランエラー（ORER）

受信時にオーバランエラーが発生して異常終了したことを示します。

ビット5	説明
ORER	
0	受信中、または正常に受信を完了したことを表示 〔クリア条件〕 (1) リセット、またはスタンバイモード時 (2) ORER=1の状態をリードした後、0をライトしたとき (初期値)* <sup>1</sup>
1	受信時にオーバランエラーが発生したことを表示* <sup>2</sup> 〔セット表示〕 RDRF=1の状態での次のシリアル受信を完了したとき

【注】 \*<sup>1</sup> SCRのREビットを0にクリアしたときには、ORERフラグは影響を受けず以前の状態を保持します。

\*<sup>2</sup> RDRではオーバランエラーが発生する前の受信データを保持し、後から受信したデータが失われます。さらに、ORER=1にセットされた状態では、以降のシリアル受信を続けることはできません。なお、クロック同期モードでは、シリアル送信を受けることもできません。

## ビット4：フレーミングエラー（FER） / エラーシグナルステータス（ERS）

通常のシリアルコミュニケーションインタフェースとスマートカードインタフェースでは、機能が異なります。また、SCMRのSMIFビットによって切り替わります。

## (a) シリアルコミュニケーションインタフェースのとき（SCMRのSMIFビットが0）

調歩同期式モードで受信時にフレーミングエラーが発生して異常終了したことを示します。

ビット4	説明
FER	
0	受信中、または正常に受信を完了したことを表示。 (初期値)* <sup>1</sup> [クリア条件] (1) リセット、またはスタンバイモード時 (2) FER=1の状態をリードした後、0をライトしたとき
1	受信時にフレーミングエラーが発生したことを表示 [セット条件] SCIが受信終了時に受信データの最終尾のストップビットが1であるかどうかをチェックし、ストップビットが0であったとき* <sup>2</sup>

【注】 \*1 SCRのREビットを0にクリアしたときには、FERフラグは影響をうけずに以前の状態を保持します。

\*2 2ストップビットモードのときは、1ビット目のストップビットが1であるかどうかのみを判断し、2ビット目のストップビットはチェックをしません。なお、フレーミングエラーが発生したときの受信データはRDRに転送されますが、RDRFフラグはセットされません。さらに、FERフラグが1にセットされた状態においては、以降のシリアル受信を続けることができません。また、クロック同期式モードでは、シリアル送信も続けることができません。

## (b) スマートカードインタフェースのとき (SCMR の SMIF ビットが 1)

スマートカードインタフェースモードでは、送信時に受信側から送り返されるエラーシグナルのステータスを示します。なお、スマートカードインタフェースではフレーミングエラーは検出しません。

ビット 4	説明
ERS	
0	正常に受信され、エラーシグナルが無いことを表示 (初期値)* <sup>1</sup> 〔クリア条件〕 (1) リセット、またはスタンバイモード時 (2) ERS=1 の状態をリードした後、0 をライトしたとき
1	受信側からパリティエラーの検出を示すエラーシグナルが送信されたことを表示 〔セット条件〕 エラーシグナル Low をサンプリングしたとき

【注】 \*1 SCR の TE ビットを 0 にクリアしたときには、ERS フラグは影響を受けずに以前の状態を保持します。

## ビット 3 : パリティエラー (PER)

調歩同期式モードで、パリティを付加した受信時にパリティが発生して異常終了したことを示します。

ビット 3	説明
PER	
0	受信中、または正常に受信を完了したことを表示* <sup>1</sup> (初期値) 〔クリア条件〕 (1) リセット、またはスタンバイモード時 (2) PER=1 の状態をリードした後、0 をライトしたとき
1	受信時にパリティエラーが発生したことを表示* <sup>2</sup> 〔セット条件〕 受信時の受信データとパリティビットをあわせた 1 の数が、SMR の O/E ビットで指定した偶数パリティ / 奇数パリティの設定と一致しなかったとき

【注】 \*1 SCR の RE ビットを 0 にクリアしたときには、PER フラグは影響を受けずに以前の状態を保持します。

\*2 パリティエラーが発生したときの受信データは RDR に転送されますが、RDRF フラグはセットされません。なお、PER フラグが 1 にセットされた状態では、以降のシリアル受信を続けることはできません。さらに、クロック同期式モードでは、シリアル送信も続けることができません。



## ビット2：トランスミットエンド（TEND）

通常のシリアルコミュニケーションインタフェースとスマートカードインタフェースでは、機能が異なります。また、SCMRのSMIFビットによって切り替わります。

## (a) シリアルコミュニケーションインタフェースのとき（SCMRのSMIFビットが0）

送信キャラクタの最終尾の送信時にTDRに有効なデータがなく、送信を終了したことを示します。TENDフラグはリード専用ですので、ライトすることはできません。

ビット2	説明
TEND	
0	送信中であることを表示 [クリア条件] (1) TDRE=1の状態をリードした後、TDREフラグに0をライトしたとき (2) DMACでTDRヘデータをライトしたとき
1	送信を終了したことを表示 (初期値) [セット条件] (1) リセット、またはスタンバイモードのとき (2) SCRのTEビットが0のとき (3) 1バイトのシリアル送信キャラクタの最後尾ビットの送信時にTDRE=1であったとき

## (b) スマートカードインタフェースのとき（SCMRのSMIFビットが1）

送信キャラクタの最後尾ビットの送信時にTDRに有効なデータがなく、送信を終了したことを示します。TENDフラグはリード専用ですので、ライトすることはできません。

ビット2	説明
TEND	
0	送信中であることを表示 [クリア条件] (1) TDRE=1の状態をリードした後、TDREフラグに0をライトしたとき (2) DMACでTDRヘデータをライトしたとき
1	送信を終了したことを表示 (初期値) [セット条件] (1) リセット、またはスタンバイモードのとき (2) SCRのTEビットが0かつFER/ERSビットが0のとき (3) 1バイトのシリアルキャラクタ送信終了2.5etu後(GM=0のとき)/1.0etu後(GM=1のとき)にTDRE=1かつFER/ERSビット=0(正常送信)のとき

【注】 etu (Elementary Time Unit) : 1ビットの転送期間

### ビット1：マルチプロセッサビット（MPB）

調歩同期式モードで受信をマルチプロセッサフォーマットで行うときに、受信データ中のマルチプロセッサビットを格納します。

MPB ビットは、リード専用であり、ライトすることはできません。

ビット1	説明
MPB	
0	マルチプロセッサビットが0のデータを受信したことを表示*（初期値）
1	マルチプロセッサビットが1のデータを受信したことを表示

【注】\* マルチプロセッサフォーマットでRE ビットを0にクリアしたときには、以前の状態を保持します。

### ビット0：マルチプロセッサビットトランスファ（MPBT）

調歩同期式モードで送信をマルチプロセッサフォーマットで行うときに、送信データに付加するマルチプロセッサビットを格納します。

クロック同期式モードやマルチプロセッサフォーマットでないとき、あるいは送信できないときにはMPBT ビットの設定は無効です。

ビット0	説明
MPBT	
0	マルチプロセッサビットが0のデータを送信（初期値）
1	マルチプロセッサビットが1のデータを送信

### 13.2.8 ビットレートレジスタ (BRR)

BRR は、SMR の CKS1、CKS0 ビットで選択されるボーレートジェネレータの動作クロックとあわせて、シリアル送信 / 受信のビットレートを設定する 8 ビットのレジスタです。

ビット	:	7	6	5	4	3	2	1	0										
		<table border="1" style="width: 100%; height: 20px;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> </tr> </table>																	
初期値	:	1	1	1	1	1	1	1	1										
R/W	:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W										

BRR は、常に CPU によるリード / ライトが可能です。

BRR は、リセット、またはスタンバイモード時に H'FF にイニシャライズされます。

なお、チャンネルごとにボーレートジェネレータの制御が独立していますので、それぞれ異なる値を設定することができます。

表 13.3 に調歩同期式モードの BRR の設定例を、表 13.4 にクロック同期式モードの BRR の設定例を示します。

表 13.3 ビットレートに対する BRR の設定例〔調歩同期式モード〕

ビットレート (bit/s)	(MHz)											
	2			2.097152			2.4576			3		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	1	141	0.03	1	148	- 0.04	1	174	- 0.26	1	212	0.03
150	1	103	0.16	1	108	0.21	1	127	0.00	1	155	0.16
300	0	207	0.16	0	217	0.21	0	255	0.00	1	77	0.16
600	0	103	0.16	0	108	0.21	0	127	0.00	0	155	0.16
1200	0	51	0.16	0	54	- 0.70	0	63	0.00	0	77	0.16
2400	0	25	0.16	0	26	1.14	0	31	0.00	0	38	0.16
4800	0	12	0.16	0	13	- 2.48	0	15	0.00	0	19	- 2.34
9600	0	6	- 6.99	0	6	- 2.48	0	7	0.00	0	9	- 2.34
19200	0	2	8.51	0	2	13.78	0	3	0.00	0	4	- 2.34
31250	0	1	0.00	0	1	4.86	0	1	22.88	0	2	0.00
38400	0	1	- 18.62	0	1	- 14.67	0	1	0.00	-	-	-

ビットレート (bit/s)	(MHz)											
	3.6864			4			4.9152			5		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	64	0.70	2	70	0.03	2	86	0.31	2	88	- 0.25
150	1	191	0.00	1	207	0.16	1	255	0.00	2	64	0.16
300	1	95	0.00	1	103	0.16	1	127	0.00	1	129	0.16
600	0	191	0.00	0	207	0.16	0	255	0.00	1	64	0.16
1200	0	95	0.00	0	103	0.16	0	127	0.00	0	129	0.16
2400	0	47	0.00	0	51	0.16	0	63	0.00	0	64	0.16
4800	0	23	0.00	0	25	0.16	0	31	0.00	0	32	- 1.36
9600	0	11	0.00	0	12	0.16	0	15	0.00	0	15	1.73
19200	0	5	0.00	0	6	- 6.99	0	7	0.00	0	7	1.73
31250	-	-	-	0	3	0.00	0	4	- 1.70	0	4	0.00
38400	0	2	0.00	0	2	8.51	0	3	0.00	0	3	1.73

ビットレート (bit/s)	(MHz)											
	6			6.144			7.3728			8		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	106	-0.44	2	108	0.08	2	130	-0.07	2	141	0.03
150	2	77	0.16	2	79	0.00	2	95	0.00	2	103	0.16
300	1	155	0.16	1	159	0.00	1	191	0.00	1	207	0.16
600	1	77	0.16	1	79	0.00	1	95	0.00	1	103	0.16
1200	0	155	0.16	0	159	0.00	0	191	0.00	0	207	0.16
2400	0	77	0.16	0	79	0.00	0	95	0.00	0	103	0.16
4800	0	38	0.16	0	39	0.00	0	47	0.00	0	51	0.16
9600	0	19	-2.34	0	19	0.00	0	23	0.00	0	25	0.16
19200	0	9	-2.34	0	9	0.00	0	11	0.00	0	12	0.16
31250	0	5	0.00	0	5	2.40	0	6	5.33	0	7	0.00
38400	0	4	-2.34	0	4	0.00	0	5	0.00	0	6	-6.99

ビットレート (bit/s)	(MHz)											
	9.8304			10			12			12.288		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	174	-0.26	2	177	-0.25	2	212	0.03	2	217	0.08
150	2	127	0.00	2	129	0.16	2	155	0.16	2	159	0.00
300	1	255	0.00	2	64	0.16	2	77	0.16	2	79	0.00
600	1	127	0.00	1	129	0.16	1	155	0.16	1	159	0.00
1200	0	255	0.00	1	64	0.16	1	77	0.16	1	79	0.00
2400	0	127	0.00	0	129	0.16	0	155	0.16	0	159	0.00
4800	0	63	0.00	0	64	0.16	0	77	0.16	0	79	0.00
9600	0	31	0.00	0	32	-1.36	0	38	0.16	0	39	0.00
19200	0	15	0.00	0	15	1.73	0	19	-2.34	0	19	0.00
31250	0	9	-1.70	0	9	0.00	0	11	0.00	0	11	2.40
38400	0	7	0.00	0	7	1.73	0	9	-2.34	0	9	0.00

ビット レート (bit/s)	(MHz)																	
	13			14			14.7456			16			18			20		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	230	-0.08	2	248	-0.17	3	64	0.70	3	70	0.03	3	79	-0.12	3	88	-0.25
150	2	168	0.16	2	181	0.16	2	191	0.00	2	207	0.16	2	233	0.16	3	64	0.16
300	2	84	-0.43	2	90	0.16	2	95	0.00	2	103	0.16	2	116	0.16	2	129	0.16
600	1	168	0.16	1	181	0.16	1	191	0.00	1	207	0.16	1	233	0.16	2	64	0.16
1200	1	84	-0.43	1	90	0.16	1	95	0.00	1	103	0.16	1	116	0.16	1	129	0.16
2400	0	168	0.16	0	181	0.16	0	191	0.00	0	207	0.16	0	233	0.16	1	64	0.16
4800	0	84	-0.43	0	90	0.16	0	95	0.00	0	103	0.16	0	116	0.16	0	129	0.16
9600	0	41	0.76	0	45	-0.93	0	47	0.00	0	51	0.16	0	58	-0.69	0	64	0.16
19200	0	20	0.76	0	22	-0.93	0	23	0.00	0	25	0.16	0	28	1.02	0	32	-1.36
31250	0	12	0.00	0	13	0.00	0	14	-1.70	0	15	0.00	0	17	0.00	0	19	0.00
38400	0	10	-3.82	0	10	3.57	0	11	0.00	0	12	0.16	0	14	-2.34	0	15	1.73

表 13.4 ビットレートに対する BRR の設定例〔クロック同期式モード〕

ビット レート (bit/s)	(MHz)															
	2		4		8		10		13		16		18		20	
	n	N	n	N	n	N	n	N	n	N	n	N	n	N	n	N
110	3	70	-	-	-	-	-	-	-	-	-	-	-	-	-	-
250	2	124	2	249	3	124	-	-	3	202	3	249	-	-	-	-
500	1	249	2	124	2	249	-	-	3	101	3	124	3	140	3	155
1k	1	124	1	249	2	124	-	-	2	202	2	249	3	69	3	77
2.5k	0	199	1	99	1	199	1	249	2	80	2	99	2	112	2	124
5k	0	99	0	199	1	99	1	124	1	162	1	199	1	224	1	249
10k	0	49	0	99	0	199	0	249	1	80	1	99	1	112	1	124
25k	0	19	0	39	0	79	0	99	0	129	0	159	0	179	0	199
50k	0	9	0	19	0	39	0	49	0	64	0	79	0	89	0	99
100k	0	4	0	9	0	19	0	24	-	-	0	39	0	44	0	49
250k	0	1	0	3	0	7	0	9	0	12	0	15	0	17	0	19
500k	0	0*	0	1	0	3	0	4	-	-	0	7	0	8	0	9
1M			0	0*	0	1	-	-	-	-	0	3	0	4	0	4
2M					0	0*	-	-	-	-	0	1	-	-	-	-
2.5M					-	-	0	0*	-	-	-	-	-	-	-	-
4M											0	0*	-	-	-	-

【注】 誤差は、なるべく 1%以内になるように設定してください。

## 《記号説明》

- 空欄 : 設定できません。  
 - : 設定可能ですが誤差がでます。  
 \* : 連続送信 / 受信はできません。

BRR の設定値は以下の計算式で求められます。

〔調歩同期式モード〕

$$N = \frac{\text{---}}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

〔クロック同期式モード〕

$$N = \frac{\text{---}}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B : ビットレート (bit/s)

N : ボーレートジェネレータの BRR の設定値 (0 ~ N ~ 255)

: 動作周波数 (MHz)

n : ボーレートジェネレータ入力クロック (n=0~3)

(n とクロックの関係は下表を参照してください。)

n	クロック	SMR の設定値	
		CKS1	CKS0
0		0	0
1	/4	0	1
2	/16	1	0
3	/64	1	1

調歩同期式モードのビットレート誤差は、以下の計算式で求められます。

$$\text{誤差}(\%) = \left\{ \frac{\text{---} \times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

表 13.5 に調歩同期式モードの各周波数における最大ビットレートを示します。また、表 13.6～表 13.7 に外部クロック入力時の最大ビットレートを示します。

表 13.5 各周波数における最大ビットレート（調歩同期式モード）

(MHz)	最大ビットレート (bit/s)	設定値	
		n	N
2	62500	0	0
2.097152	65536	0	0
2.4576	76800	0	0
3	93750	0	0
3.6864	115200	0	0
4	125000	0	0
4.9152	153600	0	0
5	156250	0	0
6	187500	0	0
6.144	192000	0	0
7.3728	230400	0	0
8	250000	0	0
9.8304	307200	0	0
10	312500	0	0
12	375000	0	0
12.288	384000	0	0
14	437500	0	0
14.7456	460800	0	0
16	500000	0	0
17.2032	537600	0	0
18	562500	0	0
20	625000	0	0



表 13.6 外部クロック入力時の最大ビットレート（調歩同期式モード）

(MHz)	外部入力クロック (MHz)	最大ビットレート (bit/s)
2	0.5000	31250
2.097152	0.5243	32768
2.4576	0.6144	38400
3	0.7500	46875
3.6864	0.9216	57600
4	1.0000	62500
4.9152	1.2288	76800
5	1.2500	78125
6	1.5000	93750
6.144	1.5360	96000
7.3728	1.8432	115200
8	2.0000	125000
9.8304	2.4576	153600
10	2.5000	156250
12	3.0000	187500
12.288	3.0720	192000
14	3.5000	218750
14.7456	3.6864	230400
16	4.0000	250000
17.2032	4.3008	268800
18	4.5000	281250
20	5.0000	312500

表 13.7 外部クロック入力時の最大ビットレート（クロック同期式モード）

(MHz)	外部入力クロック (MHz)	最大ビットレート (bit/s)
2	0.3333	333333.3
4	0.6667	666666.7
6	1.0000	1000000.0
8	1.3333	1333333.3
10	1.6667	1666666.7
12	2.0000	2000000.0
14	2.3333	2333333.3
16	2.6667	2666666.7
18	3.0000	3000000.0
20	3.3333	3333333.3

## 13.3 動作説明

### 13.3.1 概要

SCIは、キャラクタ単位で同期をとりながら通信する調歩同期式モードと、クロックパルスにより同期をとりながら通信するクロック同期式モードの2方式で、シリアル通信ができます。また、ICカードインタフェース用シリアル通信機能として、スマートカードインタフェースをサポートしています。

通常のシリアルコミュニケーションインタフェースでの調歩同期式モードと、クロック同期式モードの選択および送信フォーマットの選択は、SMRで行います。これを表13.8に示します。また、SCIのクロックソースは、SMRの $C/\bar{A}$ ビットおよびSCRのCKE1、CKE0ビットの組み合わせできまります。これを表13.9に示します。

LSBファースト/MSBファーストの切り換え方法およびデータのロジックレベルの反転方法の詳細については、「14.2.1 スマートカードモードレジスタ(SCMR)」を参照してください。

また、スマートカードモードインタフェースのフォーマットの選択は「14.3.3 データフォーマット」を参照してください。

#### (1) 調歩同期式モード

データ長：7ビット/8ビットから選択可能

パリティの付加、マルチプロセッサビットの付加、および1ビット/2ビットのストップビットの付加を選択可能(これらの組み合わせにより送信/受信フォーマットおよび、キャラクタ長を決定)

受信時にフレーミングエラー、パリティエラー、オーバランエラー、およびブレークの検出が可能

SCIのクロックソース：内部クロック/外部クロックから選択可能

- ・内部クロックを選択した場合：SCIはボーレートジェネレータのクロックで動作し、ビットレートと同じ周波数のクロックを出力することが可能
- ・外部クロックを選択した場合：ビットレートの16倍の周波数のクロックを入力することが必要(内蔵ボーレートジェネレータを使用しない)

## (2) クロック同期式モード

送信 / 受信フォーマット : 8 ビットデータ固定

受信時にオーバーランエラーの検出可能

SCI のクロックソース : 内部クロック / 外部クロックから選択可能

- ・ 内部クロックを選択した場合 : SCI はボーレートジェネレータのクロックで動作し、同期クロックを外部へ出力
- ・ 外部クロックを選択した場合 : 内部ボーレートジェネレータを使用せず、入力された同期クロックで動作

## (3) スマートカードインタフェース

1 フレームは、8 ビットデータとパリティビットで構成されます。

送信時は、パリティビットの終了から次のフレーム開始まで 2etu (Elementary Time Unit : 1 ビットの転送期間) 以上のガードタイムをおきます。

受信時はパリティエラーを検出した場合、スタートビットから 10.5etu 経過後エラーシグナル Low を 1etu 期間出力します。

送信時はエラーシグナルをサンプリングすると、2etu 以上経過後、自動的に同じデータを送信します。

調歩同期式非同期通信機能のみをサポートし、クロック同期式通信機能はありません。

スマートカードインタフェースの動作説明の詳細については、「14. スマートカードインタフェース」を参照してください。

表 13.8 SMR の設定値とシリアル送信 / 受信フォーマット

SMR の設定値					モード	SCI の送信 / 受信フォーマット									
ビット 7	ビット 6	ビット 2	ビット 5	ビット 3		データ長	マルチプロセッサビット	パリティビット	ストップビット長						
C/ $\bar{A}$	CHR	MP	PE	STOP											
0	0	0	0	0	調歩同期式モード	8 ビットデータ	なし	なし	1 ビット						
				1					2 ビット						
			1	0					0	あり	1 ビット				
									1	2 ビット					
			1	0					0	7 ビットデータ	なし	1 ビット			
									1			2 ビット			
	1	0	0	あり		なし	1 ビット								
			1				2 ビット								
	0	1	-				0	調歩同期式モード (マルチプロセッサフォーマット)	8 ビットデータ			あり	なし	1 ビット	
							1							2 ビット	
	1	-	0				0		7 ビットデータ	なし	あり			なし	1 ビット
							1								2 ビット
1	-	-	0	クロック同期式モード	8 ビットデータ	なし	なし								なし
			1												なし

表 13.9 SMR、SCR の設定と SCI クロックソースの選択

SMR	SCR の設定		モード	SCI 送信 / 受信クロック		
	ビット 7	ビット 1		ビット 0	クロックソース	SCK 端子の機能
C/ $\bar{A}$	CKE1	CKE0				
0	0	0	調歩同期式モード	内部	SCI は、SCK 端子を使用しません	
		1			ビットレートと同じ周波数のクロックを出力	
		1	0		外部	ビットレートの 16 倍の周波数のクロックを入力
			1			
1	0	0	クロック同期式モード	内部	同期クロックを出力	
		1				
	1	0	外部		同期クロックを入力	
		1				

### 13.3.2 調歩同期式モード時の動作

調歩同期式モードは、通信開始を意味するスタートビットと通信終了を意味するストップビットとをデータに付加したキャラクタを送信 / 受信し、1キャラクタ単位で同期をとりながらシリアル通信を行うモードです。

SCI 内部では、送信部と受信部は独立していますので、全二重通信を行うことができます。また、送信部と受信部が共にダブルバッファ構造になっていますので、送信 / 受信中にデータのリード / ライトができるので、連続送信 / 受信が可能です。

調歩同期式シリアル通信の一般的なフォーマットを図 13.2 に示します。

調歩同期式シリアル通信では、通信回線は通常、マーク状態 (High レベル) に保たれています。SCI は通信回線を監視し、スペース (Low レベル) になったところをスタートビットとみなしてシリアル通信を開始します。

シリアル通信の 1キャラクタは、スタートビット (Low レベル) から始まり、データ (LSB ファースト : 最下位ビットから)、パリティビット (High / Low レベル)、最後にストップビット (High レベル) の順で構成されています。

調歩同期式モードでは、SCI は受信時にスタートビットの立下がりエッジで同期化を行います。また SCI は、データを 1 ビット期間の 16 倍の周波数のクロックの 8 番目でサンプリングしますので、各ビットの中央で通信データが取り込まれます。

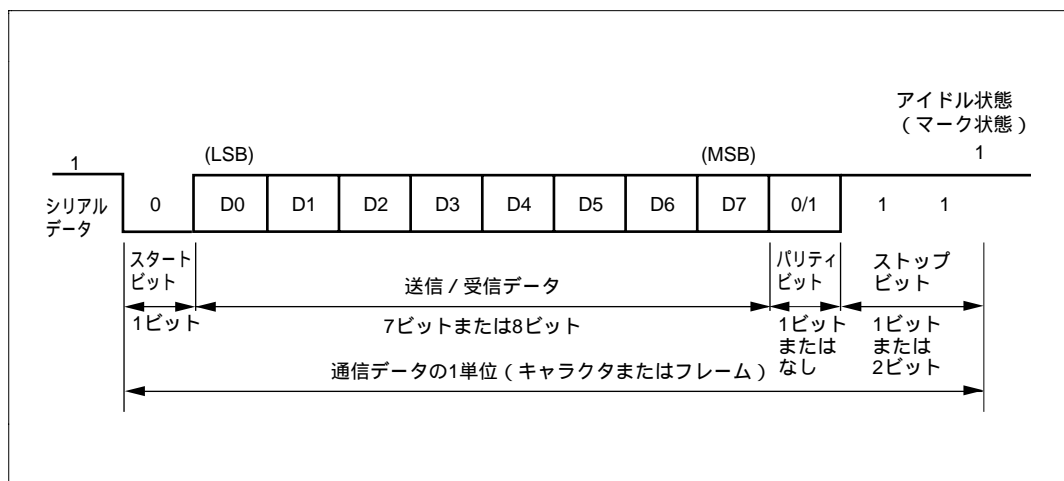


図 13.2 調歩同期式通信のデータフォーマット  
(8ビットデータ / パリティあり / 2ストップビットの例)

## (1) 送信 / 受信フォーマット

調歩同期式モードで設定できる送信 / 受信フォーマットを、表 13.10 に示します。

送信 / 受信フォーマットは 12 種類あり、SMR の設定により選択できます。

表 13.10 シリアル送信 / 受信フォーマット (調歩同期式モード)

SMRの設定				シリアル送信受信フォーマットとフレーム長													
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12		
0	0	0	0	S	8ビットデータ								STOP				
0	0	0	1	S	8ビットデータ								STOP	STOP			
0	1	0	0	S	8ビットデータ								P	STOP			
0	1	0	1	S	8ビットデータ								P	STOP	STOP		
1	0	0	0	S	7ビットデータ							STOP					
1	0	0	1	S	7ビットデータ							STOP	STOP				
1	1	0	0	S	7ビットデータ							P	STOP				
1	1	0	1	S	7ビットデータ							P	STOP	STOP			
0	-	1	0	S	8ビットデータ								MPB	STOP			
0	-	1	1	S	8ビットデータ								MPB	STOP	STOP		
1	-	1	0	S	7ビットデータ							MPB	STOP				
1	-	1	1	S	7ビットデータ							MPB	STOP	STOP			

## 《記号説明》

S : スタートビット

STOP : ストップビット

P : パリティビット

MPB : マルチプロセッサビット

## (2) クロック

SCIの送受信クロックは、SMRの $C/\bar{A}$ ビットとSCRのCKE1、CKE0ビットの設定により、内蔵ポーレートジェネレータの生成した内部クロックまたは、SCK端子から入力された外部クロックの2種類から選択できます。SCIのクロックソースについては表13.9を参照してください。

外部クロックをSCK端子に入力する場合には、使用するビットレートの16倍の周波数のクロックを入力してください。

内部クロックで動作させるとき、SCK端子からクロックを出力することができます。このとき出力されるクロックの周波数はビットレートと等しく、位相は図13.3に示すように送信データの中央にクロック立上がりエッジがくるようになります。

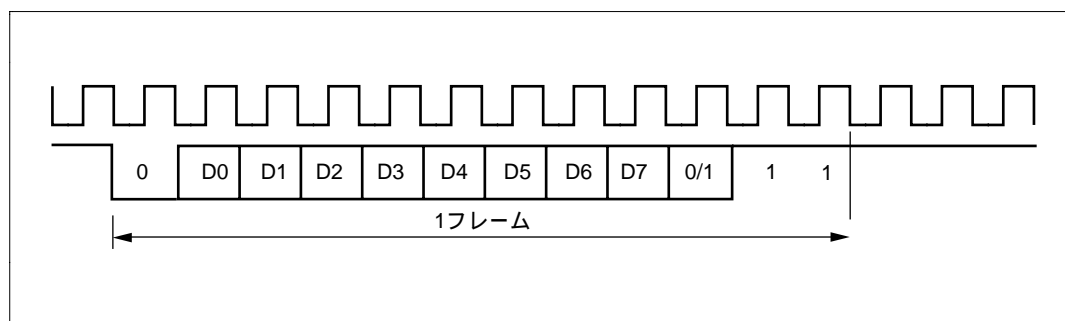


図13.3 出力クロックと通信データの位相関係（調歩同期式モード）

## (3) データの送信 / 受信動作

### (a) SCIのイニシャライズ（調歩同期式）

データの送信 / 受信前には、まずSCRのTE、REビットを0にクリアした後、以下の順でSCIをイニシャライズしてください。

動作モードの変更、通信フォーマットの変更などの場合には必ず、TEビットおよびREビットを0にクリアしてから次の手順で変更を行ってください。TEビットを0にクリアするとTDREフラグは1にセットされ、TSRがイニシャライズされます。REビットを0にクリアしても、RDRF、PER、FER、ORERの各フラグおよび、RDRの内容は保持されますので注意してください。

外部クロックを使用している場合には、動作が不確実になりますのでイニシャライズを含めた動作中にクロックを止めないでください。



図 13.4 に SCI のイニシャライズフローチャートの例を示します。

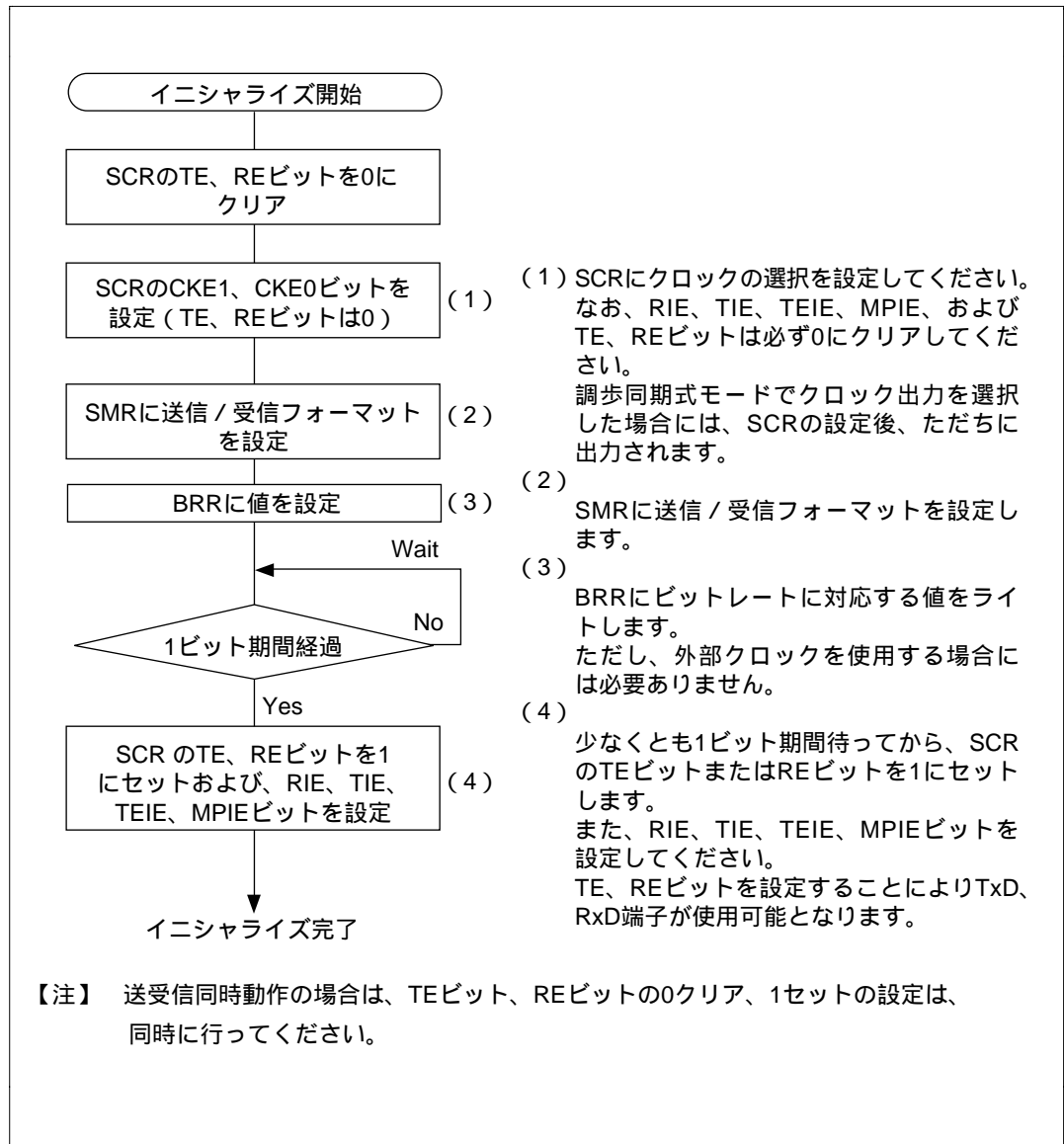


図 13.4 SCI のイニシャライズフローチャートの例

## (b) シリアルデータ送信（調歩同期式）

図 13.5 にシリアル送信のフローチャートの例を示します。

シリアルデータ送信は以下の手順に従って行ってください。

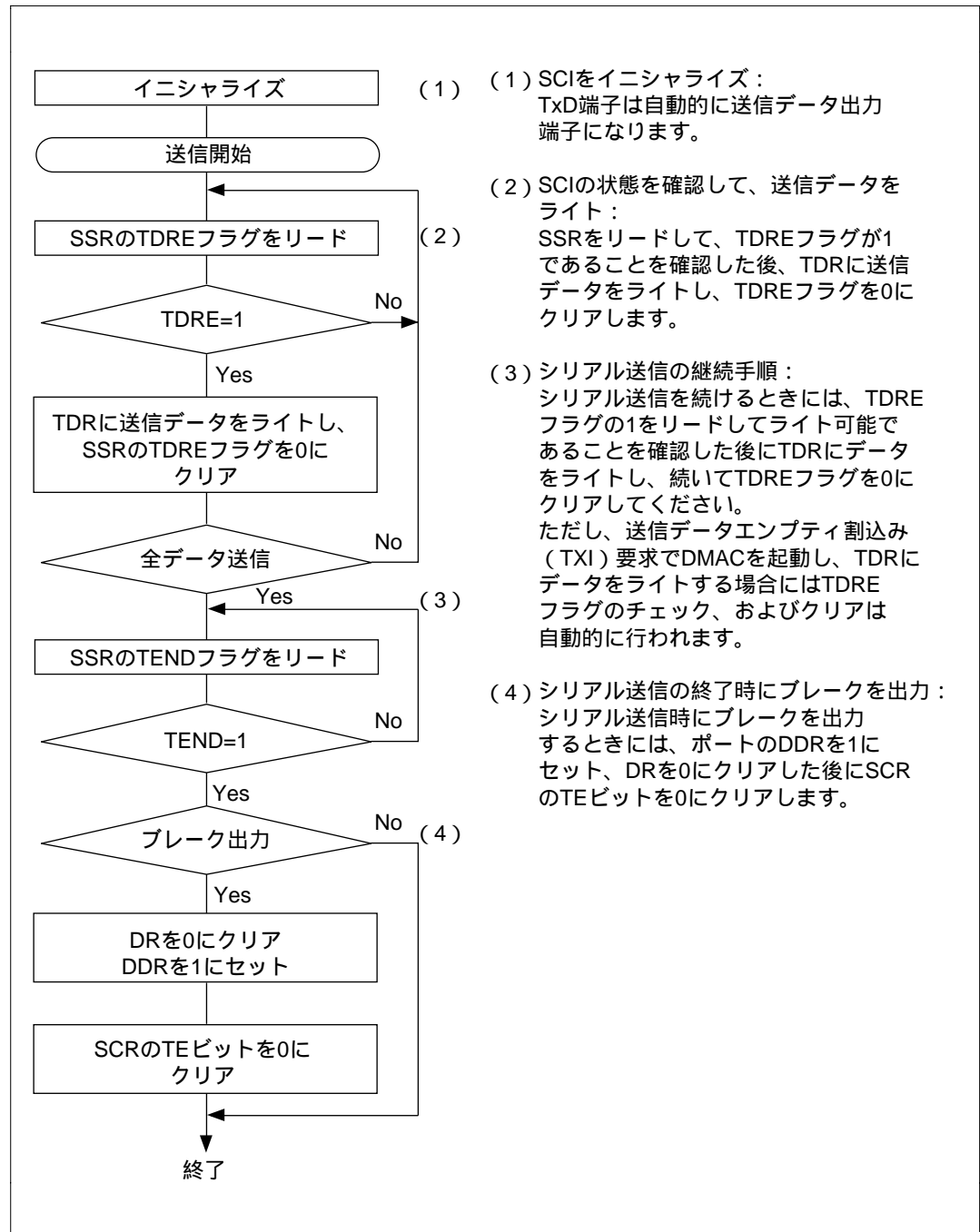


図 13.5 シリアル送信のフローチャートの例

SCI はシリアル送信時に以下のように動作します。

- (1) SCI は、SSR の TDRE フラグを監視し、0 であると TDR にデータがライトされたと認識し、TDR から TSR にデータを転送します。
- (2) TDR から TSR へデータを転送した後に TDRE フラグを 1 にセットし、送信を開始します。  
このとき、SCR の TIE ビットが 1 にセットされていると送信データエンブティ割込み (TXI) 要求を発生します。  
シリアル送信データは、以下の順に TxD 端子から送り出されます。
  - (a) スタートビット：1 ビットの 0 が出力されます。
  - (b) 送信データ：8 ビット、または 7 ビットのデータが LSB から順に出力されます。
  - (c) パリティビットまたはマルチプロセッサビット：1 ビットのパリティビット (偶数パリティ、または奇数パリティ)、または 1 ビットのマルチプロセッサビットが出力されます。なお、パリティビット、またはマルチプロセッサビットを出力しないフォーマットも選択できます。
  - (d) ストップビット：1 ビット / 2 ビットの 1 (ストップビット) が出力されます。
  - (e) マーク状態：次の送信を開始するスタートビットを送り出すまで 1 を出力し続けます。
- (3) SCI は、ストップビットを送出するタイミングで TDRE フラグをチェックします。  
TDRE フラグが 0 であると TDR から TSR にデータを転送し、ストップビットを送り出した後、次フレームのシリアル送信を開始します。  
TDRE フラグが 1 であると SSR の TEND フラグに 1 をセットし、ストップビットを送り出した後、1 を出力する "マーク状態" になります。このとき SCR の TEIE ビットが 1 にセットされていると TEI 割込み要求を発生します。

調歩同期式モードでの送信時の動作例を図 13.6 に示します。

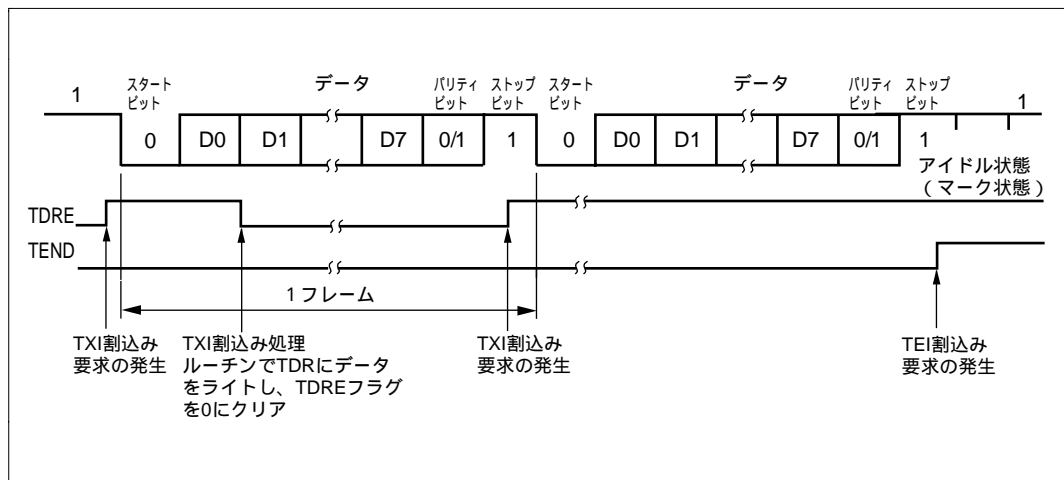


図 13.6 調歩同期式モードでの送信時の動作例  
(8ビットデータ/パリティあり/1ストップビットの例)

(c) シリアルデータ受信 (調歩同期式)

図 13.7 にシリアル受信フローチャートの例を示します。

シリアルデータ受信は以下の手順に従って行ってください。

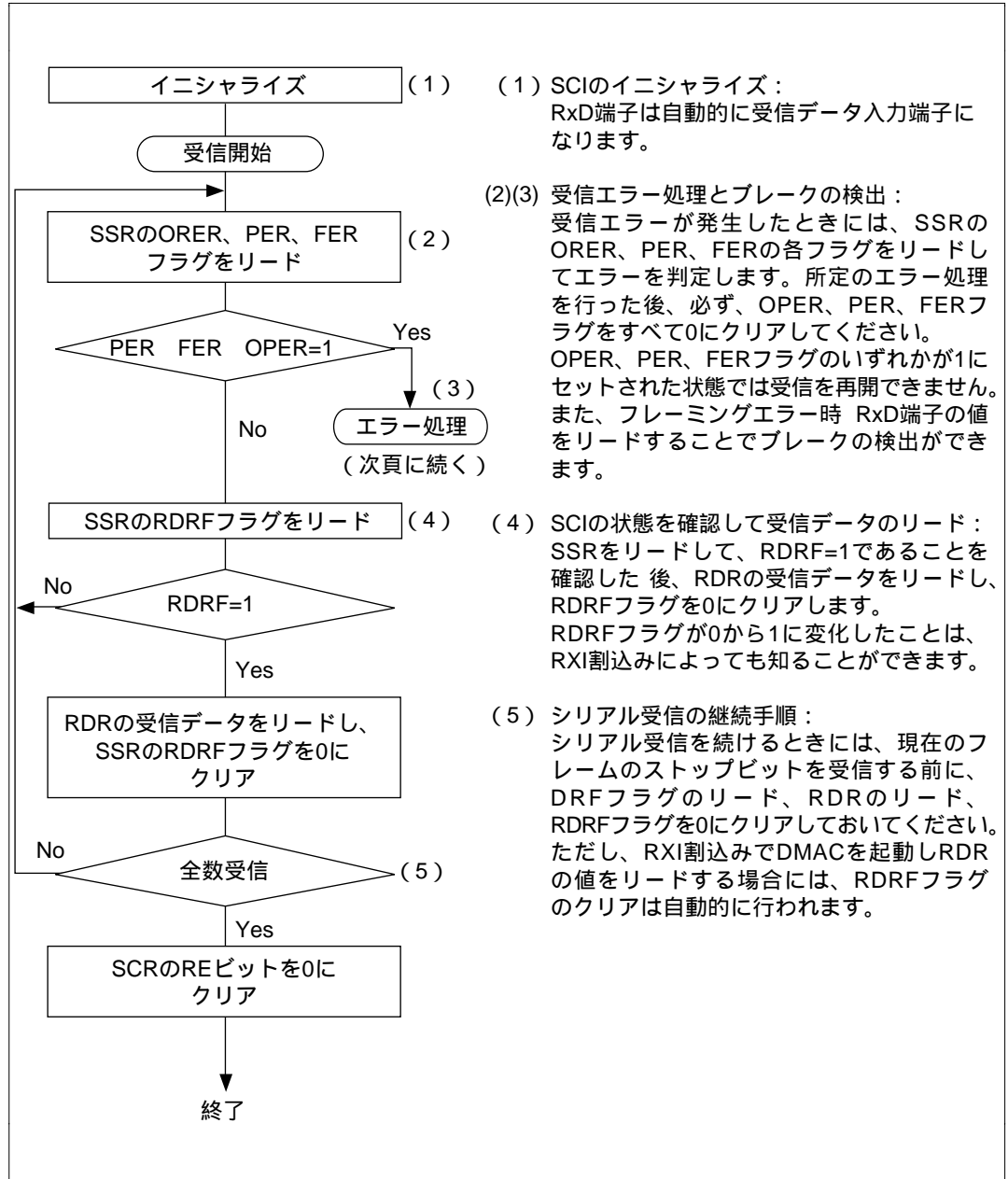


図 13.7 シリアル受信データフローチャートの例 (1)

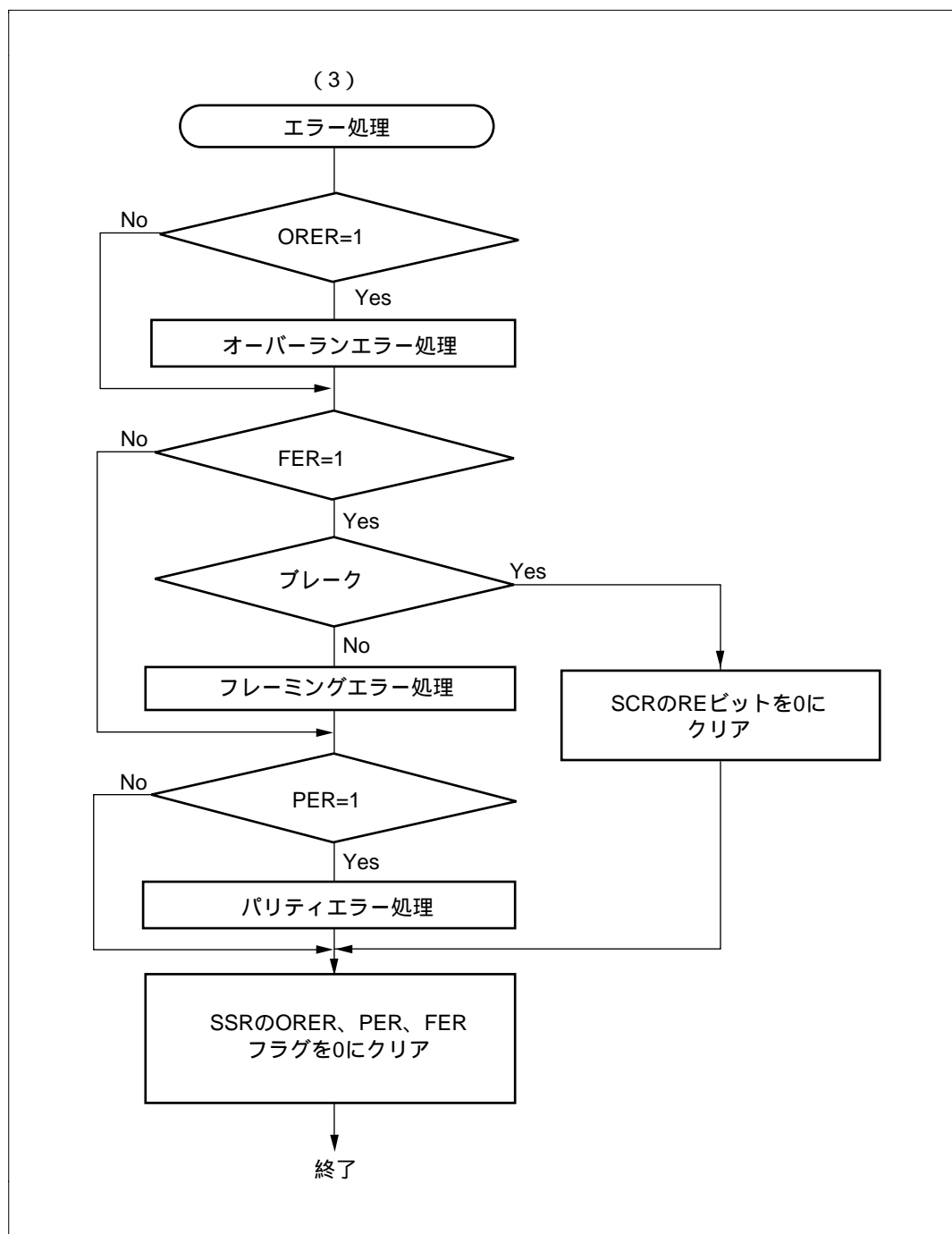


図 13.7 シリアル受信データフローチャートの例 (2)

SCI は受信時に以下のように動作します。

- (1) SCI は通信回線を監視し、スタートビットの 0 を検出すると内部を同期化し、受信を開始します。
- (2) 受信したデータを RSR の LSB から MSB の順に格納します。
- (3) パリティビットおよび、ストップビットを受信します。

受信後、SCI は以下のチェックを行います。

- (a) パリティチェック：受信データの 1 の数をチェックし、これが SMR の  $O\bar{E}$  ビットで設定した偶数 / 奇数パリティになっているかをチェックします。
- (b) ストップビットチェック：ストップビットが 1 であるかをチェックします。  
ただし、2 ストップビットの場合、1 ビット目のストップビットのみをチェックします。
- (c) ステータスチェック：RDRF フラグが 0 であり、受信データを RSR から RDR に転送できる状態であるかをチェックします。

以上のチェックがすべてパスしたとき、RDRF フラグが 1 にセットされ、RDR に受信データが格納されます。

エラーチェックで受信エラー\*を発生すると表 13.11 のように動作します。

- 【注】\* 受信エラーが発生した状態では、以後の受信動作ができません。  
また、受信時に RDRF フラグが 1 にセットされませんので、必ずエラーフラグを 0 にクリアしてください。

- (4) RDRF フラグが 1 になったとき SCR の RIE ビットが 1 にセットされていると受信データフル割込み (RXI) 要求を発生します。  
また、ORER、PER、FER フラグのいずれかが 1 になったとき、SCR の RIE ビットが 1 にセットされていると受信エラー割込み (ERI) 要求を発生します。

表 13.11 受信エラーと発生条件

受信エラー名	略称	発生条件	データ転送
オーバランエラー	ORER	SSR の RDRF フラグが 1 にセットされたまま次のデータ受信を完了したとき	RSR から RDR に受信データは転送されません。
フレーミングエラー	FER	ストップビットが 0 のとき	RSR から RDR に受信データは転送されます。
パリティエラー	PER	SMR で設定した偶数 / 奇数パリティの設定と受信したデータが異なるとき	RSR から RDR に受信データが転送されます。

調歩同期式モード受信時の動作例を図 13.8 に示します。

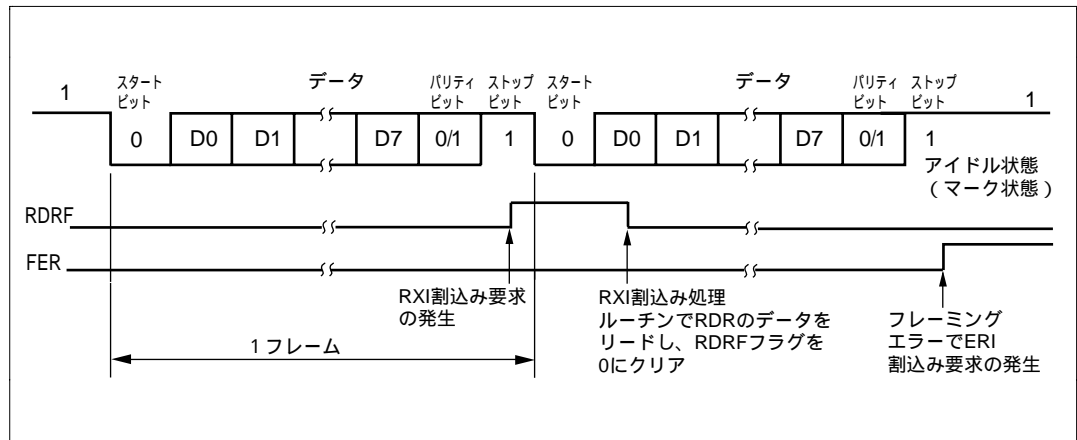


図 13.8 SCI の受信時の動作例（8 ビットデータ / パリティあり / 1 ストップビットの例）

### 13.3.3 マルチプロセッサ通信機能

マルチプロセッサ通信機能とは、調歩同期式モードでマルチプロセッサビットを付加したフォーマット（マルチプロセッサフォーマット）でシリアル通信をする機能です。この機能を使用すると、複数のプロセッサ間でシリアル通信回線を共有したデータの送受信ができます。

マルチプロセッサ通信を行うとき、受信局は各々固有の ID コードでアドレッシングされています。

シリアル通信サイクルは、受信局を指定する ID 送信サイクルとデータ送信サイクルの 2 つから構成されます。この ID 送信サイクルとデータ送信サイクルの区別は、マルチプロセッサビットで行います。

送信局は、まず、シリアル通信を行いたい受信局の ID を、マルチプロセッサビット 1 を付加したデータにして送信します。続いて、送信データを、マルチプロセッサビット 0 を付加したデータにして送信します。

受信局は、マルチプロセッサビット 1 のデータが送信されるまでは、データを読み飛ばします。

マルチプロセッサビット 1 のデータを受信したとき、受信局は自局の ID と比較します。そして、一致した局は続いて送信されるデータを受信します。一方一致しなかった局は、再びマルチプロセッサビット 1 のデータが送信されるまでは、データを読み飛ばします。このようにして複数のプロセッサ間のデータ送受信が行われます。

図 13.9 にマルチプロセッサフォーマットを使用したプロセッサ間通信の例を示します。



## (1) 送信 / 受信フォーマット

送信 / 受信フォーマットは4種類です。

マルチプロセッサフォーマットを指定した場合は、パリティビットの指定は無効です。  
詳細は表 13.10 を参照してください。

## (2) クロック

調歩同期式モードの項を参照してください。

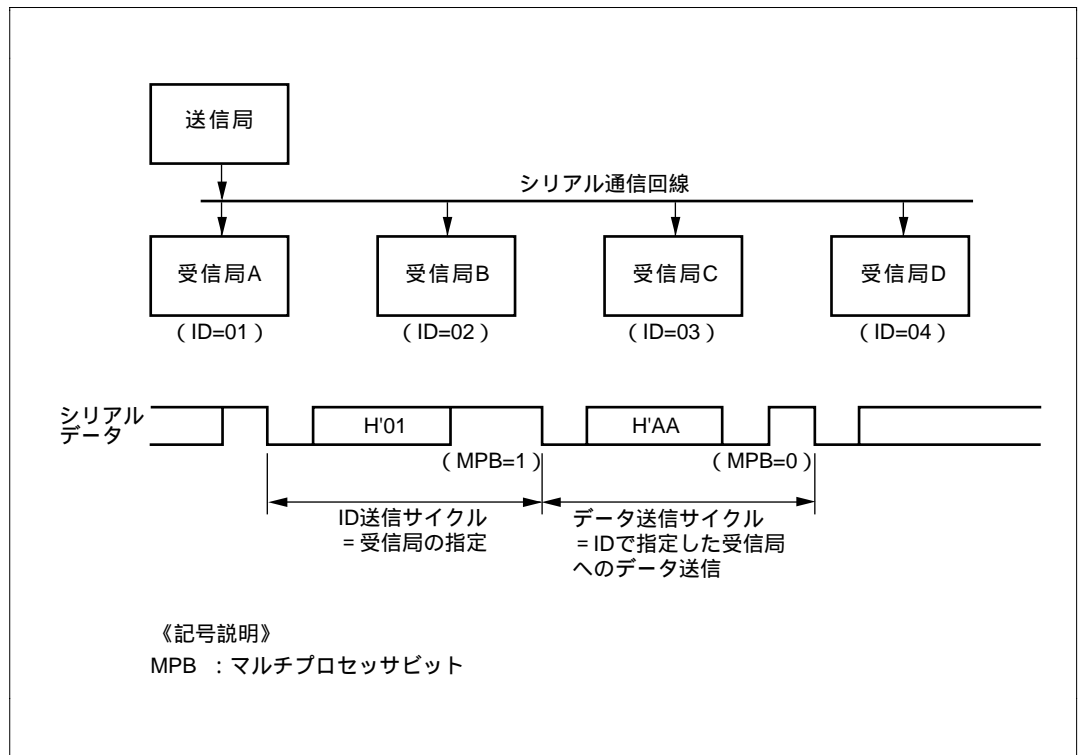


図 13.9 マルチプロセッサフォーマットを使用したプロセッサ間通信の例  
(受信局 A へのデータ H'AA の送信の例)

## (3) データの送信 / 受信動作

## (a) マルチプロセッサシリアルデータ送信

図 13.10 にマルチプロセッサシリアル送信のフローチャートの例を示します。

マルチプロセッサシリアルデータ送信は、以下の手順に従って行ってください。

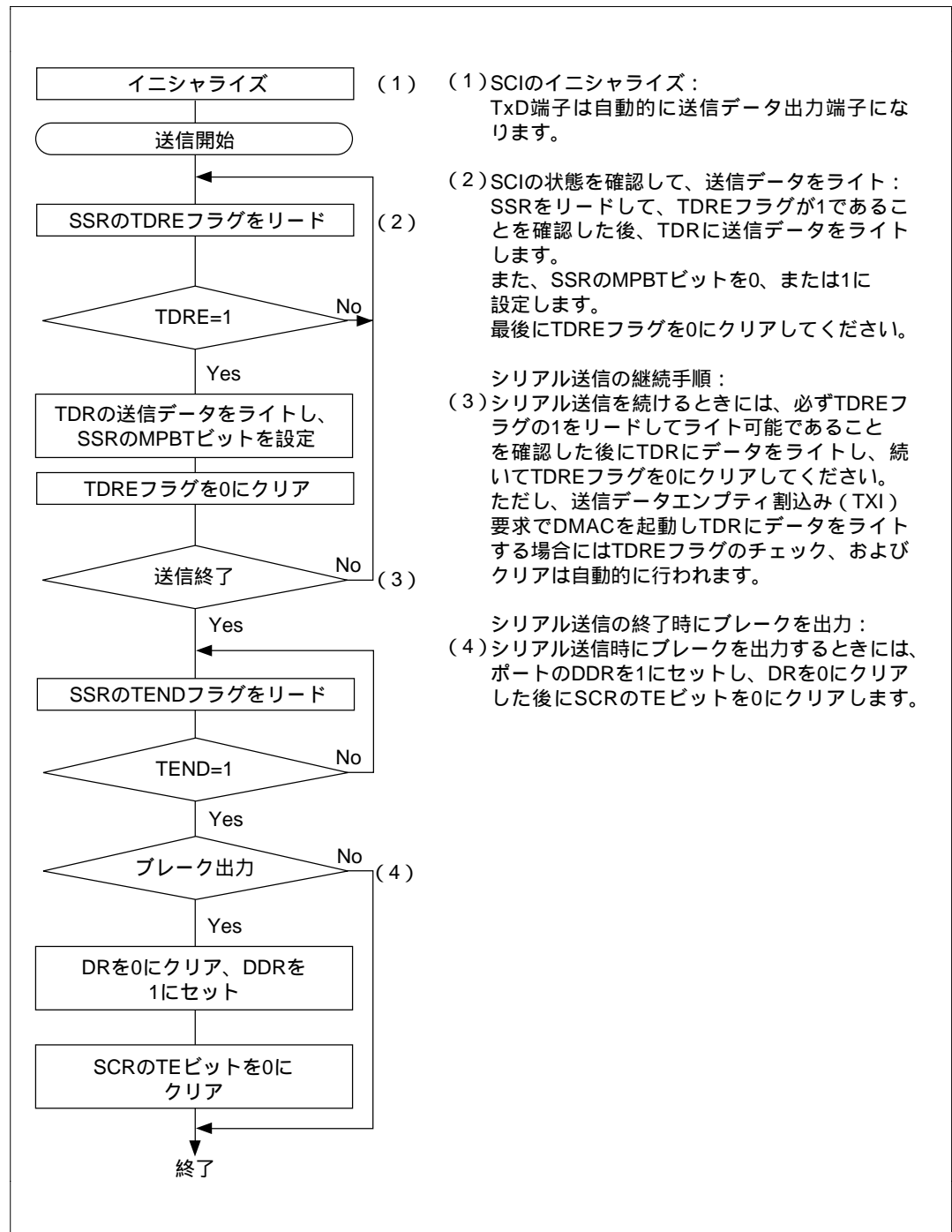


図 13.10 マルチプロセッサシリアル送信のフローチャートの例

SCIは、シリアル送信時に以下のように動作します。

- (1) SCIは、SSRのTDREフラグを監視し、0であるとTDRにデータがライトされたと認識し、TDRからTSRにデータを転送します。
- (2) TDRからTSRへデータを転送した後にTDREフラグを1にセットし、送信を開始します。このとき、SCRのTIEビットが1にセットされていると送信データエンベティ割込み(TXI)要求を発生します。  
 シリアル送信データは、以下の順にTxD端子から送りだされます。
  - (a) スタートビット：1ビットの0が出力されます。
  - (b) 送信データ：8ビット/7ビットのデータがLSBから順に出力されます。
  - (c) マルチプロセッサビット：1ビットのマルチプロセッサビット(MPBTの値)が出力されます。
  - (d) ストップビット：1ビット/2ビットの1(ストップビット)が出力されます。
  - (e) マーク状態：次の送信を開始するスタートビットを送り出すまで1を出力し続けます。
- (3) SCIは、ストップビットを送り出すタイミングでTDREフラグをチェックします。  
 TDREフラグが0であるとTDRからTSRにデータを転送し、ストップビットを送り出した後、次のフレームのシリアル送信を開始します。  
 TDREフラグが1であるとSSRのTENDフラグを1にセットし、ストップビットを送り出した後、1を出力するマーク状態になります。このときSCRのTEIEビットが1にセットされていると送信終了割込み(TEI)要求を発生します。

図 13.11 にマルチプロセッサフォーマットのSCIの送信時の動作例を示します。

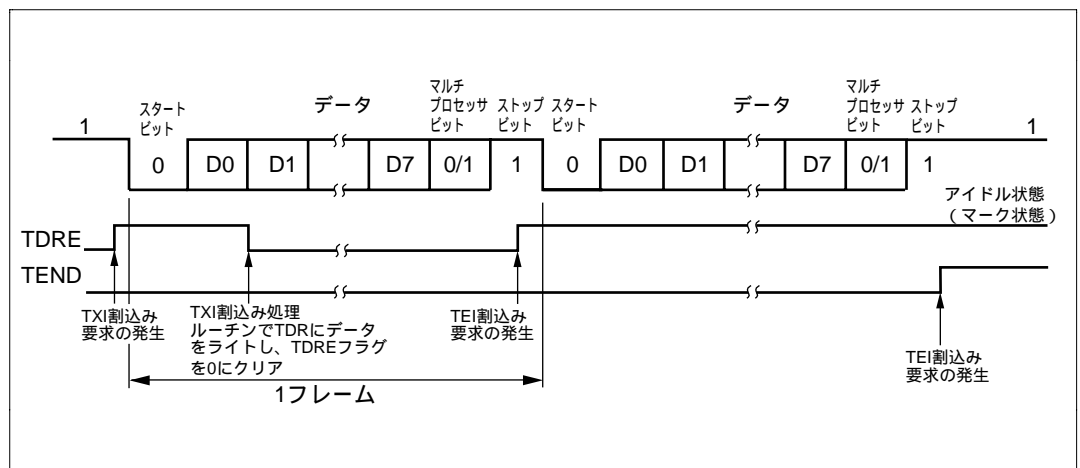


図 13.11 SCIの送信時の動作例  
 (8ビットデータ/マルチプロセッサビットあり/1ストップビットの例)

## (b) マルチプロセッサシリアルデータ受信

図 13.12 にマルチプロセッサシリアル受信のフローチャートの例を示します。  
マルチプロセッサシリアルデータ受信は、以下の手順に従って行ってください。

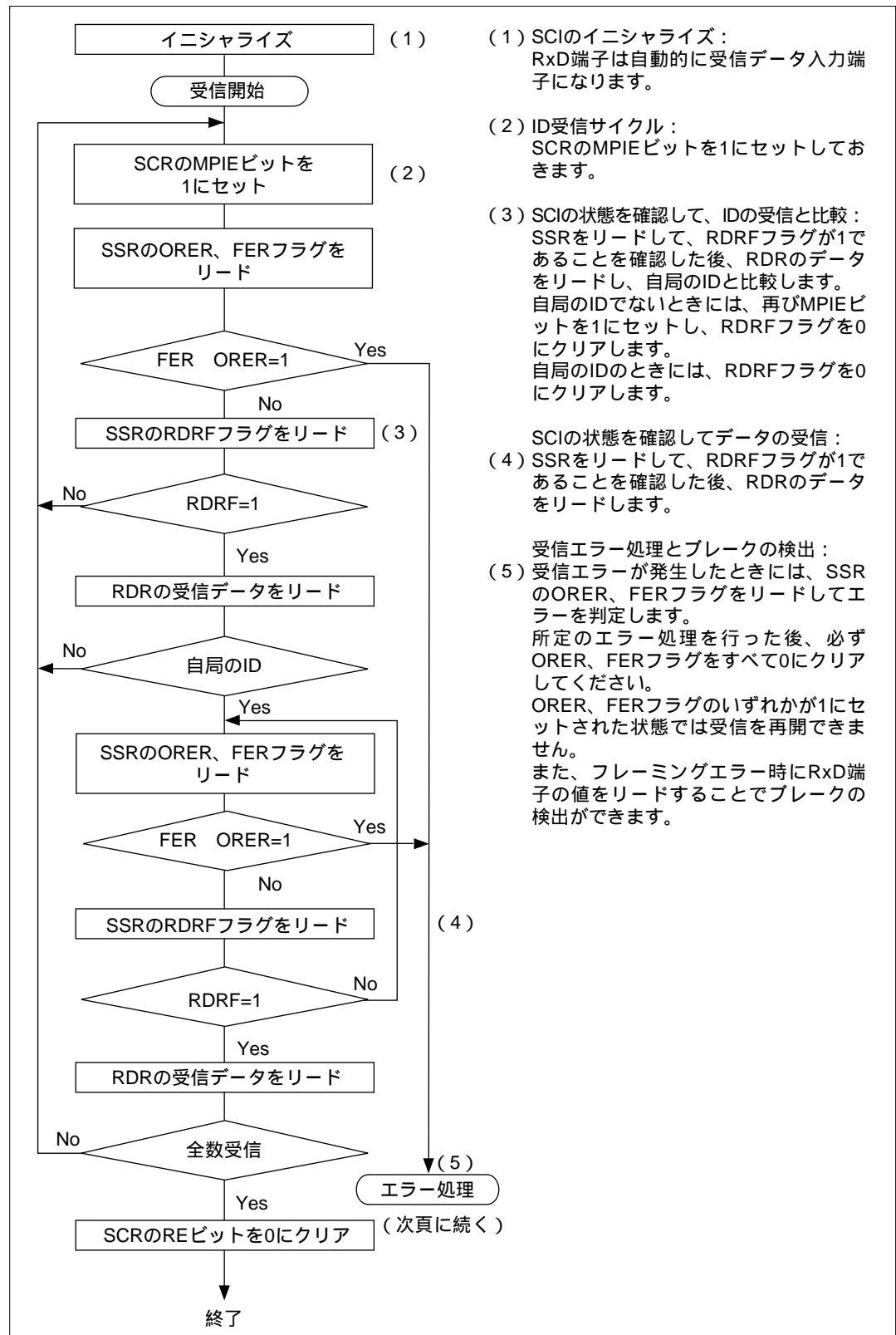


図 13.12 マルチプロセッサシリアル受信のフローチャートの例 (1)

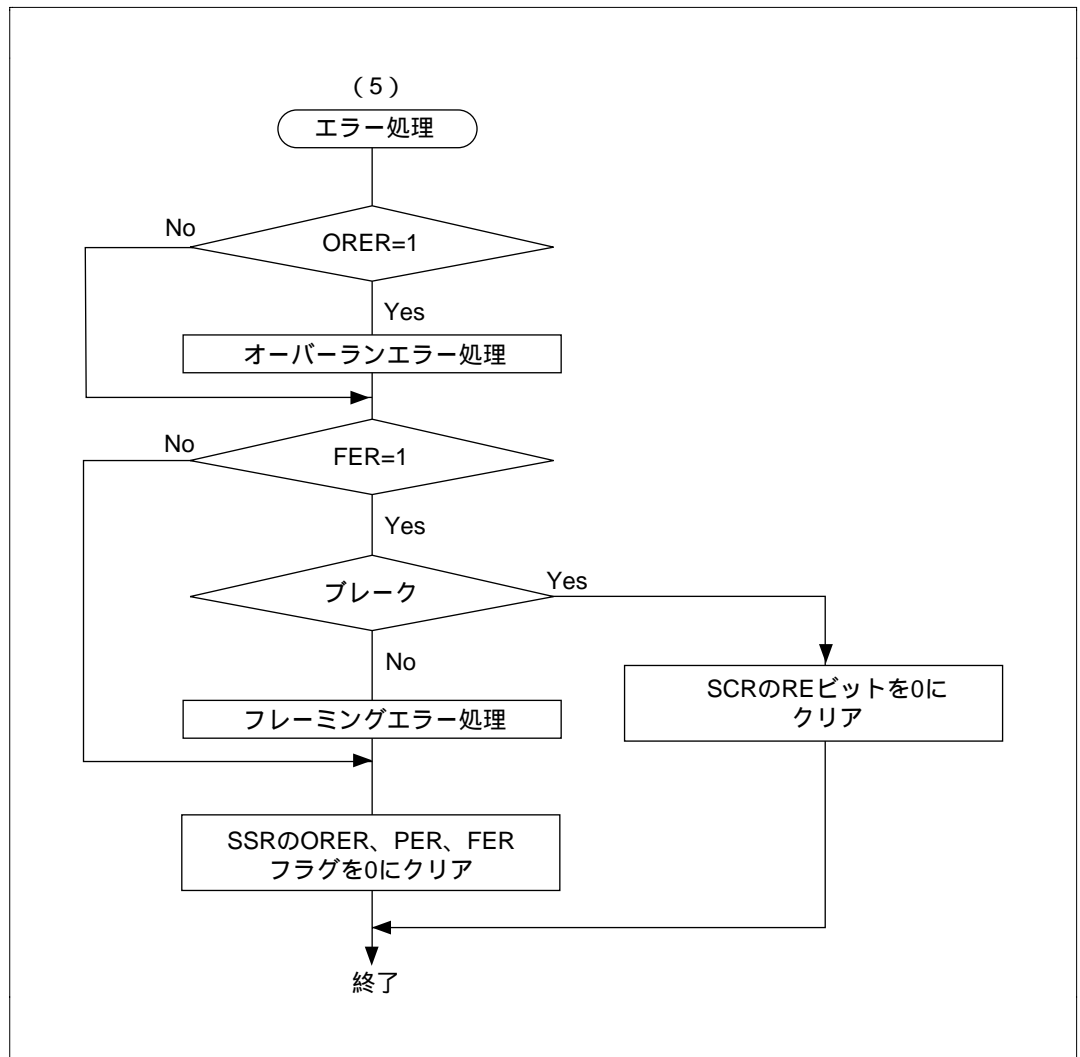


図 13.12 マルチプロセッサシリアル受信のフローチャートの例 (2)

図 13.13 にマルチプロセッサフォーマットの SCI の受信時の動作例を示します。

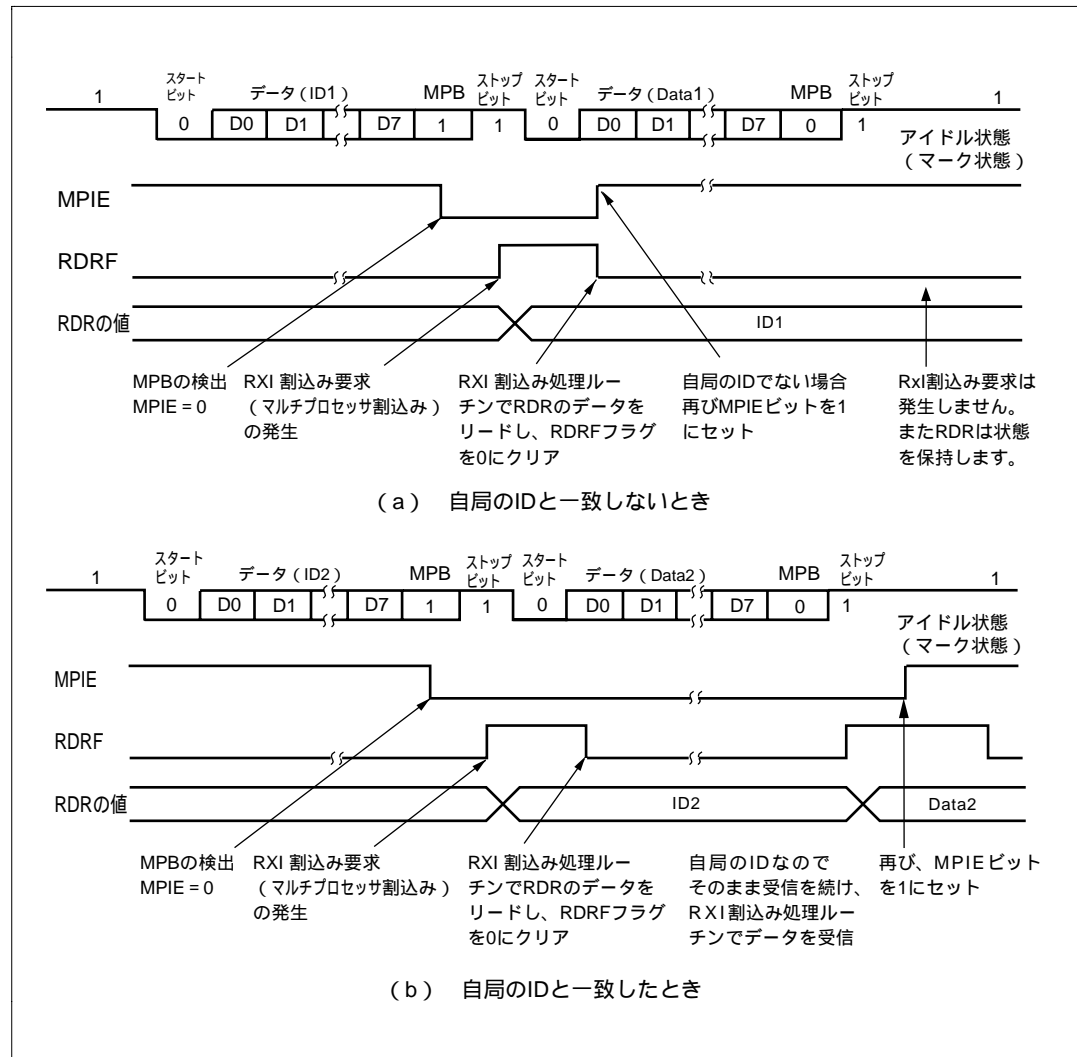


図 13.13 SCI の受信時の動作例

(8ビットデータ/マルチプロセッサビットあり/1ストップビットの例)

### 13.3.4 クロック同期式モード時の動作

クロック同期式モードは、クロックパルスに同期してデータを送信/受信するモードで、高速シリアル通信に適しています。

SCI 内部では、送信部と受信部は独立していますので、クロックを共有することで全二重通信ができます。

また、送信部と受信部が共にダブルバッファ構造になっていますので送信/受信中にデータのリード/ライトができ、連続送信/受信が可能です。

クロック同期式シリアル通信の一般的なフォーマットを図 13.14 に示します。

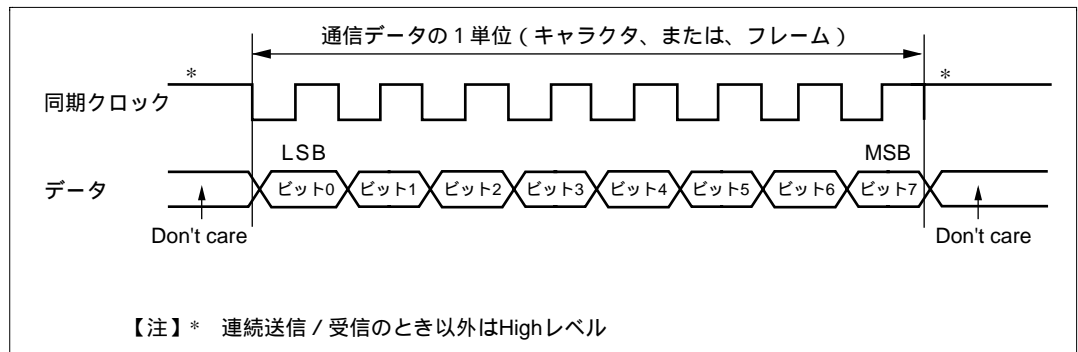


図 13.14 クロック同期式通信データフォーマット

クロック同期式シリアル通信では、通信回線のデータは同期クロックの立下がりから次の立下がりまで出力されます。また、同期クロックの立上がりでデータの確定が保証されます。

シリアル通信の1キャラクタは、データのLSBから始まり最後にMSBが出力されます。MSB出力後の通信回線の状態はMSBの状態を保ちます。

クロック同期式モードでは、SCIは同期クロックの立上がりに同期してデータを受信します。

### (1) 送信 / 受信フォーマット

8ビットデータ固定です。

パリティビットやマルチプロセッサビットの付加はできません。

### (2) クロック

SMRの $C/\bar{A}$ ビットとSCRのCKE1、CKE0ビットの設定により内蔵ポーレートジェネレータの生成した内部クロック、または、SCK端子から入力された外部同期クロックの2種類から選択できます。SCIのクロックソースの選択については表13.6を参照してください。

内部クロックで動作させるとき、SCK端子からは同期クロックが出力されます。

同期クロックは1キャラクタの送受信で8パルス出力され、送信 / 受信を行わないときにはHighレベルに固定されます。1キャラクタ単位の受信動作を行いたいときは、クロックソースは外部クロックを選択してください。

### (3) データの送信 / 受信動作

#### (a) SCIのイニシャライズ(クロック同期式)

データの送信 / 受信前には、SCRのTE、REビットを0にクリアした後、以下の手順に従いSCIをイニシャライズしてください。

モードの変更は、通信フォーマットの変更などの場合には必ず、TE、REビットを0にクリアしてから下記手順で変更してください。TEビットを0にクリアするとTDREフラグは1にセットされ、TSRがイニシャライズされます。

REビットを0にクリアしてもRDRF、PER、OREの各フラグ、およびRDRの内容は保持されますので注意してください。

図13.15にSCIのイニシャライズフローチャートの例を示します。

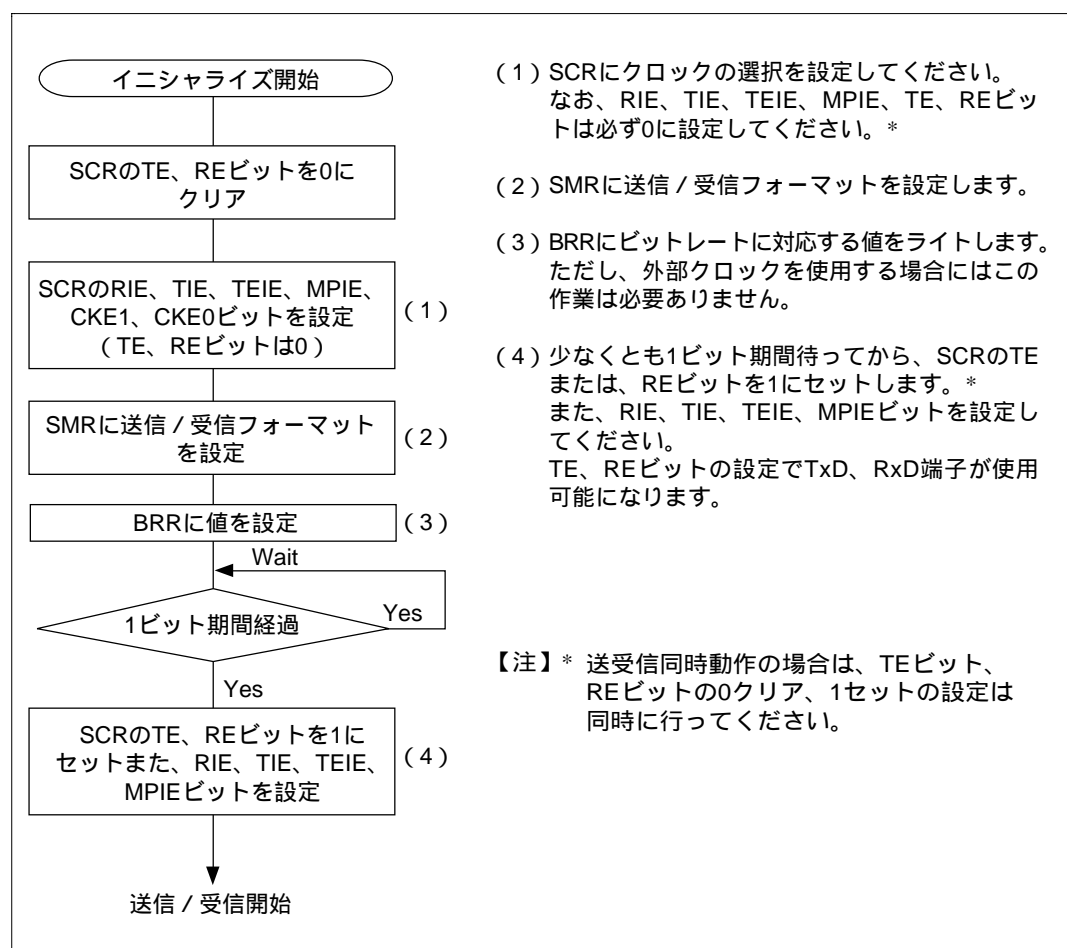


図13.15 SCIのイニシャライズフローチャートの例



## (b) シリアルデータ送信（クロック同期式）

図 13.16 にシリアル送信のフローチャートの例を示します。

シリアルデータ送信は以下の手順に従って行ってください。

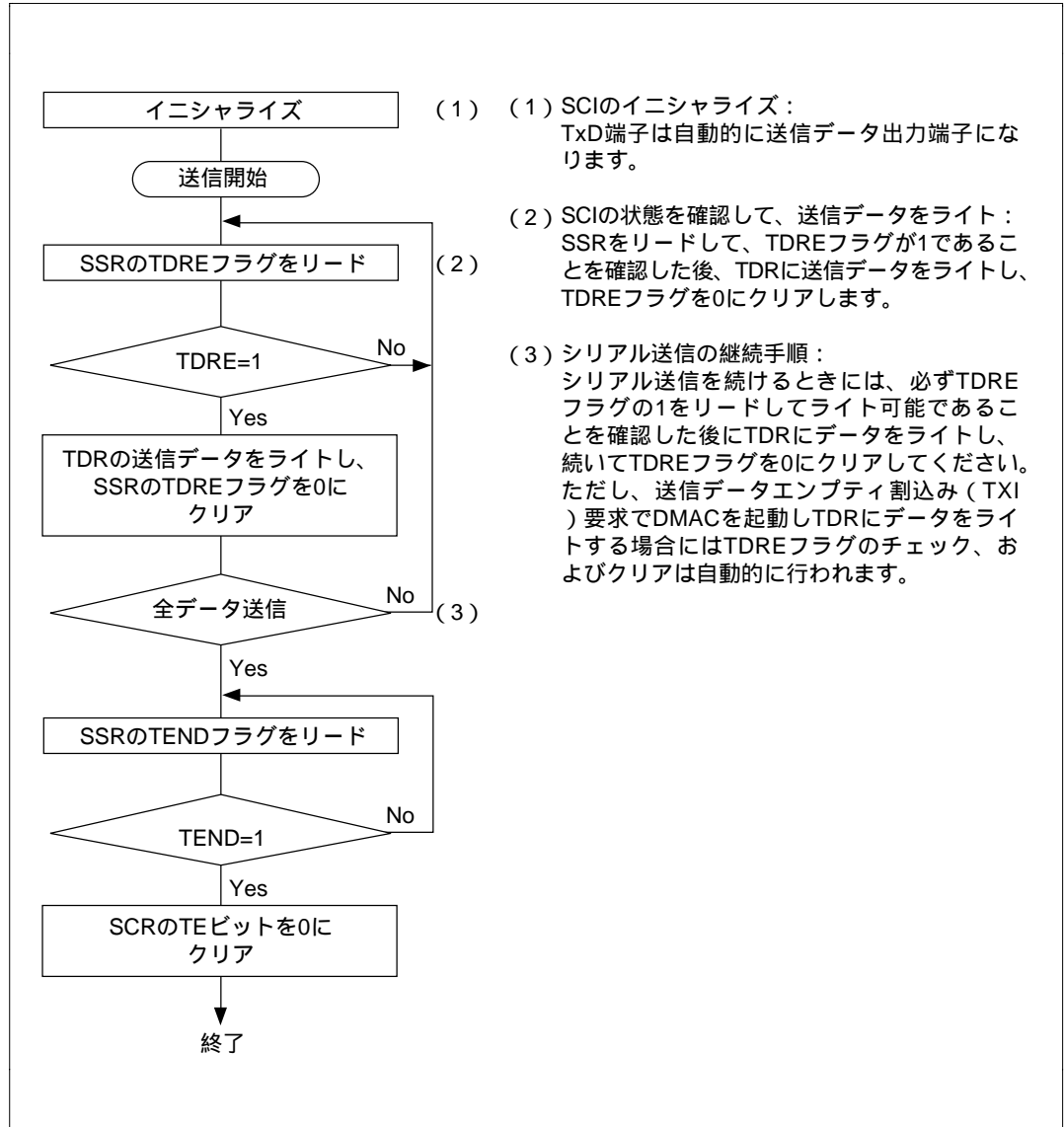


図 13.16 シリアル送信のフローチャートの例

SCIはシリアル送信時に以下のように動作します。

- (1) SCIはSSRのTDREフラグを監視し、0であるとTDRにデータがライトされたと認識し、TDRからTSRデータを転送します。
- (2) TDRからTSRへデータを転送した後にTDREフラグを1にセットし、送信を開始します。

このとき、SCR の TIE ビットが 1 にセットされていると送信データエンプティ割り込み (TXI) 要求を発生します。

クロック出力モードに設定したときには、SCI は同期クロックを 8 パルス出力します。外部クロックに設定したときには、入力クロックに同期してデータを出力します。シリアル送信データは、LSB (ビット 0) ~ MSB (ビット 7) の順に TxD 端子から送り出されます。

(3) SCI は、MSB (ビット 7) を送り出すタイミングで TDRE フラグをチェックします。TDRE フラグが 0 であると TDR から TSR にデータを転送し、次フレームのシリアル送信を開始します。

TDRE フラグが 1 であると SSR の TEND フラグを 1 にセットし、MSB (ビット 7) を送り出した後、TxD 端子は状態を保持します。

このとき SCR の TEIE ビットが 1 にセットされていると送信終了割り込み (TEI) 要求を発生します。

(4) シリアル送信終了後は、SCK 端子は固定になります。

図 13.17 に SCI の送信時の動作例を示します。

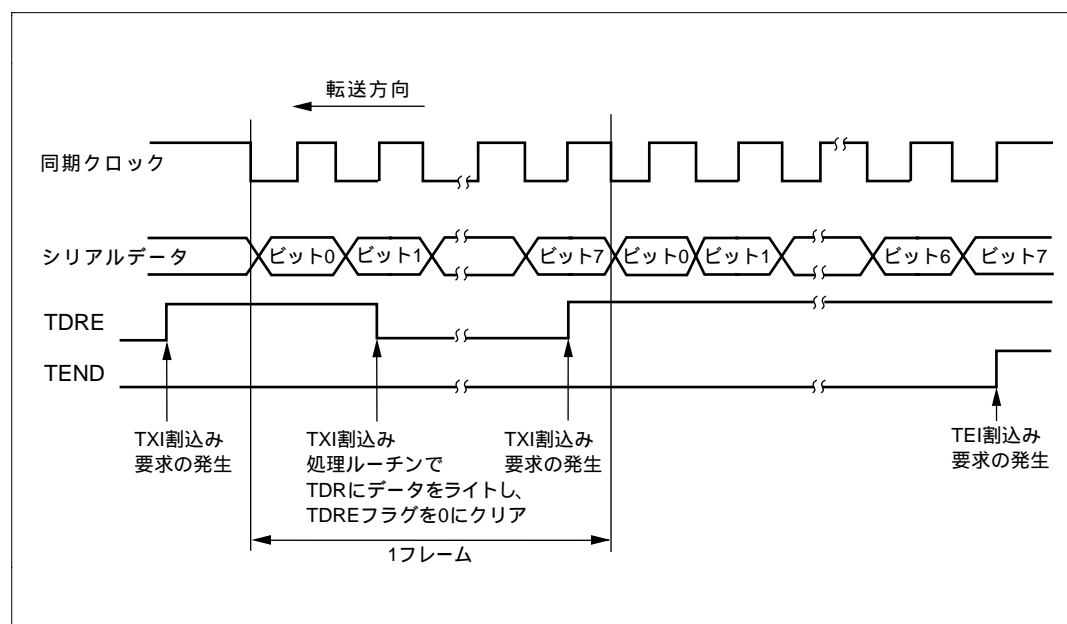


図 13.17 SCI の送信時の動作例

## (c) シリアルデータ受信 (クロック同期式)

図 13.18 にシリアル受信フローチャートの例を示します。

シリアルデータ受信は以下の手順に従って行ってください。

動作モードを調歩同期式モードからクロック同期式モードに切り換える際には、必ず ORER、PER、FER の各フラグが 0 にクリアされていることを確認してください。

FER、PER フラグが 1 にセットされていると RDRF フラグがセットされません。また、送信 / 受信動作が行えません。

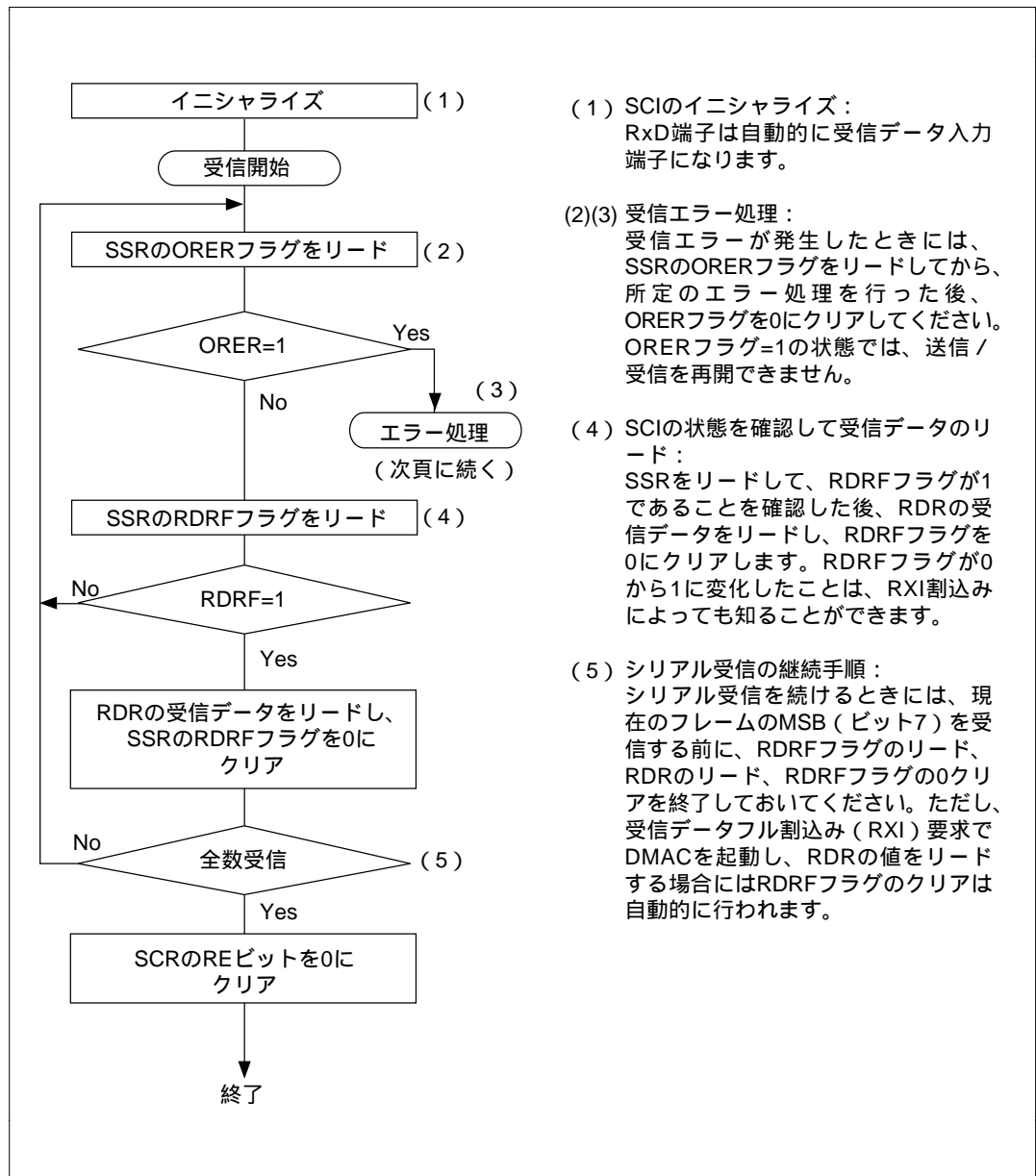


図 13.18 シリアルデータ受信フローチャートの例 (1)

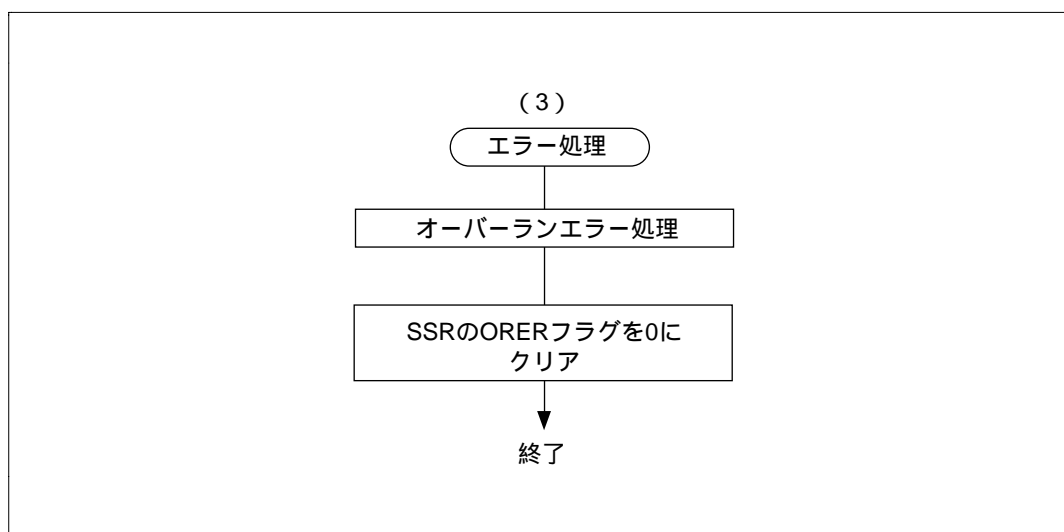


図 13.18 シリアルデータ受信フローチャートの例 (2)

SCI は受信時に以下のように動作します。

(1) SCI は同期クロックの入力または出力に同期して内部を初期化します。

(2) 受信したデータを RSR の LSB から MSB の順に格納します。

受信後、SCI は、RDRF フラグが 0 であり、受信データを RSR から RDR に転送できる状態であるかをチェックします。

このチェックがパスしたとき RDRF フラグが 1 にセットされ、RDR に受信データが格納されます。エラーチェックで受信エラーが発生すると、表 13.11 のように動作しません。

エラーチェックで受信エラーが発生した状態では以後の送信、受信動作ができません。

(3) RDRF フラグが 1 になったとき、SCR の RIE ビットが 1 にセットされていると受信データフル割込み (RXI) 要求が発生します。

また、ORER フラグが 1 になったとき、SCR の RIE ビットが 1 にセットされていると受信エラー割込み (ERI) 要求が発生します。

図 13.19 に SCI の受信時の動作例を示します。

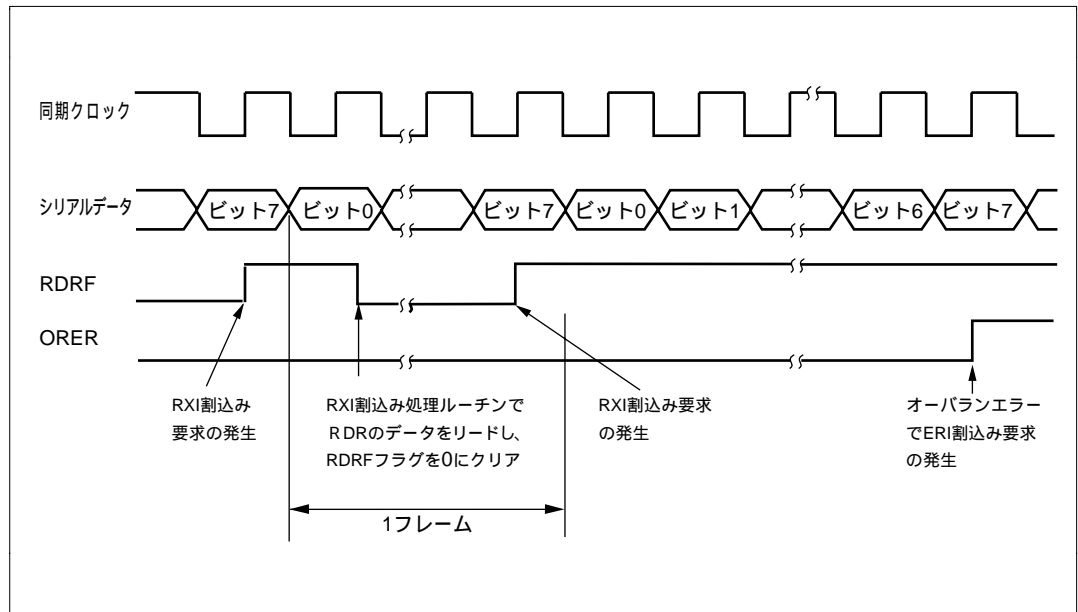


図 13.19 SCI の受信時の動作例

## (d) シリアルデータ送受信同時動作 (クロック同期式)

図 13.20 にシリアル送受信同時動作のフローチャートの例を示します。

シリアルデータ送受信同時動作は、以下の手順に従って行ってください。

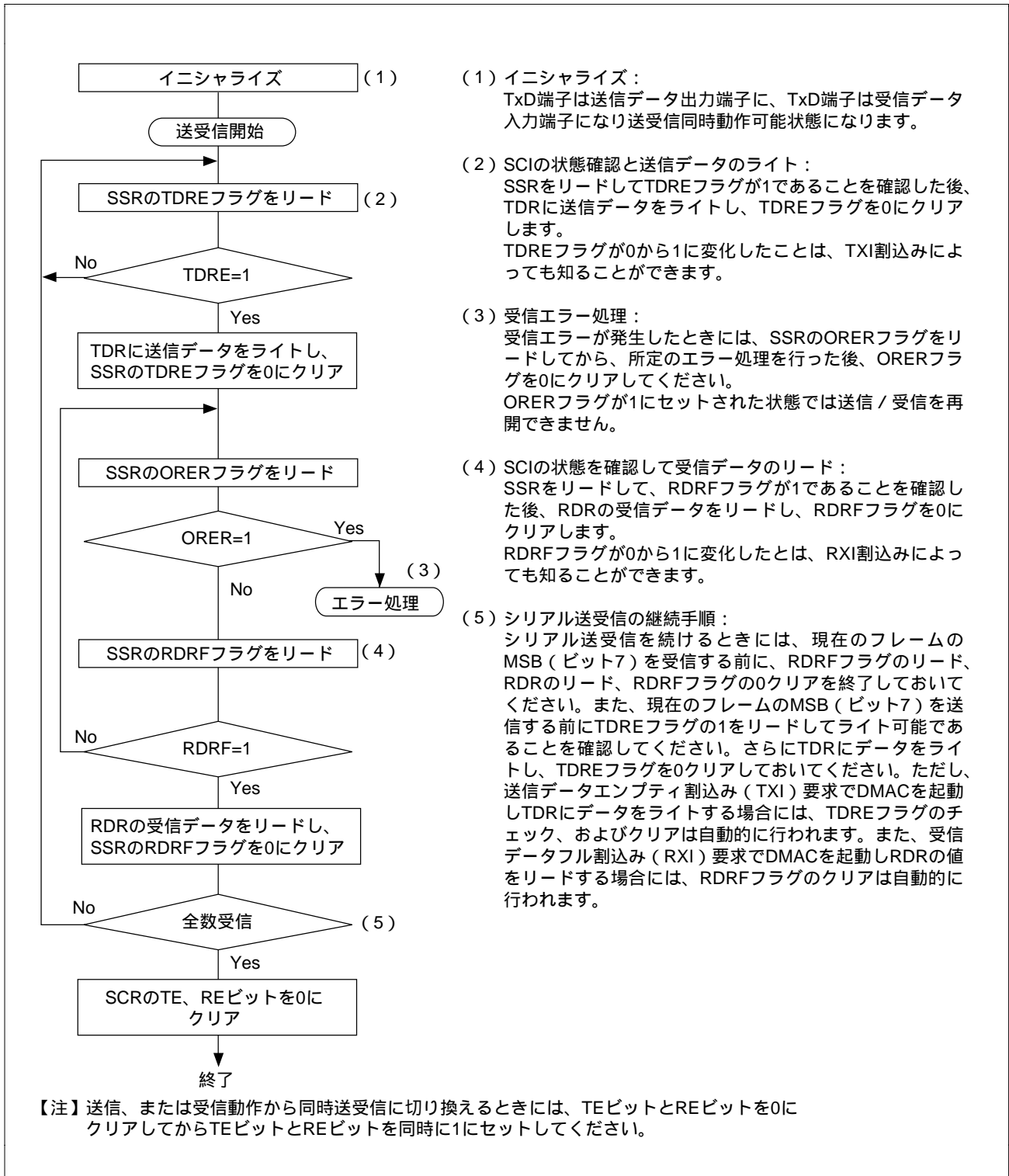


図 13.20 シリアル送受信同時動作のフローチャートの例

## 13.4 SCI 割込み

SCIには、送信終了割込み (TEI) 要求、受信エラー割込み (ERI) 要求、受信データフル割込み (RXI) 要求、送信データエンプティ割込み (TXI) 要求の4種類の割込み要因があります。表 13.12 に各割込み要因と優先順位を示します。各割込み要因は、SCRのTIEビット、RIEビットおよびTEIEビットで許可/禁止できます。また、各割込み要求はそれぞれ独立に割込みコントローラに送られます。

SSRのTDREフラグが1にセットされると、TXI割込み要求が発生します。また、SSRのTENDフラグが1にセットされると、TEI割込み要求が発生します。TXI割込み要求によりDMACを起動してデータ転送を行うことができます。TDREフラグはDMACによるデータ転送時に自動的に0にクリアされます。なお、TEI割込み要求でDMACの起動はできません。

SSRのRDRFフラグが1にセットされるとRXI割込み要求が発生します。SSRのORER、PER、FERフラグのいずれかが1にセットされるとERI割込み要求が発生します。RXI割込み要求でDMACを起動してデータ転送を行うことができます。RDRFフラグはDMACによるデータ転送時に自動的に0にクリアされます。なお、ERI割込み要求でDMACの起動はできません。

SCIチャンネル0の割込みによりDMACの起動が可能です。

表 13.12 SCI 割込み要因

割込み要因	内容	優先順位
ERI	受信エラー (ORER、FER、PER) による割込み	高 ↑ 低
RXI	受信データフル (RDRF) による割込み	
TXI	送信データエンプティ (TDRE) による割込み	
TEI	送信終了 (TEND) による割込み	

## 13.5 使用上の注意

### 13.5.1 SCI を使用する際の注意

SCI を使用する際は、以下のことに注意してください。

#### (1) TDR へのライトと TDRE フラグの関係について

SSR の TDRE フラグは TDR から TSR に送信データの転送が行われたことを示すステータスフラグです。SCI が TDR から TSR にデータを転送すると、TDRE フラグが 1 にセットされます。

TDR へのデータのライトは、TDRE フラグの状態にかかわらず行うことができます。しかし、TDRE フラグが 0 の状態で新しいデータを TDR にライトすると、TDR に格納されていたデータは、まだ TSR に転送されていないため失われてしまいます。したがって TDR への送信データのライトは、必ず TDRE フラグが 1 にセットされていることを確認してから行ってください。

#### (2) 複数の受信エラーが同時に発生した場合の動作について

複数の受信エラーが同時に発生した場合、SSR の各ステータスフラグの状態は、表 13.13 のようになります。また、オーバランエラーが発生した場合には RSR から RDR へのデータ転送は行われず、受信データは失われます。

表 13.13 SSR のステータスフラグの状態と受信データの転送

SSR のステータスフラグ				受信データ転送		受信エラーの状態
RDRF	ORER	FER	PER	RSR	RDR	
1	1	0	0	x		オーバランエラー
0	0	1	0			フレーミングエラー
0	0	0	1			パリティエラー
1	1	1	0	x		オーバランエラー + フレーミングエラー
1	1	0	1	x		オーバランエラー + パリティエラー
0	0	1	1			フレーミングエラー + パリティエラー
1	1	1	1	x		オーバランエラー + フレーミングエラー + パリティエラー

【注】 : RSR RDR に受信データを転送します。

x : RSR RDR に受信データを転送しません。



### (3) ブレークの検出と処理について

フレーミングエラー (FER) 検出時に RxD 端子の値を直接リードすることで、ブレークを検出できます。ブレークでは、RxD 端子からの入力がすべて 0 になりますので FER フラグがセットされ、またパリティエラー (PER) もセットされる場合があります。

SCI は、ブレークを受信した後も受信動作を続けますので、FER フラグを 0 にクリアしても再び 1 にセットされますので、注意してください。

### (4) ブレークの送り出し

TxD 端子は、DR と DDR により入出力方向とレベルが決まる I/O ポートと兼用になっています。これを利用してブレークの送り出しができます。

シリアル送信のイニシャライズから TE ビットを 1 にセットするまでは、マーク状態を DR の値で代替します (TE ビットを 1 にセットするまで、TxD 端子として機能しません)。このため、最初は DDR と DR を 1 に設定しておきます。

シリアル送信時にブレークを送り出したいときは DR を 0 にクリアした後、TE ビットを 0 にクリアします。

TE ビットを 0 にクリアすると現在の送信状態とは無関係に送信部は初期化され、TxD 端子は I/O ポートになり、TxD 端子から 0 が出力されます。

### (5) 受信エラーフラグと送信動作について (クロック同期式モードのみ)

受信エラーフラグ (ORER、PER、FER) が 1 にセットされた状態では、TDRE フラグを 0 にクリアしても送信を開始できません。必ず送信開始時には、受信エラーフラグを 0 にクリアしておいてください。

また、RE ビットを 0 にクリアしても受信エラーフラグは 0 にクリアできませんので注意してください。

### (6) 調歩同期式モードの受信データサンプリングタイミングと受信マージン

調歩同期式モードでは、SCI は転送レートの 16 倍の周波数の基本クロックで動作しています。

受信時に SCI は、スタートビットの立下りを基本クロックでサンプリングして、内部を同期化します。また、受信データを基本クロックの 8 クロック目の立上りエッジで内部に取り込みます。これを図 13.21 に示します。

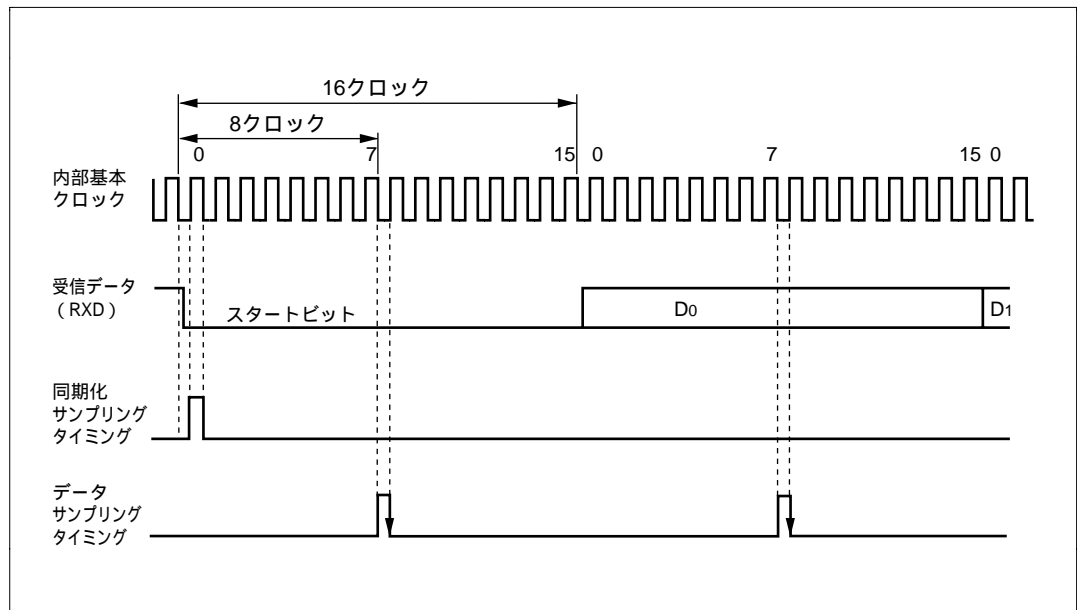


図 13.21 調歩同期式モードの受信データサンプリングタイミング

したがって、調歩同期式モードでの受信マージンは式(1)のように表すことができます。

$$M = \left| \left( 0.5 - \frac{1}{2N} - (L - 0.5)F - \frac{|D - 0.5|}{N} (1 + F) \right) \right| \times 100\% \quad \dots\dots\dots \text{式(1)}$$

- M : 受信マージン (%)
- N : クロックに対するビットレートの比 (N=16)
- D : クロックデューティ (D=0~1.0)
- L : フレーム長 (L=9~12)
- F : クロック周波数の偏差の絶対値

式(1)で、F=0、D=0.5 とすると、受信マージンは式(2)より 46.875% となります。

D = 0.5、F = 0 のとき

$$M = \left( 0.5 - \frac{1}{2 \times 16} \right) \times 100\% = 46.875\% \quad \dots\dots\dots \text{式(2)}$$

ただし、この値はあくまでも計算上の値ですので、システム設計の際には 20~30% の余裕を持たせてください。

## (7) DMAC 使用上の制約事項

- (1) 同期クロックに外部クロックソースを使用する場合、DMAC による TDR の更新後、クロックで5クロック以上経過した後に、送信クロックを入力してください。TDR の更新後4ステート以内に送信クロックを入力すると、誤動作することがあります。

(図 13.22)

- (2) DMAC により、RDR のリードを行うときは必ず DTCR の DTS2~0 ビットで起動要因を当該 SCI の受信完了割込み (RXI) に設定してください。

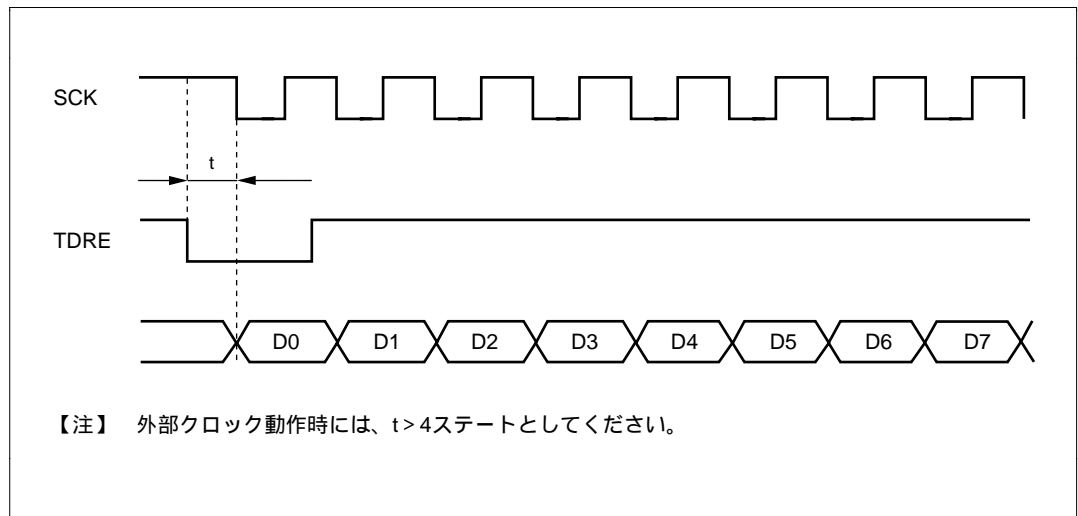


図 13.22 DMAC によるクロック同期式送信時の例

## (8) SCK 端子からポート端子へ切り換えるときの注意事項

## (1) 動作現象

DDR = 1、DR = 1、 $C/\bar{A}$  = 1、CKE1 = 0、CKE0 = 0、TE = 1 の状態（クロック同期式モード）において、以下の設定で SCK 端子機能を出力ポート機能（High 出力）に切り換える際、半サイクルの Low 出力が発生します。

1. シリアルデータ送信終了
2. TE ビット = 0
3.  $C/\bar{A}$  ビット = 0 ... ポート出力に切り換え
4. Low 出力発生（図 13.23 参照）

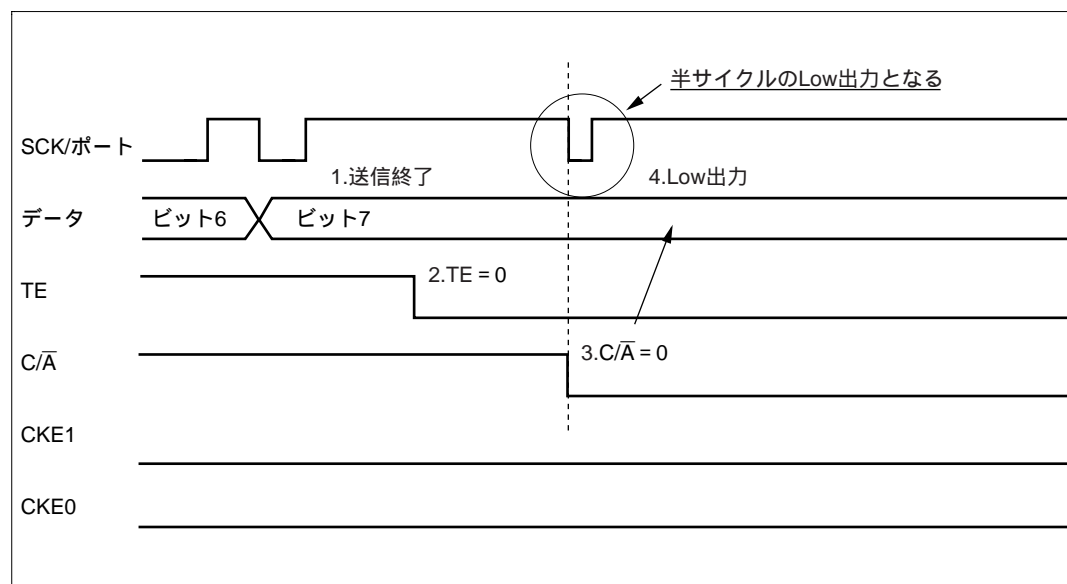


図 13.23 SCK 端子からポート端子へ切り換えるときの動作

## (2) Low 出力を回避する手順例

本手順例は、SCK 端子を一度入力状態にするため、あらかじめ SCK/ポート端子を外部回路で Pull-up してください。

DDR = 1、DR = 1、 $C/\bar{A} = 1$ 、CKE1 = 0、CKE0 = 0、TE = 1 の状態より以下の 1~5 の手順で設定してください。

1. シリアルデータ送信終了
2. TE ビット = 0
3. CKE1 ビット = 1
4.  $C/\bar{A}$  ビット = 0 ... ポート出力に切り換え
5. CKE1 ビット = 0

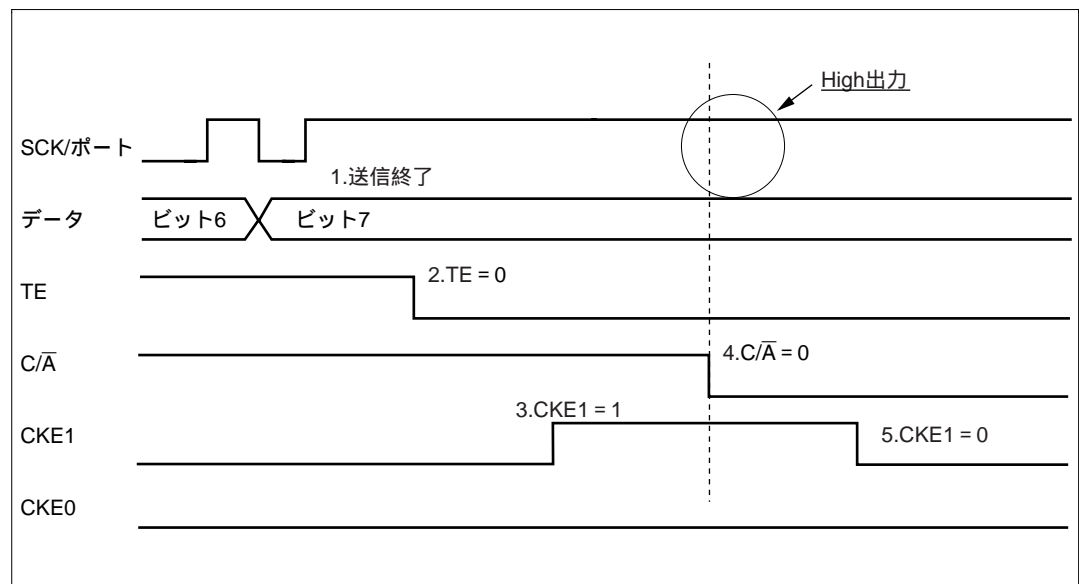


図 13.24 SCK 端子からポート端子へ切り換えるときの動作 (Low 出力の回避例)

---

# 14. スマートカード インタフェース

---

## 第14章 目次

14.1 概要	581
14.1.1 特長	581
14.1.2 ブロック図	582
14.1.3 端子構成	582
14.1.4 レジスタ構成	583
14.2 各レジスタの説明	584
14.2.1 スマートカードモードレジスタ (SCMR)	584
14.2.2 シリアルステータスレジスタ (SSR)	586
14.2.3 シリアルモードレジスタ (SMR)	587
14.2.4 シリアルコントロールレジスタ (SCR)	589
14.3 動作説明	590
14.3.1 概要	590
14.3.2 端子接続	590
14.3.3 データフォーマット	591
14.3.4 レジスタの設定	593
14.3.5 クロック	595
14.3.6 データの送信 / 受信動作	597
14.4 使用上の注意	606



## 14.1 概要

SCI は、シリアルコミュニケーションインタフェースの拡張機能として、ISO/IEC 7816-3 (Identification Card) に準拠した IC カード (スマートカード) インタフェースをサポートしています。

通常のシリアルコミュニケーションインタフェースとスマートカードインタフェースの切り替えはレジスタの設定で行います。

### 14.1.1 特長

本 LSI がサポートするスマートカードインタフェースには次の特長があります。

調歩同期式モード

- ・ データ長 : 8 ビット
- ・ パリティビットの生成およびチェック
- ・ 受信モードにおけるエラーシグナル (パリティエラー) の送出
- ・ 送信モードにおけるエラーシグナルの検出とデータの自動再送信
- ・ ダイレクトコンベンション / インバースコンベンションの両方をサポート

内蔵ボーレートジェネレータにより任意のビットレートを選択可能

3 種類の割込み要因

- ・ 送信データエンプティ、受信データフル、送受信エラーの 3 種類の割込み要因があり、それぞれ独立に要求可能
- ・ 送信データエンプティ割込みと受信データフル割込みにより、DMA コントローラ (DMAC) を起動させてデータを転送可能



### 14.1.2 ブロック図

図 14.1 にスマートカードインタフェースのブロック図を示します。

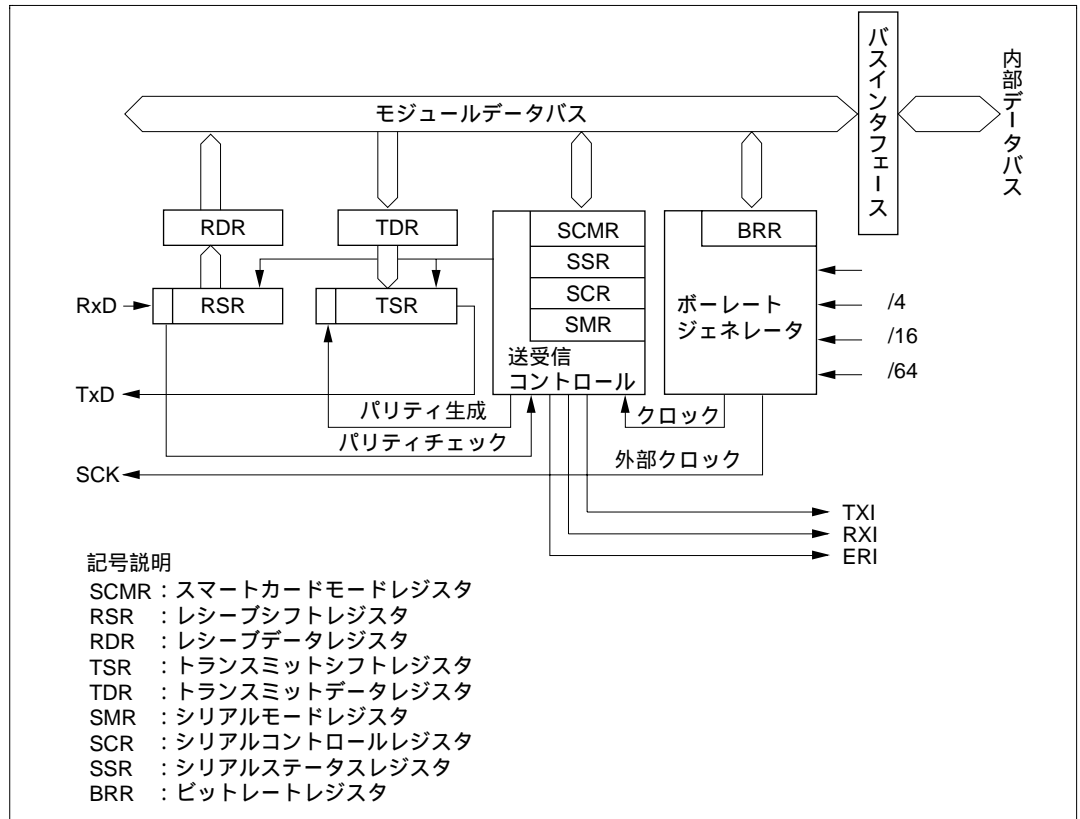


図 14.1 スマートカードインタフェースのブロック図

### 14.1.3 端子構成

スマートカードインタフェースの端子構成を表 14.1 に示します。

表 14.1 端子構成

名 称	記号	入出力	機 能
シリアルクロック端子	SCK	入出力	クロック入出力
レシーブデータ端子	RxD	入力	受信データ入力
トランスミットデータ端子	TxD	出力	送信データ出力

### 14.1.4 レジスタ構成

スマートカードインタフェースで使用するレジスタ構成を表 14.2 に示します。BRR、TDR、RDR については、通常の SCI の機能と同様ですので、「第 13 章 シリアルコミュニケーションインタフェース」のレジスタの説明を参照してください。

表 14.2 レジスタ構成

チャンネル	アドレス* <sup>1</sup>	名 称	略称	R/W	初期値
0	H'FFFB0	シリアルモードレジスタ	SMR	R/W	H'00
	H'FFFB1	ビットレートレジスタ	BRR	R/W	H'FF
	H'FFFB2	シリアルコントロールレジスタ	SCR	R/W	H'00
	H'FFFB3	トランスミットデータレジスタ	TDR	R/W	H'FF
	H'FFFB4	シリアルステータスレジスタ	SSR	R/(W)* <sup>2</sup>	H'84
	H'FFFB5	レシーブデータレジスタ	RDR	R	H'00
	H'FFFB6	スマートカードモードレジスタ	SCMR	R/W	H'F2
1	H'FFFB8	シリアルモードレジスタ	SMR	R/W	H'00
	H'FFFB9	ビットレートレジスタ	BRR	R/W	H'FF
	H'FFFB A	シリアルコントロールレジスタ	SCR	R/W	H'00
	H'FFFB B	トランスミットデータレジスタ	TDR	R/W	H'FF
	H'FFFB C	シリアルステータスレジスタ	SSR	R/(W)* <sup>2</sup>	H'84
	H'FFFB D	レシーブデータレジスタ	RDR	R	H'00
	H'FFFB E	スマートカードモードレジスタ	SCMR	R/W	H'F2
2	H'FFFC0	シリアルモードレジスタ	SMR	R/W	H'00
	H'FFFC1	ビットレートレジスタ	BRR	R/W	H'FF
	H'FFFC2	シリアルコントロールレジスタ	SCR	R/W	H'00
	H'FFFC3	トランスミットデータレジスタ	TDR	R/W	H'FF
	H'FFFC4	シリアルステータスレジスタ	SSR	R/(W)* <sup>2</sup>	H'84
	H'FFFC5	レシーブデータレジスタ	RDR	R	H'00
	H'FFFC6	スマートカードモードレジスタ	SCMR	R/W	H'F2

【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。

\*2 ビット 7~3 はフラグをクリアするための 0 ライトのみ可能です。

## 14.2 各レジスタの説明

スマートカードインタフェースで追加されるレジスタ、および機能が変更されるビットについて説明します。

### 14.2.1 スマートカードモードレジスタ (SCMR)

SCMR は、8ビットのリード/ライト可能なレジスタで、スマートカードインタフェースの機能の選択を行います。

ビット :	7	6	5	4	3	2	1	0
	-	-	-	-	SDIR	SINV	-	SMIF
初期値 :	1	1	1	1	0	0	1	0
R/W :	-	-	-	-	R/W	R/W	-	R/W

リザーブビット

リザーブビット

スマートカードインタフェース  
モードセレクト  
スマートカードインタフェース  
機能を許可/禁止するビット  
です。

スマートカードデータインポート  
データのロジックレベルの反転を  
指定するビットです。

スマートカードデータトランスファディレクション  
シリアル/パラレル変換のフォーマットを選択す  
るビットです。

SCMR は、リセットまたはスタンバイモード時に、HF2 にイニシャライズされます。

#### ビット7~4 : リザーブビット

リードすると常に1が読み出されます。ライトは無効です。

#### ビット3 : スマートカードデータトランスファディレクション (SDIR)

シリアル/パラレル変換のフォーマットを選択します。\*1

ビット3	説明
SDIR	
0	TDRの内容をLSBファーストで送信 受信データをLSBファーストとしてRDRに格納 (初期値)
1	TDRの内容をMSBファーストで送信 受信データをMSBファーストとしてRDRに格納

## ビット2：スマートカードデータインバート（SINV）

データのロジックレベルの反転を指定します。この機能は、SDIR ビットと組み合わせインバースコンベンションカードとの送受信に使用します。<sup>\*2</sup> SINV ビットは、パリティビットのロジックレベルには影響しません。パリティに関する設定方法については、「14.3.4 レジスタの設定」を参照してください。

ビット2	説明
SINV	
0	TDRの内容をそのまま送信 受信データをそのままRDRに格納 (初期値)
1	TDRの内容を反転してデータを送信 受信データを反転してRDRに格納

## ビット1：リザーブビット

リードすると常に1が読み出されます。ライトは無効です。

## ビット0：スマートカードインタフェースモードセレクト（SMIF）

スマートカードインタフェース機能をイネーブルにするビットです。

ビット0	説明
SMIF	
0	スマートカードインタフェース機能を禁止 (初期値)
1	スマートカードインタフェース機能を許可

【注】 \*1 LSB ファースト / MSB ファーストの切り換え機能は、通常のシリアルコミュニケーションインタフェースにおいても使用することができます。

通信フォーマットのデータ長を7ビットとして送受信するシリアルデータをMSBファーストとする場合には、TDRのビット0は送信されません。また、受信するデータは7ビット目から1ビット目が有効になりますので注意してください。

\*2 データのロジックレベルの反転機能は、通常のシリアルコミュニケーションインタフェースにおいても使用することができます。

送受信するシリアルデータを反転させる場合には、パリティの送信とパリティのチェックは、レジスタの値ではなくシリアルデータ入出力端子のハイレベルの数に対して行われるので、注意してください。

### 14.2.2 シリアルステータスレジスタ (SSR)

スマートカードインタフェースモードにおいては、SSRのビット4の機能が変更されま  
す。また、これに関連してビット2のTENDのセット条件が変更になります。

ビット:	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	ERS	PER	TEND	MPB	MPBT
初期値:	1	0	0	0	0	1	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/W

トランスミットエンド  
送信終了を示すフラグです。

エラーシグナルステータス (ERS)  
エラーシグナルが送出されたことを  
示すフラグです。

【注】\* フラグをクリアするための0ライトのみ可能です。

ビット7~5:

通常のSCIと同様の動作をします。詳細は「13.2.7 シリアルステータスレジスタ (SSR)」を参照してください。

ビット4: エラーシグナルステータス (ERS)

スマートカードインタフェースモードでは、送信時に受信側から送り返されるエラーシ  
グナルのステータスを示します。なお、スマートカードインタフェースではフレーミング  
エラーは検出しません。

ビット4	説明
ERS	
0	正常に受信され、エラーシグナルが無いことを表示 [クリア条件] (初期値) (1) リセット、スタンバイモード、またはモジュールストップモード時 (2) ERS = 1の状態をリードした後、0をライトしたとき
1	受信側からパリティエラーの検出を示すエラーシグナルが検出されたことを表 示 [セット条件] エラーシグナルLowをサンプリングしたとき

【注】 SCRのTEビットを0にクリアしても、ERSフラグは影響を受けず以前の状態を保持しま  
す。

ビット3~0:

通常の SCI と同様の動作をします。詳細は「13.2.7 シリアルステータスレジスタ (SSR)」を参照してください。

ただし、TEND ビットのセット条件は次のようになります。

ビット2	説明
TEND	
0	送信中であることを表示 [クリア条件] (1) TDRE=1の状態をリードした後、TDREフラグに0をライトしたとき (2) DMACでTDRヘデータをライトしたとき
1	送信を終了したことを表示 [セット条件] (初期値) (1) リセットまたはスタンバイモード時 (2) SCRのTEビットが0かつFER/ERSビットが0のとき (3) 1バイトのシリアルキャラクタを送信して、2.5etu後にTDRE=1かつERS=0(正常送信)のとき

【注】 etu (Elementary Time Unit : 1ビットの転送期間の略)

### 14.2.3 シリアルモードレジスタ (SMR)

スマートカードインタフェースモードにおいては、SMRのビット7の機能が変更されます。また、これに関連してシリアルコントロールレジスタ (SCR)のビット1およびビット0の機能も変更されます。

ビット:	7	6	5	4	3	2	1	0
	GM	CHR	PE	O/ $\bar{E}$	STOP	MP	CKS1	CKS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

## ビット7：GSMモード（GM）

通常のスマートカードインタフェース時は0に設定します。GSMモードは、本ビットを1に設定し、送信完了を示すTENDフラグのセットタイミングの前倒しと、クロック出力の制御モードの追加を行います。クロック出力の制御モードの追加内容は、シリアルコントロールレジスタ（SCR）のビット1およびビット0で指定されます。

ビット7	説明
GM	
0	通常のスマートカードインタフェースモードの動作 (1) TENDフラグは開始ビットの先頭から12.5etuのタイミングで発生 (2) クロック出力のON/OFF制御のみ (初期値)
1	GSMモードのスマートカードインタフェースモードの動作 (1) TENDフラグは開始ビットの先頭から11.0etuのタイミングで発生 (2) クロック出力のON/OFF、およびHigh/Low固定制御

## ビット6～0：

通常のSCIと同様の動作をします。詳細は「13.2.5 シリアルモードレジスタ（SMR）」を参照してください。

### 14.2.4 シリアルコントロールレジスタ (SCR)

スマートカードインタフェースモードにおいては、SCR のビット 1、0 の機能が変更されます。

ビット :	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
初期値 :	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ビット 7~2 :

通常の SCI と同様の動作をします。詳細は「13.2.6 シリアルコントロールレジスタ (SCR)」を参照してください。

ビット 1、0 : クロックイネーブル 1、0 (CKE1、CKE0)

SCI のクロックソースの選択、および SCK 端子からのクロック出力の許可 / 禁止を設定します。

スマートカードインタフェースモードにおいては、通常のクロック出力の許可 / 禁止の切り替え、およびクロック出力の High レベル固定と Low レベル固定を指定することができます。

ビット 7	ビット 1	ビット 0	説明
GM	CKE1	CKE0	
0	0	0	内部クロック / SCK 端子は入出力ポート (初期値)
		1	内部クロック / SCK 端子はクロック出力
1	0	0	内部クロック / SCK 端子は Low 出力固定
		1	内部クロック / SCK 端子はクロック出力
	1	0	内部クロック / SCK 端子は High 出力固定
		1	内部クロック / SCK 端子はクロック出力



## 14.3 動作説明

### 14.3.1 概要

スマートカードインタフェースの主な機能は次のとおりです。

- (1) 1フレームは、8ビットデータとパリティビットで構成されます。
- (2) 送信時は、パリティビットの終了から次のフレーム開始まで 2etu (Elementary Time Unit : 1ビットの転送期間) 以上のガードタイムをおきます。
- (3) 受信時はパリティエラーを検出した場合、スタートビットから 10.5etu 経過後、エラーシグナルLowを1 etu 期間出力します。
- (4) 送信時はエラーシグナルをサンプリングすると、2etu 以上経過後、自動的に同じデータを送信します。
- (5) 調歩同期式非同期通信機能のみサポートし、クロック同期式通信機能はありません。

### 14.3.2 端子接続

図 14.2 にスマートカードインタフェースに関する端子接続概略図を示します。

IC カードとの通信においては、1 本のデータ伝送線で送信と受信が行われるので、LSI 端子で TxD 端子と RxD 端子とを結線してください。また、データ伝送線は、抵抗で電源  $V_{CC}$  側にプルアップしてください。

スマートカードインタフェースで生成するクロックを IC カードで使用する場合は、SCK 端子出力を IC カードの CLK 端子に入力します。IC カードで、内部クロックを使用する場合は接続不要です。

リセット信号としては、LSI のポート出力を使用します。

端子としては、これ以外に通常、電源とグラウンドの接続が必要です。

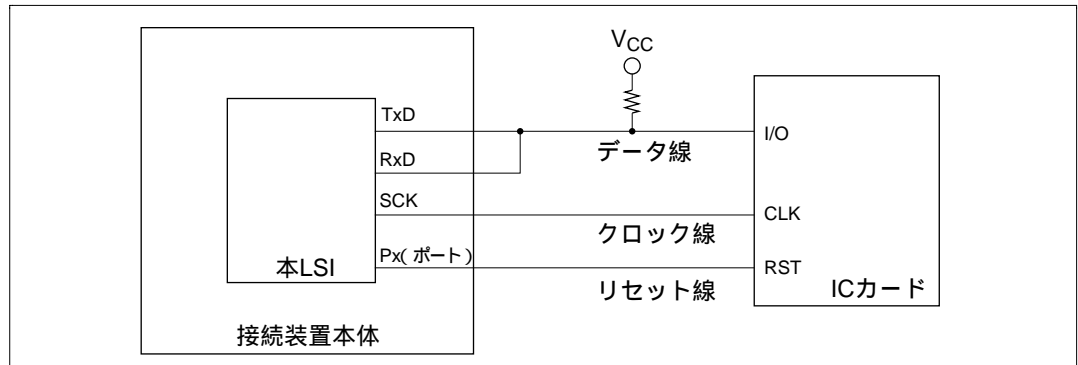


図 14.2 スマートカードインタフェース端子接続概略図

【注】 ICカードを接続しないで、RE = TE = 1 に設定すると、閉じた送信 / 受信が可能となり自己診断をすることができます。

### 14.3.3 データフォーマット

図 14.3 にスマートカードインタフェースのデータフォーマットを示します。このモードでは、受信時は1フレームごとにパリティチェックを行い、エラーが検出された場合、送信側に対してエラー信号を送り返し、データの再送信要求をします。送信時は、エラー信号をサンプリングすると同じデータを再送信します。

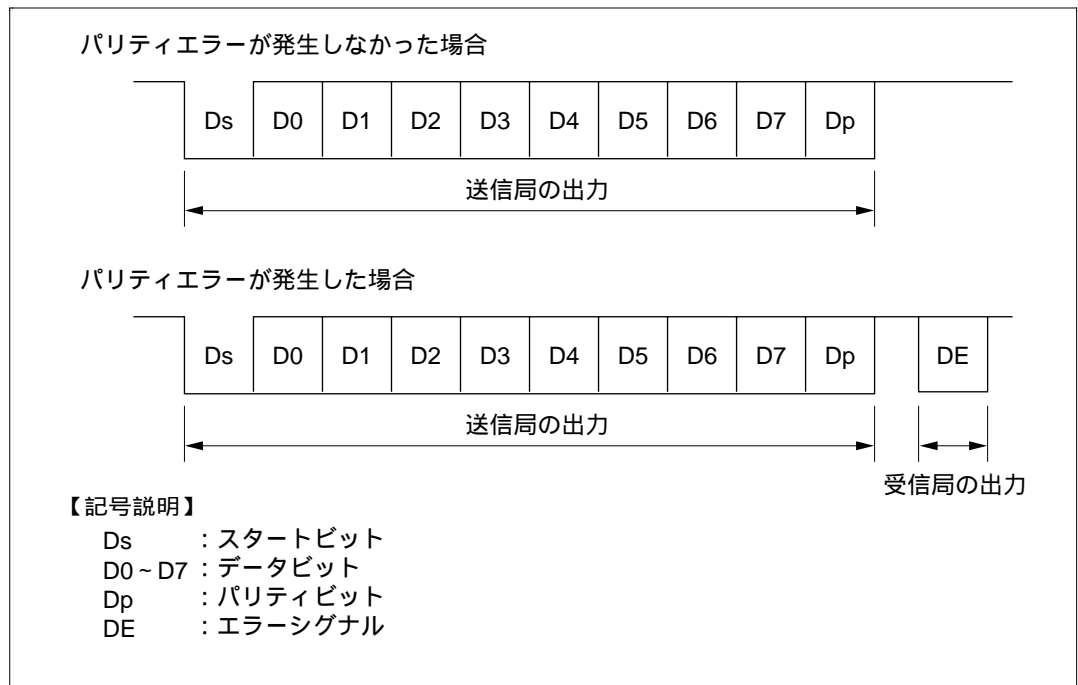


図 14.3 スマートカードインタフェースのデータフォーマット

動作シーケンスは次のようになっています。

- [1] データ線は、未使用時にはハイインピーダンス状態であり、プルアップ抵抗により High レベルに固定されます。
- [2] 送信側は、1 フレームのデータ送信を開始します。データのフレームは、スタートビット (Ds、Low レベル) から開始します。この後に、8 ビットのデータビット (D0~D7) とパリティビット (Dp) が続きます。
- [3] スマートカードインタフェースでは、この後にデータ線をハイインピーダンスに戻します。データ線はプルアップ抵抗により High レベルになります。
- [4] 受信側は、パリティチェックを行います。  
パリティエラーが無く正常に受信した場合、そのまま次のデータ受信を待ちます。  
一方、パリティエラーが発生した場合は、エラーシグナル (DE、Low レベル) を出力し、データの再送信を要求します。受信局は、規定の期間エラーシグナルを出力した後、再び信号線をハイインピーダンスにします。信号線はプルアップ抵抗により High レベルに戻ります。
- [5] 送信側は、エラーシグナルを受信しなかった場合、次のフレームのデータ送信に移ります。  
一方、エラーシグナルを受信した場合は、[2] に戻りエラーとなったデータを再送信します。

### 14.3.4 レジスタの設定

スマートカードインタフェースで使用するレジスタのビットマップを表 14.3 に示します。

0 または 1 が表示されているビットは、必ず表示されている値を設定してください。以下にそれ以外のビットの設定方法について説明します。

表 14.3 スマートカードインタフェースでのレジスタ設定

レジスタ	アドレス*1	ビット							
		ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
SMR	H'FFFB0	GM	0	1	O $\bar{E}$	1	0	CKS1	CKS0
BRR	H'FFFB1	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0
SCR	H'FFFB2	TIE	RIE	TE	RE	0	0	CKE1*2	CKE0
TDR	H'FFFB3	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0
SSR	H'FFFB4	TDRE	RDRF	ORER	ERS	PER	TEND	0	0
RDR	H'FFFB5	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0
SCMR	H'FFFB6	-	-	-	-	SDIR	SINV	-	SMIF

【注】 - : 未使用ビットを示します。

\*1 : アドバンスモード時の下位 20 ビットを示します。

\*2 : SMR の GM を 0 に設定したときは、必ず CKE1 ビットを 0 にしてください。

#### (1) SMR の設定

GM ビットは、通常のスマートカードインタフェースモード時は 0 を設定し、GSM モード時は 1 を設定します。O $\bar{E}$  ビットは、IC カードがダイレクトコンベンション時には 0 を設定し、インバースコンベンション時には 1 を設定します。

CKS1、CKS0 ビットは、内蔵ポーレートジェネレータのクロックソースを選択します。「14.3.5 クロック」を参照してください。

#### (2) BRR の設定

ビットレートを設定します。設定値の算出方法は「14.3.5 クロック」を参照してください。

#### (3) SCR の設定

TIE、RIE、TE、RE ビットの機能は通常の SCI と同様です。詳細は「第 13 章 シリアルコミュニケーションインタフェース」を参照してください。

CKE1、CKE0 ビットはクロック出力を指定します。SMR の GM ビットが 0 にクリアされているとき、クロックを出力しない場合は 00 に設定し、クロックを出力する場合は 01

に設定します。SMR の GM ビットが 1 にセットされているとき、クロック出力を行います。クロック出力を Low レベルまたは High レベルに固定することもできます。

#### (4) スマートカードモードレジスタ (SCMR) の設定

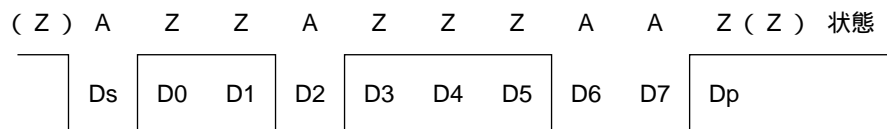
SDIR ビットは、IC カードがダイレクトコンベンション時は 0 を設定し、インバースコンベンション時は 1 を設定します。

SINV ビットは IC カードがダイレクトコンベンション時は 0 を設定し、インバースコンベンション時は 1 を設定します。

SMIF ビットはスマートカードインタフェースの場合 1 を設定します。

以下に、2 種類の IC カード (ダイレクトコンベンションタイプとインバースコンベンションタイプ) に対するレジスタ設定値と、開始キャラクタでの波形例を示します。

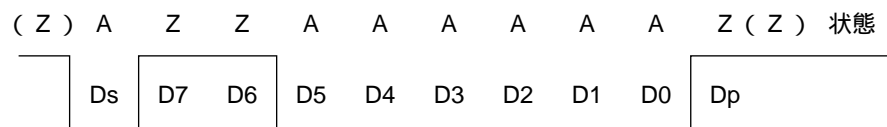
##### (a) ダイレクトコンベンション (SDIR = SINV = $O/\bar{E}$ = 0)



ダイレクトコンベンションタイプは、論理 1 レベルを状態 Z に、論理 0 レベルを状態 A に対応付け、LSB ファーストで送受信する方式です。上記の開始キャラクタのデータは H'3B となります。

パリティビットは、スマートカードの規格により偶数パリティで 1 となります。

##### (b) インバースコンベンション (SDIR = SINV = $O/\bar{E}$ = 1)



インバースコンベンションタイプは、論理 1 レベルを状態 A に、論理 0 レベルを状態 Z に対応付け、MSB ファーストで送受信する方式です。上記の開始キャラクタのデータは H'3F となります。

パリティビットは、スマートカードの規格により偶数パリティで論理 0 となり、状態 Z が対応します。

なお、本 LSI では、SINV ビットによる反転はデータビット D7 ~ D0 のみとなっています。パリティビットの反転のために SMR の  $O/\bar{E}$  ビットを奇数パリティモードに設定します (送信、受信とも同様です)。

### 14.3.5 クロック

スマートカードインタフェースにおける送受信クロックは、内蔵ポーレートジェネレータの生成した内部クロックのみ使用できます。このとき、ビットレートはBRRとSMRのCKS1、CKS0ビットで設定され、以下に示す計算式になります。ビットレートの例を表14.5に示します。

このときCKE0=1でクロック出力を選択すると、SCK端子からはビットレートの372倍の周波数のクロックが出力されます。

$$B = \frac{1488 \times 2^{2n-1} \times (N+1)}{1488 \times 2^{2n-1} \times (N+1)} \times 10^6$$

ただし、N=BRRの設定値(0 N 255)

B = ビットレート (bit/s)

= 動作周波数 (MHz)

n = 表 14.4 を参照

表 14.4 nとCKS1、CKS0の対応表

n	CKS1	CKS0
0	0	0
1		1
2	1	0
3		1

【注】 \* ギア機能によりクロックを分周して使用する場合には、動作周波数に分周比を考慮した値を設定してください。上記は分周比1:1の場合を示します。

表 14.5 BRRの設定に対するビットレートB (bit/s)の例(ただし、n=0のとき)

N	(MHz)						
	7.1424	10.00	10.7136	13.00	14.2848	16.00	18.00
0	9600.0	13440.9	14400.0	17473.1	19200.0	21505.4	24193.5
1	4800.0	6720.4	7200.0	8736.6	9600.0	10752.7	12096.8
2	3200.0	4480.3	4800.0	5824.4	6400.0	7168.5	8064.5

【注】 ビットレートは、小数点以下2桁目を四捨五入した値です。

一方、動作周波数とビットレートからビットレートレジスタ (BBR) の設定値を算出する式は次のようになります。ただし、N は整数値、0 ≤ N ≤ 255 であり、誤差の小さい方を指定します。

$$N = \frac{1488 \times 2^{2n-1} \times B}{1488 \times 2^{2n-1} \times B} \times 10^6 - 1$$

表 14.6 ビットレート B (bit/s) に対する BBR の設定例 (ただし、n=0 のとき)

bit/s	(MHz)													
	7.1424		10.00		10.7136		13.00		14.2848		16.00		18.00	
	N	誤差	N	誤差	N	誤差	N	誤差	N	誤差	N	誤差	N	誤差
9600	0	0.00	1	30	1	25	1	8.99	1	0.00	1	12.01	2	15.99

表 14.7 各周波数における最大ビットレート (スマートカードインタフェースモード時)

(MHz)	最大ビットレート (bit/s)	N	n
7.1424	9600	0	0
10.00	13441	0	0
10.7136	14400	0	0
13.00	17473	0	0
14.2848	19200	0	0
16.00	21505	0	0
18.00	24194	0	0

ビットレート誤差は以下の計算式で求められます。

$$\text{誤差 (\%)} = \left( \frac{1488 \times 2^{2n-1} \times B}{1488 \times 2^{2n-1} \times B \times (N + 1)} \times 10^6 - 1 \right) \times 100$$

### 14.3.6 データの送信 / 受信動作

#### (1) 初期設定

データの送受信の前に、以下の手順で SCI をイニシャライズしてください。送信モードから受信モードへの切り替え、受信モードから送信モードへの切り替えにおいてもイニシャライズが必要です。

- [ 1 ] SCR の TE、RE ビットを 0 にクリアします。
- [ 2 ] SSR のエラーフラグ ERS、PER、ORER を 0 にクリアしてください。
- [ 3 ] SMR の  $O/\bar{E}$  ビットと CKS1、CKS0 ビットを設定してください。このとき、 $C/\bar{A}$ 、CHR、MP ビットは 0 に、STOP、PE ビットは 1 に設定してください。
- [ 4 ] SCMR の SMIF、SDIR、SINV ビットを設定してください。  
SMIF ビットを 1 にセットすると、TxD 端子および RxD 端子は共にポートから SCI の端子に切り替えられ、ハイインピーダンス状態となります。
- [ 5 ] ビットレートに対応する値を BRR に設定します。
- [ 6 ] SCR の CKE0 ビットを設定してください。このとき、TIE、RIE、TE、RE、MPIE、TEIE、CKE1 ビットは、0 に設定してください。  
CKE0 ビットを 1 にセットした場合は、SCK 端子からクロック出力されます。
- [ 7 ] 少なくとも 1 ビット期間待ってから、SCR の TIE、RIE、TE、RE ビットを設定してください。自己診断以外は TE ビットと RE ビットを同時にセットしないでください。



(2) シリアルデータ送信

スマートカードモードにおけるデータ送信では、エラーシグナルのサンプリングと再送信処理があるため、通常の SCI とは処理手順が異なります。送信処理フローの例を図 14.5 に示します。

- [1] (1) の手順にしたがいスマートカードインタフェースモードにイニシャライズします。
- [2] SSR のエラーフラグ ERS が 0 にクリアされていることを確認してください。
- [3] SSR の TEND フラグが 1 にセットされていることが確認できるまで、[2]、[3] を繰り返してください。
- [4] TDR に送信データをライトして、TDRE フラグを 0 にクリアし送信動作を行います。  
このとき、TEND フラグは 0 にクリアされます。
- [5] 連続してデータを送信する場合は、[2] に戻ってください。
- [6] 送信を終了する場合は、TE ビットを 0 にクリアします。

以上の一連の処理は、割込み処理または DMA 転送が可能です。

TIE ビットを 1 にセットし、割込み要求を許可しておいたとき、送信が終了し TEND フラグが 1 にセットされると、送信データエンプティ割込み (TXI) 要求を発生します。RIE ビットを 1 にセットし、割込み要求を許可しておいたとき、送信時にエラーが発生し、ERS フラグが 1 にセットされると、送受信エラー割込み (ERI) 要求を発生します。

SMR の GM ビットにより、TEND フラグのセットタイミングが異なります。図 14.4 に TEND フラグセットタイミングを示します。

TXI 要求で DMAC を起動する場合、自動再転送を含め DMAC に設定したバイト数を自動的に送信することができます。

詳細は (6)、(7) を参照してください。

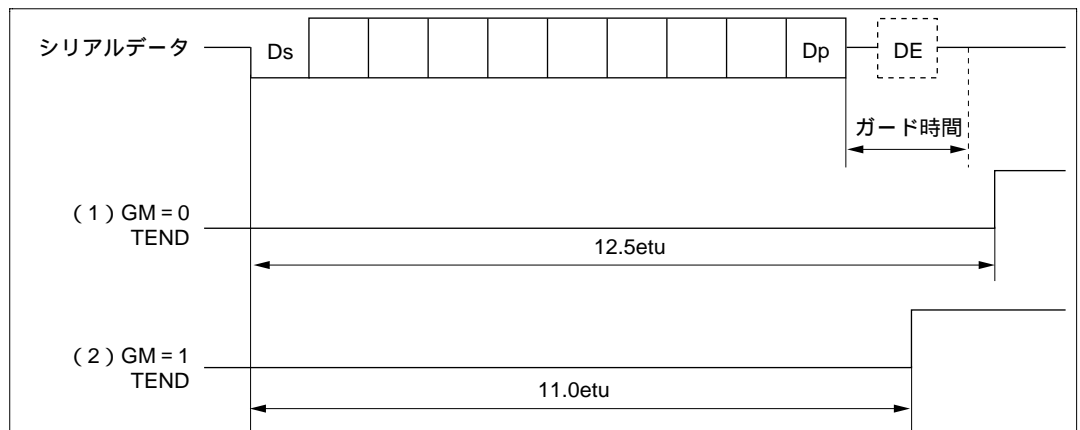


図 14.4 TEND フラグセットタイミング

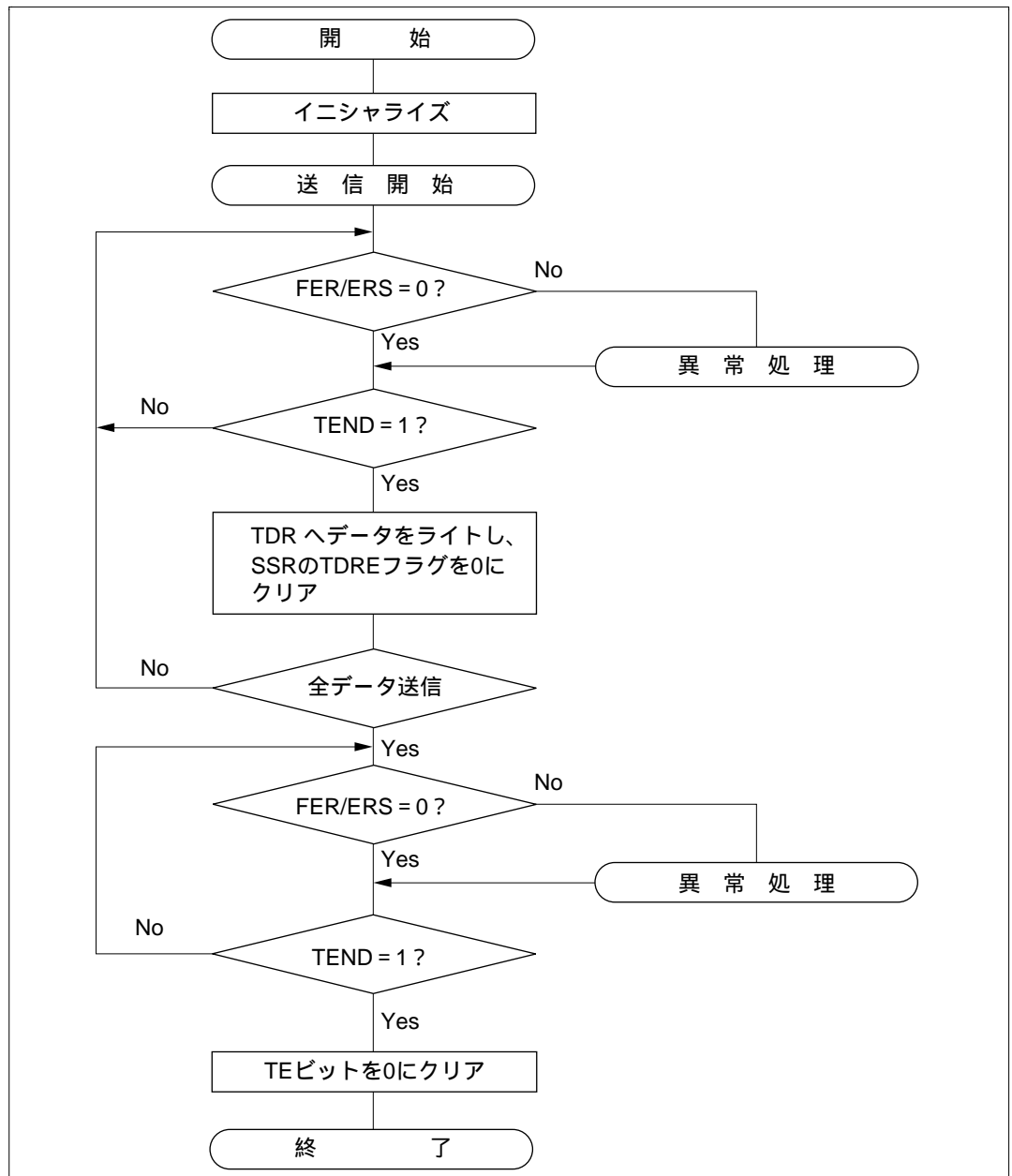


図 14.5 送信処理フローの例

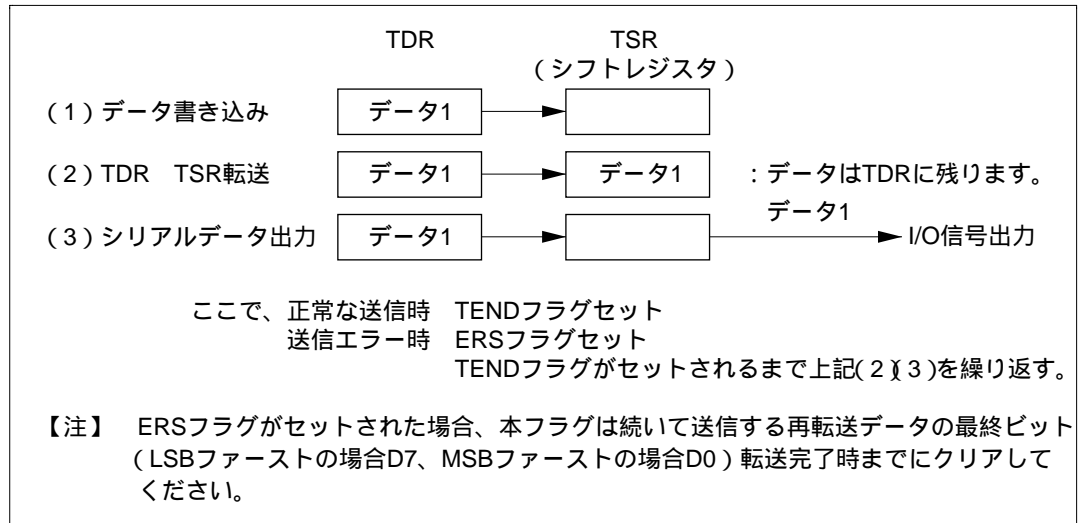


図 14.6 送信動作と内部レジスタの関連

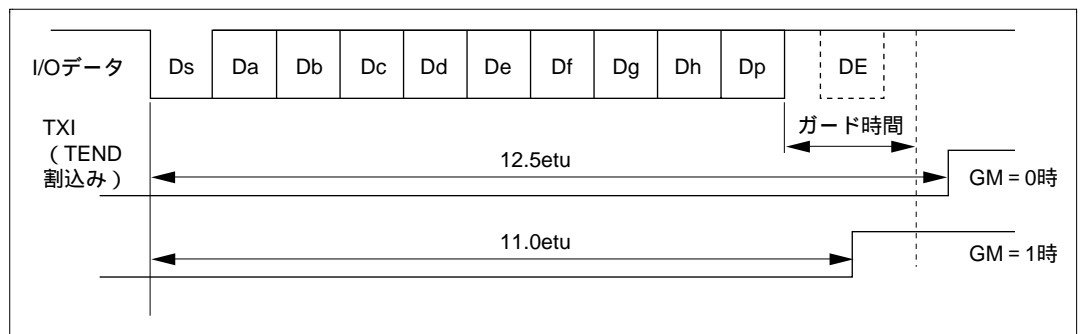


図 14.7 TEND フラグ発生タイミング

### (3) シリアルデータ受信

スマートカードモードのデータ受信は、通常のSCIと同様の処理手順になります。受信処理フローの例を図14.8に示します。

- [1] SCIを(1)に従いスマートカードインタフェースモードにイニシャライズします。
- [2] SSRのORERフラグとPERフラグが0であることを確認してください。どちらかのフラグがセットされている場合は、所定の受信異常処理を行った後、ORERとPERフラグを全て0にクリアしてください。
- [3] RDRFフラグが1であることを確認できるまで[2]、[3]を繰り返してください。
- [4] RDRから受信データをリードしてください。
- [5] 継続してデータを受信する場合は、RDRFフラグを0にクリアして[2]の手順に戻ってください。
- [6] 受信を終了する場合は、REビットを0にクリアします。

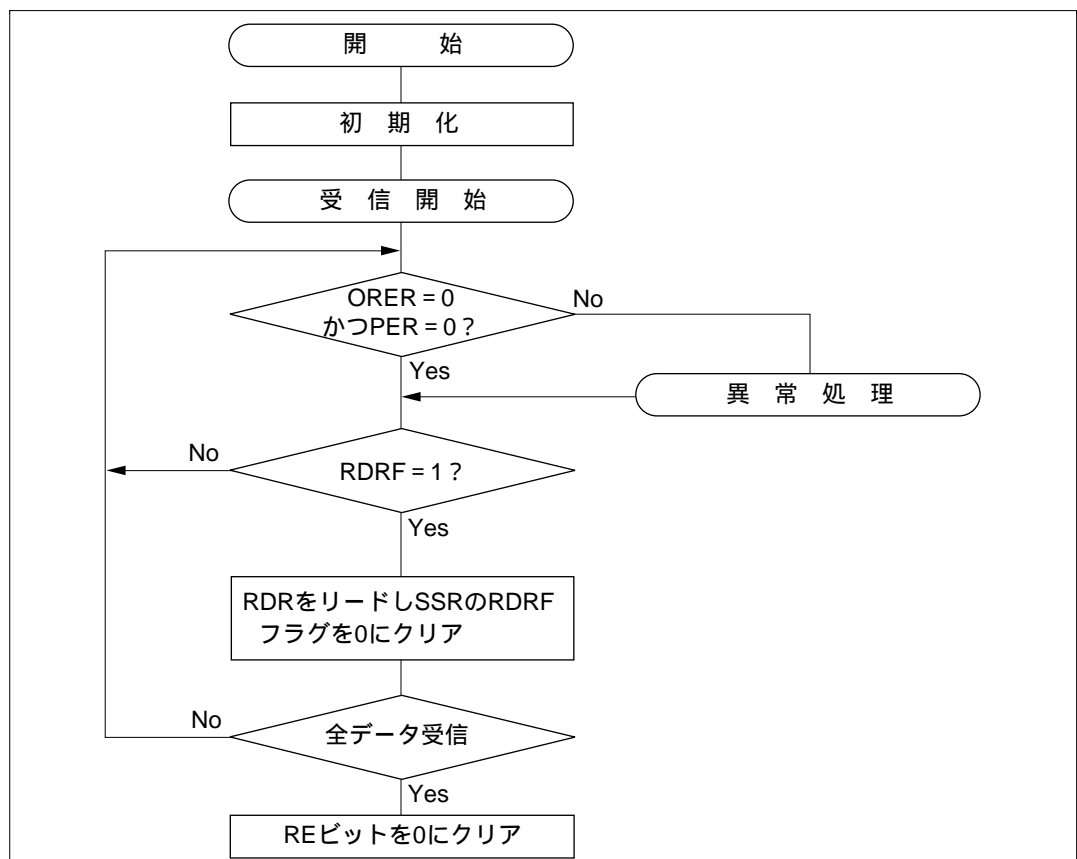


図 14.8 受信処理フローの例

以上の一連の処理は、割込み処理またはDMA転送が可能です。

RIEビットを1にセットし、割込み要求を許可しておいたとき受信が終了し、RDRFフラグが1にセットされると、受信データフル割込み(RXI)要求が発生します。また、受信時にエラーが発生しORER、PERフラグのいずれかが1にセットされると、送受信エラー割込み(ERI)要求が発生します。

RXI要求でDMACを起動する場合、エラーの発生した受信データをスキップしてDMACに設定したバイト数だけ受信データを転送します。

詳細は(6)、(7)を参照してください。

なお、受信時にパリティエラーが発生しPERが1にセットされた場合でも、受信したデータはRDRに転送されるのでこのデータをリードすることは可能です。

#### (4) モード切り替え動作

受信モードから送信モードに切り替える場合、受信動作が完了していることを確認した後、イニシャライズから開始し、RE=0、TE=1に設定してください。受信動作の完了は、RDRFフラグ、あるいはPER、ORERフラグで確認できます。

送信モードから受信モードに切り替える場合、送信動作が完了していることを確認した後、イニシャライズから開始し、TE=0、RE=1に設定してください。送信動作の完了はTENDフラグで確認できます。

#### (5) クロック出力の固定

SMRのGMビットが1にセットされているとき、SCRのCKE1、CKE0ビットによってクロック出力を固定することができます。このときクロックパルスの最小幅を指定の幅とすることができます。

図14.9にクロック出力の固定タイミングを示します。GM=1、CKE1=0とし、CKE0ビットを制御した場合の例です。

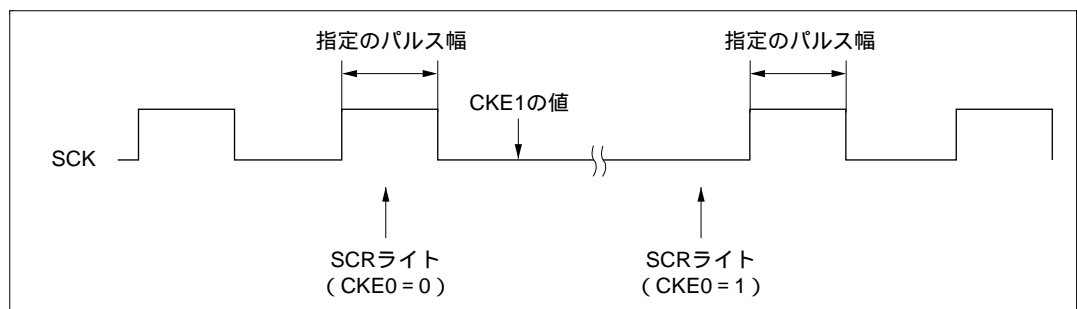


図14.9 クロック出力固定タイミング

## (6) 割込み動作

スマートカードインタフェースモードでは、送信データエンプティ割込み(TXI)要求、送受信エラー割込み(ERI)要求、受信データフル割込み(RXI)要求の3種類の割込み要因があります。なお、本モードでは、送信終了割込み(TEI)要求は使用できません。

SSRのTENDフラグが1にセットされると、TXI割込み要求を発生します。

SSRのRDRFフラグが1にセットされると、RXI割込み要求を発生します。

SSRのORER、PER、ERSフラグのいずれかが1にセットされると、ERI割込み要求を発生します。これらの関係を表14.8に示します。

表 14.8 スマートカードインタフェースモードの動作状態と割込み要因

動作状態		フラグ	許可ビット	割込み要因	DMACの起動
送信モード	正常動作	TEND	TIE	TXI	可
	エラー	ERS	RIE	ERI	不可
受信モード	正常動作	RDRF	RIE	RXI	可
	エラー	PER、ORER	RIE	ERI	不可

## (7) DMACによるデータ転送動作

スマートカードモードの場合も通常のSCIの場合と同様に、DMACを使って送受信を行うことができます。送信動作では、SSRのTENDフラグが1にセットされると同時にTDREフラグもセットされ、TXI割込みが発生します。あらかじめDMACの起動要因にTXI要求を設定しておけば、TXI要求によりDMACが起動されて送信データの転送を行います。TDREおよびTENDフラグは、DMACによるデータ転送時に自動的に0にクリアされます。エラーが発生した場合はSCIが自動的に同じデータを再送信します。この間TENDは0のまま保持され、DMACは起動されません。したがって、エラー発生時の再送信を含め、SCIとDMACが指定されたバイト数を自動的に送信します。ただし、エラー発生時、ERSフラグは自動的にクリアされませんので、RIEビットを1にセットしておき、エラー発生時にERI要求を発生させ、ERSをクリアしてください。

なお、DMACを使って送受信を行う場合は、必ず先にDMACを設定し、イネーブル状態にしてからSCIの設定を行ってください。DMACの設定方法は「第7章 DMAコントローラ」を参照してください。

また、受信動作では、SSRのRDRFフラグが1にセットされるとRXI割込み要求が発生します。あらかじめDMACの起動要因にRXI要求を設定しておけば、RXI要求でDMACが起動されて受信データの転送を行います。RDRFフラグは、DMACによるデータ転送時に、自動的に0にクリアされます。エラーが発生した場合は、RDRFフラグはセットされずエラーフラグがセットされます。DMACは起動されず、かわりにCPUに対しERIを発生しますのでエラーフラグをクリアしてください。

(8) GSM モード時の動作例

スマートカードインタフェースモードとソフトウェアスタンバイ間でモード切替えを行う際、クロックデューティを保持するため、下記の切替え手順で処理してください。

スマートカードインタフェースモードからソフトウェアスタンバイモードに遷移するとき

- [1] P9<sub>4</sub>のデータレジスタ (DR) とデータディレクションレジスタ (DDR) をソフトウェアスタンバイ時の出力固定状態の値に設定する。
- [2] シリアルコントロールレジスタ (SCR) の TE ビットと RE ビットに 0 を書き込み、送信 / 受信動作を停止させる。同時に、CKE1 ビットをソフトウェアスタンバイ時の出力固定状態の値に設定する。
- [3] SCR の CKE0 ビットに 0 を書き込み、クロックを停止させる。
- [4] シリアルクロックの 1 クロック周期の間、待つ。  
この間に、デューティを守って、指定のレベルでクロック出力は固定される。
- [5] シリアルモードレジスタ (SMR) とスマートカードモードレジスタ (SCMR) に H'00 を書き込む。
- [6] ソフトウェアスタンバイ状態に遷移させる。

ソフトウェアスタンバイモードからスマートカードインタフェースモードに戻るとき

- [1] ソフトウェアスタンバイ状態を解除する。
- [2] SCR の CKE1 ビットをソフトウェアスタンバイ開始時の出力固定状態 (現在の P9<sub>4</sub> 端子) の値に設定する。
- [3] スマートカードインタフェースモードに設定し、クロック出力させる。正常なデューティにてクロック信号発生を開始する。

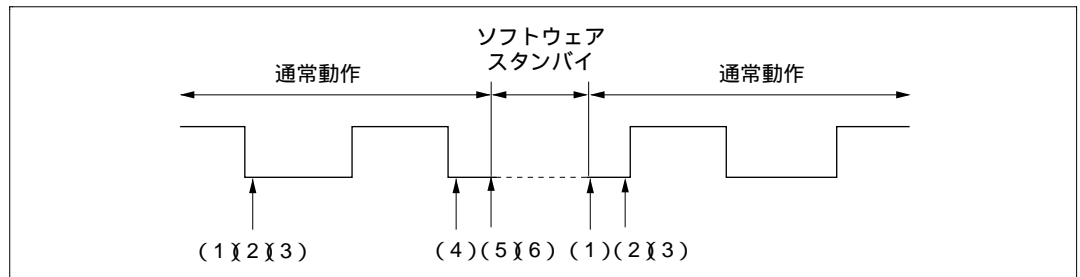


図 14.10 クロック停止・再起動手順

電源投入時からクロックデューティを確保するためには、下記の切替え手順で処理をしてください。

- [ 1 ] 初期状態は、ポート入力でありハイインピーダンスである。電位を固定するには、プルアップ抵抗 / プルダウン抵抗を使用する。
- [ 2 ] SCR の CKE1 ビットで指定の出力に固定する。
- [ 3 ] SMR と SCMR をセットし、スマートカードモードの動作に切替える。
- [ 4 ] SCR の CKE0 ビットを 1 に設定して、クロック出力を開始する。



## 14.4 使用上の注意

SCI をスマートカードインタフェースとして使用する際は、以下のことに注意してください。

### (1) スマートカードインタフェースモードの受信データサンプリングタイミングと受信マージン

スマートカードインタフェースモードでは、SCI は転送レートの 372 倍の周波数の基本クロックで動作しています。

受信時に SCI は、スタートビットの立ち下がりを基本クロックでサンプリングして、内部を同期化します。また、受信データを基本クロックの 186 クロック目の立ち上がりエッジで内部に取り込みます。これを図 14.11 に示します。

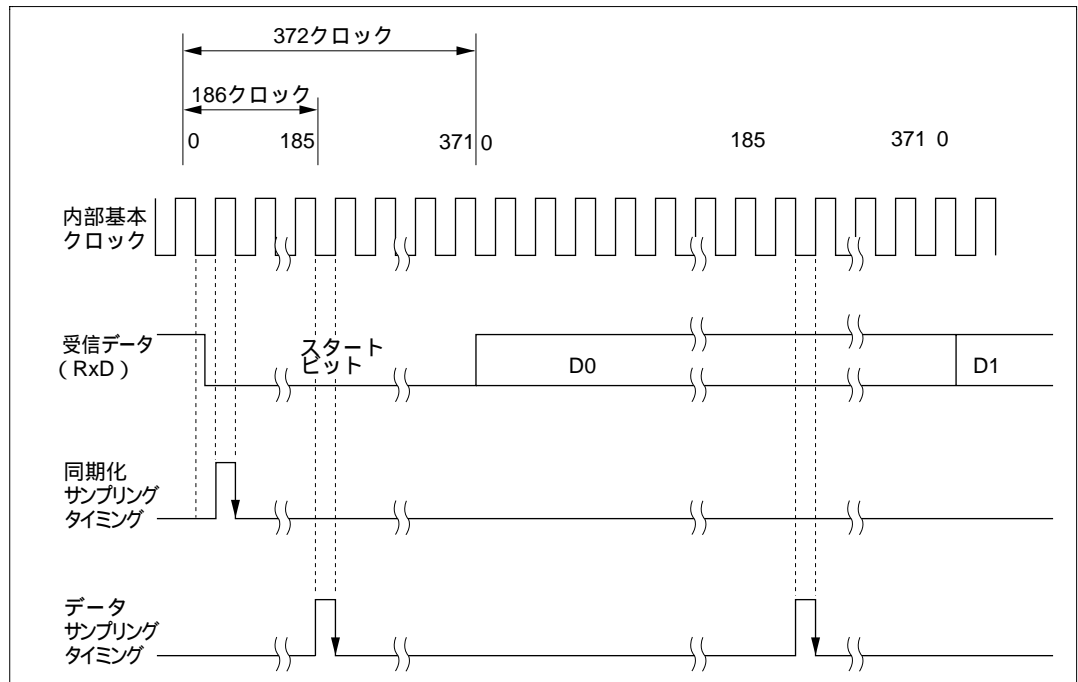


図 14.11 スマートカードインタフェースモード時の受信データサンプリングタイミング

したがって、受信マージンは、次の式のように表すことができます。

スマートカードインタフェースモード時の受信マージン式

$$M = \left| \left( 0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\%$$

M：受信マージン（%）

N：クロックに対するビットレートの比（N=372）

D：クロックデューティー（D=0~1.0）

L：フレーム長（L=10）

F：クロック周波数の偏差の絶対値

上式で、F=0、D=0.5 とすると、受信マージン式は次のようになります。

D=0.5、F=0 のとき、

$$M = \left( 0.5 - \frac{1}{2 \times 372} \right) \times 100\%$$

$$= 49.866\%$$

(2) 再転送動作

SCI がそれぞれ受信モードの場合と、送信モードの場合の再転送動作を、次に示します。

(a) SCI が受信モードの場合の再転送動作

SCI 受信モードの場合の再転送動作を図 14.12 に示します。

- [1] 受信したパリティビットをチェックした結果、エラーが検出されると、SSR の PER ビットが自動的に 1 にセットされます。このとき、SCR の RIE ビットがイネーブルになっていれば、ERI 割込み要求が発生します。次のパリティビットのサンプリングタイミングまでに、SSR の PER ビットを 0 にクリアしてください。
- [2] 異常が発生したフレームでは、SSR の RDRF ビットはセットされません。
- [3] 受信したパリティビットをチェックした結果、エラーが検出されない場合は、SSR の PER ビットはセットされません。
- [4] 受信したパリティビットをチェックした結果、エラーが検出されない場合は、正常に受信動作が完了したと判断して、SSR の RDRF ビットが自動的に 1 にセットされます。このとき SCR の RIE ビットが許可になっていれば、RXI 割込み要求が発生します。さらに、RXI 要因による DMAC のデータ転送が許可されていれば、RDR の内容を自動的にリードすることができます。DMAC で RDR のデータをリードした場合、RDRF フラグは自動的に 0 にクリアされます。
- [5] 正常なフレームを受信した場合、エラーシグナルを送信するタイミングで端子はスリーステート状態を保持します。

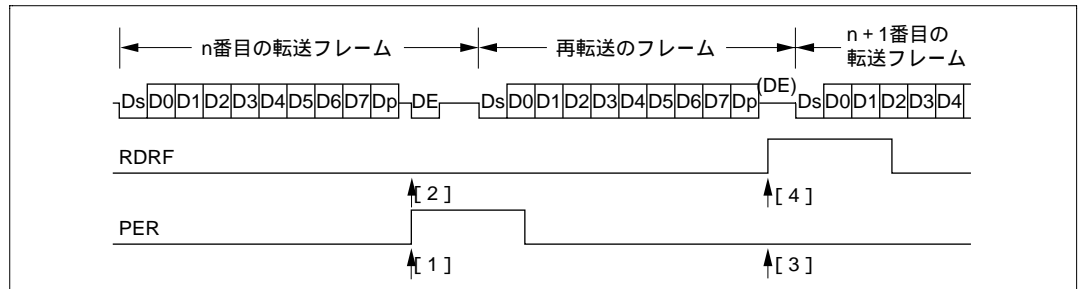


図 14.12 SCI 受信モードの場合の再転送動作

## (b) SCI が送信モードの場合の再転送動作

SCI 送信モードの場合の再転送動作を図 14.13 に示します。

- [6] 1 フレーム分の送信を完了した後、受信側からエラーシグナルが返されると、SSR の ERS ビットが 1 にセットされます。このとき、SCR の RIE ビットが許可になっていれば、ERI 割込み要求が発生します。次のパリティビットのサンプリングタイミングまでに、SSR の ERS ビットを 0 にクリアしてください。
- [7] 異常を示すエラーシグナルを受信したフレームでは、SSR の TEND ビットはセットされません。
- [8] 受信側からエラーシグナルが返ってこない場合は、SSR の ERS ビットはセットされません。
- [9] 受信側からエラーシグナルが返ってこない場合は、再転送を含む 1 フレームの送信が完了したと判断して、SSR の TEND ビットが 1 にセットされます。このとき SCR の TIE ビットが許可になっていれば、TXI 割込み要求が発生します。
- さらに、TXI 要因による DMAC のデータ転送が許可されていれば、自動的に TDR に次のデータをライトすることができます。DMAC で TDR にデータをライトした場合、TDRE ビットは自動的に 0 にクリアされます。

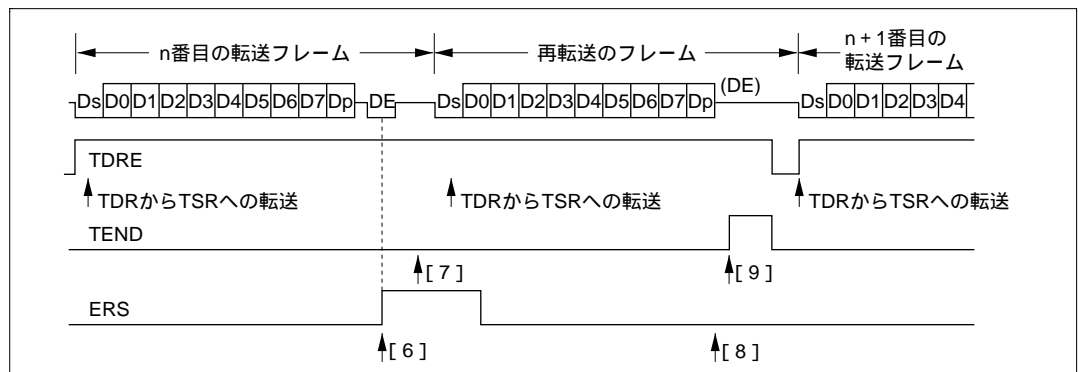


図 14.13 SCI 送信モードの場合の再転送動作



---

# 15. A/D 変換器

---

## 第 15 章 目次

15.1	概要.....	613
15.1.1	特長.....	613
15.1.2	ブロック図.....	614
15.1.3	端子構成.....	615
15.1.4	レジスタ構成.....	616
15.2	各レジスタの説明.....	617
15.2.1	A/D データレジスタ A ~ D ( ADDRA ~ D ) .....	617
15.2.2	A/D コントロール / ステータスレジスタ ( ADCSR ) .....	618
15.2.3	A/D コントロールレジスタ ( ADCR ) .....	621
15.3	CPU とのインタフェース.....	622
15.4	動作説明.....	623
15.4.1	単一モード ( SCAN = 0 ) .....	623
15.4.2	スキャンモード ( SCAN = 1 ) .....	625
15.4.3	入力サンプリングと A/D 変換時間.....	627
15.4.4	外部トリガ入力タイミング.....	628
15.5	割込み.....	629
15.6	使用上の注意.....	629



## 15.1 概要

本 LSI には、逐次比較方式で動作する 10 ビットの A/D 変換器が内蔵されており、最大 8 チャンネルのアナログ入力を選択することができます。

消費電流低減のために A/D 変換器を使用しない場合には、A/D 変換器を単独に停止することができます。詳細は「20.6 モジュールスタンバイ機能」を参照してください。

### 15.1.1 特長

A/D 変換器の特長を以下に示します。

10 ビットの分解能

入力チャンネル：8 チャンネル

アナログ変換電圧範囲の設定が可能

リファレンス電圧端子 ( $V_{REF}$ ) をアナログ基準電圧としてアナログ変換電圧範囲を設定します。

高速変換

変換時間：1 チャンネル当り最小 3.5  $\mu$ s (20MHz 動作時)

単一モード / スキャンモードの 2 種類の動作モードから選択可能

単一モード：1 チャンネルの A/D 変換

スキャンモード：1 ~ 4 チャンネルの連続 A/D 変換

4 本の 16 ビットデータレジスタ

A/D 変換された結果は、各チャンネルに対応したデータレジスタに転送され、保持されます。

サンプル & ホールド機能

3 種類の変換開始要求

ソフトウェア、外部トリガ信号または 8 ビットタイマのコンペアマッチによる、A/D 変換の開始が可能

A/D 変換終了割込み要求を発生

A/D 変換終了時には、A/D 変換終了割込み (ADI) 要求を発生させることができます。



DMAコントローラ(DMAC)の起動が可能

A/D変換終了割り込みにより、DMACの起動が可能

### 15.1.2 ブロック図

A/D変換器のブロック図を図15.1に示します。

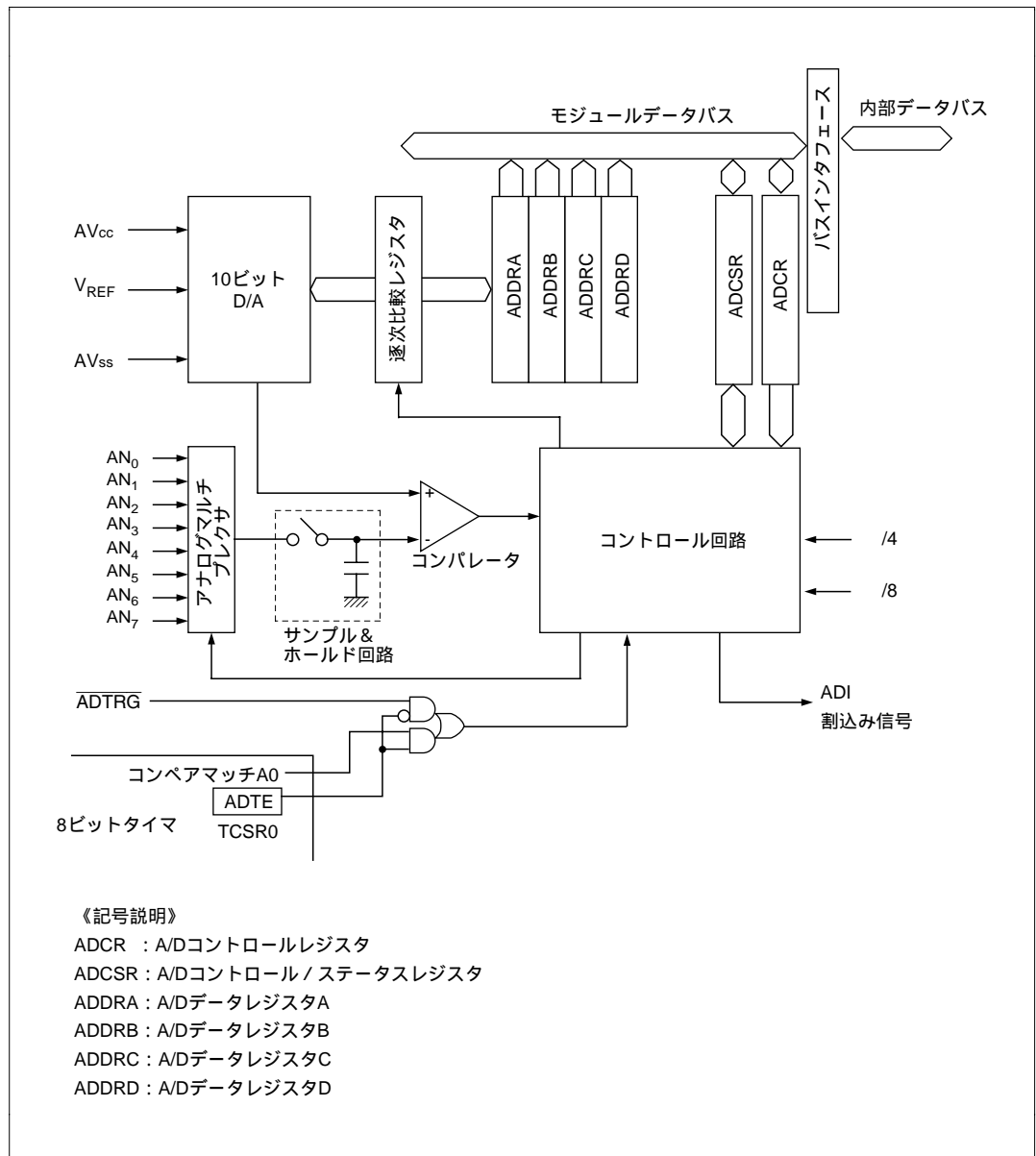


図 15.1 A/D変換器のブロック図

### 15.1.3 端子構成

A/D変換器で使用する入力端子を表15.1に示します。

8本のアナログ入力端子は2グループに分類されており、アナログ入力端子0～3 (AN<sub>0</sub>～AN<sub>3</sub>)がグループ0、アナログ入力端子4～7 (AN<sub>4</sub>～AN<sub>7</sub>)がグループ1になっています。

AV<sub>CC</sub>、AV<sub>SS</sub>端子は、A/D変換器内のアナログ部の電源です。V<sub>REF</sub>端子は、A/D変換基準電圧端子です。

表 15.1 端子構成

端子名	略称	入出力	機能
アナログ電源端子	AV <sub>CC</sub>	入力	アナログ部の電源
アナロググランド端子	AV <sub>SS</sub>	入力	アナログ部のグランドおよび基準電圧
リファレンス電圧端子	V <sub>REF</sub>	入力	アナログ部の基準電圧
アナログ入力端子0	AN <sub>0</sub>	入力	グループ0のアナログ入力
アナログ入力端子1	AN <sub>1</sub>	入力	
アナログ入力端子2	AN <sub>2</sub>	入力	
アナログ入力端子3	AN <sub>3</sub>	入力	
アナログ入力端子4	AN <sub>4</sub>	入力	グループ1のアナログ入力
アナログ入力端子5	AN <sub>5</sub>	入力	
アナログ入力端子6	AN <sub>6</sub>	入力	
アナログ入力端子7	AN <sub>7</sub>	入力	
A/D外部トリガ入力端子	$\overline{\text{ADTRG}}$	入力	A/D変換開始のための外部トリガ入力

## 15.1.4 レジスタ構成

A/D変換器のレジスタ構成を表15.2に示します。

表 15.2 レジスタ構成

アドレス <sup>*1</sup>	名称	略称	R/W	初期値
H'FFFE0	A/D データレジスタ AH	ADDRAH	R	H'00
H'FFFE1	A/D データレジスタ AL	ADDRAL	R	H'00
H'FFFE2	A/D データレジスタ BH	ADDRBH	R	H'00
H'FFFE3	A/D データレジスタ BL	ADDRBL	R	H'00
H'FFFE4	A/D データレジスタ CH	ADDRCH	R	H'00
H'FFFE5	A/D データレジスタ CL	ADDRCL	R	H'00
H'FFFE6	A/D データレジスタ DH	ADDRDH	R	H'00
H'FFFE7	A/D データレジスタ DL	ADDRDL	R	H'00
H'FFFE8	A/D コントロール/ステータスレジスタ	ADCSR	R/(W) <sup>*2</sup>	H'00
H'FFFE9	A/D コントロールレジスタ	ADCR	R/W	H'7E

【注】 \*1 アドバンスモード時のアドレス下位 20 ビットを示しています。

\*2 ビット 7 は、フラグをクリアするための 0 ライトのみ可能です。

## 15.2 各レジスタの説明

### 15.2.1 A/D データレジスタ A~D (ADDRA~D)

ビット:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRn:	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—
初期値:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(n=A~D)

A/D変換データ  
A/D変換結果の10ビットデータを  
格納するビットです。

リザーブビット

ADDR は、A/D 変換された結果を格納する 16 ビットのリード専用レジスタで、ADDRA ~ ADDR D の 4 本があります。

A/D 変換されたデータは 10 ビットデータで、選択されたチャンネルの ADDR に転送され、保持されます。A/D 変換されたデータの上位 8 ビットが ADDR の上位バイトに、また下位 2 ビットが下位バイトに対応します。ADDR の下位バイトのビット 5~0 はリザーブビットで、リードすると常に 0 が読み出されます。アナログ入力チャンネルと ADDR の対応を表 15.3 に示します。

ADDR は、常に CPU からリード可能です。上位バイトは直接リードできますが、下位バイトはテンポラリレジスタ (TEMP) を介してデータ転送が行われます。詳細は「15.3 CPU とのインタフェース」を参照してください。

ADDR は、リセットまたはスタンバイモード時に、H'0000 にイニシャライズされます。

表 15.3 アナログ入力チャンネルと ADDR A ~ ADDR D の対応

アナログ入力チャンネル		A/D データレジスタ
グループ 0	グループ 1	
AN <sub>0</sub>	AN <sub>4</sub>	ADDR A
AN <sub>1</sub>	AN <sub>5</sub>	ADDR B
AN <sub>2</sub>	AN <sub>6</sub>	ADDR C
AN <sub>3</sub>	AN <sub>7</sub>	ADDR D

### 15.2.2 A/D コントロール / ステータスレジスタ (ADCSR)



【注】 \* フラグをクリアするための0ライトのみ可能です。

ADCSR は、8 ビットのリード / ライト可能なレジスタで、モードの選択など A/D 変換器の動作を制御します。

ADCSR は、リセットまたはスタンバイモード時に、H'00 にイニシャライズされます。

#### ビット7 : A/D エンドフラグ (ADF)

A/D 変換の終了を示すステータスフラグです。

ビット7	説明
ADF	
0	[ クリア条件 ] (初期値) (1) ADF = 1 の状態で、ADF フラグをリードした後、ADF フラグに 0 をライトしたとき (2) ADI 割込みにより DMAC が起動されたとき
1	[ セット条件 ] (1) 単一モード : A/D 変換が終了したとき (2) スキャンモード : 設定されたすべてのチャンネルの A/D 変換が終了したとき

## ビット6：A/Dインタラプトイネーブル（ADIE）

A/D変換の終了による割込み（ADI）要求の許可／禁止を選択します。

ビット6	説明	
ADIE		
0	A/D変換終了による割込み（ADI）要求を禁止	（初期値）
1	A/D変換終了による割込み（ADI）要求を許可	

## ビット5：A/Dスタート（ADST）

A/D変換の開始／停止を選択します。

A/D変換中は1を保持します。また、ADSTビットはA/D外部トリガ入力端子( $\overline{\text{ADTRG}}$ )または8ビットタイマのコンペアマッチにより1にセットすることもできます。

ビット5	説明	
ADST		
0	A/D変換を停止	（初期値）
1	(1) 単一モード：A/D変換を開始し、変換が終了すると自動的に0にクリア (2) スキャンモード：A/D変換を開始し、ソフトウェア、リセット、またはスタンバイモードによって0にクリアされるまで選択されたチャンネルを順次連続変換	

## ビット4：スキャンモード（SCAN）

A/D変換のモードを、単一モード／スキャンモードから選択します。単一モード／スキャンモード時の動作については、「15.4 動作説明」を参照してください。モードの切換えは、ADST=0の状態で行ってください。

ビット4	説明	
SCAN		
0	単一モード	（初期値）
1	スキャンモード	

## ビット3：クロックセレクト（CKS）

A/D変換時間の設定を行います。

変換時間の切換えは、ADST=0の状態で行ってください。

ビット3	説明	
CKS		
0	変換時間 = 134 ステート (max)	（初期値）
1	変換時間 = 70 ステート (max)	

ビット2~0：チャンネルセレクト2~0 (CH2~0)

SCAN ビットとともにアナログ入力チャンネルを選択します。

チャンネル選択と切換えは、ADST=0の状態で行ってください。

グループ選択	チャンネル選択		説明	
	CH1	CH0	単一モード	スキャンモード
0	0	0	AN <sub>0</sub> (初期値)	AN <sub>0</sub>
	0	1	AN <sub>1</sub>	AN <sub>0</sub> 、AN <sub>1</sub>
	1	0	AN <sub>2</sub>	AN <sub>0</sub> ~ AN <sub>2</sub>
	1	1	AN <sub>3</sub>	AN <sub>0</sub> ~ AN <sub>3</sub>
1	0	0	AN <sub>4</sub>	AN <sub>4</sub>
	0	1	AN <sub>5</sub>	AN <sub>4</sub> 、AN <sub>5</sub>
	1	0	AN <sub>6</sub>	AN <sub>4</sub> ~ AN <sub>6</sub>
	1	1	AN <sub>7</sub>	AN <sub>4</sub> ~ AN <sub>7</sub>

### 15.2.3 A/D コントロールレジスタ (ADCR)

ビット:	7	6	5	4	3	2	1	0
	TRGE	—	—	—	—	—	—	—
初期値:	0	1	1	1	1	1	1	0
R/W :	R/W	—	—	—	—	—	—	R/W

リザーブビット

トリガイネーブル

外部トリガまたは8ビットタイマのコンペアマッチによるA/D変換の許可/禁止を選択するビットです。

ADCR は、8ビットのリード/ライト可能なレジスタで、外部トリガ入力あるいは8ビットタイマのコンペアマッチ信号によるA/D変換の開始の許可/禁止を選択します。

ADCR は、リセットまたはスタンバイモード時、H'7E にイニシャライズされます。

#### ビット7: トリガイネーブル (TRGE)

外部トリガ入力または8ビットタイマのコンペアマッチによるA/D変換の開始の許可/禁止を選択します。

ビット7 TRGE	説明
0	外部トリガ入力または8ビットタイマのコンペアマッチによるA/D変換の開始を禁止 (初期値)
1	外部トリガ端子 ( $\overline{\text{ADTRG}}$ ) の立下がりエッジまたは8ビットタイマのコンペアマッチでA/D変換を開始

なお、外部トリガ端子と8ビットタイマの選択は、8ビットタイマにより行います。詳細は「第10章 8ビットタイマ」を参照してください。

#### ビット6~1: リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

#### ビット0: リザーブビット

リザーブビットです。リード/ライト可能ですが、1に設定しないでください。



## 15.3 CPU とのインタフェース

ADDRA ~ ADDR D はそれぞれ 16 ビットのレジスタですが、CPU との間のデータバスは 8 ビット幅です。そのため、CPU からのアクセスは上位バイトは直接行われますが、下位バイトは 8 ビットのテンポラリレジスタ (TEMP) を介して行います。

ADDR からデータのリードは、次のように行われます。上位バイトのリードで上位バイトの値は CPU へ、下位バイトの値は TEMP へ転送されます。次に下位バイトのリードで TEMP の内容が CPU へ転送されます。

ADDR をリードする場合は、必ず上位バイト、下位バイトの順で行ってください。また、上位バイトのみのリードは可能ですが、下位バイトのみのリードでは内容は保証されませんので注意してください。

図 15.2 に、ADDR のアクセス時のデータの流れを示します。

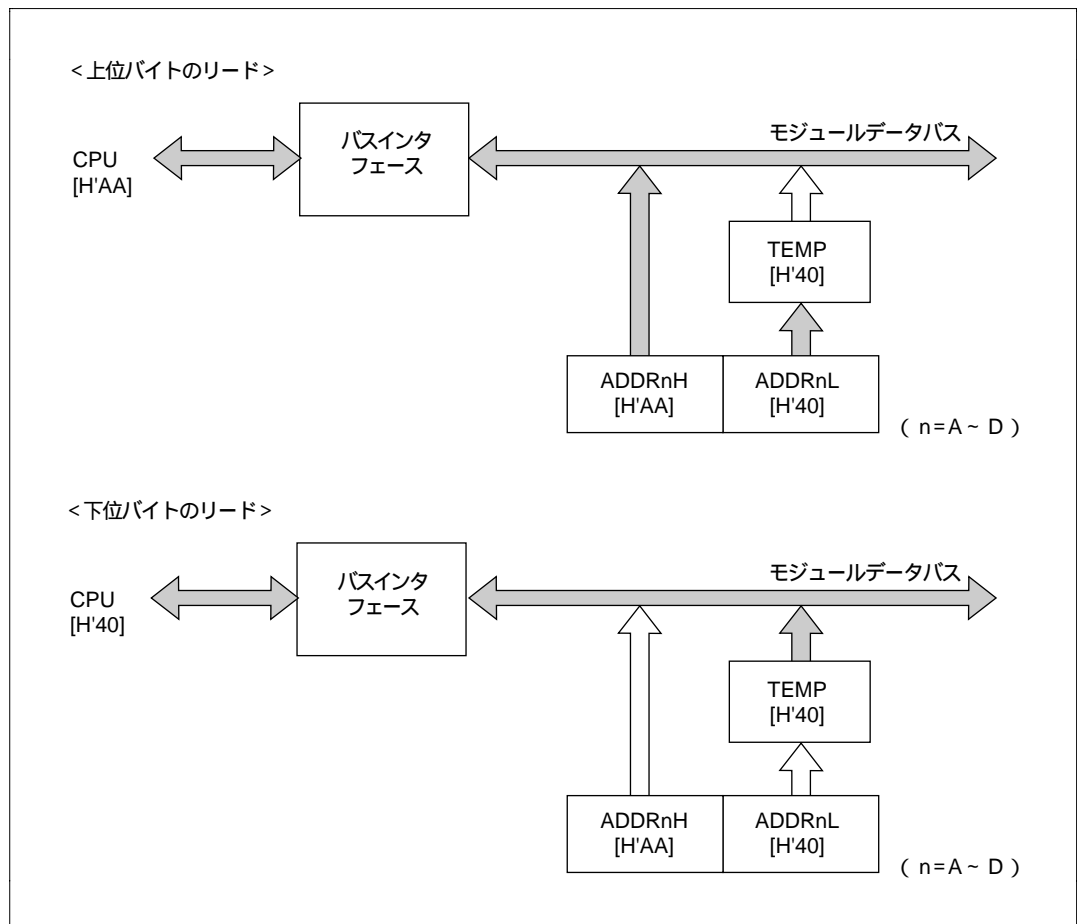


図 15.2 ADDR のアクセス動作 ( [H'AA40] リード時 )

## 15.4 動作説明

A/D変換器は逐次比較方式で動作し、10ビットの分解機能をもっています。単一モードとスキャンモードの各モードの動作についての説明をします。

### 15.4.1 単一モード (SCAN = 0)

単一モードは、1チャンネルのみのA/D変換を行う場合に選択します。ソフトウェアまたは外部トリガ入力によってADSTビットが1にセットされると、A/D変換を開始します。ADSTビットは、A/D変換中は1を保持しており、変換が終了すると自動的に0にクリアされます。

また、変換が終了すると、ADFフラグが1にセットされます。このとき、ADIEビットが1にセットされていると、ADI割込み要求が発生します。ADFフラグは、ADCSRをリードした後、0をライトするとクリアされます。

A/D変換中に、モードやアナログ入力チャンネルの切り換えを行う場合は、誤動作を避けるためにADCSRのADSTビットを0にクリアして、A/D変換を停止した状態で行ってください。

変更した後、ADSTビットを1にセットすると（モードおよびチャンネルの変更とADSTビットのセットは、同時に行うことができます）、再びA/D変換を開始します。

単一モードでチャンネル1 (AN<sub>1</sub>) が選択された場合の動作例を以下に示します。また、このときの動作タイミングを図15.3に示します。

- (1) 動作モードを単一モードに (SCAN = 0)、入力チャンネルをAN<sub>1</sub>に (CH2 = CH1 = 0、CH0 = 1)、A/D割込み要求許可 (ADIE = 1) に設定して、A/D変換を開始 (ADST = 1) します。
- (2) A/D変換が終了すると、A/D変換結果がADDRBに転送されます。同時に、ADF = 1、ADST = 0となり、A/D変換器は変換待機となります。
- (3) ADF = 1、ADIE = 1となっているため、ADI割込み要求が発生します。
- (4) A/D割込み処理ルーチンが開始されます。
- (5) ADCSRをリードした後、ADFに0をライトします。
- (6) A/D変換結果 (ADDRB) をリードして、処理します。
- (7) A/D割込み処理ルーチンの実行を終了します。この後、ADSTビットを1にセットするとA/D変換が開始され(2) ~ (7)を行います。

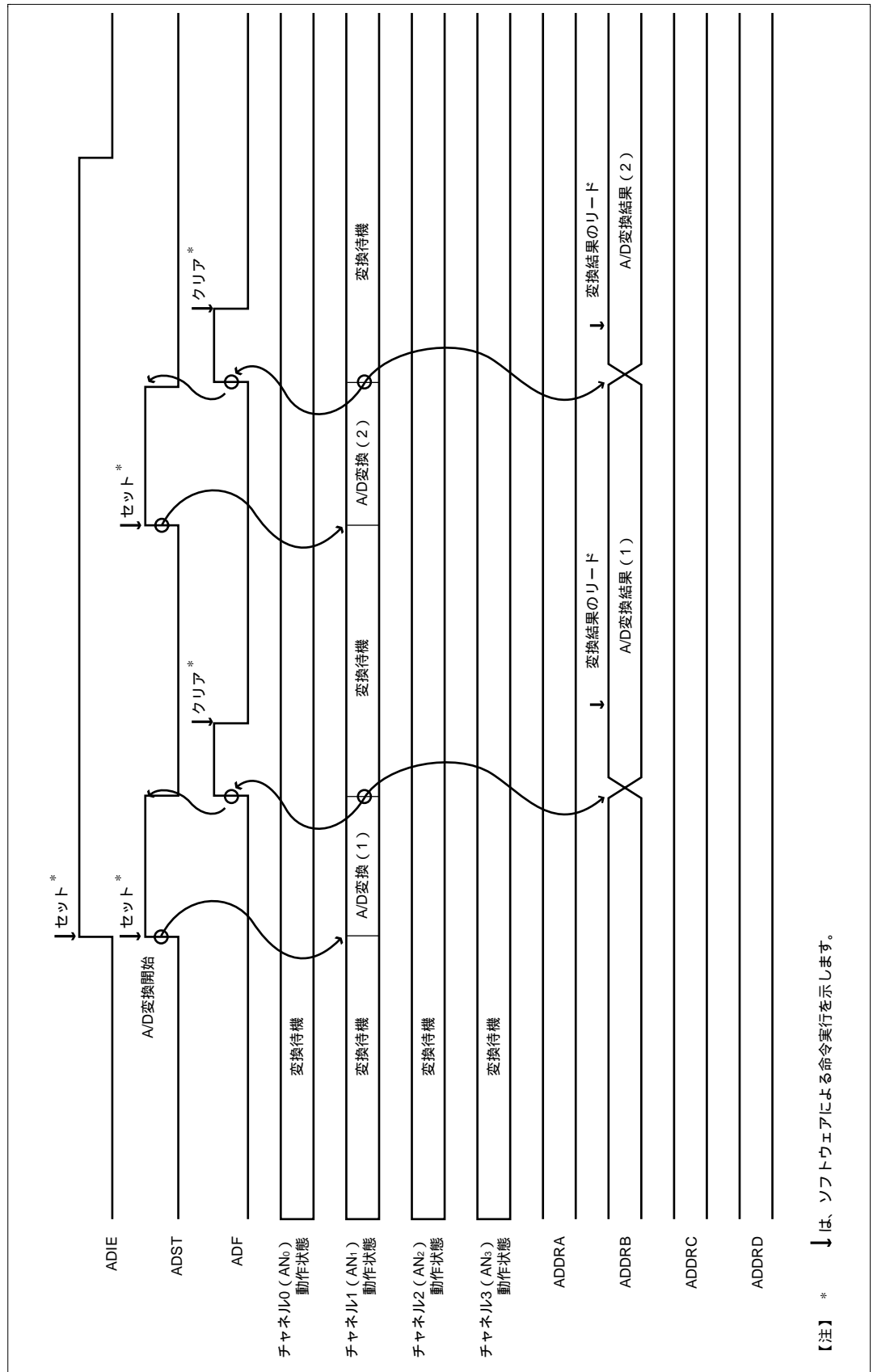


図 15.3 A/D 変換器の動作例 (単一モード、チャンネル1 選択時)

## 15.4.2 スキャンモード (SCAN = 1)

スキャンモードは、複数チャンネル (1チャンネルを含む) のアナログ入力を常にモニタするような応用に適しています。A/D 変換はソフトウェアまたは外部トリガ入力によって ADST ビットが 1 にセットされると、グループの第 1 チャンネル (CH2 = 0 のとき AN<sub>0</sub>、CH2 = 1 のとき AN<sub>4</sub>) から開始されます。複数のチャンネルが選択されている場合は、第 1 チャンネルの変換が終了した後、ただちに第 2 チャンネル (AN<sub>1</sub> または AN<sub>5</sub>) の A/D 変換を開始します。

A/D 変換は、ADST ビットが 0 にクリアされるまで、選択されたチャンネル内を連続して繰り返し行います。変換された結果は、各チャンネルに対応した ADDR に転送され保持されます。A/D 変換中に、モードやアナログ入力チャンネルの切換えを行う場合は、誤動作を避けるために ADCSR の ADST ビットを 0 にクリアして、A/D 変換を停止した状態で行ってください。変更した後、ADST ビットに 1 をセットすると (モードおよびチャンネルの変更と ADST ビットのセットは、同時に行うことができます)、第 1 チャンネルが選択され、再び A/D 変換を開始します。スキャンモードでグループ 0 の 3 チャンネル (AN<sub>0</sub> ~ AN<sub>2</sub>) を選択して A/D 変換を行う場合の動作例を以下に示します。また、このときの動作タイミングを図 15.4 に示します。

- (1) 動作モードをスキャンモードに (SCAN = 1)、スキャングループをグループ 0 に (CH2 = 0)、アナログ入力チャンネルを AN<sub>0</sub> ~ AN<sub>2</sub> (CH1 = 1、CH0 = 0) に設定して A/D 変換を開始 (ADST = 1) します。
- (2) 第 1 チャンネル (AN<sub>0</sub>) の A/D 変換が開始され、A/D 変換が終了すると、変換結果を ADDR<sub>A</sub> に転送します。次に第 2 チャンネル (AN<sub>1</sub>) が自動的に選択され、変換を開始します。
- (3) 同様に第 3 チャンネル (AN<sub>2</sub>) まで変換を行います。
- (4) 選択されたすべてのチャンネル (AN<sub>0</sub> ~ AN<sub>2</sub>) の変換が終了すると、ADF = 1 となり、再び第 1 チャンネル (AN<sub>0</sub>) を選択し、変換が行われます。このとき ADIE ビットが 1 にセットされていると、A/D 変換終了後、ADI 割込みを発生します。
- (5) ADST ビットが 1 にセットされている間は、(2) ~ (4) を繰り返します。ADST ビットを 0 にクリアすると A/D 変換が停止します。この後、ADST ビットを 1 にセットすると再び A/D 変換を開始し、第 1 チャンネル (AN<sub>0</sub>) から変換が行われます。

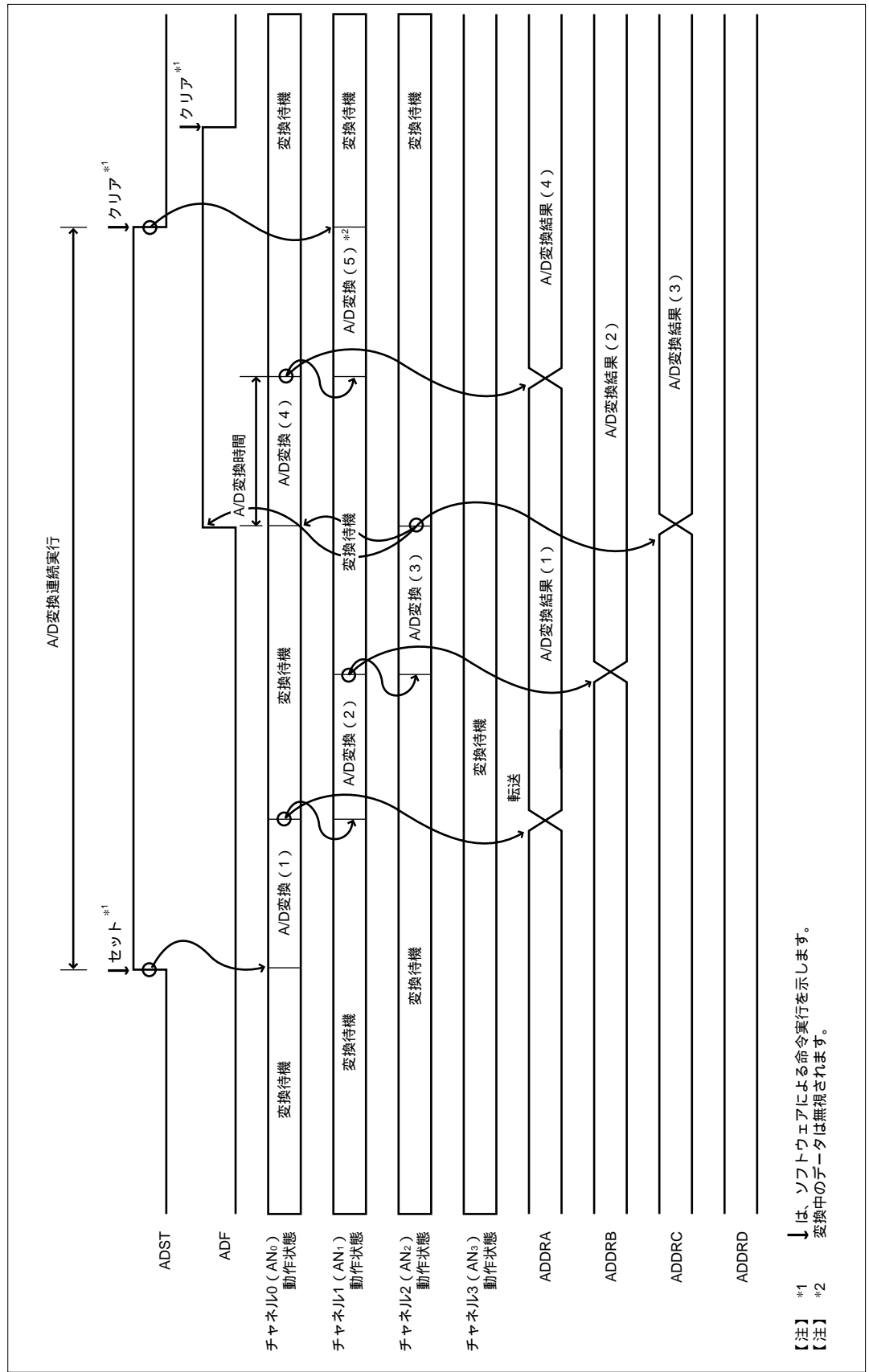


図 15.4 A/D 変換器の動作例 (スキャンモード AN<sub>0</sub> ~ AN<sub>2</sub> の 3 チャンネル選択時)

### 15.4.3 入力サンプリングとA/D変換時間

A/D変換器には、サンプル&ホールド回路が内蔵されています。A/D変換器は、ADSTビットが1にセットされてから $t_D$ 時間経過後、入力のサンプリングを行い、その後変換を開始します。A/D変換のタイミングを図15.5に示します。また、A/D変換時間を表15.4に示します。

A/D変換時間は、図15.5に示すように、 $t_D$ と入力サンプリング時間を含めた時間となります。ここで $t_D$ は、ADCSRへのライトタイミングにより決まり、一定値とはなりません。そのため、変換時間は表15.4に示す範囲で変化します。スキャンモードの変換時間は、表15.4に示す値が1回目の変換時間となりますが、2回目以降はCKS=0の場合は128ステート(固定)、CKS=1の場合は66ステート(固定)となります。

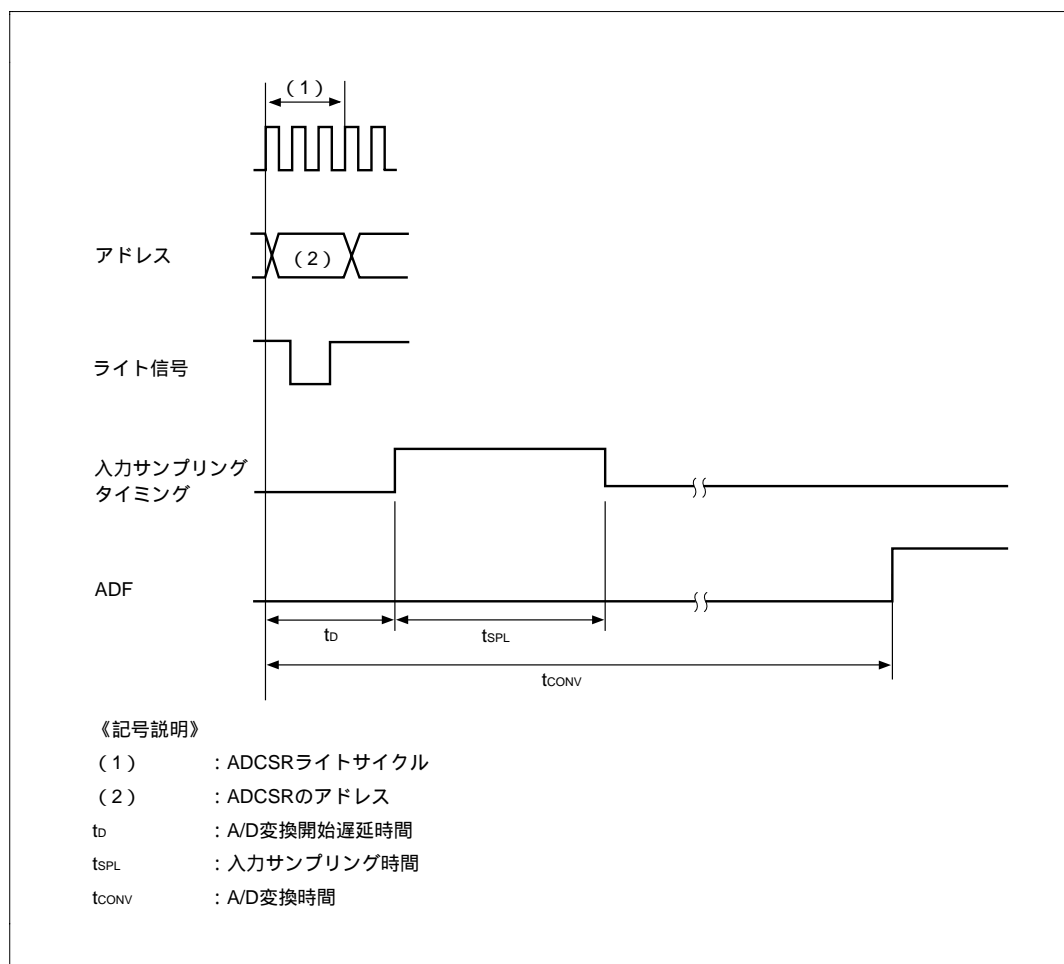


図15.5 A/D変換タイミング

表 15.4 A/D 変換時間 (単一モード)

	記号	CKS = 0			CKS = 1		
		min	typ	max	min	typ	max
A/D 変換開始遅延時間	$t_D$	6	—	9	4	—	5
入力サンプリング時間	$t_{SPL}$	—	31	—	—	15	—
A/D 変換時間	$t_{CONV}$	131	—	134	69	—	70

【注】 表中の数値の単位はステートです。

### 15.4.4 外部トリガ入力タイミング

A/D 変換は、外部トリガ入力により開始することも可能です。外部トリガ入力は、ADCR の TRGE ビットが 1 にセット、かつ 8 ビットタイマの ADTE ビットが 0 にクリアされているとき、 $\overline{ADTRG}$  端子から入力されます。 $\overline{ADTRG}$  入力端子の立下がりエッジで、ADCSR の ADST ビットが 1 にセットされ、A/D 変換が開始されます。その他の動作は、単一モード/スキャンモードによらず、ソフトウェアによって ADST ビットを 1 にセットした場合と同じです。このタイミングを図 15.6 に示します。

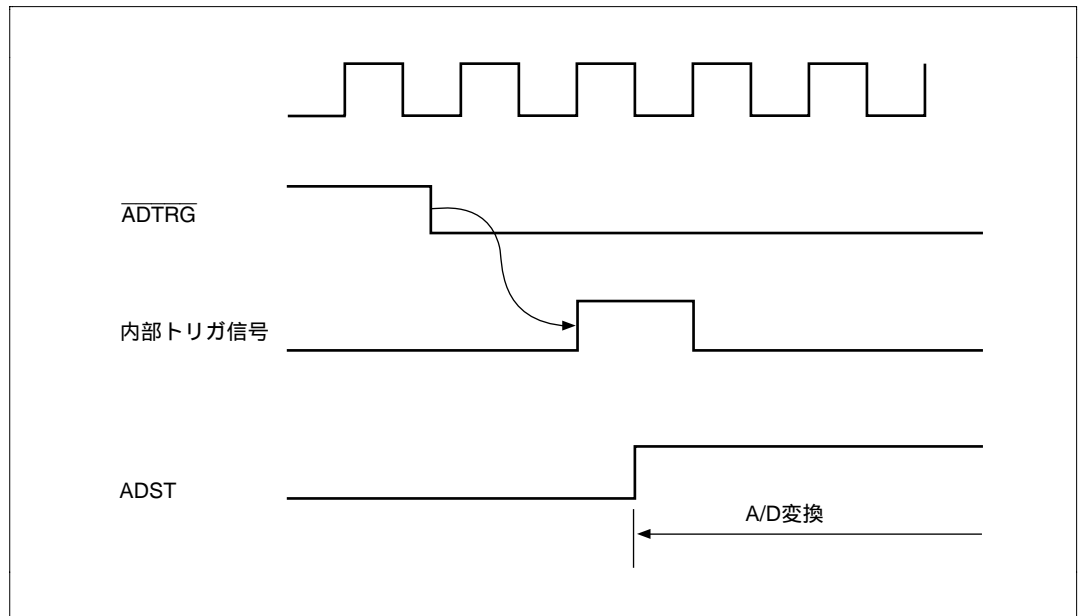


図 15.6 外部トリガ入力タイミング

## 15.5 割込み

A/D変換器は、A/D変換の終了により、A/D変換終了割込み（ADI）を発生します。ADI割込み要求は、ADCSRのADIEビットで許可/禁止することができます。ADI割込みは、DMACの起動要因とすることができます。このときは、CPUへは当該割込みは要求されません。

## 15.6 使用上の注意

A/D変換器を使用する際は、以下のことに注意してください。

### (1) アナログ入力電圧の範囲

A/D変換中、アナログ入力端子  $AN_n$  に印加する電圧は  $AV_{SS}$   $AN_n$   $V_{REF}$  の範囲としてください。

### (2) $AV_{CC}$ 、 $AV_{SS}$ と $V_{CC}$ 、 $V_{SS}$ の関係

$AV_{CC}$ 、 $AV_{SS}$ と $V_{CC}$ 、 $V_{SS}$ との関係は、 $AV_{SS} = V_{SS}$ とし、さらに、A/D変換器を使用しないときも、 $AV_{CC}$ 、 $AV_{SS}$ 端子を決してオープンにしないでください。

### (3) $V_{REF}$ の設定範囲

$V_{REF}$ 端子によるリファレンス電圧の設定範囲は  $V_{REF}$   $AV_{CC}$  にしてください。

### (4) ボード設計上の注意

ボード設計時には、デジタル回路とアナログ回路をできるだけ分離してレイアウトしてください。また、デジタル回路の信号配線とアナログ回路の信号配線を交差させたり、近接させるようなレイアウトは極力避けてください。誘導などにより、アナログ回路の誤動作や、A/D変換値に悪影響を及ぼします。なお、アナログ入力信号 ( $AN_0 \sim 7$ )、アナログ基準電源 ( $V_{REF}$ )、アナログ電源 ( $AV_{CC}$ )は、アナログ・グランド ( $AV_{SS}$ )で、デジタル回路を必ず分離してください。さらに、アナログ・グランド ( $AV_{SS}$ )は、ボード上の安定したデジタル・グランド ( $V_{SS}$ ) に一点接続してください。

### (5) ノイズ対策上の注意

アナログ入力端子 ( $AN_0 \sim 7$ )、アナログ基準電源 ( $V_{REF}$ )に、過大なサージなど異常電圧による破壊を防ぐために接続する保護回路は、図 15.7 に示すように  $AV_{CC}$  -  $AV_{SS}$  間に接続してください。また、 $AV_{CC}$ 、 $V_{REF}$  に接続するバイパス・コンデンサ、 $AN_0 \sim 7$  に接続するフィルタのコンデンサは、必ず  $AV_{SS}$  に接続してください。

なお、図 15.7 のようにフィルタ用のコンデンサを接続するとアナログ入力端子 ( $AN_0 \sim 7$ )の入力電流が平均化されるため、誤差を生じることがあります。また、スキャンモード等で頻りにA/D変換を行う場合、A/D変換器内部のサンプル&ホールド回路の容量に充放電される電流が入力インピーダンス ( $R_m$ ) を経由して入力される電流を上回るとアナログ入力端子の電圧に誤差を生じます。したがって回路定数の決定については、十分ご検討くださいますようお願いいたします。



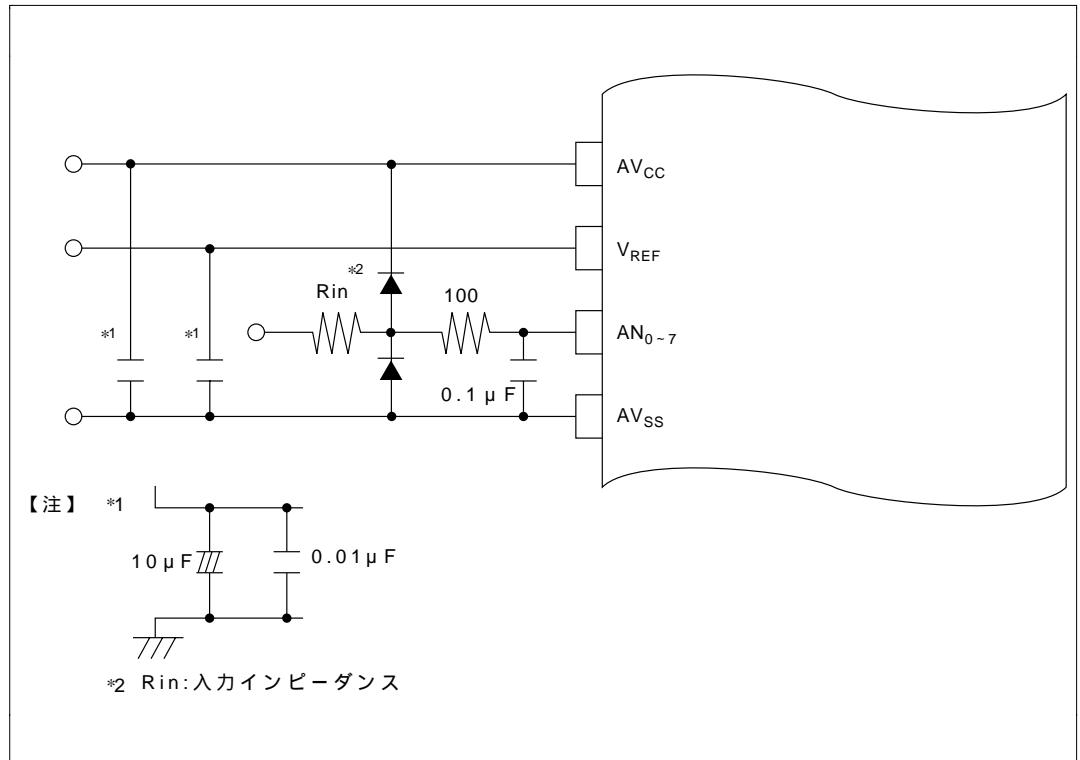


図 15.7 アナログ入力保護回路の例

表 15.5 アナログ端子の規格

項目	min	max	単位
アナログ入力容量	—	20	pF
許容信号源インピーダンス	—	10*	k

【注】 \* 変換時間 134 ステート、 $V_{CC} = 4.0 \sim 5.5V$ 、13MHz の場合。詳細は「21. 電気的特性」を参照してください。

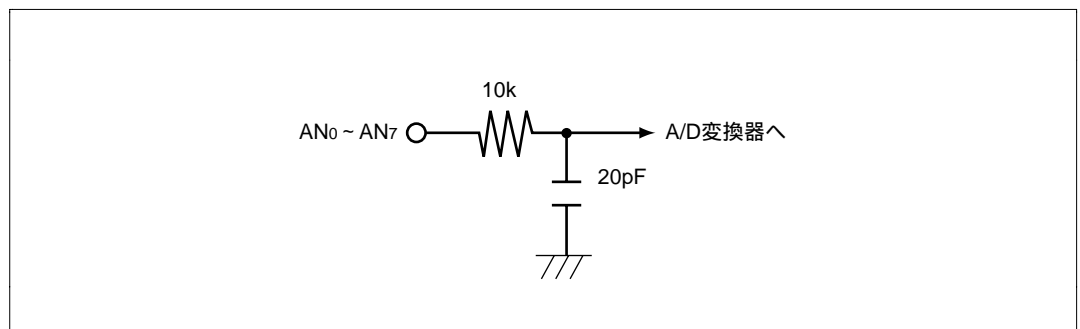


図 15.8 アナログ入力端子等価回路

【注】 表 15.5 を除く数値はいずれも参考値

## (6) A/D 変換精度の定義：

以下に、本 LSI の A/D 変換精度の定義を示します。

- ・ 分解能.....A/D 変換器のデジタル出力コード数
- ・ オフセット誤差.....デジタル出力が最小電圧値 0000000000 から 0000000001 に変化するときのアナログ入力電圧値の理想 A/D 変換特性からの偏差（図 15.10）。
- ・ フルスケール誤差...デジタル出力が 1111111110 から 1111111111 に変化する時のアナログ入力電圧値の理想 A/D 変換特性からの偏差（図 15.10）。
- ・ 量子化誤差.....A/D 変換器が本質的に有する偏差であり、1/2LSB で与えられる（図 15.9）。
- ・ 非直線性誤差.....ゼロ電圧からフルスケール電圧までの間の理想 A/D 変換特性からの誤差。ただし、オフセット誤差、フルスケール誤差、量子化誤差を含まない。
- ・ 絶対精度.....デジタル値とアナログ入力値との偏差。オフセット誤差、フルスケール誤差、量子化誤差および非直線誤差を含む。

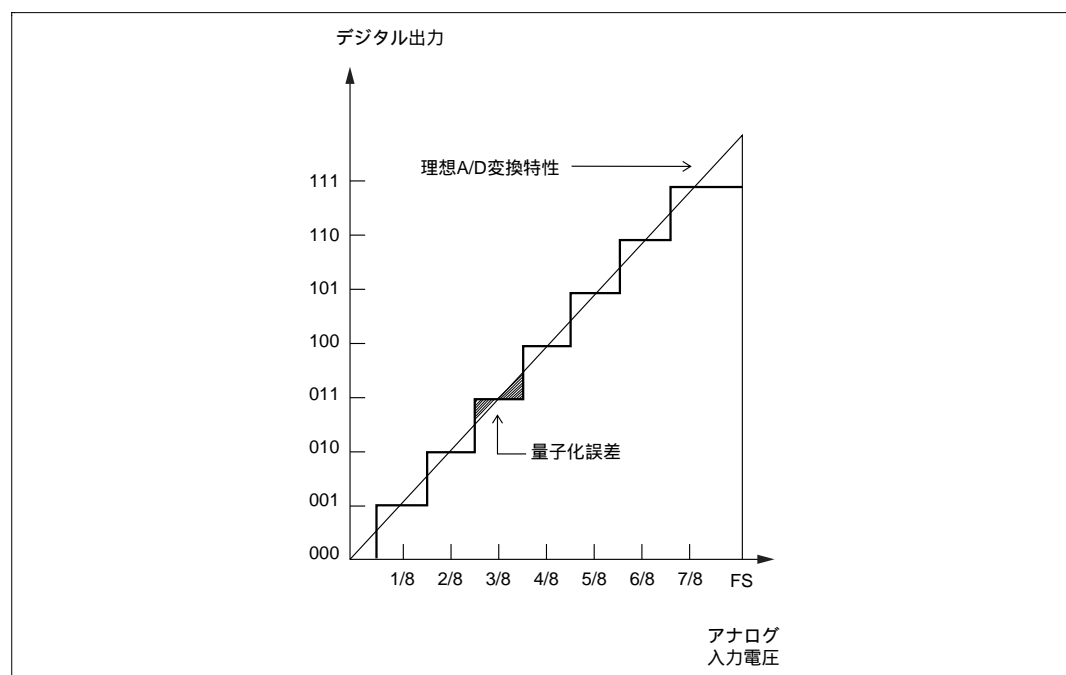


図 15.9 A/D 変換精度の定義 (1)

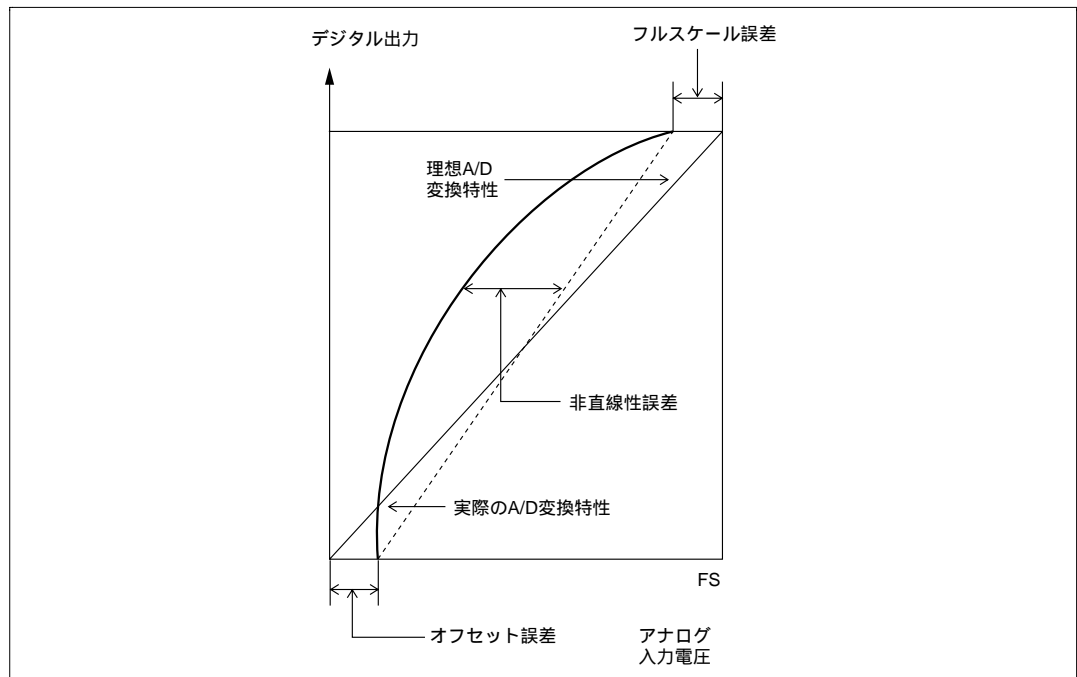


図 15.10 A/D 変換精度の定義 (2)

## (7) 許容信号源インピーダンスについて

本 LSI のアナログ入力は、信号源インピーダンスが  $10k\ \Omega$  以下の入力信号に対し、変換精度が保証される設計となっております。これは A/D 変換器のサンプル&ホールド回路の入力容量をサンプリング時間内に充電するために設けている規格で、センサの出力インピーダンスが  $10k\ \Omega$  を越える場合充電不足が生じ、A/D 変換精度が保証できなくなる場合があります。単一モードで変換を行う場合で外部に大容量を設けている場合、入力の負荷は実質的に内部入力抵抗の  $10k\ \Omega$  だけになりますので信号源インピーダンスは不問となります。ただし、この場合ローパスフィルターとなりますので、微分係数の大きなアナログ信号（例えば電圧の変動率が  $5mV/\mu s$  以上）には追従できない場合があります（図 15.11）。高速のアナログ信号を変換する場合や、スキャンモードで変換を行う場合には、低インピーダンスのバッファを入れてください。

## (8) 絶対精度への影響について

容量を付加するとにより、GND とのカップリングを受けることとなりますので、GND にノイズがあると絶対精度が悪化する可能性があります。必ず  $AV_{SS}$  等の電氣的に安定な GND に接続してください。またフィルター回路が実装基板上でデジタル信号と干渉したり、アンテナとならないように注意が必要です。

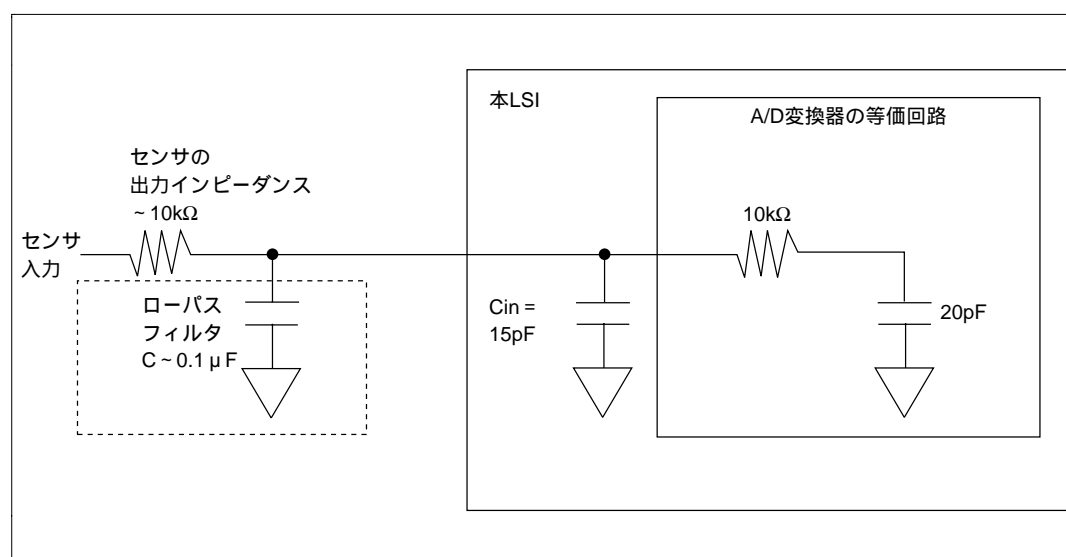


図 15.11 アナログ入力回路の例



---

# 16. D/A 変換器

---

## 第 16 章 目次

16.1	概要.....	637
	16.1.1 特長.....	637
	16.1.2 ブロック図.....	637
	16.1.3 端子構成.....	638
	16.1.4 レジスタ構成.....	638
16.2	各レジスタの説明.....	639
	16.2.1 D/A データレジスタ 0、1 (DADR0、1).....	639
	16.2.2 D/A コントロールレジスタ (DACR).....	640
	16.2.3 D/A スタンバイコントロールレジスタ (DASTCR).....	642
16.3	動作説明.....	643
16.4	D/A 出力制御.....	644



## 16.1 概要

本LSIには2チャンネルのD/A変換器が内蔵されています。

### 16.1.1 特長

D/A変換器の特長を以下に示します。

8ビットの分解能

2チャンネル出力

変換時間最大10  $\mu$ s (負荷容量20pF時)

出力電圧0V  $\sim$   $V_{REF}$

ソフトウェアスタンバイ時のD/A出力保持機能

### 16.1.2 ブロック図

D/A変換器のブロック図を図16.1に示します。

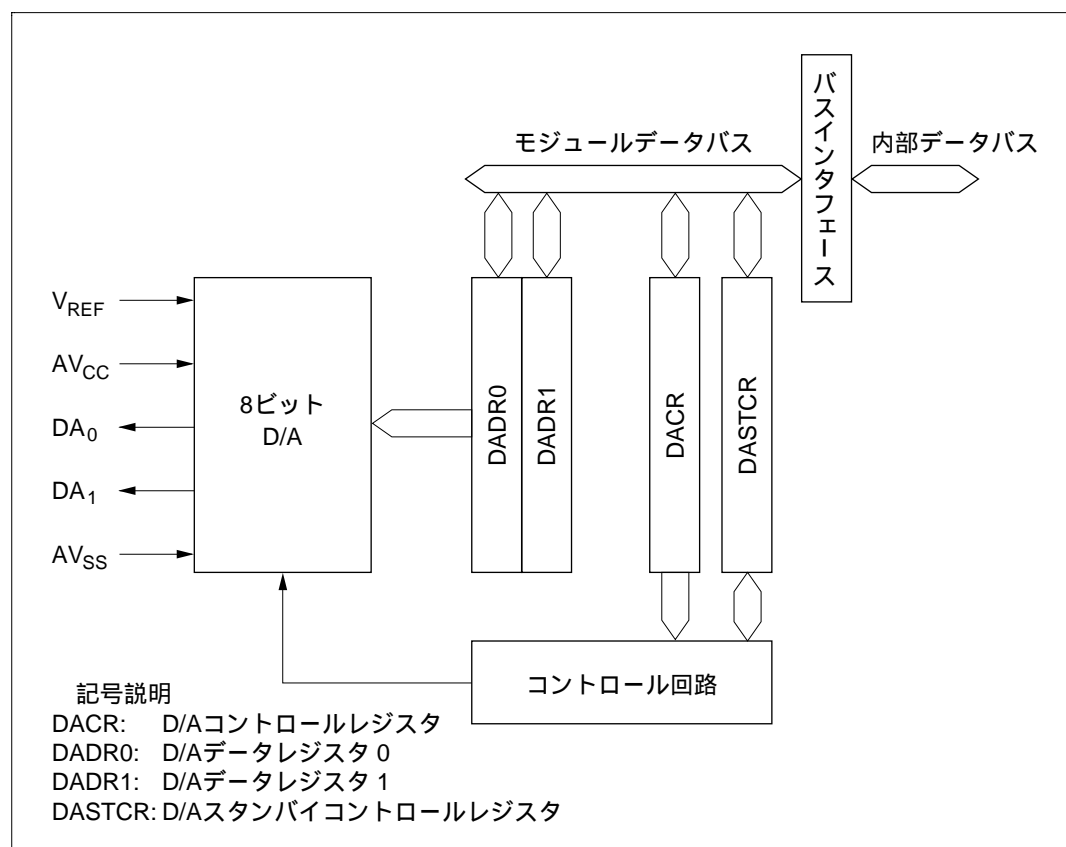


図 16.1 D/A 変換器のブロック図



### 16.1.3 端子構成

D/A 変換器で使用する入出力端子を表 16.1 に示します。

表 16.1 端子構成

端子名	略 称	入出力	機 能
アナログ電源端子	$AV_{CC}$	入力	アナログ部の電源および基準電圧
アナロググランド端子	$AV_{SS}$	入力	アナログ部のグランドおよび基準電圧
アナログ出力端子 0	$DA_0$	出力	チャンネル 0 のアナログ出力
アナログ出力端子 1	$DA_1$	出力	チャンネル 1 のアナログ出力
リファレンス電圧端子	$V_{REF}$	入力	アナログ部の基準電圧

### 16.1.4 レジスタ構成

D/A 変換器のレジスタ構成を表 16.2 に示します。

表 16.2 レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H' FFF9C	D/A データレジスタ 0	DADR0	R/W	H'00
H' FFF9D	D/A データレジスタ 1	DADR1	R/W	H'00
H'FFF9E	D/A コントロールレジスタ	DACR	R/W	H'1F
H'EE01A	D/A スタンバイコントロールレジスタ	DASTCR	R/W	H'FE

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示します。

## 16.2 各レジスタの説明

### 16.2.1 D/A データレジスタ 0、1 (DADR0、1)

ビット:	7	6	5	4	3	2	1	0
初期値:	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

D/A データレジスタ 0、1 (DADR0、1) は、変換を行うデータを格納するリード/ライト可能な 8 ビットのレジスタです。

アナログ出力を許可すると、DADR の値が常に変換され、アナログ出力端子に出力されます。

DADR は、リセットまたはスタンバイモード時に、H'00 にイニシャライズされます。

D/A スタンバイコントロールレジスタ (DASTCR) の DASTE ビットが 1 の場合、ソフトウェアスタンバイモードではイニシャライズされません。

## 16.2.2 D/A コントロールレジスタ (DACR)

ビット:	7	6	5	4	3	2	1	0
	DAOE1	DAOE0	DAE	—	—	—	—	—
初期値:	0	0	0	1	1	1	1	1
R/W :	R/W	R/W	R/W	—	—	—	—	—

D/Aイネーブル  
 D/A変換を制御するビットです。

D/Aアウトプットイネーブル0  
 D/A変換とアナログ出力を制御するビットです。

D/Aアウトプットイネーブル1 (DAOE1)  
 D/A変換とアナログ出力を制御するビットです。

DACR は、8 ビットのリード/ライト可能なレジスタで、D/A 変換器の動作を制御します。

DACR は、リセットまたはスタンバイモード時に、H'1F にイニシャライズされます。

DASTCR の DASTE ビットが 1 の場合、ソフトウェアスタンバイモードではイニシャライズされません。

### ビット 7 : D/A アウトプットイネーブル 1 (DAOE1)

D/A 変換とアナログ出力を制御します。

ビット 7	説明
DAOE1	
0	アナログ出力 DA <sub>1</sub> を禁止
1	チャンネル 1 の D/A 変換を許可。アナログ出力 DA <sub>1</sub> を許可

### ビット 6 : D/A アウトプットイネーブル 0 (DAOE0)

D/A 変換とアナログ出力を制御します。

ビット 6	説明
DAOE0	
0	アナログ出力 DA <sub>0</sub> を禁止
1	チャンネル 0 の D/A 変換を許可。アナログ出力 DA <sub>0</sub> を許可

## ビット5 : D/A イネーブル (DAE)

DAOE0、DAOE1 とともに、D/A 変換を制御します。DAE ビットが0 にクリアされているときチャンネル0、1 のD/A 変換は独立に制御され、DAE ビットが1 にセットされているときチャンネル0、1 のD/A 変換は一括して制御されます。

変換結果を出力するか否かは、DAOE0、DAOE1 により、常に独立に制御されます。

ビット7	ビット6	ビット5	説明
DAOE1	DAOE0	DAE	
0	0	-	チャンネル0、1 のD/A 変換を禁止
	1	0	チャンネル0 のD/A 変換を許可 チャンネル1 のD/A 変換を禁止
		1	チャンネル0、1 のD/A 変換を許可
1	0	0	チャンネル0 のD/A 変換を禁止 チャンネル1 のD/A 変換を許可
		1	チャンネル0、1 のD/A 変換を許可
	1	-	チャンネル0、1 のD/A 変換を許可

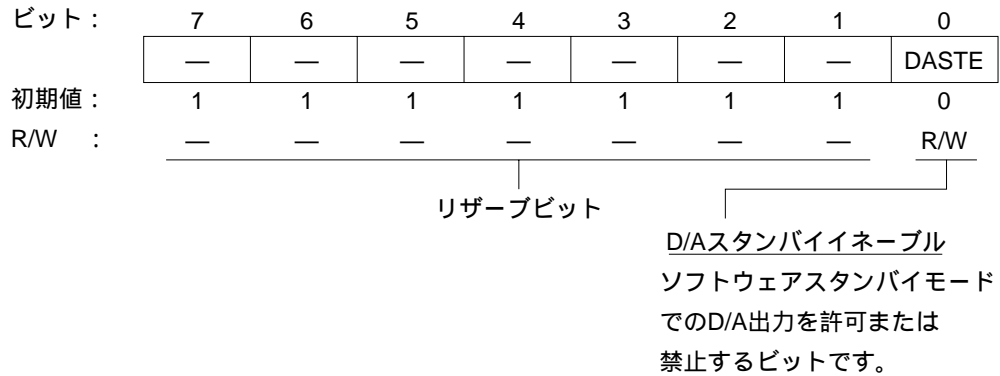
DAE ビットを1 にセットすると、DACR のDAOE0、1 ビット、ADCSR のADST ビットが0 にクリアされていても、アナログ電源電流はA/D、D/A 変換中と同等になります。

## ビット4~0 : リザーブビット

リザーブビットです。リードすると常に1 が読み出されます。ライトは無効です。

### 16.2.3 D/A スタンバイコントロールレジスタ (DASTCR)

DASTCR は 8 ビットのリード/ライト可能なレジスタで、ソフトウェアスタンバイモードでの D/A の出力を許可または禁止します。



DASTCR はリセット、またはハードウェアスタンバイモード時に、HFE にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

#### ビット7~1: リザーブビット

リザーブビットです。リードすると常に 1 が読み出されます。ライトは無効です。

#### ビット0: D/A スタンバイイネーブル (DASTE)

ソフトウェアスタンバイモードでの D/A 出力を許可または禁止します。

ビット0	説明
DASTE	
0	ソフトウェアスタンバイモードでの D/A 出力を禁止 (初期値)
1	ソフトウェアスタンバイモードでの D/A 出力を許可

## 16.3 動作説明

D/A 変換器は、2 チャンネルの D/A 変換回路を内蔵し、それぞれ独立に変換を行うことができます。

DACR によって D/A 変換が許可されている期間は常に D/A 変換が行われています。DADR0、1 を書き換えるとただちに、新しいデータが変換されます。DAOE0、1 ビットを 1 にセットすることにより、変換結果が出力されます。

チャンネル 0 の D/A 変換を行う場合の動作例を示します。動作タイミングを図 16.2 に示します。

- (1) DADR0 に変換データをライトします。
- (2) DACR の DAOE0 ビットを 1 にセットします。D/A 変換が開始され、DA0 端子が出力端子になります。変換時間経過後に変換結果が出力されます。

出力値は  $\frac{\text{DADR の内容}}{256} \times V_{\text{REF}}$  です。

次に DADR0 を書き換えるか、DAOE0 ビットを 0 にクリアするまでこの変換結果が出力され続けます。

- (3) DADR0 を書き換えるとただちに变換が開始されます。変換時間経過後に変換結果が出力されます。
- (4) DAOE0 ビットを 0 にクリアすると、DA0 端子は入力端子になります。

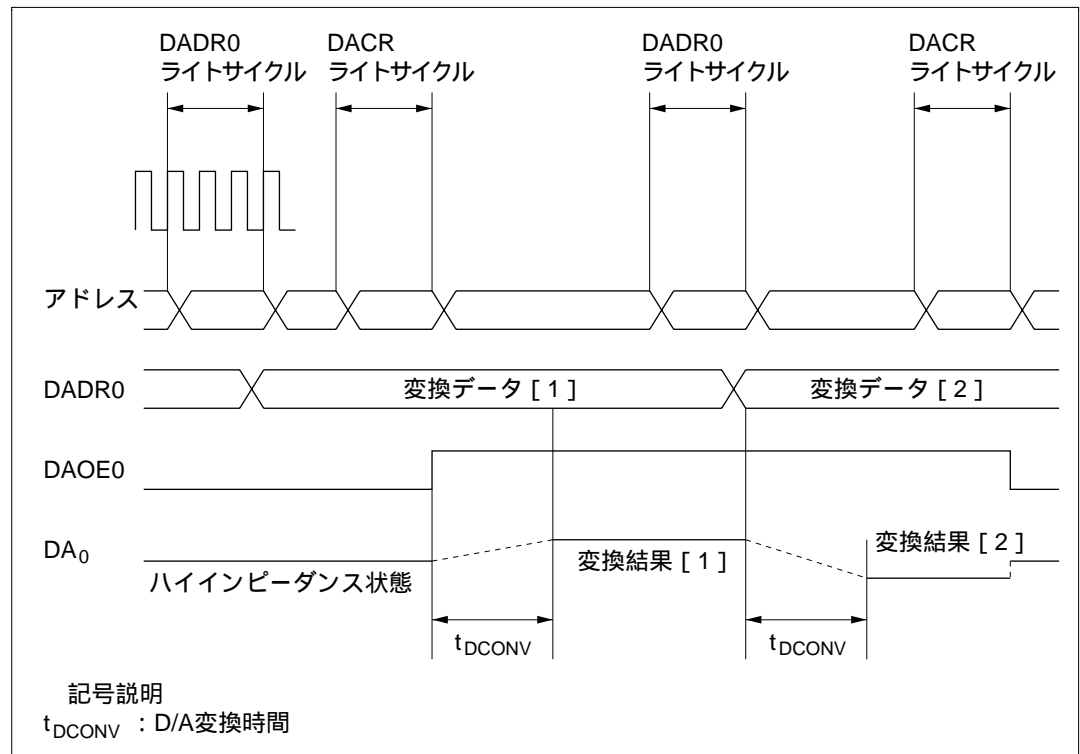


図 16.2 D/A 変換器の動作例

## 16.4 D/A 出力制御

本 LSI は、ソフトウェアスタンバイモードで D/A 変換器の出力を許可または禁止することができます。

DASTCR の DASTE ビットを 1 にセットすると、ソフトウェアスタンバイモードにおいても D/A 変換器の出力が許可されます。このとき、D/A 変換器のレジスタはソフトウェアスタンバイモードに遷移する直前の値を保持します。

なお、ソフトウェアスタンバイモードで D/A 出力を許可した場合、リファレンス電源電流は動作時と同じとなります。

---

# 17. RAM

---

## 第 17 章 目次

17.1	概要.....	647
	17.1.1	ブロック図.....647
	17.1.2	レジスタ構成.....648
17.2	システムコントロールレジスタ (SYSCR) .....	649
17.3	動作説明 .....	650





## 17.1 概要

H8/3068F は 16k バイトのスタティック RAM を内蔵しています。RAM は CPU と 16 ビット幅のデータバスで接続されており、アクセスはバイトデータ、ワードデータにかかわらず 2 ステートで行われます。したがって、データの高速度転送が可能です。

H8/3068F の内蔵 RAM は、モード 1、2、7 のとき H'FBF20 ~ H'FFF1F に、モード 3、4、5 のとき H'FFBF20 ~ H'FFFF1F に、モード 6 のとき H'E720 ~ H'FF1F に割り当てられています。

システムコントロールレジスタ (SYSCR) の RAM イネーブル (RAME) ビットにより内蔵 RAM 有効 / 無効の制御を行います。

### 17.1.1 ブロック図

RAM のブロック図を図 17.1 に示します。

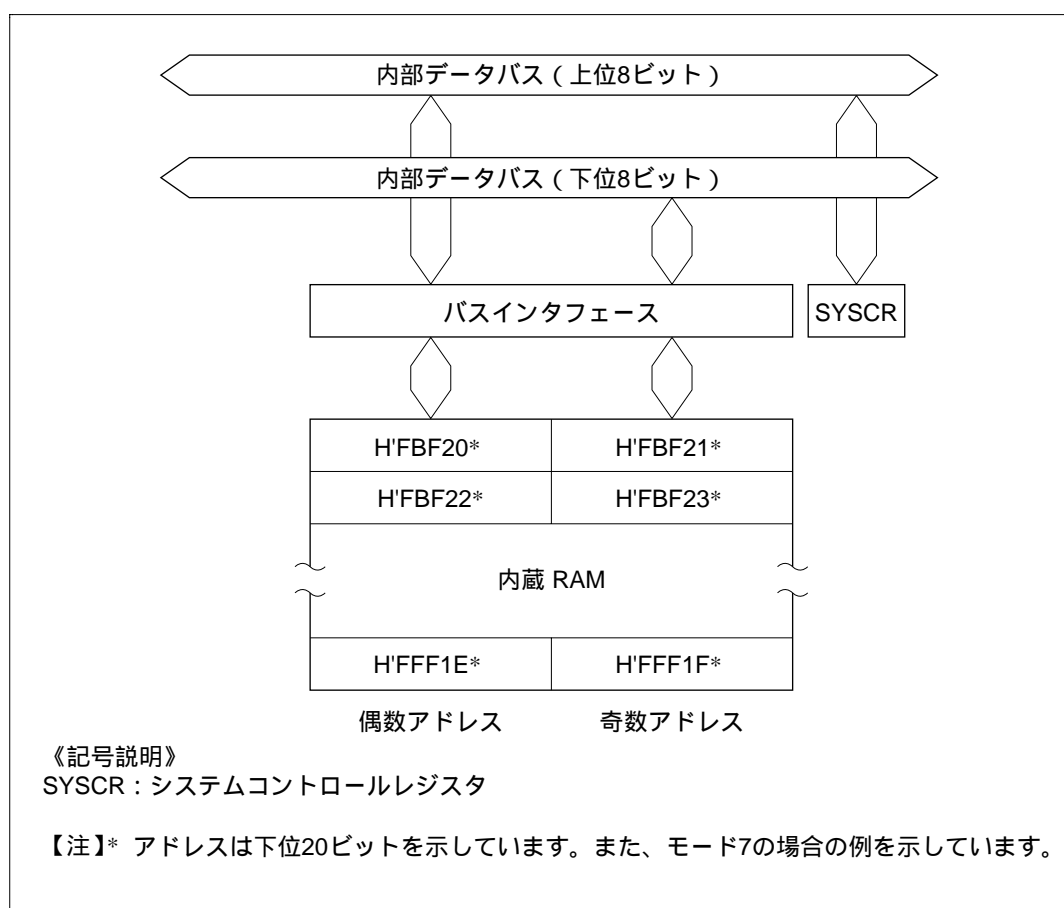


図 17.1 RAM のブロック図

## 17.1.2 レジスタ構成

内蔵RAMは、SYSCRで制御されます。

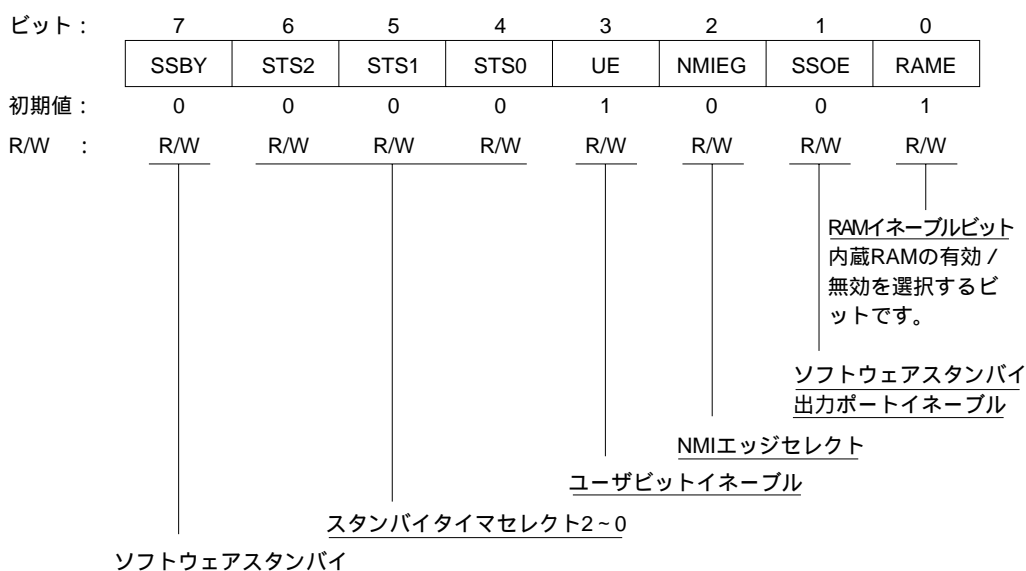
SYSCRのアドレスと初期値を表17.1に示します。

表 17.1 レジスタ構成

アドレス*	名 称	略 称	R/W	初期値
H'EE012	システムコントロールレジスタ	SYSCR	R/W	H'09

【注】 \* アドバンスモード時のアドレス下位20ビットを示します。

## 17.2 システムコントロールレジスタ (SYSCR)



SYSCR は、内蔵 RAM へのアクセスを許可 / 禁止するレジスタです。内蔵 RAM は SYSCR の RAME ビットにより有効 / 無効が選択されます。なお、SYSCR のその他のビットについての詳細は「3.3 システムコントロールレジスタ」を参照してください。

### ビット0 : RAM イネーブル (RAME)

内蔵 RAM の有効または無効を選択します。RAME ビットは  $\overline{\text{RES}}$  端子の立上がりエッジでイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

ビット0	説明
RAME	
0	内蔵 RAM 無効
1	内蔵 RAM 有効 (初期値)

## 17.3 動作説明

RAME ビットを 1 にセットすると内蔵 RAM が有効になります。モード 1、2、7 のとき H'FBF20 ~ H'FFF1F を、モード 3、4、5 のとき H'FFBF20 ~ H'FFFF1F を、モード 6 のとき H'7F20 ~ H'FF1F をアクセスすると内蔵 RAM がアクセスされます。また、モード 1 ~ 5 (拡張モード) では RAME ビットが 0 にクリアされているときは、外部アドレス空間がアクセスされます。モード 6、7 (シングルチップモード) では、RAME ビットが 0 にクリアされているときは、内蔵 RAM はアクセスされません。リードすると常に H'FF がリードされ、ライトは無効です。

RAM は CPU と内部 16 ビットデータバスで接続されており、ワード単位のリード/ライトが可能です。また、バイト単位のリード/ライトも可能です。

バイトデータは、データバス上位 8 ビットを使い 2 ステートでアクセスされ、また、偶数番地から始まるワードデータはデータバス 16 ビットを使い 2 ステートでアクセスできます。

---

# 18. フラッシュメモリ

---

## 第18章 目次

18.1	概要	653
18.2	特長	654
18.2.1	ブロック図	655
18.2.2	端子構成	655
18.2.3	レジスタ構成	656
18.3	レジスタの説明	657
18.3.1	フラッシュメモリコントロールレジスタ1 (FLMCR1)	657
18.3.2	フラッシュメモリコントロールレジスタ2 (FLMCR2)	660
18.3.3	消去ブロック指定レジスタ1 (EBR1)	661
18.3.4	消去ブロック指定レジスタ2 (EBR2)	662
18.3.5	RAMコントロールレジスタ (RAMCR)	663
18.4	動作の概要	665
18.4.1	モード遷移図	665
18.4.2	オンボードプログラミングモード	666
18.4.3	RAMによるフラッシュメモリのエミュレーション	668
18.4.4	ブロック分割法	669
18.5	オンボードプログラミングモード	670
18.5.1	ブートモード	671
18.5.2	ユーザプログラムモード	676
18.6	フラッシュメモリの書き込み/消去	678
18.6.1	プログラムモード	679
18.6.2	プログラムベリファイモード	680
18.6.3	イレースモード	685
18.6.4	イレースベリファイモード	685
18.7	プロテクト	687
18.7.1	ハードウェアプロテクト	687
18.7.2	ソフトウェアプロテクト	688

18.7.3 エラープロテクト.....	689
18.8 RAM によるフラッシュメモリのエミュレーション.....	691
18.9 NMI 入力禁止の条件.....	694
18.10 フラッシュメモリの PROM モード.....	695
18.10.1 ソケットアダプタとメモリマップ.....	695
18.10.2 PROM モード使用時の注意事項.....	695
18.11 フラッシュメモリの書き込み / 消去時の注意.....	697

## 18.1 概要

H8/3068F-ZTAT は 384k バイトのフラッシュメモリを内蔵しています。フラッシュメモリは、CPU と 16 ビット幅のデータバスで接続されており、バイトデータ、ワードデータにかかわらず、2 ステートでアクセスできます。したがって、データの高速度転送が可能です。

内蔵 ROM の有効または無効の設定は表 18.1 に示すように、モード端子 (MD<sub>2</sub> ~ MD<sub>0</sub>) により行います。

なお、H8/3068F-ZTAT は、専用の PROM ライタを用いて消去・書き込みができるほか、オンボードでの消去・書き替えが可能です。

表 18.1 動作モードと ROM

モード名	モード端子			内蔵 ROM
	MD <sub>2</sub>	MD <sub>1</sub>	MD <sub>0</sub>	
モード 1 (内蔵 ROM 無効拡張 1M バイトモード)	0	0	1	無効 (外部アドレス 空間)
モード 2 (内蔵 ROM 無効拡張 1M バイトモード)	0	1	0	
モード 3 (内蔵 ROM 無効拡張 16M バイトモード)	0	1	1	
モード 4 (内蔵 ROM 無効拡張 16M バイトモード)	1	0	0	
モード 5 (内蔵 ROM 有効拡張 16M バイトモード)	1	0	1	有効
モード 6 (シングルチップノーマルモード)	1	1	0	
モード 7 (シングルチップアドバンスモード)	1	1	1	



## 18.2 特長

H8/3068F-ZTAT は 384k バイトのフラッシュメモリを内蔵しています。フラッシュメモリの特長を以下に示します。

フラッシュメモリの 4 種類の動作モード

- ・プログラムモード
- ・イレースモード
- ・プログラムベリファイモード
- ・イレースベリファイモード

書き込み / 消去方式

書き込みは 128 バイト同時書き込みを行います。消去はブロック分割消去 (1 ブロック単位) で行います。全面消去を行う場合は、各ブロック単位に順次行ってください。ブロック分割消去では 4k バイト、32k バイト、64k バイトのブロック単位で任意に設定することができます。

書き込み / 消去時間

フラッシュメモリの書き込み時間は、128 バイト同時書き込みにて 10ms (typ.)、1 バイトあたり換算にて約 80  $\mu$ s (typ.)、消去時間は、100ms (typ.) です。

書き替え回数

フラッシュメモリの書き替えは、100 回まで可能です。

オンボードプログラミングモード

オンボードでフラッシュメモリの書き込み / 消去 / ベリファイを行う 2 種類のモードがあります。

- ・ブートモード
- ・ユーザプログラムモード

ビットレート自動合わせ込み

ブートモードでデータ転送時、ホストの転送ビットレートと本 LSI のビットレートとを自動で合わせることができます。

RAM によるフラッシュメモリのエミュレーション機能

フラッシュメモリと RAM の一部を重ね合わせることで、フラッシュメモリの書き替えをリアルタイムにエミュレートすることができます。

プロテクトモード

ソフトウェアプロテクトモード、ハードウェアプロテクトモードとエラープロテクトの 3 種類のモードがあり、フラッシュメモリの書き込み / 消去 / ベリファイのプロテクト状態を設定することができます。

PROM モード

フラッシュメモリの書き込み / 消去可能なモードとして、オンボードプログラミングモード以外に PROM ライタを用いた PROM モードがあります。

## 18.2.1 ブロック図

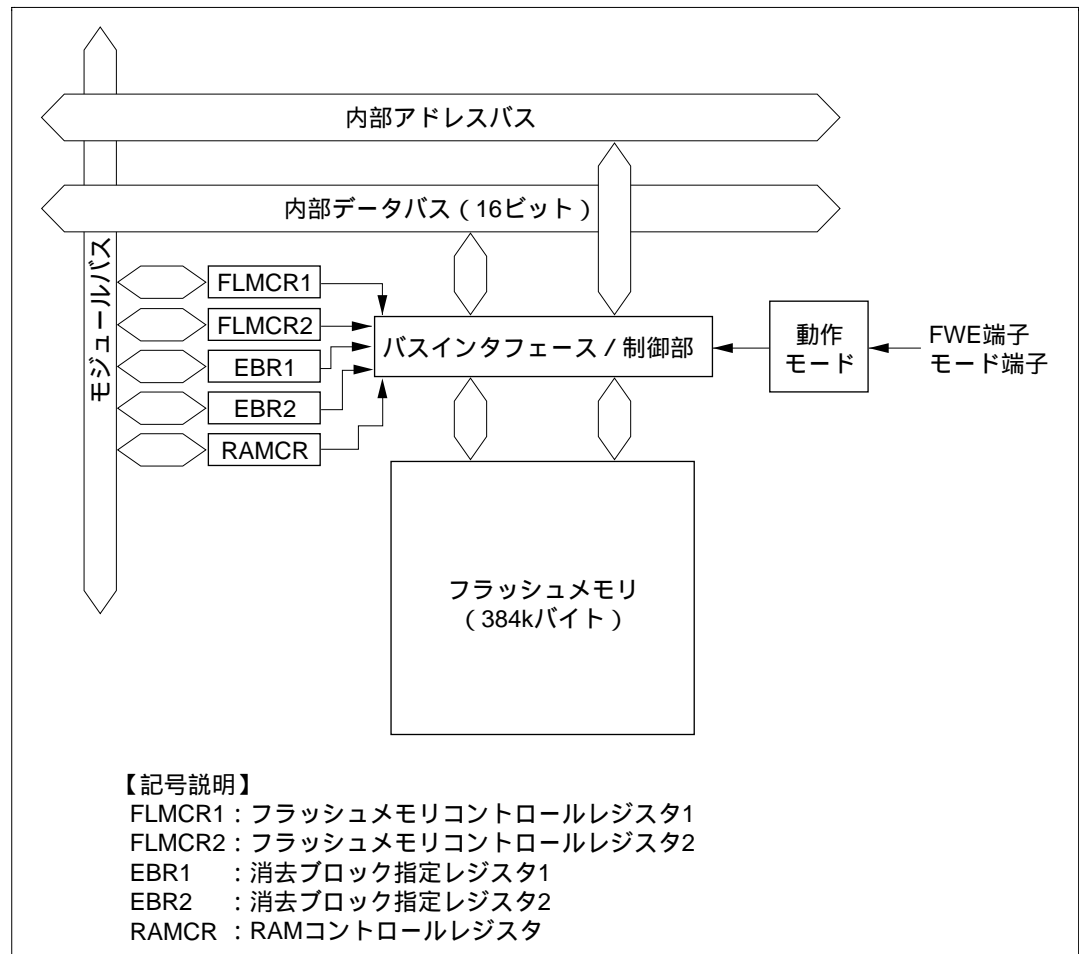


図 18.1 フラッシュメモリのブロック図

## 18.2.2 端子構成

フラッシュメモリは表 18.3 に示す端子により制御されます。

表 18.3 端子構成

端子名	略称	入出力	機能
リセット	RES	入力	リセット
フラッシュライトイネーブル	FWE	入力	フラッシュの書き込み / 消去をハードウェアプロテクト
モード2	MD <sub>2</sub>	入力	本 LSI の動作モードを設定
モード1	MD <sub>1</sub>	入力	本 LSI の動作モードを設定
モード0	MD <sub>0</sub>	入力	本 LSI の動作モードを設定
トランスミットデータ	TxD <sub>1</sub>	出力	シリアル送信データ出力
レシーブデータ	RxD <sub>1</sub>	入力	シリアル受信データ入力

### 18.2.3 レジスタ構成

内蔵フラッシュメモリが有効のときのフラッシュメモリをコントロールするレジスタを表 18.4 に示します。

表 18.4 レジスタ構成

名 称	略 称	R/W	初期値	アドレス* <sup>1</sup>
フラッシュメモリコントロールレジスタ1	FLMCR1	R/W	H'00* <sup>2</sup>	H'EE030
フラッシュメモリコントロールレジスタ2	FLMCR2	R	H'00	H'EE031
消去ブロック指定レジスタ1	EBR1	R/W	H'00	H'EE032
消去ブロック指定レジスタ2	EBR2	R/W	H'00	H'EE033
RAM コントロールレジスタ	RAMCR	R/W	H'F0	H'EE077

【注】 FLMCR1、FLMCR2、EBR1、EBR2、RAMCR は 8 ビットのレジスタです。  
アクセスはバイトアクセスとしてください。

\*1 アドレス下位 16 ビットを示しています。

\*2 FWE 端子に High レベルが入力されているときの初期値は H'80 です。

## 18.3 レジスタの説明

### 18.3.1 フラッシュメモリコントロールレジスタ 1 (FLMCR1)

ビット :	7	6	5	4	3	2	1	0
	FWE	SWE	ESU	PSU	EV	PV	E	P
初期値 :	- *	0	0	0	0	0	0	0
R/W :	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】\*FWE端子の状態により設定されます。

FLMCR1 は、フラッシュメモリの各動作モードを制御する 8 ビットのレジスタです。アドレス H'00000 ~ H'5FFFF に対してプログラムベリファイモード、イレースベリファイモードに遷移させるには、FWE = 1 時に SWE ビットをセット後、PV ビットまたは EV ビットをセットします。アドレス H'00000 ~ H'5FFFF に対して、プログラムモードへ遷移させるには、FWE = 1 時に、SWE ビットをセット後、PSU ビットをセットし、最後に P ビットをセットします。アドレス H'00000 ~ H'5FFFF に対してイレースモードへ遷移するには、FWE = 1 時に、SWE ビットをセット後、ESU ビットをセットし、最後に E ビットをセットします。FLMCR1 は、リセット、ハードウェアスタンバイモード、ソフトウェアスタンバイモードで初期化されます。FWE 端子に High レベルが入力されているときの初期値は H'80 です。ローレベルが入力されているときは H'00 です。モード 6 ではフラッシュメモリのオンボードプログラミングモードをサポートしていませんので、FWE 端子は Low レベルに固定することが必要です。内蔵フラッシュメモリが無効のときに本レジスタを読み出すと H'00 が読み出され、書き込みも無効となります。

本レジスタのビット 6~0 に 1 をセットするときには、1 ビットずつ行ってください。また、FLMCR1 の SWE ビットへの書き込みは FWE = 1 のとき、ESU、PSU、EV、PV ビットへの書き込みは FWE = 1、SWE = 1 のとき、E ビットへの書き込みは FWE = 1、SWE = 1、ESU = 1 のとき、P ビットへの書き込みは FWE = 1、SWE = 1、PSU = 1 のときのみ有効です。

- 【注】
1. 本レジスタの各ビットの設定は、誤書き込みや誤消去を防止するために書き込みフローおよび、消去フローに従ってください。
  2. 本レジスタの設定により、プログラムモード / イレースモード / プログラムベリファイモード / イレースベリファイモードへと遷移します。フラッシュメモリを通常の内蔵 ROM として読み出す際には、本レジスタのビット 6~0 をクリアした状態にしてください。

## ビット7：フラッシュライトイネーブルビット（FWE）

FWE ビットは、フラッシュメモリの書き込み/消去をハードウェアプロテクトするビットです。

ビット7	説明
FWE	
0	FWE 端子に Low レベルが入力されているとき（ハードウェアプロテクト状態）
1	FWE 端子に High レベルが入力されているとき

## ビット6：ソフトウェアライトイネーブルビット（SWE）

フラッシュメモリの書き込み/消去の有効または無効を選択するビットです（ビット5～0、EBR1の7～0ビット、EBR2の3～0ビットをセットするときにセットしてください）。

ビット6	説明
SWE	
0	書き込み/消去無効（初期値）
1	書き込み有効 [セットするための条件] FWE = 1 のとき

【注】 SWE ビットを1にセットしている間は、SLEEP 命令は実行しないでください。

## ビット5：イレースセットアップビット（ESU）

イレースモードへの遷移の準備をするビットです。FLMCR1のEビットを1にセットする前に1にセットしてください（SWE、PSU、EV、PV、E、Pビットを同時に設定しないでください）。

ビット5	説明
ESU	
0	イレースセットアップ解除（初期値）
1	イレースセットアップ [セットするための条件] FWE = 1、SWE = 1 のとき

## ビット4：プログラムセットアップビット（PSU）

プログラムモードへの遷移の準備をするビットです。FLMCR1のPビットを1にセットする前に1にセットしてください（SWE、ESU、EV、PV、E、Pビットを同時に設定しないでください）。

ビット4	説明
PSU	
0	プログラムセットアップ解除 (初期値)
1	プログラムセットアップ [セットするための条件] FWE = 1、SWE = 1 のとき

## ビット3：イレースベリファイビット（EV）

イレースベリファイモードへの遷移、解除を選択するビットです（SWE、ESU、PSU、PV、E、Pビットを同時に設定しないでください）。

ビット3	説明
EV	
0	イレースベリファイモードを解除 (初期値)
1	イレースベリファイモードに遷移 [セットするための条件] FWE = 1、SWE = 1 のとき

## ビット2：プログラムベリファイビット（PV）

プログラムベリファイモードへの遷移、解除を選択するビットです（SWE、ESU、PSU、EV、E、Pビットを同時に設定しないでください）。

ビット2	説明
PV	
0	プログラムベリファイモードを解除 (初期値)
1	プログラムベリファイモードに遷移 [セットするための条件] FWE = 1、SWE = 1 のとき

## ビット1：イレースビット (E)

イレースモードへの遷移、解除を選択するビットです (SWE、ESU、PSU、EV、PV、P ビットを同時に設定しないでください)。

ビット1	説明
E	
0	イレースモードを解除 (初期値)
1	イレースモードに遷移 [セットするための条件] FWE = 1、SWE = 1、ESU = 1 のとき

【注】 E ビットをセットしている間は、フラッシュメモリへのアクセスは行わないでください。

## ビット0：プログラムビット (P)

プログラムモードへの遷移、解除を選択するビットです (SWE、PSU、ESU、EV1、PV、E ビットを同時に設定しないでください)。

ビット0	説明
P	
0	プログラムモードを解除 (初期値)
1	プログラムモードに遷移 [セットするための条件] FWE = 1、SWE = 1、PSU = 1 のとき

【注】 P ビットをセットしている間は、フラッシュメモリへのアクセスは行わないでください。

## 18.3.2 フラッシュメモリコントロールレジスタ 2 (FLMCR2)

ビット：	7	6	5	4	3	2	1	0
	FLER	-	-	-	-	-	-	-
初期値：	0	0	0	0	0	0	0	0
R/W：	R	R	R	R	R	R	R	R

FLMCR2 は、フラッシュメモリの各動作モードを制御する 8 ビットのレジスタです。FLMCR2 は、リセット、ハードウェアスタンバイモード、ソフトウェアスタンバイモードのとき H'00 に初期化されます。内蔵フラッシュメモリが無効のときは読み出すと H'00 が読み出されます。

【注】 FLMCR2 は読み出し専用レジスタです。書き込みはしないでください。

## ビット7：フラッシュメモリエラー（FLER）

フラッシュメモリ動作中（書き込み、消去）にエラーが発生したことを示すビットです。FLER=1にセットされると、フラッシュメモリはエラープロテクトモードに遷移します。

ビット7	説明
FLER	
0	フラッシュメモリは正常に動作しています。 フラッシュメモリへの書き込み/消去プロテクト（エラープロテクト）が無効 [クリア条件] リセット（ $\overline{\text{RES}}$ 端子および WDT リセット）またはハードウェア スタンバイモードのとき (初期値)
1	フラッシュメモリへの書き込み/消去中にエラーが発生したことを示します。 フラッシュメモリへの書き込み/消去プロテクト（エラープロテクト）が有効 [セット条件] (1) 書き込み/消去中にフラッシュメモリをリード*2したとき（ベクタリード および命令フェッチを含む。ただし、フラッシュメモリ空間とオーバラッ プした RAM エリアのリードは除く）。 (2) 書き込み/消去中の例外処理（ただし、リセット、不当命令、トラップ命 令、ゼロ除算時の例外処理は除く）の開始直後。 (3) 書き込み/消去中に SLEEP 命令（ソフトウェアスタンバイを含む）を実 行したとき。 (4) 書き込み/消去中にバス権を解放したとき

## ビット6～0：リザーブビット

リザーブビットです。読み出すと常に0が読み出されます。

## 18.3.3 消去ブロック指定レジスタ 1（EBR1）

ビット：	7	6	5	4	3	2	1	0
	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0
初期値：	0	0	0	0	0	0	0	0
R/W：	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

フラッシュメモリの消去エリアをブロックごとに設定する 8 ビットのレジスタです。EBR1 は、リセット、ハードウェアスタンバイモード、ソフトウェアスタンバイモード、FWE 端子に Low レベルが入力されているとき、および FWE 端子に High レベルが入力されていても FLMCR1 の SWE ビットが設定されていないときは H'00 に初期化されます。EBR1 の各ビットに 1 をセットすると、対応するブロックが消去可能となります。それ以外のブロックは、消去プロテクト状態になります。EBR1 は EBR2 と合わせて 1 ビットの



み設定してください(2ビット以上を同時に設定しないでください)。内蔵フラッシュメモリが無効のときは本レジスタを読み出すとH'00が読み出され、消去も無効となります。

フラッシュメモリのブロック分割方法は、表 18.5 を参照してください。全面消去をする場合は、各ブロック単位に順次消去してください。

本LSIではモード6でのオンボードプログラミングモードをサポートしていませんので、本レジスタの各ビットにはモード6のときに1をセットすることはできません。

### 18.3.4 消去ブロック指定レジスタ2 (EBR2)

ビット :	7	6	5	4	3	2	1	0
	-	-	EB13	EB12	EB11	EB10	EB9	EB8
初期値 :	0	0	0	0	0	0	0	0
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W

フラッシュメモリの消去エリアをブロックごとに設定する8ビットのレジスタです。EBR2は、リセット、ハードウェアスタンバイモード、ソフトウェアスタンバイモード、FWE端子にLowレベルが入力されているとき、H'00に初期化されます。またFWE端子にHighレベルが入力されていても、FLMCR1のSWEビットが設定されていないときはビット0に初期化されます。EBR2の各ビットに1をセットすると、対応するブロックが消去可能となります(それ以外のブロックは、消去プロテクト状態になります)。EBR2はEBR1と合わせて1ビットのみ設定してください(2ビット以上を同時に設定しないでください)。ビット7、6はリザーブビットです。内蔵フラッシュメモリが無効のときは読み出すとH'00が読み出され、消去も無効となります。

フラッシュメモリのブロック分割方法は、表 18.5 を参照してください。全面消去をする場合は、各ブロック単位に順次消去してください。

本LSIではモード6でのオンボードプログラミングモードをサポートしていませんので、本レジスタの各ビットにはモード6の時に1をセットすることはできません。

【注】本レジスタのビット7、6は読み出し専用ビットです。1をセットしないでください。EBR1/EBR2のあるビットがセットされている状態でビット7、6をセットした場合、EBR1/EBR2はH'00に初期化されます。

表 18.5 消去ブロックの分割

ブロック (サイズ)	アドレス
EB0 (4k バイト)	H'000000 ~ H'000FFF
EB1 (4k バイト)	H'001000 ~ H'001FFF
EB2 (4k バイト)	H'002000 ~ H'002FFF
EB3 (4k バイト)	H'003000 ~ H'003FFF
EB4 (4k バイト)	H'004000 ~ H'004FFF
EB5 (4k バイト)	H'005000 ~ H'005FFF
EB6 (4k バイト)	H'006000 ~ H'006FFF
EB7 (4k バイト)	H'007000 ~ H'007FFF
EB8 (32k バイト)	H'008000 ~ H'00FFFF
EB9 (64k バイト)	H'010000 ~ H'01FFFF
EB10 (64k バイト)	H'020000 ~ H'02FFFF
EB11 (64k バイト)	H'030000 ~ H'03FFFF
EB12 (64k バイト)	H'040000 ~ H'04FFFF
EB13 (64k バイト)	H'050000 ~ H'05FFFF

### 18.3.5 RAM コントロールレジスタ (RAMCR)

ビット :	7	6	5	4	3	2	1	0
	-	-	-	-	RAMS	RAM2	RAM1	RAM0
初期値 :	1	1	1	1	0	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W

フラッシュメモリのリアルタイムな書き替えをエミュレートするとき、RAMの一部と重ね合わせるフラッシュメモリのエリアを設定するレジスタです。リセット、またはハードウェアスタンバイモードのときに H'00 に初期化されます。RAMCR の設定は、ユーザーモード、ユーザープログラミングモードで行ってください。

フラッシュメモリエリアの分割法は、表 18.6 を参照してください。なお、エミュレーション機能を確実に動作させるために、本レジスタの書き替え直後に RAM エミュレーションの対象 ROM をアクセスしないでください。直後にアクセスした場合には、正常なアクセスは保証されません。

#### ビット7~4 : リザーブビット

読み出すと常に 1 が読み出されます。ライトは無効です。

## ビット3 : RAMセレクト (RAMS)

RAM によるフラッシュメモリのエミュレーション選択 / 非選択を設定するビットです。RAMS=1 のときは、フラッシュメモリ全ブロックの書き込み / 消去プロテクト状態となります。

ビット3	説明
RAMS	
0	エミュレーション非選択 フラッシュメモリ全ブロック書き込み / 消去プロテクト無効 (初期値)
1	エミュレーション選択 フラッシュメモリ全ブロック書き込み / 消去プロテクト有効

## ビット2、1、0 : フラッシュメモリエリア選択

ビット3と共に使用し、RAMと重ね合わせるフラッシュメモリエリアを選択します。(表 18.6 参照)

表 18.6 フラッシュメモリエリアの分割

RAM エリア	ブロック名	RAMS	RAM2	RAM1	RAM0
H'FFE000 ~ H'FFEFF	RAM エリア 4k バイト	0	*	*	*
H'000000 ~ H'000FFF	EB0 (4k バイト)	1	0	0	0
H'001000 ~ H'001FFF	EB1 (4k バイト)	1	0	0	1
H'002000 ~ H'002FFF	EB2 (4k バイト)	1	0	1	0
H'003000 ~ H'003FFF	EB3 (4k バイト)	1	0	1	1
H'004000 ~ H'004FFF	EB4 (4k バイト)	1	1	0	0
H'005000 ~ H'005FFF	EB5 (4k バイト)	1	1	0	1
H'006000 ~ H'006FFF	EB6 (4k バイト)	1	1	1	0
H'007000 ~ H'007FFF	EB7 (4k バイト)	1	1	1	1

\* : Don't care

【注】 モード6 (シングルチップノーマルモード) のときは、RAM によるフラッシュメモリのエミュレーションはサポートしていません。そのため、ライトは可能ですが1をセットしないでください。また、RAM によるフラッシュメモリのエミュレーションを行う場合、SYSCR の RAME ビットは必ず1にセットしてください。

## 18.4 動作の概要

### 18.4.1 モード遷移図

リセット状態で各モード端子と FWE 端子を設定しリセットスタートすると、マイコンは図 18.2 に示すような各動作モードへ遷移します。ユーザモードではフラッシュメモリの読み出しはできますが、フラッシュメモリの書き込み/消去はできません。

フラッシュメモリへの書き込み/消去を行えるモードとしてブートモード、ユーザプログラムモード、PROMモードがあります。

H8/3068F-ZTATのモード6(内蔵ROM有効ノーマルモード)では、ブートモードおよびユーザプログラミングモードは使用できません。

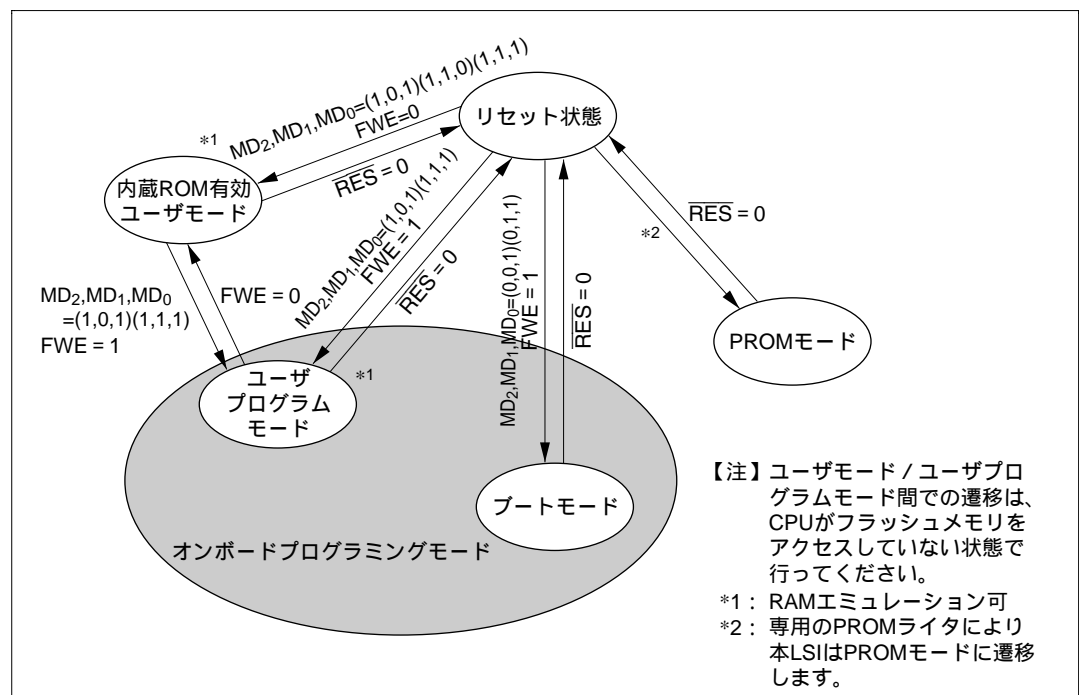


図 18.2 フラッシュメモリに関する状態遷移

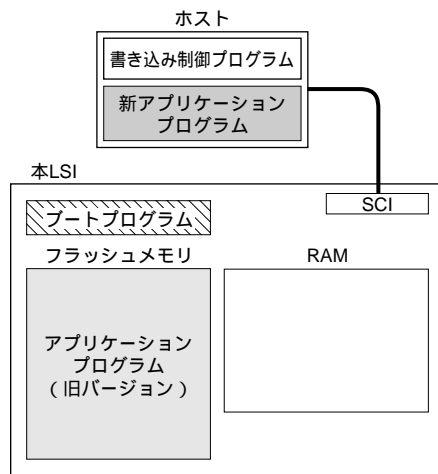
FWE 端子の High Low、Low High によって、通常のユーザモードとオンボードプログラミングモードとの状態遷移が行われます。その際、誤動作(誤書き込みや誤消去)を回避するために、フラッシュメモリコントロールレジスタ (FLMCR1) の各ビットを0にクリアした状態にしてください。また、各ビットのクリア後にはウェイト時間が必要です。このウェイト時間が不足した場合、正常動作は保証されません。

## 18.4.2 オンボードプログラミングモード

### (1) ブートモードの例

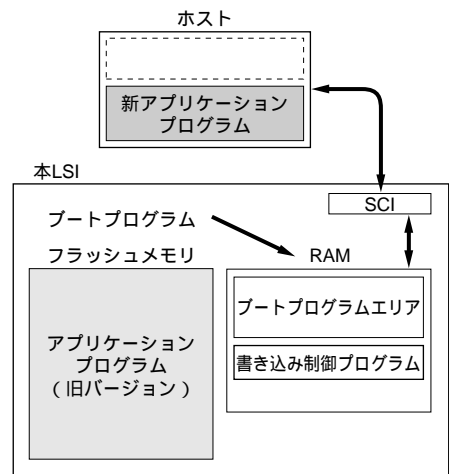
#### 1. 初期状態

フラッシュメモリには、旧バージョンのプログラムあるいはデータが書かれたままです。書き込み制御プログラムおよび新アプリケーションプログラムはユーザがあらかじめホストに用意してください。



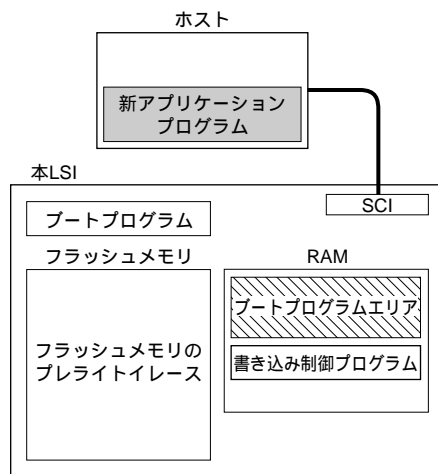
#### 2. 書き込み制御プログラムの転送

ブートモードに移行すると本LSI内のブートプログラム（既にLSIに内蔵されている）が起動し、ホストにある書き込み制御プログラムをRAMにSCI通信で転送します。また、フラッシュメモリの消去に必要なブートプログラムは、RAMのブートプログラムエリアに自動的に転送されます。



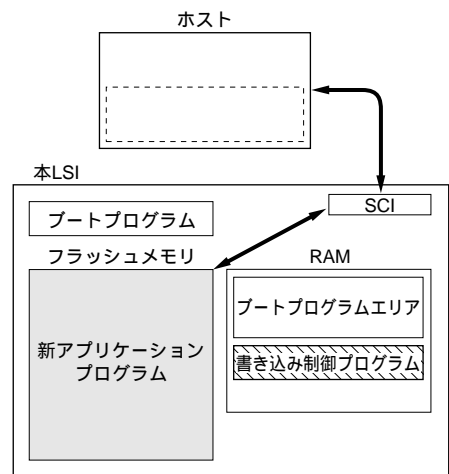
#### 3. フラッシュメモリの初期化

ブートプログラムエリア（RAM内）にある消去プログラムを実行し、フラッシュメモリを初期化（HFF）します。ブートモード時は、ブロックに関係なくフラッシュメモリを全面消去します。



#### 4. 新アプリケーションプログラムの書き込み

ホストよりRAMに転送した書き込み制御プログラムを実行して、ホストにある新アプリケーションプログラムをフラッシュメモリに書き込みます。

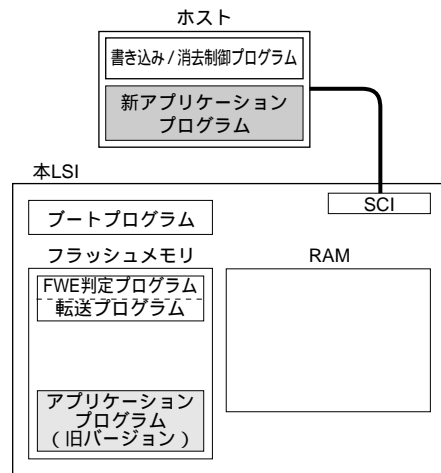


プログラム実行状態

## (2) ユーザプログラムモードの例

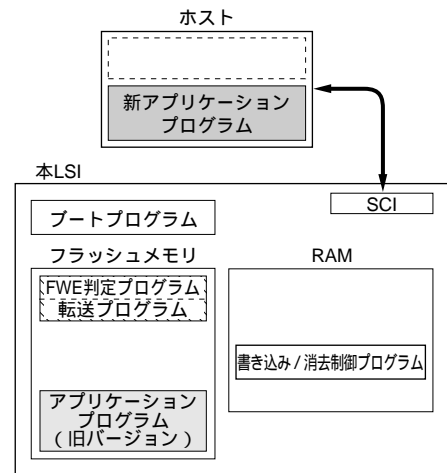
## 1. 初期状態

(1) ユーザプログラムモードに移したことを確認するFWE判定プログラム、(2) フラッシュメモリから内蔵RAMに書き込み/消去制御プログラムを転送するプログラムをあらかじめフラッシュメモリにユーザが書き込んでおいてください。(3) 書き込み/消去制御プログラムはホストまたはフラッシュメモリに用意してください。



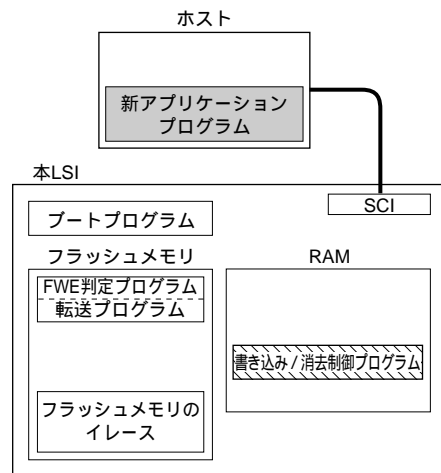
## 2. 書き込み/消去制御プログラムの転送

ユーザプログラムモードに移すると、ユーザソフトはこれを認識してフラッシュメモリ内の転送プログラムを実行して、書き込み/消去制御プログラムをRAMに転送します。



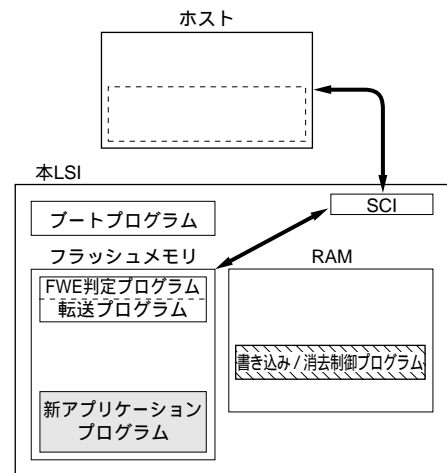
## 3. フラッシュメモリの初期化


RAM上の書き込み/消去プログラムを実行し、フラッシュメモリを初期化 (HFF) します。消去は、ブロック単位で行えます。バイト単位の消去はできません。



## 4. アプリケーションプログラムの書き込み

次にホストにある新アプリケーションプログラムを消去したフラッシュメモリのブロックに書き込みます。消去されていないブロックに対する書き込みは行わないでください。



 プログラム実行状態

### 18.4.3 RAM によるフラッシュメモリのエミュレーション

本 LSI ではフラッシュメモリと RAM の一部を重ね合わせる (オーバーラップ RAM) ことで、フラッシュメモリの書き替えをリアルタイムにエミュレートすることができます。エミュレーション機能を実行しているときに RAMCR で設定したエミュレーションブロックをアクセスすると、オーバーラップ RAM に書かれているデータが読み出されます。

ユーザモード、ユーザプログラムモードでエミュレーションを行ってください。

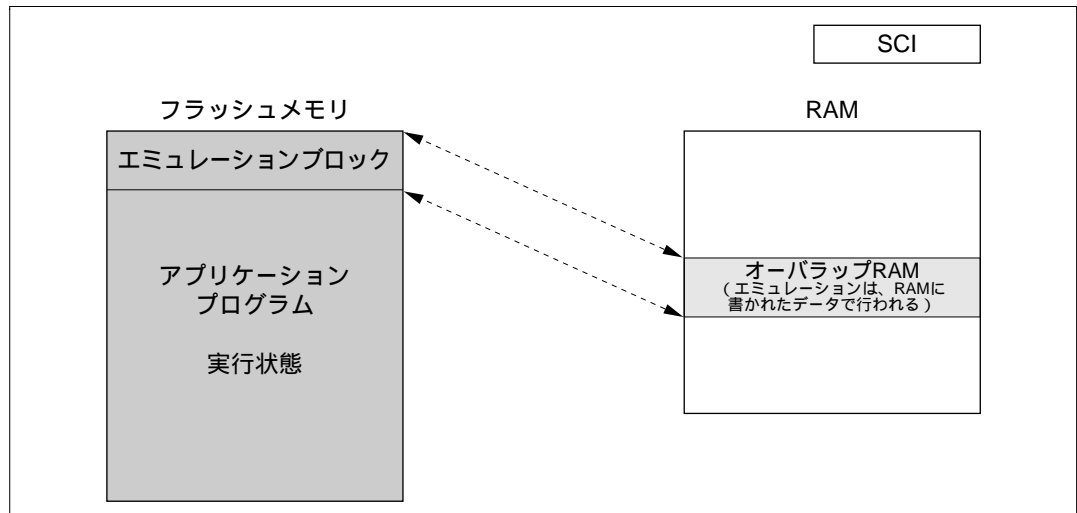


図 18.3 ユーザモード、ユーザプログラムモードのオーバーラップ RAM データの読み出し

オーバーラップ RAM のデータが確定したら、RAMS ビットをクリアして RAM のオーバーラップを解除し、実際にフラッシュメモリへの書き込みを行ってください。

ただし、オンボードプログラミングモードで、書き替え制御プログラムを RAM に転送してくるときに、転送先とオーバーラップ RAM が重ならないようにしてください。オーバーラップ RAM 内のデータが書き替えられてしまいます。

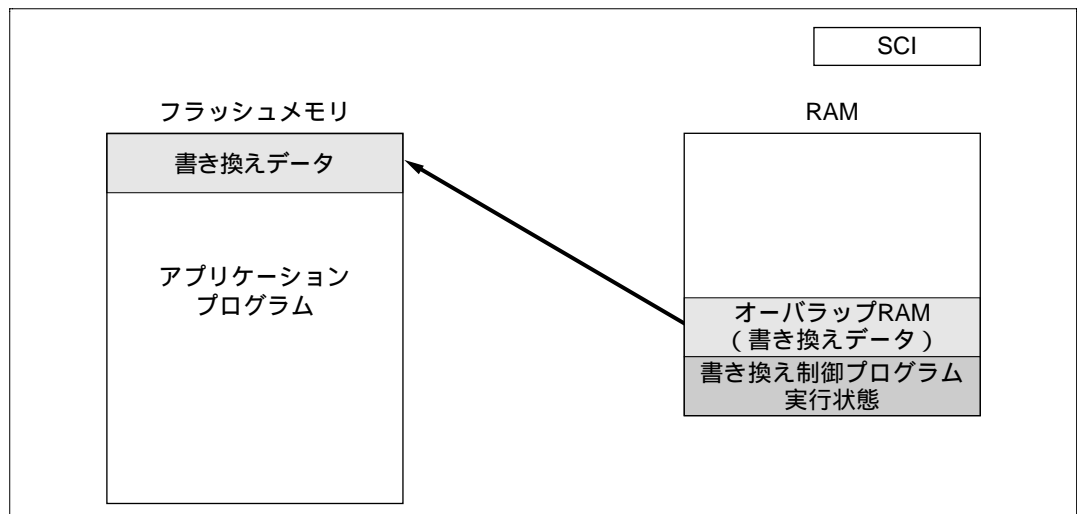
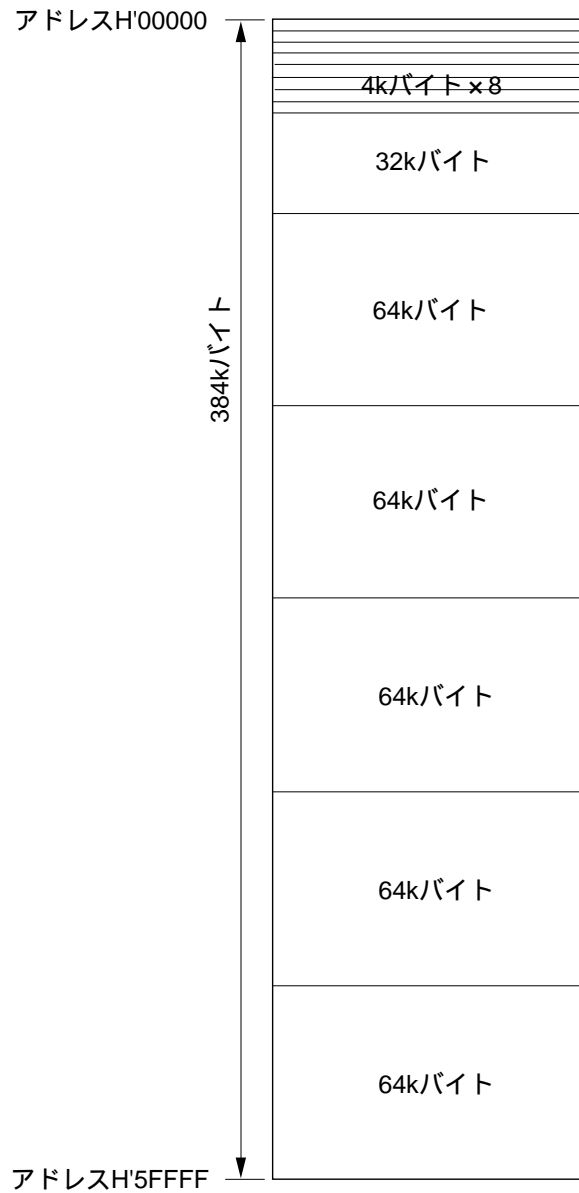


図 18.4 ユーザプログラムモードのオーバーラップ RAM データの書き込み

#### 18.4.4 ブロック分割法

本LSIのフラッシュメモリは、64kバイト（5ブロック）、32kバイト（1ブロック）、4kバイト（8ブロック）に分割されています。このブロック単位で消去することができます。





## 18.5 オンボードプログラミングモード

オンボードプログラミングモードに端子を設定しリセットスタートすると、内蔵フラッシュメモリへの書き込み/消去/ベリファイを行うことができるオンボードプログラミング状態へ遷移します。オンボードプログラミングモードには、ブートモードとユーザプログラムモードの2種類の動作モードがあります。各モードへ遷移する端子の設定方法を表 18.6 に示します。また、フラッシュメモリに関する各モードへの状態遷移図は図 18.2 を参照してください。

H8/3068F-ZTAT のモード 6 (内蔵 ROM が有効) では、ブートモードおよびユーザプログラムモードは使用できません。

表 18.6 オンボードプログラミングモードの設定方法

モード名		FWE	MD <sub>2</sub>	MD <sub>1</sub>	MD <sub>0</sub>
ブートモード	モード 5	1* <sup>1</sup>	0* <sup>2</sup>	0	1
	モード 7		0* <sup>2</sup>	1	1
ユーザプログラムモード	モード 5	1* <sup>1</sup>	1	0	1
	モード 7		1	1	1

【注】 \*1 High レベルの印加タイミングについては、ブートモード使用時の注意を参照してください。

\*2 ブートモード時は、MD<sub>2</sub>の設定を反転入力にしてください。

また、H8/3068F-ZTAT のブートモードでは、モードコントロールレジスタ (MDCR) のモードセレクト 2~0 ビット (MDS2~MDS0) はモード端子 (MD<sub>2</sub>~MD<sub>0</sub>) のレベルを反映した値になります。

### 18.5.1 ブートモード

ブートモードを使用する場合は、フラッシュメモリへの書き込み制御プログラムをホストに準備しておく必要があります。また、使用する SCI のチャンネル 1 は調歩同期式モードに設定されています。

本 LSI の端子をブートモードに設定後リセットスタートすると、あらかじめマイコン内に組み込まれているブートプログラムを起動し、ホストに用意した書き込み制御プログラムを SCI を使って本 LSI へ順次送信します。本 LSI では、SCI で受信した書き込み制御プログラムを内蔵 RAM の書き込み制御プログラムエリアに書き込みます。転送終了後書き込み制御プログラムエリアの先頭アドレス (TBD) に分岐し、書き込み制御プログラム実行状態となります (フラッシュメモリの書き込み / 消去が可能となります)。

図 18.5 にブートモード時のシステム構成図、図 18.6 にブートモード実行手順を示します。

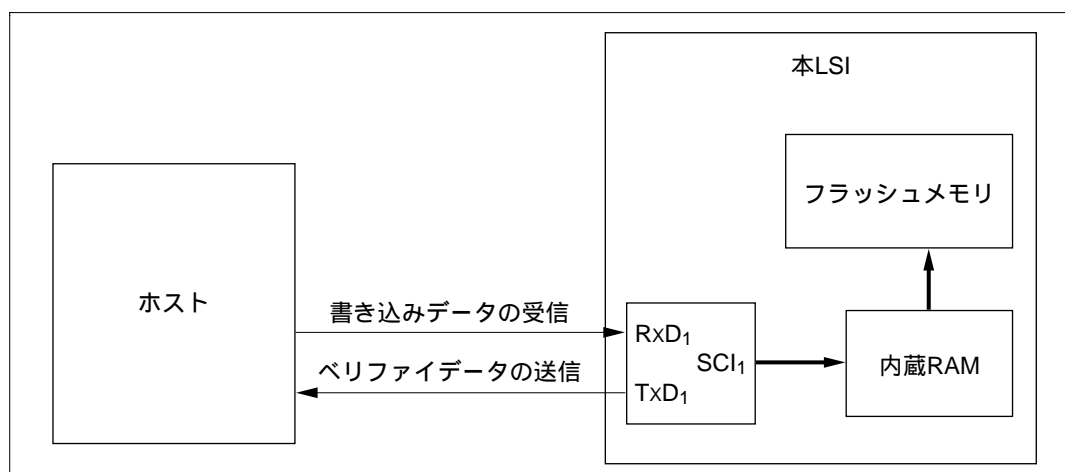


図 18.5 ブートモード時のシステム構成図

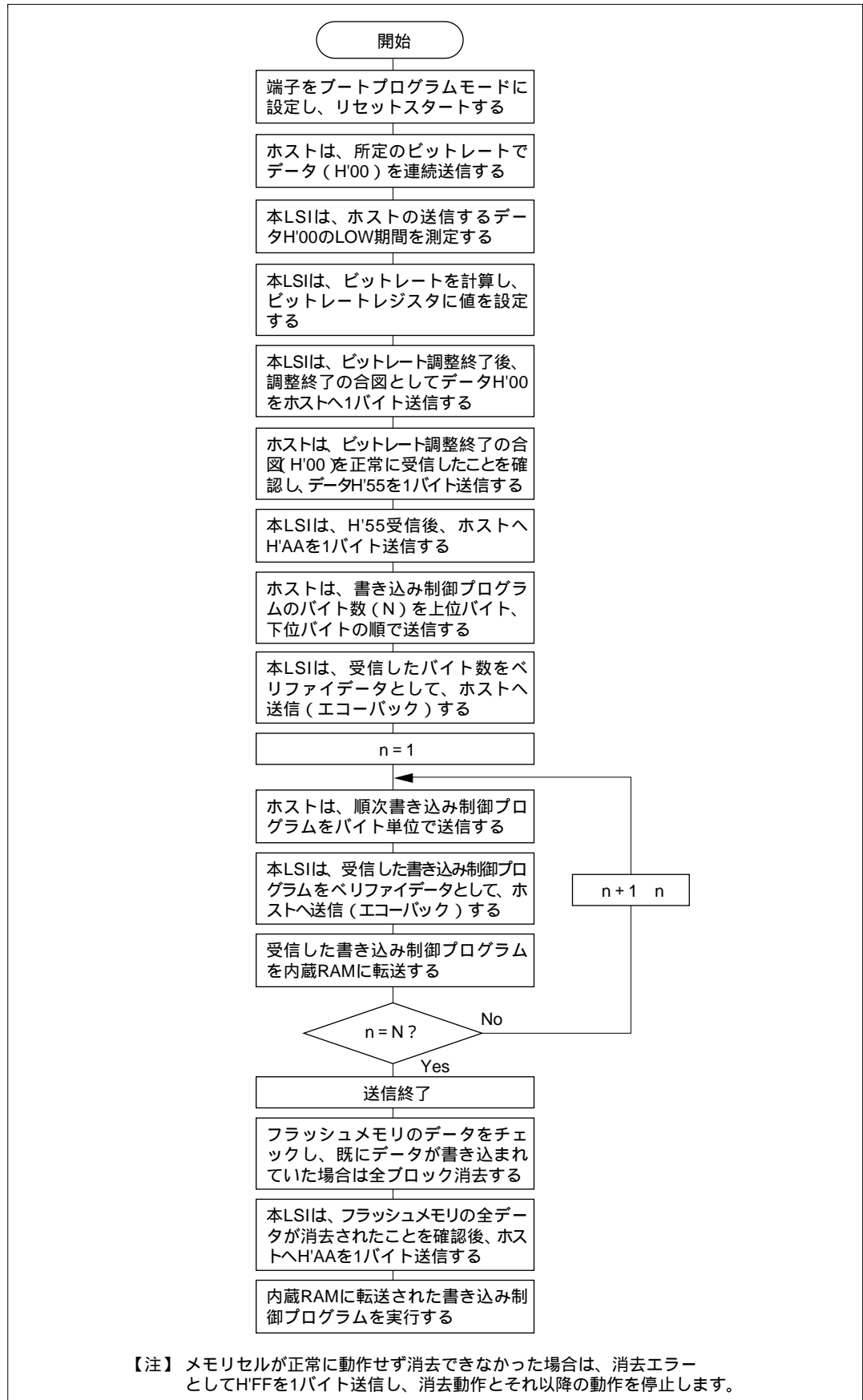


図 18.6 ブートモード実行手順

## (1) SCI ビットレートの自動合わせ込み動作



ブートモードが起動すると、本 LSI はホストより連続送信される調歩同期式 SCI 通信のデータ (H'00) の Low 期間を測定します。このときの SCI 受信 / 送信フォーマットを「8 ビットデータ、1 ストップビット、パリティなし」に設定してください。本 LSI は、測定した Low 期間よりホストの送信するビットレートを計算し、ビット調整終了合図としてホストへ H'00 を 1 バイト送信します。ホストは、この調整終了合図 (H'00) を正常に受信したことを確認し、本 LSI へ H'55 を 1 バイト送信してください。受信が正常に行われなかった場合は、再度ブートモードを起動し (リセット)、上述の操作を行ってください。ホストが送信するビットレート、および本 LSI のシステムクロックの周波数によってホストと本 LSI のビットレートに誤差が生じます。正常に SCI を動作させるために、ホストの転送ビットレートを (19,200 , 9,600 , 4,800 ) bps\*<sup>1</sup> に設定してください。

ホストの転送ビットレートと本 LSI のビットレートの自動合わせ込みが可能なシステムクロックの周波数を表 18.8 に示します。このシステムクロックの範囲内でブートプログラムを実行してください。

表 18.8 本 LSI のビットレートの自動合わせ込みが可能なシステムクロックの周波数

ホストのビットレート ( bps )	本 LSI のビットレートの自動合わせ込みが可能な システムクロックの周波数 ( MHz )
19,200	16 ~ 25
9,600	8 ~ 25
4,800	4 ~ 25

【注】 \*1 ホストのビットレートは 4800、9600、19200bps の設定のみとし、それ以外の設定は使用しないでください。

本 LSI は表 18.8 に示すビットレートとシステムクロックの組み合わせ以外にも、ビットレートの自動合わせ込みを行う場合がありますが、ホストと本 LSI とのビットレートに誤差が生じ、その後の転送が正常に行われません。このため、ブートモードの実行は、必ず表 18.8 に示すビットレートとシステムクロックの組み合わせの範囲内で行なってください。

## (2) ブートモード時の内蔵 RAM エリアの分割

ブートモードでは、RAM エリアは図 18.7 に示すようにブートプログラムで使用するエリアと SCI で書き込み制御プログラムを転送してくるエリアに分かれています。ブートプログラムエリアは、ブートモード中の実行状態が転送してきた書き込み制御プログラムへ遷移するまで使用できません。

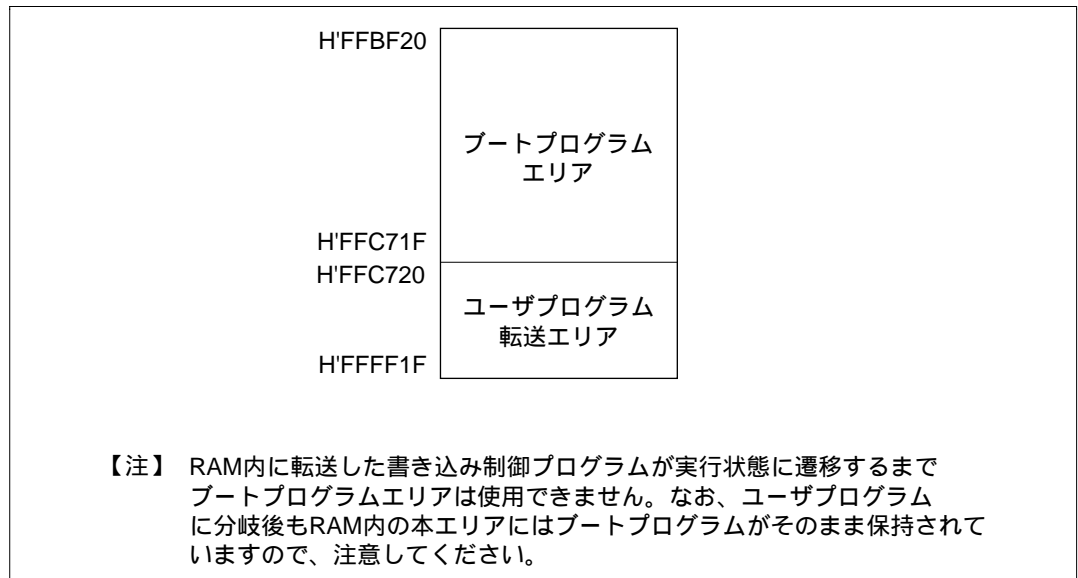
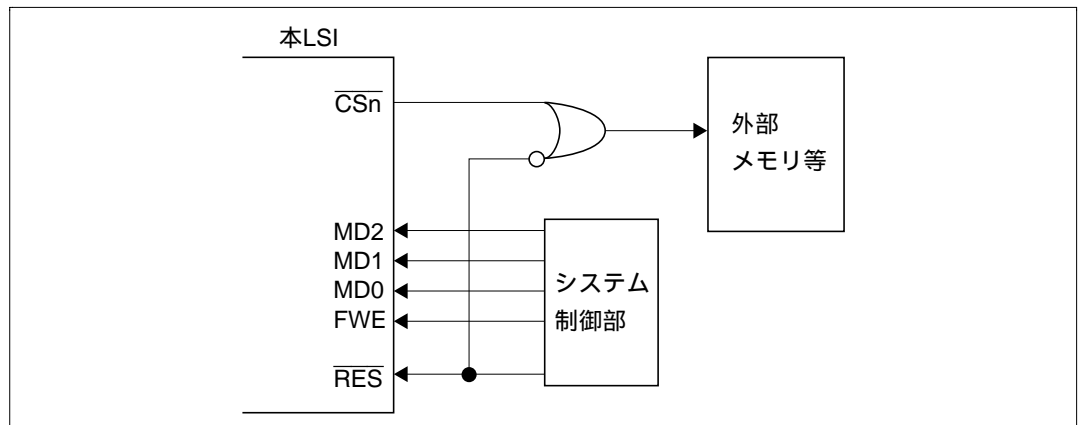


図 18.7 ブートモード時の RAM エリア

## (3) ブートモード使用時の注意事項

- (a) 本 LSI は、ブートモードでリセット解除すると、SCI の RxD<sub>1</sub> 端子の Low 期間を測定します。RxD<sub>1</sub> 端子が High の状態でリセット解除してください。リセット解除後、RxD<sub>1</sub> 端子から入力される Low 期間を測定できるようになるまで、本 LSI は約 100 ステート必要です。
- (b) ブートモードは、フラッシュメモリに書き込まれているデータがある場合(全データが 1 でないとき)、フラッシュメモリの全ブロックを消去します。本モードは、オンボード状態での初期の書き込み、あるいは、ユーザプログラムモードで起動するプログラムを誤って消去し、ユーザプログラムモードが実行できなくなった場合の強制復帰等に使用してください。
- (c) フラッシュメモリの書き込み中、あるいは消去中に割り込みを使用することはできません。
- (d) RxD<sub>1</sub> 端子および TxD<sub>1</sub> 端子は、ボード上でプルアップして使用してください。

- (e) 本 LSI は、書き込み制御プログラムに分岐するときに内蔵 SCI (チャンネル 1) の送受信動作を終了 (SCR の RE = 0、TE = 0) しますが、BRR には、合わせ込んだビットレートの値を保持しています。
- また、このときトランスミットデータ出力端子 TxD<sub>1</sub> は、High レベル出力状態 (P9DDR の P9<sub>1</sub>DDR = 1、P9DR の P9<sub>1</sub>DR = 1) となっています。
- さらにこのとき、CPU 内蔵の汎用レジスタの値は不定です。このため、書き込み制御プログラムに分岐した直後に、汎用レジスタの初期設定を必ず行ってください。特にスタックポインタ (SP) は、サブルーチンコール時などに暗黙的に使用されますので、書き込み制御プログラムで使用するスタックエリアを必ず指定してください。
- 上記以外の内蔵レジスタについては、初期値が変更されるものではありません。
- (f) ブートモードへの遷移は、表 18.6 のモード設定にしたがって、端子を設定しリセットスタートすることにより可能です。
- ・ ブートモードから通常モードへ遷移する場合は、モード遷移する前にマイコン内部のブートモード状態を、 $\overline{\text{RES}}$  端子によるリセット入力で解除する\*<sup>1</sup> 必要があります。この時、 $\overline{\text{RES}}$  端子は最低 20 システムクロック以上の間、Low レベルに保持する必要があります。\*<sup>3</sup>
  - ・ ブートモードの途中で、モード端子 (MD<sub>2</sub> ~ MD<sub>0</sub>) および FWE 端子の入力レベルを切り換えないでください。モードを遷移させる場合には、事前に  $\overline{\text{RES}}$  端子に Low レベルを入力し、リセット状態にしてください。また、ブートモード状態でウォッチドッグタイマリセットが発生した場合、マイコン内部のモード状態は解除されず、モード端子の状態にかかわらず内蔵のブートプログラムが再起動されます。
  - ・ ブートプログラム実行中やフラッシュメモリへの書き込み / 消去中に、FWE 端子を Low レベルにしないでください\*<sup>2</sup>。
- (g) リセット中にモード端子の入力レベルを変化 (例えば Low レベル High レベル) させると、マイコンの動作モードが切り替わることにより、アドレス兼用ポート、およびバス制御出力信号 ( $\overline{\text{CSn}}$ 、 $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{LWR}}$ 、 $\overline{\text{HWR}}$ ) の状態が変化することがあります。
- このため、これらの端子はリセット中の出力信号として直接使用しないよう、マイコン外部で禁止する必要があります。



- 【注】\*1 モード端子と FWE 端子の入力はリセット解除のタイミングに対し、モードプログラミングセット時間 ( $t_{MDS}$ ) を満足する必要があります。
- \*2 FWE の印加 / 解除の注意については「18.11 フラッシュメモリの書き込み / 消去時の注意」を参照してください。
- \*3 「4.2.2 リセットシーケンス」および「18.11 フラッシュメモリの書き込み / 消去時の注意」を参照してください。H8/3068F-ZTAT では最低 20 システムクロック必要です。

## 18.5.2 ユーザプログラムモード

ユーザプログラムモードに設定すると、ユーザの書き込み / 消去制御プログラムによるフラッシュメモリの書き込み、消去が可能になります。したがって、あらかじめ基板上に FWE 制御手段、および書き替えデータ供給手段を設け、必要に応じてプログラムエリアの一部に書き込み / 消去プログラムを内蔵させておくことにより、内蔵フラッシュメモリのオンボード書き替えを行うことができます。

本モードの設定では、内蔵フラッシュメモリの有効なモード 5、7 で起動し、FWE 端子に High レベルを印加します。この状態の動作では、フラッシュメモリ以外の周辺機能はモード 5、7 と同じ動作をします。また、モード 6 ではフラッシュメモリの書き込み / 消去を行わないでください。モード 6 設定時は FWE 端子を必ず Low レベルにしてください。

なお、プログラム / 消去を行っている間、フラッシュメモリを読み出すことはできませんので、書き替えプログラムを外部メモリ上に置くか、または書き替えプログラムをいったん RAM エリアに転送し、RAM 内で実行してください。

RAM 内でプログラム実行中に、ユーザプログラムモードに遷移する場合の実行手順を図 18.8 に示します。なお、リセットスタート時にユーザプログラムモードから起動することも可能です。

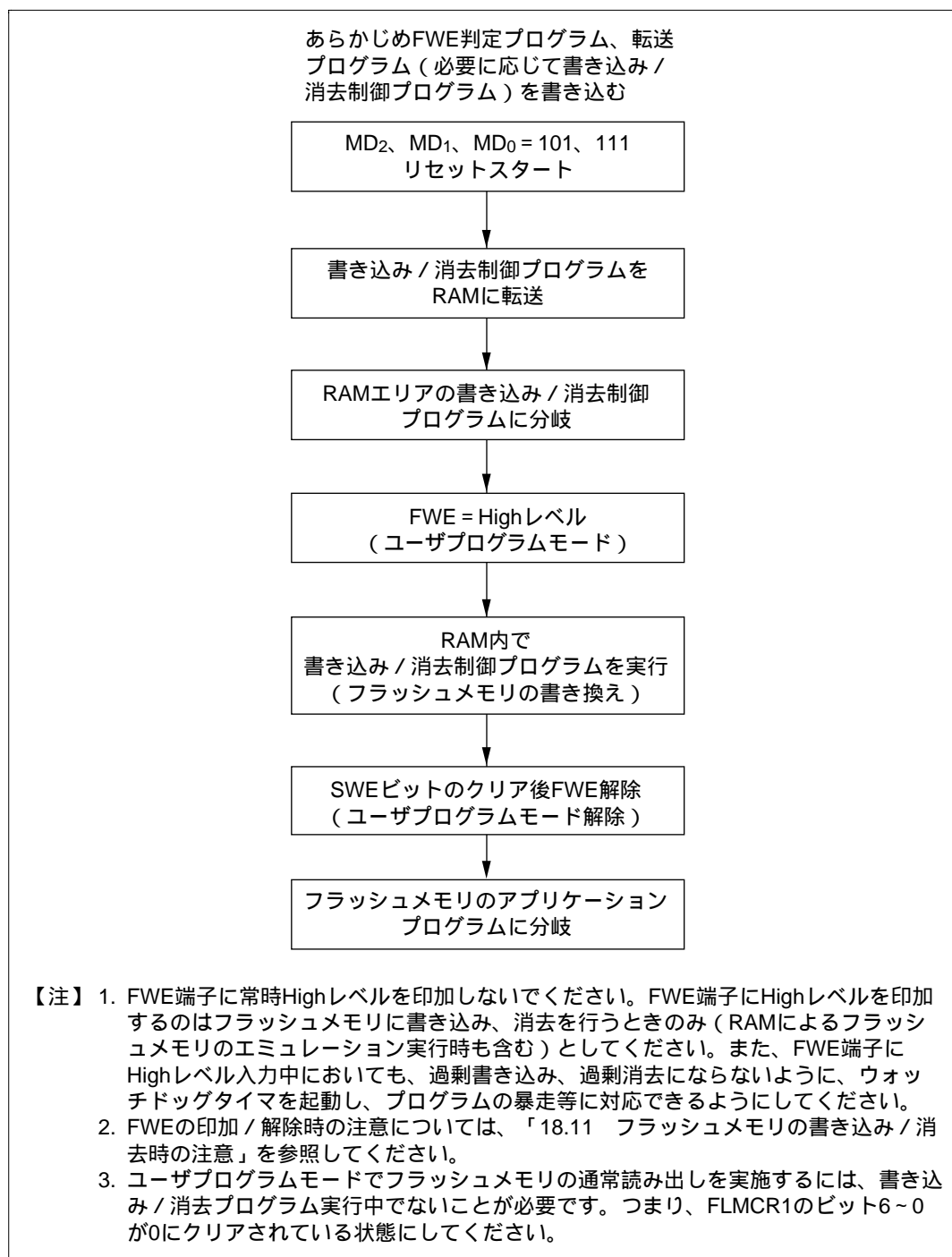


図 18.8 ユーザプログラムモードの実行手順例



## 18.6 フラッシュメモリの書き込み / 消去

オンボードプログラミングモードでのフラッシュメモリの書き込み / 消去は、CPU を用いてソフトウェア的に行う方式を採用しています。フラッシュメモリの動作モードとしては、プログラムモード / イレースモード / プログラムベリファイモード / イレースベリファイモードがあります。アドレス H'000000 ~ H'05FFFF に対しては、FLMCR1 の PSU ビット、ESU ビット、P ビット、E ビット、PV ビット、EV ビットをセットすることにより各動作モードに遷移します。

フラッシュメモリは、書き込み / 消去を行っている間は読み出すことができません。したがって、フラッシュメモリの書き込み / 消去を制御するプログラム (書き込み制御プログラム) は、内蔵 RAM、あるいは外部メモリ上に置き、実行するようにしてください。書き込み / 消去時の注意については、「18.11 フラッシュメモリの書き込み / 消去時の注意」を参照してください。また、以降の動作説明の中で、FLMCR1 の各ビットのセット / クリア後のウェイト時間のパラメータを記載しています。各ウェイト時間の詳細は「22.3.6 フラッシュメモリ特性」を参照してください。

- 【注】
1. FLMCR1 の SWE、ESU、PSU、EV、PV、E、P ビットのセット / リセットがそれぞれ、当該アドレスエリアのフラッシュメモリ上のプログラムで実行された場合の動作は保証されません。
  2. 書き込み / 消去する際は、FWE = 1 にしてください (FWE = 0 のときは、書き込み / 消去されません)。
  3. 書き込みは消去状態で行ってください。すでに書き込まれたアドレスへの追加書き込みは行わないでください。

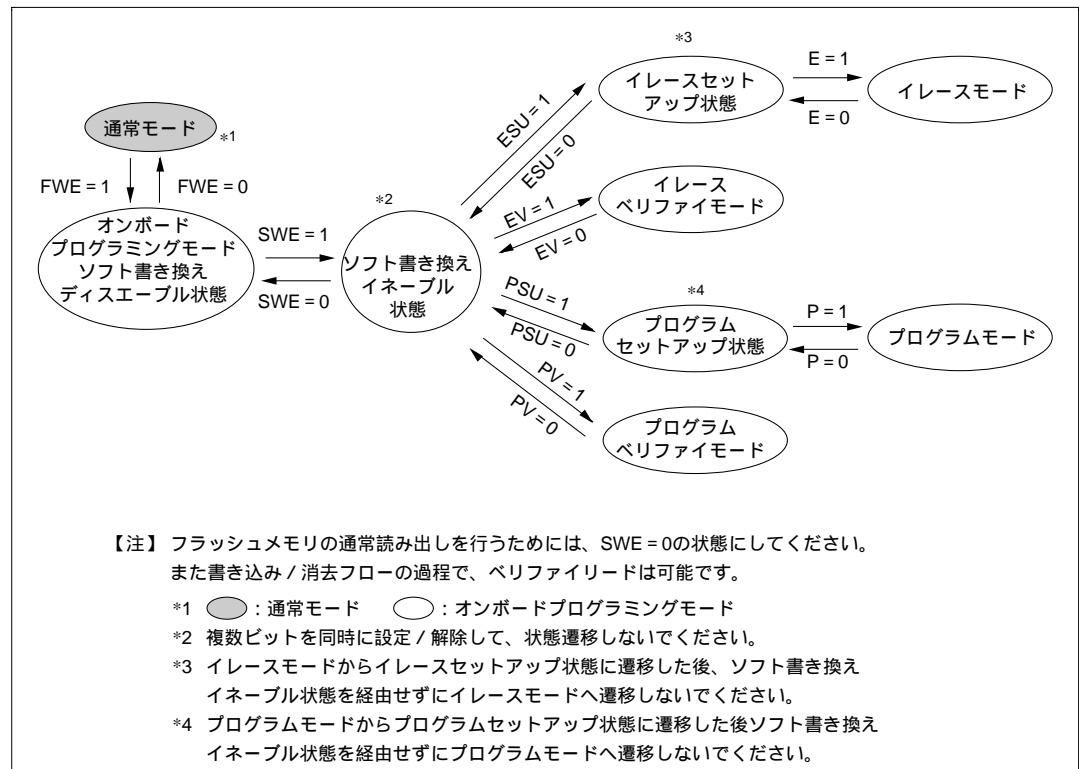


図 18.9 FLMCR1 の各ビット設定による状態遷移

### 18.6.1 プログラムモード

フラッシュメモリへのデータ / プログラムの書き込みは、図 18.10 に示すプログラム / プログラムベリファイフローチャートにしたがって行ってください。このフローチャートに沿って書き込み動作を行えば、デバイスへの電圧ストレスやプログラムデータの信頼性を損なうことなく、フラッシュメモリへデータ / プログラムの書き込みを行うことができます。また、1 回の書き込みは、128 バイト単位で行ってください。

フラッシュメモリコントロールレジスタ 1 (FLMCR1) の各ビットのセット / クリア後のウェイト時間、最大書き込み回数 (N) を「22.3.6 フラッシュメモリ特性」表 22.30 に示します。

フラッシュメモリコントロールレジスタ 1 (FLMCR1) の SWE ビットを 1 にセットした後、( $t_{sswe}$ )  $\mu s$  以上の時間が経過してから、書き込むアドレスに 128 バイトのデータを連続ライトします。ただし、ライトする先頭アドレスの下位 8 ビットは、H'00、H'80 でなければなりません。データ転送はバイト単位で 128 回連続して行います。フラッシュメモリは、プログラムアドレスとプログラムデータをそれぞれフラッシュメモリ内にラッチします。128 バイト以下の書き込みでも 128 バイトのデータ転送を行う必要があり、必要ないアドレスへの書き込みは、データを H'FF にして書き込みを行う必要があります。

次にプログラムの暴走等により過剰時間書き込みを行わないようにするために、ウォッチドッグタイマを設定します。WDT のオーバフロー周期は ( $t_{spsu} + t_{sp} + t_{cp} + t_{cpsu}$ )  $\mu s$  より大きくしてください。その後、FLMCR1 の PSU ビットをセットすることで、プログラムモードへの準備 (プログラムセットアップ) を行います。その後 ( $t_{spsu}$ )  $\mu s$  以上の時

間が経過してから、FLMCR1 の P ビットをセットすることで、動作モードはプログラムモードへ遷移します。P ビットがセットされている時間がフラッシュメモリの書き込み時間となります。一回の書き込み時間を ( $t_{sp}$ )  $\mu s$  の範囲に納まるようにプログラムで設定してください。

また P ビットセット後のウェイト時間は、書き込みの進行状態によって切り換える必要があります。詳細は下記の「プログラム/プログラムベリファイモードの注意点」を参照してください。

## 18.6.2 プログラムベリファイモード

プログラムベリファイモードは、プログラムモードでデータを書き込んだ後、そのデータを読み出し、正しくデータがフラッシュメモリへ書き込まれているかを確認するモードです。

一定の書き込み時間経過後、FLMCR1 の P ビットをクリアします。その後 ( $t_{cp}$ )  $\mu s$  以上の時間が経過してから、PSU ビットをクリアすることでプログラムモードを解除します。プログラムモード解除の後には、ウォッチドッグタイマの設定も解除します。その後 FLMCR1 の PV ビットをセットすることで、動作モードはプログラムベリファイモードへ遷移します。プログラムベリファイモードでは、リードする前にリードするアドレスにデータ HFF をダミーライトしてください。ダミーライトは ( $t_{spv}$ )  $\mu s$  以上の時間が経過してから行ってください。この状態でフラッシュメモリをリード (ベリファイデータは 16 ビットで読み出す) するとラッチしたアドレスのデータが読み出されます。このリード動作は、ダミーライト後、( $t_{spvr}$ )  $\mu s$  以上おいてから行ってください。次に、書き込んだ元データとベリファイデータを比較し、再書き込みデータを演算(図 18.10 参照)し、RAM に転送します。128 バイト分のデータのベリファイが完了後、プログラムベリファイモードを解除し、( $t_{cpv}$ )  $\mu s$  以上の待機時間を置いて、FLMCR1 の SWE ビットをクリアしてください。再度書き込みが必要な場合は、再度プログラムモードに設定し、同様にプログラム/プログラムベリファイシーケンスを繰り返してください。プログラム/プログラムベリファイフローの繰り返しの最大値は、最大書き込み回数 (N) で表わされます。SWE 解除後、( $t_{cswe}$ )  $\mu s$  以上の待機時間を置いてください。

### プログラム/プログラムベリファイフローの注意点

- (1) 本 LSI のプログラム/プログラムベリファイフローでは、128 バイト単位での書き込みアルゴリズムとなります。  
128 バイト単位の書き込みのため、ライトする先頭アドレスの下位 8 ビットは、H'00 または H'80 でなければなりません。
- (2) フラッシュメモリに 128 バイトのデータを連続ライトする際には、バイト単位転送で行います。

また 128 バイト以下のデータを書き込む際にも、128 バイトのデータ転送が必要です。つまり、必要のないアドレスへの書き込みは、データを H'FF にして書き込みを行ってください。

(3) ベリファイデータは、ワード単位で読み出します。

(4) FLMCR1 の P ビットがセットされている期間、書き込みパルスが印加されてフラッシュメモリへの書き込みが実施されます。本 LSI では、デバイスへの電圧ストレスや書き込むデータの信頼性を損なうことがないように、プログラム/プログラムベリファイフローの過程で次のように書き込みパルスを印加してください。

a) 書き込みパルス印加後、プログラムベリファイモードでベリファイリードして 1 が読み出されたビットに対して、もう一度書き込みパルスを印加してください(再書き込み処理)。128 バイトの書き込みデータで、全ての 0 書き込みビットがベリファイリードして 0 が読み出されると、プログラム/プログラムベリファイフローが終了となります。本 LSI では、再書き込み処理によるループ回数が、最大書き込み回数 (N) の最大値以下になることが保証されます。

b) 書き込みパルス印加後、プログラムベリファイモードでベリファイリードして 0 が読み出されたビットに対しては、書き込み完了と判定されます。書き込みが完了したビットに対しては、次の処理が必要です。

- ・プログラム/プログラムベリファイフロー中の早い段階で書き込み完了した場合 再書き込み処理ループ回数が 1~6 回目で書き込み完了した場合は、当該ビットへの追加書き込みを実施してください。また、追加書き込みは、ある再書き込み処理のときに初めてベリファイリードが 0 となったビットのみに実施してください。
- ・プログラム/プログラムベリファイフロー中の遅い段階で書き込み完了した場合 再書き込み処理ループ回数が 7 回目以降で書き込み完了した場合は、当該ビットへの追加書き込みは必要ありません。

c) 128 バイトの中で、他のビットが書き込み未完了の場合、再書き込み処理を実施します。一度書き込みが完了したと判定されたビットでも、それ以降のベリファイリードで 1 が読み出された場合には、当該ビットに対してもう一度書き込みパルスを印加してください。

(5) FLMCR1 の P ビットをセットする期間(書き込みパルス幅)は、プログラム/プログラムベリファイフローの過程で次のように切り替えてください。 ウェイト時間の詳細仕様は、「22.3.6 フラッシュメモリ特性」を参照してください。

項目	記号	項目	記号
Pビットセット後の ウェイト時間	tsp	再書き込みループ回数 (n) が 1~6 回目の場合	tsp30
		再書き込みループ回数 (n) が 7 回目以降の場合	tsp200
		追加書き込み処理の場合*	tsp10

【注】\* 追加書き込み処理は、再書き込みループ回数 (n) が 1~6 回目の場合のみ必要となります。

(6) 本 LSI のプログラム / プログラムベリファイのフローチャートを図 18.10 に示します。

上記の注意点を網羅するためには、再書き込み処理を実施するビット、追加書き込み処理を実施するビットは下記の演算によって決定する必要があります。

書き込みの進行に応じて、再書き込みのデータおよび追加書き込みのデータは変化しますので、RAM 上に次のデータ格納エリア (各 128 バイト) を準備することを推奨します。

再書き込みデータ演算表

(D)	書き込みパルス印加後の ベリファイリード結果 (V)	(X) 演算結果	コメント
0	0	1	書き込み完了のため、再書き込み処理は実施しない
0	1	0	書き込み未完了のため、再書き込み処理を実施する
1	0	1	
1	1	1	消去状態のまま、何も実施しない

【記号説明】

(D) : 書き込みを実施するビットの元データ

(X) : 再書き込みを実施するビットのデータ

追加書き込みデータ演算表

(X')	書き込みパルス印加後の ベリファイリード結果 (V)	(Y) 演算結果	コメント
0	0	0	書き込みパルス印加により書き込み完了したと判定 追加書き込み処理を実施する
0	1	1	書き込みパルス印加により書き込みは未完了 追加書き込み処理は実施しない
1	0	1	既に書き込みは完了している 追加書き込み処理は実施しない
1	1	1	消去状態のまま、何も実施しない

## 【記号説明】

(Y) : 追加書き込みを実施するビットのデータ

(X') : ある再書き込みループで再書き込みを実施するビットのデータ

(7) 本LSIのプログラム/プログラムベリファイフローの過程では、追加書き込み処理を実施する必要があります。

しかし、128 バイト単位の書き込みが一度終了した後、同一のアドレスエリアに追加で書き込みを行わないでください。書き替えを実施する場合は、必ず一度消去を行ってから書き込みを実施してください。一度プログラム/プログラムベリファイが終了したアドレスへ追加書き込みを実施した場合、読み出しなど正常動作は保証されませんので注意してください。

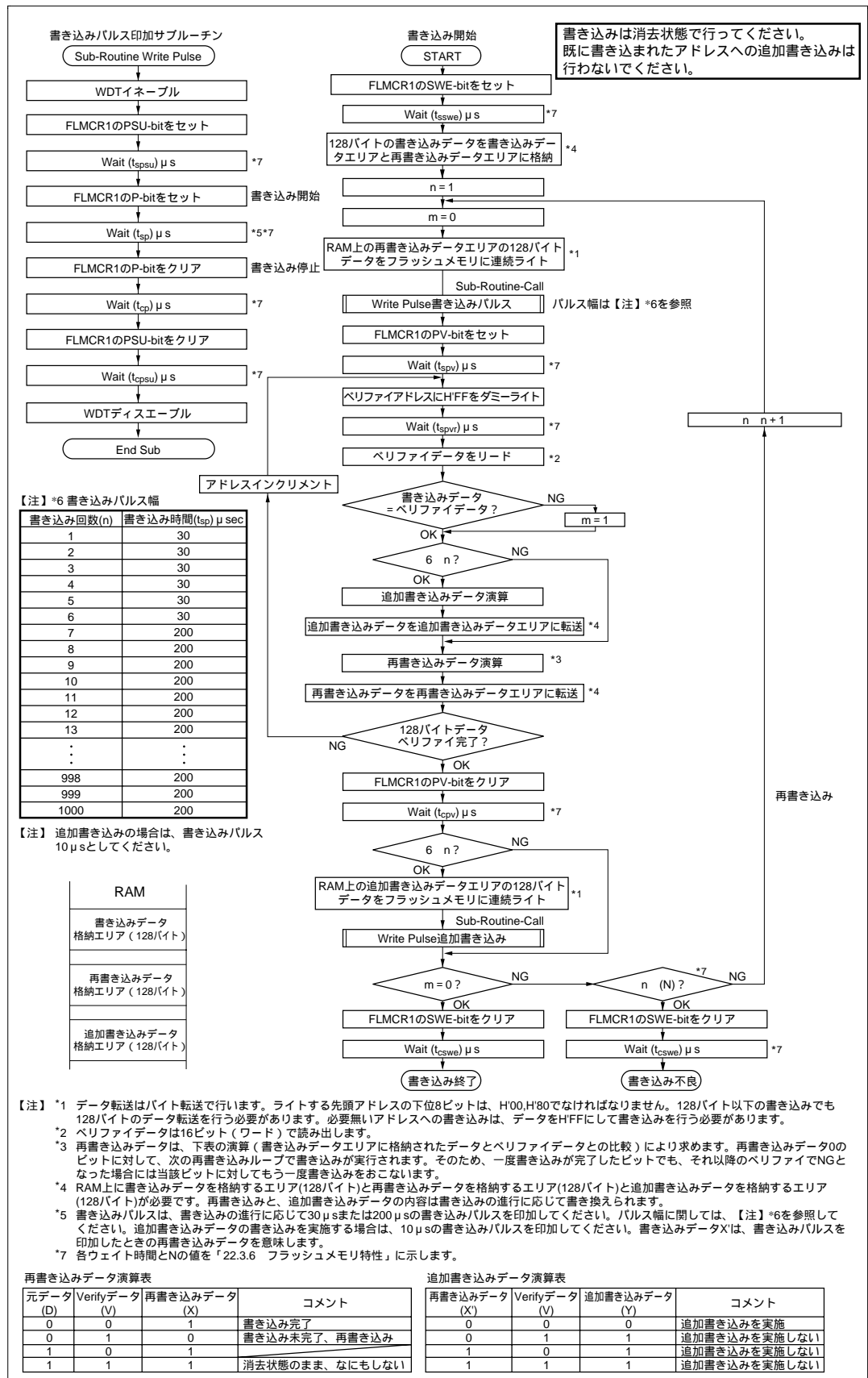


図 18.10 プログラム/プログラムベリファイフロー (128バイト書き込み)

### 18.6.3 イレースモード

フラッシュメモリの消去は1ブロックごとに、図 18.11 に示す1ブロック消去のフローチャート（単一ブロック消去）に沿って行ってください。

フラッシュメモリコントロールレジスタ 1 (FLMCR1) の各ビットのセット/クリア後のウェイト時間および最大消去回数 (N) を「22.3.6 フラッシュメモリ特性」の表 22.30 に示します。

フラッシュメモリ内容の消去は、FLMCR1 の SWE ビットを 1 にセット後、 $(t_{sswe}) \mu s$  以上の時間が経過してから、消去ブロック指定レジスタ 1、2 (EBR1、EBR2) で消去するフラッシュメモリのエリアを 1 ビット設定してください。次にプログラムの暴走などにより過剰時間消去を行わないようにするために、ウォッチドッグタイマを設定します。WDT のオーバフロー周期は  $(t_{se}) ms + (t_{sesu} + t_{ce} + t_{cesu}) \mu s$  より大きく設定してください。その後、FLMCR1 の ESU ビットをセットすることで、イレースモードへの準備（イレースセットアップ）を行います。その後、 $(t_{sesu}) \mu s$  以上の時間が経過後、FLMCR1 の E ビットをセットすることで、動作モードはイレースモードへ遷移します。E ビットが設定されている時間が消去時間となり、消去時間は  $(t_{se}) ms$  を超えないようにしてください。

【注】フラッシュメモリの消去において、消去を開始する前にプレライト（消去するメモリのデータをすべて 0 にする）を行う必要はありません。

### 18.6.4 イレースベリファイモード

イレースベリファイモードは、メモリを消去した後データを読み出し、正常に消去されているかどうかを確認するモードです。

一定の消去時間経過後、FLMCR1 の E ビットをクリアします。その後、 $(t_{ce}) \mu s$  以上の時間が経過してから ESU ビットをクリアすることでイレースモードを解除します。イレースモード解除の後には、ウォッチドッグタイマの設定も解除します。その後、FLMCR1 の EV ビットをセットすることで、動作モードはイレースベリファイモードへ遷移します。イレースベリファイモードでは、リードする前にリードするアドレスにデータ H'FF をダミーライトしてください。ダミーライトは  $(t_{sev}) \mu s$  以上の時間が経過してから行ってください。この状態でフラッシュメモリをリード（ベリファイデータは 16 ビットで読み出す）するとラッチしたアドレスのデータが読み出されます。このリード動作は、ダミーライト後、 $(t_{sevr}) \mu s$  おいてから行ってください。読み出したデータが消去（データがすべて 1）されていた場合、次のアドレスをダミーライトし、イレースベリファイを行います。読み出したデータが未消去の場合、再度イレースモードに設定し、同様にイレース/イレースベリファイシーケンスを繰り返します。

イレース/イレースベリファイの繰り返しの最大値は、最大消去回数 (N) によって表わされます。ベリファイ完了後、イレースベリファイモードを解除し、 $(t_{cev}) \mu s$  以上の待機時間を置いてください。通常モードに遷移するには、FLMCR1 の SWE ビットを解除し、 $(t_{cswe}) \mu s$  以上の待機時間を置いてください。



複数ブロックを消去する場合は、次に消去するブロックのエリアを EBR1 / EBR2 によって1ビットのみ設定し、同様にイレース/イレースベリファイシーケンスを繰り返します。

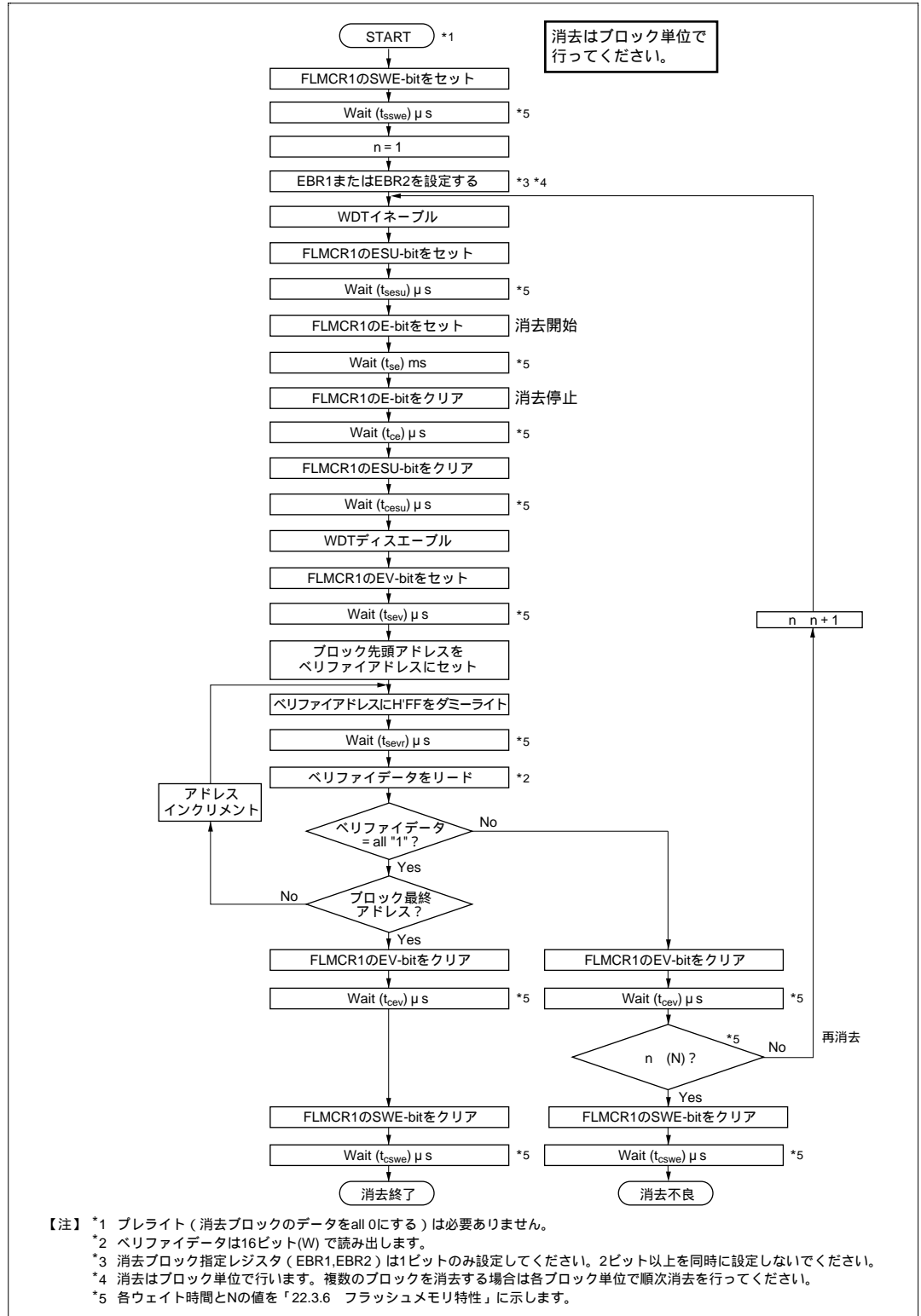


図 18.11 イレース/イレースベリファイフロー（単一ブロック消去）

## 18.7 プロテクト

フラッシュメモリに対する書き込み・消去プロテクトは、ソフトウェアプロテクトとハードウェアプロテクトとエラープロテクトの3種類あります。

### 18.7.1 ハードウェアプロテクト

ハードウェアプロテクトとは、フラッシュメモリに対する書き込み/消去が強制的に禁止、中断された状態のことで、フラッシュメモリコントロールレジスタ1 (FLMCR1)、消去ブロック指定レジスタ1 (EBR1)、消去ブロック指定レジスタ2 (EBR2) の設定はリセットされます。エラープロテクト状態では、FLMCR1、FLMCR2、EBR1、EBR2 の設定は保持し、PビットおよびEビットはセット可能ですが、プログラムモードおよびイレースモードへは遷移しません。(表 18.9 参照)

表 18.9 ハードウェアプロテクト

項目	説明	機能		
		書き込み	消去	ベリファイ
FWE 端子 プロテクト	・ FWE 端子に Low レベルが入力されているときには、FLMCR1、EBR1、EBR2 は初期化され、書き込み/消去プロテクト状態になります。	不可 <sup>*1</sup>	不可 <sup>*3</sup>	不可
リセット、 スタンバイ プロテクト	・ リセット (WDT によるオーバーフローリセットも含む) およびスタンバイ時は、FLMCR1、FLMCR2、EBR1、EBR2 は初期化され、書き込み/消去プロテクト状態になります。 ・ $\overline{\text{RES}}$ 端子によるリセットでは、電源投入後発振が安定するまで $\overline{\text{RES}}$ 端子を Low レベルに保持しないとリセット状態になりません。また、動作中のリセットは AC 特性に規定した $\overline{\text{RES}}$ パルス幅の間 $\overline{\text{RES}}$ 端子を Low レベルに保持してください。 <sup>*4</sup>	不可	不可 <sup>*3</sup>	不可
エラー プロテクト	・ フラッシュメモリへの書き込み/消去中にマイコン動作の異常を検出 (エラー発生 (FLER = 1)) した場合にエラープロテクトが有効となります。このとき FLMCR1、EBR1、EBR2 の設定は保持しますが、エラーが発生した時点で、書き込み/消去を強制的に中断します。エラープロテクトの解除は $\overline{\text{RES}}$ 端子によるリセットおよび WDT リセットまたはハードウェアスタンバイのみです。	不可	不可 <sup>*3</sup>	可 <sup>*2</sup>

【注】 <sup>\*1</sup> フラッシュメモリとオーバーラップした RAM エリアは除きます。

- \*2 書き込み中の 128 バイトのプログラムベリファイは可能です。  
消去中の 1 ブロックのイレースベリファイは可能です。
- \*3 全ブロックが消去不可となり、ブロック別の指定はできません。
- \*4 「4.2.2 リセットシーケンス」および「18.11 フラッシュメモリの書き込み/消去時の注意」を参照してください。H8/3068F-ZTAT では、動作中のリセット期間は最低 20 システムクロックが必要です。

## 18.7.2 ソフトウェアプロテクト

ソフトウェアプロテクトは、消去ブロック指定レジスタ 1 (EBR1)、消去ブロック指定レジスタ 2 (EBR2)、RAM コントロールレジスタ (RAMCR) の RAMS ビットを設定することで行えます。ソフトウェアプロテクトでは、フラッシュメモリコントロールレジスタ 1 (FLMCR1) の P ビットまたは E ビットをセットしても、プログラムモードまたはイレースモードへは遷移しません。(表 18.10 参照)

表 18.10 ソフトウェアプロテクト

項目	説明	機能		
		書き込み	消去	ベリファイ
ブロック指定プロテクト	<ul style="list-style-type: none"> <li>・消去ブロック指定レジスタ 1 (EBR1)、消去ブロック指定レジスタ 2 (EBR2) の設定により*2、ブロック毎に消去プロテクトが可能。ただし書き込みに対するプロテクトは無効です。</li> <li>・EBR1、EBR2 を H'00 に設定すると全ブロックが消去プロテクト状態になります。</li> </ul>	-	不可	可
エミュレーションプロテクト	<ul style="list-style-type: none"> <li>・RAM コントロールレジスタ (RAMCR) の RAMS ビットを 1 にセットすることにより、全ブロックの書き込み/消去プロテクト状態になります。</li> </ul>	不可*1	不可*3	可

【注】 \*1 フラッシュメモリとオーバーラップした RAM エリアへの書き込みは可能です。

\*2 EBR1、EBR2 のビットは消去時以外は H'00 にしてください。

\*3 全ブロックが消去不可となり、ブロック別の指定はできません。

### 18.7.3 エラープロテクト

エラープロテクトは、フラッシュメモリへの書き込み/消去中<sup>\*1</sup>のマイコンの暴走や書き込み/消去アルゴリズムに沿っていない動作をした場合に発生する異常を検出し、書き込み/消去動作を強制的に中断するプロテクトです。書き込み/消去動作を中断することで、過剰書き込みや過剰消去によるフラッシュメモリへのダメージを防止します。

フラッシュメモリへの書き込み/消去中にマイコンが異常動作すると、FLMCR2 の FLER ビットが1にセットされ、エラープロテクト状態へ遷移します。このとき、FLMCR1、FLMCR2、EBR1、EBR2の設定<sup>\*3</sup>は保持されますが、エラーが発生した時点でプログラムモードまたはイレースモードは強制的に中断されます。Pビット、Eビットの再設定を行ってもプログラムモードまたはイレースモードへ遷移することはできません。ただし、PVビット、EVビットの設定は有効なので、ベリファイモードへの遷移は可能です。<sup>\*2</sup>

FLER ビットのセット条件は、

- (1) 書き込み/消去中に当該アドレスエリアのフラッシュメモリを読み出したとき(ベクタリードおよび命令フェッチを含む)
- (2) 書き込み/消去中の例外処理(リセット不当命令、トラップ命令、ゼロ除算時の例外処理は除く)開始直後
- (3) 書き込み/消去中に SLEEP 命令(ソフトウェアスタンバイを含む)を実行したとき
- (4) 書き込み/消去中にバス権を解放したとき

エラープロテクト解除は、リセット ( $\overline{\text{RES}}$  端子、WDT リセット) またはハードウェアスタンバイのみで行われます。

**【注】** \*1 FLMCR1 の P ビットまたは E ビットが1にセットされた状態です。この状態では NMI 入力が禁止されますので、注意してください。

\*2 書き込み中の 128 バイトのプログラムベリファイは可能です。

また消去中の 1 ブロックのイレースベリファイは可能です。

\*3 FLMCR1、EBR1、EBR2 へのライトは可能です。

ただしエラープロテクト状態でソフトウェアスタンバイモードに遷移した場合は、レジスタはイニシャライズされます。

図 18.12 にフラッシュメモリの状態遷移図を示します。

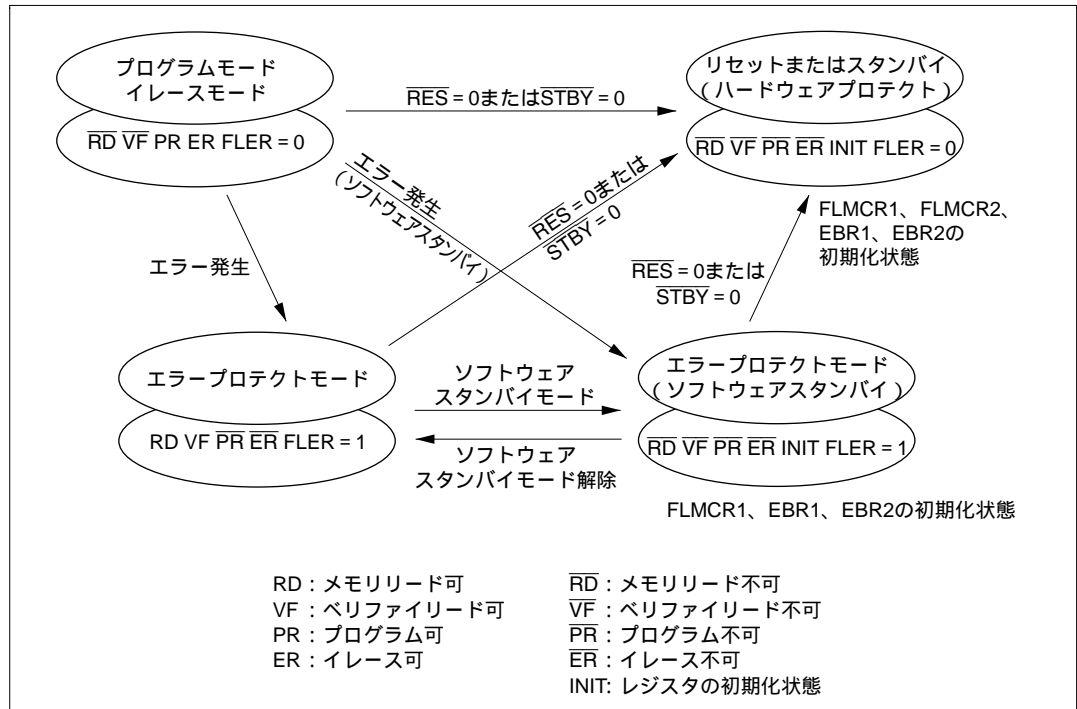


図 18.12 フラッシュメモリの状態遷移図 (モード 5、7 (内蔵 ROM が有効) で FWE 端子に High レベル印加時)

エラープロテクト機能は、FLER ビットのセット条件以外の異常動作に対しては無効です。また、このプロテクト状態に遷移するまでに相当な時間が経過している場合は、すでにフラッシュメモリにダメージを与えている可能性もあります。したがって、この機能ではフラッシュメモリへのダメージを完全に防止することはできません。

このため、このような異常動作を防止するためには、フラッシュライトイネーブル (FWE) が印加された状態で書き込み / 消去アルゴリズムに従って正しく動作させること、およびマイコンの異常をウォッチドッグタイマ等でマイコン内外で常に監視することが必要です。また、このプロテクトモードへ遷移した時点でのフラッシュメモリは誤書き込み、誤消去の状態であったり、強制停止によって書き込みや消去が不十分な場合があります。このような場合、必ずブートモードによる強制復帰 (プログラムの再書き込み) を行ってください。ただし、過剰書き込み、過剰消去によってブートモードが正常に起動されない場合があります。

## 18.8 RAM によるフラッシュメモリのエミュレーション

RAM でフラッシュメモリに書き替えるデータをリアルタイムにエミュレートするために、RAM コントロールレジスタ (RAMCR) で設定したフラッシュメモリのエリアに RAM の一部を重ね合わせて使うことができます。RAMCR の設定後はフラッシュメモリのエリアとフラッシュメモリに重ね合わせた RAM エリアの 2 エリアからアクセスできます。エミュレーション可能なモードは、ユーザモード、およびユーザプログラムモードです。図 18.13 にフラッシュメモリのリアルタイムな書き替えをエミュレートする例を示します。

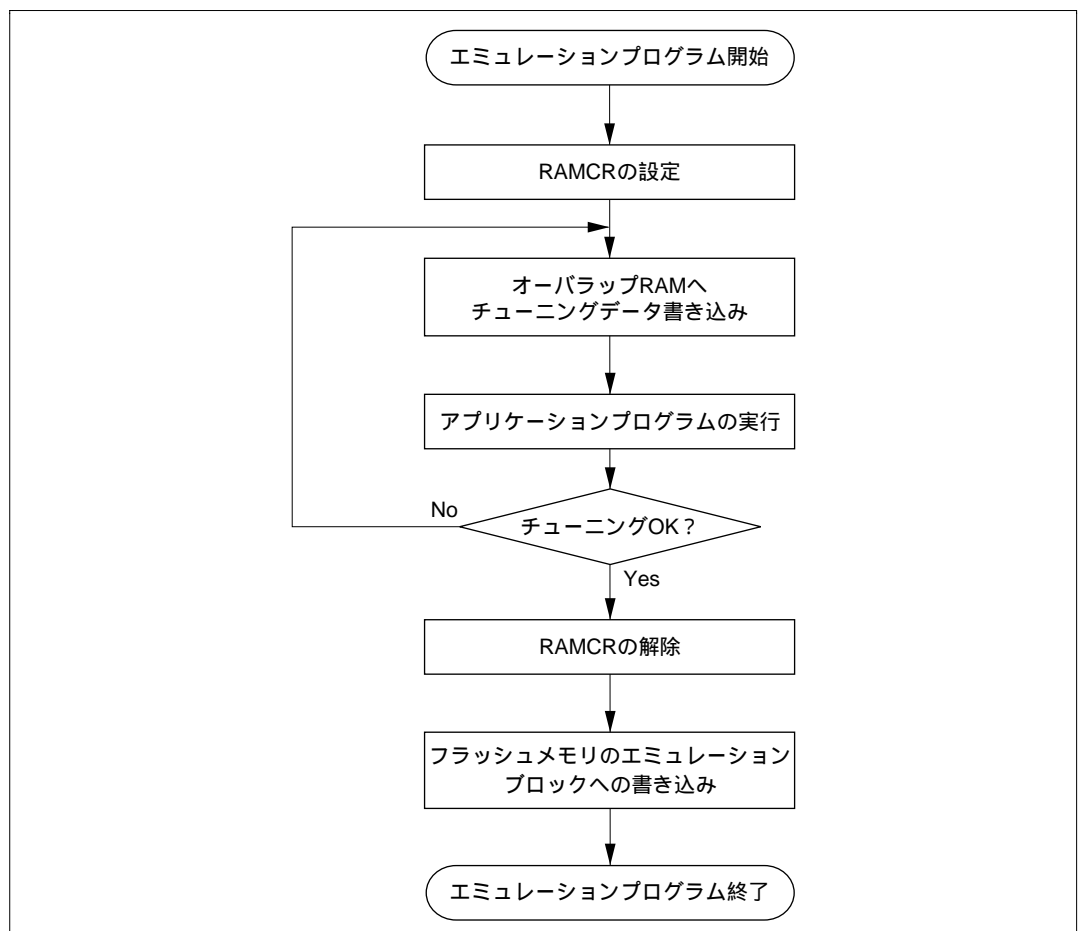


図 18.13 RAM によるエミュレーションフロー

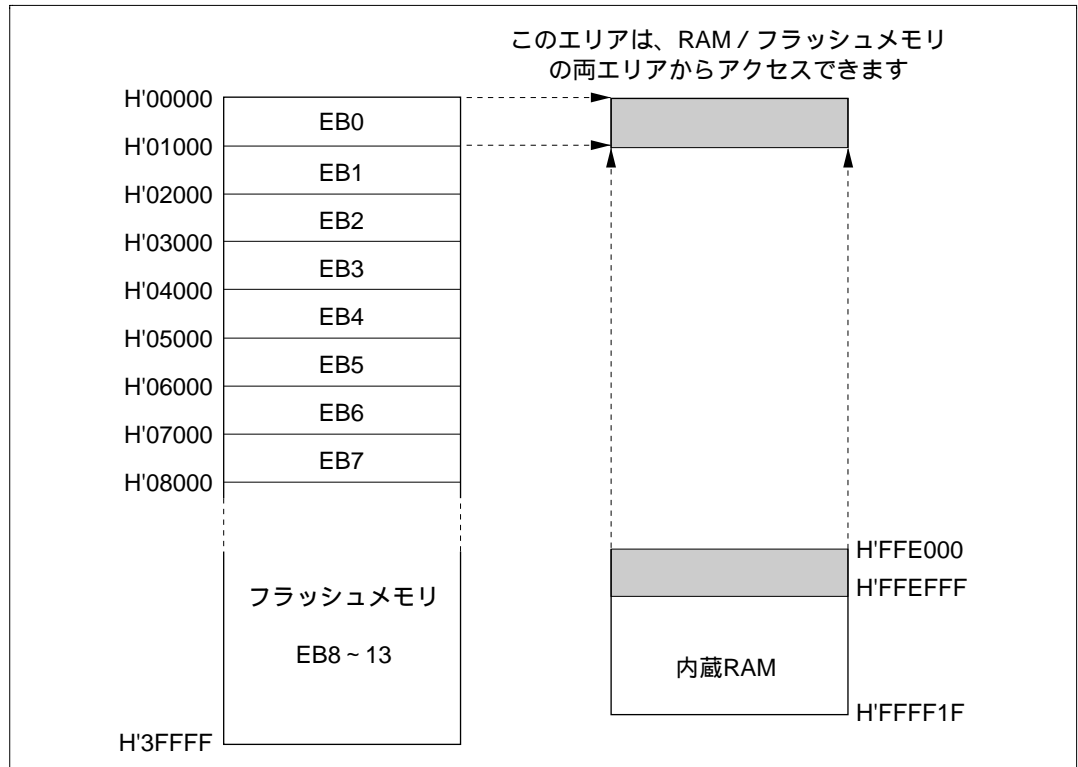


図 18.14 RAMのオーバーラップ動作例

フラッシュメモリのブロックエリア (EB0) をオーバーラップさせる例

- リアルタイムな書き替えを必要とするエリア (EB0) に RAM の一部をオーバーラップさせるには、RAMCR の RAMS ビット、RAM2~0 ビットを 1、0、0、0 に設定してください。
- リアルタイムな書き替えは、オーバーラップさせた RAM を使って行います。
- 書き替えデータ確定後、RAMS ビットをクリアして RAM のオーバーラップを解除します。
- オーバーラップさせた RAM に書き込まれたデータをフラッシュメモリ空間 (EB0) に書き込みます。

- 【注】
- RAMS ビットを 1 にセットすると RAM2~0 の値にかかわらず、フラッシュメモリの全ブロックに対して書き込み / 消去プロテクトが有効となります (エミュレーションプロテクト)。この状態では FLMCR1 の P ビットまたは E ビットをセットしてもプログラムモード、イレースモードへは遷移しません。フラッシュメモリエリアに実際に書き込み / 消去を行う場合は RAMS ビットを 0 にクリアしてください。
  - RAM によるフラッシュメモリのエミュレーション機能を使用中に、消去アルゴリズムに沿ったソフトウェアを実行しても RAM エリアを消去することはできません。
  - ブロックエリア (EB0) はベクタテーブルを含みます。RAM エミュレーションする場合、オーバーラップ RAM にはベクタテーブルが必要となります。

4. オンボードプログラミングモードと同様に、FWE の印加 / 解除時は誤書き込み / 誤消去を防止するための注意が必要です。特に FWE の印加中のプログラム暴走等による誤書き込み / 誤消去を防止するため、エミュレーション機能を使用中でも FLMCR1 の PSU ビット、P ビット、ESU ビット、E ビットを 1 にセットしているときはウォッチドッグタイマを設定してください。
5. エミュレーション機能を使用しているときも、通常書き込み / 消去と同様に、FLMCR1 の P ビット、E ビットを 1 にセットしている状態では NMI 入力が禁止されます。

なお、P ビットおよび E ビットはリセット時（ウォッチドッグタイマのリセットを含む）、スタンバイモード時、FWE 端子に High レベルが入力されていないとき、または FWE 端子に High レベル印加状態で FLMCR1 の SWE が 0 のときにクリアされます。



## 18.9 NMI 入力禁止の条件

フラッシュメモリへの書き込み / 消去中 (FLMCR1 の P ビットまたは E ビットがセット)、およびブートモードでのブートプログラム実行中\*<sup>1</sup>は書き込み / 消去中を最優先とするため NMI 入力を含むすべての割り込みを禁止してください。

これは以下のような動作状態を回避することを目的としています。

- (1) 書き込み、消去中に割り込みが発生することにより、書き込み / 消去アルゴリズムに違反し、正常な動作が保証できなくなる。
- (2) 書き込み / 消去中の割り込み例外処理ではベクタリードが正常にできない\*<sup>2</sup>ため、結果としてマイコンが暴走してします。
- (3) ブートプログラム実行中に割り込みが発生すると正常なブートモードのシーケンスが実行できなくなる。

以上のような理由から、オンボードプログラミングモードにおいてのみ例外的に NMI 割り込み入力を禁止する条件が存在しますが、これによって正常な書き込み、消去およびマイコン動作が保証されるものではありません。

このため、フラッシュメモリへの書き込み / 消去を行う場合、マイコンの内部と外部で NMI を含むすべての割り込み要求を (例外処理、バス解放) 禁止する必要があります。また、エラープロテクト状態および RAM によるフラッシュメモリのエミュレーション中で FLMCR1 の P ビットまたは E ビットが保持された状態でも NMI 割り込みは禁止状態となります。

**【注】** \*<sup>1</sup> 内蔵 RAM のブートプログラムエリアに分岐するまでの期間を示します。(この分岐は、ユーザプログラムの転送が完了した直後に発生します。) このため、RAM エリアに分岐したあとは、書き込み / 消去以外の状態では NMI 入力が可能となります。

したがってユーザプログラムによる初期書き込み (ベクタテーブルおよび NMI 処理プログラム等の書き込み) が完了するまでは、マイコン内外で割り込み要求を禁止する必要があります。

\*<sup>2</sup> この場合、以下の二つの理由によってベクタリードが正常に行われません。

- ・ 書き込み、消去中 (FLMCR1 の P ビットまたは E ビットがセット) にフラッシュメモリのリードを行っても正しい値を読み出すことはできません (値は不定)。
- ・ 割り込みベクタテーブルに値がまだ書き込まれていない場合、割り込み例外処理が正しく実行されません。

## 18.10 フラッシュメモリのPROMモード

本LSIでは、フラッシュメモリの書き込み/消去可能なモードとして、オンボードプログラミングモード以外にPROMモードがあります。PROMモードでは、日立256kバイトフラッシュメモリ内蔵マイコンのデバイスタイプをサポートしている汎用PROMライタを用いて、内蔵ROMに自由にプログラムを書き込むことができます。

### 18.10.1 ソケットアダプタとメモリマップ

PROMライタを用いたPROMモードでは、メモリ読み出し（ベリファイ）、書き込み、フラッシュメモリ初期化（全面消去）が可能です。その際には、汎用PROMライタに専用の変換ソケットアダプタを取り付けて行います。表18.11にソケットアダプタの型名を示します。本LSIのPROMモードでは、表18.11のソケットアダプタを必ず使用してください。

表 18.11 H8/3068F-ZTATソケットアダプタ型名

製品型名	パッケージ名	ソケットアダプタ型名	メーカー
HD64F3068F	100ピンQFP (FP-100B)	ME3064ESHF1H	ミナト エレクトロニクス(株)
HD64F3068TE	100ピンTQFP (TFP-100B)	ME3064ESNF1H	
HD64F3068F	100ピンQFP (FP-100B)	HF306BQ100D4001	データ・アイ・オー・ ジャパン(株)
HD64F3068TE	100ピンTQFP (TFP-100B)	HF306BT100D4001	

また、下記にPROMモード時のメモリマップを示します。



図 18.15 PROMモード時のメモリマップ

### 18.10.2 PROMモード使用時の注意事項

- (1) PROMモード時の128バイト書き込み単位への書き込みは、1回のみとします。既に書き込まれたアドレスへの書き替えは、消去を行った後に書き込みを実施してください。

- (2) オンボード書き込み/消去を行ったデバイスに対して、それを PROM ライタを用いて書き替えを行う場合は、一度消去を行った後に書き込みを実施することを推奨します。
- (3) 日立出荷品の初期状態は、消去状態です。消去来歴不明サンプルに対しては、初期化（消去）レベルをチェック・補正するために消去の実施を推奨します。
- (4) 本 LSI では、汎用 EPROM のような製品識別モードをサポートしていませんので、PROM ライタにデバイス名を自動設定することができません。
- (5) 本 LSI の PROM モードに適合する PROM ライタおよびそのプログラムバージョンに関しては、ソケットアダプタの取り扱い説明書等を参照してください。

## 18.11 フラッシュメモリの書き込み / 消去時の注意

オンボードプログラミングモード、RAM エミュレーション機能および PROM モード使用時の注意事項を示します。

(1) 規定された電圧、タイミングで書き込み / 消去を行ってください。

定格以上の電圧を印加した場合、製品の永久破壊にいたることがあります。

PROM ライタは、日立製 512k バイトフラッシュメモリ内蔵マイコンデバイスタイプ「F-ZTAT512」をサポートしているものを使用してください。

(2) 電源投入 / 切断時の注意 (図 18.16 ~ 18.18 参照)

FWE 端子への High レベル印加は  $V_{CC}$  確定後に行ってください。また  $V_{CC}$  を切断する前に FWE 端子を Low レベルにしてください。

$V_{CC}$  電源の印加 / 切断時は FWE 端子を Low レベルに固定し、フラッシュメモリをハードウェアプロテクト状態にしてください。

この電源投入 / 切断のタイミングは、停電等による電源の切断、再投入時にも満足するようにしてください。このタイミングが守られない場合は、マイコンの暴走等によって過剰書き込み、過剰消去となりメモリセルが正常に動作しなくなることがありますので充分注意してください。

(3) FWE 端子の印加 / 解除の注意

FWE 端子の印加は、マイコン動作が確定した状態で行ってください。マイコンが動作確定状態を満足しない場合は、FWE 端子を Low レベルに固定し、プロテクト状態としてください。

FWE 端子印加 / 解除では、フラッシュメモリへの誤書き込み / 誤消去を防止するため、以下に示すような注意が必要です。

$V_{CC}$  電圧が定格電圧の範囲で安定している状態で FWE 端子に印加してください。

マイコンの  $V_{CC}$  電圧が定格電圧を満足しない状態で FWE 端子に印加すると、マイコン動作が不確定の状態であることから、フラッシュメモリに誤って書き込み / 消去を行ってしまう可能性があります。

発振が安定している状態 (発振安定時間経過後) で FWE 端子に印加してください。

$V_{CC}$  電源投入時では、発振安定時間の間、 $\overline{RES}$  端子を Low レベルに保持した後に、FWE 端子に印加してください。発振が停止した状態や不安定な状態で、FWE 端子の印加は行わないでください。

ブートモードでは、FWE 端子の印加 / 解除はリセット中に行ってください。

ブートモードへの遷移では、FWE=High レベル入力と  $MD_2 \sim MD_0$  の設定は  $\overline{RES}$  入力が Low 期間中に行ってください。このとき FWE と  $MD_2 \sim MD_0$  入力は、リセット解除タイミングに対してモードプログラミングセットアップ時間 ( $t_{MDS}$ ) を満足する必要があります。

ブートモードから他のモードへ遷移する場合も  $\overline{\text{RES}}$  解除タイミングに対して、モードプログラミングセットアップ時間が必要です。

動作中のリセットでは、最低 20 システムクロックの間、 $\overline{\text{RES}}$  端子を Low レベルにする必要があります。

ユーザプログラムモードでは、 $\overline{\text{RES}}$  入力にかかわらず、FWE=High / Low の切り換えが可能です。

また、フラッシュメモリ上でプログラム実行中でも、FWE 入力の切り換えが可能です。プログラムが暴走していない状態で FWE を印加してください。

FWE 端子の印加時は、ウォッチドッグタイマ等でプログラム実行状態を監視することが必要です。

FWE 端子の解除は FLMCR1 の SWE、ESU、PSU、EV、PV、E、P ビットをクリアした状態で行ってください。

FWE 端子の印加 / 解除時に、誤って SWE、ESU、PSU、EV、PV、E、P ビットをセットしないでください。

(4) FWE 端子に常時 High レベルを印加しないでください。

プログラム暴走等による誤書き込み / 誤消去を防止するため、FWE 端子に High レベルを印加するのは、フラッシュメモリに書き込み / 消去を行うときのみ (RAM によるフラッシュメモリのエミュレーション実行時も含む) としてください。また、FWE 端子に常時 High レベルを印加するようなシステム構成は避けてください。High レベル印加中においても、過剰書き込み / 過剰消去にならないように、ウォッチドッグタイマを起動し、プログラムの暴走等に対応できるようにしてください。

(5) フラッシュメモリへの書き込み / 消去は推奨するアルゴリズムにしたがって行ってください。

推奨アルゴリズムでは、デバイスへの電圧ストレスあるいはプログラムデータの信頼性を損なうことなく書き込み / 消去を行うことができます。

FLMCR1 の P ビットおよび E ビットをセットするときは、プログラムの暴走等に備えてあらかじめウォッチドッグタイマを設定してください。

また、P ビットおよび E ビットをセットしている間に、MOV 命令等でフラッシュメモリ空間へのアクセスを行うことは禁止されています。

(6) SWE ビットのセット / クリアは、フラッシュメモリ上のプログラム実行中に行わないでください。

フラッシュメモリ上のプログラム実行とデータの読み出しは、SWE ビットをクリアした後に行ってください。

SWE ビットをセットするとフラッシュメモリのデータを書き替えることができますが、ベリファイ (プログラム / イレース中のベリファイ) 以外の目的で、フラッシュメモリを

アクセスしないでください。

FWE 端子に High レベルを入力した状態で、RAM によるエミュレーション機能を使用する場合も同様に、フラッシュメモリ上のプログラム実行とデータの読み出しは、SWE ビットをクリアした後に行ってください。ただし、フラッシュメモリ空間とオーバーラップした RAM エリアについては、SWE ビットのセット/クリアに関わらずリード/ライト可能です。

また、SWE ビットをクリアした後は、ウェイト時間が必要です。詳細は「22.3.6 フラッシュメモリ特性」表 22.30 を参照してください。

(7) フラッシュメモリのプログラム中または消去中に割り込みを使用しないでください。

FWE 端子に印加している状態では、書き込み/消去動作 (RAM によるエミュレーションを含む) を最優先とするため、NMI を含む全ての割り込み要求を禁止してください。

また、バス権の解放も禁止する必要があります。

(8) 追加書き込みは行わないでください。書き替えは消去後に行ってください。

オンボードプログラミングでは、128 バイトの書き込み単位ブロックへの書き込みは、1 回のみとしてください。書き替えは、この書き込み単位ブロックが全て消去された状態で行ってください。

(9) 書き込み前に、正しく PROM ライタに装着されていることを必ず確認してください。

PROM ライタのソケット、ソケットアダプタおよび製品それぞれのインデックスが正しく一致していないと、過剰電流が流れ、製品が破壊されることがあります。

(10) 書き込み中はソケットアダプタや製品に手を触れないでください。

接触不良などにより、書き込み不良になることがあります。

(11) 書き込み、消去、ベリファイモードから通常モードに遷移した後に、読み出しを行う場合は 100  $\mu$ s 以上の待ち時間をおいてください。

(12) フラッシュメモリを制御するレジスタ (FLMCR1、FLMCR2、EBR1、EBR2、RAMCR) へのアクセスはバイトアクセスとしてください。

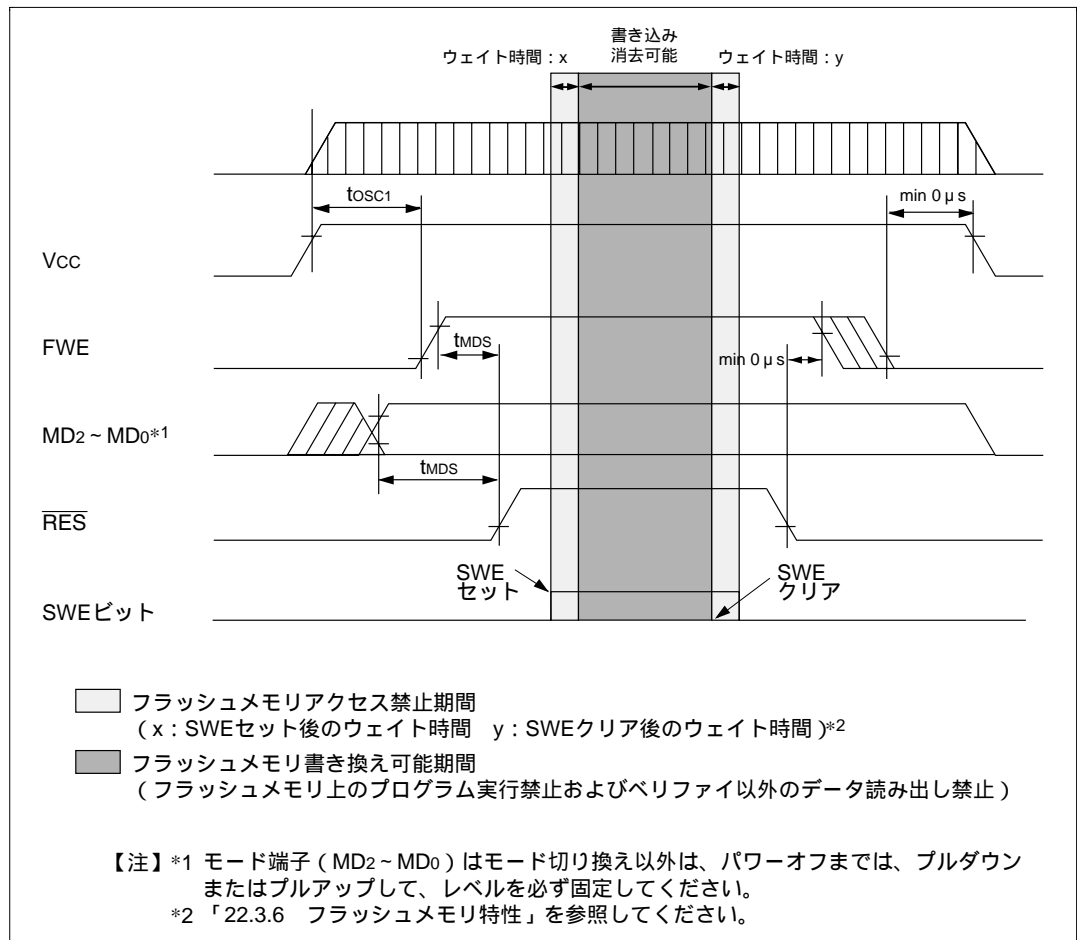


図 18.16 電源投入 / 切断タイミング (ブートモード)

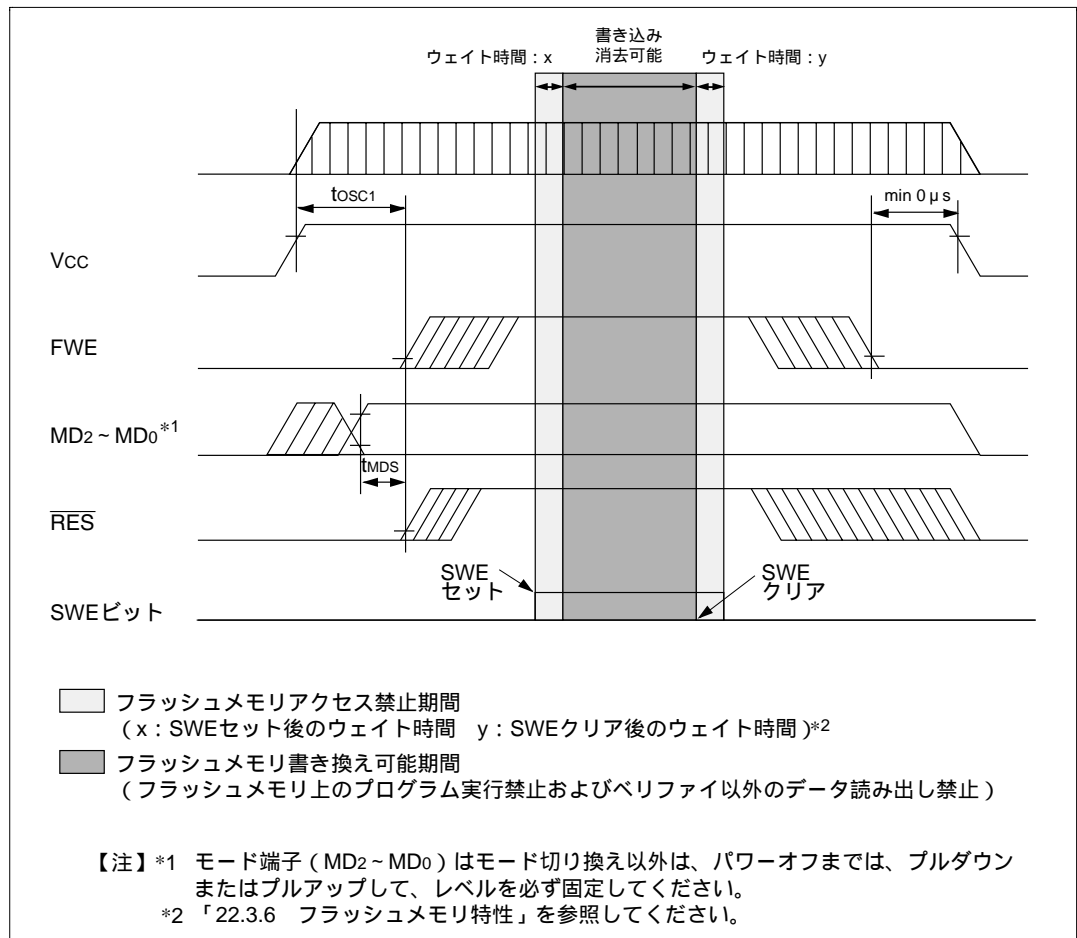


図 18.17 電源投入 / 切断タイミング (ユーザプログラムモード)



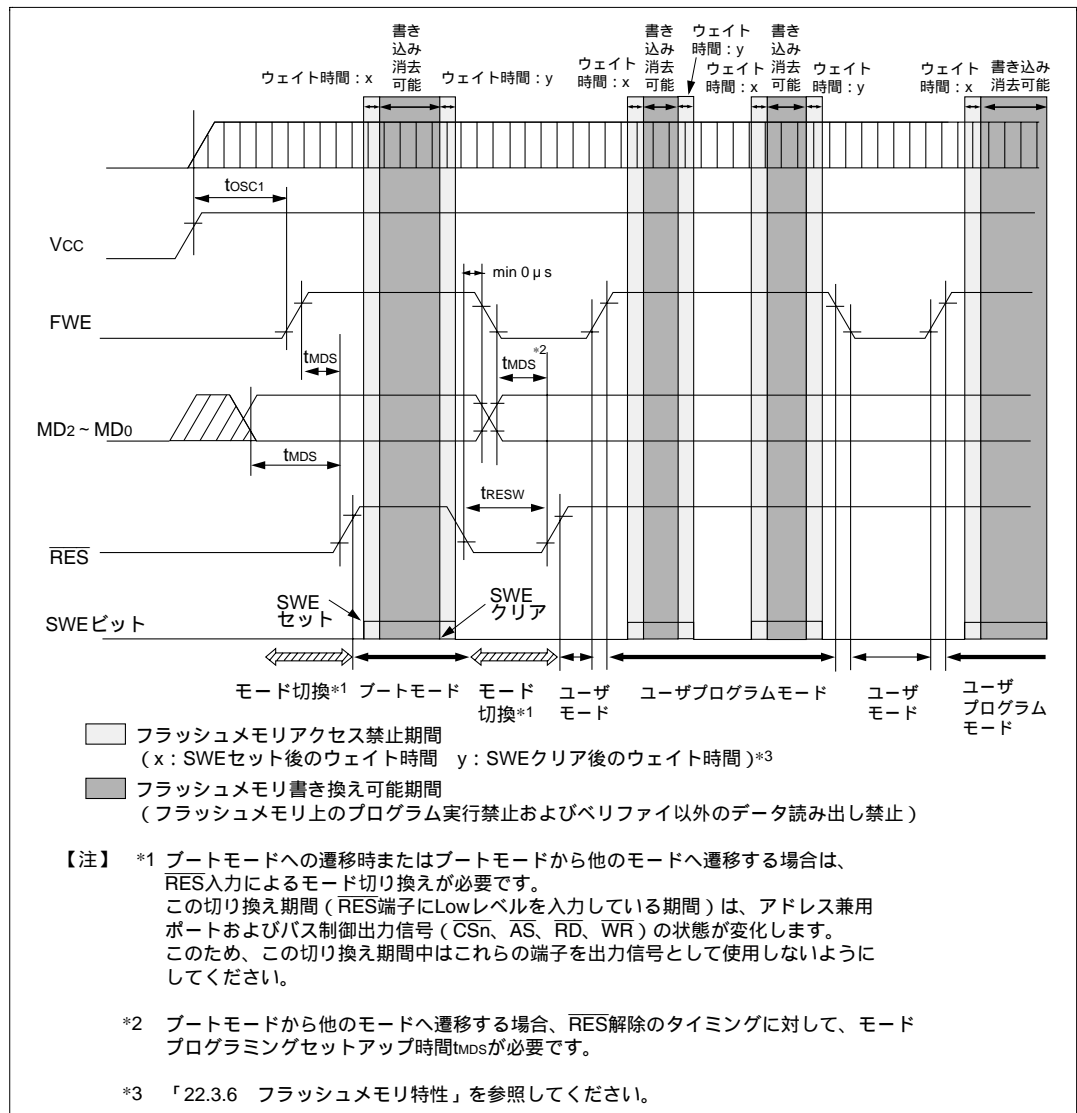


図 18.18 モード遷移タイミング

(例: ブートモード ユーザモード ユーザプログラムモード)

---

# 19. クロック発振器

---

## 第 19 章 目次

19.1 概要.....	705
19.1.1 ブロック図.....	705
19.2 発振器.....	706
19.2.1 水晶発振子を接続する方法.....	706
19.2.2 外部クロックを入力する方法.....	708
19.3 デューティ補正回路.....	711
19.4 プリスケーラ.....	711
19.5 分周器.....	711
19.5.1 レジスタ構成.....	711
19.5.2 分周比コントロールレジスタ (DIVCR).....	711
19.5.3 使用上の注意.....	712



## 19.1 概要

本 LSI は、クロック発振器 (CPG : Clock Pulse Generator) を内蔵しており、クロック発振器はシステムクロック ( )、および内部クロック ( /2 ~ /4096 ) を生成します。

分周器は、デューティ補正されたクロックを分周してシステムクロック ( ) を生成します。 は 端子に出力される\*<sup>1</sup> とともに内部モジュールへクロックを供給するプリスケアラの基本クロックとなります。なお、分周器の分周比は分周比コントロールレジスタ (DIVCR) により 1/1、1/2、1/4、1/8 の中から選択できます\*<sup>2</sup>。チップ内の消費電流は分周比にほぼ比例して低減します。

【注】 \*<sup>1</sup> 端子の状態はチップの動作モードおよびモジュールスタンバイコントロールレジスタ (MSTCR) の PSTOP の設定により異なります。詳細は「19.7 クロック出力禁止機能」を参照してください。

\*<sup>2</sup> 分周比の変更は動作中ダイナミックに変更することができます。 端子のクロック出力も分周比を変更することにより変化します。このとき 端子から出力される周波数は、以下のようになります。

$$f_{out} = XTAL \times n \quad XTAL : \text{水晶発振子または外部クロックの周波数}$$

$$n : \text{分周比 (n = 1/1、1/2、1/4、1/8)}$$

### 19.1.1 ブロック図

図 19.1 にクロック発振器のブロック図を示します。

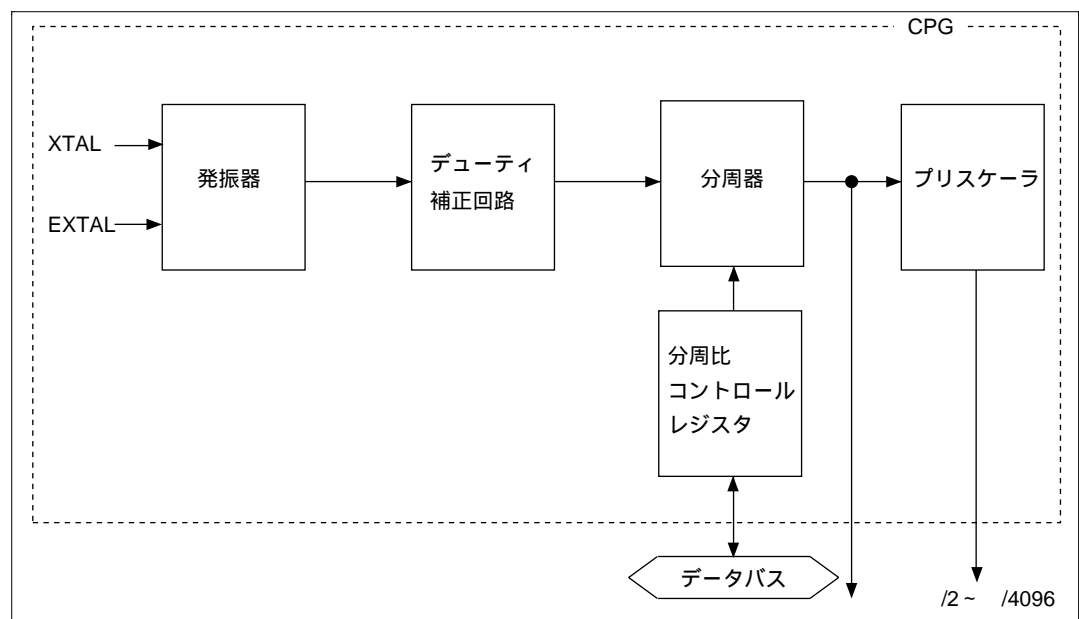


図 19.1 クロック発振器のブロック図

## 19.2 発振器

クロックを供給する方法には、水晶発振子を接続する方法と外部クロックを入力する方法の2とおりがあります。

### 19.2.1 水晶発振子を接続する方法

#### (1) 回路構成

水晶発振子を接続する場合の接続例を図 19.2 に示します。ダンピング抵抗  $R_d$  は、表 19.1 (1)、外付け容量  $CL_1$ 、 $CL_2$  は、表 19.1 (2) に示すものを使用してください。また、水晶発振子は、AT カット並列共振形を使用してください。

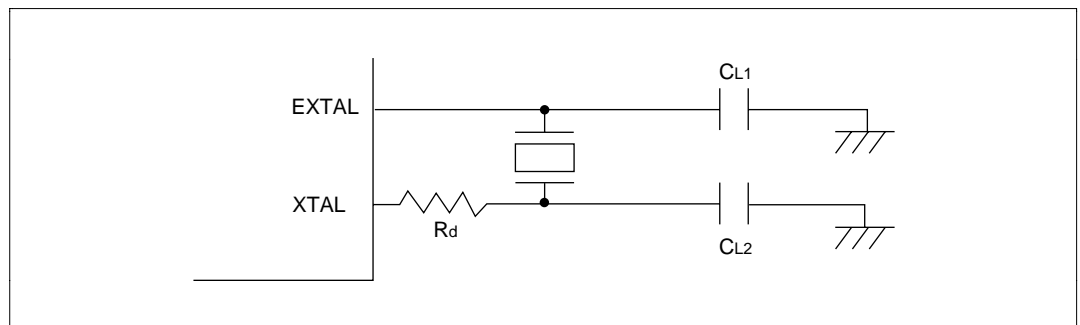


図 19.2 水晶発振子を接続する場合の接続例

20MHz より高周波の水晶発振子を接続する場合は、表 19.1 (2) のとおり外付け負荷容量値を 10[pf]以下としてください。また実装状態での発振周波数精度の向上のため、回路定数の決定については、発振のマッチング評価などを十分にご検討くださいますようお願い申し上げます。

表 19.1 (1) ダンピング抵抗値

ダンピング抵抗値	周波数 $f$ (MHz)														
	2	$2 < f$	4	$4 < f$	8	$8 < f$	10	$10 < f$	13	$13 < f$	16	$16 < f$	18	$18 < f$	25
$R_d$ ( )	1k	500	200	0	0	0	0	0	0	0	0	0	0	0	0

【注】 水晶発振子は、2MHz~20MHz が使用できます。

2MHz 未満で動作させる場合は、内蔵の分周器を使用してください (2MHz 未満の水晶発振子は使用できません)。

表 19.1 (2) 外付け容量値

外付け容量値	5V 品		低電圧品		
	周波数 $f$ (MHz)	$20 < f \leq 25$	$2 < f \leq 20$	$2 < f \leq 13$	$2 < f \leq \text{TBD}^*$
$CL_1 = CL_2$ (pF)	10	10 ~ 22		10	

## (2) 水晶発振子

図 19.3 に水晶発振子の等価回路を示します。水晶発振子は表 19.2 に示す特性のものを使用してください。

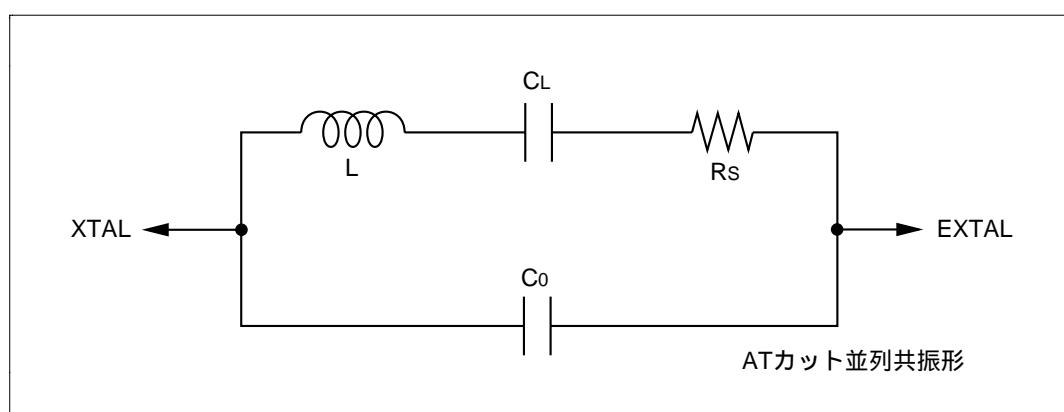


図 19.3 水晶発振子の等価回路

表 19.2 水晶発振子のパラメータ

周波数 (MHz)	2	4	8	10	12	16	18	20
$R_s$ max ( )	500	120	80	70	60	50	40	40
$C_o$ (pF)	7pF max							

水晶発振子は、 と同一の周波数のものを使用してください。

## (3) ボード設計上の注意

水晶発振子を接続して発振させる場合、次の点に注意してください。

発振回路部の近くで信号線を通させないでください。誘導により正しい発振ができなくなる場合があります (図 19.4)。

また、ボード設計に際しては、水晶発振子および負荷容量はできるだけ XTAL、EXTAL 端子の近くに配置してください。

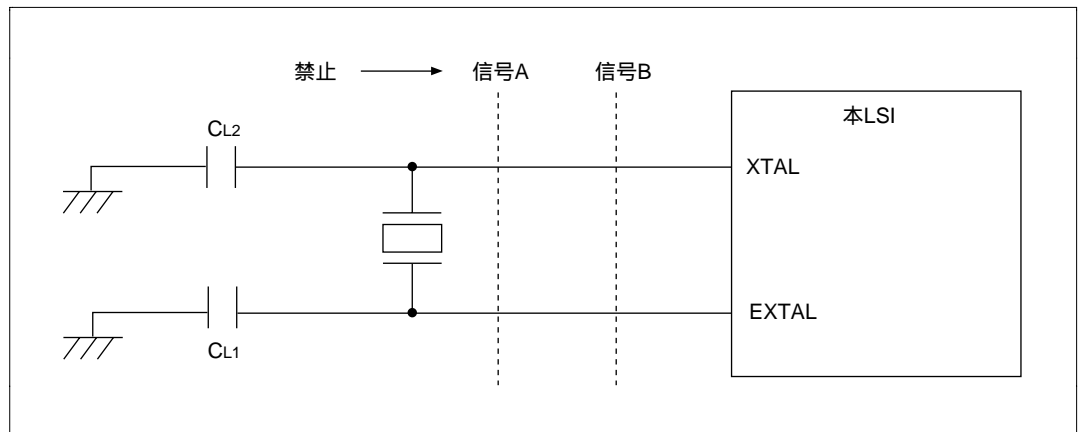


図 19.4 発振回路部のボード設計に関する注意事項

## 19.2.2 外部クロックを入力する方法

### (1) 回路構成

外部クロック入力の接続例を図 19.5 に示します。XTAL 端子をオープン状態にする場合は、XTAL 端子の寄生容量は 10pF 以下としてください。XTAL 端子の寄生容量が 10pF を超える場合は (b) のように接続し、スタンバイモード時には外部クロックが High レベルとなるようにしてください。

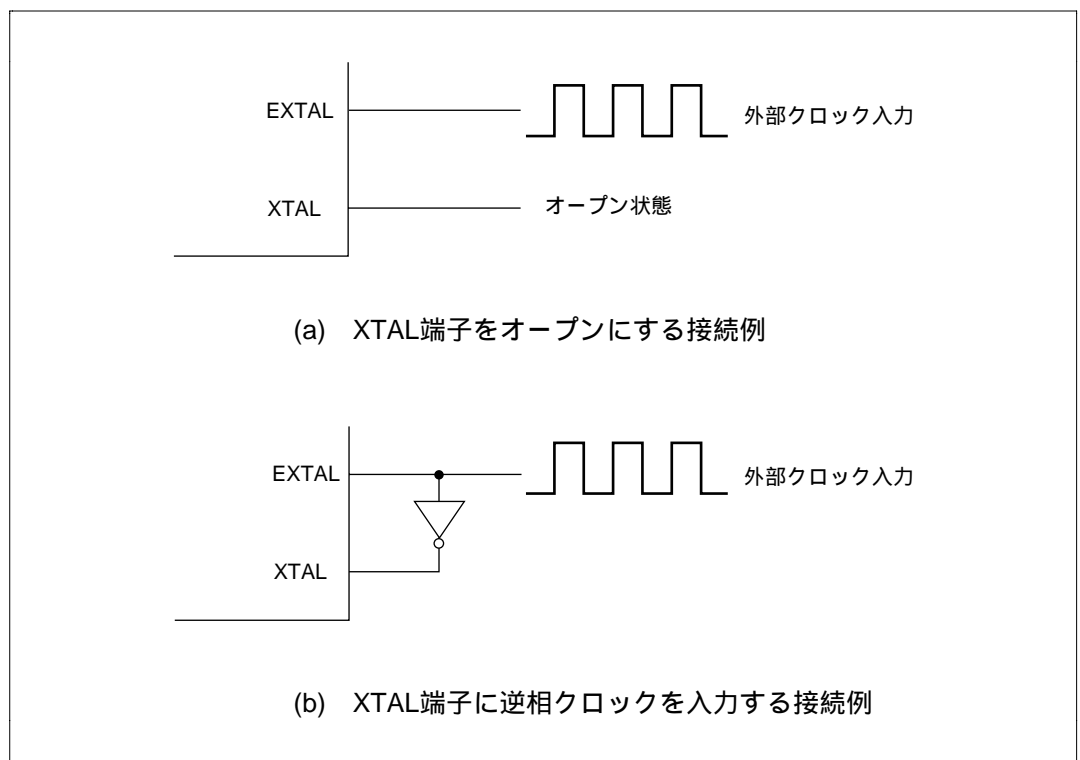


図 19.5 外部クロックを入力する場合の接続例

## (2) 外部クロック

外部クロックの周波数は、分周器によって分周しないときの周波数と同一にしてください。

表 19.3 にクロックタイミング、図 19.6 に外部クロック入力タイミングを示します。

図 19.7 に外部クロック出力安定遅延時間タイミングを示します。発振器とデューティ補正回路は、EXTAL 端子に入力した外部クロック入力の波形を調整する機能を持っています。

EXTAL 端子に規定のクロック信号を入力すると、外部クロック出力安定遅延時間 ( $t_{\text{DEXT}}$ ) 経過後に内部のクロック信号出力が確定します。 $t_{\text{DEXT}}$  期間中はクロック信号出力が確定していないので、リセット信号を Low にし、リセット状態に保持してください。

表 19.3 クロックタイミング

項目	記号	$V_{\text{CC}} = 3.0 \sim 5.5\text{V}$		$V_{\text{CC}} = 5.0\text{V} \pm 10\%$		単位	測定条件	
		min	max	min	max			
外部クロック入力パルス幅 Low レベル	$t_{\text{EXL}}$	30		15		ns	図 19.6	
外部クロック入力パルス幅 High レベル	$t_{\text{EXH}}$	30		15		ns		
外部クロック立ち上がり時間	$t_{\text{EXr}}$		8		5	ns		
外部クロック立ち下がり時間	$t_{\text{EXf}}$		8		5	ns		
クロックパルス幅 Low レベル	$t_{\text{CL}}$	0.4	0.6	0.4	0.6	$t_{\text{cyc}}$		
		80		80		ns	< 5MHz	
クロックパルス幅 High レベル	$t_{\text{CH}}$	0.4	0.6	0.4	0.6	$t_{\text{cyc}}$	5MHz	
		80		80		ns	< 5MHz	
外部クロック出力安定遅延時間	$t_{\text{DEXT}}^{*1}$	500		500		$\mu\text{s}$	図 19.7	

【注】 \*1  $t_{\text{DEXT}}$  は、 $\overline{\text{RES}}$  パルス幅 ( $t_{\text{RESW}}$ ) を含みます。 $t_{\text{RESW}} = 20 t_{\text{cyc}}$  となります。



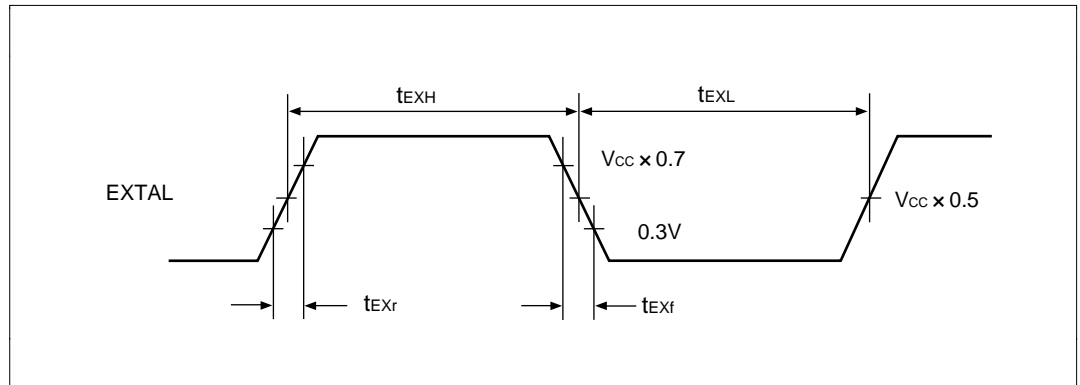


図 19.6 外部クロック入力タイミング

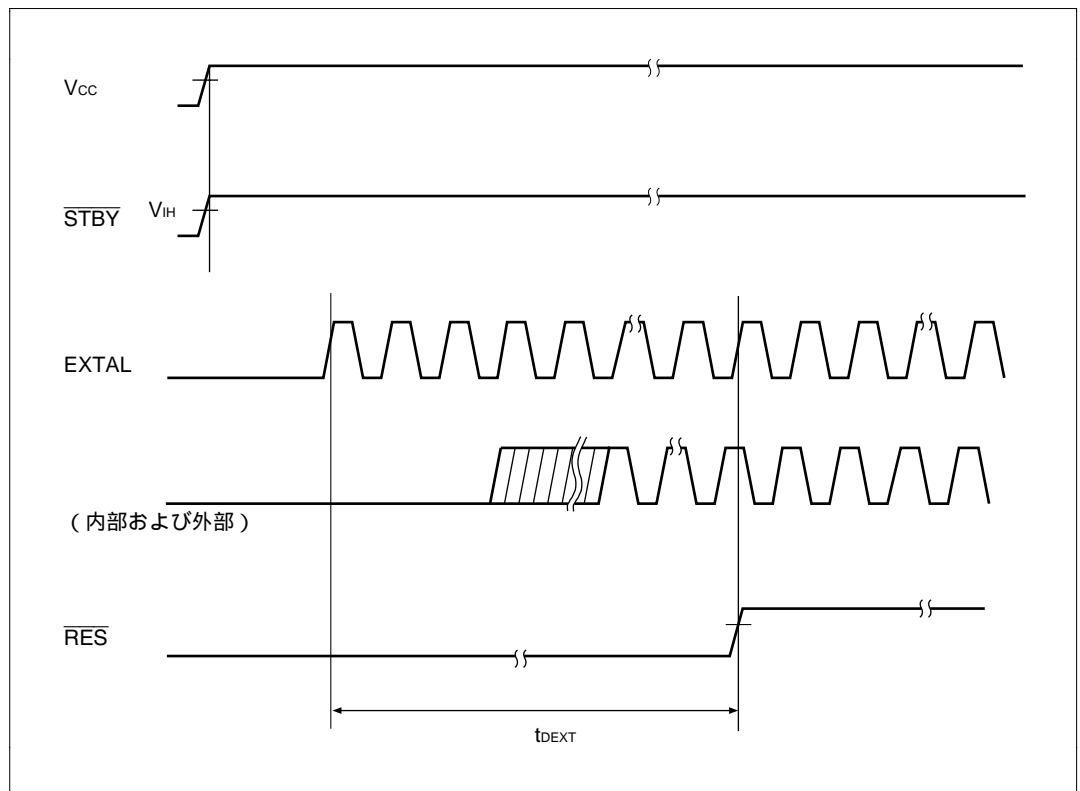


図 19.7 外部クロック出力安定遅延時間タイミング

## 19.3 デューティ補正回路

デューティ補正回路は、周波数 5MHz 以上の発振に対し発振器からのクロックのデューティを補正し、 を生成します。

## 19.4 プリスケアラ

プリスケアラは、 を分周し内部クロック ( /2 ~ /4096 ) を生成します。

## 19.5 分周器

分周器はデューティ補正されたクロックを分周して を生成します。分周比は以下に説明する DIVCR の値を書き換えることにより動作中ダイナミックに変更できます。チップ内の消費電流は分周比にほぼ比例して低減します。また分周器で生成した は 端子より出力することができます。

### 19.5.1 レジスタ構成

表 19.4 に分周器のレジスタ構成を示します。

表 19.4 分周器のレジスタ構成

アドレス*	名称	略称	R/W	初期値
H'EE01B	分周比コントロールレジスタ	DIVCR	R/W	H'FC

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

### 19.5.2 分周比コントロールレジスタ (DIVCR)

DIVCR は 8 ビットのリード/ライト可能なレジスタで分周器の分周比を選択します。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DIV1	DIV0
初期値:	1	1	1	1	1	1	0	0
R/W :	—	—	—	—	—	—	R/W	R/W

リザーブビット

分周比ビット1、0  
分周比を設定するビットです。

DIVCR はリセットまたはハードウェアスタンバイモード時に H'FC にイニシャライズされます。

ソフトウェアスタンバイモード時にはイニシャライズされません。

ビット7～2：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

ビット1、0：分周比（DIV1、0）

分周比を選択します。DIV1、0ビットと分周比の関係は以下のとおりです。

ビット1	ビット0	分周比
DIV1	DIV0	
0	0	1/1 (初期値)
0	1	1/2
1	0	1/4
1	1	1/8

### 19.5.3 使用上の注意

DIVCRの設定により の周波数が変わりますので、以下の点に注意してください。

- (1) 分周比は電気的特性のAC タイミングのクロックサイクル時間  $t_{cyc}$  の動作保証範囲内におさまるように選択してください。すなわち  $t_{cyc} \times \text{分周比} = \text{動作周波数}$  の下限とし、が下限より小さくならないように注意してください。
- (2) 内部モジュールは、全て を基準に動作します。  
このため、分周比変更の前後でタイマやSCIなどの時間処理が変わりますので注意してください。  
またソフトウェアスタンバイ解除時の待機時間も分周比を変更することで変わります。詳細は「21.4.3 ソフトウェアスタンバイモード解除後の発振安定待機時間の設定」を参照してください。

---

# 20. 低消費電力状態

---

## 第 20 章 目次

20.1	概要	715
20.2	レジスタ構成	717
20.2.1	システムコントロールレジスタ (SYSCR)	717
20.2.2	モジュールスタンバイコントロールレジスタ H (MSTCRH)	720
20.2.3	モジュールスタンバイコントロールレジスタ L (MSTCRL)	722
20.3	スリープモード	724
20.3.1	スリープモードへの遷移	724
20.3.2	スリープモードの解除	724
20.4	ソフトウェアスタンバイモード	725
20.4.1	ソフトウェアスタンバイモードへの遷移	725
20.4.2	ソフトウェアスタンバイモードの解除	725
20.4.3	ソフトウェアスタンバイモード解除後の発振安定待機時間の設定	726
20.4.4	ソフトウェアスタンバイモードの応用例	728
20.4.5	使用上の注意	728
20.5	ハードウェアスタンバイモード	729
20.5.1	ハードウェアスタンバイモードへの遷移	729
20.5.2	ハードウェアスタンバイモードの解除	729
20.5.3	ハードウェアスタンバイモードのタイミング	730
20.6	モジュールスタンバイ機能	731
20.6.1	モジュールスタンバイタイミング	731
20.6.2	モジュールスタンバイ中のリード/ライト	731
20.6.3	使用上の注意	731
20.7	クロック出力禁止機能	733



## 20.1 概要

本 LSI には、CPU 機能を停止して消費電力を著しく低下させる低消費電力状態および、内蔵モジュールの機能を選択的に停止させることにより消費電力を低減させるモジュールスタンバイ機能があります。

低消費電力状態には、次の 3 種類のモードがあります。

- (1) スリープモード
- (2) ソフトウェアスタンバイモード
- (3) ハードウェアスタンバイモード

モジュールスタンバイ機能は低消費電力状態とは独立に内蔵モジュールのうち、16 ビットタイマ、8 ビットタイマ、SCI0、SCI1、SCI2、DMAC、DRAM インタフェース、A/D 変換器を停止させることができます。

各モードへ遷移する条件と CPU や周辺機能などの状態、および各モードの解除方法を表 20.1 に示します。

表 20.1 低消費電力状態とモジュールスタンバイ機能

モード/機能	遷移条件	状態											解除方法				
		クロック	CPU	レジスタ	DMAC	DRAM インタ フェース	16ビット タイマ	8ビット タイマ	SCI0	SCI1	SCI2	AD		その他の 周辺 機能	RAM	クロック 出力 <sup>*4</sup>	I/O ポート
スリープ モード	SYSCRのSSBY = 0 の状態ですLEEP 命令を実行	動作	停止	保持	動作	動作	動作	動作	動作	動作	動作	動作	動作	保持	出力	保持	<ul style="list-style-type: none"> <li>• 割込み</li> <li>• RES端子</li> <li>• STBY端子</li> </ul>
ソフトウェア スタンバイモード	SYSCRのSSBY = 1 の状態ですLEEP 命令を実行	停止	停止	保持	停止 リセット <sup>*1</sup>	停止 保持 <sup>*1</sup>	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	High 出力	保持	<ul style="list-style-type: none"> <li>• NMI端子</li> <li>• IRQ<sub>0</sub> ~ IRQ<sub>2</sub>端子</li> <li>• RES端子</li> <li>• STBY端子</li> </ul>
ハードウェア スタンバイモード	STBY端子を Lowレベル	停止	停止	不定	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 リセット	停止 <sup>*3</sup> 保持	ハイイン ピータン ピータン ス	ハイイン ピータン ピータン ス	<ul style="list-style-type: none"> <li>• STBY端子</li> <li>• RES端子</li> <li>• RES端子</li> </ul>
モジュール スタンバイ機能	MSTCRHおよび MSTCRLの該当ビット を1にセット	動作	動作	-	停止 <sup>*2</sup> リセット <sup>*1</sup>	停止 <sup>*2</sup> 保持 <sup>*1</sup>	停止 <sup>*2</sup> リセット	停止 <sup>*2</sup> リセット	停止 <sup>*2</sup> リセット	停止 <sup>*2</sup> リセット	停止 <sup>*2</sup> リセット	停止 <sup>*2</sup> リセット	停止 <sup>*2</sup> リセット	停止 <sup>*2</sup> リセット	ハイイン ピータン ピータン ス	ハイイン ピータン ピータン ス	<ul style="list-style-type: none"> <li>• STBY端子</li> <li>• RES端子</li> <li>• MSTCRの 該当ビットを 0にクリア<sup>*5</sup></li> </ul>

【注】 \*1 RTCNT、RTMCSRのビット7、6はイニシャライズされ、その他保持となります。

\*2 MSTCRの該当するビットを1にセットした状態です。詳細は「20.2.2 モジュールスタンバイコントロールレジスタH (MSTCRH)」および「20.2.3 モジュールスタンバイコントロールレジスタL (MSTCRL)」を参照してください。

\*3 プログラム実効状態からハードウェアスタンバイ状態に遷移する場合には、事前にSYSCRのRAMEビットを0にクリアする必要があります。

\*4 P67を 出力端子として使用する場合。

\*5 MSTCRの各ビットを1にセットすると該当する周辺モジュールのレジスタは初期化されます。したがって周辺モジュールを再起動する場合はMSTCRの該当ビットを0にクリア後レジスタの再設定を行ってください。

《記号説明》

- SYSCR : システムコントロールレジスタ
- SSBY : ソフトウェアスタンバイビット
- MSTCRH : モジュールスタンバイコントロールレジスタH
- MSTCRL : モジュールスタンバイコントロールレジスタL

## 20.2 レジスタ構成

本 LSI には低消費電力状態の制御を行うシステムコントロールレジスタ (SYSCR) とモジュールスタンバイ機能の制御を行うモジュールスタンバイコントロールレジスタ H (MSTCRH) モジュールスタンバイコントロールレジスタ L (MSTCRL) があります。レジスタ構成を表 20.2 に示します。

表 20.2 レジスタ構成

アドレス	名称	略称	R/W	初期値
H'EE012	システムコントロールレジスタ	SYSCR	R/W	H'09
H'EE01C	モジュールスタンバイコントロールレジスタ H	MSTCRH	R/W	H'78
H'EE01D	モジュールスタンバイコントロールレジスタ L	MSTCRL	R/W	H'00

【注】 \* アドバンスモード時のアドレス下位 20 ビットを示しています。

### 20.2.1 システムコントロールレジスタ (SYSCR)



SYSCR は 8 ビットのリード/ライト可能なレジスタで、ビット 7 の SSBY ビットとビット 6~4 の STS2~STS0 ビットとビット 1 の SSOE ビットにより低消費電力状態の制御を行います。なお、SYSCR のその他のビットについての詳細は「3.3 システムコントロールレジスタ」を参照してください。



## ビット7：ソフトウェアスタンバイ（SSBY）

ソフトウェアスタンバイモードへの遷移を指定します。

なお、外部割込みによりソフトウェアスタンバイモードが解除され、通常動作に遷移したときこのビットは1にセットされたままです。クリアする場合は、0をライトしてください。

ビット7	説明	
SSBY		
0	SLEEP命令実行後、スリープモードに遷移	(初期値)
1	SLEEP命令実行後、ソフトウェアスタンバイモードに遷移	

## ビット6～4：スタンバイタイムセレクト2～0（STS2～0）

外部割込みによって、ソフトウェアスタンバイモードを解除する場合に、クロックが安定するまでCPUと周辺機能が待機する時間を選択します。水晶発振の場合、表20.3を参照し動作周波数に応じて待機時間が7ms（発振安定時間）以上となるように選択してください。外部クロックの場合、待機時間が100 $\mu$ s以上となるよう選択してください。

ビット6	ビット5	ビット4	説明
STS2	STS1	STS0	
0	0	0	待機時間 = 8192 ステート (初期値)
0	0	1	待機時間 = 16384 ステート
0	1	0	待機時間 = 32768 ステート
0	1	1	待機時間 = 65536 ステート
1	0	0	待機時間 = 131072 ステート
1	0	1	待機時間 = 262144 ステート
1	1	0	待機時間 = 1024 ステート
1	1	1	使用禁止

ビット1：ソフトウェアスタンバイ出力ポートイネーブル（SSOE）

ソフトウェアスタンバイモード時にアドレスバス、バス制御信号（ $\overline{CS}_0 \sim \overline{CS}_7$ 、 $\overline{AS}$ 、 $\overline{RD}$ 、 $\overline{HWR}$ 、 $\overline{LWR}$ 、 $\overline{UCAS}$ 、 $\overline{LCAS}$ 、 $\overline{RFSH}$ ）の出力を保持または High 固定するか、ハイインピーダンスにするかを指定します。

ビット1	説明
SSOE	
0	ソフトウェアスタンバイモード時、アドレスバス、バス制御信号はすべてハイインピーダンス (初期値)
1	ソフトウェアスタンバイモード時、 アドレスバス：出力状態を保持 バス制御信号：High 固定

## 20.2.2 モジュールスタンバイコントロールレジスタ H (MSTCRH)

MSTCRH は、8 ビットのリード/ライト可能なレジスタです。 クロック出力の制御を行い、また、内蔵モジュールをモジュールごとにスタンバイ状態にするモジュールスタンバイ機能を設定します。モジュールスタンバイ機能を設定できるモジュールは、SCI0、SCI1、SCI2 です。

ビット :	7	6	5	4	3	2	1	0
	PSTOP	—	—	—	—	MSTPH2	MSTPH1	MSTPH0
初期値 :	0	1	1	1	1	0	0	0
R/W :	R/W	—	—	—	—	R/W	R/W	R/W

リザーブビット

クロックストップ  
クロックの出力を許可または禁止するビットです。

モジュールスタンバイH2~0  
モジュールをスタンバイ状態にするための選択ビットです。

MSTCRH はリセット、またはハードウェアスタンバイモード時に、H'78 にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

### ビット7： クロックストップ (PSTOP)

クロックの出力を許可または禁止します。

ビット7	説明
PSTOP	
0	クロックの出力を許可 (初期値)
1	クロックの出力を禁止

### ビット6~3：リザーブビット

リザーブビットです。リードすると常に1が読み出されます。ライトは無効です。

### ビット2：モジュールスタンバイ H2 (MSTPH2)

SCI2 をスタンバイ状態にするかを選択します。

ビット2	説明
MSTPH2	
0	SCI2 は通常動作 (初期値)
1	SCI2 はスタンバイ状態

## ビット1：モジュールスタンバイ H1 (MSTPH1)

SCI1 をスタンバイ状態にするかを選択します。

ビット1	説明
MSTPH1	
0	SCI1 は通常動作 (初期値)
1	SCI1 はスタンバイ状態

## ビット0：モジュールスタンバイ H0 (MSTPH0)

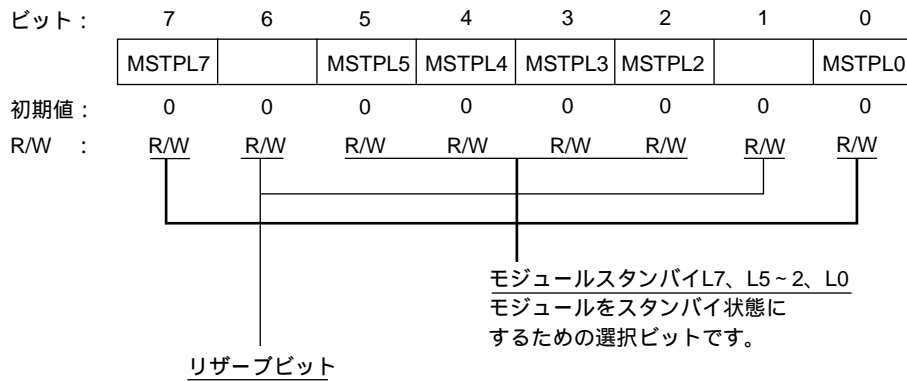
SCI0 をスタンバイ状態にするかを選択します。

ビット0	説明
MSTPH0	
0	SCI0 は通常動作 (初期値)
1	SCI0 はスタンバイ状態

### 20.2.3 モジュールスタンバイコントロールレジスタ L (MSTCRL)

MSTCRL は、8 ビットのリード/ライト可能なレジスタです。

内蔵モジュールをモジュールごとにスタンバイ状態にするモジュールスタンバイ機能を設定します。モジュールスタンバイ機能を設定できるモジュールは、DMAC、16 ビットタイマ、DRAM インタフェース、8 ビットタイマ、A/D 変換器です。



MSTCRL はリセット、またはハードウェアスタンバイモード時に、H'00 にイニシャライズされます。ソフトウェアスタンバイモードではイニシャライズされません。

#### ビット7 : モジュールスタンバイ L7 (MSTPL7)

DMAC をスタンバイ状態にするかを選択します。

ビット7	説明
MSTPL7	
0	DMAC は通常動作 (初期値)
1	DMAC はスタンバイ状態

#### ビット6 : リザーブビット

リザーブビットです。リード/ライト可能です。

#### ビット5 : モジュールスタンバイ L5 (MSTPL5)

DRAM インタフェースをスタンバイ状態にするかを選択します。

ビット5	説明
MSTPL5	
0	DRAM インタフェースは通常動作 (初期値)
1	DRAM インタフェースはスタンバイ状態

## ビット4：モジュールスタンバイ L4 (MSTPL4)

16ビットタイマをスタンバイ状態にするかを選択します。

ビット4	説明
MSTPL4	
0	16ビットタイマは通常動作 (初期値)
1	16ビットタイマはスタンバイ状態

## ビット3：モジュールスタンバイ L3 (MSTPL3)

8ビットタイマチャンネル0、1をスタンバイ状態にするかを選択します。

ビット3	説明
MSTPL3	
0	8ビットタイマチャンネル0、1は通常動作 (初期値)
1	8ビットタイマチャンネル0、1はスタンバイ状態

## ビット2：モジュールスタンバイ L2 (MSTPL2)

8ビットタイマチャンネル2、3をスタンバイ状態にするかを選択します。

ビット2	説明
MSTPL2	
0	8ビットタイマチャンネル2、3は通常動作 (初期値)
1	8ビットタイマチャンネル2、3はスタンバイ状態

## ビット1：リザーブビット

リザーブビットです。リード/ライト可能です。

## ビット0：モジュールスタンバイ L0 (MSTPL0)

A/D変換器をスタンバイ状態にするかを選択します。

ビット0	説明
MSTPL0	
0	A/D変換器は通常動作 (初期値)
1	A/D変換器はスタンバイ状態

## 20.3 スリープモード

### 20.3.1 スリープモードへの遷移

SYSCR の SSBY ビットを 0 にクリアした状態で SLEEP 命令を実行すると、プログラム実行状態からスリープモードに遷移します。CPU の動作は、SLEEP 命令実行直後に停止します。CPU の動作は停止しますが、CPU のレジスタの内容は保持されます。スリープモードでは DMA コントローラ (DMAC)、DRAM インタフェース、および内蔵周辺モジュールの機能は停止しません。ただし、モジュールスタンバイ機能により、スタンバイ状態となっている内蔵モジュールの機能は動作しません。

### 20.3.2 スリープモードの解除

スリープモードの解除は、割込み、 $\overline{\text{RES}}$  端子、 $\overline{\text{STBY}}$  端子によって行われます。

#### (1) 割込みによる解除

割込み要求が発生すると、割込み例外処理状態に遷移し、スリープモードは解除されます。なお、内蔵周辺モジュールによる割込みがモジュール側で禁止されている場合、また、NMI 以外の割込みで CCR の I、UI ビット、IPR によってマスクされている場合にはスリープモードは解除されません。

#### (2) $\overline{\text{RES}}$ 端子による解除

$\overline{\text{RES}}$  端子を Low レベルにするとリセット状態に遷移し、スリープモードは解除されます。

#### (3) $\overline{\text{STBY}}$ 端子による解除

$\overline{\text{STBY}}$  端子を Low レベルにすると、ハードウェアスタンバイモードに遷移します。

## 20.4 ソフトウェアスタンバイモード

### 20.4.1 ソフトウェアスタンバイモードへの遷移

ソフトウェアスタンバイモードに遷移するには SYSCR の SSBY ビットを 1 にセットした状態で、SLEEP 命令を実行します。

ソフトウェアスタンバイモードでは、CPU だけでなくクロックをはじめ内蔵周辺モジュールの機能が停止するため、消費電力は著しく低減されます。DMAC、内蔵周辺モジュールはリセット状態になり、停止します。規定の電圧が与えられている限り、CPU のレジスタ、および内蔵 RAM のデータは保持されます。I/O ポートおよび DRAM インタフェース\* の状態も保持されています。WDT をウォッチドッグタイマとして使用している場合 ( $WT/\overline{IT}=1$ )、必ず TME ビットを 0 クリアしてから SSBY をセットしてください。また、TME を 1 にセットするときは、SSBY を 0 クリアしてください。

また、ソフトウェアスタンバイモードに遷移する前に、BRCR の BRLE ビットをクリア (バス解放禁止状態) にしてください。

【注】\* RTCNT、RTMCSR のビット 7、6 はイニシャライズされ、その他は保持となります。

### 20.4.2 ソフトウェアスタンバイモードの解除

ソフトウェアスタンバイモードの解除は、外部割込み (NMI 端子、 $\overline{IRQ_0} \sim \overline{IRQ_2}$  端子)、 $\overline{RES}$  端子、または  $\overline{STBY}$  端子によって行われます。

#### (1) 割込みによる解除

NMI、 $IRQ_0 \sim IRQ_2$  割込み要求信号が入力されると、クロックの発振が開始され、SYSCR の STS2 ~ STS0 ビットによって設定された時間が経過した後、安定したクロックが本 LSI 全体に供給されて、ソフトウェアスタンバイモードは解除され、割込み例外処理を開始します。

なお、 $IRQ_0 \sim IRQ_2$  割込みは、対応するイネーブルビットが 0 にクリアされている場合、または CPU でマスクされている場合には、ソフトウェアスタンバイモードは解除されません。

#### (2) $\overline{RES}$ 端子による解除

$\overline{RES}$  端子を Low レベルにすると、クロックの発振が開始されます。クロックの発振開始と同時に、本 LSI 全体にクロックが供給されます。このとき  $\overline{RES}$  端子は必ずクロックの発振が安定するまで Low レベルに保持してください。 $\overline{RES}$  端子を High レベルにすると、CPU はリセット例外処理を開始します。



(3)  $\overline{\text{STBY}}$  端子による解除

$\overline{\text{STBY}}$  端子を Low レベルにすると、ハードウェアスタンバイモードに遷移します。

### 20.4.3 ソフトウェアスタンバイモード解除後の発振安定待機時間の設定

SYSCR の STS2 ~ STS0 ビットおよび DIVCR の DIV1、0 ビットの設定は、以下のよう  
にしてください。

(1) 水晶発振の場合

待機時間が 7ms (発振安定時間) 以上となるように STS2 ~ STS0、DIV1、0 を設定して  
ください。

表 20.3 は動作周波数と STS2 ~ STS0、DIV1、0 ビットの設定値に対する待機時間を示  
します。

(2) 外部クロックの場合

待機時間が 100 $\mu$ s 以上となるよう、STS2 ~ STS0、DIV0、DIV1 ビットを設定してくだ  
さい。

表 20.3 動作周波数と発振安定待機時間

DIV1	DIV0	STS2	STS1	STS0	待機時間	25MHz	20MHz	18MHz	16MHz	12MHz	10MHz	8MHz	6MHz	4MHz	2MHz	1MHz	単位
0	0	0	0	0	8192ステート	0.3	0.4	0.46	0.51	0.65	0.8	1.0	1.3	2.0	4.1	8.2	ms
		0	0	1	16384ステート	0.7	0.8	0.91	1.0	1.3	1.6	2.0	2.7	4.1	8.2	16.4	
		0	1	0	32768ステート	1.3	1.6	1.8	2.0	2.7	3.3	4.1	5.5	8.2	16.4	32.8	
		0	1	1	65536ステート	2.6	3.3	3.6	4.1	5.5	6.6	8.2	10.9	16.4	32.8	65.5	
		1	0	0	131072ステート	5.2	6.6	7.3	8.2	10.9	13.1	16.4	21.8	32.8	65.5	131.1	
		1	0	1	262144ステート	10.5	13.1	14.6	16.4	21.8	26.2	32.8	43.7	65.5	131.1	262.1	
		1	1	0	1024ステート	0.04	0.05	0.057	0.064	0.085	0.10	0.13	0.17	0.26	0.51	1.0	
		1	1	1	使用禁止												
0	1	0	0	0	8192ステート	0.7	0.8	0.91	1.02	1.4	1.6	2.0	2.7	4.1	8.2	16.4	ms
		0	0	1	16384ステート	1.3	1.6	1.8	2.0	2.7	3.3	4.1	5.5	8.2	16.4	32.8	
		0	1	0	32768ステート	2.6	3.3	3.6	4.1	5.5	6.6	8.2	10.9	16.4	32.8	65.5	
		0	1	1	65536ステート	5.2	6.6	7.3	8.2	10.9	13.1	16.4	21.8	32.8	65.5	131.1	
		1	0	0	131072ステート	10.5	13.1	14.6	16.4	21.8	26.2	32.8	43.7	65.5	131.1	262.1	
		1	0	1	262144ステート	21.0	26.2	29.1	32.8	43.7	52.4	65.5	87.4	131.1	262.1	524.3	
		1	1	0	1024ステート	0.08	0.10	0.11	0.13	0.17	0.20	0.26	0.34	0.51	1.0	2.0	
		1	1	1	使用禁止												
1	0	0	0	0	8192ステート	1.3	1.6	1.8	2.0	2.7	3.3	4.1	5.5	8.2	16.4	32.8	ms
		0	0	1	16384ステート	2.6	3.3	3.6	4.1	5.5	6.6	8.2	10.9	16.4	32.8	65.5	
		0	1	0	32768ステート	5.2	6.6	7.3	8.2	10.9	13.1	16.4	21.8	32.8	65.5	131.1	
		0	1	1	65536ステート	10.5	13.1	14.6	16.4	21.8	26.2	32.8	43.7	65.5	131.1	262.1	
		1	0	0	131072ステート	21.0	26.2	29.1	32.8	43.7	52.4	65.5	87.4	131.1	262.1	524.3	
		1	0	1	262144ステート	41.9	52.4	58.3	65.5	87.4	104.9	131.1	174.8	262.1	524.3	1048.6	
		1	1	0	1024ステート	0.16	0.20	0.23	0.26	0.34	0.41	0.51	0.68	1.02	2.0	4.1	
		1	1	1	使用禁止												
1	1	0	0	0	8192ステート	2.6	3.3	3.6	4.1	5.5	6.6	8.2	10.9	16.4	32.8	65.5	ms
		0	0	1	16384ステート	5.2	6.6	7.3	8.2	10.9	13.1	16.4	21.8	32.8	65.5	131.1	
		0	1	0	32768ステート	10.5	13.1	14.6	16.4	21.8	26.2	32.8	43.7	65.5	131.1	262.1	
		0	1	1	65536ステート	21.0	26.2	29.1	32.8	43.7	52.4	65.5	87.4	131.1	262.1	524.3	
		1	0	0	131072ステート	41.9	52.4	58.3	65.5	87.4	104.9	131.1	174.8	262.1	524.3	1048.6	
		1	0	1	262144ステート	83.9	104.9	116.5	131.1	174.8	209.7	262.1	349.5	524.3	1048.6	2097.1	
		1	1	0	1024ステート	0.33	0.41	0.46	0.51	0.68	0.82	1.0	1.4	2.0	4.1	8.2	
		1	1	1	使用禁止												

   : 推奨設定時間

#### 20.4.4 ソフトウェアスタンバイモードの応用例

ここでは、NMI 端子の立下がりによってソフトウェアスタンバイモードに遷移し、NMI 端子の立上がりで解除を行う例を、図 20.1 に示します。

SYSCR の NMI エッジ (NMIEG) ビットが 0 にクリアされている (立下がりエッジ指定) 状態で NMI 割込みを受け付けた後、NMIEG ビットを 1 にセットします (立上がりエッジ指定)。SSBY ビットを 1 にセットした後、SLEEP 命令を実行してソフトウェアスタンバイモードに遷移します。

その後、NMI 端子の立上がりエッジで、ソフトウェアスタンバイモードが解除されます。

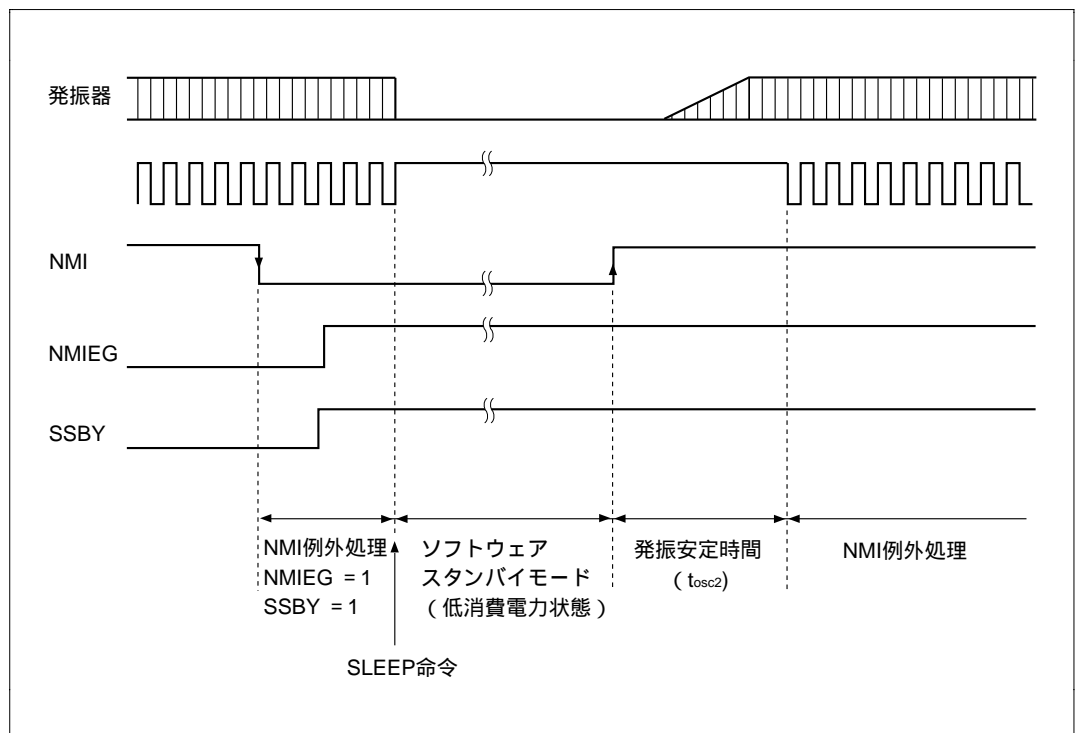


図 20.1 ソフトウェアスタンバイモード時の NMI タイミング (例)

#### 20.4.5 使用上の注意

ソフトウェアスタンバイモードでは、I/O ポートの状態が保持されます。したがって、High レベルを出力している場合、出力電流分の消費電流は低減されません。

## 20.5 ハードウェアスタンバイモード

### 20.5.1 ハードウェアスタンバイモードへの遷移

$\overline{\text{STBY}}$  端子を Low レベルにすると、どの処理状態からでもハードウェアスタンバイモードに遷移します。

ハードウェアスタンバイモードでは、CPU、DMAC、DRAM インタフェース、および内蔵周辺モジュールの機能が停止するため、消費電力は著しく低減します。内蔵 RAM 以外のすべてのモジュールはリセット状態になりますが、規定の電圧が与えられている限り、内蔵 RAM のデータは保持されます。I/O ポートは、ハイインピーダンス状態になります。

内蔵 RAM のデータを保持するためには、 $\overline{\text{STBY}}$  端子を Low レベルにする前に、SYSCR の RAME ビットを 0 にクリアしてください。

ハードウェアスタンバイモード中には、モード端子 (MD2 ~ MD0) の状態を変化させないでください。

### 20.5.2 ハードウェアスタンバイモードの解除

ハードウェアスタンバイモードの解除は、 $\overline{\text{STBY}}$  端子と  $\overline{\text{RES}}$  端子とで行われます。 $\overline{\text{RES}}$  端子を Low レベルにした状態で、 $\overline{\text{STBY}}$  端子を High レベルにすると、クロックは発振を開始します。このとき、 $\overline{\text{RES}}$  端子は必ずクロックの発振が安定するまで Low レベルに保持してください。 $\overline{\text{RES}}$  端子を High レベルにするとリセット例外処理を経て、プログラム実行状態に遷移します。

### 20.5.3 ハードウェアスタンバイモードのタイミング

ハードウェアスタンバイモードの各端子のタイミング例を図 20.2 に示します。

$\overline{\text{RES}}$  端子を Low レベルにした後、 $\overline{\text{STBY}}$  端子を Low レベルにすると、ハードウェアスタンバイモードに遷移します。解除は、 $\overline{\text{STBY}}$  端子を High レベルにし、クロックの発振安定時間経過後、 $\overline{\text{RES}}$  端子を Low レベルから High レベルにすることにより行われます。

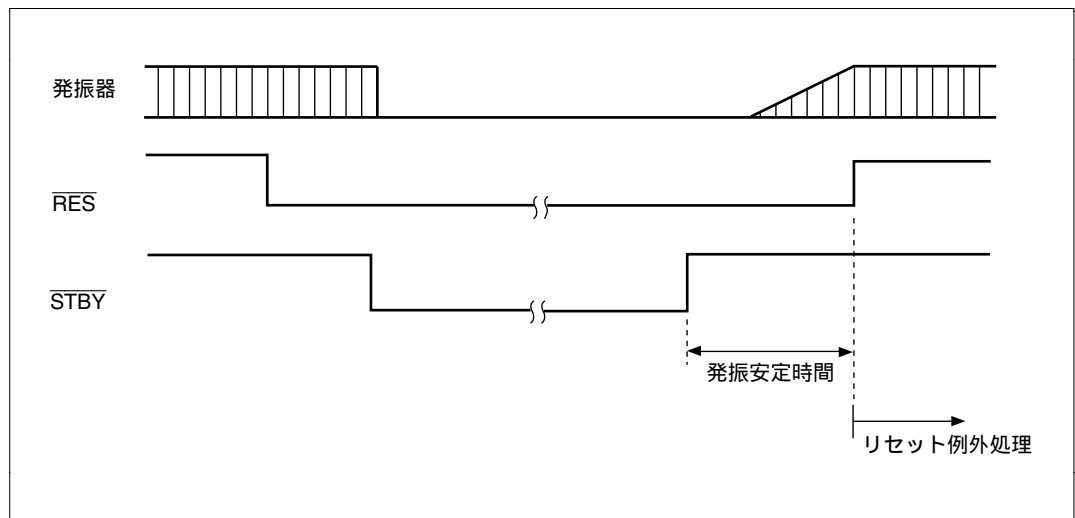


図 20.2 ハードウェアスタンバイモードのタイミング

## 20.6 モジュールスタンバイ機能

### 20.6.1 モジュールスタンバイタイミング

モジュールスタンバイ機能は、MSTCRH の MSTPH2~0 ビットおよび、MSTCRL の MSTPL7~0 ビットにより、内蔵モジュールのうち SCI2、SCI1、SCI0、DMAC、16 ビットタイマ、8 ビットタイマ、DRAM インタフェース、A/D 変換器を低消費電力状態とは独立に停止させることができます。MSTCR のビットを 1 にセットすると、MSTCR のライトサイクルの次のバスサイクルの先頭で上記周辺モジュールはスタンバイ状態となり動作が停止します。

### 20.6.2 モジュールスタンバイ中のリード/ライト

モジュールスタンバイ中の周辺モジュールのレジスタに対するリード/ライトはできません。リードすると常に H'FF が読み出されます。ライトは無効です。

### 20.6.3 使用上の注意

モジュールスタンバイ機能を使用するうえで以下のことに注意してください。

#### (1) DMAC の処置

モジュールスタンバイ機能を使用する場合、DMAC はバス権要求が発生しない状態で MSTCR のビットを 1 にセットしてください。バス権要求が発生した状態で MSTCR のビットが 1 にセットされると、バスアービタの動作が不確定となり誤動作の原因となります。

#### (2) DRAM インタフェースの処置

DRAM インタフェースに対し、モジュールスタンバイ機能を使用する場合、DRAM 空間は非選択の状態でも MSTCR のビットを 1 にセットしてください。

#### (3) 内蔵周辺モジュールの割込み

モジュールスタンバイビットをセットする場合、事前に当該モジュールの割込みを禁止してください。

モジュールスタンバイ機能によって、スタンバイ状態となった周辺モジュールは割込みフラグを含めてレジスタがすべて初期化されます。

#### (4) 端子の状態

モジュールスタンバイ機能によって、停止したモジュールが使用していた端子は、モジュールの機能が失われます。その後の端子状態については端子ごとに異なりますので、詳細は「第8章 IO ポート」を参照してください。端子の状態が入力から出力に変化する場合には、特に注意が必要です。例えば、SCII をモジュールスタンバイにした場合、受信端子はモジュールスタンバイと同時に受信端子としての機能が無くなりポートとなりますが、ポートの DDR が 1 にセットされていると、端子はデータ出力となり外部 SCI 送信データと衝突する可能性があります。この場合、ポートの DDR を 0 にクリアするなどの対策をとってください。

#### (5) レジスタの再設定

モジュールスタンバイ機能によって、停止した周辺モジュールのレジスタはすべて初期化されます。このため再動作させる際には、MSTCR のビットを 0 にクリアした後、レジスタの再設定が必要です。なお MSTCR のビットが 1 にセットされた状態ではレジスタへのライトはできません。

#### (6) DMAC による MSTCR のライト禁止

MSTCR は誤動作の防止のため CPU 以外のライトはできません。したがって DMAC によるリードはできますが、ライトはできません。

## 20.7 クロック出力禁止機能

MSTCRH の PSTOP ビットにより、クロックの出力を制御することができます。PSTOP ビットを 1 にセットすると、クロックは停止し、端子はハイインピーダンスになります。

図 20.3 にクロックの発振開始、発振停止のタイミングを示します。PSTOP ビットを 0 にクリアした状態では、クロックの出力は許可されます。表 20.4 に各処理状態における端子の状態を示します。

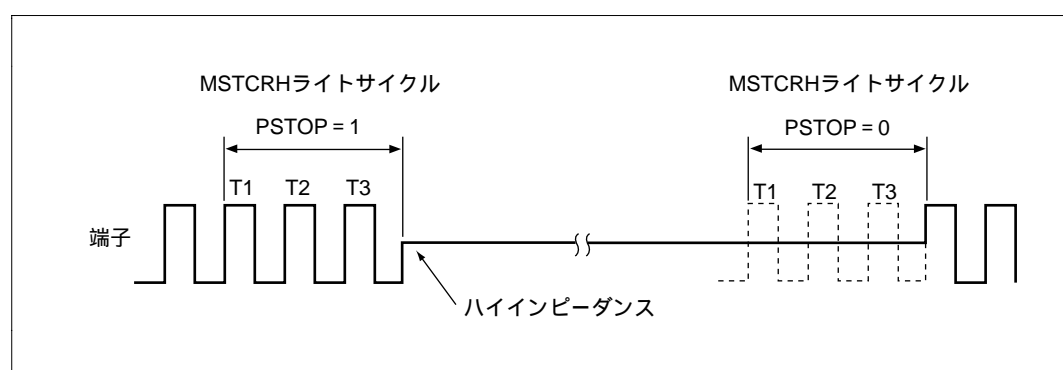


図 20.3 クロック発振開始、発振停止タイミング

表 20.4 各処理状態における端子の状態

処理状態	PSTOP = 0	PSTOP = 1
ハードウェアスタンバイ	ハイインピーダンス	ハイインピーダンス
ソフトウェアスタンバイ	H固定	ハイインピーダンス
スリープモード	出力	ハイインピーダンス
通常動作状態	出力	ハイインピーダンス





---

# 21. 電気的特性（暫定仕様）

---

## 第21章 目次

21.1 H8/3068F-ZTAT 電気的特性 .....	737
21.1.1 絶対最大定格 .....	737
21.1.2 DC 特性 .....	738
21.1.3 AC 特性 .....	743
21.1.4 A/D 変換特性 .....	748
21.1.5 D/A 変換特性 .....	749
21.1.6 フラッシュメモリ特性 .....	750
21.2 動作タイミング .....	754
21.2.1 クロックタイミング .....	754
21.2.2 制御信号タイミング .....	755
21.2.3 バスタイミング .....	757
21.2.4 DRAM インタフェースバスタイミング .....	760
21.2.5 TPC、I/Oポートタイミング .....	763
21.2.6 タイマ入出力タイミング .....	764
21.2.7 SCI 入出力タイミング .....	765
21.2.8 DMAC タイミング .....	766



## 21.1 H8/3068F-ZTAT 電気的特性

### 21.1.1 絶対最大定格

絶対最大定格を表 21.1 に示します。

表 21.1 絶対最大定格

項目	記号	定格値	単位
電源電圧	$V_{CC}^{*1}$	5V 動作品： - 0.3 ~ + 7.0 3V 動作品： - 0.3 ~ + 4.6	V
入力電圧 (FWE) <sup>*2</sup>	$V_{in}$	- 0.3 ~ $V_{CC} + 0.3$	V
入力電圧 (ポート 7 以外) <sup>*2</sup>	$V_{in}$	- 0.3 ~ $V_{CC} + 0.3$	V
入力電圧 (ポート 7)	$V_{in}$	- 0.3 ~ $AV_{CC} + 0.3$	V
リファレンス電源電圧	$V_{REF}$	- 0.3 ~ $AV_{CC} + 0.3$	V
アナログ電源電圧	$AV_{CC}$	5V 動作品： - 0.3 ~ + 7.0 3V 動作品： - 0.3 ~ + 4.6	V
アナログ入力電圧	$V_{AN}$	- 0.3 ~ $AV_{CC} + 0.3$	V
動作温度	$T_{opr}$	- 20 ~ + 75 <sup>*3</sup>	
保存温度	$T_{stg}$	- 55 ~ + 125	

#### 【使用上の注意】

絶対最大定格を超えて LSI を使用した場合、LSI の永久破壊となることがあります。

【注】 \*1 5V 動作品の  $V_{CL}$  ピンには電源電圧を印加しないでください。GND との間に外付けコンデンサを接続してください。

\*2 いずれの端子にも絶対に 12V を印加しないでください。

12V を印加した場合、LSI の永久破壊となります。

\*3 フラッシュメモリの書き込み / 消去時の動作温度範囲は 0 ~ +75 です。

## 21.1.2 DC 特性

DC 特性を表 21.2 に示します。また、出力許容電流値を表 21.33 に示します。

表 21.2 DC 特性 ( 1 )

条件 :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$

$T_a = -20 \sim +75$  ( 書き込み / 消去時の条件 :  $T_a = 0 \sim +75$  )

項目		記号	min	typ	max	単位	測定条件
シュミット トリガ入力 電圧	P8 <sub>0</sub> ~ P8 <sub>2</sub> 、 PortA	$V_T^-$	1.0	-	-	V	
		$V_T^+$	-	-	$V_{CC} \times 0.7$		
		$V_T^+ - V_T^-$	0.4	-	-		
入力 High レベル電圧	STBY、RES、 NMI、MD <sub>0</sub> ~ MD <sub>2</sub> 、FWE	$V_{IH}$	$V_{CC} - 0.7$	-	$V_{CC} + 0.3$	V	
	EXTAL		$V_{CC} \times 0.7$	-	$V_{CC} + 0.3$		
	Port7		2.0	-	$AV_{CC} + 0.3$		
	Port1 ~ Port6、 P8 <sub>3</sub> 、P8 <sub>4</sub> 、P9 <sub>0</sub> ~ P9 <sub>5</sub> 、PortB		2.0	-	$V_{CC} + 0.3$		
入力 Low レベル電圧	STBY、RES、 FWE、MD <sub>0</sub> ~ MD <sub>2</sub>	$V_{IL}$	- 0.3	-	0.5	V	
	NMI、EXTAL、 Port1 ~ Port7、 P8 <sub>3</sub> 、P8 <sub>4</sub> 、P9 <sub>0</sub> ~ P9 <sub>5</sub> 、PortB		- 0.3	-	0.8		
出力 High レベル電圧	全出力端子	$V_{OH}$	$V_{CC} - 0.5$	-	-	V	$I_{OH} = -200 \mu A$
			3.5	-	-		$I_{OH} = -1mA$
出力 Low レベル電圧	全出力端子	$V_{OL}$	-	-	0.4	V	$I_{OL} = 1.6mA$
	Port1、2、5		-	-	1.0		$I_{OL} = 10mA$
入力リーク 電流	STBY、RES、 NMI、FWE、 MD <sub>0</sub> ~ MD <sub>2</sub>	$I_{in}$	-	-	1.0	$\mu A$	$V_{in} = 0.5V \sim V_{CC} - 0.5V$
	Port7		-	-	1.0		$V_{in} = 0.5V \sim AV_{CC} - 0.5V$
スリーステ ートリーク 電流	Port1 ~ Port6、 Port8 ~ PortB	$I_{TSI}$	-	-	1.0	$\mu A$	$V_{in} = 0.5V \sim V_{CC} - 0.5V$
入力 プルアップ MOS 電流	Port2、4、5	$-I_p$	50	-	300	$\mu A$	$V_{in} = 0V$
入力容量	FWE	$C_{in}$	-	-	80	pF	$V_{in} = 0V$ 、 $f = f_{min}$ 、 $T_a = 25$
	NMI		-	-	50		
	NMI、FWE 以外 の全入力端子		-	-	15		

項目		記号	min	typ	max	単位	測定条件
消費電流	通常動作時	$I_{CC}$	-	32 (5.0V)	47	mA	f = 20MHz
				37 (5.0V)	58		f = 25MHz
	スリープ時		-	24 (5.0V)	38	f = 20MHz	
				29 (5.0V)	47	f = 25MHz	
	モジュールスタンバイ時		-	19 (5.0V)	31	f = 20MHz	
				21 (5.0V)	37	f = 25MHz	
	スタンバイ時		-	1.0	10	$\mu A$	Ta 50
-		-	80	50 < Ta			
フラッシュメモリ書き込み / 消去時*4	-	37	57	mA	f = 20MHz		
		42	68		f = 25MHz		
アナログ電源電流	A/D 変換中	$AI_{CC}$	-	0.6	1.5	mA	
	A/D、D/A 変換中		-	0.6	1.5		
	A/D、D/A 変換待機中		-	0.01	5	$\mu A$	
リファレンス電源電流	A/D 変換中	$AI_{CC}$	-	0.45	0.8	mA	
	A/D、D/A 変換中		-	2.0	3.0		
	A/D、D/A 変換待機中		-	0.01	5	$\mu A$	
RAM スタンバイ電圧		$V_{RAM}$	2.0	-	-	V	

【注】 \*1 A/D 変換器使用時に  $AV_{CC}$ 、 $V_{REF}$ 、 $AV_{SS}$  端子を開放しないでください。

$AV_{CC}$ 、 $V_{REF}$  端子は  $V_{CC}$  に、 $AV_{SS}$  端子は  $V_{SS}$  にそれぞれ接続してください。

\*2 消費電流値は、 $V_{IH} \min = V_{CC} - 0.5V$ 、 $V_{IL} \max = 0.5V$  の条件下で、全ての出力端子を無負荷状態にして、さらに内蔵プルアップ MOS をオフ状態にした場合の値です。

また、 $V_{RAM}$   $V_{CC} < 4.5V$  の時、 $V_{IH} \min = V_{CC} \times 0.9$ 、 $V_{IL} \max = 0.3V$  とした場合の値です。

\*3  $I_{CC} \max.$  (通常動作時) =  $3.0 [mA] + 0.40 [mA / (MHz \times V)] \times V_{CC} \times f$

$I_{CC} \max.$  (スリープ時) =  $3.0 [mA] + 0.32 [mA / (MHz \times V)] \times V_{CC} \times f$

$I_{CC} \max.$  (スリープ + モジュールスタンバイ時)

=  $3.0 [mA] + 0.25 [mA / (MHz \times V)] \times V_{CC} \times f$

また、消費電流の typ 値は参考値です。

\*4 通常動作 + 書き込み / 消去動作の消費電流の合計値です。

21. 電気的特性 ( 暫定仕様 )

表 21.2 DC 特性 ( 2 )

条件 :  $V_{CC} = 3.0 \sim 3.6V$ 、 $AV_{CC} = 3.3 \sim 3.6V$ 、 $V_{REF} = 3.0V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$

$T_a = -20 \sim +75$  ( 書き込み / 消去時の条件 :  $T_a = 0 \sim +75$  )

- 暫定仕様 -

項目	記号	min	typ	max	単位	測定条件	
シュミット トリガ入力 電圧	P80 ~ P82、 PortA	$V_T^-$	$V_{CC} \times 0.2$	-	-	V	
		$V_T^+$	-	-	$V_{CC} \times 0.7$		
		$V_T^+ - V_T^-$	$V_{CC} \times 0.07$	-	-		
入力 High レベル電圧	STBY、RES、 NMI、MD <sub>0</sub> ~ MD <sub>2</sub> 、FWE	$V_{IH}$	$V_{CC} \times 0.9$	-	$V_{CC} + 0.3$	V	
	EXTAL		$V_{CC} \times 0.7$	-	$V_{CC} + 0.3$		
	Port7		$V_{CC} \times 0.7$	-	$AV_{CC} + 0.3$		
	Port1 ~ Port6、 P8 <sub>3</sub> 、P8 <sub>4</sub> 、P9 <sub>0</sub> ~ P9 <sub>5</sub> 、PortB		$V_{CC} \times 0.7$	-	$V_{CC} + 0.3$		
入力 Low レベル電圧	STBY、RES、 FWE、MD <sub>0</sub> ~ MD <sub>2</sub>	$V_{IL}$	- 0.3	-	$V_{CC} \times 0.1$	V	
	NMI、EXTAL、 Port1 ~ Port7、 P8 <sub>3</sub> 、P8 <sub>4</sub> 、P9 <sub>0</sub> ~ P9 <sub>5</sub> 、PortB		- 0.3	-	$V_{CC} \times 0.2$		
					0.8		
出力 High レベル電圧	全出力端子	$V_{OH}$	$V_{CC} - 0.5$	-	-	V	$I_{OH} = -200 \mu A$
			$V_{CC} - 1.0$	-	-		$I_{OH} = -1mA$
出力 Low レベル電圧	全出力端子	$V_{OL}$	-	-	0.4	V	$I_{OL} = 1.6mA$
	Port1、2、5		-	-	1.0		$I_{OL} = 5mA$ ( $V_{CC} < 4.0V$ )
入力リーク 電流	STBY、RES、 NMI、FWE、 MD <sub>0</sub> ~ MD <sub>2</sub>	$I_{in}$	-	-	1.0	$\mu A$	$V_{in} = 0.5V \sim V_{CC} - 0.5V$
	Port7		-	-	1.0		$V_{in} = 0.5V \sim AV_{CC} - 0.5V$
スリーステ ートリーク 電流	Port1 ~ Port6、 Port8 ~ PortB	$I_{TSL}$	-	-	1.0	$\mu A$	$V_{in} = 0.5V \sim V_{CC} - 0.5V$
入力プルア ップ MOS 電流	Port2、4、5	$-I_p$	10	-	300	$\mu A$	$V_{in} = 0V$
入力容量	FWE	$C_{in}$	-	-	80	pF	$V_{in} = 0V$ 、 $f = f_{min}$ 、 $T_a = 25$
	NMI		-	-	50		
	NMI、FWE 以外 の全入力端子		-	-	15		

- 暫定仕様 -

項目	記号	min	typ	max	単位	測定条件	
消費電流	通常動作時	$I_{CC}$	-	TBD (3.5V)	TBD	mA	$f = f_{max}$
	スリープ時		-	TBD (3.5V)	TBD		$f = f_{max}$
	モジュールスタンバイ時		-	TBD (3.5V)	TBD		$f = f_{max}$
	スタンバイ時		-	TBD	10	$\mu A$	$T_a = 50$
			-	-	80		$50 < T_a$
	フラッシュメモリ書き込み / 消去時*4		-	TBD	TBD	mA	$f = f_{max}$
アナログ電源電流	A/D 変換中	$I_{CC}$	-	TBD	0.5	mA	$AV_{CC} = 3.3V$
	A/D、D/A 変換中		-	TBD	0.5		$AV_{CC} = 3.3V$
	A/D、D/A 変換待機中		-	TBD	5	$\mu A$	DASTE = 0 時
リファレンス電源電流	A/D 変換中	$I_{CC}$	-	TBD	0.5	mA	$V_{REF} = 3.0V$
	A/D、D/A 変換中		-	TBD	2.0		$V_{REF} = 3.0V$
	A/D、D/A 変換待機中		-	TBD	5	$\mu A$	DASTE = 0 時
RAM スタンバイ電圧	$V_{RAM}$	2.0	-	-	V		

【注】 \*1 A/D 変換器使用時に  $AV_{CC}$ 、 $V_{REF}$ 、 $AV_{SS}$  端子を開放しないでください。

$AV_{CC}$ 、 $V_{REF}$  端子は  $V_{CC}$  に、 $AV_{SS}$  端子は  $V_{SS}$  にそれぞれ接続してください。

\*2 消費電流値は、 $V_{IH} \min = V_{CC} - 0.5V$ 、 $V_{IL} \max = 0.5V$  の条件下で、全ての出力端子を無負荷状態にして、さらに内蔵プルアップ MOS をオフ状態にした場合の値です。

また、 $V_{RAM}$   $V_{CC} < 3.0V$  の時、 $V_{IH} \min = V_{CC} \times 0.9$ 、 $V_{IL} \max = 0.3V$  とした場合の値です。

\*3  $I_{CC} \max.$  (通常動作時) =  $3.0 [mA] + 0.40 [mA / (MHz \times V)] \times V_{CC} \times f$

$I_{CC} \max.$  (スリープ時) =  $3.0 [mA] + 0.32 [mA / (MHz \times V)] \times V_{CC} \times f$

$I_{CC} \max.$  (スリープ + モジュールスタンバイ時)

=  $3.0 [mA] + 0.25 [mA / (MHz \times V)] \times V_{CC} \times f$

また、消費電流の typ 値は参考値です。

\*4 通常動作 + 書き込み / 消去動作の消費電流の合計です。



表 21.3 出力許容電流値

条件 :  $V_{CC} = 3.0 \sim 5.5V$ 、 $AV_{CC} = 3.3 \sim 5.5V$ 、 $V_{REF} = 3.0V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
 $T_a = -20 \sim +75$

項目		記号	min	typ	max	単位
出力 Low レベル許容電流 (1 端子あたり)	ポート 1、2、5	$I_{OL}$	-	-	10	mA
	上記以外の出力端子		-	-	2.0	
出力 Low レベル許容電流 (総和)	ポート 1、2、5、20 端子の総和	$I_{OL}$	-	-	80	mA
	上記を含む、全出力端子の総和		-	-	120	
出力 High レベル許容電流 (1 端子あたり)	全出力端子	$ I_{OH} $	-	-	2.0	mA
出力 High レベル許容電流 (総和)	全出力端子の総和	$ I_{OH} $	-	-	40	mA

- 【注】 1. LSI の信頼性を確保するため、出力電流値は、表 21.3 の値を超えないようにしてください。
2. ダーリントトランジスタや、LED を直接駆動する場合には、図 21.1、図 21.2 に示すように、出力に必ず電流制限抵抗を挿入してください。

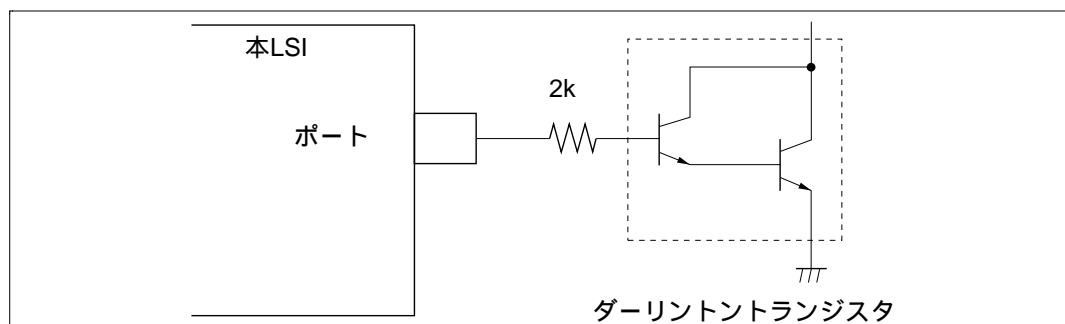


図 21.1 ダーリントトランジスタ駆動回路例

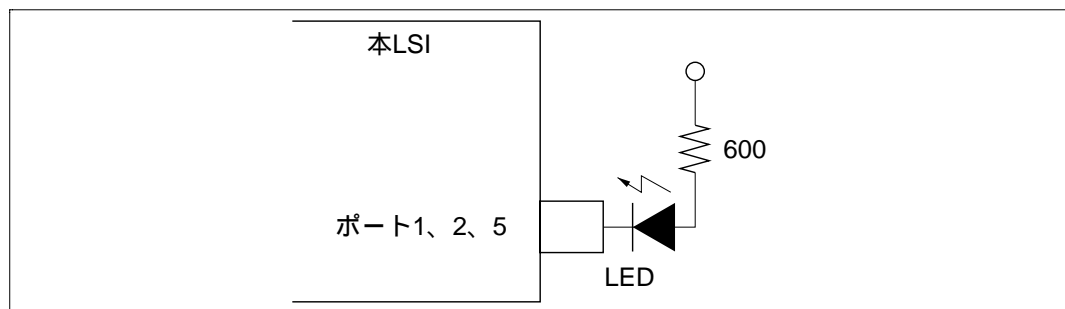


図 21.2 LED 回路例

### 21.1.3 AC 特性

表 21.4 にクロックタイミング、表 21.5 に制御信号タイミング、表 21.6 にバスタイミングを示します。

また、表 21.7 に内蔵周辺モジュールタイミングを示します。

表 21.4 クロックタイミング

全条件 :  $T_a = -20 \sim +75$

条件 A :  $V_{CC} = 3.0 \sim 3.6V$ 、 $AV_{CC} = 3.3 \sim 3.6V$ 、 $V_{REF} = 3.0 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = TBD

条件 B :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 20MHz

条件 C :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 25MHz

項目	記号	条件 A		条件 B		条件 C		単位	測定条件
		min	max	min	max	min	max		
クロックサイクル時間	$t_{cyc}$	TBD	500	50	500	40	500	ns	図 21.4 ~ 図 21.6
クロックパルス幅 (Low)	$t_{CL}$	18	-	15	-	10	-	ns	
クロックパルス幅 (High)	$t_{CH}$	18	-	15	-	10	-	ns	
クロック立ち上がり時間	$t_{Cr}$	-	15	-	10	-	10	ns	
クロック立ち下がり時間	$t_{Cf}$	-	15	-	10	-	10	ns	
リセット発振安定時間	$t_{OSC1}$	20	-	20	-	20	-	ms	図 21.4
ソフトウェアスタンバイ発振安定時間	$t_{OSC2}$	7	-	7	-	7	-	ms	

表 21.5 制御信号タイミング

全条件 :  $T_a = -20 \sim +75$

条件 A :  $V_{CC} = 3.0 \sim 3.6V$ 、 $AV_{CC} = 3.3 \sim 3.6V$ 、 $V_{REF} = 3.0 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = TBD

条件 B :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 20MHz

条件 C :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 25MHz

項目	記号	条件 A		条件 B、条件 C		単位	測定条件
		min	max	min	max		
RES セットアップ時間	$t_{RESS}$	200	-	150	-	ns	図 21.14
RES パルス幅	$t_{RESW}$	20	-	20	-	$t_{cyc}$	
モードプログラミング セットアップ時間	$t_{MDS}$	200	-	200	-	ns	
NMI、IRQ セットアップ時間	$t_{NMIS}$	200	-	150	-	ns	図 21.16
NMI、IRQ ホールド時間	$t_{NMIH}$	10	-	10	-	ns	
NMI、IRQ パルス幅	$t_{NMIW}$	200	-	200	-	ns	

21. 電気的特性 (暫定仕様)

表 21.6 バスタイミング

全条件 :  $T_a = -20 \sim +75$

条件 A :  $V_{CC} = 3.0 \sim 3.6V$ 、 $AV_{CC} = 3.3 \sim 3.6V$ 、 $V_{REF} = 3.0 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = TBD

条件 B :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 20MHz

条件 C :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 25MHz

項目	記号	条件 A		条件 B、条件 C		単位	測定条件
		min	max	min	max		
アドレス遅延時間	$t_{AD}$	-	40	-	25	ns	図 21.17、 図 21.18
アドレスホールド時間	$t_{AH}$	$0.5t_{cyc} - 35$	-	$0.5t_{cyc} - 20$	-	ns	
リードストロープ遅延時間	$t_{RSD}$	-	50	-	25	ns	
アドレスストロープ遅延時間	$t_{ASD}$	-	50	-	25	ns	
ライトストロープ遅延時間	$t_{WSD}$	-	50	-	25	ns	
ストロープ遅延時間	$t_{SD}$	-	50	-	25	ns	
ライトストロープパルス幅 1	$t_{WSW1}$	$1.0t_{cyc} - 40$	-	$1.0t_{cyc} - 25$	-	ns	
ライトストロープパルス幅 2	$t_{WSW2}$	$1.5t_{cyc} - 40$	-	$1.5t_{cyc} - 25$	-	ns	
アドレスセットアップ時間 1	$t_{AS1}$	$0.5t_{cyc} - 29$	-	$0.5t_{cyc} - 20$	-	ns	
アドレスセットアップ時間 2	$t_{AS2}$	$1.0t_{cyc} - 35$	-	$1.0t_{cyc} - 20$	-	ns	
リードデータセットアップ時間	$t_{RDS}$	40	-	25	-	ns	
リードデータホールド時間	$t_{RDH}$	0	-	0	-	ns	
ライトデータ遅延時間	$t_{WDD}$	-	50	-	35	ns	
ライトデータセットアップ時間 1	$t_{WDS1}$	$1.0t_{cyc} - 40$	-	$1.0t_{cyc} - 30$	-	ns	
ライトデータセットアップ時間 2	$t_{WDS2}$	$2.0t_{cyc} - 40$	-	$2.0t_{cyc} - 30$	-	ns	
ライトデータホールド時間	$t_{WDH}$	$0.5t_{cyc} - 25$	-	$0.5t_{cyc} - 15$	-	ns	
リードデータアクセス時間 1	$t_{ACC1}$	-	$2.0t_{cyc} - 80$	-	$2.0t_{cyc} - 45$	ns	
リードデータアクセス時間 2	$t_{ACC2}$	-	$3.0t_{cyc} - 80$	-	$3.0t_{cyc} - 45$	ns	
リードデータアクセス時間 3	$t_{ACC3}$	-	$1.5t_{cyc} - 80$	-	$1.5t_{cyc} - 45$	ns	
リードデータアクセス時間 4	$t_{ACC4}$	-	$2.5t_{cyc} - 80$	-	$2.5t_{cyc} - 45$	ns	
プリチャージ時間 1	$t_{PCH1}$	$1.0t_{cyc} - 30$	-	$1.0t_{cyc} - 20$	-	ns	
プリチャージ時間 2	$t_{PCH2}$	$0.5t_{cyc} - 30$	-	$0.5t_{cyc} - 20$	-	ns	
ウェイトセットアップ時間	$t_{WTS}$	40	-	25	-	ns	図 21.19
ウェイトホールド時間	$t_{WTH}$	5	-	5	-	ns	
バスリクエストセットアップ時間	$t_{BRQS}$	40	-	25	-	ns	図 21.20
バスアクノリッジ遅延時間 1	$t_{BACD1}$	-	50	-	30	ns	
バスアクノリッジ遅延時間 2	$t_{BACD2}$	-	50	-	30	ns	
バスフローティング時間	$t_{BZD}$	-	50	-	30	ns	
RAS プリチャージ時間	$t_{RP}$	$1.5t_{cyc} - 40$	-	$1.5t_{cyc} - 25$	-	ns	図 21.17
CAS プリチャージ時間	$t_{CP}$	$0.5t_{cyc} - 25$	-	$0.5t_{cyc} - 15$	-	ns	~
ロウアドレスホールド時間	$t_{RAH}$	$0.5t_{cyc} - 25$	-	$0.5t_{cyc} - 15$	-	ns	図 21.19
RAS 遅延時間 1	$t_{RAD1}$	-	50	-	25	ns	
RAS 遅延時間 2	$t_{RAD2}$	-	50	-	30	ns	
CAS 遅延時間 1	$t_{CASD1}$	-	50	-	25	ns	
CAS 遅延時間 2	$t_{CASD2}$	-	50	-	25	ns	
WE 遅延時間	$t_{WCD}$	-	50	-	25	ns	
CAS パルス幅 1	$t_{CAS1}$	$1.5t_{cyc} - 40$	-	$1.5t_{cyc} - 20$	-	ns	
CAS パルス幅 2	$t_{CAS2}$	$1.0t_{cyc} - 40$	-	$1.0t_{cyc} - 20$	-	ns	
CAS パルス幅 3	$t_{CAS3}$	$1.0t_{cyc} - 40$	-	$1.0t_{cyc} - 20$	-	ns	
RAS アクセス時間	$t_{RAC}$	-	$2.5t_{cyc} - 70$	-	$2.5t_{cyc} - 40$	ns	
アドレスアクセス時間	$t_{AA}$	-	$2.0t_{cyc} - 80$	-	$2.0t_{cyc} - 50$	ns	
CAS アクセス時間	$t_{CAC}$	-	$1.5t_{cyc} - 80$	-	$1.5t_{cyc} - 50$	ns	

項目	記号	条件 A		条件 B、条件 C		単位	測定条件
		min	max	min	max		
WE セットアップ時間	$t_{WCS}$	$0.5t_{cyc} - 35$	-	$0.5t_{cyc} - 20$	-	ns	
WE ホールド時間	$t_{WCH}$	$0.5t_{cyc} - 28$	-	$0.5t_{cyc} - 15$	-	ns	図 21.17 ~ 図 21.19
ライトデータ セットアップ時間	$t_{WDS}$	$0.5t_{cyc} - 35$	-	$0.5t_{cyc} - 20$	-	ns	
WE ライトデータ ホールド時間	$t_{WDH}$	$0.5t_{cyc} - 25$	-	$0.5t_{cyc} - 15$	-	ns	
CAS セットアップ時間 1	$t_{CSR1}$	$0.5t_{cyc} - 25$	-	$0.5t_{cyc} - 20$	-	ns	
CAS セットアップ時間 2	$t_{CSR2}$	$0.5t_{cyc} - 25$	-	$0.5t_{cyc} - 15$	-	ns	
CAS ホールド時間	$t_{CHR}$	$0.5t_{cyc} - 25$	-	$0.5t_{cyc} - 15$	-	ns	
RAS パルス幅	$t_{RAS}$	$1.5t_{cyc} - 25$	-	$1.5t_{cyc} - 15$	-	ns	

【注】  $\overline{RD}$  ストロープの立ち上がりに対するアドレスのホールドを確保するためには、アドレス更新モード 2 を使用してください。詳細は「6.3.5 アドレス出力方式」を参照してください。

21. 電気的特性 (暫定仕様)

表 21.7 内蔵周辺モジュールタイミング

全条件 :  $T_a = -20 \sim +75$

条件 A :  $V_{CC} = 3.0 \sim 3.6V$ 、 $AV_{CC} = 3.3 \sim 3.6V$ 、 $V_{REF} = 3.0 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = TBD

条件 B :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 20MHz

条件 C :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 25MHz

Module	項目	記号	条件 A		条件 B、条件 C		単位	測定条件	
			min	max	min	max			
Port/TPC	出力データ遅延時間	$t_{PWD}$	-	100	-	50	ns	図 21.21	
	入力データセットアップ時間	$t_{PRS}$	50	-	50	-	ns		
	入力データホールド時間	$t_{PRH}$	50	-	50	-	ns		
16 ビット タイマ	タイマ出力遅延時間	$t_{TOCD}$	-	100	-	50	ns	図 21.22	
	タイマ入力セットアップ時間	$t_{TICS}$	50	-	50	-	ns		
	タイマクロック入力セットアップ時間	$t_{TCKS}$	50	-	50	-	ns	図 21.23	
	タイマクロック パルス幅	単エッジ	$t_{TCKWH}$	1.5	-	1.5	-		$t_{cyc}$
		両エッジ	$t_{TCKWL}$	2.5	-	2.5	-		$t_{cyc}$
8 ビット タイマ	タイマ出力遅延時間	$t_{TOCD}$	-	100	-	50	ns	図 21.22	
	タイマ入力セットアップ時間	$t_{TICS}$	50	-	50	-	ns		
	タイマクロック入力セットアップ時間	$t_{TCKS}$	50	-	50	-	ns	図 21.23	
	タイマクロック パルス幅	単エッジ	$t_{TCKWH}$	1.5	-	1.5	-		$t_{cyc}$
		両エッジ	$t_{TCKWL}$	2.5	-	2.5	-		$t_{cyc}$
SCI	入力クロック サイクル	調歩同期	$t_{Scyc}$	4	-	4	-	$t_{cyc}$	図 21.24
		クロック同期		6	-	6	-		
	入力クロック立ち上がり時間	$t_{SCKr}$	1.5	-	1.5	-	$t_{cyc}$		
	入力クロック立ち下がり時間	$t_{SCKf}$	1.5	-	1.5	-	$t_{cyc}$		
	入力クロックパルス幅	$t_{SCKW}$	0.4	0.6	0.4	0.6	$t_{Scyc}$		
	送信データ遅延時間	$t_{TXD}$	-	100	-	100	ns	図 21.25	
	受信データセットアップ時間 (クロック同期)	$t_{RXS}$	100	-	100	-	ns		
	受信データホールド 時間 (クロック同期)	クロック入力	$t_{RXH}$	100	-	100	-		ns
クロック出力			0	-	0	-			
DMAC	DREQ セットアップ時間	$t_{DRQS}$	40	-	25	-	ns	図 21.27	
	DREQ ホールド時間	$t_{DRQH}$	10	-	10	-	ns		
	TEND 遅延時間 1	$t_{TED1}$	-	100	-	50	ns	図 21.25、 図 21.26	
	TEND 遅延時間 2	$t_{TED2}$	-	100	-	50	ns		

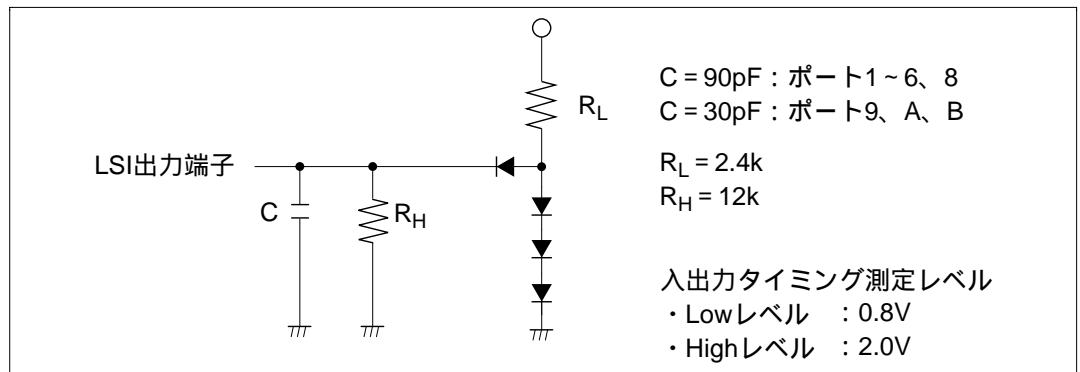


図 21.3 出力負荷回路

### 21.1.4 A/D 変換特性

A/D 変換特性を表 21.8 に示します。

表 21.8 A/D 変換特性

全条件 :  $T_a = -20 \sim +75$

条件 A :  $V_{CC} = 3.0 \sim 3.6V$ 、 $AV_{CC} = 3.3 \sim 3.6V$ 、 $V_{REF} = 3.0 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
 $f_{max} = TBD$

条件 B :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
 $f_{max} = 20MHz$

条件 C :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
 $f_{max} = 25MHz$

項目		条件 A			条件 B、条件 C			単位	
		min	typ	max	min	typ	max		
変換時間 : 134 ステート	分解能	10	10	10	10	10	10	bit	
	変換時間 (単一モード)	-	-	134	-	-	134	$t_{cyc}$	
	アナログ入力容量	-	-	20	-	-	20	pF	
	許容信号源 インピー ダンス	13MHz	-	-	-	-	-	10	k
		> 13MHz	-	-	-	-	-	5	
	ダンス	4.0V $AV_{CC} \geq 5.5V$	-	-	10	-	-	-	k
		3.0V $AV_{CC} < 4.0V$	-	-	5	-	-	-	
	非直線性誤差	-	-	$\pm 7.5$	-	-	$\pm 3.5$	LSB	
	オフセット誤差	-	-	$\pm 7.5$	-	-	$\pm 3.5$	LSB	
	フルスケール誤差	-	-	$\pm 7.5$	-	-	$\pm 3.5$	LSB	
	量子化誤差	-	-	$\pm 0.5$	-	-	$\pm 0.5$	LSB	
	絶対精度	-	-	$\pm 8.0$	-	-	$\pm 4.0$	LSB	
	変換時間 : 70 ステート	分解能	10	10	10	10	10	10	bit
		変換時間 (単一モード)	-	-	70	-	-	70	$t_{cyc}$
アナログ入力容量		-	-	20	-	-	20	pF	
許容信号源 インピー ダンス		13MHz	-	-	-	-	-	5	k
		> 13MHz	-	-	-	-	-	3	
ダンス		4.0V $AV_{CC} \geq 5.5V$	-	-	5	-	-	-	k
		3.0V $AV_{CC} < 4.0V$	-	-	3	-	-	-	
非直線性誤差		-	-	$\pm 15.5$	-	-	$\pm 7.5$	LSB	
オフセット誤差		-	-	$\pm 15.5$	-	-	$\pm 7.5$	LSB	
フルスケール誤差		-	-	$\pm 15.5$	-	-	$\pm 7.5$	LSB	
量子化誤差		-	-	$\pm 0.5$	-	-	$\pm 0.5$	LSB	
絶対精度		-	-	$\pm 16$	-	-	$\pm 8.0$	LSB	

## 21.1.5 D/A 変換特性

D/A 変換特性を表 21.9 に示します。

表 21.9 D/A 変換特性

全条件 :  $T_a = -20 \sim +75$

条件 A :  $V_{CC} = 3.0 \sim 3.6V$ 、 $AV_{CC} = 3.3 \sim 3.6V$ 、 $V_{REF} = 3.0 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = TBD

条件 B :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 20MHz

条件 C :  $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5 \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、  
fmax = 25MHz

項目	条件 A			条件 B、条件 C			単位	測定条件
	min	typ	max	min	typ	max		
分解能	8	8	8	8	8	8	bit	
変換時間 (センタリング時間)	-	-	10	-	-	10	$\mu s$	負荷容量 20pF
絶対精度	-	$\pm 2.0$	$\pm 3.0$	-	$\pm 1.5$	$\pm 2.0$	LSB	負荷抵抗 2M
	-	-	$\pm 2.0$	-	-	$\pm 1.5$		負荷抵抗 4M



## 21.1.6 フラッシュメモリ特性

表 21.10 にフラッシュメモリ特性を示します。

表 21.10 フラッシュメモリ特性 ( 1 )

条件 :  $V_{CC} = 4.5 \sim 5.5V$ 、 $AV_{CC} = 4.5 \sim 5.5V$ 、 $V_{SS} = AV_{SS} = 0V$

$T_a = 0 \sim +75$  ( 書き込み / 消去時の動作温度範囲 )

項目		記号	min.	typ.	max.	単位	特記
書き込み時間*1*2*4		$t_p$	-	10	200	ms/128 バイト	
消去時間*1*3*5		$t_E$	-	100	1200	ms/ブロック	
書き換え回数		$N_{WEC}$	-	-	100	回	
書き込み時	SWE ビットセット後のウェイト時間*1	$t_{sswe}$	1	1	-	$\mu s$	
	PSU ビットセット後のウェイト時間*1	$t_{psu}$	50	50	-	$\mu s$	
	P ビットセット後のウェイト時間*1*4	$t_{sp30}$	28	30	32	$\mu s$	書き込み時間ウェイト
		$t_{sp200}$	198	200	202	$\mu s$	書き込み時間ウェイト
		$t_{sp10}$	8	10	12	$\mu s$	追加書き込み時間ウェイト
	P ビットクリア後のウェイト時間*1	$t_{cp}$	5	5	-	$\mu s$	
	PSU ビットクリア後のウェイト時間*1	$t_{cpsu}$	5	5	-	$\mu s$	
	PV ビットセット後のウェイト時間*1	$t_{spv}$	4	4	-	$\mu s$	
	H'FF ダミーライト後のウェイト時間*1	$t_{spvr}$	2	2	-	$\mu s$	
	PV ビットクリア後のウェイト時間*1	$t_{cpv}$	2	2	-	$\mu s$	
	SWE ビットクリア後のウェイト時間*1	$t_{cswe}$	100	100	-	$\mu s$	
	最大書き込み回数*1*4	$N$	-	-	1000	回	
	消去時	SWE ビットセット後のウェイト時間*1	$t_{sswe}$	1	1	-	$\mu s$
ESU ビットセット後のウェイト時間*1		$t_{sesu}$	100	100	-	$\mu s$	
E ビットセット後のウェイト時間*1*5		$t_{se}$	10	10	100	ms	消去時間ウェイト
E ビットクリア後のウェイト時間*1		$t_{ce}$	10	10	-	$\mu s$	
ESU ビットクリア後のウェイト時間*1		$t_{cesu}$	10	10	-	$\mu s$	
EV ビットセット後のウェイト時間*1		$t_{sev}$	20	20	-	$\mu s$	
H'FF ダミーライト後のウェイト時間*1		$t_{sevr}$	2	2	-	$\mu s$	
EV ビットクリア後のウェイト時間*1		$t_{cev}$	4	4	-	$\mu s$	
SWE ビットクリア後のウェイト時間*1		$t_{cswe}$	100	100	-	$\mu s$	
最大消去回数*1*5		$N$	12	-	120	回	

【注】 \*1 各時間の設定は、書き込み / 消去のアルゴリズムに従って行ってください。

\*2 128 バイト単位の書き込み時間 ( フラッシュメモリコントロールレジスタ ( FLPCR1 ) の P ビットをセットしているトータル期間を示します。プログラムベリファイ時間は含まれません。 )

\*3 1 ブロックを消去する時間 ( FLPCR1 の E ビットをセットしているトータル期間を示します。イレースベリファイ時間は含まれません。 )

\*4 128 バイト書き込みアルゴリズムにおいて書き込み時間の最大値 ( $t_p(\max)$ ) を規定するために、最大書き込み回数 ( $N$ ) の値は max. 値 (1000) を設定してください。

また P ビットセット後のウェイト時間は、下記のように書き込み回数カウンタ ( $n$ ) の回数によって切り替えてください。

書き込み回数カウンタ ( $n$ ) 1~6 回の場合  $t_{sp30} = 30 \mu s$

書き込み回数カウンタ ( $n$ ) 7~1000 回の場合  $t_{sp200} = 200 \mu s$

[追加書き込み時]

書き込み回数カウンタ ( $n$ ) 1~6 回の場合  $t_{sp10} = 10 \mu s$

\*5 消去時間の最大値 ( $t_e(\max)$ ) に対して、E ビットセット後のウェイト時間 ( $t_{se}$ ) と最大消去回数 ( $N$ ) は以下の関係にあります。

$t_e(\max) = E$  ビットセット後のウェイト時間 ( $t_{se}$ )  $\times$  最大消去回数 ( $N$ )

消去時間の最大値を規定するために、 $t_{se}$  および  $N$  の値は上記計算式を満たすように設定してください。

(例)  $t_{se} = 100$  [ms] の場合、 $N = 12$  回

(例)  $t_{se} = 10$  [ms] の場合、 $N = 120$  回

表 21.10 フラッシュメモリ特性 ( 2 )

条件 :  $V_{CC} = 3.0 \sim 3.6 V$ 、 $AV_{CC} = 3.3 \sim 3.6 V$ 、 $V_{SS} = AV_{SS} = 0V$

$T_a = 0 \sim +75$  ( 書き込み / 消去時の動作温度範囲 )

- 暫定仕様 -

項目		記号	min.	typ.	max.	単位	特記
書き込み時間*1*2*4		$t_p$	-	10	200	ms/128 バイト	
消去時間*1*3*5		$t_e$	-	100	1200	ms/ブロック	
書き換え回数		$N_{WEC}$	-	-	100	回	
書き込み時	SWE ビットセット後のウェイト時間*1	$t_{sswe}$	1	1	-	$\mu s$	
	PSU ビットセット後のウェイト時間*1	$t_{spsu}$	50	50	-	$\mu s$	
	P ビットセット後のウェイト時間*1*4	$t_{sp30}$	28	30	32	$\mu s$	書き込み時間ウェイト
		$t_{sp200}$	198	200	202	$\mu s$	書き込み時間ウェイト
		$t_{sp10}$	8	10	12	$\mu s$	追加書き込み時間ウェイト
	P ビットクリア後のウェイト時間*1	$t_{cp}$	5	5	-	$\mu s$	
	PSU ビットクリア後のウェイト時間*1	$t_{cpsu}$	5	5	-	$\mu s$	
	PV ビットセット後のウェイト時間*1	$t_{spv}$	4	4	-	$\mu s$	
	H'FF ダミーライト後のウェイト時間*1	$t_{spvr}$	2	2	-	$\mu s$	
	PV ビットクリア後のウェイト時間*1	$t_{cpv}$	2	2	-	$\mu s$	
	SWE ビットクリア後のウェイト時間*1	$t_{cswe}$	100	100	-	$\mu s$	
最大書き込み回数*1*4		N	-	-	1000	回	
消去時	SWE ビットセット後のウェイト時間*1	$t_{sswe}$	1	1	-	$\mu s$	
	ESU ビットセット後のウェイト時間*1	$t_{sesu}$	100	100	-	$\mu s$	
	E ビットセット後のウェイト時間*1*5	$t_{se}$	10	10	100	ms	消去時間ウェイト
	E ビットクリア後のウェイト時間*1	$t_{ce}$	10	10	-	$\mu s$	
	ESU ビットクリア後のウェイト時間*1	$t_{cesu}$	10	10	-	$\mu s$	
	EV ビットセット後のウェイト時間*1	$t_{sevr}$	20	20	-	$\mu s$	
	H'FF ダミーライト後のウェイト時間*1	$t_{sevr}$	2	2	-	$\mu s$	
	EV ビットクリア後のウェイト時間*1	$t_{cev}$	4	4	-	$\mu s$	
	SWE ビットクリア後のウェイト時間*1	$t_{cswe}$	100	100	-	$\mu s$	
	最大消去回数*1*5		N	12	-	120	回

【注】 \*1 各時間の設定は、書き込み / 消去のアルゴリズムに従い行ってください。

\*2 128 バイト単位の書き込み時間 ( フラッシュメモリコントロールレジスタ ( FLMCR1 ) の P ビットをセットしているトータル期間を示します。プログラムベリファイ時間は含まれません。 )

\*3 1 ブロックを消去する時間 ( FLMCR1 の E ビットをセットしているトータル期間を示します。イレースベリファイ時間は含まれません。 )

\*4 128 バイト書き込みアルゴリズムにおいて書き込み時間の最大値 ( $t_p(\max)$ ) を規定するために、最大書き込み回数 ( $N$ ) の値は max. 値 (1000) を設定してください。

また P ビットセット後のウェイト時間は、下記のように書き込み回数カウンタ ( $n$ ) の回数によって切り替えてください。

書き込み回数カウンタ ( $n$ ) 1~6 回の場合  $t_{sp30} = 30 \mu s$

書き込み回数カウンタ ( $n$ ) 7~1000 回の場合  $t_{sp200} = 200 \mu s$

〔追加書き込み時〕

書き込み回数カウンタ ( $n$ ) 1~6 回の場合  $t_{sp10} = 10 \mu s$

\*5 消去時間の最大値 ( $t_e(\max)$ ) に対して、E ビットセット後のウェイト時間 ( $t_{se}$ ) と最大消去回数 ( $N$ ) は以下の関係にあります。

$t_e(\max) = E$  ビットセット後のウェイト時間 ( $t_{se}$ )  $\times$  最大消去回数 ( $N$ )

消去時間の最大値を規定するために、 $t_{se}$  および  $N$  の値は上記計算式を満たすように設定してください。

(例)  $t_{se} = 100$  [ms] の場合、 $N = 12$  回

(例)  $t_{se} = 10$  [ms] の場合、 $N = 120$  回

## 21.2 動作タイミング

動作タイミングを以下に示します。

### 21.2.1 クロックタイミング

クロックタイミングを以下に示します。

#### (1) 発振安定時間タイミング

図 21.4 に発振安定時間タイミングを示します。

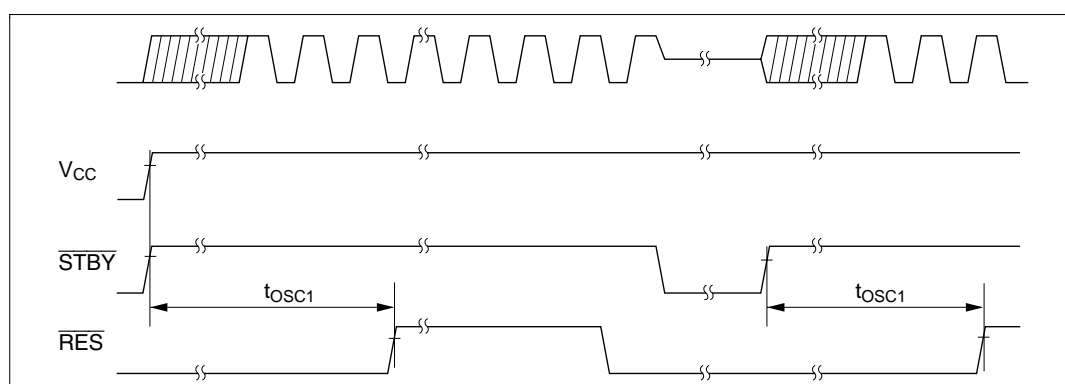


図 21.4 発振安定時間タイミング

## 21.2.2 制御信号タイミング

制御信号タイミングを以下に示します。

### (1) リセット入力タイミング

図 21.5 にリセット入力タイミングを示します。

### (2) リセット出力タイミング\*

図 21.6 にリセット出力タイミングを示します。

### (3) 割り込み入力タイミング

図 21.7 に NMI、 $\overline{\text{IRQ}}_5 \sim \overline{\text{IRQ}}_0$  割り込み入力タイミングを示します。

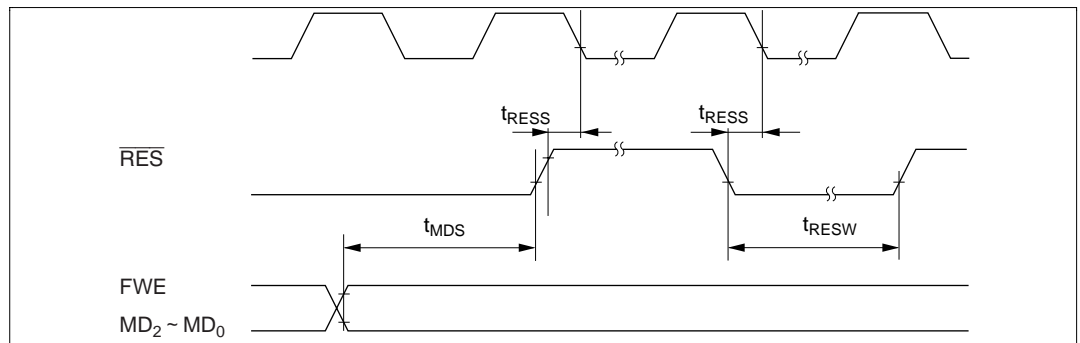


図 21.5 リセット入力タイミング

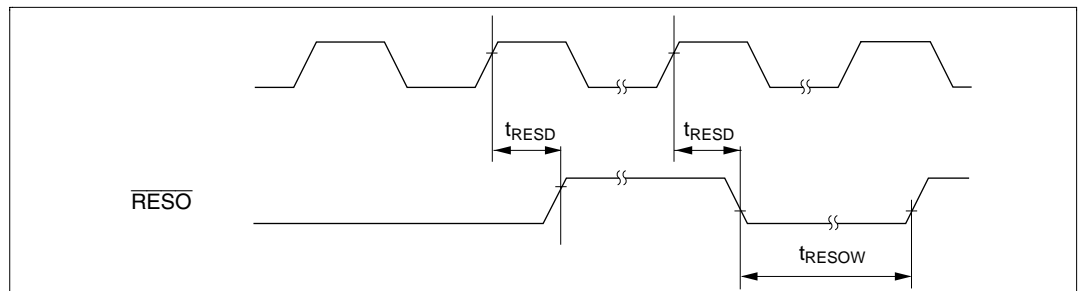


図 21.6 リセット出力タイミング\*

【注】 \* マスク ROM 内蔵製品の機能です。フラッシュメモリ内蔵製品には本機能は存在しません。

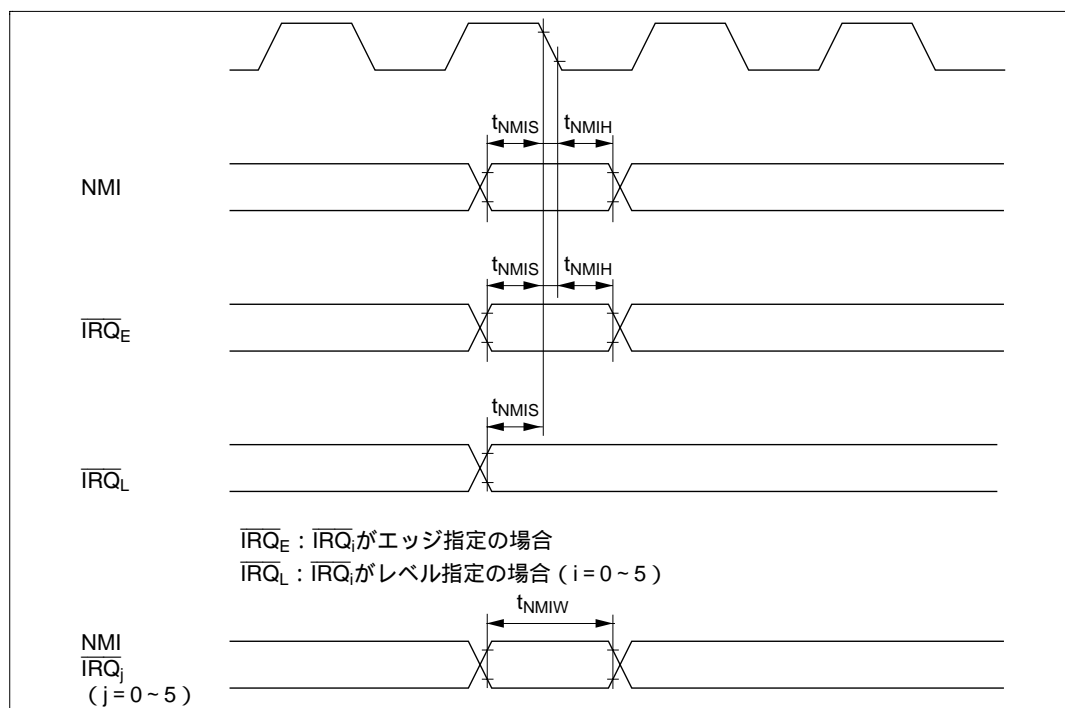


図 21.7 割り込み入力タイミング

### 21.2.3 バスタイミング

バスタイミングを以下に示します。

(1) 基本バスタイミング / 2 ステートアクセス

図 21.8 に外部 2 ステートアクセス時の動作タイミングを示します。

(2) 基本バスタイミング / 3 ステートアクセス

図 21.9 に外部 3 ステートアクセス時の動作タイミングを示します。

(3) 基本バスタイミング / 3 ステートアクセス 1 ウェイト

図 21.10 に外部 3 ステートアクセスで 1 ウェイトを挿入したときの動作タイミングを示します。

(4) バスリリースモードタイミング

図 21.11 にバスリリースモードタイミングを示します。

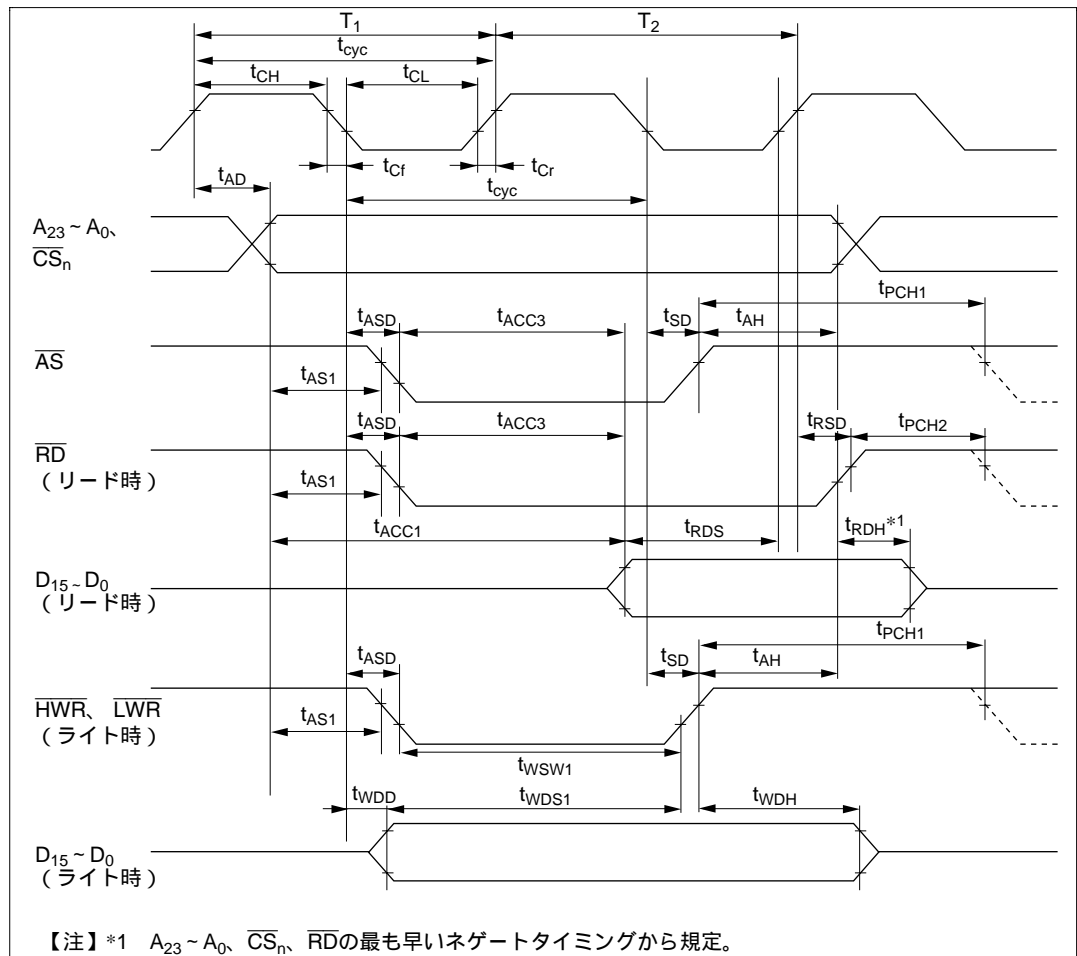


図 21.8 基本バスタイミング / 2 ステートアクセス



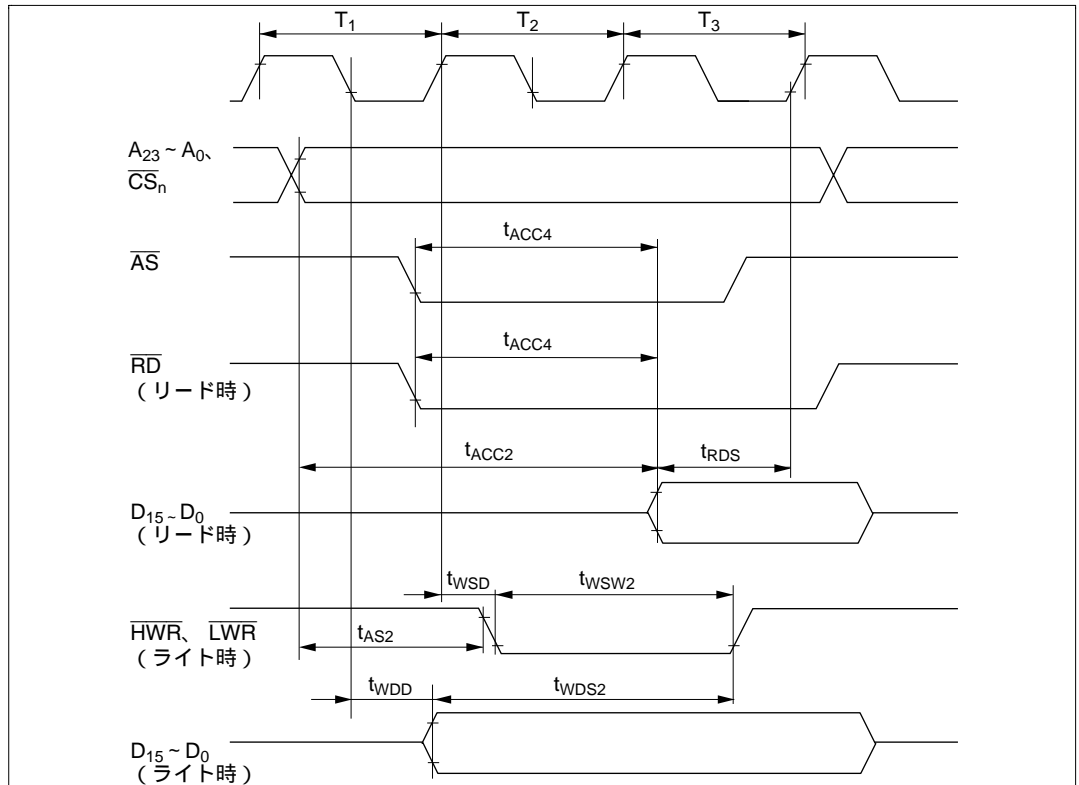


図 21.9 基本バスタイミング / 3 ステートアクセス

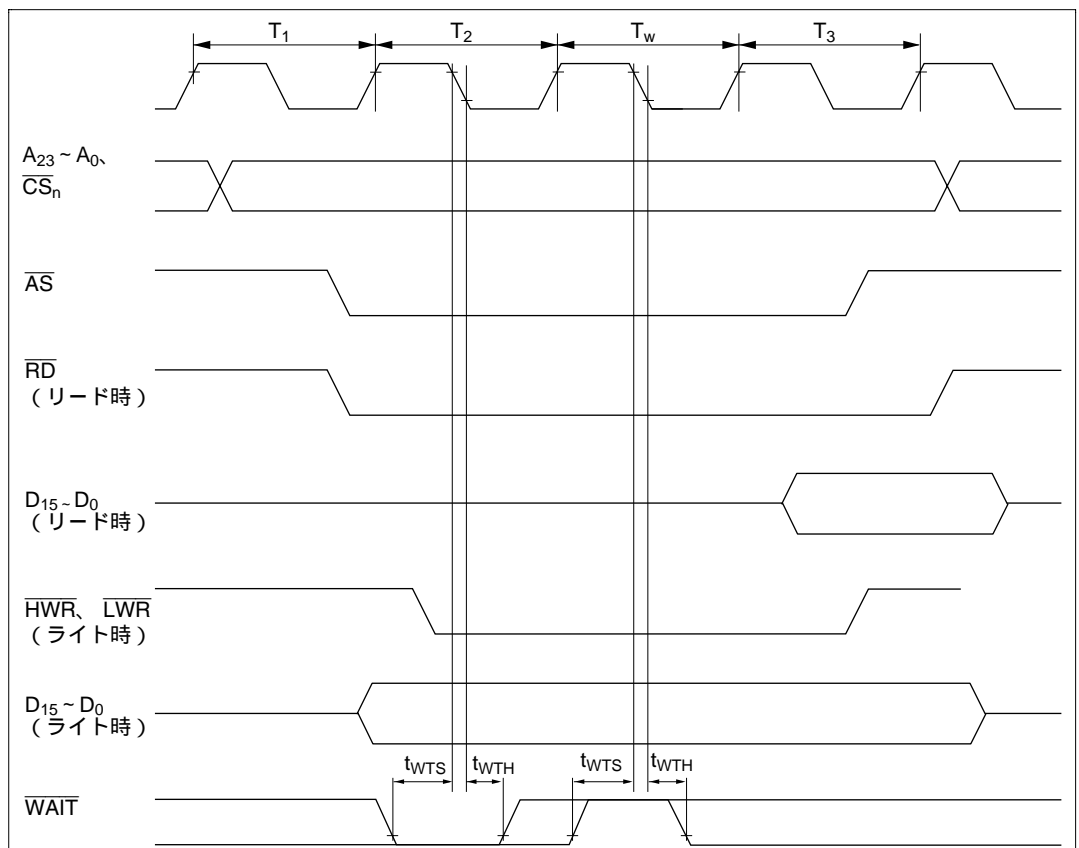


図 21.10 基本バスタイミング / 3 ステートアクセス 1 ウェイト

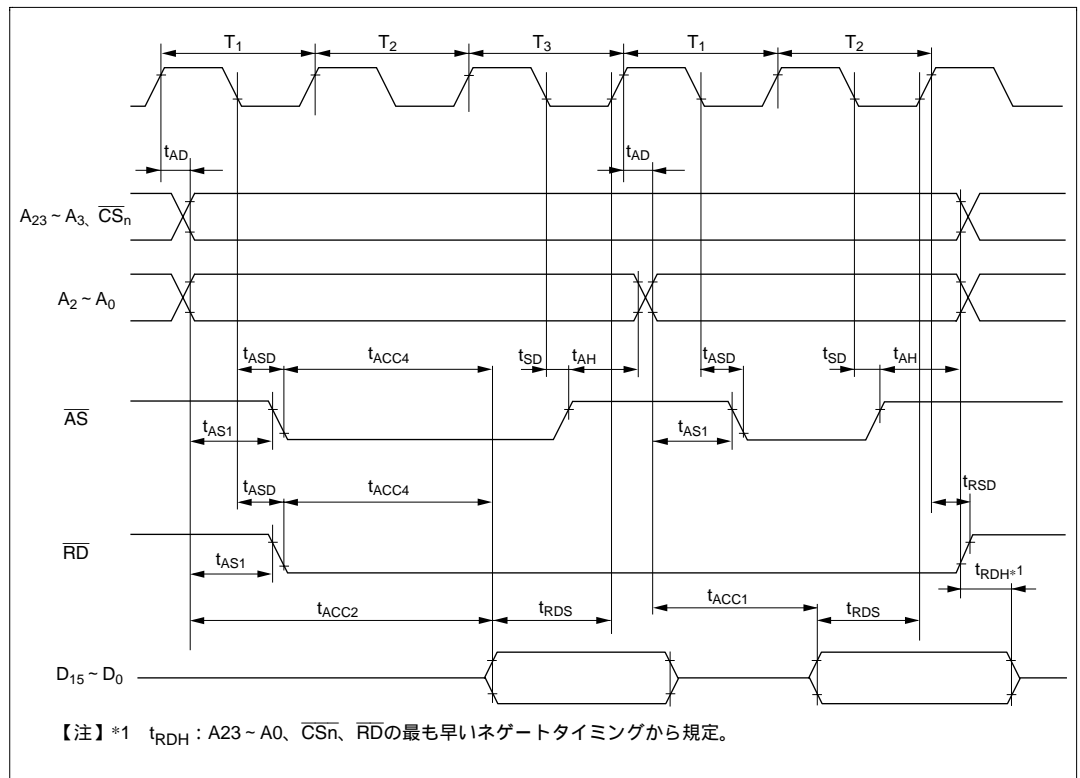


図 21.11 バーストROMアクセスタイミング (バーストサイクル2ステート)

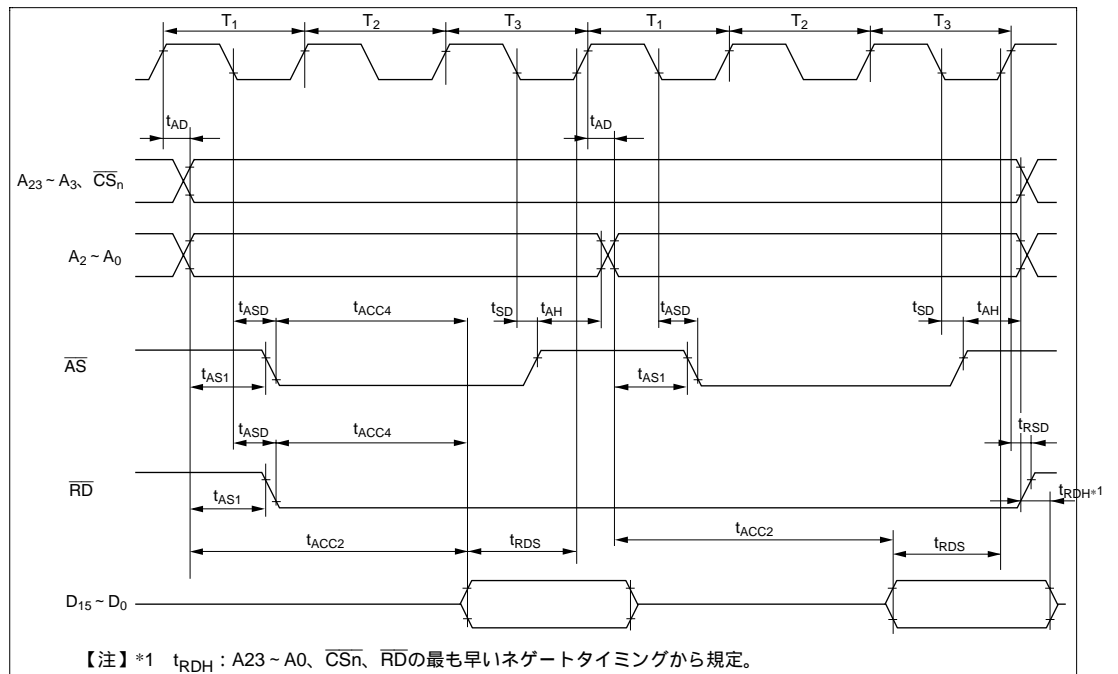


図 21.12 バーストROMアクセスタイミング (バーストサイクル3ステート)

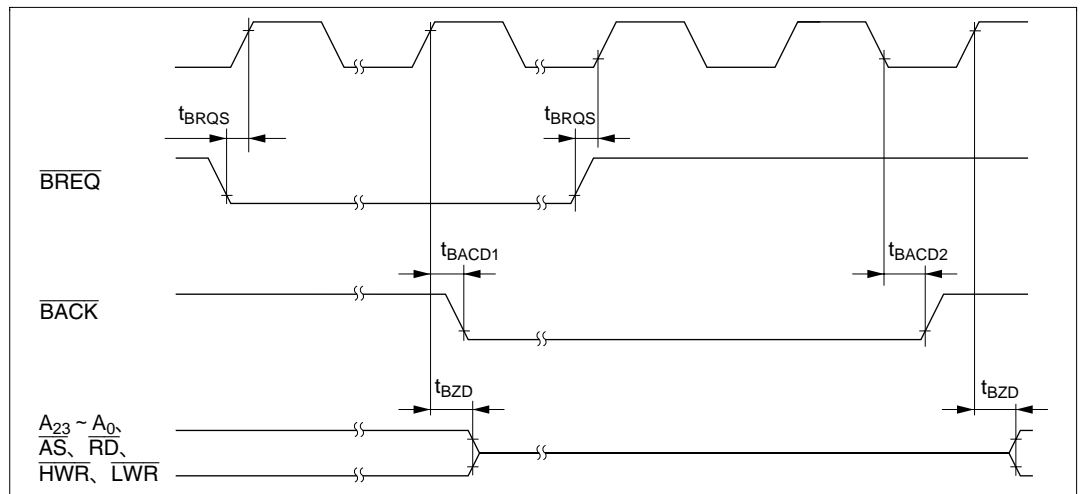


図 21.13 バスリリースモードタイミング

## 21.2.4 DRAM インタフェースバスタイミング

DRAM インタフェースバスタイミングを示します。

### (1) DRAM バスタイミング / リード / ライト時

図 21.14 にリード / ライト時のタイミングを示します。

### (2) DRAM バスタイミング / CAS ビフォア RAS リフレッシュ

図 21.15 に CAS ビフォア RAS リフレッシュ時のタイミングを示します。

### (3) DRAM バスタイミング / セルフリフレッシュ

図 21.16 にセルフリフレッシュ時のタイミングを示します。

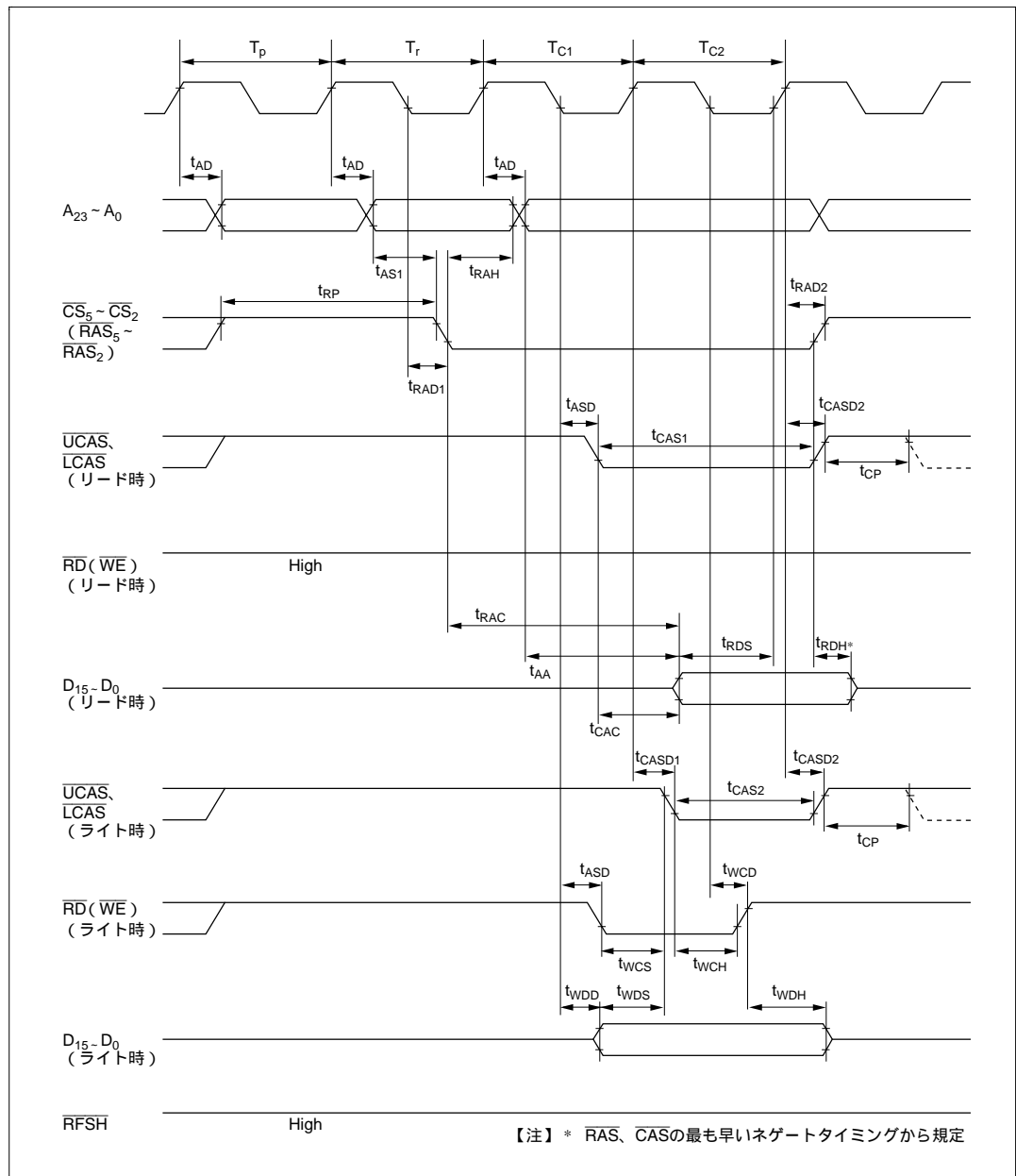


図 21.14 DRAMバスタイミング (リード/ライト時)

21. 電気的特性 (暫定仕様)

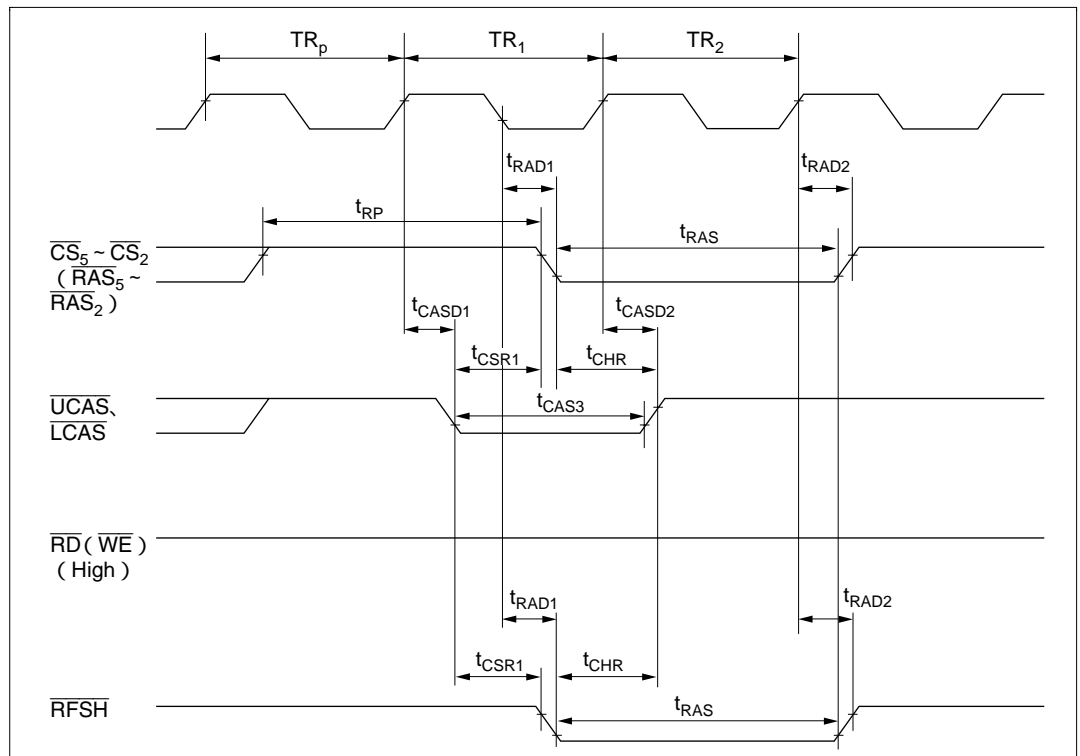


図 21.15 DRAMバスタイミング (CAS ビフォア RAS リフレッシュ)

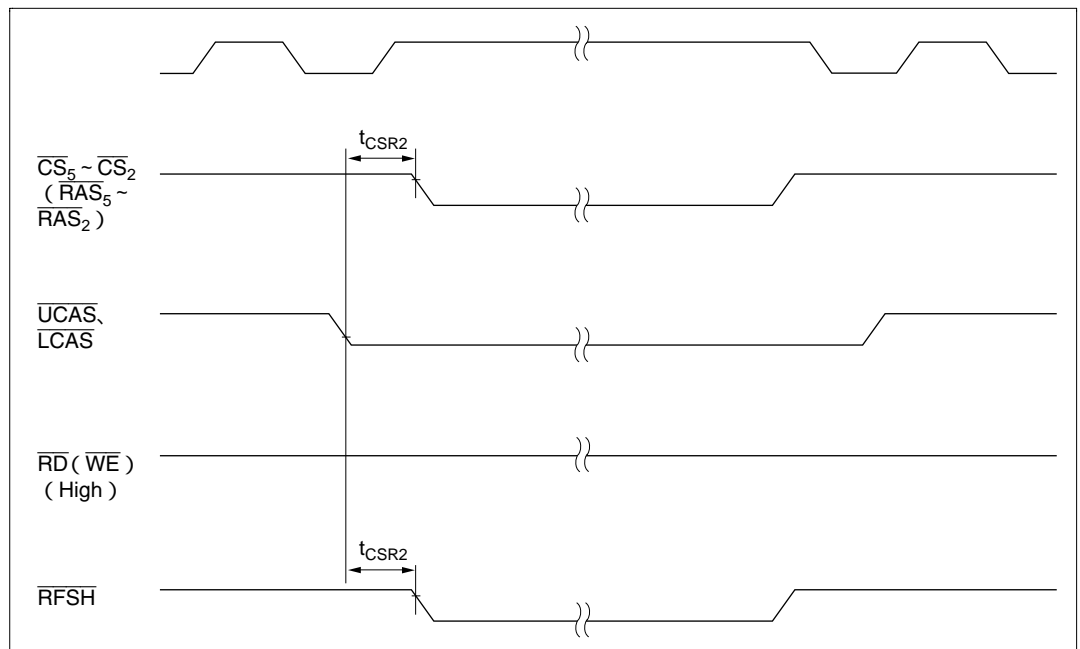


図 21.16 DRAMバスタイミング (セルフリフレッシュ)

## 21.2.5 TPC、I/Oポートタイミング

図 21.17 に TPC、I/Oポートの入出力タイミングを示します。

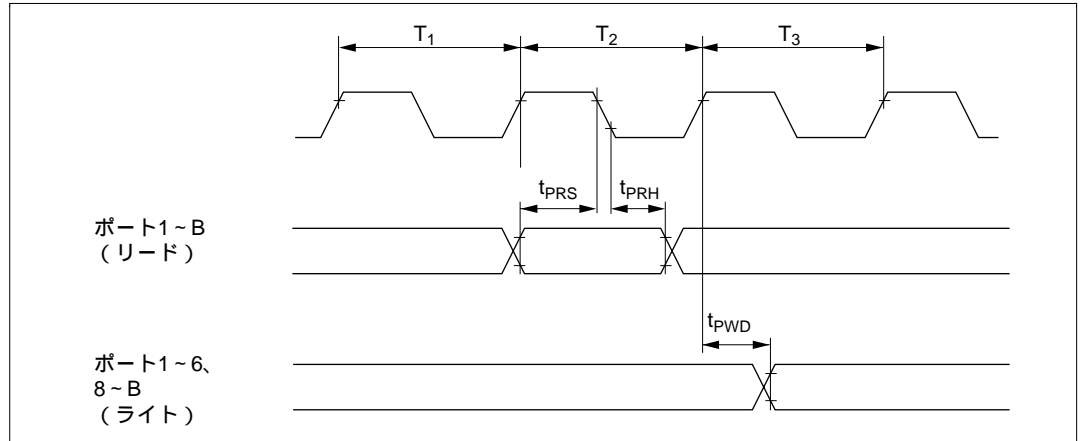


図 21.17 TPC、I/Oポート入出力タイミング

## 21.2.6 タイマ入出力タイミング

16ビットタイマ、8ビットタイマの各タイミングを以下に示します。

### (1) タイマ入出力タイミング

図 21.18 にタイマ入出力タイミングを示します。

### (2) タイマ外部クロック入力タイミング

図 21.19 にタイマ外部クロック入力タイミングを示します。

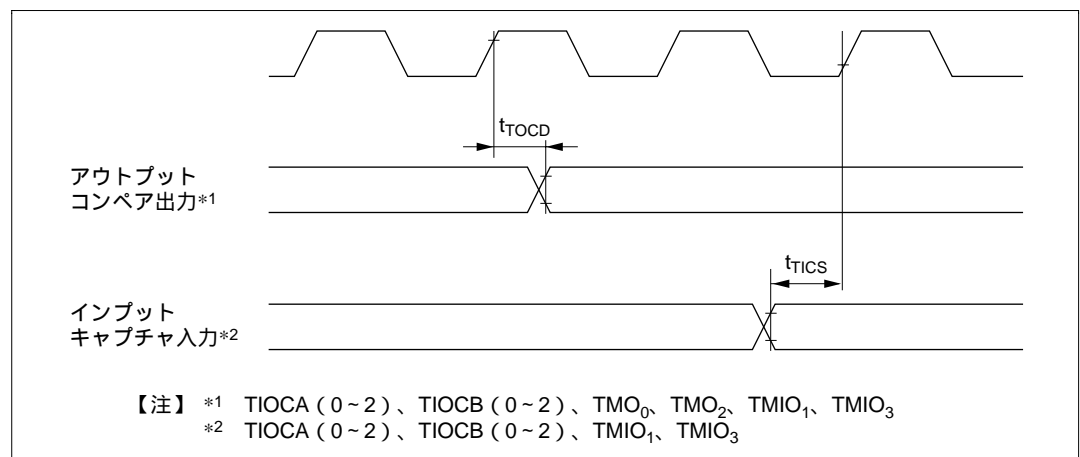


図 21.18 タイマ入出力タイミング

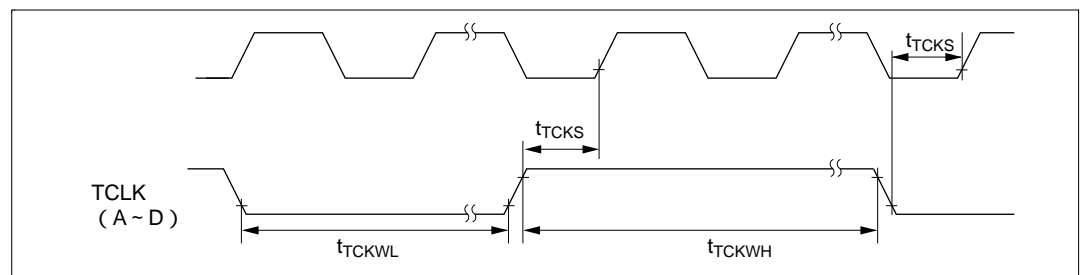


図 21.19 タイマ外部クロック入力タイミング

## 21.2.7 SCI 入出力タイミング

SCI の各タイミングを以下に示します。

### (1) SCI 入力クロックタイミング

図 21.20 に SCI 入力クロックタイミングを示します。

### (2) SCI 入出力タイミング (クロック同期式モード)

図 21.21 にクロック同期式モード時の SCI 入出力タイミングを示します。

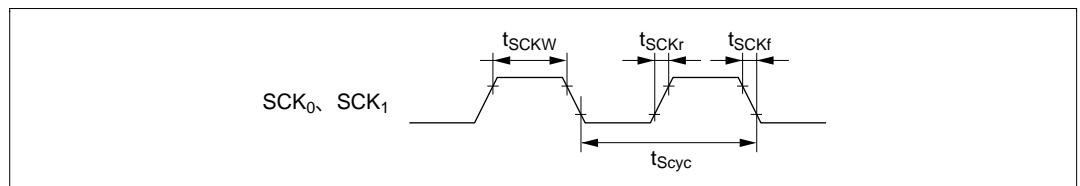


図 21.20 SCI 入力クロックタイミング

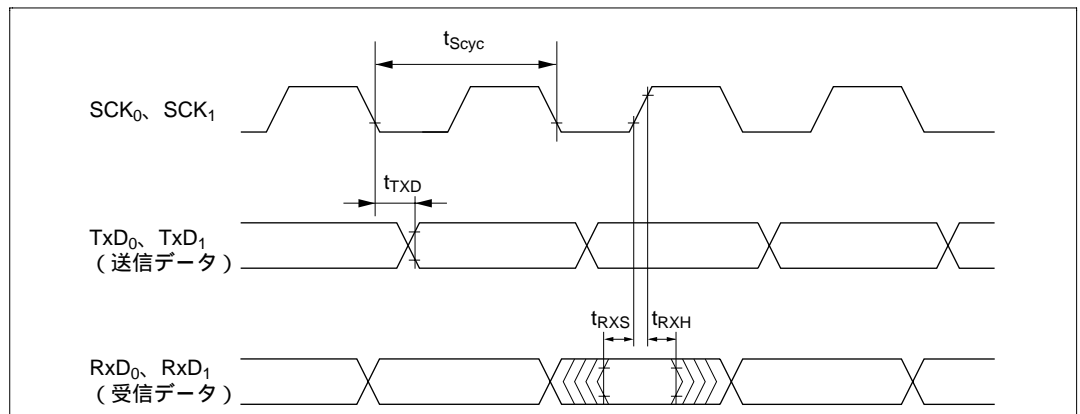


図 21.21 クロック同期式モード時の SCI 入出力タイミング



## 21.2.8 DMAC タイミング

DMAC の各タイミングを以下に示します。

(1) DMAC、 $\overline{\text{TEND}}$  出力タイミング / 2 ステートアクセス

DMAC、 $\overline{\text{TEND}}$  出力タイミング / 2 ステートアクセスを図 21.22 に示します。

(2) DMAC、 $\overline{\text{TEND}}$  出力タイミング / 3 ステートアクセス

DMAC、 $\overline{\text{TEND}}$  出力タイミング / 3 ステートアクセスを図 21.23 に示します。

(3) DMAC、 $\overline{\text{DREQ}}$  入力タイミング

DMAC、 $\overline{\text{DREQ}}$  入力タイミングを図 21.24 に示します。

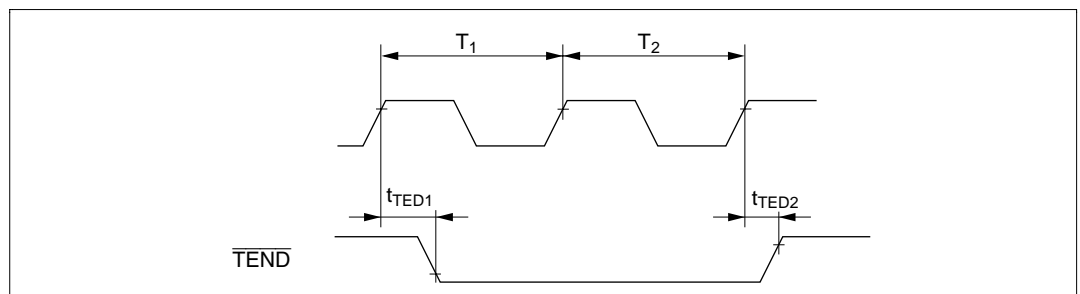


図 21.22 DMAC、 $\overline{\text{TEND}}$  出力タイミング / 2 ステートアクセス

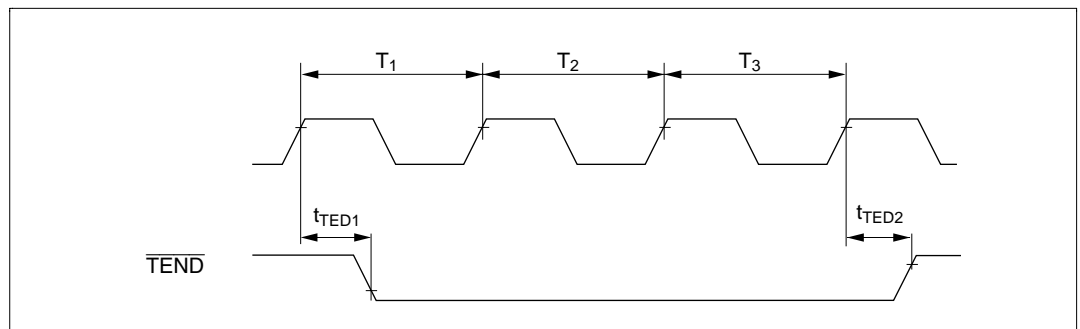


図 21.23 DMAC、 $\overline{\text{TEND}}$  出力タイミング / 3 ステートアクセス

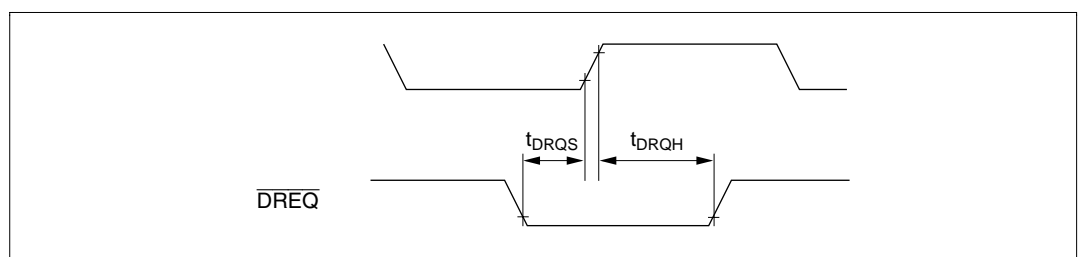


図 21.24 DMAC、 $\overline{\text{DREQ}}$  入力タイミング

---

# 付録

---

## 付録 目次

A.	命令	769	
	A.1	命令一覧	769
	A.2	オペレーションコードマップ	784
	A.3	命令実行ステート数	787
B.	内部 I/O レジスタ一覧	798	
	B.1	アドレス一覧 (EMC ビット=0 のとき)	798
	B.2	アドレス一覧 (EMC ビット=1 のとき)	809
	B.3	機能一覧	834
C.	I/O ポートブロック図	926	
	C.1	ポート 1 ブロック図	926
	C.2	ポート 2 ブロック図	927
	C.3	ポート 3 ブロック図	928
	C.4	ポート 4 ブロック図	929
	C.5	ポート 5 ブロック図	930
	C.6	ポート 6 ブロック図	931
	C.7	ポート 7 ブロック図	938
	C.8	ポート 8 ブロック図	939
	C.9	ポート 9 ブロック図	944
	C.10	ポート A ブロック図	950
	C.11	ポート B ブロック図	953
D.	端子状態	961	
	D.1	各処理状態におけるポートの状態	961
	D.2	リセット時の端子状態	968
E.	ハードウェアスタンバイモード遷移 / 復帰時のタイミングについて	972	
F.	型名一覧	973	
G.	外形寸法図	974	
H.	H8/300H シリーズ製品仕様比較	976	

H.1	H8/3068F と H8/3067、H8/3062 シリーズ、H8/3048 シリーズの相異点.....	976
H.2	100 ピンパッケージ品の端子機能比較 (FP-100B、TFP-100B の場合).....	978

## A. 命令

### A.1 命令一覧

#### 《オペレーションの記号》

記号	内容
Rd	デスティネーション側の汎用レジスタ
Rs	ソース側の汎用レジスタ
Rn	汎用レジスタ
ERd	デスティネーション側の汎用レジスタ (アドレスレジスタまたは 32 ビットレジスタ)
ERs	ソース側の汎用レジスタ (アドレスレジスタまたは 32 ビットレジスタ)
ERn	汎用レジスタ (32 ビットレジスタ)
(EAd)	デスティネーションオペランド
(EAs)	ソースオペランド
PC	プログラムカウンタ
SP	スタックポインタ
CCR	コンディションコードレジスタ
N	CCR の N (ネガティブ) フラグ
Z	CCR の Z (ゼロ) フラグ
V	CCR の V (オーバフロー) フラグ
C	CCR の C (キャリ) フラグ
disp	ディスプレースメント
	左辺のオペランドから右辺のオペランドへの転送、または左辺の状態から右辺の状態への遷移
+	両辺のオペランドを加算
-	左辺のオペランドから右辺のオペランドを減算
×	両辺のオペランドを乗算
÷	左辺のオペランドを右辺のオペランドで除算
	両辺のオペランドの論理積
	両辺のオペランドの論理和
⊕	両辺のオペランドの排他的論理和
~	反転論理 (論理的補数)
( ) < >	オペランドの内容

【注】 汎用レジスタは、8 ビット (R0H~R7H、R0L~R7L) または 16 ビット (R0~R7、E0~E7) です。

## 《コンディションコードの記号》

記号	内容
↑	実行結果にしたがって変化することを表します。
*	不確定であることを表します（値を保証しません）。
0	常に0にクリアされることを表します。
1	常に1にセットされることを表します。
-	実行結果に影響を受けないことを表します。
	条件によって異なります。注意事項を参照してください。

表 A.1 命令セット一覧

(1) データ転送命令

ニーモニック	サイズ	アドレッシングモード/命令長(バイト)				オペレーション	コンディションコード							実行ワード数 <sup>*1</sup>			
		#xx	Rn	@ERn	@(d, ERn)		@-ERn@ERn+	@aa	@(d, PC)	@aa	I	H	N	Z	V	C	ワード
MOV	B	2										↑	↑	0		2	
	B	2										↑	↑	0		2	
	B		2									↑	↑	0		4	
	B			4								↑	↑	0		6	
	B			8								↑	↑	0		10	
	B				2							↑	↑	0		6	
	B					2						↑	↑	0		4	
	B						4					↑	↑	0		6	
	B							6				↑	↑	0		8	
	B		2									↑	↑	0		4	
	B			4								↑	↑	0		6	
	B			8								↑	↑	0		10	
	B				2							↑	↑	0		6	
	B					4						↑	↑	0		8	
	B						6					↑	↑	0		8	
	W	4										↑	↑	0		4	
	W		2									↑	↑	0		2	
	W			2								↑	↑	0		4	
	W				4							↑	↑	0		6	
	W				8							↑	↑	0		10	
	W					2						↑	↑	0		6	
	W						4					↑	↑	0		6	
	W							6				↑	↑	0		8	

ニーモニック	サイズ	アドレッシングモード / 命令長 (バイト)						オペレーション	コンディションコード							実行バイト数*1				
		#xx	Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa		@(d, PC)	@@aa	I	H	N	Z	V		C			
MOV	MOV.W Rs, @ERd	W		2												↑ ↑	0		4	
	MOV.W Rs, @(d:16, ERd)	W			4											↑ ↑	0		6	
	MOV.W Rs, @(d:24, ERd)	W			8											↑ ↑	0		10	
	MOV.W Rs, @-ERd	W				2										↑ ↑	0		6	
	MOV.W Rs, @aa:16	W					4									↑ ↑	0		6	
	MOV.W Rs, @aa:24	W					6									↑ ↑	0		8	
	MOV.L #xx:32, Rd	L	6													↑ ↑	0		6	
	MOV.L ERs, ERd	L	2													↑ ↑	0		2	
	MOV.L @ERs, ERd	L		4												↑ ↑	0		8	
	MOV.L @(d:16, ERs), ERd	L				6										↑ ↑	0		10	
	MOV.L @(d:24, ERs), ERd	L				10										↑ ↑	0		14	
	MOV.L @ERs+, ERd	L					4									↑ ↑	0		10	
	MOV.L @aa:16, ERd	L						6								↑ ↑	0		10	
	MOV.L @aa:24, ERd	L							8							↑ ↑	0		12	
	MOV.L ERs, @ERd	L		4												↑ ↑	0		8	
POP	MOV.L ERs, @(d:16, ERd)	L			6											↑ ↑	0		10	
	MOV.L ERs, @(d:24, ERd)	L			10											↑ ↑	0		14	
	MOV.L ERs, @-ERd	L				4										↑ ↑	0		10	
	MOV.L ERs, @aa:16	L					6									↑ ↑	0		10	
	MOV.L ERs, @aa:24	L						8								↑ ↑	0		12	
	POP.W Rn	W										2	@SP	Rn16, SP+2	SP	↑ ↑	0		6	
	POP.L ERn	L										4	@SP	ERn32, SP+4	SP	↑ ↑	0		10	
	PUSH.W Rn	W											2	SP-2	SP, Rn16	@SP	↑ ↑	0	6	
	PUSH.L ERn	L											4	SP-4	SP, ERn32	@SP	↑ ↑	0	10	
	MOVFPE @aa:16, Rd	B											4							本LSIでは使用できません
	MOVTPPE @aa:16	B											4							本LSIでは使用できません

(2) 算術演算命令

二一モニック	サイズ	アドレッシングモード / 命令長 (バイト)				オペレーション	コンディションコード								実行回数 / フロント				
		#xx	Rn	@ERn	@(d, ERn) @-ERn/@ERn+ @aa @(d, PC)		I	H	N	Z	V	C							
ADD	ADD.B #xx:8, Rd	B	2												↑	↑	↑	↑	2
	ADD.B Rs, Rd	B	2												↑	↑	↑	↑	2
	ADD.W #xx:16, Rd	W	4												(1)	↑	↑	↑	4
	ADD.W Rs, Rd	W	2												(1)	↑	↑	↑	2
	ADD.L #xx:32, ERd	L	6												(2)	↑	↑	↑	6
	ADD.L ERs, ERd	L	2												(2)	↑	↑	↑	2
ADDX	ADDX.B #xx:8, Rd	B	2												↑	↑	(3)	↑	2
	ADDX.B Rs, Rd	B	2												↑	↑	(3)	↑	2
	ADDX.L #1, ERd	L	2																2
ADDS	ADDS.L #2, ERd	L	2																2
	ADDS.L #4, ERd	L	2																2
	INC.B Rd	B	2												↑	↑	↑	↑	2
	INC.W #1, Rd	W	2												↑	↑	↑	↑	2
INC.L	INC.W #2, Rd	W	2												↑	↑	↑	↑	2
	INC.L #1, ERd	L	2												↑	↑	↑	↑	2
	INC.L #2, ERd	L	2												↑	↑	↑	↑	2
	DAA Rd	B	2												*	↑	↑	*	2
	SUB.B Rs, Rd	B	2												↑	↑	↑	↑	2
SUB	SUB.W #xx:16, Rd	W	4												(1)	↑	↑	↑	4
	SUB.W Rs, Rd	W	2												(1)	↑	↑	↑	2
	SUB.L #xx:32, ERd	L	6												(2)	↑	↑	↑	6
	SUB.L ERs, ERd	L	2												(2)	↑	↑	↑	2
	SUBX.B #xx:8:Rd	B	2												↑	↑	(3)	↑	2
	SUBX.B Rs, Rd	B	2												↑	↑	(3)	↑	2



オペレーション	アドレッシングモード / 命令長 (バイト)				オペレーション		コンディションコード							実行バイト数	
	サイズ	Rn	@(d, ERn)	@-ERn@ERn+	@aa	@(d, PC)	@@aa	I	H	N	Z	V	C	ノール	アドバンス
SUBS	SUBS.L #1, ERd	2													2
	SUBS.L #2, ERd	2													2
	SUBS.L #4, ERd	2													2
DEC	DEC.B Rd	2								↑	↑	↑			2
	DEC.W #1, Rd	2								↑	↑	↑			2
	DEC.W #2, Rd	2								↑	↑	↑			2
	DEC.L #1, ERd	2								↑	↑	↑			2
DAS	DAS.L #2, ERd	2								↑	↑	↑			2
	DAS Rd	2							*	↑	↑	*			2
MULXU	MULXU.B Rs, Rd	2													14
	MULXU.W Rs, ERd	2													22
MULXS	MULXS.B Rs, Rd	4								↑	↑				16
	MULXS.W Rs, ERd	4								↑	↑				24
DIVXU	DIVXU.B Rs, Rd	2								(6)	(7)				14
	DIVXU.W Rs, ERd	2								(6)	(7)				22
DIVXS	DIVXS.B Rs, Rd	4								(8)	(7)				16
	DIVXS.W Rs, ERd	4								(8)	(7)				24
CMP	CMP.B #xx:8, Rd	2								↑	↑	↑	↑		2
	CMP.B Rs, Rd	2								↑	↑	↑	↑		2
	CMP.W #xx:16, Rd	4							(1)	↑	↑	↑	↑		4
	CMP.W Rs, Rd	2							(1)	↑	↑	↑	↑		2

二モニック	サイズ	アドレッシングモード/命令長(バイト)						オペレーション	コンディションコード					実行回数*1			
		#xx	Rn	@ERn	@(d,ERn)	@-ERn/@ERn+	@aa		@(d,PC)	@@aa	I	H	N	Z	V	C	ノール
CMP	CMP.L #xx:32, ERd	L	6						ERd32-#xx:32	(2)↑	↑	↑	↑	↑	↑	6	
	CMP.L ERs, ERd	L	2						ERd32-ERs32	(2)↑	↑	↑	↑	↑	↑	2	
NEG	NEG.B Rd	B	2						0-Rd8 Rd8	↑	↑	↑	↑	↑	↑	2	
	NEG.W Rd	W	2						0-Rd16 Rd16	↑	↑	↑	↑	↑	↑	2	
	NEG.L ERd	L	2						0-ERd32 ERd32	↑	↑	↑	↑	↑	↑	2	
XTU	XTU.W Rd	W	2						0 ((L' ヽ15~8) of Rd16)	0	↑	↑	0	0	0	2	
	XTU.L ERd	L	2						0 ((L' ヽ131~16) of ERd32)	0	↑	↑	0	0	0	2	
EXTS	EXTS.W Rd	W	2						((L' ヽ17) of Rd16) ((L' ヽ15~8) of Rd16)	↑	↑	↑	↑	↑	↑	2	
	EXTS.L ERd	L	2						((L' ヽ15) of ERd32) ((L' ヽ131~16) of ERd32)	↑	↑	↑	↑	↑	↑	2	

(3) 論理演算命令

ニーモニック	サイズ	アドレッシングモード / 命令長 (バイト)						オペレーション	コンディションコード							実行回数 <sup>*1</sup>		
		#xx	Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa		@(d, PC)	@@aa	I	H	N	Z	V		C	
AND	AND.B #xx:8, Rd	B	2												↑↑	0		2
	AND.B Rs, Rd	B	2												↑↑	0		2
	AND.W #xx:16, Rd	W	4												↑↑	0		4
	AND.W Rs, Rd	W	2												↑↑	0		2
	AND.L #xx:32, ERd	L	6												↑↑	0		6
	AND.L ERs, ERd	L	4												↑↑	0		4
OR	OR.B #xx:8, Rd	B	2												↑↑	0		2
	OR.B Rs, Rd	B	2												↑↑	0		2
	OR.W #xx:16, Rd	W	4												↑↑	0		4
	OR.W Rs, Rd	W	2												↑↑	0		2
	OR.L #xx:32, ERd	L	6												↑↑	0		6
	OR.L ERs, ERd	L	4												↑↑	0		4
XOR	XOR.B #xx:8, Rd	B	2												↑↑	0		2
	XOR.B Rs, Rd	B	2												↑↑	0		2
	XOR.W #xx:16, Rd	W	4												↑↑	0		4
	XOR.W Rs, Rd	W	2												↑↑	0		2
	XOR.L #xx:32, ERd	L	6												↑↑	0		6
	XOR.L ERs, ERd	L	4												↑↑	0		4
NOT	NOT.B Rd	B	2												↑↑	0		2
	NOT.W Rd	W	2												↑↑	0		2
	NOT.L ERd	L	2												↑↑	0		2

(4) シフト命令

ニーモニック	サイズ	アドレッシングモード/命令長 (バイト)				オペレーション	コンディションコード							実行スタート数 <sup>*1</sup>			
		#xx	Rn	@ERn	@(d, ERn)		@-ERn/@ERn+	@aa	@(d, PC)	@@aa	I	H	N		Z	V	C
SHAL	B	2										↕	↕	↕	↕	↕	2
	W	2										↕	↕	↕	↕	↕	2
	L	2										↕	↕	↕	↕	↕	2
SHAR	B	2										↕	↕	0	↕	↕	2
	W	2										↕	↕	0	↕	↕	2
	L	2										↕	↕	0	↕	↕	2
SHLL	B	2										↕	↕	0	↕	↕	2
	W	2										↕	↕	0	↕	↕	2
	L	2										↕	↕	0	↕	↕	2
SHLR	B	2										↕	↕	0	↕	↕	2
	W	2										↕	↕	0	↕	↕	2
	L	2										↕	↕	0	↕	↕	2
ROTXL	B	2										↕	↕	0	↕	↕	2
	W	2										↕	↕	0	↕	↕	2
	L	2										↕	↕	0	↕	↕	2
ROTXR	B	2										↕	↕	0	↕	↕	2
	W	2										↕	↕	0	↕	↕	2
	L	2										↕	↕	0	↕	↕	2
ROTL	B	2										↕	↕	0	↕	↕	2
	W	2										↕	↕	0	↕	↕	2
	L	2										↕	↕	0	↕	↕	2
ROTR	B	2										↕	↕	0	↕	↕	2
	W	2										↕	↕	0	↕	↕	2
	L	2										↕	↕	0	↕	↕	2

(5) ビット操作命令

二モック	サイズ	アドレッシングモード / 命令長 (バイト)						オペレーション	コンディションコード					実行回数 7ビット		
		Rn #xx	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa	@(d, PC)		@@aa	I	H	N	Z		V	C
BSET	BSET #xx:3, Rd	B	2						(#xx:3 of Rd8) 1							2
	BSET #xx:3, @ERd	B		4					(#xx:3 of @ERd) 1							8
	BSET #xx:3, @aa:8	B			4				(#xx:3 of @aa:8) 1							8
	BSET Rn, Rd	B	2						(Rn8 of Rd8) 1							2
	BSET Rn, @ERd	B		4					(Rn8 of @ERd) 1							8
	BSET Rn, @aa:8	B				4			(Rn8 of @aa:8) 1							8
BCLR	BCLR #xx:3, Rd	B	2						(#xx:3 of Rd8) 0							2
	BCLR #xx:3, @ERd	B		4					(#xx:3 of @ERd) 0							8
	BCLR #xx:3, @aa:8	B			4				(#xx:3 of @aa:8) 0							8
	BCLR Rn, Rd	B	2						(Rn8 of Rd8) 0							2
	BCLR Rn, @ERd	B		4					(Rn8 of @ERd) 0							8
	BCLR Rn, @aa:8	B				4			(Rn8 of @aa:8) 0							8
BNOT	BNOT #xx:3, Rd	B	2						(#xx:3 of Rd8) ~ (#xx:3 of Rd8)							2
	BNOT #xx:3, @ERd	B		4					(#xx:3 of @ERd) ~ (#xx:3 of @ERd)							8
	BNOT #xx:3, @aa:8	B			4				(#xx:3 of @aa:8) ~ (#xx:3 of @aa:8)							8
	BNOT Rn, Rd	B	2						(Rn8 of Rd8) ~ (Rn8 of Rd8)							2
	BNOT Rn, @ERd	B		4					(Rn8 of @ERd) ~ (Rn8 of @ERd)							8
	BNOT Rn, @aa:8	B				4			(Rn8 of @aa:8) ~ (Rn8 of @aa:8)							8
BTST	BTST #xx:3, Rd	B	2						~ (#xx:3 of Rd8) Z				↑			2
	BTST #xx:3, @ERd	B		4					~ (#xx:3 of @ERd) Z				↑			6
	BTST #xx:3, @aa:8	B			4				~ (#xx:3 of @aa:8) Z				↑			6
	BTST Rn, Rd	B	2						~ (Rn8 of @Rd8) Z				↑			2
	BTST Rn, @ERd	B		4					~ (Rn8 of @ERd) Z				↑			6
	BTST Rn, @aa:8	B				4			~ (Rn8 of @aa:8) Z				↑			6
BLD	BLD #xx:3, Rd	B	2						(#xx:3 of Rd8) C					↑		2
	BLD #xx:3, @ERd	B		4					(#xx:3 of @ERd) C					↑		6
	BLD #xx:3, @aa:8	B			4				(#xx:3 of @aa:8) C					↑		6
	BLD Rn, Rd	B	2						~ (#xx:3 of Rd8) C					↑		2
	BLD Rn, @ERd	B		4					~ (#xx:3 of @ERd) C					↑		6
	BLD Rn, @aa:8	B				4			~ (#xx:3 of @aa:8) C					↑		6
BILD	BILD #xx:3, Rd	B	2													2
	BILD #xx:3, @ERd	B		4												6
	BILD #xx:3, @aa:8	B			4											6

二ーモニツク	サイズ	アドレッシングモード/命令長(バイト)						オペレーション	コンディションコード					実行バイト数*1			
		#xx	Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa		@(d, PC)	@aa	I	H	N		Z	V	C
BST	BST #xx:3, Rd	B	2														2
	BST #xx:3, @ERd	B		4													8
	BST #xx:3, @aa:8	B					4										8
BIST	BIST #xx:3, Rd	B	2														2
	BIST #xx:3, @ERd	B		4													8
	BIST #xx:3, @aa:8	B					4										8
BAND	BAND #xx:3, Rd	B	2														2
	BAND #xx:3, @ERd	B		4													6
	BAND #xx:3, @aa:8	B					4										6
BIAND	BIAND #xx:3, Rd	B	2														2
	BIAND #xx:3, @ERd	B		4													6
	BIAND #xx:3, @aa:8	B					4										6
BOR	BOR #xx:3, Rd	B	2														2
	BOR #xx:3, @ERd	B		4													6
	BOR #xx:3, @aa:8	B					4										6
BIOR	BIOR #xx:3, Rd	B	2														2
	BIOR #xx:3, @ERd	B		4													6
	BIOR #xx:3, @aa:8	B					4										6
BXOR	BXOR #xx:3, Rd	B	2														2
	BXOR #xx:3, @ERd	B		4													6
	BXOR #xx:3, @aa:8	B					4										6
BIXOR	BIXOR #xx:3, Rd	B	2														2
	BIXOR #xx:3, @ERd	B		4													6
	BIXOR #xx:3, @aa:8	B					4										6

(6) 分岐命令

ニーモニック	サイズ	アドレッシングモード / 命令長 (バイト)					オペレーション	分岐条件	コンディションコード					実行回数				
		#xx	Rn	@ERn	@(d, ERn)	@-ERn@ERn+			@aa	@(d, PC)	@@aa	I	H	N	Z	V	C	ノーワ
Bcc																		
BRA d:8(BT d:8)																		4
BRA d:16(BT d:16)																		6
BRN d:8(BF d:8)																		4
BRN d:16(BF d:16)																		6
BHI d:8																		4
BHI d:16																		6
BLS d:8																		4
BLS d:16																		6
BCC d:8(BHS d:8)																		4
BCC d:16(BHS d:16)																		6
BCS d:8(BLO d:8)																		4
BCS d:16(BLO d:16)																		6
BNE d:8																		4
BNE d:16																		6
BEQ d:8																		4
BEQ d:16																		6
BVC d:8																		4
BVC d:16																		6
BVS d:8																		4
BVS d:16																		6
BPL d:8																		4
BPL d:16																		6
BMI d:8																		4
BMI d:16																		6

ニーモニック	サイズ	アドレッシングモード / 命令長 (バイト)						オペレーション	分岐条件	コンディションコード					実行バイト数 *1		
		#xx	Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa			@(d, PC)	@@aa	I	H	N	Z	V	C
Bcc	BGE d:8																4
	BGE d:16																6
	BLT d:8																4
	BLT d:16																6
	BGT d:8																4
	BGT d:16																6
JMP	BLE d:8																4
	BLE d:16																6
	JMP @ERn		2														4
	JMP @aa:24					4											6
	JMP @aa:8							2									8
	JMP @aa:8								2								10
BSR	BSR d:8																6
	BSR d:16																8
JSR	JSR @ERn																6
	JSR @aa:24																8
	JSR @aa:8					4											10
RTS	RTS																8
	RTS																10



(7) システム制御命令

ニーモニック	サイズ	アドレッシングモード / 命令長 (バイト)				オペレーション	コンディションコード							実行回数					
		#xx	Rn	@ERn	@(d, ERn)		@-ERn@ERn+	@aa	@(d, PC)	@@aa	I	H	N	Z	V	C	14	16	
TRAPA	TRAPA #x:2																		
RTE	RTE																		
SLEEP	SLEEP																		
LDC	LDC #xx:8, CCR	B	2																
	LDC Rs, CCR	B	2																
	LDC @ERs, CCR	W		4															
	LDC @(d:16, ERs), CCR	W			6														
	LDC @(d:24, ERs), CCR	W			10														
	LDC @ERst, CCR	W			4														
	LDC @aa:16, CCR	W				6													
	LDC @aa:24, CCR	W				8													
	STC	STC CCR, Rd	B	2															
		STC CCR, @ERd	W		4														
	STC CCR, @(d:16, ERd)	W			6														
	STC CCR, @(d:24, ERd)	W			10														
	STC CCR, @-ERd	W			4														
	STC CCR, @aa:16	W				6													
	STC CCR, @aa:24	W				8													
ANDC	ANDC #xx:8, CCR	B	2																
ORC	ORC #xx:8, CCR	B	2																
XORC	XORC #xx:8, CCR	B	2																
NOP	NOP																		

(8) データ転送命令

ニーモニック	サイズ	アドレッシングモード/命令長(バイト)				オペレーション	コンディションコード				実行スタート数*1				
		#xx	@ERn	@(d, ERn)	@-ERn/ERn+		@aa	@(d, PC)	@@aa	I		H	N	Z	V
EEPMOV	EEPMOV.B						4	if R4L 0 Repeat @R5 @R6 R5+1 R5 R6+1 R6 R4L-1 R4L Until R4L=0 else next;							8+4n <sup>*2</sup>
	EEPMOV.W						4	if R4 0 Repeat @R5 @R6 R5+1 R5 R6+1 R6 R4-1 R4 Until R4L=0 else next;							8+4n <sup>*2</sup>

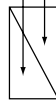
- 【注】** \*1 実行ステータ数は、オペコードおよびオペランドが内蔵メモリに存在する場合です。それ以外の場合は、「A.3 命令実行ステータ数」を参照してください。
- \*2 nはR4LまたはR4の設定値です。
- (1) ビット11から桁上がりまたはビット11へ桁下がりが発生したとき1にセットされ、それ以外のとき0にクリアされます。
  - (2) ビット27から桁上がりまたはビット27へ桁下がりが発生したとき1にセットされ、それ以外のとき0にクリアされます。
  - (3) 演算結果がゼロのとき、演算前の値を保持し、それ以外のとき0にクリアされます。
  - (4) 補正結果に桁上がりが発生したとき、1にセットされ、それ以外のとき演算前の値を保持します。
  - (5) エクロック同期転送命令の実行ステータ数は一定ではありません。
  - (6) 除数が負のとき1にセットされ、それ以外のとき0にクリアされます。
  - (7) 除数がゼロのとき1にセットされ、それ以外のとき0にクリアされます。
  - (8) 商が負のとき1にセットされ、それ以外のとき0にクリアされます。

## A.2 オペレーションコードマップ

表A.2 オペレーションコードマップ

命令コード：  

第1バイト	第2バイト
AH AL BH BL	



→ BHの最上位ビットが0の場合を示します。  
 ← BHの最上位ビットが1の場合を示します。

命令コード	第1バイト	第2バイト	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	NOP	表A.2(2)	STC	LDC	ORC	XORC	ANDC	LDC	ADD	表A.2(2)	表A.2(2)	MOV	ADDX	表A.2(2)		
1	表A.2(2)	表A.2(2)	表A.2(2)	OR.B	XOR.B	AND.B	表A.2(2)	SUB		表A.2(2)	表A.2(2)	CMP	SUBX	表A.2(2)		
2	MOV.B															
3	MOV															
4	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BNQ	BVC	BVS	BPL	BMI	BGE	BLT	BGT	BLE
5	MULXU	DIVXU	MULXU	DIVXU	RTS	BSR	RTE	TRAPA	表A.2(2)		JMP		BSR		JSR	
6	BSET	BNOT	BCLR	BTST	OR	XOR	AND	BST	BAND	BOR	BAND	BAND	MOV			
7					BOR	BXOR	BAND	BLD	BAND	BOR	BAND	MOV	表A.2(2)	表A.2(2)	表A.2(3)	
8	ADD															
9	ADDX															
A	CMP															
B	SUBX															
C	OR															
D	XOR															
E	AND															
F	MOV															

命令コード：  

第1バイト		第2バイト	
AH	AL	BH	BL

BH/AH	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
01	MOV			LDC/STC				SLEEP					表A.2 (3)	表A.2 (3)		表A.2 (3)
0A	INC	ADD														
0B	ADDS					INC		INC	ADDS					INC		INC
0F	DAA	MOV														
10	SHLL			SHLL					SHAL					SHAL		
11	SHLR			SHLR					SHAR					SHAR		
12	ROTXL			ROTXL					ROTL					ROTL		
13	ROTXR			ROTXR					ROTR					ROTR		
17	NOT			NOT				EXTU	NEG					NEG		EXTS
1A	DEC	SUB														
1B	SUBS					DEC		DEC	SUBS					DEC		DEC
1F	DAS	CMP														
58	BRA	BRN		BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE	BLT	BGT	BLE
79	MOV	ADD		SUB	OR	XOR	AND									
7A	MOV	ADD		SUB	OR	XOR	AND									



### A.3 命令実行ステート数

H8/300H CPU の各命令についての実行状態と実行ステート数の計算方法を示します。

表 A.4 に各命令の実行状態として、命令実行中に行われる命令フェッチ、データリード/ライト等のサイクル数を示し、表 A.3 に各々のサイズに必要なステート数を示します。命令実行ステート数は次の計算式で計算されます。

$$\text{実行ステート数} = I \cdot S_I + J \cdot S_J + K \cdot S_K + L \cdot S_L + M \cdot S_M + N \cdot S_N$$

実行ステート数計算例

(例) アドバンストモード、スタック領域を外部空間に設定、内部周辺モジュールアクセス時 8 ビットバス幅、外部デバイスアクセス時 16 ビットバス幅で 3 ステートアクセス 1 ウェイト挿入とした場合

1. BSET #0, @FFFFC7:8

表 A.4 より

$$I = L = 2, J = K = M = N = 0$$

表 A.3 より

$$S_I = 4, S_L = 3$$

$$\text{実行ステート数} = 2 \times 4 + 2 \times 3 = 14$$

2. JSR @@30

表 A.4 より

$$I = J = K = 2, L = M = N = 0$$

表 A.3 より

$$S_I = S_J = S_K = 4$$

$$\text{実行ステート数} = 2 \times 4 + 2 \times 4 + 2 \times 4 = 24$$

表 A.3 実行状態（サイクル）に要するステート数

実行状態 (サイクル)	ア ク セ ス 対 象						
	内 蔵 メモリ	内蔵周辺モジュール		外部デバイス			
		8ビットバス	16ビットバス	8ビットバス		16ビットバス	
				2ステート アクセス	3ステート アクセス	2ステート アクセス	3ステート アクセス
命令フェッチ S <sub>I</sub>	2	6	3	4	6+2m	2	3+m
分岐アドレスリード S <sub>J</sub>							
スタック操作 S <sub>K</sub>							
バイトデータアクセス S <sub>L</sub>							
ワードデータアクセス S <sub>M</sub>							
内部動作 S <sub>N</sub>	1						

《記号説明》

m : 外部デバイスアクセス時のウェイトステート数

表 A.4 命令実行状態 (サイクル数)

命令	ニーモニック	命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作
		ッチ	スリード	操作	ータアク	ータアク	
		I	J	K	L	M	N
ADD	ADD.B #xx:8, Rd	1					
	ADD.B Rs, Rd	1					
	ADD.W #xx:16, Rd	2					
	ADD.W Rs, Rd	1					
	ADD.L #xx:32, ERd	3					
	ADD.L ERs, ERd	1					
ADDS	ADDS #1/2/4, ERd	1					
ADDX	ADDX #xx:8, Rd	1					
	ADDX Rs, Rd	1					
AND	AND.B #xx:8, Rd	1					
	AND.B Rs, Rd	1					
	AND.W #xx:16, Rd	2					
	AND.W Rs, Rd	1					
	AND.L #xx:32, ERd	3					
	AND.L ERs, ERd	2					
ANDC	ANDC #xx:8, CCR	1					
BAND	BAND #xx:3, Rd	1					
	BAND #xx:3, @ERd	2			1		
	BAND #xx:3, @aa:8	2			1		
Bcc	BRA d:8 (BT d:8)	2					
	BRN d:8 (BF d:8)	2					
	BHI d:8	2					
	BLS d:8	2					
	BCC d:8 (BHS d:8)	2					
	BCS d:8 (BLO d:8)	2					
	BNE d:8	2					
	BEQ d:8	2					
	BVC d:8	2					
	BVS d:8	2					
	BPL d:8	2					
	BMI d:8	2					
	BGE d:8	2					
	BLT d:8	2					
	BGT d:8	2					



命令	ニーモニック	命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作
		ッチ	スリード	操作	ータアク	ータアク	
		I	J	K	L	M	N
Bcc	BLE d:8	2					
	BRA d:16 ( BT d:16 )	2					2
	BRN d:16 ( BF d:16 )	2					2
	BHI d:16	2					2
	BLS d:16	2					2
	BCC d:16 ( BHS d:16 )	2					2
	BCS d:16 ( BLO d:16 )	2					2
	BNE d:16	2					2
	BEQ d:16	2					2
	BVC d:16	2					2
	BVS d:16	2					2
	BPL d:16	2					2
	BMI d:16	2					2
	BGE d:16	2					2
	BLT d:16	2					2
	BGT d:16	2					2
	BLE d:16	2					2
BCLR	BCLR #xx:3, Rd	1					
	BCLR #xx:3, @ERd	2			2		
	BCLR #xx:3, @aa:8	2			2		
	BCLR Rn, Rd	1					
	BCLR Rn, @ERd	2			2		
	BCLR Rn, @aa:8	2			2		
BIAND	BIAND #xx:3, Rd	1					
	BIAND #xx:3, @ERd	2			1		
	BIAND #xx:3, @aa:8	2			1		
BILD	BILD #xx:3, Rd	1					
	BILD #xx:3, @ERd	2			1		
	BILD #xx:3, @aa:8	2			1		
BIOR	BIOR #xx:8, Rd	1					
	BIOR #xx:8, @ERd	2			1		
	BIOR #xx:8, @aa:8	2			1		
BIST	BIST #xx:3, Rd	1					
	BIST #xx:3, @ERd	2			2		
	BIST #xx:3, @aa:8	2			2		

命令	ニーモニック		命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作
			ッチ	スリード	操作	ータアク	ータアク	
			I	J	K	L	M	N
BIXOR	BIXOR #xx:3, Rd		1					
	BIXOR #xx:3, @ERd		2			1		
	BIXOR #xx:3, @aa:8		2			1		
BLD	BLD #xx:3, Rd		1					
	BLD #xx:3, @ERd		2			1		
	BLD #xx:3, @aa:8		2			1		
BNOT	BNOT #xx:3, Rd		1					
	BNOT #xx:3, @ERd		2			2		
	BNOT #xx:3, @aa:8		2			2		
	BNOT Rn, Rd		1					
	BNOT Rn, @ERd		2			2		
	BNOT Rn, @aa:8		2			2		
BOR	BOR #xx:3, Rd		1					
	BOR #xx:3, @ERd		2			1		
	BOR #xx:3, @aa:8		2			1		
BSET	BSET #xx:3, Rd		1					
	BSET #xx:3, @ERd		2			2		
	BSET #xx:3, @aa:8		2			2		
	BSET Rn, Rd		1					
	BSET Rn, @ERd		2			2		
	BSET Rn, @aa:8		2			2		
BSR	BSR d:8	ノーマル	2		1			
		アドバンスト	2		2			
	BSR d:16	ノーマル	2		1			2
		アドバンスト	2		2			2
BST	BST #xx:3, Rd		1					
	BST #xx:3, @ERd		2			2		
	BST #xx:3, @aa:8		2			2		
BTST	BTST #xx:3, Rd		1					
	BTST #xx:3, @ERd		2			1		
	BTST #xx:3, @aa:8		2			1		
	BTST Rn, Rd		1					
	BTST Rn, @ERd		2			1		
	BTST Rn, @aa:8		2			1		

命令	ニーモニック	命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作	
		ッチ	スリード	操作	ータアク	ータアク		
		I	J	K	L	M	N	
BXOR	BXOR #xx:3, Rd	1						
	BXOR #xx:3, @ERd	2			1			
	BXOR #xx:3, @aa:8	2			1			
CMP	CMP.B #xx:8, Rd	1						
	CMP.B Rs, Rd	1						
	CMP.W #xx:16, Rd	2						
	CMP.W Rs, Rd	1						
	CMP.L #xx:32, ERd	3						
	CMP.L ERs, ERd	1						
DAA	DAA Rd	1						
DAS	DAS Rd	1						
DEC	DEC.B Rd	1						
	DEC.W #1/2, Rd	1						
	DEC.L #1/2, ERd	1						
DIVXS	DIVXS.B Rs, Rd	2					12	
	DIVXS.W Rs, ERd	2					20	
DIVXU	DIVXU.B Rs, Rd	1					12	
	DIVXU.W Rs, ERd	1					20	
EEPMOV	EEPMOV.B	2			$2n+2^{*1}$			
	EEPMOV.W	2			$2n+2^{*1}$			
EXTS	EXTS.W Rd	1						
	EXTS.L ERd	1						
EXTU	EXTU.W Rd	1						
	EXTU.L ERd	1						
INC	INC.B Rd	1						
	INC.W #1/2, Rd	1						
	INC.L #1/2, ERd	1						
JMP	JMP @ERn	2						
	JMP @aa:24	2					2	
	JMP @@aa:8	ノーマル	2	1				2
		アドバンスト	2	2				2

命令	ニーモニック		命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作
			ッチ	スリード	操作	ータアク	ータアク	
			I	J	K	L	M	N
JSR	JSR @ERn	ノーマル	2		1			
		アドバンスト	2		2			
	JSR @aa:24	ノーマル	2		1			2
		アドバンスト	2		2			2
	JSR @@aa:8	ノーマル	2	1	1			
		アドバンスト	2	2	2			
LDC	LDC #xx:8, CCR		1					
	LDC Rs, CCR		1					
	LDC @ERs, CCR		2				1	
	LDC @(d:16, ERs), CCR		3				1	
	LDC @(d:24, ERs), CCR		5				1	
	LDC @ERs+, CCR		2				1	2
	LDC @aa:16, CCR		3				1	
	LDC @aa:24, CCR		4				1	

命令	ニーモニック	命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作	
		ッチ	スリード	操作	ータアク	ータアク		
		I	J	K	L	M	N	
MOV	MOV.B #xx:8, Rd	1						
	MOV.B Rs, Rd	1						
	MOV.B @ERs, Rd	1			1			
	MOV.B @(d:16, ERs), Rd	2			1			
	MOV.B @(d:24, ERs), Rd	4			1			
	MOV.B @ERs+, Rd	1			1		2	
	MOV.B @aa:8, Rd	1			1			
	MOV.B @aa:16, Rd	2			1			
	MOV.B @aa:24, Rd	3			1			
	MOV.B Rs, @ERd	1			1			
	MOV.B Rs, @(d:16, ERd)	2			1			
	MOV.B Rs, @(d:24, ERd)	4			1			
	MOV.B Rs, @-ERd	1			1		2	
	MOV.B Rs, @aa:8	1			1			
	MOV.B Rs, @aa:16	2			1			
	MOV.B Rs, @aa:24	3			1			
	MOV.W #xx:16, Rd	2						
	MOV.W Rs, Rd	1						
	MOV.W @ERs, Rd	1					1	
	MOV.W @(d:16, ERs), Rd	2					1	
	MOV.W @(d:24, ERs), Rd	4					1	
	MOV.W @ERs+, Rd	1					1	2
	MOV.W @aa:16, Rd	2					1	
	MOV.W @aa:24, Rd	3					1	
	MOV.W Rs, @ERd	1					1	
	MOV.W Rs, @(d:16, ERd)	2					1	
	MOV.W Rs, @(d:24, ERd)	4					1	
	MOV.W Rs, @-ERd	1					1	2
	MOV.W Rs, @aa:16	2					1	
	MOV.W Rs, @aa:24	3					1	
	MOV.L #xx:32, ERd	3						
	MOV.L ERs, ERd	1						
MOV.L @ERs, ERd	2					2		
MOV.L @(d:16, ERs), ERd	3					2		
MOV.L @(d:24, ERs), ERd	5					2		

命令	ニーモニック	命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作
		ッチ	スリード	操作	ータアク	ータアク	
		I	J	K	L	M	N
MOV	MOV.L @ERs+, ERd	2				2	2
	MOV.L @aa:16, ERd	3				2	
	MOV.L @aa:24, ERd	4				2	
	MOV.L ERs, @ERd	2				2	
	MOV.L ERs, @(d:16, ERd)	3				2	
	MOV.L ERs, @(d:24, ERd)	5				2	
	MOV.L ERs, @-ERd	2				2	2
	MOV.L ERs, @aa:16	3				2	
	MOV.L ERs, @aa:24	4				2	
MOVFPPE	MOVFPPE @aa:16, Rd* <sup>2</sup>	2			1		
MOVTPPE	MOVTPPE Rs, @aa:16* <sup>2</sup>	2			1		
MULXS	MULXS.B Rs, Rd	2					12
	MULXS.W Rs, ERd	2					20
MULXU	MULXU.B Rs, Rd	1					12
	MULXU.W Rs, ERd	1					20
NEG	NEG.B Rd	1					
	NEG.W Rd	1					
	NEG.L ERd	1					
NOP	NOP	1					
NOT	NOT.B Rd	1					
	NOT.W Rd	1					
	NOT.L ERd	1					
OR	OR.B #xx:8, Rd	1					
	OR.B Rs, Rd	1					
	OR.W #xx:16, Rd	2					
	OR.W Rs, Rd	1					
	OR.L #xx:32, ERd	3					
	OR.L ERs, ERd	2					
ORC	ORC #xx:8, CCR	1					
POP	POP.W Rn	1				1	2
	POP.L ERn	2				2	2
PUSH	PUSH.W Rn	1				1	2
	PUSH.L ERn	2				2	2

命令	ニーモニック		命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作
			ッチ	スリード	操作	ータアク	ータアク	
			I	J	K	L	M	N
ROTL	ROTL.B Rd		1					
	ROTL.W Rd		1					
	ROTL.L ERd		1					
ROTR	ROTR.B Rd		1					
	ROTR.W Rd		1					
	ROTR.L ERd		1					
ROTXL	ROTXL.B Rd		1					
	ROTXL.W Rd		1					
	ROTXL.L ERd		1					
ROTXR	ROTXR.B Rd		1					
	ROTXR.W Rd		1					
	ROTXR.L ERd		1					
RTE	RTE		2		2			2
RTS	RTS	ノーマル	2		1			2
		アドバンスト	2		2			2
SHAL	SHAL.B Rd		1					
	SHAL.W Rd		1					
	SHAL.L ERd		1					
SHAR	SHAR.B Rd		1					
	SHAR.W Rd		1					
	SHAR.L ERd		1					
SHLL	SHLL.B Rd		1					
	SHLL.W Rd		1					
	SHLL.L ERd		1					
SHLR	SHLR.B Rd		1					
	SHLR.W Rd		1					
	SHLR.L ERd		1					
SLEEP	SLEEP		1					

命令	ニーモニック	命令フェ	分岐アドレ	スタック	バイトデ	ワードデ	内部動作
		ッチ	スリード	操作	ータアク	ータアク	
		I	J	K	L	M	N
STC	STC CCR, Rd	1					2
	STC CCR, @ERd	2				1	
	STC CCR, @(d:16, ERd)	3				1	
	STC CCR, @(d:24, ERd)	5				1	
	STC CCR, @-ERd	2				1	
	STC CCR, @aa:16	3				1	
	STC CCR, @aa:24	4				1	
SUB	SUB.B Rs, Rd	1					
	SUB.W #xx:16, Rd	2					
	SUB.W Rs, Rd	1					
	SUB.L #xx:32, ERd	3					
	SUB.L ERs, ERd	1					
SUBS	SUBS #1/2/4, ERd	1					
SUBX	SUBX #xx:8, Rd	1					
	SUBX Rs, Rd	1					
TRAPA	TRAPA #x:2	ノーマル	2	1	2		4
		アドバンスト	2	2	2		4
XOR	XOR.B #xx:8, Rd	1					
	XOR.B Rs, Rd	1					
	XOR.W #xx:16, Rd	2					
	XOR.W Rs, Rd	1					
	XOR.L #xx:32, ERd	3					
	XOR.L ERs, ERd	2					
XORC	XORC #xx:8, CCR	1					

【注】 \*1 n は R4L、R4 の設定値です。ソース側、デスティネーション側のアクセスが、それぞれ (n+1) 回行われます。

\*2 本 LSI では使用できません。



## B. 内部 I/O レジスタ一覧

### B.1 アドレス一覧 (EMC ビット=1 のとき)

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット 7	ビット 6	ビット 5	ビット 4	ビット 3	ビット 2	ビット 1	ビット 0	
H'EE000	P1DDR	8	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR	ポート 1
H'EE001	P2DDR	8	P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P20DDR	ポート 2
H'EE002	P3DDR	8	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR	ポート 3
H'EE003	P4DDR	8	P47DDR	P46DDR	P45DDR	P44DDR	P43DDR	P42DDR	P41DDR	P40DDR	ポート 4
H'EE004	P5DDR	8					P53DDR	P52DDR	P51DDR	P50DDR	ポート 5
H'EE005	P6DDR	8		P66DDR	P65DDR	P64DDR	P63DDR	P62DDR	P61DDR	P60DDR	ポート 6
H'EE006											
H'EE007	P8DDR	8				P84DDR	P83DDR	P82DDR	P81DDR	P80DDR	ポート 8
H'EE008	P9DDR	8			P95DDR	P94DDR	P93DDR	P92DDR	P91DDR	P90DDR	ポート 9
H'EE009	PADDR	8	PA7DDR	PA6DDR	PA5DDR	PA4DDR	PA3DDR	PA2DDR	PA1DDR	PA0DDR	ポート A
H'EE00A	PBDDR	8	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR	PB0DDR	ポート B
H'EE00B											
H'EE00C											
H'EE00D											
H'EE00E											
H'EE00F											
H'EE010											
H'EE011	MDCR	8						MDS2	MDS1	MDS0	システム
H'EE012	SYSCR	8	SSBY	STS2	STS1	STS0	UE	NMIEG	SSOE	RAME	制御
H'EE013	BRCR	8	A23E	A22E	A21E	A20E				BRLE	バスコントローラ
H'EE014	ISCR	8			IRQ5SC	IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC	割り込み
H'EE015	IER	8			IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E	コント
H'EE016	ISR	8			IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F	ローラ
H'EE017											
H'EE018	IPRA	8	IPRA7	IPRA6	IPRA5	IPRA4	IPRA3	IPRA2	IPRA1	IPRA0	
H'EE019	IPRB	8	IPRB7	IPRB6	IPRB5		IPRB3	IPRB2	IPRB1		
H'EE01A	DASTCR	8								DASTE	D/A 変換器
H'EE01B	DIVCR	8							DIV1	DIV0	システム
H'EE01C	MSTCRH	8	PSTOP					MSTPH2	MSTPH1	MSTPH0	制御
H'EE01D	MSTCRL	8	MSTPL7		MSTPL5	MSTPL4	MSTPL3	MSTPL2		MSTPL0	
H'EE01E	ADRCR	8								ADRCTL	バスコントローラ
H'EE01F	CSCR	8	CS7E	CS6E	CS5E	CS4E					

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE020	ABWCR	8	ABW7	ABW6	ABW5	ABW4	ABW3	ABW2	ABW1	ABW0	バス
H'EE021	ASTCR	8	AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0	コント
H'EE022	WCRH	8	W71	W70	W61	W60	W51	W50	W41	W40	ローラ
H'EE023	WCRL	8	W31	W30	W21	W20	W11	W10	W01	W00	
H'EE024	BCR	8	ICIS1	ICIS0	BROME	BRSTS1	BRSTS0	EMC	RDEA	WAITE	
H'EE025											
H'EE026	DRCRA	8	DRAS2	DRAS1	DRAS0		BE	RDM	SRFMD	RFSHE	DRAM
H'EE027	DRCRB	8	MXC1	MXC0	CSEL	RCYCE		TPC	RCW	RLW	インタ
H'EE028	RTMCSR	8	CMF	CMIE	CKS2	CKS1	CKS0				フェース
H'EE029	RTCNT	8									
H'EE02A	RTCOR	8									
H'EE02B	リザーブ領域 (アクセス禁止)										
H'EE02C											
H'EE02D											
H'EE02E											
H'EE02F											
H'EE030	FLMCR1	8	FWE	SWE	ESU	PSU	EV	PV	E	P	フラッシュ
H'EE031	FLMCR2	8	FLER								メモリ
H'EE032	EBR1	8	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0	
H'EE033	EBR2	8			EB13	EB12	EB11	EB10	EB9	EB8	
H'EE034	リザーブ領域 (アクセス禁止)										
H'EE035											
H'EE036											
H'EE037											
H'EE038											
H'EE039											
H'EE03A											
H'EE03B											
H'EE03C	P2PCR	8	P27PCR	P26PCR	P25PCR	P24PCR	P23PCR	P22PCR	P21PCR	P20PCR	ポート 2
H'EE03D											
H'EE03E	P4PCR	8	P47PCR	P46PCR	P45PCR	P44PCR	P43PCR	P42PCR	P41PCR	P40PCR	ポート 4
H'EE03F	P5PCR	8					P53PCR	P52PCR	P51PCR	P50PCR	ポート 5

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE040											
H'EE041											
H'EE042											
H'EE043											
H'EE044											
H'EE045											
H'EE046											
H'EE047											
H'EE048											
H'EE049											
H'EE04A											
H'EE04B											
H'EE04C											
H'EE04D											
H'EE04E											
H'EE04F											
H'EE050											
H'EE051											
H'EE052											
H'EE053											
H'EE054											
H'EE055											
H'EE056											
H'EE057											
H'EE058											
H'EE059											
H'EE05A											
H'EE05B											
H'EE05C											
H'EE05D											
H'EE05E											
H'EE05F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名	
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0		
H'EE060												
H'EE061												
H'EE062												
H'EE063												
H'EE064												
H'EE065												
H'EE066												
H'EE067												
H'EE068												
H'EE069												
H'EE06A												
H'EE06B												
H'EE06C												
H'EE06D												
H'EE06E												
H'EE06F												
H'EE070												
H'EE071												
H'EE072												
H'EE073												
H'EE074	リザーブ領域（アクセス禁止）											
H'EE075												
H'EE076												
H'EE077	RAMCR* <sup>1</sup>	8						RAMS	RAM2	RAM1	RAM0	フラッシュ メモリ*
H'EE078	リザーブ領域（アクセス禁止）											
H'EE079												
H'EE07A												
H'EE07B												
H'EE07C												
H'EE07D												
H'EE07E												
H'EE07F												
H'EE080												
H'EE081												

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF20	MAR0AR	8									DMAC チャンネル 0A
H'FFF21	MAR0AE	8									
H'FFF22	MAR0AH	8									
H'FFF23	MAR0AL	8									
H'FFF24	ETCR0AH	8									
H'FFF25	ETCR0AL	8									
H'FFF26	IOAR0A	8									
H'FFF27	DTCR0A	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTE	DTSZ	SAID	SAIDE	DTIE	DTS2A	DTS1A	DTS0A	フルアドレ スモード
H'FFF28	MAR0BR	8									DMAC チャンネル 0B
H'FFF29	MAR0BE	8									
H'FFF2A	MAR0BH	8									
H'FFF2B	MAR0BL	8									
H'FFF2C	ETCR0BH	8									
H'FFF2D	ETCR0BL	8									
H'FFF2E	IOAR0B	8									
H'FFF2F	DTCR0B	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTME		DAID	DAIDE	TMS	DTS2B	DTS1B	DTS0B	フルアドレ スモード
H'FFF30	MAR1AR	8									DMAC チャンネル 1A
H'FFF31	MAR1AE	8									
H'FFF32	MAR1AH	8									
H'FFF33	MAR1AL	8									
H'FFF34	ETCR1AH	8									
H'FFF35	ETCR1AL	8									
H'FFF36	IOAR1A	8									
H'FFF37	DTCR1A	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTE	DTSZ	SAID	SAIDE	DTIE	DTS2A	DTS1A	DTS0A	フルアドレ スモード
H'FFF38	MAR1BR	8									DMAC チャンネル 1B
H'FFF39	MAR1BE	8									
H'FFF3A	MAR1BH	8									
H'FFF3B	MAR1BL	8									
H'FFF3C	ETCR1BH	8									
H'FFF3D	ETCR1BL	8									
H'FFF3E	IOAR1B	8									
H'FFF3F	DTCR1B	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTME		DAID	DAIDE	TMS	DTS2B	DTS1B	DTS0B	フルアドレ スモード

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF40											
H'FFF41											
H'FFF42											
H'FFF43											
H'FFF44											
H'FFF45											
H'FFF46											
H'FFF47											
H'FFF48											
H'FFF49											
H'FFF4A											
H'FFF4B											
H'FFF4											
H'FFF4											
H'FFF4E											
H'FFF4F											
H'FFF50											
H'FFF51											
H'FFF52											
H'FFF53											
H'FFF54											
H'FFF55											
H'FFF56											
H'FFF57											
H'FFF58											
H'FFF59											
H'FFF5A											
H'FFF5B											
H'FFF5											
H'FFF5											
H'FFF5E											
H'FFF5F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名	
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0		
H'FFF60	TSTR	8							STR2	STR1	STR0	16ビットタイマ 共通
H'FFF61	TSNC	8							SYNC2	SYNC1	SYNC0	
H'FFF62	TMDR	8		MDF	FDIR				PWM2	PWM1	PWM0	
H'FFF63	TOLR	8			TOB2	TOA2	TOB1	TOA1	TOB0	TOA0		
H'FFF64	TISRA	8		IMIEA2	IMIEA1	IMIEA0			IMFA2	IMFA1	IMFA0	
H'FFF65	TISRB	8		IMIEB2	IMIEB1	IMIEB0			IMFB2	IMFB1	IMFB0	
H'FFF66	TISRC	8		OVIE2	OVIE1	OVIE0			OVF2	OVF1	OVF0	
H'FFF67												
H'FFF68	TCR0	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	16ビットタイマ チャンネル0	
H'FFF69	TIOR0	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0		
H'FFF6A	TCNT0H	16										
H'FFF6B	TCNT0L											
H'FFF6C	GRA0H	16										
H'FFF6D	GRA0L											
H'FFF6E	GRB0H	16										
H'FFF6F	GRB0L											
H'FFF70	16TCR1	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	16ビットタイマ チャンネル1	
H'FFF71	TIOR1	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0		
H'FFF72	16TCNT1	16										
H'FFF73	16TCNT1L											
H'FFF74	GRA1H	16										
H'FFF75	GRA1L											
H'FFF76	GRB1H	16										
H'FFF77	GRB1L											
H'FFF78	16TCR2	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	16ビットタイマ チャンネル2	
H'FFF79	TIOR2	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0		
H'FFF7A	16TCNT2	16										
H'FFF7B	16TCNT2L											
H'FFF7C	GRA2H	16										
H'FFF7D	GRA2L											
H'FFF7E	GRB2H	16										
H'FFF7F	GRB2L											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF80	8TCR0	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	8ビット タイマ チャンネル 0/1
H'FFF81	8TCR1	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	
H'FFF82	8TCSR0	8	CMFB	CMFA	OVF	ADTE	OIS3	OIS2	OS1	OS0	
H'FFF83	8TCSR1	8	CMFB	CMFA	OVF	ICE	OIS3	OIS2	OS1	OS0	
H'FFF84	TCORA0	8									
H'FFF85	TCORA1	8									
H'FFF86	TCORB0	8									
H'FFF87	TCORB1	8									
H'FFF88	8TCNT0	8									
H'FFF89	8TCNT1	8									
H'FFF8A											
H'FFF8B											
H'FFF8C	TCSR*2	8	OVF	WT/IT	TME			CKS2	CKS1	CKS0	
H'FFF8D	TCNT*2	8									
H'FFF8E											
H'FFF8F	RSTCSR*2	8	WRST	RSTOE							
H'FFF90	8TCR2	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	8ビット タイマ チャンネル 2/3
H'FFF91	8TCR3	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	
H'FFF92	8TCSR2	8	CMFB	CMFA	OVF		OIS3	OIS2	OS1	OS0	
H'FFF93	8TCSR3	8	CMFB	CMFA	OVF	ICE	OIS3	OIS2	OS1	OS0	
H'FFF94	TCORA2	8									
H'FFF95	TCORA3	8									
H'FFF96	TCORB2	8									
H'FFF97	TCORB3	8									
H'FFF98	8TCNT2	8									
H'FFF99	8TCNT3	8									
H'FFF9A											
H'FFF9B											
H'FFF9C	DADR0	8									
H'FFF9D	DADR1	8									
H'FFF9E	DACR	8	DAOE1	DAOE0	DAE						
H'FFF9F		8									



下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFA0	TPMR	8					G3NOV	G2NOV	G1NOV	G0NOV	TPC
H'FFFA1	TPCR	8	G3CMS1	G3CMS0	G2CMS1	G2CMS0	G1CMS1	G1CMS0	G0CMS1	G0CMS0	
H'FFFA2	NDERB	8	NDER15	NDER14	NDER13	NDER12	NDER11	NDER10	NDER9	NDER8	
H'FFFA3	NDERA	8	NDER7	NDER6	NDER5	NDER4	NDER3	NDER2	NDER1	NDER0	
H'FFFA4	NDRB* <sup>3</sup>	8	NDR15	NDR14	NDR13	NDR12	NDR11	NDR10	NDR9	NDR8	
			NDR15	NDR14	NDR13	NDR12					
H'FFFA5	NDRA* <sup>3</sup>	8	NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0	
			NDR7	NDR6	NDR5	NDR4					
H'FFFA6	NDRB* <sup>3</sup>	8					NDR11	NDR10	NDR9	NDR8	
							NDR3	NDR2	NDR1	NDR0	
H'FFFA7	NDRA* <sup>3</sup>	8									
H'FFFA8											
H'FFFA9											
H'FFFAA											
H'FFFAB											
H'FFFAC											
H'FFFAD											
H'FFFAE											
H'FFFAF											
H'FFFB0	SMR	8	C/ $\bar{A}$	CHR	PE	O/ $\bar{E}$	STOP	MP	CKS1	CKS0	SCI チャンネル0
H'FFFB1	BRR	8									
H'FFFB2	SCR	8	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'FFFB3	TDR	8									
H'FFFB4	SSR	8	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT	
H'FFFB5	RDR	8									
H'FFFB6	SCMR	8					SDIR	SINV		SMIF	
H'FFFB7	リザーブ領域 (アクセス禁止)										
H'FFFB8	SMR	8	C/ $\bar{A}$	CHR	PE	O/ $\bar{E}$	STOP	MP	CKS1	CKS0	SCI チャンネル1
H'FFFB9	BRR	8									
H'FFBFA	SCR	8	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'FFBFB	TDR	8									
H'FFBFC	SSR	8	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT	
H'FFBFD	RDR	8									
H'FFBFE	SCMR	8					SDIR	SINV		SMIF	
H'FFBF	リザーブ領域 (アクセス禁止)										

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFC0	SMR	8	C/ $\bar{A}$	CHR	PE	O/ $\bar{E}$	STOP	MP	CKS1	CKS0	SCI チャンネル2
H'FFFC1	BRR	8									
H'FFFC2	SCR	8	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'FFFC3	TDR	8									
H'FFFC4	SSR	8	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT	
H'FFFC5	RDR	8									
H'FFFC6	SCMR	8					SDIR	SINV		SMIF	
H'FFFC7	リザーブ領域 (アクセス禁止)										
H'FFFC8											
H'FFFC9											
H'FFFC A											
H'FFFC B											
H'FFFC C											
H'FFFC D											
H'FFFC E											
H'FFFC F											
H'FFFD0	P1DR	8	P17	P16	P15	P14	P13	P12	P11	P10	ポート 1
H'FFFD1	P2DR	8	P27	P26	P25	P24	P23	P22	P21	P20	ポート 2
H'FFFD2	P3DR	8	P37	P36	P35	P34	P33	P32	P31	P30	ポート 3
H'FFFD3	P4DR	8	P47	P46	P45	P44	P43	P42	P41	P40	ポート 4
H'FFFD4	P5DR	8					P53	P52	P51	P50	ポート 5
H'FFFD5	P6DR	8	P67	P66	P65	P64	P63	P62	P61	P60	ポート 6
H'FFFD6	P7DR	8	P77	P76	P75	P74	P73	P72	P71	P70	ポート 7
H'FFFD7	P8DR	8				P84	P83	P82	P81	P80	ポート 8
H'FFFD8	P9DR	8			P95	P94	P93	P92	P91	P90	ポート 9
H'FFFD9	PADR	8	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	ポート A
H'FFFDA	PBDR	8	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	ポート B
H'FFFDB											
H'FFFDC											
H'FFFDD											
H'FFFDE											
H'FFPDF											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFE0	ADDRAH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D 変換器
H'FFFE1	ADDRAL	8	AD1	AD0							
H'FFFE2	ADDRBH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFFE3	ADDRBL	8	AD1	AD0							
H'FFFE4	ADDRCH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFFE5	ADDRCL	8	AD1	AD0							
H'FFFE6	ADDRDH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFFE7	ADDRDL	8	AD1	AD0							
H'FFFE8	ADCSR	8	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0	
H'FFFE9	ADCR	8	TRGE								

【注】 \*1 RAMCR、FLMSR はフラッシュメモリ版専用レジスタです。

マスク ROM 版には存在しません。

\*2 TCSR, TCNT, RSTCSR のライトについては「12.2.4 レジスタ書き換え時の注意」を参照してください。

\*3 出力トリガの設定によりアドレスが変化します。

《記号説明》

WDT: ウォッチドッグタイマ

TPC: プログラマブルタイミングパターンコントローラ

SCI: シリアルコミュニケーションインタフェース

## B.2 アドレス一覧 (EMC ビット=0 のとき)

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE000	P1DDR	8	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR	ポート 1
H'EE001	P2DDR	8	P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P20DDR	ポート 2
H'EE002	P3DDR	8	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR	ポート 3
H'EE003	P4DDR	8	P47DDR	P46DDR	P45DDR	P44DDR	P43DDR	P42DDR	P41DDR	P40DDR	ポート 4
H'EE004	P5DDR	8					P53DDR	P52DDR	P51DDR	P50DDR	ポート 5
H'EE005	P6DDR	8		P66DDR	P65DDR	P64DDR	P63DDR	P62DDR	P61DDR	P60DDR	ポート 6
H'EE006		8									
H'EE007	P8DDR	8				P84DDR	P83DDR	P82DDR	P81DDR	P80DDR	ポート 8
H'EE008	P9DDR	8			P95DDR	P94DDR	P93DDR	P92DDR	P91DDR	P90DDR	ポート 9
H'EE009	PADDR	8	PA7DDR	PA6DDR	PA5DDR	PA4DDR	PA3DDR	PA2DDR	PA1DDR	PA0DDR	ポート A
H'EE00A	PBDDR	8	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR	PB0DDR	ポート B
H'EE00B		8									
H'EE00C		8									
H'EE00D		8									
H'EE00E		8									
H'EE00F		8									
H'EE010		8									
H'EE011	MDCR	8						MDS2	MDS1	MDS0	システム
H'EE012	SYSCR	8	SSBY	STS2	STS1	STS0	UE	NMIEG	SSOE	RAME	制御
H'EE013	BRCR	8	A23E	A22E	A21E	A20E				BRLE	バスコントローラ
H'EE014	ISCR	8			IRQ5SC	IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC	割り込み
H'EE015	IER	8			IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E	コント
H'EE016	ISR	8			IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F	ローラ
H'EE017		8									
H'EE018	IPRA	8	IPRA7	IPRA6	IPRA5	IPRA4	IPRA3	IPRA2	IPRA1	IPRA0	
H'EE019	IPRB	8	IPRB7	IPRB6	IPRB5		IPRB3	IPRB2	IPRB1		
H'EE01A	DASTCR	8								DASTE	D/A 変換器
H'EE01B	DIVCR	8							DIV1	DIV0	システム
H'EE01C	MSTCR(H)	8	PSTOP					MSTPH2	MSTPH1	MSTPH0	制御
H'EE01D	MSTCR(L)	8	MSTPL7		MSTPL5	MSTPL4	MSTPL3	MSTPL2		MSTPL0	
H'EE01E	ADRCR	8								ADRCTL	バスコントローラ
H'EE01F	CSCR	8	CS7E	CS6E	CS5E	CS4E					

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE020	ABWCR	8	ABW7	ABW6	ABW5	ABW4	ABW3	ABW2	ABW1	ABW0	バス コント ローラ
H'EE021	ASTCR	8	AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0	
H'EE022	WCRH	8	W71	W70	W61	W60	W51	W50	W41	W40	
H'EE023	WCRL	8	W31	W30	W21	W20	W11	W10	W01	W00	
H'EE024	BCR	8	ICIS1	ICIS0	BROME	BRSTS1	BRSTS0	EMC	RDEA	WAITE	
H'EE025	(FLWCR)										
H'EE026	DRCRA	8	DRAS2	DRAS1	DRAS0		BE	RDM	SRFMD	RFSHE	DRAM インタ フェース
H'EE027	DRCRB	8	MXC1	MXC0	CSEL	RCYCE		TPC	RCW	RLW	
H'EE028	RTMCSR	8	CMF	CMIE	CKS2	CKS1	CKS0				
H'EE029	RTCNT	8									
H'EE02A	RTCOR	8									
H'EE02B		8									バス コント ローラ
H'EE02C	DCR0	8									
H'EE02D	DCR1	8									
H'EE02E	DCR2	8									
H'EE02F	DCR3	8									
H'EE030	FLMCR1	8	FWE	SWE	ESU	PSU	EV	PV	E	P	フラッシュ メモリ
H'EE031	FLMCR2	8	FLER								
H'EE032	EBR1	8	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0	
H'EE033	EBR2	8			EB13	EB12	EB11	EB10	EB9	EB8	
H'EE034	リザーブ領域（アクセス禁止）										
H'EE035											
H'EE036											
H'EE037											
H'EE03C	P2PCR	8	P27PCR	P26PCR	P25PCR	P24PCR	P23PCR	P22PCR	P21PCR	P20PCR	ポート 2
H'EE03D		8									
H'EE03E	P4PCR	8	P47PCR	P46PCR	P45PCR	P44PCR	P43PCR	P42PCR	P41PCR	P40PCR	ポート 4
H'EE03F	P5PCR	8					P53PCR	P52PCR	P51PCR	P50PCR	ポート 5

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE040											
H'EE041											
H'EE042											
H'EE043											
H'EE044											
H'EE045											
H'EE046											
H'EE047											
H'EE048											
H'EE049											
H'EE04A											
H'EE04B											
H'EE04C											
H'EE04D											
H'EE04E											
H'EE04F											
H'EE050											
H'EE051											
H'EE052											
H'EE053											
H'EE054											
H'EE055											
H'EE056											
H'EE057											
H'EE058											
H'EE059											
H'EE05A											
H'EE05B											
H'EE05C											
H'EE05D											
H'EE05E											
H'EE05F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE060											
H'EE061											
H'EE062											
H'EE063											
H'EE064											
H'EE065											
H'EE066											
H'EE067											
H'EE068											
H'EE069											
H'EE06A											
H'EE06B											
H'EE06C											
H'EE06D											
H'EE06E											
H'EE06F											
H'EE070	リザーブ領域（アクセス禁止）										
H'EE071											
H'EE072											
H'EE073											
H'EE074											
H'EE075											
H'EE076											
H'EE077	RAMCR* <sup>1</sup>	8					RAMS	RAM2	RAM1	RAM0	フラッシュ メモリ*
H'EE078	リザーブ領域（アクセス禁止）										
H'EE079											
H'EE07A											
H'EE07B											
H'EE07C											
H'EE07D											
H'EE07E											
H'EE07F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE090	TCSR	8	OVF	WT/IT	TME			CKS2	CKS1	CKS0	WDT
H'EE091	TCNT	8									
H'EE092											
H'EE093	RSTCSR	8	WRST								
H'EE094	リザーブ領域 (アクセス禁止)										
H'EE095											
H'EE096											
H'EE097											
H'EE098											
H'EE099											
H'EE09A											
H'EE09B											
H'EE09C											
H'EE09D											
H'EE09E											
H'EE09F											
H'EE0A0											
H'EE0A1											
H'EE0A2											
H'EE0A3											
H'EE0A4											
H'EE0A5											
H'EE0A6											
H'EE0A7											
H'EE0A8											
H'EE0A9											
H'EE0AA											
H'EE0AB											
H'EE0AC											
H'EE0AD											
H'EE0AE											
H'EE0AF											



下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE0B0	リザーブ領域（アクセス禁止）										
H'EE0B1											
H'EE0B2											
H'EE0B3											
H'EE0B4											
H'EE0B5											
H'EE0B6											
H'EE0B7											
H'EE0B8											
H'EE0B9											
H'EE0BA											
H'EE0BB											
H'EE0BC											
H'EE0BD											
H'EE0BE											
H'EE0BF											
H'EE0C0											
H'EE0C1											
H'EE0C2											
H'EE0C3											
H'EE0C4											
H'EE0C5											
H'EE0C6											
H'EE0C7											
H'EE0C8											
H'EE0C9											
H'EE0CA											
H'EE0CB											
H'EE0CC											
H'EE0CD											
H'EE0CE											
H'EE0CF											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE0D0	リザーブ領域 (アクセス禁止)										
H'EE0D1											
H'EE0D2											
H'EE0D3											
H'EE0D4											
H'EE0D5											
H'EE0D6											
H'EE0D7											
H'EE0D8											
H'EE0D9											
H'EE0DA											
H'EE0DB											
H'EE0DC											
H'EE0DD											
H'EE0DE											
H'EE0DF											
H'EE0E0											
H'EE0E1											
H'EE0E2											
H'EE0E3											
H'EE0E4											
H'EE0E5											
H'EE0E6											
H'EE0E7											
H'EE0E8											
H'EE0E9											
H'EE0EA											
H'EE0EB											
H'EE0EC											
H'EE0ED											
H'EE0EE											
H'EE0EF											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'EE0F0	リザーブ領域（アクセス禁止）										
H'EE0F1											
H'EE0F2											
H'EE0F4											
H'EE0F3											
H'EE0F5											
H'EE0F6											
H'EE0F7											
H'EE0F8											
H'EE0F9											
H'EE0FA											
H'EE0FB											
H'EE0FC											
H'EE0FD											
H'EE0FE											
H'EE0FF											
H'FFE00											
H'FFE01											
H'FFE02											
H'FFE03											
H'FFE04											
H'FFE05											
H'FFE06											
H'FFE07											
H'FFE08											
H'FFE09											
H'FFE0A											
H'FFE0B											
H'FFE0C											
H'FFE0D											
H'FFE0E											
H'FFE0F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFE10											
H'FFE11											
H'FFE12											
H'FFE13											
H'FFE14											
H'FFE15											
H'FFE16											
H'FFE17											
H'FFE18											
H'FFE19											
H'FFE1A											
H'FFE1B											
H'FFE1C											
H'FFE1D											
H'FFE1E											
H'FFE1F											
H'FFE20											
H'FFE21											
H'FFE22											
H'FFE23											
H'FFE24											
H'FFE25											
H'FFE26											
H'FFE27											
H'FFE28											
H'FFE29											
H'FFE2A											
H'FFE2B											
H'FFE2C											
H'FFE2D											
H'FFE2E											
H'FFE2F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFE30											
H'FFE31											
H'FFE32											
H'FFE33											
H'FFE34											
H'FFE35											
H'FFE36											
H'FFE37											
H'FFE38											
H'FFE39											
H'FFE3A											
H'FFE3B											
H'FFE3C											
H'FFE3D											
H'FFE3E											
H'FFE3F											
H'FFE40											
H'FFE41											
H'FFE42											
H'FFE43											
H'FFE44											
H'FFE45											
H'FFE46											
H'FFE47											
H'FFE48											
H'FFE49											
H'FFE4A											
H'FFE4B											
H'FFE4C											
H'FFE4D											
H'FFE4E											
H'FFE4F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFE50											
H'FFE51											
H'FFE52											
H'FFE53											
H'FFE54											
H'FFE55											
H'FFE56											
H'FFE57											
H'FFE58											
H'FFE59											
H'FFE5A											
H'FFE5B											
H'FFE5C											
H'FFE5D											
H'FFE5E											
H'FFE5F											
H'FFE60											
H'FFE61											
H'FFE62											
H'FFE63											
H'FFE64											
H'FFE65											
H'FFE66											
H'FFE67											
H'FFE68											
H'FFE69											
H'FFE6A											
H'FFE6B											
H'FFE6C											
H'FFE6D											
H'FFE6E											
H'FFE6F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFE70											
H'FFE71											
H'FFE72											
H'FFE73											
H'FFE74											
H'FFE75											
H'FFE76											
H'FFE77											
H'FFE78											
H'FFE79											
H'FFE7A											
H'FFE7B											
H'FFE7C											
H'FFE7D											
H'FFE7E											
H'FFE7F											
H'FFE80	MAR0AR	8									DMAC チャンネル 0A
H'FFE81	MAR0AE	8									
H'FFE82	MAR0AH	8									
H'FFE83	MAR0AL	8									
H'FFE84	ETCR0AH	8									
H'FFE85	ETCR0AL	8									
H'FFE86	IOAR0A	8									
H'FFE87	DTCR0A	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTE	DTSZ	SAID	SAIDE	DTIE	DTS2A	DTS1A	DTS0A	フルアドレス モード
H'FFE88	MAR0BR	8									DMAC チャンネル 0B
H'FFE89	MAR0BE	8									
H'FFE8A	MAR0BH	8									
H'FFE8B	MAR0BL	8									
H'FFE8C	ETCR0BH	8									
H'FFE8D	ETCR0BL	8									
H'FFE8E	IOAR0B	8									
H'FFE8F	DTCR0B	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTME		DAID	DAIDE	TMS	DTS2B	DTS1B	DTS0B	フルアドレス モード

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFE90	MAR1AR	8									DMAC チャンネル 1A
H'FFE91	MAR1AE	8									
H'FFE92	MAR1AH	8									
H'FFE93	MAR1AL	8									
H'FFE94	ETCR1AH	8									
H'FFE95	ETCR1AL	8									
H'FFE96	IOAR1A	8									
H'FFE97	DTCR1A	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTE	DTSZ	SAID	SAIDE	DTIE	DTS2A	DTS1A	DTS0A	フルアドレス モード
H'FFE98	MAR1BR	8									DMAC チャンネル 1B
H'FFE99	MAR1BE	8									
H'FFE9A	MAR1BH	8									
H'FFE9B	MAR1BL	8									
H'FFE9C	ETCR1BH	8									
H'FFE9D	ETCR1BL	8									
H'FFE9E	IOAR1B	8									
H'FFE9F	DTCR1B	8	DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ショートアド レスモード
			DTME		DAID	DAIDE	TMS	DTS2B	DTS1B	DTS0B	フルアドレス モード
H'FFEA0	TSTR	8						STR2	STR1	STR0	16ビットタ イマ
H'FFEA1	TSNC	8						SYNC2	SYNC1	SYNC0	共通
H'FFEA2	TMDR	8		MDF	FDIR			PWM2	PWM1	PWM0	
H'FFEA3	TOLR	8			TOB2	TOA2	TOB1	TOA1	TOB0	TOA0	
H'FFEA4	TISRA	8		IMIEA2	IMIEA1	IMIEA0		IMFA2	IMFA1	IMFA0	
H'FFEA5	TISRB	8		IMIEB2	IMIEB1	IMIEB0		IMFB2	IMFB1	IMFB0	
H'FFEA6	TISRC	8		OVIE2	OVIE1	OVIE0		OVF2	OVF1	OVF0	
H'FFEA7											
H'FFEA8	16TCR0	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	16ビットタ イマ
H'FFEA9	TIOR0	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0	チャンネル0
H'FFEAA	16TCNT0	16									
H'FFEAB	16TCNT0										
H'FFEAC	GRA0H	16									
H'FFEAD	GRA0L										
H'FFEAE	GRB0H	16									
H'FFEAF	GRB0L										



下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFEB0	16TCR1	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	16ビットタ チャンネル1
H'FFEB1	TIOR1	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0	
H'FFEB2	16TCNT1H	16									
H'FFEB3	16TCNT1L										
H'FFEB4	GRA1H	16									
H'FFEB5	GRA1L										
H'FFEB6	GRB1H	16									
H'FFEB7	GRB1L										
H'FFEB8	16TCR2	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	16ビットタ チャンネル2
H'FFEB9	TIOR2	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0	
H'FFEBA	16TCNT2H	16									
H'FFEBB	16TCNT2L										
H'FFEBC	GRA2H	16									
H'FFEBD	GRA2L										
H'FFEBE	GRB2H	16									
H'FFEBF	GRB2L										

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFEC0	8TCR0	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	8ビット タイマ チャンネル 0/1
H'FFEC1	8TCR1	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	
H'FFEC2	8TCSR0	8	CMFB	CMFA	OVF	ADTE	OIS3	OIS2	OS1	OS0	
H'FFEC3	8TCSR1	8	CMFB	CMFA	OVF	ICE	OIS3	OIS2	OS1	OS0	
H'FFEC4	TCORA0	8									
H'FFEC5	TCORA1	8									
H'FFEC6	TCORB0	8									
H'FFEC7	TCORB1	8									
H'FFEC8	8TCNT0	8									
H'FFEC9	8TCNT1	8									
H'FFECA	リザーブ領域 (アクセス禁止)										
H'FFECB											
H'FFECC											
H'FFECD											
H'FFECE											
H'FFECF											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFED0	8TCR2	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	8ビット タイマ チャンネル 2/3
H'FFED1	8TCR3	8	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	
H'FFED2	8TCSR2	8	CMFB	CMFA	OVF		OIS3	OIS2	OS1	OS0	
H'FFED3	8TCSR3	8	CMFB	CMFA	OVF	ICE	OIS3	OIS2	OS1	OS0	
H'FFED4	TCORA2	8									
H'FFED5	TCORA3	8									
H'FFED6	TCORB2	8									
H'FFED7	TCORB3	8									
H'FFED8	8TCNT2	8									
H'FFED9	8TCNT3	8									
H'FFEDA	リザーブ領域 (アクセス禁止)										
H'FFEDB											
H'FFEDC											
H'FFEDD											
H'FFEDE											
H'FFEDF											
H'FFEE0	SMR	8	C/ $\bar{A}$	CHR	PE	O/ $\bar{E}$	STOP	MP	CKS1	CKS0	
H'FFEE1	BRR	8									
H'FFEE2	SCR	8	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'FFEE3	TDR	8									
H'FFEE4	SSR	8	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT	
H'FFEE5	RDR	8									
H'FFEE6	SCMR	8					SDIR	SINV		SMIF	
H'FFEE7											
H'FFEE8	SMR	8	C/ $\bar{A}$	CHR	PE	O/ $\bar{E}$	STOP	MP	CKS1	CKS0	SCI チャンネル1
H'FFEE9	BRR	8									
H'FFEEA	SCR	8	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'FFEEB	TDR	8									
H'FFEEC	SSR	8	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT	
H'FFEED	RDR	8									
H'FFEEE	SCMR	8					SDIR	SINV		SMIF	
H'FFEEF											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFEF0	SMR	8	C/ $\bar{A}$	CHR	PE	O/ $\bar{E}$	STOP	MP	CKS1	CKS0	SCI チャンネル2
H'FFEF1	BRR	8									
H'FFEF2	SCR	8	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
H'FFEF3	TDR	8									
H'FFEF4	SSR	8	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT	
H'FFEF5	RDR	8									
H'FFEF6	SCMR	8					SDIR	SINV		SMIF	
H'FFEF7											
H'FFEF8	TPMR	8					G3NOV	G2NOV	G1NOV	G0NOV	
H'FFEF9	TPCR	8	G3CMS1	G3CMS0	G2CMS1	G2CMS0	G1CMS1	G1CMS0	G0CMS1	G0CMS0	
H'FFEFA	NDERB	8	NDER15	NDER14	NDER13	NDER12	NDER11	NDER10	NDER9	NDER8	
H'FFEFB	NDERA	8	NDER7	NDER6	NDER5	NDER4	NDER3	NDER2	NDER1	NDER0	
H'FFEFC	NDRB* <sup>3</sup>	8	NDR15	NDR14	NDR13	NDR12	NDR11	NDR10	NDR9	NDR8	
			NDR15	NDR14	NDR13	NDR12					
H'FFefd	NDRA* <sup>3</sup>	8	NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0	
			NDR7	NDR6	NDR5	NDR4					
H'FFEFE	NDRB* <sup>3</sup>	8									
							NDR11	NDR10	NDR9	NDR8	
H'FFEFF	NDRA* <sup>3</sup>	8									
							NDR3	NDR2	NDR1	NDR0	

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF10											
H'FFF11											
H'FFF12											
H'FFF13											
H'FFF14											
H'FFF15											
H'FFF16											
H'FFF17											
H'FFF18											
H'FFF19											
H'FFF1A											
H'FFF1B											
H'FFF1C											
H'FFF1D											
H'FFF1E											
H'FFF1F											
H'FFF20											
H'FFF21											
H'FFF22											
H'FFF23											
H'FFF24											
H'FFF25											
H'FFF26											
H'FFF27											
H'FFF28											
H'FFF29											
H'FFF2A											
H'FFF2B											
H'FFF2C											
H'FFF2D											
H'FFF2E											
H'FFF2F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF30											
H'FFF31											
H'FFF32											
H'FFF33											
H'FFF34											
H'FFF35											
H'FFF36											
H'FFF37											
H'FFF38											
H'FFF39											
H'FFF3A											
H'FFF3B											
H'FFF3C											
H'FFF3D											
H'FFF3E											
H'FFF3F											
H'FFF40											
H'FFF41											
H'FFF42											
H'FFF43											
H'FFF44											
H'FFF45											
H'FFF46											
H'FFF47											
H'FFF48											
H'FFF49											
H'FFF4A											
H'FFF4B											
H'FFF4C											
H'FFF4D											
H'FFF4E											
H'FFF4F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF50											
H'FFF51											
H'FFF52											
H'FFF53											
H'FFF54											
H'FFF55											
H'FFF56											
H'FFF57											
H'FFF58											
H'FFF59											
H'FFF5A											
H'FFF5B											
H'FFF5C											
H'FFF5D											
H'FFF5E											
H'FFF5F											
H'FFF60											
H'FFF61											
H'FFF62											
H'FFF63											
H'FFF64											
H'FFF65											
H'FFF66											
H'FFF67											
H'FFF68											
H'FFF69											
H'FFF6A											
H'FFF6B											
H'FFF6C											
H'FFF6D											
H'FFF6E											
H'FFF6F											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF70											
H'FFF71											
H'FFF72											
H'FFF73											
H'FFF74											
H'FFF75											
H'FFF76											
H'FFF77											
H'FFF78											
H'FFF79											
H'FFF7A											
H'FFF7B											
H'FFF7C											
H'FFF7D											
H'FFF7E											
H'FFF7F											
H'FFF80											
H'FFF81											
H'FFF82											
H'FFF83											
H'FFF84											
H'FFF85											
H'FFF86											
H'FFF87											
H'FFF88											
H'FFF89											
H'FFF8A											
H'FFF8B											
H'FFF8C											
H'FFF8D											
H'FFF8E											
H'FFF8F											



下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFF90											
H'FFF91											
H'FFF92											
H'FFF93											
H'FFF94											
H'FFF95											
H'FFF96											
H'FFF97											
H'FFF98											
H'FFF99											
H'FFF9A											
H'FFF9B											
H'FFF9C											
H'FFF9D											
H'FFF9E											
H'FFF9F											
H'FFFA0											
H'FFFA1											
H'FFFA2											
H'FFFA3											
H'FFFA4											
H'FFFA5											
H'FFFA6											
H'FFFA7											
H'FFFA8											
H'FFFA9											
H'FFFAA											
H'FFFAB											
H'FFFAC											
H'FFFAD											
H'FFF AE											
H'FFF AF											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFB0											
H'FFFB1											
H'FFFB2											
H'FFFB3											
H'FFFB4											
H'FFFB5											
H'FFFB6											
H'FFFB7											
H'FFFB8											
H'FFFB9											
H'FFBBA											
H'FFBBB											
H'FFBBC											
H'FFBBD											
H'FFBBE											
H'FFBBF											
H'FFFC0											
H'FFFC1											
H'FFFC2											
H'FFFC3											
H'FFFC4											
H'FFFC5											
H'FFFC6											
H'FFFC7											
H'FFFC8											
H'FFFC9											
H'FFFCA											
H'FFFCB											
H'FFCC											
H'FFCFD											
H'FFCFE											
H'FFCF											

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFD0											
H'FFFD1											
H'FFFD2											
H'FFFD3											
H'FFFD4											
H'FFFD5											
H'FFFD6											
H'FFFD7											
H'FFFD8											
H'FFFD9											
H'FFFDA											
H'FFFDB											
H'FFDC											
H'FFDD											
H'FFDE											
H'FFDF											
H'FFE0	ADDRAH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D 変換器
H'FFE1	ADDRAL	8	AD1	AD0							
H'FFE2	ADDRBH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFE3	ADDRBL	8	AD1	AD0							
H'FFE4	ADDRCH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFE5	ADDRCL	8	AD1	AD0							
H'FFE6	ADDRDH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'FFE7	ADDRDL	8	AD1	AD0							
H'FFE8	ADCSR	8	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0	
H'FFE9	ADCR	8	TRGE								
H'FFFEA	リザーブ領域 (アクセス禁止)										
H'FFFEB											
H'FFFEC	DADR0	8									D/A 変換器
H'FF FED	DADR1	8									
H'FF FEE	DACR	8									
H'FF FEF		8									

下位 アドレス	レジスタ 名	データ バス幅	ビット名								モジュール 名
			ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0	
H'FFFF0	P1DR	8	P17	P16	P15	P14	P13	P12	P11	P10	ポート 1
H'FFFF1	P2DR	8	P27	P26	P25	P24	P23	P22	P21	P20	ポート 2
H'FFFF2	P3DR	8	P37	P36	P35	P34	P33	P32	P31	P30	ポート 3
H'FFFF3	P4DR	8	P47	P46	P45	P44	P43	P42	P41	P40	ポート 4
H'FFFF4	P5DR	8					P53	P52	P51	P50	ポート 5
H'FFFF5	P6DR	8	P67	P66	P65	P64	P63	P62	P61	P60	ポート 6
H'FFFF6	P7DR	8	P77	P76	P75	P74	P73	P72	P71	P70	ポート 7
H'FFFF7	P8DR	8				P84	P83	P82	P81	P80	ポート 8
H'FFFF8	P9DR	8			P95	P94	P93	P92	P91	P90	ポート 9
H'FFFF9	PADR	8	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	ポート A
H'FFFFA	PBDR	8	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	ポート B
H'FFFFB											
H'FFFFC											
H'FFFFD											
H'FFFFE											
H'FFFFF											

【注】 \*1 RAMCR、FLMSRはフラッシュメモリ版およびフラッシュメモリR専用レジスタです。

マスクROM版には存在しません。

\*2 TCSR, TCNT, RSTCSRのライトについては「12.2.4 レジスタ書き換え時の注意」を参照してください。

\*3 出力トリガの設定によりアドレスが変化します。

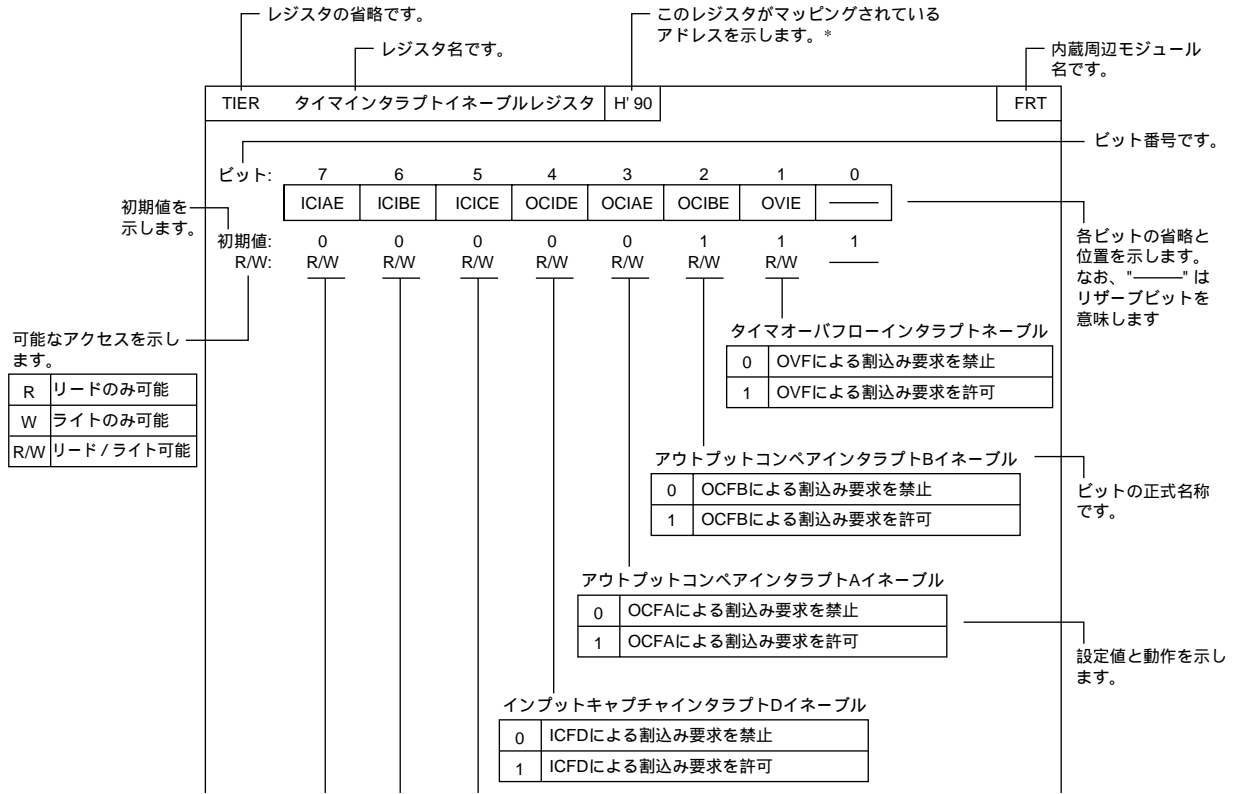
《記号説明》

WDT: ウォッチドッグタイマ

TPC: プログラマブルタイミングパターンコントローラ

SCI: シリアルコミュニケーションインタフェース

### B.3 機能一覧



\* BCR の EMC ビットを 0 にすると変化するレジスタがあります。

P1DDR ポート1 データディレクションレジスタ	H'EE000	ポート1													
<p>ビット:            7            6            5            4            3            2            1            0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%;">P17DDR</td> <td style="width: 10%;">P16DDR</td> <td style="width: 10%;">P15DDR</td> <td style="width: 10%;">P14DDR</td> <td style="width: 10%;">P13DDR</td> <td style="width: 10%;">P12DDR</td> <td style="width: 10%;">P11DDR</td> <td style="width: 10%;">P10DDR</td> </tr> </table> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 15%;"> <p>モード1~4 {</p> <p>初期値:    1            1            1            1            1            1            1            1</p> <p>R/W:        —            —            —            —            —            —            —            —</p> </div> <div style="width: 15%;"> <p>モード5~7 {</p> <p>初期値:    0            0            0            0            0            0            0            0</p> <p>R/W:        W            W            W            W            W            W            W            W</p> </div> </div> <div style="text-align: center; margin: 10px 0;"> <p>↓</p> <p>ポート1入出力選択</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td style="padding-left: 5px;">入力ポート</td> </tr> <tr> <td style="width: 20px; text-align: center;">1</td> <td style="padding-left: 5px;">出力ポート</td> </tr> </table> </div>				P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR	0	入力ポート	1	出力ポート
	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR							
0	入力ポート														
1	出力ポート														
P2DDR ポート2 データディレクションレジスタ	H'EE001	ポート2													
<p>ビット:            7            6            5            4            3            2            1            0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%;">P27DDR</td> <td style="width: 10%;">P26DDR</td> <td style="width: 10%;">P25DDR</td> <td style="width: 10%;">P24DDR</td> <td style="width: 10%;">P23DDR</td> <td style="width: 10%;">P22DDR</td> <td style="width: 10%;">P21DDR</td> <td style="width: 10%;">P20DDR</td> </tr> </table> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="width: 15%;"> <p>モード1~4 {</p> <p>初期値:    1            1            1            1            1            1            1            1</p> <p>R/W:        —            —            —            —            —            —            —            —</p> </div> <div style="width: 15%;"> <p>モード5~7 {</p> <p>初期値:    0            0            0            0            0            0            0            0</p> <p>R/W:        W            W            W            W            W            W            W            W</p> </div> </div> <div style="text-align: center; margin: 10px 0;"> <p>↓</p> <p>ポート2入出力選択</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td style="padding-left: 5px;">入力ポート</td> </tr> <tr> <td style="width: 20px; text-align: center;">1</td> <td style="padding-left: 5px;">出力ポート</td> </tr> </table> </div>				P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P20DDR	0	入力ポート	1	出力ポート
	P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P20DDR							
0	入力ポート														
1	出力ポート														

P3DDR ポート3 データディレクションレジスタ	H'EE002	ポート3																				
<p style="margin-left: 40px;">ビット:</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">7</td> <td style="padding: 2px 10px;">6</td> <td style="padding: 2px 10px;">5</td> <td style="padding: 2px 10px;">4</td> <td style="padding: 2px 10px;">3</td> <td style="padding: 2px 10px;">2</td> <td style="padding: 2px 10px;">1</td> <td style="padding: 2px 10px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px 5px;">P37DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P36DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P35DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P34DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P33DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P32DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P31DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P30DDR</td> </tr> </table> <p style="margin-left: 40px;">初期値:    0      0      0      0      0      0      0      0</p> <p style="margin-left: 40px;">R/W:        W      W      W      W      W      W      W      W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;">ポート3入出力選択</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">0</td> <td style="padding: 2px 5px;">入力ポート</td> </tr> <tr> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">出力ポート</td> </tr> </table> </div>			7	6	5	4	3	2	1	0	P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR	0	入力ポート	1	出力ポート
7	6	5	4	3	2	1	0															
P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR															
0	入力ポート																					
1	出力ポート																					

P4DDR ポート4 データディレクションレジスタ	H'EE003	ポート4																				
<p style="margin-left: 40px;">ビット:</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">7</td> <td style="padding: 2px 10px;">6</td> <td style="padding: 2px 10px;">5</td> <td style="padding: 2px 10px;">4</td> <td style="padding: 2px 10px;">3</td> <td style="padding: 2px 10px;">2</td> <td style="padding: 2px 10px;">1</td> <td style="padding: 2px 10px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px 5px;">P47DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P46DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P45DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P44DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P43DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P42DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P41DDR</td> <td style="border: 1px solid black; padding: 2px 5px;">P40DDR</td> </tr> </table> <p style="margin-left: 40px;">初期値:    0      0      0      0      0      0      0      0</p> <p style="margin-left: 40px;">R/W:        W      W      W      W      W      W      W      W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;">ポート4入出力選択</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">0</td> <td style="padding: 2px 5px;">入力ポート</td> </tr> <tr> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">出力ポート</td> </tr> </table> </div>			7	6	5	4	3	2	1	0	P47DDR	P46DDR	P45DDR	P44DDR	P43DDR	P42DDR	P41DDR	P40DDR	0	入力ポート	1	出力ポート
7	6	5	4	3	2	1	0															
P47DDR	P46DDR	P45DDR	P44DDR	P43DDR	P42DDR	P41DDR	P40DDR															
0	入力ポート																					
1	出力ポート																					

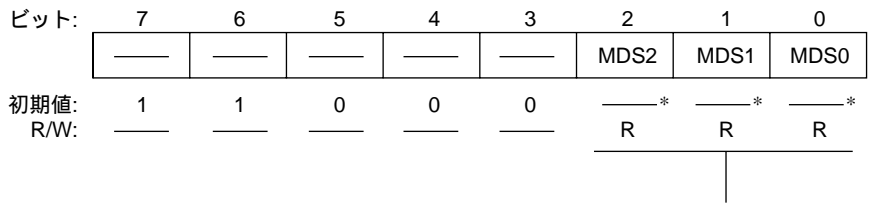
P5DDR ポート5 データディレクションレジスタ	H'EE004	ポート5																																																										
<p>ビット:      7          6          5          4          3          2          1          0</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="border: none;"></td> <td style="border: none; text-align: center;">7</td> <td style="border: none; text-align: center;">6</td> <td style="border: none; text-align: center;">5</td> <td style="border: none; text-align: center;">4</td> <td style="border: none; text-align: center;">3</td> <td style="border: none; text-align: center;">2</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">0</td> </tr> <tr> <td style="border: none;"></td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">P53DDR</td> <td style="border: 1px solid black; text-align: center;">P52DDR</td> <td style="border: 1px solid black; text-align: center;">P51DDR</td> <td style="border: 1px solid black; text-align: center;">P50DDR</td> </tr> </table> <div style="display: flex; justify-content: space-between; align-items: flex-start;"> <div style="width: 45%;"> <p>モード1~4 {</p> <table style="border-collapse: collapse;"> <tr> <td style="border: none; padding-right: 5px;">初期値:</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> </tr> <tr> <td style="border: none; padding-right: 5px;">R/W:</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> </tr> </table>   <p>モード5~7 {</p> <table style="border-collapse: collapse;"> <tr> <td style="border: none; padding-right: 5px;">初期値:</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> </tr> <tr> <td style="border: none; padding-right: 5px;">R/W:</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> </tr> </table> </div> <div style="width: 50%; text-align: center; padding-top: 20px;"> <p>ポート5入出力選択</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>入力ポート</td> </tr> <tr> <td style="width: 20px; text-align: center;">1</td> <td>出力ポート</td> </tr> </table> </div> </div>				7	6	5	4	3	2	1	0		—	—	—	—	P53DDR	P52DDR	P51DDR	P50DDR	初期値:	1	1	1	1	1	1	1	1	R/W:	—	—	—	—	—	—	—	—	初期値:	1	1	1	1	0	0	0	0	R/W:	—	—	—	—	W	W	W	W	0	入力ポート	1	出力ポート
	7	6	5	4	3	2	1	0																																																				
	—	—	—	—	P53DDR	P52DDR	P51DDR	P50DDR																																																				
初期値:	1	1	1	1	1	1	1	1																																																				
R/W:	—	—	—	—	—	—	—	—																																																				
初期値:	1	1	1	1	0	0	0	0																																																				
R/W:	—	—	—	—	W	W	W	W																																																				
0	入力ポート																																																											
1	出力ポート																																																											
P6DDR ポート6 データディレクションレジスタ	H'EE005	ポート6																																																										
<p>ビット:      7          6          5          4          3          2          1          0</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="border: none;"></td> <td style="border: none; text-align: center;">7</td> <td style="border: none; text-align: center;">6</td> <td style="border: none; text-align: center;">5</td> <td style="border: none; text-align: center;">4</td> <td style="border: none; text-align: center;">3</td> <td style="border: none; text-align: center;">2</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">0</td> </tr> <tr> <td style="border: none;"></td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">P66DDR</td> <td style="border: 1px solid black; text-align: center;">P65DDR</td> <td style="border: 1px solid black; text-align: center;">P64DDR</td> <td style="border: 1px solid black; text-align: center;">P63DDR</td> <td style="border: 1px solid black; text-align: center;">P62DDR</td> <td style="border: 1px solid black; text-align: center;">P61DDR</td> <td style="border: 1px solid black; text-align: center;">P60DDR</td> </tr> </table> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="border: none; padding-right: 5px;">初期値:</td> <td style="border: none; text-align: center;">1</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> <td style="border: none; text-align: center;">0</td> </tr> <tr> <td style="border: none; padding-right: 5px;">R/W:</td> <td style="border: none; text-align: center;">—</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> <td style="border: none; text-align: center;">W</td> </tr> </table> <div style="text-align: center; padding-top: 20px;"> <p>ポート6入出力選択</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>入力ポート</td> </tr> <tr> <td style="width: 20px; text-align: center;">1</td> <td>出力ポート</td> </tr> </table> </div>				7	6	5	4	3	2	1	0		—	P66DDR	P65DDR	P64DDR	P63DDR	P62DDR	P61DDR	P60DDR	初期値:	1	0	0	0	0	0	0	0	R/W:	—	W	W	W	W	W	W	W	0	入力ポート	1	出力ポート																		
	7	6	5	4	3	2	1	0																																																				
	—	P66DDR	P65DDR	P64DDR	P63DDR	P62DDR	P61DDR	P60DDR																																																				
初期値:	1	0	0	0	0	0	0	0																																																				
R/W:	—	W	W	W	W	W	W	W																																																				
0	入力ポート																																																											
1	出力ポート																																																											



P8DDR ポート8 データディレクションレジスタ	H'EE007	ポート8						
ビット:	7	6	5	4	3	2	1	0
	—	—	—	P84DDR	P83DDR	P82DDR	P81DDR	P80DDR
モード1~4	初期値: 1	1	1	1	0	0	0	0
	R/W: —	—	—	W	W	W	W	W
モード5~7	初期値: 1	1	1	0	0	0	0	0
	R/W: —	—	—	W	W	W	W	W
				<div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;">                     ポート8入出力選択                      0 入力ポート                      1 出力ポート                 </div>				

P9DDR ポート9 データディレクションレジスタ	H'EE008	ポート9																				
<p>ビット:</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">7</td> <td style="padding: 5px;">6</td> <td style="padding: 5px;">5</td> <td style="padding: 5px;">4</td> <td style="padding: 5px;">3</td> <td style="padding: 5px;">2</td> <td style="padding: 5px;">1</td> <td style="padding: 5px;">0</td> </tr> <tr> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">P9<sub>5</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">P9<sub>4</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">P9<sub>3</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">P9<sub>2</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">P9<sub>1</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">P9<sub>0</sub>DDR</td> </tr> </table> <p>初期値:    1      1      0      0      0      0      0      0</p> <p>R/W:       —      —      W      W      W      W      W      W</p> <div style="text-align: center; margin: 10px 0;"> <p>ポート9入出力選択</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">0</td> <td style="padding: 2px 5px;">入力ポート</td> </tr> <tr> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">出力ポート</td> </tr> </table> </div>			7	6	5	4	3	2	1	0	—	—	P9 <sub>5</sub> DDR	P9 <sub>4</sub> DDR	P9 <sub>3</sub> DDR	P9 <sub>2</sub> DDR	P9 <sub>1</sub> DDR	P9 <sub>0</sub> DDR	0	入力ポート	1	出力ポート
7	6	5	4	3	2	1	0															
—	—	P9 <sub>5</sub> DDR	P9 <sub>4</sub> DDR	P9 <sub>3</sub> DDR	P9 <sub>2</sub> DDR	P9 <sub>1</sub> DDR	P9 <sub>0</sub> DDR															
0	入力ポート																					
1	出力ポート																					
<p>ビット:</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">7</td> <td style="padding: 5px;">6</td> <td style="padding: 5px;">5</td> <td style="padding: 5px;">4</td> <td style="padding: 5px;">3</td> <td style="padding: 5px;">2</td> <td style="padding: 5px;">1</td> <td style="padding: 5px;">0</td> </tr> <tr> <td style="border: 1px solid black; text-align: center;">PA<sub>7</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PA<sub>6</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PA<sub>5</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PA<sub>4</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PA<sub>3</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PA<sub>2</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PA<sub>1</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PA<sub>0</sub>DDR</td> </tr> </table> <p>モード3、4 { 初期値:    1      0      0      0      0      0      0      0</p> <p>                  R/W:    —      W      W      W      W      W      W      W</p> <p>モード        { 初期値:    0      0      0      0      0      0      0      0</p> <p>1、2、5~7 { R/W:     W      W      W      W      W      W      W      W</p> <div style="text-align: center; margin: 10px 0;"> <p>ポートA入出力選択</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">0</td> <td style="padding: 2px 5px;">入力ポート</td> </tr> <tr> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">出力ポート</td> </tr> </table> </div>			7	6	5	4	3	2	1	0	PA <sub>7</sub> DDR	PA <sub>6</sub> DDR	PA <sub>5</sub> DDR	PA <sub>4</sub> DDR	PA <sub>3</sub> DDR	PA <sub>2</sub> DDR	PA <sub>1</sub> DDR	PA <sub>0</sub> DDR	0	入力ポート	1	出力ポート
7	6	5	4	3	2	1	0															
PA <sub>7</sub> DDR	PA <sub>6</sub> DDR	PA <sub>5</sub> DDR	PA <sub>4</sub> DDR	PA <sub>3</sub> DDR	PA <sub>2</sub> DDR	PA <sub>1</sub> DDR	PA <sub>0</sub> DDR															
0	入力ポート																					
1	出力ポート																					
<p>ビット:</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">7</td> <td style="padding: 5px;">6</td> <td style="padding: 5px;">5</td> <td style="padding: 5px;">4</td> <td style="padding: 5px;">3</td> <td style="padding: 5px;">2</td> <td style="padding: 5px;">1</td> <td style="padding: 5px;">0</td> </tr> <tr> <td style="border: 1px solid black; text-align: center;">PB<sub>7</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PB<sub>6</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PB<sub>5</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PB<sub>4</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PB<sub>3</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PB<sub>2</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PB<sub>1</sub>DDR</td> <td style="border: 1px solid black; text-align: center;">PB<sub>0</sub>DDR</td> </tr> </table> <p>初期値:    0      0      0      0      0      0      0      0</p> <p>R/W:       W      W      W      W      W      W      W      W</p> <div style="text-align: center; margin: 10px 0;"> <p>ポートB入出力選択</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;">0</td> <td style="padding: 2px 5px;">入力ポート</td> </tr> <tr> <td style="padding: 2px 5px;">1</td> <td style="padding: 2px 5px;">出力ポート</td> </tr> </table> </div>			7	6	5	4	3	2	1	0	PB <sub>7</sub> DDR	PB <sub>6</sub> DDR	PB <sub>5</sub> DDR	PB <sub>4</sub> DDR	PB <sub>3</sub> DDR	PB <sub>2</sub> DDR	PB <sub>1</sub> DDR	PB <sub>0</sub> DDR	0	入力ポート	1	出力ポート
7	6	5	4	3	2	1	0															
PB <sub>7</sub> DDR	PB <sub>6</sub> DDR	PB <sub>5</sub> DDR	PB <sub>4</sub> DDR	PB <sub>3</sub> DDR	PB <sub>2</sub> DDR	PB <sub>1</sub> DDR	PB <sub>0</sub> DDR															
0	入力ポート																					
1	出力ポート																					

MDCR モードコントロールレジスタ	H'EE011	システム制御
--------------------	---------	--------



モードセレクト2~0

ビット2	ビット1	ビット0	動作モード
MD2	MD1	MD0	
0	0	0	—
		1	モード1
	1	0	モード2
1		モード3	
1	0	0	モード4
		1	モード5
	1	0	モード6
1		モード7	

【注】\* モード端子 (MD2 ~ MD0) の状態により決定されます。

SYSCR システムコントロールレジスタ	H'EE012	システム制御																																																						
<p>ビット:           7           6           5           4           3           2           1           0</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="width: 12.5%;">SSBY</td> <td style="width: 12.5%;">STS2</td> <td style="width: 12.5%;">STS1</td> <td style="width: 12.5%;">STS0</td> <td style="width: 12.5%;">UE</td> <td style="width: 12.5%;">NMIEG</td> <td style="width: 12.5%;">SSOE</td> <td style="width: 12.5%;">RAME</td> </tr> </table> <p>初期値:           0           0           0           0           1           0           0           1</p> <p>R/W:             R/W       R/W       R/W       R/W       R/W       R/W       R/W</p>	SSBY	STS2	STS1	STS0	UE	NMIEG	SSOE	RAME	<p style="text-align: center;">RAMイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>内蔵RAM無効</td> </tr> <tr> <td style="text-align: center;">1</td> <td>内蔵RAM有効</td> </tr> </table> <p style="text-align: center;">ソフトウェアスタンバイ 出力ポートイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>ソフトウェアスタンバイモード時、アドレスバス、バス制御信号はすべてハイインピーダンス</td> </tr> <tr> <td style="text-align: center;">1</td> <td>ソフトウェアスタンバイモード時、アドレスバス：出力状態を保持 バス制御信号：High固定</td> </tr> </table> <p style="text-align: center;">NMIエッジセレクト</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>NMI入力の立下がりエッジで割り込み要求を発生</td> </tr> <tr> <td style="text-align: center;">1</td> <td>NMI入力の立上がりエッジで割り込み要求を発生</td> </tr> </table> <p style="text-align: center;">ユーザービットイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>CCRのビット6 (UI) を割り込みマスクビットとして使用</td> </tr> <tr> <td style="text-align: center;">1</td> <td>CCRのビット6 (UI) をユーザビットとして使用</td> </tr> </table> <p style="text-align: center;">スタンバイタイムセレクト2~0</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <thead> <tr> <th style="width: 10%;">ビット6 STS2</th> <th style="width: 10%;">ビット5 STS1</th> <th style="width: 10%;">ビット4 STS0</th> <th style="width: 70%;">スタンバイタイムの指定</th> </tr> </thead> <tbody> <tr> <td rowspan="4" style="text-align: center; vertical-align: middle;">0</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">0</td> <td style="text-align: center;">0</td> <td>待機時間 = 8192ステート</td> </tr> <tr> <td style="text-align: center;">1</td> <td>待機時間 = 16384ステート</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>待機時間 = 32768ステート</td> </tr> <tr> <td style="text-align: center;">1</td> <td>待機時間 = 65536ステート</td> </tr> <tr> <td rowspan="4" style="text-align: center; vertical-align: middle;">1</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">0</td> <td style="text-align: center;">0</td> <td>待機時間 = 131072ステート</td> </tr> <tr> <td style="text-align: center;">1</td> <td>待機時間 = 262144ステート</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>待機時間 = 1024ステート</td> </tr> <tr> <td style="text-align: center;">1</td> <td>使用禁止</td> </tr> </tbody> </table> <p style="text-align: center;">ソフトウェアスタンバイ</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>SLEEP命令実行後、スリープモード遷移</td> </tr> <tr> <td style="text-align: center;">1</td> <td>SLEEP命令実行後、ソフトウェアスタンバイモードに遷移</td> </tr> </table>	0	内蔵RAM無効	1	内蔵RAM有効	0	ソフトウェアスタンバイモード時、アドレスバス、バス制御信号はすべてハイインピーダンス	1	ソフトウェアスタンバイモード時、アドレスバス：出力状態を保持 バス制御信号：High固定	0	NMI入力の立下がりエッジで割り込み要求を発生	1	NMI入力の立上がりエッジで割り込み要求を発生	0	CCRのビット6 (UI) を割り込みマスクビットとして使用	1	CCRのビット6 (UI) をユーザビットとして使用	ビット6 STS2	ビット5 STS1	ビット4 STS0	スタンバイタイムの指定	0	0	0	待機時間 = 8192ステート	1	待機時間 = 16384ステート	1	0	待機時間 = 32768ステート	1	待機時間 = 65536ステート	1	0	0	待機時間 = 131072ステート	1	待機時間 = 262144ステート	1	0	待機時間 = 1024ステート	1	使用禁止	0	SLEEP命令実行後、スリープモード遷移	1	SLEEP命令実行後、ソフトウェアスタンバイモードに遷移	
SSBY	STS2	STS1	STS0	UE	NMIEG	SSOE	RAME																																																	
0	内蔵RAM無効																																																							
1	内蔵RAM有効																																																							
0	ソフトウェアスタンバイモード時、アドレスバス、バス制御信号はすべてハイインピーダンス																																																							
1	ソフトウェアスタンバイモード時、アドレスバス：出力状態を保持 バス制御信号：High固定																																																							
0	NMI入力の立下がりエッジで割り込み要求を発生																																																							
1	NMI入力の立上がりエッジで割り込み要求を発生																																																							
0	CCRのビット6 (UI) を割り込みマスクビットとして使用																																																							
1	CCRのビット6 (UI) をユーザビットとして使用																																																							
ビット6 STS2	ビット5 STS1	ビット4 STS0	スタンバイタイムの指定																																																					
0	0	0	待機時間 = 8192ステート																																																					
		1	待機時間 = 16384ステート																																																					
	1	0	待機時間 = 32768ステート																																																					
		1	待機時間 = 65536ステート																																																					
1	0	0	待機時間 = 131072ステート																																																					
		1	待機時間 = 262144ステート																																																					
	1	0	待機時間 = 1024ステート																																																					
		1	使用禁止																																																					
0	SLEEP命令実行後、スリープモード遷移																																																							
1	SLEEP命令実行後、ソフトウェアスタンバイモードに遷移																																																							

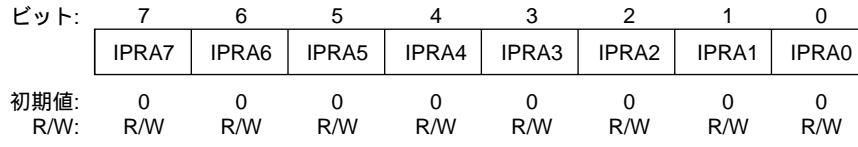
BRCR バスリリースコントロールレジスタ	H'EE013	バスコントローラ																		
ビット:      7          6          5          4          3          2          1          0																				
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%;">A23E</td> <td style="width: 10%;">A22E</td> <td style="width: 10%;">A21E</td> <td style="width: 10%;">A20E</td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> <td style="width: 10%;">BRLE</td> </tr> </table>				A23E	A22E	A21E	A20E	—	—	—	BRLE									
	A23E	A22E	A21E	A20E	—	—	—	BRLE												
モード 1、2、6、7	{ 初期値: R/W:	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">0</td> </tr> <tr> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">R/W</td> </tr> </table>	1	1	1	1	1	1	1	1	0	—	—	—	—	—	—	—	—	R/W
1	1	1	1	1	1	1	1	0												
—	—	—	—	—	—	—	—	R/W												
モード 3、4	{ 初期値: R/W:	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">0</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">0</td> </tr> <tr> <td style="border: none;">R/W</td> <td style="border: none;">R/W</td> <td style="border: none;">R/W</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">R/W</td> </tr> </table>	1	1	1	0	1	1	1	1	0	R/W	R/W	R/W	—	—	—	—	—	R/W
1	1	1	0	1	1	1	1	0												
R/W	R/W	R/W	—	—	—	—	—	R/W												
モード5	{ 初期値: R/W:	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">1</td> <td style="width: 10%;">0</td> </tr> <tr> <td style="border: none;">R/W</td> <td style="border: none;">R/W</td> <td style="border: none;">R/W</td> <td style="border: none;">R/W</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">—</td> <td style="border: none;">R/W</td> </tr> </table>	1	1	1	1	1	1	1	1	0	R/W	R/W	R/W	R/W	—	—	—	—	R/W
1	1	1	1	1	1	1	1	0												
R/W	R/W	R/W	R/W	—	—	—	—	R/W												
<table border="1" style="margin-left: auto; margin-right: auto; width: 60%;"> <tr> <td colspan="2" style="text-align: center;">アドレス23~20イネーブル</td> </tr> <tr> <td style="width: 20px; text-align: center;">0</td> <td>アドレス出力</td> </tr> <tr> <td style="text-align: center;">1</td> <td>上記以外の入出力端子</td> </tr> </table>			アドレス23~20イネーブル		0	アドレス出力	1	上記以外の入出力端子												
アドレス23~20イネーブル																				
0	アドレス出力																			
1	上記以外の入出力端子																			
<table border="1" style="margin-left: auto; margin-right: auto; width: 60%;"> <tr> <td colspan="2" style="text-align: center;">バスリリースイネーブル</td> </tr> <tr> <td style="width: 20px; text-align: center;">0</td> <td>バス権の外部に対する解放を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>バス権の外部に対する解放を許可</td> </tr> </table>			バスリリースイネーブル		0	バス権の外部に対する解放を禁止	1	バス権の外部に対する解放を許可												
バスリリースイネーブル																				
0	バス権の外部に対する解放を禁止																			
1	バス権の外部に対する解放を許可																			

ISCR IRQ センスコントロールレジスタ	H'EE014	割り込みコントローラ									
ビット:      7          6          5          4          3          2          1          0											
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> <td style="width: 10%;">IRQ5SC</td> <td style="width: 10%;">IRQ4SC</td> <td style="width: 10%;">IRQ3SC</td> <td style="width: 10%;">IRQ2SC</td> <td style="width: 10%;">IRQ1SC</td> <td style="width: 10%;">IRQ0SC</td> </tr> </table>				—	—	IRQ5SC	IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC
	—	—	IRQ5SC	IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC			
初期値:      0          0          0          0          0          0          0          0											
R/W:      R/W      R/W      R/W      R/W      R/W      R/W      R/W											
<table border="1" style="margin-left: auto; margin-right: auto; width: 60%;"> <tr> <td colspan="2" style="text-align: center;">IRQ5~IRQ0センスコントロール</td> </tr> <tr> <td style="width: 20px; text-align: center;">0</td> <td>IRQ5~IRQ0入力のLowレベルで割り込み要求を発生</td> </tr> <tr> <td style="text-align: center;">1</td> <td>IRQ5~IRQ0入力の立下がりエッジで割り込み要求を発生</td> </tr> </table>			IRQ5~IRQ0センスコントロール		0	IRQ5~IRQ0入力のLowレベルで割り込み要求を発生	1	IRQ5~IRQ0入力の立下がりエッジで割り込み要求を発生			
IRQ5~IRQ0センスコントロール											
0	IRQ5~IRQ0入力のLowレベルで割り込み要求を発生										
1	IRQ5~IRQ0入力の立下がりエッジで割り込み要求を発生										

IER IRQ イネーブルレジスタ	H'EE015	割り込みコントローラ																						
<p>ビット:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px;">7</td> <td style="width: 20px;">6</td> <td style="width: 20px;">5</td> <td style="width: 20px;">4</td> <td style="width: 20px;">3</td> <td style="width: 20px;">2</td> <td style="width: 20px;">1</td> <td style="width: 20px;">0</td> </tr> <tr> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">IRQ5E</td> <td style="text-align: center;">IRQ4E</td> <td style="text-align: center;">IRQ3E</td> <td style="text-align: center;">IRQ2E</td> <td style="text-align: center;">IRQ1E</td> <td style="text-align: center;">IRQ0E</td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0 R/W: R/W R/W R/W R/W R/W R/W R/W</p> <p style="text-align: center;">IRQ5 ~ IRQ0 イネーブル</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px;">0</td> <td>IRQ5 ~ IRQ0 割り込みを禁止</td> </tr> <tr> <td>1</td> <td>IRQ5 ~ IRQ0 割り込みを許可</td> </tr> </table>			7	6	5	4	3	2	1	0	—	—	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E	0	IRQ5 ~ IRQ0 割り込みを禁止	1	IRQ5 ~ IRQ0 割り込みを許可		
7	6	5	4	3	2	1	0																	
—	—	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E																	
0	IRQ5 ~ IRQ0 割り込みを禁止																							
1	IRQ5 ~ IRQ0 割り込みを許可																							
<p>ISR IRQ ステータスレジスタ</p> <p>H'EE016</p> <p>割り込みコントローラ</p>																								
<p>ビット:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 20px;">7</td> <td style="width: 20px;">6</td> <td style="width: 20px;">5</td> <td style="width: 20px;">4</td> <td style="width: 20px;">3</td> <td style="width: 20px;">2</td> <td style="width: 20px;">1</td> <td style="width: 20px;">0</td> </tr> <tr> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">IRQ5F</td> <td style="text-align: center;">IRQ4F</td> <td style="text-align: center;">IRQ3F</td> <td style="text-align: center;">IRQ2F</td> <td style="text-align: center;">IRQ1F</td> <td style="text-align: center;">IRQ0F</td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0 R/W: — — R/(W)* R/(W)* R/(W)* R/(W)* R/(W)* R/(W)*</p> <p style="text-align: center;">IRQ5 ~ IRQ0 フラグ</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="width: 100px;">ビット5~0</th> <th>セット/クリア条件</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>           [クリア条件]            (1) IRQnF = 1の状態ではIRQnFをリードした後、IRQnFに0をライトしたとき            (2) IRQnSC = 0、<math>\overline{\text{IRQn}}</math>入力が、Highレベルの状態では割り込み例外処理を実行したとき            (3) IRQnSC = 1の状態ではIRQn割り込み例外処理を実行したとき         </td> </tr> <tr> <td style="text-align: center;">1</td> <td>           [セット条件]            (1) IRQnSC = 0の状態では<math>\overline{\text{IRQn}}</math>入力がLowレベルになったとき            (2) IRQnSC = 1の状態では<math>\overline{\text{IRQn}}</math>入力が立上がりエッジが発生したとき         </td> </tr> </tbody> </table> <p style="text-align: right;">(n = 5 ~ 0)</p> <p>【注】* フラグをクリアするための0ライトのみ可能です。</p>			7	6	5	4	3	2	1	0	—	—	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F	ビット5~0	セット/クリア条件	0	[クリア条件] (1) IRQnF = 1の状態ではIRQnFをリードした後、IRQnFに0をライトしたとき (2) IRQnSC = 0、 $\overline{\text{IRQn}}$ 入力が、Highレベルの状態では割り込み例外処理を実行したとき (3) IRQnSC = 1の状態ではIRQn割り込み例外処理を実行したとき	1	[セット条件] (1) IRQnSC = 0の状態では $\overline{\text{IRQn}}$ 入力がLowレベルになったとき (2) IRQnSC = 1の状態では $\overline{\text{IRQn}}$ 入力が立上がりエッジが発生したとき
7	6	5	4	3	2	1	0																	
—	—	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F																	
ビット5~0	セット/クリア条件																							
0	[クリア条件] (1) IRQnF = 1の状態ではIRQnFをリードした後、IRQnFに0をライトしたとき (2) IRQnSC = 0、 $\overline{\text{IRQn}}$ 入力が、Highレベルの状態では割り込み例外処理を実行したとき (3) IRQnSC = 1の状態ではIRQn割り込み例外処理を実行したとき																							
1	[セット条件] (1) IRQnSC = 0の状態では $\overline{\text{IRQn}}$ 入力がLowレベルになったとき (2) IRQnSC = 1の状態では $\overline{\text{IRQn}}$ 入力が立上がりエッジが発生したとき																							

IPRA インタラプトプライオリティレジスタ A H'EE018 割り込みコントローラ



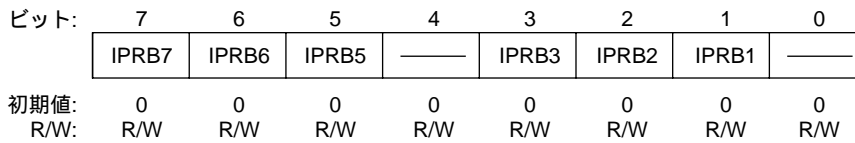
プライオリティレベルA7 ~ A0

0	プライオリティレベル0 (非優先)
1	プライオリティレベル1 (優先)

割り込み要因と各ビットの対応

IPRA	ビット	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
		IPRA7	IPRA6	IPRA5	IPRA4	IPRA3	IPRA2	IPRA1	IPRA0
	割り込み要因	IRQ0	IRQ1	IRQ2、 IRQ3	IRQ4、 IRQ5	WDT、 DRAM インタ フェース、 A/D変換器	16ビット タイマ チャンネル0	16ビット タイマ チャンネル1	16ビット タイマ チャンネル2

IPRB インタラプトプライオリティレジスタ B H'EE019 割り込みコントローラ



プライオリティレベルB7 ~ B5、B3 ~ B1

0	プライオリティレベル0 (非優先)
1	プライオリティレベル1 (優先)

割り込み要因と各ビットの対応

IPRB	ビット	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
		IPRB7	IPRB6	IPRB5	——	IPRB3	IPRB2	IPRB1	——
	割り込み要因	8ビット タイマ チャンネル 0, 1	8ビット タイマ チャンネル 2, 3	DMAC	——	SCI チャンネル0	SCI チャンネル1	SCI チャンネル2	——

DASTCR D/Aスタンバイコントロールレジスタ	H'EE01A	D/A																
ビット: <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">DASTE</td> </tr> </table>	7	6	5	4	3	2	1	0	—	—	—	—	—	—	—	DASTE		
7	6	5	4	3	2	1	0											
—	—	—	—	—	—	—	DASTE											
初期値: <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> </table>	1	1	1	1	1	1	1	0										
1	1	1	1	1	1	1	0											
R/W: <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">R/W</td> </tr> </table>	—	—	—	—	—	—	—	R/W										
—	—	—	—	—	—	—	R/W											
D/Aスタンバイネーブル <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">0</td> <td>ソフトウェアスタンバイモードでのD/A出力を禁止 (初期値)</td> </tr> <tr> <td style="text-align: center;">1</td> <td>ソフトウェアスタンバイモードでのD/A出力を許可</td> </tr> </table>			0	ソフトウェアスタンバイモードでのD/A出力を禁止 (初期値)	1	ソフトウェアスタンバイモードでのD/A出力を許可												
0	ソフトウェアスタンバイモードでのD/A出力を禁止 (初期値)																	
1	ソフトウェアスタンバイモードでのD/A出力を許可																	



DIVCR 分周比コントロールレジスタ	H'EE01B	システム制御																								
<p>ビット:      7      6      5      4      3      2      1      0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">—</td> <td style="width: 20px; text-align: center;">—</td> <td style="width: 20px; text-align: center;">—</td> <td style="width: 20px; text-align: center;">—</td> <td style="width: 20px; text-align: center;">—</td> <td style="width: 20px; text-align: center;">—</td> <td style="width: 20px; text-align: center;">DIV1</td> <td style="width: 20px; text-align: center;">DIV0</td> </tr> </table> <p>初期値:      1      1      1      1      1      1      0      0</p> <p>R/W:        —      —      —      —      —      —      R/W      R/W</p> <div style="margin-left: 200px; margin-top: 20px;"> <p>分周比1、0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 30px;">ビット1</th> <th style="width: 30px;">ビット0</th> <th style="width: 100px;">分 周 比</th> </tr> <tr> <th>DIV1</th> <th>DIV0</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>1/1    (初期値)</td> </tr> <tr> <td style="text-align: center;">1</td> <td>1/2</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>1/4</td> </tr> <tr> <td style="text-align: center;">1</td> <td>1/8</td> </tr> </tbody> </table> </div>			—	—	—	—	—	—	DIV1	DIV0	ビット1	ビット0	分 周 比	DIV1	DIV0		0	0	1/1    (初期値)	1	1/2	1	0	1/4	1	1/8
—	—	—	—	—	—	DIV1	DIV0																			
ビット1	ビット0	分 周 比																								
DIV1	DIV0																									
0	0	1/1    (初期値)																								
	1	1/2																								
1	0	1/4																								
	1	1/8																								

MSTCRH モジュールスタンバイコントロールレジスタH	H'EE01C	システム制御								
<p>ビット：        7        6        5        4        3        2        1        0</p> <table border="1"> <tr> <td>PSTOP</td> <td>—</td> <td>—</td> <td>—</td> <td>—</td> <td>MSTPH2</td> <td>MSTPH1</td> <td>MSTPH0</td> </tr> </table> <p>初期値：        0        1        1        1        1        0        0        0</p> <p>R/W        :        R/W        —        —        —        —        R/W        R/W        R/W</p> <p> <span style="margin-left: 100px;">リザーブビット</span>  <span style="margin-left: 200px;">クロックストップ</span>  <span style="margin-left: 200px;">クロックの出力を許可または禁止するビットです。</span> </p> <p style="text-align: right;"> <span style="margin-right: 100px;">モジュールスタンバイH2~0</span>  <span style="margin-right: 100px;">モジュールをスタンバイ状態に</span>  <span style="margin-right: 100px;">するための選択ビットです。</span> </p>			PSTOP	—	—	—	—	MSTPH2	MSTPH1	MSTPH0
PSTOP	—	—	—	—	MSTPH2	MSTPH1	MSTPH0			
MSTCRL モジュールスタンバイコントロールレジスタL	H'EE01D	システム制御								
<p>ビット：        7        6        5        4        3        2        1        0</p> <table border="1"> <tr> <td>MSTPL7</td> <td>—</td> <td>MSTPL5</td> <td>MSTPL4</td> <td>MSTPL3</td> <td>MSTPL2</td> <td>—</td> <td>MSTPL0</td> </tr> </table> <p>初期値：        0        0        0        0        0        0        0        0</p> <p>R/W        :        R/W        R/W        R/W        R/W        R/W        R/W        R/W</p> <p> <span style="margin-left: 100px;">リザーブビット</span>  <span style="margin-left: 200px;">モジュールスタンバイL7、L5~2、L0</span>  <span style="margin-left: 200px;">モジュールをスタンバイ状態に</span>  <span style="margin-left: 200px;">するための選択ビットです。</span> </p>			MSTPL7	—	MSTPL5	MSTPL4	MSTPL3	MSTPL2	—	MSTPL0
MSTPL7	—	MSTPL5	MSTPL4	MSTPL3	MSTPL2	—	MSTPL0			

ADRCR アドレスコントロールレジスタ	H'EE01E	バスコントローラ																																
<p>ビット:      7          6          5          4          3          2          1          0</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">ADRCTL</td> </tr> <tr> <td style="padding: 2px;">初期値:</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> <tr> <td style="padding: 2px;">R/W:</td> <td colspan="6"></td> <td style="text-align: center;">R/W</td> </tr> </table> <p style="text-align: center; margin-top: 10px;"> <span style="margin-right: 150px;">リザーブビット</span> <span>アドレスコントロール</span>  <span style="margin-right: 150px;"></span> <span>アドレス更新モード1または</span>  <span style="margin-right: 150px;"></span> <span>アドレス更新モード2を選択</span> </p> <table style="margin: auto; border-collapse: collapse; width: 80%;"> <thead> <tr> <th style="width: 10%;">ADRCTL</th> <th style="width: 80%;">説 明</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>アドレス更新モード2を選択</td> </tr> <tr> <td style="text-align: center;">1</td> <td>アドレス更新モード1を選択 <span style="float: right;">(初期値)</span></td> </tr> </tbody> </table>										ADRCTL	初期値:	1	1	1	1	1	1	1	R/W:							R/W	ADRCTL	説 明	0	アドレス更新モード2を選択	1	アドレス更新モード1を選択 <span style="float: right;">(初期値)</span>		
							ADRCTL																											
初期値:	1	1	1	1	1	1	1																											
R/W:							R/W																											
ADRCTL	説 明																																	
0	アドレス更新モード2を選択																																	
1	アドレス更新モード1を選択 <span style="float: right;">(初期値)</span>																																	
CSCR チップセレクトコントロールレジスタ	H'EE01F	バスコントローラ																																
<p>ビット:      7          6          5          4          3          2          1          0</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">CS7E</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">CS6E</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">CS5E</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">CS4E</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">—</td> </tr> <tr> <td style="padding: 2px;">初期値:</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> <tr> <td style="padding: 2px;">R/W:</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> </tr> </table> <p style="text-align: center; margin-top: 10px;">チップセレクト7~4イネーブル</p> <table style="margin: auto; border-collapse: collapse; width: 80%;"> <thead> <tr> <th style="width: 10%;">ビットn</th> <th style="width: 80%;">説 明</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">CSnE</td> <td></td> </tr> <tr> <td style="text-align: center;">0</td> <td>チップセレクト信号 (CSn) の出力を禁止 (初期値)</td> </tr> <tr> <td style="text-align: center;">1</td> <td>チップセレクト信号 (CSn) の出力を許可</td> </tr> </tbody> </table> <p style="text-align: center; margin-top: 5px;">n=7~4</p>			CS7E	CS6E	CS5E	CS4E	—	—	—	—	初期値:	0	0	0	0	1	1	1	R/W:	R/W	R/W	R/W	R/W	—	—	—	ビットn	説 明	CSnE		0	チップセレクト信号 (CSn) の出力を禁止 (初期値)	1	チップセレクト信号 (CSn) の出力を許可
CS7E	CS6E	CS5E	CS4E	—	—	—	—																											
初期値:	0	0	0	0	1	1	1																											
R/W:	R/W	R/W	R/W	R/W	—	—	—																											
ビットn	説 明																																	
CSnE																																		
0	チップセレクト信号 (CSn) の出力を禁止 (初期値)																																	
1	チップセレクト信号 (CSn) の出力を許可																																	

ABWCR	バス幅コントロールレジスタ	H'EE020	バスコントローラ																																													
	ビット: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th></th> <th>7</th> <th>6</th> <th>5</th> <th>4</th> <th>3</th> <th>2</th> <th>1</th> <th>0</th> </tr> </thead> <tbody> <tr> <td></td> <td>ABW7</td> <td>ABW6</td> <td>ABW5</td> <td>ABW4</td> <td>ABW3</td> <td>ABW2</td> <td>ABW1</td> <td>ABW0</td> </tr> <tr> <td>モード1、3、5、6、7</td> <td>初期値:</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>モード2、4</td> <td>初期値:</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td></td> <td>R/W:</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </tbody> </table>		7	6	5	4	3	2	1	0		ABW7	ABW6	ABW5	ABW4	ABW3	ABW2	ABW1	ABW0	モード1、3、5、6、7	初期値:	1	1	1	1	1	1	1	モード2、4	初期値:	0	0	0	0	0	0	0		R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W		
	7	6	5	4	3	2	1	0																																								
	ABW7	ABW6	ABW5	ABW4	ABW3	ABW2	ABW1	ABW0																																								
モード1、3、5、6、7	初期値:	1	1	1	1	1	1	1																																								
モード2、4	初期値:	0	0	0	0	0	0	0																																								
	R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W																																								
		エリア7~0バス幅コントロール <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>ビット7~0</th> <th></th> </tr> </thead> <tbody> <tr> <td>ABW7 ~ABW0</td> <td style="text-align: center;">アクセス空間の指定</td> </tr> <tr> <td>0</td> <td>エリア7~0を16ビットアクセス空間に設定</td> </tr> <tr> <td>1</td> <td>エリア7~0を8ビットアクセス空間に設定</td> </tr> </tbody> </table>		ビット7~0		ABW7 ~ABW0	アクセス空間の指定	0	エリア7~0を16ビットアクセス空間に設定	1	エリア7~0を8ビットアクセス空間に設定																																					
ビット7~0																																																
ABW7 ~ABW0	アクセス空間の指定																																															
0	エリア7~0を16ビットアクセス空間に設定																																															
1	エリア7~0を8ビットアクセス空間に設定																																															
ASTCR	アクセスステートコントロールレジスタ	H'EE021	バスコントローラ																																													
	ビット: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th></th> <th>7</th> <th>6</th> <th>5</th> <th>4</th> <th>3</th> <th>2</th> <th>1</th> <th>0</th> </tr> </thead> <tbody> <tr> <td></td> <td>AST7</td> <td>AST6</td> <td>AST5</td> <td>AST4</td> <td>AST3</td> <td>AST2</td> <td>AST1</td> <td>AST0</td> </tr> <tr> <td>初期値:</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>R/W:</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </tbody> </table>		7	6	5	4	3	2	1	0		AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0	初期値:	1	1	1	1	1	1	1	1	R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W											
	7	6	5	4	3	2	1	0																																								
	AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0																																								
初期値:	1	1	1	1	1	1	1	1																																								
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																																								
		エリア7~0アクセスステートコントロール <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>ビット7~0</th> <th></th> </tr> </thead> <tbody> <tr> <td>AST7 ~AST0</td> <td style="text-align: center;">アクセスステート数の指定</td> </tr> <tr> <td>0</td> <td>エリア7~0を2ステートアクセス空間に設定</td> </tr> <tr> <td>1</td> <td>エリア7~0を3ステートアクセス空間に設定</td> </tr> </tbody> </table>		ビット7~0		AST7 ~AST0	アクセスステート数の指定	0	エリア7~0を2ステートアクセス空間に設定	1	エリア7~0を3ステートアクセス空間に設定																																					
ビット7~0																																																
AST7 ~AST0	アクセスステート数の指定																																															
0	エリア7~0を2ステートアクセス空間に設定																																															
1	エリア7~0を3ステートアクセス空間に設定																																															

WCRH ウェイトコントロールレジスタH	H'EE022	バスコントローラ																																																
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1" style="margin-left: 20px; border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">W71</td><td style="width: 20px;">W70</td><td style="width: 20px;">W61</td><td style="width: 20px;">W60</td><td style="width: 20px;">W51</td><td style="width: 20px;">W50</td><td style="width: 20px;">W41</td><td style="width: 20px;">W40</td> </tr> </table> <p>初期値:      1          1          1          1          1          1          1          1</p> <p>R/W:        R/W      R/W      R/W      R/W      R/W      R/W      R/W</p> <div style="margin-left: 100px;"> <p>エリア4ウェイトコントロール1、0</p> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td rowspan="2" style="width: 20px;">0</td><td style="width: 20px;">0</td><td>プログラムウェイトを挿入しない</td> </tr> <tr> <td>1</td><td>プログラムウェイトを1ステート挿入</td> </tr> <tr> <td rowspan="2" style="width: 20px;">1</td><td>0</td><td>プログラムウェイトを2ステート挿入</td> </tr> <tr> <td>1</td><td>プログラムウェイトを3ステート挿入</td> </tr> </table>   <p>エリア5ウェイトコントロール1、0</p> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td rowspan="2" style="width: 20px;">0</td><td style="width: 20px;">0</td><td>プログラムウェイトを挿入しない</td> </tr> <tr> <td>1</td><td>プログラムウェイトを1ステート挿入</td> </tr> <tr> <td rowspan="2" style="width: 20px;">1</td><td>0</td><td>プログラムウェイトを2ステート挿入</td> </tr> <tr> <td>1</td><td>プログラムウェイトを3ステート挿入</td> </tr> </table>   <p>エリア6ウェイトコントロール1、0</p> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td rowspan="2" style="width: 20px;">0</td><td style="width: 20px;">0</td><td>プログラムウェイトを挿入しない</td> </tr> <tr> <td>1</td><td>プログラムウェイトを1ステート挿入</td> </tr> <tr> <td rowspan="2" style="width: 20px;">1</td><td>0</td><td>プログラムウェイトを2ステート挿入</td> </tr> <tr> <td>1</td><td>プログラムウェイトを3ステート挿入</td> </tr> </table>   <p>エリア7ウェイトコントロール1、0</p> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td rowspan="2" style="width: 20px;">0</td><td style="width: 20px;">0</td><td>プログラムウェイトを挿入しない</td> </tr> <tr> <td>1</td><td>プログラムウェイトを1ステート挿入</td> </tr> <tr> <td rowspan="2" style="width: 20px;">1</td><td>0</td><td>プログラムウェイトを2ステート挿入</td> </tr> <tr> <td>1</td><td>プログラムウェイトを3ステート挿入</td> </tr> </table> </div>			W71	W70	W61	W60	W51	W50	W41	W40	0	0	プログラムウェイトを挿入しない	1	プログラムウェイトを1ステート挿入	1	0	プログラムウェイトを2ステート挿入	1	プログラムウェイトを3ステート挿入	0	0	プログラムウェイトを挿入しない	1	プログラムウェイトを1ステート挿入	1	0	プログラムウェイトを2ステート挿入	1	プログラムウェイトを3ステート挿入	0	0	プログラムウェイトを挿入しない	1	プログラムウェイトを1ステート挿入	1	0	プログラムウェイトを2ステート挿入	1	プログラムウェイトを3ステート挿入	0	0	プログラムウェイトを挿入しない	1	プログラムウェイトを1ステート挿入	1	0	プログラムウェイトを2ステート挿入	1	プログラムウェイトを3ステート挿入
W71	W70	W61	W60	W51	W50	W41	W40																																											
0	0	プログラムウェイトを挿入しない																																																
	1	プログラムウェイトを1ステート挿入																																																
1	0	プログラムウェイトを2ステート挿入																																																
	1	プログラムウェイトを3ステート挿入																																																
0	0	プログラムウェイトを挿入しない																																																
	1	プログラムウェイトを1ステート挿入																																																
1	0	プログラムウェイトを2ステート挿入																																																
	1	プログラムウェイトを3ステート挿入																																																
0	0	プログラムウェイトを挿入しない																																																
	1	プログラムウェイトを1ステート挿入																																																
1	0	プログラムウェイトを2ステート挿入																																																
	1	プログラムウェイトを3ステート挿入																																																
0	0	プログラムウェイトを挿入しない																																																
	1	プログラムウェイトを1ステート挿入																																																
1	0	プログラムウェイトを2ステート挿入																																																
	1	プログラムウェイトを3ステート挿入																																																

WCRL ウェイトコントロールレジスタL	H'EE023	バスコントローラ
----------------------	---------	----------

ビット:	7	6	5	4	3	2	1	0
	W31	W30	W21	W20	W11	W10	W01	W00
初期値:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

エリア0 ウェイトコントロール1、0

0	0	プログラムウェイトを挿入しない
	1	プログラムウェイトを1ステート挿入
1	0	プログラムウェイトを2ステート挿入
	1	プログラムウェイトを3ステート挿入

エリア1ウェイトコントロール1、0

0	0	プログラムウェイトを挿入しない
	1	プログラムウェイトを1ステート挿入
1	0	プログラムウェイトを2ステート挿入
	1	プログラムウェイトを3ステート挿入

エリア2ウェイトコントロール1、0

0	0	プログラムウェイトを挿入しない
	1	プログラムウェイトを1ステート挿入
1	0	プログラムウェイトを2ステート挿入
	1	プログラムウェイトを3ステート挿入

エリア3ウェイトコントロール1、0

0	0	プログラムウェイトを挿入しない
	1	プログラムウェイトを1ステート挿入
1	0	プログラムウェイトを2ステート挿入
	1	プログラムウェイトを3ステート挿入

BCR バスコントロールレジスタ	H'EE024	バスコントローラ										
ビット:      7          6          5          4          3          2          1          0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;">ICIS1</td> <td style="width: 10%;">ICIS0</td> <td style="width: 10%;">BROME</td> <td style="width: 10%;">BRSTS1</td> <td style="width: 10%;">BRSTS0</td> <td style="width: 10%;"></td> <td style="width: 10%;">RDEA</td> <td style="width: 10%;">WAITE</td> </tr> </table>	ICIS1	ICIS0	BROME	BRSTS1	BRSTS0		RDEA	WAITE				
ICIS1	ICIS0	BROME	BRSTS1	BRSTS0		RDEA	WAITE					
初期値: R/W:	1 R/W	1 R/W	0 R/W	0 R/W	0 R/W	1	1 R/W	0 R/W				
							<table border="1" style="width: 100%;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>端子ウェイト入力を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>端子ウェイト入力を許可</td> </tr> </table>	0	端子ウェイト入力を禁止	1	端子ウェイト入力を許可	
0	端子ウェイト入力を禁止											
1	端子ウェイト入力を許可											
							エリア分割単位選択 <table border="1" style="width: 100%;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>                             エリア分割は、                              エリア0：2Mバイト    エリア4：1.93Mバイト                              エリア1：2Mバイト    エリア5：4kバイト                              エリア2：8Mバイト    エリア6：23.75kバイト                              エリア3：2Mバイト    エリア7：22バイト                         </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                             エリア分割は、エリア0～7まで                              等分割（2Mバイト）                         </td> </tr> </table>		0	エリア分割は、 エリア0：2Mバイト    エリア4：1.93Mバイト エリア1：2Mバイト    エリア5：4kバイト エリア2：8Mバイト    エリア6：23.75kバイト エリア3：2Mバイト    エリア7：22バイト	1	エリア分割は、エリア0～7まで 等分割（2Mバイト）
0	エリア分割は、 エリア0：2Mバイト    エリア4：1.93Mバイト エリア1：2Mバイト    エリア5：4kバイト エリア2：8Mバイト    エリア6：23.75kバイト エリア3：2Mバイト    エリア7：22バイト											
1	エリア分割は、エリア0～7まで 等分割（2Mバイト）											
							バーストサイクルセレクト0 <table border="1" style="width: 100%;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>バーストアクセスは最大4ワード</td> </tr> <tr> <td style="text-align: center;">1</td> <td>バーストアクセスは最大8ワード</td> </tr> </table>		0	バーストアクセスは最大4ワード	1	バーストアクセスは最大8ワード
0	バーストアクセスは最大4ワード											
1	バーストアクセスは最大8ワード											
							バーストサイクルセレクト1 <table border="1" style="width: 100%;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>バーストアクセスのサイクル数は2ステート</td> </tr> <tr> <td style="text-align: center;">1</td> <td>バーストアクセスのサイクル数は3ステート</td> </tr> </table>		0	バーストアクセスのサイクル数は2ステート	1	バーストアクセスのサイクル数は3ステート
0	バーストアクセスのサイクル数は2ステート											
1	バーストアクセスのサイクル数は3ステート											
							バーストROMイネーブル <table border="1" style="width: 100%;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>エリア0は基本バスインタフェース</td> </tr> <tr> <td style="text-align: center;">1</td> <td>エリア0はバーストROMインタフェース</td> </tr> </table>		0	エリア0は基本バスインタフェース	1	エリア0はバーストROMインタフェース
0	エリア0は基本バスインタフェース											
1	エリア0はバーストROMインタフェース											
							アイドルサイクル挿入0 <table border="1" style="width: 100%;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入しない</td> </tr> <tr> <td style="text-align: center;">1</td> <td>外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入する</td> </tr> </table>		0	外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入しない	1	外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入する
0	外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入しない											
1	外部リードサイクルと外部ライトサイクルが連続したとき、アイドルサイクルを挿入する											
							アイドルサイクル挿入1 <table border="1" style="width: 100%;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入しない</td> </tr> <tr> <td style="text-align: center;">1</td> <td>異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入する</td> </tr> </table>		0	異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入しない	1	異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入する
0	異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入しない											
1	異なるエリアの外部リードサイクルが連続したとき、アイドルサイクルを挿入する											

DRCRA DRAMコントロールレジスタ A	H'EE026	DRAMインタフェース																																																					
ビット：      7      6      5      4      3      2      1      0 <table border="1" style="margin-left: 20px; border-collapse: collapse;"> <tr> <td style="width: 10%;">DRAS2</td><td style="width: 10%;">DRAS1</td><td style="width: 10%;">DRAS0</td><td style="width: 10%;">—</td><td style="width: 10%;">BE</td><td style="width: 10%;">RDM</td><td style="width: 10%;">SRFMD</td><td style="width: 10%;">RFSHE</td></tr> </table> 初期値：      0      0      0      1      0      0      0      0 R/W      :      R/W    R/W    R/W    —    R/W    R/W    R/W    R/W	DRAS2	DRAS1	DRAS0	—	BE	RDM	SRFMD	RFSHE																																															
DRAS2	DRAS1	DRAS0	—	BE	RDM	SRFMD	RFSHE																																																
<b>リフレッシュ端子イネーブル</b> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td><td>RFSH端子のリフレッシュ信号出力を禁止</td></tr> <tr> <td>1</td><td>RFSH端子のリフレッシュ信号出力を許可</td></tr> </table>			0	RFSH端子のリフレッシュ信号出力を禁止	1	RFSH端子のリフレッシュ信号出力を許可																																																	
0	RFSH端子のリフレッシュ信号出力を禁止																																																						
1	RFSH端子のリフレッシュ信号出力を許可																																																						
<b>セルフリフレッシュモード</b> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td><td>ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュを禁止</td></tr> <tr> <td>1</td><td>ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュが可能</td></tr> </table>			0	ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュを禁止	1	ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュが可能																																																	
0	ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュを禁止																																																						
1	ソフトウェアスタンバイモード時に、DRAMのセルフリフレッシュが可能																																																						
<b>RASダウンモード</b> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td><td>DRAMインタフェース：RASアップモードを選択</td></tr> <tr> <td>1</td><td>DRAMインタフェース：RASダウンモードを選択</td></tr> </table>			0	DRAMインタフェース：RASアップモードを選択	1	DRAMインタフェース：RASダウンモードを選択																																																	
0	DRAMインタフェース：RASアップモードを選択																																																						
1	DRAMインタフェース：RASダウンモードを選択																																																						
<b>バーストアクセスイネーブル</b> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td><td>バースト禁止（常にフルアクセス）</td></tr> <tr> <td>1</td><td>DRAM空間アクセス時 高速ページモードでアクセス</td></tr> </table>			0	バースト禁止（常にフルアクセス）	1	DRAM空間アクセス時 高速ページモードでアクセス																																																	
0	バースト禁止（常にフルアクセス）																																																						
1	DRAM空間アクセス時 高速ページモードでアクセス																																																						
<b>DRAMエリアセレクト</b>																																																							
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>DRAS2</th><th>DRAS1</th><th>DRAS0</th><th>エリア5</th><th>エリア4</th><th>エリア3</th><th>エリア2</th></tr> </thead> <tbody> <tr> <td rowspan="4">0</td><td rowspan="2">0</td><td>0</td><td>通常</td><td>通常</td><td>通常</td><td>通常</td></tr> <tr> <td>1</td><td>通常</td><td>通常</td><td>通常</td><td>DRAM空間 (<math>\overline{CS}_2</math>)</td></tr> <tr> <td rowspan="2">1</td><td>0</td><td>通常</td><td>通常</td><td>DRAM空間 (<math>\overline{CS}_3</math>)</td><td>DRAM空間 (<math>\overline{CS}_2</math>)</td></tr> <tr> <td>1</td><td>通常</td><td>通常</td><td colspan="2">DRAM空間 (<math>\overline{CS}_2</math>) *</td></tr> <tr> <td rowspan="4">1</td><td rowspan="2">0</td><td>0</td><td>通常</td><td>DRAM空間 (<math>\overline{CS}_4</math>)</td><td>DRAM空間 (<math>\overline{CS}_3</math>)</td><td>DRAM空間 (<math>\overline{CS}_2</math>)</td></tr> <tr> <td>1</td><td>DRAM空間 (<math>\overline{CS}_5</math>)</td><td>DRAM空間 (<math>\overline{CS}_4</math>)</td><td>DRAM空間 (<math>\overline{CS}_3</math>)</td><td>DRAM空間 (<math>\overline{CS}_2</math>)</td></tr> <tr> <td rowspan="2">1</td><td>0</td><td colspan="2">DRAM空間 (<math>\overline{CS}_4</math>) *</td><td colspan="2">DRAM空間 (<math>\overline{CS}_2</math>) *</td></tr> <tr> <td>1</td><td colspan="4">DRAM空間 (<math>\overline{CS}_2</math>) *</td></tr> </tbody> </table>			DRAS2	DRAS1	DRAS0	エリア5	エリア4	エリア3	エリア2	0	0	0	通常	通常	通常	通常	1	通常	通常	通常	DRAM空間 ( $\overline{CS}_2$ )	1	0	通常	通常	DRAM空間 ( $\overline{CS}_3$ )	DRAM空間 ( $\overline{CS}_2$ )	1	通常	通常	DRAM空間 ( $\overline{CS}_2$ ) *		1	0	0	通常	DRAM空間 ( $\overline{CS}_4$ )	DRAM空間 ( $\overline{CS}_3$ )	DRAM空間 ( $\overline{CS}_2$ )	1	DRAM空間 ( $\overline{CS}_5$ )	DRAM空間 ( $\overline{CS}_4$ )	DRAM空間 ( $\overline{CS}_3$ )	DRAM空間 ( $\overline{CS}_2$ )	1	0	DRAM空間 ( $\overline{CS}_4$ ) *		DRAM空間 ( $\overline{CS}_2$ ) *		1	DRAM空間 ( $\overline{CS}_2$ ) *			
DRAS2	DRAS1	DRAS0	エリア5	エリア4	エリア3	エリア2																																																	
0	0	0	通常	通常	通常	通常																																																	
		1	通常	通常	通常	DRAM空間 ( $\overline{CS}_2$ )																																																	
	1	0	通常	通常	DRAM空間 ( $\overline{CS}_3$ )	DRAM空間 ( $\overline{CS}_2$ )																																																	
		1	通常	通常	DRAM空間 ( $\overline{CS}_2$ ) *																																																		
1	0	0	通常	DRAM空間 ( $\overline{CS}_4$ )	DRAM空間 ( $\overline{CS}_3$ )	DRAM空間 ( $\overline{CS}_2$ )																																																	
		1	DRAM空間 ( $\overline{CS}_5$ )	DRAM空間 ( $\overline{CS}_4$ )	DRAM空間 ( $\overline{CS}_3$ )	DRAM空間 ( $\overline{CS}_2$ )																																																	
	1	0	DRAM空間 ( $\overline{CS}_4$ ) *		DRAM空間 ( $\overline{CS}_2$ ) *																																																		
		1	DRAM空間 ( $\overline{CS}_2$ ) *																																																				
<p>【注】* 1本の<math>\overline{CS}_n</math>端子が複数エリアに共通の<math>\overline{RAS}</math>出力端子となります。このとき不要となる<math>\overline{CS}_n</math>端子は入出力ポートとして使用可能になります。</p>																																																							



DRCRB DRAM コントロールレジスタ B	H'EE027	DRAM インタフェース													
<p>ビット:      7      6      5      4      3      2      1      0</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">MXC1</td> <td style="width: 5%;">MXC0</td> <td style="width: 5%;">CSEL</td> <td style="width: 5%;">RCYCE</td> <td style="width: 5%;">—</td> <td style="width: 5%;">TPC</td> <td style="width: 5%;">RCW</td> <td style="width: 5%;">RLW</td> </tr> </table> <p>初期値:      0      0      0      0      1      0      0      0</p> <p>R/W :      R/W    R/W    R/W    R/W    —    R/W    R/W    R/W</p>			MXC1	MXC0	CSEL	RCYCE	—	TPC	RCW	RLW					
MXC1	MXC0	CSEL	RCYCE	—	TPC	RCW	RLW								
<p>リフレッシュサイクルウェイトコントロール</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td> <td>ウェイトステート (<math>T_{RW}</math>) の挿入を禁止</td> </tr> <tr> <td>1</td> <td>ウェイトステート (<math>T_{RW}</math>) を1ステート挿入</td> </tr> </table>			0	ウェイトステート ( $T_{RW}$ ) の挿入を禁止	1	ウェイトステート ( $T_{RW}$ ) を1ステート挿入									
0	ウェイトステート ( $T_{RW}$ ) の挿入を禁止														
1	ウェイトステート ( $T_{RW}$ ) を1ステート挿入														
<p>RAS・CAS間ウェイト</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td> <td>ウェイトステート (<math>T_{rW}</math>) の挿入を禁止</td> </tr> <tr> <td>1</td> <td>ウェイトステート (<math>T_{rW}</math>) を1ステート挿入</td> </tr> </table>			0	ウェイトステート ( $T_{rW}$ ) の挿入を禁止	1	ウェイトステート ( $T_{rW}$ ) を1ステート挿入									
0	ウェイトステート ( $T_{rW}$ ) の挿入を禁止														
1	ウェイトステート ( $T_{rW}$ ) を1ステート挿入														
<p>TPサイクルコントロール</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td> <td>プリチャージサイクルを1ステート挿入</td> </tr> <tr> <td>1</td> <td>プリチャージサイクルを2ステート挿入</td> </tr> </table>			0	プリチャージサイクルを1ステート挿入	1	プリチャージサイクルを2ステート挿入									
0	プリチャージサイクルを1ステート挿入														
1	プリチャージサイクルを2ステート挿入														
<p>リフレッシュサイクルイネーブル</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td> <td>リフレッシュサイクルを禁止</td> </tr> <tr> <td>1</td> <td>DRAMに対するリフレッシュサイクルを許可</td> </tr> </table>			0	リフレッシュサイクルを禁止	1	DRAMに対するリフレッシュサイクルを許可									
0	リフレッシュサイクルを禁止														
1	DRAMに対するリフレッシュサイクルを許可														
<p>CAS出力端子選択</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 5%;">0</td> <td>UCAS、LCASの出力端子: PB4、PB5を選択</td> </tr> <tr> <td>1</td> <td>UCAS、LCASの出力端子: HWR、LWRを選択</td> </tr> </table>			0	UCAS、LCASの出力端子: PB4、PB5を選択	1	UCAS、LCASの出力端子: HWR、LWRを選択									
0	UCAS、LCASの出力端子: PB4、PB5を選択														
1	UCAS、LCASの出力端子: HWR、LWRを選択														
<p>マルチバンクコントロール1、0</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="width: 5%;">MXC1</th> <th style="width: 5%;">MXC0</th> <th style="width: 90%;">説明</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>                     カラムアドレス: 8ビット                      比較対象アドレス:                          モード1、2      8ビットアクセス空間    <math>A_{19} \sim A_8</math>                                                16ビットアクセス空間    <math>A_{19} \sim A_9</math>                          モード3、4、5    8ビットアクセス空間    <math>A_{23} \sim A_8</math>                                                16ビットアクセス空間    <math>A_{23} \sim A_9</math> </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                     カラムアドレス: 9ビット                      比較対象アドレス:                          モード1、2      8ビットアクセス空間    <math>A_{19} \sim A_9</math>                                                16ビットアクセス空間    <math>A_{19} \sim A_{10}</math>                          モード3、4、5    8ビットアクセス空間    <math>A_{23} \sim A_9</math>                                                16ビットアクセス空間    <math>A_{23} \sim A_{10}</math> </td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>                     カラムアドレス: 10ビット                      比較対象アドレス:                          モード1、2      8ビットアクセス空間    <math>A_{19} \sim A_{10}</math>                                                16ビットアクセス空間    <math>A_{19} \sim A_{11}</math>                          モード3、4、5    8ビットアクセス空間    <math>A_{23} \sim A_{10}</math>                                                16ビットアクセス空間    <math>A_{23} \sim A_{11}</math> </td> </tr> <tr> <td style="text-align: center;">1</td> <td>設定禁止</td> </tr> </tbody> </table>			MXC1	MXC0	説明	0	0	カラムアドレス: 8ビット 比較対象アドレス: モード1、2      8ビットアクセス空間 $A_{19} \sim A_8$ 16ビットアクセス空間 $A_{19} \sim A_9$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_8$ 16ビットアクセス空間 $A_{23} \sim A_9$	1	カラムアドレス: 9ビット 比較対象アドレス: モード1、2      8ビットアクセス空間 $A_{19} \sim A_9$ 16ビットアクセス空間 $A_{19} \sim A_{10}$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_9$ 16ビットアクセス空間 $A_{23} \sim A_{10}$	1	0	カラムアドレス: 10ビット 比較対象アドレス: モード1、2      8ビットアクセス空間 $A_{19} \sim A_{10}$ 16ビットアクセス空間 $A_{19} \sim A_{11}$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_{10}$ 16ビットアクセス空間 $A_{23} \sim A_{11}$	1	設定禁止
MXC1	MXC0	説明													
0	0	カラムアドレス: 8ビット 比較対象アドレス: モード1、2      8ビットアクセス空間 $A_{19} \sim A_8$ 16ビットアクセス空間 $A_{19} \sim A_9$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_8$ 16ビットアクセス空間 $A_{23} \sim A_9$													
	1	カラムアドレス: 9ビット 比較対象アドレス: モード1、2      8ビットアクセス空間 $A_{19} \sim A_9$ 16ビットアクセス空間 $A_{19} \sim A_{10}$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_9$ 16ビットアクセス空間 $A_{23} \sim A_{10}$													
1	0	カラムアドレス: 10ビット 比較対象アドレス: モード1、2      8ビットアクセス空間 $A_{19} \sim A_{10}$ 16ビットアクセス空間 $A_{19} \sim A_{11}$ モード3、4、5    8ビットアクセス空間 $A_{23} \sim A_{10}$ 16ビットアクセス空間 $A_{23} \sim A_{11}$													
	1	設定禁止													

RTMCSR リフレッシュタイムコントロール/ステータス レジスタ B	H'EE028	DRAM インタフェース																										
ビット： 7 6 5 4 3 2 1 0 <table border="1"> <tr> <td>CMF</td> <td>CMIE</td> <td>CKS2</td> <td>CKS1</td> <td>CKS0</td> <td>—</td> <td>—</td> <td>—</td> </tr> </table> 初期値： 0 0 0 0 0 1 1 1 R/W： R(W)* R/W R/W R/W R/W — — —	CMF	CMIE	CKS2	CKS1	CKS0	—	—	—																				
CMF	CMIE	CKS2	CKS1	CKS0	—	—	—																					
リフレッシュカウンタクロックセレクト																												
<table border="1"> <thead> <tr> <th>CKS2</th> <th>CKS1</th> <th>CKS0</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td rowspan="4">0</td> <td rowspan="2">0</td> <td>0</td> <td>カウント動作停止</td> </tr> <tr> <td>1</td> <td>/2でカウント</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>/8でカウント</td> </tr> <tr> <td>1</td> <td>/32でカウント</td> </tr> <tr> <td rowspan="4">1</td> <td rowspan="2">0</td> <td>0</td> <td>/128でカウント</td> </tr> <tr> <td>1</td> <td>/512でカウント</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>/2048でカウント</td> </tr> <tr> <td>1</td> <td>/4096でカウント</td> </tr> </tbody> </table>			CKS2	CKS1	CKS0	説明	0	0	0	カウント動作停止	1	/2でカウント	1	0	/8でカウント	1	/32でカウント	1	0	0	/128でカウント	1	/512でカウント	1	0	/2048でカウント	1	/4096でカウント
CKS2	CKS1	CKS0	説明																									
0	0	0	カウント動作停止																									
		1	/2でカウント																									
	1	0	/8でカウント																									
		1	/32でカウント																									
1	0	0	/128でカウント																									
		1	/512でカウント																									
	1	0	/2048でカウント																									
		1	/4096でカウント																									
コンペアマッチ割込みイネーブル																												
<table border="1"> <tr> <td>0</td> <td>CMFフラグによる割込み要求 (CMI) を禁止</td> </tr> <tr> <td>1</td> <td>CMFフラグによる割込み要求 (CMI) を許可</td> </tr> </table>			0	CMFフラグによる割込み要求 (CMI) を禁止	1	CMFフラグによる割込み要求 (CMI) を許可																						
0	CMFフラグによる割込み要求 (CMI) を禁止																											
1	CMFフラグによる割込み要求 (CMI) を許可																											
コンペアマッチフラグ																												
<table border="1"> <tr> <td>0</td> <td>           [クリア条件]            (1) リセットまたはスタンバイモード時            (2) CMF = 1の状態、CMFフラグをリードした後、CMFフラグに0をライトしたとき         </td> </tr> <tr> <td>1</td> <td>           [セット条件]            RTCNT = RTCORになったとき         </td> </tr> </table>			0	[クリア条件] (1) リセットまたはスタンバイモード時 (2) CMF = 1の状態、CMFフラグをリードした後、CMFフラグに0をライトしたとき	1	[セット条件] RTCNT = RTCORになったとき																						
0	[クリア条件] (1) リセットまたはスタンバイモード時 (2) CMF = 1の状態、CMFフラグをリードした後、CMFフラグに0をライトしたとき																											
1	[セット条件] RTCNT = RTCORになったとき																											
【注】* フラグをクリアするための0ライトのみ可能です。																												

RTCNT リフレッシュタイムカウンタ	H'EE029	DRAM インタフェース								
<p>ビット：      7      6      5      4      3      2      1      0</p> <table border="1" style="margin: auto;"> <tr> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> </table> <p>初期値：      0      0      0      0      0      0      0      0</p> <p>R/W    :      R/W    R/W    R/W    R/W    R/W    R/W    R/W    R/W</p> <hr style="width: 50%; margin: 10px auto;"/> <p style="text-align: center;">RTMCSRのCKS2～CKS0ビットで 選択された内部クロックにより、カウントアップ</p>										
RTCOR リフレッシュタイムコンスタントレジスタ	H'EE02A	DRAM インタフェース								
<p>ビット：      7      6      5      4      3      2      1      0</p> <table border="1" style="margin: auto;"> <tr> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> </table> <p>初期値：      1      1      1      1      1      1      1      1</p> <p>R/W    :      R/W    R/W    R/W    R/W    R/W    R/W    R/W    R/W</p> <hr style="width: 50%; margin: 10px auto;"/> <p style="text-align: center;">RTCNTとのコンペアマッチ周期を設定</p> <p style="text-align: center; margin-top: 20px;"><b>【注】</b> 本レジスタへのアクセスはかならずバイトアクセスとしてください。</p>										

FLMCR1 フラッシュメモリコントロールレジスタ 1		H'EE030		フラッシュメモリ					
ビット:		7	6	5	4	3	2	1	0
		FWE	SWE	ESU	PSU	EV	PV	E	P
モード1~4、6	初期値:	0	0	0	0	0	0	0	0
	R/W:	R	R	R	R	R	R	R	R
モード5、7	初期値:	1/0	0	0	0	0	0	0	0
	R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
									プログラムモード
									0 プログラムモードを解除 (初期値)
									1 プログラムモードに遷移 [セットするための条件] FWE = 1、SWE = 1、PSU = 1のとき
									イレースモード
									0 イレースモードを解除 (初期値)
									1 イレースモードに遷移 [セットするための条件] FWE = 1、SWE = 1、ESU = 1のとき
									プログラムベリファイモード
									0 プログラムベリファイモードを解除 (初期値)
									1 プログラムベリファイモードに遷移 [セットするための条件] FWE = 1、SWE = 1のとき
									イレースベリファイモード
									0 イレースベリファイモードを解除 (初期値)
									1 イレースベリファイモードに遷移 [セットするための条件] FWE = 1、SWE = 1のとき
									プログラムセットアップ
									0 プログラムセットアップ解除 (初期値)
									1 プログラムセットアップ [セットするための条件] FWE = 1、SWE = 1のとき
									イレースセットアップ
									0 イレースセットアップ解除 (初期値)
									1 イレースセットアップ [セットするための条件] FWE = 1、SWE = 1のとき
									ソフトウェアライトイネーブルビット
									0 書き込み / 消去無効 (初期値)
									1 書き込み / 消去有効 [セットするための条件] FWE = 1のとき
									フラッシュライトイネーブルビット
									0 FWE端子にLowレベルが入力されているとき (ハードウェアプロテクト状態)
									1 FWE端子にHighレベルが入力されているとき

【注】 1. 本レジスタはフラッシュメモリ版およびフラッシュメモリR版専用のレジスタです。マスクROM版には存在しませんので、当該アドレスをリードすると常に1が読み出されます。ライトは無効です。  
2. モード6では、FWE端子はLowレベル固定してください。

FLMCR2 フラッシュメモリコントロールレジスタ2	H'EE031	フラッシュメモリ									
<p>ビット:      7      6      5      4      3      2      1      0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%; text-align: center;">FLER</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> </table> <p>初期値:      0      0      0      0      0      0      0      0</p> <p>R/W:        R</p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="text-align: center;"> <p>↑</p> <p>フラッシュメモリエラー</p> </div> <div style="text-align: center;"> <p>↑</p> <p>リザーブビット</p> </div> </div> <p style="text-align: right; margin-top: 20px;">【注】 FLMCR2へのライトは禁止されています。</p>				FLER							
	FLER										

EBR (EBR1) 消去ブロック指定レジスタ	H'EE032	フラッシュメモリ													
<p>ビット:      7      6      5      4      3      2      1      0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%; text-align: center;">EB7</td> <td style="width: 10%; text-align: center;">EB6</td> <td style="width: 10%; text-align: center;">EB5</td> <td style="width: 10%; text-align: center;">EB4</td> <td style="width: 10%; text-align: center;">EB3</td> <td style="width: 10%; text-align: center;">EB2</td> <td style="width: 10%; text-align: center;">EB1</td> <td style="width: 10%; text-align: center;">EB0</td> </tr> </table> <p>モード1~4、6 初期値:    0      0      0      0      0      0      0      0</p> <p>                 R/W:      R      R      R      R      R      R      R      R</p> <p>モード5、7 初期値:    0      0      0      0      0      0      0      0</p> <p>                 R/W:    R/W R/W R/W R/W R/W R/W R/W R/W</p> <div style="text-align: center; margin-top: 10px;"> <p>↑</p> <p>ブロック7~0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: center;">0</td> <td>EB7~EB0ブロックを選択していないとき (初期値)</td> </tr> <tr> <td style="text-align: center;">1</td> <td>EB7~EB0ブロックを選択しているとき</td> </tr> </table> </div> <p style="text-align: right; margin-top: 20px;">【注】 EBRのビットは、消去する時以外はH'00にしてください。 ライトは無効です。 モード6のときには、本レジスタに1をセットすることはできません。</p>				EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0	0	EB7~EB0ブロックを選択していないとき (初期値)	1	EB7~EB0ブロックを選択しているとき
	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0							
0	EB7~EB0ブロックを選択していないとき (初期値)														
1	EB7~EB0ブロックを選択しているとき														

EBR2 消去ブロック指定レジスタ 2		H'EE033		フラッシュメモリ					
ビット:		7	6	5	4	3	2	1	0
				EB13	EB12	EB11	EB10	EB9	EB8
モード1~4、6	初期値:	0	0	0	0	0	0	0	0
	R/W:	R	R	R	R	R	R	R	R
モード5、7	初期値:	0	0	0	0	0	0	0	0
	R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
↓ ブロック13~8									
		0	EB13~EB8ブロックを選択していないとき(初期値)						
		1	EB13~EB8ブロックを選択しているとき						
【注】 EBRのビットは、消去する時以外はH'00にしてください。 モード6のときには、本レジスタに1をセットすることはできません。									
P2PCR ポート2 入力プルアップMOSコントロールレジスタ		H'EE03C		ポート 2					
ビット:		7	6	5	4	3	2	1	0
		P27PCR	P26PCR	P25PCR	P24PCR	P23PCR	P22PCR	P21PCR	P20PCR
初期値:		0	0	0	0	0	0	0	0
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
↓ ポート2入力プルアップMOSコントロール7~0									
		0	入力プルアップMOSはOFF						
		1	入力プルアップMOSはON						
P2DDRを0に指定したとき(入力ポートに指定)									

P4PCR ポート4入力プルアップMOSコントロールレジスタ	H'EE03E	ポート4																				
<p>ビット:</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">7</td> <td style="padding: 5px;">6</td> <td style="padding: 5px;">5</td> <td style="padding: 5px;">4</td> <td style="padding: 5px;">3</td> <td style="padding: 5px;">2</td> <td style="padding: 5px;">1</td> <td style="padding: 5px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">P47PCR</td> <td style="border: 1px solid black; padding: 2px;">P46PCR</td> <td style="border: 1px solid black; padding: 2px;">P45PCR</td> <td style="border: 1px solid black; padding: 2px;">P44PCR</td> <td style="border: 1px solid black; padding: 2px;">P43PCR</td> <td style="border: 1px solid black; padding: 2px;">P42PCR</td> <td style="border: 1px solid black; padding: 2px;">P41PCR</td> <td style="border: 1px solid black; padding: 2px;">P40PCR</td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W</p> <div style="text-align: center; margin: 10px 0;"> <p>ポート4入力プルアップMOSコントロール7~0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">0</td> <td style="padding: 2px 10px;">入力プルアップMOSはOFF</td> </tr> <tr> <td style="padding: 2px 10px;">1</td> <td style="padding: 2px 10px;">入力プルアップMOSはON</td> </tr> </table> <p>P4DDRを0に指定したとき（入力ポートに指定）</p> </div>			7	6	5	4	3	2	1	0	P47PCR	P46PCR	P45PCR	P44PCR	P43PCR	P42PCR	P41PCR	P40PCR	0	入力プルアップMOSはOFF	1	入力プルアップMOSはON
7	6	5	4	3	2	1	0															
P47PCR	P46PCR	P45PCR	P44PCR	P43PCR	P42PCR	P41PCR	P40PCR															
0	入力プルアップMOSはOFF																					
1	入力プルアップMOSはON																					
P5PCR ポート5入力プルアップMOSコントロールレジスタ	H'EE03F	ポート5																				
<p>ビット:</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">7</td> <td style="padding: 5px;">6</td> <td style="padding: 5px;">5</td> <td style="padding: 5px;">4</td> <td style="padding: 5px;">3</td> <td style="padding: 5px;">2</td> <td style="padding: 5px;">1</td> <td style="padding: 5px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">—</td> <td style="border: 1px solid black; padding: 2px;">—</td> <td style="border: 1px solid black; padding: 2px;">—</td> <td style="border: 1px solid black; padding: 2px;">—</td> <td style="border: 1px solid black; padding: 2px;">P53PCR</td> <td style="border: 1px solid black; padding: 2px;">P52PCR</td> <td style="border: 1px solid black; padding: 2px;">P51PCR</td> <td style="border: 1px solid black; padding: 2px;">P50PCR</td> </tr> </table> <p>初期値: 1 1 1 1 0 0 0 0</p> <p>R/W: — — — — R/W R/W R/W R/W</p> <div style="text-align: center; margin: 10px 0;"> <p>ポート5入力プルアップMOSコントロール3~0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">0</td> <td style="padding: 2px 10px;">入力プルアップMOSはOFF</td> </tr> <tr> <td style="padding: 2px 10px;">1</td> <td style="padding: 2px 10px;">入力プルアップMOSはON</td> </tr> </table> <p>P5DDRを0に指定したとき（入力ポートに指定）</p> </div>			7	6	5	4	3	2	1	0	—	—	—	—	P53PCR	P52PCR	P51PCR	P50PCR	0	入力プルアップMOSはOFF	1	入力プルアップMOSはON
7	6	5	4	3	2	1	0															
—	—	—	—	P53PCR	P52PCR	P51PCR	P50PCR															
0	入力プルアップMOSはOFF																					
1	入力プルアップMOSはON																					

RAMCR RAMコントロールレジスタ	H'EE077	フラッシュメモリ																																											
ビット:      7          6          5          4          3          2          1          0 <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 20px;"></td> <td style="width: 20px;"></td> <td style="width: 20px;"></td> <td style="width: 20px;"></td> <td style="width: 20px; text-align: center;">RAMS</td> <td style="width: 20px; text-align: center;">RAM2</td> <td style="width: 20px; text-align: center;">RAM1</td> <td style="width: 20px; text-align: center;">RAM0</td> </tr> </table>					RAMS	RAM2	RAM1	RAM0																																					
				RAMS	RAM2	RAM1	RAM0																																						
モード1~4 { <table style="display: inline-table; vertical-align: middle;"> <tr> <td style="padding-right: 5px;">初期値:</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">0</td> <td style="padding-right: 10px;">0</td> <td style="padding-right: 10px;">0</td> <td style="padding-right: 10px;">0</td> </tr> <tr> <td>R/W:</td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">R</td> <td style="text-align: center;">R</td> <td style="text-align: center;">R</td> <td></td> </tr> </table>	初期値:	1	1	1	1	0	0	0	0	R/W:					R	R	R																												
初期値:	1	1	1	1	0	0	0	0																																					
R/W:					R	R	R																																						
モード5~7 { <table style="display: inline-table; vertical-align: middle;"> <tr> <td style="padding-right: 5px;">初期値:</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">1</td> <td style="padding-right: 10px;">0</td> <td style="padding-right: 10px;">0</td> <td style="padding-right: 10px;">0</td> <td style="padding-right: 10px;">0</td> </tr> <tr> <td>R/W:</td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">R/W*</td> <td style="text-align: center;">R/W*</td> <td style="text-align: center;">R/W*</td> <td style="text-align: center;">R/W*</td> </tr> </table>	初期値:	1	1	1	1	0	0	0	0	R/W:					R/W*	R/W*	R/W*	R/W*																											
初期値:	1	1	1	1	0	0	0	0																																					
R/W:					R/W*	R/W*	R/W*	R/W*																																					
<div style="border-top: 1px solid black; width: 30%; margin: 0 auto;"></div> <div style="margin-left: 100px;">└───┬───┘</div> <div style="margin-left: 100px;">リザーブビット</div>																																													
RAMセレクト、RAM2~1																																													
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;">ビット3</th> <th style="width: 10%;">ビット2</th> <th style="width: 10%;">ビット1</th> <th style="width: 10%;">ビット0</th> <th style="width: 30%;">RAMエリア</th> <th style="width: 20%;">RAMエミュレーション状態</th> </tr> <tr> <th>RAMS</th> <th>RAM2</th> <th>RAM1</th> <th>RAM0</th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0/1</td> <td>0/1</td> <td>0/1</td> <td>H'FFE000 ~ H'FFFEFFF</td> <td>エミュレーション</td> </tr> <tr> <td rowspan="8">1</td> <td rowspan="4">0</td> <td rowspan="2">0</td> <td>0</td> <td>H'000000 ~ H'000FFF</td> <td rowspan="8">写像RAM</td> </tr> <tr> <td>1</td> <td>H'001000 ~ H'001FFF</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>H'002000 ~ H'002FFF</td> </tr> <tr> <td>1</td> <td>H'003000 ~ H'003FFF</td> </tr> <tr> <td rowspan="4">1</td> <td rowspan="2">0</td> <td>0</td> <td>H'004000 ~ H'004FFF</td> </tr> <tr> <td>1</td> <td>H'005000 ~ H'005FFF</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>H'006000 ~ H'006FFF</td> </tr> <tr> <td>1</td> <td>H'007000 ~ H'007FFF</td> </tr> </tbody> </table>				ビット3	ビット2	ビット1	ビット0	RAMエリア	RAMエミュレーション状態	RAMS	RAM2	RAM1	RAM0			0	0/1	0/1	0/1	H'FFE000 ~ H'FFFEFFF	エミュレーション	1	0	0	0	H'000000 ~ H'000FFF	写像RAM	1	H'001000 ~ H'001FFF	1	0	H'002000 ~ H'002FFF	1	H'003000 ~ H'003FFF	1	0	0	H'004000 ~ H'004FFF	1	H'005000 ~ H'005FFF	1	0	H'006000 ~ H'006FFF	1	H'007000 ~ H'007FFF
ビット3	ビット2	ビット1	ビット0	RAMエリア	RAMエミュレーション状態																																								
RAMS	RAM2	RAM1	RAM0																																										
0	0/1	0/1	0/1	H'FFE000 ~ H'FFFEFFF	エミュレーション																																								
1	0	0	0	H'000000 ~ H'000FFF	写像RAM																																								
			1	H'001000 ~ H'001FFF																																									
		1	0	H'002000 ~ H'002FFF																																									
			1	H'003000 ~ H'003FFF																																									
	1	0	0	H'004000 ~ H'004FFF																																									
			1	H'005000 ~ H'005FFF																																									
		1	0	H'006000 ~ H'006FFF																																									
			1	H'007000 ~ H'007FFF																																									
<p>【注】* モード6（シングルチップノーマルモード）では、RAMによるフラッシュメモリのエミュレーションはサポートしていませんので、ライトは可能ですが、1をセットしないでください。</p> <p>本レジスタはフラッシュメモリ版およびフラッシュメモリR版専用のレジスタです。マスクROM版には存在しませんので、当該アドレスをリードすると常に1が読み出されます。ライトは無効です。</p>																																													





ETCR0A	H、L	転送カウンタレジスタ0A	H、L	H'FFF24 H'FFF25	DMAC0
(前頁より続く)					
フルアドレスモード					
(1) ノーマルモード					
ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 					
初期値: 不定 R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W 転送カウンタ					
(2) ブロック転送モード					
ビット: 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0 					
初期値: 不定 不定 R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W ETCR0AH ETCR0AL ブロックサイズカウンタ ブロックサイズ保持					

IOAR0A I/O アドレスレジスタ 0A	H'FFF26	DMAC0								
<div style="display: flex; justify-content: space-around; margin-bottom: 5px;"> <span>ビット:</span> <span>7</span> <span>6</span> <span>5</span> <span>4</span> <span>3</span> <span>2</span> <span>1</span> <span>0</span> </div> <table style="width: 100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> <td style="width: 12.5%; border: 1px solid black; height: 20px;"></td> </tr> </table> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> <span>初期値:</span> <span style="margin-left: 100px;">不定</span> </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> <span>R/W:</span> <span>R/W</span> <span>R/W</span> <span>R/W</span> <span>R/W</span> <span>R/W</span> <span>R/W</span> <span>R/W</span> </div> <hr style="width: 80%; margin: 0 auto;"/> <div style="text-align: center; margin-bottom: 5px;"> <span style="display: inline-block; width: 10px; height: 10px; border: 1px solid black; margin-bottom: 2px;"></span>           ショートアドレスモード：ソースアドレスまたはデスティネーションアドレスを設定         </div> <div style="text-align: center;"> <span style="display: inline-block; width: 10px; height: 10px; border: 1px solid black; margin-bottom: 2px;"></span>           フルアドレスモード      : 未使用         </div>										

DTCR0A データトランスファコントロールレジスタ 0A	H'FFF27	DMAC0																																																																																																																				
<p><b>ショートアドレスモード</b></p> <p>ビット:           7           6           5           4           3           2           1           0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 25px; text-align: center;">DTE</td> <td style="width: 25px; text-align: center;">DTSZ</td> <td style="width: 25px; text-align: center;">DTID</td> <td style="width: 25px; text-align: center;">RPE</td> <td style="width: 25px; text-align: center;">DTIE</td> <td style="width: 25px; text-align: center;">DTS2</td> <td style="width: 25px; text-align: center;">DTS1</td> <td style="width: 25px; text-align: center;">DTS0</td> </tr> </table> <p>初期値:           0           0           0           0           0           0           0           0</p> <p>R/W:             R/W       R/W       R/W       R/W       R/W       R/W       R/W</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p style="margin-left: 20px;">データトランスファセレクト</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット2 DTS2</th> <th style="width: 10%;">ビット1 DTS1</th> <th style="width: 10%;">ビット0 DTS0</th> <th style="width: 70%;">データ転送の起動要因</th> </tr> </thead> <tbody> <tr> <td rowspan="4" style="text-align: center; vertical-align: middle;">0</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">0</td> <td style="text-align: center;">0</td> <td>16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>A/D変換器の変換終了割込みで起動</td> </tr> <tr> <td rowspan="4" style="text-align: center; vertical-align: middle;">1</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">0</td> <td style="text-align: center;">0</td> <td>SCI0の送信データエンプティ割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>SCI0の受信データフル割込みで起動</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>フルアドレスモード転送を指定</td> </tr> <tr> <td style="text-align: center;">1</td> <td>フルアドレスモード転送を指定</td> </tr> </tbody> </table>   <p style="margin-left: 20px;">データトランスファインタラプトイネーブル</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>DTEビットによる割込み要求を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>DTEビットによる割込み要求を許可</td> </tr> </table>   <p style="margin-left: 20px;">リピートイネーブル</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <thead> <tr> <th style="width: 20px;">RPE</th> <th style="width: 20px;">DTIE</th> <th style="width: 60px;">説 明</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td rowspan="2">I/Oモードで転送</td> </tr> <tr> <td style="text-align: center;">1</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>リピートモードで転送</td> </tr> <tr> <td style="text-align: center;">1</td> <td>アイドルモードで転送</td> </tr> </tbody> </table>   <p style="margin-left: 20px;">データトランスファインクリメント/デクリメント</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2</td> </tr> <tr> <td style="text-align: center;">1</td> <td>デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2</td> </tr> </table>   <p style="margin-left: 20px;">データトランスファサイズ</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>バイトサイズ転送</td> </tr> <tr> <td style="text-align: center;">1</td> <td>ワードサイズ転送</td> </tr> </table>   <p style="margin-left: 20px;">データトランスファイネーブル</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>データ転送を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>データ転送を許可</td> </tr> </table> </div> <div style="width: 50%; padding-left: 20px;"> <p>データトランスファセレクト</p> <table border="1" style="border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット2 DTS2</th> <th style="width: 10%;">ビット1 DTS1</th> <th style="width: 10%;">ビット0 DTS0</th> <th style="width: 70%;">データ転送の起動要因</th> </tr> </thead> <tbody> <tr> <td rowspan="4" style="text-align: center; vertical-align: middle;">0</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">0</td> <td style="text-align: center;">0</td> <td>16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>A/D変換器の変換終了割込みで起動</td> </tr> <tr> <td rowspan="4" style="text-align: center; vertical-align: middle;">1</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">0</td> <td style="text-align: center;">0</td> <td>SCI0の送信データエンプティ割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>SCI0の受信データフル割込みで起動</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>フルアドレスモード転送を指定</td> </tr> <tr> <td style="text-align: center;">1</td> <td>フルアドレスモード転送を指定</td> </tr> </tbody> </table>   <p>データトランスファインタラプトイネーブル</p> <table border="1" style="border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>DTEビットによる割込み要求を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>DTEビットによる割込み要求を許可</td> </tr> </table>   <p>リピートイネーブル</p> <table border="1" style="border-collapse: collapse;"> <thead> <tr> <th style="width: 20px;">RPE</th> <th style="width: 20px;">DTIE</th> <th style="width: 60px;">説 明</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td rowspan="2">I/Oモードで転送</td> </tr> <tr> <td style="text-align: center;">1</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>リピートモードで転送</td> </tr> <tr> <td style="text-align: center;">1</td> <td>アイドルモードで転送</td> </tr> </tbody> </table>   <p>データトランスファインクリメント/デクリメント</p> <table border="1" style="border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2</td> </tr> <tr> <td style="text-align: center;">1</td> <td>デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2</td> </tr> </table>   <p>データトランスファサイズ</p> <table border="1" style="border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>バイトサイズ転送</td> </tr> <tr> <td style="text-align: center;">1</td> <td>ワードサイズ転送</td> </tr> </table>   <p>データトランスファイネーブル</p> <table border="1" style="border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>データ転送を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>データ転送を許可</td> </tr> </table> </div> </div>			DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0	ビット2 DTS2	ビット1 DTS1	ビット0 DTS0	データ転送の起動要因	0	0	0	16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動	1	16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動	1	0	16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動	1	A/D変換器の変換終了割込みで起動	1	0	0	SCI0の送信データエンプティ割込みで起動	1	SCI0の受信データフル割込みで起動	1	0	フルアドレスモード転送を指定	1	フルアドレスモード転送を指定	0	DTEビットによる割込み要求を禁止	1	DTEビットによる割込み要求を許可	RPE	DTIE	説 明	0	0	I/Oモードで転送	1	1	0	リピートモードで転送	1	アイドルモードで転送	0	インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2	1	デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2	0	バイトサイズ転送	1	ワードサイズ転送	0	データ転送を禁止	1	データ転送を許可	ビット2 DTS2	ビット1 DTS1	ビット0 DTS0	データ転送の起動要因	0	0	0	16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動	1	16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動	1	0	16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動	1	A/D変換器の変換終了割込みで起動	1	0	0	SCI0の送信データエンプティ割込みで起動	1	SCI0の受信データフル割込みで起動	1	0	フルアドレスモード転送を指定	1	フルアドレスモード転送を指定	0	DTEビットによる割込み要求を禁止	1	DTEビットによる割込み要求を許可	RPE	DTIE	説 明	0	0	I/Oモードで転送	1	1	0	リピートモードで転送	1	アイドルモードで転送	0	インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2	1	デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2	0	バイトサイズ転送	1	ワードサイズ転送	0	データ転送を禁止	1	データ転送を許可
DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0																																																																																																															
ビット2 DTS2	ビット1 DTS1	ビット0 DTS0	データ転送の起動要因																																																																																																																			
0	0	0	16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動																																																																																																																			
		1	16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動																																																																																																																			
	1	0	16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動																																																																																																																			
		1	A/D変換器の変換終了割込みで起動																																																																																																																			
1	0	0	SCI0の送信データエンプティ割込みで起動																																																																																																																			
		1	SCI0の受信データフル割込みで起動																																																																																																																			
	1	0	フルアドレスモード転送を指定																																																																																																																			
		1	フルアドレスモード転送を指定																																																																																																																			
0	DTEビットによる割込み要求を禁止																																																																																																																					
1	DTEビットによる割込み要求を許可																																																																																																																					
RPE	DTIE	説 明																																																																																																																				
0	0	I/Oモードで転送																																																																																																																				
	1																																																																																																																					
1	0	リピートモードで転送																																																																																																																				
	1	アイドルモードで転送																																																																																																																				
0	インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2																																																																																																																					
1	デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2																																																																																																																					
0	バイトサイズ転送																																																																																																																					
1	ワードサイズ転送																																																																																																																					
0	データ転送を禁止																																																																																																																					
1	データ転送を許可																																																																																																																					
ビット2 DTS2	ビット1 DTS1	ビット0 DTS0	データ転送の起動要因																																																																																																																			
0	0	0	16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動																																																																																																																			
		1	16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動																																																																																																																			
	1	0	16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動																																																																																																																			
		1	A/D変換器の変換終了割込みで起動																																																																																																																			
1	0	0	SCI0の送信データエンプティ割込みで起動																																																																																																																			
		1	SCI0の受信データフル割込みで起動																																																																																																																			
	1	0	フルアドレスモード転送を指定																																																																																																																			
		1	フルアドレスモード転送を指定																																																																																																																			
0	DTEビットによる割込み要求を禁止																																																																																																																					
1	DTEビットによる割込み要求を許可																																																																																																																					
RPE	DTIE	説 明																																																																																																																				
0	0	I/Oモードで転送																																																																																																																				
	1																																																																																																																					
1	0	リピートモードで転送																																																																																																																				
	1	アイドルモードで転送																																																																																																																				
0	インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2																																																																																																																					
1	デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2																																																																																																																					
0	バイトサイズ転送																																																																																																																					
1	ワードサイズ転送																																																																																																																					
0	データ転送を禁止																																																																																																																					
1	データ転送を許可																																																																																																																					

(次頁へ続く)

DTCR0A データトランスファコントロールレジスタ 0A	H'FFF27	DMAC0																																											
<p>(前頁より続く)</p> <p>フルアドレスモード</p> <p>ビット:           7           6           5           4           3           2           1           0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 12.5%;">DTE</td> <td style="width: 12.5%;">DTSZ</td> <td style="width: 12.5%;">SAID</td> <td style="width: 12.5%;">SAIDE</td> <td style="width: 12.5%;">DTIE</td> <td style="width: 12.5%;">DTS2A</td> <td style="width: 12.5%;">DTS1A</td> <td style="width: 12.5%;">DTS0A</td> </tr> </table> <p>初期値:           0           0           0           0           0           0           0           0</p> <p>R/W:            R/W       R/W       R/W       R/W       R/W       R/W       R/W       R/W</p> <div style="margin-top: 20px;"> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">データトランスファセレクト0A</td> </tr> <tr> <td style="width: 20px; text-align: center;">0</td> <td>ノーマルモードで動作</td> </tr> <tr> <td style="text-align: center;">1</td> <td>ブロック転送モードで動作</td> </tr> </table> <p style="text-align: center; margin-top: 10px;">データトランスファセレクト2A、1A いずれも1にセットしてください</p> <p style="text-align: center; margin-top: 10px;">データトランスファインタラプトイネーブル</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>DTEビットによる割込み要求を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>DTEビットによる割込み要求を許可</td> </tr> </table> <p style="text-align: center; margin-top: 10px;">ソースアドレスインクリメント/デクリメント (ビット5) ソースアドレスインクリメント/デクリメントイネーブル (ビット4)</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット5</th> <th style="width: 10%;">ビット4</th> <th style="width: 80%;">インクリメント/デクリメントイネーブル</th> </tr> <tr> <th>SAID</th> <th>SAIDE</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">0</td> <td style="text-align: center;">0</td> <td>MARA固定</td> </tr> <tr> <td style="text-align: center;">1</td> <td>インクリメント: DTSZ = 0のとき、転送後MARAを +1 DTSZ = 1のとき、転送後MARAを +2</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>MARA固定</td> </tr> <tr> <td style="text-align: center;">1</td> <td>デクリメント: DTSZ = 0のとき、転送後MARAを -1 DTSZ = 1のとき、転送後MARAを -2</td> </tr> </tbody> </table> <p style="margin-top: 20px;">データトランスファサイズ</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>バイトサイズ転送</td> </tr> <tr> <td style="text-align: center;">1</td> <td>ワードサイズ転送</td> </tr> </table> <p style="margin-top: 10px;">データトランスファイネーブル</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>データ転送を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>データ転送を許可</td> </tr> </table> </div>				DTE	DTSZ	SAID	SAIDE	DTIE	DTS2A	DTS1A	DTS0A	データトランスファセレクト0A		0	ノーマルモードで動作	1	ブロック転送モードで動作	0	DTEビットによる割込み要求を禁止	1	DTEビットによる割込み要求を許可	ビット5	ビット4	インクリメント/デクリメントイネーブル	SAID	SAIDE		0	0	MARA固定	1	インクリメント: DTSZ = 0のとき、転送後MARAを +1 DTSZ = 1のとき、転送後MARAを +2	1	0	MARA固定	1	デクリメント: DTSZ = 0のとき、転送後MARAを -1 DTSZ = 1のとき、転送後MARAを -2	0	バイトサイズ転送	1	ワードサイズ転送	0	データ転送を禁止	1	データ転送を許可
	DTE	DTSZ	SAID	SAIDE	DTIE	DTS2A	DTS1A	DTS0A																																					
データトランスファセレクト0A																																													
0	ノーマルモードで動作																																												
1	ブロック転送モードで動作																																												
0	DTEビットによる割込み要求を禁止																																												
1	DTEビットによる割込み要求を許可																																												
ビット5	ビット4	インクリメント/デクリメントイネーブル																																											
SAID	SAIDE																																												
0	0	MARA固定																																											
	1	インクリメント: DTSZ = 0のとき、転送後MARAを +1 DTSZ = 1のとき、転送後MARAを +2																																											
1	0	MARA固定																																											
	1	デクリメント: DTSZ = 0のとき、転送後MARAを -1 DTSZ = 1のとき、転送後MARAを -2																																											
0	バイトサイズ転送																																												
1	ワードサイズ転送																																												
0	データ転送を禁止																																												
1	データ転送を許可																																												

MAR0B R、E、H、L メモリアドレスレジスタ0B R、E、H、L	H'FFF28 H'FFF29 H'FFF2A H'FFF2B	DMAC0																																																																																																																																
<p>ビット: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">31</td><td style="width: 20px;">30</td><td style="width: 20px;">29</td><td style="width: 20px;">28</td><td style="width: 20px;">27</td><td style="width: 20px;">26</td><td style="width: 20px;">25</td><td style="width: 20px;">24</td><td style="width: 20px;">23</td><td style="width: 20px;">22</td><td style="width: 20px;">21</td><td style="width: 20px;">20</td><td style="width: 20px;">19</td><td style="width: 20px;">18</td><td style="width: 20px;">17</td><td style="width: 20px;">16</td> </tr> <tr> <td colspan="8">初期値: 1 1 1 1 1 1 1 1</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W: — — — — — — — —</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">MAR0BR</td> <td colspan="8">MAR0BE</td> </tr> </table> <p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">15</td><td style="width: 20px;">14</td><td style="width: 20px;">13</td><td style="width: 20px;">12</td><td style="width: 20px;">11</td><td style="width: 20px;">10</td><td style="width: 20px;">9</td><td style="width: 20px;">8</td><td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="8">初期値: 不定</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W: R/W R/W R/W R/W R/W R/W R/W R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">MAR0BH</td> <td colspan="8">MAR0BL</td> </tr> </table> <p style="text-align: center;">ソースアドレスまたはデスティネーションアドレスを設定</p>			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	初期値: 1 1 1 1 1 1 1 1								不定								R/W: — — — — — — — —								R/W R/W R/W R/W R/W R/W R/W R/W								MAR0BR								MAR0BE								15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	初期値: 不定								不定								R/W: R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W								MAR0BH								MAR0BL							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16																																																																																																																			
初期値: 1 1 1 1 1 1 1 1								不定																																																																																																																										
R/W: — — — — — — — —								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
MAR0BR								MAR0BE																																																																																																																										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																			
初期値: 不定								不定																																																																																																																										
R/W: R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
MAR0BH								MAR0BL																																																																																																																										
ETCR0B H、L 転送カウントレジスタ0B H、L	H'FFF2C H'FFF2D	DMAC0																																																																																																																																
<p style="text-align: center;">ショートアドレスモード</p> <p>(1) I/Oモードまたはアイドルモード</p> <p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">15</td><td style="width: 20px;">14</td><td style="width: 20px;">13</td><td style="width: 20px;">12</td><td style="width: 20px;">11</td><td style="width: 20px;">10</td><td style="width: 20px;">9</td><td style="width: 20px;">8</td><td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="16">初期値: 不定</td> </tr> <tr> <td colspan="16">R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> </table> <p style="text-align: center;">転送カウンタ</p> <p>(2) リピートモード</p> <p>ビット: 7 6 5 4 3 2 1 0      7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> <td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="8">初期値: 不定</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W: R/W R/W R/W R/W R/W R/W R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">ETCR0BH</td> <td colspan="8">ETCR0BL</td> </tr> </table> <p style="text-align: center;"> <span style="margin-right: 100px;">転送カウンタ</span> <span>転送回数保持</span> </p>			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	初期値: 不定																R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W																7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	初期値: 不定								不定								R/W: R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W								ETCR0BH								ETCR0BL																							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																			
初期値: 不定																																																																																																																																		
R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																																		
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0																																																																																																																			
初期値: 不定								不定																																																																																																																										
R/W: R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
ETCR0BH								ETCR0BL																																																																																																																										

ETCR0B H、L 転送カウントレジスタ0B H、L	H'FFF2C H'FFF2D	DMAC0																																
<p>(前頁より続く)</p> <p>フルアドレスモード</p> <p>(1) ノーマルモード</p> <div style="margin-left: 40px;"> <p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; height: 20px; border-collapse: collapse;"> <tr> <td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td> </tr> </table> <p>初期値: 不定</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <div style="margin-left: 100px;"> <p>↓</p> <p>未使用</p> </div> </div> <p>(2) ブロック転送モード</p> <div style="margin-left: 40px;"> <p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; height: 20px; border-collapse: collapse;"> <tr> <td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td><td style="width: 12.5%;"></td> </tr> </table> <p>初期値: 不定</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <div style="margin-left: 100px;"> <p>↓</p> <p>ブロック転送カウンタ</p> </div> </div>																																		

IOAR0B I/O アドレスレジスタ 0B	H'FFF2E	DMAC0								
<div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;"> <p>ビット:</p> <p>7      6      5      4      3      2      1      0</p> <table border="1" style="margin: 0 auto; border-collapse: collapse;"> <tr> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> </table> </div> <div style="margin-left: 20px;"> <p>初期値:</p> <p>R/W:    R/W    R/W    R/W    不定    R/W    R/W    R/W</p> <hr style="width: 100%; border: 0.5px solid black;"/> <p style="margin-left: 40px;"> </p> <p style="margin-left: 40px;">ショートアドレスモード：ソースアドレスまたはデスティネーションアドレスを設定</p> <p style="margin-left: 40px;">フルアドレスモード       : 未使用</p> </div> </div>										



DTCR0B データトランスファコントロールレジスタ 0B	H'FFF2F	DMAC0																																																										
<p><b>ショートアドレスモード</b></p> <p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" style="margin: 0 auto; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">DTE</td> <td style="width: 20px; text-align: center;">DTSZ</td> <td style="width: 20px; text-align: center;">DTID</td> <td style="width: 20px; text-align: center;">RPE</td> <td style="width: 20px; text-align: center;">DTIE</td> <td style="width: 20px; text-align: center;">DTS2</td> <td style="width: 20px; text-align: center;">DTS1</td> <td style="width: 20px; text-align: center;">DTS0</td> </tr> </table> <p>初期値:        0            0            0            0            0            0            0            0</p> <p>R/W:            R/W        R/W        R/W        R/W        R/W        R/W        R/W</p>			DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0																																																		
DTE	DTSZ	DTID	RPE	DTIE	DTS2	DTS1	DTS0																																																					
<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>データトランスファセレクト</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット2</th> <th style="width: 10%;">ビット1</th> <th style="width: 10%;">ビット0</th> <th style="width: 70%;">データ転送の起動要因</th> </tr> <tr> <th>DTS2</th> <th>DTS1</th> <th>DTS0</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="4" style="text-align: center;">0</td> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>A/D変換器の変換終了割込みで起動</td> </tr> <tr> <td rowspan="4" style="text-align: center;">1</td> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>SCI0の送信データエンブティ割込みで起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>SCI0の受信データフル割込みで起動</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>DREQ端子の立下がりエッジ入力で起動</td> </tr> <tr> <td style="text-align: center;">1</td> <td>DREQ端子のLowレベル入力で起動</td> </tr> </tbody> </table> <p>データトランスファインタラプトイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="width: 10%;">0</th> <th style="width: 90%;">DTEビットによる割込み要求を禁止</th> </tr> <tr> <th style="width: 10%;">1</th> <th style="width: 90%;">DTEビットによる割込み要求を許可</th> </tr> </table> <p>リピートイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">RPE</th> <th style="width: 10%;">DTIE</th> <th style="width: 80%;">説 明</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td rowspan="2">I/Oモードで転送</td> </tr> <tr> <td style="text-align: center;">1</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>リピートモードで転送</td> </tr> <tr> <td style="text-align: center;">1</td> <td>アイドルモードで転送</td> </tr> </tbody> </table> <p>データトランスファインクリメント/デクリメント</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="width: 10%;">0</th> <th style="width: 90%;">インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2</th> </tr> <tr> <th style="width: 10%;">1</th> <th style="width: 90%;">デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2</th> </tr> </table> <p>データトランスファサイズ</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="width: 10%;">0</th> <th style="width: 90%;">バイトサイズ転送</th> </tr> <tr> <th style="width: 10%;">1</th> <th style="width: 90%;">ワードサイズ転送</th> </tr> </table> <p>データトランスファイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="width: 10%;">0</th> <th style="width: 90%;">データ転送を禁止</th> </tr> <tr> <th style="width: 10%;">1</th> <th style="width: 90%;">データ転送を許可</th> </tr> </table> </div> </div>			ビット2	ビット1	ビット0	データ転送の起動要因	DTS2	DTS1	DTS0		0	0	0	16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動	1	16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動	1	0	16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動	1	A/D変換器の変換終了割込みで起動	1	0	0	SCI0の送信データエンブティ割込みで起動	1	SCI0の受信データフル割込みで起動	1	0	DREQ端子の立下がりエッジ入力で起動	1	DREQ端子のLowレベル入力で起動	0	DTEビットによる割込み要求を禁止	1	DTEビットによる割込み要求を許可	RPE	DTIE	説 明	0	0	I/Oモードで転送	1	1	0	リピートモードで転送	1	アイドルモードで転送	0	インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2	1	デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2	0	バイトサイズ転送	1	ワードサイズ転送	0	データ転送を禁止	1	データ転送を許可
ビット2	ビット1	ビット0	データ転送の起動要因																																																									
DTS2	DTS1	DTS0																																																										
0	0	0	16ビットタイムチャネル0のコンペアマッチ / インพุットキャプチャA割込みで起動																																																									
		1	16ビットタイムチャネル1のコンペアマッチ / インพุットキャプチャA割込みで起動																																																									
	1	0	16ビットタイムチャネル2のコンペアマッチ / インพุットキャプチャA割込みで起動																																																									
		1	A/D変換器の変換終了割込みで起動																																																									
1	0	0	SCI0の送信データエンブティ割込みで起動																																																									
		1	SCI0の受信データフル割込みで起動																																																									
	1	0	DREQ端子の立下がりエッジ入力で起動																																																									
		1	DREQ端子のLowレベル入力で起動																																																									
0	DTEビットによる割込み要求を禁止																																																											
1	DTEビットによる割込み要求を許可																																																											
RPE	DTIE	説 明																																																										
0	0	I/Oモードで転送																																																										
	1																																																											
1	0	リピートモードで転送																																																										
	1	アイドルモードで転送																																																										
0	インクリメント: DTSZ = 0のとき、転送後MARを +1 DTSZ = 1のとき、転送後MARを +2																																																											
1	デクリメント: DTSZ = 0のとき、転送後MARを -1 DTSZ = 1のとき、転送後MARを -2																																																											
0	バイトサイズ転送																																																											
1	ワードサイズ転送																																																											
0	データ転送を禁止																																																											
1	データ転送を許可																																																											

(次頁へ続く)

DTCR0B データトランスファコントロールレジスタ 0B	H'FFF2F	DMAC0
-------------------------------	---------	-------

(前頁より続く)

フルアドレスモード

ビット:	7	6	5	4	3	2	1	0
	DTME	—	DAID	DAIDE	TMS	DTS2B	DTS1B	DTS0B
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

データトランスファマスタイネーブル

0	データ転送を禁止
1	データ転送を許可

データトランスファセレクト2B~0B

データ転送の起動要因			
ビット2 DTS2B	ビット1 DTS1B	ビット0 DTS0B	
ノーマルモード		ブロック転送モード	
0	0	0	オートリクエスト (パーストモード) 16ビットタイムチャネル0のコンペアマッチ/インพุットキャプチャA割込みで起動
		1	使用できません 16ビットタイムチャネル1のコンペアマッチ/インพุットキャプチャA割込みで起動
	1	0	オートリクエスト (サイクルスチールモード) 16ビットタイムチャネル2のコンペアマッチ/インพุットキャプチャA割込みで起動
		1	使用できません A/D変換器の変換終了割込みで起動
1	0	0	使用できません
		1	使用できません
	1	0	DREQ端子の立下がりエッジ入力で起動 DREQ端子の立下がりエッジ入力で起動
		1	DREQ端子のLowレベル入力で起動 使用できません

トランスファモードセレクト

0	ブロック転送モード時、デスティネーション側をブロックエリアとして転送
1	ブロック転送モード時、ソース側をブロックエリアとして転送

デスティネーションアドレスインクリメント/デクリメント (ビット5)  
デスティネーションアドレスインクリメント/デクリメントイネーブル (ビット4)

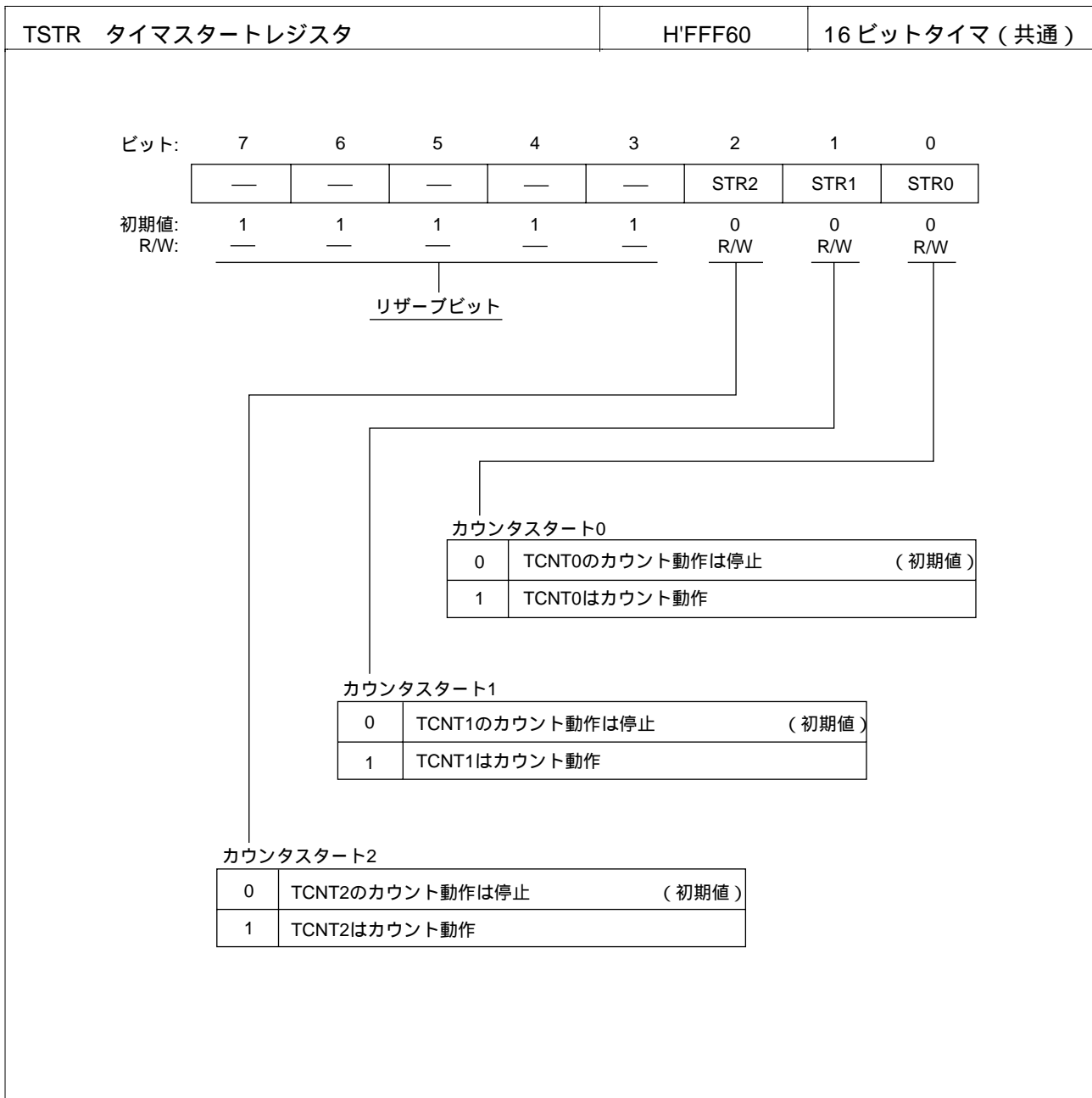
ビット5 DAID	ビット4 DAIDE	説明
0	0	MARB固定
	1	インクリメント: DTSZ = 0のとき、転送後MARBを +1 DTSZ = 1のとき、転送後MARBを +2
1	0	MARB固定
	1	デクリメント: DTSZ = 0のとき、転送後MARBを -1 DTSZ = 1のとき、転送後MARBを -2

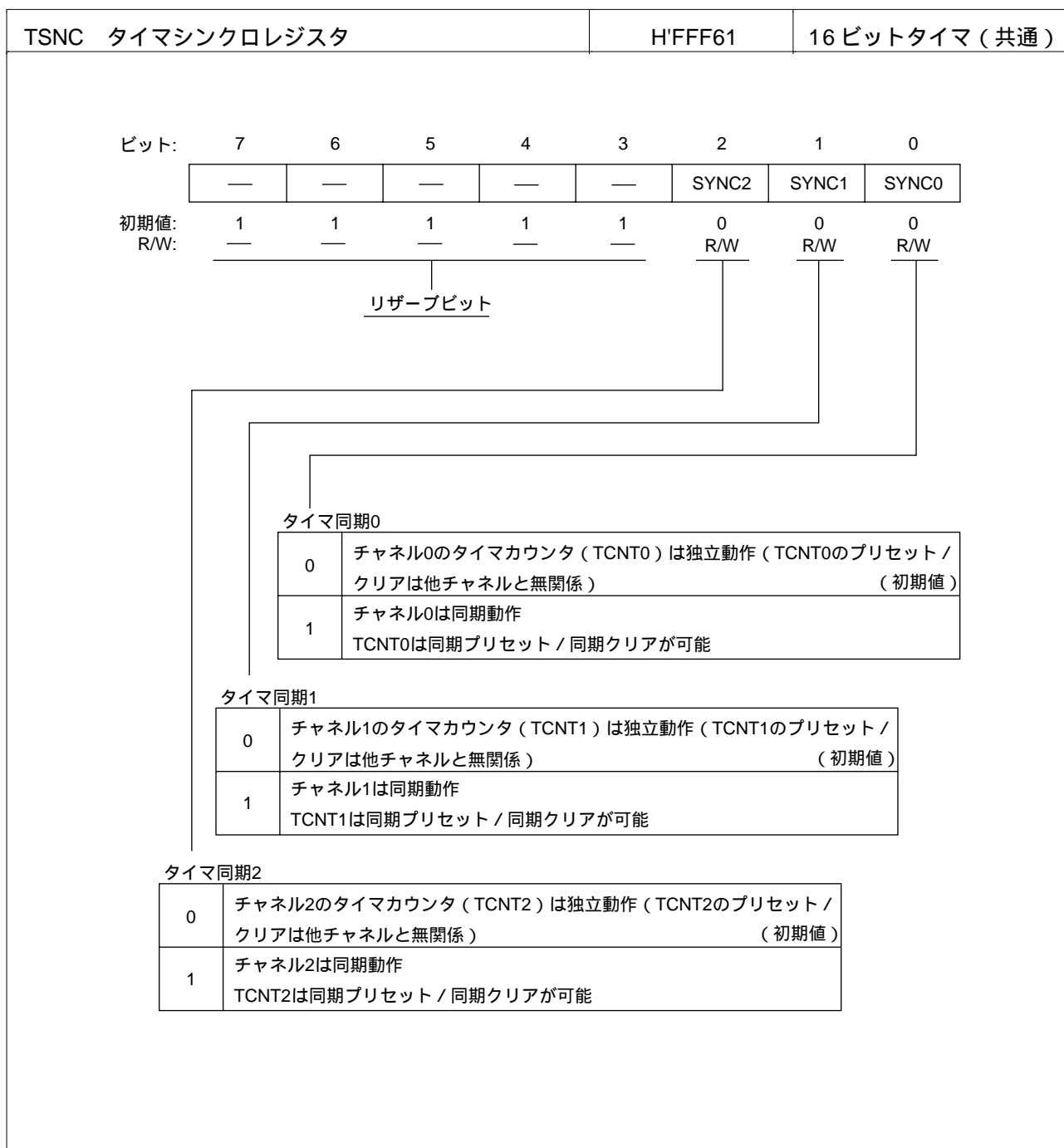
MAR1A R、E、H、L メモリアドレスレジスタ 1A R、E、H、L	H'FFF30 H'FFF31 H'FFF32 H'FFF33	DMAC1																																																																																																																																
<p>ビット: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">31</td><td style="width: 20px;">30</td><td style="width: 20px;">29</td><td style="width: 20px;">28</td><td style="width: 20px;">27</td><td style="width: 20px;">26</td><td style="width: 20px;">25</td><td style="width: 20px;">24</td><td style="width: 20px;">23</td><td style="width: 20px;">22</td><td style="width: 20px;">21</td><td style="width: 20px;">20</td><td style="width: 20px;">19</td><td style="width: 20px;">18</td><td style="width: 20px;">17</td><td style="width: 20px;">16</td> </tr> <tr> <td colspan="8">1 1 1 1 1 1 1 1</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">MAR1AR</td> <td colspan="8">MAR1AE</td> </tr> </table> <p>初期値: 1 1 1 1 1 1 1 1 不定</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">15</td><td style="width: 20px;">14</td><td style="width: 20px;">13</td><td style="width: 20px;">12</td><td style="width: 20px;">11</td><td style="width: 20px;">10</td><td style="width: 20px;">9</td><td style="width: 20px;">8</td><td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="8">不定</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">MAR1AH</td> <td colspan="8">MAR1AL</td> </tr> </table> <p>初期値: 不定 不定</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>MAR1AR MAR1AE</p> <p>MAR1AH MAR1AL</p> <p>機能はDMAC0と同じです。</p>			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	1 1 1 1 1 1 1 1								不定								R/W								R/W R/W R/W R/W R/W R/W R/W R/W								MAR1AR								MAR1AE								15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	不定								不定								R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W								MAR1AH								MAR1AL							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16																																																																																																																			
1 1 1 1 1 1 1 1								不定																																																																																																																										
R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
MAR1AR								MAR1AE																																																																																																																										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																			
不定								不定																																																																																																																										
R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
MAR1AH								MAR1AL																																																																																																																										
ETCR1A H、L 転送カウントレジスタ 1A H、L	H'FFF34 H'FFF35	DMAC1																																																																																																																																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">15</td><td style="width: 20px;">14</td><td style="width: 20px;">13</td><td style="width: 20px;">12</td><td style="width: 20px;">11</td><td style="width: 20px;">10</td><td style="width: 20px;">9</td><td style="width: 20px;">8</td><td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="16">不定</td> </tr> <tr> <td colspan="16">R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> </table> <p>初期値: 不定</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>ビット: 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> <td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="8">不定</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">ETCR1AH</td> <td colspan="8">ETCR1AL</td> </tr> </table> <p>初期値: 不定 不定</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>ETCR1AH ETCR1AL</p> <p>機能はDMAC0と同じです。</p>			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	不定																R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W																7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	不定								不定								R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W								ETCR1AH								ETCR1AL																							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																			
不定																																																																																																																																		
R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																																		
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0																																																																																																																			
不定								不定																																																																																																																										
R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
ETCR1AH								ETCR1AL																																																																																																																										



MAR1B R、E、H、L メモリアドレスレジスタ 1B R、E、H、L	H'FFF38 H'FFF39 H'FFF3A H'FFF3B	DMAC1																																																																																																																																
<p>ビット: 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">31</td><td style="width: 20px;">30</td><td style="width: 20px;">29</td><td style="width: 20px;">28</td><td style="width: 20px;">27</td><td style="width: 20px;">26</td><td style="width: 20px;">25</td><td style="width: 20px;">24</td><td style="width: 20px;">23</td><td style="width: 20px;">22</td><td style="width: 20px;">21</td><td style="width: 20px;">20</td><td style="width: 20px;">19</td><td style="width: 20px;">18</td><td style="width: 20px;">17</td><td style="width: 20px;">16</td> </tr> <tr> <td colspan="8">1 1 1 1 1 1 1 1</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">MAR1BR</td> <td colspan="8">MAR1BE</td> </tr> </table> <p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">15</td><td style="width: 20px;">14</td><td style="width: 20px;">13</td><td style="width: 20px;">12</td><td style="width: 20px;">11</td><td style="width: 20px;">10</td><td style="width: 20px;">9</td><td style="width: 20px;">8</td><td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="8">不定</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">MAR1BH</td> <td colspan="8">MAR1BL</td> </tr> </table> <p>機能はDMAC0と同じです。</p>			31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	1 1 1 1 1 1 1 1								不定								R/W								R/W R/W R/W R/W R/W R/W R/W R/W								MAR1BR								MAR1BE								15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	不定								不定								R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W								MAR1BH								MAR1BL							
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16																																																																																																																			
1 1 1 1 1 1 1 1								不定																																																																																																																										
R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
MAR1BR								MAR1BE																																																																																																																										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																			
不定								不定																																																																																																																										
R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
MAR1BH								MAR1BL																																																																																																																										
ETCR1B H、L 転送カウントレジスタ 1B H、L	H'FFF3C H'FFF3D	DMAC1																																																																																																																																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">15</td><td style="width: 20px;">14</td><td style="width: 20px;">13</td><td style="width: 20px;">12</td><td style="width: 20px;">11</td><td style="width: 20px;">10</td><td style="width: 20px;">9</td><td style="width: 20px;">8</td><td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="16">不定</td> </tr> <tr> <td colspan="16">R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> </table> <p>ビット: 7 6 5 4 3 2 1 0      7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> <td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td colspan="8">不定</td> <td colspan="8">不定</td> </tr> <tr> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> <td colspan="8">R/W R/W R/W R/W R/W R/W R/W R/W</td> </tr> <tr> <td colspan="8">ETCR1BH</td> <td colspan="8">ETCR1BL</td> </tr> </table> <p>機能はDMAC0と同じです。</p>			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	不定																R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W																7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	不定								不定								R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W								ETCR1BH								ETCR1BL																							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																																																																																																			
不定																																																																																																																																		
R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																																		
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0																																																																																																																			
不定								不定																																																																																																																										
R/W R/W R/W R/W R/W R/W R/W R/W								R/W R/W R/W R/W R/W R/W R/W R/W																																																																																																																										
ETCR1BH								ETCR1BL																																																																																																																										









TMDR タイマモードレジスタ	H'FFF62	16 ビットタイマ (共通)						
ビット:	7	6	5	4	3	2	1	0
	—	MDF	FDIR	—	—	PWM2	PWM1	PWM0
初期値:	1	0	0	1	1	0	0	0
R/W:	—	R/W	R/W	—	—	R/W	R/W	R/W

0	チャンネル0は通常動作	(初期値)
1	チャンネル0はPWMモード	

0	チャンネル1は通常動作	(初期値)
1	チャンネル1はPWMモード	

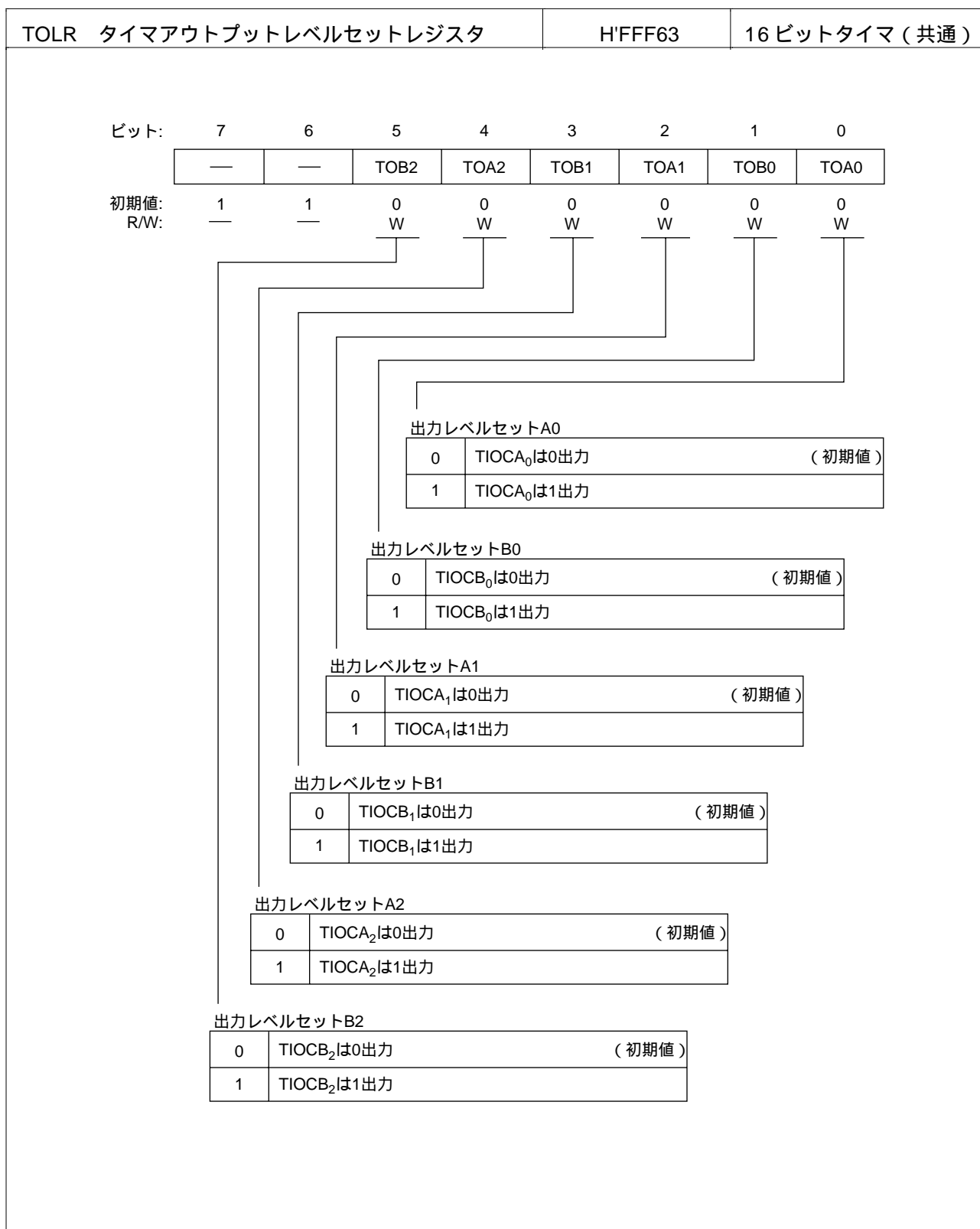
0	チャンネル2は通常動作	(初期値)
1	チャンネル2はPWMモード	

0	TISRCのOVFフラグは、TCNT2がオーバーフローまたはアンダフローしたときに1にセット	(初期値)
1	TISRCのOVFフラグは、TCNT2がオーバーフローしたときに1にセット	

0	チャンネル2は通常動作	(初期値)
1	チャンネル2は位相計数モード	



TISRA タイマインタラプトステータスレジスタ A	H'FFF64	16 ビットタイマ (共通)																								
<p>ビット :      7      6      5      4      3      2      1      0</p> <table style="margin: auto; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 40px; text-align: center;">IMIEA2</td> <td style="border: 1px solid black; width: 40px; text-align: center;">IMIEA1</td> <td style="border: 1px solid black; width: 40px; text-align: center;">IMIEA0</td> <td style="border: 1px solid black; width: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 40px; text-align: center;">IMFA2</td> <td style="border: 1px solid black; width: 40px; text-align: center;">IMFA1</td> <td style="border: 1px solid black; width: 40px; text-align: center;">IMFA0</td> </tr> <tr> <td style="padding-right: 5px;">初期値 :</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="padding-right: 5px;">R/W :</td> <td style="text-align: center;">—</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">—</td> <td style="text-align: center;">R/(W)*</td> <td style="text-align: center;">R/(W)*</td> </tr> </table>			—	IMIEA2	IMIEA1	IMIEA0	—	IMFA2	IMFA1	IMFA0	初期値 :	1	0	0	0	1	0	0	R/W :	—	R/W	R/W	R/W	—	R/(W)*	R/(W)*
—	IMIEA2	IMIEA1	IMIEA0	—	IMFA2	IMFA1	IMFA0																			
初期値 :	1	0	0	0	1	0	0																			
R/W :	—	R/W	R/W	R/W	—	R/(W)*	R/(W)*																			
<p>インプットキャプチャ / コンペアマッチフラグA0 (初期値)</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>                     [クリア条件]                      (1) IMFA0 = 1の状態、IMFA0フラグをリードした後、IMFA0フラグに0をライトしたとき                      (2) IMIA0割込みによりDMACが起動されたとき                 </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                     [セット条件]                      (1) GRA0がアウトプットコンペアレジスタとして機能している場合、TCNT0 = GRA0になったとき                      (2) GRA0がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT0の値がGRA0に転送されたとき                 </td> </tr> </table>			0	[クリア条件] (1) IMFA0 = 1の状態、IMFA0フラグをリードした後、IMFA0フラグに0をライトしたとき (2) IMIA0割込みによりDMACが起動されたとき	1	[セット条件] (1) GRA0がアウトプットコンペアレジスタとして機能している場合、TCNT0 = GRA0になったとき (2) GRA0がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT0の値がGRA0に転送されたとき																				
0	[クリア条件] (1) IMFA0 = 1の状態、IMFA0フラグをリードした後、IMFA0フラグに0をライトしたとき (2) IMIA0割込みによりDMACが起動されたとき																									
1	[セット条件] (1) GRA0がアウトプットコンペアレジスタとして機能している場合、TCNT0 = GRA0になったとき (2) GRA0がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT0の値がGRA0に転送されたとき																									
<p>インプットキャプチャ / コンペアマッチフラグA1 (初期値)</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>                     [クリア条件]                      (1) IMFA1 = 1の状態、IMFA1フラグをリードした後、IMFA1フラグに0をライトしたとき                      (2) IMIA1割込みによりDMACが起動されたとき                 </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                     [セット条件]                      (1) GRA1がアウトプットコンペアレジスタとして機能している場合、TCNT1 = GRA1になったとき                      (2) GRA1がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT1の値がGRA1に転送されたとき                 </td> </tr> </table>			0	[クリア条件] (1) IMFA1 = 1の状態、IMFA1フラグをリードした後、IMFA1フラグに0をライトしたとき (2) IMIA1割込みによりDMACが起動されたとき	1	[セット条件] (1) GRA1がアウトプットコンペアレジスタとして機能している場合、TCNT1 = GRA1になったとき (2) GRA1がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT1の値がGRA1に転送されたとき																				
0	[クリア条件] (1) IMFA1 = 1の状態、IMFA1フラグをリードした後、IMFA1フラグに0をライトしたとき (2) IMIA1割込みによりDMACが起動されたとき																									
1	[セット条件] (1) GRA1がアウトプットコンペアレジスタとして機能している場合、TCNT1 = GRA1になったとき (2) GRA1がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT1の値がGRA1に転送されたとき																									
<p>インプットキャプチャ / コンペアマッチフラグA2 (初期値)</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>                     [クリア条件]                      (1) IMFA2 = 1の状態、IMFA2フラグをリードした後、IMFA2フラグに0をライトしたとき                      (2) IMIA2割込みによりDMACが起動されたとき                 </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                     [セット条件]                      (1) GRA2がアウトプットコンペアレジスタとして機能している場合、TCNT2 = GRA2になったとき                      (2) GRA2がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT2の値がGRA2に転送されたとき                 </td> </tr> </table>			0	[クリア条件] (1) IMFA2 = 1の状態、IMFA2フラグをリードした後、IMFA2フラグに0をライトしたとき (2) IMIA2割込みによりDMACが起動されたとき	1	[セット条件] (1) GRA2がアウトプットコンペアレジスタとして機能している場合、TCNT2 = GRA2になったとき (2) GRA2がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT2の値がGRA2に転送されたとき																				
0	[クリア条件] (1) IMFA2 = 1の状態、IMFA2フラグをリードした後、IMFA2フラグに0をライトしたとき (2) IMIA2割込みによりDMACが起動されたとき																									
1	[セット条件] (1) GRA2がアウトプットコンペアレジスタとして機能している場合、TCNT2 = GRA2になったとき (2) GRA2がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT2の値がGRA2に転送されたとき																									
<p>インプットキャプチャ / コンペアマッチインタラプトイネーブルA0 (初期値)</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>IMFA0フラグによる割込み (IMIA0) 要求を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>IMFA0フラグによる割込み (IMIA0) 要求を許可</td> </tr> </table>			0	IMFA0フラグによる割込み (IMIA0) 要求を禁止	1	IMFA0フラグによる割込み (IMIA0) 要求を許可																				
0	IMFA0フラグによる割込み (IMIA0) 要求を禁止																									
1	IMFA0フラグによる割込み (IMIA0) 要求を許可																									
<p>インプットキャプチャ / コンペアマッチインタラプトイネーブルA1 (初期値)</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>IMFA1フラグによる割込み (IMIA1) 要求を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>IMFA1フラグによる割込み (IMIA1) 要求を許可</td> </tr> </table>			0	IMFA1フラグによる割込み (IMIA1) 要求を禁止	1	IMFA1フラグによる割込み (IMIA1) 要求を許可																				
0	IMFA1フラグによる割込み (IMIA1) 要求を禁止																									
1	IMFA1フラグによる割込み (IMIA1) 要求を許可																									
<p>インプットキャプチャ / コンペアマッチインタラプトイネーブルA2 (初期値)</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>IMFA2フラグによる割込み (IMIA2) 要求を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>IMFA2フラグによる割込み (IMIA2) 要求を許可</td> </tr> </table>			0	IMFA2フラグによる割込み (IMIA2) 要求を禁止	1	IMFA2フラグによる割込み (IMIA2) 要求を許可																				
0	IMFA2フラグによる割込み (IMIA2) 要求を禁止																									
1	IMFA2フラグによる割込み (IMIA2) 要求を許可																									
<p>【注】 * フラグをクリアするための0ライトのみ可能です。</p>																										

TISRB タイマインタラプトステータスレジスタ B	H'FFF65	16ビットタイマ (共通)
----------------------------	---------	---------------

ビット:	7	6	5	4	3	2	1	0
	—	IMIEB2	IMIEB1	IMIEB0	—	IMFB2	IMFB1	IMFB0
初期値:	1	0	0	0	1	0	0	0
R/W :	—	R/W	R/W	R/W	—	R/(W)*	R/(W)*	R/(W)*

インプットキャプチャ/コンペアマッチフラグB0	
0	[クリア条件] (初期値) IMFB0 = 1の状態、IMFB0フラグをリードした後、IMFB0フラグに0をライトしたとき
1	[セット条件] (1) GRB0がアウトプットコンペアレジスタとして機能している場合、TCNT0 = GRB0になったとき (2) GRB0がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT0の値がGRB0に転送されたとき

インプットキャプチャ/コンペアマッチフラグB1	
0	[クリア条件] (初期値) IMFB1 = 1の状態、IMFB1フラグをリードした後、IMFB1フラグに0をライトしたとき
1	[セット条件] (1) GRB1がアウトプットコンペアレジスタとして機能している場合、TCNT1 = GRB1になったとき (2) GRB1がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT1の値がGRB1に転送されたとき

インプットキャプチャ/コンペアマッチフラグB2	
0	[クリア条件] (初期値) IMFB2 = 1の状態、IMFB2フラグをリードした後、IMFB2フラグに0をライトしたとき
1	[セット条件] (1) GRB2がアウトプットコンペアレジスタとして機能している場合、TCNT2 = GRB2になったとき (2) GRB2がインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT2の値がGRB2に転送されたとき

インプットキャプチャ/コンペアマッチインタラプトイネーブルB0	
0	IMFB0フラグによる割込み (IMIB0) 要求を禁止 (初期値)
1	IMFB0フラグによる割込み (IMIB0) 要求を許可

インプットキャプチャ/コンペアマッチインタラプトイネーブルB1	
0	IMFB1フラグによる割込み (IMIB1) 要求を禁止 (初期値)
1	IMFB1フラグによる割込み (IMIB1) 要求を許可

インプットキャプチャ/コンペアマッチインタラプトイネーブルB2	
0	IMFB2フラグによる割込み (IMIB2) 要求を禁止 (初期値)
1	IMFB2フラグによる割込み (IMIB2) 要求を許可

【注】 \* フラグをクリアするための0ライトのみ可能です。

TISRC タイマインタラプトステータスレジスタ C	H'FFF66	16 ビットタイマ (共通)
----------------------------	---------	----------------

ビット:	7	6	5	4	3	2	1	0
	—	OVIE2	OVIE1	OVIE0	—	OVF2	OVF1	OVF0
初期値:	1	0	0	0	1	0	0	0
R/W :	—	R/W	R/W	R/W	—	R/(W)*	R/(W)*	R/(W)*

オーバーフローフラグ0	
0	[クリア条件] (初期値) OVF0=1の状態、OVF0フラグをリードした後、OVF0フラグに0をライトしたとき
1	[セット条件] TCNT0の値がオーバーフロー (H'FFFF H'0000) したとき

オーバーフローフラグ1	
0	[クリア条件] (初期値) OVF1=1の状態、OVF1フラグをリードした後、OVF1フラグに0をライトしたとき
1	[セット条件] TCNT1の値がオーバーフロー (H'FFFF H'0000) したとき

オーバーフローフラグ2	
0	[クリア条件] (初期値) OVF2=1の状態、OVF2フラグをリードした後、OVF2フラグに0をライトしたとき
1	[セット条件] TCNT2の値がオーバーフロー (H'FFFF H'0000) またはアンダフロー (H'0000 H'FFFF) したとき

オーバーフローインタラプトイネーブル0	
0	OVF0フラグによる割込み (OVI0) 要求を禁止 (初期値)
1	OVF0フラグによる割込み (OVI0) 要求を許可

オーバーフローインタラプトイネーブル1	
0	OVF1フラグによる割込み (OVI1) 要求を禁止 (初期値)
1	OVF1フラグによる割込み (OVI1) 要求を許可

オーバーフローインタラプトイネーブル2	
0	OVF2フラグによる割込み (OVI2) 要求を禁止 (初期値)
1	OVF2フラグによる割込み (OVI2) 要求を許可

【注】 \* フラグをクリアするための0ライトのみ可能です。

TCR0 タイマコントロールレジスタ 0	H'FFF68	16ビットタイマチャネル0
----------------------	---------	---------------

ビット:	7	6	5	4	3	2	1	0
	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
初期値:	1	0	0	0	0	0	0	0
R/W:	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W

タイマプリスケラ2~0			
ビット2	ビット1	ビット0	説明
TPSC2	TPSC1	TPSC0	
0	0	0	内部クロック: でカウント (初期値)
		1	内部クロック: /2でカウント
	1	0	内部クロック: /4でカウント
		1	内部クロック: /8でカウント
1	0	0	外部クロックA: TCLKA端子入力でカウント
		1	外部クロックB: TCLKB端子入力でカウント
	1	0	外部クロックC: TCLKC端子入力でカウント
		1	外部クロックD: TCLKD端子入力でカウント

クロックエッジ1、0

ビット4	ビット3	説明
CKEG1	CKEG0	
0	0	立上がりエッジでカウント (初期値)
0	1	立下がりエッジでカウント
1	—	立上がり立下がりエッジの両エッジでカウント

カウンタクリア1、0

ビット6	ビット5	説明
CCLR1	CCLR0	
0	0	TCNTのクリア禁止 (初期値)
	1	GRAのコンペアマッチ / インพุットキャプチャでTCNTをクリア
1	0	GRBのコンペアマッチ / インพุットキャプチャでTCNTをクリア
	1	同期クリア。同期動作をしている他のタイマのカウンタクリアに同期してTCNTをクリア

TIOR0 タイマ I/O コントロールレジスタ 0	H'FFF69	16ビットタイマチャンネル0
----------------------------	---------	----------------

ビット:	7	6	5	4	3	2	1	0
	—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0
初期値:	1	0	0	0	1	0	0	0
R/W :	—	R/W	R/W	R/W	—	R/W	R/W	R/W

I/OコントロールA2~0

ビット2 IOA2	ビット1 IOA1	ビット0 IOA0	説明	
0	0	0	GRAはアウトプット コンペアレジスタ	コンペアマッチによる端子出力禁止 (初期値)
		1		GRAのコンペアマッチで0出力
	1	0		GRAのコンペアマッチで1出力
		1		GRAのコンペアマッチでトグル出力 (チャンネル2のみ1出力)
1	0	0	GRAはインプット キャプチャレジスタ	立上がりエッジでGRAへインプットキャプチャ
		1		立下がりエッジでGRAへインプットキャプチャ
	1	0		立上がり / 立下がりエッジの両エッジでインプット キャプチャ
		1		

I/OコントロールB2~0

ビット6 IOB2	ビット5 IOB1	ビット4 IOB0	説明	
0	0	0	GRBはアウトプット コンペアレジスタ	コンペアマッチによる端子出力禁止 (初期値)
		1		GRBのコンペアマッチで0出力
	1	0		GRBのコンペアマッチで1出力
		1		GRBのコンペアマッチでトグル出力 (チャンネル2のみ1出力)
1	0	0	GRBはインプット キャプチャレジスタ	立上がりエッジでGRBへインプットキャプチャ
		1		立下がりエッジでGRBへインプットキャプチャ
	1	0		立上がり / 立下がりエッジの両エッジでインプット キャプチャ
		1		

TCNT0H、L タイマカウンタ 0H、L	H'FFF6A H'FFF6B	16ビットタイマチャンネル0																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p style="text-align: center;">↑</p> <p style="text-align: center;">アップカウンタ</p>																		
GRA0H、L ジェネラルレジスタ A0 H、L	H'FFF6C H'FFF6D	16ビットタイマチャンネル0																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td> </tr> </table> <p>初期値: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p style="text-align: center;">↑</p> <p style="text-align: center;">アウトプットコンペア/インプットキャプチャ兼用レジスタ</p>																		
GRB0H、L ジェネラルレジスタ B0 H、L	H'FFF6E H'FFF6F	16ビットタイマチャンネル0																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td> </tr> </table> <p>初期値: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p style="text-align: center;">↑</p> <p style="text-align: center;">アウトプットコンペア/インプットキャプチャ兼用レジスタ</p>																		



TCR1 タイマコントロールレジスタ 1	H'FFF70	16ビットタイマチャンネル1																
<p>ビット:      7            6            5            4            3            2            1            0</p> <table border="1"> <tr> <td>——</td> <td>CCLR1</td> <td>CCLR0</td> <td>CKEG1</td> <td>CKEG0</td> <td>TPSC2</td> <td>TPSC1</td> <td>TPSC0</td> </tr> </table> <p>初期値:      1            0            0            0            0            0            0            0</p> <p>R/W:      ——            R/W            R/W            R/W            R/W            R/W            R/W            R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p>			——	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0								
——	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0											
TIOR1 タイマ I/O コントロールレジスタ 1	H'FFF71	16ビットタイマチャンネル1																
<p>ビット:      7            6            5            4            3            2            1            0</p> <table border="1"> <tr> <td>——</td> <td>IOB2</td> <td>IOB1</td> <td>IOB0</td> <td>——</td> <td>IOA2</td> <td>IOA1</td> <td>IOA0</td> </tr> </table> <p>初期値:      1            0            0            0            1            0            0            0</p> <p>R/W:      ——            R/W            R/W            R/W            ——            R/W            R/W            R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p>			——	IOB2	IOB1	IOB0	——	IOA2	IOA1	IOA0								
——	IOB2	IOB1	IOB0	——	IOA2	IOA1	IOA0											
TCNT1H、L タイマカウンタ 1H、L	H'FFF72 H'FFF73	16ビットタイマチャンネル1																
<p>ビット:      15    14    13    12    11    10    9    8    7    6    5    4    3    2    1    0</p> <table border="1"> <tr> <td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table> <p>初期値:      0    0    0    0    0    0    0    0    0    0    0    0    0    0    0</p> <p>R/W:      R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p>																		

GRA1H、L ジェネラルレジスタ A1 H、L	H'FFF74 H'FFF75	16ビットタイマチャンネル1																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td> </tr> </table> <p>初期値: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p>																		
GRB1H、L ジェネラルレジスタ B1 H、L	H'FFF76 H'FFF77	16ビットタイマチャンネル1																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td><td style="width: 20px;"> </td> </tr> </table> <p>初期値: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p>																		
TCR2 タイマコントロールレジスタ 2	H'FFF78	16ビットタイマチャンネル2																
<p>ビット: 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 20px;"> </td><td style="width: 20px;">CCLR1</td><td style="width: 20px;">CCLR0</td><td style="width: 20px;">CKEG1</td><td style="width: 20px;">CKEG0</td><td style="width: 20px;">TPSC2</td><td style="width: 20px;">TPSC1</td><td style="width: 20px;">TPSC0</td> </tr> </table> <p>初期値: 1 0 0 0 0 0 0 0</p> <p>R/W: — R/W R/W R/W R/W R/W R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p> <p>【注】 チャンネル2を位相計数モードに設定したとき、TCR2のCKEG1、CKEG0ビットおよびTPSC2～TPSC0ビットの設定は無効となります。</p>				CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0								
	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0											

TIOR2 タイマ I/O コントロールレジスタ 2	H'FFF79	16 ビットタイマチャンネル 2																
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" style="margin-left: 40px;"> <tr> <td style="width: 20px;">—</td> <td style="width: 20px;">IOB2</td> <td style="width: 20px;">IOB1</td> <td style="width: 20px;">IOB0</td> <td style="width: 20px;">—</td> <td style="width: 20px;">IOA2</td> <td style="width: 20px;">IOA1</td> <td style="width: 20px;">IOA0</td> </tr> </table> <p>初期値:        1            0            0            0            1            0            0            0</p> <p>R/W:        —            R/W        R/W        R/W        —            R/W        R/W        R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p>			—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0								
—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0											
TCNT2H、L タイマカウンタ 2H、L	H'FFF7A、H'FFF7B	16 ビットタイマチャンネル 2																
<p>ビット:    15    14    13    12    11    10    9    8    7    6    5    4    3    2    1    0</p> <table border="1" style="margin-left: 40px;"> <tr> <td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td> </tr> </table> <p>初期値:    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0</p> <p>R/W:    R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p style="text-align: center;">↑</p> <p style="text-align: center;">位相計数モード時：アップ/ダウンカウンタ その他のモード時：アップカウンタ</p>																		
GRA2H、L ジェネラルレジスタ A2 H、L	H'FFF7C、H'FFF7D	16 ビットタイマチャンネル 2																
<p>ビット:    15    14    13    12    11    10    9    8    7    6    5    4    3    2    1    0</p> <table border="1" style="margin-left: 40px;"> <tr> <td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td><td style="width: 15px;"> </td> </tr> </table> <p>初期値:    1    1    1    1    1    1    1    1    1    1    1    1    1    1    1</p> <p>R/W:    R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W</p> <p>機能は16ビットタイマチャンネル0と同じです。</p>																		

GRB2H、L ジェネラルレジスタ B2 H、L	H'FFF7E、H'FFF7F	16ビットタイマチャンネル2																
ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																		
<table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>																		
初期値: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1																		
R/W: R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W																		
機能は16ビットタイマチャンネル0と同じです。																		

TCR0	タイマコントロールレジスタ 0	H'FFF80	8ビットタイマチャンネル 0
TCR1	タイマコントロールレジスタ 1	H'FFF81	8ビットタイマチャンネル 1

ビット:	7	6	5	4	3	2	1	0
	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

クロックセレクト2~0			
0	0	0	クロック入力を禁止
		1	内部クロック： /8立ち上がりエッジでカウント
	1	0	内部クロック： /64立ち上がりエッジでカウント
		1	内部クロック： /8192立ち上がりエッジでカウント
1	0	0	チャンネル0の場合： TCNT1のオーバフロー信号で カウント* チャンネル1の場合： TCNT0のコンペアマッチAで カウント*
		1	外部クロック：立ち下がりエッジでカウント
	1	0	外部クロック：立ち上がりエッジでカウント
		1	外部クロック：立ち上がり / 立ち 下がり両エッジでカウント

【注】\* チャンネル0のクロック入力をTCNT1のオーバフロー信号とし、チャンネル1のクロック入力をTCNT0のコンペアマッチ信号とすると、カウントアップクロックが発生しません。この設定は行わないでください。

カウンタクリア1、0		
0	0	クリアを禁止
	1	コンペアマッチAによりクリア
1	0	コンペアマッチB / インพุットキャプチャBによりクリア
	1	インพุットキャプチャBによりクリア

タイマオーバフローインタラプトイネーブル	
0	OVFによる割込み要求 (OVI) を禁止
1	OVFによる割込み要求 (OVI) を許可

コンペアマッチインタラプトイネーブルA	
0	CMFAによる割込み要求 (CMIA) を禁止
1	CMFAによる割込み要求 (CMIA) を許可

コンペアマッチインタラプトイネーブルB	
0	CMFBによる割込み要求 (CMIB) を禁止
1	CMFBによる割込み要求 (CMIB) を許可

TCSR0 タイマコントロール/ステータスレジスタ 0	H'FFF82	8ビットタイマチャンネル0																																																																			
ビット:	7      6      5      4      3      2      1      0																																																																				
	CMFB CMFA OVF ADTE OIS3 OIS2 OS1 OS0																																																																				
初期値:	0      0      0      0      0      0      0      0																																																																				
R/W:	R/(W)*1 R/(W)*1 R/(W)*1 R/W R/W R/W R/W R/W																																																																				
		<p>アウトプットセレクトA1、0</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>ビット1</th> <th>ビット0</th> <th>説明</th> </tr> <tr> <th>OS1</th> <th>OS0</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>コンペアマッチAで変化しない</td> </tr> <tr> <td style="text-align: center;">1</td> <td>コンペアマッチAで0出力</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>コンペアマッチAで1出力</td> </tr> <tr> <td style="text-align: center;">1</td> <td>コンペアマッチAごとに反転出力(トグル出力)</td> </tr> </tbody> </table> <p>アウトプット/インプットキャプチャエッジセレクトB3、2</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>TCSR1のICE</th> <th>ビット3 OIS3</th> <th>ビット2 OIS2</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td rowspan="4" style="text-align: center;">0</td> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>コンペアマッチBで変化しない</td> </tr> <tr> <td style="text-align: center;">1</td> <td>コンペアマッチBで0出力</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>コンペアマッチBで1出力</td> </tr> <tr> <td style="text-align: center;">1</td> <td>コンペアマッチBごとに反転出力(トグル出力)</td> </tr> <tr> <td rowspan="4" style="text-align: center;">1</td> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>立ち上がりエッジでTCORBにインプットキャプチャ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>立ち下がりエッジでTCORBにインプットキャプチャ</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>立ち上がり/立ち下がりの両エッジでTCORBにインプットキャプチャ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>立ち上がり/立ち下がりの両エッジでTCORBにインプットキャプチャ</td> </tr> </tbody> </table> <p>A/Dトリガイネーブル (TCSR0のみ)</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>TRGE*2</th> <th>ビット4 ADTE</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>コンペアマッチAまたは外部トリガによるA/D変換開始要求を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>コンペアマッチAまたは外部トリガによるA/D変換開始要求を禁止</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>外部トリガによるA/D変換開始要求を許可</td> </tr> <tr> <td style="text-align: center;">1</td> <td>コンペアマッチAによるA/D変換開始要求を許可</td> </tr> </tbody> </table> <p>タイマオーバーフローフラグ</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>[クリア条件] OVF = 1の状態、OVFをリードした後、OVFに0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>[セット条件] TCNTがH'FF H'00になったとき</td> </tr> </tbody> </table> <p>コンペアマッチフラグA</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>[クリア条件] CMFA = 1の状態、CMFAをリードした後、CMFAに0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>[セット条件] TCNT = TCORAになったとき</td> </tr> </tbody> </table> <p>コンペアマッチ/インプットキャプチャフラグB</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>[クリア条件] CMFB = 1の状態、CMFBをリードした後、CMFBに0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>[セット条件] (1) TCNT = TCORBになったとき (2) TCORBがインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT値がTCORBに転送されたとき</td> </tr> </tbody> </table>	ビット1	ビット0	説明	OS1	OS0		0	0	コンペアマッチAで変化しない	1	コンペアマッチAで0出力	1	0	コンペアマッチAで1出力	1	コンペアマッチAごとに反転出力(トグル出力)	TCSR1のICE	ビット3 OIS3	ビット2 OIS2	説明	0	0	0	コンペアマッチBで変化しない	1	コンペアマッチBで0出力	1	0	コンペアマッチBで1出力	1	コンペアマッチBごとに反転出力(トグル出力)	1	0	0	立ち上がりエッジでTCORBにインプットキャプチャ	1	立ち下がりエッジでTCORBにインプットキャプチャ	1	0	立ち上がり/立ち下がりの両エッジでTCORBにインプットキャプチャ	1	立ち上がり/立ち下がりの両エッジでTCORBにインプットキャプチャ	TRGE*2	ビット4 ADTE	説明	0	0	コンペアマッチAまたは外部トリガによるA/D変換開始要求を禁止	1	コンペアマッチAまたは外部トリガによるA/D変換開始要求を禁止	1	0	外部トリガによるA/D変換開始要求を許可	1	コンペアマッチAによるA/D変換開始要求を許可	0	[クリア条件] OVF = 1の状態、OVFをリードした後、OVFに0をライトしたとき	1	[セット条件] TCNTがH'FF H'00になったとき	0	[クリア条件] CMFA = 1の状態、CMFAをリードした後、CMFAに0をライトしたとき	1	[セット条件] TCNT = TCORAになったとき	0	[クリア条件] CMFB = 1の状態、CMFBをリードした後、CMFBに0をライトしたとき	1	[セット条件] (1) TCNT = TCORBになったとき (2) TCORBがインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT値がTCORBに転送されたとき
ビット1	ビット0	説明																																																																			
OS1	OS0																																																																				
0	0	コンペアマッチAで変化しない																																																																			
	1	コンペアマッチAで0出力																																																																			
1	0	コンペアマッチAで1出力																																																																			
	1	コンペアマッチAごとに反転出力(トグル出力)																																																																			
TCSR1のICE	ビット3 OIS3	ビット2 OIS2	説明																																																																		
0	0	0	コンペアマッチBで変化しない																																																																		
		1	コンペアマッチBで0出力																																																																		
	1	0	コンペアマッチBで1出力																																																																		
		1	コンペアマッチBごとに反転出力(トグル出力)																																																																		
1	0	0	立ち上がりエッジでTCORBにインプットキャプチャ																																																																		
		1	立ち下がりエッジでTCORBにインプットキャプチャ																																																																		
	1	0	立ち上がり/立ち下がりの両エッジでTCORBにインプットキャプチャ																																																																		
		1	立ち上がり/立ち下がりの両エッジでTCORBにインプットキャプチャ																																																																		
TRGE*2	ビット4 ADTE	説明																																																																			
0	0	コンペアマッチAまたは外部トリガによるA/D変換開始要求を禁止																																																																			
	1	コンペアマッチAまたは外部トリガによるA/D変換開始要求を禁止																																																																			
1	0	外部トリガによるA/D変換開始要求を許可																																																																			
	1	コンペアマッチAによるA/D変換開始要求を許可																																																																			
0	[クリア条件] OVF = 1の状態、OVFをリードした後、OVFに0をライトしたとき																																																																				
1	[セット条件] TCNTがH'FF H'00になったとき																																																																				
0	[クリア条件] CMFA = 1の状態、CMFAをリードした後、CMFAに0をライトしたとき																																																																				
1	[セット条件] TCNT = TCORAになったとき																																																																				
0	[クリア条件] CMFB = 1の状態、CMFBをリードした後、CMFBに0をライトしたとき																																																																				
1	[セット条件] (1) TCNT = TCORBになったとき (2) TCORBがインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNT値がTCORBに転送されたとき																																																																				
<p>【注】*1 ビット7-5は、フラグをクリアするための0ライトのみ可能です。 *2 TRGEはA/Dコントロールレジスタ (ADCR) のビット7です。</p>																																																																					

TCSR1 タイマコントロール / ステータスレジスタ 1	H'FFF83	8ビットタイマチャンネル1
-------------------------------	---------	---------------

ビット:	7	6	5	4	3	2	1	0
	CMFB	CMFA	OVF	ICE	OIS3	OIS2	OS1	OS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W	R/W

ビット1	ビット0	説明
0	0	コンペアマッチAで変化しない
	1	コンペアマッチAで0出力
1	0	コンペアマッチAで1出力
	1	コンペアマッチAごとに反転出力 (トグル出力)

TCSR1 のICE	ビット3 OIS3	ビット2 OIS2	説明
0	0	0	コンペアマッチBで変化しない
		1	コンペアマッチBで0出力
	1	0	コンペアマッチBで1出力
		1	コンペアマッチBごとに反転出力 (トグル出力)
1	0	0	立ち上がりエッジでTCORBに インプットキャプチャ
		1	立ち下がりエッジでTCORBに インプットキャプチャ
	1	0	立ち上がり / 立ち下がり両エッジで TCORBにインプットキャプチャ
		1	立ち上がり / 立ち下がり両エッジで TCORBにインプットキャプチャ

0	TCORBはコンペアマッチレジスタ
1	TCORBはインプットキャプチャレジスタ

0	{ クリア条件 } OVF = 1の状態、OVFをリードした後、OVFに0をライトしたとき
1	{ セット条件 } TCNTがH'FFF H'00になったとき

0	{ クリア条件 } CMFA = 1の状態、CMFAをリードした後、CMFAに0をライトしたとき
1	{ セット条件 } TCNT = TCORAになったとき

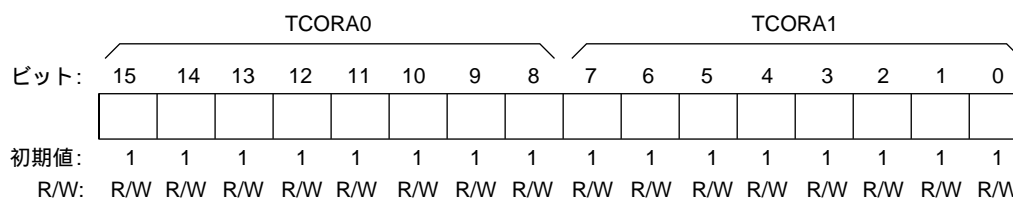
  

0	{ クリア条件 } CMFB = 1の状態、CMFBをリードした後、CMFBに0をライトしたとき
1	{ セット条件 } (1) TCNT = TCORBになったとき (2) TCORBがインプットキャプチャレジスタとして機能している場合、インプット キャプチャ信号によりTCNT値がTCORBに転送されたとき

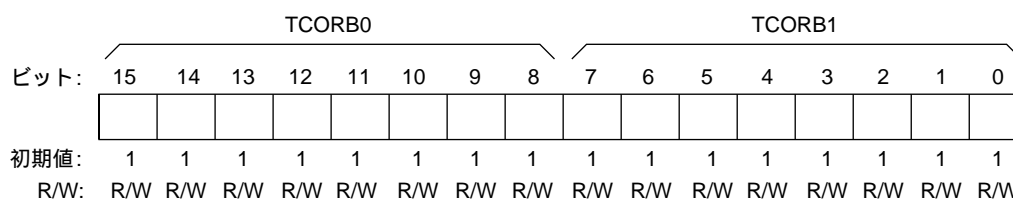
  

【注】\* ビット7~5は、フラグをクリアするための0ライトのみ可能です。

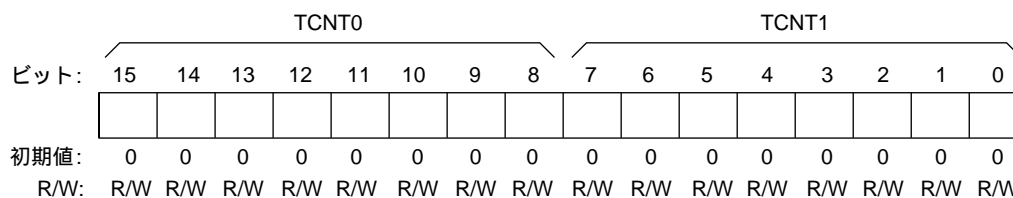
TCORA0	タイムコンスタントレジスタ A0	H'FFF84	8ビットタイマチャンネル0
TCORA1	タイムコンスタントレジスタ A1	H'FFF85	8ビットタイマチャンネル1



TCORB0	タイムコンスタントレジスタ B0	H'FFF86	8ビットタイマチャンネル0
TCORB1	タイムコンスタントレジスタ B1	H'FFF87	8ビットタイマチャンネル1



TCNT0	タイマカウンタ 0	H'FFF88	8ビットタイマチャンネル0
TCNT1	タイマカウンタ 1	H'FFF89	8ビットタイマチャンネル1





TCSR タイマコントロール / ステータスレジスタ	H'FFF8C	WDT																										
ビット:    7            6            5            4            3            2            1            0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;">OVF</td> <td style="width: 10%;">WT/IT</td> <td style="width: 10%;">TME</td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> <td style="width: 10%;">CKS2</td> <td style="width: 10%;">CKS1</td> <td style="width: 10%;">CKS0</td> </tr> </table>	OVF	WT/IT	TME	—	—	CKS2	CKS1	CKS0																				
OVF	WT/IT	TME	—	—	CKS2	CKS1	CKS0																					
初期値:    0            0            0            1            1            0            0            0 R/W:      R/(W)*    R/W        R/W        —        —        R/W        R/W        R/W																												
		クロックセレクト2~0 <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>CKS2</th> <th>CKS1</th> <th>CKS0</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td rowspan="4" style="text-align: center;">0</td> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">/ 2</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">/ 32</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">/ 64</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">/ 128</td> </tr> <tr> <td rowspan="4" style="text-align: center;">1</td> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">/ 256</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">/ 512</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">/ 2048</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">/ 4096</td> </tr> </tbody> </table>	CKS2	CKS1	CKS0	説明	0	0	0	/ 2	1	/ 32	1	0	/ 64	1	/ 128	1	0	0	/ 256	1	/ 512	1	0	/ 2048	1	/ 4096
CKS2	CKS1	CKS0	説明																									
0	0	0	/ 2																									
		1	/ 32																									
	1	0	/ 64																									
		1	/ 128																									
1	0	0	/ 256																									
		1	/ 512																									
	1	0	/ 2048																									
		1	/ 4096																									
		タイマイネーブル <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>                             タイマディスエーブル                              ・ TCNTをH'00にイニシャライズし、                              カウントアップを停止                         </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                             タイマイネーブル                              ・ TCNTはカウントアップ開始                         </td> </tr> </tbody> </table>	0	タイマディスエーブル ・ TCNTをH'00にイニシャライズし、 カウントアップを停止	1	タイマイネーブル ・ TCNTはカウントアップ開始																						
0	タイマディスエーブル ・ TCNTをH'00にイニシャライズし、 カウントアップを停止																											
1	タイマイネーブル ・ TCNTはカウントアップ開始																											
		タイマモードセレクト <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>                             インターバルタイマを選択                              (インターバルタイマ割込み要求)                         </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                             ウォッチドッグタイマを選択                              (リセット信号を発生)                         </td> </tr> </tbody> </table>	0	インターバルタイマを選択 (インターバルタイマ割込み要求)	1	ウォッチドッグタイマを選択 (リセット信号を発生)																						
0	インターバルタイマを選択 (インターバルタイマ割込み要求)																											
1	ウォッチドッグタイマを選択 (リセット信号を発生)																											
		オーバフローフラグ <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>                             [クリア条件]                              OVF = 1の状態でのOVFフラグをリードした後、                              OVFフラグに0をライトしたとき                         </td> </tr> <tr> <td style="text-align: center;">1</td> <td>                             [セット条件]                              TCNTがH'FF H'00に変化したとき                         </td> </tr> </tbody> </table>	0	[クリア条件] OVF = 1の状態でのOVFフラグをリードした後、 OVFフラグに0をライトしたとき	1	[セット条件] TCNTがH'FF H'00に変化したとき																						
0	[クリア条件] OVF = 1の状態でのOVFフラグをリードした後、 OVFフラグに0をライトしたとき																											
1	[セット条件] TCNTがH'FF H'00に変化したとき																											

【注】\* フラグをクリアするための0ライトのみ可能です。

TCNT タイマカウンタ	H'FFF8D リード時、H'FFF8C ライト時	WDT								
<p>ビット: 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0</p> <p>R/W: R/W R/W R/W R/W R/W R/W R/W R/W</p> <p style="text-align: center;"> </p> <p style="text-align: center;">カウント値</p>										

RSTCSR リセットコントロール/ステータスレジスタ	H'FFF8F リード時、H'FFF8E ライト時	WDT																
<p>ビット: 7 6 5 4 3 2 1 0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 12.5%;">WRST</td> <td style="width: 12.5%;">RSTOE</td> <td style="width: 12.5%;">—</td> <td style="width: 12.5%;">—</td> <td style="width: 12.5%;">—</td> <td style="width: 12.5%;">—</td> <td style="width: 12.5%;">—</td> <td style="width: 12.5%;">—</td> </tr> </table> <p>初期値: 0 0 1 1 1 1 1 1</p> <p>R/W: R/(W)* R/W — — — — — —</p> <p style="text-align: center;"> </p> <p style="text-align: center;">リセット出力イネーブル</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>リセット信号の外部出力を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>リセット信号の外部出力を許可</td> </tr> </table> <p style="text-align: center;"> </p> <p style="text-align: center;">ウォッチドッグタイマリセット</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>〔クリア条件〕 ・RES端子によるリセット信号 ・WRST = 1の状態、WRSTフラグをリード後、WRSTフラグに0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 TCNTがオーバーフローし、リセット信号が発生したとき</td> </tr> </table>			WRST	RSTOE	—	—	—	—	—	—	0	リセット信号の外部出力を禁止	1	リセット信号の外部出力を許可	0	〔クリア条件〕 ・RES端子によるリセット信号 ・WRST = 1の状態、WRSTフラグをリード後、WRSTフラグに0をライトしたとき	1	〔セット条件〕 TCNTがオーバーフローし、リセット信号が発生したとき
WRST	RSTOE	—	—	—	—	—	—											
0	リセット信号の外部出力を禁止																	
1	リセット信号の外部出力を許可																	
0	〔クリア条件〕 ・RES端子によるリセット信号 ・WRST = 1の状態、WRSTフラグをリード後、WRSTフラグに0をライトしたとき																	
1	〔セット条件〕 TCNTがオーバーフローし、リセット信号が発生したとき																	
<p>【注】* ビット7は、フラグをクリアする0ライトのみ可能です。</p>																		

TCR2	タイマコントロールレジスタ 2	H'FFF90	8ビットタイマチャンネル 2
TCR3	タイマコントロールレジスタ 3	H'FFF91	8ビットタイマチャンネル 3

ビット:	7	6	5	4	3	2	1	0
	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CKS2	CKS1	CKS0	説明
0	0	0	クロック入力を禁止
		1	内部クロック： /8立ち上がりエッジでカウント
	1	0	内部クロック： /64立ち上がりエッジでカウント
1		内部クロック： /8192立ち上がりエッジでカウント	
1	0	0	チャンネル2の場合： TCNT3のオーバフロー信号で カウント* チャンネル3の場合： TCNT2のコンペアマッチAで カウント*
		1	外部クロック：立ち下がりエッジでカウント
	1	0	外部クロック：立ち上がりエッジでカウント
1		外部クロック：立ち上がり/立ち 下がり両エッジでカウント	

【注】\* チャンネル2のクロック入力をTCNT3のオーバフロー信号とし、チャンネル3のクロック入力をTCNT2のコンペアマッチ信号とすると、カウントアップクロックが発生しません。この設定は行わないでください。

1	0	説明
0	0	クリアを禁止
	1	コンペアマッチAによりクリア
1	0	コンペアマッチB/インプットキャプチャBによりクリア
	1	インプットキャプチャBによりクリア

0	1	説明
0	1	OVFによる割込み要求 (OVI) を禁止 OVFによる割込み要求 (OVI) を許可

0	1	説明
0	1	CMFAによる割込み要求 (CMIA) を禁止 CMFAによる割込み要求 (CMIA) を許可

0	1	説明
0	1	CMFBによる割込み要求 (CMIB) を禁止 CMFBによる割込み要求 (CMIB) を許可

TCSR2	タイマコントロール/ステータスレジスタ 2	H'FFF92	8ビットタイマチャンネル 2
TCSR3	タイマコントロール/ステータスレジスタ 3	H'FFF93	8ビットタイマチャンネル 3

TCSR2	ビット:	7	6	5	4	3	2	1	0
		CMFB	CMFA	OVF	—	OIS3	OIS2	OS1	OS0
	初期値:	0	0	0	1	0	0	0	0
	R/W:	R/(W)*	R/(W)*	R/(W)*	—	R/W	R/W	R/W	R/W
TCSR3	ビット:	7	6	5	4	3	2	1	0
		CMFB	CMFA	OVF	ICE	OIS3	OIS2	OS1	OS0
	初期値:	0	0	0	0	0	0	0	0
	R/W:	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W	R/W

アウトプットセレクトA1、0

ビット1	ビット0	説明
0	0	コンペアマッチAで変化しない
	1	コンペアマッチAで0出力
1	0	コンペアマッチAで1出力
	1	コンペアマッチAごとに反転出力(トグル出力)

アウトプット/インプットキャプチャエッジセレクトB3、2

TCSR3 のICE	ビット3 OIS3	ビット2 OIS2	説明
0	0	0	コンペアマッチBで変化しない
		1	コンペアマッチBで0出力
	1	0	コンペアマッチBで1出力
		1	コンペアマッチBごとに反転出力(トグル出力)
1	0	0	立ち上がりエッジでTCORBにインプットキャプチャ
		1	立ち下がりエッジでTCORBにインプットキャプチャ
	1	0	立ち上がり/立ち下がりの両エッジでTCORBにインプットキャプチャ

インプットキャプチャイネーブル (TCSR3のみ)

0	TCORBはコンペアマッチレジスタ
1	TCORBはインプットキャプチャレジスタ

タイマオーバーフローフラグ

0	〔クリア条件〕 OVF = 1の状態、OVFをリードした後、OVFに0をライトしたとき
1	〔セット条件〕 TCNTがH'FF H'00になったとき

コンペアマッチフラグA

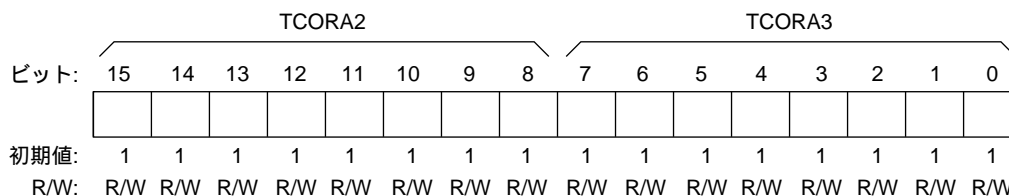
0	〔クリア条件〕 CMFA = 1の状態、CMFAをリードした後、CMFAに0をライトしたとき
1	〔セット条件〕 TCNT = TCORAになったとき

コンペアマッチ/インプットキャプチャフラグB

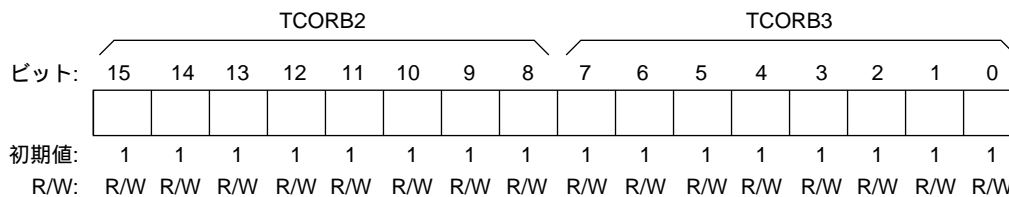
0	〔クリア条件〕 CMFB = 1の状態、CMFBをリードした後、CMFBに0をライトしたとき
1	〔セット条件〕 (1) TCNT = TCORBになったとき (2) TCORBがインプットキャプチャレジスタとして機能している場合、インプットキャプチャ信号によりTCNTの値がTCORBに転送されたとき

【注】\* ビット7-5は、フラグをクリアするための0ライトのみ可能です。

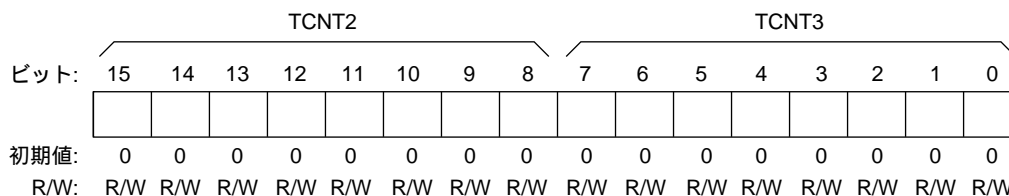
TCORA2	タイムコンスタントレジスタ A2	H'FFF94	8ビットタイマチャンネル2
TCORA3	タイムコンスタントレジスタ A3	H'FFF95	8ビットタイマチャンネル3



TCORB2	タイムコンスタントレジスタ B2	H'FFF96	8ビットタイマチャンネル2
TCORB3	タイムコンスタントレジスタ B3	H'FFF97	8ビットタイマチャンネル3



TCNT2	タイマカウンタ 2	H'FFF98	8ビットタイマチャンネル2
TCNT3	タイマカウンタ 3	H'FFF99	8ビットタイマチャンネル3



DADR0 D/A データレジスタ 0	H'FFF9C	D/A								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> </table> <p>初期値:      0          0          0          0          0          0          0          0</p> <p>R/W:        R/W       R/W       R/W       R/W       R/W       R/W       R/W</p> <hr style="width: 80%; margin: 5px auto;"/> <p style="text-align: center;">D/A変換データを格納</p>										
DADR1 D/A データレジスタ 1	H'FFF9D	D/A								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> <td style="width: 20px; height: 20px;"></td> </tr> </table> <p>初期値:      0          0          0          0          0          0          0          0</p> <p>R/W:        R/W       R/W       R/W       R/W       R/W       R/W       R/W</p> <hr style="width: 80%; margin: 5px auto;"/> <p style="text-align: center;">D/A変換データを格納</p>										

DACR D/A コントロールレジスタ	H'FFF9E	D/A																																
ビット:      7            6            5            4            3            2            1            0 <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;">DAOE1</td> <td style="width: 10%;">DAOE0</td> <td style="width: 10%;">DAE</td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> <td style="width: 10%;">—</td> </tr> </table>	DAOE1	DAOE0	DAE	—	—	—	—	—																										
DAOE1	DAOE0	DAE	—	—	—	—	—																											
初期値:      0            0            0            1            1            1            1            1 R/W:        R/W        R/W        —        —        —        —        —																																		
<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <caption>D/Aイネーブル</caption> <thead> <tr> <th style="width: 15%;">ビット7</th> <th style="width: 15%;">ビット6</th> <th style="width: 15%;">ビット5</th> <th style="width: 55%;">説 明</th> </tr> <tr> <th>DAOE1</th> <th>DAOE0</th> <th>DAE</th> <th></th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">—</td> <td>チャンネル0、1のD/A変換を禁止</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>チャンネル0のD/A変換を許可 チャンネル1のD/A変換を禁止</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td>チャンネル0、1のD/A変換を許可</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>チャンネル0のD/A変換を禁止 チャンネル1のD/A変換を許可</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td>チャンネル0、1のD/A変換を許可</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">—</td> <td>チャンネル0、1のD/A変換を許可</td> </tr> </tbody> </table>	ビット7	ビット6	ビット5	説 明	DAOE1	DAOE0	DAE		0	0	—	チャンネル0、1のD/A変換を禁止	0	1	0	チャンネル0のD/A変換を許可 チャンネル1のD/A変換を禁止	0	1	1	チャンネル0、1のD/A変換を許可	1	0	0	チャンネル0のD/A変換を禁止 チャンネル1のD/A変換を許可	1	0	1	チャンネル0、1のD/A変換を許可	1	1	—	チャンネル0、1のD/A変換を許可		
ビット7	ビット6	ビット5	説 明																															
DAOE1	DAOE0	DAE																																
0	0	—	チャンネル0、1のD/A変換を禁止																															
0	1	0	チャンネル0のD/A変換を許可 チャンネル1のD/A変換を禁止																															
0	1	1	チャンネル0、1のD/A変換を許可																															
1	0	0	チャンネル0のD/A変換を禁止 チャンネル1のD/A変換を許可																															
1	0	1	チャンネル0、1のD/A変換を許可																															
1	1	—	チャンネル0、1のD/A変換を許可																															
<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <caption>D/Aアウトプットイネーブル0</caption> <tbody> <tr> <td style="width: 10%; text-align: center;">0</td> <td>アナログ出力DA<sub>0</sub>を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>チャンネル0のD/A変換を許可 アナログ出力DA<sub>0</sub>を許可</td> </tr> </tbody> </table>	0	アナログ出力DA <sub>0</sub> を禁止	1	チャンネル0のD/A変換を許可 アナログ出力DA <sub>0</sub> を許可																														
0	アナログ出力DA <sub>0</sub> を禁止																																	
1	チャンネル0のD/A変換を許可 アナログ出力DA <sub>0</sub> を許可																																	
<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <caption>D/Aアウトプットイネーブル1</caption> <tbody> <tr> <td style="width: 10%; text-align: center;">0</td> <td>アナログ出力DA<sub>1</sub>を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>チャンネル1のD/A変換を許可 アナログ出力DA<sub>1</sub>を許可</td> </tr> </tbody> </table>	0	アナログ出力DA <sub>1</sub> を禁止	1	チャンネル1のD/A変換を許可 アナログ出力DA <sub>1</sub> を許可																														
0	アナログ出力DA <sub>1</sub> を禁止																																	
1	チャンネル1のD/A変換を許可 アナログ出力DA <sub>1</sub> を許可																																	

TPMR TPC 出力モードレジスタ	H'FFFA0	TPC																
<p>ビット:      7            6            5            4            3            2            1            0</p> <table border="1" style="width: 100%; text-align: center; border-collapse: collapse;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;">G3NOV</td> <td style="width: 12.5%;">G2NOV</td> <td style="width: 12.5%;">G1NOV</td> <td style="width: 12.5%;">G0NOV</td> </tr> </table> <p>初期値:      1            1            1            1            0            0            0            0</p> <p>R/W:        —            —            —            —            R/W        R/W        R/W        R/W</p>					G3NOV	G2NOV	G1NOV	G0NOV										
				G3NOV	G2NOV	G1NOV	G0NOV											
		<p style="text-align: center;">グループ0 ノンオーバーラップ</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: center;">0</td> <td>TPC出力グループ0は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ0は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作</td> </tr> </table> <p style="text-align: center;">グループ1 ノンオーバーラップ</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: center;">0</td> <td>TPC出力グループ1は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ1は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作</td> </tr> </table> <p style="text-align: center;">グループ2 ノンオーバーラップ</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: center;">0</td> <td>TPC出力グループ2は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ2は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作</td> </tr> </table> <p style="text-align: center;">グループ3 ノンオーバーラップ</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: center;">0</td> <td>TPC出力グループ3は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ3は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作</td> </tr> </table>	0	TPC出力グループ0は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）	1	TPC出力グループ0は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作	0	TPC出力グループ1は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）	1	TPC出力グループ1は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作	0	TPC出力グループ2は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）	1	TPC出力グループ2は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作	0	TPC出力グループ3は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）	1	TPC出力グループ3は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作
0	TPC出力グループ0は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）																	
1	TPC出力グループ0は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作																	
0	TPC出力グループ1は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）																	
1	TPC出力グループ1は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作																	
0	TPC出力グループ2は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）																	
1	TPC出力グループ2は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作																	
0	TPC出力グループ3は通常動作（選択された16ビットタイマのコンペアマッチAで出力値を更新）																	
1	TPC出力グループ3は、選択された16ビットタイマのコンペアマッチA、Bによりノンオーバーラップ動作																	



TPCR TPC 出力コントロールレジスタ	H'FFFA1	TPC													
<p>ビット:           7           6           5           4           3           2           1           0</p> <table border="1" style="width:100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="width: 12.5%;">G3CMS1</td> <td style="width: 12.5%;">G3CMS0</td> <td style="width: 12.5%;">G2CMS1</td> <td style="width: 12.5%;">G2CMS0</td> <td style="width: 12.5%;">G1CMS1</td> <td style="width: 12.5%;">G1CMS0</td> <td style="width: 12.5%;">G0CMS1</td> <td style="width: 12.5%;">G0CMS0</td> </tr> </table> <p>初期値:           1           1           1           1           1           1           1           1</p> <p>R/W:            R/W       R/W       R/W       R/W       R/W       R/W       R/W</p>			G3CMS1	G3CMS0	G2CMS1	G2CMS0	G1CMS1	G1CMS0	G0CMS1	G0CMS0					
G3CMS1	G3CMS0	G2CMS1	G2CMS0	G1CMS1	G1CMS0	G0CMS1	G0CMS0								
<p>グループ0コンペアマッチセレクト1、0</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット1</th> <th style="width: 10%;">ビット0</th> <th style="width: 80%;">出力トリガとなる16ビットタイマのチャンネル選択</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>TPC出力グループ0 (TP<sub>3</sub>~TP<sub>0</sub>端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ0 (TP<sub>3</sub>~TP<sub>0</sub>端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>TPC出力グループ0 (TP<sub>3</sub>~TP<sub>0</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ0 (TP<sub>3</sub>~TP<sub>0</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> </tbody> </table>			ビット1	ビット0	出力トリガとなる16ビットタイマのチャンネル選択	0	0	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ	1	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ	1	0	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ	1	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ
ビット1	ビット0	出力トリガとなる16ビットタイマのチャンネル選択													
0	0	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ													
	1	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ													
1	0	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													
	1	TPC出力グループ0 (TP <sub>3</sub> ~TP <sub>0</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													
<p>グループ1コンペアマッチセレクト1、0</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット3</th> <th style="width: 10%;">ビット2</th> <th style="width: 80%;">出力トリガとなる16ビットタイマのチャンネル選択</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>TPC出力グループ1 (TP<sub>7</sub>~TP<sub>4</sub>端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ1 (TP<sub>7</sub>~TP<sub>4</sub>端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>TPC出力グループ1 (TP<sub>7</sub>~TP<sub>4</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ1 (TP<sub>7</sub>~TP<sub>4</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> </tbody> </table>			ビット3	ビット2	出力トリガとなる16ビットタイマのチャンネル選択	0	0	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ	1	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ	1	0	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ	1	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ
ビット3	ビット2	出力トリガとなる16ビットタイマのチャンネル選択													
0	0	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ													
	1	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ													
1	0	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													
	1	TPC出力グループ1 (TP <sub>7</sub> ~TP <sub>4</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													
<p>グループ2コンペアマッチセレクト1、0</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット5</th> <th style="width: 10%;">ビット4</th> <th style="width: 80%;">出力トリガとなる16ビットタイマのチャンネル選択</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>TPC出力グループ2 (TP<sub>11</sub>~TP<sub>8</sub>端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ2 (TP<sub>11</sub>~TP<sub>8</sub>端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>TPC出力グループ2 (TP<sub>11</sub>~TP<sub>8</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ2 (TP<sub>11</sub>~TP<sub>8</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> </tbody> </table>			ビット5	ビット4	出力トリガとなる16ビットタイマのチャンネル選択	0	0	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ	1	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ	1	0	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ	1	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ
ビット5	ビット4	出力トリガとなる16ビットタイマのチャンネル選択													
0	0	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ													
	1	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ													
1	0	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													
	1	TPC出力グループ2 (TP <sub>11</sub> ~TP <sub>8</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													
<p>グループ3コンペアマッチセレクト1、0</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット7</th> <th style="width: 10%;">ビット6</th> <th style="width: 80%;">出力トリガとなる16ビットタイマのチャンネル選択</th> </tr> </thead> <tbody> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>TPC出力グループ3 (TP<sub>15</sub>~TP<sub>12</sub>端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ3 (TP<sub>15</sub>~TP<sub>12</sub>端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>TPC出力グループ3 (TP<sub>15</sub>~TP<sub>12</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TPC出力グループ3 (TP<sub>15</sub>~TP<sub>12</sub>端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ</td> </tr> </tbody> </table>			ビット7	ビット6	出力トリガとなる16ビットタイマのチャンネル選択	0	0	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ	1	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ	1	0	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ	1	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ
ビット7	ビット6	出力トリガとなる16ビットタイマのチャンネル選択													
0	0	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル0のコンペアマッチ													
	1	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル1のコンペアマッチ													
1	0	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													
	1	TPC出力グループ3 (TP <sub>15</sub> ~TP <sub>12</sub> 端子) の出力トリガは16ビットタイマチャンネル2のコンペアマッチ													

NDERB	ネクストデータインープルレジスタ B	H'FFFA2	TPC					
ビット:	7	6	5	4	3	2	1	0
	NDER15	NDER14	NDER13	NDER12	NDER11	NDER10	NDER9	NDER8
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ネクストデータインープル15~8								
ビット7~0		説 明						
NDER15 ~ NDER8								
0		TPC出力TP <sub>15</sub> ~ TP <sub>8</sub> を禁止 ( NDR15 ~ NDR8からPB <sub>7</sub> ~ PB <sub>0</sub> への転送禁止 )						
1		TPC出力TP <sub>15</sub> ~ TP <sub>8</sub> を許可 ( NDR15 ~ NDR8からPB <sub>7</sub> ~ PB <sub>0</sub> への転送許可 )						
NDERA	ネクストデータインープルレジスタ A	H'FFFA3	TPC					
ビット:	7	6	5	4	3	2	1	0
	NDER7	NDER6	NDER5	NDER4	NDER3	NDER2	NDER1	NDER0
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ネクストデータインープル7~0								
ビット7~0		説 明						
NDER7 ~ NDER0								
0		TPC出力TP <sub>7</sub> ~ TP <sub>0</sub> を禁止 ( NDR7 ~ NDR0からPA <sub>7</sub> ~ PA <sub>0</sub> への転送禁止 )						
1		TPC出力TP <sub>7</sub> ~ TP <sub>0</sub> を許可 ( NDR7 ~ NDR0からPA <sub>7</sub> ~ PA <sub>0</sub> への転送許可 )						

NDRB ネクストデータレジスタ B	H'FFFA4/ H'FFFA6	TPC																																				
<p>TPC 出力グループ 2、3 の出力トリガが同一の場合</p> <p>(1) アドレス： H'FFFA4</p> <div style="display: flex; justify-content: space-around; align-items: flex-start; margin-top: 10px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">ビット:</td> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">NDR15</td> <td style="border: 1px solid black; text-align: center;">NDR14</td> <td style="border: 1px solid black; text-align: center;">NDR13</td> <td style="border: 1px solid black; text-align: center;">NDR12</td> <td style="border: 1px solid black; text-align: center;">NDR11</td> <td style="border: 1px solid black; text-align: center;">NDR10</td> <td style="border: 1px solid black; text-align: center;">NDR9</td> <td style="border: 1px solid black; text-align: center;">NDR8</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> </tr> </table> <div style="margin-top: 10px; display: flex; justify-content: space-around; width: 100%;"> <div style="text-align: center; width: 45%;"> <p>TPC出力グループ3の次の出力データを格納</p> </div> <div style="text-align: center; width: 45%;"> <p>TPC出力グループ2の次の出力データを格納</p> </div> </div> </div>			ビット:	7	6	5	4	3	2	1	0		NDR15	NDR14	NDR13	NDR12	NDR11	NDR10	NDR9	NDR8	初期値:	0	0	0	0	0	0	0	0	R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0																														
	NDR15	NDR14	NDR13	NDR12	NDR11	NDR10	NDR9	NDR8																														
初期値:	0	0	0	0	0	0	0	0																														
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																														
<p>(2) アドレス： H'FFFA6</p> <div style="display: flex; justify-content: space-around; align-items: flex-start; margin-top: 10px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">ビット:</td> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> </tr> </table> </div>			ビット:	7	6	5	4	3	2	1	0		—	—	—	—	—	—	—	—	初期値:	1	1	1	1	1	1	1	1	R/W:	—	—	—	—	—	—	—	—
ビット:	7	6	5	4	3	2	1	0																														
	—	—	—	—	—	—	—	—																														
初期値:	1	1	1	1	1	1	1	1																														
R/W:	—	—	—	—	—	—	—	—																														
<p>TPC 出力グループ 2、3 の出力トリガが異なる場合</p> <p>(1) アドレス： H'FFFA4</p> <div style="display: flex; justify-content: space-around; align-items: flex-start; margin-top: 10px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">ビット:</td> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">NDR15</td> <td style="border: 1px solid black; text-align: center;">NDR14</td> <td style="border: 1px solid black; text-align: center;">NDR13</td> <td style="border: 1px solid black; text-align: center;">NDR12</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> </tr> </table> <div style="margin-top: 10px; text-align: center; width: 100%;"> <p>TPC出力グループ3の次の出力データを格納</p> </div> </div>			ビット:	7	6	5	4	3	2	1	0		NDR15	NDR14	NDR13	NDR12	—	—	—	—	初期値:	0	0	0	0	1	1	1	1	R/W:	R/W	R/W	R/W	R/W	—	—	—	—
ビット:	7	6	5	4	3	2	1	0																														
	NDR15	NDR14	NDR13	NDR12	—	—	—	—																														
初期値:	0	0	0	0	1	1	1	1																														
R/W:	R/W	R/W	R/W	R/W	—	—	—	—																														
<p>(2) アドレス： H'FFFA6</p> <div style="display: flex; justify-content: space-around; align-items: flex-start; margin-top: 10px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding-right: 5px;">ビット:</td> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">NDR11</td> <td style="border: 1px solid black; text-align: center;">NDR10</td> <td style="border: 1px solid black; text-align: center;">NDR9</td> <td style="border: 1px solid black; text-align: center;">NDR8</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> </tr> </table> <div style="margin-top: 10px; text-align: center; width: 100%;"> <p>TPC出力グループ2の次の出力データを格納</p> </div> </div>			ビット:	7	6	5	4	3	2	1	0		—	—	—	—	NDR11	NDR10	NDR9	NDR8	初期値:	1	1	1	1	0	0	0	0	R/W:	—	—	—	—	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0																														
	—	—	—	—	NDR11	NDR10	NDR9	NDR8																														
初期値:	1	1	1	1	0	0	0	0																														
R/W:	—	—	—	—	R/W	R/W	R/W	R/W																														

NDRA ネクストデータレジスタ A	H'FFFA5/ H'FFFA7	TPC																																				
<p>TPC 出力グループ 0、1 の出力トリガが同一の場合</p>																																						
<p>(1) アドレス： H'FFFA5</p>																																						
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%; text-align: right;">ビット:</td> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">NDR7</td> <td style="border: 1px solid black; text-align: center;">NDR6</td> <td style="border: 1px solid black; text-align: center;">NDR5</td> <td style="border: 1px solid black; text-align: center;">NDR4</td> <td style="border: 1px solid black; text-align: center;">NDR3</td> <td style="border: 1px solid black; text-align: center;">NDR2</td> <td style="border: 1px solid black; text-align: center;">NDR1</td> <td style="border: 1px solid black; text-align: center;">NDR0</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> </tr> </table> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p>TPC出力グループ1の次の出力データを格納</p> </div> <div style="text-align: center;"> <p>TPC出力グループ0の次の出力データを格納</p> </div> </div>			ビット:	7	6	5	4	3	2	1	0		NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0	初期値:	0	0	0	0	0	0	0	0	R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0																														
	NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0																														
初期値:	0	0	0	0	0	0	0	0																														
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																														
<p>(2) アドレス： H'FFFA7</p>																																						
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%; text-align: right;">ビット:</td> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> </tr> </table>			ビット:	7	6	5	4	3	2	1	0		—	—	—	—	—	—	—	—	初期値:	1	1	1	1	1	1	1	1	R/W:	—	—	—	—	—	—	—	—
ビット:	7	6	5	4	3	2	1	0																														
	—	—	—	—	—	—	—	—																														
初期値:	1	1	1	1	1	1	1	1																														
R/W:	—	—	—	—	—	—	—	—																														
<p>TPC 出力グループ 0、1 の出力トリガが異なる場合</p>																																						
<p>(1) アドレス： H'FFFA5</p>																																						
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%; text-align: right;">ビット:</td> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">NDR7</td> <td style="border: 1px solid black; text-align: center;">NDR6</td> <td style="border: 1px solid black; text-align: center;">NDR5</td> <td style="border: 1px solid black; text-align: center;">NDR4</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> </tr> </table> <div style="text-align: center;"> <p>TPC出力グループ1の次の出力データを格納</p> </div>			ビット:	7	6	5	4	3	2	1	0		NDR7	NDR6	NDR5	NDR4	—	—	—	—	初期値:	0	0	0	0	1	1	1	1	R/W:	R/W	R/W	R/W	R/W	—	—	—	—
ビット:	7	6	5	4	3	2	1	0																														
	NDR7	NDR6	NDR5	NDR4	—	—	—	—																														
初期値:	0	0	0	0	1	1	1	1																														
R/W:	R/W	R/W	R/W	R/W	—	—	—	—																														
<p>(2) アドレス： H'FFFA7</p>																																						
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%; text-align: right;">ビット:</td> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td></td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">—</td> <td style="border: 1px solid black; text-align: center;">NDR3</td> <td style="border: 1px solid black; text-align: center;">NDR2</td> <td style="border: 1px solid black; text-align: center;">NDR1</td> <td style="border: 1px solid black; text-align: center;">NDR0</td> </tr> <tr> <td>初期値:</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> <tr> <td>R/W:</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">—</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> <td style="text-align: center;">R/W</td> </tr> </table> <div style="text-align: center;"> <p>TPC出力グループ0の次の出力データを格納</p> </div>			ビット:	7	6	5	4	3	2	1	0		—	—	—	—	NDR3	NDR2	NDR1	NDR0	初期値:	1	1	1	1	0	0	0	0	R/W:	—	—	—	—	R/W	R/W	R/W	R/W
ビット:	7	6	5	4	3	2	1	0																														
	—	—	—	—	NDR3	NDR2	NDR1	NDR0																														
初期値:	1	1	1	1	0	0	0	0																														
R/W:	—	—	—	—	R/W	R/W	R/W	R/W																														

SMR シリアルモードレジスタ	H'FFFB0	SCIO																						
ビット:	7	6	5	4	3	2	1	0																
	C/A	CHR	PE	O/E	STOP	MP	CKS1	CKS0																
初期値:	0	0	0	0	0	0	0	0																
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																
							クロックセレクト1、0 <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>ビット1</th> <th>ビット0</th> <th>クロックの選択</th> </tr> </thead> <tbody> <tr> <td>CKS1</td> <td>CKS0</td> <td></td> </tr> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>クロック</td> </tr> <tr> <td style="text-align: center;">1</td> <td>/4クロック</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>/16クロック</td> </tr> <tr> <td style="text-align: center;">1</td> <td>/64クロック</td> </tr> </tbody> </table>		ビット1	ビット0	クロックの選択	CKS1	CKS0		0	0	クロック	1	/4クロック	1	0	/16クロック	1	/64クロック
ビット1	ビット0	クロックの選択																						
CKS1	CKS0																							
0	0	クロック																						
	1	/4クロック																						
1	0	/16クロック																						
	1	/64クロック																						
							マルチプロセッサモード <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>マルチプロセッサ機能を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>マルチプロセッサフォーマットを選択</td> </tr> </tbody> </table>		0	マルチプロセッサ機能を禁止	1	マルチプロセッサフォーマットを選択												
0	マルチプロセッサ機能を禁止																							
1	マルチプロセッサフォーマットを選択																							
							ストップビットレングス <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>1ストップビット</td> </tr> <tr> <td style="text-align: center;">1</td> <td>2ストップビット</td> </tr> </tbody> </table>		0	1ストップビット	1	2ストップビット												
0	1ストップビット																							
1	2ストップビット																							
							パリティモード <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>偶数パリティ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>奇数パリティ</td> </tr> </tbody> </table>		0	偶数パリティ	1	奇数パリティ												
0	偶数パリティ																							
1	奇数パリティ																							
							パリティイネーブル <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>パリティビットの付加、およびチェックを禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>パリティビットの付加、およびチェックを許可</td> </tr> </tbody> </table>		0	パリティビットの付加、およびチェックを禁止	1	パリティビットの付加、およびチェックを許可												
0	パリティビットの付加、およびチェックを禁止																							
1	パリティビットの付加、およびチェックを許可																							
							キャラクタレングス <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>8ビットデータ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>7ビットデータ</td> </tr> </tbody> </table>		0	8ビットデータ	1	7ビットデータ												
0	8ビットデータ																							
1	7ビットデータ																							
							コミュニケーションモード (シリアルコミュニケーションインタフェース時) <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>調歩同期式モード</td> </tr> <tr> <td style="text-align: center;">1</td> <td>クロック同期式モード</td> </tr> </tbody> </table>		0	調歩同期式モード	1	クロック同期式モード												
0	調歩同期式モード																							
1	クロック同期式モード																							
							GSMモード (スマートカードインタフェース時) <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td style="text-align: center;">0</td> <td>TENDフラグがスタートビットから12.5 etu*後に発生</td> </tr> <tr> <td style="text-align: center;">1</td> <td>TENDフラグがスタートビットから11.0 etu*後に発生</td> </tr> </tbody> </table>		0	TENDフラグがスタートビットから12.5 etu*後に発生	1	TENDフラグがスタートビットから11.0 etu*後に発生												
0	TENDフラグがスタートビットから12.5 etu*後に発生																							
1	TENDフラグがスタートビットから11.0 etu*後に発生																							
							* etu (Elementary Time Unit) : 1ビットの転送期間																	

BRR ビットレートレジスタ	H'FFFB1	SCI0								
<p>ビット:      7            6            5            4            3            2            1            0</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> </tr> </table> <p>初期値:      1            1            1            1            1            1            1            1</p> <p>R/W:        R/W        R/W        R/W        R/W        R/W        R/W        R/W</p> <hr style="width: 80%; margin: 0 auto;"/> <p style="text-align: center;">シリアル送信 / 受信のビットレートを設定</p>										

SCR シリアルコントロールレジスタ	H'FFFB2	SCI0																																																																																								
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" style="width:100%; border-collapse: collapse; text-align: center;"> <tr> <td style="width: 12.5%;">TIE</td> <td style="width: 12.5%;">RIE</td> <td style="width: 12.5%;">TE</td> <td style="width: 12.5%;">RE</td> <td style="width: 12.5%;">MPIE</td> <td style="width: 12.5%;">TEIE</td> <td style="width: 12.5%;">CKE1</td> <td style="width: 12.5%;">CKE0</td> </tr> <tr> <td>初期値: R/W:</td> <td>0 R/W</td> <td>0 R/W</td> <td>0 R/W</td> <td>0 R/W</td> <td>0 R/W</td> <td>0 R/W</td> <td>0 R/W</td> </tr> </table>	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	初期値: R/W:	0 R/W	0 R/W	0 R/W	0 R/W	0 R/W	0 R/W	0 R/W	<p>レシーブイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>受信動作を禁止</td></tr> <tr><td>1</td><td>受信動作を許可</td></tr> </table> <p>トランスミットイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>送信動作を禁止</td></tr> <tr><td>1</td><td>送信動作を許可</td></tr> </table> <p>クロックイネーブル1、0 (シリアルコミュニケーションインタフェース時)</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <th>ビット1 CKE1</th> <th>ビット0 CKE0</th> <th colspan="2">クロックの選択、出力の許可</th> </tr> <tr> <td rowspan="4">0</td> <td rowspan="2">0</td> <td>調歩同期式モード</td> <td>内部クロック / SCK端子は入出力ポート</td> </tr> <tr> <td>クロック同期式モード</td> <td>内部クロック / SCK端子は同期クロック出力</td> </tr> <tr> <td rowspan="2">1</td> <td>調歩同期式モード</td> <td>内部クロック / SCK端子はクロック出力</td> </tr> <tr> <td>クロック同期式モード</td> <td>内部クロック / SCK端子は同期クロック出力</td> </tr> <tr> <td rowspan="4">1</td> <td rowspan="2">0</td> <td>調歩同期式モード</td> <td>外部クロック / SCK端子はクロック入力</td> </tr> <tr> <td>クロック同期式モード</td> <td>外部クロック / SCK端子は同期クロック入力</td> </tr> <tr> <td rowspan="2">1</td> <td>調歩同期式モード</td> <td>外部クロック / SCK端子はクロック入力</td> </tr> <tr> <td>クロック同期式モード</td> <td>外部クロック / SCK端子は同期クロック入力</td> </tr> </table> <p>クロックイネーブル1、0 (スマートカードインタフェース時)</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr> <th>SMR GM</th> <th>ビット1 CKE1</th> <th>ビット0 CKE0</th> <th>説明</th> </tr> <tr> <td rowspan="2">0</td> <td rowspan="2">0</td> <td>0</td> <td>SCK端子は入出力ポート</td> </tr> <tr> <td>1</td> <td>SCK端子はクロック出力</td> </tr> <tr> <td rowspan="4">1</td> <td rowspan="2">0</td> <td>0</td> <td>SCK端子はLow出力固定</td> </tr> <tr> <td>1</td> <td>SCK端子はクロック出力</td> </tr> <tr> <td rowspan="2">1</td> <td rowspan="2">0</td> <td>0</td> <td>SCK端子はHigh出力固定</td> </tr> <tr> <td>1</td> <td>SCK端子はクロック出力</td> </tr> </table> <p>トランスミットエンドインタラプトイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>送信終了割込み (TEI) 要求を禁止</td></tr> <tr><td>1</td><td>送信終了割込み (TEI) 要求を許可</td></tr> </table> <p>マルチプロセッサインタラプトイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>マルチプロセッサ割込みを禁止 (通常の受信動作を行う)</td></tr> <tr><td>1</td><td>マルチプロセッサ割込みを許可</td></tr> </table> <p>レシーブインタラプトイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>受信データフル割込み (RXI) 要求、受信エラー割込み (ERI) 要求を禁止</td></tr> <tr><td>1</td><td>受信データフル割込み (RXI) 要求、受信エラー割込み (ERI) 要求を許可</td></tr> </table> <p>トランスミットインタラプトイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>送信データエンpty割込み (TXI) 要求を禁止</td></tr> <tr><td>1</td><td>送信データエンpty割込み (TXI) 要求を許可</td></tr> </table>		0	受信動作を禁止	1	受信動作を許可	0	送信動作を禁止	1	送信動作を許可	ビット1 CKE1	ビット0 CKE0	クロックの選択、出力の許可		0	0	調歩同期式モード	内部クロック / SCK端子は入出力ポート	クロック同期式モード	内部クロック / SCK端子は同期クロック出力	1	調歩同期式モード	内部クロック / SCK端子はクロック出力	クロック同期式モード	内部クロック / SCK端子は同期クロック出力	1	0	調歩同期式モード	外部クロック / SCK端子はクロック入力	クロック同期式モード	外部クロック / SCK端子は同期クロック入力	1	調歩同期式モード	外部クロック / SCK端子はクロック入力	クロック同期式モード	外部クロック / SCK端子は同期クロック入力	SMR GM	ビット1 CKE1	ビット0 CKE0	説明	0	0	0	SCK端子は入出力ポート	1	SCK端子はクロック出力	1	0	0	SCK端子はLow出力固定	1	SCK端子はクロック出力	1	0	0	SCK端子はHigh出力固定	1	SCK端子はクロック出力	0	送信終了割込み (TEI) 要求を禁止	1	送信終了割込み (TEI) 要求を許可	0	マルチプロセッサ割込みを禁止 (通常の受信動作を行う)	1	マルチプロセッサ割込みを許可	0	受信データフル割込み (RXI) 要求、受信エラー割込み (ERI) 要求を禁止	1	受信データフル割込み (RXI) 要求、受信エラー割込み (ERI) 要求を許可	0	送信データエンpty割込み (TXI) 要求を禁止	1	送信データエンpty割込み (TXI) 要求を許可
TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0																																																																																			
初期値: R/W:	0 R/W	0 R/W	0 R/W	0 R/W	0 R/W	0 R/W	0 R/W																																																																																			
0	受信動作を禁止																																																																																									
1	受信動作を許可																																																																																									
0	送信動作を禁止																																																																																									
1	送信動作を許可																																																																																									
ビット1 CKE1	ビット0 CKE0	クロックの選択、出力の許可																																																																																								
0	0	調歩同期式モード	内部クロック / SCK端子は入出力ポート																																																																																							
		クロック同期式モード	内部クロック / SCK端子は同期クロック出力																																																																																							
	1	調歩同期式モード	内部クロック / SCK端子はクロック出力																																																																																							
		クロック同期式モード	内部クロック / SCK端子は同期クロック出力																																																																																							
1	0	調歩同期式モード	外部クロック / SCK端子はクロック入力																																																																																							
		クロック同期式モード	外部クロック / SCK端子は同期クロック入力																																																																																							
	1	調歩同期式モード	外部クロック / SCK端子はクロック入力																																																																																							
		クロック同期式モード	外部クロック / SCK端子は同期クロック入力																																																																																							
SMR GM	ビット1 CKE1	ビット0 CKE0	説明																																																																																							
0	0	0	SCK端子は入出力ポート																																																																																							
		1	SCK端子はクロック出力																																																																																							
1	0	0	SCK端子はLow出力固定																																																																																							
		1	SCK端子はクロック出力																																																																																							
	1	0	0	SCK端子はHigh出力固定																																																																																						
			1	SCK端子はクロック出力																																																																																						
0	送信終了割込み (TEI) 要求を禁止																																																																																									
1	送信終了割込み (TEI) 要求を許可																																																																																									
0	マルチプロセッサ割込みを禁止 (通常の受信動作を行う)																																																																																									
1	マルチプロセッサ割込みを許可																																																																																									
0	受信データフル割込み (RXI) 要求、受信エラー割込み (ERI) 要求を禁止																																																																																									
1	受信データフル割込み (RXI) 要求、受信エラー割込み (ERI) 要求を許可																																																																																									
0	送信データエンpty割込み (TXI) 要求を禁止																																																																																									
1	送信データエンpty割込み (TXI) 要求を許可																																																																																									

TDR	トランスミットデータレジスタ	H'FFFB3	SCI0					
ビット:	7	6	5	4	3	2	1	0
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
初期値:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
シリアル送信データを格納								



SSR シリアルステータスレジスタ	H'FFFB4	SCI0																								
ビット:      7            6            5            4            3            2            1            0 <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;">TDRE</td> <td style="width: 10%;">RDRF</td> <td style="width: 10%;">ORER</td> <td style="width: 10%;">FER/ERS</td> <td style="width: 10%;">PER</td> <td style="width: 10%;">TEND</td> <td style="width: 10%;">MPB</td> <td style="width: 10%;">MPBT</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">R/(W)*1</td> <td style="text-align: center;">R/(W)*1</td> <td style="text-align: center;">R/(W)*1</td> <td style="text-align: center;">R/(W)*1</td> <td style="text-align: center;">R/(W)*1</td> <td style="text-align: center;">R</td> <td style="text-align: center;">R</td> <td style="text-align: center;">R/W</td> </tr> </table>	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT	1	0	0	0	0	1	0	0	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R/W		
TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT																			
1	0	0	0	0	1	0	0																			
R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R/W																			
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">マルチプロセッサビットトランスファ</td> </tr> <tr> <td style="text-align: center;">0</td> <td>マルチプロセッサビットが0のデータを送信</td> </tr> <tr> <td style="text-align: center;">1</td> <td>マルチプロセッサビットが1のデータを送信</td> </tr> </table>		マルチプロセッサビットトランスファ		0	マルチプロセッサビットが0のデータを送信	1	マルチプロセッサビットが1のデータを送信																		
マルチプロセッサビットトランスファ																										
0	マルチプロセッサビットが0のデータを送信																									
1	マルチプロセッサビットが1のデータを送信																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">マルチプロセッサビット</td> </tr> <tr> <td style="text-align: center;">0</td> <td>マルチプロセッサビットが0のデータを受信</td> </tr> <tr> <td style="text-align: center;">1</td> <td>マルチプロセッサビットが1のデータを受信</td> </tr> </table>		マルチプロセッサビット		0	マルチプロセッサビットが0のデータを受信	1	マルチプロセッサビットが1のデータを受信																		
マルチプロセッサビット																										
0	マルチプロセッサビットが0のデータを受信																									
1	マルチプロセッサビットが1のデータを受信																									
	トランスミットエンド (シリアルコミュニケーションインタフェース時)																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが、0のとき 3.1バイトのシリアル送信キャラクタの最後尾ビットの送信時にTDRE = 1のとき</td> </tr> </table>		0	〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき	1	〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが、0のとき 3.1バイトのシリアル送信キャラクタの最後尾ビットの送信時にTDRE = 1のとき																				
0	〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき																									
1	〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが、0のとき 3.1バイトのシリアル送信キャラクタの最後尾ビットの送信時にTDRE = 1のとき																									
	トランスミットエンド (スマートカードインタフェース時)																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが、0かつFER/ERSビットが0のとき 3.1バイトのシリアルキャラクタ送信終了 2.5 etu*2後 (GM = 0のとき) / 1.0 etu後 (GM = 1のとき) にTDRE = 1かつFER/ERSビット = 0 (正常送信) のとき</td> </tr> </table>		0	〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき	1	〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが、0かつFER/ERSビットが0のとき 3.1バイトのシリアルキャラクタ送信終了 2.5 etu*2後 (GM = 0のとき) / 1.0 etu後 (GM = 1のとき) にTDRE = 1かつFER/ERSビット = 0 (正常送信) のとき																				
0	〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき																									
1	〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが、0かつFER/ERSビットが0のとき 3.1バイトのシリアルキャラクタ送信終了 2.5 etu*2後 (GM = 0のとき) / 1.0 etu後 (GM = 1のとき) にTDRE = 1かつFER/ERSビット = 0 (正常送信) のとき																									
	パリティエラー																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.リセット、またはスタンバイモード時 2.PER = 1の状態をリードした後、0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 パリティエラーが発生したとき (受信したデータのパリティがSMRのOEビットで設定したパリティと一致しなかったとき)</td> </tr> </table>		0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.PER = 1の状態をリードした後、0をライトしたとき	1	〔セット条件〕 パリティエラーが発生したとき (受信したデータのパリティがSMRのOEビットで設定したパリティと一致しなかったとき)																				
0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.PER = 1の状態をリードした後、0をライトしたとき																									
1	〔セット条件〕 パリティエラーが発生したとき (受信したデータのパリティがSMRのOEビットで設定したパリティと一致しなかったとき)																									
	フレーミングエラー (シリアルコミュニケーションインタフェースのとき)																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.リセット、またはスタンバイモード時 2.FER = 1の状態をリードした後、0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 フレーミングエラーが発生したとき (ストップビットが0の場合)</td> </tr> </table>		0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.FER = 1の状態をリードした後、0をライトしたとき	1	〔セット条件〕 フレーミングエラーが発生したとき (ストップビットが0の場合)																				
0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.FER = 1の状態をリードした後、0をライトしたとき																									
1	〔セット条件〕 フレーミングエラーが発生したとき (ストップビットが0の場合)																									
	エラーシグナルステータス (スマートカードインタフェースのとき)																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.リセット、またはスタンバイモード時 2.ERS = 1の状態をリードした後、0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 エラーシグナルLowを受信したとき</td> </tr> </table>		0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.ERS = 1の状態をリードした後、0をライトしたとき	1	〔セット条件〕 エラーシグナルLowを受信したとき																				
0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.ERS = 1の状態をリードした後、0をライトしたとき																									
1	〔セット条件〕 エラーシグナルLowを受信したとき																									
	オーバランエラー																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.リセット、またはスタンバイモード時 2.ORER = 1の状態をリードした後、0をライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 オーバランエラーが発生したとき (RDRF = 1の状態です次のデータが受信完了したとき)</td> </tr> </table>		0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.ORER = 1の状態をリードした後、0をライトしたとき	1	〔セット条件〕 オーバランエラーが発生したとき (RDRF = 1の状態です次のデータが受信完了したとき)																				
0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.ORER = 1の状態をリードした後、0をライトしたとき																									
1	〔セット条件〕 オーバランエラーが発生したとき (RDRF = 1の状態です次のデータが受信完了したとき)																									
	レシーブデータレジスタフル																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.リセット、またはスタンバイモード時 2.RDRF = 1の状態をリードした後、0をライトしたとき 3.DMACでRDRのデータをリードしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 データが正常に受信され、RSRからRDRヘデータが転送されたとき</td> </tr> </table>		0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.RDRF = 1の状態をリードした後、0をライトしたとき 3.DMACでRDRのデータをリードしたとき	1	〔セット条件〕 データが正常に受信され、RSRからRDRヘデータが転送されたとき																				
0	〔クリア条件〕 1.リセット、またはスタンバイモード時 2.RDRF = 1の状態をリードした後、0をライトしたとき 3.DMACでRDRのデータをリードしたとき																									
1	〔セット条件〕 データが正常に受信され、RSRからRDRヘデータが転送されたとき																									
	トランスミットデータレジスタエンpty																									
	<table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td>〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき</td> </tr> <tr> <td style="text-align: center;">1</td> <td>〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが0のとき 3.TDRからTSRにデータ転送が行われ、TDRにデータライトが可能になったとき</td> </tr> </table>		0	〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき	1	〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが0のとき 3.TDRからTSRにデータ転送が行われ、TDRにデータライトが可能になったとき																				
0	〔クリア条件〕 1.TDRE = 1の状態をリードした後、0をライトしたとき 2.DMACでTDRヘデータをライトしたとき																									
1	〔セット条件〕 1.リセット、またはスタンバイモード時 2.SCRのTEビットが0のとき 3.TDRからTSRにデータ転送が行われ、TDRにデータライトが可能になったとき																									
<p>【注】 *1 フラグをクリアするための0ライトのみ可能です。 *2 etu (Elementary Time Unit) : 1ビットの転送期間</p>																										

RDR レシーブデータレジスタ	H'FFFB5		SCI0					
ビット:	7	6	5	4	3	2	1	0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
 シリアル受信データを格納								

SCMR スマートカードモードレジスタ	H'FFFB6	SCI0																																												
<p>ビット:</p> <table border="1" style="margin-left: 20px; border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">7</td><td style="width: 20px;">6</td><td style="width: 20px;">5</td><td style="width: 20px;">4</td><td style="width: 20px;">3</td><td style="width: 20px;">2</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td>—</td><td>—</td><td>—</td><td>—</td><td>SDIR</td><td>SINV</td><td>—</td><td>SMIF</td> </tr> </table> <p>初期値:</p> <table border="1" style="margin-left: 20px; border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">1</td><td style="width: 20px;">1</td><td style="width: 20px;">1</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td><td style="width: 20px;">0</td><td style="width: 20px;">1</td><td style="width: 20px;">0</td> </tr> <tr> <td>—</td><td>—</td><td>—</td><td>—</td><td>R/W</td><td>R/W</td><td>—</td><td>R/W</td> </tr> </table>	7	6	5	4	3	2	1	0	—	—	—	—	SDIR	SINV	—	SMIF	1	1	1	1	0	0	1	0	—	—	—	—	R/W	R/W	—	R/W	<div style="margin-left: 100px;"> <p>スマートカードインタフェースモードセレクト</p> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">0</td><td>スマートカードインタフェース機能を禁止 (初期値)</td> </tr> <tr> <td>1</td><td>スマートカードインタフェース機能を許可</td> </tr> </table> </div> <div style="margin-left: 100px; margin-top: 20px;"> <p>スマートカードデータインバート</p> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">0</td><td>TDRの内容をそのまま送信 (初期値) 受信したデータをそのままRDRに格納</td> </tr> <tr> <td>1</td><td>TDRの内容を反転してデータを送信 受信したデータを反転してRDRに格納</td> </tr> </table> </div> <div style="margin-left: 100px; margin-top: 20px;"> <p>スマートカードデータ転送方向</p> <table border="1" style="border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">0</td><td>TDRの内容をLSBファーストとして送信 (初期値) 受信したデータをLSBファーストとしてRDRに格納</td> </tr> <tr> <td>1</td><td>TDRの内容をMSBファーストとして送信 受信したデータをMSBファーストとしてRDRに格納</td> </tr> </table> </div>		0	スマートカードインタフェース機能を禁止 (初期値)	1	スマートカードインタフェース機能を許可	0	TDRの内容をそのまま送信 (初期値) 受信したデータをそのままRDRに格納	1	TDRの内容を反転してデータを送信 受信したデータを反転してRDRに格納	0	TDRの内容をLSBファーストとして送信 (初期値) 受信したデータをLSBファーストとしてRDRに格納	1	TDRの内容をMSBファーストとして送信 受信したデータをMSBファーストとしてRDRに格納
7	6	5	4	3	2	1	0																																							
—	—	—	—	SDIR	SINV	—	SMIF																																							
1	1	1	1	0	0	1	0																																							
—	—	—	—	R/W	R/W	—	R/W																																							
0	スマートカードインタフェース機能を禁止 (初期値)																																													
1	スマートカードインタフェース機能を許可																																													
0	TDRの内容をそのまま送信 (初期値) 受信したデータをそのままRDRに格納																																													
1	TDRの内容を反転してデータを送信 受信したデータを反転してRDRに格納																																													
0	TDRの内容をLSBファーストとして送信 (初期値) 受信したデータをLSBファーストとしてRDRに格納																																													
1	TDRの内容をMSBファーストとして送信 受信したデータをMSBファーストとしてRDRに格納																																													

SMR シリアルモードレジスタ	H'FFFB8	SCI1								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1"> <tr> <td>C<math>\bar{A}</math></td> <td>CHR</td> <td>PE</td> <td>O<math>\bar{E}</math></td> <td>STOP</td> <td>MP</td> <td>CKS1</td> <td>CKS0</td> </tr> </table> <p>初期値:      0          0          0          0          0          0          0          0  R/W:      R/W      R/W      R/W      R/W      R/W      R/W      R/W</p> <p>機能はSCI0と同じです。</p>			C $\bar{A}$	CHR	PE	O $\bar{E}$	STOP	MP	CKS1	CKS0
C $\bar{A}$	CHR	PE	O $\bar{E}$	STOP	MP	CKS1	CKS0			
BRR ビットレートレジスタ	H'FFFB9	SCI1								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table> <p>初期値:      1          1          1          1          1          1          1          1  R/W:      R/W      R/W      R/W      R/W      R/W      R/W      R/W</p> <p>機能はSCI0と同じです。</p>										
SCR シリアルコントロールレジスタ	H'FFBBA	SCI1								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1"> <tr> <td>TIE</td> <td>RIE</td> <td>TE</td> <td>RE</td> <td>MPIE</td> <td>TEIE</td> <td>CKE1</td> <td>CKE0</td> </tr> </table> <p>初期値:      0          0          0          0          0          0          0          0  R/W:      R/W      R/W      R/W      R/W      R/W      R/W      R/W</p> <p>機能はSCI0と同じです。</p>			TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0			

TDR トランスミットデータレジスタ	H'FFFBB	SCI1								
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" data-bbox="379 450 1315 501"> <tr> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> </tr> </table> <p>初期値:        1            1            1            1            1            1            1            1</p> <p>R/W:           R/W          R/W          R/W          R/W          R/W          R/W          R/W          R/W</p> <p>機能はSCI0と同じです。</p>										
SSR シリアルステータスレジスタ	H'FFFBC	SCI1								
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" data-bbox="379 981 1315 1025"> <tr> <td style="width: 25px; height: 20px;">TDRE</td> <td style="width: 25px; height: 20px;">RDRF</td> <td style="width: 25px; height: 20px;">ORER</td> <td style="width: 25px; height: 20px;">FER/ERS</td> <td style="width: 25px; height: 20px;">PER</td> <td style="width: 25px; height: 20px;">TEND</td> <td style="width: 25px; height: 20px;">MPB</td> <td style="width: 25px; height: 20px;">MPBT</td> </tr> </table> <p>初期値:        0            0            0            0            0            1            0            0</p> <p>R/W:           R/(W)*      R/(W)*      R/(W)*      R/(W)*      R/(W)*      R            R            R/W</p> <p>機能はSCI0と同じです。</p> <p>【注】* フラグをクリアするための0ライトのみ可能です。</p>			TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT
TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT			
RDR レシーブデータレジスタ	H'FFFBD	SCI1								
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" data-bbox="379 1525 1315 1570"> <tr> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> <td style="width: 25px; height: 20px;"></td> </tr> </table> <p>初期値:        0            0            0            0            0            0            0            0</p> <p>R/W:           R            R            R            R            R            R            R            R</p> <p>機能はSCI0と同じです。</p>										

SCMR スマートカードモードレジスタ	H'FFFBE				SCI1			
ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	SDIR	SINV	—	SMIF
初期値:	1	1	1	1	0	0	1	0
R/W:	—	—	—	—	R/W	R/W	—	R/W
機能はSCI0と同じです。								

SMR シリアルモードレジスタ	H'FFFC0	SCI2								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1"> <tr> <td>C<math>\bar{A}</math></td> <td>CHR</td> <td>PE</td> <td>O<math>\bar{E}</math></td> <td>STOP</td> <td>MP</td> <td>CKS1</td> <td>CKS0</td> </tr> </table> <p>初期値:      0          0          0          0          0          0          0          0  R/W:      R/W      R/W      R/W      R/W      R/W      R/W      R/W</p> <p>機能はSCI0と同じです。</p>			C $\bar{A}$	CHR	PE	O $\bar{E}$	STOP	MP	CKS1	CKS0
C $\bar{A}$	CHR	PE	O $\bar{E}$	STOP	MP	CKS1	CKS0			
BRR ビットレートレジスタ	H'FFFC1	SCI2								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1"> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table> <p>初期値:      1          1          1          1          1          1          1          1  R/W:      R/W      R/W      R/W      R/W      R/W      R/W      R/W</p> <p>機能はSCI0と同じです。</p>										
SCR シリアルコントロールレジスタ	H'FFFC2	SCI2								
<p>ビット:      7          6          5          4          3          2          1          0</p> <table border="1"> <tr> <td>TIE</td> <td>RIE</td> <td>TE</td> <td>RE</td> <td>MPIE</td> <td>TEIE</td> <td>CKE1</td> <td>CKE0</td> </tr> </table> <p>初期値:      0          0          0          0          0          0          0          0  R/W:      R/W      R/W      R/W      R/W      R/W      R/W      R/W</p> <p>機能はSCI0と同じです。</p>			TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0			

TDR トランスミットデータレジスタ	H'FFFC3	SCI2								
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> </tr> </table> <p>初期値:        1            1            1            1            1            1            1            1 R/W:        R/W        R/W        R/W        R/W        R/W        R/W        R/W</p> <p>機能はSCI0と同じです。</p>										
SSR シリアルステータスレジスタ	H'FFFC4	SCI2								
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 12.5%;">TDRE</td> <td style="width: 12.5%;">RDRF</td> <td style="width: 12.5%;">ORER</td> <td style="width: 12.5%;">FER/ERS</td> <td style="width: 12.5%;">PER</td> <td style="width: 12.5%;">TEND</td> <td style="width: 12.5%;">MPB</td> <td style="width: 12.5%;">MPBT</td> </tr> </table> <p>初期値:        1            0            0            0            0            1            0            0 R/W:        R/(W)*    R/(W)*    R/(W)*    R/(W)*    R/(W)*    R        R        R/W</p> <p>機能はSCI0と同じです。</p> <p>【注】* フラグをクリアするための0ライトのみ可能です。</p>			TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT
TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT			
RDR レシーブデータレジスタ	H'FFFC5	SCI2								
<p>ビット:        7            6            5            4            3            2            1            0</p> <table border="1" style="width: 100%; text-align: center;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> </tr> </table> <p>初期値:        0            0            0            0            0            0            0            0 R/W:        R        R        R        R        R        R        R</p> <p>機能はSCI0と同じです。</p>										



SCMR スマートカードモードレジスタ	H'FFFC6				SCI2			
ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	SDIR	SINV	—	SMIF
初期値:	1	1	1	1	0	0	1	0
R/W:	—	—	—	—	R/W	R/W	—	R/W
機能はSCI0と同じです。								

P1DR	ポート1 データレジスタ	H'FFFD0	ポート1					
ビット:	7	6	5	4	3	2	1	0
	P17	P16	P15	P14	P13	P12	P11	P10
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
 ポート1の各端子のデータを格納								
P2DR	ポート2 データレジスタ	H'FFFD1	ポート2					
ビット:	7	6	5	4	3	2	1	0
	P27	P26	P25	P24	P23	P22	P21	P20
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
 ポート2の各端子のデータを格納								
P3DR	ポート3 データレジスタ	H'FFFD2	ポート3					
ビット:	7	6	5	4	3	2	1	0
	P37	P36	P35	P34	P33	P32	P31	P30
初期値:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
 ポート3の各端子のデータを格納								

P4DR	ポート 4 データレジスタ	H'FFFD3	ポート 4								
<p style="margin-left: 40px;">ビット:      7            6            5            4            3            2            1            0</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">P47</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P46</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P45</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P44</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P43</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P42</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P41</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P40</td> </tr> </table> <p style="margin-left: 40px;">初期値:      0            0            0            0            0            0            0            0</p> <p style="margin-left: 40px;">R/W:        R/W        R/W        R/W        R/W        R/W        R/W        R/W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;"> </p> <p style="text-align: center;">ポート4の各端子のデータを格納</p> </div>				P47	P46	P45	P44	P43	P42	P41	P40
P47	P46	P45	P44	P43	P42	P41	P40				
P5DR	ポート 5 データレジスタ	H'FFFD4	ポート 5								
<p style="margin-left: 40px;">ビット:      7            6            5            4            3            2            1            0</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 20px; text-align: center;">—</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P53</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P52</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P51</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P50</td> </tr> </table> <p style="margin-left: 40px;">初期値:      1            1            1            1            0            0            0            0</p> <p style="margin-left: 40px;">R/W:        —            —            —            —            R/W        R/W        R/W        R/W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;"> </p> <p style="text-align: center;">ポート5の各端子のデータを格納</p> </div>				—	—	—	—	P53	P52	P51	P50
—	—	—	—	P53	P52	P51	P50				
P6DR	ポート 6 データレジスタ	H'FFFD5	ポート 6								
<p style="margin-left: 40px;">ビット:      7            6            5            4            3            2            1            0</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">P67</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P66</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P65</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P64</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P63</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P62</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P61</td> <td style="border: 1px solid black; width: 20px; text-align: center;">P60</td> </tr> </table> <p style="margin-left: 40px;">初期値:      1            0            0            0            0            0            0            0</p> <p style="margin-left: 40px;">R/W:        R            R/W        R/W        R/W        R/W        R/W        R/W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;"> </p> <p style="text-align: center;">ポート6の各端子のデータを格納</p> </div>				P67	P66	P65	P64	P63	P62	P61	P60
P67	P66	P65	P64	P63	P62	P61	P60				

P7DR ポート7 データレジスタ	H'FFFD6	ポート7								
<p>ビット:      7            6            5            4            3            2            1            0</p> <table border="1" style="margin: auto;"> <tr> <td style="width: 20px;">P77</td> <td style="width: 20px;">P76</td> <td style="width: 20px;">P75</td> <td style="width: 20px;">P74</td> <td style="width: 20px;">P73</td> <td style="width: 20px;">P72</td> <td style="width: 20px;">P71</td> <td style="width: 20px;">P70</td> </tr> </table> <p>初期値:      ——— *   ——— *   ——— *   ——— *   ——— *   ——— *   ——— *   ——— *</p> <p>R/W:         R            R            R            R            R            R            R</p> <p style="text-align: center;"> </p> <p style="text-align: center;">ポート7の各端子の状態を讀出す</p> <p>【注】* P77-P70端子により決定されます。</p>			P77	P76	P75	P74	P73	P72	P71	P70
P77	P76	P75	P74	P73	P72	P71	P70			
P8DR ポート8 データレジスタ	H'FFFD7	ポート8								
<p>ビット:      7            6            5            4            3            2            1            0</p> <table border="1" style="margin: auto;"> <tr> <td style="width: 20px;">———</td> <td style="width: 20px;">———</td> <td style="width: 20px;">———</td> <td style="width: 20px;">P84</td> <td style="width: 20px;">P83</td> <td style="width: 20px;">P82</td> <td style="width: 20px;">P81</td> <td style="width: 20px;">P80</td> </tr> </table> <p>初期値:      1            1            1            0            0            0            0            0</p> <p>R/W:         ———      ———      ———      R/W      R/W      R/W      R/W      R/W</p> <p style="text-align: center;"> </p> <p style="text-align: center;">ポート8の各端子のデータを格納</p>			———	———	———	P84	P83	P82	P81	P80
———	———	———	P84	P83	P82	P81	P80			

P9DR ポート9 データレジスタ	H'FFFD8	ポート9								
<p style="margin-left: 40px;">ビット:      7            6            5            4            3            2            1            0</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; height: 15px;"></td> <td style="border: 1px solid black; width: 20px; height: 15px;"></td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">P95</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">P94</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">P93</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">P92</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">P91</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">P90</td> </tr> </table> <p style="margin-left: 40px;">初期値:      1            1            0            0            0            0            0            0</p> <p style="margin-left: 40px;">R/W:        ————    ————    R/W        R/W        R/W        R/W        R/W        R/W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;"> </p> <p style="text-align: center;">ポート9の各端子のデータを格納</p> </div>					P95	P94	P93	P92	P91	P90
		P95	P94	P93	P92	P91	P90			
PADR ポートA データレジスタ	H'FFFD9	ポートA								
<p style="margin-left: 40px;">ビット:      7            6            5            4            3            2            1            0</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA7</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA6</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA5</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA4</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA3</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA2</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA1</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PA0</td> </tr> </table> <p style="margin-left: 40px;">初期値:      0            0            0            0            0            0            0            0</p> <p style="margin-left: 40px;">R/W:        R/W        R/W        R/W        R/W        R/W        R/W        R/W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;"> </p> <p style="text-align: center;">ポートAの各端子のデータを格納</p> </div>			PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0			
PBDR ポートB データレジスタ	H'FFFDA	ポートB								
<p style="margin-left: 40px;">ビット:      7            6            5            4            3            2            1            0</p> <table style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB7</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB6</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB5</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB4</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB3</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB2</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB1</td> <td style="border: 1px solid black; width: 20px; height: 15px; text-align: center;">PB0</td> </tr> </table> <p style="margin-left: 40px;">初期値:      0            0            0            0            0            0            0            0</p> <p style="margin-left: 40px;">R/W:        R/W        R/W        R/W        R/W        R/W        R/W        R/W</p> <div style="margin-left: 100px; margin-top: 10px;"> <p style="text-align: center;"> </p> <p style="text-align: center;">ポートBの各端子のデータを格納</p> </div>			PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0			

ADDRA H、L A/D データレジスタ A H、L	H'FFFE0, H'FFFE1	A/D																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1"> <tr> <td>AD9</td><td>AD8</td><td>AD7</td><td>AD6</td><td>AD5</td><td>AD4</td><td>AD3</td><td>AD2</td><td>AD1</td><td>AD0</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>R/W: R R R R R R R R R R R R R R R R</p> <p style="text-align: center;"> <span style="margin-right: 100px;">ADDRAH</span> <span>ADDRAL</span> </p> <hr style="width: 50%; margin: 10px auto;"/> <p style="text-align: center;"> <u>A/D変換データ</u>            A/D変換結果の10ビット            データを格納         </p>			AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—
AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—			
ADDRB H、L A/D データレジスタ B H、L	H'FFFE2, H'FFFE3	A/D																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1"> <tr> <td>AD9</td><td>AD8</td><td>AD7</td><td>AD6</td><td>AD5</td><td>AD4</td><td>AD3</td><td>AD2</td><td>AD1</td><td>AD0</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>R/W: R R R R R R R R R R R R R R R R</p> <p style="text-align: center;"> <span style="margin-right: 100px;">ADDRBH</span> <span>ADDRBL</span> </p> <hr style="width: 50%; margin: 10px auto;"/> <p style="text-align: center;"> <u>A/D変換データ</u>            A/D変換結果の10ビット            データを格納         </p>			AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—
AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—			

ADDRC H、L A/D データレジスタ C H、L	H'FFFE4, H'FFFE5	A/D																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1"> <tr> <td>AD9</td><td>AD8</td><td>AD7</td><td>AD6</td><td>AD5</td><td>AD4</td><td>AD3</td><td>AD2</td><td>AD1</td><td>AD0</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>R/W: R R R R R R R R R R R R R R R R</p> <p style="text-align: center;"> <span style="margin-right: 100px;">ADDRCH</span> <span>ADDRCL</span> </p> <hr style="width: 50%; margin: auto;"/> <p style="text-align: center;"> <u>A/D変換データ</u>              A/D変換結果の10ビット              データを格納         </p>			AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—
AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—			
ADDRD H、L A/D データレジスタ D H、L	H'FFFE6, H'FFFE7	A/D																
<p>ビット: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1"> <tr> <td>AD9</td><td>AD8</td><td>AD7</td><td>AD6</td><td>AD5</td><td>AD4</td><td>AD3</td><td>AD2</td><td>AD1</td><td>AD0</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td> </tr> </table> <p>初期値: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>R/W: R R R R R R R R R R R R R R R R</p> <p style="text-align: center;"> <span style="margin-right: 100px;">ADDRDH</span> <span>ADDRDL</span> </p> <hr style="width: 50%; margin: auto;"/> <p style="text-align: center;"> <u>A/D変換データ</u>              A/D変換結果の10ビット              データを格納         </p>			AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—
AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—			
ADCR A/D コントロールレジスタ	H'FFFE9	A/D																
<p>ビット: 7 6 5 4 3 2 1 0</p> <table border="1"> <tr> <td>TRGE</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td> </tr> </table> <p>初期値: 0 1 1 1 1 1 1 0</p> <p>R/W: R/W — — — — — — R/W</p> <p style="text-align: center;">             トリガイネーブル         </p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">0</td> <td>外部トリガ入力または8ビットタイマのコンペアマッチによるA/D変換の開始を禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>外部トリガ端子 (ADTRG) の立下がりエッジまたは8ビットタイマのコンペアマッチでA/D変換を開始</td> </tr> </table>			TRGE	—	—	—	—	—	—	—	0	外部トリガ入力または8ビットタイマのコンペアマッチによるA/D変換の開始を禁止	1	外部トリガ端子 (ADTRG) の立下がりエッジまたは8ビットタイマのコンペアマッチでA/D変換を開始				
TRGE	—	—	—	—	—	—	—											
0	外部トリガ入力または8ビットタイマのコンペアマッチによるA/D変換の開始を禁止																	
1	外部トリガ端子 (ADTRG) の立下がりエッジまたは8ビットタイマのコンペアマッチでA/D変換を開始																	

ADCSR A/D コントロール/ステータスレジスタ		H'FFFE8		A/D																																															
ビット:	7	6	5	4	3	2	1	0																																											
	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0																																											
初期値:	0	0	0	0	0	0	0	0																																											
R/W:	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W	R/W																																											
				クロックセレクト		チャンネルセレクト																																													
				<table border="1"> <tr> <td>0</td> <td>変換時間 = 134ステート (Max)</td> </tr> <tr> <td>1</td> <td>変換時間 = 70ステート (Max)</td> </tr> </table>		0	変換時間 = 134ステート (Max)	1	変換時間 = 70ステート (Max)	<table border="1"> <thead> <tr> <th>グループ選択</th> <th>チャンネル選択</th> <th colspan="2">説明</th> </tr> <tr> <th>CH2</th> <th>CH1</th> <th>CH0</th> <th>単一モード</th> <th>スキャンモード</th> </tr> </thead> <tbody> <tr> <td rowspan="4">0</td> <td rowspan="2">0</td> <td>0</td> <td>AN0</td> <td>AN0</td> </tr> <tr> <td>1</td> <td>AN1</td> <td>AN0、AN1</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>AN2</td> <td>AN0 - AN2</td> </tr> <tr> <td>1</td> <td>AN3</td> <td>AN0 - AN3</td> </tr> <tr> <td rowspan="4">1</td> <td rowspan="2">0</td> <td>0</td> <td>AN4</td> <td>AN4</td> </tr> <tr> <td>1</td> <td>AN5</td> <td>AN4、AN5</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>AN6</td> <td>AN4 - AN6</td> </tr> <tr> <td>1</td> <td>AN7</td> <td>AN4 - AN7</td> </tr> </tbody> </table>			グループ選択	チャンネル選択	説明		CH2	CH1	CH0	単一モード	スキャンモード	0	0	0	AN0	AN0	1	AN1	AN0、AN1	1	0	AN2	AN0 - AN2	1	AN3	AN0 - AN3	1	0	0	AN4	AN4	1	AN5	AN4、AN5	1	0	AN6	AN4 - AN6	1	AN7	AN4 - AN7
0	変換時間 = 134ステート (Max)																																																		
1	変換時間 = 70ステート (Max)																																																		
グループ選択	チャンネル選択	説明																																																	
CH2	CH1	CH0	単一モード	スキャンモード																																															
0	0	0	AN0	AN0																																															
		1	AN1	AN0、AN1																																															
	1	0	AN2	AN0 - AN2																																															
		1	AN3	AN0 - AN3																																															
1	0	0	AN4	AN4																																															
		1	AN5	AN4、AN5																																															
	1	0	AN6	AN4 - AN6																																															
		1	AN7	AN4 - AN7																																															
				スキャンモード																																															
				<table border="1"> <tr> <td>0</td> <td>単一モード</td> </tr> <tr> <td>1</td> <td>スキャンモード</td> </tr> </table>		0	単一モード	1	スキャンモード																																										
0	単一モード																																																		
1	スキャンモード																																																		
				A/Dスタート																																															
				<table border="1"> <tr> <td>0</td> <td>A/D変換停止</td> </tr> <tr> <td>1</td> <td>(1) 単一モード: A/D変換を開始し変換が終了すると、自動的に0にクリア (2) スキャンモード: A/D変換を開始し、ソフトウェア、リセット、またはスタンバイモードによって0にクリアされるまで、選択されたチャンネルを順次連続変換</td> </tr> </table>		0	A/D変換停止	1	(1) 単一モード: A/D変換を開始し変換が終了すると、自動的に0にクリア (2) スキャンモード: A/D変換を開始し、ソフトウェア、リセット、またはスタンバイモードによって0にクリアされるまで、選択されたチャンネルを順次連続変換																																										
0	A/D変換停止																																																		
1	(1) 単一モード: A/D変換を開始し変換が終了すると、自動的に0にクリア (2) スキャンモード: A/D変換を開始し、ソフトウェア、リセット、またはスタンバイモードによって0にクリアされるまで、選択されたチャンネルを順次連続変換																																																		
				A/Dインタラプトイネーブル																																															
				<table border="1"> <tr> <td>0</td> <td>A/D変換終了による割り込み要求を禁止</td> </tr> <tr> <td>1</td> <td>A/D変換終了による割り込み要求を許可</td> </tr> </table>		0	A/D変換終了による割り込み要求を禁止	1	A/D変換終了による割り込み要求を許可																																										
0	A/D変換終了による割り込み要求を禁止																																																		
1	A/D変換終了による割り込み要求を許可																																																		
				A/Dエンドフラグ																																															
				<table border="1"> <tr> <td>0</td> <td>[クリア条件] (1) ADF = 1の状態ADFフラグをリードした後、ADFフラグに0をライトしたとき (2) ADI割り込みによりDMACが起動されたとき</td> </tr> <tr> <td>1</td> <td>[セット条件] (1) 単一モード: A/Dが終了したとき (2) スキャンモード: 設定されたすべてのチャンネルのA/D変換が終了したとき</td> </tr> </table>		0	[クリア条件] (1) ADF = 1の状態ADFフラグをリードした後、ADFフラグに0をライトしたとき (2) ADI割り込みによりDMACが起動されたとき	1	[セット条件] (1) 単一モード: A/Dが終了したとき (2) スキャンモード: 設定されたすべてのチャンネルのA/D変換が終了したとき																																										
0	[クリア条件] (1) ADF = 1の状態ADFフラグをリードした後、ADFフラグに0をライトしたとき (2) ADI割り込みによりDMACが起動されたとき																																																		
1	[セット条件] (1) 単一モード: A/Dが終了したとき (2) スキャンモード: 設定されたすべてのチャンネルのA/D変換が終了したとき																																																		

【注】\* フラグをクリアにするための0ライトのみ可能です。



## C. I/Oポートブロック図

### C.1 ポート1ブロック図

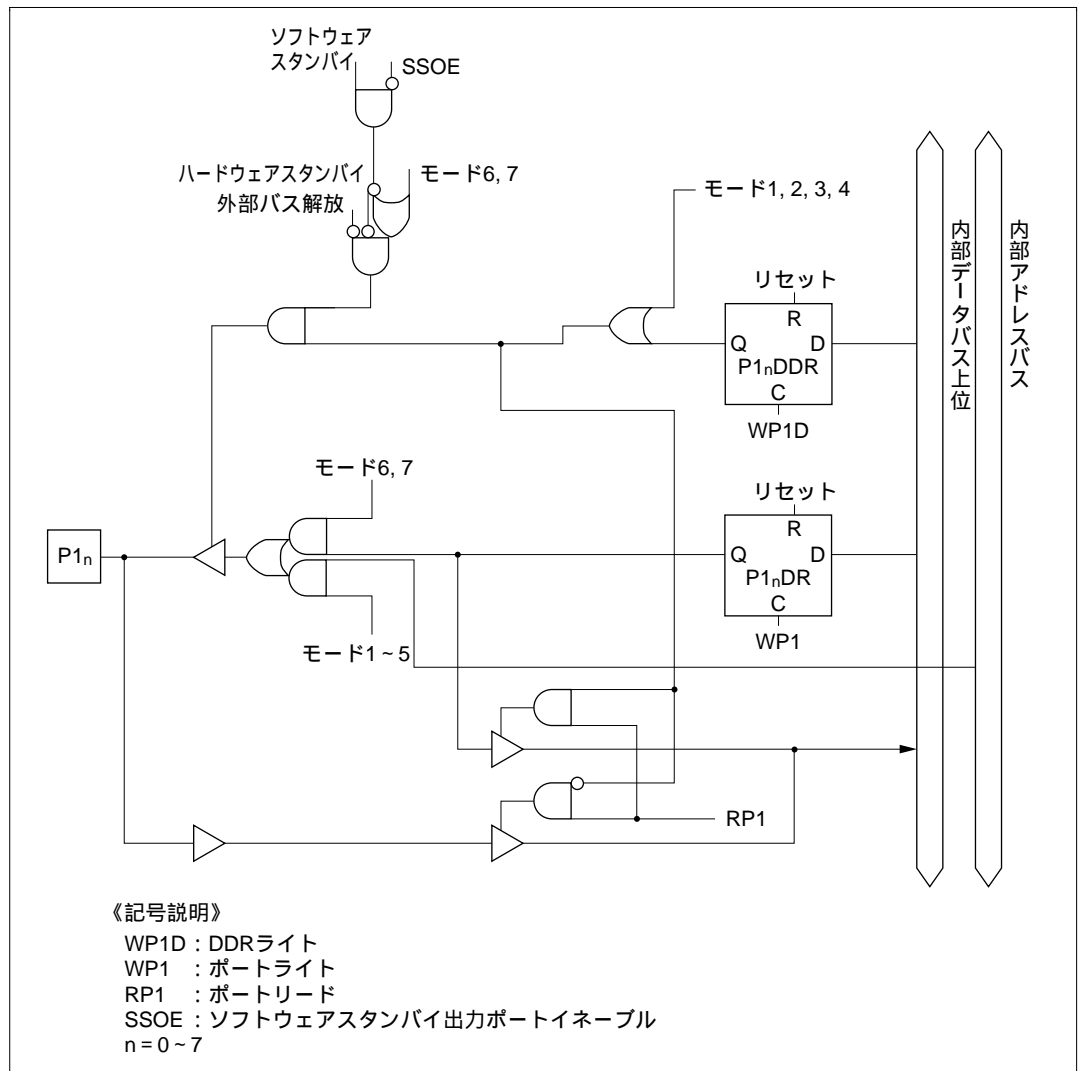


図 C.1 ポート1ブロック図

## C.2 ポート2ブロック図

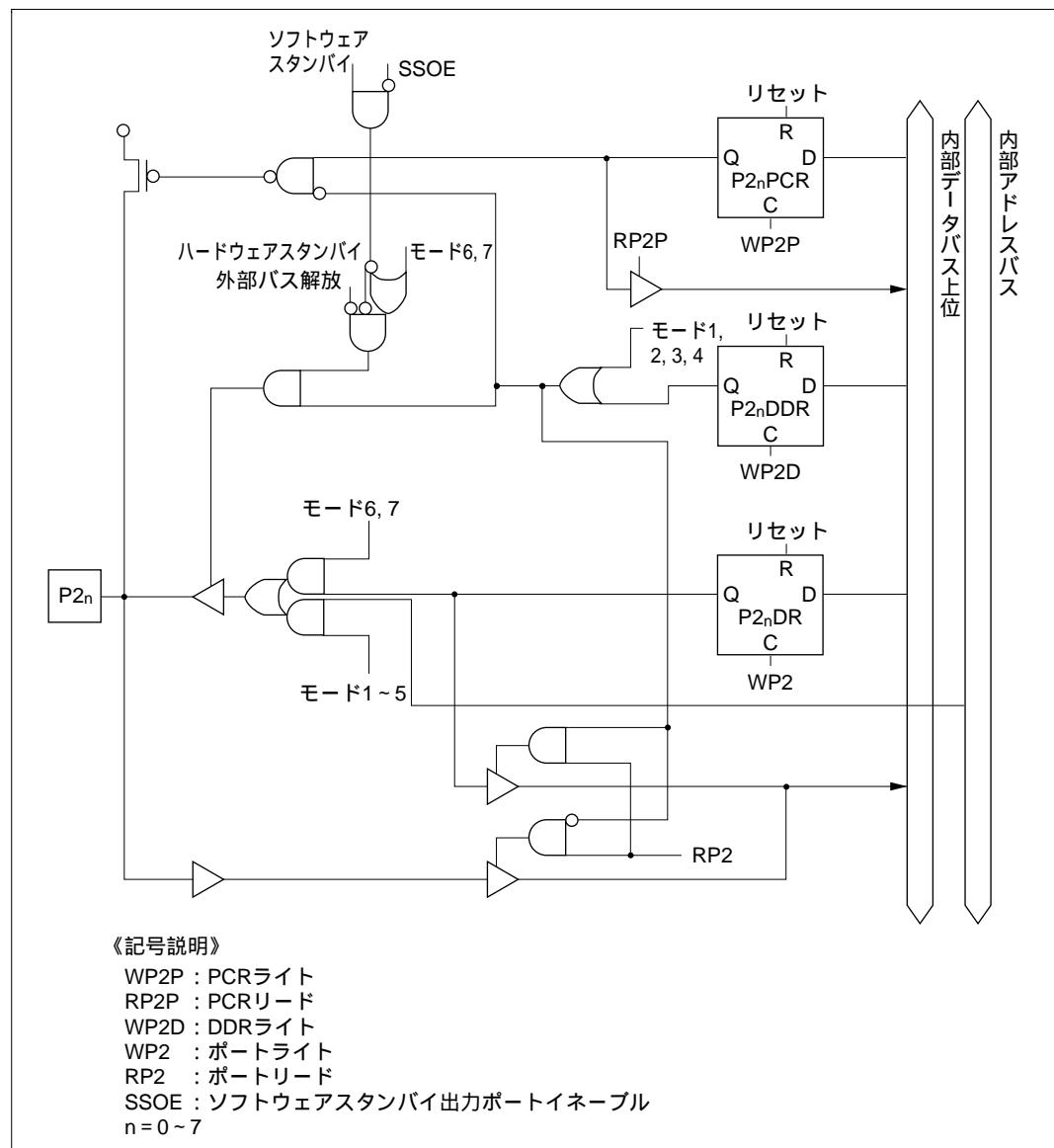


図 C.2 ポート2ブロック図

### C.3 ポート3ブロック図

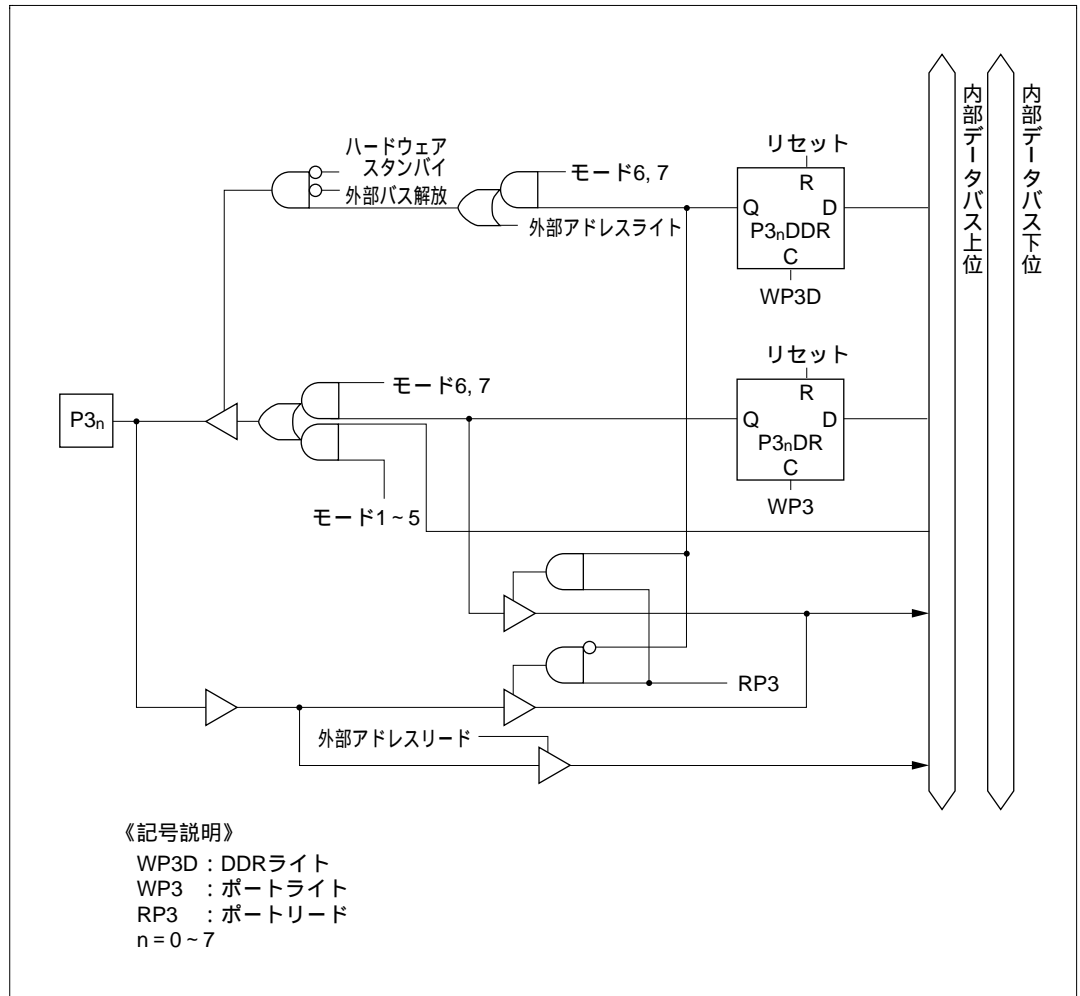


図 C.3 ポート3ブロック図

### C.4 ポート4ブロック図

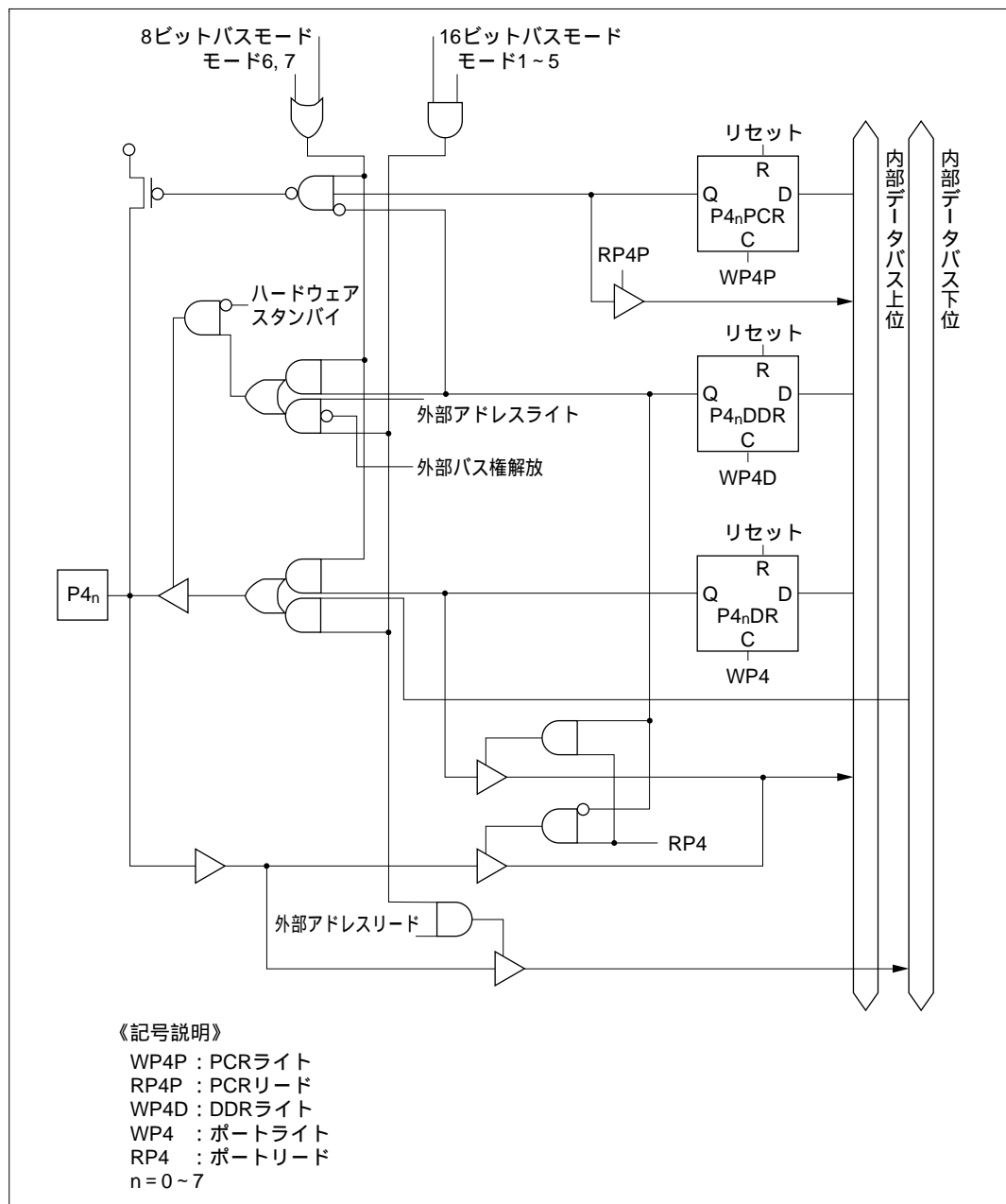


図 C.4 ポート4ブロック図

## C.5 ポート5ブロック図

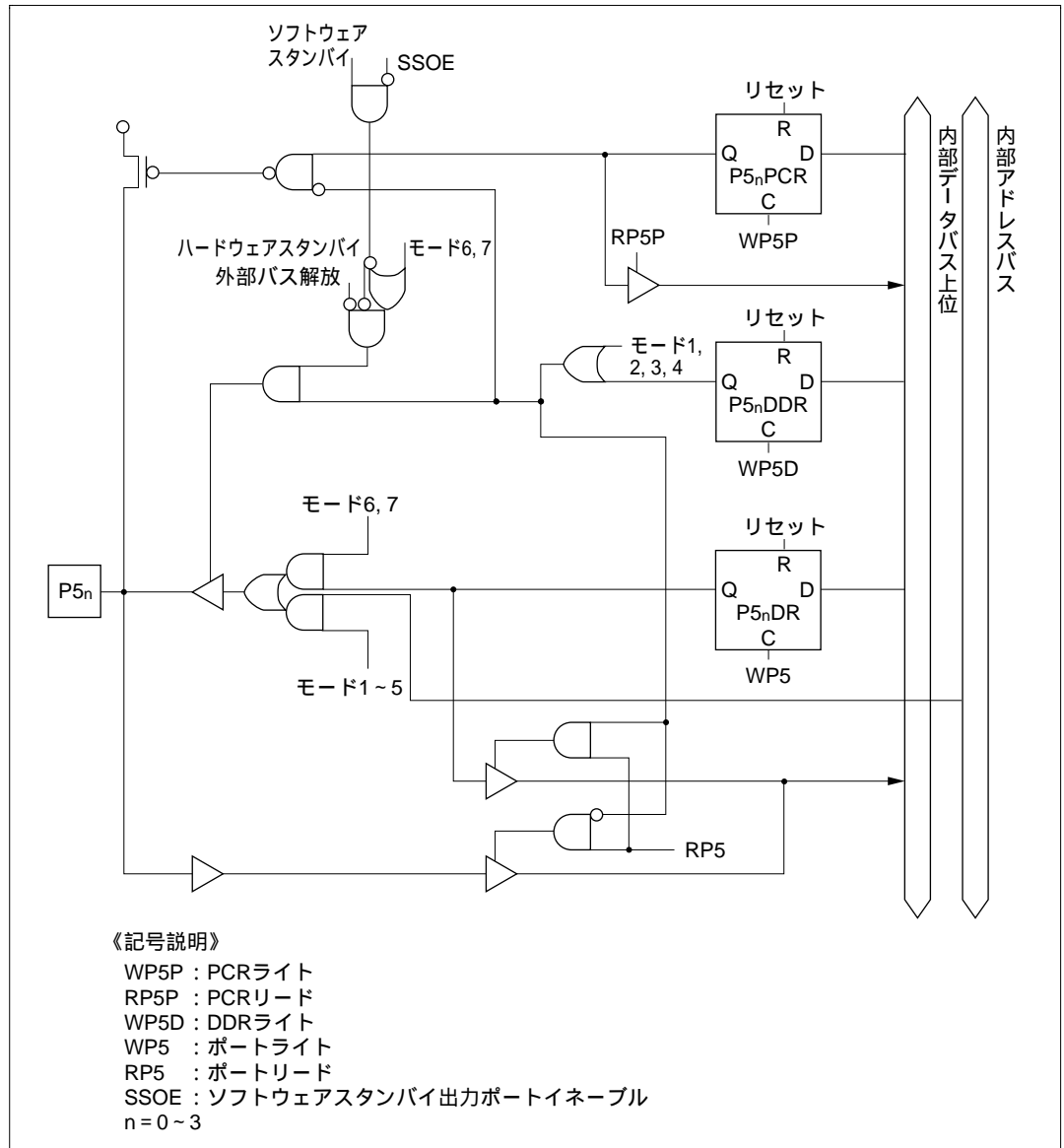


図 C.5 ポート5ブロック図

## C.6 ポート6ブロック図

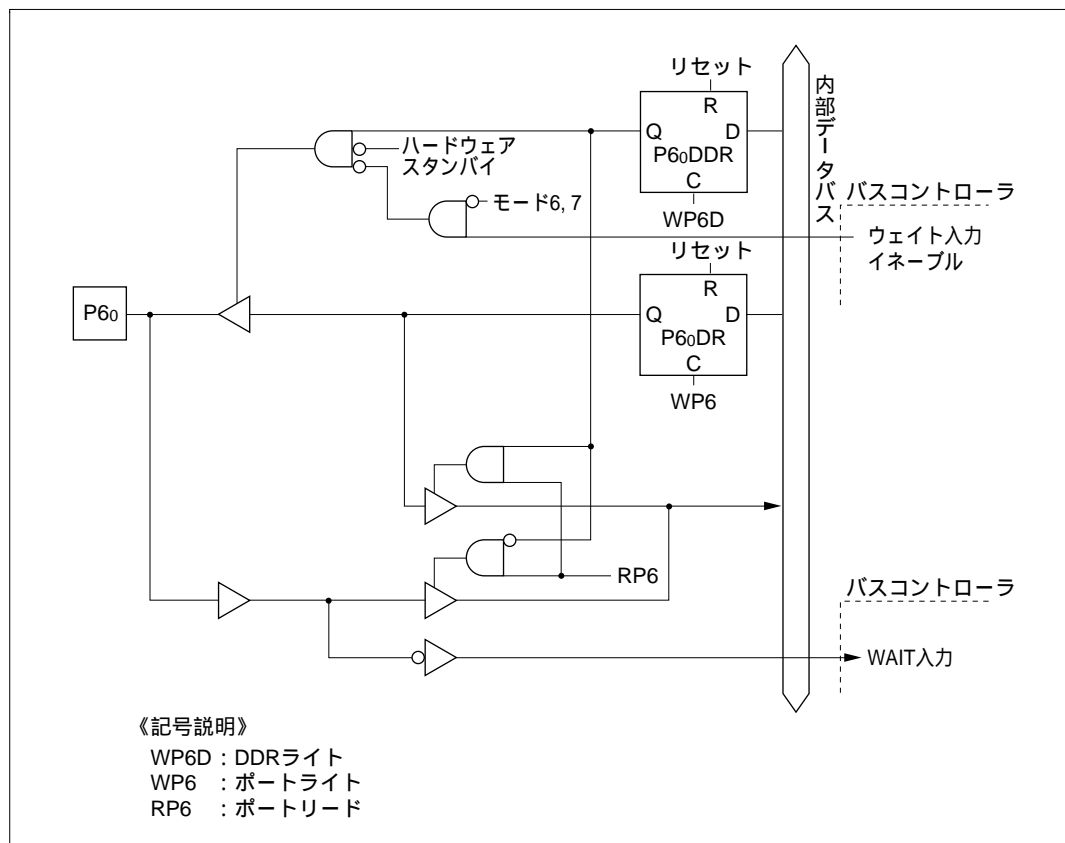


図 C.6 (a) ポート6ブロック図 (P6<sub>0</sub>端子)

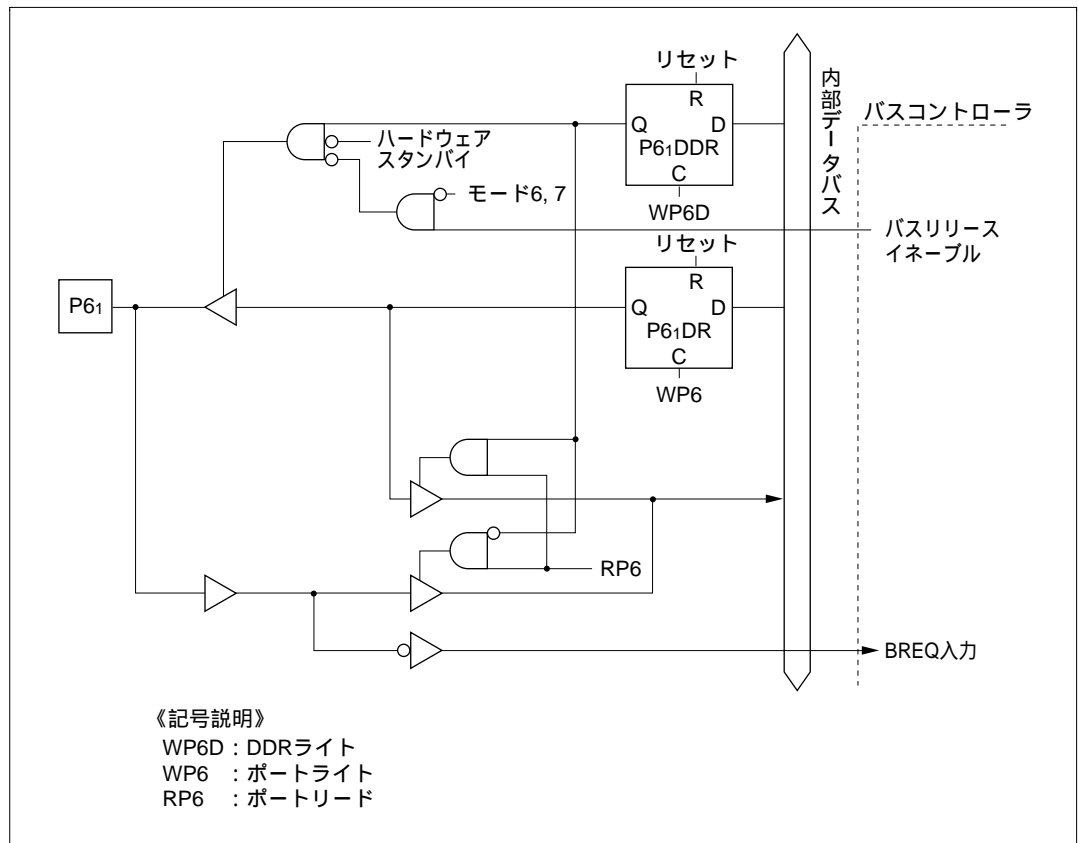


図 C.6 (b) ポート6ブロック図 (P6<sub>1</sub>端子)

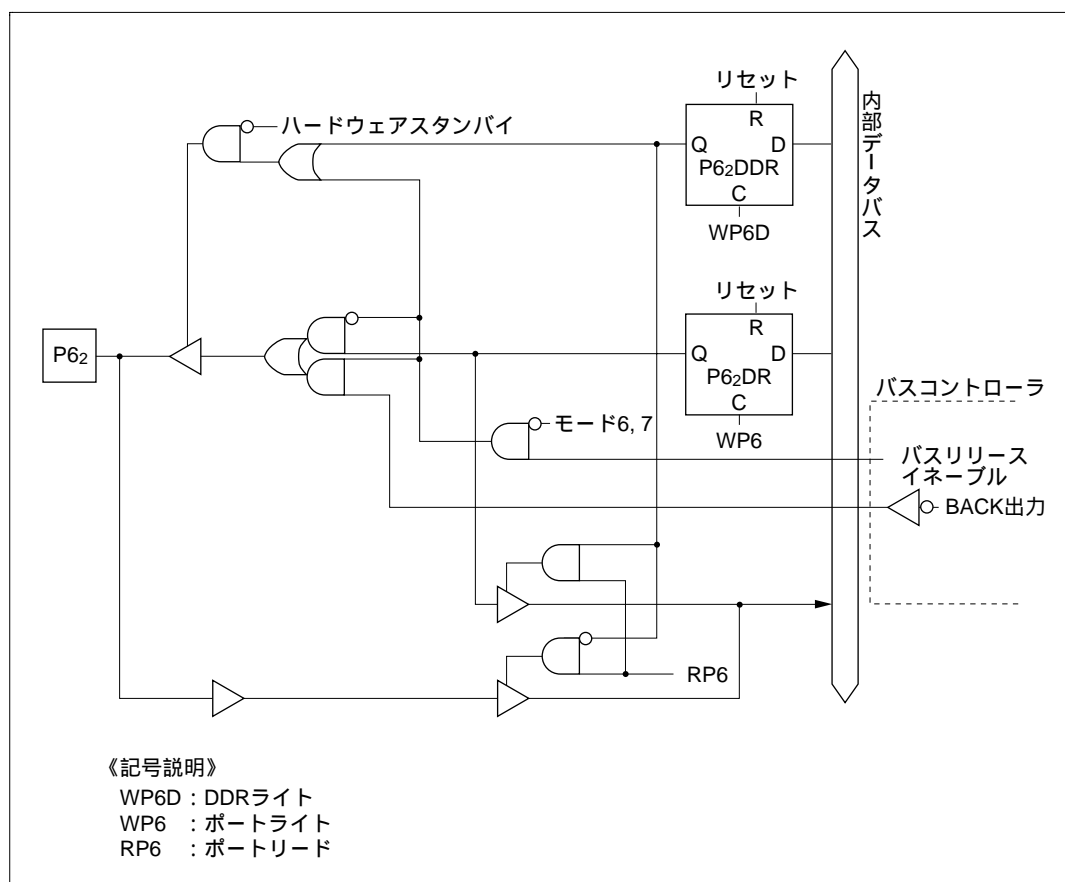


図 C.6 (c) ポート6ブロック図 (P6<sub>2</sub>端子)



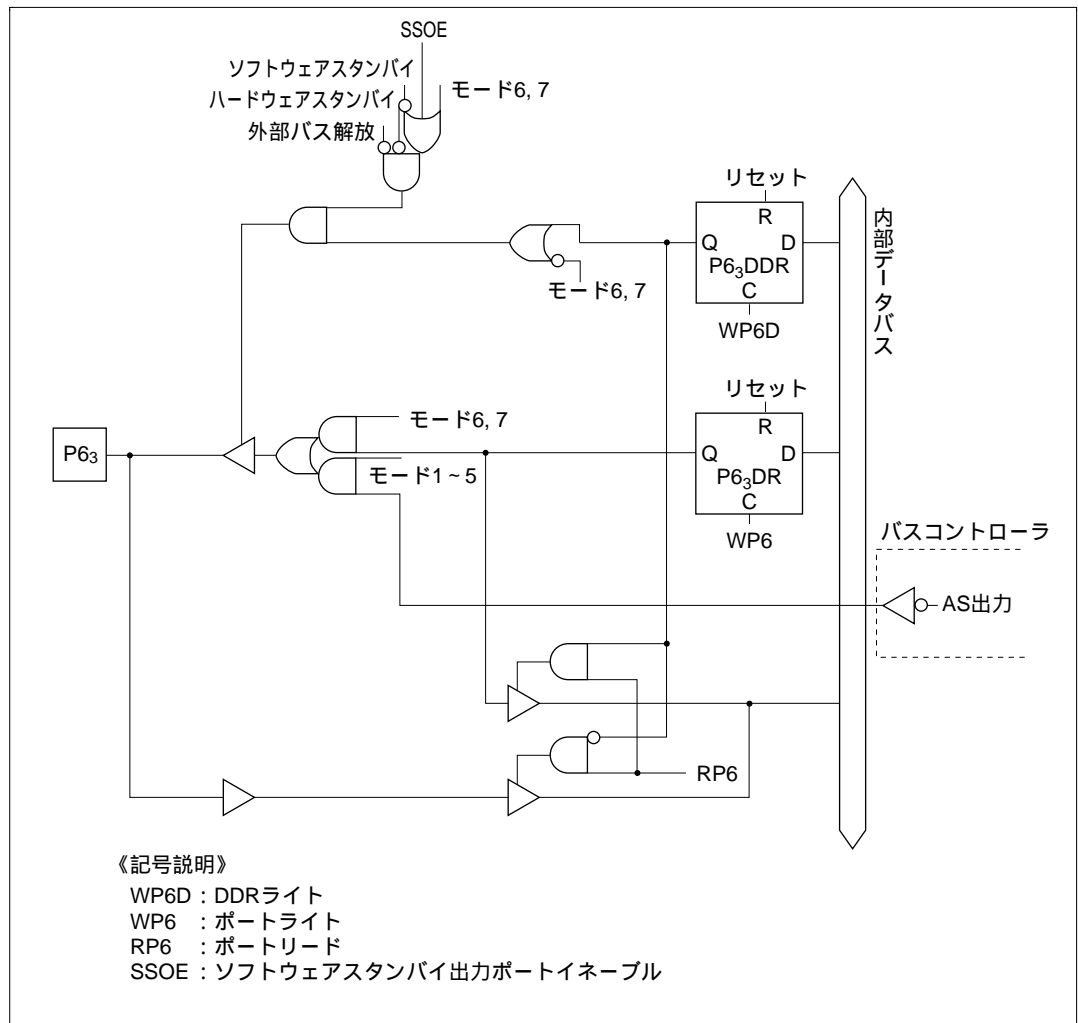


図 C.6 (d) ポート6 ブロック図 (P6<sub>3</sub>端子)

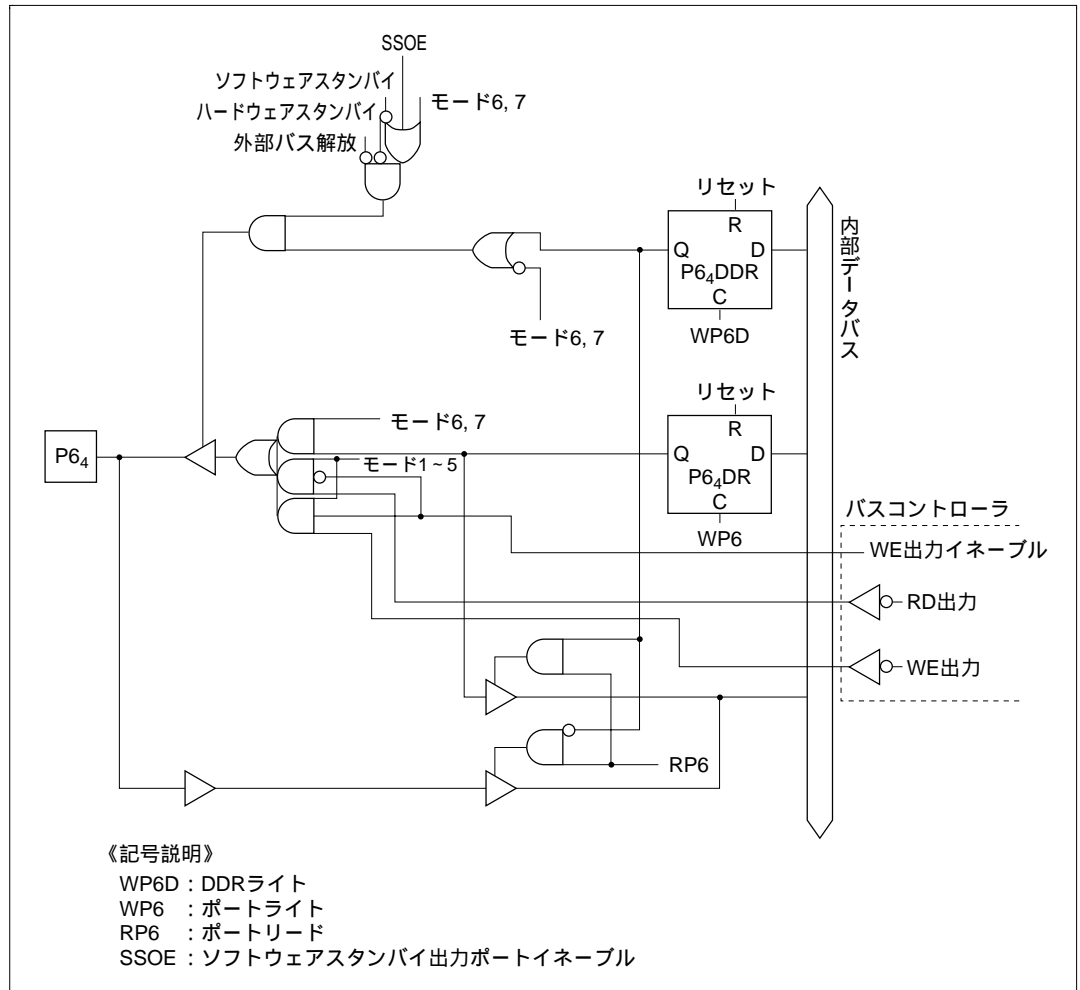


図 C.6 (e) ポート 6 ブロック図 (P6<sub>4</sub>端子)



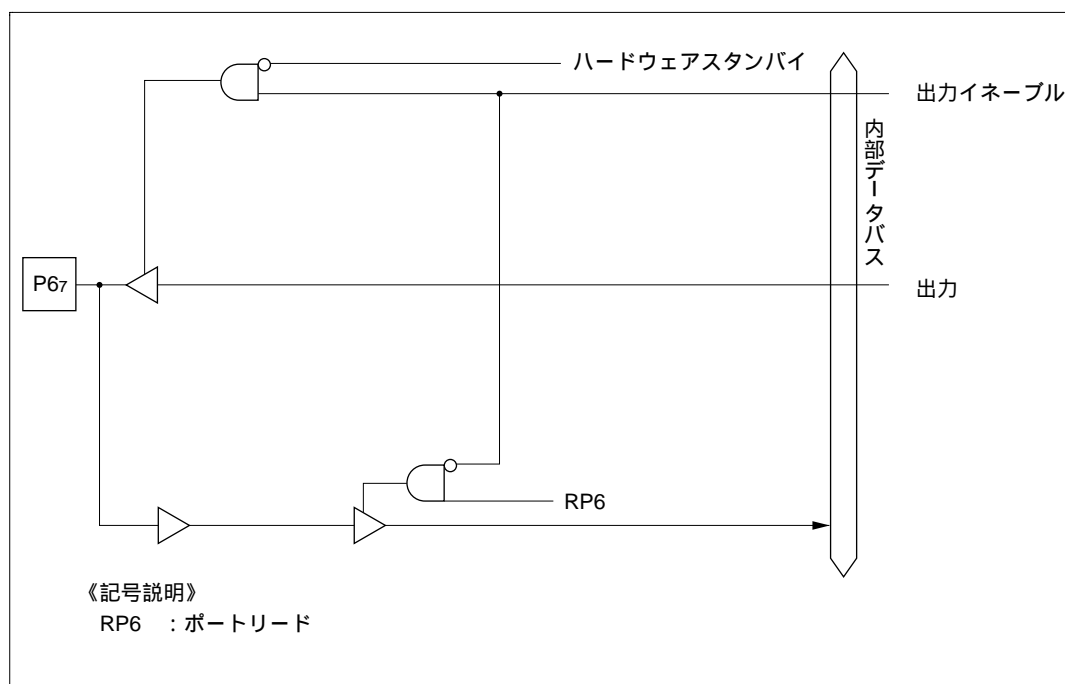


図 C.6 (g) ポート 6 ブロック図 (P6<sub>7</sub>端子)

## C.7 ポート7ブロック図

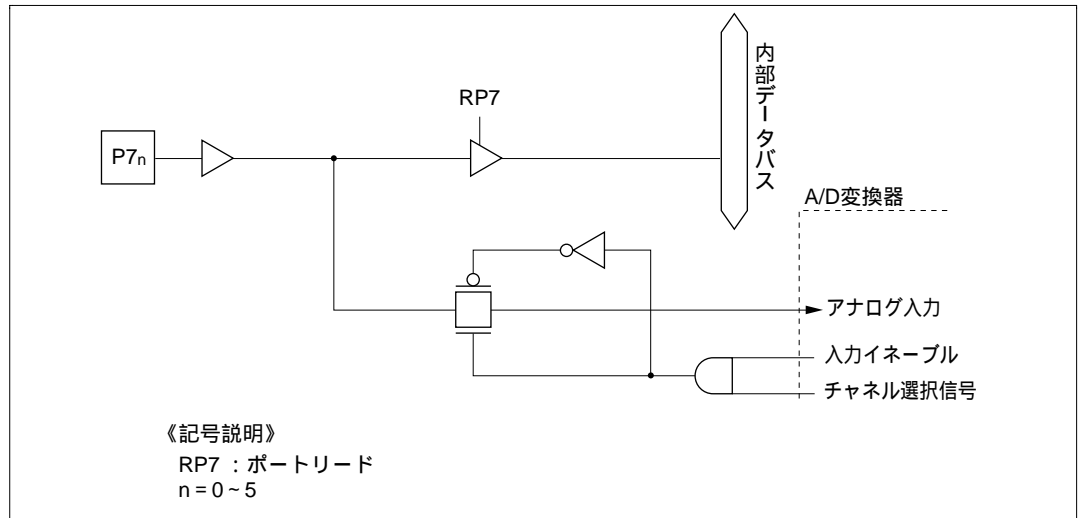


図 C.7 (a) ポート7ブロック図 (P7<sub>0</sub> ~ P7<sub>5</sub>端子)

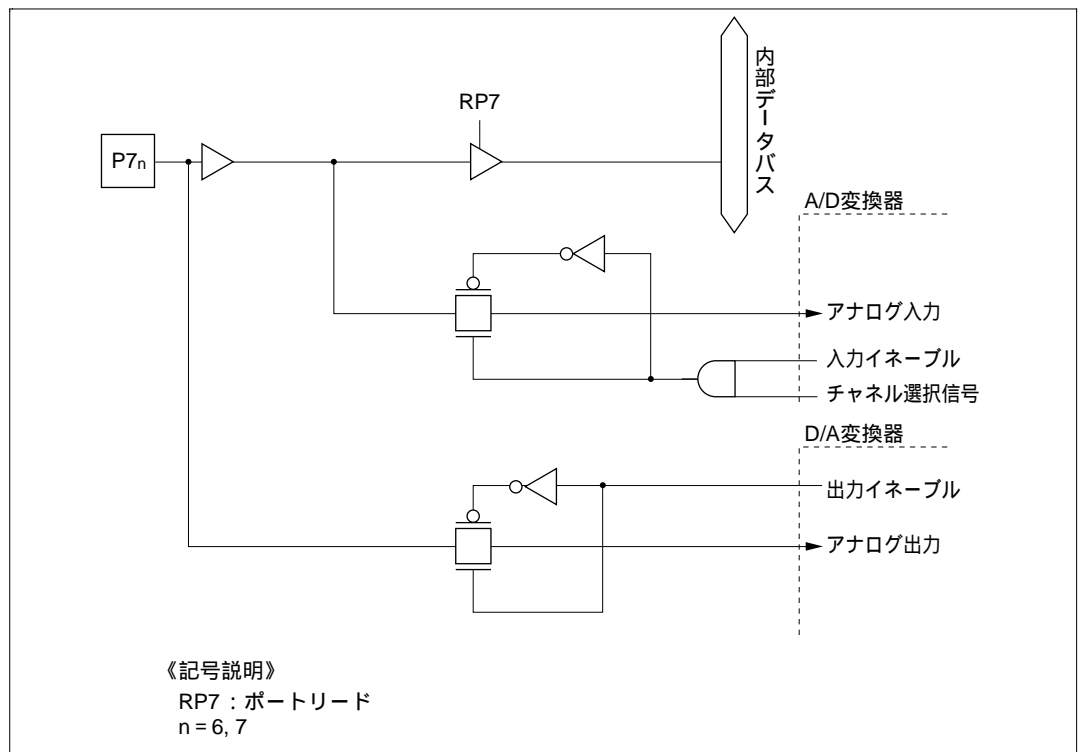
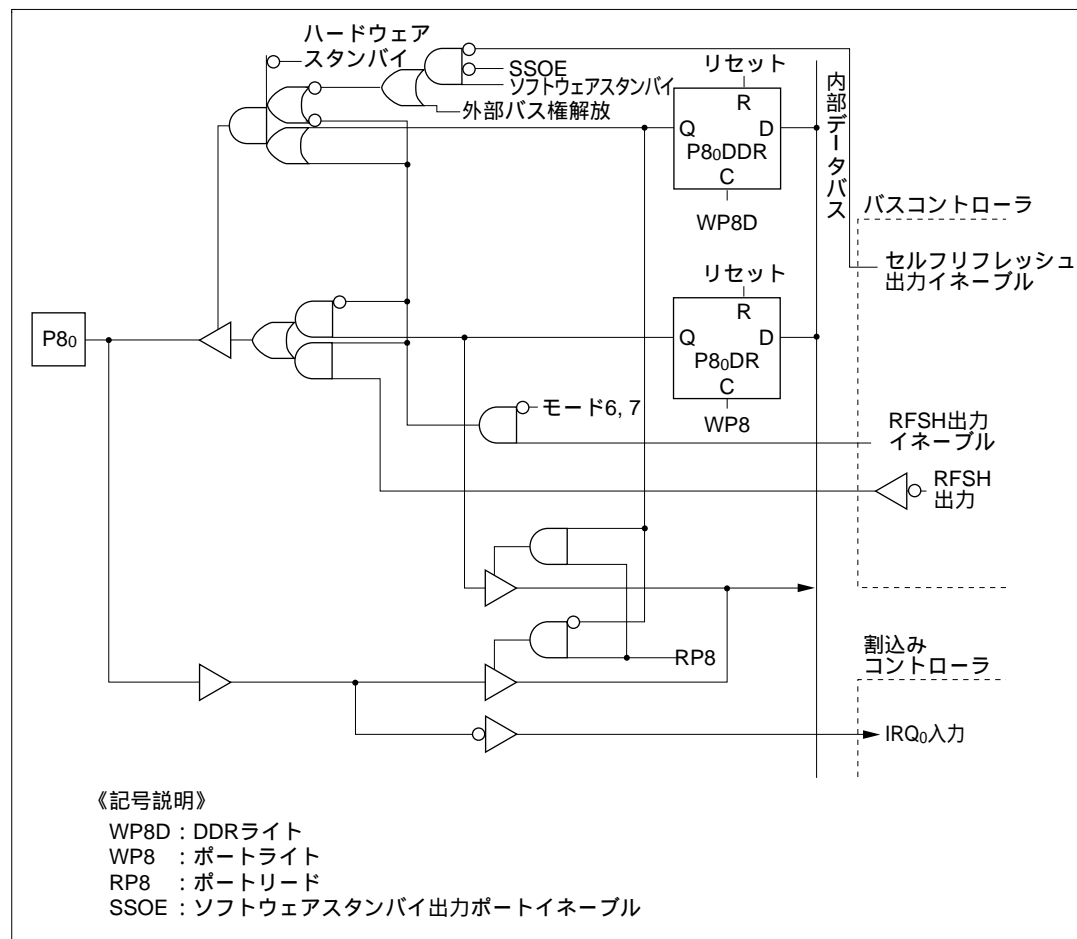


図 C.7 (b) ポート7ブロック図 (P7<sub>6</sub>、P7<sub>7</sub>端子)

## C.8 ポート 8 ブロック図

図 C.8 (a) ポート 8 ブロック図 (P8<sub>0</sub>端子)

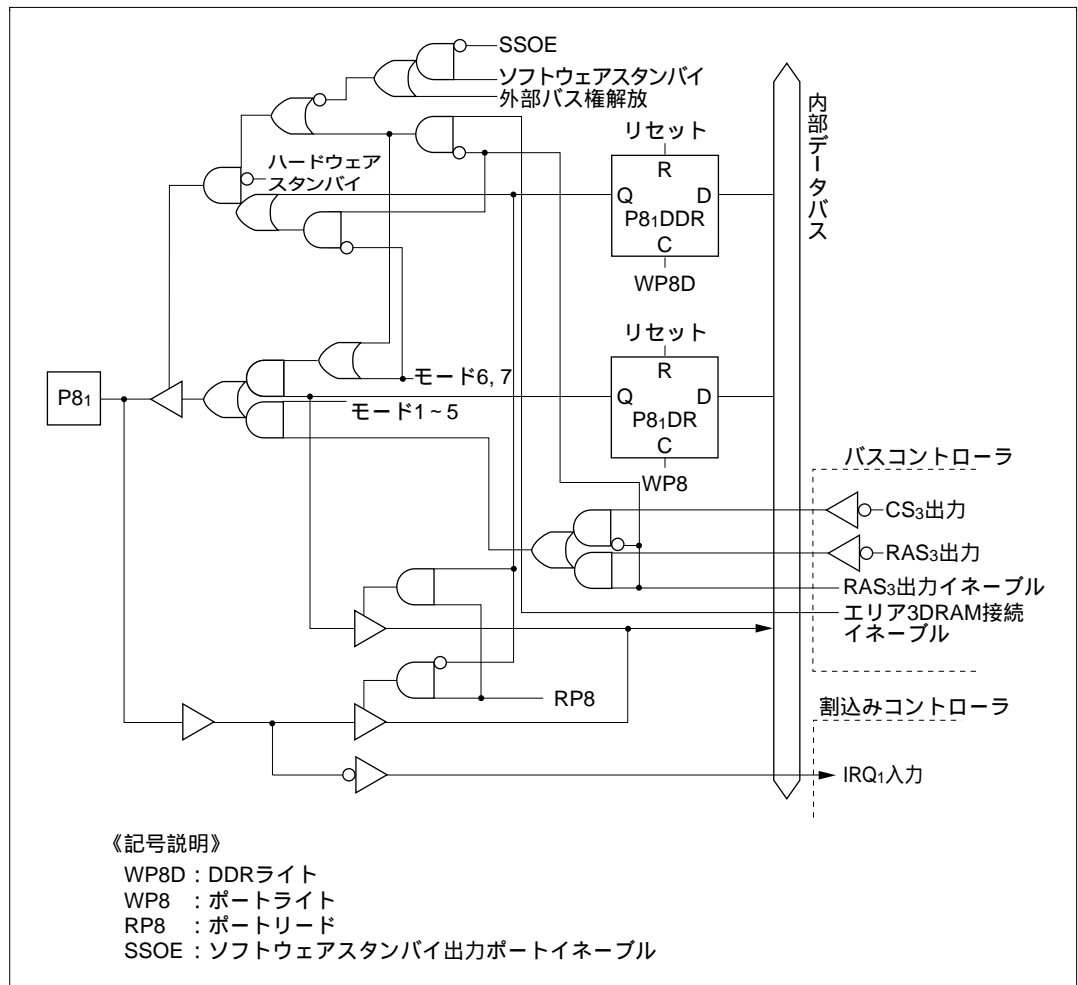


図 C.8 (b) ポート 8 ブロック図 (P8<sub>1</sub>端子)

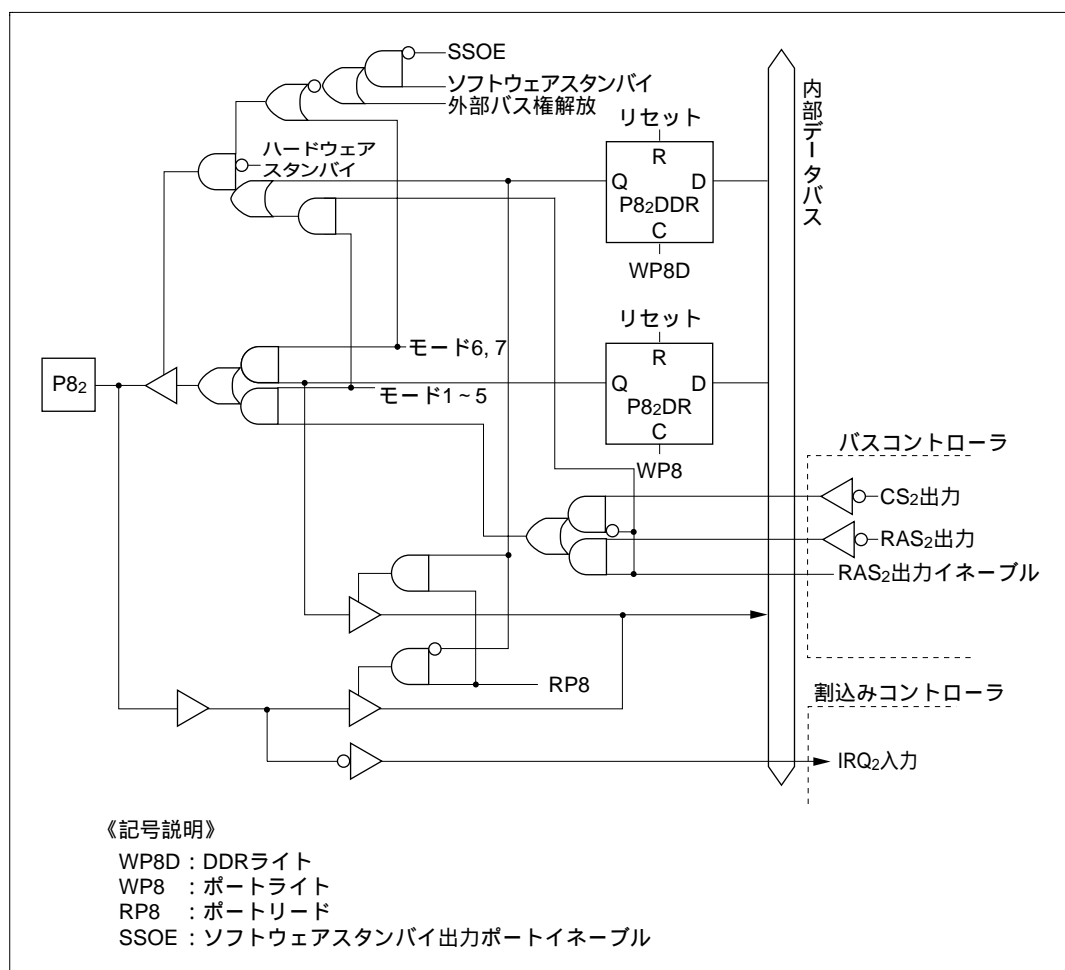


図 C.8 (c) ポート 8 ブロック図 (P8<sub>2</sub>端子)



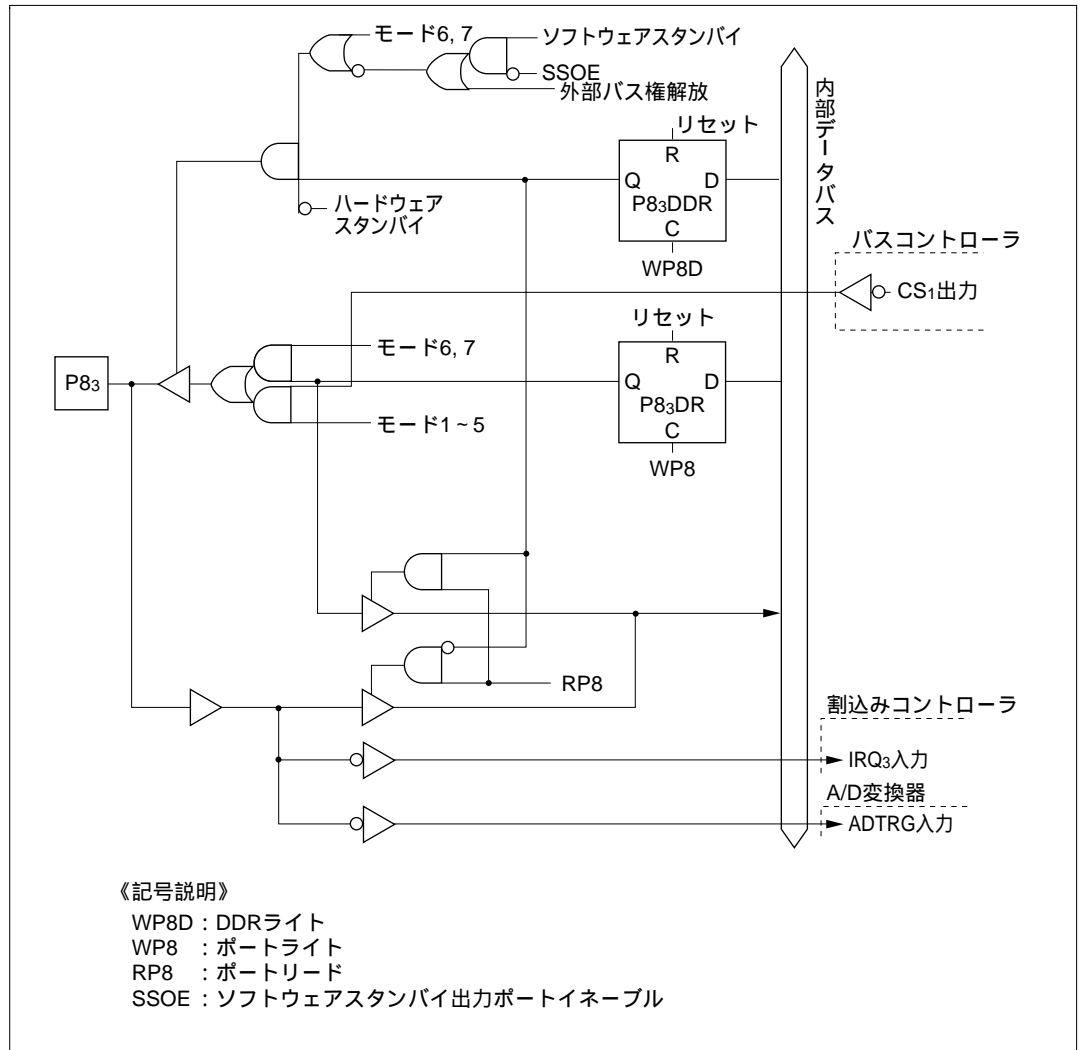


図 C.8 (d) ポート 8 ブロック図 (P8<sub>3</sub>端子)

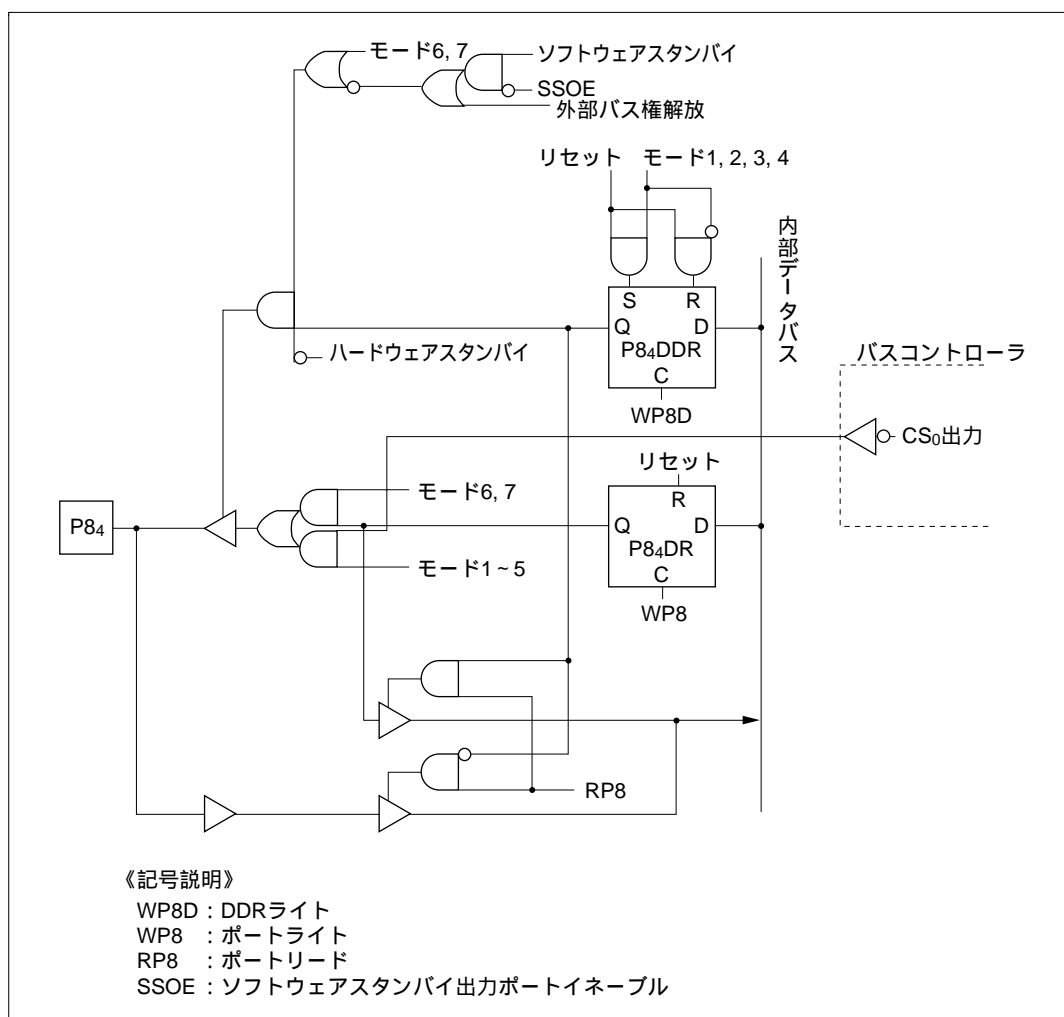


図 C.8 (e) ポート 8 ブロック図 (P8<sub>4</sub>端子)

### C.9 ポート9ブロック図

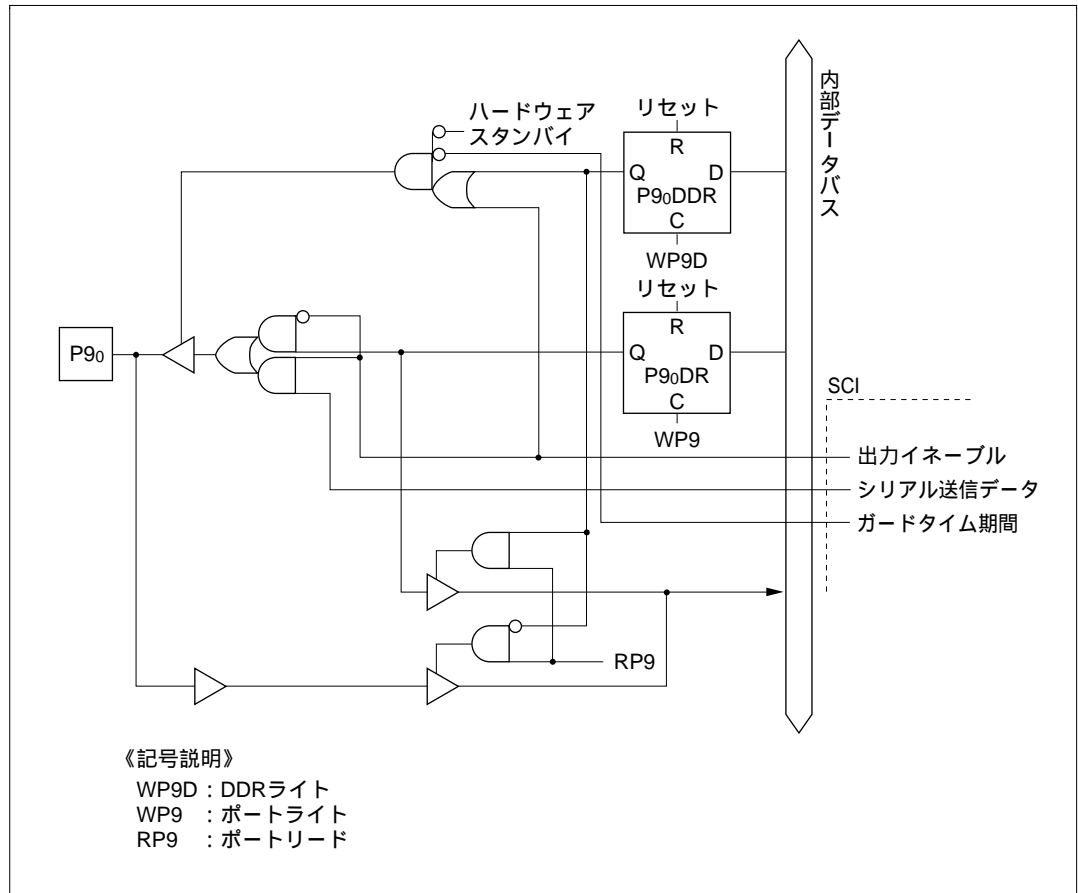


図 C.9 (a) ポート9ブロック図 (P9<sub>0</sub>端子)

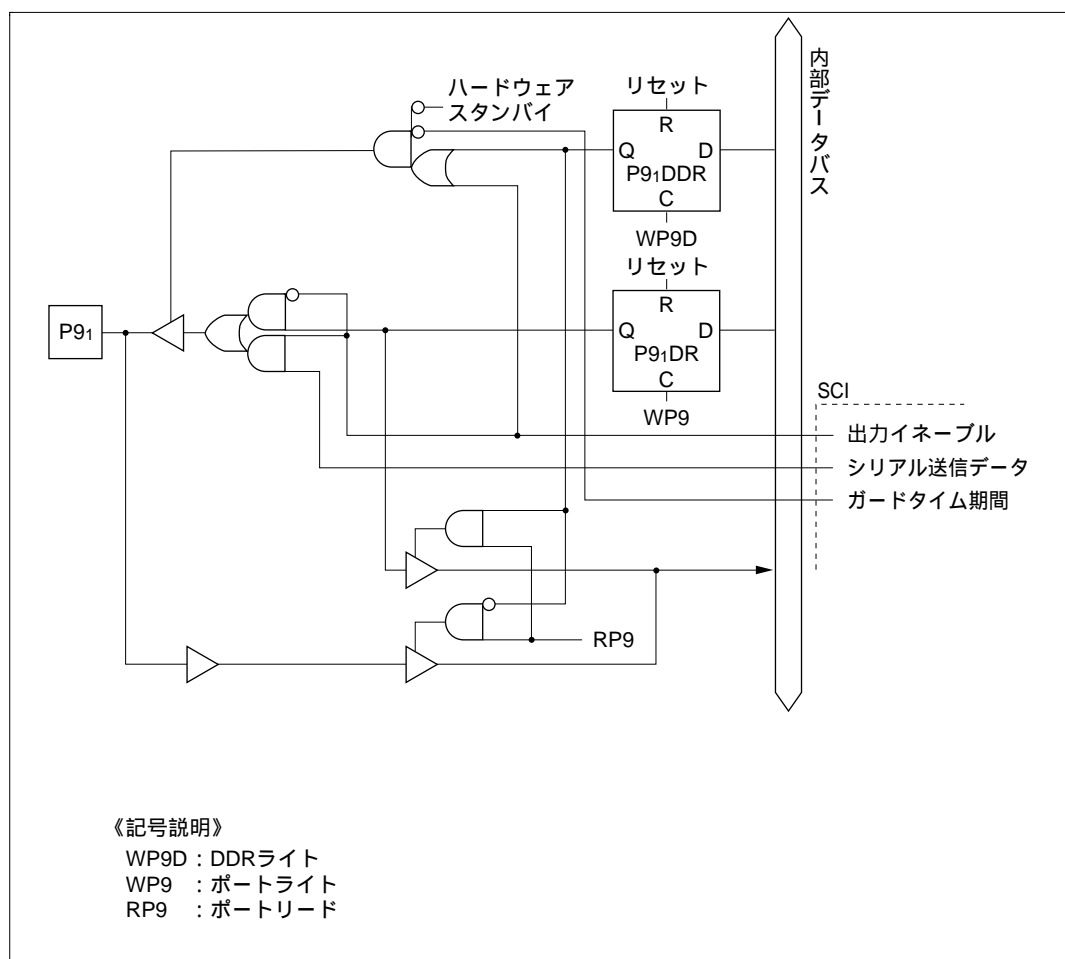


図 C.9 (b) ポート9ブロック図 (P9<sub>1</sub>端子)

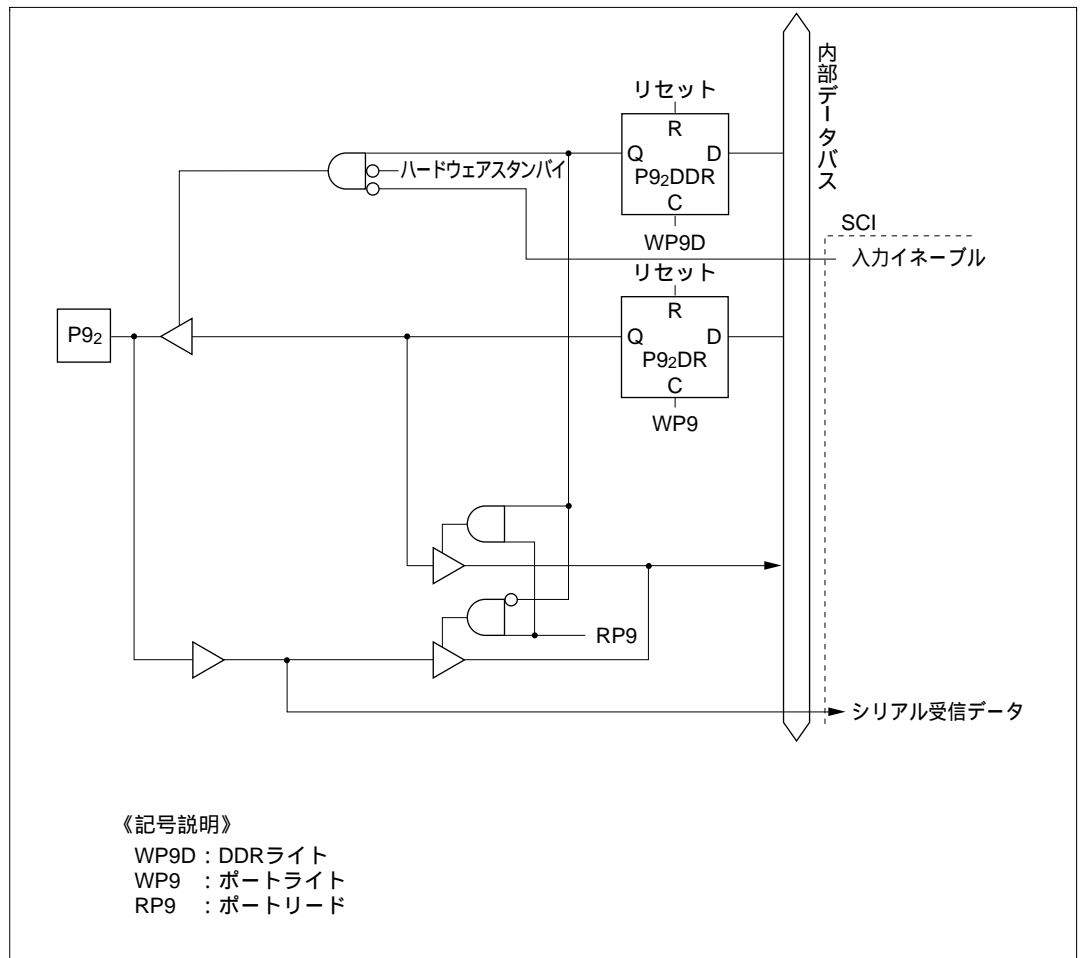


図 C.9 (c) ポート 9 ブロック図 (P9<sub>2</sub>端子)

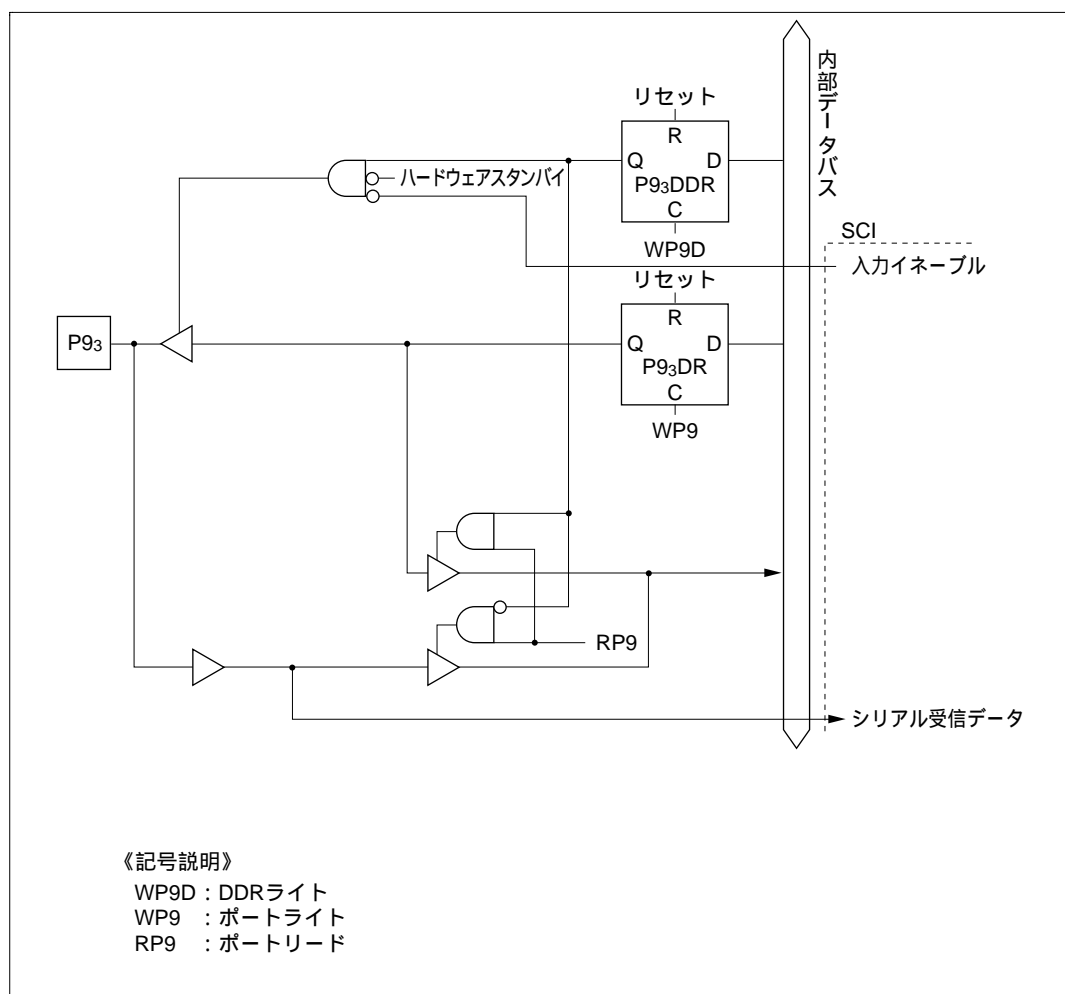


図 C.9 (d) ポート 9 ブロック図 (P9<sub>3</sub>端子)

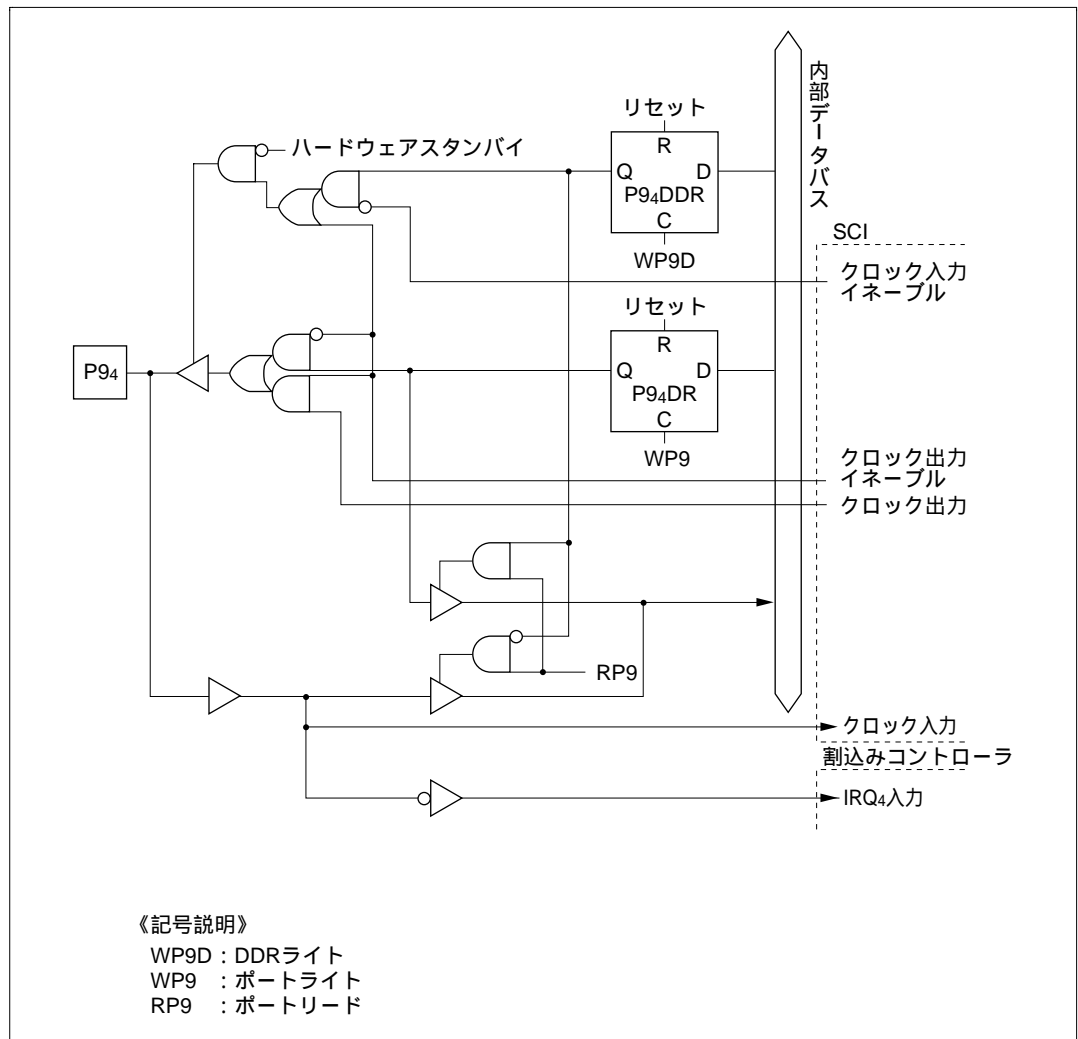


図 C.9 (e) ポート9ブロック図 (P9<sub>4</sub>端子)

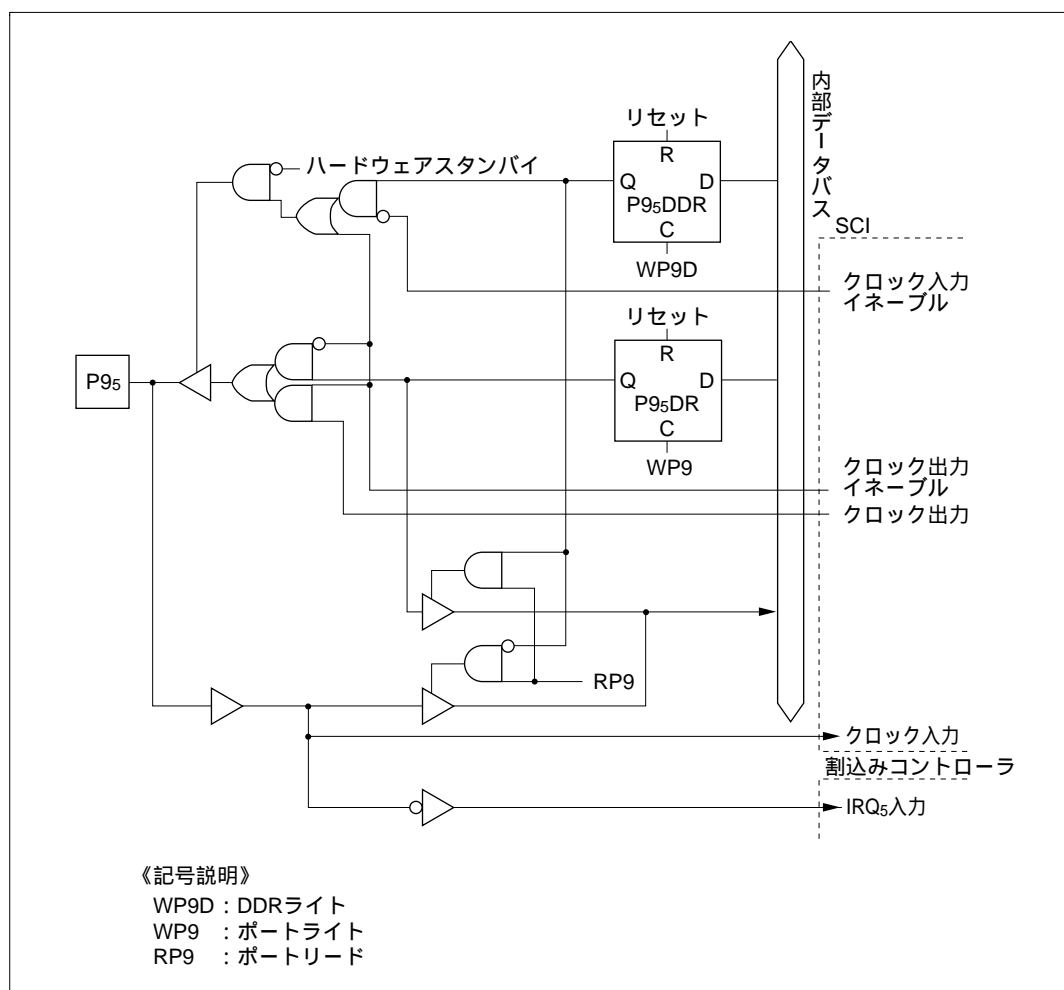


図 C.9 (f) ポート9 ブロック図 (P9<sub>5</sub>端子)



C.10 ポートAブロック図

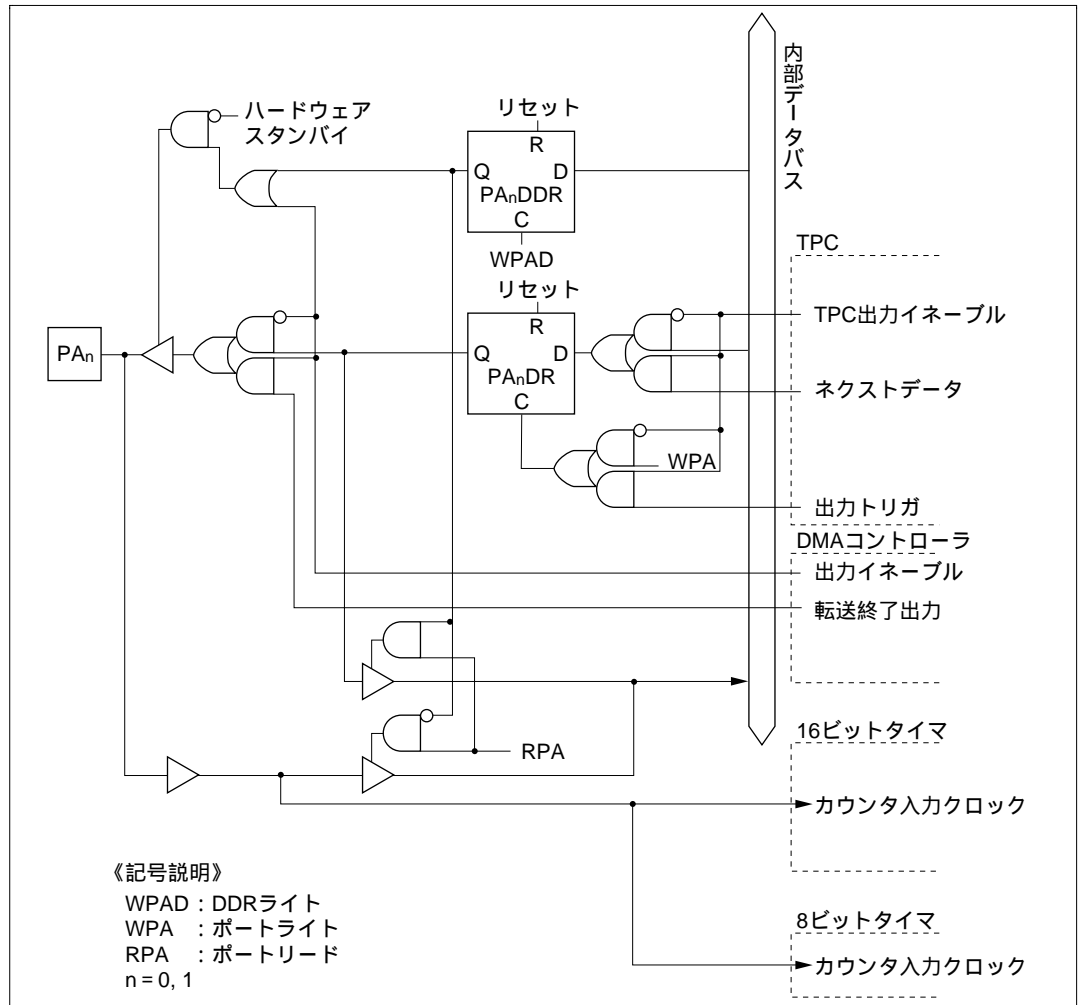


図 C.10 (a) ポートAブロック図 (PA<sub>0</sub>、PA<sub>1</sub>端子)

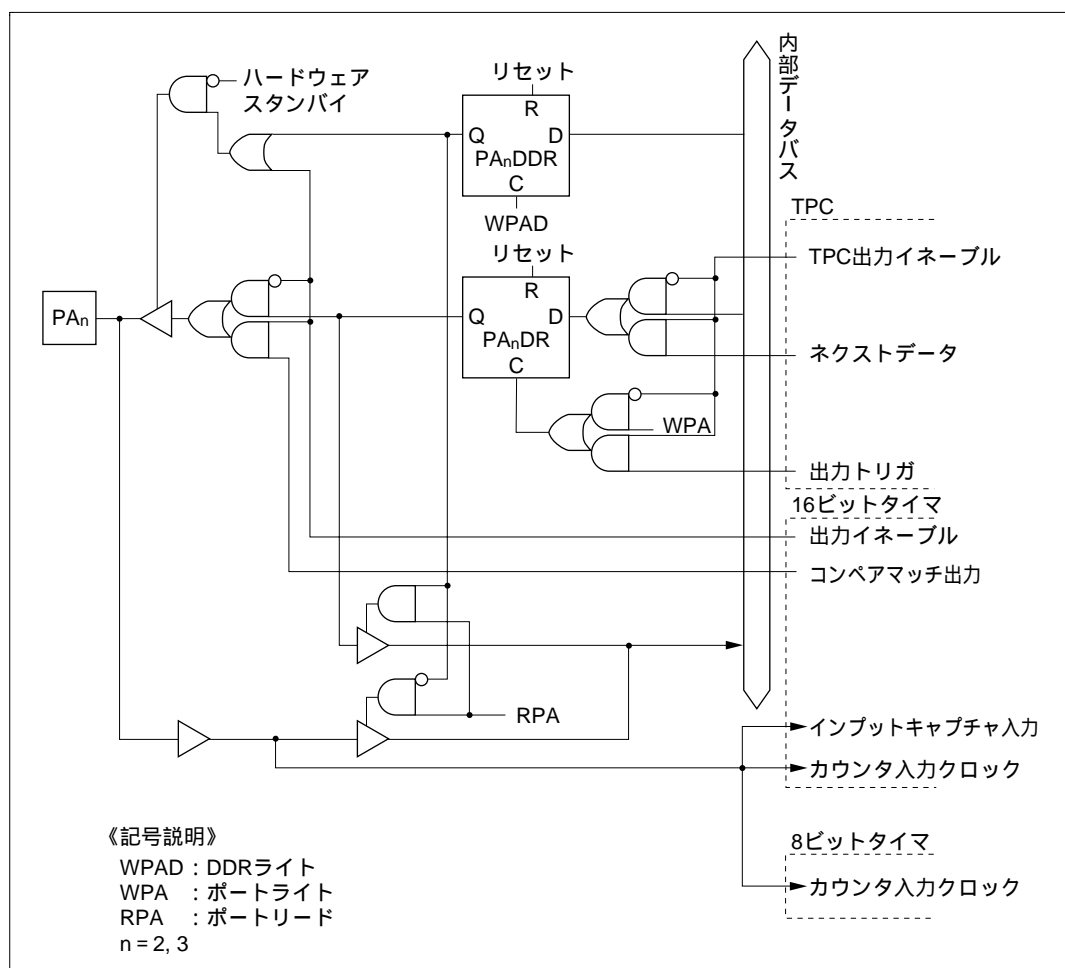


図 C.10 (b) ポート A ブロック図 (PA<sub>2</sub>、PA<sub>3</sub>端子)

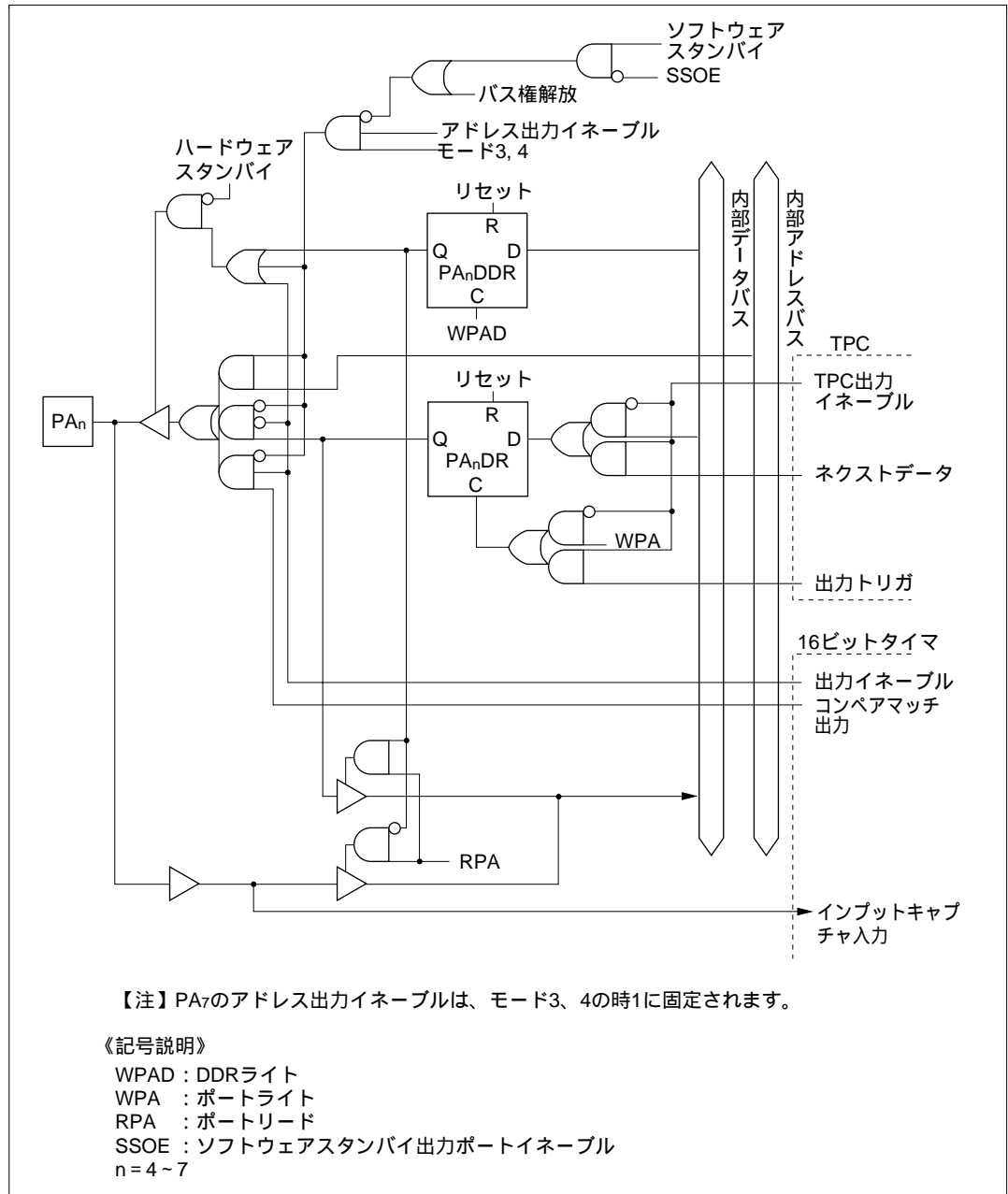


図 C.10 (c) ポート A ブロック図 (PA<sub>4</sub>~PA<sub>7</sub>端子)

### C.11 ポートBブロック図

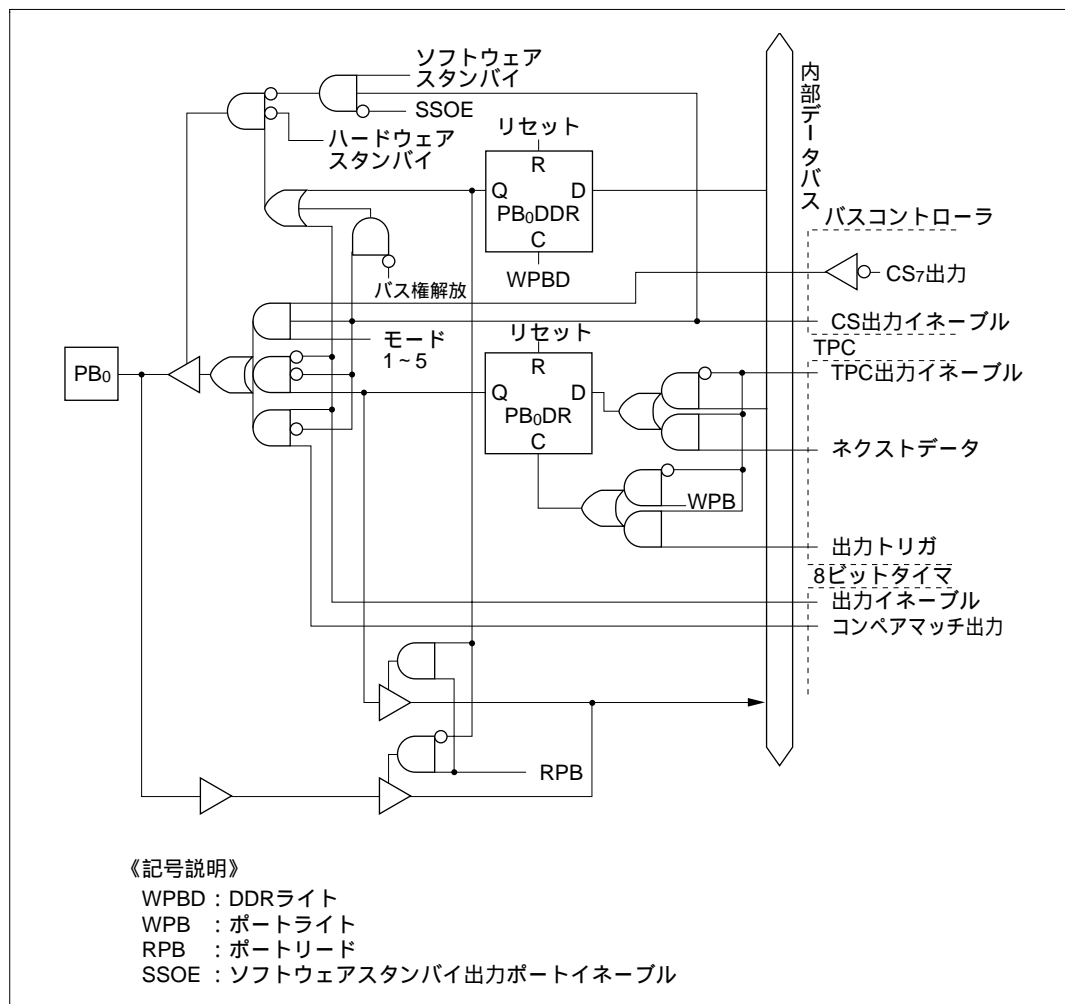


図 C.11 (a) ポートBブロック図 (PB<sub>0</sub>端子)

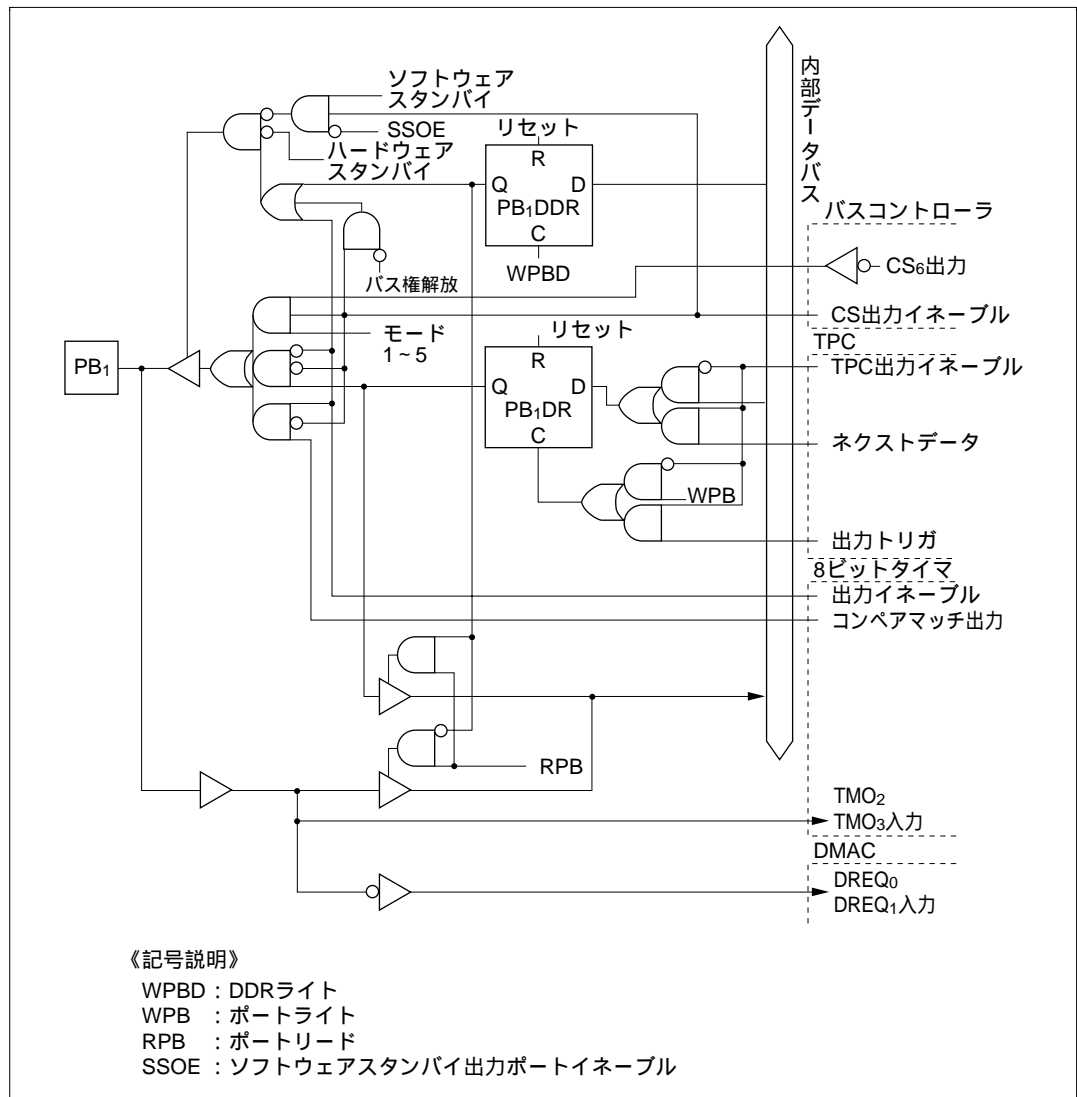


図 C.11 (b) ポート B ブロック図 (PB<sub>1</sub>端子)

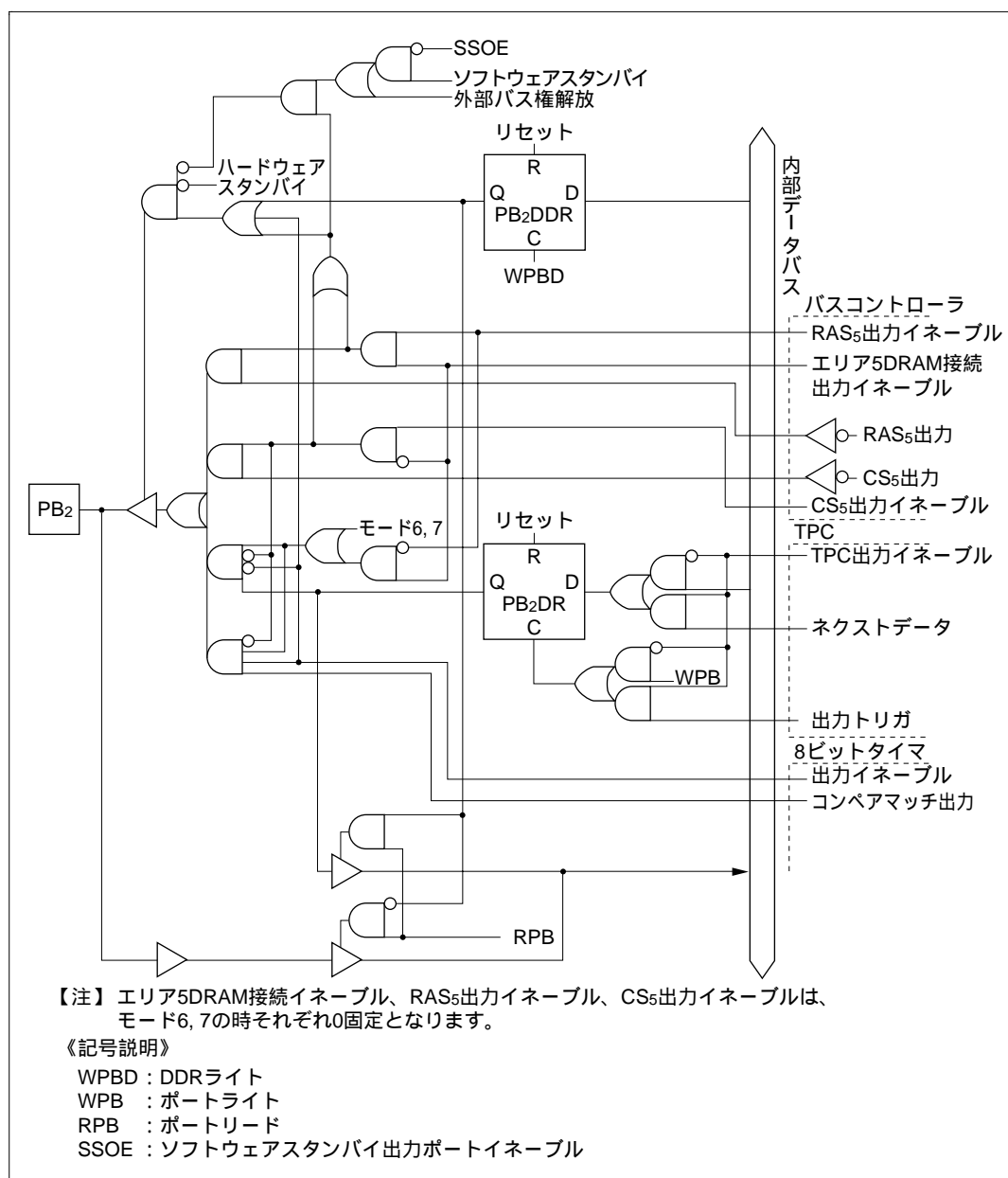


図 C.11 (c) ポート B ブロック図 (PB<sub>2</sub>端子)

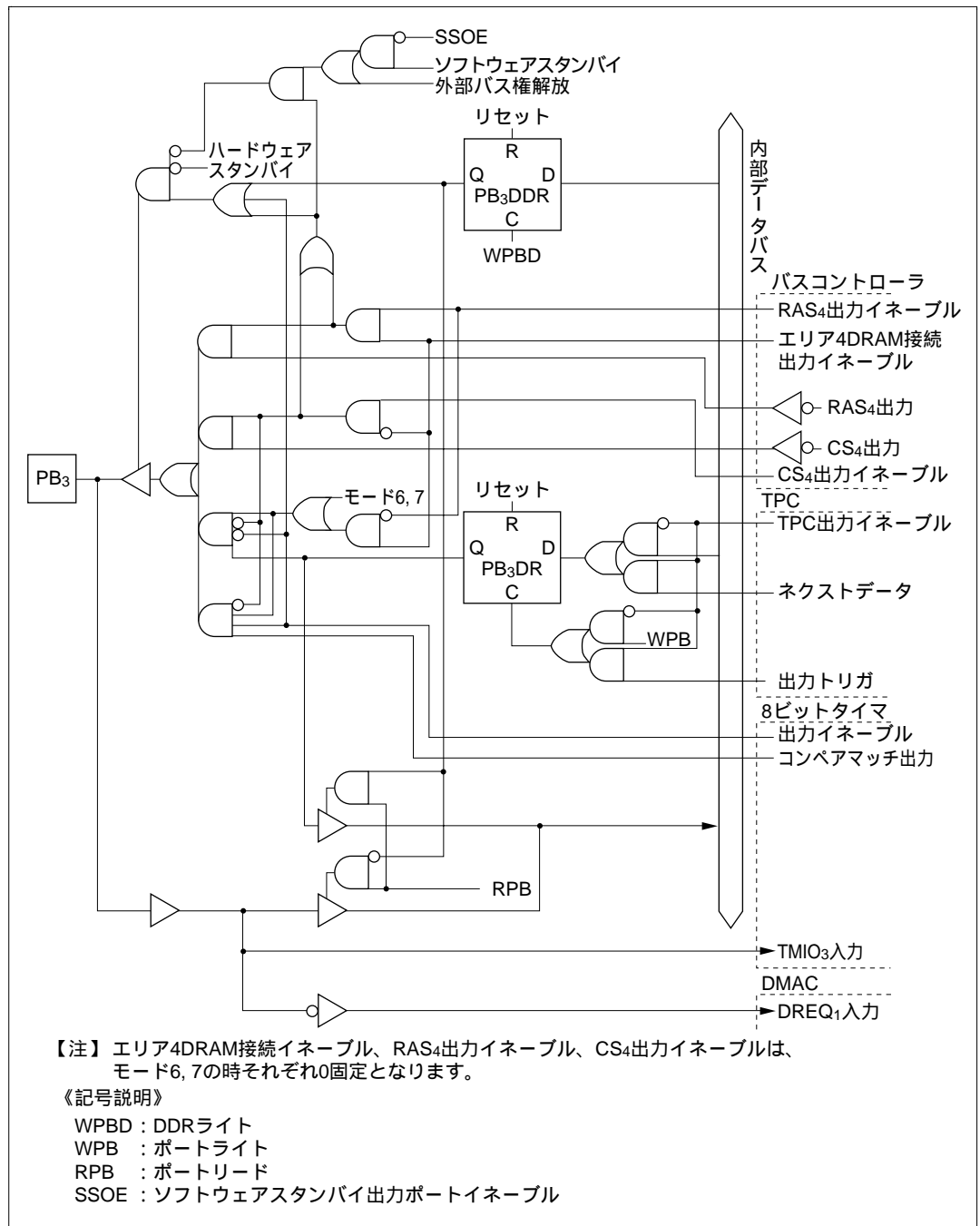


図 C.11 (d) ポート B ブロック図 (PB<sub>3</sub>端子)

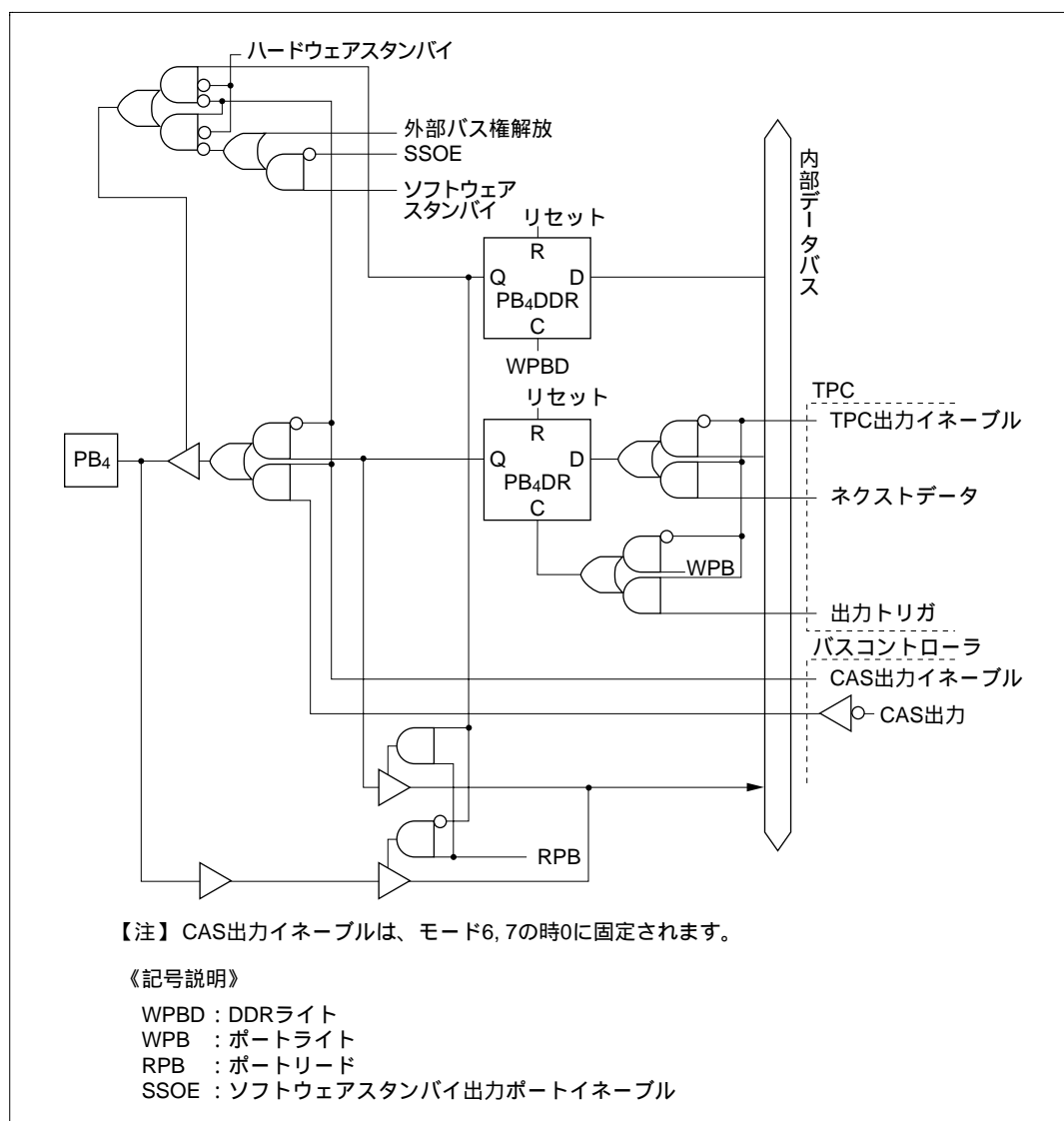


図 C.11 (e) ポート B ブロック図 (PB<sub>4</sub>端子)



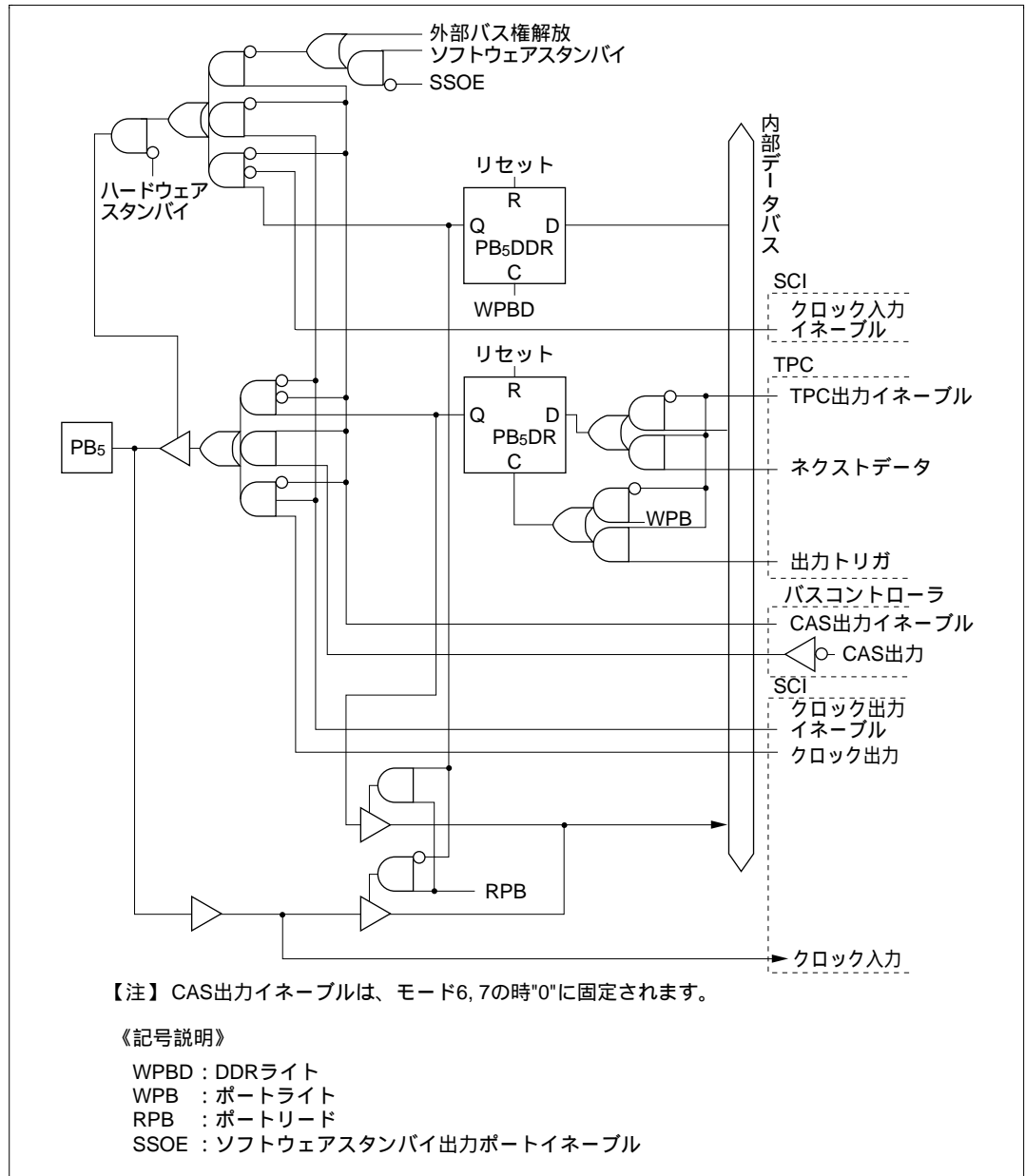


図 C.11 (f) ポート B ブロック図 (PB<sub>5</sub>端子)

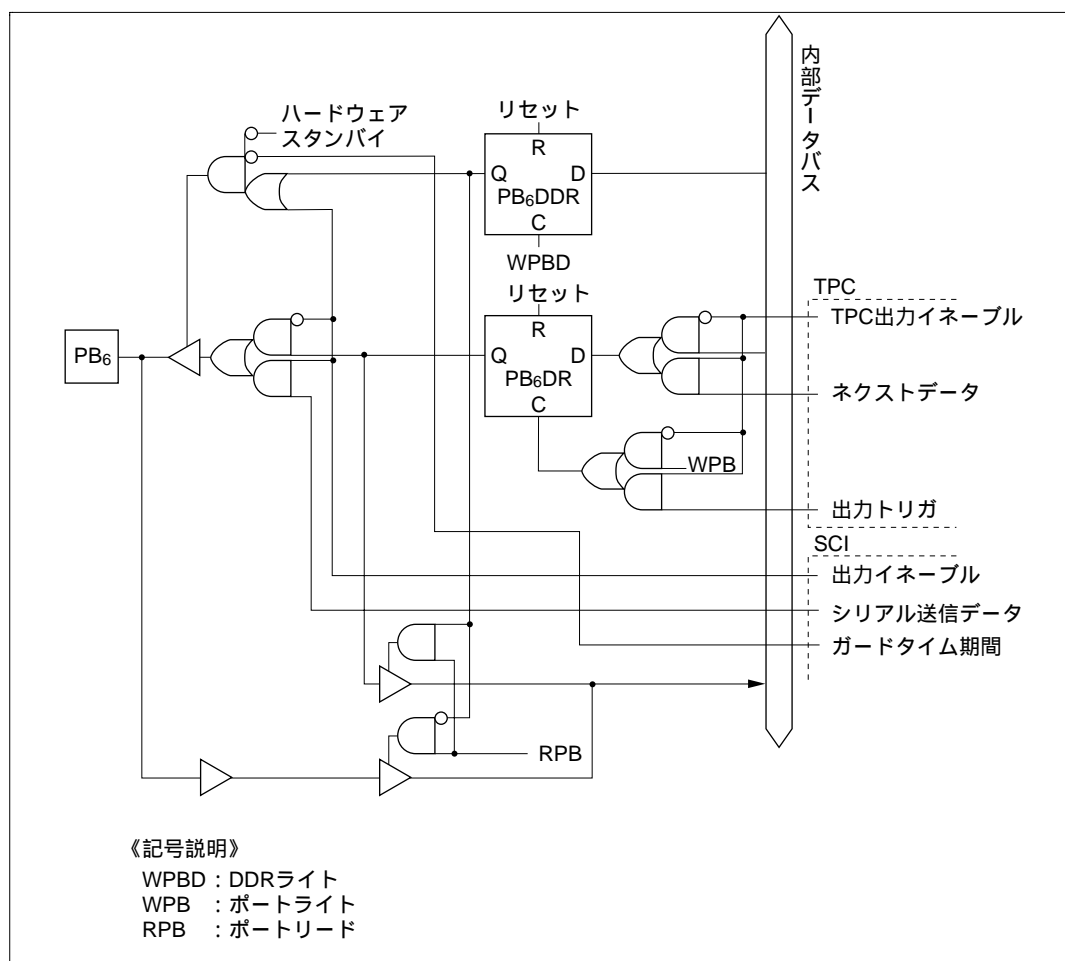


図 C.11 (g) ポート B ブロック図 (PB<sub>6</sub>端子)

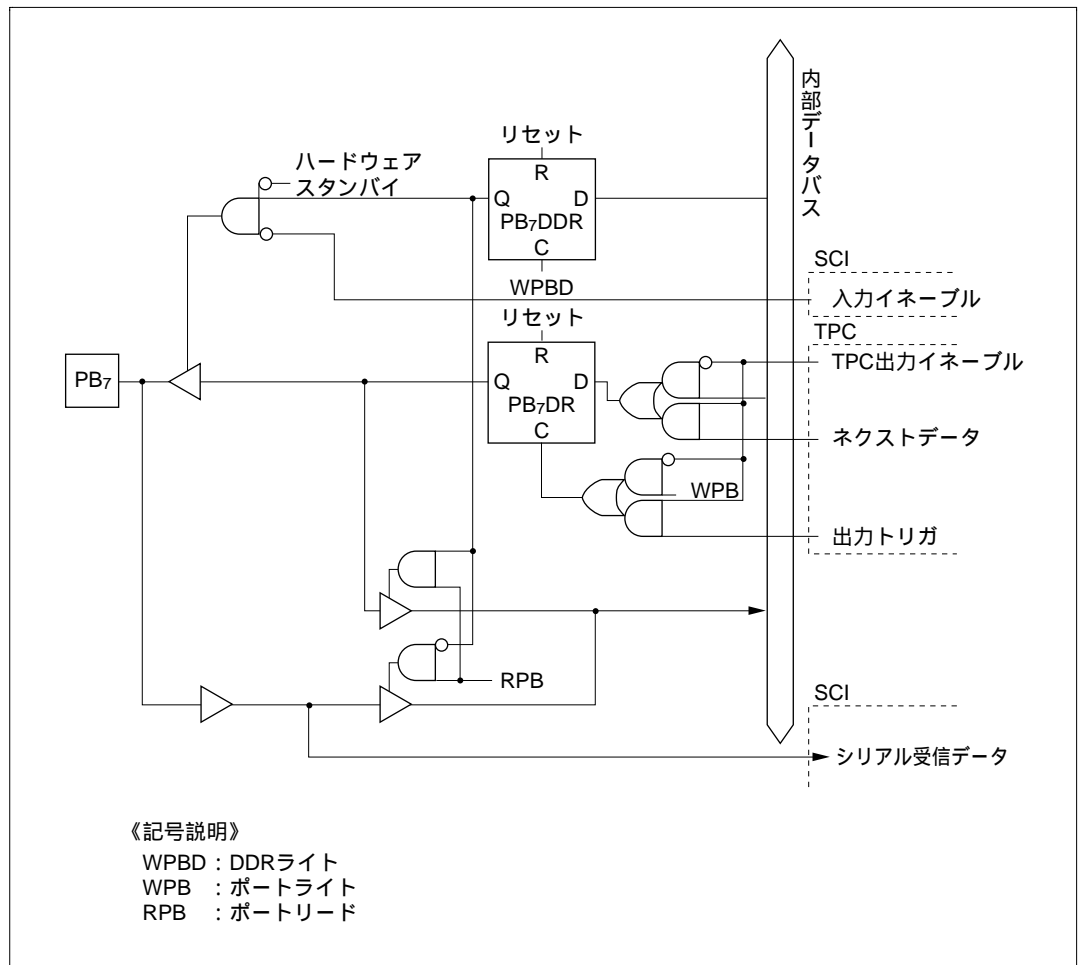


図 C.11 (h) ポート B ブロック図 (PB<sub>7</sub>端子)

## D. 端子状態

### D.1 各処理状態におけるポートの状態

表 D.1 各ポートの状態一覧

ポート名 端子名	モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	バス権 解放状態	プログラム 実行状態
P1 <sub>7</sub> ~ P1 <sub>0</sub>	1 ~ 4	L	T	[ SSOE=0 ] T [ SSOE=1 ] Keep	T	A <sub>7</sub> ~ A <sub>0</sub>
	5	T	T	[ DDR=0 ] Keep [ DDR=1, SSOE=0 ] T [ DDR=1, SSOE=1 ] Keep	T	[ DDR=0 ] 入力ポート [ DDR=1 ] A <sub>7</sub> ~ A <sub>0</sub>
	6, 7	T	T	Keep	—	入出力ポート
P2 <sub>7</sub> ~ P2 <sub>0</sub>	1 ~ 4	L	T	[ SSOE=0 ] T [ SSOE=1 ] Keep	T	A <sub>15</sub> ~ A <sub>8</sub>
	5	T	T	[ DDR=0 ] Keep [ DDR=1, SSOE=0 ] T [ DDR=1, SSOE=1 ] Keep	T	[ DDR=0 ] 入力ポート [ DDR=1 ] A <sub>15</sub> ~ A <sub>8</sub>
	6, 7	T	T	Keep	—	入出力ポート
P3 <sub>7</sub> ~ P3 <sub>0</sub>	1 ~ 5	T	T	T	T	D <sub>15</sub> ~ D <sub>8</sub>
	6, 7	T	T	Keep	—	入出力ポート
P4 <sub>7</sub> ~ P4 <sub>0</sub>	1, 3, 5	T	T	Keep	Keep	入出力ポート
	2, 4	T	T	T	T	D <sub>7</sub> ~ D <sub>0</sub>
	6, 7	T	T	Keep	—	入出力ポート

表 D.1 各ポートの状態一覧(2)

ポート名 端子名	モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	バス権 解放状態	プログラム 実行状態
P5 <sub>3</sub> ~ P5 <sub>0</sub>	1 ~ 4	L	T	[ SSOE=0 ] T [ SSOE=1 ] Keep	T	A <sub>19</sub> ~ A <sub>16</sub>
	5	T	T	[ DDR=0 ] Keep [ DDR=1, SSOE=0 ] T [ DDR=1, SSOE=1 ] Keep	T	[ DDR=0 ] 入力ポート [ DDR=1 ] A <sub>19</sub> ~ A <sub>16</sub>
	6, 7	T	T	Keep	—	入出力ポート
P6 <sub>0</sub>	1 ~ 5	T	T	Keep	Keep	入出力ポート WAIT
	6, 7	T	T	Keep	—	入出力ポート
P6 <sub>1</sub>	1 ~ 5	T	T	[ BRLE=0 ] Keep [ BRLE=1 ] T	T	入出力ポート BREQ
	6, 7	T	T	Keep	—	入出力ポート
P6 <sub>2</sub>	1 ~ 5	T	T	[ BRLE=0 ] Keep [ BRLE=1 ] H	L	[ BRLE=0 ] 入出力ポート [ BRLE=1 ] BACK
	6, 7	T	T	Keep	—	入出力ポート
P6 <sub>6</sub> ~ P6 <sub>3</sub>	1 ~ 5	H	T	[ SSOE=0 ] T [ SSOE=1 ] H	T	AS, RD HWR, LWR
	6, 7	T	T	Keep	—	入出力ポート
P6 <sub>7</sub>	1 ~ 7	クロック出力	T	[ PSTOP=0 ] H [ PSTOP=1 ] Keep	[ PSTOP=0 ]  [ PSTOP=1 ] Keep	[ PSTOP=0 ]  [ PSTOP=1 ] 入力ポート
P7 <sub>7</sub> ~ P7 <sub>0</sub>	1 ~ 7	T	T	T	T	入力ポート

表 D.1 各ポートの状態一覧(3)

ポート名 端子名	モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	バス権 解放状態	プログラム 実行状態
P8 <sub>0</sub>	1~5	T	T	DRAM空間を選択して いない場合* <sup>1</sup> [RFSHE=0] Keep [RFSHE=1] 設定禁止 DRAM空間を選択して いる場合* <sup>2</sup> [RFSHE=0] Keep [RFSHE=1, SRFMD=0, SSOE=0] T [RFSHE=1, SRFMD=0, SSOE=1] H [RFSHE=1, SRFMD=1] RFSH	DRAM空間を選択 していない場合* <sup>1</sup> [RFSHE=0] Keep [RFSHE=1] 設定禁止 DRAM空間を選択 している場合* <sup>2</sup> [RFSHE=0] Keep [RFSHE=1] T	[RFSHE=0] 入出力ポート [RFSHE=1] RFSH
	6, 7	T	T	Keep	—	入出力ポート
P8 <sub>1</sub>	1~5	T	T	DRAM空間を選択し、 RAS <sub>3</sub> を出力する場合* <sup>3</sup> [SSOE=0] T [SSOE=1] H DRAM空間を選択し、 RAS <sub>3</sub> を出力しない場合* <sup>4</sup> Keep 上記以外* <sup>5</sup> * <sup>1</sup> [DDR=0] T [DDR=1, SSOE=0] T [DDR=1, SSOE=1] H	DRAM空間を選択 し、RAS <sub>3</sub> を出力す る場合* <sup>3</sup> T DRAM空間を選択 し、RAS <sub>3</sub> を出力し ない場合* <sup>4</sup> Keep 上記以外 [DDR=0]* <sup>1</sup> Keep [DDR=1] T	DRAM空間を選択し、 RAS <sub>3</sub> を出力する場合 RAS <sub>3</sub> DRAM空間を選択し、 RAS3を出力しない場 合 入出力ポート 上記以外 [DDR=0] 入力ポート [DDR=1] CS <sub>3</sub>
	6, 7	T	T	Keep	—	入出力ポート
P8 <sub>2</sub>	1~5	T	T	RAS <sub>2</sub> 出力時* <sup>2</sup> [SSOE=0] T [SSOE=1] H 上記以外* <sup>1</sup> [DDR=0] T [DDR=1, SSOE=0] T [DDR=1, SSOE=1] H	RAS <sub>2</sub> 出力時* <sup>2</sup> T 上記以外* <sup>1</sup> [DDR=0] Keep [DDR=1] T	RAS <sub>2</sub> 出力時 RAS <sub>2</sub> 上記以外 [DDR=0] 入力ポート [DDR=1] CS <sub>2</sub>
	6, 7	T	T	Keep	—	入出力ポート

表 D.1 各ポートの状態一覧(4)

ポート名 端子名	モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	バス権 解放状態	プログラム 実行状態
P8 <sub>3</sub>	1~5	T	T	[ DDR=0 ] T [ DDR=1, SSOE=0 ] T [ DDR=1, SSOE=1 ] H	[ DDR=0 ] Keep [ DDR=1 ] T	[ DDR=0 ] 入力ポート [ DDR=1 ] $\overline{CS}_1$
	6, 7	T	T	Keep	—	入出力ポート
P8 <sub>4</sub>	1~4	H	T	[ DDR=0 ] T [ DDR=1, SSOE=0 ] T [ DDR=1, SSOE=1 ] H	[ DDR=0 ] Keep [ DDR=1 ] T	[ DDR=0 ] 入力ポート [ DDR=1 ] $\overline{CS}_0$
	5	T	T	[ DDR=0 ] T [ DDR=1, SSOE=0 ] T [ DDR=1, SSOE=1 ] H	[ DDR=0 ] Keep [ DDR=1 ] T	[ DDR=0 ] 入力ポート [ DDR=1 ] $\overline{CS}_0$
	6, 7	T	T	Keep	—	入出力ポート
P9 <sub>5</sub> ~ P9 <sub>0</sub>	1~7	T	T	Keep	Keep	入出力ポート
PA <sub>3</sub> ~ PA <sub>0</sub>	1~7	T	T	Keep	Keep	入出力ポート
PA <sub>6</sub> ~ PA <sub>4</sub>	1, 2, 6, 7	T	T	Keep	Keep	入出力ポート
	3~5	T	T	アドレス出力時 <sup>5</sup> [ SSOE=0 ] T [ SSOE=1 ] Keep 上記以外 <sup>6</sup> Keep	アドレス出力時 <sup>5</sup> T 上記以外 <sup>6</sup> Keep	アドレス出力時 A <sub>23</sub> ~ A <sub>21</sub> 上記以外 入出力ポート

表 D.1 各ポートの状態一覧(5)

ポート名 端子名	モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	バス権 解放状態	プログラム 実行状態
PA <sub>7</sub>	1, 2	T	T	Keep	Keep	入出力ポート
	3, 4	L	T	[ SSOE=0 ] T [ SSOE=1 ] Keep	T	A <sub>20</sub>
	5	L	T	A20E=0 のとき [ SSOE=0 ] T [ SSOE=1 ] Keep A20E=1 のとき Keep	A20E=0 の とき T A20E=1 の とき Keep	A20E=0 のとき A <sub>20</sub> A20E=1 のとき 入出力ポート
	6, 7	T	T	Keep	—	入出力ポート
PB <sub>1</sub> , PB <sub>0</sub>	1~5	T	T	CS 出力時* <sup>7</sup> [ SSOE=0 ] T [ SSOE=1 ] H 上記以外* <sup>8</sup> Keep	CS 出力時* <sup>7</sup> T 上記以外* <sup>8</sup> Keep	CS 出力時 $\overline{CS}_7, \overline{CS}_6$ 上記以外 入出力ポート
	6, 7	T	T	Keep	—	入出力ポート
PB <sub>2</sub>	1~5	T	T	RAS <sub>5</sub> 出力時* <sup>9</sup> [ SSOE=0 ] T [ SSOE=1 ] H CS 出力時* <sup>10</sup> [ SSOE=0 ] T [ SSOE=1 ] H 上記以外* <sup>11</sup> Keep	RAS <sub>5</sub> 出力時* <sup>9</sup> T CS 出力時* <sup>10</sup> T 上記以外* <sup>11</sup> Keep	RAS <sub>5</sub> 出力時 $\overline{RAS}_5$ CS 出力時 $\overline{CS}_5$ 上記以外 入出力ポート
	6, 7	T	T	Keep	—	入出力ポート



表 D.1 各ポートの状態一覧(6)

ポート名 端子名	モード	リセット	ハードウェア スタンバイモード	ソフトウェア スタンバイモード	バス権 解放状態	プログラム 実行状態
PB <sub>3</sub>	1~5	T	T	RAS <sub>4</sub> 出力時* <sup>12</sup> [SSOE=0] T [SSOE=1] H CS出力時* <sup>13</sup> [SSOE=0] T [SSOE=1] H 上記以外* <sup>14</sup> Keep	RAS <sub>4</sub> 出力時* <sup>12</sup> T CS出力時* <sup>13</sup> T 上記以外* <sup>14</sup> Keep	RAS <sub>4</sub> 出力時 $\overline{\text{RAS}}_4$ CS出力時 $\overline{\text{CS}}_4$ 上記以外 入出力ポート
	6, 7	T	T	Keep	—	入出力ポート
PB <sub>5</sub> , PB <sub>4</sub>	1~5	T	T	CAS出力時* <sup>15</sup> [SSOE=0] T [SSOE=1] H 上記以外* <sup>16</sup> Keep	CAS出力時* <sup>15</sup> T 上記以外* <sup>16</sup> Keep	CAS出力時 $\overline{\text{UCAS}}$ , $\overline{\text{LCAS}}$ 上記以外 入出力ポート
	6, 7	T	T	Keep	—	入出力ポート
PB <sub>7</sub> ~PB <sub>4</sub>	1~7	T	T	Keep	Keep	入出力ポート

## 【記号説明】

H: High レベル

L: Low レベル

T: ハイインピーダンス

Keep: 入力ポートはハイインピーダンス、出力ポートは保持

DDR: データディレクションレジスタ

- 【注】 \*1 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0 がすべて 0 のとき。
- \*2 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0 のいずれかが 1 のとき。
- \*3 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=010, 100, 101 のとき。
- \*4 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=010, 100, 101, 000 以外のとき。

- 
- \*5 BRCR (バスリリースコントロールレジスタ) の A23E, A22E, A21E がそれぞれ 0 のとき。
  - \*6 BRCR (バスリリースコントロールレジスタ) の A23E, A22E, A21E がそれぞれ 1 のとき。
  - \*7 CSCR (チップセレクトコントロールレジスタ) の CS7E, CS6E がそれぞれ 1 のとき。
  - \*8 CSCR (チップセレクトコントロールレジスタ) の CS7E, CS6E がそれぞれ 0 のとき。
  - \*9 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=101 のとき。
  - \*10 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=101 以外で、CSCR (チップセレクトコントロールレジスタ) の CS5E=1 のとき。
  - \*11 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=101 以外で、CSCR (チップセレクトコントロールレジスタ) の CS5E=0 のとき。
  - \*12 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=100, 101, 110 のとき。
  - \*13 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=100, 101, 110 以外で、CSCR (チップセレクトコントロールレジスタ) の CS4E=1 のとき。
  - \*14 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0=100, 101, 110 以外で、CSCR (チップセレクトコントロールレジスタ) の CS4E=0 のとき。
  - \*15 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0 のいずれかが 1 で、DRCRB (DRAM コントロールレジスタ B) の CSEL=0 のとき。
  - \*16 DRCRA (DRAM コントロールレジスタ A) の DRAS2, DRAS1, DRAS0 のいずれかが 1 で、DRCRB (DRAM コントロールレジスタ B) の CSEL=1 のとき。または、DRAS2, DRAS1, DRAS0 がすべて 0 のとき。

## D.2 リセット時の端子状態

### (1) モード 1、2

モード 1、2 で外部メモリアクセス中に、 $\overline{\text{RES}}$  端子が Low レベルになったときのタイミングを図 D.1 に示します。

$\overline{\text{RES}}$  端子が Low レベルになると同時に各ポートはイニシャライズされ入力ポートになります。また、 $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$ 、 $\overline{\text{CS}}_0$  が High レベル、 $\text{D}_{15} \sim \text{D}_0$  はハイインピーダンスになります。

アドレスバスは  $\overline{\text{RES}}$  端子が Low レベルをサンプリングしてから 2.5 クロック後にイニシャライズされ、アドレスバスは Low レベル出力となります。クロック端子  $\text{P6}_7/$  は  $\overline{\text{RES}}$  端子が Low レベルになった次の立上がりで出力端子になります。

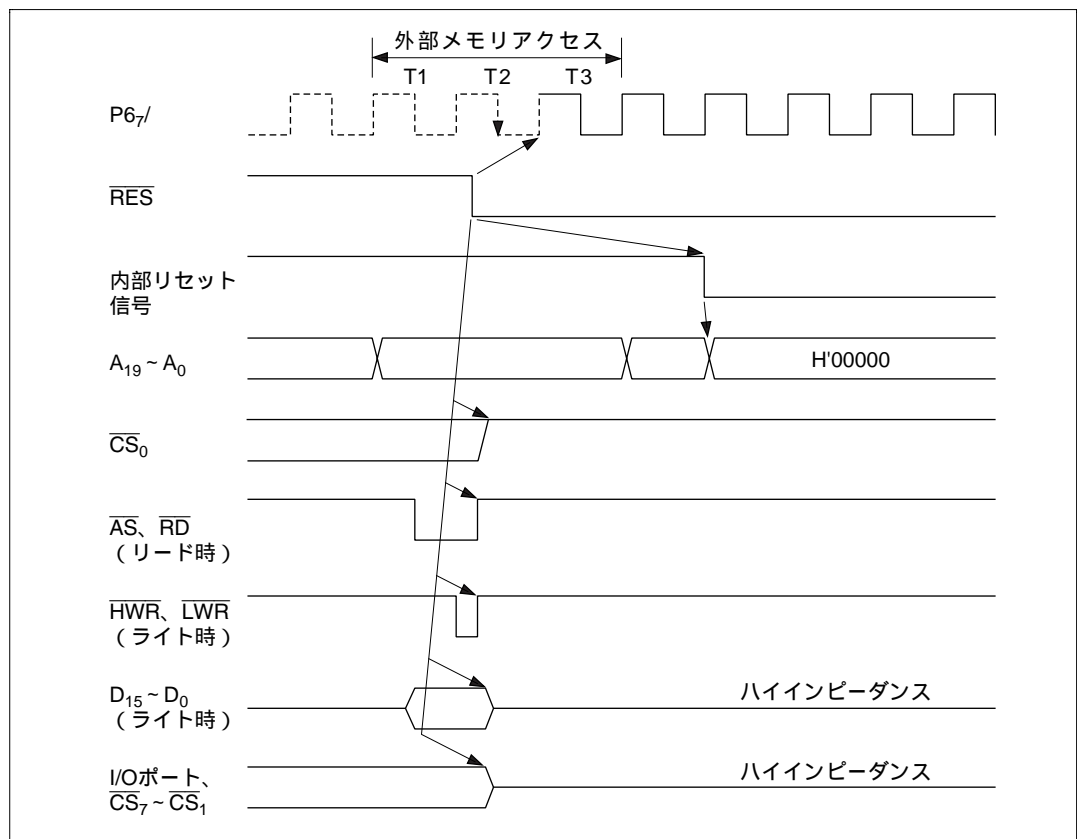


図 D.1 メモリアクセス中のリセット (モード 1、2)

## (2) モード 3、4

モード 3、4 で外部メモリアクセス中に、 $\overline{\text{RES}}$  端子が Low レベルになったときのタイミングを図 D.2 に示します。

$\overline{\text{RES}}$  端子が Low レベルになると同時に各ポートはイニシャライズされ入力ポートになります。また、 $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$ 、 $\overline{\text{CS}}_0$  が High レベル、 $\text{D}_{15} \sim \text{D}_0$  はハイインピーダンスになります。

アドレスバスは  $\overline{\text{RES}}$  端子が Low レベルをサンプリングしてから 2.5 クロック後にイニシャライズされ、アドレスバスは Low レベル出力となります。ただし、 $\text{PA}_4 \sim \text{PA}_6$  をアドレスバスとして使用している場合、 $\text{P8}_3 \sim \text{P8}_1$ 、 $\text{PB}_0 \sim \text{PB}_3$  を CS 出力端子として使用している場合は、 $\overline{\text{RES}}$  端子が Low レベルになると同時にハイインピーダンスとなります。

クロック端子  $\text{P6}_7/$  は  $\overline{\text{RES}}$  端子が Low レベルになった次の の立上がりで出力端子になります。

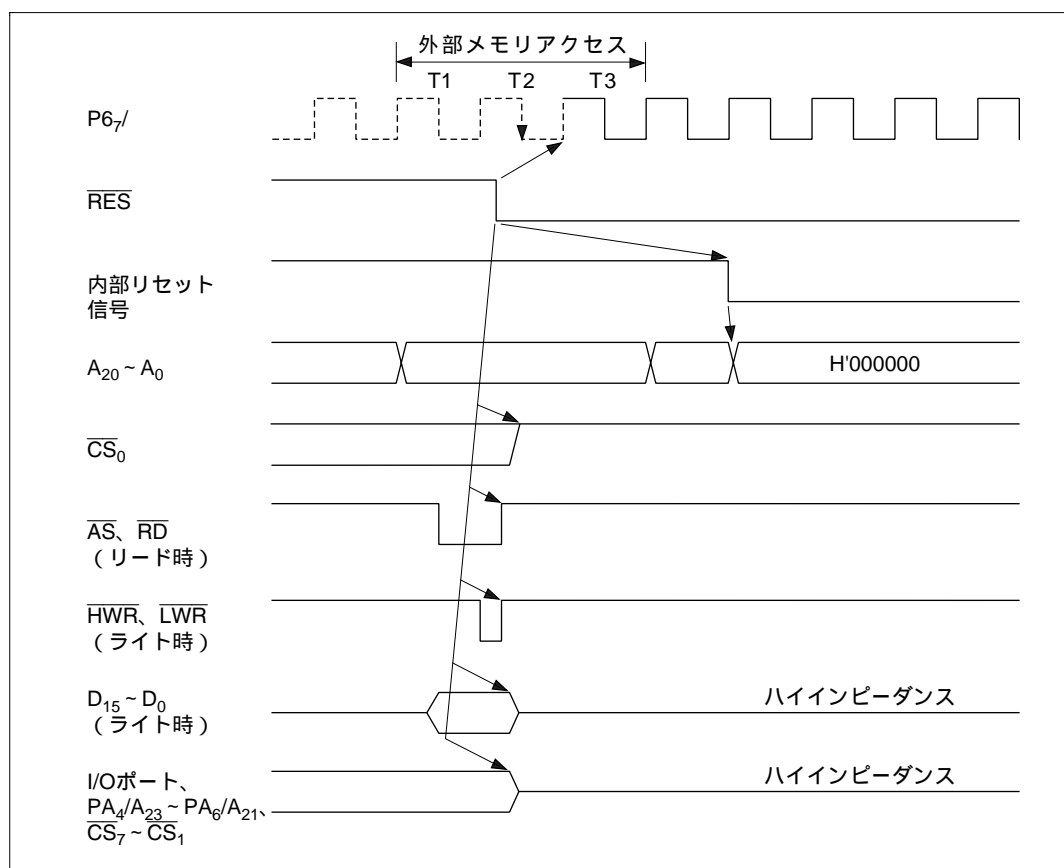


図 D.2 メモリアクセス中のリセット (モード 3、4)

(3) モード5

モード5で外部メモリアクセス中に、 $\overline{\text{RES}}$  端子が Low レベルになったときのタイミングを図D.3に示します。

$\overline{\text{RES}}$  端子が Low レベルになると同時に各ポートはイニシャライズされ、入力ポートになります。また、 $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{HWR}}$ 、 $\overline{\text{LWR}}$  が High レベル、アドレスバス、 $\text{D}_{15} \sim \text{D}_0$  はハイインピーダンスになります。

クロック端子  $\text{P6}_7/\text{}$  は、 $\overline{\text{RES}}$  端子が Low レベルになった次の の立上がりで出力端子になります。

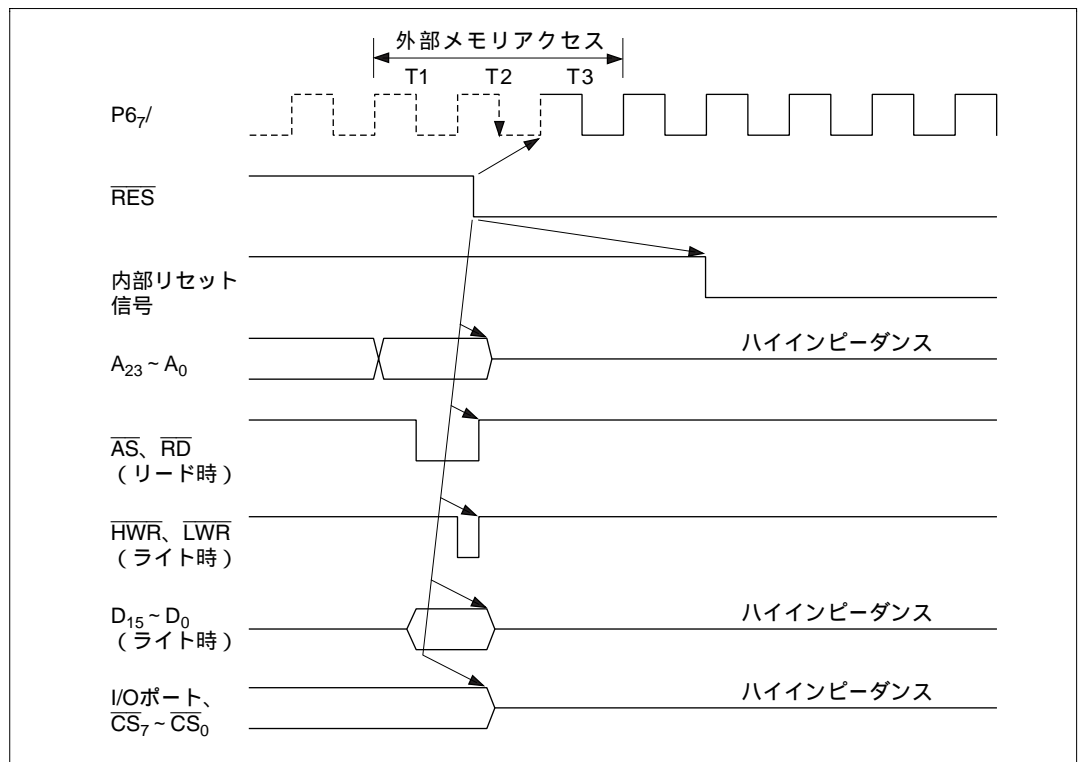


図 D.3 メモリアクセス中のリセット (モード5)

## (4) モード 6、7

モード 6、7 で動作中に、 $\overline{\text{RES}}$  端子が Low レベルになったときのタイミングを図 D.4 に示します。

$\overline{\text{RES}}$  端子が Low レベルになると同時に各ポートはイニシャライズされ、入力ポートになります。

クロック端子  $\text{P}_{6/7}$  は、 $\overline{\text{RES}}$  端子が Low レベルになった次の立上がりで出力端子になります。

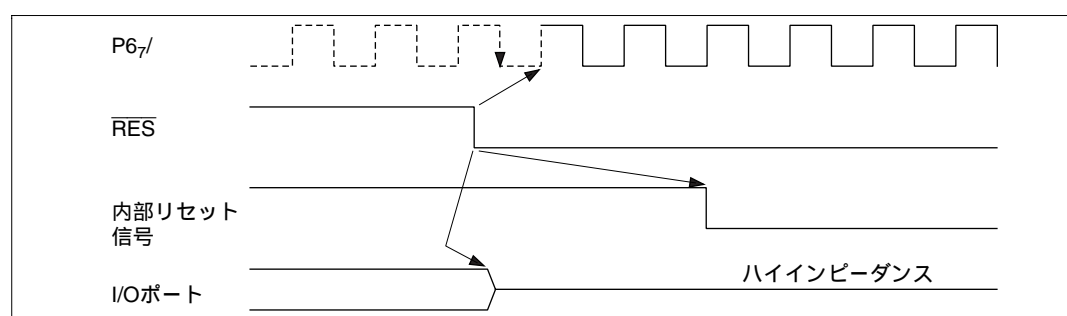


図 D.4 動作中のリセット (モード 6、7)

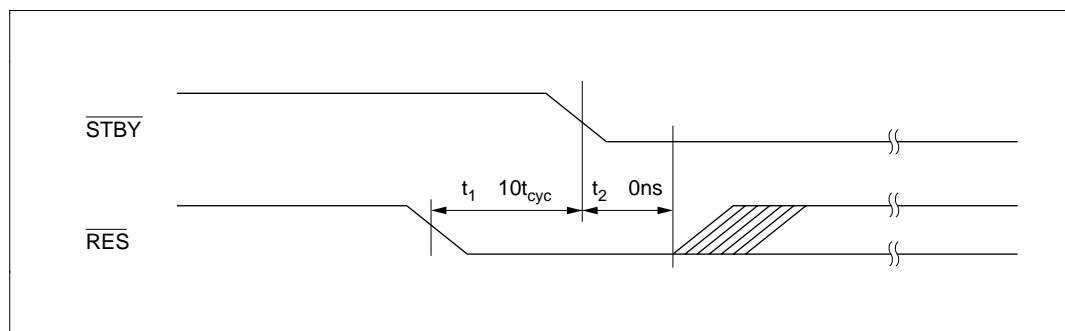
## E. ハードウェアスタンバイモード遷移 / 復帰時のタイミングについて

### (1) ハードウェアスタンバイモードの遷移タイミング

(1) SYSCR の RAME ビットを 1 にセットした状態で RAM の内容を保持する場合

下記に示すように  $\overline{\text{STBY}}$  信号の立下がりに対し、10 システムクロック前に  $\overline{\text{RES}}$  信号を Low としてください。

また、 $\overline{\text{RES}}$  信号の立下がりには、 $\overline{\text{STBY}}$  信号の立下がりに対し、min 0ns です。

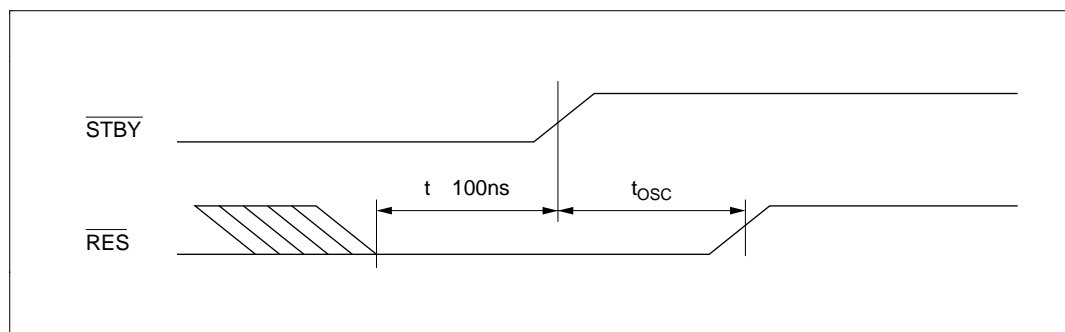


(2) SYSCR の RAME ビットを 0 にクリアした状態または RAM の内容を保持しない場合

(1) のように  $\overline{\text{RES}}$  信号を Low にする必要はありません。

### (2) ハードウェアスタンバイモードからの復帰タイミング

$\overline{\text{STBY}}$  信号の立上がりに対し、約 100ns 前に  $\overline{\text{RES}}$  信号を Low としてください。



## F. 型名一覧

表 F.1 H8/3068F 型名一覧

製品分類			製品型名	マーク型名	パッケージ (日立パッケージコード)
H8/3068F	フラッシュ メモリ 内蔵	5V 版	HD64F3068F	HD64F3068F	100 ピン QFP (FP-100B)
			HD64F3068TE	HD64F3068TE	100 ピン TQFP (TFP-100B)
		3V 版	HD64F3068VF	HD64F3068VF	100 ピン QFP (FP-100B)
			HD64F3068VTE	HD64F3068VTE	100 ピン TQFP (TFP-100B)



## G. 外形寸法図

本 LSI の外形寸法図 FP-100B を図 G.1、TFP-100B を図 G.2、FP-100A を図 G.3 に示します。

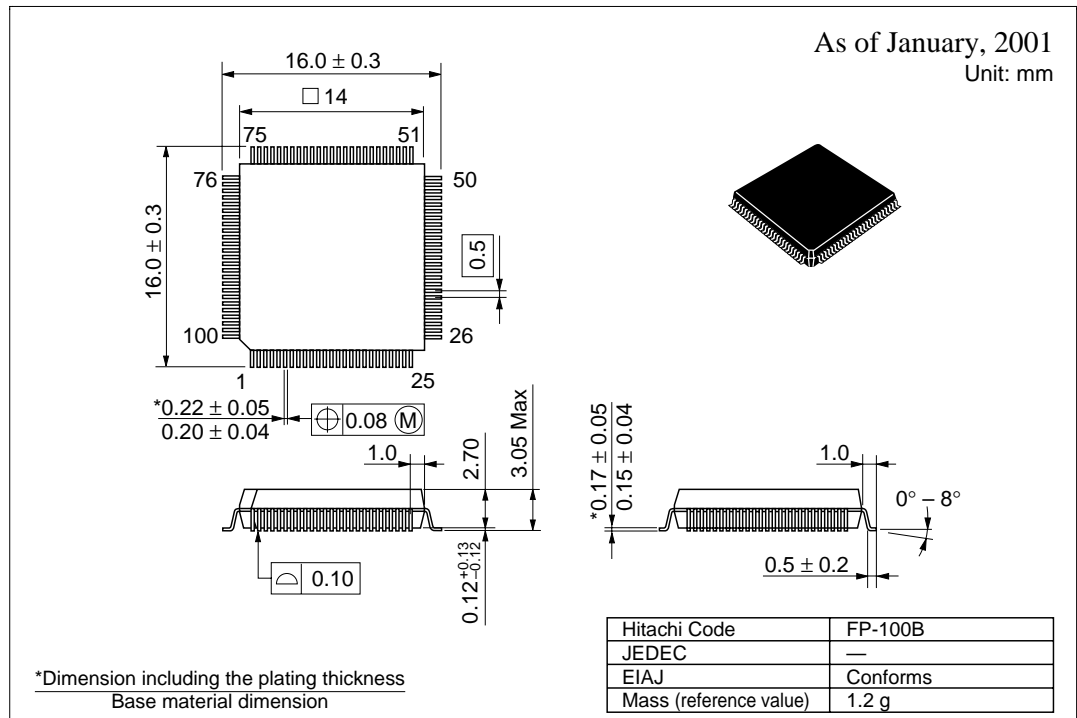


図 G.1 外形寸法図 (FP-100B) 単位 : mm

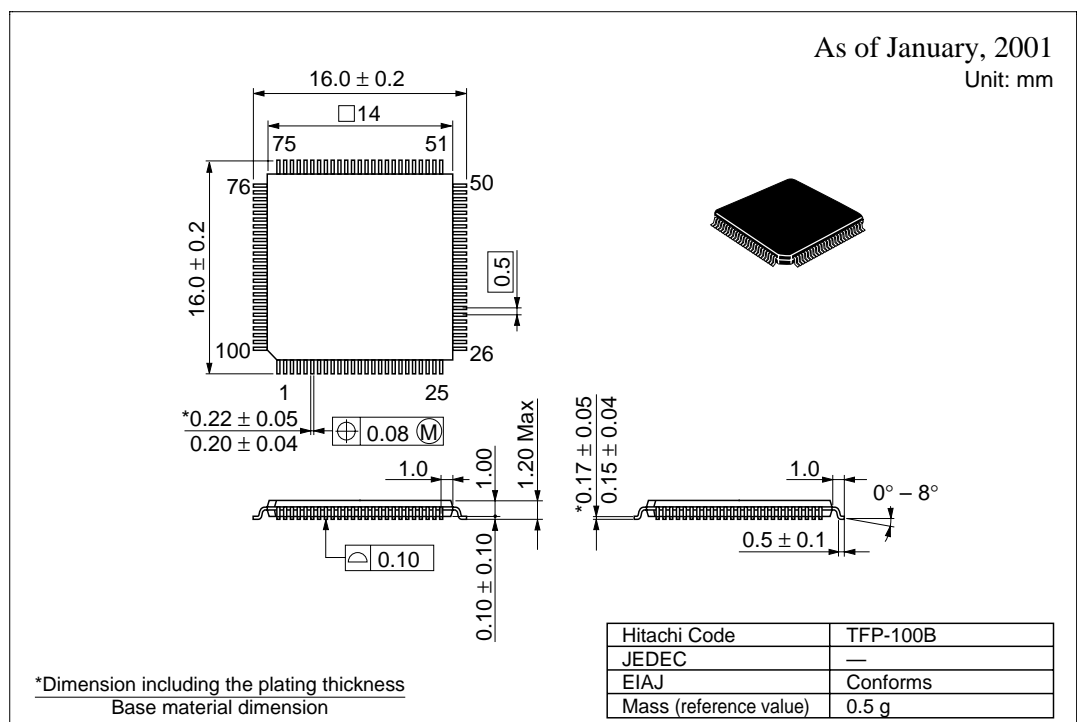


図 G.2 外形寸法図 (TFP-100B) 単位 : mm

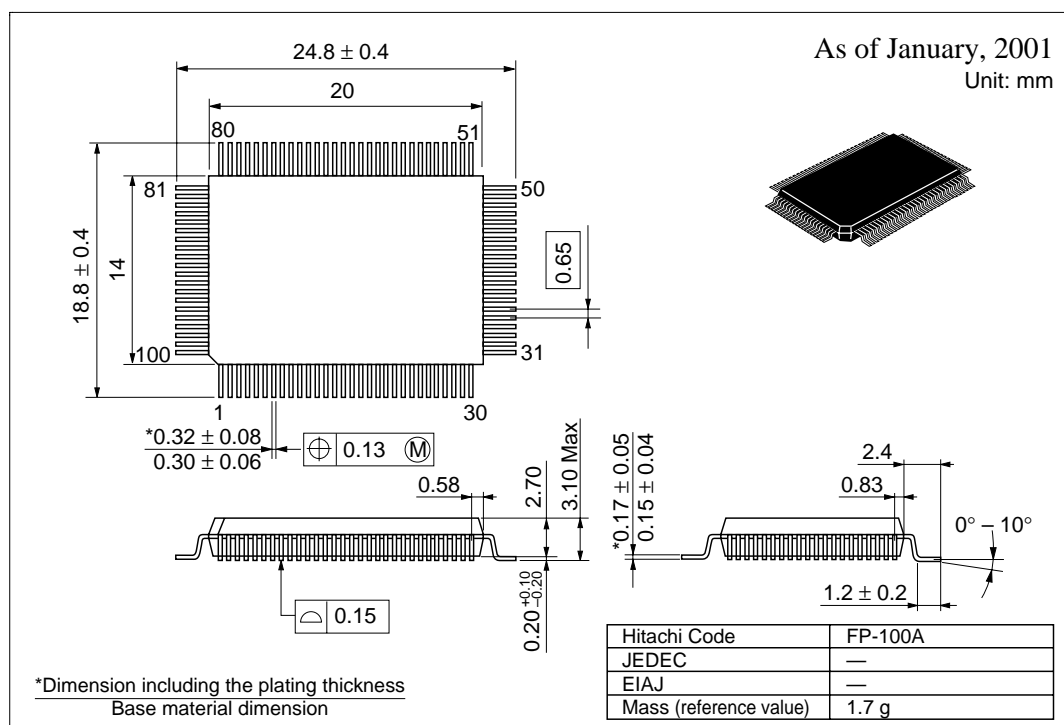


図 G.3 外形寸法図 (FP-100A) 単位 : mm

## H. H8/300H シリーズ製品仕様比較

### H.1 H8/3068F と H8/3067、H8/3062 シリーズ、H8/3048 シリーズの相異点

項目		H8/3068F		H8/3067 シリーズ H8/3062 シリーズ		H8/3048 シリーズ	
1	動作モード	モード5	16M バイトROMの有効拡張モード		16M バイト ROM有効拡張モード		
		モード6	64k バイトシングルチップモード		64k バイト シングルチップモード		
2	割込み コントローラ	内部割込み要因	36		36 (H8/3067)、 27 (H8/3062)		
3	バスコントローラ	バーストROM インタフェース	あり		あり (H8/3067) なし (H8/3062)		
		アイドルサイクル 挿入機能	あり		あり		
		ウェイトモード	2種類		2種類		
		ウェイトステート 数の設定	エリア単位		エリア単位		
		アドレス出力方式	アドレス更新モードを選択可		アドレス更新モードを選択可 (マスクROM版、フラッシュメモリ R版専用)		
4	DRAM インタフェース	接続可能エリア	エリア2/3/4/5		エリア2/3/4/5 (H8/3067のみ)		
		プリチャージサイ クル挿入機能	あり		あり (H8/3067のみ)		
		高速ページモード	あり		あり (H8/3067のみ)		
		アドレスシフト量	8bit/9bit/10bit		8bit/9bit/10bit (H8/3067のみ)		
5	タイマ機能		16ビットタイマ	8ビットタイマ	16ビットタイマ	8ビットタイマ	
		チャンネル数	16bit×3	8bit×4 (16bit× 2)	16bit×3	8bit×4 (16bit× 2)	16bit×5
		パルス出力	6端子	4端子(2端子)	6端子	4端子(2端子)	12端子
		インプットキャプ チャ	6本	2本	6本	2本	10本
		外部クロック	4系統(選択可)	4系統(固定)	4系統(選択可)	4系統(固定)	4系統(選択可)
		内部クロック	, /2, /4, /8	/8, /64, /8192	, /2, /4, /8	/8, /64, /8192	, /2, /4, /8
		相補PWM機能	なし	なし	なし	なし	あり
		リセット同期 PWM機能	なし	なし	なし	なし	あり
		バッファ動作	なし	なし	なし	なし	あり
		出力初期値設定機 能	あり	なし	あり	なし	なし
		PWM出力	3本	4本(2本)	3本	4本(2本)	5本
		DMAC起動	3チャンネル	なし	3チャンネル (H8/3067のみ)	なし	4チャンネル
		A/D変換起動	なし	あり	なし	あり	なし
		割込み要因	3要因×3	8要因	3要因×3	8要因	3要因×5

項目		H8/3068F	H8/3067 シリーズ, H8/3062 シリーズ	H8/3048 シリーズ	
6	TPC	タイムベース	16ビットタイムベースで3種類	16ビットタイムベースで3種類	ITUベースで4種類
7	WDT	リセット信号 外部出力機能	なし	あり(ただし、フラッシュメモリ内蔵品はなし)	あり
8	SCI	チャンネル数	3チャンネル	3チャンネル(H8/3067) 2チャンネル(H8/3062)	2チャンネル
		スマートカード インタフェース	全チャンネルサポート	全チャンネルサポート	SCI0のみ サポート
9	A/D変換器	変換開始トリガ 入力	外部トリガ/8ビットタイマコンペア マッチ	外部トリガ/8ビットタイマコンペア マッチ	外部トリガ
10	端子制御	端子	/入力ポート兼用	/入力ポート兼用	出力専用
		16MバイトROM 有効拡張モードに おけるA <sub>20</sub>	A <sub>20</sub> /入出力ポート兼用	A <sub>20</sub> /入出力ポート兼用	A <sub>20</sub> 出力
		ソフトウェアスタ ンバイ状態におけ る、アドレスバ ス、 $\overline{AS}$ 、 $\overline{RD}$ 、 $\overline{HWR}$ 、 $\overline{LWR}$ 、 $\overline{CS}_7 \sim \overline{CS}_0$ 、 $\overline{RFSH}$	Highレベル出力/ハイインピーダン スを選択可	Highレベル出力/ハイインピーダン スを選択可 ( $\overline{RFSH}$ 端子はH8/3067のみ)	Highレベル出力( $\overline{CS}_0$ 以外) Lowレベル出力( $\overline{CS}_0$ )
		バス解放状態にお ける $\overline{CS}_7 \sim \overline{CS}_0$	ハイインピーダンス	ハイインピーダンス	Highレベル出力
11	フラッシュメモリ 機能	書き込み/ 消去電圧	12V印加不要。単一電源書き込み	12V印加不要。単一電源書き込み。	外部から12V印加
		ブロック分割	14ブロック	8ブロック	16ブロック

## H.2 100ピンパッケージ品の端子機能比較（FP-100B、TFP-100Bの場合）

表 H.1 製品別ピン配置一覧（FP-100B、TFP-100B）（1）

ピン番号	H8/3068F	H8/3067 シリーズ	H8/3062 シリーズ	H8/3048 シリーズ	H8/3042 シリーズ
1	Vcc/V <sub>CL</sub>	Vcc	Vcc/V <sub>CL</sub>	Vcc	Vcc
2	PB <sub>0</sub> /TP <sub>8</sub> /TMO <sub>0</sub> / $\overline{CS}_7$	PB <sub>0</sub> /TP <sub>8</sub> /TMO <sub>0</sub> / $\overline{CS}_7$	PB <sub>0</sub> /TP <sub>8</sub> /TMO <sub>0</sub> / $\overline{CS}_7$	PB <sub>0</sub> /TP <sub>8</sub> /TIOCA <sub>3</sub>	PB <sub>0</sub> /TP <sub>8</sub> /TIOCA <sub>3</sub>
3	PB <sub>1</sub> /TP <sub>9</sub> /TMIO <sub>1</sub> / DREQ <sub>0</sub> / $\overline{CS}_6$	PB <sub>1</sub> /TP <sub>9</sub> /TMIO <sub>1</sub> / DREQ <sub>0</sub> / $\overline{CS}_6$	PB <sub>1</sub> /TP <sub>9</sub> /TMIO <sub>1</sub> / $\overline{CS}_6$	PB <sub>1</sub> /TP <sub>9</sub> /TIOCB <sub>3</sub>	PB <sub>1</sub> /TP <sub>9</sub> /TIOCB <sub>3</sub>
4	PB <sub>2</sub> /TP <sub>10</sub> /TMO <sub>2</sub> / $\overline{CS}_5$	PB <sub>2</sub> /TP <sub>10</sub> /TMO <sub>2</sub> / $\overline{CS}_5$	PB <sub>2</sub> /TP <sub>10</sub> /TMO <sub>2</sub> / $\overline{CS}_5$	PB <sub>2</sub> /TP <sub>10</sub> /TIOCA <sub>4</sub>	PB <sub>2</sub> /TP <sub>10</sub> /TIOCA <sub>4</sub>
5	PB <sub>3</sub> /TP <sub>11</sub> /TMIO <sub>3</sub> / DREQ <sub>1</sub> / $\overline{CS}_4$	PB <sub>3</sub> /TP <sub>11</sub> /TMIO <sub>3</sub> / DREQ <sub>1</sub> / $\overline{CS}_4$	PB <sub>3</sub> /TP <sub>11</sub> /TMIO <sub>3</sub> / $\overline{CS}_4$	PB <sub>3</sub> /TP <sub>11</sub> /TIOCB <sub>4</sub>	PB <sub>3</sub> /TP <sub>11</sub> /TIOCB <sub>4</sub>
6	PB <sub>4</sub> /TP <sub>12</sub> / $\overline{UCAS}$	PB <sub>4</sub> /TP <sub>12</sub> / $\overline{UCAS}$	PB <sub>4</sub> /TP <sub>12</sub>	PB <sub>4</sub> /TP <sub>12</sub> /TOCXA <sub>4</sub>	PB <sub>4</sub> /TP <sub>12</sub> /TOCXA <sub>4</sub>
7	PB <sub>5</sub> /TP <sub>13</sub> / $\overline{LCAS}$ / SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub> / $\overline{LCAS}$ / SCK <sub>2</sub>	PB <sub>5</sub> /TP <sub>13</sub>	PB <sub>5</sub> /TP <sub>13</sub> /TOCXB <sub>4</sub>	PB <sub>5</sub> /TP <sub>13</sub> /TOCXB <sub>4</sub>
8	PB <sub>6</sub> /TP <sub>14</sub> /TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub> /TxD <sub>2</sub>	PB <sub>6</sub> /TP <sub>14</sub>	PB <sub>6</sub> /TP <sub>14</sub> / $\overline{DREQ}_0$ / $\overline{CS}_7$	PB <sub>6</sub> /TP <sub>14</sub> / $\overline{DREQ}_0$
9	PB <sub>7</sub> /TP <sub>15</sub> /RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub> /RxD <sub>2</sub>	PB <sub>7</sub> /TP <sub>15</sub>	PB <sub>7</sub> /TP <sub>15</sub> / $\overline{DREQ}_1$ / ADTRG	PB <sub>7</sub> /TP <sub>15</sub> / $\overline{DREQ}_1$ / ADTRG
10	FWE	$\overline{RESO}$ /FWE* <sup>1</sup>	$\overline{RESO}$ /FWE* <sup>1</sup>	$\overline{RESO}$ /Vpp	$\overline{RESO}$
11	Vss	Vss	Vss	Vss	Vss
12	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>	P9 <sub>0</sub> /TxD <sub>0</sub>
13	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>	P9 <sub>1</sub> /TxD <sub>1</sub>
14	P9 <sub>2</sub> /RxD <sub>0</sub>	P9 <sub>2</sub> /RxD <sub>0</sub>	P9 <sub>2</sub> /RxD <sub>0</sub>	P9 <sub>2</sub> /RxD <sub>0</sub>	P9 <sub>2</sub> /RxD <sub>0</sub>
15	P9 <sub>3</sub> /RxD <sub>1</sub>	P9 <sub>3</sub> /RxD <sub>1</sub>	P9 <sub>3</sub> /RxD <sub>1</sub>	P9 <sub>3</sub> /RxD <sub>1</sub>	P9 <sub>3</sub> /RxD <sub>1</sub>
16	P9 <sub>4</sub> /SCK <sub>0</sub> / $\overline{IRQ}_4$	P9 <sub>4</sub> /SCK <sub>0</sub> / $\overline{IRQ}_4$	P9 <sub>4</sub> /SCK <sub>0</sub> / $\overline{IRQ}_4$	P9 <sub>4</sub> /SCK <sub>0</sub> / $\overline{IRQ}_4$	P9 <sub>4</sub> /SCK <sub>0</sub> / $\overline{IRQ}_4$
17	P9 <sub>5</sub> /SCK <sub>1</sub> / $\overline{IRQ}_5$	P9 <sub>5</sub> /SCK <sub>1</sub> / $\overline{IRQ}_5$	P9 <sub>5</sub> /SCK <sub>1</sub> / $\overline{IRQ}_5$	P9 <sub>5</sub> /SCK <sub>1</sub> / $\overline{IRQ}_5$	P9 <sub>5</sub> /SCK <sub>1</sub> / $\overline{IRQ}_5$
18	P4 <sub>0</sub> /D <sub>0</sub>	P4 <sub>0</sub> /D <sub>0</sub>	P4 <sub>0</sub> /D <sub>0</sub>	P4 <sub>0</sub> /D <sub>0</sub>	P4 <sub>0</sub> /D <sub>0</sub>
19	P4 <sub>1</sub> /D <sub>1</sub>	P4 <sub>1</sub> /D <sub>1</sub>	P4 <sub>1</sub> /D <sub>1</sub>	P4 <sub>1</sub> /D <sub>1</sub>	P4 <sub>1</sub> /D <sub>1</sub>
20	P4 <sub>2</sub> /D <sub>2</sub>	P4 <sub>2</sub> /D <sub>2</sub>	P4 <sub>2</sub> /D <sub>2</sub>	P4 <sub>2</sub> /D <sub>2</sub>	P4 <sub>2</sub> /D <sub>2</sub>
21	P4 <sub>3</sub> /D <sub>3</sub>	P4 <sub>3</sub> /D <sub>3</sub>	P4 <sub>3</sub> /D <sub>3</sub>	P4 <sub>3</sub> /D <sub>3</sub>	P4 <sub>3</sub> /D <sub>3</sub>
22	Vss	Vss	Vss	Vss	Vss
23	P4 <sub>4</sub> /D <sub>4</sub>	P4 <sub>4</sub> /D <sub>4</sub>	P4 <sub>4</sub> /D <sub>4</sub>	P4 <sub>4</sub> /D <sub>4</sub>	P4 <sub>4</sub> /D <sub>4</sub>
24	P4 <sub>5</sub> /D <sub>5</sub>	P4 <sub>5</sub> /D <sub>5</sub>	P4 <sub>5</sub> /D <sub>5</sub>	P4 <sub>5</sub> /D <sub>5</sub>	P4 <sub>5</sub> /D <sub>5</sub>
25	P4 <sub>6</sub> /D <sub>6</sub>	P4 <sub>6</sub> /D <sub>6</sub>	P4 <sub>6</sub> /D <sub>6</sub>	P4 <sub>6</sub> /D <sub>6</sub>	P4 <sub>6</sub> /D <sub>6</sub>
26	P4 <sub>7</sub> /D <sub>7</sub>	P4 <sub>7</sub> /D <sub>7</sub>	P4 <sub>7</sub> /D <sub>7</sub>	P4 <sub>7</sub> /D <sub>7</sub>	P4 <sub>7</sub> /D <sub>7</sub>
27	P3 <sub>0</sub> /D <sub>8</sub>	P3 <sub>0</sub> /D <sub>8</sub>	P3 <sub>0</sub> /D <sub>8</sub>	P3 <sub>0</sub> /D <sub>8</sub>	P3 <sub>0</sub> /D <sub>8</sub>
28	P3 <sub>1</sub> /D <sub>9</sub>	P3 <sub>1</sub> /D <sub>9</sub>	P3 <sub>1</sub> /D <sub>9</sub>	P3 <sub>1</sub> /D <sub>9</sub>	P3 <sub>1</sub> /D <sub>9</sub>

ピン番号	H8/3068F	H8/3067 シリーズ	H8/3062 シリーズ	H8/3048 シリーズ	H8/3042 シリーズ
29	P3 <sub>2</sub> /D <sub>10</sub>	P3 <sub>2</sub> /D <sub>10</sub>	P3 <sub>2</sub> /D <sub>10</sub>	P3 <sub>2</sub> /D <sub>10</sub>	P3 <sub>2</sub> /D <sub>10</sub>
30	P3 <sub>3</sub> /D <sub>11</sub>	P3 <sub>3</sub> /D <sub>11</sub>	P3 <sub>3</sub> /D <sub>11</sub>	P3 <sub>3</sub> /D <sub>11</sub>	P3 <sub>3</sub> /D <sub>11</sub>
31	P3 <sub>4</sub> /D <sub>12</sub>	P3 <sub>4</sub> /D <sub>12</sub>	P3 <sub>4</sub> /D <sub>12</sub>	P3 <sub>4</sub> /D <sub>12</sub>	P3 <sub>4</sub> /D <sub>12</sub>
32	P3 <sub>5</sub> /D <sub>13</sub>	P3 <sub>5</sub> /D <sub>13</sub>	P3 <sub>5</sub> /D <sub>13</sub>	P3 <sub>5</sub> /D <sub>13</sub>	P3 <sub>5</sub> /D <sub>13</sub>
33	P3 <sub>6</sub> /D <sub>14</sub>	P3 <sub>6</sub> /D <sub>14</sub>	P3 <sub>6</sub> /D <sub>14</sub>	P3 <sub>6</sub> /D <sub>14</sub>	P3 <sub>6</sub> /D <sub>14</sub>
34	P3 <sub>7</sub> /D <sub>15</sub>	P3 <sub>7</sub> /D <sub>15</sub>	P3 <sub>7</sub> /D <sub>15</sub>	P3 <sub>7</sub> /D <sub>15</sub>	P3 <sub>7</sub> /D <sub>15</sub>
35	Vcc	Vcc	Vcc	Vcc	Vcc
36	P1 <sub>0</sub> /A <sub>0</sub>	P1 <sub>0</sub> /A <sub>0</sub>	P1 <sub>0</sub> /A <sub>0</sub>	P1 <sub>0</sub> /A <sub>0</sub>	P1 <sub>0</sub> /A <sub>0</sub>
37	P1 <sub>1</sub> /A <sub>1</sub>	P1 <sub>1</sub> /A <sub>1</sub>	P1 <sub>1</sub> /A <sub>1</sub>	P1 <sub>1</sub> /A <sub>1</sub>	P1 <sub>1</sub> /A <sub>1</sub>
38	P1 <sub>2</sub> /A <sub>2</sub>	P1 <sub>2</sub> /A <sub>2</sub>	P1 <sub>2</sub> /A <sub>2</sub>	P1 <sub>2</sub> /A <sub>2</sub>	P1 <sub>2</sub> /A <sub>2</sub>
39	P1 <sub>3</sub> /A <sub>3</sub>	P1 <sub>3</sub> /A <sub>3</sub>	P1 <sub>3</sub> /A <sub>3</sub>	P1 <sub>3</sub> /A <sub>3</sub>	P1 <sub>3</sub> /A <sub>3</sub>
40	P1 <sub>4</sub> /A <sub>4</sub>	P1 <sub>4</sub> /A <sub>4</sub>	P1 <sub>4</sub> /A <sub>4</sub>	P1 <sub>4</sub> /A <sub>4</sub>	P1 <sub>4</sub> /A <sub>4</sub>
41	P1 <sub>5</sub> /A <sub>5</sub>	P1 <sub>5</sub> /A <sub>5</sub>	P1 <sub>5</sub> /A <sub>5</sub>	P1 <sub>5</sub> /A <sub>5</sub>	P1 <sub>5</sub> /A <sub>5</sub>
42	P1 <sub>6</sub> /A <sub>6</sub>	P1 <sub>6</sub> /A <sub>6</sub>	P1 <sub>6</sub> /A <sub>6</sub>	P1 <sub>6</sub> /A <sub>6</sub>	P1 <sub>6</sub> /A <sub>6</sub>
43	P1 <sub>7</sub> /A <sub>7</sub>	P1 <sub>7</sub> /A <sub>7</sub>	P1 <sub>7</sub> /A <sub>7</sub>	P1 <sub>7</sub> /A <sub>7</sub>	P1 <sub>7</sub> /A <sub>7</sub>
44	Vss	Vss	Vss	Vss	Vss
45	P2 <sub>0</sub> /A <sub>8</sub>	P2 <sub>0</sub> /A <sub>8</sub>	P2 <sub>0</sub> /A <sub>8</sub>	P2 <sub>0</sub> /A <sub>8</sub>	P2 <sub>0</sub> /A <sub>8</sub>
46	P2 <sub>1</sub> /A <sub>9</sub>	P2 <sub>1</sub> /A <sub>9</sub>	P2 <sub>1</sub> /A <sub>9</sub>	P2 <sub>1</sub> /A <sub>9</sub>	P2 <sub>1</sub> /A <sub>9</sub>
47	P2 <sub>2</sub> /A <sub>10</sub>	P2 <sub>2</sub> /A <sub>10</sub>	P2 <sub>2</sub> /A <sub>10</sub>	P2 <sub>2</sub> /A <sub>10</sub>	P2 <sub>2</sub> /A <sub>10</sub>
48	P2 <sub>3</sub> /A <sub>11</sub>	P2 <sub>3</sub> /A <sub>11</sub>	P2 <sub>3</sub> /A <sub>11</sub>	P2 <sub>3</sub> /A <sub>11</sub>	P2 <sub>3</sub> /A <sub>11</sub>
49	P2 <sub>4</sub> /A <sub>12</sub>	P2 <sub>4</sub> /A <sub>12</sub>	P2 <sub>4</sub> /A <sub>12</sub>	P2 <sub>4</sub> /A <sub>12</sub>	P2 <sub>4</sub> /A <sub>12</sub>
50	P2 <sub>5</sub> /A <sub>13</sub>	P2 <sub>5</sub> /A <sub>13</sub>	P2 <sub>5</sub> /A <sub>13</sub>	P2 <sub>5</sub> /A <sub>13</sub>	P2 <sub>5</sub> /A <sub>13</sub>
51	P2 <sub>6</sub> /A <sub>14</sub>	P2 <sub>6</sub> /A <sub>14</sub>	P2 <sub>6</sub> /A <sub>14</sub>	P2 <sub>6</sub> /A <sub>14</sub>	P2 <sub>6</sub> /A <sub>14</sub>
52	P2 <sub>7</sub> /A <sub>15</sub>	P2 <sub>7</sub> /A <sub>15</sub>	P2 <sub>7</sub> /A <sub>15</sub>	P2 <sub>7</sub> /A <sub>15</sub>	P2 <sub>7</sub> /A <sub>15</sub>
53	P5 <sub>0</sub> /A <sub>16</sub>	P5 <sub>0</sub> /A <sub>16</sub>	P5 <sub>0</sub> /A <sub>16</sub>	P5 <sub>0</sub> /A <sub>16</sub>	P5 <sub>0</sub> /A <sub>16</sub>
54	P5 <sub>1</sub> /A <sub>17</sub>	P5 <sub>1</sub> /A <sub>17</sub>	P5 <sub>1</sub> /A <sub>17</sub>	P5 <sub>1</sub> /A <sub>17</sub>	P5 <sub>1</sub> /A <sub>17</sub>
55	P5 <sub>2</sub> /A <sub>18</sub>	P5 <sub>2</sub> /A <sub>18</sub>	P5 <sub>2</sub> /A <sub>18</sub>	P5 <sub>2</sub> /A <sub>18</sub>	P5 <sub>2</sub> /A <sub>18</sub>
56	P5 <sub>3</sub> /A <sub>19</sub>	P5 <sub>3</sub> /A <sub>19</sub>	P5 <sub>3</sub> /A <sub>19</sub>	P5 <sub>3</sub> /A <sub>19</sub>	P5 <sub>3</sub> /A <sub>19</sub>
57	Vss	Vss	Vss	Vss	Vss
58	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT	P6 <sub>0</sub> /WAIT
59	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ	P6 <sub>1</sub> /BREQ
60	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK	P6 <sub>2</sub> /BACK
61	P6 <sub>7</sub> /	P6 <sub>7</sub> /	P6 <sub>7</sub> /		
62	STBY	STBY	STBY	STBY	STBY
63	RES	RES	RES	RES	RES
64	NMI	NMI	NMI	NMI	NMI
65	Vss	Vss	Vss	Vss	Vss

ピン番号	H8/3068F	H8/3067 シリーズ	H8/3062 シリーズ	H8/3048 シリーズ	H8/3042 シリーズ
66	EXTAL	EXTAL	EXTAL	EXTAL	EXTAL
67	XTAL	XTAL	XTAL	XTAL	XTAL
68	Vcc	Vcc	Vcc	Vcc	Vcc
69	P6 <sub>3</sub> /AS	P6 <sub>3</sub> /AS	P6 <sub>3</sub> /AS	P6 <sub>3</sub> /AS	P6 <sub>3</sub> /AS
70	P6 <sub>4</sub> /RD	P6 <sub>4</sub> /RD	P6 <sub>4</sub> /RD	P6 <sub>4</sub> /RD	P6 <sub>4</sub> /RD
71	P6 <sub>5</sub> /HWR	P6 <sub>5</sub> /HWR	P6 <sub>5</sub> /HWR	P6 <sub>5</sub> /HWR	P6 <sub>5</sub> /HWR
72	P6 <sub>6</sub> /LWR	P6 <sub>6</sub> /LWR	P6 <sub>6</sub> /LWR	P6 <sub>6</sub> /LWR	P6 <sub>6</sub> /LWR
73	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>	MD <sub>0</sub>
74	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>	MD <sub>1</sub>
75	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>	MD <sub>2</sub>
76	AVcc	AVcc	AVcc	AVcc	AVcc
77	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>	V <sub>REF</sub>
78	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>	P7 <sub>0</sub> /AN <sub>0</sub>
79	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>	P7 <sub>1</sub> /AN <sub>1</sub>
80	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>	P7 <sub>2</sub> /AN <sub>2</sub>
81	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>	P7 <sub>3</sub> /AN <sub>3</sub>
82	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>	P7 <sub>4</sub> /AN <sub>4</sub>
83	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>	P7 <sub>5</sub> /AN <sub>5</sub>
84	P7 <sub>6</sub> /AN <sub>6</sub> /DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> /DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> /DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> /DA <sub>0</sub>	P7 <sub>6</sub> /AN <sub>6</sub> /DA <sub>0</sub>
85	P7 <sub>7</sub> /AN <sub>7</sub> /DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> /DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> /DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> /DA <sub>1</sub>	P7 <sub>7</sub> /AN <sub>7</sub> /DA <sub>1</sub>
86	AVss	AVss	AVss	AVss	AVss
87	P8 <sub>0</sub> /RFSH/IRQ <sub>0</sub>	P8 <sub>0</sub> /RFSH/IRQ <sub>0</sub>	P8 <sub>0</sub> /IRQ <sub>0</sub>	P8 <sub>0</sub> /RFSH/IRQ <sub>0</sub>	P8 <sub>0</sub> /RFSH/IRQ <sub>0</sub>
88	P8 <sub>1</sub> /CS <sub>3</sub> /IRQ <sub>1</sub>	P8 <sub>1</sub> /CS <sub>3</sub> /IRQ <sub>1</sub>	P8 <sub>1</sub> /CS <sub>3</sub> /IRQ <sub>1</sub>	P8 <sub>1</sub> /CS <sub>3</sub> /IRQ <sub>1</sub>	P8 <sub>1</sub> /CS <sub>3</sub> /IRQ <sub>1</sub>
89	P8 <sub>2</sub> /CS <sub>2</sub> /IRQ <sub>2</sub>	P8 <sub>2</sub> /CS <sub>2</sub> /IRQ <sub>2</sub>	P8 <sub>2</sub> /CS <sub>2</sub> /IRQ <sub>2</sub>	P8 <sub>2</sub> /CS <sub>2</sub> /IRQ <sub>2</sub>	P8 <sub>2</sub> /CS <sub>2</sub> /IRQ <sub>2</sub>
90	P8 <sub>3</sub> /CS <sub>1</sub> /IRQ <sub>3</sub> / ADTRG	P8 <sub>3</sub> /CS <sub>1</sub> /IRQ <sub>3</sub> / ADTRG	P8 <sub>3</sub> /CS <sub>1</sub> /IRQ <sub>3</sub> / ADTRG	P8 <sub>3</sub> /CS <sub>1</sub> /IRQ <sub>3</sub>	P8 <sub>3</sub> /CS <sub>1</sub> /IRQ <sub>3</sub>
91	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>	P8 <sub>4</sub> /CS <sub>0</sub>
92	Vss	Vss	Vss	Vss	Vss
93	PA <sub>0</sub> /TP <sub>0</sub> / TEND <sub>0</sub> /TCLKA	PA <sub>0</sub> /TP <sub>0</sub> / TEND <sub>0</sub> /TCLKA	PA <sub>0</sub> /TP <sub>0</sub> /TCLKA	PA <sub>0</sub> /TP <sub>0</sub> / TEND <sub>0</sub> /TCLKA	PA <sub>0</sub> /TP <sub>0</sub> / TEND <sub>0</sub> /TCLKA
94	PA <sub>1</sub> /TP <sub>1</sub> / TEND <sub>1</sub> /TCLKB	PA <sub>1</sub> /TP <sub>1</sub> / TEND <sub>1</sub> /TCLKB	PA <sub>1</sub> /TP <sub>1</sub> /TCLKB	PA <sub>1</sub> /TP <sub>1</sub> / TEND <sub>1</sub> /TCLKB	PA <sub>1</sub> /TP <sub>1</sub> / TEND <sub>1</sub> /TCLKB
95	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> /TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> /TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> /TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> /TCLKC	PA <sub>2</sub> /TP <sub>2</sub> / TIOCA <sub>0</sub> /TCLKC

ピン番号	H8/3068F	H8/3067 シリーズ	H8/3062 シリーズ	H8/3048 シリーズ	H8/3042 シリーズ
96	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> /TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> /TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> /TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> /TCLKD	PA <sub>3</sub> /TP <sub>3</sub> / TIOCB <sub>0</sub> /TCLKD
97	PA <sub>4</sub> /TP <sub>4</sub> /TIOCA <sub>1</sub> / A <sub>23</sub>	PA <sub>4</sub> /TP <sub>4</sub> /TIOCA <sub>1</sub> / A <sub>23</sub>	PA <sub>4</sub> /TP <sub>4</sub> /TIOCA <sub>1</sub> / A <sub>23</sub>	PA <sub>4</sub> /TP <sub>4</sub> /TIOCA <sub>1</sub> / $\overline{CS}_6/A_{23}$	PA <sub>4</sub> /TP <sub>4</sub> /TIOCA <sub>1</sub> / A <sub>23</sub>
98	PA <sub>5</sub> /TP <sub>5</sub> /TIOCB <sub>1</sub> / A <sub>22</sub>	PA <sub>5</sub> /TP <sub>5</sub> /TIOCB <sub>1</sub> / A <sub>22</sub>	PA <sub>5</sub> /TP <sub>5</sub> /TIOCB <sub>1</sub> / A <sub>22</sub>	PA <sub>5</sub> /TP <sub>5</sub> /TIOCB <sub>1</sub> / $\overline{CS}_5/A_{22}$	PA <sub>5</sub> /TP <sub>5</sub> /TIOCB <sub>1</sub> / A <sub>22</sub>
99	PA <sub>6</sub> /TP <sub>6</sub> /TIOCA <sub>2</sub> / A <sub>21</sub>	PA <sub>6</sub> /TP <sub>6</sub> /TIOCA <sub>2</sub> / A <sub>21</sub>	PA <sub>6</sub> /TP <sub>6</sub> /TIOCA <sub>2</sub> / A <sub>21</sub>	PA <sub>6</sub> /TP <sub>6</sub> /TIOCA <sub>2</sub> / $\overline{CS}_4/A_{21}$	PA <sub>6</sub> /TP <sub>6</sub> /TIOCA <sub>2</sub> / A <sub>21</sub>
100	PA <sub>7</sub> /TP <sub>7</sub> /TIOCB <sub>2</sub> / A <sub>20</sub>	PA <sub>7</sub> /TP <sub>7</sub> /TIOCB <sub>2</sub> / A <sub>20</sub>	PA <sub>7</sub> /TP <sub>7</sub> /TIOCB <sub>2</sub> / A <sub>20</sub>	PA <sub>7</sub> /TP <sub>7</sub> /TIOCB <sub>2</sub> / A <sub>20</sub>	PA <sub>7</sub> /TP <sub>7</sub> /TIOCB <sub>2</sub> / A <sub>20</sub>

【注】 \*1 マスク ROM 版は  $\overline{RES}_0$ 、フラッシュメモリ版およびフラッシュメモリ R 版は FWE として機能します。





H8/3068 F-ZTAT™ ハードウェアマニュアル

発行年月 2001年 3月 第 1 版

発行 株式会社 日立製作所  
半導体グループ電子統括営業本部

編集 株式会社 日立小平セミコン  
技術ドキュメントグループ

©株式会社 日立製作所 2001