

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

お客様各位

資料中の「日立製作所」、「日立XX」等名称の株式会社ルネサス テクノロジへの変更について

2003年4月1日を以って三菱電機株式会社及び株式会社日立製作所のマイコン、ロジック、アナログ、ディスクリート半導体、及びDRAMを除くメモリ(フラッシュメモリ・SRAM等)を含む半導体事業は株式会社ルネサス テクノロジに承継されました。従いまして、本資料中には「日立製作所」、「株式会社日立製作所」、「日立半導体」、「日立XX」といった表記が残っておりますが、これらの表記は全て「株式会社ルネサス テクノロジ」に変更されておりますのでご理解の程お願い致します。尚、会社商標・ロゴ・コーポレートステートメント以外の内容については一切変更しておりませんので資料としての内容更新ではありません。

ルネサステクノロジ ホームページ (<http://www.renesas.com>)

2003年4月1日
株式会社ルネサス テクノロジ
カスタマサポート部

ご注意

安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご注意ください。

本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ (<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したのですが万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気付きの点がございましたらルネサス テクノロジ、ルネサス販売または特約店までご照会ください。

H8/3035 シリーズ

ハードウェアマニュアル

はじめに

H8/3035 シリーズは、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 ビットインテグレートドタイマユニット (ITU)、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、I/O ポートなどを内蔵しています。

MCU 動作モードは、モード 1~3 があり、拡張モードとシングルチップモードを選択することができます。

このため、H8/3035 シリーズ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。

本マニュアルは、H8/3035 シリーズ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。

目次

第1章 概要

1.1	概要	3	
1.2	内部ブロック図	7	
1.3	端子説明	8	
	1.3.1	ピン配置	8
	1.3.2	端子機能	9
1.4	端子機能	12	

第2章 CPU

2.1	概要	19	
	2.1.1	特長	19
	2.1.2	H8/300 CPU との相違点	20
2.2	CPU 動作モード	21	
2.3	アドレス空間	22	
2.4	レジスタ構成	23	
	2.4.1	概要	23
	2.4.2	汎用レジスタ	24
	2.4.3	コントロールレジスタ	25
	2.4.4	CPU 内部レジスタの初期値	26
2.5	データ構成	27	
	2.5.1	汎用レジスタのデータ構成	28
	2.5.2	メモリ上でのデータ構成	29
2.6	命令セット	30	
	2.6.1	命令セットの概要	30
	2.6.2	命令とアドレッシングモードの組み合わせ	30
	2.6.3	命令の機能別一覧	32
	2.6.4	命令の基本フォーマット	40

2.6.5	ビット操作命令使用上の注意.....	41
2.7	アドレッシングモードと実効アドレスの計算方法.....	42
2.7.1	アドレッシングモード.....	42
2.7.2	実効アドレスの計算方法.....	45
2.8	処理状態.....	49
2.8.1	概要.....	49
2.8.2	プログラム実行状態.....	49
2.8.3	例外処理状態.....	49
2.8.4	例外処理の動作.....	51
2.8.5	リセット状態.....	52
2.8.6	低消費電力状態.....	52
2.9	基本動作タイミング.....	54
2.9.1	概要.....	54
2.9.2	内蔵メモリアクセスタイミング.....	54
2.9.3	内蔵周辺モジュールアクセスタイミング.....	55
2.9.4	外部アドレス空間アクセスタイミング.....	56

第3章 MCU 動作モード

3.1	概要.....	59
3.1.1	動作モードの選択の種類.....	59
3.1.2	レジスタ構成.....	59
3.2	モードコントロールレジスタ (MDCR).....	60
3.3	システムコントロールレジスタ (SYSCR).....	61
3.4	各動作モードの説明.....	64
3.4.1	モード 1.....	64
3.4.2	モード 2.....	64
3.4.3	モード 3.....	64
3.5	各動作モードにおける端子機能.....	65
3.6	各動作モードのメモリマップ.....	66

第4章 例外処理

4.1	概要.....	71
4.1.1	例外処理の種類と優先度.....	71
4.1.2	例外処理の動作.....	71
4.1.3	例外処理要因とベクタテーブル.....	71
4.2	リセット.....	73

4.2.1	概要	73
4.2.2	リセットシーケンス	73
4.2.3	リセット直後の割込み	75
4.3	割込み	76
4.4	トラップ命令	77
4.5	例外処理後のスタックの状態	78
4.6	スタック使用上の注意	79

第5章 割込みコントローラ

5.1	概要	83
5.1.1	特長	83
5.1.2	ブロック図	83
5.1.3	端子構成	84
5.1.4	レジスタ構成	85
5.2	各レジスタの説明	86
5.2.1	システムコントロールレジスタ (SYSCR)	86
5.2.2	インタラプトプライオリティレジスタ A、B (IPRA、IPRB)	87
5.2.3	IRQ ステータスレジスタ (ISR)	93
5.2.4	IRQ イネーブルレジスタ (IER)	94
5.2.5	IRQ センスコントロールレジスタ (ISCR)	95
5.3	割込み要因	96
5.3.1	外部割込み	96
5.3.2	内部割込み	97
5.3.3	割込み例外処理ベクタテーブル	97
5.4	割込み動作	100
5.4.1	割込み動作の流れ	100
5.4.2	割込み例外処理シーケンス	105
5.4.3	割込み応答時間	106
5.5	使用上の注意	107
5.5.1	割込みの発生とディスエーブルとの競合	107
5.5.2	割込みの受け付けを禁止している命令	107
5.5.3	EEPMOV 命令実行中の割込み	108
5.5.4	外部割込み使用上の注意	108

第6章 バスコントローラ

6.1	概要	113
-----	----------	-----

6.1.1	特長	113
6.1.2	ブロック図	113
6.1.3	端子構成	114
6.1.4	レジスタ構成	114
6.2	各レジスタの説明	115
6.2.1	アクセスステートコントロールレジスタ (ASTCR)	115
6.2.2	ウェイトコントロールレジスタ (WCR)	116
6.2.3	ウェイトステートコントローラインーブルレジスタ (WCER)	117
6.2.4	アドレスコントロールレジスタ (ADRCR)	117
6.3	動作説明	119
6.3.1	エリア分割	119
6.3.2	バス制御信号タイミング	120
6.3.3	ウェイトモード	122
6.3.4	メモリとの接続例	128
6.4	使用上の注意	129
6.4.1	レジスタライトタイミング	129
6.4.2	ASTCR と ABWCR*設定上の注意事項	129

第7章 I/Oポート

7.1	概要	133
7.2	ポート 1	135
7.2.1	概要	135
7.2.2	レジスタの構成と説明	135
7.2.3	モード別端子機能	137
7.3	ポート 2	139
7.3.1	概要	139
7.3.2	レジスタの構成と説明	139
7.3.3	モード別端子機能	142
7.3.4	入力プルアップ MOS	143
7.4	ポート 3	144
7.4.1	概要	144
7.4.2	レジスタの構成と説明	144
7.4.3	モード別端子機能	146
7.5	ポート 5	148
7.5.1	概要	148
7.5.2	レジスタの構成と説明	148
7.5.3	モード別端子機能	151

	7.5.4	入力プルアップ MOS.....	152
7.6		ポート 6.....	153
	7.6.1	概要.....	153
	7.6.2	レジスタの構成と説明.....	153
	7.6.3	モード別端子機能.....	156
7.7		ポート 7.....	159
	7.7.1	概要.....	159
	7.7.2	レジスタの構成と説明.....	159
7.8		ポート 8.....	160
	7.8.1	概要.....	160
	7.8.2	レジスタの構成と説明.....	160
	7.8.3	モード別端子機能.....	163
7.9		ポート 9.....	164
	7.9.1	概要.....	164
	7.9.2	レジスタの構成と説明.....	164
	7.9.3	端子機能.....	166
7.10		ポート A.....	167
	7.10.1	概要.....	167
	7.10.2	レジスタの構成と説明.....	168
	7.10.3	端子機能.....	169
7.11		ポート B.....	174
	7.11.1	概要.....	174
	7.11.2	レジスタの構成と説明.....	174
	7.11.3	端子機能.....	176

第 8 章 16 ビットインテグレートドタイマユニット (ITU)

8.1		概要.....	181
	8.1.1	特長.....	181
	8.1.2	ブロック図.....	184
	8.1.3	端子構成.....	189
	8.1.4	レジスタ構成.....	191
8.2		各レジスタの説明.....	193
	8.2.1	タイマスタートレジスタ (TSTR).....	193
	8.2.2	タイマシンクロレジスタ (TSNC).....	195
	8.2.3	タイマモードレジスタ (TMDR).....	197
	8.2.4	タイマファンクションコントロールレジスタ (TFCR).....	201
	8.2.5	タイマアウトプットマスタイネーブルレジスタ (TOER).....	204

8.2.6	タイマアウトプットコントロールレジスタ (TOCR)	207
8.2.7	タイマカウンタ (TCNT)	209
8.2.8	ジェネラルレジスタ A、B (GRA、GRB)	210
8.2.9	バッファレジスタ A、B (BRA、BRB)	211
8.2.10	タイマコントロールレジスタ (TCR)	212
8.2.11	タイマI/O コントロールレジスタ (TIOR)	215
8.2.12	タイマステータスレジスタ (TSR)	218
8.2.13	タイマインタラプトイネーブルレジスタ (TIER)	221
8.3	CPU とのインタフェース.....	223
8.3.1	16 ビットアクセス可能なレジスタ	223
8.3.2	8 ビットアクセスのレジスタ.....	224
8.4	動作説明.....	226
8.4.1	概要	226
8.4.2	基本機能.....	227
8.4.3	同期動作.....	235
8.4.4	PWM モード	236
8.4.5	リセット同期 PWM モード.....	240
8.4.6	相補 PWM モード	243
8.4.7	位相計数モード.....	250
8.4.8	バッファ動作	251
8.4.9	ITU 出力タイミング.....	256
8.5	割込み.....	259
8.5.1	ステータスフラグのセットタイミング	259
8.5.2	ステータスフラグのクリアタイミング	261
8.5.3	割込み要因	261
8.6	使用上の注意.....	263

第9章 プログラマブルタイミングパターンコントローラ (TPC)

9.1	概要	281
9.1.1	特長	281
9.1.2	ブロック図	282
9.1.3	端子構成.....	283
9.1.4	レジスタ構成	284
9.2	各レジスタの説明	285
9.2.1	ポート A データディレクションレジスタ (PADDDR)	285
9.2.2	ポート A データレジスタ (PADR)	285
9.2.3	ポート B データディレクションレジスタ (PBDDR)	286

9.2.4	ポート B データレジスタ (PBDR)	286
9.2.5	ネクストデータレジスタ A (NDRA)	286
9.2.6	ネクストデータレジスタ B (NDRB)	288
9.2.7	ネクストデータイネーブルレジスタ A (NDERA)	290
9.2.8	ネクストデータイネーブルレジスタ B (NDERB)	290
9.2.9	TPC 出力コントロールレジスタ (TPCR)	291
9.2.10	TPC 出力モードレジスタ (TPMR)	294
9.3	動作説明.....	297
9.3.1	概要	297
9.3.2	出力タイミング.....	298
9.3.3	TPC 出力通常動作.....	299
9.3.4	TPC 出力ノンオーバーラップ動作.....	301
9.3.5	インプットキャプチャによる TPC 出力	303
9.4	使用上の注意.....	304
9.4.1	TPC 出力端子の動作.....	304
9.4.2	ノンオーバーラップ動作時の注意	304

第 10 章 ウォッチドッグタイマ

10.1	概要	309
10.1.1	特長	309
10.1.2	ブロック図	310
10.1.3	端子構成.....	310
10.1.4	レジスタ構成	311
10.2	各レジスタの説明	312
10.2.1	タイマカウンタ (TCNT)	312
10.2.2	タイマコントロール/ステータスレジスタ (TCSR)	312
10.2.3	リセットコントロール/ステータスレジスタ (RSTCSR)	314
10.2.4	レジスタ書換え時の注意	315
10.3	動作説明.....	318
10.3.1	ウォッチドッグタイマ時の動作	318
10.3.2	インターバルタイマ時の動作.....	319
10.3.3	オーバフローフラグ (OVF) セットタイミング.....	320
10.3.4	ウォッチドッグタイマリセット (WRST) のセットタイミング	321
10.4	割込み.....	322
10.5	使用上の注意.....	322

第 11 章 シリアルコミュニケーションインタフェース

11.1	概要	325
11.1.1	特長	325
11.1.2	ブロック図	326
11.1.3	端子構成	327
11.1.4	レジスタ構成	327
11.2	各レジスタの説明	328
11.2.1	レシーブシフトレジスタ (RSR)	328
11.2.2	レシーブデータレジスタ (RDR)	328
11.2.3	トランスミットシフトレジスタ (TSR)	329
11.2.4	トランスミットデータレジスタ (TDR)	329
11.2.5	シリアルモードレジスタ (SMR)	330
11.2.6	シリアルコントロールレジスタ (SCR)	334
11.2.7	シリアルステータスレジスタ (SSR)	338
11.2.8	ビットレートレジスタ (BRR)	344
11.3	動作説明	352
11.3.1	概要	352
11.3.2	調歩同期式モード時の動作	354
11.3.3	マルチプロセッサ通信機能	363
11.3.4	クロック同期式モード時の動作	369
11.4	SCI 割込み	378
11.5	使用上の注意	379

第 12 章 A/D 変換器

12.1	概要	385
12.1.1	特長	385
12.1.2	ブロック図	386
12.1.3	端子構成	386
12.1.4	レジスタ構成	387
12.2	各レジスタの説明	388
12.2.1	A/D データレジスタ A ~ D (ADDRA ~ D)	388
12.2.2	A/D コントロール / ステータスレジスタ (ADCSR)	389
12.2.3	A/D コントロールレジスタ (ADCR)	391
12.3	CPU とのインタフェース	392
12.4	動作説明	393
12.4.1	単一モード (SCAN = "0")	393
12.4.2	スキャンモード (SCAN = "1")	395

12.4.3	入力サンプリングと A/D 変換時間.....	397
12.4.4	外部トリガ入力タイミング.....	398
12.5	割込み.....	399
12.6	使用上の注意.....	399

第 13 章 RAM

13.1	概要.....	403
13.1.1	ブロック図.....	403
13.1.2	レジスタ構成.....	404
13.2	システムコントロールレジスタ (SYSCR).....	405
13.3	動作説明.....	406
13.3.1	モード 1.....	406
13.3.2	モード 2.....	406
13.3.3	モード 3.....	406

第 14 章 ROM

14.1	概要.....	409
14.1.1	ブロック図.....	409
14.2	PROM モード.....	410
14.2.1	PROM モードの設定.....	410
14.2.2	ソケットアダプタの端子対応とメモリマップ.....	410
14.3	プログラミング.....	413
14.3.1	書込み / ベリファイ.....	413
14.3.2	書込み時の注意.....	417
14.4	書込み後の信頼性.....	418

第 15 章 クロック発振器

15.1	概要.....	421
15.1.1	ブロック図.....	421
15.2	発振器.....	422
15.2.1	水晶発振子を接続する方法.....	422
15.2.2	外部クロックを入力する方法.....	423
15.3	デューティ補正回路.....	426
15.4	プリスケラ.....	426

第 16 章 低消費電力状態

16.1	概要	429
16.2	レジスタ構成	430
16.2.1	システムコントロールレジスタ (SYSCR)	430
16.3	スリープモード	432
16.3.1	スリープモードへの遷移	432
16.3.2	スリープモードの解除	432
16.4	ソフトウェアスタンバイモード	433
16.4.1	ソフトウェアスタンバイモードへの遷移	433
16.4.2	ソフトウェアスタンバイモードの解除	433
16.4.3	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	434
16.4.4	ソフトウェアスタンバイモードの応用例	435
16.4.5	使用上の注意	435
16.5	ハードウェアスタンバイモード	436
16.5.1	ハードウェアスタンバイモードへの遷移	436
16.5.2	ハードウェアスタンバイモードの解除	436
16.5.3	ハードウェアスタンバイモードのタイミング	436

第 17 章 電気的特性 (暫定仕様)

17.1	絶対最大定格	439
17.2	電気的特性	440
17.2.1	DC 特性	440
17.2.2	AC 特性	447
17.2.3	A/D 変換特性	453
17.3	動作タイミング	454
17.3.1	バスタイミング	454
17.3.2	制御信号タイミング	456
17.3.3	クロックタイミング	457
17.3.4	TPC、I/O ポートタイミング	458
17.3.5	ITU タイミング	458
17.3.6	SCI 入出力タイミング	459

付録

A.	命令	463
A.1	命令一覧	463
A.2	オペレーションコードマップ	478

	A.3	命令実行ステート数.....	481
B.		内部 I/O レジスタ一覧.....	492
	B.1	アドレス一覧.....	492
	B.2	機能一覧.....	499
C.		I/O ポートブロック図.....	553
	C.1	ポート 1 ブロック図.....	553
	C.2	ポート 2 ブロック図.....	554
	C.3	ポート 3 ブロック図.....	555
	C.4	ポート 5 ブロック図.....	556
	C.5	ポート 6 ブロック図.....	557
	C.6	ポート 7 ブロック図.....	559
	C.7	ポート 8 ブロック図.....	560
	C.8	ポート 9 ブロック図.....	562
	C.9	ポート A ブロック図.....	565
	C.10	ポート B ブロック図.....	568
D.		端子状態.....	572
	D.1	各処理状態におけるポートの状態.....	572
	D.2	リセット時の端子状態.....	574
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.....	577
F.		ROM 発注手順.....	578
	F.1	ROM 書き換え品開発の流れ (発注手順).....	578
	F.2	ROM 発注時の注意事項.....	579
G.		型名一覧.....	580
H.		外形寸法図.....	581

1. 概要

第1章 目次

1.1	概要.....	3
1.2	内部ブロック図.....	7
1.3	端子説明.....	8
	1.3.1 ピン配置.....	8
	1.3.2 端子機能.....	9
1.4	端子機能.....	12

1.1 概要

H8/3035 シリーズは、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 ビットインテグレートドタイマユニット (ITU)、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、I/O ポート、などを内蔵しています。

H8/3035 シリーズには、H8/3035、H8/3034、H8/3033 の 3 種類があります。H8/3035 には、256k バイト ROM と 4k バイト RAM、H8/3034 には、192k バイト ROM と 4k バイト RAM、H8/3033 には、128k バイト ROM と 4k バイト RAM がそれぞれ内蔵されています。

MCU 動作モードは、モード 1~3 があり、拡張モードとシングルチップモードを選択することができます。

H8/3035 シリーズには、マスク ROM 版のほかに、ユーザサイドで自由にプログラムの書込みができる PROM を内蔵した ZTAT[®]* 版があります。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。

H8/3035 シリーズの特長を表 1.1 に示します。

【注】 * ZTAT は (株) 日立製作所の登録商標です。

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汎用レジスタマシン</p> <ul style="list-style-type: none"> ・汎用レジスタ：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) <p>高速動作</p> <ul style="list-style-type: none"> ・最大動作周波数：18MHz ・加減算：111ns ・乗除算：778ns <p>2 種類の CPU 動作モード</p> <ul style="list-style-type: none"> ・ノーマルモード (アドレス空間 64k バイト)* ・アドバンスモード (アドレス空間 16M バイト) <p>特長ある命令</p> <ul style="list-style-type: none"> ・8/16/32 ビット転送・演算命令 ・符号なし / 符号付乗算命令 (8 ビット×8 ビット、16 ビット×16 ビット) ・符号なし / 符号付除算命令 (16 ビット÷8 ビット、32 ビット÷16 ビット) ・ビットアキュムレータ機能 ・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
メモリ	<p>H8/3035</p> <ul style="list-style-type: none"> ・ROM：256k バイト ・RAM：4k バイト <p>H8/3034</p> <ul style="list-style-type: none"> ・ROM：192k バイト ・RAM：4k バイト <p>H8/3033</p> <ul style="list-style-type: none"> ・ROM：128k バイト ・RAM：4k バイト
割込みコントローラ	<ul style="list-style-type: none"> ・外部割込み端子 6 本：NMI、$\overline{IRQ}_0 \sim \overline{IRQ}_4$ ・内部割込み 21 要因 ・3 レベルの割込み優先順位が設定可能

項目	仕様
バスコントローラ	<ul style="list-style-type: none"> ・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 ・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 ・4 種類のウェイトモードを設定可能
16 ビット インテグレートッド タイマユニット (ITU)	<ul style="list-style-type: none"> ・16 ビットタイマ 5 チャンネルを内蔵。最大 12 種類のパルス出力、または最大 10 種類のパルスの入力処理が可能 ・16 ビットタイマカウンタ×1 (チャンネル 0~4) ・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 0~4) ・同期動作可能 (チャンネル 0~4) ・PWM モード設定可能 (チャンネル 0~4) ・位相計数モード設定可能 (チャンネル 2) ・バッファ動作可能 (チャンネル 3、4) ・リセット同期 PWM モード設定可能 (チャンネル 3、4) ・相補 PWM モード設定可能 (チャンネル 3、4)
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"> ・ITU 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 ・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) ・ノンオーバーラップモード設定可能
ウォッチドッグ タイマ (WDT) × 1 チャンネル	<ul style="list-style-type: none"> ・オーバフローによりリセット信号を発生可能 ・リセット信号の外部出力可能 ・インターバルタイマとして使用可能
シリアル コミュニケーション インタフェース (SCI) × 1 チャンネル	<ul style="list-style-type: none"> ・調歩同期 / クロック同期式モードの選択可能 ・送受信同時動作 (全二重動作) 可能 ・専用のポーレートジェネレータ内蔵
A/D 変換器	<ul style="list-style-type: none"> ・分解能 : 10 ビット ・8 チャンネル : 単一モード / スキャンモード選択可能 ・アナログ変換電圧範囲の設定が可能 ・サンプル&ホールド機能付 ・外部トリガによる A/D 変換開始可能
I/O ポート	<ul style="list-style-type: none"> ・入出力端子 55 本 ・入力端子 8 本

1. 概要

項目	仕様																							
動作モード	3種類のMCU動作モード <table border="1" style="margin-left: 20px;"> <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</th> </tr> </thead> <tbody> <tr> <td>モード1</td> <td>1Mバイト</td> <td>A₁₉ ~ A₀</td> <td>8ビット</td> </tr> <tr> <td>モード2</td> <td>16Mバイト</td> <td>A₂₃ ~ A₀</td> <td>8ビット</td> </tr> <tr> <td>モード3</td> <td>1Mバイト</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	モード	アドレス空間	アドレス端子	バス幅	モード1	1Mバイト	A ₁₉ ~ A ₀	8ビット	モード2	16Mバイト	A ₂₃ ~ A ₀	8ビット	モード3	1Mバイト	—	—							
モード	アドレス空間	アドレス端子	バス幅																					
モード1	1Mバイト	A ₁₉ ~ A ₀	8ビット																					
モード2	16Mバイト	A ₂₃ ~ A ₀	8ビット																					
モード3	1Mバイト	—	—																					
低消費電力状態	<ul style="list-style-type: none"> ・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 																							
その他	<ul style="list-style-type: none"> ・クロック発振器内蔵 																							
製品ラインアップ	<table border="1" style="margin-left: 20px;"> <thead> <tr> <th>製品型名</th> <th>パッケージ</th> <th>ROM</th> </tr> </thead> <tbody> <tr> <td>HD6473035F</td> <td>80ピンQFP (FP-80A)</td> <td rowspan="2">PROM版</td> </tr> <tr> <td>HD6473035TE</td> <td>80ピンTQFP (TFP-80C)</td> </tr> <tr> <td>HD6433035F</td> <td>80ピンQFP (FP-80A)</td> <td rowspan="2">マスクROM版</td> </tr> <tr> <td>HD6433035TE</td> <td>80ピンTQFP (TFP-80C)</td> </tr> <tr> <td>HD6433034F</td> <td>80ピンQFP (FP-80A)</td> <td rowspan="2">マスクROM版</td> </tr> <tr> <td>HD6433034TE</td> <td>80ピンTQFP (TFP-80C)</td> </tr> <tr> <td>HD6433033F</td> <td>80ピンQFP (FP-80A)</td> <td rowspan="2">マスクROM版</td> </tr> <tr> <td>HD6433033TE</td> <td>80ピンTQFP (TFP-80C)</td> </tr> </tbody> </table>	製品型名	パッケージ	ROM	HD6473035F	80ピンQFP (FP-80A)	PROM版	HD6473035TE	80ピンTQFP (TFP-80C)	HD6433035F	80ピンQFP (FP-80A)	マスクROM版	HD6433035TE	80ピンTQFP (TFP-80C)	HD6433034F	80ピンQFP (FP-80A)	マスクROM版	HD6433034TE	80ピンTQFP (TFP-80C)	HD6433033F	80ピンQFP (FP-80A)	マスクROM版	HD6433033TE	80ピンTQFP (TFP-80C)
製品型名	パッケージ	ROM																						
HD6473035F	80ピンQFP (FP-80A)	PROM版																						
HD6473035TE	80ピンTQFP (TFP-80C)																							
HD6433035F	80ピンQFP (FP-80A)	マスクROM版																						
HD6433035TE	80ピンTQFP (TFP-80C)																							
HD6433034F	80ピンQFP (FP-80A)	マスクROM版																						
HD6433034TE	80ピンTQFP (TFP-80C)																							
HD6433033F	80ピンQFP (FP-80A)	マスクROM版																						
HD6433033TE	80ピンTQFP (TFP-80C)																							

【注】 * H8/3035 シリーズではノーマルモードは使用できません。

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。

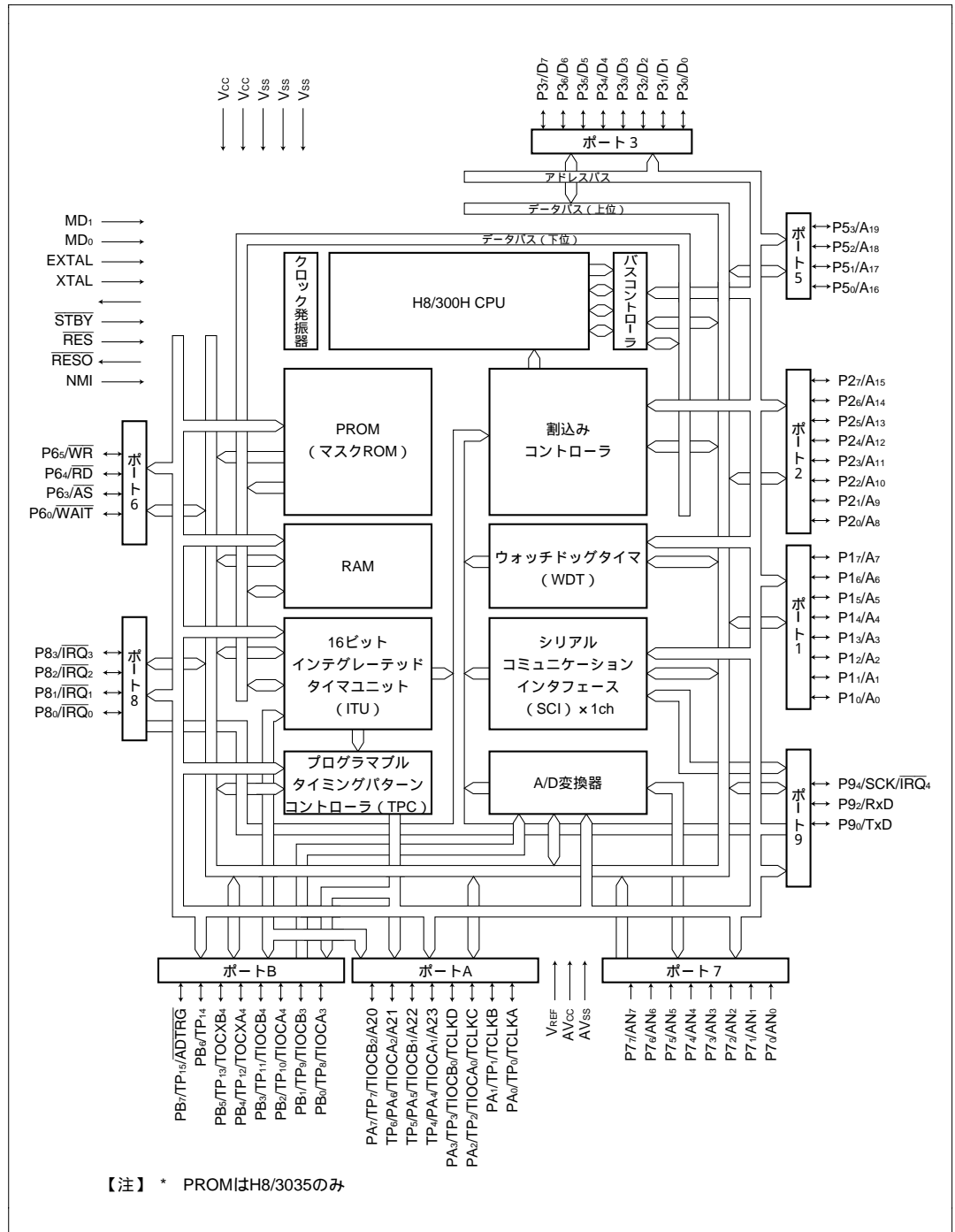


図 1.1 内部ブロック図

1.3 端子説明

1.3.1 ピン配置

H8/3035 シリーズのピン配置図を図 1.2 に示します。

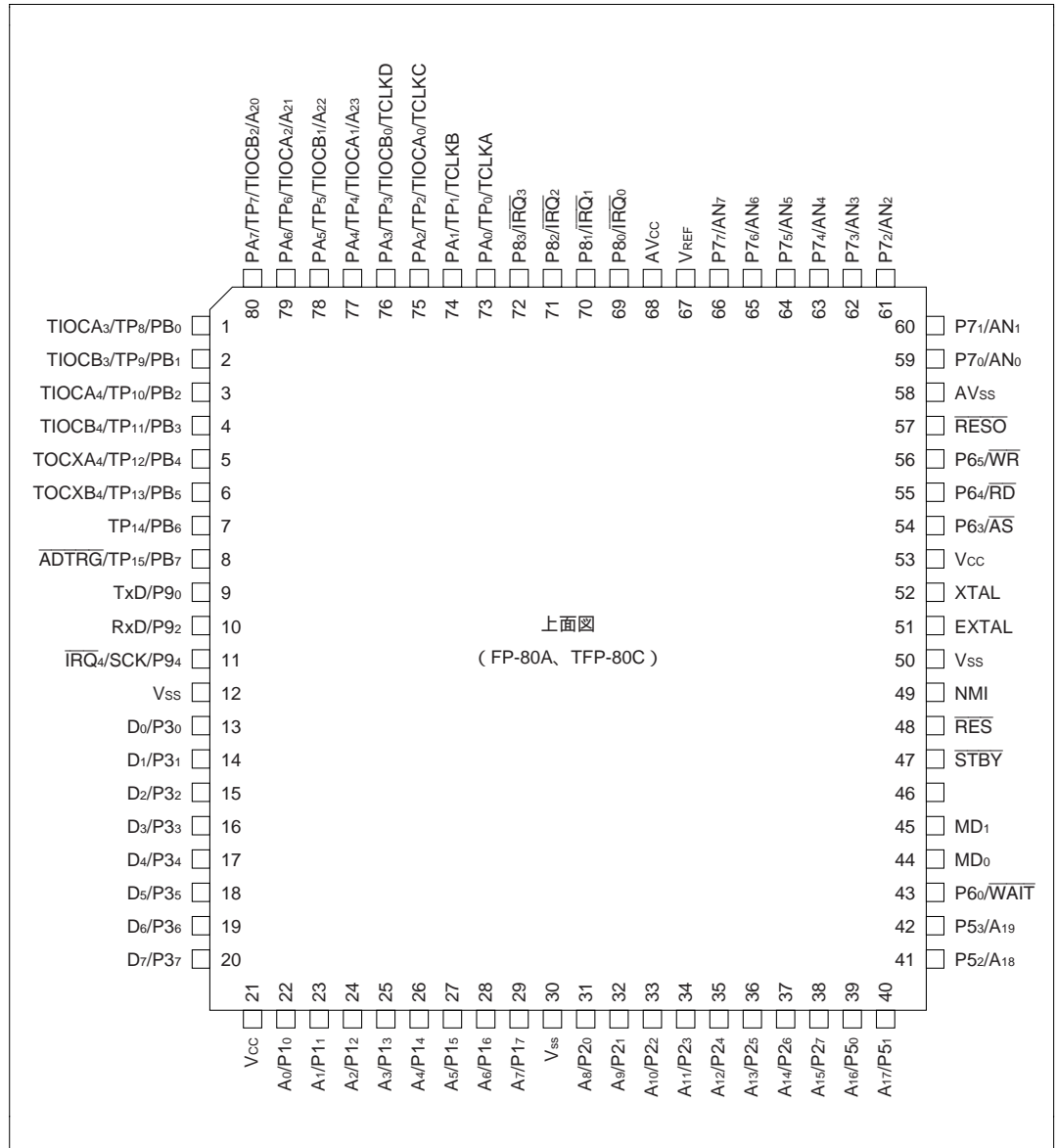


図 1.2 ピン配置 (FP-80A、TFP-80C)

1.3.2 端子機能

(1) モード別ピン配置一覧

モード別ピン配置 (FP-80A、TFP-80C) を表 1.2 に示します。

表 1.2 モード別ピン配置一覧 (FP-80A、TFP-80C)

ピン 番号	端子名			
	モード 1	モード 2	モード 3	PROM モード
1	PB ₀ /TP ₈ /TIOCA ₃	PB ₀ /TP ₈ /TIOCA ₃	PB ₀ /TP ₈ /TIOCA ₃	NC
2	PB ₁ /TP ₉ /TIOCB ₃	PB ₁ /TP ₉ /TIOCB ₃	PB ₁ /TP ₉ /TIOCB ₃	NC
3	PB ₂ /TP ₁₀ /TIOCA ₄	PB ₂ /TP ₁₀ /TIOCA ₄	PB ₂ /TP ₁₀ /TIOCA ₄	NC
4	PB ₃ /TP ₁₁ /TIOCB ₄	PB ₃ /TP ₁₁ /TIOCB ₄	PB ₃ /TP ₁₁ /TIOCB ₄	NC
5	PB ₄ /TP ₁₂ /TOCXA ₄	PB ₄ /TP ₁₂ /TOCXA ₄	PB ₄ /TP ₁₂ /TOCXA ₄	NC
6	PB ₅ /TP ₁₃ /TOCXB ₄	PB ₅ /TP ₁₃ /TOCXB ₄	PB ₅ /TP ₁₃ /TOCXB ₄	NC
7	PB ₆ /TP ₁₄	PB ₆ /TP ₁₄	PB ₆ /TP ₁₄	NC
8	PB ₇ /TP ₁₅ / $\overline{\text{ADTRG}}$	PB ₇ /TP ₁₅ / $\overline{\text{ADTRG}}$	PB ₇ /TP ₁₅ / $\overline{\text{ADTRG}}$	NC
9	P9 ₀ /TxD	P9 ₀ /TxD	P9 ₀ /TxD	V _{CC}
10	P9 ₂ /RxD	P9 ₂ /RxD	P9 ₂ /RxD	NC
11	P9 ₄ /SCK/ $\overline{\text{IRQ}}_4$	P9 ₄ /SCK/ $\overline{\text{IRQ}}_4$	P9 ₄ /SCK/ $\overline{\text{IRQ}}_4$	V _{CC}
12	V _{SS}	V _{SS}	V _{SS}	V _{SS}
13	D ₀	D ₀	P3 ₀	EO ₀
14	D ₁	D ₁	P3 ₁	EO ₁
15	D ₂	D ₂	P3 ₂	EO ₂
16	D ₃	D ₃	P3 ₃	EO ₃
17	D ₄	D ₄	P3 ₄	EO ₄
18	D ₅	D ₅	P3 ₅	EO ₅
19	D ₆	D ₆	P3 ₆	EO ₆
20	D ₇	D ₇	P3 ₇	EO ₇
21	V _{CC}	V _{CC}	V _{CC}	V _{CC}
22	P1 ₀ /A ₀	P1 ₀ /A ₀	P1 ₀	EA ₀
23	P1 ₁ /A ₁	P1 ₁ /A ₁	P1 ₁	EA ₁
24	P1 ₂ /A ₂	P1 ₂ /A ₂	P1 ₂	EA ₂
25	P1 ₃ /A ₃	P1 ₃ /A ₃	P1 ₃	EA ₃
26	P1 ₄ /A ₄	P1 ₄ /A ₄	P1 ₄	EA ₄
27	P1 ₅ /A ₅	P1 ₅ /A ₅	P1 ₅	EA ₅

1. 概要

ピン 番号	端子名			
	モード 1	モード 2	モード 3	PROM モード
28	P1 ₆ /A ₆	P1 ₆ /A ₆	P1 ₆	EA ₆
29	P1 ₇ /A ₇	P1 ₇ /A ₇	P1 ₇	EA ₇
30	V _{SS}	V _{SS}	V _{SS}	V _{SS}
31	P2 ₀ /A ₈	P2 ₀ /A ₈	P2 ₀	EA ₈
32	P2 ₁ /A ₉	P2 ₁ /A ₉	P2 ₁	NC
33	P2 ₂ /A ₁₀	P2 ₂ /A ₁₀	P2 ₂	EA ₁₀
34	P2 ₃ /A ₁₁	P2 ₃ /A ₁₁	P2 ₃	EA ₁₁
35	P2 ₄ /A ₁₂	P2 ₄ /A ₁₂	P2 ₄	EA ₁₂
36	P2 ₅ /A ₁₃	P2 ₅ /A ₁₃	P2 ₅	EA ₁₃
37	P2 ₆ /A ₁₄	P2 ₆ /A ₁₄	P2 ₆	EA ₁₄
38	P2 ₇ /A ₁₅	P2 ₇ /A ₁₅	P2 ₇	\overline{CE}
39	P5 ₀ /A ₁₆	P5 ₀ /A ₁₆	P5 ₀	V _{CC}
40	P5 ₁ /A ₁₇	P5 ₁ /A ₁₇	P5 ₁	V _{CC}
41	P5 ₂ /A ₁₈	P5 ₂ /A ₁₈	P5 ₂	NC
42	P5 ₃ /A ₁₉	P5 ₃ /A ₁₉	P5 ₃	NC
43	P6 ₀ /WAIT $\overline{}$	P6 ₀ /WAIT $\overline{}$	P6 ₀	EA ₁₅
44	MD ₀	MD ₀	MD ₀	V _{SS}
45	MD ₁	MD ₁	MD ₁	V _{SS}
46				NC
47	\overline{STBY}	\overline{STBY}	\overline{STBY}	V _{SS}
48	\overline{RES}	\overline{RES}	\overline{RES}	\overline{OE}
49	NMI	NMI	NMI	EA ₉
50	V _{SS}	V _{SS}	V _{SS}	V _{SS}
51	EXTAL	EXTAL	EXTAL	NC
52	XTAL	XTAL	XTAL	NC
53	V _{CC}	V _{CC}	V _{CC}	V _{CC}
54	\overline{AS}	\overline{AS}	P6 ₃	NC
55	\overline{RD}	\overline{RD}	P6 ₄	NC
56	\overline{WR}	\overline{WR}	P6 ₅	NC
57	\overline{RESO}	\overline{RESO}	\overline{RESO}	V _{PP}
58	AV _{SS}	AV _{SS}	AV _{SS}	V _{SS}
59	P7 ₀ /AN ₀	P7 ₀ /AN ₀	P7 ₀ /AN ₀	NC
60	P7 ₁ /AN ₁	P7 ₁ /AN ₁	P7 ₁ /AN ₁	NC
61	P7 ₂ /AN ₂	P7 ₂ /AN ₂	P7 ₂ /AN ₂	NC

ピン 番号	端子名			
	モード 1	モード 2	モード 3	PROM モード
62	P7 ₃ /AN ₃	P7 ₃ /AN ₃	P7 ₃ /AN ₃	NC
63	P7 ₄ /AN ₄	P7 ₄ /AN ₄	P7 ₄ /AN ₄	NC
64	P7 ₅ /AN ₅	P7 ₅ /AN ₅	P7 ₅ /AN ₅	NC
65	P7 ₆ /AN ₆	P7 ₆ /AN ₆	P7 ₆ /AN ₆	NC
66	P7 ₇ /AN ₇	P7 ₇ /AN ₇	P7 ₇ /AN ₇	NC
67	V _{REF}	V _{REF}	V _{REF}	V _{CC}
68	AV _{CC}	AV _{CC}	AV _{CC}	V _{CC}
69	P8 ₀ /IRQ ₀	P8 ₀ /IRQ ₀	P8 ₀ /IRQ ₀	EA ₁₆
70	P8 ₁ /IRQ ₁	P8 ₁ /IRQ ₁	P8 ₁ /IRQ ₁	EA ₁₈
71	P8 ₂ /IRQ ₂	P8 ₂ /IRQ ₂	P8 ₂ /IRQ ₂	EA ₁₇
72	P8 ₃ /IRQ ₃	P8 ₃ /IRQ ₃	P8 ₃ /IRQ ₃	NC
73	PA ₀ /TP ₀ /TCLKA	PA ₀ /TP ₀ /TCLKA	PA ₀ /TP ₀ /TCLKA	V _{SS}
74	PA ₁ /TP ₁ /TCLKB	PA ₁ /TP ₁ /TCLKB	PA ₁ /TP ₁ /TCLKB	V _{CC}
75	PA ₂ /TP ₂ /TIOCA ₀ /TCLKC	PA ₂ /TP ₂ /TIOCA ₀ /TCLKC	PA ₂ /TP ₂ /TIOCA ₀ /TCLKC	NC
76	PA ₃ /TP ₃ /TIOCB ₀ /TCLKD	PA ₃ /TP ₃ /TIOCB ₀ /TCLKD	PA ₃ /TP ₃ /TIOCB ₀ /TCLKD	NC
77	PA ₄ /TP ₄ /TIOCA ₁	PA ₄ /TP ₄ /TIOCA ₁ /A ₂₃	PA ₄ /TP ₄ /TIOCA ₁	NC
78	PA ₅ /TP ₅ /TIOCB ₁	PA ₅ /TP ₅ /TIOCB ₁ /A ₂₂	PA ₅ /TP ₅ /TIOCB ₁	NC
79	PA ₆ /TP ₆ /TIOCA ₂	PA ₆ /TP ₆ /TIOCA ₂ /A ₂₁	PA ₆ /TP ₆ /TIOCA ₂	NC
80	PA ₇ /TP ₇ /TIOCB ₂	A ₂₀	PA ₇ /TP ₇ /TIOCB ₂	NC

【注】 NC ピンは、何も接続しないでください。

1.4 端子機能

各端子の機能について表 1.3 に示します。

表 1.3 端子機能

分類	記号	ピン記号	入出力	名称および機能
電源	V_{CC}	21、53	入力	<p>電源</p> <p>電源に接続します。</p> <p>V_{CC} 端子は、全端子をシステムの電源に接続してください。</p>
	V_{SS}	12、30、50	入力	<p>グランド</p> <p>電源 (0V) に接続します。</p> <p>V_{SS} 端子は、全端子をシステムの電源 (0V) に接続してください。</p>
クロック	XTAL	52	入力	<p>水晶発振子を接続します。</p> <p>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 15 章 クロック発振器」を参照してください。</p>
	EXTAL	51	入力	<p>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。</p> <p>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 15 章 クロック発振器」を参照してください。</p>
		46	出力	<p>システムクロック</p> <p>外部デバイスにシステムクロックを供給します。</p>

分類	記号	ピン記号	入出力	名称および機能															
動作モード コントロール	MD ₁ 、MD ₀	45、44	入力	<p><u>モード端子</u> 動作モードを設定します。 MD₁、MD₀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>MD₁</th> <th>MD₀</th> <th>動作モード</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>—</td> </tr> <tr> <td>0</td> <td>1</td> <td>モード1</td> </tr> <tr> <td>1</td> <td>0</td> <td>モード2</td> </tr> <tr> <td>1</td> <td>1</td> <td>モード3</td> </tr> </tbody> </table>	MD ₁	MD ₀	動作モード	0	0	—	0	1	モード1	1	0	モード2	1	1	モード3
MD ₁	MD ₀	動作モード																	
0	0	—																	
0	1	モード1																	
1	0	モード2																	
1	1	モード3																	
システム制御	$\overline{\text{RES}}$	48	入力	<p><u>リセット入力</u> 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>															
	$\overline{\text{RESO}}$	57	出力	<p><u>リセット出力</u> 外部デバイスに対し、ウォッチドッグタイマが発生したリセット信号を出力します。</p>															
	$\overline{\text{STBY}}$	47	入力	<p><u>スタンバイ</u> 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>															
割込み	NMI	49	入力	<p><u>ノンマスクابل割込み</u> マスク不可能な割込みを要求します。</p>															
	$\overline{\text{IRQ}}_4 \sim \overline{\text{IRQ}}_0$	11、72～69	入力	<p><u>割込み要求4～0</u> マスク可能な割込みを要求します。</p>															
アドレスバス	A ₂₃ ～A ₂₀ 、 A ₁₉ ～A ₈ 、 A ₇ ～A ₀	77～80、 42～31、 29～22	出力	<p><u>アドレスバス</u> アドレスを出力します。</p>															
データバス	D ₇ ～D ₀	20～13	入出力	<p><u>データバス</u> 双方向データバスです。</p>															
バス制御	$\overline{\text{AS}}$	54	出力	<p><u>アドレスストロープ</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。</p>															

1. 概要

分類	記号	ピン記号	入出力	名称および機能
バス制御	\overline{RD}	55	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{WR}	56	出力	<u>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 $D_7 \sim D_0$ が有効であることを示します。
	\overline{WAIT}	43	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状態の挿入を要求します。
16ビット インテグレー	TCLKD ~ TCLKA	76 ~ 73	入力	<u>クロック入力 D~A</u> 外部クロックを入力します。
テッドタイム ユニット (ITU)	TIOCA ₄ ~ TIOCA ₀	3、1、79、77、 75	入出力	<u>インプットキャプチャ / アウトプットコンペア A4 ~ A0</u> GRA4 ~ 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 ₄ ~ TIOCB ₀	4、2、80、78、 76	入出力	<u>インプットキャプチャ / アウトプットコンペア B4 ~ B0</u> GRB4 ~ 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。
	TOCXA ₄	5	出力	<u>アウトプットコンペア XA4</u> PWM 出力端子です。
	TOCXB ₄	6	出力	<u>アウトプットコンペア XB4</u> PWM 出力端子です。
プログラマブル タイミング パターンコン トローラ (TPC)	TP ₁₅ ~ TP ₀	8 ~ 1、80 ~ 73	出力	<u>TPC 出力 15 ~ 0</u> パルス出力端子です。

分類	記号	ピン記号	入出力	名称および機能
シリアルコミュニケーションインタフェース (SCI)	TxD	9	出力	<u>トランスミットデータ</u> SCI のデータ出力端子です。
	RxD	10	入力	<u>レシーブデータ</u> SCI のデータ入力端子です。
	SCK	11	入出力	<u>シリアルクロック</u> SCI のクロック入出力端子です。
A/D 変換器	AN ₇ ~ AN ₀	66 ~ 59	入力	<u>アナログ 7 ~ 0</u> アナログ入力端子です。
	$\overline{\text{ADTRG}}$	8	入力	<u>A/D 変換外部トリガ入力</u> A/D 変換開始のための外部トリガ入力端子です。
	AV _{CC}	68	入力	A/D 変換器の電源端子です。 A/D 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。
	AV _{SS}	58	入力	A/D 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 _{REF}	67	入力	A/D 変換器の基準電圧入力端子です。 A/D 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。
I/O ポート	P1 ₇ ~ P1 ₀	29 ~ 22	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 ₇ ~ P2 ₀	38 ~ 31	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 ₇ ~ P3 ₀	20 ~ 13	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。

1. 概要

分類	記号	ピン記号	入出力	名称および機能
I/O ポート	P5 ₃ ~ P5 ₀	42 ~ 39	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 ₅ ~ P6 ₃ 、 P6 ₀	56 ~ 54、43	入出力	<u>ポート 6</u> 4 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 ₇ ~ P7 ₀	66 ~ 59	入力	<u>ポート 7</u> 8 ビットの入力端子です。
	P8 ₃ ~ P8 ₀	72 ~ 69	入出力	<u>ポート 8</u> 4 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 ₄ 、P9 ₂ 、P9 ₀	11 ~ 9	入出力	<u>ポート 9</u> 3 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 ₇ ~ PA ₀	80 ~ 73	入出力	<u>ポート A</u> 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 ₇ ~ PB ₀	8 ~ 1	入出力	<u>ポート B</u> 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。

2. CPU

第2章 目次

2.1	概要	19
2.1.1	特長	19
2.1.2	H8/300 CPU との相違点	20
2.2	CPU 動作モード	21
2.3	アドレス空間	22
2.4	レジスタ構成	23
2.4.1	概要	23
2.4.2	汎用レジスタ	24
2.4.3	コントロールレジスタ	25
2.4.4	CPU 内部レジスタの初期値	26
2.5	データ構成	27
2.5.1	汎用レジスタのデータ構成	28
2.5.2	メモリ上でのデータ構成	29
2.6	命令セット	30
2.6.1	命令セットの概要	30
2.6.2	命令とアドレッシングモードの組み合わせ	30
2.6.3	命令の機能別一覧	32
2.6.4	命令の基本フォーマット	40
2.6.5	ビット操作命令使用上の注意	41
2.7	アドレッシングモードと実効アドレスの計算方法	42
2.7.1	アドレッシングモード	42
2.7.2	実効アドレスの計算方法	45
2.8	処理状態	49
2.8.1	概要	49
2.8.2	プログラム実行状態	49
2.8.3	例外処理状態	49
2.8.4	例外処理の動作	51

2.8.5	リセット状態	52
2.8.6	低消費電力状態	52
2.9	基本動作タイミング	54
2.9.1	概要	54
2.9.2	内蔵メモリアクセスタイミング	54
2.9.3	内蔵周辺モジュールアクセスタイミング	55
2.9.4	外部アドレス空間アクセスタイミング	56

2.1 概要

H8/300H CPU は、H8/300 CPU 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、1M 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。

H8/300 CPU 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：18MHz
- ・ 8/16/32 ビットレジスタ間加減算 111ns
- ・ 8×8 ビットレジスタ間乗算 778ns
- ・ 16÷8 ビットレジスタ間除算 778ns
- ・ 16×16 ビットレジスタ間乗算 1.222μs
- ・ 32÷16 ビットレジスタ間除算 1.222μs

2 種類の CPU 動作モード

- ・ ノーマルモード (本 LSI では使用できません。)
- ・ 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 H8/300 CPU との相違点

H8/300H CPU は、H8/300 CPU 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、H8/300 CPU と同一の 64k バイトのアドレス空間を使用可能（本 LSI では使用できません）

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、アドバンスモードの場合最大 16M バイトとなります。

本 LSI では、アドバンスモードのみを使用できます。(以降、特に説明がない場合は、アドバンスモードについて説明します。)

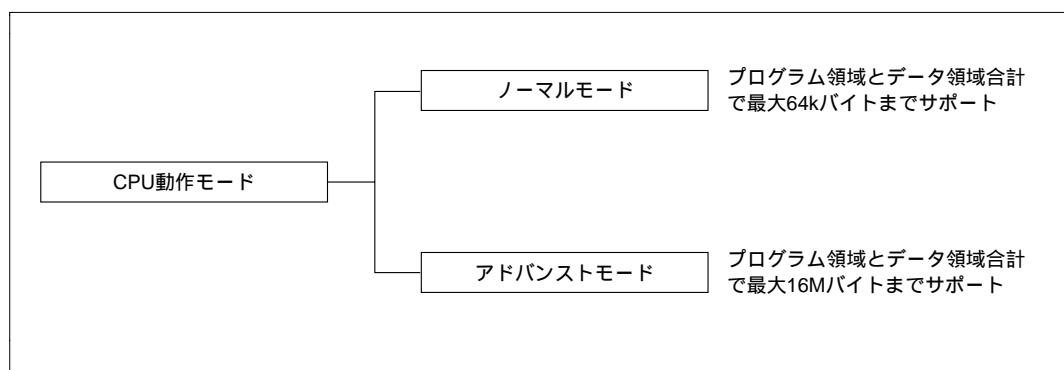


図 2.1 CPU 動作モード

2.3 アドレス空間

H8/300H CPU のアドレス空間は最大 16M バイトです。本 LSI では MCU 動作モードにより、アドレス空間は、1M バイトモードと 16M バイトモードを選択できます。

本 LSI のメモリマップの概要を図 2.2 に示します。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。

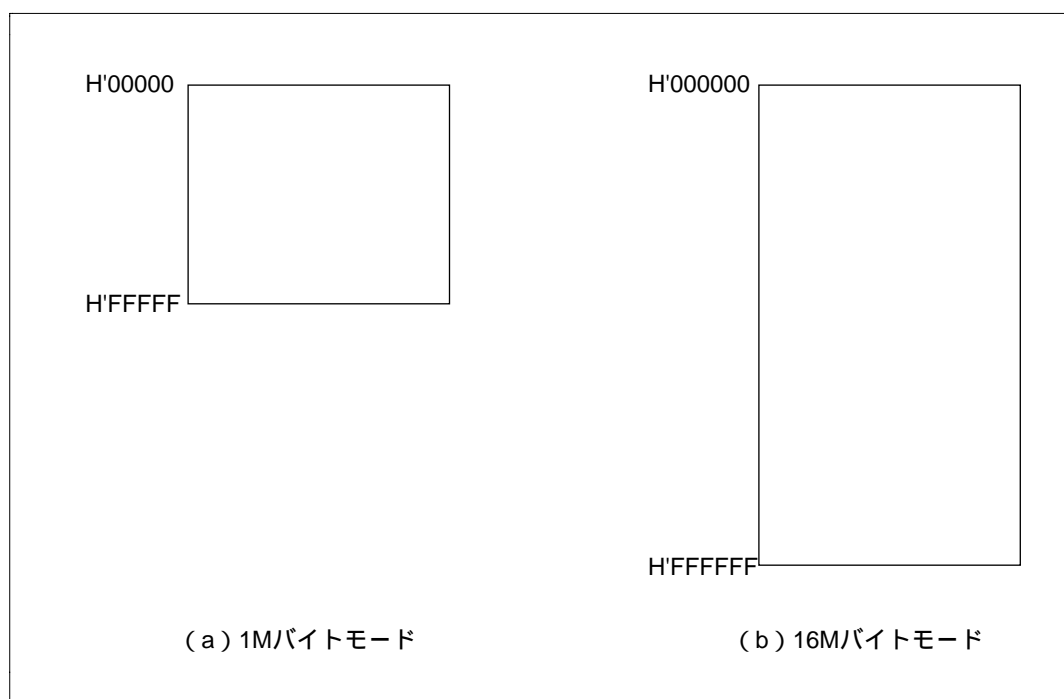


図 2.2 メモリマップ

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。

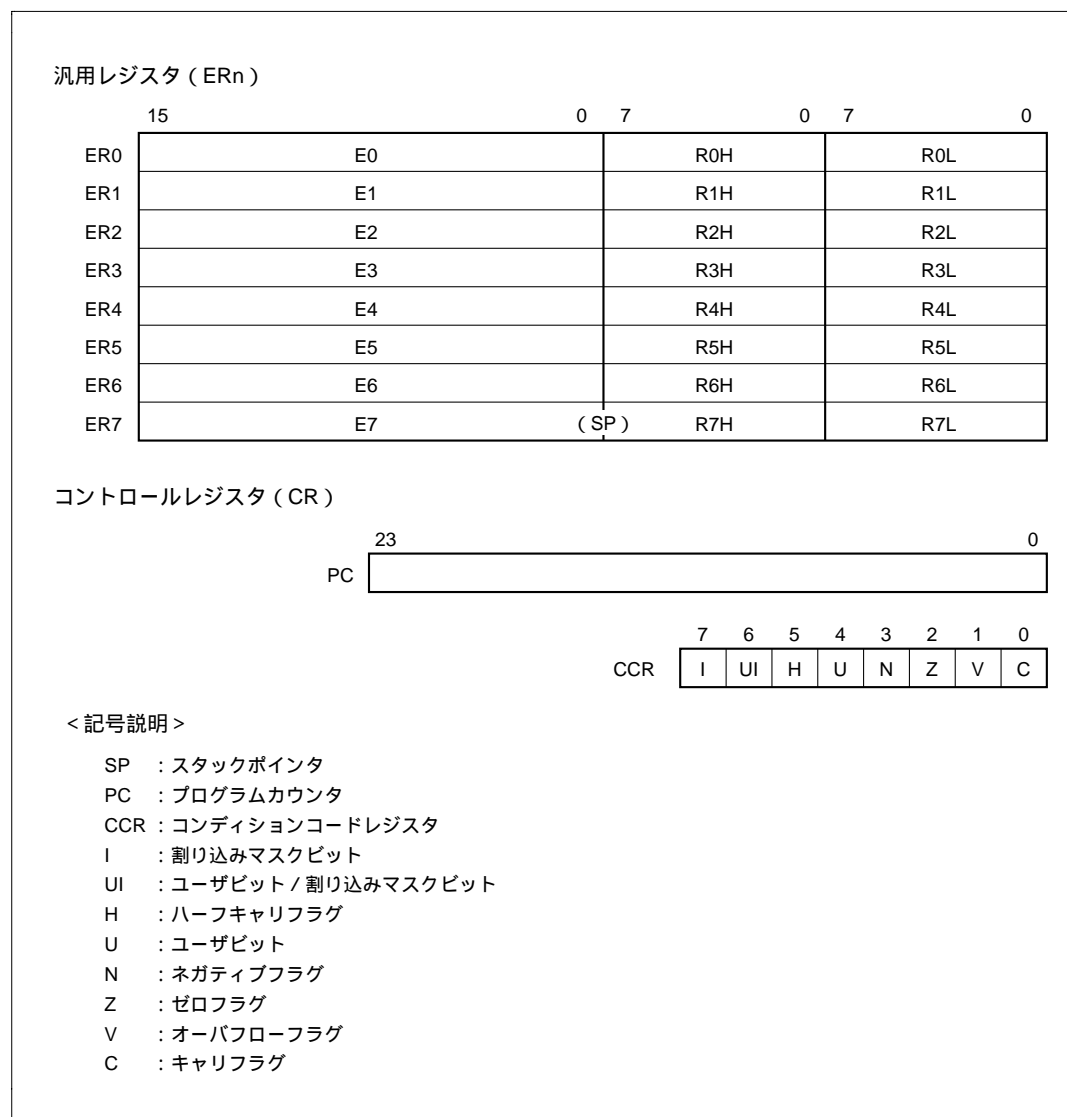


図 2.3 CPU 内部レジスタ構成

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。各レジスタは独立に使用方法を選択することができます。

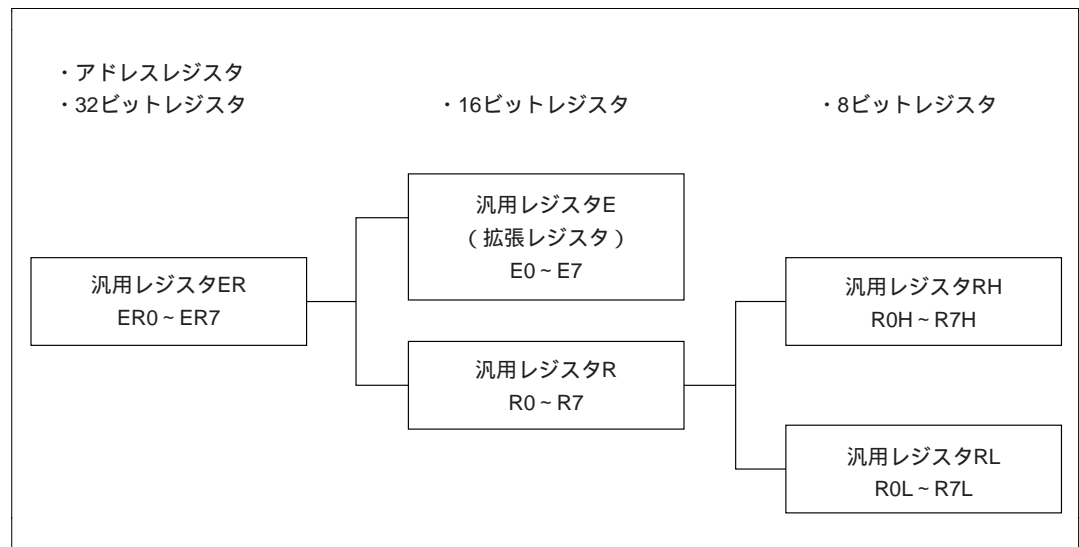


図2.4 汎用レジスタの使用方法

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。

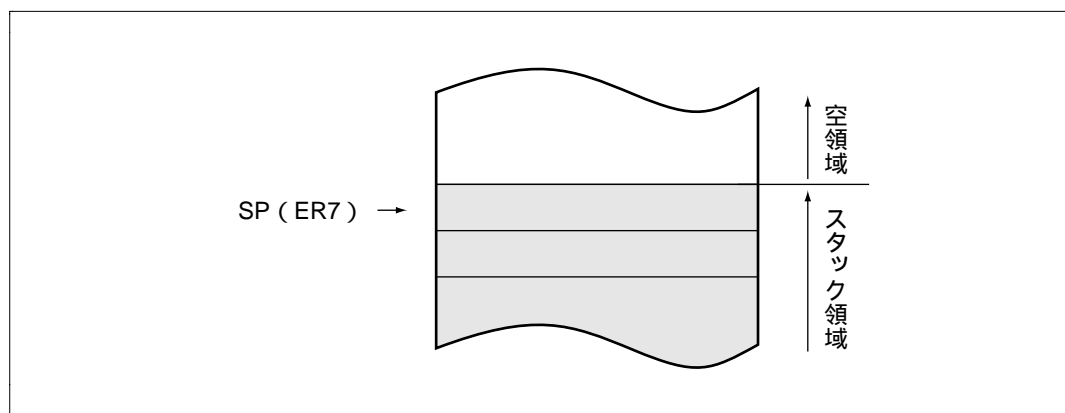


図 2.5 スタックの状態

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) を初期化してください。

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 に示します。

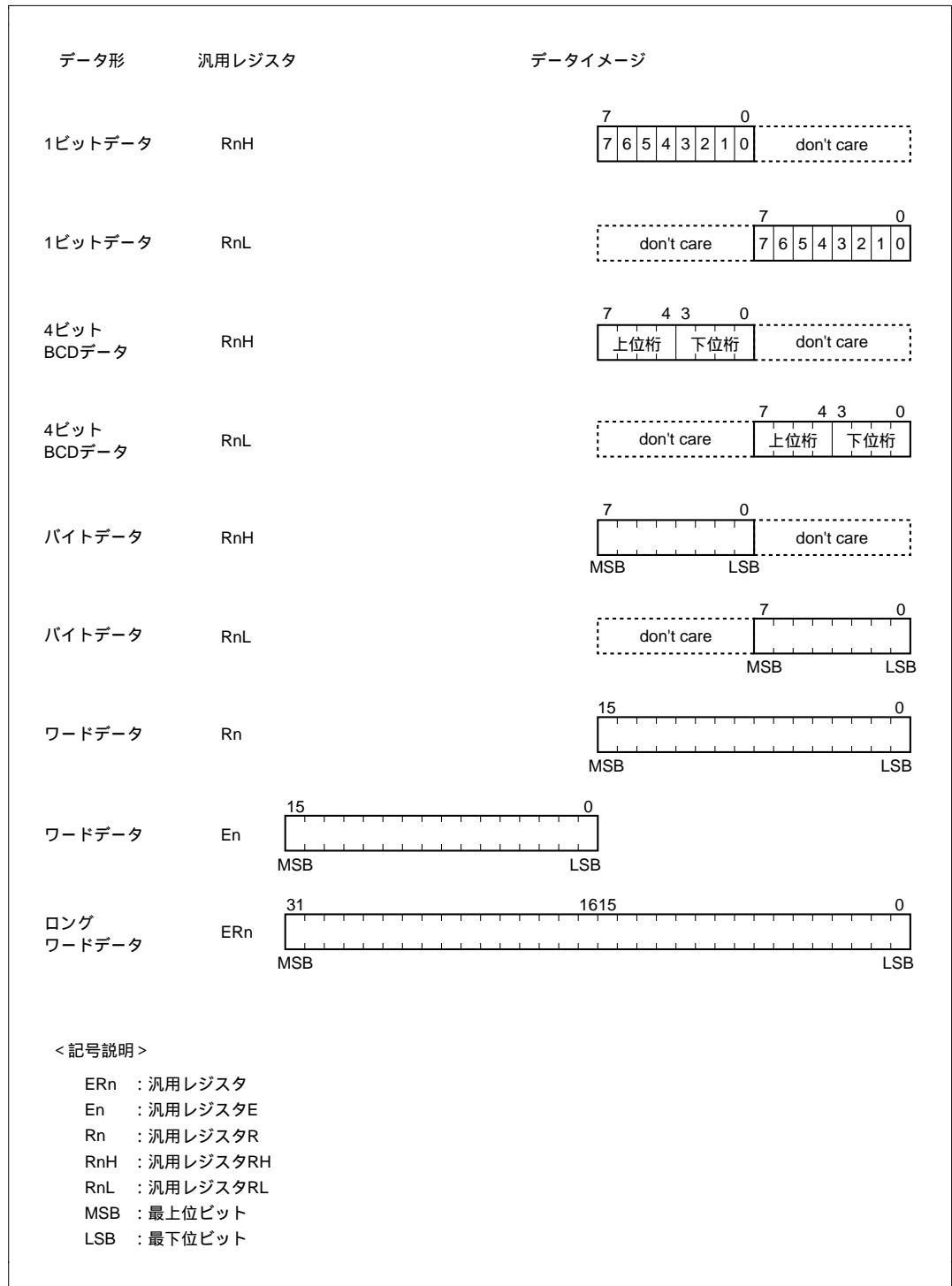


図 2.6 汎用レジスタのデータ構成

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2.7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。

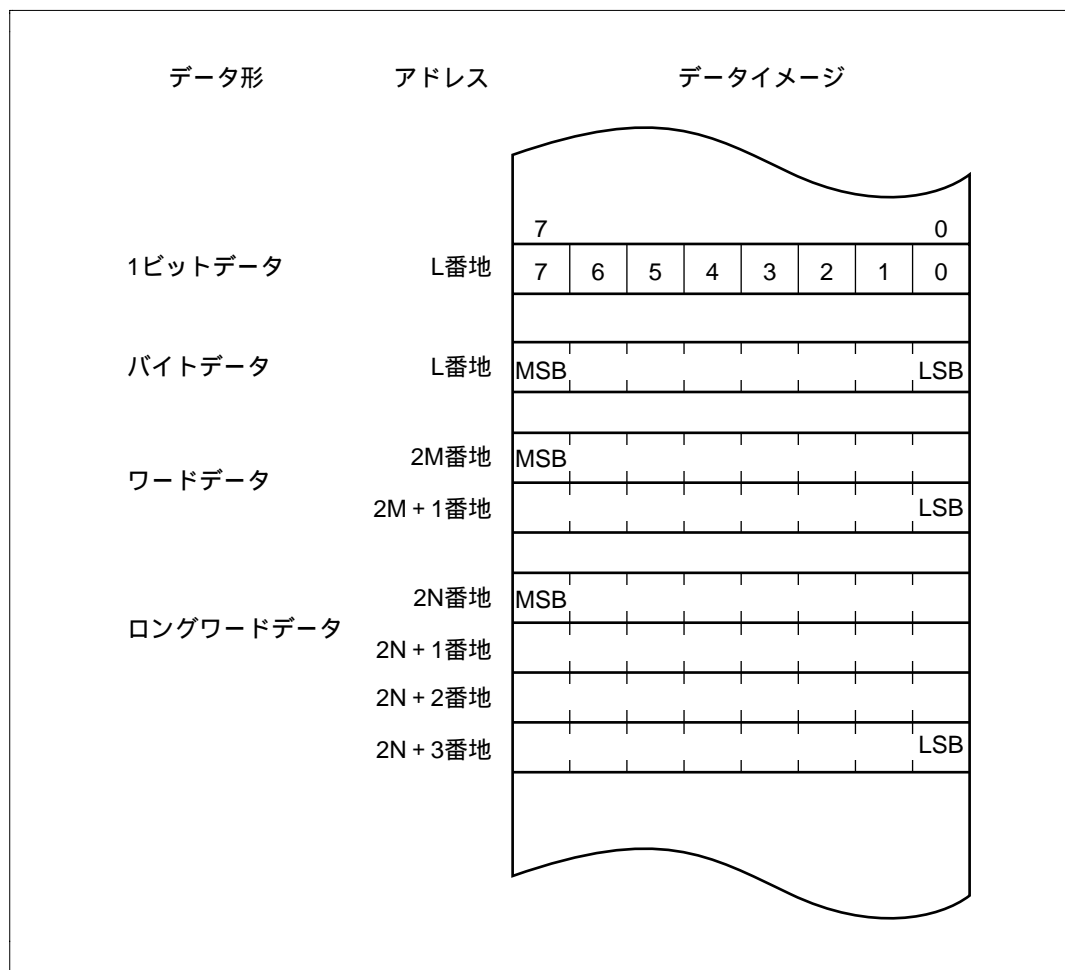


図2.7 メモリ上でのデータ構成

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* ¹ 、POP* ¹ 、MOVTP* ² 、MOVFP* ²	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* ³ 、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)	@@aa:8				
		BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL	BWL
データ転送命令	MOV																
	POP、PUSH																
	MOVFP、MOVTFE																
	MOVTFE																
	ADD、CMP	BWL	BWL														
	SUB	WL	BWL														
	ADDX、SUBX	B	B														
	ADDS、SUBS		L														
	INC、DEC		BWL														
	DAA、DAS		B														
算術演算命令	MULXU、MULXS、DIVXU、DIVXS		BW														
	NEG		BWL														
	EXTU、EXTS		WL														
	AND、OR、XOR	BWL	BWL														
	NOT		BWL														
	シフト命令		BWL														
	ビット操作命令		B	B													
	Bcc、BSR																
	JMP、JSR																
	シフト制御命令	RTS															
TRAPA																	
RTE																	
SLEEP																	
LDC		B	B	W	W	W	W	W	W	W	W	W	W	W	W	W	W
シフト制御命令	STC		B	W	W	W	W	W	W	W	W	W	W	W	W	W	W
	ANDC、ORC、XORC	B															
	NOP																
	ブロック転送命令																BW

《記号説明》

B：バイト、W：ワード、L：ロングワード

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) Rd, Rs (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。
MOVFP	B	(EAs) Rd 本 LSI では使用できません。
MOVTP	B	Rs (EAs) 本 LSI では使用できません。
POP	W/L	@SP+ 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 @-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	$Rd \pm Rs$ Rd , $Rd \pm \#IMM$ 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	$Rd \pm Rs \pm C$ Rd , $Rd \pm \#IMM \pm C$ 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	$Rd \pm 1$ Rd , $Rd \pm 2$ 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	$Rd \pm 1$ Rd , $Rd \pm 2$ Rd , $Rd \pm 4$ 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 進補正) 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	$Rd \times Rs$ 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 \times 8 ビット 16 ビット、 16 ビット \times 16 ビット 32 ビットの乗算が可能です。
MULXS	B/W	$Rd \times Rs$ 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 \times 8 ビット 16 ビット、 16 ビット \times 16 ビット 32 ビットの乗算が可能です。
DIVXU	B/W	$Rd \div Rs$ 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 \div 8 ビット 商 8 ビット 余り 8 ビット、 32 ビット \div 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	$Rd \div Rs$ 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 \div 8 ビット 商 8 ビット 余り 8 ビット、 32 ビット \div 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 Rd、Rd #IMM 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 Rd、Rd #IMM 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⊕Rs Rd、Rd⊕#IMM 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	~ Rd 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 (シフト処理) 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 (シフト処理) 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 (ローテート処理) 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 (ローテート処理) 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 (<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 (<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	~ (<ビット番号> of <EAd>) (<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	~ (<ビット番号> of <EAd>) 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 (<ビット番号> of <EAd>) 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。

命令	サイズ*	機能
BIAND	B	$C \ [\sim (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle)] \quad C$ 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。
BOR	B	$C \ (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle) \quad C$ 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	$C \ [\sim (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle)] \quad C$ 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	$C \oplus (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle) \quad C$ 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	$C \oplus [\sim (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle)] \quad C$ 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	$(\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle) \quad C$ 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	$\sim (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle) \quad C$ 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。
BST	B	$C \quad (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle)$ 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	$C \quad \sim (\langle \text{ビット番号} \rangle \text{ of } \langle \text{EAd} \rangle)$ 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>$N \oplus V = 0$</td> </tr> <tr> <td>BLT</td> <td>Less Than</td> <td>$N \oplus V = 1$</td> </tr> <tr> <td>BGT</td> <td>Greater Than</td> <td>$Z (N \oplus V) = 0$</td> </tr> <tr> <td>BLE</td> <td>Less or Equal</td> <td>$Z (N \oplus V) = 1$</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	-	低消費電力状態に遷移します。
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	-	if R4L = 0 then Repeat @ER5+ @ER6+, R4L-1 R4L Until R4L=0 else next;
EEPMOV.W	-	if R4 = 0 then Repeat @ER5+ @ER6+, R4 - 1 R4 Until R4=0 else next;

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。

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イ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イメントは上位 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 2.8 に命令フォーマットの例を示します。

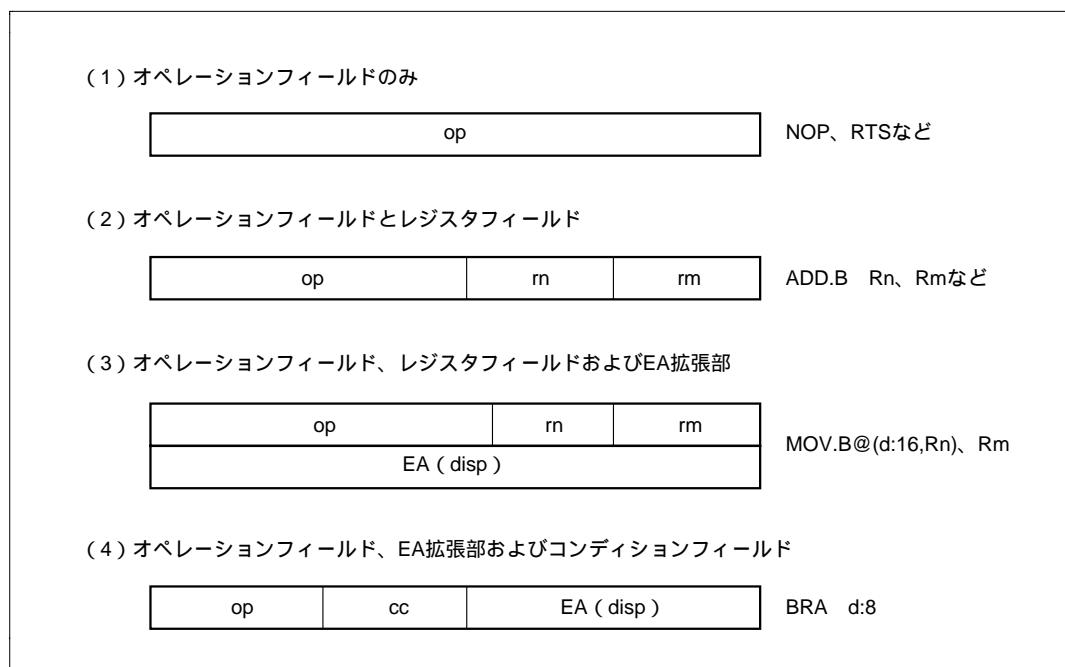


図 2.8 命令フォーマット

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。

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'FFFFF (1048320 ~ 1048575)	H'FFFF00 ~ H'FFFFFFF (16776960 ~ 16777215)
16ビット (@aa:16)	H'00000 ~ H'07FFF、H'F8000 ~ H'FFFFF (0 ~ 32767、1015808 ~ 1048575)	H'000000 ~ H'007FFF、 H'FF8000 ~ H'FFFFFFF (0 ~ 32767、16744448 ~ 16777215)
24ビット (@aa:24)	H'00000 ~ H'FFFFF (0 ~ 1048575)	H'000000 ~ H'FFFFFFF (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ビットディスプレイメントを加算して、24ビットの分岐アドレスを生成します。加算に際して、ディスプレイメントは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 2.9 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 割り込みコントローラ」を参照してください。

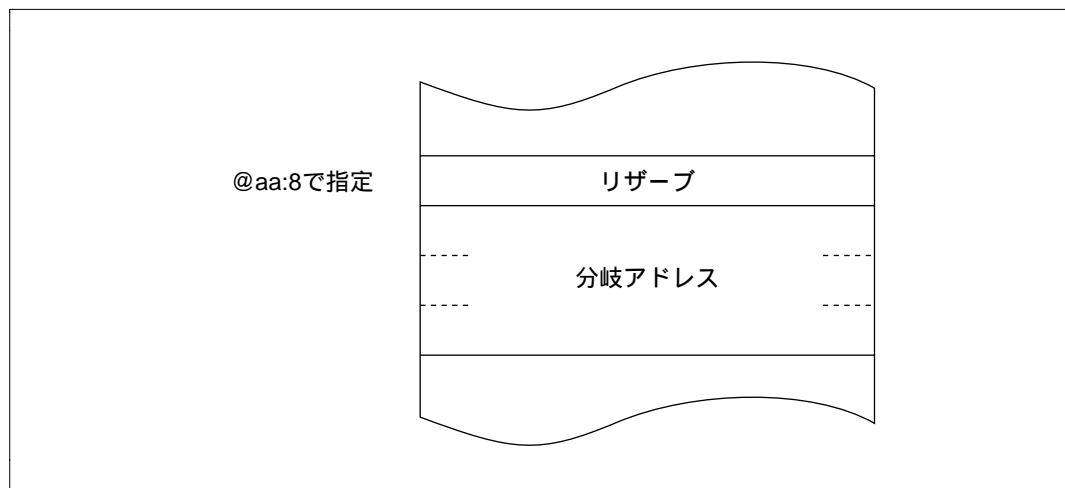


図 2.9 メモリ間接による分岐アドレスの指定

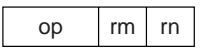
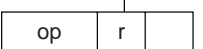
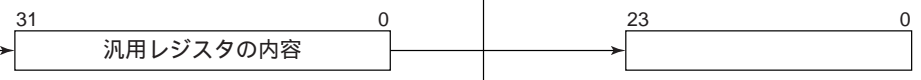
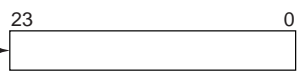
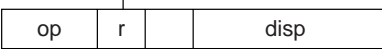
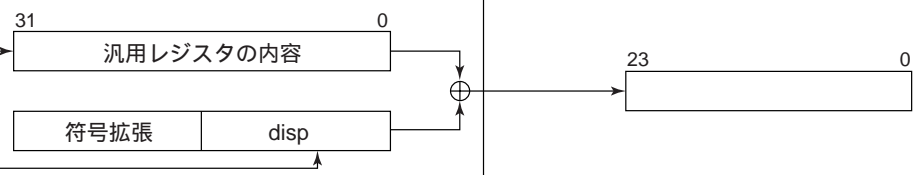
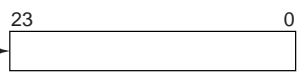
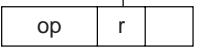
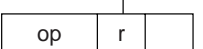
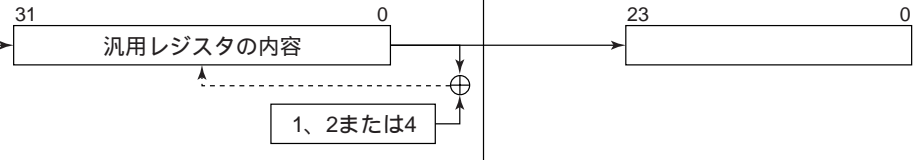
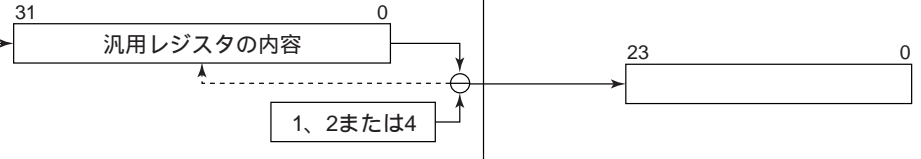
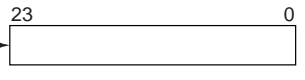
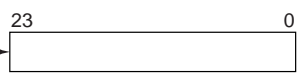
ワードサイズ、またはロングワードサイズでメモリを指定する場合、および分岐アドレスを指定する場合に奇数アドレスを指定すると、最下位ビットは"0"とみなされ、1番地前から始まるデータまたは命令コードをアクセスします（「2.5.2 メモリ上でのデータ構成」を参照してください）。

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 実行アドレスの計算方法

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)
(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)	ディスペ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 	 	 
		オペランドサイズがバイトのとき1、ワードのとき2、ロングワードのとき4が加減算されます。	

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)	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		オペランドはイミディエイトデータです。
(7)	プログラムカウンタ相対 @(d:8,PC) / @(d:16,PC)		

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 アドバンスドモード <div style="display: flex; border: 1px solid black; padding: 2px;"> <div style="border: 1px solid black; padding: 2px; width: 40px; text-align: center;">op</div> <div style="border: 1px solid black; padding: 2px; width: 40px; text-align: center;">abs</div> </div>	<p>The diagram shows the calculation of the effective address (EA). It starts with the 'abs' field from the command format, which points to a register containing the value 'H'0000'. The 'op' field from the command format points to a memory location. The value 'H'0000' is added to the memory address to determine the final effective address.</p>	<p>The final effective address (EA) is shown as a 24-bit register (bits 23 to 0).</p>

< 記号説明 >

- r、rm、rn : 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、リセット状態の 4 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 2.10 に、各状態間の遷移を図 2.12 に示します。

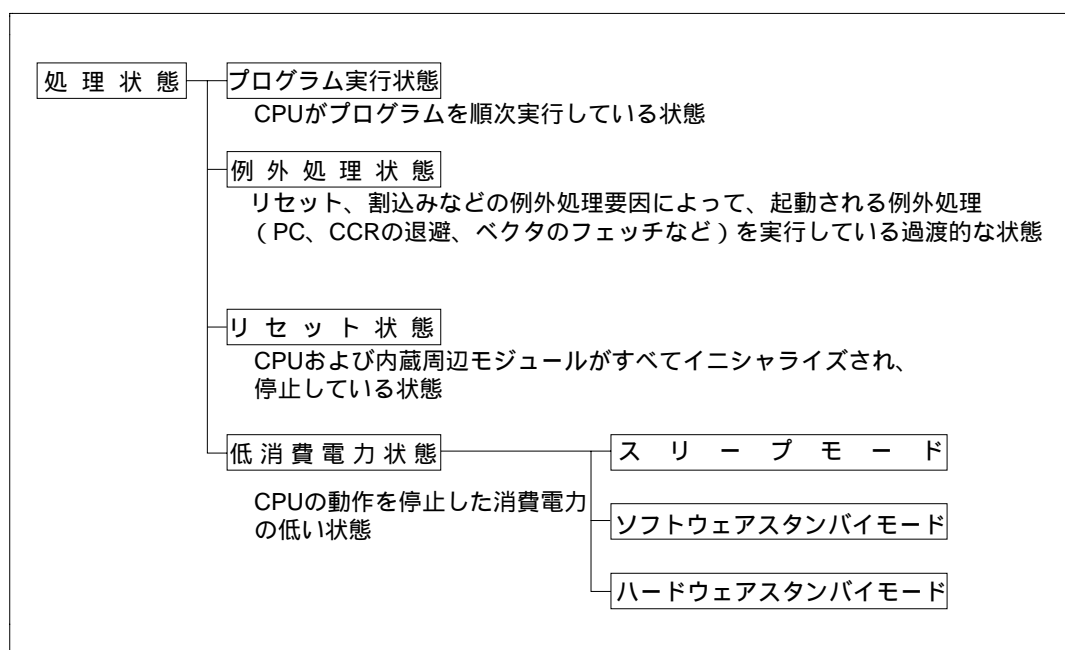


図 2.10 処理状態の分類

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、割り込み、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 ↑ 低	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。
	割り込み	命令の実行終了時 または例外処理終了時*	割り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。
	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、割り込み要因の検出を行いません。

例外処理要因は、図 2.11 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 割り込みコントローラ」を参照してください。

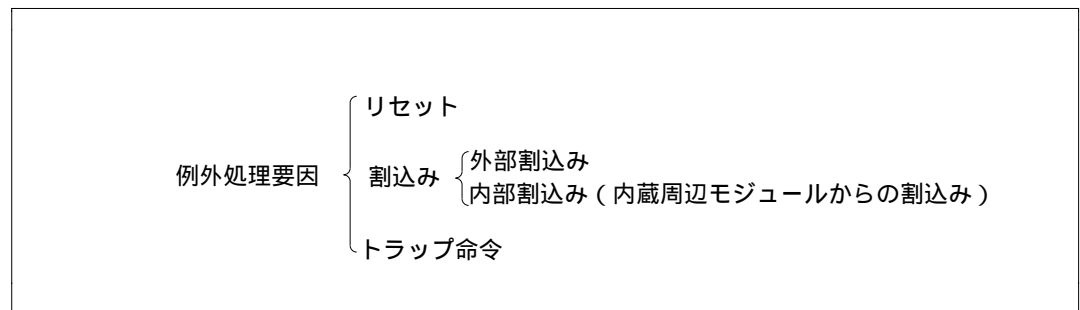


図 2.11 例外処理要因の分類

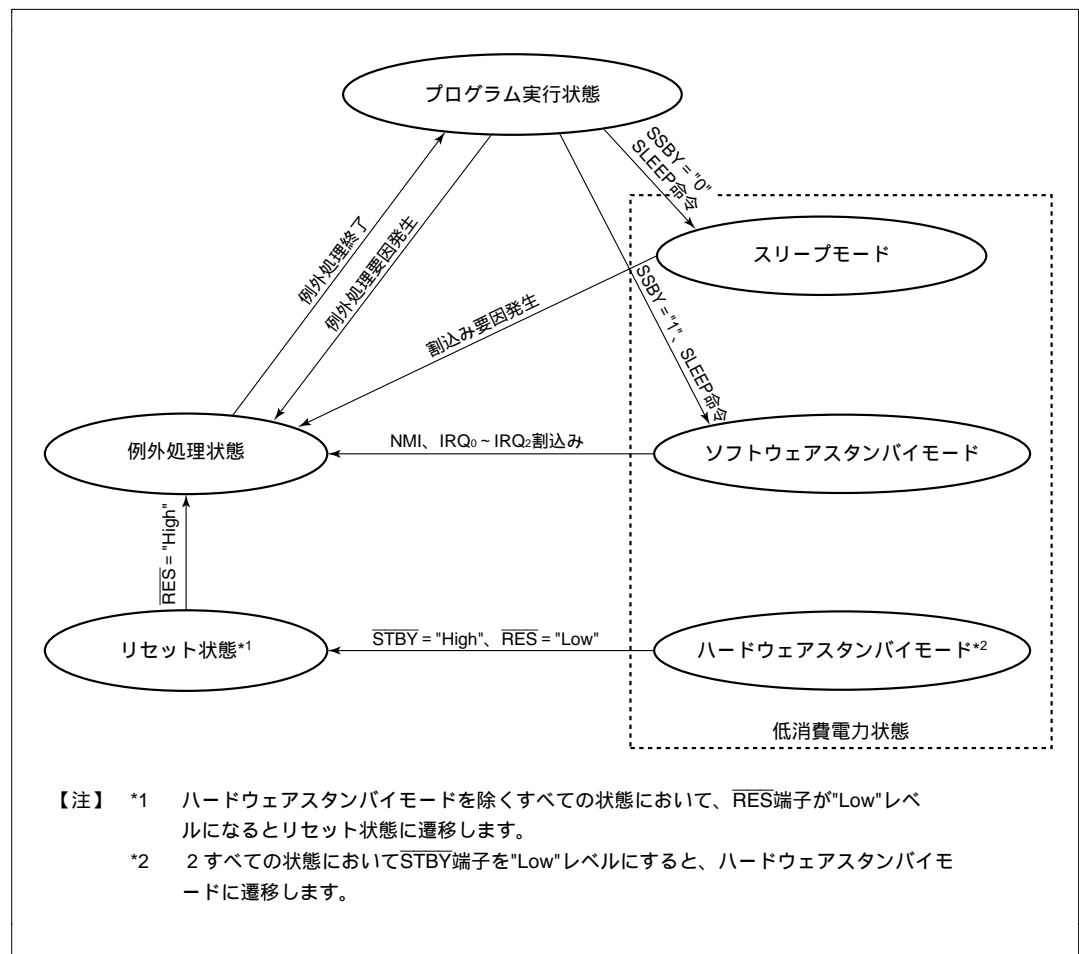


図 2.12 状態遷移図

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 2.13 に示します。

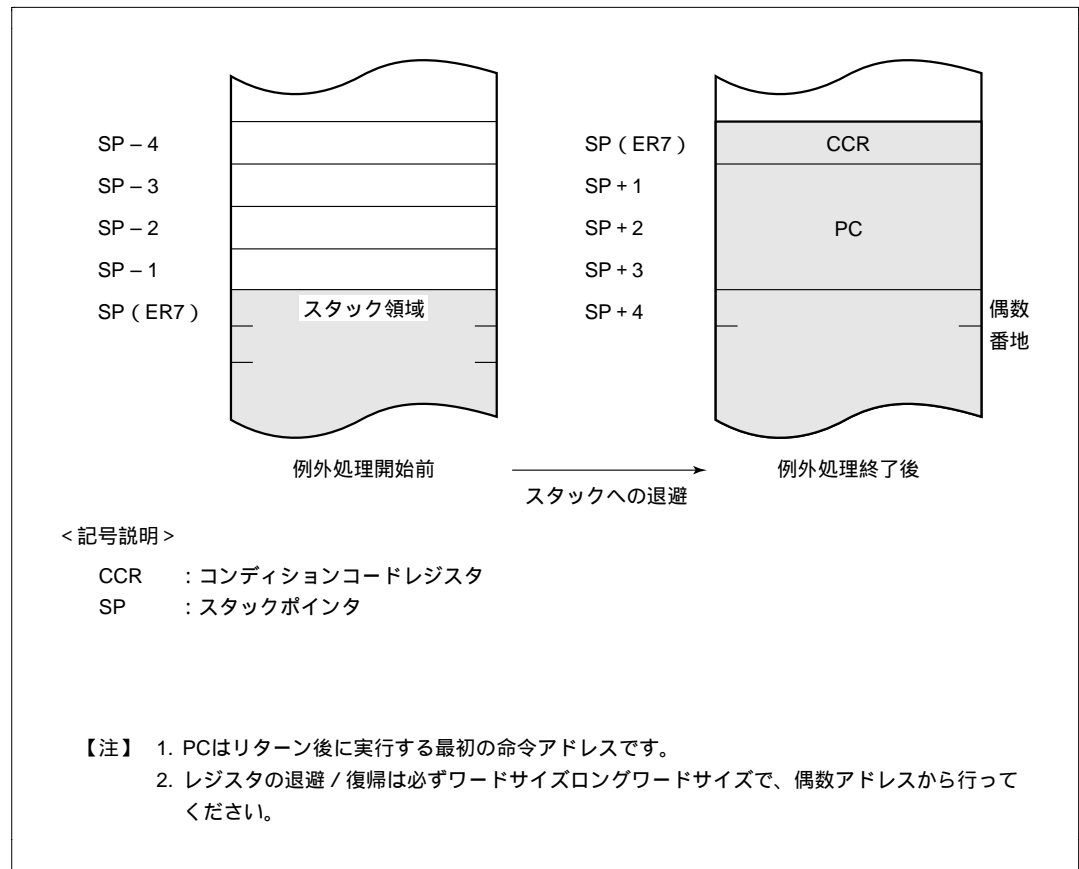


図 2.13 例外処理終了後のスタック状態

2.8.5 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り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ーフローによって、リセット状態とすることもできます。詳細は「第10章 ウォッチドッグタイマ」を参照してください。

2.8.6 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 15 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 2.14 に、端子状態を図 2.15 に示します。

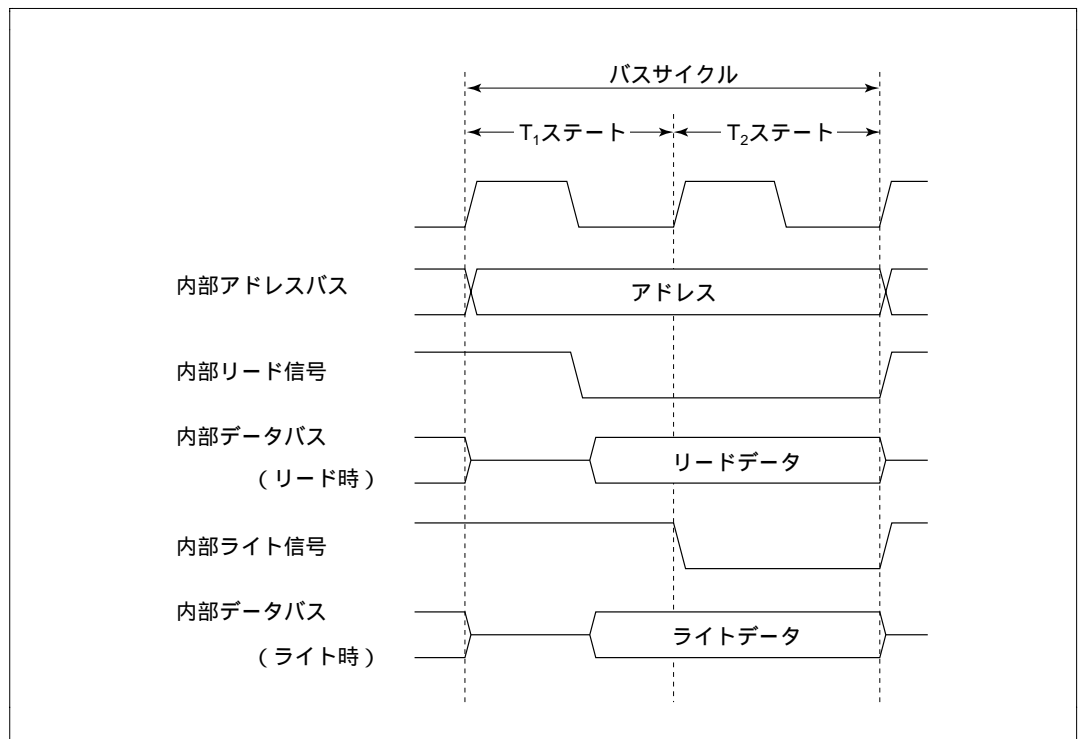


図 2.14 内蔵メモリアクセスサイクル

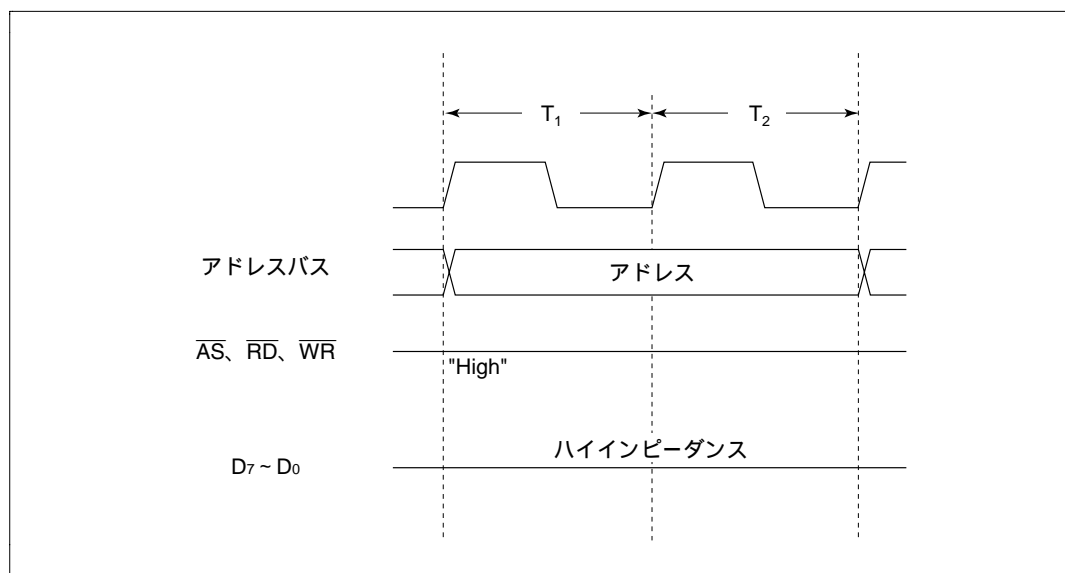


図 2.15 内蔵メモリアクセス時の端子状態

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2.16に、端子状態を図2.17に示します。

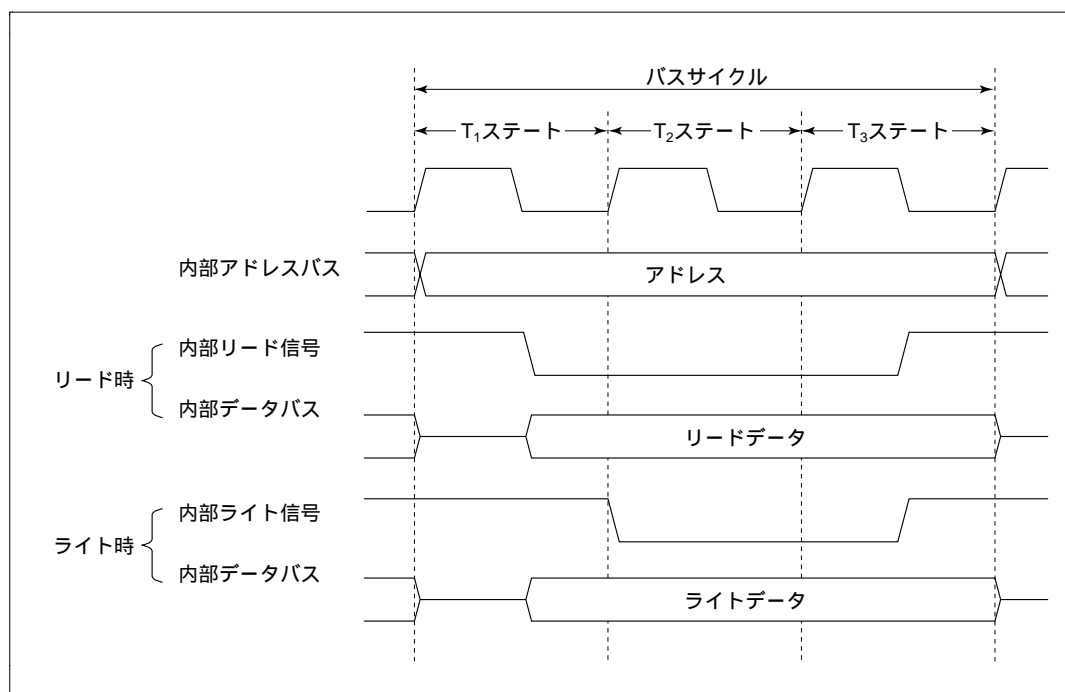


図 2.16 内蔵周辺モジュールアクセスサイクル

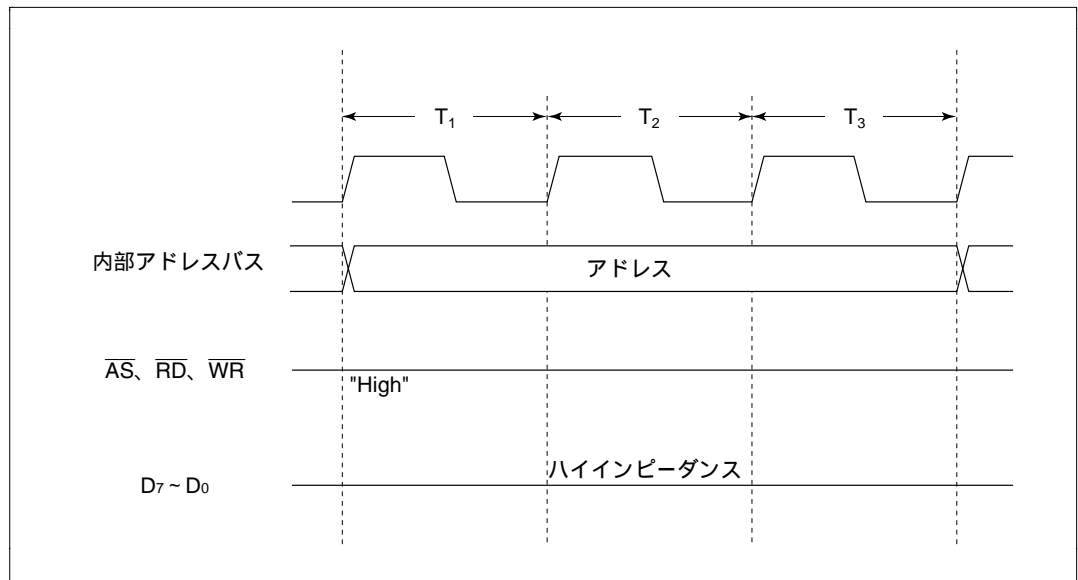


図 2.17 内蔵周辺モジュールアクセス時の端子状態

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	59
	3.1.1 動作モードの選択の種類	59
	3.1.2 レジスタ構成	59
3.2	モードコントロールレジスタ (MDCR)	60
3.3	システムコントロールレジスタ (SYSCR)	61
3.4	各動作モードの説明	64
	3.4.1 モード 1	64
	3.4.2 モード 2	64
	3.4.3 モード 3	64
3.5	各動作モードにおける端子機能	65
3.6	各動作モードのメモリマップ	66

3.1 概要

3.1.1 動作モードの選択の種類

本 LSI には、3 種類の動作モード（モード 1~3）があります。これらのモードは、モード端子（MD₁、MD₀）を表 3.1 のように設定することによって拡張モードとシングルチップモード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 ₁	MD ₀	アドレス空間	バス幅	内蔵 RAM
	0	0			
モード 1	0	1	1M バイト	8 ビット	有効* ¹
モード 2	1	0	16M バイト	8 ビット	有効* ¹
モード 3	1	1	1M バイト		有効* ²

【注】 *¹ モード 1、2 では、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。

*² モード 3 では、SYSCR の RAME ビットを"0"にクリアし内蔵 RAM 空間をアクセスすると、常に H'FF がリードされ、ライトは無効です。詳細は「12.3 動作説明」を参照してください。

アドレス空間は、1M バイト / 16M バイトいずれかを選択することができます。

モード 1、3 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 2 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 1~3 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 1~3 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（MD₁、MD₀）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'FFF1	モードコントロールレジスタ	MDCR	R	不定
H'FFF2	システムコントロールレジスタ	SYSCR	R/W	H'0B

【注】 * アドレスの下位 16 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
	—	—	—	—	—	—	MDS1	MDS0
初期値:	1	1	0	0	0	1	—*	—*
R/W :	—	—	—	—	—	—	R	R

└─ リザーブビット
└─ モードセレクト1、0
現在の動作モードを示すビットです。

【注】 * MD₁、MD₀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：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：モードセレクト1、0 (MDS1、0)

これらのビットは、モード端子 (MD₁、MD₀) 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。MDS1、MDS0 ビットはMD₁、MD₀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 (MD₁、MD₀) 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	—	RAME
初期値:	0	0	0	0	1	0	1	1
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。

リザーブビット

NMIエッジ
 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 15 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。

動作周波数に応じて待機時間が8ms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、「15.4.3 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 = 8,192 ステート (初期状態)
0	0	1	待機時間 = 16,384 ステート
0	1	0	待機時間 = 32,768 ステート
0	1	1	待機時間 = 65,536 ステート
1	0		待機時間 = 131,072 ステート
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（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：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：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、 $\overline{\text{RES}}$ 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	
RAME	説明
0	内蔵RAM無効
1	内蔵RAM有効 (初期値)

3.4 各動作モードの説明

3.4.1 モード 1

アドレス空間で 1M バイトで、内蔵 ROM を使用することができます。

ポート 3 の機能がデータ入出力端子 $D_7 \sim D_0$ となり、ポート 1、2、5 の機能がアドレス端子 $A_{19} \sim A_0$ となります。アドレスバス幅は、ポート 1、2、5 の DDR の内容を設定することにより、自由に指定できます。

3.4.2 モード 2

アドレス空間 16M バイトで、内蔵 ROM を使用することができます。ポート 3 の機能がデータ入出力端子 $D_7 \sim D_0$ となり、ポート 1、2、5 およびポート A ($PA_7 \sim PA_4$) の機能がアドレス端子 $A_{23} \sim A_0$ となります。アドレスバス幅、ポート 1、2、5 の DDR とポート A の ADRCR のそれぞれの内容を設定することにより、指定できます。

本モードでは、アドレス端子 A_{20} は常にアドレス出力となります。

3.4.3 モード 3

内蔵 ROM、RAM で動作するモードです。外部アドレス空間をアクセスすることはできません。アドレス空間は 1M バイトとなります。アドバンスモードで動作します。

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、2、3、5、6、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機能

ポート	モード1	モード2	モード3
ポート1	P1 ₇ ~ P1 ₀ * ¹	P1 ₇ ~ P1 ₀ * ¹	P1 ₇ ~ P1 ₀
ポート2	P2 ₇ ~ P2 ₀ * ¹	P2 ₇ ~ P2 ₀ * ¹	P2 ₇ ~ P2 ₀
ポート3	D ₇ ~ D ₀	D ₇ ~ D ₀	P3 ₇ ~ P3 ₀
ポート5	P5 ₃ ~ P5 ₀ * ¹	P5 ₃ ~ P5 ₀ * ¹	P5 ₃ ~ P5 ₀
ポート6	\overline{WR} 、 \overline{RD} 、 \overline{AS} 、P6 ₀ /WAIT* ²	\overline{WR} 、 \overline{RD} 、 \overline{AS} 、P6 ₀ /WAIT* ²	P6 ₅ ~ P6 ₃ 、P6 ₀
ポートA	PA ₇ ~ PA ₄	PA ₆ ~ PA ₄ 、A ₂₀ * ³	PA ₇ ~ PA ₄

【注】 *1 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 (DDR) を"1"に設定することにより、アドレスバスとなります。

*2 ウェイトステートコントローラインプットレジスタ (WCER)、ウェイトコントロールレジスタ (WCR)、およびポートデータディレクションレジスタの設定により機能が異なります。

*3 初期状態を示しています。A₂₀は常にアドレス出力です。PA₆ ~ PA₄は、ADRCRのビット7 ~ 5に"0"をライトすることによりA₂₃ ~ A₂₁出力になります。

3.6 各動作モードのメモリマップ

H8/3035 のメモリマップを図 3.1 に、H8/3034 のメモリマップを図 3.2 に H8/3033 のメモリマップを図 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。

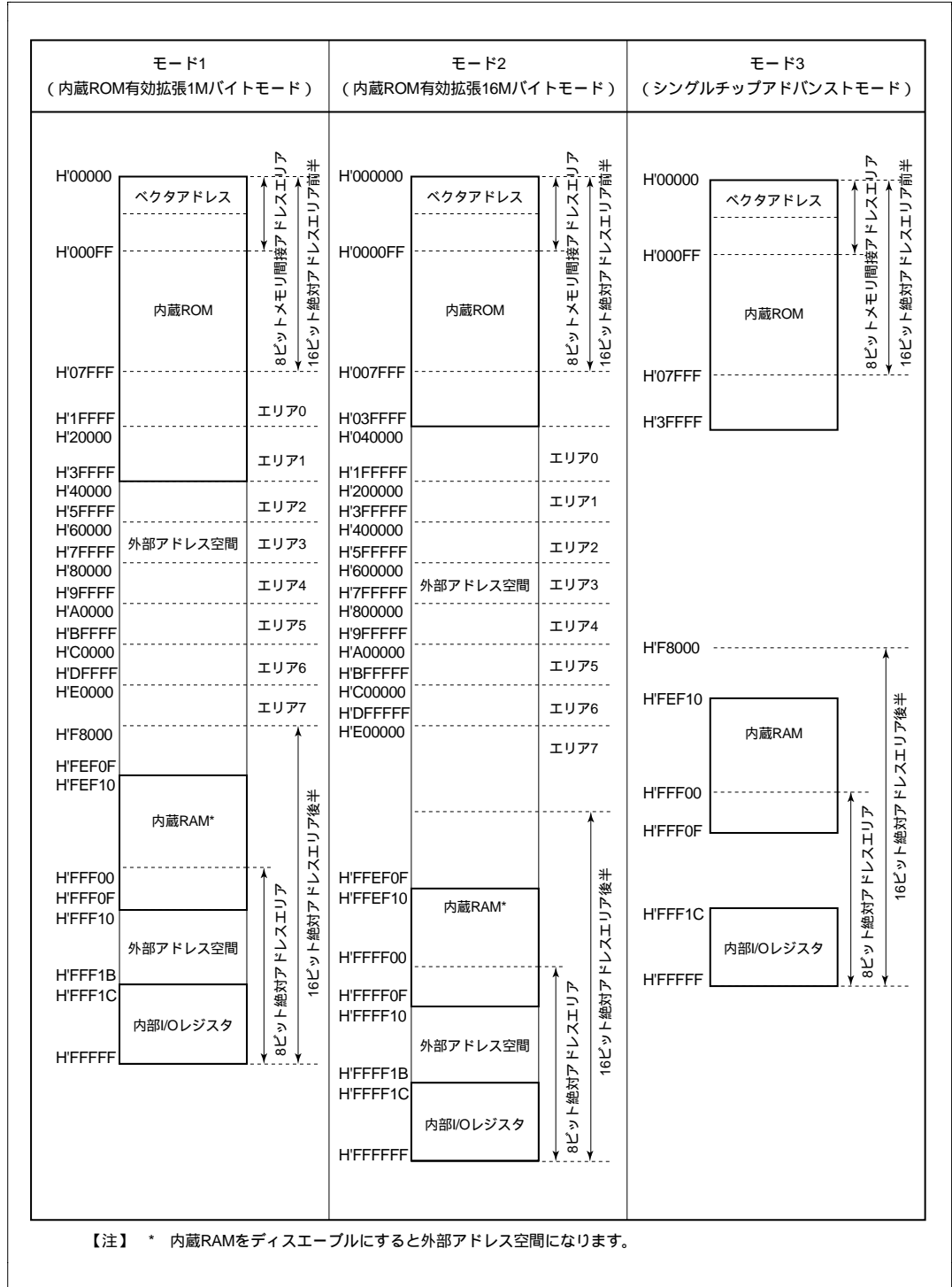


図 3.1 H8/3035 のメモリマップ

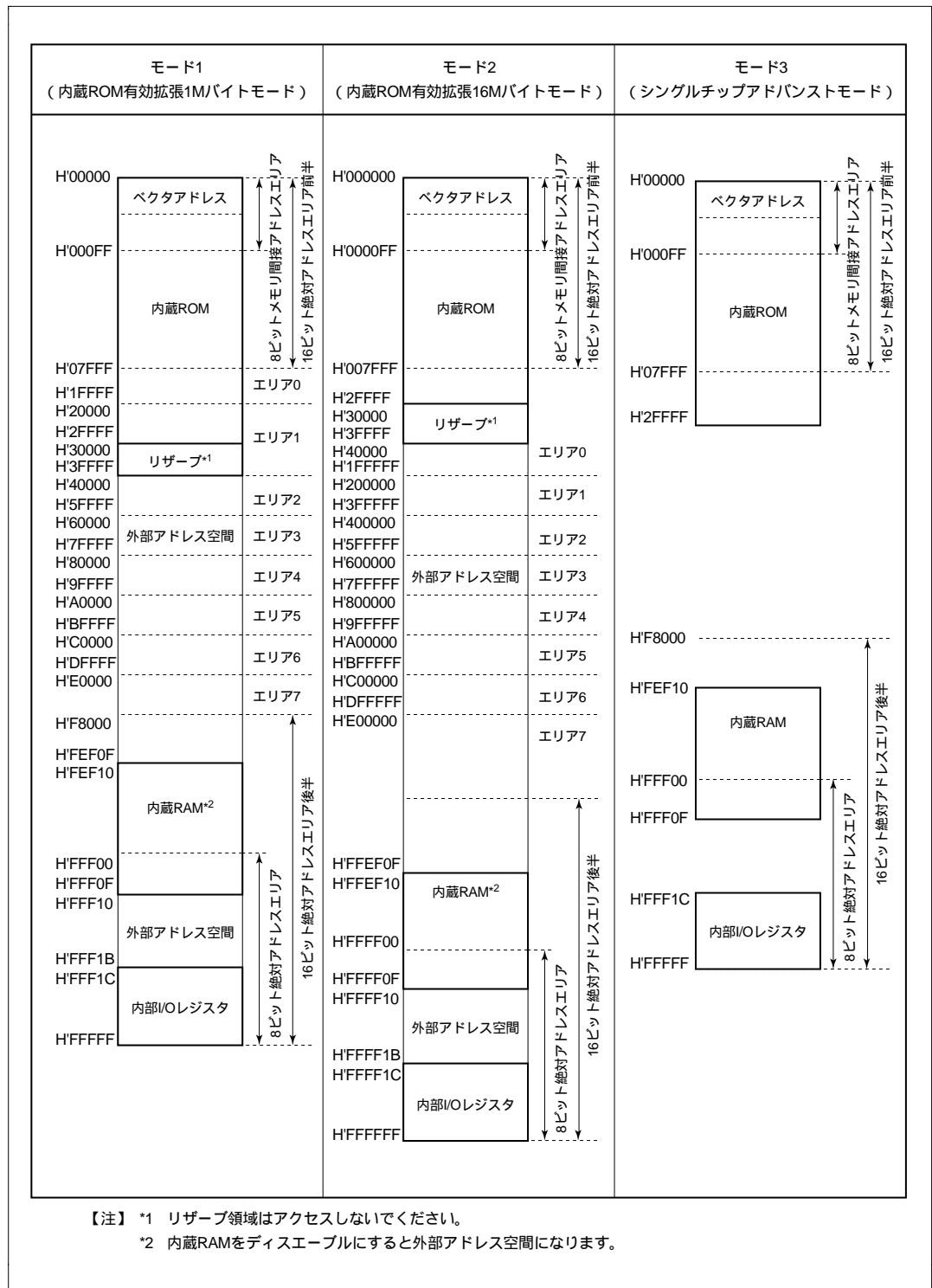


図 3.2 H8/3034 のメモリマップ

3. MCU動作モード

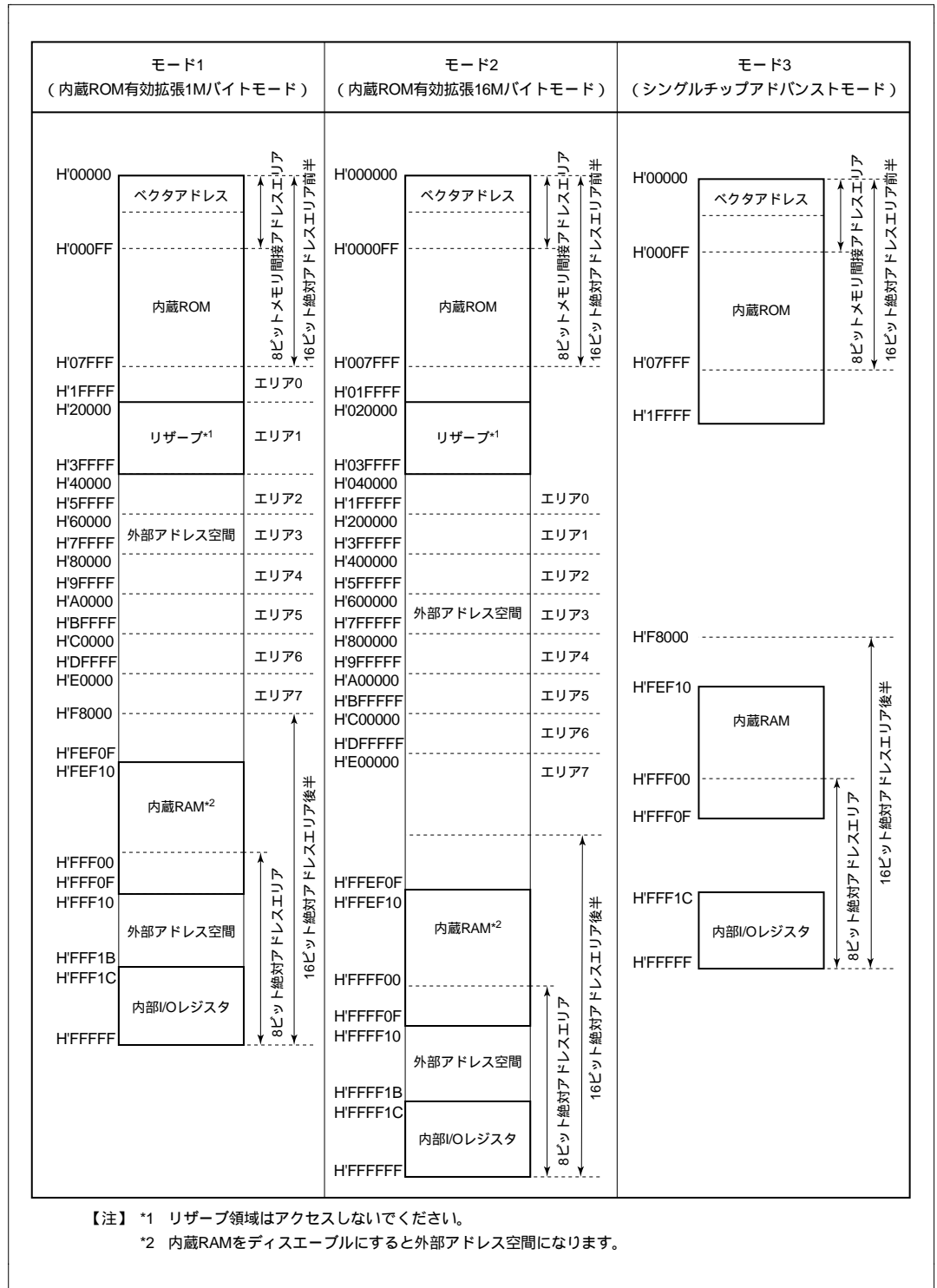


図 3.3 H8/3033 のメモリマップ

4. 例外処理

第4章 目次

4.1	概要.....	71
	4.1.1 例外処理の種類と優先度.....	71
	4.1.2 例外処理の動作.....	71
	4.1.3 例外処理要因とベクタテーブル.....	71
4.2	リセット.....	73
	4.2.1 概要.....	73
	4.2.2 リセットシーケンス.....	73
	4.2.3 リセット直後の割込み.....	75
4.3	割込み.....	76
4.4	トラップ命令.....	77
4.5	例外処理後のスタックの状態.....	78
4.6	スタック使用上の注意.....	79

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

例外処理要因	ベクタ番号	ベクタアドレス*1
		アドバンストモード
リセット	0	H'0000 ~ H'0003
システム予約	1	H'0004 ~ H'0007
	2	H'0008 ~ H'000B
	3	H'000C ~ H'000F
	4	H'0010 ~ H'0013
	5	H'0014 ~ H'0017
	6	H'0018 ~ H'001B
外部割り込み NMI	7	H'001C ~ H'001F
トラップ命令 (4 要因)	8	H'0020 ~ H'0023
	9	H'0024 ~ H'0027
	10	H'0028 ~ H'002B
	11	H'002C ~ H'002F
外部割り込み	IRQ ₀	H'0030 ~ H'0033
	IRQ ₁	H'0034 ~ H'0037
	IRQ ₂	H'0038 ~ H'003B
	IRQ ₃	H'003C ~ H'003F
	IRQ ₄	H'0040 ~ H'0043
システム予約	17	H'0044 ~ H'0047
	18	H'0048 ~ H'004B
	19	H'004C ~ H'004F
内部割り込み*2	20 ~ 60	H'0050 ~ H'0053 ~ H'00F0 ~ H'00F3

【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 10 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

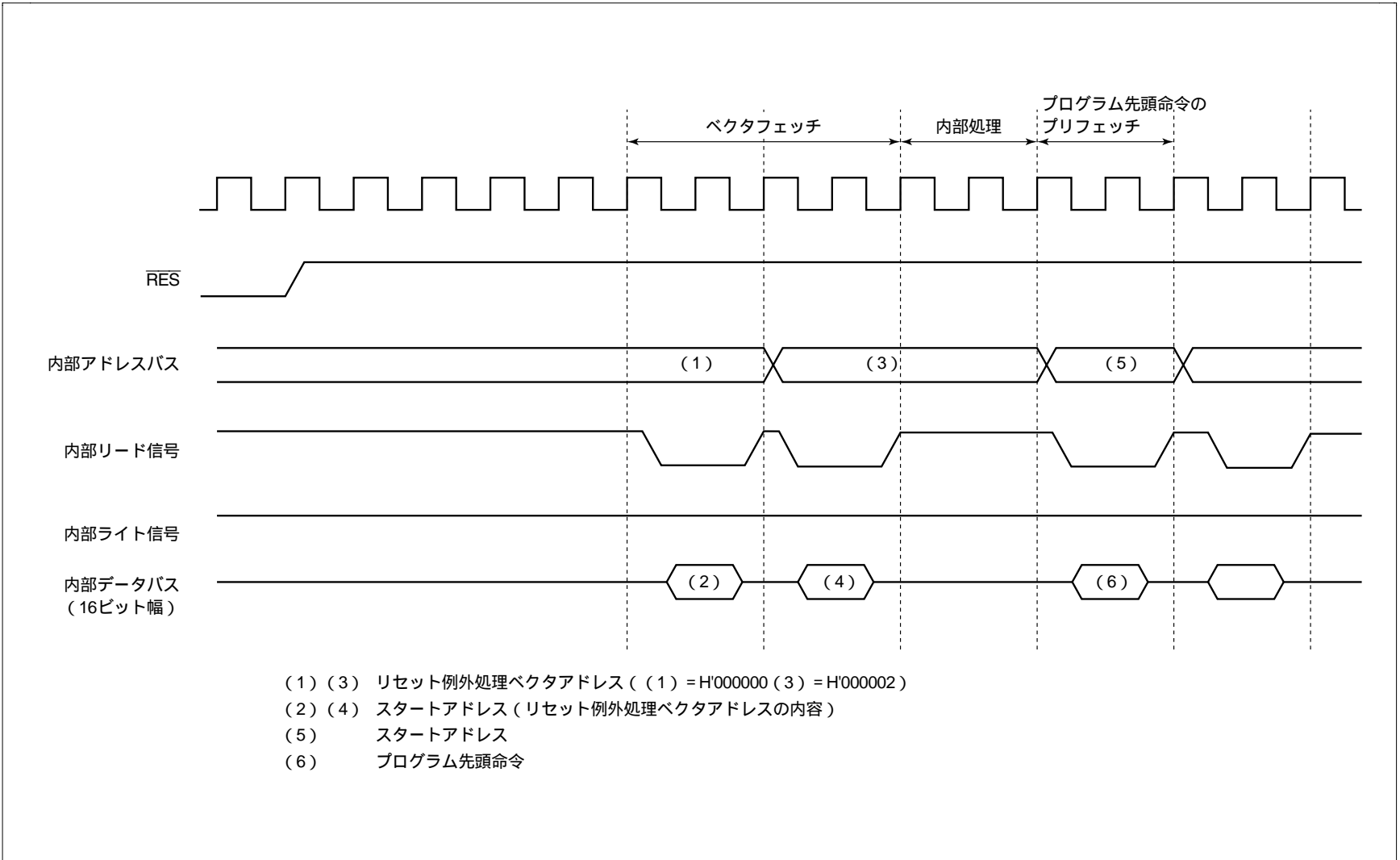
$\overline{\text{RES}}$ 端子が"Low"レベルになると本 LSI はリセット状態になります。

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。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) 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 に示します。



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 割り込み

割り込み例外処理を開始させる要因には、6つの外部割り込み（NMI、IRQ₀～IRQ₄）と、内蔵周辺モジュールからの要求による21の内部要因があります。割り込み要因と要因数を図4.3に示します。

割り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）、16ビットインテグレートドタイマユニット（ITU）、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。割り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は最優先の割り込みで、常に受け付けられます。割り込みは、割り込みコントローラによって制御されます。割り込みコントローラは、NMI以外の割り込みを2レベルの優先順位を設定して、多重割り込みの制御を行うことができます。割り込みの優先順位は、割り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。

割り込みについての詳細は「第5章 割り込みコントローラ」を参照してください。

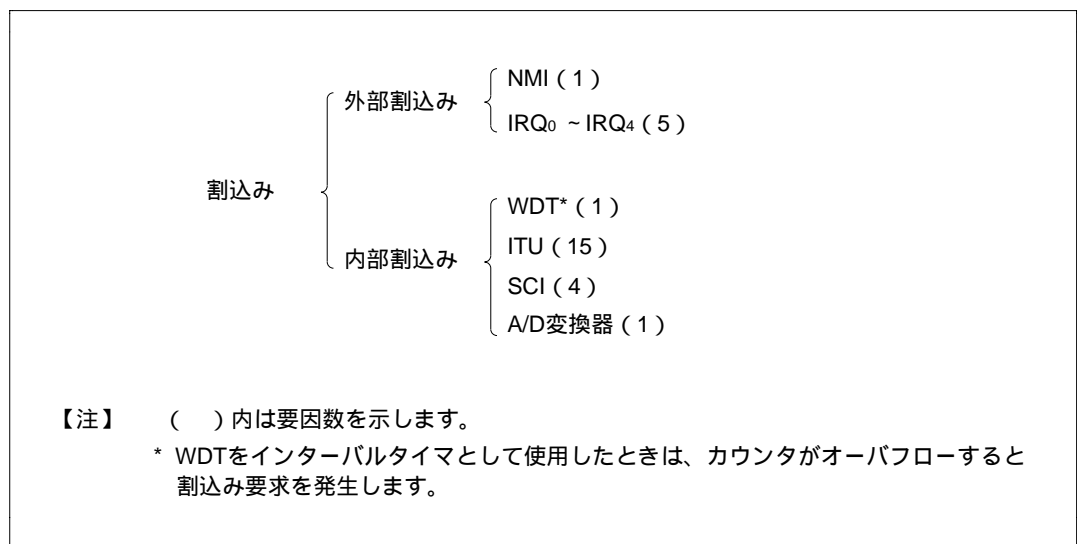


図 4.3 割り込み要因と要因数

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状態を図 4.4 に示します。

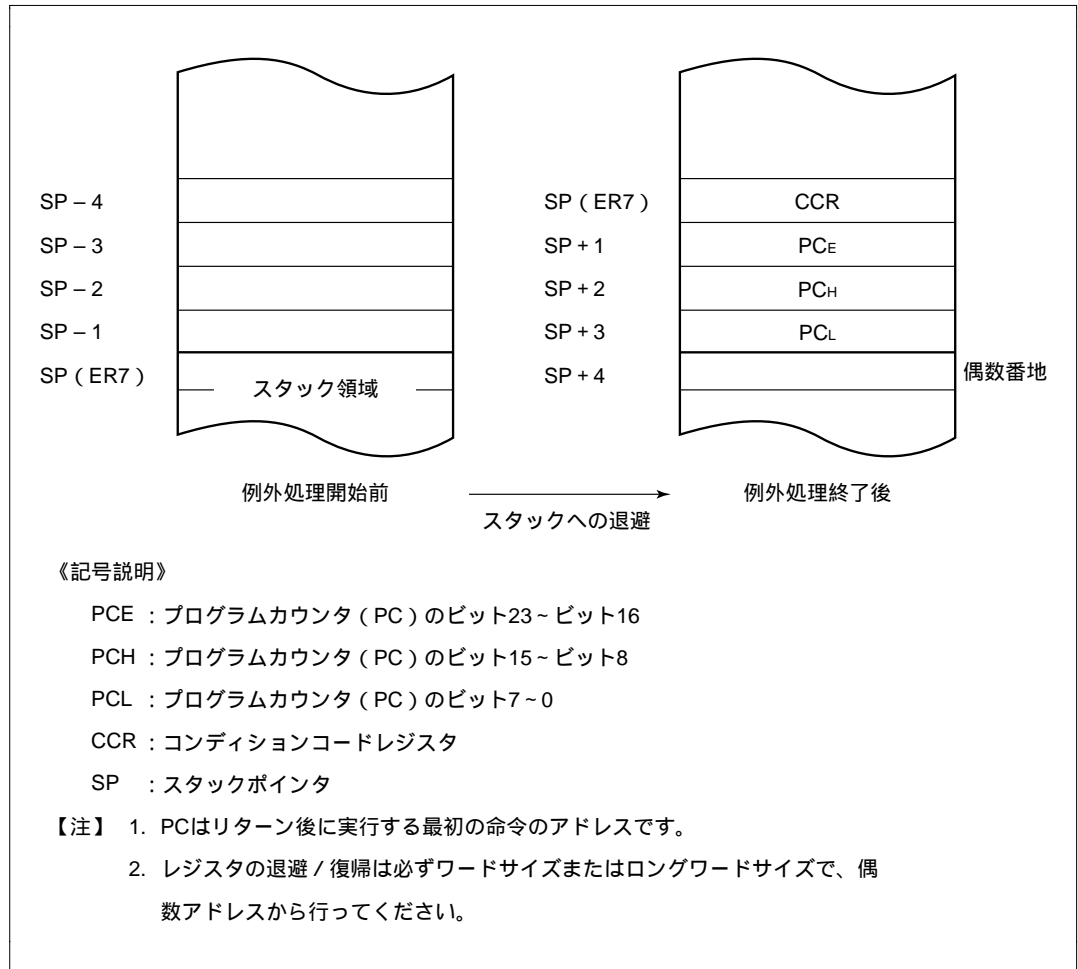


図 4.4 例外処理終了後のスタック状態 (アドバンスモード)

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4.5に示します。

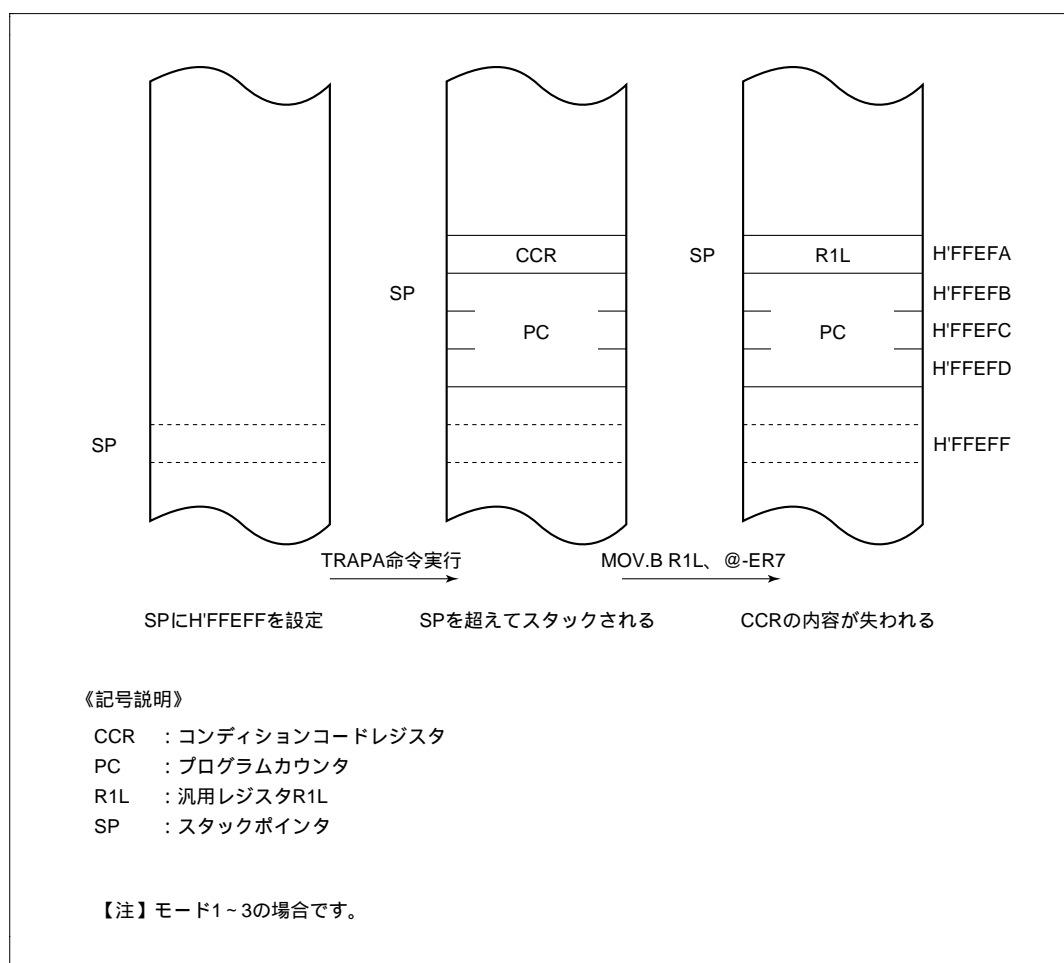


図 4.5 SP を奇数に設定したときの動作

5. 割り込みコントローラ

第5章 目次

5.1	概要.....	83
5.1.1	特長.....	83
5.1.2	ブロック図.....	83
5.1.3	端子構成.....	84
5.1.4	レジスタ構成.....	85
5.2	各レジスタの説明.....	86
5.2.1	システムコントロールレジスタ (SYSCR).....	86
5.2.2	インタラプトプライオリティレジスタ A、B (IPRA、IPRB).....	87
5.2.3	IRQ ステータスレジスタ (ISR).....	93
5.2.4	IRQ イネーブルレジスタ (IER).....	94
5.2.5	IRQ センスコントロールレジスタ (ISCR).....	95
5.3	割り込み要因.....	96
5.3.1	外部割り込み.....	96
5.3.2	内部割り込み.....	97
5.3.3	割り込み例外処理ベクタテーブル.....	97
5.4	割り込み動作.....	100
5.4.1	割り込み動作の流れ.....	100
5.4.2	割り込み例外処理シーケンス.....	105
5.4.3	割り込み応答時間.....	106
5.5	使用上の注意.....	107
5.5.1	割り込み発生とディスエーブルとの競合.....	107
5.5.2	割り込みの受け付けを禁止している命令.....	107
5.5.3	EPMOV 命令実行中の割り込み.....	108
5.5.4	外部割り込み使用上の注意.....	108

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。

独立したベクタアドレス

すべての割り込み要因には独立のベクタアドレスが割り当てられており、割り込み処理ルーチンで要因を判別する必要がありません。

6 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 IRQ₀ ~ IRQ₄ は立下がりエッジ / レベルセンスを独立に選択できます。

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。

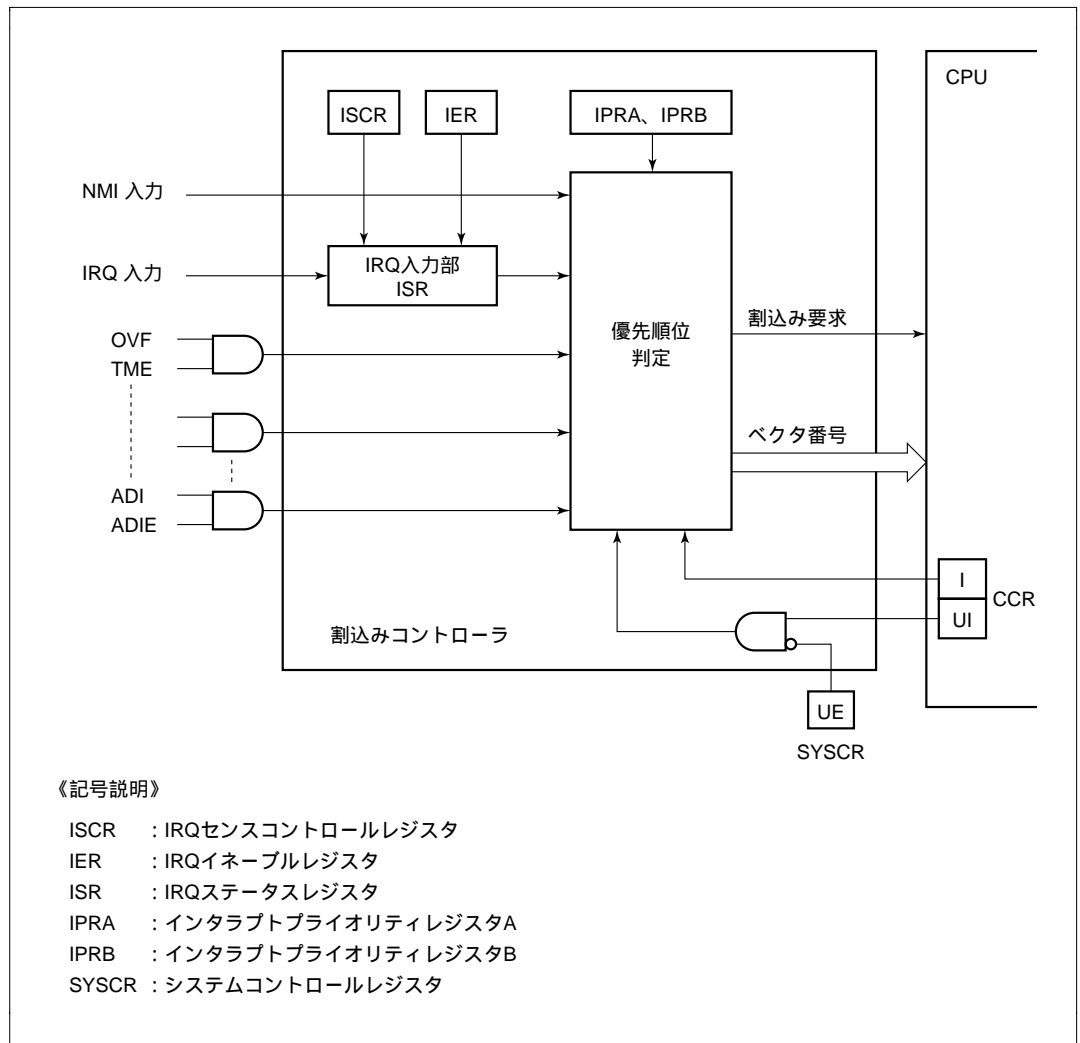


図 5.1 割り込みコントローラのブロック図

5.1.3 端子構成

割り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割り込み	NMI	入力	マスク不可能な外部割り込み、立上がりエッジ / 立下がりエッジ選択可能
外部割り込み要求 4~0	$\overline{IRQ}_4 \sim \overline{IRQ}_0$	入力	マスク可能な外部割り込み、立下がりエッジ / レベルセンス選択可能

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* ¹	名称	略称	R/W	初期値
H'FFF2	システムコントロールレジスタ	SYSCR	R/W	H'0B
H'FFF4	IRQ センスコントロールレジスタ	ISCR	R/W	H'00
H'FFF5	IRQ イネーブルレジスタ	IER	R/W	H'00
H'FFF6	IRQ ステータスレジスタ	ISR	R/(W) * ²	H'00
H'FFF8	インタラプトプライオリティレジスタ A	IPRA	R/W	H'00
H'FFF9	インタラプトプライオリティレジスタ B	IPRB	R/W	H'00

【注】 *1 アドレスの下位 16 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。

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 の有効/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'0B 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	—	RAME
初期値:	0	0	0	0	1	0	1	1
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ソフトウェアスタンバイ

スタンバイタイムセレクト2~0

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。

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。

リザーブビット

RAMイネーブル

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、割り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	
UE	説明
0	CCRのUIビットを割り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。

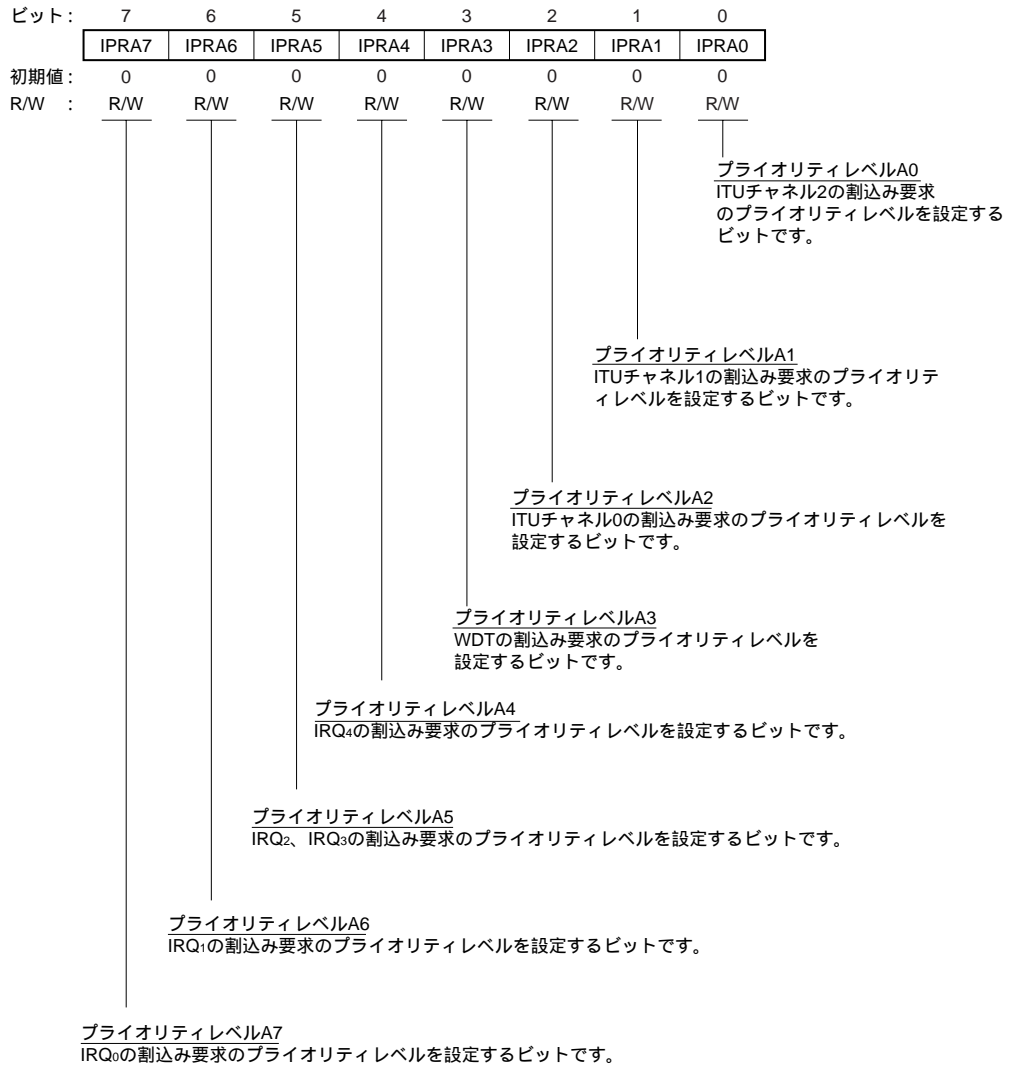
ビット2	
NMIEG	説明
0	NMI入力の立下がりエッジで割り込み要求を発生 (初期値)
1	NMI入力の立上がりエッジで割り込み要求を発生

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り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)

IRQ₀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 ₀ 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 ₀ 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)

IRQ₁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		
IPRA6	説明	
0	IRQ 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	IRQ ₁ 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 A5 (IPRA5)

IRQ₂、IRQ₃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		
IPRA5	説明	
0	IRQ ₂ 、IRQ 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	IRQ ₂ 、IRQ ₃ の割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)

IRQ₄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4		
IPRA4	説明	
0	IRQ ₄ 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	IRQ ₄ の割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		
IPRA3	説明	
0	WDT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)

ITU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		
IPRA2	説明	
0	ITU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	ITU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)

ITU 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	
IPRA1	説明
0	ITU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	ITU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)

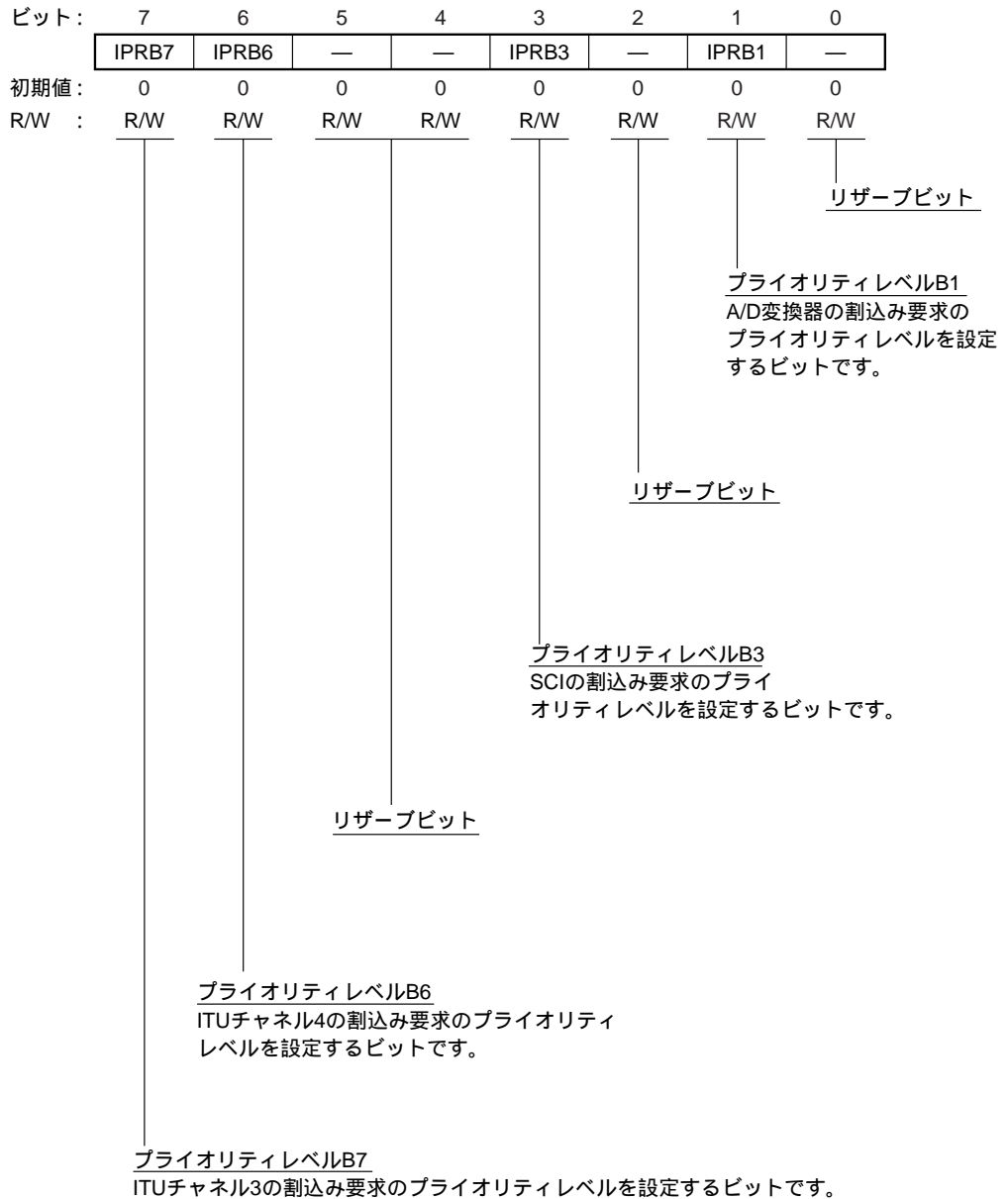
ビット0：プライオリティレベルA0 (IPRA0)

ITU 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	
IPRA0	説明
0	ITU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	ITU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)

ITU チャンネル 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 7	
IPRB7	説明
0	ITU チャンネル 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	ITU チャンネル 3 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)

ITU チャンネル 4 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	
IPRB6	説明
0	ITU チャンネル 4 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	ITU チャンネル 4 の割込み要求はプライオリティレベル 1 (優先)

ビット5、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 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	
IPRB3	説明
0	SCI 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 の割込み要求はプライオリティレベル 1 (優先)

ビット2：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1：プライオリティレベル B1 (IPRB1)

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 1	
IPRB1	説明
0	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	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、IRQ₄ ~ IRQ₀ 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
	—	—	—	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)*

リザーブビット
IRQ₄ ~ IRQ₀ フラグ
IRQ₄ ~ IRQ₀ 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7~5：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4~0：IRQ₄ ~ IRQ₀ フラグ (IRQ4F ~ IRQ0F)

IRQ₄ ~ IRQ₀ 割込み要求のステータスの表示を行います。

ビット4~0	説明
IRQ4F ~ IRQ0F	
0	[クリア条件] (初期値) (1) IRQ _n F = "1"の状態では IRQ _n F フラグをリードした後、IRQ _n F フラグに"0"をライトしたとき (2) IRQ _n SC = "0"、 $\overline{\text{IRQ}}_n$ 入力が"High"レベルの状態では 割込み例外処理を実行したとき (3) IRQ _n SC = "1"の状態では IRQ _n 割込み例外処理を実行したとき
1	[セット条件] (1) IRQ _n SC = "0"の状態では $\overline{\text{IRQ}}_n$ 入力が"Low"レベルになったとき (2) IRQ _n SC = "1"の状態では $\overline{\text{IRQ}}_n$ 入りに立上がりエッジが発生したとき

(n = 4 ~ 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、IRQ₀ ~ IRQ₄ 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
	—	—	—	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

IRQ₄ ~ IRQ₀ イネーブル
IRQ₄ ~ IRQ₀ 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 ~ 5 : 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 4 ~ 0 : IRQ₄ ~ IRQ₀ イネーブル (IRQ4E ~ IRQ0E)

IRQ₄ ~ IRQ₀ 割り込みを許可/禁止するかを選択します。

ビット 4 ~ 0	説明
0	IRQ ₄ ~ IRQ ₀ 割り込みを禁止 (初期値)
1	IRQ ₄ ~ IRQ ₀ 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、 $\overline{IRQ}_4 \sim \overline{IRQ}_0$ 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
	—	—	—	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

IRQ₄ ~ IRQ₀センスコントロール
IRQ₄ ~ 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7~5: 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4~0: IRQ₄ ~ IRQ₀ センスコントロール (IRQ4SC ~ IRQ0SC)

IRQ₄ ~ IRQ₀ 割込みを $\overline{IRQ}_4 \sim \overline{IRQ}_0$ 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4~0	説明	
IRQ4SC ~ IRQ0SC		
0	$\overline{IRQ}_4 \sim \overline{IRQ}_0$ 入力の "Low" レベルで割込み要求を発生	(初期値)
1	$\overline{IRQ}_4 \sim \overline{IRQ}_0$ 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、IRQ₄ ~ IRQ₀) と内部割り込み (21 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、IRQ₄ ~ IRQ₀ の 6 要因があります。このうち、NMI、IRQ₂ ~ IRQ₀ 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。

(2) IRQ₄ ~ IRQ₀ 割り込み

IRQ₄ ~ IRQ₀ 割り込みは $\overline{\text{IRQ}}_4$ ~ $\overline{\text{IRQ}}_0$ 端子の入力信号により要求されます。IRQ₄ ~ IRQ₀ 割り込みには次の特長があります。

- (1) $\overline{\text{IRQ}}_4$ ~ $\overline{\text{IRQ}}_0$ 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。
- (2) IRQ₄ ~ IRQ₀ 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。
- (3) IRQ₄ ~ IRQ₀ 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。

IRQ₄ ~ IRQ₀ 割り込みのブロック図を図 5.2 に示します。

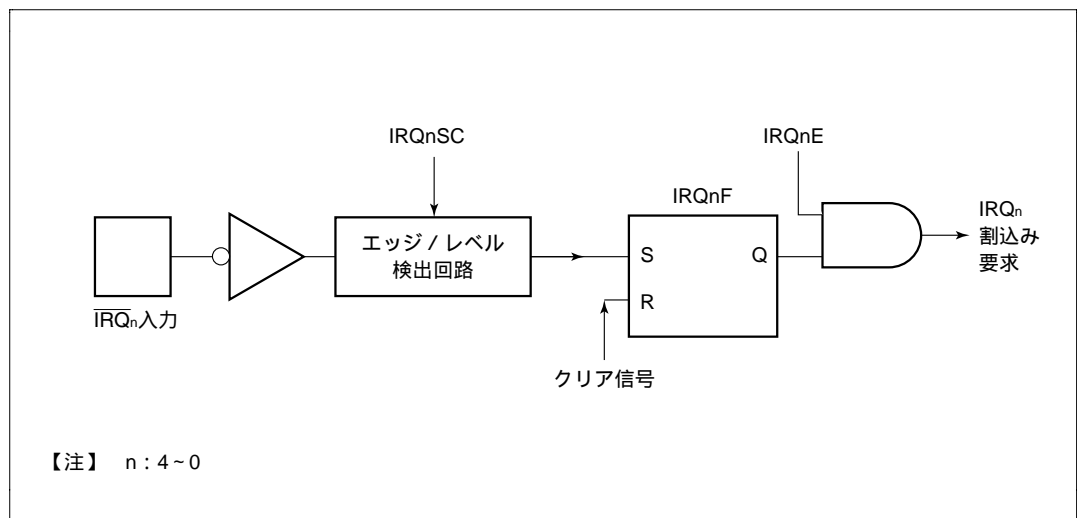


図 5.2 IRQ₄ ~ IRQ₀ 割り込みのブロック図

IRQnF のセットタイミングを図 5.3 に示します。

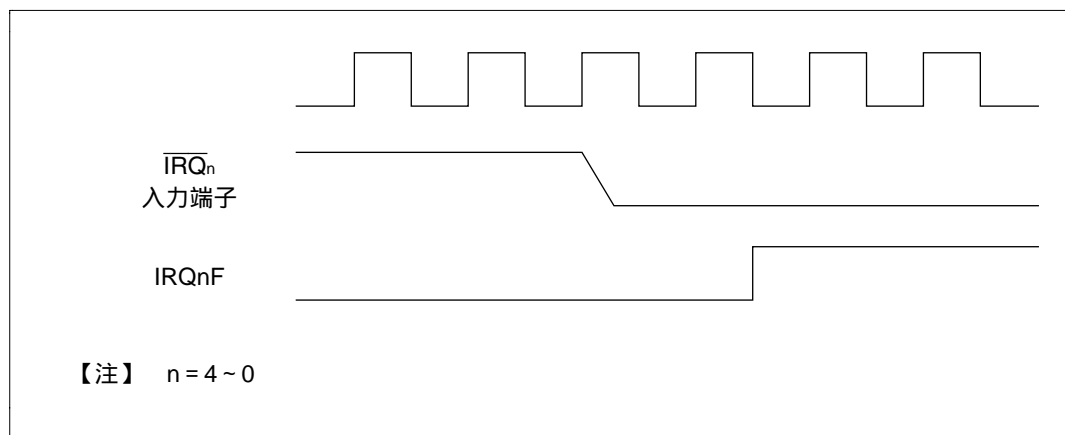


図 5.3 IRQnF セットタイミング

IRQ₄ ~ IRQ₀ 割り込み例外処理のベクタ番号は 12 ~ 16 です。

IRQ₄ ~ IRQ₀ 割り込みの検出は、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、外部割り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、SCI の入出力端子としては使用しないでください。

5.3.2 内部割り込み

内蔵周辺モジュールからの割り込みによる内部割り込みは 21 要因あります。

- (1) 各内蔵周辺モジュールには割り込み要求のステータスを表示するフラグと、これらの割り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 によって割り込みプライオリティレベルを設定できます。

5.3.3 割り込み例外処理ベクタテーブル

表 5.3 に割り込み例外処理要因とベクタアドレスおよび割り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 以外の割り込みの優先順位を変更することができます。

リセット後の割り込み優先順位は表 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

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			
NMI	外部端子	7	H'001C ~ H'001F			高 ↑
IRQ ₀		12	H'0030 ~ H'0033		IPRA7	
IRQ ₁		13	H'0034 ~ H'0037		IPRA6	
IRQ ₂		14	H'0038 ~ H'003B		IPRA5	
IRQ ₃		15	H'003C ~ H'003F			
IRQ ₄		16	H'0040 ~ H'0043		IPRA4	
リザーブ		17	H'0044 ~ H'0047			
		18	H'0048 ~ H'004B			
		19	H'004C ~ H'004F			
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		IPRA3	
リザーブ		21	H'0054 ~ H'0057			
		22	H'0058 ~ H'005B			
		23	H'005C ~ H'005F			
IMIA0 (コンペアマッチ/インプット キャプチャ A0)	ITU チャンネル 0	24	H'0060 ~ H'0063		IPRA2	
IMIB0 (コンペアマッチ/インプット キャプチャ B0)		25	H'0064 ~ H'0067			
OVI0 (オーバーフロー-0)		26	H'0068 ~ H'006B			
リザーブ		27	H'006C ~ H'006F			
IMIA1 (コンペアマッチ/インプット キャプチャ A1)	ITU チャンネル 1	28	H'0070 ~ H'0073		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			
OVI1 (オーバーフロー-1)		30	H'0078 ~ H'007B			
リザーブ		31	H'007C ~ H'007F			
IMIA2 (コンペアマッチ/インプット キャプチャ A2)	ITU チャンネル 2	32	H'0080 ~ H'0083		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			
OVI2 (オーバーフロー-2)		34	H'0088 ~ H'008B			
リザーブ		35	H'008C ~ H'008F			
IMIA3 (コンペアマッチ/インプット キャプチャ A3)	ITU チャンネル 3	36	H'0090 ~ H'0093		IPRB7	
IMIB3 (コンペアマッチ/インプット キャプチャ B3)		37	H'0094 ~ H'0097			
OVI3 (オーバーフロー-3)		38	H'0098 ~ H'009B			
リザーブ		39	H'009C ~ H'009F			

割り込み要因	要因発送元	ベクタ 番号	ベクタアドレス*	IPR	優先 順位
			アドバンストモード		
IMIA4 (コンペアマッチ/インプット キャプチャ A4)	ITU チャンネル 4	40	H'00A0 ~ H'00A3	IPRB6	↑
IMIB4 (コンペアマッチ/インプット キャプチャ B4)		41	H'00A4 ~ H'00A7		
OVI4 (オーバフロー-4)		42	H'00A8 ~ H'00AB		
リザーブ		43	H'00AC ~ H'00AF		
		44	H'00B0 ~ H'00B3		
		45	H'00B4 ~ H'00B7		
		46	H'00B8 ~ H'00BB		
		47	H'00BC ~ H'00BF		
		48	H'00C0 ~ H'00C3		
		49	H'00C4 ~ H'00C7		
		50	H'00C8 ~ H'00CB		
		51	H'00CC ~ H'00CF		
		ERI (受信エラー)	SCI		52
RXI (受信データフル)	53	H'00D4 ~ H'00D7			
TXI (送信データエンプティ)	54	H'00D8 ~ H'00DB			
TEI (送信終了)	55	H'00DC ~ H'00DF			
リザーブ		56	H'00E0 ~ H'00E3	IPRB2	
		57	H'00E4 ~ H'00E7		
		58	H'00E8 ~ H'00EB		
		59	H'00EC ~ H'00EF		
ADI (A/D エンド)	A/D	60	H'00F0 ~ H'00F3	IPRB1	低

【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。

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。
	1	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) UE ビット = "1" の場合

IRQ₀ ~ IRQ₄ 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。

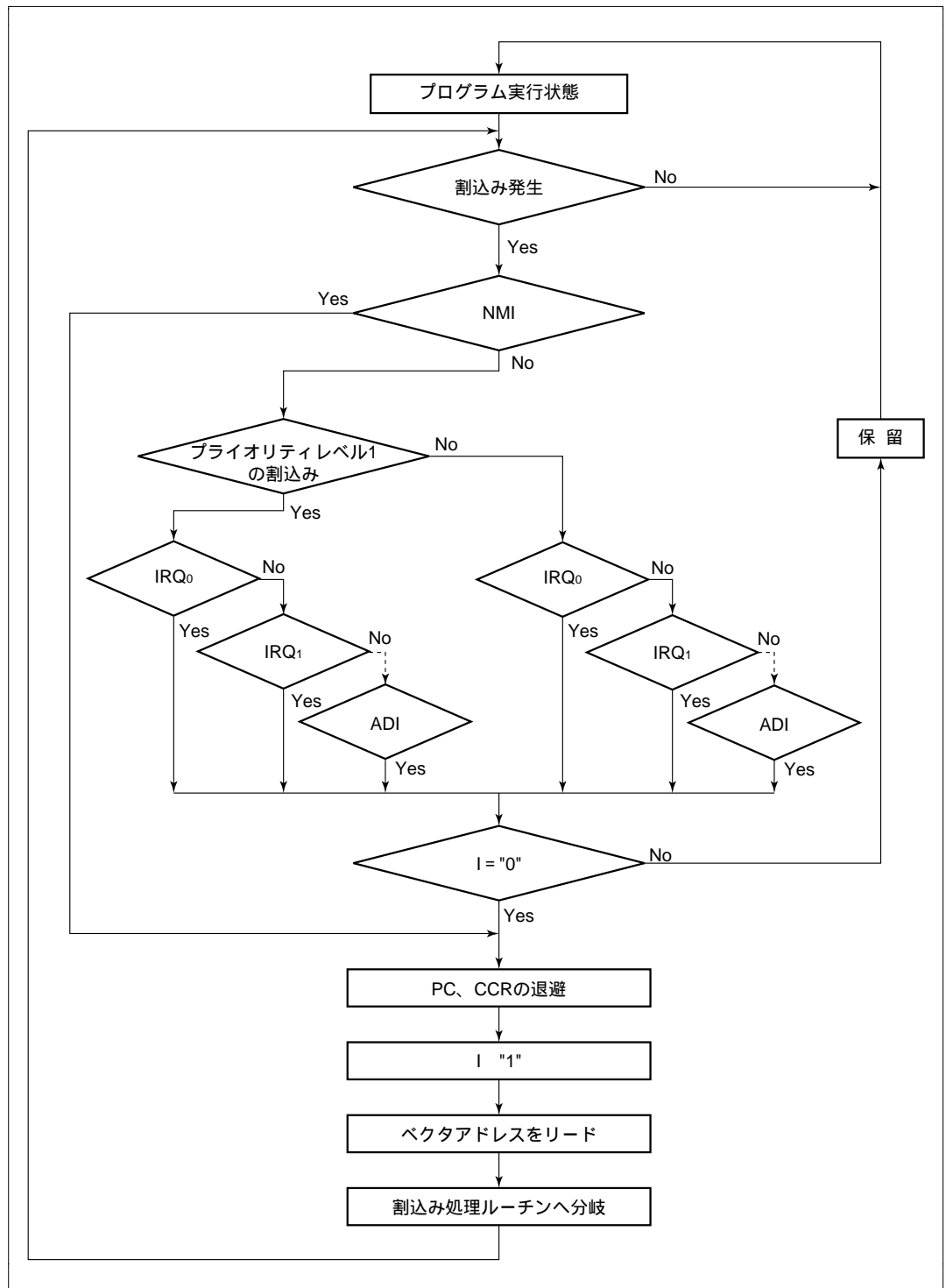


図 5.4 UE = "1" の場合の割り込み受け付けまでのフロー

- (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 に示す優先度に従って、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

IRQ₀ ~ IRQ₄ 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 (IRQ₂、IRQ₃ 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 > IRQ₂ > IRQ₃ > IRQ₀)
 - (b) I = "1"、UI = "0" のとき、NMI、IRQ₂、IRQ₃ 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。

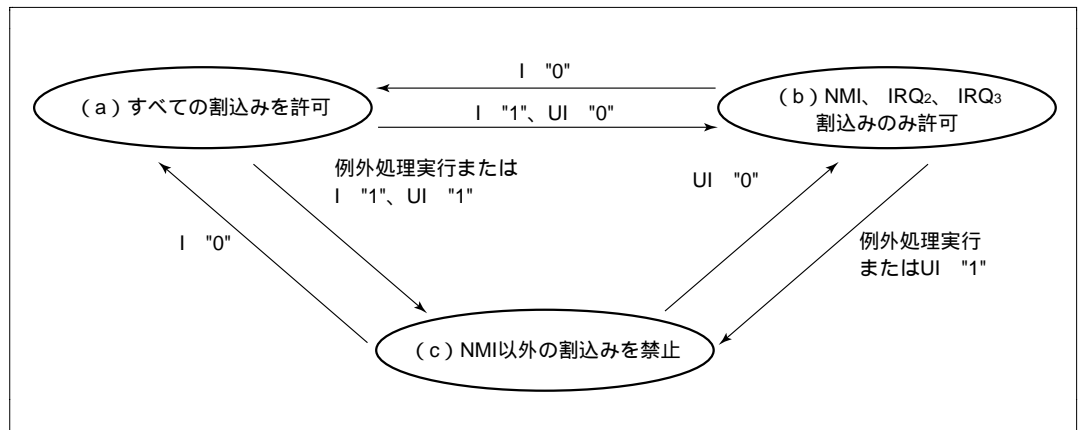


図 5.5 割り込み許可 / 禁止状態の遷移例

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。

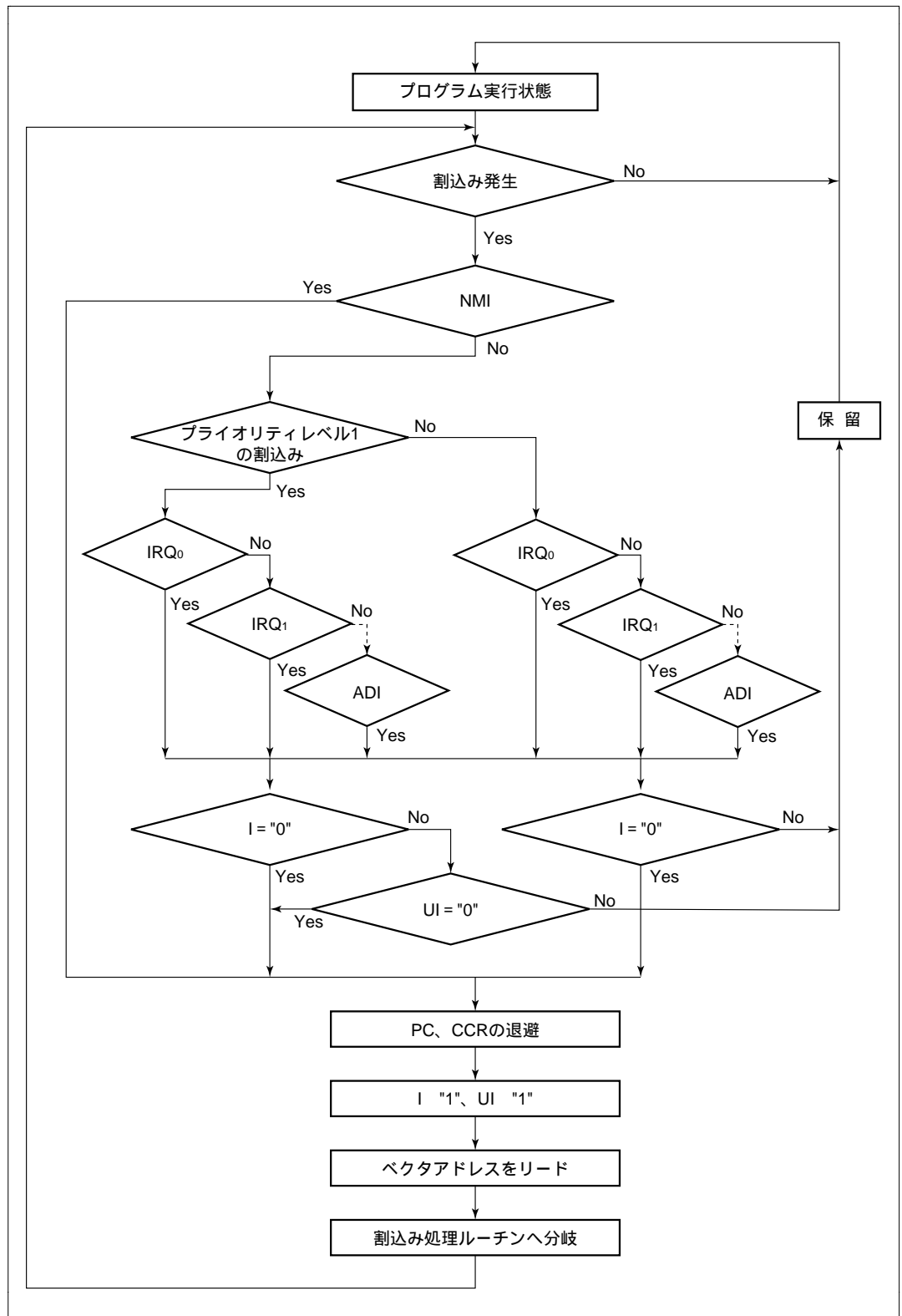


図 5.6 UE = "0" の場合の割り込み受け付けまでのフロー

5.4.2 割り込み例外処理シーケンス

モード1で、プログラム領域とスタック領域を内部メモリにといった場合の割り込みシーケンスを図5.7に示します。

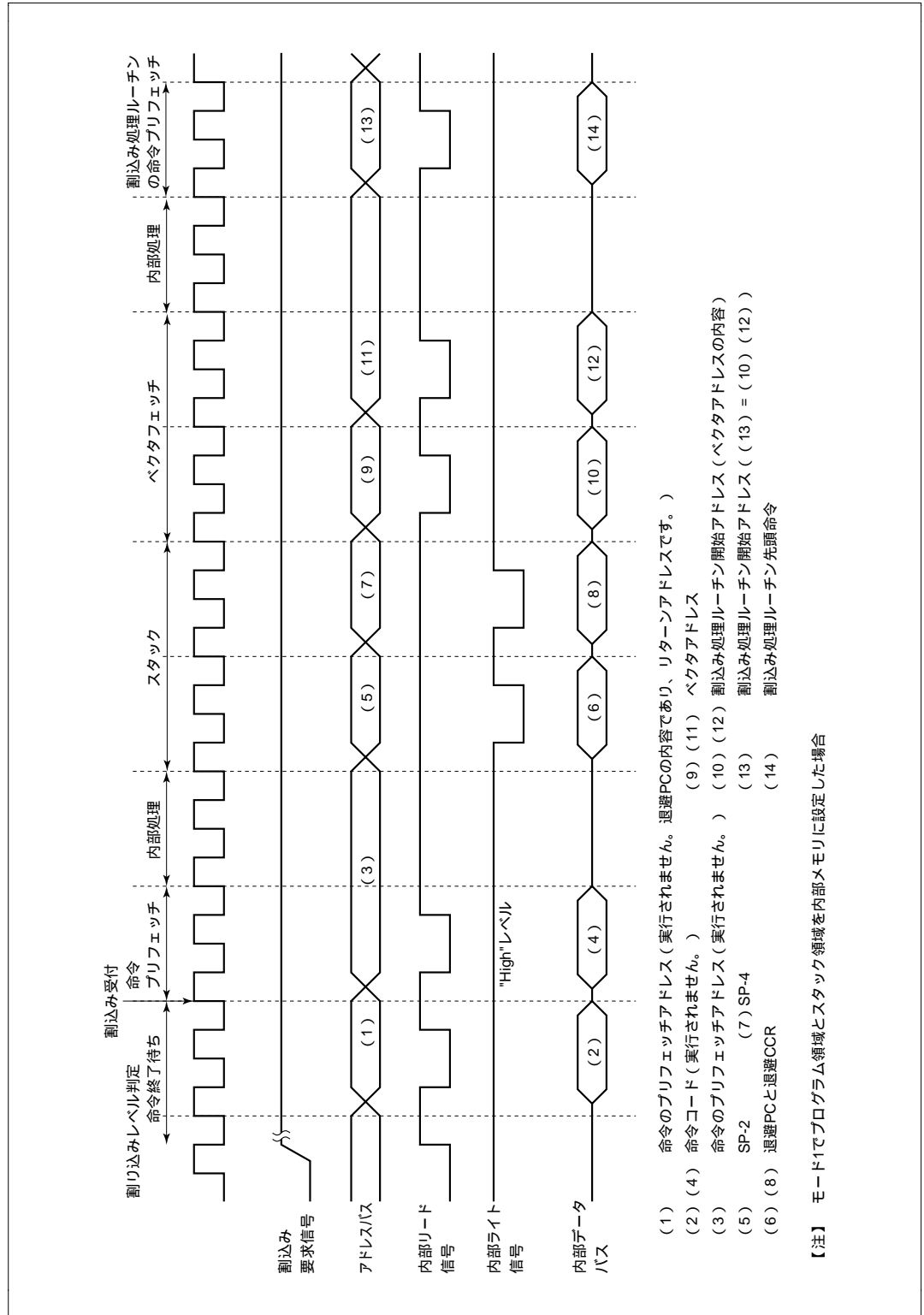


図 5.7 割り込みシーケンス（モード1、スタック内部メモリ）

5.4.3 割込み応答時間

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

表 5.5 割込み応答時間

No.	項目	内蔵メモリ	外部メモリ	
			8ビットバス	
			2状態	3状態
1	割込み優先順位判定		2 ^{*1}	
2	実行中の命令が終了するまでの最大待ち状態数	1～23	1～27	1～31 ^{*4}
3	PC、CCRのスタック	4	8	12 ^{*4}
4	ベクタフェッチ	4	8	12 ^{*4}
5	命令フェッチ ^{*2}	4	8	12 ^{*4}
6	内部処理 ^{*3}	4	4	4
	合計	19～41	31～57	43～73

【注】 *1 内部割込みの場合 1 状態となります。

*2 割込み受け付け後のプリフェッチおよび割込み処理ルーチンのプリフェッチ

*3 割込み受け付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、状態数が増加します。

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。

ITU の TIER 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。

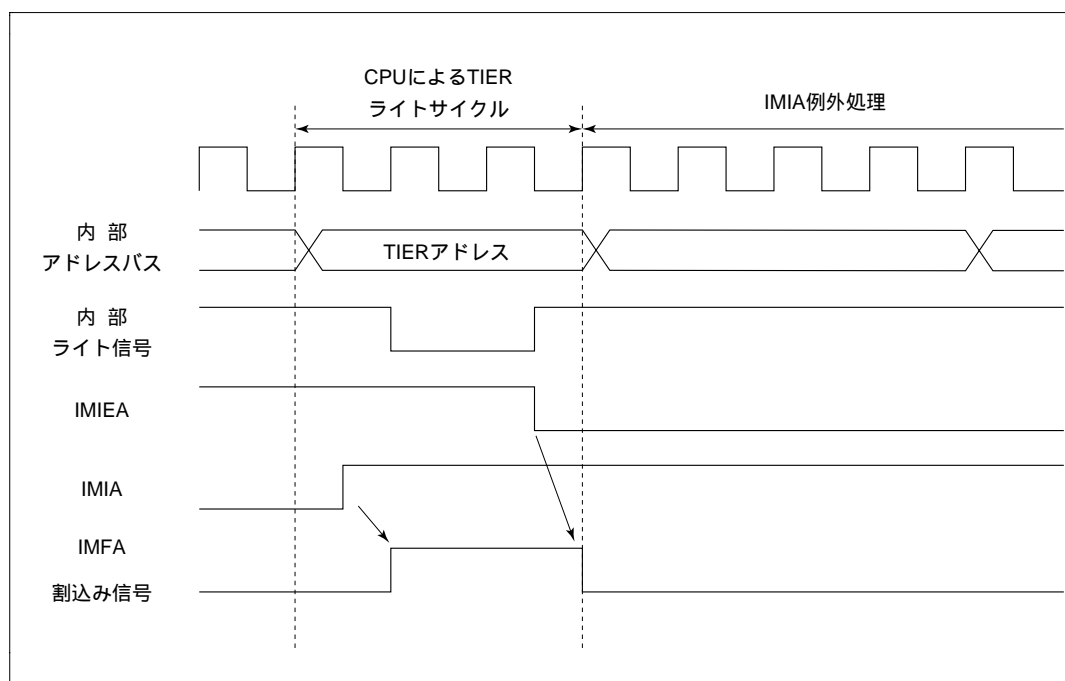


図 5.8 割り込みの発生とディスエーブルの競合

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
```

5.5.4 外部割り込み使用上の注意

IRQnF フラグは、IRQnF = "1" の状態で IRQnF フラグをリードした後、IRQnF フラグに "0" をライトしたとき、クリアされる仕様になっています。しかしながら、IRQnF = "1" の状態をリードしていないにも関わらず、IRQnF フラグに "0" をライトしただけで、誤って IRQnF フラグがクリアされ、割り込み例外処理が実行されない場合があります。

これは、以下のような条件が満たされたときに起こります。

(1) 設定条件

- (1) 複数の外部割り込み (IRQa、IRQb) を使用している。
- (2) IRQaF フラグは "0" ライトによるクリア、IRQbF フラグはハードウェアによるクリアと異なったクリア方法を用いている。
- (3) IRQaF フラグのクリアに、IRQ ステータスレジスタ (ISR) に対するビット操作命令を使用、もしくは ISR をバイト単位でリードし、IRQaF フラグのビットをクリアし、その他のビットはリードした値をバイト単位でライトしている。

(2) 発生条件

- (1) IRQaF = "1" の状態で IRQaF フラグをクリアするための ISR レジスタのリードが実行され、その後、割り込み例外処理の発生により、IRQbF フラグがクリアされた。
- (2) IRQaF フラグのクリアと IRQb の発生 (IRQaF フラグのセット) が競合したとき。(IRQaF フラグをクリアするための ISR をリードしたときには、IRQbF = "0" であったが、ISR にライトするまでの期間に IRQbF = "1" となってしまった状態)

上記の設定条件 (1) ~ (3) と発生条件 (1)、(2) がすべて成立した場合、発生条件 (2) の ISR のライト時に IRQbF は誤ってクリアされ、割り込み例外処理は実行されません。ただし、発生条件 (1)、(2) の間に一度でも IRQbF フラグに "0" をライトすると、IRQbF

フラグが誤ってクリアされることはありません。

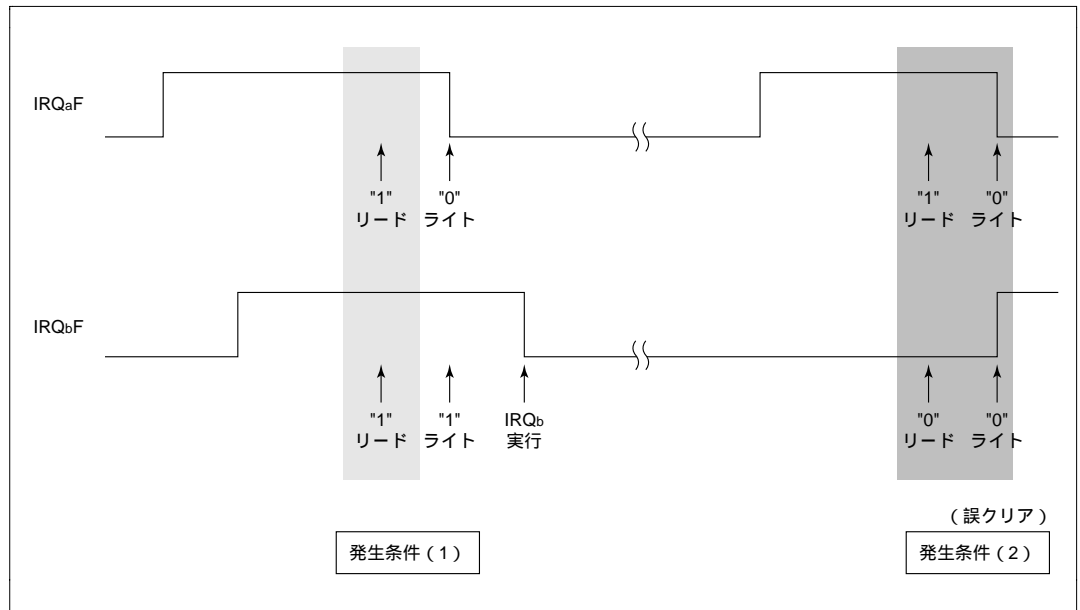


図 5.9 割込み例外処理が実行されない場合の IRQnF フラグ

このような場合には、以下の対策 1 または 2 を行い、対処してください。

対策 1

IRQaF フラグをクリアする際には、ビット演算命令を使用せず、ISR をバイト単位でリードし、IRQaF フラグだけを"0"、その他のビットは"1"となる値をバイト単位で書き込んでください。

例：a = 0 の場合

```
MOV.B @ISR,R0L
MOV.B #HFE,R0L
MOV.B R0L,@ISR
```

対策 2

IRQb 割込み例外処理の中で、IRQbF フラグをクリアするダミー処理を行ってください。

例：b = 1 の場合

```
IRQB MOV.B #HFD,R0L
MOV.B R0L,@ISR
.
.
.
```

6. バスコントローラ

第6章 目次

6.1	概要.....	113
	6.1.1 特長.....	113
	6.1.2 ブロック図.....	113
	6.1.3 端子構成.....	114
	6.1.4 レジスタ構成.....	114
6.2	各レジスタの説明.....	115
	6.2.1 アクセスステートコントロールレジスタ (ASTCR).....	115
	6.2.2 ウェイトコントロールレジスタ (WCR).....	116
	6.2.3 ウェイトステートコントローライネーブルレジスタ (WCER).....	117
	6.2.4 アドレスコントロールレジスタ (ADRCR).....	117
6.3	動作説明.....	119
	6.3.1 エリア分割.....	119
	6.3.2 バス制御信号タイミング.....	120
	6.3.3 ウェイトモード.....	122
	6.3.4 メモリとの接続例.....	128
6.4	使用上の注意.....	129
	6.4.1 レジスタライトタイミング.....	129
	6.4.2 ASTCR と ABWCR 設定上の注意事項.....	129

6.1 概要

本 LSI はバスコントローラを内蔵しており、外部アドレス空間を 8 つのエリアに分割し、エリアごとにバス仕様を設定し、バス制御を行います。これにより、複数のメモリを容易に接続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 0~7 に分割し、エリアごとに独立して設定可能

- ・ 1M バイトモードでは 128k バイトごとにエリアを設定
- ・ 16M バイトモードでは 2M バイトごとにエリアを設定
- ・ 2 ステートアクセス空間 / 3 ステートアクセス空間の選択可能

4 種類のウェイトモード

- ・ プログラブルウェイトモード、端子オートウェイトモード、端子ウェイトモード 0、1 を選択可能
- ・ 0~3 ステートのウェイトステートを自動的に挿入可能

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。

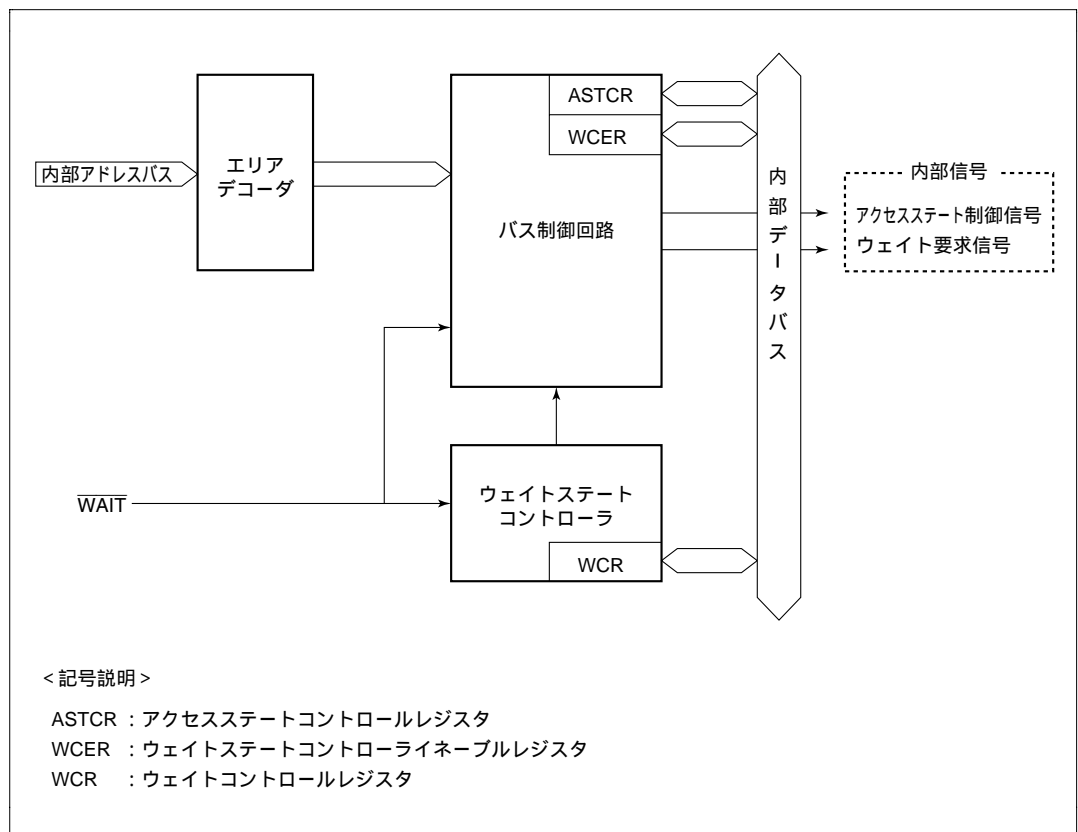


図 6.1 バスコントローラのブロック図

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
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
ライト	\overline{WR}	出力	外部アドレス空間をライトし、データバス ($D_7 \sim D_0$) 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

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*	名称	略称	R/W	初期値
H'FFED	アクセスステートコントロールレジスタ	ASTCR	R/W	H'FF
H'FFEE	ウェイトコントロールレジスタ	WCR	R/W	H'F3
H'FFEF	ウェイトステートコントローララインャブルレジスタ	WCER	R/W	H'FF
H'FFF3	アドレスコントロールレジスタ	ADRCCR	R/W	H'FE (モード 1、3 時)
				H'0A (モード 2 時)

【注】 * アドレスは下位 16 ビットを示しています。

6.2 各レジスタの説明

6.2.1 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	
AST7~AST0	説明
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。

6.2.2 ウェイトコントロールレジスタ (WCR)

WCR は 8 ビットのリード/ライト可能なレジスタで、ウェイトステートコントローラ (WSC) のウェイトモードとウェイトステート数を設定します。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	WMS1	WMS0	WC1	WC0
初期値:	1	1	1	1	0	0	1	1
R/W :	—	—	—	—	R/W	R/W	R/W	R/W

リザーブビット

ウェイトモードセレクト1、0
ウェイトモードを設定するビットです。

ウェイトカウント1、0
挿入するウェイトステート数を
設定するビットです。

WCR はリセット、またはハードウェアスタンバイモード時に HF3 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: 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、2: ウェイトモードセレクト1、0 (WMS1、0)

ウェイトモードを設定します。

ビット3	ビット2	説明
WMS1	WMS0	
0	0	プログラマブルウェイトモード (初期値)
0	1	ウェイトステートコントローラによるウェイトを禁止
1	0	端子ウェイトモード1
1	1	端子オートウェイトモード

ビット1、0: ウェイトカウント1、0 (WC1、0)

外部3ステートアクセス空間をアクセスするときに、挿入するウェイトステート数を設定します。

ビット1	ビット0	説明
WC1	WC0	
0	0	WSC によるウェイトを禁止
0	1	1ステート挿入
1	0	2ステート挿入
1	1	3ステート挿入 (初期値)

6.2.3 ウェイトステートコントローライネーブルレジスタ (WCER)

WCER は 8 ビットのリード/ライト可能なレジスタで、外部 3 ステートアクセス空間について、WSC の動作を許可/禁止します。

ビット:	7	6	5	4	3	2	1	0
	WCE7	WCE6	WCE5	WCE4	WCE3	WCE2	WCE1	WCE0
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

ウェイトステートコントローライネーブル7~0
WSCの動作を許可/禁止するビットです。

WCER 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: ウェイトステートコントローライネーブル7~0 (WCE7~WCE0)

外部 3 ステートアクセス空間について、WSC の動作を許可/禁止します。

ビット7~0	
WCE7~WCE0	説明
0	WSC の動作を禁止 (端子ウェイトモード 0)
1	WSC の動作を許可 (初期値)

6.2.4 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、アドレスバス ($A_{23} \sim A_{21}$) 出力を選択します。

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

アドレス23~21イネーブル
PA₆~PA₄をアドレス出力端子
A₂₃~A₂₁として使用するた
めのビットです。

リザーブビット

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モード 1、3 のとき H'FE に、モード 2 のとき H'0A 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)

PA₄をアドレス出力端子 A₂₃として使用するためのビットです。"0"をライトすると PA₄はアドレス出力 A₂₃となります。モード3、4以外ではこのビットはライトできず、PA₄は通常のポート機能となります。

ビット7	
A ₂₃ E	説明
0	PA ₄ はアドレス出力端子 A ₂₃
1	PA ₄ は PA ₄ /TP ₄ /TIOCA ₁ 入出力端子 (初期値)

ビット6：アドレス22イネーブル (A₂₂E)

PA₅をアドレス出力端子 A₂₂として使用するためのビットです。"0"をライトすると PA₅はアドレス出力 A₂₂となります。モード3、4以外ではこのビットはライトできず、PA₅は通常のポート機能となります。

ビット6	
A ₂₂ E	説明
0	PA ₅ はアドレス出力端子 A ₂₂
1	PA ₅ は PA ₅ /TP ₅ /TIOCB ₁ 入出力端子 (初期値)

ビット5：アドレス21イネーブル (A₂₁E)

PA₆をアドレス出力端子 A₂₁として使用するためのビットです。"0"をライトすると PA₆はアドレス出力 A₂₁となります。モード3、4以外ではこのビットはライトできず、PA₆は通常のポート機能となります。

ビット5	
A ₂₁ E	説明
0	PA ₆ はアドレス出力端子 A ₂₁
1	PA ₆ は PA ₆ /TP ₆ /TIOCA ₂ 入出力端子 (初期値)

ビット4～0：リザーブビット

リザーブビットです。

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。

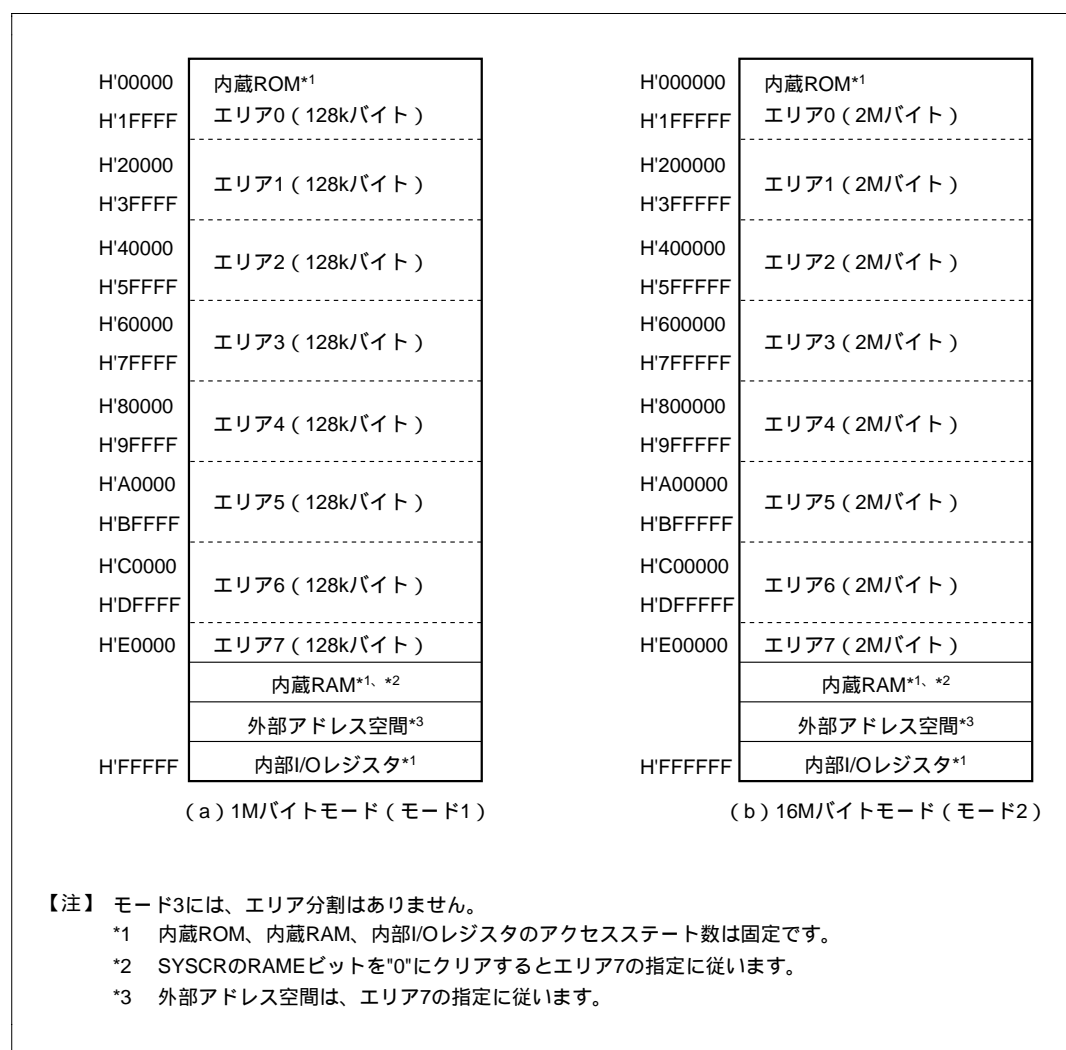


図 6.2 各動作モードにおけるアクセスエリアマップ

各エリアのバス仕様は、ASTCR、WCER、WCR で指定されます。
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

ASTCR	WCER	WCR		バス仕様		
ASTn	WCEn	WMS1	WMS0	バス幅	アクセス ステート	ウェイトモード
0				8	2	禁止
1	0			8	3	端子ウェイトモード 0
1	1	0	0	8	3	プログラマブルウェイトモード
		0	1	8	3	禁止
		1	0	8	3	端子ウェイトモード 1
		1	1	8	3	端子オートウェイトモード

【注】 n=0~7

6.3.2 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 6.3 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。
ウェイトステートを挿入することができます。

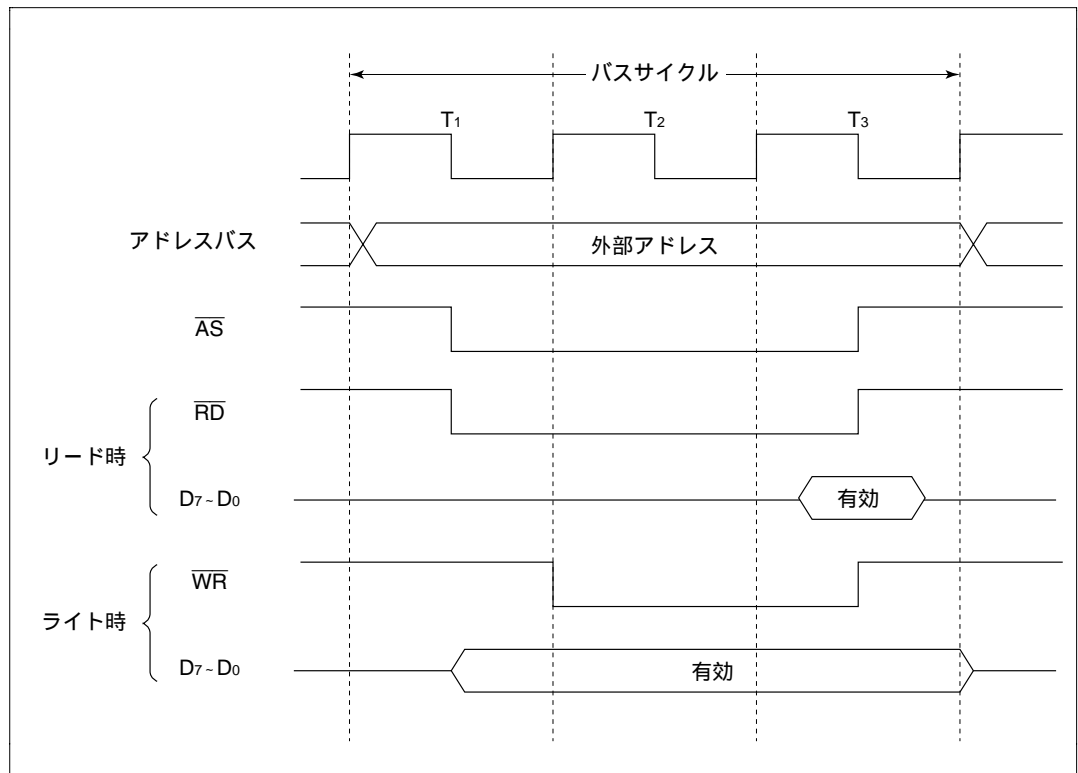


図 6.3 8 ビット 3 ステートアクセス空間のバス制御信号タイミング

(2) 8 ビット 2 ステートアクセス空間

図 6.4 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。
ウェイトステートを挿入することはできません。

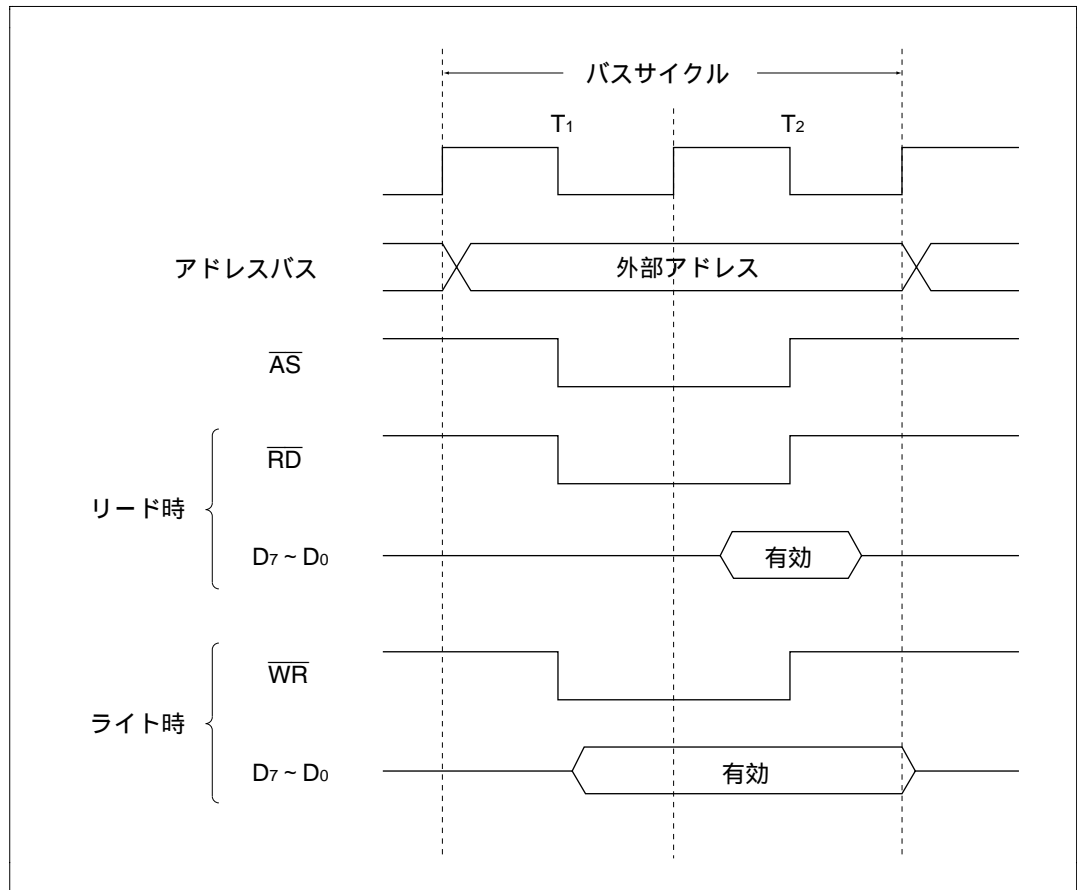


図 6.4 8 ビット 2 ステートアクセス空間のバス制御信号タイミング

6.3.3 ウェイトモード

ウェイトモードには、4種類のモードがありエリアごとに選択できます。ウェイトモードの選択方法を表6.4に示します。

表6.4 ウェイトモードの選択

ASTCR	WCER	WCR		WSC動作	ウェイトモード
ASTn ビット	WCEn ビット	WMS1 ビット	WMS0 ビット		
0				禁止	ウェイト禁止
1	0			禁止	端子ウェイトモード0
1	1	0	0	許可	プログラマブルウェイトモード
		0	1	許可	ウェイト禁止
		1	0	許可	端子ウェイトモード1
		1	1	許可	端子オートウェイトモード

【注】 n=0~7

ASTn、WCEn ビットは、エリアごとに独立して選択可能です。WMS1、0 ビットは各エリアに共通です。したがって、WSC の動作を許可したエリアの動作は同一になります。

(1) 端子ウェイトモード 0

端子ウェイトモード 0 では、WSC の動作が禁止され、 $\overline{\text{WAIT}}$ 端子によるウェイトステートのみが挿入されます。外部 3 ステートアクセス空間をアクセスするとき、 T_2 ステートの立下がりのタイミングで $\overline{\text{WAIT}}$ 端子が "Low" レベルであると、 T_w ステートが挿入されます。 $\overline{\text{WAIT}}$ 端子が "Low" レベルに保持されると $\overline{\text{WAIT}}$ 端子が "High" レベルに立上がるまで T_w が挿入されます。

このタイミングを図 6.5 に示します。

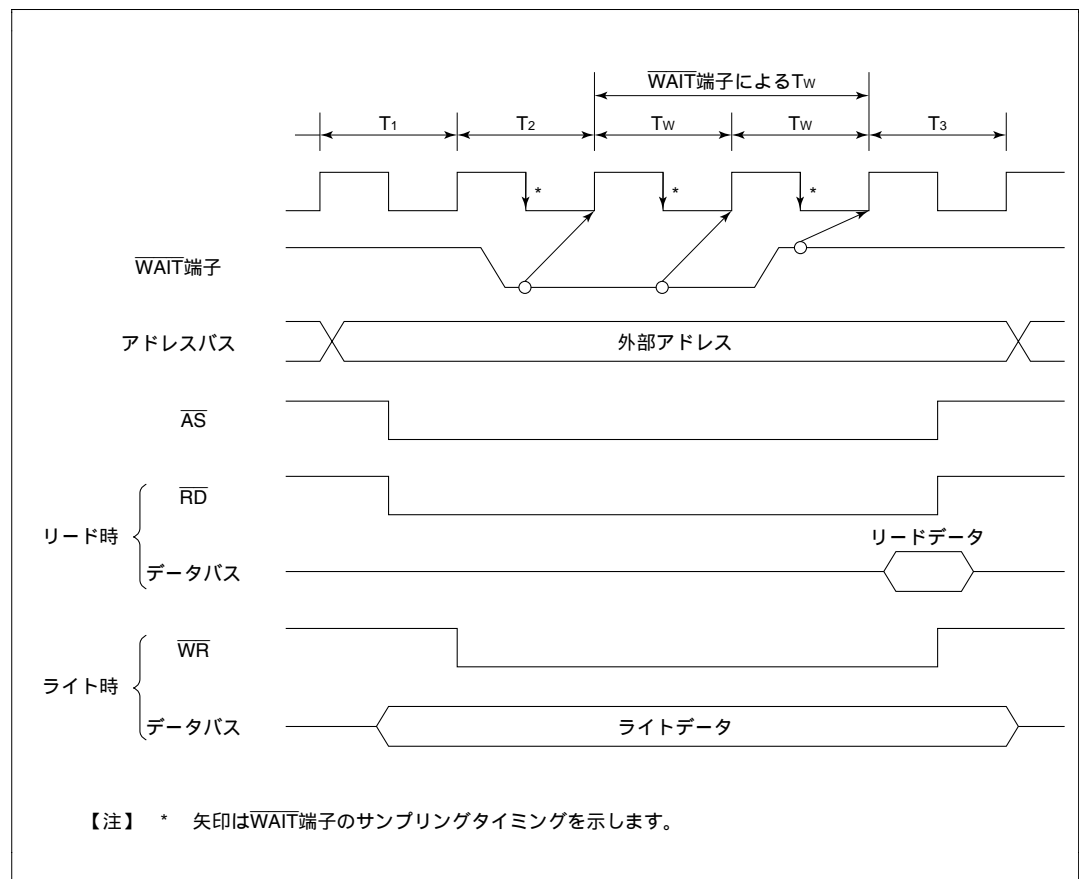


図 6.5 端子ウェイトモード 0

(2) 端子ウェイトモード 1

端子ウェイトモード 1 では、外部 3 ステートアクセス空間をアクセスすると、常に WC1、0 ビットにより設定されたステート数だけ T_w が挿入されます。この最後の T_w の 立下がりのタイミングで $\overline{\text{WAIT}}$ 端子を "Low" レベルにすることで、さらに T_w を挿入することができます。 $\overline{\text{WAIT}}$ 端子が "Low" レベルに保持されると、 $\overline{\text{WAIT}}$ 端子が "High" レベルに立上がるまで T_w が挿入されます。

端子ウェイトモード 1 は、4 ステート以上の T_w を挿入する場合や、外部デバイスごとに挿入すると T_w 数を変える場合などに有効です。

ウェイトカウントが "0" の場合は、端子ウェイトモード 0 と同様の動作になります。

ウェイトカウントが 1 (WC1 = "0"、WC0 = "1") で、かつ $\overline{\text{WAIT}}$ 端子入力による T_w が 1 ステートの場合のタイミングを図 6.6 に示します。

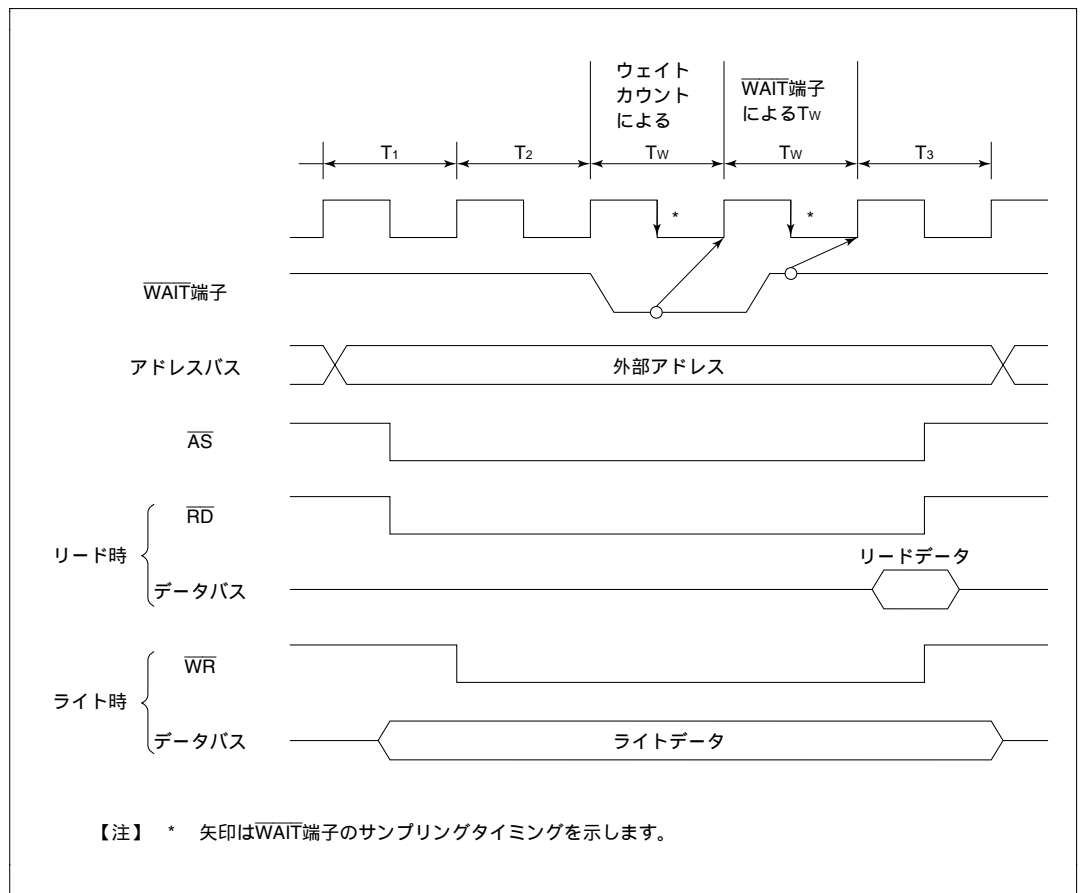


図 6.6 端子ウェイトモード 1

(3) 端子オートウェイトモード

端子オートウェイトモードでは、 $\overline{\text{WAIT}}$ 端子が"Low"レベルのとき、WC1、0ビットで設定された T_w 数が挿入されます。

端子オートウェイトモードでは、 T_2 ステートの 立下がりのタイミングで $\overline{\text{WAIT}}$ 端子が"Low"レベルであればWC1、0ビットによって設定された数だけ T_w を挿入します。

$\overline{\text{WAIT}}$ 端子を"Low"レベルに保持しても、設定された数を超える T_w は挿入されません。このタイミングを図6.7に示します。図6.7は、ウェイトカウンタが1の場合です。

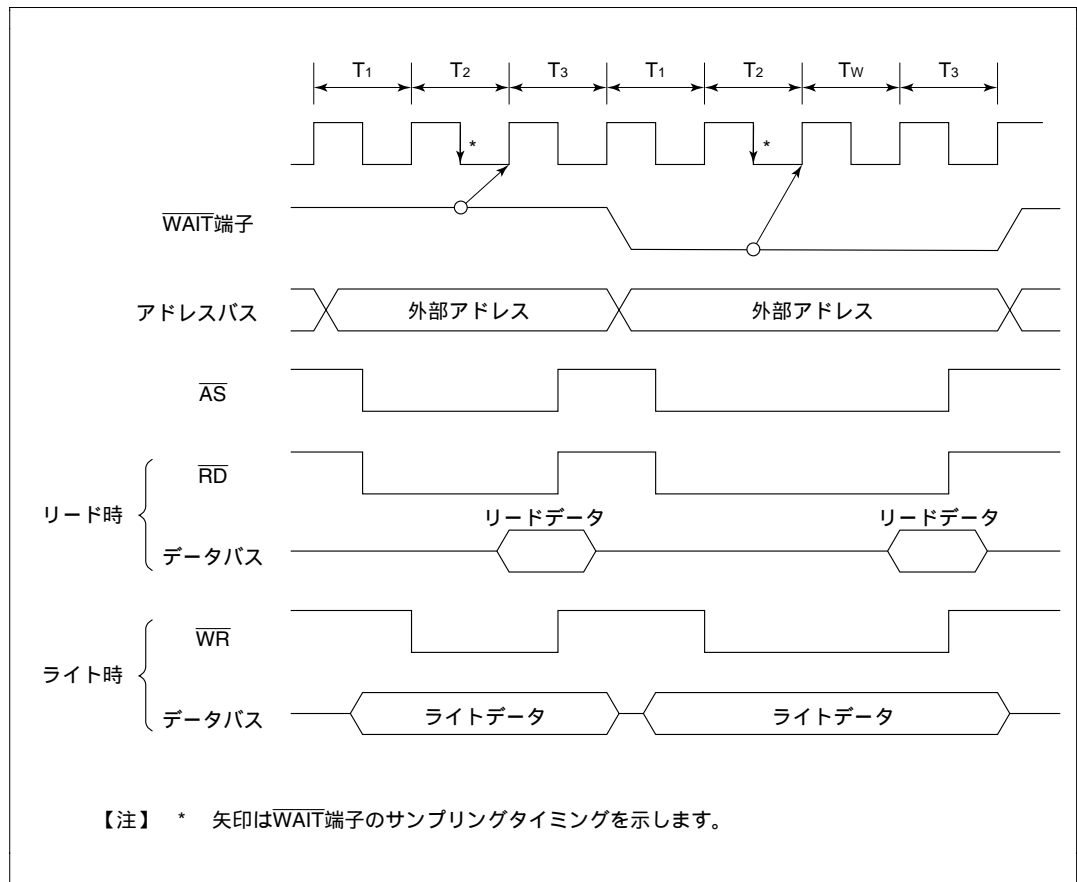


図6.7 端子オートウェイトモード

(4) プログラマブルウェイトモード

プログラマブルウェイトモードでは、外部3ステートアクセス空間をアクセスすると、常に WC1、0 ビットにより設定されたステート数だけ T_w が挿入されます。

このタイミングを図 6.8 に示します。図 6.8 は、ウェイトカウントが 1 の場合 (WC1 = "0"、WC0 = "1") です。

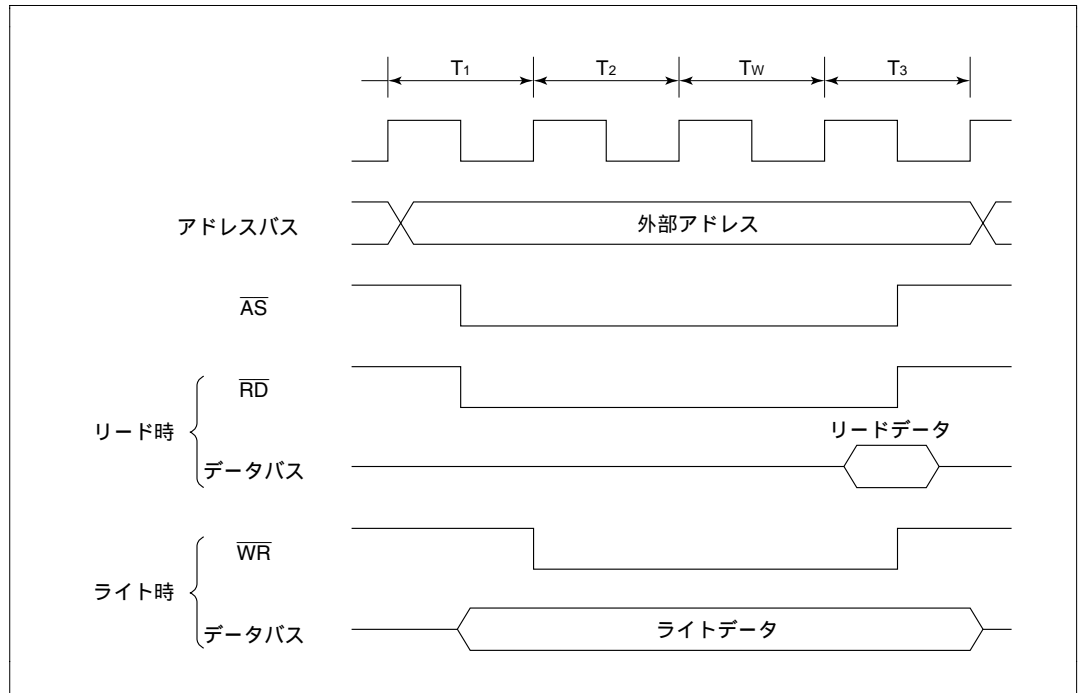


図 6.8 プログラマブルウェイトモード

(5) WSC の設定例

リセット後の WCER、ASTCR は、いずれも H'FF、WCR は H'F3 となっています。このため全エリアともプログラマブルウェイトモードの3ステート挿入となります。

その後、ソフトウェアにより、ASTCR、WCER、WCR を設定し、エリアごとのウェイトモードを選択してください。

ウェイトモードの設定例を図6.9に示します。

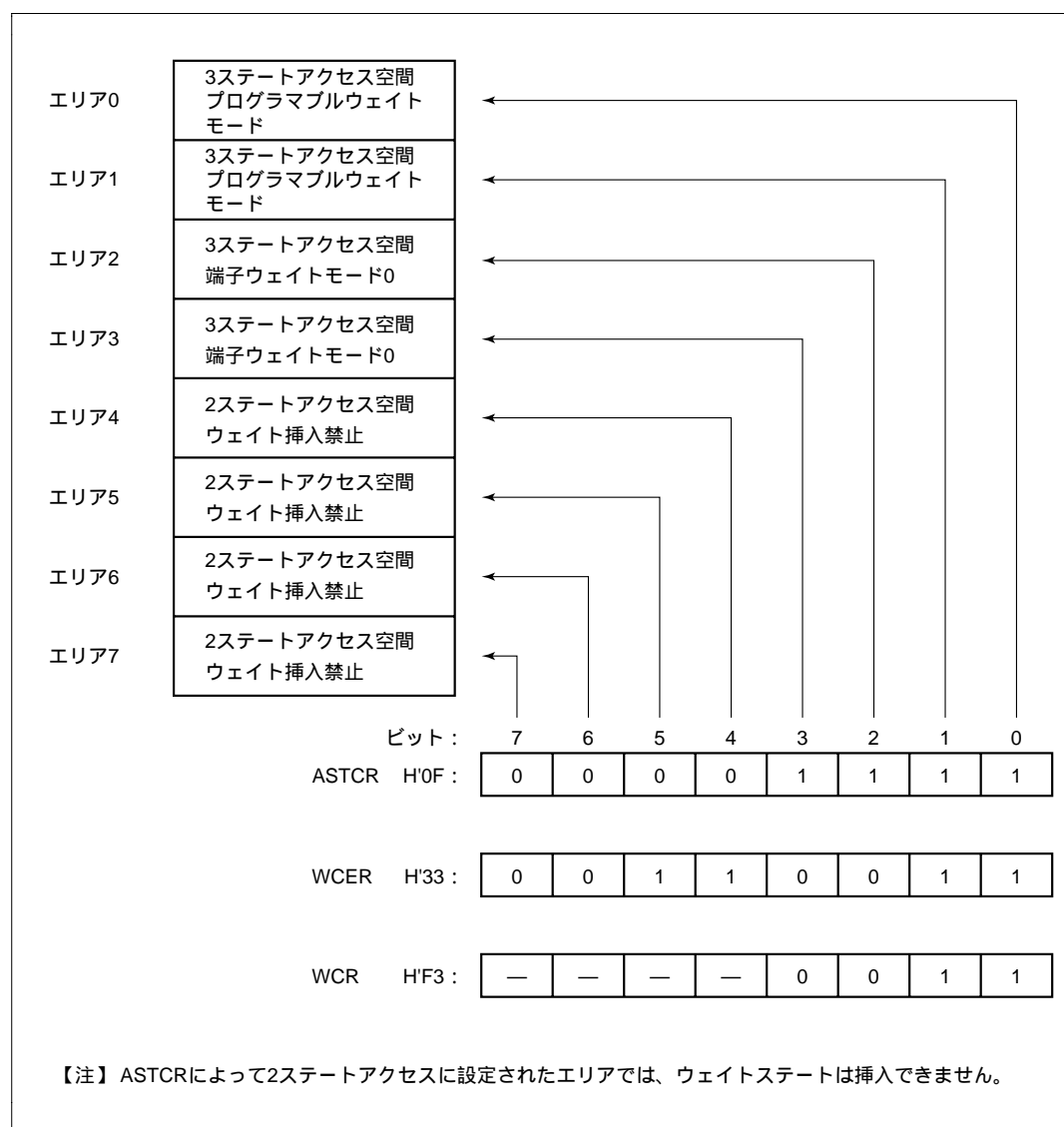


図6.9 ウェイトモードの設定例

6.3.4 メモリとの接続例

バスコントローラは、各エリアごとに、アクセスステート数を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することができます。3ステートアクセス空間に対しては多種のウェイト挿入が可能ですので高速性が要求されるデバイスと低速デバイスを容易に接続することが可能です。メモリマップ例を図6.10に示します。

エリア2に32kワード×8ビットのEPROMを接続し、8ビット3ステートアクセスを行います。

エリア3に32kワード×8ビットのSRAMを2個（SRAM1、2）接続し、8ビット2ステートアクセスを行います。

エリア7に32kワード×8ビットのSRAMを1個（SRAM3）接続し、8ビット3ステートアクセス・端子オートウェイトステートを行います。

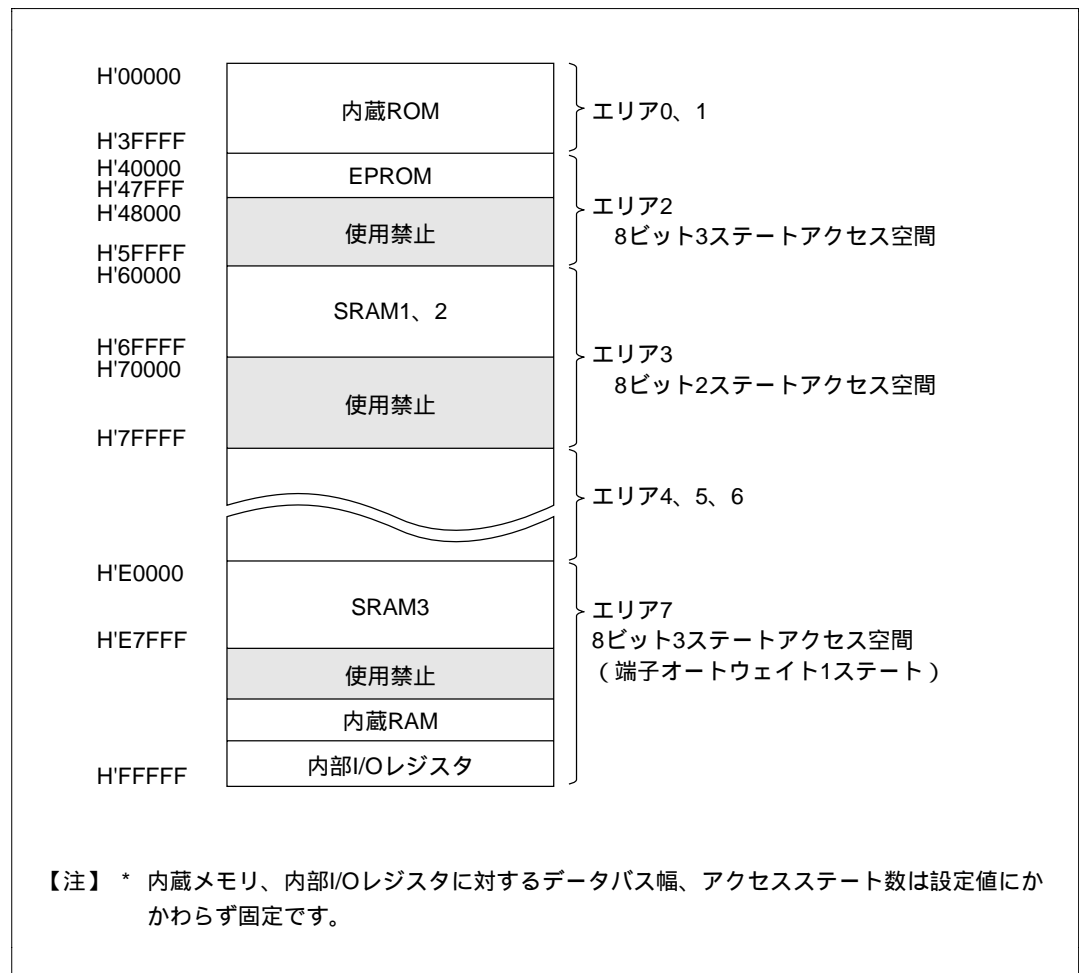


図 6.10 メモリマップ例 (H8/3035 1Mバイトモード)

6.4 使用上の注意

6.4.1 レジスタライトタイミング

(1) ASTCR および WCER のライトタイミング

ASTCR および WCER 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になります。

このタイミングを図 6.11 に示します。

エリア 2 上の命令でエリア 2 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。

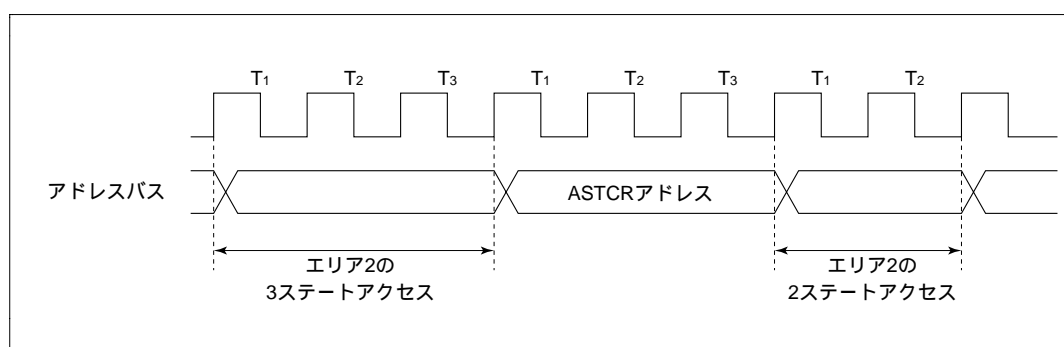


図 6.11 ASTCR ライトタイミング

6.4.2 ASTCR と ABWCR*設定上の注意事項

H8/3035 と H8/3034 の内蔵 ROM アクセスサイクルを、サポートツール用評価チップでエミュレートするために、H8/3035 と H8/3034 の内蔵プログラムにて ASTCR と ABWCR を下記のように設定してください。

(1) モード 1、3 の場合

```
ASTCR0=ASTCR1="0"
```

```
ABWCR=H'FC
```

(2) モード 2 の場合

```
ASTCR0="0"
```

```
ABWCR=H'FE
```

【注】 * ABWCR (バス幅コントロールレジスタ、下位 16 ビットアドレス H'FFEC) は本 LSI に内蔵されていません。詳細機能に関しては「H8/3048 シリーズ、H8/3048 F-ZTAT™ ハードウェアマニュアル」を参照してください。

7. I/O ポート

第7章 目次

7.1	概要	133
7.2	ポート 1	135
	7.2.1 概要	135
	7.2.2 レジスタの構成と説明	135
	7.2.3 モード別端子機能	137
7.3	ポート 2	139
	7.3.1 概要	139
	7.3.2 レジスタの構成と説明	139
	7.3.3 モード別端子機能	139
	7.3.4 入力プルアップ MOS	143
7.4	ポート 3	144
	7.4.1 概要	144
	7.4.2 レジスタの構成と説明	144
	7.4.3 モード別端子機能	146
7.5	ポート 5	148
	7.5.1 概要	148
	7.5.2 レジスタの構成と説明	148
	7.5.3 モード別端子機能	151
	7.5.4 入力プルアップ MOS	152
7.6	ポート 6	153
	7.6.1 概要	153
	7.6.2 レジスタの構成と説明	153
	7.6.3 モード別端子機能	156
7.7	ポート 7	159
	7.7.1 概要	159
	7.7.2 レジスタの構成と説明	159
7.8	ポート 8	160

7. I/Oポート

	7.8.1	概要	160
	7.8.2	レジスタの構成と説明.....	160
	7.8.3	モード別端子機能	163
7.9	ポート9.....		164
	7.9.1	概要	164
	7.9.2	レジスタの構成と説明.....	164
	7.9.3	端子機能.....	166
7.10	ポートA.....		167
	7.10.1	概要	167
	7.10.2	レジスタの構成と説明.....	168
	7.10.3	端子機能.....	169
7.11	ポートB.....		174
	7.11.1	概要	174
	7.11.2	レジスタの構成と説明.....	174
	7.11.3	端子機能.....	176

7.1 概要

本LSIには、9本の入出力ポート（ポート1、2、3、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7.1に示します。表7.1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、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1～3、5、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ます。ポート9、A、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1～3、5、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、BはLEDを駆動（シンク電流10mA）することができます。また、ポートP8₂～P8₀、PA₇～PA₀、およびPB₃～PB₀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 7.1 ポートの機能一覧

ポート	概要	端子	モード1	モード2	モード3
ポート1	・ 8ビットの入出力ポート ・ LED 駆動可能	P1 ₇ ~P1 ₀ /A ₇ ~A ₀	アドレス出力端子 (A ₇ ~A ₀) 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2	・ 8ビットの入出力ポート ・ 入力プルアップMOS内蔵 ・ LED 駆動可能	P2 ₇ ~P2 ₀ /A ₁₅ ~A ₈	アドレス出力端子 (A ₁₅ ~A ₈) 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	・ 8ビットの入出力ポート	P3 ₇ ~P3 ₀ /D ₇ ~D ₀	データ入出力端子 (D ₇ ~D ₀)		入出力ポート
ポート5	・ 4ビットの入出力ポート ・ 入力プルアップMOS内蔵 ・ LED 駆動可能	P5 ₃ ~P5 ₀ /A ₁₉ ~A ₁₆	アドレス出力端子 (A ₁₉ ~A ₁₆) 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6	・ 4ビットの入出力ポート	P6 ₉ /WR P6 ₄ /RD P6 ₃ /AS	バス制御信号出力端子 (WR、RD、AS)		入出力ポート
		P6 ₀ /WAIT	バス制御信号出力端子 (WAIT) と1ビットの入出力ポートの兼用		
ポート7	・ 8ビットの入力専用ポート	P7 ₇ ~P7 ₀ /AN ₇ ~AN ₀	A/D変換器のアナログ入力端子 (AN ₇ ~AN ₀) と入力専用ポートとの兼用		
ポート8	・ 4ビットの入出力ポート ・ P8 ₂ ~P8 ₀ はシュミット入力	P8 ₃ /IRQ ₃ P8 ₂ /IRQ ₂ P8 ₁ /IRQ ₁ P8 ₀ /IRQ ₀	IRQ ₃ ~IRQ ₀ 入力端子と4ビットの入出力ポートの兼用		
ポート9	・ 3ビットの入出力ポート	P9 ₄ /SCK/IRQ ₄	シリアルコミュニケーションインタフェース (SCI) の入出力端子 (SCK)、IRQ ₄ 入力端子と1ビットの入出力ポートの兼用		
		P9 ₂ /RxD P9 ₁ /TxD	(SCI) の入出力端子 (TxD、RxD) と2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 ₇ /TP ₇ /TIOCB ₂ /A ₂₀	プログラマブルタイミングバターンコントローラ (TPC) 出力端子 (TP ₇)、16ビットインテグレートドタイムユニット (ITU) の入出力端子 (TIOCB ₂) と入出力ポートの兼用	アドレス出力端子 (A ₂₀)	TPC出力端子 (TP ₇)、ITUの入出力端子 (TIOCB ₂) と入出力ポートの兼用
		PA ₈ /TP ₆ /TIOCA ₂₁ PA ₉ /TP ₉ /TIOCB ₁ /A ₂₂ PA ₄ /TP ₄ /TIOCA ₁ /A ₂₃	TPC出力端子 (TP ₆ ~TP ₄)、ITUの入出力端子 (TIOCA ₂₁)、TIOCB ₁ 、TIOCA ₁) と入出力ポートの兼用	TPC出力端子 (TP ₆ ~TP ₄)、ITUの入出力端子 (TIOCA ₂₁ 、TIOCB ₁ 、TIOCA ₁) 出力端子 (A ₂₃ ~A ₂₁) と入出力ポートの兼用	TPC出力端子 (TP ₆ ~TP ₄)、ITUの入出力端子 (TIOCA ₂₁ 、TIOCB ₁ 、TIOCA ₁) と入出力ポートの兼用
		PA ₃ /TP ₂ /TIOCB ₀ /TCLKD PA ₂ /TP ₂ /TIOCA ₀ /TCLKC PA ₁ /TP ₁ /TCLKB PA ₀ /TP ₀ /TCLKA	TPC出力端子 (TP ₃ ~TP ₀)、ITUの入出力端子 (TCLKD、TCLKC、TCLKB、TCLKA、TIOCB ₀ 、TIOCA ₀) と入出力ポートの兼用		
ポートB	・ 8ビットの入出力ポート ・ LED 駆動可能 ・ PB ₃ ~PB ₀ はシュミット入力	PB ₇ /TP ₁₅ /ADTRG	TPC の出力端子 (TP ₁₅)、A/D変換器のトリガ入力端子 (ADTRG) と1ビットの入出力ポートの兼用		
		PB ₆ /TP ₁₄	TPC の出力端子 (TP ₁₄) と1ビットの入出力ポートの兼用		
		PB ₉ /TP ₁₃ /TOCXB ₄ PB ₇ /TP ₁₂ /TOCXA ₄ PB ₃ /TP ₁₁ /TIOCB ₄ PB ₂ /TP ₁₀ /TIOCA ₄ PB ₁ /TP ₉ /TIOCB ₃ PB ₀ /TP ₈ /TIOCA ₃	TPC の出力端子 (TP ₈ ~TP ₁₃) ITUの入出力端子 (TIOCA ₃ ~TOCXB ₄) と入出力ポートの兼用		

7.2 ポート 1

7.2.1 概要

ポート 1 は、8 ビットの入出力ポートです。ポート 1 の各端子は、図 7.1 に示す構成となっており、モードにより端子機能が異なります。

モード 1、2 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 (A₇~A₀) 出力端子、または入力ポートとなります。モード 3 のときは、入出力ポートとなります。

また、ポート 1、2 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、ダーリントントランジスタを駆動することもできます。

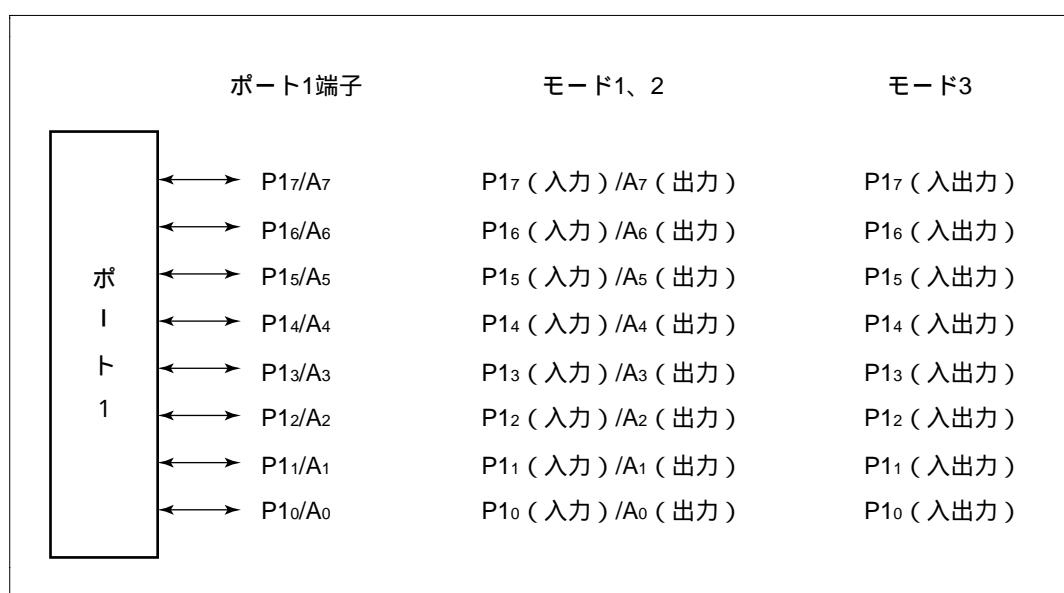


図 7.1 ポート 1 の端子構成

7.2.2 レジスタの構成と説明

表 7.2 にポート 1 のレジスタ構成を示します。

表 7.2 ポート 1 レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFC0	ポート 1 データディレクションレジスタ	P1DDR	W	H'00
H'FFC2	ポート 1 データレジスタ	P1DR	R/W	H'00

【注】 * アドレスの下位 16 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
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、2

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。

(b) モード3 (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の端子は出力端子となり、"0"にクリアすると入力端子になります。

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、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の各端子 P1₇~P1₀のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	P1 ₇	P1 ₆	P1 ₅	P1 ₄	P1 ₃	P1 ₂	P1 ₁	P1 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

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。

P1DDR が"1"のとき、ポート1のリードを行うと、P1DRの値を直接リードします。そのため端子の状態の影響を受けません。P1DDR が"0"のとき、ポート1のリードを行うと、端子の状態が読み出されます。

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。

7.2.3 モード別端子機能

ポート1は、モード1、2とモード3(シングルチップモード)では以下のように端子機能が異なります。

(1) モード1、2の端子機能

ポート1の各端子は、アドレス出力端子と入力ポートとの兼用になります。

P1DDRの各ビットを"1"にセットすると、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、入力端子になります。リセット直後は入力ポートになっています。したがって、アドレスバスとして使用する場合は、P1DDRを"1"にセットしてください。

モード1、2時の端子機能を図7.2に示します。

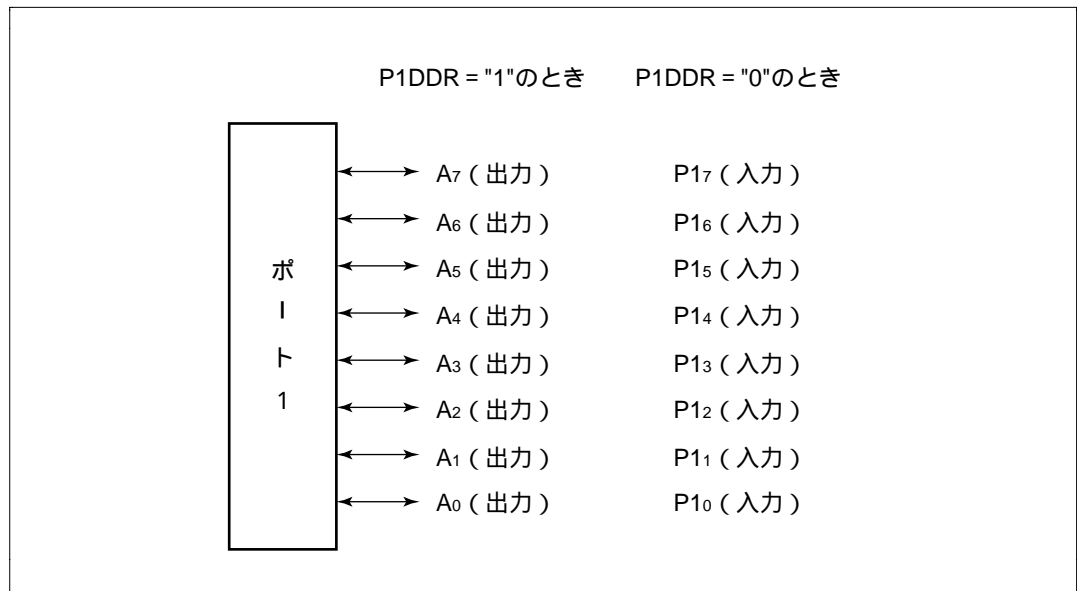


図 7.2 モード 1、2 時の端子機能 (ポート 1)

(2) モード 3 の端子機能

ポート 1 の各端子はビット単位で入出力指定可能です。

P1DDR の各ビットを"1"にセットすると対応するポート 1 の端子は出力端子となり、"0"にクリアすると入力端子となります。

モード 3 時の端子機能を図 7.3 に示します。

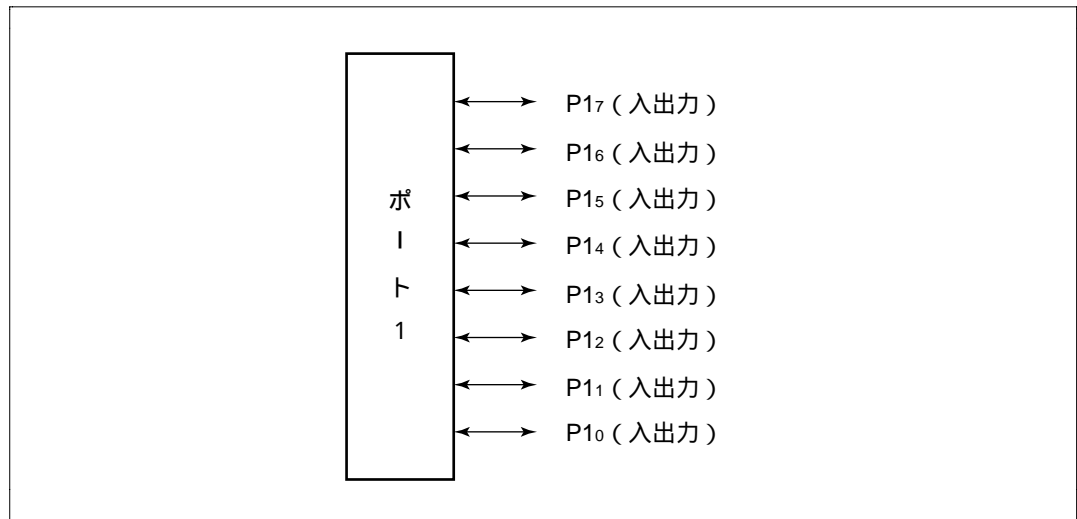


図 7.3 モード 3 時の端子機能 (ポート 1)

7.3 ポート 2

7.3.1 概要

ポート 2 は、8 ビットの入出力ポートです。ポート 2 の各端子は、図 7.4 に示す構成となっており、モードにより端子機能が異なります。

モード 1、2 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 (A₁₅ ~ A₈) または入力ポートとなります。

モード 3 のときは、入出力ポートとなります。

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、ダーリントントランジスタを駆動することができます。

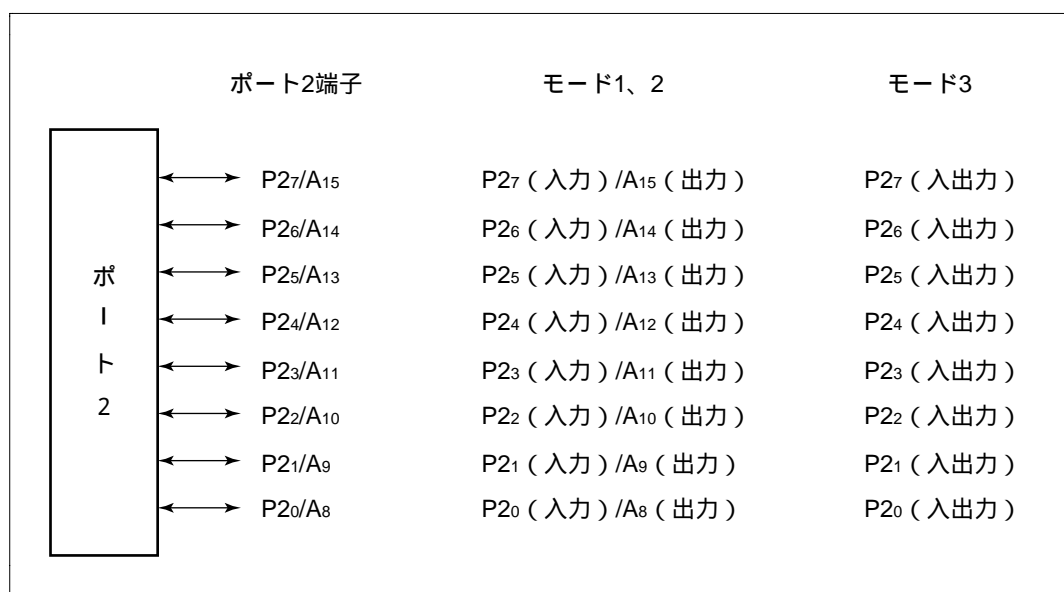


図 7.4 ポート 2 の端子構成

7.3.2 レジスタの構成と説明

表 7.3 にポート 2 のレジスタ構成を示します。

表 7.3 ポート 2 レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFC1	ポート 2 データディレクションレジスタ	P2DDR	W	H'00
H'FFC3	ポート 2 データレジスタ	P2DR	R/W	H'00
H'FFD8	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】 * アドレスの下位 16 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
	P2 ₇ DDR	P2 ₆ DDR	P2 ₅ DDR	P2 ₄ DDR	P2 ₃ DDR	P2 ₂ DDR	P2 ₁ DDR	P2 ₀ DDR
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、2

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3

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の端子は出力端子となり、"0"にクリアすると入力端子となります。

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、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の各端子 P2₇ ~ P2₀ のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	P2 ₇	P2 ₆	P2 ₅	P2 ₄	P2 ₃	P2 ₂	P2 ₁	P2 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

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。

P2DDR が"1"のとき、ポート2のリードを行うと、P2DRの値を直接リードします。そのため端子の状態の影響を受けません。P2DDR が"0"のときポート2のリードを行うと、端子の状態が読み出されます。

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
	P2 ₇ PCR	P2 ₆ PCR	P2 ₅ PCR	P2 ₄ PCR	P2 ₃ PCR	P2 ₂ PCR	P2 ₁ PCR	P2 ₀ PCR
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。

P2DDR を"0"にクリアした(入力ポートの)状態でP2₇PCR ~ P2₀PCR ビットを"1"にセットすると入力プルアップ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。

7.3.3 モード別端子機能

ポート2は、モード1、2とモード3とは、端子機能が異なります。

以下モード別に説明します。

(1) モード1、2の端子機能

ポート2の各端子は、アドレス出力端子と入力ポートとの兼用になります。ビット単位で入出力を指定可能です。P2DDRの各ビットを"1"に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端子となります。

リセット直後は入力ポートになっています。したがって、アドレスバスとして使用する場合はP2DDRを"1"にセットしてください。

モード1、2の端子機能を図7.5に示します。

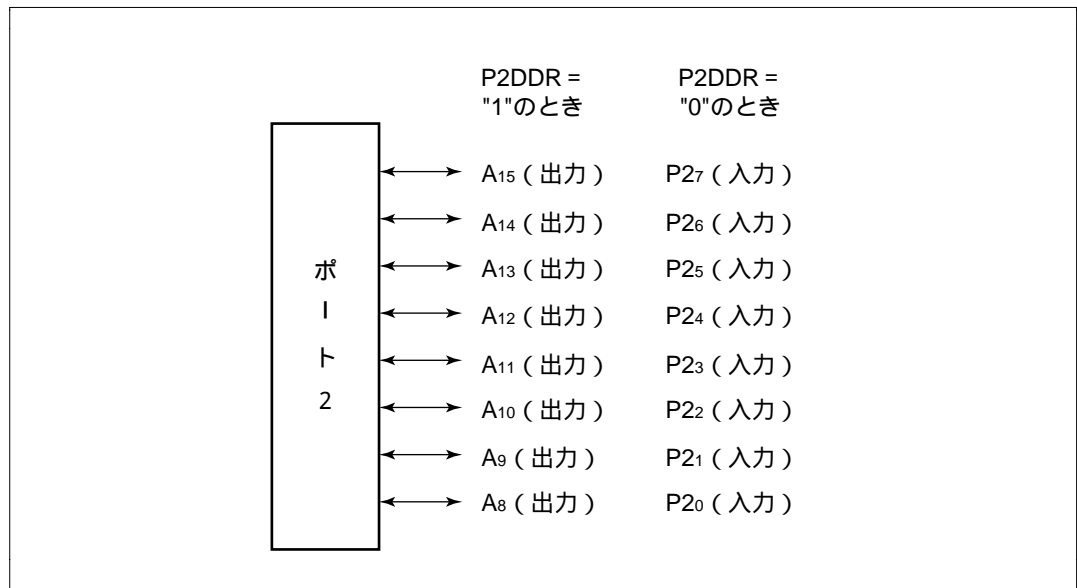


図7.5 モード1、2時の端子機能（ポート2）

(2) モード3の端子機能

ポート2の各端子は、ビット単位で入出力を指定可能です。

P2DDRの各ビットを"1"にセットすると対応するポート2の端子は出力端子となり、"0"にクリアすると入力端子となります。

モード3の端子機能を図7.6に示します。

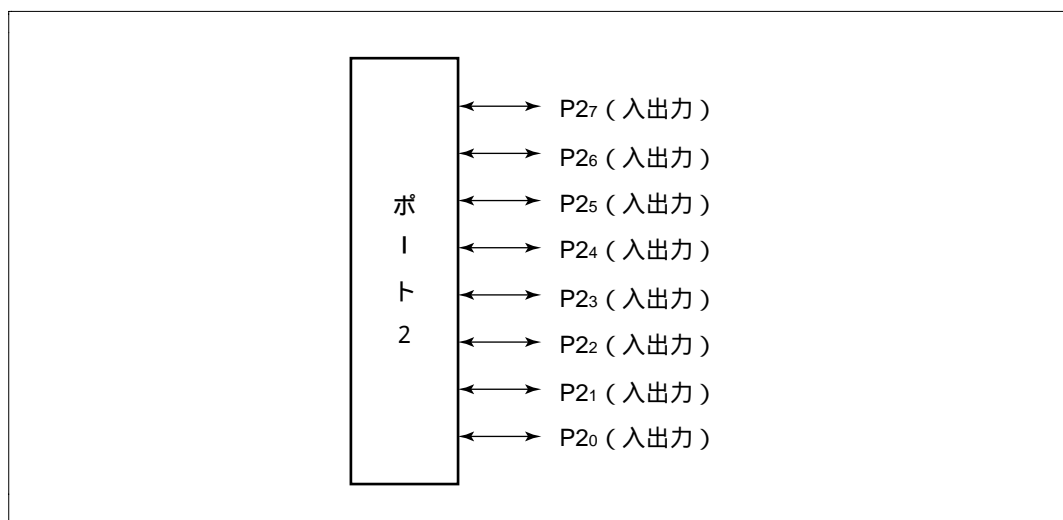


図 7.6 モード3時の端子機能 (ポート2)

7.3.4 入力プルアップMOS

ポート2は、プログラムで制御可能な入力プルアップMOSを内蔵しています。この入力プルアップMOSは、ビット単位でON/OFFを指定できます。

P2PCRを"1"にセットし、かつP2DDRを"0"にクリアすると、入力プルアップMOSはONとなります。

入力プルアップMOSは、リセット、またはハードウェアスタンバイモード時にはOFFします。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7.4に示します。

表 7.4 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	OFF		ON/OFF	
2	OFF		ON/OFF	
3	OFF		ON/OFF	

《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。

7.4 ポート3

7.4.1 概要

ポート3は、8ビットの入出力ポートです。ポート3の各端子は、図7.7に示す構成となっており、モードにより機能が異なります。モード1、2のときはデータバスとなり、モード3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、ダライントトランジスタを駆動することもできます。

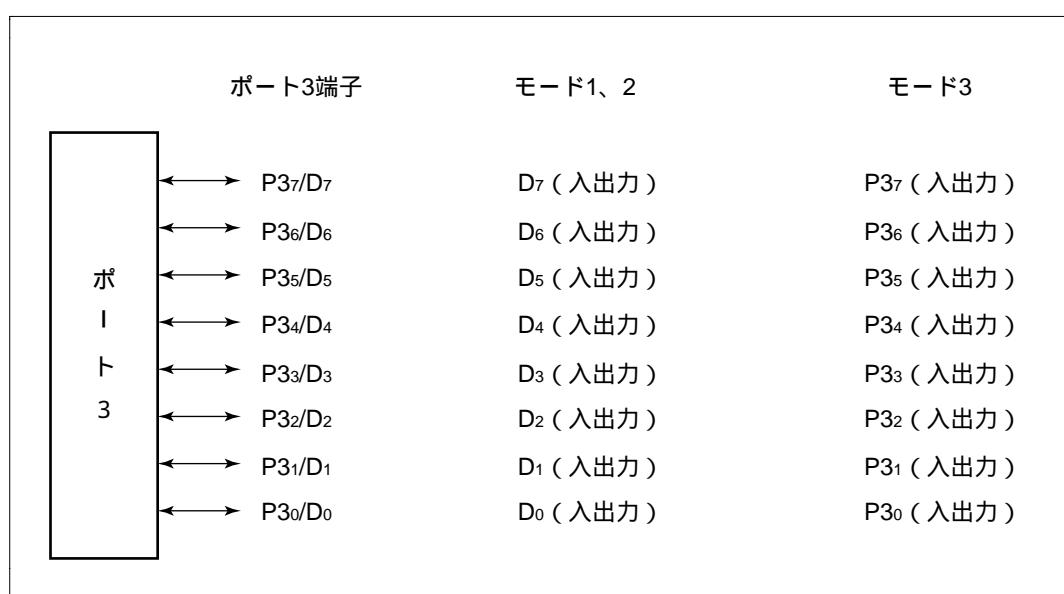


図7.7 ポート3の端子構成

7.4.2 レジスタの構成と説明

表7.5にポート3のレジスタ構成を示します。

表7.5 ポート3レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFC4	ポート3データディレクションレジスタ	P3DDR	W	H'00
H'FFC6	ポート3データレジスタ	P3DR	R/W	H'00

【注】 * アドレスの下位16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
	P3 ₇ DDR	P3 ₆ DDR	P3 ₅ DDR	P3 ₄ DDR	P3 ₃ DDR	P3 ₂ DDR	P3 ₁ DDR	P3 ₀ DDR
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、2

ポート3はデータバスとして機能します。P3DDRは無効です。

(b) モード3

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の端子は出力端子となり、"0"にクリアすると入力端子になります。

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、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の各端子P3₇~P3₀のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	P3 ₇	P3 ₆	P3 ₅	P3 ₄	P3 ₃	P3 ₂	P3 ₁	P3 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

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。

P3DDRが"1"のとき、ポート3のリードを行うと、P3DRの値を直接リードします。そのため端子の状態の影響を受けません。

P3DDRが"0"のときポート3のリードを行うと、端子の状態が読み出されます。

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。

7.4.3 モード別端子機能

ポート3は、モード1、2とモード3では、以下のように端子機能が異なります。

(1) モード1、2の端子機能

モード1、2時には、ポート3は自動的にデータ入出力端子になります。P3DDRによる入出力の方向は無視されます。モード1、2時の端子機能を図7.8に示します。

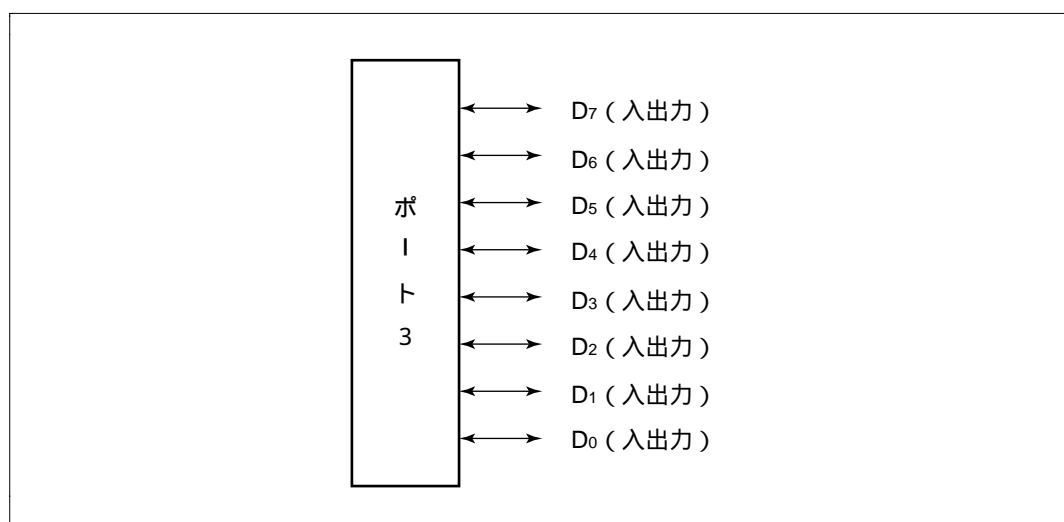


図7.8 モード1、2時の端子機能（ポート3）

(2) モード3時の端子機能

ポート3の各端子は、ビット単位で入出力を指定可能です。

P3DDRの各ビットを"1"にセットすると対応するポート3の端子は出力端子となり、"0"にクリアすると入力端子となります。

モード3の端子機能を図7.9に示します。

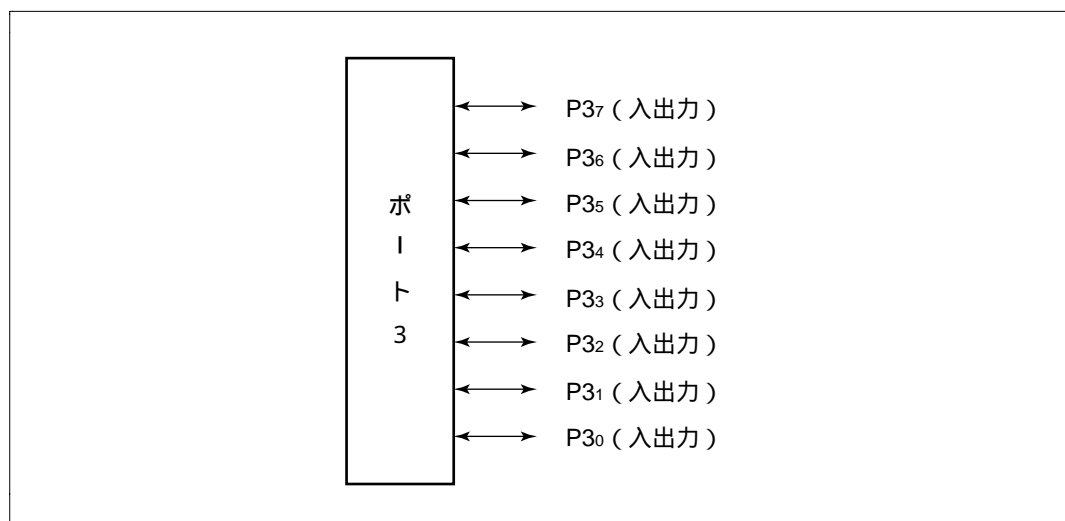


図 7.9 モード 3 時の端子機能 (ポート 3)

7.5 ポート5

7.5.1 概要

ポート5は、4ビットの入出力ポートです。ポート5の各端子は、図7.10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、2のときは、ポート5の各端子はポート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(A₁₉~A₁₆)出力端子または入力ポートとなります。

モード3のときは、入出力ポートとなります。

ポート5は、プログラムで制限可能なプルアップMOSが内蔵されています。また、1個のTTL負荷と90pFの容量負荷を駆動することができます。また、ダーリントトランジスタを駆動することも可能です。

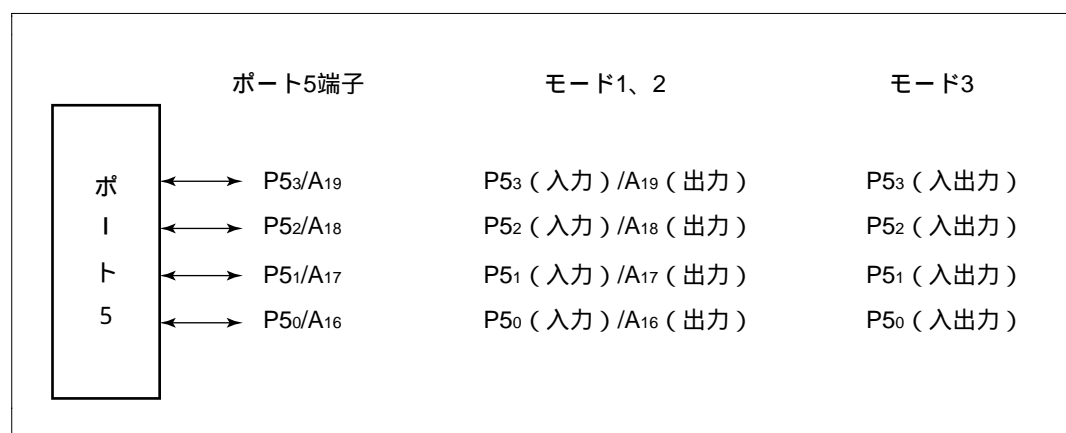


図7.10 ポート5の端子構成

7.5.2 レジスタの構成と説明

表7.6にポート5のレジスタ構成を示します。

表7.6 ポート5レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFC8	ポート5データディレクションレジスタ	P5DDR	W	H'F0
H'FFCA	ポート5データレジスタ	P5DR	R/W	H'F0
H'FFDB	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】 * アドレスの下位16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	6	5	4	3	2	1	0
	—	—	—	—	P5 ₃ DDR	P5 ₂ DDR	P5 ₁ DDR	P5 ₀ DDR
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、2

P5DDRを"1"にセットすると対応するポート5の端子はアドレス出力となり、"0"にクリアすると入力になります。

(b) モード3

P5DDRを"1"にセットすると対応するポート5の端子は出力端子となり、"0"にクリアすると入力になります。

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、HF0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、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の各端子 P5₃ ~ P5₀ のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	P5 ₃	P5 ₂	P5 ₁	P5 ₀
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。

P5DDR が"1"のとき、ポート5のリードを行うと、P5DRの値を直接リードします。そのため端子の状態の影響を受けません。P5DDR が"0"のときポート5のリードを行うと、端子の状態が読み出されます。

P5₇ ~ P5₄ビットは、リザーブビットです。リード/ライト可能ですが、ポートとしては使用できません。

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、HF0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	6	5	4	3	2	1	0
	—	—	—	—	P5 ₃ PCR	P5 ₂ PCR	P5 ₁ PCR	P5 ₀ 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。

P5DDR を"0"にクリアした(入力ポート)状態で P5₃PCR ~ P5₀PCR ビットを"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、HF0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。

7.5.3 モード別端子機能

ポート5は、モード1、2とモード3とは、端子機能が異なります。

以下モード別に説明します。

(1) モード1、2の端子機能

ポート5の各端子は、アドレス出力端子と入力ポートとの兼用になります。P5DDRの各ビットを"1"にセットすると対応するポート5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端子となります。

ポート5の各端子は、リセット直後は入力端子になっています。したがって、アドレス出力として使用する場合は、P5DDRを"1"にセットしてください。

モード5の端子機能を図7.11に示します。

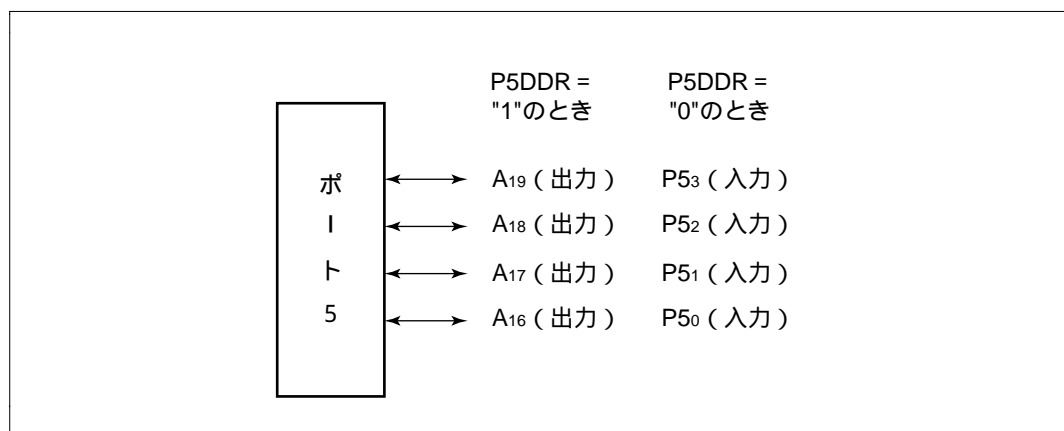


図7.11 モード1、2時の端子機能（ポート5）

(2) モード3の端子機能

ポート5の各端子は、ビット単位で入出力を指定可能です。P5DDRの各ビットを"1"にセットすると対応するポート5の端子は出力端子となり、"0"にクリアすると入力端子となります。

モード3の端子機能を図7.12に示します。

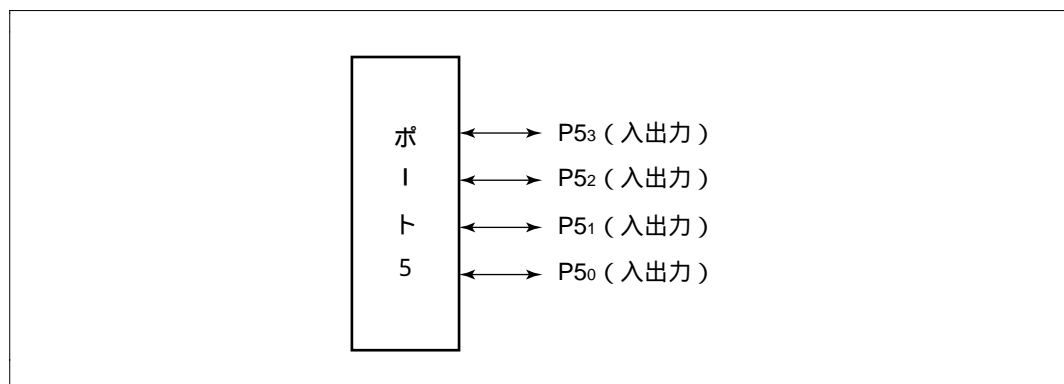


図7.12 モード3時の端子機能（ポート5）

7.5.4 入力プルアップ MOS

ポート 5 は、プログラムで制御可能な入力プルアップ MOS を内蔵しています。この入力プルアップ MOS は、ビット単位で ON/OFF を指定できます。

P5PCR を "1" にセットし、かつ P5DDR を "0" にクリアすると、入力プルアップ MOS は ON となります。

入力プルアップ MOS は、リセット、またはハードウェアスタンバイモード時には OFF します。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 7.7 に示します。

表 7.7 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		ON/OFF	
2				
3				

《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。

7.6 ポート6

7.6.1 概要

ポート6は、4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{WR} 、 \overline{RD} 、 \overline{AS} 、 \overline{WAIT}) と兼用になっています。

ポート6の端子構成を図7.13に示します。

モード1、2時には、 \overline{WR} 、 \overline{RD} 、 \overline{AS} 、 $P6_0/\overline{WAIT}$ として機能します。モード3時には、入出力ポートとなります。

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、ダライントトランジスタを駆動することができます。

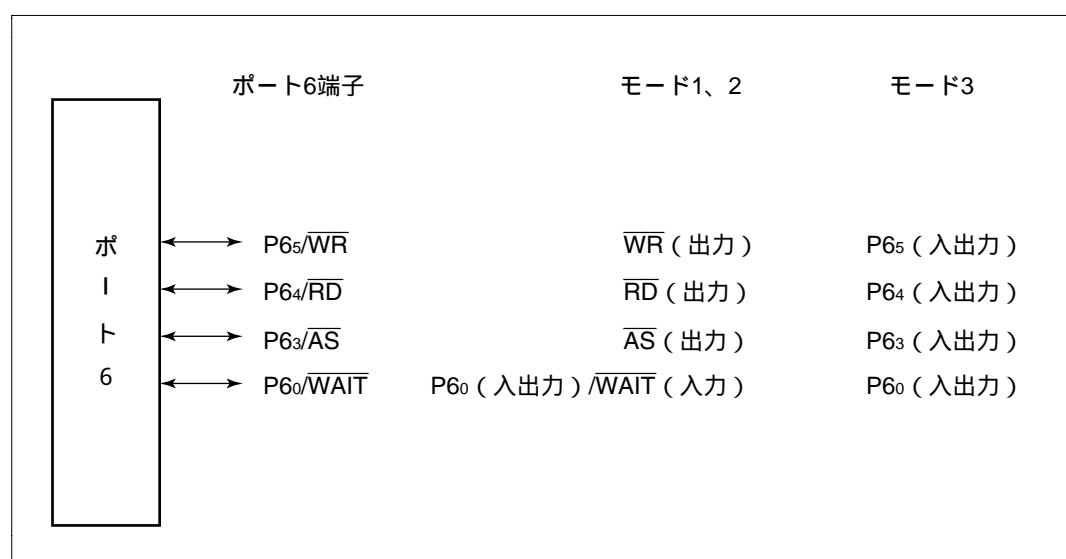


図7.13 ポート6の端子構成

7.6.2 レジスタの構成と説明

表7.8にポート6のレジスタ構成を示します。

表7.8 ポート6レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFC9	ポート6データディレクションレジスタ	P6DDR	W	H'80
H'FFCB	ポート6データレジスタ	P6DR	R/W	H'80

【注】 * アドレスの下位16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	6	5	4	3	2	1	0
	—	—	P6 ₅ DDR	P6 ₄ DDR	P6 ₃ DDR	—	—	P6 ₀ 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5~3、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、2

ポート P6₅ ~ P6₃ はバス制御出力端子 (\overline{WR} 、 \overline{RD} 、 \overline{AS}) として機能します。P6₀ は、入出力ポートとして機能します。P6₀DDR に "1" をセットすると P6₀ の端子は出力端子となり、"0" にすると入力端子になります。ただし、P6₅DDR ~ P6₃DDR は無効です。

(b) モード3

ポート6は入出力ポートとして機能します。

P6DDR に "1" をセットすると対応するポート6の端子は出力端子となり、"0" にクリアすると入力端子になります。

ビット7、6、2、1はリザーブビットです。

P6DDRは、ライト専用で、リードは無効です。このうち、ビット7はリードすると"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、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 P6₅ ~ P6₃、P6₀のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	—	—	P6 ₅	P6 ₄	P6 ₃	—	—	P6 ₀
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
ポート6データ5~3、0
ポート6の各端子のデータを格納するビットです。

P6DDR が"1"のときポート6のリードを行うと、P6DRの値を直接リードします。P6DDR が"0"のときポート6のリードを行うと端子の状態が読み出されます。ビット7、6、2、1はリザーブビットです。このうち、ビット7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。ビット6、2、1はリード/ライト可能ですが、ポートとしては使用できません。P6DDRのビット6、2、1が"1"のときリードを行うと、P6DRの各ビットの値を直接リードします。P6DDRのビット6、2、1が"0"のときリードを行うと"1"が読みだされます。

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。

7.6.3 モード別端子機能

(1) モード 1、2 時の端子機能

ポート P6₅ ~ P6₃ はバス制御出力端子として機能します。

ポート P6₀ は、バス制御入力端子と入出力ポートとの兼用になります。P6DDR0 に "1" をセットするとポート P6₀ の端子は出力端子となり、"0" にクリアすると入力端子になります。

図 7.14 と表 7.9 にモード 1、2 時の端子機能を示します。

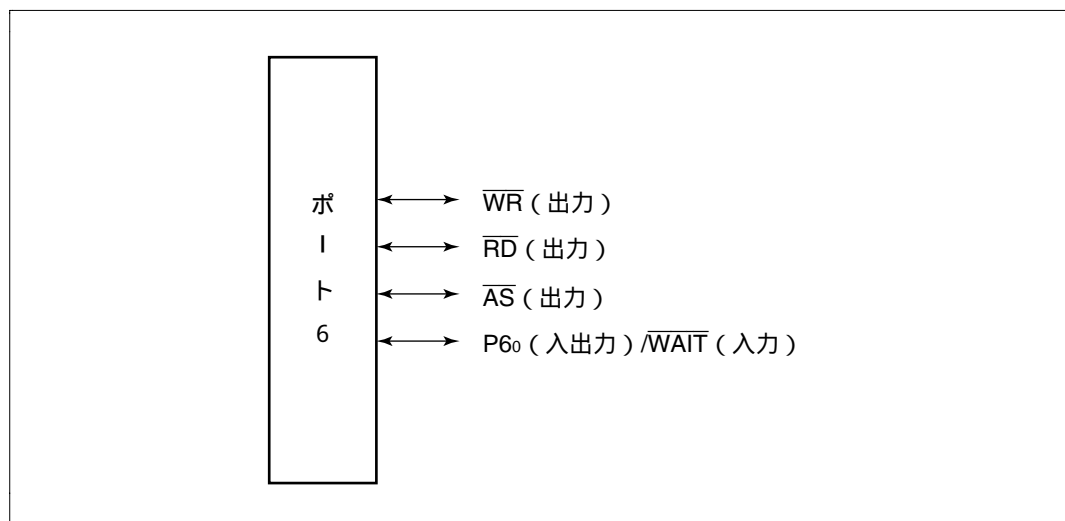


図 7.14 モード 1、2 の端子機能 (ポート 6)

表 7.9 モード 1、2 の端子機能 (ポート 6)

端子	選択方法と端子機能			
P6 ₅ / \overline{WR}	P6 ₅ DDR ビットとは無関係に、次のように切り換わります。			
	P6 ₅ DDR	0		1
	端子機能	WR出力端子		
P6 ₄ / \overline{RD}	P6 ₄ DDR ビットとは無関係に、次のように切り換わります。			
	P6 ₄ DDR	0		1
	端子機能	RD出力端子		
P6 ₃ / \overline{AS}	P6 ₃ DDR ビットとは無関係に、次のように切り換わります。			
	P6 ₃ DDR	0		1
	端子機能	\overline{AS} 出力端子		
P6 ₀ / \overline{WAIT}	WCER の WCE7 ~ WCE0、WCR の WMS1 ビットと P6 ₀ DDR ビットの組み合わせで、次のように切り換わります。			
	WCER	すべてが"1"		いずれかが"0"
	WMS1	0	1	—
	P6 ₀ DDR	0	1	0*
	端子機能	P6 ₀ 入力端子	P6 ₀ 出力端子	\overline{WAIT} 入力端子
【注】 * P6 ₀ DDRは"1"にセットしないでください。				

(2) モード3の端子機能

ポート6の各端子はビット単位で入出力を指定可能です。

P6DDRの各ビットを"1"にセットすると対応するポート6の端子は出力端子となり、"0"にクリアすると入力端子になります。

モード3時の端子機能を図7.15に示します。

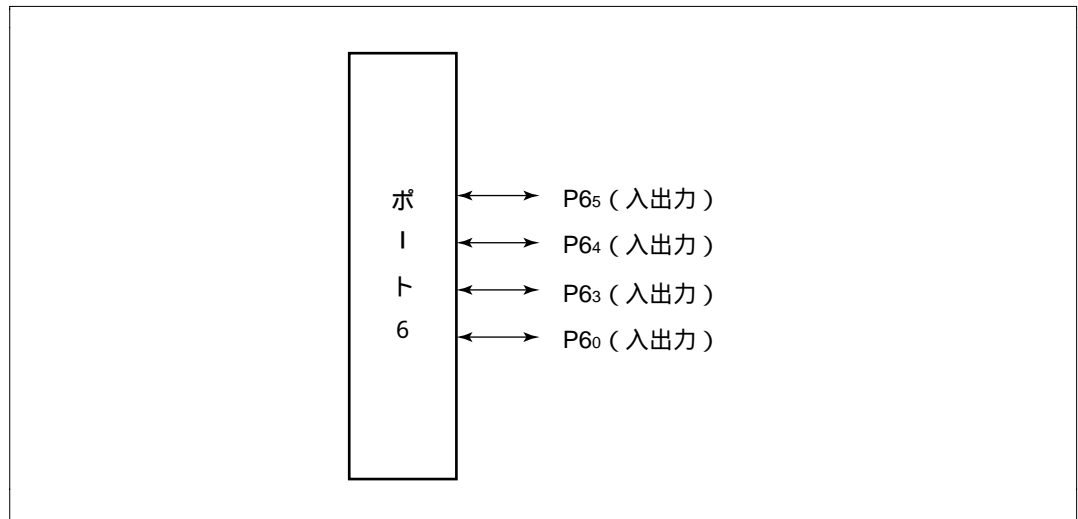


図 7.15 ポート3時の端子機能 (ポート6)

7.7 ポート7

7.7.1 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7.16に示します。

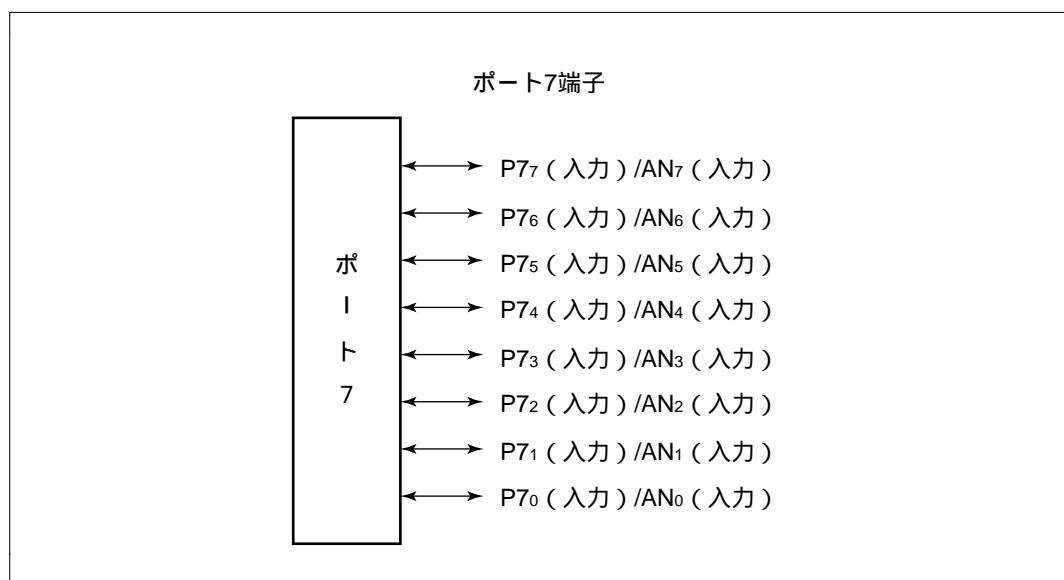


図 7.16 ポート7の端子構成

7.7.2 レジスタの構成と説明

表7.10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 7.10 ポート7レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFCE	ポート7データレジスタ	P7DR	R	不定

【注】 * アドレスの下位16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】 * P77~P70端子により決定されます。

P7DRのリードを行うと、常に端子の状態が読み出されます。

7.8 ポート 8

7.8.1 概要

ポート 8 は、4 ビットの入出力ポートです。ポート 8 は、 $\overline{IRQ}_3 \sim \overline{IRQ}_0$ 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 7.17 に示します。

ポート P8₀ は入出力ポートまたは \overline{IRQ}_0 入力端子として機能します。ポート P8₃ ~ P8₁ は、モード 1、2 時には、入力ポートまたは、 $\overline{IRQ}_3 \sim \overline{IRQ}_1$ 入力端子として機能します。モード 3 時には、入出力ポートまたは、 $\overline{IRQ}_3 \sim \overline{IRQ}_1$ 入力端子として機能します。

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。

P8₂ ~ P8₀ 端子はシュミットトリガ入力です。

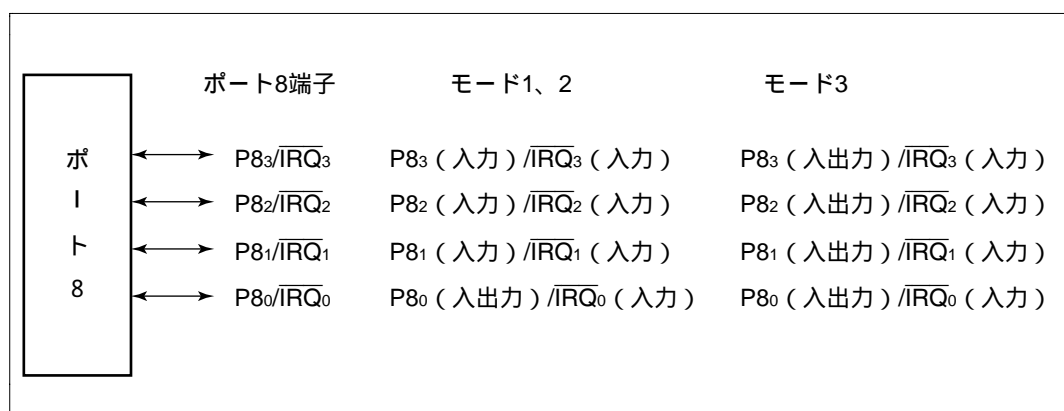


図 7.17 ポート 8 の端子構成

7.8.2 レジスタの構成と説明

表 7.11 にポート 8 のレジスタ構成を示します。

表 7.11 ポート 8 レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFCD	ポート 8 データディレクションレジスタ	P8DDR	W	H'E0
H'FFCF	ポート 8 データレジスタ	P8DR	R/W	H'E0

【注】 * アドレスの下位 16 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	6	5	4	3	2	1	0
	—	—	—	—	P8 ₃ DDR	P8 ₂ DDR	P8 ₁ DDR	P8 ₀ DDR
初期値:	1	1	1	0	0	0	0	0
R/W :	—	—	—	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3~0
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、2

P8₃~P8₁はポート入力端子となります。P8₃DDR~P8₁DDRは"1"にセットしないでください。

P8₀はP8₀DDRを"1"にセットするとポート出力端子となり、"0"にクリアすると入力端子となります。

(b) モード3

ポート8は、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ポート8の端子は出力端子となり、"0"にクリアすると入力端子となります。

ビット7~4はリザーブビットです。

P8DDRは、ライト専用レジスタで、リードは無効です。このうち、ビット7~5はリードすると"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、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。したがって、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移移すると、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 の各端子 P8₄ ~ P8₀ のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	P8 ₃	P8 ₂	P8 ₁	P8 ₀
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。

P8DDR が"1"のときポート 8 のリードを行うと、P8DR の値を直接リードします。P8DDR が"0"のときポート 8 のリードを行うと端子の状態が読み出されます。

ビット 7~4 はリザーブビットです。このうち、ビット 7~5 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。ビット 4 はリード/ライト可能ですが、ポートとしては使用できません。P8DDR のビット 4 が"1"のときリードを行うとビット 4 の値を直接リードします。P8DDR のビット 4 が"0"のときリードを行うと"1"が読み出されます。

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、HE0 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。

7.8.3 モード別端子機能

ポート8は $\overline{IRQ}_3 \sim \overline{IRQ}_0$ 入力端子と兼用になっています。

ポート8の端子機能について表7.12に示します。

表7.12 端子機能（ポート8）

端子	選択方法と端子機能			
$P8_3/\overline{IRQ}_3$	P8 ₃ DDRビットにより、次のように切り換わります。			
	P8 ₃ DDR	0	1	
			モード1、2	モード3
	端子機能	P8 ₃ 入力端子	使用禁止	P8 ₃ 出力端子
\overline{IRQ}_3 入力端子				
$P8_2/\overline{IRQ}_2$	P8 ₂ DDRビットにより、次のように切り換わります。			
	P8 ₂ DDR	0	1	
			モード1、2	モード3
	端子機能	P8 ₂ 入力端子	使用禁止	P8 ₂ 出力端子
\overline{IRQ}_2 入力端子				
$P8_1/\overline{IRQ}_1$	P8 ₁ DDRビットにより、次のように切り換わります。			
	P8 ₁ DDR	0	1	
			モード1、2	モード3
	端子機能	P8 ₁ 入力端子	使用禁止	P8 ₁ 出力端子
\overline{IRQ}_1 入力端子				
$P8_0/\overline{IRQ}_0$	P8 ₀ DDRビットにより、次のように切り換わります。			
	P8 ₀ DDR	0	1	
			P8 ₀ 出力端子	
	端子機能	P8 ₀ 入力端子		
\overline{IRQ}_0 入力端子				

7.9 ポート9

7.9.1 概要

ポート9は、3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（SCI）の入出力端子（TxD、RxD、SCK）、 \overline{IRQ}_4 入力端子と兼用になっています。

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7.18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、ダートリントランジスタを駆動することができます。

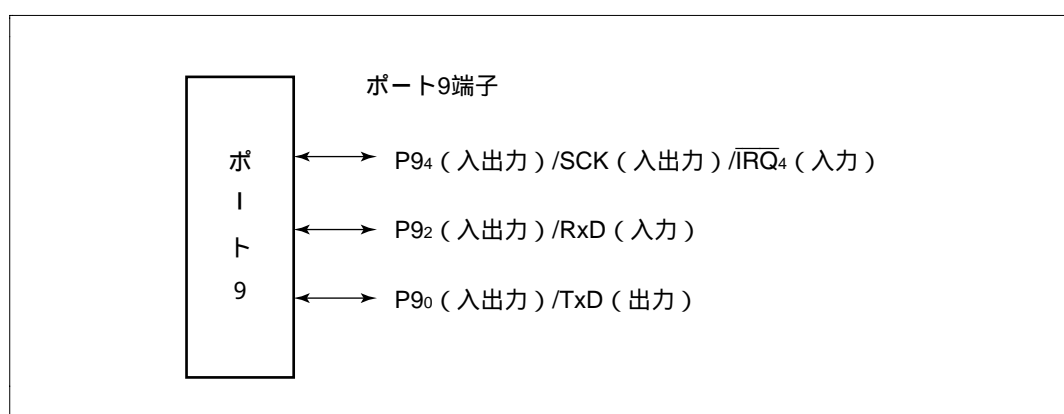


図 7.18 ポート9の端子構成

7.9.2 レジスタの構成と説明

表7.13にポート9のレジスタ構成を示します。

表 7.13 ポート9レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFD0	ポート9データディレクションレジスタ	P9DDR	W	H'C0
H'FFD2	ポート9データレジスタ	P9DR	R/W	H'C0

【注】 * アドレスの下位16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。



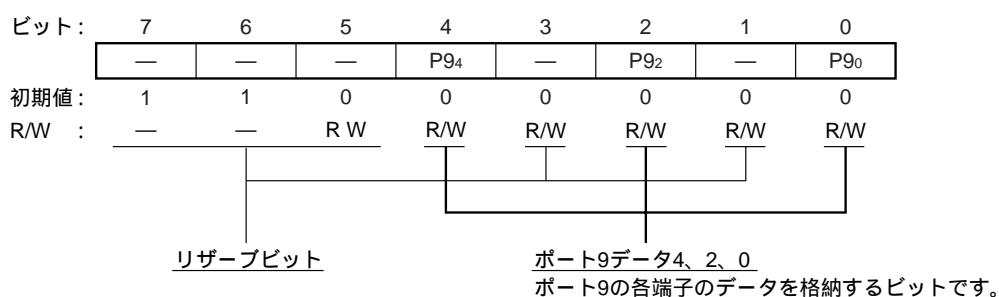
P9DDR を"1"にセットすると対応するポート9の各端子は出力となり、"0"にクリアすると入力になります。

P9DDR は、ライト専用レジスタで、リードは無効です。このうち、ビット7、6はリードすると"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、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の各端子 P9₄、P9₂、P9₀のデータを格納します。



P9DDR が"1"のとき、ポート9のリードを行うと、P9DRの値を直接リードします。

P9DDR が"0"のとき、ポート9のリードを行うと、端子の状態が読み出されます。

ビット7~5、3、1はリザーブビットです。このうち7、6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。ビット5、3、1はリード/ライト可能ですが、ポートとしては使用できません。P9DDRのビット5、3、1が"1"のときリードを行うと各ビットの値を直接リードします。P9DDRのビット5、3、1が"0"のときリードを行うと"1"が読み出されます。

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'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。

7.9.3 端子機能

ポート9の各端子はSCIの入出力端子(TxD、RxD、SCK)、および $\overline{\text{IRQ}}_4$ 入力端子と兼用になっています。ポート9の端子機能について表7.14に示します。

表7.14 ポート9の端子機能

端子	選択方法と端子機能				
P9 ₄ /SCK/ $\overline{\text{IRQ}}_4$	SCIのSMRのC/ $\overline{\text{A}}$ ビット、SCRのCKE0、1ビットとP9 ₄ 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。				
	CKE1	0			1
	C/ $\overline{\text{A}}$	0		1	—
	CKE0	0	1	—	—
	P9 ₄ DDR	0	1	—	—
端子機能	P9 ₄ 入力端子	P9 ₄ 出力端子	SCK出力端子	SCK出力端子	SCK入力端子
$\overline{\text{IRQ}}_4$ 出力端子					
P9 ₂ /RxD	SCIのSCRのREビットとP9 ₂ 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。				
	RE	0			1
	P9 ₂ DDR	0	1	—	
	端子機能	P9 ₂ 入力端子	P9 ₂ 出力端子	RxD入力端子	
P9 ₀ /TxD	SCIのSCRのTEビットとP9 ₀ 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。				
	TE	0			1
	P9 ₀ DDR	0	1	—	
	端子機能	P9 ₀ 入力端子	P9 ₀ 出力端子	TxD出力端子	

7.10 ポート A

7.10.1 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 (TP₇ ~ TP₀)、16 ビットインテグレートドタイムユニット (ITU) の入出力端子 (TIOCB₂、TIOCA₂、TIOCB₁、TIOCA₁、TIOCB₀、TIOCA₀、TCLKD、TCLKC、TCLKB、TCLKA)、およびアドレスバス (A₂₃ ~ A₂₀) と兼用になっています。ポート A の端子構成を図 7.19 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。

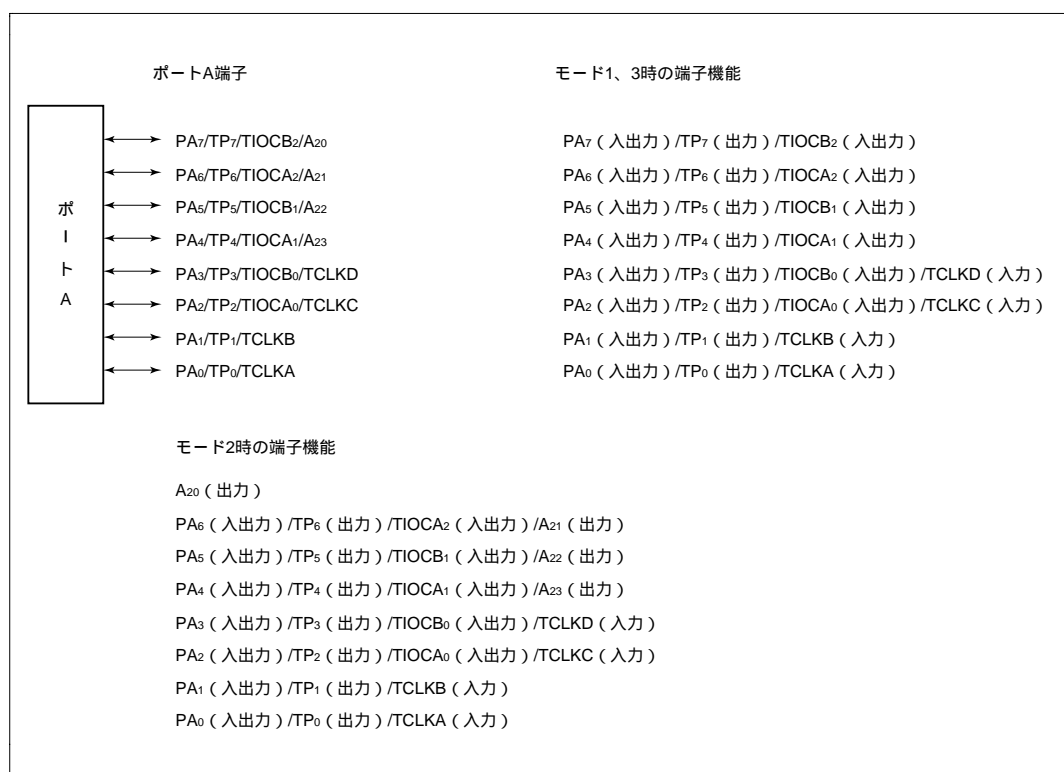


図 7.19 ポート A の端子構成

7.10.2 レジスタの構成と説明

表 7.15 にポート A のレジスタ構成を示します。

表 7.15 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、3	モード 2
H'FFD1	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'FFD3	ポート A データレジスタ	PADR	R/W	H'00	

【注】 * アドレスの下位 16 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 ₇ DDR	PA ₆ DDR	PA ₅ DDR	PA ₄ DDR	PA ₃ DDR	PA ₂ DDR	PA ₁ DDR	PA ₀ DDR
モード 2	初期値:	1	0	0	0	0	0	0	0
	R/W :	—	W	W	W	W	W	W	W
モード 1、3	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。

PADDR を "1" にセットすると対応するポート A の各端子は出力となり、"0" にクリアすると入力になります。ただし、モード 2 では PA₇DDR は "1" に固定され、PA₇ はアドレス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。このうち、ビット 7 はリードすると "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、3 では H'00 に、モード 2 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、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 の各端子 PA₇ ~ PA₀ のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	PA ₇	PA ₆	PA ₅	PA ₄	PA ₃	PA ₂	PA ₁	PA 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

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。

PADDR が"1"のとき、ポート A のリードを行うと PADR の値を直接リードします。PADDR が"0"のとき、ポート A のリードを行うと端子の状態が読み出されます。

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。

ポート A の各端子を TPC 出力端子として使用する場合は、PADR には、TPC 出力グループ 0、1 のパルス出力データが格納されます。ネクストデータイネーブルレジスタ (NDERA) の各ビット = "1"のとき、PADR の対応するビットのライトは禁止されます。

この場合、NDRA からの転送によってのみ、PADR の更新が行われます。

7.10.3 端子機能

ポート A の各端子は TPC の出力端子 (TP₇ ~ TP₀)、ITU の入出力端子 (TIOCB₂ ~ TIOCB₀、TIOCA₂ ~ TIOCA₀) と入力端子 (TCLKD、TCLKC、TCLKB、TCLKA)、およびアドレスバス (A₂₃ ~ A₂₀) と兼用になっています。ポート A の端子機能について表 7.16 に示します。

表 7.16 ポート A の端子機能

端子	選択方法と端子機能																																																																																					
PA ₇ /TP ₇ /TIOCB ₂ /A ₂₀	<p>モード設定と TMDR の PWM2 ビット、TIOCB2 の IOB2 ~ IOB0 ビットによる ITU チャンネル 2 の設定、NDERA の NDER7 ビット、および PADDR の PA₇DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>モード</th> <th colspan="4">1、3</th> <th>2</th> </tr> </thead> <tbody> <tr> <td>ITUチャンネル2の設定</td> <td>下表(1)</td> <td colspan="3">下表(2)</td> <td>—</td> </tr> <tr> <td>PA₇DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> <td>—</td> </tr> <tr> <td>NDER7</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCB₂出力</td> <td>PA₇入力</td> <td>PA₇出力</td> <td>TP₇出力</td> <td rowspan="2">A₂₀出力</td> </tr> <tr> <td colspan="3">TIOCB₂入力*</td> </tr> </tbody> </table> <p>【注】 * IOB2 = "1"、かつPWM2 = "0"の場合にTIOCB₂入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル2の設定</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> </tr> </thead> <tbody> <tr> <td>IOB2</td> <td colspan="2">0</td> <td>1</td> </tr> <tr> <td>IOB1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>IOB0</td> <td>0</td> <td>1</td> <td>—</td> </tr> </tbody> </table>	モード	1、3				2	ITUチャンネル2の設定	下表(1)	下表(2)			—	PA ₇ DDR	—	0	1	1	—	NDER7	—	—	0	1	—	端子機能	TIOCB ₂ 出力	PA ₇ 入力	PA ₇ 出力	TP ₇ 出力	A ₂₀ 出力	TIOCB ₂ 入力*			ITUチャンネル2の設定	(2)	(1)	(2)	IOB2	0		1	IOB1	0	0	1	IOB0	0	1	—																																				
モード	1、3				2																																																																																	
ITUチャンネル2の設定	下表(1)	下表(2)			—																																																																																	
PA ₇ DDR	—	0	1	1	—																																																																																	
NDER7	—	—	0	1	—																																																																																	
端子機能	TIOCB ₂ 出力	PA ₇ 入力	PA ₇ 出力	TP ₇ 出力	A ₂₀ 出力																																																																																	
		TIOCB ₂ 入力*																																																																																				
ITUチャンネル2の設定	(2)	(1)	(2)																																																																																			
IOB2	0		1																																																																																			
IOB1	0	0	1																																																																																			
IOB0	0	1	—																																																																																			
PA ₆ /TP ₆ /TIOCA ₂ /A ₂₁	<p>モード設定と BRCA の A₂₁E ビット、TMDR の PWM2 ビット、TIOCA2 の IOA2 ~ IOA0 ビットによる ITU チャンネル 2 の設定、NDERA の NDER6 ビット、および PADDR の PA₆DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>モード</th> <th colspan="4">1、3</th> <th colspan="3">2</th> </tr> </thead> <tbody> <tr> <td>A₂₁E</td> <td colspan="4">—</td> <td colspan="3">1</td> <td>0</td> </tr> <tr> <td>ITUチャンネル2の設定</td> <td>下表(1)</td> <td colspan="3">下表(2)</td> <td>下表(1)</td> <td colspan="2">下表(2)</td> <td>—</td> </tr> <tr> <td>PA₆DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER6</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCA₂出力</td> <td>PA₆入力</td> <td>PA₆出力</td> <td>TP₆出力</td> <td rowspan="2">TIOCA₂出力</td> <td>PA₆入力</td> <td>PA₆出力</td> <td>TP₆出力</td> <td rowspan="2">A₂₁出力</td> </tr> <tr> <td colspan="3">TIOCA₂入力*</td> <td colspan="3">TIOCA₂入力*</td> </tr> </tbody> </table> <p>【注】 * IOA2 = "1"の場合にTIOCA₂入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル2の設定</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> <th>(1)</th> </tr> </thead> <tbody> <tr> <td>PWM2</td> <td colspan="3">0</td> <td>1</td> </tr> <tr> <td>IOA2</td> <td colspan="2">0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA1</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	モード	1、3				2			A ₂₁ E	—				1			0	ITUチャンネル2の設定	下表(1)	下表(2)			下表(1)	下表(2)		—	PA ₆ DDR	—	0	1	1	—	0	1	1	NDER6	—	—	0	1	—	—	0	1	端子機能	TIOCA ₂ 出力	PA ₆ 入力	PA ₆ 出力	TP ₆ 出力	TIOCA ₂ 出力	PA ₆ 入力	PA ₆ 出力	TP ₆ 出力	A ₂₁ 出力	TIOCA ₂ 入力*			TIOCA ₂ 入力*			ITU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	—	—
モード	1、3				2																																																																																	
A ₂₁ E	—				1			0																																																																														
ITUチャンネル2の設定	下表(1)	下表(2)			下表(1)	下表(2)		—																																																																														
PA ₆ DDR	—	0	1	1	—	0	1	1																																																																														
NDER6	—	—	0	1	—	—	0	1																																																																														
端子機能	TIOCA ₂ 出力	PA ₆ 入力	PA ₆ 出力	TP ₆ 出力	TIOCA ₂ 出力	PA ₆ 入力	PA ₆ 出力	TP ₆ 出力	A ₂₁ 出力																																																																													
		TIOCA ₂ 入力*				TIOCA ₂ 入力*																																																																																
ITU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 ₅ /TP ₅ /TIOCB ₁ /A ₂₂	<p>モード設定と BRCR の A₂₂E ビット、TMDR の PWM1 ビット、TIOR1 の IOB2 ~ IOB0 ビットによる ITU チャンネル 1 の設定、NDERA の NDER5 ビット、および PADDR の PA₅DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>モード</th> <th colspan="4">1、3</th> <th colspan="4">2</th> </tr> </thead> <tbody> <tr> <td>A₂₂E</td> <td colspan="4">—</td> <td colspan="4">1</td> <td>0</td> </tr> <tr> <td>ITUチャンネル1の設定</td> <td>下表(1)</td> <td colspan="3">下表(2)</td> <td>下表(1)</td> <td colspan="3">下表(2)</td> <td>—</td> </tr> <tr> <td>PA₅DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> <td>—</td> </tr> <tr> <td>NDER5</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCB₁出力</td> <td>PA₅入力</td> <td>PA₅出力</td> <td>TP₅出力</td> <td rowspan="2">TIOCB₁出力</td> <td>PA₅入力</td> <td>PA₅出力</td> <td>TP₅出力</td> <td rowspan="2">A₂₂出力</td> </tr> <tr> <td colspan="3">TIOCB₁入力*</td> <td colspan="3">TIOCB₁入力*</td> </tr> </tbody> </table> <p>【注】 * IOB2 = "1"、かつPWM1 = "0"の場合にTIOCB₁入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル1の設定</th> <th>(2)</th> <th colspan="2">(1)</th> <th>(2)</th> </tr> </thead> <tbody> <tr> <td>IOB2</td> <td colspan="3">0</td> <td>1</td> </tr> <tr> <td>IOB1</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOB0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	モード	1、3				2				A ₂₂ E	—				1				0	ITUチャンネル1の設定	下表(1)	下表(2)			下表(1)	下表(2)			—	PA ₅ DDR	—	0	1	1	—	0	1	1	—	NDER5	—	—	0	1	—	—	0	1	—	端子機能	TIOCB ₁ 出力	PA ₅ 入力	PA ₅ 出力	TP ₅ 出力	TIOCB ₁ 出力	PA ₅ 入力	PA ₅ 出力	TP ₅ 出力	A ₂₂ 出力	TIOCB ₁ 入力*			TIOCB ₁ 入力*			ITUチャンネル1の設定	(2)	(1)		(2)	IOB2	0			1	IOB1	0	0	1	—	IOB0	0	1	—	—										
モード	1、3				2																																																																																											
A ₂₂ E	—				1				0																																																																																							
ITUチャンネル1の設定	下表(1)	下表(2)			下表(1)	下表(2)			—																																																																																							
PA ₅ DDR	—	0	1	1	—	0	1	1	—																																																																																							
NDER5	—	—	0	1	—	—	0	1	—																																																																																							
端子機能	TIOCB ₁ 出力	PA ₅ 入力	PA ₅ 出力	TP ₅ 出力	TIOCB ₁ 出力	PA ₅ 入力	PA ₅ 出力	TP ₅ 出力	A ₂₂ 出力																																																																																							
		TIOCB ₁ 入力*				TIOCB ₁ 入力*																																																																																										
ITUチャンネル1の設定	(2)	(1)		(2)																																																																																												
IOB2	0			1																																																																																												
IOB1	0	0	1	—																																																																																												
IOB0	0	1	—	—																																																																																												
PA ₄ /TP ₄ /TIOCA ₁ /A ₂₃	<p>モード設定と BRCR の A₂₃E ビット、TMDR の PWM1 ビット、TIOR1 の IOA2 ~ IOA0 ビットによる ITU チャンネル 1 の設定、NDERA の NDER4 ビット、および PADDR の PA₄DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>モード</th> <th colspan="4">1、3</th> <th colspan="4">2</th> </tr> </thead> <tbody> <tr> <td>A₂₃E</td> <td colspan="4">—</td> <td colspan="4">1</td> <td>0</td> </tr> <tr> <td>ITUチャンネル1の設定</td> <td>下表(1)</td> <td colspan="3">下表(2)</td> <td>下表(1)</td> <td colspan="3">下表(2)</td> <td>—</td> </tr> <tr> <td>PA₄DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> <td>—</td> </tr> <tr> <td>NDER4</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCA₁出力</td> <td>PA₄入力</td> <td>PA₄出力</td> <td>TP₄出力</td> <td rowspan="2">TIOCA₁出力</td> <td>PA₄入力</td> <td>PA₄出力</td> <td>TP₄出力</td> <td rowspan="2">A₂₃出力</td> </tr> <tr> <td colspan="3">TIOCA₁入力*</td> <td colspan="3">TIOCA₁入力*</td> </tr> </tbody> </table> <p>【注】 * IOA2 = "1"の場合にTIOCA₁入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル1の設定</th> <th>(2)</th> <th colspan="2">(1)</th> <th>(2)</th> <th>(1)</th> </tr> </thead> <tbody> <tr> <td>PWM1</td> <td colspan="3">0</td> <td colspan="2">1</td> </tr> <tr> <td>IOA2</td> <td colspan="3">0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA1</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> <tr> <td>IOA0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	モード	1、3				2				A ₂₃ E	—				1				0	ITUチャンネル1の設定	下表(1)	下表(2)			下表(1)	下表(2)			—	PA ₄ DDR	—	0	1	1	—	0	1	1	—	NDER4	—	—	0	1	—	—	0	1	—	端子機能	TIOCA ₁ 出力	PA ₄ 入力	PA ₄ 出力	TP ₄ 出力	TIOCA ₁ 出力	PA ₄ 入力	PA ₄ 出力	TP ₄ 出力	A ₂₃ 出力	TIOCA ₁ 入力*			TIOCA ₁ 入力*			ITU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	—	—	—
モード	1、3				2																																																																																											
A ₂₃ E	—				1				0																																																																																							
ITUチャンネル1の設定	下表(1)	下表(2)			下表(1)	下表(2)			—																																																																																							
PA ₄ DDR	—	0	1	1	—	0	1	1	—																																																																																							
NDER4	—	—	0	1	—	—	0	1	—																																																																																							
端子機能	TIOCA ₁ 出力	PA ₄ 入力	PA ₄ 出力	TP ₄ 出力	TIOCA ₁ 出力	PA ₄ 入力	PA ₄ 出力	TP ₄ 出力	A ₂₃ 出力																																																																																							
		TIOCA ₁ 入力*				TIOCA ₁ 入力*																																																																																										
ITU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	—	—	—																																																																																											

端子	選択方法と端子機能																																																					
PA ₃ /TP ₃ /TIOCB ₀ / TCLKD	<p>TMDR の PWM0 ビット、TIOR0 の IOB2 ~ IOB0 ビットによる ITU チャンネル 0 の設定、TCR4 ~ TCR0 の TPSC2 ~ TPSC0 ビット、NDERA の NDER3 ビット、および PADDR の PA₃DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル0の設定</th> <th>下表(1)</th> <th colspan="3">下表(2)</th> </tr> </thead> <tbody> <tr> <td>PA₃DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER3</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCB₀出力</td> <td>PA₃入力</td> <td>PA₃出力</td> <td>TP₃出力</td> </tr> <tr> <td colspan="3">TIOCB₀入力*1</td> </tr> <tr> <td colspan="5">TCLKD入力*2</td> </tr> </tbody> </table> <p>【注】 *1 IOB2 = "1"、かつPWM0 = "0"の場合にTIOCB₀入力となります。 *2 TCR4 ~ TCR0のいずれかの設定がTPSC2 = TPSC1 = TPSC0 = "1"の場合にTCLKD入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル0の設定</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> </tr> </thead> <tbody> <tr> <td>IOB2</td> <td colspan="2">0</td> <td>1</td> </tr> <tr> <td>IOB1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>IOB0</td> <td>0</td> <td>1</td> <td>—</td> </tr> </tbody> </table>	ITUチャンネル0の設定	下表(1)	下表(2)			PA ₃ DDR	—	0	1	1	NDER3	—	—	0	1	端子機能	TIOCB ₀ 出力	PA ₃ 入力	PA ₃ 出力	TP ₃ 出力	TIOCB ₀ 入力*1			TCLKD入力*2					ITUチャンネル0の設定	(2)	(1)	(2)	IOB2	0		1	IOB1	0	0	1	IOB0	0	1	—									
ITUチャンネル0の設定	下表(1)	下表(2)																																																				
PA ₃ DDR	—	0	1	1																																																		
NDER3	—	—	0	1																																																		
端子機能	TIOCB ₀ 出力	PA ₃ 入力	PA ₃ 出力	TP ₃ 出力																																																		
		TIOCB ₀ 入力*1																																																				
TCLKD入力*2																																																						
ITUチャンネル0の設定	(2)	(1)	(2)																																																			
IOB2	0		1																																																			
IOB1	0	0	1																																																			
IOB0	0	1	—																																																			
PA ₂ /TP ₂ /TIOCA ₀ / TCLKC	<p>TMDR の PWM0 ビット、TIOR0 の IOA2 ~ IOA0 ビットによる ITU チャンネル 0 の設定、TCR4 ~ TCR0 の TPSC2 ~ TPSC0 ビット、NDERA の NDER2 ビット、および PADDR の PA₂DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル0の設定</th> <th>下表(1)</th> <th colspan="3">下表(2)</th> </tr> </thead> <tbody> <tr> <td>PA₂DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER2</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCA₀出力</td> <td>PA₂入力</td> <td>PA₂出力</td> <td>TP₂出力</td> </tr> <tr> <td colspan="3">TIOCA₀入力*1</td> </tr> <tr> <td colspan="5">TCLKC入力*2</td> </tr> </tbody> </table> <p>【注】 *1 IOA2 = "1"の場合にTIOCA₀入力となります。 *2 TCR4 ~ TCR0のいずれかの設定がTPSC2 = TPSC1 = "1"、TPSC0 = "0"の場合にTCLKC入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル0の設定</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> <th>(1)</th> </tr> </thead> <tbody> <tr> <td>PWM0</td> <td colspan="3">0</td> <td>1</td> </tr> <tr> <td>IOA2</td> <td colspan="2">0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA1</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	ITUチャンネル0の設定	下表(1)	下表(2)			PA ₂ DDR	—	0	1	1	NDER2	—	—	0	1	端子機能	TIOCA ₀ 出力	PA ₂ 入力	PA ₂ 出力	TP ₂ 出力	TIOCA ₀ 入力*1			TCLKC入力*2					ITUチャンネル0の設定	(2)	(1)	(2)	(1)	PWM0	0			1	IOA2	0		1	—	IOA1	0	0	1	—	IOA0	0	1	—	—
ITUチャンネル0の設定	下表(1)	下表(2)																																																				
PA ₂ DDR	—	0	1	1																																																		
NDER2	—	—	0	1																																																		
端子機能	TIOCA ₀ 出力	PA ₂ 入力	PA ₂ 出力	TP ₂ 出力																																																		
		TIOCA ₀ 入力*1																																																				
TCLKC入力*2																																																						
ITUチャンネル0の設定	(2)	(1)	(2)	(1)																																																		
PWM0	0			1																																																		
IOA2	0		1	—																																																		
IOA1	0	0	1	—																																																		
IOA0	0	1	—	—																																																		

端子	選択方法と端子機能															
PA ₁ /TP ₁ /TCLKB	<p>NDERAの NDER1 ビット、および PADDR の PA₁DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1" data-bbox="635 434 1423 573"> <tr> <td>PA₁DDR</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER1</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td>PA₁入力</td> <td>PA₁出力</td> <td>TP₁出力</td> </tr> <tr> <td colspan="3">TCLKB入力*</td> </tr> </table> <p>【注】 * TMDRのMDF = "1"の場合、またはTCR4 ~ TCR0のいずれかの設定が TPSC2 = "1"、TPSC1 = "0"、TPSC0 = "1"の場合にTCLKB入力となります。</p>	PA ₁ DDR	0	1	1	NDER1	—	0	1	端子機能	PA ₁ 入力	PA ₁ 出力	TP ₁ 出力	TCLKB入力*		
PA ₁ DDR	0	1	1													
NDER1	—	0	1													
端子機能	PA ₁ 入力	PA ₁ 出力	TP ₁ 出力													
	TCLKB入力*															
PA ₀ /TP ₀ /TCLKA	<p>NDERAの NDER0 ビット、および PADDR の PA₀DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1" data-bbox="635 815 1423 954"> <tr> <td>PA₀DDR</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER0</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td>PA₀入力</td> <td>PA₀出力</td> <td>TP₀出力</td> </tr> <tr> <td colspan="3">TCLKA入力*</td> </tr> </table> <p>【注】 * TMDRのMDF = "1"の場合、またはTCR4 ~ TCR0のいずれかの設定が TPSC2 = "1"、TPSC1 = TPSC0 = "0"の場合にTCLKA入力となります。</p>	PA ₀ DDR	0	1	1	NDER0	—	0	1	端子機能	PA ₀ 入力	PA ₀ 出力	TP ₀ 出力	TCLKA入力*		
PA ₀ DDR	0	1	1													
NDER0	—	0	1													
端子機能	PA ₀ 入力	PA ₀ 出力	TP ₀ 出力													
	TCLKA入力*															

7.11 ポート B

7.11.1 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 TPC の出力端子 (TP₁₅ ~ TP₈)、ITU の入出力端子 (TIOCB₄、TIOCB₃、TIOCA₄、TIOCA₃) と出力端子 (TOCXB₄、TOCXA₄)、A/D 変換器の $\overline{\text{ADTRG}}$ 入力端子と兼用になっています。ポート B の端子機能はいずれの動作モードでも共通です。ポート B の端子構成を図 7.20 に示します。

ポート B は、1 個の TTL 負荷と 30pF の容量負荷を駆動できます。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。PB₃ ~ PB₀ は、シュミットトリガ入力です。

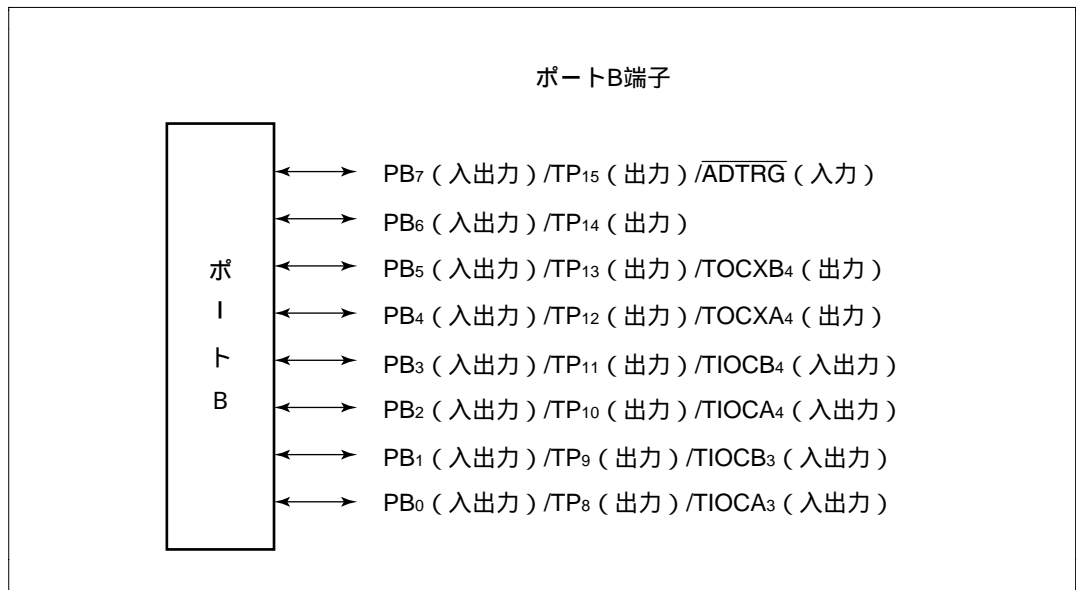


図 7.20 ポート B の端子構成

7.11.2 レジスタの構成と説明

表 7.17 にポート B のレジスタ構成を示します。

表 7.17 ポート B レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFD4	ポート B データディレクションレジスタ	PBDDR	W	H'00
H'FFD6	ポート B データレジスタ	PBDR	R/W	H'00

【注】 * アドレスの下位 16 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。

ビット:	7	6	5	4	3	2	1	0
	PB ₇ DDR	PB ₆ DDR	PB ₅ DDR	PB ₄ DDR	PB ₃ DDR	PB ₂ DDR	PB ₁ DDR	PB ₀ DDR
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。

PBDDR を"1"にセットすると対応するポート B の各端子は出力となり、"0"にクリアすると入力になります。

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、"0"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、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 の各端子 PB₇~PB₀ のデータを格納します。

ビット:	7	6	5	4	3	2	1	0
	PB ₇	PB ₆	PB ₅	PB ₄	PB ₃	PB ₂	PB ₁	PB 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

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。

PBDDR が"1"のとき、ポート B のリードを行うと PBDR の値を直接リードします。PBDDR が"0"のときポート B のリードを行うと端子の状態が読み出されます。

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。

ポート B の各端子を TPC 出力端子として使用する場合は、PBDR には TPC 出力グループ 2、3 のパルス出力データが格納されます。

ネクストデータイネーブルレジスタ B (NDERB) の各ビット = "1"のとき、PBDR の対

応するビットのライトは禁止されます。

この場合、NDRB からの転送によってのみ PBDR の更新が行われます。

7.11.3 端子機能

ポート B の各端子は TPC の出力端子 (TP₁₅ ~ TP₈)、ITU の入出力端子 (TIOCB₄、TIOCB₃、TIOCA₄、TIOCA₃) と出力端子 (TOCXB₄、TOCXA₄)、および $\overline{\text{ADTRG}}$ 入力端子と兼用になっています。ポート B の端子機能について表 7.15 に示します。

表 7.18 ポート B の端子機能

端子	選択方法と端子機能			
PB ₇ /TP ₁₅ / $\overline{\text{ADTRG}}$	ADCR の TRGE ビット、NDERB の NDER15 ビット、および PBDDR の PB ₇ 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。			
	PB ₇ DDR	0	1	1
	NDER15	—	0	1
	端子機能	PB ₇ 入力	PB ₇ 出力	TP ₁₅ 出力
		$\overline{\text{ADTRG}}$ 入力*		
	【注】 * TRGE = "1" のとき $\overline{\text{ADTRG}}$ 入力となります。			
PB ₆ /TP ₁₄	NDERB の NDER14 ビット、および PBDDR の PB ₆ 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。			
	PB ₆ DDR	0	1	1
	NDER14	—	0	1
	端子機能	PB ₆ 入力	PB ₆ 出力	TP ₁₄ 出力
PB ₅ /TP ₁₃ /TOCXB ₄	TFCR の CMD1 ビット、TOER の EXB4 ビットによる ITU チャネル 4 の設定、NDERB の NDER13 ビット、および PBDDR の PB ₅ 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。			
	EXB4、CMD1	いずれかが"0"		いずれも"1"
	PB ₅ DDR	0	1	1
	NDER13	—	0	1
	端子機能	PB ₅ 入力	PB ₅ 出力	TP ₁₃ 出力 TOCXB ₄ 出力
PB ₄ /TP ₁₂ /TOCXA ₄	TFCR の CMD1 ビット、TOER の EXA4 ビットによる ITU チャネル 4 の設定、NDERB の NDER12 ビット、および PBDDR の PB ₄ DDR ビットの組合わせにより、次のように切り換わります。			
	EXA4、CMD1	いずれかが"0"		いずれも"1"
	PB ₄ DDR	0	1	1
	NDER12	—	0	1
	端子機能	PB ₄ 入力	PB ₄ 出力	TP ₁₂ 出力 TOCXA ₄ 出力

端子	選択方法と端子機能																																																																					
PB ₃ /TP ₁₁ /TIOCB ₄	<p>TMDR の PWM4 ビット、TFCR の CMD1 ビット、TOER の EB4 ビット、および TIOR4 の IOB2 ~ IOB0 ビットによる ITU チャンネル 4 の設定、NDERB の NDER11 ビット、および PBDDR の PB₃DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル4の設定</th> <th>下表(1)</th> <th colspan="3">下表(2)</th> </tr> </thead> <tbody> <tr> <td>PB₃DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER11</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCB₄出力</td> <td>PB₃入力</td> <td>PB₃出力</td> <td>TP₁₁出力</td> </tr> <tr> <td colspan="3">TIOCB₄入力*</td> </tr> </tbody> </table> <p>【注】 * CMD1 = PWM4 = "0"、かつ IOB2 = "1" の場合に TIOCB₄ 入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル4の設定</th> <th>(2)</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> <th>(1)</th> </tr> </thead> <tbody> <tr> <td>EB4</td> <td>0</td> <td colspan="4">1</td> </tr> <tr> <td>CMD1</td> <td>—</td> <td colspan="3">0</td> <td>1</td> </tr> <tr> <td>IOB2</td> <td>—</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOB1</td> <td>—</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> <tr> <td>IOB0</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	ITUチャンネル4の設定	下表(1)	下表(2)			PB ₃ DDR	—	0	1	1	NDER11	—	—	0	1	端子機能	TIOCB ₄ 出力	PB ₃ 入力	PB ₃ 出力	TP ₁₁ 出力	TIOCB ₄ 入力*			ITUチャンネル4の設定	(2)	(2)	(1)	(2)	(1)	EB4	0	1				CMD1	—	0			1	IOB2	—	0	0	0	1	—	IOB1	—	0	0	1	—	—	IOB0	—	0	1	—	—	—							
ITUチャンネル4の設定	下表(1)	下表(2)																																																																				
PB ₃ DDR	—	0	1	1																																																																		
NDER11	—	—	0	1																																																																		
端子機能	TIOCB ₄ 出力	PB ₃ 入力	PB ₃ 出力	TP ₁₁ 出力																																																																		
		TIOCB ₄ 入力*																																																																				
ITUチャンネル4の設定	(2)	(2)	(1)	(2)	(1)																																																																	
EB4	0	1																																																																				
CMD1	—	0			1																																																																	
IOB2	—	0	0	0	1	—																																																																
IOB1	—	0	0	1	—	—																																																																
IOB0	—	0	1	—	—	—																																																																
PB ₂ /TP ₁₀ /TIOCA ₄	<p>TFCR の CMD1 ビット、TOER の EA4 ビット、TMDR の PWM4 ビット、および TIOR4 の IOA2 ~ IOA0 ビットによる ITU チャンネル 4 の設定、NDERB の NDER10 ビット、および PBDDR の PB₂DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル4の設定</th> <th>下表(1)</th> <th colspan="3">下表(2)</th> </tr> </thead> <tbody> <tr> <td>PB₂DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER10</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCA₄出力</td> <td>PB₂入力</td> <td>PB₂出力</td> <td>TP₁₀出力</td> </tr> <tr> <td colspan="3">TIOCA₄入力*</td> </tr> </tbody> </table> <p>【注】 * CMD1 = PWM4 = "0"、IOA2 = "1" の場合に TIOCA₄ 入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル4の設定</th> <th>(2)</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> <th>(1)</th> </tr> </thead> <tbody> <tr> <td>EA4</td> <td>0</td> <td colspan="4">1</td> </tr> <tr> <td>CMD1</td> <td>—</td> <td colspan="3">0</td> <td>1</td> </tr> <tr> <td>PWM4</td> <td>—</td> <td colspan="3">0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA2</td> <td>—</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA1</td> <td>—</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> <tr> <td>IOA0</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	ITUチャンネル4の設定	下表(1)	下表(2)			PB ₂ DDR	—	0	1	1	NDER10	—	—	0	1	端子機能	TIOCA ₄ 出力	PB ₂ 入力	PB ₂ 出力	TP ₁₀ 出力	TIOCA ₄ 入力*			ITUチャンネル4の設定	(2)	(2)	(1)	(2)	(1)	EA4	0	1				CMD1	—	0			1	PWM4	—	0			1	—	IOA2	—	0	0	0	1	—	IOA1	—	0	0	1	—	—	IOA0	—	0	1	—	—	—
ITUチャンネル4の設定	下表(1)	下表(2)																																																																				
PB ₂ DDR	—	0	1	1																																																																		
NDER10	—	—	0	1																																																																		
端子機能	TIOCA ₄ 出力	PB ₂ 入力	PB ₂ 出力	TP ₁₀ 出力																																																																		
		TIOCA ₄ 入力*																																																																				
ITUチャンネル4の設定	(2)	(2)	(1)	(2)	(1)																																																																	
EA4	0	1																																																																				
CMD1	—	0			1																																																																	
PWM4	—	0			1	—																																																																
IOA2	—	0	0	0	1	—																																																																
IOA1	—	0	0	1	—	—																																																																
IOA0	—	0	1	—	—	—																																																																

端子	選択方法と端子機能																																																																					
PB ₁ /TP ₉ /TIOCB ₃	<p>TMDR の PWM3 ビット、TFCR の CMD1 ビット、TOER の EB3 ビット、および TIOR3 の IOB2 ~ IOB0 ビットによる ITU チャンネル 3 の設定、NDERB の NDER9 ビット、および PBDDR の PB₁DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル3の設定</th> <th>下表(1)</th> <th colspan="3">下表(2)</th> </tr> </thead> <tbody> <tr> <td>PB₁DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER9</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCB₃出力</td> <td>PB₁入力</td> <td>PB₁出力</td> <td>TP₉出力</td> </tr> <tr> <td colspan="3">TIOCB₃入力*</td> </tr> </tbody> </table> <p>【注】 * CMD1 = PWM3 = "0"、IOB2 = "1" の場合に TIOCB₃ 入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル3の設定</th> <th>(2)</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> <th>(1)</th> </tr> </thead> <tbody> <tr> <td>EB3</td> <td>0</td> <td colspan="4">1</td> </tr> <tr> <td>CMD1</td> <td>—</td> <td colspan="3">0</td> <td>1</td> </tr> <tr> <td>IOB2</td> <td>—</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOB1</td> <td>—</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> <tr> <td>IOB0</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	ITUチャンネル3の設定	下表(1)	下表(2)			PB ₁ DDR	—	0	1	1	NDER9	—	—	0	1	端子機能	TIOCB ₃ 出力	PB ₁ 入力	PB ₁ 出力	TP ₉ 出力	TIOCB ₃ 入力*			ITUチャンネル3の設定	(2)	(2)	(1)	(2)	(1)	EB3	0	1				CMD1	—	0			1	IOB2	—	0	0	0	1	—	IOB1	—	0	0	1	—	—	IOB0	—	0	1	—	—	—							
ITUチャンネル3の設定	下表(1)	下表(2)																																																																				
PB ₁ DDR	—	0	1	1																																																																		
NDER9	—	—	0	1																																																																		
端子機能	TIOCB ₃ 出力	PB ₁ 入力	PB ₁ 出力	TP ₉ 出力																																																																		
		TIOCB ₃ 入力*																																																																				
ITUチャンネル3の設定	(2)	(2)	(1)	(2)	(1)																																																																	
EB3	0	1																																																																				
CMD1	—	0			1																																																																	
IOB2	—	0	0	0	1	—																																																																
IOB1	—	0	0	1	—	—																																																																
IOB0	—	0	1	—	—	—																																																																
PB ₀ /TP ₈ /TIOCA ₃	<p>TFCR の CMD1 ビット、TOER の EA3 ビット、TMDR の PWM3 ビット、および TIOR3 の IOA2 ~ IOA0 ビットによる ITU チャンネル 3 の設定、NDERB の NDER8 ビット、および PBDDR の PB₀DDR ビットの組み合わせにより、次のように切り換わります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル3の設定</th> <th>下表(1)</th> <th colspan="3">下表(2)</th> </tr> </thead> <tbody> <tr> <td>PB₀DDR</td> <td>—</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>NDER8</td> <td>—</td> <td>—</td> <td>0</td> <td>1</td> </tr> <tr> <td rowspan="2">端子機能</td> <td rowspan="2">TIOCA₃出力</td> <td>PB₀入力</td> <td>PB₀出力</td> <td>TP₈出力</td> </tr> <tr> <td colspan="3">TIOCA₃入力*</td> </tr> </tbody> </table> <p>【注】 * CMD1 = PWM3 = "0"、IOA2 = "1" の場合に TIOCA₃ 入力となります。</p> <table border="1"> <thead> <tr> <th>ITUチャンネル3の設定</th> <th>(2)</th> <th>(2)</th> <th>(1)</th> <th>(2)</th> <th>(1)</th> </tr> </thead> <tbody> <tr> <td>EA3</td> <td>0</td> <td colspan="4">1</td> </tr> <tr> <td>CMD1</td> <td>—</td> <td colspan="3">0</td> <td>1</td> </tr> <tr> <td>PWM3</td> <td>—</td> <td colspan="3">0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA2</td> <td>—</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> </tr> <tr> <td>IOA1</td> <td>—</td> <td>0</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> </tr> <tr> <td>IOA0</td> <td>—</td> <td>0</td> <td>1</td> <td>—</td> <td>—</td> <td>—</td> </tr> </tbody> </table>	ITUチャンネル3の設定	下表(1)	下表(2)			PB ₀ DDR	—	0	1	1	NDER8	—	—	0	1	端子機能	TIOCA ₃ 出力	PB ₀ 入力	PB ₀ 出力	TP ₈ 出力	TIOCA ₃ 入力*			ITUチャンネル3の設定	(2)	(2)	(1)	(2)	(1)	EA3	0	1				CMD1	—	0			1	PWM3	—	0			1	—	IOA2	—	0	0	0	1	—	IOA1	—	0	0	1	—	—	IOA0	—	0	1	—	—	—
ITUチャンネル3の設定	下表(1)	下表(2)																																																																				
PB ₀ DDR	—	0	1	1																																																																		
NDER8	—	—	0	1																																																																		
端子機能	TIOCA ₃ 出力	PB ₀ 入力	PB ₀ 出力	TP ₈ 出力																																																																		
		TIOCA ₃ 入力*																																																																				
ITUチャンネル3の設定	(2)	(2)	(1)	(2)	(1)																																																																	
EA3	0	1																																																																				
CMD1	—	0			1																																																																	
PWM3	—	0			1	—																																																																
IOA2	—	0	0	0	1	—																																																																
IOA1	—	0	0	1	—	—																																																																
IOA0	—	0	1	—	—	—																																																																

8. 16ビットインテグレート ッドタイマユニット (ITU)

第8章 目次

8.1	概要	181
8.1.1	特長	181
8.1.2	ブロック図	184
8.1.3	端子構成	189
8.1.4	レジスタ構成	191
8.2	各レジスタの説明	193
8.2.1	タイマスタートレジスタ (TSTR)	193
8.2.2	タイマシンクロレジスタ (TSNC)	195
8.2.3	タイマモードレジスタ (TMDR)	197
8.2.4	タイマファンクションコントロールレジスタ (TFCR)	201
8.2.5	タイマアウトプットマスタイネーブルレジスタ (TOER)	204
8.2.6	タイマアウトプットコントロールレジスタ (TOCR)	207
8.2.7	タイマカウンタ (TCNT)	209
8.2.8	ジェネラルレジスタ A、B (GRA、GRB)	210
8.2.9	バッファレジスタ A、B (BRA、BRB)	211
8.2.10	タイマコントロールレジスタ (TCR)	212
8.2.11	タイマI/Oコントロールレジスタ (TIOR)	215
8.2.12	タイマステータスレジスタ (TSR)	218
8.2.13	タイマインタラプトイネーブルレジスタ (TIER)	221
8.3	CPUとのインタフェース	223
8.3.1	16ビットアクセス可能なレジスタ	223
8.3.2	8ビットアクセスのレジスタ	224
8.4	動作説明	226

8. 16ビットインテグレートドタイマユニット (ITU)

8.4.1	概要	226
8.4.2	基本機能.....	227
8.4.3	同期動作.....	235
8.4.4	PWM モード	236
8.4.5	リセット同期 PWM モード.....	240
8.4.6	相補 PWM モード	243
8.4.7	位相計数モード.....	250
8.4.8	バッファ動作	251
8.4.9	ITU 出力タイミング	256
8.5	割込み.....	259
8.5.1	ステータスフラグのセットタイミング	259
8.5.2	ステータスフラグのクリアタイミング	261
8.5.3	割込み要因	261
8.6	使用上の注意.....	263

8.1 概要

本 LSI は、5 チャンネルの 16 ビットタイマにより構成される 16 ビットインテグレートドタイマユニット (ITU) を内蔵しています。

8.1.1 特長

ITU の特長を以下に示します。

最大 12 種類のパルス出力、または最大 10 種類のパルス入力処理が可能

各チャンネル 2 本、合計 10 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/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 (TCNT) への同時書込みが可能
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 5 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

チャンネル3、4は次の動作モードを設定可能

- ・リセット同期PWMモード : チャンネル3、4を組み合わせることにより、
正相・逆相のPWM波形を3相出力可能
- ・相補PWMモード : チャンネル3、4を組み合わせることにより、
正相・逆相がノンオーバーラップの関係にある
PWM波形を3相出力可能
- ・バッファ動作 : インพุットキャプチャレジスタのダブルバッ
ファ構成が可能
アウトプットコンペアレジスタの自動書換
えが可能

内部16ビットバスによる高速アクセス

TCNT、GR、およびバッファレジスタ (BR) 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

15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พุットキャプチャ兼用割込み×2要因、オー
バフロー割込み×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) の出力トリガを生成可能

チャンネル0~3のコンペアマッチ/インพุットキャプチャ信号をTPCの出力トリガ
として使用可能

ITUの機能一覧を表8.1に示します。

表8.1 ITU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	チャンネル3	チャンネル4
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	GRA3、GRB3	GRA4、GRB4
バッファレジスタ				BRA3、BRB3	BRA4、BRB4
入出力端子	TIOCA ₀ 、TIOCB ₀	TIOCA ₁ 、TIOCB ₁	TIOCA ₂ 、TIOCB ₂	TIOCA ₃ 、TIOCB ₃	TIOCA ₄ 、TIOCB ₄
出力端子					TOCXA ₄ 、TOCXB ₄
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	GRA3/GRB3のコ ンペアマッチまた はインプットキャ プチャ	GRA4/GRB4のコ ンペアマッチまた はインプットキャ プチャ
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					
リセット同期 PWMモード					
相補PWMモード					
位相計数モード					
バッファ動作					
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 ・オーバフロー	3要因 ・コンペアマッチ /インプットキ ャプチャA3 ・コンペアマッチ /インプットキ ャプチャB3 ・オーバフロー	3要因 ・コンペアマッチ /インプットキ ャプチャA4 ・コンペアマッチ /インプットキ ャプチャB4 ・オーバフロー

《記号説明》

○ : 可能

- : 不可

8.1.2 ブロック図

(1) ITUのブロック図 (全体図)

ITUのブロック図 (全体図) を図8.1 に示します。

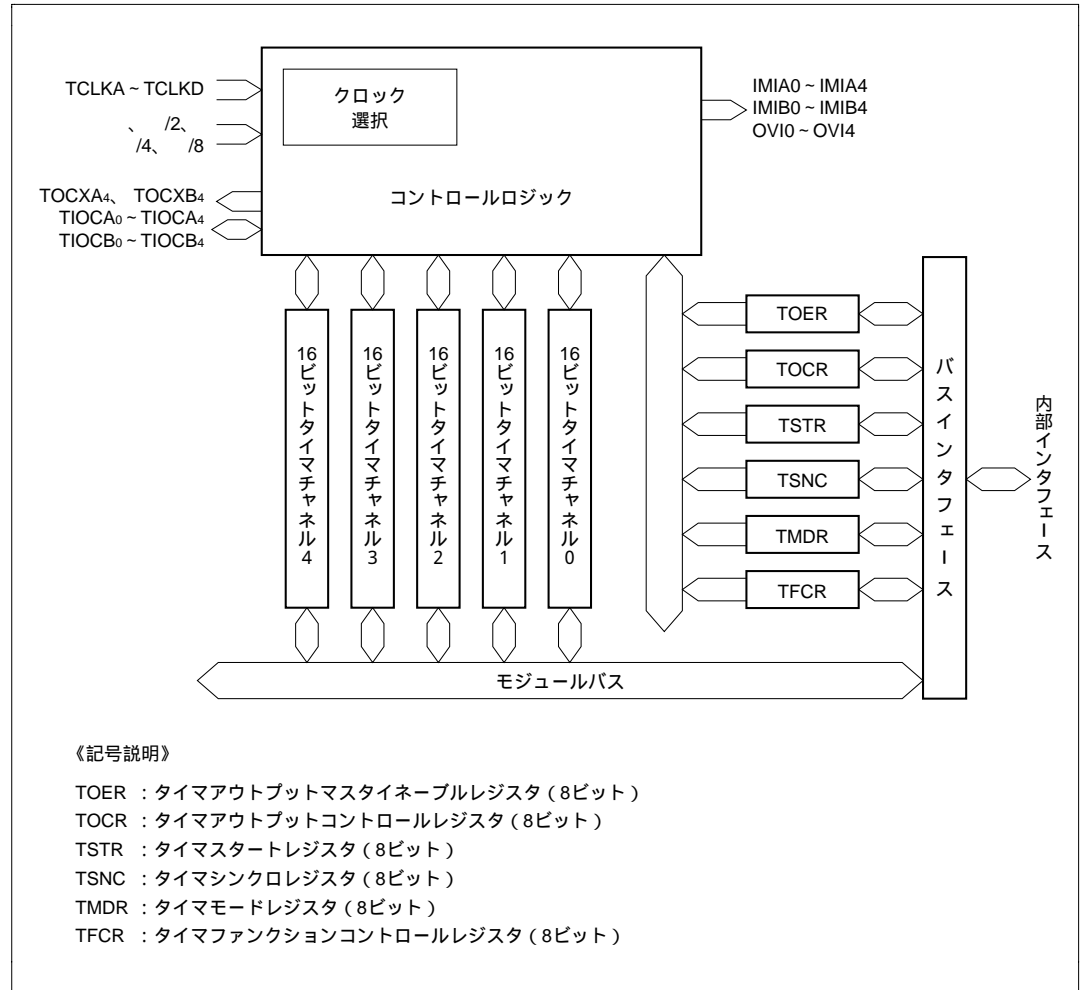


図8.1 ITUのブロック図 (全体図)

(2) チャンネル0、1のブロック図

ITUのチャンネル0、1は同一の機能をもっています。チャンネル0、1のブロック図を図8.2に示します。

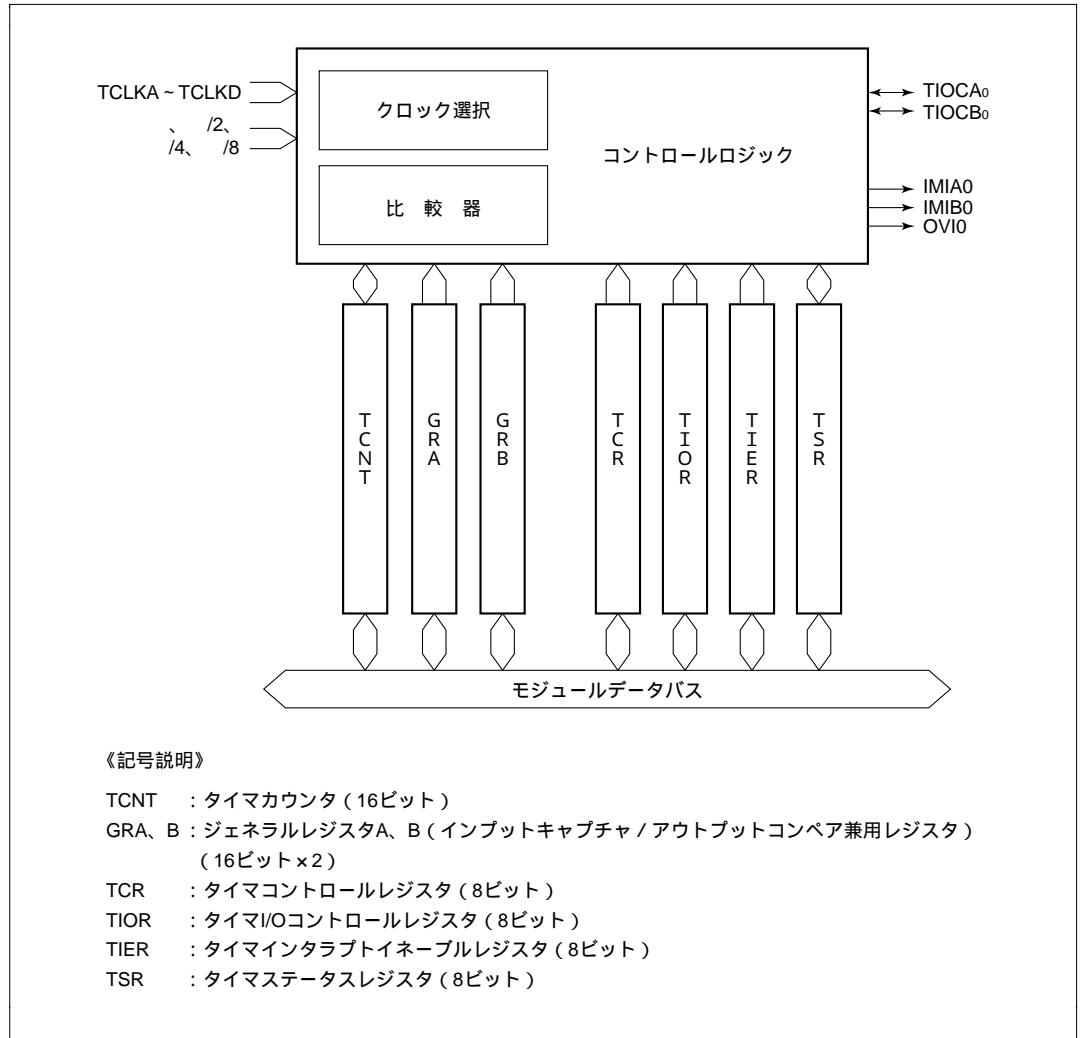


図8.2 チャンネル0、1のブロック図 (チャンネル0の場合)

(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8.3に示します。チャンネル2は0出力、1出力のみ可能です。

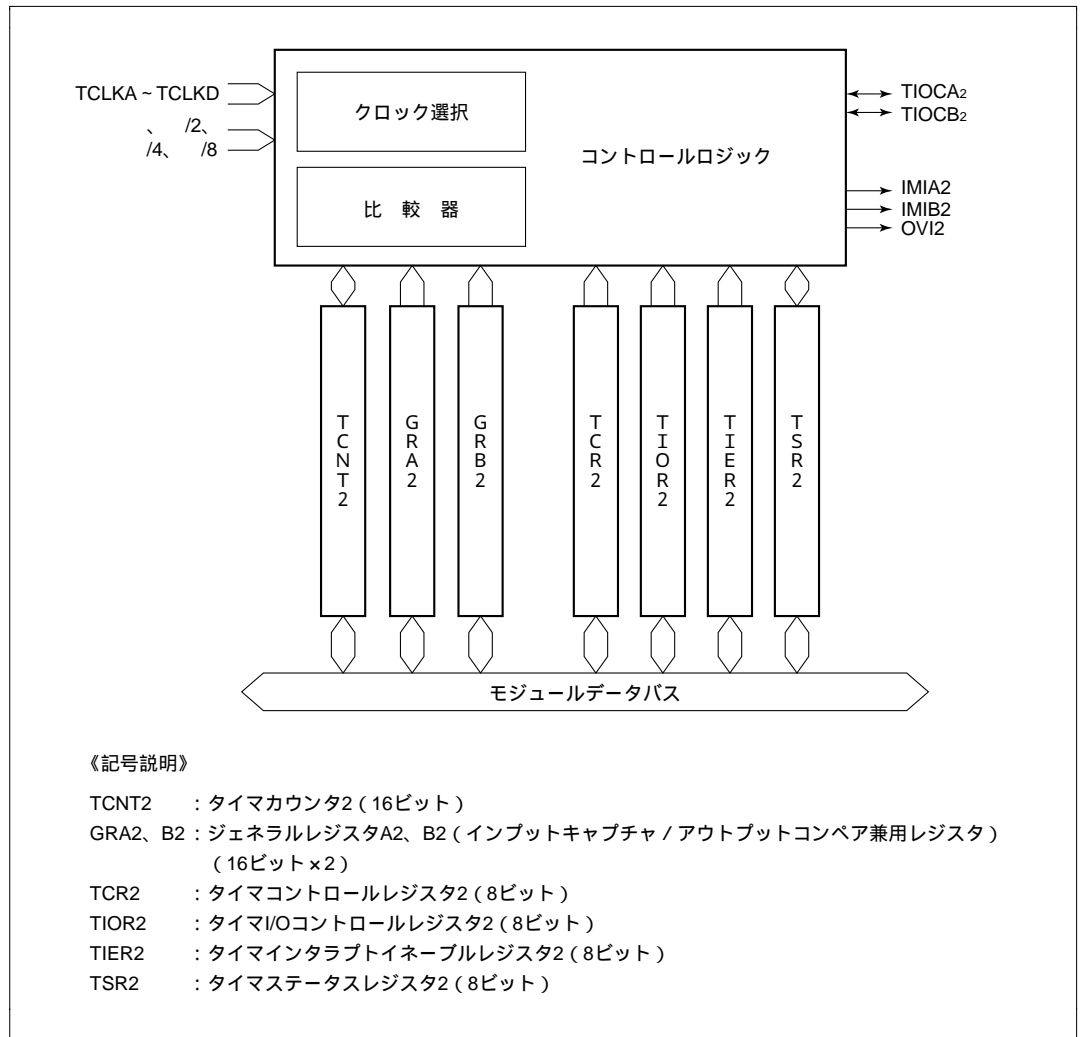


図 8.3 チャンネル2のブロック図

(4) チャンネル3、4のブロック図

チャンネル3のブロック図を図8.4、チャンネル4のブロック図を図8.5に示します。

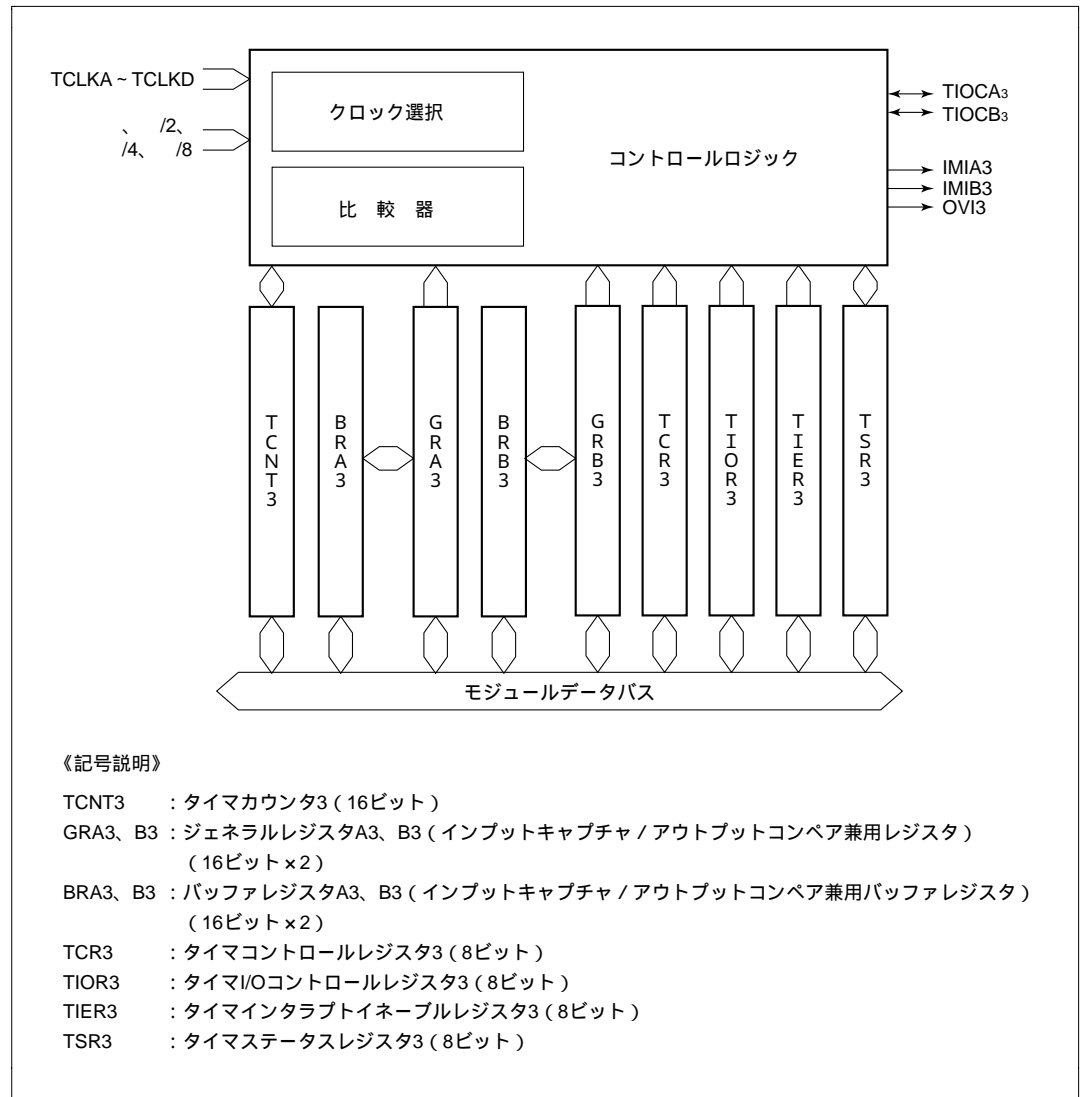


図8.4 チャンネル3のブロック図

8. 16ビットインテグレートドタイマユニット (ITU)

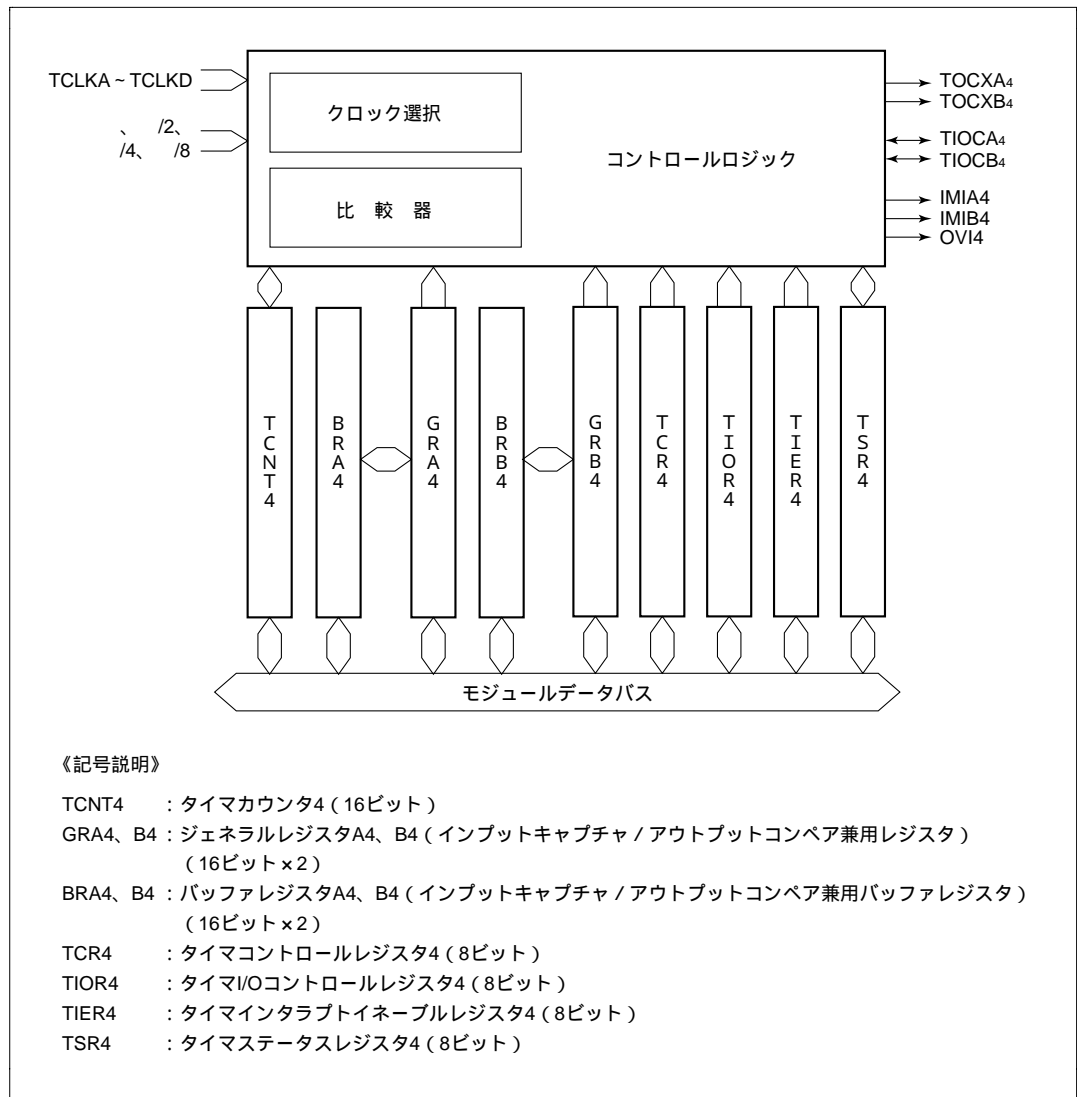


図 8.5 チャンネル4のブロック図

8.1.3 端子構成

ITU の端子構成を表 8.2 に示します。

表 8.2 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 ₀	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 ₀	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 ₁	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 ₁	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 ₂	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 ₂	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
3	インプットキャプチャ / アウトプットコンペア A3	TIOCA ₃	入出力	GRA3 アウトプットコンペア出力 / GRA3 インプットキャプチャ入力 / PWM 出力端子 (PWM モード / 相補 PWM モード / リセット同期 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 B3	TIOCB ₃	入出力	GRB3 アウトプットコンペア出力 / GRB3 インプットキャプチャ入力 / PWM 出力端子 (相補 PWM モード / リセット同期 PWM モード時)

8. 16ビットインテグレートドタイマユニット (ITU)

チャンネル	名称	略称	入出力	機能
4	インプットキャプチャ / アウトプットコンペア A4	TIOCA ₄	入出力	GRA4 アウトプットコンペア出力 / GRA4 インプットキャプチャ入力 / PWM 出力端子 (PWM モード / 相補 PWM モード / リセット同期 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 B4	TIOCB ₄	入出力	GRB4 アウトプットコンペア出力 / GRB4 インプットキャプチャ入力 / PWM 出力端子 (相補 PWM モード / リセット同期 PWM モード時)
	アウトプットコンペア XA4	TOCXA ₄	出力	PWM 出力端子 (相補 PWM モード / リセット同期 PWM モード時)
	アウトプットコンペア XB4	TOCXB ₄	出力	PWM 出力端子 (相補 PWM モード / リセット同期 PWM モード時)

8.1.4 レジスタ構成

ITU のレジスタ構成を表 8.3 に示します。

表 8.3 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
共通	H'FF60	タイマスタートレジスタ	TSTR	R/W	H'E0
	H'FF61	タイマシンクロレジスタ	TSNC	R/W	H'E0
	H'FF62	タイマモードレジスタ	TMDR	R/W	H'80
	H'FF63	タイマファンクションコントロールレジスタ	TFCR	R/W	H'C0
	H'FF90	タイマアウトプットマスタイネーブルレジスタ	TOER	R/W	H'FF
	H'FF91	タイマアウトプットコントロールレジスタ	TOCR	R/W	H'FF
0	H'FF64	タイマコントロールレジスタ 0	TCR0	R/W	H'80
	H'FF65	タイマ I/O コントロールレジスタ 0	TIOR0	R/W	H'88
	H'FF66	タイマインタラプトイネーブルレジスタ 0	TIER0	R/W	H'F8
	H'FF67	タイマステータスレジスタ 0	TSR0	R/(W)*2	H'F8
	H'FF68	タイマカウンタ 0H	TCNT0H	R/W	H'00
	H'FF69	タイマカウンタ 0L	TCNT0L	R/W	H'00
	H'FF6A	ジェネラルレジスタ A0H	GRA0H	R/W	H'FF
	H'FF6B	ジェネラルレジスタ A0L	GRA0L	R/W	H'FF
	H'FF6C	ジェネラルレジスタ B0H	GRB0H	R/W	H'FF
	H'FF6D	ジェネラルレジスタ B0L	GRB0L	R/W	H'FF
1	H'FF6E	タイマコントロールレジスタ 1	TCR1	R/W	H'80
	H'FF6F	タイマ I/O コントロールレジスタ 1	TIOR1	R/W	H'88
	H'FF70	タイマインタラプトイネーブルレジスタ 1	TIER1	R/W	H'F8
	H'FF71	タイマステータスレジスタ 1	TSR1	R/(W)*2	H'F8
	H'FF72	タイマカウンタ 1H	TCNT1H	R/W	H'00
	H'FF73	タイマカウンタ 1L	TCNT1L	R/W	H'00
	H'FF74	ジェネラルレジスタ A1H	GRA1H	R/W	H'FF
	H'FF75	ジェネラルレジスタ A1L	GRA1L	R/W	H'FF
	H'FF76	ジェネラルレジスタ B1H	GRB1H	R/W	H'FF
	H'FF77	ジェネラルレジスタ B1L	GRB1L	R/W	H'FF
2	H'FF78	タイマコントロールレジスタ 2	TCR2	R/W	H'80
	H'FF79	タイマ I/O コントロールレジスタ 2	TIOR2	R/W	H'88
	H'FF7A	タイマインタラプトイネーブルレジスタ 2	TIER2	R/W	H'F8
	H'FF7B	タイマステータスレジスタ 2	TSR2	R/(W)*2	H'F8
	H'FF7C	タイマカウンタ 2H	TCNT2H	R/W	H'00
	H'FF7D	タイマカウンタ 2L	TCNT2L	R/W	H'00

8. 16ビットインテグレートドタイマユニット (ITU)

チャンネル	アドレス*1	名称	略称	R/W	初期値	
2	H'FF7E	ジェネラルレジスタ A2H	GRA2H	R/W	H'FF	
	H'FF7F	ジェネラルレジスタ A2L	GRA2L	R/W	H'FF	
	H'FF80	ジェネラルレジスタ B2H	GRB2H	R/W	H'FF	
	H'FF81	ジェネラルレジスタ B2L	GRB2L	R/W	H'FF	
3	H'FF82	タイマコントロールレジスタ 3	TCR3	R/W	H'80	
	H'FF83	タイマ I/O コントロールレジスタ 3	TIOR3	R/W	H'88	
	H'FF84	タイマインタラプトイネーブルレジスタ 3	TIER3	R/W	H'F8	
	H'FF85	タイマステータスレジスタ 3	TSR3	R/(W)*2	H'F8	
	H'FF86	タイマカウンタ 3H	TCNT3H	R/W	H'00	
	H'FF87	タイマカウンタ 3L	TCNT3L	R/W	H'00	
	H'FF88	ジェネラルレジスタ A3H	GRA3H	R/W	H'FF	
	H'FF89	ジェネラルレジスタ A3L	GRA3L	R/W	H'FF	
	H'FF8A	ジェネラルレジスタ B3H	GRB3H	R/W	H'FF	
	H'FF8B	ジェネラルレジスタ B3L	GRB3L	R/W	H'FF	
	H'FF8C	バッファレジスタ A3H	BRA3H	R/W	H'FF	
	H'FF8D	バッファレジスタ A3L	BRA3L	R/W	H'FF	
	H'FF8E	バッファレジスタ B3H	BRB3H	R/W	H'FF	
	H'FF8F	バッファレジスタ B3L	BRB3L	R/W	H'FF	
	4	H'FF92	タイマコントロールレジスタ 4	TCR4	R/W	H'80
		H'FF93	タイマ I/O コントロールレジスタ 4	TIOR4	R/W	H'88
H'FF94		タイマインタラプトイネーブルレジスタ 4	TIER4	R/W	H'F8	
H'FF95		タイマステータスレジスタ 4	TSR4	R/(W)*2	H'F8	
H'FF96		タイマカウンタ 4H	TCNT4H	R/W	H'00	
H'FF97		タイマカウンタ 4L	TCNT4L	R/W	H'00	
H'FF98		ジェネラルレジスタ A4H	GRA4H	R/W	H'FF	
H'FF99		ジェネラルレジスタ A4L	GRA4L	R/W	H'FF	
H'FF9A		ジェネラルレジスタ B4H	GRB4H	R/W	H'FF	
H'FF9B		ジェネラルレジスタ B4L	GRB4L	R/W	H'FF	
H'FF9C		バッファレジスタ A4H	BRA4H	R/W	H'FF	
H'FF9D		バッファレジスタ A4L	BRA4L	R/W	H'FF	
H'FF9E		バッファレジスタ B4H	BRB4H	R/W	H'FF	
H'FF9F		バッファレジスタ B4L	BRB4L	R/W	H'FF	

【注】 *1 アドレスの下位 16 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。

8.2 各レジスタの説明

8.2.1 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0~4のTCNTの動作/停止を選択します。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	STR4	STR3	STR2	STR1	STR0
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

カウンタスタート4~0
TCNT4~TCNT0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、H'E0にイニシャライズされます。

ビット7~5: 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4: カウンタスタート4 (STR4)

タイマカウンタ4 (TCNT4) の動作/停止を選択します。

ビット4	
STR4	説明
0	TCNT4のカウンタ動作は停止 (初期値)
1	TCNT4はカウンタ動作

ビット3: カウンタスタート3 (STR3)

タイマカウンタ3 (TCNT3) の動作/停止を選択します。

ビット3	
STR3	説明
0	TCNT3のカウンタ動作は停止 (初期値)
1	TCNT3はカウンタ動作

ビット2 : カウンタスタート2 (STR2)

タイマカウンタ2 (TCNT2) の動作 / 停止を選択します。

ビット2	
STR2	説明
0	TCNT2 のカウント動作は停止 (初期値)
1	TCNT2 はカウント動作

ビット1 : カウンタスタート1 (STR1)

タイマカウンタ1 (TCNT1) の動作 / 停止を選択します。

ビット1	
STR1	説明
0	TCNT1 のカウント動作は停止 (初期値)
1	TCNT1 はカウント動作

ビット0 : カウンタスタート0 (STR0)

タイマカウンタ0 (TCNT0) の動作 / 停止を選択します。

ビット0	
STR0	説明
0	TCNT0 のカウント動作は停止 (初期値)
1	TCNT0 はカウント動作

8.2.2 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0~4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
	—	—	—	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0
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

タイマ同期4~0
チャンネル4~0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、H'E0にイニシャライズされます。

ビット7~5: 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4: タイマ同期4 (SYNC4)

チャンネル4の独立動作/同期動作を選択します。

ビット4	説明
SYNC4	
0	チャンネル4のタイムカウンタ(TCNT4)は独立動作(TCNT4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4は同期動作 TCNT4の同期プリセット/同期クリアが可能

ビット3: タイマ同期3 (SYNC3)

チャンネル3の独立動作/同期動作を選択します。

ビット3	説明
SYNC3	
0	チャンネル3のタイムカウンタ(TCNT3)は独立動作(TCNT3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3は同期動作 TCNT3の同期プリセット/同期クリアが可能

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	
SYNC2	説明
0	チャンネル2のタイマカウンタ (TCNT2) は独立動作 (TCNT2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 TCNT2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 (TCNT1) は独立動作 (TCNT1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 TCNT1の同期プリセット / 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 (TCNT0) は独立動作 (TCNT0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 TCNT0の同期プリセット / 同期クリアが可能

8.2.3 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0~4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	PWM4	PWM3	PWM2	PWM1	PWM0
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

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。

フラグディレクション
TSR2のOVFフラグセット条件を設定するビットです。

PWMモード4~0
チャンネル4~0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、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: 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、TCNT2 はアップ / ダウンカウンタ、TCLKA、TCLKB 端子がカウンタクロック入力端子となります。TCNT2 は TCLKA、TCLKB 端子の立上がり (\uparrow) / 立下がり (\downarrow) 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	\uparrow	"High"	\downarrow	"Low"	\downarrow	"Low"	\uparrow	"High"
TCLKB 端子	"Low"	\uparrow	"High"	\downarrow	"High"	\downarrow	"Low"	\uparrow

位相計数モードでは、TCR2 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、TCR2 の CCLR1、CCLR0 ビットによるカウンタクリア条件の設定、TIOR2、TIER2、TSR2 のコンペアマッチ / インพุ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)

TSR2 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	TSR2 の OVF フラグは、TCNT2 がオーバ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	TSR2 の OVF フラグは、TCNT2 がオーバフローしたときに"1"にセット

ビット4 : PWM モード4 (PWM4)

チャンネル4 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4	説明
PWM4	
0	チャンネル4 は通常動作 (初期値)
1	チャンネル4 はPWM モード

PWM4 を"1"にセットしてPWM モードにすると、TIOCA4 端子はPWM 出力端子となり、GRA4 のコンペアマッチで1出力、GRB4 のコンペアマッチで0出力となります。

TFCR の CMD1、CMD0 ビットにより相補PWM モードまたはリセット同期PWM モードが設定されているとき、本ビットの設定は無効となり、CMD1、CMD0 ビットの設定が優先されます。

ビット3 : PWMモード3 (PWM3)

チャンネル3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3	
PWM3	説明
0	チャンネル3は通常動作 (初期値)
1	チャンネル3はPWMモード

PWM3を"1"にセットしてPWMモードにすると、TIOCA3端子はPWM出力端子となり、GRA3のコンペアマッチで1出力、GRB3のコンペアマッチで0出力となります。

TFCRのCMD1、CMD0ビットにより相補PWMモードまたはリセット同期PWMモードが設定されているとき、本ビットの設定は無効となり、CMD1、CMD0ビットの設定が優先されます。

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	
PWM2	説明
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、TIOCA2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	
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モードに設定すると、TIOCA1端子はPWM出力端子となり、GRA1のコンペアマッチ1で出力、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	
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モードに設定すると、TIOCA0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。

8.2.4 タイマファンクションコントロールレジスタ (TFCR)

TFCR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3、4の相補PWMモード/リセット同期PWMモードの設定、およびバッファ動作の設定を行います。

ビット:	7	6	5	4	3	2	1	0
	—	—	CMD1	CMD0	BFB4	BFA4	BFB3	BFA3
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

バッファ動作B3、A3
チャンネル3のジェネラルレジスタ (GRB3、GRA3) とバッファレジスタ (BRB3、BRA3) をバッファ動作に設定するビットです。

バッファ動作B4、A4
チャンネル4のジェネラルレジスタ (GRB4、GRA4) とバッファレジスタ (BRB4、BRA4) をバッファ動作に設定するビットです。

コンビネーションモード1、0
チャンネル3、4を組み合わせ、相補PWMモード/リセット同期PWMモードに設定するビットです。

TFCR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、4：コンビネーションモード1、0 (CMD1、CMD0)

チャンネル3、4を通常動作させるか、相補PWMモードまたはリセット同期PWMモードで動作させるかを選択します。

ビット5	ビット4	説明
CMD1	CMD0	
0	0	チャンネル3、4は通常動作 (初期値)
	1	
1	0	チャンネル3、4を組み合わせ、相補PWMモードで動作
	1	チャンネル3、4を組み合わせ、リセット同期PWMモードで動作

相補PWMモード、およびリセット同期PWMモードの設定は、使用するTCNTを停止させた状態で行ってください。

本ビットにより、相補PWMモードまたはリセット同期PWMモードに設定した場合、TMDRのPWM4、PWM3ビットによるPWMモードの設定より優先されます。なお、相補PWMモード、リセット同期PWMモードの設定とTSNCのSYNC4、SYNC3ビットによる同期動作の設定は同時に有効となりますが、相補PWMモードを設定したときは、チャンネル3とチャンネル4を同期動作に設定(TSNCのSYNC4ビットとSYNC3ビットをともに"1"にセット)しないでください。

ビット3：バッファ動作B4 (BFB4)

チャンネル4のGRB4を通常動作とするか、GRB4とBRB4を組み合わせさせてバッファ動作とするかを設定します。

ビット3	説明
BFB4	
0	GRB4は通常動作 (初期値)
1	GRB4とBRB4はバッファ動作

ビット2：バッファ動作A4 (BFA4)

チャンネル4のGRA4を通常動作とするか、GRA4とBRA4を組み合わせさせてバッファ動作とするかを設定します。

ビット2	説明
BFA4	
0	GRA4は通常動作 (初期値)
1	GRA4とBRA4はバッファ動作

ビット1：バッファ動作 B3 (BFB3)

チャンネル3のGRB3を通常動作とするか、GRB3とBRB3を組み合わせるバッファ動作とするかを設定します。

ビット1		
BFB3	説明	
0	GRB3は通常動作	(初期値)
1	GRB3とBRB3はバッファ動作	

ビット0：バッファ動作 A3 (BFA3)

チャンネル3のGRA3を通常動作とするか、GRA3とBRA3を組み合わせるバッファ動作とするかを設定します。

ビット0		
BFA3	説明	
0	GRA3は通常動作	(初期値)
1	GRA3とBRA3はバッファ動作	

8.2.5 タイマアウトプットマスタイネーブルレジスタ (TOER)

TOER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3、4の出力設定を許可/禁止します。

ビット:	7	6	5	4	3	2	1	0
	—	—	EXB4	EXA4	EB3	EB4	EA4	EA3
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

マスタイネーブルTIOCA₃、TIOCB₃、TIOCA₄、TIOCB₄
 TIOCA₃、TIOCB₃、TIOCA₄、TIOCB₄端子の出力設定を許可/禁止するビットです。

マスタイネーブルTOCXA₄、TOCXB₄
 TOCXA₄、TOCXB₄端子の出力設定を許可/禁止するビットです。

リザーブビット

TOER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。

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：マスタイネーブルTOCXB₄ (EXB4)

TOCXB₄端子のITU出力を許可/禁止します。

ビット5	説明
EXB4	
0	TFCRの設定にかかわらずTOCXB ₄ 端子の出力は禁止 (TOCXB ₄ 端子は入出力ポートとして動作) XTGD="0"の状態、チャンネル1のインプットキャプチャAが発生したとき"0"にクリア
1	TFCRの設定に従いTOCXB ₄ 端子の出力は許可 (初期値)

ビット4：マスタイネーブル TOCXA₄ (EXA4)TOCXA₄端子の ITU 出力を許可 / 禁止します。

ビット4	
EXA4	説明
0	TFCR の設定にかかわらず TOCXA ₄ 端子の出力は禁止 (TOCXA ₄ 端子は入出力ポートとして動作) XTGD = "0"の状態、チャンネル1のインプットキャプチャ A が発生したとき"0"にクリア
1	TFCR の設定に従い TOCXA ₄ 端子の出力は許可 (初期値)

ビット3：マスタイネーブル TIOCB₃ (EB3)TIOCB₃端子の ITU 出力を許可 / 禁止します。

ビット3	
EB3	説明
0	TIOR3、TFCR の設定にかかわらず TIOCB ₃ 端子の出力は禁止 (TIOCB ₃ 端子は出力ポートとして動作) XTGD = "0"の状態、チャンネル1のインプットキャプチャ A が発生したとき"0"にクリア
1	TIOR3、TFCR の設定に従い TIOCB ₃ 端子の出力は許可 (初期値)

ビット2：マスタイネーブル TIOCB₄ (EB4)TIOCB₄端子の ITU 出力を許可 / 禁止します。

ビット2	
EB4	説明
0	TIOR4、TFCR の設定にかかわらず TIOCB ₄ 端子の出力は禁止 (TIOCB ₄ 端子は入出力ポートとして動作) XTGD = "0"の状態、チャンネル1のインプットキャプチャ A が発生したとき"0"にクリア
1	TIOR4、TFCR の設定に従い TIOCB ₄ 端子の出力は許可 (初期値)

ビット1：マスタイネーブル TIOCA₄ (EA4)TIOCA₄端子の ITU 出力を許可 / 禁止します。

ビット1	
EA4	説明
0	TIOR4、TMDR、TFCR の設定にかかわらず TIOCA ₄ 端子の出力は禁止 (TIOCA ₄ 端子は入出力ポートとして動作) XTGD = "0"の状態、チャンネル1のインプットキャプチャ A が発生したとき"0"にクリア
1	TIOR4、TMDR、TFCR の設定に従い TIOCA ₄ 端子の出力は許可 (初期値)

ビット0：マスタイネーブル TIOCA₃ (EA3)TIOCA₃端子の ITU 出力を許可 / 禁止します。

ビット0	
EA3	説明
0	TIOR3、TMDR、TFCR の設定にかかわらず TIOCA ₃ 端子の出力は禁止 (TIOCA ₃ 端子は入出力ポートとして動作) XTGD = "0"の状態、チャンネル1のインプットキャプチャ A が発生したとき"0"にクリア
1	TIOR3、TMDR、TFCR の設定に従い TIOCA ₃ 端子の出力は許可 (初期値)

8.2.6 タイマアウトプットコントロールレジスタ (TOCR)

TOCR は、8ビットのリード/ライト可能なレジスタで、相補 PWM モード/リセット同期 PWM モード出力の外部トリガによる禁止または出力レベル反転を行います。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	XTGD	—	—	OLS4	OLS3
初期値:	1	1	1	1	1	1	1	1
R/W :	—	—	—	R/W	—	—	R/W	R/W

リザーブビット

外部トリガディスエーブル
相補PWMモード/リセット同期PWMモード出力の
外部からのトリガによる出力禁止を設定するビットです。

リザーブビット

出力レベルセレクト3、4
相補PWMモード/リセット同期
PWMモードの出力レベルを選択
するビットです。

XTGD、OLS4 および OLS3 ビットの設定は、リセット同期 PWM モードまたは相補 PWM モードを設定しているときのみ有効となります。他の出力状態では、これらのビットの設定は無効です。

TOCR 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。

ビット7～5：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4：外部トリガディスエーブル (XTGD)

リセット同期 PWM モード/相補 PWM モード時の ITU 出力の外部トリガによる禁止を設定します。

ビット4	説明
XTGD	
0	リセット同期 PWM モード/相補 PWM モード時、チャンネル1のインプットキャプチャ A 信号を外部トリガとして使用 外部トリガの発生時、TOER のビット 5～0 が"0"にクリアされ、ITU 出力は禁止
1	外部トリガを禁止 (初期値)

ビット3、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：出力レベルセレクト4 (OLS4)

リセット同期 PWM モード / 相補 PWM モード出力のレベルを選択します。

ビット1		
OLS4	説明	
0	TIOCA ₃ 、TIOCA ₄ 、TIOCB ₄ 端子は反転出力	
1	TIOCA ₃ 、TIOCA ₄ 、TIOCB ₄ 端子は直接出力	(初期値)

ビット0：出力レベルセレクト3 (OLS3)

リセット同期 PWM モード / 相補 PWM モード出力のレベルを選択します。

ビット0		
OLS3	説明	
0	TIOCB ₃ 、TOCXA ₄ 、TOCXB ₄ 端子は反転出力	
1	TIOCB ₃ 、TOCXA ₄ 、TOCXB ₄ 端子は直接出力	(初期値)

8.2.7 タイマカウンタ (TCNT)

TCNTは16ビットのカウンタです。ITUには、各チャンネル1本、計5本のTCNTがあります。

チャンネル	略称	機能
0	TCNT0	アップカウンタ
1	TCNT1	
2	TCNT2	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
3	TCNT3	相補PWMモード : アップ/ダウンカウンタ 上記以外 : アップカウンタ
4	TCNT4	

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

TCNT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、TCRのTPSC2~TPSC0ビットにより選択します。

TCNT0、TCNT1はアップカウント動作を行います。TCNT2は位相計数モード時、またTCNT3、TCNT4は相補PWM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。

TCNT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)。

TCNTがオーバフロー(H'FFFF H'0000)すると、対応するチャンネルのTSRのOVFフラグが"1"にセットされます。

TCNTがアンダフロー(H'0000 H'FFFF)すると、対応するチャンネルのTSRのOVFフラグが"1"にセットされます。

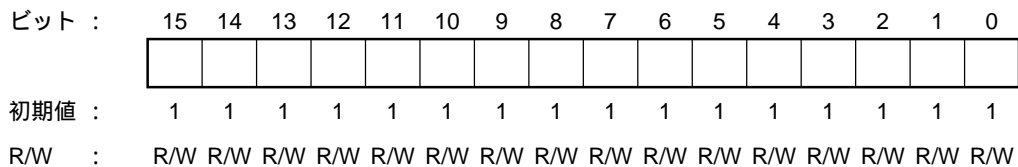
TCNTはCPUと内部16ビットバスで接続されており、ワード/バイト単位のリード/ライトが可能です。

TCNT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。

8.2.8 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。ITUには、各チャンネル2本、計10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	
3	GRA3、GRB3	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。 バッファレジスタ (BRA、BRB) と組み合わせることにより、 バッファ動作設定可能
4	GRA4、GRB4	



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。機能の切換えは、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TCNT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、TSR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、TCNTの値を格納します。このとき対応するTSR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選択はTIORにより行います。

PWMモード、相補PWMモード、またはリセット同期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。

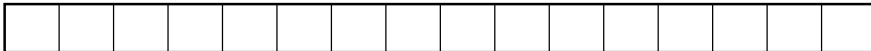
GRはCPUと内部16ビットバスで接続されており、ワード/バイト単位のリード/ライトが可能です。

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。

8.2.9 バッファレジスタ A、B (BRA、BRB)

BRは、16ビットのレジスタです。ITUには、チャンネル3、4に各2本、計4本のバッファレジスタがあります。

チャンネル	略称	機能
3	BRA3、BRB3	バッファ動作時に使用 ・対応するGRA、GRBがアウトプットコンペアレジスタのときアウトプットコンペアバッファレジスタとして機能し、コンペアマッチによりBRA、BRBの値をGRA、GRBに自動転送可能
4	BRA4、BRB4	・対応するGRA、GRBがインプットキャプチャレジスタのときインプットキャプチャバッファレジスタとして機能し、インプットキャプチャ時それまで格納されていたGRA、GRBの値をBRA、BRBに自動転送可能

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

BRは、16ビットのリード/ライト可能なレジスタで、バッファ動作設定時に使用されます。バッファ動作の設定はTFCRのBFB4、BFA4、BFB3、およびBFA3ビットにより独立に行うことができます。

BRはGRと対になって機能し、GRがアウトプットコンペアレジスタに設定されているときはアウトプットコンペアバッファレジスタとして、またGRがインプットキャプチャレジスタとして設定されているときはインプットキャプチャバッファレジスタとして機能します。

BR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。

BRは、リセット、またはスタンバイモード時にH'FFFFにイニシャライズされます。

8.2.10 タイマコントロールレジスタ (TCR)

TCRは8ビットのレジスタです。ITUには、各チャンネル1本、計5本のTCRがあります。

チャンネル	略称	機能
0	TCR0	TCRはTCNTの制御を行います。
1	TCR1	各チャンネルのTCRは同一の機能をもっています。 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。
2	TCR2	
3	TCR3	
4	TCR4	



TCRは8ビットのリード/ライト可能なレジスタで、TCNTのカウンタクロックの選択、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。

TCR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)

TCNT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	TCNT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TCNT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TCNT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TCNT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)
	1	立下がりエッジでカウント
1		立上がり/立下がり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、TCR2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: タイマプリスケアラ2~0 (TPSC2~TPSC0)

TCNTのカウンタ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

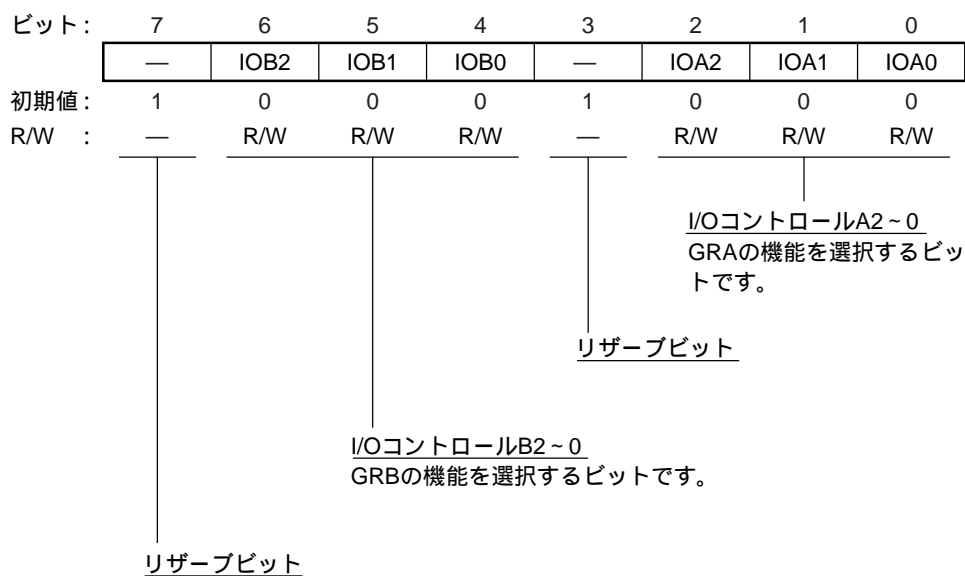
TPSC2 ビットを"0"にクリアして内部クロックを選択した場合、クロックの立下がりエッジでカウントされます。また、TPSC2 ビットを"1"にセットして外部クロックを選択した場合、カウントエッジはCKEG1、CKEG0ビットの設定に従います。

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") TCR2のTPSC2~TPSC0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。

8.2.11 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。ITU には、各チャンネル 1 本、計 5 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。 PWM モード時、一部機能が異なります。 チャンネル 3、4 を相補 PWM モード / リセット同期 PWM モードに設定したとき、TIOR3、TIOR4 の設定は無効となります。
1	TIOR1	
2	TIOR2	
3	TIOR3	
4	TIOR4	



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 TIOCA、TIOCB 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	コンペアマッチによる端子出力禁止 (初期値)
		1	プットコンペア	GRBのコンペアマッチで0出力*1
	1	0	レジスタ	GRBのコンペアマッチで1出力*1
		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、*2
1	0	0	GRBはインプ	立上がりエッジでGRBへインプットキャプチャ
		1	ットキャプチャ	立下がりエッジでGRBへインプットキャプチャ
	1	0	レジスタ	立上がり / 立下がりの両エッジでGRBへインプット 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0です。

*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		
IOA2	IOA1	IOA0	説明	
0	0	0	GRA はアウト	コンペアマッチによる端子出力禁止 (初期値)
		1	プットコンペア	GRA のコンペアマッチで 0 出力*1
	1	0	レジスタ	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 (チャンネル2のみ1出力)*1*2
1	0	0	GRA はインプ	立上がりエッジで GRA ヘインプットキャプチャ
		1	ットキャプチャ	立下がりエッジで GRA ヘインプットキャプチャ
	1	0	レジスタ	立上がり / 立下がりの両エッジで GRA ヘインプット
		1		キャプチャ

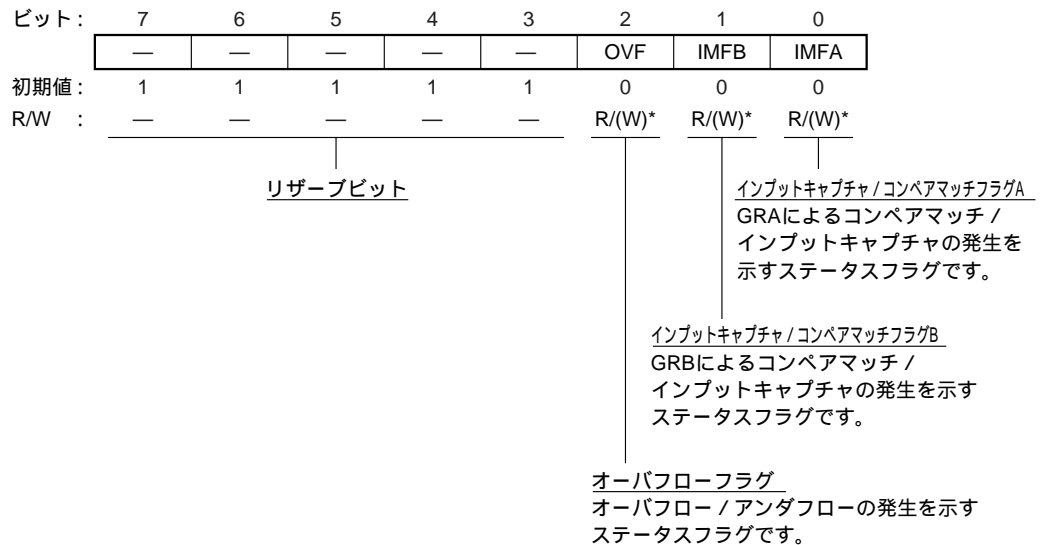
【注】 *1 リセット後、最初のコンペアマッチが発生するまでの出力値は 0 です。

*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。

8.2.12 タイマステータスレジスタ (TSR)

TSRは8ビットのレジスタです。ITUには、各チャンネル1本、計5本のTSRがあります。

チャンネル	略称	機能
0	TSR0	インプットキャプチャ/コンペアマッチやオーバフローのステータスを示します。
1	TSR1	
2	TSR2	
3	TSR3	
4	TSR4	



【注】 * フラグをクリアするための"0"ライトのみ可能です。

TSRは8ビットのリード/ライト可能なレジスタで、TCNTのオーバフロー/アンダフローの発生、およびGRA、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ます。

これらのフラグは割り込み要因であり、TIERの対応するビットにより割り込みが許可されていれば、CPUに割り込みを要求します。

TSR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：オーバーフローフラグ (OVF)

TCNTのオーバーフロー/アンダフローの発生を示すステータスフラグです。

ビット2	
OVF	説明
0	[クリア条件] (初期値) OVF = "1"の状態、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*

- 【注】 * TCNTのアンダフローは、TCNTがアップ/ダウンカウンタとして機能している場合に発生します。したがって、次の場合のみアンダフローが発生することがあります。
- (1) 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")
 - (2) チャンネル3、4が相補PWMモードに設定されているとき (TFCRのCMD1="1"、CMD0="0")

ビット1：インプットキャプチャ/コンペアマッチフラグB (IMFB)

GRB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	
IMFB	説明
0	[クリア条件] (初期値) IMFB = "1"の状態、IMFBフラグをリードした後、IMFBフラグに"0"をライトしたとき
1	[セット条件] (1) GRB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、TCNT = GRBになったとき (2) GRB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GRB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A (IMFA)

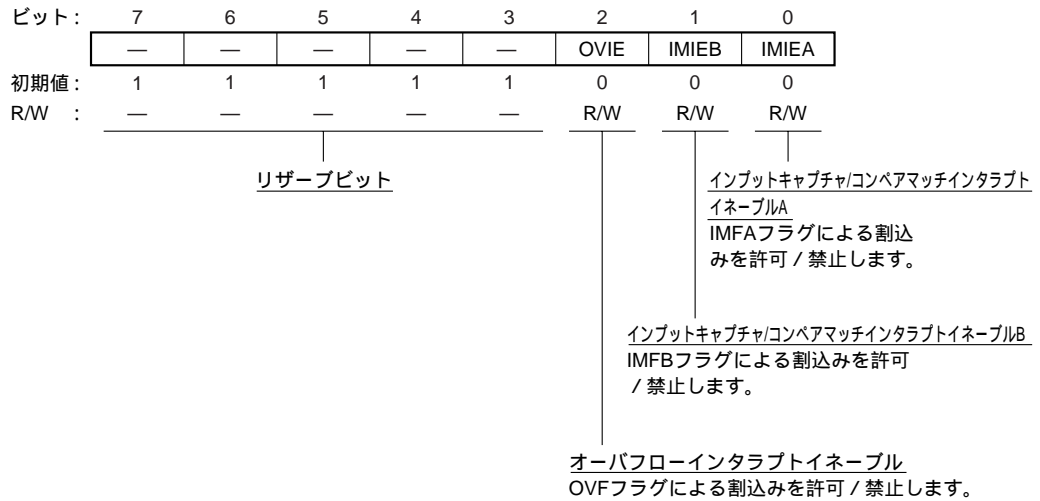
GRA のコンペアマッチまたはインプットキャプチャの発生を示すステータスフラグです。

ビット0	
IMFA	説明
0	[クリア条件] (初期値) IMFA="1"の状態、IMFAフラグをリードした後、IMFAフラグに"0"をライトしたとき
1	[セット条件] (1) GRA 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、TCNT = GRA になったとき (2) GRA 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 GRA に転送されたとき

8.2.13 タイマインタラプトイネーブルレジスタ (TIER)

TIERは8ビットのレジスタです。ITUには、各チャンネル1本、計5本のTIERがあります。

チャンネル	略称	機能
0	TIER0	割込み要求の許可 / 禁止を制御します。
1	TIER1	
2	TIER2	
3	TIER3	
4	TIER4	



TIERは8ビットのリード/ライト可能なレジスタで、オーバーフロー割込み要求、GRのコンペアマッチ/インプットキャプチャ割込み要求の許可/禁止を制御します。

TIER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: オーバフローインタラプトイネーブル (OVIE)

TSRのOVFフラグが"1"にセットされたとき、OVFフラグによる割込み要求を許可/禁止します。

ビット2	説明	
OVIE		
0	OVFフラグによる割込み (OVI) 要求を禁止	(初期値)
1	OVFフラグによる割込み (OVI) 要求を許可	

ビット1：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B (IMIEB)

TSRのIMFBフラグが"1"にセットされたとき、IMFBによる割込み要求を許可/禁止します。

ビット1	
IMIEB	説明
0	IMFBフラグによる割込み (IMIB) 要求を禁止 (初期値)
1	IMFBフラグによる割込み (IMIB) 要求を許可

ビット0：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A (IMIEA)

TSRのIMFAフラグが"1"にセットされたとき、IMFAによる割込み要求を許可/禁止します。

ビット0	
IMIEA	説明
0	IMFAフラグによる割込み (IMIA) 要求を禁止 (初期値)
1	IMFAフラグによる割込み (IMIA) 要求を許可

8.3 CPU とのインタフェース

8.3.1 16ビットアクセス可能なレジスタ

TCNT、GRA、GRB、およびBRA、BRB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。

TCNT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8.6、図8.7に示します。

また、TCNTH、TCNTL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8.8、図8.9、図8.10、図8.11に示します。

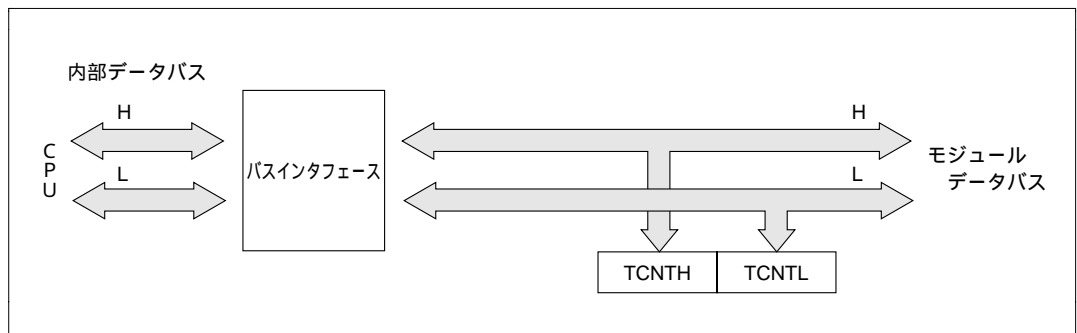


図 8.6 TCNT のアクセス動作 [CPU TCNT (ワード)]

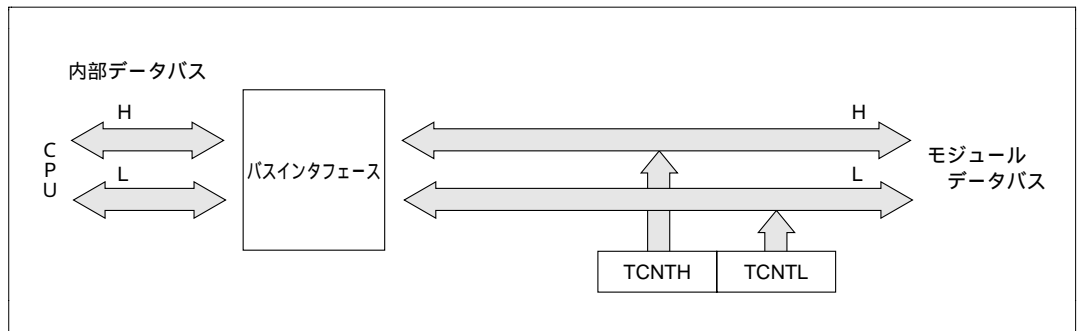


図 8.7 TCNT のアクセス動作 [TCNT CPU (ワード)]

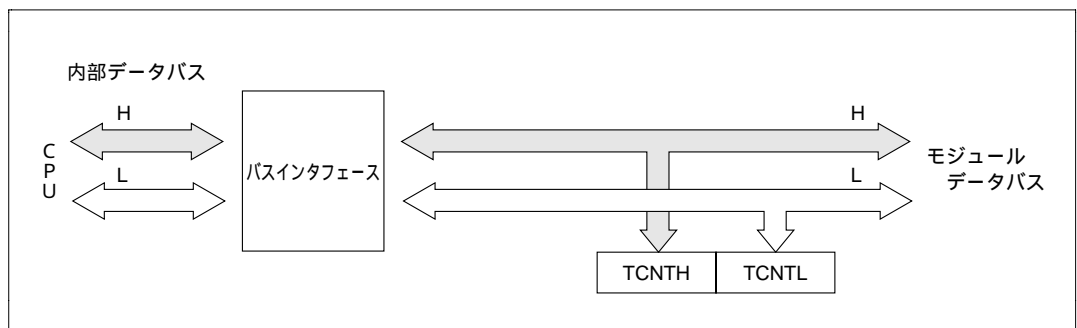


図 8.8 TCNT のアクセス動作 [CPU TCNT (上位バイト)]

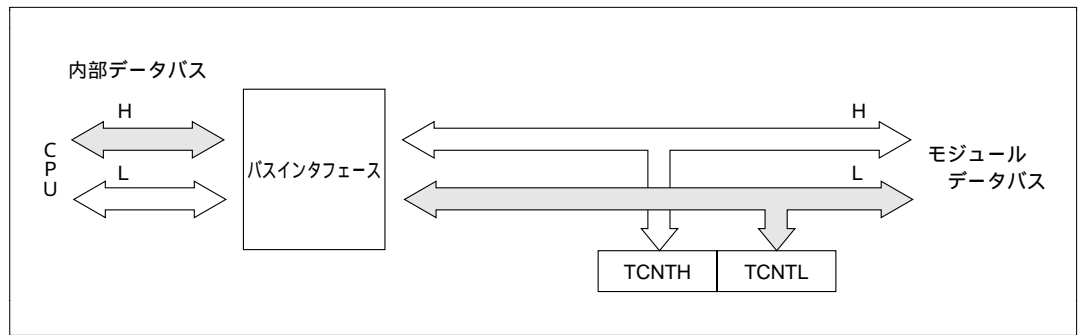


図 8.9 TCNT のアクセス動作 [CPU TCNT (下位バイト)]

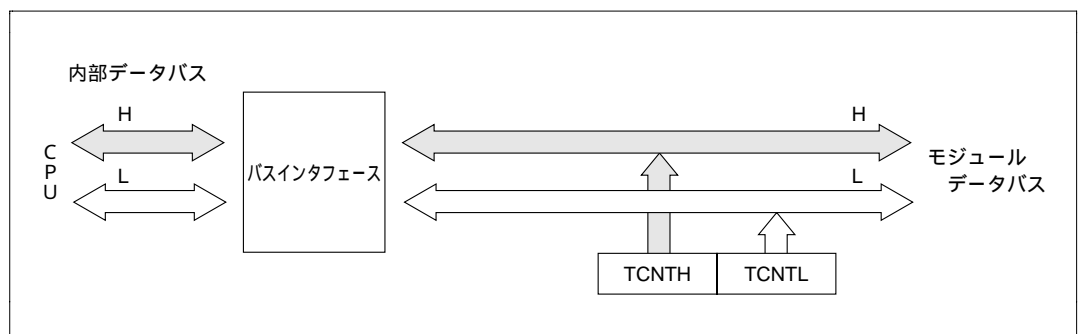


図 8.10 TCNT のアクセス動作 [TCNT CPU (上位バイト)]

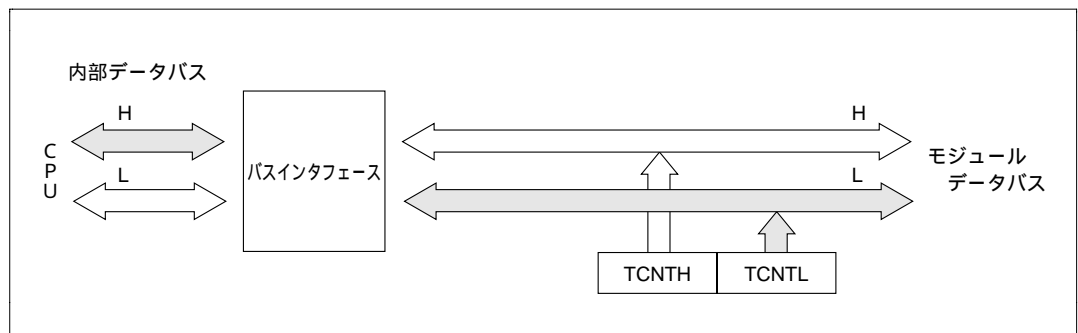


図 8.11 TCNT のアクセス動作 [TCNT CPU (下位バイト)]

8.3.2 8ビットアクセスのレジスタ

TCNT、GR、BR 以外のレジスタは 8 ビットレジスタです。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。

TCR 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 8.12、図 8.13 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。

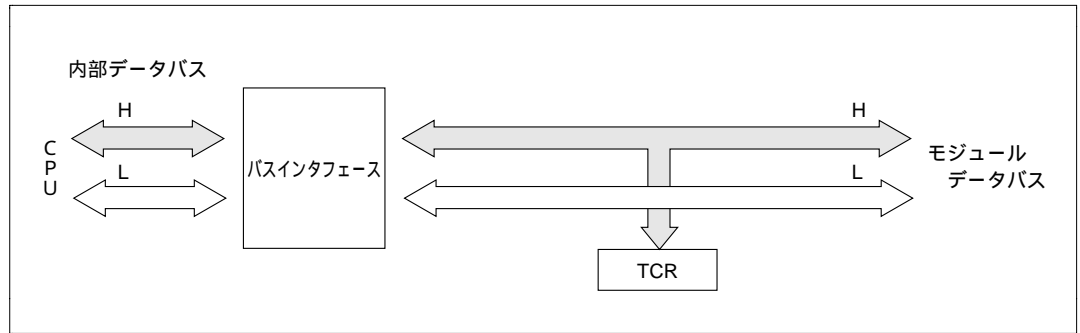


図 8.12 TCR のアクセス動作 [CPU TCR]

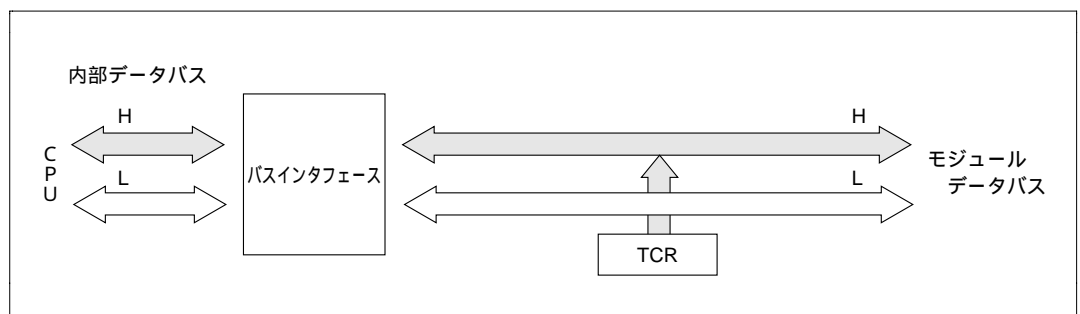


図 8.13 TCR のアクセス動作 [TCR CPU]

8.4 動作説明

8.4.1 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、TCNT と GR があります。TCNT 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 TCNT 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 TCNT を書き換えると他のチャンネルの TCNT 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 TCR の CCLR1、CCLR0 ビットの設定により、TCNT 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) リセット同期 PWM モード

チャンネル 3、4 を組み合わせて、正相と逆相の PWM 波形を 3 相出力します (3 相の PWM 波形は一方の変化点が共通となる関係になります)。リセット同期 PWM モードに設定すると GRA3、GRB3、GRA4、GRB4 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。また、TIOCA₃、TIOCB₃、TIOCA₄、TOCXA₄、TIOCB₄、TOCXB₄ 端子は自動的に PWM 出力端子となり、TCNT3 はアップカウント動作を行います。TCNT4 は独立に動作します (ただし、GRA4、GRB4 は TCNT4 とは切り離されています)。

(5) 相補 PWM モード

チャンネル 3、4 を組み合わせて、正相と逆相がノンオーバーラップの関係にある PWM 波形を 3 相出力します。相補 PWM モードに設定すると GRA3、GRB3、GRA4、GRB4 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。また、TIOCA₃、TIOCB₃、TIOCA₄、TOCXA₄、TIOCB₄、TOCXB₄ 端子は自動的に PWM 出力端子となり、TCNT3、TCNT4 はアップ/ダウンカウント動作を行います。

(6) 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、TCNT2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 TCNT2 はアップ/ダウンカウント動作を行います。

(7) バッファ動作

(1) GR がアウトプットコンペアレジスタの場合

コンペアマッチが発生すると当該チャンネルの BR の値が、GR に転送されます。

(2) GR がインプットキャプチャレジスタの場合

インプットキャプチャが発生すると TCNT の値を GR に転送すると同時に、それまで格納されていた GR の値を BR に転送します。

(3) 相補 PWM モードの場合

TCNT3、TCNT4 のカウント方向が変化すると BR の値が、GR に転送されます。

(4) リセット同期 PWM モードの場合

GRA3 のコンペアマッチにより BR の値が、GR に転送されます。

8.4.2 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~STR4 ビットを"1"にセットすると、対応するチャンネルの TCNT 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 8.14 に示します。

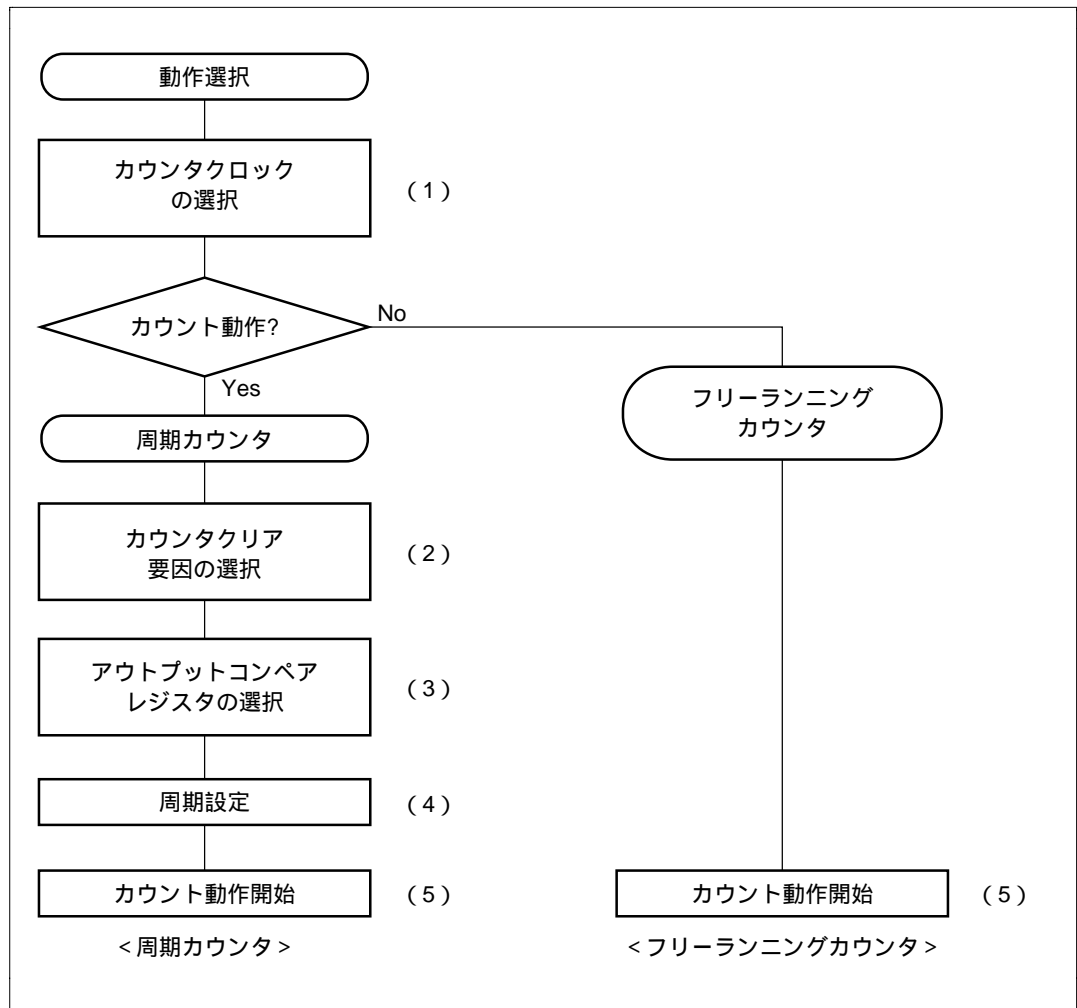


図 8.14 カウント動作設定手順例

- (1) TCR 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、TCR 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) 周期カウンタ動作の場合 TCR の CCLR1、CCLR0 ビットで TCNT 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 に周期カウンタの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タ動作と周期カウンタ動作

ITU チャンネル 0 ~ 4 のカウンタ (TCNT) 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タ動作を開始します。TCNT がオーバフロー (H'FFFF H'0000) すると TSR の OVF フラグが "1" にセットされます。このとき、対応する TIER の

OVIE ビットが"1"ならば、CPU に割込みを要求します。TCNT はオーバフロー後、H'0000 から再びアップカウント動作を続けます。

フリーランニングカウンタの動作を図 8.15 に示します。

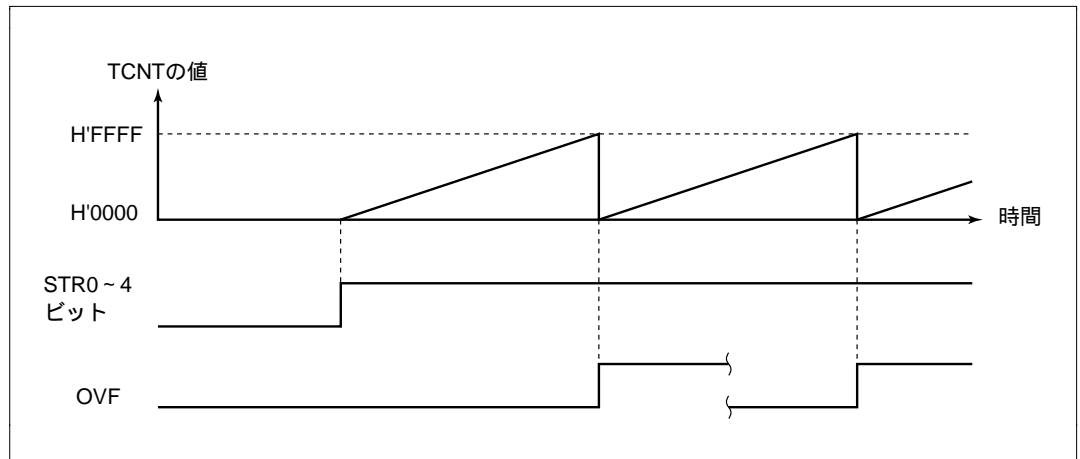


図 8.15 フリーランニングカウンタの動作

TCNT のクリア要因にコンペアマッチを選択したときは、当該チャンネルの TCNT 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、TCR 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 TSR の IMFA/IMFB フラグが"1"にセットされ、TCNT は H'0000 にクリアされます。

このとき、対応する TIER の IMIEA/IMIEB ビットが"1"ならば、CPU に割込みを要求します。TCNT 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 8.16 に示します。

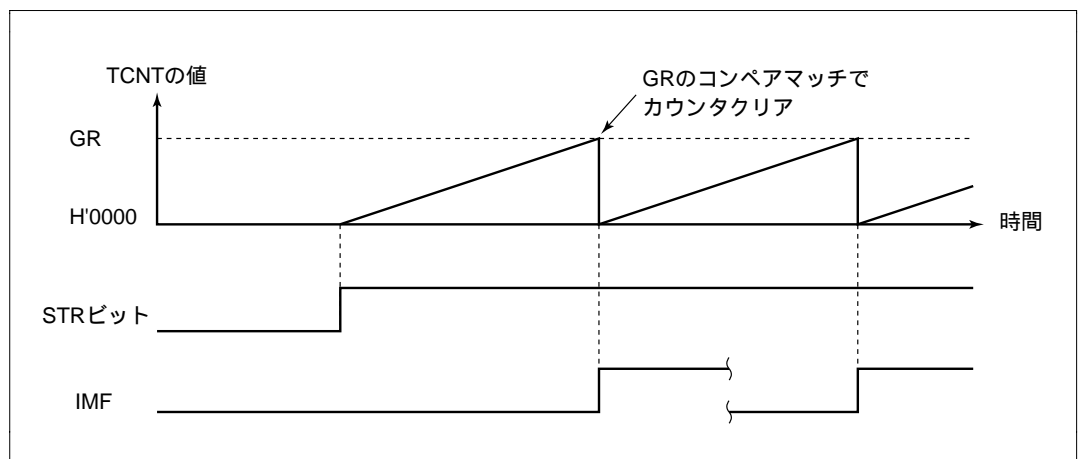


図 8.16 周期カウンタの動作

(c) TCNT のカウントタイミング

(1) 内部クロック動作の場合

TCR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 8.17 に示します。

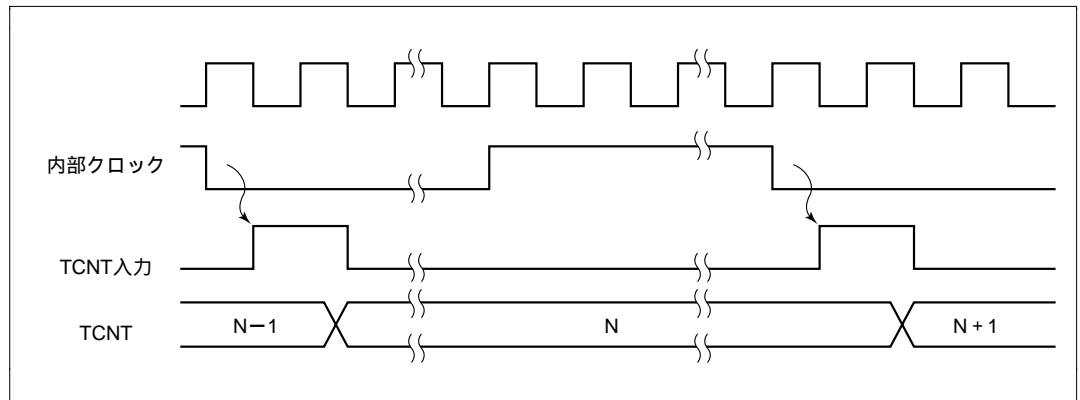


図 8.17 内部クロック動作時のカウントタイミング

(2) 外部クロック動作の場合

TCR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 8.18 に示します。

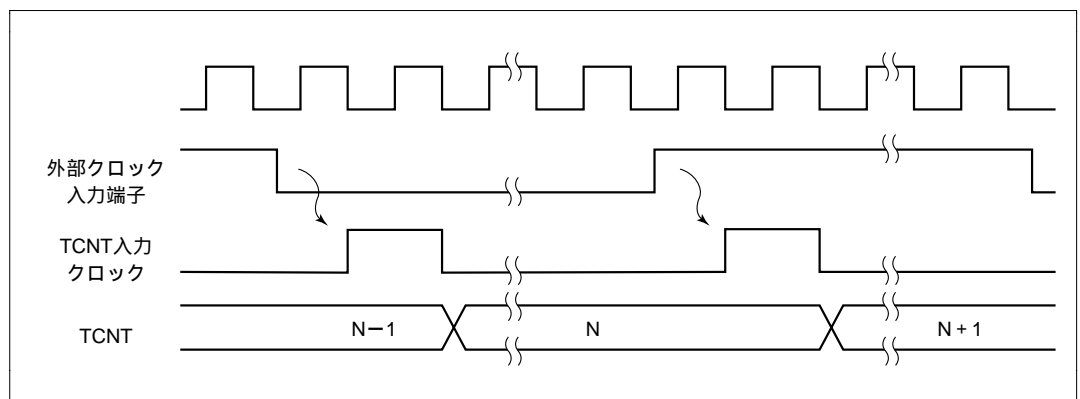


図 8.18 外部クロック動作時のカウントタイミング (両エッジ検出の場合)

(2) コンペアマッチによる波形出力機能

ITU チャンネル0、1、3、4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 8.19 に示します。

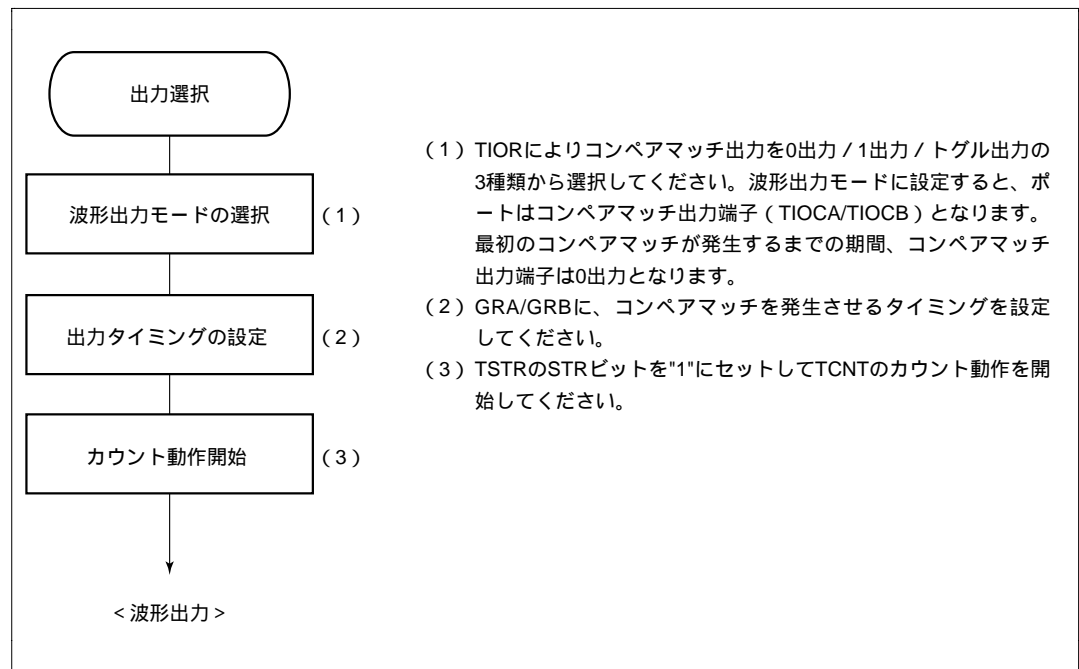


図 8.19 コンペアマッチによる波形出力動作例

(b) 波形出力動作例

0出力/1出力の例を図 8.20 に示します。

TCNTを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。

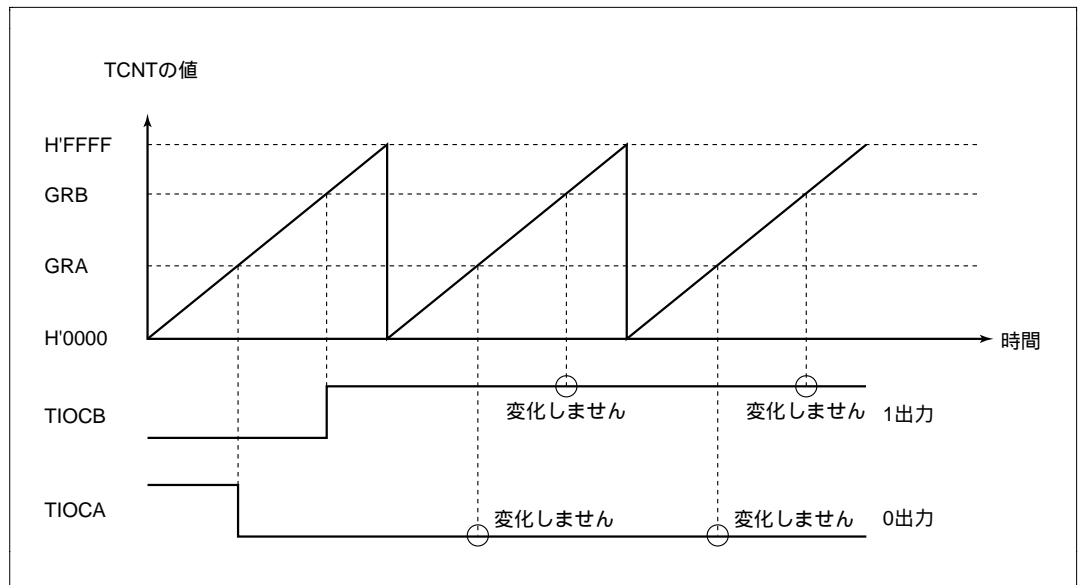


図 8.20 0 出力、1 出力の動作例

トグル出力の例を図 8.21 に示します。

TCNT 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。

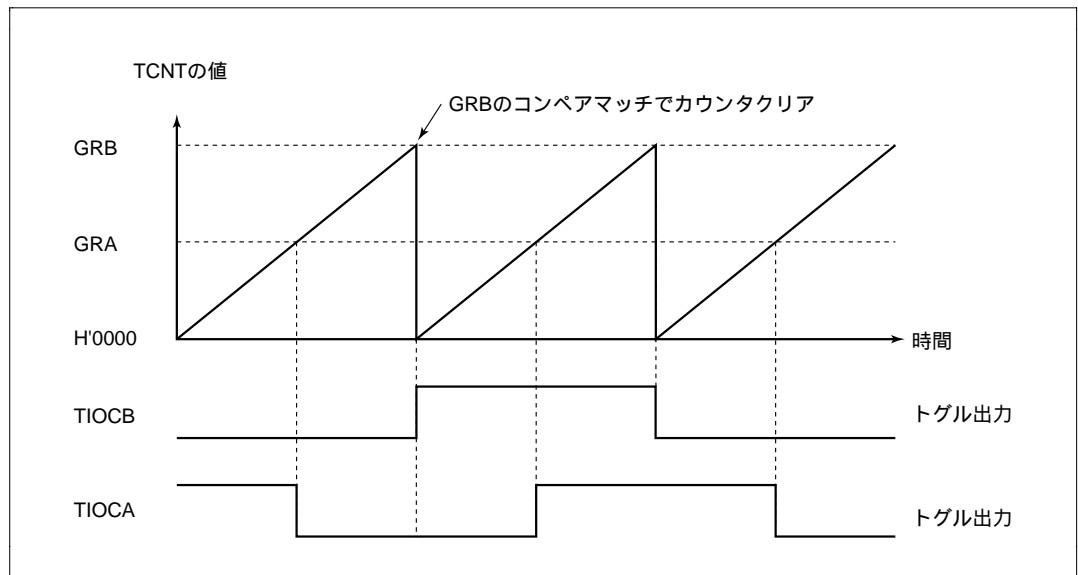


図 8.21 トグル出力の動作例

(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、TCNT と GR が一致した最後のステート (TCNT 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。TCNT と GR が一致した後、TCNT 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 8.22 に示します。

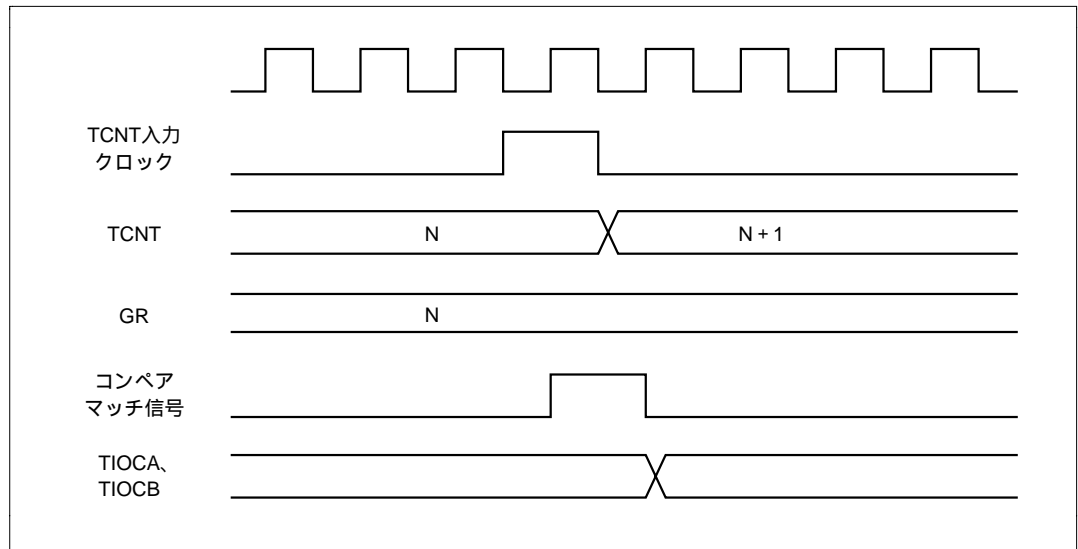


図 8.22 アウトプットコンペア出力タイミング

(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 TCNT 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 8.23 に示します。

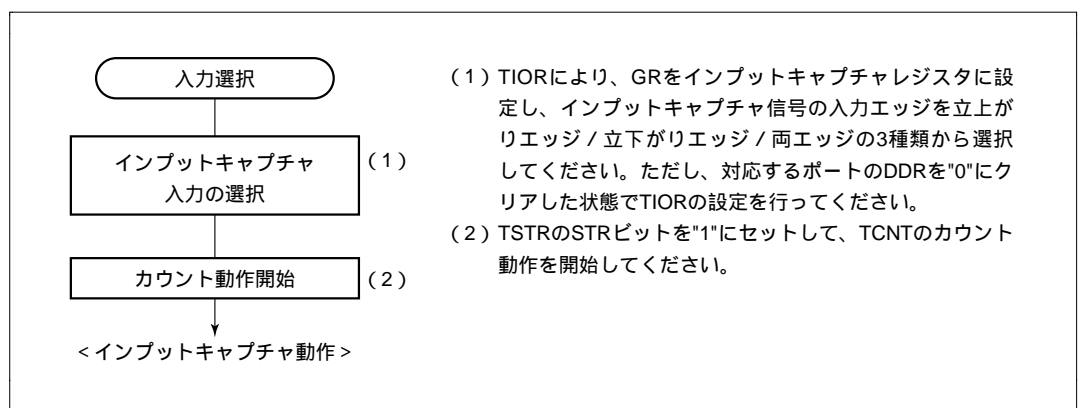


図 8.23 インพุットキャプチャ動作の設定手順例

(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 8.24 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 端子のインプットキャプチャ入力エッジは立下がりエッジを選択し、TCNT は GRB のインプ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。

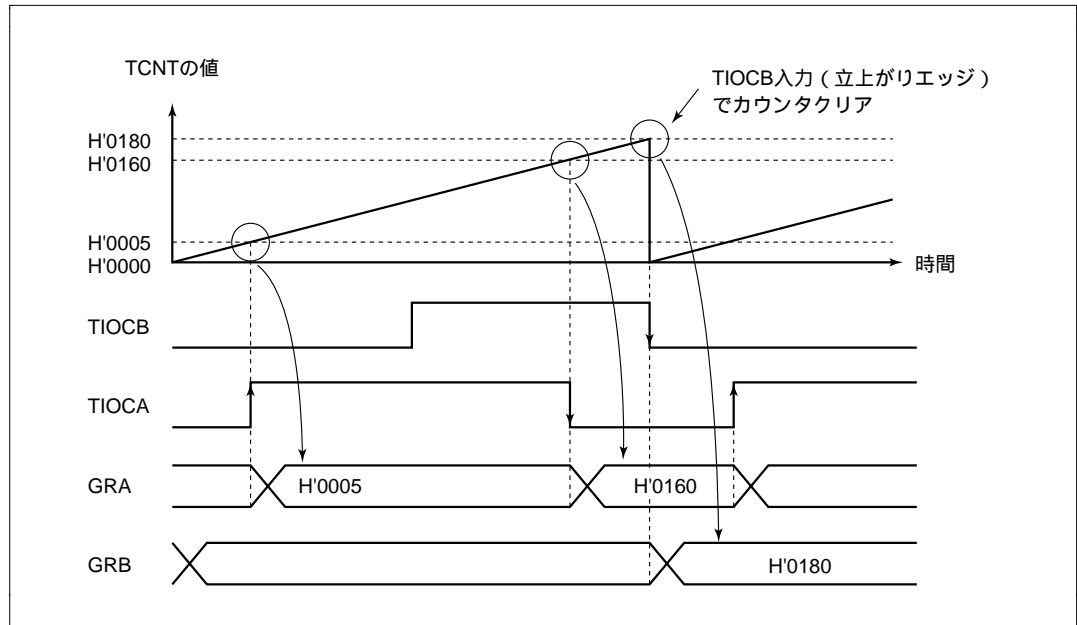


図 8.24 インプットキャプチャ動作例

(c) インプットキャプチャ信号タイミング

インプ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 8.25 に示します。

インプ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。

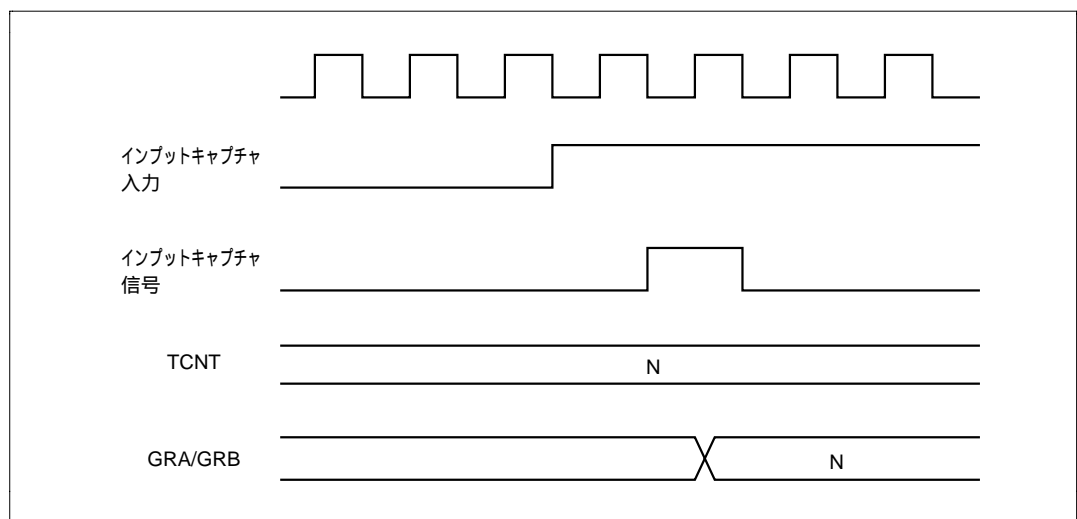


図 8.25 インプットキャプチャ入力信号タイミング

8.4.3 同期動作

同期動作は、複数の TCNT 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、TCR の設定により複数の TCNT 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0~4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 8.26 に示します。

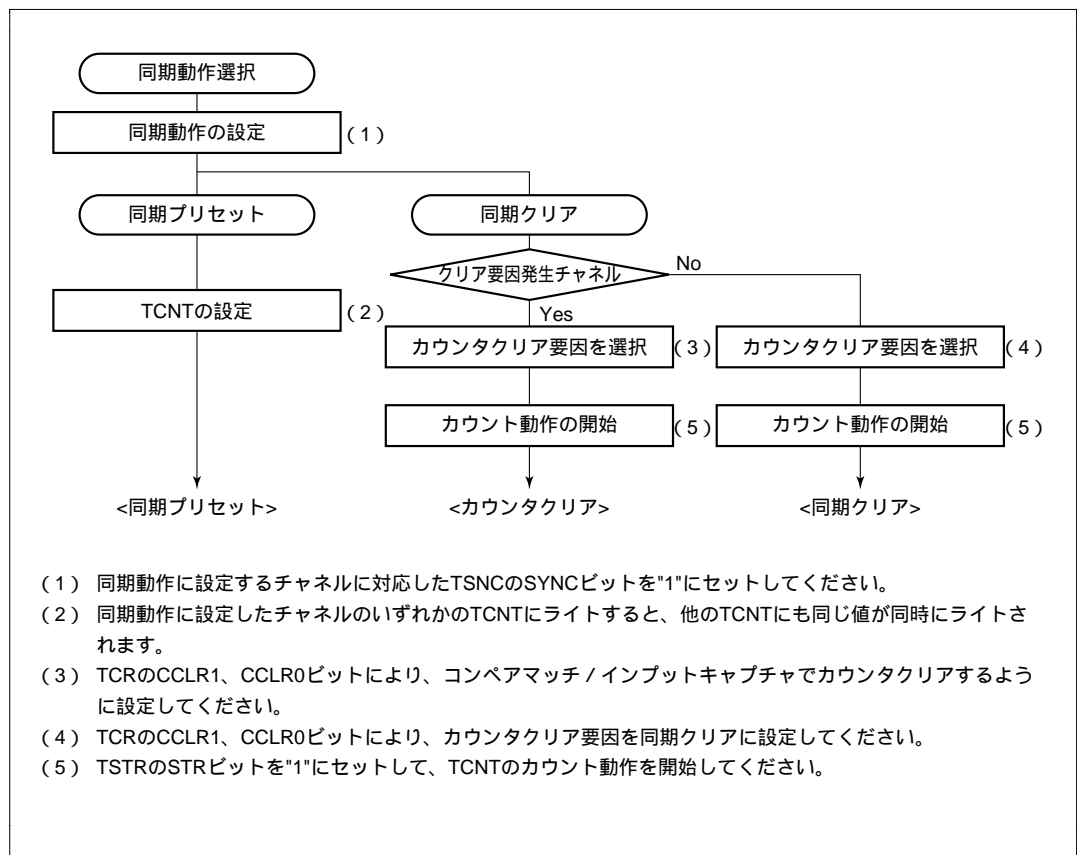


図 8.26 同期モードの設定手順例

(2) 同期動作例

同期動作例を図 8.27 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TCNT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TIOCA₀、TIOCA₁、TIOCA₂端子から出力します。

PWMモードについては「8.4.4 PWMモード」を参照してください。

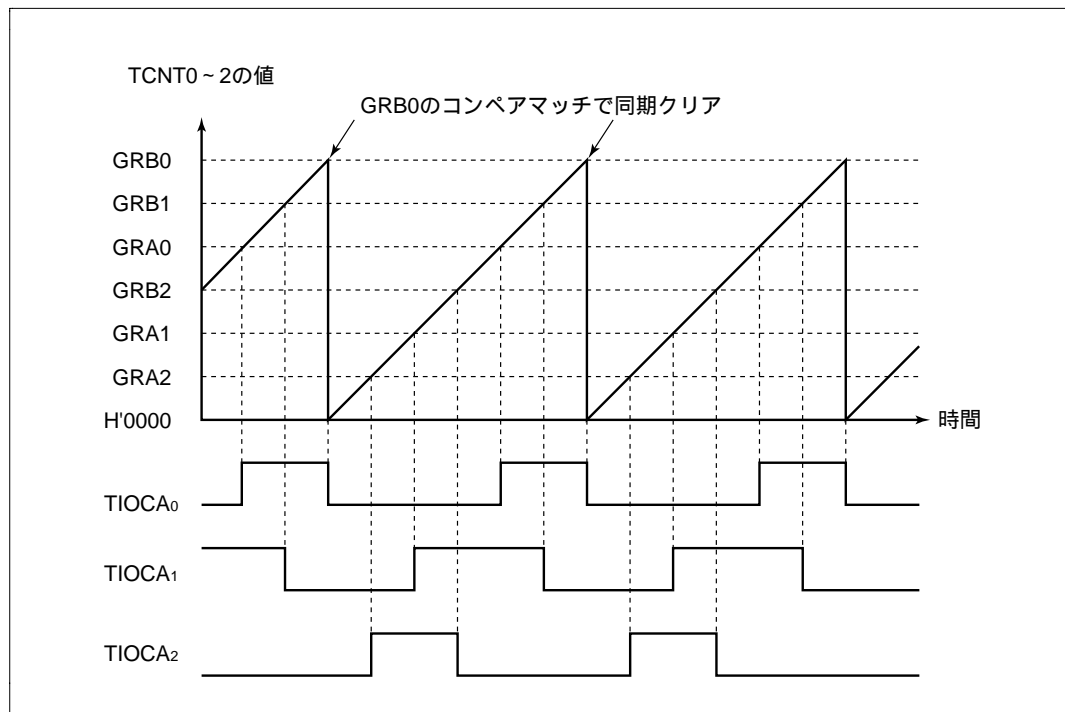


図 8.27 同期動作例

8.4.4 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TCNT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0~4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8.4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 8.4 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 ₀	GRA0	GRB0
1	TIOCA ₁	GRA1	GRB1
2	TIOCA ₂	GRA2	GRB2
3	TIOCA ₃	GRA3	GRB3
4	TIOCA ₄	GRA4	GRB4

(1) PWM モードの設定手順例

PWM モードの設定手順例を図 8.28 に示します。

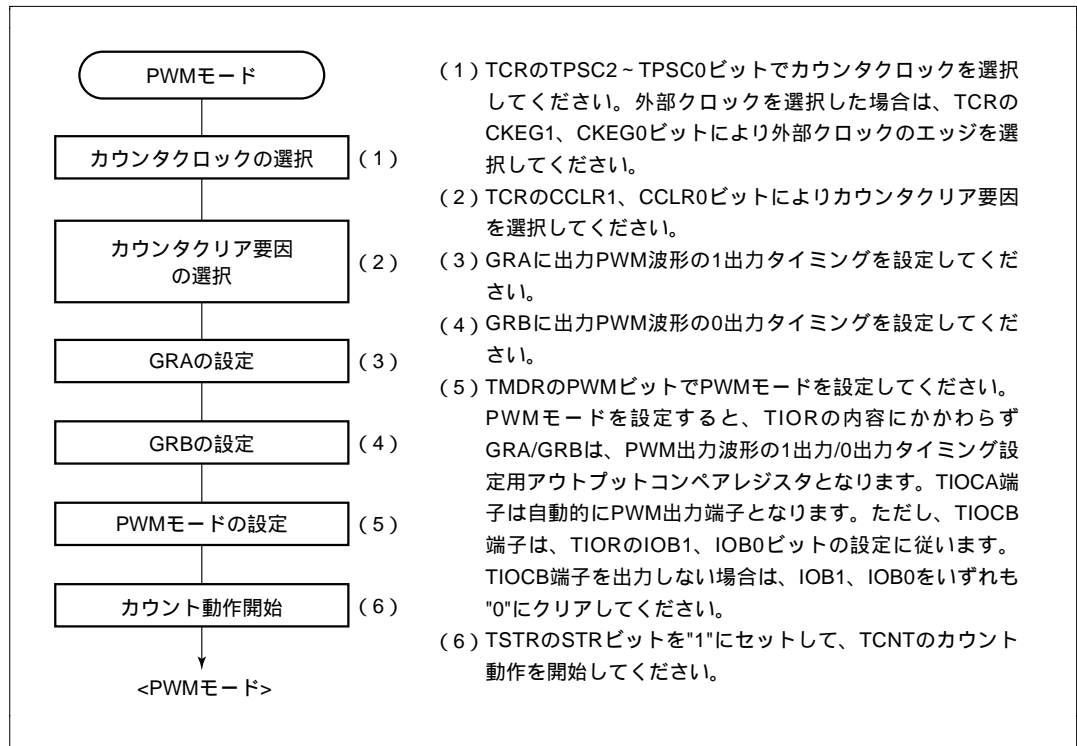


図 8.28 PWM モードの設定手順例

(2) PWM モードの動作例

PWM モードの動作例を図 8.29 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。

TCNT 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ト動作も使用できます。

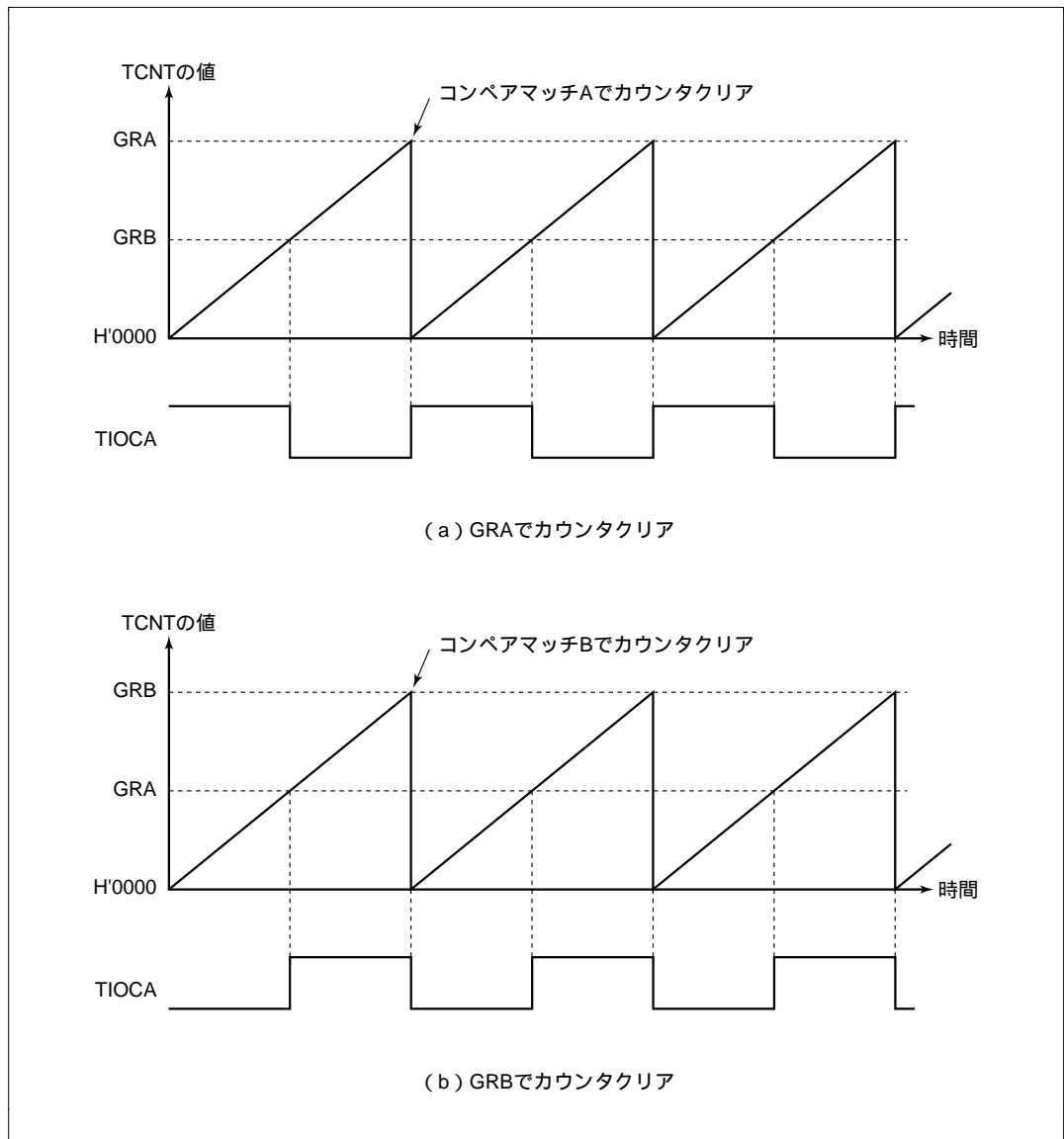


図 8.29 PWM モードの動作例 (1)

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 8.30 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。

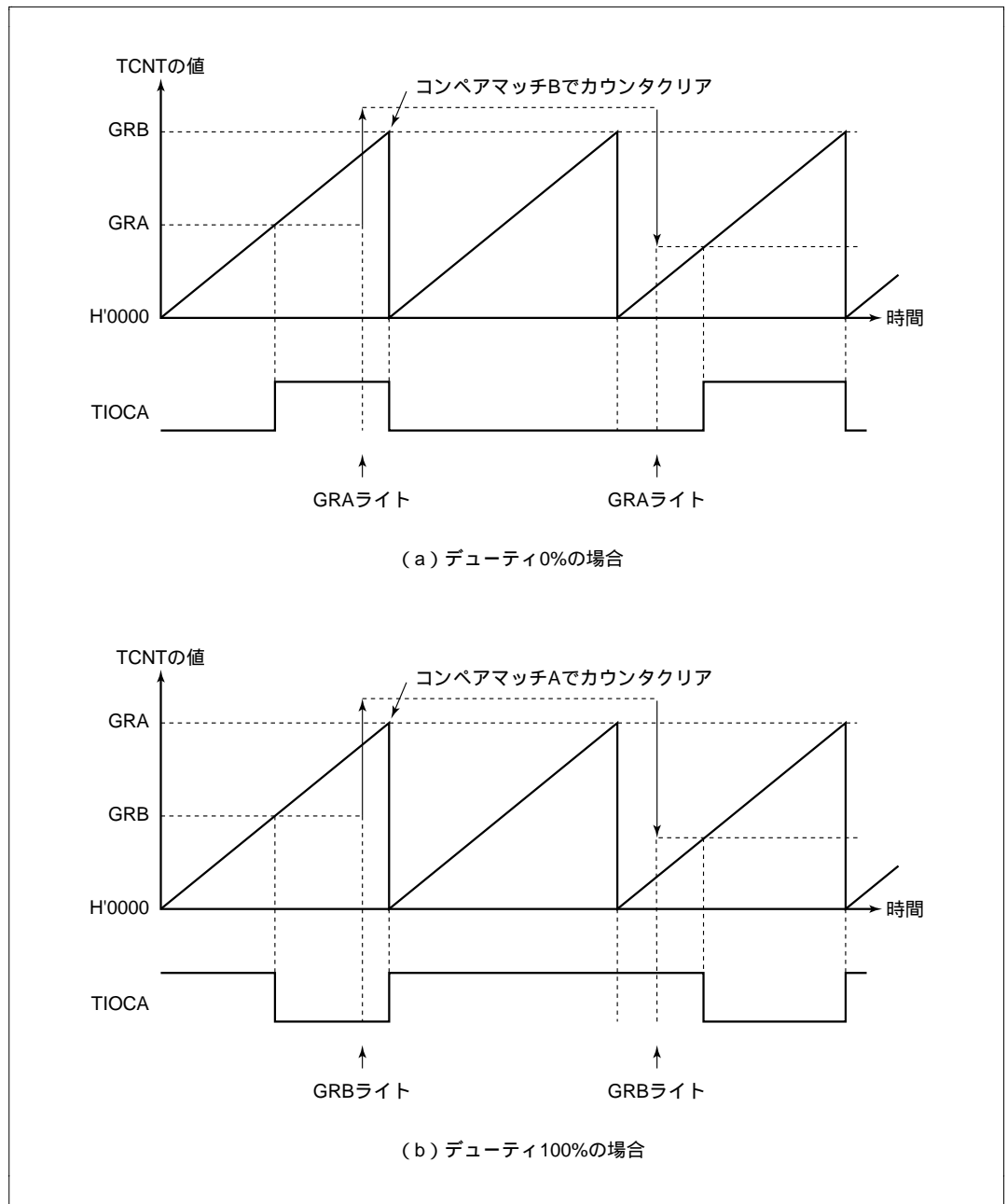


図 8.30 PWM モードの動作例 (2)

8.4.5 リセット同期 PWM モード

リセット同期 PWM モードは、チャンネル 3、4 を組み合わせることにより、一方の波形変化点が共通の関係となる PWM 波形（正相と逆相）を 3 相出力します。

リセット同期 PWM モードに設定すると、TIOCA₃、TIOCB₃、TIOCA₄、TOCXA₄、および TIOCB₄、TOCXB₄ 端子は自動的に PWM 出力端子となり、TCNT3 はアップカウンタとして機能します。使用される PWM 出力端子を表 8.5 に、使用するレジスタの設定を表 8.6 に示します。

表 8.5 リセット同期 PWM モード時の出力端子

チャンネル	出力端子	説明
3	TIOCA ₃	PWM 出力 1
	TIOCB ₃	PWM 出力 1' (PWM 出力 1 の逆相波形)
4	TIOCA ₄	PWM 出力 2
	TOCXA ₄	PWM 出力 2' (PWM 出力 2 の逆相波形)
	TIOCB ₄	PWM 出力 3
	TOCXB ₄	PWM 出力 3' (PWM 出力 3 の逆相波形)

表 8.6 リセット同期 PWM モード時のレジスタ設定

レジスタ	設定内容
TCNT3	H'0000 を初期設定
TCNT4	使用しません (独立に動作)
GRA3	TCNT3 のカウント周期を設定
GRB3	TIOCA ₃ 、TIOCB ₃ 端子より出力される PWM 波形の変化点を設定
GRA4	TIOCA ₄ 、TOCXA ₄ 端子より出力される PWM 波形の変化点を設定
GRB4	TIOCB ₄ 、TOCXB ₄ 端子より出力される PWM 波形の変化点を設定

(1) リセット同期 PWM モードの設定手順例

リセット同期 PWM モードの設定手順を図 8.31 に示します。

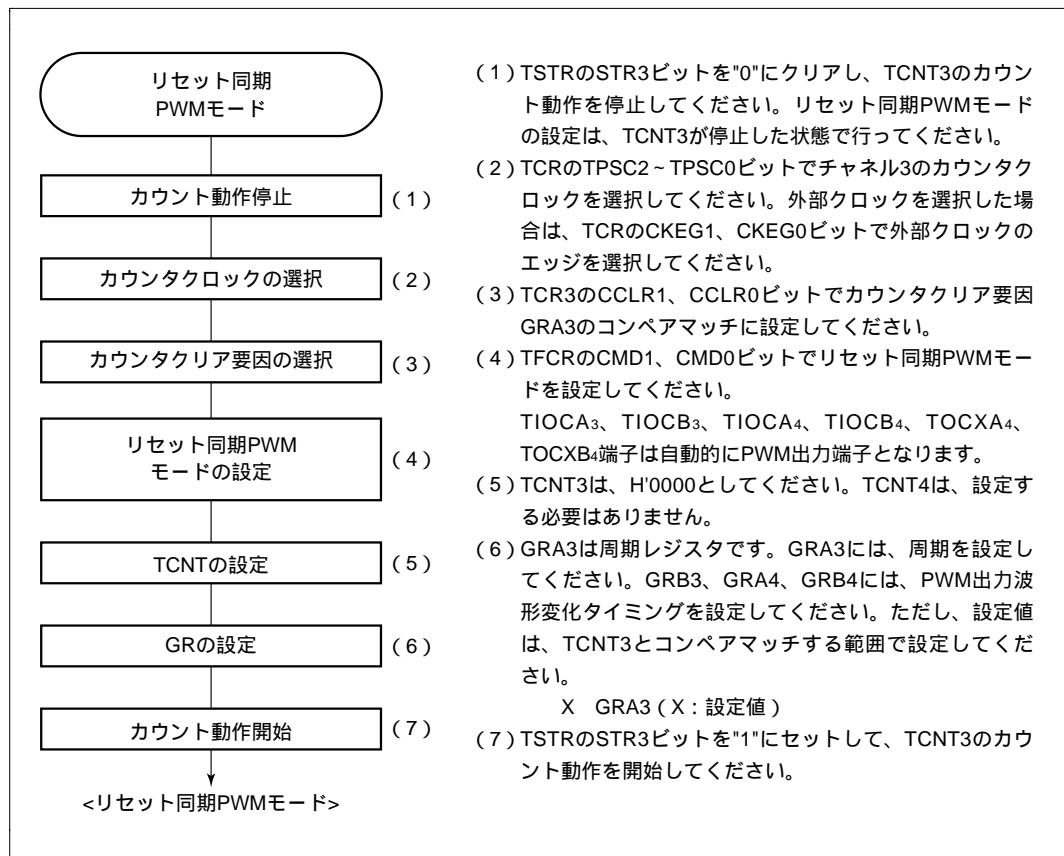


図 8.31 リセット同期 PWM モードの設定手順例

(2) リセット同期 PWM モードの動作例

リセット同期 PWM モードの動作例を図 8.32 に示します。

リセット同期 PWM モードでは、TCNT3 はアップカウンタとして動作します。TCNT4 は独立動作します。ただし、GRA4、GRB4 は TCNT4 から切り離されます。TCNT3 が GRA3 とコンペアマッチするとカウンタはクリアされ、H'0000 からカウントアップを再開します。

PWM 出力端子は、それぞれ GRB3、GRA4、GRB4 と TCNT3 のコンペアマッチおよびカウンタクリアが発生するたびにトグル出力を行います。

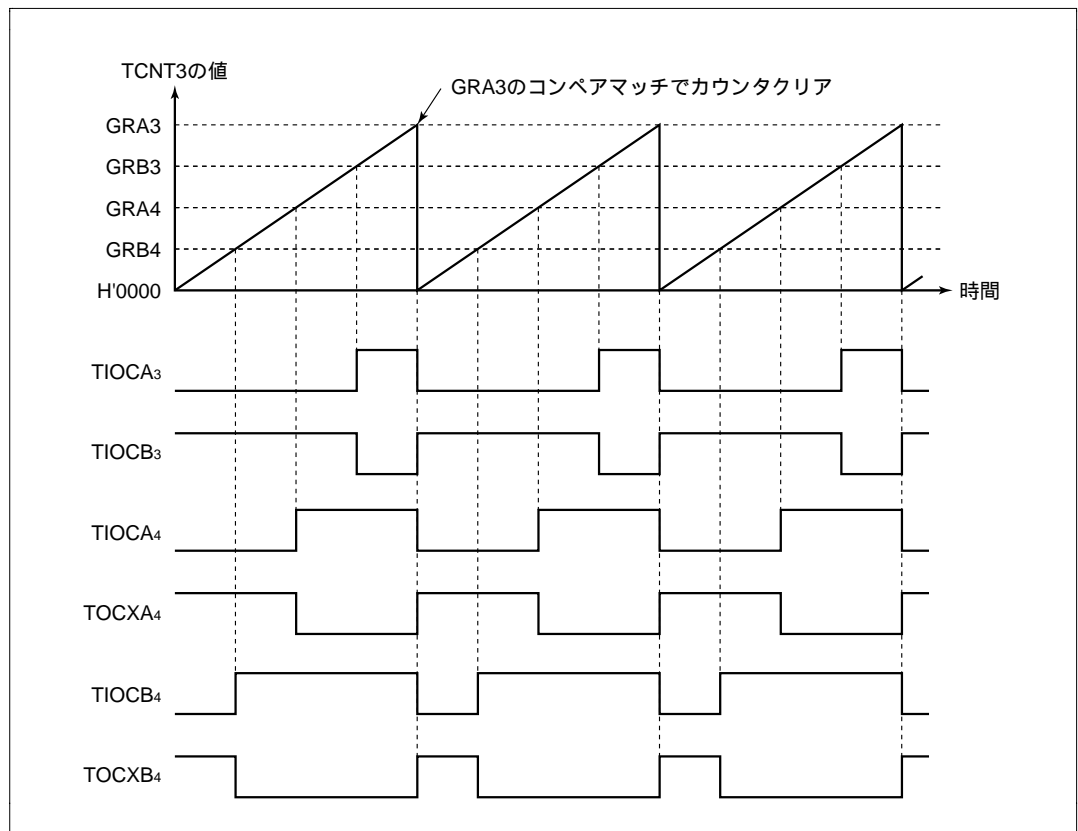


図 8.32 リセット同期 PWM モードの動作例 (OLS3 = OLS4 = 1 の場合)

リセット同期 PWM モードとバッファ動作を同時に設定した場合の動作については、「8.4.8 バッファ動作」を参照してください。

8.4.6 相補 PWM モード

相補 PWM モードは、チャンネル 3、4 を組み合わせることにより、正相と逆相がノンオーバーラップの関係にある PWM 波形を 3 相出力します。

相補 PWM モードに設定すると、TIOCA₃、TIOCB₃、TIOCA₄、TOCXA₄、および TIOCB₄、TOCXB₄ 端子は自動的に PWM 出力端子となり、TCNT3 と TCNT4 はアップ/ダウンカウンタとして機能します。

使用される PWM 出力端子を表 8.7 に、使用するレジスタの設定を表 8.8 に示します。

表 8.7 相補 PWM モード時の出力端子

チャンネル	出力端子	説明
3	TIOCA ₃	PWM 出力 1
	TIOCB ₃	PWM 出力 1' (PWM 出力 1 とノンオーバーラップの関係にある逆相波形)
4	TIOCA ₄	PWM 出力 2
	TOCXA ₄	PWM 出力 2' (PWM 出力 2 とノンオーバーラップの関係にある逆相波形)
	TIOCB ₄	PWM 出力 3
	TOCXB ₄	PWM 出力 3' (PWM 出力 3 とノンオーバーラップの関係にある逆相波形)

表 8.8 相補 PWM モード時のレジスタ設定

レジスタ	設定内容
TCNT3	ノンオーバーラップ期間を初期設定 (TCNT4 との差がノンオーバーラップ期間となります)
TCNT4	H'0000 を初期設定
GRA3	TCNT3 の上限値 - 1 を設定
GRB3	TIOCA ₃ 、TIOCB ₃ 端子より出力される PWM 波形の変化点を設定
GRA4	TIOCA ₄ 、TOCXA ₄ 端子より出力される PWM 波形の変化点を設定
GRB4	TIOCB ₄ 、TOCXB ₄ 端子より出力される PWM 波形の変化点を設定

(1) 相補 PWM モードの設定手順

相補 PWM モードの設定手順例を図 8.33 に示します。

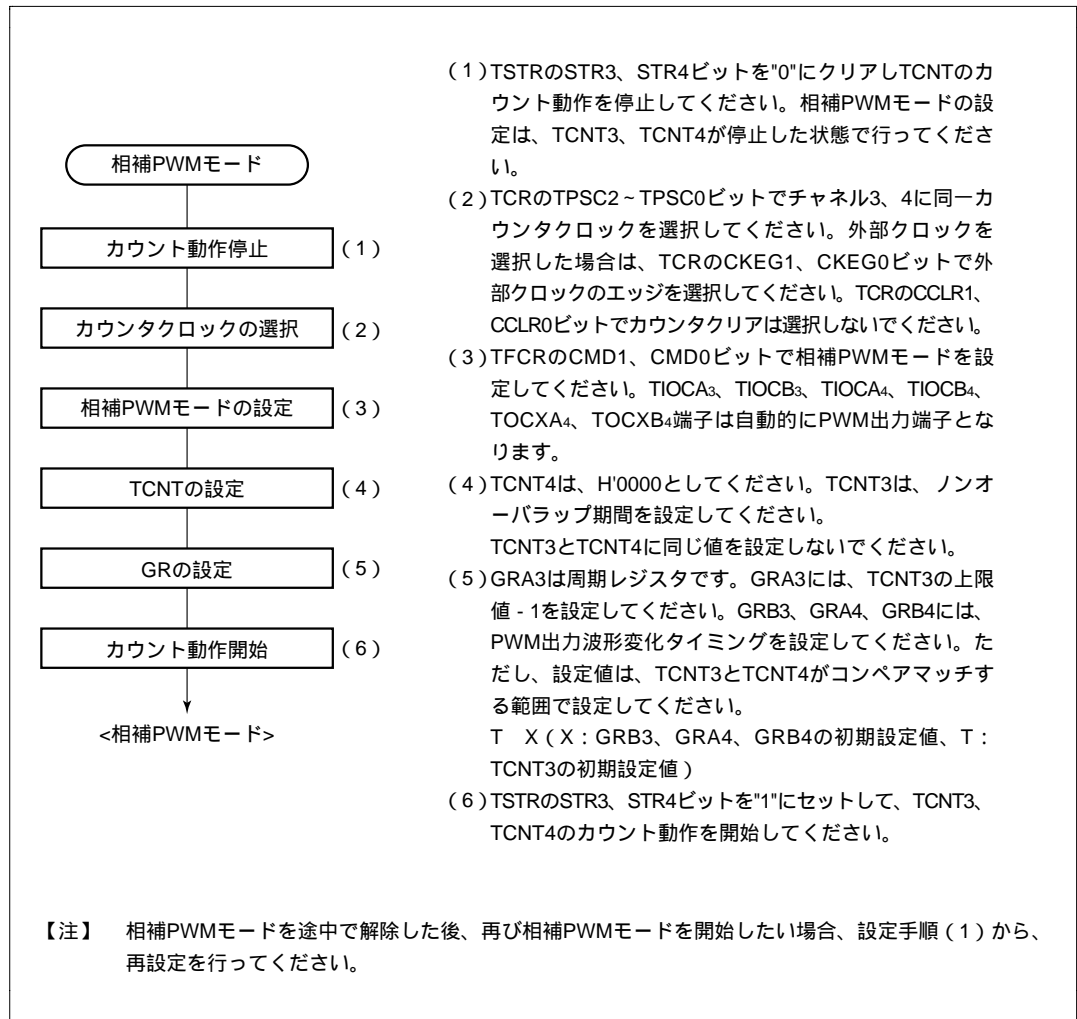


図 8.33 相補 PWM モードの設定手順例

(2) 相補 PWM モードの解除手順

相補 PWM モードの解除手順を図 8.34 に示します。

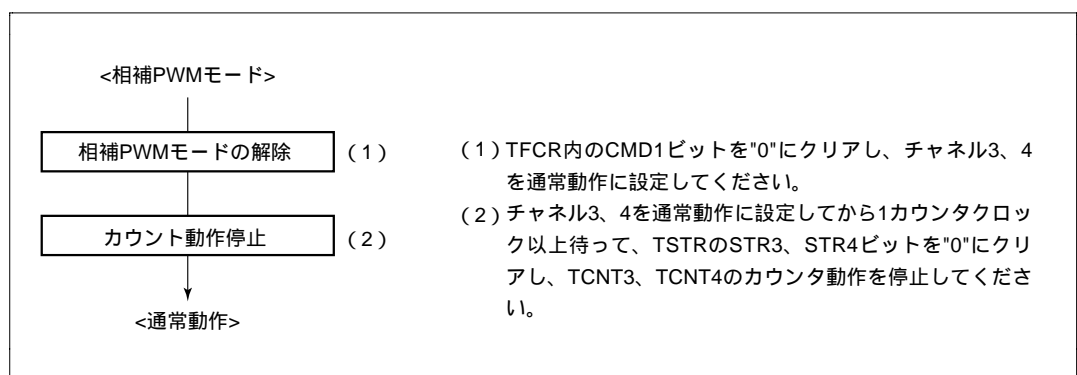


図 8.34 相補 PWM モードの解除手順

(3) 相補 PWM モードの動作例

相補 PWM モードの動作例を図 8.35 に示します。

相補 PWM モードでは、TCNT3、TCNT4 はアップ / ダウンカウンタとして動作します。TCNT3 が GRA3 とコンペアマッチするとダウンカウントし、TCNT4 がアンダフローするとアップカウントします。

GRB3、GRA4、GRB4 はカウンタのアップ / ダウン 1 周期中、それぞれ TCNT3 TCNT4 TCNT4 TCNT3 の順にコンペアマッチを行い PWM 波形を生成します (本モードでは、TCNT3 > TCNT4 に初期設定します)。

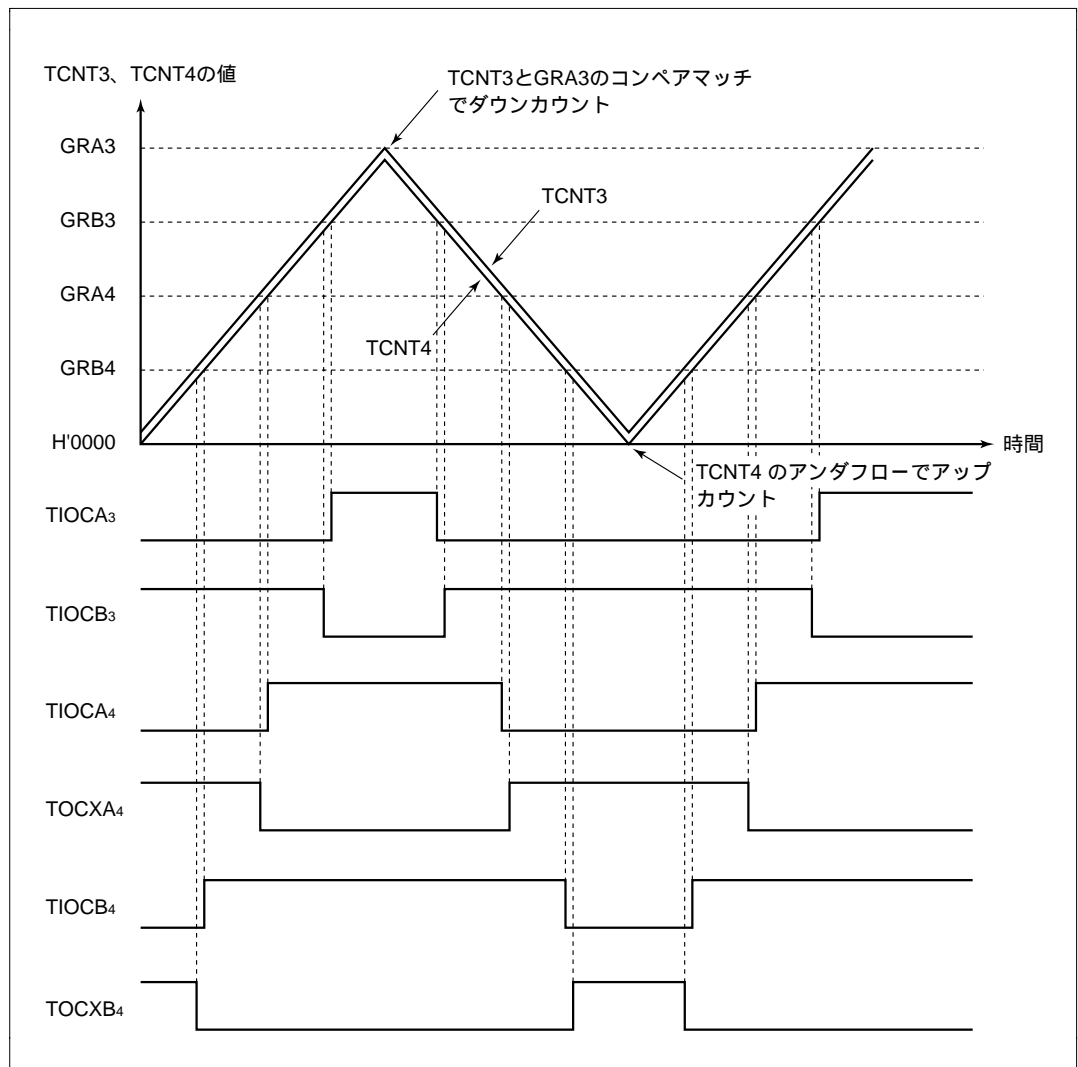


図 8.35 相補 PWM モードの動作例 (1) (OLS3 = OLS4 = 1 の場合)

相補 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 (1 相分) を図 8.36 に示します。

本例では GRB3 のコンペアマッチで端子出力が変化しますので、GRB3 の値を GRA3 の値よりも大きい値とすることでデューティ 0%、デューティ 100% の波形出力が可能となります。バッファ動作を併用すると上記操作を含め、動作中のデューティ変更を容易に行う

ことができます。

バッファ動作については「8.4.8 バッファ動作」を参照してください。

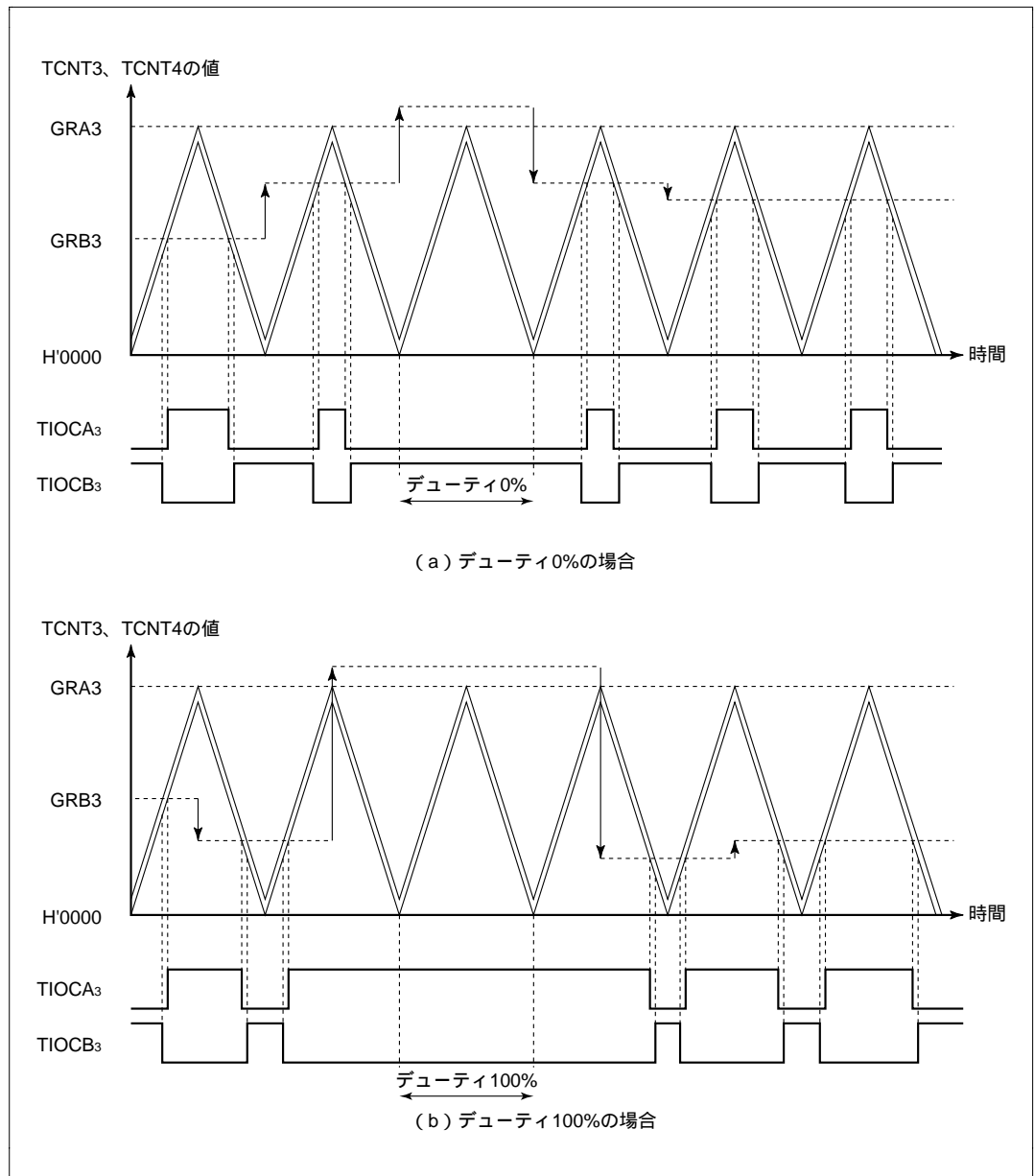


図 8.36 相補 PWM モードの動作例 (2) (OLS3 = OLS4 = 1 の場合)

相補 PWM モードを使用しているときのアップカウント / ダウンカウントの変化点で、TCNT3、TCNT4 はそれぞれオーバシュート / アンダシュートが発生します。

このとき、チャンネル 3 の IMFA フラグおよびチャンネル 4 の OVF フラグをセットする条件は通常の場合とは異なります。また、バッファ動作での転送条件も異なります。

このタイミングを図 8.37、図 8.38 に示します。

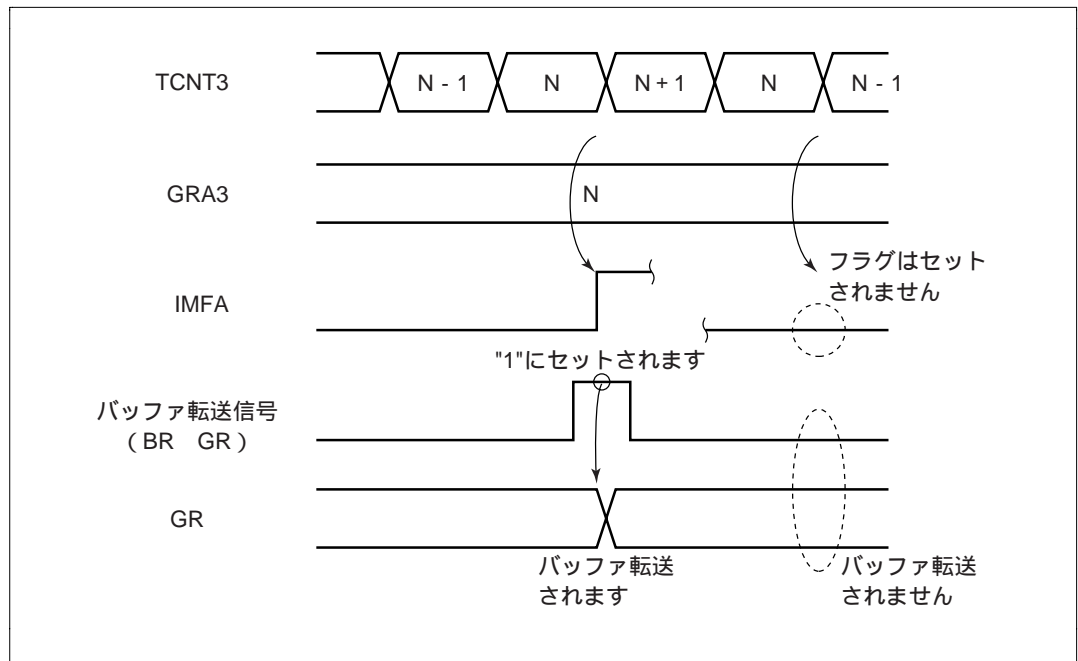


図 8.37 オーバシュート時のタイミング

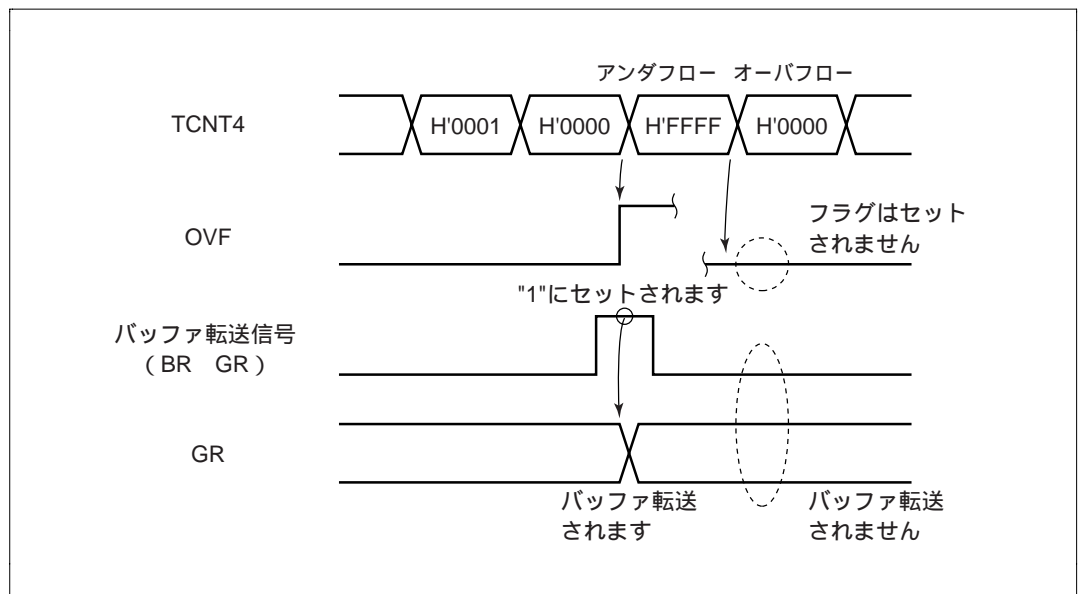


図 8.38 アンダシュート時のタイミング

チャンネル3のIMFAフラグはアップカウント時に、チャンネル4のOVFフラグはアンダフロー時のみ、それぞれ"1"にセットされます。

バッファ動作を設定されたBRは、アップカウント動作時のコンペアマッチA3またはTCNT4のアンダフローによってGRに転送されます。

(4) 相補 PWM モードでの GR の設定値

相補 PWM モードでの GR の設定および動作中の変更については、以下の点に注意してください。

(1) 初期値

H'0000 ~ T - 1 (T : TCNT3 の初期設定値) の設定は禁止です。

なお、カウントスタート後、最初に発生するコンペアマッチ A3 のタイミング以降では、この設定も可能です。

(2) 設定値の変更方法

バッファ動作を使用してください。直接 GR にライトすると、正しく波形出力されない場合があります。

(3) 設定値変更時の注意

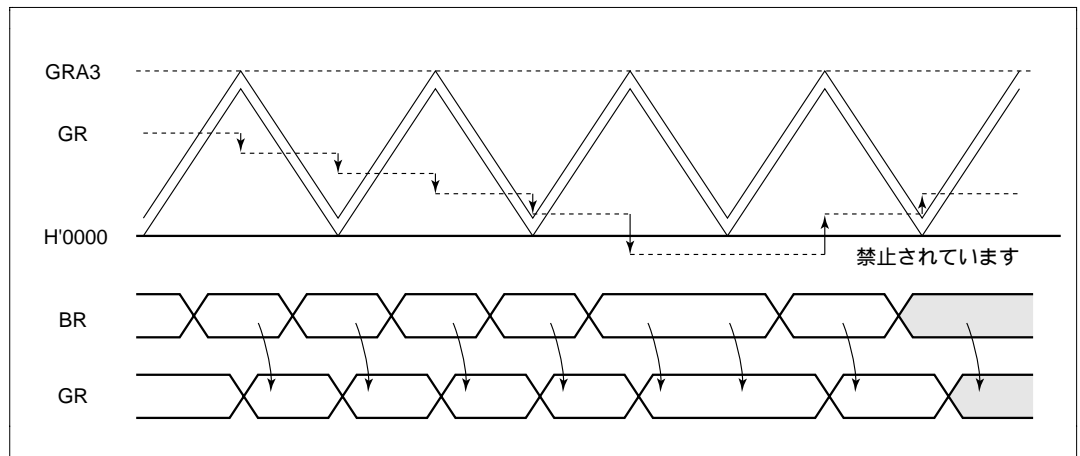


図 8.39 バッファ動作による GR の設定値変更例 (1)

(a) アップカウントからダウンカウントへの変化時のバッファ転送

GR の内容が $GRA3 - T + 1 \sim GRA3$ の範囲内であるとき、この範囲外の値は転送しないでください。また、GR の内容がこの範囲外であるとき、この範囲内の値は転送しないでください。

バッファ動作による GR の設定変更時の注意 (1) を図 8.40 に示します。

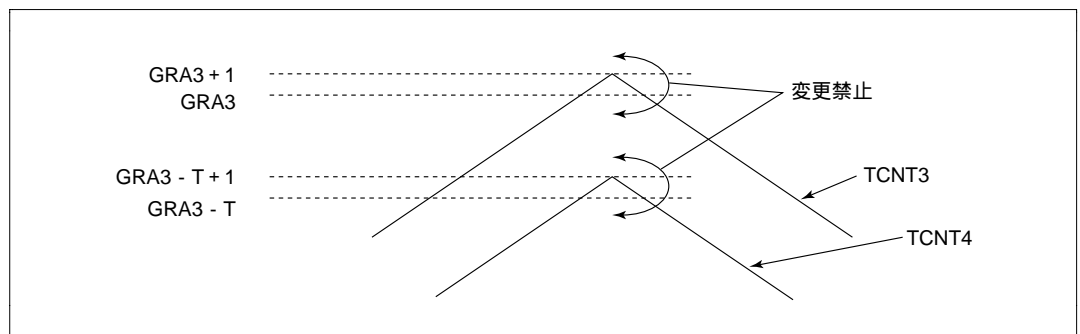


図 8.40 バッファ動作による GR の設定変更時の注意 (1)

(b) ダウンカウントからアップカウントへの変化時のバッファ転送

GR の内容が $H'0000 \sim T - 1$ の範囲であるとき、この範囲外の値は転送しないでください。また、GR の内容がこの範囲外であるとき、この範囲内の値は転送しないでください。

バッファ動作による GR の設定変更時の注意 (2) を図 8.41 に示します。

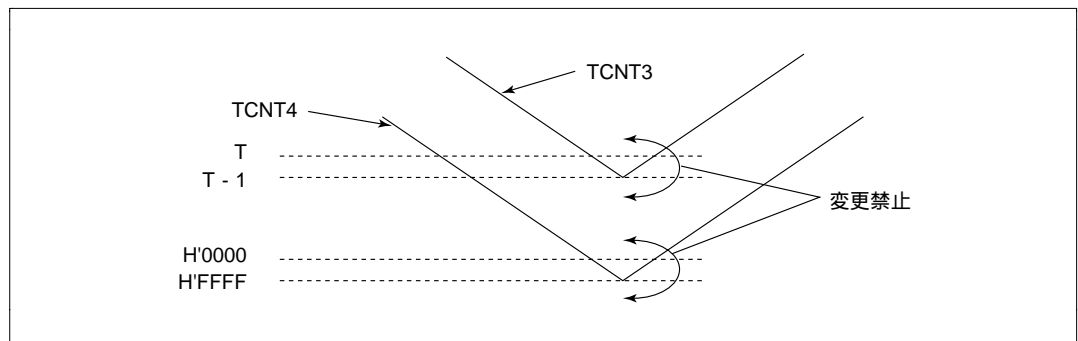


図 8.41 バッファ動作による GR の設定変更時の注意 (2)

(c) GR の設定をカウント領域 ($H'0000 \sim \text{GRA3}$) 外とするとき

デューティ 0%、100%の波形を出力する場合、GR の設定をカウント領域外とすることにより実現可能です。このとき、カウント領域外の設定値を BR にライトしたときのカウント方向 (アップ/ダウンカウント) と、カウント領域内にもどる設定値を BR にライトするときのカウント方向が同一となるようにしてください。

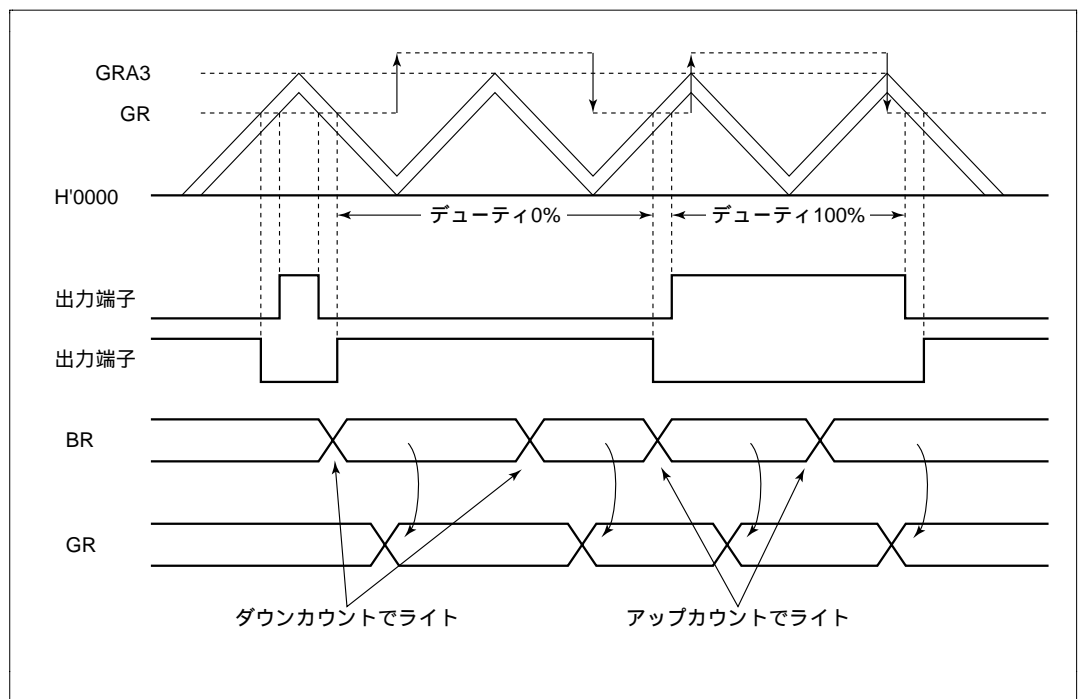


図 8.42 バッファ動作による GR の設定値変更例 (2)

上記設定は、GRA3のコンペアマッチまたはTCNT4のアンダフローが発生したことを検出して、BRヘライトをすることによって実現可能です。

8.4.7 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、TCNT2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、TCR2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TCNT2はアップ/ダウンカウンタとなります。ただし、TCR2のCCLR1、CCLR0ビット、TIOR2、TIER2、TSR2、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8.43に示します。

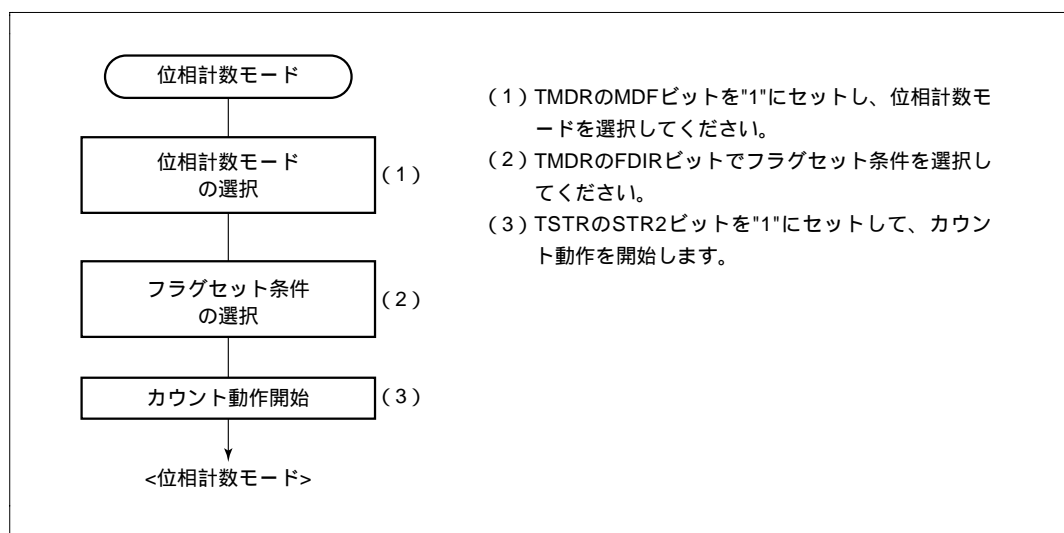


図 8.43 位相計数モードの設定手順例

(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8.44に、TCNT2のアップ/ダウンカウント条件を表8.9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。

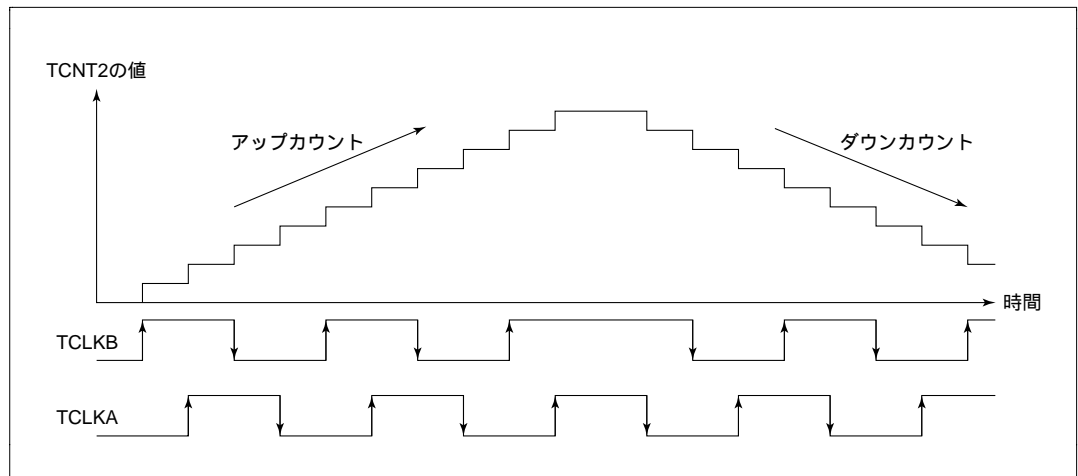


図 8.44 位相計数モードの動作例

表 8.9 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"

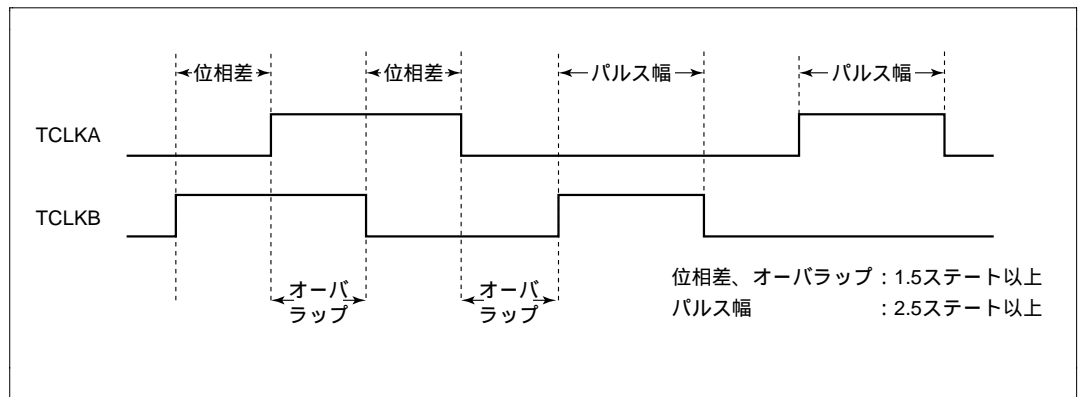


図 8.45 位相計数モード時の位相差、オーバーラップおよびパルス幅

8.4.8 バッファ動作

バッファ動作は、GR をアウトプットコンペアレジスタに設定した場合、GR をインプットキャプチャレジスタに設定した場合、リセット同期 PWM モード時、および相補 PWM モード時で機能が異なります。

バッファ動作はチャンネル 3、4 のみがもつ機能です。

上記の条件でバッファ動作に設定すると以下のように動作します。

(1) GR がアウトプットコンペアレジスタの場合

コンペアマッチが発生すると当該チャンネルの BR の値が、GR に転送されます。

この動作を図 8.46 に示します。

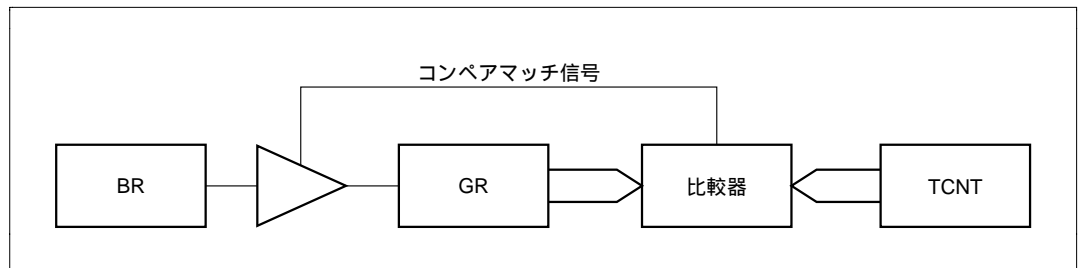


図 8.46 コンペアマッチバッファ動作

(2) GR が入力キャプチャレジスタの場合

入力キャプチャが発生すると TCNT の値を GR に転送すると同時に、それまで格納されていた GR の値を BR に転送します。

この動作を図 8.47 に示します。

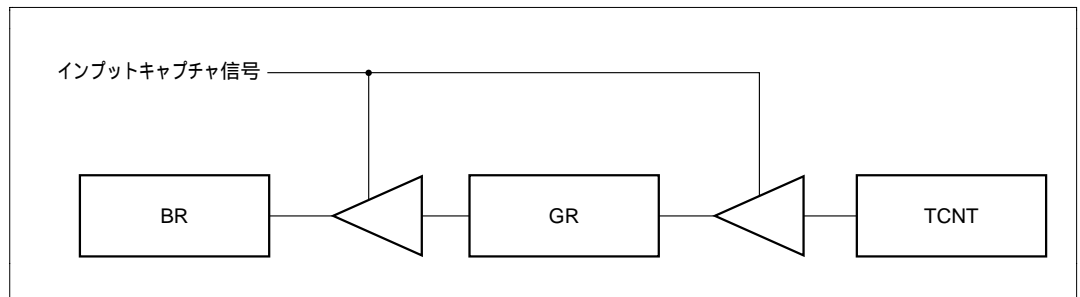


図 8.47 入力キャプチャバッファ動作

(3) 相補 PWM モードの場合

TCNT3、TCNT4 のカウント方向が変化すると BR の値が GR に転送されます。このとき、BR から GR への転送は以下のタイミングで行われます。

- ・ TCNT3 と GRA3 がコンペアマッチしたとき
- ・ TCNT4 がアンダフローしたとき

(4) リセット同期 PWM モードの場合

コンペアマッチ A3 により BR の値が、GR に転送されます。

(1) バッファ動作の設定手順例

バッファ動作の設定手順例を図 8.48 に示します。

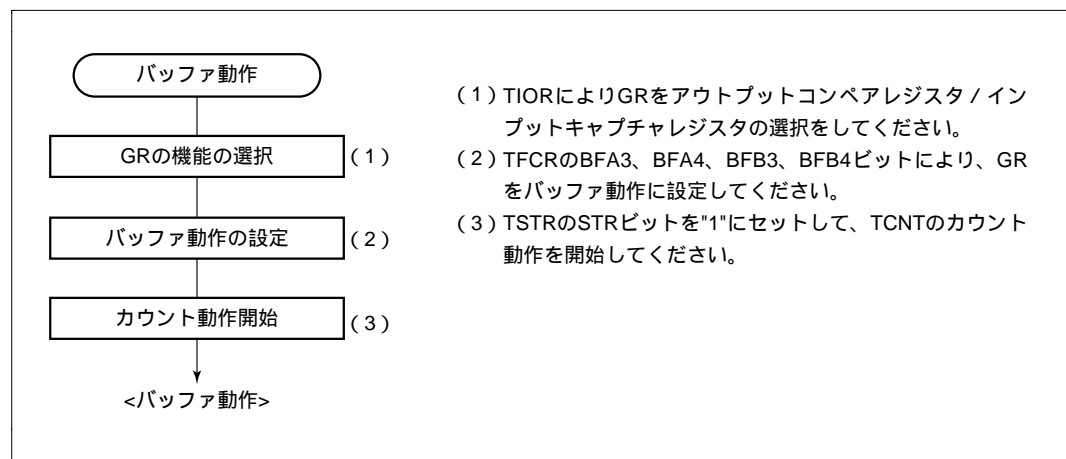


図 8.48 バッファ動作の設定手順例

(2) バッファ動作例

GRA をアウトプットコンペアレジスタに設定し、GRA と BRA をバッファ動作に設定したときの動作を図 8.49 に示します。

TCNT がコンペアマッチ B によりクリアされる周期カウント動作をしている場合の例です。また、TIOCA、TIOCB 端子は、それぞれコンペアマッチ A、B によるトグル出力が設定されています。

バッファ動作が設定されているため、コンペアマッチ A で TIOCA 端子がトグル出力を行うと同時に、BRA の値が GRA に転送されます。この動作をコンペアマッチ A が発生するたびに繰り返します。

この転送タイミングを図 8.50 に示します。

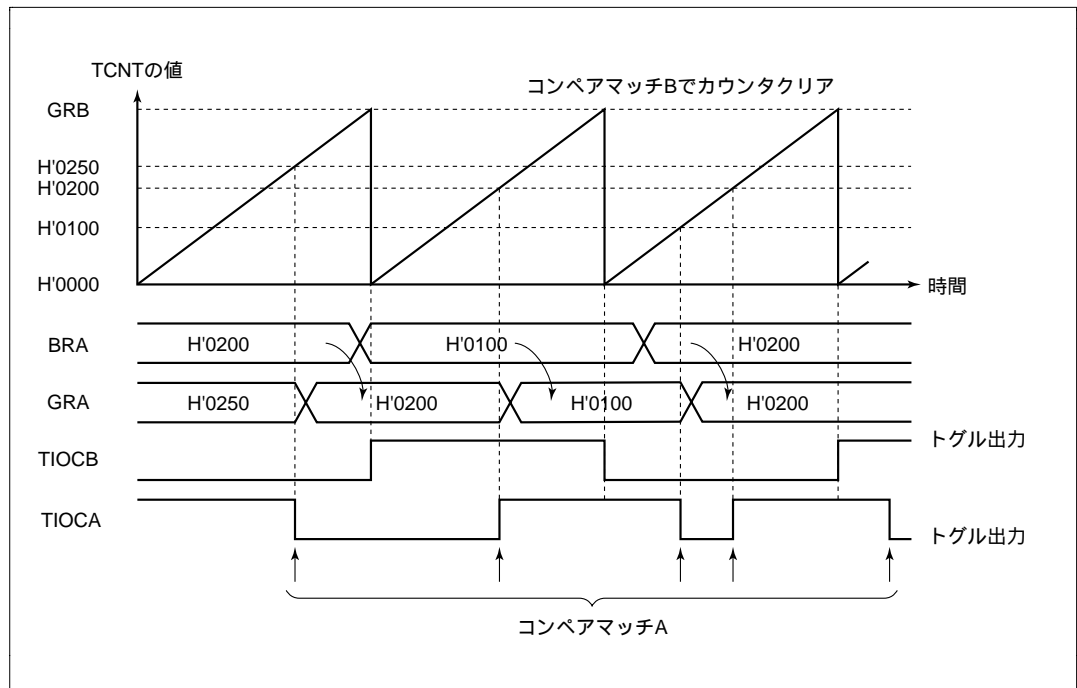


図 8.49 バッファ動作例 (1) (アウトプットコンペアレジスタに対するバッファ動作)

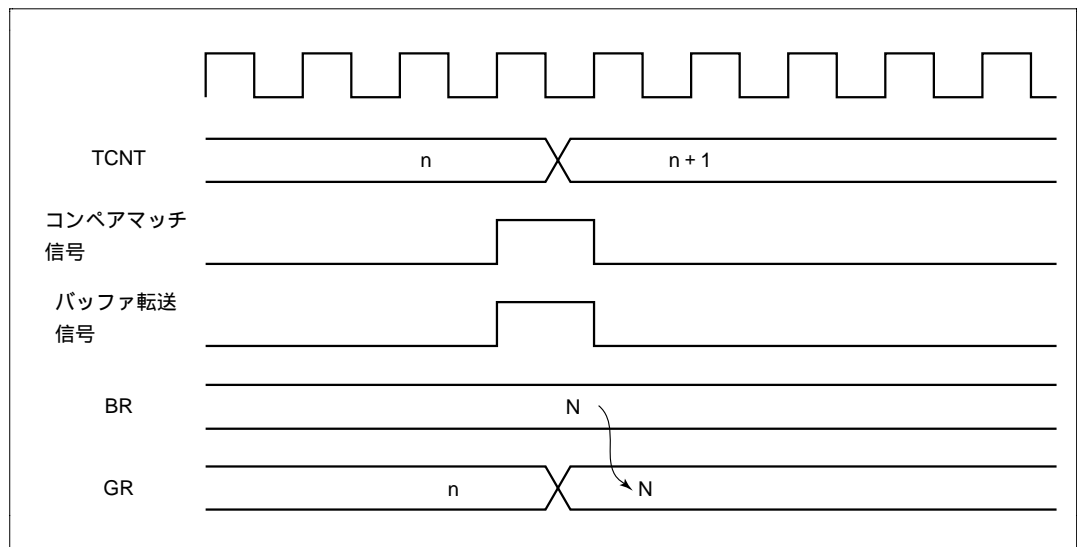


図 8.50 バッファ動作時のコンペアマッチタイミング例

GRA をインプットキャプチャレジスタに設定し、GRA と BRA をバッファ動作に設定したときの動作を図 8.51 に示します。

TCNT がインプットキャプチャ B によりカウンタクリアされる場合の例です。TIOCB 端子のインプットキャプチャ入力エッジは、立下がりエッジが選択され、また、TIOCA 端子のインプ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が選択されています。バッファ動作が設定されているため、インプットキャプチャ A により TCNT の値が GRA に格納されると同時に、それまで格納されていた GRA の値が BRA に転送されます。この転送タイミングを図 8.52 に示します。

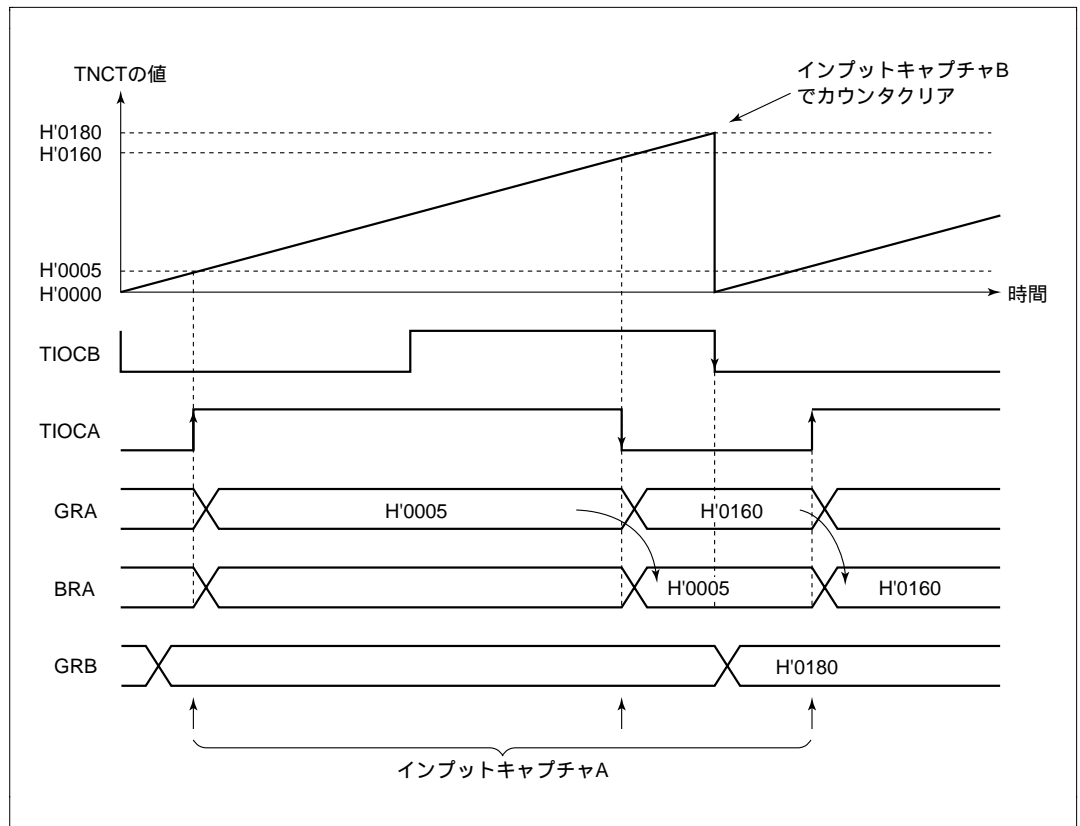


図 8.51 バッファ動作例 (2) (入力キャプチャレジスタに対するバッファ動作)

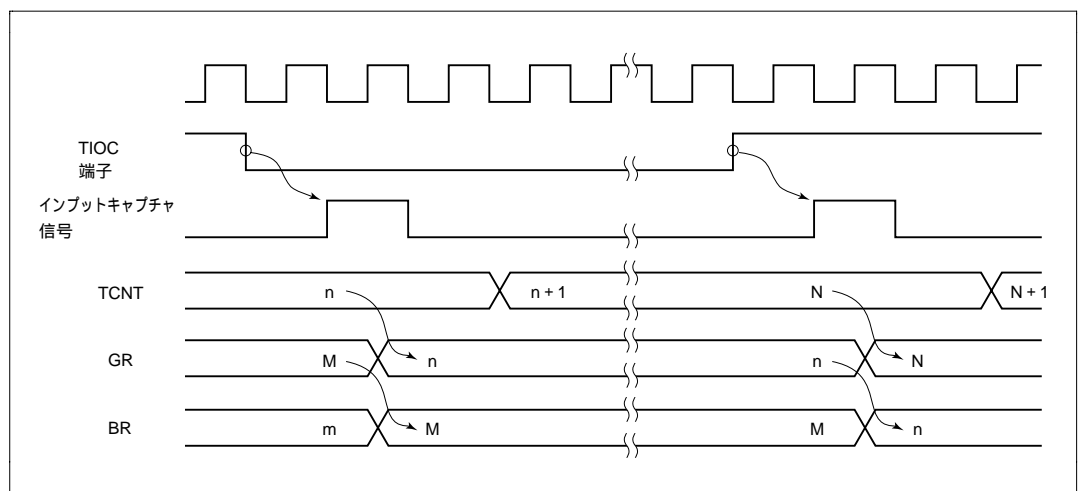


図 8.52 バッファ動作時の入力キャプチャタイミング

相補PWMモード時、GRB3とBRB3をバッファ動作に設定したときの動作例を図8.53に示します。

バッファ動作を使用してGRB3 > GRA3とすることにより、デューティ0%のPWM波形を生成した場合の例です。

BRBからGRBへの転送は、TCNT3とGRA3がコンペアマッチしたとき、およびTCNT4がアンダフローしたときに行われます。

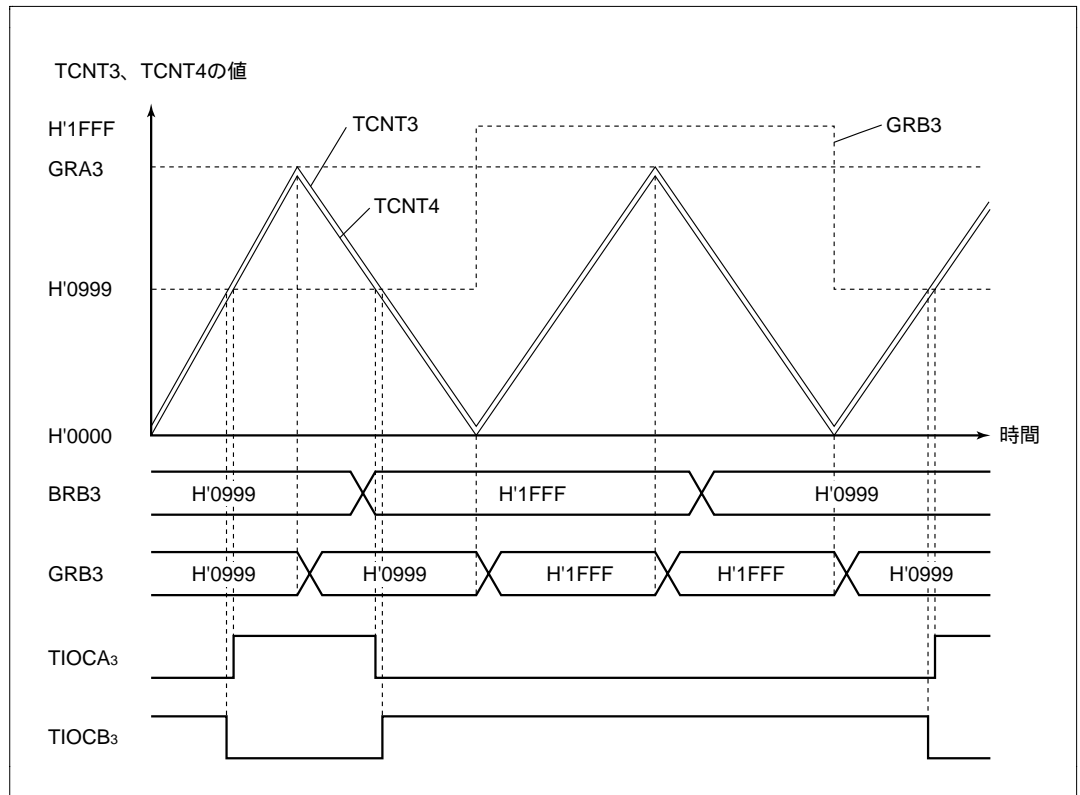


図8.53 バッファ動作例(4) (相補PWMモード時のバッファ動作)

8.4.9 ITU 出力タイミング

チャンネル3、4のITU出力は、TOER、TOCRの設定および外部トリガにより、出力を禁止したり反転したりすることができます。

(1) TOER による ITU 出力の許可 / 禁止タイミング

TOER のマスタインネーブルビットを "0" にクリアして、ITU 出力を禁止する場合の例です。対応する入出力ポートの DR、DDR をあらかじめ設定しておくことにより、任意の値を出力することができます。

TOER による ITU 出力を許可 / 禁止するタイミングを図 8.54 に示します。

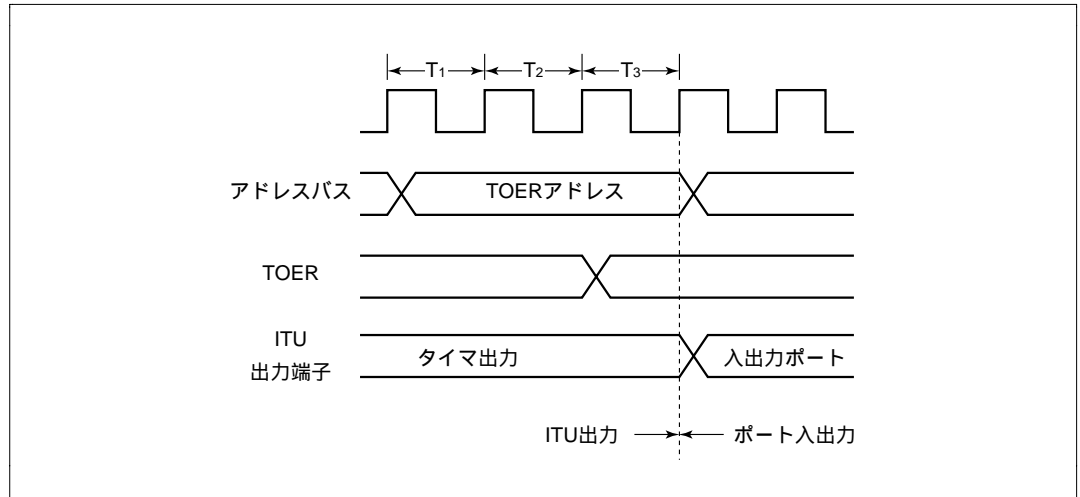


図 8.54 TOER へのライトによる ITU 出力禁止タイミングの例

(2) 外部トリガによる ITU 出力禁止タイミング

リセット同期 PWM モードまたは相補 PWM モード時に、TOCR の XTGD ビットが "0" にクリアされている状態でチャンネル 1 のインプットキャプチャ A 信号が発生すると、TOER のマスタインネーブルビットが "0" にクリアされ ITU 出力が禁止されます。

このタイミングを図 8.55 に示します。

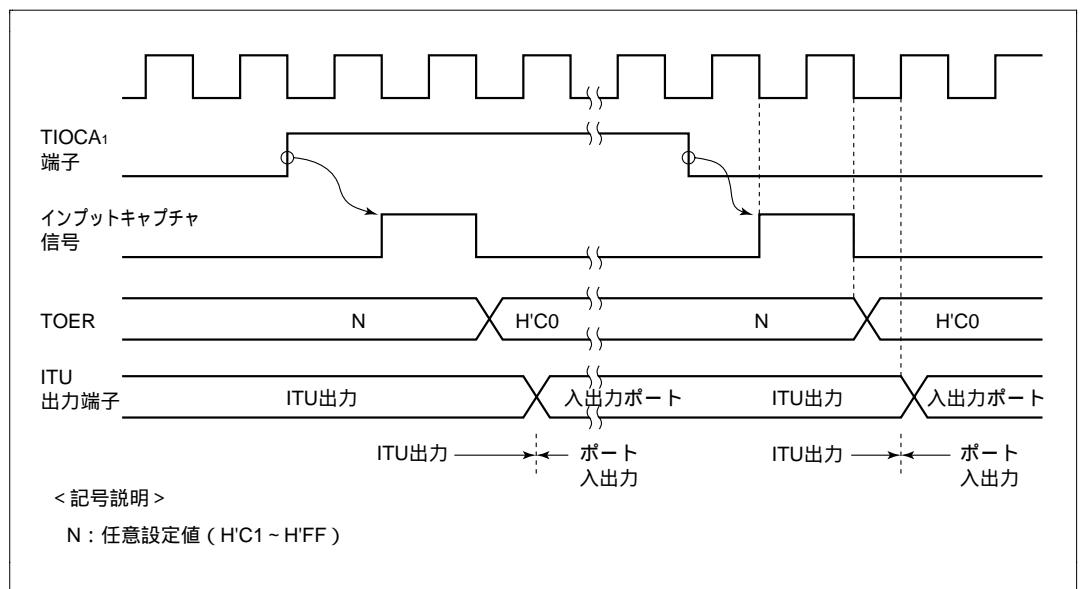


図 8.55 外部トリガによる ITU 出力禁止タイミングの例

(3) TOCR による出力反転タイミング

リセット同期PWMモードまたは相補PWMモード時に、TOCRのアウトプットレベルセレクト(OLS4、OLS3)ビットを反転することにより、出力レベルを反転することができます。

このタイミングを図8.56に示します。

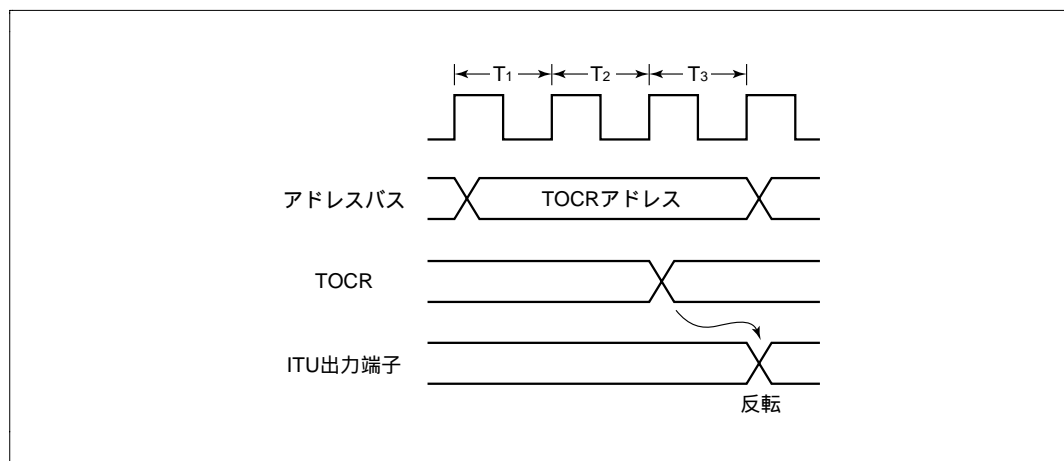


図 8.56 TOCR へのライトによる ITU 出力レベル反転タイミングの例

8.5 割込み

ITU 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、オーバフロー割込みの 2 種類があります。

8.5.1 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 TCNT 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 (TCNT 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、TCNT と GR が一致した後、TCNT 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 8.57 に IMF フラグのセットタイミングを示します。

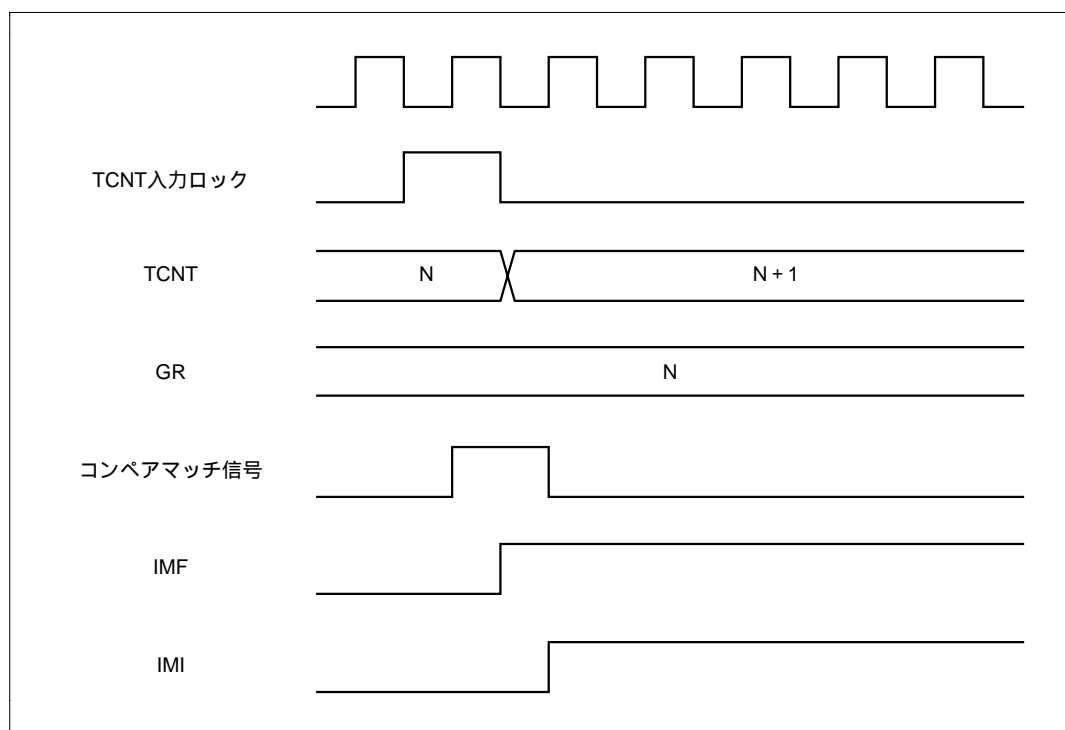


図 8.57 コンペアマッチ時の IMFA、IMFB フラグのセットタイミング

(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 TCNT の値が対応する GR に転送されます。

このタイミングを図 8.58 に示します。

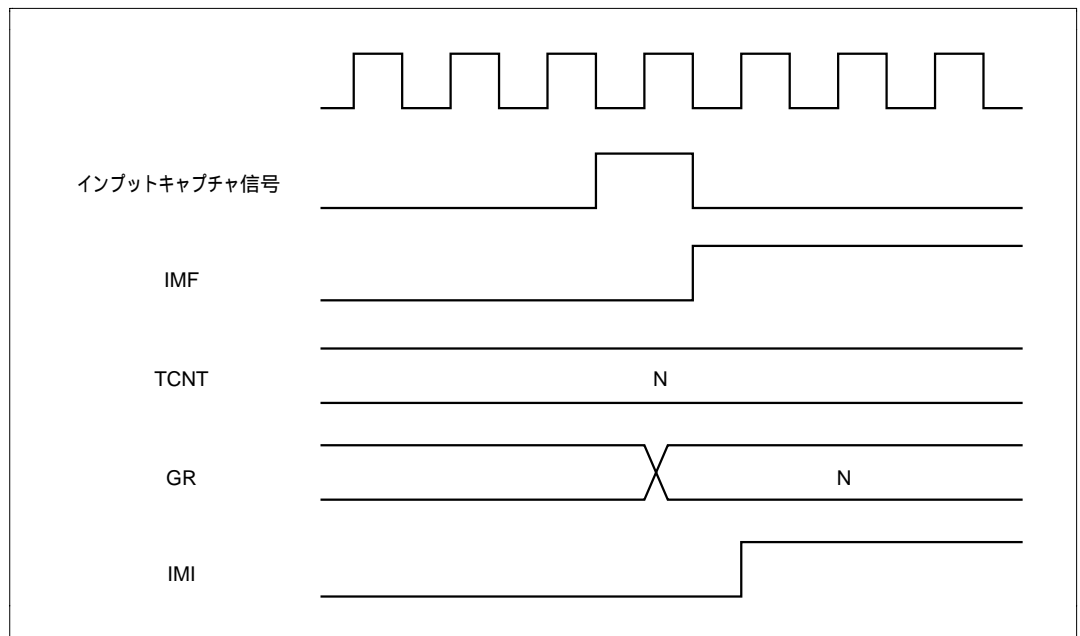


図 8.58 インพุットキャプチャ時の IMFA、IMFB フラグのセットタイミング

(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、TCNT がオーバーフロー (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。

このときのタイミングを図 8.59 に示します。

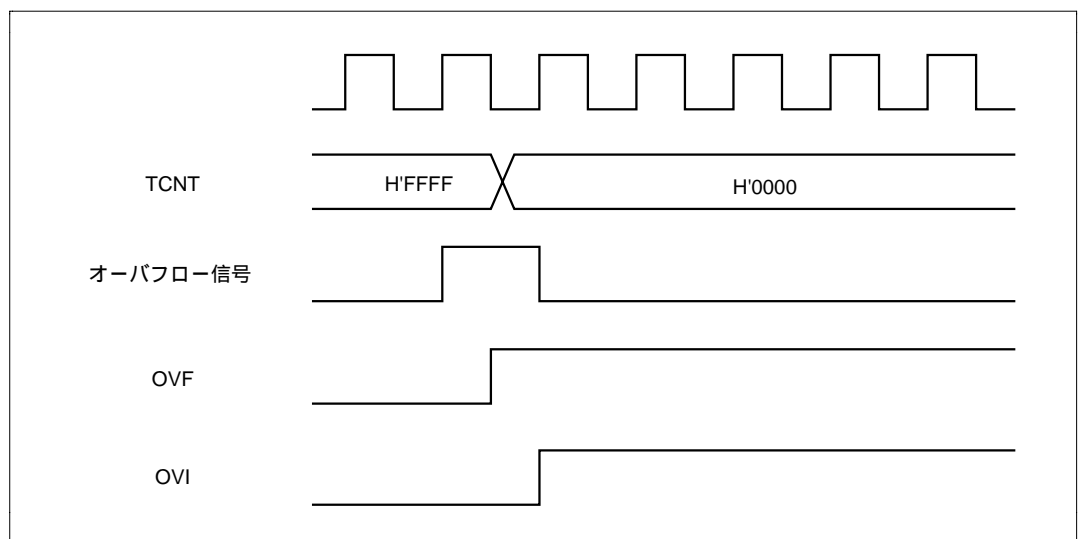


図 8.59 OVF フラグのセットタイミング

8.5.2 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 8.60 に示します。

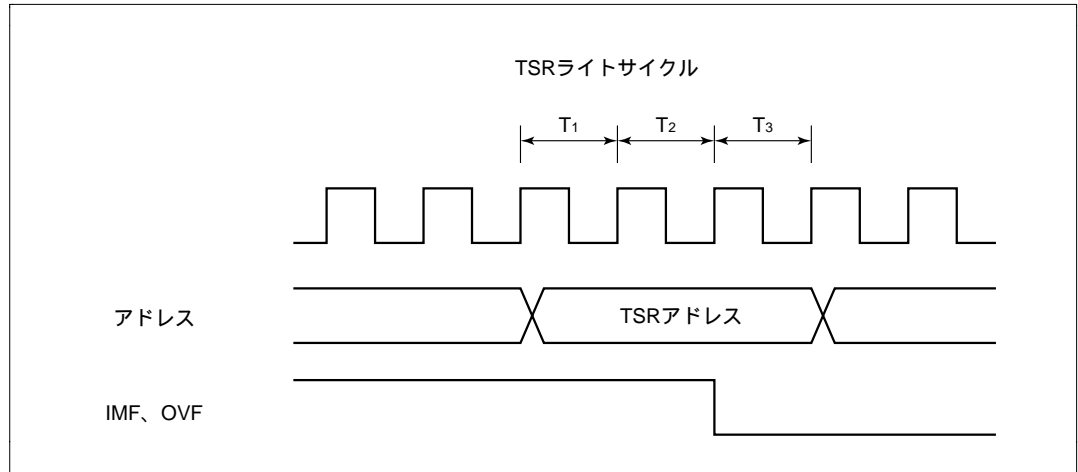


図 8.60 ステータスフラグのクリアタイミング

8.5.3 割込み要因

ITU 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 2 種類の割込み計 15 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、かつ割込み許可ビットが"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、IPRB 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。

ITU の割込み要因を表 8.10 に示します。

表 8.10 ITU 割込み要因

チャンネル	割込み要因	内 容	優先順位*
0	IMIA0	コンペアマッチ / インพุットキャプチャA0	高 ↑ 低
	IMIB0	コンペアマッチ / インพุットキャプチャB0	
	OVI0	オーバフロー0	
1	IMIA1	コンペアマッチ / インพุットキャプチャA1	
	IMIB1	コンペアマッチ / インพุットキャプチャB1	
	OVI1	オーバフロー1	
2	IMIA2	コンペアマッチ / インพุットキャプチャA2	
	IMIB2	コンペアマッチ / インพุットキャプチャB2	
	OVI2	オーバフロー2	
3	IMIA3	コンペアマッチ / インพุットキャプチャA3	
	IMIB3	コンペアマッチ / インพุットキャプチャB3	
	OVI3	オーバフロー3	
4	IMIA4	コンペアマッチ / インพุットキャプチャA4	
	IMIB4	コンペアマッチ / インพุットキャプチャB4	
	OVI4	オーバフロー4	

【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、IPRBにより変更可能です。

8.6 使用上の注意

ITU の動作中、次のような競合や動作が起こりますので、注意してください。

(1) TCNT のライトとクリアの競合

TCNT のライトサイクル中の T3 ステートで、カウントクリア信号が発生すると、TCNT への書込みサイクルは行われず TCNT のクリアが優先されます。

このタイミングを図 8.61 に示します。

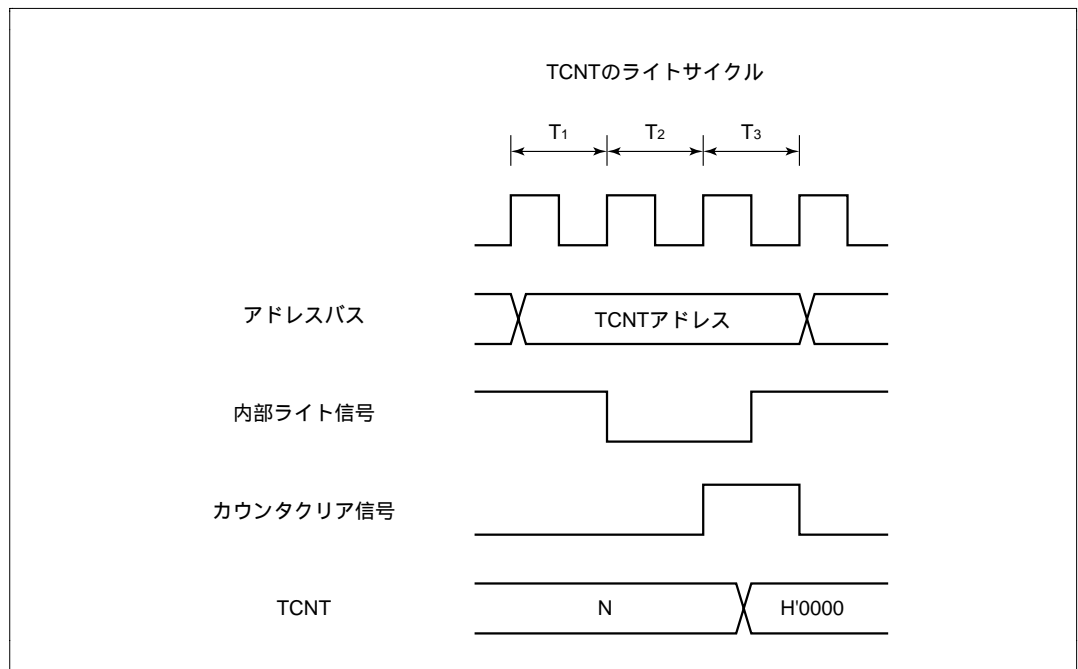


図 8.61 TCNT のライトとクリアの競合

(2) TCNT のワードライトとカウントアップの競合

TCNT のワードライトサイクル中の T3 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。

このタイミングを図 8.62 に示します。

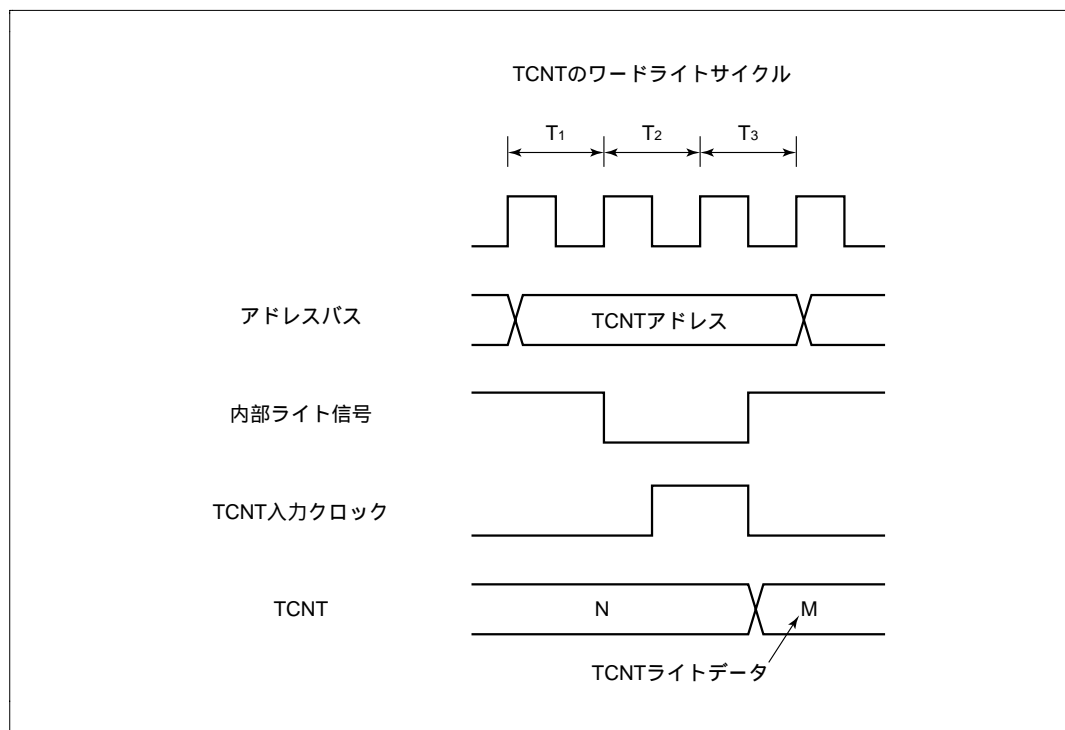


図 8.62 TCNT のワードライトとカウントアップの競合

(3) TCNT のバイトライトとカウントアップの競合

TCNTのバイトライトサイクル中のT2ステートまたはT3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8.63に示します。

TCNTHのバイトライトサイクル中のT2ステートでカウントアップが発生した場合の例です。

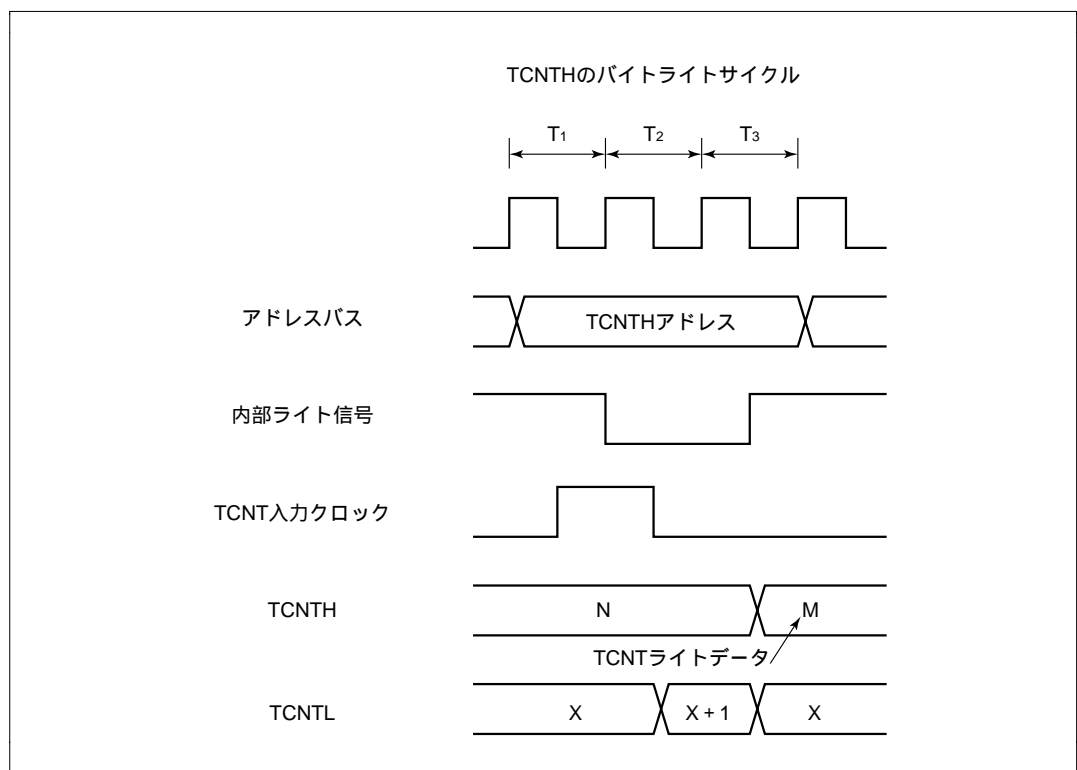


図 8.63 TCNT のバイトライトとカウントアップの競合

(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 T3 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 8.64 に示します。

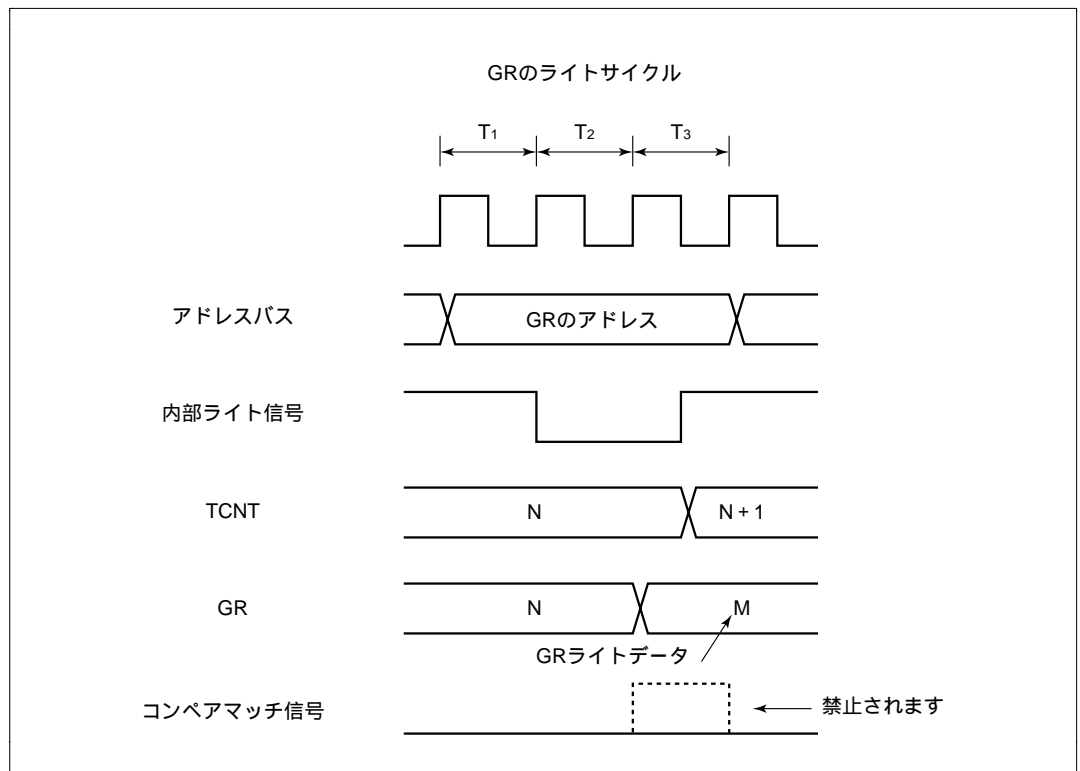


図 8.64 GR のライトとコンペアマッチの競合

(5) TCNT のライトとオーバーフロー / アンダフローとの競合

TCNT のライトサイクル中の T3 ステート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 8.65 に示します。

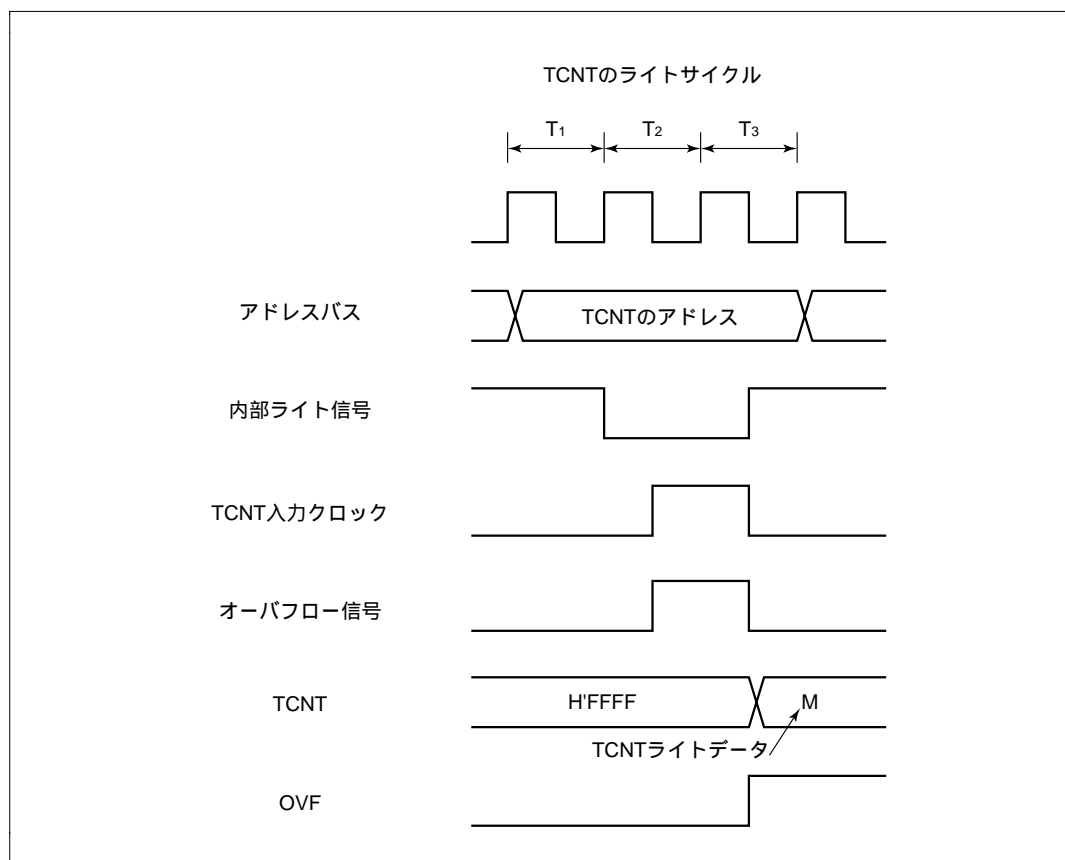


図 8.65 TCNT のライトとオーバーフローの競合

(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 8.66 に示します。

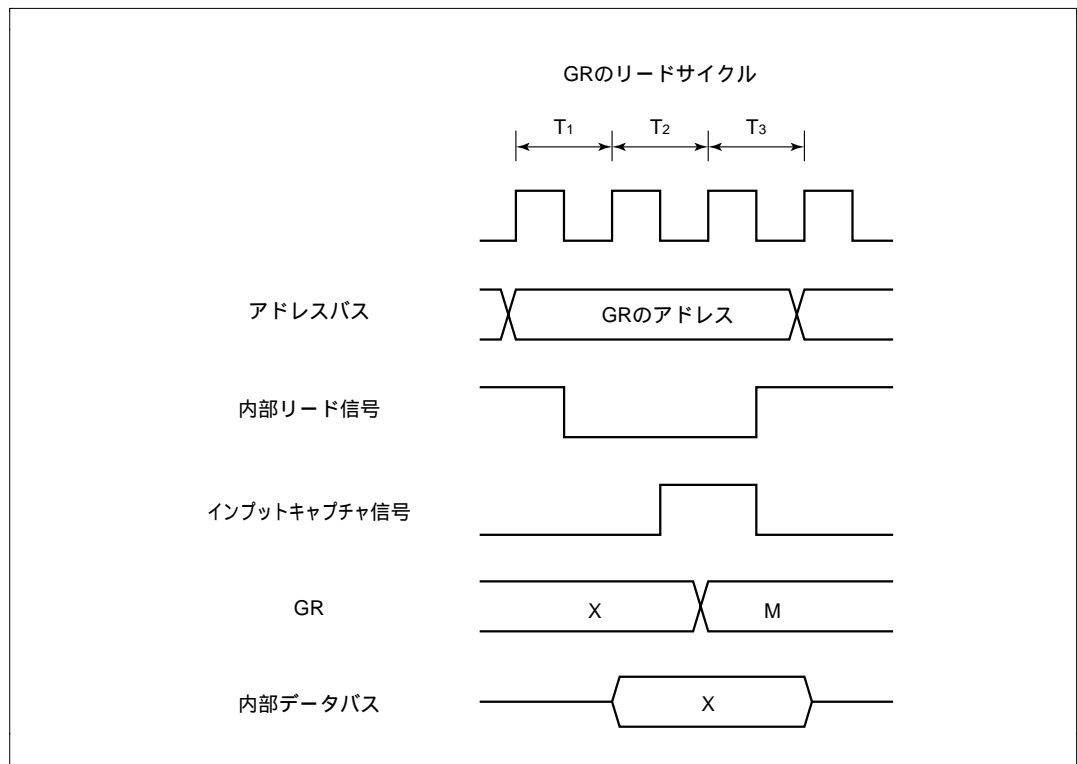


図 8.66 GR のリードとインプットキャプチャの競合

(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。また、GR にはカウンタクリア前の TCNT の内容が転送されます。

このタイミングを図 8.67 に示します。

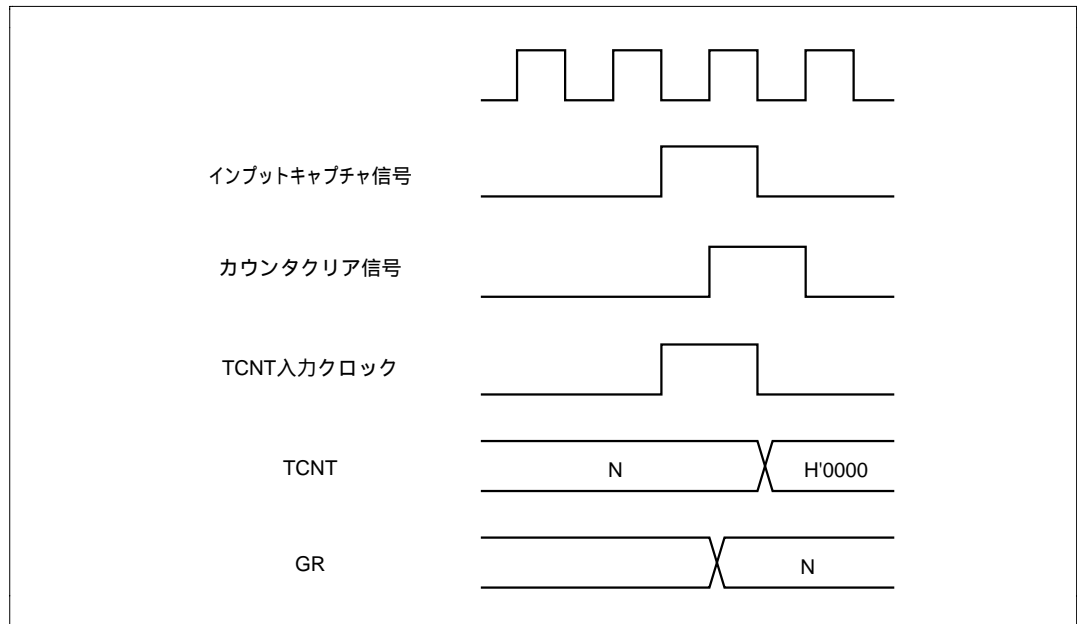


図 8.67 インพุットキャプチャによるカウンタクリアとカウントアップの競合

(8) GR のライトとインプットキャプチャの競合

GR のライトサイクル中の T3 ステートで、インプットキャプチャ信号が発生すると、GR への書込みは行われず、インプットキャプチャが優先されます。

このタイミングを図 8.68 に示します。

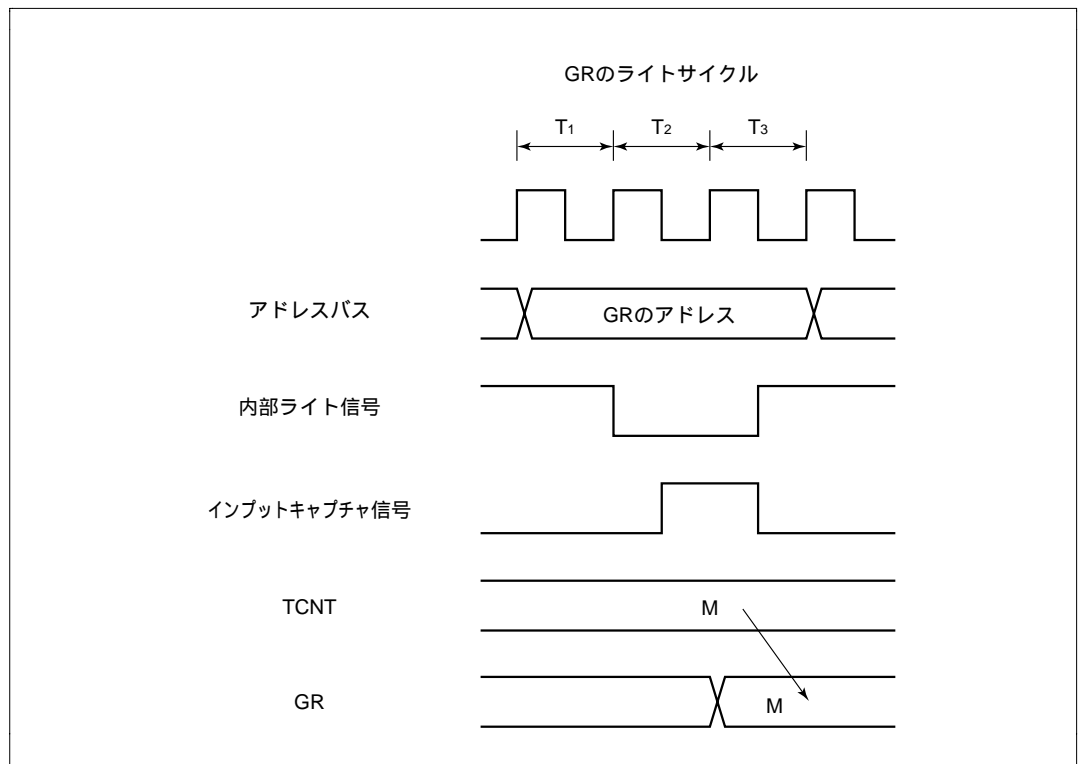


図 8.68 GR のライトとインプットキャプチャの競合

(9) 周期設定上の注意事項

コンパスマッチによるカウンタクリアを設定した場合、TCNT は GR の値と一致した最後のステート (TCNT が一致したカウント値を更新するタイミング) でクリアされます。このため、実際のカウンタ周波数は次の式ようになります。

$$f = \frac{1}{(N+1)}$$

(f : カウンタ周波数、 : 動作周波数、 N : GR の設定値)

(10) BR のライトとインプットキャプチャの競合

BR をインプットキャプチャバッファレジスタとして使用しているとき、ライトサイクル中の T3 ステートでインプットキャプチャ信号が発生すると、BR へのライトは行われずバッファ動作が優先されます。

このタイミングを図 8.69 に示します。

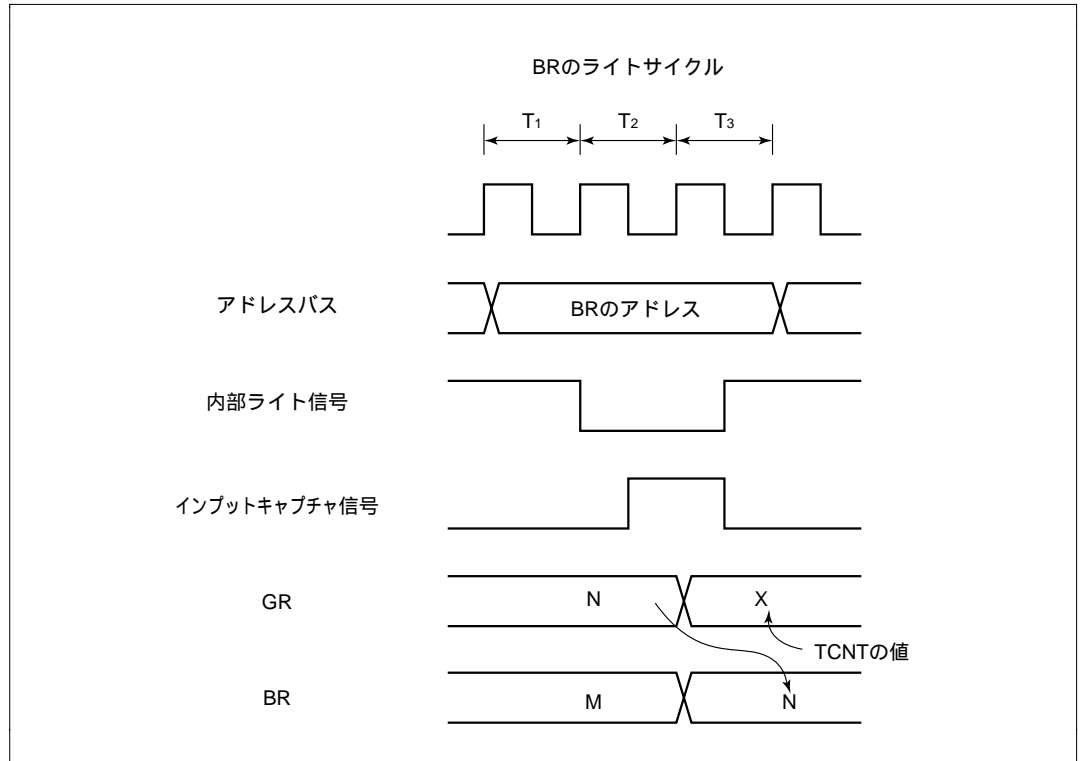
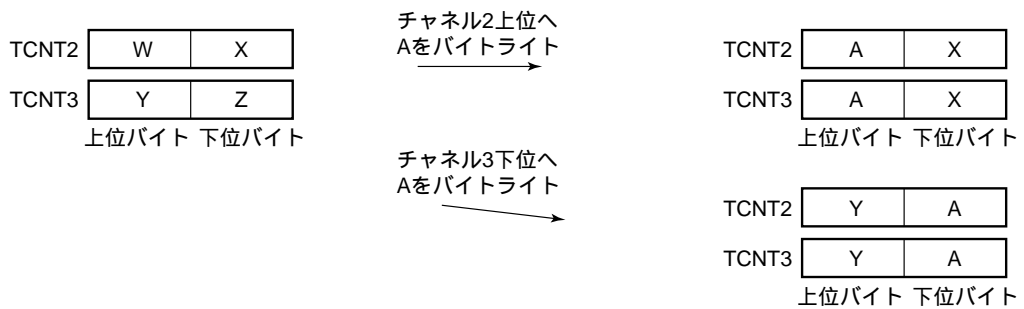


図 8.69 BR のライトとインプットキャプチャの競合

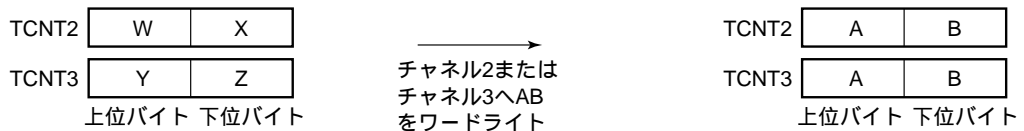
(11) 同期動作時のライト動作に関する注意事項

同期動作を設定した状態で、TCNT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TCNTと、16ビットすべて同じ値となります。

- (例) チャンネル2、3を同期モードで指定した場合
 ・チャンネル2 / チャンネル3へのバイトライト



- ・チャンネル2 / チャンネル3へのワードライト



(12) リセット同期 PWM モード / 相補 PWM モード設定時の注意事項

TFCR の CMD1、CMD0 ビットを設定するときは、次のことに注意してください。

- (1) CMD1、CMD0 ビットへのライトは、TCNT3、TCNT4 が停止中に行ってください。
- (2) リセット同期 PWM モードと相補 PWM モードの相互の設定変更は禁止されています。
 通常動作 (CMD1 ビットを"0"にクリア) に設定した後に、リセット同期 PWM モードまたは相補 PWM モードに設定してください。

(13) ITUの動作モード一覧

表 8.11 (a) ITUの動作モード (チャンネル0)

動作モード	レジスタ設定													
	TSNC		TMDR			TFCR			TOCR		TOER	TIOR0		TCR0
	同期動作	MDF	FDIR	PWM	相補PWM	リセット同期PWM	バッファ動作	XTGD	出力レベルセレクト	マスクイネーブル	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	—	—
カウティング	—	—	—	—	—	—	—	—	—	—	—	—	CCLR1="0" CCLR0="1"	—
リニアタ機能	—	—	—	—	—	—	—	—	—	—	—	—	CCLR1="1" CCLR0="0"	—
同期クリア	SYNC0="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 8.11 (b) ITU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			TFCR		TOCR		TIOR1		TCR1	
	同期動作	MDF	FDIR	PWM	相補PWM	リセット同期PWM	バッファ動作	XTGD	出力レベルレギュレーション	IOA	IOB	クリア選択	クロック選択	
同期プリセット	—	—	—	—	—	—	—	—	—	—	—	—	—	
PWMモード	—	—	—	PWM1="1"	—	—	—	—	—	—	*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"	—	—	—	*2	—	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でクリア	—	—	—	—	—	—	—	—	—	—	—	CCLR1="1" CCLR0="0"	—	
同期クリア機能	—	—	—	—	—	—	—	—	—	—	—	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】 *1 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。

*2 チャネル3とチャネル4が相補PWMモードまたはリセット同期PWMモードで動作している時のみ有効となります。

表 8.11 (c) ITU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			TFCR			TOCR		TOER	TIOR2		TCR2	
	同期動作	MDF	FDIR	PWM	相補PWM	リセット同期PWM	バッファ動作	XTGD	出力レベルセレクト	マスクイネーブル	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		
カウンタリニア機能			—		—	—	—	—	—	—			CCLR1="0" CCLR0="1"	
			—		—	—	—	—	—	—			CCLR1="1" CCLR0="0"	
同期クリア機能	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B信号は禁止されます。

表 8.11 (d) ITUの動作モード (チャンネル3)

動作モード	レジスタ設定													
	TSNC	TMDR		TFRC		TOCR		TOER	TIOR3		TCR3			
	同期動作	MDF	FDIR	PWM	相神PWM	リセット同期PWM	パッファ動作	XTGD	出力レベルレクタ	マスタイネーブル	IOA	IOB	クリア選択	クロック選択
同期プリセット	SYNC3="1"	—	—	—	*3	—	—	—	—	*1	—	—	—	—
PWMモード	—	—	—	PWM3="1"	CMD1="0"	CMD1="0"	—	—	—	—	—	*2	—	—
アウトプットコンベアA機能	—	—	—	—	CMD1="0"	CMD1="0"	—	—	—	—	IOA2="0" 他任意	—	—	—
アウトプットコンベアB機能	—	—	—	—	CMD1="0"	CMD1="0"	—	—	—	—	—	IOB2="0" 他任意	—	—
インプットキャプチャA機能	—	—	—	—	CMD1="0"	CMD1="0"	—	—	—	EA3は無効 他任意	IOA2="1" 他任意	—	—	—
インプットキャプチャB機能	—	—	—	—	CMD1="0"	CMD1="0"	—	—	—	EB3は無効 他任意	—	IOB2="1" 他任意	—	—
コンベアA/インプットキャプチャA機能	—	—	—	—	CMD1="1" CMD0="0" は禁止	*4	—	—	—	*1	—	—	CCLR1="0" CCLR0="1"	—
コンベアB/インプットキャプチャB機能	—	—	—	—	CMD1="0"	CMD1="0"	—	—	—	*1	—	—	CCLR1="1" CCLR0="0"	—
同期クリア機能	SYNC3="1"	—	—	—	CMD1="1" CMD0="0" は禁止	—	—	—	—	*1	—	—	CCLR1="1" CCLR0="1"	—
相補PWMモード	*3	—	—	—	CMD1="1" CMD0="0"	CMD1="1" CMD0="0"	—	*6	—	—	—	—	CCLR1="0" CCLR0="0"	*5
リセット同期PWMモード	—	—	—	—	CMD1="1" CMD0="1"	CMD1="1" CMD0="1"	—	*6	—	—	—	—	CCLR1="0" CCLR0="1"	—
パッファ動作 (BRA)	—	—	—	—	—	—	BFA3="1" 他任意	—	—	*1	—	—	—	—
パッファ動作 (BRB)	—	—	—	—	—	—	BFB3="1" 他任意	—	—	*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】 *1 マスタイネーブルビットの設定は、波形出力動作時のみ有効となります。
 *2 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、コンベアマッチャAとコンベアマッチャBが同時に発生した場合、コンベアマッチャ信号は禁止されます。
 *3 相補PWMモード設定時には、チャンネル3とチャンネル4を同時に同期動作に設定しないでください。
 *4 インプットキャプチャAによるカウンタクリアは、リセット同期PWMモード設定時には使用できません。
 *5 相補PWMモード設定時のクロック選択は、チャンネル3とチャンネル4を同一としてください。
 *6 チャンネル1のインプットキャプチャ機能を使用してください。

表 8.11 (e) ITUの動作モード (チャンネル4)

動作モード	レジスタ設定											
	TSNC 同期動作	TMDR		TFCR		TOCR		TOER	TIOR4		TCR4	
	MDF	FDIR	PWM	相補 PWM	リセット同期 PWM	バッファ 動作	XTGD	出力 レベルセレクト	マスタ イネーブル	IOA	IOB	クリア選択 クロック 選択
同期リセット	SYNC4="1"	—	—	—	*3	—	—	—	*1	—	—	—
PWMモード	—	—	PWM4="1"	CMD1="0"	CMD1="0"	—	—	—	—	—	*2	—
アウトプット コンベンア機能	—	—	PWM4="0"	CMD1="0"	CMD1="0"	—	—	—	—	IOA2="0" 他任意	—	—
アウトプット コンベンB機能	—	—	—	CMD1="0"	CMD1="0"	—	—	—	—	—	IOB2="0" 他任意	—
インプット キャプチャA機能	—	—	PWM4="0"	CMD1="0"	CMD1="0"	—	—	—	E/A1は無効 他任意	IOA2="1" 他任意	—	—
インプット キャプチャB機能	—	—	PWM4="0"	CMD1="0"	CMD1="0"	—	—	—	E/B4は無効 他任意	—	IOB2="1" 他任意	—
相補PWMモード	—	—	—	CMD1="1" CMD0="0" は禁止	*4	—	—	—	*1	—	—	CCLR1="0" CCLR0="1"
リセット同期 PWMモード	—	—	—	CMD1="1" CMD0="0" は禁止	*4	—	—	—	*1	—	—	CCLR1="1" CCLR0="0"
同期クリア 機能	SYNC4="1"	—	—	CMD1="1" CMD0="0" は禁止	*4	—	—	—	*1	—	—	CCLR1="1" CCLR0="1"
相補PWMモード	*3	—	—	CMD1="1" CMD0="0"	CMD1="1" CMD0="0"	—	—	—	—	—	—	CCLR1="0" CCLR0="0"
リセット同期 PWMモード	—	—	—	CMD1="1" CMD0="1"	CMD1="1" CMD0="1"	—	—	—	—	—	—	*6
バッファ動作 (BRA)	—	—	—	—	—	—	—	—	*1	BFA4="1" 他任意	—	—
バッファ動作 (BRB)	—	—	—	—	—	—	—	—	*1	BFB4="1" 他任意	—	—

<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】 *1 マスタイネーブルビットの設定は、波形出力動作時のみ有効となります。
 *2 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、コンベンアとコンベンBが同時に発生した場合、コンベンアマッチ信号は禁止されます。
 *3 相補PWMモード設定時には、チャンネル3とチャンネル4を同時に同期動作設定しないでください。
 *4 リセット同期PWMモード設定時は、カウンタクリア機能は有効ですが、TCNT4は独立動作しています。出力波形には影響しません。
 *5 相補PWMモード設定時のクロック選択は、チャンネル3とチャンネル4を同一としてください。
 *6 リセット同期PWMモード動作時は、TCR4の設定は有効ですが、TCNT4は独立動作しています。出力波形には影響しません。

9. プログラマブルタイミ ングパターンコントローラ (TPC)

第9章 目次

9.1	概要	281
9.1.1	特長	281
9.1.2	ブロック図	282
9.1.3	端子構成	283
9.1.4	レジスタ構成	284
9.2	各レジスタの説明	285
9.2.1	ポート A データディレクションレジスタ (PADDR)	285
9.2.2	ポート A データレジスタ (PADR)	285
9.2.3	ポート B データディレクションレジスタ (PBDDR)	286
9.2.4	ポート B データレジスタ (PBDR)	286
9.2.5	ネクストデータレジスタ A (NDRA)	286
9.2.6	ネクストデータレジスタ B (NDRB)	288
9.2.7	ネクストデータイネーブルレジスタ A (NDERA)	290
9.2.8	ネクストデータイネーブルレジスタ B (NDERB)	290
9.2.9	TPC 出力コントロールレジスタ (TPCR)	291
9.2.10	TPC 出力モードレジスタ (TPMR)	294
9.3	動作説明	297
9.3.1	概要	297
9.3.2	出力タイミング	298
9.3.3	TPC 出力通常動作	299
9.3.4	TPC 出力ノンオーバーラップ動作	301
9.3.5	インプットキャプチャによる TPC 出力	303
9.4	使用上の注意	304

9. プログラマブルタイミングパターンコントローラ (TPC)

9.4.1	TPC 出力端子の動作	304
9.4.2	ノンオーバーラップ動作時の注意	304

9.1 概要

本 LSI は、16 ビットインテグレートドタイマユニット (ITU) 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。

9.1.1 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

ITU の 4 チャンネルのコンペアマッチ信号の中から、グループごとに出力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。

9.1.2 ブロック図

TPCのブロック図を図9.1に示します。

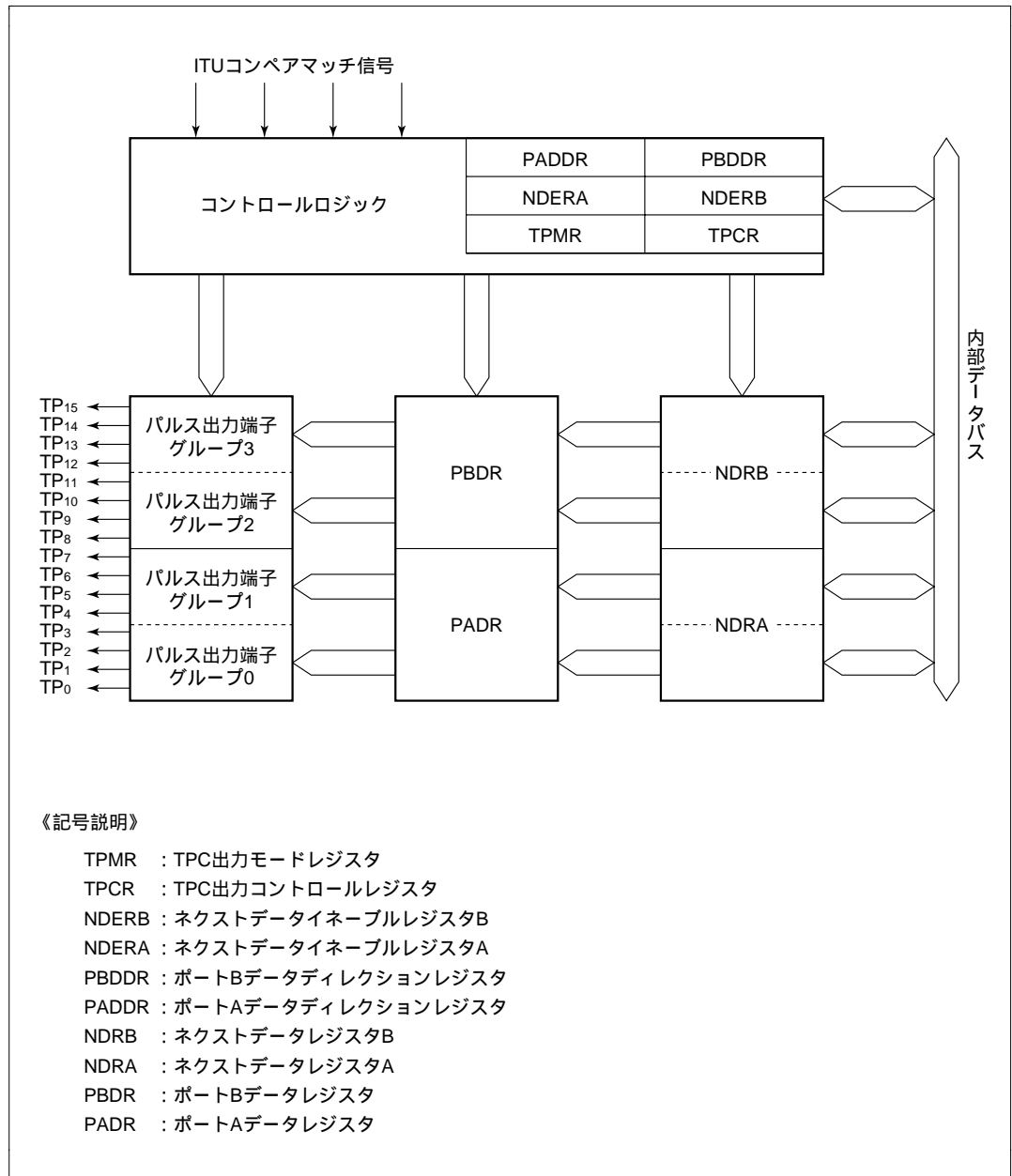


図 9.1 TPCのブロック図

9.1.3 端子構成

TPCの端子構成を表9.1に示します。

表 9.1 端子構成

名称	略称	入出力	機能
TPC 出力 0	TP ₀	出力	グループ 0 のパルス出力
TPC 出力 1	TP ₁	出力	
TPC 出力 2	TP ₂	出力	
TPC 出力 3	TP ₃	出力	
TPC 出力 4	TP ₄	出力	グループ 1 のパルス出力
TPC 出力 5	TP ₅	出力	
TPC 出力 6	TP ₆	出力	
TPC 出力 7	TP ₇	出力	
TPC 出力 8	TP ₈	出力	グループ 2 のパルス出力
TPC 出力 9	TP ₉	出力	
TPC 出力 10	TP ₁₀	出力	
TPC 出力 11	TP ₁₁	出力	
TPC 出力 12	TP ₁₂	出力	グループ 3 のパルス出力
TPC 出力 13	TP ₁₃	出力	
TPC 出力 14	TP ₁₄	出力	
TPC 出力 15	TP ₁₅	出力	

9.1.4 レジスタ構成

TPC のレジスタ構成を表 9.2 に示します。

表 9.2 レジスタ構成

アドレス* ¹	名称	略称	R/W	初期値
H'FFD1	ポート A データディレクションレジスタ	PADDR	W	H'00
H'FFD3	ポート A データレジスタ	PADR	R/(W) ^{*2}	H'00
H'FFD4	ポート B データディレクションレジスタ	PBDDR	W	H'00
H'FFD6	ポート B データレジスタ	PBDR	R/(W) ^{*2}	H'00
H'FFA0	TPC 出力モードレジスタ	TPMR	R/W	H'F0
H'FFA1	TPC 出力コントロールレジスタ	TPCR	R/W	H'FF
H'FFA2	ネクストデータイネーブルレジスタ B	NDERB	R/W	H'00
H'FFA3	ネクストデータイネーブルレジスタ A	NDERA	R/W	H'00
H'FFA5/ H'FFA7* ³	ネクストデータレジスタ A	NDRA	R/W	H'00
H'FFA4/ H'FFA6* ³	ネクストデータレジスタ B	NDRB	R/W	H'00

【注】 *1 アドレスの下位 16 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'FFA5 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'FFA7、グループ 1 に対応する NDRA のアドレスは H'FFA5 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'FFA4 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'FFA6、グループ 3 に対応する NDRB のアドレスは H'FFA4 となります。

9.2 各レジスタの説明

9.2.1 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 TP₇ ~ TP₀ 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、「7.10 ポート A」を参照してください。

9.2.2 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、「7.10 ポート A」を参照してください。

9.2.3 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 TP₁₅ ~ TP₈ 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、「7.11 ポート B」を参照してください。

9.2.4 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、「7.11 ポート B」を参照してください。

9.2.5 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 (TP₇ ~ TP₀ 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 ITU 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'FFA5 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'FFA7 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。

(1) アドレス : H'FFA5

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
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。
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。

(2) アドレス : H'FFA7

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'FFA5、NDRA の下位 4 ビット (グループ 0) のアドレスは H'FFA7 となります。このとき、アドレス H'FFA5 のビット 3~0、アドレス H'FFA7 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。

(1) アドレス : H'FFA5

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
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。
リザーブビット

(2) アドレス : H'FFA7

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。

9.2.6 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 (TP₁₅ ~ TP₈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 ITU 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'FFA4 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'FFA6 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。

(1) アドレス : H'FFA4

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
出力データを格納します。

ネクストデータ11~8
TPC出力グループ2の次の
出力データを格納します。

(2) アドレス : H'FFA6

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'FFA4、NDRB の下位 4 ビット (グループ 2) のアドレスは H'FFA6 となります。このとき、アドレス H'FFA4 のビット 3~0、アドレス H'FFA6 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。

(1) アドレス : H'FFA4

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
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。

リザーブビット

(2) アドレス : H'FFA6

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。

9.2.7 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 (TP₇ ~ TP₀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の許可/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 ITU 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 (TP₇ ~ TP₀端子) の許可/禁止をビット単位で選択します。

ビット7~0	
NDER7 ~ NDER0	説明
0	TPC 出力 TP ₇ ~ TP ₀ を禁止 (NDR7 ~ NDR0 から PA ₇ ~ PA ₀ への転送禁止) (初期値)
1	TPC 出力 TP ₇ ~ TP ₀ を許可 (NDR7 ~ NDR0 から PA ₇ ~ PA ₀ への転送許可)

9.2.8 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 (TP₁₅ ~ TP₈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の許可/禁止を
選択するビットです。

NDRB により TPC 出力が許可されたビットは、TPCR で選択された ITU のコンペアマッチが発生すると、NDRB 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、NDRB から PBDR への転送は行われず出力値も変化しません。

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。

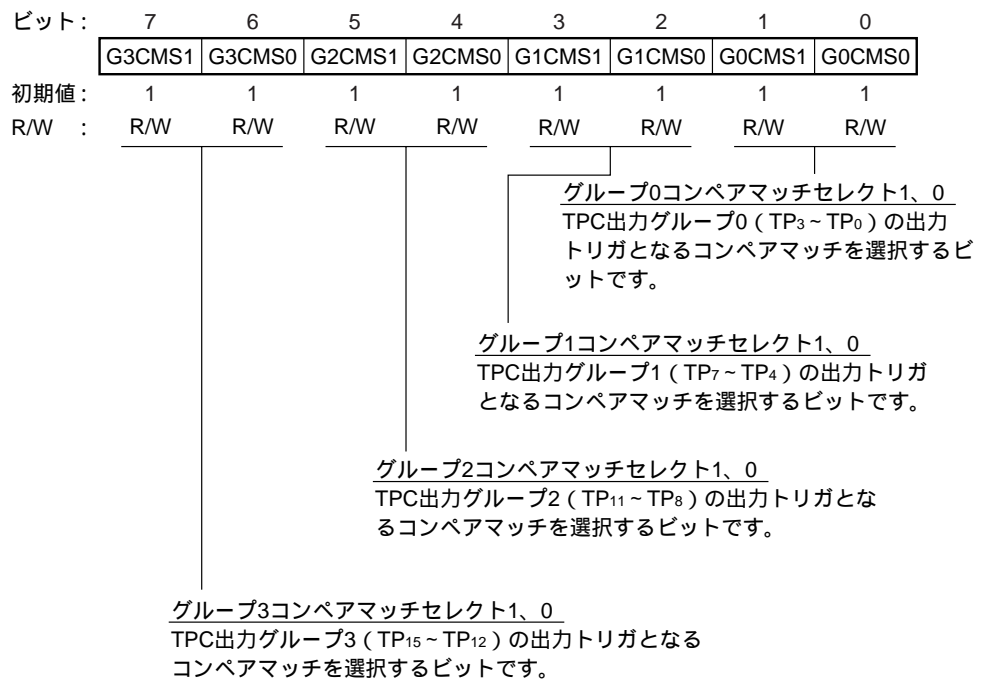
ビット7~0：ネクストデータイネーブル15~8 (NDR15~NDR8)

TPC 出力グループ3、2 (TP₁₅~TP₈端子) の許可/禁止をビット単位で選択します。

ビット7~0	
NDR15~NDR8	説明
0	TPC 出力 TP ₁₅ ~TP ₈ を禁止 (NDR15~NDR8 から PB ₇ ~PB ₀ への転送禁止) (初期値)
1	TPC 出力 TP ₁₅ ~TP ₈ を許可 (NDR15~NDR8 から PB ₇ ~PB ₀ への転送許可)

9.2.9 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。



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。

9. プログラマブルタイミングパターンコントローラ (TPC)

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 (TP₁₅ ~ TP₁₂ 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 ₁₅ ~ TP ₁₂ 端子) の出力トリガは、ITUチャンネル0のコンペアマッチ
	1	TPC 出力グループ3 (TP ₁₅ ~ TP ₁₂ 端子) の出力トリガは、ITU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 ₁₅ ~ TP ₁₂ 端子) の出力トリガは、ITUチャンネル2のコンペアマッチ
	1	TPC 出力グループ3 (TP ₁₅ ~ TP ₁₂ 端子) の出力トリガは、ITUチャンネル3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 (TP₁₁ ~ TP₈ 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
G2CMS1	G2CMS0	
0	0	TPC 出力グループ2 (TP ₁₁ ~ TP ₈ 端子) の出力トリガは、ITUチャンネル0のコンペアマッチ
	1	TPC 出力グループ2 (TP ₁₁ ~ TP ₈ 端子) の出力トリガは、ITU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 ₁₁ ~ TP ₈ 端子) の出力トリガは、ITUチャンネル2のコンペアマッチ
	1	TPC 出力グループ2 (TP ₁₁ ~ TP ₈ 端子) の出力トリガは、ITUチャンネル3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 (TP₇~TP₄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 ₇ ~TP ₄ 端子) の出力トリガは、ITU チャンネル0 のコンペアマッチ
	1	TPC 出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガは、ITU 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 ₇ ~TP ₄ 端子) の出力トリガは、ITU チャンネル2 のコンペアマッチ
	1	TPC 出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガは、ITU チャンネル3 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 (TP₃~TP₀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 ₃ ~TP ₀ 端子) の出力トリガは、ITU チャンネル0 のコンペアマッチ
	1	TPC 出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガは、ITU 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 ₃ ~TP ₀ 端子) の出力トリガは、ITU チャンネル2 のコンペアマッチ
	1	TPC 出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガは、ITU チャンネル3 のコンペアマッチ (初期値)

9.2.10 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。

ビット:	7	6	5	4	3	2	1	0
	—	—	—	—	G3NOV	G2NOV	G1NOV	G0NOV
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

グループ3ノンオーバーラップ
TPC出力グループ3 (TP₁₅ ~ TP₁₂) のノンオーバーラップ動作を設定するビットです。

グループ2ノンオーバーラップ
TPC出力グループ2 (TP₁₁ ~ TP₈) のノンオーバーラップ動作を設定するビットです。

グループ1ノンオーバーラップ
TPC出力グループ1 (TP₇ ~ TP₄) のノンオーバーラップ動作を設定するビットです。

グループ0ノンオーバーラップ
TPC出力グループ0 (TP₃ ~ TP₀) のノンオーバーラップ動作を設定するビットです。

ノンオーバーラップ動作の TPC 出力は、出力トリガとなる ITU 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、「9.3.4 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 (TP₁₅ ~ TP₁₂ 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	
G3NOV	説明
0	TPC 出力グループ 3 は、通常動作 (選択された ITU 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 ITU 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 (TP₁₁ ~ TP₈ 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	
G2NOV	説明
0	TPC 出力グループ 2 は、通常動作 (選択された ITU 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 ITU 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 (TP₇ ~ TP₄ 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	
G1NOV	説明
0	TPC 出力グループ 1 は、通常動作 (選択された ITU 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 ITU 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。)

9. プログラマブルタイミングパターンコントローラ (TPC)

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 (TP₃ ~ TP₀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	
G0NOV	説明
0	TPC 出力グループ0 は、通常動作 (選択された ITU 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 ITU 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。)

9.3 動作説明

9.3.1 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 9.2 に示します。また、TPC 動作条件を表 9.3 に示します。

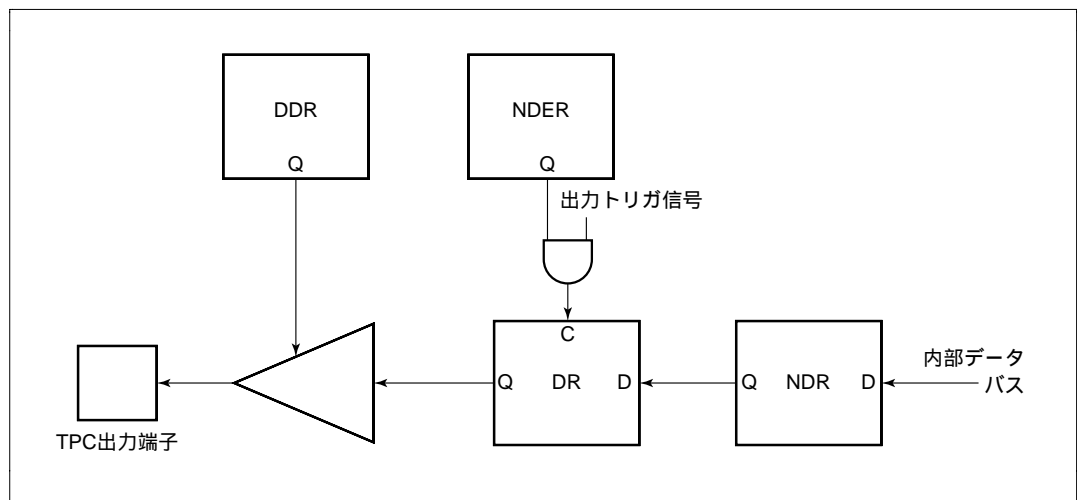


図 9.2 TPC 出力動作

表 9.3 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、「9.3.4 TPC 出力ノンオーバーラップ動作」を参照してください。

9.3.2 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 9.3 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。

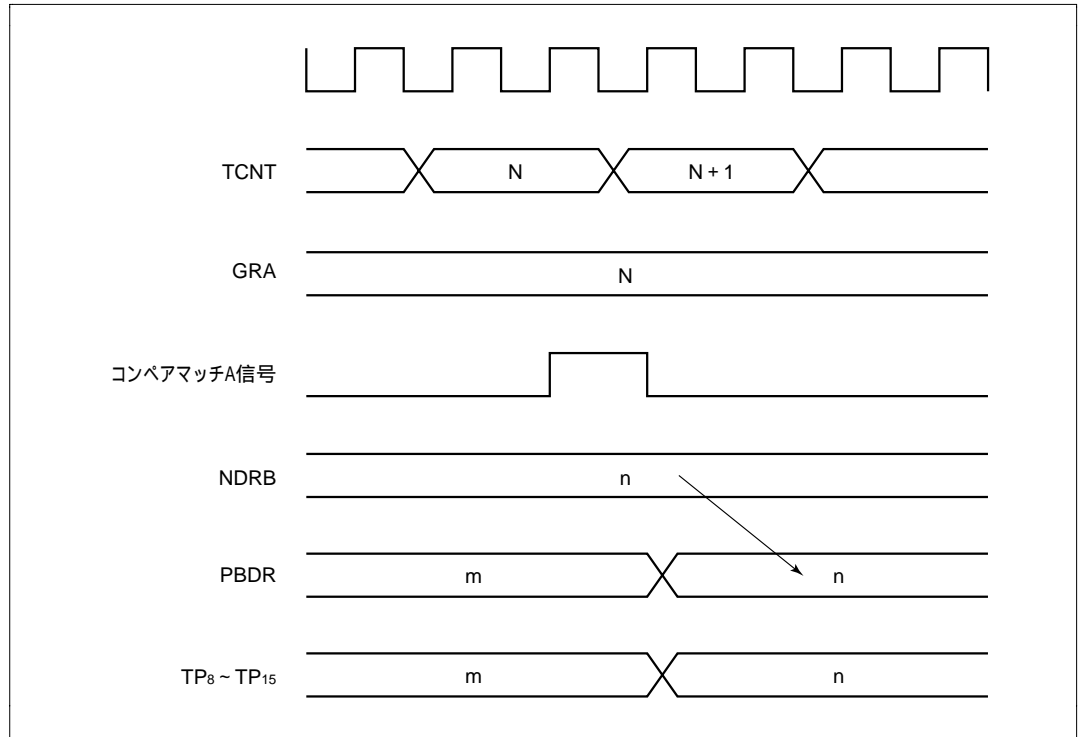


図 9.3 NDR の内容が転送・出力されるタイミング (例)

9.3.3 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 9.4 に示します。

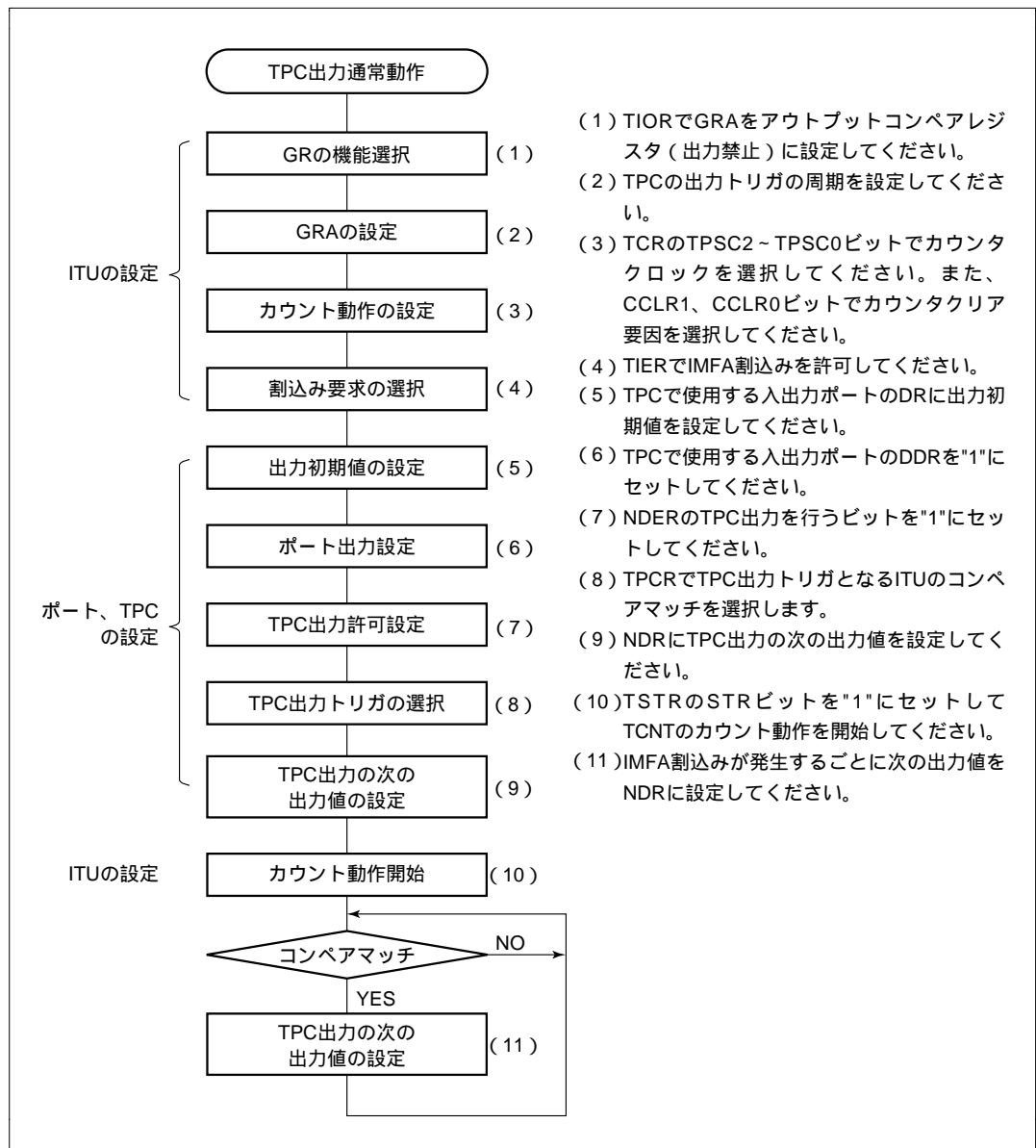
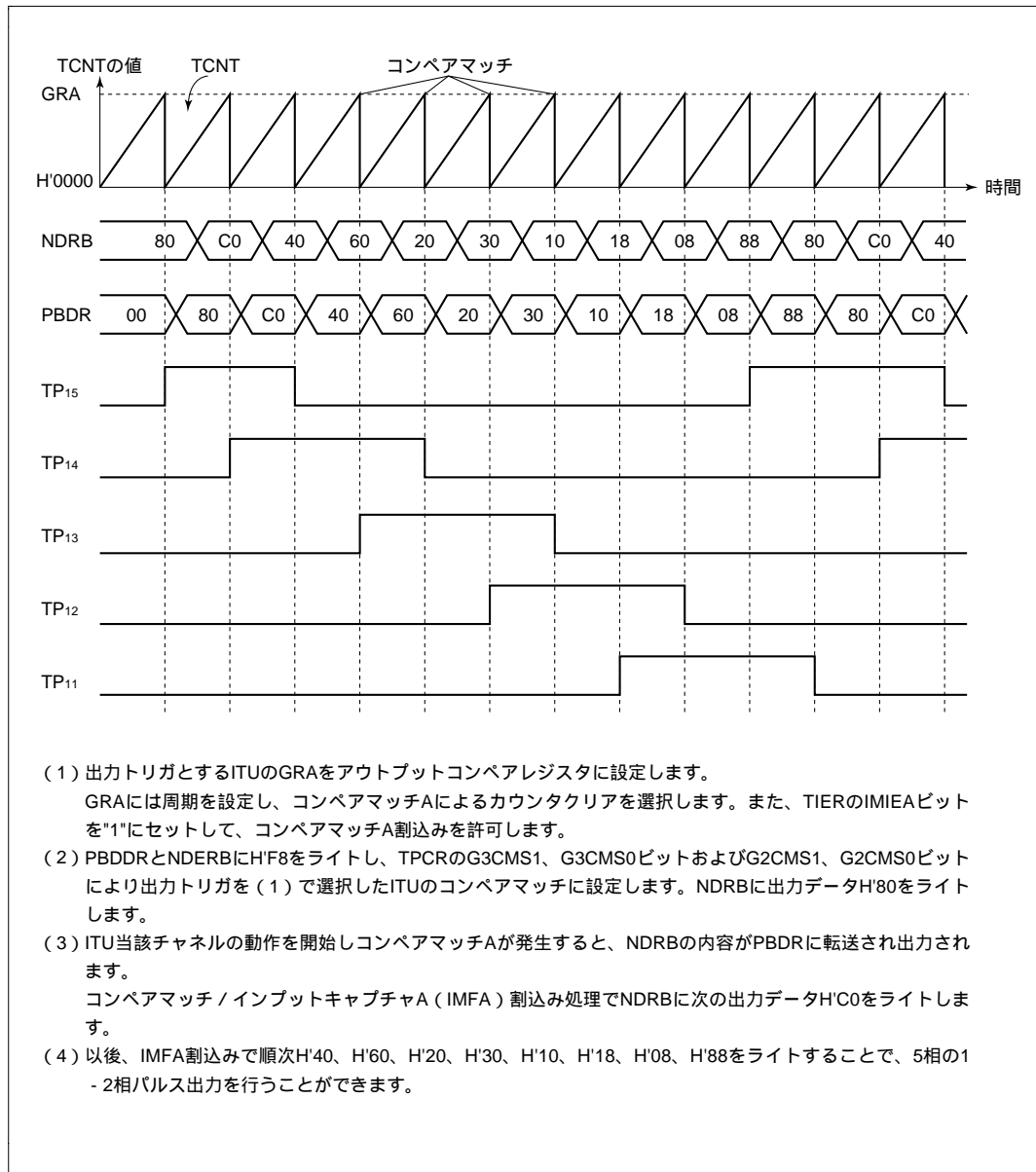


図 9.4 TPC 出力通常動作の設定手順例

(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 9.5 に示します。



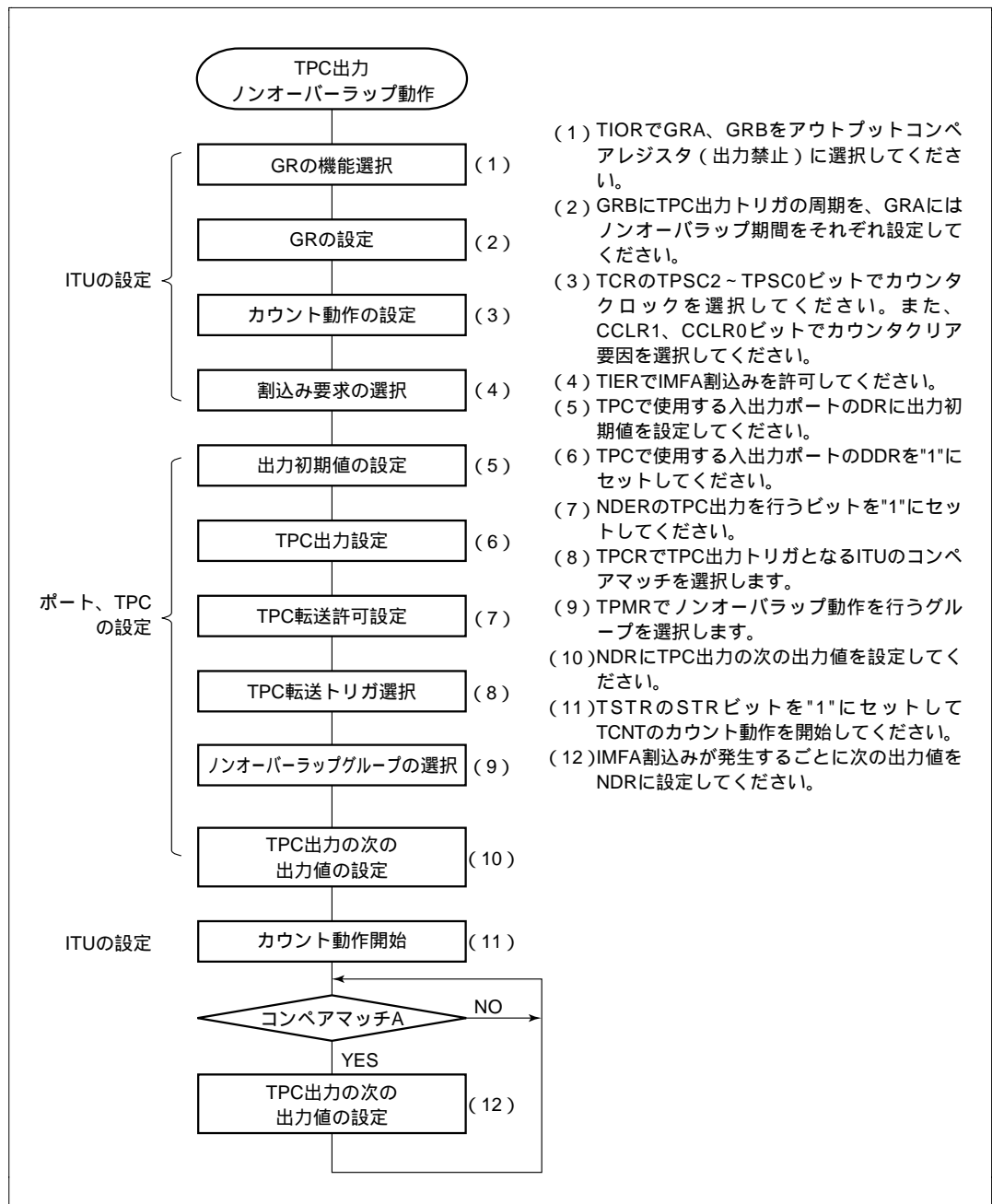
- (1) 出力トリガとするITUのGRAをアウトプットコンペアレジスタに設定します。
GRAには周期を設定し、コンペアマッチAによるカウンタクリアを選択します。また、TIERのIMIEAビットを"1"にセットして、コンペアマッチA割込みを許可します。
- (2) PBDDRとNDRBにH'F8をライトし、TPCRのG3CMS1、G3CMS0ビットおよびG2CMS1、G2CMS0ビットにより出力トリガを(1)で選択したITUのコンペアマッチに設定します。NDRBに出力データH'80をライトします。
- (3) ITU当該チャンネルの動作を開始しコンペアマッチAが発生すると、NDRBの内容がPBDR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。

図 9.5 TPC 出力通常動作例 (5 相パルス出力例)

9.3.4 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 9.6 に示します。



- (1) TIORでGRA、GRBをアウトプットコンペアレジスタ（出力禁止）に選択してください。
- (2) GRBにTPC出力トリガの周期を、GRAにはノンオーバーラップ期間をそれぞれ設定してください。
- (3) TCRのTPSC2～TPSC0ビットでカウンタクロックを選択してください。また、CCLR1、CCLR0ビットでカウンタクリア要因を選択してください。
- (4) TIERでIMFA割込みを許可してください。
- (5) TPCで使用する入出力ポートのDRIに出力初期値を設定してください。
- (6) TPCで使用する入出力ポートのDDRを"1"にセットしてください。
- (7) NDERのTPC出力を行うビットを"1"にセットしてください。
- (8) TPCRでTPC出力トリガとなるITUのコンペアマッチを選択します。
- (9) TPMRでノンオーバーラップ動作を行うグループを選択します。
- (10) NDRにTPC出力の次の出力値を設定してください。
- (11) TSTRのSTRビットを"1"にセットしてTCNTのカウント動作を開始してください。
- (12) IMFA割込みが発生するごとに次の出力値をNDRに設定してください。

図 9.6 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 9.7 に示します。

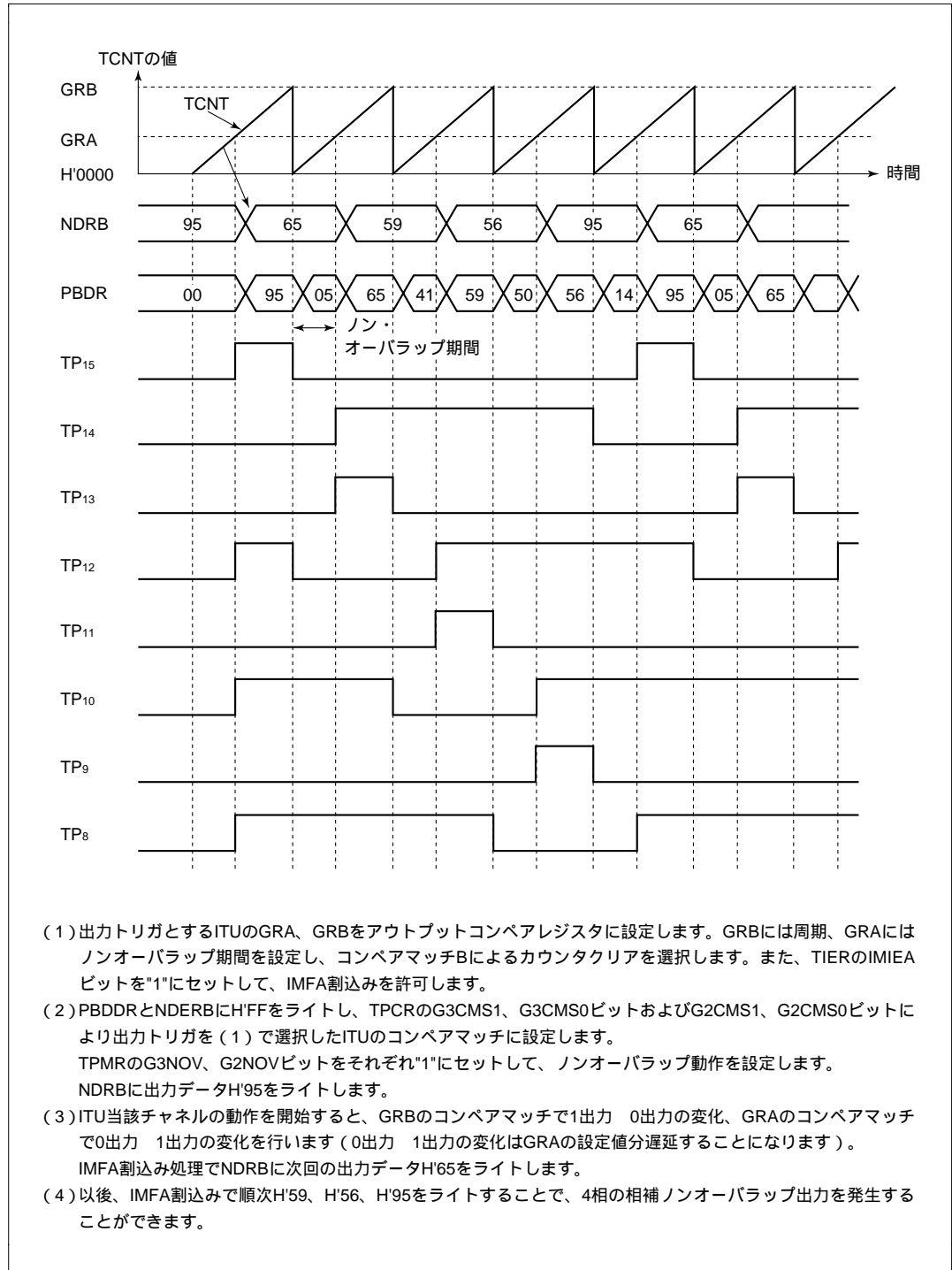


図 9.7 TPC 出力ノンオーバーラップ動作例 (4 相の相補ノンオーバーラップ出力例)

9.3.5 インพุットキャプチャによる TPC 出力

TPC 出力は、ITU のコンペアマッチだけではなく、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 ITU 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 9.8 に示します。

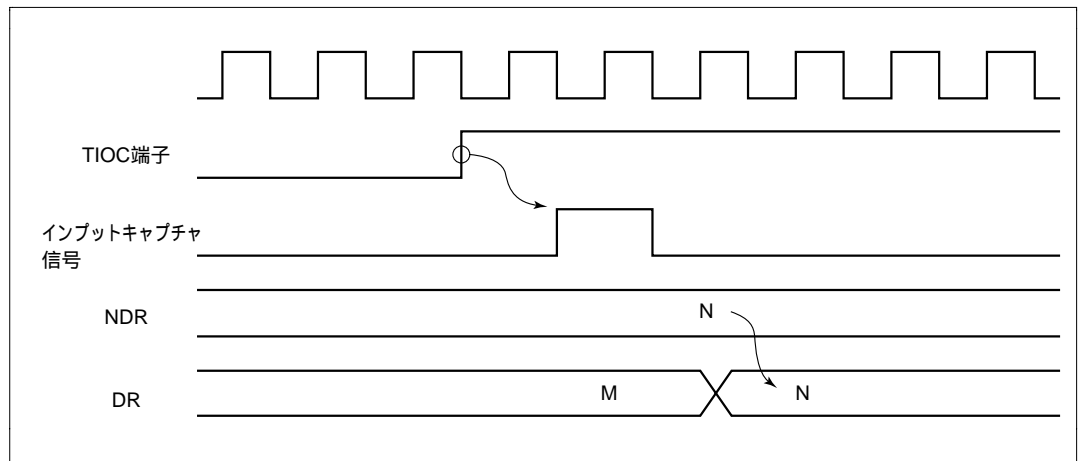


図 9.8 インพุットキャプチャによる TPC 出力例

9.4 使用上の注意

9.4.1 TPC 出力端子の動作

TP₀ ~ TP₁₅は ITU との兼用端子になっています。この端子は、ITU 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。

9.4.2 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 9.9 に示します。

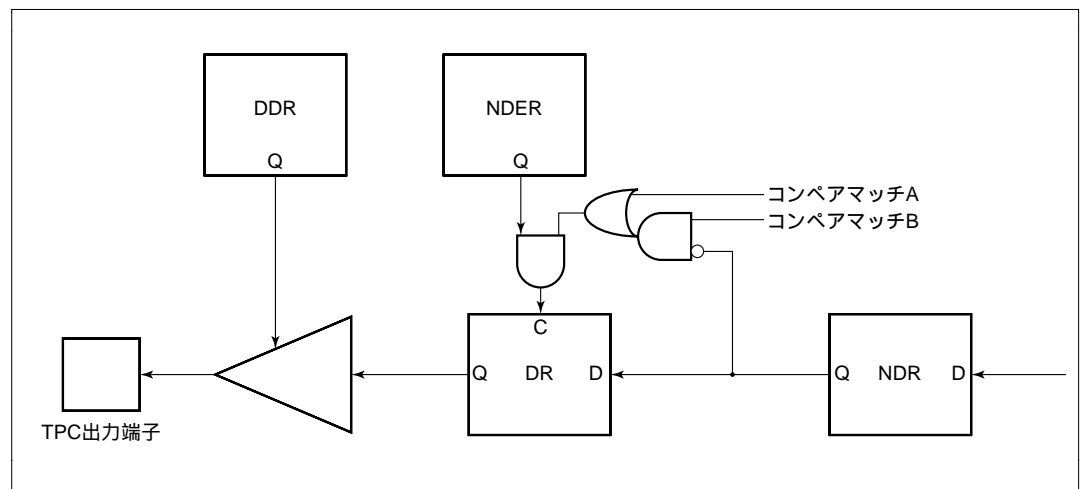


図 9.9 TPC 出力ノンオーバーラップ動作

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。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 9.10 に示します。

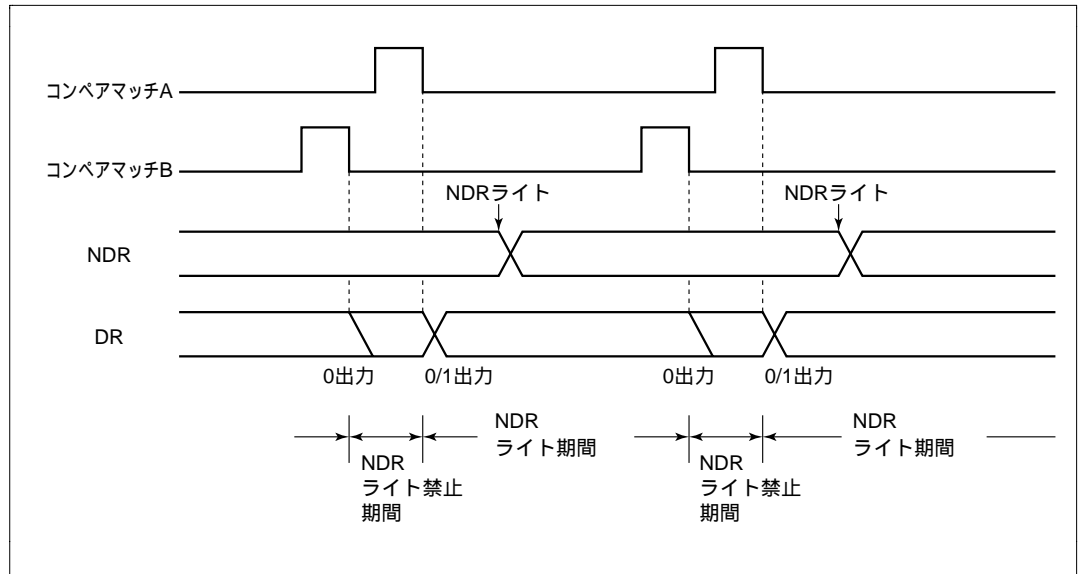


図 9.10 ノンオーバーラップ動作と NDR ライトタイミング

10. ウォッチドッグタイマ

第10章 目次

10.1	概要.....	309
	10.1.1	特長..... 309
	10.1.2	ブロック図..... 310
	10.1.3	端子構成..... 310
	10.1.4	レジスタ構成..... 311
10.2	各レジスタの説明.....	312
	10.2.1	タイマカウンタ (TCNT)..... 312
	10.2.2	タイマコントロール/ステータスレジスタ (TCSR)..... 312
	10.2.3	リセットコントロール/ステータスレジスタ (RSTCSR)..... 314
	10.2.4	レジスタ書換え時の注意..... 315
10.3	動作説明.....	318
	10.3.1	ウォッチドッグタイマ時の動作..... 318
	10.3.2	インターバルタイマ時の動作..... 319
	10.3.3	オーバフローフラグ (OVF) セットタイミング..... 320
	10.3.4	ウォッチドッグタイマリセット (WRST) のセットタイミング..... 321
10.4	割込み.....	322
10.5	使用上の注意.....	322

10.1 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。

10.1.1 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

$/2$ 、 $/32$ 、 $/64$ 、 $/108$ 、 $/256$ 、 $/512$ 、 $/2048$ 、 $/4096$

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、同時にリセット信号を外部に出力可能

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。同時に、 $\overline{\text{RESO}}$ 端子からリセット信号を外部に出力し、システム全体をリセットすることができます。

10.1.2 ブロック図

図 10.1 に WDT のブロック図を示します。

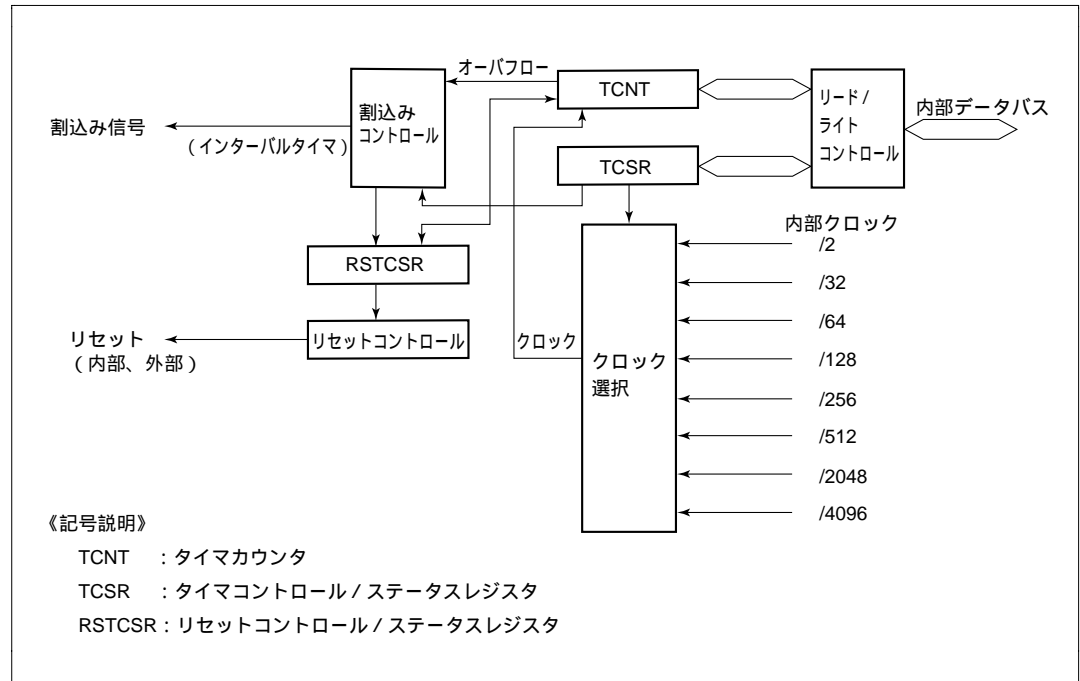


図 10.1 WDT のブロック図

10.1.3 端子構成

WDT で使用する出力端子を表 10.1 に示します。

表 10.1 端子構成

名称	略称	入出力	機能
リセット出力	$\overline{RES0}$	出力*	ウォッチドッグタイマのリセット信号の外部出力

【注】 * オープンドレイン出力端子です。

10.1.4 レジスタ構成

表 10.2 に WDT のレジスタ構成を示します。

表 10.2 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'FFA8	H'FFA8	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'FFA9	タイマカウンタ	TCNT	R/W	H'00
H'FFAA	H'FFAB	リセットコントロール/ ステータスレジスタ	RSTCSR	R/(W)*3	H'3F

【注】 *1 アドレスの下位 16 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。

10.2 各レジスタの説明

10.2.1 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

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。

【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「10.2.4 レジスタ書換え時の注意」を参照してください。

10.2.2 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*1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。

ビット:	7	6	5	4	3	2	1	0
	OVF	WT/IT	TME			CKS2	CKS1	CKS0
初期値:	0	0	0	1	1	0	0	0
R/W :	R/(W)*2	R/W	R/W			R/W	R/W	R/W

クロックセレクト
TCNTに入力するクロックを選択するビットです。

リザーブビット

タイマイネーブル
TCNTの動作/停止を選択するビットです。

タイマモードセレクト
モードを選択するビットです。

オーバーフローフラグ
オーバーフローしたことを示すステータスフラグです。

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

ソフトウェアスタンバイモードに移る直前の値を保持します。

【注】 *1 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「10.2.4 レジスタ書換え時の注意」を参照してください。

*2 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：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	
OVF	説明
0	〔クリア条件〕 OVF = "1"の状態、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/ \bar{W} T)

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	
WT/ \bar{W} T	説明
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。

ビット5	
TME	説明
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。

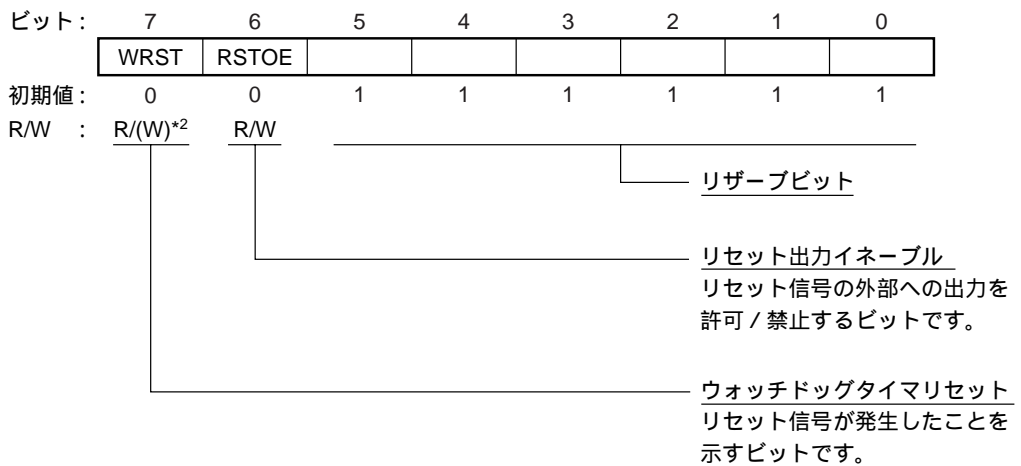
ビット2~0：クロックセレクト2~0 (CKS2~0)

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)
		1	/32
	1	0	/64
		1	/108
1	0	0	/256
		1	/512
	1	0	/2048
		1	/4096

10.2.3 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*1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のモニタ、およびリセット信号の外部への出力を制御します。



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。

【注】 *1 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「10.2.4 レジスタ書換え時の注意」を参照してください。

*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。

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。同時に、RSTOE ビットが "1" にセットされていると、このリセット信号を $\overline{\text{RES}}$ 端子から "Low" レベルを外部に出力し、システム全体のイニシャライズを行うことができます。

ビット7		
WRST	説明	
0	〔クリア条件〕 $\overline{\text{RES}}$ 端子によるリセット信号または "0" をライトしたとき	(初期値)
1	〔セット条件〕 ウォッチドッグタイマ時に、TCNT がオーバフローし、リセット信号が発生したとき	

ビット6：リセット出力イネーブル（RSTOE）

ウォッチドッグタイマ時に TCNT がオーバフローして発生したリセット信号の $\overline{\text{RES}}$ 端子からの出力の許可 / 禁止を選択します。

ビット6		
RSTOE	説明	
0	リセット信号の外部出力を禁止	(初期値)
1	リセット信号の外部出力を許可	

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。

10.2.4 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 10.2 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 ヘライトされます。

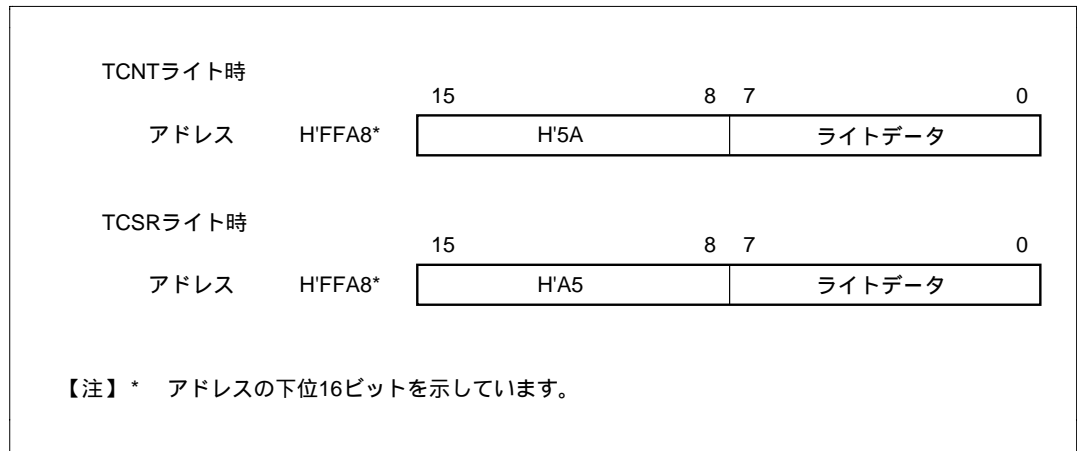


図 10.2 TCNT、TCSR へのライトデータ

(2) RSTCSR へのライト

RSTCSR ヘライトする場合、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 10.3 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 ビットヘライトされ、WRST ビットが"0"にクリアされます。

RSTOE ビットヘライトする場合、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 ビットヘライトされます。

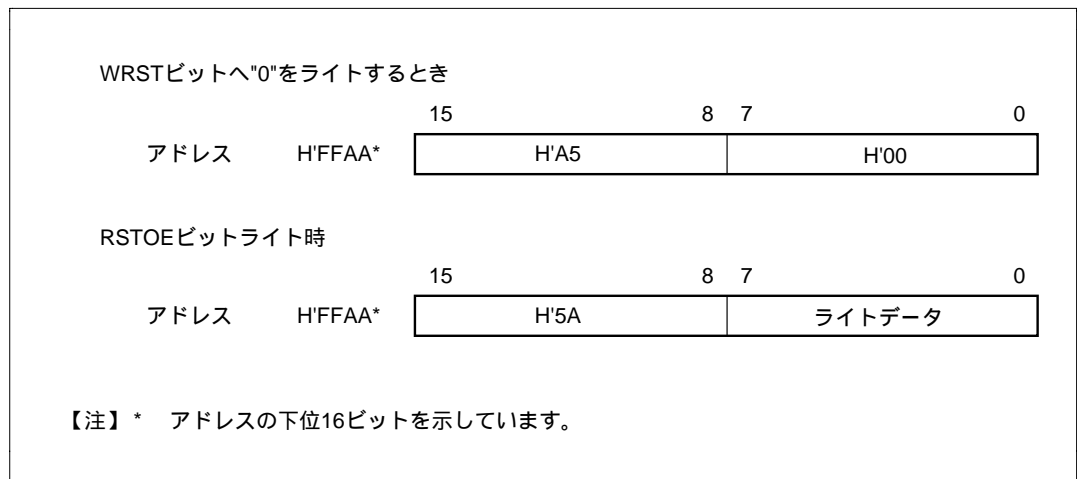


図 10.3 RSTCSR へのライトデータ

(3) TCNT、TCSR、RSTCSR のリード

TCNT、TCSR、RSTCSR をリードする場合、アドレス H'FFA8 に TCSR、H'FFA9 に TCNT、H'FFAB 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 10.3 に TCNT、TCSR、RSTCSR のリードを示します。

表 10.3 TCNT、TCSR、RSTCSR のリード

アドレス*	レジスタ
H'FFA8	TCSR
H'FFA9	TCNT
H'FFAB	RSTCSR

【注】 * アドレスの下位 16 ビットを示しています。

10.3 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。

10.3.1 ウォッチドッグタイマ時の動作

図 10.4 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 $WT\overline{IT}$ 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 によるリセット信号は、 $\overline{RES0}$ 端子から外部に出力し、外部システムをリセットすることができます。外部へのリセット信号は、132 ステート期間出力されます。外部への出力の許可 / 禁止は、RSTCSR の RSTOE ビットによって選択します。

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。

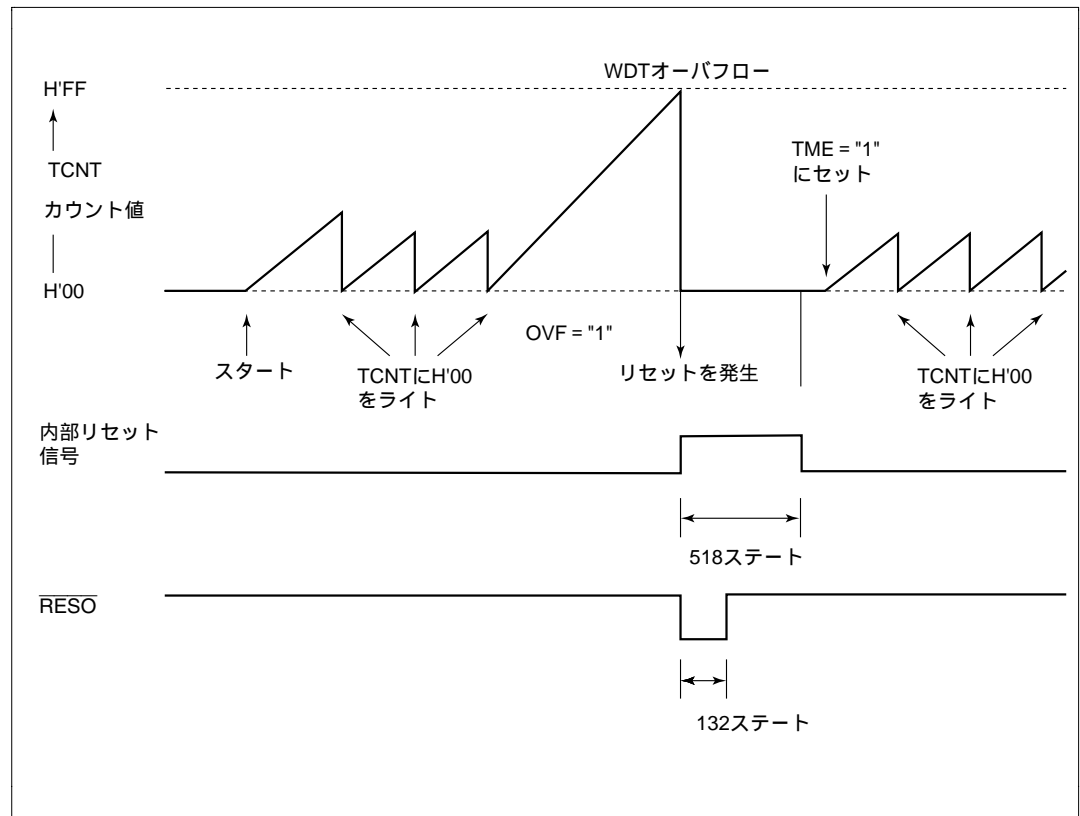


図 10.4 ウォッチドッグタイマモード時の動作

10.3.2 インターバルタイマ時の動作

図 10.5 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/\overline{IT} 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。

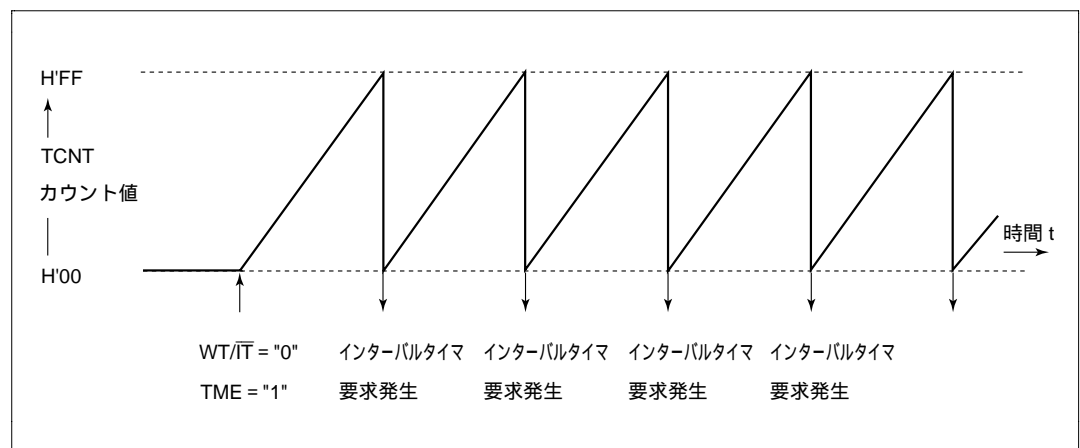


図 10.5 インターバルタイマ時の動作

10.3.3 オーバフローフラグ (OVF) セットタイミング

図 10.6 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。

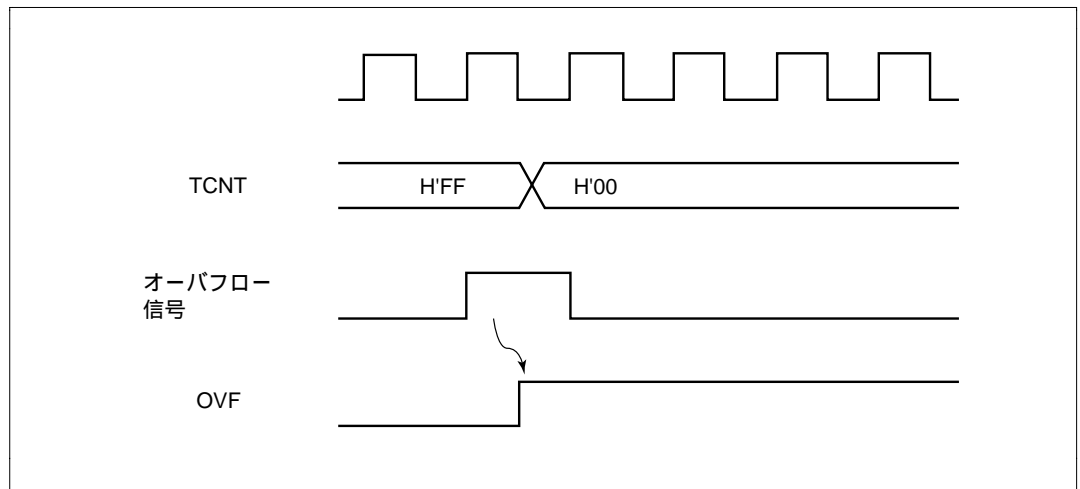


図 10.6 OVF フラグのセットタイミング

10.3.4 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 \bar{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 10.7 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。

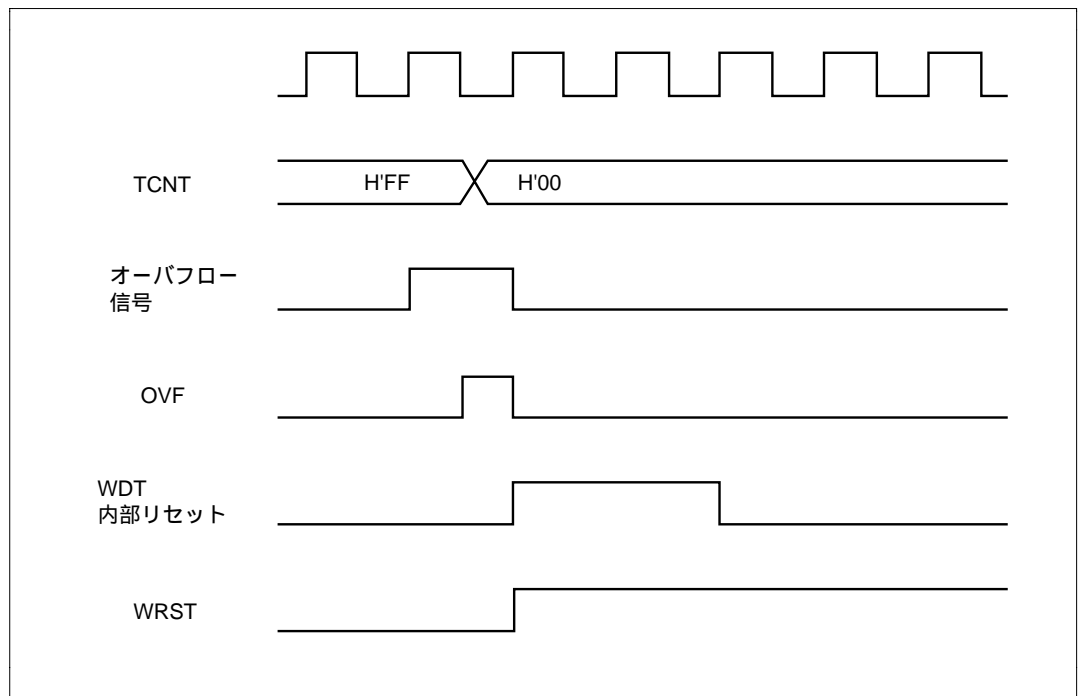


図 10.7 WRST ビットのセットおよび内部リセットタイミング

10.4 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。

10.5 使用上の注意

(1) TCNT のライトとカウントアップの競合

図 10.8 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。

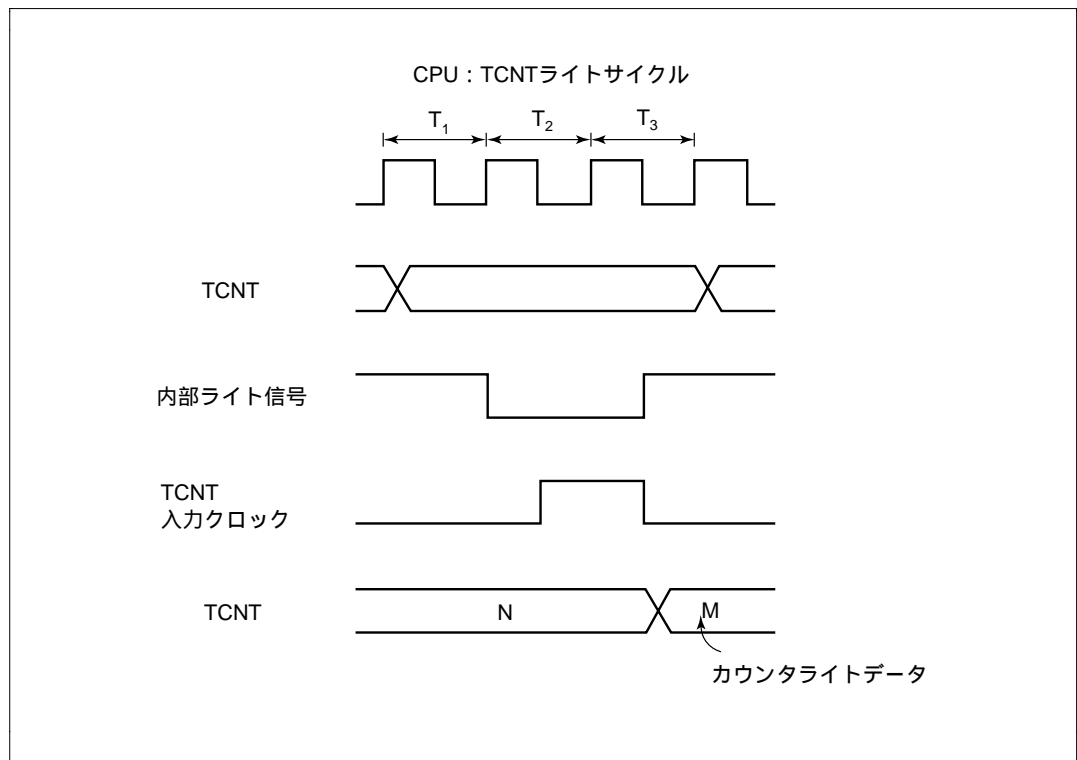


図 10.8 TCNT のライトとカウントアップの競合

(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。

11. シリアルコミュニケーションインタフェース

第11章 目次

11.1	概要	325
11.1.1	特長	325
11.1.2	ブロック図	326
11.1.3	端子構成	327
11.1.4	レジスタ構成	327
11.2	各レジスタの説明	328
11.2.1	レシーブシフトレジスタ (RSR)	328
11.2.2	レシーブデータレジスタ (RDR)	328
11.2.3	トランスミットシフトレジスタ (TSR)	329
11.2.4	トランスミットデータレジスタ (TDR)	329
11.2.5	シリアルモードレジスタ (SMR)	330
11.2.6	シリアルコントロールレジスタ (SCR)	334
11.2.7	シリアルステータスレジスタ (SSR)	338
11.2.8	ビットレートレジスタ (BRR)	344
11.3	動作説明	352
11.3.1	概要	352
11.3.2	調歩同期式モード時の動作	354
11.3.3	マルチプロセッサ通信機能	363
11.3.4	クロック同期式モード時の動作	369
11.4	SCI 割込み	378
11.5	使用上の注意	379

11.1 概要

本 LSI は、独立した 1 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。

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。

11.1.1 特長

SCI 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。

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 種類の割り込み要因

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。

11.1.2 ブロック図

図 11.1 に SCI のブロック図を示します。

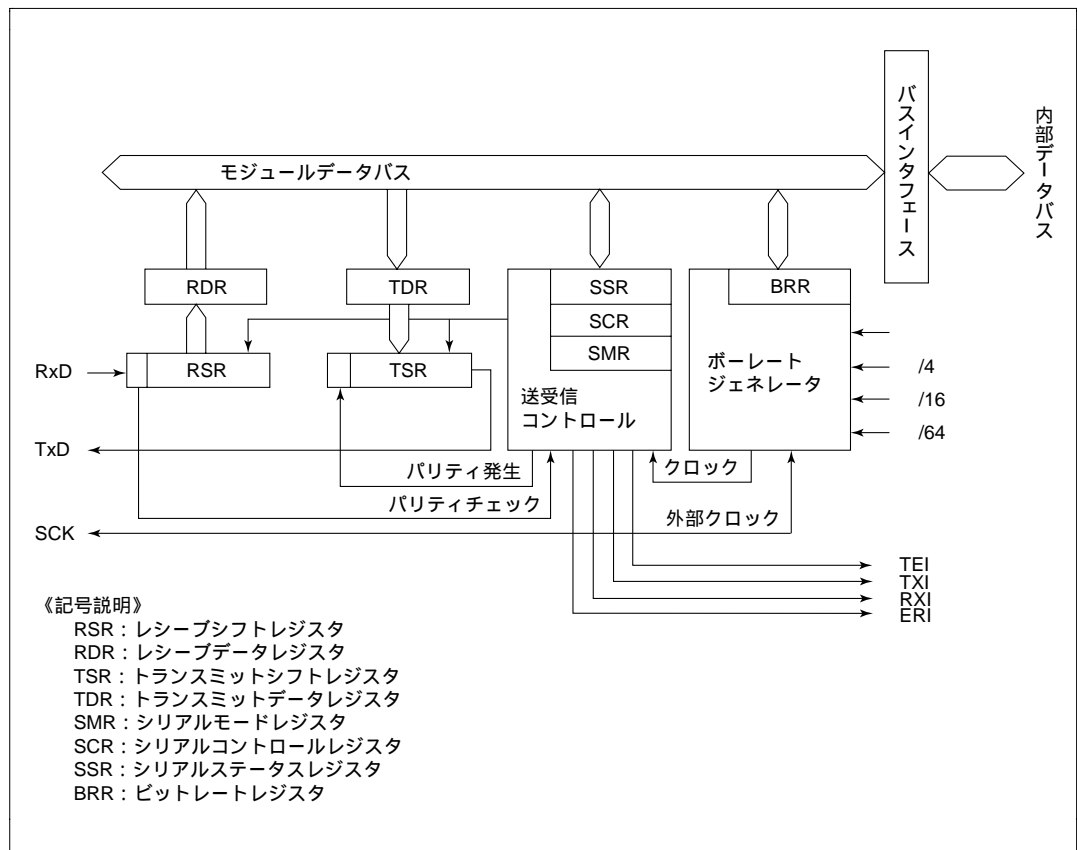


図 11.1 SCI のブロック図

11.1.3 端子構成

SCI は、表 11.1 に示すシリアル端子を持っています。

表 11.1 端子構成

名称	略称	入出力	機能
シリアルクロック端子	SCK	入出力	SCI のクロック入出力
レシーブデータ端子	RxD	入力	SCI の受信データ入力
トランスミットデータ端子	TxD	出力	SCI の送信データ出力

11.1.4 レジスタ構成

SCI には、表 11.2 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、および送信部/受信部の制御を行うことができます。

表 11.2 レジスタ構成

アドレス* ¹	名称	略称	R/W	初期値
H'FFB0	シリアルモードレジスタ	SMR	R/W	H'00
H'FFB1	ビットレートレジスタ	BRR	R/W	H'FF
H'FFB2	シリアルコントロールレジスタ	SCR	R/W	H'00
H'FFB3	トランスミットデータレジスタ	TDR	R/W	H'FF
H'FFB4	シリアルステータスレジスタ	SSR	R/(W)* ²	H'84
H'FFB5	レシーブデータレジスタ	RDR	R	H'00

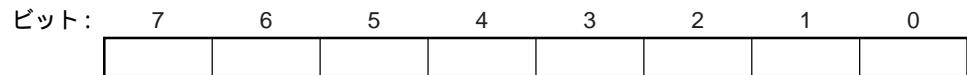
【注】 *1 アドレスの下位 16 ビットを示しています。

*2 フラグをクリアするための"0"ライトのみ可能です。

11.2 各レジスタの説明

11.2.1 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。



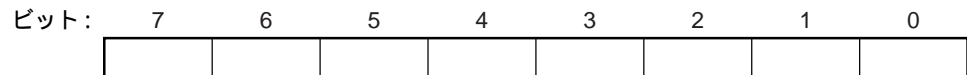
R/W : — — — — — — —

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。

11.2.2 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。



初期値: 0 0 0 0 0 0 0 0

R/W : R R R R R R R R

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。

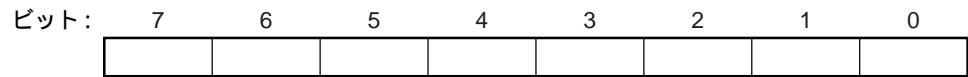
このように、RSR と RDR はダブルバッファになっているため連続した受信動作が可能です。

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。

11.2.3 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。



R/W : — — — — — — — —

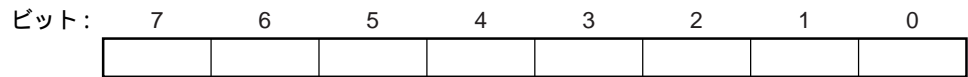
SCI は、TDR から送信データをいったん TSR に転送し、LSB (ビット0) から順に TxD 端子に送り出すことでシリアルデータ送信を行います。

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。

11.2.4 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。



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

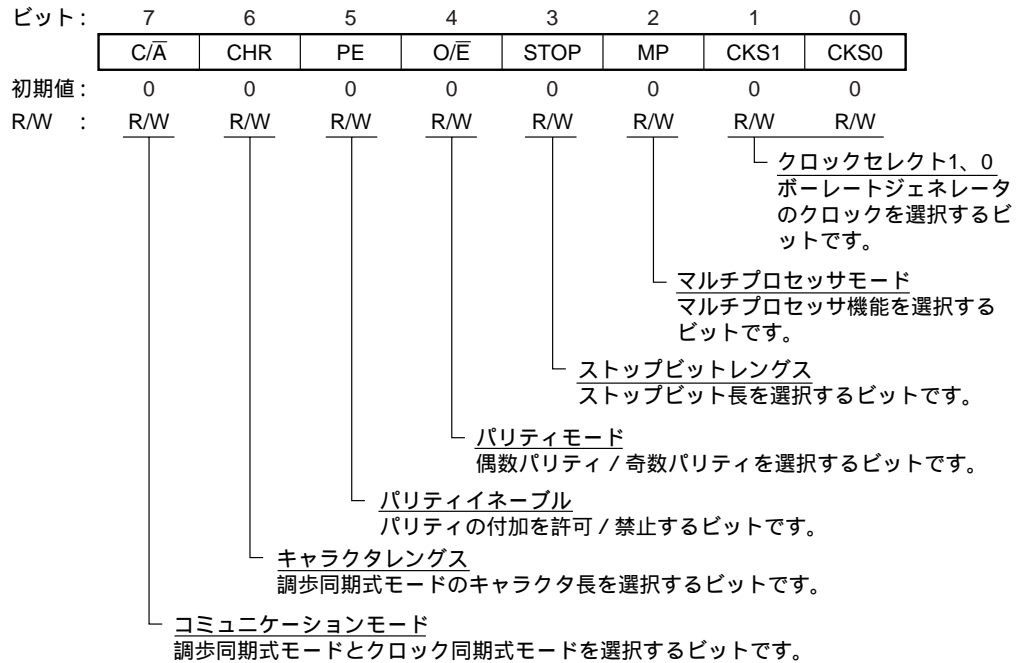
SCI は、TSR のあきを検出すると、TDR にライトされた送信データを TSR に転送してシリアル送信を開始します。TSR のシリアルデータ送信中に TDR に次の送信データをライトしておくと、連続シリアル送信ができます。

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 にイニシャライズされます。

11.2.5 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})

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/ \bar{A}	
0	調歩同期式モード (初期値)
1	クロック同期式モード

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		
CHR	説明	
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		
PE	説明	
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/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/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/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/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/E ビットの指定は無効です。

ビット4		
O/E	説明	
0	偶数パリティ* ¹	(初期値)
1	奇数パリティ* ²	

- 【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		
STOP	説明	
0	1ストップビット* ¹	(初期値)
1	2ストップビット* ²	

- 【注】 *1 送信時には、送信キャラクタの最後尾に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後尾に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、「11.3.3 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		
MP	説明	
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）

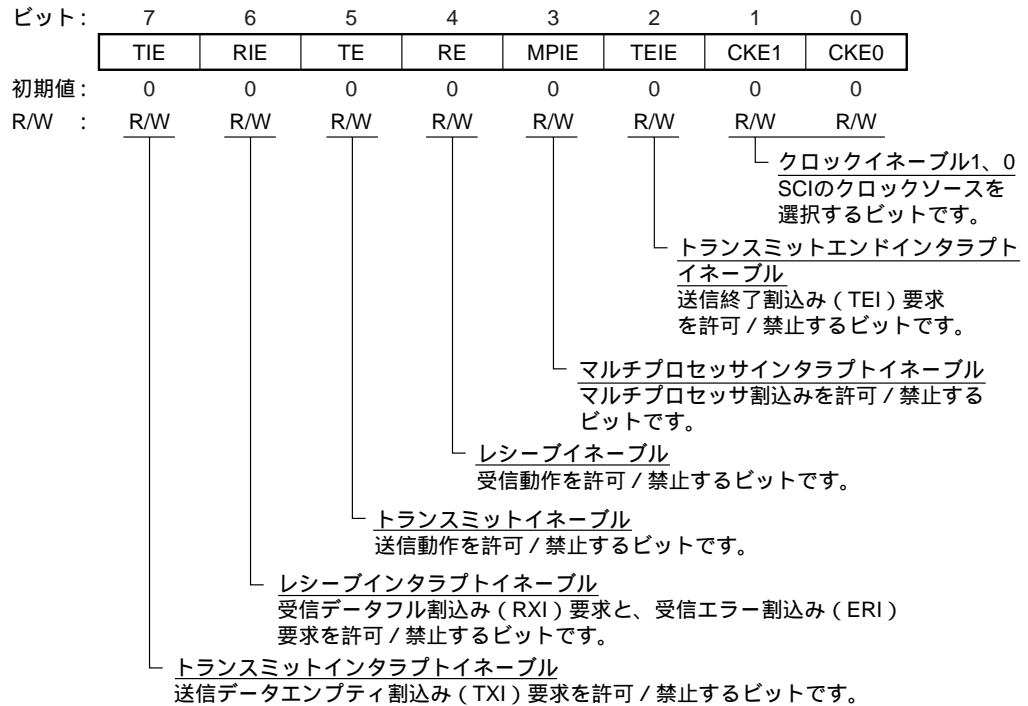
内蔵ポーレートジェネレータのクロックソースを選択します。CKS1、0 ビットの設定により、 $/4$ 、 $/16$ 、 $/64$ の4種類からクロックソースを選択できます。

クロックソースと、ビットレートレジスタの設定値、およびポーレートの関係については、「11.2.8 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		
CKS1	CKS0	説明	
0	0	クロック	(初期値)
	1	$/4$ クロック	
1	0	$/16$ クロック	
	1	$/64$ クロック	

11.2.6 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 へ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pty割り込み (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pty割り込み (TXI) 要求の禁止* (初期値)
1	送信データエンpty割り込み (TXI) 要求の許可

【注】 * TXI 割り込み要求の解除は、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	
RIE	説明
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	
TE	説明
0	送信動作を禁止* ¹ (初期値)
1	送信動作を許可* ²

【注】 *¹ SSR の TDRE フラグは"1"に固定されます。

*² 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		
RE	説明	
0	受信動作を禁止* ¹	(初期値)
1	受信動作を許可* ²	

【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にのみ有効です。

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		
MPIE	説明	
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) 〔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	(初期値)
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 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)

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") 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 (CKE = "1") の場合はCKE0ビットの設定は無効です。また、CKE1、CKE0ビットの設定を行ったのち、SMRでSCIの動作モード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「11.3 動作説明」の表11.9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*1
		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
	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*2
		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*3
		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*3
		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

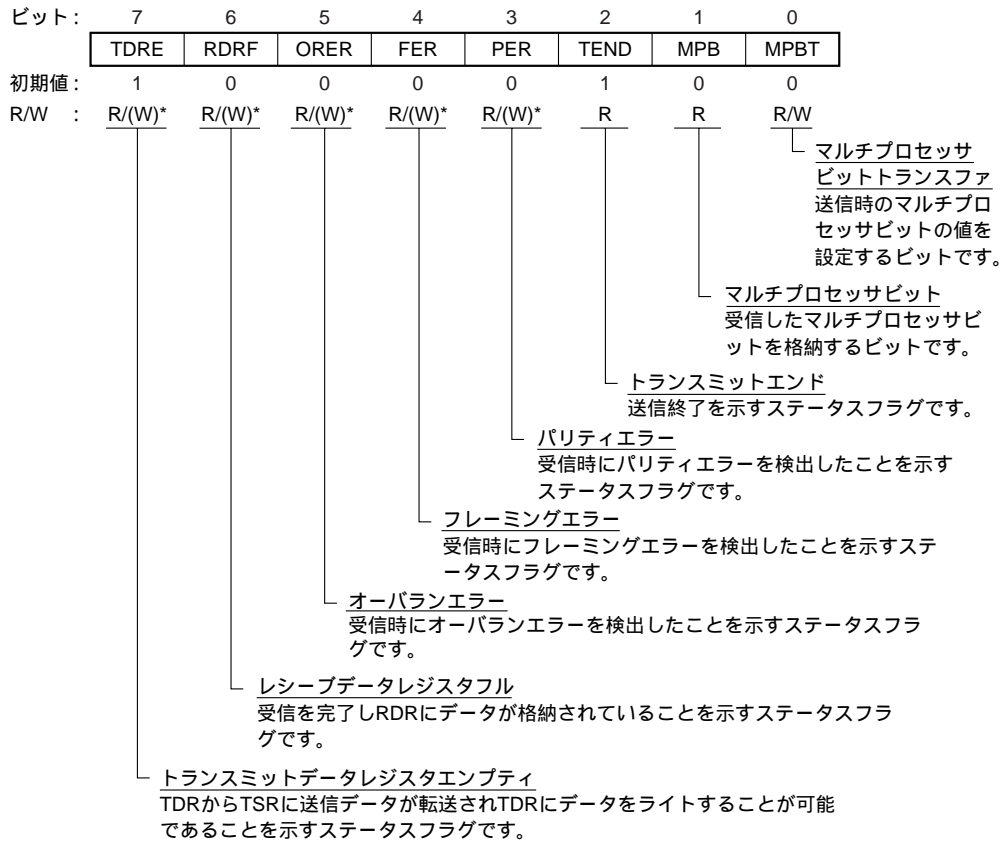
【注】 *1 初期値

*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

11.2.7 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】 * フラグをクリアするための"0"ライトのみ可能です。

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	
TDRE	説明
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〕 TDRE = "1"の状態をリードした後、"0"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 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	
RDRF	説明
0	RDR に受信データが格納されていないことを表示 (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) RDRF = "1"の状態をリードした後、"0"をライトしたとき
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	
ORER	説明
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) ^{*1} 〔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	受信時にオーバランエラーが発生したことを表示 ^{*2} 〔セット条件〕 RDRF = "1"の状態での次のシリアル受信を完了したとき

【注】 *1 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。

*2 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、以降のシリアル受信を続けることはできません。なお、クロック同期式モードでは、シリアル送信を続けることもできません。

ビット4：フレーミングエラー（FER）

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	
FER	説明
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）*1 〔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後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*2

【注】 *1 SCRのRE ビットを"0"にクリアしたときには、FERフラグは影響を受けず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定し、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はできません。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。

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エラーが発生して異常終了したことを示します。

ビット3	
PER	説明
0	受信中、または正常に受信を完了したことを表示* ¹ (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* ² 〔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）

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	
TEND	説明
0	送信中であることを表示 〔クリア条件〕 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
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"であったとき

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	
MPB	説明
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、あるいは送信でないときにはMPBT ビットの設定は無効です。

ビット0	
MPBT	説明
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

11.2.8 ビットレイトレジスタ (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
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 11.3 に調歩同期式モードの BRR の設定例を表 11.4 にクロック同期式モードの BRR の設定例を示します。

表 11.3 ビットレートに対する BRR の設定例〔調歩同期式モード〕(1)

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			

表 11.3 ビットレートに対する BRR の設定例〔調歩同期式モード〕(2)

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

表 11.3 ビットレートに対する BRR の設定例〔調歩同期式モード〕(3)

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

表 11.3 ビットレートに対する BRR の設定例〔調歩同期式モード〕(4)

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

表 11.3 ビットレートに対する BRR の設定例〔調歩同期式モード〕(5)

ビットレート (bit/s)	(MHz)											
	14			14.7456			16			18		
	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)	n	N	誤差 (%)
110	2	248	- 0.17	3	64	0.70	3	70	0.03	3	79	- 0.12
150	2	181	0.16	2	191	0.00	2	207	0.16	2	233	0.16
300	2	90	0.16	2	95	0.00	2	103	0.16	2	116	0.16
600	1	181	0.16	1	191	0.00	1	207	0.16	1	233	0.16
1200	1	90	0.16	1	95	0.00	1	103	0.16	1	116	0.16
2400	0	181	0.16	0	191	0.00	0	207	0.16	0	233	0.16
4800	0	90	0.16	0	95	0.00	0	103	0.16	0	116	0.16
9600	0	45	- 0.93	0	47	0.00	0	51	0.16	0	58	- 0.69
19200	0	22	- 0.93	0	23	0.00	0	25	0.16	0	28	1.02
31250	0	11	0.00	0	14	- 1.70	0	15	0.00	0	17	0.00
38400	0	10	3.57	0	11	0.00	0	12	0.16	0	14	- 2.34

表 11.4 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)	OSC (MHz)											
	2		4		8		10		16		18	
	n	N	n	N	n	N	n	N	n	N	n	N
110	3	70										
250	2	124	2	249	3	124			3	249		
500	1	249	2	124	2	249			3	124	3	140
1k	1	124	1	249	2	124			2	249	3	69
2.5k	0	199	1	99	1	199	1	249	2	99	2	112
5k	0	99	0	199	1	99	1	124	1	199	1	224
10k	0	49	0	99	0	199	0	249	1	99	1	112
25k	0	19	0	39	0	79	0	99	0	159	0	179
50k	0	9	0	19	0	39	0	49	0	79	0	89
100k	0	4	0	9	0	19	0	24	0	39	0	44
250k	0	1	0	3	0	7	0	9	0	15	0	17
500k	0	0*	0	1	0	3	0	4	0	7	0	8
1M			0	0*	0	1			0	3	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〕

$$N = \frac{1}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

〔クロック同期式モード〕

$$N = \frac{1}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

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、1、2、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。

$$\text{誤差 (\%)} = \left\{ \frac{\times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

表 11.5 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 11.6 ~ 表 11.7 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 11.5 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

表 11.6 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

表 11.7 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

11.3 動作説明

11.3.1 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。

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 11.8 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まります。これを表 11.9 に示します。

(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레이크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

表 11.8 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	8ビット データ	なし	なし	1ビット	
				1	モード				2ビット	
			1	0	0				あり	1ビット
					1					2ビット
			0	0	0				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	1	0	調歩同期式	8ビット データ	あり	なし	1ビット	
					1				モード (マルチ プロセッサ フォーマット)	2ビット
				1	0				0	7ビット データ
		1	2ビット							
1		0	1	0	クロック	8ビット データ	なし	なし	なし	
					1				同期式モード	

表 11.9 SMR、SCR の設定と SCI のクロックソースの選択

SMR	SCR の設定値		モード	SCI の送信 / 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	内部	SCI は、SCK 端子を使用しません	
		1	モード		ビットレートと同じ周波数のクロックを出力	
	1	0	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	内部	同期クロックを出力	
		1	同期式			
	1	0	0	モード	外部	同期クロックを入力
			1			

11.3.2 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 11.2 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。

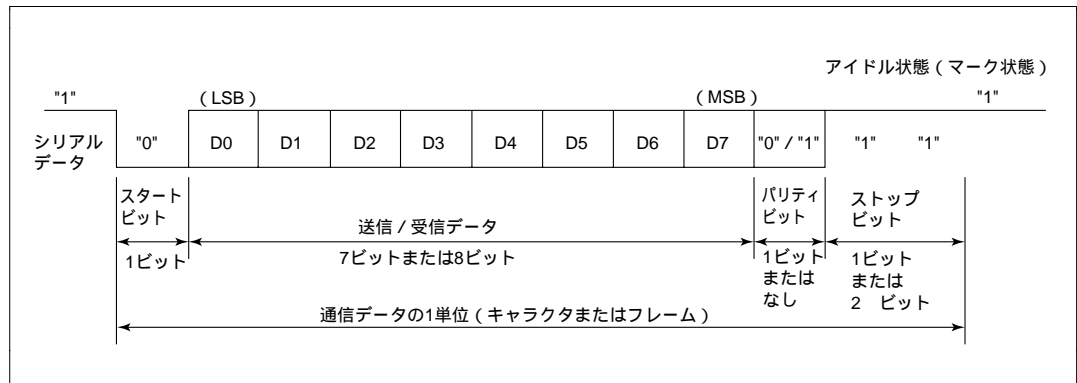


図 11.2 調歩同期式通信のデータフォーマット
(8 ビットデータ / パリティあり / 2 ストップビットの例)

(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 11.10 に示します。送信 / 受信フォーマットは 12 種類あり、SMR の選定により選択できます。

表 11.10 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 のクロックソースの選択については表 11.9 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 11.3 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。

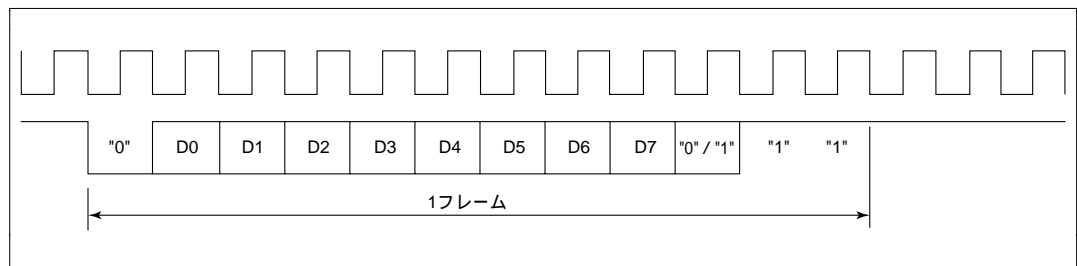


図 11.3 出力クロックと通信データの位相関係 (調歩同期式モード)

(3) データの送信 / 受信動作

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 11.4 に SCI のイニシャライズフローチャートの例を示します。

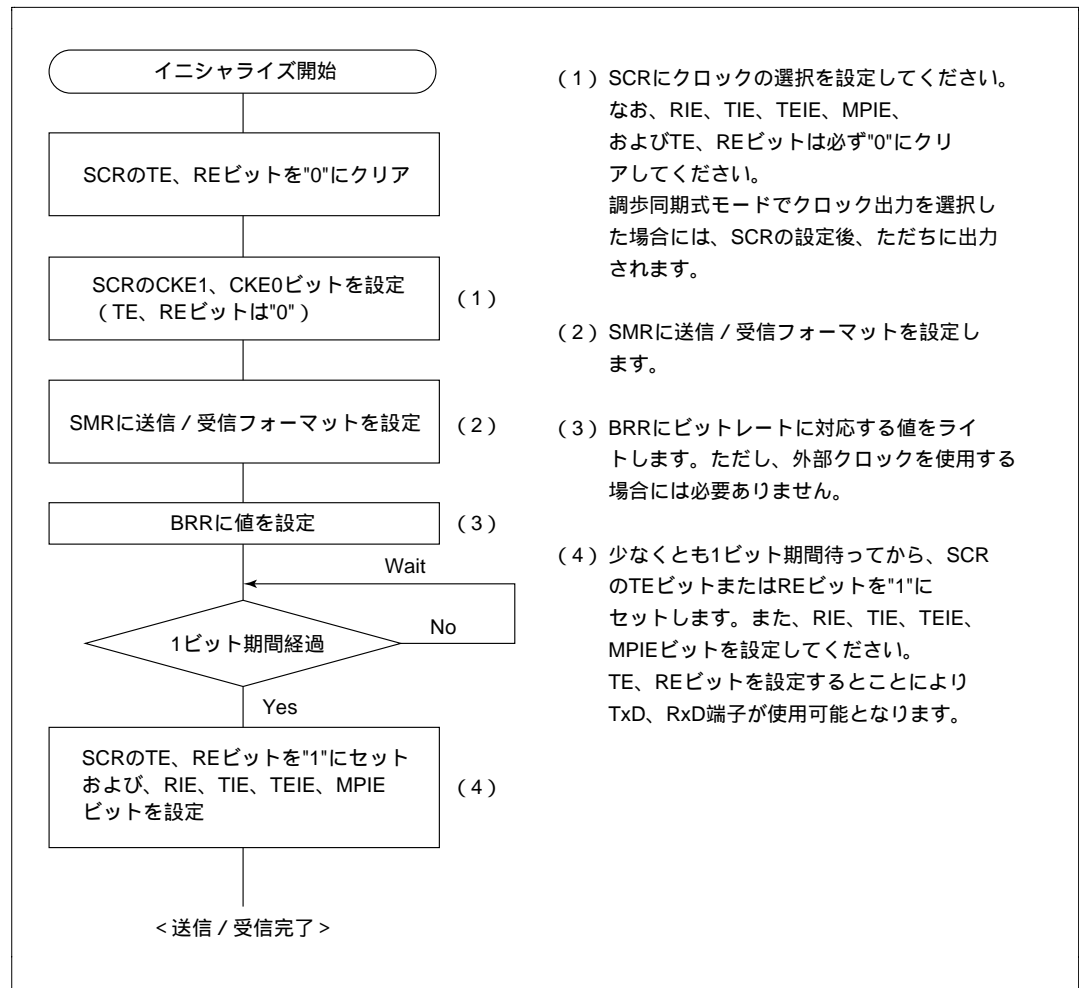


図 11.4 SCI のインシャライズフローチャートの例

シリアルデータ送信（調歩同期式）

図 11.5 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。

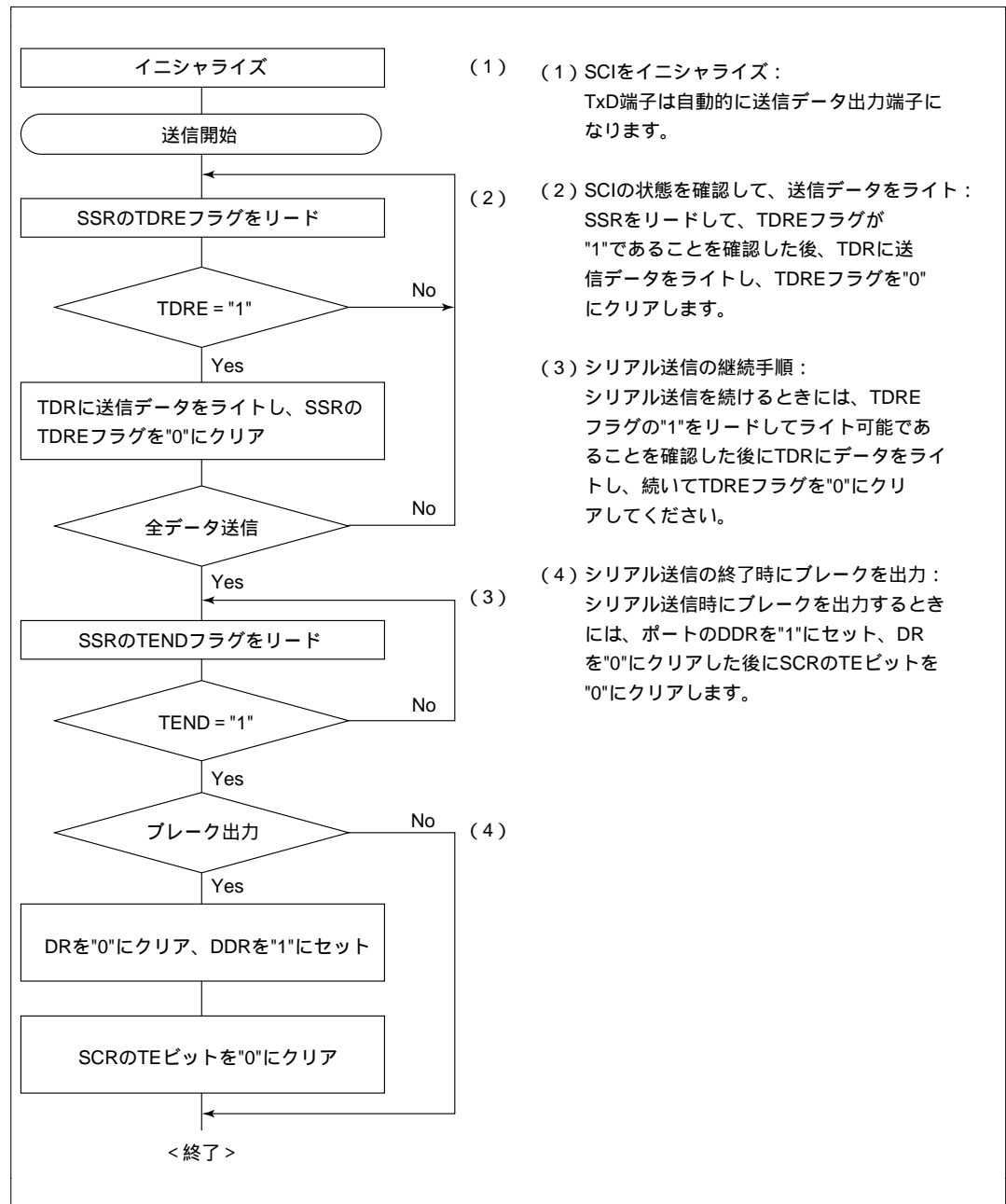


図 11.5 シリアル送信のフローチャートの例

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ことを認識し、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プティ割込み (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 11.6 に示します。

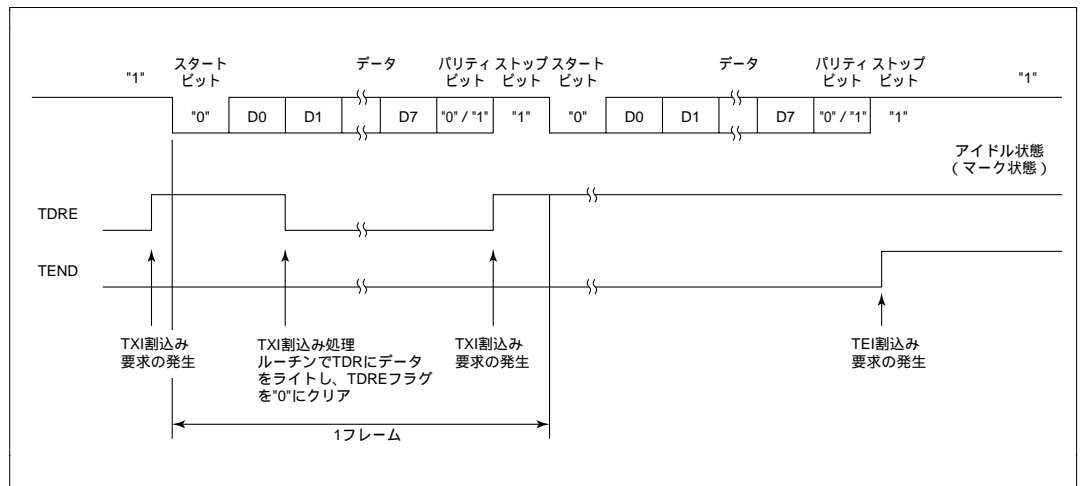


図 11.6 調歩同期式モードでの送信時の動作例

(8 ビットデータ / パリティあり / 1 ストップビットの例)

シリアルデータ受信（調歩同期式）

図 11.7 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。

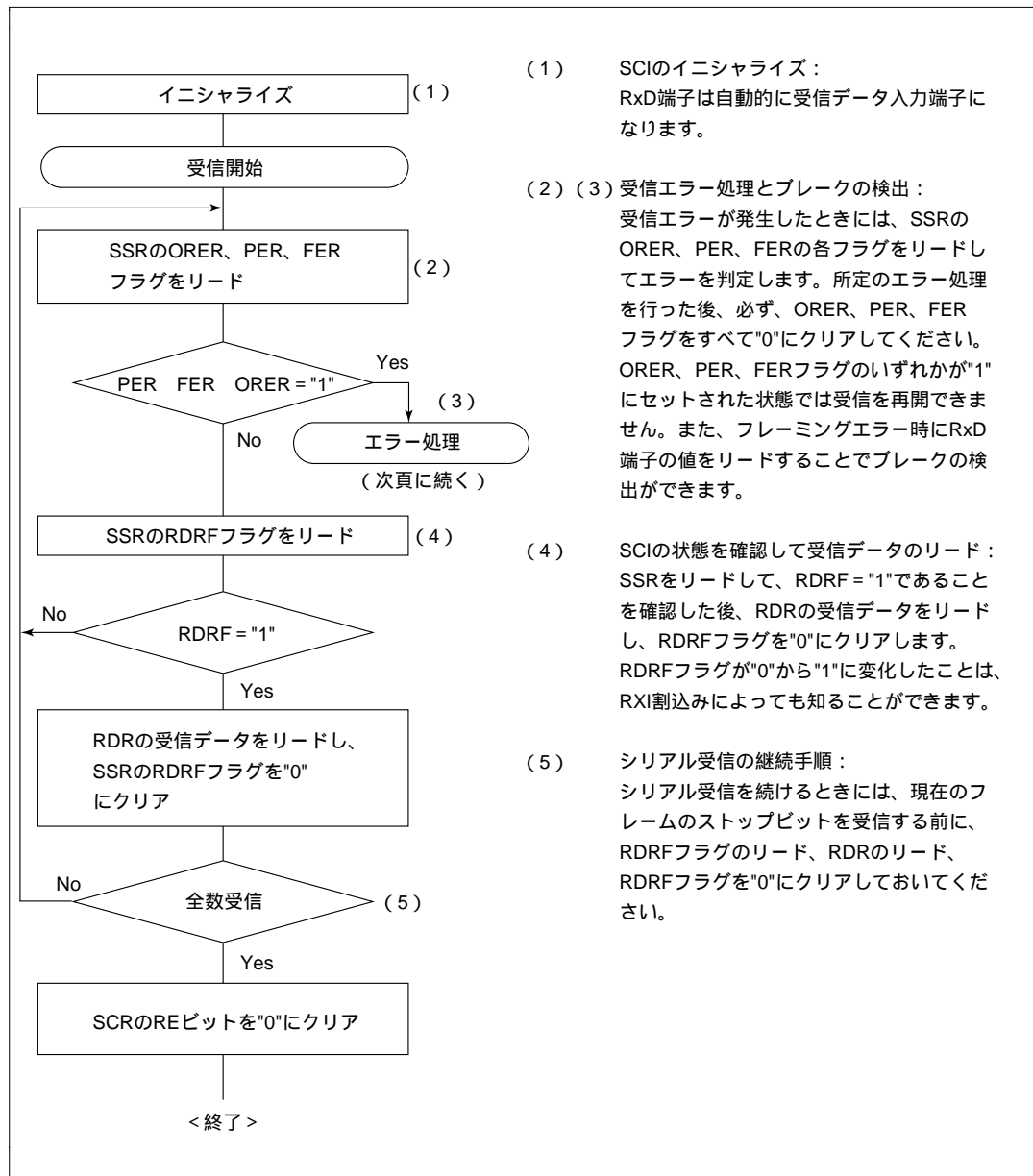


図 11.7 シリアル受信データフローチャートの例（1）

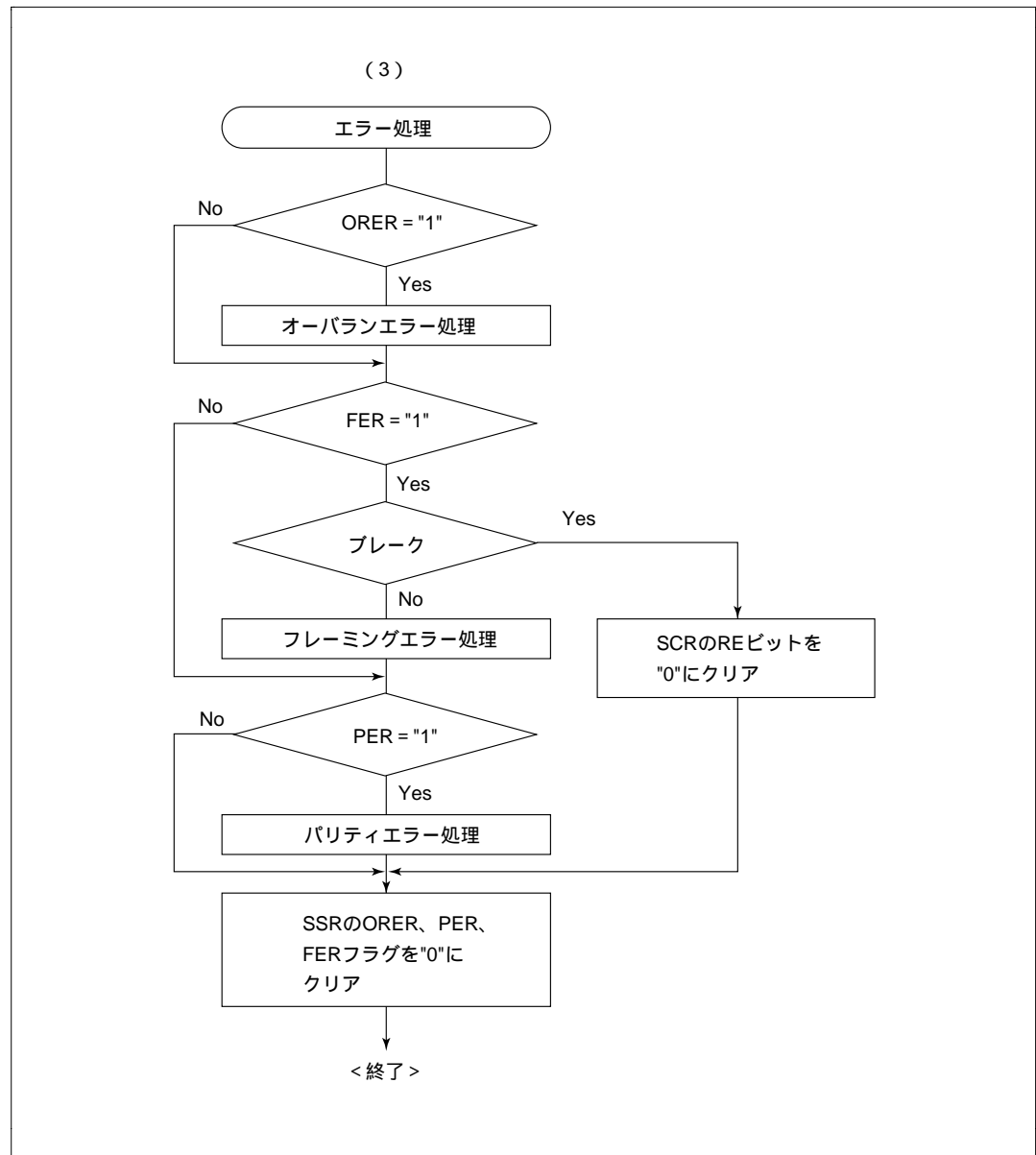


図 11.7 シリアル受信データフローチャートの例 (2)

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*が発生すると表11.11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)要求が発生します。

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)要求が発生します。

表 11.11 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に受信データが転送されます。
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 11.8 に示します。

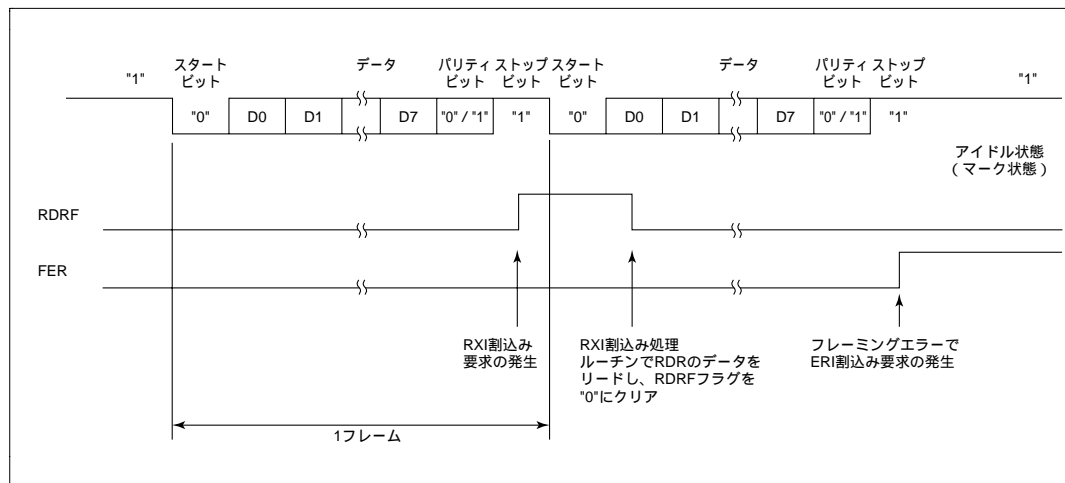


図 11.8 SCI の受信時の動作例 (8 ビットデータ / パリティあり / 1 ストップビットの例)

11.3.3 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 11.9 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 11.11 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。

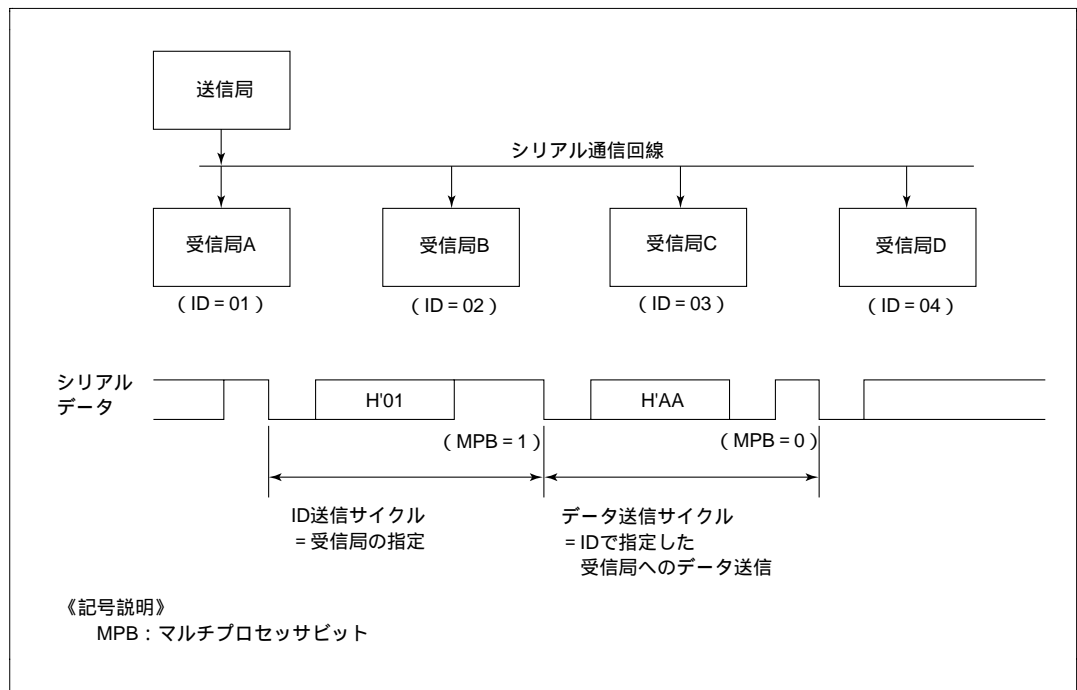


図 11.9 マルチプロセッサフォーマットを使用したプロセッサ間通信の例
(受信局 A へのデータ H'AA の送信の例)

(3) データの送信 / 受信動作

マルチプロセッサシリアルデータ送信

図 11.10 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。

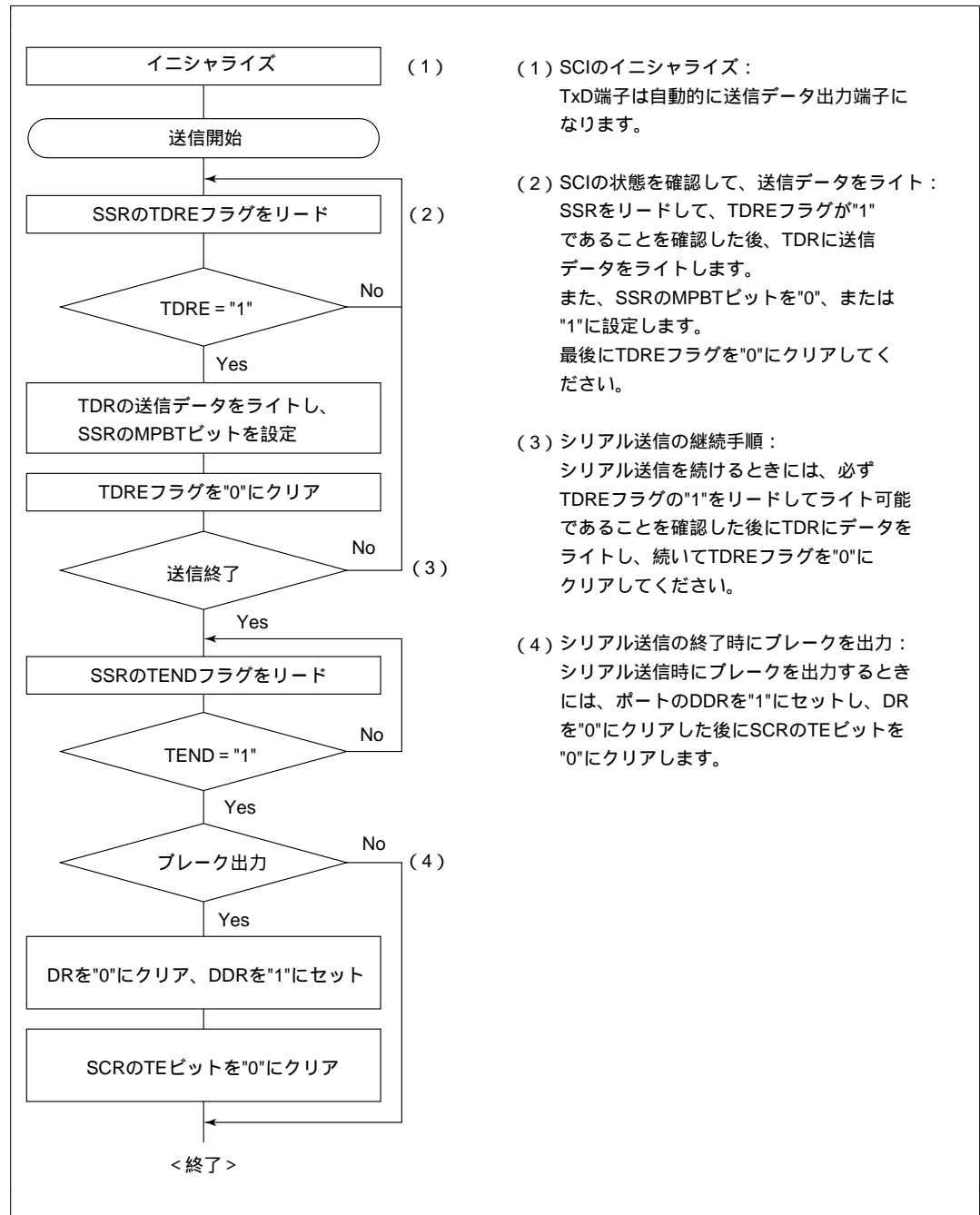


図 11.10 マルチプロセッサシリアル送信のフローチャートの例

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プティ割込み (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 11.11 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。

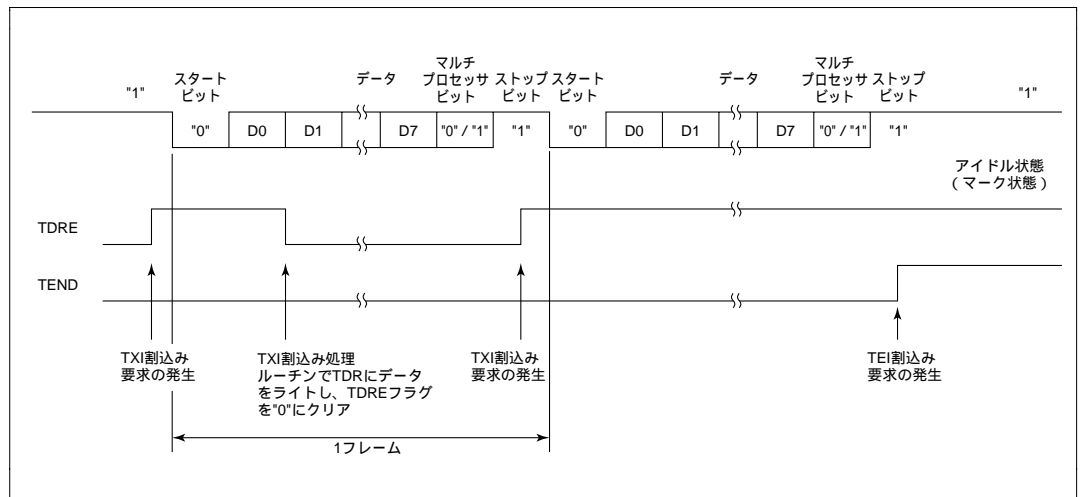


図 11.11 SCI の送信時の動作例
 (8 ビットデータ / マルチプロセッサビットあり / 1 ストップビットの例)

マルチプロセッサシリアルデータ受信

図 11.12 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。

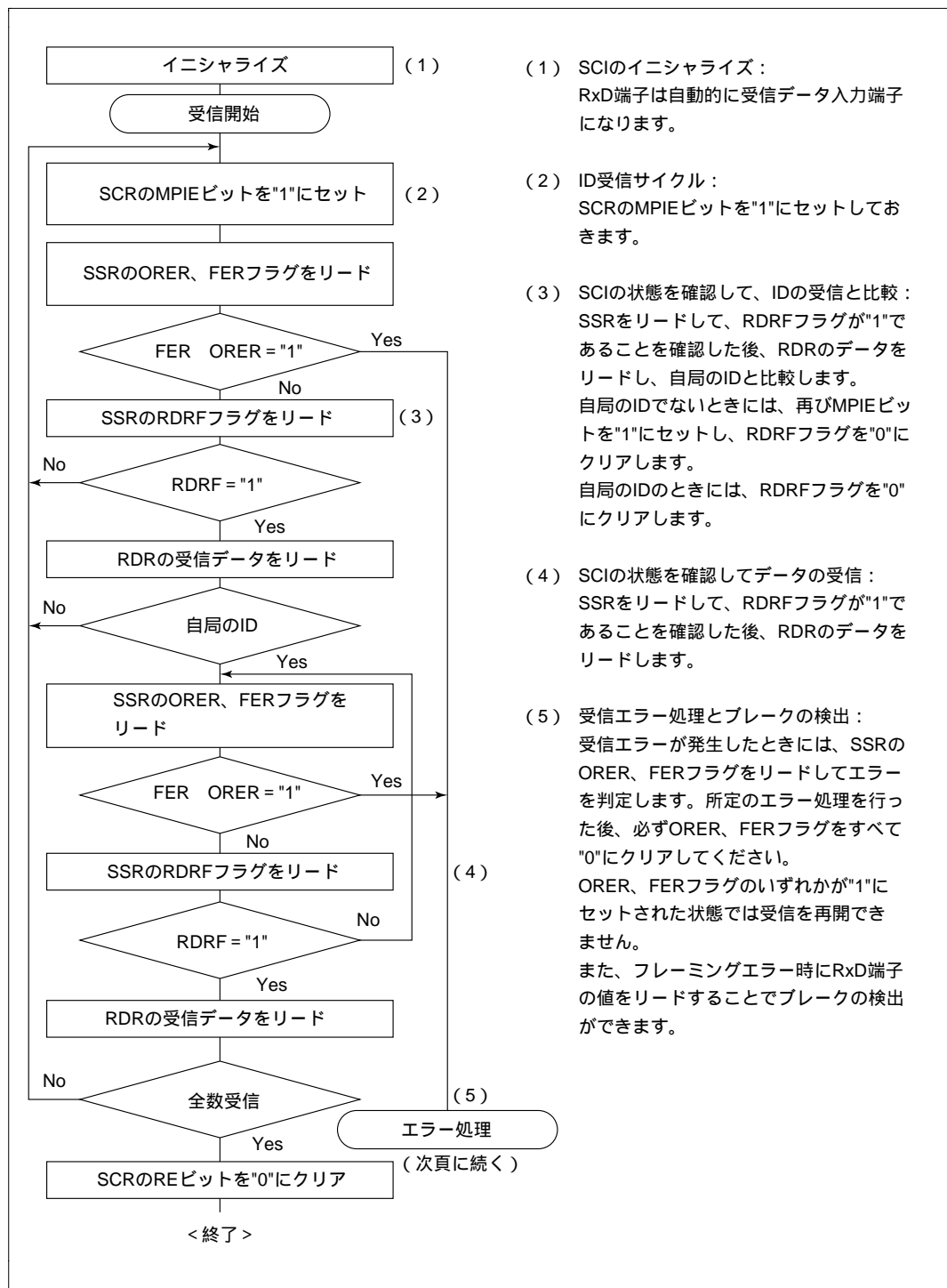


図 11.12 マルチプロセッサシリアル受信のフローチャートの例 (1)

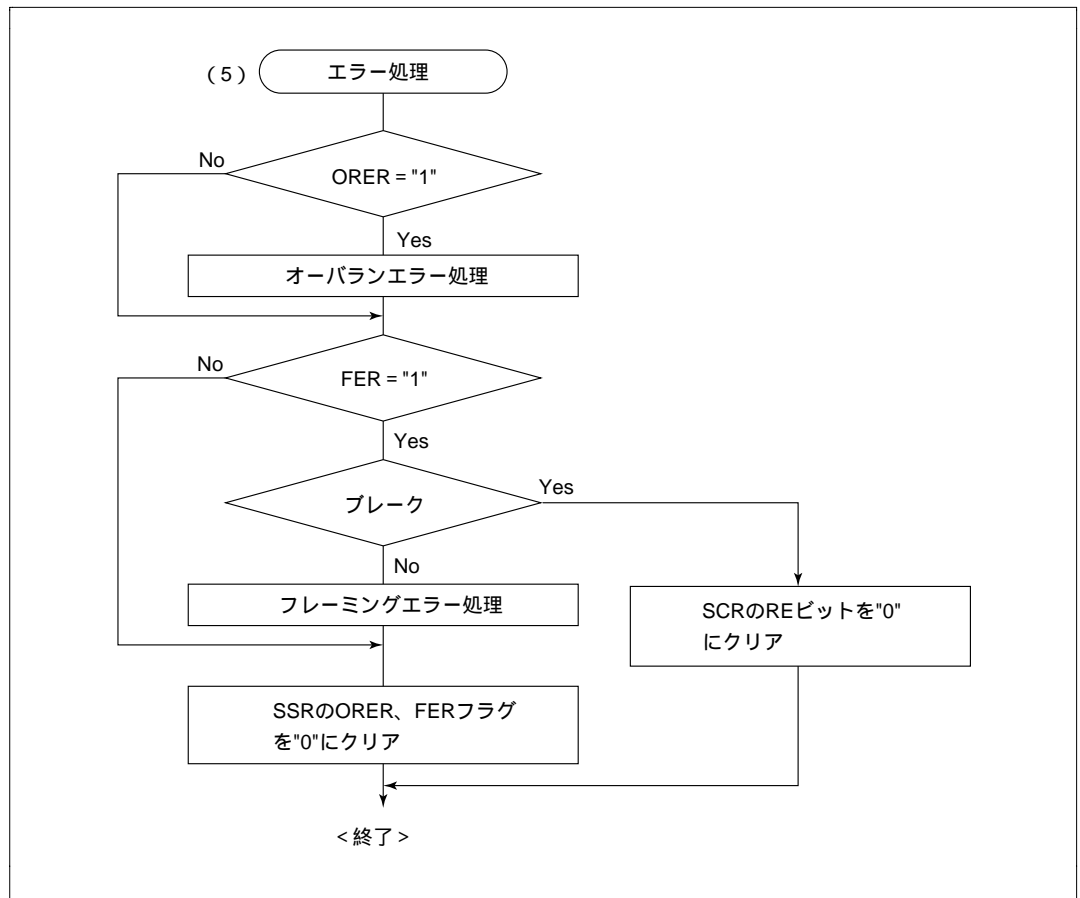


図 11.12 マルチプロセッサシリアル受信のフローチャートの例 (2)

図 11.13 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。

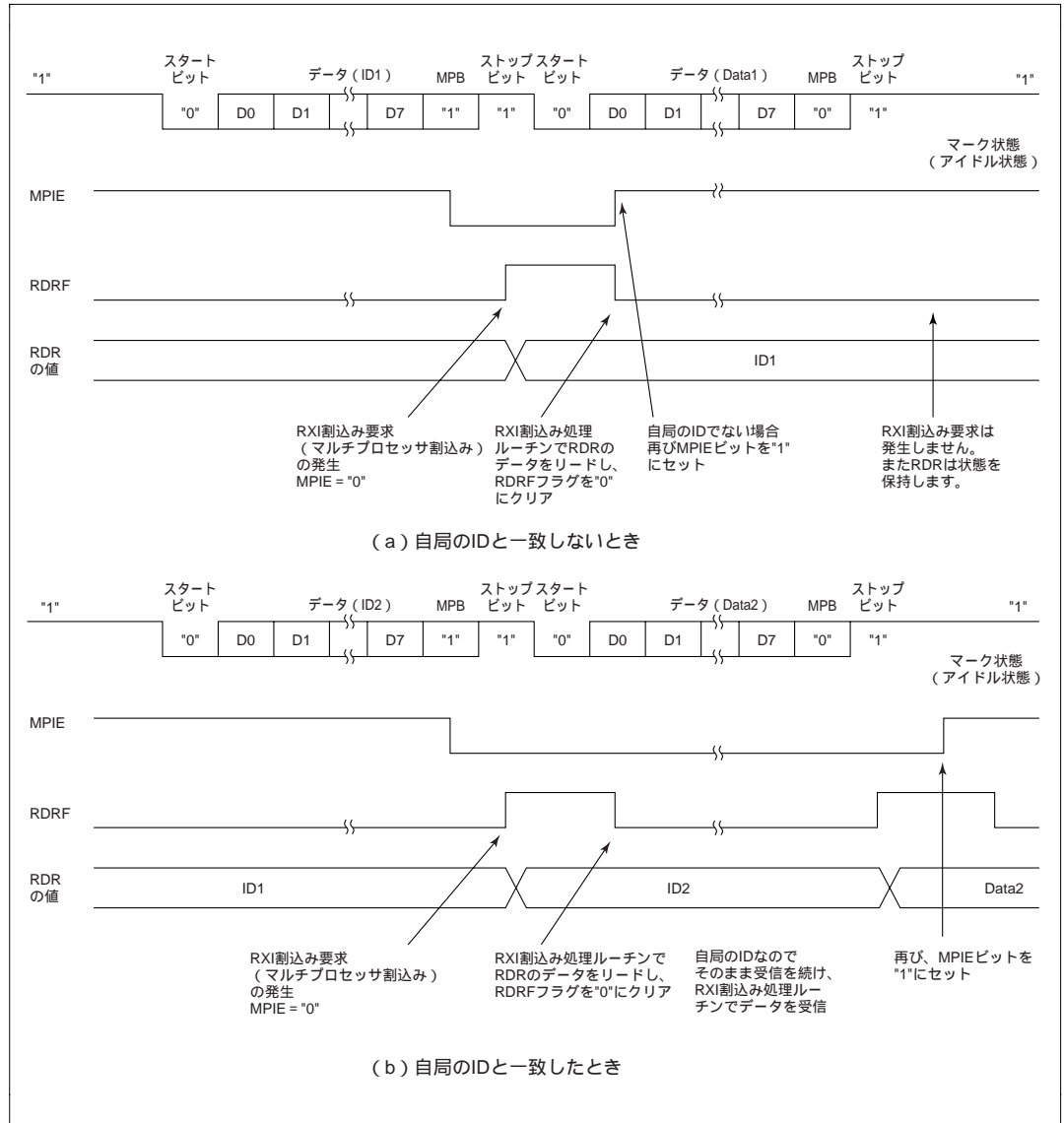


図 11.13 SCI の受信時の動作例

(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)

11.3.4 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 11.14 に示します。

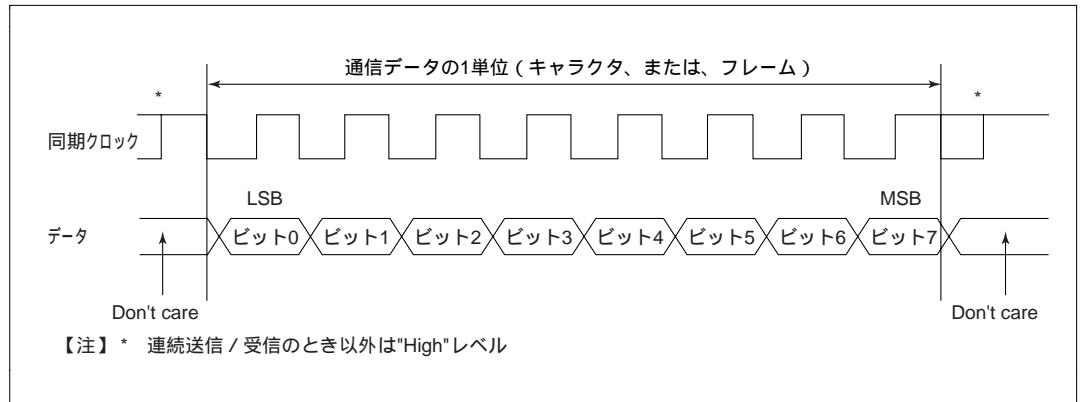


図 11.14 クロック同期式通信のデータフォーマット

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で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 11.9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。

(3) データの送信 / 受信動作

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、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、FER、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 11.15 に SCI のイニシャライズフローチャートの例を示します。

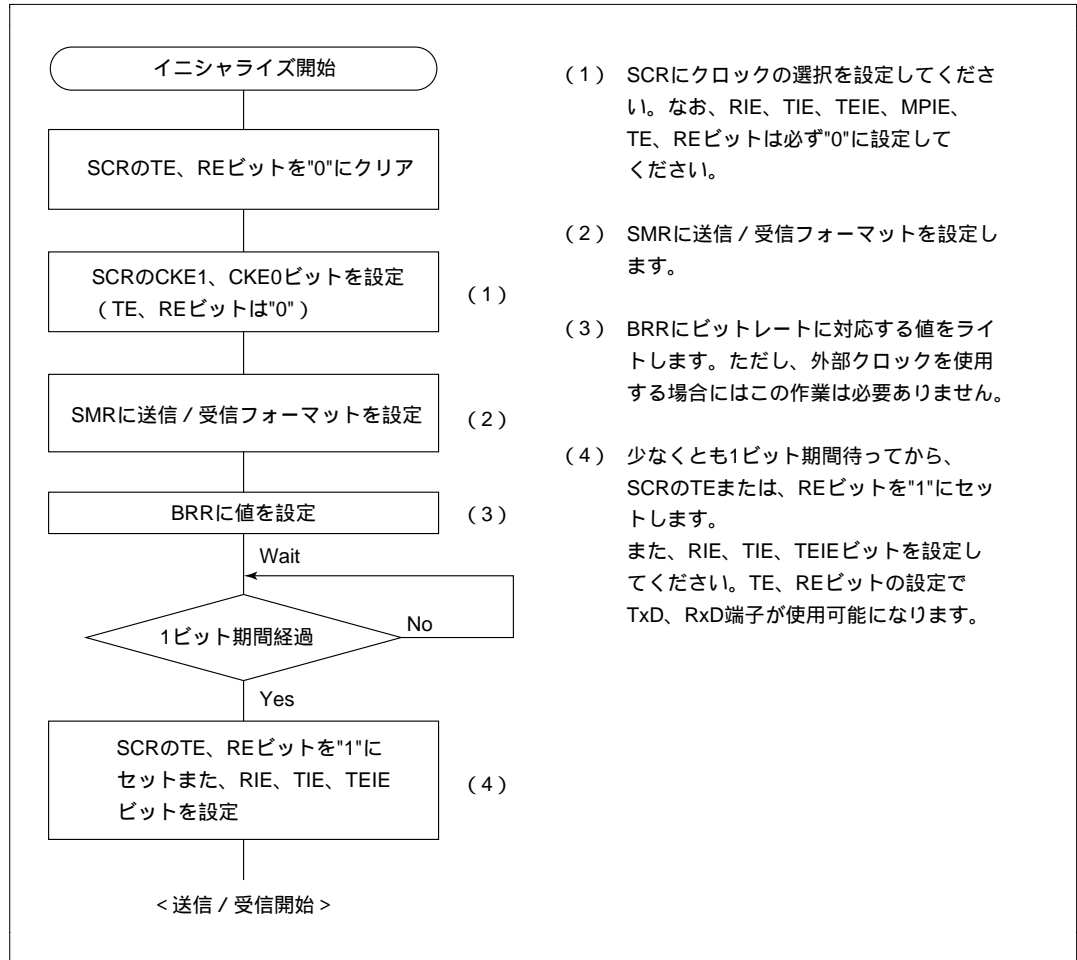


図 11.15 SCI のイニシャライズフローチャートの例

シリアルデータ送信（クロック同期式）

図 11.16 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。

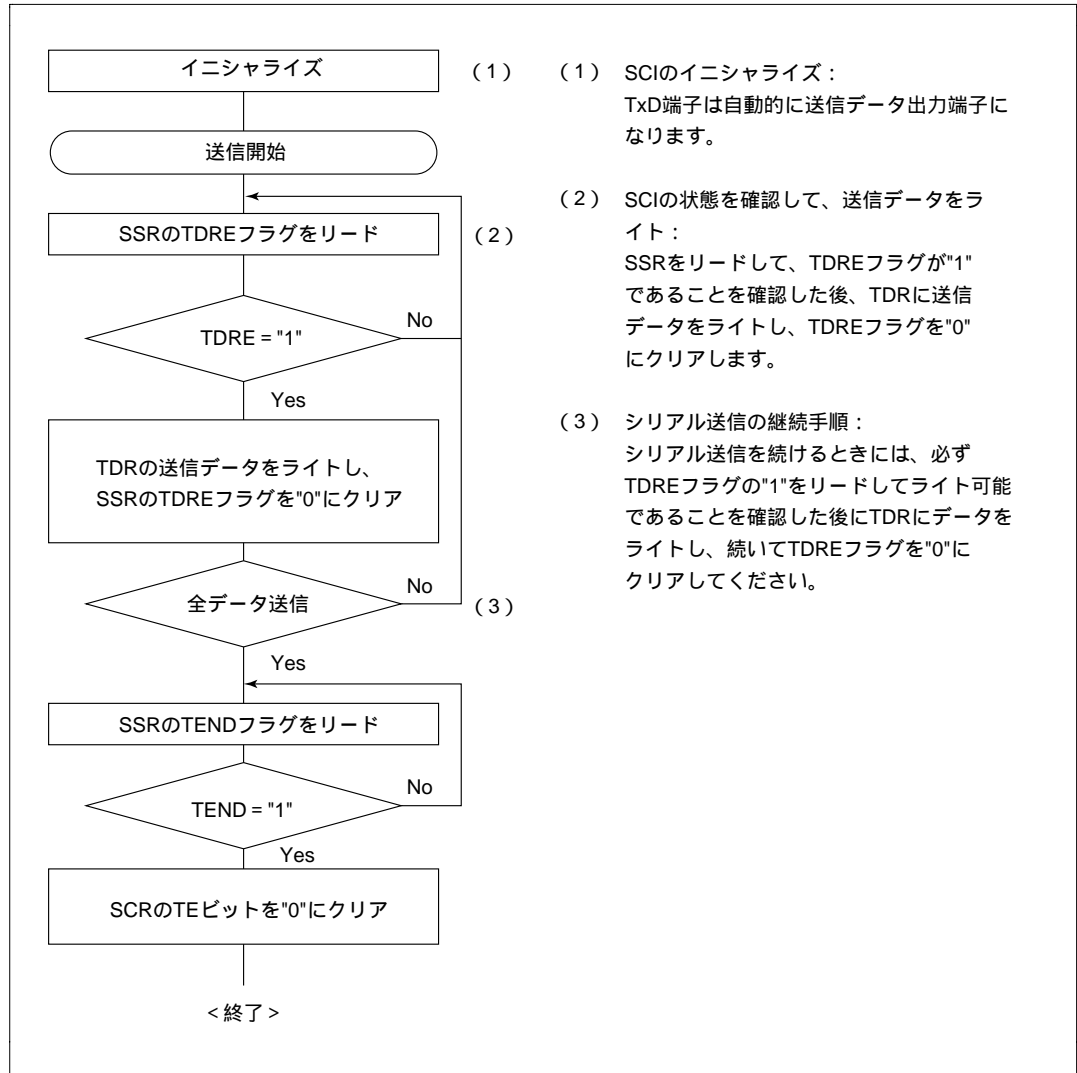


図 11.16 シリアル送信のフローチャートの例

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 にデータを転送します。
- (2) TDR から TSR ヘ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) 要求が発生します。

(4) シリアル送信終了後は、SCK 端子は固定になります。

図 11.17 に SCI の送信時の動作例を示します。

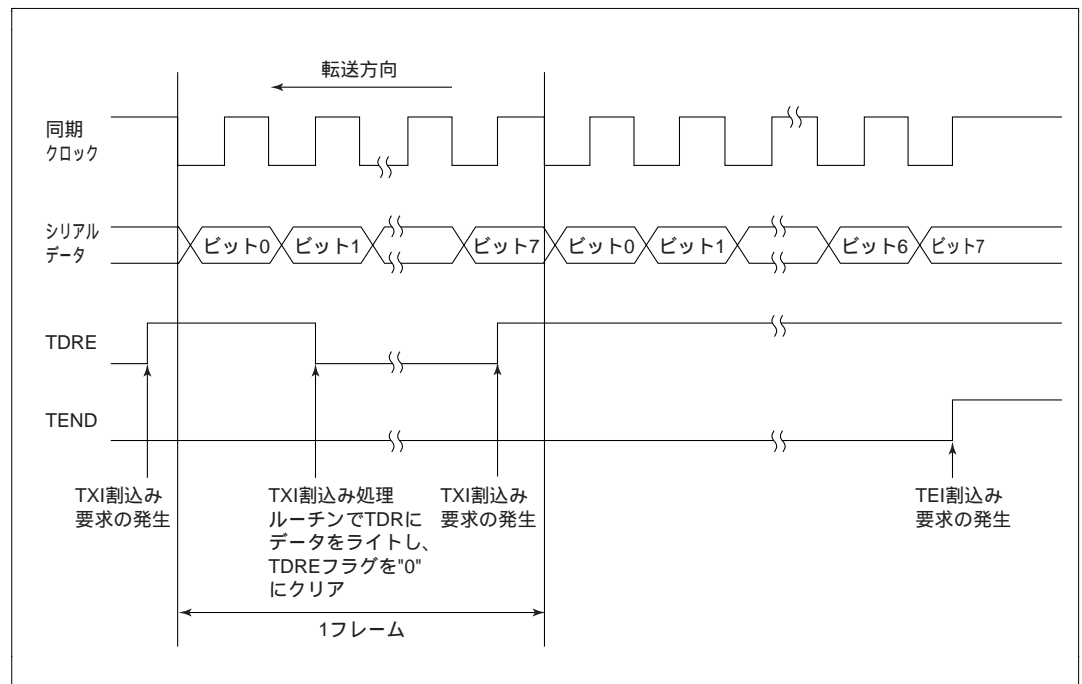


図 11.17 SCI の送信時の動作例

シリアルデータ受信（クロック同期式）

図 11.18 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。

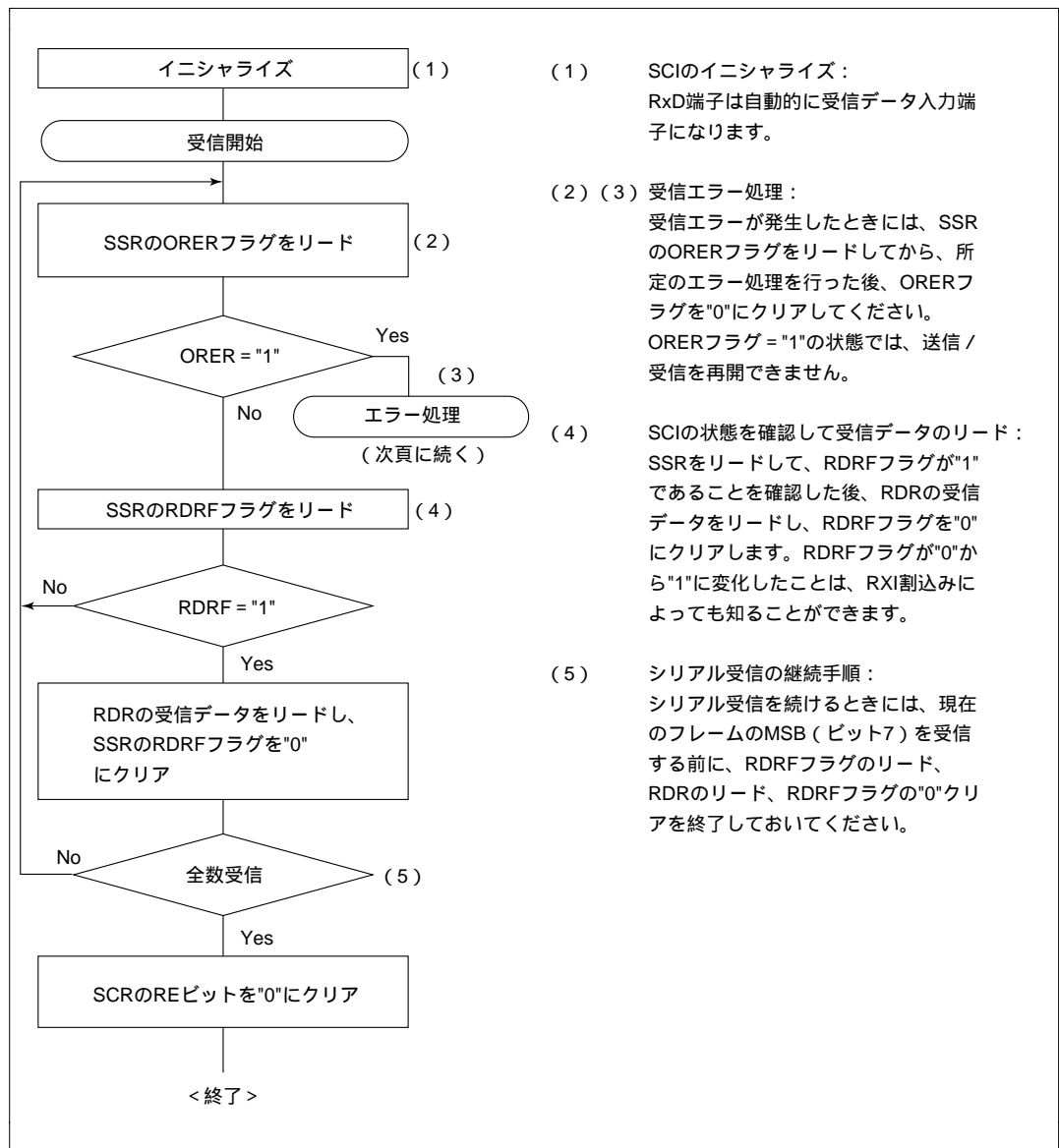


図 11.18 シリアルデータ受信フローチャートの例（1）

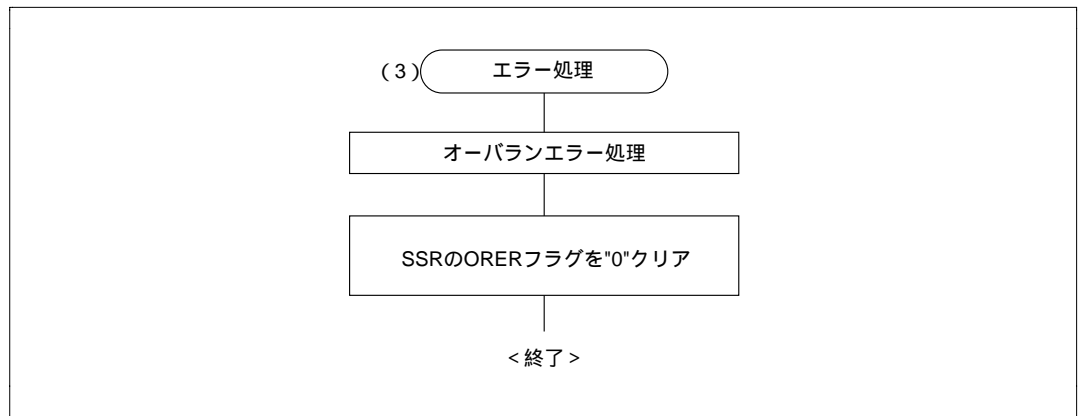


図 11.18 シリアルデータ受信フローチャートの例 (2)

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を発生すると、表 11.11 のように動作します。エラーチェックで受信エラーを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) 要求を発生します。

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) 要求を発生します。

図 11.19 に SCI の受信時の動作例を示します。

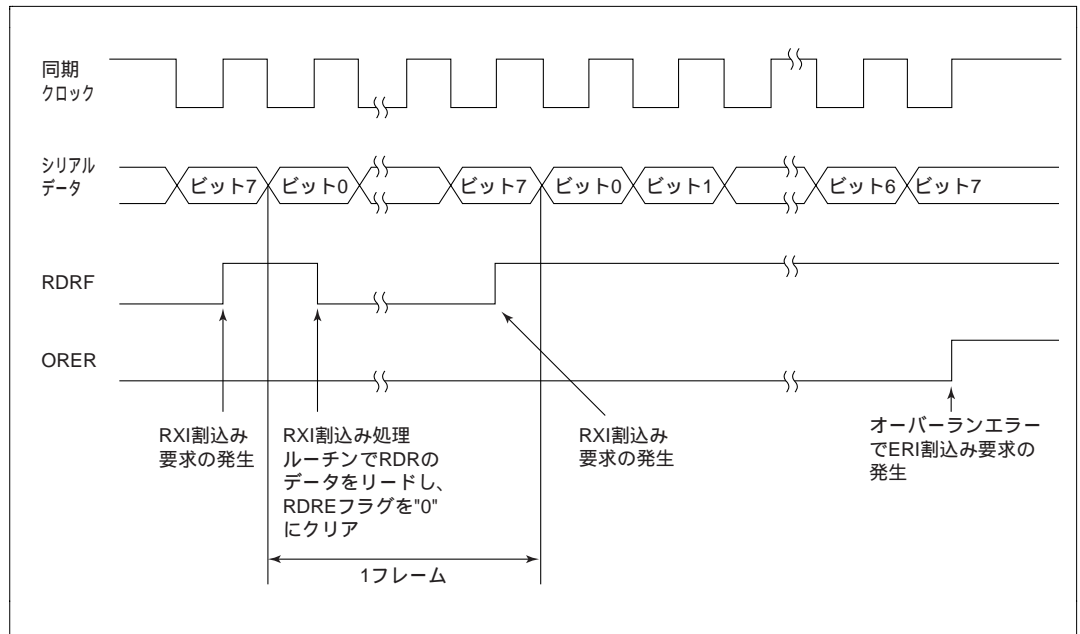


図 11.19 SCI の受信時の動作例

シリアルデータ送受信同時動作（クロック同期式）

図 11.20 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。

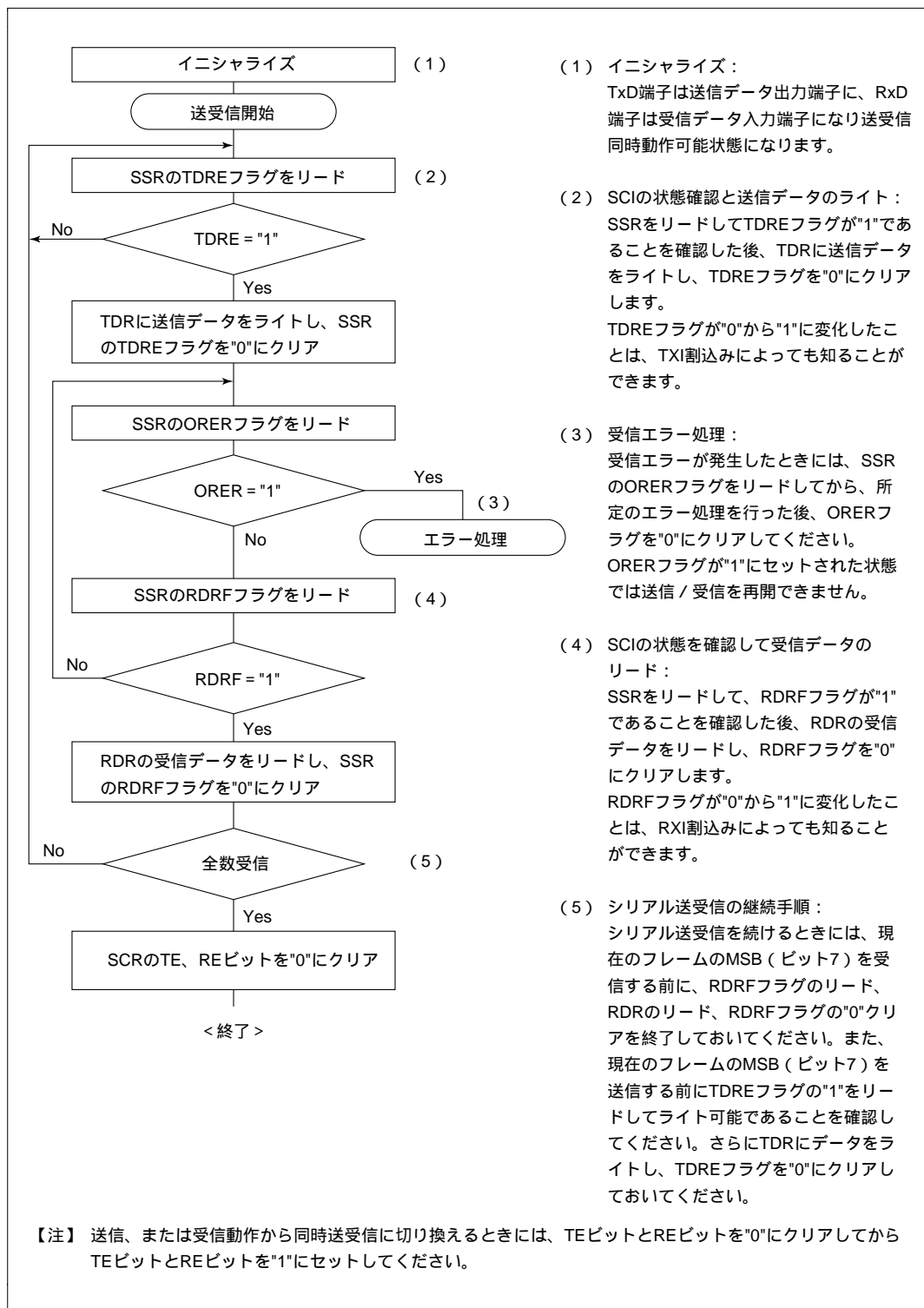


図 11.20 シリアル送受信同時動作のフローチャートの例

11.4 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 11.12 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。

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。

表 11.12 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	

11.5 使用上の注意

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 11.13 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 11.13 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、TE 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 8 クロック目の立上がりエッジで内部に取り込みます。これを図 11.21 に示します。

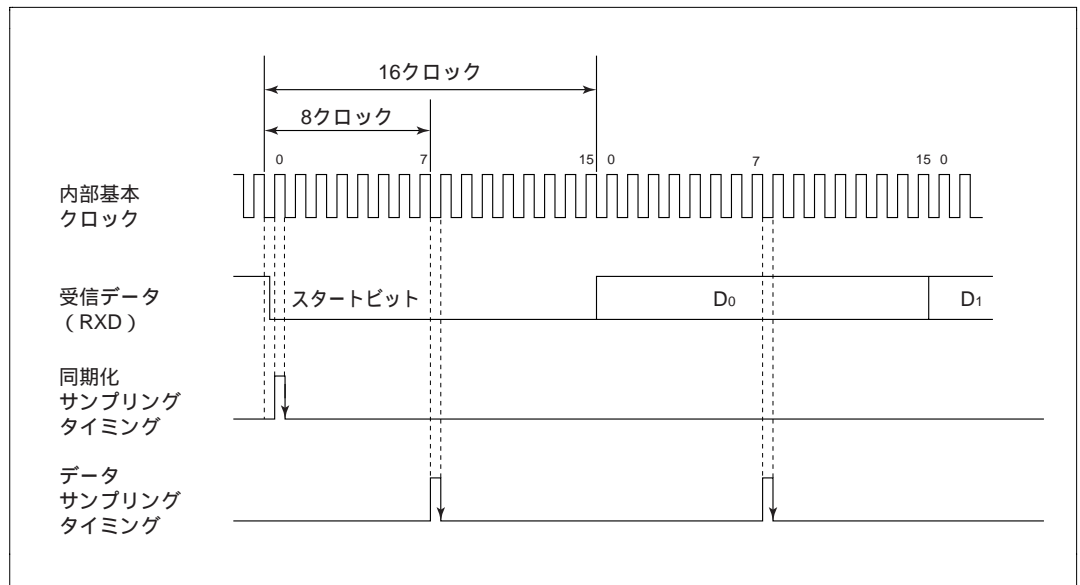


図 11.21 調歩同期式モードの受信データサンプリングタイミング

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。

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\% \quad \dots \text{式(1)}$$

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

$$M = \left(0.5 - \frac{1}{2 \times 16} \right) \times 100\% \\ = 46.875\% \quad \dots \text{式(2)}$$

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) クロック同期式使用上の制約事項

同期クロックに外部クロックソースを使用しデータ送信を行う場合、SSR の TDRE のクリアから各フレームに対応する送信クロックの先頭 (立ち上がりエッジ) までの間隔は、5 ステート以上確保してください (図 11.22 参照)。本条件は連続送信を行う際にも必要です。

本条件が満たされない場合、誤動作することがあります。

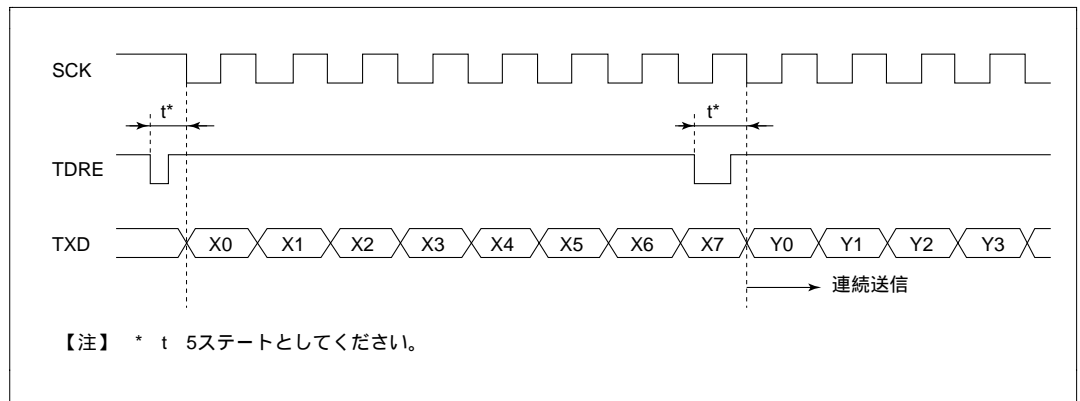


図 11.22 クロック同期式送信時の例

12. A/D 変換器

第 12 章 目次

12.1	概要	385
12.1.1	特長	385
12.1.2	ブロック図	386
12.1.3	端子構成	386
12.1.4	レジスタ構成	387
12.2	各レジスタの説明	388
12.2.1	A/D データレジスタ A ~ D (ADDRA ~ D)	388
12.2.2	A/D コントロール / ステータスレジスタ (ADCSR)	389
12.2.3	A/D コントロールレジスタ (ADCR)	391
12.3	CPU とのインタフェース	392
12.4	動作説明	393
12.4.1	単一モード (SCAN = "0")	393
12.4.2	スキャンモード (SCAN = "1")	395
12.4.3	入力サンプリングと A/D 変換時間	397
12.4.4	外部トリガ入力タイミング	398
12.5	割込み	399
12.6	使用上の注意	399

12.1 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。

12.1.1 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7.4 μ s (18MHz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

外部トリガ信号による、A/D変換の開始が可能

A/D変換終了割り込み要求を発生

A/D変換終了時には、A/D変換終了割り込み (ADI) 要求を発生させることができます。

12.1.2 ブロック図

A/D変換器のブロック図を図12.1に示します。

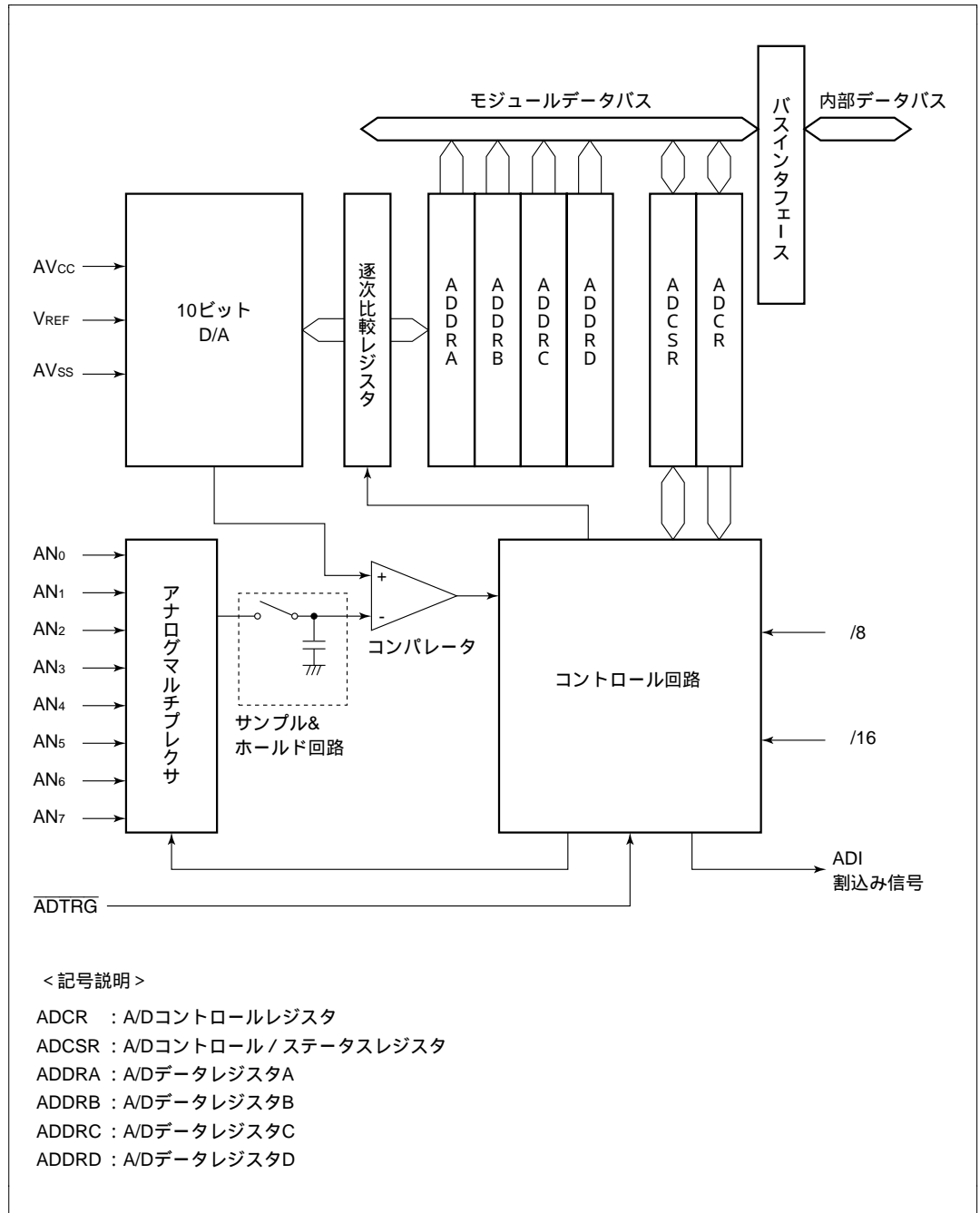


図 12.1 A/D変換器のブロック図

12.1.3 端子構成

A/D変換器で使用する入力端子を表12.1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 (AN₀~AN₃)がグループ0、アナログ入力端子4~7 (AN₄~AN₇)がグループ1になっています。

AV_{CC} 、 AV_{SS} 端子は、A/D 変換器内のアナログ部の電源です。 V_{REF} 端子は、A/D 変換基準電圧端子です。

表 12.1 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
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
リファレンス電圧端子	V_{REF}	入力	アナログ部の基準電圧
アナログ入力端子 0	AN_0	入力	グループ 0 のアナログ入力
アナログ入力端子 1	AN_1	入力	
アナログ入力端子 2	AN_2	入力	
アナログ入力端子 3	AN_3	入力	
アナログ入力端子 4	AN_4	入力	グループ 1 のアナログ入力
アナログ入力端子 5	AN_5	入力	
アナログ入力端子 6	AN_6	入力	
アナログ入力端子 7	AN_7	入力	
A/D 外部トリガ 入力端子	\overline{ADTRG}	入力	A/D 変換時間のための外部トリガ入力

12.1.4 レジスタ構成

A/D 変換器のレジスタ構成を表 12.2 に示します。

表 12.2 レジスタ構成

アドレス*1	名称	略称	R/W	初期値
H'FFE0	A/D データレジスタ AH	ADDRAH	R	H'00
H'FFE1	A/D データレジスタ AL	ADDRAL	R	H'00
H'FFE2	A/D データレジスタ BH	ADDRBH	R	H'00
H'FFE3	A/D データレジスタ BL	ADDRBL	R	H'00
H'FFE4	A/D データレジスタ CH	ADDRCH	R	H'00
H'FFE5	A/D データレジスタ CL	ADDRCL	R	H'00
H'FFE6	A/D データレジスタ DH	ADDRDH	R	H'00
H'FFE7	A/D データレジスタ DL	ADDRDL	R	H'00
H'FFE8	A/D コントロール / ステータスレジスタ	ADCSR	R / (W) *2	H'00
H'FFE9	A/D コントロールレジスタ	ADCR	R/W	H'7F

【注】 *1 アドレスの下位 16 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。

12.2 各レジスタの説明

12.2.1 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~ADDRDの4本があります。

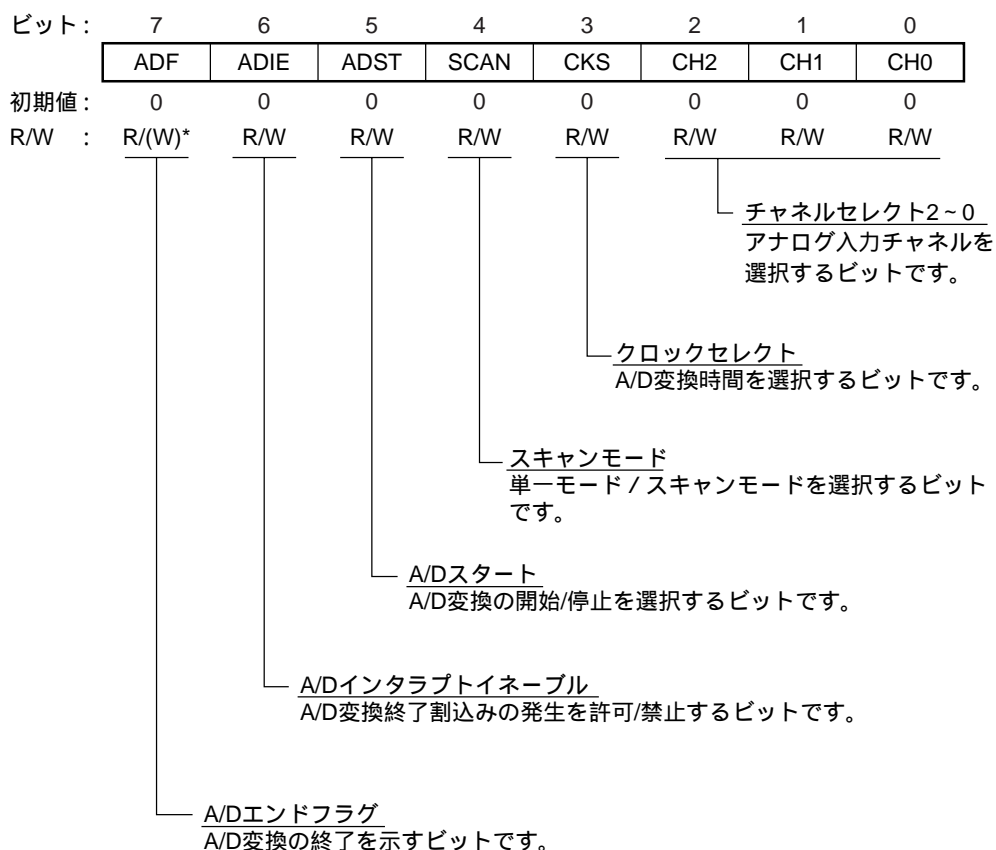
A/D変換されたデータは10ビットデータで、選択されたチャネルのADDRに転送され、保持されます。A/D変換されたデータの上位8ビットがADDRの上位バイトに、また下位2ビットが下位バイトに対応します。ADDRの下位バイトのビット5~0はリザーブビットで、リードすると常に"0"が読み出されます。アナログ入力チャネルとADDRの対応を表12.3に示します。ADDRは、常にCPUからリード可能です。上位バイトは直接リードできますが、下位バイトはテンポラリレジスタ(TEMP)を介してデータ転送が行われます。詳細は「12.3 CPUとのインタフェース」を参照してください。

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 12.3 アナログ入力チャネルとADDRA~ADDRDの対応

アナログ入力チャネル		A/Dデータレジスタ
グループ0	グループ1	
AN ₀	AN ₄	ADDRA
AN ₁	AN ₅	ADDRB
AN ₂	AN ₆	ADDRC
AN ₃	AN ₇	ADDRD

12.2.2 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) 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
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		
ADIE	説明	
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 ($\overline{\text{ADTRG}}$) により"1"にセットすることもできます。

ビット5		
ADST	説明	
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、「12.4 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		
SCAN	説明	
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		
CKS	説明	
0	変換時間 = 266 ステート (max)	(初期値)
1	変換時間 = 134 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	
CH2	CH1	CH0	単一モード	スキャンモード
0	0	0	AN ₀ (初期値)	AN ₀
		1	AN ₁	AN ₀ 、AN ₁
	1	0	AN ₂	AN ₀ ~AN ₂
		1	AN ₃	AN ₀ ~AN ₃
1	0	0	AN ₄	AN ₄
		1	AN ₅	AN ₄ 、AN ₅
	1	0	AN ₆	AN ₄ ~AN ₆
		1	AN ₇	AN ₄ ~AN ₇

12.2.3 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	1
R/W :	R/W	—	—	—	—	—	—	—

└─ トリガイネーブル
└─ リザーブビット

外部トリガによる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'7F 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	説明	
TRGE		
0	外部トリガ入力による A/D 変換の開始を禁止	(初期値)
1	外部トリガ端子 ($\overline{\text{ADTRG}}$) の立下がりエッジで A/D 変換を開始	

ビット6~0：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。

12.3 CPU とのインタフェース

ADDRA ~ ADDRD はそれぞれ 16 ビットのレジスタですが、CPU との間データバスは 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 12.2 に、ADDR のアクセス時のデータの流れを示します。

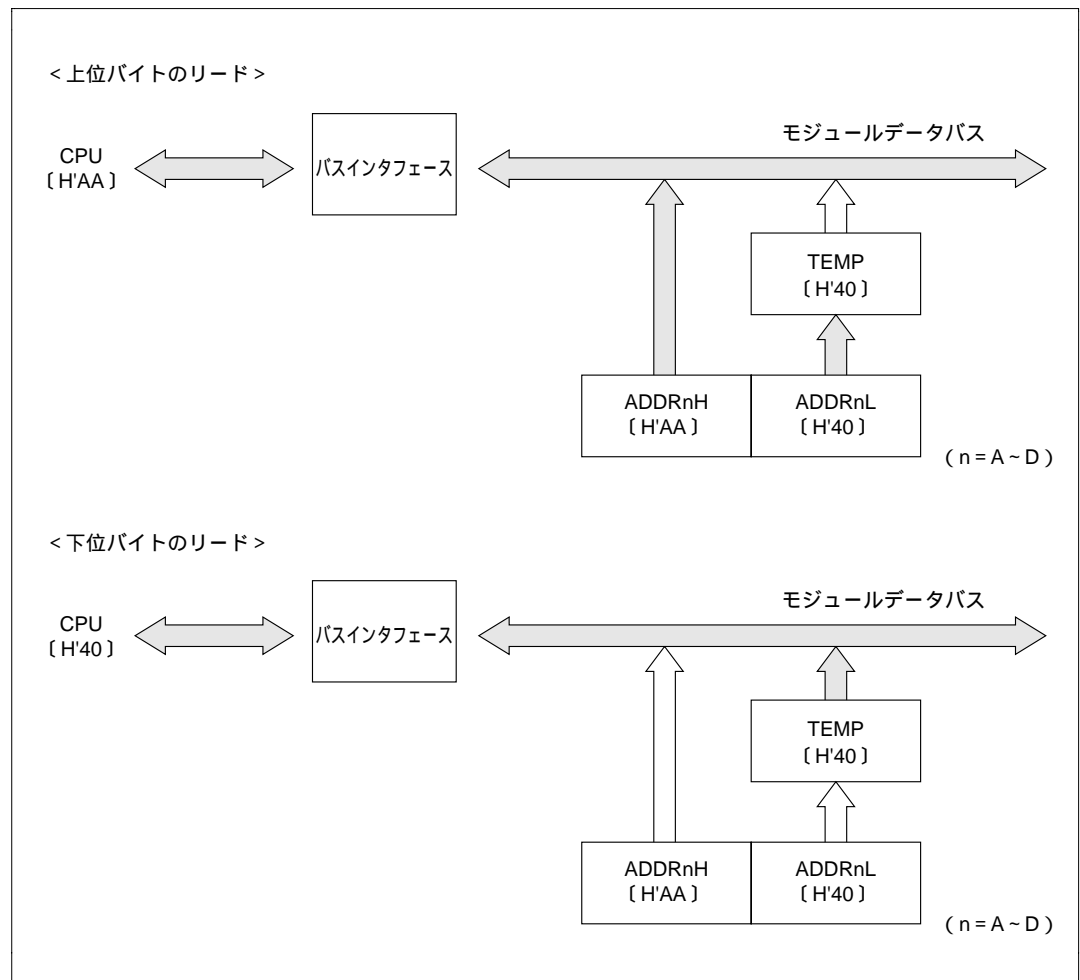


図 12.2 ADDR のアクセス動作 (〔H'AA40〕リード時)

12.4 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。

12.4.1 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。

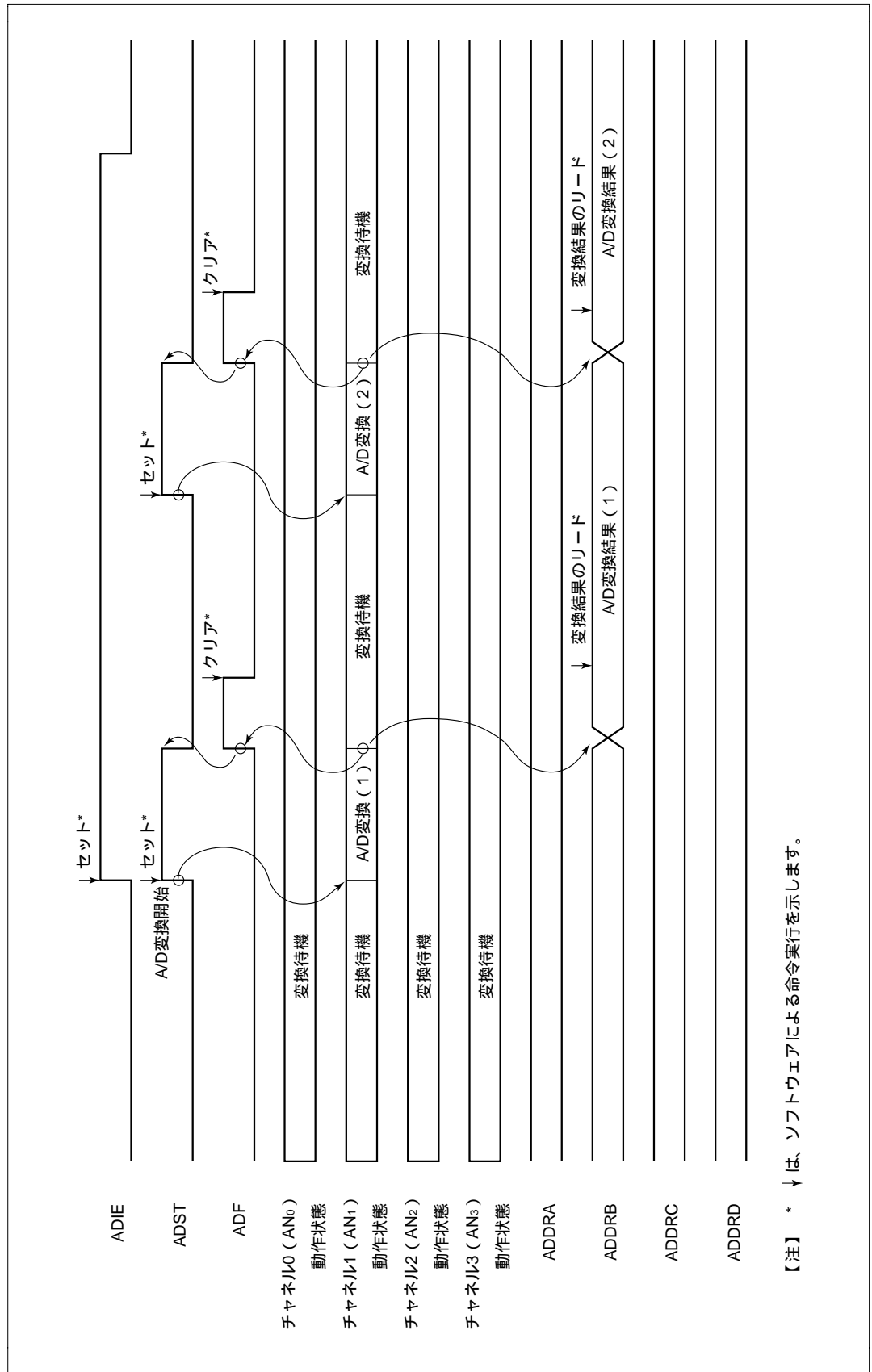
ADF フラグは、ADCSR をリードした後、"0" をライトするとクリアされます。

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 (AN₁) 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 12.3 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 AN₁ 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 ADDR_B 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 (ADDR_B) 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。



【注】 * ↓は、ソフトウェアによる命令実行を示します。

図 12.3 A/D 変換器の動作例 (単一モード: チャンネル1 選択時)

12.4.2 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ー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 AN₀、CH2 = "1" のとき AN₄) 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 (AN₁ または AN₅) 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 (AN₀ ~ AN₂) 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 12.4 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 AN₀ ~ AN₂ (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 (AN₀) 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 ADDRA に転送します。
次に第 2 チャンネル (AN₁) 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 (AN₂) まで変換を行います。
- (4) 選択されたすべてのチャンネル (AN₀ ~ AN₂) の変換が終了すると、ADF = "1" となり、再び第 1 チャンネル (AN₀) 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 (AN₀) から変換が行われます。

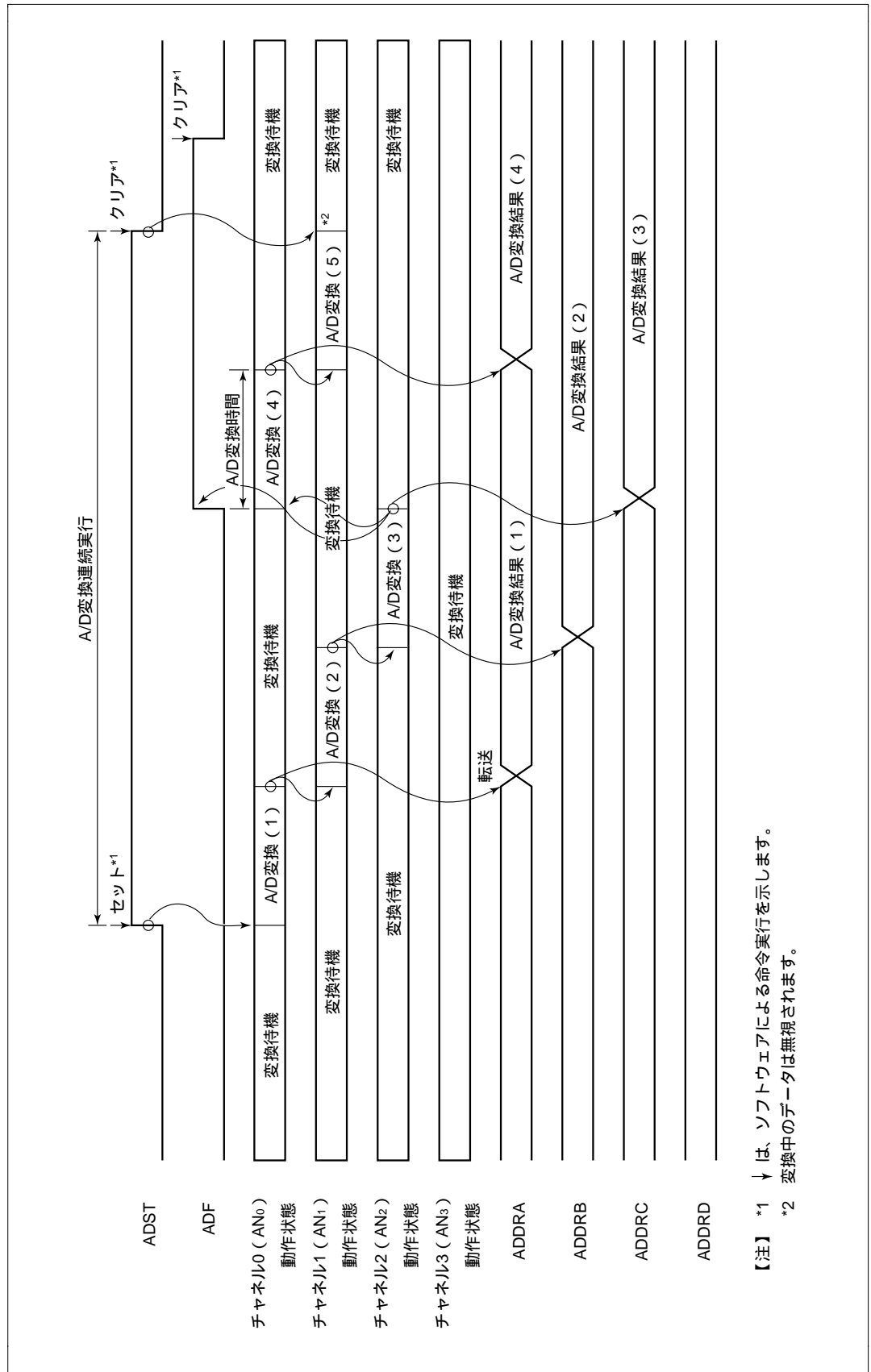


図 12.4 A/D 変換器の動作例 (スキャンモード AN₀ ~ AN₂ の 3 チャンネル選択時)

12.4.3 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12.5に示します。また、A/D変換時間を表12.4に示します。

A/D変換時間は、図12.5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12.4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12.4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256ステート(固定)、CKS="1"の場合は128ステート(固定)となります。

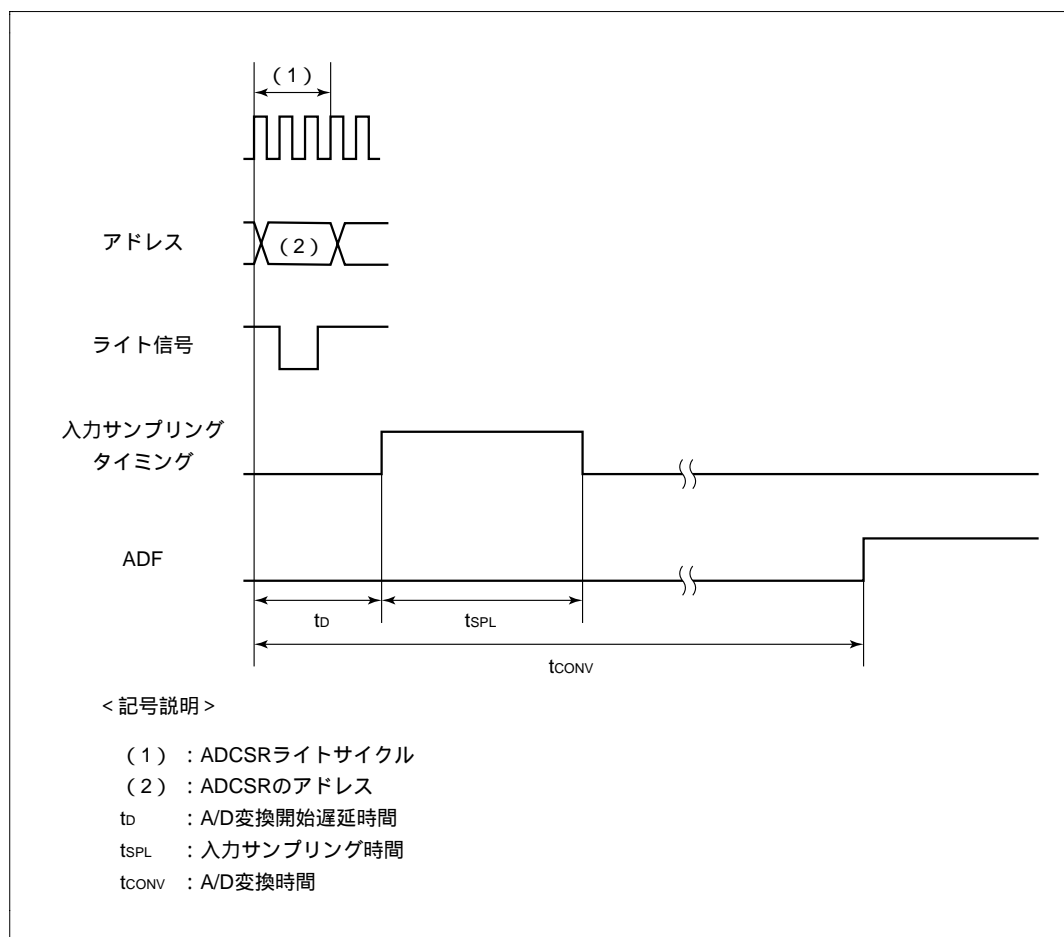


図12.5 A/D変換タイミング

表 12.4 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	10		17	6		9
入力サンプリング時間	t_{SPL}		80			40	
A/D 変換時間	t_{CONV}	259		266	131		134

【注】 表中の数値の単位はステートです。

12.4.4 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 12.6 に示します。

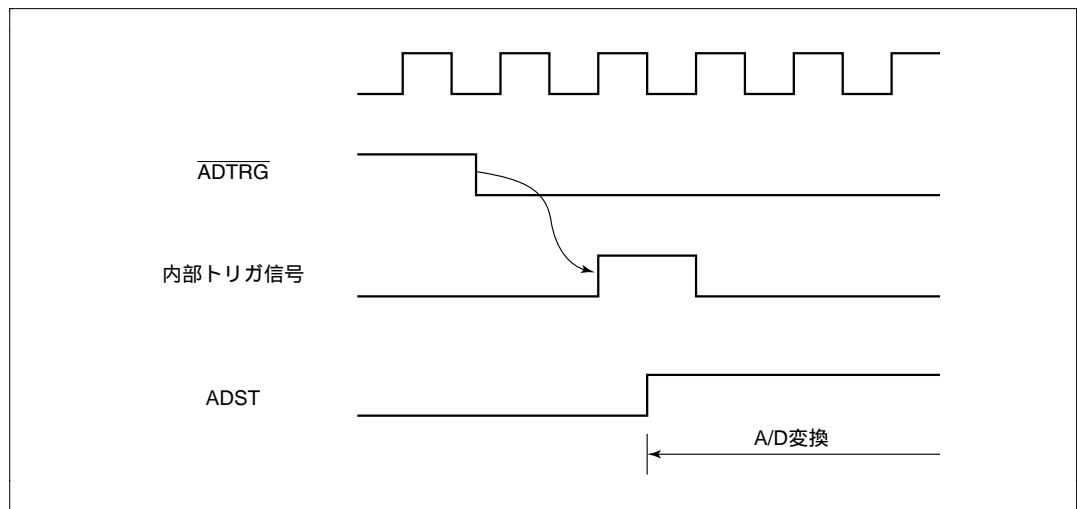


図 12.6 外部トリガ入力タイミング

12.5 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。

12.6 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。(n=0~7)

(2) AV_{CC} 、 AV_{SS} 入力電圧

AV_{SS} 入力電圧は、 $AV_{SS} = V_{SS}$ としてください。A/D変換器を使用しない場合、

$AV_{CC} = V_{CC}$ 、 $AV_{SS} = V_{SS}$ としてください。

(3) V_{REF} 入力電圧

V_{REF} 端子入力電圧のアナログ基準は V_{REF} 、 AV_{CC} としてください。A/D変換器を使用しない場合、 $V_{REF} = V_{CC}$ としてください。

13. RAM

第 13 章 目次

13.1	概要.....	403
	13.1.1	ブロック図.....403
	13.1.2	レジスタ構成404
13.2	システムコントロールレジスタ (SYSCR)	405
13.3	動作説明.....	406
	13.3.1	モード 1.....406
	13.3.2	モード 2.....406
	13.3.3	モード 3.....406

13.1 概要

本LSIは4k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状態で行われます。したがって、データの高速度転送が可能です。

本LSIの内蔵RAMはモード1、3のときH'FEF10~H'FFF0Fに、モード2のとき、H'FEF10~H'FFF0Fに割り当てられており、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。

13.1.1 ブロック図

RAMのブロック図を図13.1に示します。

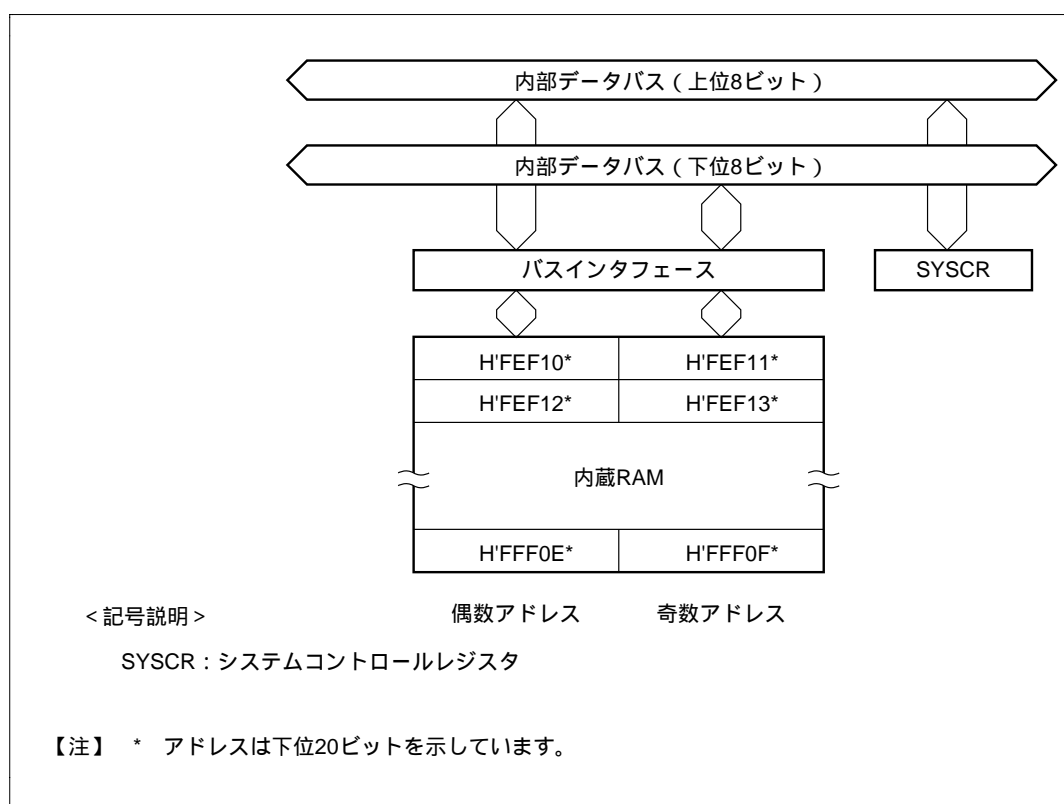


図 13.1 RAMのブロック図 (H8/3035のモード1、3の場合)

13.1.2 レジスタ構成

内蔵RAMは、SYSCRで制御されます。

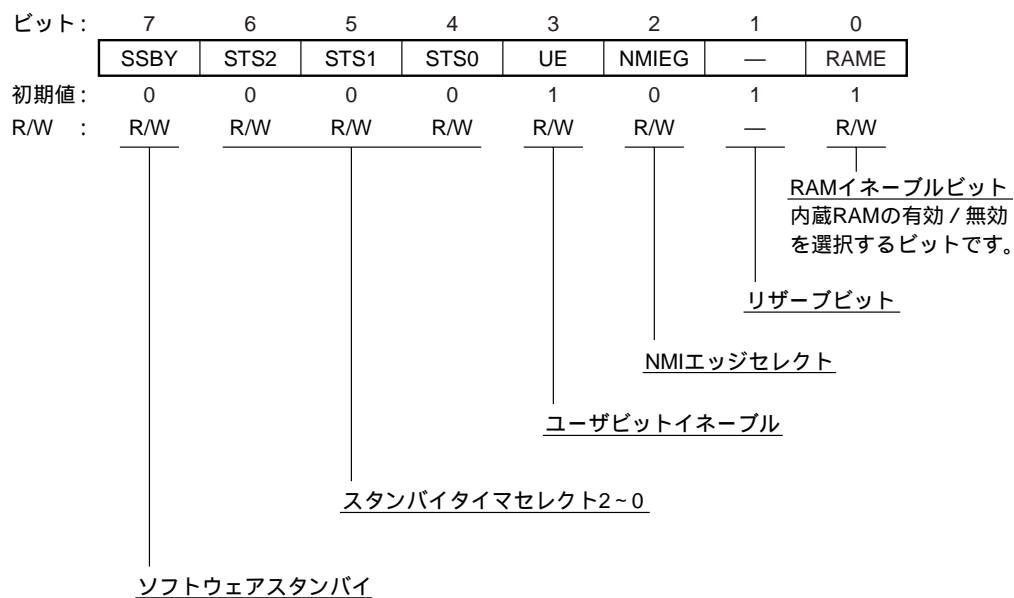
SYSCRのアドレスと初期値を表13.1に示します。

表 13.1 レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFF2	システムコントロールレジスタ	SYSCR	R/W	H'0B

【注】 * アドレスは下位16ビットを示しています。

13.2 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	説明	
0	内蔵RAM無効	
1	内蔵RAM有効	(初期値)

13.3 動作説明

13.3.1 モード 1

モード 1 で、RAME ビット = "1" のとき、本 LSI ではアドレス H'FEF10 ~ H'FFF0F をアクセスすると内蔵 RAM が選択されます。RAME ビット = "0" のとき、アドレス H'FEF10 ~ H'FFF0F をアクセスすると、外部アドレス空間が選択されます。

13.3.2 モード 2

モード 2 で、RAME ビット = "1" のとき、本 LSI ではアドレス H'FFE10 ~ H'FFF0F をアクセスすると内蔵 RAM が選択されます。RAME ビット = "0" のとき、アドレス H'FFE10 ~ H'FFF0F をアクセスすると、外部アドレス空間が選択されます。

13.3.3 モード 3

モード 3 で、RAME ビット = "1" のとき、本 LSI ではアドレス H'FEF10 ~ H'FFF0F をアクセスすると内蔵 RAM が選択されます。RAME ビット = "0" のとき、アドレス H'FEF10 ~ H'FFF0F をアクセスすると、常に H'FF がリードされ、ライトは無効となります。

14. ROM

第 14 章 目次

14.1	概要.....	409
	14.1.1	ブロック図.....409
14.2	PROM モード.....	410
	14.2.1	PROM モードの設定.....410
	14.2.2	ソケットアダプタの端子対応とメモリマップ.....410
14.3	プログラミング.....	413
	14.3.1	書込み / ベリファイ.....413
	14.3.2	書込み時の注意.....417
14.4	書込み後の信頼性.....	418

14.1 概要

H8/3035 は 256k バイト、H8/3034 は 192k バイト、H8/3033 は 128 バイトの ROM を内蔵しています。ROM 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、リ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。

なお、H8/3035 の PROM 版は、PROM モードに設定することにより、本 LSI をサポートする汎用 PROM ライタを用いて、自由にプログラムの書込みができます。

14.1.1 ブロック図

ROM のブロック図を図 14.1 に示します。

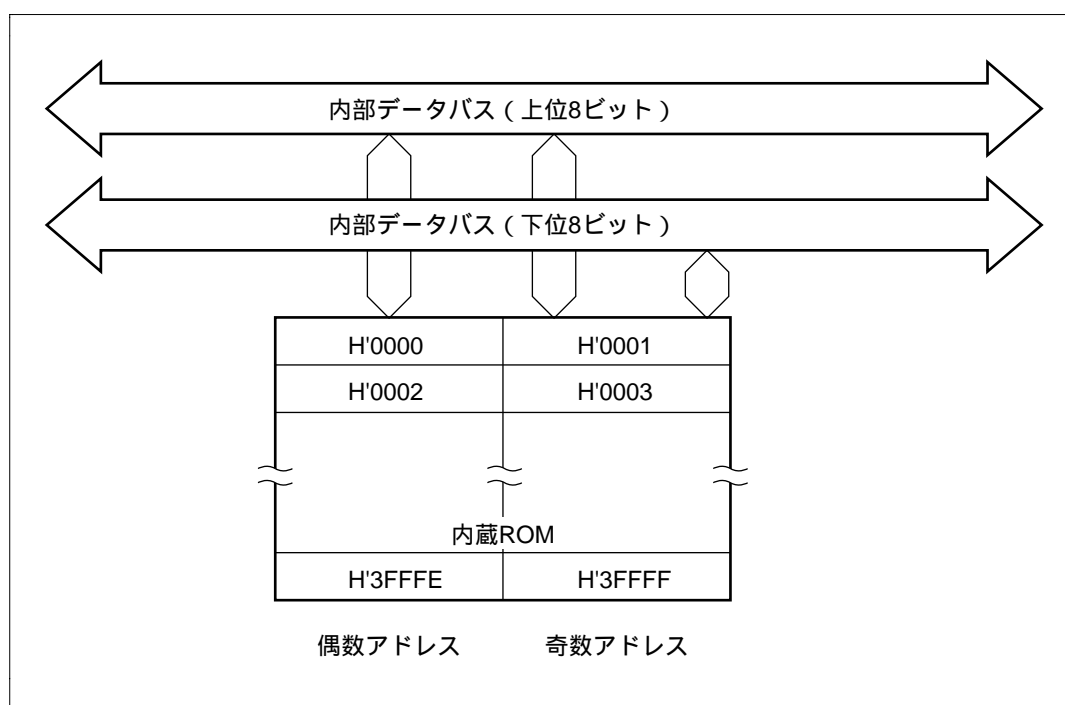


図 14.1 ROM のブロック図 (H8/3035 の場合)

14.2 PROM モード

14.2.1 PROM モードの設定

H8/3035 の PROM 版の場合、PROM モードに設定するとマイクロコンピュータとしての機能が停止して、HN27C4001 と同一の端子機能で内蔵 PROM のプログラムを行うことができます。ただし、本 LSI をサポートする PROM ライタを用いてください。

また、ページプログラミング方式、オプションページプログラミング方式はサポートしていません。PROM モードの設定方法を表 14.1 に示します。

表 14.1 PROM モード設定

端子名	設定
モード端子 (MD ₁ 、MD ₀) の 2 端子	"Low" レベル
STBY 端子	
P5 ₁ 、P5 ₀ 、P9 ₀ 、P9 ₄ 、PA ₁ 端子	"High" レベル
PA ₀ 端子	"Low" レベル

14.2.2 ソケットアダプタの端子対応とメモリマップ

PROM のプログラムは、表 14.2 で示すように各パッケージに対応した、ソケットアダプタを付けて 32 ピンに変換し、本 LSI をサポートする PROM ライタで行います。ソケットアダプタの端子対応図を図 14.2 に示します。また、メモリマップを図 14.3 に示します。

表 14.2 ソケットアダプタ

製品名	パッケージ名	ソケットアダプタの型名	
		ミナトエレクトロニクス 株式会社製	データ・アイ・オー・ ジャパン株式会社製
H8/3035	80 ピン QFP (FP-80A)	ME3035ESH01H	H73035Q080D3201
	80 ピン TQFP (TFP-80C)	ME3035ESN01H	H73035T080D3201

H8/3035 の PROM の容量は、256k バイトです。PROM モードのときのメモリマップを図 14.3 に示します。内蔵 PROM 内の未使用のアドレス領域のデータは、H'FF としてください。

H8/3035 を PROM ライタでプログラムする際に、アドレスは H'00000 ~ H'3FFFF に設定し、H'40000 以降のデータは H'FF としてください。誤って H'40000 以降にプログラムすると、PROM の書込みや確認ができなくなることがあります。また、ページプログラミング方式、オプションページプログラミング方式でプログラムしようとした場合も同様です。

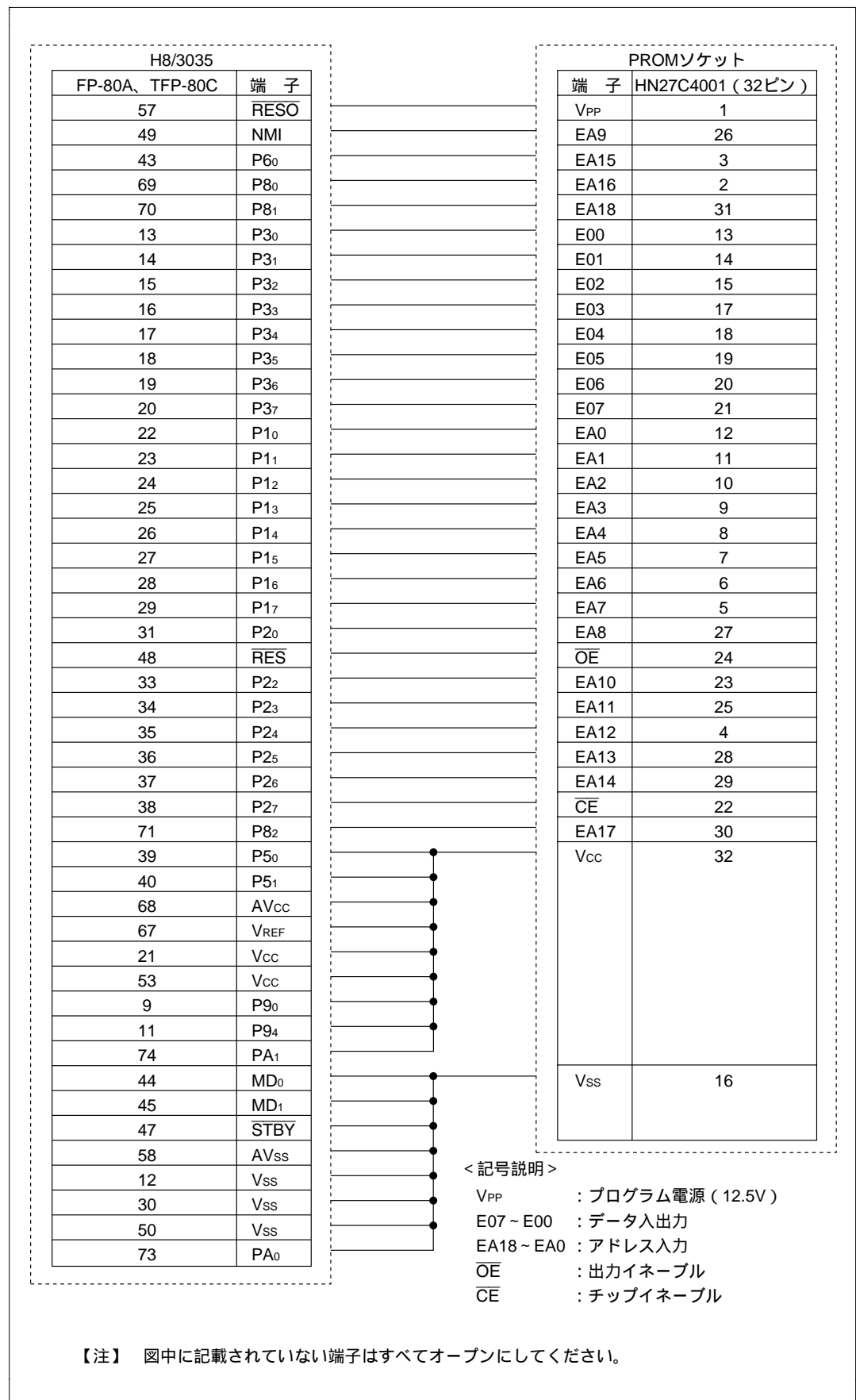


図 14.2 ソケットアダプタの端子対応図

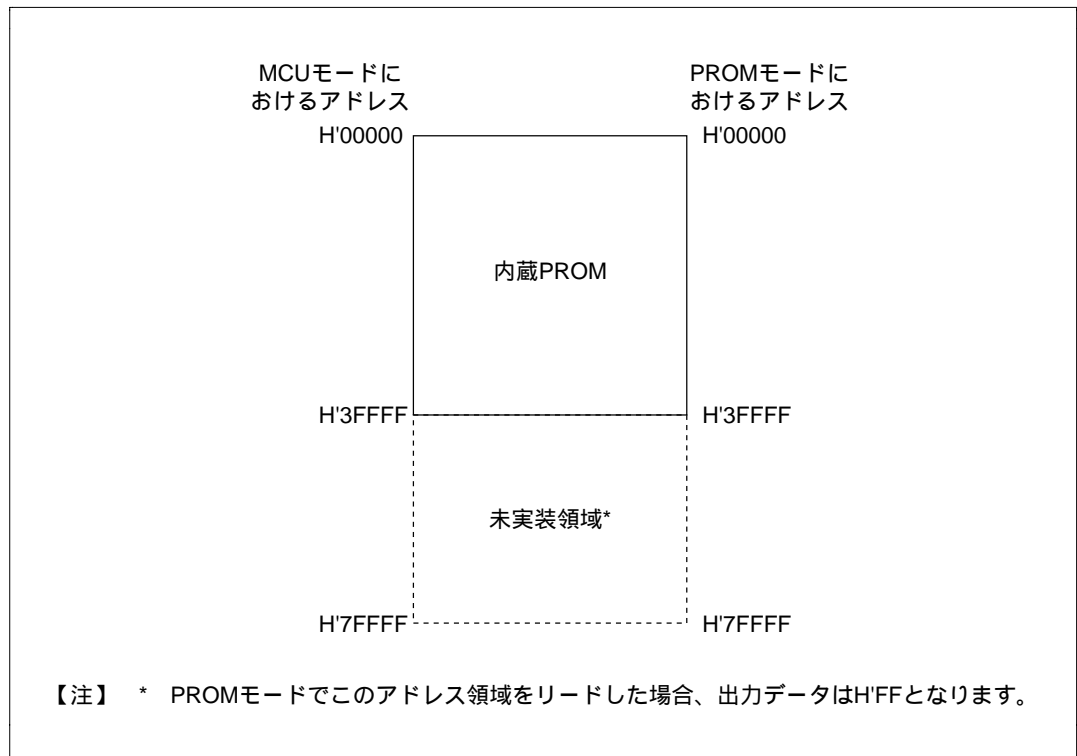


図 14.3 H8/3035 の PROM モード時のメモリマップ

14.3 プログラミング

PROM モード時の書込み、ベリファイなどのモード選択は、表 14.3 に示すような設定によって行います。

表 14.3 PROM モード時のモード選択

モード	ピン					
	\overline{CE}	\overline{OE}	V_{PP}	V_{CC}	$EO_7 \sim EO_0$	$EA_{18} \sim EA_0$
書込み	L	H	V_{PP}	V_{CC}	データ入力	アドレス入力
ベリファイ	H	L	V_{PP}	V_{CC}	データ出力	アドレス入力
プログラミング禁止	H	H	V_{PP}	V_{CC}	ハイインピーダンス	アドレス入力

記号説明

L : "Low"レベル

H : "High"レベル

V_{PP} : " V_{PP} "レベル

V_{CC} : " V_{CC} "レベル

ただし、ページプログラミング方式、オプションページプログラミング方式はサポートしていませんので、ページプログラミングモード、オプションページプログラミングモードに設定しないでください。PROM ライタを選択する場合には、本 LSI をサポートしていることを確認してください。また、アドレスは必ず H'00000 ~ H'3FFFF に設定してください。

14.3.1 書込み / ベリファイ

書込み / ベリファイは効率のよい高速プログラミング方式で行うことができます。この方式は、デバイスへの電圧ストレス、あるいは書込みデータの信頼性を損なうことなく高速な書込みを行うことができます。未使用のアドレス領域のデータは H'FF です。

高速プログラミングの基本的なフローを図 14.4 に示します。

また、プログラミング時の電気的特性を表 14.4、表 14.5 に、タイミングを図 14.5 に示します。

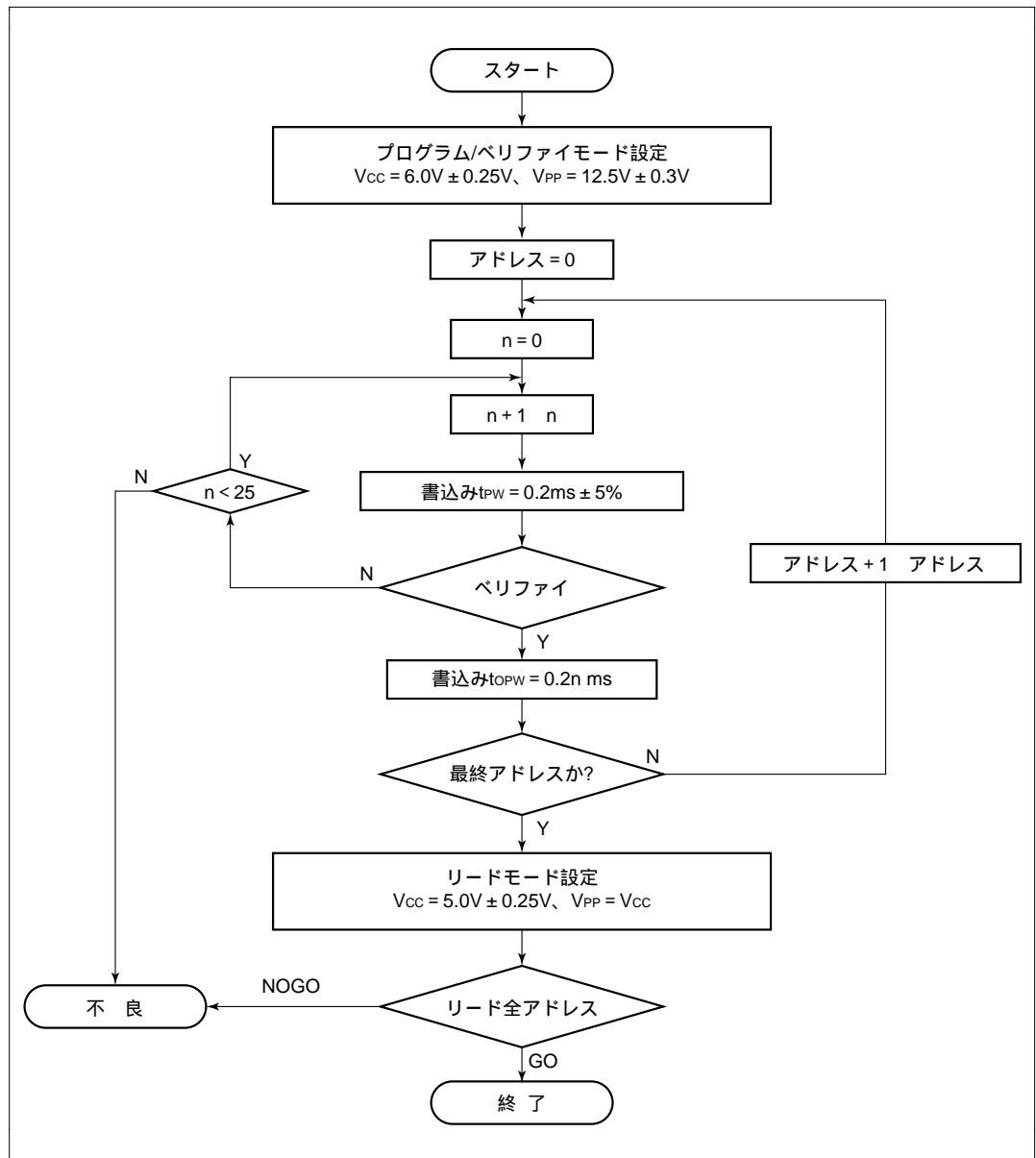


図 14.4 高速プログラミングフローチャート

表 14.4 DC 特性

(条件: $V_{CC} = 6.0V \pm 0.25V$ 、 $V_{PP} = 12.5V \pm 0.3V$ 、 $V_{SS} = 0V$ 、 $T_a = 25 \pm 5$)

項目	記号	min	typ	max	単位	測定条件
入力"High"レベル 電圧	$EO_7 \sim EO_0$ 、 EA_{18} $\sim EA_0$ 、 \overline{OE} 、 \overline{CE}	V_{IH}	2.4		$V_{CC} + 0.3$	V
入力"Low"レベル 電圧	$EO_7 \sim EO_0$ 、 EA_{18} $\sim EA_0$ 、 \overline{OE} 、 \overline{CE}	V_{IL}	- 0.3		0.8	V
出力"High"レベル 電圧	$EO_7 \sim EO_0$	V_{OH}	2.4			V $I_{OH} = - 200\mu A$
出力"Low"レベル 電圧	$EO_7 \sim EO_0$	V_{OL}			0.45	V $I_{OL} = 1.6mA$
入力リーク電流	$EO_7 \sim EO_0$ 、 EA_{18} $\sim EA_0$ 、 \overline{OE} 、 \overline{CE}	$ I_{LI} $			2	μA $V_{in} = 5.25V/0.5V$
V_{CC} 電流		I_{CC}			40	mA
V_{PP} 電流		I_{PP}			40	mA $\overline{CE} = V_{IL}$

表 14.5 AC 特性

(条件: $V_{CC} = 6.0V \pm 0.25V$ 、 $V_{PP} = 12.5V \pm 0.3V$ 、 $T_a = 25 \pm 5$)

項目	記号	min	typ	max	単位	測定条件
アドレスセットアップ時間	t_{AS}	2			μs	図 14.5*1
\overline{OE} セットアップ時間	t_{OES}	2			μs	
データセットアップ時間	t_{DS}	2			μs	
アドレスホールド時間	t_{AH}	0			μs	
データホールド時間	t_{DH}	2			μs	
データ出力ディスエーブル時間	t_{DF}^{*2}			130	ns	
V_{PP} セットアップ時間	t_{VPS}	2			μs	
プログラミング中の \overline{CE} パルス幅	t_{PW}	0.19	0.20	0.21	ms	
オーバプログラミング中の \overline{CE} パルス幅	t_{OPW}^{*3}	0.19		5.25	ms	
V_{CC} セットアップ時間	t_{VCS}	2			μs	
データ出力遅延時間	t_{OE}	0		150	ns	

【注】 *1 入力パルス・レベル : 0.8 ~ 2.2V

入力立上がり / 立下がり時間 20ns

タイミング参照レベル { 入力 : 1.0V、2.0V
出力 : 0.8V、2.0V

*2 t_{DF} は出力が開放状態に達し、出力レベルを参照できなくなった場合で定義します。

*3 t_{OPW} はフローチャートに記載した値で定義されます。

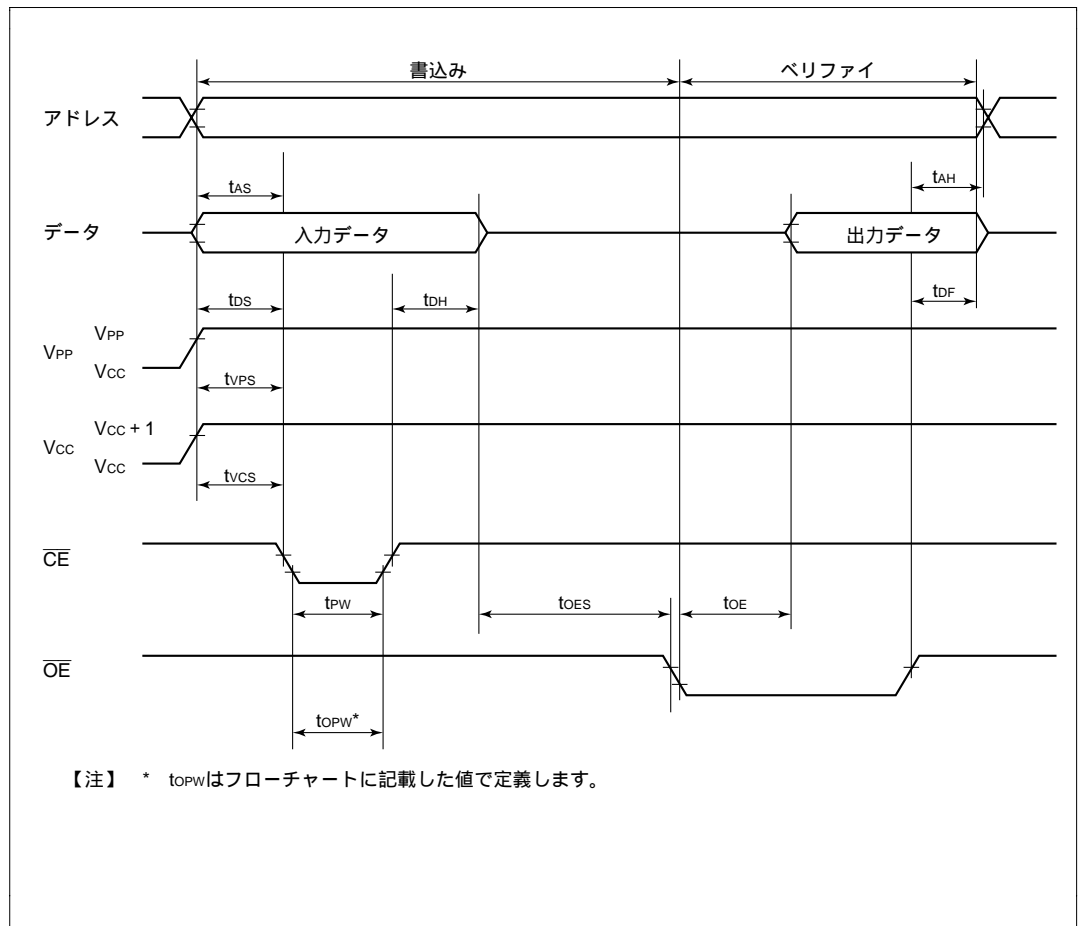


図 14.5 PROM 書込み / ベリファイ・タイミング

14.3.2 書込み時の注意

- (1) 書込みは規定された電圧、タイミングで行ってください。
PROMモード時のプログラム電圧 (V_{pp}) は 12.5V です。
定格以上の電圧を加えると、製品の永久破壊に至りますので、注意してください。
特に PROM ライタのオーバシュートなどには十分注意してください。
PROM ライタを本 LSI にセットすると、 V_{pp} は 12.5V になります。
- (2) 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、過剰電流によって製品が破壊することがあります。書込み前に正しく 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。
- (3) 書込み中はソケットアダプタおよび製品には触れないようにしてください。接触不良により書込み不良となる場合があります。
- (4) プログラミングモードは、ページプログラミング方式、オプションページプログラミング方式をサポートしていませんので、プログラミングモードの設定には注意してください。
- (5) H8/3035 の PROM の容量は 256k バイトです。アドレスは、必ず H'00000 ~ H'3FFFF に設定してください。プログラムする際には、未使用のアドレス領域 (H'40000 ~ H'7FFFF) のデータは H'FF としてください。

14.4 書込み後の信頼性

データ書込み後、データ保持特性を向上させるために、150 の高温放置をしてスクリーニングを行うと大変有効です。高温放置は、スクリーニングの1つであり、PROMメモリの初期のデータ保持不良を短時間で除くことができます。

図 14.6 に推奨するスクリーニングフローを示します。



図 14.6 推奨スクリーニングフロー

同じ、PROMライターでプログラミング中、書込み不良が連続して発生した場合には書込みを中止し、PROMライター、ソケットアダプタなどに異常がないか確認してください。

書込みあるいは高温放置後のプログラム確認において異常がありましたら、当社技術担当にご連絡ください。

15. クロック発振器

第 15 章 目次

15.1	概要.....	421	
	15.1.1	ブロック図..... 421	
15.2	発振器	422	
	15.2.1	水晶発振子を接続する方法	422
	15.2.2	外部クロックを入力する方法.....	423
15.3	デューティ補正回路	426	
15.4	プリスケータ	426	

15.1 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。

クロック発振器は、発振器、デューティ補正回路、およびプリスケアラから構成されます。

15.1.1 ブロック図

図 15.1 にクロック発振器のブロック図を示します。

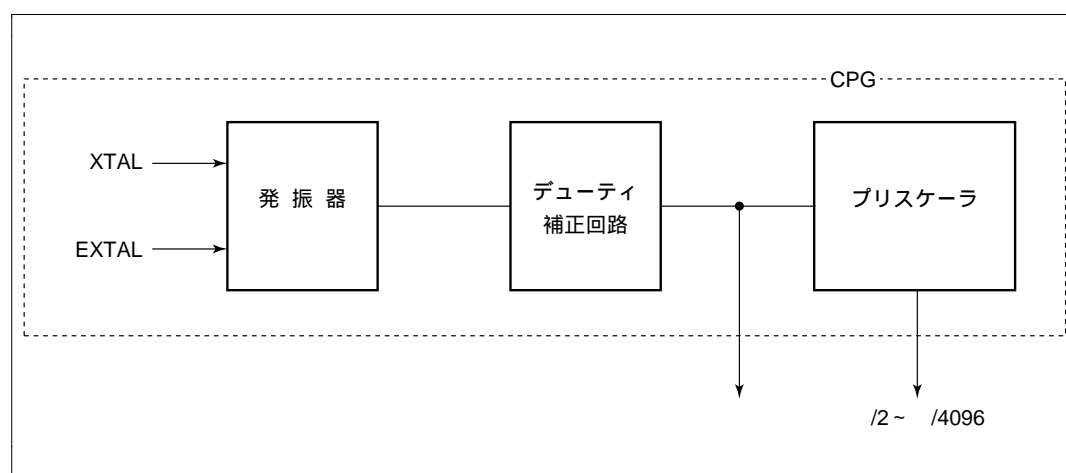


図 15.1 クロック発振器のブロック図

15.2 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。

15.2.1 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 15.2 に示します。ダンピング抵抗 R_d は、表 15.1 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。

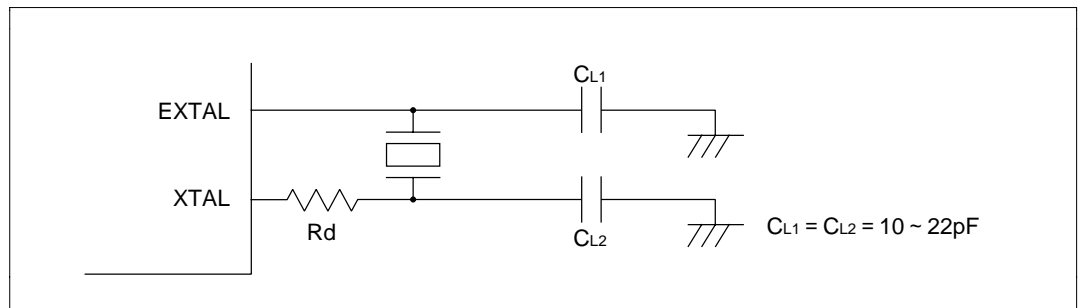


図 15.2 水晶発振子を接続する場合の接続例

表 15.1 ダンピング抵抗値

周波数 (MHz)	2	4	8	10	12	16	18
R_d ()	1k	500	200	0	0	0	0

(2) 水晶発振子

図 15.3 に水晶発振子の等価回路を示します。水晶発振子は表 15.2 に示す特性のものを使用してください。

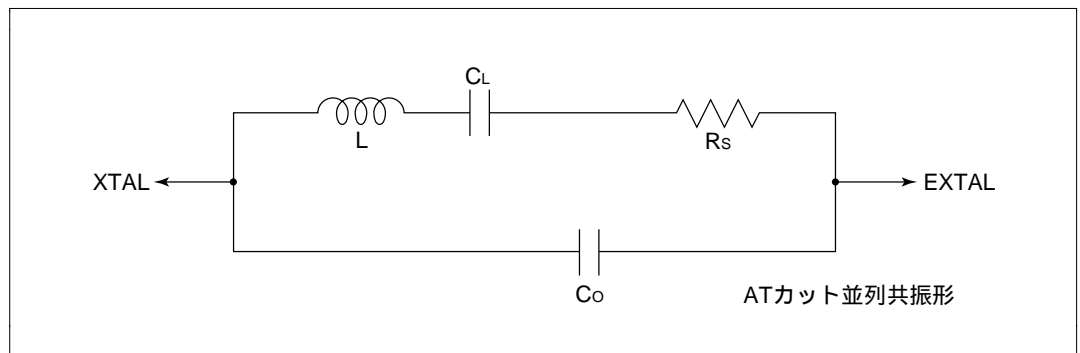


図 15.3 水晶発振子の等価回路

表 15.2 水晶発振子のパラメータ

周波数 (MHz)	2	4	8	10	12	16	18
Rs max ()	500	120	80	70	60	50	40
C ₀ (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 15.4)。

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。

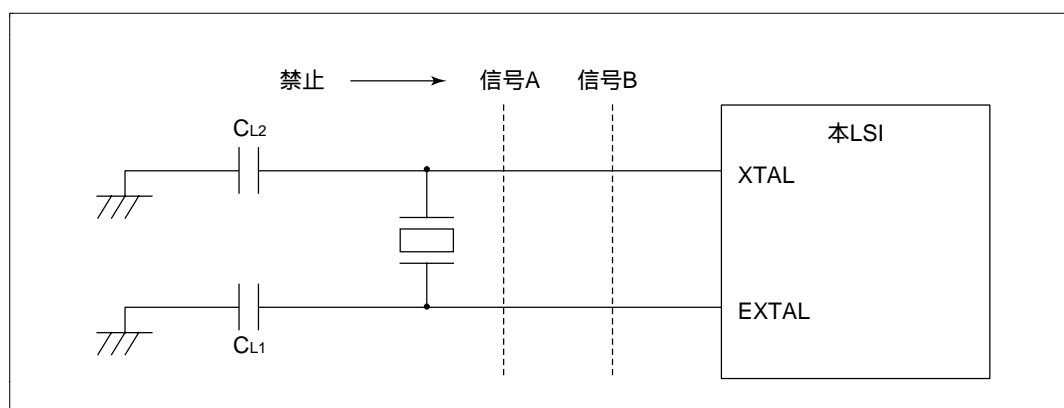


図 15.4 発振回路部のボード設計に関する注意事項

15.2.2 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 15.5 に示します。図 15.5 (b) の場合、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。

XTAL 端子をオープン状態にする場合は、寄生容量が 10pF 以下とってください。

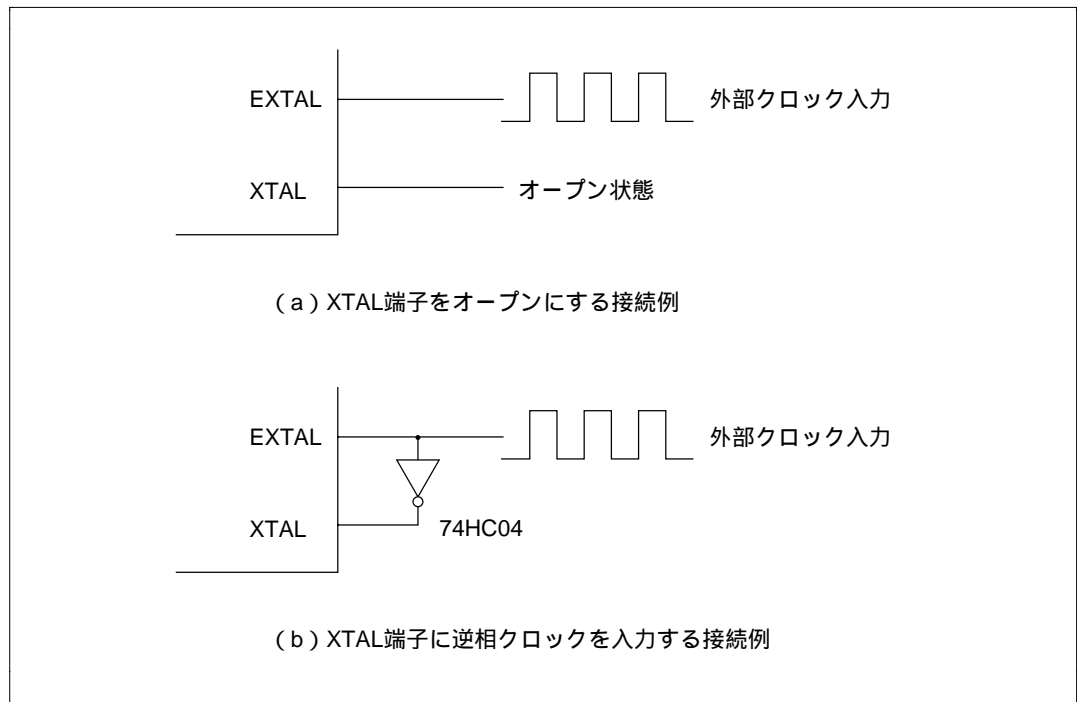


図 15.5 外部クロックを入力する場合の接続例

(2) 外部クロック

外部クロックは と同一の周波数にしてください。表 15.3 と図 15.6 にクロックタイミングを示します。

表 15.3 クロックタイミング

項目	記号	$V_{CC} = 2.7 \sim 5.5V$		$V_{CC} = 5.0V \pm 10\%$		単位	測定条件	
		min	max	min	max			
外部クロック 立ち上がり時間	t_{EXr}		10		5	ns	図 15.6	
外部クロック 立下がり時間	t_{EXf}		10		5	ns		
外部クロック 入力デューティ (a/t_{cyc})		30	70	30	70	%	5MHz	図 15.6
		40	60	40	60	%	< 5MHz	
クロック幅 デューティ (b/t_{cyc})		40	60	40	60	%		

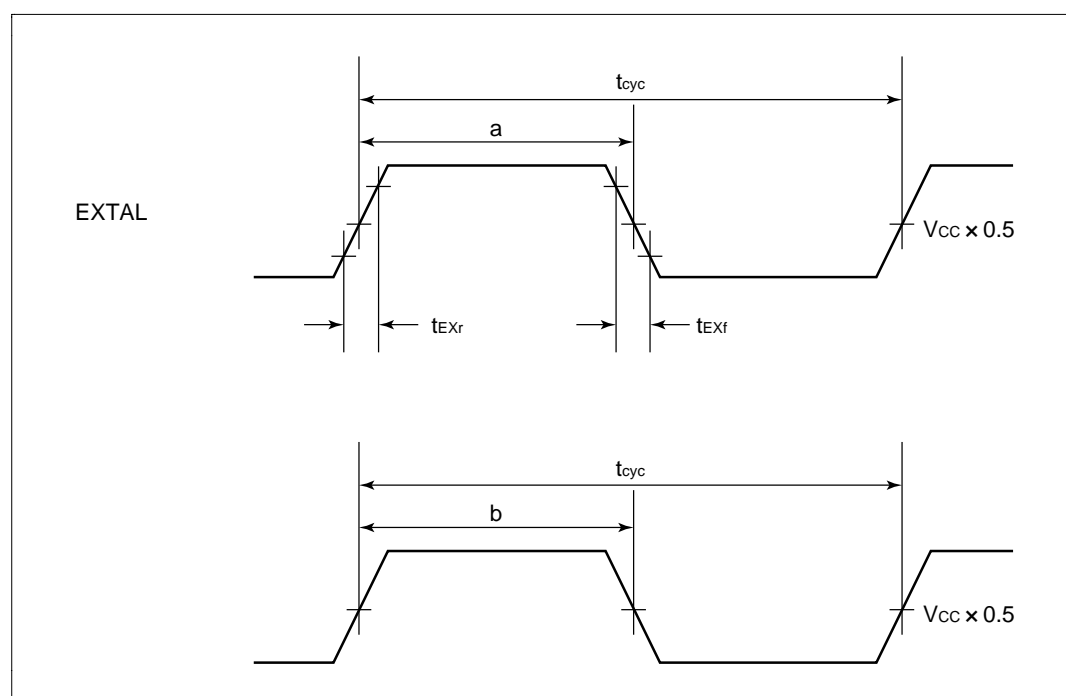


図 15.6 外部クロック入力タイミング

表 15.4 に外部クロック出力安定遅延時間、図 15.7 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_{DEXT}) 経過後に内部のクロック信号出力が確定します。 t_{DEXT} 期間中はクロック信号出力が確定していませんので、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 15.4 外部クロック出力安定遅延時間

[条件: $V_{\text{CC}} = 2.7\text{V} \sim 5.5\text{V}$ 、 $AV_{\text{CC}} = 2.7\text{V} \sim 5.5\text{V}$ 、 $V_{\text{SS}} = AV_{\text{SS}} = 0\text{V}$]

項目	記号	min	max	単位	備考
外部クロック出力安定遅延時間	t_{DEXT}^*	500		μs	図 15.7

【注】 * t_{DEXT} は、RES パルス幅 (t_{RESW}) を $10t_{\text{cyc}}$ 含みます。

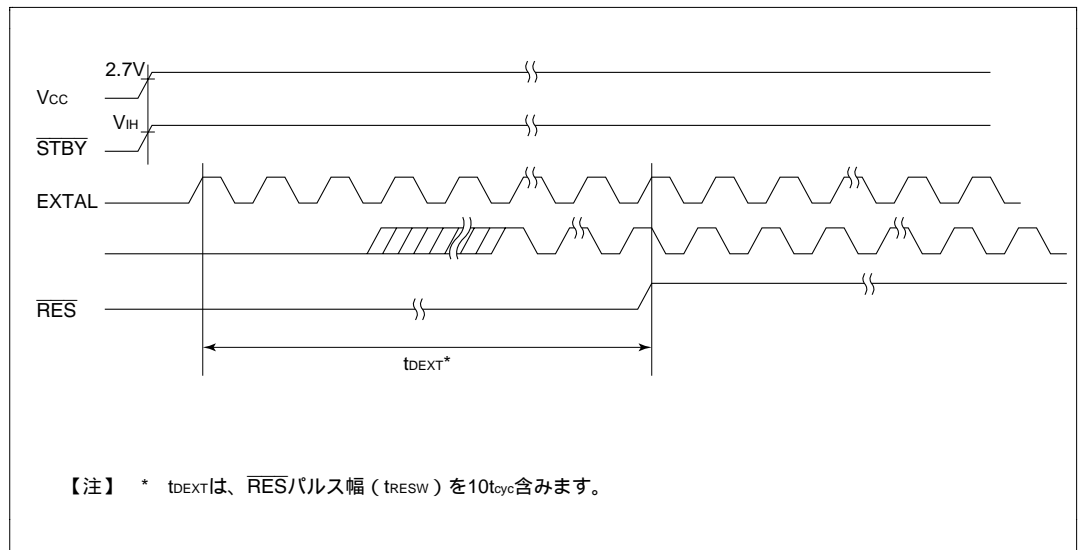


図 15.7 外部クロック出力安定遅延時間タイミング

15.3 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。

15.4 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。

16. 低消費電力状態

第 16 章 目次

16.1	概要	429
16.2	レジスタ構成	430
16.2.1	システムコントロールレジスタ (SYSCR)	430
16.3	スリープモード	432
16.3.1	スリープモードへの遷移	432
16.3.2	スリープモードの解除	432
16.4	ソフトウェアスタンバイモード	433
16.4.1	ソフトウェアスタンバイモードへの遷移	433
16.4.2	ソフトウェアスタンバイモードの解除	433
16.4.3	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	434
16.4.4	ソフトウェアスタンバイモードの応用例	435
16.4.5	使用上の注意	435
16.5	ハードウェアスタンバイモード	436
16.5.1	ハードウェアスタンバイモードへの遷移	436
16.5.2	ハードウェアスタンバイモードの解除	436
16.5.3	ハードウェアスタンバイモードのタイミング	436

16.1 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

各モードへ遷移する条件と CPU や周辺機能などの状態、および各モードからの解除方法を表 16.1 に示します。

表 16.1 低消費電力状態

モード	遷移条件	状態						解除方法
		クロック	CPU	レジスタ	周辺機能	RAM	I/O ポート	
スリープモード	SYSCR の SSBY = "0" の状態で SLEEP 命令を実行	動作	停止	保持	動作	保持	保持	<ul style="list-style-type: none"> ・ 割込み ・ $\overline{\text{RES}}$ 端子 ・ $\overline{\text{STBY}}$ 端子
ソフトウェアスタンバイモード	SYSCR の SSBY = "1" の状態で SLEEP 命令を実行	停止	停止	保持	停止リセット	保持	保持	<ul style="list-style-type: none"> ・ NMI 端子 ・ $\overline{\text{IRQ}}_0 \sim \overline{\text{IRQ}}_2$ 端子 ・ $\overline{\text{RES}}$ 端子 ・ $\overline{\text{STBY}}$ 端子
ハードウェアスタンバイモード	$\overline{\text{STBY}}$ 端子を "Low" レベル	停止	停止	不定	停止リセット	保持*	ハイインピーダンス	<ul style="list-style-type: none"> ・ $\overline{\text{STBY}}$ 端子 ・ $\overline{\text{RES}}$ 端子

【注】 * プログラム実行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。

《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

16.2 レジスタ構成

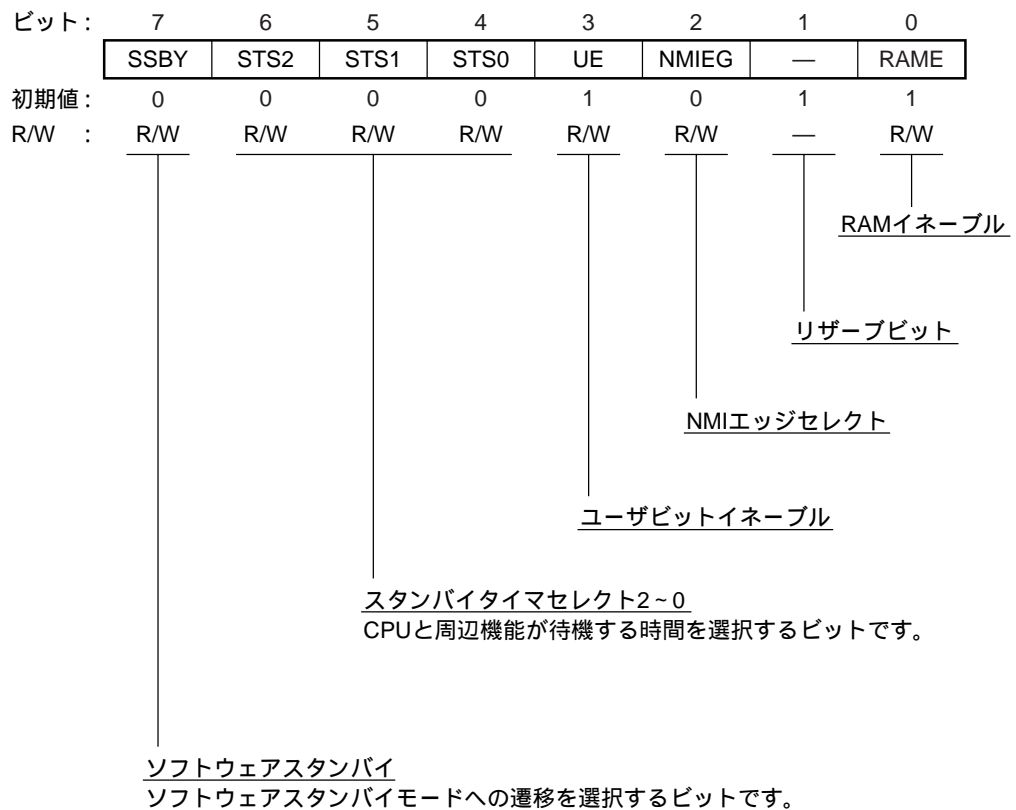
本 LSI には低消費電力状態の制御を行うシステムコントロールレジスタ (SYSCR) があります。レジスタ構成を表 16.2 に示します。

表 16.2 レジスタ構成

アドレス*	名称	略称	R/W	初期値
H'FFF2	システムコントロールレジスタ	SYSCR	R/W	H'0B

【注】 * アドレスの下位 16 ビットを示しています。

16.2.1 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 16.3 を参照し動作周波数に応じて待機時間が 8ms（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、任意の選択が可能です。

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)
		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		待機時間 = 131072 ステート
	1		使用禁止

16.3 スリープモード

16.3.1 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内蔵周辺モジュールの機能は停止しません。

16.3.2 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 CPU が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せん。

(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。

16.4 ソフトウェアスタンバイモード

16.4.1 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。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 ポートの状態も保持されています。

16.4.2 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、 $\overline{IRQ_0}$ ~ $\overline{IRQ_2}$ 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 ~ 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 ~ 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{STBY} 端子による解除

\overline{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。

16.4.3 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 ビットの設定は、以下のようになしてください。

(1) 水晶発振の場合

待機時間が 8ms (発振安定時間) 以上となるように STS2 ~ STS0 を設定してください。

表 16.3 に動作周波数と STS2 ~ STS0 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

任意の値を選択可能です。

表 16.3 動作周波数と発振安定待機時間

STS2	STS1	STS0	待機時間	18MHz	16MHz	12MHz	10MHz	8MHz	6MHz	4MHz	2MHz	単位
0	0	0	8192 ステート	0.46	0.51	0.65	0.8	1.0	1.3	2.0	4.1	ms
0	0	1	16384 ステート	0.91	1.0	1.3	1.6	2.0	2.7	4.1	8.2	
0	1	0	32768 ステート	1.8	2.0	2.7	3.3	4.1	5.5	8.2	16.4	
0	1	1	65536 ステート	3.6	4.1	5.5	6.6	8.2	10.9	16.4	32.8	
1	0		131072 ステート	7.3	8.2	10.9	13.1	16.4	21.8	32.8	65.5	
1	1		使用禁止									

【注】 : 推奨設定時間

16.4.4 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 16.1 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。

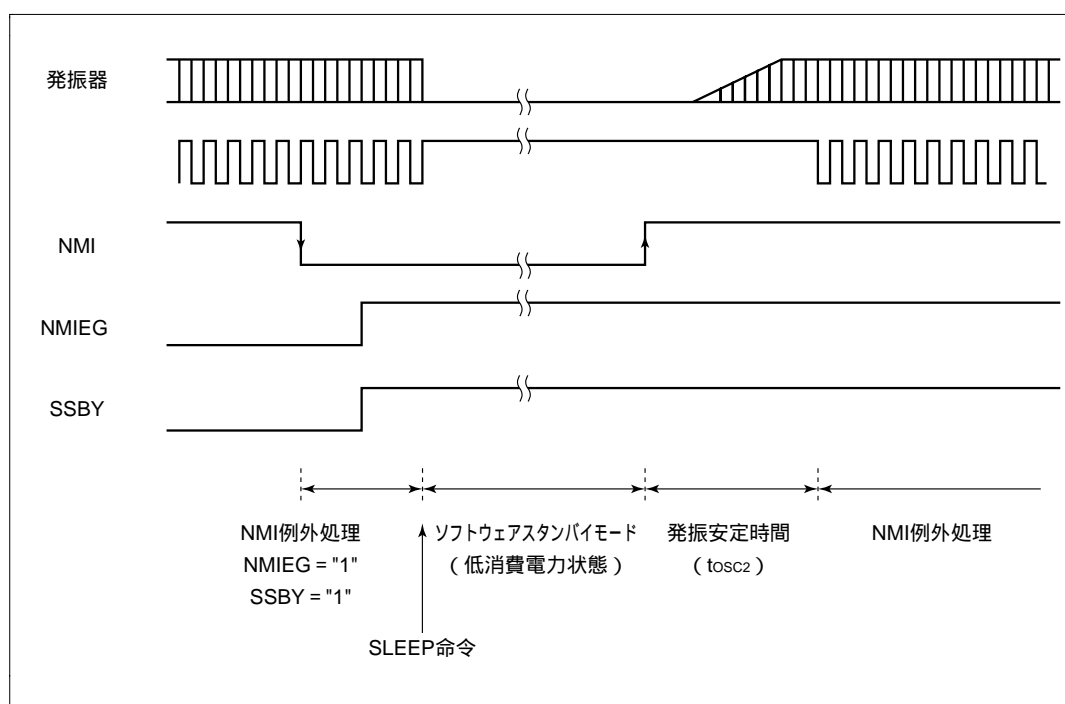


図 16.1 ソフトウェアスタンバイモード時の NMI タイミング (例)

16.4.5 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。

16.5 ハードウェアスタンバイモード

16.5.1 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、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 (MD_1 、 MD_0) の状態を変化させないでください。

16.5.2 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。

16.5.3 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 16.2 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。

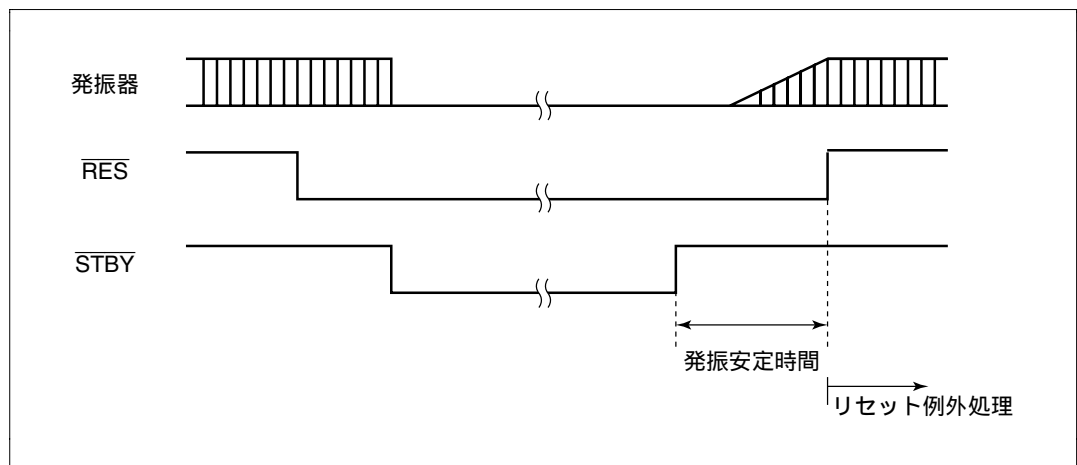


図 16.2 ハードウェアスタンバイモードのタイミング

17. 電気的特性（暫定仕様）

第 17 章 目次

17.1	絶対最大定格.....	439
17.2	電気的特性.....	440
	17.2.1 DC 特性.....	440
	17.2.2 AC 特性.....	447
	17.2.3 A/D 変換特性.....	453
17.3	動作タイミング.....	454
	17.3.1 バスタイミング.....	454
	17.3.2 制御信号タイミング.....	456
	17.3.3 クロックタイミング.....	457
	17.3.4 TPC、I/O ポートタイミング.....	458
	17.3.5 ITU タイミング.....	458
	17.3.6 SCI 入出力タイミング.....	459

17.1 絶対最大定格

絶対最大定格を表 17.1 に示します。

表 17.1 絶対最大定格

項目	記号	定格値	単位
電源電圧	V_{CC}	- 0.3 ~ + 7.0	V
プログラム電圧	V_{PP}	- 0.3 ~ + 13.5	V
入力電圧（ポート 7 以外）	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}	- 0.3 ~ + 7.0	V
アナログ入力電圧	V_{AN}	- 0.3 ~ $AV_{CC} + 0.3$	V
動作温度	T_{opr}	通常仕様品： - 20 ~ + 75	
		広温度範囲仕様品： - 40 ~ + 85	
保存温度	T_{stg}	- 55 ~ + 125	

【注】 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。

17.2 電気的特性

17.2.1 DC 特性

DC 特性を表 17.2 に示します。また、出力許容電流値を表 17.3 に示します。

表 17.2 DC 特性 (1)

条件 : $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V^{*1}$ 、 $T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

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	ポート A、	VT^-	1.0			V	
	$P8_0 \sim P8_2$ 、	VT^+			$V_{CC} \times 0.7$	V	
	$PB_0 \sim PB_3$	$VT^+ - VT^-$	0.4			V	
入力"High"レベル 電圧	\overline{RES} 、 \overline{STBY} 、 NMI 、 MD_1 、 MD_0	V_{IH}	$V_{CC} - 0.7$		$V_{CC} + 0.3$	V	
	EXTAL		$V_{CC} \times 0.7$	$V_{CC} + 0.3$	V		
	ポート 7		2.0	$AV_{CC} + 0.3$	V		
	ポート 1、2、3、5、 6、9、 $P8_3$ 、 $PB_4 \sim PB_7$		2.0	$V_{CC} + 0.3$	V		
入力"Low"レベル 電圧	\overline{RES} 、 \overline{STBY} 、 MD_1 、 MD_0	V_{IL}	- 0.3		0.5	V	
	NMI 、EXTAL、 ポート 1、2、3、5、 6、7、9、 $P8_3$ 、 $PB_4 \sim PB_7$		- 0.3		0.8	V	
出力"High"レベル 電圧	全出力端子 (\overline{RESO} を除く)	V_{OH}	$V_{CC} - 0.5$			V	$I_{OH} = -200\mu A$
			3.5			V	$I_{OH} = -1mA$
出力"Low"レベル 電圧	全出力端子 (\overline{RESO} を除く)	V_{OL}			0.4	V	$I_{OL} = 1.6mA$
	ポート 1、2、5、B				1.0	V	$I_{OL} = 10mA$
	\overline{RESO}				0.4	V	$I_{OL} = 2.6mA$
入力リーク電流	\overline{STBY} 、 \overline{NMI} 、 \overline{RES} 、 MD_1 、 MD_0	$ I_{in} $			1.0	μA	$V_{in} =$ $0.5 \sim V_{CC} - 0.5V$
	ポート 7				1.0	μA	$V_{in} =$ $0.5 \sim AV_{CC} - 0.5V$

項目		記号	min	typ	max	単位	測定条件	
スリープ状態 リーク電流 (オフ状態)	ポート 1、2、3、5、 6、8~B	$ I_{TSL} $			1.0	μA	$V_{in} = 0.5 \sim V_{CC} - 0.5\text{V}$	
	$\overline{\text{RESO}}$				10.0	μA		
入力プルアップ MOS 電流	ポート 2、5	$-I_P$	50		300	μA	$V_{in} = 0\text{V}$	
入力容量	NMI、 $\overline{\text{RES}}$	C_{in}			50	pF	$V_{in} = 0\text{V}$ $f = 1\text{MHz}$ $T_a = 25$	
	NMI、 $\overline{\text{RES}}$ 以外の 全入力端子				15	pF		
消費電流*2	通常動作時	I_{CC}		50	70	mA	$f = 18\text{MHz}$	
	スリープ時			35	50		$f = 18\text{MHz}$	
	スタンバイ時*3				0.01	5.0	μA	$T_a = 50$
								20.0
アナログ電源電流	A/D 変換中	AI_{CC}		1.2	2.0	mA		
	A/D 変換待機時			0.01	5.0			μA
リファレンス 電源電流	A/D 変換中	AI_{CC}		0.5	0.8	mA	$V_{REF} = 5.0\text{V}$	
	A/D 変換待機時			0.01	5.0			μA
RAM スタンバイ電圧		V_{RAM}	2.0			V		

【注】 *1 A/D 変換器未使用時に 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、 $V_{IH \text{ min}} = V_{CC} - 0.5\text{V}$ 、 $V_{L \text{ max}} = 0.5\text{V}$ の条件下で、すべ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。

*3 V_{RAM} $V_{CC} < 4.5\text{V}$ のとき、 $V_{IH \text{ MIN}} = V_{CC} \times 0.9$ 、 $V_{IL \text{ MAX}} = 0.3\text{V}$ とした場合の値です。

17. 電気的特性 (暫定仕様)

表 17.2 DC 特性 (2)

条件 : $V_{CC} = 2.7 \sim 5.5V$ 、 $AV_{CC} = 2.7 \sim 5.5V$ 、 $V_{REF} = 2.7V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V^{*1}$ 、 $T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

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	ポート A、	V_T^-	$V_{CC} \times 0.2$			V	
	$P8_0 \sim P8_2$ 、	V_T^+			$V_{CC} \times 0.7$	V	
	$PB_0 \sim PB_3$	$V_T^+ - V_T^-$	$V_{CC} \times 0.04$			V	
入力"High" レベル電圧	\overline{RES} 、 \overline{STBY} 、 NMI 、 MD_1 、 MD_0	V_{IH}	$V_{CC} \times 0.9$		$V_{CC} + 0.3$	V	
	EXTAL		$V_{CC} \times 0.7$		$V_{CC} + 0.3$	V	
	ポート 7		$V_{CC} \times 0.7$		$AV_{CC} + 0.3$	V	
	ポート 1、2、3、5、 6、9、 $P8_3$ 、 $PB_4 \sim PB_7$		$V_{CC} \times 0.7$		$V_{CC} + 0.3$	V	
入力"Low" レベル電圧	\overline{RES} 、 \overline{STBY} 、 MD_1 、 MD_0	V_{IL}	- 0.3		$V_{CC} \times 0.1$	V	
	NMI 、EXTAL、 ポート 1、2、3、5、 6、7、9、 $P8_3$ 、 $PB_4 \sim PB_7$		- 0.3		$V_{CC} \times 0.2$	V	$V_{CC} < 4.0V$
					0.8	V	$V_{CC} = 4.0 \sim 5.5V$
出力"High" レベル電圧	全出力端子 ($\overline{RES0}$ は除く)	V_{OH}	$V_{CC} - 0.5$			V	$I_{OH} = -200\mu A$
			$V_{CC} - 1.0$			V	$I_{OH} = -1mA$
出力"Low" レベル電圧	全出力端子 ($\overline{RES0}$ を除く)	V_{OL}			0.4	V	$I_{OL} = 1.0mA$
					1.0	V	$V_{CC} = 4V$ $I_{OL} = 5mA$ $4V < V_{CC} < 5.5V$ $I_{OL} = 10mA$
	$\overline{RES0}$				0.4	V	$I_{OL} = 1.6mA$
入力リーク電流	\overline{STBY} 、 NMI 、 \overline{RES} 、 MD_1 、 MD_0	$ I_{in} $			1.0	μA	$V_{in} =$ $0.5 \sim V_{CC} - 0.5V$
	ポート 7				1.0	μA	$V_{in} =$ $0.5 \sim AV_{CC} - 0.5V$

項目		記号	min	typ	max	単位	測定条件
スリープ状態 リーク電流 (オフ状態)	ポート 1、2、3、5、 6、8、9、A、 B	$ I_{TS1} $			1.0	μA	$V_{in} = 0.5 \sim$ $V_{CC} - 0.5\text{V}$
	$\overline{\text{RESO}}$				10.0	μA	
入力プルアップ MOS 電流	ポート 2、5	$-I_P$	10		300	μA	$V_{CC} = 2.7\text{V} \sim$ 5.5V、 $V_{in} = 0\text{V}$
入力容量	NMI、 $\overline{\text{RES}}$	C_{in}			50	pF	$V_{in} = 0\text{V}$ f = 1MHz $T_a = 25$
	NMI、 $\overline{\text{RES}}$ 以外の 全入力端子				15	pF	
消費電流*2	通常動作時	I_{CC}^{*4}		12 (3.0V)	33.8 (5.5V)	mA	f = 8MHz
	スリープ時			8 (3.0V)	25.0 (5.5V)	mA	f = 8MHz
	スタンバイ時*3			0.01	5.0	μA	$T_a = 50$
						20.0	
アナログ電源電流	A/D 変換中	AI_{CC}		1.0	2.0	mA	$AV_{CC} = 3.0\text{V}$
				1.2			$AV_{CC} = 5.0\text{V}$
	A/D 変換待機時			0.01	5.0	μA	
リファレンス 電源電流	A/D 変換中	AI_{CC}		0.3	0.5	mA	$V_{REF} = 3.0\text{V}$
				0.5			$V_{REF} = 5.0\text{V}$
	A/D 変換待機時			0.01	5.0	μA	
RAM スタンバイ電圧		V_{RAM}	2.0			V	

【注】 *1 A/D 変換器未使用時に 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、 $V_{IH} \text{ min} = V_{CC} - 0.5\text{V}$ 、 $V_{IL} \text{ max} = 0.5\text{V}$ の条件下で、すべ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。

*3 V_{RAM} $V_{CC} < 2.7\text{V}$ のとき、 $V_{IH \text{ MIN}} = V_{CC} \times 0.9$ 、 $V_{IL \text{ MAX}} = 0.3\text{V}$ とした場合の値です。

*4 I_{CC} は下記の式に従って V_{CC} と f に依存します。

$$I_{CC} \text{ max.} = 3.0 \text{ (mA)} + 0.7 \text{ (mA/MHz} \cdot \text{V)} \times V_{CC} \times f \text{ (通常動作時)}$$

$$I_{CC} \text{ max.} = 3.0 \text{ (mA)} + 0.5 \text{ (mA/MHz} \cdot \text{V)} \times V_{CC} \times f \text{ (スリープ時)}$$

17. 電気的特性 (暫定仕様)

表 17.2 DC 特性 (3)

条件 : $V_{CC} = 3.0 \sim 5.5V$ 、 $AV_{CC} = 3.0 \sim 5.5V$ 、 $V_{REF} = 3.0V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V^{*1}$ 、 $T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

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	ポート A、	V_T^-	$V_{CC} \times 0.2$			V		
	$P8_0 \sim P8_2$ 、	V_T^+			$V_{CC} \times 0.7$	V		
	$PB_0 \sim PB_3$	$V_T^+ - V_T^-$	$V_{CC} \times 0.04$			V		
入力"High"レベル 電圧	\overline{RES} 、 \overline{STBY} 、 NMI 、 $MD_1 \sim MD_0$	V_{IH}	$V_{CC} \times 0.9$		$V_{CC} + 0.3$	V		
	EXTAL		$V_{CC} \times 0.7$		$V_{CC} + 0.3$	V		
	ポート 7		$V_{CC} \times 0.7$		$AV_{CC} + 0.3$	V		
	ポート 1、2、3、5、 6、9、 $P8_3$ 、 $PB_4 \sim PB_7$		$V_{CC} \times 0.7$		$V_{CC} + 0.3$	V		
入力"Low"レベル 電圧	\overline{RES} 、 \overline{STBY} 、 MD_1 、 MD_0	V_{IL}	- 0.3		$V_{CC} \times 0.1$	V		
	NMI 、EXTAL、 ポート 1、2、3、5、 6、7、9、 $P8_3$ 、 $PB_4 \sim PB_7$		- 0.3		$V_{CC} \times 0.2$	V		$V_{CC} < 4.0V$
					0.8	V		$V_{CC} = 4.0 \sim 5.5V$
出力"High"レベル 電圧	全出力端子 (\overline{RESO} を除く)	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 (\overline{RESO} を除く)	V_{OL}			0.4	V	$I_{OL} = 1.0mA$	
	ポート 1、2、5、B				1.0	V	$V_{CC} = 4V$ $I_{OL} = 5mA$ $4V < V_{CC} < 5.5V$ $I_{OL} = 10mA$	
	\overline{RESO}				0.4	V	$I_{OL} = 1.6mA$	
入力リーク電流	\overline{STBY} 、 NMI 、 \overline{RES} 、 MD_1 、 MD_0	$ I_{in} $			1.0	μA	$V_{in} =$ $0.5 \sim V_{CC} - 0.5V$	
	ポート 7				1.0	μA	$V_{in} =$ $0.5 \sim AV_{CC} - 0.5V$	
スリープステート リーク電流 (オフ状態)	ポート 1、2、3、5、 6、8 ~ B	$ I_{TSL} $			1.0	μA	$V_{in} = 0.5 \sim$ $V_{CC} - 0.5V$	
	\overline{RESO}				10.0	μA		

項目		記号	min	typ	max	単位	測定条件
入力プルアップ MOS 電流	ポート 2、5	$-I_p$	10		300	μA	$V_{CC} = 3.0\text{V} \sim 5.5\text{V}$ 、 $V_{in} = 0\text{V}$
入力容量	NMI、 $\overline{\text{RES}}$	C_{in}			50	pF	$V_{in} = 0\text{V}$
	NMI、 $\overline{\text{RES}}$ 以外の 全入力端子				15	pF	$f = 1\text{MHz}$ $T_a = 25$
消費電流*2	通常動作時	I_{CC}^{*4}		15 (3.0V)	41.5 (5.5V)	mA	$f = 10\text{MHz}$
	スリープ時			10 (3.0V)	30.5 (5.5V)	mA	$f = 10\text{MHz}$
	スタンバイ時*3			0.01	5.0	μA	$T_a = 50$ $50 < T_a$
アナログ電源電流	A/D 変換中	I_{CC}		1.0	2.0	mA	$AV_{CC} = 3.0\text{V}$
				1.2			$AV_{CC} = 5.0\text{V}$
	A/D 変換待機時			0.01	5.0	μA	
リファレンス 電源電流	A/D 変換中	I_{CC}		0.3	0.5	mA	$V_{REF} = 3.0\text{V}$
				0.5			$V_{REF} = 5.0\text{V}$
	A/D 変換待機時			0.01	5.0	μA	
RAM スタンバイ電圧		V_{RAM}	2.0			V	

【注】 *1 A/D 変換器未使用時に 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、 $V_{IH\ min} = V_{CC} - 0.5\text{V}$ 、 $V_{L\ max} = 0.5\text{V}$ の条件下で、すべ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。

*3 V_{RAM} $V_{CC} < 3.0\text{V}$ のとき、 $V_{IH\ MIN} = V_{CC} \times 0.9$ 、 $V_{L\ MAX} = 0.3\text{V}$ とした場合の値です。

*4 I_{CC} は下記の式に従って V_{CC} と f に依存します。

$$I_{CC\ max.} = 3.0\ (\text{mA}) + 0.7\ (\text{mA/MHz} \cdot \text{V}) \times V_{CC} \times f\ (\text{通常動作時})$$

$$I_{CC\ max.} = 3.0\ (\text{mA}) + 0.5\ (\text{mA/MHz} \cdot \text{V}) \times V_{CC} \times f\ (\text{スリープ時})$$

表 17.3 出力許容電流値 (1)

条件 : $V_{CC} = 2.7 \sim 5.5V$ 、 $AV_{CC} = 2.7 \sim 5.5V$ 、 $V_{REF} = 2.7V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

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、B	I_{OL}			10	mA
	上記以外の出力端子				2.0	mA
出力"Low"レベル許容電流 (総和)	ポート 1、2、5、B 28端子の総和	I_{OL}			80	mA
	ポート 8、9、A、B 23端子の総和				75 ^{*2} / 65 ^{*1}	mA
	上記を含む、全出力端子の総和				120	mA
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】 LSIの信頼性を確保するため、出力電流値は表 17.3の値を超えないようにしてください。
 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 17.1、図 17.2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。

*1 条件が $V_{CC} = 2.7 \sim 5.5V$ 、 $AV_{CC} = 2.7 \sim 5.5V$ 、 $V_{REF} = 2.7V \sim AV_{CC}$ の場合。

*2 条件が $V_{CC} = 4.5 \sim 5.5V$ 、 $AV_{CC} = 4.5 \sim 5.5V$ 、 $V_{REF} = 4.5V \sim AV_{CC}$ の場合。

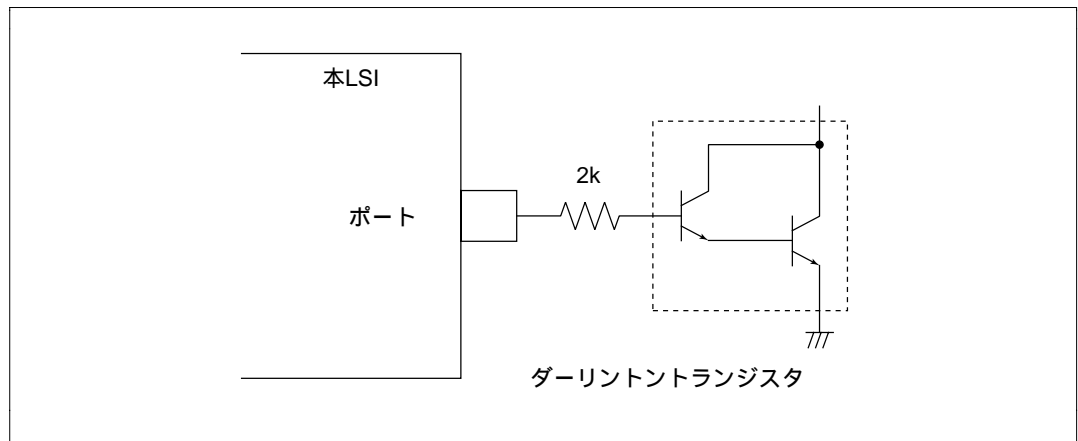


図 17.1 ダーリントトランジスタ駆動回路例

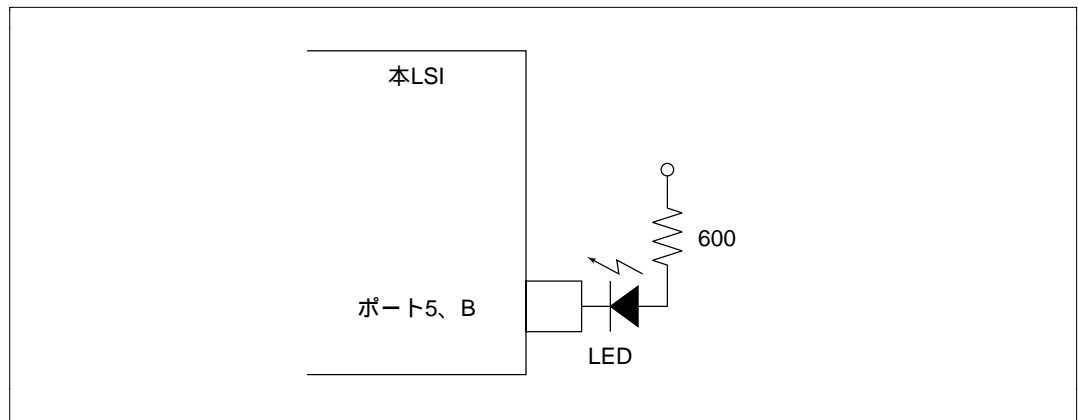


図 17.2 LED 駆動回路例

17.2.2 AC 特性

表 17.4 にバスタイミング、表 17.5 に制御信号タイミング、表 17.6 に内蔵周辺モジュールタイミングを示します。

17. 電気的特性 (暫定仕様)

表 17.4 バスタイミング (1)

条件 A : $V_{CC} = 2.7 \sim 5.5V$ 、 $AV_{CC} = 2.7 \sim 5.5V$ 、 $V_{REF} = 2.7V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、 $f = 2 \sim 8MHz$

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

条件 B : $V_{CC} = 3.0 \sim 5.5V$ 、 $AV_{CC} = 3.0 \sim 5.5V$ 、 $V_{REF} = 3.0V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、 $f = 2 \sim 10MHz$

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

条件 C : $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、 $f = 2 \sim 18MHz$

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

項目	記号	条件 A		条件 B		条件 C		測定条件
		8MHz		10MHz		18MHz		
		min	max	min	max	min	max	
クロックサイクル時間	t_{cyc}	125	500	100	500	55.5	500	図 17.4、 図 17.5
クロックパルス幅 "Low"レベル時間	t_{CL}	40		30		17		
クロックパルス幅 "High"レベル時間	t_{CH}	40		30		17		
クロック立上がり時間	t_{CR}		20		15		10	
クロック立下がり時間	t_{CF}		20		15		10	
アドレス遅延時間	t_{AD}		60		50		25	
アドレスホールド時間	t_{AH}	25		20		10		
アドレスストロープ遅延時間	t_{ASD}		60		40		25	
ライトストロープ遅延時間	t_{WSD}		60		50		25	
ストロープ遅延時間	t_{SD}		60		50		25	
ライトデータストロープパルス幅 1	t_{WSW1}^*	85		60		32		
ライトデータストロープパルス幅 2	t_{WSW2}^*	150		110		62		
アドレスセットアップ時間 1	t_{AS1}	20		15		10		
アドレスセットアップ時間 2	t_{AS2}	80		65		38		
リードデータセットアップ時間	t_{RDS}	50		35		15		
リードデータホールド時間	t_{RDH}	0		0		0		
ライトデータ遅延時間	t_{WDD}		75		75		55	
ライトデータセットアップ時間 1	t_{WDS1}	60		40		10		
ライトデータセットアップ時間 2	t_{WDS2}	5		-10		-10		
ライトデータホールド時間	t_{WDH}	25		20		20		
リードデータアクセス時間 1	t_{ACC1}^*		120		100		50	

項目	記号	条件 A		条件 B		条件 C		測定条件
		8MHz		10MHz		18MHz		
		min	max	min	max	min	max	
リードデータアクセス時間 2	t_{ACC2}^*		240		200		105	図 17.4、 図 17.5
リードデータアクセス時間 3	t_{ACC3}^*		70		50		20	
リードデータアクセス時間 4	t_{ACC4}^*		180		150		80	
プリチャージ時間	t_{PCH}^*	85		60		40		図 17.6
ウェイトセットアップ時間	t_{WTS}	40		40		25		
ウェイトセットホールド時間	t_{WTH}	10		10		5		

単位：ns

【注】 * 8MHz のとき、下記の時間は以下に示すようにクロックサイクル時間に依存します。

$$t_{ACC1} = 1.5 \times t_{cyc} - 68 \text{ (ns)} \quad t_{WSW1} = 1.0 \times t_{cyc} - 40 \text{ (ns)}$$

$$t_{ACC2} = 2.5 \times t_{cyc} - 73 \text{ (ns)} \quad t_{WSW2} = 1.5 \times t_{cyc} - 38 \text{ (ns)}$$

$$t_{ACC3} = 1.0 \times t_{cyc} - 55 \text{ (ns)} \quad t_{PCH} = 1.0 \times t_{cyc} - 40 \text{ (ns)}$$

$$t_{ACC4} = 2.0 \times t_{cyc} - 70 \text{ (ns)}$$

10MHz のとき、下記の時間は以下に示すようにクロックサイクル時間に依存します。

$$t_{ACC1} = 1.5 \times t_{cyc} - 50 \text{ (ns)} \quad t_{WSW1} = 1.0 \times t_{cyc} - 40 \text{ (ns)}$$

$$t_{ACC2} = 2.5 \times t_{cyc} - 50 \text{ (ns)} \quad t_{WSW2} = 1.5 \times t_{cyc} - 40 \text{ (ns)}$$

$$t_{ACC3} = 1.0 \times t_{cyc} - 50 \text{ (ns)} \quad t_{PCH} = 1.0 \times t_{cyc} - 40 \text{ (ns)}$$

$$t_{ACC4} = 2.0 \times t_{cyc} - 50 \text{ (ns)}$$

18MHz のとき、下記の時間は以下に示すようにクロックサイクル時間に依存します。

$$t_{ACC1} = 1.5 \times t_{cyc} - 34 \text{ (ns)} \quad t_{WSW1} = 1.0 \times t_{cyc} - 24 \text{ (ns)}$$

$$t_{ACC2} = 2.5 \times t_{cyc} - 34 \text{ (ns)} \quad t_{WSW2} = 1.5 \times t_{cyc} - 22 \text{ (ns)}$$

$$t_{ACC3} = 1.0 \times t_{cyc} - 36 \text{ (ns)} \quad t_{PCH} = 1.0 \times t_{cyc} - 21 \text{ (ns)}$$

$$t_{ACC4} = 2.0 \times t_{cyc} - 31 \text{ (ns)}$$

表 17.5 制御信号タイミング

条件 A : $V_{CC} = 2.7 \sim 5.5V$ 、 $AV_{CC} = 2.7 \sim 5.5V$ 、 $V_{REF} = 2.7V \sim AV_{CC}$ 、 $V_{SS} = 0V$ 、 $f = 2 \sim 8MHz$

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

条件 B : $V_{CC} = 3.0 \sim 5.5V$ 、 $AV_{CC} = 3.0 \sim 5.5V$ 、 $V_{REF} = 3.0V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、

$f = 2 \sim 10MHz$

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

条件 C : $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、

$f = 2 \sim 18MHz$

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

項目	記号	条件 A		条件 B		条件 C		単位	測定条件
		8MHz		10MHz		18MHz			
		min	max	min	max	min	max		
RES セットアップ時間	t_{RESS}	200		200		200		ns	図 17.7
RES パルス幅	t_{RESW}	10		10		10		t_{cyc}	
\overline{RESO} 出力遅延時間	t_{RESL}		100		100		100	ns	図 17.8
\overline{RESO} 出力パルス幅	t_{RESOW}	132		132		132		t_{cyc}	
NMI セットアップ時間 (NMI、 $\overline{IRQ_4} \sim \overline{IRQ_0}$)	t_{NMIS}	200		200		150		ns	図 17.9
NMI ホールド時間 (NMI、 $\overline{IRQ_4} \sim \overline{IRQ_0}$)	t_{NMIH}	10		10		10		ns	
割込みパルス幅 (NMI、 $\overline{IRQ_4} \sim \overline{IRQ_0}$ ソフトウェアスタンバイモードからの復帰時)	t_{NMIW}	200		200		200		ns	
リセット発振安定時間 (水晶)	t_{OSC1}	20		20		20		ms	図 17.10
ソフトウェアスタンバイ 発振安定時間 (水晶)	t_{OSC2}	8		8		7		ms	図 16.1

表 17.6 内蔵周辺モジュールタイミング

条件 A : $V_{CC} = 2.7 \sim 5.5V$ 、 $AV_{CC} = 2.7 \sim 5.5V$ 、 $V_{REF} = 2.7V \sim AV_{CC}$ 、 $V_{SS} = 0V$ 、 $f = 2 \sim 8MHz$

$T_a = -20 \sim +75$ （通常仕様品）、 $T_a = -40 \sim +85$ （広温度範囲仕様品）

条件 B : $V_{CC} = 3.0 \sim 5.5V$ 、 $AV_{CC} = 3.0 \sim 5.5V$ 、 $V_{REF} = 3.0V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、 $f = 2 \sim 10MHz$

$T_a = -20 \sim +75$ （通常仕様品）、 $T_a = -40 \sim +85$ （広温度範囲仕様品）

条件 C : $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、 $f = 2 \sim 18MHz$

$T_a = -20 \sim +75$ （通常仕様品）、 $T_a = -40 \sim +85$ （広温度範囲仕様品）

モジュール	項目	記号	条件 A		条件 B		条件 C		単位	測定条件
			8MHz		10MHz		18MHz			
			min	max	min	max	min	max		
ITU	タイマ出力遅延時間	t_{TOCD}		100		100		100	ns	図 17.12
	タイマ入力セットアップ時間	t_{TICS}	50		50		50			
	タイマクロック入力 セットアップ時間	t_{TCKS}	50		50		50			t _{cyc}
	タイマクロック パルス幅	単エッジ指定 t_{TCKWH}	1.5		1.5		1.5			
		両エッジ指定 t_{TCKWL}	2.5		2.5		2.5			
SCI	入力クロック サイクル	調歩同期	t_{SCYC}	4		4		4	t _{scyc}	図 17.14
		クロック同期		6		6		6		
	入力クロック立上がり時間	t_{SCKr}		1.5		1.5		1.5		
	入力クロック立下がり時間	t_{SCKf}		1.5		1.5		1.5		
	入力クロックパルス幅	t_{SCKW}	0.4	0.6	0.4	0.6	0.4	0.6		
	送信データ遅延時間	t_{TXD}		100		100		100	ns	図 17.15
	受信データセットアップ時間 (クロック同期)	t_{RXS}	100		100		100			
	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		100			
	クロック出力		0		0		0			
ポート	出力データ遅延時間	t_{PWD}		100		100		100	ns	図 17.11
TPC	入力データセットアップ時間	t_{PRS}	50		50		50			
	入力データホールド時間	t_{PRH}	50		50		50			

17. 電気的特性 (暫定仕様)

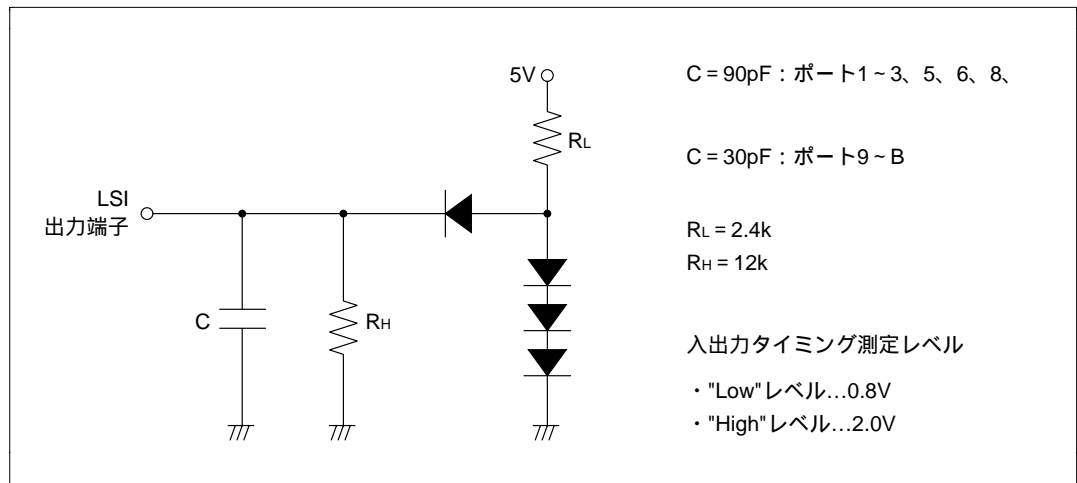


図 17.3 出力負荷回路

17.2.3 A/D 変換特性

A/D 変換特性を表 17.7 に示します。

表 17.7 A/D 変換特性

条件 A : $V_{CC} = 2.7 \sim 5.5V$ 、 $AV_{CC} = 2.7 \sim 5.5V$ 、 $V_{REF} = 2.7V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、
= 2 ~ 8MHz

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

条件 B : $V_{CC} = 3.0 \sim 5.5V$ 、 $AV_{CC} = 3.0 \sim 5.5V$ 、 $V_{REF} = 3.0V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、
= 2 ~ 10MHz

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

条件 C : $V_{CC} = 5.0V \pm 10\%$ 、 $AV_{CC} = 5.0V \pm 10\%$ 、 $V_{REF} = 4.5V \sim AV_{CC}$ 、 $V_{SS} = AV_{SS} = 0V$ 、
= 2 ~ 18MHz

$T_a = -20 \sim +75$ (通常仕様品)、 $T_a = -40 \sim +85$ (広温度範囲仕様品)

項目	条件 A			条件 B			条件 C			単位
	8MHz			10MHz			18MHz			
	min	typ	max	min	typ	max	min	typ	max	
分解能	10	10	10	10	10	10	10	10	10	ビット
変換時間			16.8			13.4			7.5	μ S
アナログ入力容量			20			20			20	pF
許容信号源			10^{*1}			10^{*1}			10^{*4}	k
インピーダンス			5^{*2}			5^{*3}			5^{*5}	
非直線性誤差			± 6.0			± 6.0			± 3.5	LSB
オフセット誤差			± 4.0			± 4.0			± 3.5	LSB
フルスケール誤差			± 4.0			± 4.0			± 3.5	LSB
量子化誤差			± 0.5			± 0.5			± 0.5	LSB
絶対精度			± 8.0			± 8.0			± 4.0	LSB

【注】 *1 4.0 AV_{CC} 5.5 の場合です。

*2 2.7 $AV_{CC} < 4.0$ の場合です。

*3 3.0 $AV_{CC} < 4.0$ の場合です。

*4 12MHz の場合です。

*5 > 12MHz の場合です。

17.3 動作タイミング

動作タイミングを以下に示します。

17.3.1 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 17.4 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 17.5 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 17.6 に外部 2 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。

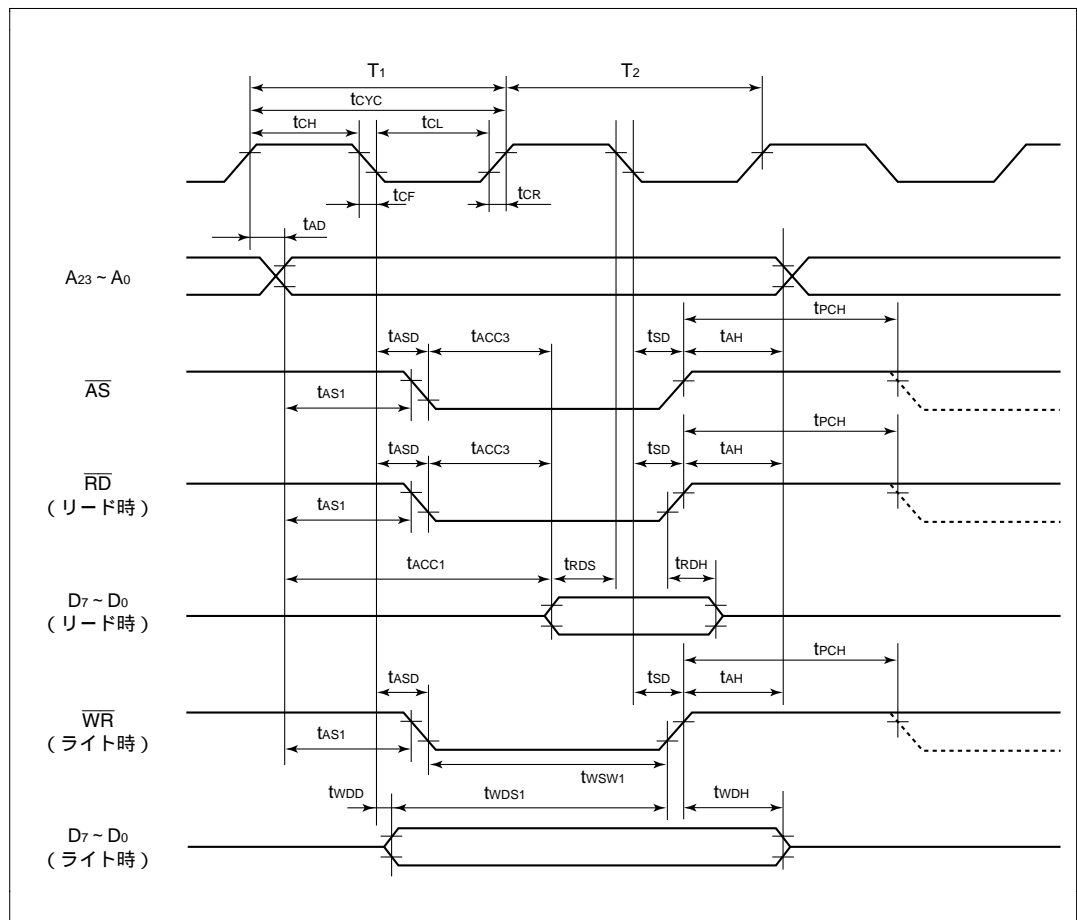


図 17.4 基本バスタイミング / 2 ステートアクセス

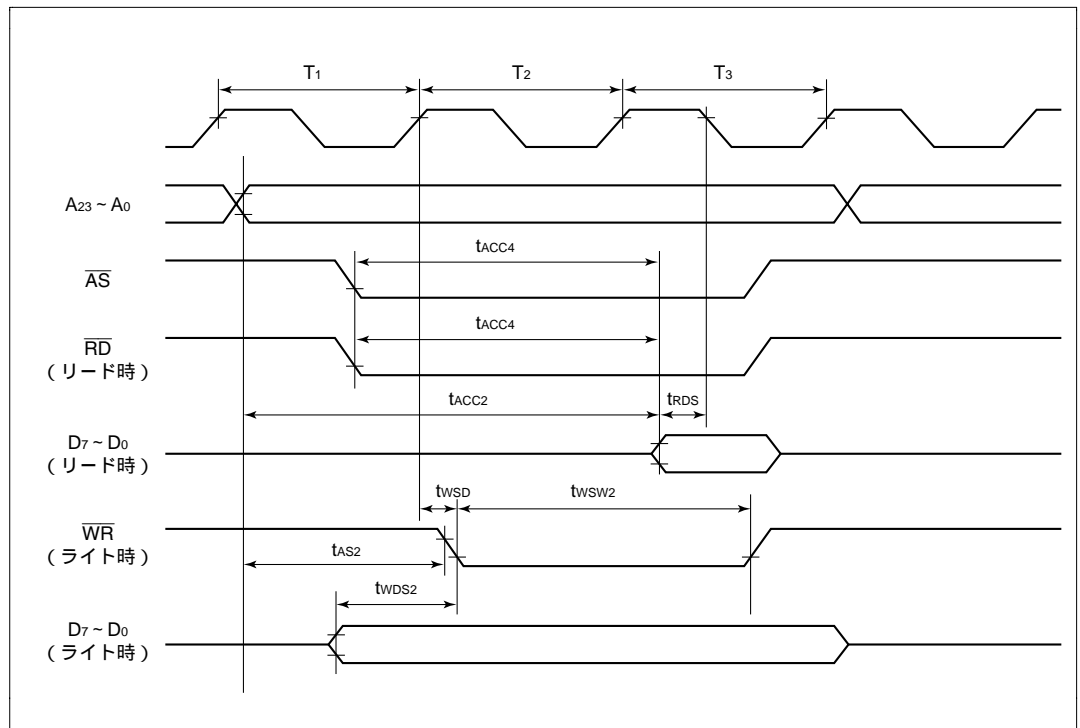


図 17.5 基本バスタイミング / 3 ステートアクセス

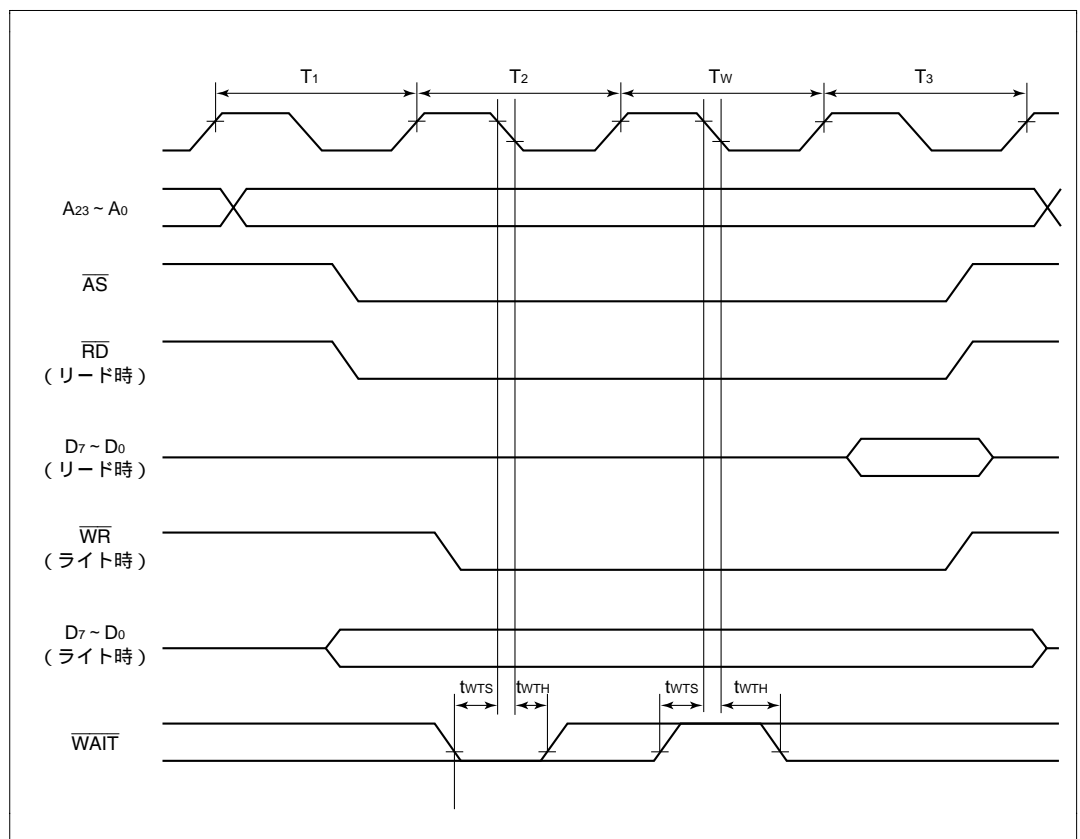


図 17.6 基本バスタイミング / 3 ステートアクセス 1 ウェイト

17.3.2 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 17.7 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 17.8 にリセット出力タイミングを示します。

(3) 割込み入力タイミング

図 17.9 に NMI、 $\overline{IRQ}_4 \sim \overline{IRQ}_0$ 割込み入力タイミングを示します。

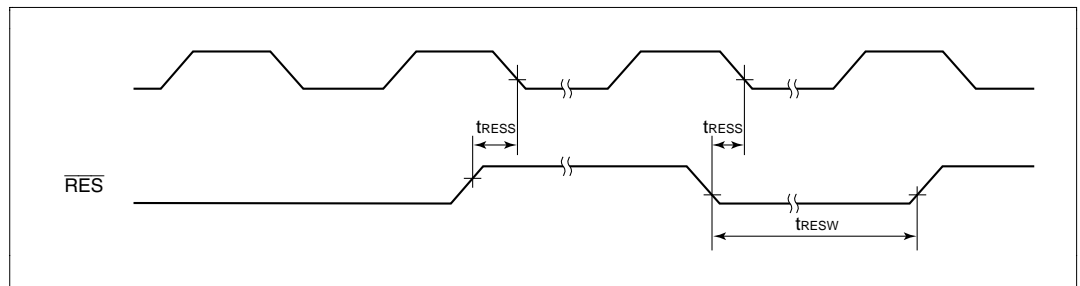


図 17.7 リセット入力タイミング

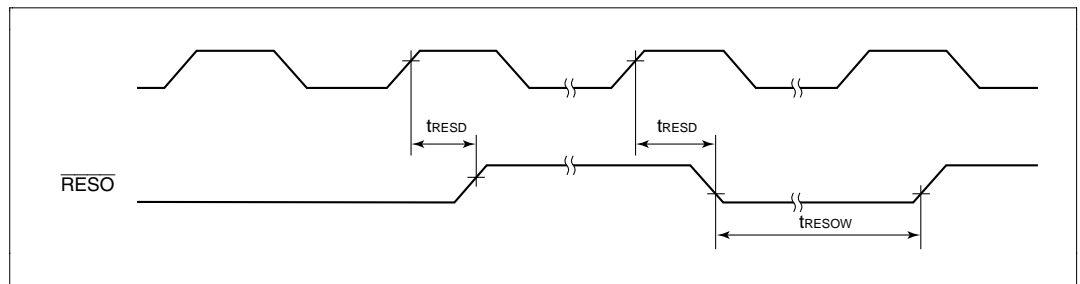


図 17.8 リセット出力タイミング

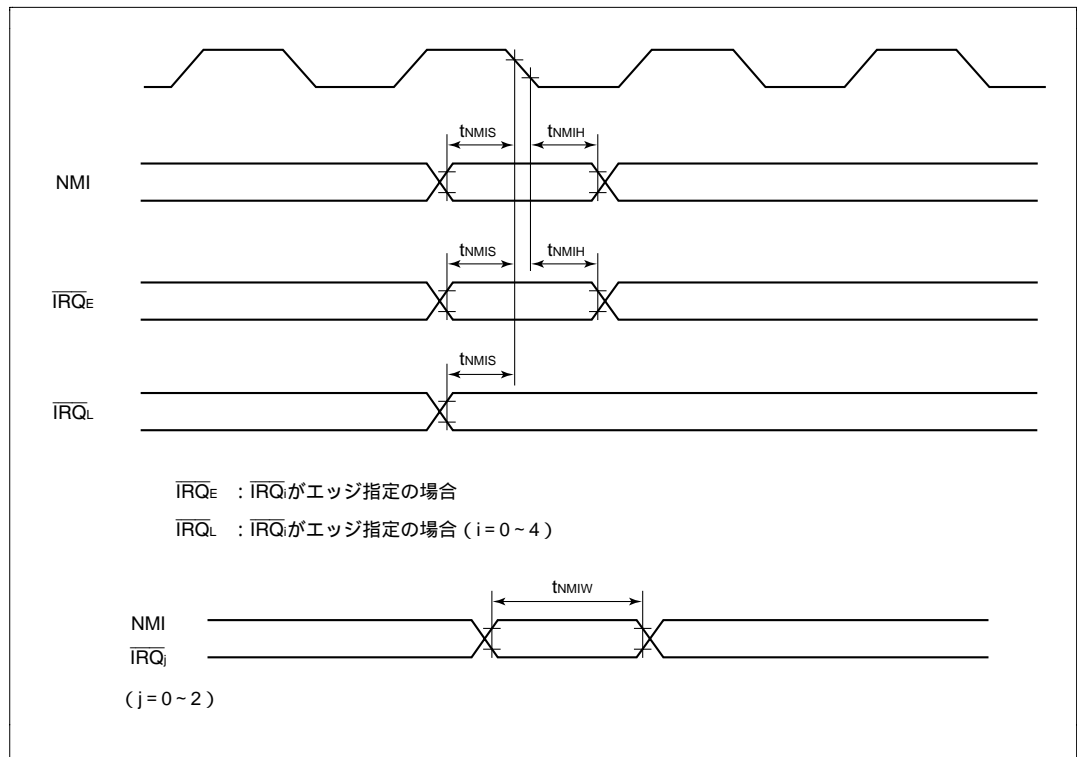


図 17.9 割込み入力タイミング

17.3.3 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 17.10 に発振安定時間タイミングを示します。

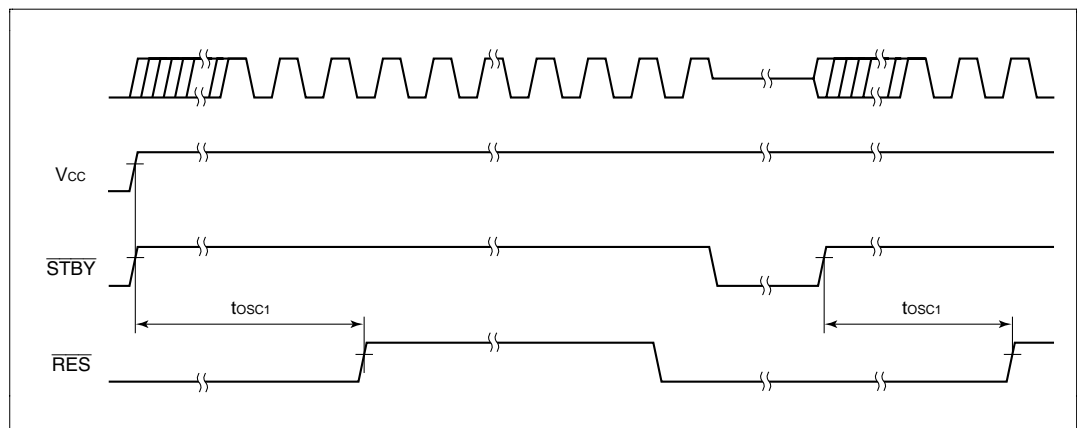


図 17.10 発振安定時間タイミング

17.3.4 TPC、I/Oポートタイミング

TPC、I/Oポートの入出力タイミングを以下に示します。

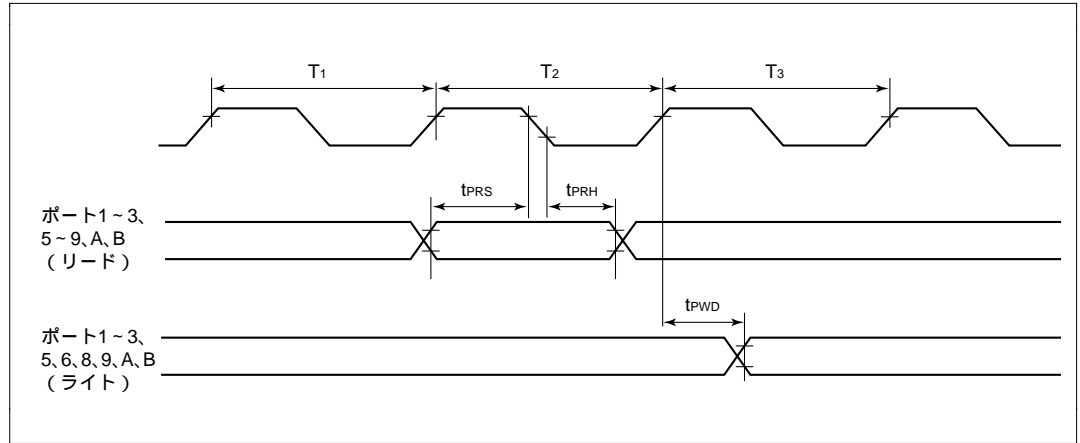


図 17.11 TPC、I/Oポート入出力タイミング

17.3.5 ITU タイミング

ITU の各タイミングを以下に示します。

(1) ITU 入出力タイミング

図 17.12 に ITU 入出力タイミングを示します。

(2) ITU 外部クロック入力タイミング

図 17.13 に ITU 外部クロック入力タイミングを示します。

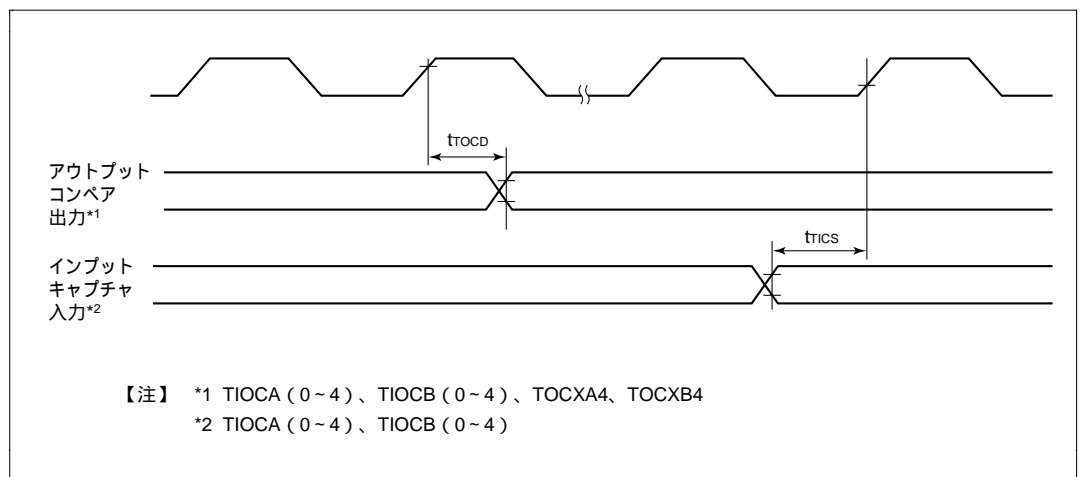


図 17.12 ITU 入出力タイミング

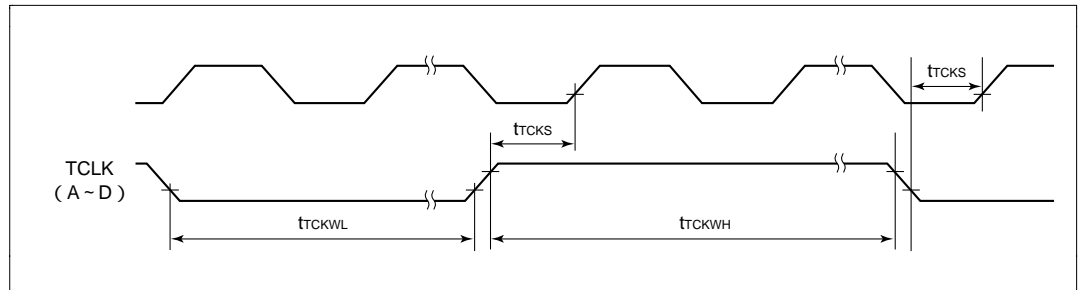


図 17.13 ITU 外部クロック入力タイミング

17.3.6 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 17.14 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 17.15 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。

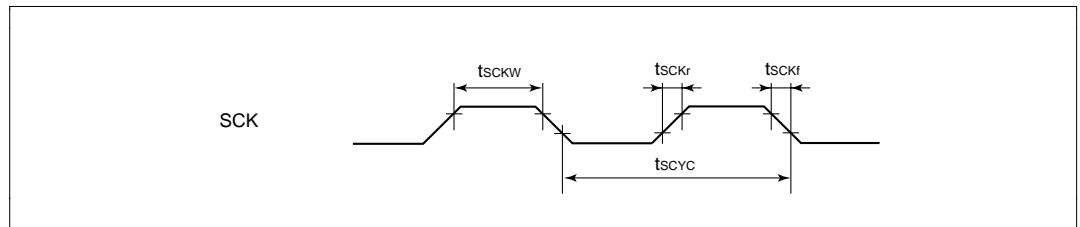


図 17.14 SCK 入力クロックタイミング

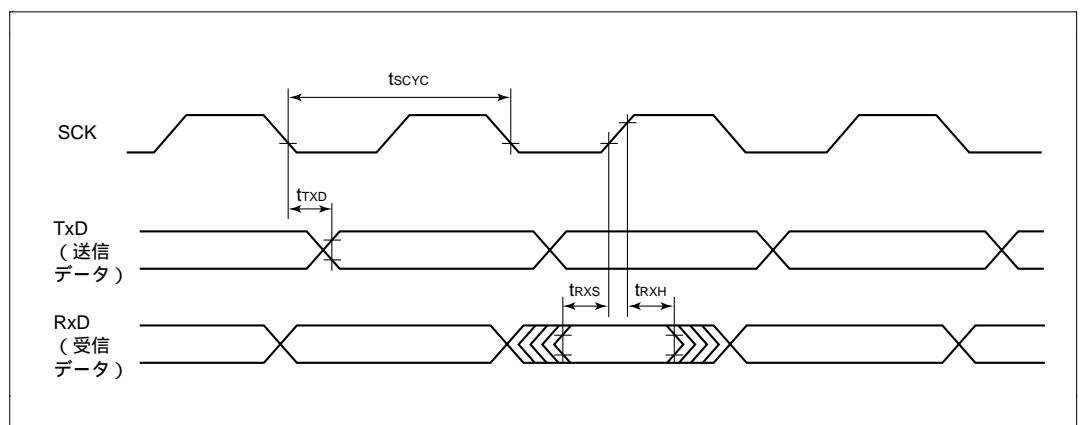


図 17.15 クロック同期式モード時の SCI 入出力タイミング

付録

付録 目次

A.	命令	463
	A.1	命令一覧..... 463
	A.2	オペレーションコードマップ..... 478
	A.3	命令実行ステート数..... 481
B.	内部 I/O レジスタ一覧	492
	B.1	アドレス一覧..... 492
	B.2	機能一覧..... 499
C.	I/O ポートブロック図	553
	C.1	ポート 1 ブロック図..... 553
	C.2	ポート 2 ブロック図..... 554
	C.3	ポート 3 ブロック図..... 555
	C.4	ポート 5 ブロック図..... 556
	C.5	ポート 6 ブロック図..... 557
	C.6	ポート 7 ブロック図..... 559
	C.7	ポート 8 ブロック図..... 560
	C.8	ポート 9 ブロック図..... 562
	C.9	ポート A ブロック図..... 565
	C.10	ポート B ブロック図..... 568
D.	端子状態	572
	D.1	各処理状態におけるポートの状態..... 572
	D.2	リセット時の端子状態..... 574
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	577
F.	ROM 発注手順	578
	F.1	ROM 書き換え品開発の流れ（発注手順）..... 578
	F.2	ROM 発注時の注意事項..... 579
G.	型名一覧	580
H.	外形寸法図	581

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(オーバーフロー)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)

MOV	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						実行ステップ数*					
			#xx	Rn	@ERn	@(d, ERn)		@-ERn/@ERn+	@aa	@(d, PC)	@@aa	—	I	H	N	Z	V	C	ノーマル ³⁾
	MOV.B #xx:8, Rd	B	2											↑	↑	0	—	2	
	MOV.B Rs, Rd	B	2	2										↑	↑	0	—	2	
	MOV.B @ERS, Rd	B		2										↑	↑	0	—	4	
	MOV.B @(d:16, ERS), Rd	B			4									↑	↑	0	—	6	
	MOV.B @(d:24, ERS), Rd	B			8									↑	↑	0	—	10	
	MOV.B @ERS+, Rd	B				2								↑	↑	0	—	6	
	MOV.B @aa:8, Rd	B							2					↑	↑	0	—	4	
	MOV.B @aa:16, Rd	B							4					↑	↑	0	—	6	
	MOV.B @aa:24, Rd	B							6					↑	↑	0	—	8	
	MOV.B Rs, @ERd	B		2										↑	↑	0	—	4	
	MOV.B Rs, @(d:16, ERd)	B			4									↑	↑	0	—	6	
	MOV.B Rs, @(d:24, ERd)	B			8									↑	↑	0	—	10	
	MOV.B Rs, @-ERd	B				2								↑	↑	0	—	6	
	MOV.B Rs, @aa:8	B							2					↑	↑	0	—	4	
	MOV.B Rs, @aa:16	B							4					↑	↑	0	—	6	
	MOV.B Rs, @aa:24	B							6					↑	↑	0	—	8	
	MOV.W #xx:16, Rd	W	4											↑	↑	0	—	4	
	MOV.W Rs, Rd	W		2										↑	↑	0	—	2	
	MOV.W @ERS, Rd	W			2									↑	↑	0	—	4	
	MOV.W @(d:16, ERS), Rd	W				4								↑	↑	0	—	6	
	MOV.W @(d:24, ERS), Rd	W				8								↑	↑	0	—	10	
	MOV.W @ERS+, Rd	W				2								↑	↑	0	—	6	
	MOV.W @aa:16, Rd	W							4					↑	↑	0	—	6	
	MOV.W @aa:24, Rd	W							6					↑	↑	0	—	8	

表A.1 命令セット一覧(2)

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							実行ステート数*1 ノーマル ³⁾ アドバンスト
		Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa	@(d, PC)		@aa	I	H	N	Z	V	C	
MOV	W		2						Rs16 @ERd	—	—	↑	↑	0	—	4
	W			4					Rs16 @(d:16, ERd)	—	—	↑	↑	0	—	6
	W								Rs16 @(d:24, ERd)	—	—	↑	↑	0	—	8
	W				2				ERd32-2 ERd32, Rs16 @ERd	—	—	↑	↑	0	—	6
	W					4			Rs16 @aa:16	—	—	↑	↑	0	—	6
	W					6			Rs16 @aa:24	—	—	↑	↑	0	—	8
	L	6							#xx:32 Rd32	—	—	↑	↑	0	—	6
	L	2							ERS32 ERd32	—	—	↑	↑	0	—	2
	L	4							@ERS ERd32	—	—	↑	↑	0	—	8
	L	6							@(d:16, ERS) ERd32	—	—	↑	↑	0	—	10
	L	10							@(d:24, ERS) ERd32	—	—	↑	↑	0	—	14
	L	4							@ERS ERd32, ERS32+4 ERS32	—	—	↑	↑	0	—	10
	L	6				6			@aa:16 ERd32	—	—	↑	↑	0	—	10
	L	8				8			@aa:24 ERd32	—	—	↑	↑	0	—	12
	L	4							ERS32 @ERd	—	—	↑	↑	0	—	8
	L	6							ERS32 @(d:16, ERd)	—	—	↑	↑	0	—	10
	L	10							ERS32 @(d:24, ERd)	—	—	↑	↑	0	—	14
	L	4							ERd32-4 ERd32, ERS32 @ERd	—	—	↑	↑	0	—	10
	L	6				6			ERS32 @aa:16	—	—	↑	↑	0	—	10
	L	8				8			ERS32 @aa:24	—	—	↑	↑	0	—	12
POP	W								@SP Rn16, SP+2 SP	—	—	↑	↑	0	—	6
	L								@SP ERn32, SP+4 SP	—	—	↑	↑	0	—	10
PUSH	W								SP-2 SP, Rn16 @SP	—	—	↑	↑	0	—	6
	L								SP-4 SP, ERn32 @SP	—	—	↑	↑	0	—	10
MOVFP	B					4			本LSIでは使用できません							
MOVTP	B					4			本LSIでは使用できません							

表A.1 命令セット一覧(3)

アーキテクチャ	命令	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							実行ステップ数*			
			Rn	@ERn	@(d, ERn)	@-ERn/@ERn+ @aa @ (d, PC)		@@aa	I	H	N	Z	V	C	ノーマル ³⁾	アドバンスト		
ニーモニック	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	
ADDS	ADDS.L #1, ERd	L	2									—	—	—	—	—	2	
	ADDS.L #2, ERd	L	2									—	—	—	—	—	2	
	ADDS.L #4, ERd	L	2									—	—	—	—	—	2	
	INC.B Rd	B	2									—	—	—	—	—	2	
INC	INC.W #1, Rd	W	2									—	—	—	—	—	2	
	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	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	

表 A.1 命令セット一覧 (4)

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							実行ステータス*	
		Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa	@(d, PC)		@aa	I	H	N	Z	V	C	ノーマル ⁸	アドバンスト
SUBS	SUBS.L #1, ERd	L	2						ERd32-1	ERd32	—	—	—	—	—	—	2
	SUBS.L #2, ERd	L	2						ERd32-2	ERd32	—	—	—	—	—	—	2
	SUBS.L #4, ERd	L	2						ERd32-4	ERd32	—	—	—	—	—	—	2
DEC	DEC.B Rd	B	2						Rd8-1	Rd8	—	↑	↑	—	—	—	2
	DEC.W #1, Rd	W	2						Rd16-1	Rd16	—	↑	↑	—	—	—	2
	DEC.W #2, Rd	W	2						Rd16-2	Rd16	—	↑	↑	—	—	—	2
	DEC.L #1, ERd	L	2						ERd32-1	ERd32	—	↑	↑	—	—	—	2
DAS	DAS.Rd	B	2						Rd8 10進補正	Rd8	—	*	↑	—	—	—	2
	MULXU.B Rs, Rd	B	2						Rd8 x Rs8	Rd16 (符号なし乗算)	—	—	—	—	—	—	14
MULXS	MULXU.W Rs, ERd	W	2						Rd16 x Rs16	ERd32 (符号なし乗算)	—	—	—	—	—	—	22
	MULXS.B Rs, Rd	B	4						Rd8 x Rs8	Rd16 (符号付乗算)	—	↑	↑	—	—	—	16
	MULXS.W Rs, ERd	W	4						Rd16 x Rs16	ERd32 (符号付乗算)	—	↑	↑	—	—	—	24
DIVXU	DIVXU.B Rs, Rd	B	2						Rd16 ÷ Rs8	Rd16 (RdH:余り, RdL:商) (符号なし除算)	—	—	(6)	(7)	—	—	14
	DIVXU.W Rs, ERd	W	2						ERd32 ÷ Rs16	ERd32 (Ed:余り, Rd:商) (符号なし除算)	—	—	(6)	(7)	—	—	22
DIVXS	DIVXS.B Rs, Rd	B	4						Rd16 ÷ Rs8	Rd16 (RdH:余り, RdL:商) (符号付除算)	—	—	(8)	(7)	—	—	16
	DIVXS.W Rs, ERd	W	4						ERd32 ÷ Rs16	ERd32 (Ed:余り, Rd:商) (符号付除算)	—	—	(8)	(7)	—	—	24
CMP	CMP.B #xx:8, Rd	B	2						Rd8-#xx:8		—	↑	↑	↑	↑	↑	2
	CMP.B Rs, Rd	B	2						Rd8-Rs8		—	↑	↑	↑	↑	↑	2
	CMP.W #xx:16, Rd	W	4						Rd16-#xx:16		—	(1)	↑	↑	↑	↑	4
	CMP.W Rs, Rd	W	2						Rd16-Rs16		—	(1)	↑	↑	↑	↑	2

表A.1 命令セット一覧(5)

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	Rn	@ERn	@(d, ERn)			@-ERn/@ERn+	@aa	@(d, PC)	@@aa	—	I	H	N	Z	V	C	ノーマル ³⁾
CMP	CMP.L #xx:32, ERd	L	6										(2)	↑	↑	↑	↑	4	
	CMP.L ERs, ERd	L	2										(2)	↑	↑	↑	↑	2	
NEG	NEG.B Rd	B	2										↑	↑	↑	↑	↑	2	
	NEG.W Rd	W	2										↑	↑	↑	↑	↑	2	
	NEG.L ERd	L	2										↑	↑	↑	↑	↑	2	
EXTU	EXTU.W Rd	W	2										—	0	↑	0	—	2	
	EXTU.L ERd	L	2										—	0	↑	0	—	2	
EXTS	EXTS.W Rd	W	2										—	↑	↑	↑	↑	2	
	EXTS.L ERd	L	2										—	↑	↑	↑	↑	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							実行スタート数*1				
		#xx	Rn	@ERn	@(d, ERn)			@-ERn/@ERn+	@aa	@(d, PC)	@@aa	—	I	H	N	Z	V	C	ノーマル ³⁾
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	
OR	AND.L ERs, ERd	L	4										—	↑	↑	0	—	4	
	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	

表A.1 命令セット一覧(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	コンディションコード							実行スタート数*	
		Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa	@(d, PC)		@aa	I	H	N	Z	V	C	ノーマル*	アドバンスト
OR	OR.L #xx:32, ERd	L	6						ERd32 #xx:32 ERd32	—	—	↑	↑	0	—	6	
	OR.L ERs, ERd	L	4						ERd32 ERs32 ERd32	—	—	↑	↑	0	—	4	
XOR	XOR.B #xx:8, Rd	B	2						Rd8 ⊕ #xx:8 Rd8	—	—	↑	↑	0	—	2	
	XOR.B Rs, Rd	B	2						Rd8 ⊕ Rs8 Rd8	—	—	↑	↑	0	—	2	
	XOR.W #xx:16, Rd	W	4						Rd16 ⊕ #xx:16 Rd16	—	—	↑	↑	0	—	4	
	XOR.W Rs, Rd	W	2						Rd16 ⊕ Rs16 Rd16	—	—	↑	↑	0	—	2	
NOT	XOR.L #xx:32, ERd	L	6						ERd32 ⊕ #xx:32 ERd32	—	—	↑	↑	0	—	6	
	XOR.L ERs, ERd	L	4						ERd32 ⊕ ERs32 ERd32	—	—	↑	↑	0	—	4	
	NOT.B Rd	B	2						~ Rd8 Rd8	—	—	↑	↑	0	—	2	
NOT	NOT.W Rd	W	2						~ Rd16 Rd16	—	—	↑	↑	0	—	2	
	NOT.L ERd	L	2						~ Rd32 Rd32	—	—	↑	↑	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*	
		Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa	@(d, PC)		@aa	I	H	N	Z	V	C	ノーマル*	アドバンスト
SHAL	SHAL.B Rd	B	2							—	—	↑	↑	↑	↑	2	
	SHAL.W Rd	W	2							—	—	↑	↑	↑	↑	2	
	SHAL.L ERd	L	2							—	—	↑	↑	↑	↑	2	
SHAR	SHAR.B Rd	B	2							—	—	↑	↑	0	↑	2	
	SHAR.W Rd	W	2							—	—	↑	↑	0	↑	2	
	SHAR.L ERd	L	2							—	—	↑	↑	0	↑	2	
SHLL	SHLL.B Rd	B	2							—	—	↑	↑	0	↑	2	
	SHLL.W Rd	W	2							—	—	↑	↑	0	↑	2	
	SHLL.L ERd	L	2							—	—	↑	↑	0	↑	2	

表A.1 命令セット一覧(7)

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	Rn	@ERn	@(d, ERn)		@-ERn/@ERn+	@aa	@(d, PC)	@@aa	—	I	H	N	Z	V	C	ノーマル ^{*3}	アドバンスト	
SHLR	SHLR.B Rd	B	2													↑	↑	0	↑	2
	SHLR.W Rd	W	2													↑	↑	0	↑	2
	SHLR.L ERd	L	2													↑	↑	0	↑	2
ROTXL	ROTXL.B Rd	B	2													↑	↑	0	↑	2
	ROTXL.W Rd	W	2													↑	↑	0	↑	2
	ROTXL.L ERd	L	2													↑	↑	0	↑	2
ROTXR	ROTXR.B Rd	B	2													↑	↑	0	↑	2
	ROTXR.W Rd	W	2													↑	↑	0	↑	2
	ROTXR.L ERd	L	2													↑	↑	0	↑	2
ROTL	ROTL.B Rd	B	2													↑	↑	0	↑	2
	ROTL.W Rd	W	2													↑	↑	0	↑	2
	ROTL.L ERd	L	2													↑	↑	0	↑	2
ROTR	ROTR.B Rd	B	2													↑	↑	0	↑	2
	ROTR.W Rd	W	2													↑	↑	0	↑	2
	ROTR.L ERd	L	2													↑	↑	0	↑	2

(5) ビット操作命令

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	Rn	@ERn	@(d, ERn)		@-ERn/@ERn+	@aa	@(d, PC)	@@aa	—	I	H	N	Z	V	C	ノーマル ^{*3}	アドバンスト	
BSET	BSET #xx:3, Rd	B	2													↑	↑	—	—	2
	BSET #xx:3, @ERd	B	4													↑	↑	—	—	8
	BSET #xx:3, @aa:8	B		4												↑	↑	—	—	8
	BSET Rn, Rd	B	2													↑	↑	—	—	2
	BSET Rn, @ERd	B		4												↑	↑	—	—	8
BSET Rn, @aa:8	B			4											↑	↑	—	—	8	

表 A.1 命令セット一覧 (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							実行ステート数 ^{*1}		
		#xx	Rn	@ERn	@(d, ERn)		@-ERn/@ERn+	@aa	@(d, PC)	@aa	I	H	N		Z	V
BCLR	BCLR #xx:3, Rd		2													2
	BCLR #xx:3, @ERd			4												8
	BCLR #xx:3, @aa:8				4											8
	BCLR Rn, Rd		2													2
	BCLR Rn, @ERd			4												8
	BCLR Rn, @aa:8				4											8
BNOT	BNOT #xx:3, Rd		2													2
	BNOT #xx:3, @ERd			4												8
	BNOT #xx:3, @aa:8				4											8
	BNOT Rn, Rd		2													2
	BNOT Rn, @ERd			4												8
	BNOT Rn, @aa:8				4											8
BTST	BTST #xx:3, Rd		2													2
	BTST #xx:3, @ERd			4												6
	BTST #xx:3, @aa:8				4											6
	BTST Rn, Rd		2													2
	BTST Rn, @ERd			4												6
	BTST Rn, @aa:8				4											6
BLD	BLD #xx:3, Rd		2													2
	BLD #xx:3, @ERd			4												6
	BLD #xx:3, @aa:8				4											6
	BILD #xx:3, Rd		2													2
	BILD #xx:3, @ERd			4												6
	BILD #xx:3, @aa:8				4											6

表A.1 命令セット一覧(9)

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							実行ステップ数 ^{*1}		
		#xx	Rn	@ERn	@(d, ERn)		@-ERn/@ERn+	@aa	@(d, PC)	@aa	I	H	N		Z	V
BST	B	2														2
BST #xx:3, @ERd	B		4													8
BST #xx:3, @aa:8	B				4											8
BIST	B	2														2
BIST #xx:3, @ERd	B		4													8
BIST #xx:3, @aa:8	B				4											8
BAND	B	2														2
BAND #xx:3, @ERd	B		4													6
BAND #xx:3, @aa:8	B				4											6
BIAND	B	2														2
BIAND #xx:3, @ERd	B		4													6
BIAND #xx:3, @aa:8	B				4											6
BOR	B	2														2
BOR #xx:3, @ERd	B		4													6
BOR #xx:3, @aa:8	B				4											6
BIOR	B	2														2
BIOR #xx:3, @ERd	B		4													6
BIOR #xx:3, @aa:8	B				4											6
BXOR	B	2														2
BXOR #xx:3, @ERd	B		4													6
BXOR #xx:3, @aa:8	B				4											6
BIXOR	B	2														2
BIXOR #xx:3, @ERd	B		4													6
BIXOR #xx:3, @aa:8	B				4											6

表A.1 命令セット一覧(10)

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							実行ステップ数 ^{*1}	
		#xx	Rn	@ERn	@(d, ERn)	@-ERn@ERn+	@aa		@(d, PC)	@aa	I	H	N	Z	V	C	ノーマル ^{*3}
BCC	—	—	—	—	—	—	—	if condition is true then PC PC+d else next;	—	—	—	—	—	—	—	4	—
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	—

(6) 分岐命令

表 A.1 命令セット一覧 (13)

(8) ブロック転送命令

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	Rn	@ERn	@(d, ERn)	@-ERn/@ERn+	@aa		@(d, PC)	@aa	—	I	H	N	Z	V	C	ノーマル*3
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*2	
EEPMOV.W	—								4	if R4 = 0 Repeat @R5 @R6 R5+1 R5 R6+1 R6 R4-1 R4 Until R4=0 else next;	—	—	—	—	—	—	8+4n*2	

【注】 *1 実行スタート数は、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 命令実行スタート数」を参照してください。

*2 nはR4LまたはR4の設定値です。

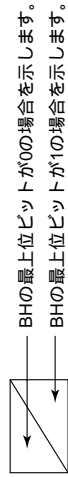
- (1) ビット11から桁上上がりまたはビット11へ桁下がりが発生したとき"1"にセットされ、それ以外のとき"0"にクリアされます。
- (2) ビット27から桁上上がりまたはビット27へ桁下がりが発生したとき"1"にセットされ、それ以外のとき"0"にクリアされます。
- (3) 演算結果がゼロのとき、演算前の値を保持し、それ以外のとき"0"にクリアされます。
- (4) 補正結果に桁上がりが発生したとき、"1"にセットされ、それ以外のとき演算前の値を保持します。
- (5) エクロック同期転送命令の実行スタート数は一定ではありません。
- (6) 除数が負のとき"1"にセットされ、それ以外のとき"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。

*3 本LSIでは、ノーマルモードは使用できません。

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(1)

命令コード:		第1バイト	第2バイト															
		AH	AL	BH	BL													
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
AL	AH	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)	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	BEQ	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	AND	AND	AND	AND	AND	AND	AND	AND	AND	AND	
7					BOR	BXOR	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	
					BOR	BXOR	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	BAND	
8																		
9																		
A																		
B																		
C																		
D																		
E																		
F																		



表A.2 オペレーションコードマップ(2)

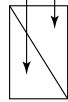
命令コード： 第1バイト 第2バイト
 AH AL AH BH BL

BH	AH	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		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		EXTS
1A		DEC												SUB			
1B		SUBS					DEC		DEC	SUBS					DEC		DEC
1F		DAS												CMP			
58	BRN	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE	BLT	BGT	BLE
79	MOV	ADD	ADD	CMP	SUB	OR	XOR	AND									
7A	MOV	ADD	ADD	CMP	SUB	OR	XOR	AND									

表A.2 オペレーションコードマップ(3)

命令コード: AH AL BH BL CH CL DH DL	第1バイト		第2バイト		第3バイト		第4バイト		9	8	7	6	5	4	3	2	1	0	A	B	C	D	E	F
	CL	DL	AH	AL	BH	BL	CH	CL																
01406																								
01C05			MULXS				MULXS																	
01D05				DIVXS				DIVXS																
01F06									OR	XOR	AND													
7Cr06*1																								
7Cr07*1																								
7Dr06*1			BSET		BNOT																			
7Dr07*1			BSET		BNOT																			
7Eaa6*2																								
7Eaa7*2																								
7Faa6*2			BSET		BNOT																			
7Faa7*2			BSET		BNOT																			

命令コード: 第1バイト 第2バイト 第3バイト 第4バイト
 AH AL BH BL CH CL DH DL
 ← DHの最上位ビットが0の場合を示します。
 → DHの最上位ビットが1の場合を示します。



【注】*1 rはレジスタ指定部

*2 aaは絶対アドレス指定部

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.3 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/ライトなどのサイクル数を示し、表 A.4 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.3 より

$$I = L = 2, J = K = M = N = 0$$

表 A.2 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.3 より

$$I = J = K = 2, L = M = N = 0$$

表 A.2 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 _I	2	6	3	4	6+2m	2	3+m			
分岐アドレスリード S _J										
スタック操作 S _K										
バイトデータアクセス S _L								3	2	3+m
ワードデータアクセス S _M								6	4	6+2m
内部動作 S _N	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	分岐		バイト	ワード	内部動作
		フェッチ	アドレス	スタック	データ	データ	
		I	リード	操作	アクセス	アクセス	
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					

命令	ニーモニック	命令	分岐		バイト	ワード	内部動作
		フェッチ	アドレス	スタック	データ	データ	
		I	リード	操作	アクセス	アクセス	
			J	K	L	M	N
Bcc	BGE d:8	2					
	BLT d:8	2					
	BGT d:8	2					
	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		

命令	ニーモニック	命令	分岐	スタック	バイト	ワード	内部動作
		フェッチ	アドレス	操作	データ	データ	
		I	リード	J	アクセス	アクセス	
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		
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}	2		1		
		アドバンスト	2		2		
	BSR d:16	ノーマル ^{*2}	2		1		2
		アドバンスト	2		2		2

命令	ニーモニック	命令	分岐		バイト	ワード	内部動作
		フェッチ	アドレス リード	スタック 操作	データ アクセス	データ アクセス	
		I	J	K	L	M	N
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		
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
EEMOV	EEMOV.B	2			$2n + 2^{*1}$		
	EEMOV.W	2			$2n + 2^{*1}$		
EXTS	EXTS.W Rd	1					
	EXTS.L ERd	1					
EXTU	EXTU.W Rd	1					
	EXTU.L ERd	1					

命令	ニーモニック	命令	分岐	スタック	バイト	ワード	内部動作	
		フェッチ	アドレス リード	操作	データ アクセス	データ アクセス		
		I	J	K	L	M	N	
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	2	1				2
		アドバンスト	2	2				2
JSR	JSR @ERn	ノーマル*2	2		1			
		アドバンスト	2		2			
	JSR @aa:24	ノーマル*2	2			1		2
		アドバンスト	2			2		2
	JSR @@aa:8	ノーマル*2	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	
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		

命令	ニーモニック	命令	分岐	スタック 操作	バイト	ワード	内部動作	
		フェッチ	アドレス リード		データ	データ		
		I	J		L	M		N
MOV	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	
	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*2	2			1			

命令	ニーモニック	命令	分岐	スタック	バイト	ワード	内部動作
		フェッチ	アドレス リード	操作	データ アクセス	データ アクセス	
		I	J	K	L	M	N
MOVTPE	MOVTPE Rs, @aa:16*2	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
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					

命令	ニーモニック	命令	分岐	スタック	バイト	ワード	内部動作
		フェッチ	アドレス リード	操作	データ アクセス	データ アクセス	
		I	J	K	L	M	N
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	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					
STC	STC CCR, Rd	1					
	STC CCR, @ERd	2				1	
	STC CCR, @(d:16, ERd)	3				1	
	STC CCR, @(d:24, ERd)	5				1	
	STC CCR, @-ERd	2				1	2
	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					

命令	ニーモニック		命令	分岐	スタック	バイト	ワード	内部動作
			フェッチ	アドレス		データ	データ	
			I	J	K	L	M	N
SUBX	SUBX #xx:8, Rd		1					
	SUBX Rs, Rd		1					
TRAPA	TRAPA #x:2	ノーマル*2	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 ではノーマルモードは使用できません。

B. 内部 I/O レジスタ一覧

B.1 アドレス一覧

下位 アドレス	レジスタ名	データ パス幅	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'1C											
H'1D											
H'1E											
H'1F											
H'20											
H'21											
H'22											
H'23											
H'24											
H'25											
H'26											
H'27											
H'28											
H'29											
H'2A											
H'2B											
H'2C											
H'2D											
H'2E											
H'2F											
H'30											
H'31											
H'32											
H'33											
H'34											
H'35											
H'36											
H'37											
H'38											
H'39											
H'3A											

下位 アドレス	レジスタ名	データ パス幅	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'3B											
H'3C											
H'3D											
H'3E											
H'3F											
H'40											
H'41											
H'42											
H'43											
H'44											
H'45											
H'46											
H'47											
H'48											
H'49											
H'4A											
H'4B											
H'4C											
H'4D											
H'4E											
H'4F											
H'50											
H'51											
H'52											
H'53											
H'54											
H'55											
H'56											
H'57											
H'58											
H'59											
H'5A											
H'5B											
H'5C											
H'5D											

下位 アドレス	レジスタ名	データ パス幅	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'5E												
H'5F												
H'60	TSTR	8				STR4	STR3	STR2	STR1	STR0	ITU 共通	
H'61	TSNC	8				SYNC4	SYNC3	SYNC2	SYNC1	SYNC0		
H'62	TMDR	8		MDF	FDIR	PWM4	PWM3	PWM2	PWM1	PWM0		
H'63	TFCR	8			CMD1	CMD0	BFB4	BFA4	BFB3	BFA3		
H'64	TCR0	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	ITU チャンネル0	
H'65	TIOR0	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0		
H'66	TIER0	8						OVIE	IMIEB	IMIEA		
H'67	TSR0	8						OVF	IMFB	IMFA		
H'68	TCNT0H	16										
H'69	TCNT0L											
H'6A	GRA0H	16										
H'6B	GRA0L											
H'6C	GRB0H	16										
H'6D	GRB0L											
H'6E	TCR1	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	ITU チャンネル1	
H'6F	TIOR1	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0		
H'70	TIER1	8						OVIE	IMIEB	IMIEA		
H'71	TSR1	8						OVF	IMFB	IMFA		
H'72	TCNT1H	16										
H'73	TCNT1L											
H'74	GRA1H	16										
H'75	GRA1L											
H'76	GRB1H	16										
H'77	GRB1L											
H'78	TCR2	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	ITU チャンネル2	
H'79	TIOR2	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0		
H'7A	TIER2	8						OVIE	IMIEB	IMIEA		
H'7B	TSR2	8						OVF	IMFB	IMFA		
H'7C	TCNT2H	16										
H'7D	TCNT2L											
H'7E	GRA2H	16										
H'7F	GRA2L											

下位 アドレス	レジスタ名	データ パス幅	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'80	GRB2H	16									ITU チャンネル2
H'81	GRB2L										
H'82	TCR3	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	ITU チャンネル3
H'83	TIOR3	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0	
H'84	TIER3	8						OVIE	IMIEB	IMIEA	
H'85	TSR3	8						OVF	IMFB	IMFA	
H'86	TCNT3H	16									
H'87	TCNT3L										
H'88	GRA3H	16									
H'89	GRA3L										
H'8A	GRB3H	16									
H'8B	GRB3L										
H'8C	BRA3H	16									
H'8D	BRA3L										
H'8E	BRB3H	16									
H'8F	BRB3L										
H'90	TOER	8			EXB4	EXA4	EB3	EB4	EA4	EA3	ITU 共通
H'91	TOCR	8				XTGD			OLS4	OLS3	
H'92	TCR4	8		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	ITU チャンネル4
H'93	TIOR4	8		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0	
H'94	TIER4	8						OVIE	IMIEB	IMIEA	
H'95	TSR4	8						OVF	IMFB	IMFA	
H'96	TCNT4H	16									
H'97	TCNT4L										
H'98	GRA4H	16									
H'99	GRA4L										
H'9A	GRB4H	16									
H'9B	GRB4L										
H'9C	BRA4H	16									
H'9D	BRA4L										
H'9E	BRB4H	16									
H'9F	BRB4L										
H'A0	TPMR	8					G3NOV	G2NOV	G1NOV	G0NOV	TPC
H'A1	TPCR	8	G3CMS1	G3CMS0	G2CMS1	G2CMS0	G1CMS1	G1CMS0	G0CMS1	G0CMS0	
H'A2	NDERB	8	NDER15	NDER14	NDER13	NDER12	NDER11	NDER10	NDER9	NDER8	

下位 アドレス	レジスタ名	データ パス幅	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'A3	NDERA	8	NDER7	NDER6	NDER5	NDER4	NDER3	NDER2	NDER1	NDER0	TPC	
H'A4	NDRB*1	8	NDR15	NDR14	NDR13	NDR12	NDR11	NDR10	NDR9	NDR8		
		8	NDR15	NDR14	NDR13	NDR12						
H'A5	NDRA*1	8	NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0		
		8	NDR7	NDR6	NDR5	NDR4						
H'A6	NDRB*1	8										
		8					NDR11	NDR10	NDR9	NDR8		
H'A7	NDRA*1	8										
		8					NDR3	NDR2	NDR1	NDR0		
H'A8	TCSR*2	8	OVF	WT/IT	TME			CKS2	CKS1	CKS0		WDT
H'A9	TCNT*2	8										
H'AA												
H'AB	RSTCSR*2	8	WRST	RSTOE								
H'AC												
H'AD												
H'AE												
H'AF												
H'B0	SMR	8	C/A	CHR	PE	O/E	STOP	MP	CKS1	CKS0	SCI	
H'B1	BRR	8										
H'B2	SCR	8	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0		
H'B3	TDR	8										
H'B4	SSR	8	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT		
H'B5	RDR	8										
H'B6												
H'B7												
H'B8												
H'B9												
H'BA												
H'BB												
H'BC												
H'BD												
H'BE												
H'BF												
H'C0	P1DDR	8	P1 ₇ DDR	P1 ₆ DDR	P1 ₅ DDR	P1 ₄ DDR	P1 ₃ DDR	P1 ₂ DDR	P1 ₁ DDR	P1 ₀ DDR	ポート1	
H'C1	P2DDR	8	P2 ₇ DDR	P2 ₆ DDR	P2 ₅ DDR	P2 ₄ DDR	P2 ₃ DDR	P2 ₂ DDR	P2 ₁ DDR	P2 ₀ DDR	ポート2	

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'C2	P1DR	8	P1 ₇	P1 ₆	P1 ₅	P1 ₄	P1 ₃	P1 ₂	P1 ₁	P1 ₀	ポート1
H'C3	P2DR	8	P2 ₇	P2 ₆	P2 ₅	P2 ₄	P2 ₃	P2 ₂	P2 ₁	P2 ₀	ポート2
H'C4	P3DDR	8	P3 ₇ DDR	P3 ₆ DDR	P3 ₅ DDR	P3 ₄ DDR	P3 ₃ DDR	P3 ₂ DDR	P3 ₁ DDR	P3 ₀ DDR	ポート3
H'C5		8									
H'C6	P3DR	8	P3 ₇	P3 ₆	P3 ₅	P3 ₄	P3 ₃	P3 ₂	P3 ₁	P3 ₀	ポート3
H'C7		8									
H'C8	P5DDR	8					P5 ₃ DDR	P5 ₂ DDR	P5 ₁ DDR	P5 ₀ DDR	ポート5
H'C9	P6DDR	8			P6 ₅ DDR	P6 ₄ DDR	P6 ₃ DDR			P6 ₀ DDR	ポート6
H'CA	P5DR	8					P5 ₃	P5 ₂	P5 ₁	P5 ₀	ポート5
H'CB	P6DR	8			P6 ₅	P6 ₄	P6 ₃			P6 ₀	ポート6
H'CC											
H'CD	P8DDR	8					P8 ₃ DDR	P8 ₂ DDR	P8 ₁ DDR	P8 ₀ DDR	ポート8
H'CE	P7DR	8	P7 ₇	P7 ₆	P7 ₅	P7 ₄	P7 ₃	P7 ₂	P7 ₁	P7 ₀	ポート7
H'CF	P8DR	8					P8 ₃	P8 ₂	P8 ₁	P8 ₀	ポート8
H'D0	P9DDR	8				P9 ₄ DDR		P9 ₂ DDR		P9 ₀ DDR	ポート9
H'D1	PADDR	8	PA ₇ DDR	PA ₆ DDR	PA ₅ DDR	PA ₄ DDR	PA ₃ DDR	PA ₂ DDR	PA ₁ DDR	PA ₀ DDR	ポートA
H'D2	P9DR	8				P9 ₄		P9 ₂		P9 ₀	ポート9
H'D3	PADR	8	PA ₇	PA ₆	PA ₅	PA ₄	PA ₃	PA ₂	PA ₁	PA ₀	ポートA
H'D4	PBDDR	8	PB ₇ DDR	PB ₆ DDR	PB ₅ DDR	PB ₄ DDR	PB ₃ DDR	PB ₂ DDR	PB ₁ DDR	PB ₀ DDR	ポートB
H'D5											
H'D6	PBDR	8	PB ₇	PB ₆	PB ₅	PB ₄	PB ₃	PB ₂	PB ₁	PB ₀	ポートB
H'D7											
H'D8	P2PCR	8	P2 ₇ PCR	P2 ₆ PCR	P2 ₅ PCR	P2 ₄ PCR	P2 ₃ PCR	P2 ₂ PCR	P2 ₁ PCR	P2 ₀ PCR	ポート2
H'D9											
H'DA											
H'DB	P5PCR	8					P5 ₃ PCR	P5 ₂ PCR	P5 ₁ PCR	P5 ₀ PCR	ポート5
H'DC											
H'DD											
H'DE											
H'DF											
H'E0	ADDRAH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D 変換器
H'E1	ADDRAL	8	AD1	AD0							
H'E2	ADDRBH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'E3	ADDRBL	8	AD1	AD0							
H'E4	ADDRCH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	

下位 アドレス	レジスタ名	データ パス幅	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'E5	ADDRCL	8	AD1	AD0							A/D 変換器
H'E6	ADDRDH	8	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
H'E7	ADDRDL	8	AD1	AD0							
H'E8	ADCSR	8	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0	
H'E9	ADCR	8	TRGE								
H'EA											
H'EB											バスコント ローラ
H'EC											
H'ED	ASTCR	8	AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0	
H'EE	WCR	8					WMS1	WMS0	WC1	WC0	
H'EF	WCER	8	WCE7	WCE6	WCE5	WCE4	WCE3	WCE2	WCE1	WCE0	
H'F0											
H'F1	MDCR	8							MDS1	MDS0	システム 制御
H'F2	SYSCR	8	SSBY	STS2	STS1	STS0	UE	NMIEG		RAME	
H'F3	ADRCR	8	A ₂₃ E	A ₂₂ E	A ₂₁ E						バスコント ローラ
H'F4	ISCR	8				IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC	割込み コント ローラ
H'F5	IER	8				IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E	
H'F6	ISR	8				IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F	
H'F7											
H'F8	IPRA	8	IPRA7	IPRA6	IPRA5	IPRA4	IPRA3	IPRA2	IPRA1	IPRA0	
H'F9	IPRB	8	IPRB7	IPRB6			IPRB3		IPRB1		
H'FA											
H'FB											
H'FC											
H'FD											
H'FE											
H'FF											

【注】 *1 出力トリガの設定によりアドレスが変化します。

*2 TCSR、TCNT、RSTCSR のライトについては「10.2.4 レジスタ書換え時の注意」を参照してください。

《記号説明》

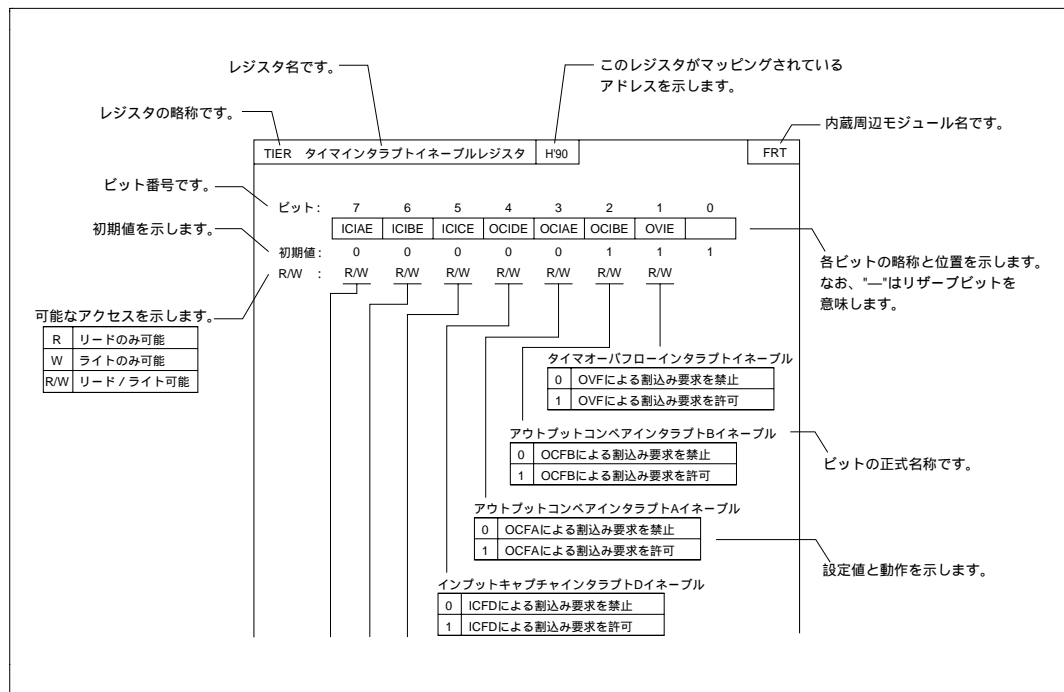
ITU : 16 ビットインテグレートドタイマユニット

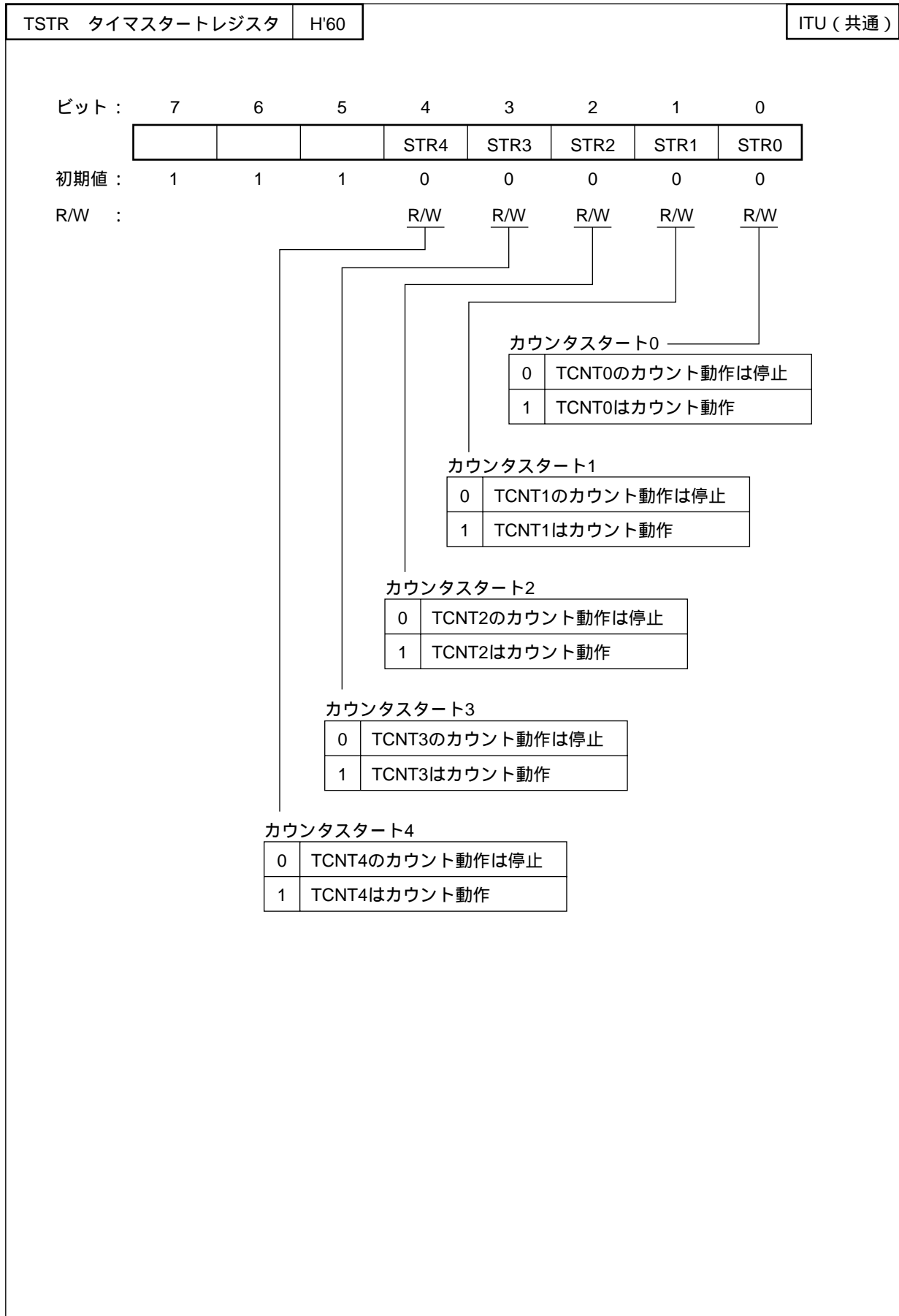
TPC : プログラマブルタイミングパターンコントローラ

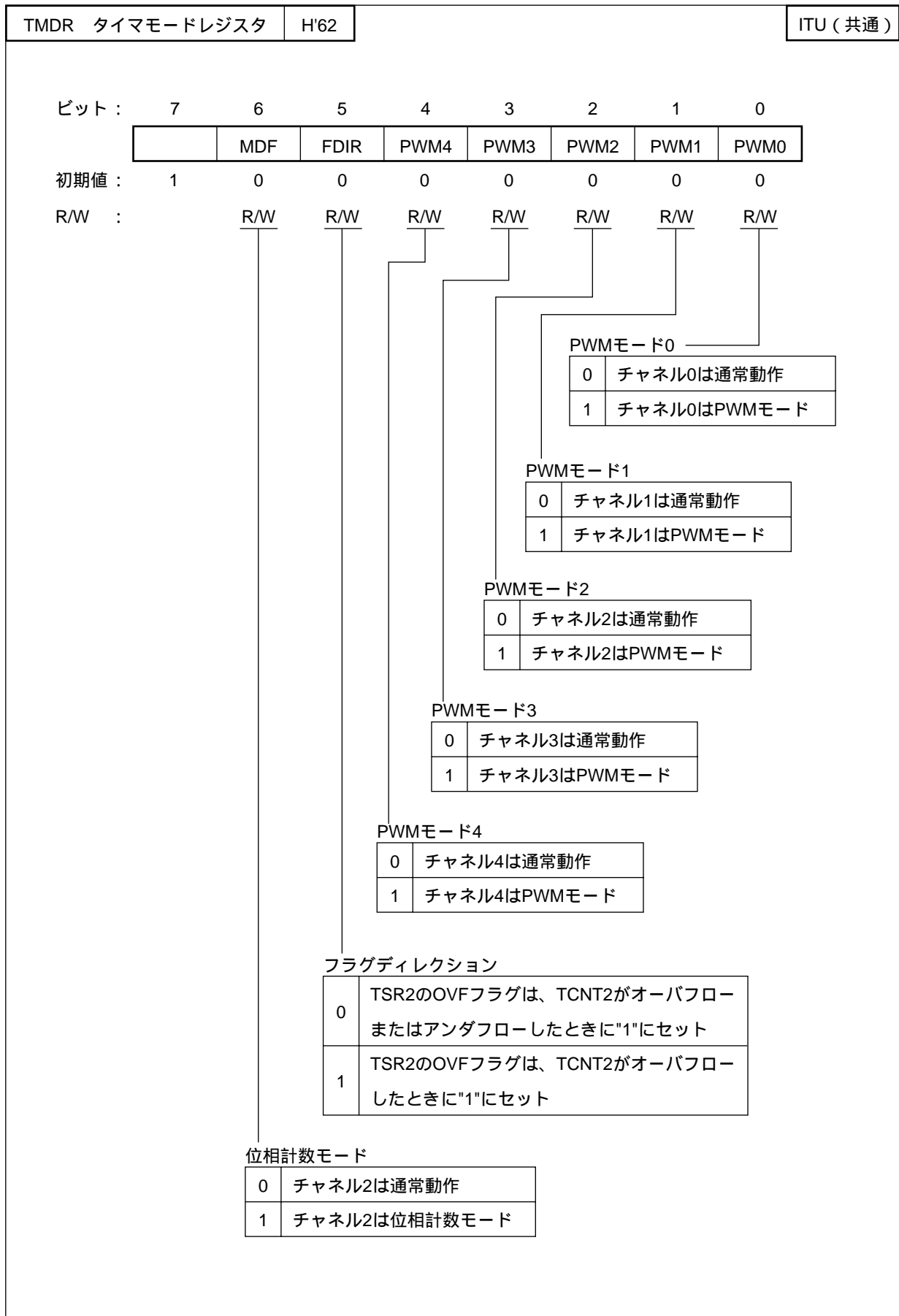
WDT : ウォッチドッグタイマ

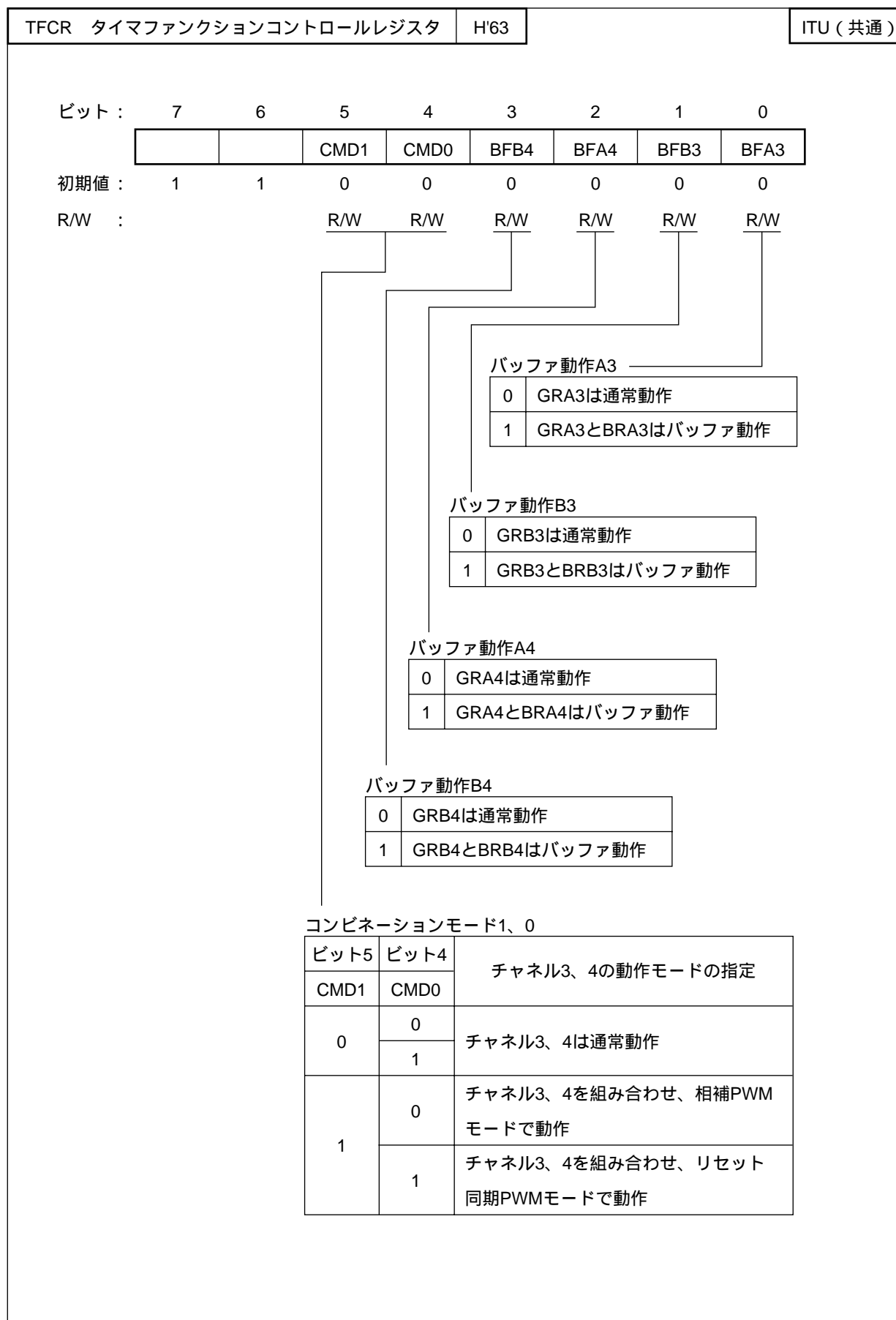
SCI : シリアルコミュニケーションインタフェース

B.2 機能一覧



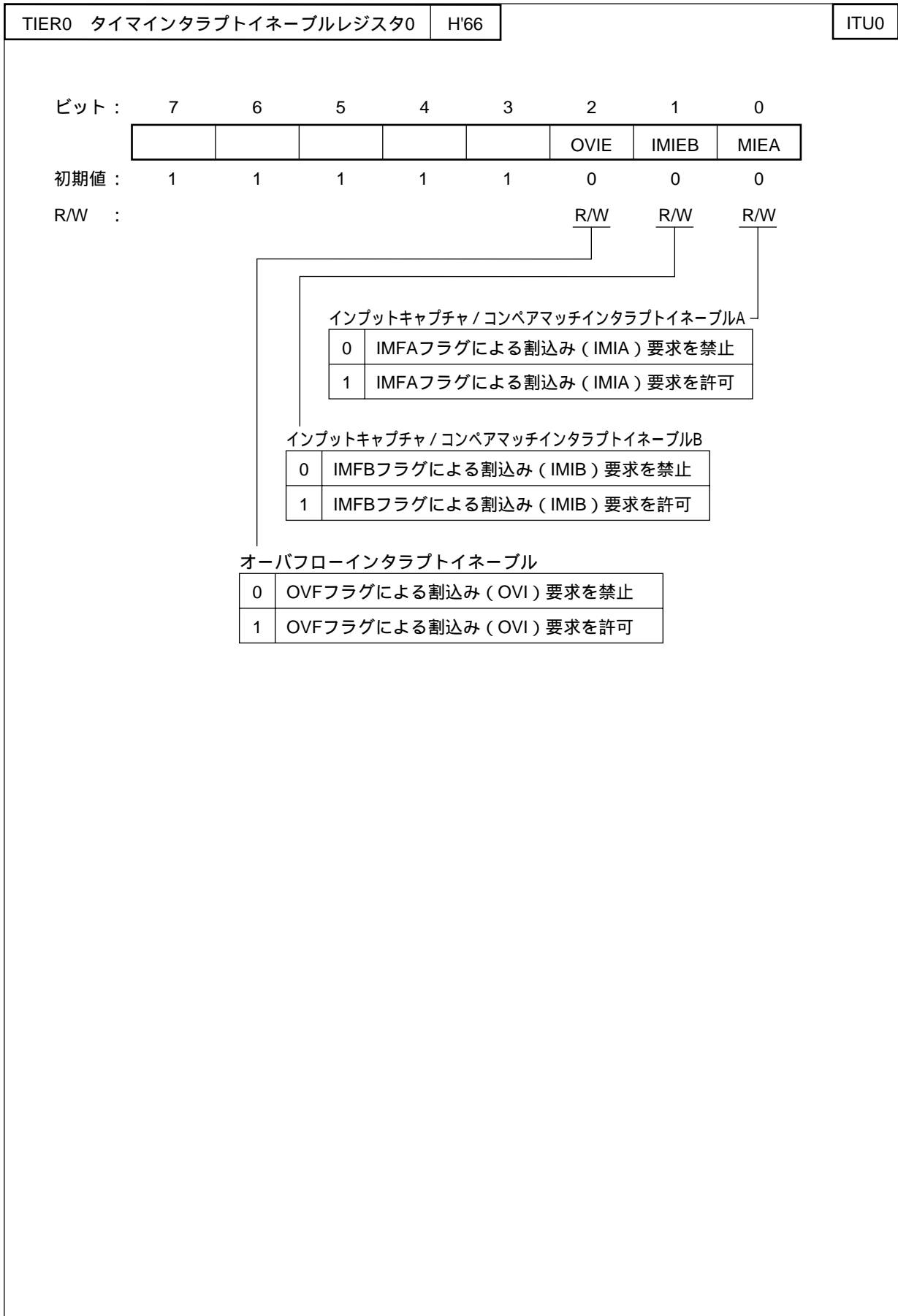






TCR0 タイマコントロールレジスタ0	H'64	ITU0																														
ビット : 7 6 5 4 3 2 1 0 <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%; text-align: center;">CCLR1</td> <td style="width: 10%; text-align: center;">CCLR0</td> <td style="width: 10%; text-align: center;">CKEG1</td> <td style="width: 10%; text-align: center;">CKEG0</td> <td style="width: 10%; text-align: center;">TPSC2</td> <td style="width: 10%; text-align: center;">TPSC1</td> <td style="width: 10%; text-align: center;">TPSC0</td> </tr> </table> 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		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0																								
	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0																									
タイマプリスケラ2~0 <table border="1" style="margin: 10px auto; 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%;">TCNTのカウンタクロック</th> </tr> <tr> <th>TPSC2</th> <th>TPSC1</th> <th>TPSC0</th> <th></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>内部クロック :</td> </tr> <tr> <td style="text-align: center;">1</td> <td>内部クロック : /2</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>内部クロック : /4</td> </tr> <tr> <td style="text-align: center;">1</td> <td>内部クロック : /8</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>外部クロックA : TCLKA端子入力でカウント</td> </tr> <tr> <td style="text-align: center;">1</td> <td>外部クロックB : TCLKB端子入力でカウント</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>外部クロックC : TCLKC端子入力でカウント</td> </tr> <tr> <td style="text-align: center;">1</td> <td>外部クロックD : TCLKD端子入力でカウント</td> </tr> </tbody> </table>			ビット2	ビット1	ビット0	TCNTのカウンタクロック	TPSC2	TPSC1	TPSC0		0	0	0	内部クロック :	1	内部クロック : /2	1	0	内部クロック : /4	1	内部クロック : /8	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
ビット2	ビット1	ビット0	TCNTのカウンタクロック																													
TPSC2	TPSC1	TPSC0																														
0	0	0	内部クロック :																													
		1	内部クロック : /2																													
	1	0	内部クロック : /4																													
		1	内部クロック : /8																													
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 <table border="1" style="margin: 5px auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット4</th> <th style="width: 10%;">ビット3</th> <th style="width: 80%;">外部クロックの検出エッジ</th> </tr> <tr> <th>CKEG1</th> <th>CKEG0</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>立上がりエッジでカウント</td> </tr> <tr> <td style="text-align: center;">1</td> <td>立下がりエッジでカウント</td> </tr> <tr> <td style="text-align: center;">1</td> <td></td> <td>立上がり / 立下がりの両エッジでカウント</td> </tr> </tbody> </table>			ビット4	ビット3	外部クロックの検出エッジ	CKEG1	CKEG0		0	0	立上がりエッジでカウント	1	立下がりエッジでカウント	1		立上がり / 立下がりの両エッジでカウント																
ビット4	ビット3	外部クロックの検出エッジ																														
CKEG1	CKEG0																															
0	0	立上がりエッジでカウント																														
	1	立下がりエッジでカウント																														
1		立上がり / 立下がりの両エッジでカウント																														
カウンタクリア1、0 <table border="1" style="margin: 5px auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット6</th> <th style="width: 10%;">ビット5</th> <th style="width: 80%;">TCNTのクリアソース</th> </tr> <tr> <th>CCLR1</th> <th>CCLR0</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>TCNTのクリア禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>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</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td>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</td> </tr> <tr> <td style="text-align: center;">1</td> <td>同期クリア。同期動作中の他のタイマのカウンタクリアに同期してTCNTをクリア</td> </tr> </tbody> </table>			ビット6	ビット5	TCNTのクリアソース	CCLR1	CCLR0		0	0	TCNTのクリア禁止	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。同期動作中の他のタイマのカウンタクリアに同期してTCNTをクリア														
ビット6	ビット5	TCNTのクリアソース																														
CCLR1	CCLR0																															
0	0	TCNTのクリア禁止																														
	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。同期動作中の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'65	ITU0																																				
<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;"></td> <td style="width: 20px; text-align: center;">IOB2</td> <td style="width: 20px; text-align: center;">IOB1</td> <td style="width: 20px; text-align: center;">IOB0</td> <td style="width: 20px;"></td> <td style="width: 20px; text-align: center;">IOA2</td> <td style="width: 20px; text-align: center;">IOA1</td> <td style="width: 20px; text-align: center;">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>				IOB2	IOB1	IOB0		IOA2	IOA1	IOA0																												
	IOB2	IOB1	IOB0		IOA2	IOA1	IOA0																															
<p>I/OコントロールA2~0</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 colspan="2" style="text-align: center;">GRAの機能の選択</th> </tr> <tr> <th style="text-align: center;">IOA2</th> <th style="text-align: center;">IOA1</th> <th style="text-align: center;">IOA0</th> <th colspan="2"></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 rowspan="2" style="text-align: center; vertical-align: middle;">GRAはアウトプット</td> <td>コンペアマッチによる端子出力禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>GRAのコンペアマッチで0出力</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">コンペアレジスタ</td> <td>GRAのコンペアマッチで1出力</td> </tr> <tr> <td style="text-align: center;">1</td> <td>GRAのコンペアマッチでトグル出力</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 rowspan="2" style="text-align: center; vertical-align: middle;">GRAはインプット</td> <td>立上がりエッジでGRAへインプットキャプチャ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>立下がりエッジでGRAへインプットキャプチャ</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">キャプチャレジスタ</td> <td>立上がり / 立下がり両エッジでGRAへインプットキャプチャ</td> </tr> <tr> <td style="text-align: center;">1</td> <td></td> </tr> </tbody> </table>			ビット2	ビット1	ビット0	GRAの機能の選択		IOA2	IOA1	IOA0			0	0	0	GRAはアウトプット	コンペアマッチによる端子出力禁止	1	GRAのコンペアマッチで0出力	1	0	コンペアレジスタ	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	1	0	0	GRAはインプット	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	キャプチャレジスタ	立上がり / 立下がり両エッジでGRAへインプットキャプチャ	1	
ビット2	ビット1	ビット0	GRAの機能の選択																																			
IOA2	IOA1	IOA0																																				
0	0	0	GRAはアウトプット	コンペアマッチによる端子出力禁止																																		
		1		GRAのコンペアマッチで0出力																																		
	1	0	コンペアレジスタ	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																																		
1	0	0	GRAはインプット	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	キャプチャレジスタ	立上がり / 立下がり両エッジでGRAへインプットキャプチャ																																		
		1																																				
<p>I/OコントロールB2~0</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">ビット6</th> <th style="width: 10%;">ビット5</th> <th style="width: 10%;">ビット4</th> <th colspan="2" style="text-align: center;">GRBの機能の選択</th> </tr> <tr> <th style="text-align: center;">IOB2</th> <th style="text-align: center;">IOB1</th> <th style="text-align: center;">IOB0</th> <th colspan="2"></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 rowspan="2" style="text-align: center; vertical-align: middle;">GRBはアウトプット</td> <td>コンペアマッチによる端子出力禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>GRBのコンペアマッチで0出力</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">コンペアレジスタ</td> <td>GRBのコンペアマッチで1出力</td> </tr> <tr> <td style="text-align: center;">1</td> <td>GRBのコンペアマッチでトグル出力</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 rowspan="2" style="text-align: center; vertical-align: middle;">GRBはインプット</td> <td>立上がりエッジでGRBへインプットキャプチャ</td> </tr> <tr> <td style="text-align: center;">1</td> <td>立下がりエッジでGRBへインプットキャプチャ</td> </tr> <tr> <td rowspan="2" style="text-align: center; vertical-align: middle;">1</td> <td style="text-align: center;">0</td> <td rowspan="2" style="text-align: center; vertical-align: middle;">キャプチャレジスタ</td> <td>立上がり / 立下がり両エッジでGRBへインプットキャプチャ</td> </tr> <tr> <td style="text-align: center;">1</td> <td></td> </tr> </tbody> </table>			ビット6	ビット5	ビット4	GRBの機能の選択		IOB2	IOB1	IOB0			0	0	0	GRBはアウトプット	コンペアマッチによる端子出力禁止	1	GRBのコンペアマッチで0出力	1	0	コンペアレジスタ	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	1	0	0	GRBはインプット	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	キャプチャレジスタ	立上がり / 立下がり両エッジでGRBへインプットキャプチャ	1	
ビット6	ビット5	ビット4	GRBの機能の選択																																			
IOB2	IOB1	IOB0																																				
0	0	0	GRBはアウトプット	コンペアマッチによる端子出力禁止																																		
		1		GRBのコンペアマッチで0出力																																		
	1	0	コンペアレジスタ	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																																		
1	0	0	GRBはインプット	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	キャプチャレジスタ	立上がり / 立下がり両エッジでGRBへインプットキャプチャ																																		
		1																																				



TSR0 タイムステータスレジスタ0		H'67						ITU0																																												
ビット :	7	6	5	4	3	2	1	0																																												
						OVF	IMFB	IMFA																																												
初期値 :	1	1	1	1	1	0	0	0																																												
R/W :						R/(W)*	R/(W)*	R/(W)*																																												
インพุットキャプチャ/コンペアマッチフラグA																																																				
<table border="1"> <tr> <td colspan="9">〔クリア条件〕</td> </tr> <tr> <td>0</td> <td colspan="8">IMFA="1"の状態、IMFAフラグをリードした後、IMFAフラグに"0"をライトしたとき</td> </tr> <tr> <td colspan="9">〔セット条件〕</td> </tr> <tr> <td rowspan="2">1</td> <td colspan="8">(1) GRAがアウトพุットコンペアレジスタとして機能している場合、TCNT=GRAになったとき</td> </tr> <tr> <td colspan="8">(2) 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場合、インพุットキャプチャ信号によりTCNTの値がGRAに転送されたとき</td> </tr> </table>									〔クリア条件〕									0	IMFA="1"の状態、IMFAフラグをリードした後、IMFAフラグに"0"をライトしたとき								〔セット条件〕									1	(1) GRAがアウトพุットコンペアレジスタとして機能している場合、TCNT=GRAになったとき								(2) 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場合、インพุットキャプチャ信号によりTCNTの値がGRAに転送されたとき							
〔クリア条件〕																																																				
0	IMFA="1"の状態、IMFAフラグをリードした後、IMFAフラグに"0"をライトしたとき																																																			
〔セット条件〕																																																				
1	(1) GRAがアウトพุットコンペアレジスタとして機能している場合、TCNT=GRAになったとき																																																			
	(2) 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場合、インพุットキャプチャ信号によりTCNTの値がGRAに転送されたとき																																																			
インพุットキャプチャ/コンペアマッチフラグB																																																				
<table border="1"> <tr> <td colspan="9">〔クリア条件〕</td> </tr> <tr> <td>0</td> <td colspan="8">IMFB="1"の状態、IMFBフラグをリードした後、IMFBフラグに"0"をライトしたとき</td> </tr> <tr> <td colspan="9">〔セット条件〕</td> </tr> <tr> <td rowspan="2">1</td> <td colspan="8">(1) GRBがアウトพุットコンペアレジスタとして機能している場合、TCNT=GRBになったとき</td> </tr> <tr> <td colspan="8">(2) GRBがインพุットキャプチャレジスタとして機能している場合、インพุットキャプチャ信号によりTCNTの値がGRBに転送されたとき</td> </tr> </table>									〔クリア条件〕									0	IMFB="1"の状態、IMFBフラグをリードした後、IMFBフラグに"0"をライトしたとき								〔セット条件〕									1	(1) GRBがアウトพุットコンペアレジスタとして機能している場合、TCNT=GRBになったとき								(2) GRBがインพุットキャプチャレジスタとして機能している場合、インพุットキャプチャ信号によりTCNTの値がGRBに転送されたとき							
〔クリア条件〕																																																				
0	IMFB="1"の状態、IMFBフラグをリードした後、IMFBフラグに"0"をライトしたとき																																																			
〔セット条件〕																																																				
1	(1) GRBがアウトพุットコンペアレジスタとして機能している場合、TCNT=GRBになったとき																																																			
	(2) GRBがインพุットキャプチャレジスタとして機能している場合、インพุットキャプチャ信号によりTCNTの値がGRBに転送されたとき																																																			
オーバフローフラグ																																																				
<table border="1"> <tr> <td colspan="9">〔クリア条件〕</td> </tr> <tr> <td>0</td> <td colspan="8">OVF="1"の状態、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 colspan="9">〔セット条件〕</td> </tr> <tr> <td>1</td> <td colspan="8">TCNTの値がオーバーフロー (H'FFFF H'0000) したとき</td> </tr> </table>									〔クリア条件〕									0	OVF="1"の状態、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								〔セット条件〕									1	TCNTの値がオーバーフロー (H'FFFF H'0000) したとき															
〔クリア条件〕																																																				
0	OVF="1"の状態、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																																																			
〔セット条件〕																																																				
1	TCNTの値がオーバーフロー (H'FFFF H'0000) したとき																																																			
【注】 * フラグクリアのための"0"ライトのみ可能です。																																																				

TCNT0 H、L タイマカウンタ0 H、L H'68、H'69	ITU0
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <div style="border: 1px solid black; width: 100%; height: 20px; margin-bottom: 5px;"></div> <p>初期値： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</p> <p>R/W : <u>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</u></p> <p style="text-align: center;">↑</p> <p style="text-align: center;">アップカウンタ</p>	
GRA0 H、L ジェネラルレジスタA0 H、L H'6A、H'6B	ITU0
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <div style="border: 1px solid black; width: 100%; height: 20px; margin-bottom: 5px;"></div> <p>初期値： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W : <u>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</u></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>	
GRB0 H、L ジェネラルレジスタB0 H、L H'6C、H'6D	ITU0
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <div style="border: 1px solid black; width: 100%; height: 20px; margin-bottom: 5px;"></div> <p>初期値： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W : <u>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</u></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>	

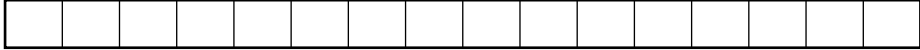


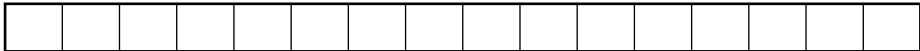
TCR1 タイマコントロールレジスタ1		H'6E						ITU1
ビット :	7	6	5	4	3	2	1	0
		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
初期値 :	1	0	0	0	0	0	0	0
R/W :		R/W	R/W	R/W	R/W	R/W	R/W	R/W
機能はITU0と同じです。								
TIOR1 タイマI/Oコントロールレジスタ1		H'6F						ITU1
ビット :	7	6	5	4	3	2	1	0
		IOB2	IOB1	IOB0		IOA2	IOA1	IOA0
初期値 :	1	0	0	0	1	0	0	0
R/W :		R/W	R/W	R/W		R/W	R/W	R/W
機能はITU0と同じです。								
TIER1 タイマインタラプトイネーブルレジスタ1		H'70						ITU1
ビット :	7	6	5	4	3	2	1	0
						OVIE	IMIEB	IMIEA
初期値 :	1	1	1	1	1	0	0	0
R/W :						R/W	R/W	R/W
機能はITU0と同じです。								
TSR1 タイマステータスレジスタ1		H'71						ITU1
ビット :	7	6	5	4	3	2	1	0
						OVF	IMFB	IMFA
初期値 :	1	1	1	1	1	0	0	0
R/W :						R/(W)*	R/(W)*	R/(W)*
機能はITU0と同じです。								
【注】 * フラグクリアのための"0"ライトのみ可能です。								

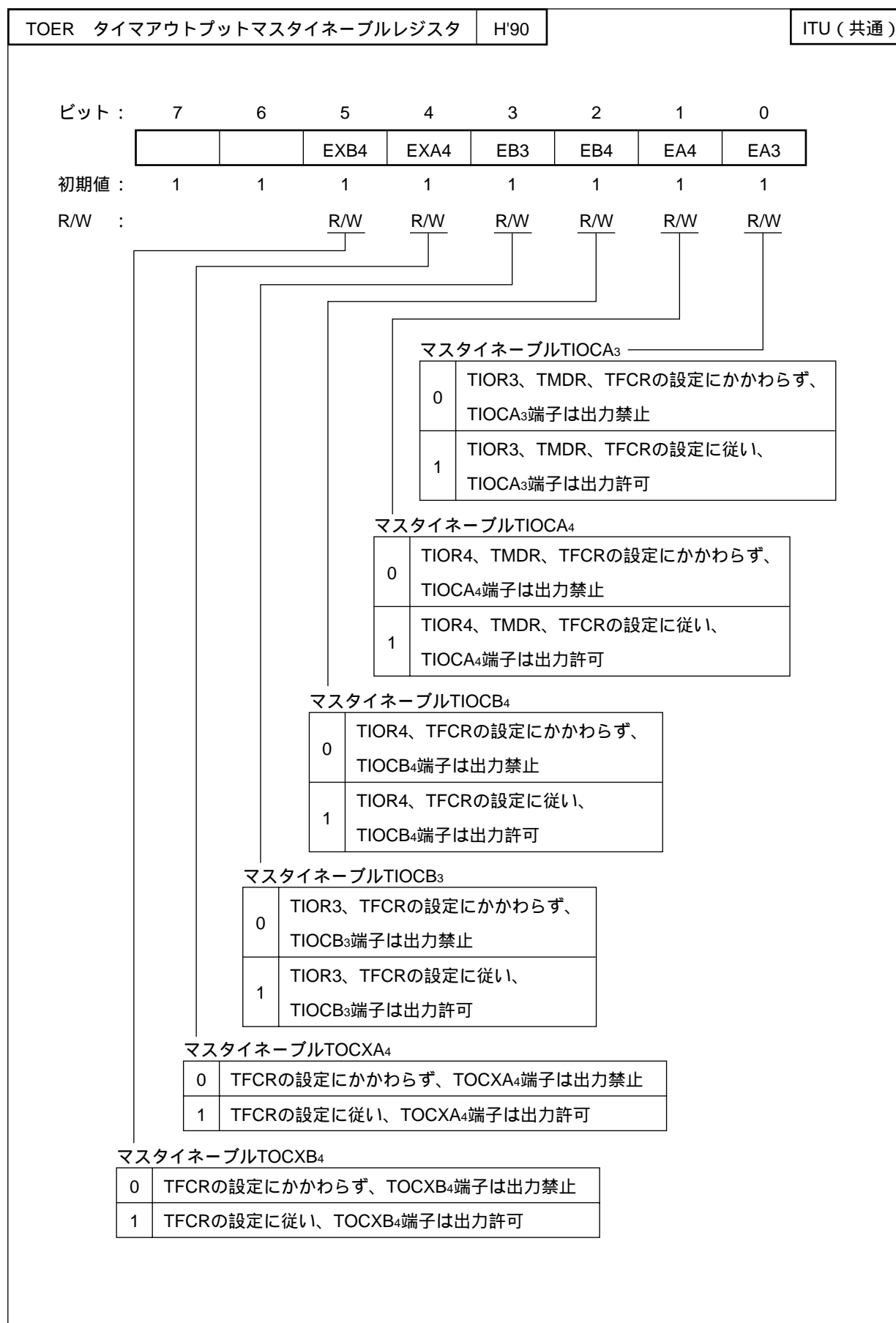
TCNT1 H、L タイマカウンタ1 H、L	H'72、H'73	ITU1																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 439 1278 488"> <tr> <td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td><td> </td> </tr> </table> <p>初期値： 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 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 R/W</p> <p>機能はITU0と同じです。</p>																		
GRA1 H、L ジェネラルレジスタA1 H、L	H'74、H'75	ITU1																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 1032 1278 1081"> <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： 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 R/W</p> <p>機能はITU0と同じです。</p>																		
GRB1 H、L ジェネラルレジスタB1 H、L	H'76、H'77	ITU1																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 1626 1278 1675"> <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： 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 R/W</p> <p>機能はITU0と同じです。</p>																		

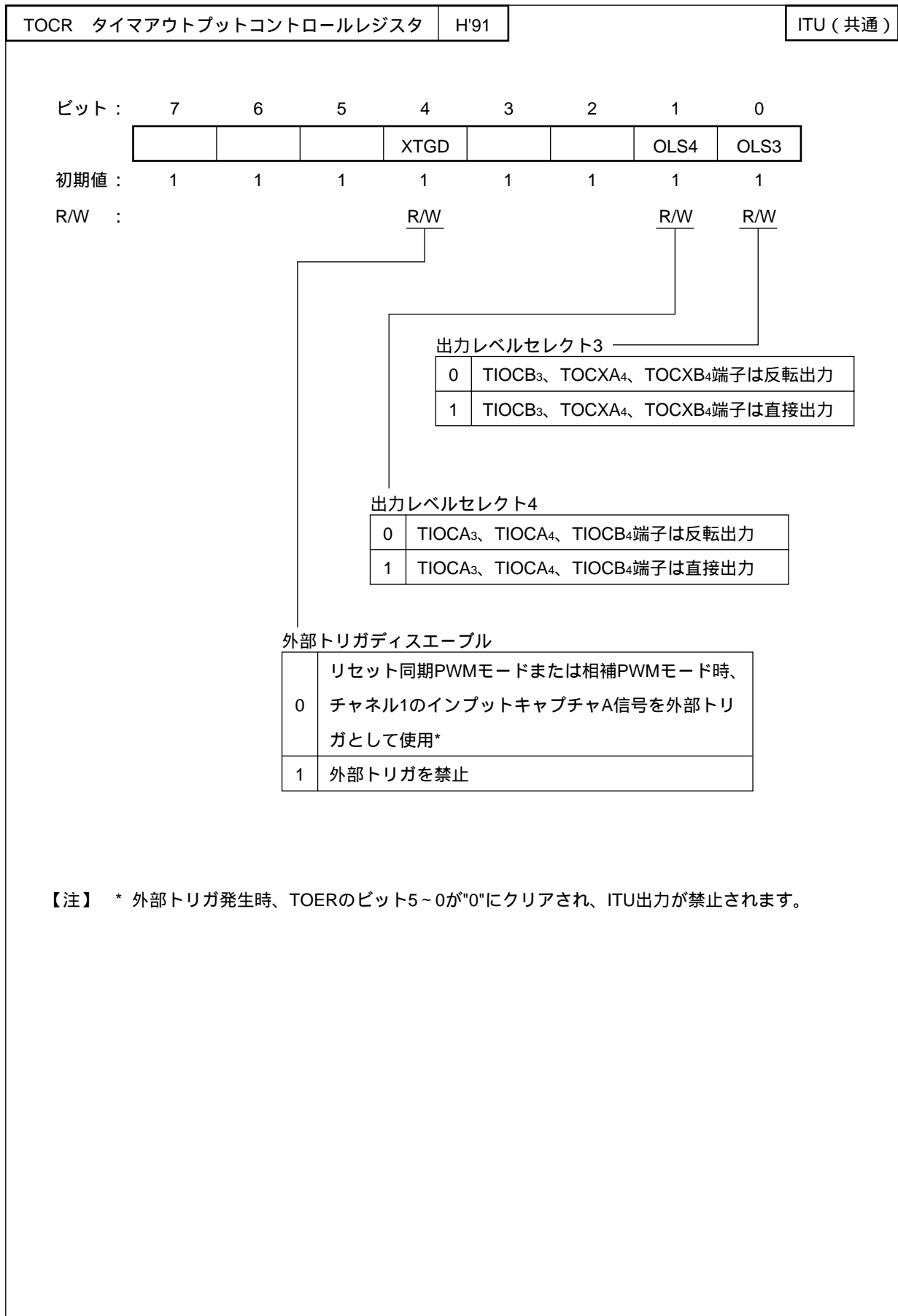
TCR2 タイマコントロールレジスタ2		H'78						ITU2
ビット :	7	6	5	4	3	2	1	0
		CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
初期値 :	1	0	0	0	0	0	0	0
R/W :		R/W	R/W	R/W	R/W	R/W	R/W	R/W
機能はITU0と同じです。								
【注】 チャンネル2を位相計数モードに設定したとき、CKEG1、CKEG0ビットおよびTPSC2～TPSC0ビットによるカウントクロックの選択は無効となります。								
TIOA2 タイマI/Oコントロールレジスタ2		H'79						ITU2
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
機能はITU0と同じです。								
【注】 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。								

GRA2 H、 L ジェネラルレジスタA2 H、 L	H'7E、 H'7F	ITU2																
<p data-bbox="252 398 1257 427">ビット : 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 439 1278 488"><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 data-bbox="252 499 1257 528">初期値 : 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p data-bbox="252 551 1257 580">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 data-bbox="252 658 512 687">機能はITU0と同じです。</p>																		
GRB2 H、 L ジェネラルレジスタB2 H、 L	H'80、 H'81	ITU2																
<p data-bbox="252 1294 1257 1323">ビット : 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 1335 1278 1384"><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 data-bbox="252 1395 1257 1424">初期値 : 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p data-bbox="252 1447 1257 1476">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 data-bbox="252 1554 512 1583">機能はITU0と同じです。</p>																		

TCR3 タイマコントロールレジスタ3	H'82	ITU3								
<p>ビット： 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse; width: 80%;"> <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 R/W</p> <p style="margin-top: 20px;">機能はITU0と同じです。</p>				CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0			
TIOR3 タイマI/Oコントロールレジスタ3	H'83	ITU3								
<p>ビット： 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse; width: 80%;"> <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 style="margin-top: 20px;">機能はITU0と同じです。</p>				IOB2	IOB1	IOB0		IOA2	IOA1	IOA0
	IOB2	IOB1	IOB0		IOA2	IOA1	IOA0			
TIER3 タイマインタラプトイネーブルレジスタ3	H'84	ITU3								
<p>ビット： 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse; width: 80%;"> <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;">OVIE</td> <td style="width: 20px;">IMIEB</td> <td style="width: 20px;">IMIEA</td> </tr> </table> <p>初期値： 1 1 1 1 1 0 0 0</p> <p>R/W : R/W R/W R/W</p> <p style="margin-top: 20px;">機能はITU0と同じです。</p>								OVIE	IMIEB	IMIEA
					OVIE	IMIEB	IMIEA			

GRA3 H、L ジェネラルレジスタA3 H、L	H'88、H'89	ITU3
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <div style="text-align: center;">  </div> <p>初期値： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W : <u>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</u></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 (バッファ動作可能)</p>		
GRB3 H、L ジェネラルレジスタB3 H、L	H'8A、H'8B	ITU3
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <div style="text-align: center;">  </div> <p>初期値： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W : <u>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</u></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 (バッファ動作可能)</p>		
BRA3 H、L バッファレジスタA3 H、L	H'8C、H'8D	ITU3
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <div style="text-align: center;">  </div> <p>初期値： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W : <u>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</u></p> <p style="text-align: center;">↓</p> <p style="text-align: center;">バッファ動作時にGRAと組み合わせて使用</p>		
BRB3 H、L バッファレジスタB3 H、L	H'8E、H'8F	ITU3
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <div style="text-align: center;">  </div> <p>初期値： 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</p> <p>R/W : <u>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</u></p> <p style="text-align: center;">↓</p> <p style="text-align: center;">バッファ動作時にGRBと組み合わせて使用</p>		





TCR4 タイマコントロールレジスタ4		H'92						ITU4
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
機能はITU0と同じです。								
TIOA4 タイマI/Oコントロールレジスタ4		H'93						ITU4
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
機能はITU0と同じです。								
TIER4 タイマインタラプトイネーブルレジスタ4		H'94						ITU4
ビット :	7	6	5	4	3	2	1	0
						OVIE	IMIEB	IMIEA
初期値 :	1	1	1	1	1	0	0	0
R/W :						R/W	R/W	R/W
機能はITU0と同じです。								
TSR4 タイマステータスレジスタ4		H'95						ITU4
ビット :	7	6	5	4	3	2	1	0
						OVF	IMFB	IMFA
初期値 :	1	1	1	1	1	0	0	0
R/W :						R/(W)*	R/(W)*	R/(W)*
機能はITU0と同じです。								
【注】 * フラグクリアのための"0"ライトのみ可能です。								

TCNT4 H、L ジェネラルレジスタA4 H、L	H'96、H'97	ITU4																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 439 1278 488"> <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 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 R/W</p> <p>機能はITU3と同じです。</p>																		
GRA4 H、L ジェネラルレジスタA4 H、L	H'98、H'99	ITU4																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 1335 1278 1384"> <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： 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 R/W</p> <p>機能はITU3と同じです。</p>																		

GRB4 H、L ジェネラルレジスタB4 H、L	H'9A、H'9B	ITU4																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 439 1278 488"> <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： 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ITU3と同じです。</p>																		
BRA4 H、L バッファレジスタA4 H、L	H'9C、H'9D	ITU4																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 1032 1278 1081"> <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： 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ITU3と同じです。</p>																		
BRB4 H、L バッファレジスタB4 H、L	H'9E、H'9F	ITU4																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table border="1" data-bbox="355 1628 1278 1677"> <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： 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ITU3と同じです。</p>																		

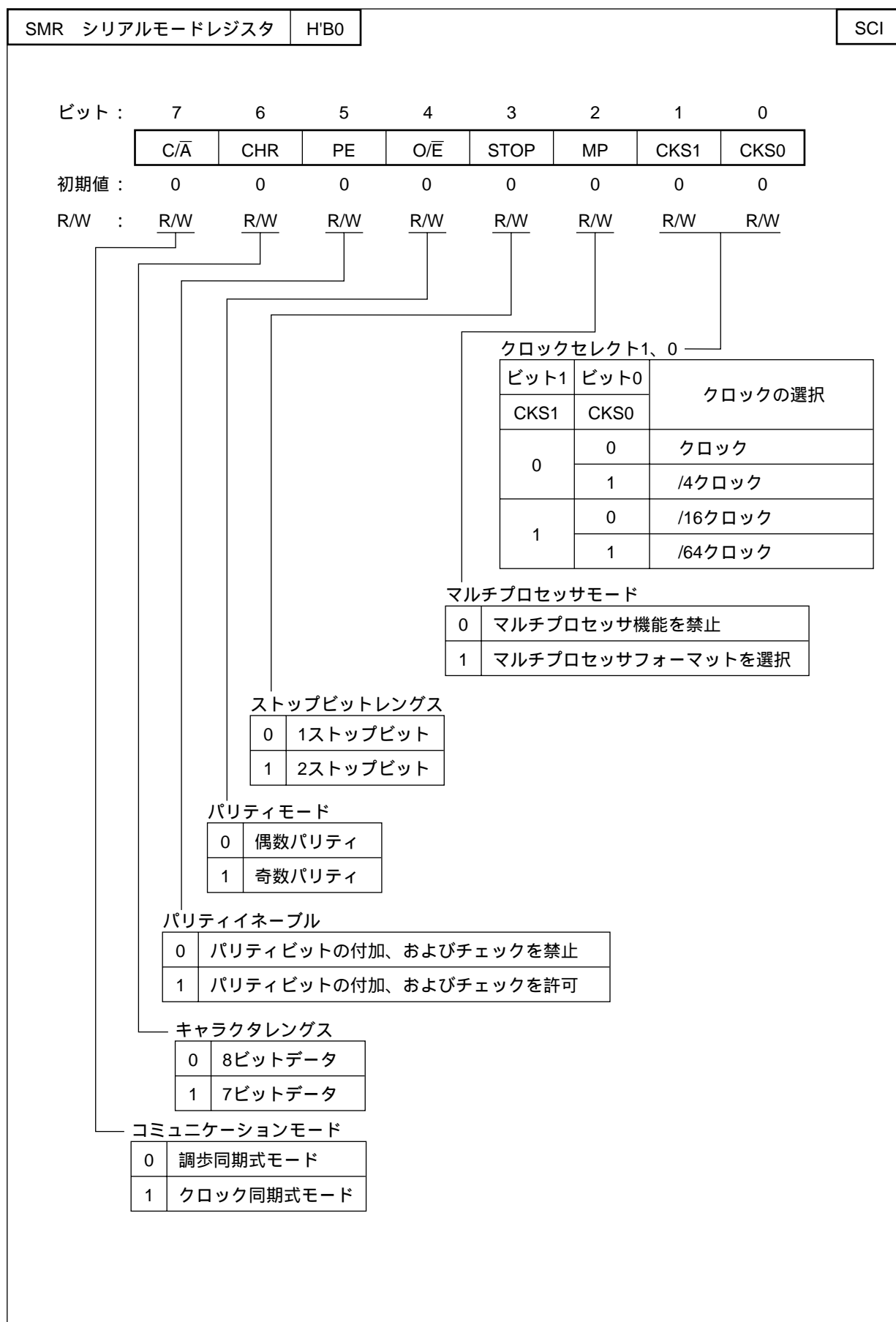
TPCR		TPC出力コントロールレジスタ				H'A1		TPC																																																																	
ビット:	7	6	5	4	3	2	1	0																																																																	
	G3CMS1	G3CMS0	G2CMS1	G2CMS0	G1CMS1	G1CMS0	G0CMS1	G0CMS0																																																																	
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																																																																	
<p>グループ0コンペアマッチセレクト1、0</p> <table border="1"> <thead> <tr> <th>ビット1</th> <th>ビット0</th> <th>出力トリガとなるITUのチャンネル選択</th> </tr> <tr> <th>G0CMS1</th> <th>G0CMS0</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td>0</td> <td>TPC出力グループ0 (TP₃~TP₀端子) の出力トリガはITUチャンネル0のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ0 (TP₃~TP₀端子) の出力トリガはITUチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>TPC出力グループ0 (TP₃~TP₀端子) の出力トリガはITUチャンネル2のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ0 (TP₃~TP₀端子) の出力トリガはITUチャンネル3のコンペアマッチ</td> </tr> </tbody> </table> <p>グループ1コンペアマッチセレクト1、0</p> <table border="1"> <thead> <tr> <th>ビット3</th> <th>ビット2</th> <th>出力トリガとなるITUのチャンネル選択</th> </tr> <tr> <th>G1CMS1</th> <th>G1CMS0</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td>0</td> <td>TPC出力グループ1 (TP₇~TP₄端子) の出力トリガはITUチャンネル0のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ1 (TP₇~TP₄端子) の出力トリガはITUチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>TPC出力グループ1 (TP₇~TP₄端子) の出力トリガはITUチャンネル2のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ1 (TP₇~TP₄端子) の出力トリガはITUチャンネル3のコンペアマッチ</td> </tr> </tbody> </table> <p>グループ2コンペアマッチセレクト1、0</p> <table border="1"> <thead> <tr> <th>ビット5</th> <th>ビット4</th> <th>出力トリガとなるITUのチャンネル選択</th> </tr> <tr> <th>G2CMS1</th> <th>G2CMS0</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td>0</td> <td>TPC出力グループ2 (TP₁₁~TP₈端子) の出力トリガはITUチャンネル0のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ2 (TP₁₁~TP₈端子) の出力トリガはITUチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>TPC出力グループ2 (TP₁₁~TP₈端子) の出力トリガはITUチャンネル2のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ2 (TP₁₁~TP₈端子) の出力トリガはITUチャンネル3のコンペアマッチ</td> </tr> </tbody> </table> <p>グループ3コンペアマッチセレクト1、0</p> <table border="1"> <thead> <tr> <th>ビット7</th> <th>ビット6</th> <th>出力トリガとなるITUのチャンネル選択</th> </tr> <tr> <th>G3CMS1</th> <th>G3CMS0</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2">0</td> <td>0</td> <td>TPC出力グループ3 (TP₁₅~TP₁₂端子) の出力トリガはITUチャンネル0のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ3 (TP₁₅~TP₁₂端子) の出力トリガはITUチャンネル1のコンペアマッチ</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>TPC出力グループ3 (TP₁₅~TP₁₂端子) の出力トリガはITUチャンネル2のコンペアマッチ</td> </tr> <tr> <td>1</td> <td>TPC出力グループ3 (TP₁₅~TP₁₂端子) の出力トリガはITUチャンネル3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ITUのチャンネル選択	G0CMS1	G0CMS0		0	0	TPC出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガはITUチャンネル0のコンペアマッチ	1	TPC出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガはITU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 ₃ ~TP ₀ 端子) の出力トリガはITUチャンネル2のコンペアマッチ	1	TPC出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガはITUチャンネル3のコンペアマッチ	ビット3	ビット2	出力トリガとなるITUのチャンネル選択	G1CMS1	G1CMS0		0	0	TPC出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガはITUチャンネル0のコンペアマッチ	1	TPC出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガはITU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 ₇ ~TP ₄ 端子) の出力トリガはITUチャンネル2のコンペアマッチ	1	TPC出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガはITUチャンネル3のコンペアマッチ	ビット5	ビット4	出力トリガとなるITUのチャンネル選択	G2CMS1	G2CMS0		0	0	TPC出力グループ2 (TP ₁₁ ~TP ₈ 端子) の出力トリガはITUチャンネル0のコンペアマッチ	1	TPC出力グループ2 (TP ₁₁ ~TP ₈ 端子) の出力トリガはITU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 ₁₁ ~TP ₈ 端子) の出力トリガはITUチャンネル2のコンペアマッチ	1	TPC出力グループ2 (TP ₁₁ ~TP ₈ 端子) の出力トリガはITUチャンネル3のコンペアマッチ	ビット7	ビット6	出力トリガとなるITUのチャンネル選択	G3CMS1	G3CMS0		0	0	TPC出力グループ3 (TP ₁₅ ~TP ₁₂ 端子) の出力トリガはITUチャンネル0のコンペアマッチ	1	TPC出力グループ3 (TP ₁₅ ~TP ₁₂ 端子) の出力トリガはITU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 ₁₅ ~TP ₁₂ 端子) の出力トリガはITUチャンネル2のコンペアマッチ	1	TPC出力グループ3 (TP ₁₅ ~TP ₁₂ 端子) の出力トリガはITUチャンネル3のコンペアマッチ
ビット1	ビット0	出力トリガとなるITUのチャンネル選択																																																																							
G0CMS1	G0CMS0																																																																								
0	0	TPC出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガはITUチャンネル0のコンペアマッチ																																																																							
	1	TPC出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガはITU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 ₃ ~TP ₀ 端子) の出力トリガはITUチャンネル2のコンペアマッチ																																																																							
	1	TPC出力グループ0 (TP ₃ ~TP ₀ 端子) の出力トリガはITUチャンネル3のコンペアマッチ																																																																							
ビット3	ビット2	出力トリガとなるITUのチャンネル選択																																																																							
G1CMS1	G1CMS0																																																																								
0	0	TPC出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガはITUチャンネル0のコンペアマッチ																																																																							
	1	TPC出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガはITU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 ₇ ~TP ₄ 端子) の出力トリガはITUチャンネル2のコンペアマッチ																																																																							
	1	TPC出力グループ1 (TP ₇ ~TP ₄ 端子) の出力トリガはITUチャンネル3のコンペアマッチ																																																																							
ビット5	ビット4	出力トリガとなるITUのチャンネル選択																																																																							
G2CMS1	G2CMS0																																																																								
0	0	TPC出力グループ2 (TP ₁₁ ~TP ₈ 端子) の出力トリガはITUチャンネル0のコンペアマッチ																																																																							
	1	TPC出力グループ2 (TP ₁₁ ~TP ₈ 端子) の出力トリガはITU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 ₁₁ ~TP ₈ 端子) の出力トリガはITUチャンネル2のコンペアマッチ																																																																							
	1	TPC出力グループ2 (TP ₁₁ ~TP ₈ 端子) の出力トリガはITUチャンネル3のコンペアマッチ																																																																							
ビット7	ビット6	出力トリガとなるITUのチャンネル選択																																																																							
G3CMS1	G3CMS0																																																																								
0	0	TPC出力グループ3 (TP ₁₅ ~TP ₁₂ 端子) の出力トリガはITUチャンネル0のコンペアマッチ																																																																							
	1	TPC出力グループ3 (TP ₁₅ ~TP ₁₂ 端子) の出力トリガはITU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 ₁₅ ~TP ₁₂ 端子) の出力トリガはITUチャンネル2のコンペアマッチ																																																																							
	1	TPC出力グループ3 (TP ₁₅ ~TP ₁₂ 端子) の出力トリガはITUチャンネル3のコンペアマッチ																																																																							

NDERB ネクストデータインーブルレジスタB	H'A2	TPC															
<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="padding: 2px 5px;">NDER15</td> <td style="padding: 2px 5px;">NDER14</td> <td style="padding: 2px 5px;">NDER13</td> <td style="padding: 2px 5px;">NDER12</td> <td style="padding: 2px 5px;">NDER11</td> <td style="padding: 2px 5px;">NDER10</td> <td style="padding: 2px 5px;">NDER9</td> <td style="padding: 2px 5px;">NDER8</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 style="text-align: center; margin-top: 10px;">ネクストデータインーブル15～8</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">ビット7～0</td> <td rowspan="2" style="padding: 5px; text-align: center;">説明</td> </tr> <tr> <td style="padding: 5px;">NDER15 ～ NDER8</td> </tr> <tr> <td style="padding: 5px; text-align: center;">0</td> <td style="padding: 5px;">TPC出力TP₁₅～TP₈を禁止 (NDR15～NDR8からPB₇～PB₀への転送禁止)</td> </tr> <tr> <td style="padding: 5px; text-align: center;">1</td> <td style="padding: 5px;">TPC出力TP₁₅～TP₈を許可 (NDR15～NDR8からPB₇～PB₀への転送許可)</td> </tr> </table>			NDER15	NDER14	NDER13	NDER12	NDER11	NDER10	NDER9	NDER8	ビット7～0	説明	NDER15 ～ NDER8	0	TPC出力TP ₁₅ ～TP ₈ を禁止 (NDR15～NDR8からPB ₇ ～PB ₀ への転送禁止)	1	TPC出力TP ₁₅ ～TP ₈ を許可 (NDR15～NDR8からPB ₇ ～PB ₀ への転送許可)
NDER15	NDER14	NDER13	NDER12	NDER11	NDER10	NDER9	NDER8										
ビット7～0	説明																
NDER15 ～ NDER8																	
0	TPC出力TP ₁₅ ～TP ₈ を禁止 (NDR15～NDR8からPB ₇ ～PB ₀ への転送禁止)																
1	TPC出力TP ₁₅ ～TP ₈ を許可 (NDR15～NDR8からPB ₇ ～PB ₀ への転送許可)																
NDERA ネクストデータインーブルレジスタA	H'A3	TPC															
<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="padding: 2px 5px;">NDER7</td> <td style="padding: 2px 5px;">NDER6</td> <td style="padding: 2px 5px;">NDER5</td> <td style="padding: 2px 5px;">NDER4</td> <td style="padding: 2px 5px;">NDER3</td> <td style="padding: 2px 5px;">NDER2</td> <td style="padding: 2px 5px;">NDER1</td> <td style="padding: 2px 5px;">NDER0</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 style="text-align: center; margin-top: 10px;">ネクストデータインーブル7～0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">ビット7～0</td> <td rowspan="2" style="padding: 5px; text-align: center;">説明</td> </tr> <tr> <td style="padding: 5px;">NDER7 ～ NDER0</td> </tr> <tr> <td style="padding: 5px; text-align: center;">0</td> <td style="padding: 5px;">TPC出力TP₇～TP₀を禁止 (NDR7～NDR0からPA₇～PA₀への転送禁止)</td> </tr> <tr> <td style="padding: 5px; text-align: center;">1</td> <td style="padding: 5px;">TPC出力TP₇～TP₀を許可 (NDR7～NDR0からPA₇～PA₀への転送許可)</td> </tr> </table>			NDER7	NDER6	NDER5	NDER4	NDER3	NDER2	NDER1	NDER0	ビット7～0	説明	NDER7 ～ NDER0	0	TPC出力TP ₇ ～TP ₀ を禁止 (NDR7～NDR0からPA ₇ ～PA ₀ への転送禁止)	1	TPC出力TP ₇ ～TP ₀ を許可 (NDR7～NDR0からPA ₇ ～PA ₀ への転送許可)
NDER7	NDER6	NDER5	NDER4	NDER3	NDER2	NDER1	NDER0										
ビット7～0	説明																
NDER7 ～ NDER0																	
0	TPC出力TP ₇ ～TP ₀ を禁止 (NDR7～NDR0からPA ₇ ～PA ₀ への転送禁止)																
1	TPC出力TP ₇ ～TP ₀ を許可 (NDR7～NDR0からPA ₇ ～PA ₀ への転送許可)																

NDRB	ネクストデータレジスタB	H'A4/H'A6	TPC					
TPC出力グループ2、3の出力トリガが同一の場合								
(1) アドレス : H'FFA4								
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
	TPC出力グループ3の次の 出力データを格納				TPC出力グループ2の次の 出力データを格納			
(2) アドレス : H'FFA6								
ビット :	7	6	5	4	3	2	1	0
初期値 :	1	1	1	1	1	1	1	1
R/W :								
TPC出力グループ2、3の出力トリガが異なる場合								
(1) アドレス : H'FFA4								
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				
	TPC出力グループ3の次の 出力データを格納							
(2) アドレス : H'FFA6								
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
					TPC出力グループ2の次の 出力データを格納			

NDR A ネクストデータレジスタA	H'A5/H'A7	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'FFA5</p> <p>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;">NDR7</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR6</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR5</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR4</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR3</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR2</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR1</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR0</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-around; margin-top: 10px;"> <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> <p>(2) アドレス : H'FFA7</p> <p>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: 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;"></td> </tr> </table> <p>初期値 : 1 1 1 1 1 1 1 1</p> <p>R/W : </p> <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'FFA5</p> <p>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;">NDR7</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR6</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR5</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR4</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;"></td> </tr> </table> <p>初期値 : 0 0 0 0 1 1 1 1</p> <p>R/W : R/W R/W R/W R/W</p> <div style="display: flex; justify-content: center; margin-top: 10px;"> <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> <p>(2) アドレス : H'FFA7</p> <p>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: 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; text-align: center;">NDR3</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR2</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR1</td> <td style="border: 1px solid black; width: 20px; text-align: center;">NDR0</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="display: flex; justify-content: center; margin-top: 10px;"> <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>			NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0									NDR7	NDR6	NDR5	NDR4									NDR3	NDR2	NDR1	NDR0
NDR7	NDR6	NDR5	NDR4	NDR3	NDR2	NDR1	NDR0																											
NDR7	NDR6	NDR5	NDR4																															
				NDR3	NDR2	NDR1	NDR0																											

TCNT タイマカウンタ	H'A9リード時、H'A8ライト時	WDT																				
<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> <p style="text-align: center; margin-top: 10px;">└───┬───┘ 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'A9リード時、H'A8ライト時	WDT																				
<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; text-align: center;">WRST</td> <td style="width: 20px; height: 20px; text-align: center;">RSTOE</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 1 1 1 1 1 1</p> <p>R/W : R/(W)* R/W</p> <div style="margin-top: 10px;"> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <th colspan="2" style="text-align: center;">リセット出力イネーブル</th> </tr> <tr> <td style="width: 20px; text-align: center;">0</td> <td>リセット信号を外部に出力しない</td> </tr> <tr> <td style="width: 20px; text-align: center;">1</td> <td>リセット信号を外部に出力する</td> </tr> </table> </div> <div style="margin-top: 20px;"> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <th colspan="2" style="text-align: center;">ウォッチドッグタイマリセット</th> </tr> <tr> <td style="width: 20px; text-align: center;">0</td> <td>〔クリア条件〕 RES端子によるリセット信号またはソフトウェアによる"0"クリア</td> </tr> <tr> <td style="width: 20px; 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> </div>			WRST	RSTOE							リセット出力イネーブル		0	リセット信号を外部に出力しない	1	リセット信号を外部に出力する	ウォッチドッグタイマリセット		0	〔クリア条件〕 RES端子によるリセット信号またはソフトウェアによる"0"クリア	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																					
リセット出力イネーブル																						
0	リセット信号を外部に出力しない																					
1	リセット信号を外部に出力する																					
ウォッチドッグタイマリセット																						
0	〔クリア条件〕 RES端子によるリセット信号またはソフトウェアによる"0"クリア																					
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>																						



BBR ビットレートレジスタ	H'B1		SCI														
ビット :	7	6	5	4	3	2	1	0									
	<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> </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									
<div style="display: flex; justify-content: center; align-items: center; margin-bottom: 5px;"> <div style="border-left: 1px solid black; border-right: 1px solid black; width: 1px; height: 10px; margin-right: 5px;"></div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; width: 100%; height: 1px; margin-right: 5px;"></div> </div> <p>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'B2			SCI																														
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																														
クロックイネーブル1、0 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>ビット2</th> <th>ビット1</th> <th colspan="2">クロックの選択、出力の許可</th> </tr> <tr> <th>CKE1</th> <th>CKE2</th> <th></th> <th></th> </tr> </thead> <tbody> <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> </tbody> </table>									ビット2	ビット1	クロックの選択、出力の許可		CKE1	CKE2			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
ビット2	ビット1	クロックの選択、出力の許可																																				
CKE1	CKE2																																					
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																																			
トランスミットエンドインタラプトイネーブル <table border="1" style="margin-left: 20px;"> <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>									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	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																																					
マルチプロセッサインタラプトイネーブル <table border="1" style="margin-left: 20px;"> <tr> <td>0</td> <td>マルチプロセッサ割込みを禁止 (通常 of 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>									0	マルチプロセッサ割込みを禁止 (通常 of 受信動作を行う)	1	マルチプロセッサ割込みを許可																										
0	マルチプロセッサ割込みを禁止 (通常 of 受信動作を行う)																																					
1	マルチプロセッサ割込みを許可																																					
レシーブイネーブル <table border="1" style="margin-left: 20px;"> <tr> <td colspan="2">トランスミットイネーブル</td> <td colspan="2">レシーブイネーブル</td> </tr> <tr> <td>0</td> <td>送信動作を禁止</td> <td>0</td> <td>受信動作を禁止</td> </tr> <tr> <td>1</td> <td>送信動作を許可</td> <td>1</td> <td>受信動作を許可</td> </tr> </table>									トランスミットイネーブル		レシーブイネーブル		0	送信動作を禁止	0	受信動作を禁止	1	送信動作を許可	1	受信動作を許可																		
トランスミットイネーブル		レシーブイネーブル																																				
0	送信動作を禁止	0	受信動作を禁止																																			
1	送信動作を許可	1	受信動作を許可																																			
レシーブインタラプトイネーブル <table border="1" style="margin-left: 20px;"> <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>									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	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																																					
トランスミットインタラプトイネーブル <table border="1" style="margin-left: 20px;"> <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	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																										
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'B3		SCI									
<p>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: 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;"></td> <td style="border: 1px solid black; width: 20px; height: 20px;"></td> </tr> </table> <p>初期値 : 1 1 1 1 1 1 1 1</p> <p>R/W : <u>R/W</u> 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>シリアル送信データを格納</p> </div>												

SSR シリアルステータスレジスタ	H'B4	SCI								
ビット :	7 6 5 4 3 2 1 0									
	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 12.5%; text-align: center;">TDRE</td> <td style="width: 12.5%; text-align: center;">RDRF</td> <td style="width: 12.5%; text-align: center;">ORER</td> <td style="width: 12.5%; text-align: center;">FER</td> <td style="width: 12.5%; text-align: center;">PER</td> <td style="width: 12.5%; text-align: center;">TEND</td> <td style="width: 12.5%; text-align: center;">MPB</td> <td style="width: 12.5%; text-align: center;">MPBT</td> </tr> </table>	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
TDRE	RDRF	ORER	FER	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									
<table border="1" style="width: 100%; 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="width: 20%; 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="width: 100%; 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="width: 20%; 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									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">トランスミットエンド</td> </tr> <tr> <td style="width: 20%; text-align: center;">0</td> <td>〔クリア条件〕 TDRE = "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〕 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>			トランスミットエンド		0	〔クリア条件〕 TDRE = "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〕 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										
0	〔クリア条件〕 TDRE = "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〕 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									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">パリティエラー</td> </tr> <tr> <td style="width: 20%; 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O\bar{E}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>			パリティエラー		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O \bar{E} ビットで設定したパリティと一致しなかったとき)		
パリティエラー										
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O \bar{E} ビットで設定したパリティと一致しなかったとき)									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">フレーミングエラー</td> </tr> <tr> <td style="width: 20%; 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>			フレーミングエラー		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)		
フレーミングエラー										
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)									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">オーバランエラー</td> </tr> <tr> <td style="width: 20%; 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"の状態で次のデータが受信完了したとき)</td> </tr> </table>			オーバランエラー		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"の状態で次のデータが受信完了したとき)		
オーバランエラー										
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"の状態で次のデータが受信完了したとき)									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">レシーブデータレジスタフル</td> </tr> <tr> <td style="width: 20%; 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</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へデータが転送されたとき</td> </tr> </table>			レシーブデータレジスタフル		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	1	〔セット条件〕 データが正常に受信され、RSRからRDRへデータが転送されたとき		
レシーブデータレジスタフル										
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									
1	〔セット条件〕 データが正常に受信され、RSRからRDRへデータが転送されたとき									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">トランスミットデータレジスタEMPTY</td> </tr> <tr> <td style="width: 20%; text-align: center;">0</td> <td>〔クリア条件〕 TDRE = "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〕 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>			トランスミットデータレジスタEMPTY		0	〔クリア条件〕 TDRE = "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〕 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		
トランスミットデータレジスタEMPTY										
0	〔クリア条件〕 TDRE = "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〕 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】 * フラグをクリアするための"0"ライトのみ可能です。</p>										

RDR レシーブデータレジスタ	H'B5	SCI								
<p>ビット : 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse; width: 60%;"> <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 R R R R R R R</p>									<p>↓</p> <p>シリアル受信データを格納</p>	

P1DDR ポート1データディレクションレジスタ							H'C0		ポート1					
ビット :	7	6	5	4	3	2	1	0						
	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR						
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入出力選択 <table border="1" style="margin: auto;"> <tr> <td>0</td> <td>入力ポート</td> </tr> <tr> <td>1</td> <td>出力ポート</td> </tr> </table>											0	入力ポート	1	出力ポート
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'C1		ポート2					
ビット :	7	6	5	4	3	2	1	0						
	P27DDR	P26DDR	P25DDR	P24DDR	P23DDR	P22DDR	P21DDR	P20DDR						
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入出力選択 <table border="1" style="margin: auto;"> <tr> <td>0</td> <td>入力ポート</td> </tr> <tr> <td>1</td> <td>出力ポート</td> </tr> </table>											0	入力ポート	1	出力ポート
0	入力ポート													
1	出力ポート													

P1DR ポート1データレジスタ	H'C2	ポート1												
<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="padding: 2px 5px;">P17</td> <td style="padding: 2px 5px;">P16</td> <td style="padding: 2px 5px;">P15</td> <td style="padding: 2px 5px;">P14</td> <td style="padding: 2px 5px;">P13</td> <td style="padding: 2px 5px;">P12</td> <td style="padding: 2px 5px;">P11</td> <td style="padding: 2px 5px;">P10</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 style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">ポート1の各端子のデータを格納</p>			P17	P16	P15	P14	P13	P12	P11	P10				
P17	P16	P15	P14	P13	P12	P11	P10							
P2DR ポート2データレジスタ	H'C3	ポート2												
<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="padding: 2px 5px;">P27</td> <td style="padding: 2px 5px;">P26</td> <td style="padding: 2px 5px;">P25</td> <td style="padding: 2px 5px;">P24</td> <td style="padding: 2px 5px;">P23</td> <td style="padding: 2px 5px;">P22</td> <td style="padding: 2px 5px;">P21</td> <td style="padding: 2px 5px;">P20</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 style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">ポート2の各端子のデータを格納</p>			P27	P26	P25	P24	P23	P22	P21	P20				
P27	P26	P25	P24	P23	P22	P21	P20							
P3DDR ポート3データディレクションレジスタ	H'C4	ポート3												
<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="padding: 2px 5px;">P37DDR</td> <td style="padding: 2px 5px;">P36DDR</td> <td style="padding: 2px 5px;">P35DDR</td> <td style="padding: 2px 5px;">P34DDR</td> <td style="padding: 2px 5px;">P33DDR</td> <td style="padding: 2px 5px;">P32DDR</td> <td style="padding: 2px 5px;">P31DDR</td> <td style="padding: 2px 5px;">P30DDR</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> <p style="text-align: center; margin-top: 10px;">↓</p> <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; text-align: center;"> <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>			P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR	0	入力ポート	1	出力ポート
P37DDR	P36DDR	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR							
0	入力ポート													
1	出力ポート													

P3DR ポート3データレジスタ	H'C6	ポート3												
<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;">P3₇</td> <td style="width: 20px;">P3₆</td> <td style="width: 20px;">P3₅</td> <td style="width: 20px;">P3₄</td> <td style="width: 20px;">P3₃</td> <td style="width: 20px;">P3₂</td> <td style="width: 20px;">P3₁</td> <td style="width: 20px;">P3₀</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 style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">ポート3の各端子のデータを移動</p>			P3 ₇	P3 ₆	P3 ₅	P3 ₄	P3 ₃	P3 ₂	P3 ₁	P3 ₀				
P3 ₇	P3 ₆	P3 ₅	P3 ₄	P3 ₃	P3 ₂	P3 ₁	P3 ₀							
P5DDR ポート5データディレクションレジスタ	H'C8	ポート5												
<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;"></td> <td style="width: 20px;"></td> <td style="width: 20px;"></td> <td style="width: 20px;"></td> <td style="width: 20px;">P5₃DDR</td> <td style="width: 20px;">P5₂DDR</td> <td style="width: 20px;">P5₁DDR</td> <td style="width: 20px;">P5₀DDR</td> </tr> </table> <p>初期値： 1 1 1 1 0 0 0 0</p> <p>R/W : W W W W</p> <p style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">ポート5入出力選択</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">0</td> <td>入力ポート</td> </tr> <tr> <td>1</td> <td>出力ポート</td> </tr> </table>							P5 ₃ DDR	P5 ₂ DDR	P5 ₁ DDR	P5 ₀ DDR	0	入力ポート	1	出力ポート
				P5 ₃ DDR	P5 ₂ DDR	P5 ₁ DDR	P5 ₀ DDR							
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'C9	ポート6												
<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;"></td> <td style="width: 20px;"></td> <td style="width: 20px;">P6₅DDR</td> <td style="width: 20px;">P6₄DDR</td> <td style="width: 20px;">P6₃DDR</td> <td style="width: 20px;"></td> <td style="width: 20px;"></td> <td style="width: 20px;">P6₀DDR</td> </tr> </table> <p>初期値： 1 0 0 0 0 0 0 0</p> <p>R/W : W W W W W W W</p> <p style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">ポート6入出力選択</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse; text-align: center;"> <tr> <td style="width: 20px;">0</td> <td>入力ポート</td> </tr> <tr> <td>1</td> <td>出力ポート</td> </tr> </table>					P6 ₅ DDR	P6 ₄ DDR	P6 ₃ DDR			P6 ₀ DDR	0	入力ポート	1	出力ポート
		P6 ₅ DDR	P6 ₄ DDR	P6 ₃ DDR			P6 ₀ DDR							
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'CD		ポート8					
ビット :	7	6	5	4	3	2	1	0				
					P8 ₃ DDR	P8 ₂ DDR	P8 ₁ DDR	P8 ₀ DDR				
初期値 :	1	1	1	0	0	0	0	0				
R/W :				W	W	W	W	W				
					ポート8入出力選択 <table border="1"> <tr> <td>0</td> <td>入力ポート</td> </tr> <tr> <td>1</td> <td>出力ポート</td> </tr> </table>				0	入力ポート	1	出力ポート
0	入力ポート											
1	出力ポート											
P7DR ポート7データレジスタ					H'CE		ポート7					
ビット :	7	6	5	4	3	2	1	0				
	P7 ₇	P7 ₆	P7 ₅	P7 ₄	P7 ₃	P7 ₂	P7 ₁	P7 ₀				
初期値 :	*	*	*	*	*	*	*	*				
R/W :	R	R	R	R	R	R	R	R				
					ポート7の各端子の状態を讀出す							
【注】 * P7 ₇ ~P7 ₀ 端子により決定されます。												

P9DR ポート9データレジスタ	H'D2	ポート9												
<p>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: 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;">P9₄</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;">P9₂</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;">P9₀</td> </tr> </table> <p>初期値： 1 1 0 0 0 0 0 0</p> <p>R/W : R/W R/W R/W R/W R/W</p> <div style="text-align: center; margin-top: 10px;"> <p>ポート9の各端子のデータを格納</p> </div>						P9 ₄		P9 ₂		P9 ₀				
			P9 ₄		P9 ₂		P9 ₀							
PADR ポートAデータレジスタ	H'D3	ポートA												
<p>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: 20px; text-align: center;">PA₇</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PA₆</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PA₅</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PA₄</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PA₃</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PA₂</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PA₁</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PA₀</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-top: 10px;"> <p>ポートAの各端子のデータを格納</p> </div>			PA ₇	PA ₆	PA ₅	PA ₄	PA ₃	PA ₂	PA ₁	PA ₀				
PA ₇	PA ₆	PA ₅	PA ₄	PA ₃	PA ₂	PA ₁	PA ₀							
P8DDR ポートBデータディレクションレジスタ	H'D4	ポートB												
<p>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: 20px; text-align: center;">PB₇DDR</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PB₆DDR</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PB₅DDR</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PB₄DDR</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PB₃DDR</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PB₂DDR</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PB₁DDR</td> <td style="border: 1px solid black; width: 20px; height: 20px; text-align: center;">PB₀DDR</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-top: 10px;"> <p>ポートB入出力選択</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">0</td> <td style="border: 1px solid black; padding: 2px;">入力ポート</td> </tr> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">1</td> <td style="border: 1px solid black; padding: 2px;">出力ポート</td> </tr> </table> </div>			PB ₇ DDR	PB ₆ DDR	PB ₅ DDR	PB ₄ DDR	PB ₃ DDR	PB ₂ DDR	PB ₁ DDR	PB ₀ DDR	0	入力ポート	1	出力ポート
PB ₇ DDR	PB ₆ DDR	PB ₅ DDR	PB ₄ DDR	PB ₃ DDR	PB ₂ DDR	PB ₁ DDR	PB ₀ DDR							
0	入力ポート													
1	出力ポート													

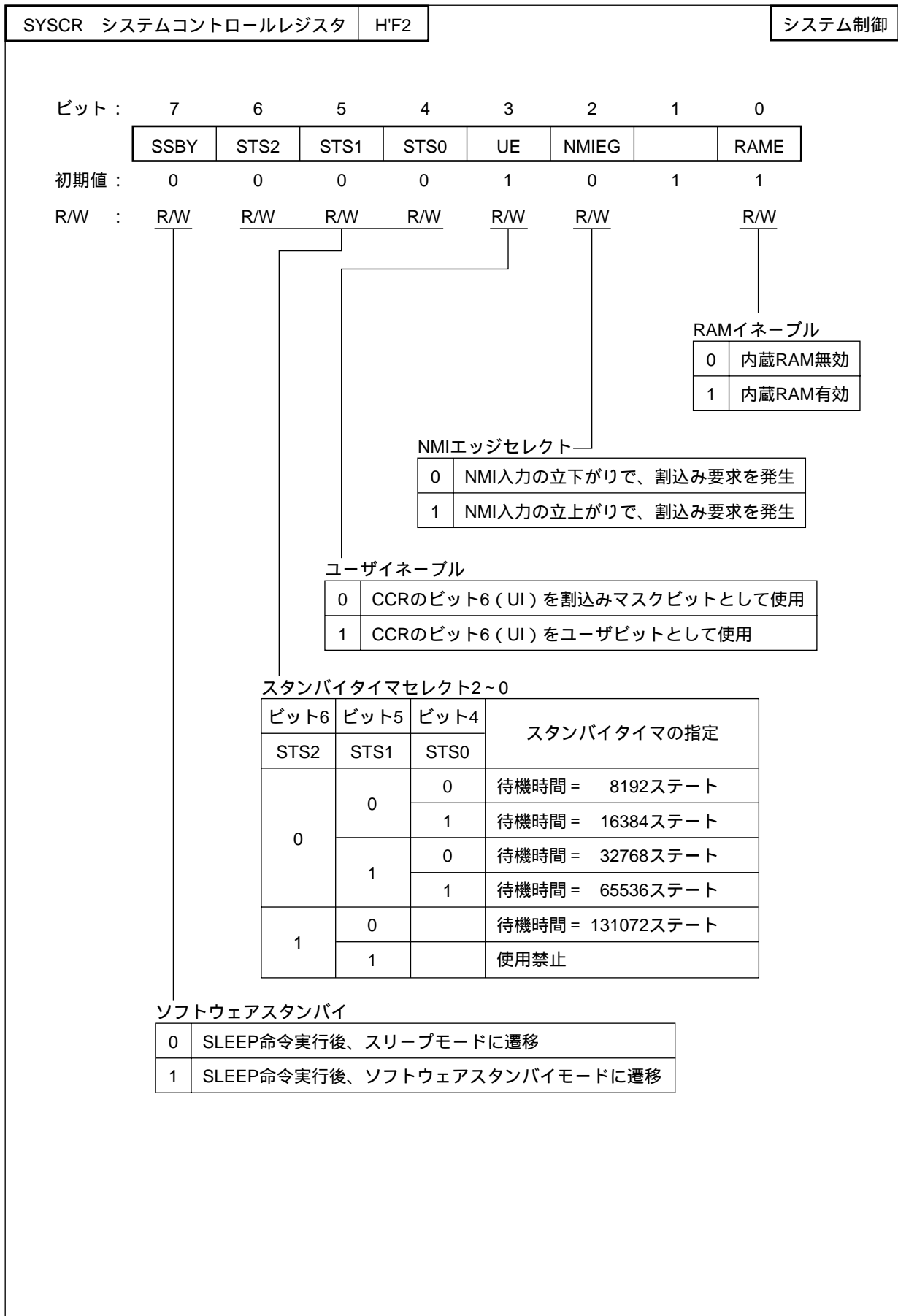
PBDR ポートBデータレジスタ	H'D6	ポートB												
<p>ビット： 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="padding: 2px;">PB₇</td> <td style="padding: 2px;">PB₆</td> <td style="padding: 2px;">PB₅</td> <td style="padding: 2px;">PB₄</td> <td style="padding: 2px;">PB₃</td> <td style="padding: 2px;">PB₂</td> <td style="padding: 2px;">PB₁</td> <td style="padding: 2px;">PB₀</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 style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">ポートBの各端子のデータを格納</p>	PB ₇	PB ₆	PB ₅	PB ₄	PB ₃	PB ₂	PB ₁	PB ₀						
PB ₇	PB ₆	PB ₅	PB ₄	PB ₃	PB ₂	PB ₁	PB ₀							
P2PCR ポート2入力プルアップMOSコントロールレジスタ	H'D8	ポート2												
<p>ビット： 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="padding: 2px;">P2₇PCR</td> <td style="padding: 2px;">P2₆PCR</td> <td style="padding: 2px;">P2₅PCR</td> <td style="padding: 2px;">P2₄PCR</td> <td style="padding: 2px;">P2₃PCR</td> <td style="padding: 2px;">P2₂PCR</td> <td style="padding: 2px;">P2₁PCR</td> <td style="padding: 2px;">P2₀PCR</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 style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">ポート2入力プルアップMOSコントロール7~0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px; width: 30px;">0</td> <td style="padding: 2px;">入力プルアップMOSはOFF</td> </tr> <tr> <td style="padding: 2px;">1</td> <td style="padding: 2px;">入力プルアップMOSはON</td> </tr> </table> <p style="text-align: center; margin-top: 10px;">P2DDRを"0"に指定したとき（入力ポートに指定）</p>	P2 ₇ PCR	P2 ₆ PCR	P2 ₅ PCR	P2 ₄ PCR	P2 ₃ PCR	P2 ₂ PCR	P2 ₁ PCR	P2 ₀ PCR	0	入力プルアップMOSはOFF	1	入力プルアップMOSはON		
P2 ₇ PCR	P2 ₆ PCR	P2 ₅ PCR	P2 ₄ PCR	P2 ₃ PCR	P2 ₂ PCR	P2 ₁ PCR	P2 ₀ PCR							
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'DB	ポート5												
<p>ビット： 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;">P5₃PCR</td> <td style="padding: 2px;">P5₂PCR</td> <td style="padding: 2px;">P5₁PCR</td> <td style="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> <p style="text-align: center; margin-top: 10px;">↓</p> <p style="text-align: center;">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-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px; width: 30px;">0</td> <td style="padding: 2px;">入力プルアップMOSはOFF</td> </tr> <tr> <td style="padding: 2px;">1</td> <td style="padding: 2px;">入力プルアップMOSはON</td> </tr> </table> <p style="text-align: center; margin-top: 10px;">P5DDRを"0"に指定したとき（入力ポートに指定）</p>					P5 ₃ PCR	P5 ₂ PCR	P5 ₁ PCR	P5 ₀ PCR	0	入力プルアップMOSはOFF	1	入力プルアップMOSはON		
				P5 ₃ PCR	P5 ₂ PCR	P5 ₁ PCR	P5 ₀ PCR							
0	入力プルアップMOSはOFF													
1	入力プルアップMOSはON													

ADDR A H, L	A/DデータレジスタA H, L	H'E0、H'E1	A/D																
ビット : 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 <table border="1" style="width: 100%; text-align: center;"> <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>				AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0						
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																			
<div style="display: flex; justify-content: space-around; width: 100%;"> ADDRAH ADDRAL </div>																			
<hr style="width: 100%;"/> <div style="display: flex; justify-content: center; align-items: center;"> <div style="border-left: 1px solid black; border-bottom: 1px solid black; width: 100px; height: 100px; margin-right: 10px;"></div> <div style="text-align: left;"> <p>A/D変換データ</p> <p>A/D変換結果の10ビット</p> <p>データを格納</p> </div> </div>																			

ADDR B H, L	A/DデータレジスタB H, L	H'E2、H'E3	A/D																
ビット : 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 <table border="1" style="width: 100%; text-align: center;"> <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>				AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0						
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																			
<div style="display: flex; justify-content: space-around; width: 100%;"> ADDRBH ADDRBL </div>																			
<hr style="width: 100%;"/> <div style="display: flex; justify-content: center; align-items: center;"> <div style="border-left: 1px solid black; border-bottom: 1px solid black; width: 100px; height: 100px; margin-right: 10px;"></div> <div style="text-align: left;"> <p>A/D変換データ</p> <p>A/D変換結果の10ビット</p> <p>データを格納</p> </div> </div>																			

ADDRCH、L A/DデータレジスタCH、L	H'E4、 H'E5		A/D																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table style="width: 100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">AD9</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD8</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD7</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD6</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD5</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD4</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD3</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD2</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD1</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD0</td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; 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 R R R R R R R R R R R R R R R</p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> └── ADDRCH └── ADDRCL </div> <div style="margin-top: 20px; text-align: center;"> <p>└── A/D変換データ</p> <p>A/D変換結果の10ビット</p> <p>データを格納</p> </div>				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'E6、 H'E7		A/D																
<p>ビット： 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</p> <table style="width: 100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">AD9</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD8</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD7</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD6</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD5</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD4</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD3</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD2</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD1</td> <td style="border: 1px solid black; width: 20px; text-align: center;">AD0</td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; 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 R R R R R R R R R R R R R R R</p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> └── ADDRDH └── ADDRDL </div> <div style="margin-top: 20px; text-align: center;"> <p>└── A/D変換データ</p> <p>A/D変換結果の10ビット</p> <p>データを格納</p> </div>				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'E9		A/D																
<p>ビット： 7 6 5 4 3 2 1 0</p> <table style="width: 100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">TRGE</td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> <td style="border: 1px solid black; width: 20px;"></td> </tr> </table> <p>初期値： 0 1 1 1 1 1 1 1</p> <p>R/W： R/W</p> <div style="margin-top: 10px;"> <p>トリガイネーブル</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">0</td> <td>外部トリガ入力による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/A/D変換を開始</td> </tr> </table> </div>				TRGE								0	外部トリガ入力によるA/D変換の開始を禁止	1	外部トリガ端子 (ADTRG) の立下がり/A/D変換を開始				
TRGE																			
0	外部トリガ入力によるA/D変換の開始を禁止																		
1	外部トリガ端子 (ADTRG) の立下がり/A/D変換を開始																		

ASTCR アクセスステートコントロールレジスタ	H'ED	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: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">AST7</td> <td style="width: 20px; text-align: center;">AST6</td> <td style="width: 20px; text-align: center;">AST5</td> <td style="width: 20px; text-align: center;">AST4</td> <td style="width: 20px; text-align: center;">AST3</td> <td style="width: 20px; text-align: center;">AST2</td> <td style="width: 20px; text-align: center;">AST1</td> <td style="width: 20px; text-align: center;">AST0</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 style="text-align: center; margin-top: 10px;">エリア7~0アクセスステートコントロール</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">ビット7~0</td> <td rowspan="4" style="text-align: center; vertical-align: middle;">アクセスステート数の指定</td> </tr> <tr> <td style="text-align: center;">AST7 ~AST0</td> </tr> <tr> <td style="text-align: center;">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 style="text-align: center;">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> </table>	AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0	ビット7~0	アクセスステート数の指定	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																									
AST7	AST6	AST5	AST4	AST3	AST2	AST1	AST0																																	
ビット7~0	アクセスステート数の指定																																							
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																																						
<p>ビット： 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: 40px; 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;">WMS1</td> <td style="width: 20px; text-align: center;">WMS0</td> <td style="width: 20px; text-align: center;">WC1</td> <td style="width: 20px; text-align: center;">WC0</td> </tr> </table> <p>初期値： 1 1 1 1 0 0 1 1</p> <p>R/W : R/W R/W R/W R/W</p> <p style="text-align: center; margin-top: 10px;">ウェイトカウント1、0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">ビット1</td> <td style="width: 20px; text-align: center;">ビット0</td> <td rowspan="4" style="text-align: center; vertical-align: middle;">ウェイトステート数の指定</td> </tr> <tr> <td style="text-align: center;">WC1</td> <td style="text-align: center;">WC0</td> </tr> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>WSCによるウェイトを禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>1ステート挿入</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>2ステート挿入</td> </tr> <tr> <td style="text-align: center;">1</td> <td>3ステート挿入</td> </tr> </table> <p style="text-align: center; margin-top: 10px;">ウェイトモードセレクト1、0</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 20px; text-align: center;">ビット3</td> <td style="width: 20px; text-align: center;">ビット2</td> <td rowspan="4" style="text-align: center; vertical-align: middle;">ウェイトモードの指定</td> </tr> <tr> <td style="text-align: center;">WMS1</td> <td style="text-align: center;">WMS0</td> </tr> <tr> <td rowspan="2" style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>プログラマブルウェイトモード</td> </tr> <tr> <td style="text-align: center;">1</td> <td>WSCによるウェイトを禁止</td> </tr> <tr> <td rowspan="2" style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>端子ウェイトモード1</td> </tr> <tr> <td style="text-align: center;">1</td> <td>端子オートウェイトモード</td> </tr> </table>					WMS1	WMS0	WC1	WC0	ビット1	ビット0	ウェイトステート数の指定	WC1	WC0	0	0	WSCによるウェイトを禁止	1	1ステート挿入	1	0	2ステート挿入	1	3ステート挿入	ビット3	ビット2	ウェイトモードの指定	WMS1	WMS0	0	0	プログラマブルウェイトモード	1	WSCによるウェイトを禁止	1	0	端子ウェイトモード1	1	端子オートウェイトモード		バスコントローラ
				WMS1	WMS0	WC1	WC0																																	
ビット1	ビット0	ウェイトステート数の指定																																						
WC1	WC0																																							
0	0		WSCによるウェイトを禁止																																					
	1		1ステート挿入																																					
1	0	2ステート挿入																																						
	1	3ステート挿入																																						
ビット3	ビット2	ウェイトモードの指定																																						
WMS1	WMS0																																							
0	0		プログラマブルウェイトモード																																					
	1		WSCによるウェイトを禁止																																					
1	0	端子ウェイトモード1																																						
	1	端子オートウェイトモード																																						



ADRCR アドレスコントロールレジスタ	H'F3	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%;">A₂₃E</td> <td style="width: 10%;">A₂₂E</td> <td style="width: 10%;">A₂₁E</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>	A ₂₃ E	A ₂₂ E	A ₂₁ E						
A ₂₃ E	A ₂₂ E	A ₂₁ E								
モード1,3 {	初期値 :	1 1 1 1 1 1 1 0								
	R/W :	R/W								
モード2 {	初期値 :	0 0 0 0 1 0 1 0								
	R/W :	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;">R/W</td> <td style="width: 10%;">R/W</td> <td style="width: 10%;">R/W</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%;">R/W</td> </tr> </table>	R/W	R/W	R/W					R/W
R/W	R/W	R/W					R/W			
		PA ₆ ~ PA ₄ (A ₂₃ ~ A ₂₁) アドレス出力選択 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10%;">0</td> <td style="width: 10%;">アドレス出力</td> </tr> <tr> <td style="width: 10%;">1</td> <td style="width: 10%;">上記以外の入出力端子</td> </tr> </table>	0	アドレス出力	1	上記以外の入出力端子				
0	アドレス出力									
1	上記以外の入出力端子									

ISCR IRQセンスコントロールレジスタ	H'F4	割り込みコントローラ													
<p>ビット : 7 6 5 4 3 2 1 0</p> <table style="width: 100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%; border: 1px solid black;"></td> <td style="width: 10%; border: 1px solid black;"></td> <td style="width: 10%; border: 1px solid black;"></td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ4SC</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ3SC</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ2SC</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ1SC</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ0SC</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="text-align: center; margin: 10px 0;"> <p>↓</p> <p>IRQ₄ ~ IRQ₀センスコントロール</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>IRQ₄ ~ 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>IRQ₄ ~ IRQ₀入力の立下がりエッジで割り込み要求を発生</td> </tr> </table> </div>							IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC	0	IRQ ₄ ~ IRQ ₀ 入力の"Low"レベルで割り込み要求を発生	1	IRQ ₄ ~ IRQ ₀ 入力の立下がりエッジで割り込み要求を発生
				IRQ4SC	IRQ3SC	IRQ2SC	IRQ1SC	IRQ0SC							
0	IRQ ₄ ~ IRQ ₀ 入力の"Low"レベルで割り込み要求を発生														
1	IRQ ₄ ~ IRQ ₀ 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'F5	割り込みコントローラ													
<p>ビット : 7 6 5 4 3 2 1 0</p> <table style="width: 100%; border-collapse: collapse; margin-bottom: 5px;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%; border: 1px solid black;"></td> <td style="width: 10%; border: 1px solid black;"></td> <td style="width: 10%; border: 1px solid black;"></td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ4E</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ3E</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ2E</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ1E</td> <td style="width: 10%; border: 1px solid black; text-align: center;">IRQ0E</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="text-align: center; margin: 10px 0;"> <p>↓</p> <p>IRQ₄ ~ IRQ₀イネーブル</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 30px; text-align: center;">0</td> <td>IRQ₄ ~ IRQ₀割り込みを禁止</td> </tr> <tr> <td style="text-align: center;">1</td> <td>IRQ₄ ~ IRQ₀割り込みを許可</td> </tr> </table> </div>							IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E	0	IRQ ₄ ~ IRQ ₀ 割り込みを禁止	1	IRQ ₄ ~ IRQ ₀ 割り込みを許可
				IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E							
0	IRQ ₄ ~ IRQ ₀ 割り込みを禁止														
1	IRQ ₄ ~ IRQ ₀ 割り込みを許可														

ISR IRQステータスレジスタ			H'F6		割り込みコントローラ			
ビット :	7	6	5	4	3	2	1	0
				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)*	R/(W)*	R/(W)*
IRQ ₄ ~ IRQ ₀ フラグ								
ビット4~0	セット/クリア条件							
IRQ4F ~ IRQ0F								
0	[クリア条件] (1) IRQ _n F = "1"の状態ではIRQ _n Fをリードした後、IRQ _n Fに"0"をライトしたとき (2) IRQ _n SC = "0"、 $\overline{\text{IRQ}}_n$ 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) IRQ _n SC = "1"の状態ではIRQ _n 割り込み例外処理を実行したとき							
1	[セット条件] (1) IRQ _n SC = "0"の状態では $\overline{\text{IRQ}}_n$ 入力が"Low"レベルになったとき (2) IRQ _n SC = "1"の状態では $\overline{\text{IRQ}}_n$ 入力に立下がりエッジが発生したとき							
(n = 4 ~ 0)								
【注】 * フラグをクリアするための"0"ライトのみ可能です。								

IPRA インタラプトプライオリティレジスタA	H'F8	割り込みコントローラ																																																	
<p>ビット : 7 6 5 4 3 2 1 0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">IPRA7</td> <td style="padding: 2px;">IPRA6</td> <td style="padding: 2px;">IPRA5</td> <td style="padding: 2px;">IPRA4</td> <td style="padding: 2px;">IPRA3</td> <td style="padding: 2px;">IPRA2</td> <td style="padding: 2px;">IPRA1</td> <td style="padding: 2px;">IPRA0</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-top: 10px;"> <p>プライオリティレベルA7~A0</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">0</td> <td style="padding: 2px;">プライオリティレベル0 (非優先)</td> </tr> <tr> <td style="padding: 2px;">1</td> <td style="padding: 2px;">プライオリティレベル1 (優先)</td> </tr> </table> </div> <p style="margin-top: 20px;">割り込み要因と各ビットの対応</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th rowspan="2">IPRA</th> <th rowspan="2">ビット</th> <th>ビット7</th> <th>ビット6</th> <th>ビット5</th> <th>ビット4</th> <th>ビット3</th> <th>ビット2</th> <th>ビット1</th> <th>ビット0</th> </tr> <tr> <th>IPRA7</th> <th>IPRA6</th> <th>IPRA5</th> <th>IPRA4</th> <th>IPRA3</th> <th>IPRA2</th> <th>IPRA1</th> <th>IPRA0</th> </tr> </thead> <tbody> <tr> <td rowspan="2">IPRA</td> <td>割り込み</td> <td>IRQ₀</td> <td>IRQ₁</td> <td>IRQ₂、</td> <td>IRQ₄</td> <td>WDT</td> <td>ITU</td> <td>ITU</td> <td>ITU</td> </tr> <tr> <td>要因</td> <td></td> <td></td> <td>IRQ₃</td> <td></td> <td></td> <td>チャンネル0</td> <td>チャンネル1</td> <td>チャンネル2</td> </tr> </tbody> </table>			IPRA7	IPRA6	IPRA5	IPRA4	IPRA3	IPRA2	IPRA1	IPRA0	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)	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	IPRA	割り込み	IRQ ₀	IRQ ₁	IRQ ₂ 、	IRQ ₄	WDT	ITU	ITU	ITU	要因			IRQ ₃			チャンネル0	チャンネル1	チャンネル2
IPRA7	IPRA6	IPRA5	IPRA4	IPRA3	IPRA2	IPRA1	IPRA0																																												
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)																																																		
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																																										
IPRA	割り込み	IRQ ₀	IRQ ₁	IRQ ₂ 、	IRQ ₄	WDT	ITU	ITU	ITU																																										
	要因			IRQ ₃			チャンネル0	チャンネル1	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: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">IPRB7</td> <td style="padding: 2px;">IPRB6</td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;">IPRB3</td> <td style="padding: 2px;"></td> <td style="padding: 2px;">IPRB1</td> <td style="padding: 2px;"></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-top: 10px;"> <p>プライオリティレベルB7、B6、B3、B1</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">0</td> <td style="padding: 2px;">プライオリティレベル0 (非優先)</td> </tr> <tr> <td style="padding: 2px;">1</td> <td style="padding: 2px;">プライオリティレベル1 (優先)</td> </tr> </table> </div> <p style="margin-top: 20px;">割り込み要因と各ビットの対応</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th rowspan="2">IPRB</th> <th rowspan="2">ビット</th> <th>ビット7</th> <th>ビット6</th> <th>ビット5</th> <th>ビット4</th> <th>ビット3</th> <th>ビット2</th> <th>ビット1</th> <th>ビット0</th> </tr> <tr> <th>IPRB7</th> <th>IPRB6</th> <th></th> <th></th> <th>IPRB3</th> <th></th> <th>IPRB1</th> <th></th> </tr> </thead> <tbody> <tr> <td rowspan="2">IPRB</td> <td>割り込み</td> <td>ITU</td> <td>ITU</td> <td></td> <td></td> <td>SCI</td> <td></td> <td>A/D</td> <td></td> </tr> <tr> <td>要因</td> <td>チャンネル3</td> <td>チャンネル4</td> <td></td> <td></td> <td></td> <td></td> <td>変換器</td> <td></td> </tr> </tbody> </table>			IPRB7	IPRB6			IPRB3		IPRB1		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)	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			IPRB3		IPRB1		IPRB	割り込み	ITU	ITU			SCI		A/D		要因	チャンネル3	チャンネル4					変換器	
IPRB7	IPRB6			IPRB3		IPRB1																																													
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)																																																		
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			IPRB3		IPRB1																																											
IPRB	割り込み	ITU	ITU			SCI		A/D																																											
	要因	チャンネル3	チャンネル4					変換器																																											

C. I/Oポートブロック図

C.1 ポート1ブロック図

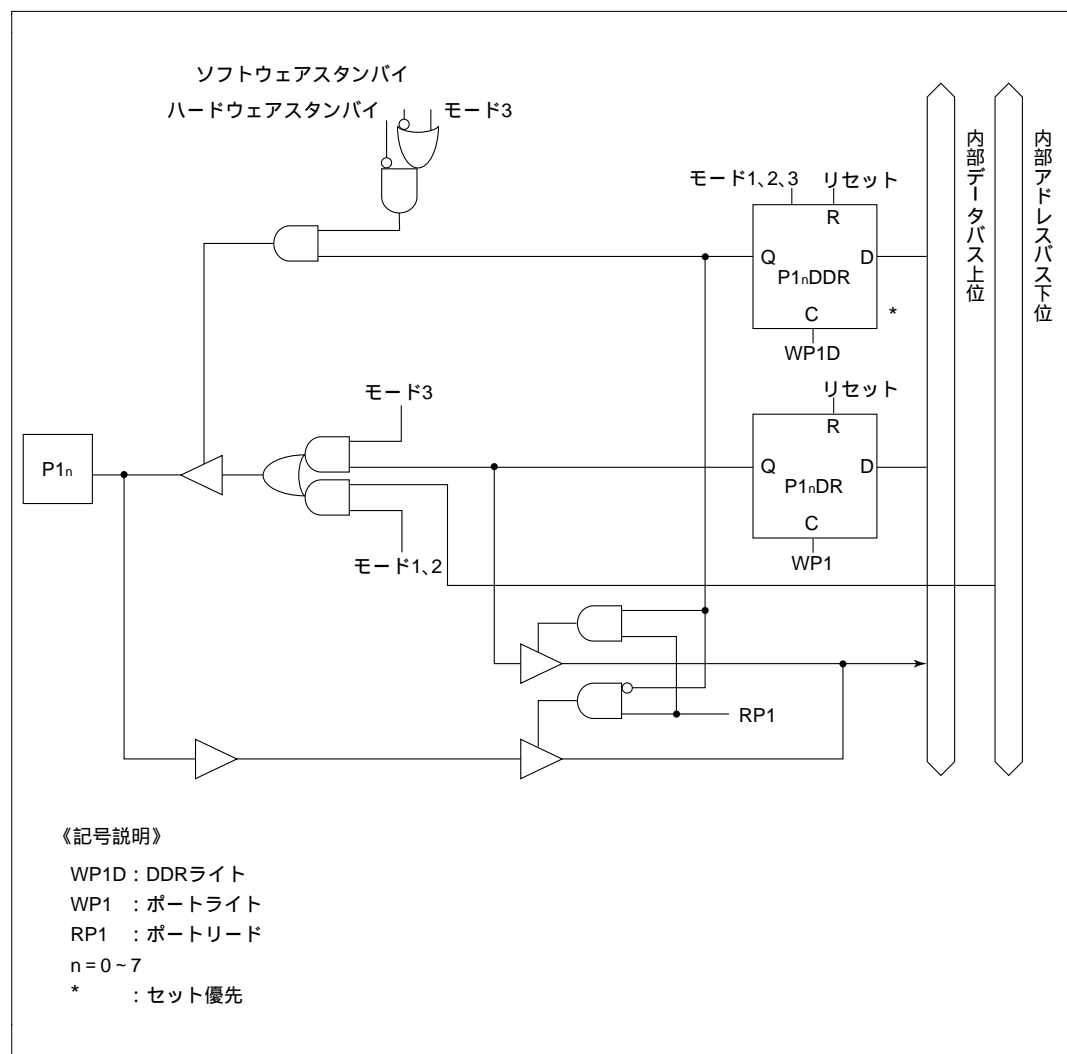


図 C.1 ポート1ブロック図

C.2 ポート2ブロック図

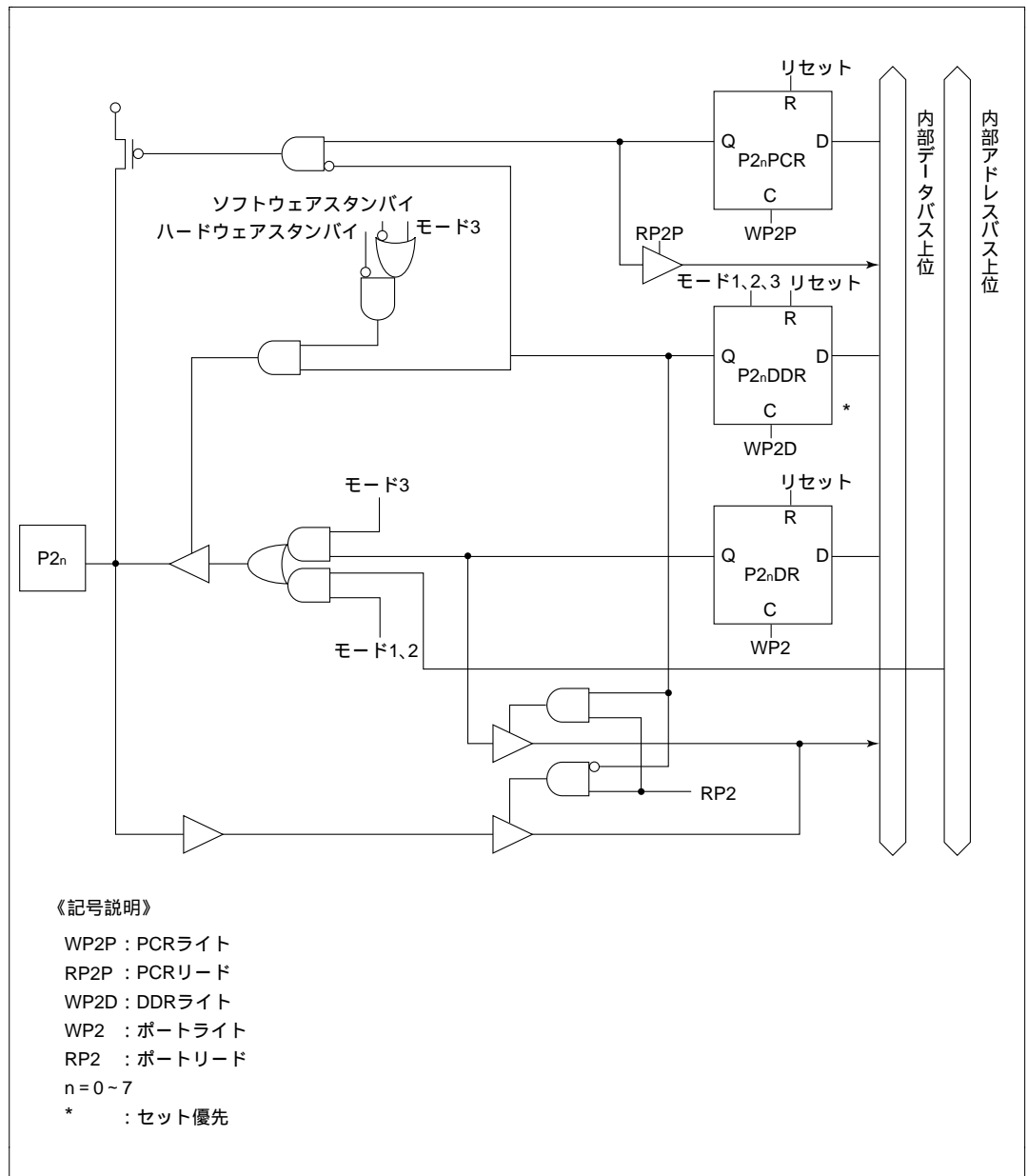


図 C.2 ポート2ブロック図

C.3 ポート3ブロック図

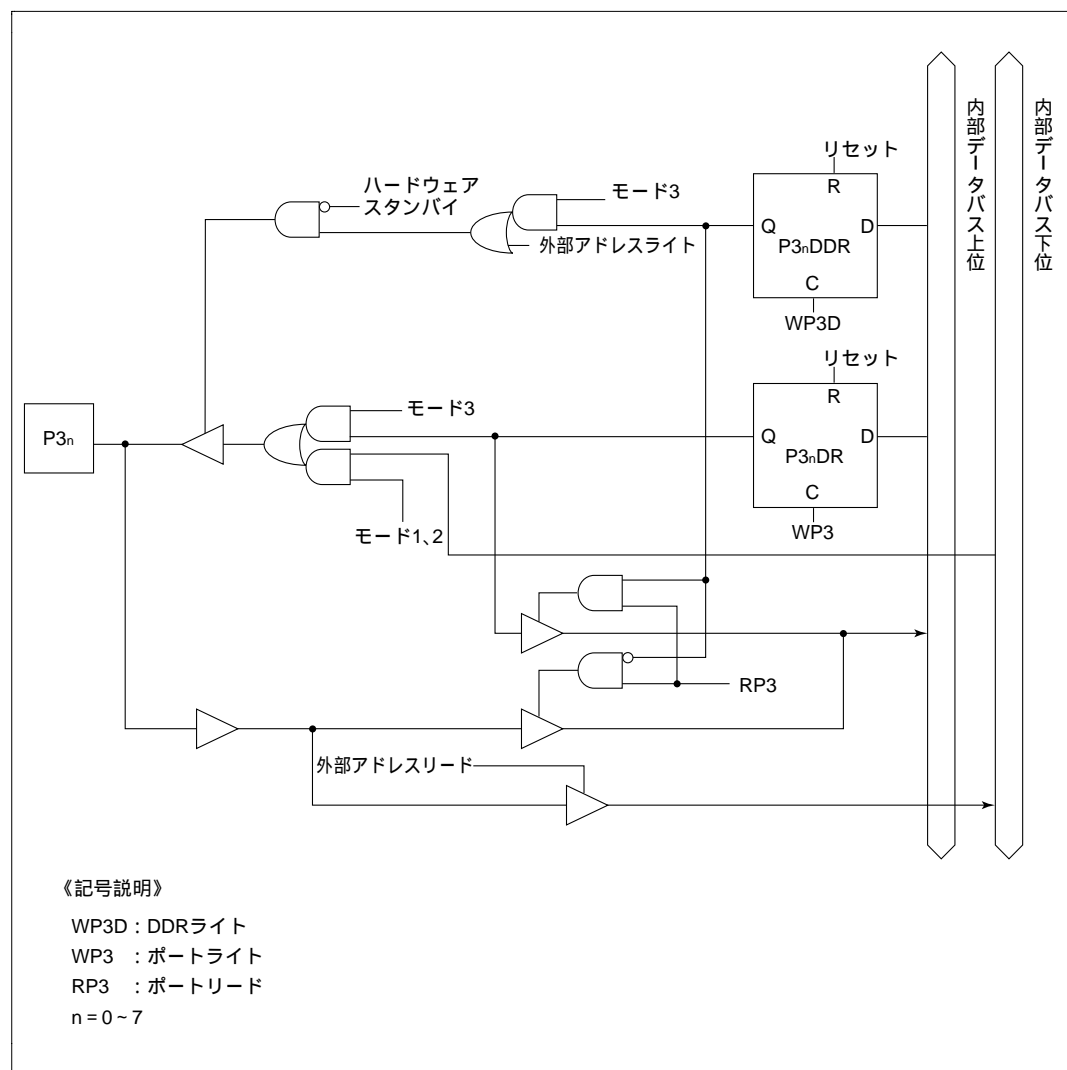


図 C.3 ポート3ブロック図

C.4 ポート5ブロック図

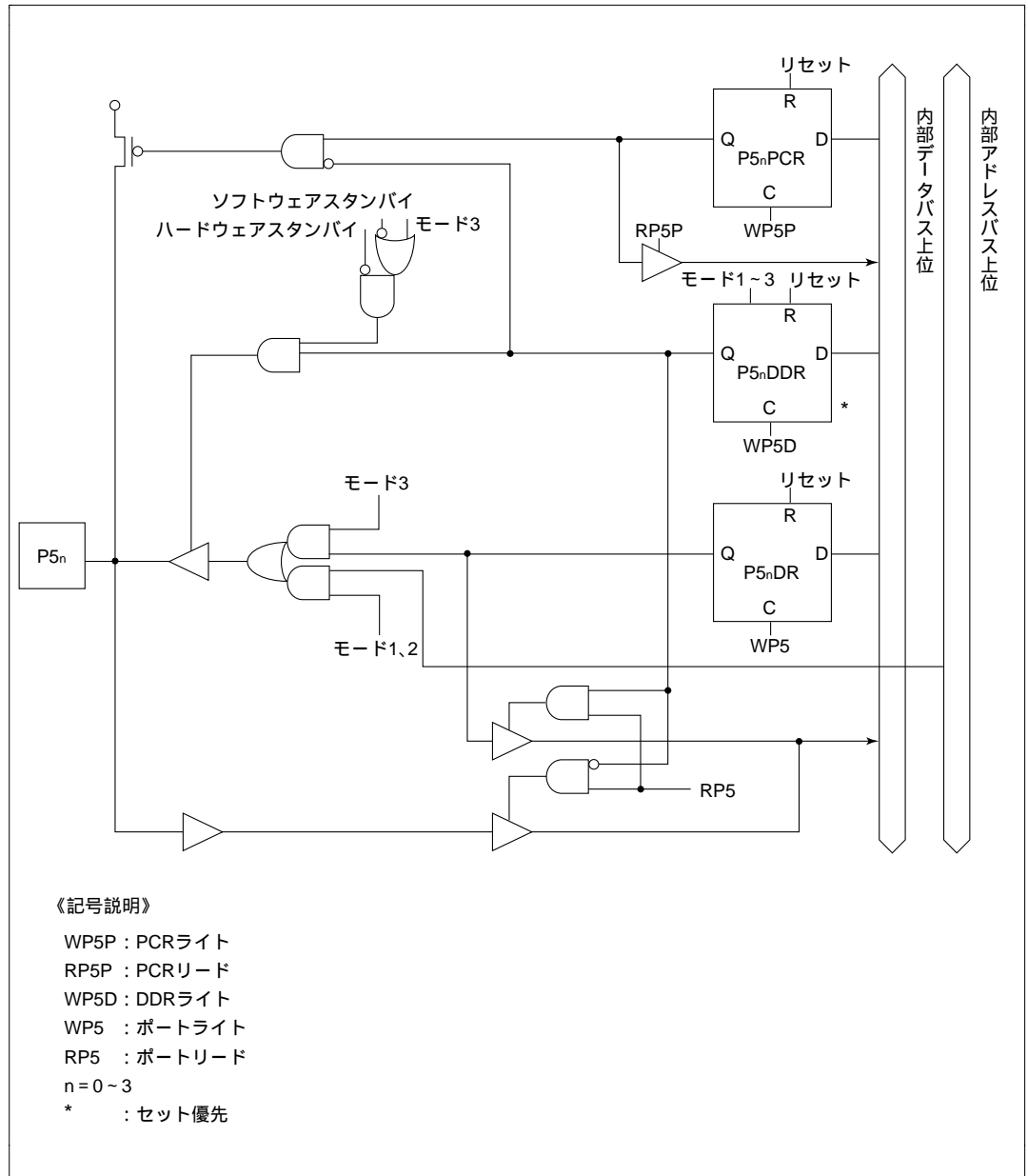


図 C.4 ポート5ブロック図

C.5 ポート6ブロック図

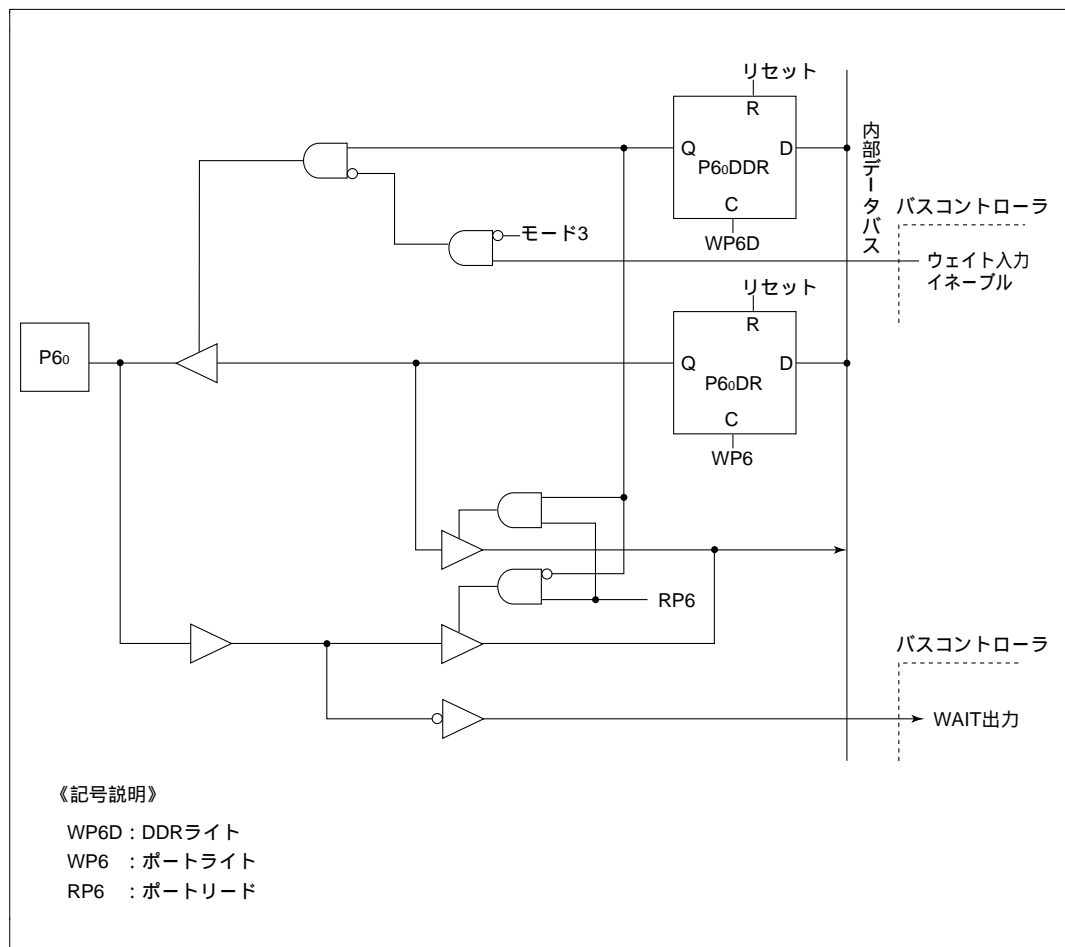


図 C.5 (a) ポート6ブロック図 (P6₀端子)

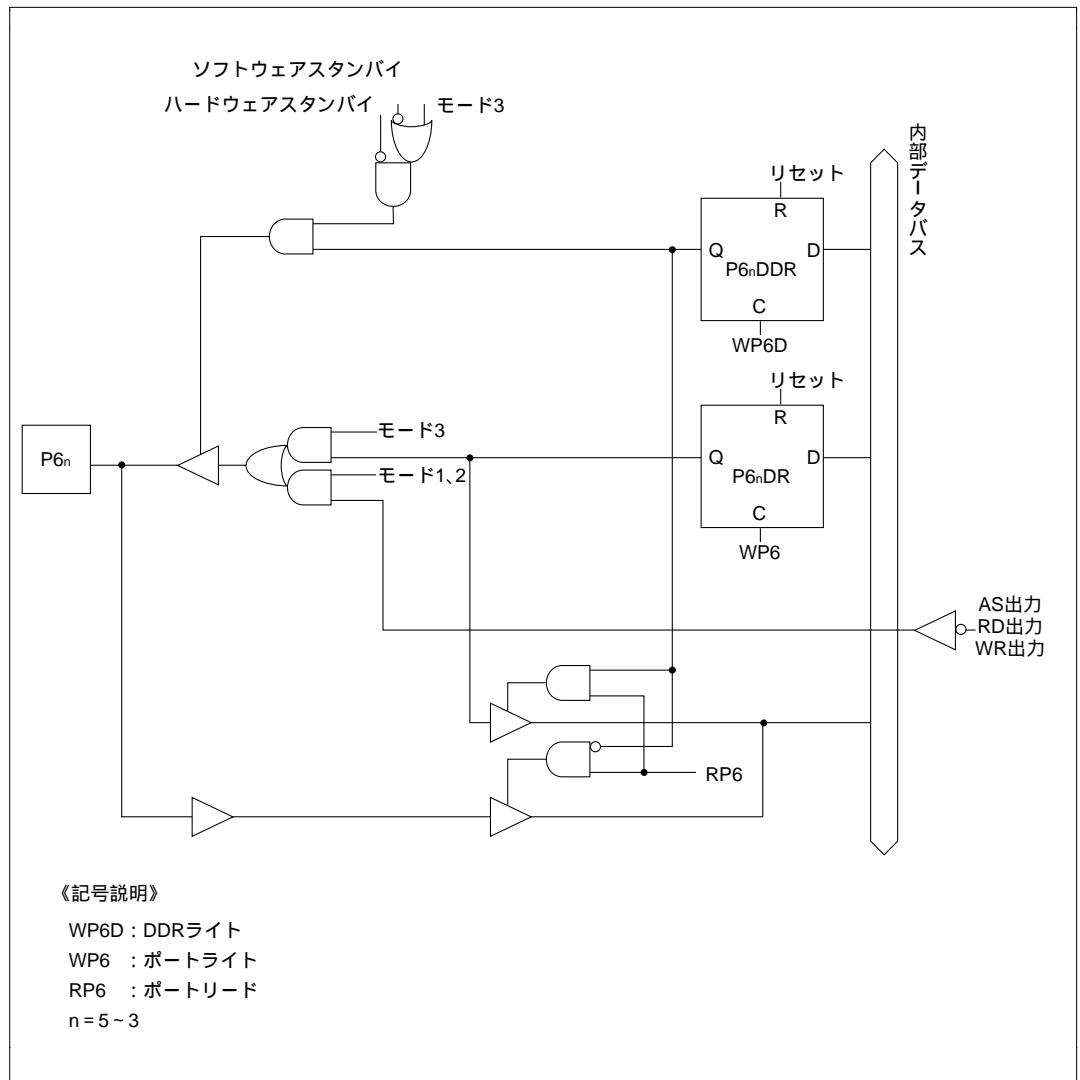


図 C.5 (b) ポート6 ブロック図 (P6₅ ~ P6₃端子)

C.6 ポート7ブロック図

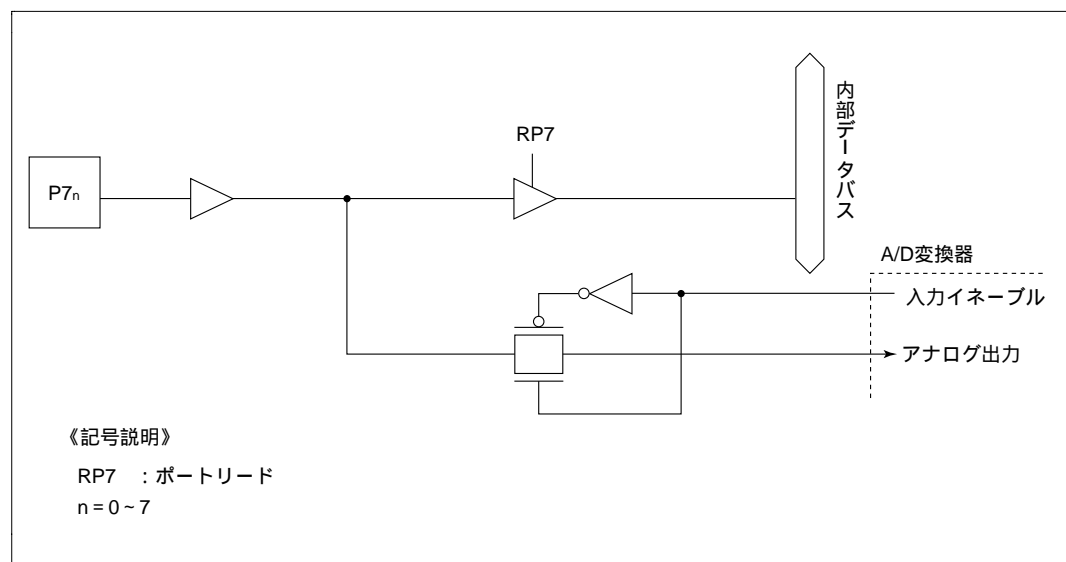


図 C.6 ポート7ブロック図

C.7 ポート 8 ブロック図

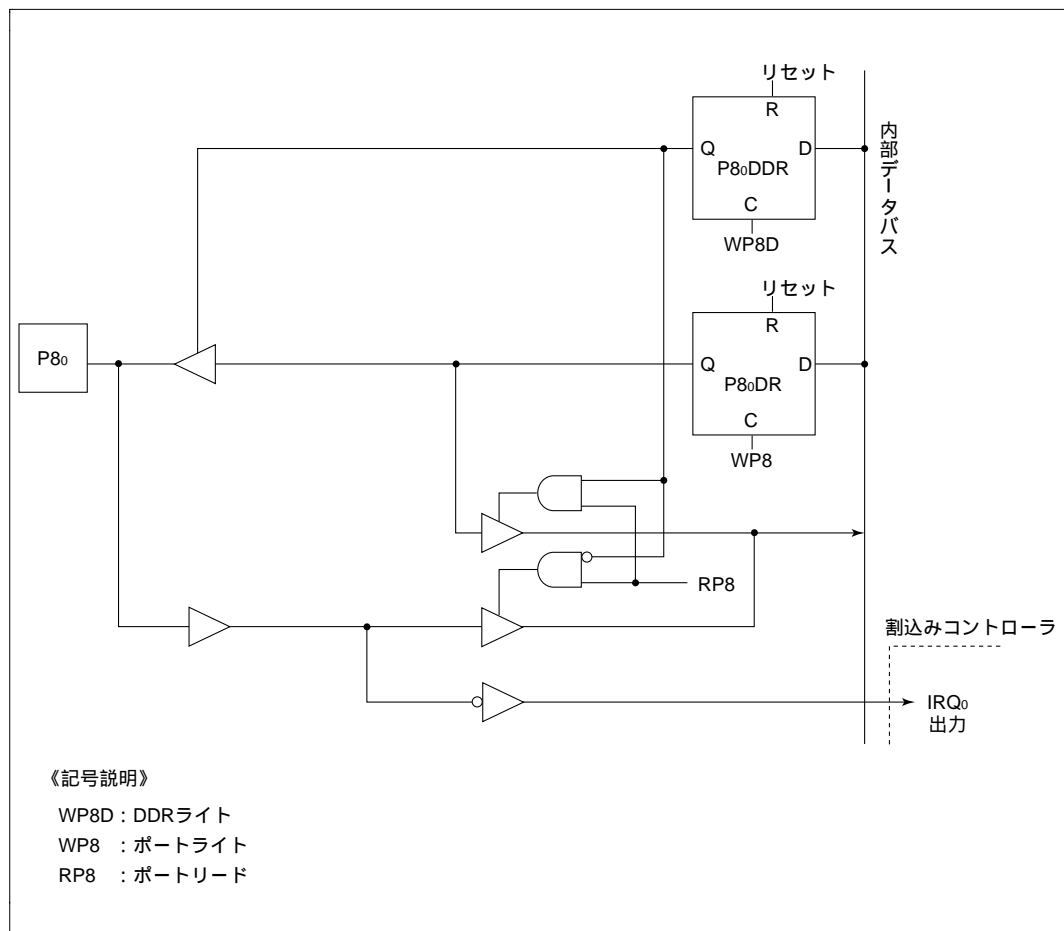


図 C.7 (a) ポート 8 ブロック図 (P8₀端子)

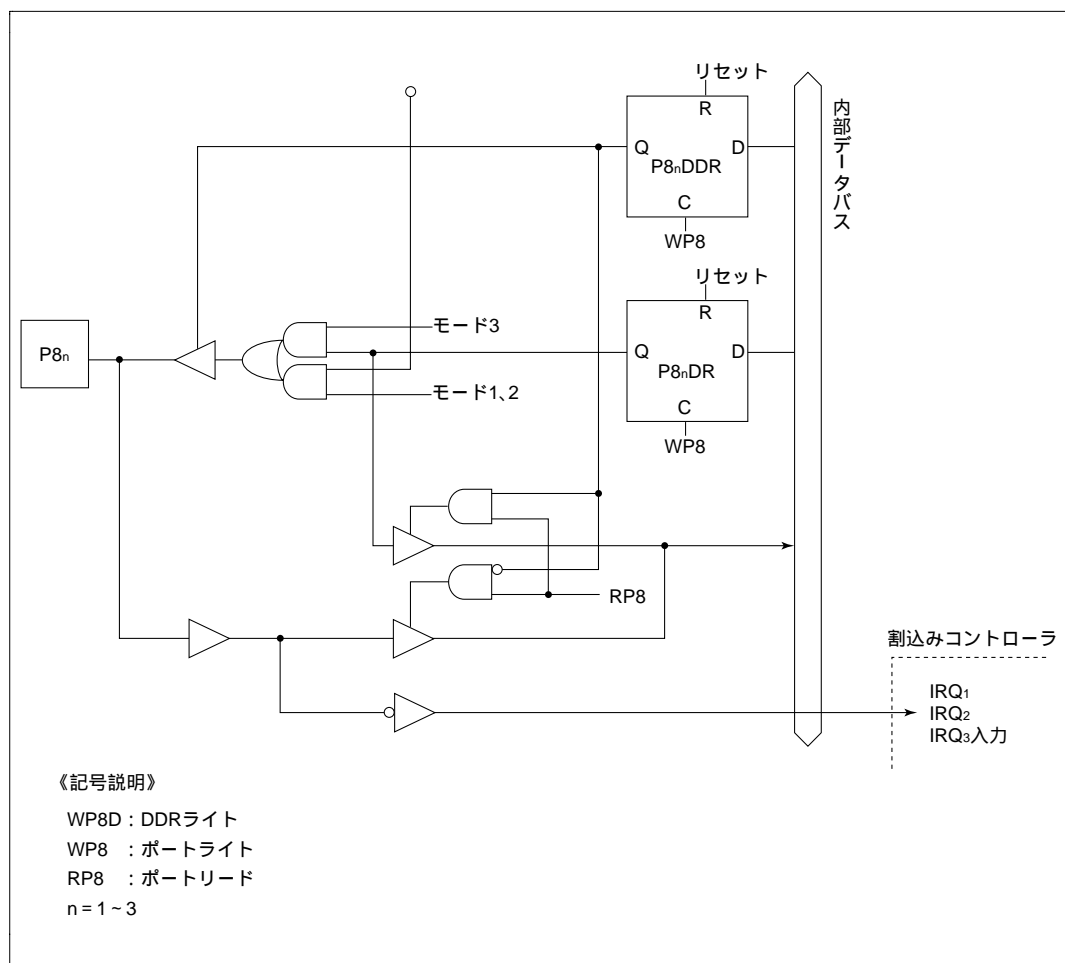


図 C.7 (b) ポート 8 ブロック図 (P8₁、P8₂、P8₃端子)

C.8 ポート9ブロック図

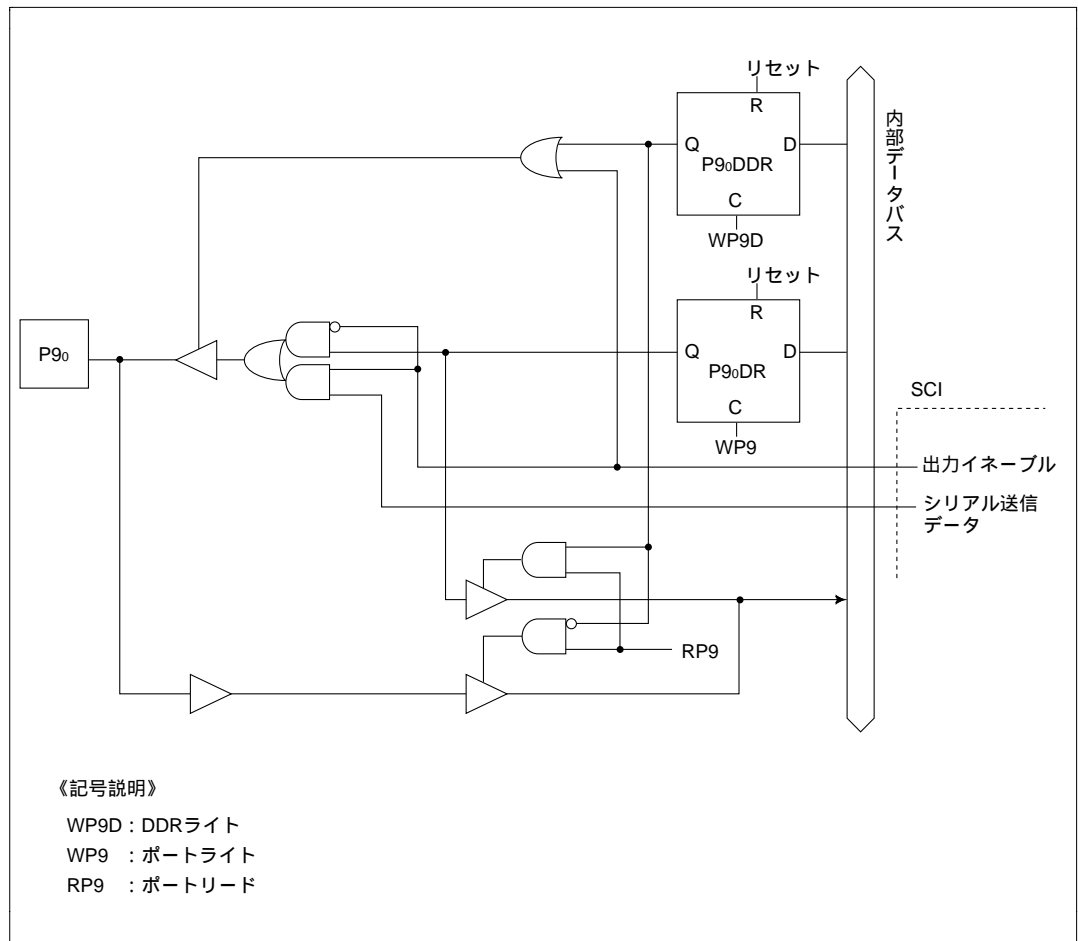


図 C.8 (a) ポート9ブロック図 (P9₀端子)

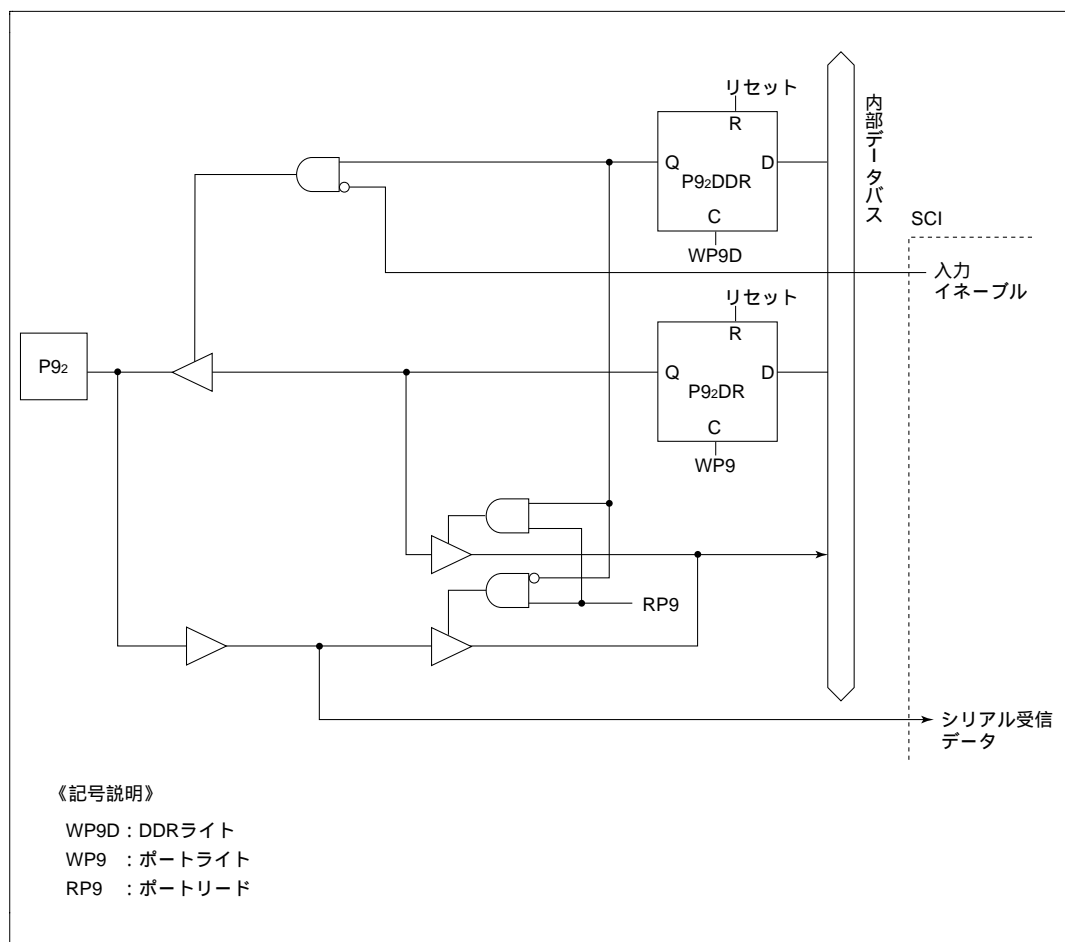


図 C.8 (b) ポート9ブロック図 (P9₂端子)

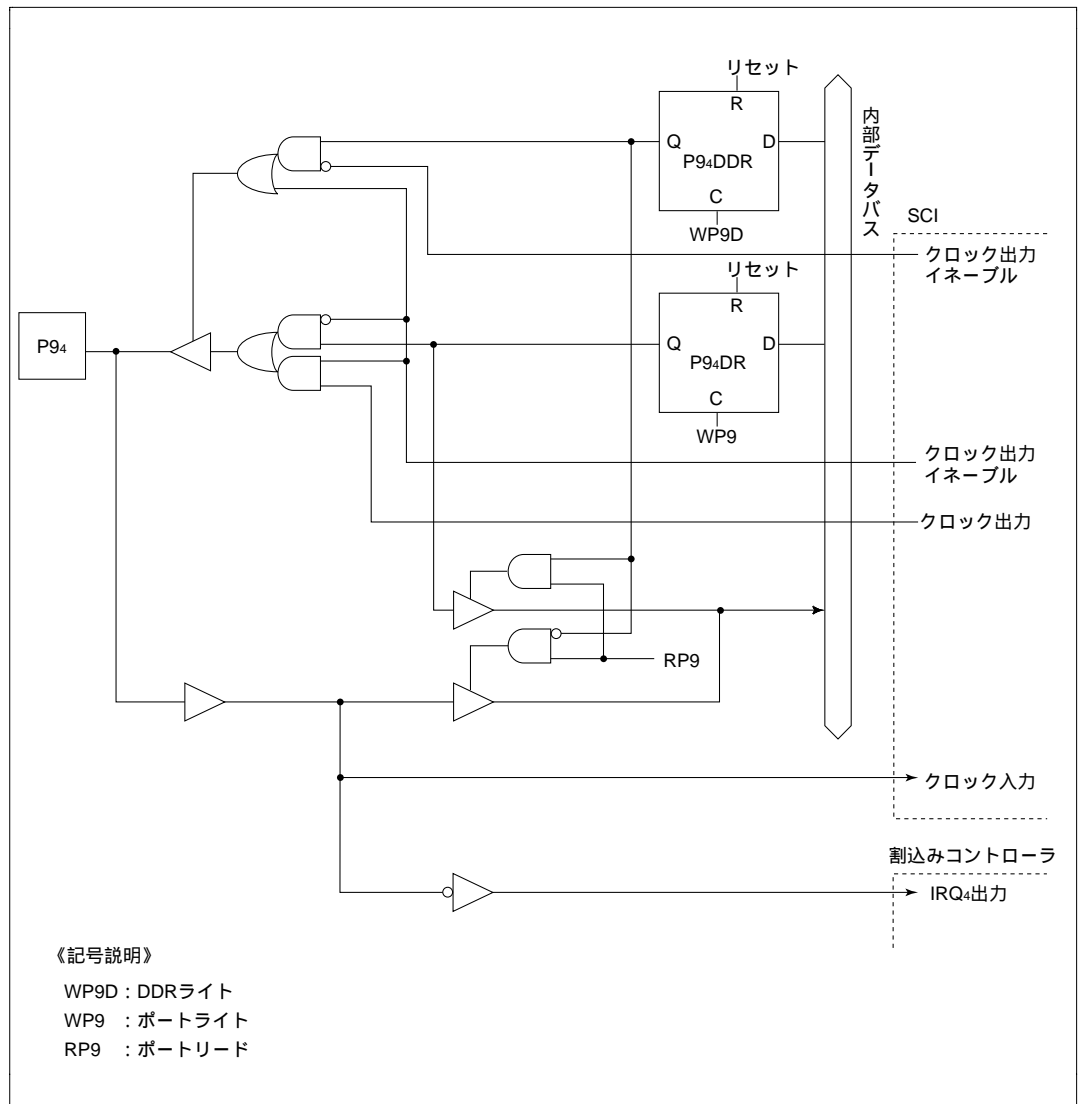


図 C.8 (c) ポート9ブロック図 (P9₄端子)

C.9 ポートAブロック図

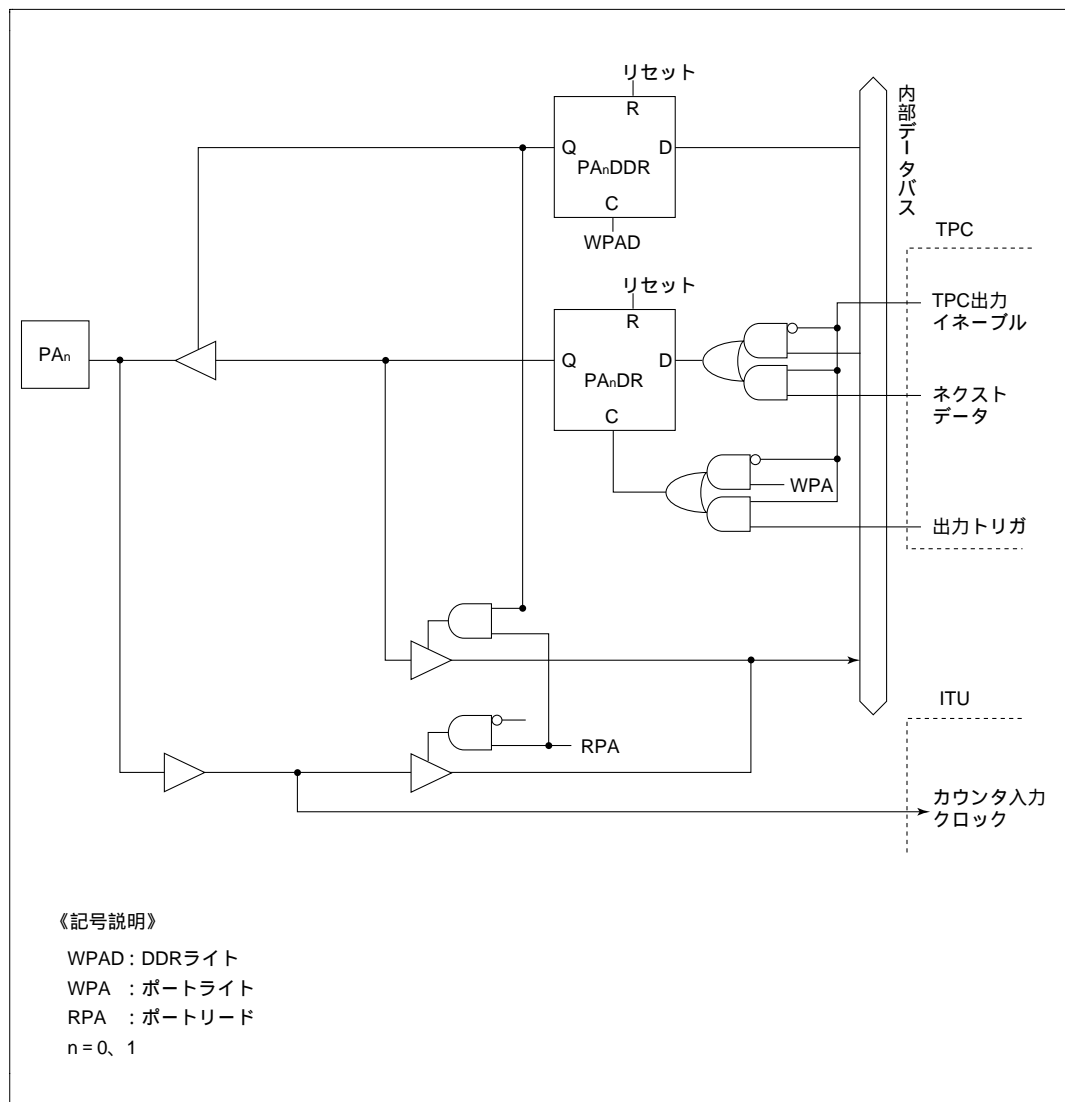


図 C.9 (a) ポート A ブロック図 (PA₀, PA₁端子)

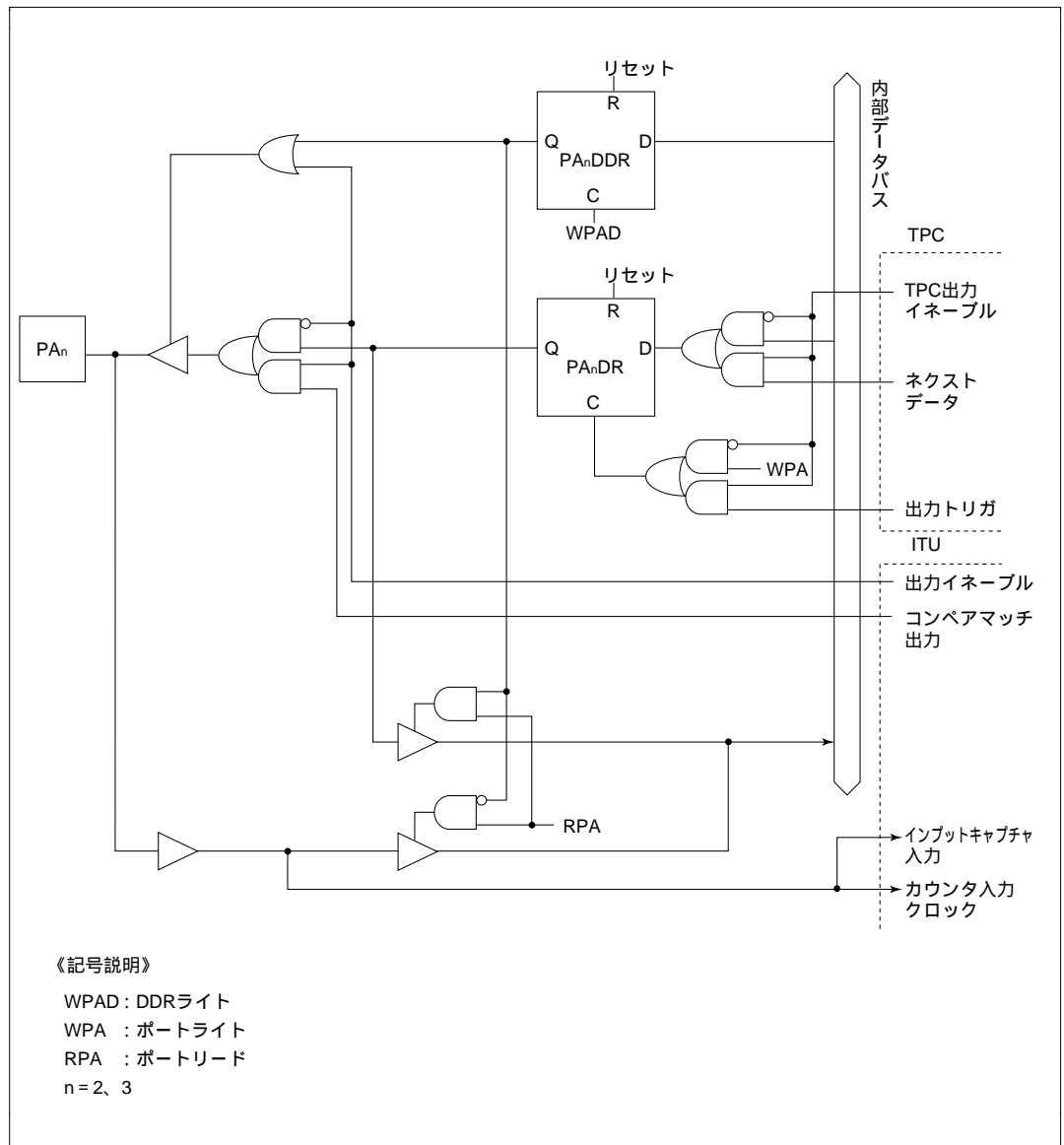


図 C.9 (b) ポート A ブロック図 (PA₂, PA₃端子)

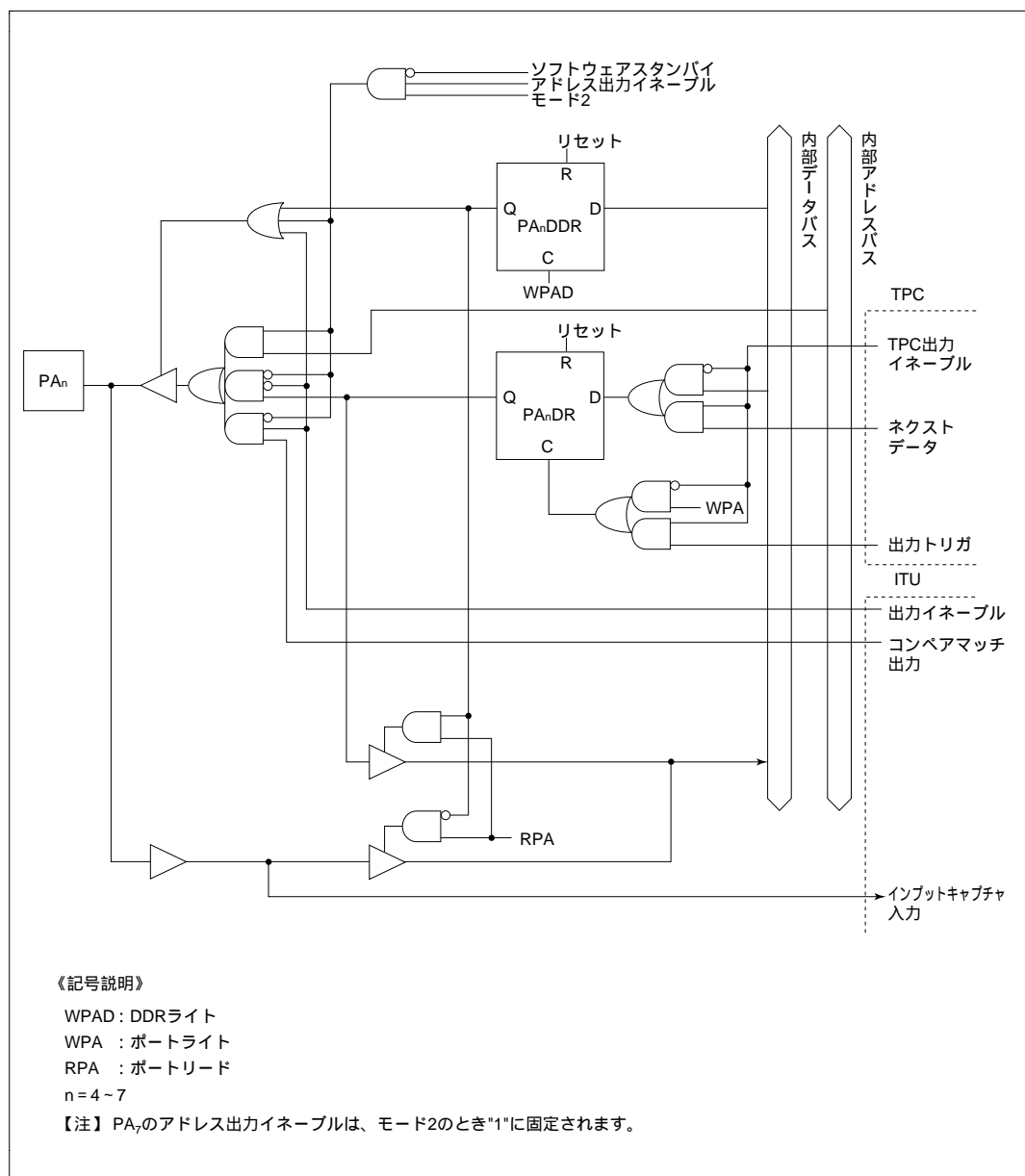


図 C.9 (c) ポート A ブロック図 (PA₄ ~ PA₇端子)

C.10 ポートBブロック図

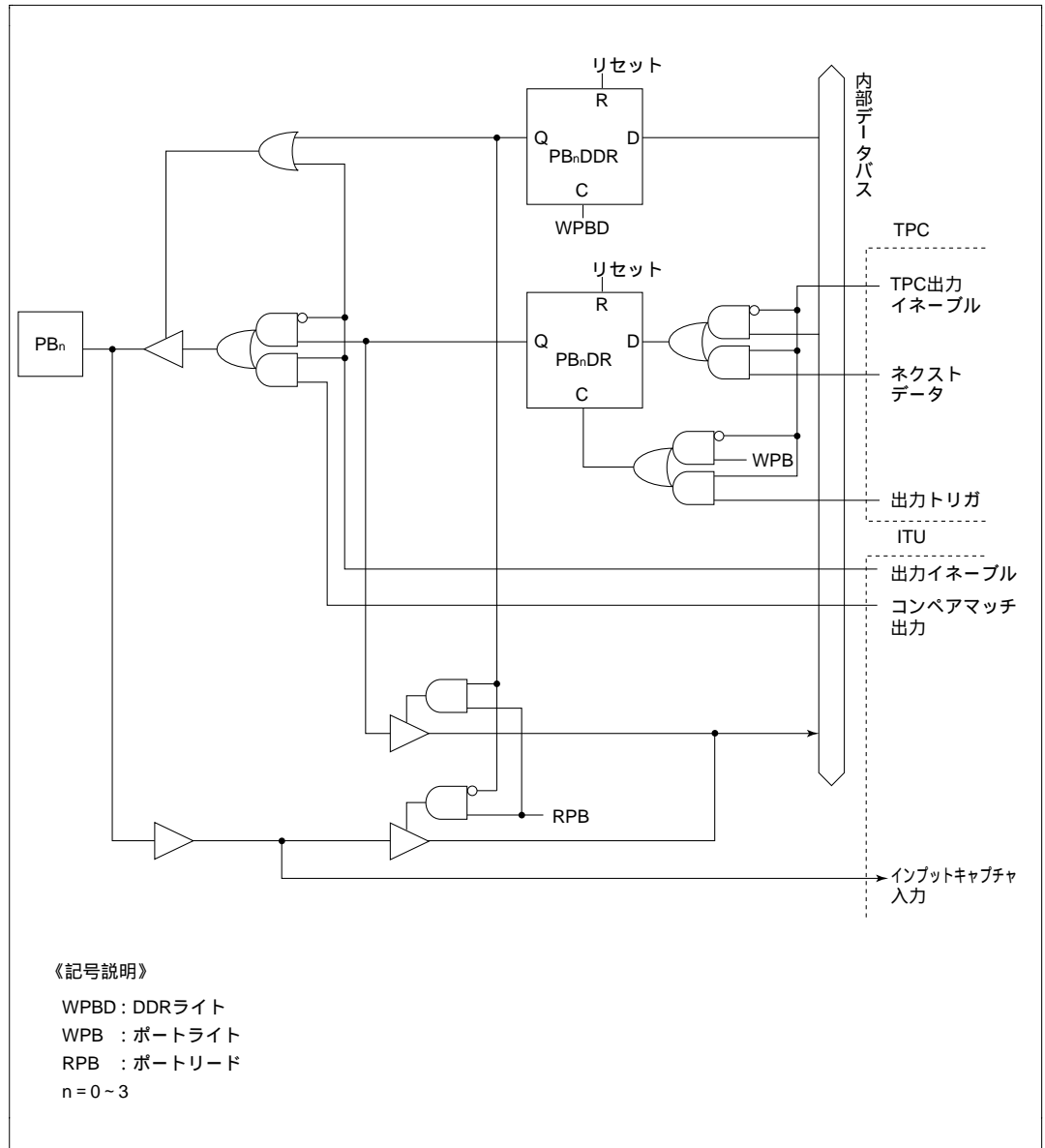


図 C.10 (a) ポート B ブロック図 (PB₀ ~ PB₃端子)

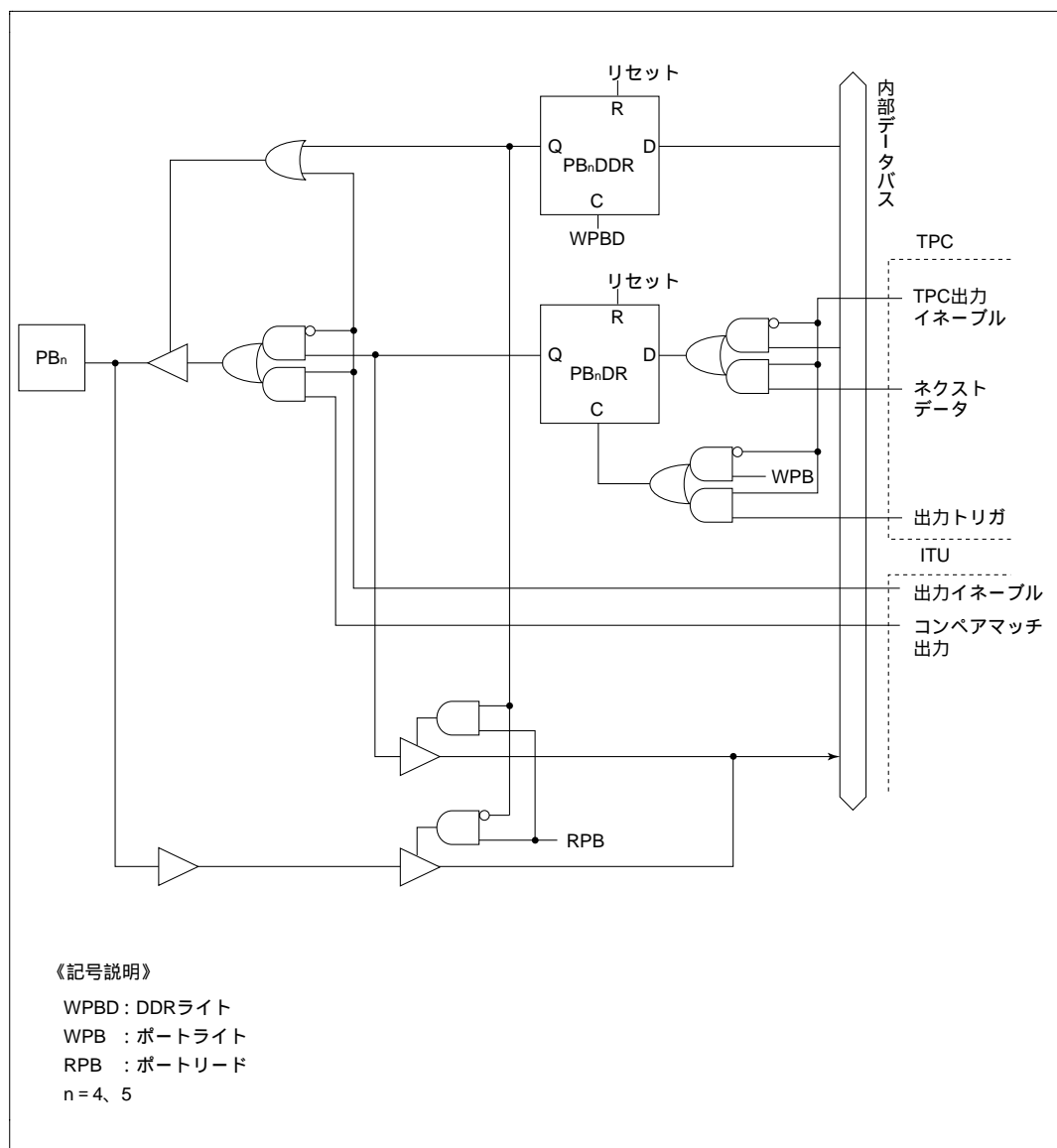


図 C.10 (b) ポート B ブロック図 (PB₄、PB₅端子)

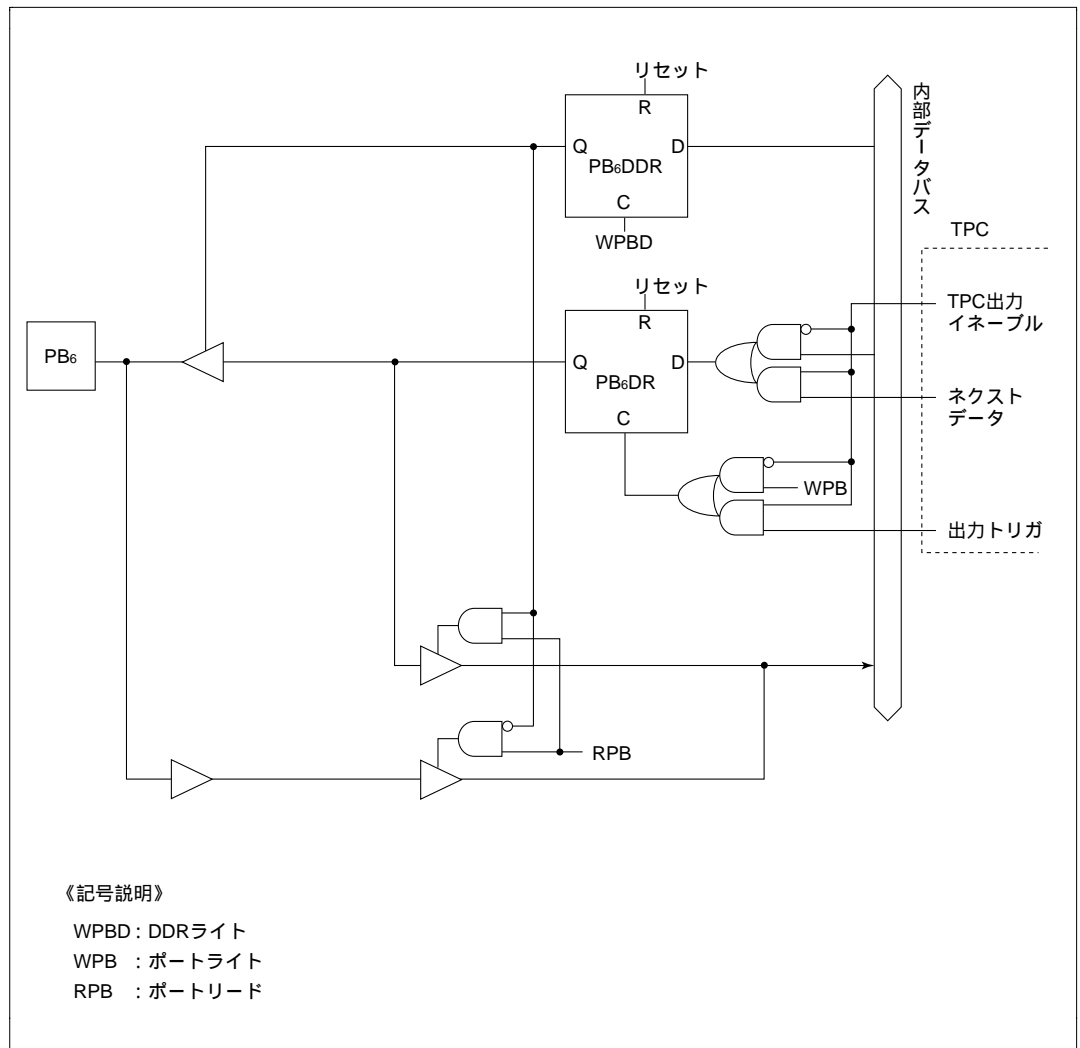


図 C.10 (c) ポート B ブロック図 (PB₆端子)

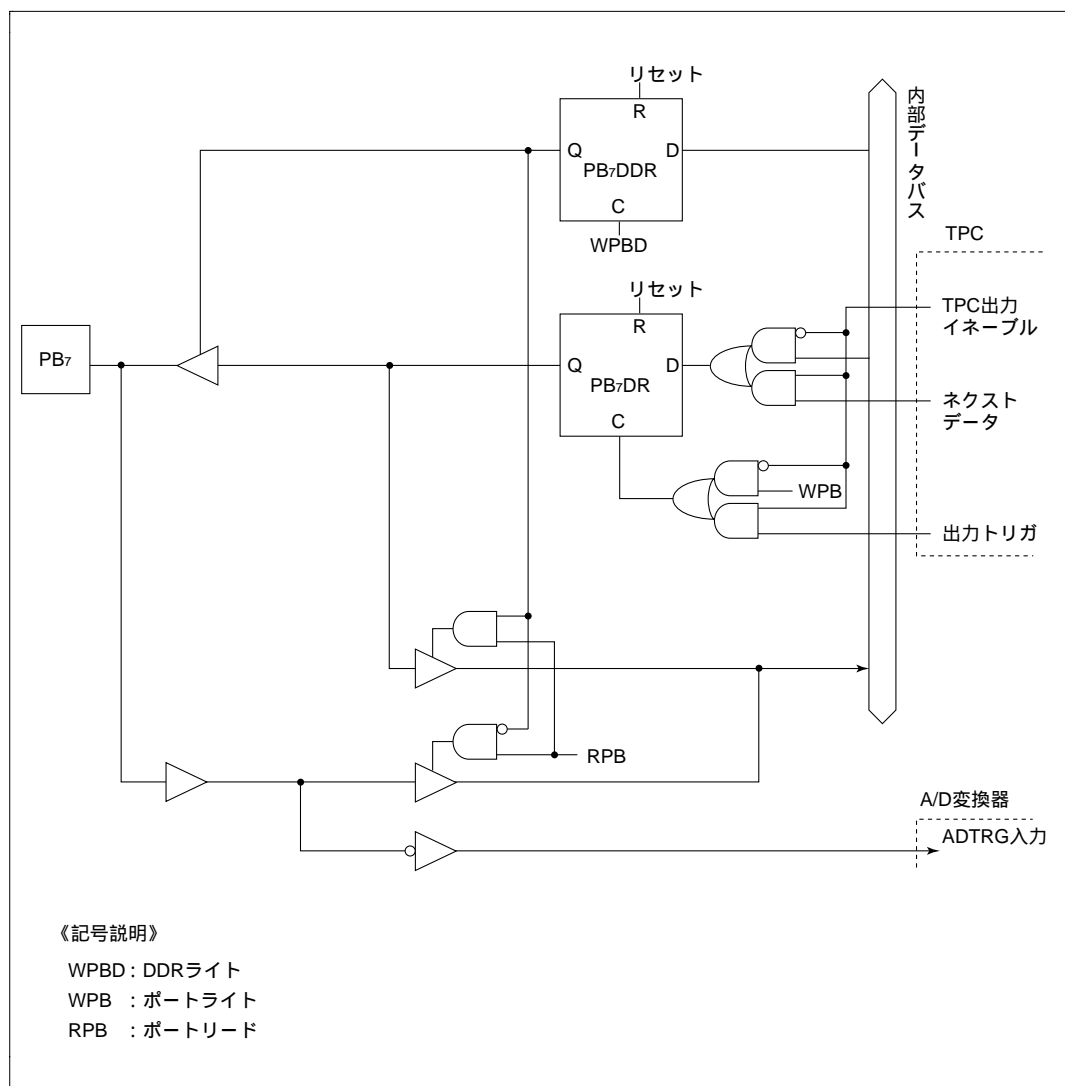


図 C.10 (d) ポート B ブロック図 (PB₇端子)

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

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	スリープ モード	プログラム 実行状態
		クロック 出力	T	H	クロック 出力	クロック出力
P1 ₇ ~ P1 ₀	1、2	T	T	keep	keep	[DDR = 0] 入力ポート
				T	keep	[DDR = 1] A ₇ ~ A ₀
	3	T	T	keep	keep	入出力ポート
P2 ₇ ~ P2 ₀	1、2	T	T	keep	keep	[DDR = 0] 入力ポート
				T	keep	[DDR = 1] A ₁₅ ~ A ₈
	3	T	T	keep	keep	入出力ポート
P3 ₇ ~ P3 ₀	1、2	T	T	T	T	D ₇ ~ D ₀
	3	T	T	keep	keep	入出力ポート
P5 ₃ ~ P5 ₀	1、2	T	T	keep	keep	[DDR = 0] 入力ポート
				T	keep	[DDR = 1] A ₁₉ ~ A ₁₆
	3	T	T	keep	keep	入出力ポート
P6 ₀	1、2	WAITとして機能		T	T	WAIT
		P6 ₀ として機能	T	T	keep	keep
	3	T	T	keep	keep	入出力ポート
P6 ₅ ~ P6 ₃	1、2	H	T	T	H	\overline{WR} 、 \overline{RD} 、 \overline{AS}
	3	T	T	keep	keep	入出力ポート
P7 ₇ ~ P7 ₀	1~3	T	T	T	T	入力ポート
P8 ₃ ~ P8 ₀	1~3	T	T	keep	keep	入出力ポート

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	スリープ モード	プログラム 実行状態
P9 ₄ 、P9 ₂ 、 P9 ₀	1~3	T	T	keep	keep	入出力ポート
PA ₇	2	L	T	T	T	A ₂₀
	1、3	T	T	keep	keep	入出力ポート
PA ₆ ~PA ₄	2	T	T	T	T	[ADRCR=0] A ₂₁ ~A ₂₃
		T	T	keep	keep	[ADRCR=1] 入出力ポート
	1、3	T	T	keep	keep	入出力ポート
PA ₃ ~PA ₀	1~3	T	T	keep	keep	入出力ポート
PB ₇ ~PB ₀	1~3	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

ADRCR : アドレスコントロールレジスタ

D.2 リセット時の端子状態

(1) T1 ステートでのリセット

外部メモリアクセス中の T1 ステートで、 $\overline{\text{RES}}$ 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、 $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{WR}}$ が "High" レベル、データバス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 (の立下がりでのサンプリング) してから、0.5 ステート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。

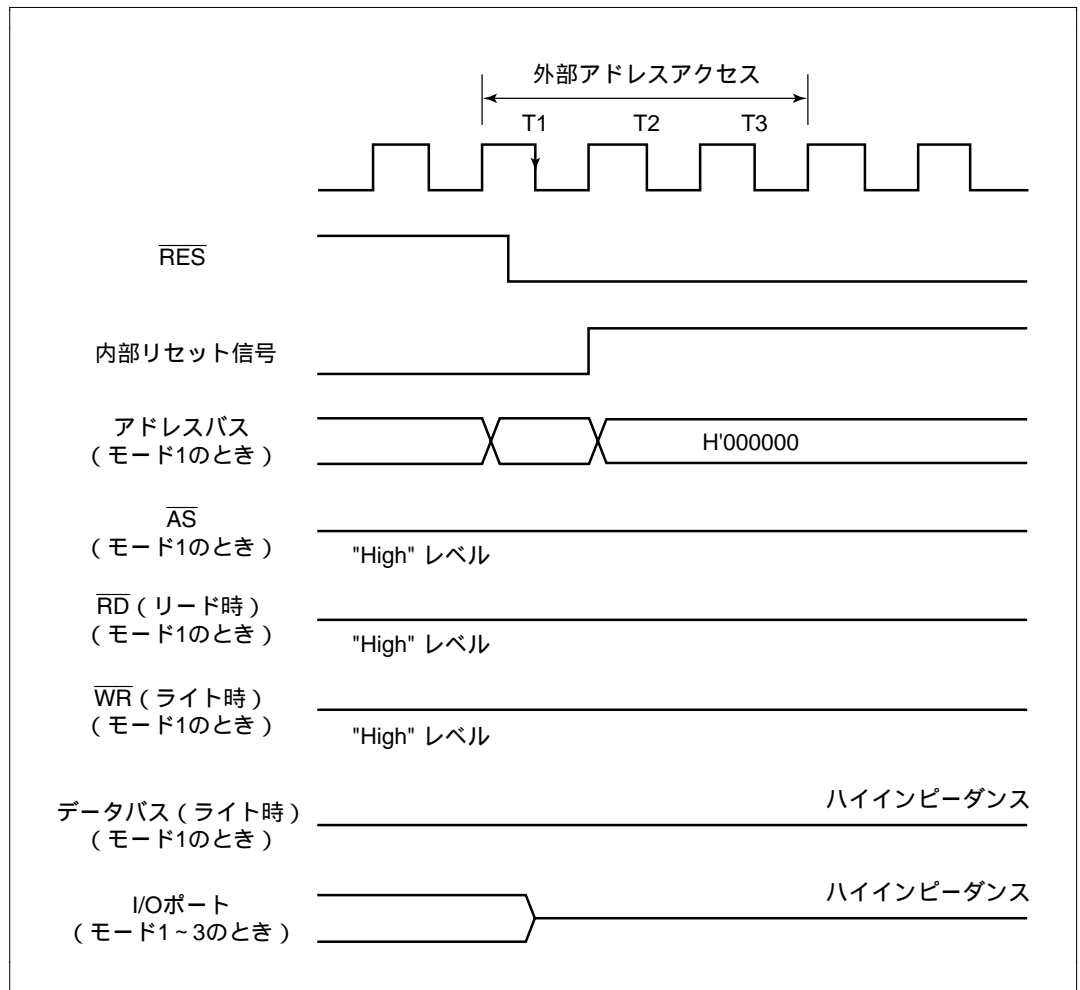


図 D.1 メモリアクセス中のリセット (T1 ステートでのリセット)

(2) T2 ステートでのリセット

外部メモリアクセス中の T2 ステートで、 $\overline{\text{RES}}$ 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、 $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{WR}}$ が "High" レベル、データバス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、0.5 ステート後にイニシャライズされアドレスバスは "Low" レベルとなります。

T_w サイクルでのリセットについても同様です。

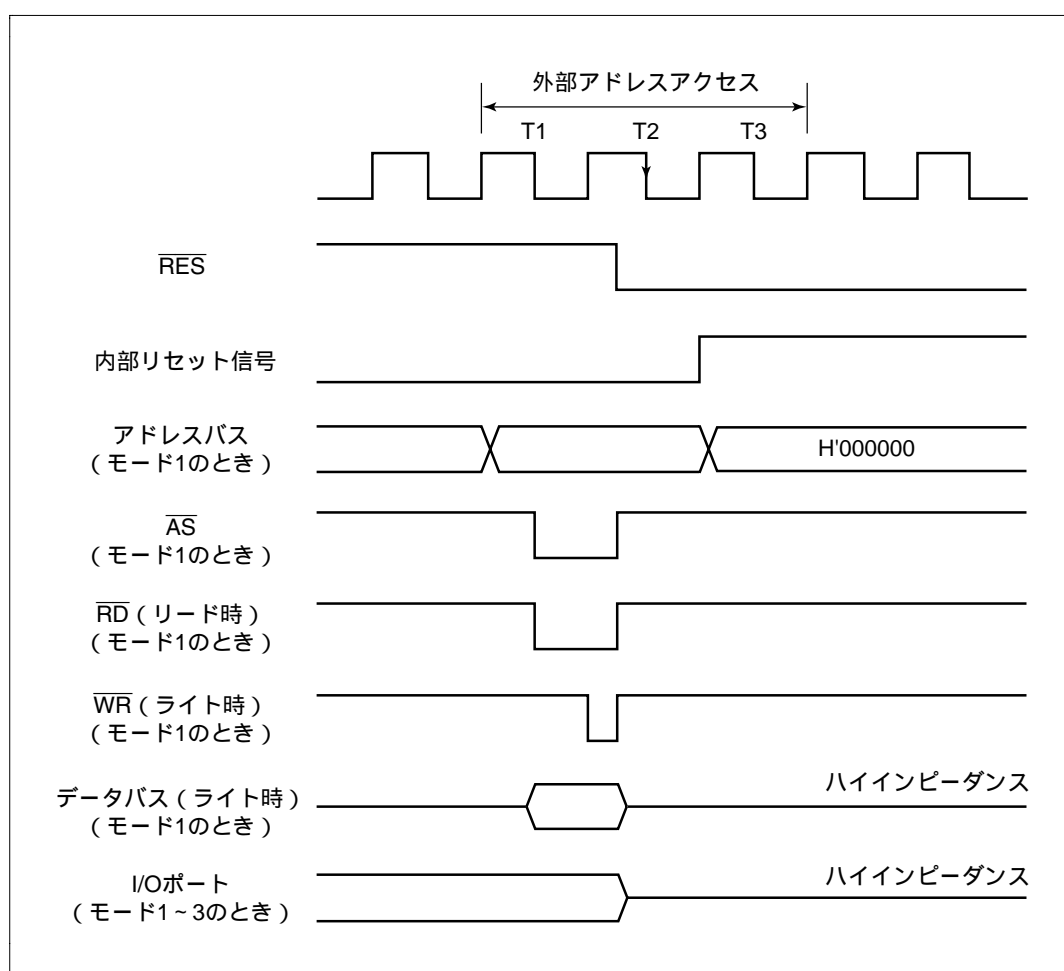


図 D.2 メモリアクセス中のリセット (T2 ステートでのリセット)

(3) T3 ステートでのリセット

外部3 ステート空間アクセス中の T3 ステートで、 $\overline{\text{RES}}$ 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、 $\overline{\text{AS}}$ 、 $\overline{\text{RD}}$ 、 $\overline{\text{WR}}$ が "High" レベル、データバスはハイインピーダンスになります。

アドレスバスは T3 ステート中保持されます。

2 ステートアクセス空間の T2 ステートでのリセットについても同様です。

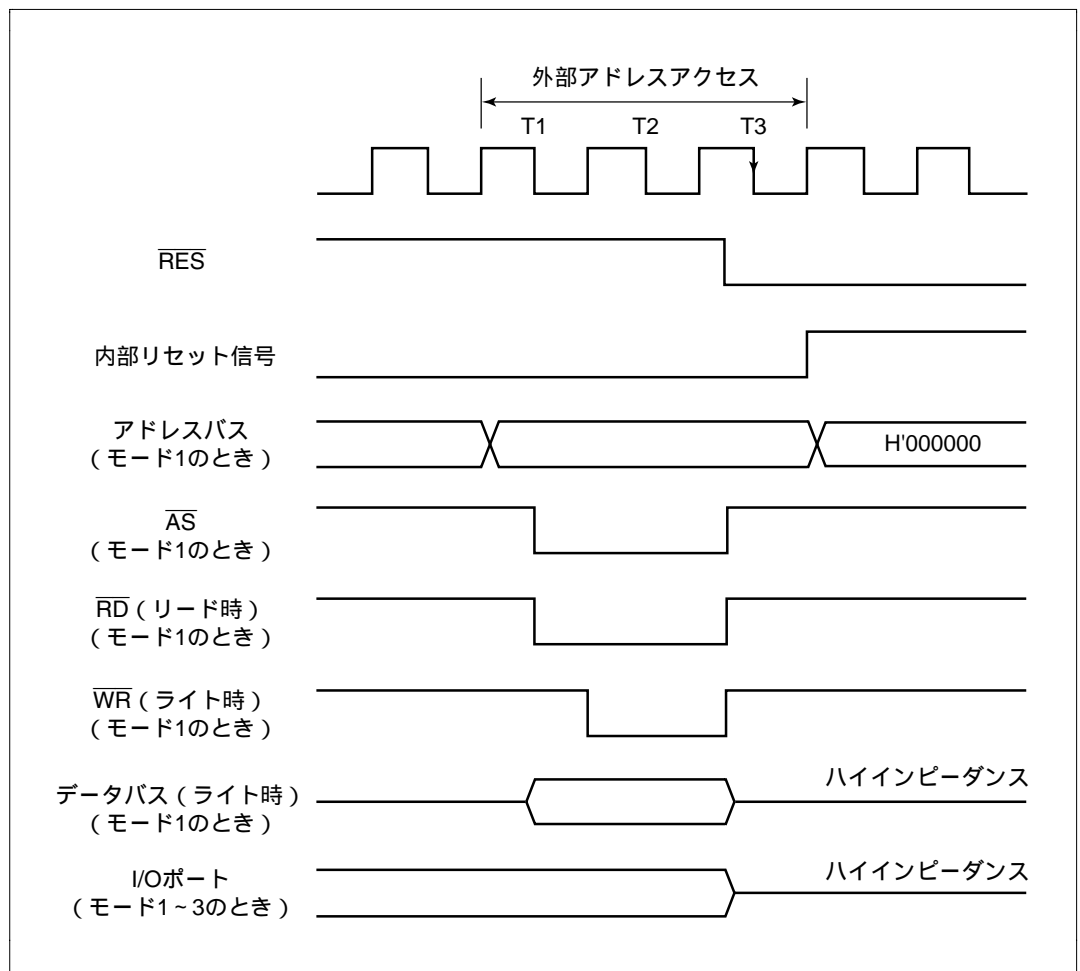


図 D.3 メモリアクセス中のリセット (T3 ステートでのリセット)

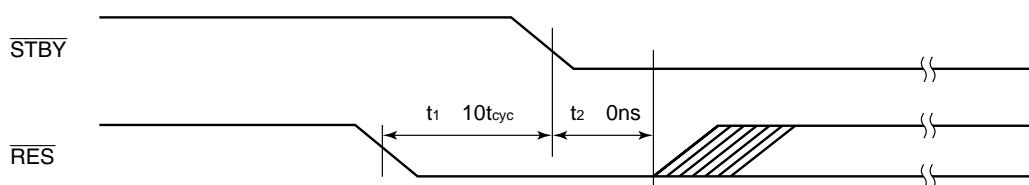
E. ハードウェアスタンバイモード遷移 / 復帰時のタイミングについて

【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、 $\overline{\text{RES}}$ 信号の立上がりは、 $\overline{\text{STBY}}$ 信号の立下がりに対し、min 0ns です。

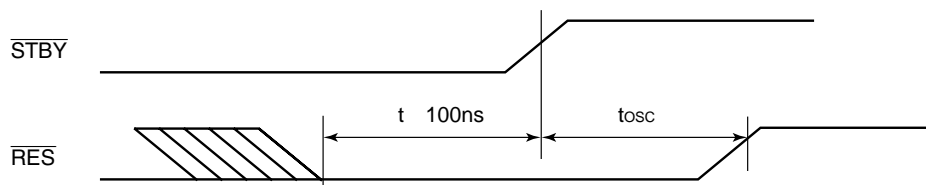


(2) SYSCR の RAME ビットを "0" にクリアした状態または RAM の内容を保持しない場合

(1) のように $\overline{\text{RES}}$ 信号を "Low" にする必要はありません。

【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. ROM 発注手順

F.1 ROM 書き換え品開発の流れ（発注手順）

マイコン応用システムプログラムの開発終了後、ROM データ（2 組以上）、注文仕様書、オプションリストおよびマーク仕様を一緒に提出していただきます。これにより、弊社では図 F.1 の流れ図に沿って ROM 書き換え品の開発を行います。

表 F.1 に ROM 発注時に必要な提出物を示します。なお、詳細については、弊社担当営業へお問い合わせください。

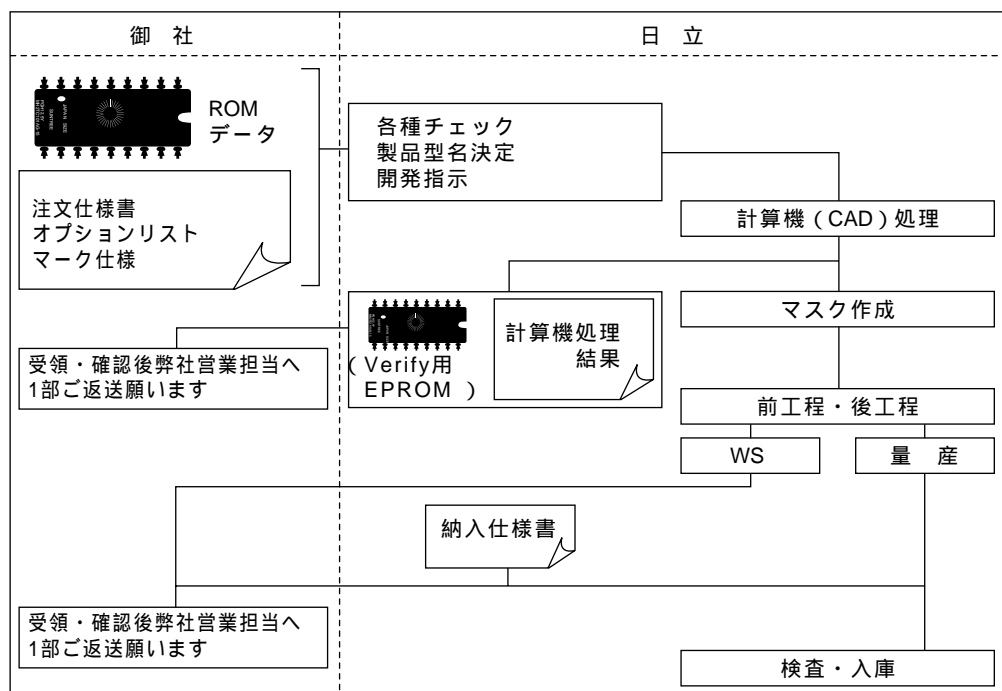


図 F.1 ROM 書き換え品開発の流れ

表 F.1 ROM 発注時に必要な提出物

発注媒体	EPROM または ZTAT [®] マイコン
提出物	ROM データ
	注文仕様書
	オプションリスト* ¹
	マーク仕様例* ²

【注】 *1 製品シリーズにより必要ないものがあります。また、内容も異なります。

*2 特別仕様の場合には、提出してください。

F.2 ROM 発注時の注意事項

提出していただく ROM データは、次の注意事項にしたがって、EPROMまたはZTAT®マイコンで提出してください。なお、EPROMまたはZTAT®マイコン以外の媒体（フロッピーディスク等）では対応できませんのでご注意ください。

- (1) EPROM に ROM データを書き込む際は、事前にデータを充分消去し、中途半端なレベルが出力されないことを確認してから使用してください。
- (2) 発注用 EPROM において、ROM データの未使用（NOT USED）領域またはリザーブ領域には、必ず'FF'を書き込んでください。
- (3) 提出していただく EPROM には遮光ラベルを貼り、御社の品番等を記入してください。
- (4) EPROM に書き込みを行った後は、静電気による素子の破壊、紫外線や放射線による書き込みデータの損失を招かないようにするとともに、運搬の際は導伝性のシートに梱包するなど取り扱いに充分注意してください（アルミ箔、発泡スチロール等は不可）。なお、これらによるデータの読み取りエラーに備え、同一内容の EPROM を 2 組以上提出してください。

G. 型名一覧

表 G.1 H8/3035 シリーズ型名一覧

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)
H8/3035	PROM 版	標準品	HD6473035F	HD6473035F	80 ピン QFP (FP-80A)
			HD6473035TE	HD6473035TE	80 ピン TQFP (TFP-80C)
	ROM 版	標準品	HD6433035F	HD6433035 (***) F	80 ピン QFP (FP-80A)
			HD6433035TE	HD6433035 (***) TE	80 ピン TQFP (TFP-80C)
H8/3034	マスク	標準品	HD6433034F	HD6433034 (***) F	80 ピン QFP (FP-80A)
	ROM 版		HD6433034TE	HD6433034 (***) TE	80 ピン TQFP (TFP-80C)
H8/3033	マスク	標準品	HD6433033F	HD6433033 (***) F	80 ピン QFP (FP-80A)
	ROM 版		HD6433033TE	HD6433033 (***) TE	80 ピン TQFP (TFP-80C)

【注】 マスク ROM 版の (***) は ROM コードです。

H. 外形寸法図

本 LSI の外形寸法図を図 H.1、図 H.2 に示します。

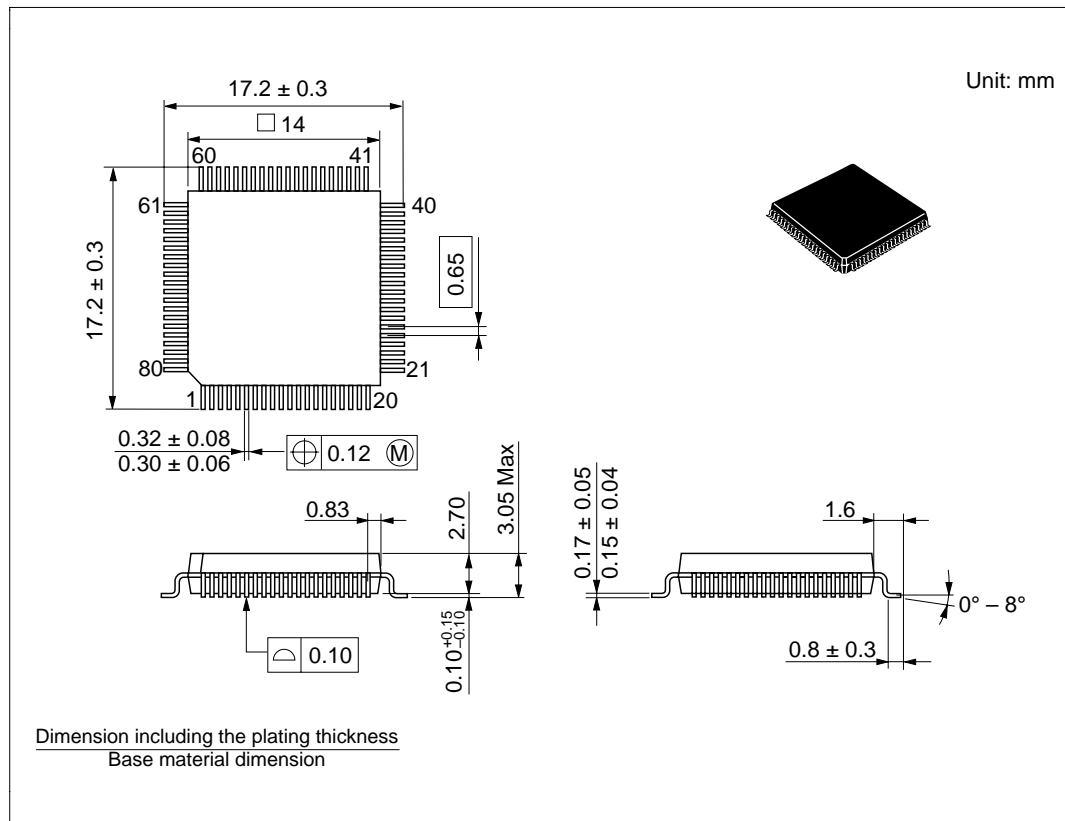


図 H.1 外形寸法図 (FP-80A)

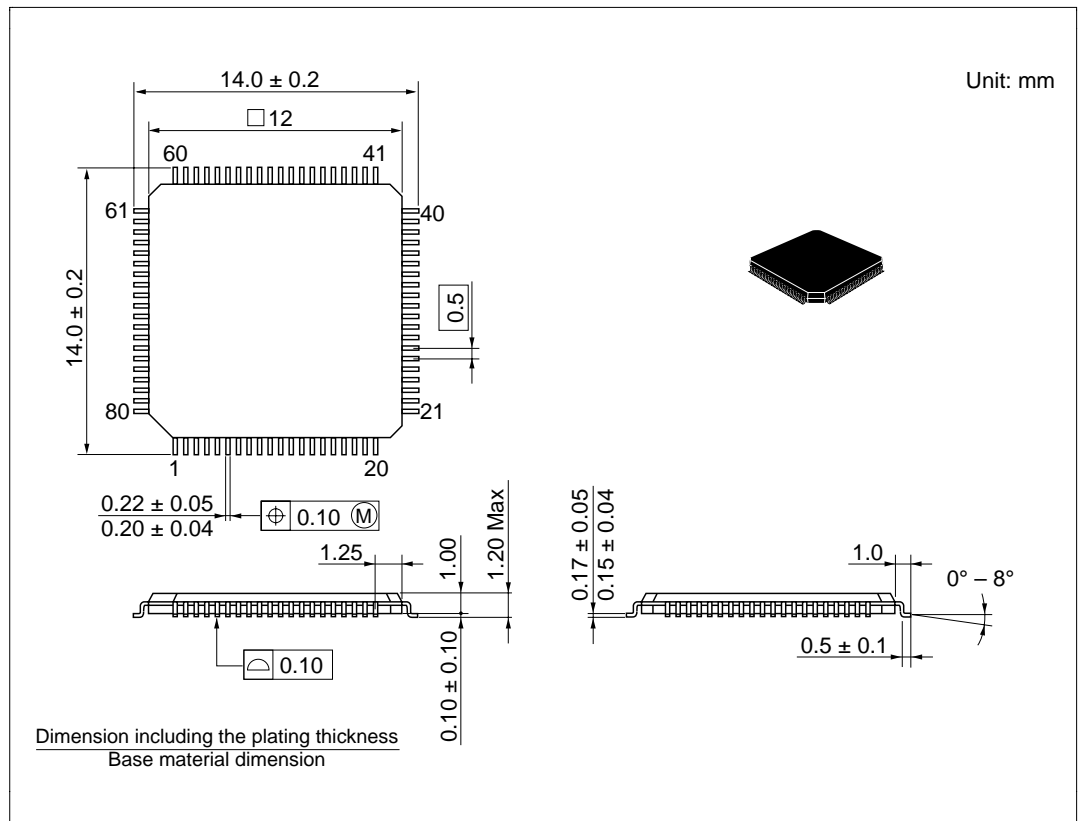


図 H.2 外形寸法図 (TFP-80C)

H8/3035 シリーズ ハードウェアマニュアル



ルネサスエレクトロニクス株式会社
神奈川県川崎市中原区下沼部1753 〒211-8668