



本ドキュメントはCypress (サイプレス) 製品に関する情報が記載されております。本ドキュメントには、仕様の開発元企業として「スパンション」, 「Spansion」, 「富士通」または「Fujitsu」の名が記載されておりますが、これらの製品は Cypress が新規および既存のお客様に引き続き提供してまいります。

商品仕様の継続性について

Cypress 製品として提供することに伴う商品仕様としての変更はなく、ドキュメントとしての変更もありません。また本ページのお知らせは、変更情報として追記いたしません。本ドキュメントに変更情報が記載されている場合、それは本お知らせを除いた前版からの変更点です。なお、今後改訂は必要に応じて行われますが、その際の変更内容は改訂後のドキュメントに記載いたします。

オーダ型格および品名について

Cypress は既存のオーダ型格および品名を引き続きサポートいたします。これらの製品をご注文の際は、このドキュメントに記載されているオーダ型格および品名をご使用ください。

詳しいお問い合わせ先

Cypress 製品およびそのソリューションの詳細につきましては、お近くの営業所へお問い合わせください。

サイプレスについて

サイプレス (銘柄コード: CY) は、車載や産業機器、ネットワーキング プラットフォームから高機能民生機器およびモバイル機器まで、今日の最先端組み込みシステム向けに高性能で高品質のソリューションを提供します。NOR フラッシュ メモリや F-RAMTM、SRAM、TraveoTM マイクロコントローラー、業界唯一の PSoC[®] プログラマブル システムオンチップ ソリューション、アナログおよび PMIC Power Management IC、CapSense[®] 静電容量タッチセンシング コントローラー、Wireless BLE Bluetooth[®] Low-Energy、USB コネクティビティ ソリューションなど、幅広い差別化製品ポートフォリオを、一貫した革新性と業界最高クラスの技術サポート、比類のないシステム バリューとともにグローバルに提供します。

F²MC-16LX

16 ビット マイクロコントローラ

MB90820B Series

ハードウェアマニュアル

F²MC-16LX

16 ビット マイクロコントローラ

MB90820B Series

ハードウェアマニュアル

富士通マイクロエレクトロニクスのマイコンを効率的に開発するための情報を下記 URL にてご紹介いたします。
ご採用を検討中、またはご採用いただいたお客様に有益な情報を公開しています。

開発における最新の注意事項に関しては、必ず「Check Sheet」を参照してください。
「Check Sheet」はシステム開発において、問題を未然に防ぐことを目的として、最低限必要と思われるチェック項目をリストにしたものです。

<http://edevicе.fujitsu.com/micom/jp-support/>

富士通マイクロエレクトロニクス株式会社

はじめに

■ 本書の目的と対象読者

富士通マイクロエレクトロニクス製品につきまして、平素より格別のご愛読を賜り厚くお礼申し上げます。

MB90820B シリーズは、ASIC(Application Specific IC) 対応が可能なオリジナル 16 ビット・ワンチップマイクロコントローラである F²MC-16LX ファミリの汎用品の一つとして開発された製品です。

本書は、実際にこの半導体を使って製品を設計する技術者の方を対象に、MB90820B シリーズの機能や動作について記載しています。本書をご一読ください。

■ 商標

F²MC は富士通マイクロエレクトロニクス株式会社の商標です。

本書に記載されている社名および製品名などの固有名詞は、各社の商標または登録商標です。

■ ライセンス

本製品には、お客様が Philips 社の定めた I²C 標準仕様書に従う I²C システムの中で使用されることを条件に、Philips 社 I²C 特許がライセンスされております。

Purchase of Fujitsu I²C components conveys a license under the Philips I²C Patent Rights to use these components in an I²C system provided that the system conforms to the I²C Standard Specification as defined by Philips.

■ 本書の全体構成

本書は、以下に示す 23 の章、および付録から構成されています。

第 1 章 概要

この章では、MB90820B シリーズの特長や基本的な仕様について示します。

第 2 章 CPU

この章では、MB90820B シリーズのメモリ空間について説明します。

第 3 章 リセット

この章では、リセットの機能と動作について説明します。

第 4 章 クロック

この章では、クロックの機能と動作について説明します。

第 5 章 クロックスーパーバイザ

この章では、クロックスーパーバイザの機能と動作について説明します。

第 6 章 低消費電力モード

この章では、低消費電力モードの機能と動作について説明します。

第 7 章 割込み

この章では、割込みと拡張インテリジェント I/O サービスについて説明します。

第 8 章 モード設定

この章では、動作モードとメモリアクセスモード設定について説明します。

第9章 I/O ポート

この章では、I/O ポートの機能と動作について説明します。

第10章 タイムベースタイマ

この章では、タイムベースタイマの機能と動作について説明します。

第11章 ウォッチドッグタイマ

この章では、ウォッチドッグタイマの機能と動作について説明します。

第12章 16 ビットリロードタイマ

この章では、16 ビットリロードタイマの機能と動作について説明します。

第13章 PWC タイマ

この章では、PWC タイマの機能と動作について説明します。

第14章 16 ビット PPG タイマ

この章では、16 ビット PPG タイマの機能と動作について説明します。

第15章 多機能タイマ

この章では、多機能タイマの機能と動作について説明します。

第16章 遅延割込み発生モジュール

この章では、遅延割込み発生モジュールの機能と動作および使用上の注意について説明します。

第17章 DTP/ 外部割込み回路

この章では、DTP/外部割込み回路の機能と動作および使用上の注意について説明します。

第18章 8/10 ビット A/D コンバータ

この章では、8/10 ビット A/D コンバータの機能と動作および使用上の注意について説明します。

第19章 D/A コンバータ

この章では、D/A コンバータの機能と動作および使用上の注意について説明します。

第20章 UART

この章では、UART の機能と動作および使用上の注意について説明します。

第21章 ROM 修正機能

この章では、ROM 修正機能の機能と動作、使用上の注意および応用例について説明します。

第22章 ROM ミラー機能選択モジュール

この章では、ROM ミラー機能選択モジュールの機能と動作について説明します。

第23章 512K ビットフラッシュメモリ

この章では、512K/1024 ビットフラッシュメモリについて説明します。

付録

付録 A, B, C では、I/O マップ、MB90F822B/F823B シリアル書込み接続例および命令について示します。

- 本資料の記載内容は、予告なしに変更することがありますので、ご用命の際は営業部門にご確認ください。
- 本資料に記載された動作概要や応用回路例は、半導体デバイスの標準的な動作や使い方を示したもので、実際に使用する機器での動作を保証するものではありません。したがって、これらを使用するにあたってはお客様の責任において機器の設計を行ってください。これらの使用に起因する損害などについては、当社はその責任を負いません。
- 本資料に記載された動作概要・回路図を含む技術情報は、当社もしくは第三者の特許権、著作権等の知的財産権やその他の権利の使用権または実施権の許諾を意味するものではありません。また、これらの使用について、第三者の知的財産権やその他の権利の実施ができることの保証を行うものではありません。したがって、これらの使用に起因する第三者の知的財産権やその他の権利の侵害について、当社はその責任を負いません。
- 本資料に記載された製品は、通常の産業用、一般事務用、パーソナル用、家庭用などの一般的用途に使用されることを意図して設計・製造されています。極めて高度な安全性が要求され、仮に当該安全性が確保されない場合、社会的に重大な影響を与えかつ直接生命・身体に対する重大な危険性を伴う用途（原子力施設における核反応制御、航空機自動飛行制御、航空交通管制、大量輸送システムにおける運行制御、生命維持のための医療機器、兵器システムにおけるミサイル発射制御をいう）、ならびに極めて高い信頼性が要求される用途（海底中継器、宇宙衛星をいう）に使用されるよう設計・製造されたものではありません。したがって、これらの用途にご使用をお考えのお客様は、必ず事前に営業部門までご相談ください。ご相談なく使用されたことにより発生した損害などについては、責任を負いかねますのでご了承ください。
- 半導体デバイスはある確率で故障が発生します。当社半導体デバイスが故障しても、結果的に人身事故、火災事故、社会的な損害を生じさせないよう、お客様は、装置の冗長設計、延焼対策設計、過電流防止対策設計、誤動作防止設計などの安全設計をお願いします。
- 本資料に記載された製品を輸出または提供する場合は、外国為替及び外国貿易法および米国輸出管理関連法規等の規制をご確認の上、必要な手続きをおとりください。
- 本書に記載されている社名および製品名などの固有名詞は、各社の商標または登録商標です。

目次

第1章	概要	1
1.1	MB90820B シリーズの特長	2
1.2	MB90820B シリーズの品種構成	5
1.3	MB90820B シリーズのブロックダイアグラム	7
1.4	端子配列図	8
1.5	外形寸法図	10
1.6	端子機能説明	13
1.7	入出力回路形式	17
1.8	デバイス取扱い上の注意	21
第2章	CPU の機能	23
2.1	CPU	24
2.2	メモリ空間	25
2.3	メモリマップ	27
2.4	アドレス指定	29
2.4.1	リニア方式によるアドレス指定	30
2.4.2	バンク方式によるアドレス指定	31
2.5	多バイト長データのメモリ配置	33
2.6	レジスタ	35
2.7	専用レジスタ	36
2.7.1	アキュムレータ (A)	38
2.7.2	スタックポインタ (USP, SSP)	41
2.7.3	プロセッサステータス	43
2.7.4	コンディションコードレジスタ (PS:CCR)	44
2.7.5	レジスタバンクポインタ (PS:RP)	46
2.7.6	割込みレベルマスクレジスタ (PS:ILM)	47
2.7.7	プログラムカウンタ (PC)	48
2.7.8	ダイレクトページレジスタ (DPR)	49
2.7.9	バンクレジスタ (PCB, DTB, USB, SSB, ADB)	50
2.8	汎用レジスタ	51
2.9	プリフィックスコード	53
2.9.1	バンクセレクトプリフィックス (PCB, DTB, ADB, SPB)	54
2.9.2	コモンレジスタバンクプリフィックス (CMR)	56
2.9.3	フラグ変化抑止プリフィックス (NCC)	57
2.9.4	プリフィックスコードに関する制約	58
第3章	リセット	61
3.1	リセットの概要	62
3.2	リセット要因と発振安定待ち時間	64
3.3	外部リセット端子	66
3.4	リセット動作	67
3.5	リセット要因ビット	69
3.6	リセットによる各端子の状態	71

第4章	クロック	73
4.1	クロック	74
4.2	クロック発生部のブロックダイアグラム	76
4.3	クロック選択レジスタ	78
4.3.1	クロック選択レジスタ (CKSCR)	79
4.3.2	PLL クロック制御レジスタ (PCKCR)	81
4.4	クロックモード	83
4.5	発振安定待ち時間	85
4.6	振動子および外部クロックの接続	86
第5章	クロックスーパバイザ	87
5.1	クロックスーパバイザの概要	88
5.2	クロックスーパバイザの構成	89
5.3	クロックスーパバイザのレジスタ	91
5.3.1	クロックスーパバイザ制御レジスタ (CSVCR)	92
5.4	クロックスーパバイザの動作説明	94
5.5	クロックスーパバイザ使用上の注意	97
第6章	低消費電力モード	99
6.1	低消費電力モードの概要	100
6.2	低消費電力制御回路のブロックダイアグラム	102
6.3	低消費電力モード制御レジスタ (LPMCR)	104
6.4	CPU 間欠動作モード	107
6.5	スタンバイモード	108
6.5.1	スリープモード	109
6.5.2	タイムベースタイマモード	112
6.5.3	ストップモード	114
6.6	状態遷移図	116
6.7	スタンバイモード、リセット時の端子状態	119
6.8	低消費電力モード使用上の注意	120
第7章	割込み	123
7.1	割込み	124
7.2	割込み要因と割込みベクタ	126
7.3	割込み制御レジスタとリソース	130
7.3.1	割込み制御レジスタ (ICR00 ~ ICR15)	132
7.3.2	割込み制御レジスタの機能	134
7.4	ハードウェア割込み	137
7.4.1	ハードウェア割込みの動作	140
7.4.2	割込み動作時の処理	142
7.4.3	ハードウェア割込み使用手順	143
7.4.4	複数の割込み	144
7.4.5	ハードウェア割込み処理時間	146
7.5	ソフトウェア割込み	148
7.6	拡張インテリジェント I/O サービス (EI ² OS) の割込み	150
7.6.1	拡張インテリジェント I/O サービス (EI ² OS) ディスクリプタ (ISD)	152
7.6.2	EI ² OS ディスクリプタ (ISD) のレジスタ	153
7.6.3	拡張インテリジェント I/O サービス (EI ² OS) の動作	156
7.6.4	拡張インテリジェント I/O サービス (EI ² OS) 使用手順	157

7.6.5	拡張インテリジェント I/O サービス (EI ² OS) 処理時間.....	158
7.7	例外処理割り込み	160
7.8	割り込み処理のスタック動作.....	161
第 8 章	モード設定	163
8.1	モード設定.....	164
8.2	モード端子 (MD2 ~ MD0).....	165
8.3	モードデータ	166
第 9 章	I/O ポート.....	169
9.1	I/O ポートの概要	170
9.2	I/O ポートのレジスタ	172
9.3	ポート 0.....	173
9.3.1	ポート 0 のレジスタ (PDR0, DDR0, RDR0)	176
9.3.2	ポート 0 の動作説明	178
9.4	ポート 1.....	180
9.4.1	ポート 1 のレジスタ (PDR1, DDR1, RDR1)	183
9.4.2	ポート 1 の動作説明	185
9.5	ポート 2.....	187
9.5.1	ポート 2 のレジスタ (PDR2, DDR2, RDR2)	190
9.5.2	ポート 2 の動作	192
9.6	ポート 3.....	194
9.6.1	ポート 3 のレジスタ (PDR3, DDR3, RDR3)	197
9.6.2	ポート 3 の動作説明	199
9.7	ポート 4.....	201
9.7.1	ポート 4 のレジスタ (PDR4, DDR4)	204
9.7.2	ポート 4 の動作説明	205
9.8	ポート 5.....	207
9.8.1	ポート 5 のレジスタ (PDR5, DDR5)	210
9.8.2	ポート 5 の動作説明	211
9.9	ポート 6.....	213
9.9.1	ポート 6 のレジスタ (PDR6, DDR6, ADER0)	215
9.9.2	ポート 6 の動作説明	217
9.10	ポート 7.....	219
9.10.1	ポート 7 のレジスタ (PDR7, DDR7, ADER1)	223
9.10.2	ポート 7 の動作説明	225
9.11	ポート 8.....	227
9.11.1	ポート 8 のレジスタ (PDR8, DDR8)	230
9.11.2	ポート 8 の動作説明	231
第 10 章	タイムベースタイマ.....	233
10.1	タイムベースタイマの概要.....	234
10.2	タイムベースタイマの構成.....	236
10.3	タイムベースタイマ制御レジスタ (TBTC)	238
10.4	タイムベースタイマの割り込み	240
10.5	タイムベースタイマの動作説明	241
10.6	タイムベースタイマ使用上の注意.....	243

第 11 章 ウォッチドッグタイマ	245
11.1 ウォッチドッグタイマの概要	246
11.2 ウォッチドッグタイマの構成	248
11.3 ウォッチドッグタイマ制御レジスタ (WDTC)	249
11.4 ウォッチドッグタイマの動作説明	251
11.5 ウォッチドッグタイマ使用上の注意	253
 第 12 章 16 ビットリロードタイマ	 255
12.1 16 ビットリロードタイマの概要	256
12.2 16 ビットリロードタイマのブロックダイアグラム	258
12.3 16 ビットリロードタイマの端子	260
12.4 16 ビットリロードタイマのレジスタ	261
12.4.1 タイマ制御ステータスレジスタ, 上位 (TMCSRH0/TMCSRH1)	262
12.4.2 タイマ制御ステータスレジスタ, 下位 (TMCSRL0/TMCSRL1)	264
12.4.3 16 ビットタイマレジスタ (TMR0/TMR1)	266
12.4.4 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1)	267
12.5 16 ビットリロードタイマの割込み	268
12.6 16 ビットリロードタイマの動作説明	269
12.6.1 内部クロックモード (リロードモード)	271
12.6.2 内部クロックモード (ワンショットモード)	274
12.6.3 イベントカウントモード	277
12.7 16 ビットリロードタイマの使用上の注意	279
 第 13 章 PWC タイマ	 281
13.1 PWC タイマの概要	282
13.2 PWC タイマのブロックダイアグラム	283
13.3 PWC タイマの端子	284
13.4 PWC タイマのレジスタ	287
13.4.1 PWC 状態制御レジスタ (PWCSH0/PWCSH1, PWCSL0/PWCSL1)	288
13.4.2 PWC データバッファレジスタ (PWC0/PWC1)	293
13.4.3 分周比制御レジスタ (DIV0/DIV1)	294
13.5 PWC タイマの割込み	295
13.6 PWC タイマの動作	297
13.6.1 動作モードの選択	300
13.6.2 タイマとパルス幅測定の起動 / 停止とタイマクリア	302
13.6.3 タイマモード時の動作	304
13.6.4 パルス幅測定モード時の動作	307
13.7 PWC タイマ使用上の注意	313
 第 14 章 16 ビット PPG タイマ	 315
14.1 16 ビット PPG タイマの概要	316
14.2 16 ビット PPG タイマのブロックダイアグラム	317
14.3 16 ビット PPG タイマの端子	318
14.4 16 ビット PPG タイマのレジスタ	320
14.4.1 PPG ダウンカウンタレジスタ (PDCR0 ~ PDCR2)	322
14.4.2 PPG 周期設定バッファレジスタ (PCSR0 ~ PCSR2)	323
14.4.3 PPG デューティ設定バッファレジスタ (PDUT0 ~ PDUT2)	324
14.4.4 PPG 状態制御レジスタ (PCNTL0 ~ PCNTL2, PCNTH0 ~ PCNTH2)	325
14.5 16 ビット PPG タイマの割込み	329

14.6	16 ビット PPG タイマの動作説明	331
14.7	16 ビット PPG タイマ使用上の注意	335
第 15 章	多機能タイマ	337
15.1	多機能タイマの概要	338
15.2	多機能タイマのブロックダイアグラム	340
15.3	多機能タイマの端子	344
15.4	多機能タイマのレジスタ	347
15.4.1	コンペアクリアバッファレジスタ (CPCLRB) とコンペアクリアレジスタ (CPCLR)	351
15.4.2	タイマデータレジスタ (TCDT)	353
15.4.3	タイマ状態制御レジスタ (TCCSH, TCCSL)	354
15.4.4	アウトプットコンペアバッファレジスタ (OCCPB0 ~ OCCPB5)/ アウトプットコンペアレジスタ (OCCP0 ~ OCCP5)	360
15.4.5	コンペア制御レジスタ (OCS0 ~ OCS5)	362
15.4.6	インプットキャプチャレジスタ (IPCP0 ~ IPCP3)	367
15.4.7	インプットキャプチャ状態制御レジスタ (ICS23, PICS01)	368
15.4.8	16 ビットタイマレジスタ (TMRR0 ~ TMRR2)	375
15.4.9	16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2)	376
15.4.10	波形制御レジスタ (SIGCR)	380
15.5	多機能タイマ割込み	382
15.6	多機能タイマの動作	387
15.6.1	16 ビットフリーランタイマの動作	388
15.6.2	16 ビットアウトプットコンペアの動作	395
15.6.3	16 ビットインプットキャプチャの動作	401
15.6.4	波形ジェネレータの動作	403
15.6.5	タイマモードの動作	406
15.6.6	デッドタイムタイマモード時の動作	408
15.6.7	DTTI 端子制御の動作	412
15.7	多機能タイマ使用上の注意	414
第 16 章	遅延割込み発生モジュール	417
16.1	遅延割込み発生モジュールの概要	418
16.2	遅延割込み発生モジュールのレジスタ	419
16.3	遅延割込み発生モジュールの動作説明	420
16.4	遅延割込み発生モジュールの使用上の注意	421
第 17 章	DTP/ 外部割込み	423
17.1	DTP/ 外部割込みの概要	424
17.2	DTP/ 外部割込みのブロックダイアグラム	426
17.3	DTP/ 外部割込みの端子	428
17.4	DTP/ 外部割込み回路のレジスタ	430
17.4.1	DTP/ 外部割込み要因レジスタ (EIRR)	431
17.4.2	DTP/ 外部割込み許可レジスタ (ENIR)	432
17.4.3	要求レベル設定レジスタ (ELVR)	434
17.5	DTP/ 外部割込みの動作説明	436
17.5.1	外部割込み機能	439
17.5.2	DTP 機能	440
17.6	DTP/ 外部割込みの使用上の注意	441

第 18 章	8/10 ビット A/D コンバータ	443
18.1	8/10 ビット A/D コンバータの概要	444
18.2	8/10 ビット A/D コンバータのブロックダイアグラム	446
18.3	8/10 ビット A/D コンバータの構成	449
18.3.1	A/D 制御ステータスレジスタ上位 (ADCS1)	451
18.3.2	A/D 制御ステータスレジスタ下位 (ADCS0)	455
18.3.3	A/D データレジスタ (ADCR0/ADCR1)	457
18.3.4	A/D セットアップレジスタ (ADSR0/ADSR1)	458
18.3.5	アナログ入力許可レジスタ (ADER0/ADER1)	463
18.4	8/10 ビット A/D コンバータの割込み	465
18.5	8/10 ビット A/D コンバータの動作説明	466
18.5.1	単発変換モード	467
18.5.2	連続変換モード	469
18.5.3	停止変換モード	471
18.5.4	EI ² OS 機能を使用した変換動作	473
18.5.5	A/D 変換データ保護機能	474
18.6	8/10 ビット A/D コンバータ使用上の注意	477
第 19 章	D/A コンバータ	479
19.1	D/A コンバータの概要	480
19.2	D/A コンバータのブロックダイアグラム	481
19.3	D/A コンバータの構成	482
19.4	D/A コンバータレジスタ	483
19.4.1	D/A コンバータレジスタ 1(DAT1)	484
19.4.2	D/A コンバータレジスタ 0(DAT0)	485
19.4.3	D/A 制御レジスタ 1(DACR1)	486
19.4.4	D/A 制御レジスタ 0(DACR0)	487
第 20 章	UART	489
20.1	UART の概要	490
20.2	UART のブロックダイアグラム	492
20.3	UART の端子	495
20.4	UART のレジスタ	498
20.4.1	シリアル制御レジスタ (SCR0/SCR1)	499
20.4.2	シリアルモードレジスタ (SMR0/SMR1)	501
20.4.3	シリアルステータスレジスタ (SSR0/SSR1)	503
20.4.4	シリアル入力データレジスタ (SIDR0/SIDR1), シリアル出力データレジスタ (SODR0/SODR1)	506
20.4.5	通信プリスケアラ制御レジスタ (CDCR)	508
20.5	UART の割込み	510
20.5.1	受信割込み要求の発生とフラグセットのタイミング	512
20.5.2	送信割込み要求の発生とフラグセットのタイミング	514
20.6	UART のボーレート	516
20.6.1	専用ボーレートジェネレータによるボーレート	518
20.6.2	内部タイマ (16 ビットリロードタイマ) によるボーレート	521
20.6.3	外部クロックによるボーレート	523
20.7	UART の動作説明	524
20.7.1	非同期モード (動作モード 0, 1) での動作	526
20.7.2	クロック同期モード (動作モード 2) での動作	529

20.7.3	双方向通信機能 (ノーマルモード)	531
20.7.4	マスタ / スレーブ型通信機能 (マルチプロセッサモード)	533
20.8	UART 使用上の注意	536
第 21 章	ROM 修正機能	537
21.1	ROM 修正機能の概要	538
21.2	ROM 修正機能のブロックダイアグラム	539
21.3	ROM 修正機能レジスタ	540
21.3.1	プログラムアドレス検出レジスタ (PADR0/PADR1)	541
21.3.2	プログラムアドレス検出制御ステータスレジスタ (PACSR)	542
21.4	ROM 修正機能の動作説明	544
21.5	ROM 修正機能使用例	545
第 22 章	ROM ミラー機能選択モジュール	549
22.1	ROM ミラー機能選択モジュールの概要	550
22.2	ROM ミラー機能選択レジスタ (ROMM)	551
第 23 章	512K/1024K ビットフラッシュメモリ	553
23.1	512K/1024K ビットフラッシュメモリの概要	554
23.2	512K/1024K フラッシュメモリのレジスタとセクタ構成	556
23.3	フラッシュメモリ制御ステータスレジスタ (FMCS)	558
23.4	フラッシュメモリ自動アルゴリズム起動方法	561
23.5	自動アルゴリズム実行状態の確認	562
23.5.1	データポーリングフラグ (DQ7)	564
23.5.2	トグルビットフラグ (DQ6)	566
23.5.3	タイミングリミット超過フラグ (DQ5)	567
23.5.4	セクタ消去タイマフラグ (DQ3)	568
23.6	フラッシュメモリ書込み / 消去の詳細説明	569
23.6.1	フラッシュメモリを読み出し / リセット状態にする	570
23.6.2	フラッシュメモリヘータを書き込む	571
23.6.3	フラッシュメモリのデータを消去する (チップ消去)	573
23.6.4	フラッシュメモリの任意のデータを消去する (セクタ消去)	574
23.6.5	フラッシュメモリのセクタ消去を一時停止する	576
23.6.6	フラッシュメモリのセクタ消去を再開する	577
23.7	フラッシュセキュリティの特長	578
付録	579
付録 A	I/O マップ	580
付録 B	シリアル書込み例	587
B.1	シリアルオンボード書込み (富士通標準) の基本構成	588
B.2	シリアル書込み接続例 (ユーザ電源)	591
B.3	シリアル書込み接続例 (ライタ電源)	593
B.4	フラッシュマイコンプログラマとの最小限の接続例 (ユーザ電源)	595
B.5	フラッシュマイコンプログラマとの最小限の接続例 (ライタ電源)	597
付録 C	命令	599
C.1	命令の種類	600
C.2	アドレッシング	601
C.3	直接アドレッシング	603
C.4	間接アドレッシング	610

C.5	実行サイクル数.....	617
C.6	実効アドレスフィールド	619
C.7	命令一覧表の読み方.....	621
C.8	F ² MC-16LX 命令一覧表	624
C.9	命令マップ	638
索引	661

本版での主な変更内容

ページ		変更内容 (詳細は本文を参照してください。)
		型格を追加 (MB90F828B) 「第 6 章 クロックスーパーバイザ」を追加 パッケージを変更 (FPT-80P-M05 FPT-80P-M21) (FPT-80P-M11 FPT-80P-M22)
7	第 1 章 概要 1.3 MB90820B シリーズ のブロックダイアグラム	図 1.3-1 MB90820B シリーズの全体ブロックダイアグラムを変更
62	第 3 章 リセット	表 3.1-1 リセット要因 に クロックスーパーバイザリセットを追加
63	3.1 リセットの概要	リセット要因に クロックスーパーバイザリセットを追加
64	第 3 章 リセット 3.2 リセット要因と 発振安定待ち時間	表 3.2-1 リセット要因と発振安定待ち時間にクロックスーパーバイザ リセットを追加
67	第 3 章 リセット 3.4 リセット動作	図 3.4-1 リセット動作フローに クロックスーパーバイザリセットを追加
69	第 3 章 リセット	図 3.5-1 リセット要因ビットのブロックダイアグラムを変更
70	3.5 リセット要因ビット	表 3.5-1 リセット要因フラグビットとリセット要因の対応 に クロック スーパーバイザリセットを追加
75	第 4 章 クロック 4.1 クロック	図 4.1-1 クロック供給マップ を変更
76	第 4 章 クロック 4.2 クロック生成部の ブロックダイアグラム	・クロックセクタ ・動作クロックセクタ に変更 図 4.2-1 クロック生成部のブロックダイアグラム を変更
77	第 4 章 クロック 4.2 クロック生成部のブ ロックダイアグラム	クロックセクタ 動作クロックセクタ に変更
102	第 6 章 低消費電力モード 6.2 低消費電力制御回路 のブロックダイアグラム	図 6.2-1 低消費電力制御回路のブロックダイアグラム を変更
544	第 21 章 ROM 修正機能 21.4 ROM 修正機能の 動作説明	ROM 修正機能の動作説明に < 注意事項 > を追加
551	第 22 章 ROM ミラー機能 選択モジュール 22.2 ROM ミラー機能選 択レジスタ (ROMM)	ROM ミラー機能選択レジスタ (ROMM) のアドレス 1, アドレス 2 の 表を変更

ページ		変更内容 (詳細は本文を参照してください。)
584	付録 付録 A I/O マップ	表 A-1 I/O マップ (5/7) のアドレス 00008A _H を変更
599 ~ 659	付録 付録 C 命令	「付録 C 命令」全体を変更

変更箇所は、本文中のページ左側の によって示しています。

第1章

概要

MB90820B シリーズの特長や基本的な仕様について説明します。

- 1.1 MB90820B シリーズの特長
- 1.2 MB90820B シリーズの品種構成
- 1.3 MB90820B シリーズのブロックダイヤグラム
- 1.4 端子配列図
- 1.5 外形寸法図
- 1.6 端子機能説明
- 1.7 入出力回路形式
- 1.8 デバイス取扱い上の注意

1.1 MB90820B シリーズの特長

MB90820B シリーズは、高速リアルタイム処理を必要とするアプリケーション用に設計された 16 ビット汎用マイクロコントローラであり、さまざまな工業用機械やモータ（交流インダクションモータやブラシレス直流モータ）の制御に適した製品です。このマイクロコントローラは、交流 / 直流モータ制御用の多機能タイマと、さまざまな型の波形を生成できる直流モータ制御用のマルチパルスジェネレータから構成されています。

命令セットは、コントローラのアプリケーションに対して最適化されるように設計されており、かつ F²MC-16LX ファミリの AT アーキテクチャを継承していることにより、広範囲にわたる制御作業を効率的かつ高速で処理することを可能とします。

■ MB90820B シリーズの特長

- クロック
 - 組込み PLL クロック逡倍回路
 - 動作クロック (PLL クロック) としては、原発振の 2 分周かあるいは原発振の 1 から 4 または 6 逡倍した周波数 (原発振が 4MHz の場合は 4MHz から 16MHz または 24MHz) から選択できます。
 - 最小命令実行時間は 42ns (原発振が 4MHz の場合は、PLL クロックが 6 逡倍、V_{CC} が 5.0V)
- 16M バイトの CPU アドレス空間
 - 24 ビットの内部アドレス
- アプリケーションに対して最適化された命令セット
 - 豊富なデータタイプ (ビット、バイト、ワード、ロングワード)
 - 豊富なアドレス指定モード (23 種類)
 - 高いコード効率
 - 32 ビットアキュムレータにより実現される高精度演算
- 高級言語 (C) とマルチタスク処理用に設計された命令セット
 - システムスタックポインタを採用
 - 高度なポインタ間接命令
 - バレルシフト命令
- プログラムパッチ機能 (2 アドレスポインタ)
- 向上した実行速度
 - 4 バイト命令キュー
- 強力な割込み機能
 - 優先順位を設定可能 : 8 レベル
 - 32 の強力な割込み要因

- CPU 動作に依存しない自動データ転送機能
 - 拡張インテリジェント I/O サービス機能 (EI²OS)
 - 最大 16 チャンネル
- 低消費電力モード (スタンバイモード)
 - スリープモード (CPU 動作クロックが停止しているモード)
 - タイムベースタイマモード (発振とタイムベースタイマ以外が停止しているモード)
 - ストップモード (発振が停止しているモード)
 - CPU 間欠動作モード
- パッケージ
 - LQFP-80 (FPT-80P-M21: 0.50mm ピッチ)
 - LQFP-80 (FPT-80P-M22: 0.65mm ピッチ)
 - QFP-80 (FPT-80P-M06: 0.80mm ピッチ)
- プロセス
CMOS

■ リソース

- I/O ポート
最大 66 ポート
- 18 ビットタイムベースカウンタ / ウォッチドッグタイマ : 1 チャンネル
- ウォッチドッグタイマ : 1 チャンネル
- PWC: 2 チャンネル
- 16 ビットリロードタイマ : 2 チャンネル
- 16 ビット PPG タイマ : 3 チャンネル
- 多機能タイマ (交流 / 直流モータ制御用) : 1 チャンネル
 - 16 ビットフリーランタイマ (アップモードまたはアップ / ダウンモードの選択が可能であり、バッファ付き) : 1 チャンネル
 - 16 ビットアウトプットコンペア (バッファ付き) : 6 チャンネル
 - 16 ビットインプットキャプチャ : 4 チャンネル
 - 16 ビット PPG タイマ : 1 チャンネル
 - 波形ジェネレータ (16 ビットタイマ : 3 チャンネル, 3 位相波形またはデッドタイム)
- UART: 2 チャンネル
 - 全二重の二重バッファ (8 ビット長) を搭載
 - クロック非同期転送またはクロック同期転送 (スタートビットとストップビットを使用) を選択的に使用可能
- DTP/ 外部割込み回路 : 8 チャンネル
拡張インテリジェント I/O サービス (EI²OS) の起動, および外部入力をトリガとする外部割込み生成モジュール

- 遅延割込み生成モジュール
タスク切換え割込み要求を生成
- 8/10 ビット A/D コンバータ :16 チャンネル
8/10 ビット分解能選択可能
- 8 ビット D/A コンバータ :2 チャンネル
- クロックスーパーバイザ

1.2 MB90820B シリーズの品種構成

MB90820B シリーズの品種は 3 種類あります。表 1.2-1 に品種構成に共通機能を示します。

■ MB90820B シリーズの品種構成

表 1.2-1 MB90820B シリーズの品種構成 (1 / 2)

品種名 項目	MB90V820B	MB90F822B	MB90F823B	MB90F828B	MB90822B	MB90823B
分類	開発 / 評価 用製品	量産製品 (フラッシュセキュリティ付き フラッシュ ROM)			量産製品 (マスク ROM)	
ROM 容量		64K バイト	128K バイト		64K バイト	128K バイト
RAM 容量	16K バイト	4K バイト		8K バイト	4K バイト	
CPU 機能	命令数 : 351 最小実行時間 : 42ns/4MHz (PLL: 4MHz × 6) アドレス指定モード : 23 データビット長 : 1, 8, 16 ビット 最大メモリ空間 : 16M バイト					
I/O ポート	I/O ポート (CMOS) : 66					
PWC	パルス幅カウンタタイマ : 2 チャンネル タイマ機能 (3 種類の内部クロックからカウンタタイマを選択可能) 各種パルス幅計測機能 ("H" パルス幅 , "L" パルス幅 , ↑↓ 周期 , ↓↑ 周期 , ↑↑ 周期 , ↓↓ 周期)					
UART	UART : 2 チャンネル 全二重の二重バッファ (8 ビット) を搭載 クロック非同期転送またはクロック同期転送 (スタートビットとストップビット使用) を 選択的に使用可能 転送は 1 対 1 (双方向通信) または 1 対 n (マスタスレーブ通信)					
16 ビット リロードタイマ	リロードタイマ : 2 チャンネル リロードモード , シングルショットモードまたはイベントカウントモードを選択可能					
16 ビット PPG タイマ	PPG タイマ : 3 チャンネル PWM モードまたはシングルショットモードを選択可能 多機能タイマと一緒に動作することが可能。または単独での動作も可能。					
多機能タイマ (交流 / 直流モー タ制御用)	16 ビットフリーランタイマ (アップモードまたはアップ / ダウンモード選択可能。バッファ付き) : 1 チャンネル 16 ビットアウトプットコンペア: 6 チャンネル 16 ビットインプットキャプチャ: 4 チャンネル 16 ビット PPG タイマ: 1 チャンネル 波形ジェネレータ (16 ビットタイマ : 3 チャンネル , 3 位相波形またはデッドタイム)					
8/10 ビット A/D コンバータ	8/10 ビット分解能 (16 チャンネル) 変換時間 : 最小 3μs(24 MHz 内部クロック , サンプル時間も含む)					

表 1.2-1 MB90820B シリーズの品種構成 (2 / 2)

品種名 項目	MB90V820B	MB90F822B	MB90F823B	MB90F828B	MB90822B	MB90823B
クロック スーパバイザ	なし			あり	なし	
8 ビット D/A コンバータ	8 ビット分解能 (2 チャンネル)					
DTP/ 外部割込み	独立した 8 チャンネル 選択可能な要因 :↑, ↓, "L" レベルまたは "H" レベル					
低消費電力	停止モード, スリープモード, CPU 間欠動作モード					
パッケージ	PGA299	LQFP-80 (FPT-80P-M21: 0.50mm ピッチ) LQFP-80(FPT-80P-M22: 0.65mm ピッチ) QFP-80(FPT-80P-M06: 0.80mm ピッチ)				
動作電源電圧	4.5V ~ 5.5V *1	3.5V ~ 5.5V: A/D コンバータおよび D/A コンバータを使用していないときの 通常動作 4.0V ~ 5.5V: D/A コンバータを使用していないときの通常動作 4.5V ~ 5.5V: 通常動作				
プロセス	CMOS					
エミュレータ 専用電源 *2	あり					

*1: MB90V820B に対する保証は, 動作温度 0 ~ +25 にのみ適用されます。

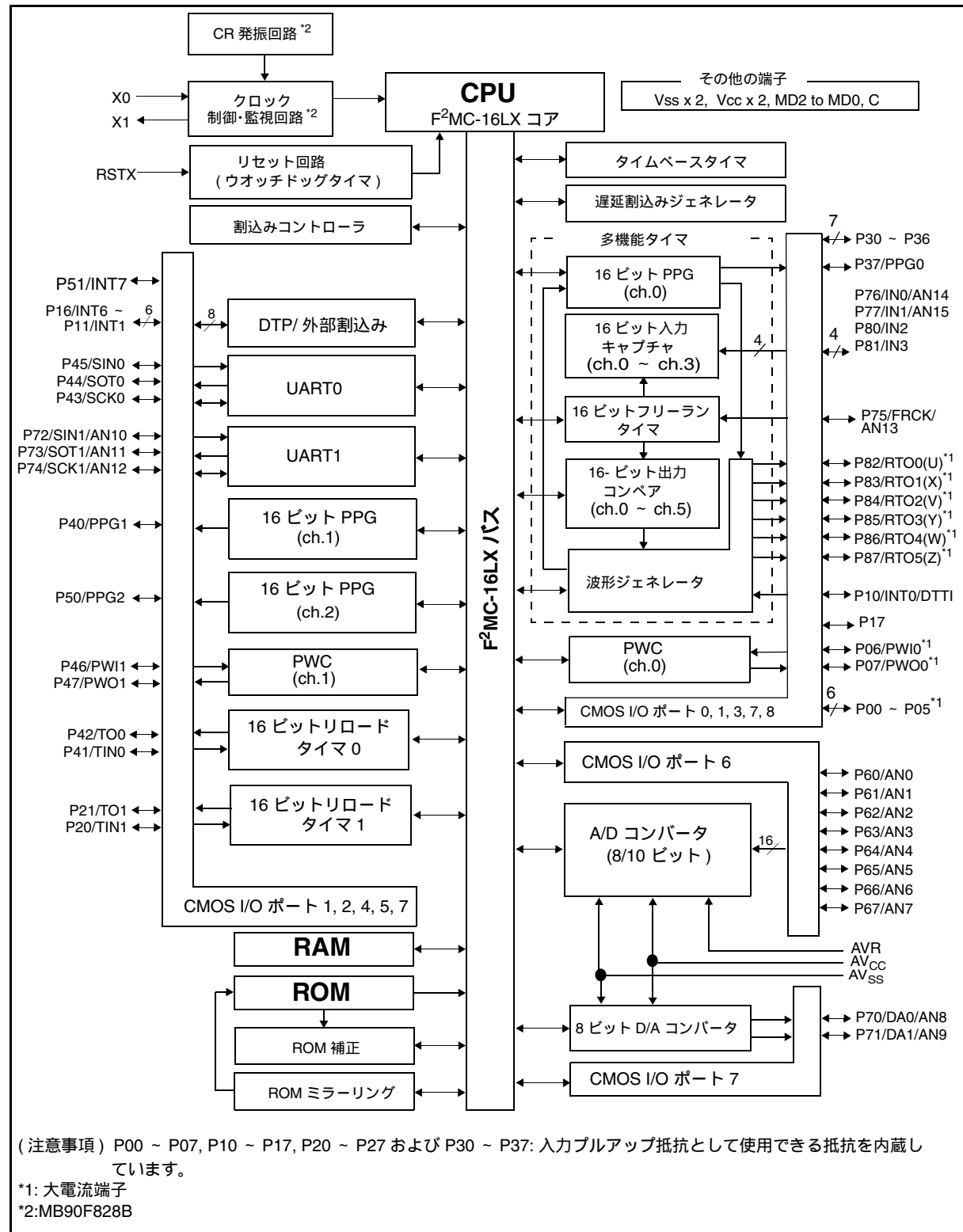
*2: エミュレータ (MB2147-01) をご使用いただく際のジャンプスイッチ (TOOL, VCC) の設定です。詳細につきましては, MB2147-01 または MB2147-20 ハードウェアマニュアル (「3.3 エミュレータ専用電源切換え」) を参照してください。

1.3 MB90820B シリーズのブロックダイアグラム

MB90820B シリーズの全体ブロックダイアグラムを図 1.3-1 に示します。

■ MB90820B シリーズのブロックダイアグラム

図 1.3-1 MB90820B シリーズの全体ブロックダイアグラム

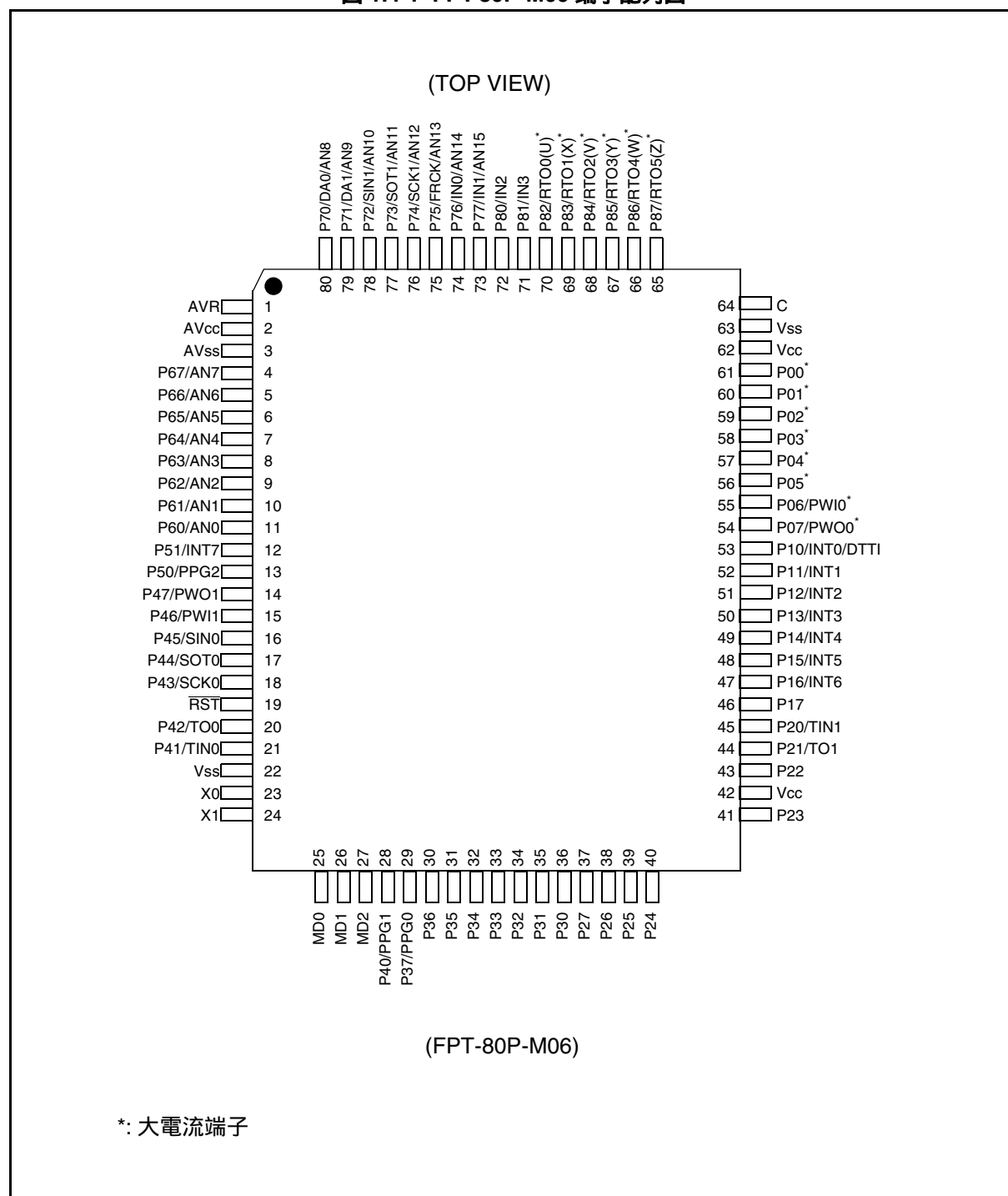


1.4 端子配列図

MB90820B シリーズの端子配列図を、図 1.4-1, 図 1.4-2 に示します。

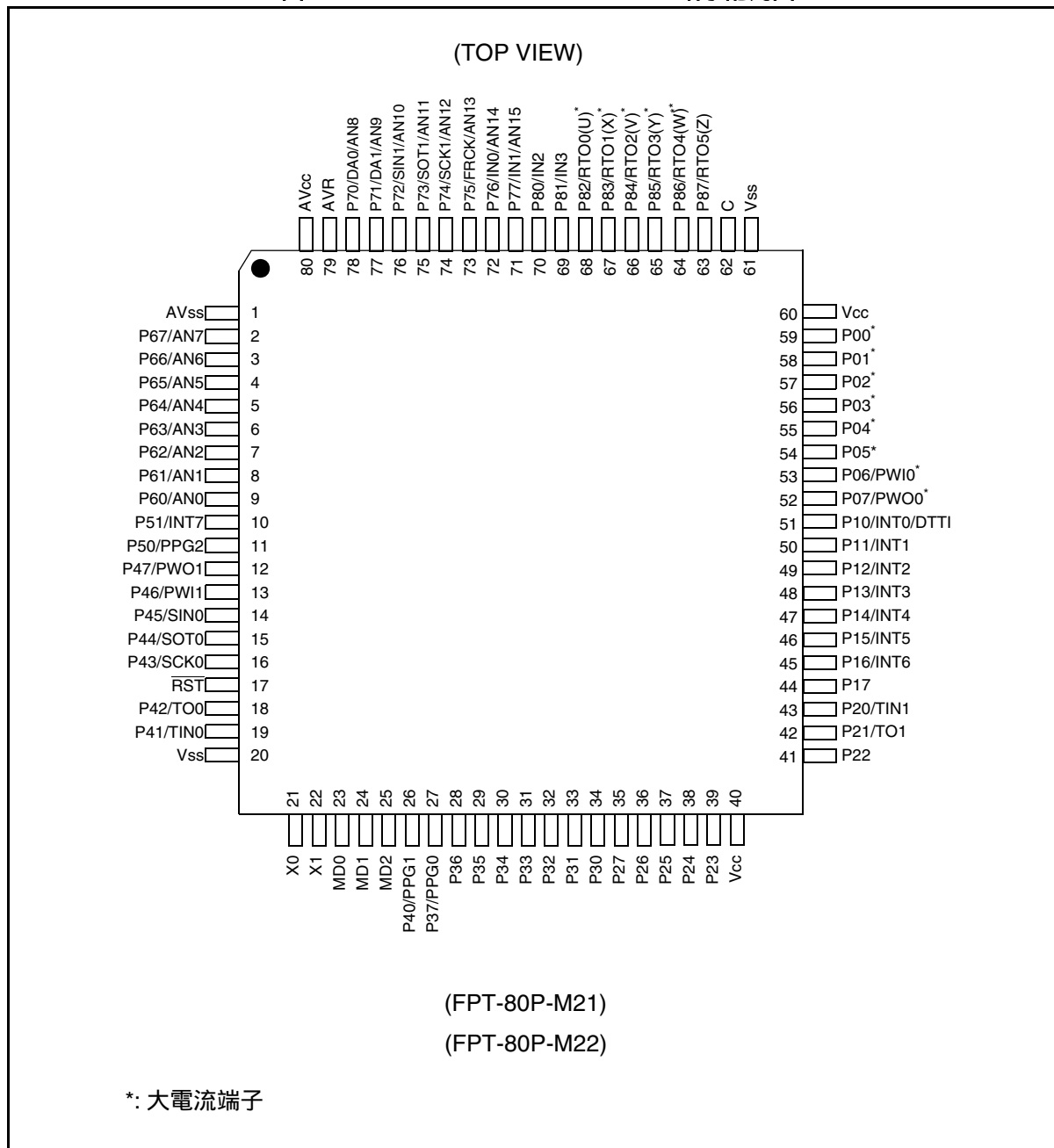
■ FPT-80P-M06 端子配列図

図 1.4-1 FPT-80P-M06 端子配列図



■ FPT-80P-M21/FPT-80P-M22 端子配列図

図 1.4-2 FPT-80P-M21/FPT-80P-M22 端子配列図



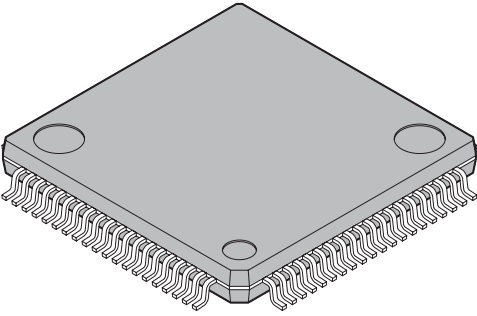
1.5 外形寸法図

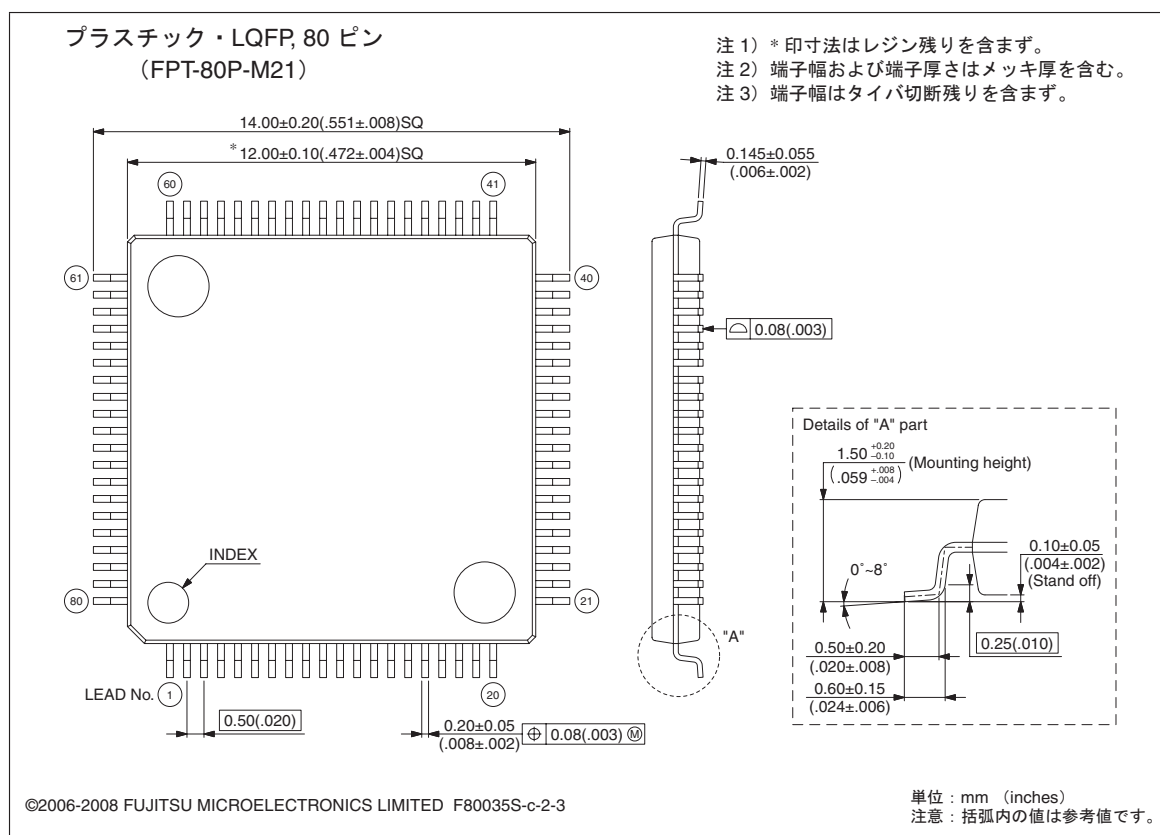
MB90820B シリーズには、3 種類のパッケージが用意されています。

図 1.5-1 ~ 図 1.5-3 に外形寸法を示します。

■ FPT-80P-M21 の外形寸法図

図 1.5-1 FPT-80P-M21 の外形寸法図

<p>プラスチック・LQFP, 80 ピン</p>  <p>(FPT-80P-M21)</p>	リードピッチ	0.50 mm
	パッケージ幅× パッケージ長さ	12.0 mm × 12.0 mm
	リード形状	ガルウィング
	封止方法	プラスチックモールド
	取付け高さ	1.70 mm Max
	質量	0.47 g
	コード (参考)	P-LFQFP80-12×12-0.50



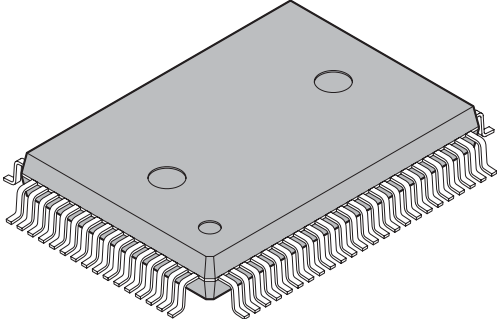
最新の外形寸法図については、下記の URL にてご確認ください。

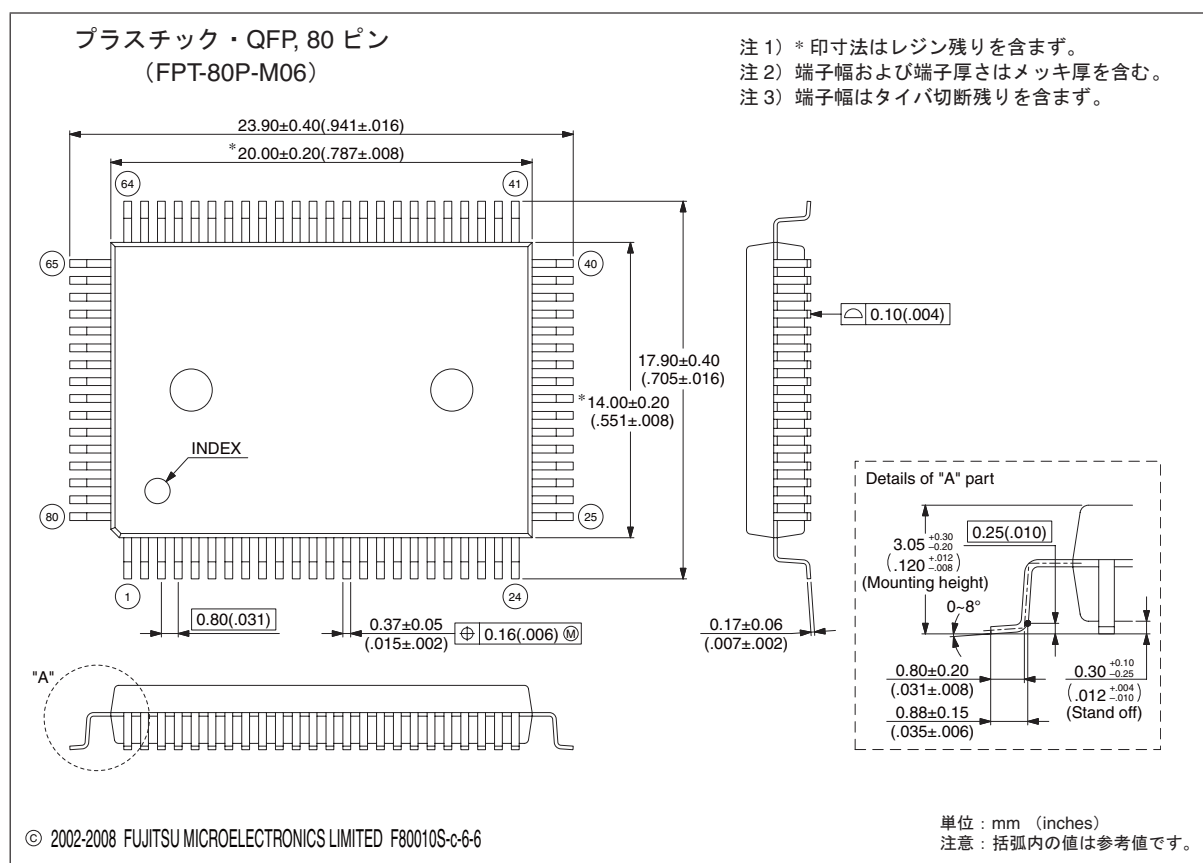
<http://edevic.fujitsu.com/package/jp-search/>

MB90820B シリーズ

■ FPT-80P-M06 の外形寸法図

図 1.5-2 FPT-80P-M06 の外形寸法図

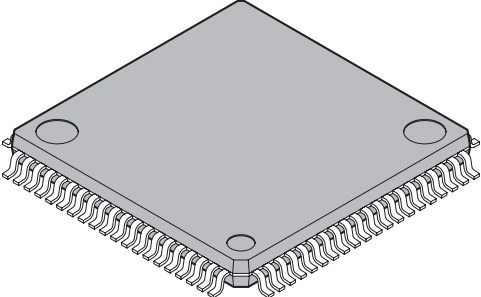
<p>プラスチック・QFP, 80 ピン</p>  <p>(FPT-80P-M06)</p>	リードピッチ	0.80mm
	パッケージ幅× パッケージ長さ	14.00 × 20.00mm
	リード形状	ガルウィング
	封止方法	プラスチックモールド
	取付け高さ	3.35mm MAX
	コード (参考)	P-QFP80-14×20-0.80

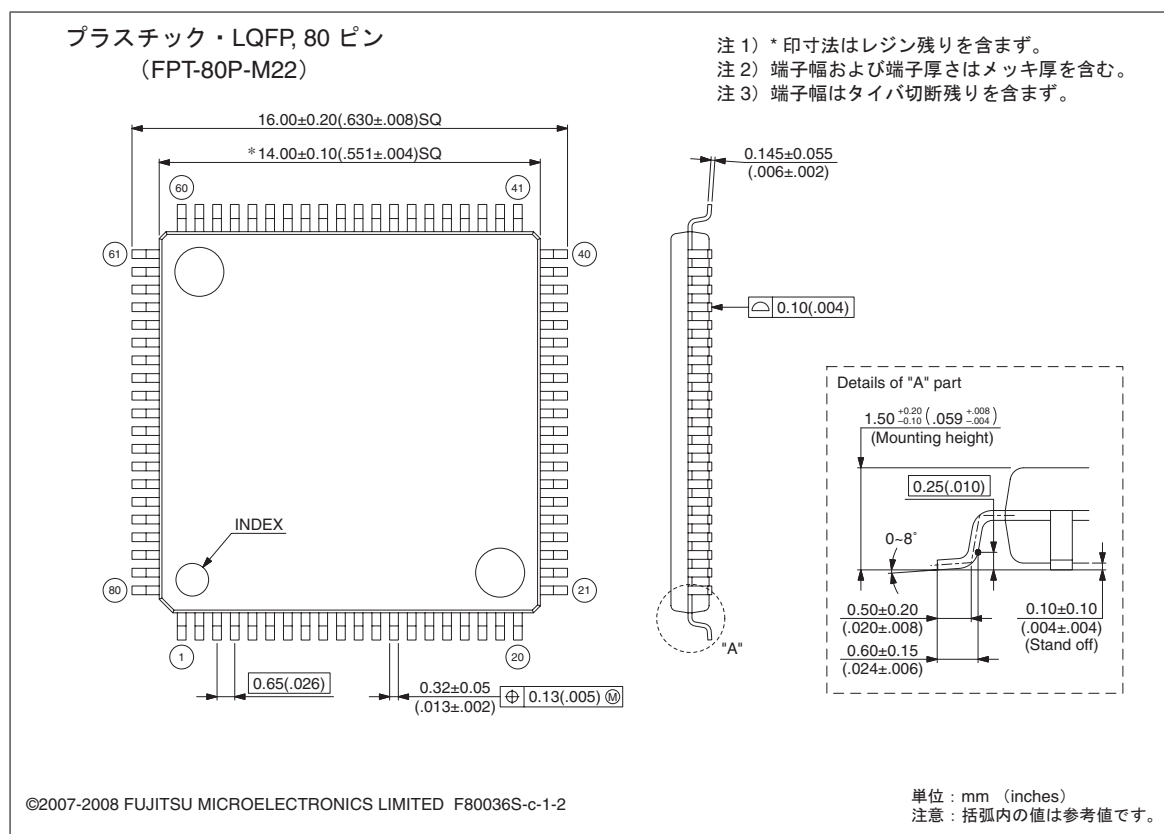


最新の外形寸法図については、下記の URL にてご確認ください。
<http://edevice.fujitsu.com/package/jp-search/>

■ FPT-80P-M22 の外形寸法図

図 1.5-3 FPT-80P-M22 の外形寸法図

<p>プラスチック・LQFP, 80 ピン</p>  <p>(FPT-80P-M22)</p>	リードピッチ	0.65 mm
	パッケージ幅× パッケージ長さ	14.00 mm × 14.00 mm
	リード形状	ガルウィング
	封止方法	プラスチックモールド
	取付け高さ	1.70 mm Max
	質量	0.62 g
	コード (参考)	P-LFQFP80-14×14-0.65



最新の外形寸法図については, 下記の URL にてご確認ください。
<http://edevice.fujitsu.com/package/jp-search/>

1.6 端子機能説明

表 1.6-1 に入出力端子と端子ごとの機能を示します。また表 1.7-1 に入出力回路形式を示します。

表 1.6-1 の " 入出力回路形式 " 欄にある文字は表 1.7-1 の " 分類 " にある文字を参照しています。

■ I/O 端子と端子機能

表 1.6-1 端子機能 (1 / 4)

端子番号		端子名	入出力回路形式 ^{*3}	リセット時の端子状態	機能
LQFP ^{*1}	QFP ^{*2}				
21, 22	23, 24	X0, X1	A	発振中	発振入力端子
17	19	$\overline{\text{RST}}$	B	リセット入力	外部リセット入力端子
59 ~ 54	61 ~ 56	P00 ~ P05	C	ポート入力	汎用入出力ポート
53	55	P06	C		汎用入出力ポート
		PWI0			PWC 0 信号入力端子
52	54	P07	C		汎用入出力ポート
		PWO0			PWC 0 信号出力端子
51	53	P10	D		汎用入出力ポート
		INT0			割込み要求入力 ch.0 として使用可能です。スタンバイモード時に EN0 に "1" を設定した場合入力が有効になります。
		DTTI			RTO0 ~ RTO5 端子使用時のレベル固定入力端子です。波形ジェネレータで入力ピットを有効にすると、この機能は有効になります。
50 ~ 45	52 ~ 47	P11 ~ P16	D		汎用入出力ポート
		INT1 ~ INT6			割込み要求入力 ch.1 ~ ch.6 として使用可能です。スタンバイモード時に EN1 ~ EN6 に "1" を設定した場合入力が有効になります。
44	46	P17	D		汎用入出力ポート
43	45	P20	D		汎用入出力ポート
		TIN1			リロードタイマ 1 用の外部クロック入力端子
42	44	P21	D		汎用入出力ポート
		TO1			リロードタイマ 1 用のイベント出力端子
41, 39 ~ 35	43, 41 ~ 37	P22 ~ P27	D		汎用入出力ポート
34 ~ 28	36 ~ 30	P30 ~ P36	E		汎用入出力ポート

表 1.6-1 端子機能 (2 / 4)

端子番号		端子名	入出力回路形式 ^{*3}	リセット時の端子状態	機能
LQFP ^{*1}	QFP ^{*2}				
27	29	P37	E	ポート入力	汎用入出力ポート
		PPG0			PPG ch.0 用の出力端子。PPG ch.0 が出力を有効にした場合、この機能は有効になります。
26	28	P40	F		汎用入出力ポート
		PPG1			PPG ch.1 用の出力端子。PPG ch.1 が出力を有効にした場合、この機能は有効になります。
19	21	P41	F		汎用入出力ポート
		TIN0			リロードタイマ 0 用の外部クロック入力端子。
18	20	P42	F		汎用入出力ポート
		TO0			リロードタイマ 0 用のイベント出力端子。
16	18	P43	F		汎用入出力ポート
		SCK0			ch.0 用のシリアルクロック入出力端子。UART0 がクロック出力を有効にした場合この機能は有効になります。
15	17	P44	F		汎用入出力ポート
		SOT0			ch.0 用のシリアルデータ出力端子。UART0 がデータ出力を有効にすると、この機能は有効になります。
14	16	P45	G		汎用入出力ポート
		SIN0			ch.0 用のシリアルデータ入力端子。UART0 が入力動作中は、シリアルデータ以外は入力しないでください。CMOS 入力はユーザプログラムで選択可能です。
13	15	P46	F		汎用入出力ポート
		PWI1			PWC 1 信号入力端子
12	14	P47	F		汎用入出力ポート
		PWO1			PWC 1 信号出力端子
11	13	P50	F		汎用入出力ポート
		PPG2			PPG ch.2 用の出力端子。PPG ch.2 が出力を有効にすると場合、この機能は有効になります。
10	12	P51	F		汎用入出力ポート
		INT7			割込み要求入力 ch.7 として使用できます。スタンバイモード時に EN7 に "1" を設定した場合、入力が有効になります。
9 ~ 2	11 ~ 4	P60 ~ P67	H	アナログ入力	汎用入出力ポート
		AN0 ~ AN7			A/D コンバータアナログ入力端子。アナログ入力 (ADER0) を有効にすると、この機能は有効になります。

表 1.6-1 端子機能 (3 / 4)

端子番号		端子名	入出力回路形式 *3	リセット時の端子状態	機能
LQFP*1	QFP*2				
78, 77	80, 79	P70, P71	I	アナログ入力	汎用入出力ポート
		DA0, DA1			D/A コンバータアナログ出力端子。D/A コンバータを有効にすると、この端子は有効になります。
		AN8, AN9			A/D コンバータアナログ入力端子。アナログ入力 (ADER1) を有効にすると、この機能は有効になります。
76	78	P72	J	アナログ入力	汎用入出力ポート
		SIN1			UART1 用のシリアルデータ入力端子。UART1 入力動作中は、シリアルデータ以外は入力しないでください。CMOS 入力はユーザプログラムで選択可能です。
		AN10			A/D コンバータアナログ入力端子。アナログ入力 (ADER1) を有効にすると、この端子は有効になります。
75	77	P73	K		汎用入出力ポート
		SOT1			UART1 用のシリアルデータ出力端子。UART1 がデータ出力を有効にすると、この機能は有効になります。
		AN11			A/D コンバータアナログ入力端子。アナログ入力 (ADER1) を有効にすると、この機能は有効になります。
74	76	P74	K		汎用入出力ポート
		SCK1			UART1 用のシリアルクロック入出力端子。UART1 がクロック出力を有効にすると、この機能は有効になります。
		AN12			A/D コンバータアナログ入力端子。アナログ入力 (ADER1) を有効にすると、この端子は有効になります。
73	75	P75	K		汎用入出力ポート
		FRCK			フリーランタイム用の外部クロック入力端子。
		AN13			A/D コンバータアナログ入力端子。アナログ入力 (ADER1) を有効にすると、この端子は有効になります。
72, 71	74, 73	P76, P77	K		汎用入出力ポート
		IN0, IN1			インプットキャプチャチャネル 0, 1 のトリガ入力端子。インプットキャプチャチャネル 0, 1 が入力動作に使用された場合、インプットキャプチャ以外には使用しないでください。
		AN14, AN15			A/D コンバータアナログ入力端子。アナログ入力 (ADER1) を有効にすると、この機能は有効になります。

表 1.6-1 端子機能 (4 / 4)

端子番号		端子名	入出力回路形式 *3	リセット時の端子状態	機能
LQFP*1	QFP*2				
70, 69	72, 71	P80, P81	F	ポート入力	汎用入出力ポート
		IN2, IN3			インプットキャプチャ ch.2, ch.3 のトリガ入力端子。インプットキャプチャ ch.2, ch.3 が入力動作に使用された場合、インプットキャプチャ以外には使用しないでください。
68 ~ 63	70 ~ 65	P82 ~ P87	L		汎用入出力ポート
		RTO0 ~ RTO5			波形ジェネレータ出力端子。これらの端子は、波形ジェネレータに指定した波形を出力します。波形ジェネレータ出力を有効にすると生成されます。
23	25	MD0	M	モード入力	動作モードの指定用入力端子。この端子を Vcc または Vss に直結してください。
24, 25	26, 27	MD1, MD2	N		動作モードの指定用入力端子。この端子を Vcc または Vss に直結してください。
80	2	AV _{CC}	-	電源入力	アナログ回路用の Vcc 電源入力端子。
79	1	AVR	-		A/D コンバータ用の Vref+ 入力端子。電圧は AVcc を超えてはいけません。Vref- は AVss 固定です。
1	3	AV _{ss}	-		アナログ回路用の Vss 電源入力端子。
20, 61	22, 63	Vss	-	電源入力	電源入力端子 (0 V)。
40, 60	42, 62	Vcc	-		電源入力端子 (5 V)。
62	64	C	-	-	電源安定化のための容量端子です。約 0.1 μF セラミックキャパシタに接続してください。

*1: FPT-80P-M21, FPT-80P-M22

*2: FPT-80P-M06

*3: 入出力回路形式については、「1.7 入出力回路形式」を参照してください。

1.7 入出力回路形式

MB90820B シリーズの各端子の入出力回路形式を示します。

■ 入出力回路形式

表 1.7-1 に MB90820B シリーズの各端子の入出力回路形式を示します。

表 1.7-1 入出力回路形式 (1 / 4)

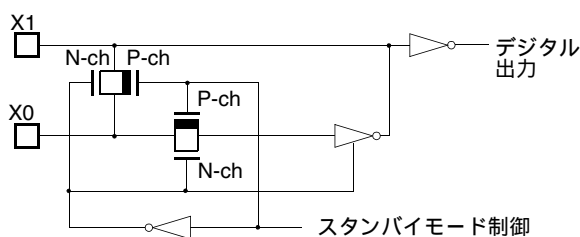
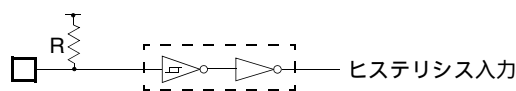
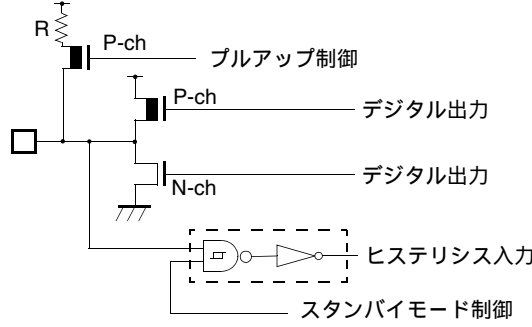
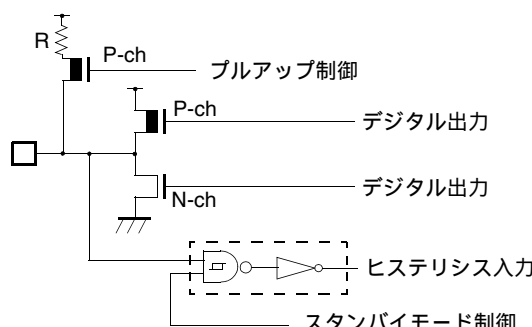
分類	回路	備考
A		メインクロック (メインクロック水晶発振器) ・ 発振帰還抵抗 約 1M Ω
B		・ ヒステリシス入力 ・ ブルアップ抵抗 約 50k Ω
C		・ CMOS 出力 ・ ヒステリシス入力 ・ 選択可能ブルアップ抵抗 約 50k Ω ・ I _{OL} =12mA
D		・ CMOS 出力 ・ ヒステリシス入力 ・ 選択可能ブルアップ抵抗 約 50k Ω ・ I _{OL} =4mA

表 1.7-1 入出力回路形式 (2 / 4)

分類	回路	備考
E	<p>プルアップ制御</p> <p>デジタル出力</p> <p>デジタル出力</p> <p>CMOS 入力</p> <p>スタンバイモード制御</p>	<ul style="list-style-type: none"> CMOS 出力 CMOS 入力 選択可能プルアップ抵抗 約 50kΩ $I_{OL}=4\text{mA}$
F	<p>デジタル出力</p> <p>デジタル出力</p> <p>ヒステリシス入力</p> <p>スタンバイモード制御</p>	<ul style="list-style-type: none"> CMOS 出力 ヒステリシス入力 $I_{OL}=4\text{mA}$
G	<p>デジタル出力</p> <p>デジタル出力</p> <p>ヒステリシス入力</p> <p>CMOS 入力</p> <p>スタンバイモード制御</p>	<ul style="list-style-type: none"> CMOS 出力 ヒステリシス入力 CMOS 入力 (UART0 データ入力端子に選択可能) $I_{OL}=4\text{mA}$
H	<p>デジタル出力</p> <p>デジタル出力</p> <p>CMOS 入力</p> <p>アナログ入力制御</p> <p>アナログ入力</p>	<ul style="list-style-type: none"> CMOS 出力 CMOS 入力 アナログ入力 $I_{OL}=4\text{mA}$

表 1.7-1 入出力回路形式 (3 / 4)

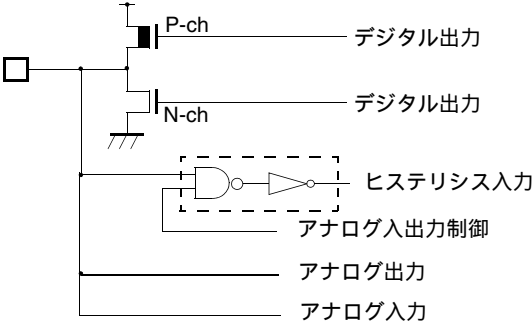
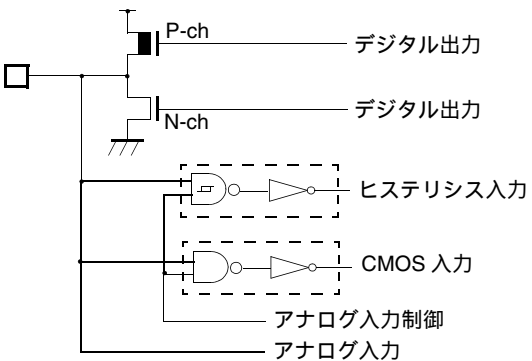
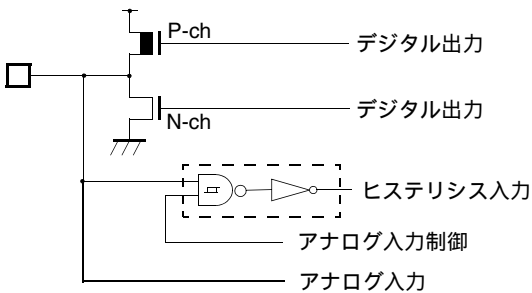
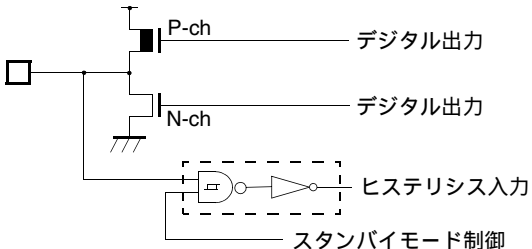
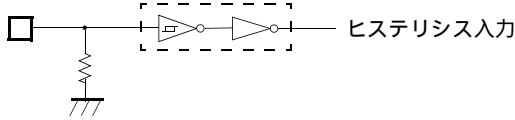
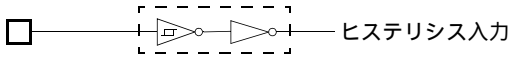
分類	回路	備考
I	 <p>デジタル出力 (P-ch)</p> <p>デジタル出力 (N-ch)</p> <p>ヒステリシス入力</p> <p>アナログ入出力制御</p> <p>アナログ出力</p> <p>アナログ入力</p>	<ul style="list-style-type: none"> CMOS 出力 ヒステリシス入力 アナログ出力 アナログ入力 $I_{OL}=4\text{mA}$
J	 <p>デジタル出力 (P-ch)</p> <p>デジタル出力 (N-ch)</p> <p>ヒステリシス入力</p> <p>CMOS 入力</p> <p>アナログ入力制御</p> <p>アナログ入力</p>	<ul style="list-style-type: none"> CMOS 出力 ヒステリシス入力 CMOS 入力 (UART1 データ入力端子に選択可能) $I_{OL}=4\text{mA}$
K	 <p>デジタル出力 (P-ch)</p> <p>デジタル出力 (N-ch)</p> <p>ヒステリシス入力</p> <p>アナログ入力制御</p> <p>アナログ入力</p>	<ul style="list-style-type: none"> CMOS 出力 ヒステリシス入力 アナログ入力 $I_{OL}=4\text{mA}$
L	 <p>デジタル出力 (P-ch)</p> <p>デジタル出力 (N-ch)</p> <p>ヒステリシス入力</p> <p>スタンバイモード制御</p>	<ul style="list-style-type: none"> CMOS 出力 ヒステリシス入力 $I_{OL}=12\text{mA}$

表 1.7-1 入出力回路形式 (4 / 4)

分類	回路	備考
M		マスク ROM/ 評価用製品 ・ ヒステリシス入力 ・ 選択可能プルアップ抵抗 約 50kΩ フラッシュ製品 ・ CMOS 入力 ・ プルダウン抵抗なし
N		マスク ROM/ 評価用製品 ・ ヒステリシス入力 フラッシュ製品 ・ CMOS 入力

1.8 デバイス取扱い上の注意

デバイスを取扱う際は、下記項目について特に注意してください。

- ラッチアップの防止のために
 - 供給電圧の安定化
 - 電源投入の注意
 - 未使用端子の処理
 - A/D コンバータ、D/A コンバータの未使用端子の接続について
 - 外部クロック使用時の注意
 - 電源端子について (Vcc/Vss)
 - A/D コンバータ、D/A コンバータのアナログ電源投入順序
 - 初期化
 - スタンバイ状態からの復帰
-

■ デバイスの取り扱いについて

● ラッチアップの防止のために

CMOS IC では、次のような状態になるとラッチアップ現象が生じる場合があります。

- 入力端子や出力端子に、Vcc より高い電圧や Vss より低い電圧を印加
- Vcc と Vss の間に定格を超える電圧を印加
- Vcc 電源が供給される前に AVcc 電源を供給

ラッチアップが発生すると電源電流が激増し、素子の熱破壊に至る場合があります。使用に際しては、最大定格を超えないようにしてください。

また、ラッチアップを避けるには、アナログ電源がデジタル電源を超えないようにしてください。

● 供給電圧の安定化

- Vcc 電源電圧の動作保証範囲内においても、電源電圧が急激に変化した場合は、誤動作を起こしますので、Vcc 電源電圧を安定させてください。
- 安定化の基準としましては、商用周波数 (50Hz ~ 60Hz) での Vcc リプル変動 (peak to peak 値) を標準 Vcc 電源電圧値の 10% 以下に、また電源の切換えを行う場合の瞬時変化におきましては、過渡変動率が 0.1V/ms 以下になるように電源電圧を安定させてください。

● 電源投入時の注意

内部降圧回路の誤動作を防止するために、電源投入時の電圧立上り時間を 50 μ s 以上に設定してください。

● 未使用端子の処理

使用していない入力端子を開放のままにしておくと誤作動およびラッチアップによる永久破壊の原因になります。未使用入力端子には最低 2k の抵抗によりプルアップまたはプルダウンの処置をしてください。

未使用入出力端子は出力状態に設定して開放するか、入力状態に設定して入力端子と同じ処理をしてください。

使用していない出力端子がある場合には、開放としてください。各量産品において、ご使用される発振子メーカーに発振評価依頼をしてください。

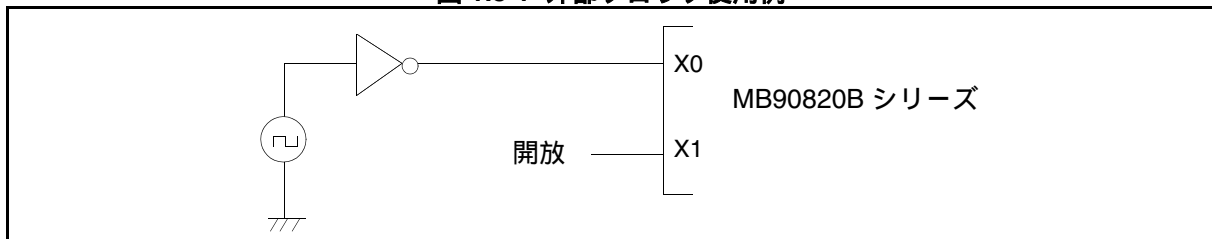
● A/D コンバータ、D/A コンバータの未使用端子の接続について

A/D コンバータおよび D/A コンバータ電源の未使用端子は、 $AV_{CC} = V_{CC}$ 、 $AV_{SS} = AVR = V_{SS}$ となるよう接続してください

● 外部クロック使用時の注意

外部クロックを、使用する際には、X0 端子のみを駆動し、X1 端子は開放としてください。

図 1.8-1 外部クロック使用例



● 電源端子について (V_{CC}/V_{SS})

V_{CC} または V_{SS} が複数ある場合、デバイス設計上はラッチアップなどの誤動作を防止するためにデバイス内部で同電位にすべきもの同士を接続してあります。不要輻射の低減、グラウンドレベルの上昇によるストローク信号の誤動作防止、総出力電流規格を遵守などのための、必ずすべての V_{CC} と V_{SS} を外部で電源およびグラウンドに接続してください。また電流供給源と本デバイスの V_{CC} 端子と V_{SS} 端子は、低インピーダンスで接続してください。

本デバイスの近くで V_{CC} 端子と V_{SS} 端子との間に 0.1 μ F 程度のセラミックコンデンサをバイパスコンデンサとして接続することをお勧めします。

● A/D コンバータ、D/A コンバータのアナログ電源投入順序

A/D コンバータ電源、D/A コンバータ電源 (AV_{CC} 、 AV_{SS} 、 AVR) およびアナログ入力 ($AN0 \sim AN15$) への印加は、必ずデジタル電源 (V_{CC}) の投入後に行ってください。

また、デジタル電源を切断する場合は、必ず A/D コンバータ電源、D/A コンバータ電源およびアナログ入力を切断した後で行ってください。

この場合、 AVR 電圧が AV_{CC} 電圧を超えないようにしてください。

● 初期化

本デバイスには、パワーオンリセットによってのみ初期化する内部レジスタが含まれています。これらのレジスタを初期化するには、電源の再投入を行ってください。

● スタンバイ状態からの復帰

スタンバイ状態に入っているときに電源電圧がスタンバイ RAM 保持電圧より低下すると、デバイスはスタンバイ状態からの復帰に失敗する場合があります。このような場合には、外部リセット端子をノーマル状態にしてデバイスをリセットしてください。

第2章

CPU の機能

MB90820B シリーズの CPU の機能と動作について説明します

- 2.1 CPU
- 2.2 メモリ空間
- 2.3 メモリマップ
- 2.4 アドレス指定
- 2.5 多バイト長データのメモリ配置
- 2.6 レジスタ
- 2.7 専用レジスタ
- 2.8 汎用レジスタ
- 2.9 プリフィックスコード

2.1 CPU

F²MC-16LX ファミリ CPU コアは、民生用や車載用機器などの高速リアルタイム処理が要求されるアプリケーション用に設計された 16 ビット CPU です。F²MC-16LX ファミリ命令セットは制御アプリケーション用に設計されており、各種制御の高速かつ高効率処理が可能です。

F²MC-16LX ファミリ CPU コアは、内部 32 ビットアキュムレータを搭載しているので 16 ビットだけでなく 32 ビット処理も可能です。メモリ空間は最大 16M バイトであり、このメモリ空間へはリニア方式およびバンク方式のいずれでもアクセス可能です。また、命令体系は、F²MC-8L ファミリの AT アーキテクチャをベースにして C 言語対応命令の追加、アドレス指定モードの拡張、乗除算命令の強化、ビット処理の充実化により強化されています。以下に、F²MC-16LX ファミリ CPU の特長を示します。

■ CPU の概要

- 最小命令実行時間：42 ns(原発振 4MHz, PLL クロック 6 逓倍のとき)
 - 最大メモリ空間：16M バイト、リニア / バンク方式アドレス指定にてアクセス
 - コントローラ用途に最適化された命令体系
 - ・ データタイプ：ビット、バイト、ワード、ロングワード
 - ・ アドレス指定モード：23 種類
 - ・ 32 ビットアキュムレータ採用による演算精度の強化
 - ・ 符号付き乗除算、拡張 RETI 命令
 - 割込み機能の強化
 - 8 つのプライオリティレベル (プログラマブル)
 - CPU に依存しない自動転送機能
 - 最大 16 チャンネルまでの拡張インテリジェント I/O サービス
 - C 言語、マルチタスクに対応した命令体系
 - システムスタックポイントの採用、命令セットの対称性、パレルシフト命令
 - 実行速度の向上：4 バイトの命令キュー
-

< 注意事項 >

MB90820B シリーズは、シングルチップモードによる使用となるため、内蔵 ROM、内蔵 RAM、リソースの空間のみへアクセスできほかへはアクセスできません。

2.2 メモリ空間

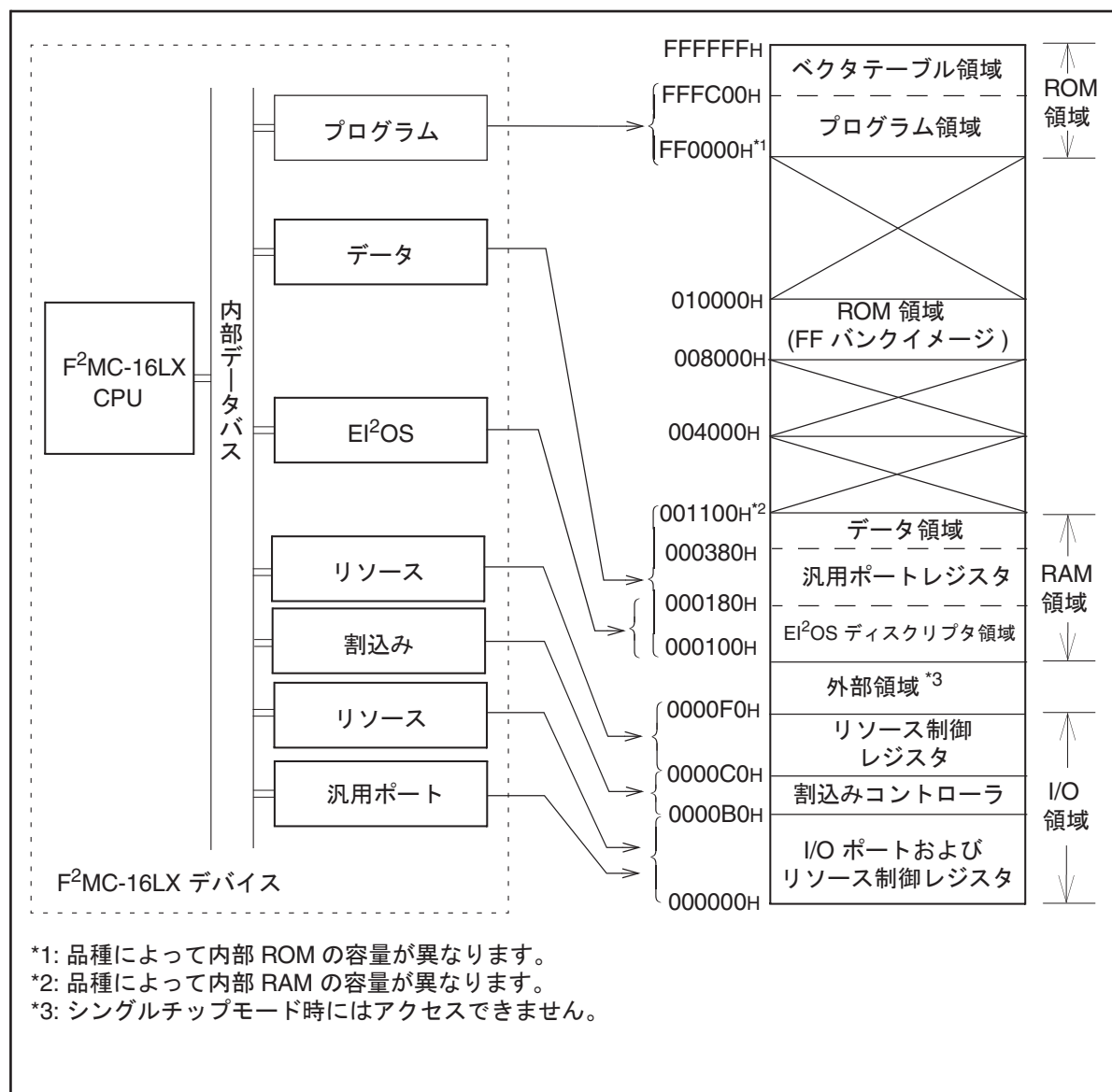
F²MC-16LX ファミリの I/O, プログラムおよびデータは, 16M バイトのメモリ空間に配置されます。メモリ空間の一部は拡張インテリジェント I/O サービス (EI²OS) ディスクリプタ, 汎用レジスタおよびベクタテーブルなどに使用されます。

■ メモリ空間

すべての I/O, プログラムおよびデータは, F²MC-16LX ファミリ CPU が持つ 16M バイトのメモリ空間に配置されます。CPU は, 24 ビットのアдресバスによって示されるメモリ空間のアдресを通じて各リソースへアクセスできます。

MB90820B シリーズのメモリマップを図 2.2-1 に示します

図 2.2-1 F²MC-16LX システムとメモリマップとの関係例



■ ROM 領域

- ベクタテーブル領域 (アドレス: "FFFC00H ~ FFFFFFFH")
 - ・ベクタコール命令, 割込みベクタおよびリセットベクタの各々のベクタテーブルとして使用します。
 - ・ROM 領域の最上位アドレスに割当てられており, 対応する処理ルーチンの開始アドレスをベクタコール命令, 割込みベクタおよびリセットベクタのベクタテーブルのアドレスにデータとして設定します。
- プログラム領域 (アドレス: " ~ FFFBFFH")
 - ・ROM が内部プログラム領域として内蔵されています。
 - ・内蔵 ROM の容量は品種によって異なります。

■ RAM 領域

- データ領域 (アドレス: "000100H ~ ")
 - ・スタティック RAM が内部データ領域として内蔵されています。
 - ・内蔵 RAM の容量は品種によって異なります。
- 汎用レジスタ領域 (アドレス: "000180H ~ 00037FH")
 - ・8ビット, 16ビット, 32ビットの演算や転送に使用する汎用レジスタが配置されています。
 - ・RAM 領域の一部として割当てられるので, 通常の RAM として使用できます。
 - ・汎用レジスタとして使用した場合, 汎用レジスタアドレス指定は短い命令による高速アクセスを可能とします。
- 拡張インテリジェント I/O サービス (EI²OS) ディスクリプタ領域 (アドレス: "000100H ~ 00017FH")
 - ・拡張インテリジェント I/O サービス (EI²OS) の転送モード, I/O のアドレス, 転送数およびバッファアドレスを設定します。
 - ・RAM 領域の一部として割当てられるので, 通常の RAM として使用できます。

■ I/O 領域

- 割込み制御レジスタ領域 (アドレス: "0000B0H ~ 0000BFH")

割込み制御レジスタ (ICR00 ~ ICR15) は, 割込み機能を持つすべてのリソースに対応しています。これらのレジスタは割込みレベルと拡張インテリジェント I/O サービス (EI²OS) を設定します。
- リソース制御レジスタ領域 (アドレス: "000020H ~ 0000AFH", "0000C0H ~ 0000EFH")

これらのレジスタはリソースを制御し, かつデータを入出力します。I/O アドレスを指定する命令 (MOV A, io) は 003FE0H ~ 003FFFH レジスタ領域をサポートしていません。
- I/O ポート制御レジスタ領域 (アドレス: "000000H ~ 000001H")

これらのレジスタは I/O ポートを制御し, かつデータを入出力します。

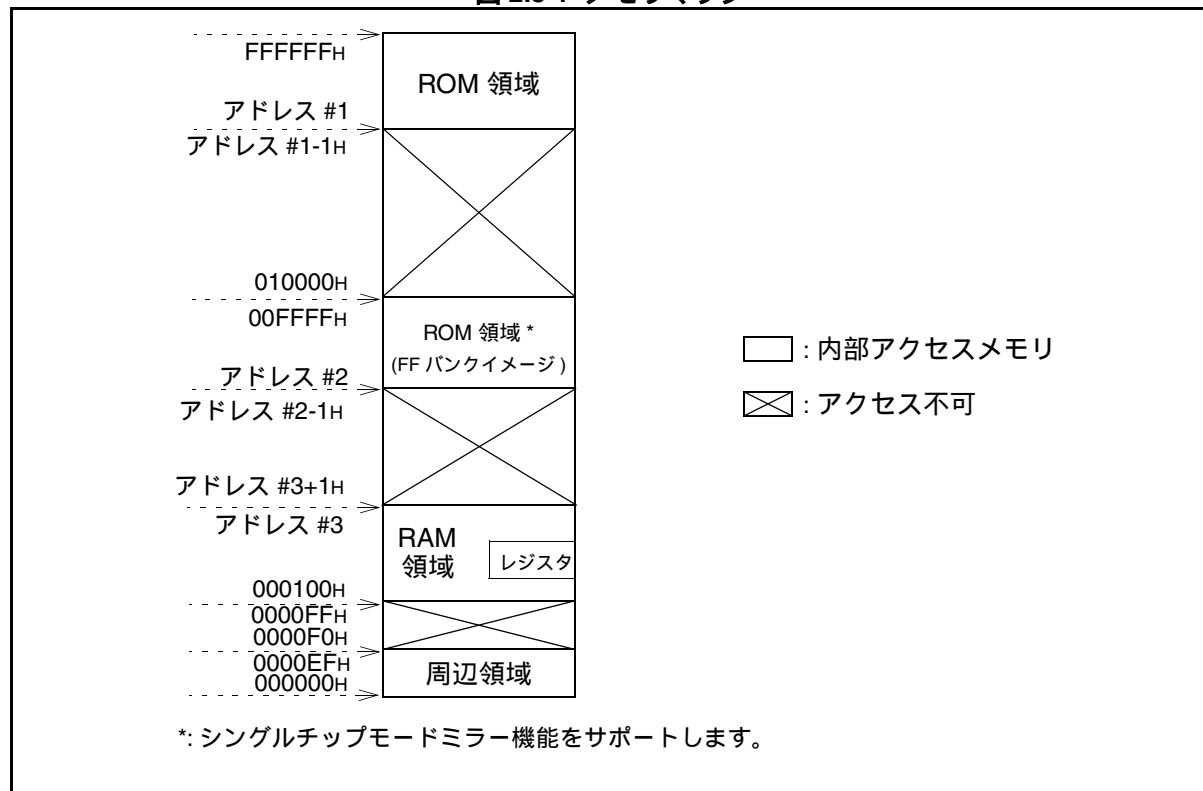
2.3 メモリマップ

MB90820B シリーズの各製品のメモリマップを示します。

■ メモリマップ

図 2.3-1 に MB90820B シリーズのメモリマップを示します。

図 2.3-1 メモリマップ



製品番号	アドレス #1	アドレス #2	アドレス #3
MB90822B	FF0000 _H	008000 _H	0010FF _H
MB90823B	FE0000 _H	008000 _H	0010FF _H
MB90F822B	FF0000 _H	008000 _H	0010FF _H
MB90F823B	FE0000 _H	008000 _H	0010FF _H
MB90V820B	(FE0000 _H)	008000 _H	0040FF _H
MB90F828B	FE0000 _H	008000 _H	0020FF _H

< 参考 >

シングルチップモード (ROM ミラー機能なし) を選択した場合, 「第 22 章 ROM ミラー機能 選択モジュール」をご参照ください。

< 注意事項 >

FF バンクの ROM データは、00 バンクの上位にイメージで見えるようになっており、C コンパイラ・スモールモデルの効率的な使用を実現します。下位 16 ビットは同じアドレスに割り当てられており、ポインタの宣言において "far" を指定せずに ROM 上のテーブルを参照できます。例えば、00C000_H へのアクセスを試みた場合、実際には FFC000_H の ROM の内容にアクセスすることになります。FF バンクの ROM 領域が 32K バイトを超えるので、全領域を 00 バンクのイメージに反映することはできません。したがって、あたかも FF8000_H ~ FFFFFFF_H の ROM データが 008000_H ~ 00FFFF_H のイメージであるかのように見えます。したがって、ROM データテーブルは FF8000_H ~ FFFFFFF_H の領域に格納してください。

2.4 アドレス指定

アドレス指定には、リニア方式とバンク方式があります。

リニア方式は、完備した 24 ビットアドレスを命令により直接指定します。

バンク方式は、上位 8 ビットアドレスをバンクレジスタにより指定し、下位 16 ビットアドレスを命令により直接指定します。

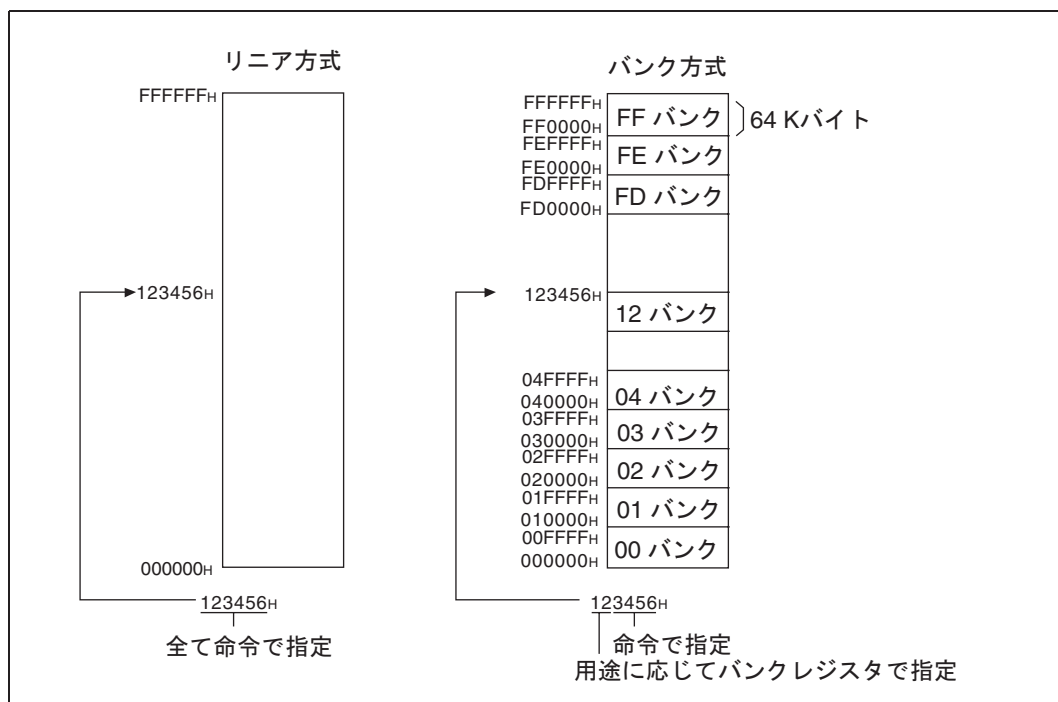
F²MC-16LX ファミリは、通常バンク方式を使用します。

■ リニア方式とバンク方式

リニア方式では、連続アドレス空間として 16M バイト空間がアクセスされます。バンク方式では、16M バイト空間は 256 の 64K バイトバンクとして分割され管理されます。

図 2.4-1 にリニア方式とバンク方式メモリ管理の概要を示します。

図 2.4-1 リニア方式およびバンク方式メモリ管理

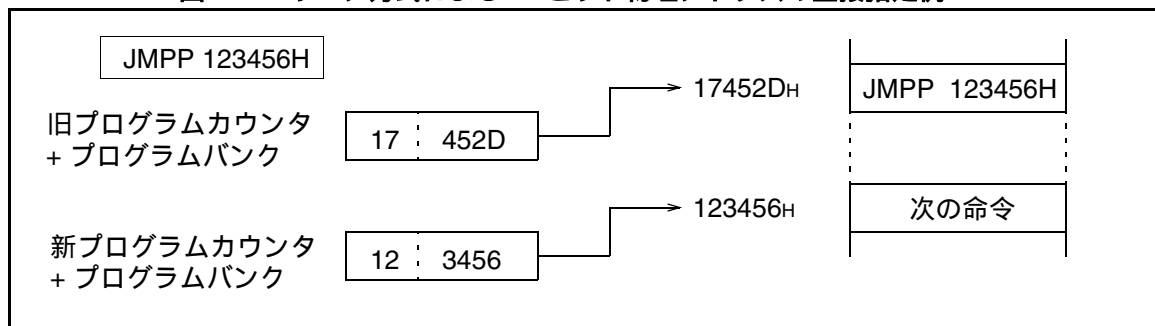


2.4.1 リニア方式によるアドレス指定

リニア方式によるアドレス指定には、さらにオペランドで 24 ビットのアドレスを直接指定する方法と、32 ビット汎用レジスタの下位 24 ビットをアドレスとして引用指定 (間接指定) する方法があります。

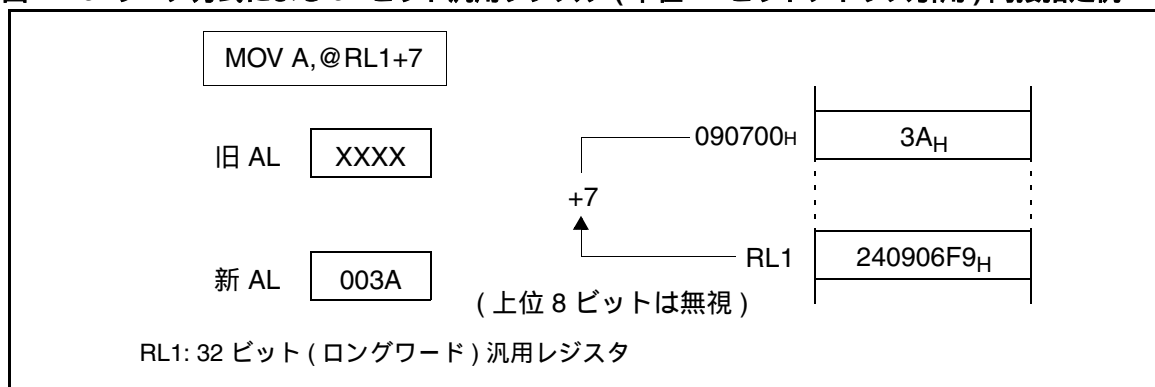
■ 24 ビットオペランドによるリニア方式アドレス指定

図 2.4-2 リニア方式による 24 ビット物理アドレスの直接指定例



■ 32 ビット汎用レジスタの下位 24 ビットアドレス引用指定 (間接指定)

図 2.4-3 リニア方式による 32 ビット汎用レジスタ (下位 24 ビットアドレス引用) 間接指定例



2.4.2 バンク方式によるアドレス指定

バンク方式は、16M バイトのメモリ空間を 64K バイトごとの 256 バンクに分割し、各々の空間に対応するバンクアドレスをアドレスの上位 8 ビットを決めるバンクレジスタによって指定します。下位 16 ビットアドレスは命令により直接指定します。バンクレジスタには、機能別に以下に示す 5 種類があります。

- プログラムバンクレジスタ (PCB)
- データバンクレジスタ (DTB)
- ユーザスタックバンクレジスタ (USB)
- システムスタックバンクレジスタ (SSB)
- アディショナルバンクレジスタ (ADB)

■ バンクレジスタとアクセス空間

各バンクレジスタのアクセス空間と主な機能について表 2.4-1 に示します。

表 2.4-1 各バンクレジスタのアクセス空間と主な機能

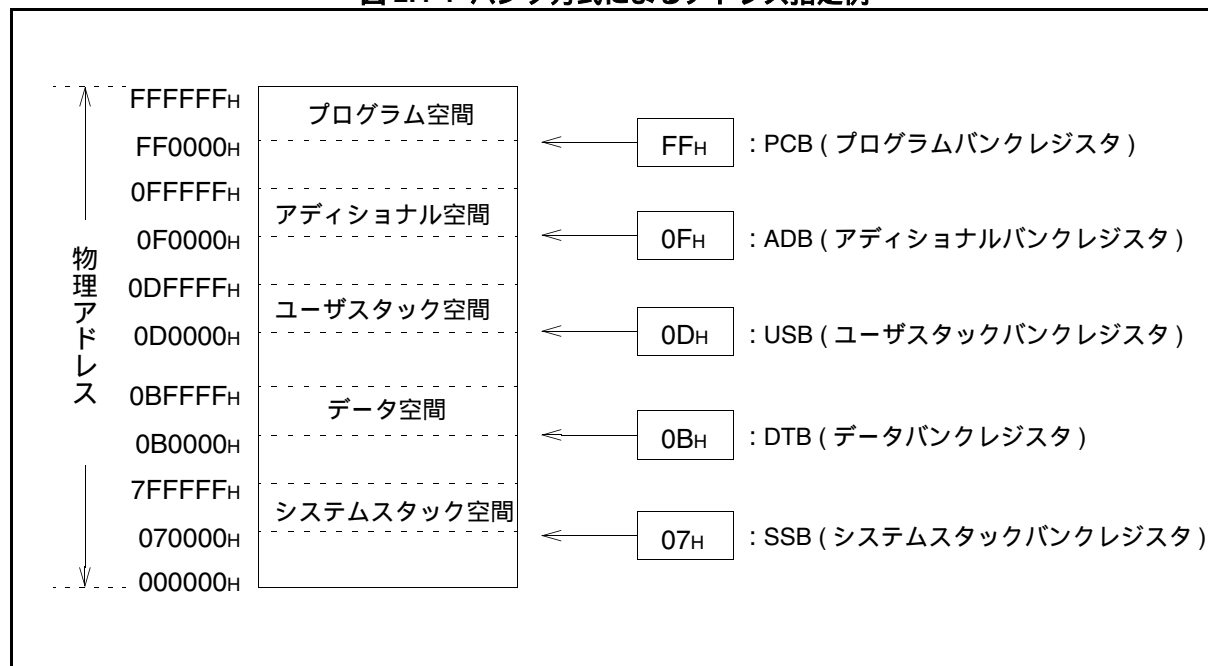
バンクレジスタ名	アクセス空間	主な機能	リセット後の初期値
プログラムバンクレジスタ (PCB)	プログラム (PC) 空間	命令コード、ベクタテーブル、即値データの格納用	FF _H
データバンクレジスタ (DTB)	データ (DT) 空間	読み書き可能なデータを格納。内外周辺の制御レジスタやデータレジスタへのアクセス用	00 _H
ユーザスタックバンクレジスタ (USB)	スタック (SP) 空間	PUSH/POP 命令や、割込みのレジスタ退避などのスタックアクセス用。コンディションレジスタ内のスタックフラグ (CCR : S) が "1" の場合は SSB レジスタを、"0" の場合は USB レジスタを使用 *	00 _H
システムスタックバンクレジスタ (SSB) *			00 _H
アディショナルバンクレジスタ (ADB)	アディショナル (AD) 空間	データ (DT) 空間でオーバフローしたデータの格納用	00 _H

*: 割込み時のスタックには、必ず SSB レジスタが使われます。

バンクレジスタの詳細は、「2.7.9 バンクレジスタ (PCB, DTB, USB, SSB, ADB)」をご参照ください。

メモリ空間の分割と各レジスタの関係を図 2.4-4 に示します。

図 2.4-4 バンク方式によるアドレス指定例



■ バンク方式によるアドレス指定とデフォルト空間

命令のコード効率向上のために、各命令には、アドレス指定方式ごとに表 2.4-2 に示すようなデフォルト空間が決められています。デフォルト以外の空間を使用したい場合は、各バンクに対応しているプリフィックスコードを、命令に先行して設定してください。これにより、プリフィックスコードに対応したバンク空間へアクセスできます。プリフィックスコードの詳細は、「2.9 プリフィックスコード」をご参照ください。

表 2.4-2 アドレス指定とデフォルト空間

デフォルト空間	アドレス指定
プログラム空間	PC 間接, プログラムアクセス, 分岐系
データ空間	@RW0, @RW1, @RW4, @RW5, @A, addr16, dir を用いたアドレス指定
スタック空間	PUSHW, POPW, @RW3, @RW7 を用いたアドレス指定
アディショナル空間	@RW2, @RW6 を用いたアドレス指定

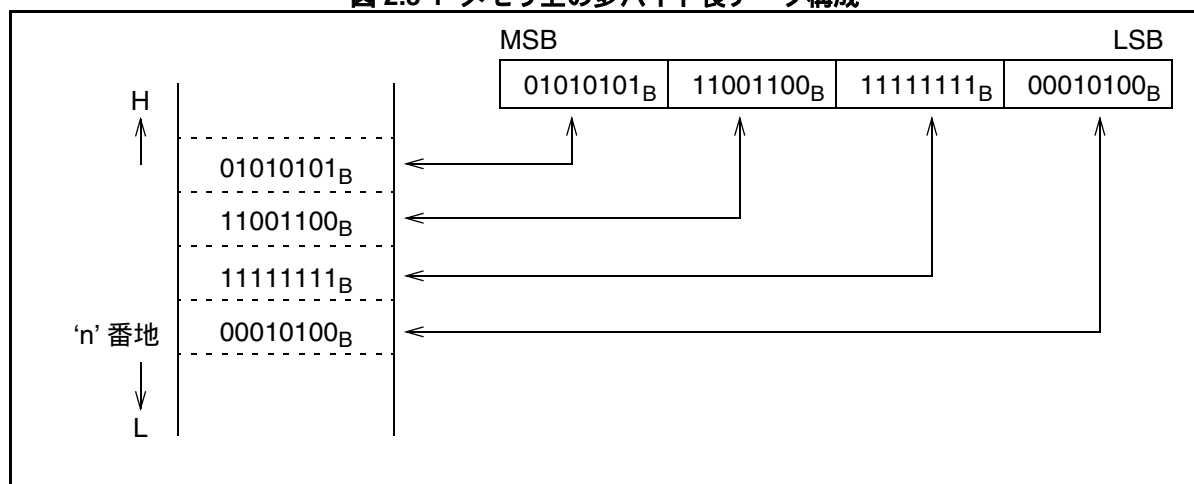
2.5 多バイト長データのメモリ配置

多バイト長データは、下位アドレスから順にメモリへ書き込まれます。多バイト長データが 32 ビットであれば、下位 16 ビットの次に上位 16 ビットの順に転送されます。ただし、下位データを書き込んだ直後に外部リセット信号が入力された場合、上位データが書き込まれない場合があります。

■ メモリ上の多バイト長データの格納状態

- メモリ上における多バイト長データの構成を図 2.5-1 に示します。
- データは下位 8 ビットが n 番地に、次に $n + 1$ 番地、 $n + 2$ 番地、 $n + 3$ 番地の順に配置されます。

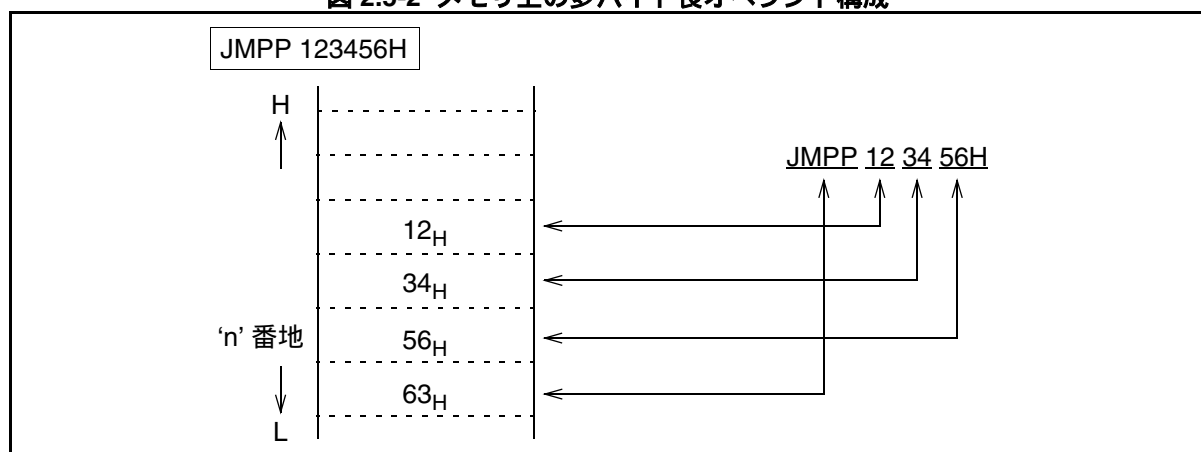
図 2.5-1 メモリ上の多バイト長データ構成



■ 多バイト長オペランドの格納状態

メモリ上における多バイト長オペランドの構成を図 2.5-2 に示します。

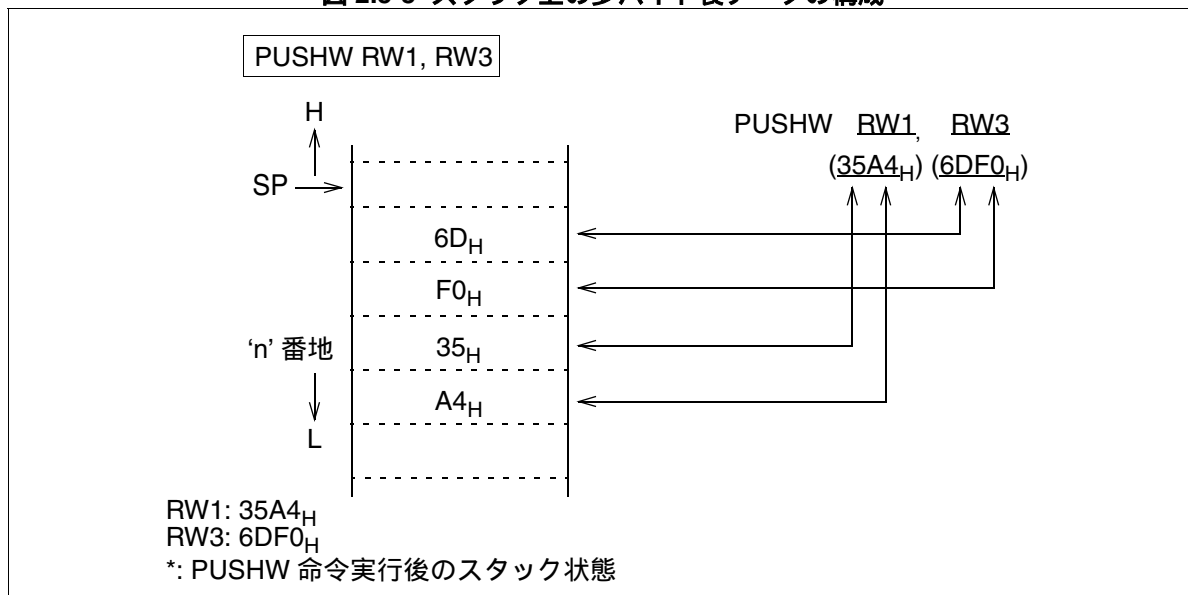
図 2.5-2 メモリ上の多バイト長オペランド構成



■ スタック上の多バイト長データの格納状態

スタック上の多バイト長データの構成を図 2.5-3 に示します。

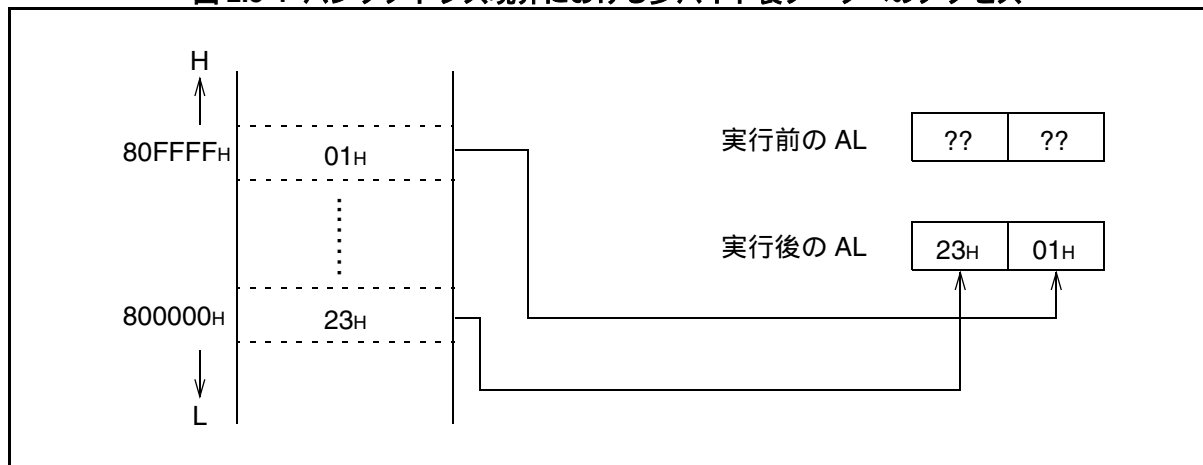
図 2.5-3 スタック上の多バイト長データの構成



■ 多バイト長のアクセス

- バイト長データのアクセスは、バンク内が基本です。多バイト長データへアクセスする命令を実行した場合は、"FFFF_H" 番地の次のアドレスは同じバンクの "0000_H" 番地になります。
- バンクアドレス境界における多バイト長データへのアクセスを図 2.5-4 に示します。

図 2.5-4 バンクアドレス境界における多バイト長データへのアクセス



2.6 レジスタ

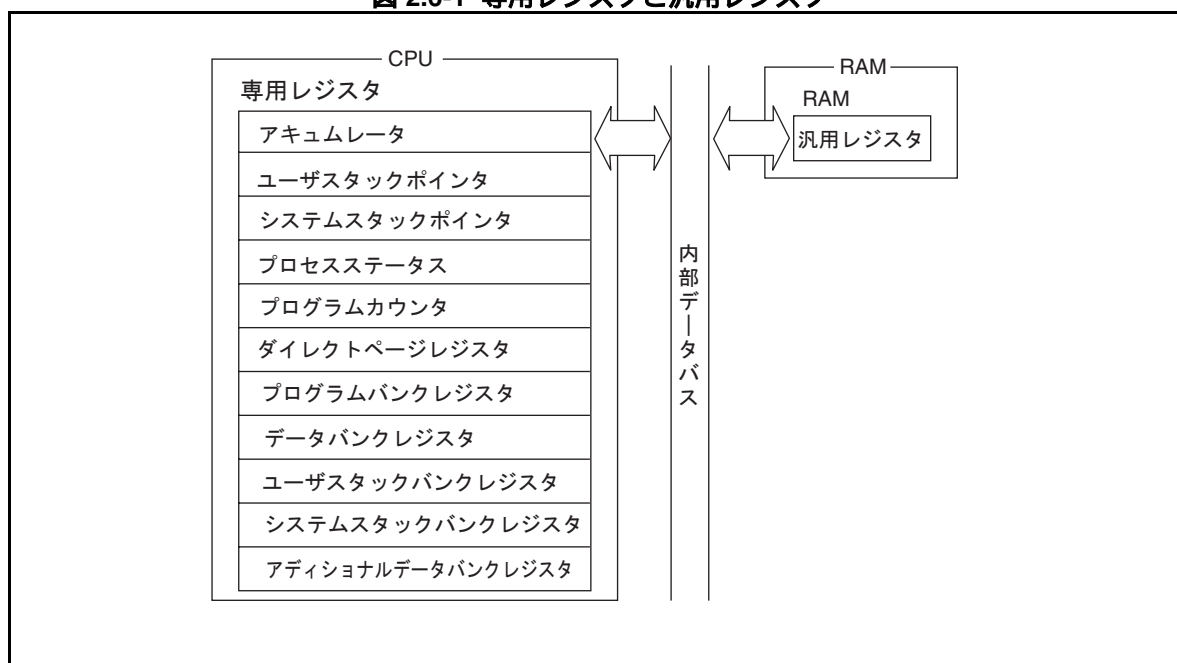
F²MC-16LX ファミリのレジスタには、CPU に内蔵されている専用レジスタと、内蔵 RAM 上に配置されている汎用レジスタがあります。

■ 専用レジスタと汎用レジスタ

- 専用レジスタは、CPU アーキテクチャ上で使用制限のある CPU 内にある専用ハードウェアです。
- 汎用レジスタは、CPU アドレス空間上に RAM と共存しています。専用レジスタと同じく、アドレス指定なしにアクセスできます。通常のメモリと同じように、ユーザ側でレジスタの使用方法を指定できます。

図 2.6-1 に専用レジスタおよび汎用レジスタの配置を示します。

図 2.6-1 専用レジスタと汎用レジスタ



2.7 専用レジスタ

専用レジスタは、以下に示す 11 種類のレジスタで構成されています。

- アキュムレータ (A)
- システムスタックポインタ (SSP)
- プログラムカウンタ (PC)
- プログラムバンクレジスタ (PCB)
- ユーザスタックバンクレジスタ (USB)
- アディショナルデータバンクレジスタ (ADB)
- ユーザスタックポインタ (USP)
- プロセッサステータス (PS)
- ダイレクトページレジスタ (DPR)
- データバンクレジスタ (DTB)
- システムスタックバンクレジスタ (SSB)

■ 専用レジスタの構成

専用レジスタの構成を図 2.7-1 に示します。専用レジスタ初期値を表 2.7-1 に示します。

図 2.7-1 割込み要求と割込みベクタ

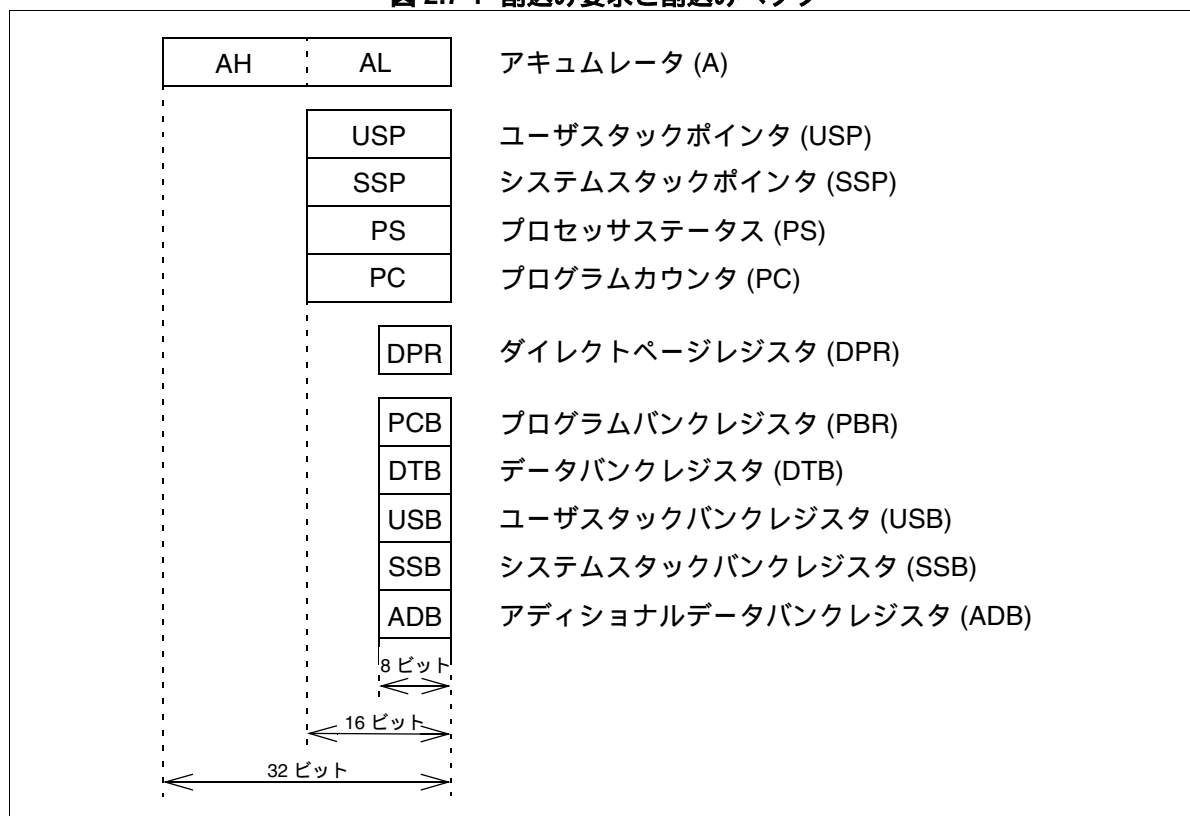


表 2.7-1 専用レジスタの初期値

専用レジスタ	初期値
アキュムレータ (A)	不定
ユーザスタックポインタ (USP)	不定
システムスタックポインタ (SSP)	不定
プロセッサステータス (PS)	<div style="text-align: center;"> bit 15 \longleftrightarrow 13 12 \longleftrightarrow 8 7 \longleftrightarrow 0 PS ILM RP CCR デフォルト値 000 00000 -01xxxxx </div>
プログラムカウンタ (PC)	リセットベクタ中の値 (FFFFDC _H , FFFFDD _H の値)
ダイレクトページレジスタ (DPR)	01 _H
プログラムバンクレジスタ (PCB)	リセットベクタ中の値 (FFFFDE _H の値)
データバンクレジスタ (DTB)	00 _H
ユーザスタックバンクレジスタ (USB)	00 _H
システムスタックバンクレジスタ (SSB)	00 _H
アディショナルデータバンクレジスタ (ADB)	00 _H

-: 未使用

x : 不定

< 注意事項 >

上記の初期値は本製品の初期値であり, ICE(エミュレータなど) 値とは異なります。

2.7.1 アキュムレータ (A)

アキュムレータ (A) は、2つの16ビット長演算用レジスタ (AH/AL) から構成されています。演算結果やデータの一時記憶に使用されるレジスタです。

アキュムレータ (A) は、32/16/8 ビットのレジスタとして使用できます。メモリとほかのレジスタ間、あるいは上位側の16ビット長演算レジスタ (AH) と下位側の16ビット長演算レジスタ (AL) 間で演算できます。また、ワード長以下 (16ビット長以下) のデータを下位側の16ビット長演算レジスタ (AL) へ転送した場合、それまでに存在した下位側の16ビット長演算レジスタ (AL) のデータが、上位側の16ビット長演算レジスタ (AH) へ自動的に転送されるデータ保持機能があります (一部の命令ではこの "データ保持機能" は無効となります)。

■ アキュムレータ (A)

● アキュムレータへのデータ転送

アキュムレータは、32ビット長 (ロングワード)、16ビット長 (ワード)、8ビット長 (バイト) のデータを処理します。例外として、4ビットデータ転送命令 (MOVN) も8ビットデータと同様に処理します。

- 32ビットデータの処理時は、上位演算用レジスタ (AH) と下位演算用レジスタ (AL) が連結して使用されます。
- 16ビットデータや8ビットデータの処理時は、下位演算用レジスタ (AL) が使用されます。
- 下位演算用レジスタ (AL) へバイト長以下 (8ビット長以下) のデータが転送された場合は、そのデータは16ビット長へ符号拡張またはゼロ拡張され、下位演算用レジスタ (AL) へ格納されます。また、下位演算用レジスタ (AL) へ格納されたデータは、16ビットデータまたは8ビットデータとして処理できます。

図 2.7-2 にアキュムレータへのデータ転送を示します。図 2.7-3 ~ 図 2.7-6 に具体的な転送例を示します。

図 2.7-2 アキュムレータへのデータ転送

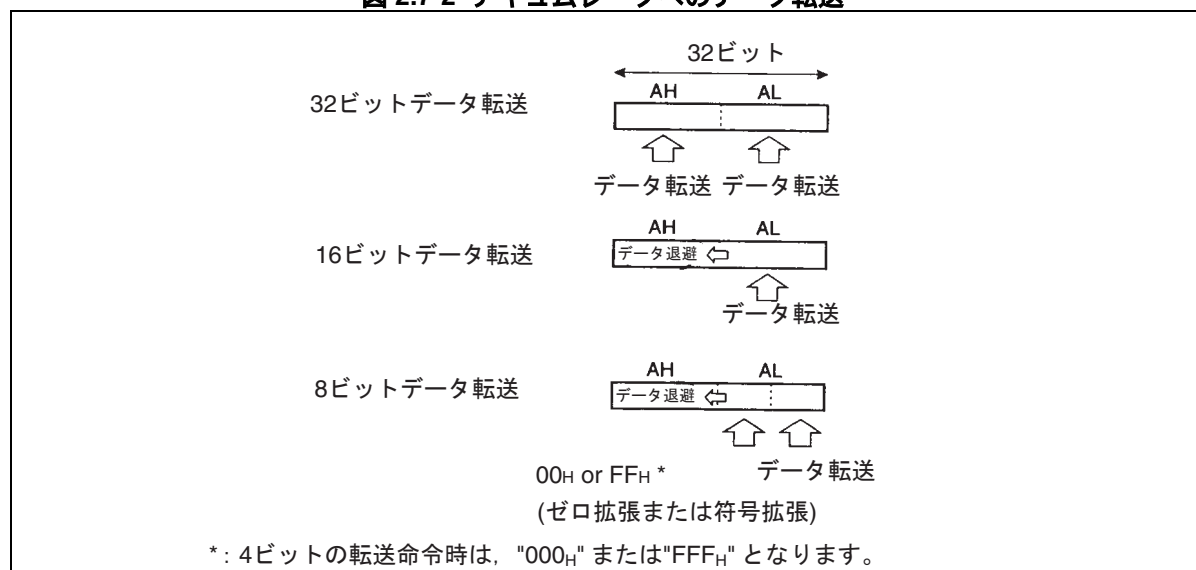


図 2.7-3 アキュムレータ (A) の AL-AH レジスタ間の転送例 (8 ビット即値, ゼロ拡張)

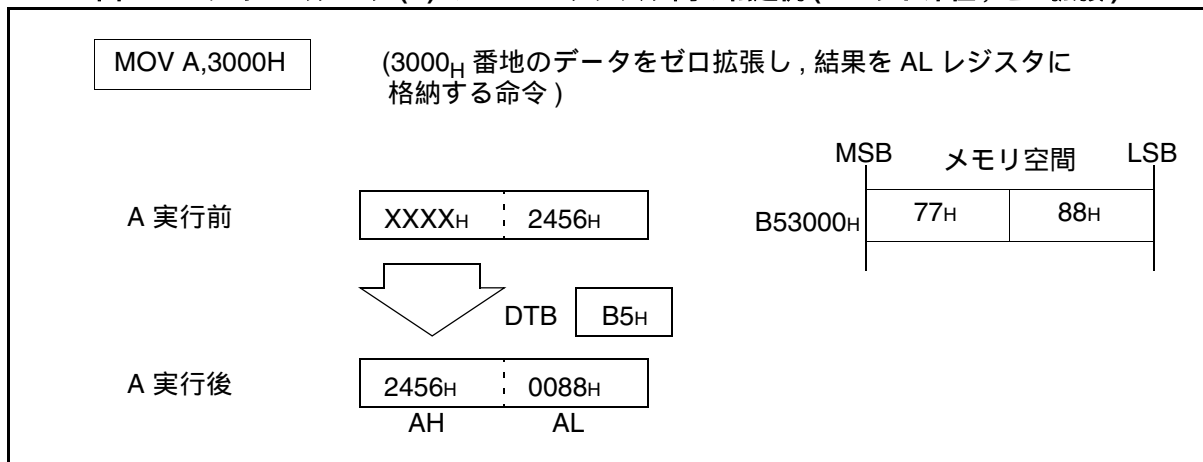


図 2.7-4 アキュムレータ (A) の AL-AH レジスタ間の転送例 (8 ビット即値, 符号拡張)

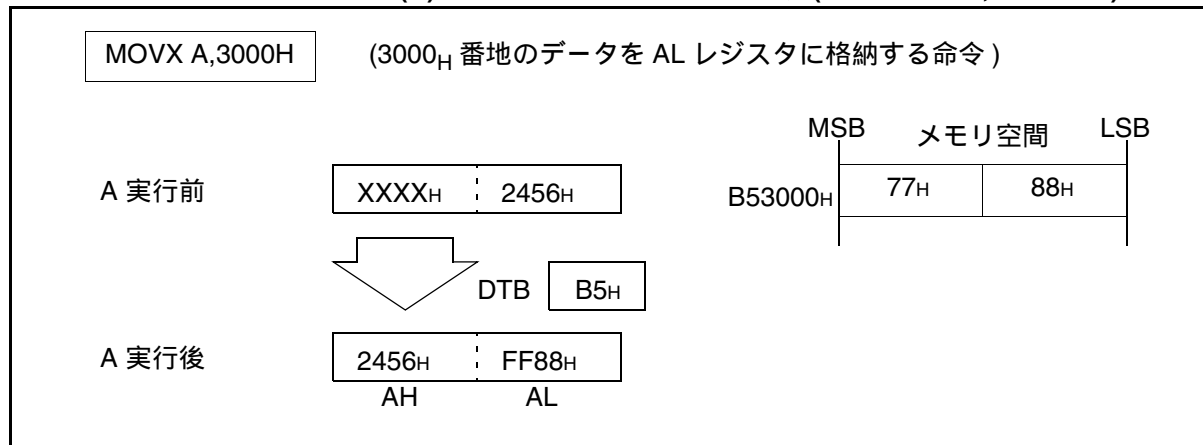


図 2.7-5 アキュムレータ (A) への 32 ビットデータ転送例 (レジスタ間接)

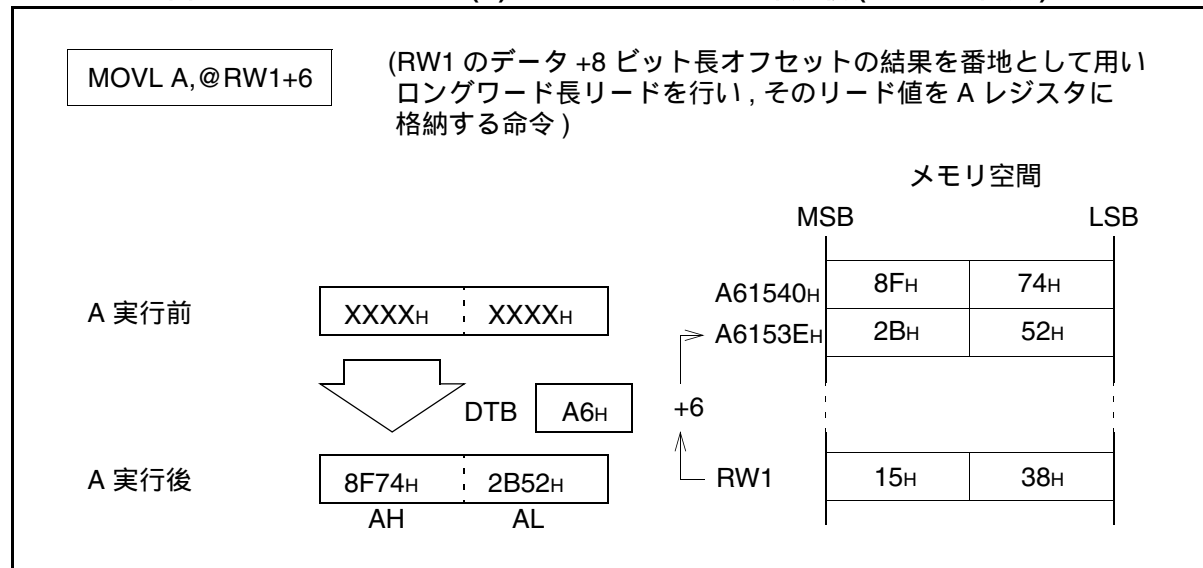
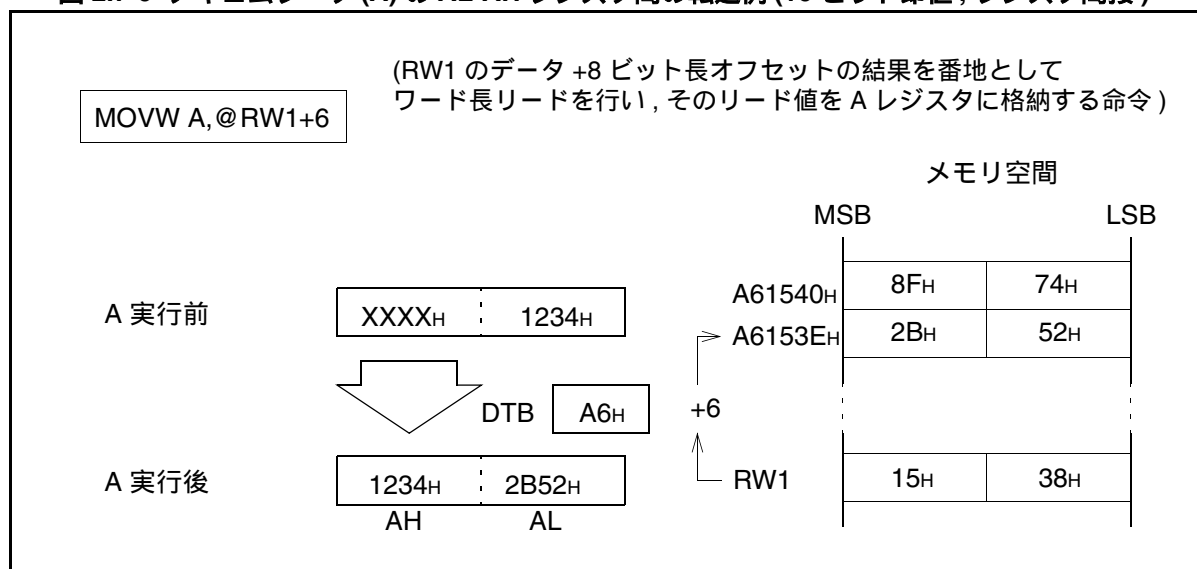


図 2.7-6 アキュムレータ (A) の AL-AH レジスタ間の転送例 (16 ビット即値, レジスタ間接)



● アキュムレータ (A) のバイト処理算術演算

バイト処理の算術演算命令を下位演算用レジスタ (AL) に対して実行した場合, 演算前の下位演算用レジスタ (AL) の上位 8 ビットは無視されます。演算結果の上位 8 ビットはすべて 0 になります。

● アキュムレータ (A) の初期値

リセット後の初期値は不定です。

MB90820B シリーズ

2.7.2 スタックポインタ (USP, SSP)

スタックポインタには、ユーザスタックポインタ (USP) とシステムスタックポインタ (SSP) があります。これらのレジスタは、PUSH 命令や POP 命令、およびサブルーチンを実行する場合に、データの退避先や復帰先のメモリアドレスを示す 24 ビットのレジスタです。スタックアドレス (24 ビット) の上位 8 ビットは、ユーザスタックバンクレジスタ (USB) もしくはシステムスタックバンクレジスタ (SSB) で指定されます。

コンディションコードレジスタ (CCR) の S フラグが "0" の場合は、ユーザスタックポインタ (USP) およびユーザスタックバンクレジスタ (USB) が有効になります。S フラグが "1" の場合は、システムスタックポインタ (SSP) およびシステムスタックバンクレジスタ (SSB) が有効になります。

■ スタックの設定

F²MC-16LX ファミリでは、システムスタックとユーザスタックの 2 種類のスタックを使用できます。スタックアドレスは、プロセッサステータスレジスタ (PS : CCR) 内の S フラグによって表 2.7-2 のように決定されます。

表 2.7-2 スタックアドレスの指定

S フラグ	スタックアドレス	
	上位 8 ビット	下位 16 ビット
0	ユーザスタックバンクレジスタ (USB)	ユーザスタックポインタ (USP)
1	システムスタックバンクレジスタ (SSB)	システムスタックポインタ (SSP)

初期値

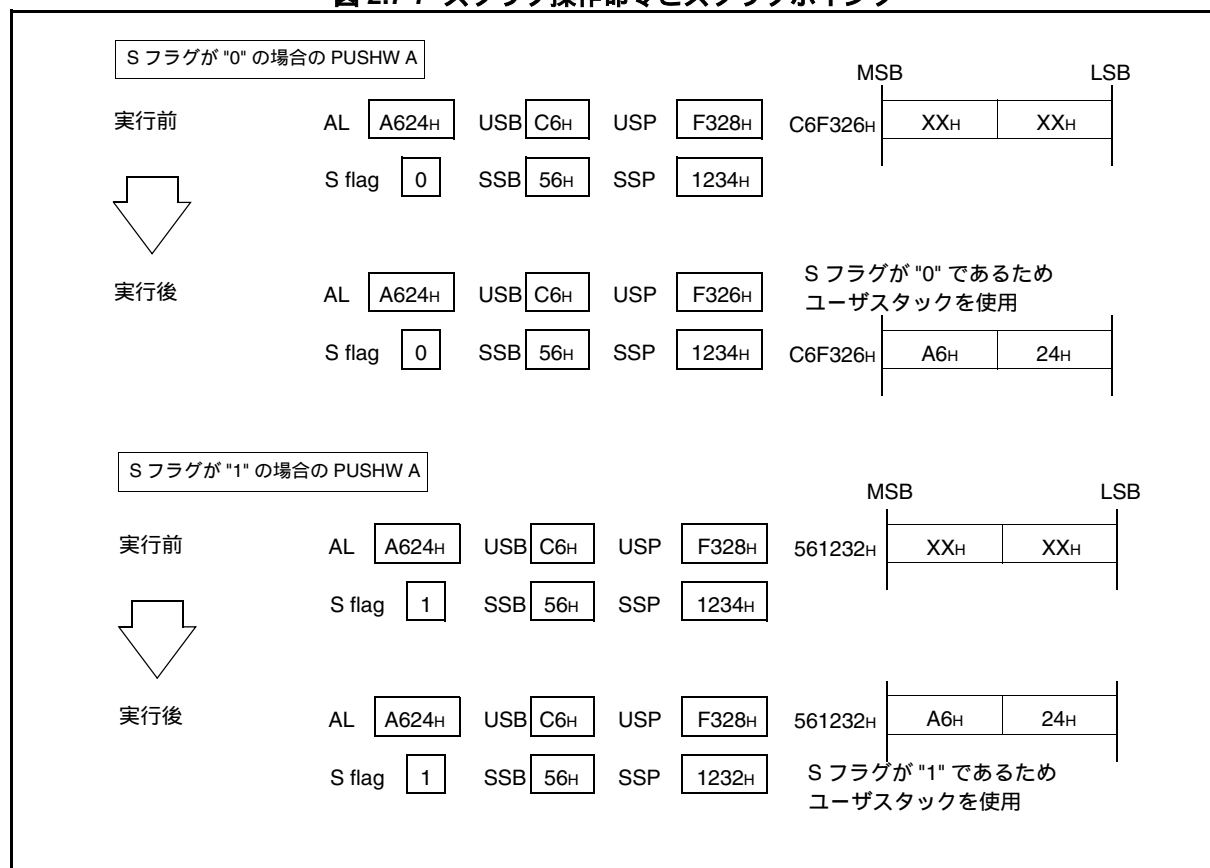
リセットによって、S フラグは "1" に初期化されます。したがって初期設定ではシステムスタックが使用されます。ただし、割込みルーチンのスタック操作時はシステムスタックが使用され、割込みルーチン以外のスタック操作時にはユーザスタックが使用されます。スタック空間を分割しないのであれば、システムスタックを使用してください。

< 注意事項 >

割込みが受け付けられた場合、S フラグに "1" がセットされ、システムスタックが使用されます。

スタック操作命令とスタックポインタを図 2.7-7 に示します。

図 2.7-7 スタック操作命令とスタックポインタ



< 参考 >

- スタックポインタにスタックアドレスを設定する場合は、偶数アドレスを設定してください。奇数アドレスを設定した場合は、ワードアクセスが2回に分割され効率が低下します。
- USP レジスタおよび SSP レジスタの初期値は不定です。

■ システムスタックポインタ (SSP)

システムスタックポインタ (SSP) を設定する場合は、プロセッサステータス (PS) のコンディションコードレジスタ (CCR) 内の S フラグに "1" を設定してください。スタックを操作する際に使用されるアドレスの上位8ビットは、システムスタックバンクレジスタ (SSB) により示されます。

■ ユーザスタックポインタ (USP)

ユーザスタックポインタ (USP) を設定する場合は、プロセッサステータス (PS) のコンディションコードレジスタ (CCR) 内の S フラグに "0" を設定してください。スタックを操作する際に使用されるアドレスの上位8ビットは、ユーザスタックバンクレジスタ (USB) により示されます。

2.7.3 プロセッサステータス

プロセッサステータス (PS) は、CPU を制御するビットと CPU の状態を示すビットから構成されています。プロセッサステータス (PS) は、以下に示す 3 つのレジスタで構成されています。

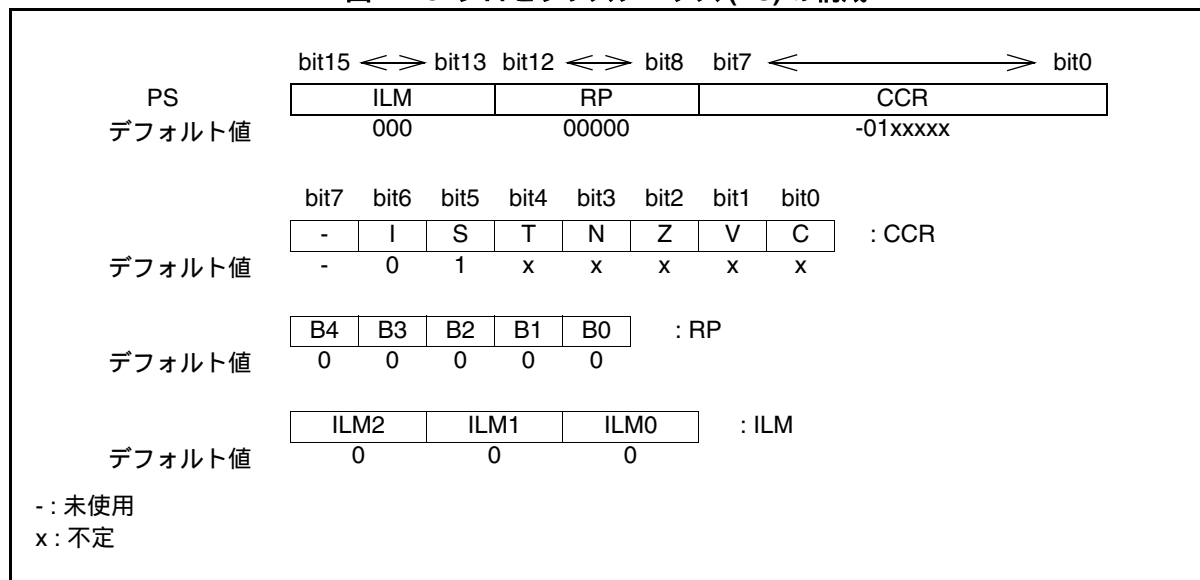
- 割り込みレベルマスクレジスタ (ILM)
- レジスタバンクポインタ (RP)
- コンディションコードレジスタ (CCR)

■ プロセッサステータスの構成

プロセッサステータス (PS) は、CPU を制御するビットと CPU の状態を示すビットから構成されています。

図 2.7-8 にプロセッサステータス (PS) の構成を示します。

図 2.7-8 プロセッサステータス (PS) の構成



● 割り込みレベルマスクレジスタ (ILM)

CPU が現在受付けている割り込みのレベルを示します。またリソースごとの割り込み要求に対応して設定されている割り込み制御レジスタの割り込みレベル設定ビット (ICR : IL0 ~ IL2) 値と比較します。

● レジスタバンクポインタ (RP)

- RAM 領域の中で汎用レジスタとして使用するメモリブロック (レジスタバンク) の先頭アドレスを設定するポインタです。
- 汎用レジスタは 32 バンクで構成されています。レジスタバンクポインタ (RP) に "0 ~ 31" の値を設定してバンクを指定してください。

● コンディションコードレジスタ (CCR)

命令実行結果や割り込み出力によって "1" が設定または "0" にクリアされる各種フラグから構成されています。

2.7.4 コンディションコードレジスタ (PS:CCR)

コンディションコードレジスタ (CCR) は、以下に示す 8 ビットで構成されています。

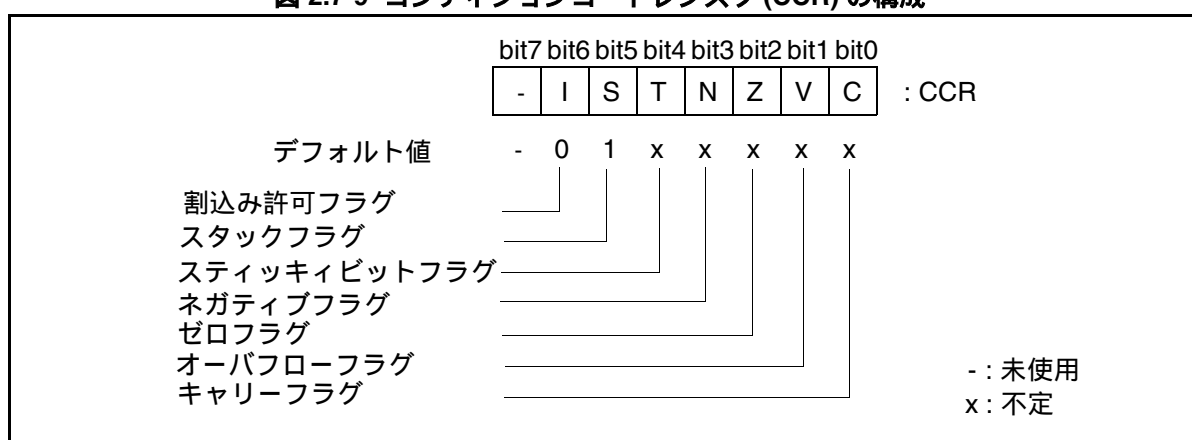
- 演算結果や転送データを示すビット
- 割込み要求の受け付けを制御するビット

■ コンディションコードレジスタ (CCR) の構成

命令実行時のコンディションコードレジスタ (CCR) の状態については、「F²MC-16LX ファミリ プログラミングマニュアル」をご参照ください。

コンディションコードレジスタ (CCR) の構成を図 2.7-9 に示します。

図 2.7-9 コンディションコードレジスタ (CCR) の構成



● 割込み許可フラグ (I)

割込み許可フラグ (I) に "1" が設定された場合はソフトウェア割込み以外の割込み要求を許可し、割込み許可フラグ (I) が "0" にクリアされた場合はソフトウェア割込み以外の割込み要求を禁止します。リセットにより "0" にクリアされます。

● スタックフラグ (S)

スタック操作に用いられるポインタを示します。スタックフラグ (S) が "0" にクリアされた場合はユーザスタックポインタ (USP) が有効になり、スタックフラグ (S) に "1" が設定された場合はシステムスタックポインタ (SSP) が有効になります。割込みが受けられた場合、またはリセットが発生した場合は、"1" が設定されます。

● スティッキビットフラグ (T)

論理右シフト命令または算術右シフト命令を実行した場合、キャリーからシフトアウトされたデータ内に "1" があればスティッキビットフラグ (T) に "1" が設定され、"1" がなければスティッキビットフラグ (T) は "0" にクリアされます。また、シフト量がゼロの場合でも "0" にクリアされます。

- ネガティブフラグ (N)

演算の結果最上位ビット (MSB) が "1" の場合は、ネガティブフラグ (N) に "1" が設定されます。演算の結果最上位ビット (MSB) が "0" の場合は、ネガティブフラグ (N) が "0" にクリアされます。

- ゼロフラグ (Z)

演算結果の値がすべて "0" の場合は、ゼロフラグ (Z) に "1" が設定されます。それ以外の場合は、ゼロフラグ (Z) は "0" にクリアされます。

- オーバフローフラグ (V)

演算の結果、符号付き数値としてオーバフローが発生した場合は、オーバフローフラグ (V) に "1" が設定されます。オーバフローが発生しなかった場合は、オーバフローフラグ (V) は "0" にクリアされます。

- キャリーフラグ (C)

演算の結果、最上位ビットからの桁上りまたは最下位ビットからの桁下りが発生した場合は、キャリーフラグ (C) に "1" が設定されます。発生しなかった場合は、キャリーフラグ (C) は "0" にクリアされます。

2.7.5 レジスタバンクポインタ (PS:RP)

レジスタバンクポインタ (RP) は、現在使用している汎用レジスタバンクの先頭アドレスを示すレジスタです。レジスタバンクポインタは、汎用レジスタアドレス指定を使用する場合、リアルアドレス変換を行うために使用されます。

■ レジスタバンクポインタ (RP)

図 2.7-10 にレジスタバンクポインタ (RP) レジスタの構成を示します。

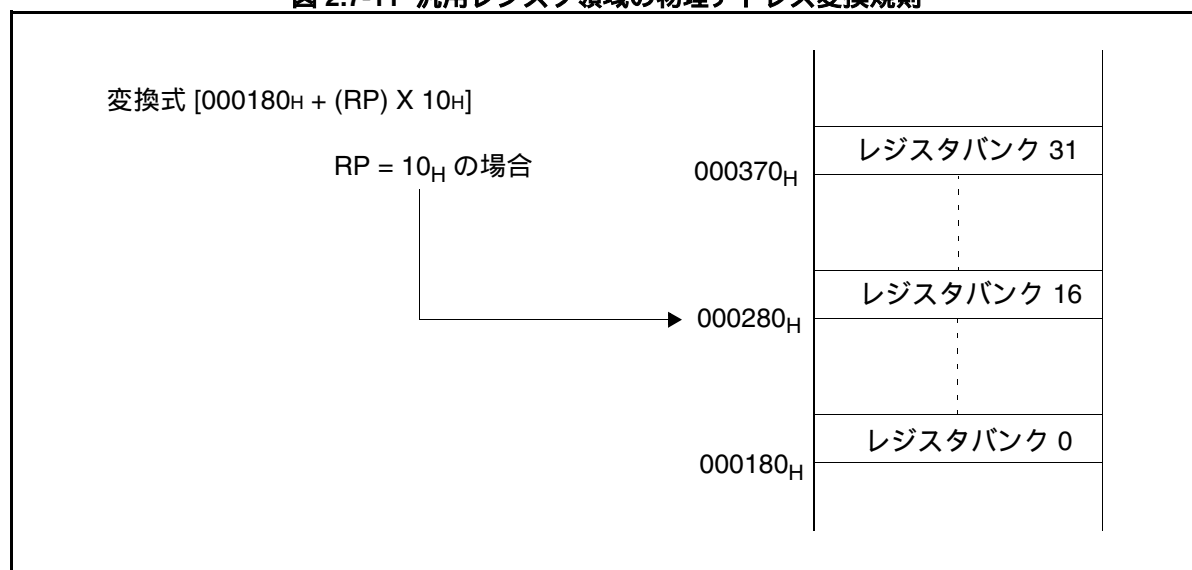
図 2.7-10 レジスタバンクポインタ (RP) の構成



■ 汎用レジスタ領域とレジスタバンクポインタ (RP)

レジスタバンクポインタ (RP) は、F²MC-16LX ファミリの汎用レジスタと、汎用レジスタのある内部 RAM のアドレスとの関係を示すポインタです。レジスタバンクポインタ (RP) のデータとリアルアドレスの関係に使用される変換規則を図 2.7-11 に示します。

図 2.7-11 汎用レジスタ領域の物理アドレス変換規則



- レジスタバンクポインタは "00_H" ~ "1F_H" までの値を取るので、レジスタバンクの最初のアドレスは "000180_H" ~ "000370_H" までの範囲で設定可能です。
- アセンブラ命令では、レジスタバンクポインタ (RP) へ転送する 8 ビットの即値転送命令を使用できます。ただし、下位 5 ビットのデータが有効となります。
- リセット後のレジスタバンクポインタ (RP) の初期値は、"00_H" となります。

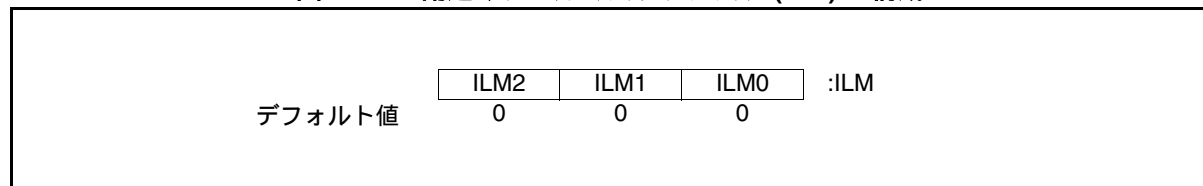
2.7.6 割込みレベルマスクレジスタ (PS:ILM)

割込みレベルマスクレジスタ (ILM) は、現在 CPU に受け付けられている割込みレベルを示す 3 ビットのレジスタです。

■ 割込みレベルマスクレジスタ (ILM)

割込みレベルマスクレジスタ (ILM) の構成を図 2.7-12 に示します。割込みの詳細は、「第 7 章 割込み」をご参照ください。

図 2.7-12 割込みレベルマスクレジスタ (ILM) の構成



割込みレベルマスクレジスタ (ILM) は、現在 CPU に受け付けられている割込みレベルを示します。このレベルは、リソース機能からの割込み要求に従って設定された割込み制御レジスタ (ICR00 ~ ICR15) の IL0 ~ IL2 ビットの値と比較されます。割込み許可フラグが許可されている場合 (CCR:I = 1)、割込み要求の値 (割込みレベル) がこれらのビットで示された値より小さいときのみ CPU は処理を行います。

- 割込みが受け付けられた場合、割込みレベル値は割込みレベルマスクレジスタ (ILM) に設定されます。そのため、同等または低いレベルの割込みは受け付けられません。
- 割込みレベルは、リセットによって割込みレベルマスクレジスタ (ILM) がすべて "0" に初期化されるため、割込み禁止ステータスである最強レベルに設定されます。
- アセンブラ命令では、割込みレベルマスクレジスタ (ILM) へ転送する 8 ビットの即値転送命令を使用できます。ただし、下位 3 ビットが有効となります。

表 2.7-3 割込みレベルマスクレジスタ (ILM) と割込みレベルの強弱

ILM2	ILM1	ILM0	割込みレベル	割込みレベルの強弱
0	0	0	0	<div style="display: flex; align-items: center; justify-content: center;"> <div style="text-align: center;"> ↑ ↓ </div> <div style="margin: 0 10px;"> <div style="text-align: center;">強 (割込み禁止)</div> <div style="text-align: center;">弱</div> </div> </div>
0	0	1	1	
0	1	0	2	
0	1	1	3	
1	0	0	4	
1	0	1	5	
1	1	0	6	
1	1	1	7	

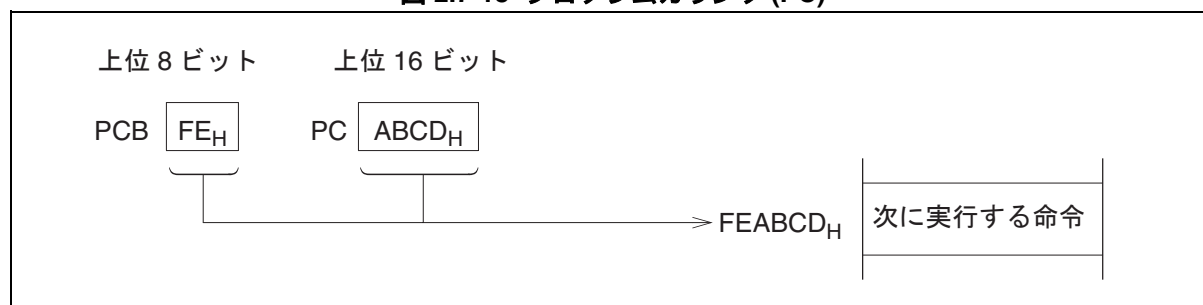
2.7.7 プログラムカウンタ (PC)

プログラムカウンタ (PC) は 16 ビットカウンタで、CPU が実行する次の命令コードのメモリアドレスの下位 16 ビットを示します。

■ プログラムカウンタ (PC)

- CPU が実行する次の命令コードが格納されているアドレスの上位 8 ビットをプログラムバンクレジスタ (PCB) で設定し、下位 16 ビットをプログラムカウンタ (PC) で設定します。使用以前は、図 2.7-13 に示すように、実際のアドレスは組み合わせられ 24 ビットとなります。
- プログラムカウンタ (PC) は条件分岐命令、サブルーチンコール命令、割込みやリセットで内容が更新されます。
- オペランドを読み出す場合のバスポインタとしても使用できます。

図 2.7-13 プログラムカウンタ (PC)



< 注意事項 >

プログラムカウンタ (PC) やプログラムバンクレジスタ (PCB) をプログラム (MOV PC および #FF などの命令) で直接書き換えることはできません。

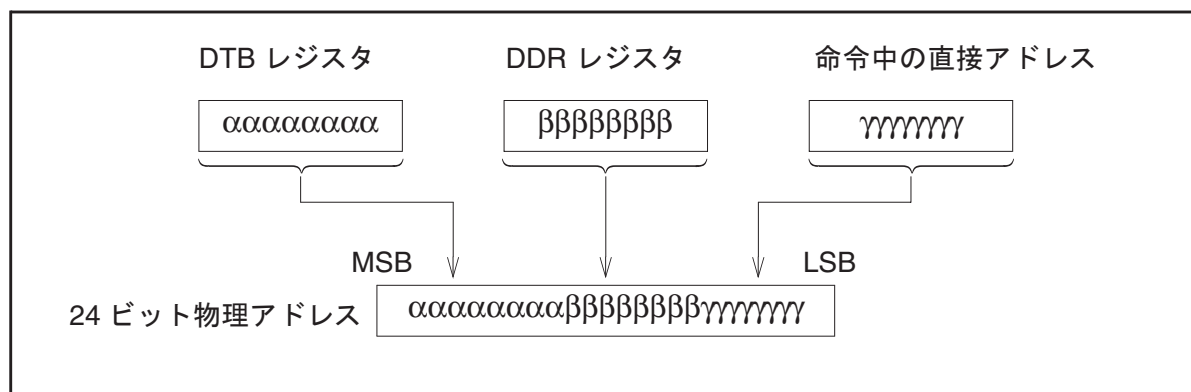
2.7.8 ダイレクトページレジスタ (DPR)

ダイレクトページレジスタ (DPR) は、短縮直接アドレス指定方式の命令実行時に、オペランドアドレスの bit8 ~ bit15(addr8 ~ addr15) を指定する 8 ビットレジスタです。

■ ダイレクトページレジスタ (DPR)

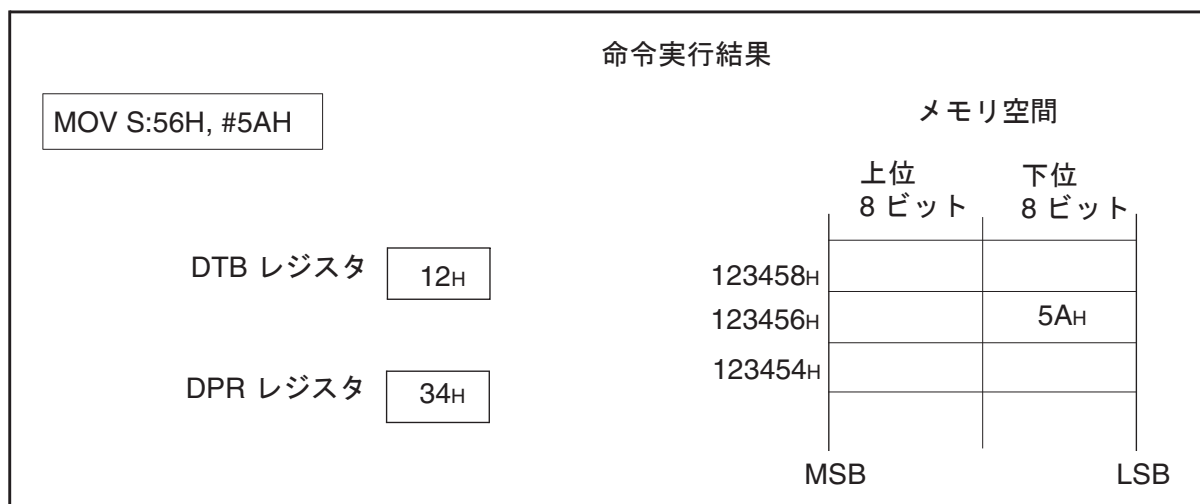
図 2.7-14 に示すように、ダイレクトページレジスタ (DPR) は、短縮直接アドレス指定方式の命令実行時に、オペランドアドレスの bit8 ~ bit15(addr8 ~ addr15) を指定します。ダイレクトページレジスタ (DPR) は 8 ビット長で、リセットにより "01_H" に初期化されます。命令を用いて読み出しおよび書き込みが可能です。

図 2.7-14 ダイレクトページレジスタ (DPR) による物理アドレスの生成



ダイレクトページレジスタ (DPR) の設定とデータアクセス例を図 2.7-15 に示します。

図 2.7-15 ダイレクトページレジスタ (DPR) の設定とデータアクセス例



2.7.9 バンクレジスタ (PCB, DTB, USB, SSB, ADB)

バンクレジスタは、バンク方式アドレス指定の最上位 8 ビットアドレスを指定するレジスタであり、以下に示す 5 種類のレジスタで構成されます。

- プログラムバンクレジスタ (PCB)
- データバンクレジスタ (DTB)
- ユーザスタックバンクレジスタ (USB)
- システムスタックバンクレジスタ (SSB)
- アディショナルバンクレジスタ (ADB)

各バンクレジスタは、プログラム空間、データ空間、ユーザスタック空間、システムスタック空間およびアディショナル空間に配置されるメモリバンクを示します。

■ バンクレジスタ (PCB, DTB, USB, SSB, ADB)

● プログラムバンクレジスタ (PCB)

- プログラム (PC) 空間を指定するバンクレジスタです。
- ソフトウェア割込み命令が実行された場合、16M バイト空間内どこへも分岐する JMP, CALLP, RETP および RETI 命令が実行された場合、およびハードウェア割込みもしくは例外が発生した場合、プログラムバンクレジスタ (PCB) は書き換えられます。

● データバンクレジスタ (DTB)

データ (DT) 空間を指定するバンクレジスタです。

● ユーザスタックバンクレジスタ (USB)/ システムスタックバンクレジスタ (SSB)

スタック (SP) 空間を指定するバンクレジスタです。ユーザスタックバンクレジスタ (USB) またはシステムスタックバンクレジスタ (SSB) が使用されるかは、プロセッサステータス (PS: CCR) の S フラグ値によります。詳細は、「2.7.2 スタックポインタ (USP, SSP)」をご参照ください。

● アディショナルバンクレジスタ (ADB)

アディショナル (AD) 空間を指定するバンクレジスタです。

● 各バンクの設定とデータアクセス

バンクレジスタは、バイト長です。リセットによりプログラムバンクレジスタ (PCB) は "FF_H" に初期化され、それ以外のレジスタは "00_H" に初期化されます。プログラムバンクレジスタ (PCB) は、読出しは可能ですが書込みはできません。プログラムバンクレジスタ (PCB) 以外のバンクレジスタは、読み書き可能です。

< 注意事項 >

MB90820B シリーズは、デバイスに内蔵されるメモリ空間までをサポートします。

各レジスタ動作の詳細は、「2.4.2 バンク方式によるアドレス指定」をご参照ください。

2.8 汎用レジスタ

汎用レジスタは、RAM 上の "000180H ~ 00037FH" に、16 ビット × 8 本を 1 つのレジスタバンクとして割当てたメモリブロックです。汎用の 8 ビットレジスタ (バイトレジスタ R0 ~ R7)、16 ビットレジスタ (ワードレジスタ RW0 ~ RW7) もしくは 32 ビットレジスタ (ロングワードレジスタ RL0 ~ RL3) として使用できます。

汎用レジスタへは、短い命令で高速にアクセスできます。またレジスタバンクへブロック化されているため、データの保護や機能単位の分割が容易です。また、ロングワードレジスタとして使用した場合は、全空間へ直接アクセスするリニアポインタとしても使用できます。

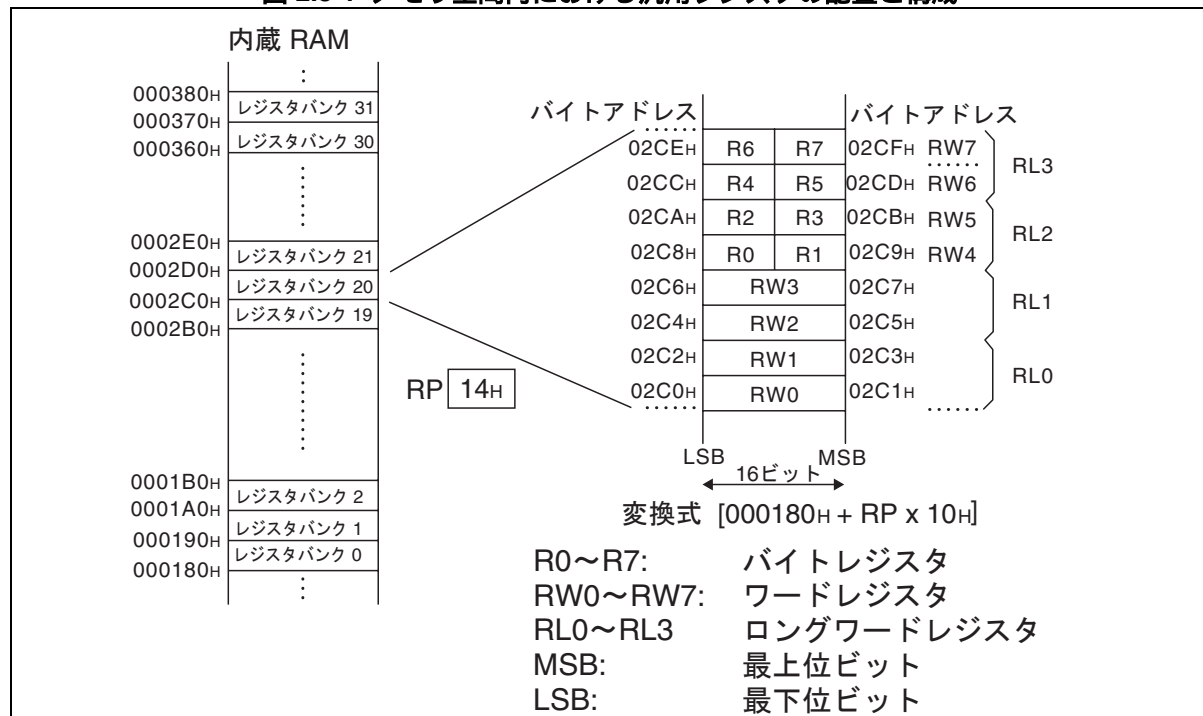
■ 汎用レジスタの構成

- すべての汎用レジスタは、RAM 上の "000180H ~ 00037FH" に 32 バンク存在しています。レジスタバンクポインタ (RP) で汎用レジスタとして使用されるバンクを指定し、現在使用中のバンクを示します。
- レジスタバンクポインタ (RP) で設定するバンクの先頭アドレスは、下式のようになります。

汎用レジスタの先頭アドレス = 000180H + レジスタバンクポインタ (RP) × 10H

メモリ空間内における汎用レジスタの配置と構成を図 2.8-1 に示します。

図 2.8-1 メモリ空間内における汎用レジスタの配置と構成



< 注意事項 >

レジスタバンクポインタ (RP) は、リセット後 "00H" に初期化されます。

■ レジスタバンク

レジスタバンクは、種々の演算やポインタの汎用レジスタ（バイトレジスタ R0 ~ R7, ワードレジスタ RW0 ~ RW7, ロングワードレジスタ RL0 ~ RL3）として使用できます。ロングワードレジスタは、全体のメモリ空間を直接アクセスできるリニアにポインタとして使用することも可能です。

レジスタバンク内の汎用レジスタは、RAM と同様にリセットでは初期化されず、リセット前のデータを保持します。ただし、パワーオンリセット時には不定となります。汎用レジスタの代表的な機能を表 2.8-1 に示します。

表 2.8-1 汎用レジスタの代表的な機能

レジスタ名	機能
R0 ~ R7	各種命令のオペランドとして使用 (注意事項) R0 はバレルシフトのカウンタおよびノーマライズ(正規化)命令のカウンタとしても使用
RW0 ~ RW7	ポインタとして使用 各種命令のオペランドとして使用 (注意事項) RW0 はストリング命令のカウンタとしても使用
RL0 ~ RL3	ロングポインタとして使用 各種命令のオペランドとして使用

2.9 プリフィックスコード

命令の前にプリフィックスコードを設定した場合は、プリフィックスコード直後の命令動作を一部変更できます。プリフィックスコードには、以下に示す 3 種類があります。

- バンクセレクトプリフィックス (PCB, DTB, ADB, SPB)
 - コモンレジスタバンクプリフィックス (CMR)
 - フラグ変化抑止プリフィックス (NCC)
-

■ プリフィックスコード

- バンクセレクトプリフィックス (PCB, DTB, ADB, SPB)

バンクセレクトプリフィックスを命令の前に設定した場合は、アドレス指定方式とは無関係に命令がアクセスするメモリ空間を設定できます。

- コモンレジスタバンクプリフィックス (CMR)

レジスタバンクにアクセスする命令の前にコモンレジスタバンクプリフィックスを設定した場合は、レジスタバンクポインタ (RP) の値とは関係なく、命令によってレジスタアクセスを "000180H ~ 00018Fh" に存在するコモンバンク (RP = 00H の場合に設定されるレジスタバンク) へ変更できます。

- フラグ変化抑止プリフィックス (NCC)

命令の実行に伴うフラグ変化を抑止する命令の前に、フラグ変化抑止プリフィックスコードを設定します。

2.9.1 バンクセレクトプリフィックス (PCB, DTB, ADB, SPB)

バンクセレクトプリフィックスを命令の前に設定した場合は、アドレス指定方式とは無関係に、アクセスするメモリ空間を設定できます。

■ バンクセレクトプリフィックス (PCB, DTB, ADB, SPB)

データアクセスに使用されるメモリ空間は、アドレス指定方式ごとに定められています。バンクセレクトプリフィックスを命令の前に設定した場合は、アドレス指定方式とは無関係に、アクセスするメモリ空間を設定できます。バンクセレクトプリフィックスコードと設定されるメモリ空間を表 2.9-1 に示します。

表 2.9-1 バンクセレクトプリフィックスコードと設定されるメモリ空間

バンクセレクトプリフィックス	設定される空間
PCB	プログラム空間
DTB	データ空間
ADB	アディショナル空間
SPB	コンディションコードレジスタ (CCR) の S フラグの値が "0" の場合はユーザスタック空間が、"1" の場合はシステムスタック空間が用いられます。

バンクセレクトプリフィックスを使用した場合、例外的な動作をする命令があります。

表 2.9-2 にバンクセレクトプリフィックスに影響を受けない命令を示します。表 2.9-3 にはバンクセレクトプリフィックスの使用時に注意すべき命令を示します。

表 2.9-2 バンクセレクトプリフィックスの影響を受けない命令

命令の種類	命令	バンクセレクトプリフィックスの効果
ストリング命令	MOVS MOVSW SCEQ SCWEQ FILS FILSW	プリフィックスの有無にかかわらず、オペランドで指定されたバンクレジスタが使用されます。
スタック操作命令	PUSHW POPW	プリフィックスの有無にかかわらず、S フラグが "0" の場合はユーザスタックバンクレジスタ (USB) が、S フラグが "1" の場合はシステムスタックバンクレジスタ (SSB) が使用されます。
I/O アクセス命令	MOV A MOVX A, io MOVW A, io MOV io, A MOVW io, A MOV io, #imm8 MOVW io, #imm16 MOVB A, io : bp MOVB io : bp, A SETB io : bp CLRB io : bp BBC io : bp, rel BBS io : bp, rel WBTC io, bp WBTS io : bp	プリフィックスの有無にかかわらず、I/O 空間 ("000000 _H " ~ "0000FF _H ") がアクセスされます。
割込み復帰命令	RETI	プリフィックスの有無にかかわらず、システムスタックバンクレジスタ (SSB) が使用されます。

表 2.9-3 バンクセレクトプリフィックスの使用時に注意すべき命令

命令の種類	命令	説明
フラグ変更命令	AND CCR, #imm8 OR CCR, #imm8	プリフィックスの効果は次の命令まで及びます。
ILM 設定命令	MOV ILM, #imm8	プリフィックスの効果は次の命令まで及びます。
PS 復帰命令	POPW PS	PS 復帰命令に対してはバンクセレクトプリフィックスを付加しないでください。

2.9.2 コモンレジスタバンクプリフィックス (CMR)

コモンレジスタバンクプリフィックス (CMR) を、レジスタバンクをアクセスする命令の前に設定した場合は、レジスタバンクポインタ (RP) の値とは関係なく、レジスタアクセスを "000180H ~ 00018FH" に存在するコモンバンク (RP = "0" の場合に設定されるレジスタバンク) へ変更できます。

■ コモンレジスタバンクプリフィックス (CMR)

- 複数のタスク間におけるデータ交換を容易にするために、レジスタバンクポインタ (RP) の値に関係なく、比較的容易に固定レジスタバンクにアクセスする方法が必要です。そのため、F²MC-16LX ファミリではタスクで共通に使用できるコモンレジスタと呼ばれるコモンレジスタバンクを用意しています。コモンバンクは、"000180H ~ 00018FH" に存在します。
- コモンレジスタバンクプリフィックス (CMR) を、レジスタバンクにアクセスする命令の前に設定した場合、レジスタバンクポインタ (RP) の値とは関係なく、レジスタアクセスを "000180H ~ 00018FH" に存在するコモンバンク (RP=0 の場合に設定されるレジスタバンク) へ変更できます。

ただし、コモンレジスタバンクプリフィックス (CMR) を使用する際は表 2.9-4 に示した命令に注意してください。

表 2.9-4 コモンレジスタバンクプリフィックス (CMR) 使用時に注意すべき命令

命令の種類	命令	説明
ストリング命令	MOVS MOVSW SCEQ SCWEQ FILS FILSW	ストリング命令に対しては、CMR プリフィックスを付加しないでください。
フラグ変更命令	AND CCR, #imm8 OR CCR, #imm8	プリフィックスの効果は、次の命令まで及びます。
PS 復帰命令	POPW PS	プリフィックスの効果は、次の命令まで及びます。
ILM 設定命令	MOV ILM, #imm8	プリフィックスの効果は、次の命令まで及びます。

2.9.3 フラグ変化抑止プリフィックス (NCC)

実行に伴うフラグ変化を抑止する命令の前に、フラグ変化抑止プリフィックス (NCC) を設定します。

■ フラグ変化抑止プリフィックス (NCC)

不要なフラグ変化を抑止するために、フラグ変化抑止プリフィックス (NCC) を用います。命令の前にフラグ変化抑止プリフィックスが設定された場合、命令実行に伴うフラグ変化は抑止されます。フラグ変化の抑止対象となるフラグは T, N, Z, V, C です。

ただし、フラグ変化抑止プリフィックスを使用する際は表 2.9-5 に示した命令に注意してください。

表 2.9-5 フラグ変化抑止プリフィックス (NCC) の使用時に注意すべき命令

命令の種類	命令		説明
ストリング命令	MOVS SCEQ FILS	MOVSW SCWEQ FILSW	ストリング命令に対しては、NCC プリフィックスを付加しないでください。
フラグ変更命令	AND CCR, #imm8 OR CCR, #imm8		プリフィックスの有無にかかわらず、コンディションコードレジスタ (CCR) は命令の仕様どおりに変化します。プリフィックスの効果は、次の命令まで及びます。
PS 復帰命令	POPW PS		プリフィックスの有無にかかわらず、コンディションコードレジスタ (CCR) は命令の仕様どおりに変化します。プリフィックスの効果は、次の命令まで及びます。
ILM 設定命令	MOV ILM, #imm8		プリフィックスの効果は、次の命令まで及びます。
割込み命令、 割込み復帰命令	INT #vct8 INT adder16 RETI	INT9 INTP addr24	プリフィックスの有無にかかわらず、コンディションコードレジスタ (CCR) は命令の仕様どおりに変化します。
コンテキスト スイッチ命令	JCTX @A		プリフィックスの有無にかかわらず、コンディションコードレジスタ (CCR) は命令の仕様どおりに変化します。

2.9.4 プリフィックスコードに関する制約

プリフィックスコードの使用には、以下のような制約があります。

- プリフィックスコードや割込み / ホールド抑止命令の実行中は、割込み / ホールド要求を受付けません。
- 割込み / ホールド命令の前にプリフィックスコードを置いた場合、プリフィックスコードの効果は遅延します。
- 競合するプリフィックスコードが連続している場合は、最後のプリフィックスコードが有効となります。

■ プリフィックスコードと割込み抑止命令

プリフィックスコードと割込み抑止命令を表 2.9-6 に示します。

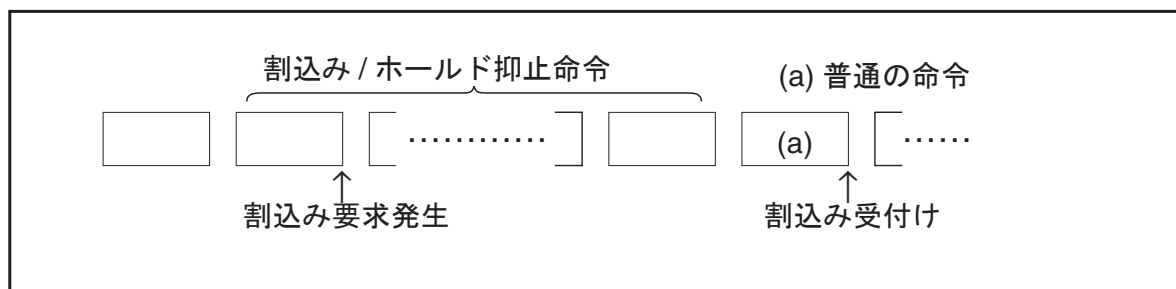
表 2.9-6 プリフィックスコードと割込み抑止命令サブスリープモード

	プリフィックスコード	割込み / ホールド抑止命令 (プリフィックスコードの効果が遅延させる命令)
割込みやホールド要求を受付けない	PCB DTB ADB SPB CMR NCC	MOV ILM, #imm8 OR CCR, #imm8 AND CCR, #imm8 POPW PS

● 割込み / ホールドの抑止

図 2.9-1 に示すように、プリフィックスコードや割込み / ホールド命令の実行中に発生した割込みやホールド要求は受付けられません。プリフィックスコードや割込み / ホールド命令以外の最初の命令が実行されるまで割込み / ホールドは処理されません。

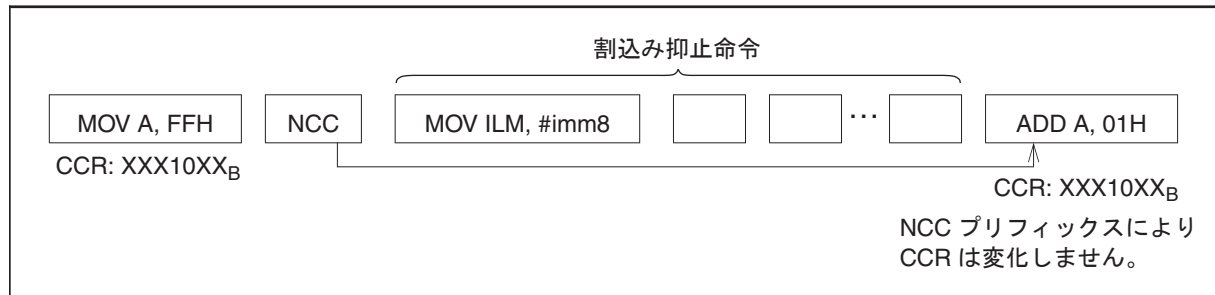
図 2.9-1 割込み / ホールドの抑止



● プリフィックスコードの効果遅延

図 2.9-2 に示すように、割込み / ホールド抑止命令の前にプリフィックスコードを設定した場合、プリフィックスコードの効果は、割込み / ホールド抑止命令後の命令に対して有効となります。

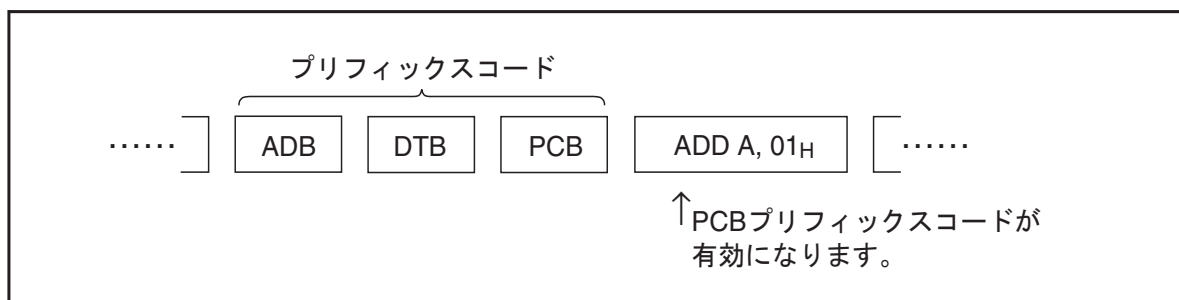
図 2.9-2 割込み / ホールド抑止命令とプリフィックスコード



■ プリフィックスコードの連続

競合するプリフィックスコード (PCB, ADB, DTB, SPB) が連続していた場合、最後のプログラム空間 (PCB) に設定された値が有効になります。

図 2.9-3 プリフィックスコードの連続



第3章

リセット

MB90820B シリーズの CPU の機能と動作について説明します。

- 3.1 リセットの概要
- 3.2 リセット要因と発振安定待ち時間
- 3.3 外部リセット端子
- 3.4 リセット動作
- 3.5 リセット要因ビット
- 3.6 リセットによる各端子の状態

3.1 リセットの概要

リセット要因が発生した場合，CPU は現在実行中の処理を中断し，リセット解除待ち状態になります。リセット解除後は，リセットベクタが示すアドレスから処理を開始します。

リセットには，以下に示す 5 種類の要因があります。

- ・ パワーオンリセットの発生
- ・ ウォッチドッグタイマのオーバフロー
- ・ $\overline{\text{RST}}$ 端子からの外部リセット要求
- ・ ソフトウェアリセット要求
- ・ クロックスーパバイザリセット要求 (MB90F828B のみ)

■ リセット要因

表 3.1-1 にリセット要因を示します。

表 3.1-1 リセット要因

リセットの種類	発生要因	マシクロック	ウォッチドッグタイマ	発振安定待ち
外部リセット	$\overline{\text{RST}}$ 端子への "L" レベル入力	前の状態を保持	前の状態を保持	なし
ソフトウェアリセット	低消費電力モード制御レジスタ (LPMCR) の内部リセット信号発生ビット (RST) に "0" を設定	前の状態を保持	前の状態を保持	なし
ウォッチドッグタイマリセット	ウォッチドッグタイマがオーバフローした場合	メインクロック (MCLK)	カウント停止	あり
パワーオンリセット	電源投入時	メインクロック (MCLK)	カウント停止	あり
クロックスーパバイザリセット*	メインクロックの故障を検出した場合	内蔵 CR 発振クロック	停止	なし

MCLK: メインクロック周波数 (発振クロックの 2 分周クロック)

※: MB90F828B のみ

● 外部リセット

外部リセットは，外部リセット端子 ($\overline{\text{RST}}$ 端子) に "L" レベルを入力すると発生します。 $\overline{\text{RST}}$ 端子への "L" レベル入力最小必要時間は，16 マシンサイクル (16/φ) です。外部リセットは発振安定待ち時間を取りません。

< 参考 >

RST端子からの外部リセット要求は、ライト動作中(MOVなどの転送系命令実行中)にリセット要因が発生した場合、命令が終了した後 CPU はリセットが解除されるまで待ちます。したがって、リセットが同時に入力されても通常のライト動作は終了します。

しかし、ストリング系命令で指定したカウンタのデータ転送が終了する前に、リセット解除待ち時間が開始するかもしれませんので注意してください。

● ソフトウェアリセット

ソフトウェアリセットは、低消費電力モード制御レジスタ (LPMCR) の内部リセット信号発生ビット (RST) に "0" を設定することにより、3 マシンサイクル ($3/\phi$) のリセットを発生します。ソフトウェアリセット時は、発振安定待ち時間を取りません。

● ウォッチドッグリセット

ウォッチドッグタイマ起動後設定した時間内に、ウォッチドッグタイマ制御レジスタ (WDTC) のウォッチドッグ制御ビット (WTE) に "0" を設定しなかった場合に発生するウォッチドッグタイマオーバフローによって、ウォッチドッグリセットは発生します。発振安定待ち時間は、クロック選択レジスタ (CKSCR) によって設定できます。

● パワーオンリセット

パワーオンリセットは、電源投入時に発生するリセットです。

発振安定待ち時間は 2^{16} 発振クロックサイクル ($2^{16}/\text{HCLK}$) の固定です。発振安定待ち時間が経過した後、リセットが実行されます。

● クロックスーパバイザリセット

メインクロックの故障を検出するとリセットを発生します。

クロックスーパバイザリセットは発振安定待ち時間の経過を待ちません。

< 参考 >

• クロックの定義

HCLK : 発振クロック周波数 (発振端子から供給されるクロック)

MCLK : メインクロック周波数 (発振クロックの2分周クロック)

ϕ : マシンクロック周波数 (CPU 動作クロック)

$1/\phi$: マシンサイクル (CPU 動作クロック周期)

マシンクロックの詳細は、「4.1 クロック」をご参照ください。

3.2 リセット要因と発振安定待ち時間

F²MC-16LX ファミリには5種類のリセット要因があり、リセット時の発振安定待ち時間はリセット要因によって異なります。

■ リセット要因と発振安定待ち時間

表 3.2-1 にリセット要因と発振安定待ち時間を示します。

表 3.2-1 リセット要因と発振安定待ち時間

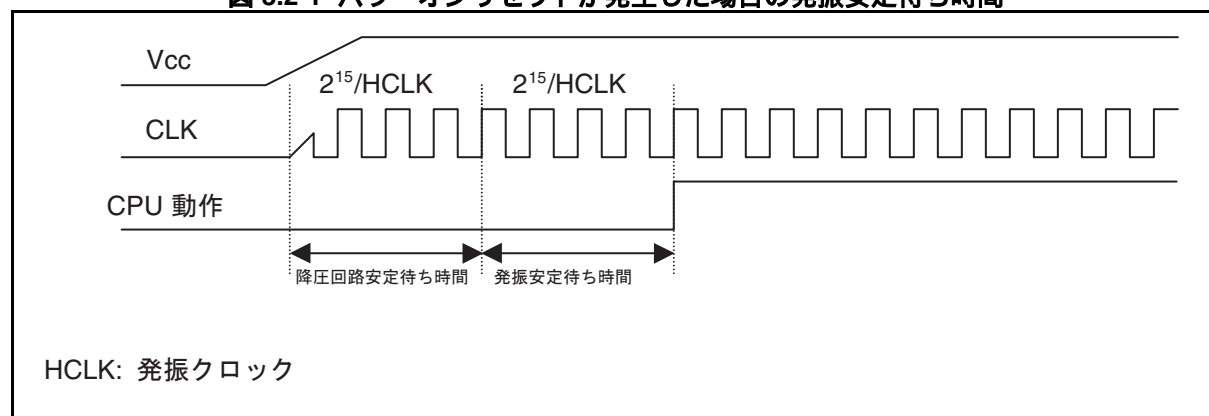
リセット要因	発振安定待ち時間 () 内は発振クロック周波数 4MHz 時
パワーオンリセット	$2^{16}/\text{HCLK}$ (約 16.39ms)
ウォッチドッグタイマ	$2^{16}/\text{HCLK}$ (約 16.39ms)
RST 端子からの外部リセット	なし (WS1, WS0 ビットは "11 _B " に初期化されます)
ソフトウェアリセット	なし (WS1, WS0 ビットは "11 _B " に初期化されます)
クロックスーパバイザリセット*	なし (WS1, WS0 ビットは "11 _B " に初期化されます)

HCLK: 発振クロック周波数

*: MB90F828B のみ

パワーオンリセットが発生した場合の発振安定待ち時間を図 3.2-1 に示します。

図 3.2-1 パワーオンリセットが発生した場合の発振安定待ち時間



< 注意事項 >

セラミックや水晶などの振動子は、発振を開始してから固有の振動数に安定するまで、一般に数 ms から数十 ms の発振安定待ち時間が必要です。そのため、使用する振動子に合わせた値を設定してください。詳細は、「4.5 発振安定待ち時間」を参照してください。

■ 発振安定待ちリセット状態

電源投入時のリセット、ストップモード、およびサブクロックモード中のリセットに対するリセット動作は、タイムベースタイマによって作られる発振安定待ち時間が経過してからとなります。このとき、外部リセット入力解除されていない場合は、外部リセット解除後にリセット動作を行います。

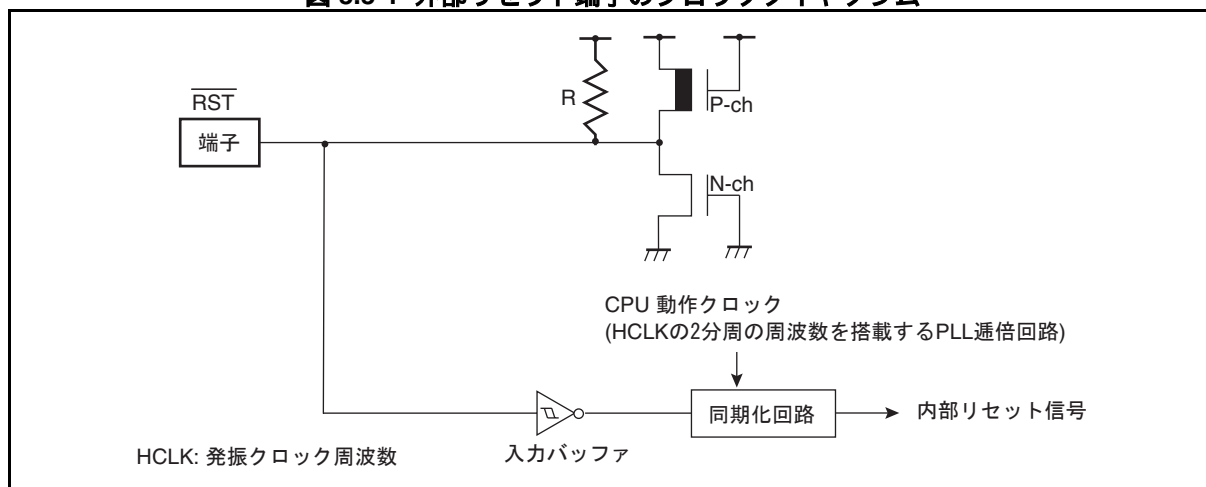
3.3 外部リセット端子

外部リセット端子 ($\overline{\text{RST}}$ 端子) は、リセット入力専用端子で "L" レベルの入力によって内部リセットを発生します。MB90820B シリーズのマイクロコントローラでは、CPU 動作クロックに同期してリセットがかかりますが、外部端子のみ非同期でリセットがかかります。

■ 外部リセット端子のブロックダイアグラム

外部リセット端子のブロックダイアグラムを図 3.3-1 に示します。

図 3.3-1 外部リセット端子のブロックダイアグラム



< 注意事項 >

- 書込み動作中のリセットによるメモリ破壊を防ぐため、 $\overline{\text{RST}}$ 端子入力の受付けをメモリが破壊されないサイクルで行います。
また、内部回路の初期化には、クロックが必要です。特に外部クロックで動作させる場合は、リセット入力時にクロックを入力する必要があります。

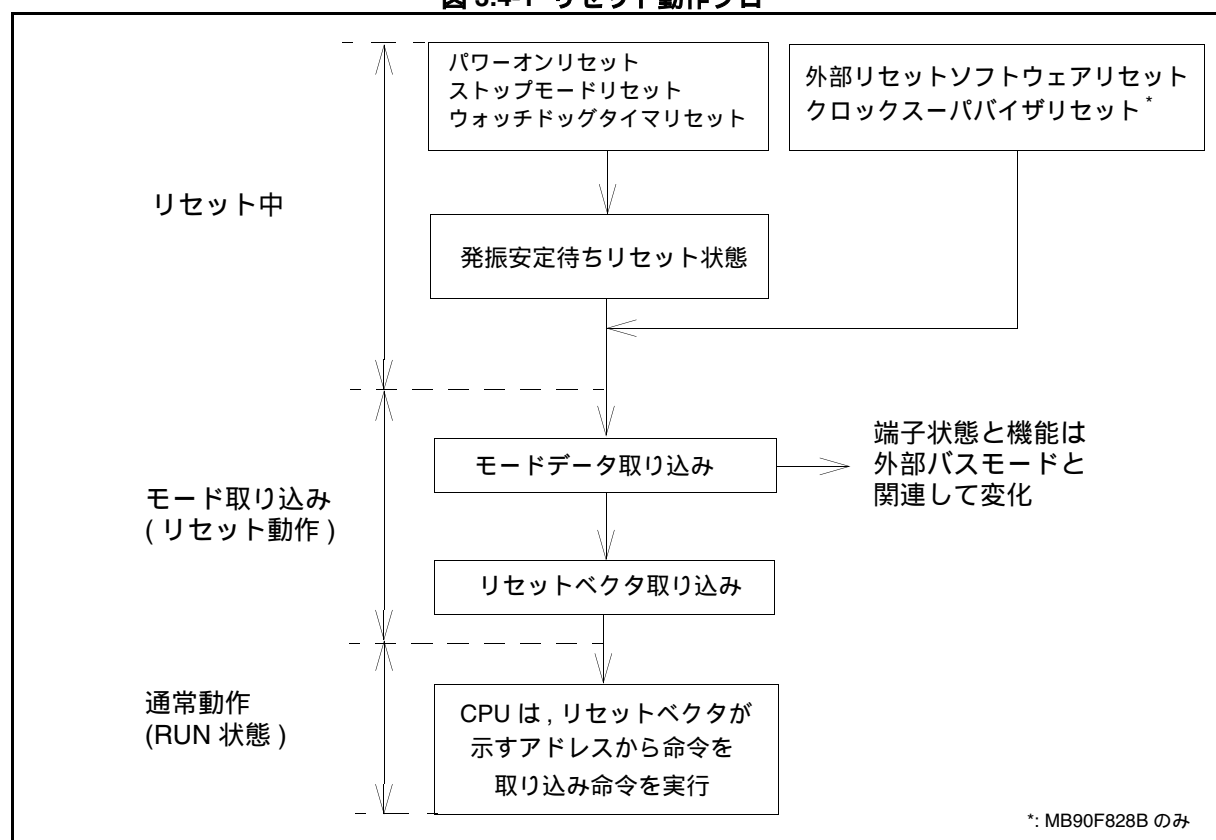
3.4 リセット動作

リセットが解除されると、モード端子の設定でモードデータとリセットベクタの読み出し先を選択し、モードフェッチを行います。このモードフェッチで、CPU の動作モードとリセット動作終了後の実行開始アドレスが決定されます。電源投入時、ストップモードからのリセットによる復帰では、発振安定待ち時間が経過してからモードフェッチを行います。

■ リセット動作の概要

リセット動作フローを図 3.4-1 に示します。

図 3.4-1 リセット動作フロー



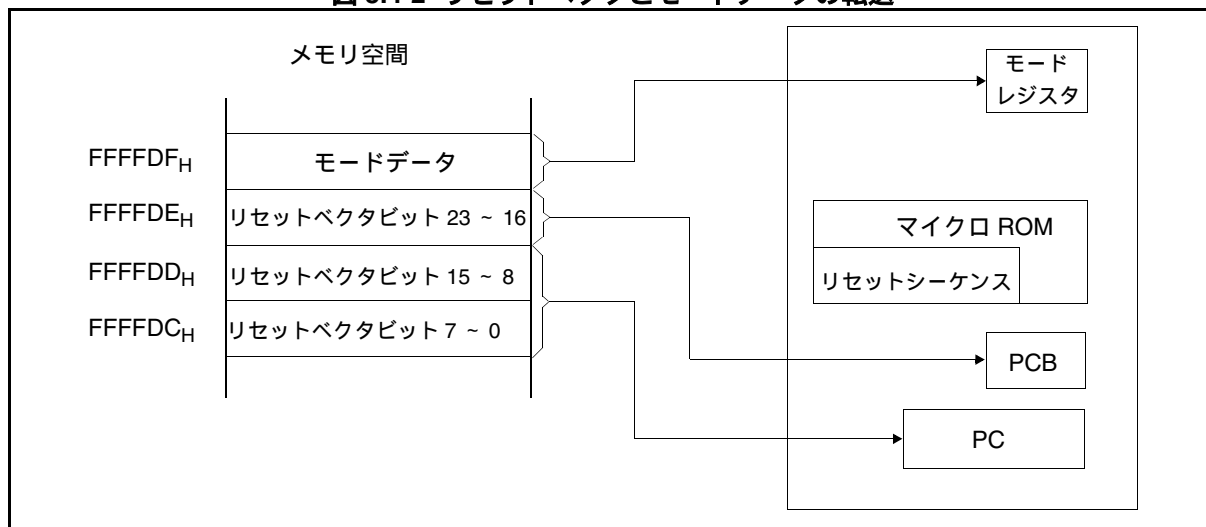
■ モード端子

モード端子 (MD0 ~ MD2) は、リセットベクタとモードデータの取込み方法を指定します。リセットベクタとモードデータの取込みは、リセットシーケンスで行います。モード端子の詳細は「8.2 モード端子 (MD2 ~ MD0)」を参照してください。

■ モードデータの取り込み

リセットが解除されると、CPU はリセットベクタとモードデータを CPU コア内の該当レジスタ内にハードウェア転送します。リセットベクタとモードデータは、"FFFFDC_H" ~ "FFFFDF_H" の 4 バイトに割り当てられています。CPU は、リセット解除で直ちにこれらのアドレスをバスに出力し、リセットベクタとモードデータを取り込みます。このモードフェッチで、CPU はリセットベクタが指すアドレスから処理を開始します。図 3.4-2 にリセットベクタとモードデータの転送を示します。

図 3.4-2 リセットベクタとモードデータの転送



● モードデータ (アドレス : FFFFDF_H)

モードレジスタの内容を変更できるのはリセット動作だけで、モードレジスタの設定はリセット動作以降に有効となります。モードデータの詳細は、「8.3 モードデータ」を参照してください。

● リセットベクタ (アドレス : FFFFDC_H ~ FFFFDE_H)

リセット動作終了後の実行開始アドレスをリセットベクタとして書き込みます。この内容のアドレスから実行を開始します。

3.5 リセット要因ビット

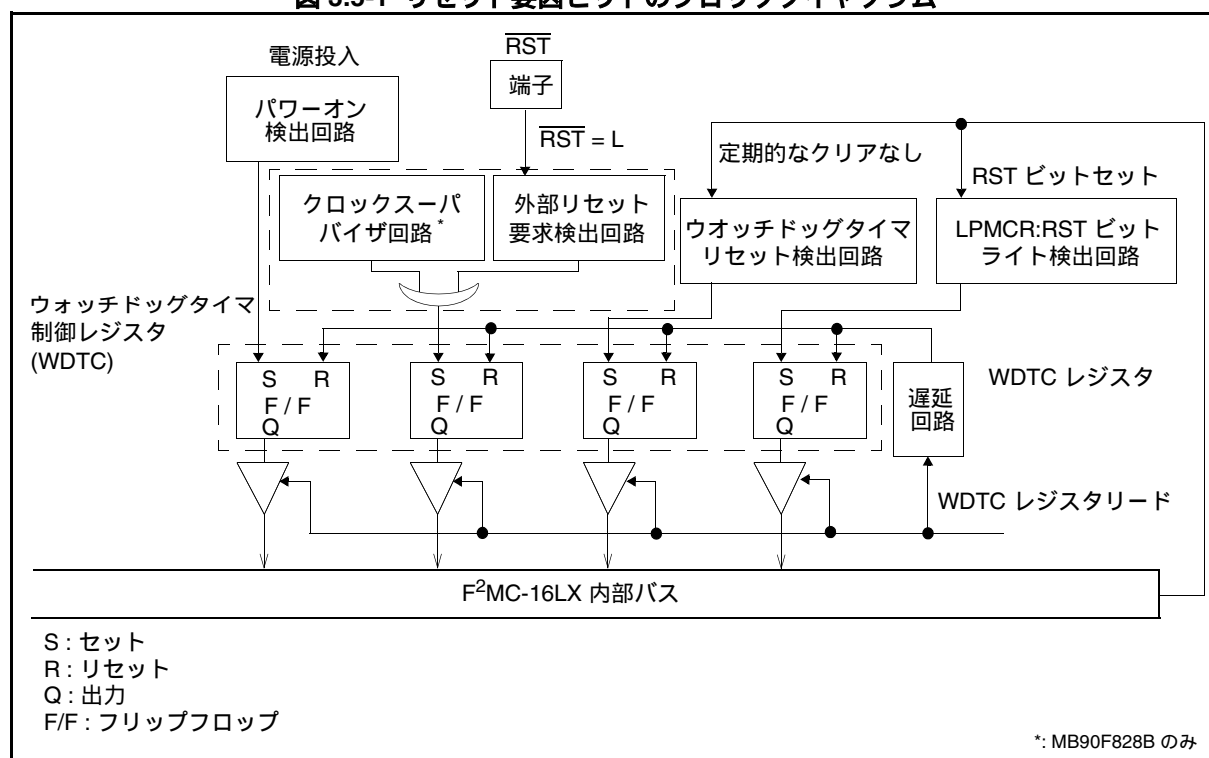
リセット発生要因は、ウォッチドッグタイマ制御レジスタ (WDTC) を読み出すことで識別することができます。

■ リセット要因ビット

図 3.5-1 に示すように、それぞれのリセット要因に対応したフリップフロップがあります。これらの内容は、ウォッチドッグタイマ制御レジスタ (WDTC) を読み出すと得られます。リセット解除後にリセット発生要因を識別する必要がある場合には、ウォッチドッグタイマ制御レジスタ (WDTC) の読出し値をソフトウェアで処理した上で、適切なプログラムへ分岐するようにしてください。

リセット要因ビットのブロック図を図 3.5-1 に示します。

図 3.5-1 リセット要因ビットのブロックダイアグラム



■ リセット要因フラグビットとリセット要因の対応

図 3.5-2 に、ウォッチドッグタイマ制御レジスタ (WDTC) のリセット要因フラグビットの構成を、表 3.5-1 にリセット要因フラグビットとリセット要因の対応を示します。

図 3.5-2 リセット要因フラグビットの構成 (ウォッチドッグタイマ制御レジスタ)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
ウォッチドッグタイマ制御レジスタ (WDTC)	PONR	-	WRST	ERST	SRST	WTE	WT1	WT0	XXXXXX111 _B
	(R)	(-)	(R)	(R)	(R)	(R)	(R)	(R)	

R: リードオンリ
W: ライトオンリ
X: 不定

表 3.5-1 リセット要因フラグビットとリセット要因の対応

リセット要因	PONR	WRST	ERST	SRST
パワーオンリセット	1	X	X	X
ウォッチドッグタイマオーバフロー	*	1	*	*
RST 端子からの外部リセット要求 クロックスーパバイザリセット (MB90F828B のみ)	*	*	1	*
ソフトウェアリセット要求	*	*	*	1

※: 前の状態を保持

X: 不定

■ リセット要因ビットの注意事項

● 複数のリセット要因が発生した場合

リセット要因が複数発生する場合は、ウォッチドッグタイマ制御レジスタ (WDTC) の対応するそれぞれのリセット要因ビットが "1" にセットされます。例えば、RST 端子からの外部リセット要求の発生とウォッチドッグタイマのオーバフローが同時に発生した場合、ERST ビットと WRST ビットの両方が "1" になります。

● パワーオンリセットの場合

パワーオンリセットの場合には、PONR ビットが "1" にセットされますが、PONR ビット以外のリセット要因ビットは不定となります。このため、PONR ビットが "1" の場合は、PONR ビット以外のリセット要因ビットを無視するようにソフトウェアを作成してください。

● リセット要因ビットのクリア

リセット要因ビットは、ウォッチドッグタイマ制御レジスタ (WDTC) を読み出したときのみクリアされます。それぞれのリセット要因に対応するビットに生じたフラグは、その後ほかの要因でリセットが発生してもクリアされず、"1" のままとなります。

3.6 リセットによる各端子の状態

リセットによる各端子の状態について説明します。

■ リセット動作中の端子の状態

リセット中の端子の状態は、モード端子の設定によって決定されます。(MD2 ~ MD0 = 011_B)

- 内部ベクタモード設定時

I/O 端子（周辺機能端子）はすべてハイインピーダンスになり、モードデータの読出し先は内部 ROM になります。

■ モードデータ読出し後の端子の状態

モードデータ読み出し後の端子の状態は、モードデータによって決定されます。(M1 と M0 = 00_B)

- シングルチップモード選択時 (M1, M0 = 00_B)

I/O 端子（周辺機能端子）はすべてハイインピーダンスになり、モードデータの読出し先は内部 ROM になります。

< 注意事項 >

リセット要因が発生したときハイインピーダンスとなる端子は、その端子に接続した機器が誤動作しないように配慮してください。

リセット中の各端子状態は、表 6.7-1 を参照してください。

第4章

クロック

クロックの機能と動作について説明します。

- 4.1 クロック
- 4.2 クロック発生部のブロックダイアグラム
- 4.3 クロック選択レジスタ
- 4.4 クロックモード
- 4.5 発振安定待ち時間
- 4.6 振動子および外部クロックの接続

4.1 クロック

クロック発生部では、CPU と周辺機能の動作を制御する内部クロックを制御しています。この内部動作クロックをマシンクロック (ϕ) とよび、その1周期がマシンサイクルです。

原発振から供給されるクロックを発振クロックといい、内部の PLL 発振によって供給されるクロックを PLL クロックといいます。

■ クロックの概要

- クロック発生部には、発振クロックを生成する発振回路が含まれています。外部発振器はこの回路に接続しています。発振クロックの供給は、クロック発生部に外部クロックを入力することによっても行うことができます。
- またクロック発生部には、発振クロックの倍数である5つのクロックを生成する PLL クロック通倍回路も含まれています。
- クロック発生部は、発振安定待ち時間および PLL クロック通倍を制御します。また、クロックセクタでクロックを切換えることによって内部クロックの動作も制御します。

● 発振クロック (HCLK)

発振回路に外部振動子を接続するか、または外部クロックの入力によって発生させたクロックです。

● メインクロック (MCLK)

発振クロックの2分周クロックであり、タイムベースタイマおよびクロックセクタへクロック入力を供給します。

● PLL クロック (PCLK)

発振クロックを、PLL 通倍回路 (PLL 発振回路) により通倍したクロックです。4種類のクロックから選択可能です。

● マシンクロック (ϕ)

CPU およびリソースの動作を制御するクロックです。1クロック周期が1マシンサイクルとなります ($1/\phi$)。マシンクロックは、メインクロック (2分周されたソースクロック周波数から生成) および5種類のクロック (ソースクロックの倍数) から選択可能です。

< 注意事項 >

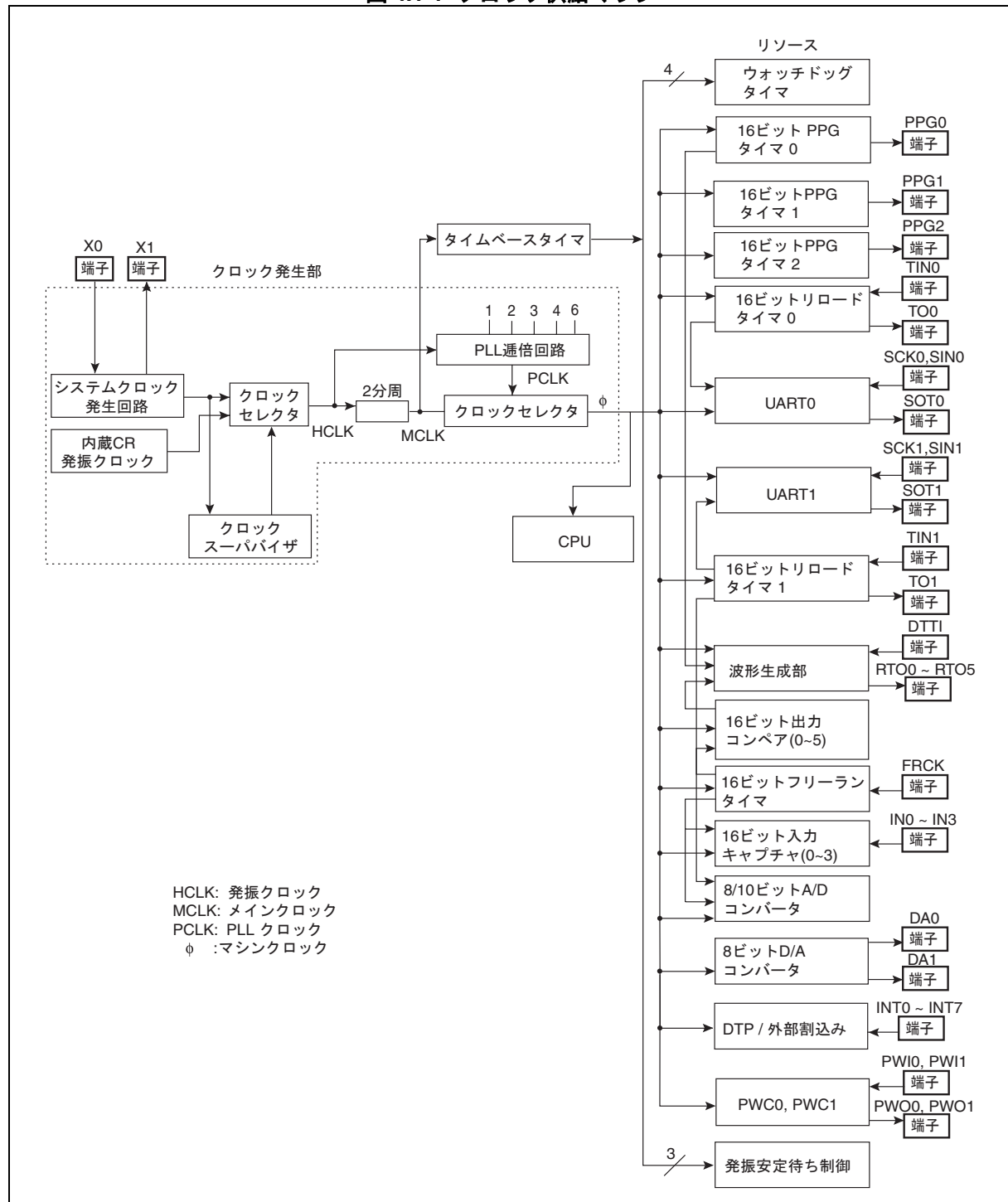
- 動作電圧が5Vの場合、発振クロックは3MHz ~ 48MHz を生成可能ですが、CPU およびリソース機能の最大動作周波数は24MHzです。最大動作周波数を超える通倍率を設定した場合、デバイスは正常に動作しません。例えば、原発振が12MHzの原発振が生成された場合には、2通倍だけが設定可能です。
- 4MHz ~ 24MHz の PLL クロック発振が可能です。この範囲は動作電圧および通倍数によります。詳細は「データシート」をご参照ください。

● クロック供給マップ

- クロック発生部で生成したマシンクロックは、CPU やリソース機能の動作を制御するクロックとして供給しているため、CPU とリソース機能の動作は、メインクロックと PLL クロック（クロックモード）の切換えや PLL クロック通倍率の切換えによって影響を受けます。
- 一部のリソース機能には、タイムベースタイマから分周出力を受信しているため、リソース部で動作に最適なクロックを選択可能です。

図 4.1-1 に、クロック供給マップを示します。

図 4.1-1 クロック供給マップ



4.2 クロック発生部のブロックダイアグラム

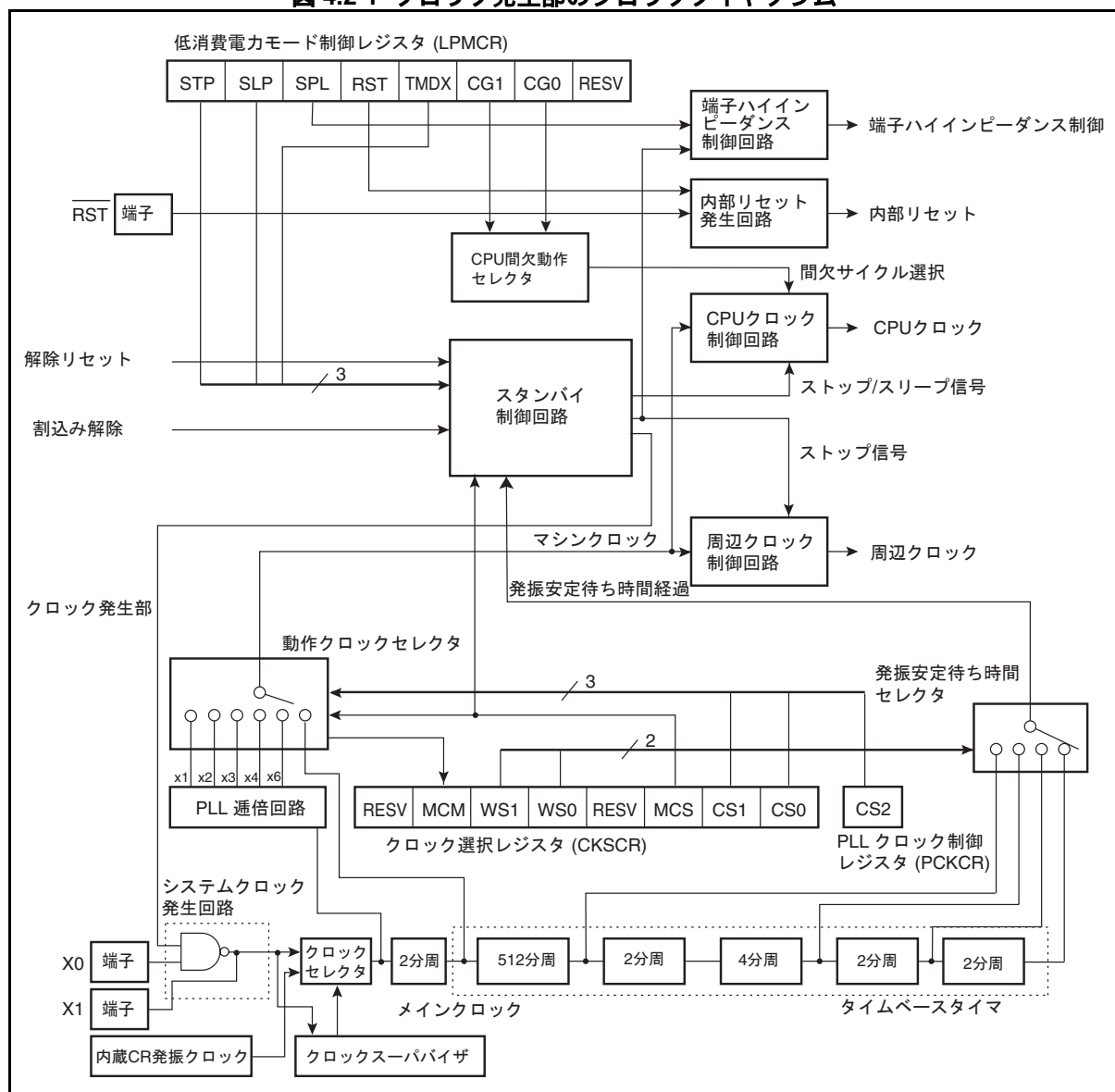
クロック発生部は、次の5つのブロックで構成されています。

- ・システムクロック発生回路
- ・PLL 逡倍回路
- ・動作クロックセクタ
- ・クロック選択レジスタ (CKSCR)
- ・発振安定待ち時間セクタ

■ クロック発生部のブロックダイアグラム

図 4.2-1 はクロック発生部のブロックダイアグラムを示します。また、図 4.2-1 はスタンバイ制御回路およびタイムベースタイマ回路も含まれます。

図 4.2-1 クロック発生部のブロックダイアグラム



● システムクロック発生回路

外部に接続した振動子によって発振クロック (HCLK) を発生します。その代わりに、外部クロックを入力できます。

● PLL 逡倍回路

PLL 発振により発振クロック (HCLK) を逡倍し、周波数の逡倍であるクロックを CPU クロックセクタへ供給します。

● 動作クロックセクタ

メインクロックと5種類の PLL クロックの中から、CPU クロック制御回路とリソースクロック制御回路に供給するクロックを選択します。

● クロック選択レジスタ (CKSCR) と PLL クロック制御レジスタ (PCKCR)

発振クロックと PLL クロックの切換え、発振安定待ち時間の選択、および PLL クロックの逡倍率の選択などを行います。

● 発振安定待ち時間セクタ

- ストップモード解除時、およびウォッチドッグタイマリセットが発生した場合の発振クロックの発振安定待ち時間を選択する回路です。
- 3種類のタイムベースタイマ出力から選択します。それ以外の場合は、発振安定待ち時間は選択されません。

4.3 クロック選択レジスタ

クロック選択レジスタは、クロック選択レジスタ (CKSCR) と PLL クロック制御レジスタ (PCKCR) から構成されます。

■ クロック選択レジスタ

図 4.3-1 に、クロック選択レジスタ (CKSCR) と PLL クロック制御レジスタ (PCKCR) を示します。

図 4.3-1 クロック選択レジスタ

CKSCR	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
	RESV	MCM	WS1	WS0	RESV	MCS	CS1	CS0	11111100 _B
	(R/W)	(R)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
PCKCR	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
	-	-	-	-	RESV	RESV	RESV	CS2	XXXX0000 _B
	-	-	-	-	(W)	(W)	(W)	(W)	

R : リードオンリ
W : ライトオンリ
R/W : リード / ライト可能
- : 未定義

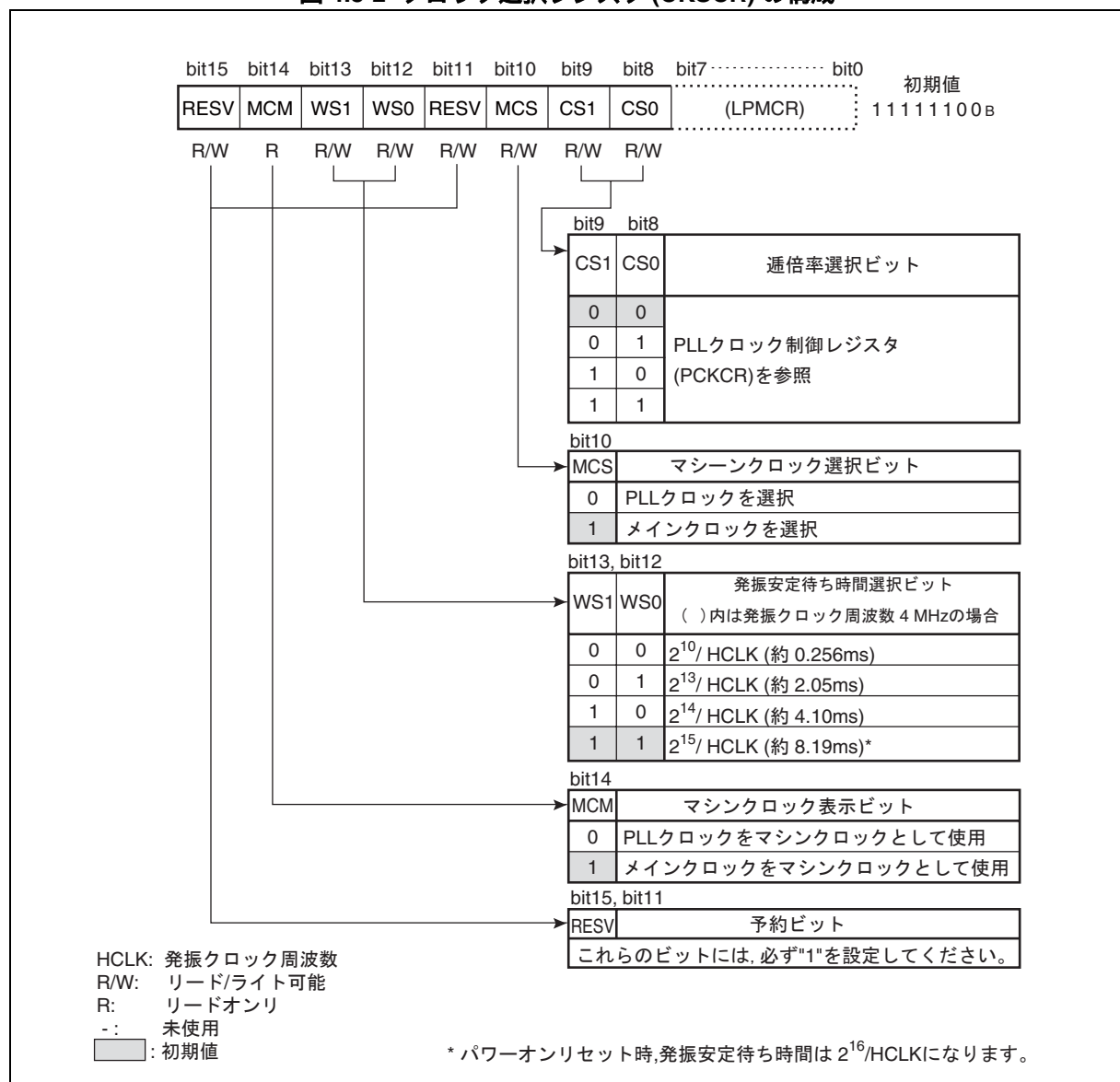
4.3.1 クロック選択レジスタ (CKSCR)

クロック選択レジスタ (CKSCR) は、メインクロックと PLL クロックの切換え、発振安定待ち時間の選択、および PLL クロックの通倍率の選択などを行うレジスタです。

■ クロック選択レジスタ (CKSCR) の構成

図 4.3-2 はクロック選択レジスタ (CKSCR) の構成を示し、表 4.3-1 にクロック選択レジスタ (CKSCR) の各ビットの機能について説明します。

図 4.3-2 クロック選択レジスタ (CKSCR) の構成



< 注意事項 >

マシニングロック選択ビットが設定されていない場合、メインクロックがマシニングロックとして使用されます。

表 4.3-1 クロック選択レジスタ (CKSCR) の各ビットの機能説明

ビット名		機能
bit15, bit11	RESV: 予約ビット	(注意事項) 必ず "1" を設定してください。
bit14	MCM: マシナクロック表示 ビット	<ul style="list-style-type: none"> メインクロックか PLL クロックのどちらがマシナクロックとして動作しているかを表示します。 MCM = 0 の場合: PLL クロックが選択されています。 MCM = 1 の場合: メインクロックが選択されています。 MCS = 0 および MCM = 1 である場合, PLL クロック発振安定待ち状態です。 書込みは動作に影響しません。
bit13, bit12	WS1, WS0: 発振安定待ち時間選 択ビット	<ul style="list-style-type: none"> ストップモード解除後, 発振クロックの発振安定待ち時間を選択します。 すべてのリセット要因で "11B" に初期化されます。 <p>(注意事項)</p> <ul style="list-style-type: none"> 発振安定待ち時間は, 使用する振動子に合わせて適切な値を設定する必要があります。「3.2 リセット要因と発振安定待ち時間」をご参照ください。 <p>(参考)</p> <ul style="list-style-type: none"> PLL クロック時の発振安定待ち時間は, $2^{14}/\text{HCLK}$ で固定です。
bit10	MCS: マシナクロック選択 ビット	<ul style="list-style-type: none"> メインクロックか PLL クロックのどちらをマシナクロックに選択するかを設定します。 "0" に設定した場合: PLL クロックを選択します。 "1" に設定した場合: メインクロックを選択します。 この MCS ビットが "1" のときに "0" を設定した場合は, PLL クロックの発振安定待ち時間が発生するため, 自動的にタイムベースタイマがクリアされ, 同時にタイムベースタイマ制御レジスタの TBOF ビットもクリアされます。 PLL クロックの発振安定待ち時間は, $2^{14}/\text{HCLK}$ で固定です (発振クロック周波数 4MHz 時, 発振安定待ち時間は約 2ms になります)。 メインクロック選択時の動作クロック周波数は, 発振クロックを 2 分周したクロックとなります (発振クロック周波数 4MHz 時, 動作クロックは 2MHz になります)。 パワーオンウォッチドッグリセットで "1" に初期化されます。 <p>(注意事項)</p> <p>MCS ビットが "1" のときに, タイムベースタイマ制御レジスタ (TBTC) の TBIE ビットまたは割込みレベルレジスタ (ILM) によりタイムベースタイマ割込みがマスクされている場合のみ "0" を書き込んでください。</p>
bit9, bit8	CS1, CS0: 通倍率選択ビット	<ul style="list-style-type: none"> PCKCR の CS2 ビットと共に, PLL クロックの通倍率を選択します。 5 種類の通倍率から選択できます。 すべてのリセット要因で "00B" に初期化されます。 CS2, CS1 および CS0 ビットと PLL クロック通倍率選択の関係については, PCKCR をご参照ください。 <p>(注意事項)</p> <p>MCS ビットが "0" のときには, 書込みは禁止です。MCS ビットを "1" にした (メインクロックモード) 後に, CS1 と CS0 ビットに書き込んでください。</p>

4.3.2 PLL クロック制御レジスタ (PCKCR)

PLL クロック制御レジスタは、CKSCR の CS1 と CS0 ビットと共に、PLL クロック通倍率を選択します。

■ PLL クロック制御レジスタ (PCKCR) の構成

図 4.3-3 に PLL クロック制御レジスタ (PCKCR) の構成を示します。図 4.3-3 に PLL クロック制御レジスタ (PCKCR) の各ビットの機能を示します。

図 4.3-3 PLL クロック制御レジスタ (PCKCR) の構成

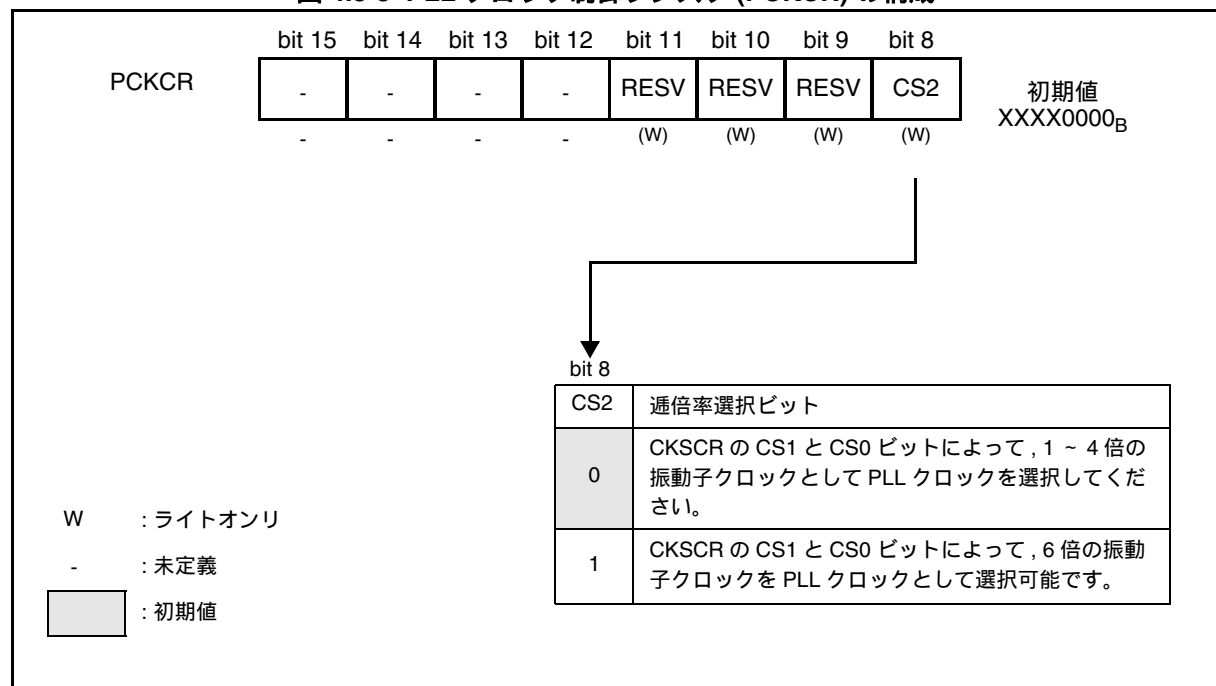


表 4.3-2 PLL クロック制御レジスタ (PCKCR) の各ビットの機能説明

ビット名		機能																												
bit 15 ~ bit 12	未定義ビット	<ul style="list-style-type: none">読出し時：値は不定です。書込み時：動作に影響を与えません。																												
bit 11 ~ bit 9	予約ビット	<ul style="list-style-type: none">読み出し時：値は不定です。必ず "0" を設定してください。																												
bit8	CS2: 通倍率選択 ビット	<ul style="list-style-type: none">PCKCR の C2 ビットと共に、PLL クロック通倍率を選びます。5 つの異なる通倍率から選択可能です。すべてのリセット要因によって "0" に初期化されます。読出し値は不定です。CS2, CS1 および CS0 ビットの推奨設定																												
		<table><tr><td>CS2</td><td>CS1</td><td>CS0</td><td>PLL クロック複合時間</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1 × HCLK (4 MHz)</td></tr><tr><td>0</td><td>0</td><td>1</td><td>2 × HCLK (8 MHz)</td></tr><tr><td>0</td><td>1</td><td>0</td><td>3 × HCLK (12 MHz)</td></tr><tr><td>0</td><td>1</td><td>1</td><td>4 × HCLK (16 MHz)</td></tr><tr><td>1</td><td>1</td><td>0</td><td>6 × HCLK (24 MHz)</td></tr><tr><td colspan="3">(その他)</td><td>設定禁止</td></tr></table>	CS2	CS1	CS0	PLL クロック複合時間	0	0	0	1 × HCLK (4 MHz)	0	0	1	2 × HCLK (8 MHz)	0	1	0	3 × HCLK (12 MHz)	0	1	1	4 × HCLK (16 MHz)	1	1	0	6 × HCLK (24 MHz)	(その他)			設定禁止
		CS2	CS1	CS0	PLL クロック複合時間																									
		0	0	0	1 × HCLK (4 MHz)																									
		0	0	1	2 × HCLK (8 MHz)																									
		0	1	0	3 × HCLK (12 MHz)																									
		0	1	1	4 × HCLK (16 MHz)																									
		1	1	0	6 × HCLK (24 MHz)																									
		(その他)			設定禁止																									
		(注意事項)																												
CKSCR の MCS ビットが "0" の場合、このビットへの書込みは禁止です。																														
CKSCR の MCS を "1" に設定した後に CS2 ビットに書込みをしてください。(メインクロックモード)																														

4.4 クロックモード

クロックモードには、メインクロックモードと PLL クロックモードの2種類があります。

■ メインクロックモード, PLL クロックモード

● メインクロックモード

メインクロックモードでは、CPUとリソース機能の動作クロックとして、発振クロック (HCLK) を2分周したメインクロックを使用し、PLL クロックは禁止になります。

● PLL クロック

PLL クロックモードでは、CPUとリソース機能の動作クロックとして、PLL クロックを使用します。PLL クロックの通倍率は、クロック選択レジスタ (CKSCR: CS1, CS0) および PLL クロック制御レジスタ (PCKCR: CS2) により設定できます。

■ クロックモードの移行

クロック選択レジスタ (CKSCR) の MCS ビットへの書込みによって、メインクロックモードと PLL クロックモードが切り換わります。

● メインクロックモードから PLL クロックモードへの移行

メインクロックモードの状態では、クロック選択レジスタ (CKSCR) の MCS ビットを "1" から "0" に書き換えた場合には、PLL クロックの発振安定待ち期間 ($2^{14}/HCLK$) 後に、メインクロックから PLL クロックに切り換わります。

● PLL クロックモードからメインクロックモードへの移行

PLL クロックモードの状態では、クロック選択レジスタ (CKSCR) の MCS ビットを "0" から "1" に書き換えた場合には、PLL クロックとメインクロックのエッジが一致するタイミング (1 ~ 12PLL クロック後) で PLL クロックからメインクロックに切り換わります。

< 注意事項 >

クロック選択レジスタ (CKSCR) の MCS ビットを書き換えても即座にマシンクロックの切換えが行われません。マシンクロックに依存するリソースを操作する場合には、リソースの操作を行う前に、クロック選択レジスタ (CKSCR) の MCM ビットを参照してマシンクロックの切換えが行われたことを確認してください。

■ PLL クロック通倍率の選択

クロック選択レジスタ (CKSCR: CS1, CS0) および PLL クロック制御レジスタ (PCKCR: CS2) ビットに "000_B" ~ "011_B" または "110_B" の値を設定することによって、1 ~ 5 通倍の5種類の PLL クロック通倍率が選択できます。

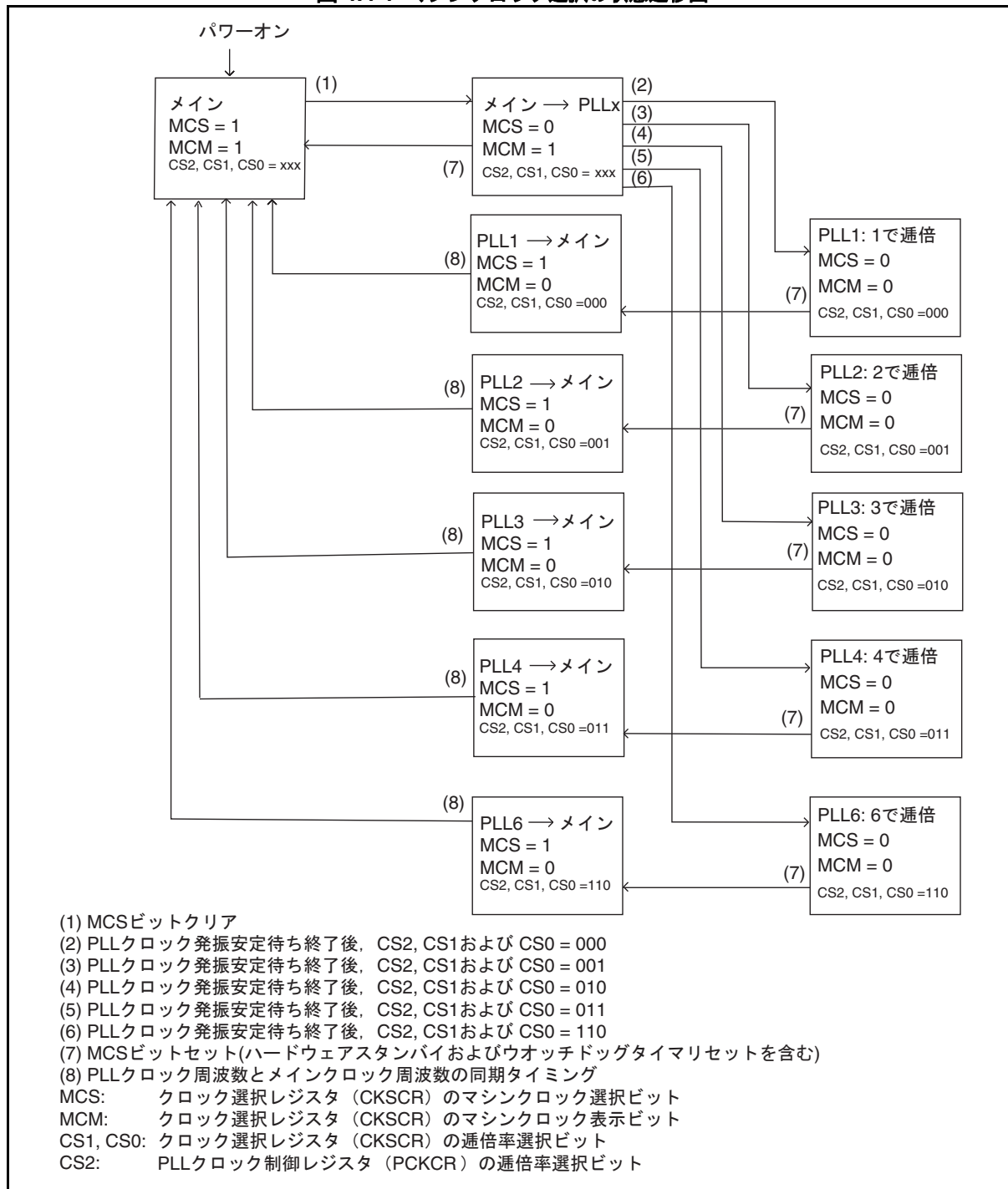
■ マシンクロックの選択

PLL 通倍回路から出力される PLL クロックまたは原発振の2分周クロックが、マシンクロックとなります。このマシンクロックがCPUおよびリソース機能に供給されます。メインクロック、PLL クロックは、クロック選択レジスタ (CKSCR) の MCS ビットへの

書込みにより選択できます。

図 4.4-1 に、マシンのクロックの切換えによる状態遷移を示します。

図 4.4-1 マシンのクロック選択の状態遷移図



< 注意事項 >

マシンのクロックの初期値は、メインクロック (CKSCR の MCS = 1) です。

4.5 発振安定待ち時間

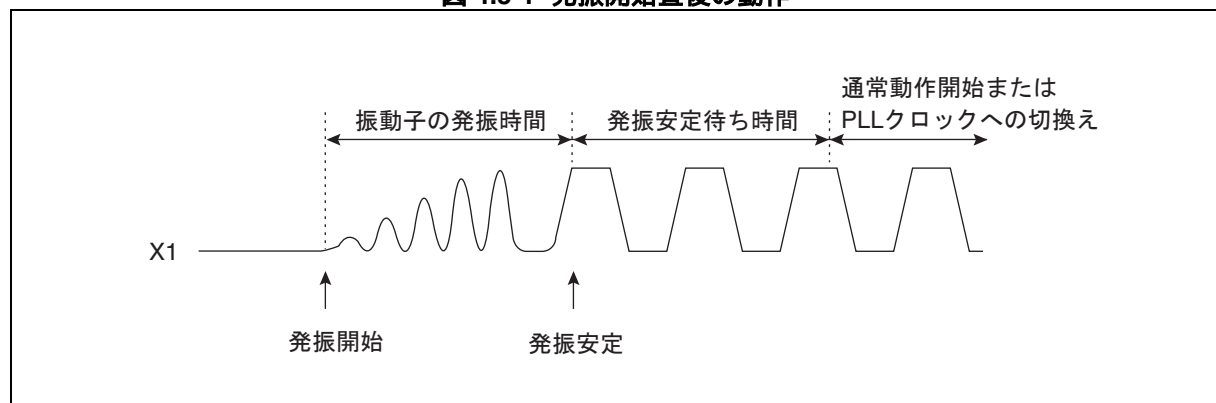
電源投入時，ストップモード解除時，ウォッチドッグタイマリセット発生の場合は，発振クロックが開始しますが，発振は当初不安定なため，発振安定待ち時間をとる必要があります。また，メインクロックから PLL クロックへ切換えるときも，PLL 発振開始後に発振安定待ち時間が必要となります。

■ 発振安定待ち時間

- セラミックや水晶などの振動子は，発振を開始してから固有の周波数で安定するまでに，一般的に数 ms から数十 ms の時間が必要です。
- このため，発振開始直後は CPU の動作を禁止し，発振安定待ち時間が経過して十分発振が安定した時点で CPU にクロックを供給するようにします。
- 振動子の種類（水晶，セラミックなど）によって発振が安定するまでの時間が異なるため，使用する振動子に合わせて適切な発振安定待ち時間を選択する必要があります。発振安定待ち時間は，クロック選択レジスタ (CKSCR) の設定で選択できます。

メインクロックから PLL クロックへ切り換える際は，PLL 発振安定待ち時間の間，CPU はメインクロックで動作し続け，その後 PLL クロックに切り換わります。図 4.5-1 に発振開始直後の動作を示します。

図 4.5-1 発振開始直後の動作



4.6 振動子および外部クロックの接続

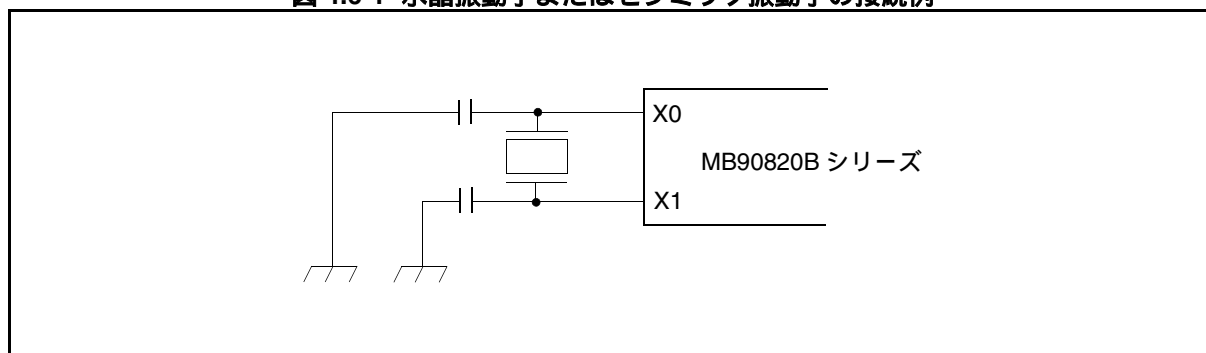
MB90820B シリーズには、システムクロック発生回路が内蔵されており、この回路に外部振動子を接続することでシステムクロックを発生できます。
また、外部で生成したクロックを入力することもできます。

■ 振動子と外部クロックの接続

● 水晶振動子またはセラミック振動子の接続例

水晶振動子またはセラミック振動子は、図 4.6-1 に示すように接続してください。

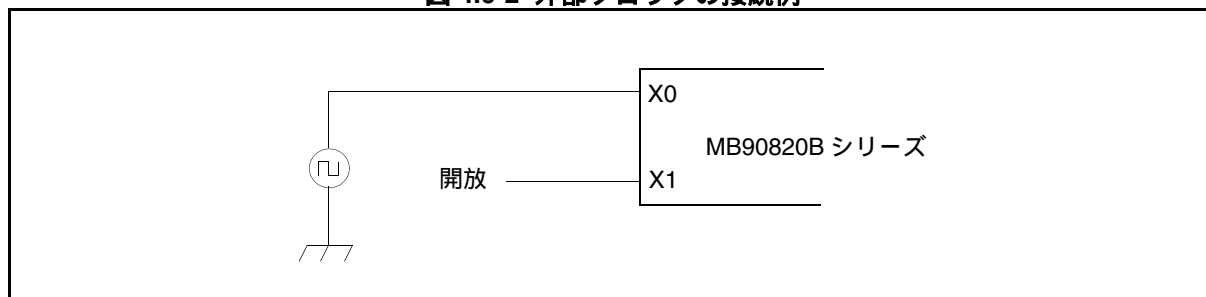
図 4.6-1 水晶振動子またはセラミック振動子の接続例



● 外部クロックの接続例

図 4.6-2 に示すように、外部クロックは、X0 端子に接続し、X1 端子は開放にしてください。

図 4.6-2 外部クロックの接続例



第5章

クロックスーパーバイザ

クロックスーパーバイザの機能と動作について説明します。

(MB90F828B のみ本機能は使用可能です)

- 5.1 クロックスーパーバイザの概要
- 5.2 クロックスーパーバイザの構成
- 5.3 クロックスーパーバイザのレジスタ
- 5.4 クロックスーパーバイザの動作説明
- 5.5 クロックスーパーバイザ使用上の注意

5.1 クロックスーパーバイザの概要

クロックスーパーバイザは、メインクロックの発振を監視して発振停止となった場合に、内蔵の CR 発振回路で生成される CR クロックに切り換わることによって暴走することを防止します。

■ クロックスーパーバイザの概要

- メインクロック発振を監視して、発振が停止したことを検出すると内部リセットを発生し、内蔵の CR クロックに切り換えます。
リセット要因が、クロックスーパーバイザのリセットなのか、あるいは他のリセット要因によるものかはリセット要因レジスタ (RSRR) にて確認できます。
 - メインクロックの発振停止が検出されるのは、メインクロックの立上りエッジが、CR クロックで 4 サイクルの間検出されなかった場合です。したがって、メインクロックの周期が CR クロックの 4 サイクル以上の場合、メインクロックの発振停止を検出してしまう可能性があります。
 - レジスタ設定によりリセット出力を禁止することが可能です。
 - メインクロックのストップモード時はクロック停止となりますが、その間クロックの監視は禁止状態（監視禁止）となります。
-

< 注意事項 >

CR クロックの周期などはデータシートを参照してください。

5.2 クロックスーパーバイザの構成

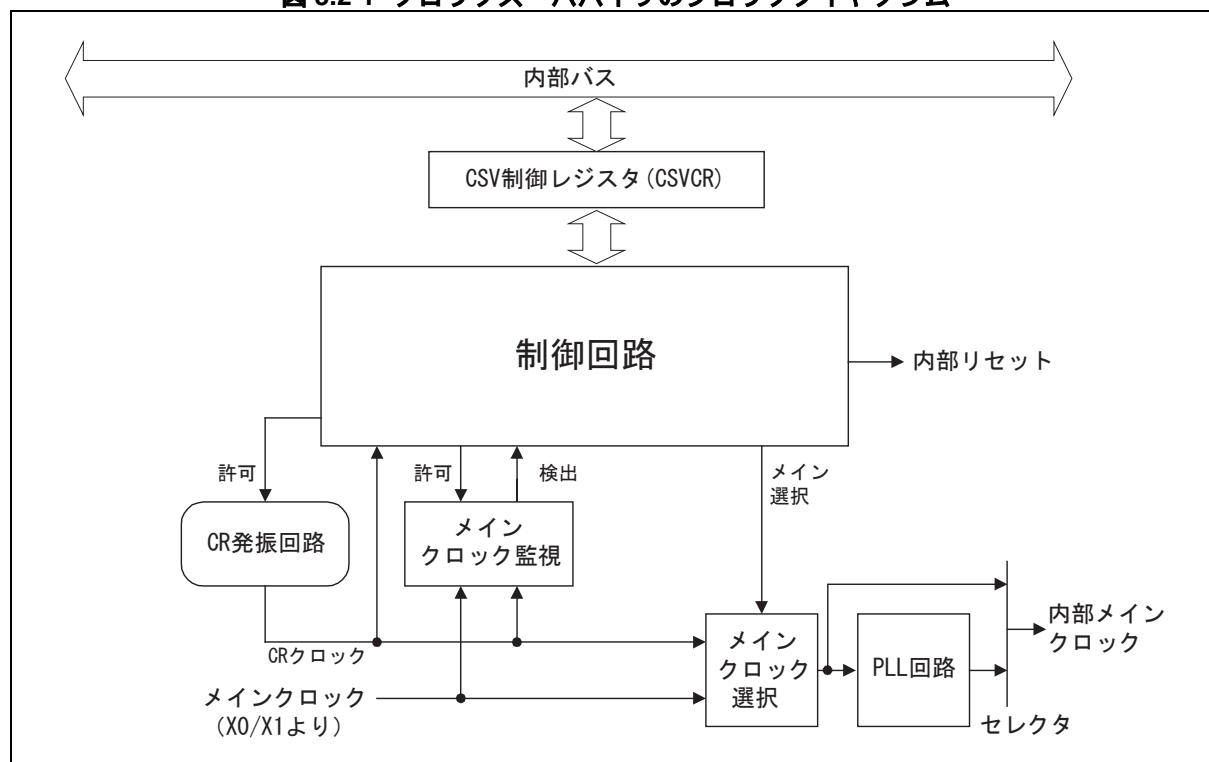
クロックスーパーバイザは、以下のブロックで構成されています。

- 制御回路
- CR 発振回路
- メインクロック監視
- メインクロック選択
- CSV 制御レジスタ (CSVCR)

■ クロックスーパーバイザのブロックダイアグラム

クロックスーパーバイザのブロックダイアグラムを図 5.2-1 に示します。

図 5.2-1 クロックスーパーバイザのブロックダイアグラム



● 制御回路

CSV 制御レジスタ (CSVCR) の情報により、各種クロックおよびリセットなどの制御を行います。

● CR 発振回路

内蔵 CR 発振回路です。制御回路の制御信号により発振 ON/OFF の制御が可能です。
クロック停止検出後の内部クロックとしても使用します。

● メインクロック監視

メインクロックの停止を監視します。

● メインクロック選択

メインクロックの停止を検出したときに、CR クロックを内部メインクロックとして出力します。

● CSV 制御レジスタ (CSVCR)

クロックの監視および CR クロックの制御や停止検出の情報を確認します。

5.3 クロックスーパーバイザのレジスタ

クロックスーパーバイザのレジスタについて説明します。

■ クロックスーパーバイザのレジスタ一覧

クロックスーパーバイザのレジスタを図 5.3-1 に示します。

図 5.3-1 クロックスーパーバイザのレジスタ

クロックスーパーバイザ制御レジスタ (CSVCR)								
bit	7	6	5	4	3	2	1	0
アドレス 00008A _H	予約	MM	予約	RCE	MSVE	予約	予約	予約
	-	R	-	R/W	R/W	-	-	R/W
初期値 00011100 _B								
R/W : リード / ライト可能								
R : リードオンリ								

5.3.1 クロックスーパーバイザ制御レジスタ (CSVCR)

クロックスーパーバイザ制御レジスタ (CSVCR) は、各機能の許可および状態の確認を行うレジスタです。

■ クロックスーパーバイザ制御レジスタ (CSVCR)

図 5.3-2 クロックスーパーバイザ制御レジスタ (CSVCR)

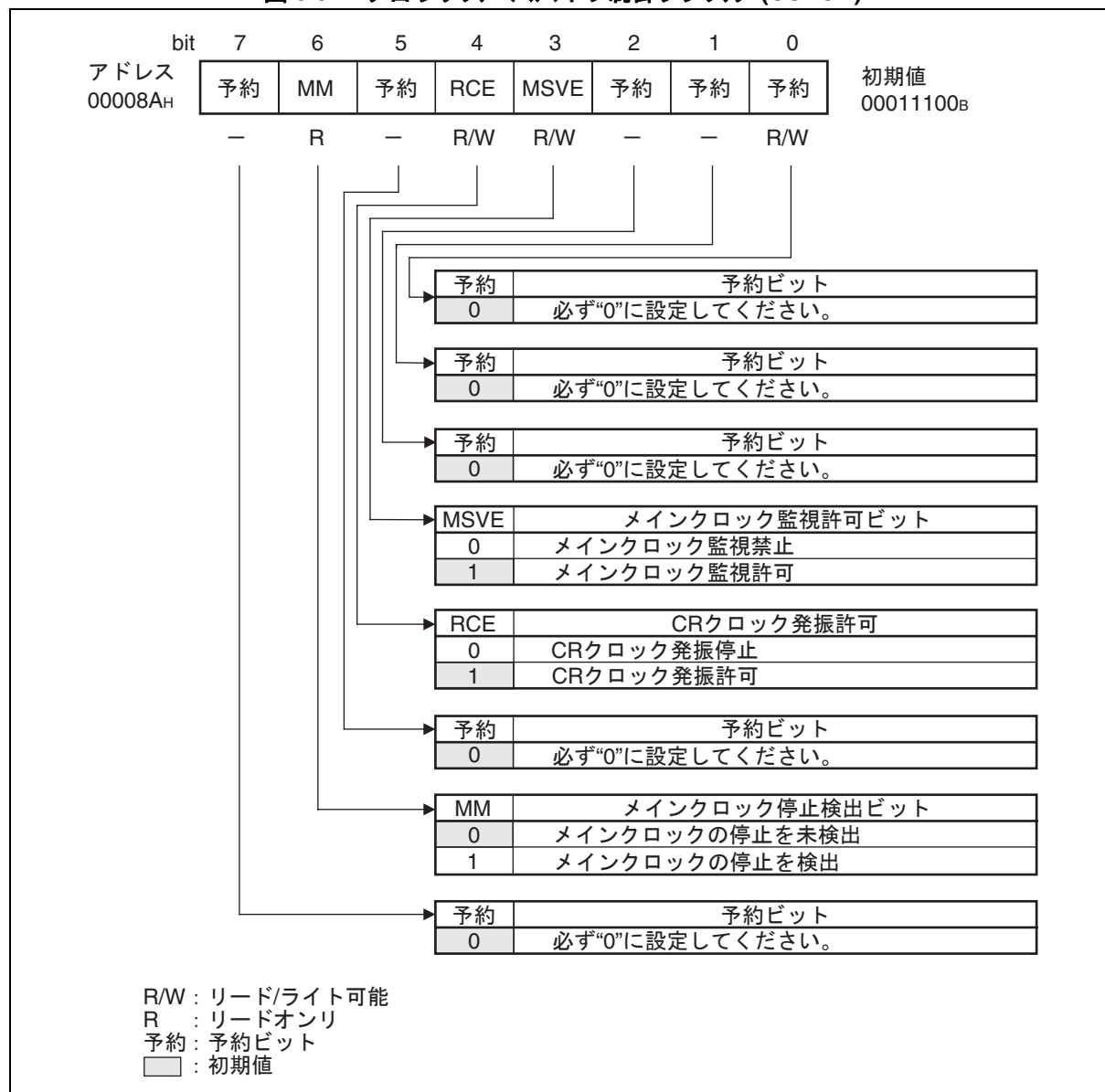


表 5.3-1 クロックスーパーバイザ制御レジスタ (CSVCR) の各ビットの機能説明

ビット名		機 能
bit7	予約ビット	予約ビットです。 このビットへは必ず "0" を書き込んでください。また、読出し値は常に "0" です。
bit6	MM : メインクロック 停止検出ビット	読出し専用のビットでメインクロック発振の停止を検出したことを示すビットです。 "0" に設定された場合：メインクロック発振の停止を検出していない状態。 "1" に設定された場合：メインクロック発振の停止を検出した状態。 このビットへの "1" の書込みは動作に影響を与えません。
bit5	予約ビット	予約ビットです。 このビットへは必ず "0" を書き込んでください。また、読出し値は常に "0" です。
bit4	RCE: CR 発振許可 ビット	CR 発振を許可するビットです。 "0" に設定した場合：発振禁止 "1" に設定した場合：発振許可 このビットに "0" を書き込む際は、あらかじめクロック監視機能が禁止となっており、 かつ MM, SM ビットが "0" であることを確認してください。
bit3	MSVE: メインクロック 監視許可ビット	メインクロック発振の監視を許可するビットです。 "0" に設定した場合：メインクロック監視の禁止。 "1" に設定した場合：メインクロック監視の許可。 このビットはパワーオンリセットでのみ "1" に設定されます。
bit2 ~ bit0	予約ビット	予約ビットです。 このビットへは必ず "0" を書き込んでください。また、読出し値は常に "0" です。

< 注意事項 >

電源投入時、クロックスーパーバイザはメインクロックの発振安定待ち時間を経過した後、監視を開始します。したがって、メインクロックの発振安定待ち時間以上動作していないとクロックスーパーバイザは動作しません。

5.4 クロックスーパーバイザの動作説明

クロックスーパーバイザの動作を説明します。

■ クロックスーパーバイザの動作

クロックスーパーバイザはメインクロックの発振を監視し、一定期間のクロック停止を検出すると、動作クロックを CR クロックに切り換え、リセットを発生します。

以下に、各クロックモード時の動作について説明します。

● メインクロックモード時のメインクロック発振停止

メインクロックモード中にメインクロックの発振が停止と判断される条件は、メインクロックの立上りが CR クロックで 4 サイクルの期間検出されなかった場合です。

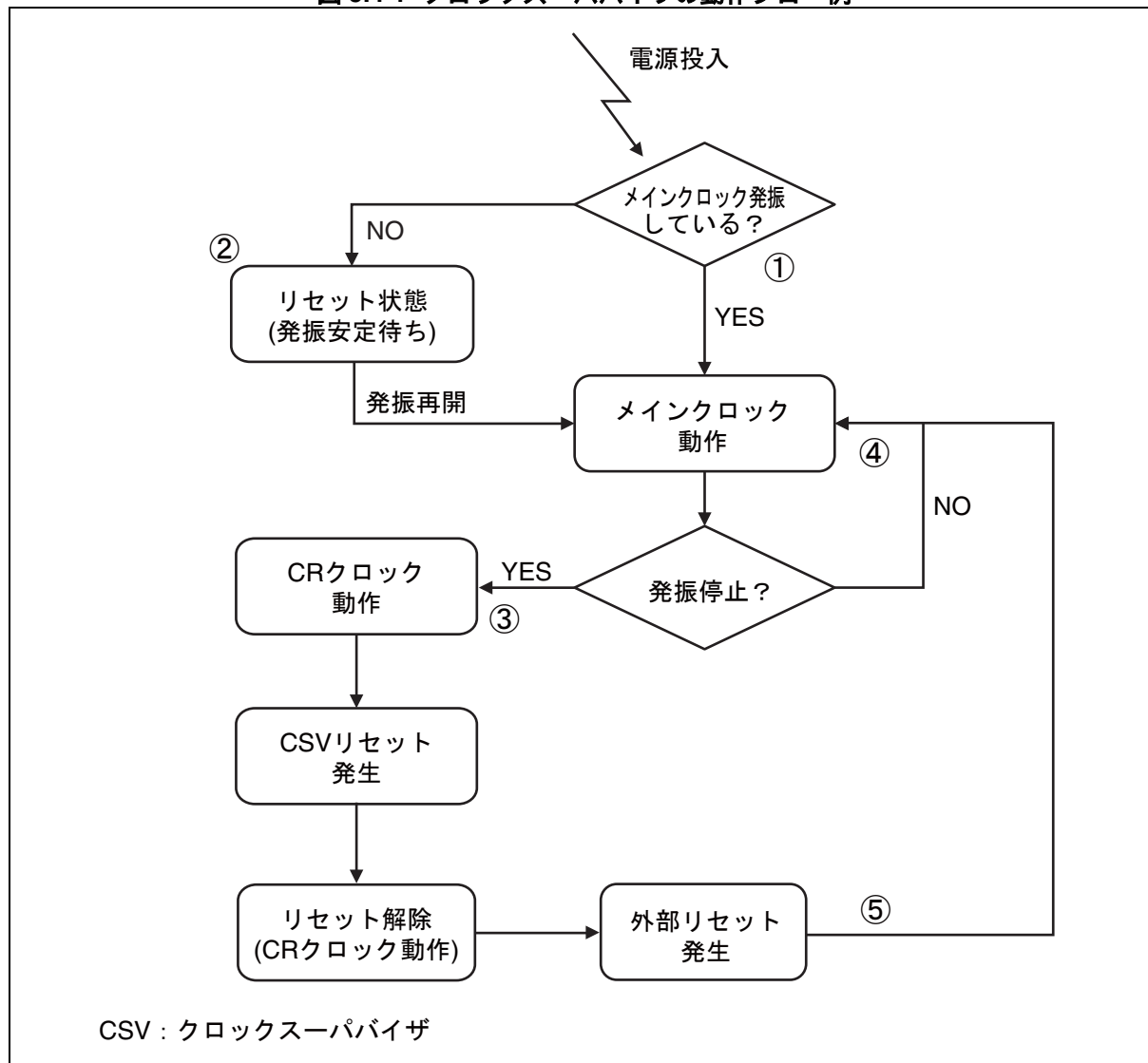
メインクロックの停止を検出するとリセットを発生し、メインクロックは CR クロックに切り換わります。

メインクロックの発振停止の検出を CR クロックで行っているため、メインクロックが低速 (CR クロックの 4 サイクル以上) になるとメインクロックの停止を検出してしまう可能性があります。

また、ストップモード時はメインクロックの検出は行いません。

■ クロックスーパーバイザの動作フロー例

図 5.4-1 クロックスーパーバイザの動作フロー例



電源投入後は、メインクロックの発振による発振安定待ち時間が経過すると、メインクロック動作を開始します。

電源投入時にメインクロックが停止している場合は、リセット状態（発振安定待ち状態）で待機します。さらに、発振が再開して発振安定待ち時間が経過後すると、メインクロック動作に遷移します。

メインクロック動作中に発振停止を検出した場合は、動作クロックを CR クロックに切り換え、リセットを発生します。

メイン発振が継続している（発振が停止していない）場合は、メインクロック動作を継続します。

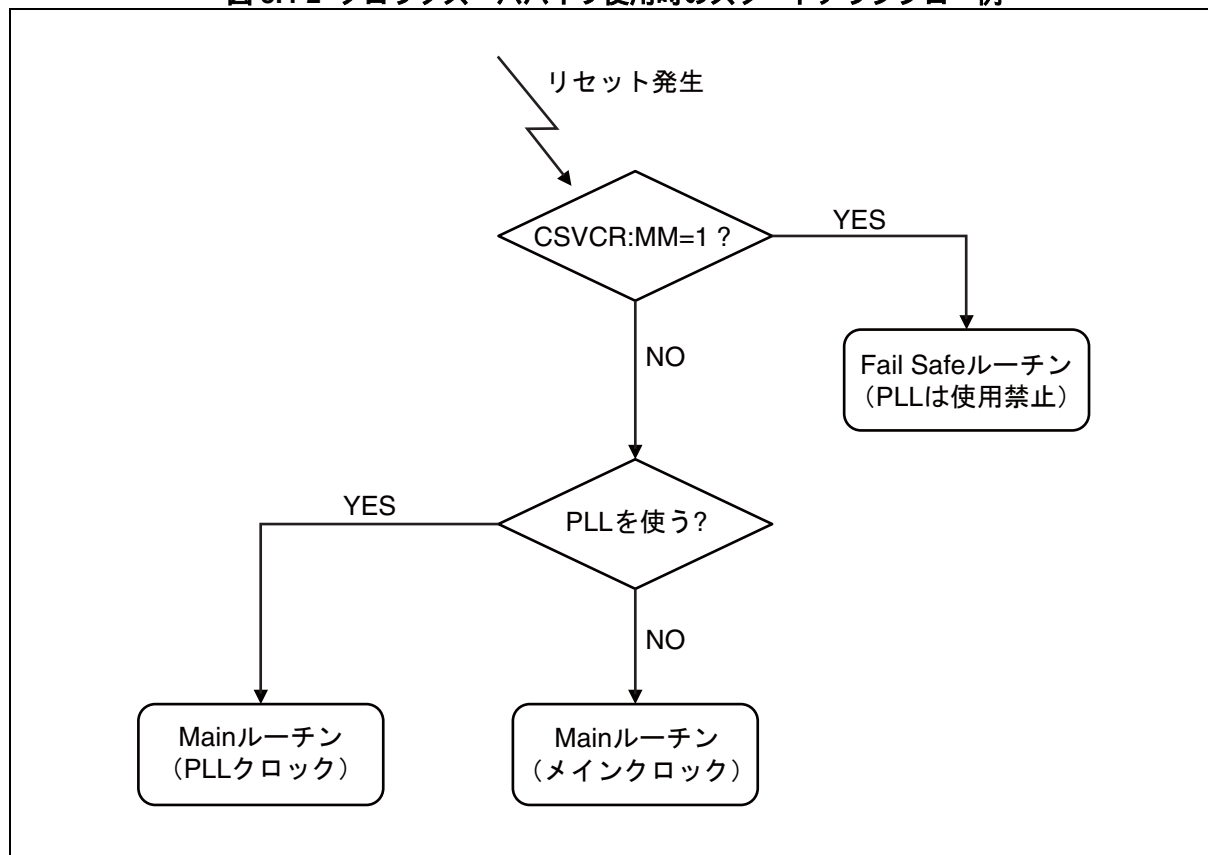
CRクロック動作中に外部リセットが発生した場合は、メインクロック動作に遷移します。ただし、このときに発振が停止していると、再度 CSV リセットを発生して CR クロック動作に戻ります。

■ クロックスーパーバイザ使用時のスタートアップフロー例

ユーザプログラムの先頭に、メインクロック停止検出ビット (CSVCR: MM) の判定を入れることにより、ユーザプログラムでの Fail Safe ルーチン制御が可能となります。

図 5.4-2 に、クロックスーパーバイザ使用時のスタートアップフロー例を示します。

図 5.4-2 クロックスーパーバイザ使用時のスタートアップフロー例



5.5 クロックスーパーバイザ使用上の注意

クロックスーパーバイザを使用する際の注意事項を説明します。

■ クロックスーパーバイザ使用上の注意

クロックスーパーバイザを使用する場合は、以下の点にご注意ください。

- 電源投入時のクロックスーパーバイザの動作について
電源投入後、クロックスーパーバイザはメインクロックの発振安定待ち時間が経過すると監視を開始します。したがって、メインクロックの発振安定待ち時間以上動作していないとクロックスーパーバイザは動作しません。
- CR クロックモードの遷移後について
CR クロックモード遷移後、PLL を "ON" にしないでください。
PLL 回路の入力周波数の下限値を下回るため、PLL の動作は保証されません。
- CR 発振禁止の設定について
CR クロックモード中では、CR 発振許可ビット (CSVCR: RCE) による CR 発振禁止の設定はしないでください。
内部クロックが停止するため、デッドロックを発生するおそれがあります。
- メインクロック停止検出ビットの初期化について
メインクロック停止検出ビット (CSVCR: MM) は、パワーオンリセットまたは外部リセットのみで初期化されます。
ウォッチドッグリセット / ソフトウェアリセット / CSV リセットでは初期化されません。そのため、CR クロックモードでこれらのリセットが発生しても、CR クロックモードを継続します。
- クロックスーパーバイザ機能によるリセット実行確認
クロック監視機能によってリセットが実行されたかを知るためには、ソフトウェアで、WDTC レジスタをリードしてリセット要因をチェックします。ERST (WDTC の bit4) がセットされている場合、外部端子からのリセットもしくはクロックスーパーバイザリセットの発生が確認できます。MM bit (CSVCR の bit6) が "0" であるならリセット要因は外部リセットによるものです。MM が "1" のときはメインクロックが喪失したことによるものです。

第6章

低消費電力モード

低消費電力モードの機能と動作について説明します。

- 6.1 低消費電力モードの概要
- 6.2 低消費電力制御回路のブロックダイアグラム
- 6.3 低消費電力モード制御レジスタ (LPMCR)
- 6.4 CPU 間欠動作モード
- 6.5 スタンバイモード
- 6.6 状態遷移図
- 6.7 スタンバイモード, リセット時の端子状態
- 6.8 低消費電力モード使用上の注意

6.1 低消費電力モードの概要

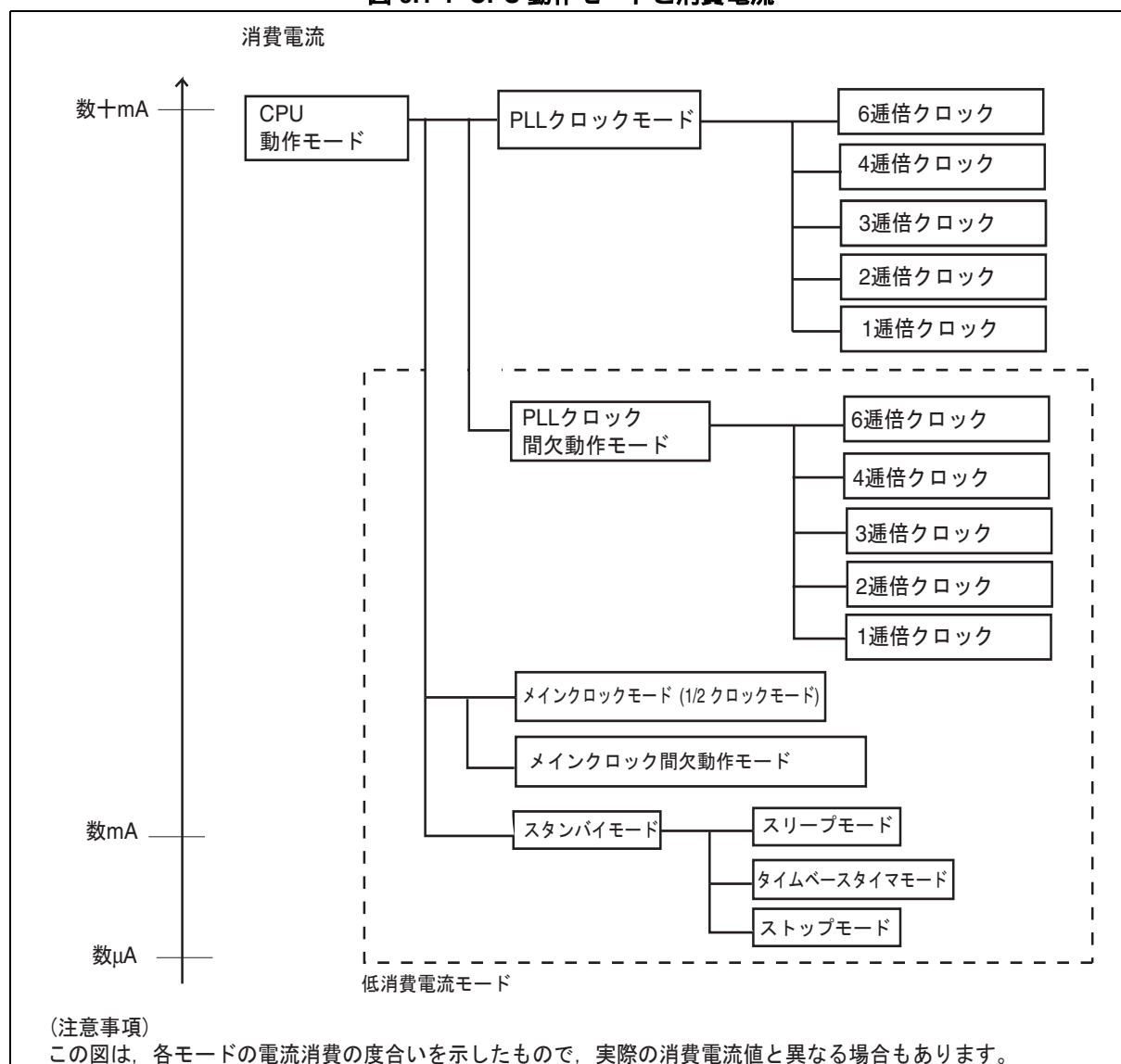
MB90820B シリーズでは、動作クロックの選択とクロックの動作制御による、次のような CPU 動作モードを備えています。

- クロックモード (PLL クロック / メインクロックモード)
 - CPU 間欠動作モード (PLL クロック間欠動作 / メインクロック間欠動作モード)
 - スタンバイモード (スリープ、タイムベースタイマおよびストップの各モード)
- PLL クロックモード以外は、すべて低消費電力モードになります。

■ CPU 動作モードおよび消費電流

CPU 動作モードと消費電流の関係を図 6.1-1 に示します。

図 6.1-1 CPU 動作モードと消費電流



■ クロックモード

● PLL クロックモード

PLL クロックは、発振クロック (HCLK) の通倍クロックで、CPU およびリソースを動作させます。

● メインクロックモード

メインクロックは、発振クロック (HCLK) の 2 分周クロックで、CPU およびリソースを動作させます。メインクロックモードでは、PLL 通倍回路は停止します。

クロックモードについては、「4.1 クロック」を参照してください。

■ CPU 間欠動作モード

CPU 間欠動作モード時は、CPU が間欠的に動作すると同時に、高速クロックパルスがリソースへ供給され、消費電力が抑えられます。CPU 間欠動作モード時、間欠クロックパルスが CPU に供給されるのは、レジスタ、内部メモリ、リソース、または外部ユニットにアクセスするときのみです。

■ スタンバイモード

スタンバイモードにおける低消費電力制御回路は、CPU へのクロック供給を停止（スリープモード）、または CPU およびリソースへのクロック供給を停止（タイムベースタイマモード）、または発振クロックを完全に停止（ストップモード）することにより消費電力を抑えます。

● PLL スリープモード

PLL スリープモードが起動され、PLL クロックモードになると CPU の動作クロックが停止します。ほかのリソースは PLL クロックで動作を継続します。

● メインスリープモード

メインスリープモードが起動され、メインクロックモードになると CPU の動作クロックが停止します。ほかのリソースはメインクロックで動作を継続します。

● PLL タイムベースタイマモード

PLL タイムベースタイマモード時は、発振クロック、PLL クロックおよびタイムベースタイマ以外の動作が停止します。タイムベースタイマ以外のすべての機能は停止します。

● メインタイムベースタイマモード

メインタイムベースタイマモード時は、発振クロック、メインクロックおよびタイムベースタイマ以外の動作が停止します。タイムベースタイマ以外のすべての機能は停止します。

● ストップモード

ストップモード時は、発振クロック (HCLK) が停止し、すべての機能が停止します。

< 注意事項 >

ストップモード時は原発振が停止するため、最も低い消費電力でデータを保持できます。

6.2 低消費電力制御回路のブロックダイアグラム

- CPU 間欠動作セレクト
- スタンバイクロック制御回路
- CPU クロック制御回路
- 周辺クロック制御回路
- 端子ハイインピーダンス制御回路
- 内部リセット発生回路
- 低消費電力モード制御レジスタ (LPMCR)

図 6.2-1 に低消費電力制御回路のブロックダイアグラムを示します。

Figure 6-2 is a detailed block diagram of the internal clock control circuit. It illustrates the path from external clock inputs (X0, X1) and an internal CR oscillator through a clock divider and selector to the main clock. The diagram also shows the control logic for the low-power mode (LPMCR) and the clock selection register (CKSCR).

Low-Power Mode Control Register (LPMCR): This register controls various power-saving features. It includes fields for STP, SLP, SPL, RST, TMDX, CG1, CG0, and RESV. These fields are connected to various control blocks, including the terminal high-impedance control, internal reset, and CPU clock control.

Clock Selection Register (CKSCR): This register selects the clock source for the system. It includes fields for RESV, MCM, WS1, WS0, RESV, MCS, CS1, CS0, and CS2. These fields are connected to the clock selector and the PLL clock control register.

PLL Clock Control Register (PCKCR): This register controls the PLL clock. It includes fields for RESV, MCM, WS1, WS0, RESV, MCS, CS1, CS0, and CS2. These fields are connected to the PLL clock control logic.

System Clock Generation: The system clock is generated by the PLL clock control logic, which takes inputs from the clock selector and the PLL clock control register. The output of the PLL is connected to the clock selector.

Internal Clock Control: The internal clock control logic includes the clock selector, the clock divider, and the clock output. The clock selector selects between the external clock inputs and the internal CR oscillator. The clock divider divides the selected clock by a factor of 2, 4, or 8. The clock output is connected to the main clock.

Other Components: The diagram also shows the terminal high-impedance control, internal reset, and CPU clock control. These components are connected to the LPMCR and CKSCR registers.

- CPU 間欠動作セレクト

CPU 間欠動作モード時に動作中の CPU 動作クロックの一時停止サイクル数を設定します。

- スタンバイ制御回路

CPU クロック制御回路および周辺クロック制御回路を制御し、低消費電力モードへの移行と解除を行います。

- CPU クロック制御回路

CPU に供給するクロックを制御する回路です。この回路は、周辺クロック制御のリソースに供給されるクロックを制御します。

- 周辺クロック制御回路

周辺機能に動作クロックを供給する回路です。

- 端子ハイインピーダンス制御回路

- ・ タイムベースタイマモードおよびストップモードにおいて、外部端子をハイインピーダンスにする回路です。
- ・ プルアップオプションの選択された端子は、ストップモードではプルアップ抵抗を切離します。

- 内部リセット発生回路

内部リセット信号を発生させる回路です。

- 低消費電力モード制御レジスタ (LPMCR)

スタンバイモードへの遷移と解除、また CPU 間欠動作機能の設定などを行うレジスタです。

6.3 低消費電力モード制御レジスタ (LPMCR)

低消費電力モード制御レジスタ (LPMCR) は、低消費電力モードへの遷移 / 解除、および CPU 間欠動作モードでの CPU クロック一時停止サイクル数を設定します。

■ 低消費電力モード制御レジスタ (LPMCR)

図 6.3-1 に低消費電力モード制御レジスタ (LPMCR) の構成を示します。

図 6.3-1 低消費電力モード制御レジスタ (LPMCR) の構成

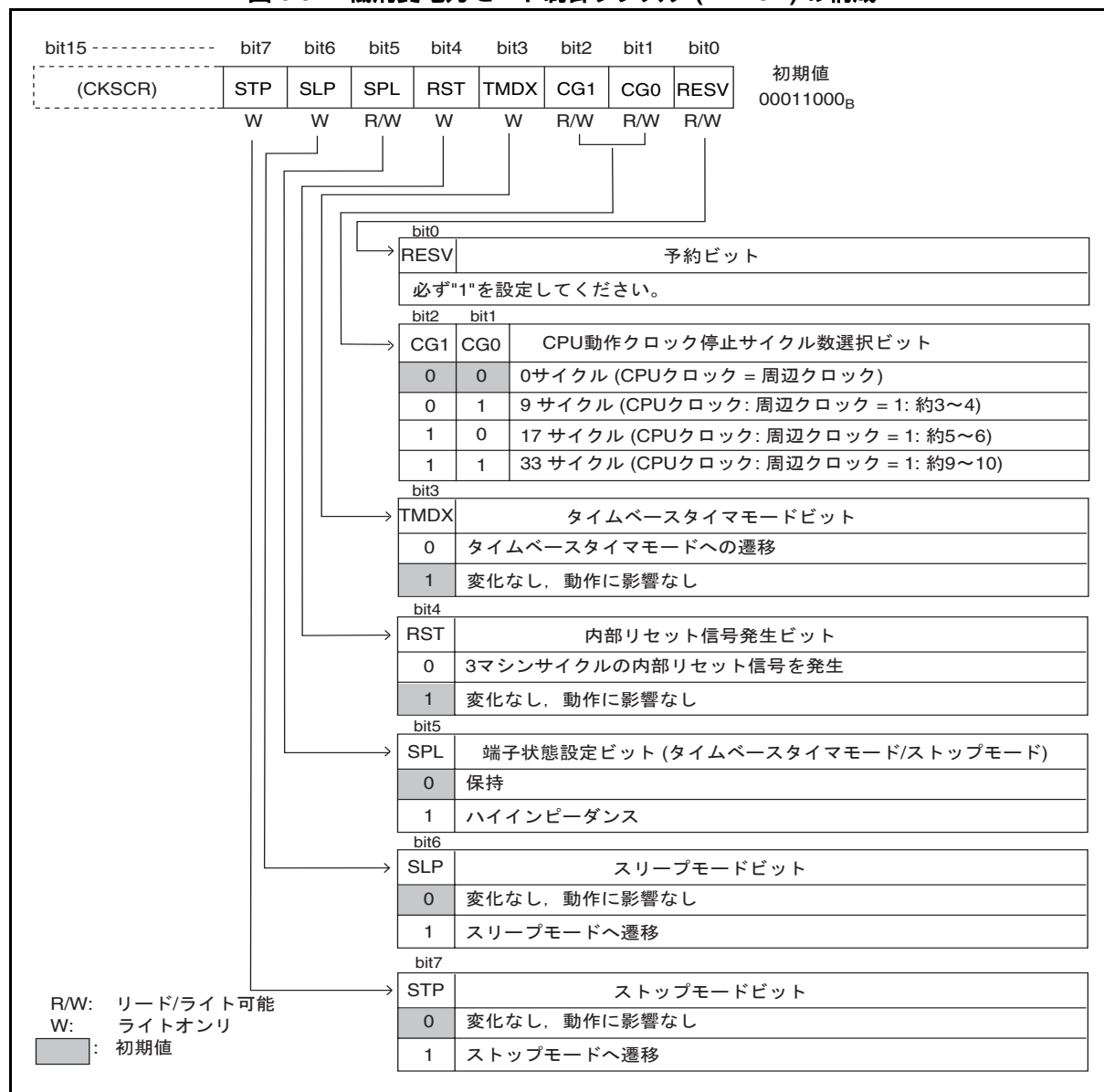


表 6.3-1 低消費電力モード制御レジスタ (LPMCR) の機能

ビット名		機能
bit7	STP: ストップモードビット	<ul style="list-style-type: none"> ストップモードへの遷移を設定します。 "1" を書き込んだ場合：ストップモードに遷移します。 "0" を書き込んだ場合：動作に影響しません。 リセットまたはストップモード解除によって "0" にクリアされます。 読出し値は、常に "0" です。
bit6	SLP: スリープモードビット	<ul style="list-style-type: none"> スリープモードへの遷移を設定します。 "1" を書き込んだ場合：スリープモードに遷移します。 "0" を書き込んだ場合：動作に影響しません。 リセットまたはスリープ解除で "0" にクリアされます。 読出し値は、常に "0" です。
bit5	SPL: 端子状態指定ビット (タイムベースタイマモード / ストップモード)	<ul style="list-style-type: none"> このビットはタイムベースタイマモードまたはストップモードの場合だけ有効です。 "0" 設定の場合：外部端子のレベルを保持します。 "1" 設定の場合：外部端子をハインピーダンスにします。 リセットで "0" に初期化されます。
bit4	RST: 内部リセット信号発生ビット	<ul style="list-style-type: none"> "0" を書き込んだ場合：3 マシンサイクルの内部リセット信号を発生します。 "1" を書き込んだ場合：動作に影響しません。 読出し値は、常に "1" です。
bit3	TMDX: タイムベースタイマモード ビット	<ul style="list-style-type: none"> タイムベースタイマモードへの遷移を指示するビットです。 "0" を書き込んだ場合：タイムベースタイマモードに遷移します。 "1" を書き込んだ場合：動作に影響しません。 リセットまたはタイムベースタイマモードの解除で "1" に設定されます。 読出し値は、常に "1" です。
bit2, bit1	CG1, CG0: CPU 一時停止クロック数選択 ビット	<ul style="list-style-type: none"> CPU 間欠動作での CPU 動作クロックの一時停止サイクル数を設定します。 一命令ごとに指定サイクル数 CPU クロック供給を停止します。 4 種類のクロック数から選択できます。 電源投入、ウォッチドッグタイマリセットによって、"00_B" に初期化されます。ほかのリセットでは初期化できません。
bit0	RESV: 予約ビット	<ul style="list-style-type: none"> このビットには必ず "1" を書き込んでください。

STP ビットと SLP ビットに "1" を書き込み、TMDX ビットに "0" が同時に書き込まれた場合、ストップモードへの遷移が最優先され、タイムベースタイマモードとスリープモードは優先順位が最下位になります。

■ 低消費電力モード制御レジスタへのアクセス

低消費電力モード制御レジスタへの書込みで、低消費電力モード（ストップモード、スリープモード、タイムベースタイマモード、時計モード）へ遷移しますが、低消費電力モードへの遷移に使用する命令は、表 6.3-2 の命令を使用してください。

表 6.3-2 の命令による低消費電力モード遷移命令の直後には必ず下記「 」内の命令列を配置してください。

MOV LPMCR,#H'xx	; 表 6.3-2 の低消費電力モード遷移命令
NOP	
NOP	
JMP \$+3	; 次の命令へのジャンプ
MOV A,#H'10	; 任意の命令

「 」内の命令以外が配置されると低消費電力モード解除後の動作は保証されません。

C 言語を使用して低消費電力モード制御レジスタにアクセスする場合は、「6.8 低消費電力モード使用上の注意」の「■ スタンバイモード遷移のための低消費電力モード制御レジスタ (LPMCR) へアクセスする際の注意事項」を参照してください。

ワード長で低消費電力モード制御レジスタ (LPMCR) へ書き込む場合には、偶数アドレスで書き込むようにしてください。奇数アドレスの書込みで低消費電力モードへ遷移した場合には、誤動作の原因となる場合があります。

表 6.3-2 に示す以外の機能を制御する場合には、どの命令を使用してもかまいません。

表 6.3-2 低消費電力モードへ遷移する場合に使用する命令一覧

MOV io, #imm8	MOV dir, #imm8	MOV eam, #imm8	MOV eam, Ri
MOV io, A	MOV dir, A	MOV addr16, A	MOV eam, A
MOV @RLi+disp8, A			
MOVW io, #imm16	MOVW dir, #imm16	MOVW eam, #imm16	MOVW eam, RWi
MOVW io, A	MOVW dir, A	MOVW addr16, A	MOVW eam, A
MOVW @RLi+disp8, A			
SETB io:bp	SETB dir:bp	SETB addr16:bp	
CLRB io:bp	CLRB dir:bp	CLRB addr16:bp	

6.4 CPU 間欠動作モード

CPU 間欠動作モードは、外部バスや周辺機能を高速動作させたまま CPU を間欠動作させることで、消費電力を抑えることができます。

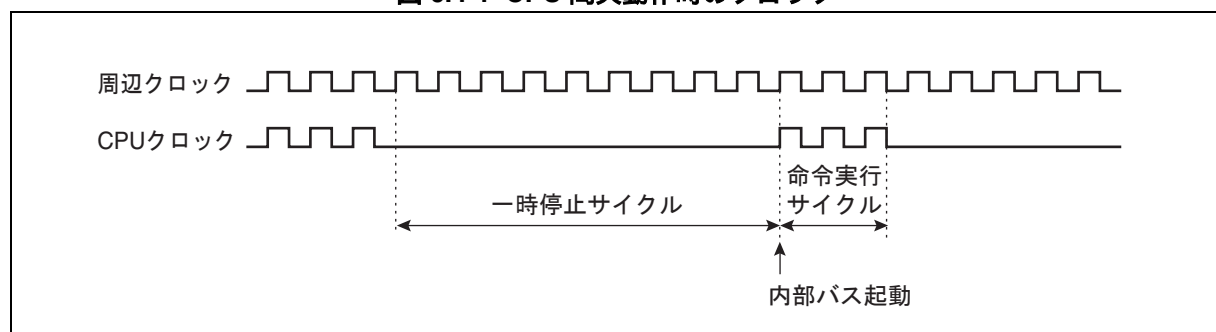
■ CPU 間欠動作モード

CPU 間欠動作モードは、レジスタ、内部メモリ (ROM, RAM), I/O, リソース、および外部バスアクセスを行う場合、CPU に供給するクロックを一命令実行ごとに一定期間停止させ、内部バスサイクルの起動を遅らせるモードです。リソースに高速の周辺クロックを供給したまま、CPU の実行速度を下げると、低消費電力で処理できます。

- 低消費電力モード制御レジスタ (LPMCR) の CG1 と CG0 は、CPU に供給するクロックの一時停止サイクル当りのクロックパルス数を選択します。
- 外部バス動作は、リソースと同じクロックを使用します。
- CPU 間欠モード時の命令実行時間は、次のように算出できます。レジスタ、内部メモリ、リソースおよび外部バスをアクセスする命令が実行された回数に、一時停止サイクル当りのクロックパルス数をかけることによって、補正値が求められます。この補正値を正常実行時間に加算します。

図 6.4-1 に CPU 間欠動作モード時の動作クロックを示します。

図 6.4-1 CPU 間欠動作時のクロック



6.5 スタンバイモード

スタンバイモードには、スリープ (PLL スリープ, メインスリープ), タイムベースタイマ, ストップの各モードがあります。

■ スタンバイモードの動作状態

表 6.5-1 にスタンバイモードの動作状態を示します。

表 6.5-1 スタンバイモードの動作状態

スタンバイモード		遷移条件	発振	クロック	CPU	リソース	端子	解除方法	
スリープ モード	PLL スリープ モード	MCS = 0 SLP = 1	動作	動作	停止	動作	動作	リセット または 割込み	
	メインスリープ モード	MCS = 1 SLP = 1							
タイム ベース タイマ モード	PLL タイム ベースタイマ モード (SPL = 0)	MCS = 0 TMDX = 0				停止 *	保持		
	PLL タイム ベースタイマ モード (SPL = 1)								Hi-Z
	メインタイム ベースタイマ モード (SPL = 0)	MCS = 1 STP = 1							保持
	メインタイム ベースタイマ モード (SPL = 1)								
ストップ モード	メイン /PLL ス トップモード (SPL = 0)	MCS = x STP = 1	停止	停止	停止	保持			
	メイン /PLL ス トップモード (SPL = 1)						Hi-Z		

* : タイムベースタイマのみは動作します。

SPL : 低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット

SLP : 低消費電力モード制御レジスタ (LPMCR) のスリープモードビット

STP : 低消費電力モード制御レジスタ (LPMCR) のストップモードビット

TMDX: 低消費電力モード制御レジスタ (LPMCR) のタイムベースタイマモードビット

MCS : クロック選択レジスタ (CKSCR) のマシナクロック選択ビット

Hi-Z : ハイインピーダンス

MB90820B シリーズ

6.5.1 スリープモード

スリープモードは、CPU の動作クロックを停止させるモードで、CPU 以外は動作を続行します。低消費電力モード制御レジスタ (LPMCR) でスリープモードへの遷移を表示すると、PLL クロックモードを設定している場合は PLL スリープモードへ遷移し、メインクロックモードを設定している場合はメインスリープモードへ遷移します。

■ スリープモードへの遷移

低消費電力モード制御レジスタ (LPMCR) の SLP ビットおよび TMDX ビットに "1" を、STP ビットに "0" を書き込むと、スリープモードに遷移します。このとき、クロック選択レジスタ (CKSCR) の MCS ビットが "0" であれば PLL スリープモードに、MCS ビットが "1" であれば、メインスリープモードに遷移します。

< 注意事項 >

SLP ビットと STP ビットへの "1" の書込みと TMDX ビットへの "0" の書込みが同時に行われると、STP/TMDX ビット設定値は SLP ビット設定値を無効にするので、ストップモードまたはタイムベースタイマモードへの切換えが発生します。

● データ保持機能

スリープモードでは、アキュムレータや内部 RAM などの専用レジスタのデータが保持されます。

● 割込み要求時の動作

割込み要求時に、低消費電力モード制御レジスタ (LPMCR) の SLP ビットに "1" を設定しても、スリープモードへ遷移しません。CPU が割込みを受け付けない場合、CPU は次の命令を実行します。CPU が割込みを受け付けた場合、直ちに割込み処理ルーチンに分岐します。

● 端子状態

スリープモード時は、すべての端子はスリープモードへの切換え直前の状態を保持します。バス入出力またはバス制御に使用される端子は例外です。

■ スリープモードの解除

低消費電力制御回路は、リセット入力または割込みの発生によってスリープモードを解除します。

● リセットによる復帰

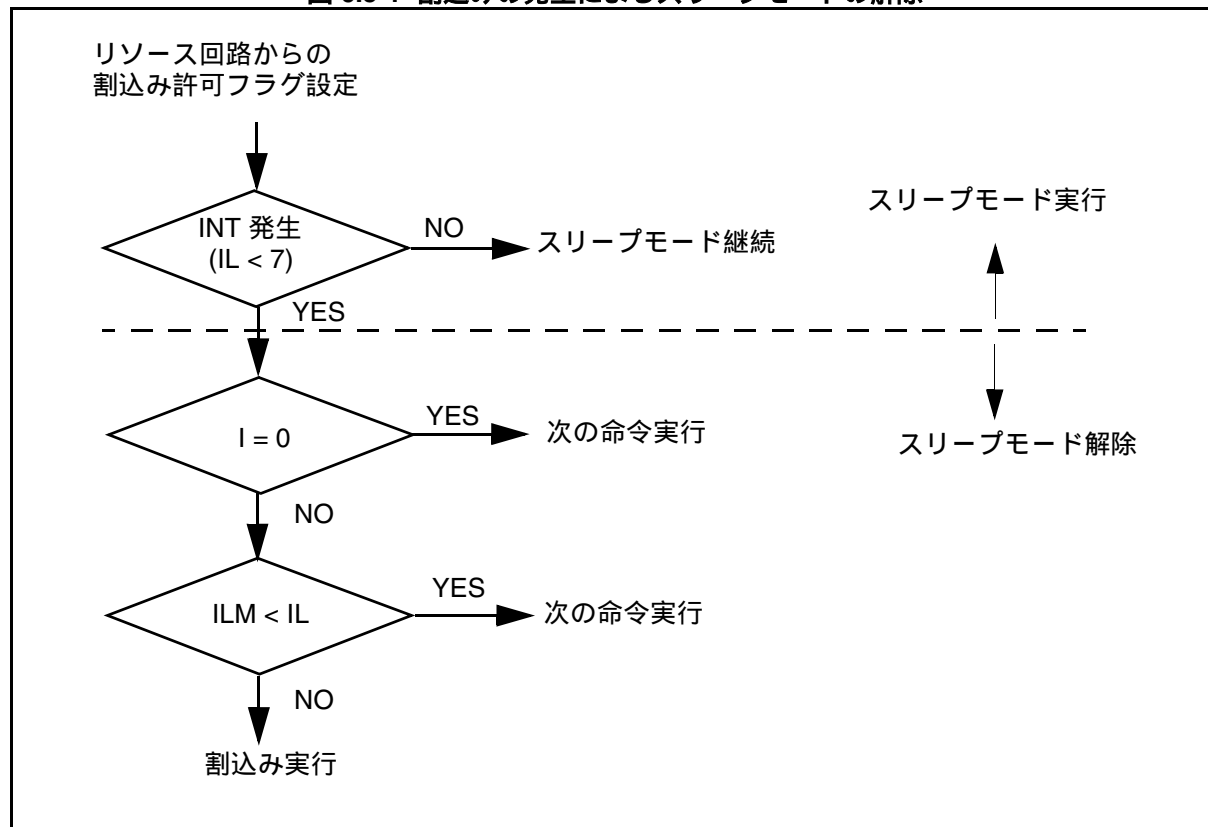
スリープモード時がリセットにより解除されると、スリープモードから解除されるとリセット状態になります。

● 割込みによる復帰

スリープモード時にリソースからレベル7より高い割込み要求が発生されると、スリープモードが解除されます。スリープモードが解除されると、CPU はこの割込みをほかの割込みと同様な方法で処理します。CPU はコンディションコードレジスタ (CCR)、割込みレベルマスクレジスタ (ILM)、および割込み制御レジスタ (ICR) の設定に従って処理を実行します。この割込みが受け付けられた場合は、CPU は割込み処理を実行します。この割込みが受け付けられなかった場合は、CPU はスリープモードへの切換えが指定されている命令のすぐ次の命令を実行します。

割込みによるスリープモードの解除を図 6.5-1 に示します。

図 6.5-1 割込みの発生によるスリープモードの解除



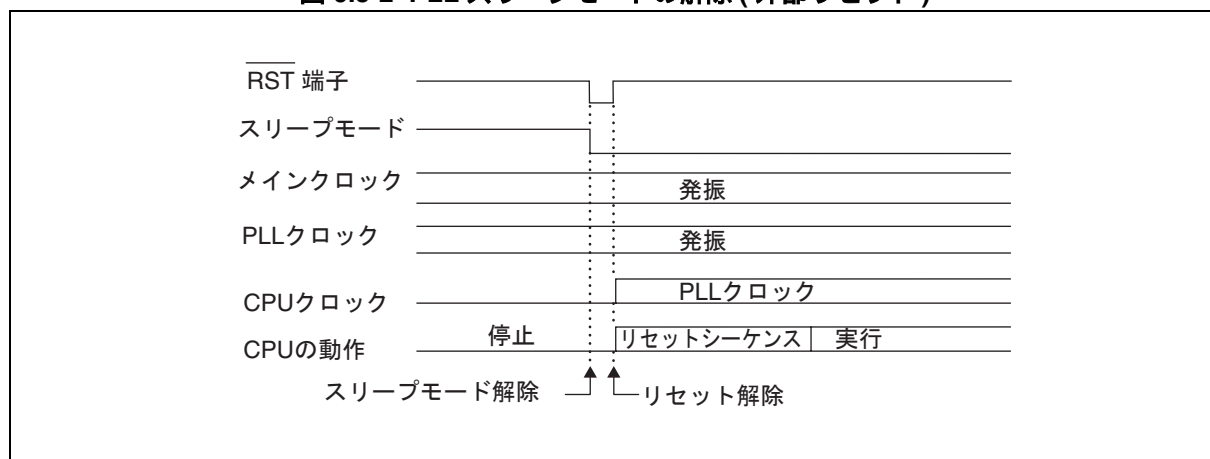
< 注意事項 >

割込み処理が正常に実行されると、CPUはまずスリープモードへの切換えが指定されている命令のすぐ次の命令を実行します。その後 CPU は割込み処理に移ります。

● 外部リセットによる PLL スリープモードからノーマルモードへの復帰

PLL スリープモード時には、メインクロックと PLL クロックがクロックパルスを生成します。外部リセットはクロック選択レジスタ (CKSCR) の MCS ビットを "1" に初期化しないので、PLL ロックモードが選択された状態 (CKSCR の MCS = 0) のままとなります。外部リセットで PLL スリープモードからノーマルモードに復帰すると、PLL スリープモードが解除された後 CPU は直ちに PLL クロックで動作を開始します。図 6.5-2 を参照してください。

図 6.5-2 PLL スリープモードの解除 (外部リセット)



6.5.2 タイムベースタイマモード

タイムベースタイマモードは、原発振とタイムベースタイマ以外の動作を停止させるモードで、タイムベースタイマ以外すべての機能が停止します。

■ タイムベースタイマモードへの遷移

- 低消費電力モード制御レジスタ (LPMCR) の TMDX ビットと STP ビットに "0" を書き込むと、タイムベースタイマモードへの切換えが発生します。
- このとき、クロック選択レジスタ (CKSCR) の MCS ビットに "0" が設定されている場合、PLL タイムベースタイマモードになり、クロック選択レジスタ (CKSCR) の MCS ビットに "1" が設定されている場合は、メインタイムベースタイマモードになります。

< 注意事項 >

TMDX ビットと STP ビットへの "0" の書込みが同時に行われると、STP ビット設定値は TMDX ビットの設定値を無効にするので、ストップモードへの切換えが発生します。

● データ保持機能

タイムベースタイマモード時は、アキュムレータや内部 RAM などの専用レジスタのデータを保持します。

● 割込み要求時の動作

割込み要求時に LPMCR の TMDX ビットに "0" を書き込んでも、タイムベースタイマモードへの切換えは発生しません。

● 端子状態

タイムベースタイマモードへの切換えが発生する直前の状態を外部端子に保持させるか、またはこのモードへの切換えによって外部端子をハイインピーダンスに設定するかを選択に関する制御は、LPMCR の SPL ビットで行うことができます。

■ タイムベースタイマモードの解除

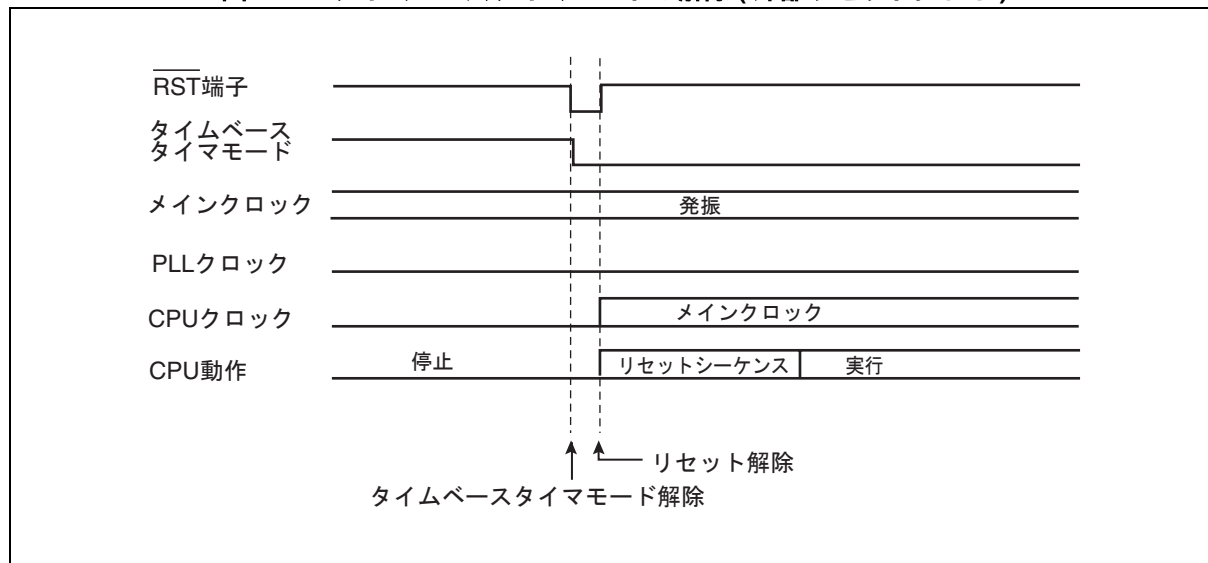
低消費電力制御回路は、リセット入力または割込みでタイムベースタイマモードを解除します。タイムベースタイマモードがリセットによって解除されると、本マイコンはタイムベースタイマモードから解除された後でリセット状態になります。

● リセットによるノーマルモードへの復帰

タイムベースタイマモードがリセットによって解除されると、本マイコンはタイムベースタイマモードから解除された後でリセット状態になります。タイムベースタイマは、リセットによりメインクロックモードへと初期化されます。

外部リセットによるタイムベースタイマモードからノーマルモードへの復帰動作を図 6.5-3 に示します。

図 6.5-3 タイムベースタイマモードの解除 (外部リセットによる)



● 割込みによるノーマルモードへの復帰

タイムベースタイマモード中にリソース回路などから割込みレベルが 7 より高い割込み要求が発生すると (割込み制御レジスタ ICR : IL2, IL1, IL0 が "111B" 以外), 低消費電力制御回路はタイムベースタイマモードを解除します。タイムベースタイマモードの解除後は, 通常の割込み処理と同じ扱いとなります。CPU は, コンディションコードレジスタ (CCR) の I フラグ, 割込みレベルマスクレジスタ (ILM), および割込み制御レジスタ (ICR) の設定に従って処理を実行します。この割込みが受け付けられる場合は, 割込み処理を実行します。割込みが受け付けられない場合は, タイムベースタイマモードへの切換えを指定する命令の次の命令から処理を実行します。

< 注意事項 >

割込み処理が正常に実行されると, CPUはまずスリープモードへの切換えが指定されている命令のすぐ次の命令を実行します。その後 CPU は割込み処理に移ります。

6.5.3 ストップモード

ストップモードでは、すべての発振を停止させ、全機能が停止します。したがって、最も少ない消費電力でデータを保持できます。

■ ストップモードへの移行

低消費電力モード制御レジスタ (LPMCR) の STP ビットに "1" を書き込むと、ストップモードに移行します。

この時点でクロック選択レジスタ (CKSCR) の MCS ビットに "0" が設定されている場合は、PLL ストップモードになり、クロック選択レジスタ (CKSCR) の MCS ビットに "1" が設定されている場合は、メインストップモードになります。

- データ保持機能

ストップモードでは、アキュムレータなどの専用レジスタと内部 RAM の内容が保持されます。

- 割込み要求が発生している場合の動作

割込み要求が発生している最中に低消費電力モード制御レジスタ (LPMCR) の STP ビットに "1" を設定した場合は、ストップモードに移行しません。

- 端子状態の設定

外部端子がストップモード遷移する直前の状態を保持するか、ハイインピーダンス状態になるかは、低消費電力モード制御レジスタ (LPMCR) の SPL ビットで制御できます。

■ ストップモードの解除

低消費電力制御回路は、リセット入力、または割込みの発生によりストップモードを解除します。ストップモードから復帰する前は動作クロックの発振が停止しているため、低消費電力制御回路はまず発振安定待ち状態に移行してから、ストップモードを解除します。

- リセットによる復帰

リセット要因によるストップモード解除の場合は、ストップモードを解除した上で、発振安定待ちリセット状態になります。リセットシーケンスは発振安定待ち時間経過後に移ります。

- 割込みによる復帰

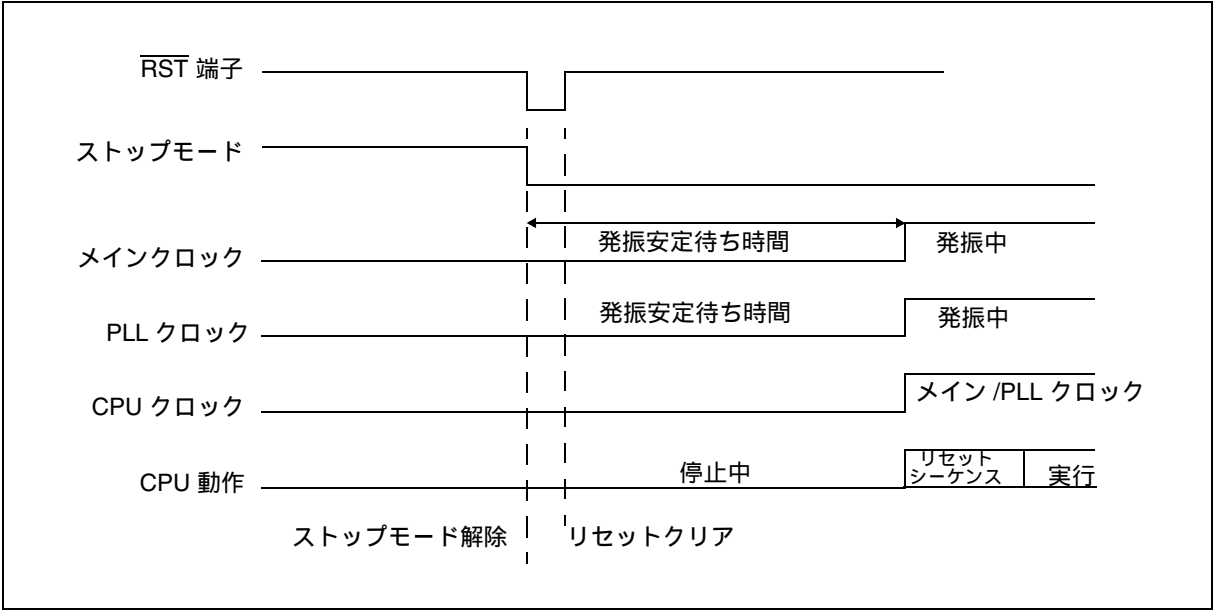
ストップモード中に周辺回路などから割込みレベルが 7 より高い割込み要求が発生すると (割込み制御レジスタ ICR: IL2, IL1, IL0 が "111b" 以外)、低消費電力制御回路はストップモードを解除します。ストップモードの解除後は、CPU はこの割込みをほかの割込みと同様な方法で処理します。ただし、CPU の起動は、クロック選択レジスタ (CKSCR) の WS1, WS0 ビットで指定されたメインクロックの発振安定待ち時間を経過した後で行われます。コンディションコードレジスタ (CCR) の I フラグ、割込みレベルマスクレジスタ (ILM)、および割込み制御レジスタ (ICR) の設定に従って、CPU は処理を実行します。この割込みが受け付けられなかった場合、ストップモードへの切り換えが指定されている命令の次の命令を実行します。

< 注意事項 >

割り込み処理が正常に実行されると, CPUはストップモードへの切換えが指定されている命令の次の命令を実行します。次に CPU は割り込み処理に移行します。

図 6.5-4 にストップモードからの復帰動作を示します。

図 6.5-4 ストップモードの解除 (外部リセット)

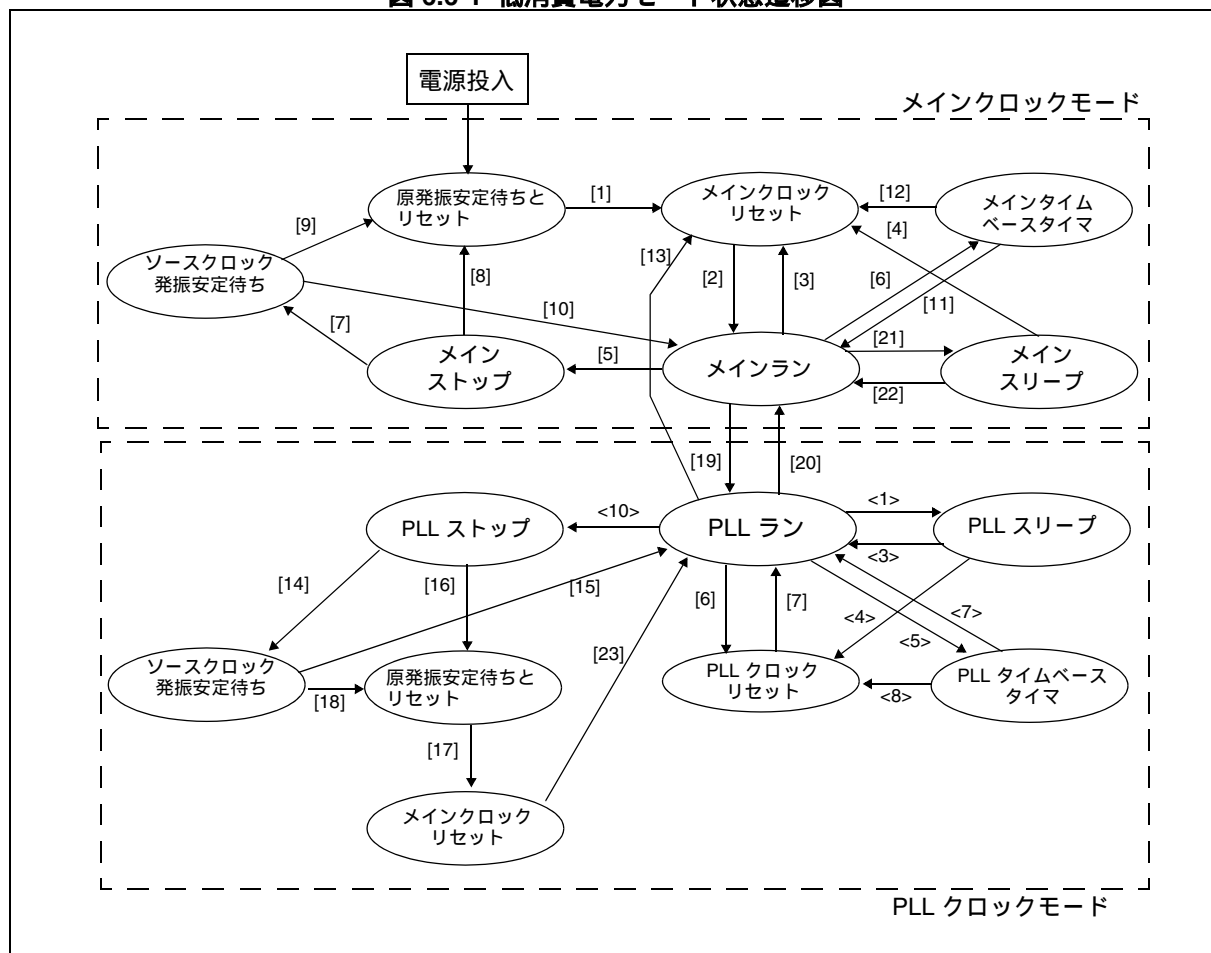


6.6 状態遷移図

MB90820B シリーズの動作状態の遷移図と、遷移条件について示します。

■ 状態遷移図

図 6.6-1 低消費電力モード状態遷移図



■ 低消費電力モードの動作状態

表 6.6-1 に低消費電力モードの動作状態を示します。

表 6.6-1 低消費電力動作モードの動作状態

低消費電力モード	状態遷移発生条件	発振	クロック	CPU	リソース	端子	解除イベント
メインスリープ	MCS = 1 SLP = 1	動作	動作	停止	動作	動作	リセット または 割込み
PLL スリープ	MCS = 0 SLP = 1	動作	動作	停止	動作	動作	リセット または 割込み
メイン /PLL タイムベースタイマ (SPL = 0)	MCS = x TMDX = 0	動作	動作	停止	停止	保持	リセット または 割込み
メイン /PLL タイムベースタイマ (SPL = 1)	MCS = x TMDX = 0	動作	動作	停止	停止	Hi-Z	リセット または 割込み
メイン /PLL ストップ (SPL = 0)	MCS = x STP = 1	停止	停止	停止	停止	保持	リセット または 割込み
メイン /PLL ストップ (SPL = 1)	MCS = x STP = 1	停止	停止	停止	停止	Hi-Z	リセット または 割込み

● クロックモードへの切換えおよび解除 (スタンバイモードは除く)

クロックモードへの切換えおよびクロックモードの解除を表 6.6-2 に示します。

表 6.6-2 クロックモードへの切替えおよびクロックモードの解除

状態遷移	状態遷移発生条件
パワーオン後、メインラン状態への遷移	[1] ソースクロック発振安定化待ち時間が終了します (タイムベースタイマ出力)。 [2] リセット入力解除される。
メインラン状態時のリセット	[3] 外部リセット、ソフトリセット、またはウォッチドッグタイマリセット
メインラン状態から PLL ラン状態への遷移	[19]MCS = 0(PLL クロック発振安定化待ち時間後、PLL クロック動作) *
PLL ラン状態からメインラン状態への復帰	[20]MCS = 1 (PLL クロックが停止します)
PLL ラン状態におけるリセット	[6] 外部リセットまたはソフトリセット ([7]: リセット後に PLL ラン状態に復帰します) [13] ウォッチドッグリセット ([2]: リセット後にメインラン状態に復帰します)

*: PLL クロック発振安定化待ち状態の間は、メインクロックで動作します。

● スタンバイモードへの切換えおよびスタンバイモードの解除

表 6.6-3 にスタンバイモードへの切換えおよびスタンバイモードの解除を示します。

表 6.6-3 スタンバイモードへの切換えおよびスタンバイモードの解除

状態遷移	状態遷移発生条件
メインスリープモードへの遷移	[21] SLP = 1, MCS = 1 (メインラン状態から遷移) [2] SLP = 1, MCS = 1 (PLL ラン状態から遷移)
メインスリープモードの解除	[22] 割込み入力 [4] 外部リセット
メインストップモードへの遷移	[5] STP = 1, MCS = 1 (メインラン状態から遷移)
PLL ストップモードへの遷移	<10> STP = 1, MCS = 0 (PLL ラン状態から遷移)
メインストップモードの解除	[7] 割込み入力 ([10] 発振安定化待ち後にメインラン状態に復帰します) [8] 外部リセット ([9] 発振安定化待ち状態において外部リセットが発生します)
PLL ストップモードの解除	[14] 割込み入力 ([15] 発振安定化待ち後に PLL ラン状態に復帰します) [16] 外部リセット ([18] 発振安定化待ち状態において外部リセットが発生します)
PLL スリープモードへの遷移	<1> SLP = 1, MCS = 0 (PLL ラン状態から遷移) <2> SLP = 1, MCS = 0 (メインラン状態から遷移。PLL クロック発振安定化待ち後に PLL クロックへの切換え)*
PLL スリープモードの解除	<3> 割込み入力 <4> 外部リセット
メインタイムベースタイマモードへの遷移	[6] STP = 1, MCS = 1 (メインラン状態から遷移)
PLL タイムベースタイマモードへの遷移	<5> STP = 1, MCS = 0 (PLL ラン状態から遷移)
メインタイムベースタイマモードの解除	[11] 割込み入力 [12] 外部リセット ([2] リセット後にメイン実行状態に復帰します)
PLL タイムベースタイマモードの解除	<7> 割込み入力 <8> 外部リセット ([7] リセット後に PLL 実行状態に復帰します)

*: PLL クロック発振安定化待ち状態の間は、メインクロックで動作します。

MB90820B シリーズ

6.7 スタンバイモード，リセット時の端子状態

スタンバイモード，リセット時の端子の状態を，各メモリアクセスモードごとに示します。

■ ソフトウェアブルアップ抵抗

ソフトウェアでブルアップ抵抗を選択している端子は，"L" レベル出力時にはブルアップ抵抗は切断されます。

■ シングルチップモード時の端子状態

表 6.7-1 にシングルチップモード時の各端子状態について示します。

表 6.7-1 シングルチップモード時の各端子状態

端子名	スタンバイモード			リセット時
	スリープ モード	ストップモード		
		SPL = 0	SPL = 1	
P00 ~ P07, P17, P20 ~ P27, P30 ~ P37, P40 ~ P47, P50, P60 ~ P63, P70 ~ P77, P80 ~ P87,	直前の状態を 保持 ^{* 2}	直前の状態を 保持 ^{* 2}	入力遮断 ^{* 3} / 出力 Hi-Z ^{* 3}	出力 Hi-Z ^{* 3}
P10 ~ P16, P63		入力可 ^{* 1}		

*1: " 入力可 " とは，対応する外部割込み端子を許可している場合に入力機能が可能な状態であることを意味します。ブルアップまたはブルダウソプションを選択してください。または外部からの入力が必要です。出力ポートとして使用している端子はほかのポートと同じです。

*2: " 直前の状態を保持 " とは，このモードになる直前に出力していた状態をそのまま保持することを意味します。ただし，入力状態であった場合は入力不可となるので注意してください。

• " 出力していた状態をそのまま保持 " とは，出力のある内部リソースが動作中であればその値を，ポートとして出力している場合にはその値をそれぞれ保持することを意味します。

• " 入力不可 " とは，端子からすぐの入力ゲート動作は許可状態にあるが，内部回路が動作していないため，端子への入力値が内部で受け付けられない状態を意味します。

*3: 入力遮断状態では，入力はマスクされ "L" レベルが内部に伝わります。" 出力 Hi-Z " とは端子駆動用トランジスタを駆動禁止状態にして端子をハイインピーダンスにすることを意味します。

6.8 低消費電力モード使用上の注意

低消費電力モードを使用する場合には、以下の点に関して注意が必要です。

- ・スタンバイモードへの移行と割込み
 - ・割込みによるスタンバイモードの解除
 - ・スタンバイモードの設定
 - ・ストップモードの解除
 - ・タイムベースタイマモードの解除
 - ・発振安定待ち時間
 - ・スタンバイモード遷移のための低消費電力モード制御レジスタ (LPMCR) へアクセスする際の注意事項
-

■ スタンバイモードへの移行と割込み

周辺機能から CPU に対しての割込み要求発生中は、低消費電力モード制御レジスタ (LPMCR) のストップモードビット (STP) やスリープモードビット (SLP) に "1" が設定されている場合でも、CPU これらのビットを無視します。従って、スタンバイモードへは移行しません (割込み処理後でも、スタンバイモードへの移行は行われません)。割込みレベルが 7 より高い場合は、この動作は割込み要求が CPU に受け付けられるかどうかには関係しません。

しかし、CPU による割込み処理中であっても、その割込み要求フラグビットがクリアされておりほかに割込み要求がない場合には、スタンバイモードへ移行できます。

■ 割込みによるスタンバイモードの解除

- ・スリープ、タイムベースタイマ、またはストップモード中に、周辺機能などから割込みレベルが 7 より高い割込み要求が発生した場合、スタンバイモードは解除されません。これは CPU が割込みを受け付けるかどうかには関係しません。
- ・スタンバイモード解除後、通常の割込み動作が行われます。割込みレベル設定ビット (ICR : IL2, IL1, IL0) で示された割込み要求優先順位が、割込みレベルマスケジスタ (ILM) より高く、コンディションコードレジスタ (CCR) の割込み許可フラグ (I) が "1" (許可) に設定されている場合は、CPU は割込み処理ルーチンへ分岐します。割込みが受け付けられない場合は、スタンバイモードへの遷移が指定されている命令の次の命令を実行します。
- ・割込み処理が正常に実行されると、CPU は先ずスタンバイモードへの遷移を指定された命令の次の命令を実行し、その後割込み処理に移ります。スタンバイモードに移行する際の条件によって、CPU は次の命令を実行する前に割込み処理に移ることがあります。
- ・スタンバイモードから通常モードへ復帰した直後に CPU が割込み処理ルーチンへ分岐しない場合は、スタンバイモードが設定される前に割込みを禁止する処置を行わなければなりません。

■ スタンバイモードの設定

LPMCR の STP ビットと SLP ビットに同時に "1" を書き込むと、スタンバイモードへの遷移が実行されます。クロック選択レジスタ (CKSCR) の MCS ビットが "0" の場合は、タイムベースタイマモードへの切り換えが実行されます。このビットが "1" の場合、ストップモードへの切り換えが実行されます。

■ ストップモード解除

ストップモード解除のために外部割込みを使用する場合、システムがストップモードに入る前に割込み入力要因として設定された入力を使用してください。入力要因として "H" レベル、"L" レベル、立上りエッジまたは立下りエッジを選択できます。

■ タイムベースタイマモードの解除

- タイムベースタイマモードが解除される場合、PLL クロック発振安定化待ち状態になります。PLL クロックを使用しない場合は、リセット直後または割込みからの復帰時に実行される命令によって、クロック選択レジスタ (CKSCR) の MCS ビットを "1" に書き換えてください。
- 外部割込みを使用してタイムベースタイマモードを解除する場合には、入力要因として "H" レベル、"L" レベル、立上りエッジまたは立下りエッジを選択できます。

■ 発振安定待ち時間

● 発振クロックの発振安定待ち時間

ストップモード時は、原発振用の発振子が停止しているため、発振安定待ち時間をとる必要があります。クロック選択レジスタ (CKSCR) の WS0 と WS1 ビットで選択された時間幅は発振安定化待ち時間として使用されます。

● PLL クロック発振安定待ち時間

- CPU がメインクロックで動作する場合には、PLL クロックは停止することもあります。CPU およびリソースが PLL クロックで動作するモードに移行する場合、PLL クロックは当初発振安定化待ち状態になり、CPU は依然としてメインクロックで動作します。
- PLL クロック発振安定待ち時間は、 $2^{14}/\text{HCLK}$ (HCLK: 発振クロック周波数) 固定です。
- ただし、PLL クロック発振安定化待ち状態になる前にタイムベースタイマがクリアされなかった場合、タイムベースタイマの状態によって待ち時間は $2^{14}/\text{HCLK}$ から $2 \times 2^{14}/\text{HCLK}$ の範囲で変化します (例えば、外部リセットによるタイムベースタイマモードから PLL モードへの復帰が生じます)。

■ スタンバイモード遷移のための低消費電力モード制御レジスタ (LPMCR) へアクセスする際の注意事項

● アセンブラ言語を使用して低消費電力モード制御レジスタ (LPMCR) へアクセスする場合

低消費電力モード制御レジスタ (LPMCR) において低消費電力モードに移行する設定を行う場合は、表 6.3-2 の命令を使用してください。

表 6.3-2 の命令による低消費電力モード遷移命令の直後には必ず下記 [] 内の命令列を配置してください。

```
MOV LPMCR,#H'xx          ; 表 6.3-2 の低消費電力モード遷移命令
[
NOP
NOP
JMP $+3                    ; 次の命令へのジャンプ
MOV A,#H'10                ; 任意の命令
]
```

[] 内の命令以外が配置されると低消費電力モード解除後の動作は保証されません。

● C 言語を使用して低消費電力モード制御レジスタ (LPMCR) にアクセスする場合

低消費電力モード制御レジスタ (LPMCR) においてスタンバイモードに移行する設定を行う場合は、以下の (1) から (3) のいずれかの方法でアクセスしてください。

- (1) スタンバイモードに遷移させる命令を関数化し、スタンバイモード遷移命令の後に `_wait_nop()` のビルトイン関数を 2 個挿入してください。関数内で、スタンバイ復帰の割込みが発生する可能性がある場合は、コンパイル時に最適化を実施し、LINK/UNLINK 命令の発生を抑止してください。

例 (時計モードまたはタイムベースタイマモード遷移関数の場合)

```
void enter_watch(){
    IO_LPMCR.byte = 0x10;      /* LPMCR の TMDX ビットに "0" をセット */
    _wait_nop();
    _wait_nop();
}
```

- (2) スタンバイモードに遷移させる命令を `_asm` 文で記述し、スタンバイモード遷移命令の後に 2 個の NOP と JMP 命令を挿入してください。

例 (スリープモード遷移の場合)

```
_asm(" MOV I:_IO_LPMCR, #H'58"); /* LPMCR の SLP ビットに "1" をセット */
_asm(" NOP");
_asm(" NOP");
_asm(" JMP $+3");                /* 次の命令へのジャンプ */
```

- (3) スタンバイモードに遷移させる命令を `#pragma asm ~ #pragma endasm` 間に記述し、スタンバイモード遷移命令の後に 2 個の NOP と JMP 命令を挿入してください。

例 (ストップモード遷移の場合)

```
#pragma asm
MOV I:_IO_LPMCR, #H'98      /* LPMCR の STP ビットに "1" をセット */
NOP
NOP
JMP $+3                    /* 次の命令へのジャンプ */
#pragma endasm
```

第7章

割込み

MB90820B シリーズの割込みおよび拡張インテリジェント I/O サービス (EI²OS) の機能と動作について説明します。

- 7.1 割込み
- 7.2 割込み要因と割込みベクタ
- 7.3 割込み制御レジスタとリソース
- 7.4 ハードウェア割込み
- 7.5 ソフトウェア割込み
- 7.6 拡張インテリジェント I/O サービス (EI²OS) の割込み
- 7.7 例外処理割込み
- 7.8 割込み処理のスタック動作

7.1 割込み

MB90820B シリーズの割込みおよび拡張インテリジェント I/O サービス (EI²OS) の機能と動作について説明します。

- ・ハードウェア割込み
 - ・ソフトウェア割込み
 - ・拡張インテリジェント I/O サービス (EI²OS) による割込み
 - ・例外処理
-

■ 割込みの種類と機能

● ハードウェア割込み

ハードウェア割込みは、リソースからの割込み要求に対して、ユーザ定義の割込み処理プログラムへ制御を移します。

● ソフトウェア割込み

ソフトウェア割込みは、専用ソフトウェア割込み命令 (INT 命令など) の実行により起動されたユーザ定義の割込み処理プログラムへ制御を移します。

● 拡張インテリジェント I/O サービス (EI²OS) による割込み

- ・拡張インテリジェント I/O サービス (EI²OS) は、リソースとメモリ間の自動データ転送機能です。通常割込み処理プログラムで行っていたデータ転送をダイレクトメモリアクセス (DMA) のように行うことができます。指定回数のデータ転送処理が終了すると、自動的に割込み処理プログラムを実行します。
- ・EI²OS による割込みは、ハードウェア割込みの一種です。

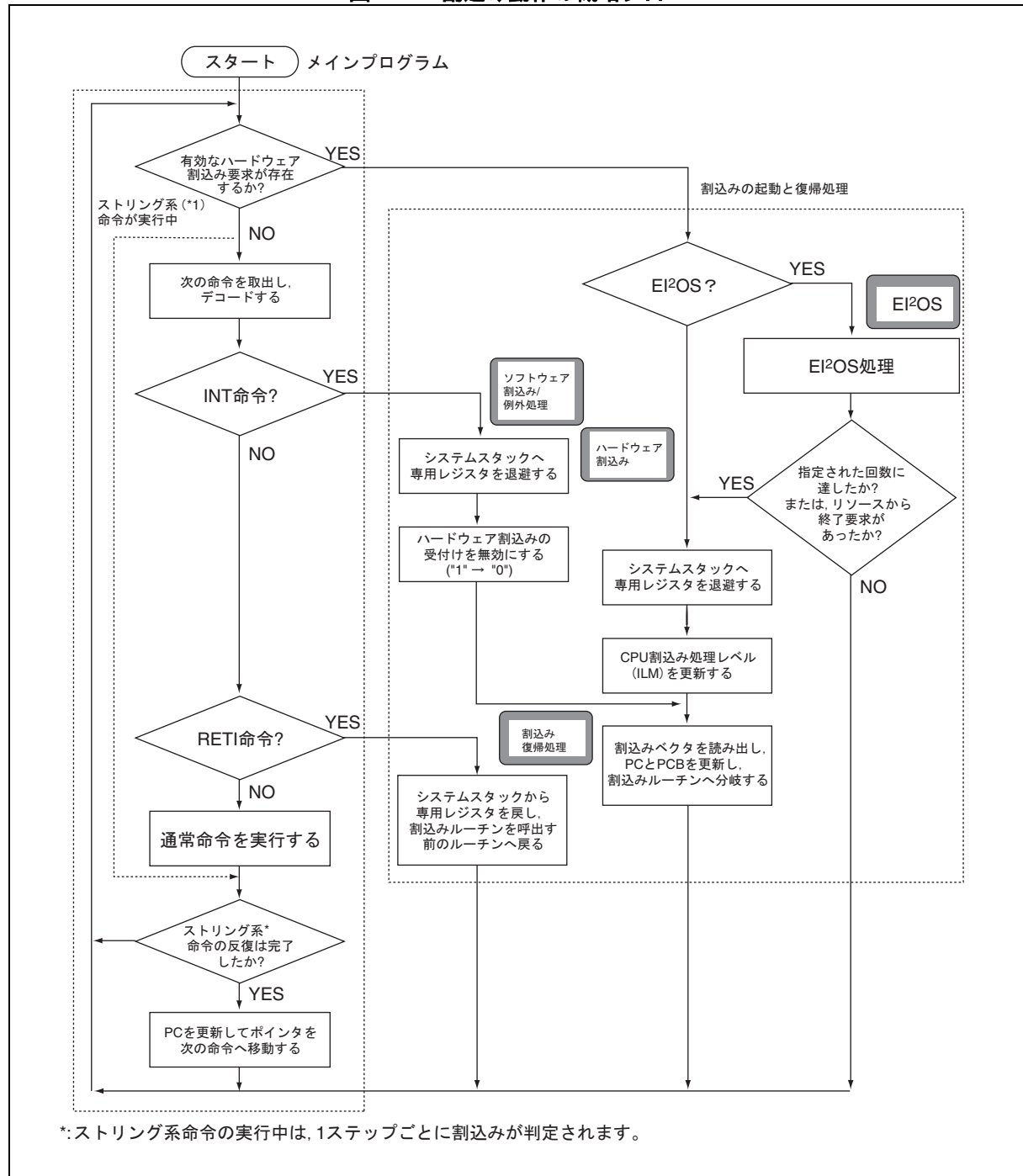
● 例外処理

例外処理は、基本的には割込みと同じものであり、命令の境界で例外事項 (未定義命令の実行) が発生したことを検出した場合、通常処理を中断して行われます。ソフトウェア割込み命令の "INT10" と等価です。

■ 割り込み動作

4種類の割り込み機能の、起動および復帰処理を図 7.1-1 に示します。

図 7.1-1 割り込み動作の概略フロー



7.2 割り込み要因と割り込みベクタ

F²MC-16LX には、256 種類の割り込み要因に対応する機能があり、256 の割り込みベクタテーブルは最上位アドレスのメモリに割り当てられています。この割り込みベクタはすべての割り込みで共有します。

ソフトウェア割り込みは、このすべての割り込み (INT0 ~ INT255) を使用できますが、ソフトウェア割り込みは、同じ割り込みベクタをハードウェア割り込みや例外処理割り込みと共用します。ハードウェア割り込みは、各リソースの固定割り込みベクタと割り込み制御レジスタ (ICR) を使用します。

■ 割り込みベクタ

● 割り込みベクタ

割り込み処理のときに参照される割り込みベクタテーブルは、メモリ領域の最上位アドレス ("FFFC00_H" ~ "FFFFFF_H") に割り当てられています。また、割り込みベクタは、EI²OS、例外処理、ハードウェアおよびソフトウェア割り込みと同じ領域を共用しています。

割り込み番号と割り込みベクタの割当てを表 7.2-1 に示します。

表 7.2-1 割り込みベクタ

ソフトウェア 割り込み命令	ベクタ アドレス L	ベクタ アドレス M	ベクタ アドレス H	モードデータ	割り込み 番号	ハードウェア割り込み
INT0	FFFFFC _H	FFFFFD _H	FFFFFE _H	未使用	#0	なし
:	:	:	:	:	:	:
INT7	FFFFE0 _H	FFFFE1 _H	FFFFE2 _H	未使用	#7	なし
INT8	FFFFDC _H	FFFFDD _H	FFFFDE _H	FFFFDF _H	#8	(リセットベクタ)
INT9	FFFFD8 _H	FFFFD9 _H	FFFFDA _H	未使用	#9	なし
INT10	FFFFD4 _H	FFFFD5 _H	FFFFD6 _H	未使用	#10	<例外処理>
INT11	FFFFD0 _H	FFFFD1 _H	FFFFD2 _H	未使用	#11	ハードウェア割り込み番号 0
INT12	FFFFCC _H	FFFFCD _H	FFFFCE _H	未使用	#12	ハードウェア割り込み番号 1
INT13	FFFFC8 _H	FFFFC9 _H	FFFFCA _H	未使用	#13	ハードウェア割り込み番号 2
INT14	FFFFC4 _H	FFFFC5 _H	FFFFC6 _H	未使用	#14	ハードウェア割り込み番号 3
:	:	:	:	:	:	:
INT254	FFFC04 _H	FFFC05 _H	FFFC06 _H	未使用	#254	なし
INT255	FFFC00 _H	FFFC01 _H	FFFC02 _H	未使用	#255	なし

< 参考 >

使用されない割込みベクタも、例外処理などのアドレスに設定することを推奨します。

■ 割り込み要因と割り込みベクタ・割り込み制御レジスタ

表 7.2-2 に割り込み要因 (ソフトウェア割り込みを除く)、割り込みベクタおよび割り込み制御レジスタの関係を示します。

表 7.2-2 割り込み要因と割り込みベクタ・割り込み制御レジスタ

割り込み要因	EI ² OS 対応	割り込みベクタ		割り込み制御レジスタ		優先度 (*2)	
		番号	アドレス	ICR	アドレス		
リセット	×	#08	08 _H	FFFFDC _H	-	-	高い ↑

: EI²OS は使用可能であり、割り込み要求フラグは EI²OS 割り込みクリア信号によってクリアされます。

× : EI²OS は使用不可。

: EI²OS は使用可能であり、EI²OS 停止要求もサポートされます。

: EI²OS は、ICR を共用する割り込み要因を使用しない場合は使用可能です。

- *1: - ICR レジスタを共用している周辺機能は、割り込みレベルが同一になります。
- ICR レジスタを共用している周辺機能で拡張インテリジェント I/O サービスを使用する場合は、どちらか一方しか利用できません。EI²OS クリアがサポートされると、両方の割り込み要因の割り込み要求フラグが EI²OS 割り込みクリア信号によってクリアされます。EI²OS 使用中にどちらかの割り込み要求をマスクすることをお勧めします。
- EI²OS サービスは同時に複数回使用できません。EI²OS 動作中に動作割り込み以外の割り込みがマスクされます。EI²OS 使用中にどちらかの割り込み要求をマスクすることをお勧めします。

*2: 同時に同じレベルの割り込みが発生した場合の優先度です。

7.3 割り込み制御レジスタとリソース

割り込み制御レジスタ (ICR00 ~ ICR15) は、割り込みコントローラ内にあり、割り込み機能を持つすべてのリソースに対応しています。これらのレジスタは、割り込みと拡張インテリジェント I/O サービス (EI²OS) を制御します。

■ 割り込み制御レジスタ一覧

表 7.3-1 に、割り込み制御レジスタと対応するリソースの一覧を示します。

表 7.3-1 割り込み制御レジスタ一覧

アドレス	レジスタ	略語	対応するリソース
0000B0 _H	割り込み制御レジスタ 00	ICR00	A/D コンバータ変換停止, アウトプットコンペア 0
0000B1 _H	割り込み制御レジスタ 01	ICR01	PWC0 による計測終了 /PWC0 オーバフロー
0000B2 _H	割り込み制御レジスタ 02	ICR02	アウトプットコンペア 1 一致, 16 ビット PPG タイマ 1
0000B3 _H	割り込み制御レジスタ 03	ICR03	アウトプットコンペア 2 一致, 16 ビットリロードタイマ 1 アンダフロー
0000B4 _H	割り込み制御レジスタ 04	ICR04	アウトプットコンペア 3 一致, DTP/ 外部割り込み 0/1 検出, DTTI
0000B5 _H	割り込み制御レジスタ 05	ICR05	アウトプットコンペア 4 一致, DTP/ 外部割り込み 2/3 検出
0000B6 _H	割り込み制御レジスタ 06	ICR06	アウトプットコンペア 5 一致, PWC1 による計測終了 / PWC1 オーバフロー
0000B7 _H	割り込み制御レジスタ 07	ICR07	DTP/ 外部割り込み 4/5 検出
0000B8 _H	割り込み制御レジスタ 08	ICR08	DTP/ 外部割り込み 6/7 検出
0000B9 _H	割り込み制御レジスタ 09	ICR09	波形生成部 16 ビットリロードタイマ 0/1/2 アンダフロー, 16 ビットリロードタイマ 0 アンダフロー
0000BA _H	割り込み制御レジスタ 10	ICR10	16 ビットフリーランタイム 0 検出, 16 ビット PPG タイマ 2
0000BB _H	割り込み制御レジスタ 11	ICR11	インプットキャプチャ 0/1, 16 ビットフリーランタイムコンペアクリア
0000BC _H	割り込み制御レジスタ 12	ICR12	インプットキャプチャ 2/3, タイムベースタイマ
0000BD _H	割り込み制御レジスタ 13	ICR13	UART1 受信, UART1 送信
0000BE _H	割り込み制御レジスタ 14	ICR14	UART0 受信, UART0 送信
0000BF _H	割り込み制御レジスタ 15	ICR15	フラッシュメモリステータス, 遅延割り込みジェネレータモジュール

■ 割り込み制御レジスタ機能

すべての割り込み制御レジスタ (ICR) は、以下の処理を行います。

- 対応するリソースの割り込みレベルを設定する。
- 対応するリソースの割り込みとして, " 通常割り込み " または " 拡張インテリジェント I/O サービス " を選択する。
- 拡張インテリジェント I/O サービス (EI²OS) のチャンネルを選択する。
- 拡張インテリジェント I/O サービス (EI²OS) の状態を表示する。

割り込み制御レジスタ (ICR) は、図 7.3-1 と図 7.3-2 に示すように、書込み時と読出し時で一部機能が異なります。

< 注意事項 >

割込み制御レジスタ (ICR) へアクセスする際，正常に動作しなくなりますのでリードモディファイライト命令は行わないでください。

7.3.1 割り込み制御レジスタ (ICR00 ~ ICR15)

割り込み制御レジスタは、割り込み機能を持つすべてのリソースに対応し、割り込み要求発生時の処理を制御します。割り込み制御レジスタは、書き込み時と読み出し時で一部機能が異なります。

■ 割り込み制御レジスタ (ICR00 ~ ICR15)

図 7.3-1 割り込み制御レジスタ (ICR00 ~ ICR15) 書き込み機能

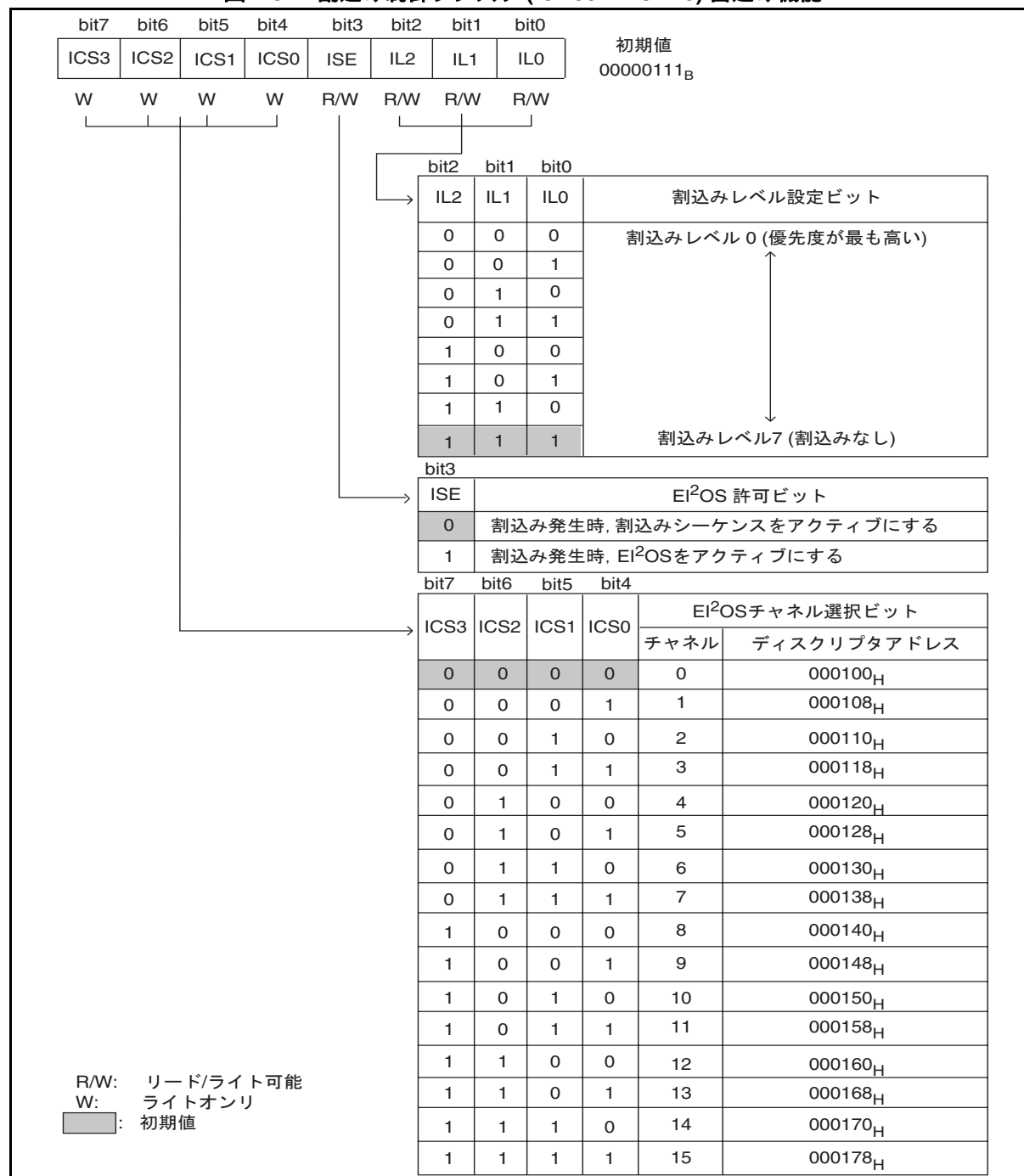
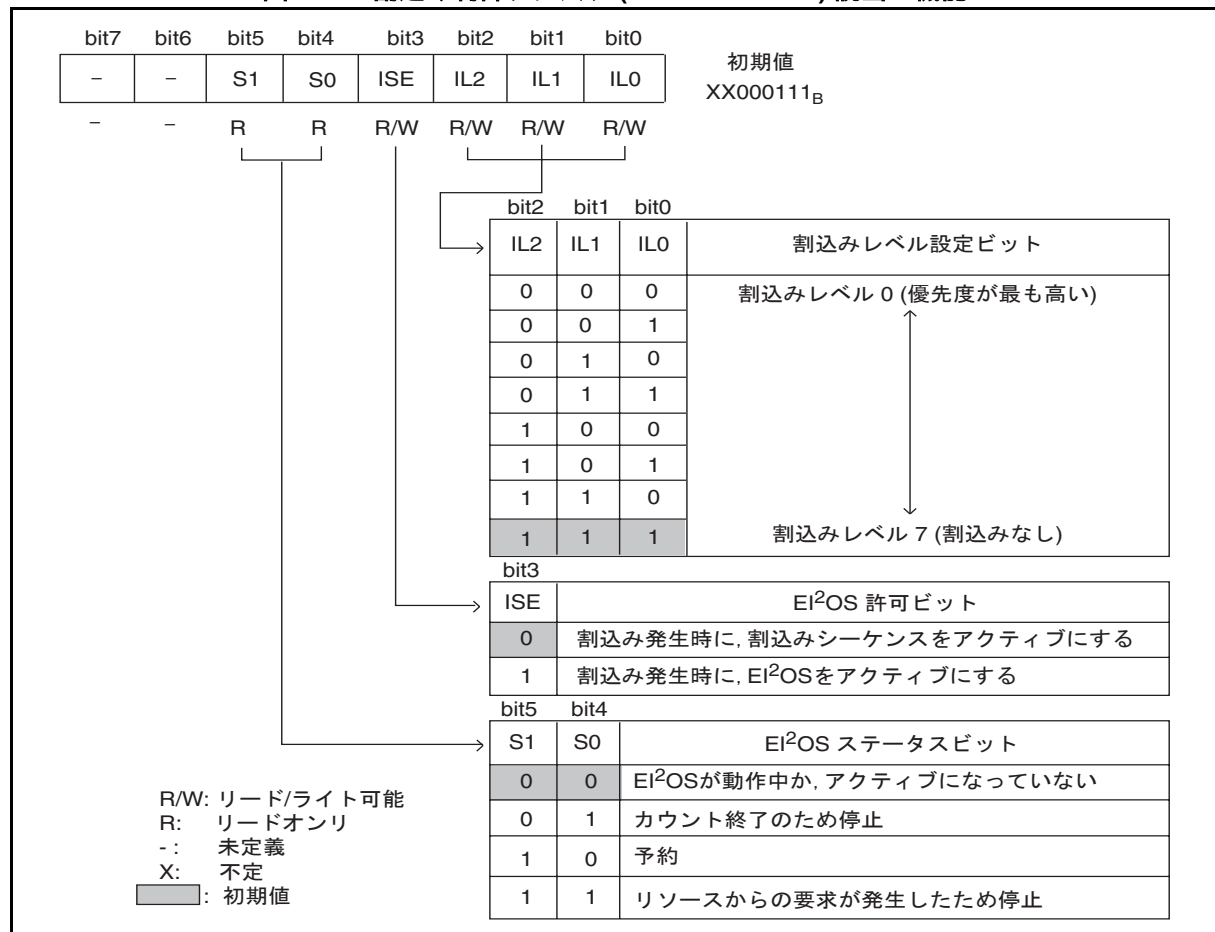


図 7.3-2 割り込み制御レジスタ (ICR00 ~ ICR15) 読出し機能



7.3.2 割込み制御レジスタの機能

割込み制御レジスタ (ICR00 ~ ICR15) は、以下の4つの機能ビットから構成されています。

- 割込みレベル設定ビット (IL2 ~ IL0)
- 拡張インテリジェント I/O サービス (EI²OS) 許可ビット (ISE)
- 拡張インテリジェント I/O サービス (EI²OS) チャンネル選択ビット (ICS3 ~ ICS0)
- 拡張インテリジェント I/O サービス (EI²OS) ステータスビット (S1, S0)

■ 割込み制御レジスタ (ICR) のビット構成

割込み制御レジスタ (ICR) のビット構成を図 7.3-3 に示します。

図 7.3-3 割込み制御レジスタ (ICR) の構成

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
割込み制御レジスタ (ICR) への書込み	ICS3	ICS2	ICS1	ICS0	ISE	IL2	IL1	IL0	00000111 _B
	W	W	W	W	W	W	W	W	
割込み制御レジスタ (ICR) の読出し	-	-	S1	S0	ISE	IL2	IL1	IL0	XX000111 _B
	-	-	R	R	R	R	R	R	

R: リードオンリ
W: ライトオンリ
-: 未定義

< 注意事項 >

- ICS3 ~ ICS0 ビットは、拡張インテリジェント I/O サービス (EI²OS) がアクティブな場合のみ有効となります。EI²OS を起動する場合は EI²OS 許可ビット (ISE) を "1" に設定し、起動しない場合は EI²OS 許可ビットを "0" に設定してください。EI²OS を起動しない場合は、ICS3 ~ ICS0 の設定はオプションです。
- ICS1, ICS0 は書込みのみ有効であり、S1 と S0 は読出しのみ有効となります。

■ 割り込み制御レジスタの機能

● 割り込みレベル設定ビット (IL2 ~ IL0)

これらのビットは、対応するリソースの割り込みレベルを設定します。リセットによりレベル 7(割り込みなし)に初期化されます。

表 7.3-2 に割り込みレベル設定ビットと割り込みレベルとの関係を示します。

表 7.3-2 割り込みレベル設定ビットと割り込みレベルの関係

IL2	IL1	IL0	割り込みレベル
0	0	0	0(優先度:最高) ↑ ↓ 6(優先度:最低)
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	7(割り込みなし)
1	1	1	

● 拡張インテリジェント I/O サービス (EI²OS) 許可ビット (ISE)

割り込み要求生成時にこのビットが "1" の場合は、EI²OS が起動されます。割り込み要求発生時にこのビットが "0" の場合は、割り込みシーケンスが起動されます。EI²OS の終了条件が成立した場合 (ステータスビット S1, S0 が "00_B" 以外) は、ISE ビットがクリアされます。対応するリソースに EI²OS 機能がない場合、ソフトウェアにより ISE ビットを "0" に設定してください。ISE ビットは、リセットにより "0" に初期化されます。

● 拡張インテリジェント I/O サービス (EI²OS) チャンネル選択ビット (ICS3 ~ ICS0)

これら書き込み専用のビットは、EI²OS のチャンネルを指定します。EI²OS ディスクリプタのアドレスは、これらのビット値に基づいて決定されます。ICS ビットは、リセットにより "0000_B" に初期化されます。

表 7.3-3 に EI²OS チャンネル選択ビットとディスクリプタアドレスの対応を示します。

表 7.3-3 EI²OS チャンネル選択ビットとディスクリプタアドレスの対応

ICS3	ICS2	ICS1	ICS0	セレクトされるチャンネル	ディスクリプタアドレス
0	0	0	0	0	000100 _H
0	0	0	1	1	000108 _H
0	0	1	0	2	000110 _H
0	0	1	1	3	000118 _H
0	1	0	0	4	000120 _H
0	1	0	1	5	000128 _H
0	1	1	0	6	000130 _H
0	1	1	1	7	000138 _H
1	0	0	0	8	000140 _H
1	0	0	1	9	000148 _H
1	0	1	0	10	000150 _H
1	0	1	1	11	000158 _H
1	1	0	0	12	000160 _H
1	1	0	1	13	000168 _H
1	1	1	0	14	000170 _H
1	1	1	1	15	000178 _H

● 拡張インテリジェント I/O サービス (EI²OS) ステータスビット (S1, S0)

これらは読出し専用ビットです。EI²OS 終了時にこの値をチェックすると、動作状態と終了状態を区別できます。これらのビットは、リセットにより "00_B" に初期化されます。

EI²OS ステータスビット (S1, S0) と EI²OS ステータスの関係を表 7.3-4 に示します。

表 7.3-4 EI²OS ステータスビットと EI²OS ステータスの関係

S1	S0	EI ² OS ステータス
0	0	EI ² OS が動作中あるいはアクティブになっていない
0	1	カウント終了のため停止
1	0	予約
1	1	リソースからの要求のため停止

7.4 ハードウェア割込み

ハードウェア割込み機能は、リソースからの割込み信号に応じて、CPU が実行中のプログラムを一時中断し、ユーザ定義の割込み処理プログラムへ制御を移します。拡張インテリジェント I/O サービス (EI²OS) や外部割込みなどは、ハードウェア割込みの一種として実行されます。

■ ハードウェア割込み

● ハードウェア割込み機能

- ハードウェア割込み機能は、リソースから出力された割込み要求信号の割込みレベルを、CPU のプロセッサステータス (PS) の割込みレベルマスクレジスタ (ILM) と比較します。そして、ハードウェアを介してプロセッサステータス (PS) の I フラグの値を参照し、割込みが受け付け可能か否かを決定します。
- ハードウェア割込みが受け付けられた場合、CPU 内部のレジスタは自動的にシステムスタックへ退避されます。現在要求されている割込みレベルは、割込みレベルマスクレジスタ (ILM) に格納され、ハードウェア割込み機能は対応する割込みベクタへ分岐します。

● 複数の割込み

複数のハードウェア割込みを起動できます。

● 拡張インテリジェント I/O サービス (EI²OS)

EI²OS は、メモリと I/O 間の自動転送機能です。指定された転送数が完了すると、ハードウェア割込みが起動されます。複数の EI²OS の起動は発生しません。EI²OS 処理中は、ほかの割込み要求と EI²OS 要求はすべて保留されます。

● 外部割込み

外部割込み (ウェイクアップ割込みも含む) は、ハードウェア割込みとしてリソース (割込み要求検出回路) から受け付けられます。

● 割込みベクタ

割込処理中に参照される割込みベクタテーブルは、"FFFC00_H" ~ "FFFFFF_H" までのメモリに割当てられており、これらのテーブルはソフトウェア割込みと共用されます。割込み番号と割込みベクタの割当てについては、「7.2 割込み要因と割込みベクタ」を参照してください。

■ ハードウェア割り込みの構造

表 7.4-1 にハードウェア割り込みに使用される 4 つの機構を示します。これら 4 つの機構は、ハードウェア割り込みを使用する前にプログラムに組込まれていなければなりません。

表 7.4-1 ハードウェア割り込みに関連する機構

	ハードウェア割り込みに関する機構	機能
リソース	割り込み許可ビット、割り込み要求ビット	リソースからの割り込み要求の制御
割り込みコントローラ	割り込み制御レジスタ (ICR)	割り込みレベルの設定および EI ² OS の制御
CPU	割り込み許可フラグ (I)	割り込み許可状態の識別
	割り込みレベルマスクレジスタ (ILM)	要求割り込みレベルと現割り込みレベルの比較
	マイクロコード	割り込み処理ルーチンの実行
メモリ上の "FFFC00 _H " ~ "FFFFFF _H "	割り込みベクタテーブル	割り込み処理の分岐先のアドレスを格納

■ ハードウェアの割り込みの抑止

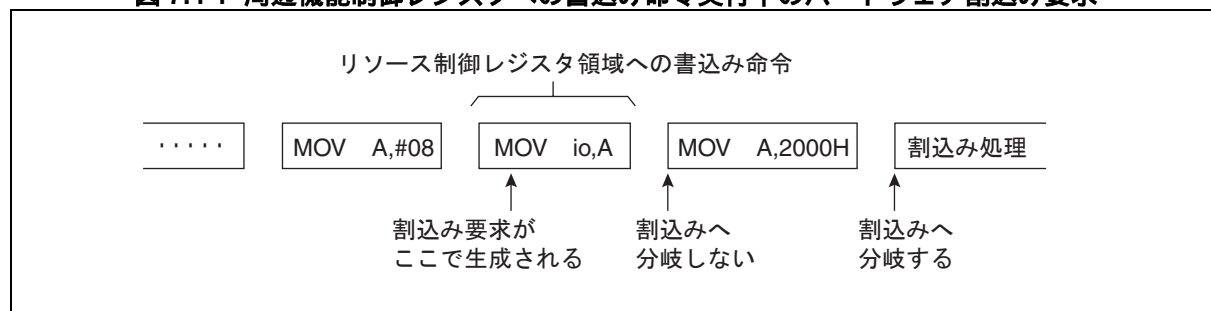
以下の条件において、ハードウェア割り込みの受付は抑制されます。

● リソース制御レジスタ領域への書き込み中のハードウェア割り込み抑止

リソース制御レジスタへのデータ書き込み中、ハードウェア割り込み要求は受け付けられません。これにより、CPU の誤動作を防ぎます。リソースの割り込み制御レジスタへの書き込み時に割り込み要求が生成された場合、この誤動作が発生する場合があります。リソース制御レジスタ領域は、"000000_H" ~ "0000FF_H" の I/O アドレス指定領域ではなく、リソース制御レジスタとデータレジスタに割当てられた領域のことです。

図 7.4-1 に内蔵リソース領域への書き込み時のハードウェア割り込み動作を示します。

図 7.4-1 周辺機能制御レジスタへの書き込み命令実行中のハードウェア割り込み要求



● 割込み抑止命令実行中のハードウェア割込抑止

表 7.4-2 に示す 10 種類のハードウェア割込み抑止命令は、ハードウェア割込み要求の有無の検出をせず、割込み要求を無視します。

表 7.4-2 ハードウェア割込抑止命令

	プリフィックス コード	割込み / ホールド抑止命令 (プリフィックスコードの効果を遅延させる命令)
割込みを受け付けず、 要求を保持しない命令	PCB DTB ADB SPB CMR NCC	MOV ILM, #imm8 OR CCR, #imm8 AND CCR, #imm8 POPW PS

これらいずれかの抑止命令を実行中に有効なハードウェア割込み要求が生成されても、この割込みは、別種類の命令が初めて実行されるまで処理されません。

● ソフトウェア割込の実行中

ソフトウェア割込みが起動した場合は、I フラグが "0" にクリアされますので、ほかの割込み要求を受付けることはできません。

7.4.1 ハードウェア割込みの動作

ここでは、ハードウェア割込み要求発生から割込み処理完了までのハードウェア割込み動作について説明します。

■ ハードウェア割込みの起動

● リソースの動作 (割込み要求の発生)

ハードウェア割込み要求機能を持つリソースは、割込み要求が存在することを示す割込み要求フラグと、割込み要求の許可 / 禁止を判定する割込み許可フラグがあります。割込み要求フラグは、リソース固有のイベントが発生した場合に設定されます。

● 割込みコントローラの動作 (割込み要求の制御)

割込みコントローラは、同時に受け取った割込み要求の割込みレベル (IL) を比較し、最も高いレベル (IL 値の最も小さい) の要求を選択し、それを CPU へ通知します。複数の要求の割込みレベルが同じ場合は、割込み番号の小さい要求が最も高い優先度となります。

● CPU の動作 (割込み要求の受付けと割込み処理)

CPU は、受け取った割込みレベル (ICR: IL2 ~ IL0) と割込みレベルマスクレジスタ (ILM) の値を比較します。割込みレベル (IL) が割込みレベルマスクレジスタ (ILM) の値よりも小さく、割込みが許可 (PS: CCR: I = 1) されている場合に、現在実行中の命令が終了した後、割込み処理マイクロコードを起動します。

CPU は、割込み処理マイクロコードの先頭で、EI²OS 許可ビット (ISE) を参照し、ISE ビットが "0" の場合、引き続き割込処理を実行します (ISE ビットが "1" の場合、EI²OS が起動されます)。

割込み処理では、専用レジスタ (A, DPR, ADB, DTB, PCB, PC, PS を含む 12 バイト) の値がシステムスタック (SSB と SSP の示すシステムスタック空間) へ退避されます。

次に CPU は、割込みベクタプログラムカウンタ (PCB, PC) ヘデータをロードし、割込みレベルマスクレジスタ (ILM) を更新し、スタックフラグを設定します (CCR: S = 1 を設定し、システムスタックを起動する)。

■ ハードウェア割込みからの復帰

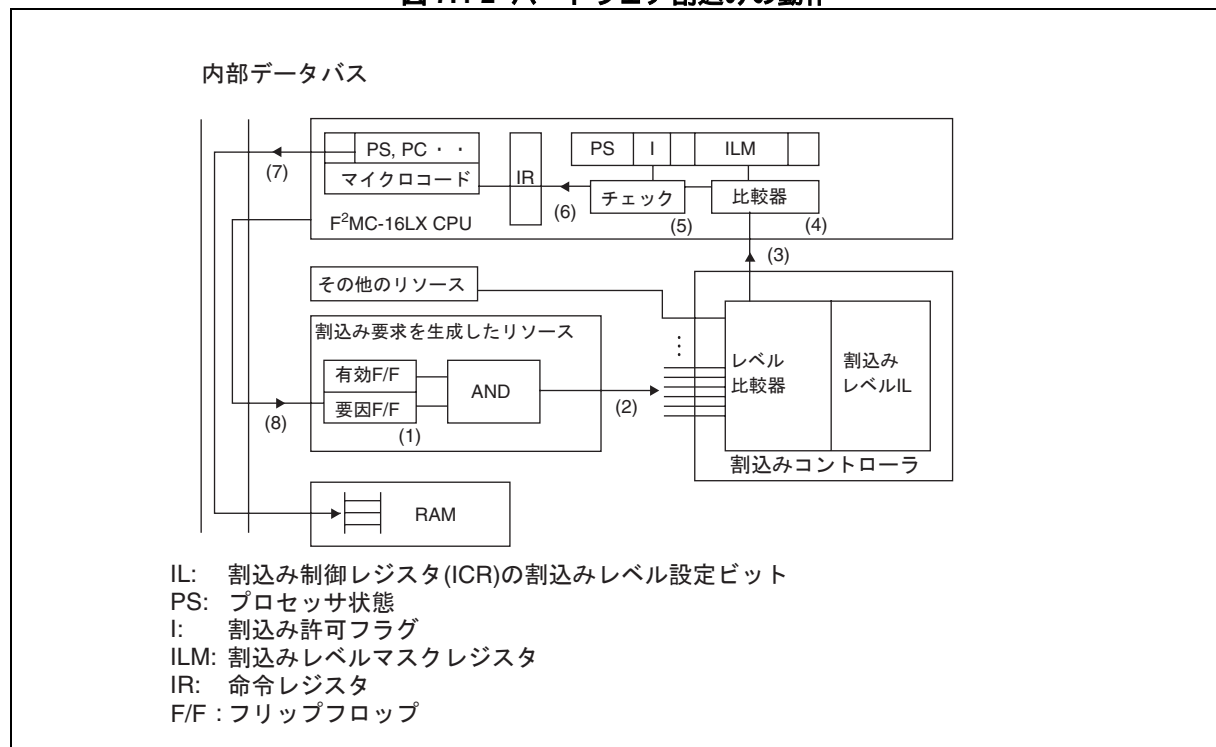
割込み処理プログラムでは、割込み要因を発生したリソースの割込み要求フラグがクリアされて RETI 命令が実行されると、システムスタックに退避した 12 バイトデータが専用レジスタへ復元され、割込み分岐前に実行していた処理を再開します。

割込み要求フラグがクリアされると、リソースが割込みコントローラへ出力した割込み要求は自動的にキャンセルされます。

■ ハードウェア割り込みの動作

ハードウェア割り込みの発生から割り込み処理完了までの動作を図 7.4-2 に示します。

図 7.4-2 ハードウェア割り込みの動作



1. 割り込み要求がリソース内部で発生します。
2. リソースの割り込み許可ビットが参照され、割り込み許可ビットが割り込み許可に設定されている場合は、割り込み要求がリソースから割り込みコントローラへ出力されます。
3. 割り込み要求を受け取る割り込みコントローラは、同時に受け取った割り込み要求の優先度を決定し、対応する割り込み要求に一致する割り込みレベル(IL)をCPUへ転送します。
4. CPUは、割り込みコントローラにより要求された割り込みレベル(IL)を割り込みレベルマスクレジスタ(ILM)と比較します。
5. 比較の結果、当該割り込みレベルの優先度が現割り込み処理レベルよりも高いことを示している場合は、CPUはコンディションコードレジスタの(CCR)のIフラグをチェックします。
6. 上記5のチェックにおいてIフラグが割り込み許可(I=1)されている場合、CPUは現在実行中の命令が終了するまで待ちます。終了後、CPUは要求されたレベル(IL)を割り込みレベルマスクレジスタ(ILM)に設定します。
7. レジスタが退避され、処理は割り込み処理ルーチンへ分岐します。
8. 上記1で発生した割り込み要因は、割り込み処理ルーチンによりソフトウェア的にクリアされます。RETI命令を実行すると、割り込み処理は終了します。

7.4.2 割り込み動作時の処理

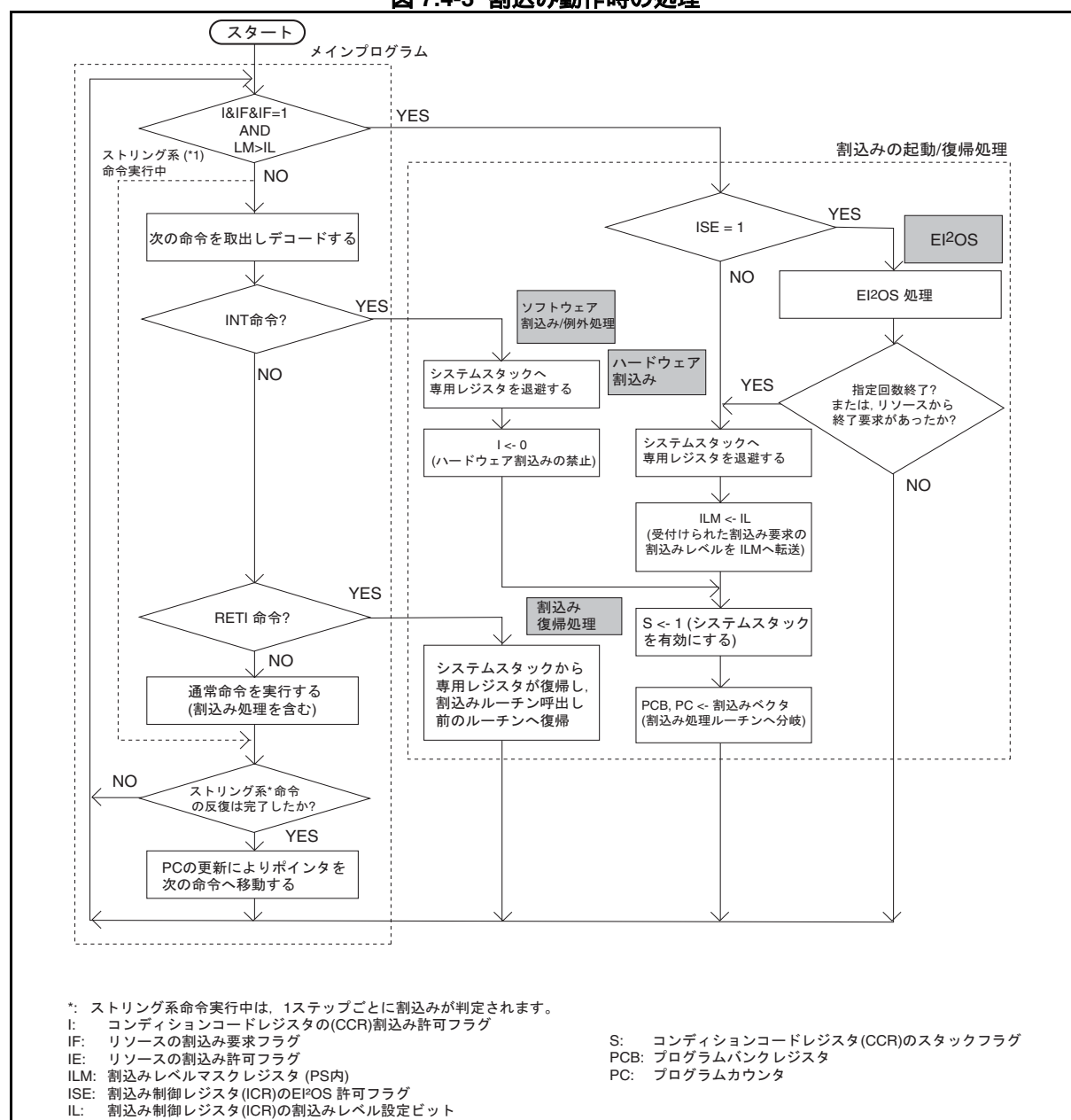
リソースにより割り込み要求が発生すると、割り込みコントローラは割り込みレベルを CPU に送信し、CPU が割り込みを受付ける可能な場合は、現在実行中の命令を一時中断します。そして割り込みコントローラは、割り込み処理ルーチンの実行または、拡張インテリジェント I/O サービス (EI²OS) を起動します。

ソフトウェア割り込みが INT 命令により生成された場合、CPU の状態に関係なく割り込み処理ルーチンが実行されます。この場合、ハードウェア割り込みは許可されません。

■ 割り込み動作時の処理

図 7.4-3 に割り込み動作時の処理フローを示します。

図 7.4-3 割り込み動作時の処理



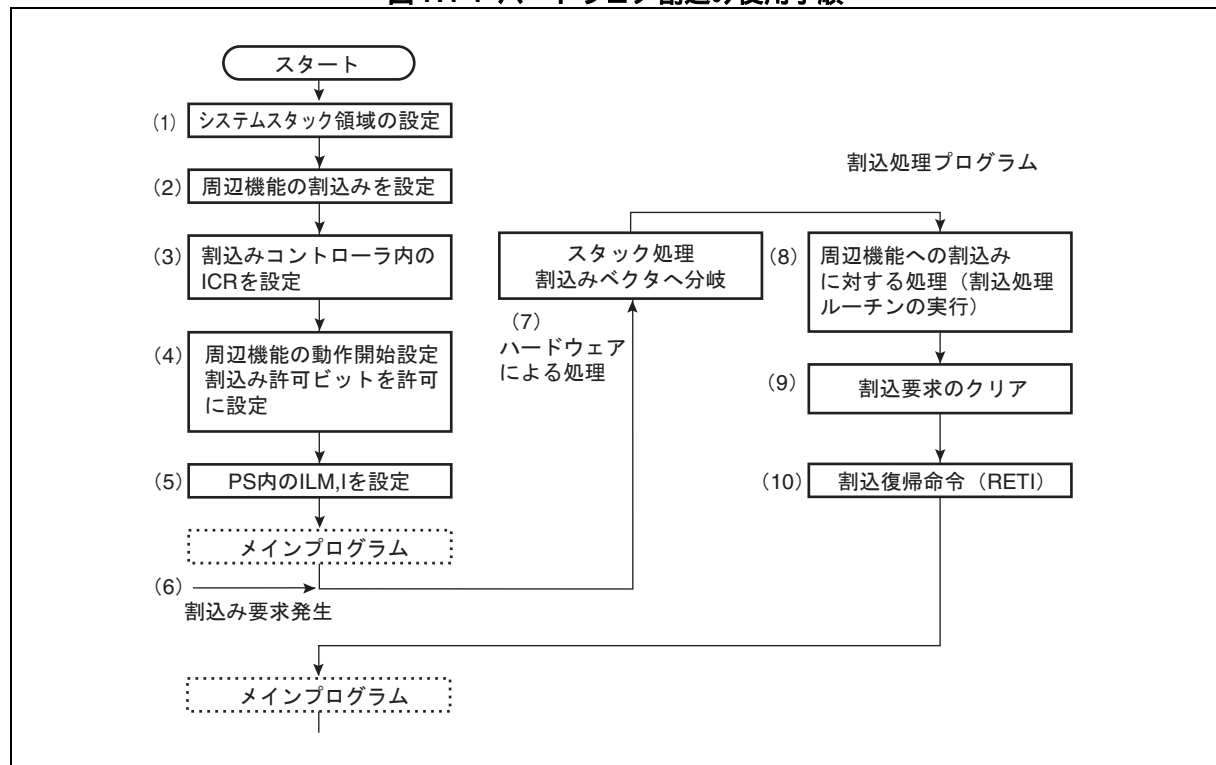
7.4.3 ハードウェア割込み使用手順

ハードウェア割込みを使用する前に、システムスタック領域、リソース、および割込制御レジスタ (ICR) を設定する必要があります。

■ ハードウェア割込み使用手順

図 7.4-4 にハードウェア割込みの使用手順の一例を示します。

図 7.4-4 ハードウェア割込み使用手順



1. システムスタック領域を設定してください。
2. 割込み要求を発生できるリソースを初期化してください。
3. 割込みコントローラの割込み制御レジスタ (ICR) を設定してください。
4. リソースを動作開始状態にし、割込み許可ビットを "許可" に設定してください。
5. 割込みレベルマスクレジスタ (ILM) と割込み許可フラグ (I) を割込み受け可能に設定してください。
6. リソースにおいて発生した割込みによりハードウェア割込み要求が発生されます。
7. 割込み処理ハードウェアがレジスタを退避し、割込み処理プログラムへ分岐します。
8. 割込み処理プログラムは、発生した割込みに対するリソースを処理します。
9. リソースからの割込み要求をクリアしてください。
10. 割込み復帰命令を実行し、分岐する前のプログラムに復帰します。

7.4.4 複数の割込み

リソースからの複数の割込み要求が発生した場合、割込みレベル設定ビット (IL0 ~ IL2) に異なる割込みレベルを設定することで、複数のハードウェア割込みを実行できます。ただし、拡張インテリジェント I/O サービスで複数の割込みを使用することはできません。

■ 複数の割込み

● 複数の割込み動作

- 割込み処理ルーチンの実行中に、優先度がより高い割込みレベルが発生すると、現在の割込み処理が中断され、より高い割込み優先度を持つ割込み要求を受け付けます。この割込み要求が終了すると、CPU は中断されていた割込み処理を再開します。
- 割込みレベル (IL) としては "0" から "7" までを設定できますが、レベル 7 を設定すると CPU は割込み要求を受け付けません。
- 割込み処理実行中に、割込み優先順位が同じかまたは低いレベルの割込み要求が生成されると、コンディションコードレジスタ (CCR) の I フラグまたは割込みレベルマスクレジスタ (ILM) が変更されない限り、現在の割込み処理が復帰するまで新しい割込み要求は保留されます。
- 割込み処理ルーチンのコンディションコードレジスタ (CCR) の I フラグを割込み禁止 (I = 0) にするか、または割込みレベルマスクレジスタ (ILM) を割込み禁止 (ILM = 000_B) に設定すると、割込み中に起動予定であるほかの複数の割込みを一時的に禁止できます。

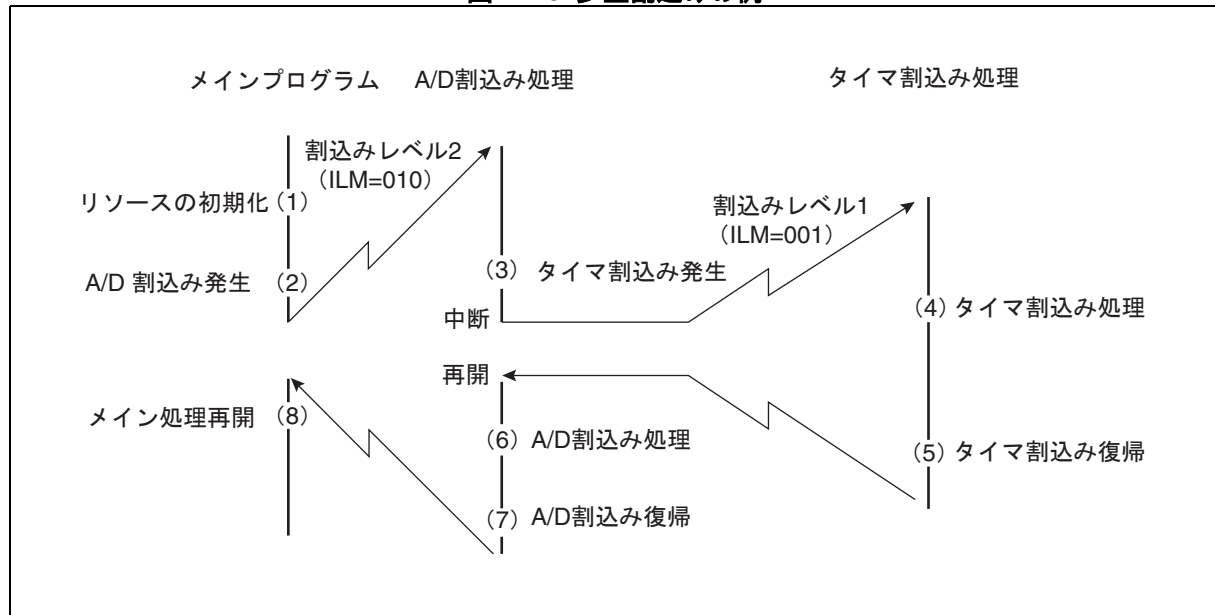
< 注意事項 >

拡張インテリジェント I/O サービス (EI²OS) を使用して複数の割込みを起動することはできません。拡張インテリジェント I/O サービス (EI²OS) の処理中は、ほかの割込み要求および拡張インテリジェント I/O サービス要求はすべて保留されます。

● 複数の割込み処理例

複数の割込み処理例では、タイマ割込みの優先度が A/D コンバータ割込みの優先度よりも高いものと仮定します。この例では、A/D コンバータの割込みレベルとして "2" が設定されており、タイマ割込みレベルとして "1" が設定されています。A/D コンバータの割込み処理中にタイマ割込みが発生すると、図 7.4-5 に示すような処理が実行されます。

図 7.4-5 多重割り込みの例



(1) A/D 割り込み発生

A/D コンバータの割り込み処理が開始されると、割り込みレベルマスクレジスタ (ILM) には自動的に A/D コンバータの割り込みレベル (ICR: IL2 ~ IL0) と同じ値 (例では 2) が設定されます。レベルが "1" または "0" の割り込み要求が発生した場合は、これらの割り込み処理が優先されます。

(2) 割り込み処理の終了

割り込み処理が終了し、復帰命令 (RETI) が実行されると、専用レジスタ (A, DPR, ADB, DTB, PCB, PC, PS) の値がスタックから戻され、割り込みレベルマスクレジスタ (ILM) には、割り込み発生前の値が設定されます。

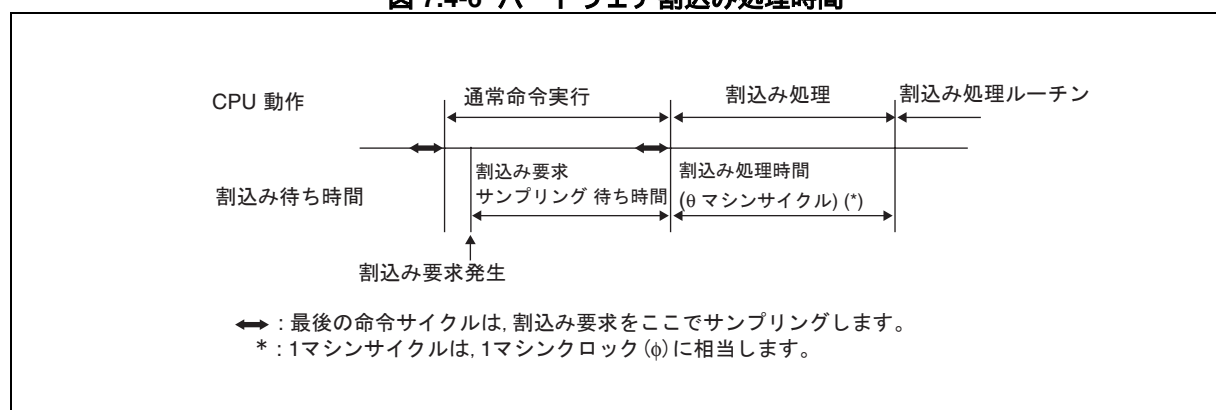
7.4.5 ハードウェア割込み処理時間

ハードウェア割込み要求が発生してから割込み処理ルーチンが実行されるまでには、現在実行中の命令を終了するまでの時間と、割込みを処理するための時間が必要です。

■ ハードウェア割込みの処理時間

ハードウェア割込み要求が発生してから割込みを受け、割込み処理ルーチンを実行するまでには、割込み要求をサンプリングするための待ち時間と割込みを処理するための時間（割込み処理準備に要する時間）が必要です。図 7.4-6 に、割込み処理時間を示します。

図 7.4-6 ハードウェア割込み処理時間



● 割込み要求サンプリング待ち時間

- 割込み要求サンプリング待ち時間は、割込み要求が生成されてから現在実行中の命令が終了するまでの時間です。
- 割込み要求が発生したか否かは、最後の命令サイクルの割込み要求命令をサンプリングして判断されます。そのため、各命令の実行中において CPU は割込み要求を識別できず、待ち時間が発生します。
- 割込み要求サンプリング待ち時間は、実行時間が最も長い POPW RW0, ... RW7 命令 (45 マシンサイクル) 開始された直後に割込み要求が発生した時点で最大となります。

● 割り込み処理時間 (θ マシンサイクル)

CPU は、割り込み要求を受け取ると、システムスタックに専用レジスタを退避し、割り込みベクタを読み出します。割り込み処理時間は、θ マシンサイクルです。割り込み処理時間は、以下の式で算出されます。

割り込み起動時： $= 24 + 6 \times Z$ マシンサイクル

割り込みから制御が復帰した時： $= 15 + 6 \times Z$ マシンサイクル (RETI 命令)

割り込み処理時間は、スタックポインタが指す各アドレスによって異なります。表 7.4-3 に割り込み処理時間の補間値 (Z) を示します。

表 7.4-3 割り込みハンドリング時間の補間値 (Z)

スタックポインタが指すアドレス	補間値 (Z)
偶数番号の内部アドレス	0
奇数番号の内部アドレス	+2

< 参考 >

1 マシンサイクルは、マシクロック (φ) の 1 クロック周期に相当します。

7.5 ソフトウェア割込み

ソフトウェア割込み命令 (INT 命令) を実行すると、ソフトウェア割込み機能は、CPU で実行中のプログラムからユーザ定義した割込み処理プログラムへ制御を移します。ソフトウェア割込みの実行中は、ハードウェア割込みは禁止されます。

■ ソフトウェア割込みの起動

● ソフトウェア割込みの起動

INT 命令は、ソフトウェア割込みを起動するために使用します。ソフトウェア割込み要求には、割込み要求フラグおよび割込み許可フラグは存在しません。INT 命令を実行すると、割込み要求が常に発生します。

● ハードウェア割込みの抑止

INT 命令は割込みレベルを持たないので、割込みレベルマスクレジスタ (ILM) は更新されません。INT 命令の実行中、コンディションコードレジスタ (CCR) の割込み許可フラグ (I) が "0" に設定され、ハードウェア割込みがマスクされます。

ソフトウェア割込み処理中にハードウェア割込みを許可する場合は、ソフトウェア割込み処理ルーチンにおいて、I フラグに "1" に設定してください。

● ソフトウェア割込みの動作

CPU が INT 命令を取込むと、ソフトウェア割込み処理マイクロコードが起動されます。マイクロコードは、内部 CPU レジスタをシステムスタックに退避し、ハードウェア割込みをマスクし (CCR : I = 0)、対応する割込みベクタへ分岐します。

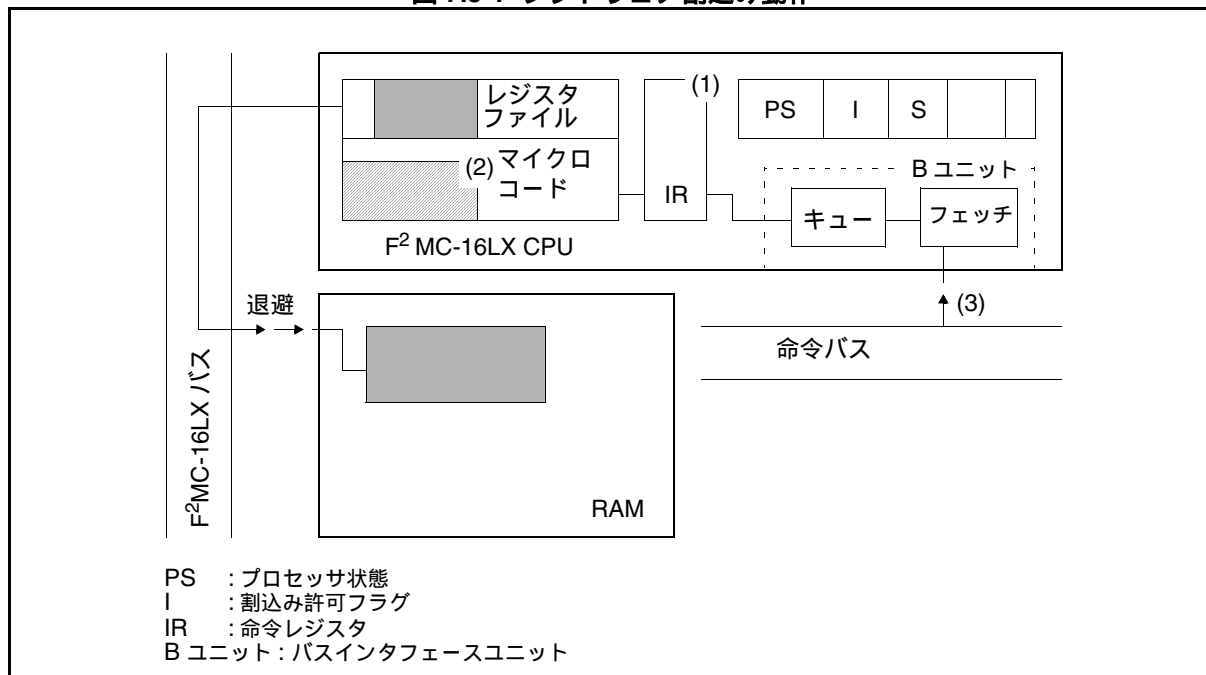
■ ソフトウェア割込みからの復帰

割込み処理プログラムにおいて、割込み復帰命令 (RETI 命令) を実行すると、システムスタックに退避されている 12 バイトデータが専用レジスタに復元され、割込みへ分岐する前に実行していた処理を再開します。

■ ソフトウェア割り込み動作

図 7.5-1 にソフトウェア割り込みの発生から割り込み処理完了までの動作を示します。

図 7.5-1 ソフトウェア割り込み動作



1. ソフトウェア割り込み命令が実行されます。
2. ソフトウェア割り込み命令に対応するマイクロコードに従って、専用レジスタは退避され、ほかの必要な処理が実行されます。次に分岐処理が行われます。
3. ユーザの割り込み処理ルーチン内の RETI 命令で割り込み処理が終了します。

< 注意事項 >

プログラムバンクレジスタ (PCB) が "FF_H" の場合、CALLV 命令のベクタ領域は INT #vct8 命令のテーブルとオーバーラップします。ソフトウェアを作成する場合は、CALLV 命令と INT #vct8 命令のアドレス重複に注意してください。

7.6 拡張インテリジェント I/O サービス (EI²OS) の割込み

拡張インテリジェント I/O サービス (EI²OS) は、周辺機能 (I/O) とメモリとの間で自動データ転送を行う機能です。データ転送が終了した場合は、ハードウェア割込みが発生します。

■ 拡張インテリジェント I/O サービス (EI²OS)

拡張インテリジェント I/O サービスは、ハードウェア割込みの一種です。拡張インテリジェント I/O サービスは、自動的にリソース (I/O) とメモリの間においてデータ転送を行います。従来、リソース (I/O) とメモリの間におけるデータ転送は、割込み処理プログラムで行っていました。EI²OS は、DMA (ダイレクト メモリ アクセス) と同じ方法でデータを転送します。終了時に EI²OS は終了条件を設定し、自動的に割込み処理ルーチンへ分岐します。ユーザは、EI²OS 起動および終了するためのプログラムのみを作成する必要があり、データ転送プログラムの作成は不要です。

● 拡張インテリジェント I/O サービス (EI²OS) の利点

割込み処理ルーチンで実行されるデータ転送と比較すると、EI²OS は以下の利点があります。

- データ転送プログラムをコード化する必要がなく、プログラムサイズを小さくすることが可能です。
- データ転送はリソース (I/O) の状態によって停止できるので、不要なデータ転送を削除することが可能です。
- バッファアドレスを更新するかまたは更新しないかの選択ができます。
- I/O レジスタアドレスの更新または更新しないかの選択ができます。

● 拡張インテリジェント I/O サービス (EI²OS) の終了割込み

EI²OS によるデータ転送が終了すると、終了条件は割込み制御レジスタ (ICR) の S1, S0 ビットに設定され、処理は自動的に割込み処理ルーチンへ分岐します。

EI²OS 終了要因は、割込み処理プログラムで EI²OS 状態 (ICR : S1, S0) をチェックすることで決定できます。

割込み番号と割込みベクタは、各リソースに対して固定されています。詳細は「7.2 割込み要因と割込みベクタ」をご参照ください。

● 割込み制御レジスタ (ICR)

割込みコントローラ内に存在するこのレジスタは、EI²OS の起動、EI²OS のチャネル指定、および EI²OS 終了状態を表示します。

● 拡張インテリジェント I/O サービス (EI²OS) のディスクリプタ (ISD)

このディスクリプタは、"000100_H" から "00017F_H" までの内部 RAM に配置され、転送モード、I/O アドレス、転送数およびバッファアドレスを保持する 8 バイト × 16 チャネルで構成されています。チャネルは、割込み制御レジスタ (ICR) で指定します。

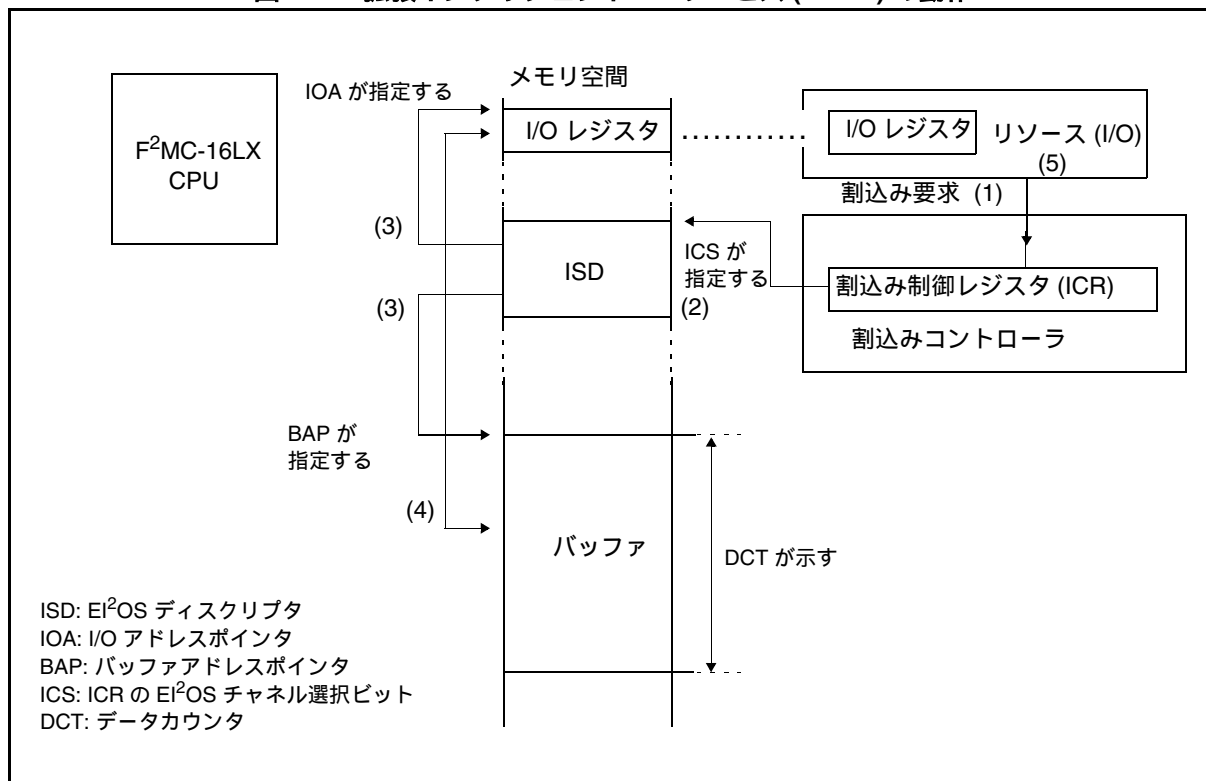
< 注意事項 >

拡張インテリジェント I/O サービス (EI²OS) 動作中の場合は、CPU プログラムは停止します。

■ 拡張インテリジェント I/O サービス (EI²OS) の動作

EI²OS の動作を図 7.6-1 に示します。

図 7.6-1 拡張インテリジェント I/O サービス (EI²OS) の動作



1. I/O が転送を要求します。
2. 割り込みコントローラがディスクリプタを選択します。
3. ディスクリプタから転送元と転送先が読み出されます。
4. I/O とメモリの間で転送が行われます。
5. 割り込み要因が自動的にクリアされます。

7.6.1 拡張インテリジェント I/O サービス (EI²OS) ディスクリプタ (ISD)

拡張インテリジェント I/O サービス (EI²OS) のディスクリプタ (ISD) は, "000100_H" から "00017F_H" までの内部 RAM に存在します。ISD は, 8 バイト × 16 チャンネルで構成されています。

■ 拡張インテリジェント I/O サービス (EI²OS) ディスクリプタ (ISD) の構成

ISD は, 8 バイト × 16 チャンネルで構成されており, 各 ISD は図 7.6-2 に示すような構成になっています。チャンネル番号と ISD のアドレスの対応を表 7.6-1 に示します。

図 7.6-2 EI²OS ディスクリプタ (ISD) の構成

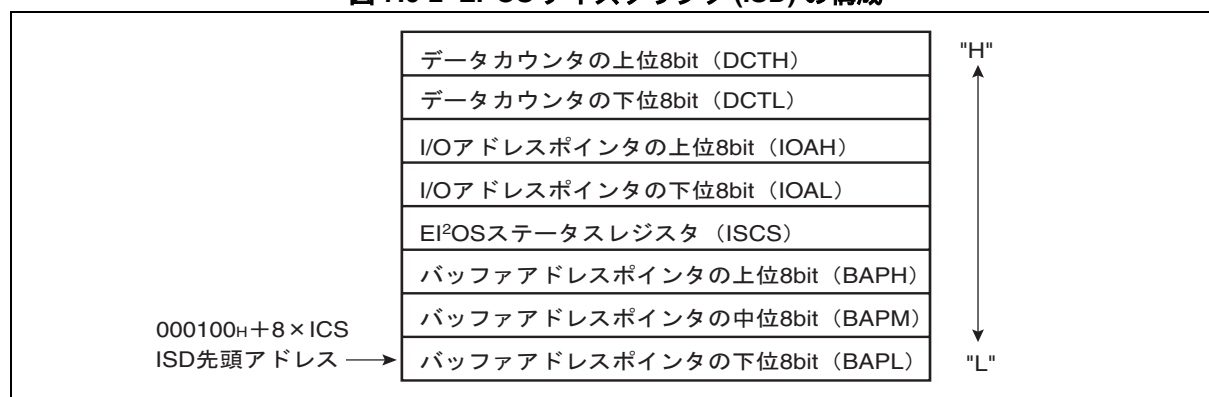


表 7.6-1 チャンネル番号と ISD のアドレスの対応

チャンネル	ディスクリプタアドレス
0	000100 _H
1	000108 _H
2	000110 _H
3	000118 _H
4	000120 _H
5	000128 _H
6	000130 _H
7	000138 _H
8	000140 _H
9	000148 _H
10	000150 _H
11	000158 _H
12	000160 _H
13	000168 _H
14	000170 _H
15	000178 _H

7.6.2 EI²OS ディスクリプタ (ISD) のレジスタ

- データカウンタ (DCT)
- I/O レジスタアドレスポインタ (IOA)
- EI²OS ステータスレジスタ (ISCS)
- バッファアドレスポインタ (BAP)

リセット後の各レジスタの初期値は不定なので注意してください。

■ データカウンタ (DCT)

データカウンタ (DCT) は、データ転送数のカウンタとしての役割を果たす 16 ビットのレジスタです。各データ転送が実行されると、カウンタは一つデクリメントします。カウンタが "0" になると、EI²OS は終了します。

データカウンタ (DCT) の構成を図 7.6-3 に示します。

図 7.6-3 データカウンタ (DCT) の構成

データカウンタの 上位バイト DCTH	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
	B15	B14	B13	B12	B11	B10	B09	B08	XXXXXXXX _B
データカウンタの 下位バイト DCTL	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
	B07	B06	B05	B04	B03	B02	B01	B00	XXXXXXXX _B

■ I/O アドレスポインタ (IOA)

IOA は、バッファへ (から) データを転送するために使用される、I/O レジスタの下位アドレス (A15 ~ A00) を示す 16 ビットのレジスタです。上位アドレス (A23 ~ A16) はすべて "0" であり、"000000_H" から "00FFFF_H" までの領域は、すべてアドレスで指定できます。

IOA の構成を図 7.6-4 に示します。

図 7.6-4 I/O アドレスポインタ (IOA) の構成

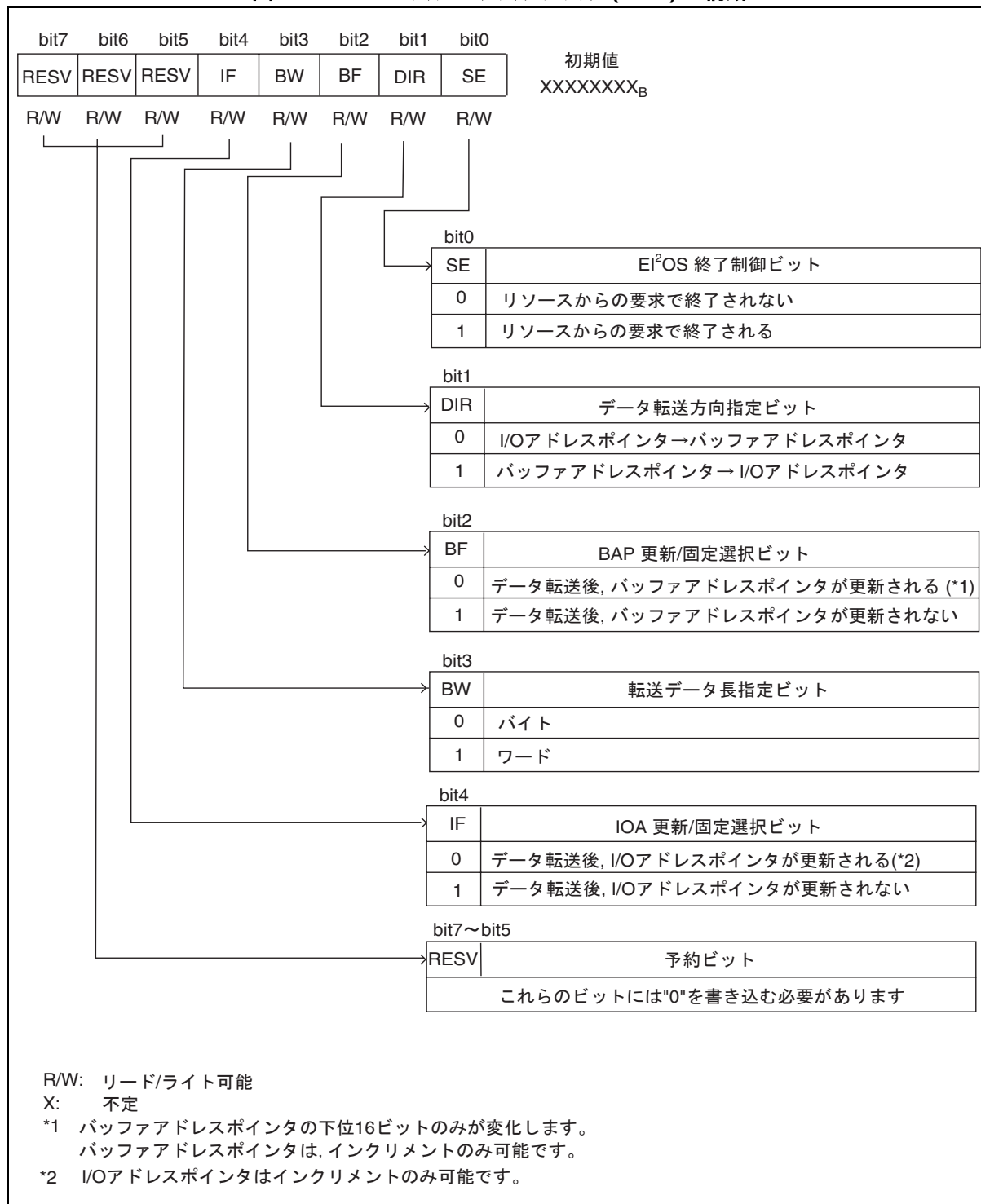
上位アドレスポインタ IOAH	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
	A15	A14	A13	A12	A11	A10	A09	A08	XXXXXXXX _B
下位アドレスポインタ IOAL	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
	A07	A06	A05	A04	A03	A02	A01	A00	XXXXXXXX _B

■ 拡張インテリジェント I/O サービス (EI²OS) ステータスレジスタ (ISCS)

ISCS は、8 ビットのレジスタです。バッファアドレスポインタと I/O アドレスポインタを更新するか否か、転送データ形式 (バイト / ワード)、および転送方向を示します。

EI²OS ステータスレジスタ (ISCS) のビット構成を図 7.6-5 に示します。

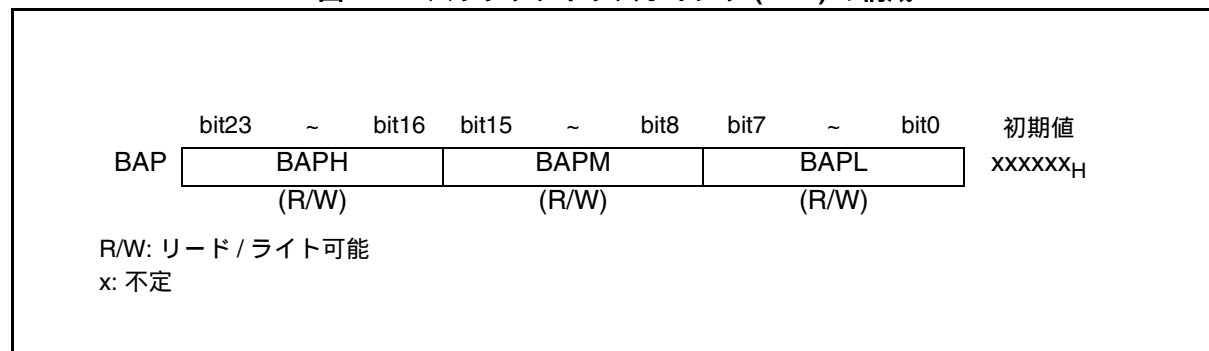
図 7.6-5 EI²OS ステータスレジスタ (ISCS) の構成



■ バッファアドレスポインタ (BAP)

BAP は、次の転送で EI²OS が使用するアドレスを保持する 24 ビットレジスタです。各 EI²OS チャンネルには 1 つの独立した BAP が存在するので、各 EI²OS チャンネルは 16M バイト空間内のアドレスと I/O との間でデータを転送できます。EI²OS 状態レジスタ (ISCS) の BF ビット (BAP 更新 / 固定選択ビット) で "更新" と設定した場合、BAP の下位 16 ビット (BAPM, BAPL) のみが変わり、上位 8 ビット (BAPH) は変化しません。図 7.6-6 に BAP の構成を示します。

図 7.6-6 バッファアドレスポインタ (BAP) の構成



< 参考 >

- I/O アドレスポインタ (IOA) で指定できる領域は、"000000_H" ~ "00FFFF_H" です。
- バッファアドレスポインタ (BAP) で指定できる領域は、"000000_H" ~ "FFFFFF_H" です。
- データカウンタ (DCT) で指定できる最大転送回数は、65,536 回 (64K バイト) です。

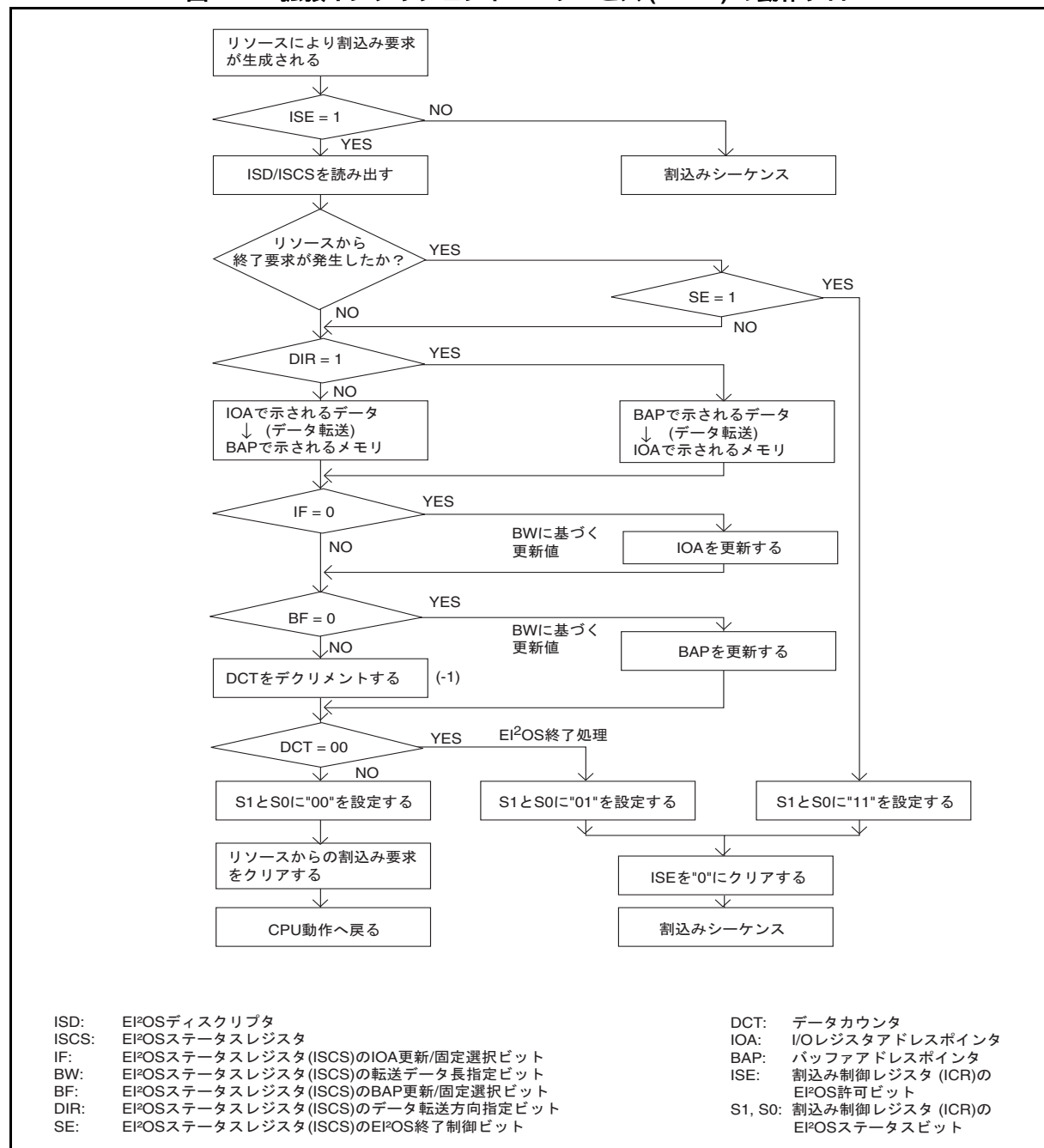
7.6.3 拡張インテリジェント I/O サービス (EI²OS) の動作

割り込み要求がリソースにて発生し、対応する割り込み制御レジスタ (ICR) で EI²OS の起動が設定され、CPU は EI²OS を使用してデータ転送を行います。指定されたデータ転送数が終了すると、ハードウェア割り込みが自動的に処理されます。

■ 拡張インテリジェント I/O サービス (EI²OS) の処理手順

図 7.6-7 に CPU 内部のマイクロコードによる、EI²OS の動作フローを示します。

図 7.6-7 拡張インテリジェント I/O サービス (EI²OS) の動作フロー



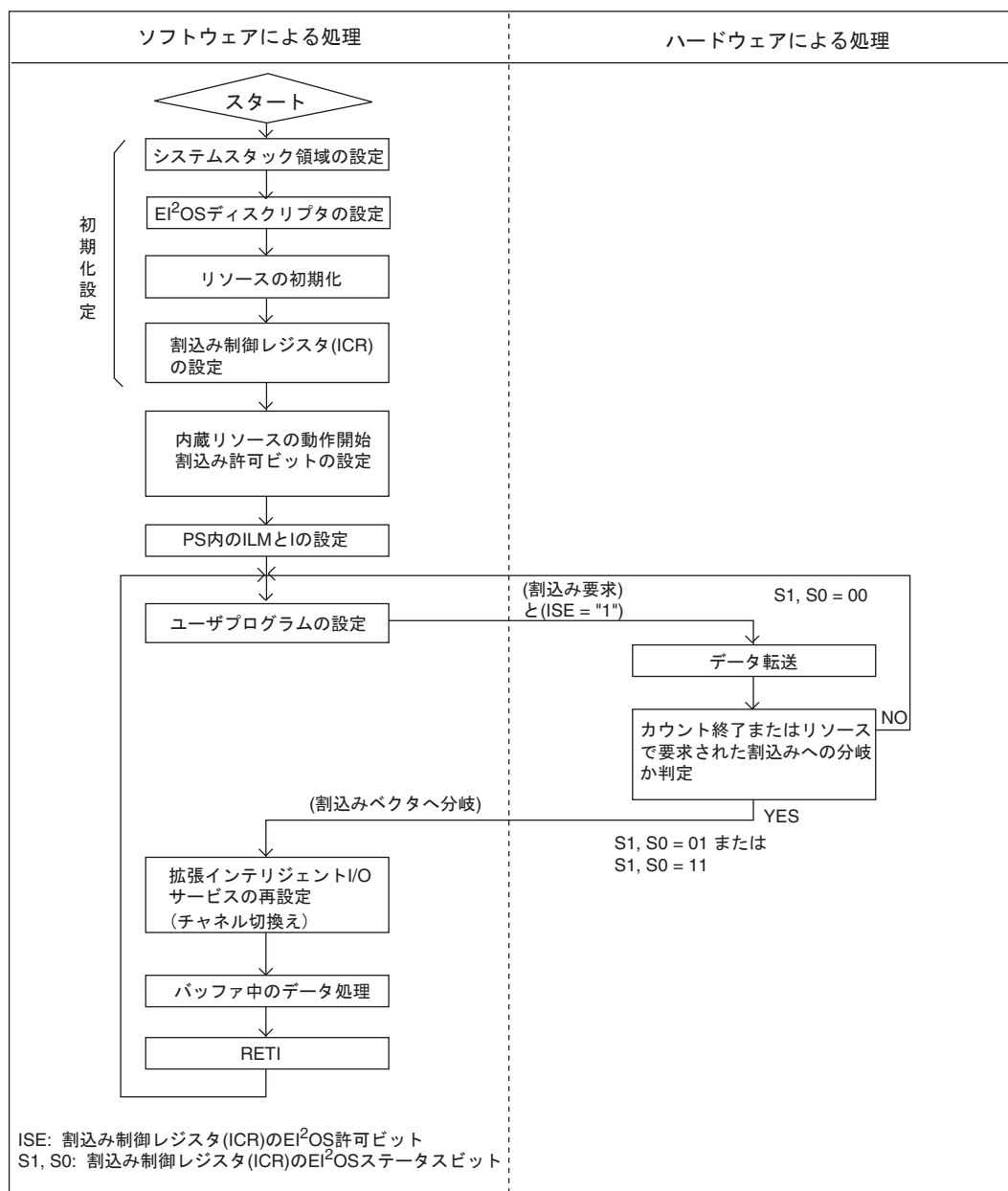
7.6.4 拡張インテリジェント I/O サービス (EI²OS) 使用手順

拡張インテリジェント I/O サービス (EI²OS) を使用する前に、システムスタック領域、拡張インテリジェント I/O サービス (EI²OS) ディスクリプタ、割り込み機能、および割り込み制御レジスタ (ICR) などの設定が必要です。

■ 拡張インテリジェント I/O サービス (EI²OS) の使用手順

図 7.6-8 に EI²OS ソフトウェアとハードウェア処理を示します。

図 7.6-8 拡張インテリジェント I/O サービス (EI²OS) 使用手順



7.6.5 拡張インテリジェント I/O サービス (EI²OS) 処理時間

拡張インテリジェント I/O サービス (EI²OS) の処理に要する時間は、以下の要素によって変化します。

- EI²OS ステータスレジスタ (ISCS) の設定
- I/O レジスタアドレスポインタ (IOA) の示すアドレス (領域)
- バッファアドレスポインタ (BAP) の示すアドレス (領域)
- 外部アクセス用の外部データバス幅
- 転送データのデータ長

EI²OS によるデータ転送が終了すると、ハードウェア割込みが起動するので、割込み処理時間が追加されます。

■ 拡張インテリジェント I/O サービス (EI²OS) 処理時間 (1 回の転送時間)

● データ転送を継続する場合

データ転送を継続する場合の EI²OS 処理時間を表 7.6-2 に示します。(EI²OS ステータスレジスタ (ISCS) 設定に基づいています)。

表 7.6-2 拡張インテリジェント I/O サービス実行時間

EI ² OS 終了制御ビット (SE) の設定		リソースからの終了要求により終了		リソースからの終了要求を無視	
IOA 更新 / 固定選択ビット (IF) の設定		固定	更新	固定	更新
BAP アドレス更新 / 固定選択ビット (BF) の設定	固定	32	34	33	35
	更新	34	36	35	37

単位：マシンサイクル (1 マシンサイクルは、マシンクロック (φ) の 1 クロック周期に相当します。)

表 7.6-3 に示すように、EI²OS 実行状態によっては補間する必要があります。

表 7.6-3 EI²OS 実行時間のデータ転送の補間値

I/O レジスタアドレスポインタ			内部アクセス		外部アクセス	
			B/ 偶数	奇数	B/ 偶数	8/ 奇数
バッファアドレスポインタ	内部アクセス	B/ 偶数	0	+2	+1	+4
		奇数	+2	+4	+3	+6
	外部アクセス	B/ 偶数	+1	+3	+2	+5
		8/ 奇数	+4	+6	+5	+8

B: バイトデータ転送

8: 外部バス幅 8 ビットでワード転送

偶数：偶数番号アドレスのワード転送

奇数：奇数番号アドレスのワード転送

● データカウンタ (DCT) のカウント終了時 (最終回のデータ転送時)

EI²OSによるデータ転送が終了するとハードウェア割込みが起動するため、割込み処理時間が加算されます。カウント終了時のEI²OSの処理時間は、以下の式で算出されます。

カウント終了時のEI²OS 処理時間 =

$$\text{データ転送時の EI}^2\text{OS 処理時間} + (21 + 6 \times Z) \text{ マシンサイクル}$$

割込み処理時間

割込み処理時間は、スタックポインタが指す各アドレスによって異なります。表 7.6-4 に割込み処理時間の補間値 (Z) を示します。

表 7.6-4 割込みハンドリング時間の補間値 (Z)

スタックポインタが指すアドレス	補間値 (Z)
外部 8 ビット	+4
偶数番号の外部アドレス	+1
奇数番号の外部アドレス	+4
偶数番号の内部アドレス	0
奇数番号の内部アドレス	+2

● リソース (I/O) からの終了要求によって終了する場合

リソース (I/O) からの終了要求による停止前にEI²OSによるデータ転送を終了した場合 (ICR : S1, S0 = 11_B), データ転送されず、ハードウェア割込みが起動します。EI²OS 処理時間は、以下の式で算出されます。式中の Z は、割込み処理時間の補間値を示します。(表 7.6-4)

$$\text{データ転送を終了させるために要する EI}^2\text{OS 処理時間} = 36 + 6 \times Z \text{ マシンサイクル}$$

< 参考 >

1 マシンサイクルは、マシクロック (φ) の 1 サイクルに相当します。

7.7 例外処理割込み

F²MC-16LX では、未定義命令を実行すると、その結果例外処理が発生します。

例外処理は、基本的には割込みと同じものであり、命令と命令の間で例外処理の発生が検出された場合、通常処理を中断して例外処理が実行されます。

一般的に、例外処理は予想外の動作の結果として発生しますので、デバッグ時や緊急時に必要となる復帰ソフトウェアを起動する場合のみに使用してください。

■ 例外処理

● 例外処理の動作

F²MC-16LX では、命令マップで未定義命令として定義されていないコードをすべて扱います。未定義命令を実行した場合、INT #10 ソフトウェア割込み命令に相当する処理が実行されます。

例外処理が割込みルーチンへ分岐する前に、以下の処理が行われます。

- A, DPR, ADB, DTB, PCB, PC, PS レジスタがシステムスタックへ退避されます。
- コンディションコードレジスタ (CCR) の I が "0" にクリアされ、ハードウェア割込みがマスクされます。
- コンディションコードレジスタ (CCR) のシステムフラグ (S) に "1" が設定され、システムスタックが起動されます。

システムスタックに退避されたプログラムカウンタ (PC) の値は、未定義命令が格納されるアドレスです。2 バイト以上の命令コードの場合、未定義として識別されたコードはこのアドレスに格納されます。例外処理ルーチン内で例外要因の種類を判定する必要がある場合は、この PC 値を使用してください。

● 例外処理からの復帰

RETI 命令で例外処理から制御が復帰すると、プログラムカウンタ (PC) が未定義命令を指しているため、例外処理が再開されます。ソフトウェアリセットを行うなどの対策を用いて解決してください。

7.8 割り込み処理のスタック動作

割り込みがいったん受け付けられると、割り込み処理へ分岐する前に専用レジスタの値が自動的にシステムスタックへ退避されます。割り込み処理が終了すると、専用レジスタの値はシステムスタックから自動的に復元されます。

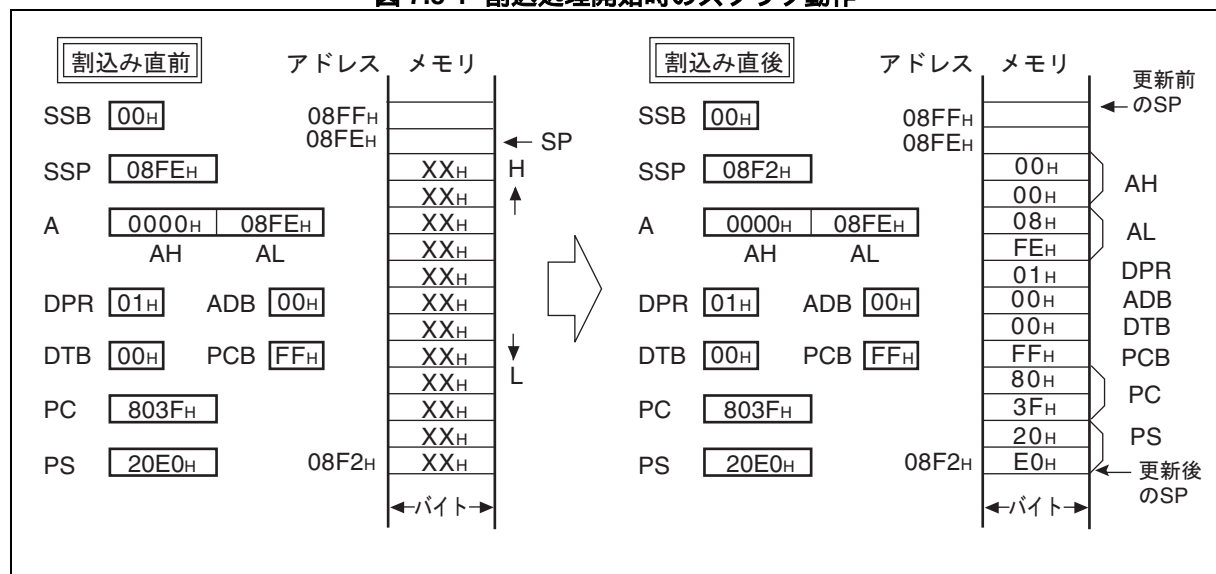
■ 割り込み処理開始時のスタック動作

割り込みがいったん受け付けられると、CPUは現在の専用レジスタの値を、以下に示す順番で自動的にシステムスタックに退避します。

- アキュムレータ (A)
- ダイレクトページレジスタ (DPR)
- アディショナルデータバンクレジスタ (ADB)
- データバンクレジスタ (DTB)
- プログラムバンクレジスタ (PCB)
- プログラムカウンタ (PC)
- プロセッサステータス (PS)

割り込み処理開始時のスタック動作を図 7.8-1 に示します。

図 7.8-1 割り込み処理開始時のスタック動作



■ 割り込み処理からの復帰時のスタック動作

割り込み処理終了時に割り込み復帰命令 (RETI) を実行すると、専用レジスタ値が退避時とは逆の順序 (PS, PC, PCB, DTB, ADB, DPR, A) でスタックから復帰します。専用レジスタは、割り込み開始直前の状態に戻ります。

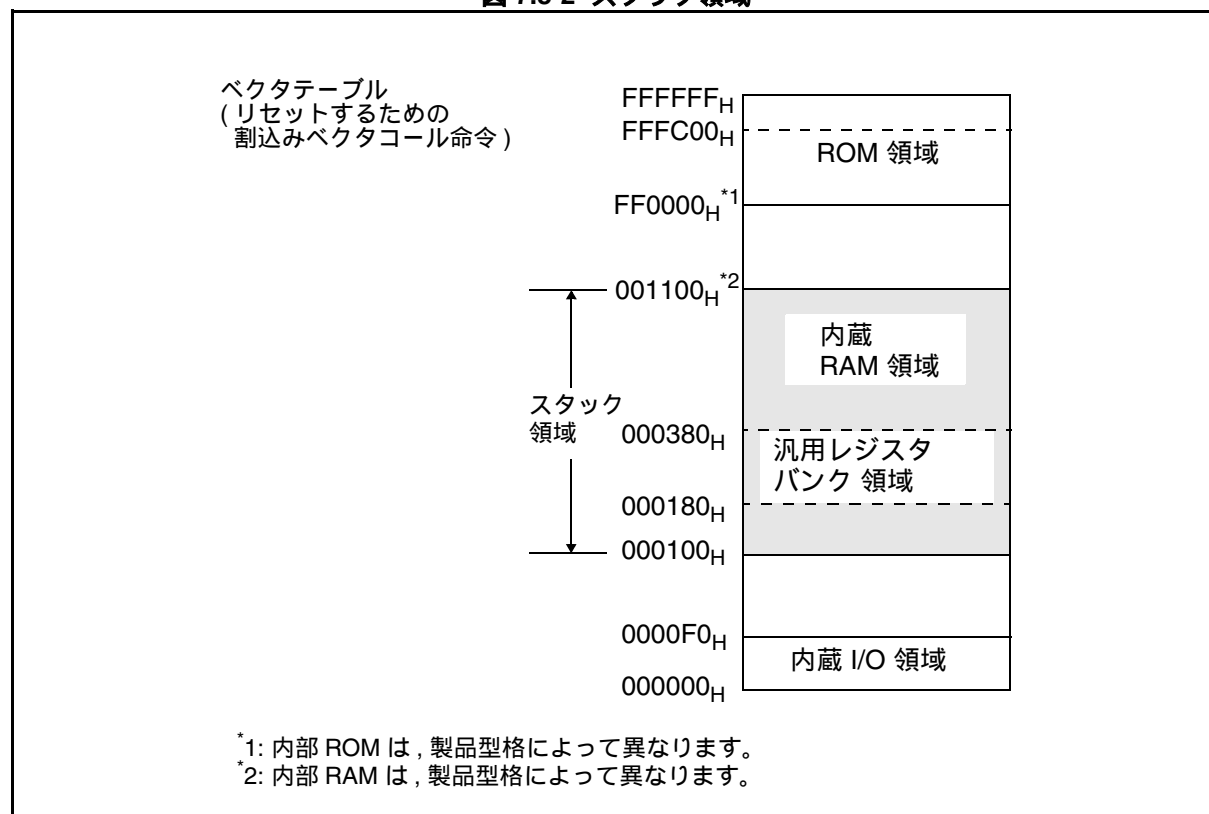
■ スタック領域

● スタック領域の割当て

スタック領域は、割込み処理以外にサブルーチンコール命令 (CALL) やベクタコール命令 (CALLV) の実行時にもプログラムカウンタ (PC) の退避や復元するために使用されます。スタック領域は、PUSHW 命令や POPW 命令でレジスタを一時的に退避または復元するためにも使用されます。

スタック領域は、RAM のデータ領域と一緒に割当てられます。図 7.8-2 にスタック領域を示します。

図 7.8-2 スタック領域



< 注意事項 >

- 一般的に、偶数番号のアドレスはスタックポインタ (SSP, USP) へ設定してください。
- オーバラップが発生しないようにシステムスタック領域、ユーザスタック領域、およびデータ領域を割当ててください。

● システムスタックとユーザスタック

システムスタック領域は、割込み処理に使用されます。割込みが発生すると、使用中のユーザスタック領域は強制的にシステムスタックに切換えられます。主にユーザスタック領域を使用するシステムであっても、システムスタック領域を正しく設定しなければなりません。

スタック空間を分割する必要がある場合は、システムスタックのみを使用してください。

第8章

モード設定

MB90820B シリーズにサポートされる動作モードとメモリアクセスモードについて説明します。

8.1 モード設定

8.2 モード端子 (MD2 ~ MD0)

8.3 モードデータ

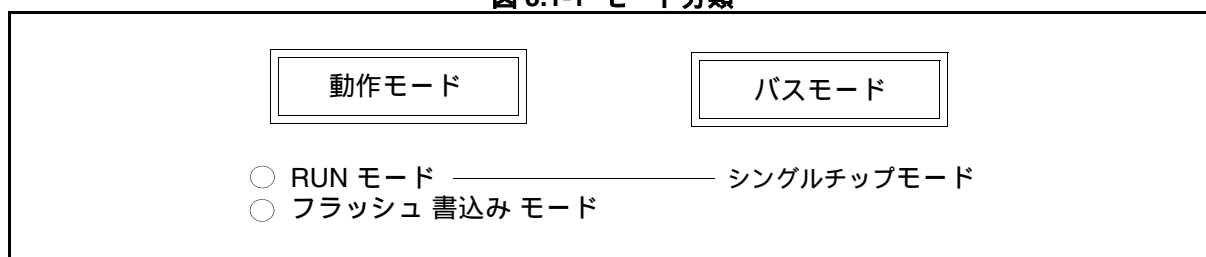
8.1 モード設定

F²MC-16LX には、アクセスメソッドおよびアクセス領域のモードをサポートしています。リセット時のモード端子とモードフェッチされたモードデータの設定に基づいてモードが設定されます。

■ モード設定

F²MC-16LX は、アクセスメソッドおよびアクセス領域のモードをサポートしており、本モジュールでは図 8.1-1 のような分類になっています。

図 8.1-1 モード分類



■ 動作モード

動作モードは、デバイスの動作状態を制御し、モード設定用端子 (MD_x) とモードデータ内の M_x ビットの値で指定されます。

● バスモード

バスモードは、内部 ROM の動作と外部アクセス機能の動作を制御するモードで、モード設定用端子 (MD_x) とモードデータ内の M_x ビットの値で指定されます。モード設定用端子 (MD_x) は、リセットベクタおよびモードデータを読み出すときのバスモードを指定するもので、モードデータ内の M_x ビットは、通常動作時のバスモードを指定するものです。

● RUN モード

RUN モードは、CPU 動作モードのことをいいます。RUN モードには、メインクロックモードや PLL クロックモードのほかに、各種の低消費電力モードがあります。詳細は、「第 6 章 低消費電力モード」を参照してください。

8.2 モード端子 (MD2 ~ MD0)

3 本の外部モード端子は、リセットベクタとモードデータを取込み方法を設定するのに使用します。

■ CPU モード端子 (MD2 ~ MD0)

モード端子で、リセットベクタの読出しを、外部データバスとするか内部データバスとするかの選択および外部データバス選択時のバス幅の選択を行います。

フラッシュメモリ内蔵品の場合は、内蔵フラッシュメモリにプログラムなどを書き込む間に使用され、フラッシュ書込みモードの指定もモード端子で行います。

モード端子の設定を表 8.2-1 に示します。

表 8.2-1 モード端子の設定

MD2	MD1	MD0	モード名	リセットベクタ アクセス領域	外部データ バス幅	備考
0	0	0	設定禁止			
0	0	1				
0	1	0				
0	1	1	内部ベクタモード	内部	モードデータ	リセットシーケンス以降はモードデータで制御
1	0	0	設定禁止			
1	0	1				
1	1	0	フラッシュシリアル書込みモード*	-	-	-
1	1	1	フラッシュメモリモード	-	-	パラレルライタ使用時のモードです。

0 = V_{SS}, 1 = V_{CC} としてください。

*: フラッシュシリアル書込みは、モード端子の設定だけでは書込みが行えません。ほかの端子の設定も必要です。

フラッシュシリアル書込みについての詳細は、「付録 B」を参照してください。

8.3 モードデータ

モードデータは, "FFFFDF_H" 番地のメモリ上にあり, リセットシーケンス後の動作を指定します。モードデータは, モードフェッチで CPU に自動的に取り込まれます。

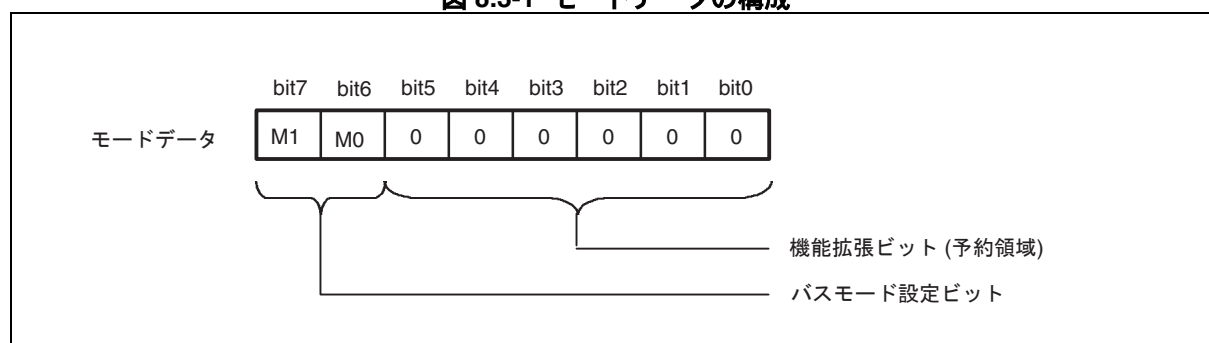
■ モードデータ

リセットシーケンス実行中に, "FFFFDF_H" 番地のモードデータを CPU 内のモードデータレジスタに取り込みます。CPU は, モードデータでメモリアクセスモードを設定します。

- モードデータレジスタの内容を変更できるのは, リセットシーケンスだけです。
- モードデータで設定されたメモリアクセスモードは, リセットシーケンス終了後に有効となります。

図 8.3-1 にモードデータの構成を示します。

図 8.3-1 モードデータの構成



■ バスモード設定ビット

リセットシーケンス終了後の動作モードの指定を行うビットです。各ビットと機能の関係は, 表 8.3-1 のようになっています。

表 8.3-1 バスモード設定ビットとその機能

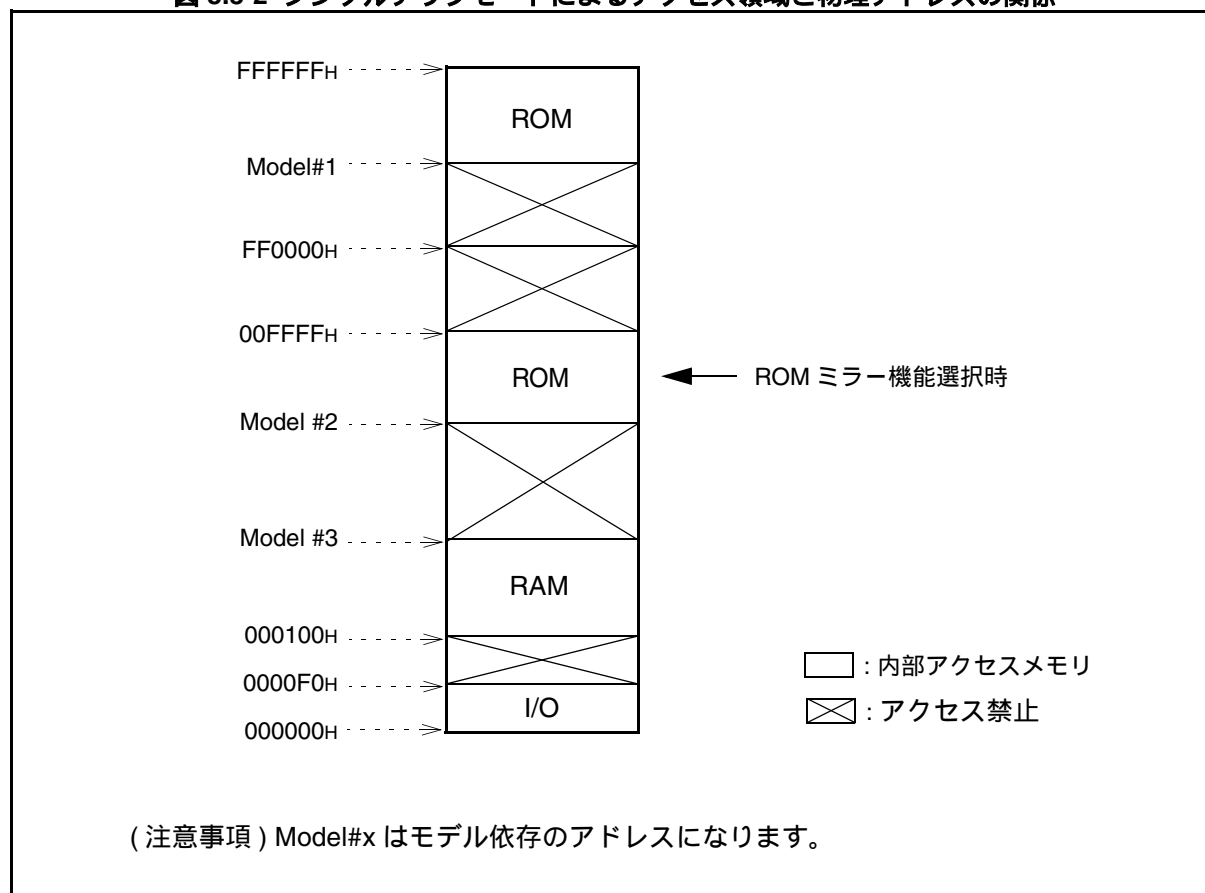
M1	M0	機能
0	0	シングルチップモード
0	1	(設定禁止)
1	0	
1	1	

< 注意事項 >

MB90820B シリーズでは, シングルチップモードのみでの使用となりますので, MD2, MD1, MD0 は "011_B" に, M1, M0 は "00_B" に設定してください。

シングルチップモード時のアクセス領域と物理アドレスの対応を図 8.3-2 に示します。

図 8.3-2 シングルチップモードによるアクセス領域と物理アドレスの関係



■ モード端子とモードデータの関係

表 8.3-2 にモード端子とモードデータの関係を示します。

表 8.3-2 モード端子とモードデータの関係

モード	モード端子			モードデータ	
	MD2	MD1	MD0	M1	M0
シングルチップモード	0	1	1	0	0

< 注意事項 >

MB90820B シリーズでは、シングルチップモードのみでの使用となります。

第9章

I/O ポート

I/O ポートの機能と動作について説明します。

- 9.1 I/O ポートの概要
- 9.2 I/O ポートのレジスタ
- 9.3 ポート 0
- 9.4 ポート 1
- 9.5 ポート 2
- 9.6 ポート 3
- 9.7 ポート 4
- 9.8 ポート 5
- 9.9 ポート 6
- 9.10 ポート 7
- 9.11 ポート 8

9.1 I/O ポートの概要

I/O ポートは、すべて 汎用入出力ポート (パラレル入出力ポート) として使用できます。MB90820B シリーズでは 9 ポート (66 本) あります。これらのポートは、リソース入出力端子 (周辺機能の入出力端子) と兼用になっています。

■ 入出力ポート機能

各入出力ポートは、ポートデータレジスタ (PDR) を介して CPU からのデータを I/O 端子に出力したり、I/O 端子から CPU に信号を取り込んだりします。ポート方向レジスタ (DDR) によって、各 I/O 端子のデータの入出力方向をビット単位で指定できます。

各ポートの機能と兼用されるリソース I/O を以下に示します。

- ポート 0: 汎用入出力ポート / リソース (PWC)
- ポート 1: 汎用入出力ポート / リソース (DTP/ 多機能タイマ)
- ポート 2: 汎用入出力ポート / リソース (16 ビットリロードタイマ)
- ポート 3: 汎用入出力ポート / リソース (16 ビット PPG タイマ)
- ポート 4: 汎用入出力ポート / リソース (16 ビット PPG タイマ /16 ビットリロードタイマ /UART/PWC)
- ポート 5: 汎用入出力ポート / リソース (16 ビット PPG タイマ /DTP)
- ポート 6: 汎用入出力ポート / リソース (8/10 ビット A/D コンバータ)
- ポート 7: 汎用入出力ポート / リソース (8/10 ビット A/D コンバータ /8 ビット D/A コンバータ /UART/16ビットフリーランタイマ/16ビット入力キャプチャ)
- ポート 8: 汎用入出力ポート / リソース (16 ビット入力キャプチャ / 多機能タイマ)

表 9.1-1 に各ポートの機能一覧を示します。

表 9.1-1 ポートの機能一覧

ポート	端子	入力形式	出力形式	機能	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
ポート0	P00~P07/ PWO0	CMOS (ヒステリシス)	CMOS プルアップ 抵抗を 選択可	汎用I/Oポート	—	—	—	—	—	—	—	—	P07	P06	P05	P04	P03	P02	P01	P00	
				リソース	—	—	—	—	—	—	—	—	PWO0	PWI0	—	—	—	—	—	—	
ポート1	P10/INT0/ DTT1~ P17			汎用I/Oポート	P17	P16	P15	P14	P13	P12	P11	P10	—	—	—	—	—	—	—	—	
				リソース	—	INT6	INT5	INT4	INT3	INT2	INT1	INT0 DTT1	—	—	—	—	—	—	—	—	
ポート2	P20/ TIN1~P27	CMOS		汎用I/Oポート	—	—	—	—	—	—	—	—	—	P27	P26	P25	P24	P23	P22	P21	P20
				リソース	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TO1	TIN1	
ポート3	P30~P37/ PPG0			汎用I/Oポート	P37	P36	P35	P34	P33	P32	P31	P30	—	—	—	—	—	—	—	—	
				リソース	PPG0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
ポート4	P40/ PPG1~ P47/PWO1	CMOS (ヒステリシス)	CMOS	汎用I/Oポート	—	—	—	—	—	—	—	—	P47	P46	P45	P44	P43	P42	P41	P40	
				アナログ出力	—	—	—	—	—	—	—	—	PWO1	PWI1	SIN0*	SOT0	SCK0	TO0	TIN0	PPG1	
ポート5	P50/ PPG2~ P51/INT7			汎用I/Oポート	—	—	—	—	—	—	P51	P50	—	—	—	—	—	—	—	—	
				リソース	—	—	—	—	—	—	INT7	PPG2	—	—	—	—	—	—	—	—	
ポート6	P60/ AN0~P67/ AN7	CMOS		汎用I/Oポート	—	—	—	—	—	—	—	—	P67	P66	P65	P64	P63	P62	P61	P60	
				アナログ入力	—	—	—	—	—	—	—	—	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0	
ポート7	P70/DA0/ AN8 ~ P77/IN1/ AN15			汎用I/Oポート	P77	P76	P75	P74	P73	P72	P71	P70	—	—	—	—	—	—	—	—	
				リソース	IN1	IN0	FRCK	SCK1	SOT1	SIN1*	—	—	—	—	—	—	—	—	—	—	
		アナログ出力		—	—	—	—	—	—	DA1	DA0	—	—	—	—	—	—	—	—		
		アナログ入力		AN15	AN14	AN13	AN12	AN11	AN10	AN9	AN8	—	—	—	—	—	—	—	—		
ポート8	P80/ IN2~P87/ RTO5		汎用I/Oポート	—	—	—	—	—	—	—	—	P87	P86	P85	P84	P83	P82	P81	P80		
			リソース	—	—	—	—	—	—	—	—	RTO5	RTO4	RTO3	RTO2	RTO1	RTO0	IN3	IN2		

* : UART0/UART1 データ入力端子 SIN0 と SIN1 は , ユーザプログラムにより CMOS 入力として選択可能です。

< 注意事項 >

ポート 6 およびポート 7 は , アナログ入力端子と兼用になっています。汎用入出力ポートとして使用する場合は , 必ずアナログ入力許可レジスタ (ADER0/ADER1) の対応するビットを "0" に設定してください。MCU のリセットによって , ADER0/ADER1 レジスタのビットは "1" に設定されます。

9.2 I/O ポートのレジスタ

I/O ポートの設定に関連するレジスタの一覧を示します。

■ I/O ポートのレジスタ一覧

表 9.2-1 に各ポートに対応するレジスタの一覧を示します。

表 9.2-1 各ポートのレジスタ一覧

レジスタ名	リードライト	アドレス	初期値
ポート 0 データレジスタ (PDR0)	R/W	000000 _H	XXXXXXXX _B
ポート 1 データレジスタ (PDR1)	R/W	000001 _H	XXXXXXXX _B
ポート 2 データレジスタ (PDR2)	R/W	000002 _H	XXXXXXXX _B
ポート 3 データレジスタ (PDR3)	R/W	000003 _H	XXXXXXXX _B
ポート 4 データレジスタ (PDR4)	R/W	000004 _H	XXXXXXXX _B
ポート 5 データレジスタ (PDR5)	R/W	000005 _H	XXXXXXXX _B
ポート 6 データレジスタ (PDR6)	R/W	000006 _H	XXXXXXXX _B
ポート 7 データレジスタ (PDR7)	R/W	000007 _H	XXXXXXXX _B
ポート 8 データレジスタ (PDR8)	R/W	000008 _H	XXXXXXXX _B
ポート 0 データ方向レジスタ (DDR0)	R/W	000010 _H	00000000 _B
ポート 1 データ方向レジスタ (DDR1)	R/W	000011 _H	00000000 _B
ポート 2 データ方向レジスタ (DDR2)	R/W	000012 _H	00000000 _B
ポート 3 データ方向レジスタ (DDR3)	R/W	000013 _H	00000000 _B
ポート 4 データ方向レジスタ (DDR4)	R/W	000014 _H	00000000 _B
ポート 5 データ方向レジスタ (DDR5)	R/W	000015 _H	XXXXXX00 _B
ポート 6 データ方向レジスタ (DDR6)	R/W	000016 _H	00000000 _B
ポート 7 データ方向レジスタ (DDR7)	R/W	000017 _H	00000000 _B
ポート 8 データ方向レジスタ (DDR8)	R/W	000018 _H	00000000 _B
アナログ入力許可レジスタ 0(ADER0)	R/W	0000C5 _H	11111111 _B
アナログ入力許可レジスタ 1(ADER1)	R/W	0000D0 _H	11111111 _B
ポート 0 プルアップ抵抗設定レジスタ (RDR0)	R/W	00008C _H	00000000 _B
ポート 1 プルアップ抵抗設定レジスタ (RDR1)	R/W	00008D _H	00000000 _B
ポート 2 プルアップ抵抗設定レジスタ (RDR2)	R/W	00008E _H	00000000 _B
ポート 3 プルアップ抵抗設定レジスタ (RDR3)	R/W	00008F _H	00000000 _B

R/W: リード / ライト可能

X: 不定

9.3 ポート 0

ポート 0 は汎用入出力ポートで、リソース入出力と兼用となっています。各ポートの端子は、入出力ポートとリソース入出力との間で切換えることができます。ここでは、汎用入出力ポートの機能を中心に、ポート 0 の構成、端子の一覧と端子のブロックダイアグラム、また対応するレジスタについても説明します。

■ ポート 0 の構成

ポート 0 は、以下から構成されています。

- 汎用入出力端子 / リソース入出力端子 (P00 ~ P07/PWO0)
- ポート 0 データレジスタ (PDR0)
- ポート 0 データ方向レジスタ (DDR0)
- ポート 0 プルアップ抵抗設定レジスタ (RDR0)

■ ポート 0 の端子配列

ポート 0 入出力端子はリソース入出力端子としても使用されるため、リソース入出力端子として使用の場合は、汎用入出力ポート端子として使用できません。

表 9.3-1 にポート 0 の端子一覧を示します。

表 9.3-1 ポート 0 の端子

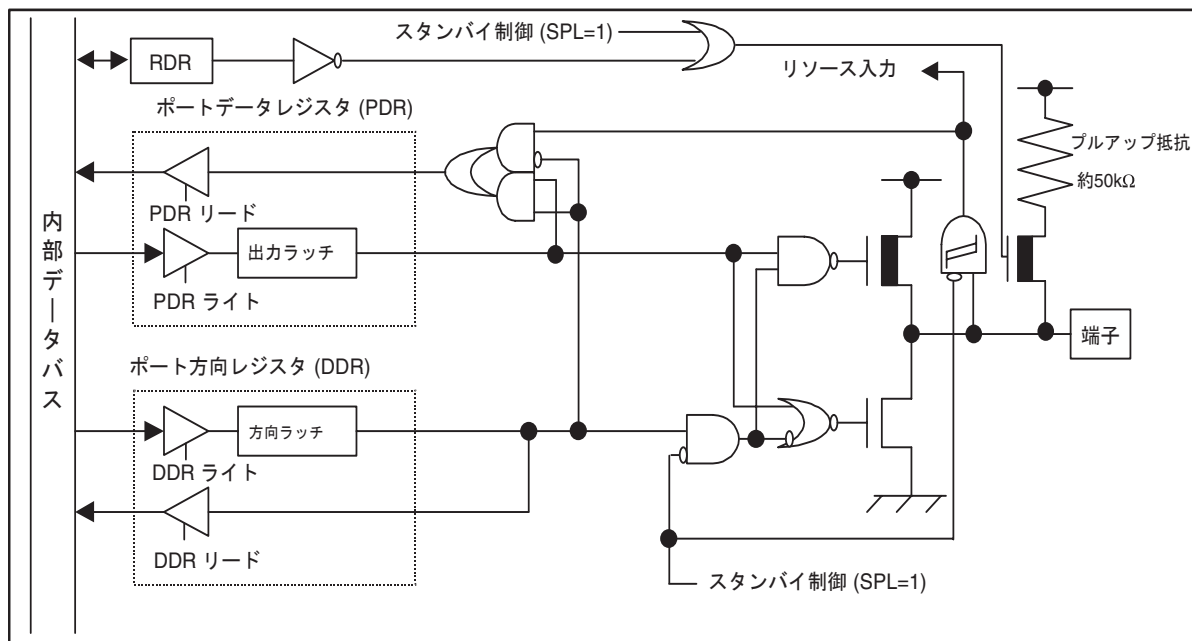
ポート	端子名	ポート機能 (シングルチップモード)		リソース機能		入出力形式		入出力 回路 形式
						入力	出力	
ポート 0	P00	P00	汎用入出力	-	-	CMOS (ヒステリシス)	CMOS	C
	P01	P01		-	-			
	P02	P02		-	-			
	P03	P03		-	-			
	P04	P04		-	-			
	P05	P05		-	-			
	P06/PWI0	P06		PWI0	PWC0 入力			
	P07/PWO0	P07		PWO0	PWC0 出力			

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート 0 の端子のブロックダイヤグラム

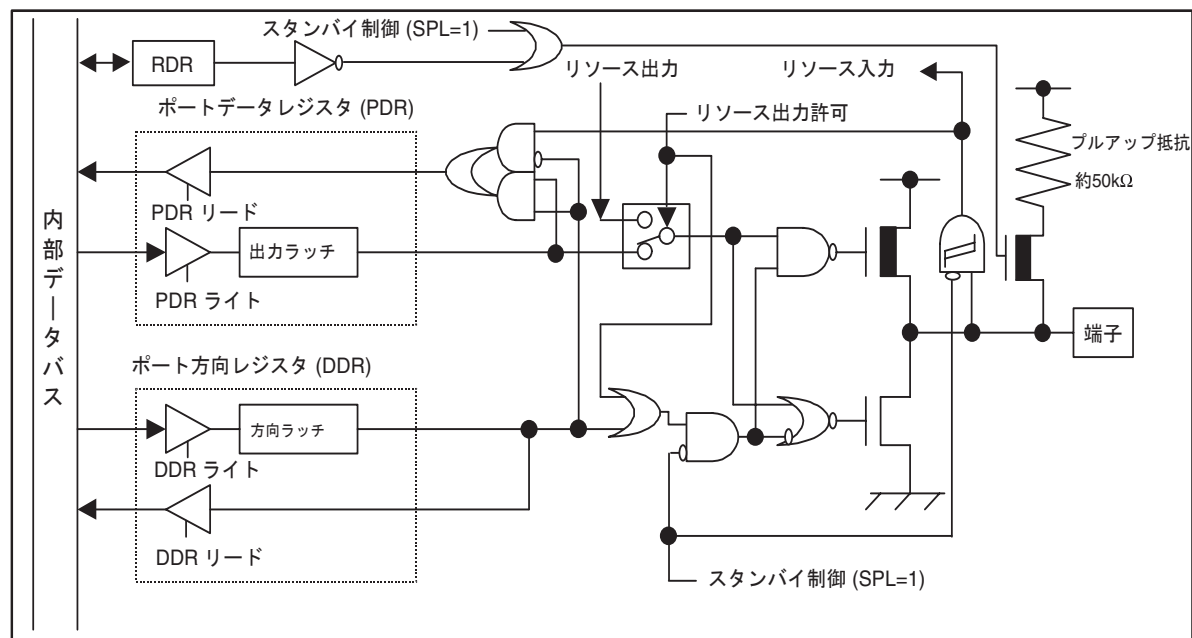
P00 ~ P06/PW10 端子のブロックダイヤグラムを図 9.3-1 に示します。

図 9.3-1 P00 ~ P06/PW10 の端子のブロックダイヤグラム



P07/PW00 端子のブロックダイヤグラムを図 9.3-2 に示します。

図 9.3-2 P07/PW00 端子のブロックダイヤグラム



リソース出力許可ビットが設定された場合、DDR0レジスタの値に関係なくポートは強制的にリソース出力として機能します。

■ ポート 0 のレジスタ

ポート 0 のレジスタには、PDR0, DDR0 および RDR0 があります。各レジスタを構成しているビットは、ポート 0 の端子に 1 対 1 で対応しています。

表 9.3-2 にポート 0 の端子との対応するレジスタビットを示します。

表 9.3-2 ポート 0 の端子と対応レジスタビット

ポート名	関連するレジスタのビットと対応する端子								
ポート 0	PDR0, DDR0, RDR0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	対応する端子	P07	P06	P05	P04	P03	P02	P01	P00

入出力回路形式については「1.7 入出力回路形式」を参照してください。

9.3.1 ポート 0 のレジスタ (PDR0, DDR0, RDR0)

ポート 0 のレジスタについて説明します。

■ ポート 0 のレジスタの機能

- ポート 0 データレジスタ (PDR0)

ポート 0 データレジスタは、ポート 0 の各端子の状態を示します。

- ポート 0 データ方向レジスタ (DDR0)

DDR0 レジスタは、ポート 0 の各端子（ビット）のデータ入出力方向を指定します。DDR0 レジスタのいずれかのビットが "1" の場合は、対応するポート（端子）は出力ポートとして設定されます。いずれかのビットが "0" の場合は、ポート（端子）は入力ポートとして設定されます。

< 注意事項 >

- 出力端子と兼用しているリソースを使用する場合は、その端子に対応するリソース出力許可ビットが許可に設定されている限り、DDR0 レジスタの値とは無関係にポートはリソース出力端子として機能します。
- 入力端子と兼用しているリソースを使用するには、各リソース入力端子に対応する DDR0 レジスタのビットに "0" を設定しポートを入力モードにしてください。ポート 0 レジスタの機能を表 9.3-3 に示します。

< 参考 >

MCU がリセットされた場合、DDR0 レジスタは、汎用 I/O ポート入力用に "0" にクリアされます。

- ポート 0 プルアップ抵抗設定レジスタ (RDR0)

RDR0 レジスタは、ポート 0 の各端子（ビット）のプルアップ抵抗を選択します。RDR0 レジスタのいずれかのビットが "1" の場合、対応するポート（端子）にプルアップ抵抗が選択され、ビットが "0" の場合は、プルアップ抵抗が切断されます。

表 9.3-3 にポート 0 のレジスタの機能を示します。

表 9.3-3 ポート 0 のレジスタの機能

レジスタ名	データ	読出し時	書込み時	リード / ライト	アドレス	初期値
ポート 0 データレジスタ (PDR0)	0	端子は "L" レベル	出力ラッチに "0" をロードし、 端子が出力ポートとして機能 する場合は、端子に "L" レベ ルが設定されます	R/W	000000 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチに "1" をロードし、 端子が出力ポートとして機能 する場合は、端子に "H" レベ ルが設定されます			
ポート 0 データ方向レジ スタ (DDR0)	0	方向ラッチ は "0"	出力バッファが " オフ " にな り、ポートが入力モードにな ります	R/W	000010 _H	00000000 _B
	1	方向ラッチ は "1"	出力バッファが " オン " にな り、ポートが出力モードにな ります			
ポート 0 プルアップ抵抗 設定レジスタ (RDR0)	0	設定ラッチ は "0"	プルアップ抵抗が遮断され、 入力モードにてポートは Hi-Z 状態になります	R/W	00008C _H	00000000 _B
	1	設定ラッチ は "1"	プルアップ抵抗が設定され、 入力モードにてポートは "H" レベルを保持します			

R/W: リード / ライト可能

X: 不定

9.3.2 ポート 0 の動作説明

ポート 0 の動作について説明します。

■ ポート 0 の動作

● 出力モード時のポート動作

- DDR0 レジスタのいずれかのビットに "1" を設定すると、対応するポート端子は出力モードになります。
- 出力モード時に PDR0 レジスタに書き込まれたデータは、PDR の出力ラッチに保持され、そのまま端子に出力されます。
- PDR0 レジスタを読み出すと、ポート端子における値 (PDR の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト系命令 (ビット設定命令など) を PDR0 に対して使用すると、ターゲットビットには書き込んだ値が設定されます。DDR レジスタによって出力に設定されているビットは影響を受けませんが、入力に設定されているビットの場合は、端子からの入力値は出力ラッチに書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切替える前に、出力データを PDR レジスタに書き込み、DDR レジスタを出力モードに設定してください。

● 入力ポートの動作

- DDR0 レジスタのビットに "0" を設定すると、対応するポート端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- ただし、RDR0 レジスタの対応するビットに "1" を設定しプルアップ抵抗を選択すると、端子は "H" レベルを保持します。
- 入力モード時において PDR0 レジスタに書き込まれたデータは、PDR の出力ラッチにて保持されますが、端子には出力されません。
- PDR0 レジスタを読み出すと、端子のレベル値 ("0" または "1") を読み出せます。

● リソース出力としてのポートの動作

リソース出力許可ビットを "許可" にすると、ポートはリソース出力として使用できません。入力と出力の切換えを指定する場合、リソース許可ビットの状態が優先されます。すなわち、たとえ DDR0 レジスタのいずれかのビットが "0" であっても、リソース出力が許可になっている場合は、そのビットに対応する端子はリソース出力として使用されます。リソース出力が許可であっても端子の値を読み出すことはできるので、リソースの出力値を読み出すことができます。

● リソース入力としてポートの動作

ポートをリソース入力として使用する場合、端子の値は常にリソースの入力値となります。リソースの外部信号として使用する場合は、DDR0 レジスタの対応するビットに "0" を設定しポートを入力モードにしてください。

● リセット後のポート動作

- MCU がリセットされた場合、DDR0 および PDR0 レジスタは "0" に初期化されます。その結果、出力バッファが " オフ " になり (I/O モードは " 入力 " に変化)、プルアップ抵抗が遮断され、端子はハイインピーダンス状態になります。
- PDR0 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モードで使用するには、出力データを PDR0 レジスタに設定した後に、出力モードで DDR0 レジスタを設定しなければなりません。

● ストップモードまたはタイムベースタイマモード時のポート動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、ポート端子はハイインピーダンス状態になります。これは、出力バッファが DDR0 レジスタの値とは無関係に "OFF" されるからです。

なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。ポート 0 の端子状態を表 9.3-4 に示します。

表 9.3-4 ポート 0 の端子状態

端子	通常動作	スリープモード	ストップモードか タイムベースタイマ モード (SPL=0)	ストップモードか タイムベースタイマ モード (SPL=1, RDR=0)	ストップモードか タイムベースタイマ モード (SPL=1, RDR=1)
P00 ~ P07/ PWO0	汎用 I/O ポート	汎用 I/O ポート	汎用 I/O ポート	入力禁止 / 出力は Hi-Z 状態	入力禁止 / "H" レベル で保持

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)

Hi-Z: ハイインピーダンス

9.4 ポート 1

ポート 1 は汎用入出力ポートですが、リソース入力と兼用しています。各ポートの端子は、汎用ポートとリソース入力を切換えることができます。本節では、汎用入出力ポート機能を中心に、ポート 1 の構成、端子一覧、端子のブロックダイアグラムを示し、対応するレジスタについて説明します。

■ ポート 1 の構成

ポート 1 は、以下から構成されています。

- 汎用入出力端子 / リソース入力端子 (P10/INT0/DTTI ~ P17)
- ポート 1 データレジスタ (PDR1)
- ポート 1 データ方向レジスタ (DDR1)
- ポート 1 プルアップ抵抗設定レジスタ (RDR1)

■ ポート 1 の端子

ポート 1 入出力端子は、リソースの入力端子としても使用します。したがって、これらの端子がリソース入力端子として使用された場合、汎用入出力ポートとしては使用できません。

表 9.4-1 にポート 1 の端子一覧を示します。

表 9.4-1 ポート 1 の端子

ポート	端子	ポート機能		リソース機能		入出力形式		入出力回路形式
						入力	出力	
ポート 1	P10/INT0/DTTI	P10	汎用入出力	INT0/DTTI	外部割込み入力 / 波形ジェネレータ入力	CMOS (ヒステリシス)	CMOS	D
	P11/INT1	P11		INT1	外部割込み入力			
	P12/INT2	P12		INT2	外部割込み入力			
	P13/INT3	P13		INT3	外部割込み入力			
	P14/INT4	P14		INT4	外部割込み入力			
	P15/INT5	P15		INT5	外部割込み入力			
	P16/INT6	P16		INT6	外部割込み入力			
	P17	P17		-	-			

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート1の端子のブロックダイアグラム

図 9.4-1 に P10/INT0/DTTI ~ P16/INT6 端子のブロックダイアグラムを示します。

図 9.4-1 P10/INT0/DTTI ~ P16/INT6 端子のブロックダイアグラム

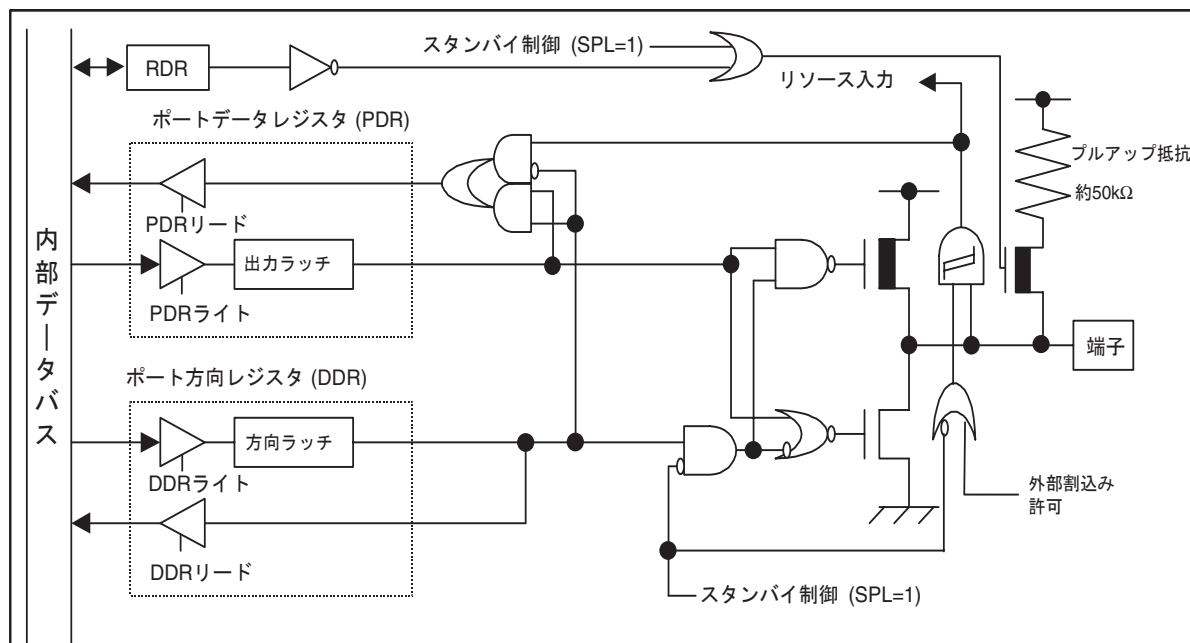
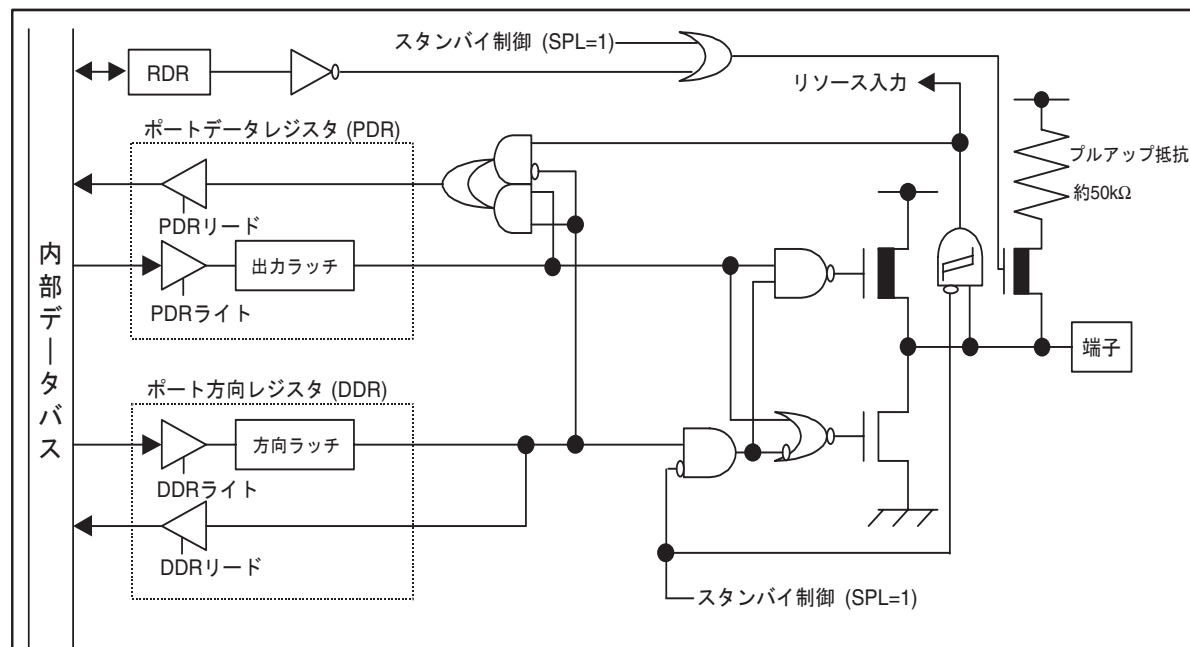


図 9.4-2 に P17 端子のブロックダイアグラムを示します。

図 9.4-2 P17 端子のブロックダイアグラム



■ ポート1のレジスタ

ポート1のレジスタには、PDR1, DDR1 および RDR1 があります。各レジスタを構成するビットは、ポート1の端子に1対1で対応しています。

表 9.4-2 にポート1の端子と対応するレジスタビットを示します。

表 9.4-2 ポート1の端子と対応するレジスタビット

ポート名	レジスタビットと対応するポート端子								
ポート1	PDR1, DDR1, RDR1	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	対応する端子	P17	P16	P15	P14	P13	P12	P11	P10

9.4.1 ポート1のレジスタ (PDR1, DDR1, RDR1)

ポート1のレジスタについて説明します。

■ ポート1のレジスタの機能

● ポート1データレジスタ (PDR1)

ポート1データレジスタは、ポート1の各端子の状態を示します。

● ポート1データ方向レジスタ (DDR1)

DDR1レジスタは、ポート1の各端子（ビット）のデータ入出力方向を指定します。DDR1レジスタのいずれかのビットが"1"の場合、そのビットに対応するポート（端子）は出力ポートとして設定されます。ビットが"0"の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

リソースの入力と兼用している端子を使用する場合は、各入力端子に対応するDDR1レジスタのビットに"0"を設定してポートを入力モードにしてください。

MCU がリセットされた場合は、DDR1レジスタは"0"にクリアされ汎用入出力ポートとなります。

● ポート1プルアップ抵抗設定レジスタ (RDR1)

RDR1レジスタは、ポート1の各端子（ビット）のプルアップ抵抗を選択します。RDR1レジスタのいずれかのビットが"1"の場合、対応するポート（端子）にプルアップ抵抗が選択され、ビットが"0"の場合は、プルアップ抵抗が切断されます。

表 9.4-3 にポート1のレジスタの機能を示します。

表 9.4-3 ポート1のレジスタの機能 (1 / 2)

レジスタ名	データ	読み出し時	書き込み時	リード ライト	アドレス	初期値
ポート1 データレジスタ (PDR1)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、 端子が出力ポートとして機能すると、 端子は "L" レベルに設定されます	R/W	000001 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチへ "1" がロードされ、 端子が出力ポートとして機能すると、 端子は "H" レベルに設定されます			
ポート1 データ方向レジスタ (DDR1)	0	方向ラッチは "0"	出力バッファを "OFF" し、ポートが 入力モードになります	R/W	000011 _H	00000000 _B
	1	方向ラッチは "1"	出力バッファを "ON" し、ポートが 出力モードになります			

表 9.4-3 ポート 1 のレジスタの機能 (2 / 2)

レジスタ名	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート 1 プルアップ抵抗 設定レジスタ (RDR1)	0	設定ラッチは "0"	プルアップ抵抗が遮断され、入 力モードでポートが Hi-Z 状態 になります	R/W	00008D _H	00000000 _B
	1	設定ラッチは "1"	プルアップ抵抗が選択され、入 力モードでポートが H レベル で保持されます			

R/W: リード / ライト可能

X: 不定

9.4.2 ポート 1 の動作説明

ポート 1 の動作について説明します。

■ ポート 1 の動作

● 出力モード時のポート動作

- DDR1 レジスタのいずれかのビットに "1" を設定すると、ビットに対応するポートの端子は出力モードになります。
- 出力モードにおいて PDR1 レジスタに書き込まれたデータは、PDR1 の出力ラッチにおいて保持され、そのまま端子へ出力されます。
- PDR1 レジスタを読み出すと、ポート端子における値 (PDR1 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力ポートの動作

- DDR1 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- ただし、RDR1 レジスタのいずれかのビットに "1" を設定してプルアップ抵抗を選択すると、端子は "H" レベルで保持されます。
- 入力モードで PDR1 レジスタに書き込まれたデータは、PDR1 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR1 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● リソース出力としてのポート動作

ポートをリソース入力として使用する場合、端子の値が常にリソースの入力値となります。リソース用に外部信号を使用するには、DDR1 レジスタの対応するビットに "0" を設定し、ポートを入力モードにしてください。

● リセット後のポート動作

- MCU がリセットされた場合、DDR1 レジスタは "0" に初期化されます。その結果、出力バッファが "OFF" になり (入出力モードは、入力に変化), 端子はハイインピーダンス状態になります。

- PDR1 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モード時に使用する場合は、出力データを PDR1 レジスタに設定した後、出力モードは DDR1 レジスタにおいて設定しなければなりません。

● ストップモードまたはタイムベースタイマモード時のポート動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、端子はハイインピーダンス状態になります。これは、出力バッファが DDR1 レジスタの値とは無関係に "OFF" されるからです。なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。

表 9.4-4 にポート 1 の端子状態を示します。

表 9.4-4 ポート 1 の端子状態

端子	通常動作	スリープモード	ストップモード、 タイムベースタイ マモード (SPL = 0)	ストップモード、 タイムベースタイ マモード (SPL = 1, RDR = 0)	ストップモード、タイ ムベースタイマモード (SPL = 1, RDR = 1)
P10/INT0/ DTTI ~ P16/INT6	汎用 I/O ポート	汎用 I/O ポート	汎用 I/O ポート	入力許可 */ 出力 は Hi-Z 状態	入力許可 */ "H" レベル で保持
P17	汎用 I/O ポート	汎用 I/O ポート	汎用 I/O ポート	入力禁止 / 出力は Hi-Z 状態	入力禁止 / "H" レベル で保持

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)

Hi-Z: ハイインピーダンス

*: 入力は、P10/INT0/DTTI ~ P16/INT6 を外部割込み端子として構成した場合のみ有効になり、それ以外の場合は遮断されます。

9.5 ポート 2

ポート 2 は汎用入出力ポートですが、リソース入出力と兼用しています。各ポートの端子は、入出力ポートとリソース入出力の間で切換えることができます。本節では、汎用入出力ポート機能を中心に、ポート 2 の構成と端子一覧、端子のブロックダイヤグラムを示し、対応するレジスタについて説明します。

■ ポート 2 の構成

ポート 2 は、以下から構成されています。

- ・ 汎用入出力ポート / リソース入出力端子 (P20/TIN1 ~ P27)
- ・ ポート 2 データレジスタ (PDR2)
- ・ ポート 2 データ方向レジスタ (DDR2)
- ・ ポート 2 プルアップ抵抗設定レジスタ (RDR2)

■ ポート 2 の端子

ポート 2 入出力 端子は、リソースの入出力端子としても使用します。したがって、これらの端子がリソースの入出力端子として使用されているときは、汎用入出力ポートとしては使用できません。ポート 2 の端子一覧を表 9.5-1 に示します。

表 9.5-1 ポート 2 の端子

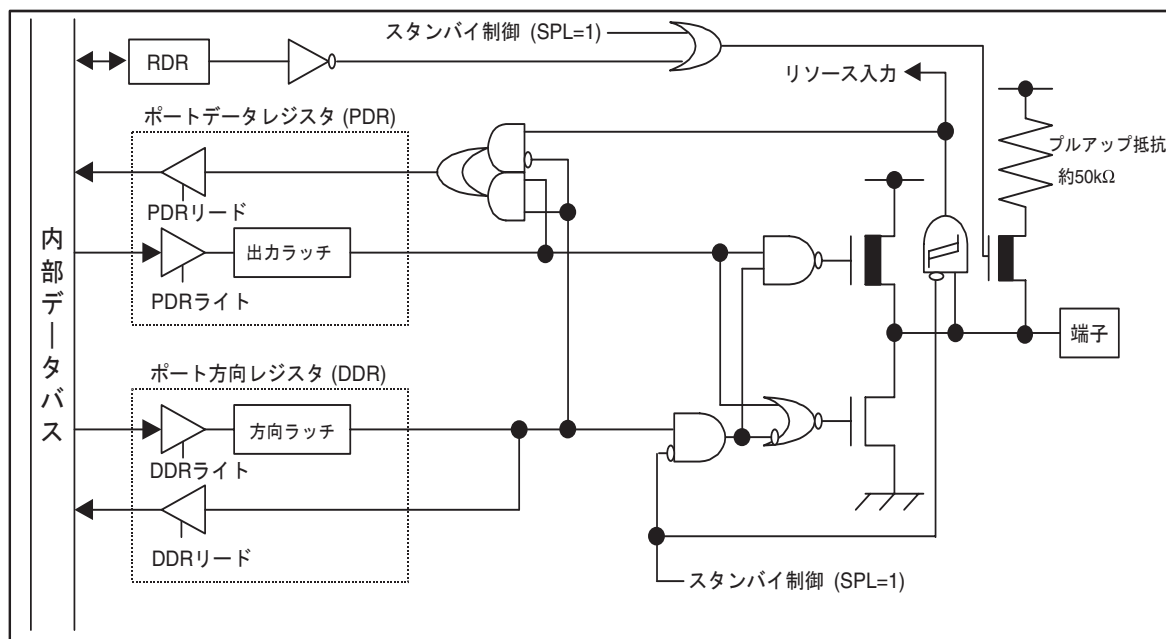
ポート	端子	ポート機能		リソース機能		入出力形式		入出力回路形式
						入力	出力	
ポート 2	P20/TIN1	P20	汎用入出力	TIN1	16 ビットリロード タイマ 1 入力	CMOS (ヒステリシス)	CMOS	D
	P21/TO1	P21		TO1	16 ビットリロード タイマ 1 出力			
	P22	P22		-	-			
	P23	P23		-	-			
	P24	P24		-	-			
	P25	P25		-	-			
	P26	P26		-	-			
	P27	P27		-	-			

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート 2 の端子のブロックダイアグラム

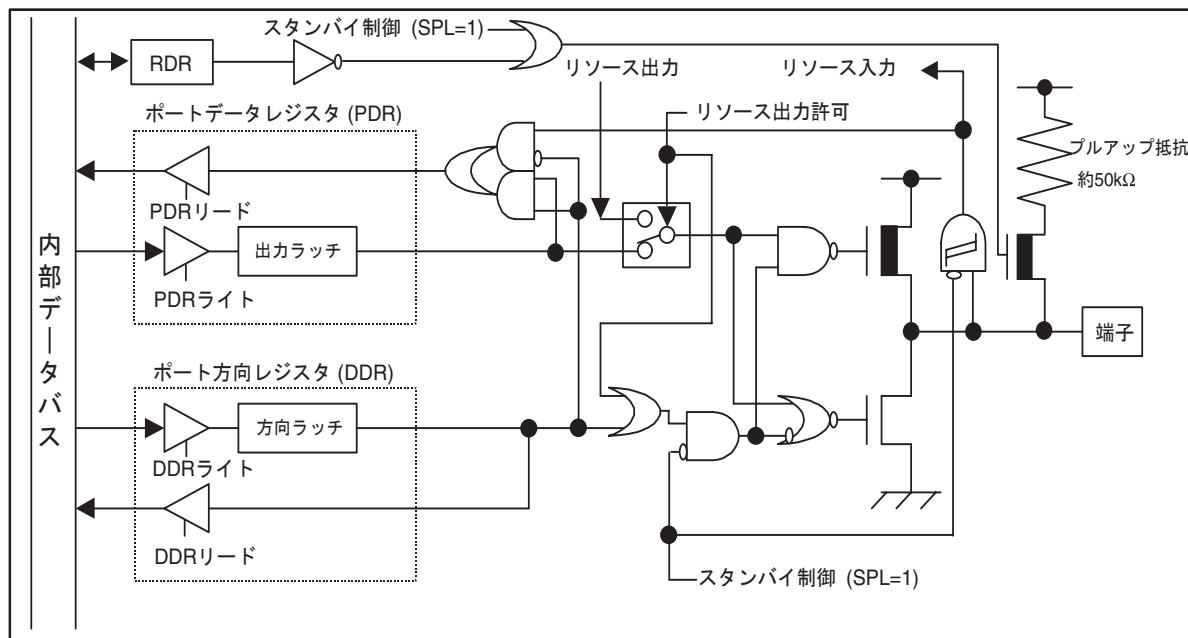
ポート 2 端子 (P21/TO1 を除く) のブロックダイアグラムを図 9.5-1 に示します。

図 9.5-1 ポート 2 の端子 (P21/TO1 を除く) のブロックダイアグラム



P21/TO1 端子のブロックダイアグラムを図 9.5-2 に示します。

図 9.5-2 P21/TO1 端子のブロックダイアグラム



リソース出力許可ビットを設定すると、ポートは、DDR2 レジスタの値に関係なくリソースの出力端子として機能します。

■ ポート2のレジスタ

ポート2のレジスタには、PDR2, DDR2 と RDR2 があります。各レジスタを構成しているビットは、ポート2の端子に1対1で対応しています。

表 9.5-2 にポート2の端子との対応するレジスタビットを示します。

表 9.5-2 ポート2のレジスタと端子の対応

ポート名	関連するレジスタのビットと対応する端子								
ポート 2	PDR2, DDR2, RDR2	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	対応する端子	P27	P26	P25	P24	P23	P22	P21	P20

9.5.1 ポート 2 のレジスタ (PDR2, DDR2, RDR2)

ポート 2 のレジスタについて説明します。

■ ポート 2 のレジスタの機能

● ポート 2 データレジスタ (PDR2)

ポート 2 データレジスタは、ポート 2 の各端子の状態を示します。

● ポート 2 データ方向レジスタ (DDR2)

DDR2 レジスタは、ポート 2 の各端子（ビット）のデータ入出力方向を指定します。DDR2 レジスタのいずれかのビットが "1" の場合は、対応するポート（端子）は出力ポートとして設定されます。ビットが "0" の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

- リソースの出力と兼用している端子は、その端子に対応するリソース出力許可ビットが設定されている限り、ポートは DDR2 レジスタの値とは無関係にリソースの出力端子として機能します。
- 入力端子と兼用しているリソースを使用するには、各リソース入力端子に対応するポート方向レジスタのビットに "0" を設定して、ポートを入力モードにしてください。

● ポート 2 プルアップ抵抗設定レジスタ (RDR2)

RDR2 レジスタは、ポート 2 の各端子（ビット）のプルアップ抵抗を選択します。RDR2 レジスタのいずれかのビットが "1" の場合、プルアップ抵抗はそのビットに対応するポート（端子）に対して選択され、いずれかのビットが "0" の場合は、ビットに対応するプルアップ抵抗は選択されません。

表 9.5-3 にポート 2 のレジスタの機能を示します。

表 9.5-3 ポート 2 のレジスタの機能

レジスタ名	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート 2 データレジスタ (PDR2)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "L" レベルに 設定されます	R/W	000002 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチへ "1" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "H" レベルに 設定されます			
ポート 2 データ方向レジ スタ (DDR2)	0	方向ラッチは "0"	出力バッファが "OFF" され、 ポートが入力モードになります	R/W	000012 _H	00000000 _B
	1	方向ラッチは "1"	出力バッファが "ON" され、 ポートが出力モードになります			

表 9.5-3 ポート 2 のレジスタの機能

レジスタ名	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート 2 プルアップ 抵抗設定 レジスタ (RDR2)	0	設定ラッチは "0"	プルアップ抵抗が遮断され、入 力モードでポートは Hi-Z 状態 になります	R/W	00008E _H	00000000 _B
	1	設定ラッチは "1"	プルアップ抵抗が選択され、入 力モードでポートは H レベルに 保持されます			

R/W: リード / ライト可能

X: 不定

9.5.2 ポート 2 の動作

ポート 2 の動作を説明します。

■ ポート 2 の動作

● 出力モード時のポート動作

- DDR2 レジスタのいずれかのビットに "1" を設定すると、対応するポートの端子は出力モードになります。
- 出力モード時に PDR2 レジスタに書き込まれたデータは、PDR2 の出力ラッチに保持され、端子へ出力されます。
- PDR2 レジスタを読み出すと、ポート端子における値 (PDR2 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力モード時のポート動作

- DDR2 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- ただし、RDR2 レジスタのいずれかのビットに "1" を設定してプルアップ抵抗を選択すると、端子は "H" レベルで保持されます。
- 入力モードで PDR1 レジスタに書き込まれたデータは、PDR2 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR2 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● リソース出力としてのポート動作

リソース出力許可ビットを許可にすると、ポートはリソース出力として使用できます。入力と出力の切換えを指定する場合、リソース許可ビットの状態が優先されます。すなわち、たとえ DDR2 レジスタのいずれかのビットが "0" であっても、リソース出力が許可になっている場合は、対応するポートの端子はリソース出力として使用されます。リソース出力が許可であっても端子の値を読み出すことはできるので、リソースの出力値を読み出すことができます。

● リソース入力としてのポート動作

ポートをリソース入力として使用する場合、端子の値は常にリソースの入力値となります。リソースの外部信号として使用するには、DDR2 レジスタに "0" を設定し、ポートを入力モードにしてください。

● リセット後のポート動作

- MCU がリセットされた場合、DDR2 レジスタは "0" に初期化されます。その結果、出力バッファが "OFF" になり（入出力モードは、入力に変化）、端子はハイインピーダンス状態になります。
- PDR2 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モード時に使用する場合は、出力データを PDR2 レジスタに設定した後、出力モードは DDR2 レジスタにおいて設定しなければなりません。

● ストップモードまたはタイムベースタイマモードの動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、端子はハイインピーダンス状態になります。これは、出力バッファが DDR2 レジスタの値とは無関係に "OFF" されるからです。なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。

表 9.5-4 にポート 2 の端子状態を示します。

表 9.5-4 ポート 2 の端子状態

端子	通常動作	スリープモード	ストップモード、 タイムベースタイマモード (SPL = 0)	ストップモード、 タイムベースタイマモード (SPL = 1, RDR = 0)	ストップモード、タイムベースタイマモード (SPL = 1, RDR = 1)
P20/TIN1 ~ P27	汎用 I/O ポート	汎用 I/O ポート	汎用 I/O ポート	入力禁止 / 出力は Hi-Z 状態	入力禁止 / H レベル保持

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)

Hi-Z: ハイインピーダンス

9.6 ポート 3

ポート 3 は汎用入出力ポートですが、リソース出力と兼用しています。各ポートの端子は、入出力ポートとリソース出力の間で切換えることができます。本節では、汎用入出力ポート機能を中心に、ポート 3 の構成と端子一覧、端子のブロックダイアグラムを示し、対応するレジスタについて説明します。

■ ポート 3 の構成

ポート 3 は、以下から構成されています。

- 汎用入出力端子 / リソース出力端子 (P30 ~ P37/PPG0)
- ポート 3 データレジスタ (PDR3)
- ポート 3 データ方向レジスタ (DDR3)
- ポート 3 プルアップ抵抗設定レジスタ (RDR3)

■ ポート 3 の端子配列

ポート 3 は、リソースの出力端子としても使用します。したがって、これらの端子は、リソースの入出力端子として使用した場合、汎用入出力ポートとしては使用できません。

表 9.6-1 にポート 3 の端子配列を示します。

表 9.6-1 ポート 3 の端子

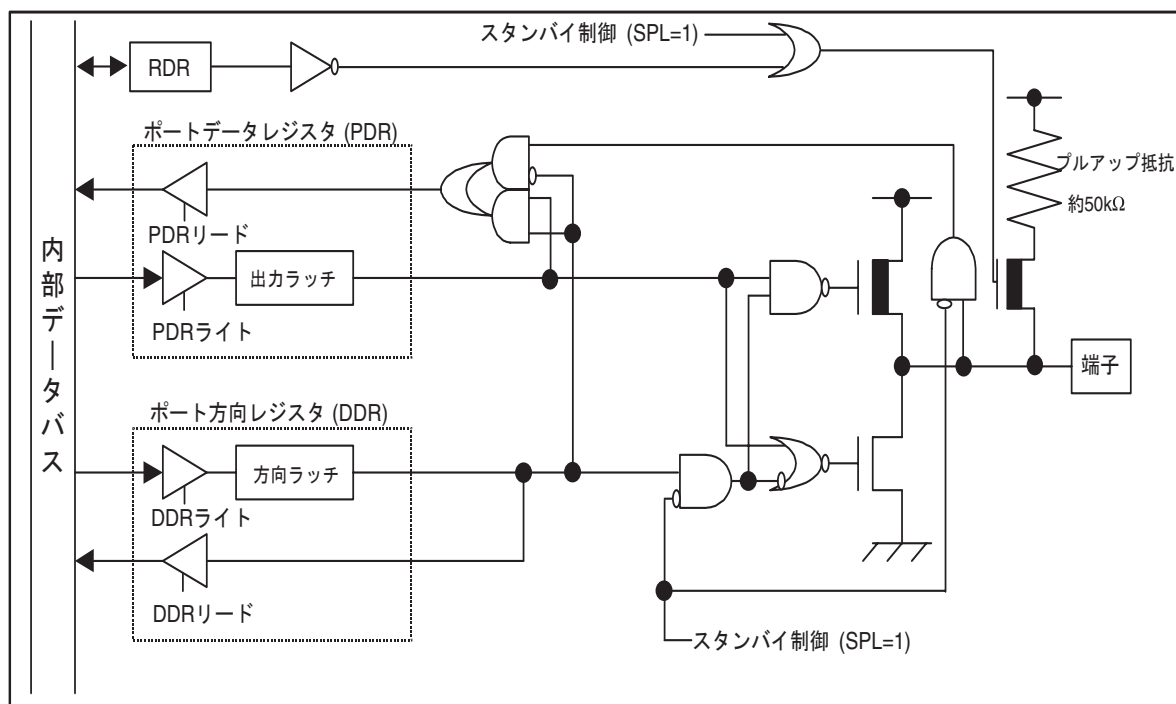
ポート	端子	ポート機能		リソース機能		入出力形式		入出力回路形式
						入力	出力	
ポート 3	P30	P30	汎用入出力	-	-	CMOS	CMOS	E
	P31	P31		-	-			
	P32	P32		-	-			
	P33	P33		-	-			
	P34	P34		-	-			
	P35	P35		-	-			
	P36	P36		-	-			
	P37/PPG0	P37		PPG0	PPG0 出力			

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート 3 の端子のブロックダイアグラム

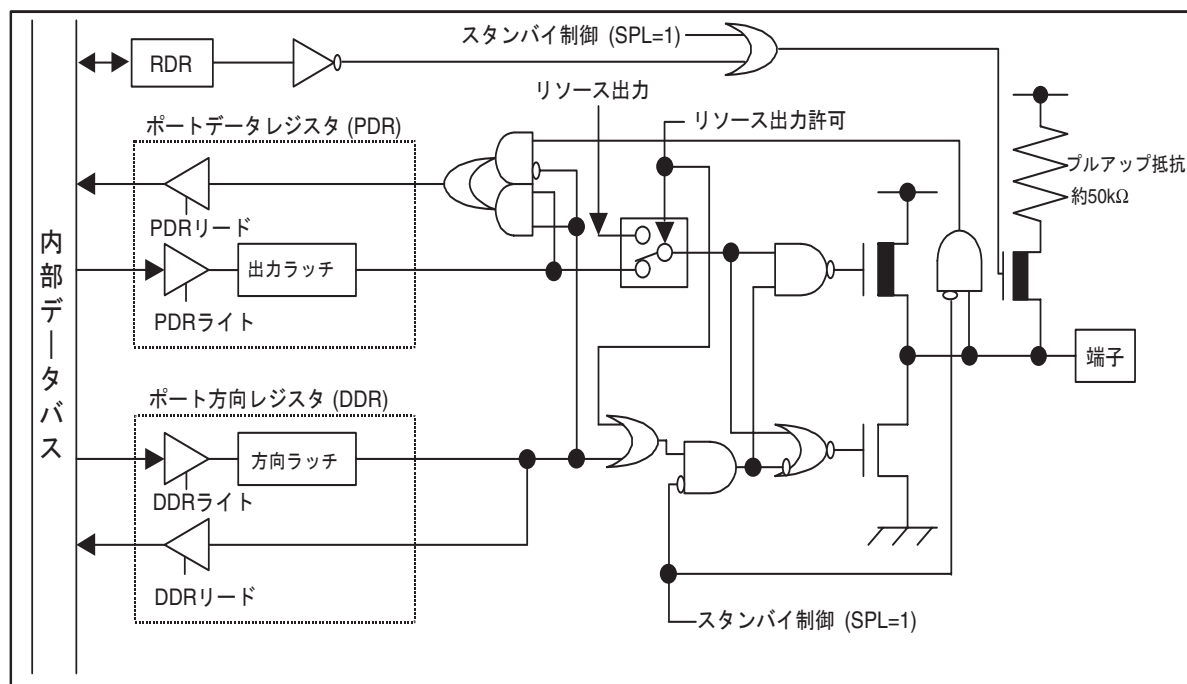
ポート 3 の端子のブロックダイアグラムを図 9.6-1 に示します。

図 9.6-1 ポート 3 の端子 (P37/PPG0 を除く) のブロックダイアグラム



P37/PPG0 端子のブロックダイアグラムを図 9.6-2 に示します。

図 9.6-2 ポート 3 の端子のブロックダイアグラム



リソース出力許可ビットを設定すると、ポートは、DDR3 レジスタの値に関係なくリソースの出力端子として機能します。

■ ポート 3 のレジスタ

ポート 3 のレジスタには、PDR3, DDR3 および RDR3 があります。各レジスタを構成しているビットは、ポート 3 の端子に 1 対 1 で対応しています。

表 9.6-2 にポート 3 の端子と対応するレジスタビットを示します。

表 9.6-2 ポート 3 のレジスタと端子の対応

ポート名	関連するレジスタのビットと対応する端子								
ポート 3	PDR3, DDR3, RDR3	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	対応する端子	P37	P36	P35	P34	P33	P32	P31	P30

9.6.1 ポート3のレジスタ (PDR3, DDR3, RDR3)

ポート3のレジスタについて説明します。

■ ポート3のレジスタの機能

● ポート3データレジスタ (PDR3)

PDR3 レジスタは、ポート3の各端子の状態を示します。

● ポート3データ方向レジスタ (DDR3)

DDR3 レジスタは、ポート3の各端子（ビット）のデータ入出力方向を指定します。DDR3 レジスタのいずれかのビットが"1"の場合は、対応するポート（端子）は出力ポートとして設定されます。ビットが"0"の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

リソースの出力と兼用している端子は、その端子に対応するリソース出力許可ビットが許可に設定されている限り、ポートはDDR3レジスタの値とは無関係にリソースの出力端子として機能します。

● ポート3プルアップ抵抗設定レジスタ (RDR3)

RDR3 レジスタは、ポート3の各端子（ビット）のプルアップ抵抗を選択します。RDR3 レジスタのいずれかのビットが"1"の場合、プルアップ抵抗はそのビットに対応するポート（端子）に対して選択され、いずれかのビットが"0"の場合は、ビットに対応するプルアップ抵抗は選択されません。

表 9.6-3 にポート3のレジスタ機能を示します。

表 9.6-3 ポート3のレジスタ機能

レジスタ名	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート3 データレジスタ (PDR3)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "L" レベルに 設定されます	R/W	000002 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチへ "1" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "H" レベルに 設定されます			
ポート3 データ方向レジ スタ (DDR3)	0	方向ラッチは "0"	出力バッファが "OFF" され、 ポートが入力モードになります	R/W	000013 _H	00000000 _B
	1	方向ラッチは "1"	出力バッファが "ON" され、 ポートが出力モードになります			

表 9.6-3 ポート 3 のレジスタ機能

レジスタ名	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート 3 プルアップ 抵抗設定 レジスタ (RDR3)	0	設定ラッチは "0"	プルアップ抵抗が遮断され、入 力モードでポートは Hi-Z 状態 になります	R/W	00008F _H	00000000 _B
	1	設定ラッチは "1"	プルアップ抵抗が選択され、入 力モードでポートは H レベルに 保持されます			

R/W: リード / ライト可能

X: 不定

9.6.2 ポート 3 の動作説明

ポート 3 の動作について説明します。

■ ポート 3 の動作

● 出力モード時のポート動作

- DDR3 レジスタのいずれかのビットに "1" を設定すると、対応するポートの端子は出力モードになります。
- 出力モード時において PDR3 レジスタに書き込まれたデータは、PDR3 の出力ラッチにて保持され、端子へ出力されます。
- PDR3 レジスタを読み出すと、ポート端子における値 (PDR3 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力モード時のポート動作

- DDR3 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- ただし、RDR3 レジスタのいずれかのビットに "1" を設定してプルアップ抵抗を選択すると、端子は "H" レベルで保持されます。
- 入力モードで PDR3 レジスタに書き込まれたデータは、PDR3 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR3 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● リソース出力としてのポート動作

リソース出力許可ビットを許可にすると、ポートはリソース出力として使用できます。入力と出力の切換えを指定する場合、リソース許可ビットの状態が優先されます。すなわち、たとえ DDR3 レジスタのいずれかのビットが "0" であっても、リソース出力が許可になっている場合は、対応するポートの端子はリソース出力として使用されます。リソース出力が許可であっても端子の値を読み出すことはできるので、リソースの出力値を読み出すことができます。

● リセット後のポート動作

- MCU がリセットされた場合、DDR3 レジスタは "0" に初期化されます。その結果、出力バッファが "OFF" になり（入出力モードは、入力に変化）、端子はハイインピーダンス状態になります。
- PDR3 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モード時に使用する場合は、出力データを PDR3 レジスタに設定した後、出力モードは DDR3 レジスタにおいて設定しなければなりません。

● ストップモードまたはタイムベースタイマモードの動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、端子はハイインピーダンス状態になります。これは、出力バッファが DDR3 レジスタの値とは無関係に "OFF" されるからです。なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。ポート 3 の端子状態を表 9.6-4 に示します。

表 9.6-4 ポート 3 の端子状態

端子	通常動作	スリープモード	ストップモード、 タイムベースタイ マモード (SPL = 0)	ストップモード、 タイムベースタイ マモード (SPL = 1, RDR = 0)	ストップモード、タイ ムベースタイマモード (SPL = 1, RDR = 1)
P30 ~ P37/ PPG0	汎用 I/O ポート	汎用 I/O ポート	汎用 I/O ポート	入力禁止 / 出力は Hi-Z 状態	入力禁止 / "H" レベル保持

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)
Hi-Z: ハイインピーダンス

9.7 ポート 4

ポート 4 は汎用入出力ポートですが、リソース入出力と兼用しています。各ポートの端子は、入出力ポートとリソース入出力の間で切換えることができます。本節は、汎用入出力ポート機能を中心に、ポート 4 の構成と端子一覧、端子のブロックダイアグラム、対応するレジスタについて説明します。

■ ポート 4 の構成

ポート 4 は、以下から構成されています。

- ・ 汎用入出力端子 / リソース / 入出力端子 (P40 / PPG1 ~ P47/PWO1)
- ・ ポート 4 データレジスタ (PDR4)
- ・ ポート 4 データ方向レジスタ (DDR4)

■ ポート 4 の端子

ポート 4 は、リソースの入出力端子としても使用します。したがって、これらの端子は、リソースの入出力端子として使用した場合は、汎用入出力ポートとしては使用できません。

表 9.7-1 にポート 4 の端子一覧を示します。

表 9.7-1 ポート 4 の端子

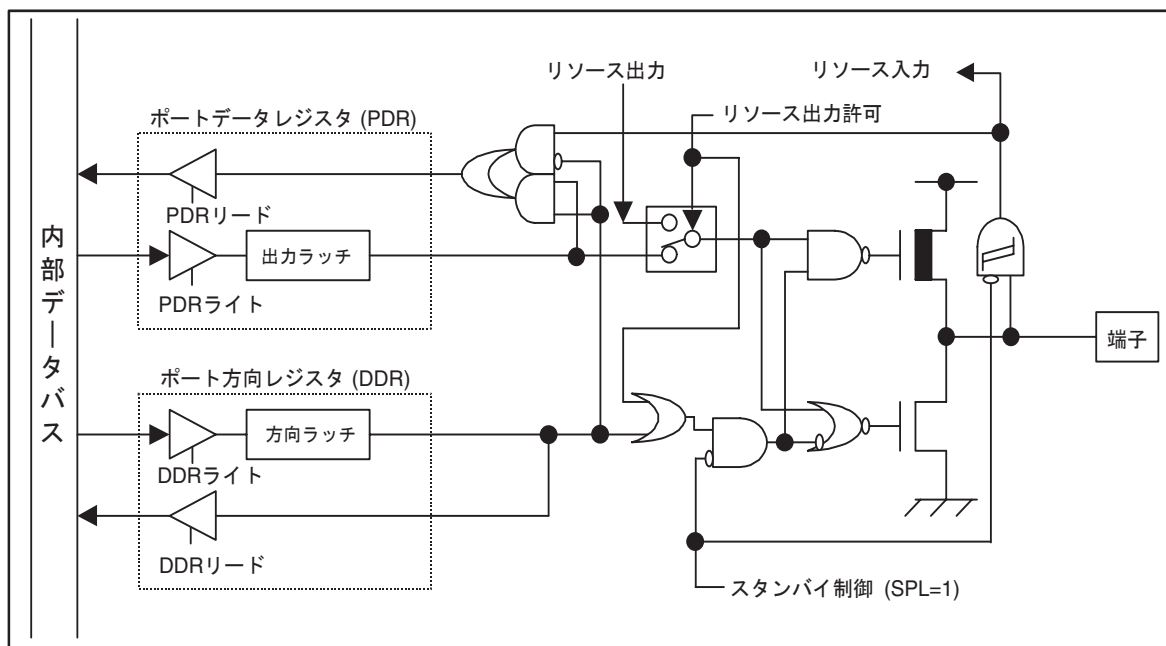
ポート	端子	ポート機能	リソース機能	入出力形式		入出力回路形式
				入力	出力	
ポート 4	P40/PPG1	P40	PPG1	PPG1 出力	CMOS (ヒステリシス)	F
	P41/TIN0	P41	TIN0	16 ビットリロードタイマ 0 入力		
	P42/TO0	P42	TO0	16 ビットリロードタイマ 0 出力		
	P43/SCK0	P43	SCK0	UART0 シリアルクロック I/O		
	P44/SOT0	P44	SOT0	UART0 データ出力		
	P45/SIN0	P45	SIN0	UART0 データ入力		
	P46/PWI1	P46	PWI1	PWC1 入力		G
	P47/PWO1	P47	PWO1	PWC1 出力		F

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート 4 の端子のブロックダイアグラム

図 9.7-1 にポート 4 の端子 (P41/TIN0, P45/SIN0, P46/PWI1 を除く) のブロックダイアグラムを示します。

図 9.7-1 ポート 4 の端子 (P41/TIN0, P45/SIN0, P46/PWI1 を除く) のブロックダイアグラム



リソース出力許可ビットを設定すると、ポートは、DDR4 レジスタの値に関係なくリソースの出力端子として機能します。

図 9.7-2 も P41/TIN0 および P46/PWI1 端子のブロックダイアグラムを示します。

図 9.7-2 P41/TIN0 および P46/PWI1 端子のブロックダイヤグラム

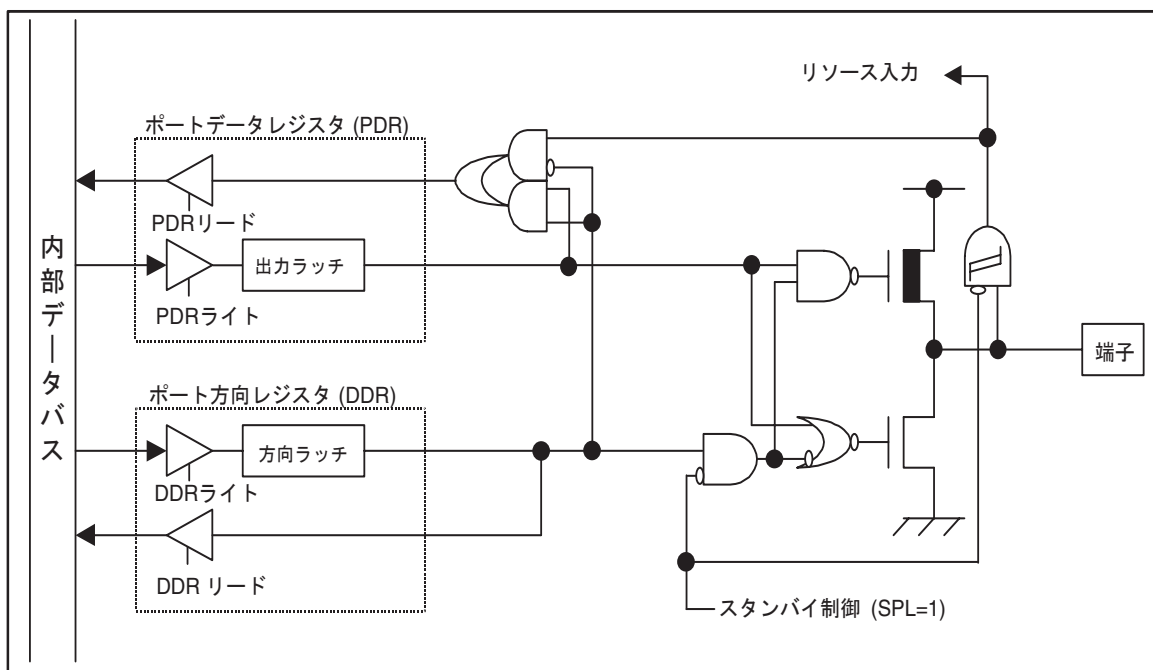
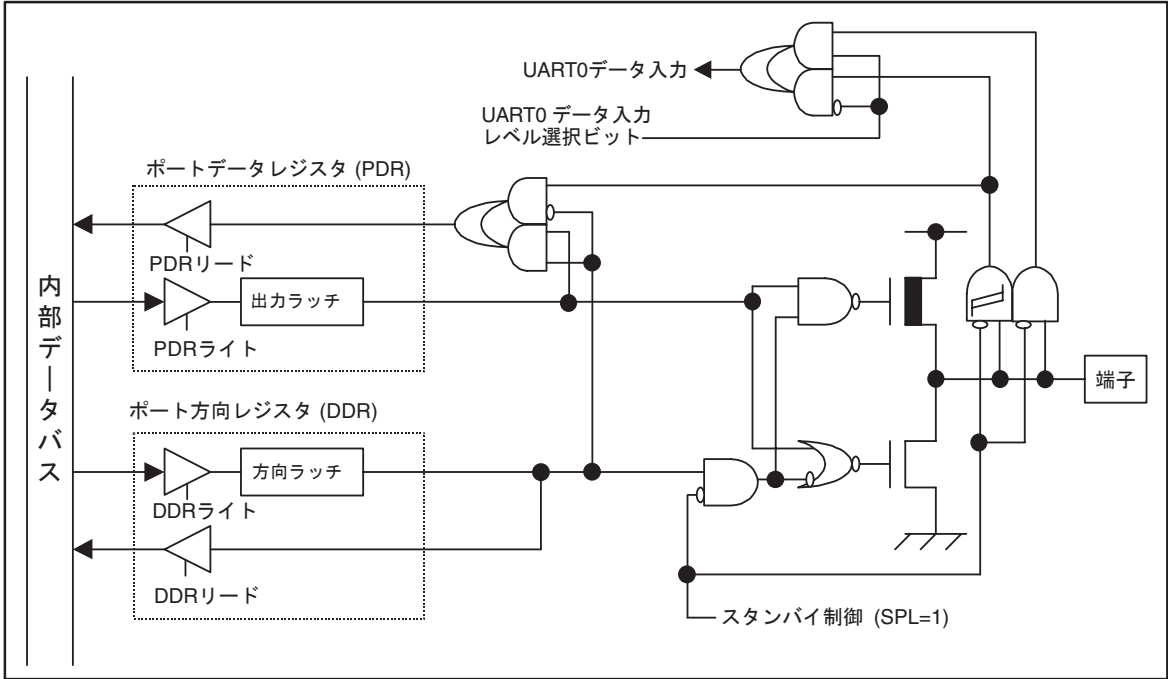


図 9.7-3 に P45/SIN0 端子のブロックダイヤグラムを示します。

図 9.7-3 P45/SIN0 端子のブロックダイヤグラム



■ ポート 4 のレジスタ

ポート 4 のレジスタには、PDR4 と DDR4 があります。各レジスタを構成しているビットは、ポート 4 の端子に 1 対 1 で対応しています。

表 9.7-2 にポート 4 のレジスタと端子の対応を示します。

表 9.7-2 ポート 4 端子と対応するレジスタビット

ポート名	関連するレジスタのビットと対応する端子								
	PDR4, DDR4	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ポート 4	対応する端子	P47	P46	P45	P44	P43	P42	P41	P40

9.7.1 ポート 4 のレジスタ (PDR4, DDR4)

ポート 4 のレジスタについて説明します。

■ ポート 4 のレジスタの機能

● ポート 4 データレジスタ (PDR4)

PDR4 レジスタは、ポート 4 の各端子の状態を示します。

● ポート 4 データ方向レジスタ (DDR4)

DDR4 レジスタは、ポート 4 の各端子（ビット）のデータ入出力方向を指定します。DDR4 レジスタのいずれかのビットが "1" の場合は、対応するポート（端子）は出力ポートとして設定されます。いずれかのビットが "0" の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

- リソースの出力と兼用している端子は、その端子に対応するリソース出力許可ビットが設定されている限り、ポートは DDR4 レジスタの値とは無関係にリソースの出力端子として機能します。
- 入力端子と兼用しているリソースを使用するには、各リソース入力端子に対応するポート方向レジスタのビットに "0" を設定して、ポートを入力モードにしてください。

表 9.7-3 にポート 4 のレジスタ機能を示します。

表 9.7-3 ポート 4 のレジスタ機能

レジスタ名	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート 4 データレジスタ (PDR4)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "L" レベルに設定 されます	R/W	000004 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチへ "1" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "H" レベルに設定 されます			
ポート 4 データ方向レジ スタ (DDR4)	0	方向ラッチ は "0"	出力バッファが "OFF" され、ポー トが入力モードになります	R/W	000014 _H	00000000 _B
	1	方向ラッチ は "1"	出力バッファが "ON" され、ポー トが出力モードになります			

R/W: リード / ライト可能

X: 不定

9.7.2 ポート 4 の動作説明

ポート 4 の動作を説明します。

■ ポート 4 の動作

● 出力モード時のポート動作

- DDR4 レジスタのいずれかのビットに "1" を設定すると、対応するポートの端子は出力モードになります。
- 出力モード時において PDR4 レジスタに書き込まれたデータは、PDR4 の出力ラッチにて保持され、端子へ出力されます。
- PDR4 レジスタを読み出すと、ポート端子における値 (PDR4 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力モード時のポート動作

- DDR4 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- 入力モードで PDR4 レジスタに書き込まれたデータは、PDR4 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR4 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● リソース出力としてのポート動作

リソース出力許可ビットを許可にすると、ポートはリソース出力として使用できます。入力と出力の切換えを指定する場合、リソース許可ビットの状態が優先されます。すなわち、たとえ DDR4 レジスタのいずれかのビットが "0" であっても、リソース出力が許可になっている場合は、対応するポートの端子はリソース出力として使用されます。リソース出力が許可であっても端子の値を読み出すことはできるので、リソースの出力値を読み出すことができます。

● リソース入力としてのポート動作

ポートをリソース入力として使用する場合、端子の値は常にリソースの入力値となります。リソースの外部信号として使用するには、DDR4 レジスタに "0" を設定し、ポートを入力モードにしてください。

● リセット後のポート動作

- MCU がリセットされた場合、DDR4 レジスタは "0" に初期化されます。その結果、出力バッファが "OFF" になり（入出力モードは、入力に変化）、端子はハイインピーダンス状態になります。
- PDR4 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モード時に使用する場合は、出力データを PDR4 レジスタに設定した後、出力モードは DDR4 レジスタにおいて設定しなければなりません。

● ストップモードまたはタイムベースタイマモード時のポート動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、端子はハイインピーダンス状態になります。これは、出力バッファが DDR4 レジスタの値とは無関係に "OFF" されるからです。なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。

表 9.7-4 にポート 4 の端子状態を示します。

表 9.7-4 ポート 4 の端子状態

端子	通常動作	スリープモード	ストップモード、 タイムベースタイマモード (SPL = 0)	ストップモード、 タイムベースタイマモード (SPL = 1)
P40/PPG1 ~ P47/PWO1	汎用入出力 ポート	汎用入出力 ポート	汎用入出力ポート	入力禁止 / 出力は Hi-Z 状態

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)

Hi-Z: ハイインピーダンス

9.8 ポート 5

ポート 5 は汎用入出力ポートですが、リソース入出力と兼用しています。各ポートの端子は、入出力ポートとリソース入出力の間で切換えることができます。本節は、汎用入出力ポート機能を中心に、ポート 5 の構成と端子一覧、端子のブロックダイアグラム、対応するレジスタについて説明します。

■ ポート 5 の構成

ポート 5 は、以下から構成されています。

- ・ 汎用入出力端子 / リソース入出力端子 (P50 /PPG2 と P51/INT7)
- ・ ポート 5 データレジスタ (PDR5)
- ・ ポート 5 データ方向レジスタ (DDR5)

■ ポート 5 の端子

ポート 5 は、リソースの入出力端子としても使用します。したがって、これらの端子は、リソースの入出力端子として使用した場合は、汎用入出力ポートとしては使用できません。

表 9.8-1 にポート 5 の端子一覧を示します。

表 9.8-1 ポート 5 の端子

ポート	端子	ポート機能		リソース機能		入出力形式		入出力回路形式
						入力	出力	
ポート 5	P50/PPG2	P50	汎用入出力	PPG2	PPG2 出力	CMOS (ヒステリシス)	CMOS	F
	P51/INT7	P51		INT7	外部割込み 入力			

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート 5 の端子のブロックダイヤグラム

図 9.8-1 に P50/PPG2 端子のブロックダイヤグラムを示します。

図 9.8-1 P50/PPG2 端子のブロックダイヤグラム

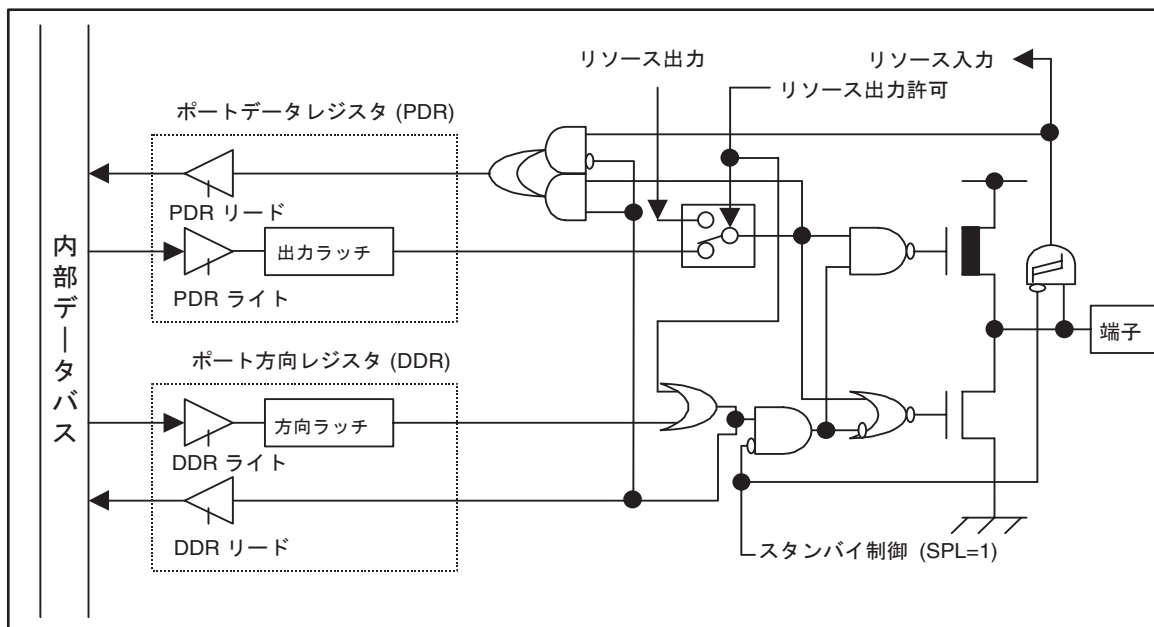
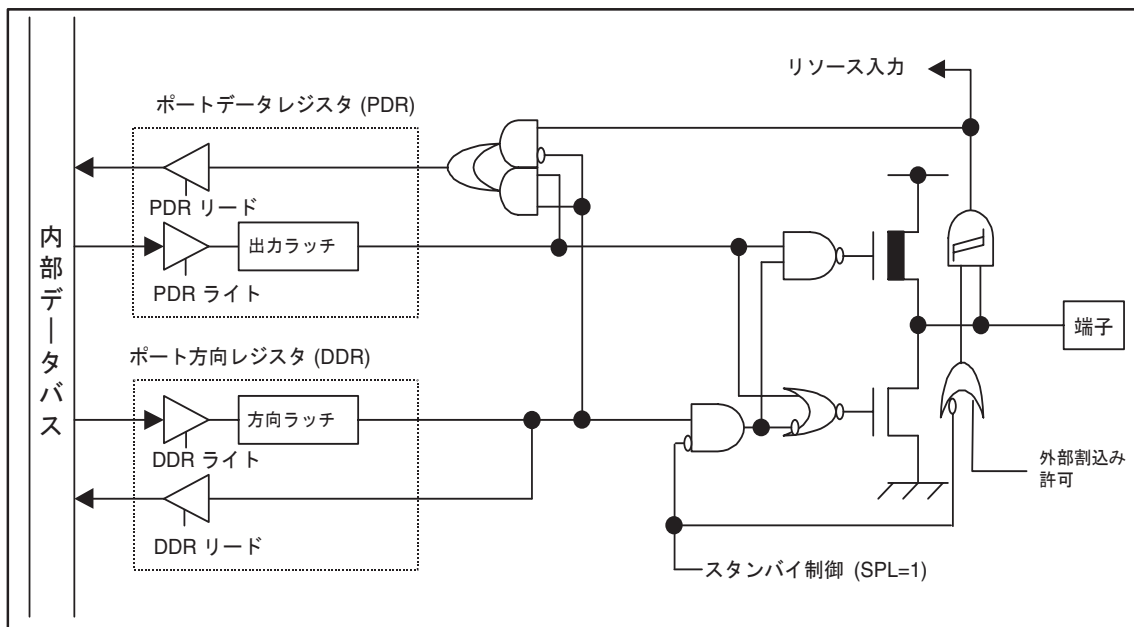


図 9.8-2 に P51/INT7 端子のブロックダイヤグラムを示します。

図 9.8-2 P51/INT7 端子のブロックダイヤグラム



リソース出力許可ビットを設定すると、ポートは、DDR5 レジスタの値に関係なくリソースの出力端子として機能します。

■ ポート5のレジスタ

ポート5のレジスタには、PDR5とDDR5があります。各レジスタを構成しているビットは、ポート5の端子に1対1で対応しています。

表9.8-2にポート5のレジスタと端子の対応を示します。

表 9.8-2 ポート5のレジスタと端子の対応

ポート	レジスタビットと対応するポートの端子								
ポート5	PDR5, DDR5	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	対応する端子	-	-	-	-	-	-	P51	P50

9.8.1 ポート 5 のレジスタ (PDR5, DDR5)

ポート 5 のレジスタについて説明します。

■ ポート 5 のレジスタの機能

● ポート 5 データレジスタ (PDR5)

ポート 5 データレジスタは、端子の状態を示します。

● ポート 5 データ方向レジスタ (DDR5)

DDR5 レジスタは、ポート 5 の各端子（ビット）のデータ入出力方向を指定します。DDR5 レジスタのいずれかのビットが "1" の場合は、対応するポート（端子）は出力ポートとして設定されます。いずれかのビットが "0" の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

- リソースの出力と兼用している端子は、その端子に対応するリソース出力許可ビットが設定されている限り、ポートは DDR5 レジスタの値とは無関係にリソースの出力端子として機能します。
- 入力端子と兼用しているリソースを使用するには、各リソース入力端子に対応するポート方向レジスタのビットに "0" を設定して、ポートを入力モードにしてください。

表 9.8-3 にポート 5 のレジスタの機能を示します。

表 9.8-3 ポート 5 のレジスタの機能

レジスタ名	データ	読み出し時	書き込み時	リード ライト	アドレス	初期値
ポート 5 データレジスタ (PDR5)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "L" レベルに 設定されます	R/W	000005 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチへ "1" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "H" レベルに 設定されます			
ポート 5 データ方向レジ スタ (DDR5)	0	方向ラッチは "0"	出力バッファが "OFF" され、 ポートが入力モードになります	R/W	000015 _H	XXXXXX00 _B
	1	方向ラッチは "1"	出力バッファが "ON" され、 ポートが出力モードになります			

R/W: リード / ライト可能

X: 不定

9.8.2 ポート 5 の動作説明

ポート 5 の動作について説明します。

■ ポート 5 の動作

● 出力モード時のポート動作

- DDR5 レジスタのいずれかのビットに "1" を設定すると、対応するポートの端子は出力モードになります。
- 出力モード時において PDR5 レジスタに書き込まれたデータは、PDR5 の出力ラッチにて保持され、端子へ出力されます。
- PDR5 レジスタを読み出すと、ポート端子における値 (PDR5 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力モード時のポート動作

- DDR5 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- 入力モードで PDR5 レジスタに書き込まれたデータは、PDR5 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR5 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● リソース出力としてポート動作

リソース出力許可ビットを許可にすると、ポートはリソース出力として使用できます。入力と出力の切換えを指定する場合、リソース許可ビットの状態が優先されます。すなわち、たとえ DDR5 レジスタのいずれかのビットが "0" であっても、リソース出力が許可になっている場合は、対応するポートの端子はリソース出力として使用されます。リソース出力が許可であっても端子の値を読み出すことはできるので、リソースの出力値を読み出すことができます。

● リソース入力としてのポート動作

ポートをリソース入力として使用する場合、端子の値は常にリソースの入力値となります。リソースの外部信号として使用するには、DDR5 レジスタに "0" を設定し、ポートを入力モードにしてください

● リセット後のポート動作

- MCU がリセットされた場合、DDR5 レジスタは "0" に初期化されます。その結果、出力バッファが "OFF" になり（入出力モードは入力に変化）、端子はハイインピーダンス状態になります。
- PDR5 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モード時に使用する場合は、出力データを PDR5 レジスタに設定した後、出力モードは DDR5 レジスタにおいて設定しなければなりません。

● ストップモードまたはタイムベースタイマモード時のポート動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、端子はハイインピーダンス状態になります。これは、出力バッファが DDR5 レジスタの値とは無関係に "OFF" されるからです。なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。

表 9.8-4 にポート 5 の端子状態を示します。

表 9.8-4 ポート 5 の端子状態

端子	通常動作	スリープモード	ストップモード, タイムベースタイマモード (SPL = 0)	ストップモード, タイムベースタイマモード (SPL = 1)
P50 / PPG2	汎用入出力ポート	汎用入出力ポート	汎用入出力ポート	入力禁止 / 出力は Hi-Z 状態
P51 / INT7	汎用入出力ポート	汎用入出力ポート	汎用入出力ポート	入力許可 */ 出力は Hi-Z 状態

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)

Hi-Z: ハイインピーダンス

*: P51/INT7 を外部割込み端子として構成した場合のみ有効になり、それ以外の場合は禁止されます。

9.9 ポート 6

ポート 6 は汎用入出力ポートですが、A/D コンバータアナログ入力と兼用しています。ポートの端子は、入出力ポートとアナログ入力機能の間で切換えることができます。本節では、汎用入出力ポート機能を中心に、ポート 6 の構成と端子一覧、端子のブロックダイアグラム、対応するレジスタについて説明します。

■ ポート 6 の構成

ポート 6 は、以下から構成されています。

- 汎用入出力ポート、アナログ入力端子 (P60/AN0 ~ P67/AN7)
- ポート 6 データレジスタ (PDR6)
- ポート 6 データ方向レジスタ (DDR6)
- アナログ入力許可レジスタ (ADER0)

■ ポート 6 の端子

ポート 6 は、アナログの入力端子としても使用します。したがって、これらの端子は、アナログ入力として使用した場合、汎用入出力ポートとしては使用できません。

表 9.9-1 にポート 6 の端子一覧を示します。

表 9.9-1 ポート 6 の端子

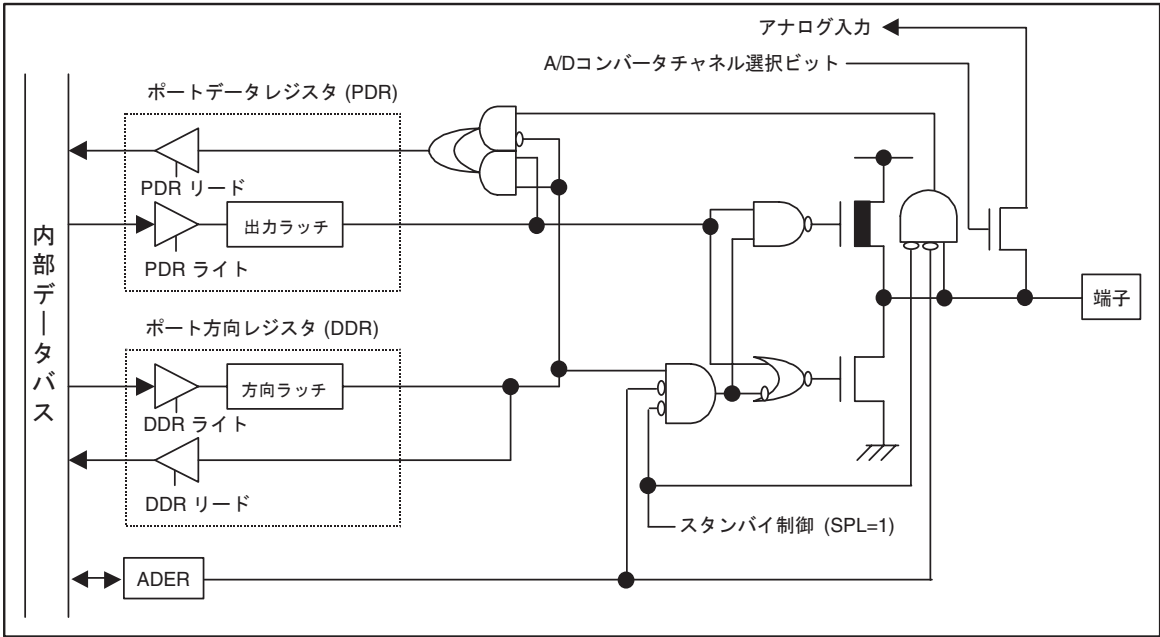
ポート	端子	ポート機能		リソース機能		入出力形式		入出力回路形式
						入力	出力	
ポート 6	P60/AN0	P60	汎用 入出力	AN0	アナログ入力 0	アナログ / CMOS	CMOS	H
	P61/AN1	P61		AN1	アナログ入力 1			
	P62/AN2	P62		AN2	アナログ入力 2			
	P63/AN3	P63		AN3	アナログ入力 3			
	P64/AN4	P64		AN4	アナログ入力 4			
	P65/AN5	P65		AN5	アナログ入力 5			
	P66/AN6	P66		AN6	アナログ入力 6			
	P67/AN7	P67		AN7	アナログ入力 7			

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート 6 の端子のブロックダイアグラム

図 9.9-1 にポート P60/AN0 ~ P67/AN7 のブロックダイアグラムを示します。

図 9.9-1 ポート 6 の端子のブロックダイアグラム



アナログ入力許可ビットを許可すると、DDR6 レジスタの値にかかわらず強制的に A/D コンバータ入力端子となります。

■ ポート 6 のレジスタ

ポート 6 のレジスタには、PDR6, DDR6, ADER0 があります。各レジスタを構成しているビットは、ポート 6 の端子に 1 対 1 に対応しています。

表 9.9-2 にポート 6 の端子と対応するレジスタビット示します。

表 9.9-2 ポート 6 の端子と対応するレジスタビット

ポート名	関連するレジスタのビットと対応する端子								
ポート 6	PDR6, DDR6, ADER0	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	対応する端子	P67	P66	P65	P64	P63	P62	P61	P60

9.9.1 ポート 6 のレジスタ (PDR6, DDR6, ADER0)

ポート 6 のレジスタについて説明します。

■ ポート 6 のレジスタの機能

- ポート 6 データレジスタ (PDR6)

PDR6 レジスタは、ポート 6 の各端子の状態を示します。

- ポート 6 データ方向レジスタ (DDR6)

DDR6 レジスタは、ポート 6 の各端子（ビット）のデータ入出力方向を指定します。DDR6 レジスタのいずれかのビットが "1" の場合は、対応するポート（端子）は出力ポートとして設定されます。いずれかのビットが "0" の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

- A/D 入力許可ビットが設定された場合、DDR6 レジスタの設定値に関係なく、対応するポートは A/D コンバータ入力端子となります。
- 汎用入出力ポートとして使用するには、対応する A/D コンバータ入力許可レジスタのビットに "0" を設定し、汎用入出力ポートとしてください。

- アナログ入力許可レジスタ 0(ADER0)

ADER0 レジスタの各ビットは、対応するポート 6 端子を汎用入出力ポートとして使用するかアナログ入力端子として使用するかを指定します。ADE ビットに "1" を設定すると、対応する端子をアナログ入力として使用します。ADE ビットに "0" を設定すると、対応する端子を汎用入出力として使用します。

< 注意事項 >

ポートを入力モードに設定して中間レベルの信号が入力されると、入力リーク電流が発生します。したがって、端子をアナログ入力として使用する場合は、対応する ADE ビットには必ず "1" を設定してください。

< 参照 >

MCU がリセットされた場合、DDR6 レジスタは "0" にクリアされ、ADER0 レジスタには "1" (アナログ入力として使用する) が設定されます。

表 9.9-3 にポート 6 のレジスタの機能を示します。

表 9.9-3 ポート 6 のレジスタ機能

レジスタ	データ	読み出し時	書き込み時	リード ライト	アドレス	初期値
ポート 6 データレジスタ (PDR6)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、端子が出力ポートとして機能する場合は、端子は "L" レベルに設定されます。	R/W	000006 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチへ "1" がロードされ、端子が出力ポートとして機能する場合は、端子は "H" レベルに設定されます。			
ポート 6 データ方向レジスタ (DDR6)	0	方向ラッチは "0"	出力バッファが "OFF" され、ポートが入力モードになります。	R/W	000016 _H	00000000 _B
	1	方向ラッチは "1"	出力バッファが "ON" され、ポートが出力モードになります。			
A/D 入力許可 レジスタ 0 (ADER0)	0	ポート入出力モード		R/W	0000C5 _H	11111111 _B
	1	アナログ入力モード				

R/W: リード / ライト可能

X: 不定

9.9.2 ポート 6 の動作説明

ポート 6 の動作について説明します。

■ ポート 6 の動作

● 出力モード時のポート動作

- ADER0 レジスタのいずれかのビットに "0" を設定すると、対応するポートの端子はポート入出力モードになります。
- DDR6 レジスタのいずれかのビットに "1" を設定すると、対応するポートの端子は出力モードになります。
- 出力モード時において PDR6 レジスタに書き込まれたデータは、PDR6 の出力ラッチにおいて保持され、ポート端子へ出力されます。
- PDR6 レジスタを読み出すと、ポート端子における値 (PDR6 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力モード時のポート動作

- ADER0 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子はポート入出力モードになります。
- DDR6 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- 入力モードで PDR6 レジスタに書き込まれたデータは、PDR6 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR6 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● アナログ入力としてのポート動作

ポート端子をアナログ入力として使用する場合は、対応する ADE ビットに "1" を書き込んでください。それによって、端子は汎用ポート端子としては動作せず、アナログ入力端子として動作します。この状態で PDR6 レジスタを読み出すと "0" が読み出されません。

● リセット後のポート動作

- MCU がリセットされた場合、DDR6 レジスタは "0" に ADER0 レジスタは "1" に初期化され、ポートはアナログ入力モードになります。汎用ポートとして使用する場合は、事前に ADER0 レジスタに "0" を書き込んでポート入出力モードにしてください。
- MCU がリセットされた場合、DDR6 レジスタは "0" に初期化されます。その結果、出力バッファが "OFF" になり（入出力モードは、入力に変化）、端子はハイインピーダンス状態になります。
- PDR6 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モード時に使用する場合は、出力データを PDR6 レジスタに設定した後、出力モードは DDR6 レジスタにおいて設定しなければなりません。

● ストップモード、タイムベースタイマモードの動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、端子はハイインピーダンス状態になります。これは、出力バッファが DDR6 レジスタの値とは無関係に "OFF" されるからです。なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。

表 9.9-4 にポート 6 の端子状態を示します。

表 9.9-4 ポート 6 の端子状態

端子	通常動作	スリープモード	ストップモード、 タイムベースタイマモード (SPL = 0)	ストップモード、 タイムベースタイマモード (SPL = 1)
P60/AN0 ~ P67/AN7	汎用入出力 ポート	汎用入出力 ポート	汎用入出力ポート	入力禁止 / 出力は Hi-Z 状態

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)

Hi-Z: ハイインピーダンス

9.10 ポート 7

ポート 7 は汎用入出力ポートですが、リソース入出力と兼用しています。各ポートの端子は、入出力ポートとリソース入出力の間で切換えることができます。本節は、汎用入出力ポート機能を中心に、ポート 7 の構成と端子一覧、端子のブロックダイアグラム、対応するレジスタについて説明します。

■ ポート 7 の構成

ポート 7 は、以下から構成されています。

- ・ 汎用入出力端子 / リソース入出力端子 (P70/DA0/AN8 ~ P77/IN1/AN15)
- ・ ポート 7 データレジスタ (PDR7)
- ・ ポート 7 データ方向レジスタ (DDR7)
- ・ アナログ入力許可レジスタ 1 (ADER1)

■ ポート 7 の端子

ポート 7 は、リソースの入出力端子としても使用します。したがって、これらの端子は、リソースの入出力端子として使用した場合は、汎用入出力ポートとしては使用できません。

表 9.10-1 にポート 7 の端子一覧を示します。

表 9.10-1 ポート 7 の端子

ポート	端子	ポート機能	リソース機能	入出力形式		入出力回路形式
				入力	出力	
ポート 7	P70/DA0/AN8	P70	DA0/AN8	D/A コンバータ出力 0 / A/D コンバータチャネル 8 入力	CMOS / (ヒステリシス) / アナログ	I
	P71/DA1/AN9	P71	DA1/AN9	D/A コンバータ出力 1 / A/D コンバータチャネル 9 入力		
	P72/SIN1/AN10	P72	SIN1/AN10	UART1 データ入力 / A/D コンバータチャネル 10 入力		J
	P73/SOT1/AN11	P73	SOT1/AN12	UART1 データ出力 / A/D コンバータチャネル 11 入力		K
	P74/SCK1/AN12	P74	SCK1/AN12	UART1 シリアルクロック入力 / A/D コンバータチャネル 12 入力		
	P75/FRCK/AN13	P75	FRCK/AN13	フリーランタイムクロック入力 / A/D コンバータチャネル 13 入力		
	P76/IN0/AN14	P76	IN0/AN14	入力キャプチャチャネル 0 入力 / A/D コンバータチャネル 14 入力		
	P77/IN1/AN15	P77	IN1/AN15	入力キャプチャチャネル 1 入力 / A/D コンバータチャネル 15 入力		

入出力回路形式については「1.7 入出力回路形式」を参照してください。

図 9.10-1 に P70/DA0/AN8 および P71/DA1/AN9 端子のブロックダイヤグラムを示します。

The diagram illustrates the internal data bus system for the A/D converter channel selection bit. It shows the following components and connections:

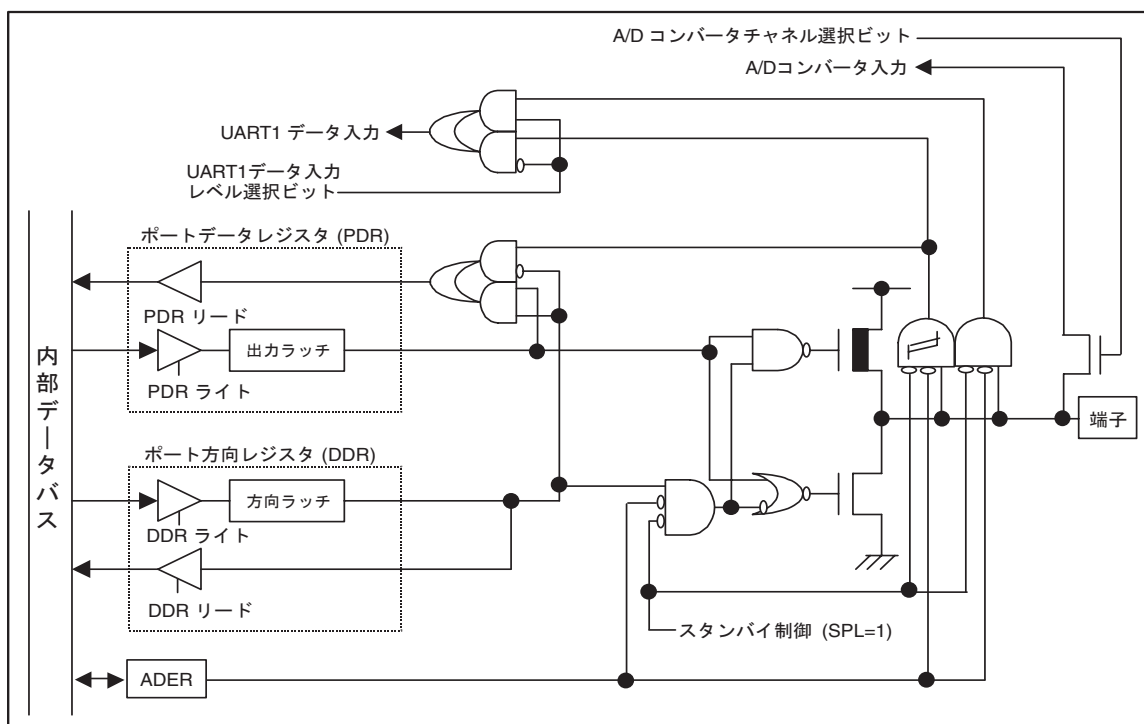
- Internal Data Bus (内部データバス):** The central horizontal bus connecting various components.
- Port Data Register (PDR):** Contains a **PDR リード** (PDR Read) and a **PDR ライト** (PDR Write) control signal.
- Port Direction Register (DDR):** Contains a **方向ラッチ** (Direction Latch) and a **DDR ライト** (DDR Write) control signal.
- ADER (A/D Converter Enable Register):** Receives the **A/Dコンバータ出力許可ビット** (A/D Converter Output Enable Bit) and outputs a **スタンバイ制御 (SPL=1)** (Standby Control) signal.
- A/D Converter Channel Selection Bit:** The input signal that selects the A/D converter channel.
- A/Dコンバータ入力 (A/D Converter Input):** The input signal to the A/D converter.
- D/Aコンバータ出力 (D/A Converter Output):** The output signal from the D/A converter.
- 端子 (Terminal):** The output terminal of the system.

The circuit logic involves several logic gates (AND, OR, NOT) and latches to manage the data flow and control signals between the internal bus and the external components.

D/A コンバータ出力許可ビットもしくはアナログ入力許可ビットを許可すると、DDR7 レジスタの値にかかわらず、ポートは強制的に D/A コンバータ出力端子または A/D コンバータ入力端子となります。

図 9.10-2 に P72/SIN1/AN10 のブロックダイアグラムを示します。

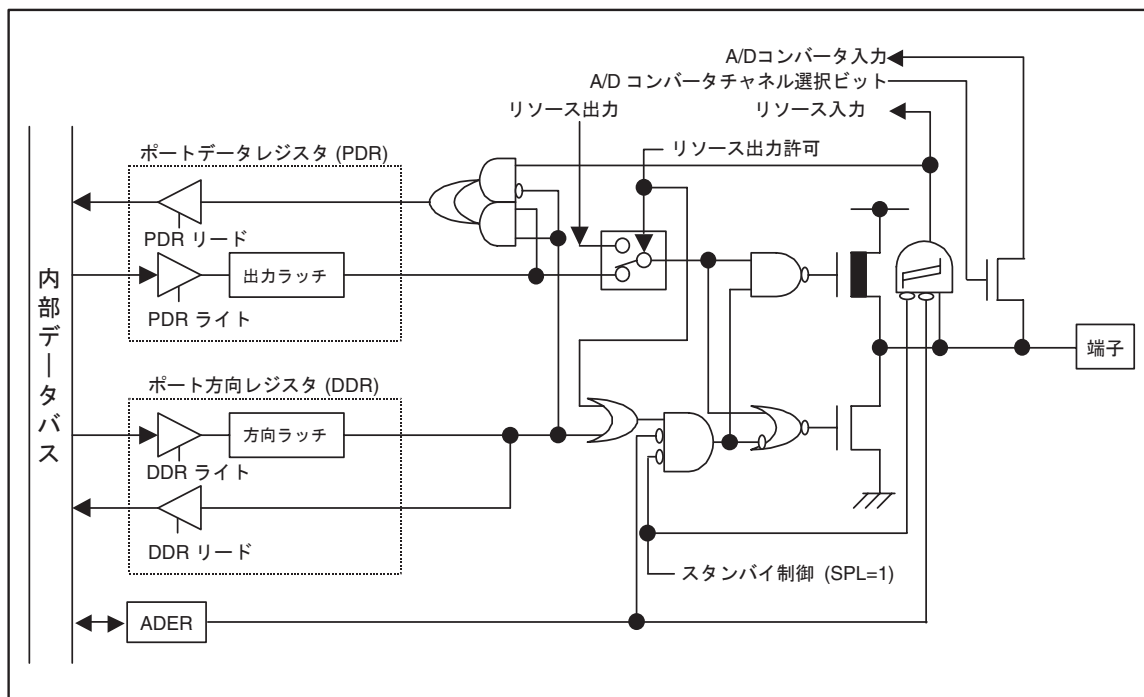
図 9.10-2 P72/SIN1/AN10 のブロックダイアグラム



アナログ入力許可ビットを許可すると、DDR7 レジスタの値にかかわらず、ポートは強制的に A/D コンバータ入力端子となります。

図 9.10-3 にポート 7(P73/SOT1/AN11, P74/SCK1/AN12) 端子のブロックダイアグラムを示します。

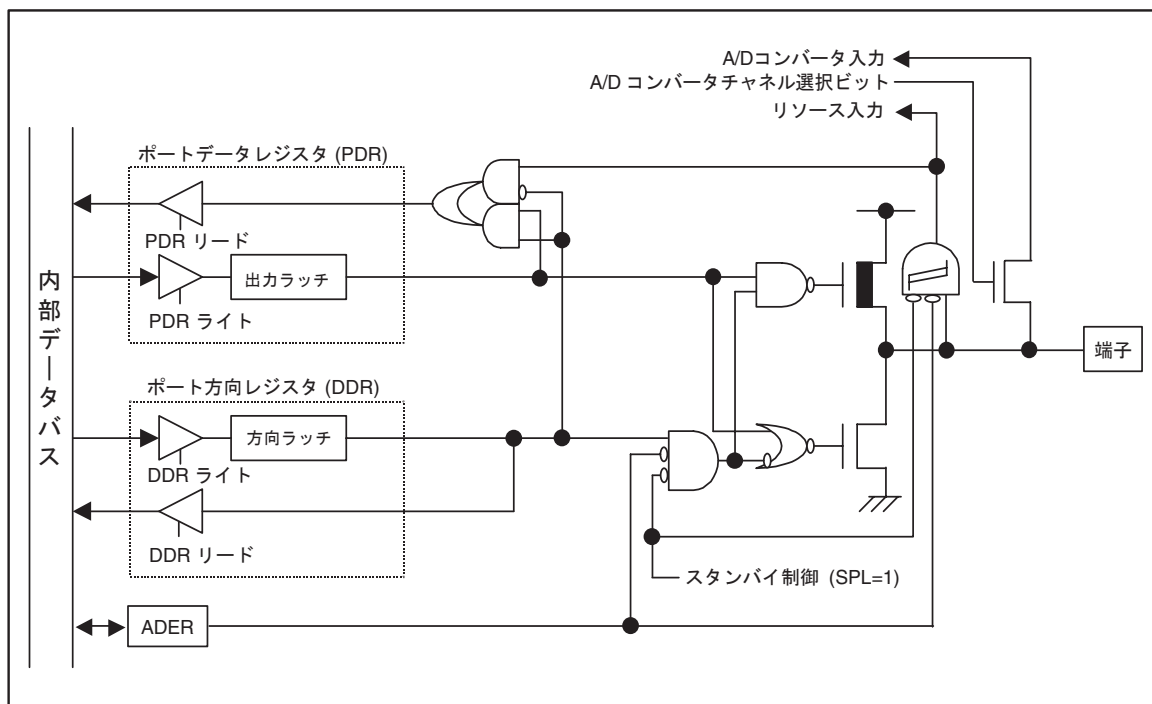
図 9.10-3 P73/SOT1/AN11, P74/SCK1/AN12 端子のブロックダイアグラム



リソース出力許可ビットを許可すると、DDR7 レジスタの値にかかわらず、ポートは強制的にリソース出力端子となります。

図 9.10-4 に P75/FRCK/AN13 ~ P77/IN1/AN15 端子のブロックダイヤグラムを示します。

図 9.10-4 P75/FRCK/AN13 ~ P77/IN1/AN15 端子のブロックダイヤグラム



■ ポート 7 のレジスタ

ポート 7 のレジスタには、PDR7, DDR7, ADER1 があります。各レジスタを構成しているビットは、ポート 7 の端子に 1 対 1 で対応しています。

表 9.10-2 にポート 7 のレジスタと端子の対応を示します。

表 9.10-2 ポート 7 のレジスタと端子の対応

ポート名	関連するレジスタのビットと対応する端子								
ポート 7	PDR7, DDR7	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	ADER1	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	対応する端子	P77	P76	P75	P74	P73	P72	P71	P70

9.10.1 ポート 7 のレジスタ (PDR7, DDR7, ADER1)

ポート 7 のレジスタについて説明します。

■ ポート 7 のレジスタの機能

● ポート 7 データレジスタ (PDR7)

PDR7 レジスタは、ポート 7 の各端子の状態を示します。

● ポート 7 データ方向レジスタ (DDR7)

DDR7 レジスタは、ポート 7 の各端子（ビット）のデータ入出力方向を指定します。DDR7 レジスタのいずれかのビットが "1" の場合は、対応するポート（端子）は出力ポートとして設定されます。いずれかのビットが "0" の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

- D/A コンバータ出力許可ビットまたは A/D 入力許可ビットが設定された場合、DDR7 レジスタの設定値に関係なく、対応するポートは D/A コンバータ出力端子または A/D コンバータ入力端子となります。
- 汎用入出力ポートとして使用するには、対応する A/D コンバータ入力許可レジスタのビットに "0" を設定し、汎用入出力ポートとしてください。

● アナログ入力許可レジスタ 1 (ADER1)

ADER1 レジスタの各ビットは、対応するポート 7 端子を汎用入出力ポートとして使用するかアナログ入力端子として使用するかを指定します。ADE ビットに "1" を設定すると、対応する端子をアナログ入力として使用します。ADE ビットに "0" を設定すると、対応する端子を汎用入出力として使用します。

< 注意事項 >

ポートを入力モードに設定して中間レベルの信号が入力されると、入力リーク電流が発生します。したがって、端子をアナログ入力として使用する場合は、対応する ADE ビットには必ず "1" を設定してください。

< 参考 >

MCU がリセットされた場合、DDR7 レジスタは "0" にクリアされ、ADER0 レジスタには "1"（アナログ入力として使用する）が設定されます。

表 9.10-3 にポート 7 のレジスタの機能を示します。

表 9.10-3 ポート 7 のレジスタの機能

レジスタ	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート 7 データレジスタ (PDR7)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、端子が出力ポートとして機能する場合は、端子は "L" レベルに設定されます	R/W	000007 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力ラッチへ "1" がロードされ、端子が出力ポートとして機能する場合は、端子は "H" レベルに設定されます			
ポート 7 データ方向レジスタ (DDR7)	0	方向ラッチは "0"	出力バッファが "OFF" され、ポートが入力モードになります	R/W	000017 _H	00000000 _B
	1	方向ラッチは "1"	出力バッファが "ON" され、ポートが出力モードになります			
A/D 入力許可 レジスタ 1 (ADER1)	0	ポート入出力モード		R/W	0000D0 _H	11111111 _B
	1	アナログ入力モード				

R/W: リード / ライト可能

X: 不定

9.10.2 ポート7の動作説明

ポート7の動作を説明します。

■ ポート7の動作

● 出力モード時のポート動作

- ADER1 レジスタのいずれかのビットに "0" を設定すると、対応するポートの端子はポート入出力モードになります。
- DDR7 レジスタのいずれかのビットに "1" を設定すると、対応するポートの端子は出力モードになります。
- 出力モード時において PDR7 レジスタに書き込まれたデータは、PDR7 の出力ラッチにおいて保持され、ポート端子へ出力されます。
- PDR7 レジスタを読み出すと、ポート端子における値 (PDR7 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力モード時のポート動作

- ADER1 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子はポート入出力モードになります。
- DDR7 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- 入力モードで PDR7 レジスタに書き込まれたデータは、PDR7 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR7 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● アナログ入力としてのポート動作

ポート端子をアナログ入力として使用する場合は、対応する ADE ビットに "1" を書き込んでください。それによって、端子は汎用ポート端子としては動作せず、アナログ入力端子として動作します。この状態で PDR7 レジスタを読み出すと "0" が読み出されません。

● リソース出力としてのポート動作

リソース出力許可ビットを許可にすると、ポートはリソース出力として使用できます。入力と出力の切換えを指定する場合、リソース許可ビットの状態が優先されます。すなわち、たとえ DDR7 レジスタのいずれかのビットが "0" であっても、リソース出力が許可になっている場合は、対応するポートの端子はリソース出力として使用されます。リソース出力が許可であっても端子の値を読み出すことはできるので、リソースの出力値を読み出すことができます。

● リソース入力としてのポート動作

ポートをリソース入力として使用する場合、端子の値は常にリソースの入力値となります。リソースの外部信号として使用するには、DDR7 レジスタに "0" を設定し、ポートを入力モードにしてください。

● リセット後のポート動作

- MCU がリセットされた場合、DDR7 レジスタは "0" に ADER1 レジスタは "1" に初期化され、ポートはアナログ入力モードになります。汎用ポートとして使用する場合は、事前に ADER1 レジスタに "0" を書き込んでポート入出力モードにしてください。
- MCU がリセットされた場合、DDR7 レジスタは "0" に初期化されます。その結果、出力バッファが "OFF" になり（入出力モードは、入力に変化）、端子はハイインピーダンス状態になります。
- PDR7 レジスタは、MCU がリセットされても初期化されません。したがって、ポートを出力モード時に使用する場合は、出力データを PDR7 レジスタに設定した後、出力モードは DDR7 レジスタにおいて設定しなければなりません。

● ストップモード、タイムベースタイマモードの動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は、端子はハイインピーダンス状態になります。これは、出力バッファが DDR7 レジスタの値とは無関係に "OFF" されるからです。なお、開放回路によるリークを防ぐため、入力は "H" または "L" レベルに固定してあります。

表 9.10-4 にポート 7 の端子状態を示します。

表 9.10-4 ポート 7 の端子状態

端子	通常動作	スリープモード	ストップモード、 タイムベースタイマモード (SPL = 0)	ストップモード、 タイムベースタイマモード (SPL = 1)
P70/DA0/AN8 ~ P77/IN1/AN15	汎用入出力ポート	汎用入出力ポート	汎用入出力ポート	入力禁止 / 出力は Hi-Z 状態

SPL: 低消費電力モード制御レジスタ状態指定ビット (LPMCR)

Hi-Z: ハイインピーダンス

9.11 ポート 8

ポート 8 は汎用入出力ポートですが、リソース入出力と兼用しています。各ポートの端子は、入出力ポートとリソース入出力の間で切換えることができます。本節は、汎用入出力ポート機能を中心に、ポート 8 の構成と端子一覧、端子のブロックダイアグラム、対応するレジスタについて説明します。

■ ポート 8 の構成

ポート 8 は、以下から構成されています。

- ・ 汎用入出力端子 / リソース入出力端子 (P80 / IN2 ~ P87/RTO5)
- ・ ポート 8 データレジスタ (PDR8)
- ・ ポート 8 データ方向レジスタ (DDR8)

■ ポート 8 の端子

ポート 8 は、リソースの入出力端子としても使用します。したがって、これらの端子は、リソースの入出力端子として使用した場合は、汎用入出力ポートとしては使用できません。

表 9.11-1 にポート 8 の端子一覧を示します。

表 9.11-1 ポート 8 の端子

ポート	端子	ポート機能		リソース機能		入出力形式		入出力回路形式
						入力	出力	
ポート 8	P80/IN2	P80	汎用 入出力	IN2	入力キャプチャチャネル 2	CMOS (ヒステリシス)	CMOS	F
	P81/TIN3	P81		IN3	入力キャプチャチャネル 3			
	P82/RTO0	P82		RTO0	波形ジェネレータ 出力 0			L
	P83/RTO1	P83		RTO1	波形ジェネレータ 出力 1			
	P84/RTO2	P84		RTO2	波形ジェネレータ 出力 2			
	P85/RTO3	P85		RTO3	波形ジェネレータ 出力 3			
	P86/RTO4	P86		RTO4	波形ジェネレータ 出力 4			
	P87/RTO5	P87		RTO5	波形ジェネレータ 出力 5			

入出力回路形式については「1.7 入出力回路形式」を参照してください。

■ ポート 8 の端子のブロックダイアグラム

図 9.11-1 に P80/IN2 および P81/IN3 端子のブロックダイアグラムを示します。

図 9.11-1 P80/IN2 および P81/IN3 端子のブロックダイアグラム

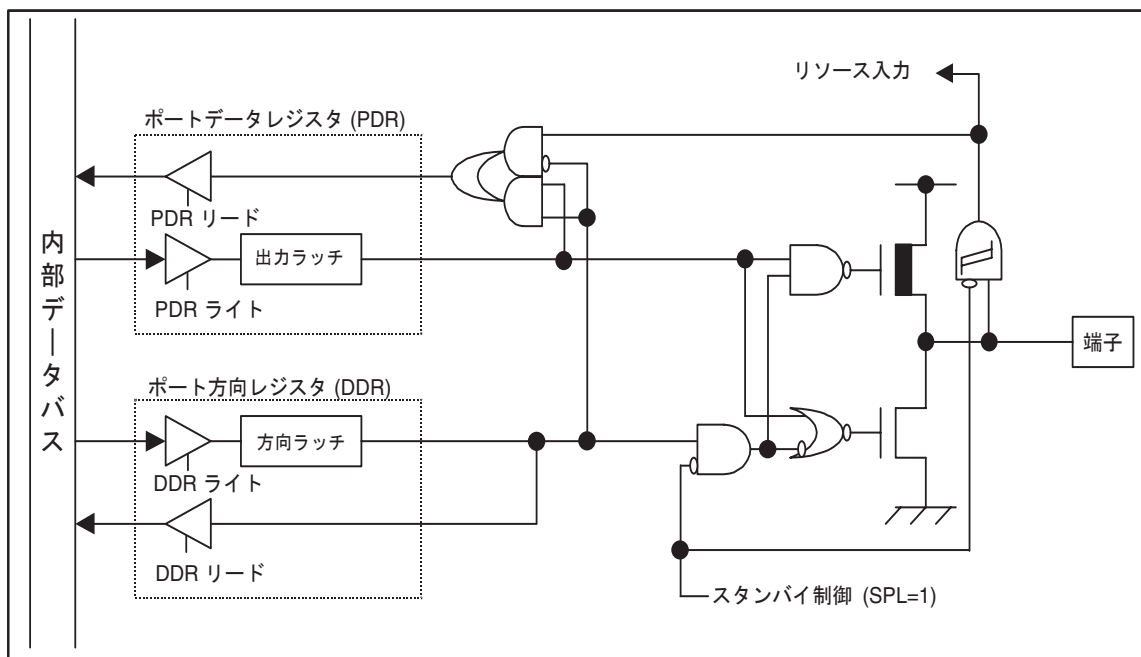
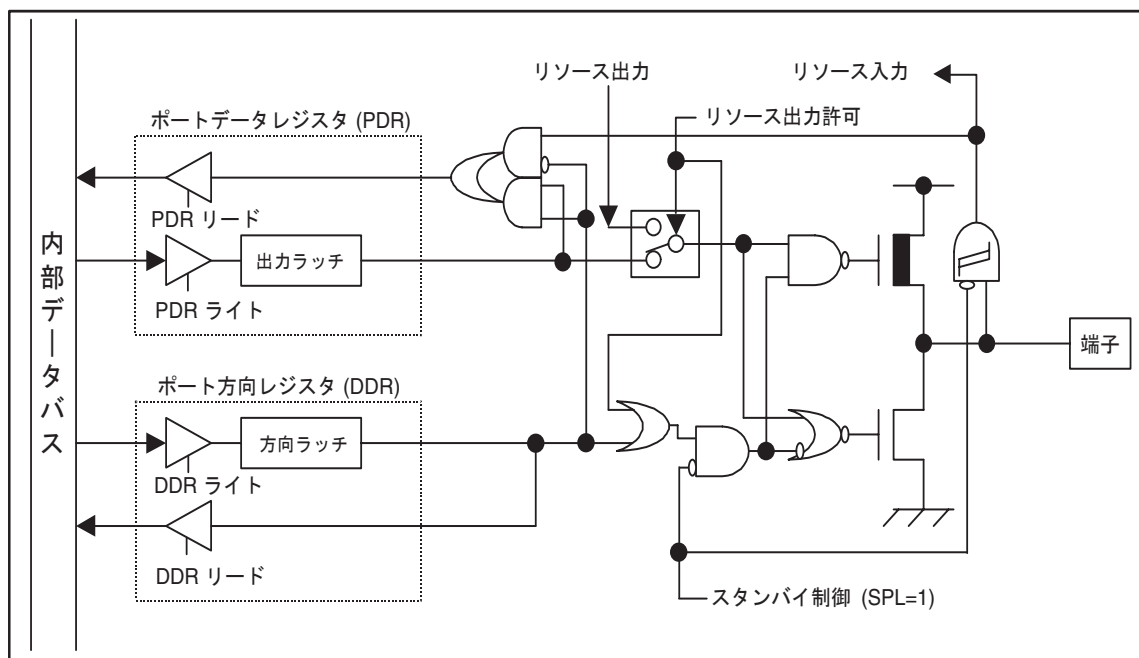


図 9.11-2 に P82/RT00 ~ P87/RT05 端子のブロックダイアグラムを示します。

図 9.11-2 P82/RT00 ~ P87/RT05 端子のブロックダイアグラム



リソース出力許可ビットを設定すると、ポートは、DDR8 レジスタの値に関係なくリソースの出力端子として機能します。

■ ポート 8 のレジスタ

ポート 8 のレジスタには、PDR8 と DDR8 があります。各レジスタを構成しているビットは、ポート 8 の端子に 1 対 1 で対応しています。

表 9.11-2 にポート 8 のレジスタと端子の対応を示します。

表 9.11-2 ポート 8 の端子と対応するレジスタビット

ポート名	関連するレジスタのビットと対応する端子								
ポート 8	PDR8, DDR8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	対応する端子	P87	P86	P85	P84	P83	P82	P81	P80

9.11.1 ポート 8 のレジスタ (PDR8, DDR8)

ポート 8 のレジスタについて説明します。

■ ポート 8 のレジスタの機能

● ポート 8 データレジスタ (PDR8)

PDR8 レジスタは、ポート 8 の各端子の状態を示します。

● ポート 8 データ方向レジスタ (DDR8)

DDR8 レジスタは、ポート 8 の各端子（ビット）のデータ入出力方向を指定します。DDR8 レジスタのいずれかのビットが "1" の場合は、対応するポート（端子）は出力ポートとして設定されます。いずれかのビットが "0" の場合は、対応するポート（端子）は入力ポートとして設定されます。

< 注意事項 >

- リソースの出力と兼用している端子は、その端子に対応するリソース出力許可ビットが設定されている限り、ポートは DDR8 レジスタの値とは無関係にリソースの出力端子として機能します。
- 入力端子と兼用しているリソースを使用するには、各リソース入力端子に対応するポート方向レジスタのビットに "0" を設定して、ポートを入力モードにしてください。

表 9.11-3 にポート 8 のレジスタの機能を示します。

表 9.11-3 ポート 8 のレジスタの機能

レジスタ名	データ	読出し時	書込み時	リード ライト	アドレス	初期値
ポート 8 データレジスタ (PDR8)	0	端子は "L" レベル	出力ラッチへ "0" がロードされ、 端子が出力ポートとして機能する 場合は、端子は "L" レベルに 設定されます	R/W	000008 _H	XXXXXXXX _B
	1	端子は "H" レベル	出力バッファが "OFF" され、 ポートが入力モードになります			
ポート 8 データ方向レジ スタ (DDR8)	0	方向ラッチは "0"	出力バッファが "OFF" され、 ポートが入力モードになります	R/W	000018 _H	00000000 _B
	1	方向ラッチは "1"	出力バッファが "OFF" され、 ポートが入力モードになります			

R/W: リード / ライト可能

X: 不定

9.11.2 ポート 8 の動作説明

ポート 8 の動作を説明します。

■ ポート 8 の動作

● 出力モード時のポート動作

- DDR8 レジスタのいずれかのビットに "1" を設定すると、対応するポートの端子は出力モードになります。
- 出力モード時において PDR8 レジスタに書き込まれたデータは、PDR8 の出力ラッチにて保持され、端子へ出力されます。
- PDR8 レジスタを読み出すと、ポート端子における値 (PDR8 の出力ラッチと同じ値) を読み出すことができます。

< 注意事項 >

リードモディファイライト命令 (ビット設定命令など) を PDR レジスタと共に使用すると、このレジスタのターゲットビットは指定された値に設定されます。DDR レジスタで出力に指定されているビットは影響を受けませんが、入力に指定されているビットは、端子からの入力値は出力ラッチへ書き込まれ、そのまま出力されます。したがって、ビットのモードを入力から出力に切換える前に、出力データを PDR レジスタに書き込み、次に DDR レジスタを出力モードに設定してください。

● 入力モード時のポート動作

- DDR8 レジスタのいずれかのビットを "0" に設定すると、対応するポートの端子は入力モードになります。
- 入力モード時は、出力バッファは "OFF" になり、端子はハイインピーダンス状態になります。
- 入力モードで PDR8 レジスタに書き込まれたデータは、PDR8 の出力ラッチで保持されますが、ポートの端子へは出力されません。
- PDR8 レジスタを読み出すとポートの端子レベル ("0" または "1") を読み出すことができます。

● リソース出力としてのポート動作

リソース出力許可ビットを許可にすると、ポートはリソース出力として使用できます。入力と出力の切換えを指定する場合、リソース許可ビットの状態が優先されます。すなわち、たとえ DDR8 レジスタのいずれかのビットが "0" であっても、リソース出力が許可になっている場合は、対応するポートの端子はリソース出力として使用されます。リソース出力が許可であっても端子の値を読み出すことはできるので、リソースの出力値を読み出すことができます。

● リソース入力としてのポート動作

ポートをリソース入力として使用する場合，端子の値は常にリソースの入力値となります。リソースの外部信号として使用するには，DDR8 レジスタに "0" を設定し，ポートを入力モードにしてください。

● リセット後のポート動作

- MCU がリセットされた場合，DDR8 レジスタは "0" に初期化されます。その結果，出力バッファが "OFF" になり（入出力モードは，入力に変化），端子はハイインピーダンス状態になります。
- PDR8 レジスタは，MCU がリセットされても初期化されません。したがって，ポートを出力モード時に使用する場合は，出力データを PDR8 レジスタに設定した後，出力モードは DDR8 レジスタにおいて設定しなければなりません。

● ストップモードまたはタイムベースタイマモード時のポート動作

ストップモードまたはタイムベースタイマモードに遷移されたときに低消費電力モード制御レジスタ (LPMCR) の端子状態指定ビット (SPL) がすでに "1" の場合は，端子はハイインピーダンス状態になります。これは，出力バッファが DDR8 レジスタの値とは無関係に "OFF" されるからです。なお，開放回路によるリークを防ぐため，入力は "H" または "L" レベルに固定してあります。

表 9.11-4 にポート 8 の端子状態を示します。

表 9.11-4 ポート 8 の端子状態

端子	通常動作	スリープモード	ストップモード， タイムベースタイマモード (SPL = 0)	ストップモード， タイムベースタイマモード (SPL = 1)
P80/IN2 ~ P87/RT05	汎用入出力 ポート	汎用入出力 ポート	汎用入出力ポート	入力禁止 / 出力は Hi-Z 状態

SPL: 低消費電力モード制御レジスタの端子状態指定ビット (LPMCR)
Hi-Z: ハイインピーダンス

第10章

タイムベースタイマ

タイムベースタイマの機能と動作について説明します。

- 10.1 タイムベースタイマの概要
- 10.2 タイムベースタイマの構成
- 10.3 タイムベースタイマ制御レジスタ (TBTC)
- 10.4 タイムベースタイマの割込み
- 10.5 タイムベースタイマの動作説明
- 10.6 タイムベースタイマ使用上の注意

10.1 タイムベースタイマの概要

タイムベースタイマは、内部カウントクロック（原発振を 2 分周したもの）に同期してカウントアップする 18 ビットフリーランカウンタ（タイムベースカウンタ）です。タイムベースタイマは、4 つのインターバル時間を選択できるインターバルタイマ機能を備えています。

またタイムベースタイマは、発振安定待ち時間用のタイマおよびウォッチドッグタイマへクロックを供給するための機能も備えています。

■ インターバルタイマ機能

インターバルタイマ機能は、割込み要求をある一定のインターバルで繰り返し生成します。

- タイムベースカウンタのインターバルタイマビットがオーバーフローすると、割込み要求が発生します。
- インターバル時間ビットは、4 種類から選択できます。表 10.1-1 にタイムベースタイマのインターバル時間を示します。

表 10.1-1 タイムベースタイマのインターバル時間

内部カウントクロック周期	インターバル時間
2/HCLK(0.5μs)	$2^{12}/\text{HCLK}$ (約 1.0ms)
	$2^{14}/\text{HCLK}$ (約 4.1ms)
	$2^{16}/\text{HCLK}$ (約 16.4ms)
	$2^{19}/\text{HCLK}$ (約 131.1ms)

HCLK: 発振クロック周波数

() 内の数値は、発振クロック周波数 4MHz で動作させた場合のインターバル時間です。

■ クロック供給機能

クロック供給機能は、発振安定待ち時間用のタイマおよび一部のリソースへクロックを供給します。

タイムベースタイマから各リソースへ供給されるクロック周期を表 10.1-2 に示します。

表 10.1-2 タイムベースタイマから供給されるクロック周期

クロック供給先	クロック周期	備考
発振安定待ち時間	$2^{13}/\text{HCLK}$ (約 2.0 ms)	セラミック振動子の発振安定待ち時間
	$2^{15}/\text{HCLK}$ (約 8.2 ms)	水晶振動子の発振安定待ち時間
	$2^{18}/\text{HCLK}$ (約 65.4 ms)	
ウォッチドッグタイマ	$2^{12}/\text{HCLK}$ (約 1.0 ms)	ウォッチドッグタイマのカウントアップクロック
	$2^{14}/\text{HCLK}$ (約 4.1 ms)	
	$2^{16}/\text{HCLK}$ (約 16.4 ms)	
	$2^{19}/\text{HCLK}$ (約 131.1ms)	

HCLK: 発振クロック周波数

() 内は発振クロック周波数が 4MHz で動作時の算出例です。

発振開始時は発振周期が不安定なので、上記の発振安定待ち時間は目安として使用してください。

10.2 タイムベースタイマの構成

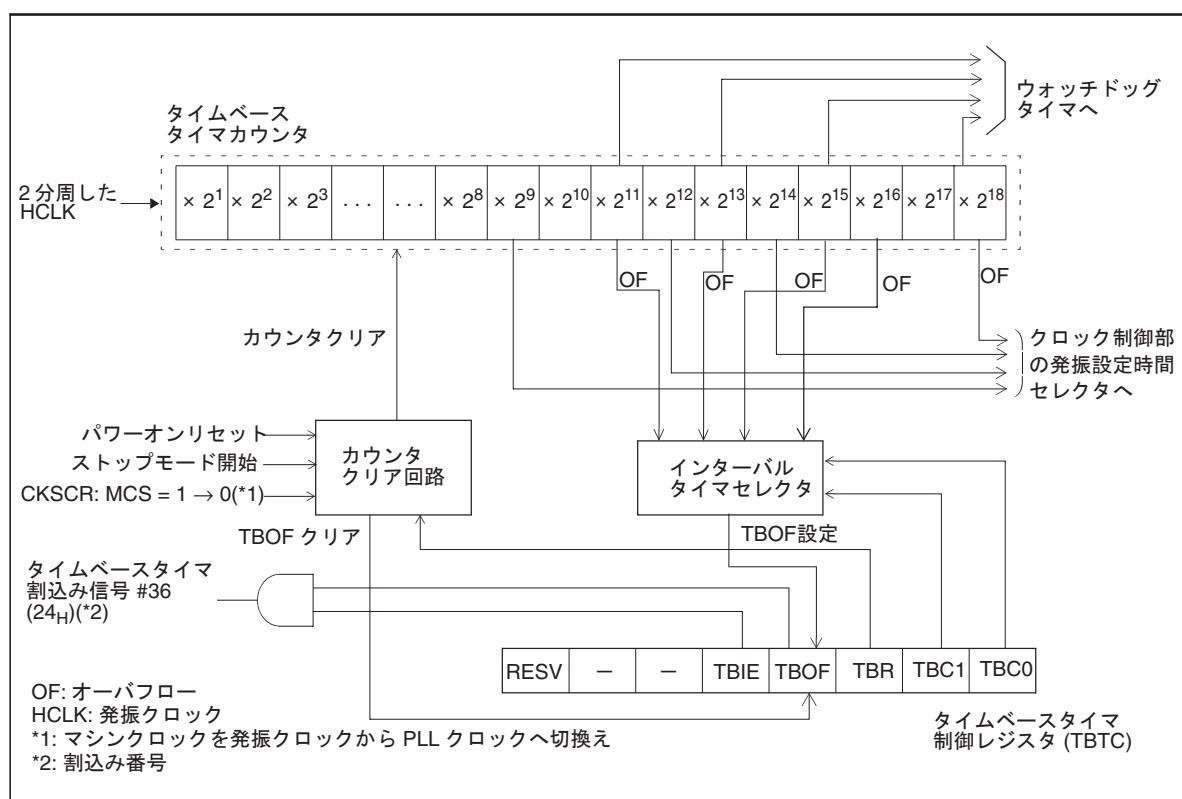
タイムベースタイマは、以下の 4 つのブロックで構成されています。

- タイムベースタイマカウンタ
- カウンタクリア回路
- インターバルタイマセクタ
- タイムベースタイマ制御レジスタ (TBTC)

■ タイムベースタイマのブロックダイアグラム

図 10.2-1 にタイムベースタイマのブロックダイアグラムを示します。

図 10.2-1 タイムベースタイマのブロックダイアグラム



- タイムベースタイマカウンタ

18 ビットのアップカウンタは、発振クロック (HCLK) の 2 分周クロックをカウントクロックとして使用します。

- カウンタクリア回路

TBTC の TBR ビットへの "0" 書込み、パワーオンリセットおよびストップモードへの遷移 (LPMCR : STP = 1) でカウンタをクリアするのに使用される回路です。

- インターバルタイマセレクタ

タイムベースタイマカウンタの 4 種類の出力から 1 つを選択します。選択したビットのオーバーフローが割込み要因となります。

- タイムベースタイマ制御レジスタ (TBTC)

インターバル時間の選択、タイムベースタイマのカウンタのクリア、割込み要求の制御および状態確認を行います。

10.3 タイムベースタイマ制御レジスタ (TBTC)

タイムベースタイマ制御レジスタ (TBTC) では、インターバル時間の選択、カウンタのクリア、割り込み要求の制御、および状態の確認を行います。

■ タイムベースタイマ制御レジスタ (TBTC)

図 10.3-1 タイムベースタイマ制御レジスタ (TBTC)

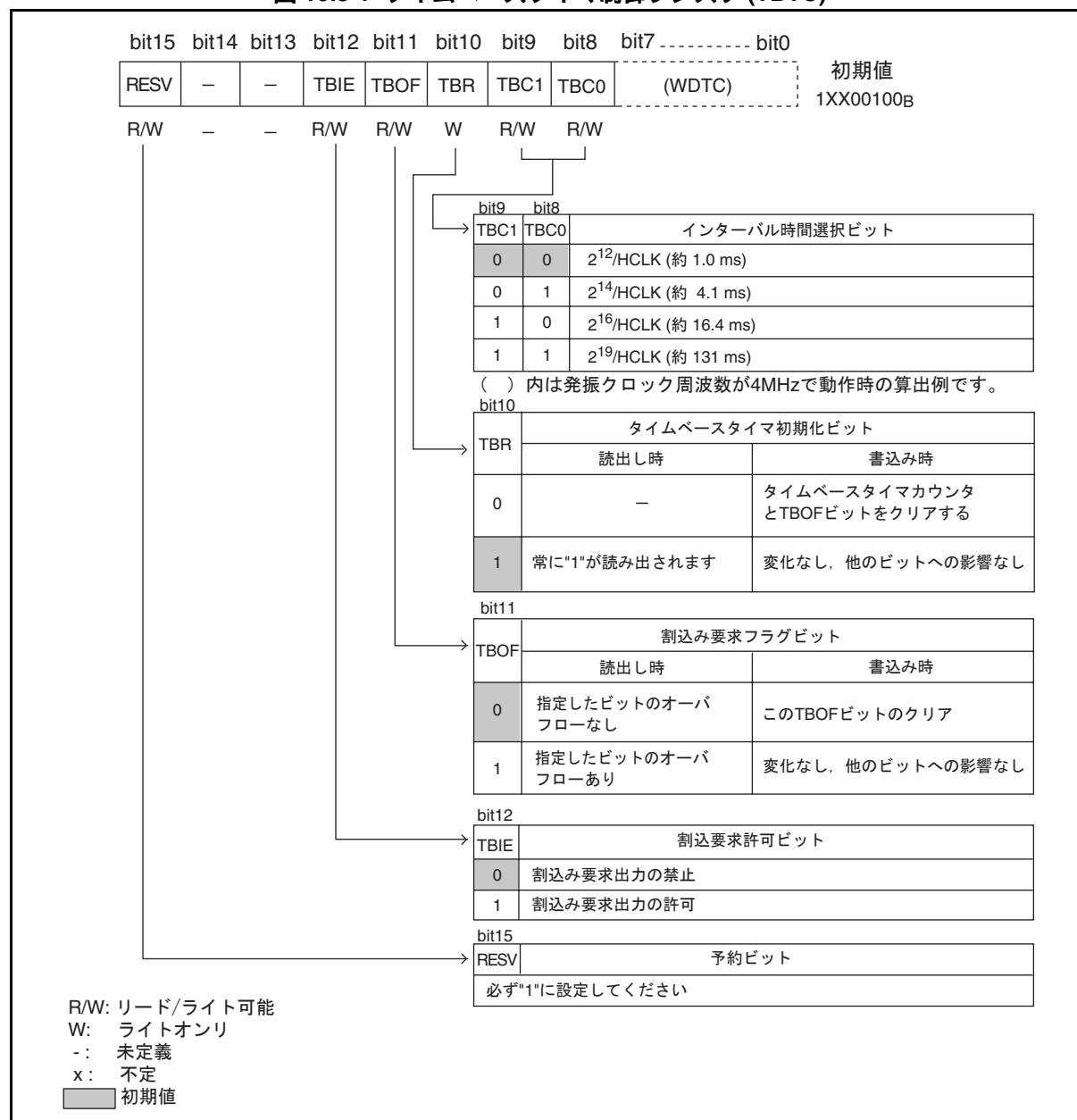


表 10.3-1 タイムベースタイマ制御レジスタ (TBTC) の各ビットの機能説明

ビット名		機能
bit15	RESV: 予約ビット	(注意事項) このビットには必ず "1" に設定してください。
bit13, bit14	未定義ビット	読み出した場合 : 値は不定です。 書き込んだ場合 : 動作への影響はありません。
bit12	TBIE: 割り込み要求許可ビット	CPU への割り込み要求を許可または禁止します。 このビットと割り込み要求フラグビット (TBOF) ビットが "1" の場合、 割り込み要求を出力します。
bit11	TBOF: 割り込み要求フラグビット	このビットは、タイムベースタイマカウンタの指定するビットが オーバフローすると、"1" にセットされます。 このビットと割り込み要求許可ビット (TBIE) が "1" に設定されている 場合、割り込み要求が発生します。 "0" に設定した場合: この TBOF ビットはクリアされます。 "1" に設定した場合: 変化せず、動作に影響しません。 (注意事項) • TBOF ビットをクリアするには、TBIE ビットまたはプロセッサス テータス (PS) の ILM ビットの指定でタイムベースタイマ割り込みを 禁止してください。 • TBOF ビットのクリアは、"0" 書込み、ストップモードへの遷移、 TBR ビットによるタイムベースタイマのクリア、またはリセット によりクリアされます。
bit10	TBR: タイムベースタイマ初期化 ビット	タイムベースタイマカウンタをクリアします。 "0" に設定した場合: タイムベースタイマカウンタがクリアされ、 TBOF ビットもクリアされます。 "1" に設定した場合: 変化せず、動作に影響しません。 リードした場合: 常に "1" が読み出されます。
bit9, bit8	TBC1, TBC0: インターバル時間選択 ビット	インターバル時間を選択します。 • タイムベースタイマカウンタのインターバルタイマ用のビットが 指定されます。 • 4 種類のインターバル時間が選択できます。

10.4 タイムベースタイマの割込み

タイムベースタイマは、タイムベースタイマカウンタを指定するビットがオーバーフローする場合、割込み要求を発生できます。

■ タイムベースタイマの割込み

割込み要求フラグビット (TBTC : TBOF) に "1" が設定されるのは、タイムベースタイマカウンタが内部カウントクロックでカウントアップし、選択されたインターバルタイマビットのオーバーフローが発生した場合です。割込み要求フラグビットに "1" が設定されたとき、割込み要求許可ビットが許可になっている (TBTC : TBIE = 1) 場合は、割込み要求 (#36) が CPU で生成されます。割込み処理ルーチンで TBOF ビットに "0" を書き込むと、割込み要求はクリアされます。指定されたビットのオーバーフローが発生すると、TBIE ビット値とは無関係に TBOF ビットが "1" に設定されます。

< 注意事項 >

割込み要求フラグビット (TBTC : TBOF) のクリアは、TBIE ビットまたはプロセッサステータス (PS) ILM ビットの設定によりタイムベースタイマ割込みを禁止している間に行ってください。

< 参考 >

TBOF ビットが "1" で、TBIE ビットの状態が禁止から許可に遷移する ("0" → "1") と、割込み要求が直ちに発生します。

■ タイムベースタイマの割込みと EI²OS の対応

表 10.4-1 にタイムベースタイマの割込みと EI²OS を示します。

表 10.4-1 タイムベースタイマの割込みと EI²OS

割込み番号	割込みレベル設定レジスタ		ベクタテーブルのアドレス			EI ² OS
	レジスタ名	アドレス	下位	中位	上位	
#36(24H)	ICR12	0000BCH	FFFF6Ch	FFFF6Dh	FFFF6Eh	Δ

Δ : ICR を共有する割込み要因を使用しない場合に使用可能になります。

< 注意事項 >

ICR12 は、タイムベースタイマ割込みとインプットキャプチャチャネル 2/3 割込みに共用されます。割込みは 2 つのアプリケーションに対して使用できますが、これらの割込みレベルは同一です。

10.5 タイムベースタイマの動作説明

タイムベースタイマは、インターバルタイマ機能とクロック供給機能 (各リソースへクロックを供給する) を備えています。

■ インターバルタイマ機能 (タイムベースタイマ)

インターバルタイマ機能は、割込み要求を各インターバルで生成します。

すべてのタイマをインターバルタイマとして動作させるには、図 10.5-1 のように設定する必要があります。

図 10.5-1 タイムベースタイマの設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TBTC	RESV	-	-	TBIE	TBOF	TBR	TBC1	TBC0	-	-	-	-	-	-	-	WDTC
	1				0	0										

: 使用
 0 : "0" 設定
 1 : "1" 設定

- ・ タイムベースタイマカウンタは、クロックが発振されている限り内部クロック (発振クロックを 2 分周したもの) と同期してカウントアップを継続します。
- ・ カウンタがクリアされると (TBR = 0), カウンタは "0" からカウントアップを開始します。インターバルタイマビットのオーバフローが発生すると、割込み要求フラグビット (TBOF) に "1" が設定されます。このとき割込み要求出力がすでに許可になっている場合は (TBIE = 1), クリアされた時間を基準に選択された各インターバルで割込みを生成します。
- ・ タイムベースタイマがクリアされるため、インターバル時間は設定された時間より長くなることがあります。

■ 発振安定待ち時間用タイマの機能

タイムベースタイマは、発振クロックおよび PLL クロックの発振安定待ち時間のタイマとしても使用されます。

発振安定待ち時間は、カウンタが "0" (カウントクリア) からカウントアップを開始してから発振安定待ち時間ビットのオーバフローが発生するまでの時間をインターバルとして設定されます。タイムベースタイマモードから PLL クロックモードに制御が復帰すると、タイムベースタイマカウンタはクリアされていないので、発振安定待ち時間はカウントの途中から開始します。タイムベースタイマカウンタのクリアと、発振安定待ち時間を表 10.5-1 に示します。

表 10.5-1 タイムベースタイマカウンタのクリア動作と発振安定待ち時間

動作	カウンタ のクリア	TBOF の クリア	発振安定待ち時間
TBTC:TBR への "0" 書込み			-
パワーオンリセット			発振クロック発振安定待ち時間
ウォッチドッグリセット			
ストップモードの解除			発振クロック発振安定待ち時間 (メインクロック モード復帰時)
発振クロックモードから PLL クロックモードへの移行 (MCS = 1 0)			PLL クロック発振安定待ち時間
タイムベースタイマモードの 解除	×	×	PLL クロック発振安定待ち時間 (PLL クロック モード復帰時)
スリープモードの解除	×	×	なし

: あり
× : なし

■ 動作クロック供給

タイムベースタイマは、ウォッチドッグタイマへクロックを供給します。タイムベース
カウンタのクリアは、ウォッチドッグタイマの動作に影響を与えます。

10.6 タイムベースタイマ使用上の注意

割込み要求のクリアおよびタイムベースタイマカウンタのクリアによるリソース機能への影響などの注意点を示します。

■ タイムベースタイマ使用上の注意

● 割込み要求のクリア

タイムベースタイマ制御レジスタの TBOF ビットは、タイムベースタイマ割込みが、TBIE ビットまたはプロセッサステータス (PS) の割込みレベルマスクレジスタ (ILM) によってマスクされている間にクリアしなければなりません。

● タイムベースタイマのクリアによる影響

タイムベースタイマカウンタをクリアすると、以下が影響を受けます。

- タイムベースタイマで使用中のインターバルタイマ機能 (インターバル割込み)
- 使用中のウォッチドッグタイマ

● 発振安定待ち時間用タイマとしてのタイムベースタイマの使用

パワーオンすると、メインクロックの原発振はメインストップモード時に停止します。発振子の動作開始後、タイムベースタイマから供給される動作クロックはメインクロックの発振安定待ち時間を設けるために使用されます。適切な発振安定待ち時間を、メインクロック発振子 (クロック生成部) に接続されている発振子のタイプに基づいて選択しなければなりません。詳細については、「4.5 発振安定待ち時間」をご参照ください。

● タイムベースタイマからクロックが供給されるリソースについての注意

メインクロックソース発振が停止しているモード時は、タイムベースタイマカウンタがクリアされ、タイムベースタイマ動作が停止します。タイムベースタイマカウンタがクリアされると、タイムベースタイマから供給されるクロックはタイムベースタイマの初期状態から供給されます。その結果、「H」レベルが 1/2 サイクル短くなり、「L」レベルが 1/2 サイクル長くなります。ウォッチドッグタイマへのクロックもタイムベースタイマの初期状態から供給されますが、ウォッチドッグタイマは正常サイクルで動作します。ウォッチドッグタイマカウンタのクリアが、タイムベースタイマカウンタのクリアと同時に行為されるためです。

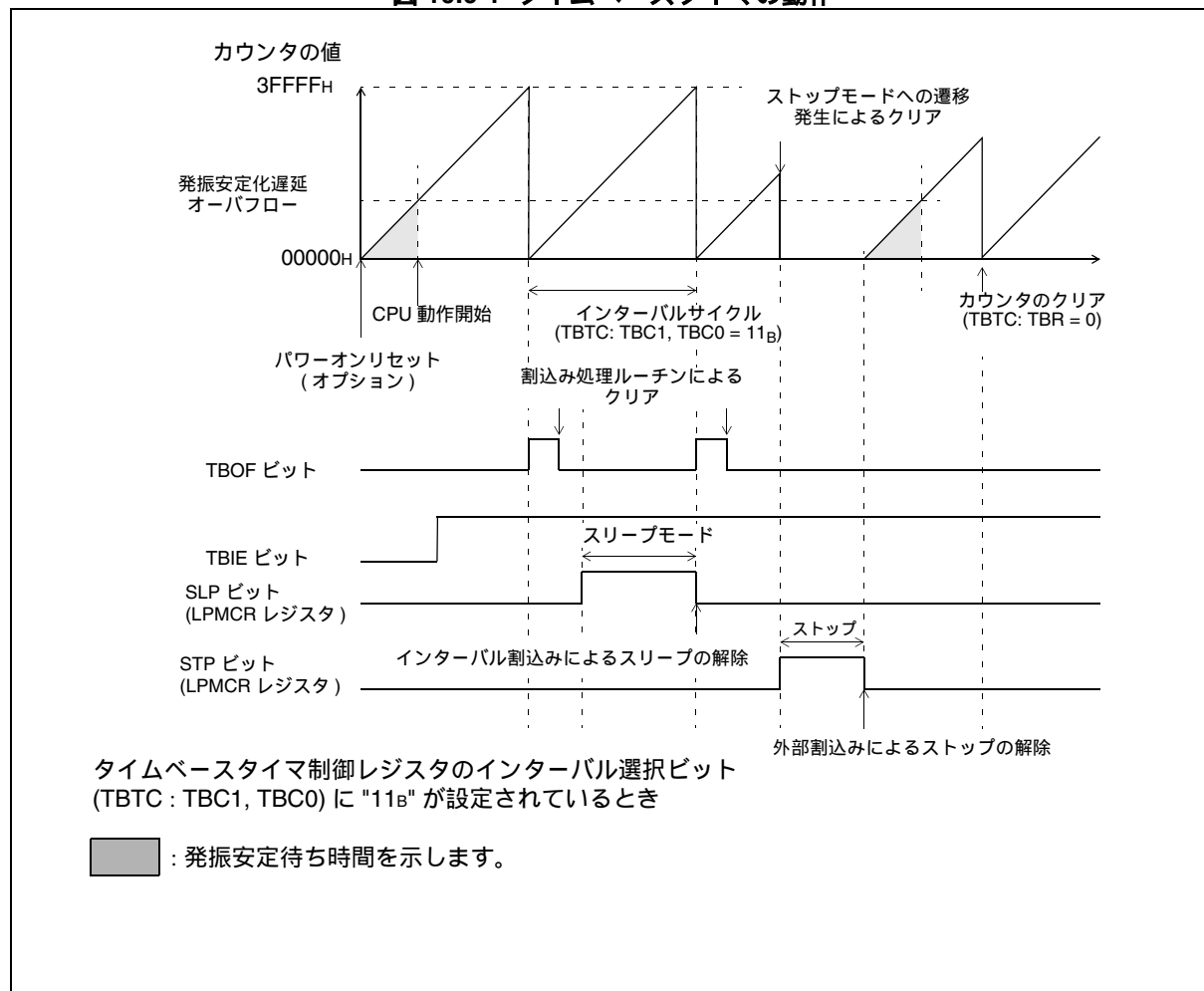
■ タイムベースタイマの動作

以下の動作を図 10.6-1 に示します。

- パワーオンリセットの発生
- インターバルタイマ機能動作時のスリープモードへの遷移
- カウンタクリア要求の発生

ストップモードに入ると、タイムベースタイマがクリアされ、タイムベースタイマの動作が停止します。ストップモードから復帰すると、タイムベースタイマが直ちに発振安定待ち時間のカウントを開始します。

図 10.6-1 タイムベースタイマの動作



第11章

ウォッチドッグタイマ

ウォッチドッグタイマの機能と動作について説明します。

11.1 ウォッチドッグタイマの概要

11.2 ウォッチドッグタイマの構成

11.3 ウォッチドッグタイマ制御レジスタ (WDTC)

11.4 ウォッチドッグタイマの動作説明

11.5 ウォッチドッグタイマ使用上の注意

11.1 ウォッチドッグタイマの概要

ウォッチドッグタイマは、タイムベースタイマから供給されるクロックをカウントクロックとして使用する 2 ビットカウンタです。ウォッチドッグタイマが起動されてからある一定時間内にウォッチドッグタイマがクリアされない場合は、CPU がリセットされます。

■ ウォッチドッグタイマ機能

ウォッチドッグタイマは、プログラムの暴走に対処するためのカウンタです。ウォッチドッグタイマはいったん起動されると、ある一定時間ごとに定期的にクリアする必要があります。プログラムが無限ループに入ってしまいウォッチドッグタイマがある一定時間経過後もクリアされない場合は、CPU にウォッチドッグリセットが生成されます。

表 11.1-1 にウォッチドッグタイマのインターバル時間を示します。ウォッチドッグタイマがクリアされない場合は、最小時間と最大時間の範囲内でウォッチドッグリセットが発生します。カウンタは、この表に示されている最小時間内でクリアしてください。

表 11.1-1 ウォッチドッグタイマのインターバル時間

インターバル時間		
最小 *	最大 *	発振クロックサイクルカウント
約 3.58 ms	約 4.61 ms	$2^{14} \pm 2^{11}$
約 14.33 ms	約 18.3 ms	$2^{16} \pm 2^{13}$
約 57.23 ms	約 73.73 ms	$2^{18} \pm 2^{15}$
約 458.75 ms	約 589.82 ms	$2^{21} \pm 2^{18}$

*: 発振クロック周波数 4MHz で動作させた場合のインターバル時間です。

ウォッチドッグタイマの最大および最小インターバル時間と発振クロックサイクル数は、ウォッチドッグタイマのクリアのタイミングによって異なります。

インターバル時間は、カウントクロック（タイムベースタイマから供給されるクロック）のサイクルの 3.5 ~ 4.5 倍になります。詳細については、「11.4 ウォッチドッグタイマの動作説明」をご参照ください。

< 注意事項 >

ウォッチドッグタイマは、タイムベースタイマの桁上り信号をカウントクロックとする 2 ビットカウンタにより構成されます。タイムベースタイマがクリアされると、ウォッチドッグリセットの発生時間は、設定された時間より長くなる場合があります。

< 参考 >

ウォッチドッグタイマは、起動されるとパワーオンリセットまたはウォッチドッグリセットで初期化された後停止状態になります。ウォッチドッグタイマは、外部端子リセット、ソフトリセット、WTE ビット (ウォッチドッグタイマ制御レジスタ) への書込み、スリープモードまたはストップモードへの遷移によりクリアされます。ただし、ウォッチドッグタイマはクリアされても停止状態にはなりません。

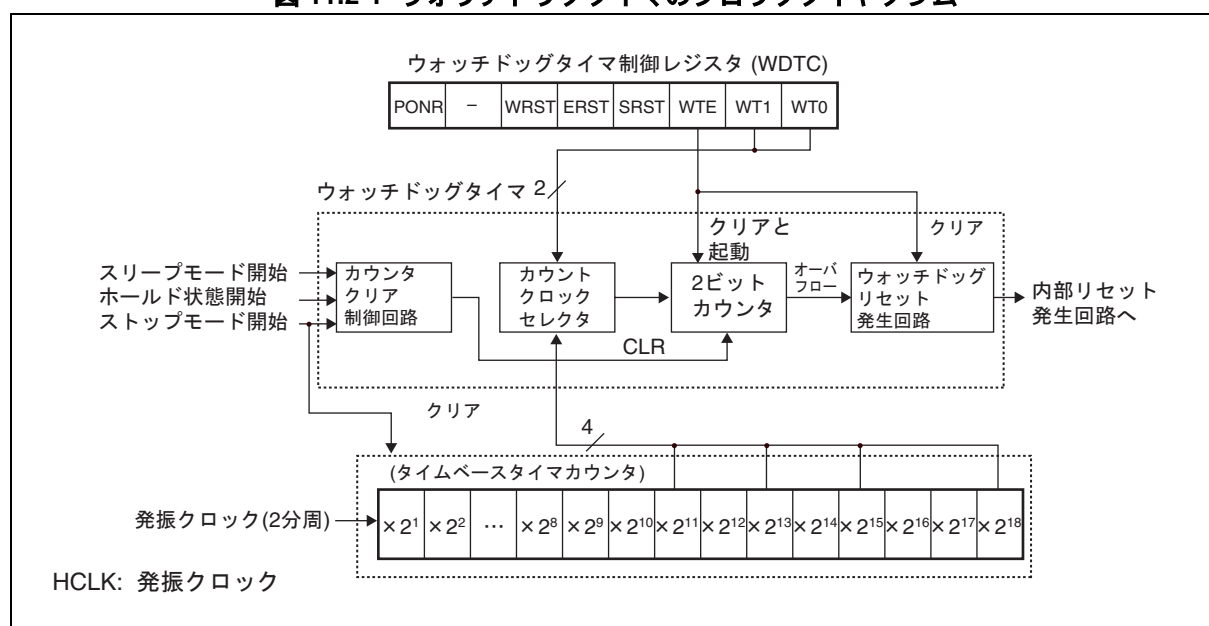
11.2 ウォッチドッグタイマの構成

ウォッチドッグタイマは、以下の 5 つのブロックで構成されています。

- カウントクロックセクタ
- ウォッチドッグカウンタ (2 ビットカウンタ)
- ウォッチドッグリセット発生回路
- カウンタクリア制御回路
- ウォッチドッグタイマ制御レジスタ (WDTC)

■ ウォッチドッグタイマのブロックダイアグラム

図 11.2-1 ウォッチドッグタイマのブロックダイアグラム



● カウントクロックセクタ

この回路は、4 種類のタイムベースタイマ出力からウォッチドッグタイマのカウントクロックを選択するのに使用されます。これにより、ウォッチドッグリセットの発生時間が決まります。

● ウォッチドッグタイマ (2 ビットカウンタ)

この 2 ビットアップカウンタは、タイムベースタイマ出力をカウントクロックとして使用します。

● ウォッチドッグリセット発生回路

ウォッチドッグカウンタのオーバーフローによって、リセット信号を発生します。

● カウンタクリア回路

ウォッチドッグカウンタのクリアと、カウンタの動作または停止を制御します。

● ウォッチドッグタイマ制御レジスタ (WDTC)

ウォッチドッグタイマの起動やクリアを行い、リセット発生要因の保持を行います。

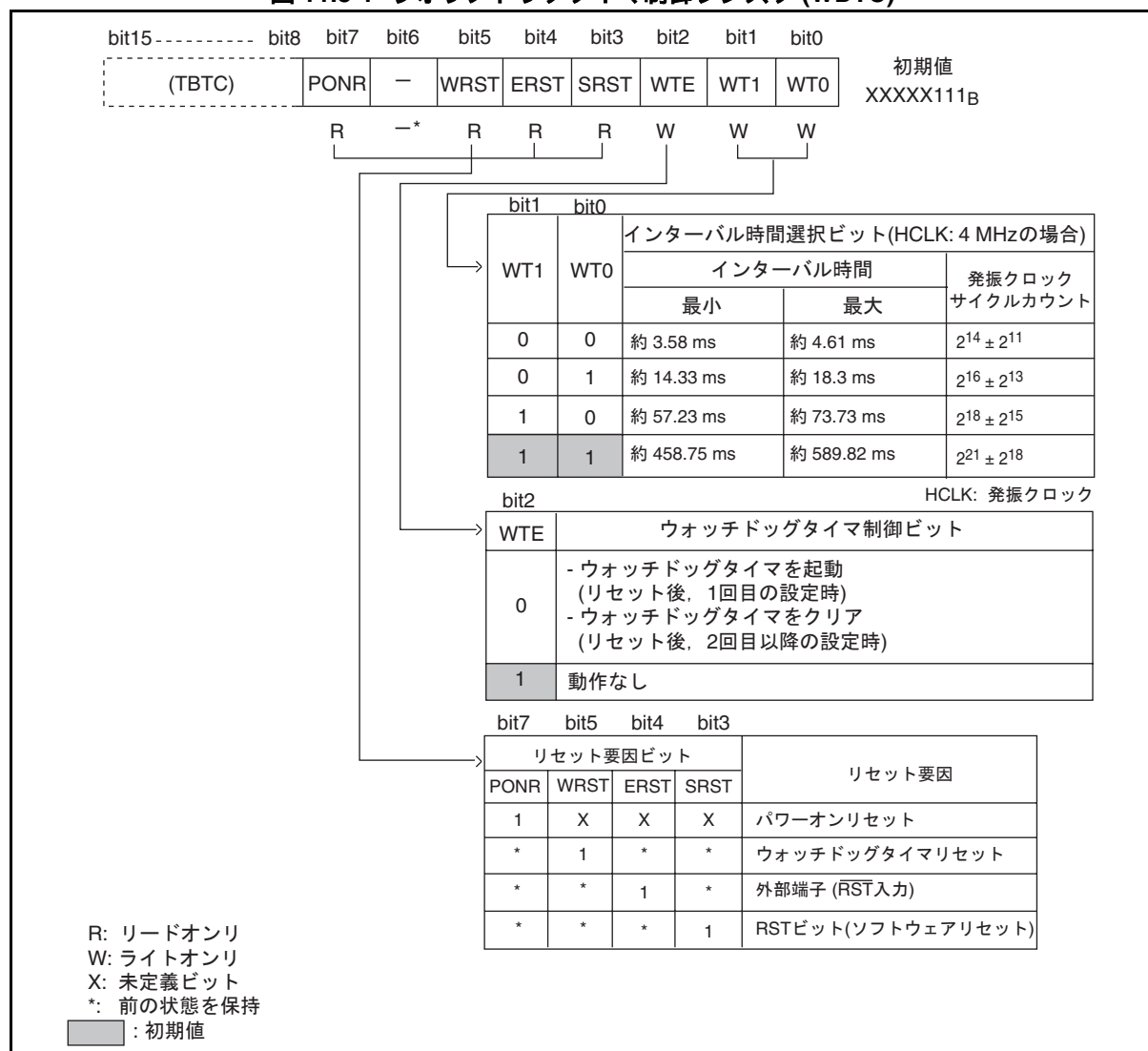
11.3 ウォッチドッグタイマ制御レジスタ (WDTC)

ウォッチドッグタイマ制御レジスタ (WDTC) は、ウォッチドッグタイマの起動、クリア、およびリセット要因の表示を行います。

■ ウォッチドッグタイマ制御レジスタ (WDTC)

図 11.3-1 にウォッチドッグタイマ制御レジスタ (WDTC) を示します。表 11.1-1 にはウォッチドッグタイマ制御レジスタ (WDTC) の各ビットの機能説明を示します。

図 11.3-1 ウォッチドッグタイマ制御レジスタ (WDTC)



インターバル時間はカウントクロック (タイムベースタイマの出力値) のサイクルの 3.5 ~ 4.5 倍となります。詳細は「11.4 ウォッチドッグタイマの動作説明」をご参照ください。

表 11.3-1 ウォッチドッグタイマ制御レジスタ (WDTC) の機能

ビット名		機能
bit7, bit5, bit4, bit3	PONR, WRST, ERST, SRST: リセット要因ビット	<ul style="list-style-type: none"> リセット要因を示すリードオンリのビットです。複数のリセット要因が発生した場合は、各リセット要因に対応するビットに "1" が設定されます。 リセット要因ビットは、ウォッチドッグタイマ制御レジスタ (WDTC) をリードした後、"0" にクリアされます。 パワーオン時、PONR ビット以外のビットの値は保証されません。PONR ビットが "1" の場合、この PONR ビット以外のビットの値は無視してください。
bit6	未定義	<p>読み込んだ場合: 値は不定です。</p> <p>書き込んだ場合: 動作に影響はありません。</p>
bit2	WTE: ウォッチドッグタイマ 制御ビット	<p>"0" を設定した場合: ウォッチドッグタイマが起動する (リセット後の最初の書込み) か、または 2 ビットカウンタがクリアされます。 (リセット後の 2 番目以降の書込み)</p> <p>"1" を設定した場合: 動作に影響しません。</p>
bit1, bit0	WT1, WT0: インターバル時間選択 ビット	<ul style="list-style-type: none"> ウォッチドッグタイマのインターバル時間を選択します。 ウォッチドッグタイマの起動時のデータのみが有効です。ウォッチドッグタイマ起動後の設定は無視されます。 これらのビットは、ライトオンリです。

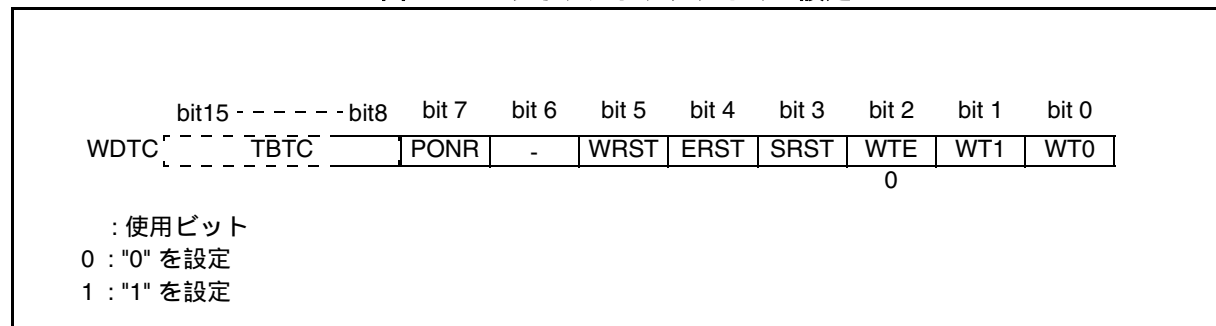
11.4 ウォッチドッグタイマの動作説明

ウォッチドッグタイマは、ウォッチドッグカウンタのオーバフローによりウォッチドッグリセットを生成します。

■ ウォッチドッグタイマの動作

ウォッチドッグタイマを動作させるには、図 11.4-1 の設定が必要です。

図 11.4-1 ウォッチドッグタイマの設定



● ウォッチドッグタイマの起動

- ウォッチドッグタイマは、ウォッチドッグタイマ制御レジスタ (WDTC) の WTE ビットにリセット後の最初の "0" 書込みを行うと起動されます。インターバル時間を設定するには、ウォッチドッグタイマ制御レジスタの WT1 ビットと WT0 ビットを同時に設定してください。
- ウォッチドッグタイマがいったん起動すると、パワーオンリセットまたはウォッチドッグリセットでしかウォッチドッグタイマを停止できません。

● ウォッチドッグタイマのクリア

- 2 番目以降の "0" 書込みを WTE ビットに行うと、ウォッチドッグタイマの 2 ビットカウンタはクリアされます。このカウンタが指定インターバル時間内にクリアされない場合は、オーバフローが発生し、ウォッチドッグリセットが発生します。
- ウォッチドッグカウンタは、リセットの生成、スリープモードへの遷移、ストップモードへの遷移、またはクロックモードへの遷移によりリセットされます。

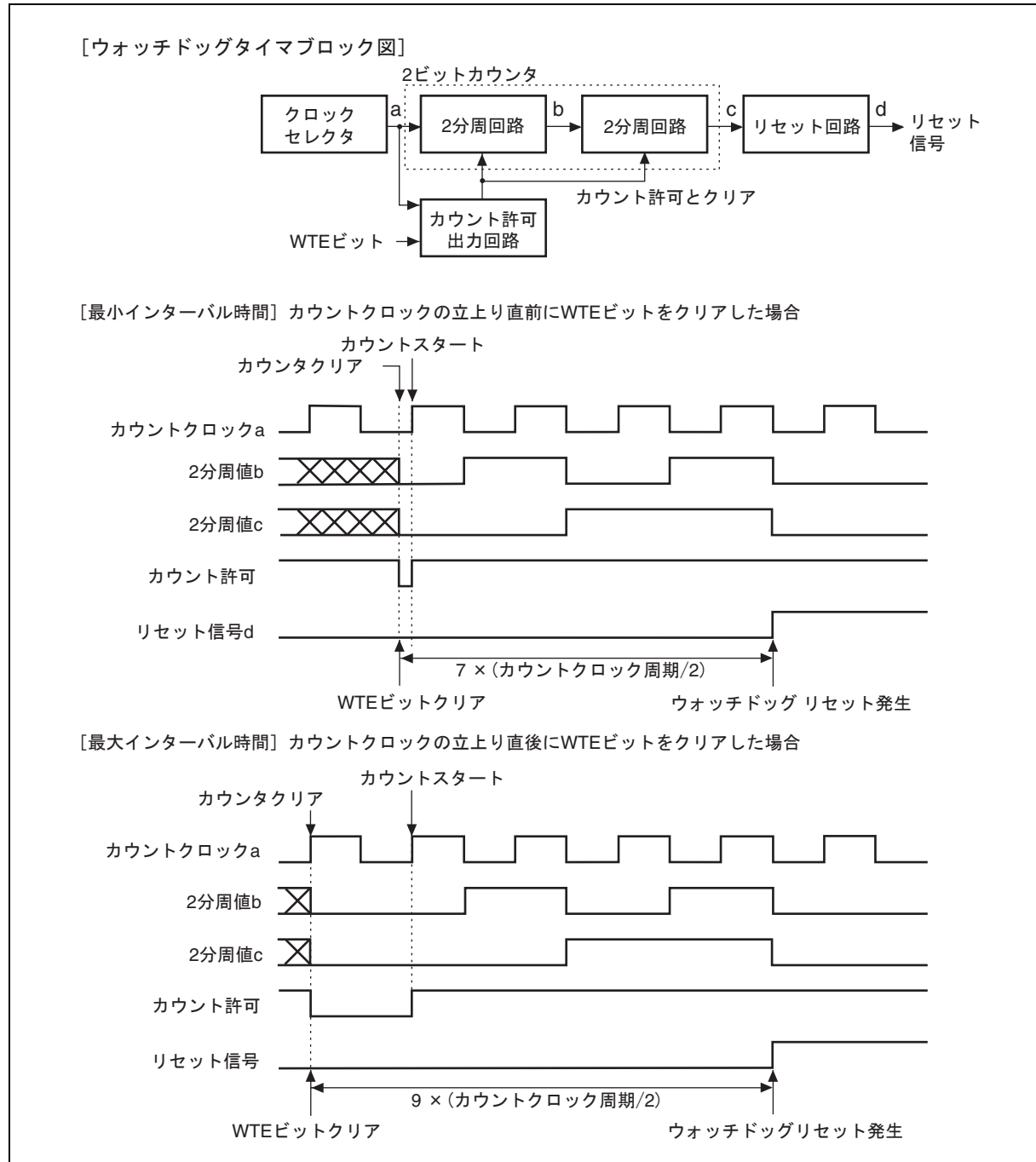
● ウォッチドッグタイマのインターバル

図 11.4-2 にウォッチドッグタイマのクリアタイミングとインターバル時間の関係を示します。ウォッチドッグタイマのインターバル時間は、ウォッチドッグタイマのクリアタイミングに従って変化し、またカウントクロックサイクルの 3.5 倍 ~ 4.5 倍の長さを必要とします。

● リセット要因の確認

リセット要因は、リセット後のウォッチドッグタイマ制御レジスタ (WDTC) の PONR, WRST, ERST, SRST ビットを確認することで判断できます。

図 11.4-2 ウォッチドッグタイマのクリアタイミングとインターバル時間



11.5 ウォッチドッグタイマ使用上の注意

ウォッチドッグタイマを使用する場合は、以下の点にご注意ください。

■ ウォッチドッグタイマ使用上の注意

- ウォッチドッグタイマの停止について

ウォッチドッグタイマはいったん起動されると、パワーオンリセットまたはウォッチドッグリセットが発生するまで停止できません。ウォッチドッグタイマカウンタは、外部リセットまたはソフトリセットでクリアされます。ただし、カウンタがクリアされてもウォッチドッグタイマは停止状態にはなりません。

- インターバル時間について

タイムベースタイマの桁上り信号をインターバルカウントクロックとして使用するので、ウォッチドッグタイマのインターバル時間は、タイムベースタイマがクリアされると、設定された時間より長くなる場合があります。

- インターバル時間の選択について

インターバルは、ウォッチドッグタイマが起動されたとき選択できます。起動以外の動作時に書き込まれたデータは無視されます。

- プログラム作成上の注意

メインループ内でウォッチドッグタイマを繰り返しクリアするプログラムを作成する場合、メインループの処理時間（割込み処理を含む）は、ウォッチドッグタイマの最小インターバル時間に等しいかまたはそれ未満でなければなりません。

- タイムベースタイマモード中のウォッチドッグタイマ動作

タイムベースタイマは、タイムベースタイマモードが設定されている間動作します。ただし、ウォッチドッグタイマは一時的に停止状態になります。

第12章

16 ビットリロードタイマ

16 ビットリロードタイマの機能と動作について説明します。

- 12.1 16 ビットリロードタイマの概要
- 12.2 16 ビットリロードタイマのブロックダイアグラム
- 12.3 16 ビットリロードタイマの端子
- 12.4 16 ビットリロードタイマのレジスタ
- 12.5 16 ビットリロードタイマの割込み
- 12.6 16 ビットリロードタイマの動作説明
- 12.7 16 ビットリロードタイマの使用上の注意

12.1 16 ビットリロードタイマの概要

16 ビットリロードタイマにおいては、以下の 2 つのクロックモードと 2 つのカウンタ動作モードを選択できます。

- 内部クロックモード：3 種類の内部クロックから選択した 1 つのクロックに同期してカウントダウン
- イベントカウントモード：外部端子におけるパルスエッジの検出によってカウントダウン

タイマは、カウント値が "0000_H" から "FFFF_H" へ変更されたときにアンダフロー条件結果を定義します。すなわち、アンダフローは [リロードレジスタ設定値 + 1] の後、発生することになります。

- リロードモード：カウント設定値を再リロードしてカウントを繰り返す
- ワンショットモード：アンダフローによってカウントを停止

カウンタアンダフローは、割込みを生成することもでき、拡張インテリジェント I/O サービス (EI²OS) に対応しています。

■ 16 ビットリロードタイマの動作モード

表 12.1-1 に 16 ビットリロードタイマの動作モードを示します。

表 12.1-1 16 ビットリロードタイマの動作モード

クロックモード	カウンタ動作モード	動作モード
内部クロックモード	リロードモード	ソフトウェアトリガ動作 外部トリガ入力動作 外部ゲート入力動作
	ワンショットモード	
イベントカウントモード (外部クロックモード)	リロードモード	ソフトウェアトリガ動作
	ワンショットモード	

■ 内部クロックモード

以下の動作を行う場合、3 種類の内部クロックモードから 1 種類のカウンタクロックを選択できます。

● ソフトウェアトリガ動作

タイマ制御ステータスレジスタ (TMCSR0/TMCSR1) の TRG ビットに "1" が設定してカウントを開始させます。TRG ビットによるトリガ入力、外部トリガ入力および外部ゲート入力に対し有効です。

● 外部トリガ動作

選択されたエッジ (立上り、立下り、両エッジ) が TIN0/TIN1 端子へ入力されると、カウントを開始します。

● 外部ゲート入力動作

選択された信号レベル ("L" または "H") が TIN0/TIN1 端子へ入力されると、カウントを継続します。

■ イベントカウントモード (外部クロックモード)

選択された有効エッジ (立上り, 立下り, 両エッジ) が TIN0/TIN1 端子へ入力されると, カウントダウンを開始します。一定周期の外部クロックを使用する場合, インターバルタイマとしても使用できます。

■ カウンタ動作

● リロードモード

カウントダウンでアンダフロー ("0000_H" "FFFF_H") が発生すると, カウント設定値がリロードされカウント動作を継続します。また, アンダフローにより発生した割込み要求は, インターバルタイマとして使用することも可能です。

また, アンダフローごとに反転するトグル波形は, TO0/TO1 端子から出力されます。16 ビットリロードタイマ用インターバル時間を表 12.1-2 に示します。

表 12.1-2 16 ビットリロードタイマのインターバル時間

カウントクロック	カウントクロック周期	インターバル時間
内部カウントクロック	$2^1/\phi$ (0.125 μ s)	0.125 μ s ~ 8.192 ms
	$2^3/\phi$ (0.5 μ s)	0.5 μ s ~ 32.768 ms
	$2^5/\phi$ (2.0 μ s)	2.0 μ s ~ 131.1 ms
外部カウントクロック	$2^3/\phi$ (0.5 μ s) 以上	0.5 μ s 以上

ϕ : マシクロック。()内はマシクロックが16MHzでFSELビットが"1"の場合のクロックインターバル時間です。

表 12.1-3 16 ビットリロードタイマのインターバル時間

カウントクロック	カウントクロック周期	インターバル時間
内部カウントクロック	$2^1/\phi$ (0.167 μ s)	0.167 μ s ~ 10.923 ms
	$2^3/\phi$ (0.667 μ s)	0.667 μ s ~ 43.690 ms
	$2^5/\phi$ (2.667 μ s)	2.667 μ s ~ 174.760 ms
外部カウントクロック	$2^3/\phi$ (0.667 μ s) 以上	0.667 μ s 以上

ϕ : マシクロック。()内はマシクロックが24MHzでFSELビットが"0"の場合のクロックインターバル時間です。

● ワンショットモード

カウントダウンがアンダフロー ("0000_H" "FFFF_H") が発生した場合, カウント動作は停止し, アンダフロー発生のために割込みが発生します。カウンタ動作中, カウンタが進行中であることを示す矩形波を TO0/TO1 端子から出力できます。

< 参考 >

- 16 ビットリロードタイマは, UART のボーレート生成に使用されます。
- 16 ビットリロードタイマは, A/D コンバータの起動トリガに使用することもできます。

12.2 16ビットリロードタイマのブロックダイアグラム

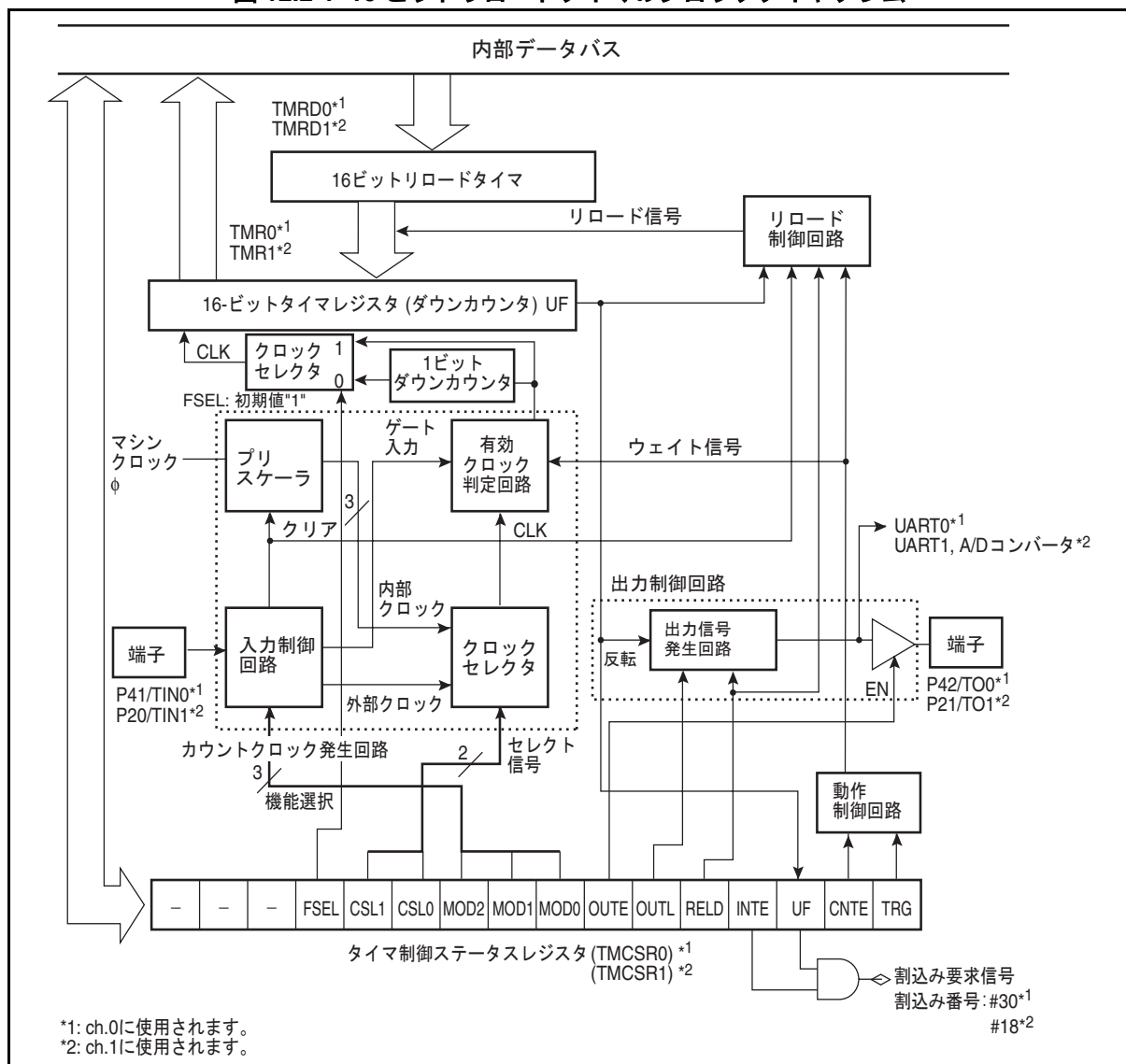
16 ビットリロードタイマは、以下の7種類のブロックで構成されています。

- カウントクロック生成回路
- リロード制御回路
- 出力制御回路
- 動作制御回路
- 16 ビットタイマレジスタ (TMRH0/TMRH1)
- 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1)
- タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1, TMCSRH0/TMCSRH1)

■ 16 ビットリロードタイマのブロックダイヤグラム

図 12.2-1 に 16 ビットリロードタイマのブロックダイヤグラムを示します。

図 12.2-1 16 ビットリロードタイマのブロックダイヤグラム



- カウントクロック生成回路

マシントクロックまたは外部入力クロックから 16 ビットリロードタイマ用のカウントクロックを生成します。

- リロード制御回路

タイマ起動とアンダフロー発生時にリロード動作を制御します。

- 出力制御回路

16 ビットリロードタイマのアンダフローによる TO0/TO1 端子出力の反転制御と、TO0/TO1 端子出力の許可 / 禁止を制御します。

- 動作制御回路

16 ビットリロードタイマを起動または停止します。

- 16 ビットタイマレジスタ (TMRL0/TMRL1, TMRH0/TMRH1)

これらのレジスタは、16 ビットダウンカウンタの現カウンタ値を読み出すのに使用されます。

- 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1)

16 ビットリロードタイマのインターバル時間を設定するレジスタです。このレジスタの設定値は 16 ビットタイマレジスタにロードされ、ダウンカウントします。

- タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1, TMCSRH0/TMCSRH1)

このレジスタは、16 ビットリロードタイマのカウントクロックと動作モードの選択、動作条件の設定、ソフトウェアによる起動、カウントの許可 / 禁止、リロードまたはワンショットモードおよび端子出力レベルの選択、タイマ出力の許可 / 禁止、クロック分割と割込みの制御、状態の確認を行います。

12.3 16 ビットリロードタイマの端子

16 ビットリロードタイマの端子について説明します。

■ 16 ビットリロードタイマの端子

16 ビットリロードタイマの端子は、汎用入出力ポートと兼用になっています。

表 12.3-1 に 16 ビットリロードタイマとして使用する場合の端子機能、入出力形式および必要な設定を示します。

表 12.3-1 16 ビットリロードタイマの端子

端子名	端子機能	入出力形式	プルアップ オプション	スタンバイ 制御	端子設定
P41/TIN0	ポート 4 の I/O とタイマ入力	CMOS 出力 / CMOS ヒステ リシス入力	なし	あり	入力ポートに設定 (DDR4: bit1 = 0)
P42/TO0	ポート 4 の I/O とタイマ出力				タイマ出力許可に設定 (TMCSRL0 : OUTE = 1) サウンドジェネレータ 出力禁止
P20/TIN1	ポート 2 の I/O とタイマ入力		選択可		入力ポートに設定 (DDR2: bit0 = 0) PPG1 出力禁止
P21/TO1	ポート 2 の I/O とタイマ出力				タイマ出力許可に設定 (TMCSRL1 : OUTE = 1) PPG4 出力禁止

端子ブロックダイアグラムについては、「第 9 章 I/O ポート」をご参照ください。

12.4 16 ビットリロードタイマのレジスタ

16 ビットリロードタイマのレジスタ一覧を示します。

■ 16 ビットリロードタイマのレジスタ

図 12.4-1 に 16 ビットリロードタイマのレジスタ一覧を示します。

図 12.4-1 16 ビットリロードタイマのレジスタ

		アドレス	bit15	bit8	bit7	bit0
16ビットリロードタイマ 0	{	000082 _H	-	TMCSR0 (タイマ制御ステータスレジスタ)		
		000084 _H	-	TMR0/TMRD0 (16ビットタイマレジスタ/16ビットリロードレジスタ)*		
16ビットリロードタイマ 1	{	000086 _H	-	TMCSR1 (タイマ制御ステータスレジスタ)		
		000088 _H	-	TMR1/TMRD1 (16ビットタイマレジスタ/16ビットリロードレジスタ)*		

*: リード時は16ビットタイマレジスタ (TMR)として、ライト時は16ビットリロードレジスタ (TMRLR)として機能します。

12.4.1 タイマ制御ステータスレジスタ，上位 (TMCSRH0/TMCSRH1)

タイマ制御ステータスレジスタ (TMCSRH0/TMCSRH1) の上位 bit12 ~ bit8 と下位 bit7 は 16 ビットリロードタイマの動作モードの選択，動作条件の設定をします。下位 bit7(MOD0 ビット) の使用も説明します。

■ タイマ制御ステータスレジスタ上位ビットおよび bit7(TMCSRH0/TMCSRH1)

図 12.4-2 タイマ制御ステータスレジスタ上位 (TMCSRH0/TMCSRH1) の上位ビットおよび bit7

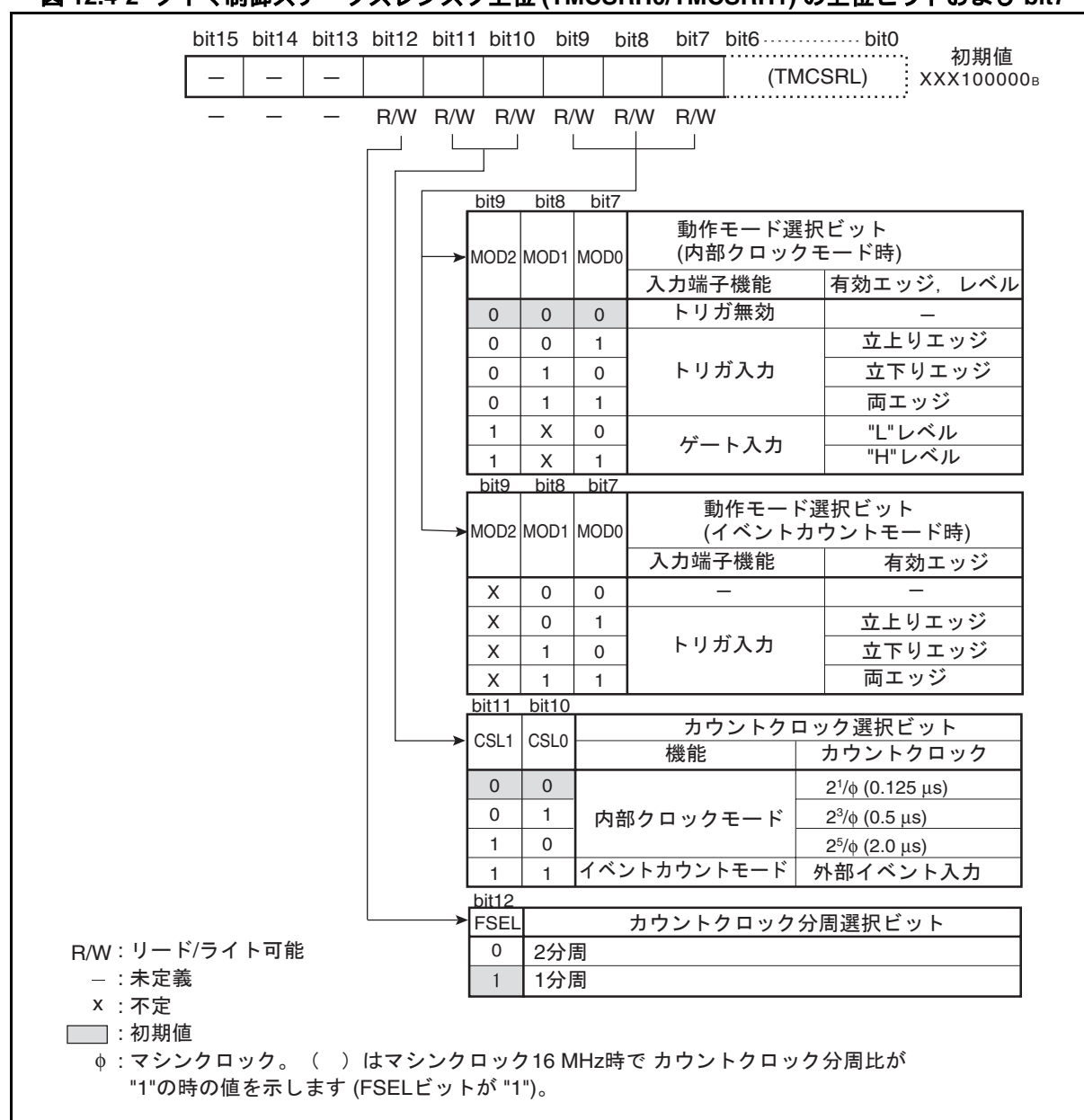


表 12.4-1 タイマ制御ステータスレジスタの上位ビットおよび bit7 の機能 (TMCSRH0/TMCSRH1)

ビット名		機能
bit15, bit14, bit13	未定義ビット	読み出した場合：値は不定です。 書き込んだ場合：値は常に "1" となります。
bit12	FSEL: カウントクロック分周選択 ビット	<ul style="list-style-type: none"> • カウントクロック分周比を設定します。 • FSEL ビットが "0" に設定された場合、カウントクロック選択ビット (CSL1 および CSL0) で指定されたカウントクロックは、2 分周となります。
bit11, bit10	CSL1, CSL0: カウントクロック選択ビット	<ul style="list-style-type: none"> • 16 ビットリロードタイマのカウントクロックを選択します。 "11_B" 以外に設定した場合：内部クロックモードが選択され、内部クロックをカウントします。 "11_B" に設定した場合：イベントカウントモードが選択され、外部クロックのエッジをカウントします。
bit9, bit8, bit7	MOD2, MOD1, MOD0: 動作モード選択ビット	<p><内部クロックモード> MOD2 ビットで入力端子の機能を選択します。 MOD2 ビットが "0" の場合：</p> <ul style="list-style-type: none"> • 入力端子は、トリガ入力端子として機能します。 • 有効エッジが入力されると、16 ビットリロードレジスタの内容をカウンタへロードし、カウント動作を継続します。MOD1/MOD0 ビットで有効エッジの種類を選択します。 <p>MOD2 ビットが "1" の場合：</p> <ul style="list-style-type: none"> • 入力端子はゲート入力となり、有効レベル信号が入力されている間のみカウントします。MOD0 ビットによって、有効レベルを選択します。 • MOD1 ビットの値は動作に影響しないので、任意の値 ("0" か "1") を設定してください。 <p><イベントカウントモード></p> <ul style="list-style-type: none"> • MOD2 ビットの値は動作に影響しないので、任意の値 ("0" か "1") を設定してください。 • 入力端子はイベント入力用のトリガ入力端子となり、有効エッジは MOD1/MOD0 ビットで設定されます。 <p>(注意事項) 動作モード選択は、カウンタ動作停止モードにしてください。 (TMCSRL0/TMCSRL1: CNTE = 0)</p>

12.4.2 タイマ制御ステータスレジスタ , 下位 (TMCSRL0/TMCSRL1)

タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1) の下位 7 ビット (下位ビットの一部) は, 16 ビットリロードタイマの動作条件の設定, カウント動作の許可 / 禁止の設定, 割り込み制御, および動作の状態を確認する機能があります。

■ タイマ制御ステータスレジスタ下位 (TMCSRL0/TMCSRL1)

図 12.4-3 タイマ制御ステータスレジスタ下位 (TMCSRL0/TMCSRL1)

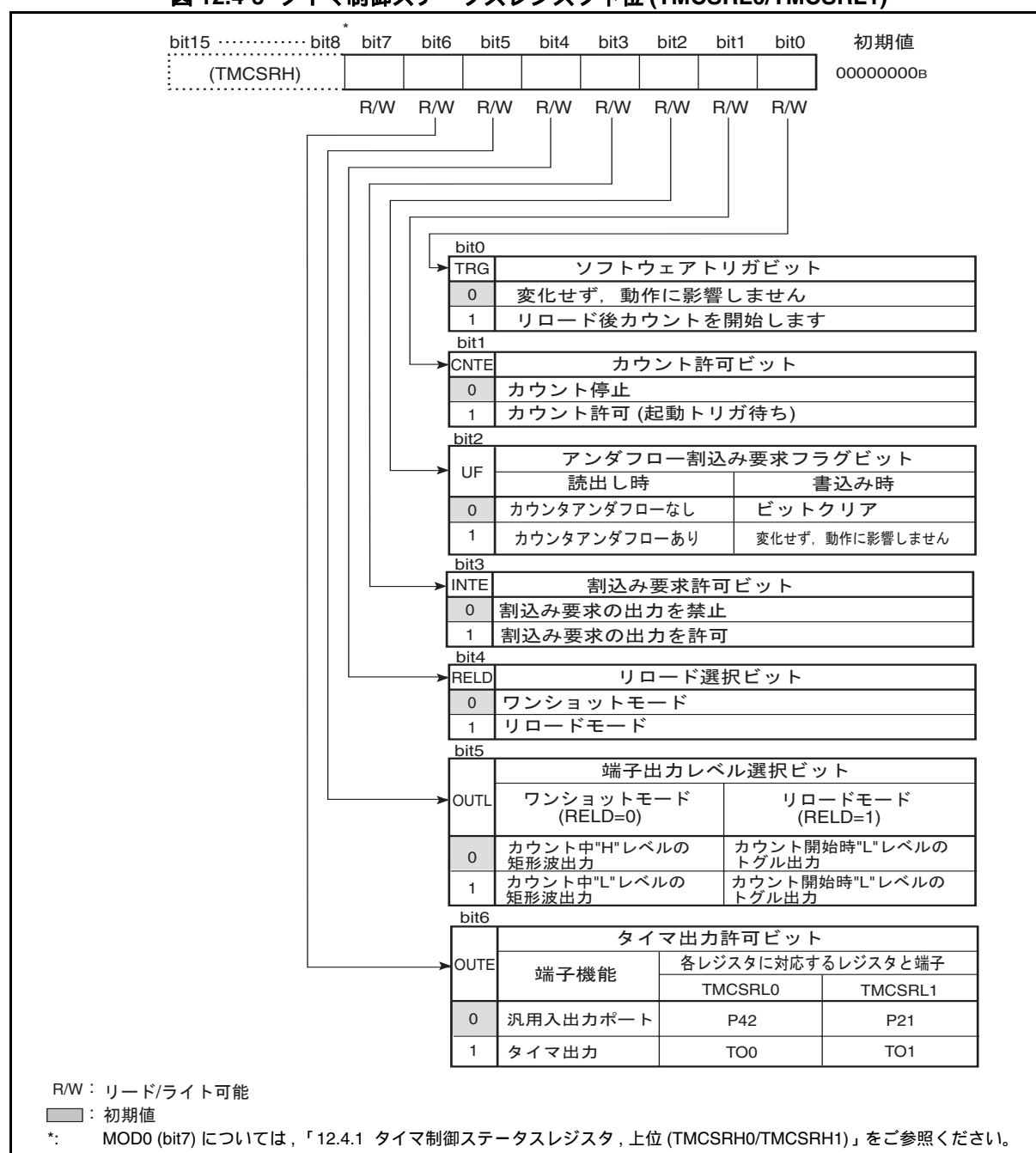


表 12.4-2 タイマ制御ステータスレジスタの下位ビットの機能 (TMCSRL0/TMCSRL1)

ビット名		機能
bit6	OUTE: タイマ出力許可ビット	<ul style="list-style-type: none"> タイマ出力端子への出力を許可または禁止します。 "0" に設定した場合：汎用入出力ポートとして機能します。 "1" に設定した場合：タイマ出力端子として機能します。 タイマ出力端子の出力波形は、リロードモードではトグル波形を出力し、ワンショットモードでは、カウント動作中を示す矩形波を出力します。
bit5	OUTL: 端子出力レベル選択ビット	<ul style="list-style-type: none"> タイマ出力端子への出力レベルを設定するビットです。 このビットに "0" を設定した場合と "1" を設定した場合では、端子出力レベルが逆になります。
bit4	RELD: リロード選択ビット	<ul style="list-style-type: none"> リロード動作を設定します。 "1" に設定した場合：リロードモードとなり、アンダフローの発生と同時に、16 ビットリロードレジスタに設定された値が 16 ビットカウンタにロードされ、カウント動作を継続します。 "0" に設定した場合：ワンショットモードになり、アンダフローが発生した場合は、カウント動作を停止します。
bit3	INTE: 割込み要求許可ビット	<ul style="list-style-type: none"> 割込み要求を許可または禁止できます。 このビットと割込み要求フラグビット (UF) が "1" にセットされると、タイマは割込み要求を出力します。
bit2	UF: アンダフロー割込み要求フラグビット	<ul style="list-style-type: none"> 16 ビットカウンタのアンダフローにより、"1" にセットされます。 "0" に設定した場合：クリアされます。 "1" に設定した場合：動作に影響しません。 このビットは、EI²OS の起動時もクリアされます。
bit1	CNTE: カウント許可ビット	<ul style="list-style-type: none"> カウント動作を許可または禁止できます。 "1" に設定した場合：起動トリガ待ち状態になります。トリガの発生により、実カウントを開始します。
bit0	TRG: ソフトウェアトリガビット	<ul style="list-style-type: none"> このビットは、インターバルタイマ機能またはカウンタ機能をソフトウェアで起動します。 "0" に設定した場合：動作に影響しません。 "1" に設定した場合：ソフトウェアトリガとなり、16 ビットリロードレジスタに設定した内容がカウンタにリロードされ、カウント動作を開始します。 CNTE ビットに "1" が設定された場合、動作モードに関わりなくトリガ入力に常に許可されます。 リードした場合：常に "0" が読み出されます。

12.4.3 16 ビットタイマレジスタ (TMR0/TMR1)

16 ビットタイマレジスタ (TMR0/TMR1) は、16 ビットダウンカウンタのカウント値を読み出すことができます。

■ 16 ビットタイマレジスタ (TMR0/TMR1)

図 12.4-4 に 16 ビットタイマレジスタ (TMR0/TMR1) のビット構成を示します。

図 12.4-4 16 ビットタイマレジスタ (TMR0/TMR1)

TMR0/TMR1	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	初期値
	D15	D14	D13	D12	D11	D10	D9	D8	XXXXXXXX _B
	R	R	R	R	R	R	R	R	
TMR0/TMR1	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値
	D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX _B
	R	R	R	R	R	R	R	R	

R: リードオンリ
X: 不定

16 ビットタイマレジスタは 16 ビットダウンカウンタのカウント値を読み出すことができます。カウント開始にあたりカウントが許可 (TMCSR0/TMCSR1: CNTE = 1) されている場合、16 ビットリロードレジスタに書き込まれた値はこのレジスタへリロードされ、カウントダウンを開始します。このレジスタ値はカウンタ停止状態 (TMCSR0/TMCSR1: CNTE = 0) のときに格納されます。

< 注意事項 >

- このレジスタはカウント中でも値を読み出すことが可能です。読出し時は必ずワード転送命令 (MOVW A, 003AH など) をご使用ください。
- 16 ビットタイマレジスタ (TMR0/TMR1) は読出し専用のレジスタであり、書き込み専用の 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) の同一アドレスに配置されています。したがって、このレジスタへの書き込みは TMR レジスタの値へ影響しませんが、TMRDL0/TMRDL1 と TMRDH0/TMRDH1 レジスタへの書き込みは実行されます。
- TMR0/TMR1 レジスタへはワードアクセスで必ず行ってください。

12.4.4 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1)

16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) は、16 ビットダウンカウンタへのリロード値を設定するレジスタです。16 ビットリロードレジスタに設定された値は、16 ビットダウンカウンタにリロードされ、ダウンカウントします。

■ 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1)

図 12.4-5 に 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) のビット構成を示します。

図 12.4-5 16 ビットリロードレジスタのビット構成 (TMRDL0/TMRDL1, TMRDH0/TMRDH1)

TMRDH0/TMRDH1	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	初期値
	D15	D14	D13	D12	D11	D10	D9	D8	XXXXXXXX _B
	W	W	W	W	W	W	W	W	--
TMRDL0/TMRDL1	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値
	D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX _B
	W	W	W	W	W	W	W	W	--

W: ライトオンリ
X: 不定

- 16 ビットリロードタイマのモードに関係なく、カウンタ動作が禁止 (TMCSR0/TMCSR1: CNTE=0) のときは、カウンタの初期値がこのレジスタに設定されます。カウントが許可 (TMCSR0/TMCSR1: CNTE=1) でカウントを開始した場合、レジスタに書き込まれた値からカウントダウンを開始します。
- アンダフローが発生した場合に、リロードモード時に 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) に設定された値はカウンタへリロードされ、その後カウントダウンは続きます。ワンショットモード時は、アンダフローが発生するとカウンタは "FFFF_H" で停止します。
- 16 ビットリロードレジスタへ値を設定する場合は、カウンタ動作を停止 (TMCSR0/TMCSR1: CNTE=0) してください。また、必ずワード転送命令 (MOVW 003AH, A) を使用して書き込んでください。
- 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) は、機能的に書き込み専用のレジスタで、読み込み専用の 16 ビットタイマレジスタ (TMR0/TMR1) と同一アドレスに配置されています。したがって、読出し値は TMR0/TMR1 の値となるため、INC/DEC 命令などリードモディファイライト (RMW) 動作をする命令は使用できません。

12.5 16 ビットリロードタイマの割込み

16 ビットリロードタイマは、16 ビットダウンカウンタのアンダフロー時に割込み要求を出力します。拡張インテリジェント I/O サービス (EI²OS) に対応しています。

■ 16 ビットリロードタイマの割込み

表 12.5-1 に 16 ビットリロードタイマの割込み制御ビットと割込み要因を示します。

表 12.5-1 16 ビットリロードタイマの割込み制御ビットと割込み要因

割込み要因	タイマ制御ステータスレジスタの下位ビット (TMCSRL0/TMCSRL1)		
	割込みフラグビット	割込み許可ビット	割込みフラグのクリア
16 ビットダウンカウンタのアンダフロー (TMR0/TMR1) (0000 _H → FFFF _H)	UF	INTE	<ul style="list-style-type: none"> UF ビットに "0" を書き込み リセット EI²OS を開始

表 12.5-1 にある割込み要因が生成された場合、16 ビットリロードタイマの割込みフラグビットが "1" に設定されます。割込みフラグビットが "1" に設定されていて、16 ビットリロードタイマの割込み許可ビットが "1" の場合、割込み要求を割込みコントローラへ出力します。

■ 16 ビットリロードタイマの割込みと EI²OS

表 12.5-2 に 16 ビットリロードタイマの割込みと EI²OS の関係を示します。

表 12.5-2 16 ビットリロードタイマの割込みと EI²OS

チャンネル	割込み番号	割込み制御レジスタ		ベクタテーブルのアドレス			EI ² OS
		レジスタ名	アドレス	下位	上位	バンク	
16 ビットリロードタイマ 0	#30 (1E _H)	ICR09	0000B9 _H	FFFF84 _H	FFFF85 _H	FFFF86 _H	*
16 ビットリロードタイマ 1	#18 (12 _H)	ICR03	0000B3 _H	FFFFB4 _H	FFFFB5 _H	FFFFB6 _H	*

*: ICR03, ICR09 または割込みベクタと共用の割込み要因を使用しない場合は可能

■ 16 ビットリロードタイマの EI²OS 機能

16 ビットリロードタイマには EI²OS に対応する回路がありますので、アンダフロー発生によりカウンタが EI²OS を起動します。ただし EI²OS は、割込み制御レジスタ (ICR) を共用するリソース機能が割込みを使用していない場合のみ使用可能です。16 ビットリロードタイマ 0 で EI²OS を使用する場合には、波形ジェネレータの割込みは禁止しなければなりません。16 ビットリロードタイマ 1 で EI²OS を使用する場合には、出力コンペア 2 の割込みを禁止する必要があります。

12.6 16 ビットリロードタイマの動作説明

16 ビットリロードタイマの設定とカウンタの動作状態について説明します。

■ 16 ビットリロードタイマの設定

● 内部クロックモードの設定

インターバルタイマとして動作させるには、図 12.6-1 のように設定する必要があります。

図 12.6-1 内部クロックモードの設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMCSR	-	-	-	FSEL	CSL1	CSL0	MOD2	MOD1	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG
															1	
	"11" 以外															
TMRD	カウンタの初期値 (リロード値) を設定															

◎ : 使用ビット
1 : "1" を設定

● イベントカウントモードの設定

イベントカウンタとして動作させるには、図 12.6-2 のように設定する必要があります。

図 12.6-2 イベントカウントモードの設定

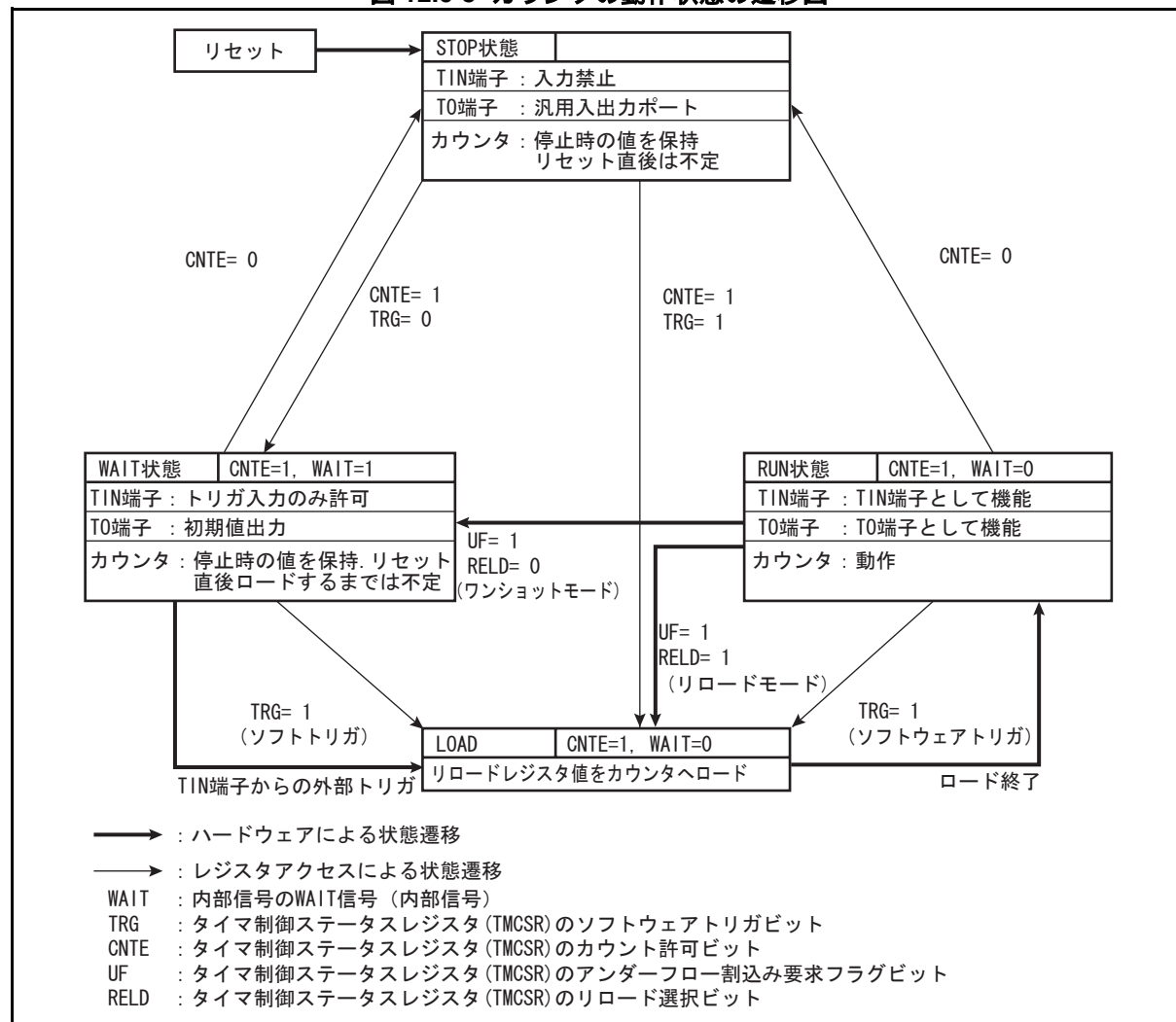
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TMCSR	-	-	-	FSEL	CSL1	CSL0	MOD2	MOD1	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG
					1	1									1	
TMRD	カウンタの初期値 (リロード値) を設定															
DDR5																
DDR0																

◎ : 使用ビット
1 : "1" を設定
○ : 入力端子を使用する場合 "0" を設定

■ カウンタの動作状態

カウンタの状態は、タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1, TMCSRH0/TMCSRH1) の CNTE ビットと内部 WAIT 信号で決まります。STOP 状態 (停止状態)、起動トリガ待ち状態 (WAIT 状態)、および動作状態 (RUN 状態) が設定可能です。図 12.6-3 にカウンタの状態遷移図を示します。

図 12.6-3 カウンタの動作状態の遷移図



12.6.1 内部クロックモード (リロードモード)

内部カウントクロックに同期して、16 ビットカウンタをダウンカウントし、カウンタのアンダフローのために、CPU への割込み要求を発生します。また、タイマ出力端子からトグル波形を出力できます。

■ 内部クロックモード (リロードモード) の動作

カウント動作が許可の場合 (TMCSR0/TMCSR1: CNTE = 1) で、ソフトウェアトリガビット (TMCSR: TRG) あるいは外部トリガによりタイマを開始すると、16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) に設定した値が 16 ビットダウンカウンタにリロードされ、カウント動作を開始します。カウント許可ビットとソフトウェアトリガビットが同時に "1" を設定された場合、カウント動作が許可されると同時にカウントを開始します。

16 ビットカウンタ値のアンダフロー ("0000_H" "FFFF_H") により、16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) に設定した値が 16 ビットカウンタにリロードされ、カウントを継続します。アンダフロー割込み要求フラグビット (UF) とアンダフロー割込み要求許可ビット (INTE) に "1" が設定された場合、割込み要求を出力します。

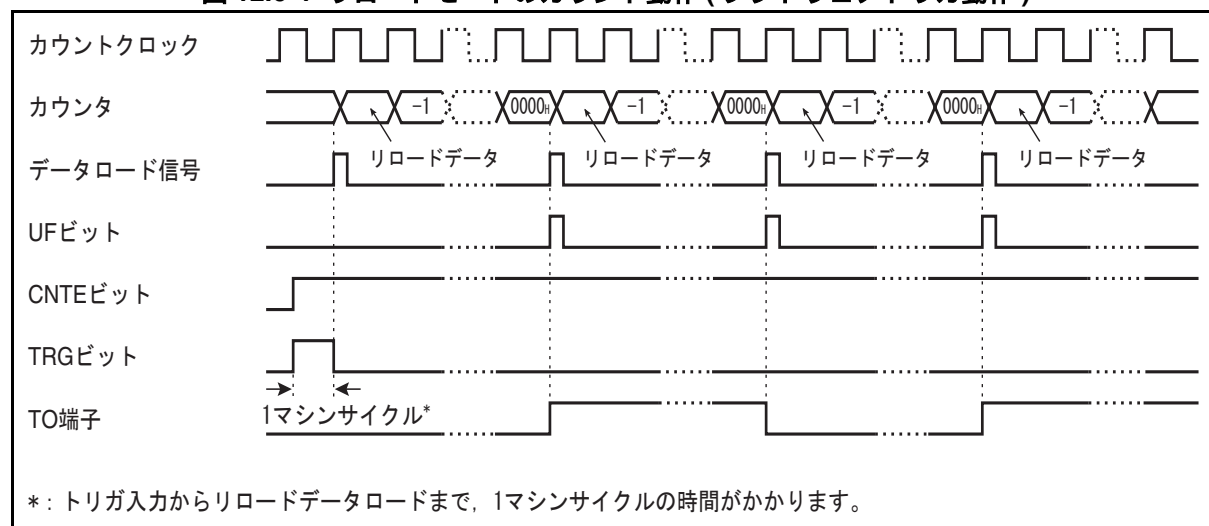
TO 端子からは、アンダフローごとに反転するトグル波形を出力できます。

● ソフトウェアトリガ動作

タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1, TMCSRH0/TMCSRH1) の TRG ビットが "1" に設定されている場合、カウントを開始します。

図 12.6-4 にリロードモード時のソフトウェアトリガの動作を示します。

図 12.6-4 リロードモードのカウント動作 (ソフトウェアトリガ動作)

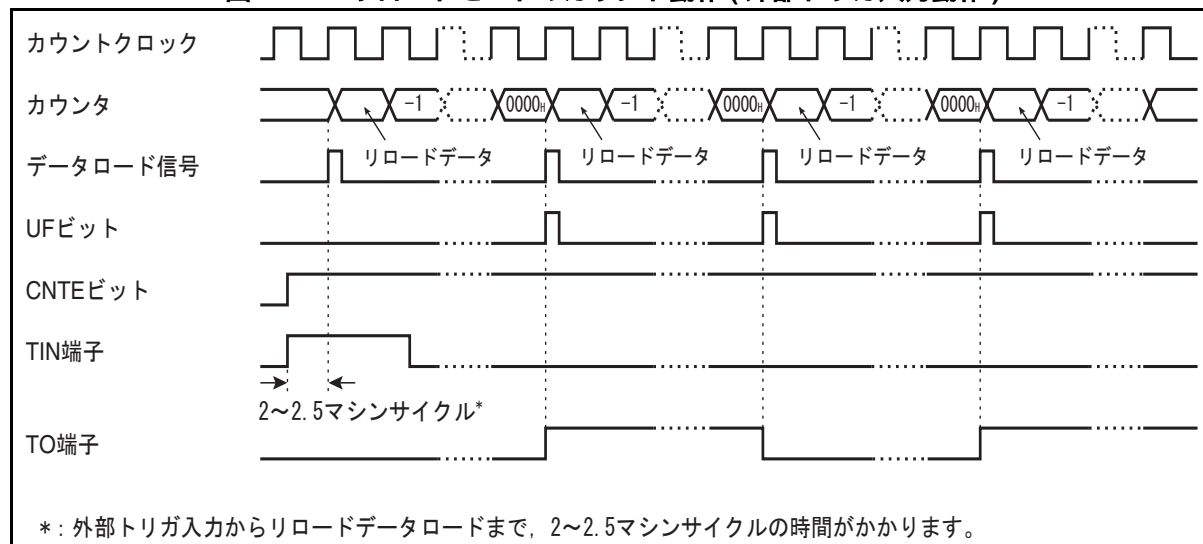


● 外部トリガ入力動作

有効エッジ（立上り、立下り、両エッジから選択可能）が TIN 端子に入力されると、カウントを開始します。

図 12.6-5 にリロードモード時における外部トリガ動作を示します。

図 12.6-5 リロードモードのカウント動作（外部トリガ入力動作）



< 注意事項 >

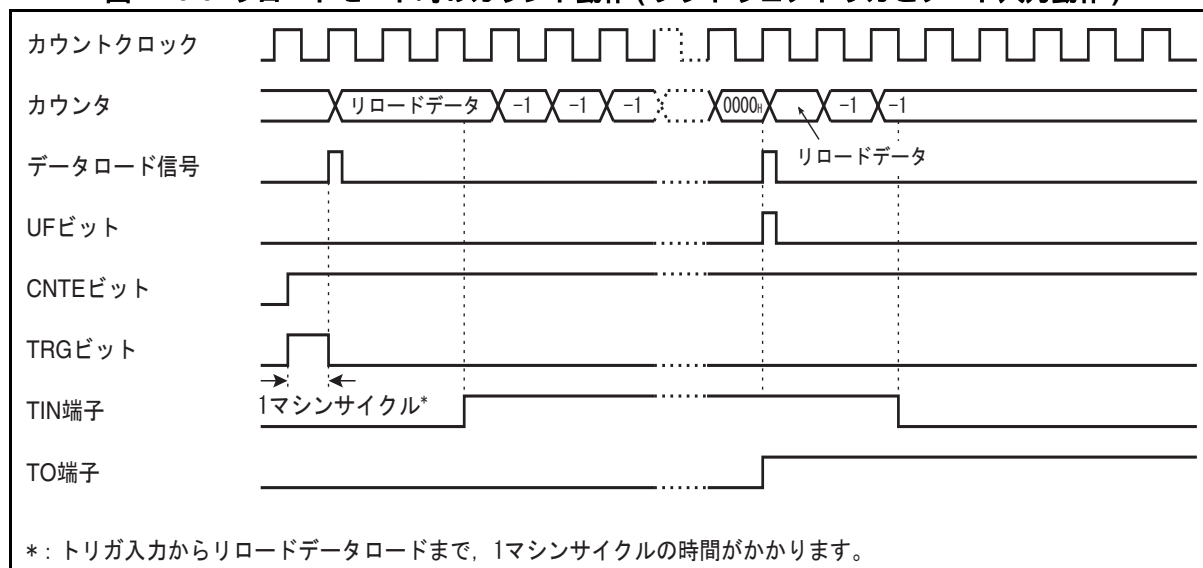
TIN 端子へ入力するトリガパルス幅は $2/\phi$ (ϕ : マシンクロック) 以上としてください。

● ゲート入力動作

有効レベル（"L" または "H" を選択可能）が TIN 端子に入力されると同時に、カウントを開始します。

図 12.6-6 にリロードモード時におけるゲート入力を示します。

図 12.6-6 リロードモード時のカウント動作 (ソフトウェアトリガとゲート入力動作)



< 注意事項 >

TIN 端子へ入力するトリガパルス幅は $2/\phi$ (ϕ : マシンクロック) 以上としてください。

12.6.2 内部クロックモード (ワンショットモード)

内部カウントクロックに同期して、16 ビットカウンタをカウントダウンし、カウンタのアンダフローが発生したときに CPU に対する割込み要求を出力します。また、TO0/TO1 端子から、カウント中を示す矩形波を出力します。

■ 内部クロックモード (ワンショットモード)

カウントが許可の場合 (TMCSR0/TMCSR1: CNTE = 1) に、ソフトウェアトリガビット (TMCSR0/TMCSR1: TRG) あるいは外部トリガによりタイマを開始すると、カウント動作を開始します。カウント許可ビットとソフトウェアトリガビットが "1" に設定されている場合、カウント動作を許可すると同時にカウントを開始します。

16 ビットカウンタ値のアンダフロー ("0000_H" "FFFF_H") により、16 ビットカウンタは "FFFF_H" の状態でカウントを停止し、アンダフロー割込みフラグビット (UF) は "1" に設定されます。割込み要求許可ビット (INTE) に "1" が設定されると、割込み要求を出力します。

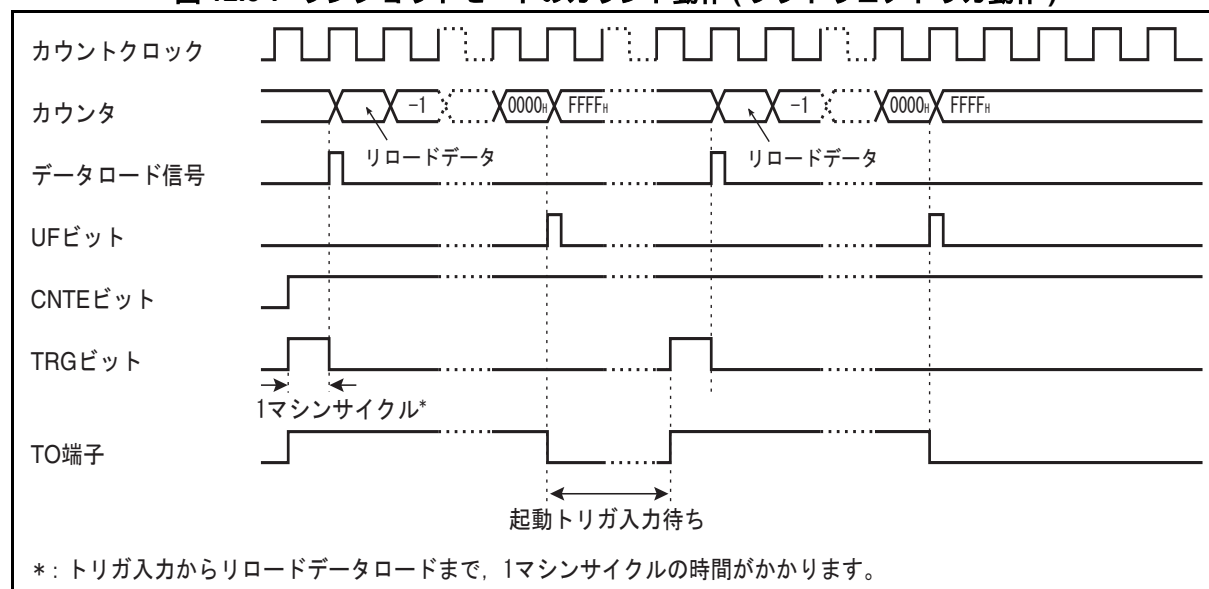
また、TO 端子からは、カウント中を示す矩形波を出力できます。

● ソフトウェアトリガ動作

タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1, TMCSRH0/TMCSRH1) の TRG ビットが "1" に設定されると同時にカウントを開始します。

図 12.6-7 にワンショットモード時におけるソフトウェアトリガ動作を示します。

図 12.6-7 ワンショットモードのカウント動作 (ソフトウェアトリガ動作)

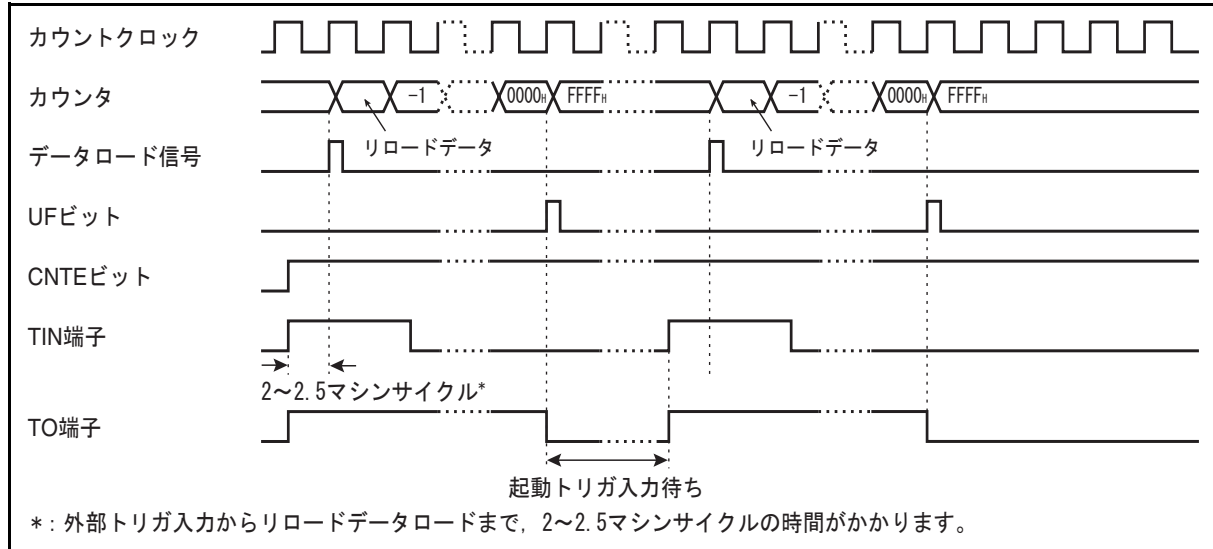


● 外部トリガ入力動作

有効エッジ (立上り , 立下り , 両エッジから選択可能) が TIN0/TIN1 端子に入力されると , カウントを開始します。

図 12.6-8 にワンショットモード時における外部トリガ動作を示します。

図 12.6-8 ワンショットモードのカウント動作 (外部トリガ動作)



< 注意事項 >

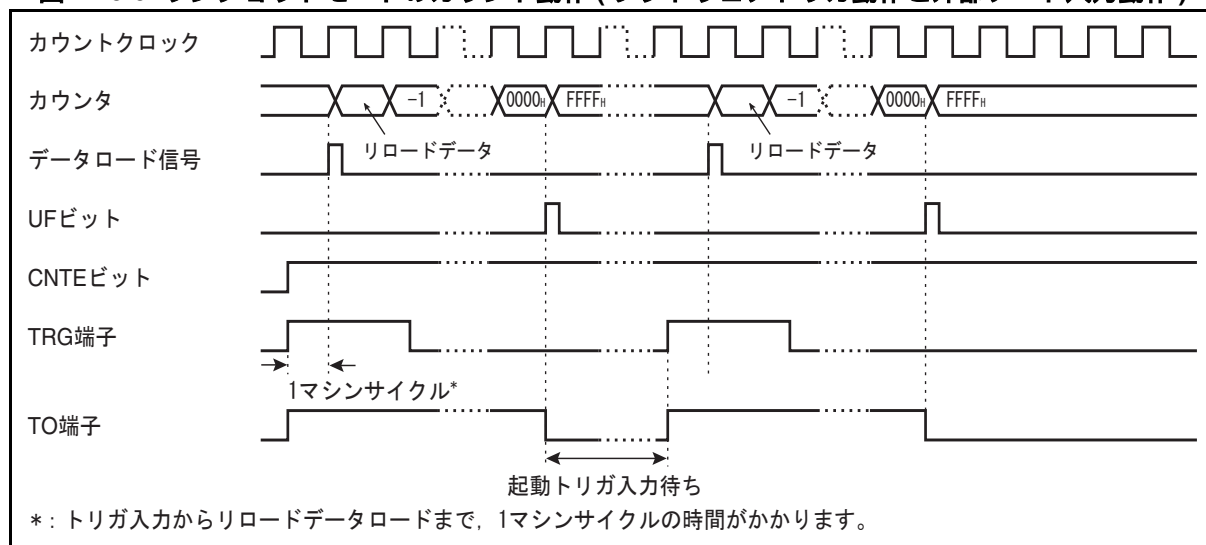
TIN 端子へ入力するトリガパルス幅は $2/\phi$ (ϕ : マシンクロック) 以上としてください。

● ゲート入力動作

有効レベル ("L" または "H" を選択可能) が TIN 端子に入力された場合 , カウントを開始します。

図 12.6-9 にワンショットモード時におけるゲート入力を示します。

図 12.6-9 ワンショットモードのカウンタ動作 (ソフトウェアトリガ動作と外部ゲート入力動作)



< 注意事項 >

TIN 端子へ入力するトリガパルス幅は $2/\phi$ (ϕ : マシンクロック) 以上としてください。

12.6.3 イベントカウントモード

TIN 端子からの入力エッジをカウントして、16 ビットカウンタをダウンカウントし、カウンタのアンダフローにより、CPU への割込み要求を出力します。また、TO0/TO1 端子からトグル波形または矩形波を出力できます。

■ イベントカウントモード

カウント動作を許可 (TMCSR0/TMCSR1: CNTE = 1) し、カウンタを起動 (TMCSR0/TMCSR1: TRG = 1) すると、16 ビット リロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) の値がカウンタにロードされます。TIN0/TIN1 端子に入力されたパルス (外部カウントクロック) の有効エッジ (立上り, 立下り, 両エッジ選択可) を検出することによりカウントダウンします。

カウント許可ビットとソフトウェアトリガビットを同時に "1" にセットすれば、カウント許可と同時にカウントを開始します。

● リロードモードの動作

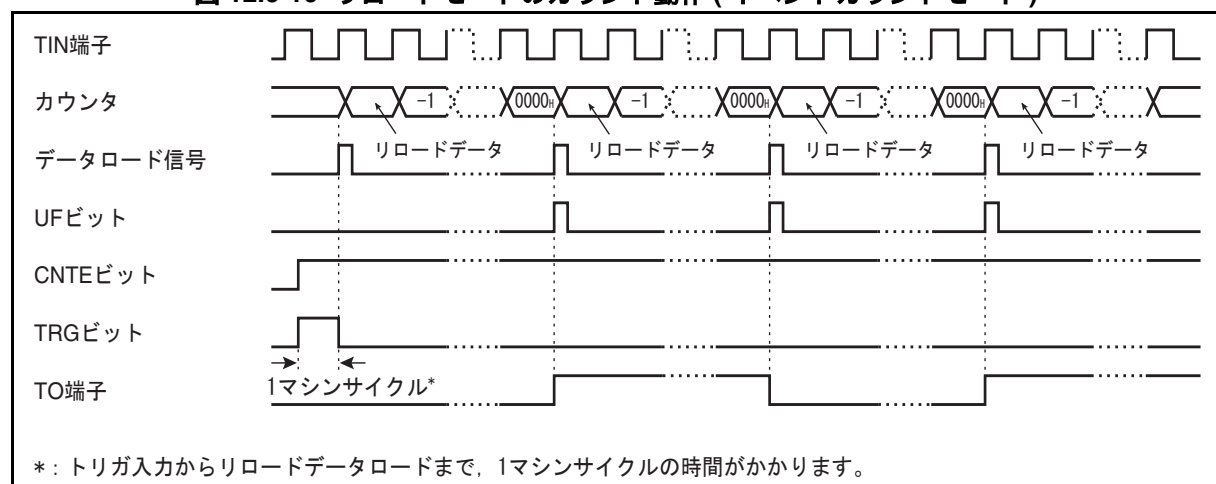
カウンタ値がアンダフロー ("0000_H" "FFFF_H") すると、リロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) の値をカウンタにロードして、カウント動作を続けます。このとき、アンダフロー割込みフラグビットとアンダフロー割込み許可ビット (TMCSR0/TMCSR1: INTE) が "1" であれば、割込み要求を発生します。

また、TO0/TO1 端子からは、アンダフローごとに反転するトグル波形を出力できます。
図 12.6-10 リロード時におけるカウント動作を示します。

< 注意事項 >

TIN 端子へ入力されるクロックの "H" 幅および "L" 幅は、 $4/\phi$ (ϕ : マシンクロック) 以上としてください。

図 12.6-10 リロードモードのカウント動作 (イベントカウントモード)



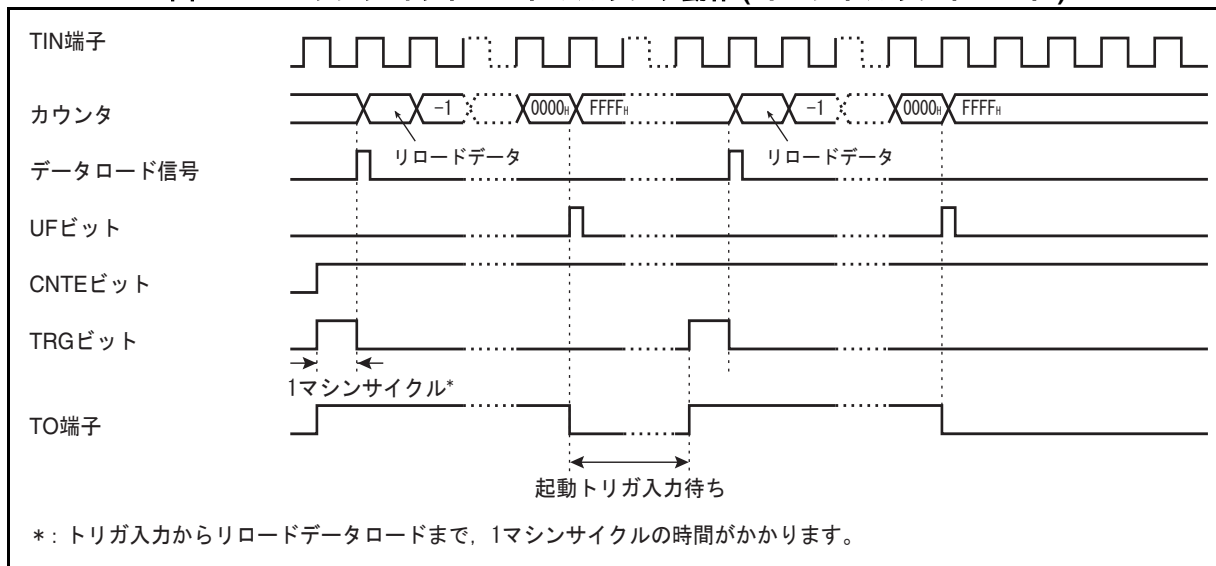
● ワンショットモードの動作

カウンタ値がアンダフロー("0000_H" "FFFF_H")時は, 16 ビットカウンタは "FFFF_H" の状態で停止します。このとき, アンダフロー割込みフラグビット (UF) を "1" にセットし, 割込み要求許可ビット (INTE) が "1" であれば, 割込み要求を発生します。

また, TO0/TO1 端子からは, カウント中を示す矩形波を出力できます。

図 12.6-11 にワンショットモード時におけるカウンタ動作を示します。

図 12.6-11 ワンショットモードのカウンタ動作 (イベントカウントモード)



< 注意事項 >

TIN 端子へ入力されるクロックの "H" 幅および "L" 幅は, $4/\phi$ (ϕ : マシンクロック) 以上としてください。

12.7 16 ビットリロードタイマの使用上の注意

16 ビットリロードタイマを使用上の注意を以下に示します。

■ 16 ビットリロードタイマの使用上の注意

● プログラム設定上の注意

- 16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1) に値を設定する場合は、カウントを停止 (TMCSR0/TMCSR1: CNTE = 0) させてください。カウンタ動作中でも 16 ビットタイマレジスタ (TMR0/TMR1) を読出しできますが、この場合は必ずワード転送命令 (MOVW A, dir など) を使用してください。
- タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1, TMCSRH0/TMCSRH1) の FSEL/CSL1/ CSL0/MOD2/MOD1/MOD0 ビットのデータを変更する場合は、カウントを停止 (TMCSRL0/TMCSRL1: CNTE = 0) してください。

● 割込みに関連する注意

- タイマ制御ステータスレジスタ (TMCSRL0/TMCSRL1, TMCSRH0/TMCSRH1) の UF ビットが "1" に設定され、割込み要求許可された状態 (TMCSRL0/TMCSRL1: INTE = 1) では、割込み処理から復帰できません。UF ビットは必ずクリアしてください。
- 16 ビットリロードタイマは、波形ジェネレータおよび出力コンペア 2 と割込みベクタを共用しているため、割込みを使用する場合には、割込み処理ルーチンで割込み要因をチェックする必要があります。
- 16 ビットリロードタイマが EI²OS を使用する場合、共用するリソースの割込みは禁止されなければなりません。

第13章

PWC タイマ

16 ビット PWC タイマの起動と動作について説明します。

- 13.1 PWC タイマの概要
- 13.2 PWC タイマのブロックダイアグラム
- 13.3 PWC タイマの端子
- 13.4 PWC タイマのレジスタ
- 13.5 PWC タイマの割込み
- 13.6 PWC タイマの動作
- 13.7 PWC タイマ使用上の注意

13.1 PWC タイマの概要

PWC(パルス幅測定)タイマはリロード機能を備えた多機能 16 ビットアップカウンタであり、入力信号のパルス幅を算出する機能も備えています。

PWC タイマは、16 ビットカウンタ、入力パルス分周器、分周比制御レジスタ、カウント入力端子、パルス出力端子、および 16 ビット制御レジスタから構成されています。

■ PWC タイマ

MB90820B シリーズは 2 つの PWC タイマチャンネルを装備しており、PWC タイマは以下の特性を備えています。

● タイマ機能

- 指定された時間間隔で割込み要求が生成されます。
- タイマ周期と同期したパルス信号を生成できます。
- 3 つの内部クロックからカウンタクロックを選択できます。

● パルス幅測定機能

- 外部パルス入力イベント間の時間を測定できます。
- 3 つの内部クロックからカウンタクロックを選択できます。
- カウントモード
 - "H" パルス幅 (立上りエッジから立下りエッジまでの幅)、"L" パルス幅 (立下りエッジから立上りエッジまでの幅)
 - 立上りエッジ周期 (立上りエッジから立上りエッジまでの周期)、立下りエッジ周期 (立下りエッジから立下りエッジまでの周期)
 - 中間エッジカウント (立上りまたは立下りエッジから立下りエッジまたは立上りエッジまでのカウント)
- 8 ビット入力分周器を使用して入力パルスを 2^2 , 2^4 , 2^6 または 2^8 で分周して周期を測定します。
- カウント完了時に割込み要求が生成されます。
- 単一カウントまたは連続カウントを選択できます。

■ PWC タイマの動作

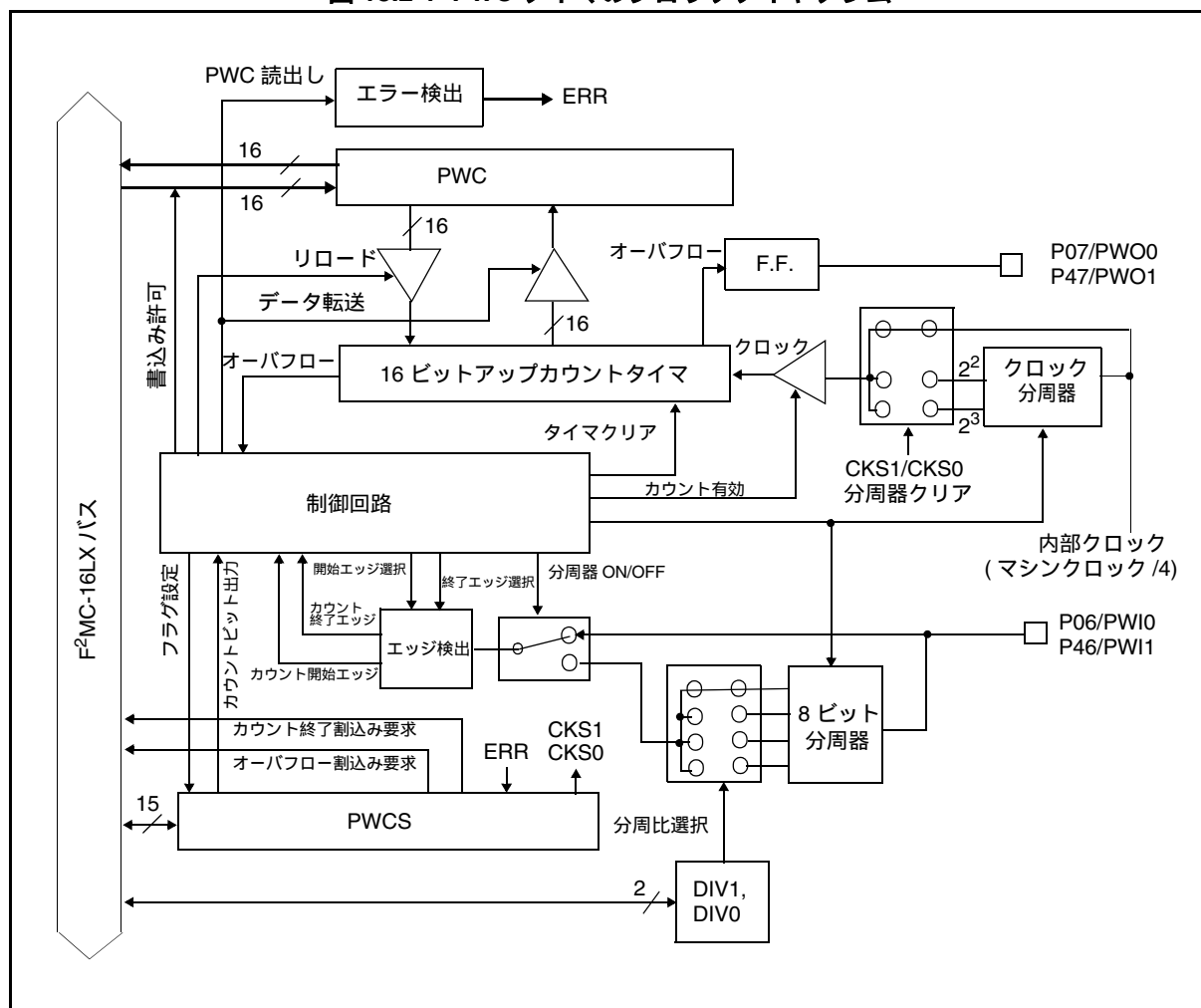
PWC タイマは、16 ビットアップカウントタイマを中核とした多機能タイマであり、カウント入力端子と 8 ビット入力分周器を備えています。PWC タイマは 2 つの主要機能 (タイマ機能とパルス幅測定機能) を備えており、これらの機能はそれぞれ 2 種類のカウンタクロックを選択できます。

13.2 PWC タイマのブロックダイアグラム

PWC タイマのブロックダイアグラムを図 13.2-1 に示します。

■ PWC タイマのブロックダイアグラム

図 13.2-1 PWC タイマのブロックダイアグラム



13.3 PWC タイマの端子

PWC タイマの端子について説明します。また端子のブロックダイアグラムも示します。

■ PWC タイマの端子

PWC タイマの端子は、汎用入出力ポートと共用されます。端子機能と入出力形式、PWC タイマを使用するために必要な設定を表 13.3-1 に示します。

表 13.3-1 PWC タイマの端子

端子名	端子機能	入出力形式	プルアップ オプション	待機制御	設定
P06/PWI0	ポート 0 入出力 / タイマ入力	CMOS 出力 / CMOS 入力	選択可能	使用可能	入力ポートの設定 (DDR0: bit6= 0)
P07/PWO0	ポート 0 入出力 / タイマ出力				タイマ有効の設定 (PWCSL0: MOD2 ~ MOD0 が 0 ではない)
P46/PWI1	ポート 4 入出力 / タイマ入力	CMOS 出力 / CMOS ヒステ リシス入力	なし		入力ポートの設定 (DDR4: bit6= 0)
P47/PWO1	ポート 4 入出力 / タイマ出力				タイマ有効の設定 (PWCSL1: MOD2 ~ MOD0 が 0 ではない)

■ PWC タイマ端子のブロックダイアグラム

図 13.3-1 に PWC タイマ 0 入力端子 (PWI0) のブロックダイアグラムを示します。

図 13.3-1 PWC タイマ 0 入力端子 (PWI0) のブロックダイアグラム

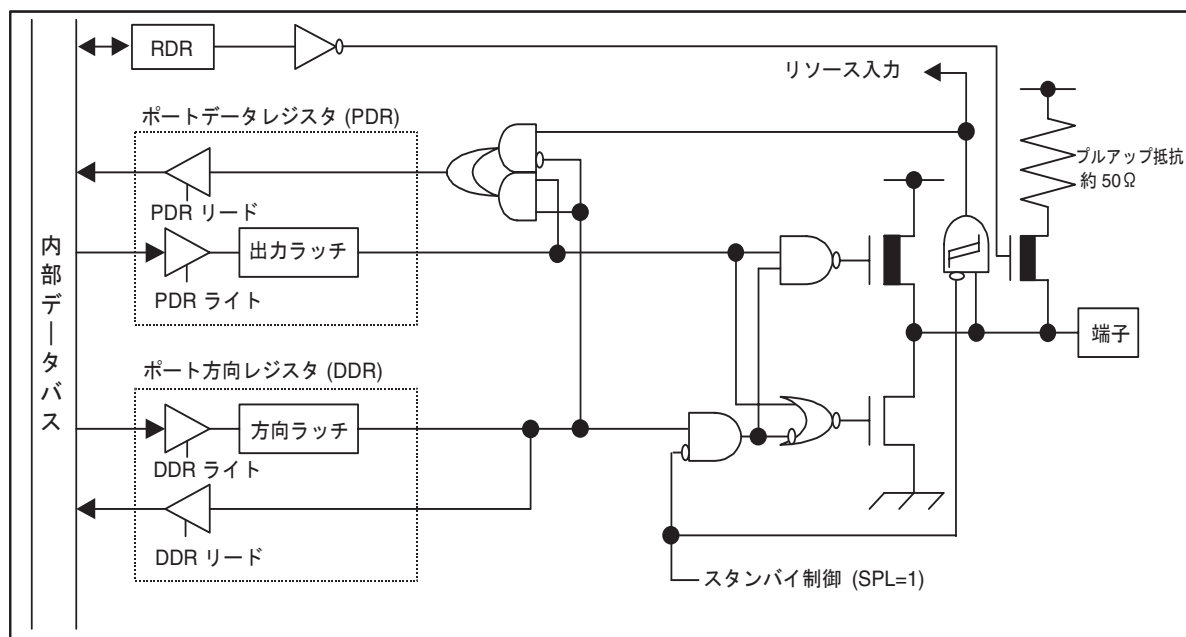


図 13.3-2 に PWC タイマ 0 の出力端子 (PWO0) のブロックダイアグラムを示します。

図 13.3-2 PWC タイマ 0 の出力端子 (PWO0) のブロックダイアグラム

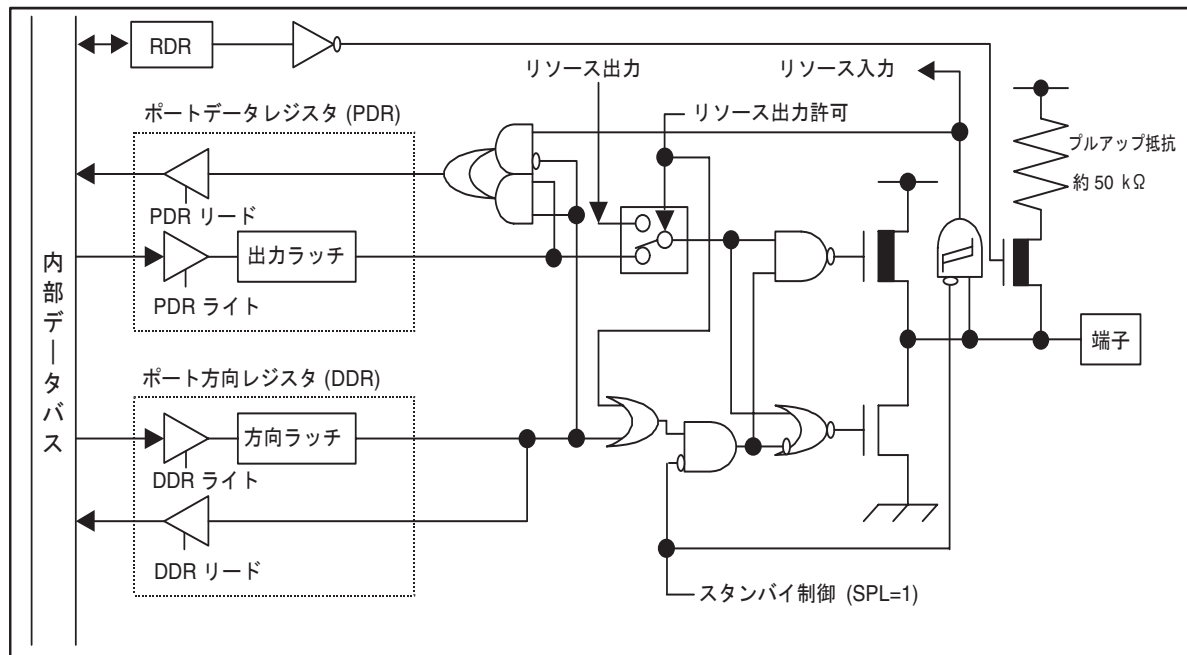


図 13.3-3 に PWC タイマ 1 の入力端子 (PWI1) のブロックダイアグラムを示します。

図 13.3-3 PWC タイマ 1 の入力端子 (PWI1) のブロックダイアグラム

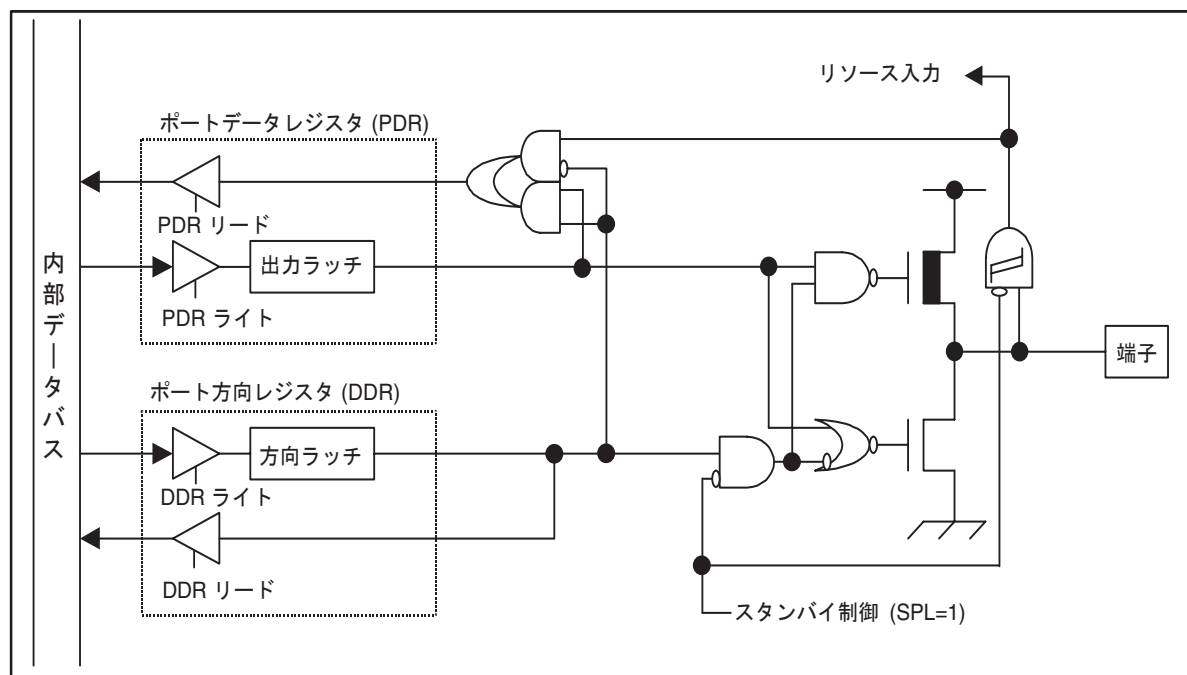
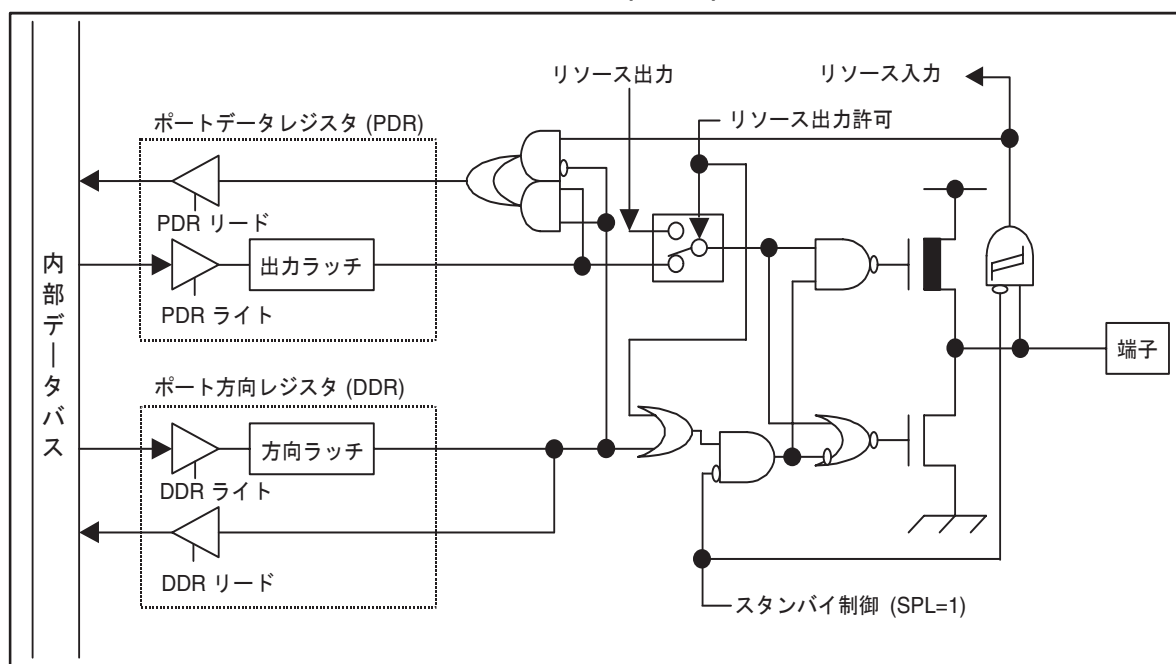


図 13.3-4 に PWC タイマ 1 の出力端子 (PWO1) のブロックダイアグラムを示します。

図 13.3-4 PWC タイマ 1 の出力端子 (PWO1) のブロックダイアグラム



13.4 PWC タイマのレジスタ

PWC タイマのレジスタ一覧を示します。

■ PWC タイマのレジスタ

図 13.4-1 PWC タイマのレジスタ

PWC 状態制御レジスタ (上位) PWCSH0, PWCSH1	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	初期値 00000000 _B
	STRT	STOP	EDIR	EDIE	OVIR	OVIE	ERR	POUT	
	R/W	R/W	W	R/W	R/W	R/W	W	R/W	
PWC 状態制御レジスタ (下位) PWCSL0, PWCSL1	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値 00000000 _B
	CKS1	CKS0	予約	予約	S/C	MOD2	MOD1	MOD0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PWC データバッファレジスタ (上位) PWC0, PWC1	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	初期値 XXXXXXXX _B
	PW15	PW14	PW13	PW12	PW11	PW10	PW09	PW08	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PWC データバッファレジスタ (下位) PWC0, PWC1	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値 XXXXXXXX _B
	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
分周比制御レジスタ DIV0, DIV1	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値 XXXXXX00 _B
	-	-	-	-	-	-	DIV1	DIV0	
	-	-	-	-	-	-	R/W	R/W	

13.4.1 PWC 状態制御レジスタ (PWCSH0/PWCSH1, PWCSL0/PWCSL1)

PWC 状態制御レジスタ (PWCSH0/PWCSH1, PWCSL0/PWCSL1) は, PWC タイマの動作を制御し, PWC タイマの状態を読み出します。

■ PWC 状態制御レジスタ, 上位バイト (PWCSH0/PWCSH1)

図 13.4-2 PWC 状態制御レジスタ (PWCSH0/PWCSH1)

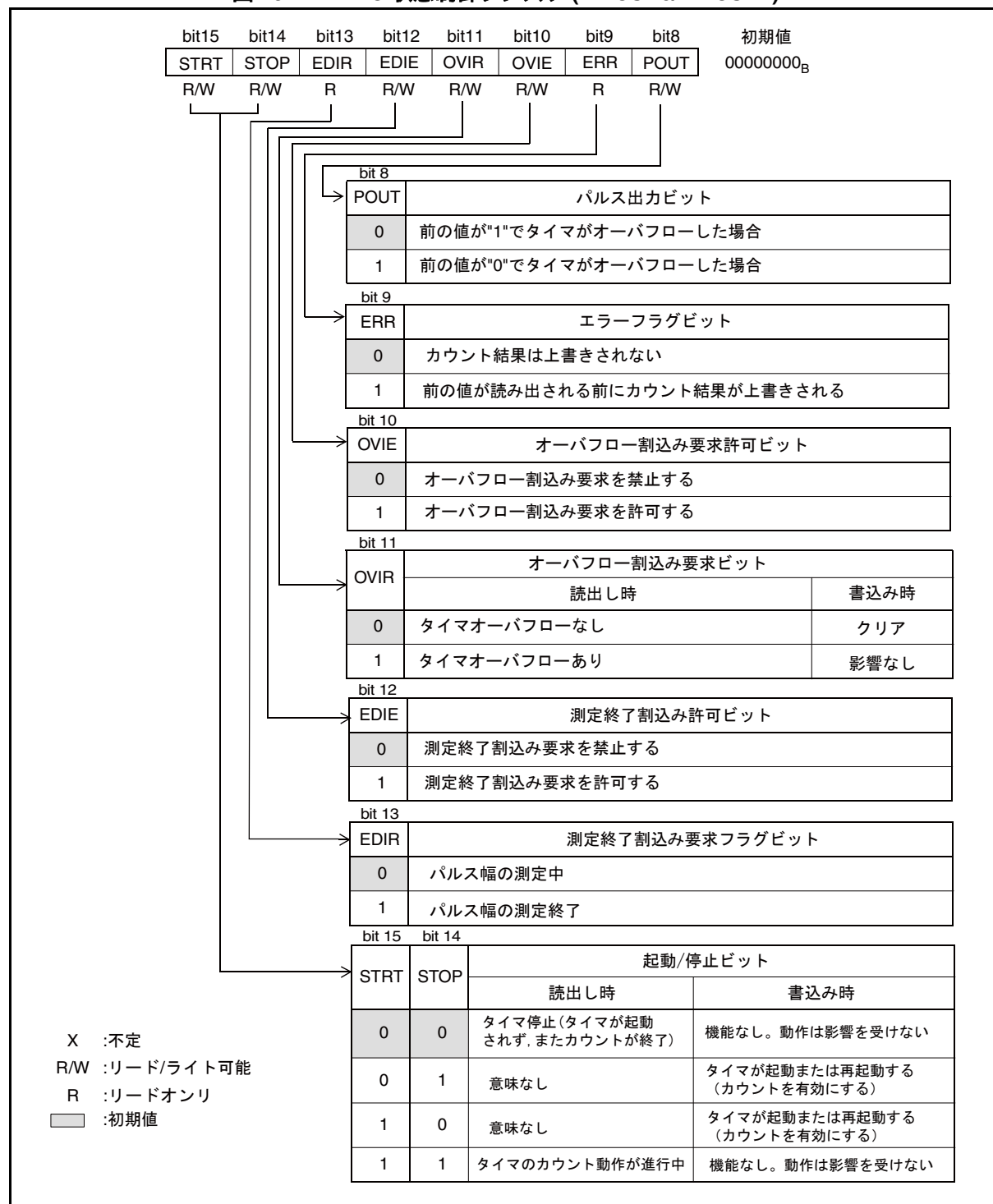


表 13.4-1 PWC 状態制御レジスタ (PWCSH0/PWCSH1) (1 / 2)

ビット名		機能
bit15, bit14	STRT, STOP: 起動 / 停止ビット	<ul style="list-style-type: none"> これらのビットは、16 ビットアップカウントタイマを起動 / 再起動および停止するために使用します。 これらのビットを読み出すと、タイマ動作状態が復帰します。 これらのビットは、リード / ライト可能です。これらのビットの意味は、リードまたはライトにより異なります。 リードモディファイライト動作時は、必ず "11_B" が読み出されます。 STRT, STOP ビットを書き込んでタイマを起動 / 停止する場合、ビット操作命令 (ビットクリア命令など) が使用可能です。ただし、動作状態 (タイマが動作中であることなどを必ず示す) の読出し時には、ビット操作命令を使用することはできません。
bit13	EDIR: 測定終了割込み 要求フラグビット	<ul style="list-style-type: none"> このビットは、パルス幅測定モードでの測定が終了したことを示します。 パルス幅測定が終了すると、このビットは設定されます (PWC0/PWC1 に測定結果が含まれます)。 PWC データバッファレジスタ内の測定結果や PWC0/PWC1 を読み出した場合に、このビットは自動的にクリアされます。 タイマモードでは、このビットは意味を持ちません。 このビットは読出し専用ビットであり、ライトは意味を持ちません。
bit12	EDIE: 測定終了割込み 許可ビット	<ul style="list-style-type: none"> このビットは、パルス幅カウントモード時の測定終了割込み要求を制御するために使用します。 このビットが "1" で、かつ EDIR ビットに "1" が設定されたとき、測定終了割込み要求が CPU に対して生成されます。 タイマモード時はこのビットには必ず "0" を設定してください。
bit11	OVIR: オーバーフロー 割込み要求ビット	<ul style="list-style-type: none"> このビットは、16 ビットアップカウントタイマのオーバーフローがいつ発生するかを指定するために使用します。動作はすべてのモードに影響を与えます。 タイマオーバーフローが ("FFFF_H" から "0000_H") 発生すると、ビットは設定されます。 "0" を設定した場合: このビットはクリアされます。 "1" を設定した場合: 影響はありません。 リードモディファイライト動作時は、必ず "1" 読み出されます。 (注意事項) "H"/"L" パルス幅カウントモード時は、パルス幅時間測定にこのビットを使用しないでください。
bit10	OVIE: オーバーフロー 割込み要求許可 ビット	<ul style="list-style-type: none"> このビットは、タイマオーバーフロー割込み要求を許可するために使用します。 このビットが "1" で、かつ OVIR が "1" に設定された場合、オーバーフロー割込み要求が CPU に対して生成されます。 (注意事項) "H"/"L" パルス幅カウントモード時は、このビットを "0" に設定してください。

表 13.4-1 PWC 状態制御レジスタ (PWCSH0/PWCSH1) (2 / 2)

ビット名		機能
bit9	ERR: エラー フラグビット	<ul style="list-style-type: none"> このビットは、パルス幅カウントモードで連続カウントするために使用します。このフラグは、前のカウント結果が PWC0/PWC1 レジスタから読み出される前に次のカウント動作が完了していることを示します。この状態が発生すると、PWC0/PWC1 レジスタは新しいカウント結果で上書きされ、前のカウント結果は失われます。カウント動作はこのビット値とは無関係に継続します。 このビットは読み出し専用ビットであり、このビットへのライトは意味を持ちません。 読み出されていないカウント結果が次のカウント結果で上書きされると、このビットが設定されます。 PWC データバッファレジスタ内の測定結果や PWC0/PWC1 を読み出したとき、このビットは自動的にクリアされます。
bit8	POUT: パルス出力ビット	<ul style="list-style-type: none"> タイマモード時で 16 ビットアップカウントタイマオーバーフローが発生すると、このビットは反転します。 パルス幅カウントモード時は、このビットは意味を持ちません。 このビットは、リード/ライト可能です。ただし、このビットへの書込みは、タイマが停止した場合 (bit15 : STRT と bit14 : STOP の両方に "0" が設定) に限り可能です。タイマ動作中 (bit15 : STRT と bit14 : STOP の両方に "1" が設定) にこのビットに値を書き込んでも、ビットの値は変化しません。 POUT 値が "0" であり、かつタイマオーバーフローが "FFFF_H" から "0000_H" の範囲において発生した場合、またタイマが停止し "1" が書き込まれた場合、このビットは設定されます。 POUT 値が "1" であり、かつタイマオーバーフローが "FFFF_H" から "0000_H" の範囲において発生した場合、またはタイマが停止し "0" が書き込まれた場合、このビットはクリアされます。このビットはリセットでもクリアされます。

■ PWC 状態制御レジスタ, 下位バイト (PWCSL0/PWCSL1)

図 13.4-3 PWC 状態制御レジスタ, 下位バイト (PWCSL0/PWCSL1)

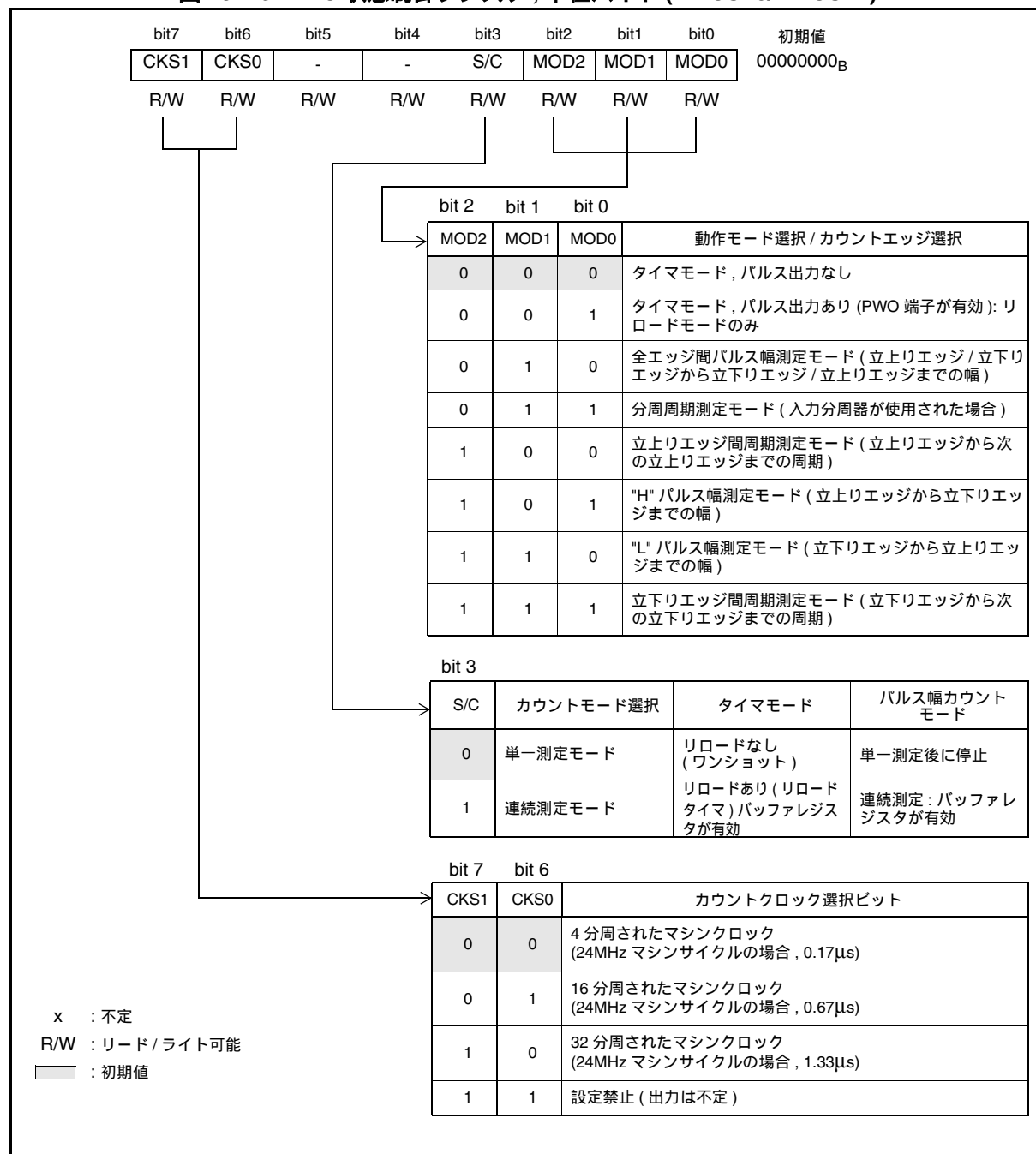


表 13.4-2 PWC 状態制御レジスタ (PWCSL0/PWCSL1)

ビット名		機能
bit7, bit6	CKS1, CKS0: クロック選択ビット	<ul style="list-style-type: none"> CKS1, CKS0 ビットは、内部カウントクロックを選択するために使用します。 リセット後、これらのビットは "00_B" に初期化されます。これらのビットはリード/ライト可能です。ただし、"11_B" を設定することはできません。 <p>(注意事項) タイマ起動後にこの設定を変更することは禁止されています。これらのビットへの書込みは、タイマ起動前またはタイマ停止後に行ってください。</p>
bit5, bit4	予約ビット	<ul style="list-style-type: none"> これらのビットは未定義ビットです。必ず "00_B" を書き込んでください。
bit3	S/C: カウントモード選択 ビット	<ul style="list-style-type: none"> このビットは、カウントモードを選択するために使用します。 リセット後、このビットは "0" に初期化されます。このビットはリード/ライト可能です。 <p>(注意事項) タイマ起動後にこの設定を変更することは禁止されています。このビットへの書込みは、タイマ起動前またはタイマ停止後に行ってください。</p>
bit2, bit1, bit0	MOD2, MOD1, MOD0: 動作モード選択 / カウントエッジ選択 ビット	<ul style="list-style-type: none"> これらのビットに値を設定すると、動作モードの選択、およびパルス幅カウントに適したパルスエッジの選択が可能になります。 リセット後、これらのビットは "000_B" に初期化されます。これらのビットはリード/ライト可能です。 <p>(注意事項) タイマ起動後にこの設定を変更することは禁止されています。これらのビットへの書込みは、タイマ起動前またはタイマ停止後に行ってください。</p> <ul style="list-style-type: none"> 連続測定モードを * マーク (前ページ) が付いた設定に対して設定すると、エッジ総数が算出され、内部カウントクロックの分周器はカウント終了時にクリアされません。ほかのすべてのモード時は、内部カウントクロックの分周器はカウント終了時にクリアされます。

13.4.2 PWC データバッファレジスタ (PWC0/PWC1)

PWC データバッファレジスタ (PWC0/PWC1) は、PWC タイマの動作モードに依存する機能を備えています。

■ PWC データバッファレジスタ (PWC0/PWC1)

図 13.4-4 PWC データバッファレジスタ (PWC0/PWC1)

PWC データバッファレジスタ (上位)									
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
PWC0, PWC1	PW15	PW14	PW13	PW12	PW11	PW10	PW09	PW08	XXXXXXXX _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

PWC データバッファレジスタ (下位)									
	bit7	bit6	bit5	bit4	bit3	bit 2	bit1	bit0	初期値
PWC0, PWC1	PW07	PW06	PW05	PW04	PW03	PW02	PW01	PW00	XXXXXXXX _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

● タイマモード

リロードタイマ動作モード (PWCSL0/PWCSL1: S/C = 1) 時は、このレジスタはリロード値を格納します。このレジスタは、リード/ライト可能です。

単一タイマ動作モード (PWCSL0/PWCSL1: S/C = 0) 時は、このレジスタへ直接アクセスするとアップカウントタイマがアクセスされます。このモード時は、このレジスタはリード/ライト可能です。ただし、このレジスタへの書込みは、タイマが停止した場合のみ行われます。このレジスタの読出しは常に可能であり、読出し値は現タイマ値です。

● パルス幅測定モード (リード専用)

連続測定モード (PWCSL0/PWCSL1: S/C = 1) 時は、このレジスタはバッファレジスタとして機能し、前のカウント結果を格納します。このレジスタはリード専用レジスタであり、値を書き込んでもレジスタ値は変化しません。

単一測定モード (PWCSL0/PWCSL1: S/C = 0) 時は、このレジスタへ直接アクセスするとアップカウントタイマがアクセスされます。このモード時も、このレジスタはリード専用レジスタであり、値を書き込んでもレジスタ値は変化しません。このレジスタの読出しは常に可能で、読出し値は現タイマ値です。カウントが実行されると、レジスタはカウント結果を格納します。

< 注意事項 >

このレジスタへアクセスする場合は、必ずワード転送命令をご使用ください。
リセット後、このレジスタは "0000_H" に初期化されます。

13.4.3 分周比制御レジスタ (DIV0/DIV1)

分周比制御レジスタ (DIV0/DIV1) は、分周周期測定モード (PWCSL:MOD2 ~ MOD0 = 011_B) 時に使用します。このレジスタは、ほかのモード時は意味を持ちません。

■ 分周比制御レジスタ (DIV0/DIV1)

図 13.4-5 分周比制御レジスタ (DIV0/DIV1)

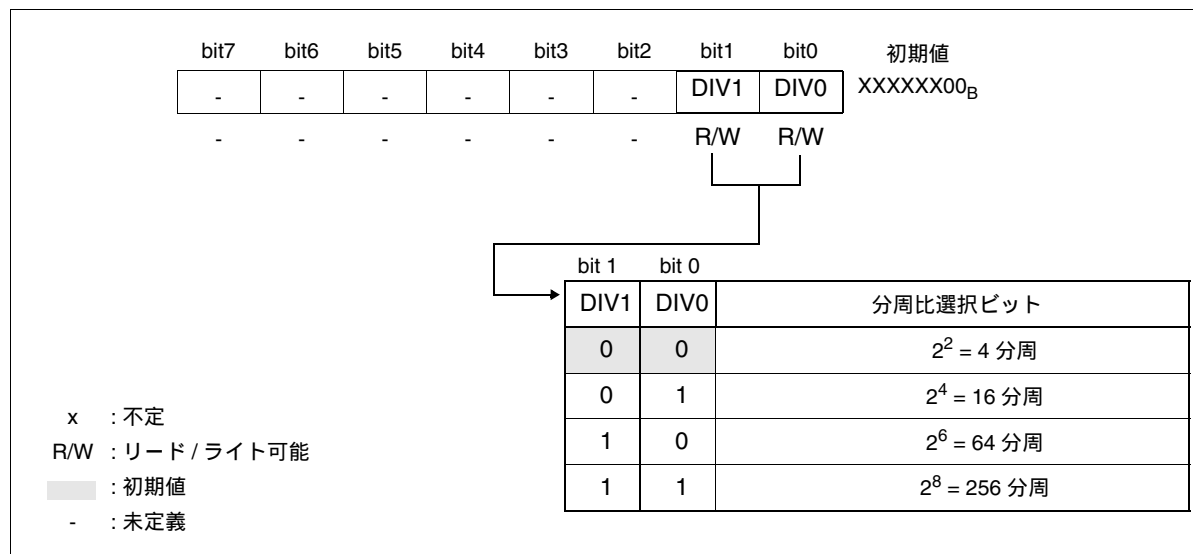


表 13.4-3 分周比制御レジスタ (DIV0/DIV1)

ビット名		機能
bit7 ~ bit2	未定義ビット	<ul style="list-style-type: none"> 読出し値は不定です。 これらのビットへライトは、動作に影響を与えません。
bit1, bit0	DIV1, DIV0: 分周比選択ビット	<ul style="list-style-type: none"> 分周範囲測定モード時は、このレジスタは測定端子からのパルス入力を分周し、分周後の 1 周期幅を測定するために使用します。 リセット後、これらのビットは "00_B" に初期化されます。これらのビットは、リード / ライト可能です。 <p>(注意事項)</p> <p>タイマ起動後にこの設定を変更することはできません。これらのビットへの書込みは、タイマ起動前またはタイマ停止後に行ってください。</p>

13.5 PWC タイマの割込み

PWC タイマは、カウンタオーバフローが発生した場合、またはパルス幅測定モードで測定が終了した場合に、割込み要求を生成できます。また、PWC タイマは拡張インテリジェント I/O サービス (EI²OS) とも連係しています。

■ PWC タイマの割込み

表 13.5-1 に PWC タイマの割込み制御ビットと割込み要因を示します。

表 13.5-1 PWC タイマの割込み制御ビットと割込み要因

	PWC タイマ 0		PWC タイマ 1	
割込み要求フラグビット	PWCSL0: OVIR	PWCSL0: EDIR	PWCSL1: OVIR	PWCSL1: EDIR
割込み要求許可ビット	PWCSL0: OVIE	PWCSL0: EDIE	PWCSL1: OVIE	PWCSL1: EDIE
割込み要因	16 ビットアップ カウンタの オーバフロー	パルス幅測定 モード時の測定 終了	16 ビットアップ カウンタの オーバフロー	パルス幅測定 モード時の測定 終了

PWC タイマでは PWC 状態制御レジスタ (PWCSL) の OVIR ビットは、アップカウンタのオーバフロー ("FFFF_H" から "0000_H") によって "1" に設定されます。この状態において割込み要求が許可 (PWCSL: OVIE = 1) になると、割込み要求が割込みコントローラへ出力されます。

PWC 状態制御レジスタ (PWCSL) の EDIR ビットは、パルス幅測定モード時の測定終了によって "1" にセットされます。この状態において割込み要求が許可 (PWCSL: EDIE = 1) になると、割込み要求が割込みコントローラへ出力されます。

■ PWC タイマ割込みと EI²OS

表 13.5-2 に PWC タイマ割込みと EI²OS を示します。

表 13.5-2 16 ビット PPG タイマ割込みと EI²OS

チャンネル	割込み 番号	割込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ	アドレス	下位	中位	上位	
PWC タイマ 0 ^{*1}	#13 (0D _H)	ICR01	0000B1 _H	FFFFC8 _H	FFFFC9 _H	FFFFCA _H	
PWC タイマ 1 ^{*2}	#24 (18 _H)	ICR06	0000B6 _H	FFFF9C _H	FFFF9D _H	FFFF9E _H	

^{*1}: 16 ビット PPG タイマ 0 の割込み番号と同じ番号が PWC タイマ 0 に割当てられます。

^{*2}: アウトプットコンペア ch.5 一致の割込み番号と同じ番号が PWC タイマ 1 に割当てられます。

■ PWC タイマの EI²OS 機能

PWC タイマは EI²OS と関係する回路を持っているので、カウンタは、オーバフローまたは測定終了が発生すると EI²OS を起動できます。

ただし EI²OS は、割込み制御レジスタ (ICR) を共用するほかのリソースが割込みを使用しない場合に限り使用可能です。例えば、PWC タイマ 0 が EI²OS を使用している場合、16 ビット PPG タイマ 0 の割込みは禁止する必要があります。

13.6 PWC タイマの動作

PWC タイマは、16 ビットアップカウントタイマを中核とした多機能タイマであり、カウント入力端子と 8 ビット入力分周器を備えています。PWC タイマは 2 つの主要機能 (タイマ機能とパルス幅測定機能) を備えており、これらの機能はそれぞれ 2 種類のカウンタクロックを選択できます。

■ タイマ機能

- タイマ機能は、単一モードまたはリロードモード時の動作選択を有効にするアップカウントタイマです。
- タイマが起動すると、各カウンタクロックでタイマはカウントします。
- "FFFF_H" から "0000_H" の範囲においてオーバーフローが発生すると、割込み要求が送出されます。

オーバーフローが発生すると、以下のイベントが発生します。

- 単一モード時は、カウントは停止します (図 13.6-1 を参照)。
- リロードモード時は、リロードレジスタの値がタイマにリロードされ、カウントが再起動します (図 13.6-2 を参照)。

図 13.6-1 タイマ動作 (単一モード)

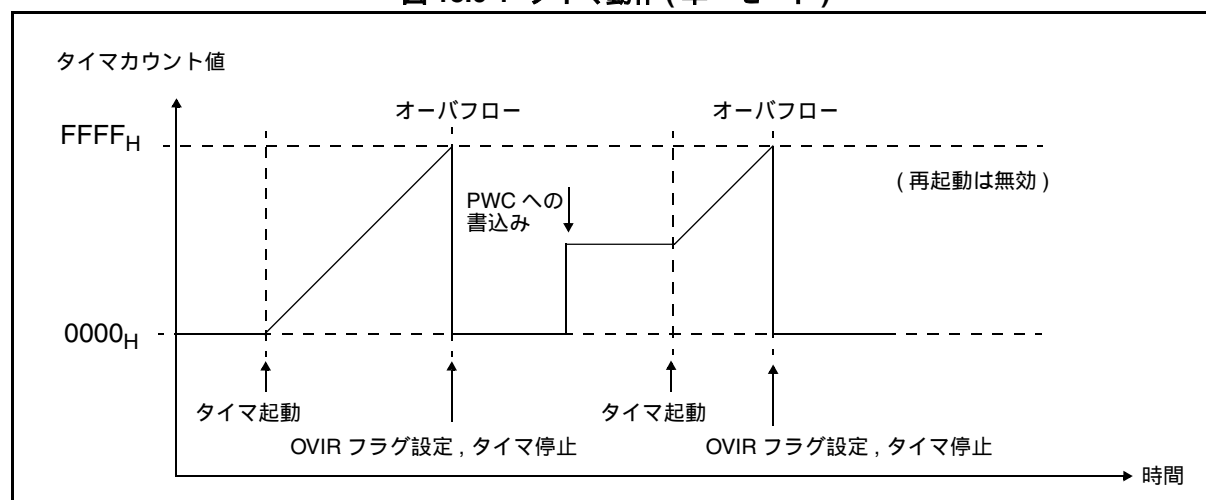
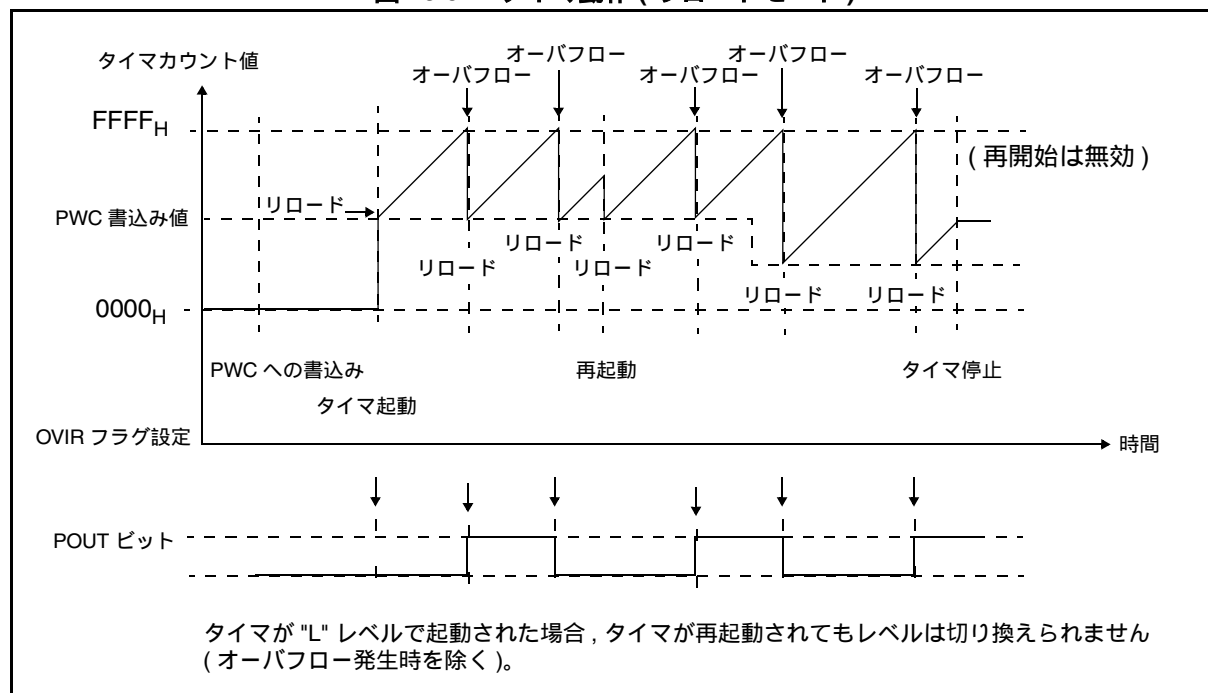


図 13.6-2 タイマ動作 (リロードモード)



■ パルス幅測定機能

- パルス幅測定機能は、入力パルスに関する指定されたイベント間の時間を算出します。
- この機能が起動されると、指定されたカウント開始エッジが入力された後にカウントが開始します。カウンタが "0000_H" にクリアされると、カウントは開始エッジ検出時に開始され、次に停止エッジが検出されます。この期間中のカウント値は、パルス幅としてレジスタに保持されます。

測定終了時またはオーバーフロー発生時に、割込み要求を生成できます。測定が完了すると以下のイベントが発生します。

- 単一測定モード
動作が停止します（図 13.6-3 を参照）。
- 連続測定モード
タイマ値がバッファレジスタへ転送され、次のエッジが入力されるまでタイマはフリーラン状態になります（図 13.6-4 を参照）。

図 13.6-3 パルス幅測定動作 (単一測定モード, "H" 幅測定モード)

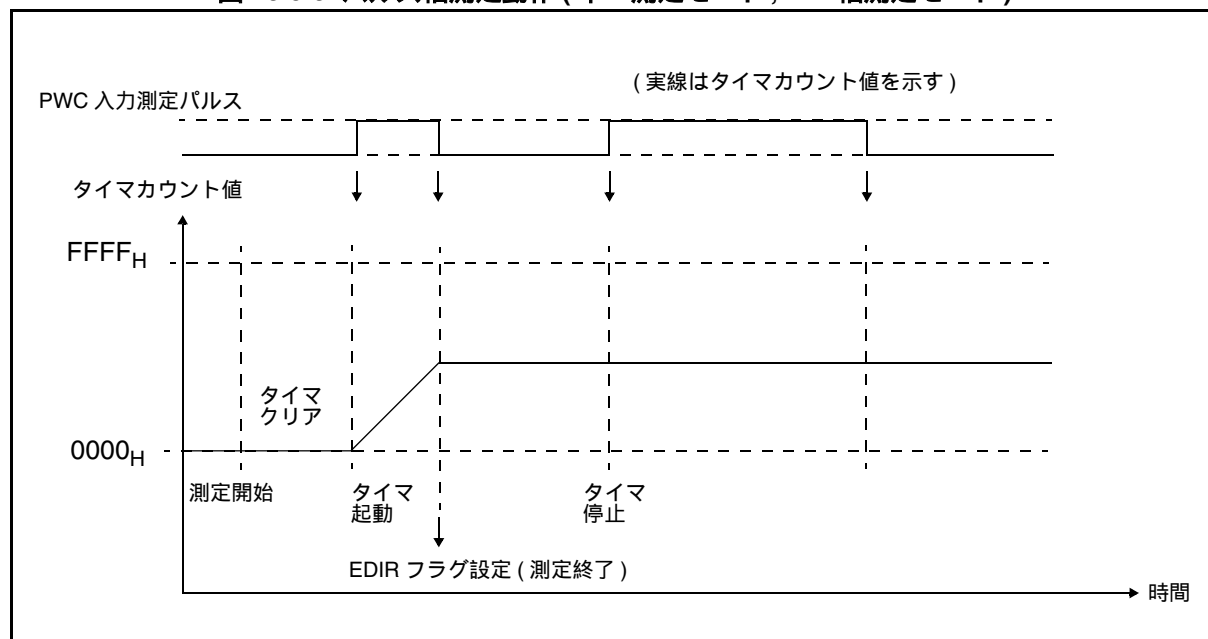
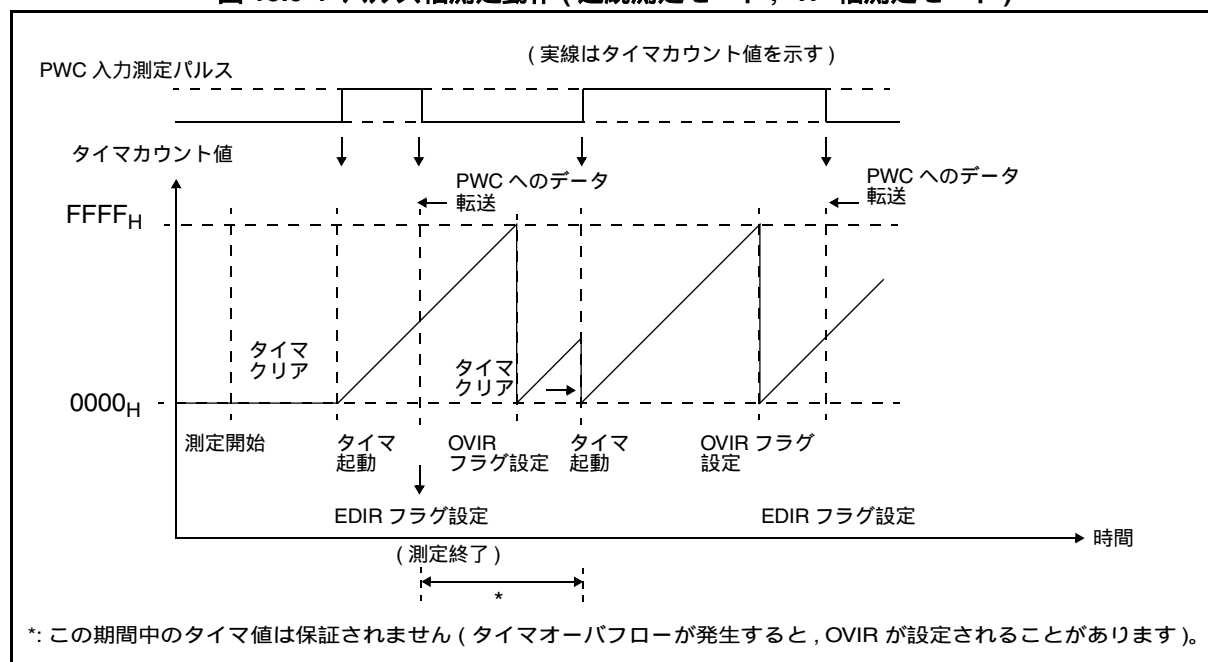


図 13.6-4 パルス幅測定動作 (連続測定モード, "H" 幅測定モード)



13.6.1 動作モードの選択

動作モードとカウントモードは、PWCSL レジスタの設定に従って選択されます。

■ 動作モードの選択

以下のレジスタは、動作モードとカウントモードの選択を設定するために使用されます。

- 動作モードの設定 : PWCSL : MOD2, MOD1, MOD0 ビット

カウント動作の制御を指定するには、タイマモードまたはパルス幅測定モードを選択してください。

- カウントモードの設定 : PWCSL : S/C ビット

単一測定、連続測定、リロード動作、またはワンショット動作を選択してください。

表 13.6-1 に動作モードビットで選択された動作モードを示します。

表 13.6-1 動作モードの選択

動作モード			S/C	MOD2	MOD1	MOD0
タイマ	ワンショットタイマ		0	0	0	0
	リロードタイマ		1	0	0	0 / 1
	設定禁止		0	0	0	1
パルス幅測定	立上りエッジか立下りエッジから立下りエッジか立上りエッジまで : 全エッジ間測定	単一測定 : バッファは無効	0	0	1	0
		連続測定 : バッファは有効	1	0	1	0
	分周カウント : 4 分周から 256 分周まで	単一測定 : バッファは無効	0	0	1	1
		連続測定 : バッファは有効	1	0	1	1
	立上りエッジから次の立上りエッジまで : 立上りエッジから次の立上りエッジまでの周期測定	単一測定 : バッファは無効	0	1	0	0
		連続測定 : バッファは有効	1	1	0	0
	立上りエッジから立下りエッジまで : "H" パルス幅測定	単一測定 : バッファは無効	0	1	0	1
		連続測定 : バッファは有効	1	1	0	1
	立下りエッジから立上りエッジまで : "L" パルス幅測定	単一測定 : バッファは無効	0	1	1	0
		単一測定 : バッファは無効	1	1	1	0
	立下りエッジから次の立下りエッジまで : 立下りエッジから次の立下りエッジまでの周期測定	単一測定 : バッファは無効	0	1	1	1
		連続測定 : バッファは有効	1	1	1	1

リセット後、ワンショットタイマが初期値として選択されます。

< 注意事項 >

タイマを起動する前に、必ず動作モードを選択してください。

13.6.2 タイマとパルス幅測定 of 起動 / 停止とタイマクリア

タイマとパルス幅測定を起動，再起動，強制的に停止するには，PWCSH0/
PWCSH1：STRT および PWCSH0/PWCSH1：STOP を使用してください。

16 ビットアップカウントタイマは，リセット時，測定開始エッジ検出時，またはパルス幅測定モード時のカウント開始時に "0000_H" にクリアされます。

■ タイマとパルス幅測定 of 起動 / 停止

PWCSH0/PWCSH1: STRT ビットに "0" を書き込むと，動作が起動または再起動され，PWCSH0/PWCSH1: STOP ビットに "0" を書き込むと，動作が停止します。ただし，これら 2 つのビットに書き込まれる値が同じである限り，これらのビットは動作を実行しません。ビット操作命令以外の命令（バイト命令またはワード命令）を使用した場合は，値の書込みは以下のビット組合せに対してのみ行われます。

表 13.6-2 パルス幅測定動作（単一測定モード，"H" 幅測定モード）

機能	STRT	STOP
タイマまたはパルス幅測定を起動 / 再起動する	0	1
タイマまたはパルス幅測定を停止する	1	0

ビット操作命令のクリアビット命令を使用する場合，ハードウェアは自動的に上記の値の組合せを書き込みます。ユーザは，どの値を書き込むべきかについて意識する必要はありません。

● 起動後の動作

タイマモード： カウント動作を直ちに開始します。

パルス幅測定モード： 測定は，測定開始エッジが入力された後に開始します。測定開始エッジが検出されると，16 ビットアップカウントタイマは "0000_H" にクリアされカウントを開始します。

● タイマの再起動

タイマモードまたはパルス幅測定モード時にタイマを起動し，その後にタイマ動作が継続されている間にタイマを再起動する（PWCSH0/PWCSH1: STRT ビットに "0" を書き込む）ことを，「タイマ再起動」と呼びます。再起動中に実行される動作は，以下に示すモードによって異なります。

ワンショットモード： 動作は影響を受けません。

リロードタイマモード： リロードが実行され，動作は継続します。オーバフロー発生時にタイマが再起動されると， オーバフローフラグ（PWCSH0/PWCSH1: OVIR）が設定され，POUT ビットが反転されます。

パルス幅測定モード： 測定開始エッジ待ち状態時は，動作は影響を受けません。測定中，カウントは停止しタイマ状態は「測定開始エッジ待ち」状態へ戻ります。測定終了時にタイマが再起動されると，測定終了フラグ（PWCSH0/PWCSH1: EDIR ビット）が設定され，連続測定モード時に測定結果は PWC へ転送されます。

● タイマの停止

ワンショットタイマモードまたは単一測定モード時は、測定はタイマオーバフロー発生時またはカウント終了時に自動的に停止します。ユーザは、タイマが停止したか否かを意識する必要はありません。ただし、ほかのモード時をタイマは停止しなればなりません。これはタイマが自動的に停止する前にタイマを停止する場合も含まれます。

● 動作状態の検査

既述の STRT ビットと STOP ビットは、読出し中、タイマの動作状態を示すビットとして機能します。

表 13.6-3 に動作状態指示ビットの機能を示します。

表 13.6-3 動作状態指示ビットの機能

STRT	STOP	動作状態
0	0	タイマは停止している（測定開始エッジ待ち状態を除く）。これらのビットは、タイマが起動されていないか、あるいは測定が終了していることを示します。
1	1	測定開始エッジ待ち状態またはタイマカウント動作

読出し中、STRT ビットと STOP ビットは両方とも同じ値となります。ただし、リードモディファイライト命令を使用しての読出し中は、これらのビットの値は常に "11_B" になります。リードモディファイライト命令を使用して、これらのビットの値を読み出さないでください。

■ タイマクリア

以下の場合、16 ビットアップカウントタイマは "0000_H" にクリアされます。

- リセット時
- パルス幅測定モードでカウント開始エッジが検出された後でカウントが開始した時

13.6.3 タイマモード時の動作

タイマモードには、ワンショット動作モードとリロード動作モードがあります。

■ ワンショット動作モード

このモード時にタイマが起動すると、カウンタはカウントクロックごとにインクリメントされます。タイマは、オーバフローが "FFFF_H" から "0000_H" で発生すると自動的に停止します。

タイマが起動される前に PWC0/PWC1 が設定されると、カウントはこの設定値から開始します。オーバフローが発生すると、設定値が削除され、現カウント値が PWC0/PWC1 に残ります。

オーバフローが発生すると、PWCSH0/PWCSH1: POUT は反転します。

■ リロード動作モード

このモード時にタイマが起動すると、PWC0/PWC1 のリロード値がタイマへ設定され、カウンタがカウントクロックごとにインクリメントされます。タイマが "FFFF_H" から "0000_H" までをカウントしている間にオーバフローが発生すると、PWC0/PWC1 のリロード値がタイマへ再び設定され、PWCSH0/PWCSH1: POUT ビットが反転し、カウント動作が反復します。PWCSH0/PWCSH1: STOP ビット（タイマを停止する）に値が書き込まれるか、リセットが発生するまでは、タイマは停止しません。ポートビットは、パルス出力モードが指定されると、PWO0/PWO1 端子に出力されます。

タイマが起動される前に PWC0/PWC1 に設定されたリロード値は、カウント時に格納されます。タイマが起動または再起動し、オーバフローが発生すると、タイマにリロード値が必ず設定されます。カウント時に設定された値が変更されると、新しいリロード値は、次のオーバフローが発生した場合、またはタイマが再起動された時に有効になります。

■ タイマ値とリロード値

ワンショット動作モード時で、PWC レジスタへ直接アクセスするとアップカウントタイマにアクセスされます。PWC0/PWC1 に値が書き込まれると、この値はタイマへ直接書き込まれます。カウント動作時に PWC0/PWC1 が読み出されると、現タイマ値が読み出されます。タイマが起動される前に値が PWC に設定されると、タイマは指定された値からカウントを開始します。

リロード動作モード時は、アップカウントタイマはアクセスできず、PWC0/PWC1 はリロードレジスタ（リロード値を格納する）として機能します。タイマが起動または再起動し、オーバフローが発生すると、PWC に書き込まれた値は必ずタイマへ設定されます。PWC0/PWC1 が読み出されると、格納されたリロード値が読み出されます。

PWC 値とタイマ値は、リロードモード時に停止された後でタイマがワンショットモードに設定されると、不定となります。したがって、必ずタイマを使用する前に値を設定してください。

PWC 値は、ワンショット動作モード時に強制停止された後でタイマがリロードモードに設定されると、不定となります。したがって、必ずタイマを使用する前に値を設定してください。

■ 割込み要求の生成

タイマモード時の動作時に、オーバーフローによって割込み要求の生成が可能になります。タイマカウン트의インクリメントによりオーバーフローが発生した場合、オーバーフローフラグが設定され、オーバーフロー割込み要求がイネーブルになり、割込み要求が生成されます。

■ タイマ周期

PWC0/PWC1 に "0000_H" が設定された後にタイマがワンショットモードで起動された場合、カウン트가 65536 を超えると、タイマオーバーフローが発生し、カウン트는停止します。以下の公式を使用してタイマの起動から停止までの時間を計算します。

$$T_1 = (65536 - n_1) \times t \quad \left\{ \begin{array}{l} T_1 \cdots \cdots \text{タイマの起動から停止までの時間} (\mu\text{s}) \\ n_1 \cdots \cdots \text{タイマ起動時に PWC に設定されたタイマ値} \\ t \cdots \cdots \text{カウントクロック周期} (\mu\text{s}) \end{array} \right.$$

PWC0/PWC1 に "0000_H" が設定された後にタイマが起動されると、カウン트가 65536 を超えるたびにタイマオーバーフローが発生します。以下の公式を使用してリロード周期と PWO 端子出力パルス周期を計算します。

$$T_R = (65536 - N_R) \times t \quad \left\{ \begin{array}{l} T_R \cdots \cdots \text{リロード周期 (オーバーフロー周期)} (\mu\text{s}) \\ T_{\text{POUT}} \cdots \cdots \text{PWO0/PWO1 端子出力パルス周期} (\mu\text{s}) \\ N_R \cdots \cdots \text{PWC0/PWC1 に格納されたりロード値} (\mu\text{s}) \\ t \cdots \cdots \text{タイマの起動から停止までの時間} (\mu\text{s}) \end{array} \right.$$

■ カウントクロック周期と最大周期

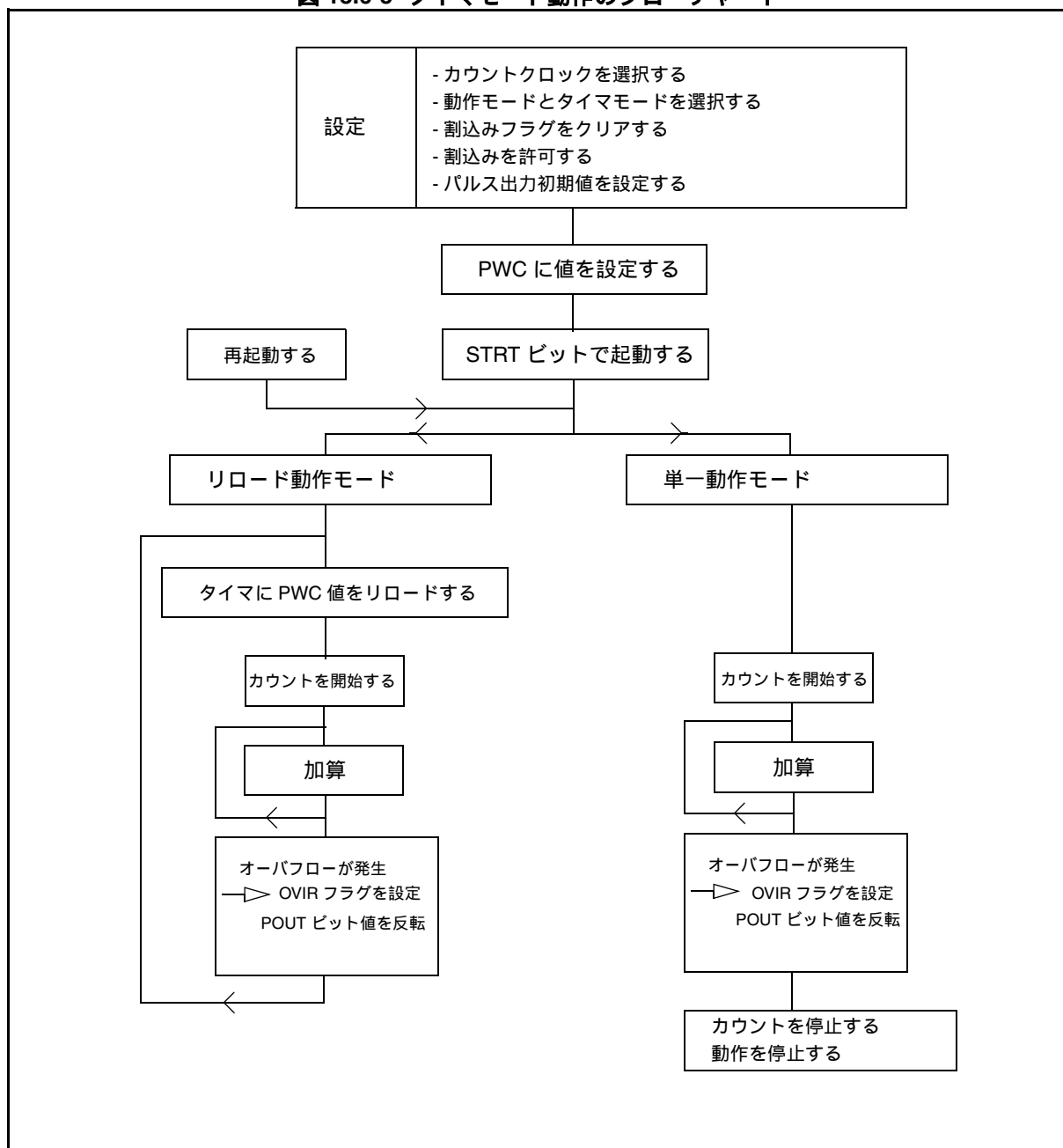
タイマモード時は、PWC0/PWC1 に "0000_H" が設定されると、最大周期を発生します。
表 13.6-4 にマシクロック 24 MHz(表中では ϕ で表示) に対応するカウントクロック周期とタイマ最大周期を示します。

表 13.6-4 カウントクロック周期と最大周期

カウントクロック選択	CKS1, CKS0 = 00 _B の場合 ($\phi/4$)	CKS1, CKS0 = 01 _B の場合 ($\phi/16$)	CKS1, CKS0 = 10 _B の場合 ($\phi/32$)
カウントクロック周期	0.17 μs	0.67 μs	1.33 μs
タイマ最大周期	10.92 ms	43.69 ms	87.38 ms

■ タイマモード動作のフローチャート

図 13.6-5 タイマモード動作のフローチャート



13.6.4 パルス幅測定モード時の動作

パルス幅を測定するための信号が PWI 端子から入力されます。

パルス幅測定モードには、1 回のみカウントする単一測定モードと、パルス幅が連続的に測定される連続測定モードがあります。

■ 単一測定モードと連続測定モード

単一測定モードと連続測定モードの違いを以下に示します。

● 単一測定モード

先頭のカウント終了エッジが入力されると、タイマはカウントを停止し、PWCSH0/PWCSH1 レジスタのカウント終了フラグ (EDIR) が設定され、以降測定は行われません。ただし、タイマの再起動も指定されると、タイマ状態は測定開始エッジ待ち状態に変化します。

● 連続測定モード

["H"/"L" パルス幅測定モード]

カウント終了エッジが入力されると、PWCSH0/PWCSH1 レジスタのカウント終了フラグ (EDIR) が設定され、タイマカウント結果が PWC に転送され、タイマはフリーラン状態でカウントのインクリメントを継続できます。次のカウント開始エッジが入力されると、タイマは "0000_H" にクリアされ、パルス幅カウントを開始します。

< 注意事項 >

カウント終了エッジが入力され、タイマがフリーラン状態になることにより、タイマがオーバフローすることがあり、OVIR フラグがセットされることがありますので、"H"/"L" パルス幅測定モードでは、OVIR フラグを用いたパルス幅時間測定は行わないでください。

[全エッジ間パルス幅測定モード、分周周期測定モード、立上りエッジ間周期測定モード、立下りエッジ間周期測定モード]

カウント終了エッジ (カウント開始エッジ) が入力されると、PWCSH0/PWCSH1 レジスタのカウント終了フラグ (EDIR) が設定され、タイマカウント結果が PWC0/PWC1 へ転送され、タイマは "0000_H" にクリアされ、カウントは再び開始します。

■ 測定結果データ

測定結果、タイマの値、PWC0/PWC1 機能に対する処理は、単一測定モードと連続測定モード時では以下のように異なります。

● 単一測定モード

タイマ動作時に PWC0/PWC1 が読み出されると、現タイマ値が読み出されます。

測定終了後に PWC0/PWC1 が読み出されると、測定結果が読み出されます。

● 連続測定モード

測定終了時にタイマ測定結果が PWC0/PWC1 へ転送されます。

PWC 読出し時には、前の測定結果が読み出されます。測定が進行中の場合は、前の測定結果が PWC0/PWC1 に格納されます。測定時には、タイマ値を読み出すことはできません。

連続測定モード時は、次の測定が完了する前に前の測定結果を読み出さない場合は除き、既存の値は新しい測定結果で上書きされます。この場合、PWCSH0/PWCSH1 レジスタのエラーフラグビット (ERR) が設定されます。PWC0/PWC1 が読み出されると、エラーフラグビット (ERR) は自動的にクリアされます。

■ 最小入力パルス幅

下記の最小入力パルス幅よりも長いパルスをパルス幅カウント入力端子 (PWI0/PWI1) へ入力する必要があります。

パルス幅: 2 マシンサイクル (マシンクロック 24MHz の場合, 83.3ns 以上)

ただし、上記の仕様よりも短い入力パルスも有効パルスとして認識されることもあります。

■ パルス幅 / 周期の算出

測定オブジェクトのパルス幅またはパルス周期は、以下のように、カウント終了時に PWC0/PWC1 から読み出されたカウント結果に基づいて算出されます。

$$T_W = n \times t / \text{Div} (\mu\text{s})$$

{

T_W …… 測定されたパルス幅またはパルス周期 (μs)

n …… PWC0/PWC1 に格納されている測定結果

t …… カウントクロック周期 (μs)

Div …… 分周比レジスタ (DIV0/DIV1) に設定されている分周比
(値 "1" は、分周カウントモード以外のモードで使用されます)

■ パルス幅 / 周期測定範囲

測定可能なパルス幅 / 周期の範囲は、入力分周器のカウントクロックと分周比によって異なります。

表 13.6-5 にマシンサイクル 24MHz 時の (ϕ で表示) の測定範囲を示します。

表 13.6-5 パルス幅測定範囲

分周比	DIV1, DIV0	CKS1, CKS0 = 00 _B ($\phi/4$)	CKS1, CKS0 = 01 _B ($\phi/16$)	CKS1, CKS0 = 10 _B ($\phi/32$)
分周なし	-	83.3 ns ~ 10.92ms [0.17 μs]	83.3 ns ~ 43.7ms [0.67 μs]	83.3 ns ~ 87.38ms [1.33 μs]
4 分周	00 _B	83.3 ns ~ 2.73ms [41.7 ns]	83.3 ns ~ 10.92ms [0.17 μs]	83.3 ns ~ 21.85ms [333 ns]
16 分周	01 _B	83.3 ns ~ 682.7 μs [10.4 ns]	83.3 ns ~ 2.73ms [41.7 ns]	83.3 ns ~ 5.46ms [83.3 ns]
64 分周	10 _B	83.3 ns ~ 170.7 μs [2.60 ns]	83.3 ns ~ 682.7 μs [10.4 ns]	83.3 ns ~ 1.37ms [20.83 ns]
256 分周	11 _B	83.3 ns ~ 42.7 μs [0.65 ns]	83.3 ns ~ 170.7 μs [2.60 ns]	83.3 ns ~ 0.34ms [5.21 ns]

(注意事項) [] 内の数字はビット当りの分解能を示します。

■ 割込み要求の生成

パルス幅測定モード時は、以下の 2 つの割込み要求を生成できます。

● タイマオーバーフロー割込み要求

カウント時にオーバーフローが発生すると、オーバーフローフラグが設定されます。オーバーフロー割込み要求が許可されると、割込み要求が生成されます。

● 測定終了割込み要求

測定終了エッジが検出されると、PWCSH0/PWCSH1 レジスタのカウント終了フラグ (EDIR) が設定されます。測定終了割込みが許可されると、割込み要求が生成されます。測定終了割込み要求フラグビット (EDIR) は、PWC0/PWC1 が読み出されると自動的にクリアされます。

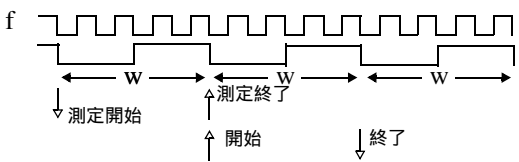
■ 測定モードと測定動作

表 13.6-6 に測定モード時における動作を示します。

表 13.6-6 測定モード時における動作

測定モード	MOD2	MOD1	MOD0	測定動作
"H" パルス幅測定	1	0	1	 <p>"H" 周期幅が測定されます 測定開始: 立上りエッジが検出されたとき 測定終了: 立下りエッジが検出されたとき</p>
"L" パルス幅測定	1	1	0	 <p>"L" 周期幅が測定されます 測定開始: 立上りエッジが検出されたとき 測定終了: 立下りエッジが検出されたとき</p>
立上りエッジ間 周期測定	1	0	0	 <p>立上りエッジ間時間が測定されます 測定開始: 立上りエッジが検出されたとき 測定終了: 立下りエッジが検出されたとき</p>
立下りエッジ間 周期測定	1	1	1	 <p>立下りエッジ間時間が測定されます 測定開始: 立上りエッジが検出されたとき 測定終了: 立下りエッジが検出されたとき</p>
全エッジパルス 幅測定	0	1	0	 <p>連続入力エッジ間の幅が測定されます 測定開始: 立上りエッジが検出されたとき 測定終了: 立下りエッジが検出されたとき</p>

表 13.6-6 測定モード時における動作

測定モード	MOD2	MOD1	MOD0	測定動作
分周測定	0	1	1	 <p>(上記の例では4分周されている) 入力パルスは分周比レジスタ (DIV0/DIV1) に設定されている分周比で分周され、測定周期が求められる。 測定開始: 動作が起動された後、立下りエッジが検出されます 測定終了: 分周信号の1周期が終了する</p>

W: 測定されるパルス幅

すべてのモードにおいて、測定が開始されてから測定開始エッジが入力されるまでの間、タイマはカウントを開始しません。測定開始エッジが入力されると、タイマは "0000_H" にクリアされ、カウントは測定終了エッジが入力されるまでカウントクロックごとにインクリメントされます。

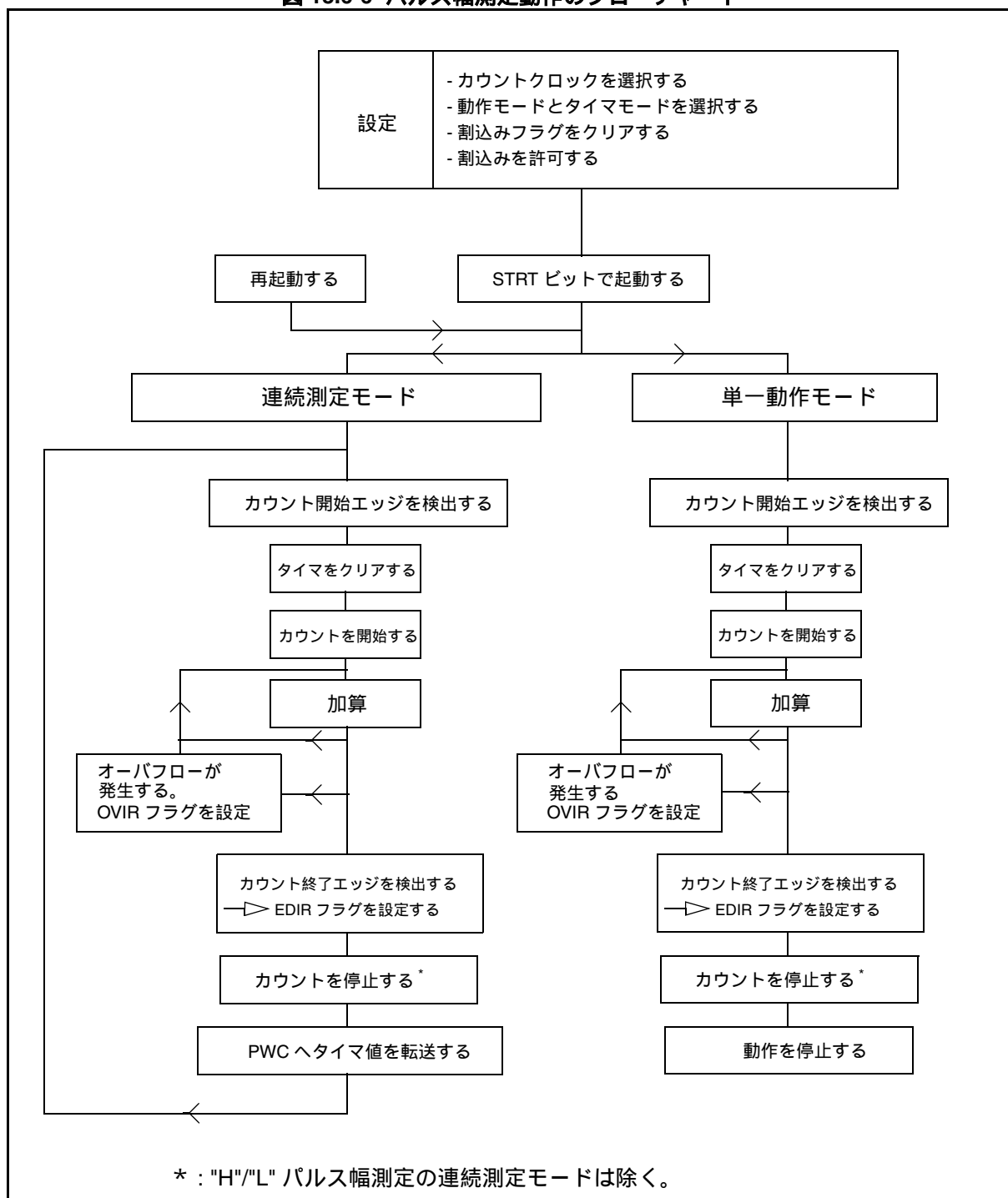
測定終了エッジが入力されると、以下の動作が実行されます。

- (1) PWCSH0/PWCSH1 レジスタのカウント終了フラグ (EDIR) が設定されます。
- (2) タイマがカウント動作を停止します (タイマが測定終了エッジ入力時と同時に再起動された場合、または "H"/"L" パルス幅測定の連続測定モードが使用された場合は除きます)。
- (3) 連続測定モード: タイマ値 (測定結果) が PWC0/PWC1 へ転送されます。
- (4) 単一測定モード: 測定が終了する (タイマが測定終了エッジ入力時と同時に再起動された場合は除く)。

連続測定モードで全エッジ間パルス幅測定、周期測定、立下りエッジ間周期測定、または立上りエッジ間周期測定が行われる場合、終了エッジが次の測定開始エッジになります。

■ パルス幅測定動作のフローチャート

図 13.6-6 パルス幅測定動作のフローチャート



13.7 PWC タイマ使用上の注意

PWC タイマ使用の上の注意点を以下に示します。

■ PWC タイマ使用上の注意

● プログラム設定上の注意

- 以下の PWCS0/PWCS1 レジスタのビット値は、タイマ動作時に変更することは禁止されています。これらのビット値の変更は、タイマが起動される前かまたは動作が停止された後でのみ行われます。

[bit7, bit6] CKS1, CKS0: カウントクロック選択ビット

[bit3] S/C: 測定モード (単一または連続) 選択ビット

[bit2 ~ bit0] MOD2, MOD1, MOD0: 動作モードおよび測定エッジ選択ビット

タイマ動作時にパルス出力レベル表示ビット (POUT: bit8) に値が書き込まれても、このビットの値は変化しませんのでご注意ください。

- タイマ動作時における DIV0/DIV1 値の変更は禁止されています。DIV0/DIV1 レジスタ値の変更はタイマが起動される前かまたは動作が停止された後で行ってください。
- PWC 態制御レジスタ (PWCSL0/PWCSL1) のカウントクロック選択ビット (CKS1, CKS0) に "11_B" を設定することは禁止されています。
- PWC0/PWC1 の値とタイマ値は、タイマがワンショットモードで設定されたとき、または動作がリロードタイマモードで終了した後に決定されます。したがって、これらの値の設定は必ずタイマを使用した後で行ってください。
- PWC0/PWC1 の値は、動作がワンショットモードで停止された後にタイマがリロードタイマモードで設定されると不定となります。したがって、PWC 値は必ずタイマを使用する前に設定してください。
- モードをパルス幅測定モードからタイマモードへ変更する場合は、PWC0/PWC1 への値は必ずタイマが起動される前に設定してください。
- パルス幅測定モード時に分周周期測定モードが使用されると、入力パルスは分周されます。カウント結果から算出されたパルス幅は平均値になるので注意してください。
- パルス幅測定モード時に連続測定が実行されている間は、内部カウントクロックの分周回路はクリアされず、カウントクロックよりも小さいエッジ数がカウント結果に加算されます。

● プログラムで状態を検査する際の注意

- タイマモード時は、PWCSH0/PWCSH1 レジスタの測定終了割込み要求フラグビット (EDIR) の値は意味がありません。したがって、PWCSH0/PWCSH1 レジスタの測定終了割込み許可ビット (EDIE) に必ず "0" を設定してください。
- PWC 状態制御レジスタ上位 (PWCSH0/PWCSH1) の STRT ビットと STOP ビットの動作は、読出し時と書込み時では異なります (レジスタの詳細を参照)。リードモディファイライト命令では、これらのビットは必ず "11_B" に読み出されます。したがって、ビット操作命令を使用して動作状態を読み出すことはできません。
ただし、ビット操作命令 (ビットクリア命令) は、STRT ビットまたは STOP ビットに値を書き込むことによってタイマを起動または停止する際には使用できます。
- パルス幅測定モード時は、測定開始エッジによってタイマはクリアされます。また、前のタイマデータは意味がありません。

● パルス幅測定入力端子に入力されるパルスに関する注意

- 最小パルス幅は、マシンサイクルを 2 分周した幅です (マシンサイクルが 24MHz の場合は 83.33 ns 以上)。
- 最大入力周波数は、マシンサイクルを 4 分周した幅です (マシクロックが 24MHz の場合は 4MHz 未満)。
- 上記よりも小さいパルス幅または上記よりも大きい周波数が入力されると、タイマ動作は保証されません。上記の制約を超える入力信号のノイズは、低減する必要があります。

● 動作時のタイマ再起動に関する注意

- リロードタイマモード時にオーバフローが発生すると、タイマが再起動しますが、オーバフローフラグ (OVIR) が設定され、POUT ビットが反転します (つまり、通常のオーバフローの場合と同じ動作になります)。
- ワンショットパルス幅測定モード時に終了エッジが検出されると、タイマが再起動し、測定開始エッジ待ち状態になりますが、測定終了フラグ (EDIR) も設定されます。
- 連続パルス幅測定モード時に測定終了エッジが検出されると、タイマが再起動し、測定開始エッジ待ち状態になり、カウント終了フラグ (EDIR) が設定され、測定結果が PWC0/PWC1 へ転送されます。
- 動作時にタイマを再起動する場合は、割込み生成およびほかの制御用フラグビット (OVIR, EDIR) の動作に注意してください。

● 割込みについての注意

- PWC 状態制御レジスタ (PWCSH0/PWCSH1) の OVIR ビットに "1" を設定し、割込み要求を許可にすると (PWCSH0/PWCSH1: OVIE = 1), 制御は割込み処理から復帰できません。OVIR ビットは必ずクリアしてください。
- PWC 状態制御レジスタ (PWCSH0/PWCSH1) の EDIR ビットに "1" を設定し、割込み要求を許可にすると (PWCSH0/PWCSH1: EDIE = 1), 制御は割込み処理から復帰できません。OVIR ビットは必ずクリアしてください。
- PWC タイマは割込みベクタをほかのリソースと共用するので、割込みを使用する場合は、割込み処理ルーチンで割込み要因を注意深くチェックしなければなりません。また、PWC タイマで EI²OS を使用する場合は、共用リソース割込みを禁止しなければなりません。

第14章

16 ビット PPG タイマ

16 ビット PPG タイマの起動と動作について説明します。

14.1 16 ビット PPG タイマの概要

14.2 16 ビット PPG タイマのブロックダイアグラム

14.3 16 ビット PPG タイマの端子

14.4 16 ビット PPG タイマのレジスタ

14.5 16 ビット PPG タイマの割込み

14.6 16 ビット PPG タイマの動作説明

14.7 16 ビット PPG タイマ使用上の注意

14.1 16 ビット PPG タイマの概要

16 ビット PPG タイマは、16 ビットダウンカウンタ、プリスケアラ、16 ビット周期設定レジスタ、16 ビットデューティ設定レジスタ、16 ビット制御レジスタおよび PPG 出力端子から構成されています。

■ 16 ビット PPG タイマ (× 3)

16 ビット PPG タイマは、16 ビットダウンカウンタ、プリスケアラ、16 ビット周期設定レジスタ、16 ビットデューティ設定レジスタ、16 ビット制御レジスタおよび PPG 出力端子から構成されています。このモジュールを使用すると、多機能タイマからのソフトウェアトリガ、または GATE 信号で同期化されたパルスを出力できます。多機能タイマについての詳細は、「第 15 章 多機能タイマ」をご参照ください。

- カウンタ動作クロックは、8 種類 (ϕ , $\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$, $\phi/32$, $\phi/64$, $\phi/128$) の中から選択できます (ϕ はマシンのクロック)。
- 割込みは、トリガまたはカウンタボローが発生した場合、もしくは PPG 立上り (通常極性) / PPG 立下り (反転極性) が発生した場合に生成されます。
- PPG 出力動作:

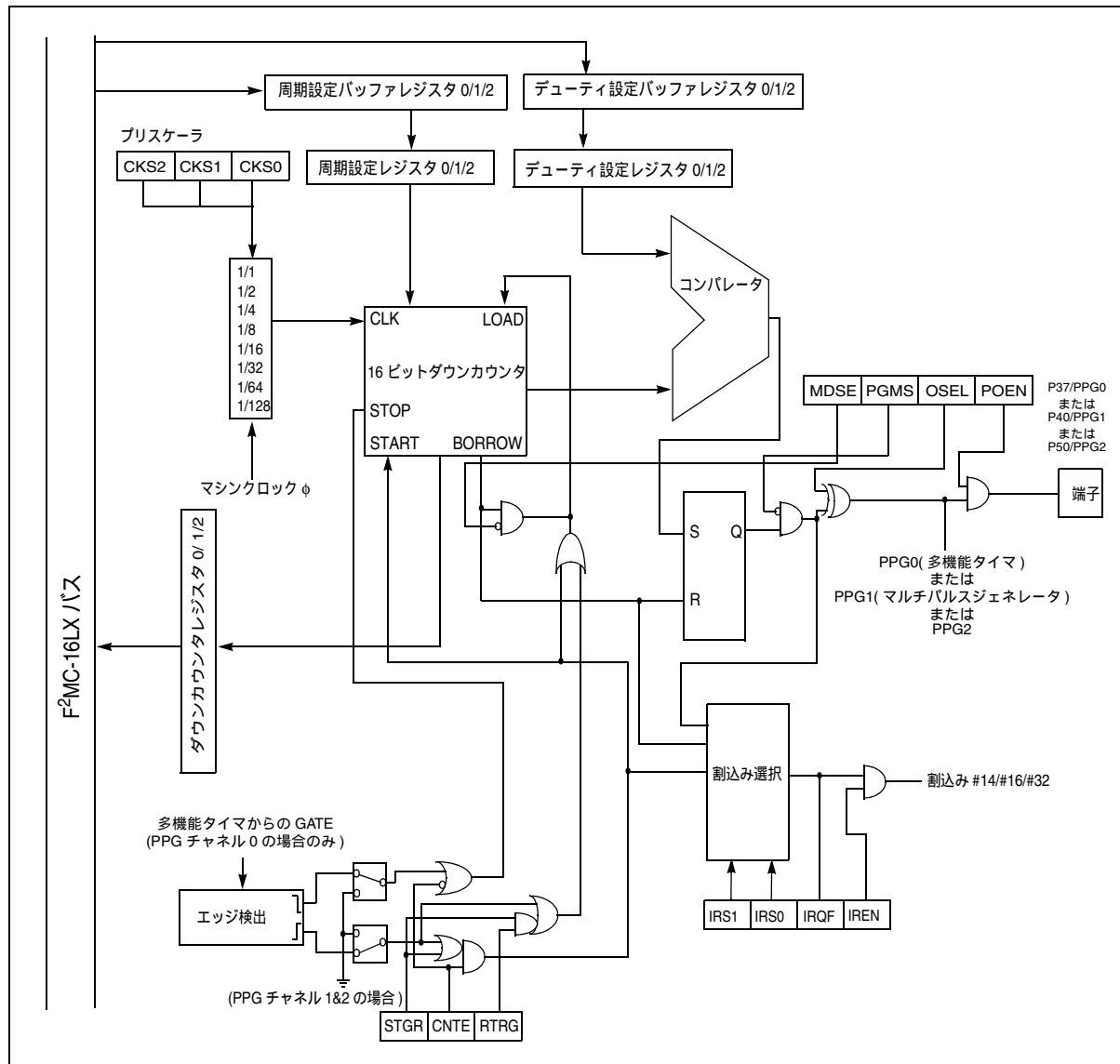
16 ビット PPG タイマは、周期およびデューティ比が可変のパルス波形を出力できます。また、16 ビット PPG タイマは、外付け回路により D/A コンバータとしても使用できます。

14.2 16 ビット PPG タイマのブロックダイアグラム

16 ビット PPG タイマのブロックダイアグラムを示します。

■ 16 ビット PPG タイマのブロックダイアグラム

図 14.2-1 16 ビット PPG タイマのブロックダイアグラム



14.3 16 ビット PPG タイマの端子

16 ビット PPG タイマの端子について説明します。また端子のブロックダイアグラムも示します。

■ 16 ビット PPG タイマの端子

16 ビット PPG タイマの端子は、汎用入出力ポートと共用されます。表 14.3-1 に端子の機能や入出力形式、端子の設定 (16 ビット PPG タイマを使用するために必要な設定) を示します。

表 14.3-1 16 ビット PPG タイマの端子

端子名	端子機能	入出力形式	プルアップ オプション	スタンバイ 制御	端子設定
P37/PPG0	ポート 3 入出力 / PPG0 出力	CMOS 出力 / CMOS 入力	選択可能	使用可能	PPG0 タイマ出力設定 (PNCTL0 : POEN = 1)
P40/PPG1	ポート 4 入出力 / PPG1 出力	CMOS 出力 /CMOS ヒステリシス入力	なし		PPG1 タイマ出力設定 (PNCTL1 : POEN = 1)
P50/PPG2	ポート 5 入出力 / PPG2 出力				PPG2 タイマ出力設定 (PNCTL2 : POEN = 1)

■ 16 ビット PPG タイマ端子のブロックダイアグラム

図 14.3-1 16 ビット PPG タイマ端子 (PPG1, PPG2) のブロックダイアグラム

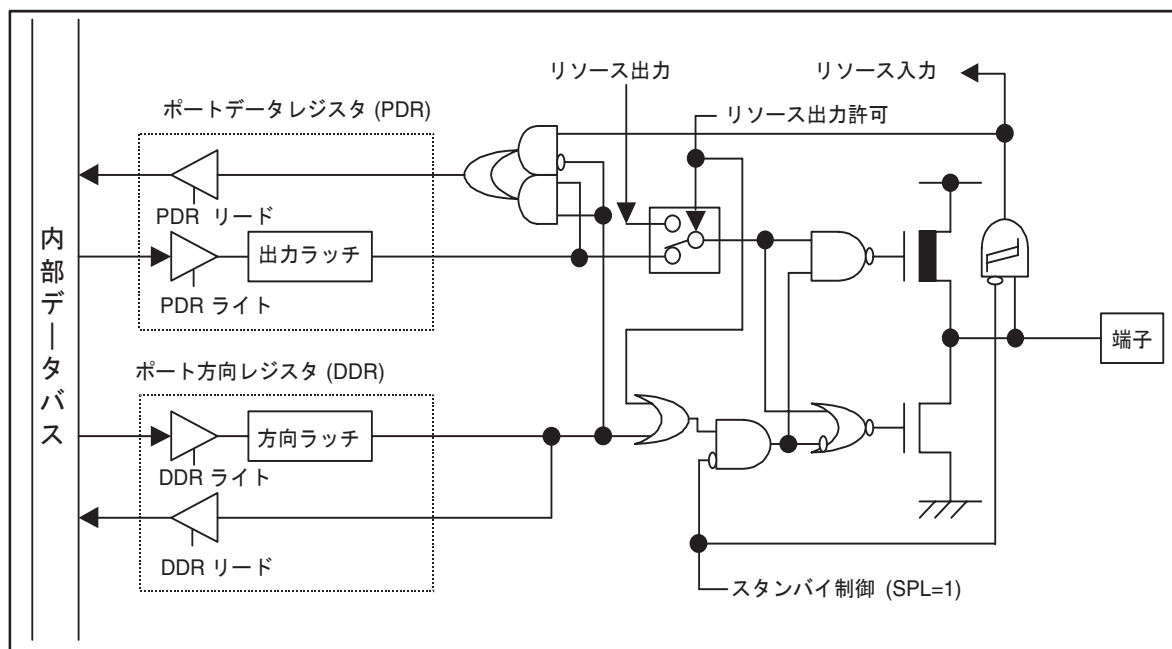
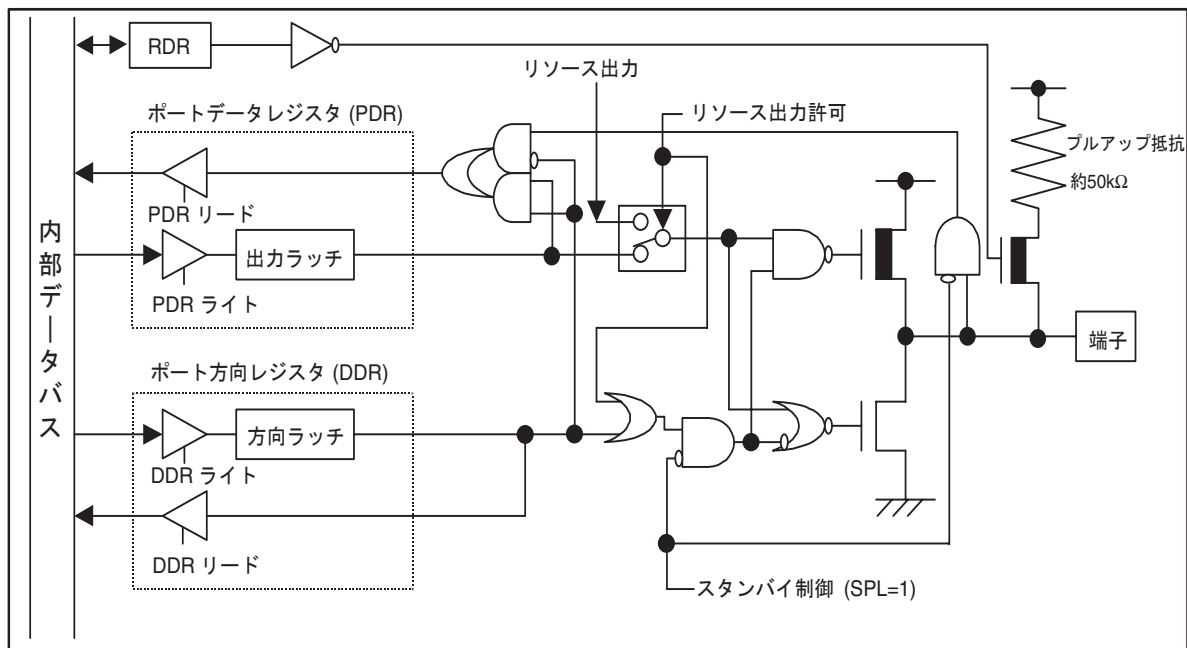


図 14.3-2 16 ビット PPG タイマ 0 端子 (PPG0) のブロックダイアグラム

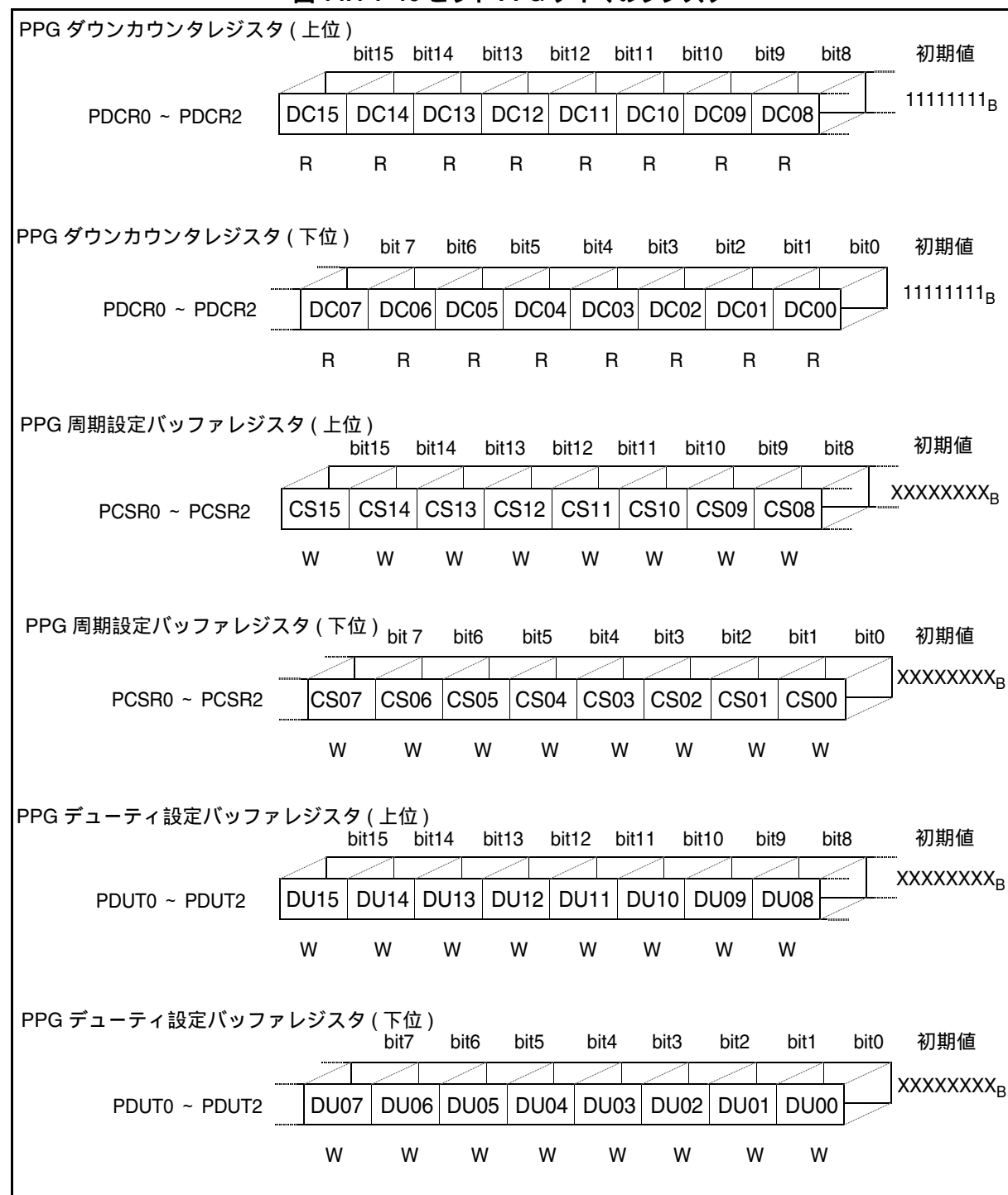


14.4 16 ビット PPG タイマのレジスタ

16 ビット PPG タイマのレジスタ一覧を示します。

■ 16 ビット PPG タイマのレジスタ

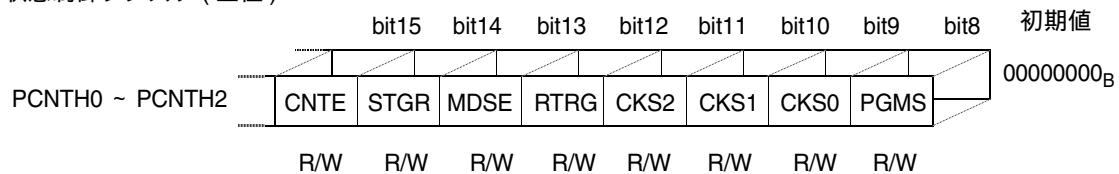
図 14.4-1 16 ビット PPG タイマのレジスタ



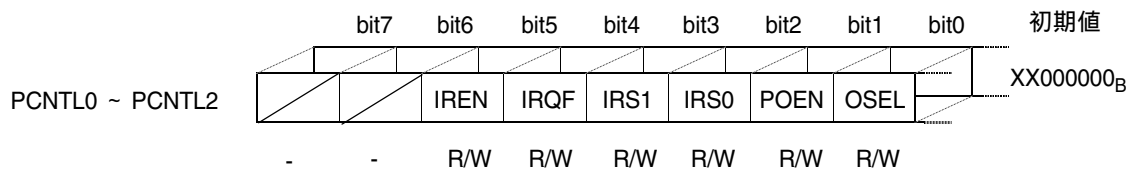
(続く)

(続き)

PPG 状態制御レジスタ (上位)



PPG 状態制御レジスタ (下位)

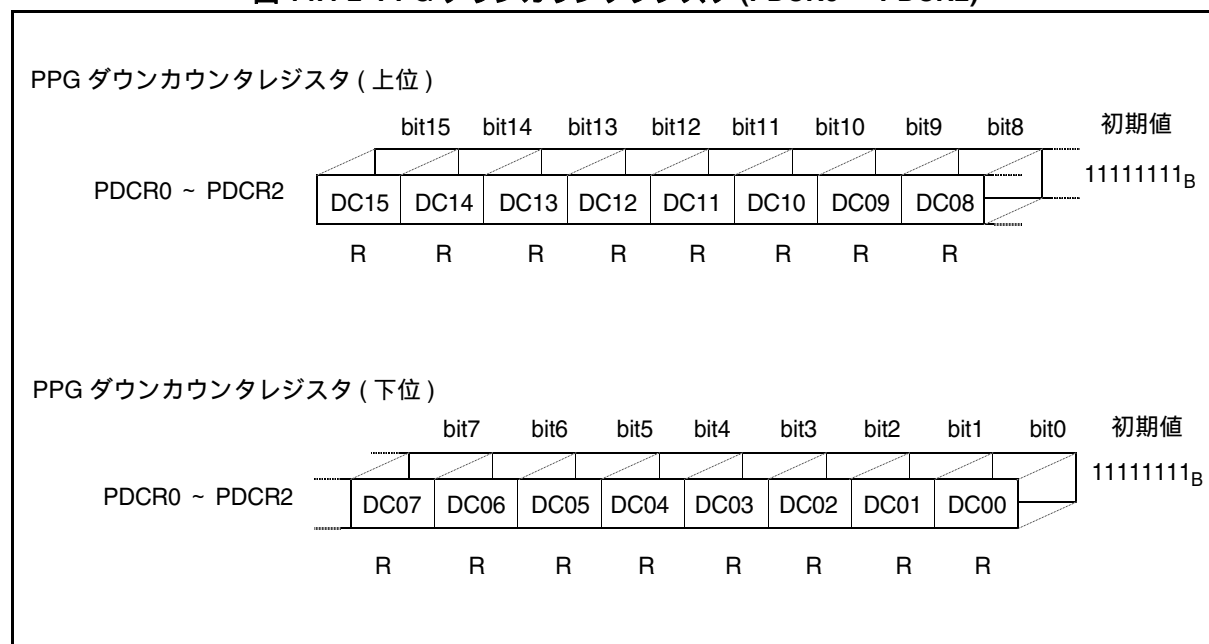


14.4.1 PPG ダウンカウンタレジスタ (PDCR0 ~ PDCR2)

PPG ダウンカウンタレジスタ (PDCR0 ~ PDCR2) は、16 ビット PPG ダウンカウンタのカウンタ値を読み出すために使用する 16 ビットレジスタです。

■ PPG ダウンカウンタレジスタ (PDCR0 ~ PDCR2)

図 14.4-2 PPG ダウンカウンタレジスタ (PDCR0 ~ PDCR2)

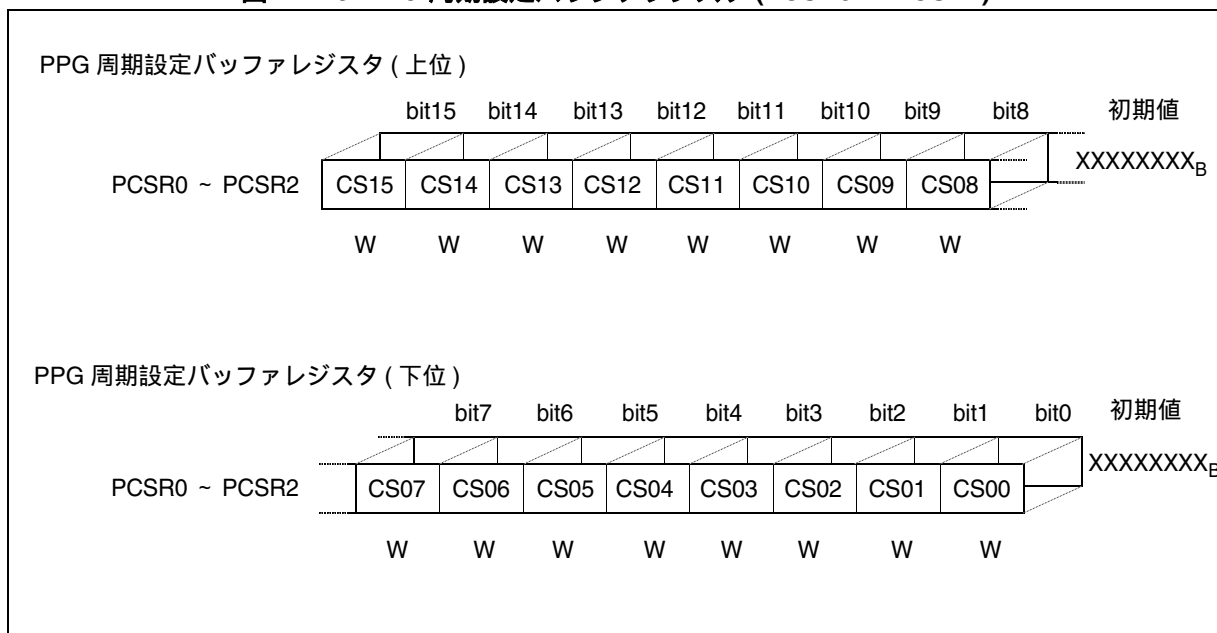


これらのレジスタは、16 ビットダウンカウンタの値を格納するために使用する 16 ビットレジスタです。これらのレジスタの初期値は、すべて "1" です。これらのレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。これらのレジスタは読み出し専用レジスタです。

14.4.2 PPG 周期設定バッファレジスタ (PCSR0 ~ PCSR2)

PPG 周期設定バッファレジスタは、PPG で生成される出力パルスの周期を設定するために使用します。

図 14.4-3 PPG 周期設定バッファレジスタ (PCSR0 ~ PCSR2)



これらのレジスタは、PPG で生成される出力パルスの周期を設定するために使用する 16 ビットレジスタです。これらのレジスタの初期値は不定なので、動作を開始する前にレジスタに値を設定しなければなりません。これらのレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。これらのレジスタは書込み専用レジスタです。

PPG 周期設定バッファレジスタから周期設定レジスタへのデータ転送は、カウンタローまたはトリガまたは再トリガが発生した場合に実行されます。

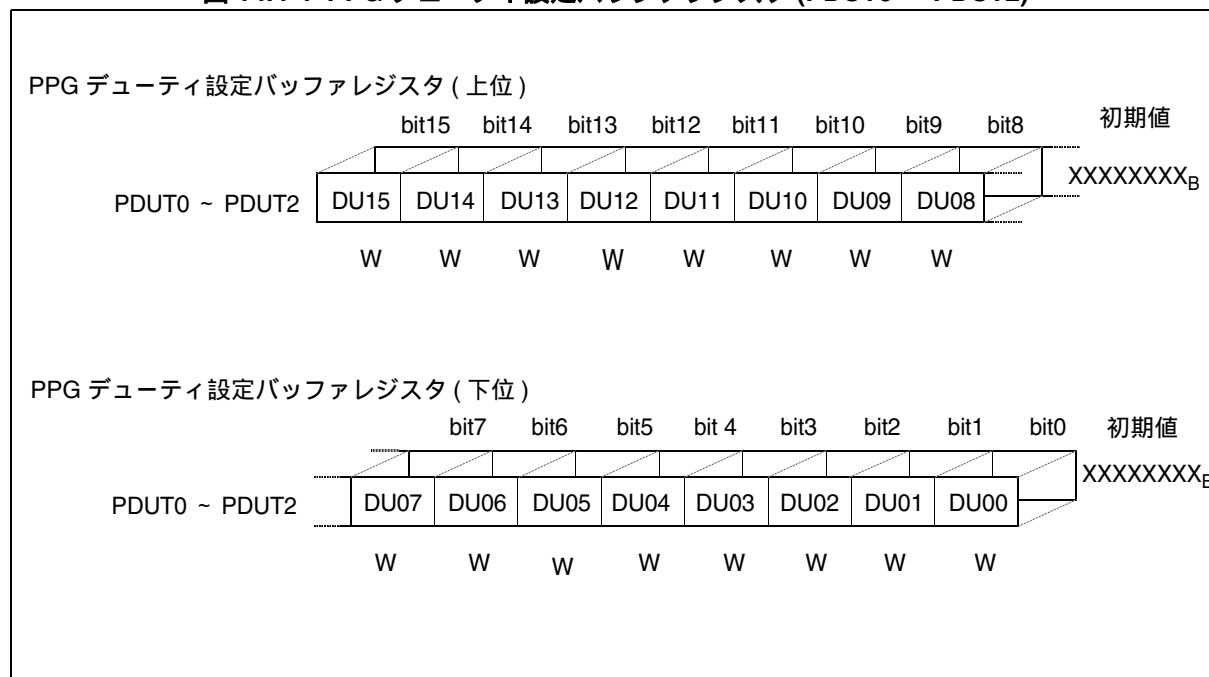
< 注意事項 >

PPG 周期設定バッファレジスタを更新する場合は、この PPG 周期設定バッファレジスタを更新した後に、更新値をデューティ設定バッファレジスタに書き込まなければなりません。すなわち、PPG 周期設定バッファレジスタのみの更新は禁止されています。

14.4.3 PPG デューティ設定バッファレジスタ (PDUT0 ~ PDUT2)

PPG デューティ設定バッファレジスタは、PPG で生成される出力パルスのデューティ比を制御するために使用します。

図 14.4-4 PPG デューティ設定バッファレジスタ (PDUT0 ~ PDUT2)



これらのレジスタは、PPG で生成される出力パルスのデューティ比を制御するために使用する 16 ビットレジスタです。これらのレジスタの初期値は不定なので、動作を開始する前にレジスタに値を設定しなければなりません。これらのレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。これらのレジスタは書き込み専用レジスタです。

PPG デューティ設定バッファレジスタからデューティ設定レジスタへのデータ転送は、カウンタゼロまたはトリガまたは再トリガが発生した場合に実行されます。

PPG 周期設定レジスタとデューティ設定レジスタに同じ値を設定すると、通常極性の場合にはすべて "H" を出力し、反転極性の場合にはすべて "L" を出力します。

PPG の出力は、"PCSR < PDUT" の場合は不定です。

< 注意事項 >

PPG 周期設定バッファレジスタを更新しない場合には、PPG デューティ設定バッファレジスタへの書き込みが可能です。

14.4.4 PPG 状態制御レジスタ (PCNTL0 ~ PCNTL2, PCNTH0 ~ PCNTH2)

PPG 状態制御レジスタは、16 ビット PPG タイマの許可/禁止、ソフトウェアトリガ、再トリガ制御割込み、および出力極性に関する動作状態を設定するために使用します。また状態チェックのためにも使用します。

■ PPG 状態制御レジスタ、上位 (PCNTH0 ~ PCNTH2)

図 14.4-5 PPG 状態制御レジスタ、上位 (PCNTH0 ~ PCNTH2)

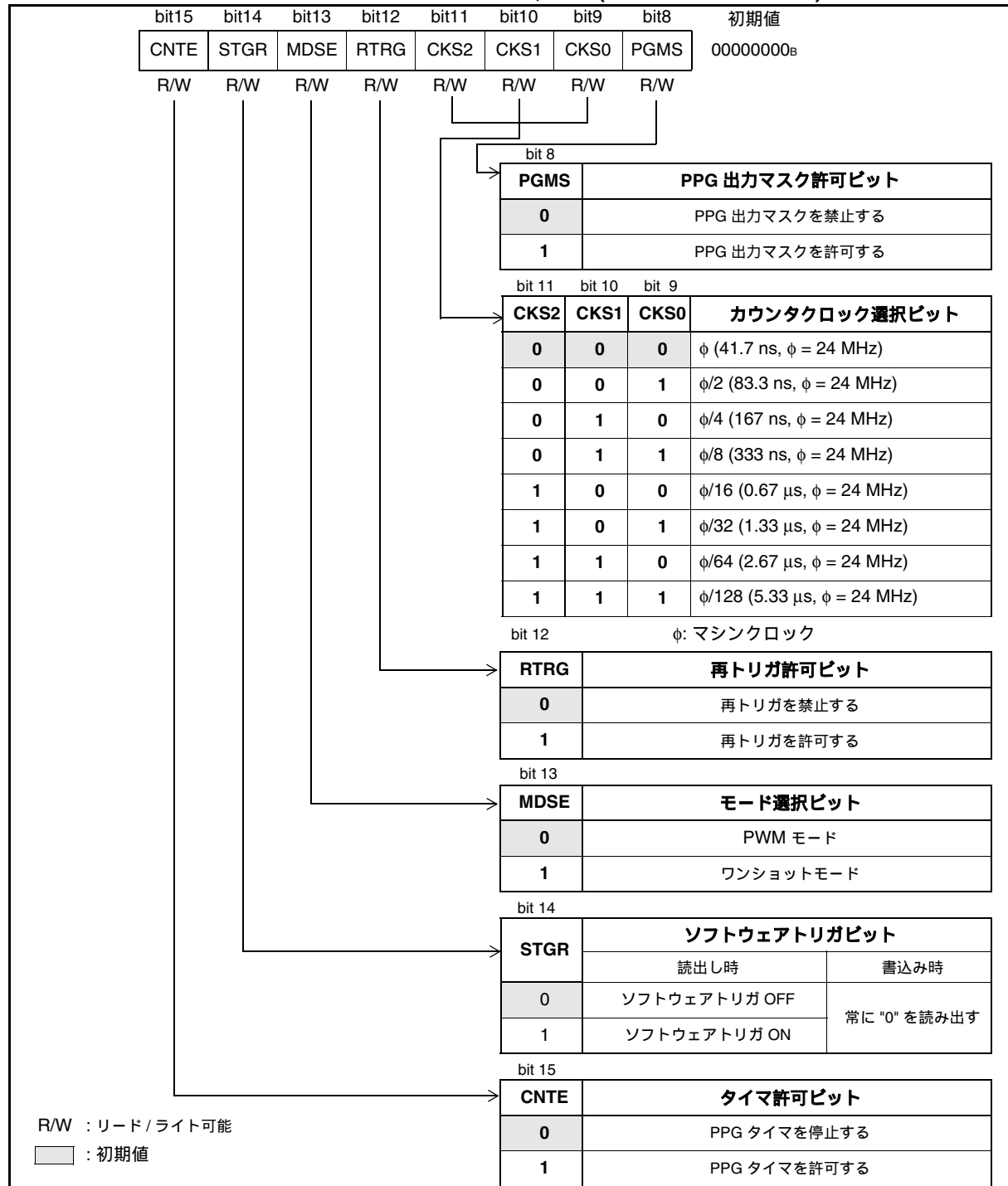


表 14.4-1 PPG 状態制御レジスタ, 上位 (PCNTH0 ~ PCNTH2) ビット

ビット名		機能
bit15	CNTE: タイマ許可ビット	<ul style="list-style-type: none"> このビットは, PPG タイマ動作を許可するために使用します。 "0" を設定した場合: 動作が停止します。 "1" を設定した場合: PPG 動作が "許可" になり, PPG 動作がトリガによって開始されるのを待ちます。
bit14	STGR: ソフトウェアトリガビット	<ul style="list-style-type: none"> このビットは, PPG をソフトウェアで起動するために使用します。 "1" を設定した場合: PPG がソフトウェアで起動します。 このビットからは必ず "0" が読み出されます。
bit13	MDSE: モード選択ビット	<ul style="list-style-type: none"> "0" を設定した場合: PPG は PWM モードで動作します。 "1" を設定した場合: PPG はワンショットモードで動作します。
bit12	RTRG: 再トリガ許可ビット	<ul style="list-style-type: none"> このビットは, 動作中に PPG の再トリガ機能を許可するために使用します。 "0" を設定した場合: 再トリガ機能を禁止します。 "1" を設定した場合: 再トリガ機能を許可します。
bit11, bit10, bit9	CKS2, CKS1, CKS0: カウンタクロック選択ビット	<ul style="list-style-type: none"> これらのビットは, 16 ビット PPG タイマの動作クロックを選択するために使用します。
bit8	PGMS: PPG 出力マスク許可ビット	<ul style="list-style-type: none"> このビットは, モード設定 (PCNTH : MDSE), 周期設定 (PCSR) またはデューティ設定 (PDUT) とは無関係に, 特定レベルへの PPG 出力をマスクするために使用します。 "0" を設定した場合: PPG 出力マスク機能を禁止します。 "1" を設定した場合: PPG 出力は, 極性設定が "通常" (PCNTL : OSEL = 0) の場合は必ず "L" がマスクされます。 "1" を設定した場合: PPG 出力は, 極性設定が "反転" (PCNTL : OSEL = 1) の場合は必ず "H" がマスクされます。 <p>(注意事項) PPG 周期設定バッファレジスタ (PCSR) と PPG デューティ設定バッファレジスタ (PDUT) に同じ値を設定すると, このビットが "1" の場合, 通常極性の場合はオール "H" を出力でき, 反転極性の場合はオール "L" を出力できます。</p>

■ PPG 状態制御レジスタ, 下位 (PCNTL1 ~ PCNTL3)

図 14.4-6 PPG 状態制御レジスタ (PCNTL1 ~ PCNTL3)

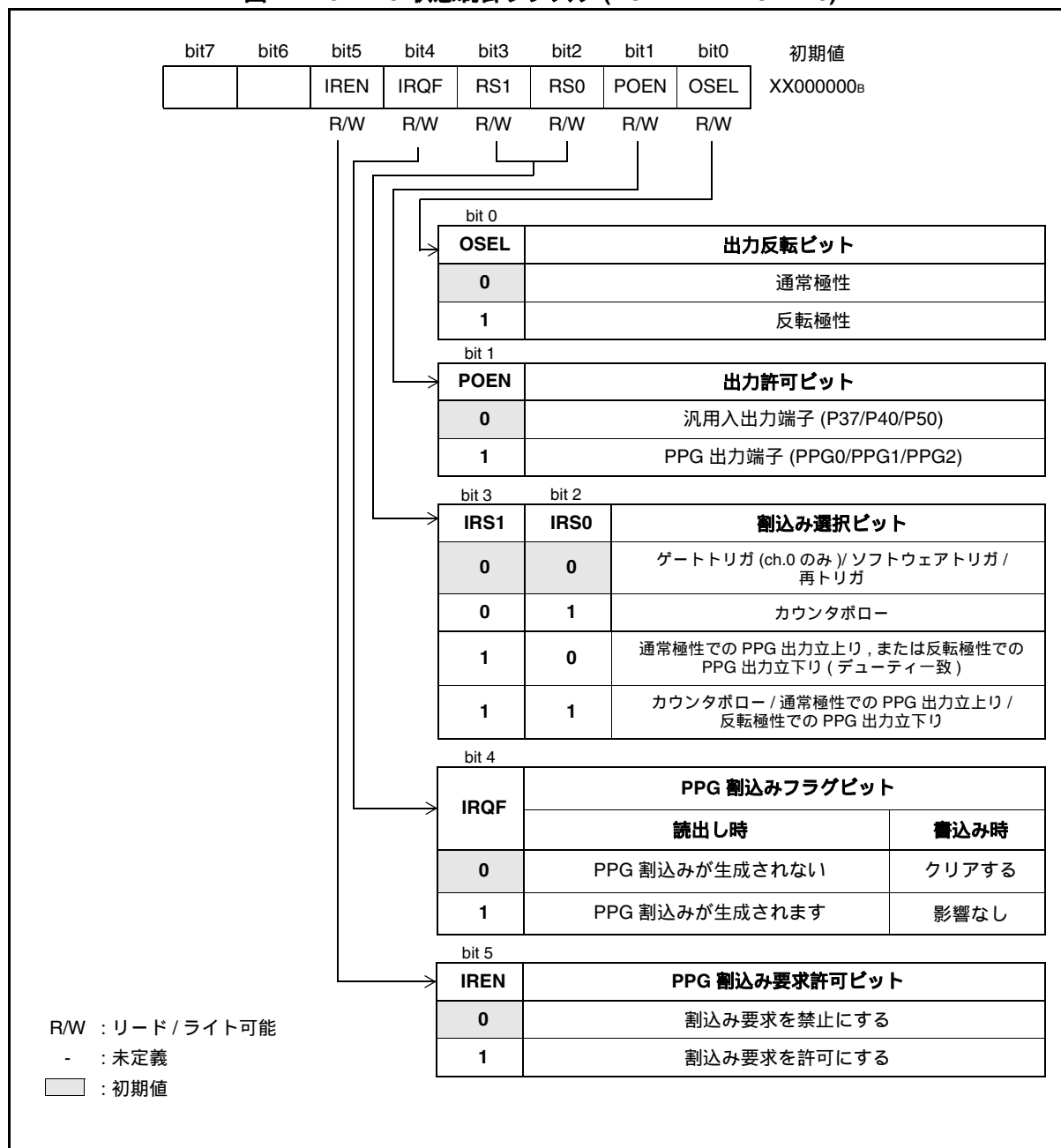


表 14.4-2 PPG 状態制御レジスタ (PCNTL1 ~ PCNTL3)

ビット名		機能
bit7, bit6	未定義ビット	読み出した場合：値は不定です。 書き込んだ場合：動作に影響しません。
bit5	IREN: PPG 割込み要求許可ビット	<ul style="list-style-type: none"> このビットは、CPU への PPG 割込み要求を許可または禁止します。 このビットと割込みフラグ (IRQF) ビットが "1" の場合、PPG は割込み要求を出力します。
bit4	IRQF: PPG 割込みフラグビット	<ul style="list-style-type: none"> このビットは、PPG 割込みが発生すると "1" に設定されます。 "0" を設定した場合：クリアされます。 "1" を設定した場合：影響を受けません。 リードモディファイライト動作時は必ず "1" が読み出されます。 このビットは、EI²OS がアクティブになった場合にもクリアされます。
bit3, bit2	IRS1, IRS0: 割込み選択ビット	<ul style="list-style-type: none"> これらのビットは、PPG タイマの割込み動作を選択するために使用します。
bit1	POEN: 出力許可ビット	<ul style="list-style-type: none"> このビットは、PPG 出力端子からの出力を許可または禁止します。 "0" を設定した場合：端子は汎用ポートとして機能します。 "1" を設定した場合：端子は PPG タイマ出力端子として機能します。
bit0	OSEL: 出力反転ビット	<ul style="list-style-type: none"> このビットは、PPG 出力端子の極性を選択します。 "0" を設定した場合：通常極性が選択されます。PPG は、16 ビットダウンカウンタ値が PDUT よりも大きい場合は "L" を出力し、16 ビットダウンカウンタ値が PDUT 以下の場合は "H" を出力します。 "1" を設定した場合：PPG 出力は反転します。

14.5 16 ビット PPG タイマの割込み

16 ビット PPG タイマは、以下の場合に割込み要求を生成できます。

- ・トリガまたはカウンタポローが発生した場合
- ・通常極性で PPG 立上りが発生した場合
- ・反転極性で PPG 立下りが発生した場合

これらの割込み動作は、PCNTL: IRS1, IRS0 で設定されます。

また 16 ビット PPG タイマは、拡張インテリジェント I/O サービス (EI²OS) とも関係しています。

■ 16 ビット PPG タイマの割込み

表 14.5-1 に 16 ビット PPG タイマの割込み制御ビットと割込み要因を示します。

表 14.5-1 16 ビット PPG タイマの割込み制御ビットと割込み要因

	16 ビット PPG タイマ 0	16 ビット PPG タイマ 1	16 ビット PPG タイマ 2
割込みフラグビット	PCNTL0: IRQF	PCNTL1: IRQF	PCNTL2: IRQF
割込み要求許可ビット	PCNTL0: IREN	PCNTL1: IREN	PCNTL2: IREN
割込み種類選択ビット	PCNTL0: IRS1, IRS0	PCNTL1: IRS1, IRS0	PCNTL2: IRS1, IRS0
割込み要因	PCNTL0: IRS1, IRS0= 00 16 ビットダウンカウンタ 0 のゲートトリガ / ソフ トウェアトリガ / 再トリガ	PCNTL1: IRS1, IRS0= 00 16 ビットダウンカウンタ 1 のソフトウェアトリガ / 再トリガ	PCNTL2: IRS1, IRS0= 00 16 ビットダウンカウンタ 2 のソフトウェアトリガ / 再トリガ
	PCNTL0: IRS1, IRS0= 01 16 ビットダウンカウンタ 0 のカウンタポロー	PCNTL1: IRS1, IRS0= 01 16 ビットダウンカウンタ 1 のカウンタポロー	PCNTL2: IRS1, IRS0= 01 16 ビットダウンカウンタ 2 のカウンタポロー
	PCNTL0: IRS1, IRS0= 10 通常極性での PPG0 出力 立上り、または反転極性 での PPG0 出力立下り	PCNTL1: IRS1, IRS0= 10 通常極性での PPG1 出力立 上り / 反転極性での PPG1 出力立下り	PCNTL2: IRS1, IRS0= 10 通常極性での PPG2 出力 立上り / 反転極性での PPG2 出力立下り
	PCNTL0: IRS1, IRS0= 11 16 ビットダウンカウンタ 0 のカウンタポロー / 通 常極性での PPG0 出力立 上り / 反転極性での PPG0 出力立下り	PCNTL1: IRS1, IRS0= 11 16 ビットダウンカウンタ 1 のカウンタポロー / 通常極 性での PPG1 出力立上り / 反転極性での PPG1 出力立 下り	PCNTL2: IRS1, IRS0= 11 16 ビットダウンカウンタ 2 のカウンタポロー / 通 常極性での PPG2 出力立 上り / 反転極性での PPG2 出力立下り

16 ビット PPG タイマでは、PPG 状態制御レジスタ (PCNTL) の IRQF ビットは "1" に設定され、割込み要求が許可されると (PCNTL: IREN = 1)、割込み要求が割込みコントローラへ出力されます。

■ 16 ビット PPG タイマ割込みと EI²OS

表 14.5-2 に 16 ビット PPG タイマ割込みと EI²OS を示します。

表 14.5-2 16 ビット PPG タイマ割込みと EI²OS

チャンネル	割込み番号	割込み制御レジスタ		ベクトルテーブルアドレス			EI ² OS
		レジスタ	アドレス	下位	中位	上位	
16 ビット PPG タイマ 0 ^{*1}	#14 (0E _H)	ICR01	0000B1 _H	FFFFC4 _H	FFFFC5 _H	FFFFC6 _H	O
16 ビット PPG タイマ 1 ^{*2}	#16 (10 _H)	ICR02	0000B2 _H	FFFFBC _H	FFFFBD _H	FFFFBE _H	
16 ビット PPG タイマ 2 ^{*3}	#32 (20 _H)	ICR10	0000BA _H	FFFF7C _H	FFFF7D _H	FFFF7E _H	

*1: 16 ビット PPG タイマ 0 に割り当てられた割込み制御レジスタと同じレジスタが PWC タイマ 0 に割り当てられます。

*2: 16 ビット PPG タイマ 1 に割り当てられた割込み制御レジスタと同じレジスタが 16 ビットアウトプットコンペアチャンネル 1 一致に割り当てられます。

*3: 16 ビット PPG タイマ 2 に割り当てられた割込み制御レジスタと同じレジスタが 16 ビットフリーランタイマ 0 検出に割り当てられます。

■ 16 ビット PPG タイマの EI²OS 機能

16 ビット PPG タイマは EI²OS と関係する回路を持っているので、カウンタは、PPG 割込みが発生すると EI²OS を起動できます。

ただし、EI²OS は割込み制御レジスタ (ICR) を共用するほかのリソースが割込みを使用しない場合に限り使用可能です。例えば、16 ビット PPG タイマ 0 が EI²OS を使用している場合、出力コンペアチャンネル 0 の一致は 禁止にしなければなりません。

14.6 16 ビット PPG タイマの動作説明

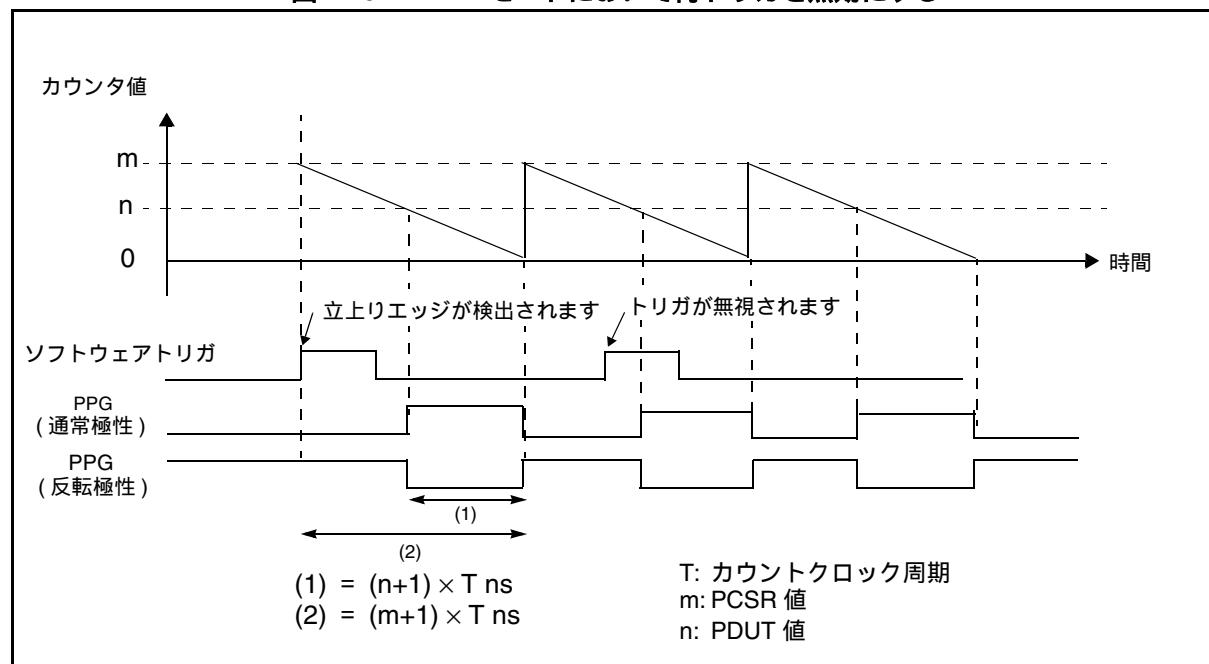
16 ビット PPG タイマは、PWM モードまたはワンショットモードで動作します。また、16 ビット PPG タイマでは再トリガを使用することもできます。

■ PWM モード (PCNTL : MDSE = 0)

PWM 動作の場合、16 ビットダウンカウンタは、PCSR 値がロードされ有効なトリガが検出された後、カウントを開始します。16 ビットダウンカウンタは "0" に達したら、PCSR 値がリロードされ、カウントを反復します。PPG 出力は、16 ビットダウンカウンタが再リロードされるとトグルされます。出力パルスの周期は PCSR レジスタを設定することで制御でき、デューティ比は PDUT レジスタを設定することで制御できます。

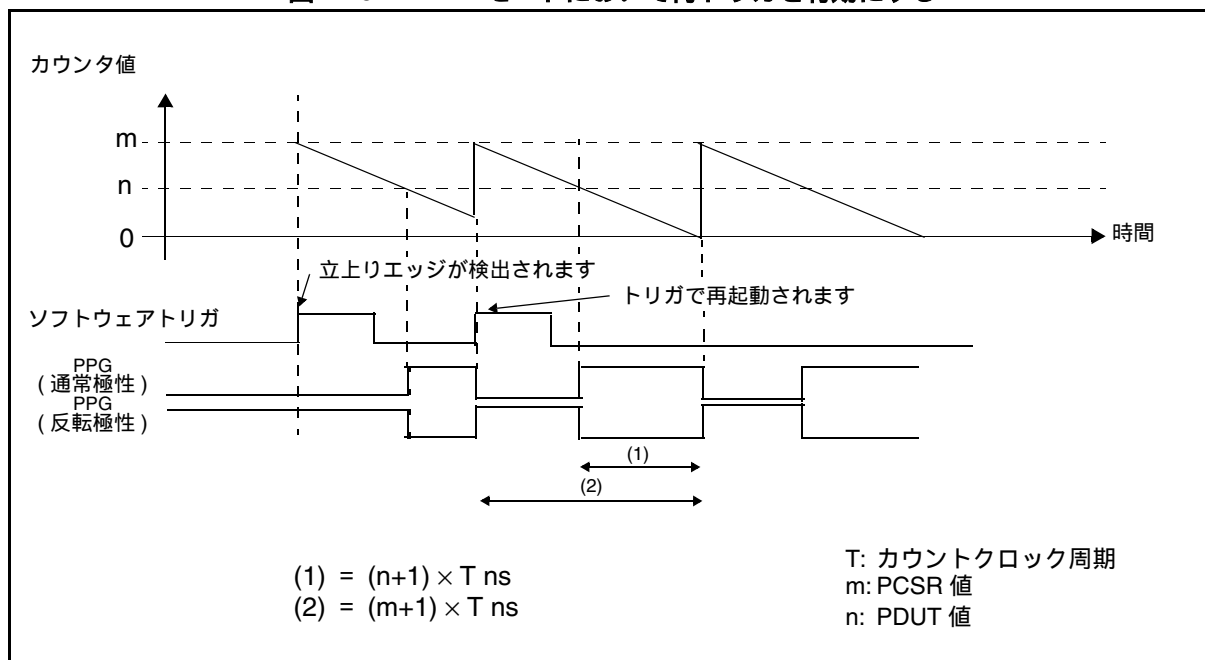
(a) 再トリガを無効にする (PCNTH : RTRG = 0)

図 14.6-1 PWM モードにおいて再トリガを無効にする



(b) 再トリガを有効にする (PCNTH : RTRG = 1)

図 14.6-2 PWM モードにおいて再トリガを有効にする

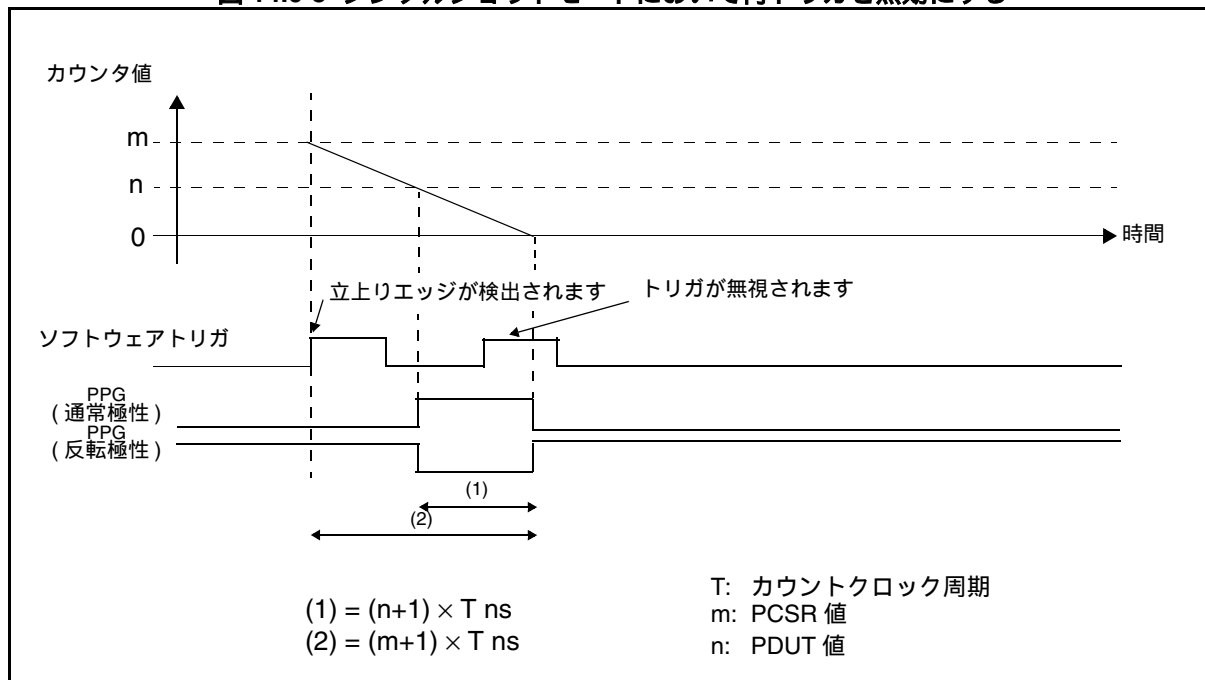


■ ワンショットモード (PCNTL : MDSE = 1)

ワンショット動作の場合、指定された幅の単一パルスを有効なトリガで出力できます。再トリガを有効にすると、動作中にエッジが検出された場合、カウンタは再リロードされます。

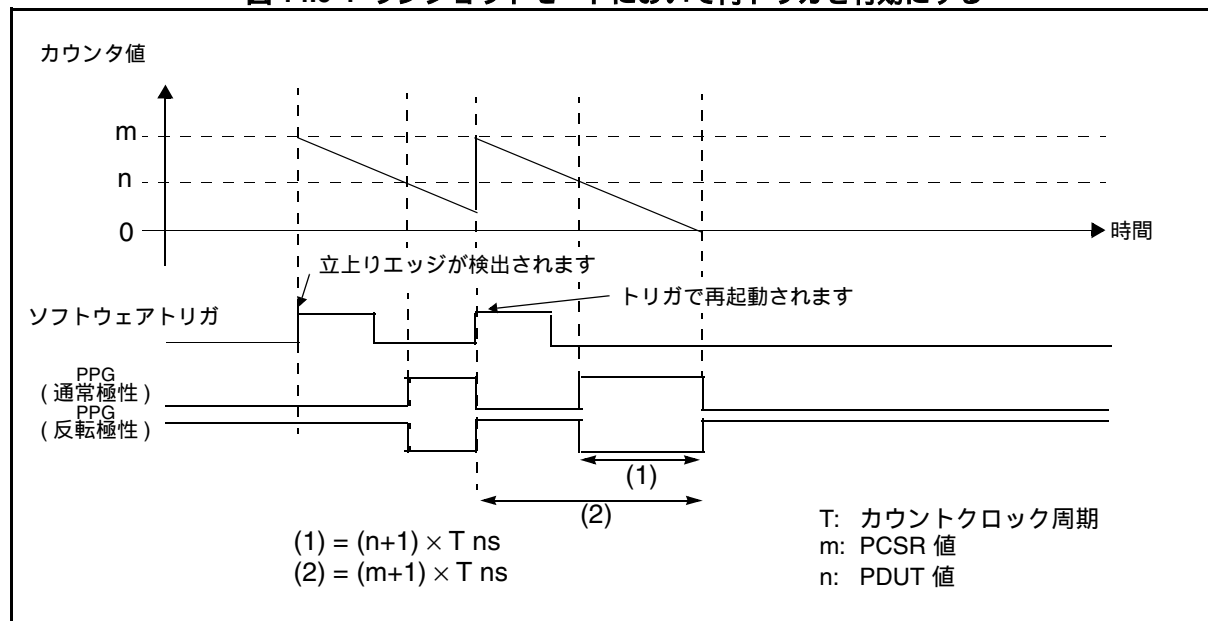
(a) 再トリガを無効にする (PCNTH : RTRG = 0)

図 14.6-3 シングルショットモードにおいて再トリガを無効にする



(b) 再トリガを有効にする (PCNTH : RTRG = 1)

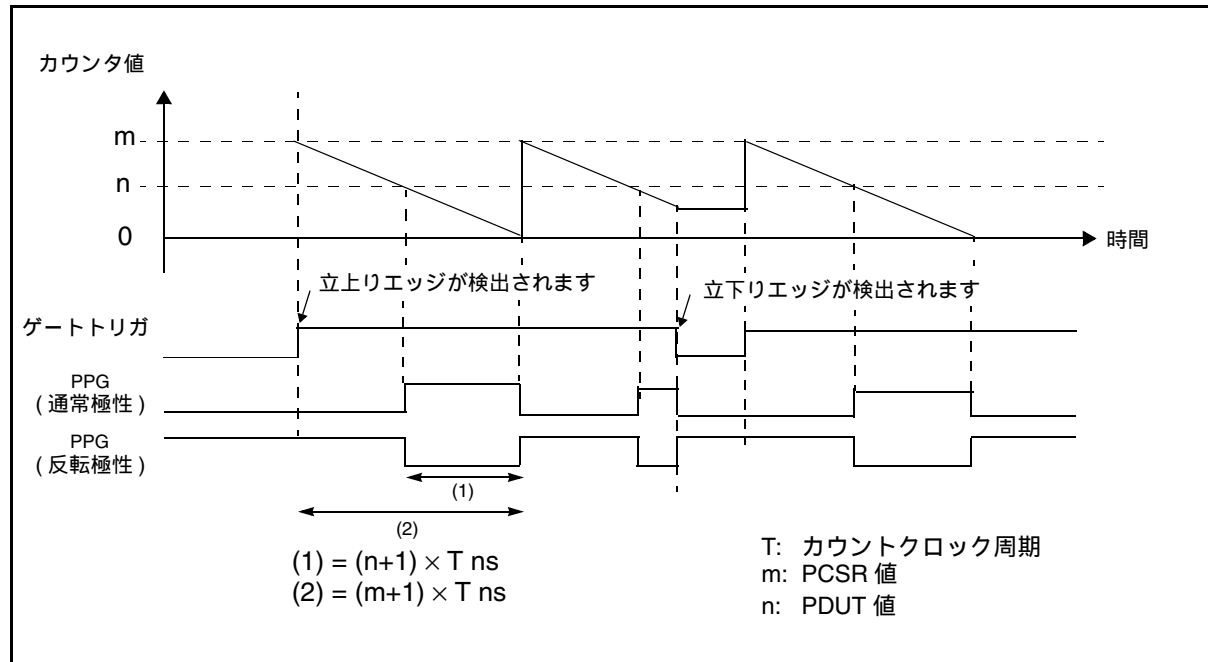
図 14.6-4 ワンショットモードにおいて再トリガを有効にする



■ ゲートトリガ (PPG チャンネル 0 の場合のみ)

ゲートトリガを使用した場合、PPG はゲートトリガの立上りエッジが検出されると動作を開始し、ゲートトリガの立下りエッジが検出されると動作を停止します。PPG は、その次の立上りエッジで動作を再開します。

図 14.6-5 再トリガを有効にした場合の PWM モードにおけるゲートトリガ

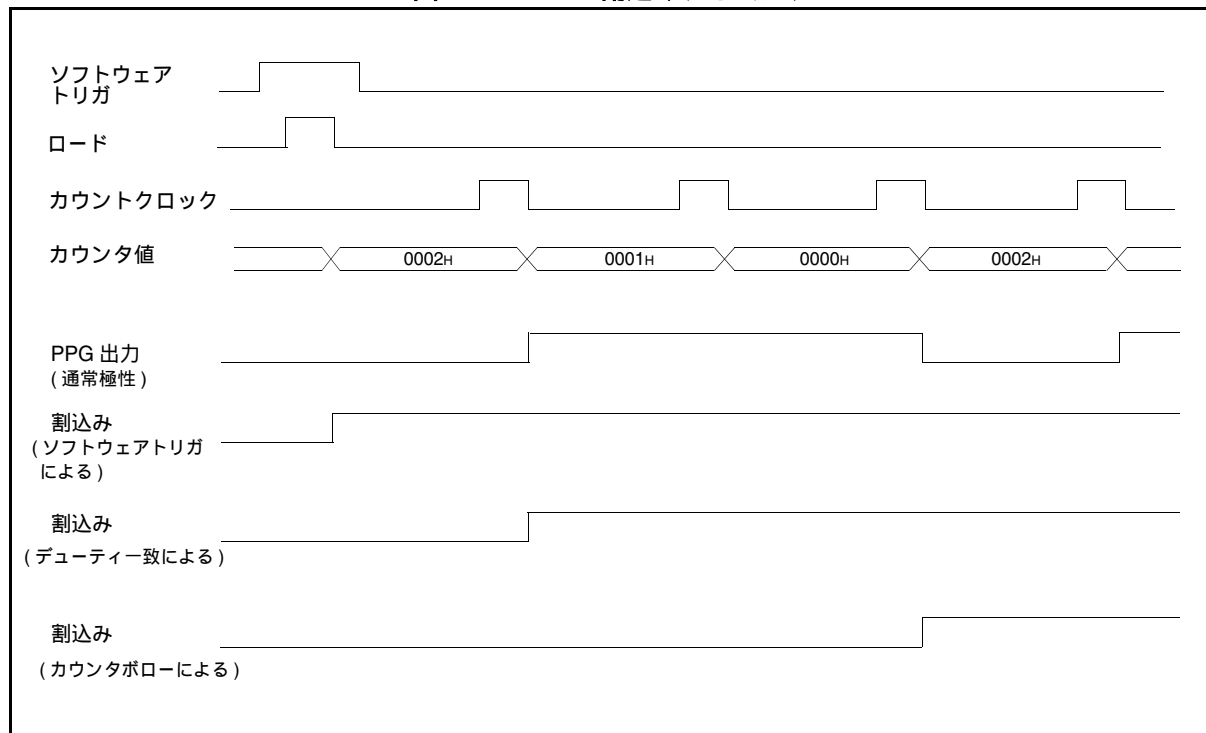


■ PPG 割込み

割込み種類選択ビット(PCNTL: IRS1, IRS0)で選択された1つの割込みフラグ(PCNTL: IRQF)を共用する割込みとしては、以下の4種類があります。

- ゲートトリガ (PPG ch.0 のみ)、ソフトウェアトリガまたは再トリガ
- カウンタボロー
- デューティー致 (通常極性において PPG 出力立上りが発生した場合、または反転極性において PPG 出力立下りが発生した場合)
- カウンタボローまたはデューティー致

図 14.6-6 PPG 割込みタイミング



14.7 16 ビット PPG タイマ使用上の注意

16 ビット PPG タイマを使用上の注意を以下に示します。

■ 16 ビット PPG タイマ使用上の注意

● プログラム設定上の注意

- PPG 周期設定バッファレジスタ (PCSR) を更新する場合は、この PPG 周期設定バッファレジスタを更新した後、この更新値を PPG デューティ設定バッファレジスタ (PDUT) に書き込まなければなりません。すなわち、PCSR レジスタのみの更新は禁止されています。ワード転送命令 (MOVW A, dir など) で PCSR と PDUT へアクセスしてください。
- PPG デューティ設定バッファレジスタ (PDUT) へは、必ず PPG 周期設定バッファレジスタ (PCSR) 以下の値を設定してください。PCSR の値よりも大きい値を設定すると、PPG 出力は不定になります。
- PPG が停止したとき (PCNTH : CNTE = 0) は、PPG 状態制御レジスタ (PCNTH) の CKS2, CKS1, CKS0 ビットを変更してください。

● 割込みについての注意

- PPG 状態制御レジスタ (PCNTL) の IRQF ビットが "1" に設定され、割込み要求を有効にすると (PCNTL : IREN = 1), 制御は割込み処理から戻ることができません。IRQF ビットは必ずクリアしてください。
- 16 ビット PPG タイマは割込みベクタをほかのリソースと共用するので、割込みを使用する場合は、割込み処理ルーチンで割込み要因を注意深くチェックしなければなりません。

また、16 ビット PPG タイマで EI²OS を使用する場合は、共用リソース割込みを無効にしなければなりません。

第15章

多機能タイマ

多機能タイマの機能と動作について説明します。

- 15.1 多機能タイマの概要
- 15.2 多機能タイマのブロックダイアグラム
- 15.3 多機能タイマの端子
- 15.4 多機能タイマのレジスタ
- 15.5 多機能タイマ割込み
- 15.6 多機能タイマの動作
- 15.7 多機能タイマ使用上の注意

15.1 多機能タイマの概要

多機能タイマは、1 つの 16 ビットフリーランタイマ、6 つの 16 ビットアウトプットコンペア、4 つの 16 ビットインプットキャプチャ、1 チャンネルの 16 ビット PPG タイマ、および 1 つの波形ジェネレータから構成されています。この波形ジェネレータを使用すると、12 個の別々の波形を 16 ビットフリーランタイマから出力でき、また入力パルス幅と外部クロックサイクルを測定することもできます。

■ 16 ビットフリーランタイマ (× 1)

- 16 ビットフリーランタイマは、16 ビットアップ / アップダウンカウンタ、タイマ制御ステータスレジスタ、16 ビットコンペアクリアレジスタ (バッファレジスタを持っています)、およびプリスケアラから構成されています。
- 8 種類のカウンタ動作クロック (ϕ , $\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$, $\phi/32$, $\phi/64$, $\phi/128$) を選択できます (ϕ はマシクロック)。
- コンペアクリア割込みは、コンペアクリアレジスタと 16 ビットフリーランタイマが比較され、一致した場合に生成されます。ゼロ検出割込みは、16 ビットフリーランタイマがカウント値 "0" を検出している間に生成されます。
- コンペアクリアレジスタは、選択可能なバッファレジスタを持っています (このバッファレジスタに書き込まれたデータはコンペアクリアレジスタへ転送されます)。16 ビットフリーランタイマが停止し、バッファにデータが書き込まれると、転送は直ちに実行されます。16 ビットフリーランタイマの動作中にタイマ値 "0" が検出されると、バッファからデータが転送されます。
- アップカウントモードにおいてリセットやソフトウェアクリア、あるいはコンペアクリアレジスタとのコンペア一致が発生すると、カウンタ値は "0000H" にリセットされます。
- このカウンタの出力値は、多機能タイマのアウトプットコンペアとインプットキャプチャのクロックカウントとして使用できます。

■ 16 ビットアウトプットコンペア (× 6)

- 16 ビットアウトプットコンペアは、6 つの 16 ビット出力コンペアレジスタ (選択可能なバッファレジスタを持っています)、コンペア出力ラッチ、コンペア制御レジスタから構成されています。16 ビットフリーランタイマ値と出力コンペアレジスタが一致すると、割込みが生成され、出力レベルが反転します。
- 6 つの出力コンペアレジスタは、別々に動作させることができます。
出力端子と割込みフラグは、各出力コンペアレジスタに対応しています。
- 2 つの出力コンペアレジスタを対 (ペア) にして出力端子を制御できます。
2 つの出力コンペアレジスタを一緒に使用することによって出力端子を反転させます。
- 各出力端子の初期値を設定できます。
- 割込みは、アウトプットコンペアレジスタが 16 ビットフリーランタイマと一致した場合に生成されます。

■ 16 ビットインプットキャプチャ (× 4)

インプットキャプチャは、4 つの独立した外部入力端子と、この端子に対応する入力キャプチャデータレジスタおよび入力キャプチャ制御レジスタから構成されています。外部端子において入力信号のエッジを検出すると、16 ビットフリーランタイムの値をキャプチャレジスタへ格納でき、また割込みも同時に生成されます。

- 外部入力信号の 3 種類のトリガエッジ (立上りエッジ、立下りエッジ、およびその両方のエッジ) を選択でき、またトリガエッジが立上りエッジであるか立下りエッジであるかを示すビットを持っています。
- 4 つの入力キャプチャを別々に動作させることができます。
- 割込みは、外部入力からの有効なエッジが検出されると生成されます。
- ch.0 と ch.1 は、割込み #33 を共用します。
- ch.2 と ch.3 は、割込み #35 を共用します。

■ 16 ビット PPG タイマ (× 1)

16 ビット PPG タイマ 0 は、波形ジェネレータへ PPG 信号を供給するために使用します。16 ビット PPG タイマ 0 の詳細については、「第 14 章 16 ビット PPG タイマ」をご参照ください。

■ 波形ジェネレータ

波形ジェネレータは、3 つの 16 ビットタイマレジスタ、3 つのタイマ制御レジスタ、および 1 つの 16 ビット波形制御レジスタから構成されています。

波形ジェネレータは、リアルタイム出力、16 ビット PPG 波形出力、ノンオーバーラップ 3 相波形出力 (インバータ制御用)、DC チョッパ波形出力を生成できます。

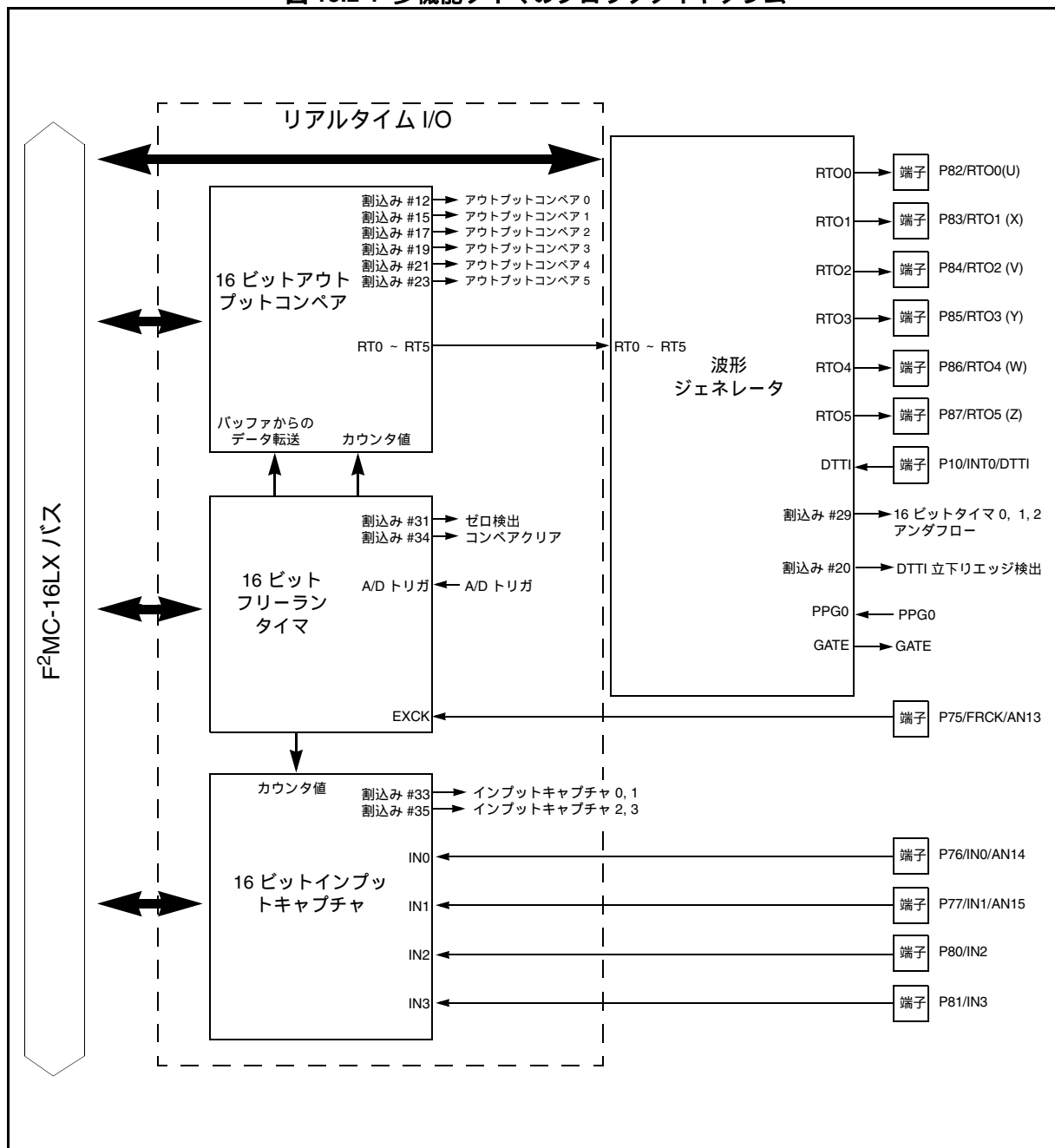
- 16 ビットタイマのデッドタイムに基づいてノンオーバーラップ波形出力を生成できます (デッドタイムタイマ機能)。
- 2 チャンネルモード時にリアルタイムアウトプットを動作させることにより、ノンオーバーラップ波形出力を生成できます (デッドタイムタイマ機能)。
- リアルタイムアウトプットコンペア一致を検出すると、GATE 信号が生成され、この信号により PPG タイマの動作が開始または停止します (GATE 機能)。
- リアルタイムアウトプットコンペア一致が検出されると、16 ビットタイマが起動になり、PPG 動作の制御用 GATE 信号を生成することによって、PPG タイマを容易に開始または停止させることができます (GATE 機能)。
- DTTI 端子入力を使用することによって強制的に停止を制御できます。

15.2 多機能タイマのブロックダイアグラム

多機能タイマのブロックダイアグラムを示します。

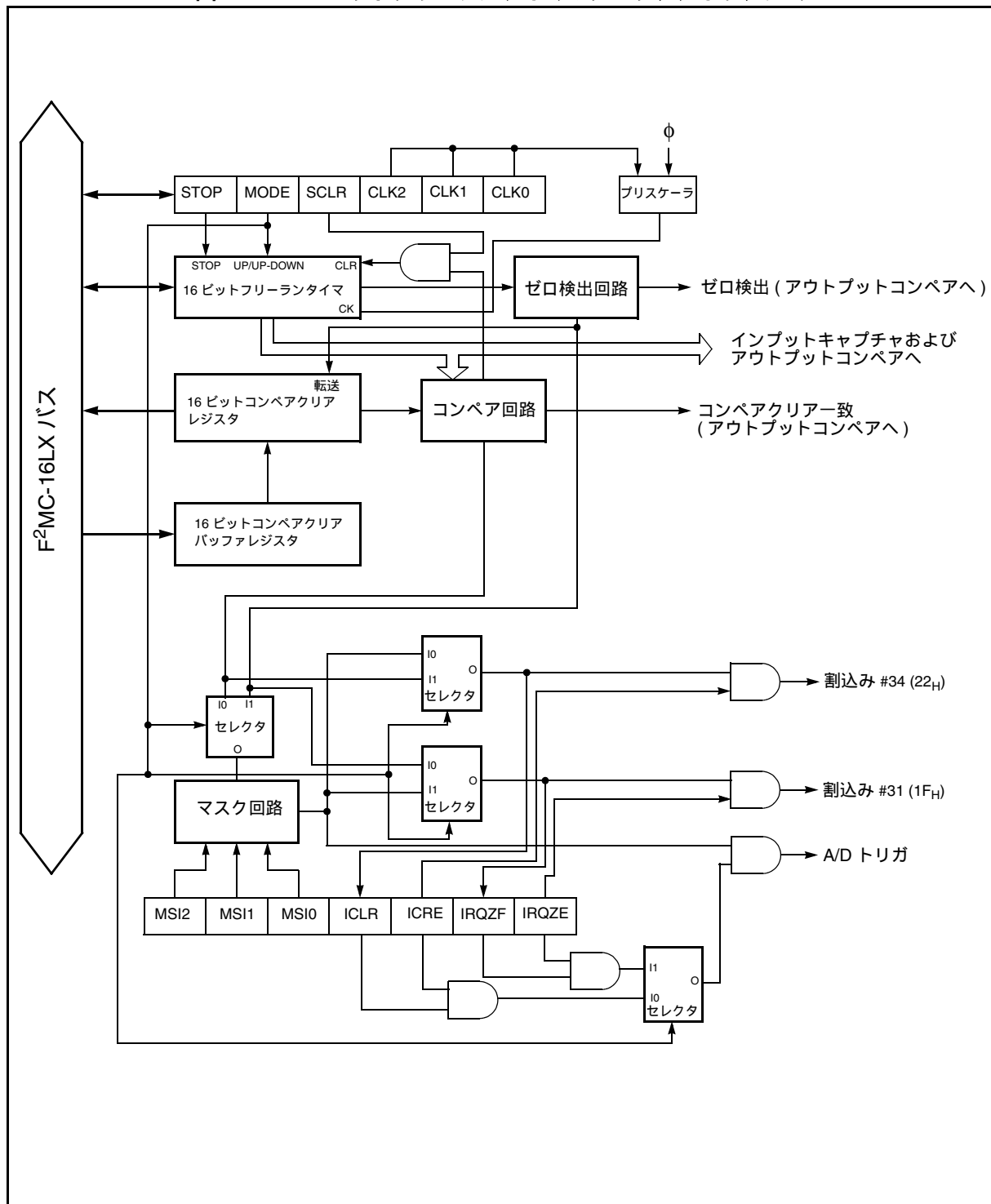
■ 多機能タイマのブロックダイアグラム

図 15.2-1 多機能タイマのブロックダイアグラム



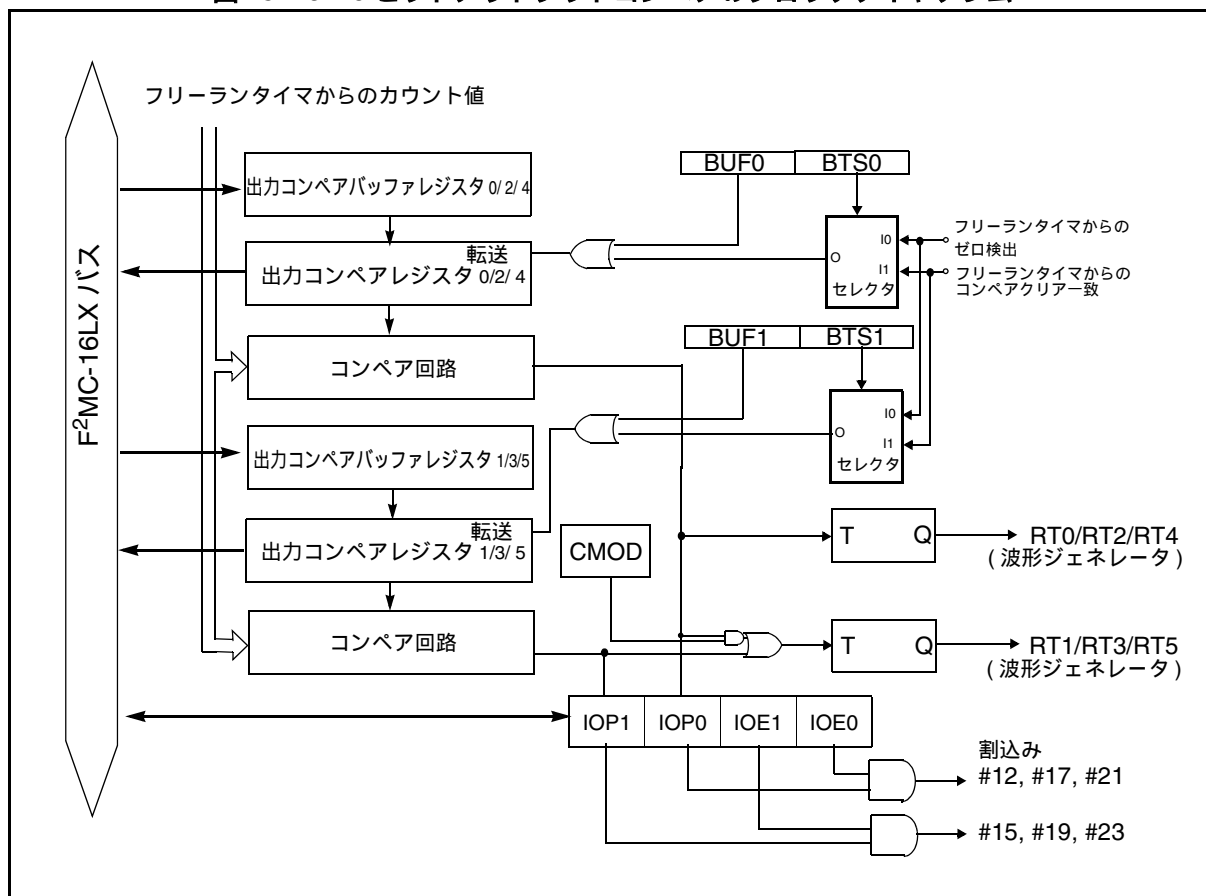
■ 16 ビットフリーランタイムのブロックダイアグラム

図 15.2-2 16 ビットフリーランタイムのブロックダイアグラム



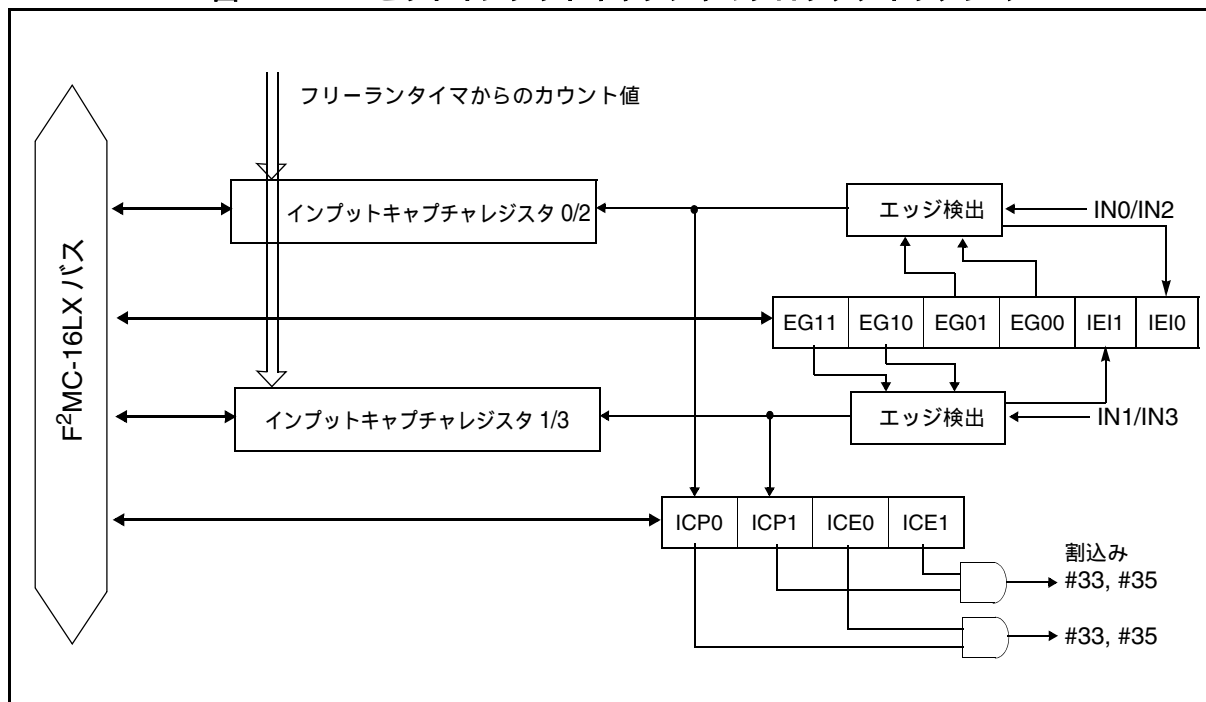
■ 16 ビットアウトプットコンペアのブロックダイアグラム

図 15.2-3 16 ビットアウトプットコンペアのブロックダイアグラム



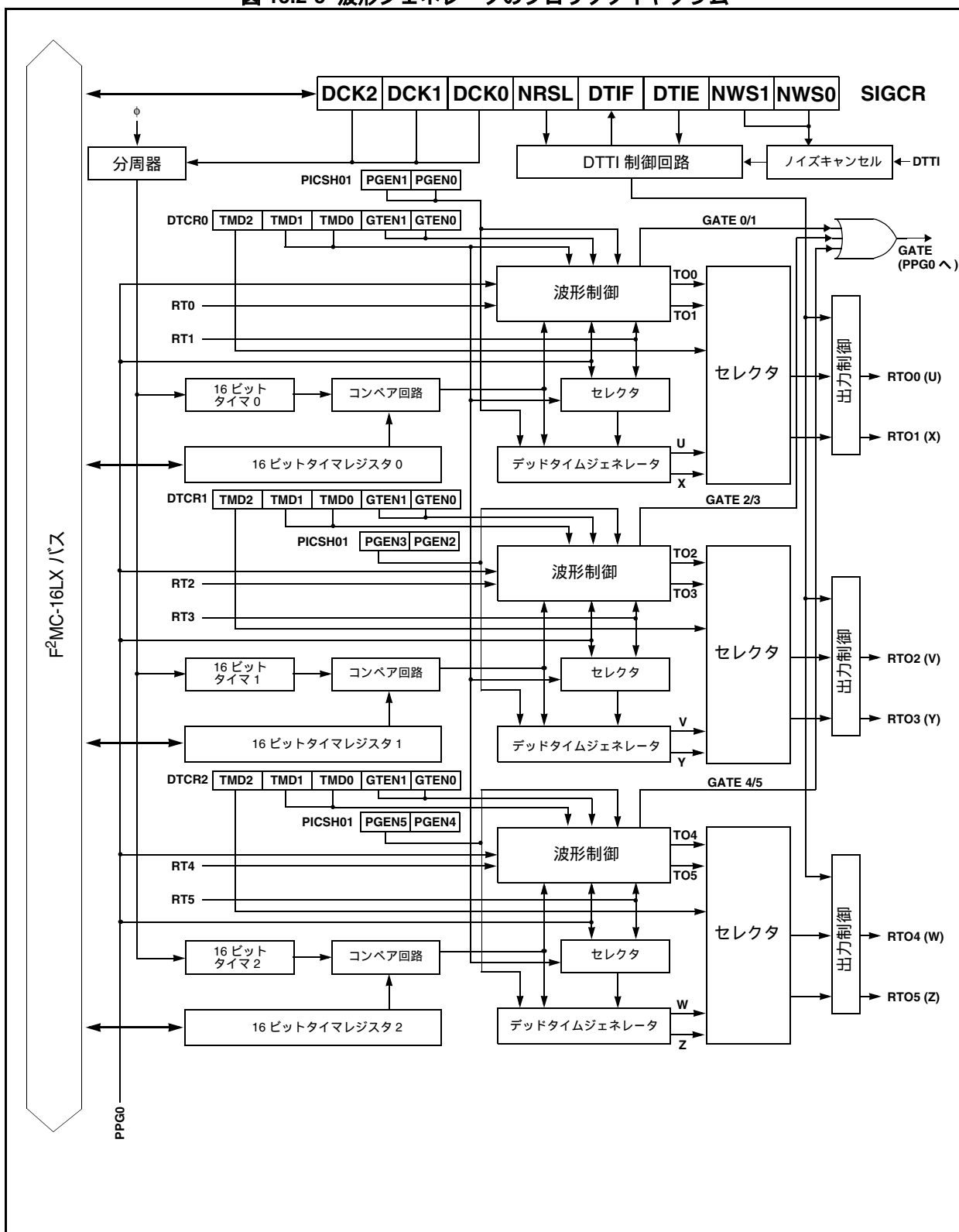
■ 16 ビットインプットキャプチャのブロックダイアグラム

図 15.2-4 16 ビットインプットキャプチャのブロックダイアグラム



■ 波形ジェネレータのブロックダイアグラム

図 15.2-5 波形ジェネレータのブロックダイアグラム



15.3 多機能タイマの端子

多機能タイマの端子について説明します。また端子のブロックダイアグラムも示します。

■ 多機能タイマの端子

表 15.3-1 多機能タイマの端子

端子名	端子機能	入出力形式	プルアップオプション	スタンバイ制御	端子設定
P10/INT0/DTTI	ポート 1 入出力 / 外部割込み入力 / DTTI	CMOS 出力 / CMOS ヒステリシス入力	選択可能	あり	端子を入力ポートとして設定する (DDR1: bit8 = 0)
P75/FRCK/AN13	ポート 7 入出力 / 外部クロック		なし		端子を入力ポートとして設定する (DDR7: bit13 = 0)
P76/IN0/AN14	ポート 7 入出力 / インプット キャプチャ 0				端子を入力ポートとして設定する (DDR7: bit14 = 0)
P77/IN1/AN15	ポート 7 入出力 / インプット キャプチャ 1				端子を入力ポートとして設定する (DDR7: bit15 = 0)
P80/IN2	ポート 8 入出力 / インプット キャプチャ 2				端子を入力ポートとして設定する (DDR8: bit0 = 0)
P81/IN3	ポート 8 入出力 / インプット キャプチャ 3				端子を入力ポートとして設定する (DDR8: bit1 = 0)
P82/RTO0 (U)	ポート 8 入出力 / RTO0				RTO0 出力を設定する (OCS1: OTE0 = 1)
P83/RTO1 (X)	ポート 8 入出力 / RTO1				RTO1 出力を設定する (OCS1: OTE1 = 1)
P84/RTO2 (V)	ポート 8 入出力 / RTO2				RTO2 出力を設定する (OCS3: OTE0 = 1)
P85/RTO3 (Y)	ポート 8 入出力 / RTO3				RTO3 出力を設定する (OCS3: OTE1 = 1)
P86/RTO4 (W)	ポート 8 入出力 / RTO4				RTO4 出力を設定する (OCS5: OTE0 = 1)
P87/RTO5 (Z)	ポート 8 入出力 / RTO5				RTO5 出力を設定する (OCS5: OTE1 = 1)

■ 多機能タイマ端子のブロックダイアグラム

図 15.3-1 P10/INT0/DTTI のブロックダイアグラム

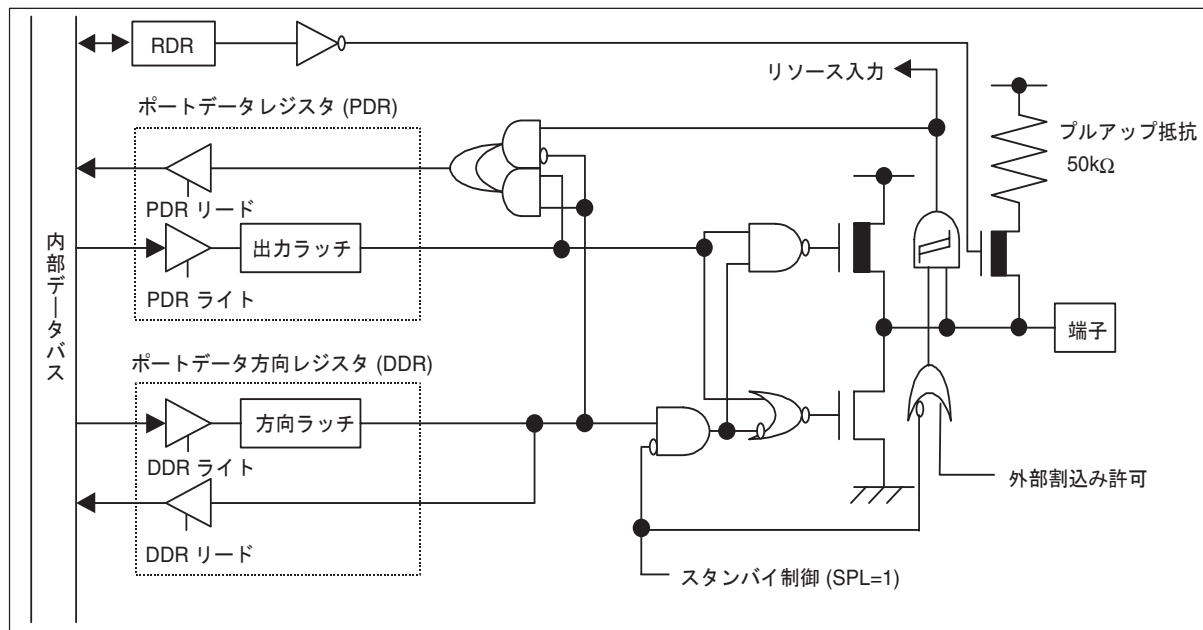


図 15.3-2 P75/FRCK/AN13 ~ P77/IN1/AN15 のブロックダイアグラム

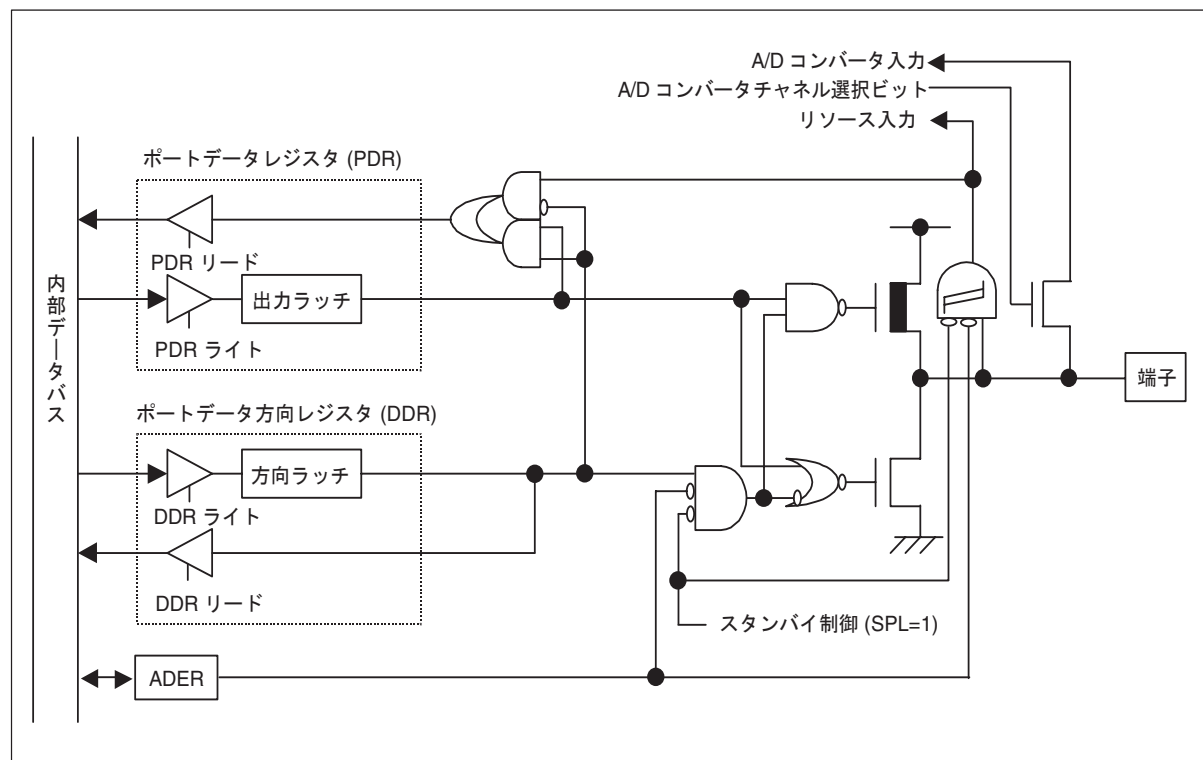


図 15.3-3 P80/IN2 および P81/IN3 のブロックダイヤグラム

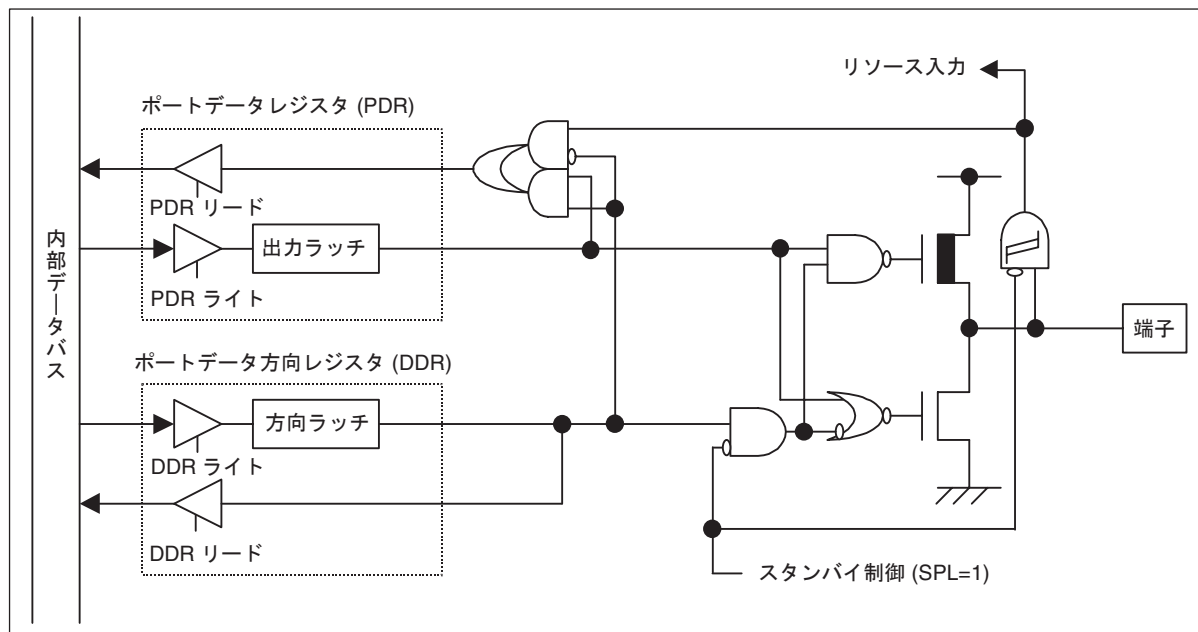
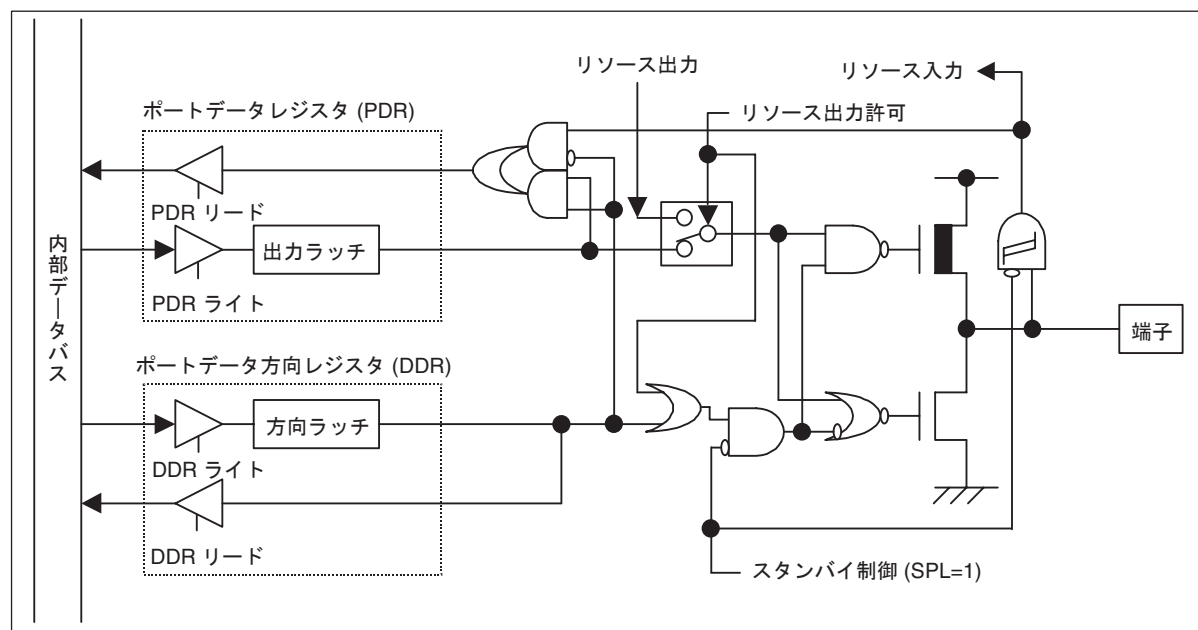


図 15.3-4 P82/RT00 ~ P87/RT05 のブロックダイヤグラム



15.4 多機能タイマのレジスタ

多機能タイマのレジスタについて説明します。

■ 16 ビットフリーランタイマのレジスタ

図 15.4-1 16 ビットフリーランタイマのレジスタ

コンペアクリアバッファレジスタ / コンペアクリアレジスタ (上位)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
CPCLRB/ CPCLR	CL15	CL14	CL13	CL12	CL11	CL10	CL09	CL08	11111111 _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

コンペアクリアバッファレジスタ / コンペアクリアレジスタ (下位)

	bit7	bit6	bit5	bit4	bit3	bit 2	bit1	bit0	初期値
CPCLRB/ CPCLR	CL07	CL06	CL05	CL04	CL03	CL02	CL01	CL00	11111111 _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

タイマデータレジスタ (上位)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
TCDT	T15	T14	T13	T12	T11	T10	T09	T08	00000000 _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

タイマデータレジスタ (下位)

	bit7	bit6	bit5	bit4	bit3	bit 2	bit1	bit0	初期値
TCDT	T07	T06	T05	T04	T03	T02	T01	T00	00000000 _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

タイマ状態制御レジスタ (上位)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
TCCSH	ECKE	IRQZF	IRQZE	MSI2	MSI1	MSI0	ICLR	ICRE	00000000 _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

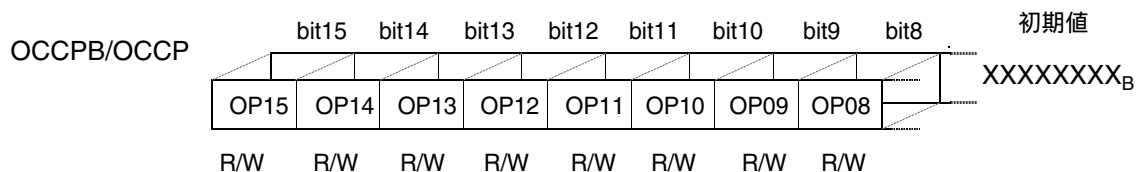
タイマ状態制御レジスタ (下位)

	bit7	bit6	bit5	bit4	bit3	bit 2	bit1	bit0	初期値
TCCSL	-	BFE	STOP	MODE	SCLR	CLK2	CLK1	CLK0	X0100000 _B
	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

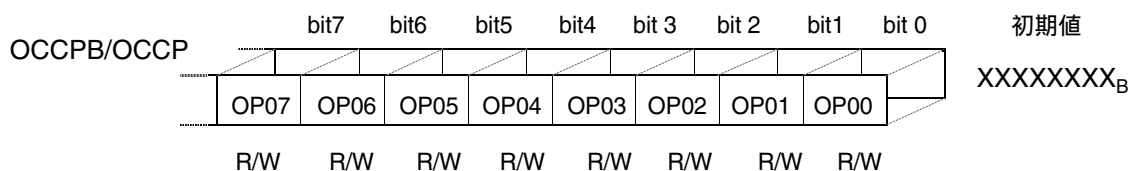
■ 16 ビットアウトプットコンペアのレジスタ

図 15.4-2 16 ビットアウトプットコンペアのレジスタ

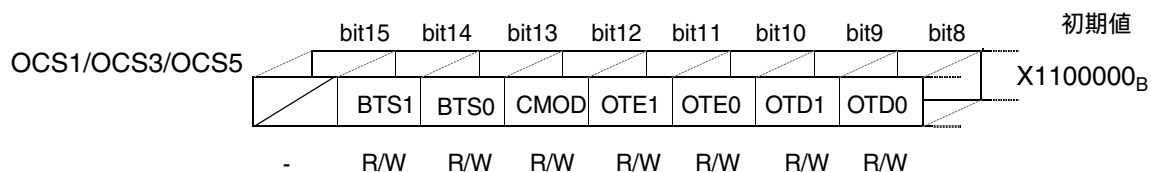
アウトプットコンペアバッファレジスタ / アウトプットコンペアレジスタ (上位)



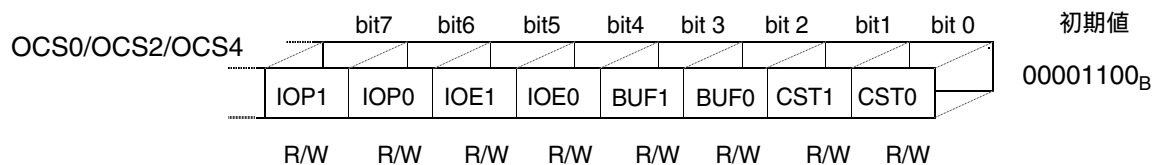
アウトプットコンペアバッファレジスタ / アウトプットコンペアレジスタ (下位)



コンペア制御レジスタ (上位)



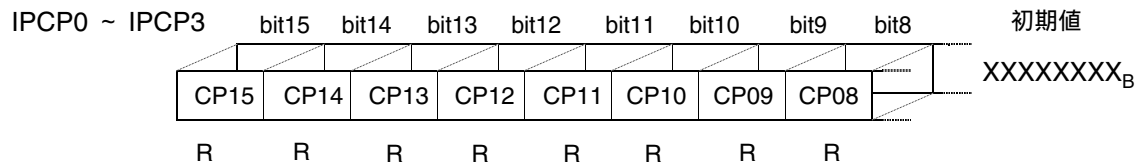
コンペア制御レジスタ (下位)



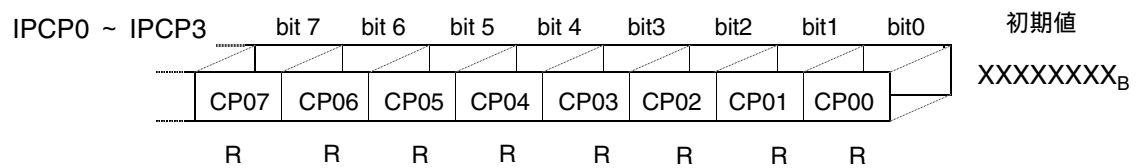
■ インพุットキャプチャのレジスタ

図 15.4-3 16 ビットインพุットキャプチャのレジスタ

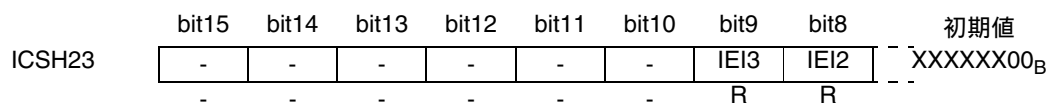
インพุットキャプチャデータレジスタ (上位)



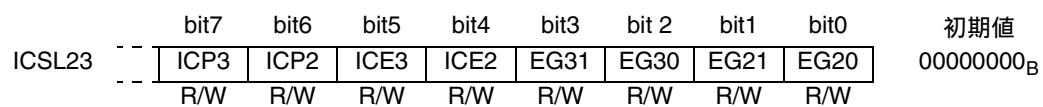
インพุットキャプチャデータレジスタ (下位)



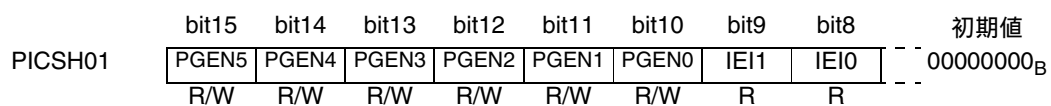
インพุットキャプチャ状態制御レジスタ (ch.2/ch.3)(上位)



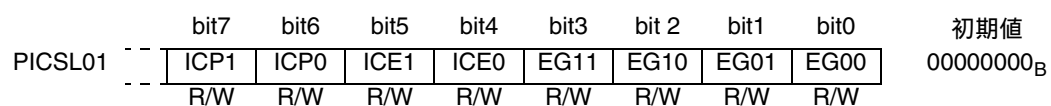
インพุットキャプチャ状態制御レジスタ (ch.2/ch.3)(下位)



PPG アウトプット制御 / インพุットキャプチャ状態制御レジスタ (ch.0/ch.1)(上位)



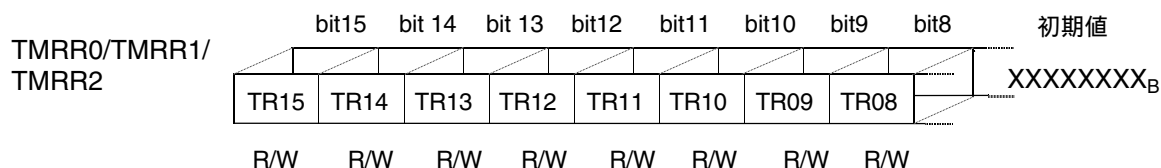
インพุットキャプチャ制御レジスタ (ch.0/ch.1)(下位)



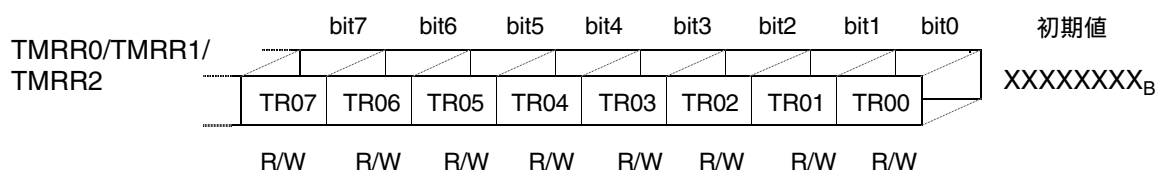
■ 波形ジェネレータのレジスタ

図 15.4-4 波形ジェネレータのレジスタ

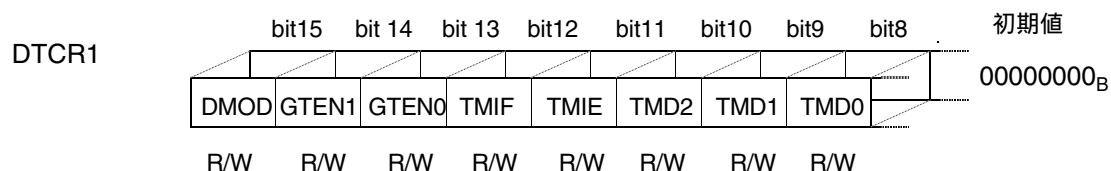
16 ビットタイマレジスタ (上位)



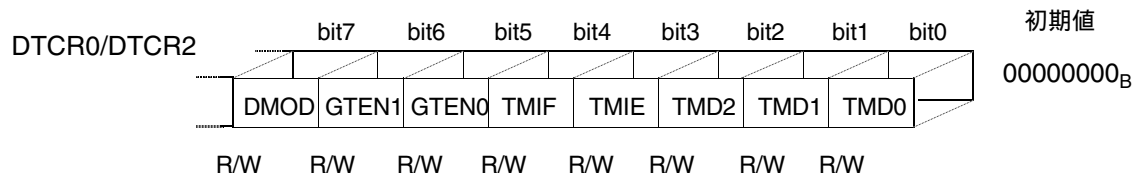
16 ビットタイマレジスタ (下位)



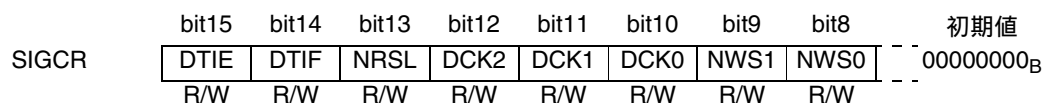
16 ビットタイマ制御レジスタ



16 ビットタイマ制御レジスタ



波形制御レジスタ



MB90820B シリーズ

15.4.1 コンペアクリアバッファレジスタ (CPCLRB) と コンペアクリアレジスタ (CPCLR)

コンペアクリアバッファレジスタ (CPCLRB) は、コンペアクリアレジスタ (CPCLR) に存在する 16 ビットバッファレジスタです。CPCLRB レジスタと CPCLR レジスタは、両方とも同じアドレスに存在します。

■ コンペアクリアバッファレジスタ (CPCLRB)

図 15.4-5 コンペアクリアバッファレジスタ (CPCLRB)

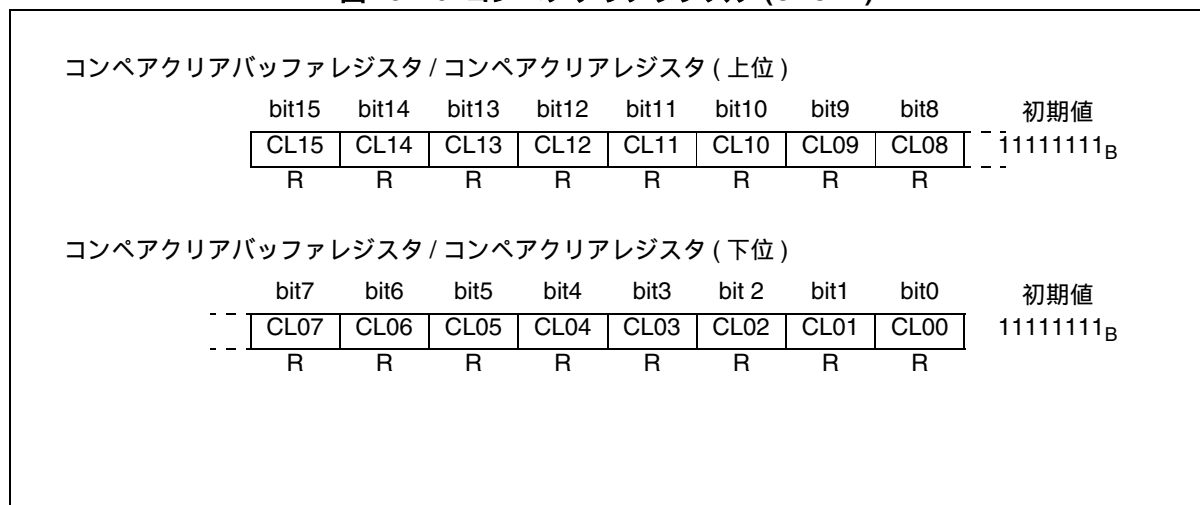
コンペアクリアバッファレジスタ / コンペアクリアレジスタ (上位)								初期値
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
CL15	CL14	CL13	CL12	CL11	CL10	CL09	CL08	11111111 _B
W	W	W	W	W	W	W	W	
コンペアクリアバッファレジスタ / コンペアクリアレジスタ (下位)								初期値
bit7	bit6	bit5	bit4	bit3	bit 2	bit1	bit0	
CL07	CL06	CL05	CL04	CL03	CL02	CL01	CL00	11111111 _B
W	W	W	W	W	W	W	W	

コンペアクリアバッファレジスタは、コンペアクリアレジスタのバッファレジスタです。バッファ機能が無効になるか (TCCSL : BFE = 0), またはフリーランタイムが停止すると、コンペアクリアバッファレジスタの値が直ちにコンペアクリアレジスタへ転送されます。バッファ機能が有効になると、16 ビットフリーランタイムのカウント値 "0" が検出されたときに値がコンペアクリアレジスタへ転送されます。

このレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。

■ コンペアクリアレジスタ (CPCLR)

図 15.4-6 コンペアクリアレジスタ (CPCLR)



コンペアクリアレジスタは、16 ビットフリーランタイムのカウント値と比較されるために使用します。アップカウントモード時は、このレジスタが 16 ビットフリーランタイムのカウント値と一致すると、16 ビットフリーランタイムは "0000_H" にリセットされます。アップダウンカウントモード時は、このレジスタが 16 ビットフリーランタイムのカウント値と一致すると、16 ビットフリーランタイムは "0" 検出時にアップカウントからダウンカウントに変わるか、またはダウンカウントからアップカウントに変わります。

このレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。

15.4.2 タイマデータレジスタ (TCDT)

タイマデータレジスタ (TCDT) は、16 ビットフリーランタイマのカウント値を読み出すために使用します。

■ タイマデータレジスタ (TCDT)

図 15.4-7 タイマデータレジスタ (TCDT)

タイマデータレジスタ (上位)								初期値
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	00000000 _B
T15	T14	T13	T12	T11	T10	T09	T08	--
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	--
タイマデータレジスタ (下位)								初期値
bit7	bit6	bit5	bit4	bit3	bit 2	bit1	bit0	00000000 _B
T07	T06	T05	T04	T03	T02	T01	T00	--
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	--

タイマデータレジスタは、16 ビットフリーランタイマのカウント値を読み出すために使用します。カウンタ値は、リセットが発生すると直ちに "0000_H" にクリアされます。タイマ値は、このレジスタへ値を書き込むことで設定できます。ただし、値の書き込みはタイマの停止中 (STOP = 1) でなければなりません。タイマデータレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。

16 ビットフリーランタイマは、以下の要因が発生すると直ちに初期化されます。

- リセット
- タイマ状態制御レジスタのクリアビット (SCLR)
- アップカウントモード (TCCSL : MODE = 0) 時におけるコンペアクリアレジスタとタイマカウンタ値の一致

15.4.3 タイマ状態制御レジスタ (TCCSH, TCCSL)

タイマ状態制御レジスタ (TCCS) は、16 ビットフリーランタイマの動作を制御するために使用する 16 ビットレジスタです。

■ タイマ状態制御レジスタ，上位バイト (TCCSH)

図 15.4-8 タイマ状態制御レジスタ，上位バイト (TCCSH)

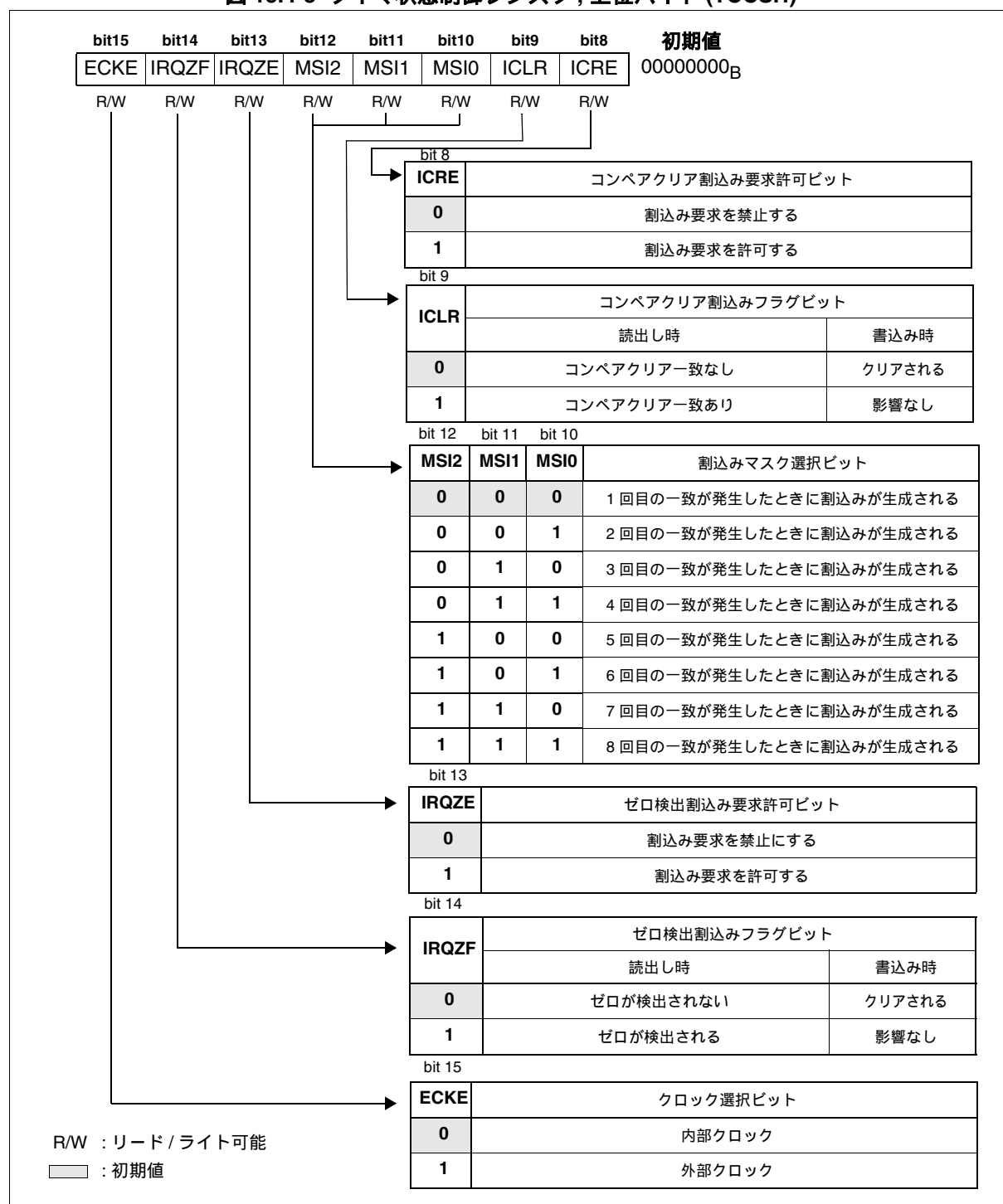


表 15.4-1 タイマ状態制御レジスタ, 上位バイト (TCCSH) (1 / 2)

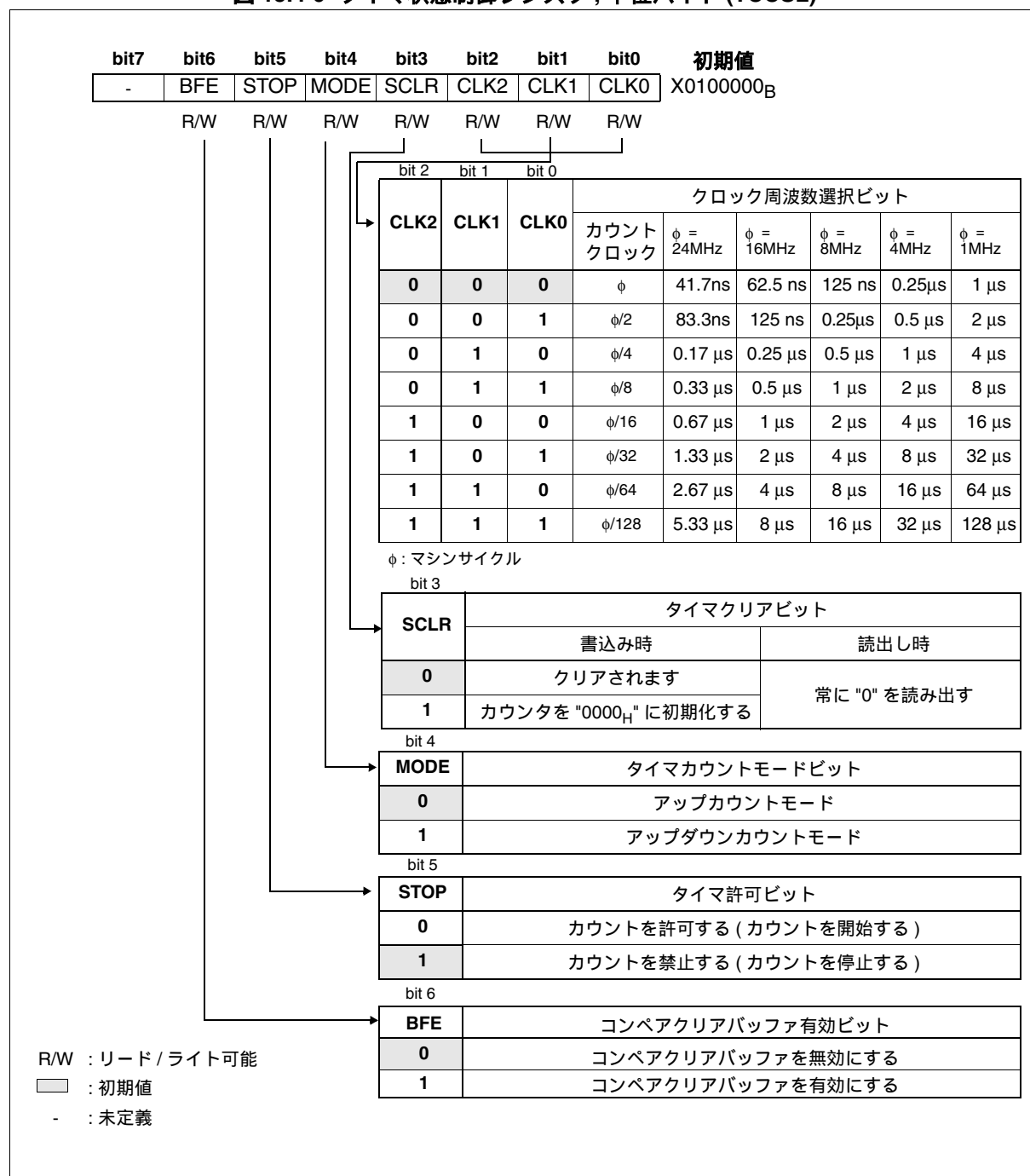
ビット名		機能
bit15	ECKE: クロック選択 ビット	<p>このビットは、内部クロックまたは外部クロックを 16 ビットフリーランタイムのカウントクロックとして選択するために使用します。</p> <p>"0" に設定した場合: 内部クロックが選択されます。カウントクロック周波数を選択するためには、TCCSL レジスタのクロック周波数選択ビット (CLK2 ~ CLK0) も設定しなければなりません。</p> <p>"1" に設定した場合: 外部クロックが選択されます。外部クロックは、"P75/FRCK/AN13" 端子から入力されます。したがって、ポート方向レジスタ (DDR1) の bit7 へ "0" を書き込んで外部クロック入力を有効にしなければなりません。</p> <p>(注意事項)</p> <p>カウントクロックは、このビットが設定されると直ちに変更されます。したがって、このビットの変更は、アウトプットコンペアとインプットキャプチャが停止している間でなければなりません。</p>
bit14	IRQZF: ゼロ検出割込み フラグビット	<p>このビットは、割込みフラグのゼロ検出です。</p> <p>16 ビットフリーランタイムのカウント値が "0000_H" のとき、このビットには "1" がセットされます。</p> <p>"0" に設定した場合: クリアされます。</p> <p>"1" に設定した場合: 影響しません。</p> <p>リードモディファイライト時は、必ず "1" が読み出されます。</p> <p>(注意事項)</p> <ul style="list-style-type: none"> ソフトウェアクリア (TCCSL : SCLR = 1 書き込み) では、このビットは設定されません。 アップダウンカウントモード (MODE = 1) 時は、割込みマスク選択ビット (MSI2 ~ MSI0 が "000_B" 以外) で設定した割込みが発生したときに、このビットに "1" が設定されます。割込みが発生しないときは、このビットに "1" は設定されません。 アップカウントモード (MODE = 0) 時は、MSI2 ~ MSI0 の値とは無関係に、このビットはゼロ検出が発生するたびに設定されます。
bit13	IRQZE: ゼロ検出割込み 要求許可ビット	<p>このビットは、ゼロ検出の割込み要求許可ビットです。</p> <p>このビットと割込みフラグビット (IRQZF : bit14) に "1" が設定されると、CPU に対する割込み要求が生成されます。</p>
bit12 ~ bit10	MSI2 ~ MSI0: 割込みマスク選択 ビット	<p>これらのビットは、アップカウントモード (MODE = 0) 時はコンペアクリア割込みのマスク回数を設定するために使用します。アップダウンカウントモード (MODE = 1) 時は、ゼロ検出割込みのマスク回数を設定するために使用します。</p> <p>"0" に設定した場合: 割込み要因はマスクされません。</p> <p>(注意事項)</p> <p>割込み要因を 2 回マスク、3 回目の割込みを処理する際には、これらのビットに "010_B" を設定しなければなりません。</p>

表 15.4-1 タイマ状態制御レジスタ，上位バイト (TCCSH) (2 / 2)

ビット名		機能
bit9	ICLR: コンペアクリア 割込みフラグ ビット	<p>このビットは，コンペアクリアの割込みフラグです。 コンペアクリア値と 16 ビットフリーランタイム値が一致すると，このビットには "1" が設定されます。 "0" に設定した場合：このビットはクリアされます。 "1" に設定した場合：このビットは影響を受けません。 リードモディファイライト時は，必ず "1" が読み出されます。 (注意事項) ・ アップダウンカウントモード (MODE = 0) 時は，割込みマスク選択ビット (MSI2 ~ MSI0 が "000_B" 以外) で設定した割込みが発生したときに，このビットに "1" が設定されます。割込みが発生しないときは，このビットに "1" は設定されません。 ・ アップダウンカウントモード (MODE = 1) 時は，MSI2 ~ MSI0 ビットの値とは無関係に，このビットはコンペアクリアが発生するたびに設定されます。</p>
bit8	ICRE: コンペアクリア 割込み要求 許可ビット	<p>このビットは，コンペアクリアの割込み要求許可ビットです。 このビットとコンペアクリア割込みフラグビット (ICLR : bit9) に "1" が設定されると，CPU に対する割込み要求が生成されます。</p>

■ タイマ状態制御レジスタ, 下位バイト (TCCSL)

図 15.4-9 タイマ状態制御レジスタ, 下位バイト (TCCSL)



R/W : リード / ライト可能

■ : 初期値

- : 未定義

表 15.4-2 タイマ状態制御レジスタ, 下位バイト (TCCSL) (1 / 2)

ビット名		機能
bit7	未定義ビット	読み出した場合: 値は不定です。 書き込んだ場合: 影響しません。
bit6	BFE: コンペアクリアバッファ有効ビット	このビットは, コンペアクリアバッファを有効にするために使用します。 "0" に設定した場合: コンペアクリアバッファは無効になります。したがってコンペアクリアレジスタに直接書き込むことが可能です。 "1" に設定した場合: コンペアクリアバッファは有効になります。コンペアクリアバッファに書き込まれ保持されていたデータは, 16 ビットフリーランタイムのカウンタ値 "0" が検出されると, コンペアクリアレジスタへ転送されます。
bit5	STOP: タイマ許可ビット	このビットは, 16 ビットフリーランタイムのカウンタを停止 / 開始するために使用します。 "0" に設定した場合: 16 ビットフリーランタイムのカウンタは開始します。 "1" に設定した場合: 16 ビットフリーランタイムのカウンタは停止します。 • 16 ビットフリーランタイムが停止すると, アウトプットコンペアの動作も停止します。
bit4	MODE: タイマカウントモードビット	このビットは, 16 ビットフリーランタイムのカウンタモードを選択するために使用します。 "0" に設定した場合: アップカウントモードが選択されます。タイマは, カウンタ値がコンペアクリアレジスタと一致して "0000 _H " にリセットされるまでカウンタアップし, その後, 再びカウンタアップします。 "1" に設定した場合: アップダウンカウントモードが選択されます。 アップダウンカウントモードでは, タイマデータレジスタはゼロ検出時にタイマカウントモードはアップカウントにリセットされます。 タイマは, カウンタ値がコンペアクリアレジスタと一致すると, カウンタモードは変わります。 このビットは, タイマが動作中であっても停止されていても書き込みが可能です。タイマが動作中の場合は, このビットに書き込まれた値はバッファに入れられ, その後, タイマ値が "0000 _H " になるとバッファの値によりカウントモードが変わります。 (注意事項) アップダウンカウントモード (MODE = 1) では, コンペア一致検出時にカウントモードは変わるため, タイマがダウンカウントしているとき, コンペアクリアレジスタとタイマデータレジスタの設定に注意してください。
bit3	SCLR: タイマクリアビット	このビットは, 16 ビットフリーランタイムを "0000 _H " に初期化するために使用します。 "0" に設定した場合: このビットは "1" であれば, "0" にクリアされます。 "1" に設定した場合: 16 ビットフリーランタイムは, その次のカウンタクロックで "0000 _H " に初期化されます。 リードした場合: 常に "0" が読み出されます。 (注意事項) • このビットは, タイマが停止しているとき (STOP = 1) にタイマを初期化するためには使用できません。タイマデータレジスタ (TCDT) に "0000 _H " を書き込むと, タイマを初期化できます。 • このビットに "1" を書き込んでも, ゼロ検出割込みは生成されません。 • タイマは "0000 _H " に初期化された後, このビットがハードウェアによってクリアされます。タイマは初期化される前に "0" に設定した場合, このビットがクリアされ, タイマも初期化されません。

表 15.4-2 タイマ状態制御レジスタ, 下位バイト (TCCSL) (2 / 2)

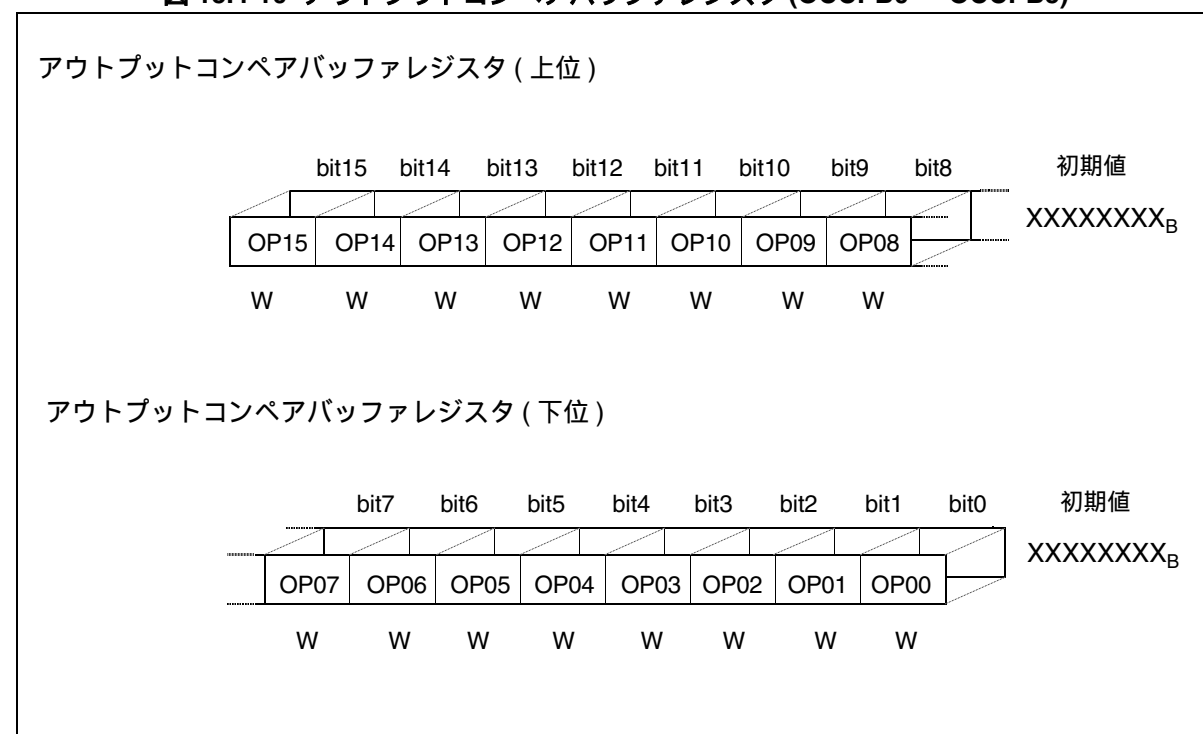
ビット名		機能
bit2 ~ bit0	CLK2 ~ CLK0: クロック周波数 選択ビット	このビットは, 16 ビットフリーランタイマのカウントクロック周波数を選択するために使用します。 カウントクロックは, これらのビットが設定されると直ちに変更されます。 したがって, これらのビットの変更は, アウトプットコンペアとインプットキャプチャが停止している間でなければなりません。

15.4.4 アウトプットコンペアバッファレジスタ (OCCPB0 ~ OCCPB5)/ アウトプットコンペアレジスタ (OCCP0 ~ OCCP5)

アウトプットコンペアバッファレジスタ (OCCPB) は、アウトプットコンペアレジスタ (OCCP) 用の 16 ビットバッファレジスタです。OCCPB レジスタと OCCP レジスタは、両方とも同じアドレスに存在します。

■ アウトプットコンペアバッファレジスタ (OCCPB0 ~ OCCPB5)

図 15.4-10 アウトプットコンペアバッファレジスタ (OCCPB0 ~ OCCPB5)

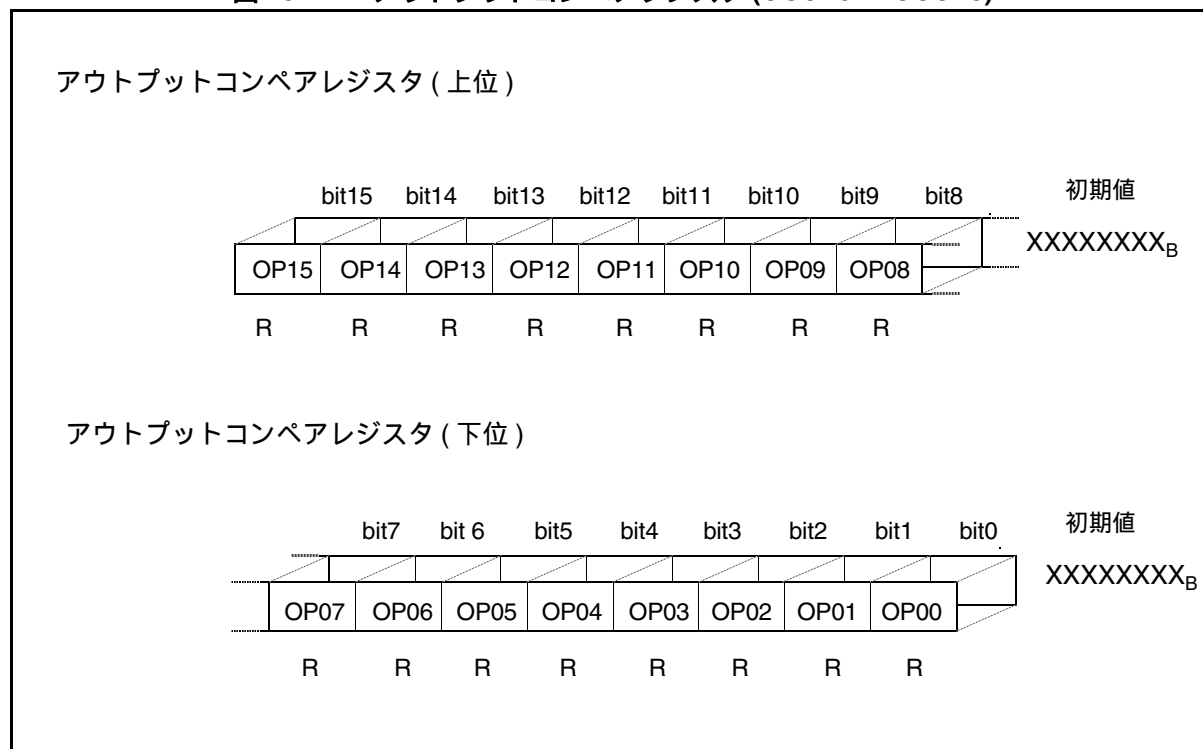


アウトプットコンペアバッファレジスタは、アウトプットコンペアレジスタ (OCCP) 用のバッファレジスタです。バッファ機能が無効になるか (OCS0/OCS2/OCS4 : BUF0/BUF1 = 1), またはフリーランタイムが停止すると、アウトプットコンペアバッファレジスタの値は、直ちにアウトプットコンペアレジスタへ転送されます。バッファ機能が有効になると (OCS0/OCS2/OCS4 : BUF0/BUF1 = 0), 値はコンペア制御レジスタ (OCS1/OCS3/OCS5) の転送選択ビット (BTS) に従ってコンペアクリア一致時またはゼロ検出時に転送されます。

このレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。

■ アウトプットコンペアレジスタ (OCCP0 ~ OCCP5)

図 15.4-11 アウトプットコンペアレジスタ (OCCP0 ~ OCCP5)



アウトプットコンペアレジスタは、16 ビットフリーランタイムのカウント値と比較するために使用する 16 ビットレジスタです。アウトプットコンペアレジスタの初期値は不定なので、タイマの動作を有効にする前にアウトプットコンペアバッファレジスタ (OCCPB) に値を設定しなければなりません。

アウトプットコンペアレジスタの値が 16 ビットフリーランタイムのカウント値と一致すると、コンペア信号が生成され、アウトプットコンペア割込みフラグビット (OCS0/OCS2/OCS4:IOP0/IOP1) が設定されます。出力レベルが設定されると (OCS1/OCS3/OCS5:OTD0/OTD1), アウトプットコンペアレジスタ (OCCP0 ~ OCCP5) に対応する出力レベル波形ジェネレータ RT0 ~ RT5 を反転させることができます。

このレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。

15.4.5 コンペア制御レジスタ (OCS0 ~ OCS5)

コンペア制御レジスタは、RTO0 ~ RTO5 の出力レベル、出力許可、出力レベル反転モード、コンペア動作許可、コンペアー一致割込み許可、コンペアー一致割込みフラグを制御するために使用します。

■ コンペア制御レジスタ、上位バイト (OCS1/OCS3/OCS5)

図 15.4-12 コンペア制御レジスタ、上位バイト (OCS1/OCS3/OCS5)

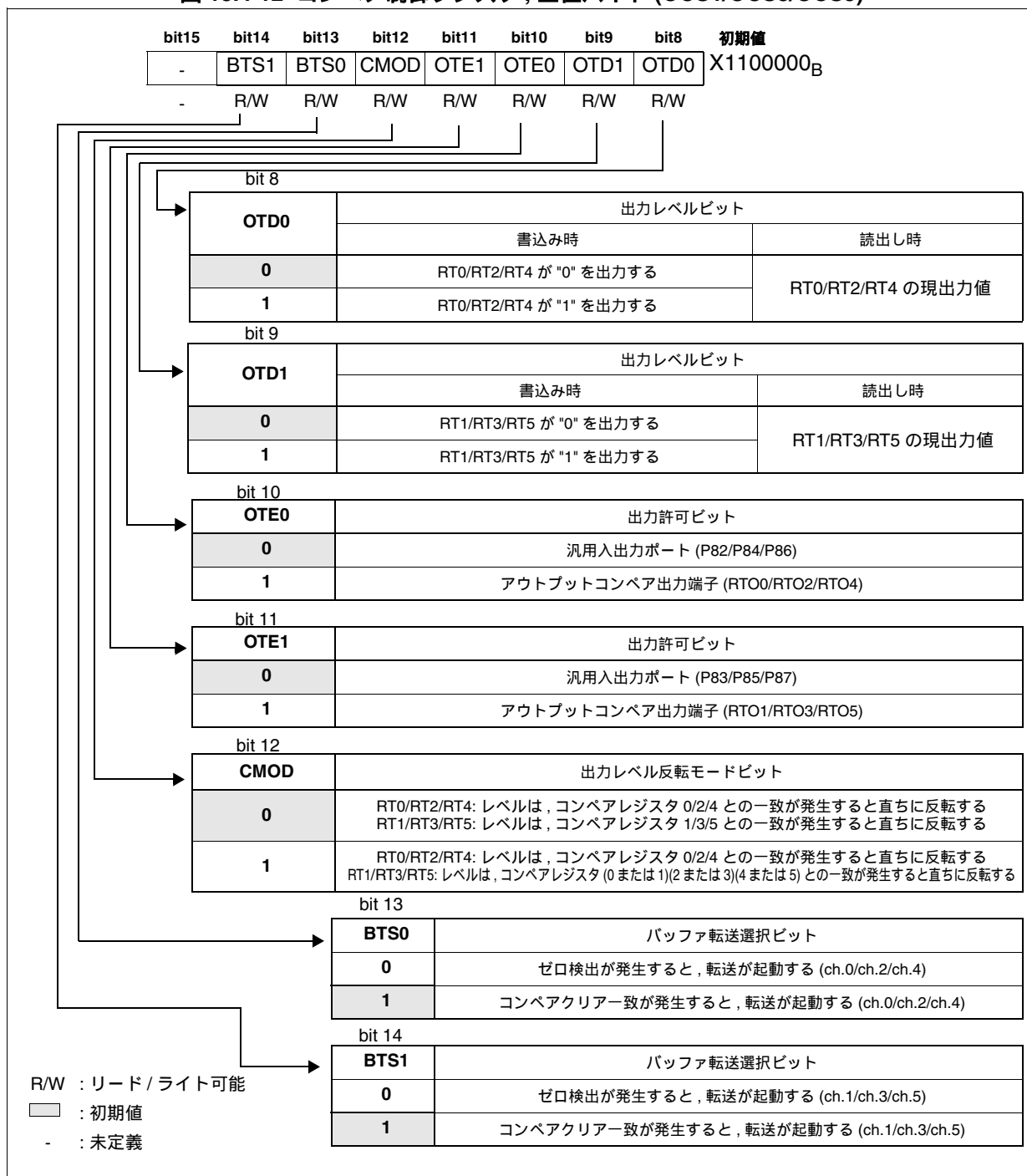


表 15.4-3 コンペア制御レジスタ, 上位バイト (OCS1, 3, 5) (1 / 2)

ビット名		機能
bit15	未定義ビット	読み出した場合: 値は不定です。 書き込んだ場合: 影響しません。
bit14	BTS1: バッファ転送選択 ビット	このビットは, アウトプットコンペアバッファレジスタ (OCCP1/OCCP3/OCCP5) からアウトプットコンペアレジスタ (OCCP1/OCCP3/OCCP5) へのデータ転送時期を選択するために使用します。 "0" に設定した場合: データ転送は, 16 ビットフリーランタイムのカウント値 "0" が検出されると起動します。 "1" を設定した場合: データ転送は, 16 ビットフリーランタイムでコンペアクリア一致が発生すると起動します。
bit13	BTS0: バッファ転送選択 ビット	このビットは, アウトプットコンペアバッファレジスタ (OCCPB0/OCCPB2/OCCPB4) からアウトプットコンペアレジスタ (OCCP0/OCCP2/OCCP4) へのデータ転送時期を選択するために使用します。 "0" を設定した場合: データ転送は, 16 ビットフリーランタイムのカウント値 "0" が検出されると起動します。 "1" を設定した場合: データ転送は, 16 ビットフリーランタイムでコンペアクリア一致が発生すると起動します。
bit12	CMOD: 出力レベル反転 モードビット	このビットは, 端子出力が有効の間 (OTE1 = 1 または OTE0 = 1) に一致が発生した場合に端子出力レベル反転モードを直ちに切換えるために使用します。 "0" を設定した場合: 端子の出力レベルは, 対応するコンペアレジスタとの一致が発生すると直ちに反転します。 RT0/RT2/RT4: レベルは, 16 ビットフリーランタイムとコンペアレジスタ 0/2/4 が一致すると直ちに反転します。 RT1/RT3/RT5: レベルは, 16 ビットフリーランタイムとコンペアレジスタ 1/3/5 が一致すると直ちに反転します。 "1" を設定した場合: コンペアレジスタに対応する RT0/RT2/RT4 の出力レベルは, 本ビットが "0" の場合と同様に反転します。ただし, コンペアレジスタ 1/3/5 に対応する RT1/RT3/RT5 の出力レベルは, コンペアレジスタ 0/2/4 または 1/3/5 で一致が発生すると反転します。コンペアレジスタ 0/2/4 と 1/3/5 が同じ値の場合は, ただ 1 つのコンペアレジスタが使用される場合と同じ動作になります。 RT0/RT2/RT4: レベルは, 16 ビットフリーランタイムとコンペアレジスタ 0/2/4 が一致すると直ちに反転します。 RT1/RT3/RT5: レベルは, 16 ビットフリーランタイムとコンペアレジスタ (0 または 1)(2 または 3)(4 または 5) が一致すると直ちに反転します。
bit11	OTE1: 出力許可ビット	このビットは, ポートへの波形ジェネレータ出力 (RTO1/RTO3/RTO5 ~ P83/P85/P87) を許可するために使用します。 このビットの初期値は "0" です。 (注意事項) 波形ジェネレータが無効 (DTCR: TMD2 ~ TMD0 = 000 _B) の場合は, RTO1/RTO3/RTO5 はアウトプットコンペアと同じ値を出力します。
bit10	OTE0: 出力許可ビット	このビットは, ポートへの波形ジェネレータ出力 (RTO0/2/4 ~ P82/P84/P86) を許可するために使用します。 このビットの初期値は "0" です。 (注意事項) 波形ジェネレータが無効 (DTCR: TMD2 ~ TMD0 = 000 _B) の場合は, RTO0/RTO2/RTO4 はアウトプットコンペアと同じ値を出力します。

表 15.4-3 コンペア制御レジスタ, 上位バイト (OCS1, 3, 5) (2 / 2)

ビット名		機能
bit9	OTD1: 出力レベルビット	このビットは, アウトプットコンペア 1/3/5(RT1/RT3/RT5) の端子出力レベルを変更するために使用します。 コンペア端子出力の初期値は "0" です。 値を書き込む場合は, 必ず前もってコンペア動作を停止させてください。このビットの読出し値は, RT1/RT3/RT5 におけるアウトプットコンペア値を示します。
bit8	OTD0: 出力レベルビット	このビットは, アウトプットコンペア 0/2/4(RT0/RT2/RT4) の端子出力レベルを変更するために使用します。 コンペア端子出力の初期値は "0" です。 値を書き込む場合は, 必ず前もってコンペア動作を停止させてください。このビットの読出し値は, RT0/RT2/RT4 におけるアウトプットコンペア値を示します。

■ コンペア制御レジスタ, 下位バイト (OCS0/OCS2/OCS4)

図 15.4-13 コンペア制御レジスタ, 下位バイト (OCS0/OCS2/OCS4)

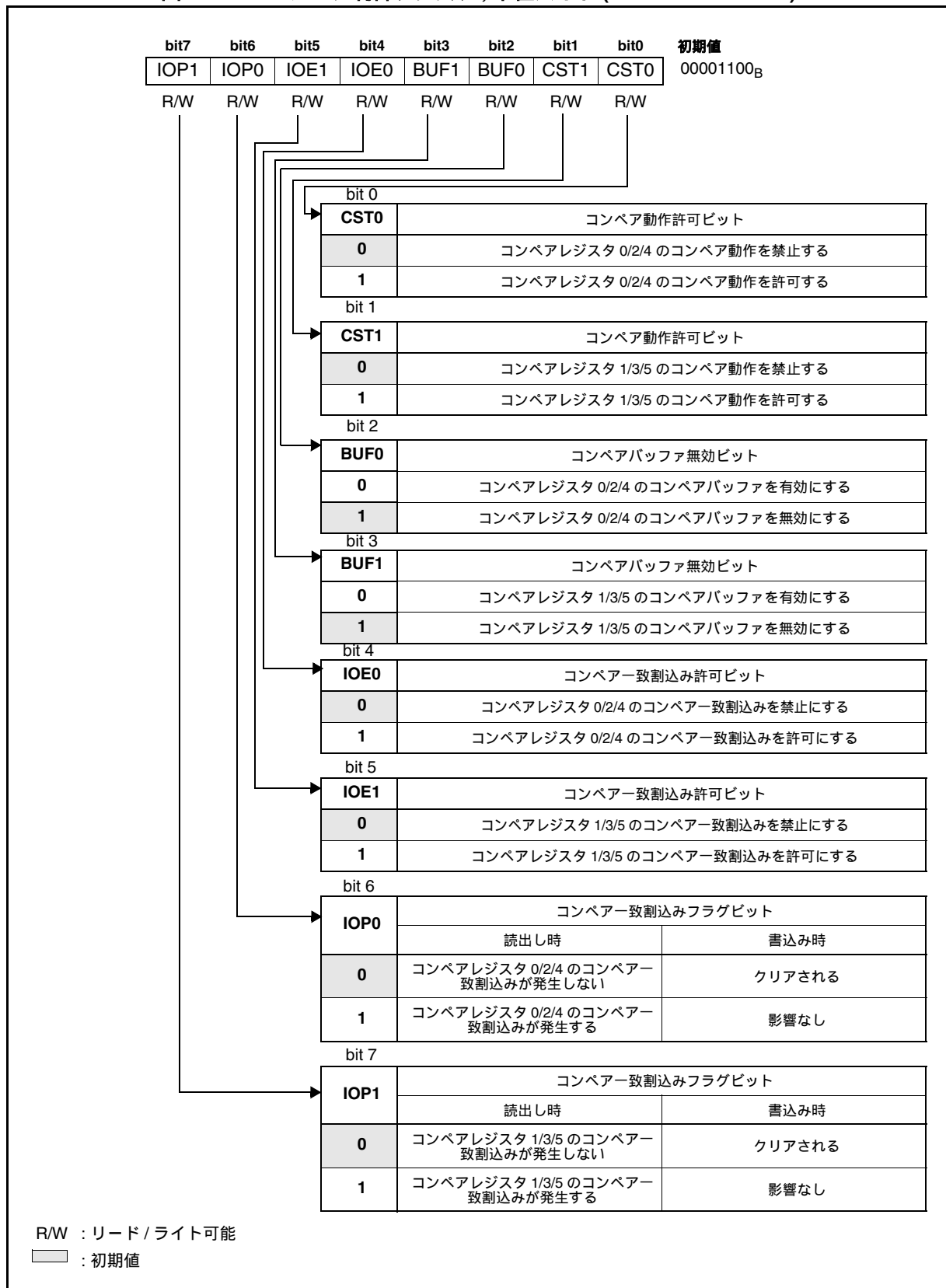


表 15.4-4 コンペア制御レジスタ, 下位バイト (OCS0/OCS2/OCS4)

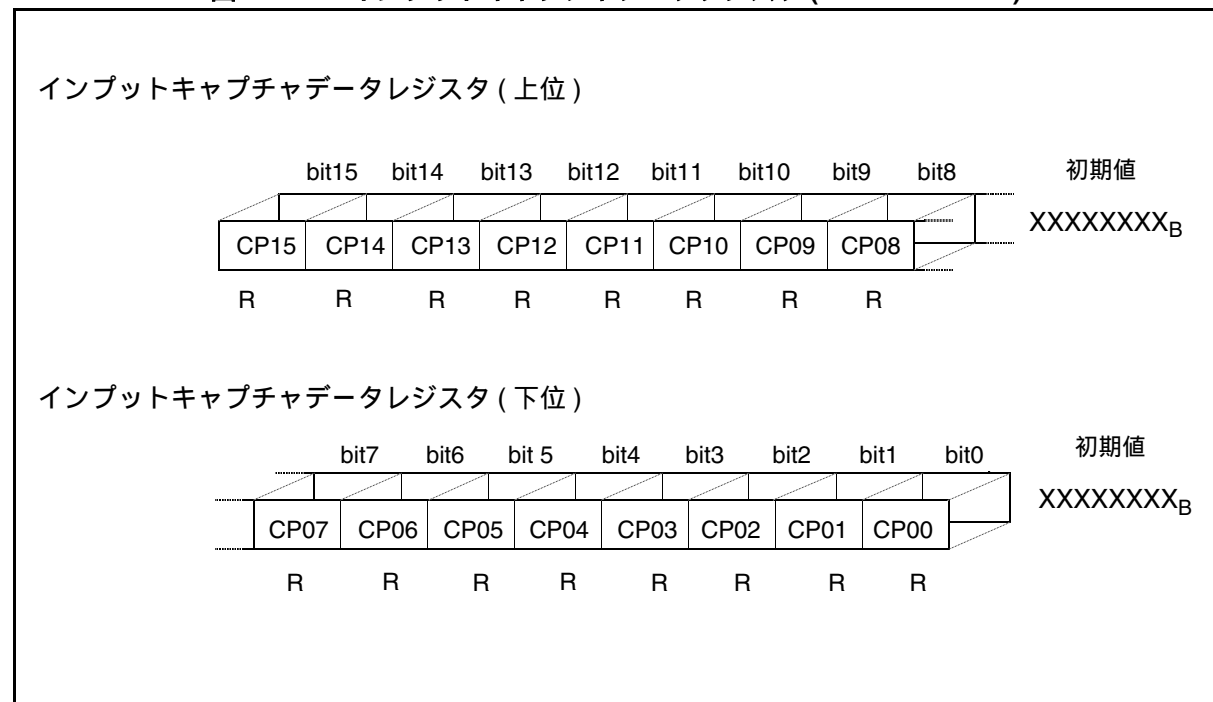
ビット名		機能
bit7	IOP1: コンペア一致割込み フラグビット	このビットは、コンペアレジスタ 1/3/5 が 16 ビットフリーランタイムの値と一致したことを示す割込みフラグです。 このビットは、コンペアレジスタ値が 16 ビットフリーランタイム値に一致した場合に "1" が設定されます。 コンペア一致割込み許可ビット (IOE1) が "許可" になっている間にこのビットが設定されると、アウトプットコンペア割込みが発生します。 "0" を設定した場合: クリアされます。 "1" を設定した場合: 影響しません。 リードモディファイライト時は、必ず "1" が読み出されます。
bit6	IOP0: コンペア一致割込み フラグビット	このビットは、コンペアレジスタ 0/2/4 が 16 ビットフリーランタイムの値と一致したことを示す割込みフラグです。 このビットは、コンペアレジスタ値が 16 ビットフリーランタイム値に一致した場合に "1" が設定されます。 コンペア一致割込み許可ビット (IOE0) が "許可" になっている間にこのビットが設定されると、アウトプットコンペア割込みが発生します。 "0" を設定した場合: クリアされます。 "1" を設定した場合: 影響しません。 リードモディファイライト時は、必ず "1" が読み出されます。
bit5	IOE1: コンペア一致割込み 許可ビット	このビットは、コンペアレジスタ 1/3/5 のアウトプットコンペア割込みを "許可" にするために使用します。 このビットに "1" が書き込まれている間にコンペア一致割込みフラグビット (IOP1) が設定されると、アウトプットコンペア割込みが発生します。
bit4	IOE0: コンペア一致割込み 許可ビット	このビットは、コンペアレジスタ 0/2/4 のアウトプットコンペア割込みを "許可" にするために使用します。 このビットに "1" が書き込まれている間にコンペア一致割込みフラグビット (IOP0) が設定されると、アウトプットコンペア割込みが発生します。
bit3	BUF1: コンペアバッファ 無効ビット	このビットは、アウトプットコンペアレジスタ 1/3/5 のバッファ機能を無効にするために使用します。 "0" を設定した場合: バッファ機能が有効になります。
bit2	BUF0: コンペアバッファ 無効ビット	このビットは、アウトプットコンペアレジスタ 0/2/4 のバッファ機能を無効にするために使用します。 "0" を設定した場合: バッファ機能が有効になります。
bit1	CST1: コンペア動作許可 ビット	このビットは、16 ビットフリーランタイムとコンペアレジスタ 1/3/5 の間のコンペア動作を許可するために使用します。 コンペア動作を許可にする場合は、必ず前もってコンペアレジスタとタイマデータレジスタに値を書き込んでください。 (注意事項) アウトプットコンペアは 16 ビットフリーランタイムクロックと同期化されるので、16 ビットフリーランタイムを停止すると、コンペア動作も停止します。
bit0	CST0: コンペア動作許可 ビット	このビットは、16 ビットフリーランタイムとコンペアレジスタ 0/2/4 の間のコンペア動作を許可するために使用します。 コンペア動作を許可にする場合は、必ず前もってコンペアレジスタとタイマデータレジスタに値を書き込んでください。 (注意事項) アウトプットコンペアは 16 ビットフリーランタイムクロックと同期化されるので、16 ビットフリーランタイムを停止するとゼロ検出、コンペア動作も停止します。

15.4.6 インプットキャプチャレジスタ (IPCP0 ~ IPCP3)

インプットキャプチャレジスタは、入力波形の有効エッジが検出されたときの 16 ビットタイマのカウント値を保持するために使用します。

■ インプットキャプチャレジスタ (IPCP0 ~ IPCP3)

図 15.4-14 インプットキャプチャデータレジスタ (IPCP0 ~ IPCP3)



このレジスタは、対応する外部端子入力波形の有効エッジが検出されたときの 16 ビットタイマ値を格納するために使用します (このレジスタへアクセスする場合は、ワードアクセス命令をご使用ください。このレジスタにデータを書き込むことはできません)。

15.4.7 インพุットキャプチャ状態制御レジスタ (ICS23, PICS01)

インพุットキャプチャ状態制御 (ICS23, PICS01) は、エッジ選択、割込み要求許可、割込み要求フラグを制御するために使用します。またインพุットキャプチャ 0 ~ 3 において検出された有効なエッジを示すためにも使用します。

■ インพุットキャプチャ状態制御レジスタ、上位バイト (ICSH23)

図 15.4-15 インพุットキャプチャ状態制御レジスタ、上位バイト (ICSH23)

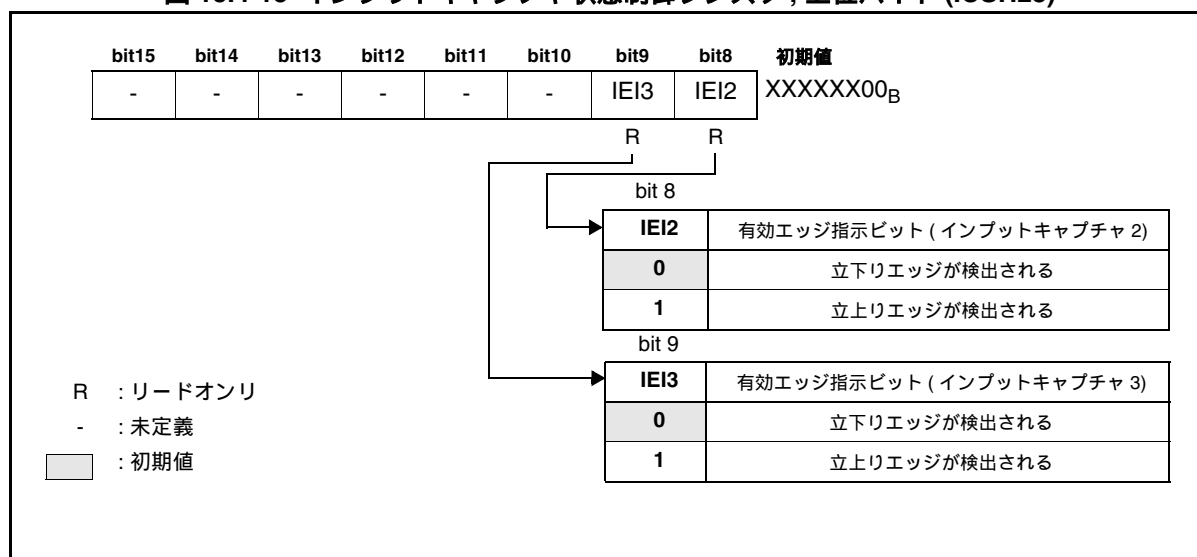


表 15.4-5 インพุットキャプチャ状態制御レジスタ、上位バイト (ICSH23)

ビット名		機能
bit15 ~ bit10	未定義ビット	読み出した場合：値は不定です。 書き込んだ場合：影響しません。
bit9	IEI3: 有効エッジ指示 ビット (インพุット キャプチャ 3)	このビットは、キャプチャレジスタ 3 の有効エッジ指示ビットであり、立上りエッジまたは立下りエッジが検出されたことを示します。 立下りエッジが検出されると、このビットに "0" が書き込まれます。 立上りエッジが検出されると、このビットに "1" が書き込まれます。 このビットは読出し専用ビットです。 (注意事項) EG31, EG30 = 00 _B の場合、読出し値は意味がありません。
bit8	IEI2: 有効エッジ指示 ビット (インพุット キャプチャ 2)	このビットは、キャプチャレジスタ 2 の有効エッジ指示ビットであり、立上りエッジまたは立下りエッジが検出されたことを示します。 立下りエッジが検出されると、このビットに "0" が書き込まれます。 立上りエッジが検出されると、このビットに "1" が書き込まれます。 このビットは読出し専用ビットです。 (注意事項) EG21, EG20 = 00 _B の場合、読出し値は意味がありません。

■ インพุットキャプチャ状態制御レジスタ, 下位バイト (ICSL23)

図 15.4-16 インพุットキャプチャ状態制御レジスタ, 下位バイト (ICSL23)

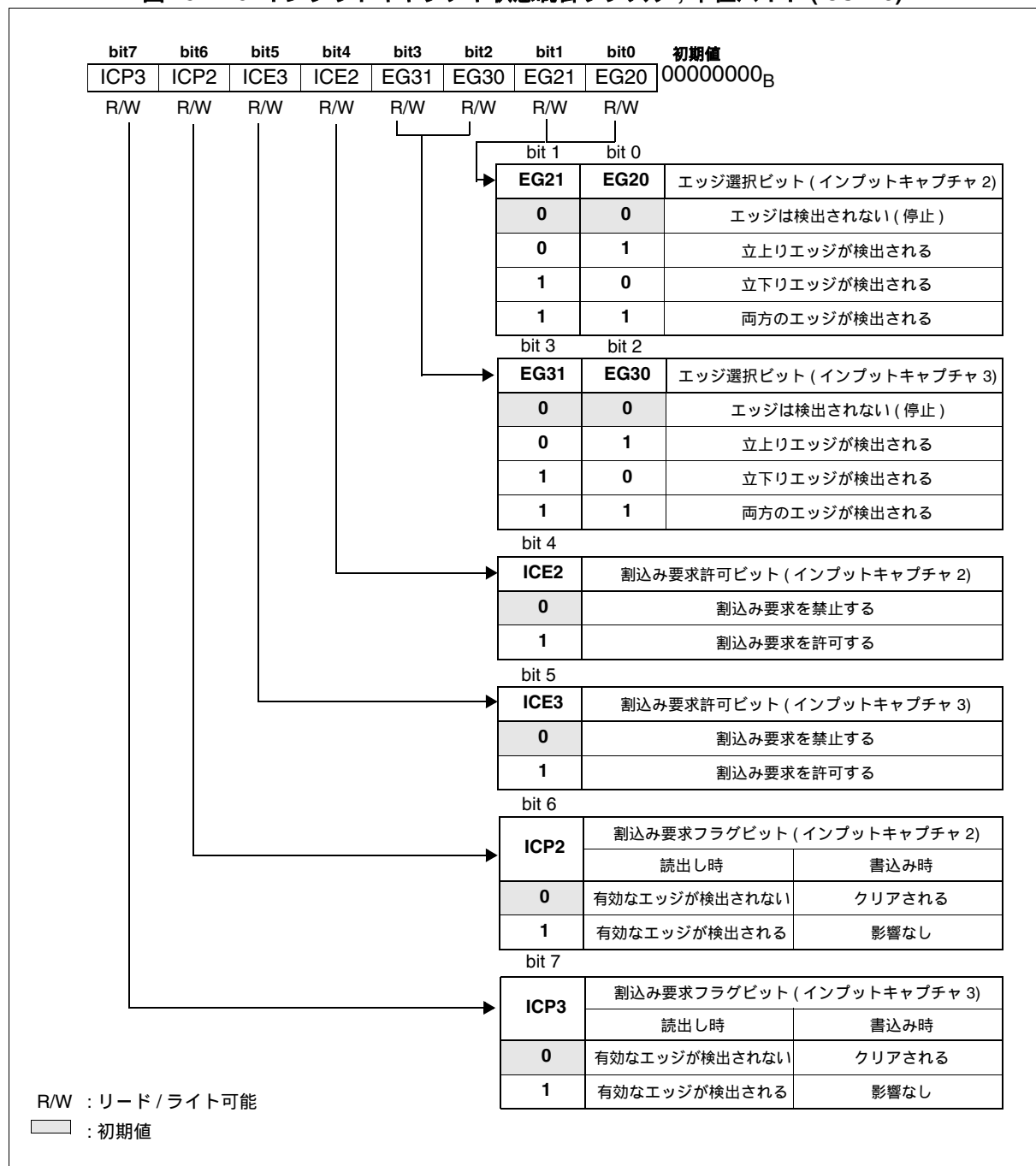


表 15.4-6 インพุットキャプチャ状態制御レジスタ, 下位バイト (ICSL23)

ビット名		機能
bit7	ICP3: 割込み要求フラグビット (インพุットキャプチャ 3)	このビットは, インพุットキャプチャ 3 の割込み要求フラグとして使用します。 このビットは, 外部入力端子の有効エッジが検出されると直ちに "1" が設定されます。 割込み許可ビット (ICE3) が設定されている間に有効エッジが検出されると, 直ちに割込みを生成できます。 "0" を設定した場合: クリアされます。 "1" を設定した場合: 影響しません。 リードモディファイライト時は, 必ず "1" が読み出されます。
bit6	ICP2: 割込み要求フラグビット (インพุットキャプチャ 2)	このビットは, インพุットキャプチャ 2 の割込み要求フラグとして使用します。 このビットは, 外部入力端子の有効エッジが検出されると直ちに "1" が設定されます。 割込み許可ビット (ICE2) が設定されている間に有効エッジが検出されると, 直ちに割込みを生成できます。 "0" を設定した場合: クリアされます。 "1" を設定した場合: 影響しません。 リードモディファイライト時は, 必ず "1" が読み出されます。
bit5	ICE3: 割込み要求許可ビット (インพุットキャプチャ 3)	このビットは, インพุットキャプチャ 3 のインพุットキャプチャ割込み要求を許可にするために使用します。 このビットに "1" が設定されている間に割込みフラグ (ICP3) が設定されると, インพุットキャプチャ割込みが生成されます。
bit4	ICE2: 割込み要求許可ビット (インพุットキャプチャ 2)	このビットは, インพุットキャプチャ 2 のインพุットキャプチャ割込み要求を許可にするために使用します。 このビットに "1" が設定されている間に割込みフラグ (ICP2) が設定されると, インพุットキャプチャ割込みが生成されます。
bit3, bit2	EG31, EG30: エッジ選択ビット (インพุットキャプチャ 3)	これらのビットは, インพุットキャプチャ 3 の外部入力の有効エッジ極性を指定するために使用します。 これらのビットは, インพุットキャプチャ動作を有効にするためにも使用します。
bit1, bit0	EG21, EG20: エッジ選択ビット (インพุットキャプチャ 2)	これらのビットは, インพุットキャプチャ 2 の外部入力の有効エッジ極性を指定するために使用します。 これらのビットは, インพุットキャプチャ動作を有効にするためにも使用します。

■ PPG 出力制御 / インプットキャプチャ状態制御レジスタ, 上位バイト (PICSH01)

図 15.4-17 PPG 出力制御 / インプットキャプチャ状態制御レジスタ, 上位バイト (PICSH01)

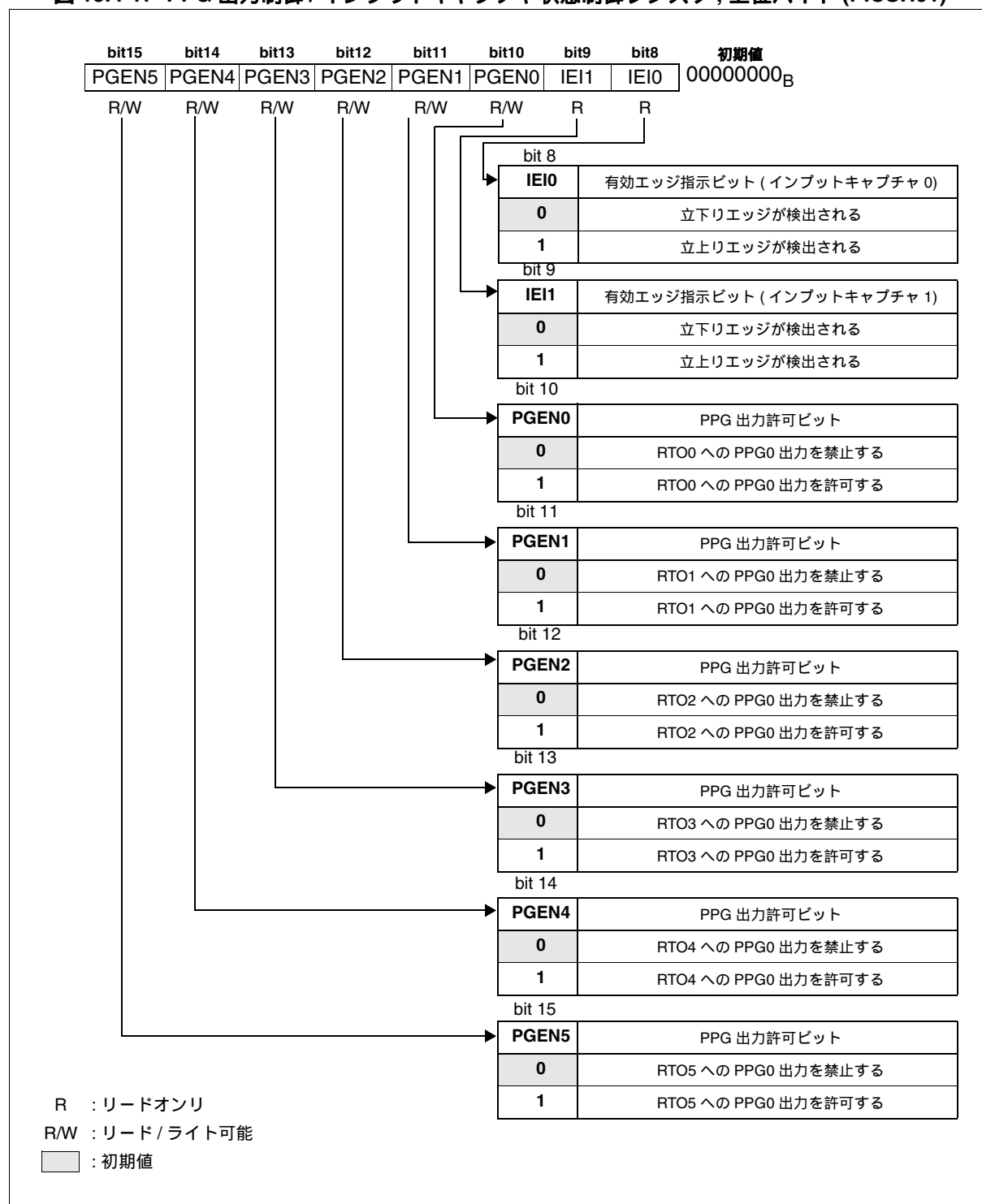


表 15.4-7 PPG 出力制御 / インプットキャプチャ状態制御レジスタ , 上位バイト (PICSH01)

ビット名		機能
bit15 ~ bit10	PGEN5 ~ PGEN0: PPG 出力許可ビット	これらのビットは , RTO0 ~ RTO5 への PPG0 出力を選択するために使用します。
bit9	IEI1: 有効エッジ指示ビット (インプットキャプチャ 1)	このビットは , キャプチャレジスタ 1 の有効エッジ指示ビットであり , 立上りエッジまたは立下りエッジが検出されたことを示します。 立下りエッジが検出されると , このビットに "0" が書き込まれます。 立上りエッジが検出されると , このビットに "1" が書き込まれます。 このビットは読出し専用ビットです。 (注意事項) EG11, EG10 = 00 _B の場合 , 読出し値は意味がありません。
bit8	IEI0: 有効エッジ指示ビット (インプットキャプチャ 0)	このビットは , キャプチャレジスタ 0 の有効エッジ指示ビットであり , 立上りエッジまたは立下りエッジが検出されたことを示します。 立下りエッジが検出されると , このビットに "0" が書き込まれます。 立上りエッジが検出されると , このビットに "1" が書き込まれます。 このビットは読出し専用ビットです。 (注意事項) EG01, EG00 = 00 _B の場合 , 読出し値は意味がありません。

■ インพุットキャプチャ状態制御レジスタ, 下位バイト (PICSL01)

図 15.4-18 インพุットキャプチャ状態制御レジスタ, 下位バイト (PICSL01)

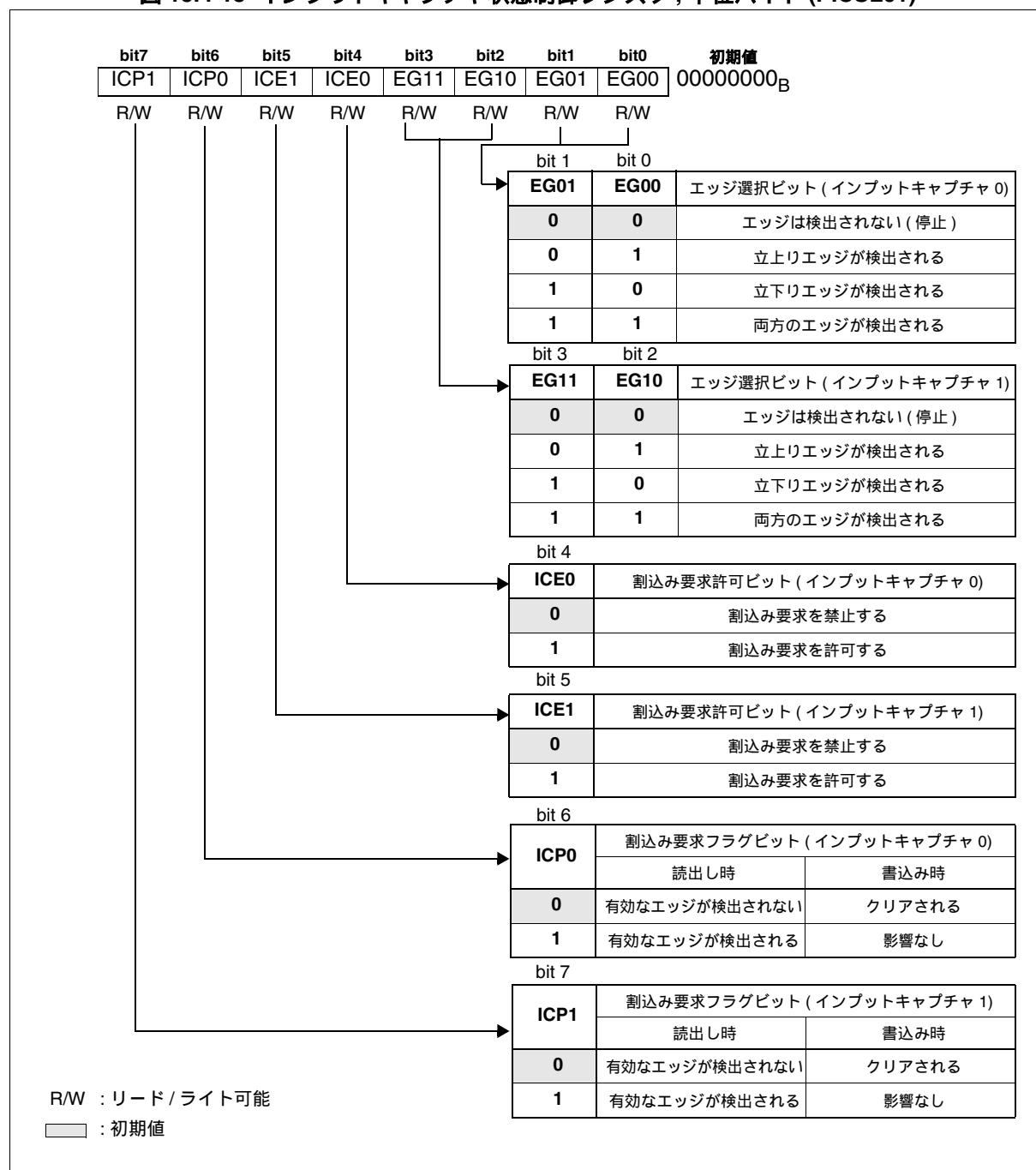


表 15.4-8 インプットキャプチャ状態制御レジスタ，下位バイト (PICSL01)

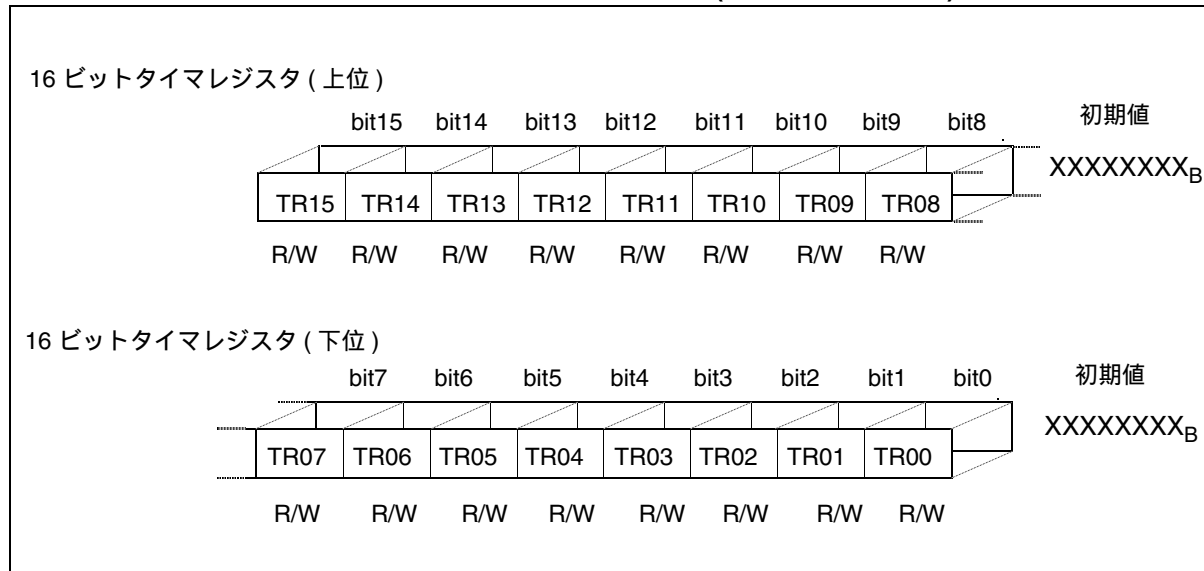
ビット名		機能
bit7	ICP1: 割込み要求フラグ ビット (インプットキャプ チャ 1)	このビットは、インプットキャプチャ 1 の割込み要求フラグとして使用しま す。 このビットは、外部入力端子の有効エッジが検出されると直ちに "1" が設定さ れます。 割込み許可ビット (ICE1) が設定されている間に有効エッジが検出されると、 直ちに割込みが生成されます。 "0" を設定した場合：クリアされます。 "1" を設定した場合：影響しません。 リードモディファイライト時は、必ず "1" が読み出されます。
bit6	ICP0: 割込み要求フラグ ビット (インプットキャプ チャ 0)	このビットは、インプットキャプチャ 0 の割込み要求フラグとして使用しま す。 このビットは、外部入力端子の有効エッジが検出されると直ちに "1" が設定さ れます。 割込み許可ビット (ICE0) が設定されている間に有効エッジが検出されると、 直ちに割込みが生成されます。 "0" を設定した場合：クリアされます。 "1" を設定した場合：影響しません。 リードモディファイライト時は、必ず "1" が読み出されます。
bit5	ICE1: 割込み要求 許可ビット (インプットキャプ チャ 1)	このビットは、インプットキャプチャ 1 のインプットキャプチャ割込み要求を 許可するために使用します。 このビットに "1" が設定されている間に割込みフラグ (ICP1) が設定されると、 インプットキャプチャ 1 割込みが生成されます。
bit4	ICE0: 割込み要求 許可ビット (インプットキャプ チャ 0)	このビットは、インプットキャプチャ 0 のインプットキャプチャ割込み要求を 許可するために使用します。 このビットに "1" が設定されている間に割込みフラグ (ICP0) が設定されると、 インプットキャプチャ 0 割込みが生成されます。
bit3, bit2	EG11, EG10: エッジ選択ビット (インプットキャプ チャ 1)	これらのビットは、インプットキャプチャ 1 の外部入力の有効エッジ極性を指 定するために使用します。 これらのビットは、インプットキャプチャ 1 の動作を有効にするためにも使用 します。
bit1, bit0	EG01, EG00: エッジ選択ビット (インプットキャプ チャ 0)	これらのビットは、インプットキャプチャ 0 の外部入力の有効エッジ極性を指 定するために使用します。 これらのビットは、インプットキャプチャ 0 の動作を有効にするためにも使用 します。

15.4.8 16 ビットタイマレジスタ (TMRR0 ~ TMRR2)

16 ビットタイマレジスタは、16 ビットタイマのコンペア値を保持します。

■ 16 ビットタイマレジスタ (TMRR0 ~ TMRR2)

図 15.4-19 16 ビットタイマレジスタ (TMRR0 ~ TMRR2)



これらのレジスタは、16 ビットタイマの比較値を格納するために使用します。これらのレジスタ値は、16 ビットタイマが動作を開始するとリロードされます。タイマ動作中にこれらのレジスタに値が再書き込みされると、この新しい値は次のタイマ開始/動作時に有効になります。

デッドタイムタイマモード時は、これらのレジスタはノンオーバーラップ時間を設定するために使用します。

$$\text{ノンオーバーラップ時間} = (\text{設定値} + 1) \times \text{選択されたクロック}$$

< 注意事項 >

- "0000_H" を設定することはできません。
- ノンオーバーラップ時間の最大オフセットは、"設定値 - 1" のカウンタ値です。

タイマモード時は、これらのレジスタは PPG0 タイマ動作の GATE 時間を設定するために使用します。

$$\text{GATE 時間} = (\text{設定値} + 1) \times \text{選択されたクロック}$$

< 注意事項 >

- "0000_H" を設定することはできません。最大オフセットは、"設定値 - 1" のカウンタ値です。
- GATE 時間の最大オフセットは、"設定値 - 1" のカウンタ値です。

15.4.9 16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2)

16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2) は、波形ジェネレータの動作モード、割り込み要求許可、割り込み要求フラグ、GATE 信号許可、出力レベル極性を制御するため使用します。

■ 16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2)

図 15.4-20 16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2)

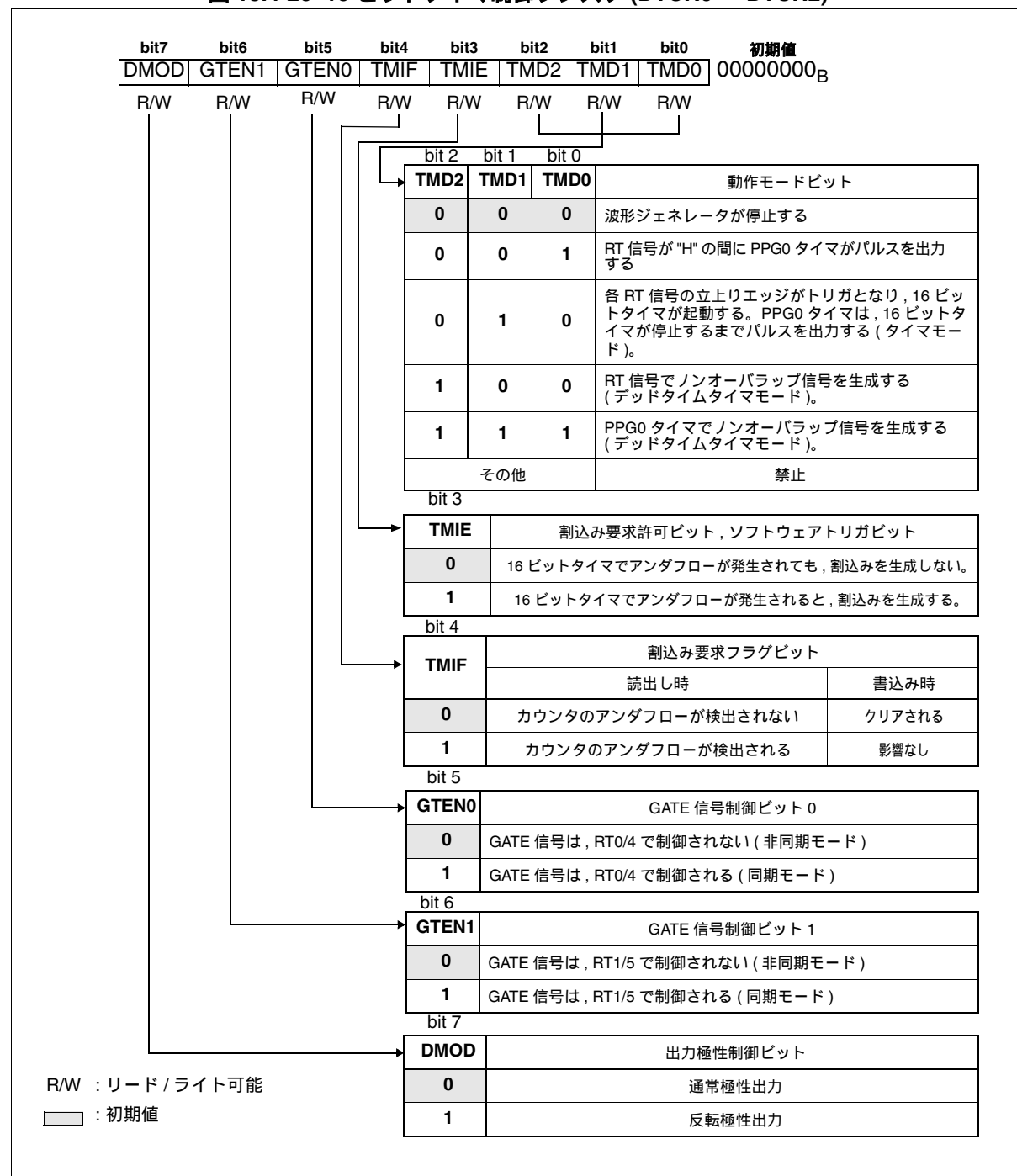


表 15.4-9 16 ビットタイマ制御レジスタ (DTCR0/DTCR2)

ビット名		機能
bit7	DMOD: 出力極性制御ビット	このビットは、デッドタイムタイマモードにおいて U/V/W の出力極性を設定するために使用します。 このビットを設定すると、U/V/W の出力極性は反転します。 (注意事項) このビットは、デッドタイムタイマモードが選択されていない場合 (TMD2 : bit2 = 0) は意味がありません。
bit6	GTEN1: GATE 信号制御ビット 1	このビットは、RT1/RT5 で PPG0 タイマの GATE 信号出力を制御するために使用します。
bit5	GTEN0: GATE 信号制御ビット 0	このビットは、RT0/RT4 で PPG0 タイマの GATE 信号出力を制御するために使用します。
bit4	TMIF: 割り込み要求フラグビット	このビットは、16 ビットタイマの割り込み要求フラグとして使用します。 このビットは、16 ビットタイマ 0/2 でアンダフローが発生すると "1" が設定されます。 "0" を設定した場合 : クリアされます。 "1" を設定した場合 : 影響しません。 リードモディファイライト時は、必ず "1" が読み出されます。 (注意事項) <ul style="list-style-type: none"> このビットは、TMD2 ~ TMD0 が "000_B" または "001_B" の場合のみ機能し、ほかの値の場合は必ず "0" になります。 ソフトウェアクリア ("0" 書込み) とハードウェアセット (16 ビットタイマ 0/2 でアンダフローが発生する) が同時に発生した場合は、ソフトウェアクリアがハードウェアセットよりも優先され、このビットはクリアされます。
bit3	TMIE: 割り込み要求許可ビット、ソフトウェアトリガビット	このビットは、16 ビットタイマ 0/2 のソフトウェアトリガビットおよび割り込み許可ビットとして使用されます。 TMD2 ~ TMD0 : bit2 ~ bit0 が "000 _B " または "001 _B " の場合、このビットは 16 ビットタイマのソフトウェアトリガとして使用されます。このビットを "0" から "1" へ変更すると、16 ビットタイマがトリガとなり、値がリロードされ、ダウンカウントが開始します。 このビットが "1" であり、割り込み要求フラグビット (TMIF : bit4) が "1" の場合、割り込み要求が CPU へ送られます。 (注意事項) 16 ビットタイマを再度トリガとする場合には、このビットに "1" を書き込む前に必ず "0" を書き込んでください。
bit2 ~ bit0	TMD2 ~ TMD0: 動作モードビット	これらのビットは、波形ジェネレータの動作モードを選択するために使用します。 TMD2 ~ TMD0 が "000 _B " の場合、アウトプットコンペアの RT0/RT4 と RT1/RT5 の信号は、RTO0/RTO4 と RTO1/RTO5 のそれぞれに出力されます。また、16 ビットタイマはリロードタイマとしても使用できます。 TMD2 ~ TMD0 が "001 _B " の場合、アウトプットコンペアの RT0/RT4 と RT1/RT5 の信号は、PPG0 出力が禁止 (PICSH01 : PGEN0/PGEN4 = 0, PGEN1/PGEN5 = 0) になると、RTO0/RTO4 と RTO1/RTO5 のそれぞれに出力されます。また、16 ビットタイマはリロードタイマとしても使用できます。 (注意事項) <ul style="list-style-type: none"> デッドタイムタイマモードで波形ジェネレータを動作させるには、必ず RT1/RT5 に対して 2 チャネルモード (OCS1/OCS5 : CMOD = 1) を選択してください。 TMD2 ~ TMD0 = 111_B の場合、RTO0/RTO4 出力と RTO1/RTO5 出力は、PICSH01 : PGEN0/PGEN4, PGEN1/PGEN5 の設定に依存しません。

■ 16 ビットタイマ制御レジスタ (DTCR1)

図 15.4-21 16 ビットタイマ制御レジスタ (DTCR1)

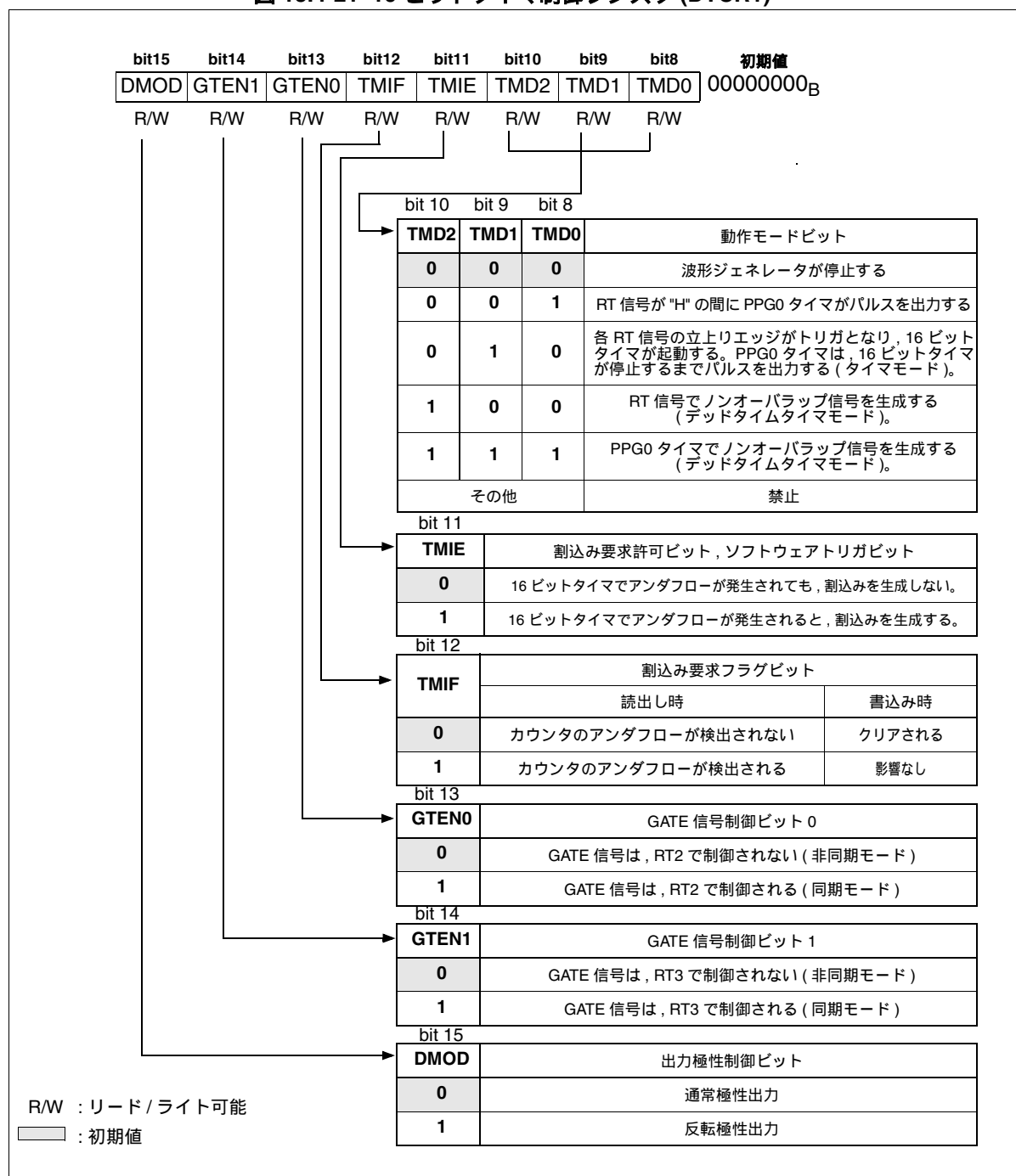


表 15.4-10 16 ビットタイマ制御レジスタ (DTCR1)

ビット名		機能
bit15	DMOD: 出力極性制御ビット	このビットは、デッドタイムタイマモードにおいて U/V/W の出力極性を設定するために使用します。 このビットを設定すると、U/V/W の出力極性は反転します。 (注意事項) このビットは、デッドタイムタイマモードが選択されていない場合 (TMD2 : bit10 = 0) は意味がありません。
bit14	GTEN1: GATE 信号 制御ビット 1	このビットは、RT3 で PPG0 タイマの GATE 信号出力を制御するために使用します。
bit13	GTEN0: GATE 信号制御ビット 0	このビットは、RT2 で PPG0 タイマの GATE 信号出力を制御するために使用します。
bit12	TMIF: 割込み要求フラグビット	このビットは、16 ビットタイマの割込み要求フラグとして使用します。 このビットは、16 ビットタイマ 1 でアンダフローが発生すると "1" が設定されます。 "0" を設定した場合：クリアされます。 "1" を設定した場合：影響しません。 リードモディファイライト時は、必ず "1" が読み出されます。 (注意事項) <ul style="list-style-type: none"> このビットは、TMD2 ~ TMD0 が "000_B" または "001_B" の場合のみ機能し、ほかの値の場合は必ず "0" になります。 ソフトウェアクリア ("0" 書込み) とハードウェアセット (16 ビットタイマ 1 でアンダフローが発生する) が同時に発生した場合は、ソフトウェアクリアがハードウェアセットよりも優先され、このビットはクリアされます。
bit11	TMIE: 割込み要求許可ビット、 ソフトウェア トリガビット	このビットは、16 ビットタイマのソフトウェアトリガビットおよび割込み許可ビットとして使用されます。 TMD2 ~ TMD0 が "000 _B " または "001 _B " の場合、このビットは 16 ビットタイマのソフトウェアトリガとして使用されます。このビットを "0" から "1" へ変更すると、16 ビットタイマがトリガとなり、値がリロードされ、ダウンカウントが開始します。 このビットが "1" であり、割込み要求フラグビット (TMIF : bit12) が "1" の場合、割込み要求が CPU へ送られます。 (注意事項) 16 ビットタイマを再度トリガとする場合には、このビットに "1" を書き込む前に必ず "0" を書き込んでください。
bit10 ~ bit8	TMD2 ~ TMD0: 動作モードビット	これらのビットは、波形ジェネレータの動作モードを選択するために使用します。 TMD2 ~ TMD0 が "000 _B " の場合、アウトプットコンペアの RT2 と RT3 の信号は、RTO2 と RTO3 のそれぞれに出力されます。また、16 ビットタイマはリロードタイマとしても使用できます。 TMD2 ~ TMD0 が "001 _B " の場合、アウトプットコンペアの RT2 と RT3 の信号は、PPG0 出力が禁止 (PICSH01 : PGEN2 = 0, PGEN3 = 0) になると、RTO2 と RTO3 のそれぞれに出力されます。また、16 ビットタイマはリロードタイマとしても使用できます。 (注意事項) <ul style="list-style-type: none"> デッドタイムタイマモードで波形ジェネレータを動作させるには、必ず RT3 に対して 2 チャネルモード (OCS3 : CMOD = 1) を選択してください。 TMD2 ~ TMD0 = 111_B の場合、RTO2 出力と RTO3 出力は、PICSH01 : PGEN2, PGEN3 の設定に依存しません。

15.4.10 波形制御レジスタ (SIGCR)

波形制御レジスタは、動作クロック周波数、ノイズキャンセル機能有効、DTTI 入力有効、DTTI 割込みを制御するために使用します。

■ 波形制御レジスタ (SIGCR)

図 15.4-22 波形制御レジスタ (SIGCR)

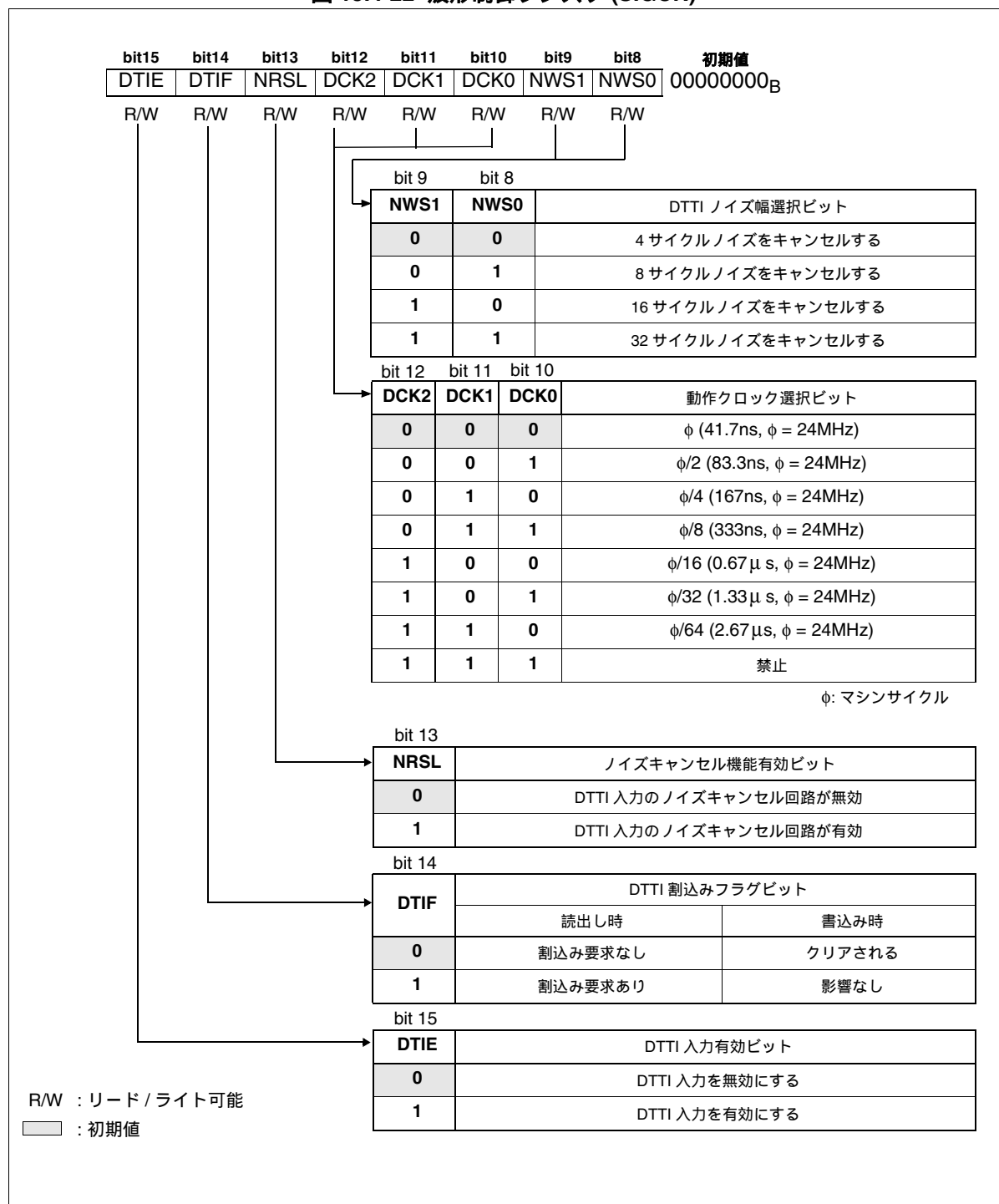


表 15.4-11 波形制御レジスタ (SIGCR)

ビット名		機能
bit15	DTIE: DTTI 入力 有効ビット	このビットは、RTO0 ~ RTO5 端子の出力レベル制御用 DTTI 端子を有効にするために使用します。
bit14	DTIF: DTTI 割込みフラグ ビット	<p>このビットは、DTTI の割込みフラグです。</p> <p>DTTI 入力が無効になり (DTIE = 1)、DTTI の "L" レベルが検出されると、このビットが設定され、割込み要求が CPU へ送られます。</p> <p>"0" を設定した場合：クリアされます。</p> <p>"1" を設定した場合：影響しません。</p> <p>リードモディファイライト時では、必ず "1" が読み出されます。</p> <p>(注意事項)</p> <ul style="list-style-type: none"> ノイズキャンセル機能が有効になった場合 (NRSL = 1)、ノイズパルス幅が渡されると、このビットには "1" が設定されます。 ソフトウェアクリア ("0" 書込み) とハードウェアセット (DTTI の "L" レベル検出) が同時に発生した場合は、ソフトウェアクリアがハードウェアリセットよりも優先され、このビットはクリアされます。
bit13	NRSL: ノイズキャンセル 機能有効ビット	<p>このビットは、ノイズキャンセル機能を有効にするために使用します。</p> <p>ノイズキャンセル回路は、カウンタでオーバフローが発生するまで "L" レベルが保持されると、DTTI 入力信号を受け取ります。カウンタは、"L" レベル入力で作動される n ビットカウンタです。n は、NWS1、NWS0 ビットの設定に基づいて 2、3、4 または 5 のいずれかの値になります。</p> <p>(注意事項)</p> <ul style="list-style-type: none"> ノイズパルス幅をキャンセルするには、約 2ⁿ マシンサイクルが必要になります。 ノイズキャンセル回路を選択すると、内部クロックが停止するモード (停止モードなど) 時は入力が無効になります。
bit12 ~ bit10	DCK2 ~ DCK0: 動作クロック選択 ビット	これらのビットは、16 ビットタイマの動作クロックを選択するために使用します。
bit9, bit8	NWS1, NWS0: DTTI ノイズ幅選択 ビット	これらのビットは、除去する DTTI 端子ノイズパルス幅を選択するために使用します。

15.5 多機能タイマ割込み

多機能タイマは、16 ビットフリーランタイム割込み、16 ビットアウトプットコンペア割込み、16 ビットインプットキャプチャ割込み、および波形ジェネレータ割込みを生成できます。

■ 16 ビットフリーランタイム割込み

16 ビットフリーランタイムの割込み制御ビットと割込み要因を表 15.5-1 に示します。

表 15.5-1 16 ビットフリーランタイムの割込み制御ビットと割込み要因

	16 ビットフリーランタイム	
	コンペアクリア	ゼロ検出
割込み要求フラグビット	タイマ状態レジスタ上位 (TCCSH) の ICLR : bit9	タイマ状態レジスタ上位 (TCCSH) の IRQZF : bit14
割込み要求許可ビット	タイマ状態レジスタ上位 (TCCSH) の ICRE : bit8	タイマ状態レジスタ上位 (TCCSH) の IRQZE : bit13
割込み要因	16 ビットフリーランタイム値がコンペアクリアレジスタ (CPCLR) と一致する	16 ビットフリーランタイム値が "0" になる

16 ビットフリーランタイムの値がコンペアクリアレジスタ (CPCLR) と一致すると、タイマ状態制御レジスタ (TCCSH) の ICLR : bit9 に "1" が設定されます。この状態において割込み要求が許可 (TCCSH レジスタの ICRE : bit8 = 1) になると、割込み要求が割込みコントローラへ出力されます。

タイマ値が "0000_H" になると、タイマ状態制御レジスタ (TCCSH) の IRQZF : bit14 に "1" が設定されます。この状態において割込み要求が許可 (TCCSH レジスタの IRQZE : bit13 = 1) になると、割込み要求が割込みコントローラへ出力されます。

■ 16 ビットフリーランタイム割込みと EI²OS

16 ビットフリーランタイム割込みと EI²OS を表 15.5-2 に示します。

表 15.5-2 16 ビットフリーランタイム割込みと EI²OS

チャンネル	割込み番号	割込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ名	アドレス	下位	中位	上位	
コンペアクリア *1	#34 (22 _H)	ICR11	0000BB _H	FFFF74 _H	FFFF75 _H	FFFF76 _H	
ゼロ検出 *2	#31 (1F _H)	ICR10	0000BA _H	FFFF80 _H	FFFF81 _H	FFFF82 _H	

*1: 16 ビットフリーランタイムコンペアクリアの割込み制御レジスタと同じ値が 16 ビット入力キャプチャチャンネル 0/1 に割当てられます。

*2: 16 ビットフリーランタイム 0 検出の割込み制御レジスタと同じ値が 16 ビット PPG タイマ 2 に割当てられます。

■ 16 ビットアウトプットコンペア割込み

16 ビットアウトプットコンペアの割込み制御ビットと割込み要因を表 15.5-3 に示します。

表 15.5-3 16 ビットアウトプットコンペア 0 ~ 5 の割込み制御ビットと割込み要因

	16 ビットアウトプット コンペア 0/1	16 ビットアウトプット コンペア 2/3	16 ビットアウトプット コンペア 4/5
割込み要求フラグ ビット	コンペア制御レジスタ下位 (OCS0) の IOP1/IOP0 (bit7, bit6)	コンペア制御レジスタ下位 (OCS2) の IOP1/IOP0 (bit7, bit6)	コンペア制御レジスタ下位 (OCS4) の IOP1/IOP0 (bit7, bit6)
割込み要求許可 ビット	コンペア制御レジスタ下位 (OCS0) の IOE1/IOE0 (bit5, bit4)	コンペア制御レジスタ下位 (OCS2) の IOE1/IOE0 (bit5, bit4)	コンペア制御レジスタ下位 (OCS4) の IOE1/IOE0 (bit5, bit4)
割込み原因	16 ビットフリーランタイム 値がアウトプットコンペアレ ジスタ (OCCP0/OCCP1) と一 致する	16 ビットフリーランタイム 値がアウトプットコンペアレ ジスタ (OCCP2/OCCP3) と一 致する	16 ビットフリーランタイム 値がアウトプットコンペアレ ジスタ (OCCP4/OCCP5) と一 致する

16 ビットフリーランタイム値がアウトプットコンペアレジスタ (OCCP0 ~ OCCP5) と一致すると、コンペア制御レジスタ下位 (OCS0/OCS2/OCS4) の IOP1/IOP0 : bit7, bit6 に "1" に設定されます。この状態において割込み要求が許可 (OCS0/OCS2/OCS4 レジスタの IOE1/IOE0 : bit5, bit4 = 1) になると、割込み要求が割込みコントローラへ出力されます。

■ 16 ビットアウトプットコンペア割込みと EI²OS

16 ビットアウトプットコンペア割込みと EI²OS を表 15.5-4 に示します。

表 15.5-4 16 ビットアウトプットコンペア割込みと EI²OS

チャンネル	割込み番号	割込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ名	アドレス	下位	中位	上位	
出力コンペア 0 一致 ^{*1}	#12 (0C _H)	ICR00	0000B0 _H	FFFFCC _H	FFFFCD _H	FFFFCE _H	
出力コンペア 1 一致 ^{*2}	#15 (0F _H)	ICR02	0000B2 _H	FFFFC0 _H	FFFFC1 _H	FFFFC2 _H	
出力コンペア 2 一致 ^{*3}	#17 (11 _H)	ICR03	0000B3 _H	FFFFB8 _H	FFFFB9 _H	FFFFBA _H	
出力コンペア 3 一致 ^{*4}	#19 (13 _H)	ICR04	0000B4 _H	FFFFB0 _H	FFFFB1 _H	FFFFB2 _H	
出力コンペア 4 一致 ^{*5}	#21 (15 _H)	ICR05	0000B5 _H	FFFAA8 _H	FFFAA9 _H	FFFAAA _H	
出力コンペア 5 一致 ^{*6}	#23 (17 _H)	ICR06	0000B6 _H	FFFA0 _H	FFFA1 _H	FFFA2 _H	

*1: 16 ビットアウトプットコンペア 0 の割込み制御レジスタと同じ値が A/D 変換終了に割当てられています。

*2: 16 ビットアウトプットコンペア 1 の割込み制御レジスタと同じ値が 16 ビット PPG タイマ 1 に割当てられています。

*3: 16 ビットアウトプットコンペア 2 の割込み制御レジスタと同じ値が 16 ビットリロードタイマ 1 アンダフローに割当てられています。

*4: 16 ビットアウトプットコンペア 3 の割込み制御レジスタと同じ値が DTP, 外部割込み 0, 1 検出, DTTI に割当てられています。

*5: 16 ビットアウトプットコンペア 4 の割込み制御レジスタと同じ値が DTP, 外部割込み 2, 3 検出, DTTI に割当てられています。

*6: 16 ビットアウトプットコンペア 5 の割込み制御レジスタと同じ値が PWC タイマ 1 に割当てられています。

■ 16 ビットインプットキャプチャ割込み

16 ビットインプットキャプチャの割込み制御ビットと割込み要因を表 15.5-5 に示します。

表 15.5-5 16 ビットインプットキャプチャ 0 ~ 3 の割込み制御ビットと割込み要因

	16 ビットインプットキャプチャ 0/1	16 ビットインプットキャプチャ 2/3
割込み要求フラグビット	インプットキャプチャ状態制御レジスタ下位 (PICSL01) の ICP1/ICP0 (bit7, bit6)	インプットキャプチャ状態制御レジスタ下位 (ICSL23) の ICP3/ICP2 (bit7, bit6)
割込み要求許可ビット	インプットキャプチャ状態制御レジスタ下位 (PICSL01) の ICE1/ICP0 (bit5, bit4)	インプットキャプチャ状態制御レジスタ下位 (ICSL23) の ICP3/ICP2 (bit5, bit4)
割込み要因	有効なエッジが IN0/IN1 端子で検出される	有効なエッジが IN2/IN3 端子で検出される

16 ビットインプットキャプチャでは、有効なエッジが IN0/IN1/ IN2/IN3 端子で検出されると、インプットキャプチャ状態制御レジスタ (PICSL01/ ICSL23) の ICP3/ICP2/ ICP1/ ICP0 : 共に bit7, bit6 に "1" が設定されます。この状態において割込み要求が許可 (PICSL01/ICSL23 レジスタの ICE1/ICE0 : 共に bit5, bit4 = 1) になると、割込み要求は割込みコントローラへ出力されます。

■ 16 ビットインプットキャプチャ割込みと EI²OS

16 ビットインプットキャプチャ割込みと EI²OS を表 15.5-6 に示します。

表 15.5-6 16 ビットインプットキャプチャ割込みと EI²OS

チャンネル	割込み番号	割込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ名	アドレス	下位	中位	上位	
インプットキャプチャ 0/1 ^{*1}	#33 (21 _H)	ICR11	0000BB _H	FFFF78 _H	FFFF79 _H	FFFF7A _H	
インプットキャプチャ 2/3 ^{*2}	#35 (23 _H)	ICR12	0000BC _H	FFFF70 _H	FFFF71 _H	FFFF72 _H	

*1: 16 ビットインプットキャプチャ 0/1 の割込み制御レジスタと同じ値が 16 ビットフリーランタイムコンペアクリアに割当てられています。

*2: 16 ビットインプットキャプチャ 2/3 の割込み制御レジスタと同じ値がタイムベースタイマに割当てられています。

■ 波形ジェネレータ割込み

波形ジェネレータの割込み制御ビットと割込み要因を表 15.5-7 に示します。

表 15.5-7 波形ジェネレータの割込み制御ビットと割込み要因

	波形ジェネレータ	
	16 ビットデッドタイム 0/1/2	DTTI
割込み要求フラグビット	16 ビットタイマ制御レジスタ 上位, 下位 (DTCR0 ~ DTCR2) の TMIF(上位は bit12, 下位は bit4)	波形制御レジスタ (SIGCR) の DTIF(bit14)
割込み要求許可ビット	16 ビットタイマ制御レジスタ 上位, 下位 (DTCR0 ~ DTCR2) の TMIE(上位は bit11, 下位は bit3)	-
割込み要因	16 ビットタイマ 0/1/2 アンダフ ロー	DTTI で "L" レベルが検出され る

波形ジェネレータでは, 16 ビットタイマのアンダフローが発生し, かつ DTCR0 ~ DTCR2 レジスタの TMD2 ~ TMD0(上位は bit10 ~ bit8, 下位は bit2 ~ bit0) が "000_B" または "001_B" のとき, 16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2) の TMIF(上位は bit12, 下位は bit4) には "1" が設定されます。この状態において割込み要求が許可 (DTCR0 ~ DTCR2 レジスタの TMIE (上位は bit11, 下位は bit3) = 1) になると, 割込み要求は割込みコントローラへ出力されます。

■ 波形ジェネレータ割込みと EI²OS

波形ジェネレータ割込みと EI²OS を表 15.5-8 に示します。

表 15.5-8 波形ジェネレータ割込みと EI²OS

チャンネル	割込み番号	割込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ名	アドレス	下位	中位	上位	
16 ビットタイマ 0/1/2 アンダフロー *1	#29 (1D _H)	ICR09	0000B9 _H	FFFF88 _H	FFFF89 _H	FFFF8A _H	△
DTTI *2	#20 (14 _H)	ICR04	0000B4 _H	FFFFAC _H	FFFFAD _H	FFFFAE _H	

*1: 16 ビットタイマ 0/1/2 アンダフローの割込み制御レジスタと同じ値が 16 ビットリロードタイマ 0 アンダフローに割当てられています。

*2: DTTI の割込み制御レジスタと同じ値が多機能タイマの DTP, 外部割込み 0, 1 検出, 16 ビットアウトプットコンペア 3 に割当てられています。

■ 多機能タイマの EI²OS 機能

多機能タイマは EI²OS と関係する回路を持っているので、割込みが発生すると EI²OS を起動できます。

ただし、EI²OS は、割込み制御レジスタ (ICR) を共用するほかのリソースが割込みを使用しない場合に限り使用可能です。例えば、16 ビットフリーランタイマのコンペアクリアが EI²OS を使用している場合、16 ビットインプットキャプチャ ch.0/ch.1 割込みを禁止しなければなりません。

15.6 多機能タイマの動作

多機能タイマの動作について説明します。

■ 多機能タイマの動作

● 16 ビットフリーランタイム

16 ビットフリーランタイムは、リセット解除後、タイマデータレジスタ (TCDT) に設定されている値からカウントアップを開始します。カウンタ値は、16 ビットアウトプットコンペアと 16 ビットインプットキャプチャの基準時間として使用されます。

● 16 ビットアウトプットコンペア

16 ビットアウトプットコンペアは、" 指定されたアウトプットコンペアレジスタに設定されている値 " と "16 ビットフリーランタイム値 " の比較に使用します。一致が検出された場合は、割込みフラグが設定され、出力レベルは反転します。

● 16 ビットインプットキャプチャ

16 ビットインプットキャプチャは、指定された有効なエッジを検出するために使用します。有効なエッジが検出されると、割込みフラグが設定され、16 ビットフリーランタイム値が取出され、インプットキャプチャデータレジスタへ格納されます。

● 波形ジェネレータ

波形ジェネレータは、リアルタイム出力 (RTO0 ~ RTO5), 16 ビット PPG タイマ 0, 16 ビットタイマを使用してさまざまな波形 (デッドタイムを含む) を生成します。

15.6.1 16 ビットフリーランタイムの動作

16 ビットフリーランタイムは、リセット完了後、タイマデータレジスタ (TCDT) に設定されている値からカウントアップを開始します。カウンタ値は、16 ビットアウトプットコンペアと 16 ビットインプットキャプチャの基準時間として使用されます。

■ タイマクリア

16 ビットフリーランタイムのカウンタ値は、下記のいずれかの場合にクリアされます。

- アップカウントモード (TCCSL : MODE = 0) によってコンペアクリアレジスタとの一致が検出された場合
- 動作中に TCCSL レジスタの SCLR ビットに "1" が書き込まれた場合。タイマは、カウントクロックの有効なエッジにクリアされます。

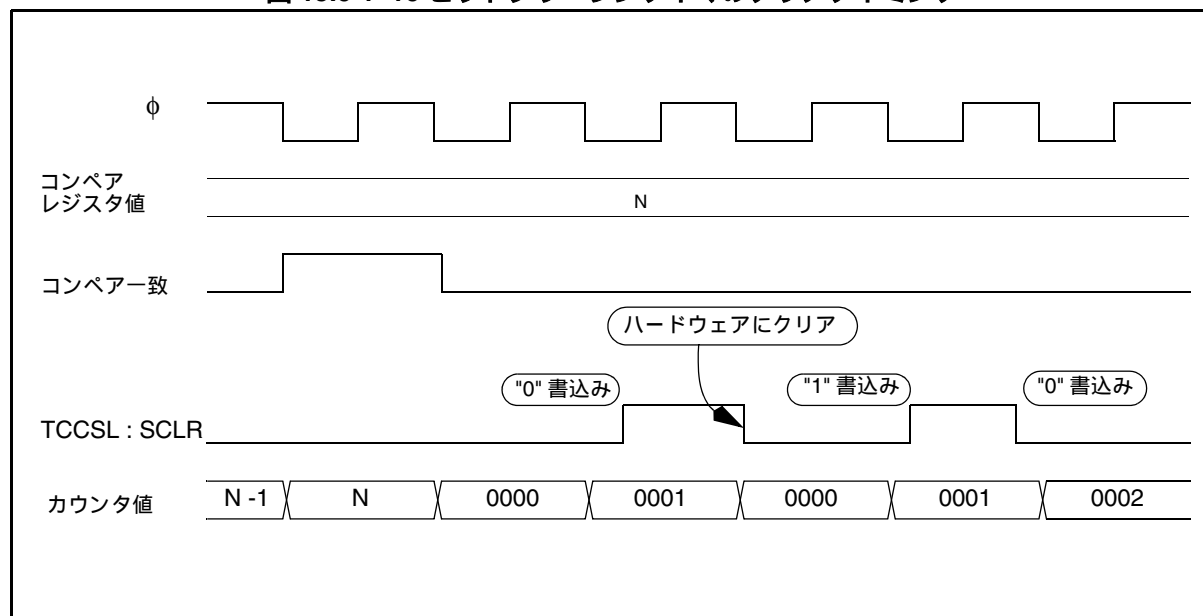
< 注意事項 >

カウントクロックの有効なエッジの前に SCLR ビットに "0" が書き込まれた場合、SCLR ビットはクリアされます。タイマが "0000_H" にクリアされません。

- 停止中に TCDT レジスタに "0000_H" が書き込まれた場合
- リセットされた場合

リセットされると、カウンタは直ちにクリアされます。ソフトウェアクリアされた場合、またはコンペアクリアレジスタとの一致が発生した場合は、カウンタはカウントタイミングと同期してクリアされます。

図 15.6-1 16 ビットフリーランタイムのクリアタイミング



■ タイマモード

16 ビットフリーランタイマでは、以下のどちらかのモードを選択できます。

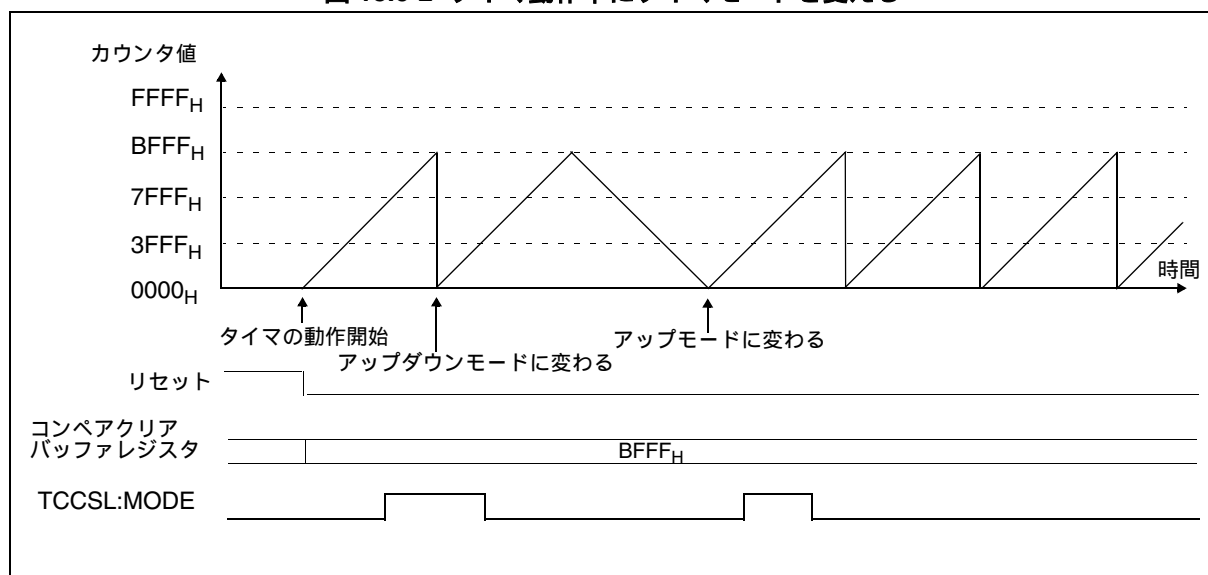
- アップカウントモード (TCCSL レジスタの MODE : bit4 = 0)
- アップダウンカウントモード (TCCSL レジスタの MODE : bit4 = 1)

アップカウントモード時は、カウンタは事前に設定されているタイマデータレジスタ (TCDT) からカウントを開始し、カウンタ値がコンペアクリアレジスタ (CPCLR) の値と一致するまでカウントアップし、次にカウンタは "0000_H" にクリアされ、次に再びカウントアップします。

アップダウンカウントモード時は、カウンタは事前に設定されているタイマデータレジスタ (TCDT) からカウントを開始し、カウンタ値がコンペアクリアレジスタ (CPCLR) の値と一致するまでカウントアップし、次にカウンタがアップカウントからダウンカウントに変わり、カウンタ値が "0000_H" に達するまでカウントダウンし、次に再びカウントアップします。

モードビット (TCCSL レジスタの MODE : bit4) には、タイマが動作中であろうと停止していようといつでも値を書き込むことができます。タイマ動作中にこのビットに書き込まれた値はバッファに入れられ、カウントモードはタイマ値が "0000_H" になると変わります。

図 15.6-2 タイマ動作中にタイマモードを変える



■ コンペアクリアバッファ

コンペアクリアレジスタ (CPCLR) には、有効または無効にできるバッファ機能が存在します。バッファ機能が有効 (TCCSL レジスタの BFE : bit6 = 1) の場合は、コンペアクリアバッファレジスタ (CPCLRB) に書き込まれたデータは、16 ビットフリーランタイム値 "0" が検出されると CPCLR レジスタに転送されます。バッファ機能が無効 (TCCSL ビットの BFE : bit6 = 0) の場合は、CPCLRB レジスタは透過であり、データは CPCLR レジスタに直接書き込むことができます。

図 15.6-3 コンペアクリアバッファが無効 (TCCSL レジスタの BFE : bit6 = 0) 時のアップカウントモードによる動作

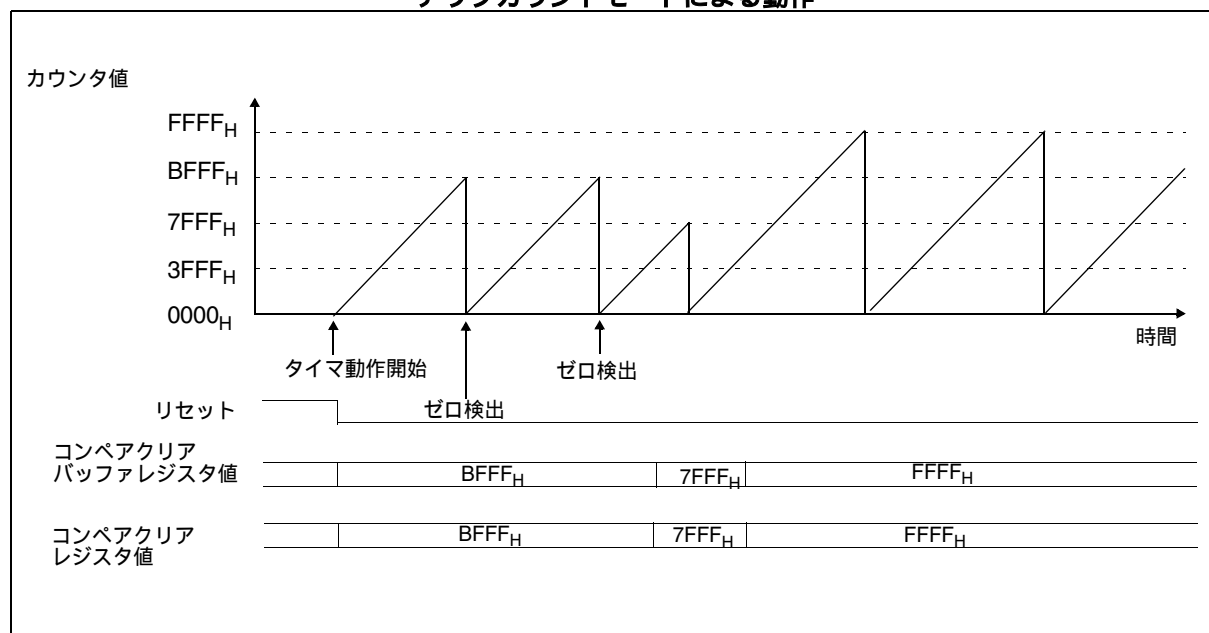


図 15.6-4 コンペアクリアバッファが有効 (TCCSL レジスタの BFE : bit6 = 1) 時のアップカウントモードによる動作

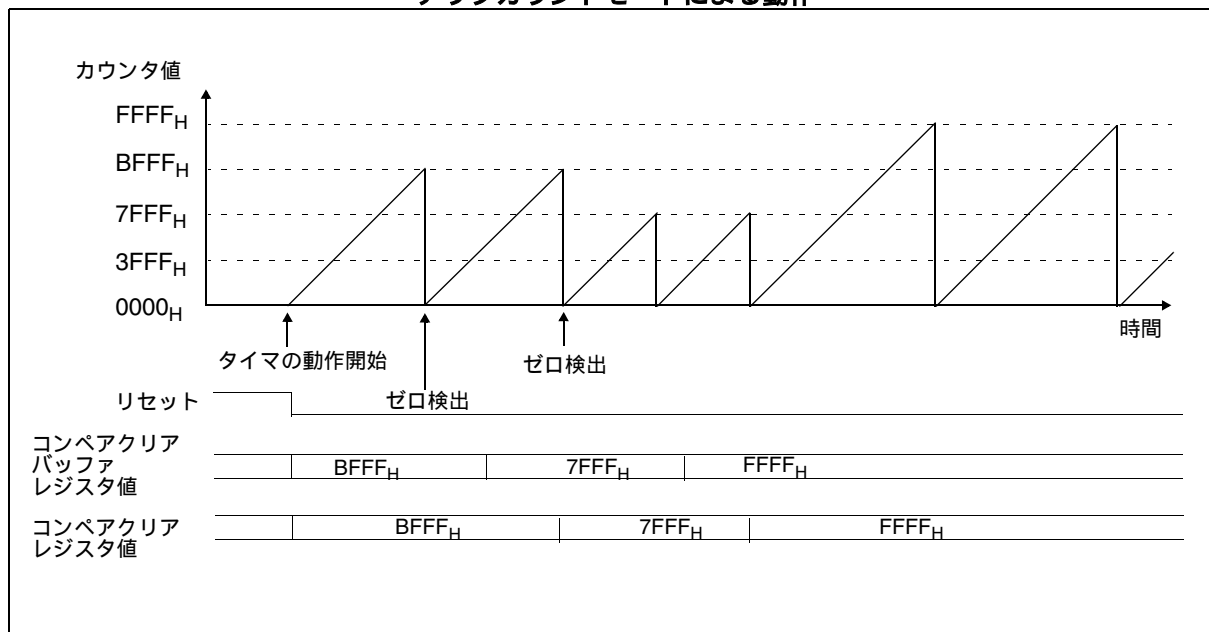
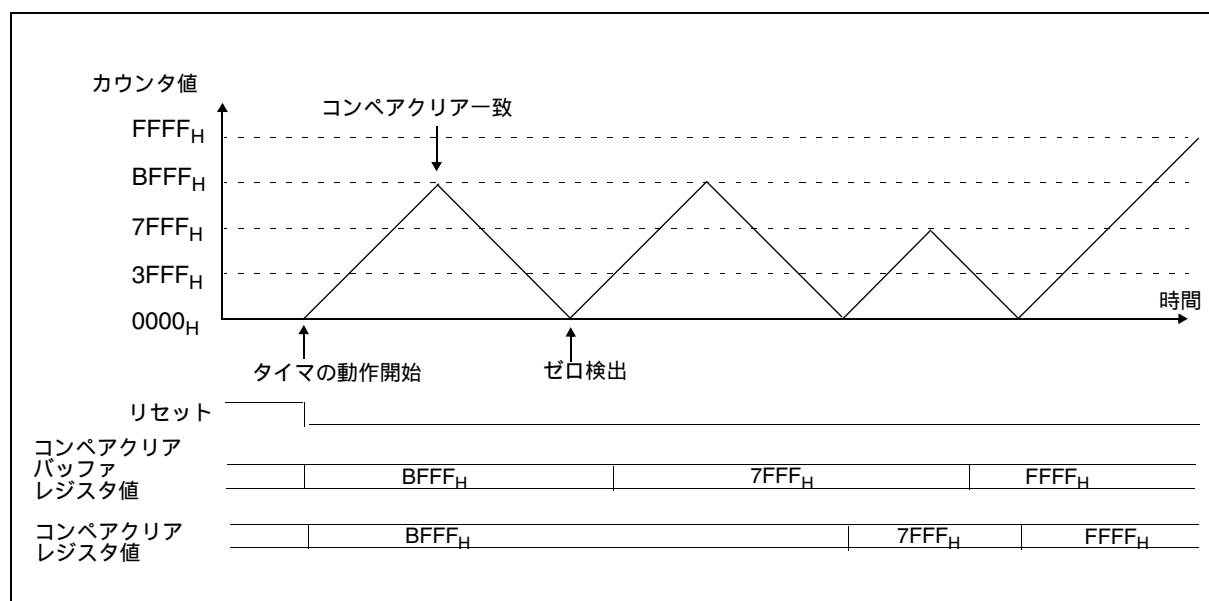


図 15.6-5 コンペアクリアバッファが有効 (TCCSL レジスタの BFE : bit6 = 1) 時の
アップダウンカウントモードによる動作

■ タイマ割込み

16 ビットフリーランタイマでは、以下の 2 つの割込みを生成できます。

- コンペアクリア割込み
- ゼロ検出割込み

コンペアクリア割込みは、タイマ値がコンペアクリアレジスタ (CPCLR) の値と一致すると生成されます。

ゼロ検出割込みは、タイマ値が "0000H" に達すると生成されます。

< 注意事項 >

ソフトウェアクリア (TCCSL レジスタの SCLR : bit3 = 1) は、ゼロ検出割込みを生成しません。

図 15.6-6 アップカウントモード (TCCSL レジスタの MODE : bit4 = 0) で生成された割込み

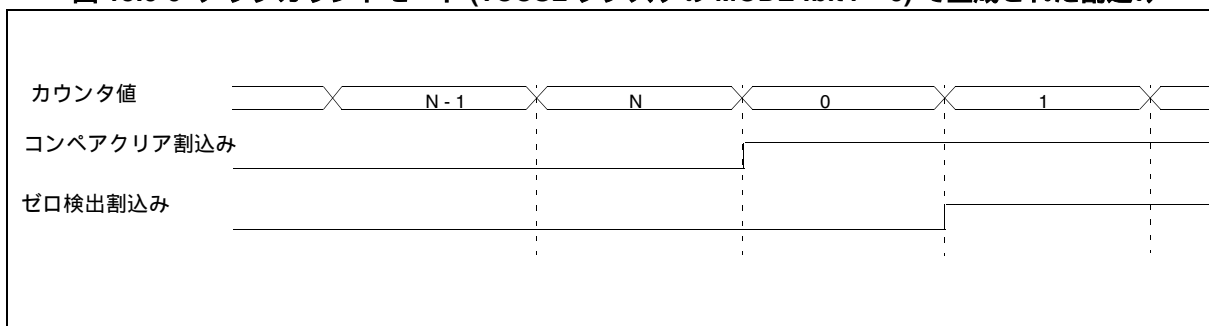
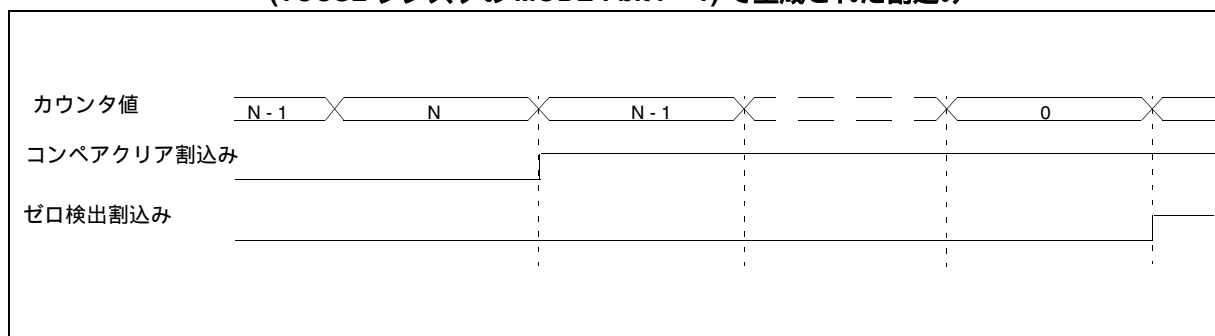


図 15.6-7 アップダウンカウントモード
(TCCSL レジスタの MODE : bit4 = 1) で生成された割込み



■ 割込みマスク機能

TCCSH レジスタの MSI2 ~ MSI0 : bit12 ~ bit10 を設定すると、割込み要求をマスクできます。MSI2 ~ MSI0 ビットは、カウント値が "000_B" に達すると値をリロードする 3 ビットリロードダウンカウンタです。カウント値は、MSI2 ~ MSI0 ビットに直接書き込むことによってロードできます。マスクカウントは、MSI2 ~ MSI0 ビットに設定された値です。MSI2 ~ MSI0 ビットが "000_B" になると、割込み要因はマスクされません。

割込み要因は、カウントモード (TCCSL レジスタの MODE : bit4) によって異なります。アップカウントモード時は、コンペアクリア割込みのみをマスクでき、ゼロ検出割込みは "0" が検出されるたびに生成されます。アップダウンカウントモード時は、ゼロ検出割込みのみをマスクでき、コンペアクリア割込みはコンペアクリアが検出されるたびに生成されます。

< 注意事項 >

ソフトウェアクリア (TCCSL レジスタの SCLR : bit3 = 1) は、ゼロ検出割込みを生成しません。

図 15.6-8 アップカウントモードでマスクされるコンペアクリア割込み

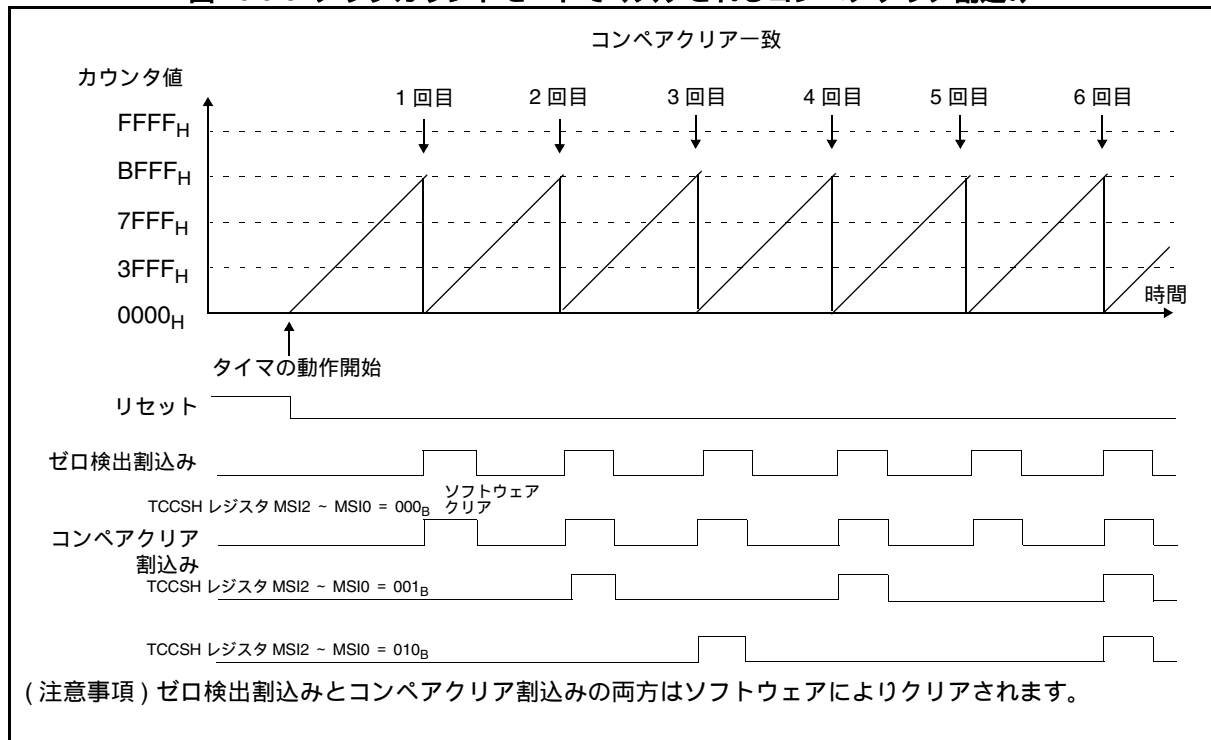
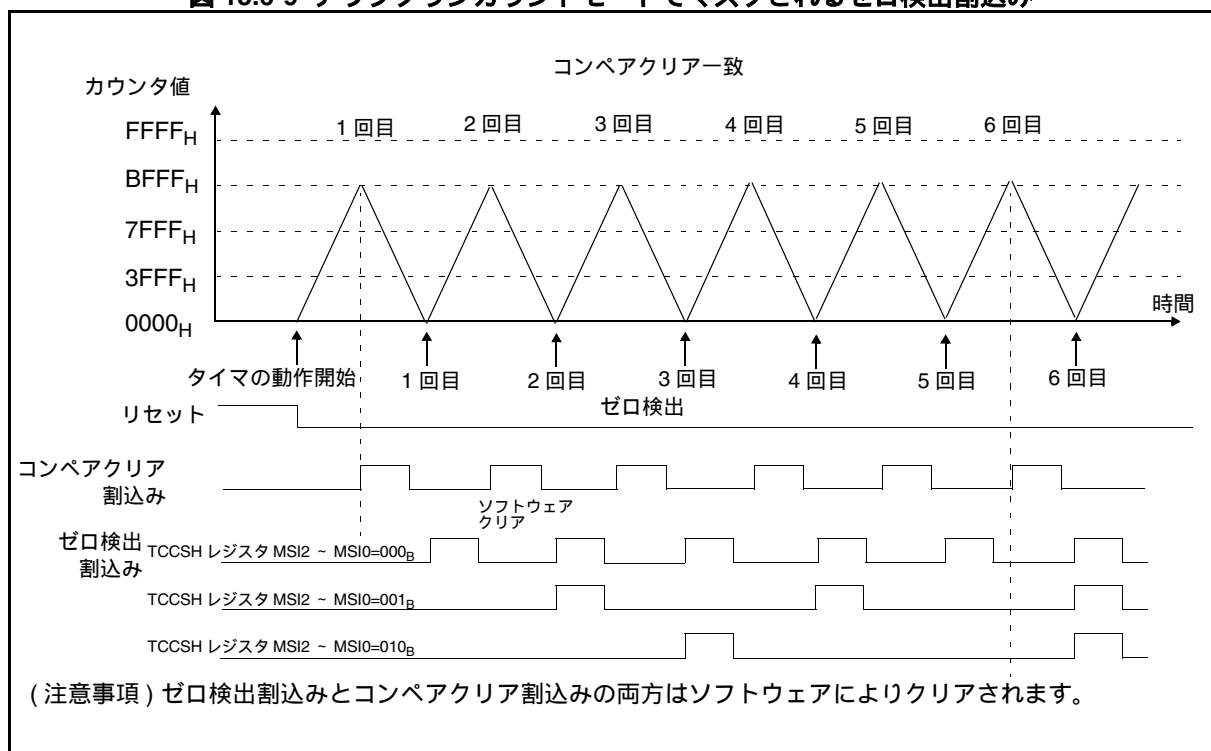


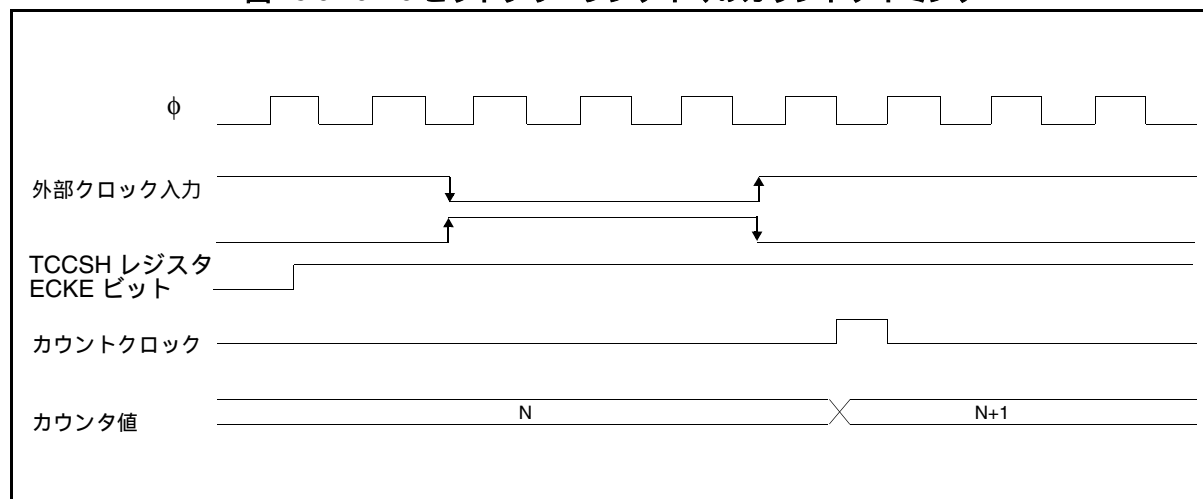
図 15.6-9 アップダウンカウントモードでマスクされるゼロ検出割込み



■ 選択された外部カウントクロック

16 ビットフリーランタイマは、入力クロック（内部クロックまたは外部クロック）に基づいてインクリメントされます。外部クロックが選択されると、外部クロックモード（TCCSH レジスタの ECKE : bit15 = 1）が選択された後、16 ビットフリーランタイマは外部入力の初期値が "1" のとき立上りエッジでカウントアップするか、または外部入力の初期値が "0" のとき立下りエッジでカウントアップします。

図 15.6-10 16 ビットフリーランタイマのカウンタタイミング



15.6.2 16 ビットアウトプットコンペアの動作

アウトプットコンペアは、" 指定されたコンペアレジスタに設定されている値 " と "16 ビットフリーランタイムの値 " の比較に使用します。一致が検出された場合は、割込みフラグが設定され、出力レベルが反転します。

■ 16 ビットアウトプットコンペアの動作

- (a) コンペア動作は、各チャネル (コンペア制御レジスタ (OCS1/OCS3/OCS5) の CMOD : bit12 = 0) において実行できます。

図 15.6-11 出力初期値が "0" のときにコンペアレジスタ 0 とコンペアレジスタ 1 を別々に使用した際の出力波形例 (フリーランタイムはアップカウントモード)

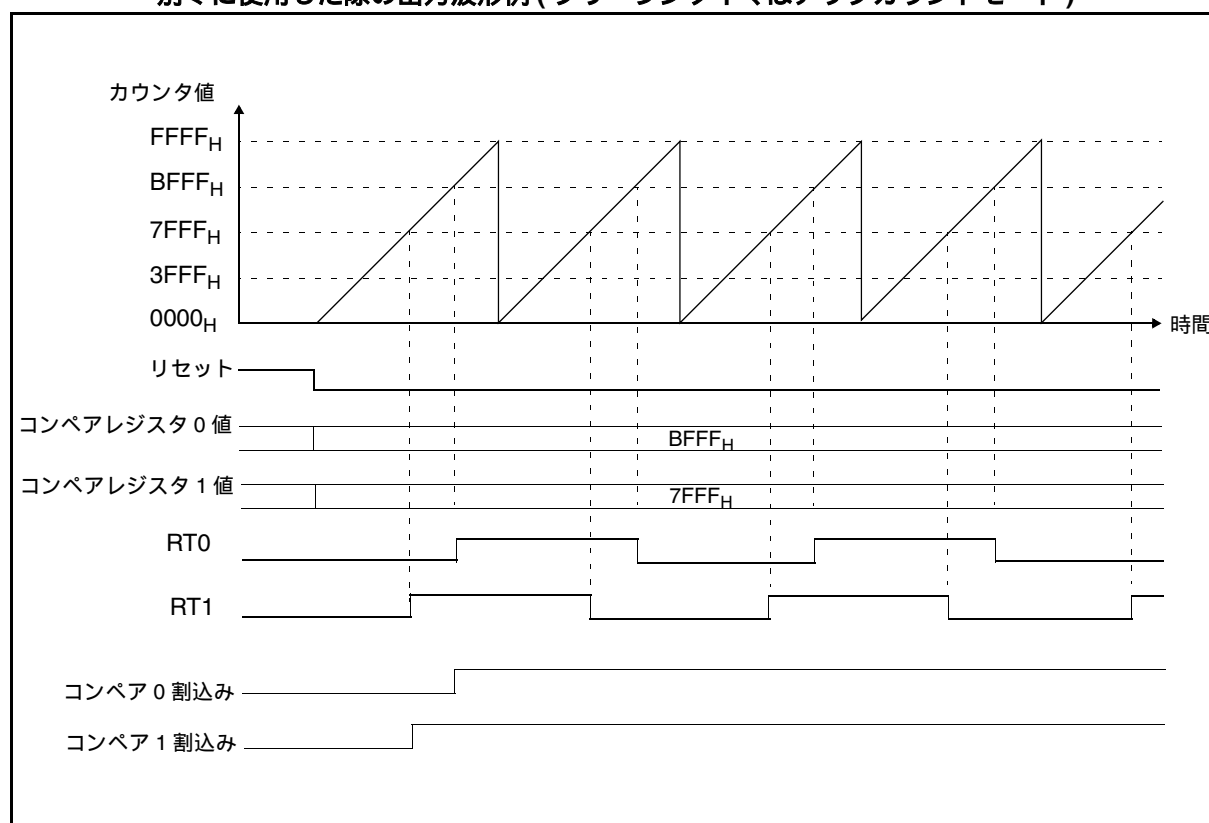
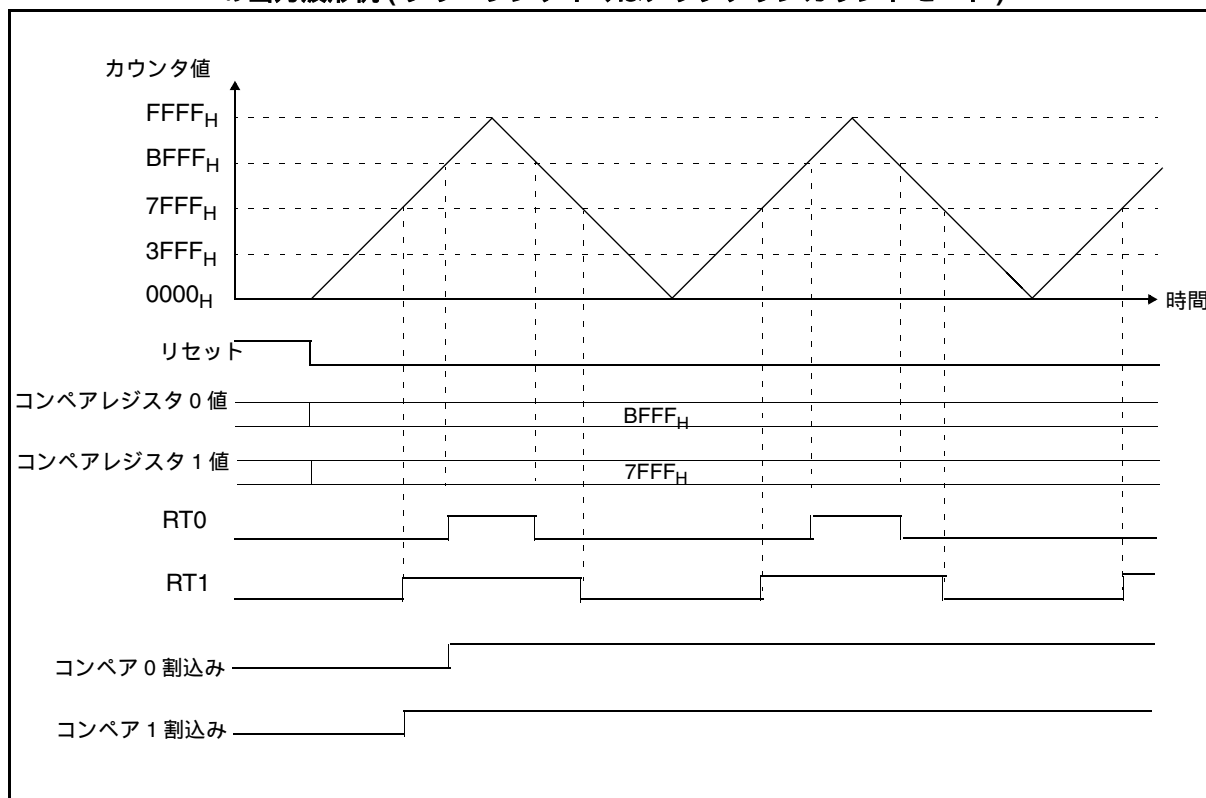


図 15.6-12 出力初期値が "0" のときにコンペアレジスタ 0 とコンペアレジスタ 1 を別々に使用した際の出力波形例 (フリーランタイムはアップダウンカウントモード)



(b) 出力レベルは、一対のコンペアレジスタ (コンペア制御レジスタ (OCS1/OCS3/OCS5) の CMOD: bit12 = 1) を使用して変更できます。

図 15.6-13 出力初期値が "0" のときにコンペアレジスタ 0 とコンペアレジスタ 1 をペアで使用した際の出力波形例 (フリーランタイムはアップカウントモード)

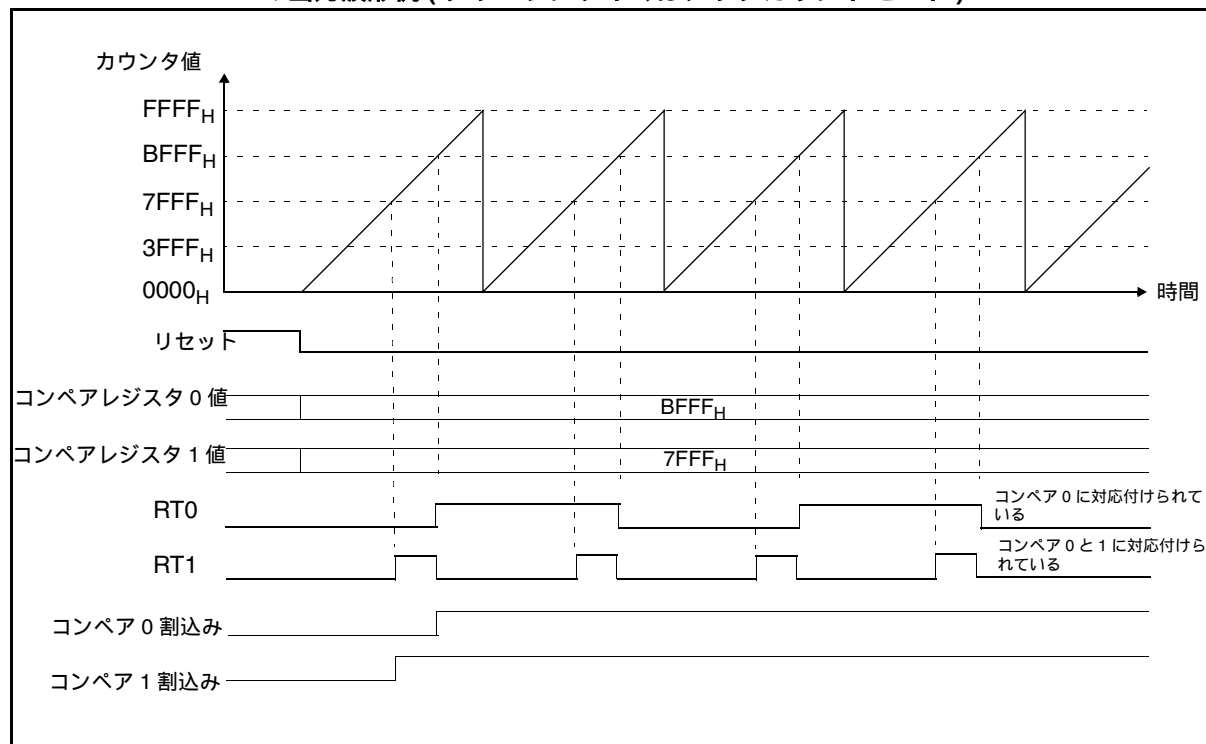
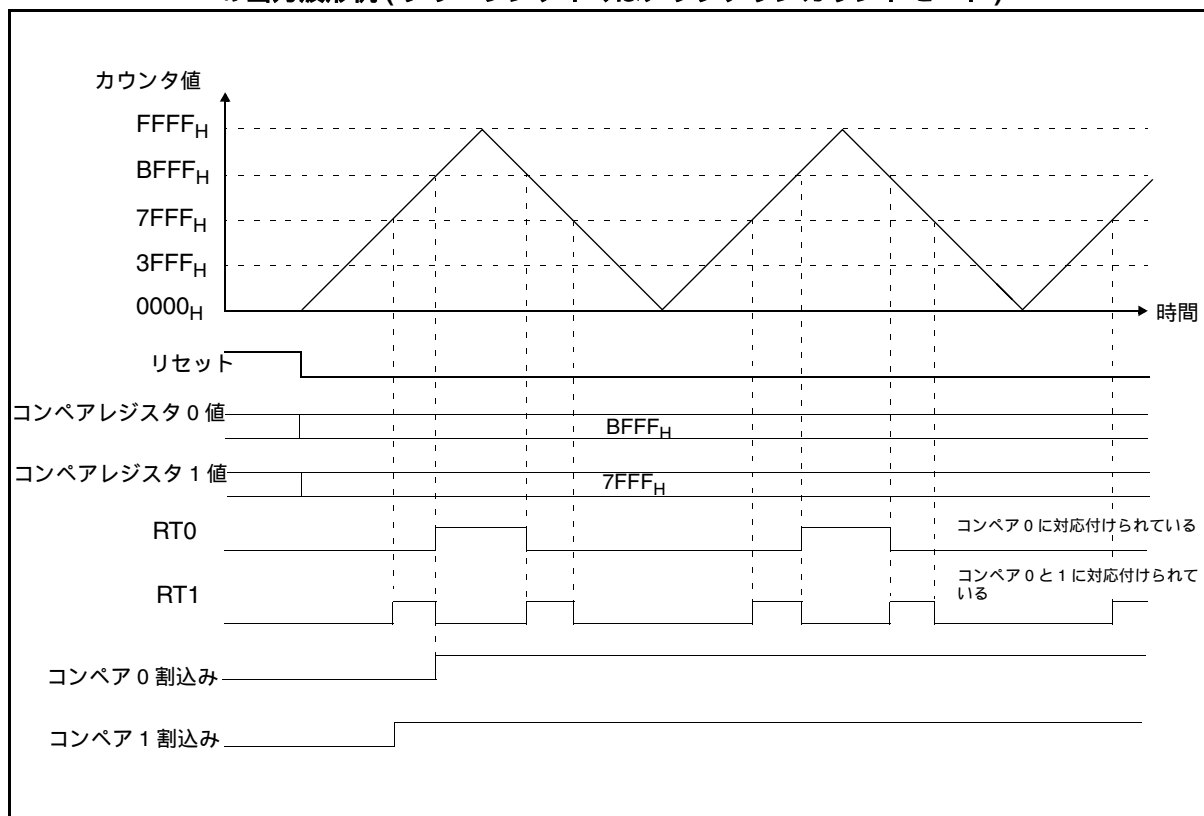
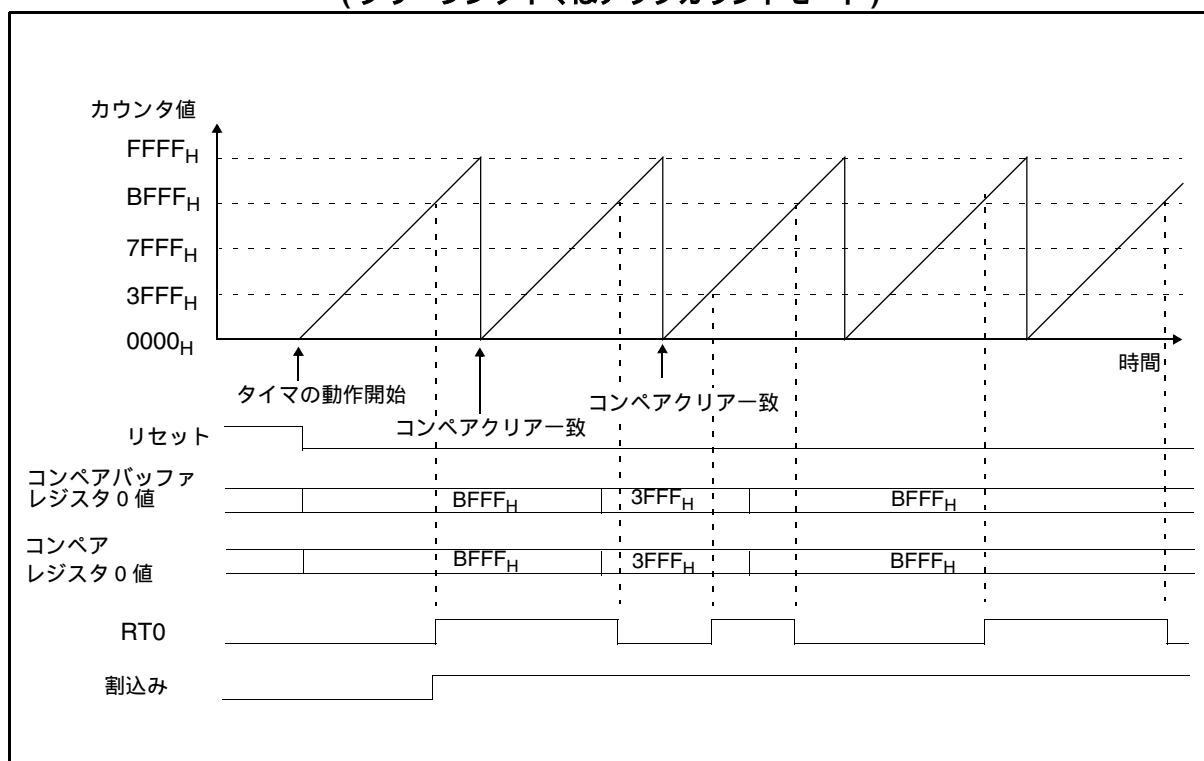


図 15.6-14 出力初期値が "0" のときにコンペアレジスタ 0 とコンペアレジスタ 1 を一緒に使用した際の出力波形例 (フリーランタイマはアップダウンカウントモード)



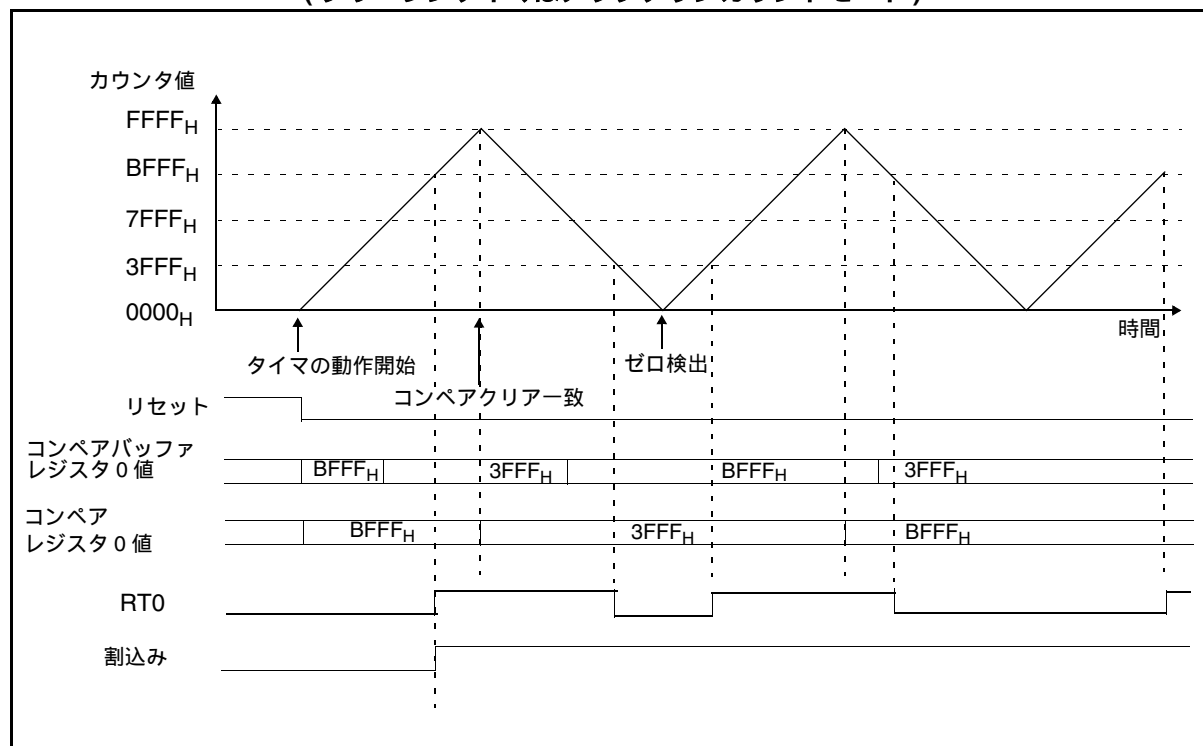
(c) コンペアバッファが無効になったときの出力レベル

図 15.6-15 コンペアバッファが無効になっているときの出力波形例 (フリーランタイマはアップカウントモード)



(d) コンペアクリアー発生時にコンペアバッファが選択された際の出力レベル

図 15.6-16 コンペアバッファが有効になったときの出力波形例
(フリーランタイムはアップダウンカウントモード)



■ 16 ビットアウトプットコンペアタイミング

フリーランタイマがコンペアレジスタ値と一致すると、アウトプットコンペアはコンペア一致信号を生成して出力を反転し、割込みを生成します。コンペア一致が発生すると、出力はカウンタのカウントタイミングと同期して反転します。

< 注意事項 >

コンペアレジスタが更新されると、カウンタ値とは比較されません。

図 15.6-17 コンペアレジスタが更新されたときのコンペア動作

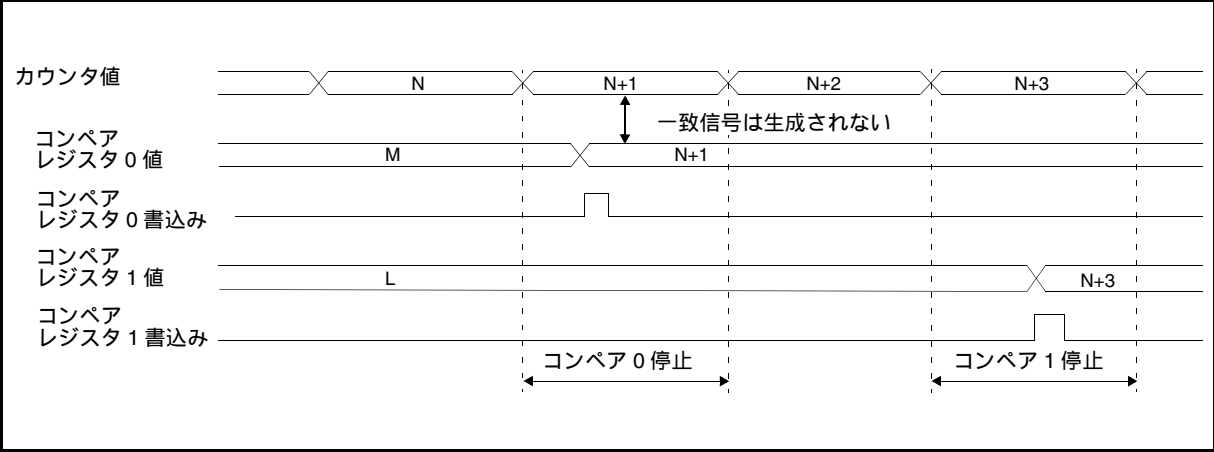


図 15.6-18 コンペア割込みタイミング

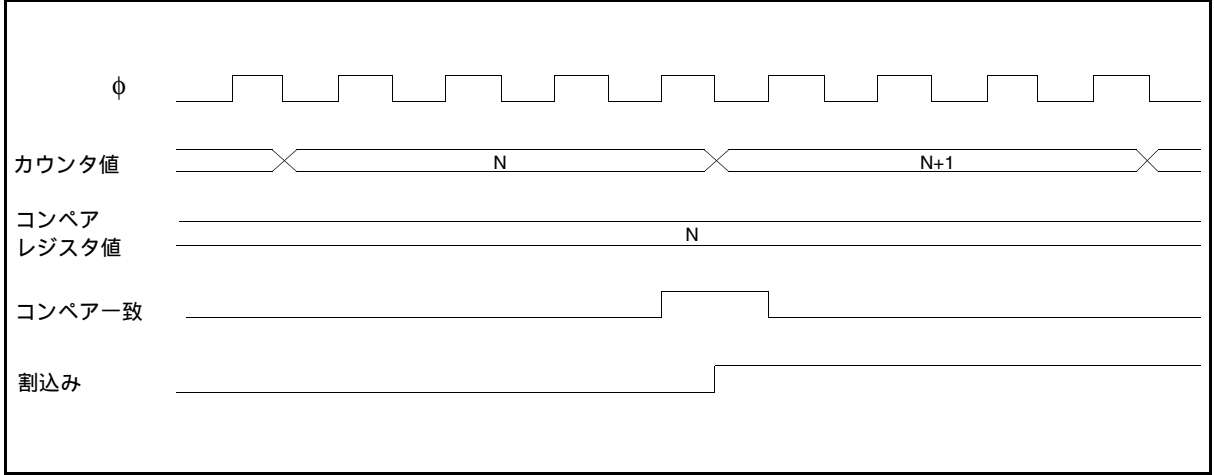
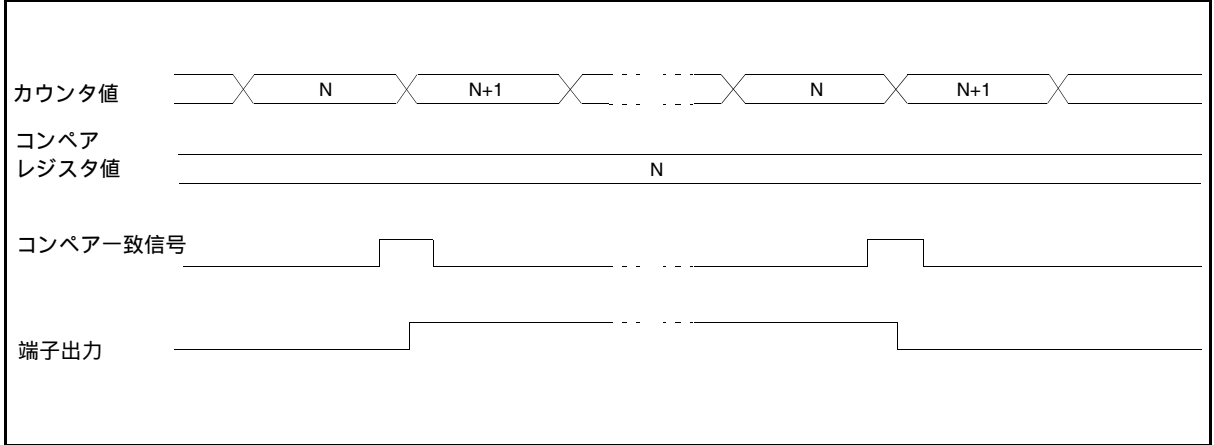


図 15.6-19 出力端子変更タイミング

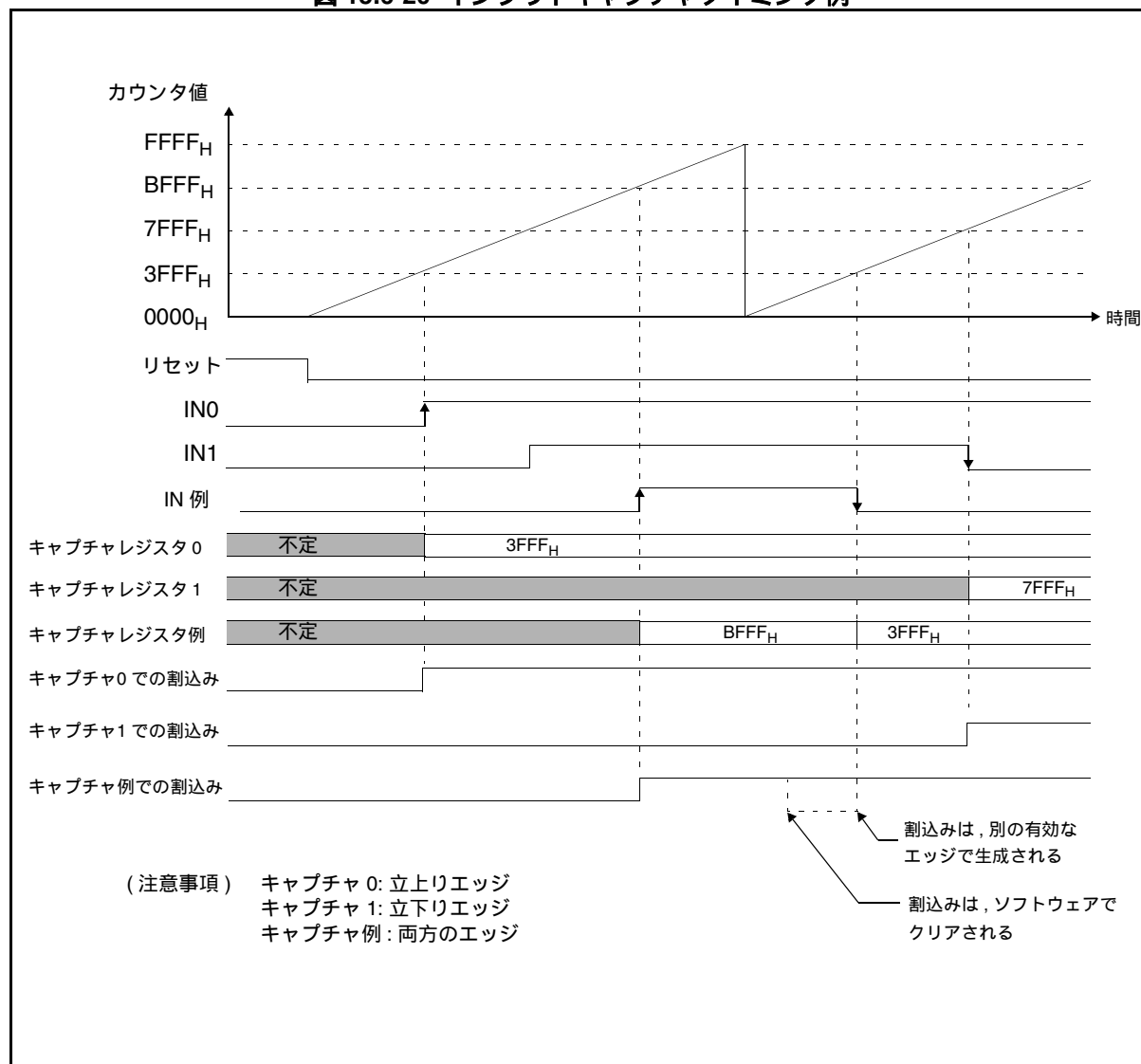


15.6.3 16 ビットインプットキャプチャの動作

インプットキャプチャは、指定された有効なエッジを検出するために使用します。有効なエッジが検出されると、割込みフラグが設定され、16 ビットフリーランタイムの値がキャプチャレジスタへロードされます。

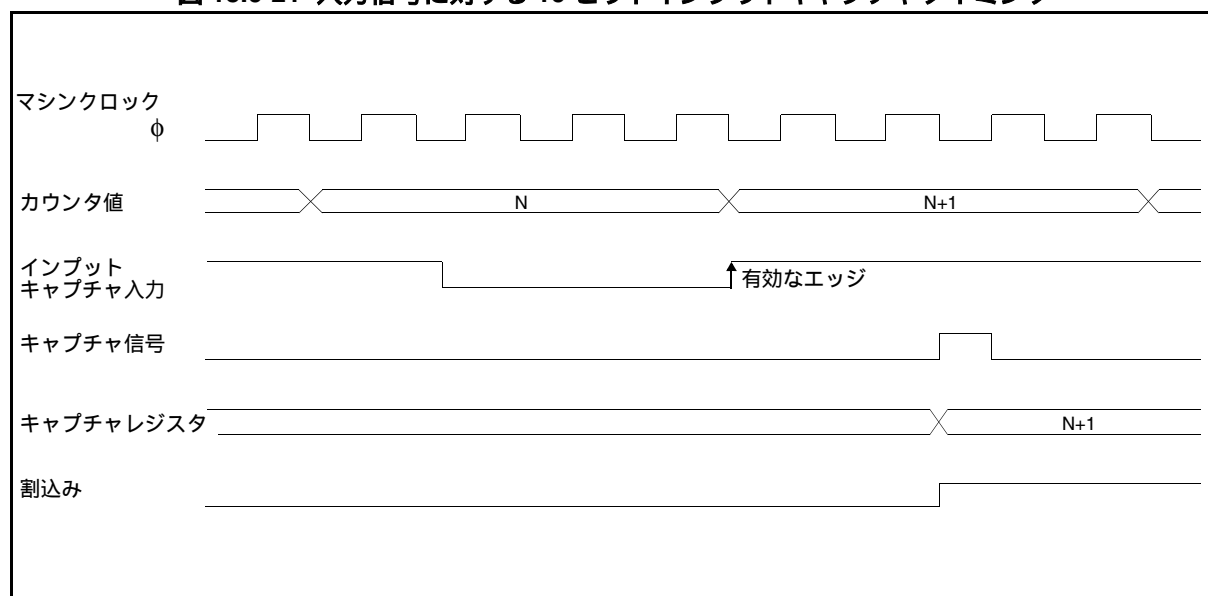
■ 16 ビットインプットキャプチャの動作

図 15.6-20 インプットキャプチャタイミング例



■ 16 ビットインプットキャプチャ入力タイミング

図 15.6-21 入力信号に対する 16 ビットインプットキャプチャタイミング



15.6.4 波形ジェネレータの動作

波形ジェネレータは、リアルタイム出力 (RTO0 ~ RTO5), 16 ビット PPG タイマ 0, 16 ビットタイマ 0/1/2 を使用してさまざまな波形 (デッドタイムを含む) を生成できます。

■ RTO0 ~ RTO5 と GATE の出力状態

表 15.6-1 RTO0 ~ RTO5/GATE 出力状態とビット設定

TMD2	TMD1	TMD0	GTENx	PGENx	RTOx*2	GATE
0	0	0	X	X	リアルタイム出力, RTx	常に "0"
0	0	1	X	0	リアルタイム出力, RTx	OR(RTx & GTENx)
0	0	1	0	1	RTx が "H" の期間に PPG0 のパルスを出力	常に "0"
0	0	1	1	1	RTx が "H" の期間に GATE 信号により起動された PPG0 のパルスを出力	OR(RTx)
0	1	0	X	0	RTx の立上りエッジにより 16 ビットタイマ 0 がアンダフローするまで "H" を出力 (x = 0, 1)	OR(RTOx & GTENx)
					RTx の立上りエッジにより 16 ビットタイマ 1 がアンダフローするまで "H" を出力 (x = 2, 3)	
					RTx の立上りエッジにより 16 ビットタイマ 2 がアンダフローするまで "H" を出力 (x = 4, 5)	
0	1	0	0	1	RTx の立上りエッジにより 16 ビットタイマ 0 がアンダフローするまで PPG0 のパルスを出力 (x = 0, 1)	常に "0"
					RTx の立上りエッジにより 16 ビットタイマ 1 がアンダフローするまで PPG0 のパルスを出力 (x = 2, 3)	
					RTx の立上りエッジにより 16 ビットタイマ 2 がアンダフローするまで PPG0 のパルスを出力 (x = 4, 5)	
0	1	0	1	1	RTx の立上りエッジにより 16 ビットタイマ 0 がアンダフローするまで GATE 信号により起動された PPG0 のパルスを出力 (x = 0, 1)	OR(RTx/y/z の立上りエッジによりタイマ 0/1/2 がアンダフローするまで "H" を出力 x = 0, 1 y = 2, 3 z = 4, 5)
					RTx の立上りエッジにより 16 ビットタイマ 1 がアンダフローするまで GATE 信号により起動された PPG0 のパルスを出力 (x = 2, 3)	
					RTx の立上りエッジにより 16 ビットタイマ 2 がアンダフローするまで GATE 信号により起動された PPG0 のパルスを出力 (x = 4, 5)	
1	0	0	X	X	RT1 でノンオーバーラップ信号を生成 (x = 0, 1)*1	常に "0"
					RT3 でノンオーバーラップ信号を生成 (x = 2, 3)*1	
					RT5 でノンオーバーラップ信号を生成 (x = 4, 5)*1	
1	1	1	0	X	PPG0 でノンオーバーラップ信号を生成	常に "0"
1	1	1	1	X	GATE 信号により起動された PPG0 でノンオーバーラップ信号を生成	OR(RTx)
その他					常に "0"	常に "0"

*1: ノンオーバーラップ信号を生成するには、必ず RT1/RT3/RT5 に対して 2 チャンネルモード (コンペア制御レジスタ (OCS1/OCS3/OCS5) の CMOD : bit12 = 1) を選択してください。

*2: RTO0/RTO1 は 16 ビットタイマ制御レジスタ下位 (DTCR0) の TMD2 ~ TMD0 : bit2 ~ bit0 により, RTO2/RTO3 は (DTCR1) 上位レジスタの TMD2 ~ TMD0 : bit10 ~ bit8 により, RTO4/RTO5 は下位レジスタ (DTCR2) の TMD2 ~ TMD0 : bit2 ~ bit0 により制御されます。

■ PPG0 出力制御

RT00 ~ RT05 端子への PPG0 出力は, PPG 出力制御 / インプットキャプチャ状態制御レジスタ上位 (PICS01) の PGEN5 ~ PGEN0: bit15 ~ bit10 で許可にできます。

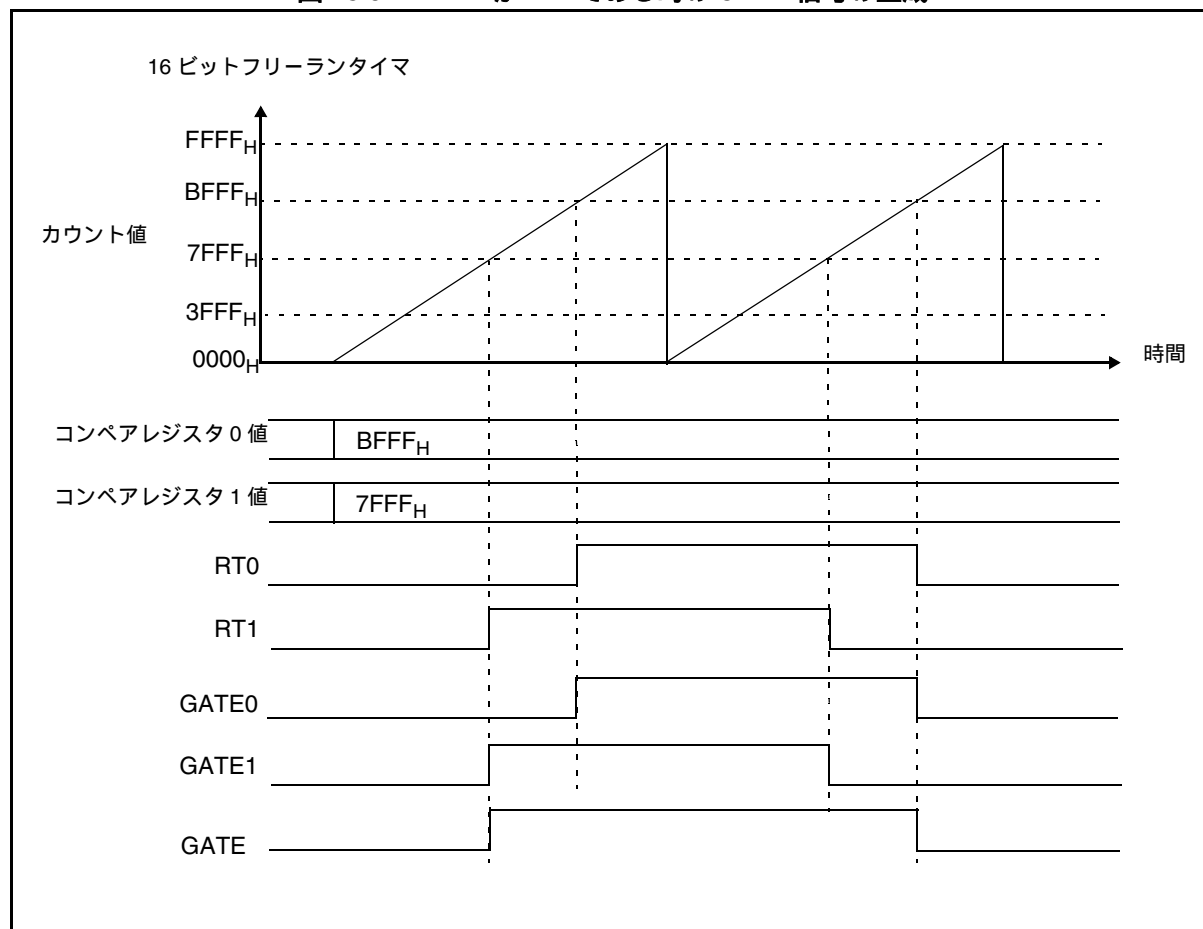
■ ゲートトリガされた PPG0 出力

波形ジェネレータではリアルタイム出力 RT0 ~ RT5 により GATE 信号を生成でき, 16 ビットタイマ 0/1/2 では PPG0 カウントをトリガとして動作できます。1 つの 16 ビットタイマ 0/1/2 で 2 つのリアルタイム出力 (RT0/RT2/RT4, RT1/RT3/RT5) が操作され, 6 つの別々のゲート信号が生成されます。これら 6 つのゲート信号は論理和がとられて GATE 信号を生成し, PPG0 カウントのトリガとなります。

また, PGEN0 ~ PGEN5 信号を使用すると, PPG0 のみを使用することで RT00 ~ RT05 端子に 6 つの異なる波形を出力できます。

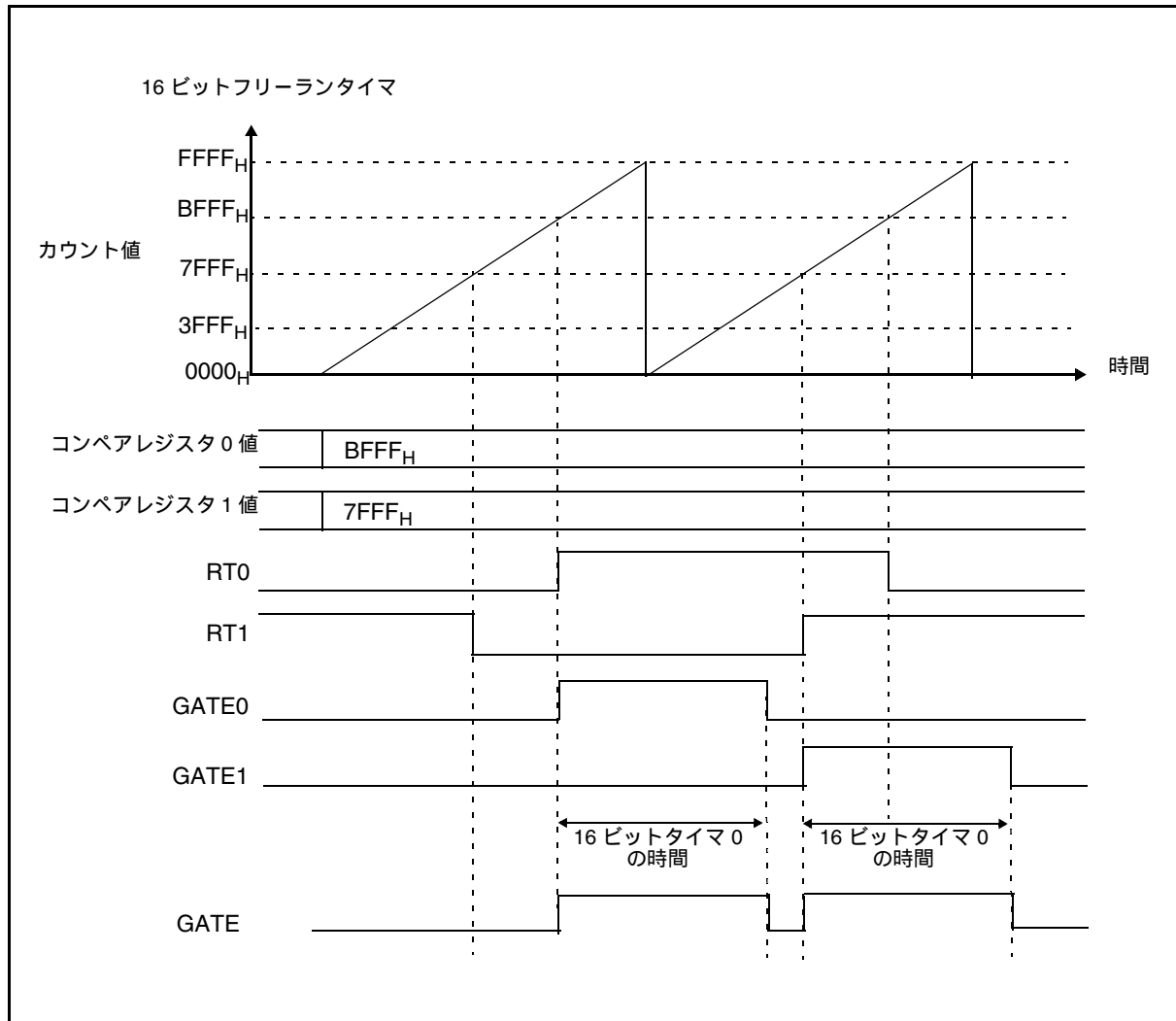
■ GTENxがアクティブであり各RTxが"H"であるとき(16ビットタイマ制御レジスタ (DTCR0 ~ DTCR2) の TMD2 ~ TMD0(上位は bit10 ~ bit8, 下位は bit2 ~ bit0) が "001_B" または "111_B") の GATE 信号生成

図 15.6-22 RTx が "H" である時の GATE 信号の生成



■ GTENx がアクティブ(DTCR0 ~ DTCR2 レジスタの TMD2 ~ TMD0 = 010_B)である
ときの RTx 立上りエッジから 16 ビットタイマ 0/1/2 アンダフローまでにおける
GATE 信号の生成

図 15.6-23 RTx 立上りエッジから 16 ビットタイマアンダフローまでにおける
GATE 信号の生成



< 注意事項 >

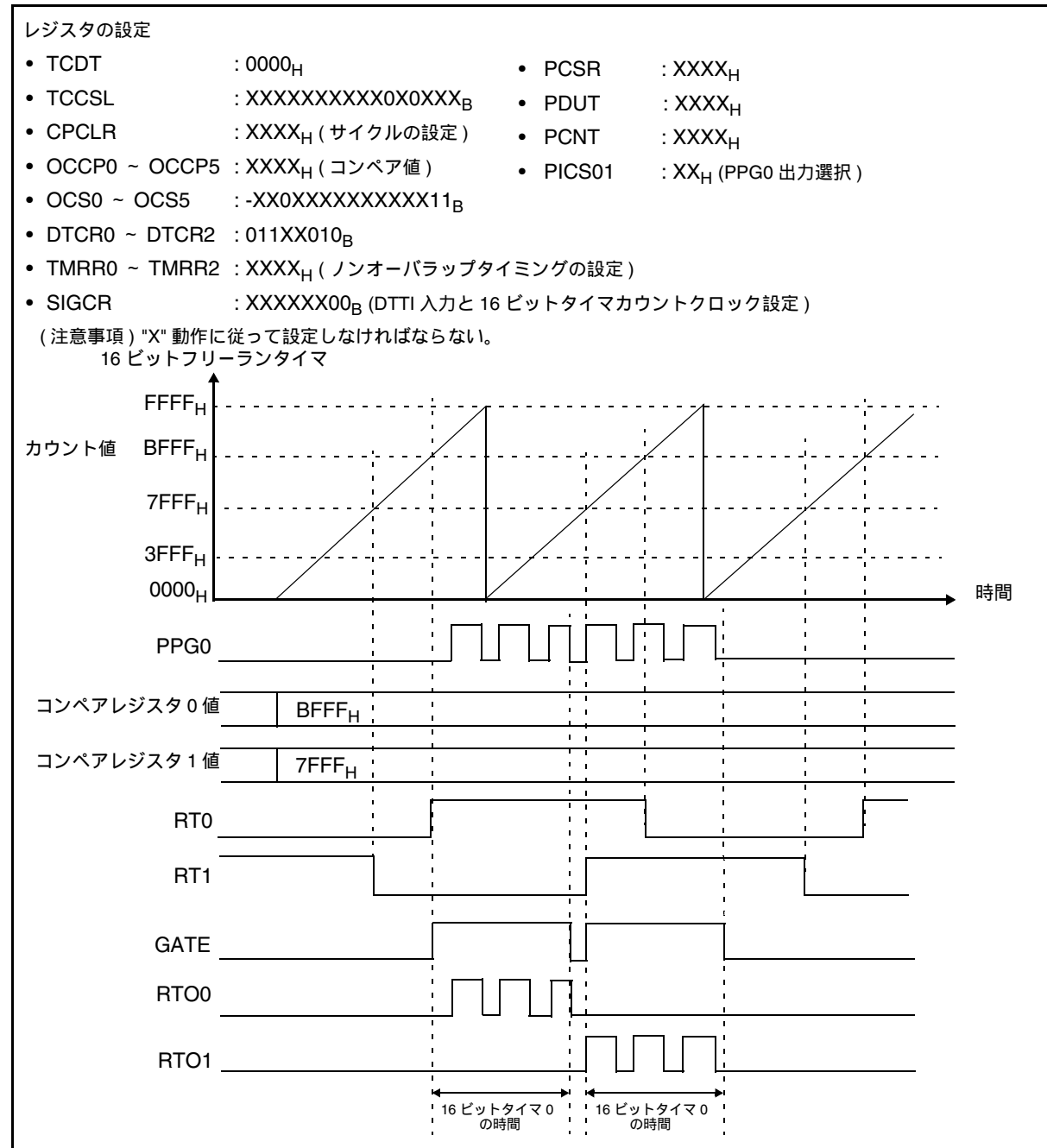
各 16 ビットタイマは、2 つの RT に対して使用されます。すなわち、16 ビットタイマ 0 は RT0 と RT1 に対して使用され、16 ビットタイマ 1 は RT2 と RT3 に対して使用され、16 ビットタイマ 2 は RT4 と RT5 に対して使用されます。したがって、RT を使用して、すでに動作中のタイマの起動を試みてはなりません。このような試みを行った場合は、GATE 信号出力が拡張され、その結果、誤動作が発生する場合があります。

15.6.5 タイマモードの動作

RT0 ~ 5 端子の立上りエッジが検出されると、16 ビットタイマに値がリロードされて 16 ビットタイマがダウンカウントを開始します。PPG タイマ 0 は 16 ビットタイマでアンダフローが発生するまで RTO0 ~ RTO5 端子へ出力し続けます。

■ RT 立上りエッジから 16 ビットタイマアンダフローまでにおける PPG0 出力パルス
(DTCR0/1/2 レジスタの TMD2 ~ TMD0 (上位は bit10 ~ bit8, 下位は bit2 ~ bit0) = 010_B)

図 15.6-24 TMD2 ~ TMD0

(上位は bit10 ~ bit8, 下位は bit2 ~ bit0) が "010_B" のときに生成される波形

< 注意事項 >

各 16 ビットタイマは、2 つの RT に対して使用されます。すなわち、16 ビットタイマ 0 は RT0 と RT1 に対して使用され、16 ビットタイマ 1 は RT2 と RT3 に対して使用され、16 ビットタイマ 2 は RT4 と RT5 に対して使用されます。したがって、RT を使用して、すでに動作中の PPG0 の起動を試みてはなりません。このような試みを行った場合は、GATE 信号出力が拡張され、その結果、誤動作が発生する場合があります。

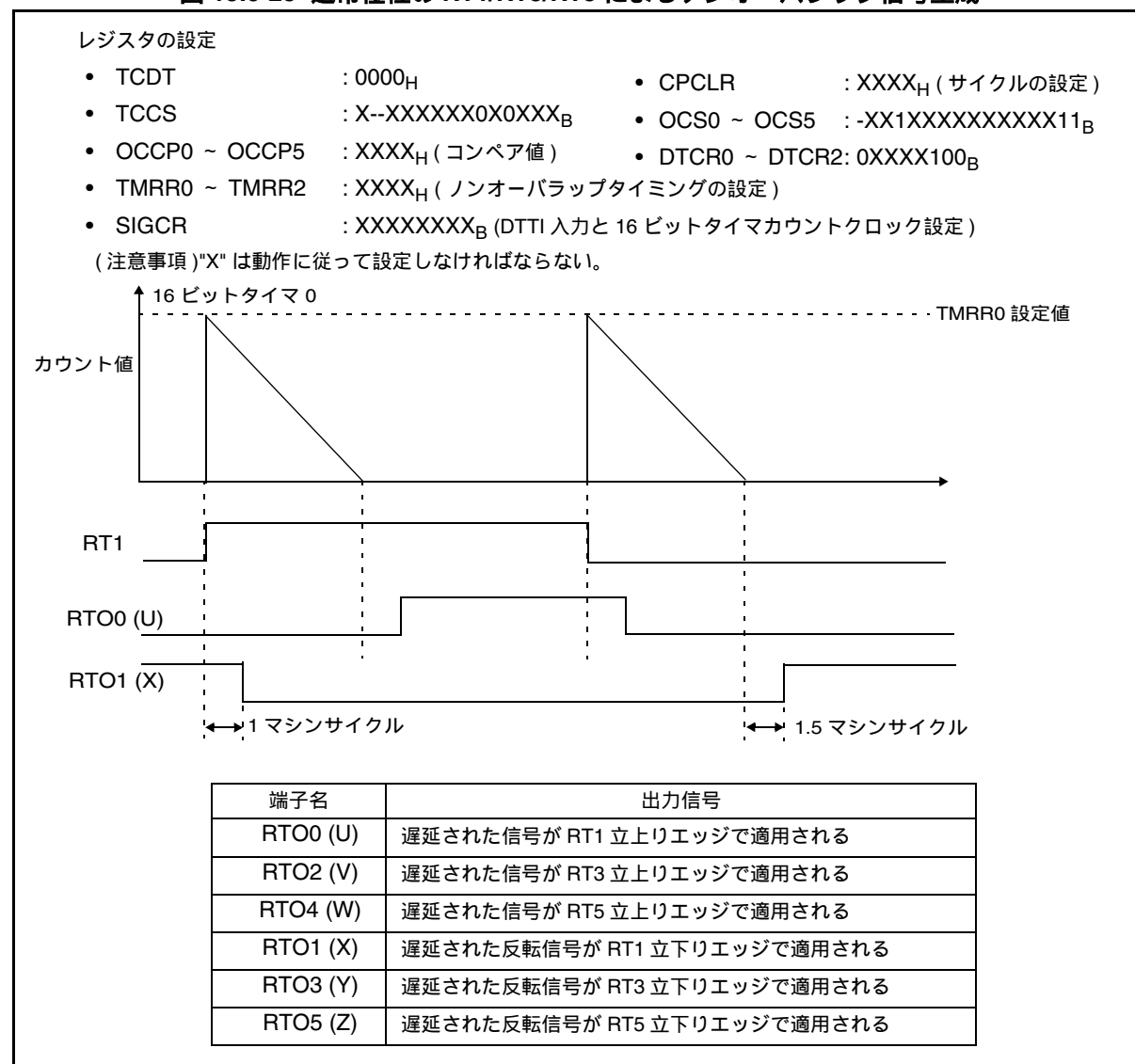
15.6.6 デッドタイムタイマモード時の動作

デッドタイムジェネレータは、リアルタイム出力 (RT1/RT3/RT5), もしくは PPG0 タイマパルス出力を入力し、外部端子 (RTO0 ~ RTO5) へノンオーバーラップ信号 (反転信号) を出力します。

■ 通常極性の RT1/RT3/RT5 による ノンオーバーラップ信号生成 (16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2) の TMD2 ~ TMD0 (上位は bit10 ~ bit8, 下位は bit2 ~ bit0) = 100_B)

DTCR0 ~ DTCR2 レジスタの DMOD が "0" (通常極性) である ノンオーバーラップ信号を選択すると、16 ビットタイマレジスタ (TMRR0 ~ TMRR2) に設定されている ノンオーバーラップ時間に相当する遅延が適用されます。この遅延は、RT1/RT3/RT5 端子の立上りエッジまたは立下りエッジで適用されます。RT1/RT3/RT5 パルス幅が、設定されている ノンオーバーラップ時間より小さい場合は、16 ビットタイマはその次の RT エッジの TMRR0 ~ TMRR2 レジスタ値からダウンカウントを再開します。

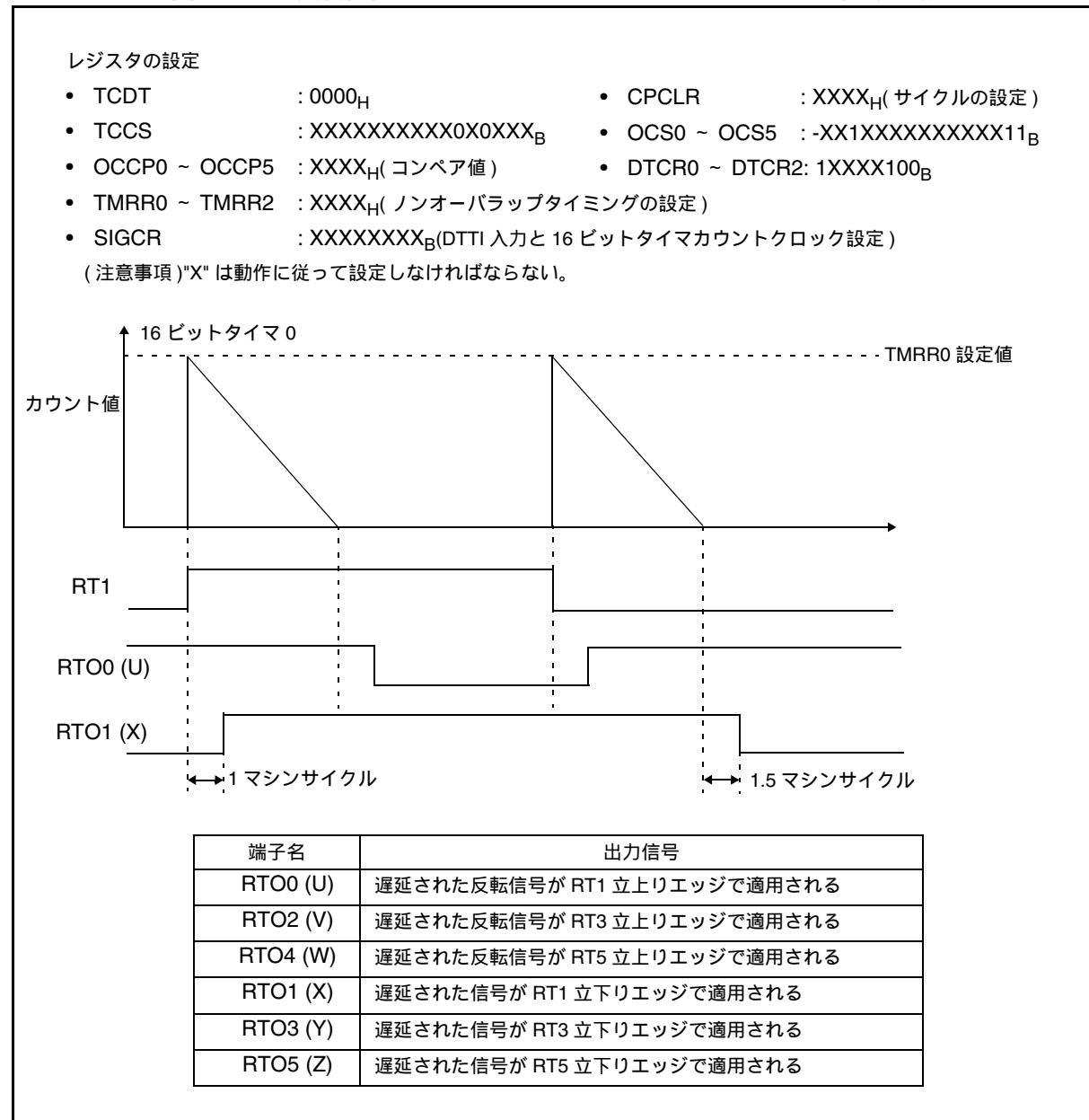
図 15.6-25 通常極性の RT1/RT3/RT5 による ノンオーバーラップ信号生成



■ 反転極性の RT1/RT3/RT5 による ノンオーバーラップ信号生成 (16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2) の TMD2 ~ TMD0(上位は bit10 ~ bit8, 下位は bit2 ~ bit0) = 100_B)

DTCR0 ~ DTCR2 レジスタの DMOD(上位は bit15, 下位は bit7) が "1"(反転極性)であるノンオーバーラップ信号を選択すると, 16 ビットタイマレジスタ (TMRR0 ~ TMRR2) に設定されているノンオーバーラップ時間に相当する遅延が適用されます。この遅延は, RT1/RT3/RT5 の立上りエッジまたは立下りエッジで適用されます。RT1/RT3/RT5 パルス幅が, 設定されているノンオーバーラップ時間より小さい場合は, 16 ビットタイマはその次の RT エッジの TMRR0 ~ TMRR2 値からダウンカウントを再開します。

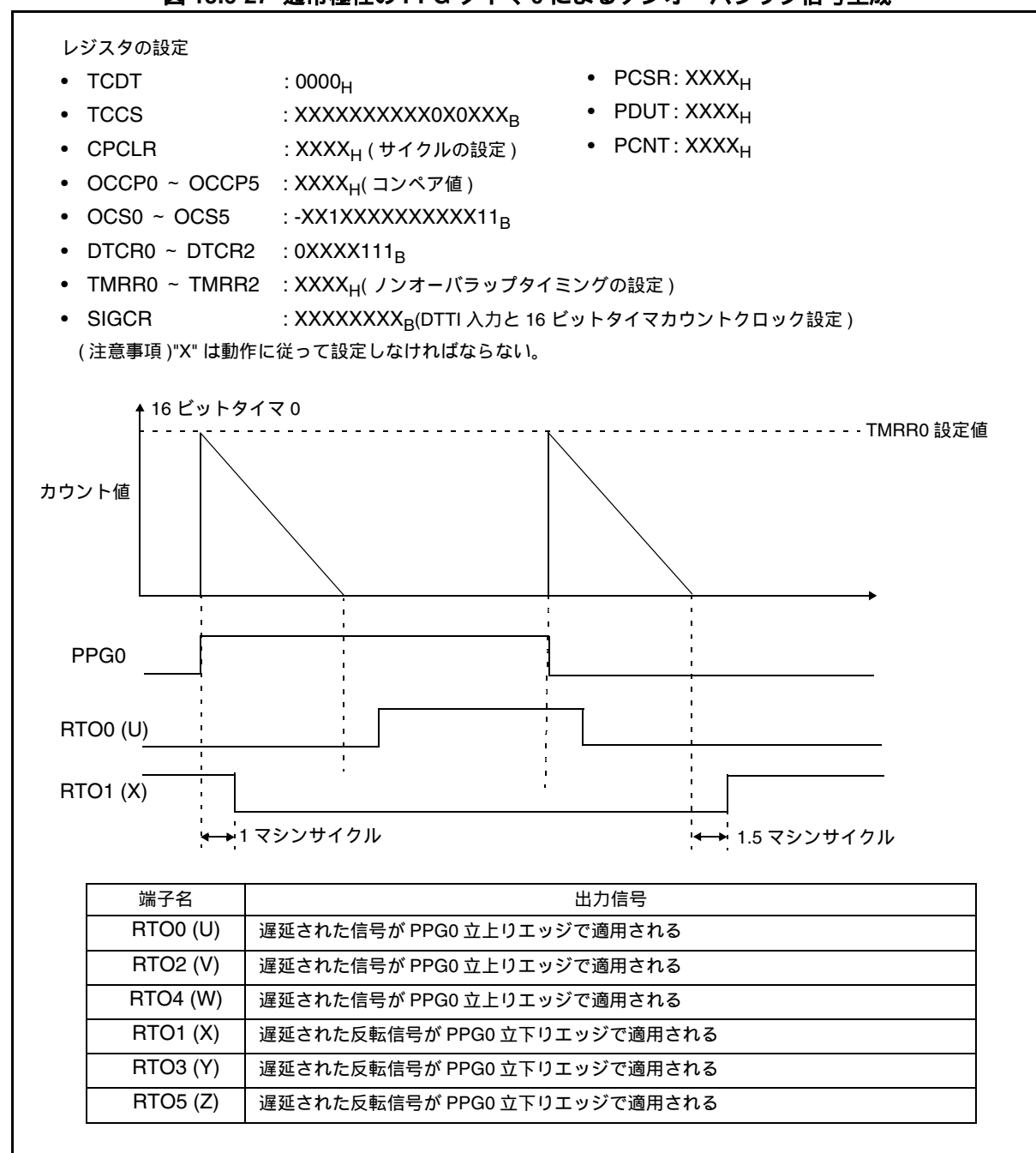
図 15.6-26 反転極性の RT1/RT3/RT5 による ノンオーバーラップ信号生成



■ 通常極性の PPG によるノンオーバーラップ信号生成 (16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2) の TMD2 ~ TMD0(上位は bit10 ~ bit8, 下位は bit2 ~ bit0) = 111_B)

DTCR0 ~ DTCR2 レジスタの DMOD(上位は bit15, 下位は bit7) が "0"(通常極性)であるノンオーバーラップ信号を選択すると, 16 ビットタイマレジスタ (TMRR0 ~ TMRR2) に設定されているノンオーバーラップ時間に相当する遅延が適用されます。この遅延は, PPG0 タイマパルス信号または反転信号の立上りエッジで適用されます。PPG タイマパルス幅が, 設定されているノンオーバーラップ時間より小さい場合は, 16 ビットタイマは PPG0 パルスのその次のエッジの TMRR0 ~ TMRR2 値からダウンカウントを再開します。

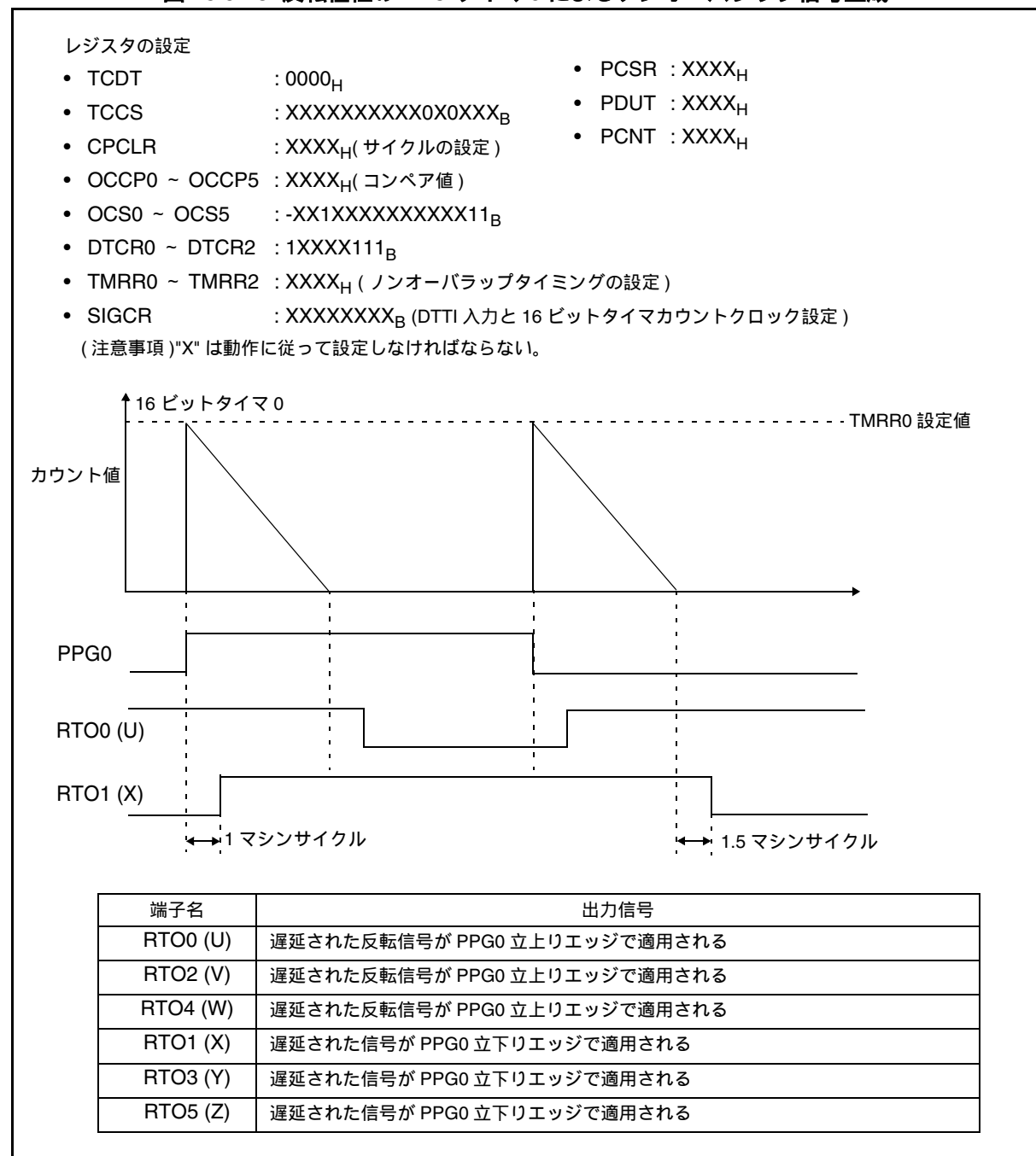
図 15.6-27 通常極性の PPG タイマ 0 によるノンオーバーラップ信号生成



■ 反転極性の PPG によるノンオーバーラップ信号生成 (16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2) の TMD2 ~ TMD0(上位は bit10 ~ bit8, 下位は bit2 ~ bit0) = 111_B)

DTCR0 ~ DTCR2 ビットの DMOD(上位は bit15, 下位は bit7) が "1"(反転極性)であるノンオーバーラップ信号を選択すると, 16 ビットタイマレジスタ (TMRR0 ~ TMRR2) に設定されているノンオーバーラップ時間に相当する遅延が適用されます。この遅延は, PPG0 タイマパルス信号または反転信号の立上りエッジで適用されます。PPG0 タイマパルス幅が, 設定されているノンオーバーラップ時間より小さい場合は, 16 ビットタイマは PPG0 パルスのその次のエッジの TMRR0 ~ TMRR2 値からダウンカウントを再開します。

図 15.6-28 反転極性の PPG タイマ 0 によるノンオーバーラップ信号生成



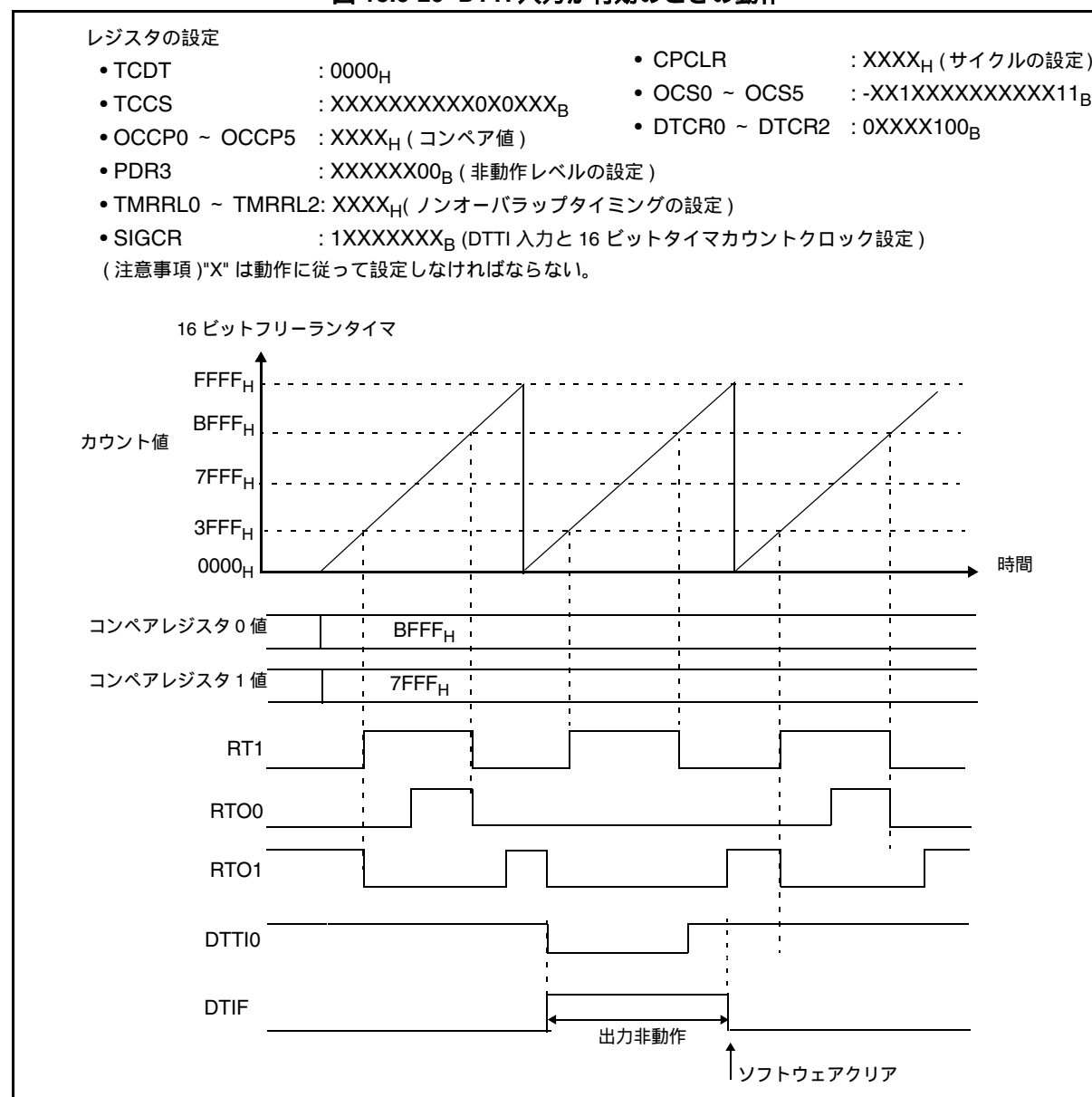
15.6.7 DTTI 端子制御の動作

波形制御レジスタ (SIGCR) の DTIE : bit15 に "1" を設定すると, RTO0 ~ RTO5 出力を DTTI 端子で制御できます。DTTI 端子の "L" レベルが検出されると, RTO0 ~ RTO5 出力は, 割込みフラグ (SIGCR レジスタの DTIF : bit14) がクリアされるまで非動作レベルに固定されます。RTO0 ~ RTO5 の非動作レベルは, これらの端子を共用しているポートデータレジスタ (PDR8) を使ってソフトウェア的に設定できます。

■ DTTI 端子入力への動作

DTTI 端子入力の "L" が検出された場合でも, タイマは波形ジェネレータが動作している間は動作を継続しますが, 波形は外部端子 P82/RTO0 ~ P87/RTO5 へは出力されません。

図 15.6-29 DTTI 入力があるときの動作



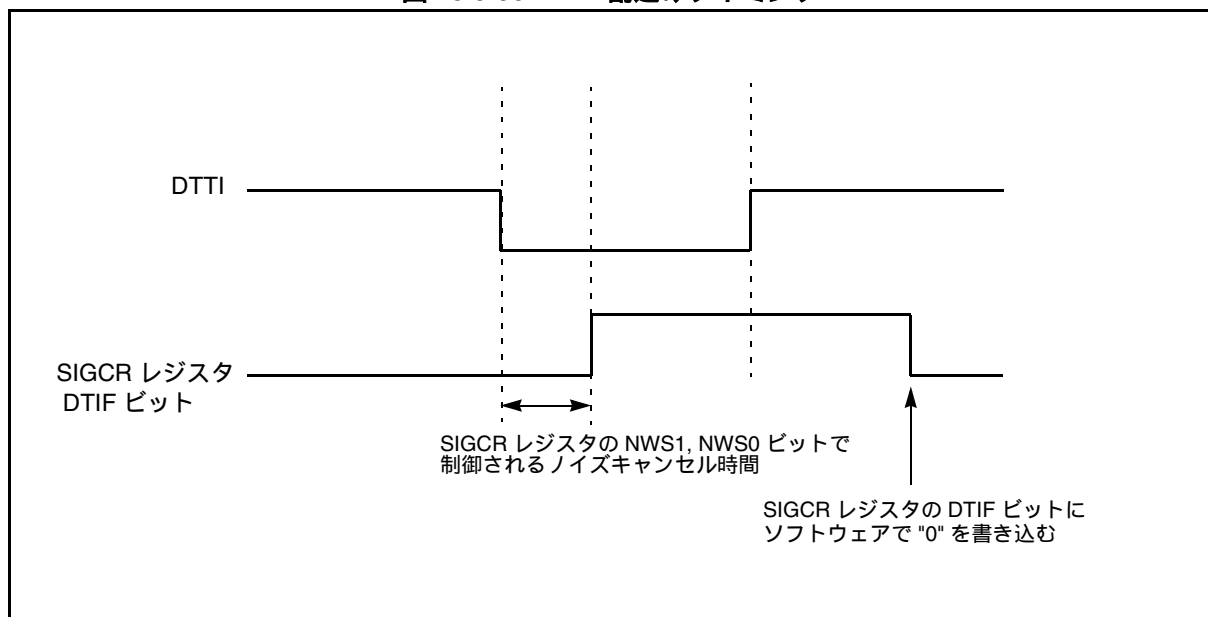
■ DTTI 端子ノイズキャンセル機能

波形制御レジスタ (SIGCR) の NRSL : bit13 に "1" を設定すると、DTTI 端子入力のノイズキャンセル機能が有効になります。ノイズキャンセル機能が有効になると、出力端子 (RTO0 ~ RTO5) を非動作レベルに固定するために要する時間が 4, 8, 16 または 32 マシンサイクル (SIGCR レジスタの NWS1, NWS0 : bit9, bit8 で選択) だけ遅延します。ノイズキャンセル回路はリソースを使用するので、発振が停止するモード(停止モードなど)時において DTTI 入力が無効になった場合でも入力は無効になります。

■ DTTI 割込み

DTTI の "L" レベルが検出されると、ノイズキャンセル時間が経過した後で DTTI 割込みフラグ (SIGCR レジスタの DTIF : bit14) に "1" が設定され、割込み要求は割込みコントローラへ送信されます。

図 15.6-30 DTTI 割込みタイミング



< 注意事項 >

- ノイズキャンセル時間内に SIGCR レジスタの NWS1, NWS0 ビットの値が変化した場合は、さらに大きな (NWS1, NWS0) ノイズサイクル値が有効になります。
- SIGCR レジスタの DTIF : bit14 は、ソフトウェアでのみクリアできます。

15.7 多機能タイマ使用上の注意

多機能タイマ使用上の注意を以下に示します。

■ 16 ビットフリーランタイム使用上の注意

● プログラムによる設定上の注意

- リセットを実行すると、タイマが有効 (TCCSL レジスタの STOP : bit5 = 0) になった後、その次のカウントクロックでタイマ値が "0000_H" になり、ゼロ検出割込みフラグに "1" が設定されます。
- タイマモードビット (TCCSL レジスタの MODE : bit4) はバッファを持っているので、その次のカウントサイクルでタイマモードの変更は有効になります。ゼロ検出割込みは、タイマモードがアップカウントモードからアップダウンカウントモードに変わると必ず生成されます。
- ソフトウェアクリア (TCCSL レジスタの SCLR : bit3 = 1) はタイマを初期化しますが、ゼロ検出割込みを生成しません。

● 割込みの注意

- タイマ状態制御レジスタ上位 (TCCSH) の IRQZF : bit14 に "1" を設定し、次に割込み要求を許可にすると (TCCSH レジスタの IRQZE : bit13 = 1)、制御は割込み処理から戻ることができません。IRQZF : bit14 は、必ずクリアしてください。
- タイマ状態制御レジスタ上位 (TCCSH) の ICLR : bit9 に "1" を設定し、次に割込み要求を許可にすると (TCCSH レジスタの ICRE : bit8 = 1)、制御は割込み処理から戻ることができません。ICLR : bit9 は、必ずクリアしてください。
- 16 ビットフリーランタイムは割込みベクタをほかのリソースと共用しますので、割込みを使用する場合は、割込み処理ルーチンで割込み要因を注意深くチェックしなければなりません。

また、16 ビットフリーランタイムで EI²OS を使用する場合は、共用リソース割込みを禁止にしなければなりません。

■ 16 ビットアウトプットコンペアの使用上の注意

● 割込みの注意

- コンペア制御レジスタ (OCS0/OCS2/OCS4) の IOP1, 0 : bit7, bit6 に "1" を設定し、次に割込み要求を許可にすると (OCS0/OCS2/OCS4 レジスタの IOE1, 0 : bit6, bit5 = 1)、制御は割込み処理から戻ることができません。IOP0, IOP1 ビットは、必ずクリアしてください。
- 16 ビットアウトプットコンペアは、割込みベクタをほかのリソースと共用するので、割込みを使用する場合は、割込み処理ルーチンで割込み要因を注意深くチェックしなければなりません。

また、16 ビットアウトプットコンペアで EI²OS を使用する場合は、共用リソース割込みを禁止にしなければなりません。

■ 16 ビットインプットキャプチャ使用上の注意

● 割込みの注意

- インプットキャプチャ状態制御レジスタ下位 (PICSL01/ICSL23) の ICP3, ICP2 (共に bit7, bit6) に "1" を設定し、次に割込み要求を許可にすると (PICSL01/ICSL23 レジスタの ICE3, ICE2 (共に bit5, bit4) = 1), 制御は割込み処理から戻ることができません。ICP3, ICP2 (共に bit7, bit6) は、必ずクリアしてください。
- インプットキャプチャ端子 (IN) レベルが、ICP3, ICP2 ビットが設定されてから割込みルーチンが処理されるまでの間に切換わると、有効エッジ指示ビット (ICSH23 レジスタの IEI3, IEI2 : bit9, bit8 または PICSH01 レジスタの IEI1, IEI0 : bit9, bit8) は、検出された最新のエッジを示します。
- 16 ビットインプットキャプチャは割込みベクタをほかのリソースと共用するので、割込みを使用する場合は、割込み処理ルーチンで割込み要因を注意深くチェックしなければなりません。

また、16 ビットインプットキャプチャで EI²OS を使用する場合は、共用リソース割込みを禁止にしなければなりません。

■ 波形ジェネレータ使用上の注意

● プログラムによる設定上の注意

- 波形ジェネレータが動作中 (DTCR0/1/2 レジスタの TMD2 ~ TMD0 が "001_B", "010_B", "100_B" または "111_B") に、16 ビットタイマ制御レジスタ (DTCR0/1/2) の TMD2 ~ TMD0 (上位は bit10 ~ bit8, 下位は bit2 ~ bit0) ビット値を変更する場合は、トリガソースおよび 16 ビットタイマがカウント中でないことを必ず確認してください。この操作を行わない場合は、以前のトリガでスケジュールされた出力が原因となり予期しない波形が RTO 端子から出力されます。ただし、RTO 出力は、タイマでアンダフローが発生したり、新しいトリガソースで再トリガされたりすると、正常動作に戻ります。

トリガソースとは、DTCR0 ~ DTCR2 レジスタの TMD2 ~ TMD0 (上位は bit10 ~ bit8, 下位は bit2 ~ bit0) が "001_B" の場合は "RT の "H" レベル" であり、TMD2 ~ TMD0 ビットが "010_B" の場合は "RT の立上りエッジ" であり、TMD2 ~ TMD0 ビットが "100_B" の場合は "RT の立上りエッジまたは立下りエッジ" であり、TMD2 ~ TMD0 ビットが "111_B" の場合は "PPG0 の立上りエッジまたは立下りエッジ" です。

例えば、TMD2 ~ TMD0 ビットが "100_B" から "111_B" へ変更すると、下記の手順を実行できます。

- (1) 16 ビットタイマレジスタ (TMRR0 ~ TMRR2) に "0001_H" のような非常に小さな値を設定する。
 - (2) RT1/RT3/RT5 の出力を "L" または "H" に設定し、タイマ 0/1/2 でアンダフローが発生するまで待つ。
 - (3) モードビット (TMD2 ~ TMD0), および対応する設定を変更する。
 - (4) 修正された出力波形が 1 マシンサイクル後 RTO 端子に現れる。
- タイマがカウント中に 16 ビットタイマレジスタ (TMRR0 ~ TMRR2) に値が書き込まれると、この新しい値は次のタイマトリガ時に有効になります。タイマレジスタへアクセスする場合は、必ずワード転送命令 (MOVW A, dir など) をご使用ください。

- タイマがカウントしていない場合のみ，波形制御レジスタ (SIGCR) の DCK2 ~ DCK0 : bit12 ~ bit10 を変更してください。
- ノイズキャンセル機能が無効になった場合のみ (SIGCR : NRSL = 0), 波形制御レジスタ (SIGCR) の NWS1, NWS0 : bit9, bit8 を変更してください。

● 割込みの注意

- 16 ビットタイマ制御レジスタ (DTCR0/1/2) の TMIF(上位は bit12, 下位は bit4) に "1" を設定し，次に割込み要求を許可にすると (DTCR0/1/2 レジスタの TMIE(上位は bit11, 下位は bit3 = 1), 制御は割込み処理から戻ることができません。TMIF ビットは，必ずクリアしてください。
- 波形制御レジスタ (SIGCR) の DTIF : bit14 に "1" を設定すると，制御は割込み処理から戻ることができません。DTIF ビットは，必ずクリアしてください。
- 波形ジェネレータは割込みベクタをほかのリソースと共用するので，割込みを使用する場合は，割込み処理ルーチンで割込み要因を注意深くチェックしなければなりません。

また，波形ジェネレータで EI²OS を使用する場合は，共用リソース割込みをディセーブルにしなければなりません。

第16章

遅延割込み発生モジュール

遅延割込み発生モジュールの機能と動作について説明します。

- 16.1 遅延割込み発生モジュールの概要
- 16.2 遅延割込み発生モジュールのレジスタ
- 16.3 遅延割込み発生モジュールの動作説明
- 16.4 遅延割込み発生モジュールの使用上の注意

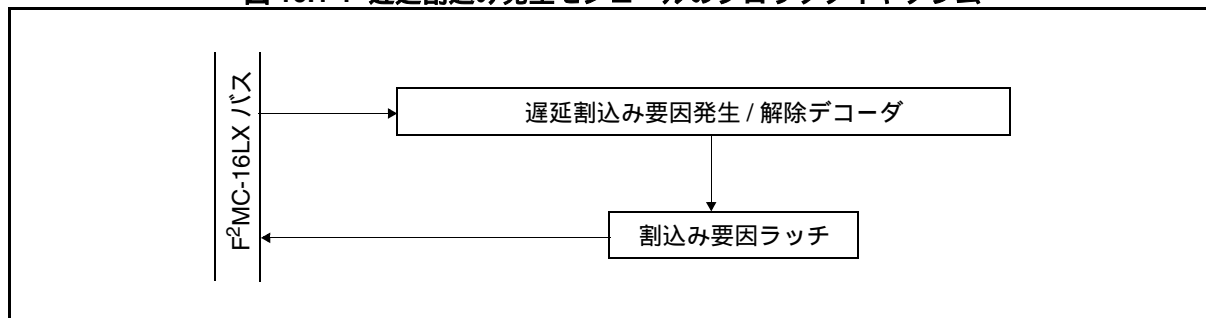
16.1 遅延割込み発生モジュールの概要

遅延割込み発生モジュールは、タスク切換え用の割込みを発生するためのモジュールです。

ソフトウェアで、ハードウェア割込み要求を発生、または解除できます。

■ 遅延割込み発生モジュールのブロックダイアグラム

図 16.1-1 遅延割込み発生モジュールのブロックダイアグラム



16.2 遅延割込み発生モジュールのレジスタ

遅延割込み発生モジュールのレジスタを記述します。

■ 遅延割込み発生モジュールレジスタ (DIRR)

図 16.2-1 遅延割込み発生モジュールレジスタ (DIRR)

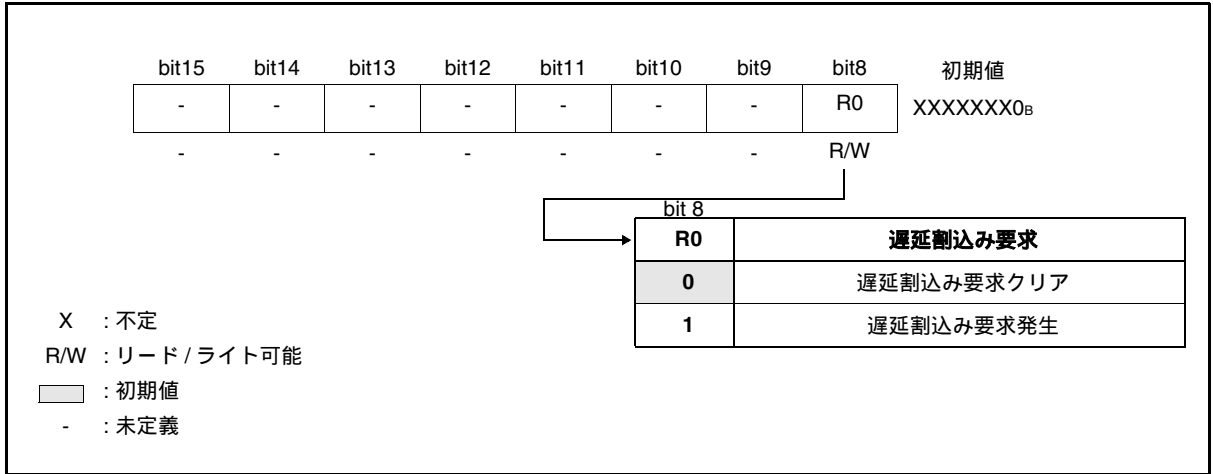


表 16.2-1 遅延割込み要求出力 / 解除レジスタ (DIRR) の機能

ビット名		機能
bit 15 ~ bit 9	予約ビット	"0" および "1" がこれらの予約ビットに書き込まれる可能性があります。動作に影響はありません。
bit 8	R0: 遅延割込み 要求ビット	遅延割込み要求の発生および解除を制御するビットです。 "0" を設定した場合：遅延割込み要求を解除します。 "1" を設定した場合：遅延割込み要求を発生します。 レジスタはリセット時にクリアされます。 "0" および "1" が予約ビット領域に書き込まれる可能性があります。ただし、設定ビットとクリアビット命令を用いて、将来の拡張に備えるためにレジスタにアクセスする必要があります。

16.3 遅延割り込み発生モジュールの動作説明

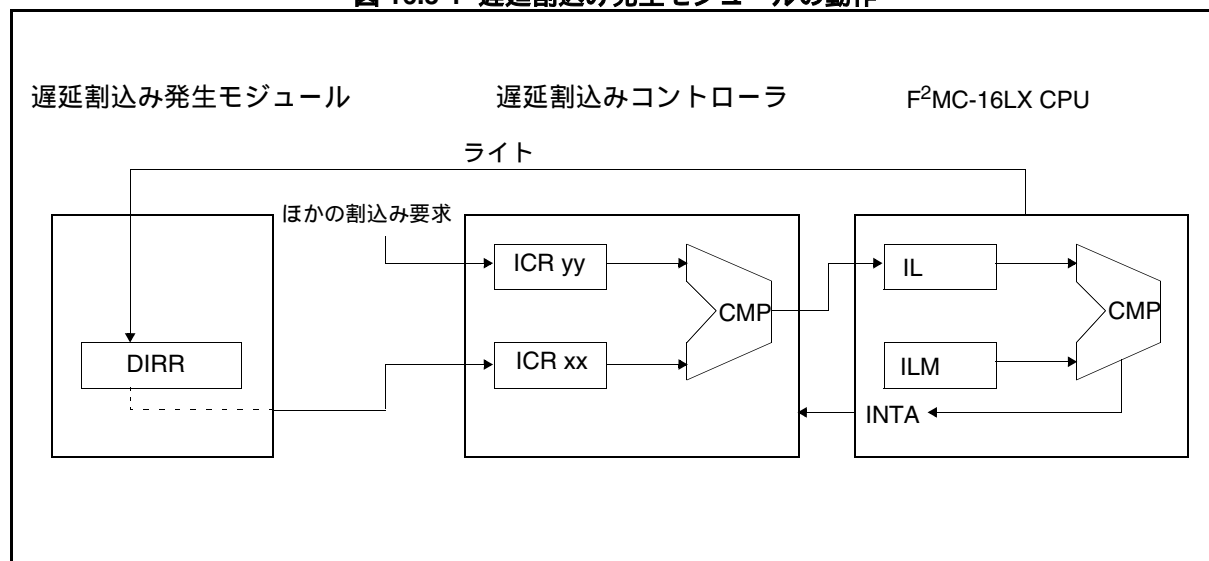
遅延割り込み発生モジュールには、ソフトウェアによって DIRR の R0 ビットに "1" を設定した場合、割り込み要求ラッチが設定され、割り込みコントローラに対して割り込み要求が発生します。

■ 遅延割り込み発生モジュールの動作

- 遅延割り込み発生モジュールレジスタの R0 ビットに "1" を設定した場合、割り込み要求ラッチが設定され、割り込みコントローラに対して、割り込み要求が発生します。
- 割り込みコントローラで、ほかの割り込み要求よりも割り込み優先順位が高いまたはほかの割り込み要求がないと判定された場合は、F²MC-16LX CPU に対して割り込み要求が発生します。
- F²MC-16LX CPU 側では、コンディションコードレジスタの割り込みレベルマスクビット (CCR: ILM) と割り込み要求が比較され、割り込み要求レベルが ILM より高かった場合は、現在実行中の命令の終了後に、遅延割り込み処理が実行されます。
- 割り込み処理ルーチン内で、ユーザプログラムの R0 ビットに "0" を設定して、割り込み要求を解除し、タスクの切換えを行います。

図 16.3-1 に遅延割り込み発生モジュールの動作を示します。

図 16.3-1 遅延割り込み発生モジュールの動作



16.4 遅延割込み発生モジュールの使用上の注意

遅延割込み発生モジュールを使用する場合は、以下の点に注意してください。

■ 遅延割込み要求ラッチの使用上の注意

遅延割込み発生モジュールレジスタのR0ビットに"1"を設定した場合、割込み要求ラッチが設定され、"0"を設定した場合、クリアされます。割込み処理ルーチン内で、ソフトウェアによって要因をクリアするように設定した場合、割込み処理から復帰したときに再度割込み処理を起動します。

第17章

DTP/ 外部割込み

DTP/ 外部割込みの機能と動作について説明します。

17.1 DTP/ 外部割込みの概要

17.2 DTP/ 外部割込みのブロックダイアグラム

17.3 DTP/ 外部割込みの端子

17.4 DTP/ 外部割込み回路のレジスタ

17.5 DTP/ 外部割込みの動作説明

17.6 DTP/ 外部割込みの使用上の注意

17.1 DTP/ 外部割込みの概要

DTP/ 外部割込みは、外部の周辺装置と F²MC-16LX CPU の間にあり、外部の周辺装置が発生する割込み要求、またはデータ転送要求を CPU に伝達することで割込み要求を発生します。また、拡張インテリジェント I/O サービス EI²OS を起動することもできます。

■ DTP/ 外部割込み機能

外部の周辺装置が発生する割込み要求を検出し、周辺機能の割込みと同じ手順で CPU に割込み要求を出力し、割込みを発生させます。また、拡張インテリジェント I/O サービス (EI²OS) を起動させることもできます。

割込み要求が CPU に受け付けられたとき、割込み制御レジスタで拡張インテリジェント I/O サービス (EI²OS) を禁止に設定している場合は (ICR: ISE=0), 外部割込み機能が有効になり、割込み処理へ分岐します。

EI²OS をあらかじめ許可に設定している場合は (ICR: ISE=1), DTP 機能が有効となり、EI²OS による自動データ転送が行われ、指定回数のデータ転送終了後に割込み処理へ分岐します。

表 17.1-1 に DTP/ 外部割込みの概要について示します。

表 17.1-1 DTP/ 外部割込みの概要

	外部割込み	DTP 機能
入力端子	8 本 (P10/INT0/DTTI ~ P16/INT6, P51/INT7)	
割込み要因	要求レベル設定レジスタ (ELVR) にて、端子ごとにレベルまたはエッジを設定	
	"H" レベル / "L" レベル / 立上りエッジ / 立下りエッジの入力	"H" レベル / "L" レベルの入力
割込み番号	#20(14 _H), #22(16 _H), #25(19 _H), #26(1A _H), #27(1B _H), #28(1C _H)	
割込み制御	DTP/ 外部割込み許可レジスタ (ENIR) による、割込み要求出力の許可または禁止	
割込みフラグ	DTP/ 外部割込み要因レジスタ (EIRR) による、割込み要因の保持	
処理選択	EI ² OS を禁止に設定 (ICR*: ISE=0)	EI ² OS を許可に設定 (ICR*: ISE=1)
処理	外部割込み処理へ分岐	EI ² OS によるデータ自動転送、指定回数処理後、割込み処理へ分岐

*: ICR: 割込み制御レジスタ

■ DTP/ 外部割込み機能および EI²OS の割込み表 17.1-2 DTP/ 外部割込み機能および EI²OS の割込み

チャンネル	割込み番号	割込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ名	アドレス	下位	中位	上位	
INT0/INT1	#20 (14 _H)	ICR04	0000B4 _H	FFFFAC _H	FFFFAD _H	FFFFAE _H	*
INT2/INT3	#22 (16 _H)	ICR05	0000B5 _H	FFFA4 _H	FFFA5 _H	FFFA6 _H	
INT4	#25 (19 _H)	ICR07	0000B7 _H	FFF98 _H	FFF99 _H	FFF9A _H	
INT5	#26 (1A _H)			FFF94 _H	FFF95 _H	FFF96 _H	
INT6	#27 (1B _H)	ICR08	0000B8 _H	FFF90 _H	FFF91 _H	FFF92 _H	
INT7	#28 (1C _H)			FFF8C _H	FFF8D _H	FFF8E _H	

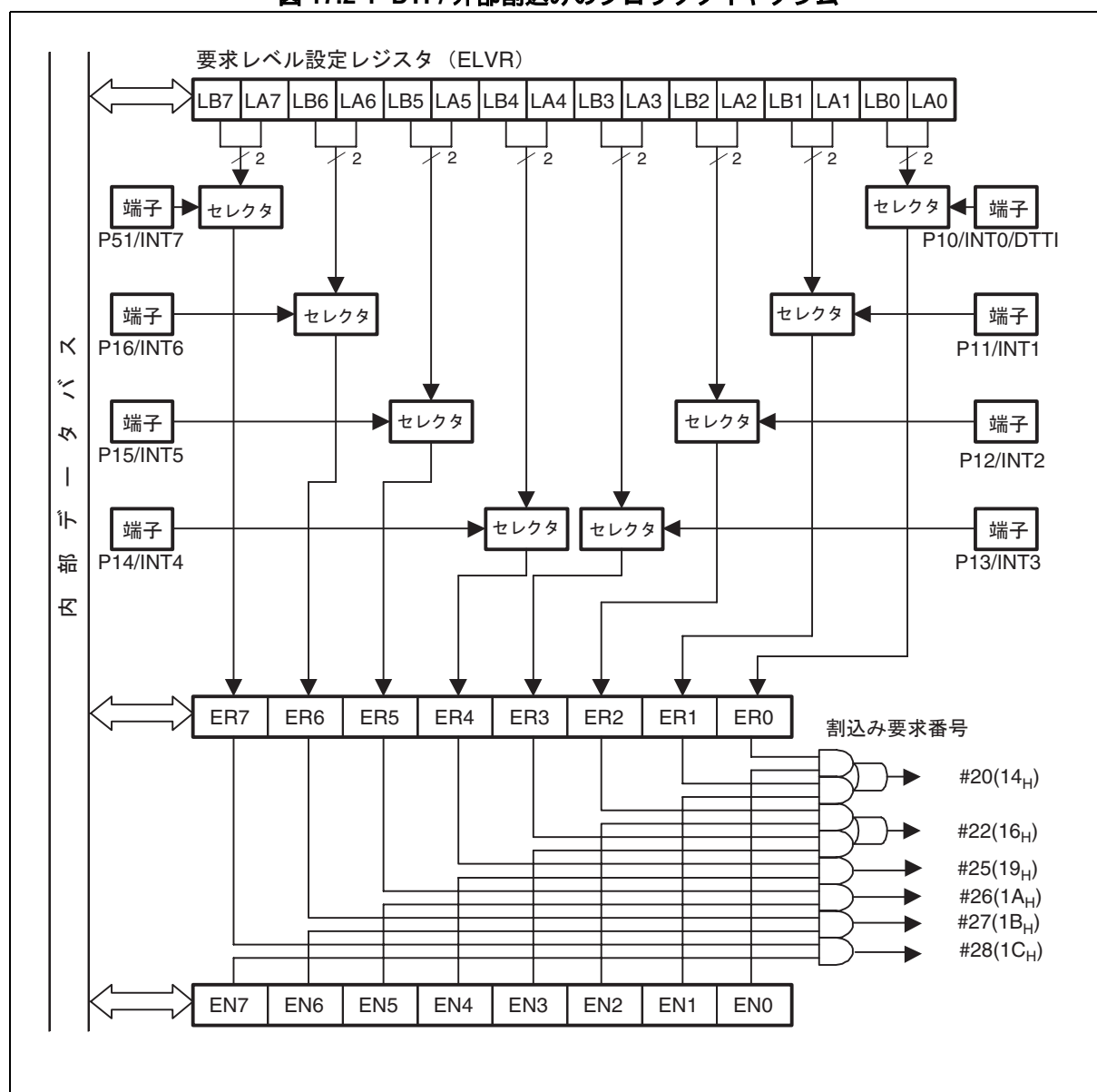
※: : 使用可能, EI²OS 割込みクリア信号により割込み要求フラグがクリアされます。

17.2 DTP/ 外部割込みのブロックダイアグラム

DTP/ 外部割込みは、4 つのブロックによって構成されます。図 17.2-1 にブロックダイアグラムを示します。

■ DTP/ 外部割込みのブロックダイアグラム

図 17.2-1 DTP/ 外部割込みのブロックダイアグラム



- DTP/ 外部割込み入力検出回路

端子ごとに割込み要求レベル設定レジスタ (ELVR) で選択したレベル, またはエッジの検出をして, 端子に対応する DTP/ 外部割込み要因レジスタ (EIRR) の IR ビットに "1" をセットします。

- 要求レベル設定レジスタ (ELVR)

端子ごとに有効レベルまたはエッジを選択するレジスタです。

- DTP/ 外部割込み要因レジスタ (EIRR)

DTP/ 外部割込みの要因を保持するレジスタです。

DTP/ 外部割込み端子に信号の入力があった場合に, 対応する DTP/ 外部割込み要求フラグビットに "1" をセットします。

- DTP/ 外部割込み許可レジスタ (ENIR)

DTP/ 外部割込み許可レジスタ (ENIR) では, 外部周辺装置の DTP/ 外部割込み要求を, 許可または禁止に設定します。

17.3 DTP/ 外部割込みの端子

DTP/ 外部割込みの端子，端子のブロックダイアグラムを示します。

■ DTP/ 外部割込みの端子

DTP/ 外部割込みで使用する端子は，汎用入出力ポートと兼用しています。

端子の機能と DTP/ 外部割込みで使用する場合の設定を表 17.3-1 に示します。

表 17.3-1 DTP/ 外部割込みの端子

端子名	端子機能	入出力形式	プルアップ 設定	スタンバイ 制御	端子の使用に必要な設定
P10/INT0/ DTTI	ポート 1 入出力 / 外部割込み入力 / リソース入出力	CMOS 出力 / CMOS ヒステリシス 入力	選択可能	あり	ポート方向レジスタ (DDR1: bit8 = 0) により " 入力ポート " に設定
P11/INT1					ポート方向レジスタ (DDR1: bit9 = 0) により " 入力ポート " に設定
P12/INT2					ポート方向レジスタ (DDR1: bit10 = 0) により " 入力ポート " に設定
P13/INT3					ポート方向レジスタ (DDR1: bit11 = 0) により " 入力ポート " に設定
P14/INT4					ポート方向レジスタ (DDR1: bit12 = 0) により " 入力ポート " に設定
P15/INT5					ポート方向レジスタ (DDR1: bit13 = 0) により " 入力ポート " に設定
P16/INT6					ポート方向レジスタ (DDR1: bit14 = 0) により " 入力ポート " に設定
P51/INT7	ポート 5 入出力 / 外部割込み入力		なし		ポート方向レジスタ (DDR5: bit9 = 0) により " 入力ポート " に設定

■ 端子のブロックダイアグラム

図 17.3-1 DTP/ 外部割込み回路の端子 (INT0 ~ INT6) のブロックダイアグラム

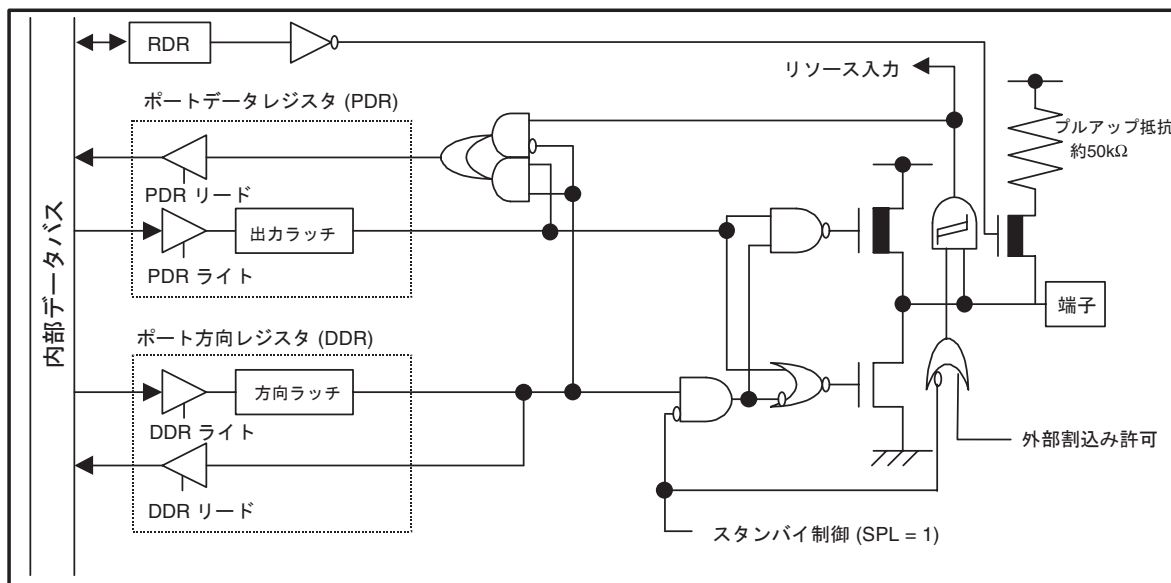
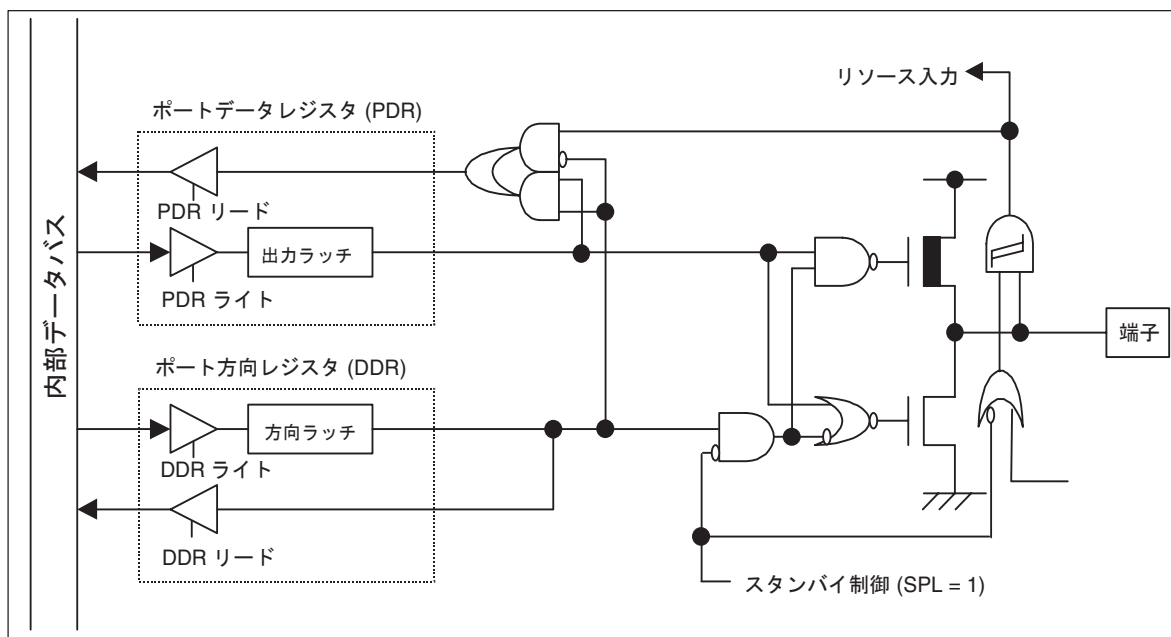


図 17.3-2 DTP/ 外部割込み回路の端子 (INT7) のブロックダイアグラム



17.4 DTP/ 外部割込み回路のレジスタ

DTP/ 外部割込み回路の端子のレジスタを記載します。

図 17.4-1 DTP/ 外部割込み回路の端子のレジスタ

DTP / 割込み要因レジスタ								
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	XXXXXXXX _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
DTP / 割込み許可レジスタ								
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	00000000 _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
要求レベル設定レジスタ (上位)								
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	00000000 _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
要求レベル設定レジスタ (下位)								
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	00000000 _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

17.4.1 DTP/ 外部割込み要因レジスタ (EIRR)

DTP/ 外部割込みの要因を保持とクリアするレジスタです。

■ DTP/ 外部割込み要因レジスタ (EIRR)

図 17.4-2 DTP/ 外部割込み要因レジスタ (EIRR)

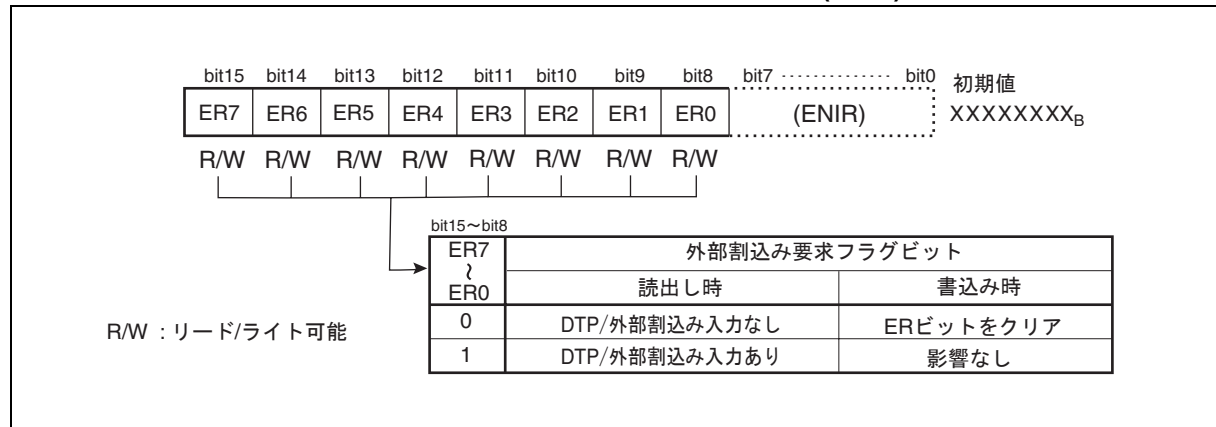


表 17.4-1 DTP/ 外部割込み要因レジスタ (EIRR) の機能

ビット名		機能
bit15 ~ bit8	ER7 ~ ER0: DTP/ 外部割込み要求フラ グビット	<p>DTP/ 外部割込み端子に、要求レベル設定レジスタ (ELVR) の LB7, LA7 ~ LB0, LA0 ビットで選択されたエッジまたはレベル信号が入力された場合は、"1" にセットされます (割込み要因の保持)。</p> <p>"1" がセットされた場合: このビットと対応する DTP/ 外部割込み要求許可ビット (ENIR) の EN7 ~ EN0 ビットを "1" に設定している場合は、CPU へ割込み要求を出力します。</p> <p>"0" に設定した場合: クリアされます。</p> <p>"1" に設定した場合: 影響しません。</p> <p>(注意事項)</p> <p>複数の外部割込み要求が許可されている場合は (ENIR : EN7 ~ EN0 = 1), 割込み (ER7 ~ ER0 ビットを "1" に設定) を受け付けたチャンネルのビットだけをクリアしてください。 (EIRR : ER = 0) それ以外のビットをクリアしないでください。</p> <p>(参考)</p> <p>拡張インテリジェント I/O サービス (EI²OS) が起動された場合は、1 データの転送が終了した時点で、対応する外部割込み要求フラグビットが自動的にクリアされます。</p>

17.4.2 DTP/ 外部割込み許可レジスタ (ENIR)

DTP/ 外部割込み許可レジスタ (ENIR) では、外部周辺装置の DTP/ 外部割込み要求を許可または禁止に設定します。

■ DTP/ 外部割込み許可レジスタ (ENIR)

図 17.4-3 DTP/ 外部割込み許可レジスタ (ENIR)

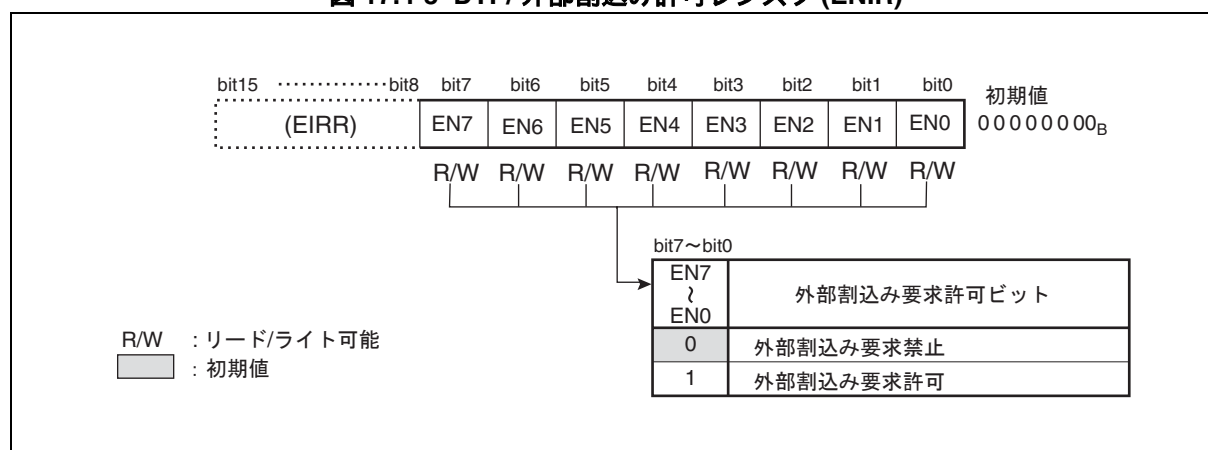


表 17.4-2 DTP/ 外部割込み許可レジスタ (ENIR) の機能

ビット名		機能
bit7 ~ bit0	EN7 ~ EN0: DTP/ 外部割込み要求許可 ビット	<p>CPU に対して、割込み要求出力の許可 / 禁止をするビットです。</p> <p>DTP/ 外部割込み要求許可ビット (ENIR: EN7 ~ EN0) に "1" を設定している場合に、DTP/ 外部割込み要求フラグビット (EIRR: ER7 ~ ER0) に "1" がセットされると、対応する DTP/ 外部割込み端子の割込み要求を発生します。</p> <p>(参考)</p> <p>DTP/ 外部割込み端子を使用する場合に、ポート方向レジスタの対応するビットに "0" を書き込み、端子を入力ポートに設定してください。</p> <p>外部割込み要求許可ビットの設定にかかわらず、DTP/ 外部割込み端子の状態は、ポートデータレジスタにより、読み出せます。</p> <p>DTP/ 割込み要因レジスタ (EIRR) の ER7 ~ ER0 ビットは外部割込み要求許可ビットの値に関係なく、割込み要因を検出すると "1" にセットされます。</p>

< 注意事項 >

- DTP/ 外部割込み要求フラグビット (EIRR:ER) の値は、対応する DTP/ 外部割込み要求許可ビット (ENIR:EN) が "1" に設定されている時のみ有効です。DTP/ 外部割込みが許可されていない状態 (ENIR:EN=0) では、DTP/ 外部割込み要因の有無にかかわらず DTP/ 外部割込要因ビットがセットされる可能性があります。
- DTP/ 外部割込みを許可 (ENIR:EN=1) する直前に、対応する DTP/ 外部割込み要求フラグビット (EIRR:ER) をクリアしてください。

表 17.4-3 DTP/ 外部割込み端子と割込み要求フラグビット，割込み許可ビットの対応

DTP/ 外部割込み端子	割込み番号	DTP/ 外部割込み要求 フラグビット	DTP/ 外部割込み要求 許可ビット
P51/INT7	#28 (1C _H)	ER7	EN7
P16/INT6	#27 (1B _H)	ER6	EN6
P15/INT5	#26 (1A _H)	ER5	EN5
P14/INT4	#25 (19 _H)	ER4	EN4
P13/INT3	#22 (16 _H)	ER3	EN3
P12/INT2		ER2	EN2
P11/INT1	#20 (14 _H)	ER1	EN1
P10/INT0/DTTI		ER0	EN0

17.4.3 要求レベル設定レジスタ (ELVR)

要求レベル設定レジスタ (ELVR) では、DTP/ 外部割込み端子のうち、割込み要因となる入力信号レベルまたはエッジを設定します。

■ 要求レベル設定レジスタ (ELVR)

図 17.4-4 要求レベル設定レジスタ (ELVR)

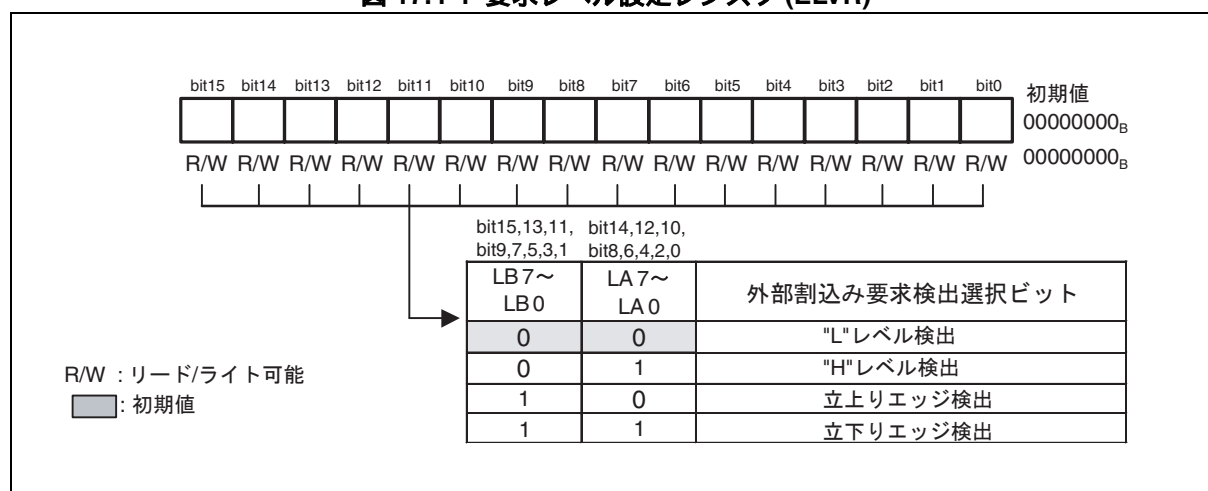


表 17.4-4 要求レベル設定レジスタ (ELVR) の機能

ビット名		機能
bit15 ~ bit0	LB7, LA0 ~ LB7, LA0: 要求条件選択ビット	<p>DTP/ 外部割込み端子に対して、割込み要因の発生の条件となる、外部周辺装置からの入力信号のレベルまたはエッジを設定します。</p> <ul style="list-style-type: none"> 外部割込みの場合は 2 種類のレベル 2 種類のエッジから、EI²OS の場合は、2 種類のレベルから選択します。 <p>(参考)</p> <p>設定した検出信号が、DTP/ 外部割込み端子に入力されると、DTP/ 外部割込み要求を禁止に設定していても (ENIR: EN=0), DTP/ 外部割込み要求フラグビットに "1" がセットされます。</p>

< 注意事項 >

DTP/ 外部割込みを許可 (ENIR:EN=1) する直前に、対応する DTP/ 外部割込み要求フラグビット (EIRR:ER) をクリアしてください。

表 17.4-5 要求レベル設定レジスタ上位と各チャネルの対応

DTP/ 外部割込み端子	割込み番号	ビット名
P51/INT7	#28 (1C _H)	LB7, LA7
P16/INT6	#27 (1B _H)	LB6, LA6
P15/INT5	#26 (1A _H)	LB5, LA5
P14/INT4	#25 (19 _H)	LB4, LA4
P13/INT3	#22 (16 _H)	LB3, LA3
P12/INT2		LB2, LA2
P11/INT1	#20 (14 _H)	LB1, LA1
P10/INT0/DTTI		LB0, LA0

17.5 DTP/ 外部割込みの動作説明

DTP/ 外部割込みには、外部割込み機能と DTP 機能があります。各機能の設定と動作について説明します。

■ DTP/ 外部割込みの設定

DTP/ 外部割込みを使用するには、図 17.5-1 の設定が必要です。

図 17.5-1 DTP/ 外部割込みの設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ICR08/ICR07 または ICR05/ICR04	ICS3	ICS2	ICS1	ICS0	ISE	IL2	IL1	IL0	ICS3	ICS2	ICS1	ICS0	ISE	IL2	IL1	IL0
外部割込み機能	0										0					
DTP 機能	1										1					
EIRR/ENIR	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
ELVR	LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0
DDR1		P16	P15	P14	P13	P12	P11	P10								
DDR6					P63											

: 使用ビット
 : 使用する端子に対応するビットに "1" を設定
 : 使用する端子に対応するビットに "0" を設定
 0 : "0" を設定
 1 : "1" を設定

● 設定の手順

DTP/ 外部割込みを使用する場合は、次の手順で各レジスタの設定を行ってください。

1. 外部割込み入力として使用する端子と兼用する汎用入出力ポートを入力ポートに設定します。
2. DTP/ 外部割込み許可レジスタ (ENIR) のビットの内、使用する DTP/ 外部割込みチャネルに対応する割込み要求許可ビットを禁止状態にします。
3. 要求レベル設定レジスタ (ELVR) のビットの内、使用する DTP/ 外部端子に対応する検出条件選択ビットを設定します。
4. DTP/ 外部割込み要因レジスタ (EIRR) のビットの内、使用する DTP/ 外部割込みチャネルに対応する割込み要求フラグビットをクリアします。
5. DTP/ 外部割込み許可レジスタ (ENIR) のビットの内、使用する DTP/ 外部割込みチャネルに対応する割込み要求許可ビットを許可にします。

- DTP/ 外部割込みのレジスタの設定は、外部割込み要求を禁止に (ENIR: EN7 ~ EN0 = 0) 設定してから行ってください。
- DTP/ 外部割込み要求を許可 (ENIR: EN7 ~ EN0 = 1) する場合は、先に対応する DTP/ 外部割込み要求フラグビットをクリア (ENIR: EN7 ~ EN0 = 0) してください。レジスタ設定時に誤って割込み要求が発生するのを防ぐためです。

● 外部割込み機能と DTP 機能の選択

外部割込み機能と DTP 機能のどちらが実行されるかは、対応する割込み制御レジスタの EI²OS 許可ビット (ICR: ISE) の設定によって決まります。

ISE ビットを "1" に設定した場合は、拡張インテリジェント I/O サービス (EI²OS) が許可され、DTP 機能が実行されます。

ISE ビットを "0" に設定した場合は、EI²OS は禁止され、外部割込み機能が実行されます。

< 注意事項 >

- 1 つの割込み制御レジスタに割り当てられている割込み要求は、すべて割込みレベル (IL2 ~ IL0) が同じになります。
- 1 つの割込み制御レジスタに複数の割込み要求が割り当てられている場合、そのうちの 1 つで EI²OS を使用した場合、ほかの割込み要求は使用できません。

■ DTP/ 外部割込み動作

DTP/ 外部割込みの制御ビットと割込み要因を表 17.5-1 に示します。

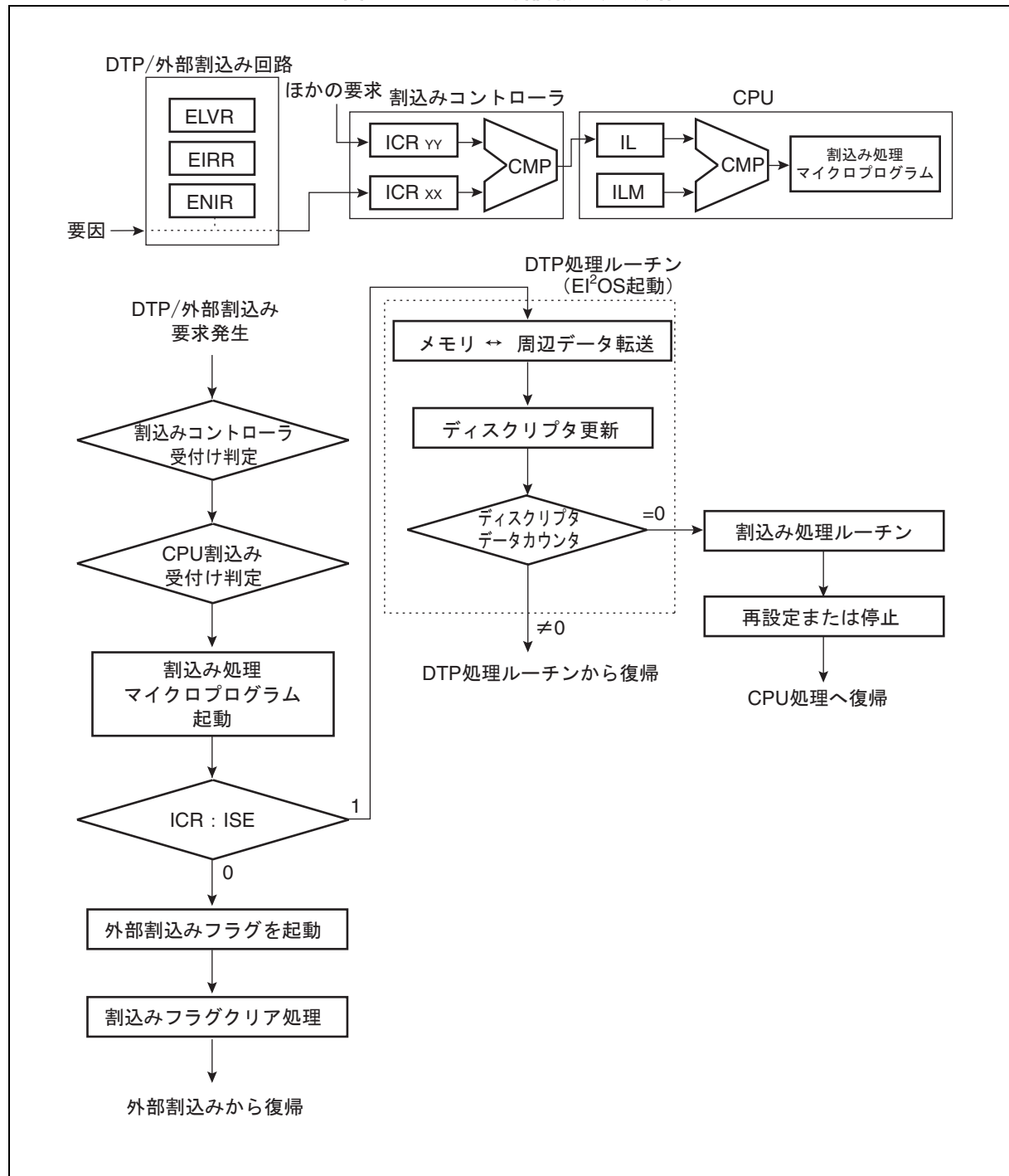
表 17.5-1 DTP/ 外部割込みの制御ビットと割込み要因

	DTP/ 外部割込み
割込み要求フラグビット	EIRR: ER7 ~ ER0
割込み要求許可ビット	ENIR: EN7 ~ EN0
割込み要因	INT7 ~ INT0 端子への、有効エッジ / レベルの入力

DTP/ 外部割込みの割込み要求が割込みコントローラに対して出力された場合、割込み制御レジスタ (ICR) の EI²OS 許可ビット (ISE: bit11) に "0" を設定している場合は、割込み処理が実行されます。"1" を設定している場合は、拡張インテリジェント I/O サービス (EI²OS) が実行されます。

図 17.5-2 に DTP/ 外部割込みの動作を示します。

図 17.5-2 DTP/ 外部割込みの動作



17.5.1 外部割込み機能

DTP/ 外部割込み機能には、DTP/ 外部割込み端子での信号 (エッジまたはレベル) の検出により、割込み要求を発生させる外部割込み機能があります。

■ 外部割込み機能

- DTP/ 外部割込み端子に、要求レベル設定レジスタ (ELVR) で設定した信号 (エッジまたはレベル) が検出されると、DTP/ 外部割込み要因レジスタの割込み要求フラグビット (EIRR: ER7 ~ ER0) に "1" がセットされます。
- 割込み要求フラグビットに "1" がセットされた場合に、DTP/ 外部割込み許可レジスタの割込み要求許可ビットを許可に設定していますと (ENIR: EN7 ~ EN0 = 1), 割込みコントローラに対して、割込み要求の発生を通知します。
- 割込みコントローラで、ほかの割込み要求よりも割込み優先順位が高いと判定された場合は、割込み要求が発生します。
- CPU では、プロセッサステータス (PS) の割込みレベルマスクレジスタ (PS: ILM) と割込み要求のレベル (ICR: IL) が比較され、割込み要求レベルが ILM より高く、割込み要求許可ビットが許可に設定されていた場合は (PS: CCR: I=1), 実行中の命令の終了後に、割込み処理が実行され、割込み処理へ分岐します。
- 割込み処理で対応する DTP/ 外部割込み要求フラグビットを "0" に設定し、DTP/ 外部割込み要求を "0" にクリアしてください。

< 注意事項 >

- DTP/ 外部割込み要求フラグビット (EIRR: ER) は、対応する DTP/ 外部割込み要求許可ビット (ENIR: EN) の設定にかかわらず、DTP/ 外部割込み起動要因が発生すると "1" がセットされます。
- 割込み処理が起動された場合は、起動要因となった DTP/ 外部割込み要求フラグビットを "0" にクリアしてください。DTP/ 外部割込み要求フラグビットが "1" のままでは割込みから復帰できません。"0" にクリアする場合は、受け付けられている DTP/ 外部割込み要因以外のフラグビットをクリアしないでください。

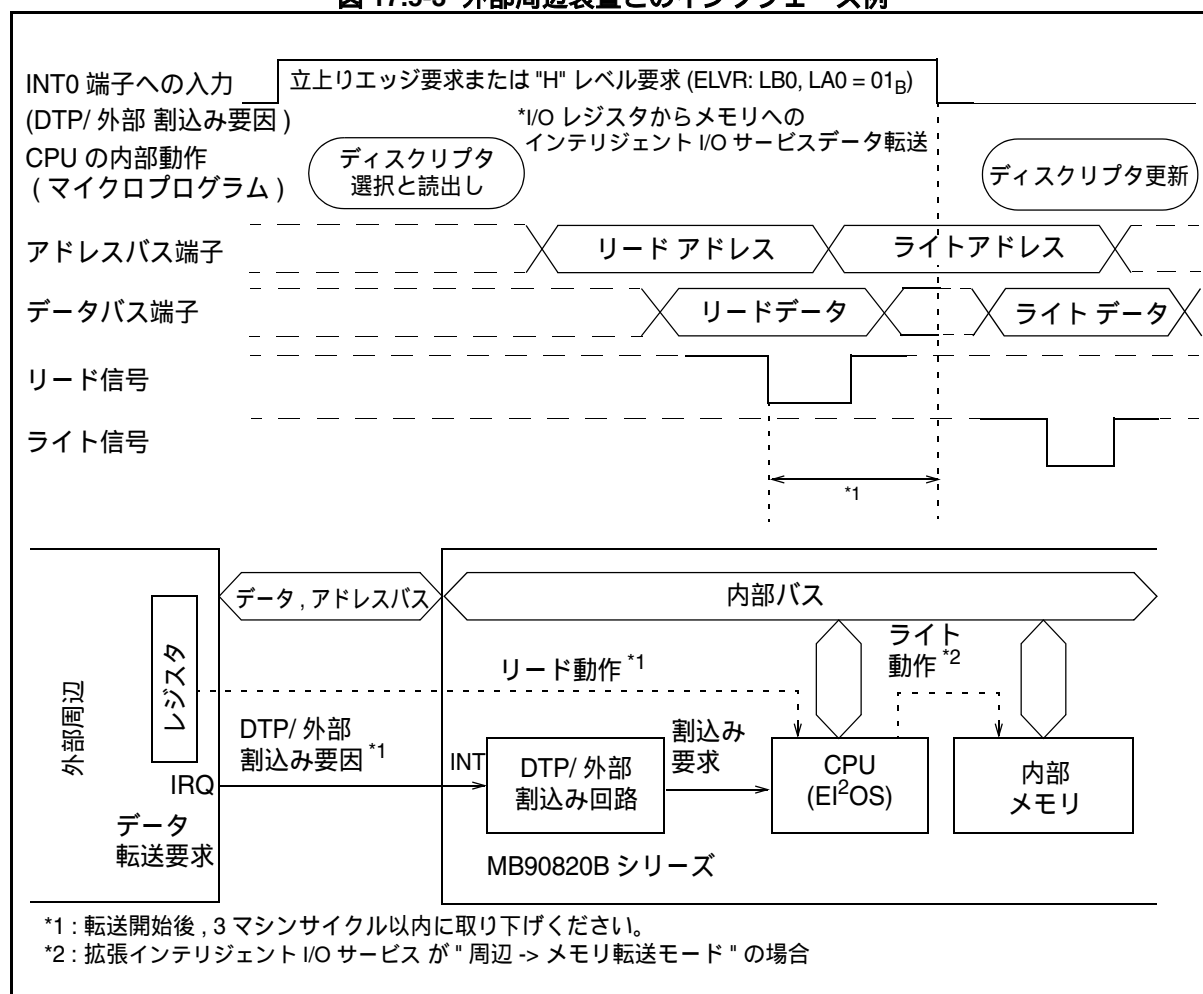
17.5.2 DTP 機能

DTP/ 外部割込みには、外部の周辺装置からの信号を DTP/ 外部割込み端子から検出し、拡張インテリジェント I/O サービスを起動する DTP 機能があります。

■ DTP 機能

- DTP 機能は、メモリと周辺の間で、外部の周辺装置からのデータ転送要求信号を検出し、データを自動転送する機能です。
- レベル検出の外部割込み機能で拡張インテリジェント I/O サービス (EI²OS) を起動します。CPU に割込み要求が受け付けられるまでは、外部割込み機能と同様の動作をしますが、EI²OS の動作が許可 (ICR : ISE = 1) されていれば、割込み要求が受け付けられた時点で、EI²OS を起動し、データ転送を開始します。1 データの転送が終了するとディスクリプタの更新が行われ、割込み要求フラグビットがクリアされて、DTP/ 外部割込み端子からの次の要求に備えます。EI²OS による転送がすべて終了すると、割込み処理ルーチンに分岐します。
- 外部周辺装置は最初の転送が開始されてから 3 マシンサイクル以内にデータ転送要求信号 (DTP 外部割込み要因) のレベルだけを取り下げてください。

図 17.5-3 外部周辺装置とのインタフェース例



17.6 DTP/ 外部割込みの使用上の注意

DTP/ 外部割込みを使用する場合，以下の点にご注意ください。

■ 使用上の注意

● DTP 機能を使用する場合の外部に接続する周辺装置の条件

- DTP 機能を使用する場合は，転送が行われたことにより自動的にデータ転送要求をクリアする周辺装置であることが必要です。
- 転送を開始してから 3 マシンクロック以内で転送要求信号をインアクティブにしてください。アクティブの場合，DTP/ 外部割込みでは，次の転送要求が発生したものとして扱ってしまいます。

● 外部割込みの入力極性について

- 要求レベル設定レジスタ (ELVR) でエッジ検出を設定している場合は，エッジを検出するためのパルス幅は最小 3 マシンクロックが必要です。
- 検出レベル設定レジスタでレベル検出を設定している場合は，割込み要因となるレベルが入力されると，DTP/ 外部割込み要因レジスタ (EIRR) 内部の要因 F/F に "1" がセットされ，図 17.6-1 に示すように，要因が保持されます。

要因 F/F に要因が保持された場合は，DTP/ 外部割込み要求が取り下げられても，割込み要求許可状態 (ENIR: EN=1) であれば，割込みコントローラへの要求はアクティブを保持します。割込みコントローラへの要求を取り下げるには割込み要求フラグビット (EIRR: ER) をクリアして，図 17.6-2 に示すように要因 F/F をクリアしてください。

図 17.6-1 レベル設定の場合の要因保持回路のクリア

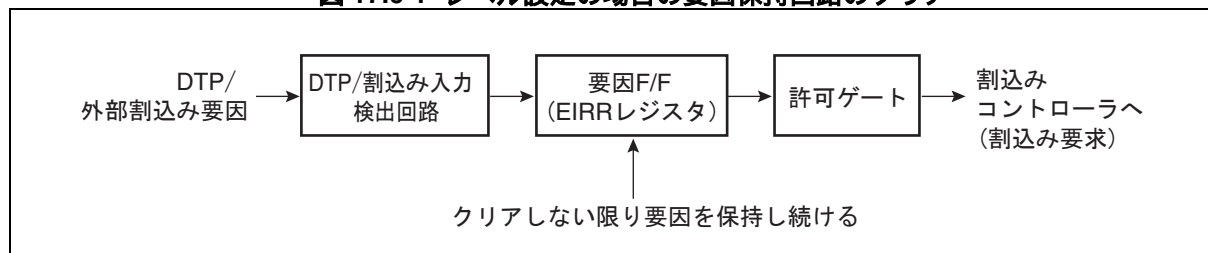
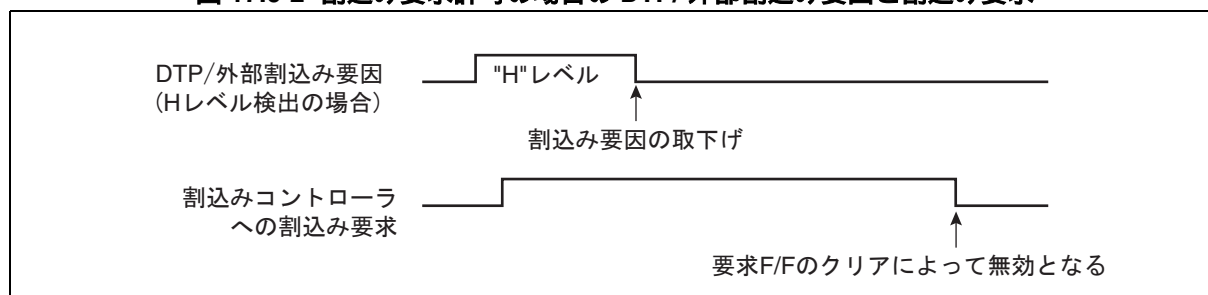


図 17.6-2 割込み要求許可の場合の DTP/ 外部割込み要因と割込み要求



● 割込みに関する注意

- 外部割込み機能として使用している場合は、DTP/外部割込み要求フラグビットに"1"がセットされ (EIRR: ER), DTP/ 外部割込み要求を許可に設定した場合 (ENIR: EN=1) では、割込み処理から復帰できません。割込み処理内で必ず DTP/ 外部割込み要求フラグビットを "0" に設定してください。 (EIRR: ER)
- 要求レベル設定レジスタでレベル検出を設定している場合は、割込み要因となるレベルが入力された状態を保持していれば、DTP/ 外部割込み要求フラグビットをクリアしても (EIRR: ER=0), すぐに再セットされます。必要に応じて DTP/ 外部割込み要求を禁止するか (ENIR: EN=0), または割込み要因そのものを解消してください。

第18章

8/10 ビット A/D コンバータ

この章では、8/10 ビット A/D コンバータの機能と動作について説明します。

- 18.1 8/10 ビット A/D コンバータの概要
- 18.2 8/10 ビット A/D コンバータのブロックダイアグラム
- 18.3 8/10 ビット A/D コンバータの構成
- 18.4 8/10 ビット A/D コンバータの割込み
- 18.5 8/10 ビット A/D コンバータの動作説明
- 18.6 8/10 ビット A/D コンバータ使用上の注意

18.1 8/10 ビット A/D コンバータの概要

8/10 ビット A/D コンバータは、RC 型逐次比較変換方式によって、アナログ入力電圧を 8 ビットまたは 10 ビットのデジタル値に変換します。

- 入力信号は、最大 16 チャンネルのアナログ入力端子から選択します。
 - 起動トリガは、ソフトウェアトリガ、内部タイマ出力、外部トリガから選択できます。
-

■ 8/10 ビット A/D コンバータの機能

アナログ入力端子に入力されたアナログ電圧（入力電圧）を 8 ビットまたは 10 ビットのデジタル値に変換します（A/D 変換）。

8/10 ビット A/D コンバータには以下の機能があります。

- A/D 変換時間は、サンプリング時間を含めて 1 チャンネルで最小 $1.9\ \mu\text{s}^*$ です。
- サンプリング時間は、1 チャンネルで最小 $0.5\ \mu\text{s}^*$ です。
- 変換方式は、サンプル & ホールド回路付き RC 型逐次変換比較方式です。
- 8 ビットまたは 10 ビットの分解能を設定できます。
- アナログ入力端子は、最大 16 チャンネルまで使用できます。
- A/D 変換結果を A/D データレジスタに格納することにより、割込み要求を発生できます。
- 割込み要求が発生した場合に EI²OS を起動することができます。
- 起動トリガは、ソフトウェア、内部タイマ出力（16 ビットリロードタイマ 1、16 ビットフリーランタイマ 0 検出または、コンベアクリア）から選択できます。

*: マシンクロック周波数 24MHz, AV_{CC} 4.5 V で動作している場合

■ 8/10 ビット A/D コンバータの変換モード

8/10 ビット A/D コンバータの変換モードには、次の種類があります。

表 18.1-1 8/10 ビット A/D コンバータの変換モード

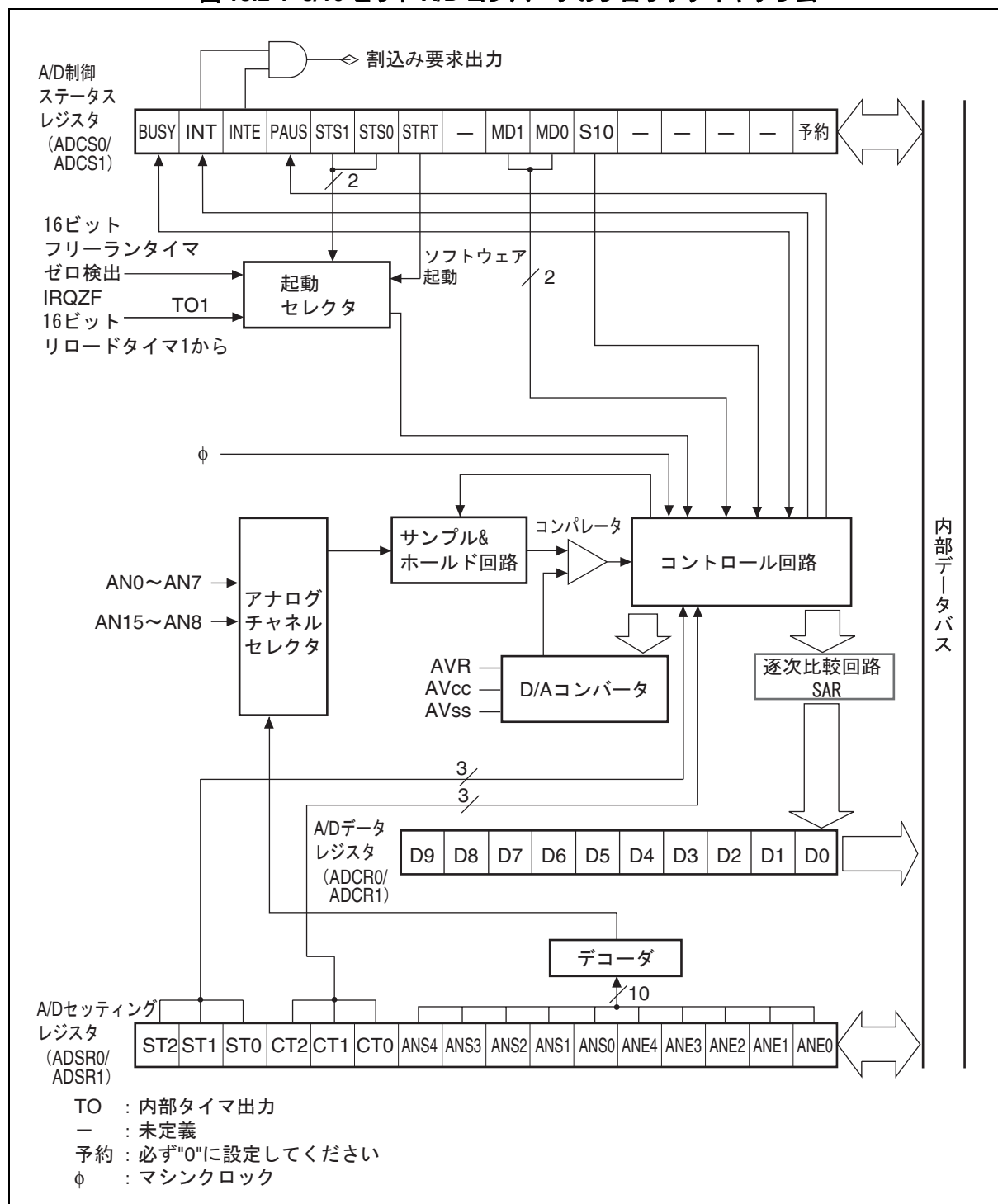
変換モード	内容
単発変換モード	開始チャネルから終了チャネルまで順次 A/D 変換を行います。終了チャネルの A/D 変換が終了すると、A/D 変換機能を停止します。
連続変換モード	開始チャネルから終了チャネルまで順次 A/D 変換を行います。終了チャネルの A/D 変換が終了すると、開始チャネルに戻って A/D 変換動作を継続します。
停止変換モード	1 チャネルごとに停止しながら A/D 変換を行います。終了チャネルの A/D 変換が終了すると、開始チャネルに戻って A/D 変換と停止を繰り返します。

18.2 8/10 ビット A/D コンバータのブロックダイアグラム

8/10 ビット A/D コンバータは、以下のブロックで構成されています。

■ 8/10 ビット A/D コンバータのブロックダイアグラム

図 18.2-1 8/10 ビット A/D コンバータのブロックダイアグラム



● ブロックダイヤグラム中の端子などの詳細

8/10 ビット A/D コンバータの実際の端子名および割込み要求番号を表 18.2-1 に示します。

表 18.2-1 ブロックダイヤグラムの端子および割込み要求番号

ブロックダイヤグラム中の端子名 / 割込み要求番号		実際の端子名 / 割込み要求番号
TO1	内部タイマ出力	16 ビットリロードタイマ 1 の出力
IRQZF	内部タイマ出力	16 ビットフリーランタイマ 0 検出またはコンペアクリア
AN0 ~ AN7	アナログ入力端子 ch.0 ~ ch.7	P60/AN0 ~ P67/AN7
AN8 ~ AN15	アナログ入力端子 ch.8 ~ ch.15	P70/AN8 ~ P77/AN15
AVR	V _{ref} 入力端子	AVR
AV _{CC}	V _{CC} 入力端子	AV _{CC}
AV _{SS}	V _{SS} 入力端子	AV _{SS}
割込み要求出力	割込み要求出力	#29 (1D _H)

● A/D 制御ステータスレジスタ (ADCS)

ソフトウェアによる A/D 変換機能の起動, A/D 変換機能の起動トリガの選択, 変換モードの選択, 割込み要求の許可または禁止, 割込み要求フラグの確認とクリア, A/D 変換動作の一時停止および変換中の状態確認, 分解能の選択を行います。

● 逐次比較回路 (SAR)

1 ビットずつ逐次比較を実行し, 変換結果を格納します。次の A/D 変換が開始されると本回路内の A/D 変換結果は破壊されます。

● A/D データレジスタ (ADCR)

A/D 変換結果は A/D 変換実行時に逐次比較回路に 1bit ずつ格納され, A/D 変換が終了して変換結果が確定した時点で本レジスタに格納されます。本レジスタから A/D 変換結果をリードする事ができます。

● A/D セットアップレジスタ (ADSR)

A/D 変換の開始チャネルおよび終了チャネルの設定, A/D 変換のコンペア時間, サンプリング時間の設定を行います。

● 起動セレクト

A/D 変換を起動するトリガを選択します。起動トリガには, 内部タイマ出力または外部端子入力を設定できます。

● デコーダ

A/D セットアップレジスタの A/D 変換開始チャネル選択ビット (ADSR: ANS3 ~ ANS0) および A/D 変換終了チャネル選択ビット (ADSR: ANE3 ~ ANE0) の設定から, A/D 変換に使用するアナログ入力端子を選択します。

● アナログチャネルセクタ

デコーダからの信号を受けて、16 チャネルのアナログ入力端子から A/D 変換に使用する端子を選択します。

● サンプル & ホールド回路

アナログチャネルセクタで選択された入力電圧を保持します。A/D 変換を開始した直後の入力電圧を保持することによって、A/D 変換中の入力電圧の変動の影響を受けずに変換できます。

● D/A コンバータ

サンプル & ホールド回路で保持された入力電圧と比較するための、基準電圧を発生します。

● コンパレータ

サンプル & ホールド回路で保持された入力電圧と、D/A コンバータの出力電圧を比較して電圧の大小を判定します。

● コントロール回路

コンパレータからの大小信号を受けて、A/D 変換値を決定します。変換結果が確定すると、変換結果のデータを A/D データレジスタに格納します。割込み要求が許可されている場合は、割込みが発生します。

18.3 8/10 ビット A/D コンバータの構成

A/D コンバータの端子，レジスタ，割込み要因を示します。

■ 8/10 ビット A/D コンバータの端子

8/10 ビット A/D コンバータの端子は，汎用入出力ポートと兼用です。端子の機能，8/10 ビット A/D コンバータを使用する場合の設定を表 18.3-1 に示します。

表 18.3-1 8/10 ビット A/D コンバータの端子

機能名	端子名	端子機能	8/10 ビット A/D コンバータを 使用した場合の設定
ch.0	P60/AN0	汎用入出力ポート / アナログ入力	アナログ信号の入力許可 (ADER0: ADE7 ~ ADE0 の対応するビットを "1" に設定)
ch.1	P61/AN1		
ch.2	P62/AN2		
ch.3	P63/AN3		
ch.4	P64/AN4		
ch.5	P65/AN5		
ch.6	P66/AN6		
ch.7	P67/AN7		
ch.8	P70/DA0/AN8	汎用入出力ポート / アナログ入力 / D/A コンバータ出力	アナログ信号の入力許可 (ADER1: ADE15 ~ ADE8 の対応するビットを "1" に設定)
ch.9	P71/DA1/AN9		
ch.10	P72/SIN1/AN10	汎用入出力ポート / アナログ入力 / UART1 入出力	
ch.11	P73/SOT1/AN11		
ch.12	P74/SCK1/AN12		
ch.13	P75/FRCK/AN13	汎用入出力ポート / アナログ入力 / フリーランタイム クロック入力	
ch.14	P76/IN0/AN14	汎用入出力ポート / アナログ入力 / 外部割込み入力	
ch.15	P77/IN1/AN15		

■ 8/10 ビット A/D コンバータのレジスタと初期値の一覧

図 18.3-1 8/10 ビット A/D コンバータのレジスタと初期値の一覧

A/D制御ステータスレジスタ上位 ADCS1									
bit	15	14	13	12	11	10	9	8	初期値
	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	—	0000000XB
	R/W	R/W	R/W	R/W	R/W	R/W	W	—	
A/D制御ステータスレジスタ下位 ADCS0									
bit	7	6	5	4	3	2	1	0	初期値
	MD1	MD0	S10	—	—	—	—	予約	000XXXX0B
	R/W	R/W	R/W	—	—	—	—	R/W	
データレジスタ上位 ADCR1									
bit	15	14	13	12	11	10	9	8	初期値
	—	—	—	—	—	—	D9	D8	XXXXXXXXXB
	—	—	—	—	—	—	R	R	
データレジスタ下位 ADCR0									
bit	7	6	5	4	3	2	1	0	初期値
	D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXXXB
	R	R	R	R	R	R	R	R	
A/Dセッティングレジスタ上位 ADSR1									
bit	15	14	13	12	11	10	9	8	初期値
	ST2	ST1	ST0	CT2	CT1	CT0	予約	ANS3	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
A/Dセッティングレジスタ下位 ADSR0									
bit	7	6	5	4	3	2	1	0	初期値
	ANS2	ANS1	ANS0	予約	ANE3	ANE2	ANE1	ANE0	00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
R/W : リード/ライト可能 R : リードオンリ W : ライトオンリ — : 未定義ビット X : 不定									

18.3.1 A/D 制御ステータスレジスタ上位 (ADCS1)

A/D 制御ステータスレジスタ上位 (ADCS1) では、次の設定ができます。

- ソフトウェアによる A/D 変換機能の起動
- A/D 変換機能の起動トリガの選択
- A/D データレジスタへの A/D 変換結果格納による割込み要求の許可または禁止
- A/D データレジスタへの A/D 変換結果格納による割込み要求フラグの確認とクリア
- A/D 変換動作の一時停止および変換中の状態確認

■ A/D 制御ステータスレジスタ上位 (ADCS1)

図 18.3-2 A/D 制御ステータスレジスタ上位 (ADCS1)

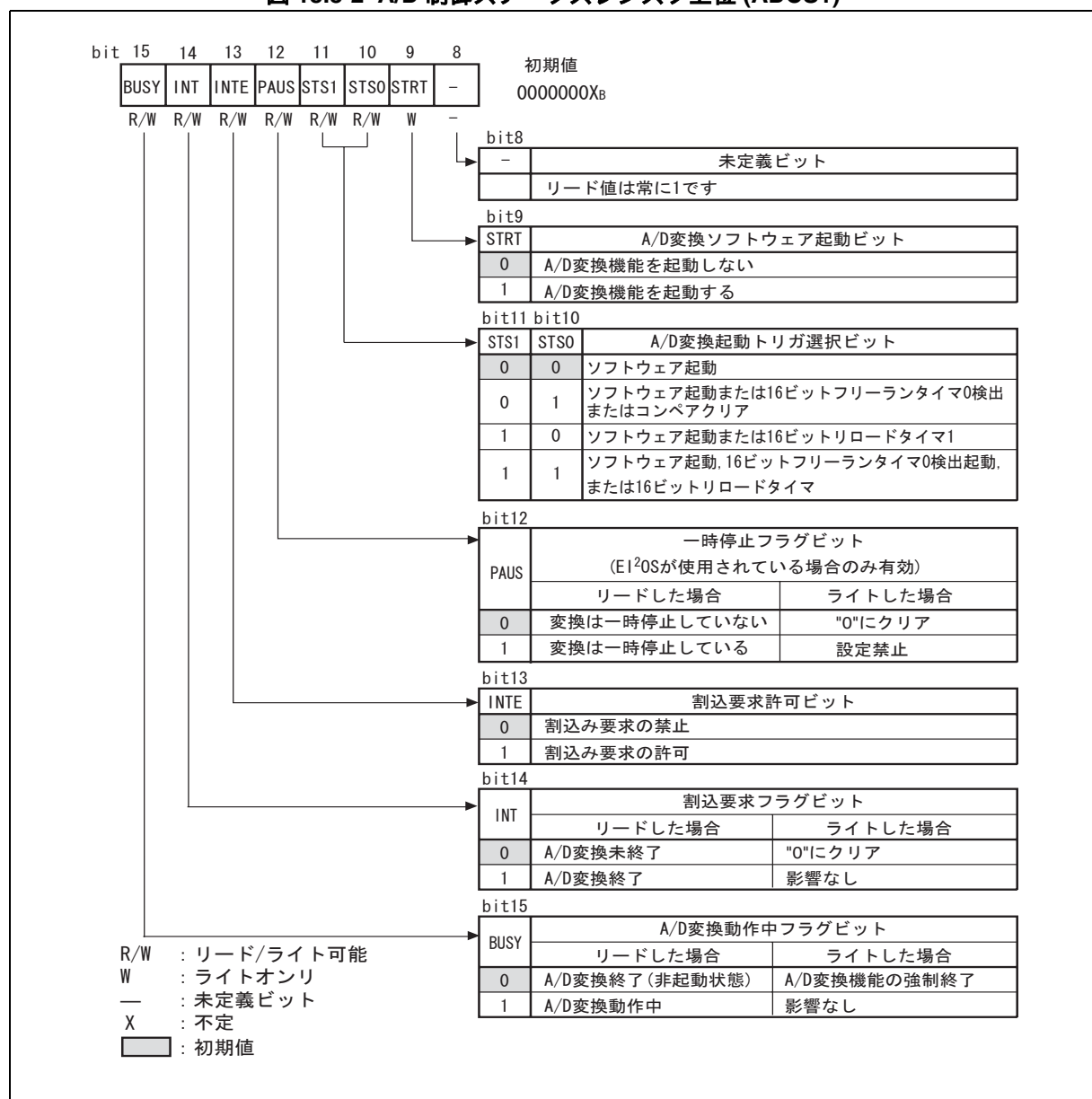


表 18.3-2 A/D 制御ステータスレジスタ上位 (ADCS1) の機能 (1 / 3)

ビット名		機能
bit15	BUSY: A/D 変換動作中 フラグビット	<p>8/10 ビット A/D コンバータを強制終了します。リードした場合は、8/10 ビット A/D コンバータが動作中か停止中かを示します。</p> <p>"0" に設定した場合：8/10 ビット A/D コンバータを強制終了します。</p> <p>"1" に設定した場合：影響しません。</p> <p>リードした場合：8/10 ビット A/D コンバータが動作中であれば "1" が、停止中であれば "0" が読み出されます。停止変換モードの " 停止状態 " では "1" が読み出されます。</p> <p>< 注意事項 ></p> <ul style="list-style-type: none"> • RMW 系命令では "1" が読み出されます。 • 単発変換モードでは A/D 変換終了でクリアされます。 • 連続変換モード、停止変換モードでは "0" 書込みで停止するまでクリアされません。 • A/D コンバータの強制終了 (BUSY=0) と起動 (ソフト (STRT=1) / タイマのいずれも) は同時に行わないでください。
bit14	INT: 割込み要求フラグ ビット	<p>割込み要求が発生したことを示します。</p> <ul style="list-style-type: none"> • A/D 変換が終了して、A/D 変換結果が A/D データレジスタ (ADCR) に格納されると INT ビットに "1" がセットされます。 • 割込み要求を許可した場合に (INTE=1), 割込み要求フラグビットがセットされると (INT=1), 割込み要求が発生します。 • 本ビットは "0" を書き込んだ場合にクリアされます。また、EI²OS による A/D 変換結果データの転送が完了すると自動的にクリアされます。 <p>"0" に設定した場合：クリアされます。</p> <p>"1" に設定した場合：影響しません。</p> <p>< 注意事項 ></p> <ul style="list-style-type: none"> • RMW 系命令では "1" が読み出されます。
bit13	INTE: 割込み要求許可 ビット	<p>割込み要求の出力を許可または禁止します。</p> <ul style="list-style-type: none"> • 割込み要求を許可した場合に (INTE=1), 割込み要求フラグビットがセットされると (INT=1) 割込み要求が発生します。 <p>< 注意事項 ></p> <p>EI²OS で A/D 変換結果を転送する場合は、必ず "1" に設定してください。</p>

表 18.3-2 A/D 制御ステータスレジスタ上位 (ADCS1) の機能 (2 / 3)

	ビット名	機能
bit12	PAUS: 一時停止フラグ ビット	<p>PAUS ビットは A/D 変換データ保護機能が働いたことを示します。PAUS ビットは割り込み要求の出力を許可 (ADCS:INTE=1) に設定している場合のみ有効です。</p> <p>A/D 変換データ保護機能が働いた場合：</p> <p style="padding-left: 40px;">"1" にセットされます。</p> <p>"0" に設定した場合： "0" にクリアされます。</p> <p>"1" に設定した場合： "1" にセットされます。</p> <ul style="list-style-type: none"> • 割り込み要求の出力を許可 (ADCS:INTE=1) に設定して A/D 変換を実行した場合、A/D 変換が 1 回終了すると割り込み要求フラグビット (ADCS:INT) がセットされると同時に割り込み要求が発生します。割り込み要求フラグビット (ADCS:INT) をクリアしないまま次の A/D 変換が終了した場合、前データが上書き破壊されることを防ぐために A/D 変換動作が一時停止します (A/D 変換データ保護機能)。 • A/D 変換動作が一時停止すると PAUS ビットが "1" にセットされます。 • 割り込み要求フラグビット (ADCS:INT) がクリアされると、8/10 ビット A/D コンバータは一時停止状態を解除し、A/D 変換動作を再開します。 • 割り込み要求フラグビット (ADCS:INT) は "0" を書き込むことでクリアされます。また、EI²OS で A/D 変換結果を A/D データレジスタから転送する設定の場合は、A/D 変換結果の転送が完了した時点で割り込み要求フラグビット (ADCS:INT) は EI²OS によってクリアされます。 <p>< 注意事項 ></p> <ul style="list-style-type: none"> • A/D 変換データ保護機能に関しては「18.5.5 A/D 変換データ保護機能」を参照してください。 • 一時停止状態が解除されても PAUS ビットは自動的にクリアされません。PAUS ビットをクリアするためには "0" をライトしてください。

表 18.3-2 A/D 制御ステータスレジスタ上位 (ADCS1) の機能 (3 / 3)

ビット名		機能
bit11, bit10	STS1, STS0: A/D 変換起動トリ ガ選択ビット	<p>8/10 ビット A/D コンバータを起動するトリガ (起動トリガ) を選択します。</p> <ul style="list-style-type: none"> • 00_B: ソフト起動 • 01_B: 16 ビットフリーランタイム 0 検出またはコンペアクリア / ソフト起動 • 10_B: 16 ビットリロードタイム / ソフト起動 • 11_B: 16 ビットフリーランタイム 0 検出またはコンペアクリア / 16 ビットリロードタイム / ソフト起動 <p>< 注意事項 ></p> <ul style="list-style-type: none"> • 16 ビットフリーランタイム 0 検出またはコンペアクリアが選択された場合 (01_B, 11_B), 16 ビットフリーランタイム 0 検出されると A/D 変換を開始します。 • 16 ビットリロードタイムを選択した場合 (10_B, 11_B), 16 ビットリロードタイム 1 の出力が "1" になると A/D 変換を開始します。 <p>< 注意事項 ></p> <ul style="list-style-type: none"> • 起動トリガを複数に設定した場合 (STS1, STS0=00_B 以外), 8/10 ビット A/D コンバータは最初に発生した起動トリガで起動します。 • 起動トリガの設定を変更する場合は, 起動トリガを発生させる周辺機能の動作が停止している状態 (トリガがインアクティブの状態) で設定してください。
bit9	STRT: A/D 変換 ソフトウェア起動 ビット	<p>8/10 ビット A/D コンバータをソフトウェアで起動します。</p> <p>"1" に設定した場合: 8/10 ビット A/D コンバータが起動します。</p> <ul style="list-style-type: none"> • 停止変換モードで A/D 変換動作が一時停止している場合は, STRT ビットに "1" を書き込むことによって A/D 変換動作を再開します。 <p>"0" に設定した場合: 無効です。変化しません。</p> <p>< 注意事項 ></p> <ul style="list-style-type: none"> • RMW 系命令では "0" が読み出されます。 • RMW 系命令以外で読んだ場合は, 書き込んだ値ではなく "1" が読み出されます。 • 8/10 ビット A/D コンバータの強制終了 (BUSY=0) とソフトウェア起動 (STRT=1) は同時に行わないでください。
bit8	未定義ビット	<ul style="list-style-type: none"> • 読み出した場合: 常に "1" が読み出されます。 • 書き込んだ場合: 影響しません。

18.3.2 A/D 制御ステータスレジスタ下位 (ADCS0)

A/D 制御ステータスレジスタ下位 (ADCS0) では、次の設定ができます。

- A/D 変換モードの選択
- A/D 変換の開始チャンネルおよび終了チャンネルの選択

■ A/D 制御ステータスレジスタ下位 (ADCS0)

図 18.3-3 A/D 制御ステータスレジスタ下位 (ADCS0)

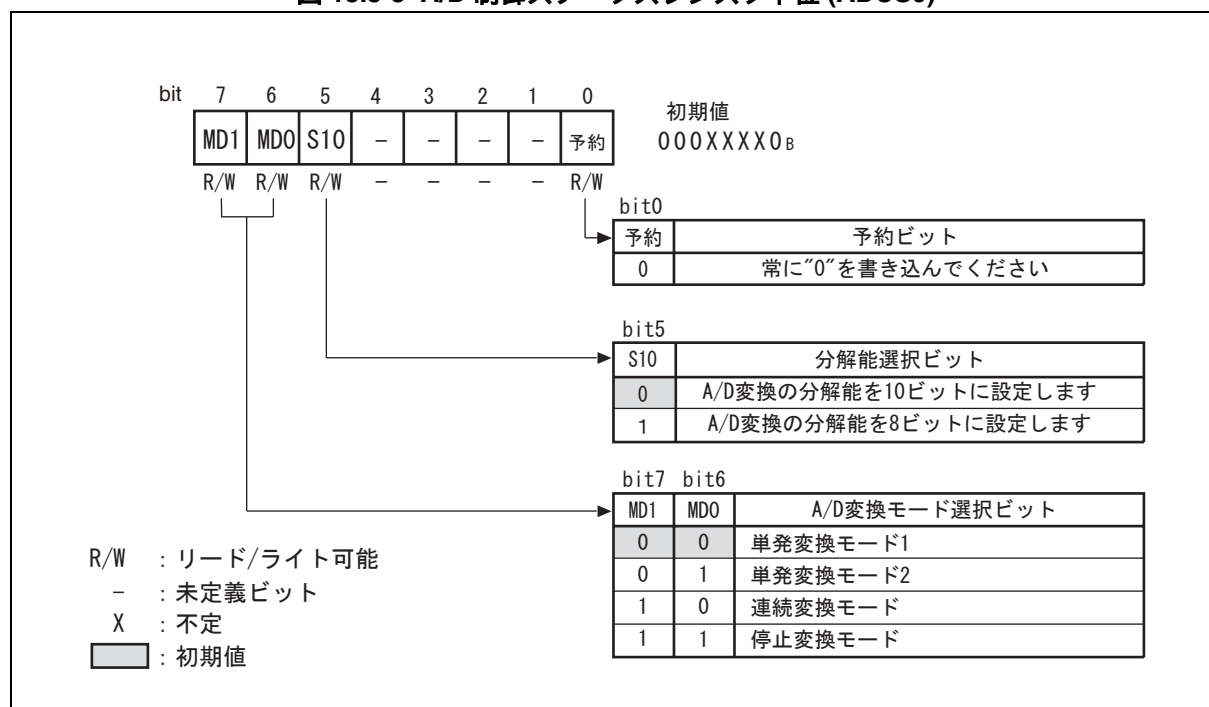


表 18.3-3 A/D 制御ステータスレジスタ下位 (ADCS0) の機能

ビット名	機能
bit7, bit6	<p>A/D 変換の変換モードを設定します。 各モードの使用方法的詳細につきましては「18.5 8/10 ビット A/D コンバータの動作説明」を参照してください。</p> <p>単発変換モード 1 および単発変換モード 2 の場合：</p> <ul style="list-style-type: none"> 開始チャネル (ADSR0/1: ANS3 ~ ANS0) から終了チャネル (ADSR0/1: ANE3 ~ ANE0) までのアナログ入力を連続して A/D 変換していきます。 終了チャネルの A/D 変換が終了すると、A/D 変換動作を停止します。 単発変換モード 1, 単発変換モード 2 の違いについては「18.5 8/10 ビット A/D コンバータの動作説明」を参照してください。 <p>連続変換モードの場合：</p> <ul style="list-style-type: none"> 開始チャネル (ADSR0/1: ANS3 ~ ANS0) から終了チャネル (ADSR0/1: ANE3 ~ ANE0) までのアナログ入力を連続して A/D 変換していきます。 終了チャネルの A/D 変換が終了すると、開始チャネルのアナログ入力に戻って A/D 変換を継続して行います。 <p>停止変換モードの場合：</p> <ul style="list-style-type: none"> 開始チャネル (ADSR0/1: ANS3 ~ ANS0) から A/D 変換が開始されます。1 チャネルの A/D 変換が終了すると、A/D 変換動作は停止します。A/D 変換動作が停止している間に起動トリガを入力すると、次のチャネルの A/D 変換が行われます。 終了チャネルの A/D 変換が終了すると、A/D 変換動作は停止します。A/D 変換動作が停止している間に起動トリガを入力すると、開始チャネルのアナログ入力に戻って A/D 変換を継続します。 <p>< 注意事項 ></p> <ul style="list-style-type: none"> 変換モードを変更する場合は、A/D 変換を開始する前の停止状態で行ってください。
bit5	<p>A/D 変換の分解能を設定します。</p> <p>"0" に設定した場合： A/D 変換の分解能を A/D 変換データビット D9 ~ D0 までの 10 ビットに設定します。</p> <p>"1" に設定した場合： A/D 変換の分解能を A/D 変換データビット D7 ~ D0 までの 8 ビットに設定します。</p> <p>< 注意事項 ></p> <p>S10 ビットを変更する場合は、A/D 変換を開始する前の停止状態で行ってください。A/D 変換開始後に S10 ビットを変更すると、A/D 変換データビット (D9 ~ D0) に格納された変換結果は無効になります。</p>

18.3.3 A/D データレジスタ (ADCR0/ADCR1)

データレジスタ (ADCR0/ADCR1) は、変換の結果として生成されたデジタル値を記憶するために使用されます。ADCR0 は下位 8 ビットを記憶し、ADCR1 は変換結果の最上位の 2 ビットを記憶します。本レジスタは変換が完了するたびに書き換えられ、通常は最後の変換値が記憶されています。

■ A/D データレジスタ (ADCR0/ADCR1)

図 18.3-4 A/D データレジスタ (ADCR0/ADCR1)

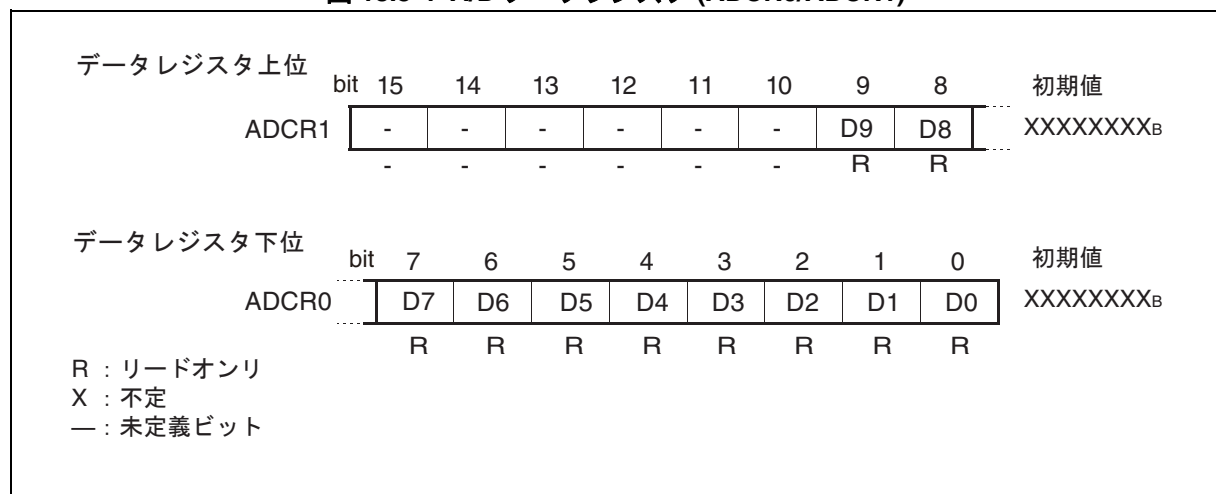


表 18.3-4 A/D データレジスタ (ADCR0/ADCR1) の機能

ビット名		機能
bit15 ~ bit10	未定義ビット	リード時は常に "1" が読み出されます。
bit9 ~ bit0	D9 ~ D0: A/D 変換データ ビット	A/D 変換の結果を格納します。 分解能を 10 ビットに設定した場合 (S10=0) : 変換データは D9 ~ D0 までの 10 ビットに格納されます。 分解能を 8 ビットに設定した場合 (S10=1) : 変換データは D7 ~ D0 までの 8 ビットに格納されます。このとき、D9 ~ D8 のリード値は "1" となります。 < 注意事項 > ・ 本レジスタへの書込みは禁止です ・ A/D 変換データビット (D9 ~ D0) に格納された変換結果を読み出す場合は、ワード命令 (MOVW) を使用してください。

18.3.4 A/D セットアップレジスタ (ADSR0/ADSR1)

A/D セットアップレジスタ (ADSR0/ADSR1) では、次の設定ができます。

- A/D 変換時間 (サンプリング時間, コンペア時間) の設定
- サンプリングチャネル (開始チャネル, 終了チャネル) の設定
- 現在のサンプリングチャネルを表示

■ A/D セットアップレジスタ (ADSR0/ADSR1)

図 18.3-5 A/D セットアップレジスタ (ADSR0/ADSR1)

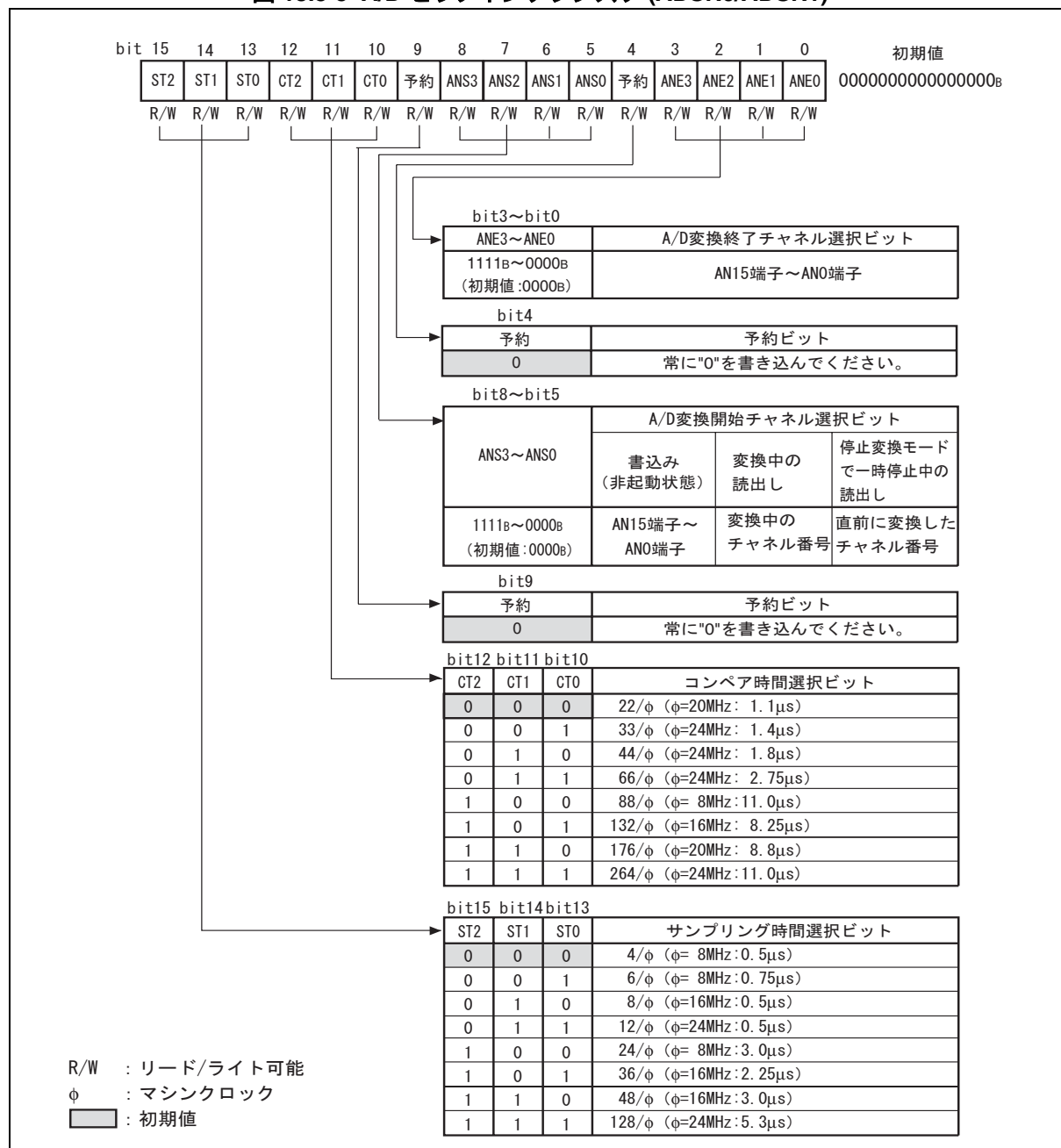


表 18.3-5 A/D セットアップレジスタ (ADSR0/ADSR1) の機能 (1 / 2)

ビット名		機能
bit15 ~ bit13	ST2, ST1, ST0: サンプリング時間 選択ビット	<p>A/D 変換のサンプリング時間を設定します。</p> <ul style="list-style-type: none"> A/D 変換を開始してから、入力されたアナログ電圧がサンプル & ホールド回路でサンプリングされ、保持されるまでの時間を設定します。 本ビットの設定については表 18.3-6 を参照してください。
bit12 ~ bit10	CT2, CT1, CT0: コンペア時間選択 ビット	<p>A/D 変換のコンペア時間 (比較時間) を設定します。</p> <ul style="list-style-type: none"> アナログ入力を A/D 変換してデータビット (D9 ~ D0) に格納するまでの時間を設定します。 本ビットの設定については表 18.3-7 を参照してください。
bit8 ~ bit5	ANS3 ~ ANS0: A/D 変換開始 チャンネル選択ビット	<p>A/D 変換を開始するチャンネルを設定します。リードした場合は、A/D 変換中であれば現在変換中のチャンネル番号、A/D 変換終了後または停止中であれば最後に A/D 変換したチャンネル番号が確認できます。</p> <p>また、本ビットに値を設定した場合でも、A/D 変換が開始されるまでは、設定した値ではなく、前回に A/D 変換したチャンネル番号が読み出されます。リセット時は、"0000_B" に初期化されます。</p> <p>開始チャンネル < 終了チャンネルの場合：</p> <p>A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルから A/D 変換を開始し、A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) で設定されたチャンネルで A/D 変換を終了します。</p> <p>開始チャンネル = 終了チャンネルの場合：</p> <p>A/D 変換開始 (= 終了) チャンネル選択ビット (ANS3 ~ ANS0=ANE3 ~ ANE0) で設定された 1 チャンネルのみ A/D 変換を行います。</p> <p>連続変換モード、停止変換モードの場合：</p> <p>A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) に設定されたチャンネルで A/D 変換が終了すると、A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルに戻ります。</p> <p>リードした場合 (停止変換モード以外)：</p> <p>A/D 変換中のチャンネル番号 (15 ~ 0) が読み出されます。</p> <p>リードした場合 (停止変換モード)：</p> <p>停止中にリードすると、停止直前に A/D 変換したチャンネル番号が読み出されます。</p> <p>< 注意事項 ></p> <ul style="list-style-type: none"> 開始チャンネル > 終了チャンネルとなるような設定は行わないでください。 A/D 変換中は、A/D 変換開始チャンネルビット (ANS3 ~ ANS0) の設定は行わないでください。 本ビットへの書込みは Word アクセスで行ってください。Byte ライトやビット操作を行った場合は意図しないチャンネルから A/D 変換が開始される可能性があります。

表 18.3-5 A/D セットアップレジスタ (ADSR0/ADSR1) の機能 (2 / 2)

	ビット名	機能
bit3 ~ bit0	ANE3 ~ ANE0: A/D 変換終了 チャンネル選択ビット	<p>A/D 変換を終了するチャンネルを設定します。</p> <p>開始チャンネル < 終了チャンネルの場合 :</p> <p>A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルから A/D 変換を開始し, A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) で設定されたチャンネルで A/D 変換を終了します。</p> <p>開始チャンネル = 終了チャンネルの場合 :</p> <p>A/D 変換開始 (= 終了) チャンネル選択ビット (ANS3 ~ ANS0=ANE3 ~ ANE0) で設定された 1 チャンネルのみ A/D 変換を行います。</p> <p>連続変換モード, 停止変換モードの場合 :</p> <p>A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) に設定されたチャンネルで A/D 変換が終了すると, A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルに戻ります。</p> <p>< 注意事項 ></p> <ul style="list-style-type: none"> 開始チャンネル > 終了チャンネルとなるような設定は行わないでください。 A/D 変換中は, A/D 変換終了チャンネルビット (ANE3 ~ ANE0) の設定は行わないでください。 A/D 変換開始チャンネル選択ビット (ANS3, ANS2, ANS1, ANS0) を設定した後に, サンプリング時間選択ビット (ST2, ST1, ST0), コンペア時間選択ビット (CT2, CT1, CT0) および A/D 変換終了チャンネル選択ビット (ANE3, ANE2, ANE1, ANE0) をリードモディファイライト系命令で設定しないでください。ANS3, ANS2, ANS1, ANS0 ビットは A/D 変換動作が開始するまでは前回の変換チャンネルが読み出されるため, ANS3, ANS2, ANS1, ANS0 ビット設定後に ST2, ST1, ST0 ビット, CT2, CT1, CT0 ビットおよび ANE3, ANE2, ANE1, ANE0 ビットをリードモディファイ系命令で設定した場合, ANS3, ANS2, ANS1, ANS0 ビットの値が書き換わる可能性があります。

■ サンプルング時間の設定 (ST2 ~ ST0 ビット)

表 18.3-6 ST2 ~ ST0 ビットとサンプルング時間の関係

ST2	ST1	ST0	サンプルング時間設定	設定例 (ϕ : 内部動作周波数)
0	0	0	4 マシンサイクル	$\phi = 8\text{MHz} : 0.5\mu\text{s}$
0	0	1	6 マシンサイクル	$\phi = 8\text{MHz} : 0.75\mu\text{s}$
0	1	0	8 マシンサイクル	$\phi = 16\text{MHz} : 0.5\mu\text{s}$
0	1	1	12 マシンサイクル	$\phi = 24\text{MHz} : 0.5\mu\text{s}$
1	0	0	24 マシンサイクル	$\phi = 8\text{MHz} : 3\mu\text{s}$
1	0	1	36 マシンサイクル	$\phi = 16\text{MHz} : 2.25\mu\text{s}$
1	1	0	48 マシンサイクル	$\phi = 16\text{MHz} : 3.0\mu\text{s}$
1	1	1	128 マシンサイクル	$\phi = 24\text{MHz} : 5.3\mu\text{s}$

サンプルング時間は、アナログ入力に接続される駆動インピーダンス R_{ext} に応じて設定する必要があります。下記の条件を満たさない場合は変換精度が保証されません。

- $R_{\text{ext}} = 1.5\text{k}\Omega$ の場合 :
 - ・ 4.5V $AV_{\text{CC}} < 5.5\text{V}$: サンプルング時間を $0.5\mu\text{s}$ 以上に設定してください。
 - ・ 4.0V $AV_{\text{CC}} < 4.5\text{V}$: サンプルング時間を $1.2\mu\text{s}$ 以上に設定してください。
- $R_{\text{ext}} > 1.5\text{k}\Omega$ の場合 : サンプルング時間を下式の T_{samp} 以上に設定してください。

フラッシュメモリ品

 - ・ 4.5V $AV_{\text{CC}} < 5.5\text{V}$: $T_{\text{samp}} = (2\text{k}\Omega + R_{\text{ext}}) \times 16\text{pF} \times 7$
 - ・ 4.0V $AV_{\text{CC}} < 4.5\text{V}$: $T_{\text{samp}} = (8.2\text{k}\Omega + R_{\text{ext}}) \times 16\text{pF} \times 7$

マスク ROM 品

 - ・ 4.5V $AV_{\text{CC}} < 5.5\text{V}$: $T_{\text{samp}} = (2\text{k}\Omega + R_{\text{ext}}) \times 14.4\text{pF} \times 7$
 - ・ 4.0V $AV_{\text{CC}} < 4.5\text{V}$: $T_{\text{samp}} = (8.2\text{k}\Omega + R_{\text{ext}}) \times 14.4\text{pF} \times 7$

■ コンペア時間の設定 (CT2 ~ CT0 ビット)

表 18.3-7 CT2 ~ CT0 ビットとコンペア時間の関係

CT2	CT1	CT0	コンペア時間設定	設定例 (ϕ : 内部動作周波数)
0	0	0	22 マシンサイクル	$\phi = 20\text{MHz} : 1.1\mu\text{s}$
0	0	1	33 マシンサイクル	$\phi = 24\text{MHz} : 1.4\mu\text{s}$
0	1	0	44 マシンサイクル	$\phi = 24\text{MHz} : 1.8\mu\text{s}$
0	1	1	66 マシンサイクル	$\phi = 24\text{MHz} : 2.75\mu\text{s}$
1	0	0	88 マシンサイクル	$\phi = 8\text{MHz} : 11.0\mu\text{s}$
1	0	1	132 マシンサイクル	$\phi = 16\text{MHz} : 8.25\mu\text{s}$
1	1	0	176 マシンサイクル	$\phi = 20\text{MHz} : 8.8\mu\text{s}$
1	1	1	264 マシンサイクル	$\phi = 24\text{MHz} : 11.0\mu\text{s}$

コンペア時間は、アナログ電源電圧 AV_{CC} に応じて設定する必要があります。下記の条件を満たさない場合は変換精度が保証されません。

- 4.5V $AV_{CC} < 5.5\text{V}$: コンペア時間を $1.00\mu\text{s}$ 以上に設定してください。
- 4.0V $AV_{CC} < 4.5\text{V}$: コンペア時間を $2.00\mu\text{s}$ 以上に設定してください。

18.3.5 アナログ入力許可レジスタ (ADER0/ADER1)

8/10 ビット A/D コンバータで使用するアナログ入力端子を許可または禁止します。

■ アナログ入力許可レジスタ (ADER0/ADER1)

図 18.3-6 アナログ入力許可レジスタ (ADER0/ADER1)

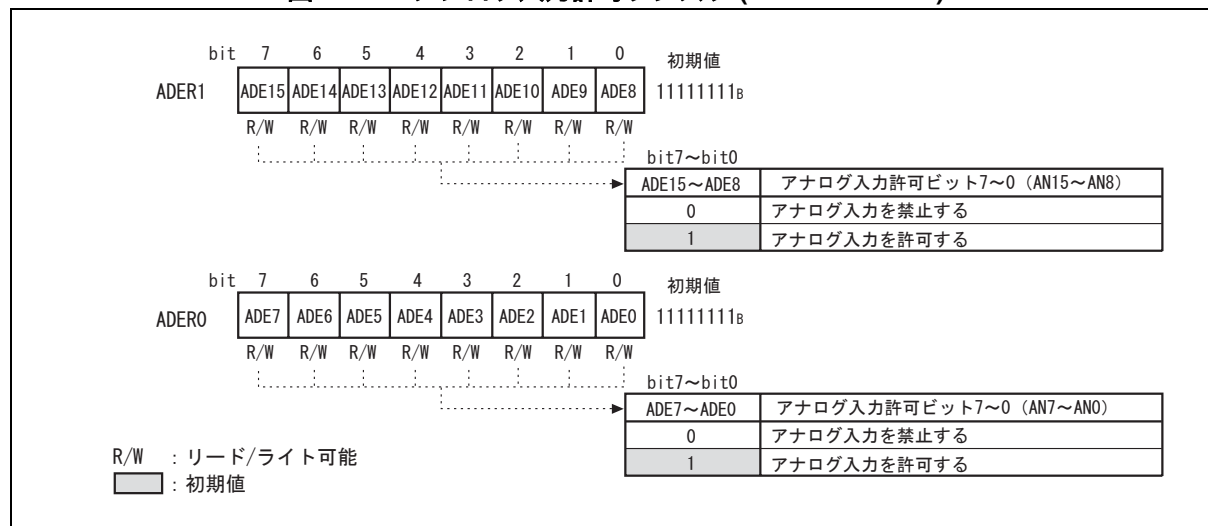


表 18.3-8 ポート 5 アナログ入力許可レジスタ (ADER1) の機能

ビット名		機能
bit7 ~ bit0	ADE15 ~ ADE8: アナログ入力許可 ビット 7 ~ 0	ポート 5 上に配置された A/D 変換アナログ入力端子 AN15 ~ AN8 のアナログ入力を許可または禁止します。 "0" に設定した場合: アナログ入力を禁止します。 "1" に設定した場合: アナログ入力を許可します。

表 18.3-9 ポート 6 アナログ入力許可レジスタ (ADER0) の機能

ビット名		機能
bit7 ~ bit0	ADE7 ~ ADE0: アナログ入力許可 ビット 7 ~ 0	ポート 6 上に配置された A/D 変換アナログ入力端子 AN7 ~ AN0 のアナログ入力を許可または禁止します。 "0" に設定した場合: アナログ入力を禁止します。 "1" に設定した場合: アナログ入力を許可します。

< 注意事項 >

- アナログ入力端子として使用する場合は，使用する端子に対応するアナログ入力許可レジスタ (ADER0/ADER1) のビットに "1" を書き込んで，アナログ入力に設定してください。
 - アナログ信号入力時にアナログ入力端子を ADEx=0 に設定することは禁止します。常に ADEx=1 に設定してください。
 - 各アナログ入力端子は，汎用入出力ポートおよび周辺機能の入出力と兼用になっています。ADEx=1 に設定された端子は，ポート方向レジスタ (DDR6/DDR7) および，各周辺機能の入出力設定にかかわらず強制的にアナログ入力端子になり，それ以外の使用はできません。
-

18.4 8/10 ビット A/D コンバータの割込み

8/10 ビット A/D コンバータでは、A/D 変換が終了して変換結果が A/D データレジスタ (ADCR) に格納されると、割込み要求が発生します。拡張インテリジェント I/O サービス (EI²OS) が利用できます。

■ A/D コンバータの割込み

アナログ入力電圧の A/D 変換が終了して、A/D 変換結果が A/D データレジスタ (ADCR) に格納されると、A/D 制御ステータスレジスタの割込み要求フラグビット (ADCS: INT) に "1" がセットされます。割込み要求の出力が許可されている場合に (ADCS: INTE=1)、割込み要求フラグビットがセットされると (ADCS: INT=1)、割込み要求が発生します。

■ 8/10 ビット A/D コンバータの割込みと EI²OS

< 参照 >

割込み番号、割込み制御レジスタ、割込みベクタアドレスについては、「第 7 章 割込み」を参照してください。

■ 8/10 ビット A/D コンバータの EI²OS

8/10 ビット A/D コンバータでは、EI²OS を使用して、A/D 変換結果を A/D データレジスタ (ADCR) からメモリに転送することができます。EI²OS 機能の使用方法につきましては「18.5.4 EI²OS 機能を使用した変換動作」、および、「18.5.5 A/D 変換データ保護機能」を参照してください。

18.5 8/10 ビット A/D コンバータの動作説明

8/10 ビット A/D コンバータの A/D 変換動作には、次の変換モードがあります。A/D 制御ステータスレジスタの A/D 変換モード選択ビット (ADCS: MD1, MD0) の設定によって、各モードを設定します。

- 単発変換モード
- 連続変換モード
- 停止変換モード

■ 単発変換モード (ADCS: MD1, MD0=00_B または 01_B)

- 起動トリガを入力すると、開始チャンネル (ADSR: ANS3 ~ ANS0) から終了チャンネル (ADSR: ANE3 ~ ANE0) までのアナログ入力を連続して A/D 変換していきます。
- 終了チャンネルの A/D 変換が終了すると、A/D 変換動作を停止します。

< 注意事項 >

- 単発変換モード 1 (ADCS:MD1, MD0=00_B) では A/D 変換中または一時停止状態^{*}のときに起動トリガが入力されると、8/10 ビット A/D コンバータが再起動する場合がありますので、A/D 変換中または一時停止状態のときは起動トリガを入力しないでください。
単発変換モード 2 (ADCS:MD1, MD0=01_B) では A/D 変換中または一時停止状態^{*}のときに起動トリガが入力されても、8/10 ビット A/D コンバータが再起動することはありません。
 - 単発変換モード 1, 単発変換モード 2 いずれの場合も再起動する場合は「18.5.1 単発変換モード」に示す手順で再起動を行ってください。
- ^{*}: 一時停止状態は A/D 変換保護機能が動作して変換が一時停止している状態です。詳細は「18.5.5 A/D 変換データ保護機能」を参照してください。

■ 連続変換モード (ADCS: MD1, MD0=10_B)

- 起動トリガを入力すると、開始チャンネル (ADSR: ANS3 ~ ANS0) から終了チャンネル (ADSR: ANE3 ~ ANE0) までのアナログ入力を連続して A/D 変換していきます。
- 終了チャンネルの A/D 変換が終了すると、開始チャンネルのアナログ入力に戻って A/D 変換を継続して行います。

■ 停止変換モード (ADCS: MD1, MD0=11_B)

- 起動トリガを入力すると、開始チャンネル (ADSR: ANS3 ~ ANS0) の A/D 変換が開始されます。1 チャンネルの A/D 変換が終了すると、A/D 変換動作は停止します。この状態を " 停止状態 " といいます。A/D 変換動作が停止している間に起動トリガを入力すると、次のチャンネルの A/D 変換が行われます。
- 終了チャンネルの A/D 変換が終了すると、A/D 変換動作は停止します。A/D 変換動作が停止している間に起動トリガを入力すると、開始チャンネルのアナログ入力に戻って A/D 変換を継続します。

18.5.1 単発変換モード

単発変換モードでは、開始チャンネルから終了チャンネルまで順次 A/D 変換を行います。終了チャンネルの A/D 変換が終了すると、A/D 変換動作を停止します。

■ 単発変換モードの設定

8/10 ビット A/D コンバータを単発変換モードで動作させるには、図 18.5-1 の設定が必要です。

図 18.5-1 単発変換モードの設定

	bit15	14	13	12	11	10	9	bit8	bit7	6	5	4	3	2	1	bit0
ADCS	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	—	MD1	MD0	S10	—	—	—	—	予約
	◎	◎	◎	◎	◎	◎	◎		0	◎	◎					0
ADCR	—	—	—	—	—	—	D9～D0（変換結果を保持）									
ADSR	ST2	ST1	ST0	CT2	CT1	CT0	予約	ANS3	ANS2	ANS1	ANS0	予約	ANE3	ANE2	ANE1	ANE0
	◎	◎	◎	◎	◎	◎	0	◎	◎	◎	◎	0	◎	◎	◎	◎
ADER1	ADE15	ADE14	ADE13	ADE12	ADE11	ADE10	ADE9	ADE8								
	○	○	○	○	○	○	○	○								
ADER0	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0								
	○	○	○	○	○	○	○	○								

— : 未定義
 ◎ : 使用ビット
 ○ : アナログ入力端子として使用する端子に対応するビットに"1"を設定
 0 : "0"を設定

■ 単発変換モードの動作と使用方法

- 起動トリガを入力すると、A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルから A/D 変換を開始し、A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) で設定されたチャンネルまで連続して A/D 変換を行います。
- A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) で設定されたチャンネルの A/D 変換が終了すると、A/D 変換動作を停止します。
- A/D 変換動作を強制終了するには、A/D 制御ステータスレジスタの A/D 変換動作中フラグビット (ADCS:BUSY) に "0" を書き込みます。

[開始チャンネルと終了チャンネルが同じ場合]

- 開始チャンネルと終了チャンネルを同じチャンネル番号に設定した場合は (ADSR: ANS3 ~ ANS0=ADSR: ANE3 ~ ANE0), 開始チャンネル (= 終了チャンネル) として設定した 1 チャンネルだけを 1 回だけ A/D 変換して終了します。

[単発変換モードでの変換順序]

単発変換モードでの変換順序の例を表 18.5-1 に示します。

表 18.5-1 単発変換モードでの変換順序

開始チャンネル	終了チャンネル	単発変換モードでの変換順序
AN0 端子 (ADSR: ANS=0000 _B)	AN3 端子 (ADSR: ANE=0011 _B)	AN0 AN1 AN2 AN3 終了
AN3 端子 (ADSR: ANS=0011 _B)	AN3 端子 (ADSR: ANE=0011 _B)	AN3 終了

[再起動について]

A/D 変換実行中および、一時停止状態中に A/D 変換を再起動する場合は、変換をいったん強制終了させてから再度起動させます。以下の手順で実施してください。

- 1) A/D 変換動作中フラグビット (ADCS:BUSY) をクリア
- 2) 割込み要求フラグビット (ADCS:INT) をクリア
- 3) A/D 変換ソフトウェア起動ビット (ADCS:STRT) をセット

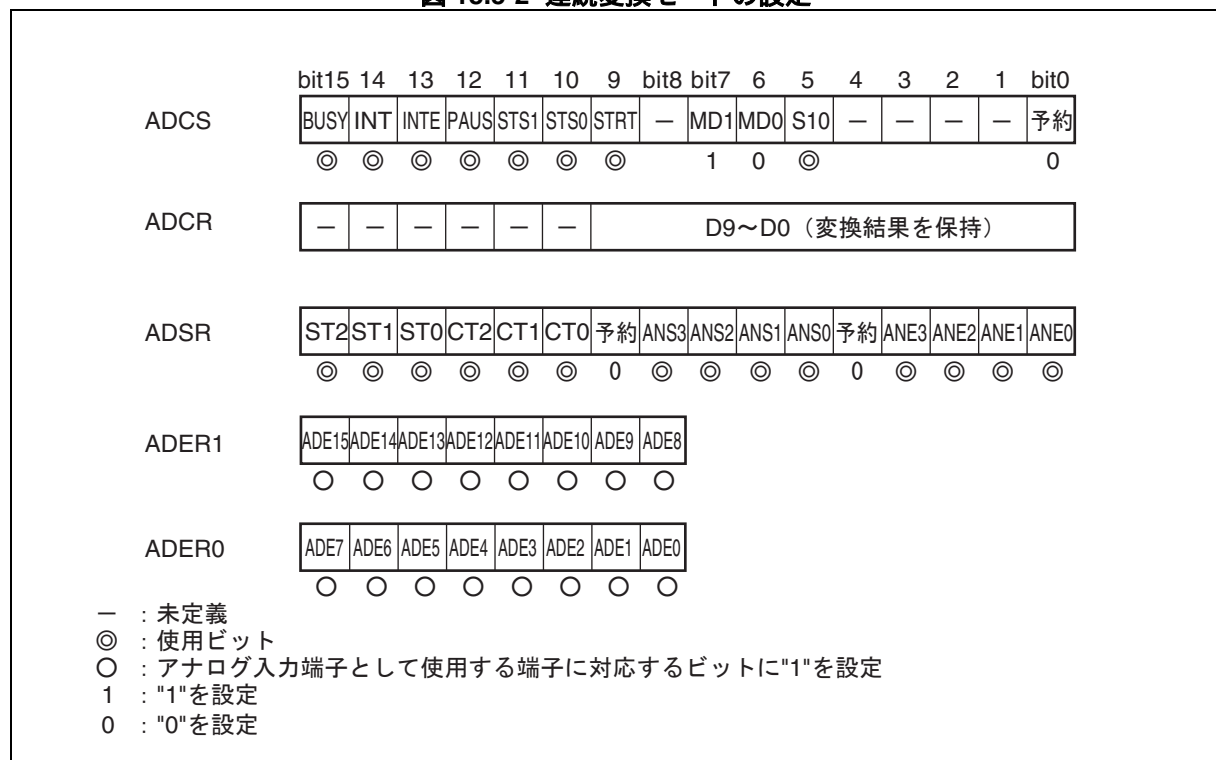
18.5.2 連続変換モード

連続変換モードでは、開始チャンネルから終了チャンネルまで順次 A/D 変換を行います。終了チャンネルの A/D 変換が終了すると、開始チャンネルに戻って A/D 変換動作を継続します。

■ 連続変換モードの設定

8/10 ビット A/D コンバータを連続変換モードで動作させるには、図 18.5-2 の設定が必要です。

図 18.5-2 連続変換モードの設定



■ 連続変換モードの動作と使用方法

- 起動トリガを入力すると、A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルから A/D 変換を開始し、A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) で設定されたチャンネルまで連続して A/D 変換を行います。
- A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) で設定されたチャンネルの A/D 変換が終了すると、A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルに戻って A/D 変換を継続します。
- A/D 変換動作を強制終了するには、A/D 制御ステータスレジスタの A/D 変換動作中フラグビット (ADCS:BUSY) に "0" を書き込みます。

[開始チャンネルと終了チャンネルが同じ場合]

- 開始チャンネルと終了チャンネルを同じチャンネルに設定した場合は (ADSR: ANS3 ~ ANS0=ADSR: ANE3 ~ ANE0), 開始チャンネル (=終了チャンネル) として設定した 1 チャンネルの A/D 変換を繰り返して行います。

〔連続変換モードでの変換順序〕

連続変換モードでの変換順序の例を表 18.5-2 に示します。

表 18.5-2 連続変換モードでの変換順序

開始チャネル	終了チャネル	連続変換モードでの変換順序
AN0 端子 (ADSR: ANS=0000 _B)	AN3 端子 (ADSR: ANE=0011 _B)	AN0 AN1 AN2 AN3 AN0 繰返し
AN3 端子 (ADSR: ANS=0011 _B)	AN3 端子 (ADSR: ANE=0011 _B)	AN3 AN3 繰返し

〔再起動について〕

A/D 変換実行中および、一時停止状態中に A/D 変換を再起動する場合は、変換をいったん強制終了させてから再度起動させます。以下の手順で実施してください。

- 1) A/D 変換動作中フラグビット (ADCS:BUSY) をクリア
- 2) 割込み要求フラグビット (ADCS:INT) をクリア
- 3) A/D 変換ソフトウェア起動ビット (ADCS:STRT) をセット

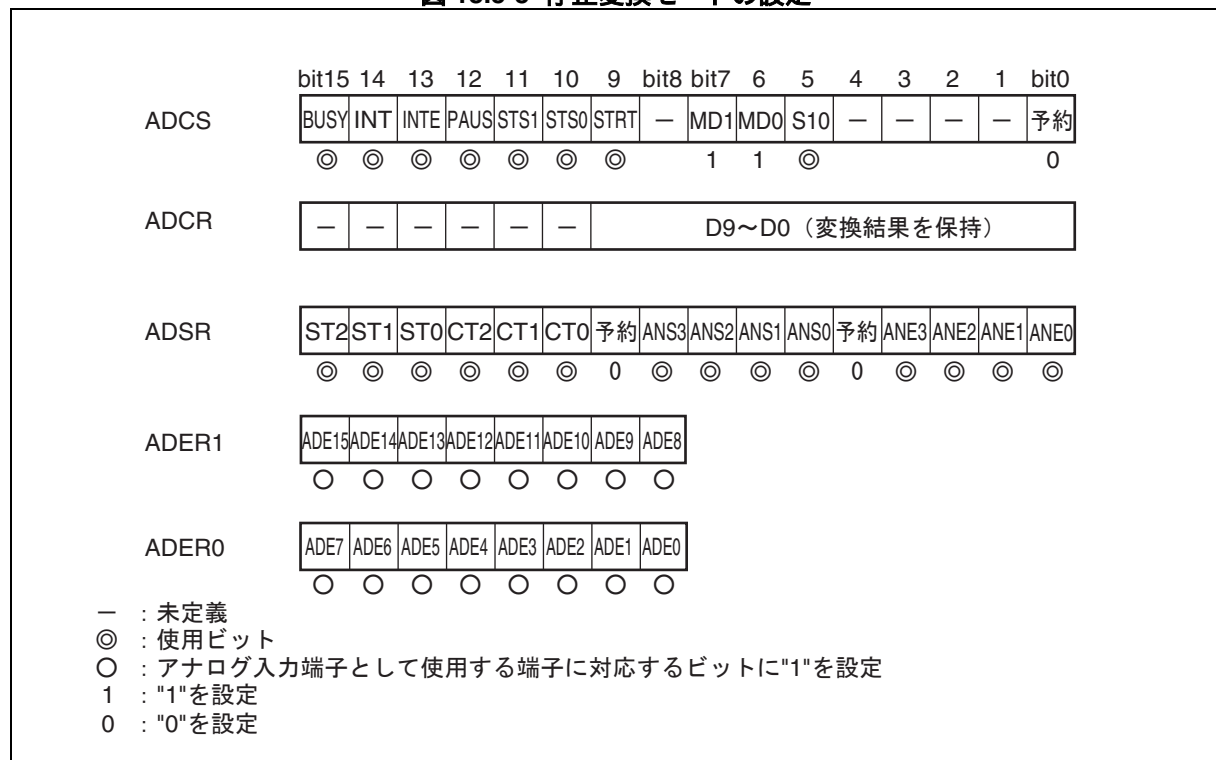
18.5.3 停止変換モード

停止変換モードでは、1 チャンネルごとに停止と起動を繰り返しながら A/D 変換を行います。終了チャンネルの A/D 変換が終了して A/D 変換動作が停止した後、起動トリガを入力すると開始チャンネルに戻って A/D 変換を継続します。

■ 停止変換モードの設定

8/10 ビット A/D コンバータを停止変換モードで動作させるには、図 18.5-3 の設定が必要です。

図 18.5-3 停止変換モードの設定



■ 停止変換モードの動作と使用方法

- 起動トリガを入力すると、A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルから A/D 変換を開始します。1 チャンネルの A/D 変換が終了すると A/D 変換動作は停止します。A/D 変換動作が停止している間に起動トリガを入力すると、次のチャンネルの A/D 変換を行います。
- A/D 変換終了チャンネル選択ビット (ANE3 ~ ANE0) で設定されたチャンネルの A/D 変換が終了すると、A/D 変換動作は停止します。A/D 変換動作が停止している間に起動トリガを入力すると、A/D 変換開始チャンネル選択ビット (ANS3 ~ ANS0) で設定されたチャンネルに戻って A/D 変換を継続します。
- A/D 変換動作を強制終了するには、A/D 制御ステータスレジスタの A/D 変換動作中フラグビット (ADCS:BUSY) に "0" を書き込みます。

〔開始チャンネルと終了チャンネルが同じ場合〕

開始チャンネルと終了チャンネルを同じチャンネルに設定した場合は (ADSR:ANS3 ~ ANS0=ADSR: ANE3 ~ ANE0), 開始チャンネル (= 終了チャンネル) として設定した 1 チャンネルの A/D 変換と停止を繰り返します。

〔停止変換モードでの変換順序〕

停止変換モードでの変換順序の例を表 18.5-3 に示します。

表 18.5-3 停止変換モードでの変換順序

開始チャンネル	終了チャンネル	単発変換モードでの変換順序
AN0 端子 (ADSR: ANS=0000 _B)	AN3 端子 (ADSR: ANE=0011 _B)	AN0 停止・起動 AN1 停止・起動 AN2 停止・起動 AN3 停止・起動 AN0 繰返し
AN3 端子 (ADSR: ANS=0011 _B)	AN3 端子 (ADSR: ANE=0011 _B)	AN3 停止・起動 AN3 停止・起動 繰返し

〔再起動について〕

A/D 変換実行中および、一時停止状態中に A/D 変換を再起動する場合は、変換をいったん強制終了させてから再度起動させます。以下の手順で実施してください。

- 1) A/D 変換動作中フラグビット (ADCS:BUSY) をクリア
- 2) 割込み要求フラグビット (ADCS:INT) をクリア
- 3) A/D 変換ソフトウェア起動ビット (ADCS:STRT) をセット

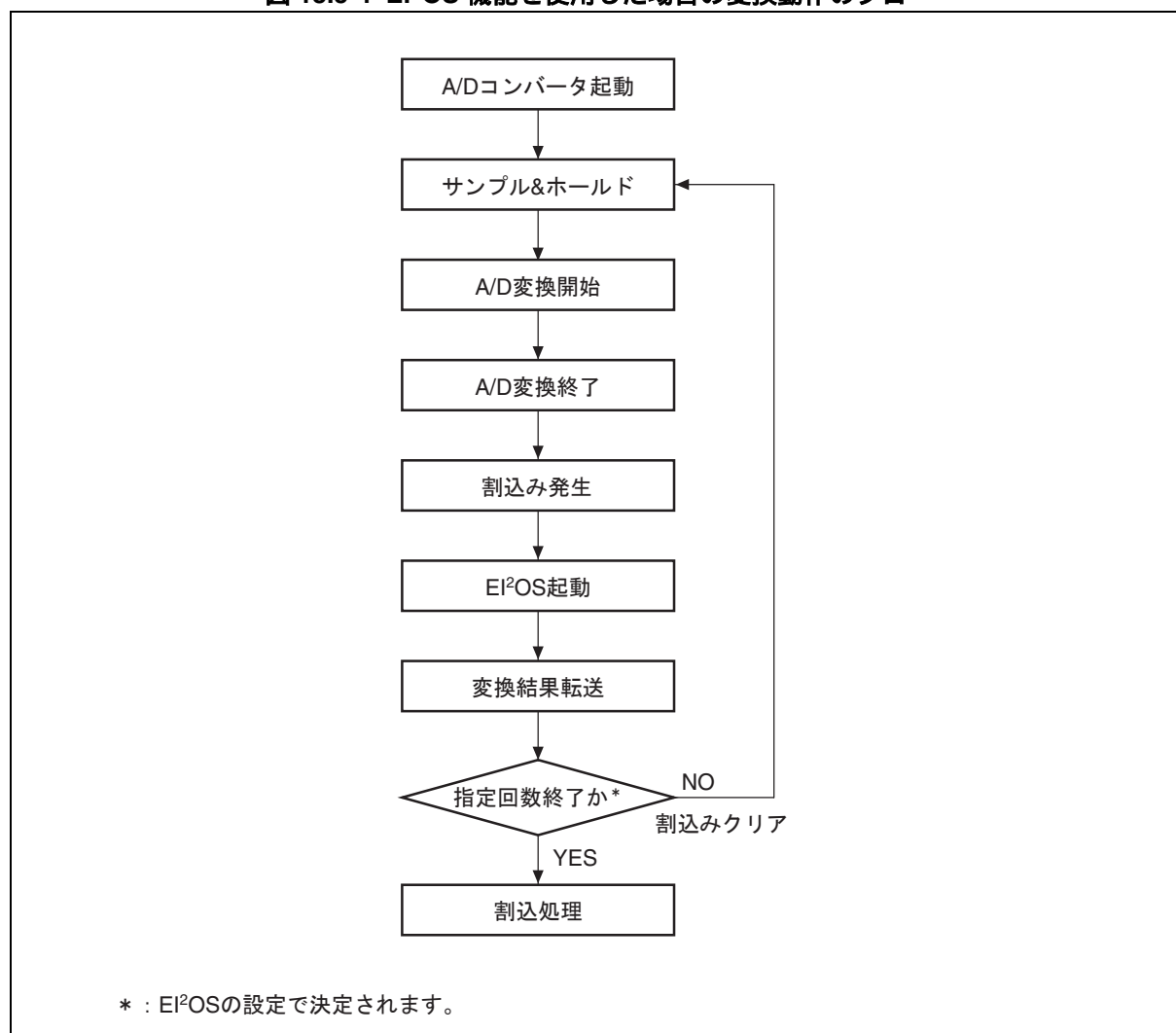
18.5.4 EI²OS 機能を使用した変換動作

8/10 ビット A/D コンバータでは、EI²OS 機能を使用して、A/D 変換結果をメモリに転送することができます。

■ EI²OS 機能を使用した変換動作

EI²OS 機能を使用した場合の変換動作のフローを図 18.5-4 に示します。

図 18.5-4 EI²OS 機能を使用した場合の変換動作のフロー



18.5.5 A/D 変換データ保護機能

割込み要求の出力を許可した状態で A/D 変換を実行すると、データ保護機能が働きます。

■ 8/10 ビット A/D コンバータの A/D 変換データ保護機能の説明

A/D 変換データ保護機能は A/D 変換データの取りこぼしを防ぐ機能です。

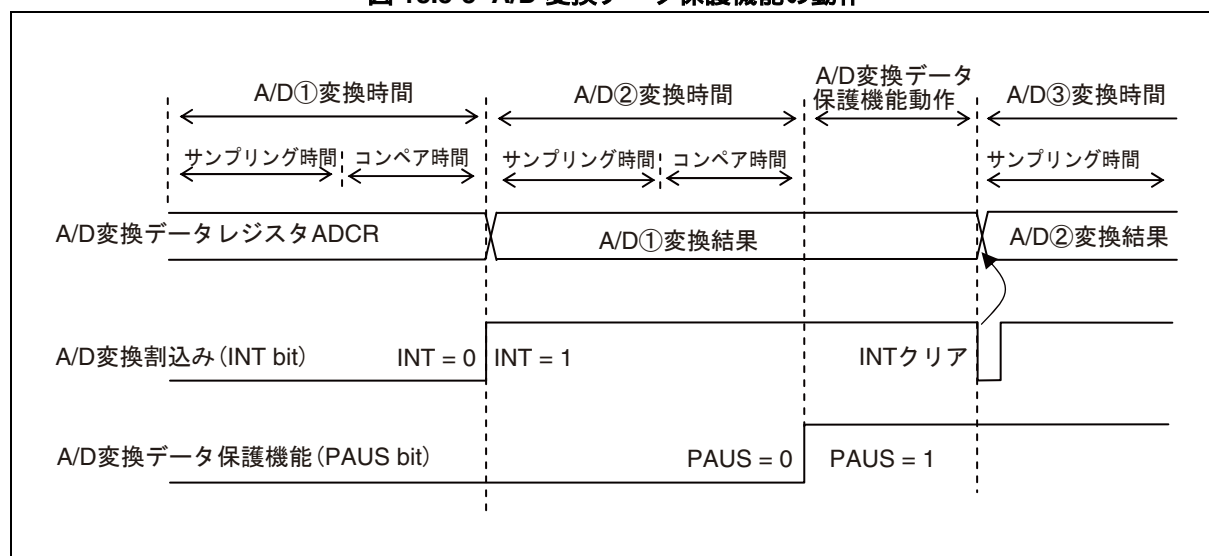
8/10 ビット A/D コンバータには、変換データ格納用の A/D データレジスタ (ADCR1/ADCR0)1 個と現在 A/D 変換中のデータを格納する逐次比較回路 1 個があります。

A/D 変換実行中、8/10 ビット A/D コンバータは変換データを 1 ビットずつ逐次比較回路に格納し、A/D 変換が完了すると A/D 変換結果を A/D データレジスタに格納します。

A/D 変換データ保護機能を使用する場合、使用しない場合によって、8/10 ビット A/D コンバータの動作は下記のようになります。

- 割込み要求許可ビット (ADCS:INTE) = 0 に設定すると、データ保護機能は無効になります。この場合、A/D 変換が連続して行われると、8/10 ビット A/D コンバータは変換が終了するたびに A/D データレジスタに変換結果を格納します。(常に最新の变換データが格納されることになります。)
- 割込み要求許可ビット (ADCS:INTE) = 1 に設定すると、データ保護機能が有効になります。この状態で A/D 変換が連続して行われると、はじめの変換が終了したときに割込み要求フラグビット : ADCS:INT = 1 になります。さらに、次の A/D 変換が行われ、INT = 1 の状態で変換終了すると、変換結果を逐次比較回路から A/D データレジスタに転送する直前で 8/10 ビット A/D コンバータは "一時停止状態" となり、変換データの上書きを防ぎます。このとき、A/D 制御ステータスレジスタの一時停止フラグビット (ADCS: PAUS) に "1" がセットされます。一時停止状態中に割込み要求フラグビット (ADCS:INT) を "0" にクリアすると逐次比較回路に格納されているデータは A/D データレジスタへ転送されます (図 18.5-5 を参照)。

図 18.5-5 A/D 変換データ保護機能の動作



● CPU で A/D 変換結果を読み出す場合の A/D 変換データ保護機能

- アナログ入力 A/D 変換された後、A/D 変換結果が A/D データレジスタ (ADCR) に格納されると、A/D 制御ステータスレジスタの割込み要求フラグビット (ADCS: INT) に "1" がセットされます。
- 次の回の A/D 変換が終了した時点で、前回の A/D 変換終了時にセットされた 割込み要求フラグビット (ADCS: INT) がセットされたままの場合、割込み要求が許可されていれば (ADCS: INTE=1), A/D 変換動作は新しいデータを A/D データレジスタに上書きする直前でデータ保護のため一時停止状態となります。
- A/D 制御ステータスレジスタの割込み要求が許可されているため (ADCS: INTE=1), INT ビットがセットされると割込み要求が発生します。INT ビットをクリアすると、A/D 変換動作の一時停止状態が解除されます。
- 連続して A/D 変換を行っている場合、8/10 ビット A/D コンバータは次の A/D 変換動作を開始します。このとき、一時停止フラグビット (ADCS: PAUS) は自動的に "0" にクリアされません。クリアするためには同ビットに "0" を書き込んでください。

< 注意事項 >

- 一時停止状態中に割込み要求の出力を禁止すると (ADCS: INTE=0), A/D 変換が開始され、A/D データレジスタのデータが書き換えられてしまう場合があります。
- 複数回の A/D 変換を連続実行する場合、割込み要求フラグビット (ADCS: INT) をクリアする前に必ず A/D データレジスタに格納されたデータを読み出してください。A/D 変換が一時停止の状態 A/D データレジスタに格納されたデータを読み出す前に割込み要求フラグビット (ADCS: INT) をクリアすると最初に格納された変換データが次の変換データによって上書きされ破壊されます。

● EI²OS で A/D 変換結果を転送する場合の A/D 変換データ保護機能

EI²OS 機能を利用して、A/D 変換後、A/D 変換結果を A/D データレジスタからメモリに転送している間に次の回の A/D 変換が終了した場合、データ保護のため A/D 変換動作は新しいデータを A/D データレジスタに上書きする直前で一時停止状態になります。A/D 変換動作が停止すると、A/D 制御ステータスレジスタの一時停止フラグビット (ADCS: PAUS) に "1" がセットされます。

EI²OS 機能による A/D 変換結果のメモリ転送が終了すると、A/D 変換の一時停止状態が解除されます。連続して A/D 変換を行っている場合は、A/D 変換動作が再開されます。このとき、一時停止フラグビット (ADCS: PAUS) は自動的に "0" にクリアされません。クリアする場合は同ビットに "0" を書き込んでください。

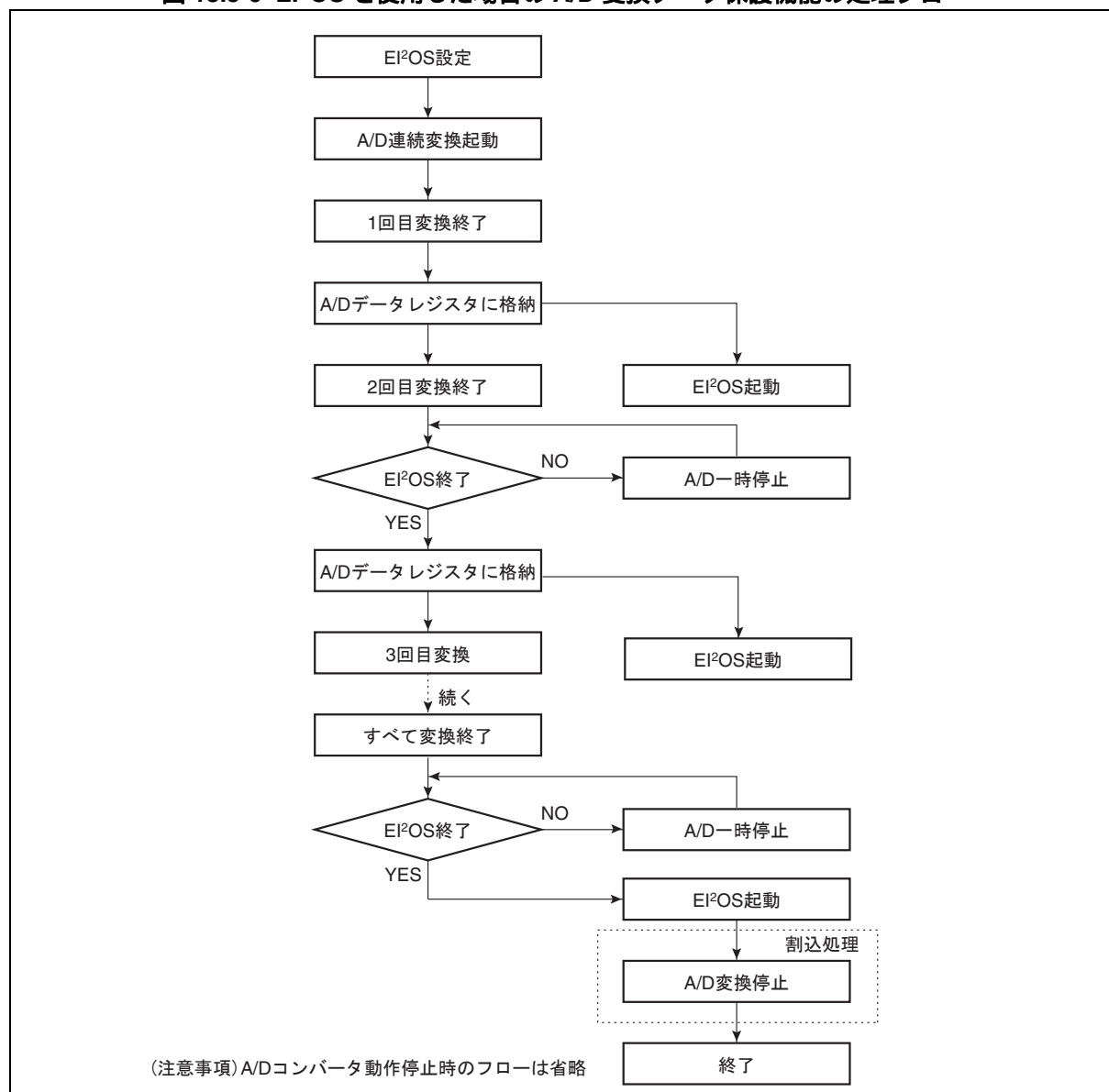
< 注意事項 >

- EI²OS 機能によって A/D 変換結果がメモリ転送されている場合、CPU から割込み要求フラグビットをクリア (ADCS:INT=0) しないでください。転送中の A/D データレジスタのデータが書き換えられてしまう場合があります。
- EI²OS 機能によって A/D 変換結果がメモリ転送されている場合、割込み要求の出力を禁止しないでください。一時停止中に割込み要求の出力を禁止すると (ADCS:INTE=0), A/D 変換が開始され、転送中の A/D データレジスタのデータが書き換えられてしまう場合があります。
- EI²OS 機能によって A/D 変換結果がメモリ転送されている場合に再起動をかけないでください。A/D 変換一時停止中に再起動をかけると、変換結果が壊れる場合があります。

● EI²OS を使用した場合の A/D 変換データ保護機能の処理フロー

EI²OS を使用した場合の A/D 変換データ保護機能の処理フローを図 18.5-6 に示します。

図 18.5-6 EI²OS を使用した場合の A/D 変換データ保護機能の処理フロー



18.6 8/10 ビット A/D コンバータ使用上の注意

8/10 ビット A/D コンバータを使用する場合は、次の点に注意してください。

■ 8/10 ビット A/D コンバータ使用上の注意

● アナログ入力端子について

- アナログ入力端子は、ポート 6, 7 の汎用入出力ポートと兼用になっています。アナログ入力端子として使用する場合はアナログ入力許可レジスタ (ADER0/ADER1) の設定によってアナログ入力端子に切り換えてください。
- アナログ入力端子として使用する場合は、使用する端子に対応するアナログ入力許可レジスタ (ADER0/ADER1) のビットに "1" を書き込んで、アナログ入力許可に設定してください。
- 汎用入出力ポートに設定したままで中間レベルの信号が入力されると、ゲートに入力リーク電流が流れます。アナログ入力端子として使用する場合は、必ずアナログ入力許可に設定して使用してください。

● 内部タイマまたは外部トリガで起動する場合の注意

8/10 ビット A/D コンバータを内部タイマ出力または外部トリガで起動するように A/D 制御ステータスレジスタの A/D 起動トリガ選択ビット (ADCS: STS1, STS0) を設定する場合は、タイマ出力および外部トリガのレベルをインアクティブ側 (外部トリガの場合は "H" 側) に設定してください。起動トリガの入力値をアクティブ側に設定しておく、A/D 制御ステータスレジスタの A/D 起動トリガ選択ビット (ADCS: STS1, STS0) の設定と同時に動作を開始するおそれがあります。

● 8/10 ビット A/D コンバータの電源・アナログ入力の投入順序

- 8/10 ビット A/D コンバータの電源、アナログ入力 (AN0 ~ AN15 端子) の印加は、必ずデジタル電源 (V_{CC}) の投入後に行ってください。
- 電源切断時は、8/10 ビット A/D コンバータの電源およびアナログ入力の遮断の後でデジタル電源の遮断を行ってください。
- AVR は、 AV_{CC} を超えないように投入および切断を行ってください (アナログ電源とデジタル電源を同時に投入、遮断することは問題ありません)。

● 8/10 ビット A/D コンバータの電源電圧について

ラッチアップ防止のため、8/10 ビット A/D コンバータの電源 (AV_{CC}) は、デジタル電源 (V_{CC}) の電圧を超えないように注意してください。

第19章

D/A コンバータ

D/A コンバータの機能と動作について説明します。

19.1 D/A コンバータの概要

19.2 D/A コンバータのブロックダイアグラム

19.3 D/A コンバータの構成

19.4 D/A コンバータレジスタ

19.1 D/A コンバータの概要

D/A コンバータは、8 ビットデジタル入力を、R-2R 方式によってアナログ出力に変換をします。

D/A コンバータは 2 チャンネルを内蔵しており、D/A 制御レジスタによって各チャンネルごと個々に出力を制御できます。

■ D/A コンバータの機能と動作

D/A コンバータは、8 ビットデジタル入力からアナログ出力を発生させる回路です。D/A 制御レジスタ (DACR) の許可ビットに "1" を設定すると、対応する D/A 出力チャンネルが許可されます。したがって、このビットに "0" を設定すると、チャンネルは禁止されます。

D/A 出力が禁止された場合、D/A コンバータチャンネルの出力部に直列に挿入されたアナログスイッチがオフになります。D/A コンバータでは、ビットは "0" にクリアされ、直流電流の流れる経路が遮断されます。また、停止モードでも同様になります。

D/A コンバータの出力電圧の範囲は、 $0V \sim 255/256 \times AV_{CC}$ です。出力電圧の範囲を変更するには、 AV_{CC} 電圧を外部で調節してください。

D/A コンバータの出力には、バッファアンプを内蔵していません。また、出力部にアナログスイッチ ($= 100\Omega$) が直列に内蔵されていますので、外部の出力負荷には、十分な安定時間を考慮してください。

D/A コンバータの出力電圧の理論値を、表 19.1-1 に示します。

表 19.1-1 D/A コンバータの出力電圧の理論値

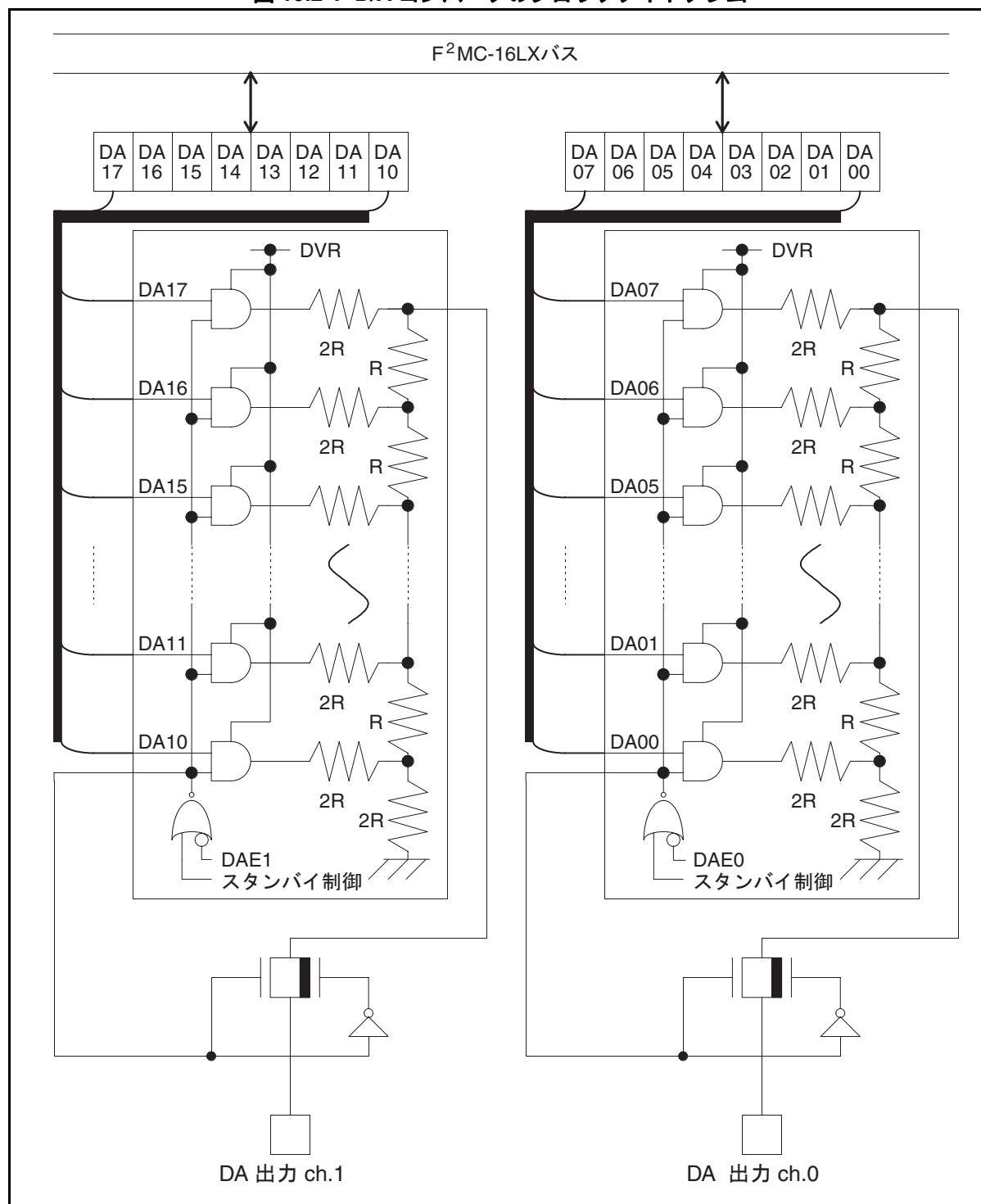
DA07 ~ DA00, DA17 ~ DA10 に書き込んだ値	出力電圧の理論値
00 _H	$0/256 \times AV_{CC}(= 0V)$
01 _H	$1/256 \times AV_{CC}$
02 _H	$2/256 \times AV_{CC}$
:	:
FD _H	$253/256 \times AV_{CC}$
FE _H	$254/256 \times AV_{CC}$
FF _H	$255/256 \times AV_{CC}$

19.2 D/A コンバータのブロックダイアグラム

D/A コンバータのブロックダイアグラムを示します。

■ D/A コンバータのブロックダイアグラム

図 19.2-1 D/A コンバータのブロックダイアグラム



19.3 D/A コンバータの構成

D/A コンバータの端子，ブロックダイアグラムを示します。

■ D/A コンバータの端子

D/A コンバータで使用する端子は，汎用入出力ポートと兼用端子になっています。

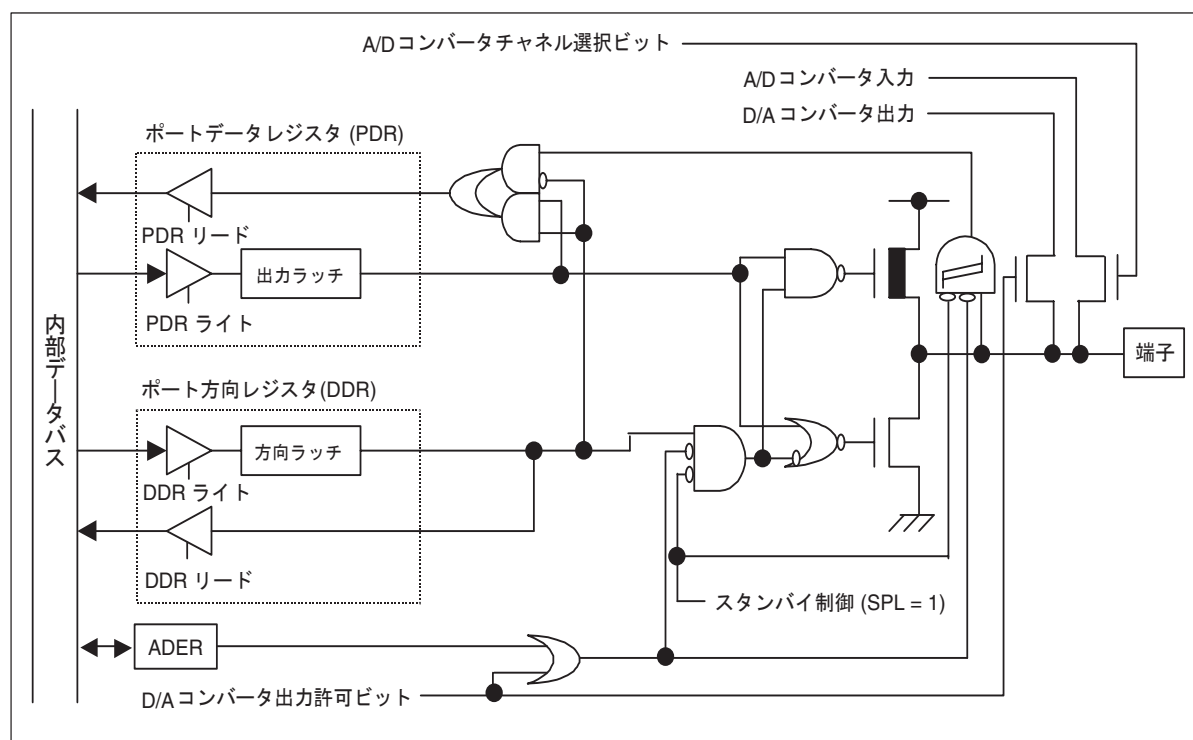
端子の機能，入出力形式および D/A コンバータで使用する場合の設定を表 19.3-1 に示します。

表 19.3-1 D/A コンバータの端子

端子名	端子機能	入出力形式	プルアップ 設定	スタンバイ	D/A コンバータの 使用に必要な設定
P70/DA0/ AN8	ポート 7 入出力 / アナログ / D/A コンバータ端子	アナログ /CMOS 出力 /CMOS ヒステリシス入力	なし	あり	DACR0 : DAE0 = 1
P71/DA1/ AN9					DACR1 : DAE1 = 1

■ 端子のブロックダイアグラム

図 19.3-1 D/A コンバータの端子のブロックダイアグラム



19.4 D/A コンバータレジスタ

D/A コンバータには、2 種類のレジスタがあります。

- D/A コンバータレジスタ (DAT0, DAT1)
- D/A 制御レジスタ (DACR0, DACR1)

■ D/A コンバータレジスタ

図 19.4-1 D/A コンバータレジスタ

D/A データレジスタ 1								
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
DA17	DA16	DA15	DA14	DA13	DA12	DA11	DA10	XXXXXXXX _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
D/A データレジスタ 0								
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
DA07	DA06	DA05	DA04	DA03	DA02	DA01	DA00	XXXXXXXX _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
D/A 制御レジスタ 1								
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	初期値
-	-	-	-	-	-	-	DAE1	-----0 _B
-	-	-	-	-	-	-	R/W	
D/A 制御レジスタ 0								
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
-	-	-	-	-	-	-	DAE0	-----0 _B
-	-	-	-	-	-	-	R/W	

19.4.1 D/A コンバータレジスタ 1(DAT1)

D/A コンバータレジスタ 1(DAT1) は、アナログ出力に変換されるデジタル入力データを ch.1 に設定するレジスタです。

■ D/A コンバータレジスタ 1(DAT1)

図 19.4-2 D/A コンバータレジスタ 1(DAT1)

D/A コンバータレジスタ 1								初期値
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
DA17	DA16	DA15	DA14	DA13	DA12	DA11	DA10	XXXXXXXX _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W: リード / ライト可能
X : 不定

19.4.2 D/A コンバータレジスタ 0(DAT0)

D/A コンバータレジスタ 0(DAT0) は , ch.0 にアナログ出力に変換されるデジタル入力データを設定するレジスタです。

■ D/A コンバータレジスタ 0(DAT0)

図 19.4-3 D/A コンバータレジスタ 0(DAT0)

D/A コンバータレジスタ 0								初期値
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
DA07	DA06	DA05	DA04	DA03	DA02	DA01	DA00	XXXXXXXX _B
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W: リード / ライト可能
X : 不定

19.4.3 D/A 制御レジスタ 1(DACR1)

D/A 制御レジスタ 1 は, ch.1 に出力許可信号を保持します。

■ D/A 制御レジスタ 1(DACR1)

図 19.4-4 D/A 制御レジスタ 1(DACR1)

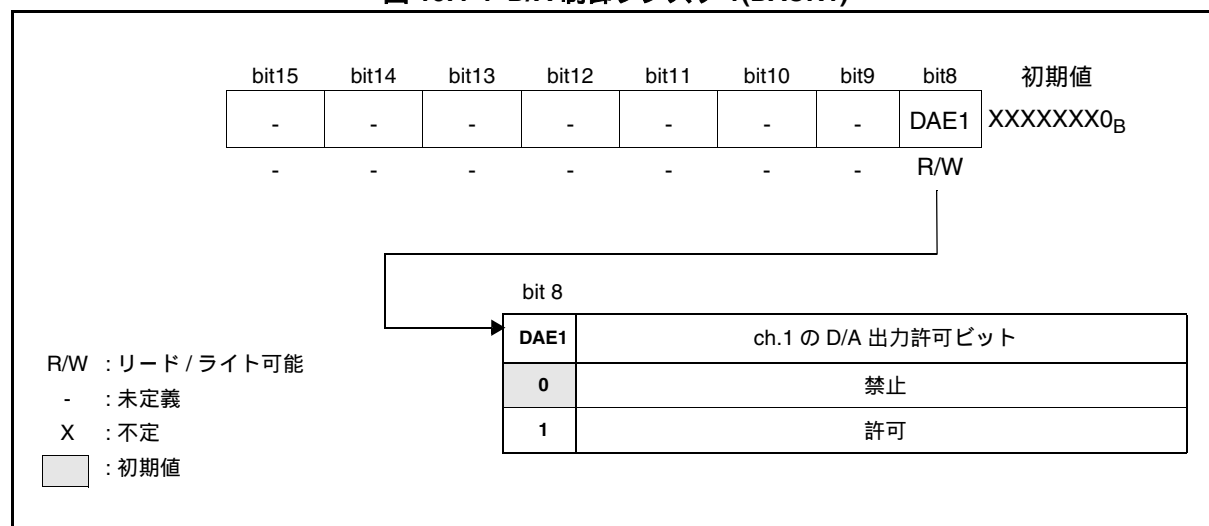


表 19.4-1 D/A 制御レジスタ 1(DACR1)

ビット名		機能
bit15 ~ bit9	未定義ビット	読み出した場合 : 値は不定です。 書き込んだ場合 : 影響しません。
bit8	DAE1: ch.1 の D/A 出力許可ビット	D/A コンバータ出力を許可または禁止に設定します。DAE1 は ch.1 に使用されます。 "1" に設定した場合 : D/A 出力が許可されます。 "0" に設定した場合 : D/A 出力が禁止されます。 リセットの場合 : "0" に初期化されます。 リードとライト可能です。

19.4.4 D/A 制御レジスタ 0(DACR0)

D/A 制御レジスタ 0 は , ch.0 に出力許可信号を保持します。

■ D/A 制御レジスタ 0(DACR0)

図 19.4-5 D/A 制御レジスタ 0(DACR0)

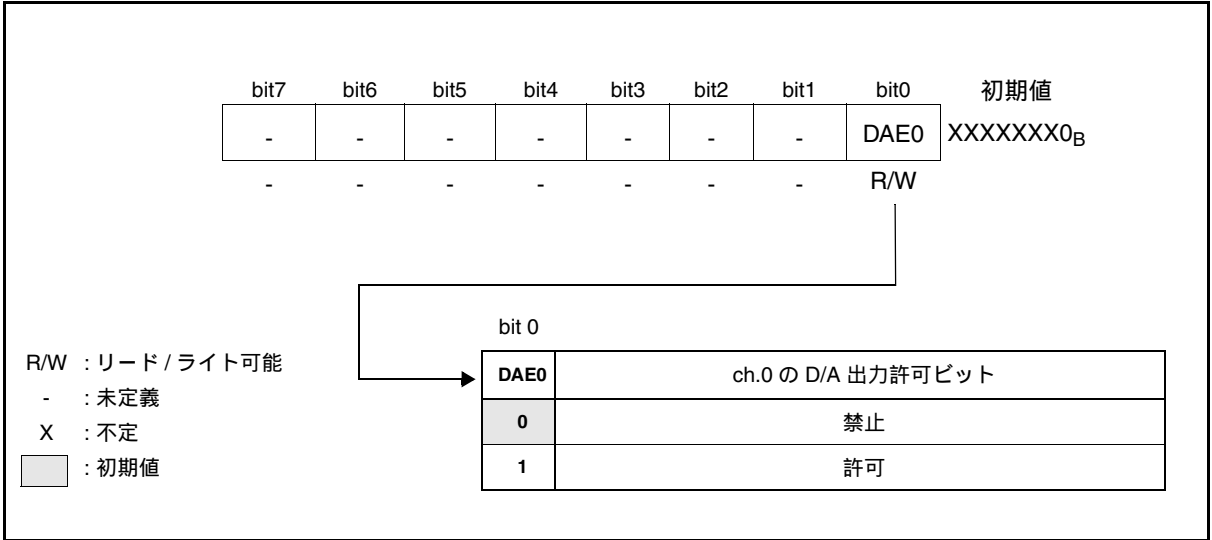


表 19.4-2 D/A 制御レジスタ 0(DACR0)

ビット名		機能
bit7 ~ bit1	未定義ビット	読み出した場合 : 値は不定です。 書き込んだ場合 : 影響しません。
bit0	DAE0: ch.0 の D/A 出力許可ビット	D/A コンバータ出力を許可または禁止に設定します。DAE0 は ch.0 に使用されます。 "1" に設定した場合 : D/A 出力が許可されます。 "0" に設定した場合 : D/A 出力が禁止されます。 リセットの場合 : "0" に初期化されます。 リードとライト可能です。

第20章

UART

UART の機能の動作について説明します。

- 20.1 UART の概要
- 20.2 UART のブロックダイアグラム
- 20.3 UART の端子
- 20.4 UART のレジスタ
- 20.5 UART の割込み
- 20.6 UART のボーレート
- 20.7 UART の動作説明
- 20.8 UART 使用上の注意

20.1 UART の概要

UART は、外部と同期通信または非同期通信（調歩同期）を行うための、汎用シリアルデータ通信インタフェースです。

- クロック同期、双方向通信機能（ノーマルモード）を備えています。
- マスタ/スレーブ型通信機能（マルチプロセッサモード）を備えています（マスタ側のみ使用可能）。

■ UART の機能（× 2）

● UART の機能

UART は、ほかの CPU および外部装置とシリアルデータの送受信を行うための汎用シリアルデータ通信インタフェースで、表 20.1-1 の機能を備えています。

表 20.1-1 UART の機能

	機能
データバッファ	全二重ダブルバッファ
転送モード	<ul style="list-style-type: none"> • クロック同期 • クロック非同期（調歩同期）
ボーレート	<ul style="list-style-type: none"> • 専用ボーレートジェネレータ使用可能、8 種類の値を選択可。 • 外部クロック入力可能 • 内部クロック（16 ビットリロードタイマ 0, 1 から供給されるクロックについては、UART0 には 16 ビットリロードタイマ 0 のクロックを供給、UART1 には 16 ビットリロードタイマ 1 のクロックが供給されます。
データ長	<ul style="list-style-type: none"> • 7 ビット（非同期ノーマルモードのみ） • 8 ビット
信号方式	NRZ(Non Return to Zero) 方式
受信エラー検出	<ul style="list-style-type: none"> • フレーミングエラー • オーバランエラー • パリティエラー（マルチプロセッサモード時は検出不可）
割込み要求	<ul style="list-style-type: none"> • 受信割込み（受信完了、受信エラー検出） • 送信割込み（送信完了） • 送受信とも拡張インテリジェント I/O サービス (EI²OS) に対応
マスタ/スレーブ型通信機能（マルチプロセッサモード）	1(マスタ) 対 n(スレーブ) 間の通信が可能（マスタとしてのみ使用できます。）

< 注意事項 >

クロック同期転送時にスタートビット/ストップビットは付加されず、データのみを転送します。

表 20.1-2 UART の動作モード

動作モード		データ長		同期方式	ストップビット長
		パリティあり	パリティなし		
0	ノーマルモード	7 ビットまたは 8 ビット		非同期	1 ビットまたは 2 ビット ^{*2}
1	マルチプロセッサモード	8+1 ^{*1} ビット	-	非同期	
2	ノーマルモード	8 ビット	-	同期	なし

- : 設定不可

*1: "+1" は通信制御に使用するアドレス / データ選択ビット (AD ビット) です。

*2: 受信時のストップビットは 1 ビットのみを検出します。

■ UART 割り込みおよび EI²OS

表 20.1-3 UART 割り込みおよび EI²OS

割り込み要因	割り込み番号	割り込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ	アドレス	下位	上位	バンク	
UART1 受信割り込み	#37(25 _H)	ICR13	0000BD _H	FFFF68 _H	FFFF69 _H	FFFF6A _H	
UART1 送信割り込み	#38(26 _H)	ICR13	0000BD _H	FFFF64 _H	FFFF65 _H	FFFF66 _H	
UART0 受信割り込み	#39(27 _H)	ICR14	0000BE _H	FFFF60 _H	FFFF61 _H	FFFF62 _H	
UART0 送信割り込み	#40(28 _H)	ICR14	0000BE _H	FFFF5C _H	FFFF5D _H	FFFF5E _H	

:EI²OS は UART 受信エラー検出, EI²OS 停止要求もサポートされています。

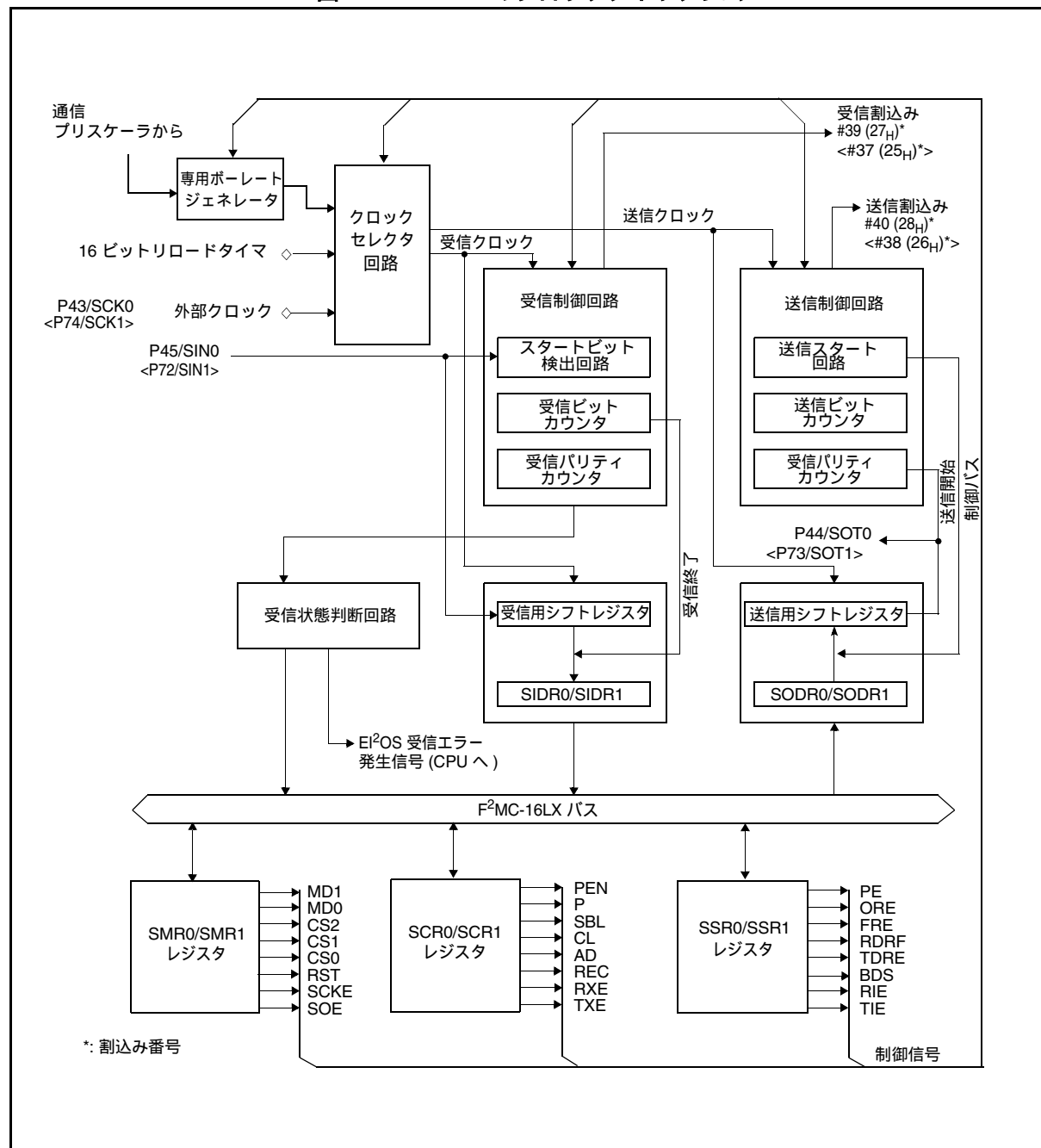
: EI²OS は, 割り込みベクタを共用する ICR13, ICR14 あるいは割り込み要因を使用しない場合, 使用可能です。

20.2 UART のブロックダイアグラム

UART のブロックダイアグラムを以下に示します。

■ UART のブロックダイアグラム

図 20.2-1 UART のブロックダイアグラム



● クロックセレクト

専用ボーレートジェネレータ、外部入力クロック、内部タイマ出力のクロック (16 ビットリロードタイマから供給されるクロック) から選択できます。

● 受信制御回路

受信制御回路は、受信ビットカウンタ、スタートビット検出回路、受信パリティカウンタで構成されます。

受信ビットカウンタは受信データをカウントして、1 フレームのデータ受信が終了した場合に、受信割込み要求を発生します。

スタートビット検出回路は、シリアル入力信号からスタートビットを検出し、受信データをシリアル入力データレジスタへ指定された転送速度に応じて 1 ビットずつシフトしながらシリアル入力データレジスタ (SIDR0/SIDR1) へ書き込みます。受信パリティカウンタは、受信データのパリティビットを検出します。

● 送信制御回路

送信制御回路は、送信ビットカウンタ、送信スタート回路、送信パリティカウンタで構成されます。

送信ビットカウンタは送信データをカウントして、1 フレームのデータ送信が終了した場合に、送信割込み要求を発生します。

送信スタート回路は、シリアル出力データレジスタ (SODR0/SODR1) への書込みにより送信を開始します。送信パリティカウンタは、" パリティあり " の場合にパリティビットを生成します。

● 受信用シフトレジスタ

SIN 端子から入力された受信データを、1 ビットずつシフトしながら取り込み、受信が終了すると、シリアル入力データレジスタ (SIDR0/SIDR1) へ受信データを転送します。

● 送信用シフトレジスタ

シリアル出力データレジスタ (SODR0/SODR1) に書き込まれたデータを送信用シフトレジスタ自身へ転送し、1 ビットずつシフトしながら SOT 端子へ出力します。

● モード制御レジスタ (SMR0/SMR1)

このレジスタは、以下の動作を行います。

- ・ UART 動作モードの選択
- ・ クロック入力ソースの選択
- ・ 専用ボーレートジェネレータのセットアップ
- ・ 専用ボーレートジェネレータを使用したときのクロック速度(クロック分周値)の選択
- ・ シリアルデータ出力の端子出力許可または禁止を設定
- ・ クロックの端子出力の許可または禁止を設定

● 制御レジスタ (SCR0/SCR1)

- このレジスタは、以下の動作を行います。
- パリティの有無の設定
- パリティの種類の選択
- ストップビット長の設定
- データ長の設定
- 動作モード 1 時におけるフレームデータ形式の選択
- エラーフラグのクリア
- 送信の許可または禁止の設定
- 受信の許可または禁止の設定

● シリアルステータスレジスタ (SSR0/SSR1)

送受信の状態やエラーの状態の確認、送受信割込み要求の許可 / 禁止を設定できます。

● シリアル入力データレジスタ (SIDR0/SIDR1)

受信データを保持するレジスタです。シリアル入力データは変換されてこのレジスタに格納されます。

● シリアル出力データレジスタ (SODR0/SODR1)

送信データを設定します。このレジスタに書き込まれたデータは、シリアル変換されて出力されます。

20.3 UART の端子

UART の端子およびブロックダイヤグラムを記載します。

■ UART の端子

UART で使用する端子は、汎用入出力ポートと兼用端子です。

UART 機能を使用する場合の端子機能と必要な設定を、表 20.3-1 に示します。

表 20.3-1 UART の端子

端子名	端子機能	入出力形式	プルアップ	スタンバイ制御	UART の使用に必要な設定
P45/SIN0	ポート 4 入出力 / シリアルデータ入力	CMOS 出力および CMOS ヒステリシス入力	なし	あり	ポート方向レジスタで入力ポートに設定 (DDR4: bit5 = 0)
P44/SOT0	ポート 4 入出力 / シリアルデータ出力				出力許可に設定 (SMR0: SOE=1)
P43/SCK0	ポート 4 入出力 / シリアルクロック入出力				シリアルクロック入力時はポート方向レジスタで入力ポートに設定 (DDR4: bit3 = 0)
					シリアルクロック出力時は出力許可に設定 (SMR0: SCKE=1)
P72/SIN1	ポート 7 入出力 / シリアルデータ入力	CMOS 出力および CMOS ヒステリシス入力	なし	あり	ポート方向レジスタで入力ポートに設定 (DDR7: bit10 = 0)
P73/SOT1	ポート 7 入出力 / シリアルデータ出力				出力許可に設定 (SMR1: SOE=1)
P74/SCK1	ポート 7 入出力 / シリアルクロック入出力				シリアルクロック入力時はポート方向レジスタで入力ポートに設定 (DDR7: bit12 = 0)
					シリアルクロック出力時は出力許可に設定 (SMR1: SCKE=1)

■ UART の端子のブロックダイアグラム

図 20.3-1 UART シリアルデータ入力端子 (P45/SIN0) のブロックダイアグラム

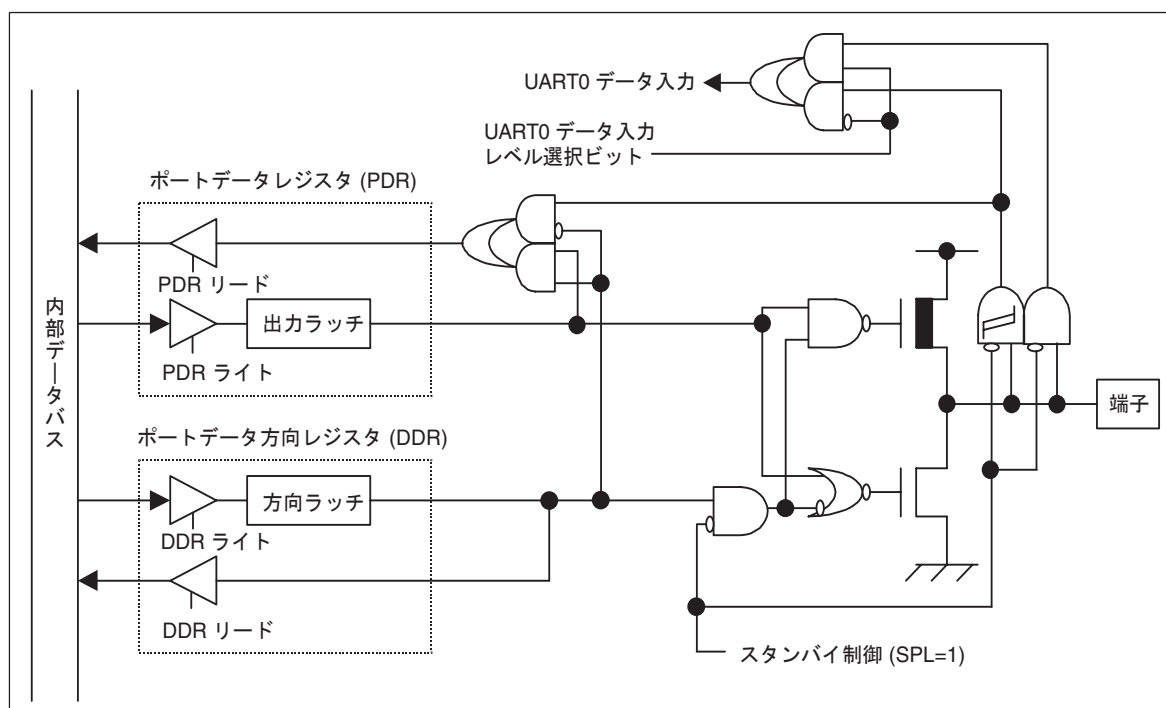
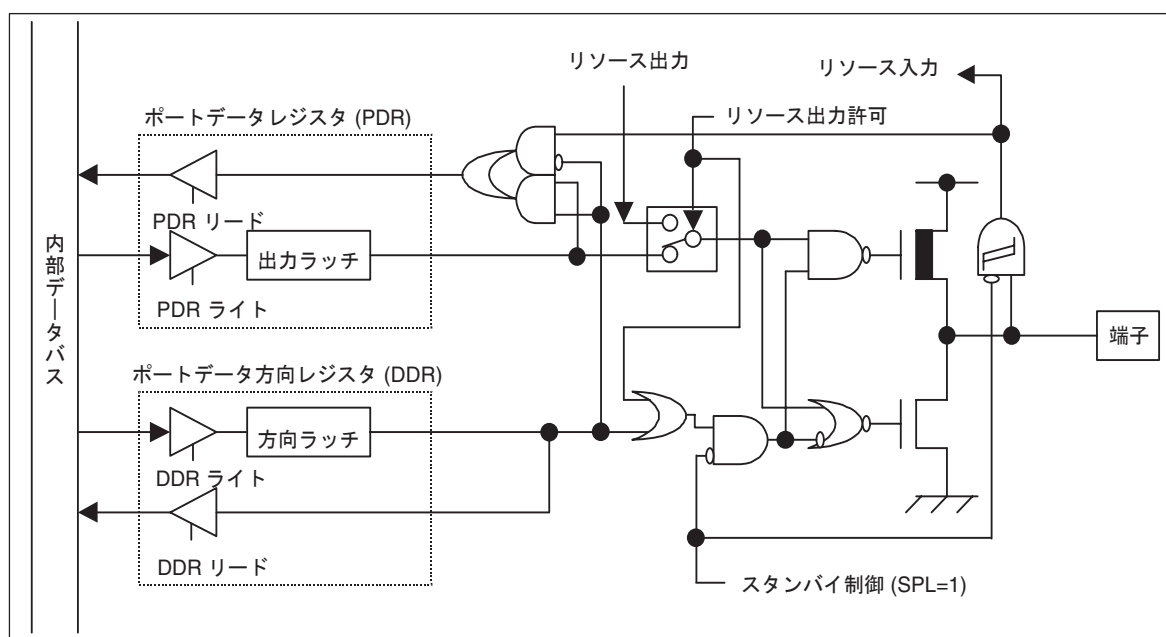


図 20.3-2 UART シリアルデータ入出力端子 (P43/SCK0) およびシリアルデータ出力端子 (P44/SOT0) のブロックダイアグラム



The diagram illustrates the internal data bus architecture. On the left, the **内部データバス** (Internal Data Bus) is shown with bidirectional connections to the **PORT DATA REGISTER (PDR)**, **PORT DATA DIRECTION REGISTER (DDR)**, and **ADDRESS DECODE REGISTER (ADER)**. The **PDR** block includes **PDR リード** (Read), **PDR ライト** (Write), and an **出力ラッチ** (Output Latch). The **DDR** block includes **方向ラッチ** (Direction Latch), **DDR ライト** (Write), and **DDR リード** (Read). External components connected to the bus include **UART1 データ入力** (UART1 Data Input), **UART1 データ入力 レベル選択ビット** (UART1 Data Input Level Select Bit), **A/D コンバータチャネル 選択ビット** (A/D Converter Channel Select Bit), and **A/D コンバータ入力** (A/D Converter Input). The bus also controls a **端子** (Terminal) through a series of logic gates and a **スタンバイ制御 (SPL=1)** (Standby Control) signal. The diagram uses standard logic symbols for AND, OR, NOT, and tri-state buffers to show the control logic for each component.

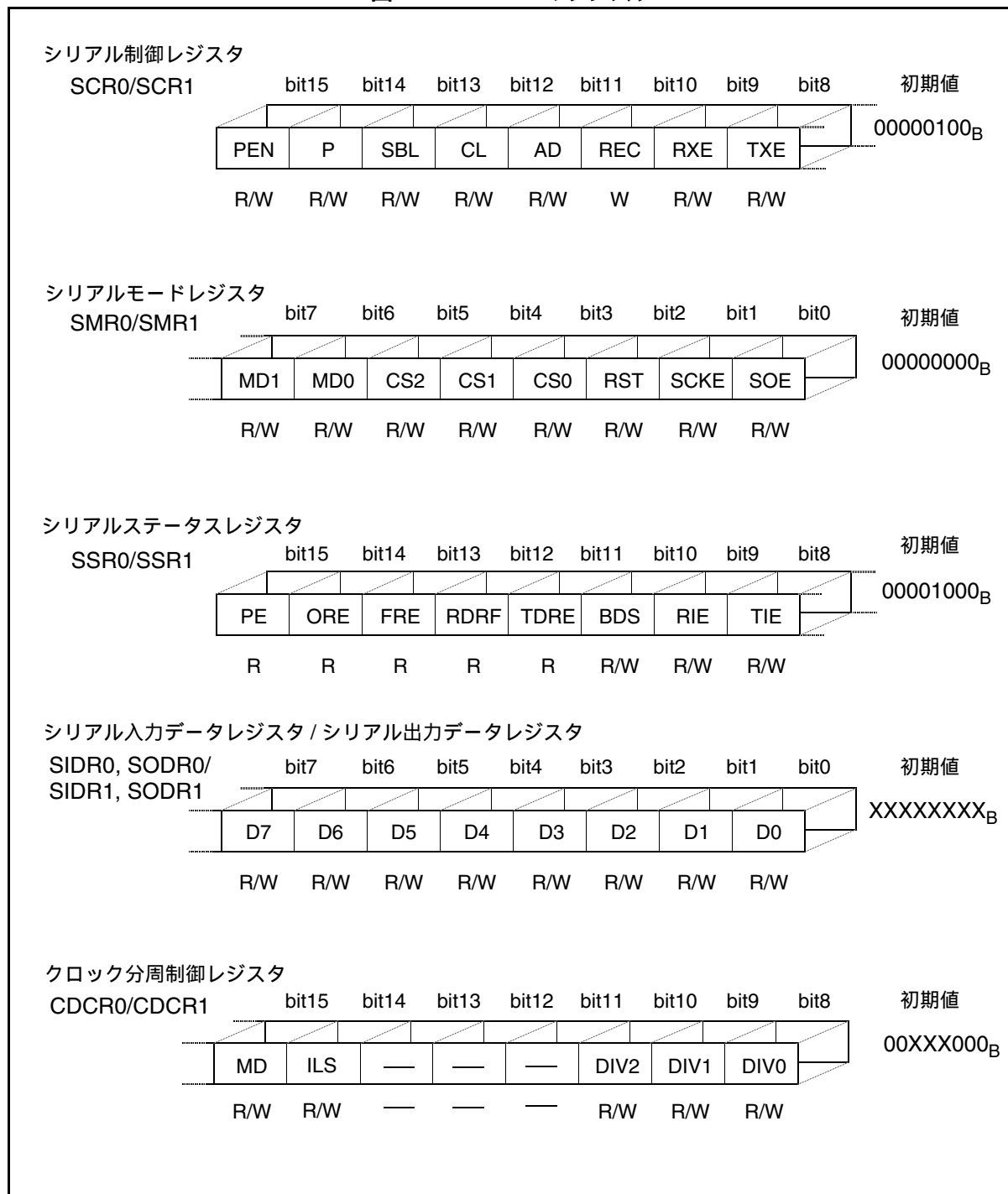
[illegible]

20.4 UART のレジスタ

UART のレジスタを記載します。

■ UART のレジスタ

図 20.4-1 UART のレジスタ



20.4.1 シリアル制御レジスタ (SCR0/SCR1)

シリアル制御レジスタ (SCR0/SCR1) は、パリティビットの設定、ストップビット長やデータ長の選択、動作モード 1 でのフレームデータ形式の選択、受信エラーフラグのクリア、送受信動作の許可 / 禁止を設定できます。

■ シリアル制御レジスタ (SCR0/SCR1)

図 20.4-2 シリアル制御レジスタ (SCR0/SCR1)

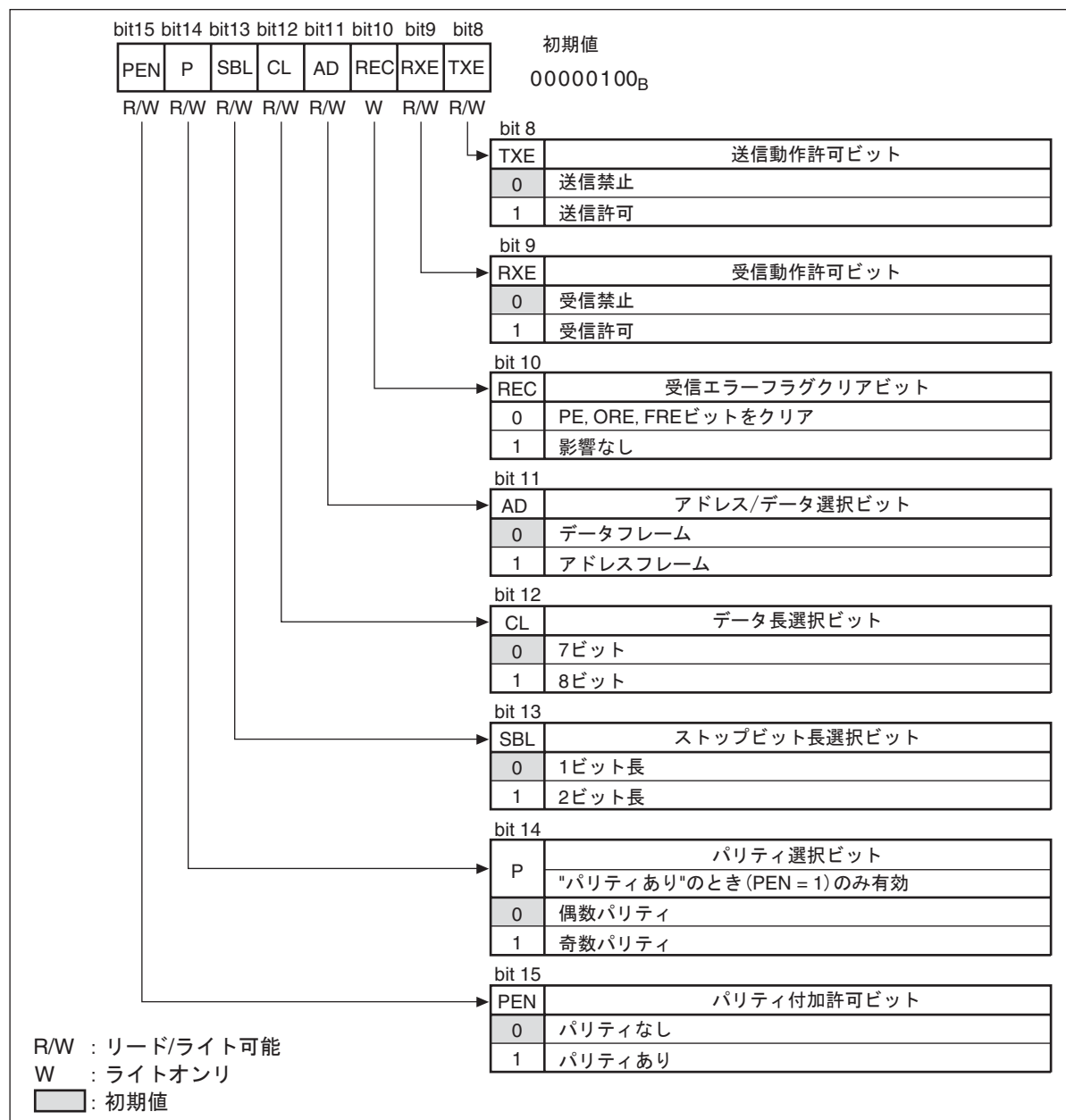


表 20.4-1 シリアル制御レジスタ (SCR0/SCR1) の機能

ビット名		機能
bit15	PEN: パリティ付加許可ビット	パリティビットの付加 (送信時) と検出 (受信時) を行うか行わないかを設定します。 (注意事項) 動作モード 1, 2 ではパリティビットは付加できません。必ず "0" に設定してください。
bit14	P: パリティ選択ビット	パリティビットあり (PEN=1) に設定した場合に、奇数パリティか偶数パリティのいずれかを設定します。
bit13	SBL: ストップビット長選択ビット	非同期転送モードの場合のストップビットまたは送信データのフレームエンドマークのビット長を設定します。 (注意事項) 受信時は、常にストップビットの 1 ビット目のみを検出します。
bit12	CL: データ長選択ビット	送受信データのデータ長を設定します。 (注意事項) 7 ビットを選択できるのは、動作モード 0 だけです (非同期)。動作モード 1 (マルチプロセッサモード) と動作モード 2 (同期) は、必ず 8 ビット長 (CL=1) に設定してください。
bit11	AD: アドレス / データ選択ビット	マルチプロセッサモード (動作モード 1) で、送受信するフレームのデータ形式を設定します。 "0" に設定した場合: データフレームに設定。 "1" に設定した場合: アドレスデータのフレームに設定。
bit10	REC: 受信エラーフラグクリアビット	シリアルステータスレジスタの受信エラーフラグ (SSR0/SSR1: FRE, ORE, PE) を "0" にクリアします。 "0" に設定した場合: FRE, ORE, PE フラグがクリアされます。 "1" に設定した場合: 影響しません。 (注意事項) 受信割込みを許可に設定した場合、FRE, ORE, PE フラグのいずれかが "1" にセットされた場合のみ、この REC ビットを "0" に設定してください。
bit9	RXE: 受信動作許可ビット	UART の受信動作を許可または禁止します。 "0" に設定した場合: 受信動作が禁止されます。 "1" に設定した場合: 受信動作が許可されます。 (注意事項) 受信中に受信動作を禁止に設定した場合は、そのフレームの受信を完了し、受信データをシリアル入力データレジスタ (SIDR0/SIDR1) に格納した後に受信動作を停止します。
bit8	TXE: 送信動作許可ビット	UART の送信動作を許可または禁止します。 "0" に設定した場合: 送信動作が禁止されます。 "1" に設定した場合: 送信動作が許可されます。 (注意事項) 送信中に送信動作を禁止に設定した場合は、シリアル出力データレジスタ (SODR0/SODR1) のデータの送信が完了した後に送信動作が停止します。

20.4.2 シリアルモードレジスタ (SMR0/SMR1)

シリアルモードレジスタ (SMR0/SMR1) は、動作モードの選択、ボーレートクロックの選択、シリアルデータとクロックの端子への出力許可 / 禁止を設定できます。

■ シリアルモードレジスタ (SMR0/SMR1)

図 20.4-3 シリアルモードレジスタ (SMR0/SMR1)

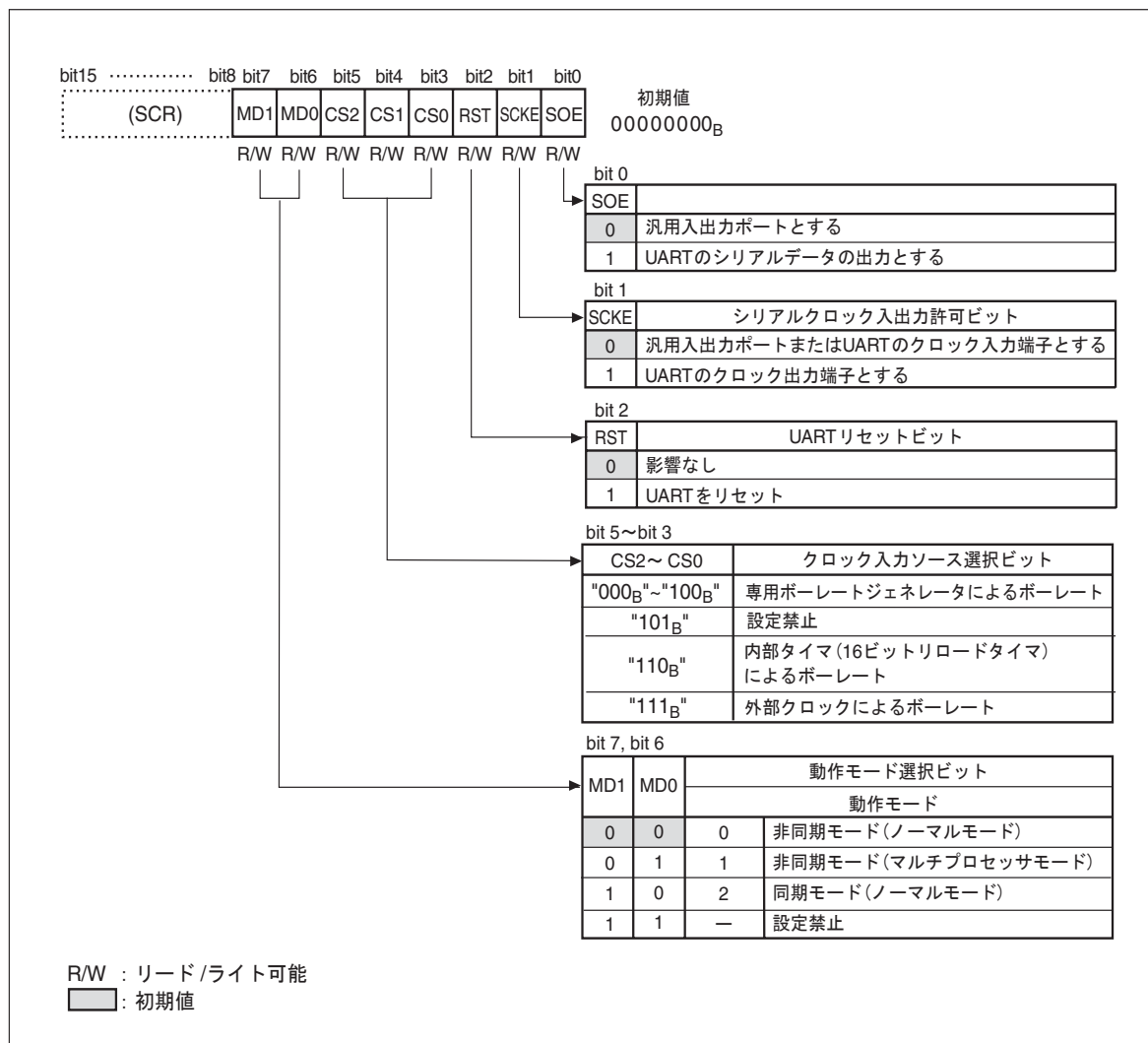


表 20.4-2 シリアルモードレジスタ (SMR0/SMR1) の機能

ビット名		機能
bit7, bit6	MD1, MD0: 動作モード選択ビット	UART の動作モードを設定します。 (注意事項) 動作モード 1(マルチプロセッサモード)では、マスタスレーブ型通信のマスタとしてのみ使用できます。UART は、受信時に 9 ビット目のアドレス / データビットが受信できないためスレーブとしては使用できません。
bit5 ~ bit3	CS2 ~ CS0: クロック入力ソース選択ビット	ボーレートクロックソースを選択します。専用ボーレートジェネレータを選択した場合には、同時にボーレートも決定されます。 <ul style="list-style-type: none"> 専用ボーレートジェネレータを選択する場合は、6 ボーレートから選択し、同期 / 非同期転送モードを設定してください。内部および外部タイマによって生成されたボーレートは、全部で 8 ボーレートが設定できます。 クロック入力ソースは、外部クロック (SCK0/SCK1 端子)、16 ビットリロードタイマ 0、専用ボーレートジェネレータから選択します。 (注意事項) <ul style="list-style-type: none"> 同期転送時に専用ボーレートジェネレータを使用する場合、以下の設定は禁止です 1) CS2 ~ CS0 = 000_B 2) CS2 ~ CS0 = 001_B かつ、DIV2 ~ DIV0 = 000_B
bit2	RST: UART リセットビット	UART および CDCR0/CDCR1, SSR0/SSR1, SCR0/SCR1 レジスタにリセットをかけます。 "0" に設定した場合 : 影響しません。 "1" に設定した場合 : UART および CDCR0/CDCR1, SSR0/SSR1, SCR0/SCR1 レジスタにリセットをかけます。リセット動作後自動的にクリアされます。 リードした場合 : 値は常に "0" です。 (注意事項) "1" を書き込んだ後で UART および CDCR0/CDCR1, SSR0/SSR1, SCR0/SCR1 レジスタの設定を再度初期化する必要があります。
bit1	SCKE: シリアルクロック入出力許可ビット	SCK0/SCK1 端子のシリアルクロック入出力を制御します。 "0" に設定した場合 : P43/SCK0 および P74/SCK1 端子は、汎用入出力ポート (P43, P74) またはシリアルクロック入力端子に設定されます。 "1" に設定した場合 : シリアルクロック出力端子に設定されます。 (注意事項) 1) P43/SCK0 および P74/SCK1 端子をシリアルクロック入力端子として使用する場合は (SCKE=0)、P43 および P74 端子を入力ポートに設定してください。また、クロック選択ビットを外部クロックの設定にしてください (SMR0/SMR1: CS2 ~ CS0=111 _B) 2) シリアルクロック出力端子として使用する場合は (SCKE=1)、クロック選択ビットを外部タイマ以外に設定してください。 (SMR0/SMR1: CS2 ~ CS0=111 _B 以外) (参考) SCK0/SCK1 端子をシリアルクロック出力端子 (SCKE=1) に設定した場合は、汎用入出力ポートの設定にかかわらずシリアルクロック出力端子として機能します。
bit0	SOE: シリアルデータ出力許可ビット	シリアルデータ出力を許可または禁止します。 "0" に設定した場合 : P44/SOT0 および P73/SOT1 端子を汎用入出力ポート (P44, P73) に設定されます。 "1" に設定した場合 : P44/SOT0 および P73/SOT1 端子をシリアルデータ出力端子 (SOT0/SOT1) に設定されます。 (参考) シリアルデータ出力端子に設定した場合は (SOE=1)、汎用入出力ポート (P44, P73) の設定にかかわらず SOT0/SOT1 端子として機能します。

20.4.3 シリアルステータスレジスタ (SSR0/SSR1)

シリアルステータスレジスタ (SSR0/SSR1) は、送受信の状態やエラーの状態の確認、割込みの許可 / 禁止を設定できます。

■ シリアルステータスレジスタ (SSR0/SSR1)

図 20.4-4 シリアルステータスレジスタ (SSR0/SSR1)

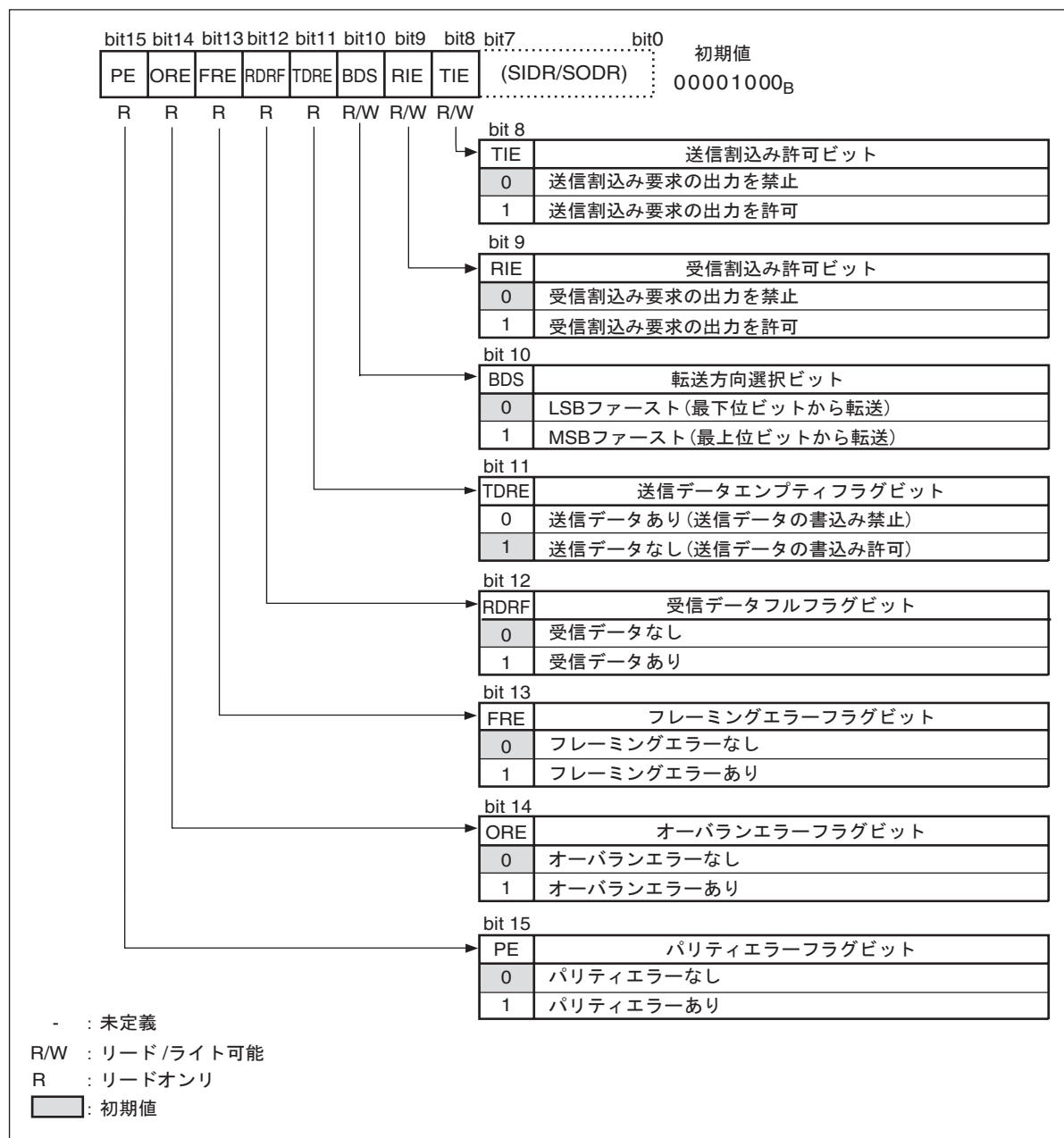


表 20.4-3 シリアルステータスレジスタ (SSR0/SSR1) の機能 (1 / 2)

ビット名		機能
bit15	PE: パリティエラーフラグ ビット	<p>受信時のパリティエラーの有無を示すフラグです。</p> <ul style="list-style-type: none"> 受信時にパリティエラーが発生した場合, "1" にセットされます。受信エラーフラグクリアビットを "0" に設定した場合 (SCR0/SCR1: REC=0), "0" にクリアされます。 受信割込みを許可に設定し (RIE=1), パリティエラーが発生した場合, 受信割込み要求を発生します。 この PE ビットがセットされた場合は, シリアル入力データレジスタ (SIDR0/SIDR1) のデータは無効です。
bit14	ORE: オーバランエラー フラグビット	<p>受信時のオーバランエラーの有無を示すフラグです。</p> <ul style="list-style-type: none"> 受信時にオーバランエラーが発生した場合, "1" にセットされます。受信エラーフラグクリアビットを "0" に設定した場合 (SCR0/SCR1: REC=0), "0" にクリアされます。 受信割込みを許可に設定し (RIE=1), オーバランエラーが発生した場合, 受信割込み要求を発生します。 この ORE ビットがセットされた場合は, シリアル入力データレジスタ (SIDR0/SIDR1) のデータは無効です。
bit13	FRE: フレーミングエラー フラグビット	<p>受信時のフレーミングエラーの有無を示すフラグです。</p> <ul style="list-style-type: none"> 受信時にフレーミングエラーが発生した場合, "1" にセットされます。受信エラーフラグクリアビットを "0" に設定した場合 (SCR0/SCR1: REC=0), "0" にクリアされます。 受信割込みを許可に設定し (SSR0/SSR1: RIE=1), フレーミングエラーが発生した場合, 受信割込み要求を発生します。 この FRE ビットがセットされた場合は, シリアル入力データレジスタ (SIDR0/SIDR1) のデータは無効です。
bit12	RDRF: 受信データフル フラグビット	<p>シリアル入力データレジスタ (SIDR0/SIDR1) の状態を示すフラグです。</p> <ul style="list-style-type: none"> シリアル入力データレジスタ (SIDR0/SIDR1) に受信データがロードされた場合は, "1" にセットされます。 シリアル入力データレジスタ (SIDR0/SIDR1) のデータを読み出した場合は "0" にクリアされます。 受信割込みを許可に設定し (SSR0/SSR1: RIE=1), 受信データがシリアル入力データレジスタにロードされた場合, 受信割込み要求を発生します。
bit11	TDRE: 送信データエンプティ フラグビット	<p>シリアル出力データレジスタ (SODR0/SODR1) の状態を示すフラグです。</p> <ul style="list-style-type: none"> シリアル出力データレジスタ (SODR0/SODR1) に送信データを設定した場合, "0" にクリアされます。 シリアル出力データレジスタ (SODR0/SODR1) のデータが送信用シフトレジスタにロードされ, 送信が開始された場合, "1" にセットされます。 送信割込みを許可に設定し (SSR0/SSR1: TIE=1), シリアル出力データレジスタに設定されたデータが送信された場合, 送信割込み要求を発生します。 <p>(注意事項) 初期化により, この TDRE ビットは "1"(送信データなし) にセットされます。</p>

表 20.4-3 シリアルステータスレジスタ (SSR0/SSR1) の機能 (2 / 2)

ビット名		機能
bit10	BDS: 転送方向選択ビット	シリアルデータの転送方向を設定します。 "0" に設定した場合：最下位ビット側から転送します (LSB ファースト)。 "1" に設定した場合：最上位ビット側から転送します (MSB ファースト)。 (注意事項) シリアルデータレジスタへの読出しおよび書込み時にデータの上位側と下位側を入れ替えるため、シリアル出力データレジスタ (SODR0/SODR1) にデータを書き込んだ後に、転送方向選択ビット (BDS) を書き換えた場合は、書き込まれたデータは無効になります。
bit9	RIE: 受信割込み許可ビット	受信割込みを許可または禁止します。 • 受信割込みを許可に設定し (SSR0/SSR1: RIE=1), 受信データがシリアル入力データレジスタにロードされるか (RDRF=1) または受信エラーが発生した場合 (PE, ORE, FRE のいずれか 1 つ以上のエラーフラグが "1"), 受信割込み要求を発生します。
bit8	TIE: 送信割込み許可ビット	送信割込みを許可または禁止します。 • 送信割込みを許可に設定し (SSR0/SSR1: TIE=1), シリアル出力データレジスタに書き込まれたデータが送信用シフトレジスタにロードされた場合 (TDRE=1), 送信割込み要求を発生します。

20.4.4 シリアル入力データレジスタ (SIDR0/SIDR1), シリアル出力データレジスタ (SODR0/SODR1)

シリアル入力データレジスタ (SIDR0/SIDR1) は, シリアルデータ受信用レジスタで, シリアル出力データレジスタ (SODR0/SODR1) は, シリアルデータ送信用レジスタです。

SIDR0/SIDR1 レジスタと SODR0/SODR1 レジスタは同一アドレスに配置されています。

■ シリアル入力データレジスタ (SIDR0/SIDR1)

シリアル入力データレジスタのビット構成を図 20.4-5 に示します。

図 20.4-5 シリアル入力データレジスタ (SIDR0/SIDR1)

シリアル入力データレジスタ								初期値
bit7	bit6	bit5	bit4	bit3	bit2	bit	bit0	
D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX _B
(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	

シリアル入力データレジスタは, シリアルデータ受信用のデータバッファレジスタです。

- シリアル入力データレジスタには, SIN0/SIN1 端子に入力されたシリアルデータ信号がシフトレジスタで変換されて格納されます。
- シリアル入力データレジスタの上位 1 ビット (D7) は, データ長が 7 ビットに設定されている場合, 無効なデータになります。
- シリアル入力データレジスタに受信データが格納された場合は, 受信データフルフラグが "1" にセットされます (SSR0/SSR1: RDRF=1)。このとき, 受信割込みを許可に設定している場合は (SSR0/SSR1: RIE=1), 受信割込み要求を発生します。
- シリアル入力データレジスタの受信データは, 受信データフルフラグが "1" にセットされている状態で (SSR0/SSR1: RDRF=1) 読み出してください。シリアル入力データレジスタのデータを読み出した場合は, 受信データフルフラグが "0" にクリアされます。
- 受信エラーが発生した場合は (SSR0/SSR1: PE, ORE, FRE のいずれか 1 つ以上のエラーフラグが "1"), シリアル入力データレジスタのデータは無効です。

■ シリアル出力データレジスタ (SODR0/SODR1)

シリアル出力データレジスタのビット構成を図 20.4-6 に示します。

図 20.4-6 シリアル出力データレジスタ (SODR0/SODR1)

シリアル出力データレジスタ								初期値
bit7	bit6	bit5	bit4	bit3	bit2	bit	bit0	
D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX _B
(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	

シリアル出力データレジスタは、シリアルデータ送信用のデータバッファレジスタです。

- シリアル出力データレジスタに送信データを設定した場合、送信動作があらかじめ許可されている場合は、送信データが送信用シフトレジスタに転送され、シリアルデータ出力端子 (SOT0/SOT1 端子) から出力されます。
- シリアル出力データレジスタの上位 1 ビット (D7) は、データ長を 7 ビットに設定している場合、無効なデータになります。
- シリアル出力データレジスタに送信データを設定した場合、送信データエンプティフラグは "0" にクリアされます (SSR0/SSR1: TDRE=0)。またシリアル出力データレジスタから送信用シフトレジスタへの転送が終了した場合は、送信データエンプティフラグは "1" にセットされ (SSR0/SSR1: TDRE=1)、送信データの設定が可能になります。
- 送信割込みを許可に設定し (SSR0/SSR1: TIE=1)、送信割込み要求を発生します。次の送信データの書込みは、送信割込み要求を発生したまたは送信データエンプティフラグ (SSR0/SSR1: TDRE=1) が "1" の状態で行ってください。

< 注意事項 >

シリアル出力データレジスタは書込み専用のレジスタで、シリアル入力データレジスタは読み専用レジスタです。ただし同一アドレスに配置されているため書込み値と読み出し値が異なります。そのため、INC/DEC 命令などリードモディファイライト (RMW) 動作をする命令は使用できません。

20.4.5 通信プリスケアラ制御レジスタ (CDCR)

通信プリスケアラ制御レジスタ (CDCR) は、マシクロックの分周比を設定します。

■ 通信プリスケアラ制御レジスタ (CDCR)

UART の動作クロックは、マシクロックを分周することにより得られます。この通信プリスケアラによって、種々のマシクロックに対して一定のボーレートを得られるように設計されています。通信プリスケアラからの出力は、I/O 拡張シリアルインタフェースの動作クロックに使用されます。

図 20.4-7 通信プリスケアラ制御レジスタ (CDCR)

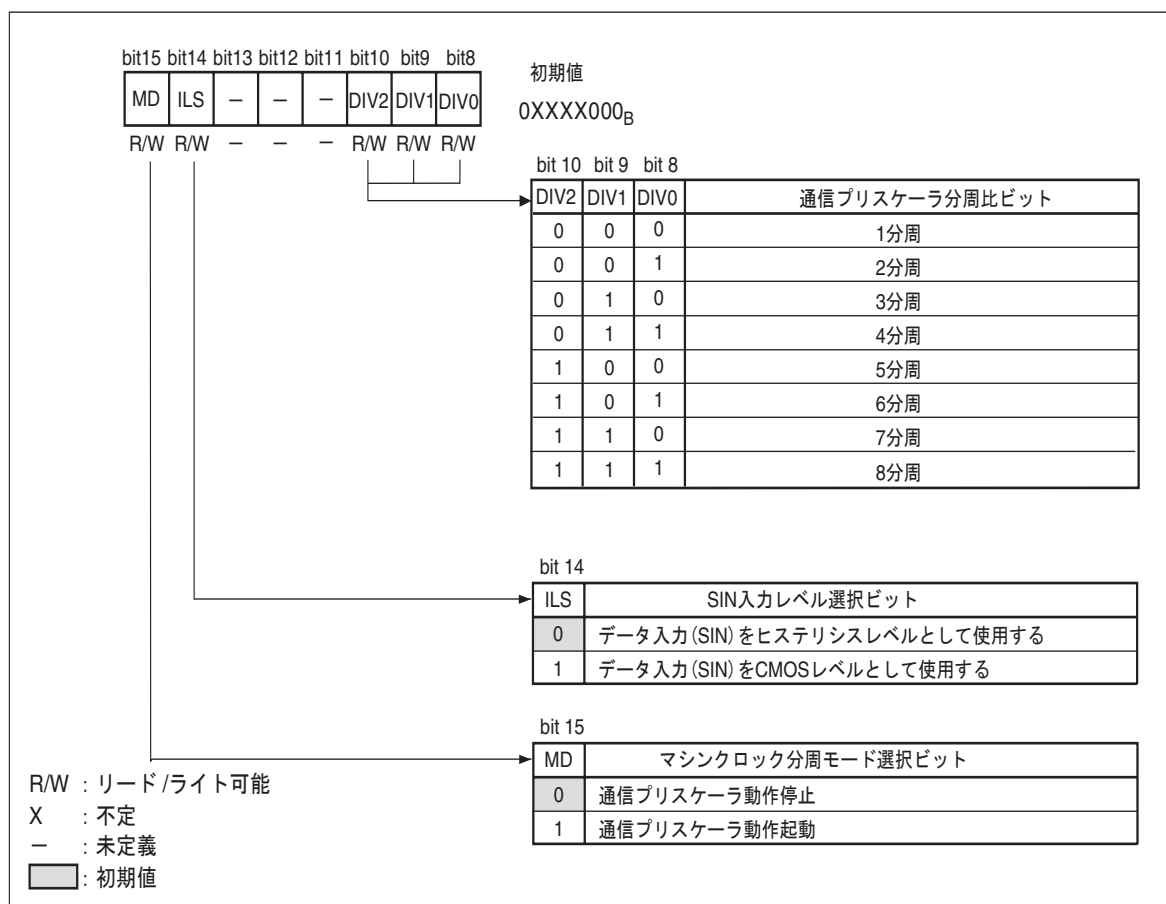


表 20.4-4 通信プリスケアラ制御レジスタ (CDCR)

ビット名		機能
bit15	MD: マシクロック分周モード選択ビット	通信プリスケアラの動作を許可または停止します。 "0" に設定した場合: 停止します。 "1" に設定した場合: 許可します。
bit14	ILS: SIN 入力レベル選択ビット	UART データ入力端子 (SIN) の入力レベル選択ビット。 "0" に設定した場合: ヒステリシス入力レベルを選択します。 "1" に設定した場合: CMOS レベルを選択します。
bit13, bit12	予約ビット	リードした場合: 値は常に "0" です。
bit10 ~ bit8	DIV2, DIV1, DIV0: 2 ~ 0 通信プリスケアラ分周比ビット	マシクロックの分周比を設定するビットです。 MD ビットは "1" にセットされた場合のみ分周比を選択できます。 <ul style="list-style-type: none"> 分周比は変更されると、通信開始までに 2 サイクルを待ちます。 (注意事項) <ul style="list-style-type: none"> 同期転送時に専用ポーレートジェネレータを使用する場合、以下の設定は禁止です。 <ol style="list-style-type: none"> CS2 ~ CS0 = 000_B CS2 ~ CS0 = 001_B かつ、DIV2 ~ DIV0 = 000_B

20.5 UART の割込み

UART には、受信割込みと送信割込みがあり、次の要因で割込み要求を発生できます。

- 受信データがシリアル入力データレジスタ (SIDR0/SIDR1) に設定された場合、または受信エラーが発生した場合。
- 送信データがシリアル出力データレジスタ (SODR0/SODR1) から送信用シフトレジスタに転送された場合。

すべての割込みが、拡張インテリジェント I/O サービス (EI²OS) 機能に対応しています。

■ UART の割込み

UART の割込み制御ビットと割込み要因を表 20.5-1 に示します。

表 20.5-1 UART の割込み制御ビットと割込み要因

送受信	割込み要求フラグ	動作モード			割込み要因	割込み出力許可ビット	割込み要求フラグのクリア
		0	1	2			
受信	RDRF				受信データをシリアル入力データレジスタ (SIDR0/SIDR1) へロード	SSR0/SSR1: RIE	受信データの読出し
	ORE				オーバランエラー発生		受信エラーフラグクリアビット (SSR0/SSR1: REC) への "0" の書込み
	FRE			×	フレーミングエラー発生		
	PE		×	×	パリティエラー発生		
送信	TDRE				シリアル出力データレジスタ (SODR0/SODR1) が空	SSR0/SSR1: TIE	送信データの書込み

: 有
× : 無

● 受信割込み

受信割込みを許可に設定し (SSR0/SSR1: RIE=1), 以下のいずれかを検出した場合、割込み要求を発生します。

- データ受信完了 (SSR0/SSR1: RDRF=1)
- オーバランエラー発生 (SSR0/SSR1: ORE=1)
- フレームエラー発生 (SSR0/SSR1: FRE=1)
- パリティエラー発生 (SSR0/SSR1: PE=1)

受信割込みを許可に設定し (SSR0/SSR1: RIE=1), 以上のいずれかのフラグが "1" にセットされた場合、受信割込み要求を割込みコントローラに出力します。

受信データフルフラグ (SSR0/SSR1: RDRF) は、シリアル入力データレジスタ (SIDR0/SIDR1) を読み出すと、自動的に "0" にクリアされます。各受信エラーフラグ (SSR0/SSR1: PE, ORE, FRE) は、REC ビットに "0" を書き込めば、すべて "0" にクリアされます。

● 送信割込み

送信データがシリアル出力データレジスタから送信用シフトレジスタに転送された場合、送信データエンプティフラグが "1" にセットされます (SSR0/SSR1: TDRE=1)。

- 送信割込みを許可に設定した (SSR0/SSR1: TIE=1) 場合、送信割込み要求を割込みコントローラに出力します。

■ UART の割込みと EI²OS表 20.5-2 UART の割込みと EI²OS

割込み要因	割込み番号	割込み制御レジスタ		ベクタテーブルアドレス			EI ² OS
		レジスタ名	アドレス	下位	中位	上位	
UART1 受信割込み	#37(25 _H)	ICR13	0000BD _H	FFFF68 _H	FFFF69 _H	FFFF6A _H	
UART1 送信割込み	#38(26 _H)	ICR13	0000BD _H	FFFF64 _H	FFFF65 _H	FFFF66 _H	
UART0 受信割込み	#39(27 _H)	ICR14	0000BE _H	FFFF60 _H	FFFF61 _H	FFFF62 _H	
UART0 送信割込み	#40(28 _H)	ICR14	0000BE _H	FFFF5C _H	FFFF5D _H	FFFF5E _H	

: UART の受信エラー検出による EI²OS 停止機能付き

: ICR13, ICR14 または割込みベクタを共有する割込み要因を使用しないとき使用可能

■ UART の EI²OS 機能

UART は EI²OS 機能に対応しており、受信/送信の各割込み要因で EI²OS を起動できます。

● 受信時

ほかのリソースの状態に関係なく、EI²OS を使用できます。

● 送信時

割込み制御レジスタ(ICR13, ICR14)を、UARTの受信割込みと共有しているので、UARTの受信で割込みを使用しない場合に、EI²OS を起動できます。

20.5.1 受信割込み要求の発生とフラグセットのタイミング

受信動作中に発生する割込みには、受信完了 (SSR0/SSR1: RDRF) と受信エラーの発生 (SSR0/SSR1: PE, ORE, FRE) があります。

■ 受信割込み要求の発生とフラグセットのタイミング

● 受信データロードフラグと各受信エラーフラグのセット

受信データは、動作モード 0, 1 ではストップビットが検出された場合に、動作モード 2 では最終ビット (D7) が検出された場合に、シリアル入力データレジスタ (SIDR0/SIDR1) へ格納されます。

受信データを格納した時点で受信エラーが発生していた場合は、受信エラーの内容により対応するエラーフラグ (SSR0/SSR1: PE, ORE, FRE) が "1" にセットされ、その後、受信データフルフラグが "1" にセットされます (SSR0/SSR1: RDRF=1)。

どの動作モードの場合でも、いずれかのエラーフラグが "1" にセットされている場合は、シリアル入力データレジスタの受信データは無効です。

動作モード 0(非同期ノーマルモード)

- ストップビットを検出した場合、受信データフルフラグが "1" にセットされます (SSR0/SSR1: RDRF=1)。
- 受信エラーが発生した場合、エラーフラグが "1" にセットされます (SSR0/SSR1: PE, ORE, FRE)。

動作モード 1(非同期マルチプロセッサモード)

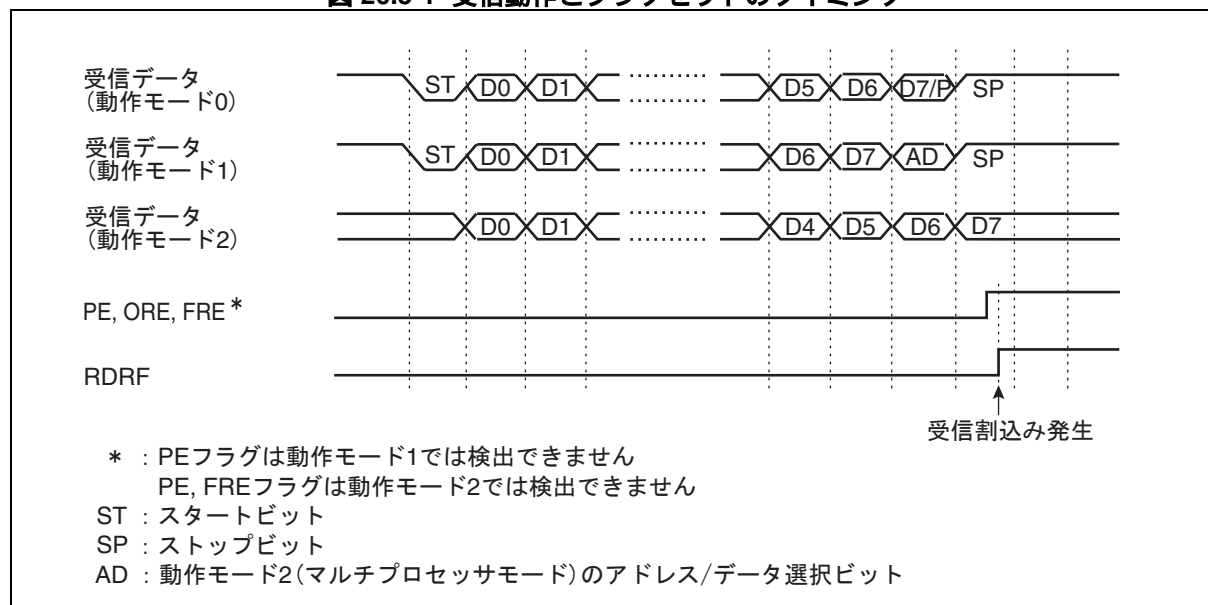
- ストップビットを検出した場合、受信データフルフラグが "1" にセットされます (SSR0/SSR1: RDRF=1)。
- 受信エラーが発生した場合、エラーフラグが "1" にセットされます (SSR0/SSR1: ORE, FRE)。パリティエラーは検出されません。

動作モード 2(同期ノーマルモード)

- 受信データの最終ビットを検出した場合、受信データフルフラグが "1" にセットされます (SSR0/SSR1: RDRF=1)。
- 受信エラーが発生した場合、エラーフラグがセットされます (SSR0/SSR1: ORE) パリティエラーとフレーミングエラーは検出されません。

受信動作とフラグセットのタイミングを、図 20.5-1 に示します。

図 20.5-1 受信動作とフラグセットのタイミング



● 受信割り込み要求発生のタイミング

受信割り込み許可ビットを "1" に設定し (SSR0/SSR1: RIE=1), シリアルステータスレジスタの受信データフルフラグかエラーフラグのいずれかのビットが "1" にセットされた場合 (SSR0/SSR1: RDRF, PE, ORE, FRE), 受信割り込み要求 (#37, #39) を発生します。

20.5.2 送信割込み要求の発生とフラグセットのタイミング

送信動作中の割込み要求は、シリアル出力データレジスタ (SODR0/SODR1) に、次の送信データの書込みが可能な状態になった場合に発生します。

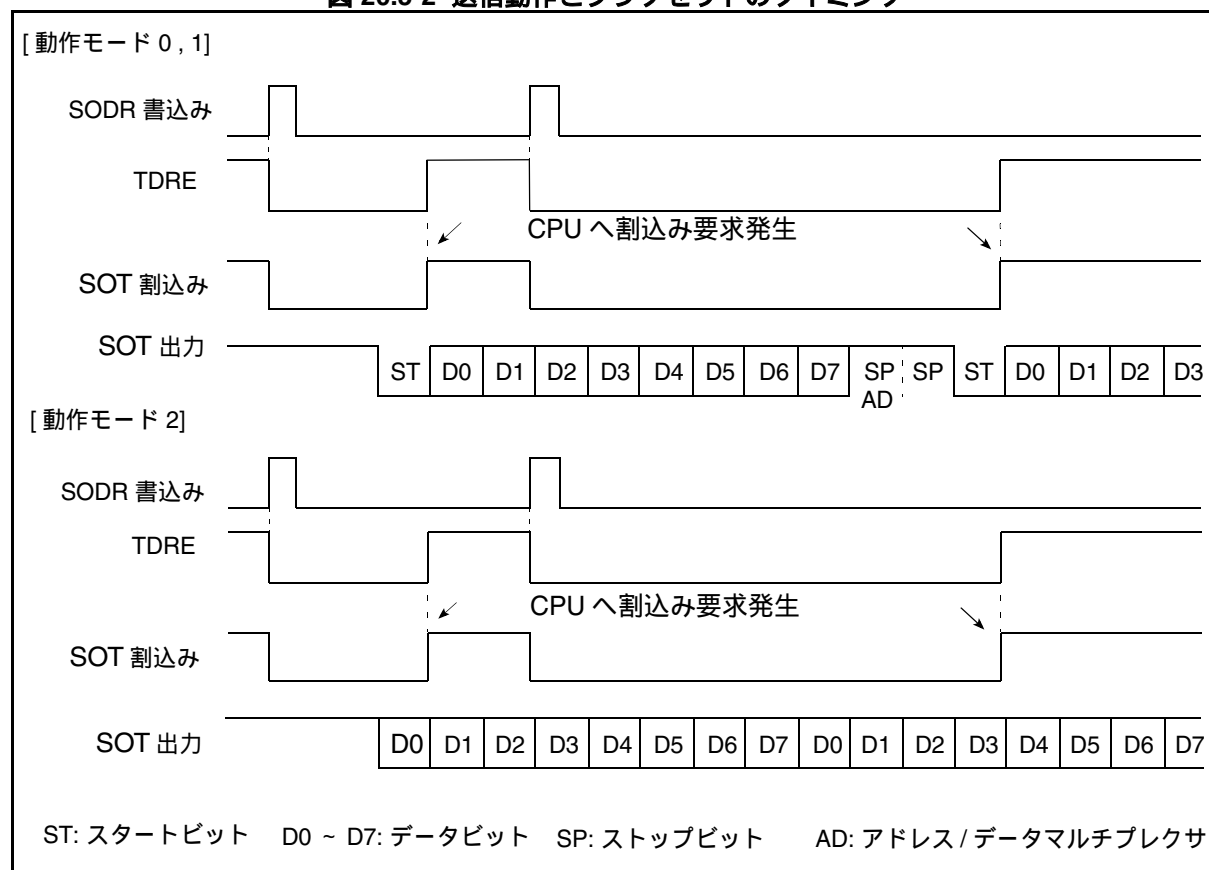
■ 送信割込み要求の発生とフラグセットのタイミング

● 送信データエンプティフラグのセットとクリア

送信データエンプティフラグは、シリアル出力データレジスタの送信データが送信用シフトレジスタに転送され、次の送信データの書込みが可能な状態になった場合に、"1" にセットされます (SSR0/SSR1: TDRE=1)。送信データエンプティフラグは、シリアル出力データレジスタに送信データを書き込んだ場合、"0" にクリアされます。

送信動作とフラグセットのタイミングを、図 20.5-2 に示します。

図 20.5-2 送信動作とフラグセットのタイミング



● 送信割込み要求発生のタイミング

送信割込み許可ビットを "1" に設定し (SSR0/SSR1: TIE=1), 送信データエンプティフラグが "1" にセットされた場合、送信割込み要求 (#38, #40) を発生します。

< 注意事項 >

初期状態で TDRE ビットが "1" のため、送信割込みを許可 (TIE = 1) すると直ちに送信完了割込みが発生します。TDRE ビットはリードオンリビットで、シリアル出力データレジスタ (SODR0/SODR1) に新規データを書き込むしかクリアする方法がありません。送信割込み許可タイミングの設定に注意してください。

20.6 UART のボーレート

UART の送受信クロックは、次のいずれかを選択します。

- 専用ボーレートジェネレータ
 - 内部クロック (16 ビットリロードタイマ)
 - 外部クロック (SCK 端子入力のクロック)
-

■ UART のボーレートの選択

ボーレートの選択回路は図 20.6-1 に示すように、ボーレートは次の 3 種類の中から 1 種類を選択できます。

● 専用ボーレートジェネレータによるボーレート

UART に内蔵されている専用ボーレートジェネレータがあります。モード制御レジスタ (SMR0/SMR1) で、8 種類のボーレートから 1 種類を選択できます。

マシンのクロックの周波数とモード制御レジスタ (SMR0/SMR1) の CS2 ~ CS0 ビットによって、非同期または同期ボーレートを選択します。

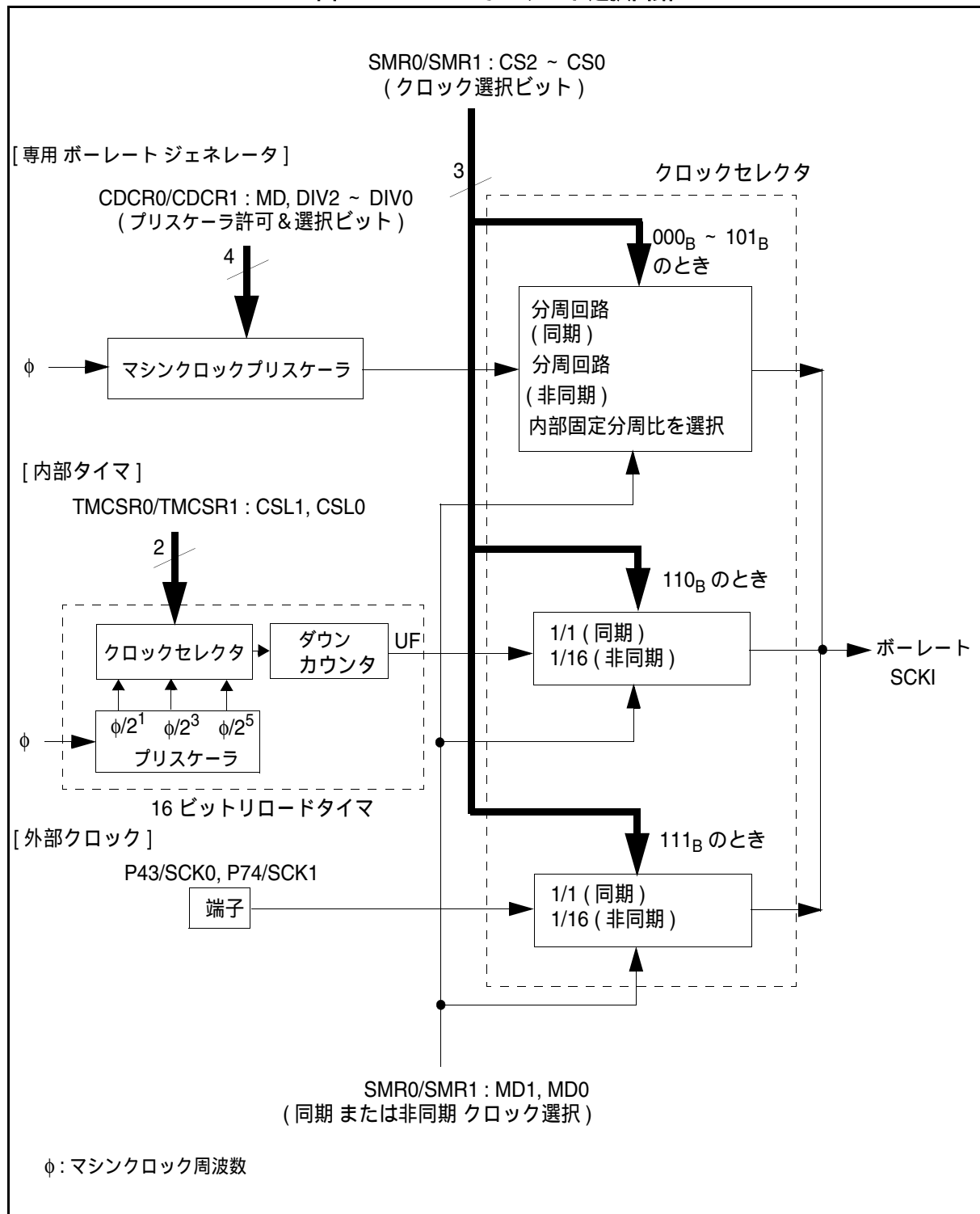
● 内部タイマによるボーレート

16 ビットリロードタイマから供給される内部クロックを、そのまま (同期) もしくは 16 分周 (非同期) して、ボーレートとして使用します。リロード値の設定で任意のボーレートを設定できます。

● 外部クロックによるボーレート

UART のクロック入力端子 (P43/SCK0, P74/SCK1) から入力されたクロックを、そのまま (同期) もしくは 16 分周 (非同期) して、ボーレートとして使用します。外部で任意のボーレートを設定できます。

図 20.6-1 UART ボーレート選択回路



20.6.1 専用ボーレートジェネレータによるボーレート

UART の送受信クロックとして、専用ボーレートジェネレータの出力クロックを選択した場合の設定について説明します。

■ 専用ボーレートジェネレータによるボーレート

専用ボーレートジェネレータで送受信クロックを生成する場合は、マシニングクロックがマシニングクロックプリスケアラで分周された後、クロックセクタによって選択された転送クロック分周比で再び分周されます。

マシニングクロック分周比は、非同期 / 同期共通ですが、転送クロック分周比は、非同期と同期で、別々に内部設定された値を選択するようになっています。

したがって実際の転送レートは、次式で与えられます。

$$\text{非同期ボーレート} = \phi \times (\text{プリスケアラ分周比}) \times (\text{非同期転送クロック分周比})$$

$$\text{同期転ボーレート} = \phi \times (\text{プリスケアラ分周比}) \times (\text{同期転送クロック分周比})$$

ϕ : マシニングクロック周波数

● 通信プリスケアラによる分周比 (非同期, 同期に共通)

マシニングクロックの分周比は、通信プリスケアラレジスタの分周比選択ビットで (CDCR: DIV2 ~ DIV0), 表 20.6-1 に示すように設定できます。

表 20.6-1 通信プリスケアラの出力周波数

MD	DIV2	DIV1	DIV0	分周比
0	-	-	-	停止
1	0	0	0	1
1	0	0	1	2
1	0	1	0	3
1	0	1	1	4
1	1	0	0	5
1	1	0	1	6
1	1	1	0	7
1	1	1	1	8

● 同期ボーレート分周比

分周比はシリアルモードレジスタ (SMR0/SMR1) の CS2 ~ CS0 ビットで、表 20.6-2 に示すように設定できます。

表 20.6-2 周期ボーレート分周比の選択

CS2	CS1	CS0	CLK 同期	算出式
0	0	0	2MHz	$(\phi \div \text{div}) / 1$
0	0	1	1MHz	$(\phi \div \text{div}) / 2$
0	1	0	500kHz	$(\phi \div \text{div}) / 4$
0	1	1	250kHz	$(\phi \div \text{div}) / 8$
1	0	0	125kHz	$(\phi \div \text{div}) / 16$
1	0	1	62.5kHz	$(\phi \div \text{div}) / 32$

ϕ (マシニングロック) : 16MHz

div(マシニングロック分周比) = 8

最大ボーレート : 1/8 マシニングロックと仮定した場合。

● 非同期ボーレート分周比

分周比はシリアルモードレジスタ (SMR0/SMR1) の CS2 ~ CS0 ビットで、表 20.6-3 に示すように設定できます。

表 20.6-3 非周期ボーレート分周比の選択

CS2	CS1	CS0	非同期 (調歩同期)	算出式
0	0	0	76923Hz	$(\phi \div \text{div}) / (8 \times 13 \times 2)$
0	0	1	38461Hz	$(\phi \div \text{div}) / (8 \times 13 \times 4)$
0	1	0	19230Hz	$(\phi \div \text{div}) / (8 \times 13 \times 8)$
0	1	1	9615Hz	$(\phi \div \text{div}) / (8 \times 13 \times 16)$
1	0	0	500kHz	$(\phi \div \text{div}) / (8 \times 2 \times 2)$
1	0	1	250kHz	$(\phi \div \text{div}) / (8 \times 2 \times 4)$

ϕ (マシニングロック) : 16MHz

div(マシニングロック分周比) = 1 と仮定した場合。

● 内部タイマ

CS2 ~ CS0 を "110_B" に設定して内部タイマを選択した場合のボーレート計算 (リロードタイマ使用)

非同期 (調歩同期): $(\phi \div N)/(16 \times 2 \times (n + 1))$

CLK 同期 : $(\phi \div N)/(2 \times (n + 1))$

N: 16 ビットリロードタイマのプリスケアラによる分周比

n: 16 ビットリロードタイマのリロード値

< 注意事項 >

モード 2(CLK 同期モード) では, SCK0/SCK1 は SCKI より最大 3 クロック遅れます。論理上達成可能転送速度は, システムクロック周波数の 1/3 です。ただし, 実仕様にはシステムクロック周波率の 1/4 をお奨めします。

● 外部クロック

CS2 ~ CS0 を "111_B" に設定して外部タイマを選択した場合, 外部クロック周波数を f と指定すれば, 以下のボーレートを仮定します。

非同期 (調歩同期): $f/16$

CLK 同期: f

ただし, 最大外部クロック周波数 f は, 2MHz です。

20.6.2 内部タイマ (16 ビットリロードタイマ) によるボーレート

UART の送受信クロックとして、16 ビットリロードタイマから供給される内部クロックを選択する場合の設定について説明します。

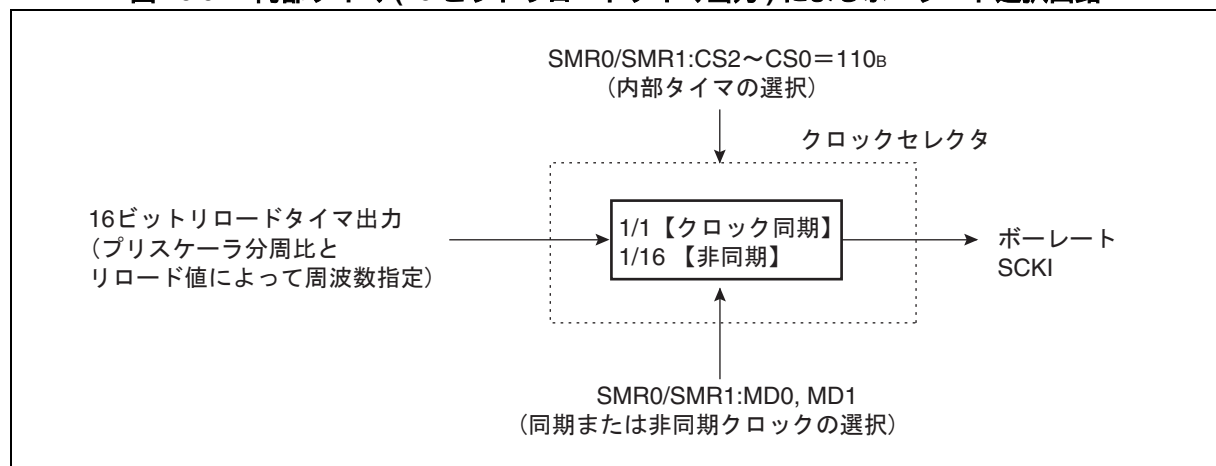
■ 内部タイマ (16 ビットリロードタイマ) によるボーレート

内部タイマによるボーレートを選択する場合は、シリアルモードレジスタのクロック入力ソース選択ビット (SMR0/SMR1: CS2 ~ CS0) を "110_B" に設定します。

ボーレートは、16 ビットリロードタイマでプリスケアラによる分周比とリロード値により、自由に設定できます。

内部タイマによるボーレートの選択回路を図 20.6-2 に示します。

図 20.6-2 内部タイマ (16 ビットリロードタイマ出力) によるボーレート選択回路



● ボーレート計算式

$$\text{非同期ボーレート} = \frac{\phi}{X(n+1) \times 2 \times 16} \quad \text{bps}$$

$$\text{同期ボーレート} = \frac{\phi}{X(n+1) \times 2} \quad \text{bps}$$

ϕ : マシンクロック周波数

X : 16 ビットリロードタイマのプリスケアラによる分周比 ($2^1, 2^3, 2^5$)

n : 16 ビットリロードタイマのリロード値 (0 ~ 65535)

- ボーレートとリロードレジスタ設定値の設定例 (マシンクロック : 7.3728MHz の場合)

表 20.6-4 ボーレートとリロードレジスタ設定値

ボーレート (bps)	リロードレジスタ設定値			
	非同期 (調歩同期)		クロック同期	
	X=2 ¹ (マシン クロックの 2 分周)	X=2 ³ (マシン クロックの 8 分周)	X=2 ¹ (マシン クロックの 2 分周)	X=2 ³ (マシン クロックの 8 分周)
38400	2	-	47	11
19200	5	-	95	23
9600	11	2	191	47
4800	23	5	383	95
2400	47	11	767	191
1200	95	23	1535	383
600	191	47	9071	767
300	383	95	6143	1535

X : 16 ビットリロードタイマのプリスケラによる分周比
- : 設定不可

< 注意事項 >

クロック同期時 , 以下の設定は禁止です。
N = 1, n = 0

20.6.3 外部クロックによるボーレート

UART の送受信クロックとして、外部クロックを選択した場合の設定について説明します。

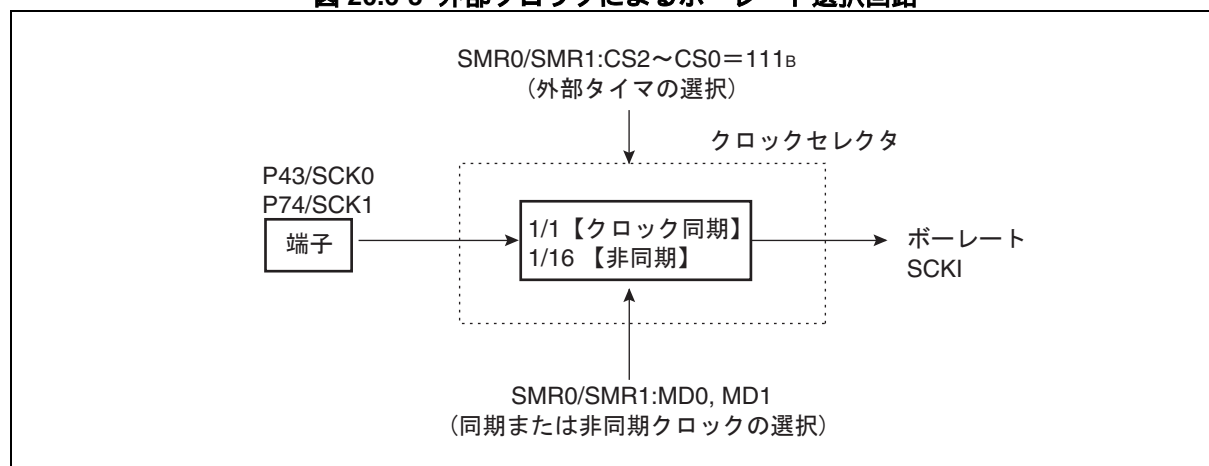
■ 外部クロックによるボーレート

外部クロック入力によるボーレートを選択する場合は、以下のように設定してください。

- シリアルモードレジスタのクロック入力ソース選択ビットを "111_B" に設定します。
(SMR0/SMR1: CS2 ~ CS0=111_B)
- P43/SCK0およびP74/SCK1端子をポート方向レジスタ(DDR4: bit3 = 0, DDR7 : bit12 = 0) で入力ポートに設定します
- 端子を外部クロック入力端子にするため、シリアルクロック入出力切換えビットを "0" に設定します (SMR0/SMR1: SCKE=0)。

図 20.6-3 に示すように SCK1 端子から入力された外部クロックをもとに、ボーレートを設定します。内部の分周比は固定されています。ボーレートを変更する場合は外部の入力クロック周期を変更してください。

図 20.6-3 外部クロックによるボーレート選択回路



● ボーレート計算式

非同期ボーレート = $f/16$

クロック同期ボーレート = f

f: 外部クロック周波数 (f は最大で 2MHz)

20.7 UART の動作説明

UART 機能には、双方向シリアル通信機能 (動作モード 0, 2) と、マスタ・スレーブ型接続の通信機能 (動作モード 1) があります。

■ UART の動作

● 動作モード

UART の動作モードにはモード 0 ~ 2 の 3 種類があります。表 20.7-1 に示すように接続方式やデータ通信方式に適合したものを選択できます。

表 20.7-1 UART の動作モード

動作モード		データ長		同期モード	ストップビット
		パリティなし	パリティあり		
0	ノーマルモード	7 ビットまたは 8 ビット		非同期	1 ビットまたは 2 ビット *2
1	マルチプロセッサモード	8+1 ビット *1	-	非同期	
2	ノーマルモード	8 ビット	-	同期	なし

- : 設定不可

*1: "+" は通信制御に使用するアドレス / データ選択ビット (AD) です。

*2: 受信時のストップビットは 1 ビットのみを検出します。

< 注意事項 >

UART の動作モード 1 は、マスタ / スレーブ型接続のマスタ側のみ使用できます。

● CPU 間接続方式

CPU 間接続方式としては、1 対 1 接続 (ノーマルモード) とマスタ / スレーブ型接続 (マルチプロセッサモード) のどちらかを選択できます。いずれの方式でも、データ長、パリティ有無、同期方式は、すべての CPU 間で統一しておく必要があります。動作モードは以下のように選択します。

- 1 対 1 接続では、2 つの CPU 間で動作モード 0, 2 のいずれか同じ方式を採用する必要があります。非同期方式の場合、動作モード 0 を、クロック同期方式の場合、動作モード 2 を設定してください。
- マスタ / スレーブ型接続では、動作モード 1 を選択し、マスタとして使用できます。なお、この接続時は、パリティなしを設定してください。

● 同期方式

非同期方式 (調歩同期) またはクロック同期方式に設定してください。

● 信号方式

UART は、NRZ(Non Return to Zero) 形式のデータだけを扱えます。

● 動作許可ビット

UART は、TXE(送信), RXE(受信) の動作許可ビットによって、送受信動作を制御できます。

どちらかの動作が禁止された場合、次のように停止してください。

- 受信時(受信用シフトレジスタにデータが入力されているとき)に受信動作を禁止したときには、そのフレームの受信を完了し、シリアル入力データレジスタ (SIDR0/SIDR1) に受信データを格納したところで受信動作を停止します。
- 送信時(送信用シフトレジスタからデータが出力されているとき)に送信動作を禁止したときは、シリアル出力データレジスタ (SODR0/SODR1) にデータがなくなった後に送信動作を停止します。

20.7.1 非同期モード (動作モード 0, 1) での動作

UART を動作モード 0(ノーマルモード) または動作モード 1(マルチプロセッサモード) で使用する場合は、非同期方式で通信を行うことができます。

■ 非同期モードでの動作

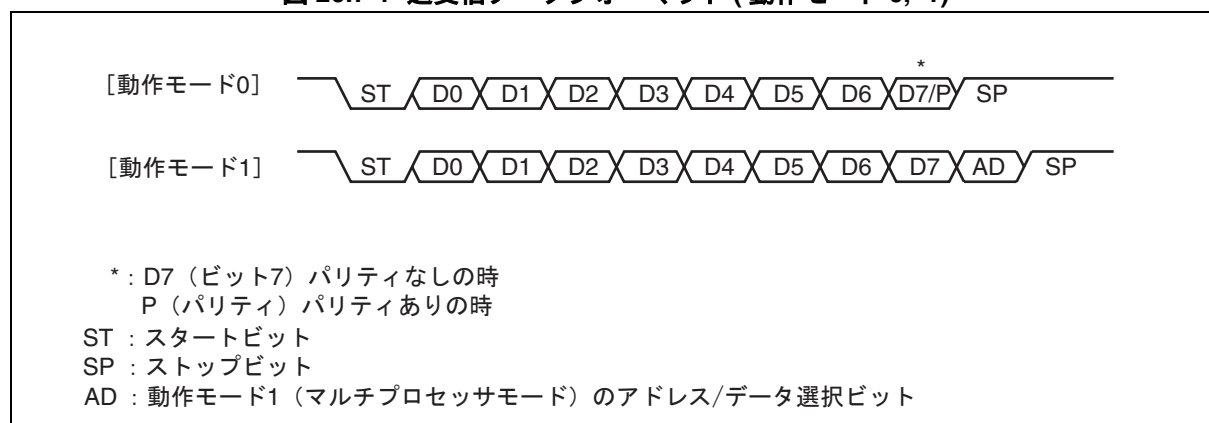
● 送受信データフォーマット

送受信は、必ずスタートビット ("L" レベル) から始まり、指定されたデータビット長の送受信は LSB ファーストで行われ、ストップビット ("H" レベル) で終了します。

- 動作モード 0 のノーマルモードでは、データ長は 7 ビットまたは 8 ビットに設定可能です。
- 動作モード 1 時は、データ長は 8 ビットに固定されます。パリティビットは付加されません。そのかわりアドレス / データ選択ビット (AD) が付加されます。

非同期モードの送受信データフォーマットを図 20.7-1 に示します。

図 20.7-1 送受信データフォーマット (動作モード 0, 1)



● 送信動作

- 送信データは、送信データエンプティフラグが"1"にセットされている場合に(SSR0/SSR1:TDRE=1)、シリアル出力データレジスタに書き込むことができます。シリアル制御レジスタの送信動作許可ビットを "1" に設定した場合 (SCR0/SCR1:TXE=1)、送信動作を開始します。
- 送信データが送信用シフトレジスタに転送され、送信が開始されると TDRE フラグが再び "1" にセットされ、次の送信データをセットできるようになります。このとき送信割込み要求が許可 (SSR0/SSR1:TIE=1) されていれば、送信割込み要求を出力して SODR0/SODR1 に送信データをセットするように要求します。TDRE フラグは、SODR0/SODR1 に送信データを書き込むと "0" にクリアされます。

● 受信動作

- 受信動作が許可 (SCR0/SCR1:RXE=1) されていると、常に受信動作が行われています。スタートビットを検出すると、制御レジスタ (SCR0/SCR1) で決められたデータフォーマットに従って 1 フレームデータの受信が行われます。1 フレームの受信が終わると、エラーが発生した時にはエラーフラグがセットが行われた後、受信データフルフラグビット (SSR0/SSR1:RDRF) が "1" にセットされます。このとき受信割込み要求が許可 (SSR0/SSR1:TIE=1) されていれば受信割込み要求を出力します。
- 入力データレジスタ (SIDR0/SIDR1) の各フラグの状態を確認し、正常受信なら入力データレジスタ (SIDR0/SIDR1) をリードして、エラーが発生していればエラー処理を行うようにしてください。RDRF フラグは、SIDR0/SIDR1 から受信データをリードすると "0" にクリアされます。

● ストップビット

1 ビットまたは 2 ビットを選択できます。受信側では、常に最初の 1 ビットだけを検出します。

● エラー検出

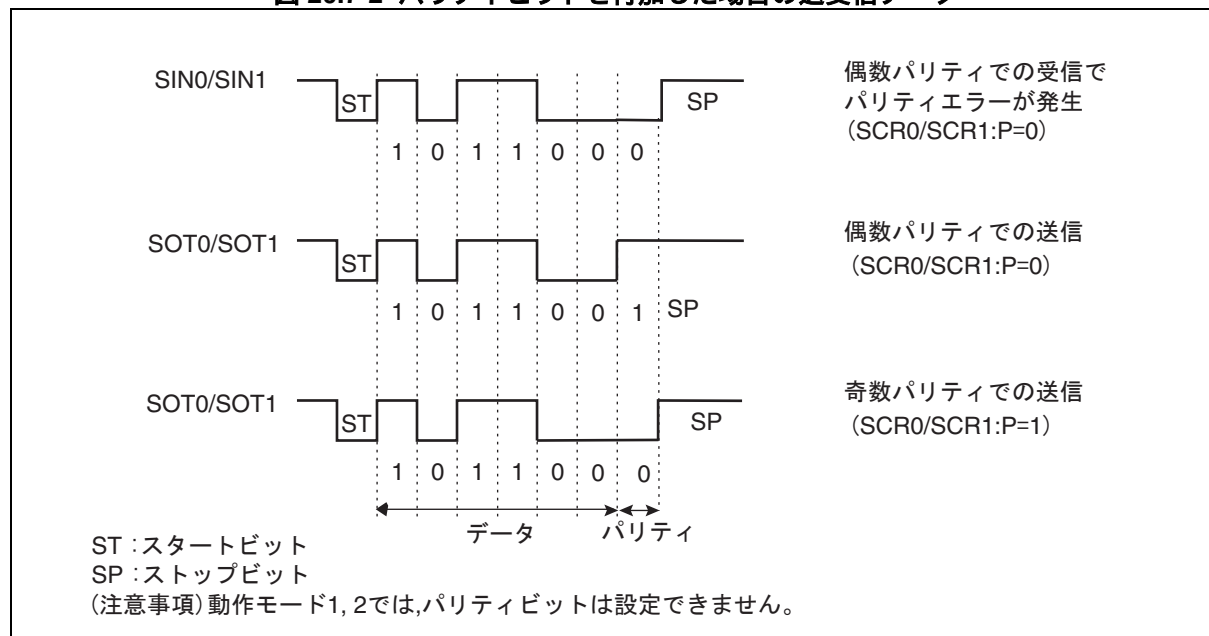
- 動作モード 0 では、パリティエラー、オーバランエラー、フレームエラーが検出できません。
- 動作モード 1 では、オーバランエラーとフレームエラーが検出できます。

● パリティビット

パリティは、動作モード 0(非同期, ノーマルモード) の場合のみ付加できます。制御レジスタ (SCR0/SCR1) の PEN ビットでパリティの有無を、P ビットで偶数/奇数パリティを設定できます。動作モード 1(非同期, マルチプロセッサモード), および動作モード 2(同期, ノーマルモード) では、パリティビットは付加できません。

パリティビットを付加した場合の送受信データを図 20.7-2 に示します。

図 20.7-2 パリティビットを付加した場合の送受信データ



20.7.2 クロック同期モード (動作モード 2) での動作

UART を動作モード 2 で使用する場合は、クロック同期方式で通信を行います。

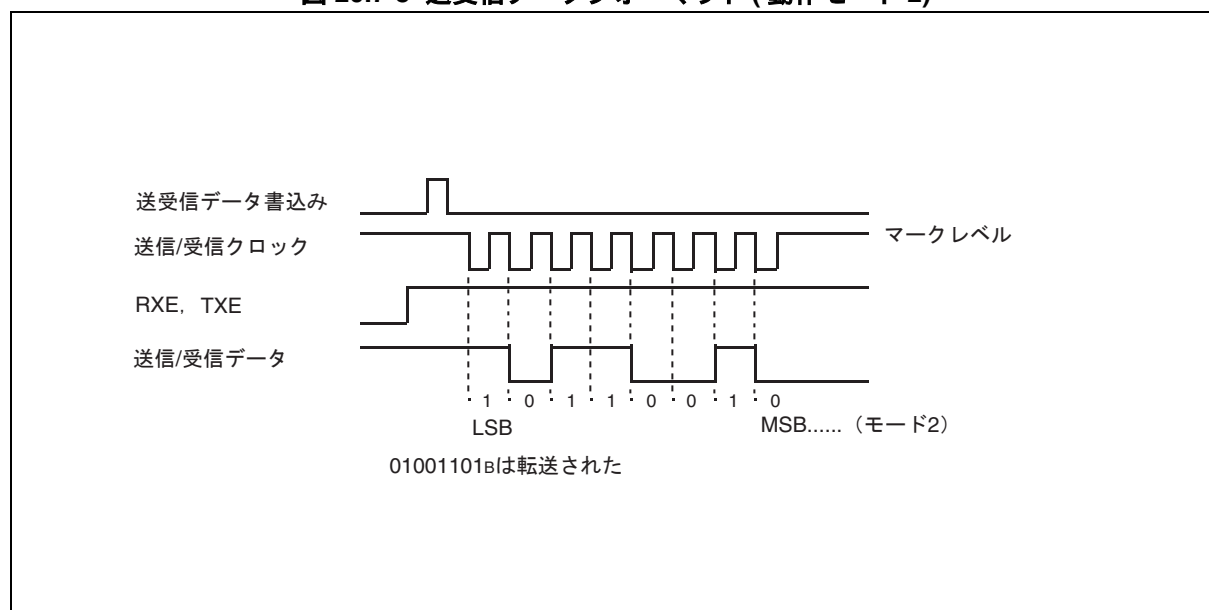
■ クロック同期モードでの動作

● 送受信データフォーマット

クロック同期モードでは、8 ビットのデータを LSB ファーストで送受信します。送受信データには、スタートビット、ストップビットは付加できません。

クロック同期モードの送受信データフォーマットを図 20.7-3 に示します。

図 20.7-3 送受信データフォーマット (動作モード 2)



● クロック供給

クロック同期 (I/O 拡張シリアル) 方式では、送受信ビット数に等しい数のクロックの供給が必要になります。

- 内部クロック (専用ボーレートジェネレータまたは、内部タイマ) を選択しているときは、データ送信を行う場合に、データ受信用同期クロックが自動的に生成されます。
- 外部クロックを選択しているときは送信側 UART の出力データレジスタ (SODR0/SODR1) にデータがあること、(SSR0/SSR1: TDRE=0) にデータがあることを確認した後に、1 バイト分のクロックを外部から供給する必要があります。

なお、送信前と送信後は、必ずマークレベル ("H") に戻してください

● エラー検出

オーバランエラーのみ検出できます。パリティエラー、フレーミングエラーは検出できません。

● 初期化

同期モードを使用する場合に、各制御レジスタの設定値を示します。

[シリアルモードレジスタ (SMR0/SMR1)]

MD1, MD0 : "10_B"

CS2, CS1, CS0 : クロックセレクタのクロック入力を指定

SCKE : 専用ボーレートジェネレータまたは内部タイマの場合 "1", クロック出力, 外部クロック (クロック入力) の場合 "0"

SOE : 送信する場合 "1", 受信だけの場合 "0"

[シリアル制御レジスタ (SCR0/SCR1)]

PEN : "0"

P, SBL, AD : これらのビットは意味を持ちません

CL : "1"(8 ビットデータ)

REC : "0"(初期化するため、エラーフラグクリア)

RXE, TXE : 少なくとも、どちらか一方を "1"

[シリアルステータスレジスタ (SSR0/SSR1)]

RIE : 割込みを使用する場合 "1", 割込みを使用しない場合 "0"

TIE : 割込みを使用する場合 "1", 割込みを使用しない場合 "0"

● 通信開始

シリアル出力データレジスタ (SODR0/SODR1) に送信データを書き込んだ場合、送信動作が開始されます。受信のみ行う場合でも通信を開始する場合には、仮の送信データをシリアル出力データレジスタに書き込んでください。

● 通信終了

1 フレームデータの送受信が終了した場合は、受信データフルフラグが "1" にセットされます (SSR0/SSR1: RDRF=1)。受信データを読み出す場合は、オーバランエラーフラグをチェックし (SSR0/SSR1: ORE)、通信が正常に行われたか確認してください。

20.7.3 双方向通信機能 (ノーマルモード)

動作モード 0, 2 では, 通常のシリアル双方向通信 (1 対 1 接続) が可能です。
動作モード 0 の場合は非同期方式で, 動作モード 2 の場合はクロック同期方式で通信を行います。

■ 双方向通信機能

UART を動作モード 0, 2 (ノーマルモード) で使用する場合は, 図 20.7-4 に示すような設定を行ってください。

図 20.7-4 UART の動作モード 0, 2 の設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SCR0/SCR1, SMR0/SMR1	PEN	P	SBL	CL	AD	REC	RXE	TXE	MD1	MD0	CS2	CS1	CS0	RST	SCKE	SOE
モード 0					X	0			0	0				X		
モード 2	0	X	X	1	X	0			1	0				X		

SSR0/SSR1, SIDR0/SIDR1, SODR0/SODR1	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE	変換データを設定 (書き込んだ場合) / 受信データを保持 (読み出した場合)							
モード 0																
モード 2	X		X													

DDR4 (UART0)

--	--	--	--	--	--	--	--

DDR6 (UART1)

--	--	--	--	--	--	--	--

: 使用ビット

X : 未使用ビット

1 : "1" を設定

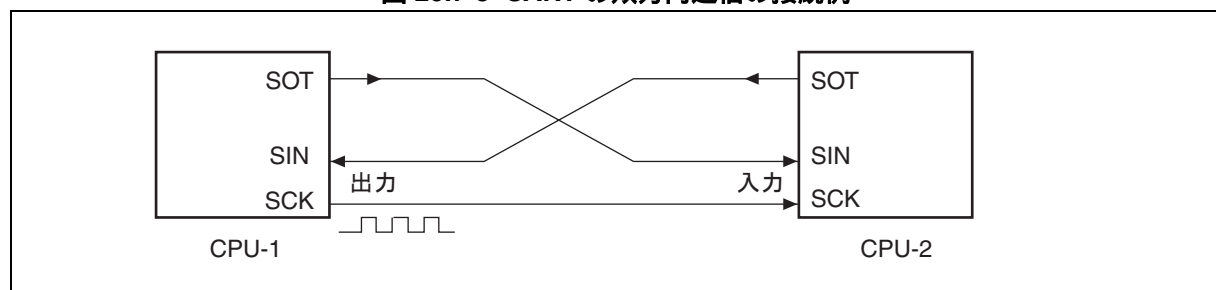
0 : "0" を設定

: 使用する端子に対応するビットに "0" を設定

● CPU 間の接続

図 20.7-5 に示すように, 2 つの CPU を接続します。

図 20.7-5 UART の双方向通信の接続例

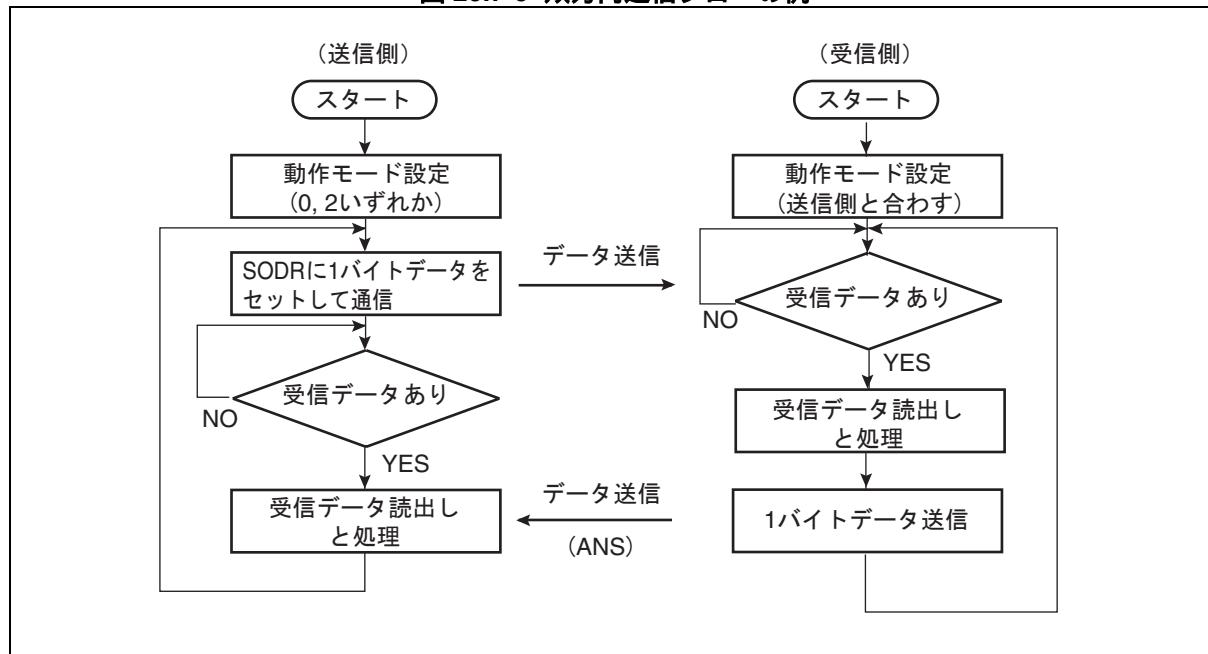


● 通信手順

通信は任意なタイミングで、送信データが準備できたときに送信側から開始します。受信側で送信データを受け取ると定期的に ANS(本例では、1 バイトごと) を返します。

双方向通信フローの例を、図 20.7-6 に示します。

図 20.7-6 双方向通信フローの例



20.7.4 マスタ/スレーブ型通信機能 (マルチプロセッサモード)

動作モード 1 では、複数の CPU をマスタ/スレーブ型に接続して通信を行うことができます。ただし、マスタ側のみ機能します。

■ マスタ/スレーブ型通信機能

UART を動作モード 1 (マルチプロセッサモード) で使用する場合は、図 20.7-7 に示す設定を行ってください。

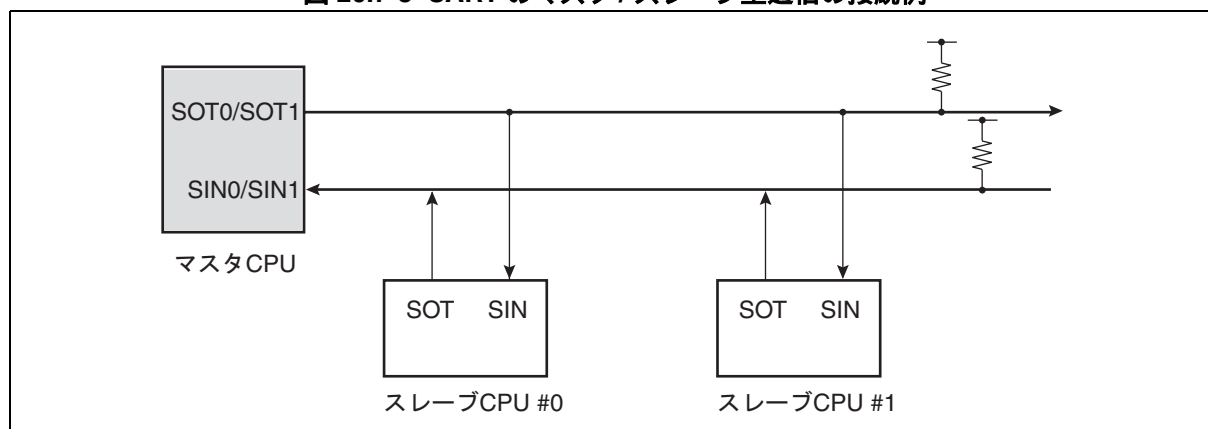
図 20.7-7 UART の動作モード 1 の設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SCR0/SCR1, SMR0/SMR1	PEN	P	SBL	CL	AD	REC	RXE	TXE	MD1	MD0	CS2	CS1	CS0	RST	SCKE	SOE
	X		1		0			0	1					X	0	
SSR0/SSR1, SIDR0/SIDR1, SODR0/SODR1	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE	送信データを設定 (書き込んだ場合) / 受信データを保持 (読み出した場合)							
	X															
DDR4 (UART0)																
DDR6 (UART1)																
: 使用ビット X : 未使用ビット 1 : "1" を設定 0 : "0" を設定 : 使用する端子に対応するビットに "0" を設定																

● CPU 間の接続

2 本の共通通信ラインに 1 つのマスタ CPU と、複数のスレーブ CPU を接続します。
UART は、マスタ側でのみ使用できます。接続例を図 20.7-8 に示します。

図 20.7-8 UART のマスタ/スレーブ型通信の接続例



● 機能選択

マスタ/スレーブ型通信での、動作モードと通信方式を表 20.7-2 に示します。

表 20.7-2 マスタ/スレーブ型通信機能の選択

	動作モード		データ	パリティ	同期方式	ストップビット
	マスタ CPU	スレーブ CPU				
アドレス送受信	動作モード 1	-	AD=1 + 8 ビットアドレス	なし	非同期	1 ビット または 2 ビット
データ送受信			AD=0 + 8 ビットデータ			

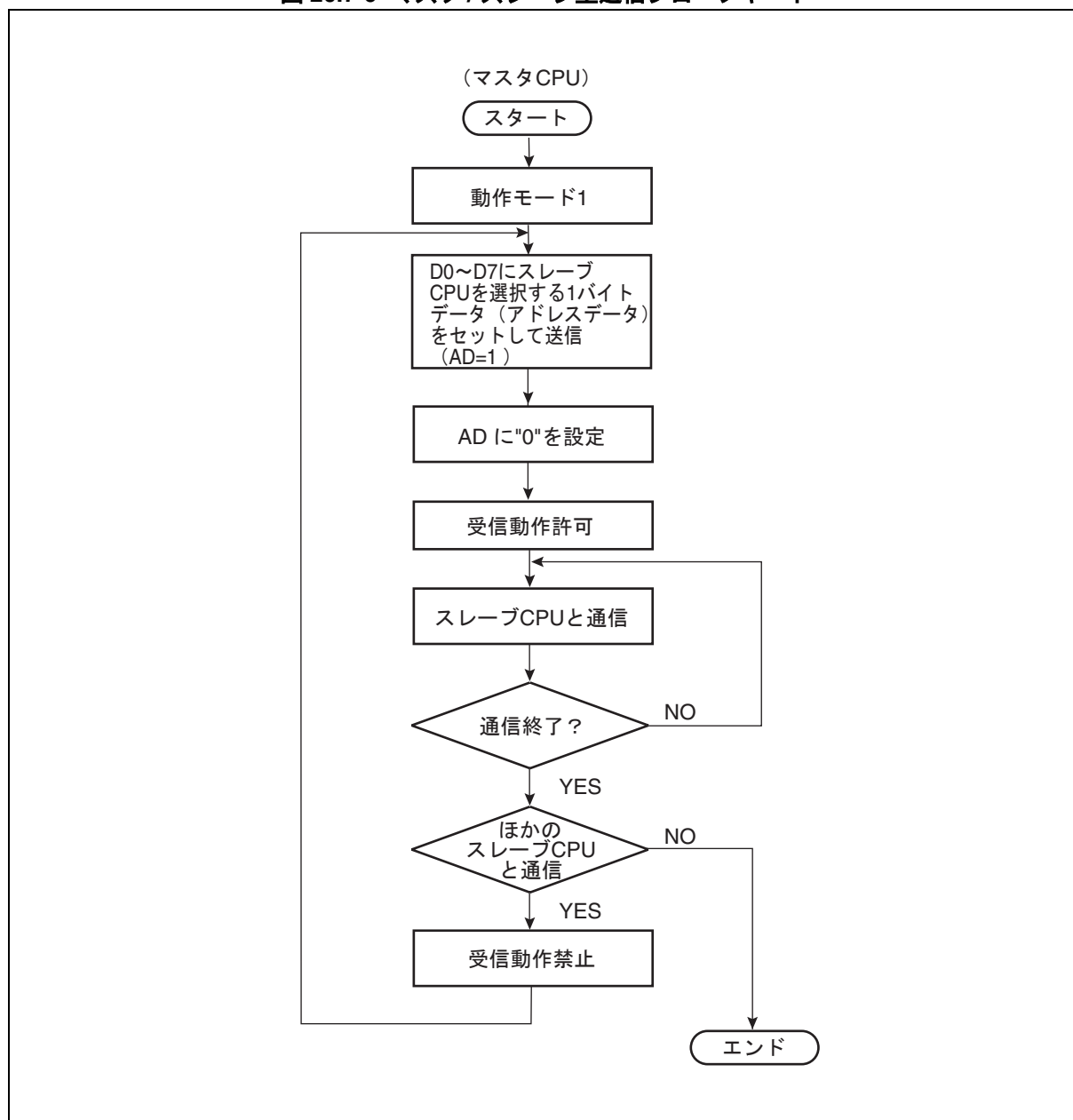
● 通信手順

通信は、マスタ CPU 側がアドレスデータを送信することによって開始します。

アドレスデータとは、AD ビットが "1" のデータであり、アドレス / データ選択ビット (SCR0/SCR1 レジスタの AD: bit11) の付加によって、通信先となるスレーブ CPU が選択されます。各スレーブ CPU はプログラムによりアドレスデータを判断し、割り当てられたアドレスと一致する場合に、マスタ CPU との通信 (通常データ) が開始します。

マスタ / スレーブ型通信のフローチャートを図 20.7-9 に示します。

図 20.7-9 マスタ / スレーブ型通信フローチャート



20.8 UART 使用上の注意

UART を使用するときは、以下の点に注意してください。

■ UART 使用上の注意

● 動作許可

UART には、送信と受信のそれぞれに対して送信動作許可ビット (SCR0/SCR1:TXE) と受信動作許可ビット (SCR0/SCR1:RXE) があります。

- リセット後の初期状態は、送受信とも動作禁止になっているため、送受信を行う前に動作許可ビットを "1" に設定する必要があります。
- 送受信の動作許可ビットを禁止に設定した場合、送受信を中止できます。

● 動作モードの設定について

- 動作モードの設定は、動作許可ビットを禁止に設定し、動作停止中に行ってください。
- 送受信中に動作モードを切り換えた場合は、送受信データを保証できません。

● 同期モードについて

UART のクロック同期モード (動作モード 2) は、クロック制御 (I/O 拡張シリアル) 方式を採用しており、スタートビット、ストップビットはデータに付加されません。

● 送信割込み許可のタイミング

送信データ書込み許可フラグビット (SSR0/SSR1:TRE) は初期値が "1" (送信データなし、送信データ書込み許可) にセットされているので、送信割込みを許可 (SSR0/SSR1:TIE=1) したと同時に、送信割込み要求が発生します。必ず、送信データを準備してから、送信割込みを許可 (SSR0/SSR1:TIE=1) してください。

第21章

ROM 修正機能

ROM 修正機能の機能と動作について説明します。

- 21.1 ROM 修正機能の概要
- 21.2 ROM 修正機能のブロックダイアグラム
- 21.3 ROM 修正機能レジスタ
- 21.4 ROM 修正機能の動作説明
- 21.5 ROM 修正機能使用例

21.1 ROM 修正機能の概要

対応するアドレスがプログラムアドレス検出レジスタに設定された値と等しい場合、CPU で読み出される命令コードは INT9 命令コードと強制的に置き換えられます。プログラムパッチアプリケーション機能は、INT#9 割込みルーチンで処理することで実行します。

■ プログラムアドレス検出レジスタ (× 2)

プログラムアドレス検出レジスタは 2 種類あり (PADR0/PADR1), それぞれ割込み許可ビットと割込みフラグがあります。

■ ROM 修正割込み

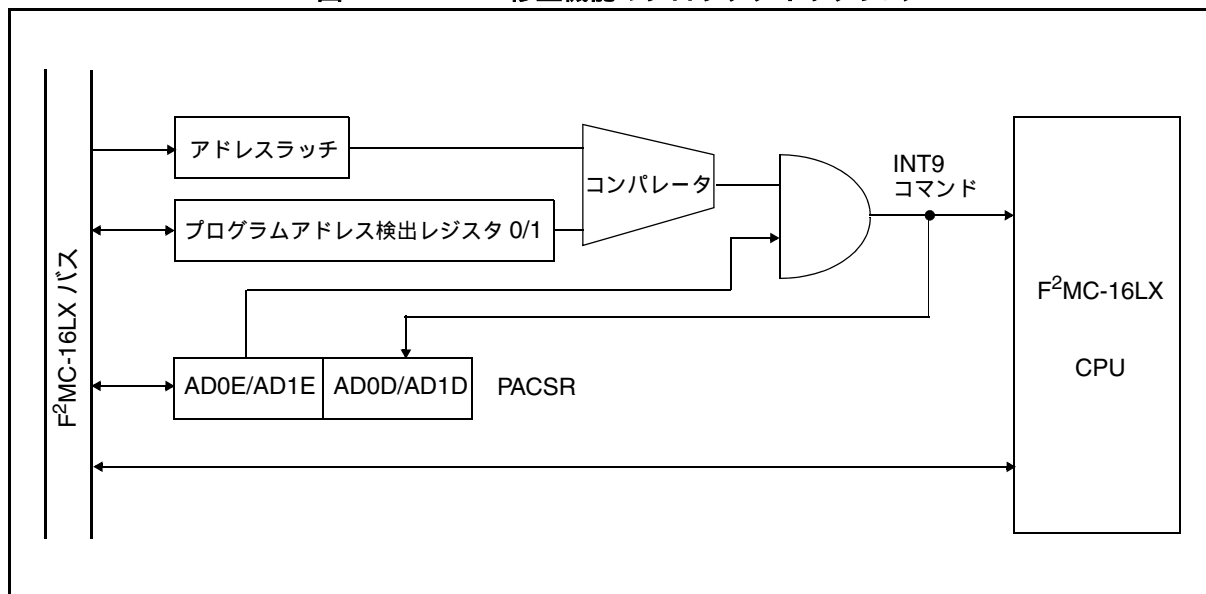
割込み許可ビットが "1" の場合、プログラムアドレス検出レジスタの設定値はアドレスと比較されます。その値がアドレスと一致した場合、割込みフラグビットに "1" が設定され、CPU に読み出される命令コードが INT9 命令コードと強制的に置き換えられます。割込みフラグビットは、命令を用いて "0" を書き込むことによって "0" にクリアされます。

21.2 ROM 修正機能のブロックダイアグラム

ROM 修正機能のブロックダイアグラムを以下に示します。

■ ROM 修正機能のブロックダイアグラム

図 21.2-1 ROM 修正機能のブロックダイアグラム



21.3 ROM 修正機能レジスタ

ROM 修正機能レジスタを示します。

■ ROM 修正機能レジスタ

図 21.3-1 ROM 修正機能のレジスタ

プログラムアドレス検出レジスタ 0/1

	上位バイト	中位バイト	下位バイト	
リード/ライト 初期値	PADRH0	PADRM0	PADRL0	PADR0
	PADRH1	PADRM1	PADRL1	PADR1
	(R/W)	(R/W)	(R/W)	
	(XXXXXXXX _B)	(XXXXXXXX _B)	(XXXXXXXX _B)	

プログラムアドレス検出制御ステータスレジスタ

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
-	-	-	-	AD1E	AD1D	AD0E	AD0D	XXXX0000 _B
(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	

21.3.1 プログラムアドレス検出レジスタ (PADR0/PADR1)

プログラムアドレス検出レジスタ (PADR0/PADR1) は 24 ビットのレジスタで、内部アドレスバスと比較されるアドレスを格納します。

■ プログラムアドレス検出レジスタ 0/1(PADR0/PADR1)

図 21.3-2 プログラムアドレス検出レジスタ

プログラムアドレス検出レジスタ 0/1			
	上位バイト	中位バイト	下位バイト
	PADRH0	PADRM0	PADRL0
	PADRH1	PADRM1	PADRL1
リード/ライト	(R/W)	(R/W)	(R/W)
初期値	(XXXXXXXX _B)	(XXXXXXXX _B)	(XXXXXXXX _B)
	PADR0 PADR1		

このレジスタに書き込まれた値は目標アドレスと比較されます。この値がアドレスと一致し、PACSR レジスタの対応する割込み許可ビットが "1" の場合、INT9 命令を生成することを CPU に要求するために、対応する割込みビットを "1" に設定されます。この対応する割込み許可ビットが "0" の場合、動作は行われません。

表 21.3-1 にプログラムアドレス検出レジスタと PACSR の対応を示します。

表 21.3-1 プログラムアドレス検出レジスタと PACSR の対応

プログラムアドレス検出レジスタ	割込み許可ビット	割込みビット
PADR0	AD0E	AD0D
PADR1	AD1E	AD1D

21.3.2 プログラムアドレス検出制御ステータスレジスタ (PACSR)

プログラムアドレス検出制御ステータスレジスタ (PACSR) は 8 ビットレジスタで ROM 修正機能の動作を制御します。

■ プログラムアドレス検出制御ステータスレジスタ (PACSR)

図 21.3-3 プログラムアドレス検出制御ステータスレジスタ

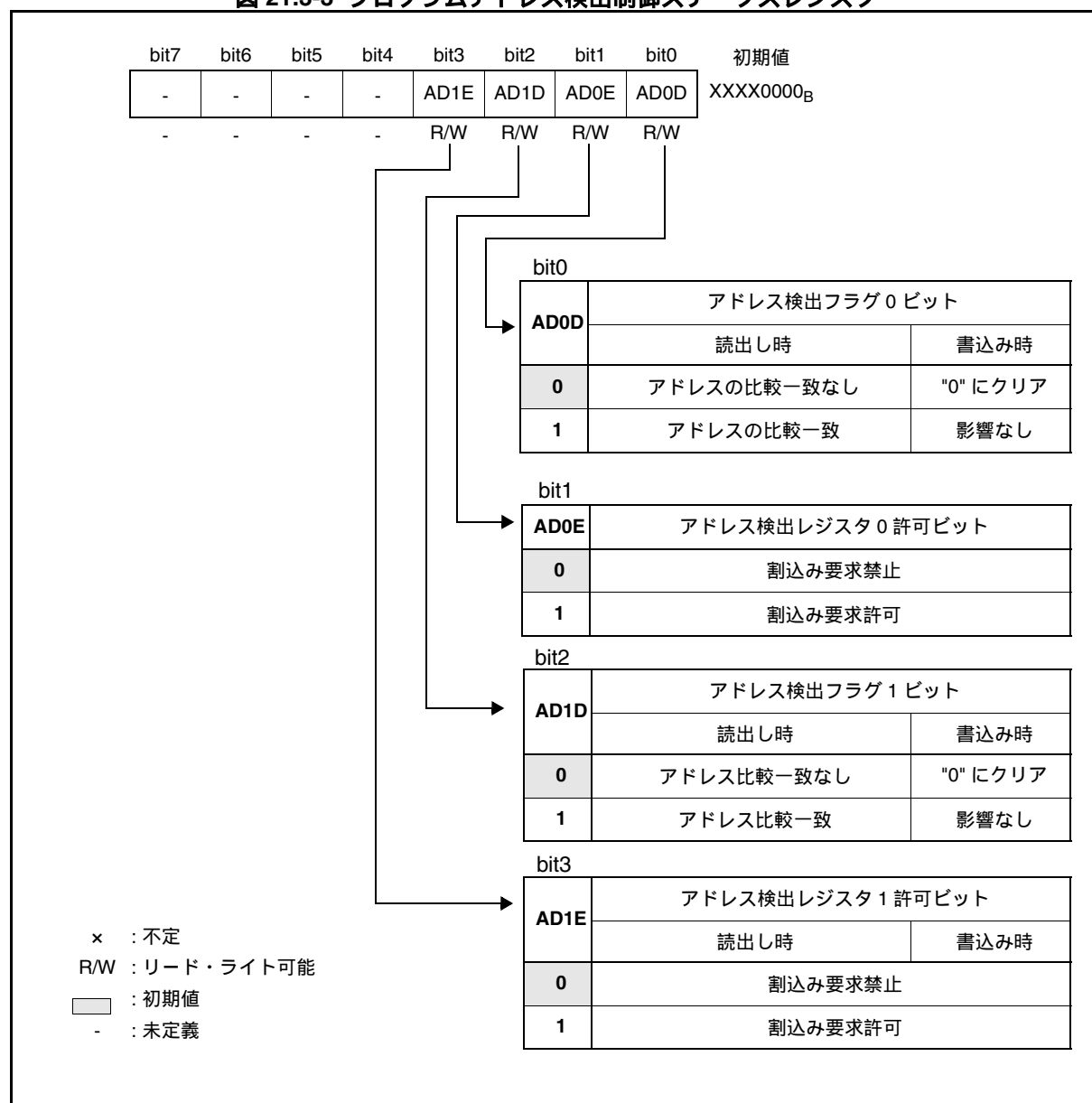


表 21.3-2 プログラムアドレス検出制御ステータスレジスタ

ビット名		機能
bit7 ～ bit4	予約ビット	必ず "0" に設定してください。
bit3	AD1E: アドレス検出 レジスタ 1 許可ビット	PADR1 動作許可ビット "1" に設定した場合: PADR1 レジスタの設定値とアドレスが比較され、2 つの値が等しければ、INT9 命令が発生し、AD1D ビットを "1" に設定されます。
bit2	AD1D: アドレス検出 フラグ 1 ビット	PADR1 アドレス一致検出ビット このビットを "1" に設定すると、PADR1 レジスタの設定値がアドレスと一 致したことを示します。 "0" に設定した場合: "0" にクリアされます。 "1" に設定した場合: 影響しません。
bit1	AD0E: アドレス検出 レジスタ 0 許可ビット	PADR0 動作許可ビット "1" に設定した場合: PADR0 レジスタの設定値はアドレスと比較されま す。2 つの値が等しければ、INT9 命令が発生し、 AD0D ビットを "1" に設定されます。
bit0	AD0D: アドレス検出 フラグ 0 ビット	PADR0 アドレス一致検出ビット このビットを "1" に設定すると、PADR0 レジスタの設定値がアドレスと 一致することを示します。 "0" に設定した場合: "0" にクリアされます。 "1" に設定した場合: 影響しません。

21.4 ROM 修正機能の動作説明

プログラムカウンタが、プログラムアドレス検出レジスタ (PADR) と同じアドレスを指定する場合、INT9 命令が実行されます。ROM 修正機能は、INT9 命令ルーチンを処理することで作動します。

■ ROM 修正機能の動作説明

- 対応するアドレスがアドレス検出レジスタの設定値と等しい場合、CPU によって読み込まれる命令コードは、INT9 命令コード (01_H) と強制的に置換させられます。そのため、設定した命令を実行するときに、CPU は INT9 命令を実行します。
- INT#9 割込みルーチンで処理することによって、プログラムパッチアプリケーション機能が実行されます。
- アドレス検出レジスタには 2 つあり、それぞれ割込み許可ビットと割込みフラグが備え付けられています。アドレスがアドレス検出レジスタに設定した値と等しく、割込み許可ビットが "1" の場合、次のように仮定します：
割込みフラグが "1" に設定され、CPU によって読み込まれる命令コードは、INT9 命令コードと強制的に置換させられます。インストラクションを用いて "0" を書き込むことで、割込みフラグは "0" にクリアされます。

< 注意事項 >

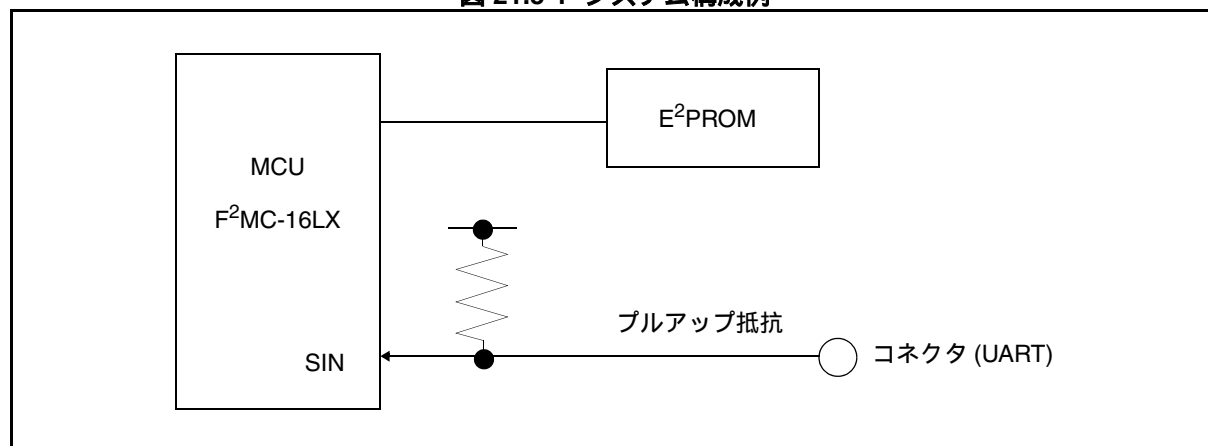
- アドレス検出レジスタにインストラクションの第 1 バイト以降のアドレスが設定された場合、アドレス一致検出機能は働かなくなります。設定したアドレスの値が "01_H" と置換されるので、間違った命令が実行されるか無効なアドレスがアクセスされます。アドレス検出レジスタに設定した値を変更する前に、割込み許可ビットを "0" に設定してください。割込み許可ビットが "1" のときにデータが書き込まれた場合、書き込み動作中にアドレスが誤って検出され、誤動作を起こす原因になります。
- プログラムアドレス検出レジスタは 001FF0_H ~ 001FF5_H にレジスタをもっているため、MB90F828B の RAM 領域と重なりますので、MB90F828B 使用時、本機能をご使用の場合には RAM アクセスを行わないようにしてください。

21.5 ROM 修正機能使用例

アドレス一致検出機能使用例を示します。

■ システム構成

図 21.5-1 システム構成例



■ E²PROM メモリマップ

表 21.5-1 に E²PROM メモリマップを示します。

表 21.5-1 E²PROM メモリマップ

アドレス	意味
0000 _H	パッチプログラム番号 0 のバイト数 (0 : プログラムエラーはなし)
0001 _H	プログラムアドレス番号 0 の bit 7 ~ bit 0
0002 _H	プログラムアドレス番号 0 の bit 15 ~ bit 8
0003 _H	プログラムアドレス番号 0 の bit 24 ~ bit 26
0004 _H	パッチプログラム番号 1 のバイト数 (0 : プログラムエラーはなし)
0005 _H	プログラムアドレス番号 1 の bit 7 ~ bit 0
0006 _H	プログラムアドレス番号 1 の bit 15 ~ bit 8
0007 _H	プログラムアドレス番号 1 の bit 24 ~ bit 16
0008 _H ~ 0010 _H ⁺ パッチプログラム番号 0 のバイト数	パッチプログラム番号 0 の本体

■ 初期状態

E²PROM のデータはすべて "0" です。

■ プログラムエラーが発生した場合

パッチプログラムの本体とそのアドレスは、コネクタ (UART) を通じて MCU に転送されます。MCU は E²PROM へ情報を書き込みます。

■ リセットシーケンス

リセットシーケンスが終了した後、MCU は E²PROM の値を読み出します。パッチプログラムのバイト数が "0" でない場合、MCU はパッチプログラム（本体）を読み出し、RAM へ書き込みます。そして MCU はプログラムアドレスを PADR0 または PADR1 に設定し、プログラムを動作させます。RAM に書き込まれるプログラムの最初のアドレスは、各アドレス検出レジスタの指定のように RAM に配置されます。

■ INT9 割込み

割込みルーチン実行中に、割込みに許可されたアドレスの割込みフラグをチェックし、対応するプログラムへ分岐します。割込みによってスタックされた情報は破棄されます。割込みフラグもクリアされます。

図 21.5-2 システム構成の例

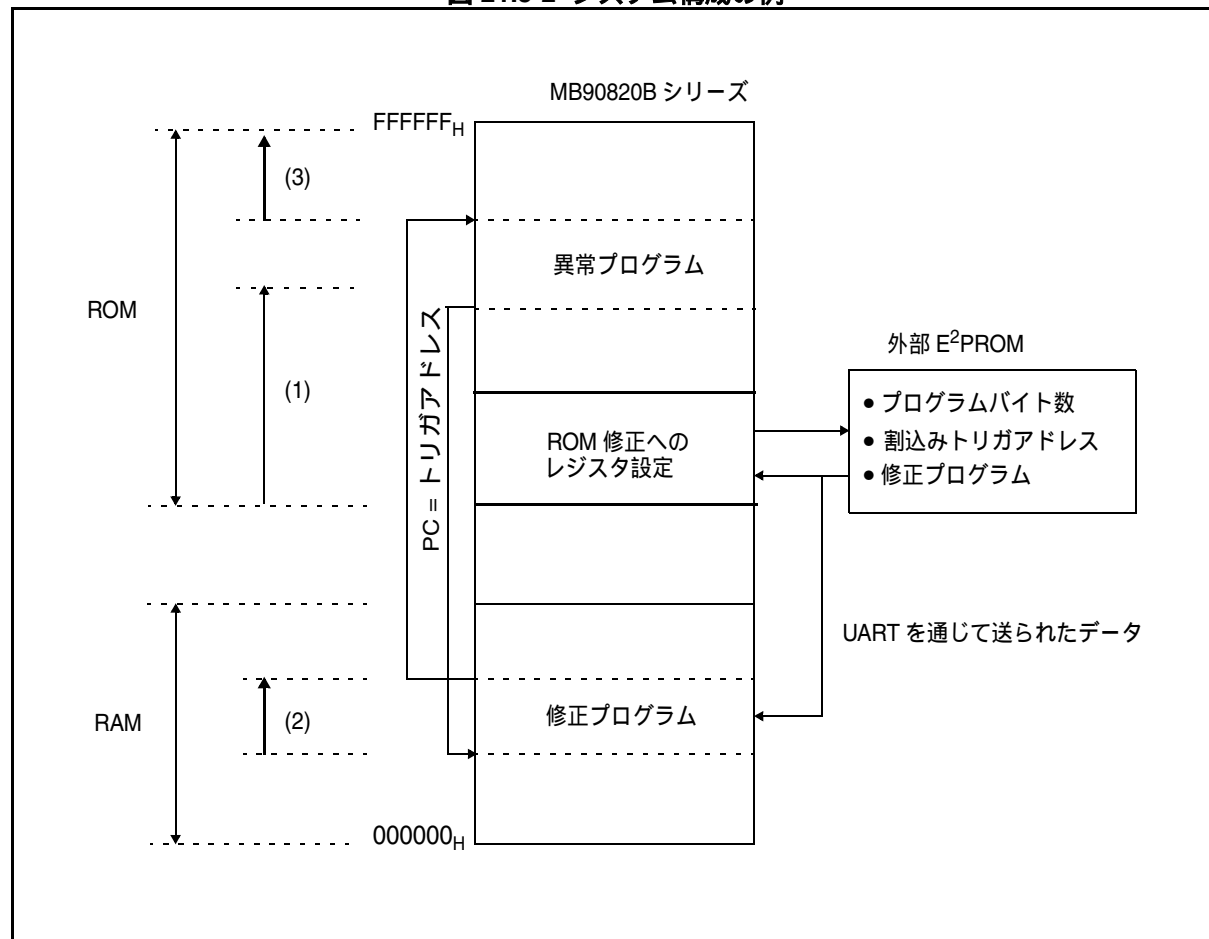
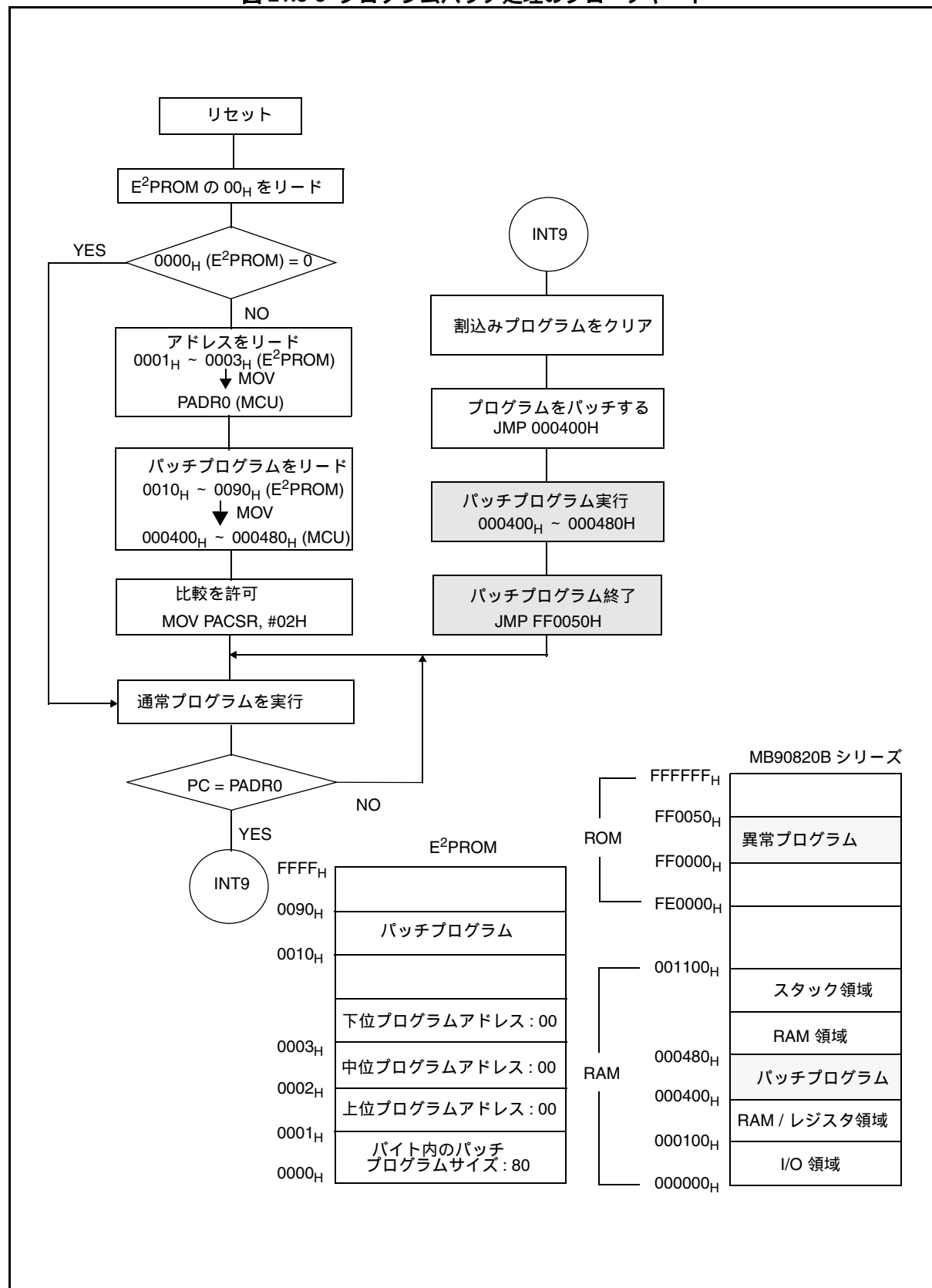


図 21.5-3 プログラムパッチ処理のフローチャート



第22章

ROM ミラー機能 選択モジュール

MB90820B シリーズの ROM ミラー機能選択モジュールの機能と動作について説明します。

22.1 ROM ミラー機能選択モジュールの概要

22.2 ROM ミラー機能選択レジスタ (ROMM)

22.1 ROM ミラー機能選択モジュールの概要

ROM ミラー機能選択モジュールでは、レジスタを設定することで 00 バンクから ROM に配置されている FF バンクにアクセスできます。

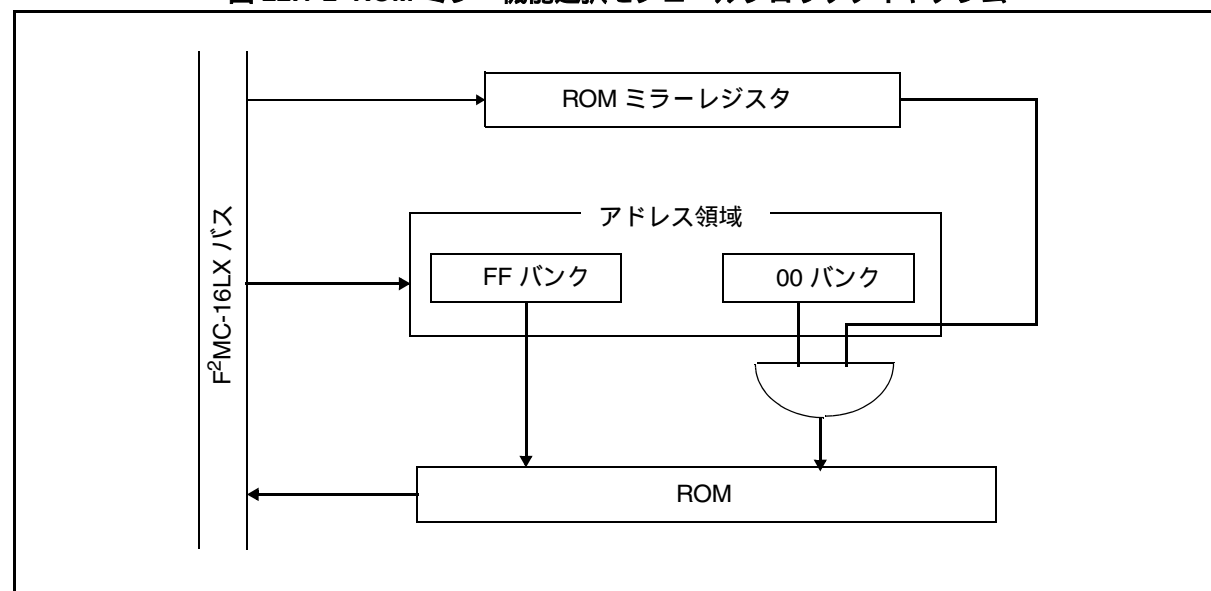
■ ROM ミラー機能選択モジュールのレジスタ

図 22.1-1 ROM ミラー機能選択モジュールのレジスタ

ROM ミラー機能選択レジスタ								初期値
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	
-	-	-	-	-	-	-	MI	XXXXXXXX _{1B}
(-)	(-)	(-)	(-)	(-)	(-)	(-)	(W)	

■ ROM ミラー機能選択モジュールブロックダイアグラム

図 22.1-2 ROM ミラー機能選択モジュールブロックダイアグラム



22.2 ROM ミラー機能選択レジスタ (ROMM)

ROM ミラー機能選択レジスタ (ROMM) は , ミラー機能を許可します。

■ ROM ミラー機能選択レジスタ (ROMM)

図 22.2-1 ROM ミラー機能選択レジスタ (ROMM)

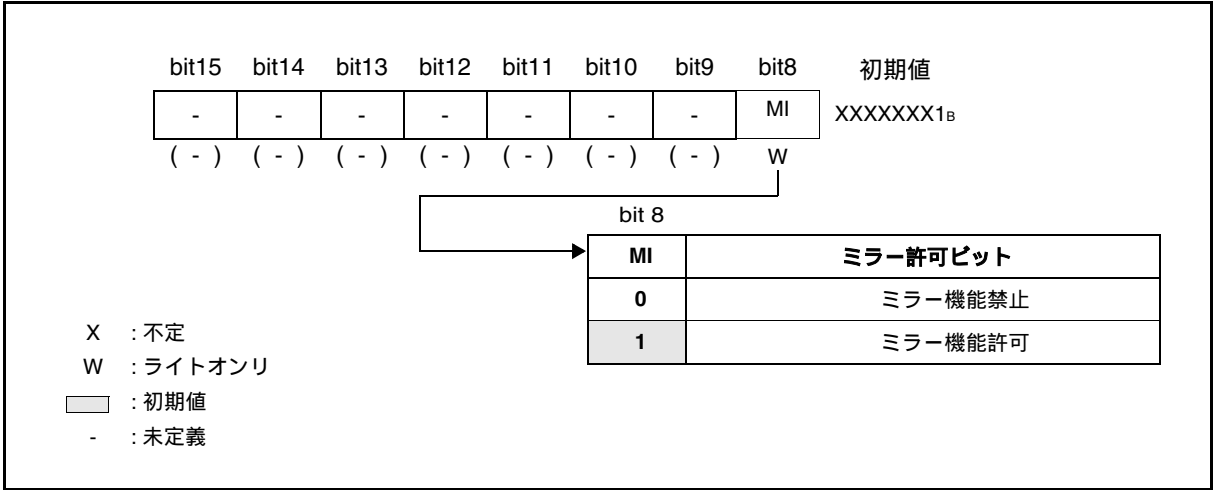


表 22.2-1 ROM ミラー機能選択レジスタ (ROMM) の機能

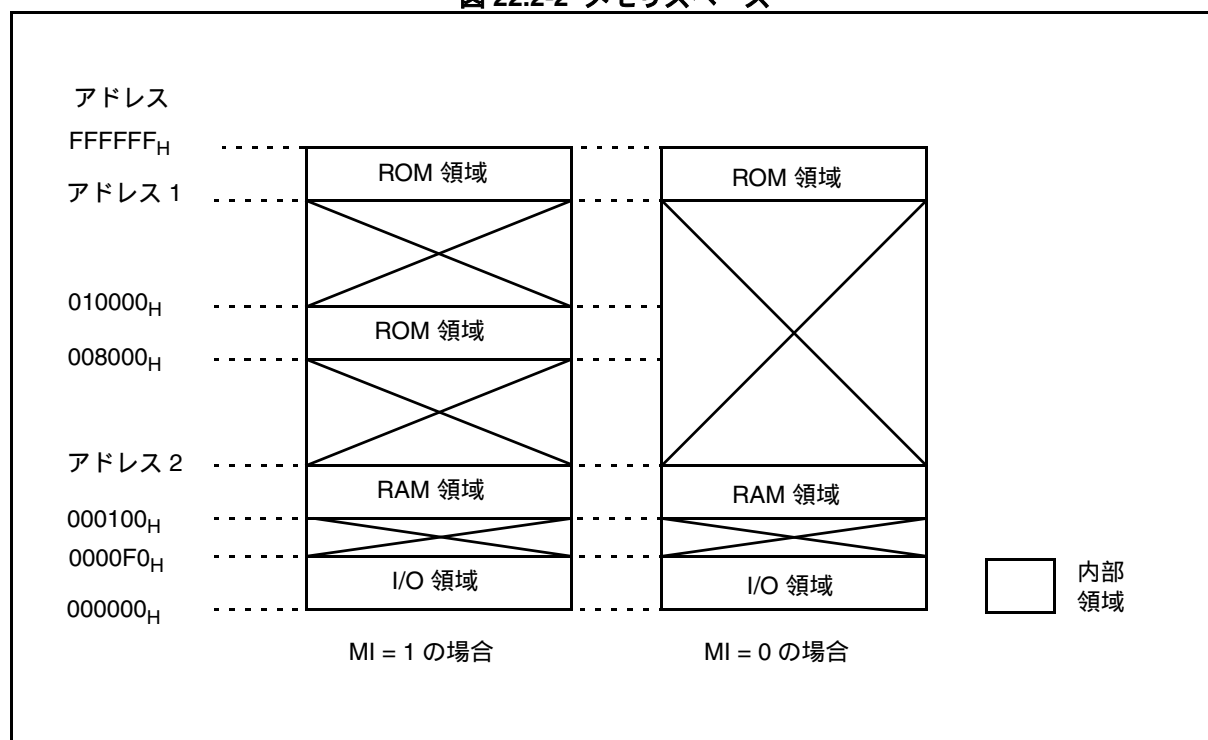
ビット名		機能
bit15 ~ bit9	未定義ビット	読み出した場合 : 値は不定です。 書き込んだ場合 : 影響しません。
bit8	MI: ミラー許可ビット	"0" に設定した場合 : バンク 00 において機能が禁止されます。書込み専用です。 "1" に設定した場合 : バンク FF の ROM データは , バンク 00 から読み出されます。

< 参考 >

バンク 00 は , "008000_H" ~ "00FFFF_H" 番地から "FF8000_H" ~ "FFFFFF_H" 番地にアクセスしますので , "FFF000_H" ~ "FF7FFF_H" 番地のアドレスは , ROM ミラー機能を設定してもアクセスできません。

	MB90822B	MB90823B	MB90F822B	MB90F823B	MB90F828B	MB90V820B
アドレス 1	FF0000 _H	FE0000 _H	FF0000 _H	FE0000 _H	FE0000 _H	FE0000 _H
アドレス 2	0010FF _H	0010FF _H	0010FF _H	0010FF _H	0020FF _H	0040FF _H

図 22.2-2 メモリスペース



第23章

512K/1024K ビット フラッシュメモリ

512K/1024K ビットフラッシュメモリの機能および動作について説明します。

- 23.1 512K/1024K ビットフラッシュメモリの概要
- 23.2 512K/1024K フラッシュメモリのレジスタとセクタ構成
- 23.3 フラッシュメモリ制御ステータスレジスタ (FMCS)
- 23.4 フラッシュメモリ自動アルゴリズム起動方法
- 23.5 自動アルゴリズム実行状態の確認
- 23.6 フラッシュメモリ書込み / 消去の詳細説明
- 23.7 フラッシュセキュリティの特長

23.1 512K/1024K ビットフラッシュメモリの概要

512K/1024K フラッシュメモリへのデータ書込み / 消去の方法には、以下の 3 とおり
の方法があります。

1. パラレルライタ (ミナトエレクトロニクス製モデル 1890A)
 2. シリアル専用ライタ (横河デジタルコンピュータ株式会社製 AF200)
 3. プログラム実行による書込み / 消去
- 以下に、「3. プログラム実行による書込み / 消去」について解説します。

■ 512K/1024K ビットフラッシュメモリの概要

512K ビットフラッシュメモリは、CPU メモリマップ上の FF バンクに配置され、1024K
ビットフラッシュメモリは、CPU メモリマップ上の FE および FF バンクに配置されま
す。フラッシュメモリインタフェース回路の機能により、マスク ROM と同様に CPU
からのリードアクセスおよびプログラムアクセスができます。

フラッシュメモリへの書込み / 消去は、フラッシュメモリインタフェース回路を介して
CPU からの命令で行えるため、実装状態での書換えができ、プログラムやデータの改善
が効率よく行うことができます。ただし、許可セクタプロテクタなどのセクタ動作を
使用できません。

■ 512K/1024K ビットフラッシュメモリの特長

- 512K ビット : 64K ワード × 8 ビット / 32K ワード × 16 ビット (16K+8K+8K+32K)
セクタ構成
- 1024K ビット : 128K ワード × 8 ビット / 64K ワード × 16 ビット (64K+16K+8K+8K+32K)
セクタ構成
- 自動プログラムアルゴリズム (Embedded Algorithm: MBM29F400TA と同様)
- 消去一時停止 / 消去再開機能の搭載
- データポーリング、トグルビットによる書込み / 消去完了検出
- CPU 割込みによる書込み / 消去の完了検出
- JEDEC 標準タイプコマンドとの互換性
- セクタごとの消去が可能 (セクタ組合せ自由)
- 書込み / 消去回数 (最小) 10,000 回
- フラッシュセキュリティの特長

■ フラッシュメモリ書込み / 消去

- フラッシュメモリは、書込みと読出しを同時に行うことはできません。
- フラッシュメモリにデータの書込み / 消去動作を行う場合には、フラッシュメモリ
上にあるプログラムをいったん RAM にコピーし、RAM にコピーしたプログラムを
実行することにより、フラッシュメモリへの書込みを行うことができます。

■ フラッシュメモリのレジスタ

図 23.1-1 フラッシュメモリ制御ステータスレジスタ (FMCS)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	初期値
INTE	RDYINT	WE	RDY	予約	予約	予約	予約	000X0000 _B
(R/W)	(R/W)	(R/W)	(R)	(-)	(-)	(-)	(-)	

23.2 512K/1024K フラッシュメモリのレジスタとセクタ構成

図 23.2-1 および図 23.2-2 に 512K ビットフラッシュメモリのセクタ構成を示します。図中アドレスには、各セクタの上位アドレスと下位アドレスを示します。

■ セクタ構成

CPU から 512K ビットフラッシュメモリをアクセスする場合 FF バンクレジスタに SA0 ~ SA3 が配置されています。

図 23.2-1 512K ビットフラッシュメモリのセクタ構成

フラッシュメモリ	CPUアドレス	ライターアドレス*
SA3 (16Kバイト)	FFFFFFH	7FFFFH
	FFC000H	7C000H
SA2 (8Kバイト)	FFBFFFH	7BFFFH
	FFA000H	7A000H
SA1 (8Kバイト)	FF9FFFH	79FFFH
	FF8000H	78000H
SA0 (32Kバイト)	FF7FFFH	77FFFH
	FF0000H	70000H

※: ライターアドレスとは、フラッシュメモリにパラレルライターでデータ書込みを行う場合、CPU アドレスに相当するアドレスです。汎用ライターを使用し書込み / 消去を行う場合は、このライターアドレスで書込み / 消去を行います。

CPU から 1024K ビットフラッシュメモリをアクセスする場合 FE および FF バンクレジスタに SA0 ~ SA4 が配置されています。

図 23.2-2 1024K ビットフラッシュメモリのセクタ構成

フラッシュメモリ	CPUアドレス	ライタアドレス*
SA4 (16Kバイト)	FFFFFFH	7FFFFH
	FFC000H	7C000H
SA3 (8Kバイト)	FFBFFFH	7BFFFH
	FFA000H	7A000H
SA2 (8Kバイト)	FF9FFFH	79FFFH
	FF8000H	78000H
SA1 (32Kバイト)	FF7FFFH	77FFFH
	FF0000H	70000H
SA0 (64Kバイト)	FEFFFFH	6FFFFH
	FE0000H	60000H

*: ライタアドレスとは、フラッシュメモリにパラレルライタでデータ書込みを行う場合、CPU アドレスに相当するアドレスです。汎用ライタを使用し書込み / 消去を行う場合は、このライタアドレスで書込み / 消去を行います。

23.3 フラッシュメモリ制御ステータスレジスタ (FMCS)

フラッシュメモリ制御ステータスレジスタ (FMCS) の機能を図 23.3-1 に示します。

■ フラッシュメモリ制御ステータスレジスタ (FMCS)

図 23.3-1 フラッシュメモリ制御ステータスレジスタ (FMCS)

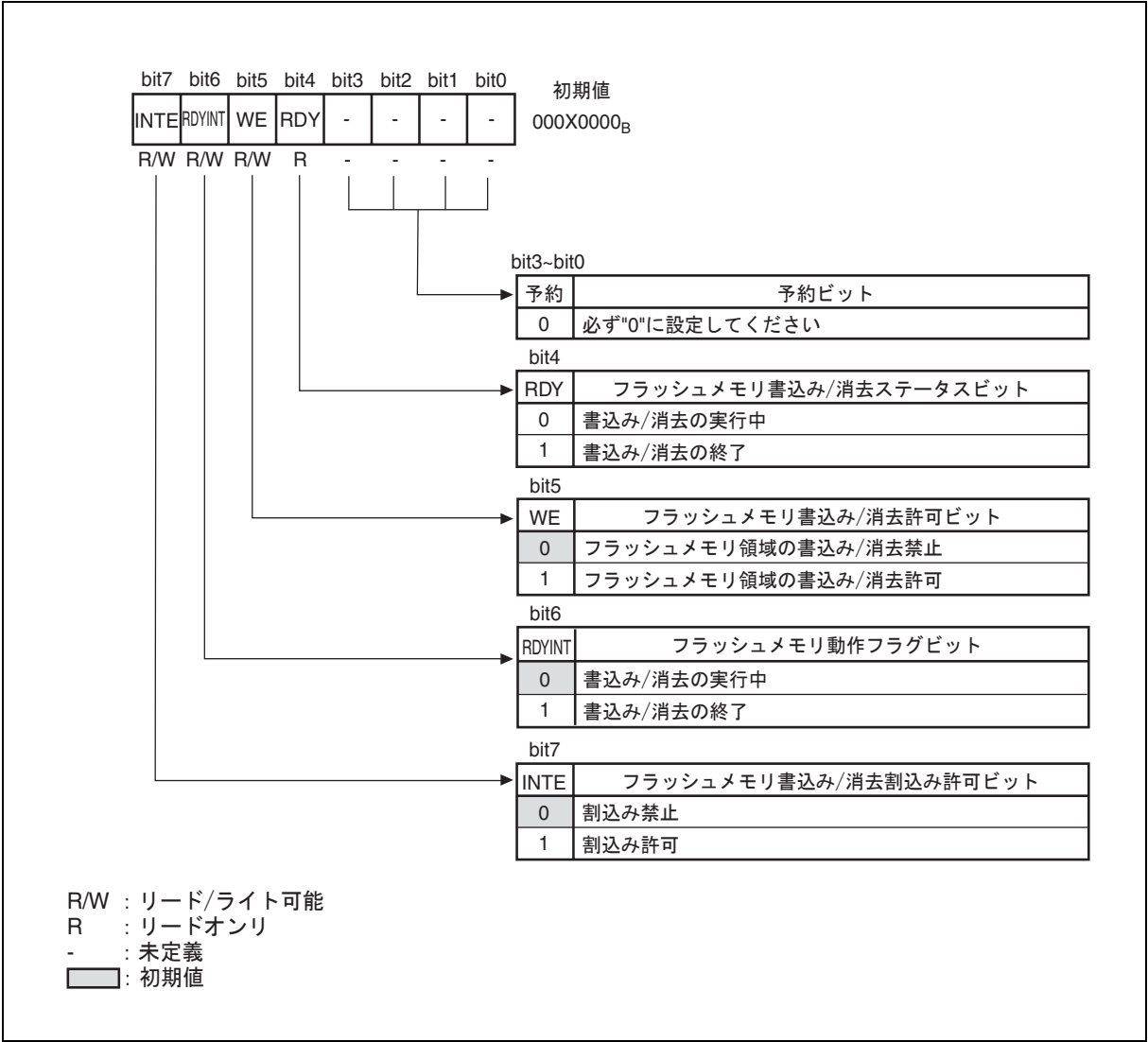
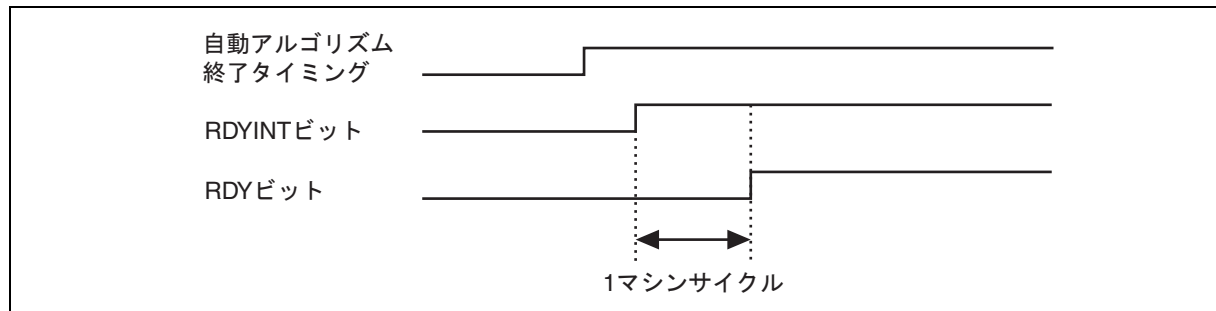


表 23.3-1 フラッシュメモリ制御ステータスレジスタ (FMCS) の機能

ビット名		機能
bit7	INTE: フラッシュメモリ書込み / 消去割込み許可ビット	フラッシュメモリの書込み / 消去の終了で CPU に割込みを発生させるビットです。 INTE ビットが "1" でかつ RDYINT ビットが "1" のときに CPU へ割り込みが発生します。INTE ビットが "0" であれば割込みは発生しません。 "0" に設定した場合 : 書込み / 消去終了での割込み許可 "1" に設定した場合 : 書込み / 消去終了での割込み禁止
bit6	RDYINT: フラッシュメモリ動作 フラグビット	フラッシュメモリの動作状態を示すビットです。 フラッシュメモリの書込み / 消去が終了した場合 "1" となります。フラッシュメモリ書込み / 消去後このビットが "0" の間は、フラッシュメモリへの書込み / 消去はできません。書込み / 消去が終了して "1" に設定された後は、フラッシュメモリへの書込み / 消去が可能になります。 "0" 書込みによって "0" にクリアされ、"1" の書込みは無視されます。フラッシュメモリ自動アルゴリズム (「23.4 フラッシュメモリ自動アルゴリズム起動方法」参照) 終了タイミングで、"1" にセットされます。リードモディファイライト (RMW) 命令を使用した場合は、必ず "1" が読み出されます。 "0" に設定した場合 : 書込み / 消去動作実行中 "1" に設定した場合 : 書込み / 消去動作終了 (割込み要求発生)
bit5	WE: フラッシュメモリ書込み / 消去許可ビット	フラッシュメモリ領域へのライトイネーブルビットです。 このビットが "1" のとき、FF バンクへのコマンドシーケンス (「23.4 フラッシュメモリ自動アルゴリズム起動方法」参照) 発行後の書込みは、フラッシュメモリ領域への書込みになります。本ビットが "0" のとき、書込み / 消去の信号は発生されません。このビットはフラッシュメモリの書込み / 消去のコマンドを起動する際に使用します。 "0" に設定した場合 : フラッシュメモリ書込み / 消去禁止 "1" に設定した場合 : フラッシュメモリ書込み / 消去許可
bit4	RDY: フラッシュメモリ書込み / 消去ステータスビット	フラッシュメモリの書込み / 消去許可ビットです。 <ul style="list-style-type: none"> この RDY ビットが "0" の場合は、フラッシュメモリへの書込み / 消去はできません。 この RDY ビットが "0" の場合でも、読出し / リセットコマンド、セクタ消去一時停止コマンドは受け付けることができます。 "0" に設定した場合 : 書込み / 消去動作実行中 "1" に設定した場合 : 書込み / 消去動作終了 (次データ書込み / 消去許可)
bit3 ~ bit0	予約: 予約ビット	必ず "0" に設定してください。

< 注意事項 >

フラッシュメモリ動作フラグビット (RDYINT) とフラッシュメモリ書込み / 消去ステータスビット (RDY) は同時には変化しません。どちらかのビットで書込み / 消去の終了を判定するようにプログラムを作成してください。



23.4 フラッシュメモリ自動アルゴリズム起動方法

フラッシュメモリの自動アルゴリズムを起動するコマンドには、読出し / リセット、書込み、チップ消去、セクタ消去の 4 種類があり、セクタ消去については一時停止と再開の制御ができます。

■ コマンドシーケンス表

表 23.4-1 に、フラッシュメモリの書込み / 消去時に使用するコマンドの一覧を示します。コマンドレジスタにライトするデータはすべてバイト長ですが、ワードアクセスで書き込んでください。ワードアクセスで書き込んだ場合の上位バイトのデータは無視されます。

表 23.4-1 コマンドシーケンス表

コマンド シーケ ス	バスラ イトア クセス	1st バスライト サイクル		2nd バスライトサ イクル		3rd バスライト サイクル		4th バスライト サイクル		5th バスライト サイクル		6th バスライト サイクル	
		アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ
読出し / リセット*	1	YYYXXX _H	XXF0 _H	-	-	-	-	-	-	-	-	-	-
読出し / リセット*	4	YYYAAA _H	XXAA _H	YYY554 _H	XX55 _H	YYYAAA _H	XXF0 _H	RA	RD	-	-	-	-
書込み	4	YYYAAA _H	XXAA _H	YYY554 _H	XX55 _H	YYYAAA _H	XXA0 _H	PA	PD	-	-	-	-
チップ消去	6	YYYAAA _H	XXAA _H	YYY554 _H	XX55 _H	YYYAAA _H	XX80 _H	YYYAAA _H	XXAA _H	YYY554 _H	XX55 _H	YYYAAA _H	XX10 _H
セクタ消去	6	YYYAAA _H	XXAA _H	YYY554 _H	XX55 _H	YYYAAA _H	XX80 _H	YYYAAA _H	XXAA _H	YYY554 _H	XX55 _H	SA	XX30 _H
セクタ消去一時停止	アドレス YYYXXX _H , データ "XXB0 _H " 入力で、セクタ消去中の消去一時停止												
セクタ消去再開	アドレス YYYXXX _H , データ "XX30 _H " 入力で、セクタ消去一時停止後、消去開始												

RA : 読出しアドレス

PA : 書込みアドレス

SA : セクタアドレス (セクタ内の任意の 1 アドレスを指定)

RD : 読出しデータ

PD : 書込みデータ

YYY: フラッシュメモリ領域中の、任意のアドレスの上位 12 ビット

* : 2 種類の読出し / リセットコマンドはどちらもフラッシュメモリを読出しモードにリセットすることができます。

< 注意事項 >

表中のアドレスは、CPU メモリマップ上の値です。アドレスおよびデータはすべて 16 進数で表記しています。ただし、"X" は任意の値です。

23.5 自動アルゴリズム実行状態の確認

フラッシュメモリでは、書込み / 消去のフローを自動アルゴリズムで行うため、フラッシュメモリ内部の動作状態や動作完了した事を知らせるハードウェアを持ちます。この自動アルゴリズムは、下記のハードウェアシーケンスフラグによって内蔵フラッシュメモリの動作状態の確認ができます。

■ ハードウェアシーケンスフラグ

ハードウェアシーケンスフラグは、DQ7, DQ6, DQ5, DQ3 の 4 ビットの出力で構成されます。それぞれがデータポーリングフラグ (DQ7), トグルビットフラグ (DQ6), タイミングリミット超過フラグ (DQ5), セクタ消去タイマフラグ (DQ3) の機能を持ちます。これにより、書込み / チップ・セクタ消去終了、消去コードライトが有効かを確認できます。ハードウェアシーケンスフラグを参照するには、コマンドシーケンス (「23.4 フラッシュメモリ自動アルゴリズム起動方法」) 設定後にフラッシュメモリ内部の対象セクタのアドレスにリードアクセスすることで参照できます。

表 23.5-1 にハードウェアシーケンスフラグのビット割当てを示します。

表 23.5-1 ハードウェアシーケンスフラグのビット割当て

ビット No.	7	6	5	4	3	2	1	0
ハードウェアシーケンスフラグ	DQ7	DQ6	DQ5	-	DQ3	-	-	-

- 自動書込み / チップ・セクタ消去が実行中か、終了しているか判断するためには、ハードウェアシーケンスフラグを確認するか、またはフラッシュメモリ制御ステータスレジスタのフラッシュメモリ書込み / 消去ステータスビット (FMCS: RDY) を確認してください。書込み / 消去の終了後は、読出し / リセット状態に戻ります。
- 書込み / 消去プログラムを作成する場合には、いずれかのフラグで自動書込み / 消去の終了を確認後に、データの読出しの処理を行ってください。
- 2回目以降のセクタ消去コードライトが有効であるかについても、ハードウェアシーケンスフラグによって確認できます。

表 23.5-2 に、ハードウェアシーケンスフラグ機能の一覧を示します。

表 23.5-2 ハードウェアシーケンスフラグ機能の一覧

状態		DQ7	DQ6	DQ5	DQ3
正常動作時の 状態変化	書込み 書込み完了 (書込みアドレス指定時)	$\overline{\text{DQ7}}$ DATA: 7	Toggle DATA: 6	0 DATA: 5	0 DATA: 3
	チップ・セクタ消去 消去完了	0 1	Toggle Stop	0 1	1
	セクタ消去ウェイト 消去開始	0	Toggle	0	0 1
	消去 セクタ消去一時停止 (消去中のセクタ)	0 1	Toggle 1	0	1 0
	セクタ消去一時停止 消去再開 (消去中のセクタ)	1 0	1 Toggle	0	0 1
	セクタ消去一時停止中 (消去中ではないセクタ)	DATA: 7	DATA: 6	DATA: 5	DATA: 3
異常動作	書込み	$\overline{\text{DQ7}}$	Toggle	1	0
	チップ・セクタ消去	0	Toggle	1	1

23.5.1 データポーリングフラグ (DQ7)

データポーリングフラグ (DQ7) は、自動アルゴリズム実行が進行中もしくは終了状態であることをデータポーリング機能によって知らせるハードウェアシーケンスフラグです。データポーリングフラグの状態遷移を表 23.5-3 と表 23.5-4 に示します。

■ 書込みの場合

- 自動書込みアルゴリズム実行中にリードアクセスした場合、指定したリードアドレスに関係なく、最後に書き込まれたデータの bit7 を反転させた値を出力します。
- 自動書込みアルゴリズム終了時にリードアクセスを行った場合、フラッシュメモリはリードアクセスを行ったアドレスの bit7 を出力します。

■ チップ消去 / セクタ消去の場合

- チップ消去 / セクタ消去のアルゴリズム実行中に現在消去しているセクタをリードアクセスし、チップ消去実行中に指定したアドレスに関係なく、フラッシュメモリは "0" を出力します。チップ消去 / セクタ消去が終了すると、フラッシュメモリは "1" を出力します。

■ セクタ消去一時停止の場合

- セクタ消去一時停止状態でリードアクセスを行った場合、フラッシュメモリは、リードアドレスが消去中のセクタであれば、"1" を出力します。消去中のセクタでなければ、リードアドレスの読出し値の bit7 (DATA : 7) を出力します。
- トグルビットフラグ (DQ6) とともに参照することで、セクター一時停止状態であるか、どのセクタが消去中であるかを判定できます。

< 注意事項 >

自動アルゴリズム起動した場合は、指定したアドレスへのリードアクセスは無視されません。データの読出しは、データポーリングフラグ (DQ7) の終了を受けてほかのビットの出力が可能となります。自動アルゴリズム終了後のデータの読出しは、データポーリング終了を確認したリードアクセスの後に行ってください。

表 23.5-3 と表 23.5-4 に、データポーリングフラグの状態遷移を示します。

表 23.5-3 データポーリングフラグの状態遷移 (正常動作時の状態変化)

動作状態	書込み 書込み完了	チップ・ セクタ消去 消去完了	セクタ消去 ウェイト 消去開始	セクタ消去 セクタ消去 一時停止 (消去中の セクタ)	セクタ消去 一時停止 消去再開 (消去中の セクタ)	セクタ消去 一時停止中 (消去中では ないセクタ)
DQ7	$\overline{\text{DQ7}} \rightarrow$ DATA: 7	0→1	0	0→1	1→0	DATA: 7

表 23.5-4 データポーリングフラグの状態遷移 (異常動作時の状態変化)

動作状態	書込み	チップ・セクタ消去
DQ7	$\overline{\text{DQ7}}$	0

23.5.2 トグルビットフラグ (DQ6)

トグルビットフラグ (DQ6) は、自動アルゴリズム実行が進行中、もしくは終了状態であることをトグルビット機能によって知らせるハードウェアシーケンスフラグです。データポーリングフラグの状態遷移を表 23.5-5 と表 23.5-6 に示します。

■ 書込みとチップ消去 / セクタ消去の場合

- 自動書込みアルゴリズムおよびチップ消去 / セクタ消去のアルゴリズム実行中に、リードアクセスを連続して行った場合、フラッシュメモリは、指定したリードアドレスに関係なく、リードを行うごとに "1" と "0" を交互にトグル出力します。
- 自動書込みアルゴリズムおよびチップ消去 / セクタ消去のアルゴリズムが終了したあとに、リードアクセスを連続して行った場合、フラッシュメモリは bit6 のトグル動作を停止し、リードアドレスの読出し値の bit6(DATA: 6) を出力します。

■ セクタ消去一時停止の場合

セクタ消去一時停止状態でリードアクセスを行った場合、フラッシュメモリは、指定したリードアドレスが消去中のセクタであるならば、"1" を出力します。消去中のセクタでなければ、リードアドレスの読出し値の bit6(DATA: 6) を出力します。

< 参考 >

書込みを行うセクタが書換え保護されている場合は、約 2 μ s 間トグル出力を行った後、データを書き換えることなくトグル出力を終了します。

消去を行うすべてのセクタが書換え保護されている場合、トグルビットフラグ (DQ6) は、約 100 μ s の間トグル出力を行った後、データを書き換えることなく読出し / リセット状態に戻ります。

表 23.5-5 と表 23.5-6 に、トグルビットフラグの状態遷移を示します。

表 23.5-5 トグルビットフラグの状態遷移 (正常動作時の状態変化)

動作状態	書込み 書込み完了	チップ・ セクタ消去 消去完了	セクタ消去 ウェイト 消去開始	セクタ消去 セクタ消去 一時停止 (消去中の セクタ)	セクタ消去 一時停止 消去再開 (消去中の セクタ)	セクタ消去 一時停止中 (消去中では ないセクタ)
DQ6	Toggle→ DATA: 6	Toggle → Stop	Toggle	Toggle → 1	1 → Toggle	DATA: 6

表 23.5-6 トグルビットフラグの状態遷移 (異常動作時の状態変化)

動作状態	書込み	チップ・セクタ消去
DQ6	Toggle	Toggle

23.5.3 タイミングリミット超過フラグ (DQ5)

タイミングリミット超過フラグ (DQ5) は、自動アルゴリズムの実行がフラッシュメモリ内部に規定時間 (内部パルスカウントなど) を超えてしまったことを知らせるハードウェアシーケンスフラグです。タイミングリミット超過フラグの状態遷移を表 23.5-7 と表 23.5-8 に示します。

■ 書込みとチップ消去 / セクタ消去の場合

- 書込み、またはチップ消去 / セクタ消去の自動アルゴリズム起動後にリードアクセスを行った場合、規定時間 (書込み / 消去に要する時間) 内であれば "0" を、規定時間を超えている場合は "1" を出力します。
- タイミングリミット超過フラグ (DQ5) は、自動アルゴリズムが実行中か終了状態にかかわらず、書込み / 消去の成功または、失敗の判定を行うことができます。タイミングリミット超過フラグ (DQ5) が "1" を出力した場合、データポーリング機能、またはトグルビット機能により自動アルゴリズムが実行中であれば、書込みが失敗していると判断できます。
- 例えば "0" が書き込まれているフラッシュメモリアドレスを "1" に設定した場合は、フラッシュメモリはロックされ、自動アルゴリズムは終了せず、データポーリングフラグ (DQ7) から有効なデータが出力されません。またトグルビットフラグ (DQ6) はトグル動作を終了せず、タイムリミットを超えてしまい、タイミングリミット超過フラグ (DQ5) は "1" を出力します。タイミングリミット超過フラグ (DQ5) が "1" を出力した場合は、フラッシュメモリが不良ではなく、正しく使用されなかったことを示していますので、リセットコマンドを実行してください。

表 23.5-7 と表 23.5-8 に、タイミングリミット超過フラグの状態遷移を示します。

表 23.5-7 タイミングリミット超過フラグの状態遷移 (正常動作時の状態変化)

動作状態	書込み 書込み完了	チップ・ セクタ消去 消去完了	セクタ消去 ウェイト 消去開始	セクタ消去 セクタ消去 一時停止 (消去中の セクタ)	セクタ消去 一時停止 消去再開 (消去中の セクタ)	セクタ消去 一時停止中 (消去中では ないセクタ)
DQ5	0 → DATA: 5	0 → 1	0	0	0	DATA: 5

表 23.5-8 タイミングリミット超過フラグの状態遷移 (異常動作時の状態変化)

動作状態	書込み	チップ・セクタ消去
DQ5	1	1

23.5.4 セクタ消去タイマフラグ (DQ3)

セクタ消去タイマフラグ (DQ3) は、セクタ消去コマンド起動後セクタ消去ウェイト期間中であるか否かを知らせるハードウェアシーケンスフラグです。セクタ消去タイマフラグの状態遷移を表 23.5-9 と表 23.5-10 に示します。

■ セクタ消去の場合

- セクタ消去コマンド起動後にリードアクセスを行った場合に、セクタ消去ウェイト期間中であれば、コマンドを入力したセクタでのアドレス信号に指定されたリードアドレスに関係なく、“0” を出力します。セクタ消去ウェイト期間を超えている場合は “1” を出力します。
- データポーリング機能やトグルビット機能により消去アルゴリズムが実行中を示している場合、セクタ消去タイマフラグ (DQ3) が “1” であれば、内部制御消去動作を行っています。続けて消去一時停止以外のコマンドを設定した場合は、消去が終了されるまで無視されます。
- セクタ消去タイマフラグ (DQ3) が “0” であった場合、フラッシュメモリはセクタ消去コマンドを受け付けることができます。セクタ消去コマンドを書き込む場合は、セクタ消去タイマフラグ (DQ3) が “0” であることを確認してください。セクタ消去タイマ (DQ3) が “1” であった場合、一時停止のセクタ消去コマンドが受け付けられない場合があります。

● セクタ消去一時停止の場合

- セクタ消去一時停止状態でリードアクセスを行った場合、フラッシュメモリは、指定したリードアドレスが消去中のセクタであるならば、“1” を出力します。消去中のセクタでなければ、指定したリードアドレスの読出し値の bit3(DATA: 3) を出力します。

表 23.5-9 と表 23.5-10 に、セクタ消去タイマフラグの状態遷移を示します。

表 23.5-9 セクタ消去タイマフラグの状態遷移 (正常動作時の状態変化)

動作状態	書込み 書込み完了	チップ・ セクタ消去 消去完了	セクタ消去 ウェイト 消去開始	セクタ消去 セクタ消去 一時停止 (消去中の セクタ)	セクタ消去 一時停止 消去再開 (消去中の セクタ)	セクタ消去 一時停止中 (消去中では ないセクタ)
DQ3	0 → DATA: 3	1	0 → 1	1 → 0	0 → 1	DATA: 3

表 23.5-10 セクタ消去タイマフラグの状態遷移 (異常動作時の状態変化)

動作状態	書込み	チップ・セクタ消去
DQ3	0	1

23.6 フラッシュメモリ書込み / 消去の詳細説明

自動アルゴリズムを起動するコマンドを入力し、フラッシュメモリに読出し / リセット、書込み、チップ消去、セクタ消去、セクタ消去一時停止、セクタ消去再開のそれぞれの動作を行う手順を説明します。

■ フラッシュメモリ書込み / 消去の詳細説明

自動アルゴリズムは、読出し / リセット、書込み、チップ消去、セクタ消去、セクタ消去一時停止、消去再開のコマンドシーケンス（詳細は、「23.4 フラッシュメモリ自動アルゴリズム起動方法」を参考にしてください）を CPU からフラッシュメモリに書き込むことにより起動できます。CPU からフラッシュメモリへの書込みは、必ず連続して行ってください。また、自動アルゴリズムはデータポーリング機能により終了状態を確認できます。正常終了後は読出し / リセット状態に戻ります。

各動作について、下記の順に説明します。

- フラッシュメモリを読出し / リセット状態にする
- フラッシュメモリへデータを書き込む
- フラッシュメモリのデータを消去する（チップ消去）
- フラッシュメモリの任意のデータを消去する（セクタ消去）
- フラッシュメモリのセクタ消去を一時停止する
- フラッシュメモリのセクタ消去を再開する

23.6.1 フラッシュメモリを讀出し/リセット状態にする

讀出し/リセットコマンドを入力し、フラッシュメモリを讀出し/リセット状態にする手順について説明します。

■ フラッシュメモリを讀出し/リセット状態にする

- フラッシュメモリを讀出し/リセット状態にするには、コマンドシーケンス表（詳細は、「23.4 フラッシュメモリ自動アルゴリズム起動方法」を参考にしてください）の讀出し/リセットコマンドを CPU からフラッシュメモリの目的のセクタへ連続してコードを送ってください。
- 讀出し/リセットコマンドには 1 回と 3 回のバス動作を行う 2 通りのコマンドシーケンスがありますが、違いはありません。
- 讀出し/リセット状態はフラッシュメモリの初期状態ですので、電源投入後、コマンドの正常終了後は常に讀出し/リセット状態になります。讀出し/リセット状態は、コマンドの入力待ち状態でもあります。
- 讀出し/リセット状態では、フラッシュメモリへ、リードアクセスコマンドを行うことによりデータを読み出すことができます。マスク ROM と同様に CPU からのプログラムアクセスができます。フラッシュメモリへ、リードアクセスを行う場合は、讀出し/リセットコマンドは必要ありません。コマンドが正常に終了しなかった場合は、自動アルゴリズムを初期化するために、讀出し/リセットコマンドを使用してください。

23.6.2 フラッシュメモリヘデータを書き込む

書込みコマンドを入力し、フラッシュメモリヘデータを書き込む手順について説明します。図 23.6-1 にフラッシュメモリ書込み手順の例を示します。

■ フラッシュメモリヘデータを書き込む

- フラッシュメモリのデータ書込み自動アルゴリズムを起動するためには、コマンドシーケンス表（詳細は、「23.4 フラッシュメモリ自動アルゴリズム起動方法」を参考にしてください）の書込みコマンドを CPU からフラッシュメモリの目的のセクタへ連続して送ってください。
- 4 サイクル目に目的のアドレスへのデータ書込みが終了した場合、自動アルゴリズムが起動され自動書込みを開始します。

■ アドレス指定方法

- 書込みデータサイクルで指定する書込みアドレスは、偶数アドレスのみ設定できます。奇数アドレスを設定した場合は、正しく書き込むことができません。偶数アドレスへワードデータ単位で書込みを行ってください。
- 書込みはどのようなアドレスの順番でも、セクタの境界を超えても行えます。1 回の書込みコマンドによって書き込まれるデータは 1 ワードのみです。

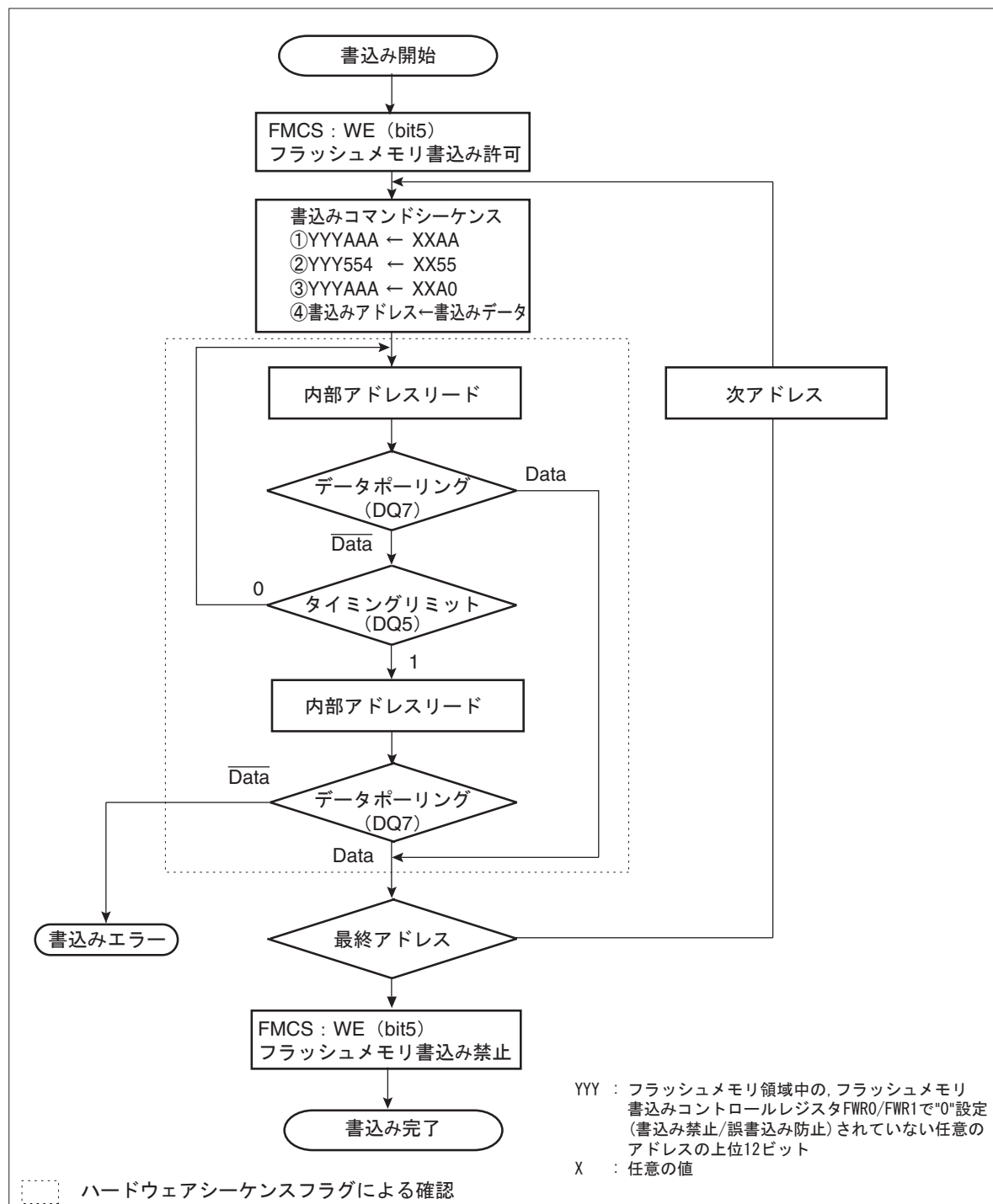
■ データ書込み上の注意

- 書込みによって、ビットデータを "0" から "1" に戻すことはできません。ビットデータ "0" にビットデータ "1" を書き込むと、データポーリングアルゴリズム (DQ7) または、トグル動作 (DQ6) が終了せず、フラッシュメモリ素子が不良と判定され、書込み規定時間を超えてタイミングリミット超過フラグ (DQ5) がエラーと判定します。"1" の書込みは可能ですが、実際には行いません。
- 読出し / リセット状態でデータを読み出した場合、ビットデータは "0" のままです。ビットデータを "0" から "1" に戻すには、フラッシュメモリの消去を行ってください。
- 自動書込み実行中は、すべてのコマンドが無視されます。書込み中にハードウェアリセットが起動された場合は、書込みアドレスのデータは保証されません。

■ フラッシュメモリ書込み手順

- 図 23.6-1 にフラッシュメモリ書込み手順の例を示します。ハードウェアシーケンスフラグ（「23.5 自動アルゴリズム実行状態の確認」を参考にしてください）を用いることでフラッシュメモリ内部の自動アルゴリズムの状態を判定できます。ここでは、フラッシュメモリへの書込み終了の確認にデータポーリングフラグ (DQ7) を用いています。
- フラグチェックのために読み込むデータは、最後に書込みを行ったアドレスからの読込みとなります。
- データポーリングフラグ (DQ7) は、タイミングリミット超過フラグ (DQ5) と同時に変化するので、タイミングリミット超過フラグ (DQ5) が "1" であった場合でもデータポーリングフラグビット (DQ7) を確認してください。
- トグルビットフラグ (DQ6) も、タイミングリミット超過フラグビット (DQ5) が "1" に変化すると同時にトグル動作を止めますので、トグルビットフラグ (DQ6) を確認してください。

図 23.6-1 フラッシュメモリ書き込み手順の例



23.6.3 フラッシュメモリのデータを消去する (チップ消去)

チップ消去コマンドを入力し、フラッシュメモリの全データを消去する手順について説明します。

■ フラッシュメモリのデータを消去する (チップ消去)

- フラッシュメモリからすべてのデータを消去するためには、コマンドシーケンス表 (「23.4 フラッシュメモリ自動アルゴリズム起動方法」を参照) のチップ消去コマンドを CPU からフラッシュメモリの目的のセクタへ連続して送ってください。
- チップ消去コマンドは 6 回のバス動作で行われ、6 サイクル目の書込みが完了した時点でチップ消去動作を開始します。
- チップ消去では、消去前にユーザがフラッシュメモリに書込みを行う必要はありません。自動消去アルゴリズム実行中に、フラッシュメモリはすべてのセルを消去する前に "0" を自動的に書き込んでから消去します。

23.6.4 フラッシュメモリの任意のデータを消去する (セクタ消去)

セクタ消去コマンドを入力し、フラッシュメモリの任意のセクタを消去する (セクタ消去) 手順について説明します。セクタごとの消去が可能であり、複数のセクタを同時に指定することもできます。

■ フラッシュメモリの任意のデータを消去する (セクタ消去)

フラッシュメモリの任意のセクタを消去するためには、コマンドシーケンス表 (「23.4 フラッシュメモリ自動アルゴリズム起動方法」を参照) のセクタ消去コマンドを CPU からフラッシュメモリの目的のセクタへ連続して送ってください。

● セクタ指定方法

- セクタ消去コマンドは 6 回のバス動作で行われます。6 サイクル目のアドレスを目的のセクタ内の偶数アドレスに設定し、データにセクタ消去コード (30_H) を書き込むことにより 50 μ s のセクタ消去ウェイトが開始します。
- 複数のセクタ消去を行う場合は、上記に続き消去する目的のセクタ内のアドレスに消去コード (30_H) を書き込みます。

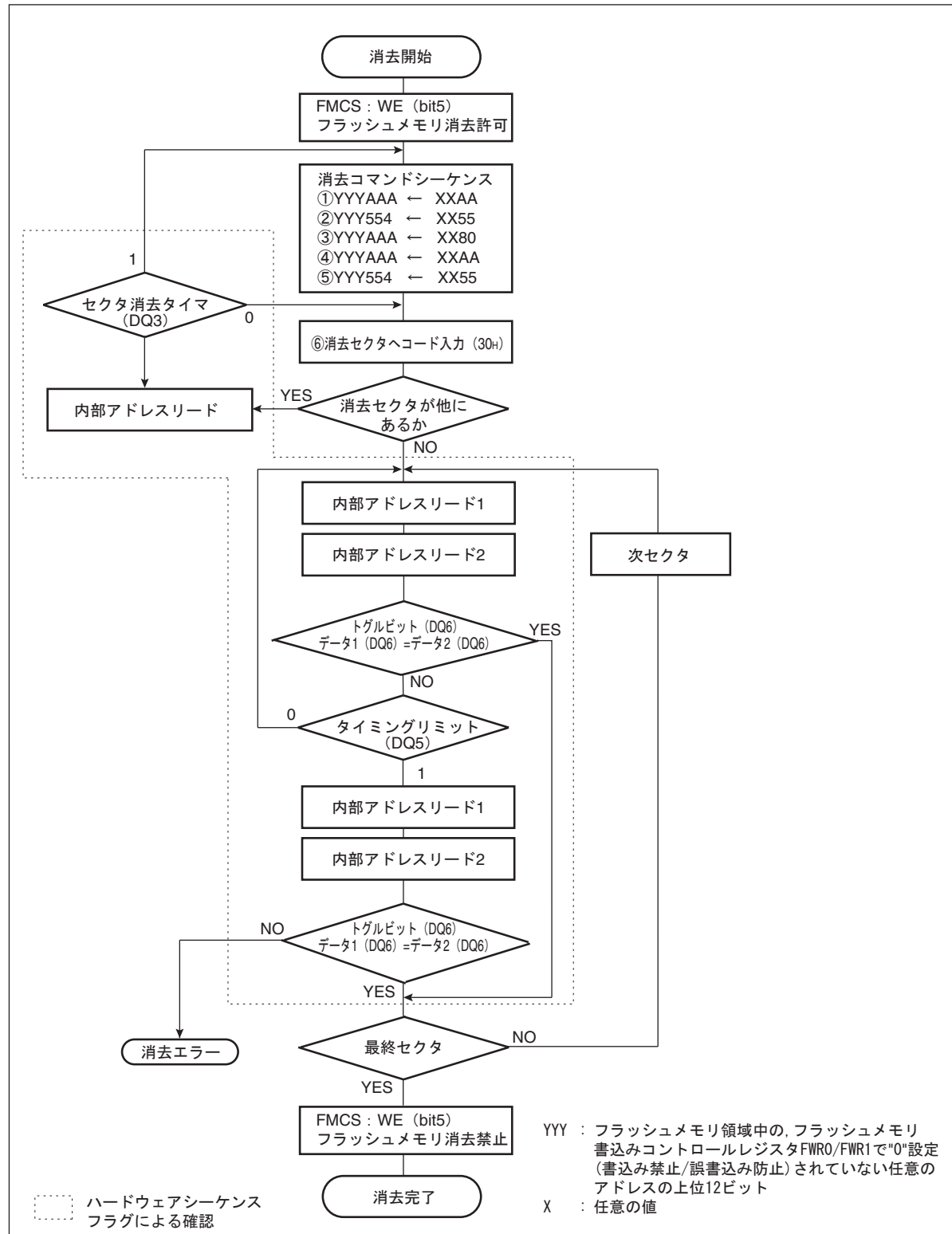
■ 複数のセクタを指定する場合の注意

- 最後のセクタ消去コードの書込みから 50 μ s のセクタ消去ウェイト期間終了により消去が開始します。
- 複数のセクタを同時に消去する場合は、消去セクタのアドレスと消去コード (コマンドシーケンス 6 サイクル目) を 50 μ s 以内に入力してください。50 μ s を超えて消去コードを入力した場合は、受け付けられません。
- 連続したセクタ消去コードの書込みが有効であるかは、セクタ消去タイマフラグ (DQ3) によって確認できます。
- セクタ消去タイマフラグ (DQ3) をリードする場合のアドレスは、消去しようとしているセクタを指定してください。

■ フラッシュメモリのセクタ消去手順

- ハードウェアシーケンスフラグ (「23.5 自動アルゴリズム実行状態の確認」を参照) を用いることでフラッシュメモリ内部の自動アルゴリズムの状態を判定できます。図 23.6-2 にフラッシュメモリセクタの消去手順の例を示します。ここでは、消去終了の確認にトグルビットフラグ (DQ6) を用いています。フラグを確認するデータは、消去しようとしているセクタから読み込むことに注意してください。
- トグルビットフラグ (DQ6) は、タイミングリミット超過フラグ (DQ5) が "1" に変化するのと同時にトグル動作を終了します。タイミングリミット超過フラグ (DQ5) が "1" の場合でも、トグルビットフラグ (DQ6) を確認してください。
- データポーリングフラグ (DQ7) は、タイミングリミット超過フラグ (DQ5) と同時に変化するので、データポーリングフラグ (DQ7) を確認してください。

図 23.6-2 フラッシュメモリのセクタ消去手順の例



23.6.5 フラッシュメモリのセクタ消去を一時停止する

セクタ消去一時停止コマンドを入力し、フラッシュメモリセクタ消去の一時停止を行う手順について説明します。消去中でないセクタから、データを読み出すことが可能です。

■ フラッシュメモリのセクタ消去を一時停止する

- フラッシュメモリのセクタ消去を一時停止するためには、コマンドシーケンス表（「23.4 フラッシュメモリ自動アルゴリズム起動方法」を参照）のセクタ消去一時停止コマンドを CPU からフラッシュメモリの目的のセクタへ送ってください。
- セクタ消去一時停止コマンドは、セクタ消去中に消去を一時停止し、消去中でないセクタからデータを読み出すことができます。セクタ消去一時停止状態では、読出しのみが可能で書込みはできません。
- セクタ消去一時停止コマンドは、消去ウェイト時間を含むセクタ消去中のみ有効で、チップ消去中や書込み中は無視されます。
- セクタ消去一時停止コマンドは、消去一時停止コード (B0_H) を書き込むことで実行されます。このときのアドレスは、フラッシュメモリ内の任意のアドレスを設定してください。消去一時停止中に再度、消去一時停止コマンドを実行した場合、再度入力したコマンドは無視されます。
- セクタ消去ウェイト期間中に、セクタ消去一時停止コマンドを入力した場合、直ちにセクタ消去ウェイトを終了し、消去動作を中断して消去停止状態になります。
- セクタ消去ウェイト期間後のセクタ消去中に消去一時停止コマンドを入力した場合、最大 20 μ s 後に、消去一時停止状態になります。セクタ消去一時停止コマンドは、セクタ消去コマンドあるいはセクタ消去再開コマンド発行後、20 μ s 以上後に行ってください。

■ 注意

一時停止コマンドを発行する場合は、セクタ消去コマンド発行後 20 ms 以上経過した後発行するか、あるいは、セクタ消去再開コマンドの発行後 20 ms 以上経過した後発行してください。または、発行回数は数回にとどめてください。

23.6.6 フラッシュメモリのセクタ消去を再開する

セクタ消去再開コマンドを入力し、一時停止したフラッシュメモリセクタ消去を再開する手順について説明します。

■ フラッシュメモリのセクタ消去を再開する

- 一時停止したセクタ消去を再開させるためには、コマンドシーケンス表（「23.4 フラッシュメモリ自動アルゴリズム起動方法」を参照）のセクタ消去再開コマンドを CPU からフラッシュメモリの目的のセクタへ送ってください。
- セクタ消去再開コマンドは、セクタ消去一時停止コマンドによるセクタ消去一時停止状態から、セクタ消去を再開するためのコマンドです。セクタ消去再開コマンドは消去再開コード (30_H) の書込みを行うことで実行されますが、アドレスはフラッシュメモリ内の任意のアドレスを指定します。
- セクタ消去中のセクタ消去再開コマンドの入力は無視されます。

23.7 フラッシュセキュリティの特長

フラッシュセキュリティコントローラ機能により、フラッシュメモリの内容を外部端子から読み出されることを防止できます。

- 事前定義されたフラッシュメモリのアドレスが、フラッシュセキュリティコントローラ (MB90F822B: FF0001_H; MB90F823B: FE0001_H) に割り当てられます。このアドレスに保護コード "01_H" が書き込まれるとフラッシュメモリへのアクセスが制限されます。フラッシュメモリが一度保護されると、チップ消去を行うまで、機能のロックを解除することはできません。ロックを解除しない限り、いずれの外部端子からもフラッシュメモリへの読み出し / 書き込みはできません。
- この機能は、フラッシュメモリに格納される自己完結型プログラムやデータのセキュリティを必要とするアプリケーションに適しています。マイクロコントローラの外部に位置するプログラム部分を対象アプリケーションが必要とする場合には、フラッシュセキュリティコントローラは、目的の機能を提供できません。このため、保護コードを設定した場合は、外部ベクタフェッチモードは使用しません。
- 標準パラレルプログラマによるフラッシュマイクロコントローラのプログラミングには、独特のセットアップが必要になることがあります。例えば、ミナトエレクトロニクスプログラマでは、デバイスチェックを切断する必要があります。保護コードは、フラッシュプログラミングの終わりにコーディングすることを推奨します。これは、プログラミング中の不要な保護を回避するためです。
- 一度保護されたフラッシュメモリを再度プログラムするには、チップ消去操作を行う必要があります。

詳細については、営業部門担当者にお問い合わせください。

付録 A, B, C では , I/O マップ , MB90F822B/F823B
シリアル書込み接続例および命令について示しま
す。

付録 A I/O マップ

付録 B シリアル書込み例

付録 C 命令

付録 A I/O マップ

MB90820B シリーズでは、周辺機能のレジスタに割当てられているアドレスを表 A-1 に示します。

■ I/O マップ

表 A-1 I/O マップ (1 / 7)

アドレス	略称	レジスタ	バイト アクセス	ワード アクセス	リソース名	初期値
000000 _H	PDR0	ポート 0 データレジスタ	R/W	R/W	ポート 0	XXXXXXXX _B
000001 _H	PDR1	ポート 1 データレジスタ	R/W	R/W	ポート 1	XXXXXXXX _B
000002 _H	PDR2	ポート 2 データレジスタ	R/W	R/W	ポート 2	XXXXXXXX _B
000003 _H	PDR3	ポート 3 データレジスタ	R/W	R/W	ポート 3	XXXXXXXX _B
000004 _H	PDR4	ポート 4 データレジスタ	R/W	R/W	ポート 4	XXXXXXXX _B
000005 _H	PDR5	ポート 5 データレジスタ	R/W	R/W	ポート 5	XXXXXXXX _B
000006 _H	PDR6	ポート 6 データレジスタ	R/W	R/W	ポート 6	XXXXXXXX _B
000007 _H	PDR7	ポート 7 データレジスタ	R/W	R/W	ポート 7	XXXXXXXX _B
000008 _H	PDR8	ポート 8 データレジスタ	R/W	R/W	ポート 8	XXXXXXXX _B
000009 _H ~ 00000F _H	使用禁止					
000010 _H	DDR0	ポート 0 データ方向レジスタ	R/W	R/W	ポート 0	00000000 _B
000011 _H	DDR1	ポート 1 データ方向レジスタ	R/W	R/W	ポート 1	00000000 _B
000012 _H	DDR2	ポート 2 データ方向レジスタ	R/W	R/W	ポート 2	00000000 _B
000013 _H	DDR3	ポート 3 データ方向レジスタ	R/W	R/W	ポート 3	00000000 _B
000014 _H	DDR4	ポート 4 データ方向レジスタ	R/W	R/W	ポート 4	00000000 _B
000015 _H	DDR5	ポート 5 データ方向レジスタ	R/W	R/W	ポート 5	XXXXXXXX00 _B
000016 _H	DDR6	ポート 6 データ方向レジスタ	R/W	R/W	ポート 6	00000000 _B
000017 _H	DDR7	ポート 7 データ方向レジスタ	R/W	R/W	ポート 7	00000000 _B
000018 _H	DDR8	ポート 8 データ方向レジスタ	R/W	R/W	ポート 8	00000000 _B
000019 _H ~ 00001F _H	使用禁止					
000020 _H	SMR0	シリアルモードレジスタ 0	R/W	R/W	UART0	00000000 _B
000021 _H	SCR0	シリアル制御レジスタ 0	W,R/W	W,R/W		00000100 _B
000022 _H	SIDR0 / SODR0	シリアル入力データレジスタ 0 / シリアル出力データレジスタ 0	R/W	R/W		XXXXXXXX _B
000023 _H	SSR0	シリアルステータスレジスタ 0	W,R/W	W,R/W		00001000 _B

表 A-1 I/O マップ (2 / 7)

アドレス	略称	レジスタ	バイト アクセス	ワード アクセス	リソース名	初期値
000024 _H	SMR1	シリアルモードレジスタ 1	R/W	R/W	UART 1	00000000 _B
000025 _H	SCR1	シリアル制御レジスタ 1	W,R/W	W,R/W		00000100 _B
000026 _H	SIDR1 / SODR1	シリアル入力データレジスタ 1 / シリアル出力データレジスタ 1	R/W	R/W		XXXXXXXX _B
000027 _H	SSR1	ステータスレジスタ 1	R,R/W	R,R/W		00001000 _B
000028 _H	PWCSL1	PWC 制御ステータスレジスタ 1	R/W	R/W	PWC タイマ 1	00000000 _B
000029 _H	PWCSH1		R,R/W	R,R/W		00000000 _B
00002A _H	PWC1	PWC データバッファレジスタ 1	-	R/W		XXXXXXXX _B
00002B _H						XXXXXXXX _B
00002C _H	DIV1	分周比制御レジスタ 1	R/W	R/W		XXXXXX00 _B
00002D _H , 00002E _H	使用禁止					
00002F _H	PCKCR	PLL クロック制御レジスタ	W	W	PLL	XXXX0000 _B
000030 _H	ENIR	DTP / 割込み許可レジスタ	R/W	R/W	DTP/ 外部割込み 0 ~ 7	00000000 _B
000031 _H	EIRR	DTP / 割込み要因レジスタ	R/W	R/W		XXXXXXXX _B
000032 _H	ELVRL	要求レベル設定レジスタ (下位)	R/W	R/W		00000000 _B
000033 _H	ELVRH	要求レベル設定レジスタ (上位)	R/W	R/W		00000000 _B
000034 _H	使用禁止					
000035 _H	CDCR0	クロック分周制御レジスタ 0	R/W	R/W	通信プリスケラ 0	00XXX000 _B
000036 _H	使用禁止					
000037 _H	CDCR1	クロック分周制御レジスタ 1	R/W	R/W	通信プリスケラ 1	00XXX000 _B
000038 _H	PDCR0	PPG ダウンカウンタレジスタ 0	-	R	16 ビット PPG タイマ 0	11111111 _B
000039 _H						11111111 _B
00003A _H	PCSR0	PPG 周期設定レジスタ 0	-	W		XXXXXXXX _B
00003B _H						XXXXXXXX _B
00003C _H	PDUT0	PPG デューティ設定レジスタ 0	-	W		XXXXXXXX _B
00003D _H						XXXXXXXX _B
00003E _H	PCNTL0	PPG 制御ステータスレジスタ 0	R/W	R/W		XX000000 _B
00003F _H	PCNTH0		R/W	R/W		00000000 _B
000040 _H	PDCR1	PPG ダウンカウンタレジスタ 1	-	R	16 ビット PPG タイマ 1	11111111 _B
000041 _H						11111111 _B
000042 _H	PCSR1	PPG 周期設定レジスタ 1	-	W		XXXXXXXX _B
000043 _H						XXXXXXXX _B
000044 _H	PDUT1	PPG デューティ設定レジスタ 1	-	W		XXXXXXXX _B
000045 _H						XXXXXXXX _B
000046 _H	PCNTL1	PPG 制御ステータスレジスタ 1	R/W	R/W		XX000000 _B
000047 _H	PCNTH1		R/W	R/W		00000000 _B

表 A-1 I/O マップ (3 / 7)

アドレス	略称	レジスタ	バイト アクセス	ワード アクセス	リソース名	初期値
000048 _H	PDCR2	PPG ダウンカウンタレジスタ 2	-	R	16 ビット PPG タイマ 2	11111111 _B
000049 _H						11111111 _B
00004A _H	PCSR2	PPG 周期設定レジスタ 2	-	W		XXXXXXXX _B
00004B _H						XXXXXXXX _B
00004C _H	PDUT2	PPG デューティ設定レジスタ 2	-	W		XXXXXXXX _B
00004D _H						XXXXXXXX _B
00004E _H	PCNTL2	PPG 状態制御レジスタ 2	R/W	R/W		XX000000 _B
00004F _H	PCNTH2		R/W	R/W		00000000 _B
000050 _H	TMRR0	16 ビットタイマレジスタ 0	-	R/W	波形ジェネレータ	XXXXXXXX _B
000051 _H						XXXXXXXX _B
000052 _H	TMRR1	16 ビットタイマレジスタ 1	-	R/W		XXXXXXXX _B
000053 _H						XXXXXXXX _B
000054 _H	TMRR2	16 ビットタイマレジスタ 2	-	R/W		XXXXXXXX _B
000055 _H						XXXXXXXX _B
000056 _H	DTCR0	16 ビットタイマ制御レジスタ ch0	R/W	R/W		00000000 _B
000057 _H	DTCR1	16 ビットタイマ制御レジスタ ch1	R/W	R/W		00000000 _B
000058 _H	DTCR2	16 ビットタイマ制御レジスタ ch2	R/W	R/W		00000000 _B
000059 _H	SIGCR	波形制御レジスタ	R/W	R/W		00000000 _B
00005A _H	CPCLRB / CPCLR	コンペアクリアバッファレジスタ / コンペアクリアレジスタ (下位)	-	R/W	16 ビットフリーラン タイマ	11111111 _B
00005B _H						11111111 _B
00005C _H	TCDT	タイマデータレジスタ (下位)	-	R/W		00000000 _B
00005D _H						00000000 _B
00005E _H	TCCSL	タイマ制御ステータスレジスタ (下位)	R/W	R/W	16 ビットフリーラン タイマ	X0000000 _B
00005F _H	TCCSH	タイマ制御ステータスレジスタ (上位)	R/W	R/W		00000000 _B

表 A-1 I/O マップ (4 / 7)

アドレス	略称	レジスタ	バイト アクセス	ワード アクセス	リソース名	初期値
000060 _H	IPCP0	インプットキャプチャレジスタ 0	-	R	16 ビットインプット キャプチャ 0 ~ 3	XXXXXXXXX _B
000061 _H						XXXXXXXXX _B
000062 _H	IPCP1	インプットキャプチャレジスタ 1	-	R		XXXXXXXXX _B
000063 _H						XXXXXXXXX _B
000064 _H	IPCP2	インプットキャプチャレジスタ 2	-	R		XXXXXXXXX _B
000065 _H						XXXXXXXXX _B
000066 _H	IPCP3	インプットキャプチャレジスタ 3	-	R		XXXXXXXXX _B
000067 _H						XXXXXXXXX _B
000068 _H	PICSL01	インプットキャプチャコントロール ステータスレジスタ 0, 1(下位)	R/W	R/W		00000000 _B
000069 _H	PICSH01	PPG 出力制御 / インプットキャプ チャコントロールステータスレジス タ 0, 1(上位)	R/W	R/W		00000000 _B
00006A _H	ICSL23	インプットキャプチャコントロール ステータスレジスタ 2, 3 (下位)	R/W	R/W	00000000 _B	
00006B _H	ICSH23	インプットキャプチャコントロール ステータスレジスタ 2, 3 (上位)	R	R	XXXXXX00 _B	
00006C _H ~ 00006E _H	使用禁止					
00006F _H	ROMM	ROM ミラー機能選択レジスタ	W	W	ROM ミラー機能	XXXXXXXX1 _B
000070 _H	OCCPB0 / OCCP0	アウトプットコンペアバッファ レジスタ / アウトプットコンペア レジスタ 0	-	R/W	アウトプット コンペア 0 ~ 5	XXXXXXXXX _B
000071 _H						XXXXXXXXX _B
000072 _H	OCCPB1 / OCCP1	アウトプットコンペアバッファ レジスタ / アウトプットコンペア レジスタ 1	-	R/W		XXXXXXXXX _B
000073 _H						XXXXXXXXX _B
000074 _H	OCCPB2 / OCCP2	アウトプットコンペアバッファ レジスタ / アウトプットコンペア レジスタ 2	-	R/W		XXXXXXXXX _B
000075 _H						XXXXXXXXX _B
000076 _H	OCCPB3 / OCCP3	アウトプットコンペアバッファ レジスタ / アウトプットコンペア レジスタ 3	-	R/W		XXXXXXXXX _B
000077 _H						XXXXXXXXX _B
000078 _H	OCCPB4 / OCCP4	アウトプットコンペアバッファ レジスタ / アウトプットコンペア レジスタ 4	-	R/W		XXXXXXXXX _B
000079 _H						XXXXXXXXX _B
00007A _H	OCCPB5 / OCCP5	アウトプットコンペアバッファ レジスタ / アウトプットコンペア レジスタ 5	-	R/W		XXXXXXXXX _B
00007B _H						XXXXXXXXX _B
00007C _H	OCS0	コンペア制御レジスタ 0	R/W	R/W		00001100 _B
00007D _H	OCS1	コンペア制御レジスタ 1	R/W	R/W		X1100000 _B
00007E _H	OCS2	コンペア制御レジスタ 2	R/W	R/W		00001100 _B
00007F _H	OCS3	コンペア制御レジスタ 3	R/W	R/W		X1100000 _B
000080 _H	OCS4	コンペア制御レジスタ 4	R/W	R/W		00001100 _B
000081 _H	OCS5	コンペア制御レジスタ 5	R/W	R/W		X1100000 _B

表 A-1 I/O マップ (5 / 7)

アドレス	略称	レジスタ	バイト アクセス	ワード アクセス	リソース名	初期値
000082 _H	TMCSRL0	タイマ制御ステータスレジスタ 0 (下位)	R/W	R/W	16 ビットリロード タイマ 0	00000000 _B
000083 _H	TMCSRH0	タイマ制御ステータスレジスタ 0 (上位)	R/W	R/W		XXX10000 _B
000084 _H	TMR0 / TMRD0	16 ビットタイマレジスタ 0 /	-	R/W		XXXXXXXX _B
000085 _H		16 ビットリロードレジスタ 0				XXXXXXXX _B
000086 _H	TMCSRL1	タイマ制御ステータスレジスタ 1 (下位)	R/W	R/W	16 ビットリロード タイマ 1	00000000 _B
000087 _H	TMCSRH1	タイマ制御ステータスレジスタ 1 (上位)	R/W	R/W		XXX10000 _B
000088 _H	TMR1 / TMRD1	16 ビットタイマレジスタ 1 /	-	R/W		XXXXXXXX _B
000089 _H		16 ビットリロードレジスタ 1				XXXXXXXX _B
00008A _H	CSVCR	クロックスーパーバイザ 制御レジスタ *	R, R/W	-	クロックスーパーバイザ	00011100 _B
00008B _H	使用禁止					
00008C _H	RDR0	ポート 0 プルアップ抵抗設定 レジスタ	R/W	R/W	ポート 0	00000000 _B
00008D _H	RDR1	ポート 1 プルアップ抵抗設定 レジスタ	R/W	R/W	ポート 1	00000000 _B
00008E _H	RDR2	ポート 2 プルアップ抵抗設定 レジスタ	R/W	R/W	ポート 2	00000000 _B
00008F _H	RDR3	ポート 3 プルアップ抵抗設定 レジスタ	R/W	R/W	ポート 3	00000000 _B
000090 _H ~ 00009D _H	使用禁止					
00009E _H	PACSR	プログラムアドレス検出制御 ステータスレジスタ	R/W	R/W	アドレス一致検出	00000000 _B
00009F _H	DIRR	遅延割込み要求発生 / 解除レジスタ	R/W	R/W	遅延割込み	XXXXXXXX0 _B
0000A0 _H	LPMCR	低消費電力モード制御レジスタ	W,R/W	W,R/W	低消費電力制御 レジスタ	00011000 _B
0000A1 _H	CKSCR	クロック選択レジスタ	R,R/W	R,R/W		11111100 _B
0000A2 _H ~ 0000A7 _H	使用禁止					
0000A8 _H	WDTC	ウォッチドッグタイマ制御レジスタ	R,R/W	R,R/W	ウォッチドッグタイマ	XXXXX111 _B
0000A9 _H	TBTC	タイムベースタイマ制御レジスタ	W,R/W	W,R/W	タイムベースタイマ	1XX00100 _B
0000AA _H ~ 0000AD _H	使用禁止					
0000AE _H	FMCS	フラッシュメモリ制御ステータス レジスタ	R,R/W	R,R/W	フラッシュメモリ インタフェース	000X0000 _B
0000AF _H	使用禁止					

表 A-1 I/O マップ (6 / 7)

アドレス	略称	レジスタ	バイト アクセス	ワード アクセス	リソース名	初期値
0000B0 _H	ICR00	割込み制御レジスタ 00	R/W	R/W	割込み制御	00000111 _B
0000B1 _H	ICR01	割込み制御レジスタ 01	R/W	R/W		00000111 _B
0000B2 _H	ICR02	割込み制御レジスタ 02	R/W	R/W		00000111 _B
0000B3 _H	ICR03	割込み制御レジスタ 03	R/W	R/W		00000111 _B
0000B4 _H	ICR04	割込み制御レジスタ 04	R/W	R/W		00000111 _B
0000B5 _H	ICR05	割込み制御レジスタ 05	R/W	R/W		00000111 _B
0000B6 _H	ICR06	割込み制御レジスタ 06	R/W	R/W		00000111 _B
0000B7 _H	ICR07	割込み制御レジスタ 07	R/W	R/W		00000111 _B
0000B8 _H	ICR08	割込み制御レジスタ 08	R/W	R/W		00000111 _B
0000B9 _H	ICR09	割込み制御レジスタ 09	R/W	R/W		00000111 _B
0000BA _H	ICR10	割込み制御レジスタ 10	R/W	R/W		00000111 _B
0000BB _H	ICR11	割込み制御レジスタ 11	R/W	R/W		00000111 _B
0000BC _H	ICR12	割込み制御レジスタ 12	R/W	R/W		00000111 _B
0000BD _H	ICR13	割込み制御レジスタ 13	R/W	R/W		00000111 _B
0000BE _H	ICR14	割込み制御レジスタ 14	R/W	R/W		00000111 _B
0000BF _H	ICR15	割込み制御レジスタ 15	R/W	R/W		00000111 _B
0000C0 _H	PWCSL0	PWC 状態制御レジスタ 0	R/W	R/W	PWC タイマ 0	00000000 _B
0000C1 _H	PWCSH0		R,R/W	R,R/W		00000000 _B
0000C2 _H	PWC0	PWC データバッファレジスタ 0	-	R/W		XXXXXXXX _B
0000C3 _H						XXXXXXXX _B
0000C4 _H	DIV0	分周比制御レジスタ 0	R/W	R/W		XXXXXXXX00 _B
0000C5 _H	ADER0	A/D 入力許可レジスタ 0	R/W	R/W	ポート 6, A/D	11111111 _B
0000C6 _H	ADCS0	A/D 制御ステータスレジスタ 0	R/W	R/W	8/10 ビット A/D コンバータ	000XXXX0 _B
0000C7 _H	ADCS1	A/D 制御ステータスレジスタ 1	W,R/W	W,R/W		0000000X _B
0000C8 _H	ADCR0	A/D データレジスタ 0	R	R		00000000 _B
0000C9 _H	ADCR1	A/D データレジスタ 1	R/W	R/W		XXXXXX00 _B
0000CA _H	ADSR0	A/D セットアップレジスタ 0	R/W	R/W		00000000 _B
0000CB _H	ADSR1	A/D セットアップレジスタ 1	R/W	R/W		00000000 _B
0000CC _H	DAT0	D/A データレジスタ 0	R/W	R/W	8 ビット D/A コンバータ	XXXXXXXX _B
0000CD _H	DAT1	D/A データレジスタ 1	R/W	R/W		XXXXXXXX _B
0000CE _H	DACR0	D/A 制御レジスタ 0	R/W	R/W		XXXXXXXX0 _B
0000CF _H	DACR1	D/A 制御レジスタ 1	R/W	R/W		XXXXXXXX0 _B
0000D0 _H	ADER1	A/D 入力許可レジス 1	R/W	R/W	ポート 7, A/D	11111111 _B
0000D1 _H ~ 0000EF _H	使用禁止					
0000F0 _H ~ 0000FF _H	外部領域					

表 A-1 I/O マップ (7 / 7)

アドレス	略称	レジスタ	バイト アクセス	ワード アクセス	リソース名	初期値
001FF0 _H	PADRL0	プログラムアドレス検出レジスタ 0 (下位)	R/W	R/W	アドレス一致検出	XXXXXXXX _B
001FF1 _H	PADRM0	プログラムアドレス検出レジスタ 0 (中位)	R/W	R/W		XXXXXXXX _B
001FF2 _H	PADRH0	プログラムアドレス検出レジスタ 0 (上位)	R/W	R/W		XXXXXXXX _B
001FF3 _H	PADRL1	プログラムアドレス検出レジスタ 1 (下位)	R/W	R/W		XXXXXXXX _B
001FF4 _H	PADRM1	プログラムアドレス検出レジスタ 1 (中位)	R/W	R/W		XXXXXXXX _B
001FF5 _H	PADRH1	プログラムアドレス検出レジスタ 1 (上位)	R/W	R/W		XXXXXXXX _B

* : MB90F828B のみ。他品種では使用禁止。

● 書込み / 読出しについての説明

R/W : リード / ライト可能

R : リードオンリ

W : ライトオンリ

● 初期値についての説明

0: 初期値は "0" です。

1: 初期値は "1" です。

X: 初期値は不定です。

付録 B シリアル書込み例

付録 B では、F²MC-16LX MB90F822B/F823B シリアル書込み例について説明します。

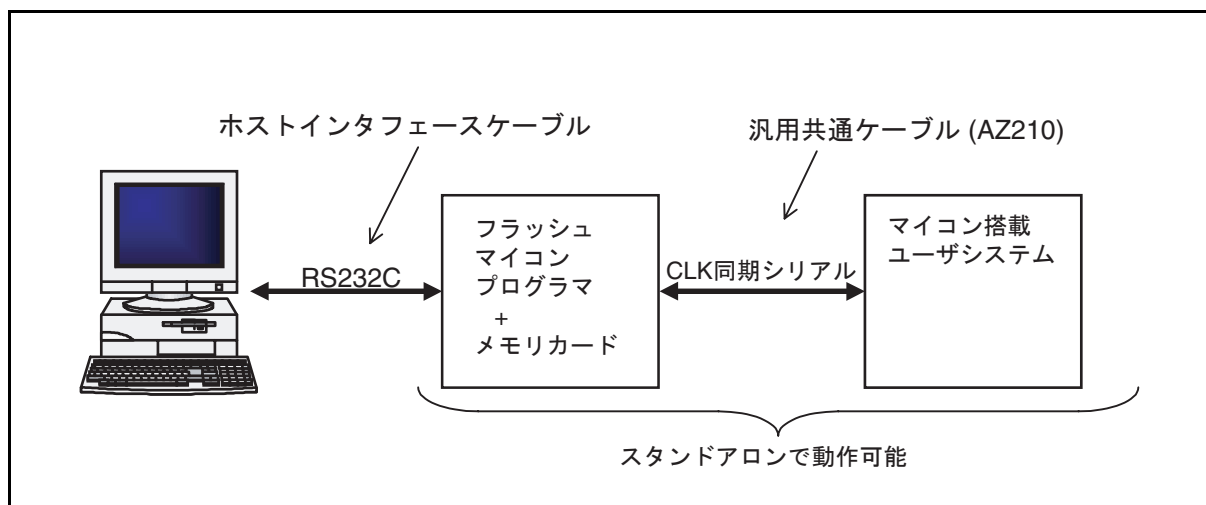
- B.1 シリアルオンボード書込み (富士通標準) の基本構成
- B.2 シリアル書込み接続例 (ユーザ電源)
- B.3 シリアル書込み接続例 (ライタ電源)
- B.4 フラッシュマイコンプログラマとの最小限の接続例 (ユーザ電源)
- B.5 フラッシュマイコンプログラマとの最小限の接続例 (ライタ電源)

B.1 シリアルオンボード書込み (富士通標準) の基本構成

MB90F822B/F823B ではフラッシュ ROM のシリアルオンボード書込み (富士通標準) をサポートしています。シリアルオンボード書込み仕様について以下に解説します。

■ シリアルオンボード書込み (富士通標準) の基本構成

富士通標準シリアルオンボード書込みには、横河デジタルコンピュータ株式会社製フラッシュマイコンプログラマを使用します。



< 注意事項 >

フラッシュマイコンプログラマの機能・操作方法と、接続用汎用共通ケーブル (AZ210)、コネクタにつきましては、横河デジタルコンピュータ株式会社へお問い合わせください。

付表 B.1-1 富士通標準シリアルオンボード書き込みに使用する端子 (1 / 2)

端子	機能	補足説明
MD2, MD1, MD0	モード端子	MD2=1, MD1=1, MD0=0 に設定した場合、フラッシュシリアル書込みモードになります。
X0, X1	発振子 端子	フラッシュシリアル書込みモードでは、CPU 内部動作クロックは PLL クロックの 1 週倍になります。発振クロック周波数が CPU 内部動作クロックになりますので、シリアル書込みに使用できる振動子は 1MHz ~ 16MHz になります。
P00, P01	書込みプログラマ起動端子	P00 に "L" レベル, P01 に "H" レベルを入力してください。
$\overline{\text{RST}}$	リセット端子	-
SIN0	シリアルデータ入力端子	UART0 をクロック同期モードで使用します。
SOT0	シリアルデータ出力端子	
SCK0	シリアルクロック入力端子	

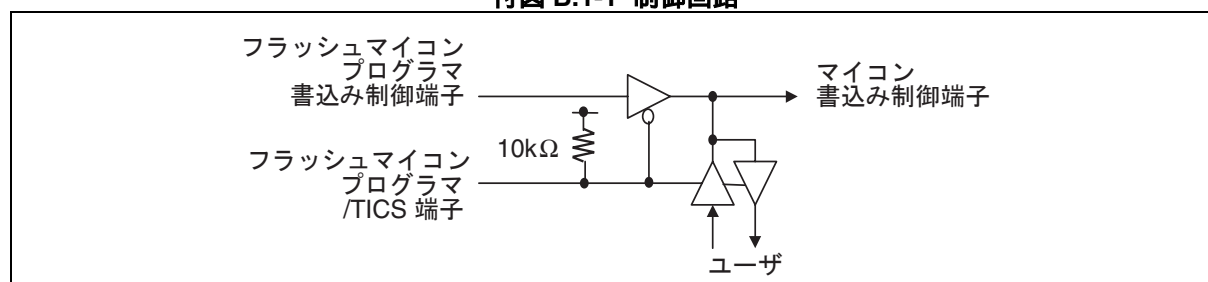
付表 B.1-1 富士通標準シリアルオンボード書き込みに使用する端子 (2 / 2)

端子	機能	補足説明
C	C 端子	電源安定化の容量端子です。外部に 0.1 μ F 程度のセラミックコンデンサを接続してください。
V _{CC}	電源供給端子	書き込み電圧 (5V \pm 10%) をユーザシステムから供給する場合には、フラッシュマイコンプログラマと接続する必要はありません。接続時にはユーザ電源と短絡しないようにしてください。
V _{SS}	GND 端子	フラッシュマイコンプログラマの GND と共通にしてください。

< 注意事項 >

P00, P01, SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には、付図 B.1-1 の制御回路が必要になります。
(フラッシュマイコンプログラマの /TICS 信号により、シリアル書き込み中の場合はユーザ回路を切り離すことができます。接続例をご参照ください。)

付図 B.1-1 制御回路



「B.2 シリアル書き込み接続例 (ユーザ電源)」～「B.5 フラッシュマイコンプログラマとの最小限の接続例 (ライター電源)」にシリアル書き込み接続例を示しますのでご参照ください。

- ・ シリアル書き込み接続例 (ユーザ電源)
- ・ シリアル書き込み接続例 (ライター電源)
- ・ フラッシュマイコンプログラマとの最小限の接続例 (ユーザ電源)
- ・ フラッシュマイコンプログラマとの最小限の接続例 (ライター電源)

付表 B.1-2 AF200 フラッシュマイコンプログラマ システム構成
(横河デジタルコンピュータ株式会社製)

型 格		機 能
本体	AF220/AC4P	イーサネットインタフェース内蔵モデル /100V ~ 220V 電源アダプタ
	AF210/AC4P	スタンダードモデル /100V ~ 220V 電源アダプタ
	AF120/AC4P	単キーイーサネットインタフェースモデル /100V ~ 220V 電源アダプタ
	AF110/AC4P	単キーモデル /100V ~ 220V 電源アダプタ
AZ221		ライタ専用 PC/AT 用 RS232C ケーブル
AZ210		標準ターゲットプローブ (a) 長 : 1m
FF201		富士通マイクロエレクトロニクス製 F ² MC-16LX フラッシュマイコン用コントロールモジュール
AZ290		リモートコントロール
/P2		2MB PC Card (Option) FLASH メモリ容量 ~ 128KB 対応
/P4		4MB PC Card (Option) FLASH メモリ容量 ~ 512KB 対応

問い合わせ先 : 横河デジタルコンピュータ株式会社

電話 : 042-333-6224

< 注意事項 >

フラッシュマイコンプログラマについては、終息製品ですが、コントロールモジュール FF201 を用いることで対応可能です。シリアル書込み接続例に関しては、「発振クロック周波数とシリアルクロック入力周波数」に示した接続例にて対応可能です。

■ 発振クロック周波数とシリアルクロック入力周波数

マイコンの入力可能なシリアルクロック周波数は以下の計算式より求められます。

入力可能なシリアルクロック周波数 = 0.125 × 発振クロック周波数例

したがって、ご使用の発振クロック周波数によって、シリアルクロック入力周波数をフラッシュマイコンプログラマの設定にて変更してください。

付表 B.1-3 入力可能なシリアルクロック周波数の例

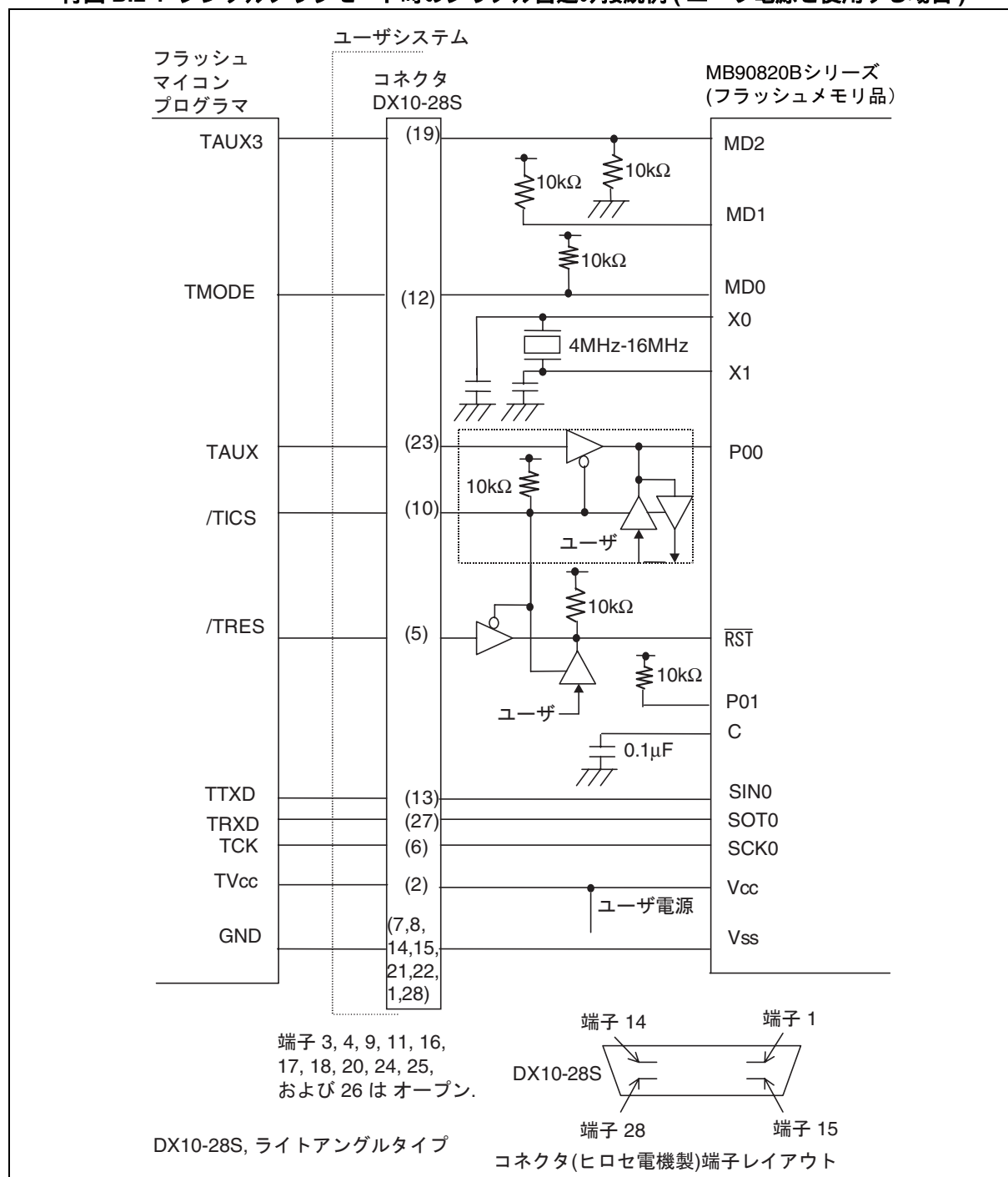
発振クロック周波数	マイコンの入力可能な最大シリアルクロック周波数	AF220/AF210/AF120/ AF110 の設定可能な最大 シリアルクロック周波数	AF200 の設定可能な 最大シリアルクロック 周波数
4MHz 時	500kHz	500kHz	500kHz
8MHz 時	1MHz	850kHz	500kHz
16MHz 時	2MHz	1.25MHz	500kHz

B.2 シリアル書込み接続例 (ユーザ電源)

付図 B.2-1 は、ユーザから電源を供給する場合のシリアル書込み接続例です。MD2 = 1 と MD0 = 0 は、フラッシュマイコンプログラムの TAUX3 と TMODE からそれぞれ入力されます。シリアル書込みモード : MD2, MD1, MD0 = 110_B

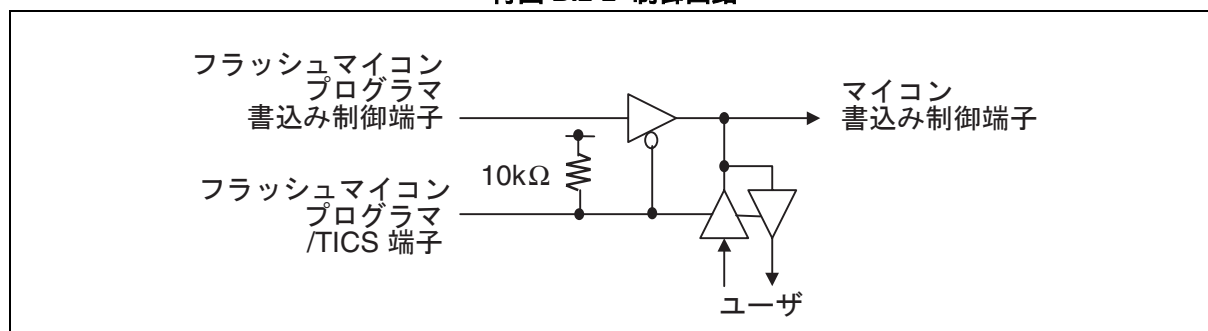
■ シリアル書込み接続例 (ユーザ電源を使用する場合)

付図 B.2-1 シングルチップモード時のシリアル書込み接続例 (ユーザ電源を使用する場合)



- SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には, P00 に接続されている制御回路と同様の回路 (付図 B.2-2) が必要となります (フラッシュマイコンプログラムの /TICS 信号により, シリアル書込み中はユーザ回路を切り離すことができます)。
- フラッシュマイコンプログラマとの接続はユーザ電源がOFFの状態で行ってください。

付図 B.2-2 制御回路

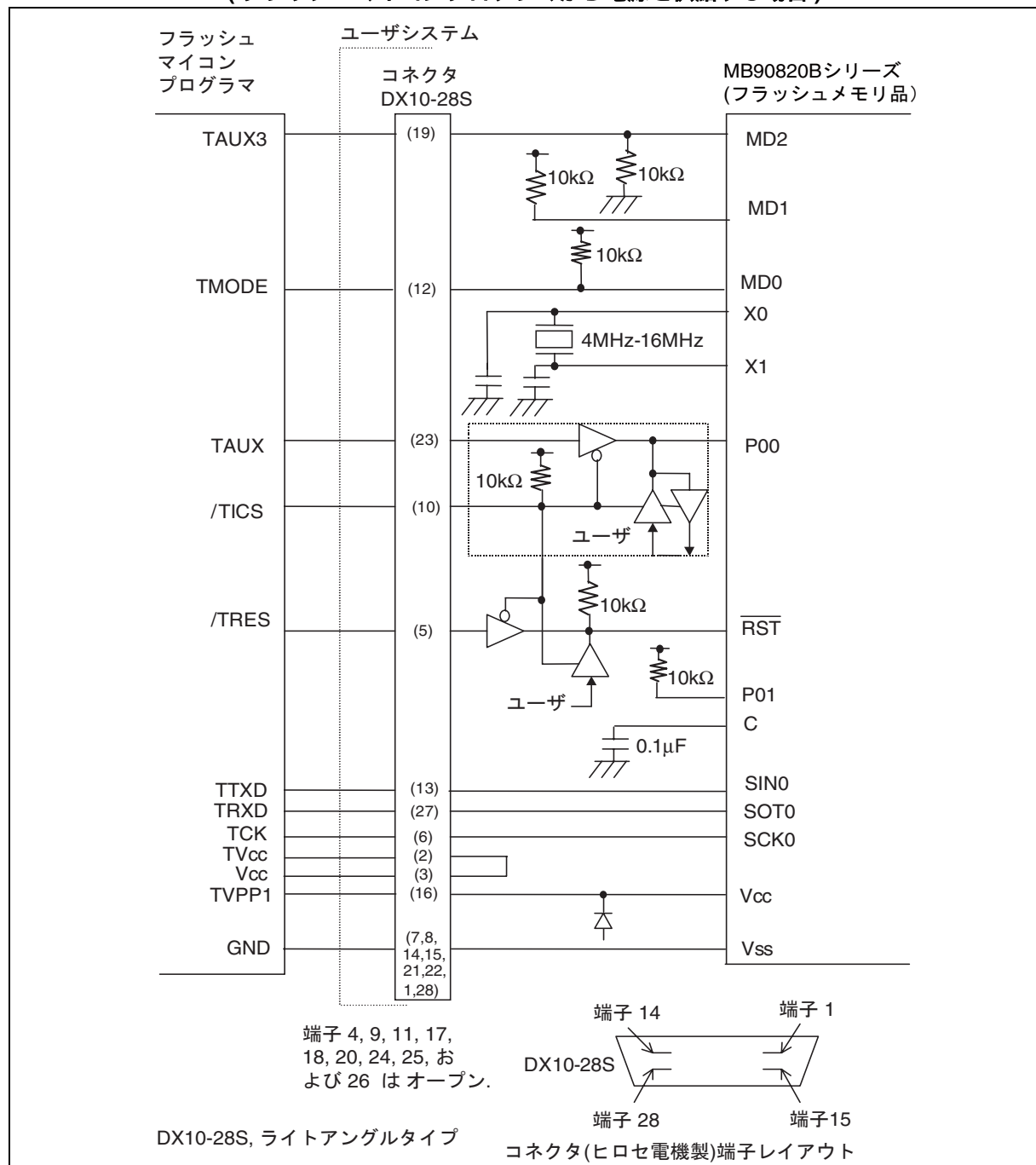


B.3 シリアル書込み接続例 (ライタ電源)

付図 B.3-1 は、ライタから電源を供給する場合のシリアル書込み接続例です。
MD2 = 1 と MD0 は、フラッシュマイコンプログラムの TAUX3 と TMODE からそれぞれに入力されます。シリアル書込みモード : MD2, MD1, MD0 = 110_B

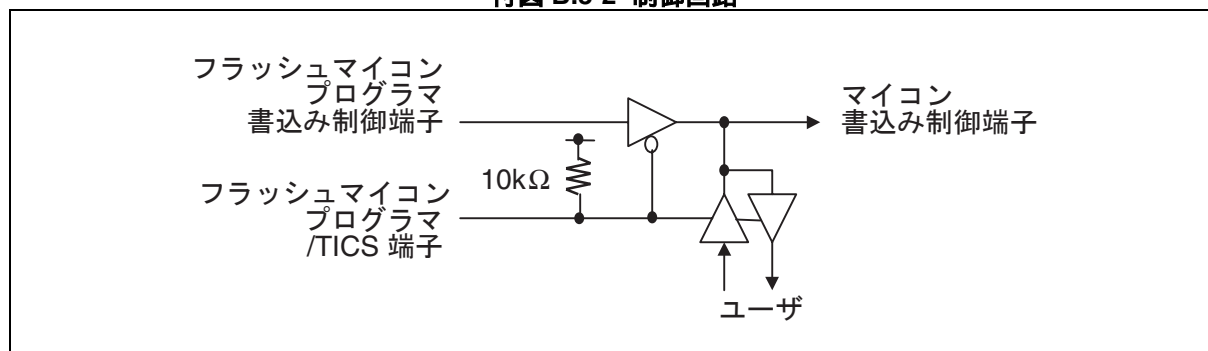
■ シリアル書込み接続例 (フラッシュマイコンプログラマから電源を供給する場合)

付図 B.3-1 MB90F822B/F823B 内部ベクタモードのシリアル書込み接続例
(フラッシュマイコンプログラマから電源を供給する場合)



- SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には、P00 に接続されている制御回路と同様の制御回路（付図 B.3-2）が必要となります（フラッシュマイコンプログラムの /TICS 信号により、シリアル書込み中はユーザ回路を切り離すことができます）。
- フラッシュマイコンプログラマとの接続はユーザ電源がOFFの状態で行ってください。
- 書込み電源をフラッシュマイコンプログラマから供給する場合は、ユーザ電源と短絡しないでください。

付図 B.3-2 制御回路



B.4 フラッシュマイコンプログラムの最小限の接続例 (ユーザ電源)

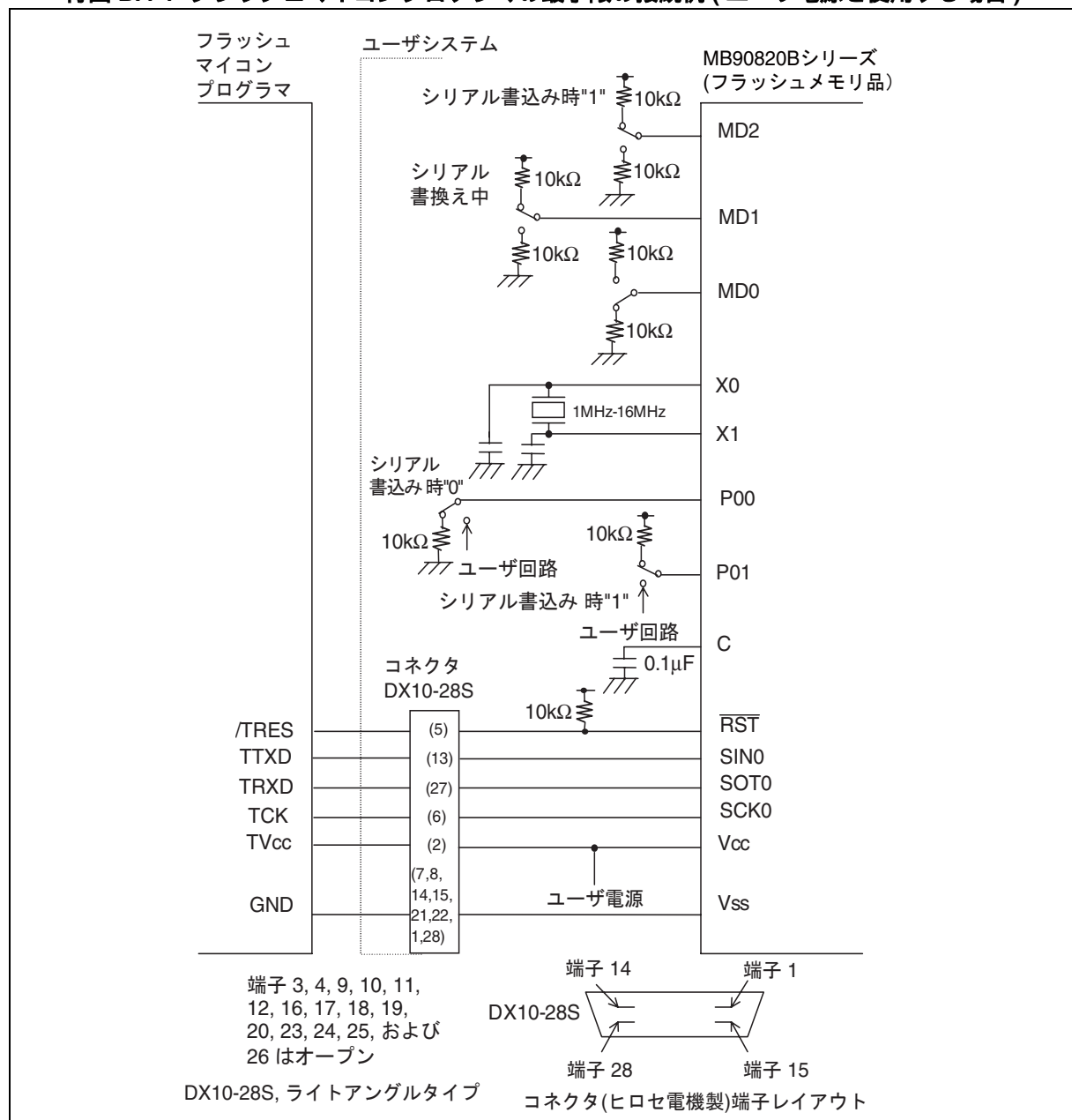
付図 B.4-1 は、ユーザから電源を供給する場合のフラッシュマイコンプログラムの最小限の接続例です。

シリアル書込みモード : MD2, MD1, MD0 = 110_B

■ フラッシュマイコンプログラムの最小限の接続例 (ユーザ電源を使用する場合)

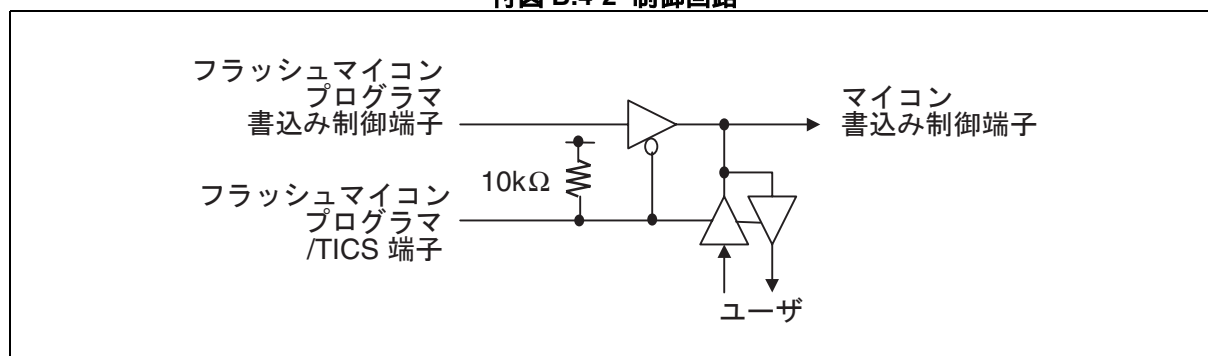
シリアル書込み時に、各端子を付図 B.4-1 のように設定した場合は、MD2, MD1, MD0, P00 とフラッシュマイコンプログラムの接続は必要ありません。

付図 B.4-1 フラッシュマイコンプログラムの最小限の接続例 (ユーザ電源を使用する場合)



- SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には、付図 B.4-2 の制御回路が必要となります（フラッシュマイコンプログラムの /TICS 信号により、シリアル書込み中はユーザ回路を切り離すことができます）。
- フラッシュマイコンプログラムの接続はユーザ電源がOFFの状態で行ってください。

付図 B.4-2 制御回路



付図 B.5-1 は、ライタから電源を供給する場合のフラッシュマイコンプログラマとの最小限の接続例です。

シリアル書込み時に、各端子を付図 B.5-1 のように設定した場合は、MD2, MD1, MD0, P00 とフラッシュマイコンプログラムの接続は必要ありません。

フラッシュマイコンプログラマ

ユーザシステム

MB90820Bシリーズ
(フラッシュメモリ品)

シリアル書き込み時"1"

シリアル書き込み時"1"

シリアル書き込み時"0"

4MHz-16MHz

シリアル書き込み時"0"

10kΩ

ユーザ回路

シリアル書き込み時"1"

ユーザ回路

コネクタ DX10-28S

端子4, 9, 10, 11, 12, 17, 18, 19, 20, 23, 24, 25, および 26 はオープン

端子14

端子1

端子28

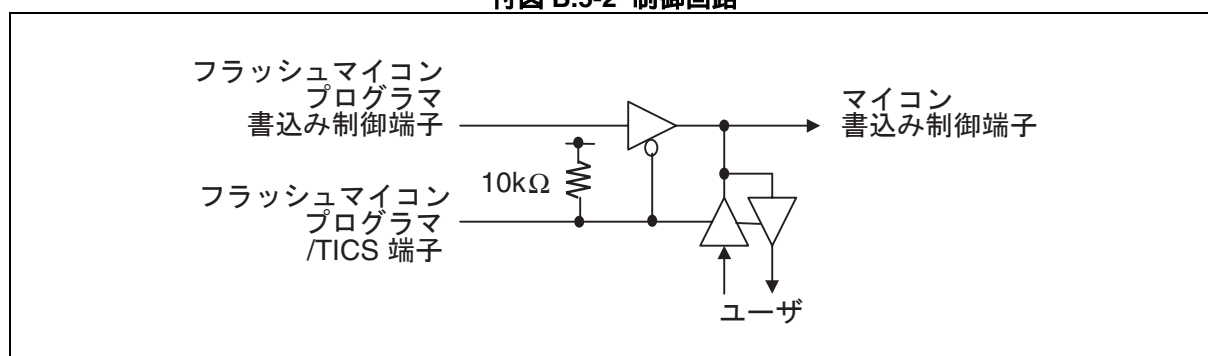
端子15

DX10-28S

コネクタ(ヒロセ電機製)端子レイアウト

- SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には、付図 B.5-2 の制御回路が必要となります(フラッシュマイコンプログラムの/TICS 信号により、シリアル書込み中はユーザ回路を切り離すことができます)。
- フラッシュマイコンプログラマとの接続はユーザ電源がOFFの状態で行ってください。
- 書込み電源をフラッシュマイコンプログラマから供給する場合は、ユーザ電源と短絡しないでください。

付図 B.5-2 制御回路



付録 C 命令

F²MC-16LX に使用している命令について説明します。

- C.1 命令の種類
- C.2 アドレッシング
- C.3 直接アドレッシング
- C.4 間接アドレッシング
- C.5 実行サイクル数
- C.6 実効アドレスフィールド
- C.7 命令一覧表の読み方
- C.8 F²MC-16LX 命令一覧表
- C.9 命令マップ

C.1 命令の種類

F²MC-16LX には、以下に示す 351 種類の命令があります。

■ 命令の種類

- 転送系命令 (バイト) 41 命令
- 転送系命令 (ワード, ロングワード) 38 命令
- 加減算命令 (バイト, ワード, ロングワード) 42 命令
- 増減算命令 (バイト, ワード, ロングワード) 12 命令
- 比較命令 (バイト, ワード, ロングワード) 11 命令
- 符号なし乗除算命令 (ワード, ロングワード) 11 命令
- 符号付き乗除算命令 (ワード, ロングワード) 11 命令
- 論理演算命令 (バイト, ワード) 39 命令
- 論理演算命令 (ロングワード) 6 命令
- 符号反転命令 (バイト, ワード) 6 命令
- ノーマライズ命令 (ロングワード) 1 命令
- シフト命令 (バイト, ワード, ロングワード) 18 命令
- 分岐命令 50 命令 (分岐命令 1: 31 命令, 分岐命令 2: 19 命令)
- アキュムレータ操作命令 (バイト, ワード) 6 命令
- その他制御命令 (バイト, ワード, ロングワード) 28 命令
- ビット操作命令 21 命令
- スtring命令 10 命令

C.2 アドレッシング

F²MC-16LX では、命令の実効アドレスフィールドまたは命令コード自体 (インプライド) でアドレス形式が決定されます。命令コード自体でアドレス形式が決定する場合は、使用する命令コードに合わせてアドレスを指定します。命令によっては、数種類のアドレス指定方式から設定できるものがあります。

■ アドレッシング

F²MC-16LX には、以下に示す 23 種類のアドレッシングがあります。

- 即値 (#imm)
- レジスタ直接
- 直接分岐アドレス (addr16)
- 物理直接分岐アドレス (addr24)
- I/O 直接 (io)
- 短縮直接アドレス (dir)
- 直接アドレス (addr16)
- I/O 直接ビットアドレス (io: bp)
- 短縮直接ビットアドレス (dir: bp)
- 直接ビットアドレス (addr16: bp)
- ベクタアドレス (#vct)
- レジスタ間接 (@RWj j=0 ~ 3)
- ポストインクリメント付レジスタ間接 (@RWj+ j=0 ~ 3)
- ディスプレースメント付レジスタ間接 (@RWi+disp8 i=0 ~ 7, @RWj+disp16 j=0 ~ 3)
- ディスプレースメント付ロングレジスタ間接 (@RLi+disp8 i=0 ~ 3)
- ディスプレースメント付プログラムカウンタ間接 (@PC+disp16)
- ベースインデックス付レジスタ間接 (@RW0+RW7, @RW1+RW7)
- プログラムカウンタ相対分岐アドレス (rel)
- レジスタリスト (rlist)
- アキュムレータ間接 (@A)
- アキュムレータ間接分岐アドレス (@A)
- 間接指定分岐アドレス (@ear)
- 間接指定分岐アドレス (@eam)

■ 実効アドレスフィールド

実効アドレスフィールドで指定される、アドレス形式を表 C.2-1 に示します。

表 C.2-1 実効アドレスフィールド

コード	表記			アドレス形式	デフォルトバンク
00	R0	RW0	RL0	レジスタ直接 ea は左から順に バイト ワード ロングワード の型に対応する。	なし
01	R1	RW1	(RL0)		
02	R2	RW2	RL1		
03	R3	RW3	(RL1)		
04	R4	RW4	RL2		
05	R5	RW5	(RL2)		
06	R6	RW6	RL3		
07	R7	RW7	(RL3)		
08	@RW0			レジスタ間接	DTB
09	@RW1				DTB
0A	@RW2				ADB
0B	@RW3				SPB
0C	@RW0+			ポストインクリメント付 レジスタ間接	DTB
0D	@RW1+				DTB
0E	@RW2+				ADB
0F	@RW3+				SPB
10	@RW0+disp8			8 ビットディスプレースメント付 レジスタ間接	DTB
11	@RW1+disp8				DTB
12	@RW2+disp8				ADB
13	@RW3+disp8				SPB
14	@RW4+disp8				DTB
15	@RW5+disp8				DTB
16	@RW6+disp8				ADB
17	@RW7+disp8				SPB
18	@RW0+disp16			16 ビットディスプレースメント付 レジスタ間接	DTB
19	@RW1+disp16				DTB
1A	@RW2+disp16				ADB
1B	@RW3+disp16				SPB
1C	@RW0+RW7			インデックス付レジスタ間接	DTB
1D	@RW1+RW7			インデックス付レジスタ間接	DTB
1E	@PC+disp16			16 ビットディスプレースメント付 PC 間接	PCB
1F	addr16			直接アドレス	DTB

C.3 直接アドレッシング

直接アドレッシングでは、オペランド値、レジスタおよびアドレスを直接指定します。

■ 直接アドレッシング

● 即値 (#imm)

オペランドの値を直接指定します。(#imm4/#imm8/#imm16/#imm32)。
図 C.3-1 に例を示します。

図 C.3-1 即値 (#imm) 例

MOVW A, #01212H (A にオペランドの値を格納する命令)		
実行前	A	2 2 3 3 : 4 4 5 5
実行後	A	4 4 5 5 : 1 2 1 2 (命令によっては AL AH に転送が行われる)

● レジスタ直接

オペランドとして、直接レジスタを指定します。指定できるレジスタを表 C.3-1 に示します。

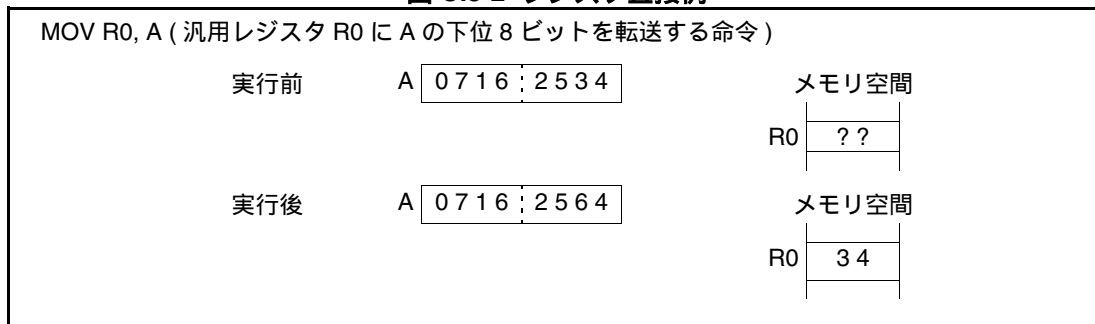
表 C.3-1 レジスタ直接

汎用レジスタ	バイト	R0, R1, R2, R3, R4, R5, R6, R7
	ワード	RW0, RW1, RW2, RW3, RW4, RW5, RW6, RW7
	ロングワード	RL0, RL1, RL2, RL3
専用レジスタ	アキュムレータ	A, AL
	ポインタ	SP *
	バンク	PCB, DTB, USB, SSB, ADB
	ページ	DPR
	制御	PS, CCR, RP, ILM

*: SP は、コンディションコードレジスタ (CCR) 中の S フラグビットの値に応じてユーザスタックポインタ (USP) またはシステムスタックポインタ (SSP) のどちらか一方が選択され、使用されます。分岐系の命令ではプログラムカウンタ (PC) は命令のオペランドには記述されることなく指定されます。

図 C.3-2 に例を示します。

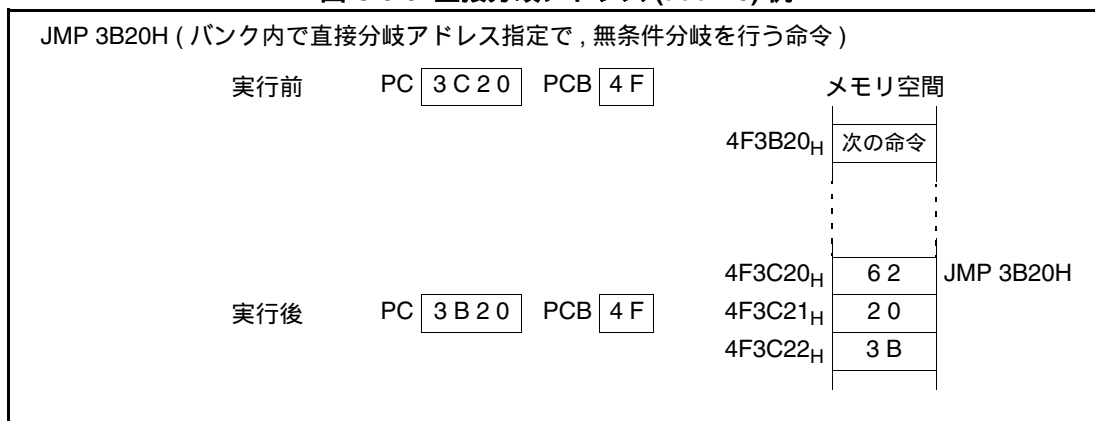
図 C.3-2 レジスタ直接例



● 直接分岐アドレス (addr16)

分岐先のアドレスをディスプレースメントで直接指定します。ディスプレースメントのデータ長は 16 ビットで、論理空間内での分岐先を示します。無条件分岐命令、サブルーチンコール命令およびソフトウェア割込み命令に使用します。アドレスの bit23 ~ bit16 はプログラムカウンタバンクレジスタ (PCB) で指定されます。図 C.3-3 に例を示します。

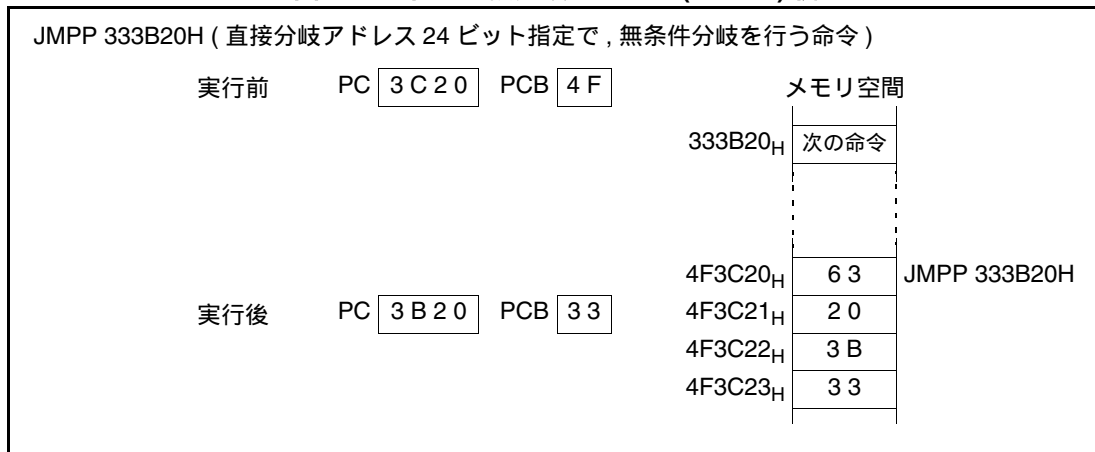
図 C.3-3 直接分岐アドレス (addr16) 例



● 物理直接分岐アドレス (addr24)

分岐先のアドレスをディスプレースメントで直接指定します。ディスプレースメントのデータ長は 24 ビットです。無条件分岐命令、サブルーチンコール命令およびソフトウェア割込み命令に使用します。図 C.3-4 に例を示します。

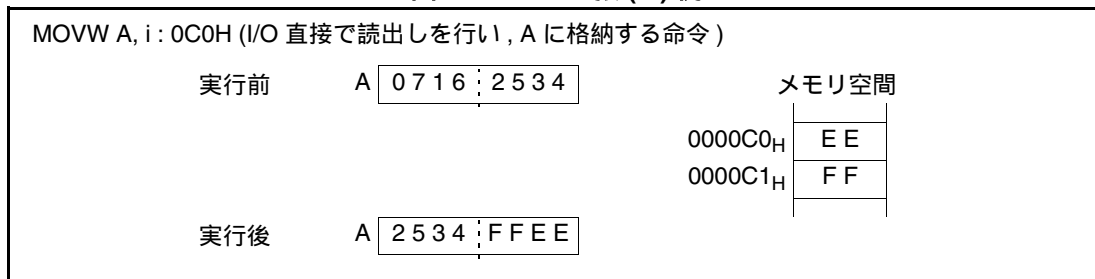
図 C.3-4 物理直接分岐アドレス (addr24) 例



● I/O 直接 (io)

オペランドのメモリアドレスを 8 ビットのディスプレースメントで直接指定します。データバンクレジスタ (DTB)、ダイレクトページレジスタ (DPR) の値にかかわらず、物理アドレス "000000_H" ~ "0000FF_H" の空間の I/O 空間がアクセスされます。I/O 直接アドレス指定を使用した命令の前にバンク指定用のバンクセレクトプリフィックスを記述しても無効です。図 C.3-5 に例を示します。

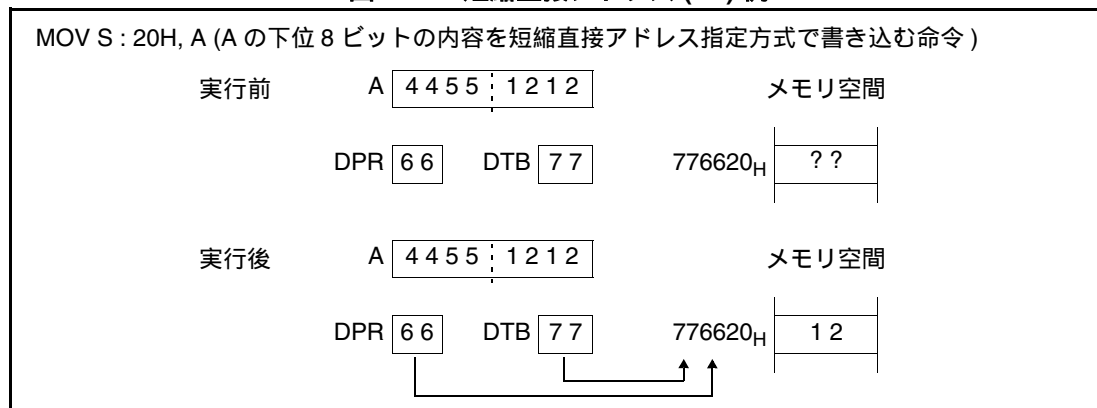
図 C.3-5 I/O 直接 (io) 例



● 短縮直接アドレス (dir)

オペランドでメモリアドレスの下位 8 ビットを直接指定します。アドレスの bit15 ~ bit8 はダイレクトページレジスタ (DPR) により指定されます。アドレスの bit23 ~ bit16 はデータバンクレジスタ (DTB) により指定されます。図 C.3-6 に例を示します。

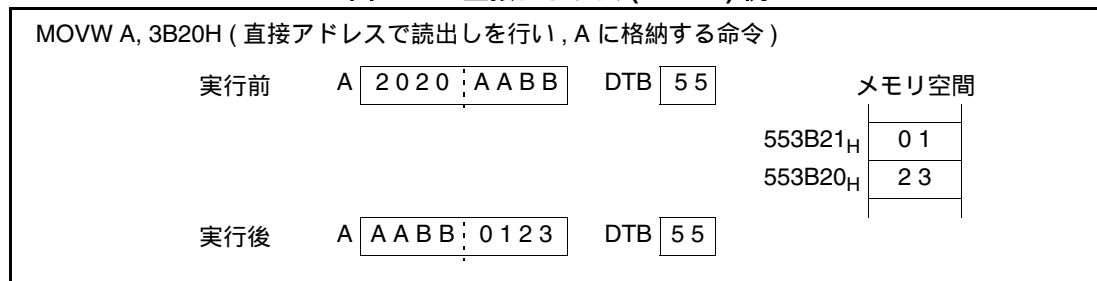
図 C.3-6 短縮直接アドレス (dir) 例



● 直接アドレス (addr16)

オペランドで、メモリアドレス下位 16 ビットを直接指定します。アドレスの bit23 ~ bit16 はデータバンクレジスタ (DTB) で指定されます。直接アドレスのアドレッシングに対しては、アクセス空間指定用のプリフィックス命令は無効です。図 C.3-7 に例を示します。

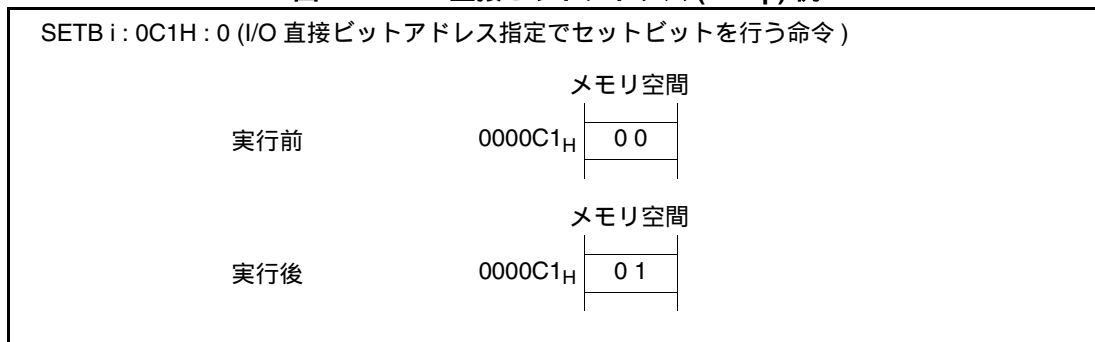
図 C.3-7 直接アドレス (addr16) 例



● I/O 直接ビットアドレス (io: bp)

物理アドレス "000000_H" ~ "0000FF_H" 内のビットを直接指定します。ビットの位置は ": bp" で表され, 数字の大きいほうが最上位ビット (MSB), 小さい方が最下位ビット (LSB) となります。図 C.3-8 に例を示します。

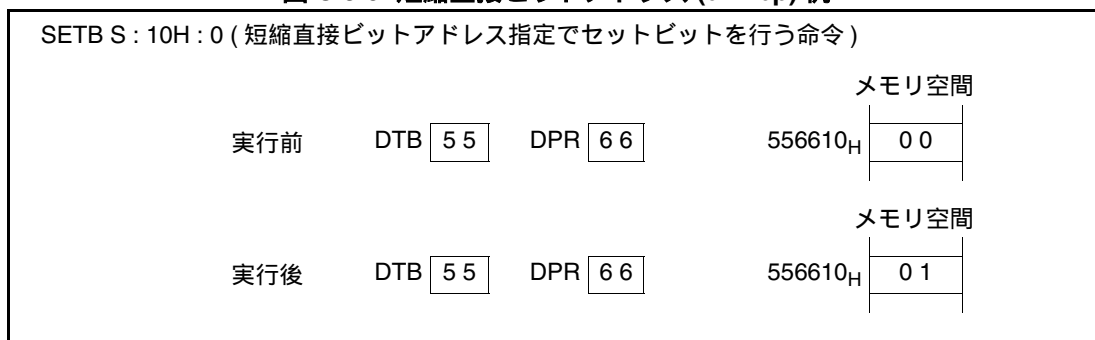
図 C.3-8 I/O 直接ビットアドレス (io: bp) 例



● 短縮直接ビットアドレス (dir: bp)

オペランドでメモリアドレスの下位 8 ビットを直接指定します。アドレスの bit15 ~ bit8 はダイレクトページレジスタ (DPR) で指定されます。アドレスの bit23 ~ bit16 はデータバンクレジスタ (DTB) で指定されます。ビットの位置は ": bp" で表され, 数字の大きい方が最上位ビット, 小さい方が最下位ビットとなります。図 C.3-9 に例を示します。

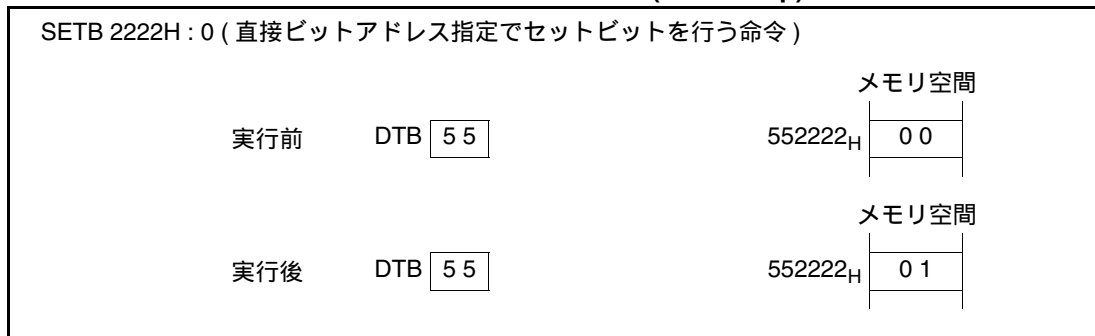
図 C.3-9 短縮直接ビットアドレス (dir: bp) 例



● 直接ビットアドレス (addr16: bp)

64K バイト内の任意のビットに対し直接指定します。アドレスの bit23 ~ bit16 はデータバンクレジスタ (DTB) で指定されます。ビットの位置は ": bp" で表され、数字の大きい方が最上位ビット、小さい方が最下位ビットとなります。図 C.3-10 に例を示します。

図 C.3-10 直接ビットアドレス (addr16: bp) 例



● ベクタアドレス (#vct)

分岐先のアドレスは指定されたベクタの内容となります。ベクタ番号のデータ長には 4 ビットと 8 ビットの 2 種類があります。サブルーチンコール命令、ソフトウェア割り込み命令に使用します。図 C.3-11 に例を示します。

図 C.3-11 ベクタアドレス (#vct) 例

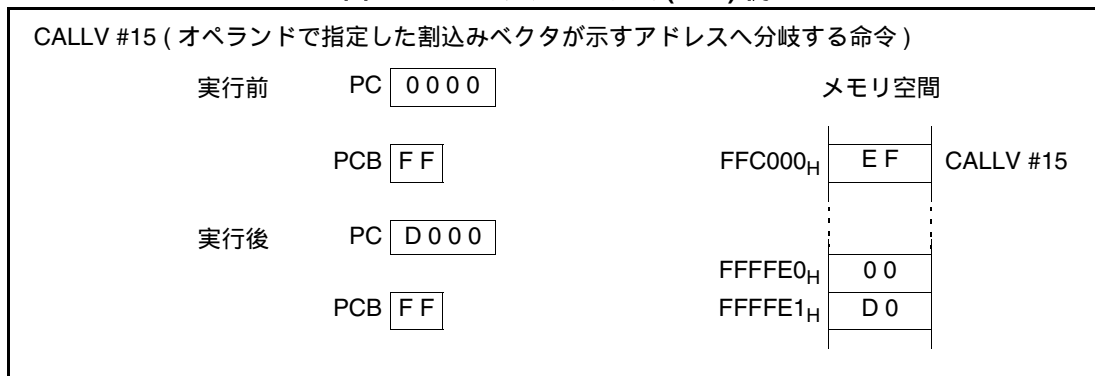


表 C.3-2 CALLV ベクター一覧表

命令	ベクタアドレス L	ベクタアドレス H
CALLV #0	XXFFFE _H	XXFFFF _H
CALLV #1	XXFFFC _H	XXFFFD _H
CALLV #2	XXFFFA _H	XXFFFB _H
CALLV #3	XXFFF8 _H	XXFFF9 _H
CALLV #4	XXFFF6 _H	XXFFF7 _H
CALLV #5	XXFFF4 _H	XXFFF5 _H
CALLV #6	XXFFF2 _H	XXFFF3 _H
CALLV #7	XXFFF0 _H	XXFFF1 _H
CALLV #8	XXFFEE _H	XXFFEF _H
CALLV #9	XXFFEC _H	XXFFED _H
CALLV #10	XXFFEA _H	XXFFEB _H
CALLV #11	XXFFE8 _H	XXFFE9 _H
CALLV #12	XXFFE6 _H	XXFFE7 _H
CALLV #13	XXFFE4 _H	XXFFE5 _H
CALLV #14	XXFFE2 _H	XXFFE3 _H
CALLV #15	XXFFE0 _H	XXFFE1 _H

(注意事項) XX には PCB レジスタの値が入ります。

< 注意事項 >

プログラムカウンタバンクレジスタ (PCB) が "FF_H" の場合ベクタ領域は INT #vct8 (#0 ~ #7) のベクタ領域と共有しているため、使用する場合は注意が必要です。(表 C.3-2 を参照してください)

C.4 間接アドレッシング

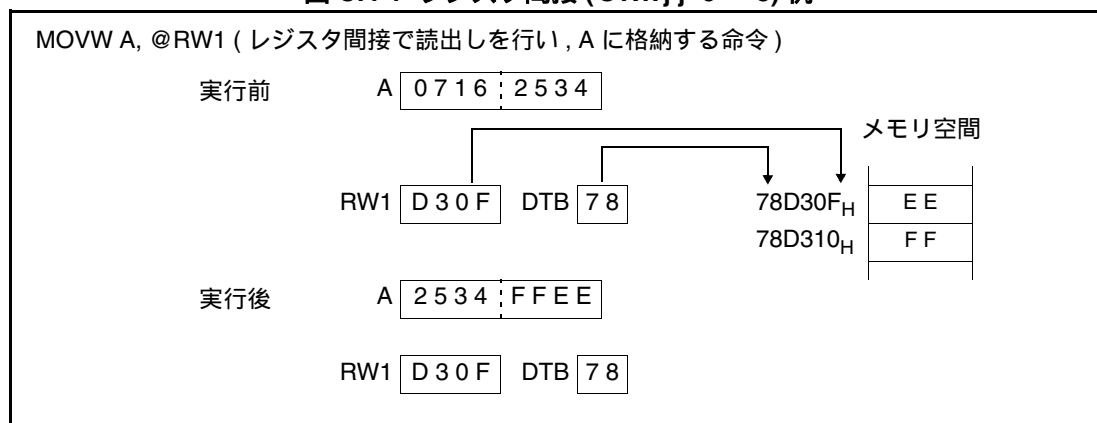
間接アドレッシングでは、記述したオペランドが示すアドレスのデータで、間接的にアドレスを指定します。

■ 間接アドレッシング

● レジスタ間接 (@RWj j=0 ~ 3)

汎用レジスタ RWj の内容をアドレスとしたメモリをアクセスするアドレッシングです。アドレスの bit23 ~ bit16 は、RW0, RW1 を用いた場合はデータバンクレジスタ (DTB) で示され、RW3 の場合はシステムスタックバンクレジスタ (SSB) またはユーザスタックバンクレジスタ (USB) で示され、RW2 の場合はアディショナルデータバンクレジスタ (ADB) で示されます。図 C.4-1 に例を示します。

図 C.4-1 レジスタ間接 (@RWj j=0 ~ 3) 例



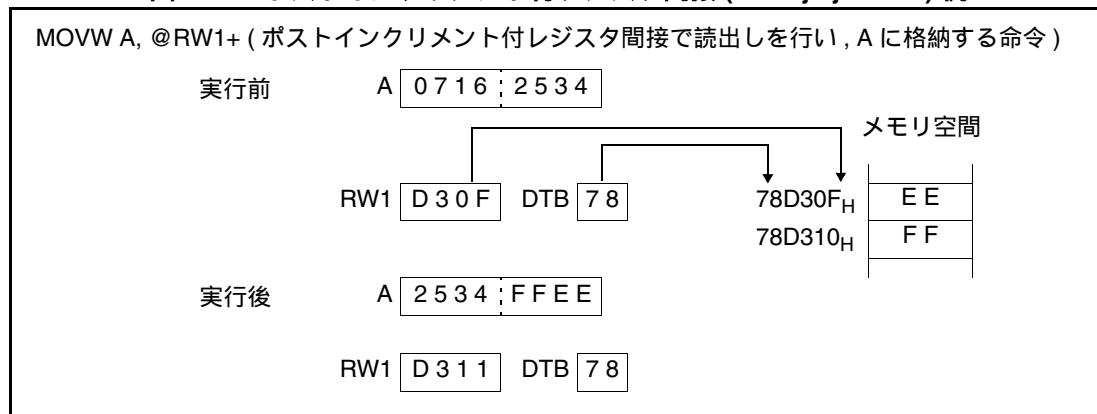
● ポストインクリメント付レジスタ間接 (@RWj+ j=0 ~ 3)

汎用レジスタ RWj の内容をアドレスとしたメモリをアクセスするアドレッシングです。オペランド操作後RWjはオペランドデータ長 (バイトの場合は1, ワードの場合は2, ロングワードの場合は4) 分だけ加算されます。アドレスの bit23 ~ bit16 は、RW0, RW1 を用いた場合はデータバンクレジスタ (DTB) で示され、RW3 の場合はシステムスタックバンクレジスタ (SSB) またはユーザスタックバンクレジスタ (USB) で示され、RW2 の場合はアディショナルデータバンクレジスタ (ADB) で示されます。

ポストインクリメントした結果がインクリメント指定したレジスタ自身のアドレスだった場合は、参照される値はインクリメントした値となり、命令が書き込みだった場合は命令による書き込みが優先されるので、インクリメントするはずだったレジスタは書き込みデータとなります。

図 C.4-2 に例を示します。

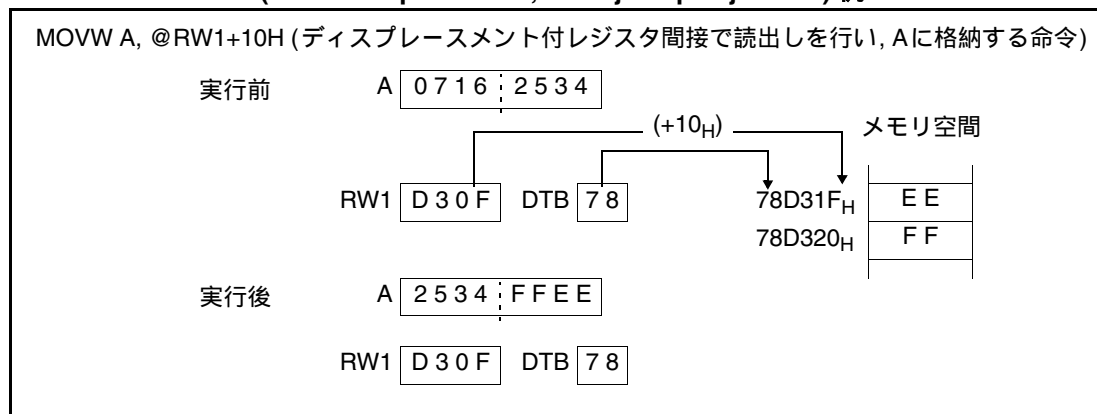
図 C.4-2 ポストインクリメント付レジスタ間接 (@RWj+ j=0 ~ 3) 例



● ディスプレースメント付レジスタ間接

(@RWi+disp8 i=0 ~ 7, @RWj+disp16 j=0 ~ 3)

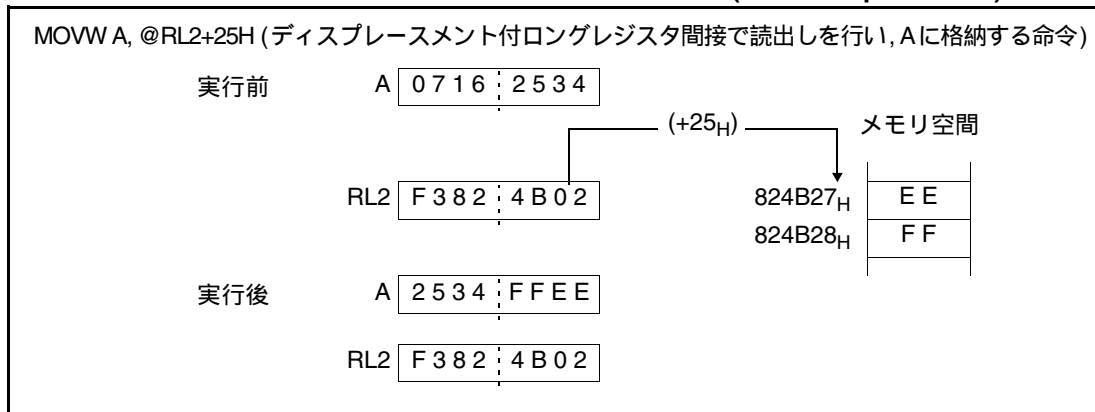
汎用レジスタ RWj の内容にディスプレースメントを加算したものをアドレスとしたメモリをアクセスするアドレッシングです。ディスプレースメントは、バイトとワードの2種類があり、符号付数値として加算されます。アドレスの bit23 ~ bit16 は, RW0, RW1, RW4, RW5 を用いた場合はデータバンクレジスタ (DTB) で示され, RW3, RW7 を用いた場合はシステムスタックバンクレジスタ (SSB) またはユーザスタックバンクレジスタ (USB) で示され, RW2, RW6 を用いた場合はアディショナルデータバンクレジスタ (ADB) で示されます。図 C.4-3 に例を示します。

図 C.4-3 ディスプレースメント付レジスタ間接
(@RWi+disp8 i=0 ~ 7, @RWj+disp16 j=0 ~ 3) 例

● ディスプレースメント付ロングレジスタ間接 (RLi+disp8 i=0 ~ 3)

汎用レジスタ RLi の内容にディスプレースメントを加算した結果の下位 24 ビットをアドレスとしたメモリをアクセスするアドレッシングです。ディスプレースメントは 8 ビットで符号付数値として加算されます。図 C.4-4 に例を示します。

図 C.4-4 ディスプレースメント付ロングレジスタ間接 (@RLi+disp8 i=0 ~ 3) 例



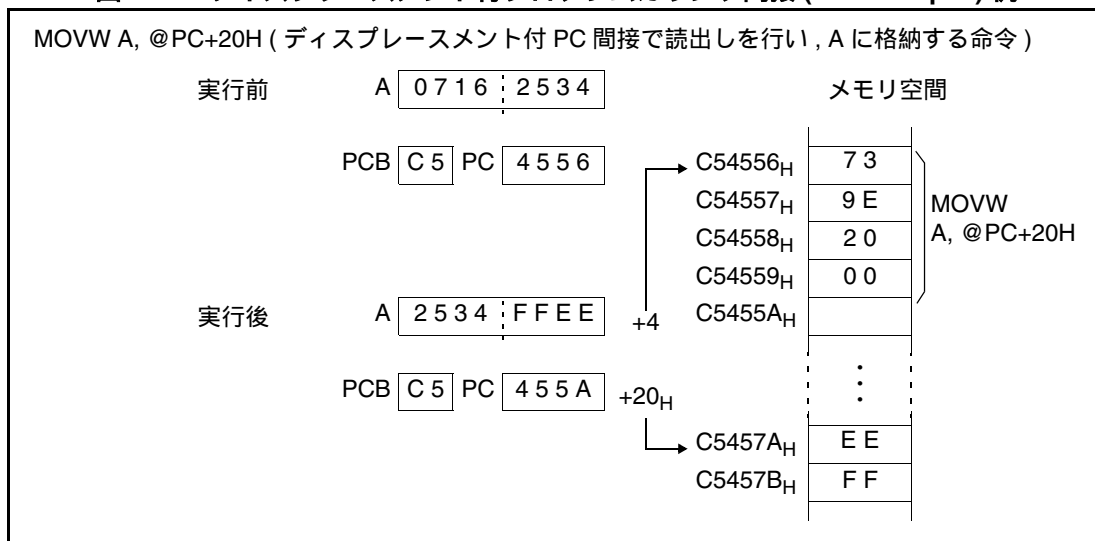
● ディスプレースメント付プログラムカウンタ間接 (@PC+disp16)

(命令のアドレス+4+disp16) で示されるアドレスのメモリをアクセスするアドレッシングです。ディスプレースメントはワード長です。アドレスの bit23 ~ bit16 はプログラムカウンタバンクレジスタ (PCB) により指定されます。次に示す命令のオペランドアドレスは, (次の命令のアドレス +disp16) とは, みなしませんでしたので注意してください。

- DBNZ eam, rel
- CBNE eam, #imm8, rel
- MOV eam, #imm8
- DWBNZ eam, rel
- CWBNE eam, #imm16, rel
- MOVW eam, #imm16

図 C.4-5 に例を示します。

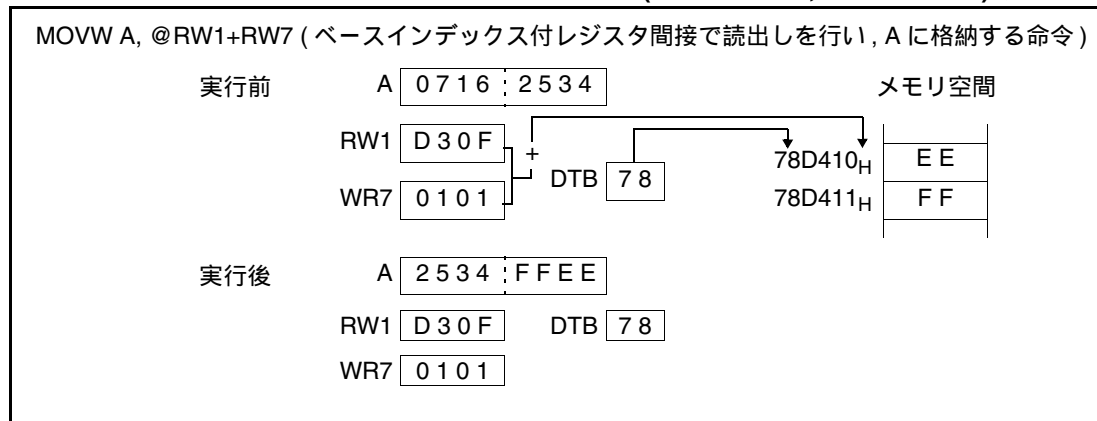
図 C.4-5 ディスプレースメント付プログラムカウンタ間接 (@PC+disp16) 例



● ベースインデックス付レジスタ間接 (@RW0+RW7, @RW1+RW7)

汎用レジスタ RW7 の内容に, RW0 あるいは RW1 を加算したものをアドレスとしたメモリをアクセスするアドレッシングです。アドレスの bit23 ~ bit16 はデータバンクレジスタ (DTB) で示されます。図 C.4-6 に例を示します。

図 C.4-6 ベースインデックス付レジスタ間接 (@RW0+RW7, @RW1+RW7) 例

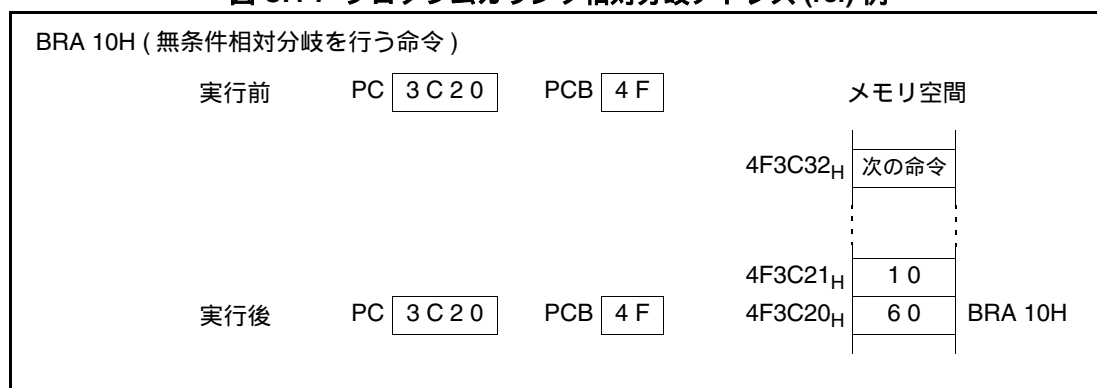


● プログラムカウンタ相対分岐アドレス (rel)

分岐先のアドレスはプログラムカウンタ (PC) の値と 8 ビットのディスプレースメントを加算した値となります。加算の結果が 16 ビットを超えた場合は, バンクレジスタのインクリメントまたはデクレメントをせずに, 超えた分は無視されますので, 64K バイトのバンク内で閉じたアドレスとなります。プログラムカウンタ相対分岐アドレスによるアドレッシングは, 無条件 / 条件分岐命令に使用します。アドレスの bit23 ~ bit16 はプログラムカウンタバンクレジスタ (PCB) で示されます。

図 C.4-7 に例を示します。

図 C.4-7 プログラムカウンタ相対分岐アドレス (rel) 例



● レジスタリスト (rlst)

スタックに対するプッシュ / ポップの対象となるレジスタを指定します。図 C.4-8 にレジスタリストの構成、図 C.4-9 に例を示します。

図 C.4-8 レジスタリストの構成

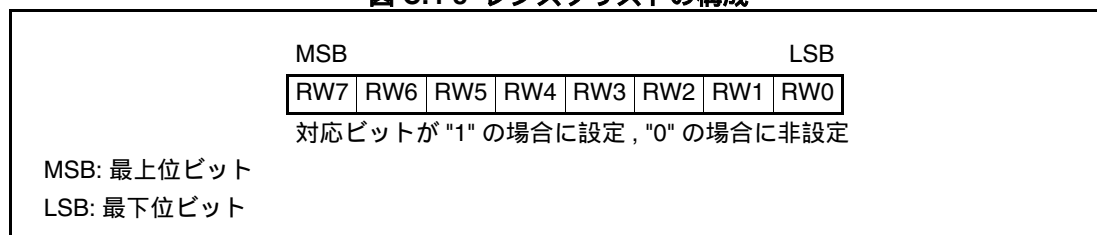
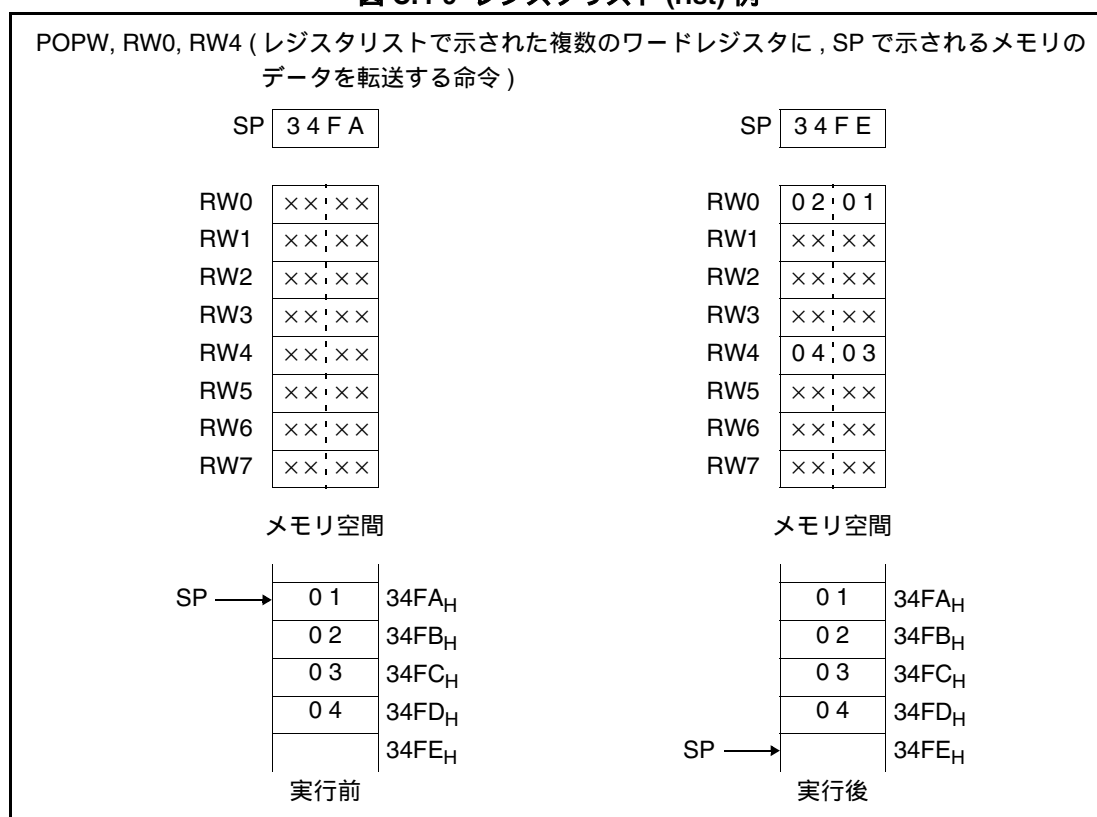


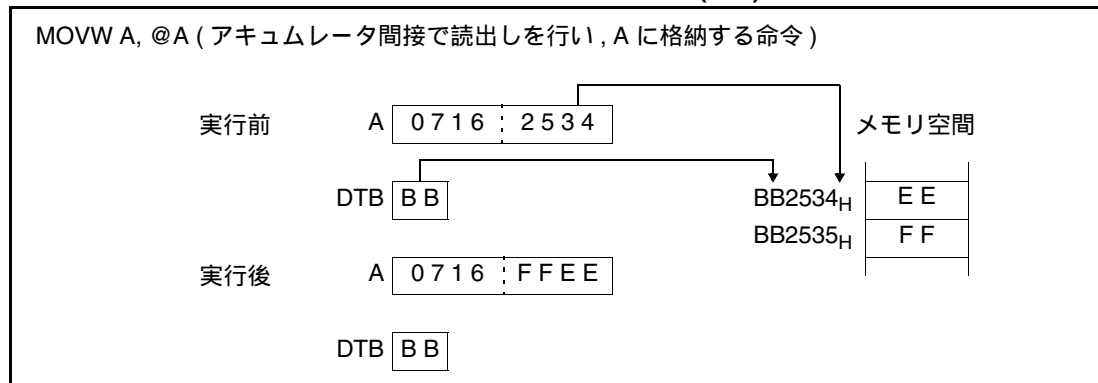
図 C.4-9 レジスタリスト (rlst) 例



● アキュムレータ間接 (@A)

アキュムレータの下位バイト (AL) の内容 (16 ビット) で示されるアドレスのメモリをアクセスするアドレス方式です。アドレスの bit23 ~ bit16 はデータバンクレジスタ (DTB) によりニーモニックで指定されます。図 C.4-10 に例を示します。

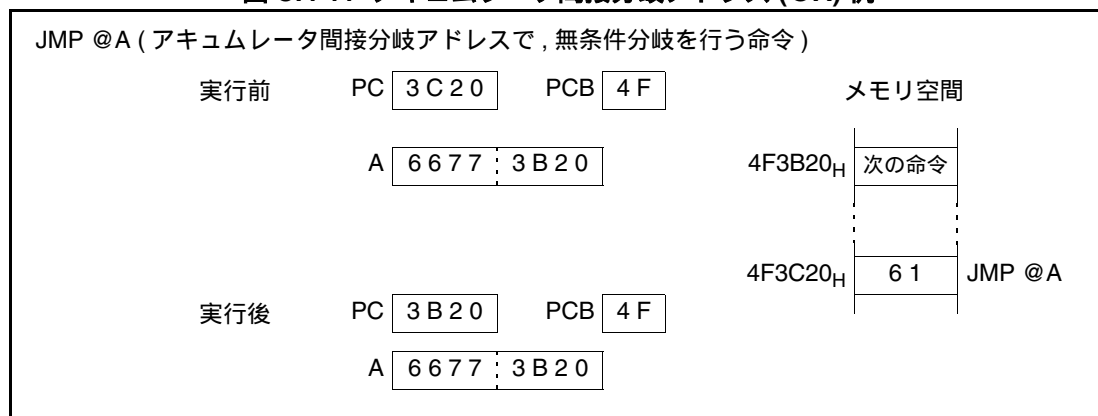
図 C.4-10 アキュムレータ間接 (@A) 例



● アキュムレータ間接分岐アドレス (@A)

分岐先のアドレスは, アキュムレータの下位バイト (AL) の内容 (16 ビット) となります。バンク空間内での分岐先を示し, アドレスの bit23 ~ bit16 はプログラムカウンタバンクレジスタ (PCB) で指定されますが, JCTX (Jump Context) 命令の場合は, アドレスの bit23 ~ bit16 はデータバンクレジスタ (DTB) で指定されます。アキュムレータ間接分岐アドレスによるアドレッシングは, 無条件分岐命令に使用します。図 C.4-11 に例を示します。

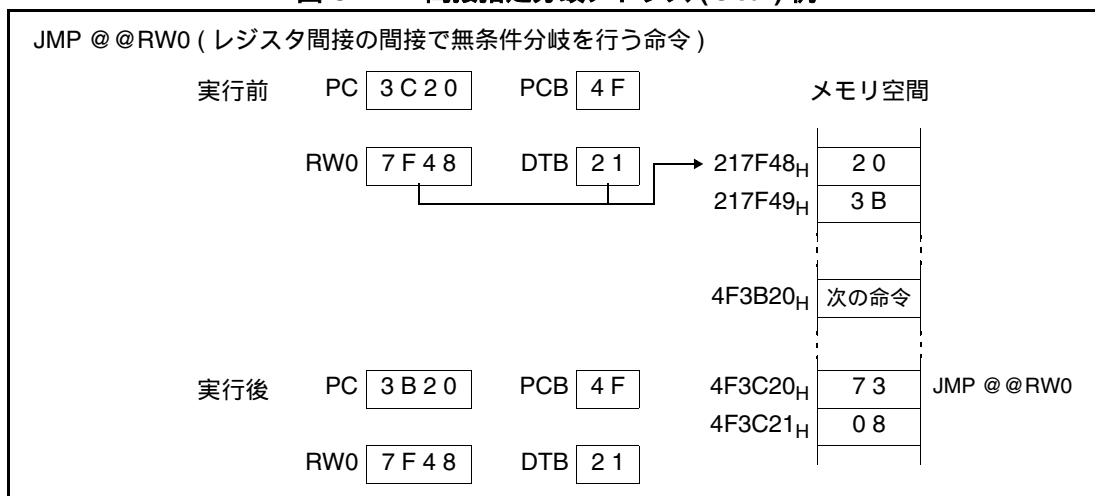
図 C.4-11 アキュムレータ間接分岐アドレス (@A) 例



● 間接指定分岐アドレス (@ear)

ear で示される番地のワードデータが分岐先アドレスとなります。図 C.4-12 に例を示します。

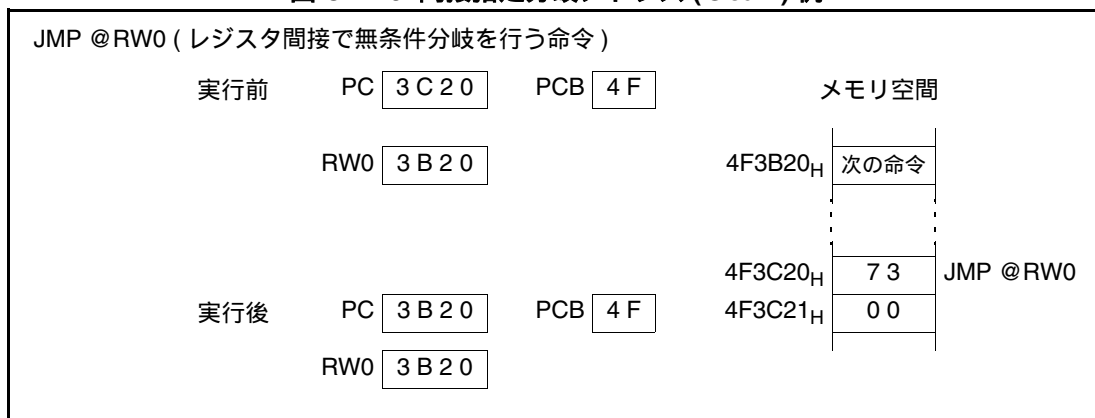
図 C.4-12 間接指定分岐アドレス (@ear) 例



● 間接指定分岐アドレス (@eam)

eam で示される番地のワードデータが分岐先アドレスとなります。図 C.4-13 に例を示します。

図 C.4-13 間接指定分岐アドレス (@eam) 例



C.5 実行サイクル数

命令の実行に要するサイクル数 (実行サイクル数) は、各命令の「サイクル数」の値と、条件で決まる「補正值」の値およびプログラムフェッチの「サイクル数」の値を加算することで得られます。

■ 実行サイクル数

内蔵 ROM などの 16 ビットバスに接続されたメモリ上のプログラムをフェッチする場合には、実行中の命令がワード境界を越えるごとにプログラムフェッチをするため、データのアクセスなどに干渉すると実行サイクル数が増大します。

外部データバスの 8 ビットバスに接続されたメモリ上のプログラムをフェッチする場合は、実行中の命令の 1 バイトごとにプログラムフェッチをするため、データのアクセスなどに干渉すると実行サイクル数が増大します。

CPU 間欠動作時は、汎用レジスタ、内蔵 ROM、内蔵 RAM、内蔵 I/O、外部データバスのアクセスをすると、低消費電力モード制御レジスタの CG0、CG1 ビットで指定されるサイクル数分 CPU へ供給されるクロックが一時停止しますので、CPU 間欠動作の命令の実行に要するサイクル数は、通常の実行サイクル数に、汎用レジスタ、内蔵 ROM、内蔵 RAM、内蔵 I/O、外部データバスの「アクセス回数」×一時停止の「サイクル数」の値を「補正值」として加算してください。

■ 実行サイクル数計算方法

表 C.5-1、表 C.5-2、表 C.5-3 に命令実行サイクル数および補正值のデータを示します。

表 C.5-1 各種アドレッシングに対する実行サイクル数

コード	オペランド	(a) *	各種アドレッシングに対するレジスタアクセス回数
		各種アドレッシングに対する実行サイクル数	
00 ~ 07	Ri Rwi RLi	命令一覧表に記載	命令一覧表に記載
08 ~ 0B	@RWj	2	1
0C ~ 0F	@RWj+	4	2
10 ~ 17	@Rwi+disp8	2	1
18 ~ 1B	@Rwi+disp16	2	1
1C	@RW0+RW7	4	2
1D	@RW1+RW7	4	2
1E	@PC+disp16	2	0
1F	addr16	1	0

*: (a) は「C.8 F2MC-16LX 命令一覧表」の ~ (サイクル数)、B (補正值) で使用されています。" ~ " と "B" の意味については「C.7 命令一覧表の読み方」を参照してください。

表 C.5-2 実サイクル数算出用サイクル数の補正值

オペランド	(b) バイト *1		(c) ワード *1		(d) ロング *1	
	サイクル数	アクセス回数	サイクル数	アクセス回数	サイクル数	アクセス回数
内部レジスタ	+0	1	+0	1	+0	2
内部メモリ 偶数アドレス	+0	1	+0	1	+0	2
内部メモリ 奇数アドレス	+0	1	+2	2	+4	4
外部データバス *2 16 ビット偶数アドレス	+1	1	+1	1	+2	2
外部データバス *2 16 ビット奇数アドレス	+1	1	+4	2	+8	4
外部データバス *2 8 ビット	+1	1	+4	2	+8	4

*1: (b), (c), (d) は「C.8 F2MC-16LX 命令一覧表」の～ (サイクル数), B (補正值) で使用されています。

*2: 外部データバスを使用した場合は、レディ入力および自動レディによりウェイトしたサイクル数も加算する必要があります。

表 C.5-3 プログラムフェッチサイクル数算出用サイクル数の補正值

命令	バイト境界	ワード境界
内部メモリ	-	+2
外部データバス 16 ビット	-	+3
外部データバス 8 ビット	+3	-

(注意事項) ・ 外部データバスを使用した場合は、レディ入力および自動レディでウェイトしたサイクル数も加算する必要があります。
・ 実際にはすべてのプログラムフェッチで、命令実行が遅くなるわけではないので、この補正值は最悪ケースを算出する場合に使用してください。

C.6 実効アドレスフィールド

表 C.6-1 に実効アドレスフィールドを示します。

■ 実効アドレスフィールド

表 C.6-1 実効アドレスフィールド (1 / 2)

コード	表記			アドレス形式	アドレス拡張部の バイト数 [*]
00	R0	RW0	RL0	レジスタ直接 ea は左から順に バイト ワード ロングワード の型に対応する。	-
01	R1	RW1	(RL0)		
02	R2	RW2	RL1		
03	R3	RW3	(RL1)		
04	R4	RW4	RL2		
05	R5	RW5	(RL2)		
06	R6	RW6	RL3		
07	R7	RW7	(RL3)		
08	@RW0			レジスタ間接	0
09	@RW1				
0A	@RW2				
0B	@RW3				
0C	@RW0+			ポストインクリメント付 レジスタ間接	0
0D	@RW1+				
0E	@RW2+				
0F	@RW3+				
10	@RW0+disp8			8 ビットディスプレースメント付 レジスタ間接	1
11	@RW1+disp8				
12	@RW2+disp8				
13	@RW3+disp8				
14	@RW4+disp8				
15	@RW5+disp8				
16	@RW6+disp8				
17	@RW7+disp8				

表 C.6-1 実効アドレスフィールド (1 / 2)

コード	表記	アドレス形式	アドレス拡張部の バイト数*
18	@RW0+disp16	16 ビットディスプレースメント付 レジスタ間接	2
19	@RW1+disp16		
1A	@RW2+disp16		
1B	@RW3+disp16		
1C	@RW0+RW7	インデックス付レジスタ間接	0
1D	@RW1+RW7	インデックス付レジスタ間接	0
1E	@PC+disp16	16 ビットディスプレースメント付 PC 間接	2
1F	addr16	直接アドレス	2

*: アドレス拡張部のバイト数は、「C.8 F2MC-16LX 命令一覧表」の#(バイト数)の "+" に当てはまります。
"#" の意味については「C.7 命令一覧表の読み方」を参照してください。

C.7 命令一覧表の読み方

「C.8 F2MC-16LX 命令一覧表」で使用している項目の説明を表 C.7-1 に、記号の説明を表 C.7-2 に示します。

■ 命令の表示記号の説明

表 C.7-1 命令一覧表の項目の説明

項目	説明
ニーモニック	英大文字、記号：アセンブラ上もそのまま表記します。 英小文字：アセンブラ上では、書き替えて記述します。 英小文字の後の数：命令中のビット幅を示します。
#	バイト数を示します。
~	サイクル数を示します。
RG	命令実行時のレジスタアクセス回数を示します。 CPU 間欠動作時の補正値を算出するのに使用します。
B	命令実行時の実サイクル数の算出用補正値を示します。 命令実行時の実サイクルは ~ 欄の数値を加算したものとなります。
オペレーション	命令の動作を示します。
LH	アキュムレータの bit15 ~ bit8 に対する特殊動作を示します。 Z: 0 を転送する。 X: 符号を拡張して転送する。 - : 転送しない。
AH	アキュムレータの上位 16 ビットに対する特殊動作を示します。 *: AL から AH へ転送する。 - : 転送しない。 Z: AH へ 00 _H を転送する。 X: AL の符号拡張で AH へ 00 _H または FF _H を転送する。
I	I (割込み許可), S (スタック), T (ステッキビット), N (ネガティブ), Z (ゼロ), V (オーバフロー), C (キャリー) の各フラグの状態を示します。 *: 命令の実行で変化する。 - : 変化しない。 S: 命令の実行でセットされる。 R: 命令の実行でリセットされる。
S	
T	
N	
Z	
V	
C	
RMW	
	リードモディファイライト命令 (1 命令でメモリなどからデータを読み出し、メモリへ書き込む) であるかどうかを示します。 *: リードモディファイライト命令である。 - : リードモディファイライト命令ではない。 (注意事項) 読み書きで意味の異なるアドレスには使用できません。

表 C.7-2 命令一覧表の記号の説明 (1 / 2)

表記	意味
A	32 ビットアキュムレータ 命令により、使用されるビット長が変わります。 バイト：AL の下位 8 ビット ワード：AL の 16 ビット ロング：AL: AH の 32 ビット
AH	A の上位 16 ビット
AL	A の下位 16 ビット
SP	スタックポインタ (USP or SSP)
PC	プログラムカウンタ
PCB	プログラムカウンタバンクレジスタ
DTB	データバンクレジスタ
ADB	アディショナルデータバンクレジスタ
SSB	システムスタックバンクレジスタ
USB	ユーザスタックバンクレジスタ
SPB	カレントスタックバンクレジスタ (SSB or USB)
DPR	ダイレクトページレジスタ
brg1	DTB, ADB, SSB, USB, DPR, PCB, SPB
brg2	DTB, ADB, SSB, USB, DPR, SPB
Ri	R0, R1, R2, R3, R4, R5, R6, R7
RWi	RW0, RW1, RW2, RW3, RW4, RW5, RW6, RW7
RWj	RW0, RW1, RW2, RW3
RLi	RL0, RL1, RL2, RL3
dir	短縮直接アドレス指定
addr16	直接アドレス指定
addr24	物理直接アドレス指定
ad24 0 ~ 15	addr24 の bit0 ~ bit15
ad24 16 ~ 23	addr24 の bit16 ~ bit23
io	I/O 領域 (000000 _H ~ 0000FF _H)
#imm4	4 ビット即値データ
#imm8	8 ビット即値データ
#imm16	16 ビット即値データ
#imm32	32 ビット即値データ
ext (imm8)	8 ビット即値データを符号拡張した 16 ビットデータ

表 C.7-2 命令一覧表の記号の説明 (1 / 2)

表記	意味
disp8	8 ビットディスプレースメント
disp16	16 ビットディスプレースメント
bp	ビットオフセット値
vct4	ベクタ番号 (0 ~ 15)
vct8	ベクタ番号 (0 ~ 255)
() b	ビットアドレス
rel	PC 相対分岐指定
ear	実効アドレス指定 (コード 00 ~ 07)
eam	実効アドレス指定 (コード 08 ~ 1F)
rlst	レジスタ並び

C.8 F²MC-16LX 命令一覧表

F²MC-16LX で使用している命令の一覧を示します。

■ F²MC-16LX 命令一覧表

表 C.8-1 転送系命令 (バイト) 41 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
MOV A,dir	2	3	0	(b)	byte (A) (dir)	Z	*	-	-	-	*	*	-	-	-
MOV A,addr16	3	4	0	(b)	byte (A) (addr16)	Z	*	-	-	-	*	*	-	-	-
MOV A,Ri	1	2	1	0	byte (A) (Ri)	Z	*	-	-	-	*	*	-	-	-
MOV A,ear	2	2	1	0	byte (A) (ear)	Z	*	-	-	-	*	*	-	-	-
MOV A,eam	2+	3 + (a)	0	(b)	byte (A) (eam)	Z	*	-	-	-	*	*	-	-	-
MOV A,io	2	3	0	(b)	byte (A) (io)	Z	*	-	-	-	*	*	-	-	-
MOV A,#imm8	2	2	0	0	byte (A) imm8	Z	*	-	-	-	*	*	-	-	-
MOV A,@A	2	3	0	(b)	byte (A) ((A))	Z	-	-	-	-	*	*	-	-	-
MOV A,@RLi+disp8	3	10	2	(b)	byte (A) ((RLi)+disp8)	Z	*	-	-	-	*	*	-	-	-
MOVN A,#imm4	1	1	0	0	byte (A) imm4	Z	*	-	-	-	R	*	-	-	-
MOVX A,dir	2	3	0	(b)	byte (A) (dir)	X	*	-	-	-	*	*	-	-	-
MOVX A,addr16	3	4	0	(b)	byte (A) (addr16)	X	*	-	-	-	*	*	-	-	-
MOVX A,Ri	2	2	1	0	byte (A) (Ri)	X	*	-	-	-	*	*	-	-	-
MOVX A,ear	2	2	1	0	byte (A) (ear)	X	*	-	-	-	*	*	-	-	-
MOVX A,eam	2+	3 + (a)	0	(b)	byte (A) (eam)	X	*	-	-	-	*	*	-	-	-
MOVX A,io	2	3	0	(b)	byte (A) (io)	X	*	-	-	-	*	*	-	-	-
MOVX A,#imm8	2	2	0	0	byte (A) imm8	X	*	-	-	-	*	*	-	-	-
MOVX A,@A	2	3	0	(b)	byte (A) ((A))	X	-	-	-	-	*	*	-	-	-
MOVX A,@RWi+disp8	2	5	1	(b)	byte (A) ((RWi)+disp8)	X	*	-	-	-	*	*	-	-	-
MOVX A,@RLi+disp8	3	10	2	(b)	byte (A) ((RLi)+disp8)	X	*	-	-	-	*	*	-	-	-
MOV dir,A	2	3	0	(b)	byte (dir) (A)	-	-	-	-	-	*	*	-	-	-
MOV addr16,A	3	4	0	(b)	byte (addr16) (A)	-	-	-	-	-	*	*	-	-	-
MOV Ri,A	1	2	1	0	byte (Ri) (A)	-	-	-	-	-	*	*	-	-	-
MOV ear,A	2	2	1	0	byte (ear) (A)	-	-	-	-	-	*	*	-	-	-
MOV eam,A	2+	3 + (a)	0	(b)	byte (eam) (A)	-	-	-	-	-	*	*	-	-	-
MOV io,A	2	3	0	(b)	byte (io) (A)	-	-	-	-	-	*	*	-	-	-
MOV @RLi+disp8,A	3	10	2	(b)	byte ((RLi)+disp8) (A)	-	-	-	-	-	*	*	-	-	-
MOV Ri,ear	2	3	2	0	byte (Ri) (ear)	-	-	-	-	-	*	*	-	-	-
MOV Ri,eam	2+	4 + (a)	1	(b)	byte (Ri) (eam)	-	-	-	-	-	*	*	-	-	-
MOV ear,Ri	2	4	2	0	byte (ear) (Ri)	-	-	-	-	-	*	*	-	-	-
MOV eam,Ri	2+	5 + (a)	1	(b)	byte (eam) (Ri)	-	-	-	-	-	*	*	-	-	-
MOV Ri,#imm8	2	2	1	0	byte (Ri) imm8	-	-	-	-	-	*	*	-	-	-
MOV io,#imm8	3	5	0	(b)	byte (io) imm8	-	-	-	-	-	-	-	-	-	-
MOV dir,#imm8	3	5	0	(b)	byte (dir) imm8	-	-	-	-	-	-	-	-	-	-
MOV ear,#imm8	3	2	1	0	byte (ear) imm8	-	-	-	-	-	*	*	-	-	-
MOV eam,#imm8	3+	4 + (a)	0	(b)	byte (eam) imm8	-	-	-	-	-	-	-	-	-	-
MOV @AL,AH	2	3	0	(b)	byte ((A)) (AH)	-	-	-	-	-	*	*	-	-	-
XCH A,ear	2	4	2	0	byte (A) (ear)	Z	-	-	-	-	-	-	-	-	-
XCH A,eam	2+	5 + (a)	0	2 × (b)	byte (A) (eam)	Z	-	-	-	-	-	-	-	-	-
XCH Ri,ear	2	7	4	0	byte (Ri) (ear)	-	-	-	-	-	-	-	-	-	-
XCH Ri,eam	2+	9 + (a)	2	2 × (b)	byte (Ri) (eam)	-	-	-	-	-	-	-	-	-	-

(注意事項) 表中の (a), (b) は , 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-2 転送系命令 (ワード・ロングワード) 38 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
MOVW A,dir	2	3	0	(c)	word (A) (dir)	-	*	-	-	-	*	*	-	-	-
MOVW A,addr16	3	4	0	(c)	word (A) (addr16)	-	*	-	-	-	*	*	-	-	-
MOVW A,SP	1	1	0	0	word (A) (SP)	-	*	-	-	-	*	*	-	-	-
MOVW A,RWi	1	2	1	0	word (A) (RWi)	-	*	-	-	-	*	*	-	-	-
MOVW A,ear	2	2	1	0	word (A) (ear)	-	*	-	-	-	*	*	-	-	-
MOVW A,eam	2+	3 + (a)	0	(c)	word (A) (eam)	-	*	-	-	-	*	*	-	-	-
MOVW A,io	2	3	0	(c)	word (A) (io)	-	*	-	-	-	*	*	-	-	-
MOVW A,@A	2	3	0	(c)	word (A) ((A))	-	-	-	-	-	*	*	-	-	-
MOVW A,#imm16	3	2	0	0	word (A) imm16	-	*	-	-	-	*	*	-	-	-
MOVW A,@RWi+disp8	2	5	1	(c)	word (A) ((RWi)+disp8)	-	*	-	-	-	*	*	-	-	-
MOVW A,@RLi+disp8	3	10	2	(c)	word (A) ((RLi)+disp8)	-	*	-	-	-	*	*	-	-	-
MOVW dir,A	2	3	0	(c)	word (dir) (A)	-	-	-	-	-	*	*	-	-	-
MOVW addr16,A	3	4	0	(c)	word (addr16) (A)	-	-	-	-	-	*	*	-	-	-
MOVW SP,A	1	1	0	0	word (SP) (A)	-	-	-	-	-	*	*	-	-	-
MOVW RWi,A	1	2	1	0	word (RWi) (A)	-	-	-	-	-	*	*	-	-	-
MOVW ear,A	2	2	1	0	word (ear) (A)	-	-	-	-	-	*	*	-	-	-
MOVW eam,A	2+	3 + (a)	0	(c)	word (eam) (A)	-	-	-	-	-	*	*	-	-	-
MOVW io,A	2	3	0	(c)	word (io) (A)	-	-	-	-	-	*	*	-	-	-
MOVW @RWi+disp8,A	2	5	1	(c)	word ((RWi)+disp8) (A)	-	-	-	-	-	*	*	-	-	-
MOVW @RLi+disp8,A	3	10	2	(c)	word ((RLi)+disp8) (A)	-	-	-	-	-	*	*	-	-	-
MOVW RWi,ear	2	3	2	0	word (RWi) (ear)	-	-	-	-	-	*	*	-	-	-
MOVW RWi,eam	2+	4 + (a)	1	(c)	word (RWi) (eam)	-	-	-	-	-	*	*	-	-	-
MOVW ear,RWi	2	4	2	0	word (ear) (RWi)	-	-	-	-	-	*	*	-	-	-
MOVW eam,RWi	2+	5 + (a)	1	(c)	word (eam) (RWi)	-	-	-	-	-	*	*	-	-	-
MOVW RWi,#imm16	3	2	1	0	word (RWi) imm16	-	-	-	-	-	*	*	-	-	-
MOVW io,#imm16	4	5	0	(c)	word (io) imm16	-	-	-	-	-	-	-	-	-	-
MOVW ear,#imm16	4	2	1	0	word (ear) imm16	-	-	-	-	-	*	*	-	-	-
MOVW eam,#imm16	4+	4 + (a)	0	(c)	word (eam) imm16	-	-	-	-	-	-	-	-	-	-
MOVW @AL,AH	2	3	0	(c)	word ((A)) (AH)	-	-	-	-	-	*	*	-	-	-
XCHW A,ear	2	4	2	0	word (A) (ear)	-	-	-	-	-	-	-	-	-	-
XCHW A,eam	2+	5 + (a)	0	2 × (c)	word (A) (eam)	-	-	-	-	-	-	-	-	-	-
XCHW RWi, ear	2	7	4	0	word (RWi) (ear)	-	-	-	-	-	-	-	-	-	-
XCHW RWi, eam	2+	9 + (a)	2	2 × (c)	word (RWi) (eam)	-	-	-	-	-	-	-	-	-	-
MOVL A,ear	2	4	2	0	long (A) (ear)	-	-	-	-	-	*	*	-	-	-
MOVL A,eam	2+	5 + (a)	0	(d)	long (A) (eam)	-	-	-	-	-	*	*	-	-	-
MOVL A,#imm32	5	3	0	0	long (A) imm32	-	-	-	-	-	*	*	-	-	-
MOVL ear,A	2	4	2	0	long (ear) (A)	-	-	-	-	-	*	*	-	-	-
MOVL eam,A	2+	5 + (a)	0	(d)	long(eam) (A)	-	-	-	-	-	*	*	-	-	-

(注意事項) 表中の (a), (c), (d) は、表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-3 加減算命令 (バイト・ワード・ロングワード) 42 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
ADD A,#imm8	2	2	0	0	byte (A) (A) + imm8	Z	-	-	-	-	*	*	*	*	-
ADD A,dir	2	5	0	(b)	byte (A) (A) + (dir)	Z	-	-	-	-	*	*	*	*	-
ADD A,ear	2	3	1	0	byte (A) (A) + (ear)	Z	-	-	-	-	*	*	*	*	-
ADD A,eam	2+	4 + (a)	0	(b)	byte (A) (A) + (eam)	Z	-	-	-	-	*	*	*	*	-
ADD ear,A	2	3	2	0	byte (ear) (ear) + (A)	-	-	-	-	-	*	*	*	*	-
ADD eam,A	2+	5 + (a)	0	2 × (b)	byte (eam) (eam) + (A)	Z	-	-	-	-	*	*	*	*	*
ADDC A	1	2	0	0	byte (A) (AH) + (AL) + (C)	Z	-	-	-	-	*	*	*	*	-
ADDC A,ear	2	3	1	0	byte (A) (A) + (ear) + (C)	Z	-	-	-	-	*	*	*	*	-
ADDC A,eam	2+	4 + (a)	0	(b)	byte (A) (A) + (eam) + (C)	Z	-	-	-	-	*	*	*	*	-
ADDC A	1	3	0	0	byte (A) (AH) + (AL) + (C) (10 進数)	Z	-	-	-	-	*	*	*	*	-
SUB A,#imm8	2	2	0	0	byte (A) (A) - imm8	Z	-	-	-	-	*	*	*	*	-
SUB A,dir	2	5	0	(b)	byte (A) (A) - (dir)	Z	-	-	-	-	*	*	*	*	-
SUB A,ear	2	3	1	0	byte (A) (A) - (ear)	Z	-	-	-	-	*	*	*	*	-
SUB A,eam	2+	4 + (a)	0	(b)	byte (A) (A) - (eam)	Z	-	-	-	-	*	*	*	*	-
SUB ear,A	2	3	2	0	byte (ear) (ear) - (A)	-	-	-	-	-	*	*	*	*	-
SUB eam,A	2+	5 + (a)	0	2 × (b)	byte (eam) (eam) - (A)	-	-	-	-	-	*	*	*	*	*
SUBC A	1	2	0	0	byte (A) (AH) - (AL) - (C)	Z	-	-	-	-	*	*	*	*	-
SUBC A,ear	2	3	1	0	byte (A) (A) - (ear) - (C)	Z	-	-	-	-	*	*	*	*	-
SUBC A,eam	2+	4 + (a)	0	(b)	byte (A) (A) - (eam) - (C)	Z	-	-	-	-	*	*	*	*	-
SUBDC A	1	3	0	0	byte (A) (AH) - (AL) - (C) (10 進数)	Z	-	-	-	-	*	*	*	*	-
ADDW A	1	2	0	0	word (A) (AH) + (AL)	-	-	-	-	-	*	*	*	*	-
ADDW A,ear	2	3	1	0	word (A) (A) + (ear)	-	-	-	-	-	*	*	*	*	-
ADDW A,eam	2+	4+(a)	0	(c)	word (A) (A) + (eam)	-	-	-	-	-	*	*	*	*	-
ADDW A,#imm16	3	2	0	0	word (A) (A) + imm16	-	-	-	-	-	*	*	*	*	-
ADDW ear,A	2	3	2	0	word (ear) (ear) + (A)	-	-	-	-	-	*	*	*	*	-
ADDW eam,A	2+	5+(a)	0	2 × (c)	word (eam) (eam) + (A)	-	-	-	-	-	*	*	*	*	*
ADDCW A,ear	2	3	1	0	word (A) (A) + (ear) + (C)	-	-	-	-	-	*	*	*	*	-
ADDCW A,eam	2+	4+(a)	0	(c)	word (A) (A) + (eam) + (C)	-	-	-	-	-	*	*	*	*	-
SUBW A	1	2	0	0	word (A) (AH) - (AL)	-	-	-	-	-	*	*	*	*	-
SUBW A,ear	2	3	1	0	word (A) (A) - (ear)	-	-	-	-	-	*	*	*	*	-
SUBW A,eam	2+	4+(a)	0	(c)	word (A) (A) - (eam)	-	-	-	-	-	*	*	*	*	-
SUBW A,#imm16	3	2	0	0	word (A) (A) - imm16	-	-	-	-	-	*	*	*	*	-
SUBW ear,A	2	3	2	0	word (ear) (ear) - (A)	-	-	-	-	-	*	*	*	*	-
SUBW eam,A	2+	5+(a)	0	2 × (c)	word (eam) (eam) - (A)	-	-	-	-	-	*	*	*	*	*
SUBCW A,ear	2	3	1	0	word (A) (A) - (ear) - (C)	-	-	-	-	-	*	*	*	*	-
SUBCW A,eam	2+	4+(a)	0	(c)	word (A) (A) - (eam) - (C)	-	-	-	-	-	*	*	*	*	-
ADDL A,ear	2	6	2	0	long (A) (A) + (ear)	-	-	-	-	-	*	*	*	*	-
ADDL A,eam	2+	7+(a)	0	(d)	long (A) (A) + (eam)	-	-	-	-	-	*	*	*	*	-
ADDL A,#imm32	5	4	0	0	long (A) (A) + imm32	-	-	-	-	-	*	*	*	*	-
SUBL A,ear	2	6	2	0	long (A) (A) - (ear)	-	-	-	-	-	*	*	*	*	-
SUBL A,eam	2+	7+(a)	0	(d)	long (A) (A) - (eam)	-	-	-	-	-	*	*	*	*	-
SUBL A,#imm32	5	4	0	0	long (A) (A) - imm32	-	-	-	-	-	*	*	*	*	-

(注意事項) 表中の (a) ~ (d) は、表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-4 増減算命令 (バイト・ワード・ロングワード) 12 命令

ニーモニック		#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
INC	ear	2	3	2	0	byte (ear) <-- (ear) + 1	-	-	-	-	-	*	*	*	-	-
INC	eam	2+	5+(a)	0	2 × (b)	byte (eam) (eam) + 1	-	-	-	-	-	*	*	*	-	*
DEC	ear	2	3	2	0	byte (ear) (ear) - 1	-	-	-	-	-	*	*	*	-	-
DEC	eam	2+	5+(a)	0	2 × (b)	byte (eam) (eam) - 1	-	-	-	-	-	*	*	*	-	*
INCW	ear	2	3	2	0	word (ear) (ear) + 1	-	-	-	-	-	*	*	*	-	-
INCW	eam	2+	5+(a)	0	2 × (c)	word (eam) (eam) + 1	-	-	-	-	-	*	*	*	-	*
DECW	ear	2	3	2	0	word (ear) (ear) - 1	-	-	-	-	-	*	*	*	-	-
DECW	eam	2+	5+(a)	0	2 × (c)	word (eam) (eam) - 1	-	-	-	-	-	*	*	*	-	*
INCL	ear	2	7	4	0	long (ear) (ear) + 1	-	-	-	-	-	*	*	*	-	-
INCL	eam	2+	9+(a)	0	2 × (d)	long (eam) (eam) + 1	-	-	-	-	-	*	*	*	-	*
DECL	ear	2	7	4	0	long (ear) (ear) - 1	-	-	-	-	-	*	*	*	-	-
DECL	eam	2+	9+(a)	0	2 × (d)	long (eam) (eam) - 1	-	-	-	-	-	*	*	*	-	*

(注意事項) 表中の (a) ~ (d) は、表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-5 比較命令 (バイト・ワード・ロングワード) 11 命令

ニーモニック		#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
CMP	A	1	1	0	0	byte (AH) - (AL)	-	-	-	-	-	*	*	*	*	-
CMP	A,ear	2	2	1	0	byte (A) - (ear)	-	-	-	-	-	*	*	*	*	-
CMP	A,eam	2+	3+(a)	0	(b)	byte (A) - (eam)	-	-	-	-	-	*	*	*	*	-
CMP	A,#imm8	2	2	0	0	byte (A) - imm8	-	-	-	-	-	*	*	*	*	-
CMPW	A	1	1	0	0	word (AH) - (AL)	-	-	-	-	-	*	*	*	*	-
CMPW	A,ear	2	2	1	0	word (A) - (ear)	-	-	-	-	-	*	*	*	*	-
CMPW	A,eam	2+	3+(a)	0	(c)	word (A) - (eam)	-	-	-	-	-	*	*	*	*	-
CMPW	A,#imm16	3	2	0	0	word (A) - imm16	-	-	-	-	-	*	*	*	*	-
CMPL	A,ear	2	6	2	0	long (A) - (ear)	-	-	-	-	-	*	*	*	*	-
CMPL	A,eam	2+	7+(a)	0	(d)	long (A) - (eam)	-	-	-	-	-	*	*	*	*	-
CMPL	A,#imm32	5	3	0	0	long (A) - imm32	-	-	-	-	-	*	*	*	*	-

(注意事項) 表中の (a) ~ (d) は、表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-6 符号なし乗除算命令 (ワード・ロングワード) 11 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
DIVU A	1	*1	0	0	word (AH) / byte (AL) 商 byte (AL) 余り byte (AH)	-	-	-	-	-	-	-	*	*	-
DIVU A,ear	2	*2	1	0	word (A) / byte (ear) 商 byte (A) 余り byte (ear)	-	-	-	-	-	-	-	*	*	-
DIVU A,eam	2+	*3	0	*6	word (A) / byte (eam) 商 byte (A) 余り byte (eam)	-	-	-	-	-	-	-	*	*	-
DIVUW A,ear	2	*4	1	0	long (A) / word (ear) 商 word (A) 余り word (ear)	-	-	-	-	-	-	-	*	*	-
DIVUW A,eam	2+	*5	0	*7	long (A) / word (eam) 商 word (A) 余り word (eam)	-	-	-	-	-	-	-	*	*	-
MULU A	1	*8	0	0	byte (AH) * byte (AL) word (A)	-	-	-	-	-	-	-	-	-	-
MULU A,ear	2	*9	1	0	byte (A) * byte (ear) word (A)	-	-	-	-	-	-	-	-	-	-
MULU A,eam	2+	*10	0	(b)	byte (A) * byte (eam) word (A)	-	-	-	-	-	-	-	-	-	-
MULUW A	1	*11	0	0	word (AH) * word (AL) Long (A)	-	-	-	-	-	-	-	-	-	-
MULUW A,ear	2	*12	1	0	word (A) * word (ear) Long (A)	-	-	-	-	-	-	-	-	-	-
MULUW A,eam	2+	*13	0	(c)	word (A) * word (eam) Long (A)	-	-	-	-	-	-	-	-	-	-

*1: ゼロディバイトの場合, 3 オーバフローの場合, 7 正常の場合, 15
 *2: ゼロディバイトの場合, 4 オーバフローの場合, 8 正常の場合, 16
 *3: ゼロディバイトの場合, 6+(a) オーバフローの場合, 9+(a) 正常の場合, 19+(a)
 *4: ゼロディバイトの場合, 4 オーバフローの場合, 7 正常の場合, 22
 *5: ゼロディバイトの場合, 6+(a) オーバフローの場合, 8+(a) 正常の場合, 26+(a)
 *6: ゼロディバイト, またはオーバフローの場合, (b) 正常の場合, 2 × (b)
 *7: ゼロディバイト, またはオーバフローの場合, (c) 正常の場合, 2 × (c)
 *8: byte(AH) がゼロの場合, 3 byte(AH) がゼロでない場合, 7
 *9: byte(ear) がゼロの場合, 4 byte(ear) がゼロでない場合, 8
 *10: byte(eam) がゼロの場合, 5+(a) byte(eam) がゼロでない場合, 9+(a)
 *11: word(AH) がゼロの場合, 3 word(AH) がゼロでない場合, 11
 *12: word(ear) がゼロの場合, 4 word(ear) がゼロでない場合, 12
 *13: word(eam) がゼロの場合, 5+(a) word(eam) がゼロでない場合, 13+(a)
 (注意事項) 表中の (a),(b),(c) は, 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-7 符号付き乗除算命令 (ワード・ロングワード) 11 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
DIV A	2	*1	0	0	word (AH) / byte (AL) 商 --> byte (AL) 余り byte (AH)	Z	-	-	-	-	-	-	*	*	-
DIV A,ear	2	*2	1	0	word (A) / byte (ear) 商 --> byte (A) 余り byte (ear)	Z	-	-	-	-	-	-	*	*	-
DIV A,eam	2+	*3	0	*6	word (A) / byte (eam) 商 byte (A) 余り byte (eam)	Z	-	-	-	-	-	-	*	*	-
DIVW A,ear	2	*4	1	0	long (A) / word (ear) 商 --> word (A) 余り word (ear)	-	-	-	-	-	-	-	*	*	-
DIVW A,eam	2+	*5	0	*7	long (A) / word (eam) 商 word (A) 余り word (eam)	-	-	-	-	-	-	-	*	*	-
MUL A	2	*8	0	0	byte (AH) * byte (AL) word (A)	-	-	-	-	-	-	-	-	-	-
MUL A,ear	2	*9	1	0	byte (A) * byte (ear) word (A)	-	-	-	-	-	-	-	-	-	-
MUL A,eam	2+	*10	0	(b)	byte (A) * byte (eam) word (A)	-	-	-	-	-	-	-	-	-	-
MULW A	2	*11	0	0	word (AH) * word (AL) Long (A)	-	-	-	-	-	-	-	-	-	-
MULW A,ear	2	*12	1	0	word (A) * word (ear) Long (A)	-	-	-	-	-	-	-	-	-	-
MULW A,eam	2+	*13	0	(c)	word (A) * word (eam) Long (A)	-	-	-	-	-	-	-	-	-	-

*1: ゼロディバイトの場合, 3 オーバフローの場合, 8 or 18 正常の場合, 18

*2: ゼロディバイトの場合, 4 オーバフローの場合, 11 or 22 正常の場合, 23

*3: ゼロディバイトの場合, 5+(a) オーバフローの場合 12+(a) or 23+(a) 正常の場合, 24+(a)

*4: 被除数が正: ゼロディバイトの場合, 4 オーバフローの場合, 12 or 30 正常の場合, 31

被除数が負: ゼロディバイトの場合, 4 オーバフローの場合, 12 or 31 正常の場合, 32

*5: 被除数が正: ゼロディバイトの場合, 5+(a) オーバフローの場合, 12+(a) or 31+(a) 正常の場合, 32+(a)

被除数が負: ゼロディバイトの場合, 5+(a) オーバフローの場合, 13+(a) or 32+(a) 正常の場合, 33+(a)

*6: ゼロディバイト, またはオーバフローの場合, (b) 正常の場合, 2 × (b)

*7: ゼロディバイト, またはオーバフローの場合, (c) 正常の場合, 2 × (c)

*8: byte(AH) がゼロの場合, 3 結果が正の場合, 12 結果が負の場合, 13

*9: byte(ear) がゼロの場合, 4 結果が正の場合, 13 結果が負の場合, 14

*10: byte(eam) がゼロの場合, 5+(a) 結果が正の場合, 14+(a) 結果が負の場合, 15+(a)

*11: word(AH) がゼロの場合, 3 結果が正の場合, 16 結果が負の場合, 19

*12: word(ear) がゼロの場合, 4 結果が正の場合, 17 結果が負の場合, 20

*13: word(eam) がゼロの場合, 5+(a) 結果が正の場合, 18+(a) 結果が負の場合, 21+(a)

(注意事項) ・ DIV, DIVW 命令でオーバフロー発生時の実行サイクル数に 2 種類ありますが, 演算前と演算後に検出する場合があります。

DIV, DIVW 命令でオーバフロー発生時には, AL の内容は破壊されます。

・ 表中の (a),(b),(c) は, 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-8 論理 1 命令 (バイト・ワード) 39 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
AND A,#imm8	2	2	0	0	byte (A) (A) and imm8	-	-	-	-	-	*	*	R	-	-
AND A,ear	2	3	1	0	byte (A) (A) and (ear)	-	-	-	-	-	*	*	R	-	-
AND A,eam	2+	4+(a)	0	(b)	byte (A) (A) and (eam)	-	-	-	-	-	*	*	R	-	-
AND ear,A	2	3	2	0	byte (ear) (ear) and (A)	-	-	-	-	-	*	*	R	-	-
AND eam,A	2+	5+(a)	0	2 × (b)	byte (eam) (eam) and (A)	-	-	-	-	-	*	*	R	-	*
OR A,#imm8	2	2	0	0	byte (A) (A) or imm8	-	-	-	-	-	*	*	R	-	-
OR A,ear	2	3	1	0	byte (A) (A) or (ear)	-	-	-	-	-	*	*	R	-	-
OR A,eam	2+	4+(a)	0	(b)	byte (A) (A) or (eam)	-	-	-	-	-	*	*	R	-	-
OR ear,A	2	3	2	0	byte (ear) (ear) or (A)	-	-	-	-	-	*	*	R	-	-
OR eam,A	2+	5+(a)	0	2 × (b)	byte (eam) (eam) or (A)	-	-	-	-	-	*	*	R	-	*
XOR A,#imm8	2	2	0	0	byte (A) (A) xor imm8	-	-	-	-	-	*	*	R	-	-
XOR A,ear	2	3	1	0	byte (A) (A) xor (ear)	-	-	-	-	-	*	*	R	-	-
XOR A,eam	2+	4+(a)	0	(b)	byte (A) (A) xor (eam)	-	-	-	-	-	*	*	R	-	-
XOR ear,A	2	3	2	0	byte (ear) (ear) xor (A)	-	-	-	-	-	*	*	R	-	-
XOR eam,A	2+	5+(a)	0	2 × (b)	byte (eam) (eam) xor (A)	-	-	-	-	-	*	*	R	-	*
NOT A	1	2	0	0	byte (A) not (A)	-	-	-	-	-	*	*	R	-	-
NOT ear	2	3	2	0	byte (ear) not (ear)	-	-	-	-	-	*	*	R	-	-
NOT eam	2+	5+(a)	0	2 × (b)	byte (eam) not (eam)	-	-	-	-	-	*	*	R	-	*
ANDW A	1	2	0	0	word (A) (AH) and (A)	-	-	-	-	-	*	*	R	-	-
ANDW A,#imm16	3	2	0	0	word (A) (A) and imm16	-	-	-	-	-	*	*	R	-	-
ANDW A,ear	2	3	1	0	word (A) (A) and (ear)	-	-	-	-	-	*	*	R	-	-
ANDW A,eam	2+	4+(a)	0	(c)	word (A) (A) and (eam)	-	-	-	-	-	*	*	R	-	-
ANDW ear,A	2	3	2	0	word (ear) (ear) and (A)	-	-	-	-	-	*	*	R	-	-
ANDW eam,A	2+	5+(a)	0	2 × (c)	word (eam) (eam) and (A)	-	-	-	-	-	*	*	R	-	*
ORW A	1	2	0	0	word (A) (AH) or (A)	-	-	-	-	-	*	*	R	-	-
ORW A,#imm16	3	2	0	0	word (A) (A) or imm16	-	-	-	-	-	*	*	R	-	-
ORW A,ear	2	3	1	0	word (A) (A) or (ear)	-	-	-	-	-	*	*	R	-	-
ORW A,eam	2+	4+(a)	0	(c)	word (A) (A) or (eam)	-	-	-	-	-	*	*	R	-	-
ORW ear,A	2	3	2	0	word (ear) (ear) or (A)	-	-	-	-	-	*	*	R	-	-
ORW eam,A	2+	5+(a)	0	2 × (c)	word (eam) (eam) or (A)	-	-	-	-	-	*	*	R	-	*
XORW A	1	2	0	0	word (A) (AH) xor (A)	-	-	-	-	-	*	*	R	-	-
XORW A,#imm16	3	2	0	0	word (A) (A) xor imm16	-	-	-	-	-	*	*	R	-	-
XORW A,ear	2	3	1	0	word (A) (A) xor (ear)	-	-	-	-	-	*	*	R	-	-
XORW A,eam	2+	4+(a)	0	(c)	word (A) (A) xor (eam)	-	-	-	-	-	*	*	R	-	-
XORW ear,A	2	3	2	0	word (ear) (ear) xor (A)	-	-	-	-	-	*	*	R	-	-
XORW eam,A	2+	5+(a)	0	2 × (c)	word (eam) (eam) xor (A)	-	-	-	-	-	*	*	R	-	*
NOTW A	1	2	0	0	word (A) not (A)	-	-	-	-	-	*	*	R	-	-
NOTW ear	2	3	2	0	word (ear) not (ear)	-	-	-	-	-	*	*	R	-	-
NOTW eam	2+	5+(a)	0	2 × (c)	word (eam) not (eam)	-	-	-	-	-	*	*	R	-	*

(注意事項) 表中の (a), (b), (c) は , 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-9 論理 2 演算命令 (ロングワード) 6 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
ANDL A,ear	2	6	2	0	long (A) (A) and (ear)	-	-	-	-	-	*	*	R	-	-
ANDL A,eam	2+	7+(a)	0	(d)	long (A) (A) and (eam)	-	-	-	-	-	*	*	R	-	-
ORL A,ear	2	6	2	0	long (A) (A) or (ear)	-	-	-	-	-	*	*	R	-	-
ORL A,eam	2+	7+(a)	0	(d)	long (A) (A) or (eam)	-	-	-	-	-	*	*	R	-	-
XORL A,ear	2	6	2	0	long (A) (A) xor (ear)	-	-	-	-	-	*	*	R	-	-
XORL A,eam	2+	7+(a)	0	(d)	long (A) (A) xor (eam)	-	-	-	-	-	*	*	R	-	-

(注意事項) 表中の (a), (d) は, 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-10 符号反転命令 (バイト・ワード) 6 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
NEG A	1	2	0	0	byte (A) 0 - (A)	X	-	-	-	-	*	*	*	*	-
NEG ear	2	3	2	0	byte (ear) 0 - (ear)	-	-	-	-	-	*	*	*	*	-
NEG eam	2+	5+(a)	0	2 × (b)	byte (eam) 0 - (eam)	-	-	-	-	-	*	*	*	*	*
NEGW A	1	2	0	0	word (A) 0 - (A)	-	-	-	-	-	*	*	*	*	-
NEGW ear	2	3	2	0	word (ear) 0 - (ear)	-	-	-	-	-	*	*	*	*	-
NEGW eam	2+	5+(a)	0	2 × (c)	word (eam) 0 - (eam)	-	-	-	-	-	*	*	*	*	*

(注意事項) 表中の (a), (b), (c) は, 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-11 ノーマライズ命令 (ロングワード) 1 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
NRML A,R0	2	*1	1	0	long (A) 最初に "1" が立っていた所まで左 シフトする byte (R0) その場合のシフト回数	-	-	-	-	-	-	*	-	-	-

*1: アキュムレータがすべて "0" であった場合は, 4 "0" 以外であった場合は, 6+(R0)

表 C.8-12 シフト命令 (バイト・ワード・ロングワード) 18 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
RORC A	2	2	0	0	byte (A) 右ローテートキャリー付	-	-	-	-	-	*	*	-	*	-
ROLC A	2	2	0	0	byte (A) 左ローテートキャリー付	-	-	-	-	-	*	*	-	*	-
RORC ear	2	3	2	0	byte (ear) 右ローテートキャリー付	-	-	-	-	-	*	*	-	*	-
RORC eam	2+	5+(a)	0	2 × (b)	byte (eam) 右ローテートキャリー付	-	-	-	-	-	*	*	-	*	*
ROLC ear	2	3	2	0	byte (ear) 左ローテートキャリー付	-	-	-	-	-	*	*	-	*	-
ROLC eam	2+	5+(a)	0	2 × (b)	byte (eam) 左ローテートキャリー付	-	-	-	-	-	*	*	-	*	*
ASR A,R0	2	*1	1	0	byte (A) 算術右バレルシフト (A,R0)	-	-	-	-	*	*	*	-	*	-
LSR A,R0	2	*1	1	0	byte (A) ロジカル右バレルシフト (A,R0)	-	-	-	-	*	*	*	-	*	-
LSL A,R0	2	*1	1	0	byte (A) ロジカル左バレルシフト (A,R0)	-	-	-	-	-	*	*	-	*	-
ASRW A	1	2	0	0	word (A) 算術右シフト (A,1 ビット)	-	-	-	-	*	*	*	-	*	-
LSRW A/SHRW A	1	2	0	0	word (A) ロジカル右シフト (A,1 ビット)	-	-	-	-	*	R	*	-	*	-
LSLW A/SHLW A	1	2	0	0	word (A) ロジカル左シフト (A,1 ビット)	-	-	-	-	-	*	*	-	*	-
ASRW A,R0	2	*1	1	0	word (A) 算術右バレルシフト (A,R0)	-	-	-	-	*	*	*	-	*	-
LSRW A,R0	2	*1	1	0	word (A) ロジカル右バレルシフト (A,R0)	-	-	-	-	*	*	*	-	*	-
LSLW A,R0	2	*1	1	0	word (A) ロジカル左バレルシフト (A,R0)	-	-	-	-	-	*	*	-	*	-
ASRL A,R0	2	*2	1	0	long (A) 算術右バレルシフト (A,R0)	-	-	-	-	*	*	*	-	*	-
LSRL A,R0	2	*2	1	0	long (A) ロジカル右バレルシフト (A,R0)	-	-	-	-	*	*	*	-	*	-
LSLL A,R0	2	*2	1	0	long (A) <- ロジカル左バレルシフト (A,R0)	-	-	-	-	-	*	*	-	*	-

*1 : R0 が "0" であった場合 , 6 "0" 以外であった場合は , 5+(R0)

*2 : R0 が "0" であった場合 , 6 "0" 以外であった場合は , 6+(R0)

(注意事項) 表中の (a),(b) は , 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-13 分岐命令 1 31 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
BZ/BEQ rel	2	*1	0	0	(Z) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BNZ/BNE rel	2	*1	0	0	(Z) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BC/BLO rel	2	*1	0	0	(C) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BNC/BHS rel	2	*1	0	0	(C) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BN rel	2	*1	0	0	(N) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BP rel	2	*1	0	0	(N) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BV rel	2	*1	0	0	(V) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BNV rel	2	*1	0	0	(V) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BT rel	2	*1	0	0	(T) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BNT rel	2	*1	0	0	(T) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BLT rel	2	*1	0	0	(V) xor (N) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BGE rel	2	*1	0	0	(V) xor (N) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BLE rel	2	*1	0	0	((V) xor (N)) or (Z) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BGT rel	2	*1	0	0	((V) xor (N)) or (Z) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BLS rel	2	*1	0	0	(C) or (Z) = 1 の場合分岐	-	-	-	-	-	-	-	-	-	-
BHI rel	2	*1	0	0	(C) or (Z) = 0 の場合分岐	-	-	-	-	-	-	-	-	-	-
BRA rel	2	*1	0	0	無条件分岐	-	-	-	-	-	-	-	-	-	-
JMP @A	1	2	0	0	word (PC) (A)	-	-	-	-	-	-	-	-	-	-
JMP addr16	3	3	0	0	word (PC) addr16	-	-	-	-	-	-	-	-	-	-
JMP @ear	2	3	1	0	word (PC) (ear)	-	-	-	-	-	-	-	-	-	-
JMP @eam	2+	4+(a)	0	(c)	word (PC) (eam)	-	-	-	-	-	-	-	-	-	-
JMPP @ear 注 1)	2	5	2	0	word (PC) (ear), (PCB) (ear+2)	-	-	-	-	-	-	-	-	-	-
JMPP @eam 注 1)	2+	6+(a)	0	(d)	word (PC) (eam), (PCB) (eam+2)	-	-	-	-	-	-	-	-	-	-
JMPP addr24	4	4	0	0	word (PC) ad24 0-15, (PCB) ad24 16-23	-	-	-	-	-	-	-	-	-	-
CALL @ear 注 2)	2	6	1	(c)	word (PC) (ear)	-	-	-	-	-	-	-	-	-	-
CALL @eam 注 2)	2+	7+(a)	0	2 × (c)	word (PC) (eam)	-	-	-	-	-	-	-	-	-	-
CALL addr16 注 3)	3	6	0	(c)	word (PC) addr16	-	-	-	-	-	-	-	-	-	-
CALLV #vct4 注 3)	1	7	0	2 × (c)	ベクタコール命令	-	-	-	-	-	-	-	-	-	-
CALLP @ear 注 4)	2	10	2	2 × (c)	word (PC) (ear)0-15, (PCB) (ear)16-23	-	-	-	-	-	-	-	-	-	-
CALLP @eam 注 4)	2+	11+(a)	0	*2	word (PC) (eam)0-15, (PCB) (eam)16-23	-	-	-	-	-	-	-	-	-	-
CALLP addr24 注 5)	4	10	0	2 × (c)	word (PC) addr0-15, (PCB) addr16-23	-	-	-	-	-	-	-	-	-	-

*1 : 分岐が行われる場合 , 4 行われない場合 , 3

*2 : 3 × (c)+(b)

注 1) 分岐先アドレスの読み (word)

注 2) W : スタックへの退避 (word), R : 分岐先アドレスの読み (word)

注 3) スタックへの退避 (word)

注 4) W : スタックへの退避 (long), R : 分岐先アドレスの読み (long)

注 5) スタックへの退避 (long)

(注意事項) 表中の (a) ~ (d) は、表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-14 分岐命令 2 19 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
CBNE A,#imm8,rel	3	*1	0	0	byte (A) imm8 で分岐	-	-	-	-	-	*	*	*	*	-
CWBNE A,#imm16,rel	4	*1	0	0	word (A) imm16 で分岐	-	-	-	-	-	*	*	*	*	-
CBNE ear,#imm8,rel	4	*2	1	0	byte (ear) imm8 で分岐	-	-	-	-	-	*	*	*	*	-
CBNE eam,#imm8,rel 注 1)	4+	*3	0	(b)	byte (eam) imm8 で分岐	-	-	-	-	-	*	*	*	*	-
CWBNE ear,#imm16,rel	5	*4	1	0	word (ear) imm16 で分岐	-	-	-	-	-	*	*	*	*	-
CWBNE eam,#imm16,rel 注 1)	5+	*3	0	(c)	word (eam) imm16 で分岐	-	-	-	-	-	*	*	*	*	-
DBNZ ear,rel	3	*5	2	0	byte (ear) = (ear) - 1, (ear) 0 で分岐	-	-	-	-	-	*	*	*	-	-
DBNZ eam,rel	3+	*6	2	2 × (b)	byte (eam) = (eam) - 1, (eam) 0 で分岐	-	-	-	-	-	*	*	*	-	*
DWBNZ ear,rel	3	*5	2	0	word (ear) = (ear) - 1, (ear) 0 で分岐	-	-	-	-	-	*	*	*	-	-
DWBNZ eam,rel	3+	*6	2	2 × (c)	word (eam) = (eam) - 1, (eam) 0 で分岐	-	-	-	-	-	*	*	*	-	*
INT #vct8	2	20	0	8 × (c)	ソフトウェア割込み	-	-	R	S	-	-	-	-	-	-
INT addr16	3	16	0	6 × (c)	ソフトウェア割込み	-	-	R	S	-	-	-	-	-	-
INTP addr24	4	17	0	6 × (c)	ソフトウェア割込み	-	-	R	S	-	-	-	-	-	-
INT9	1	20	0	8 × (c)	ソフトウェア割込み	-	-	R	S	-	-	-	-	-	-
RETI	1	*8	0	*7	割込みからの復帰	-	-	*	*	*	*	*	*	*	-
LINK #imm8	2	6	0	(c)	関数の入口で、旧フレームポインタをスタックに保存し、新フレームポインタの設定、およびローカルポインタの領域を確保する	-	-	-	-	-	-	-	-	-	-
UNLINK	1	5	0	(c)	関数の出口で、旧フレームポインタをスタックから復帰させる	-	-	-	-	-	-	-	-	-	-
RET 注 2)	1	4	0	(c)	サブルーチンからの復帰	-	-	-	-	-	-	-	-	-	-
RETP 注 3)	1	6	0	(d)	サブルーチンからの復帰	-	-	-	-	-	-	-	-	-	-

*1: 分岐が行われる場合, 5 行われない場合, 4
 *2: 分岐が行われる場合, 13 行われない場合, 12
 *3: 分岐が行われる場合, 7+(a) 行われない場合, 6+(a)
 *4: 分岐が行われる場合, 8 行われない場合, 7
 *5: 分岐が行われる場合, 7 行われない場合, 6
 *6: 分岐が行われる場合, 8+(a) 行われない場合, 7+(a)
 *7: 次の割込み要求へ分岐する場合, 3 × (b)+2 × (c) 今の割込みから復帰の場合, 6 × (c)
 *8: 次の割込みへ分岐する場合, 15 今の割込みから復帰する場合, 17
 注 1) CBNE / CWBNE 命令では, RWj+ のアドレッシングモードは, 使用しないでください。
 注 2) スタックからの復帰 (word)
 注 3) スタックからの復帰 (long)
 (注意事項) 表中の (a) ~ (d) は, 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-15 その他制御命令 (バイト・ワード・ロングワード) 28 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
PUSHW A	1	4	0	(c)	word (SP) (SP) - 2, ((SP)) (A)	-	-	-	-	-	-	-	-	-	-
PUSHW AH	1	4	0	(c)	word (SP) (SP) - 2, ((SP)) (AH)	-	-	-	-	-	-	-	-	-	-
PUSHW PS	1	4	0	(c)	word (SP) (SP) - 2, ((SP)) (PS)	-	-	-	-	-	-	-	-	-	-
PUSHW rlst	2	*3	*5	*4	(SP) (SP) - 2n, ((SP)) (rlst)	-	-	-	-	-	-	-	-	-	-
POPW A	1	3	0	(c)	word (A) ((SP)), (SP) (SP) + 2	-	*	-	-	-	-	-	-	-	-
POPW AH	1	3	0	(c)	word (AH) ((SP)), (SP) (SP) + 2	-	-	-	-	-	-	-	-	-	-
POPW PS	1	4	0	(c)	word (PS) ((SP)), (SP) (SP) + 2	-	-	*	*	*	*	*	*	*	-
POPW rlst	2	*2	*5	*4	(rlst) ((SP)), (SP) (SP) + 2n	-	-	-	-	-	-	-	-	-	-
JCTX @A	1	14	0	6 × (c)	コンテキストスイッチ命令	-	-	*	*	*	*	*	*	*	-
AND CCR,#imm8	2	3	0	0	byte (CCR) (CCR) and imm8	-	-	*	*	*	*	*	*	*	-
OR CCR,#imm8	2	3	0	0	byte (CCR) (CCR) or imm8	-	-	*	*	*	*	*	*	*	-
MOV RP,#imm8	2	2	0	0	byte (RP) imm8	-	-	-	-	-	-	-	-	-	-
MOV ILM,#imm8	2	2	0	0	byte (ILM) imm8	-	-	-	-	-	-	-	-	-	-
MOVEA RWi,ear	2	3	1	0	word (RWi) ear	-	-	-	-	-	-	-	-	-	-
MOVEA RWi,eam	2+	2+(a)	1	0	word (RWi) eam	-	-	-	-	-	-	-	-	-	-
MOVEA A,ear	2	1	0	0	word (A) ear	-	*	-	-	-	-	-	-	-	-
MOVEA A,eam	2+	1+(a)	0	0	word (A) eam	-	*	-	-	-	-	-	-	-	-
ADDSP #imm8	2	3	0	0	word (SP) (SP) + ext(imm8)	-	-	-	-	-	-	-	-	-	-
ADDSP #imm16	3	3	0	0	word (SP) (SP) + imm16	-	-	-	-	-	-	-	-	-	-
MOV A,brg1	2	*1	0	0	byte (A) (brg1)	Z	*	-	-	-	*	*	-	-	-
MOV brg2,A	2	1	0	0	byte (brg2) (A)	-	-	-	-	-	*	*	-	-	-
NOP	1	1	0	0	ノーオペレーション	-	-	-	-	-	-	-	-	-	-
ADB	1	1	0	0	AD 空間アクセス用プリフィックスコード	-	-	-	-	-	-	-	-	-	-
DTB	1	1	0	0	DT 空間アクセス用プリフィックスコード	-	-	-	-	-	-	-	-	-	-
PCB	1	1	0	0	PC 空間アクセス用プリフィックスコード	-	-	-	-	-	-	-	-	-	-
SPB	1	1	0	0	SP 空間アクセス用プリフィックスコード	-	-	-	-	-	-	-	-	-	-
NCC	1	1	0	0	フラグ無変化用プリフィックスコード	-	-	-	-	-	-	-	-	-	-
CMR	1	1	0	0	コモンレジスタバンク用プリフィックス	-	-	-	-	-	-	-	-	-	-

*1 : PCB,ADB,SSB,USB 1 ステート

DTB,DPR 2 ステート

*2 : 7+3 × (POP する回数)+2 × (POP する最後のレジスタ番号), RLST=0(転送レジスタ無し) の場合 7

*3 : 29+3 × (PUSH する回数)-3 × (PUSH する最後のレジスタ番号), RLST=0(転送レジスタ無し) の場合 8

*4 : (POP する回数) × (c), または (PUSH する回数) × (c)

*5 : (POP する回数), または (PUSH する回数)

(注意事項) 表中の (a),(c) は, 表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-16 ビット操作命令 21 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
MOVB A,dir:bp	3	5	0	(b)	byte (A) (dir:bp)b	Z	*	-	-	-	*	*	-	-	-
MOVB A,addr16:bp	4	5	0	(b)	byte (A) (addr16:bp)b	Z	*	-	-	-	*	*	-	-	-
MOVB A,io:bp	3	4	0	(b)	byte (A) (io:bp)b	Z	*	-	-	-	*	*	-	-	-
MOVB dir:bp,A	3	7	0	2×(b)	bit (dir:bp)b (A)	-	-	-	-	-	*	*	-	-	*
MOVB addr16:bp,A	4	7	0	2×(b)	bit (addr16:bp)b (A)	-	-	-	-	-	*	*	-	-	*
MOVB io:bp,A	3	6	0	2×(b)	bit (io:bp)b (A)	-	-	-	-	-	*	*	-	-	*
SETB dir:bp	3	7	0	2×(b)	bit (dir:bp)b 1	-	-	-	-	-	-	-	-	-	*
SETB addr16:bp	4	7	0	2×(b)	bit (addr16:bp)b 1	-	-	-	-	-	-	-	-	-	*
SETB io:bp	3	7	0	2×(b)	bit (io:bp)b 1	-	-	-	-	-	-	-	-	-	*
CLRB dir:bp	3	7	0	2×(b)	bit (dir:bp)b 0	-	-	-	-	-	-	-	-	-	*
CLRB addr16:bp	4	7	0	2×(b)	bit (addr16:bp)b 0	-	-	-	-	-	-	-	-	-	*
CLRB io:bp	3	7	0	2×(b)	bit (io:bp)b 0	-	-	-	-	-	-	-	-	-	*
BBC dir:bp,rel	4	*1	0	(b)	(dir:bp) b = 0 の場合分岐	-	-	-	-	-	-	*	-	-	-
BBC addr16:bp,rel	5	*1	0	(b)	(addr16:bp) b = 0 の場合分岐	-	-	-	-	-	-	*	-	-	-
BBC io:bp,rel	4	*2	0	(b)	(io:bp) b = 0 の場合分岐	-	-	-	-	-	-	*	-	-	-
BBS dir:bp,rel	4	*1	0	(b)	(dir:bp) b = 1 の場合分岐	-	-	-	-	-	-	*	-	-	-
BBS addr16:bp,rel	5	*1	0	(b)	(addr16:bp) b = 1 の場合分岐	-	-	-	-	-	-	*	-	-	-
BBS io:bp,rel	4	*2	0	(b)	(io:bp) b = 1 の場合分岐	-	-	-	-	-	-	*	-	-	-
SBBS addr16:bp,rel	5	*3	0	2×(b)	(addr16:bp) b = 1 で分岐, bit (addr16:bp) b 1	-	-	-	-	-	-	*	-	-	*
WBTS io:bp	3	*4	0	*5	(io:bp) b = 1 になるまで待つ	-	-	-	-	-	-	-	-	-	-
WBTC io:bp	3	*4	0	*5	(io:bp) b = 0 になるまで待つ	-	-	-	-	-	-	-	-	-	-

*1：分岐が行われる場合，8 分岐が行われない場合，7

*2：分岐が行われる場合，7 分岐が行われない場合，6

*3：条件成立の場合，10 未成立の場合，9

*4：不定回数

*5：条件が成立するまで

(注意事項) 表中の (b) は，表 C.5-1 および表 C.5-2 を参照してください。

表 C.8-17 アキュムレータ操作命令 (バイト・ワード) 6 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
SWAP	1	3	0	0	byte (A)0-7 (A)8-15	-	-	-	-	-	-	-	-	-	-
SWAPW	1	2	0	0	word (AH) (AL)	-	*	-	-	-	-	-	-	-	-
EXT	1	1	0	0	byte 符号拡張	X	-	-	-	-	*	*	-	-	-
EXTW	1	2	0	0	word 符号拡張	-	X	-	-	-	*	*	-	-	-
ZEXT	1	1	0	0	byte ゼロ拡張	Z	-	-	-	-	R	*	-	-	-
ZEXTW	1	1	0	0	word ゼロ拡張	-	Z	-	-	-	R	*	-	-	-

表 C.8-18 スtring命令 10 命令

ニーモニック	#	~	RG	B	オペレーション	LH	AH	I	S	T	N	Z	V	C	RMW
MOVS / MOVSI	2	*2	*5	*3	byte 転送 @AH+ @AL+, カウンタ = RW0	-	-	-	-	-	-	-	-	-	-
MOVSD	2	*2	*5	*3	byte 転送 @AH- @AL-, カウンタ = RW0	-	-	-	-	-	-	-	-	-	-
SCEQ / SCEQI	2	*1	*8	*4	byte 検索 @AH+ AL, カウンタ = RW0	-	-	-	-	-	*	*	*	*	-
SCEQD	2	*1	*8	*4	byte 検索 @AH- AL, カウンタ = RW0	-	-	-	-	-	*	*	*	*	-
FILS / FILSI	2	6m+6	*8	*3	byte 充填 @AH+ AL, カウンタ = RW0	-	-	-	-	-	*	*	-	-	-
MOVSW / MOVSWI	2	*2	*5	*6	word 転送 @AH+ @AL+, カウンタ = RW0	-	-	-	-	-	-	-	-	-	-
MOVSWD	2	*2	*5	*6	word 転送 @AH- @AL-, カウンタ = RW0	-	-	-	-	-	-	-	-	-	-
SCWEQ / SCWEQI	2	*1	*8	*7	word 検索 @AH+ - AL, カウンタ = RW0	-	-	-	-	-	*	*	*	*	-
SCWEQD	2	*1	*8	*7	word 検索 @AH- - AL, カウンタ = RW0	-	-	-	-	-	*	*	*	*	-
FILSW / FILSWI	2	6m+6	*8	*6	word 充填 @AH+ AL, カウンタ = RW0	-	-	-	-	-	*	*	-	-	-

*1 : RW0 が "0" の場合, 5 カウントアウトの場合, $4+7 \times (RW0)$ 一致した場合, $7n+5$

*2 : RW0 が "0" の場合, 5 "0" 以外の場合, $4+8 \times (RW0)$

*3 : (b) $\times (RW0) + (b) \times (RW0)$ ソースとデスティネーションで異なる領域をアクセスする場合には, それぞれ別々に (b) の項目を算出してください。

*4 : (b)+n

*5 : $2 \times (b) \times (RW0)$

*6 : (c) $\times (RW0) + (c) \times (RW0)$ ソースとデスティネーションで異なる領域をアクセスする場合には, それぞれ別々に (c) の項目を算出してください。

*7 : (c) $\times n$

*8 : (b) $\times (RW0)$

(注意事項) ・ m : RW0 値 (カウンタ値)

n : ループした回数

・ 表中の (b), (c) は, 表 C.5-1 および表 C.5-2 を参照してください。

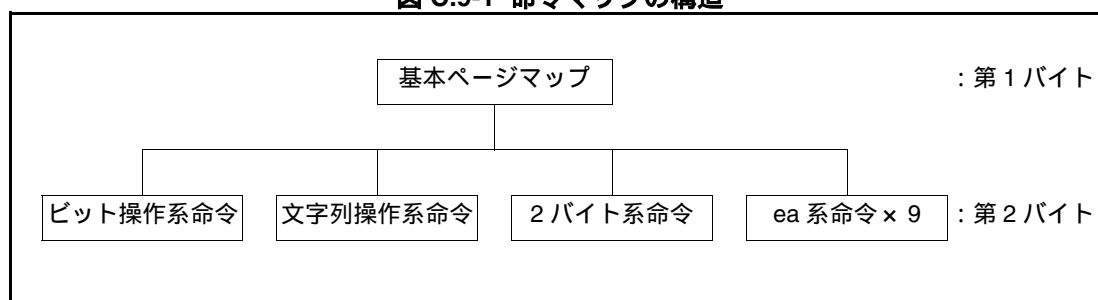
C.9 命令マップ

F²MC-16LX の命令コードは、1 ～ 2 バイトで構成されていますので、命令マップは複数のページで構成されています。

F²MC-16LX の命令マップを示します。

■ 命令マップの構造

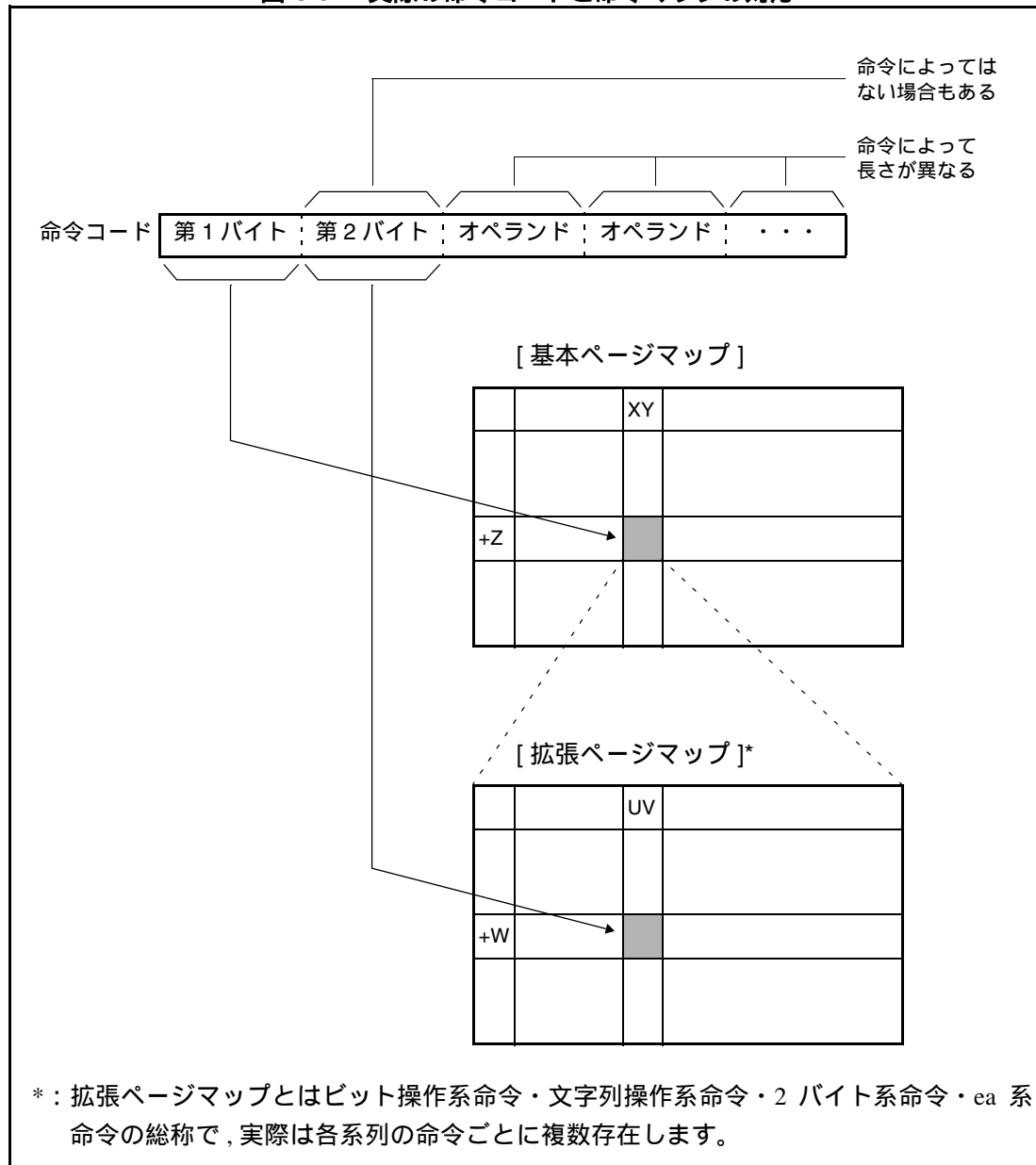
図 C.9-1 命令マップの構造



1 バイトで終わる命令 (NOP 命令など) は基本ページで完結し、2 バイト必要とする命令 (MOVS 命令など) は第 1 バイトを参照した段階で第 2 バイト用のマップの存在がわかり、第 2 バイト用のマップを参照して続く 1 バイトを調べることができます。

実際の命令コードと命令マップの対応を図 C.9-2 に示します。

図 C.9-2 実際の命令コードと命令マップの対応



命令コードの例を表 C.9-1 に示します。

表 C.9-1 命令コードの例

命令	第1バイト (基本ページマップより)	第2バイト (拡張ページマップより)
NOP	$00 + 0 = 00$	
AND A, #8	$30 + 4 = 34$	
MOV A, ADB	$60 + F = 6F$	$00 + 0 = 00$
@RW2+d8, #8, rel	$70 + 0 = 70$	$F0 + 2 = F2$

表 C.9-2 基本ページマップ

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	NOP	CMR	ADD A, dir	ADD A, #8	MOV A, dir	MOV A, io	BRA rel	ea 系命令 その 1	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOV A, Ri	MOVX A, @RWi+d8	MOV A, #4	CALL #4	BZ/BEQ rel
+1	INT9	NCC	SUB A, dir	SUB A, #8	MOV dir, A	MOV io, A	JMP @A	ea 系命令 その 2								BNZ/BNE rel
+2	ADDCA	SUBCA	ADDC A	SUBC A	MOV A, #8	MOV A, addr16	JMP addr16	ea 系命令 その 3								BC/BLO rel
+3	NEG A	JCTX @A	CMP A	CMP A, #8	MOVX A, #8	MOV addr16, A	JMPP addr24	ea 系命令 その 4								BNC/BHS rel
+4	PCB	EXT	AND CCR, #8	AND A, #8	MOV dir, #8	MOV io, #8	CALL addr16	ea 系命令 その 5								BN rel
+5	DTB	ZEXT	OR CCR, #8	OR A, #8	MOVX A, dir	MOVX A, io	CALLP addr24	ea 系命令 その 6								BP rel
+6	ADB	SWAP	DIVU A	XOR A, #8	MOVW A, SP	MOVW io, #16	RETP	ea 系命令 その 7								BV rel
+7	SPB	ADDSP #8	MULU #8	NOT A	MOVW SP, A	MOVX A, addr16	RET	ea 系命令 その 8								BNV rel
+8	LINK #imm8	ADDL A, #32	ADDW A	ADDW A, #16	MOVW A, dir	MOVW A, io	INT #vct8	ea 系命令 その 9	MOVW A, RWi	MOVW RWi, A	MOVW RWi, #16	MOVX A, @RWi+d8	MOVW @RWi+d8 @RWi+d8,A			BT rel
+9	UNLINK	SUBL A, #32	SUBW A	SUBW A, #16	MOVW dir, A	MOVW io, A	INT	MOVEA RWi, ea								BNT rel
+A	MOV RP, #8	MOV ILM, #8	CBNE A, #8, rel	CWNE A, #16, rel	MOVW A, #16	MOVW A, addr16	INTP addr24	MOV Ri, ea								BLT rel
+B	NEGW A	CMPL A, #32	CMPW A	CMPW A, #16	MOVL A, #32	MOVW addr16, A	RETI	MOVW RWi, ea								BGE rel
+C	LSLW A	EXTW	ANDW A	ANDW A, #16	PUSHW A	POPW A	ビット操 作系命令	MOV ea, Ri								BLE rel
+D		ZEXTW	ORW A	ORW A, #16	PUSHW AH	POPW AH		MOVW ea, RWi								BGT rel
+E	ASRW A	SWAPW	XORW A	XORW A, #16	PUSHW PS	POPW PS	文字列操 作系命令	XCH Ri, ea								BLS rel
+F	LSRW A	ADDSP #16	MULW A	NOTW A	PUSHW r1st	POPW r1st	2 バイト 系命令	XCHW RWi, ea								BHI rel

表 C.9-3 ビット操作系命令マップ (第 1 バイト =6C_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	MOV B, A, io:bp		MOV B, io:bp, A		CLRB, io:bp		SETB, io:bp		BBC, io:bp, rel		BBS, io:bp, rel		WBTS, io:bp		WBTC, io:bp	
+1																
+2																
+3																
+4																
+5																
+6																
+7																
+8	MOV B, A, dir:bp	MOV B, A, addr16:bp	MOV B, dir:bp, A	MOV B, addr16:bp, A	CLRB, dir:bp	CLRB, addr16:bp	SETB, dir:bp	SETB, addr16:bp	BBC, dir:bp, rel	BBC, addr16:bp, rel	BBS, dir:bp, rel	BBS, addr16:bp, rel				SBS, addr16:bp
+9																
+A																
+B																
+C																
+D																
+E																
+F																

表 C.9-4 文字列操作系命令マップ (第 1 バイト =6E_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	MOVSI, MOVSD, MOVSWI, MOVSWD PCB, PCB								SCWEQI PCB	SCEQD PCB	SCWEQI PCB	SCWEQD PCB	FILSI PCB			
+1	PCB, DTB								PCB DTB	PCB DTB	PCB DTB	PCB DTB	PCB DTB			
+2	PCB, ADB								PCB ADB	PCB ADB	PCB ADB	PCB ADB	PCB ADB			
+3	PCB, SPB								PCB SPB	PCB SPB	PCB SPB	PCB SPB	PCB SPB			
+4	DTB, PCB															
+5	DTB, DTB															
+6	DTB, ADB															
+7	DTB, SPB															
+8	ADB, PCB															
+9	ADB, DTB															
+A	ADB, ADB															
+B	ADB, SPB															
+C	SPB, PCB															
+D	SPB, DTB															
+E	SPB, ADB															
+F	SPB, SPB															

表 C.9-5 2 バイト系命令マップ (第 1 バイト = 6F_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	MOV A, DTB	MOV DTB, A	MOVX A, @RL0+d8	MOV @RL0+d8, A	MOV @RL0+d8, A											
+1	MOV A, ADB	MOV ADB, A														
+2	MOV A, SSB	MOV SSB, A	MOVX A, @RL1+d8	MOV @RL1+d8, A	MOV @RL1+d8, A											
+3	MOV A, USB	MOV USB, A														
+4	MOV A, DPR	MOV DPR, A	MOVX A, @RL2+d8	MOV @RL2+d8, A	MOV @RL2+d8, A											
+5	MOV A, @A	MOV @AL, AH														
+6	MOV A, PCB	MOV A, @A	MOVX A, @RL3+d8	MOV @RL3+d8, A	MOV @RL3+d8, A											
+7	ROL A	ROL A														
+8				MOVW @RL0+d8, A	MOVW A, @RL0+d8			MUL A								
+9								MULW A								
+A				MOVW @RL1+d8, A	MOVW A, @RL1+d8			DIVU A								
+B																
+C	LSLW A, R0	LSLL A, R0	LSL A, R0	MOVW @RL2+d8, A	MOVW A, @RL2+d8											
+D	MOVW A, @A	MOVW @AL, AH	NRML A, R0													
+E	ASRW A, R0	ASRL A, R0	ASR A, R0	MOVW @RL3+d8, A	MOVW A, @RL3+d8											
+F	LSRW A, R0	LSRL A, R0	LSR A, R0													

表 C.9-6 ea 系命令 その 1 (第 1 バイト = 70_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
					CWBNEL	CWBNEL									CBNE↓	
+0	ADDL A, A, RL0, @RW0+d8	SUBL A, A, RL0, @RW0+d8	SUBL A, A, RL0, @RW0+d8	SUBL A, A, RL0, @RW0+d8	RW0, @RW0+d8 #16, rel	CMPL A, A, RL0, @RW0+d8	CMPL A, A, RL0, @RW0+d8	CMPL A, A, RL0, @RW0+d8	ANDL A, A, RL0, @RW0+d8	ANDL A, A, RL0, @RW0+d8	ORL A, A, RL0, @RW0+d8	ORL A, A, RL0, @RW0+d8	XORL A, A, RL0, @RW0+d8	XORL A, A, RL0, @RW0+d8	R0, @RW0+d8 #8, rel	R0, @RW0+d8 #8, rel
+1	ADDL A, A, RL0, @RW1+d8	SUBL A, A, RL0, @RW1+d8	SUBL A, A, RL0, @RW1+d8	SUBL A, A, RL0, @RW1+d8	RW1, @RW1+d8 #16, rel	CMPL A, A, RL0, @RW1+d8	CMPL A, A, RL0, @RW1+d8	CMPL A, A, RL0, @RW1+d8	ANDL A, A, RL0, @RW1+d8	ANDL A, A, RL0, @RW1+d8	ORL A, A, RL0, @RW1+d8	ORL A, A, RL0, @RW1+d8	XORL A, A, RL0, @RW1+d8	XORL A, A, RL0, @RW1+d8	R1, @RW1+d8 #8, rel	R1, @RW1+d8 #8, rel
+2	ADDL A, A, RL1, @RW2+d8	SUBL A, A, RL1, @RW2+d8	SUBL A, A, RL1, @RW2+d8	SUBL A, A, RL1, @RW2+d8	RW2, @RW2+d8 #16, rel	CMPL A, A, RL1, @RW2+d8	CMPL A, A, RL1, @RW2+d8	CMPL A, A, RL1, @RW2+d8	ANDL A, A, RL1, @RW2+d8	ANDL A, A, RL1, @RW2+d8	ORL A, A, RL1, @RW2+d8	ORL A, A, RL1, @RW2+d8	XORL A, A, RL1, @RW2+d8	XORL A, A, RL1, @RW2+d8	R2, @RW2+d8 #8, rel	R2, @RW2+d8 #8, rel
+3	ADDL A, A, RL1, @RW3+d8	SUBL A, A, RL1, @RW3+d8	SUBL A, A, RL1, @RW3+d8	SUBL A, A, RL1, @RW3+d8	RW3, @RW3+d8 #16, rel	CMPL A, A, RL1, @RW3+d8	CMPL A, A, RL1, @RW3+d8	CMPL A, A, RL1, @RW3+d8	ANDL A, A, RL1, @RW3+d8	ANDL A, A, RL1, @RW3+d8	ORL A, A, RL1, @RW3+d8	ORL A, A, RL1, @RW3+d8	XORL A, A, RL1, @RW3+d8	XORL A, A, RL1, @RW3+d8	R3, @RW3+d8 #8, rel	R3, @RW3+d8 #8, rel
+4	ADDL A, A, RL2, @RW4+d8	SUBL A, A, RL2, @RW4+d8	SUBL A, A, RL2, @RW4+d8	SUBL A, A, RL2, @RW4+d8	RW4, @RW4+d8 #16, rel	CMPL A, A, RL2, @RW4+d8	CMPL A, A, RL2, @RW4+d8	CMPL A, A, RL2, @RW4+d8	ANDL A, A, RL2, @RW4+d8	ANDL A, A, RL2, @RW4+d8	ORL A, A, RL2, @RW4+d8	ORL A, A, RL2, @RW4+d8	XORL A, A, RL2, @RW4+d8	XORL A, A, RL2, @RW4+d8	R4, @RW4+d8 #8, rel	R4, @RW4+d8 #8, rel
+5	ADDL A, A, RL2, @RW5+d8	SUBL A, A, RL2, @RW5+d8	SUBL A, A, RL2, @RW5+d8	SUBL A, A, RL2, @RW5+d8	RW5, @RW5+d8 #16, rel	CMPL A, A, RL2, @RW5+d8	CMPL A, A, RL2, @RW5+d8	CMPL A, A, RL2, @RW5+d8	ANDL A, A, RL2, @RW5+d8	ANDL A, A, RL2, @RW5+d8	ORL A, A, RL2, @RW5+d8	ORL A, A, RL2, @RW5+d8	XORL A, A, RL2, @RW5+d8	XORL A, A, RL2, @RW5+d8	R5, @RW5+d8 #8, rel	R5, @RW5+d8 #8, rel
+6	ADDL A, A, RL3, @RW6+d8	SUBL A, A, RL3, @RW6+d8	SUBL A, A, RL3, @RW6+d8	SUBL A, A, RL3, @RW6+d8	RW6, @RW6+d8 #16, rel	CMPL A, A, RL3, @RW6+d8	CMPL A, A, RL3, @RW6+d8	CMPL A, A, RL3, @RW6+d8	ANDL A, A, RL3, @RW6+d8	ANDL A, A, RL3, @RW6+d8	ORL A, A, RL3, @RW6+d8	ORL A, A, RL3, @RW6+d8	XORL A, A, RL3, @RW6+d8	XORL A, A, RL3, @RW6+d8	R6, @RW6+d8 #8, rel	R6, @RW6+d8 #8, rel
+7	ADDL A, A, RL3, @RW7+d8	SUBL A, A, RL3, @RW7+d8	SUBL A, A, RL3, @RW7+d8	SUBL A, A, RL3, @RW7+d8	RW7, @RW7+d8 #16, rel	CMPL A, A, RL3, @RW7+d8	CMPL A, A, RL3, @RW7+d8	CMPL A, A, RL3, @RW7+d8	ANDL A, A, RL3, @RW7+d8	ANDL A, A, RL3, @RW7+d8	ORL A, A, RL3, @RW7+d8	ORL A, A, RL3, @RW7+d8	XORL A, A, RL3, @RW7+d8	XORL A, A, RL3, @RW7+d8	R7, @RW7+d8 #8, rel	R7, @RW7+d8 #8, rel
+8	ADDL A, A, @RW0, @RW0+d16	SUBL A, A, @RW0, @RW0+d16	SUBL A, A, @RW0, @RW0+d16	SUBL A, A, @RW0, @RW0+d16	@RW0, @RW0+d16 #16, rel	CMPL A, A, @RW0, @RW0+d16	CMPL A, A, @RW0, @RW0+d16	CMPL A, A, @RW0, @RW0+d16	ANDL A, A, @RW0, @RW0+d16	ANDL A, A, @RW0, @RW0+d16	ORL A, A, @RW0, @RW0+d16	ORL A, A, @RW0, @RW0+d16	XORL A, A, @RW0, @RW0+d16	XORL A, A, @RW0, @RW0+d16	@RW0, @RW0+d16 #8, rel	@RW0, @RW0+d16 #8, rel
+9	ADDL A, A, @RW1, @RW1+d16	SUBL A, A, @RW1, @RW1+d16	SUBL A, A, @RW1, @RW1+d16	SUBL A, A, @RW1, @RW1+d16	@RW1, @RW1+d16 #16, rel	CMPL A, A, @RW1, @RW1+d16	CMPL A, A, @RW1, @RW1+d16	CMPL A, A, @RW1, @RW1+d16	ANDL A, A, @RW1, @RW1+d16	ANDL A, A, @RW1, @RW1+d16	ORL A, A, @RW1, @RW1+d16	ORL A, A, @RW1, @RW1+d16	XORL A, A, @RW1, @RW1+d16	XORL A, A, @RW1, @RW1+d16	@RW1, @RW1+d16 #8, rel	@RW1, @RW1+d16 #8, rel
+A	ADDL A, A, @RW2, @RW2+d16	SUBL A, A, @RW2, @RW2+d16	SUBL A, A, @RW2, @RW2+d16	SUBL A, A, @RW2, @RW2+d16	@RW2, @RW2+d16 #16, rel	CMPL A, A, @RW2, @RW2+d16	CMPL A, A, @RW2, @RW2+d16	CMPL A, A, @RW2, @RW2+d16	ANDL A, A, @RW2, @RW2+d16	ANDL A, A, @RW2, @RW2+d16	ORL A, A, @RW2, @RW2+d16	ORL A, A, @RW2, @RW2+d16	XORL A, A, @RW2, @RW2+d16	XORL A, A, @RW2, @RW2+d16	@RW2, @RW2+d16 #8, rel	@RW2, @RW2+d16 #8, rel
+B	ADDL A, A, @RW3, @RW3+d16	SUBL A, A, @RW3, @RW3+d16	SUBL A, A, @RW3, @RW3+d16	SUBL A, A, @RW3, @RW3+d16	@RW3, @RW3+d16 #16, rel	CMPL A, A, @RW3, @RW3+d16	CMPL A, A, @RW3, @RW3+d16	CMPL A, A, @RW3, @RW3+d16	ANDL A, A, @RW3, @RW3+d16	ANDL A, A, @RW3, @RW3+d16	ORL A, A, @RW3, @RW3+d16	ORL A, A, @RW3, @RW3+d16	XORL A, A, @RW3, @RW3+d16	XORL A, A, @RW3, @RW3+d16	@RW3, @RW3+d16 #8, rel	@RW3, @RW3+d16 #8, rel
+C	ADDL A, A, @RW0+RW7	SUBL A, A, @RW0+RW7	SUBL A, A, @RW0+RW7	SUBL A, A, @RW0+RW7	@RW0+RW7 #16, rel	CMPL A, A, @RW0+RW7	CMPL A, A, @RW0+RW7	CMPL A, A, @RW0+RW7	ANDL A, A, @RW0+RW7	ANDL A, A, @RW0+RW7	ORL A, A, @RW0+RW7	ORL A, A, @RW0+RW7	XORL A, A, @RW0+RW7	XORL A, A, @RW0+RW7	@RW0+RW7 #8, rel	@RW0+RW7 #8, rel
+D	ADDL A, A, @RW1+RW7	SUBL A, A, @RW1+RW7	SUBL A, A, @RW1+RW7	SUBL A, A, @RW1+RW7	@RW1+RW7 #16, rel	CMPL A, A, @RW1+RW7	CMPL A, A, @RW1+RW7	CMPL A, A, @RW1+RW7	ANDL A, A, @RW1+RW7	ANDL A, A, @RW1+RW7	ORL A, A, @RW1+RW7	ORL A, A, @RW1+RW7	XORL A, A, @RW1+RW7	XORL A, A, @RW1+RW7	使用禁止	使用禁止
+E	ADDL A, A, @PC+d16	SUBL A, A, @PC+d16	SUBL A, A, @PC+d16	SUBL A, A, @PC+d16	@PC+d16 #16, rel	CMPL A, A, @PC+d16	CMPL A, A, @PC+d16	CMPL A, A, @PC+d16	ANDL A, A, @PC+d16	ANDL A, A, @PC+d16	ORL A, A, @PC+d16	ORL A, A, @PC+d16	XORL A, A, @PC+d16	XORL A, A, @PC+d16	@PC+d16 #8, rel	@PC+d16 #8, rel
+F	ADDL A, A, @RW3+addr16	SUBL A, A, @RW3+addr16	SUBL A, A, @RW3+addr16	SUBL A, A, @RW3+addr16	addr16 #16, rel	CMPL A, A, @RW3+addr16	CMPL A, A, @RW3+addr16	CMPL A, A, @RW3+addr16	ANDL A, A, @RW3+addr16	ANDL A, A, @RW3+addr16	ORL A, A, @RW3+addr16	ORL A, A, @RW3+addr16	XORL A, A, @RW3+addr16	XORL A, A, @RW3+addr16	使用禁止	使用禁止

表 C.9-7 ea 系命令 その 2 (第 1 バイト = 71_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	JMPP @ RL0, @ RW0+d8	JMPP @ RL0, @ RW0+d8	CALLP @ RL0, @ RW0+d8	CALLP @ RL0, @ RW0+d8	INCL RL0, @ RW0+d8	INCL RL0, @ RW0+d8	DECL RL0, @ RW0+d8	DECL RL0, @ RW0+d8	MOVL A, RL0, @ RW0+d8	MOVL A, RL0, @ RW0+d8	MOVL A, RL0, @ RW0+d8	MOVL A, RL0, @ RW0+d8	MOV R0, #8, @ RW0+d8, #8	MOV R0, #8, @ RW0+d8, #8	MOVEA A, RW0, @ RW0+d8	MOVEA A, RW0, @ RW0+d8
+1	JMPP @ RL0, @ RW1+d8	JMPP @ RL0, @ RW1+d8	CALLP @ RL0, @ RW1+d8	CALLP @ RL0, @ RW1+d8	INCL RL0, @ RW1+d8	INCL RL0, @ RW1+d8	DECL RL0, @ RW1+d8	DECL RL0, @ RW1+d8	MOVL A, RL0, @ RW1+d8	MOVL A, RL0, @ RW1+d8	MOVL A, RL0, @ RW1+d8	MOVL A, RL0, @ RW1+d8	MOV R1, #8, @ RW1+d8, #8	MOV R1, #8, @ RW1+d8, #8	MOVEA A, RW1, @ RW1+d8	MOVEA A, RW1, @ RW1+d8
+2	JMPP @ RL1, @ RW2+d8	JMPP @ RL1, @ RW2+d8	CALLP @ RL1, @ RW2+d8	CALLP @ RL1, @ RW2+d8	INCL RL1, @ RW2+d8	INCL RL1, @ RW2+d8	DECL RL1, @ RW2+d8	DECL RL1, @ RW2+d8	MOVL A, RL1, @ RW2+d8	MOVL A, RL1, @ RW2+d8	MOVL A, RL1, @ RW2+d8	MOVL A, RL1, @ RW2+d8	MOV R2, #8, @ RW2+d8, #8	MOV R2, #8, @ RW2+d8, #8	MOVEA A, RW2, @ RW2+d8	MOVEA A, RW2, @ RW2+d8
+3	JMPP @ RL1, @ RW3+d8	JMPP @ RL1, @ RW3+d8	CALLP @ RL1, @ RW3+d8	CALLP @ RL1, @ RW3+d8	INCL RL1, @ RW3+d8	INCL RL1, @ RW3+d8	DECL RL1, @ RW3+d8	DECL RL1, @ RW3+d8	MOVL A, RL1, @ RW3+d8	MOVL A, RL1, @ RW3+d8	MOVL A, RL1, @ RW3+d8	MOVL A, RL1, @ RW3+d8	MOV R3, #8, @ RW3+d8, #8	MOV R3, #8, @ RW3+d8, #8	MOVEA A, RW3, @ RW3+d8	MOVEA A, RW3, @ RW3+d8
+4	JMPP @ RL2, @ RW4+d8	JMPP @ RL2, @ RW4+d8	CALLP @ RL2, @ RW4+d8	CALLP @ RL2, @ RW4+d8	INCL RL2, @ RW4+d8	INCL RL2, @ RW4+d8	DECL RL2, @ RW4+d8	DECL RL2, @ RW4+d8	MOVL A, RL2, @ RW4+d8	MOVL A, RL2, @ RW4+d8	MOVL A, RL2, @ RW4+d8	MOVL A, RL2, @ RW4+d8	MOV R4, #8, @ RW4+d8, #8	MOV R4, #8, @ RW4+d8, #8	MOVEA A, RW4, @ RW4+d8	MOVEA A, RW4, @ RW4+d8
+5	JMPP @ RL2, @ RW5+d8	JMPP @ RL2, @ RW5+d8	CALLP @ RL2, @ RW5+d8	CALLP @ RL2, @ RW5+d8	INCL RL2, @ RW5+d8	INCL RL2, @ RW5+d8	DECL RL2, @ RW5+d8	DECL RL2, @ RW5+d8	MOVL A, RL2, @ RW5+d8	MOVL A, RL2, @ RW5+d8	MOVL A, RL2, @ RW5+d8	MOVL A, RL2, @ RW5+d8	MOV R5, #8, @ RW5+d8, #8	MOV R5, #8, @ RW5+d8, #8	MOVEA A, RW5, @ RW5+d8	MOVEA A, RW5, @ RW5+d8
+6	JMPP @ RL3, @ RW6+d8	JMPP @ RL3, @ RW6+d8	CALLP @ RL3, @ RW6+d8	CALLP @ RL3, @ RW6+d8	INCL RL3, @ RW6+d8	INCL RL3, @ RW6+d8	DECL RL3, @ RW6+d8	DECL RL3, @ RW6+d8	MOVL A, RL3, @ RW6+d8	MOVL A, RL3, @ RW6+d8	MOVL A, RL3, @ RW6+d8	MOVL A, RL3, @ RW6+d8	MOV R6, #8, @ RW6+d8, #8	MOV R6, #8, @ RW6+d8, #8	MOVEA A, RW6, @ RW6+d8	MOVEA A, RW6, @ RW6+d8
+7	JMPP @ RL3, @ RW7+d8	JMPP @ RL3, @ RW7+d8	CALLP @ RL3, @ RW7+d8	CALLP @ RL3, @ RW7+d8	INCL RL3, @ RW7+d8	INCL RL3, @ RW7+d8	DECL RL3, @ RW7+d8	DECL RL3, @ RW7+d8	MOVL A, RL3, @ RW7+d8	MOVL A, RL3, @ RW7+d8	MOVL A, RL3, @ RW7+d8	MOVL A, RL3, @ RW7+d8	MOV R7, #8, @ RW7+d8, #8	MOV R7, #8, @ RW7+d8, #8	MOVEA A, RW7, @ RW7+d8	MOVEA A, RW7, @ RW7+d8
+8	JMPP @ @ RW0, @ RW0+d16	JMPP @ @ RW0, @ RW0+d16	CALLP @ @ RW0, @ RW0+d16	CALLP @ @ RW0, @ RW0+d16	INCL @ RW0, @ RW0+d16	INCL @ RW0, @ RW0+d16	DECL @ RW0, @ RW0+d16	DECL @ RW0, @ RW0+d16	MOVL A, @ RW0, @ RW0+d16	MOVL A, @ RW0, @ RW0+d16	MOVL A, @ RW0, @ RW0+d16	MOVL A, @ RW0, @ RW0+d16	MOV @ RW0, #8, @ RW0+d16, #8	MOV @ RW0, #8, @ RW0+d16, #8	MOVEA A, @ RW0, @ RW0+d16	MOVEA A, @ RW0, @ RW0+d16
+9	JMPP @ @ RW1, @ RW1+d16	JMPP @ @ RW1, @ RW1+d16	CALLP @ @ RW1, @ RW1+d16	CALLP @ @ RW1, @ RW1+d16	INCL @ RW1, @ RW1+d16	INCL @ RW1, @ RW1+d16	DECL @ RW1, @ RW1+d16	DECL @ RW1, @ RW1+d16	MOVL A, @ RW1, @ RW1+d16	MOVL A, @ RW1, @ RW1+d16	MOVL A, @ RW1, @ RW1+d16	MOVL A, @ RW1, @ RW1+d16	MOV @ RW1, #8, @ RW1+d16, #8	MOV @ RW1, #8, @ RW1+d16, #8	MOVEA A, @ RW1, @ RW1+d16	MOVEA A, @ RW1, @ RW1+d16
+A	JMPP @ @ RW2, @ RW2+d16	JMPP @ @ RW2, @ RW2+d16	CALLP @ @ RW2, @ RW2+d16	CALLP @ @ RW2, @ RW2+d16	INCL @ RW2, @ RW2+d16	INCL @ RW2, @ RW2+d16	DECL @ RW2, @ RW2+d16	DECL @ RW2, @ RW2+d16	MOVL A, @ RW2, @ RW2+d16	MOVL A, @ RW2, @ RW2+d16	MOVL A, @ RW2, @ RW2+d16	MOVL A, @ RW2, @ RW2+d16	MOV @ RW2, #8, @ RW2+d16, #8	MOV @ RW2, #8, @ RW2+d16, #8	MOVEA A, @ RW2, @ RW2+d16	MOVEA A, @ RW2, @ RW2+d16
+B	JMPP @ @ RW3, @ RW3+d16	JMPP @ @ RW3, @ RW3+d16	CALLP @ @ RW3, @ RW3+d16	CALLP @ @ RW3, @ RW3+d16	INCL @ RW3, @ RW3+d16	INCL @ RW3, @ RW3+d16	DECL @ RW3, @ RW3+d16	DECL @ RW3, @ RW3+d16	MOVL A, @ RW3, @ RW3+d16	MOVL A, @ RW3, @ RW3+d16	MOVL A, @ RW3, @ RW3+d16	MOVL A, @ RW3, @ RW3+d16	MOV @ RW3, #8, @ RW3+d16, #8	MOV @ RW3, #8, @ RW3+d16, #8	MOVEA A, @ RW3, @ RW3+d16	MOVEA A, @ RW3, @ RW3+d16
+C	JMPP @ @ RW0+, @ RW0+RW7	JMPP @ @ RW0+, @ RW0+RW7	CALLP @ @ RW0+, @ RW0+RW7	CALLP @ @ RW0+, @ RW0+RW7	INCL @ RW0+, @ RW0+RW7	INCL @ RW0+, @ RW0+RW7	DECL @ RW0+, @ RW0+RW7	DECL @ RW0+, @ RW0+RW7	MOVL A, @ RW0+, @ RW0+RW7	MOVL A, @ RW0+, @ RW0+RW7	MOVL A, @ RW0+, @ RW0+RW7	MOVL A, @ RW0+, @ RW0+RW7	MOV @ RW0+, #8, @ RW0+RW7, #8	MOV @ RW0+, #8, @ RW0+RW7, #8	MOVEA A, @ RW0+, @ RW0+RW7	MOVEA A, @ RW0+, @ RW0+RW7
+D	JMPP @ @ RW1+, @ RW1+RW7	JMPP @ @ RW1+, @ RW1+RW7	CALLP @ @ RW1+, @ RW1+RW7	CALLP @ @ RW1+, @ RW1+RW7	INCL @ RW1+, @ RW1+RW7	INCL @ RW1+, @ RW1+RW7	DECL @ RW1+, @ RW1+RW7	DECL @ RW1+, @ RW1+RW7	MOVL A, @ RW1+, @ RW1+RW7	MOVL A, @ RW1+, @ RW1+RW7	MOVL A, @ RW1+, @ RW1+RW7	MOVL A, @ RW1+, @ RW1+RW7	MOV @ RW1+, #8, @ RW1+RW7, #8	MOV @ RW1+, #8, @ RW1+RW7, #8	MOVEA A, @ RW1+, @ RW1+RW7	MOVEA A, @ RW1+, @ RW1+RW7
+E	JMPP @ @ RW2+, @ PC+d16	JMPP @ @ RW2+, @ PC+d16	CALLP @ @ RW2+, @ PC+d16	CALLP @ @ RW2+, @ PC+d16	INCL @ RW2+, @ PC+d16	INCL @ RW2+, @ PC+d16	DECL @ RW2+, @ PC+d16	DECL @ RW2+, @ PC+d16	MOVL A, @ RW2+, @ PC+d16	MOVL A, @ RW2+, @ PC+d16	MOVL A, @ RW2+, @ PC+d16	MOVL A, @ RW2+, @ PC+d16	MOV @ RW2+, #8, @ PC+d16, #8	MOV @ RW2+, #8, @ PC+d16, #8	MOVEA A, @ RW2+, @ PC+d16	MOVEA A, @ RW2+, @ PC+d16
+F	JMPP @ @ RW3+, @ addr16	JMPP @ @ RW3+, @ addr16	CALLP @ @ RW3+, @ addr16	CALLP @ @ RW3+, @ addr16	INCL @ RW3+, @ addr16	INCL @ RW3+, @ addr16	DECL @ RW3+, @ addr16	DECL @ RW3+, @ addr16	MOVL A, @ RW3+, @ addr16	MOVL A, @ RW3+, @ addr16	MOVL A, @ RW3+, @ addr16	MOVL A, @ RW3+, @ addr16	MOV @ RW3+, #8, @ addr16, #8	MOV @ RW3+, #8, @ addr16, #8	MOVEA A, @ RW3+, @ addr16	MOVEA A, @ RW3+, @ addr16

表 C.9-8 ea 系命令 その 3 (第 1 バイト =72_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	ROL R0, @RW0+d8	ROL R0, @RW0+d8	ROR R0, @RW0+d8	ROR R0, @RW0+d8	INC R0, @RW0+d8	DEC R0, @RW0+d8	DEC R0, @RW0+d8	DEC R0, @RW0+d8	MOV A, R0, @RW0+d8	MOV A, R0, @RW0+d8	MOV R0, A, @RW0+d8	MOV R0, A, @RW0+d8	MOVX A, R0, @RW0+d8	MOVX A, R0, @RW0+d8	XCH A, R0, @RW0+d8	XCH A, R0, @RW0+d8
+1	ROL R1, @RW1+d8	ROL R1, @RW1+d8	ROR R1, @RW1+d8	ROR R1, @RW1+d8	INC R1, @RW1+d8	DEC R1, @RW1+d8	DEC R1, @RW1+d8	DEC R1, @RW1+d8	MOV A, R1, @RW1+d8	MOV A, R1, @RW1+d8	MOV R1, A, @RW1+d8	MOV R1, A, @RW1+d8	MOVX A, R1, @RW1+d8	MOVX A, R1, @RW1+d8	XCH A, R1, @RW1+d8	XCH A, R1, @RW1+d8
+2	ROL R2, @RW2+d8	ROL R2, @RW2+d8	ROR R2, @RW2+d8	ROR R2, @RW2+d8	INC R2, @RW2+d8	DEC R2, @RW2+d8	DEC R2, @RW2+d8	DEC R2, @RW2+d8	MOV A, R2, @RW2+d8	MOV A, R2, @RW2+d8	MOV R2, A, @RW2+d8	MOV R2, A, @RW2+d8	MOVX A, R2, @RW2+d8	MOVX A, R2, @RW2+d8	XCH A, R2, @RW2+d8	XCH A, R2, @RW2+d8
+3	ROL R3, @RW3+d8	ROL R3, @RW3+d8	ROR R3, @RW3+d8	ROR R3, @RW3+d8	INC R3, @RW3+d8	DEC R3, @RW3+d8	DEC R3, @RW3+d8	DEC R3, @RW3+d8	MOV A, R3, @RW3+d8	MOV A, R3, @RW3+d8	MOV R3, A, @RW3+d8	MOV R3, A, @RW3+d8	MOVX A, R3, @RW3+d8	MOVX A, R3, @RW3+d8	XCH A, R3, @RW3+d8	XCH A, R3, @RW3+d8
+4	ROL R4, @RW4+d8	ROL R4, @RW4+d8	ROR R4, @RW4+d8	ROR R4, @RW4+d8	INC R4, @RW4+d8	DEC R4, @RW4+d8	DEC R4, @RW4+d8	DEC R4, @RW4+d8	MOV A, R4, @RW4+d8	MOV A, R4, @RW4+d8	MOV R4, A, @RW4+d8	MOV R4, A, @RW4+d8	MOVX A, R4, @RW4+d8	MOVX A, R4, @RW4+d8	XCH A, R4, @RW4+d8	XCH A, R4, @RW4+d8
+5	ROL R5, @RW5+d8	ROL R5, @RW5+d8	ROR R5, @RW5+d8	ROR R5, @RW5+d8	INC R5, @RW5+d8	DEC R5, @RW5+d8	DEC R5, @RW5+d8	DEC R5, @RW5+d8	MOV A, R5, @RW5+d8	MOV A, R5, @RW5+d8	MOV R5, A, @RW5+d8	MOV R5, A, @RW5+d8	MOVX A, R5, @RW5+d8	MOVX A, R5, @RW5+d8	XCH A, R5, @RW5+d8	XCH A, R5, @RW5+d8
+6	ROL R6, @RW6+d8	ROL R6, @RW6+d8	ROR R6, @RW6+d8	ROR R6, @RW6+d8	INC R6, @RW6+d8	DEC R6, @RW6+d8	DEC R6, @RW6+d8	DEC R6, @RW6+d8	MOV A, R6, @RW6+d8	MOV A, R6, @RW6+d8	MOV R6, A, @RW6+d8	MOV R6, A, @RW6+d8	MOVX A, R6, @RW6+d8	MOVX A, R6, @RW6+d8	XCH A, R6, @RW6+d8	XCH A, R6, @RW6+d8
+7	ROL R7, @RW7+d8	ROL R7, @RW7+d8	ROR R7, @RW7+d8	ROR R7, @RW7+d8	INC R7, @RW7+d8	DEC R7, @RW7+d8	DEC R7, @RW7+d8	DEC R7, @RW7+d8	MOV A, R7, @RW7+d8	MOV A, R7, @RW7+d8	MOV R7, A, @RW7+d8	MOV R7, A, @RW7+d8	MOVX A, R7, @RW7+d8	MOVX A, R7, @RW7+d8	XCH A, R7, @RW7+d8	XCH A, R7, @RW7+d8
+8	ROL @RW0, @RW0+d16	ROL @RW0, @RW0+d16	ROR @RW0, @RW0+d16	ROR @RW0, @RW0+d16	INC @RW0, @RW0+d16	DEC @RW0, @RW0+d16	DEC @RW0, @RW0+d16	DEC @RW0, @RW0+d16	MOV A, @RW0, @RW0+d16	MOV A, @RW0, @RW0+d16	MOV @RW0, A, @RW0+d16	MOV @RW0, A, @RW0+d16	MOVX A, @RW0, @RW0+d16	MOVX A, @RW0, @RW0+d16	XCH A, @RW0, @RW0+d16	XCH A, @RW0, @RW0+d16
+9	ROL @RW1, @RW1+d16	ROL @RW1, @RW1+d16	ROR @RW1, @RW1+d16	ROR @RW1, @RW1+d16	INC @RW1, @RW1+d16	DEC @RW1, @RW1+d16	DEC @RW1, @RW1+d16	DEC @RW1, @RW1+d16	MOV A, @RW1, @RW1+d16	MOV A, @RW1, @RW1+d16	MOV @RW1, A, @RW1+d16	MOV @RW1, A, @RW1+d16	MOVX A, @RW1, @RW1+d16	MOVX A, @RW1, @RW1+d16	XCH A, @RW1, @RW1+d16	XCH A, @RW1, @RW1+d16
+A	ROL @RW2, @RW2+d16	ROL @RW2, @RW2+d16	ROR @RW2, @RW2+d16	ROR @RW2, @RW2+d16	INC @RW2, @RW2+d16	DEC @RW2, @RW2+d16	DEC @RW2, @RW2+d16	DEC @RW2, @RW2+d16	MOV A, @RW2, @RW2+d16	MOV A, @RW2, @RW2+d16	MOV @RW2, A, @RW2+d16	MOV @RW2, A, @RW2+d16	MOVX A, @RW2, @RW2+d16	MOVX A, @RW2, @RW2+d16	XCH A, @RW2, @RW2+d16	XCH A, @RW2, @RW2+d16
+B	ROL @RW3, @RW3+d16	ROL @RW3, @RW3+d16	ROR @RW3, @RW3+d16	ROR @RW3, @RW3+d16	INC @RW3, @RW3+d16	DEC @RW3, @RW3+d16	DEC @RW3, @RW3+d16	DEC @RW3, @RW3+d16	MOV A, @RW3, @RW3+d16	MOV A, @RW3, @RW3+d16	MOV @RW3, A, @RW3+d16	MOV @RW3, A, @RW3+d16	MOVX A, @RW3, @RW3+d16	MOVX A, @RW3, @RW3+d16	XCH A, @RW3, @RW3+d16	XCH A, @RW3, @RW3+d16
+C	ROL @RW0+, @RW0+RW7	ROL @RW0+, @RW0+RW7	ROR @RW0+, @RW0+RW7	ROR @RW0+, @RW0+RW7	INC @RW0+, @RW0+RW7	DEC @RW0+, @RW0+RW7	DEC @RW0+, @RW0+RW7	DEC @RW0+, @RW0+RW7	MOV A, @RW0+, @RW0+RW7	MOV A, @RW0+, @RW0+RW7	MOV @RW0+, A, @RW0+RW7	MOV @RW0+, A, @RW0+RW7	MOVX A, @RW0+, @RW0+RW7	MOVX A, @RW0+, @RW0+RW7	XCH A, @RW0+, @RW0+RW7	XCH A, @RW0+, @RW0+RW7
+D	ROL @RW1+, @RW1+RW7	ROL @RW1+, @RW1+RW7	ROR @RW1+, @RW1+RW7	ROR @RW1+, @RW1+RW7	INC @RW1+, @RW1+RW7	DEC @RW1+, @RW1+RW7	DEC @RW1+, @RW1+RW7	DEC @RW1+, @RW1+RW7	MOV A, @RW1+, @RW1+RW7	MOV A, @RW1+, @RW1+RW7	MOV @RW1+, A, @RW1+RW7	MOV @RW1+, A, @RW1+RW7	MOVX A, @RW1+, @RW1+RW7	MOVX A, @RW1+, @RW1+RW7	XCH A, @RW1+, @RW1+RW7	XCH A, @RW1+, @RW1+RW7
+E	ROL @RW2+, @PC+d16	ROL @RW2+, @PC+d16	ROR @RW2+, @PC+d16	ROR @RW2+, @PC+d16	INC @RW2+, @PC+d16	DEC @RW2+, @PC+d16	DEC @RW2+, @PC+d16	DEC @RW2+, @PC+d16	MOV A, @RW2+, @PC+d16	MOV A, @RW2+, @PC+d16	MOV @RW2+, A, @PC+d16	MOV @RW2+, A, @PC+d16	MOVX A, @RW2+, @PC+d16	MOVX A, @RW2+, @PC+d16	XCH A, @RW2+, @PC+d16	XCH A, @RW2+, @PC+d16
+F	ROL @RW3+, addr16	ROL @RW3+, addr16	ROR @RW3+, addr16	ROR @RW3+, addr16	INC @RW3+, addr16	DEC @RW3+, addr16	DEC @RW3+, addr16	DEC @RW3+, addr16	MOV A, @RW3+, addr16	MOV A, @RW3+, addr16	MOV @RW3+, A, addr16	MOV @RW3+, A, addr16	MOVX A, @RW3+, addr16	MOVX A, @RW3+, addr16	XCH A, @RW3+, addr16	XCH A, @RW3+, addr16

表 C.9-9 ea 系命令 その 4 (第 1 バイト = 73_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	JMP @RW0, @@RW0+d8	JMP @RW0, @@RW0+d8	CALL RW0, @@RW0+d8	CALL RW0, @@RW0+d8	INCW RW0, @@RW0+d8	INCW RW0, @@RW0+d8	DECW RW0, @@RW0+d8	DECW RW0, @@RW0+d8	MOVW A, RW0, @@RW0+d8	MOVW A, RW0, @@RW0+d8	MOVW RW0, #16, @RW0+d8, #16	MOVW RW0, #16, @RW0+d8, #16	MOVW RW0, #16, @RW0+d8, #16	MOVW RW0, #16, @RW0+d8, #16	XCHW A, RW0, @@RW0+d8	XCHW A, RW0, @@RW0+d8
+1	JMP @RW1, @@RW1+d8	JMP @RW1, @@RW1+d8	CALL RW1, @@RW1+d8	CALL RW1, @@RW1+d8	INCW RW1, @@RW1+d8	INCW RW1, @@RW1+d8	DECW RW1, @@RW1+d8	DECW RW1, @@RW1+d8	MOVW A, RW1, @@RW1+d8	MOVW A, RW1, @@RW1+d8	MOVW RW1, #16, @RW1+d8, #16	MOVW RW1, #16, @RW1+d8, #16	MOVW RW1, #16, @RW1+d8, #16	MOVW RW1, #16, @RW1+d8, #16	XCHW A, RW1, @@RW1+d8	XCHW A, RW1, @@RW1+d8
+2	JMP @RW2, @@RW2+d8	JMP @RW2, @@RW2+d8	CALL RW2, @@RW2+d8	CALL RW2, @@RW2+d8	INCW RW2, @@RW2+d8	INCW RW2, @@RW2+d8	DECW RW2, @@RW2+d8	DECW RW2, @@RW2+d8	MOVW A, RW2, @@RW2+d8	MOVW A, RW2, @@RW2+d8	MOVW RW2, #16, @RW2+d8, #16	MOVW RW2, #16, @RW2+d8, #16	MOVW RW2, #16, @RW2+d8, #16	MOVW RW2, #16, @RW2+d8, #16	XCHW A, RW2, @@RW2+d8	XCHW A, RW2, @@RW2+d8
+3	JMP @RW3, @@RW3+d8	JMP @RW3, @@RW3+d8	CALL RW3, @@RW3+d8	CALL RW3, @@RW3+d8	INCW RW3, @@RW3+d8	INCW RW3, @@RW3+d8	DECW RW3, @@RW3+d8	DECW RW3, @@RW3+d8	MOVW A, RW3, @@RW3+d8	MOVW A, RW3, @@RW3+d8	MOVW RW3, #16, @RW3+d8, #16	MOVW RW3, #16, @RW3+d8, #16	MOVW RW3, #16, @RW3+d8, #16	MOVW RW3, #16, @RW3+d8, #16	XCHW A, RW3, @@RW3+d8	XCHW A, RW3, @@RW3+d8
+4	JMP @RW4, @@RW4+d8	JMP @RW4, @@RW4+d8	CALL RW4, @@RW4+d8	CALL RW4, @@RW4+d8	INCW RW4, @@RW4+d8	INCW RW4, @@RW4+d8	DECW RW4, @@RW4+d8	DECW RW4, @@RW4+d8	MOVW A, RW4, @@RW4+d8	MOVW A, RW4, @@RW4+d8	MOVW RW4, #16, @RW4+d8, #16	MOVW RW4, #16, @RW4+d8, #16	MOVW RW4, #16, @RW4+d8, #16	MOVW RW4, #16, @RW4+d8, #16	XCHW A, RW4, @@RW4+d8	XCHW A, RW4, @@RW4+d8
+5	JMP @RW5, @@RW5+d8	JMP @RW5, @@RW5+d8	CALL RW5, @@RW5+d8	CALL RW5, @@RW5+d8	INCW RW5, @@RW5+d8	INCW RW5, @@RW5+d8	DECW RW5, @@RW5+d8	DECW RW5, @@RW5+d8	MOVW A, RW5, @@RW5+d8	MOVW A, RW5, @@RW5+d8	MOVW RW5, #16, @RW5+d8, #16	MOVW RW5, #16, @RW5+d8, #16	MOVW RW5, #16, @RW5+d8, #16	MOVW RW5, #16, @RW5+d8, #16	XCHW A, RW5, @@RW5+d8	XCHW A, RW5, @@RW5+d8
+6	JMP @RW6, @@RW6+d8	JMP @RW6, @@RW6+d8	CALL RW6, @@RW6+d8	CALL RW6, @@RW6+d8	INCW RW6, @@RW6+d8	INCW RW6, @@RW6+d8	DECW RW6, @@RW6+d8	DECW RW6, @@RW6+d8	MOVW A, RW6, @@RW6+d8	MOVW A, RW6, @@RW6+d8	MOVW RW6, #16, @RW6+d8, #16	MOVW RW6, #16, @RW6+d8, #16	MOVW RW6, #16, @RW6+d8, #16	MOVW RW6, #16, @RW6+d8, #16	XCHW A, RW6, @@RW6+d8	XCHW A, RW6, @@RW6+d8
+7	JMP @RW7, @@RW7+d8	JMP @RW7, @@RW7+d8	CALL RW7, @@RW7+d8	CALL RW7, @@RW7+d8	INCW RW7, @@RW7+d8	INCW RW7, @@RW7+d8	DECW RW7, @@RW7+d8	DECW RW7, @@RW7+d8	MOVW A, RW7, @@RW7+d8	MOVW A, RW7, @@RW7+d8	MOVW RW7, #16, @RW7+d8, #16	MOVW RW7, #16, @RW7+d8, #16	MOVW RW7, #16, @RW7+d8, #16	MOVW RW7, #16, @RW7+d8, #16	XCHW A, RW7, @@RW7+d8	XCHW A, RW7, @@RW7+d8
+8	JMP @RW0, @@RW0+d16	JMP @RW0, @@RW0+d16	CALL @RW0, @@RW0+d16	CALL @RW0, @@RW0+d16	INCW @RW0, @@RW0+d16	INCW @RW0, @@RW0+d16	DECW @RW0, @@RW0+d16	DECW @RW0, @@RW0+d16	MOVW A, @RW0, @@RW0+d16	MOVW A, @RW0, @@RW0+d16	MOVW @RW0, #16, @RW0+d16, #16	MOVW @RW0, #16, @RW0+d16, #16	MOVW @RW0, #16, @RW0+d16, #16	MOVW @RW0, #16, @RW0+d16, #16	XCHW A, @RW0, @@RW0+d16	XCHW A, @RW0, @@RW0+d16
+9	JMP @RW1, @@RW1+d16	JMP @RW1, @@RW1+d16	CALL @RW1, @@RW1+d16	CALL @RW1, @@RW1+d16	INCW @RW1, @@RW1+d16	INCW @RW1, @@RW1+d16	DECW @RW1, @@RW1+d16	DECW @RW1, @@RW1+d16	MOVW A, @RW1, @@RW1+d16	MOVW A, @RW1, @@RW1+d16	MOVW @RW1, #16, @RW1+d16, #16	MOVW @RW1, #16, @RW1+d16, #16	MOVW @RW1, #16, @RW1+d16, #16	MOVW @RW1, #16, @RW1+d16, #16	XCHW A, @RW1, @@RW1+d16	XCHW A, @RW1, @@RW1+d16
+A	JMP @RW2, @@RW2+d16	JMP @RW2, @@RW2+d16	CALL @RW2, @@RW2+d16	CALL @RW2, @@RW2+d16	INCW @RW2, @@RW2+d16	INCW @RW2, @@RW2+d16	DECW @RW2, @@RW2+d16	DECW @RW2, @@RW2+d16	MOVW A, @RW2, @@RW2+d16	MOVW A, @RW2, @@RW2+d16	MOVW @RW2, #16, @RW2+d16, #16	MOVW @RW2, #16, @RW2+d16, #16	MOVW @RW2, #16, @RW2+d16, #16	MOVW @RW2, #16, @RW2+d16, #16	XCHW A, @RW2, @@RW2+d16	XCHW A, @RW2, @@RW2+d16
+B	JMP @RW3, @@RW3+d16	JMP @RW3, @@RW3+d16	CALL @RW3, @@RW3+d16	CALL @RW3, @@RW3+d16	INCW @RW3, @@RW3+d16	INCW @RW3, @@RW3+d16	DECW @RW3, @@RW3+d16	DECW @RW3, @@RW3+d16	MOVW A, @RW3, @@RW3+d16	MOVW A, @RW3, @@RW3+d16	MOVW @RW3, #16, @RW3+d16, #16	MOVW @RW3, #16, @RW3+d16, #16	MOVW @RW3, #16, @RW3+d16, #16	MOVW @RW3, #16, @RW3+d16, #16	XCHW A, @RW3, @@RW3+d16	XCHW A, @RW3, @@RW3+d16
+C	JMP @RW0+, @@RW0+RW7	JMP @RW0+, @@RW0+RW7	CALL @RW0+, @@RW0+RW7	CALL @RW0+, @@RW0+RW7	INCW @RW0+, @@RW0+RW7	INCW @RW0+, @@RW0+RW7	DECW @RW0+, @@RW0+RW7	DECW @RW0+, @@RW0+RW7	MOVW A, @RW0+, @@RW0+RW7	MOVW A, @RW0+, @@RW0+RW7	MOVW @RW0+, #16, @RW0+RW7, #16	MOVW @RW0+, #16, @RW0+RW7, #16	MOVW @RW0+, #16, @RW0+RW7, #16	MOVW @RW0+, #16, @RW0+RW7, #16	XCHW A, @RW0+, @@RW0+RW7	XCHW A, @RW0+, @@RW0+RW7
+D	JMP @RW1+, @@RW1+RW7	JMP @RW1+, @@RW1+RW7	CALL @RW1+, @@RW1+RW7	CALL @RW1+, @@RW1+RW7	INCW @RW1+, @@RW1+RW7	INCW @RW1+, @@RW1+RW7	DECW @RW1+, @@RW1+RW7	DECW @RW1+, @@RW1+RW7	MOVW A, @RW1+, @@RW1+RW7	MOVW A, @RW1+, @@RW1+RW7	MOVW @RW1+, #16, @RW1+RW7, #16	MOVW @RW1+, #16, @RW1+RW7, #16	MOVW @RW1+, #16, @RW1+RW7, #16	MOVW @RW1+, #16, @RW1+RW7, #16	XCHW A, @RW1+, @@RW1+RW7	XCHW A, @RW1+, @@RW1+RW7
+E	JMP @RW2+, @@RW2+PC+d16	JMP @RW2+, @@RW2+PC+d16	CALL @RW2+, @@RW2+PC+d16	CALL @RW2+, @@RW2+PC+d16	INCW @RW2+, @@RW2+PC+d16	INCW @RW2+, @@RW2+PC+d16	DECW @RW2+, @@RW2+PC+d16	DECW @RW2+, @@RW2+PC+d16	MOVW A, @RW2+, @@RW2+PC+d16	MOVW A, @RW2+, @@RW2+PC+d16	MOVW @RW2+, #16, @RW2+PC+d16, #16	MOVW @RW2+, #16, @RW2+PC+d16, #16	MOVW @RW2+, #16, @RW2+PC+d16, #16	MOVW @RW2+, #16, @RW2+PC+d16, #16	XCHW A, @RW2+, @@RW2+PC+d16	XCHW A, @RW2+, @@RW2+PC+d16
+F	JMP @RW3+, @@RW3+addr16	JMP @RW3+, @@RW3+addr16	CALL @RW3+, @@RW3+addr16	CALL @RW3+, @@RW3+addr16	INCW @RW3+, @@RW3+addr16	INCW @RW3+, @@RW3+addr16	DECW @RW3+, @@RW3+addr16	DECW @RW3+, @@RW3+addr16	MOVW A, @RW3+, @@RW3+addr16	MOVW A, @RW3+, @@RW3+addr16	MOVW @RW3+, #16, @RW3+addr16, #16	MOVW @RW3+, #16, @RW3+addr16, #16	MOVW @RW3+, #16, @RW3+addr16, #16	MOVW @RW3+, #16, @RW3+addr16, #16	XCHW A, @RW3+, @@RW3+addr16	XCHW A, @RW3+, @@RW3+addr16

表 C.9-10 ea 系命令 その 5 (第 1 バイト = 74_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	ADD A, R0; @RW0+d8	SUB A, R0; @RW0+d8	SUB A, R0; @RW0+d8	SUB A, R0; @RW0+d8	ADDC A, R0; @RW0+d8	ADDC A, R0; @RW0+d8	CMP A, R0; @RW0+d8	CMP A, R0; @RW0+d8	AND A, R0; @RW0+d8	AND A, R0; @RW0+d8	OR A, R0; @RW0+d8	OR A, R0; @RW0+d8	XOR A, R0; @RW0+d8	XOR A, R0; @RW0+d8	DBNZ @R0, r; RW0+d8, r	DBNZ @R0, r; RW0+d8, r
+1	ADD A, R1; @RW1+d8	SUB A, R1; @RW1+d8	SUB A, R1; @RW1+d8	SUB A, R1; @RW1+d8	ADDC A, R1; @RW1+d8	ADDC A, R1; @RW1+d8	CMP A, R1; @RW1+d8	CMP A, R1; @RW1+d8	AND A, R1; @RW1+d8	AND A, R1; @RW1+d8	OR A, R1; @RW1+d8	OR A, R1; @RW1+d8	XOR A, R1; @RW1+d8	XOR A, R1; @RW1+d8	DBNZ @R1, r; RW1+d8, r	DBNZ @R1, r; RW1+d8, r
+2	ADD A, R2; @RW2+d8	SUB A, R2; @RW2+d8	SUB A, R2; @RW2+d8	SUB A, R2; @RW2+d8	ADDC A, R2; @RW2+d8	ADDC A, R2; @RW2+d8	CMP A, R2; @RW2+d8	CMP A, R2; @RW2+d8	AND A, R2; @RW2+d8	AND A, R2; @RW2+d8	OR A, R2; @RW2+d8	OR A, R2; @RW2+d8	XOR A, R2; @RW2+d8	XOR A, R2; @RW2+d8	DBNZ @R2, r; RW2+d8, r	DBNZ @R2, r; RW2+d8, r
+3	ADD A, R3; @RW3+d8	SUB A, R3; @RW3+d8	SUB A, R3; @RW3+d8	SUB A, R3; @RW3+d8	ADDC A, R3; @RW3+d8	ADDC A, R3; @RW3+d8	CMP A, R3; @RW3+d8	CMP A, R3; @RW3+d8	AND A, R3; @RW3+d8	AND A, R3; @RW3+d8	OR A, R3; @RW3+d8	OR A, R3; @RW3+d8	XOR A, R3; @RW3+d8	XOR A, R3; @RW3+d8	DBNZ @R3, r; RW3+d8, r	DBNZ @R3, r; RW3+d8, r
+4	ADD A, R4; @RW4+d8	SUB A, R4; @RW4+d8	SUB A, R4; @RW4+d8	SUB A, R4; @RW4+d8	ADDC A, R4; @RW4+d8	ADDC A, R4; @RW4+d8	CMP A, R4; @RW4+d8	CMP A, R4; @RW4+d8	AND A, R4; @RW4+d8	AND A, R4; @RW4+d8	OR A, R4; @RW4+d8	OR A, R4; @RW4+d8	XOR A, R4; @RW4+d8	XOR A, R4; @RW4+d8	DBNZ @R4, r; RW4+d8, r	DBNZ @R4, r; RW4+d8, r
+5	ADD A, R5; @RW5+d8	SUB A, R5; @RW5+d8	SUB A, R5; @RW5+d8	SUB A, R5; @RW5+d8	ADDC A, R5; @RW5+d8	ADDC A, R5; @RW5+d8	CMP A, R5; @RW5+d8	CMP A, R5; @RW5+d8	AND A, R5; @RW5+d8	AND A, R5; @RW5+d8	OR A, R5; @RW5+d8	OR A, R5; @RW5+d8	XOR A, R5; @RW5+d8	XOR A, R5; @RW5+d8	DBNZ @R5, r; RW5+d8, r	DBNZ @R5, r; RW5+d8, r
+6	ADD A, R6; @RW6+d8	SUB A, R6; @RW6+d8	SUB A, R6; @RW6+d8	SUB A, R6; @RW6+d8	ADDC A, R6; @RW6+d8	ADDC A, R6; @RW6+d8	CMP A, R6; @RW6+d8	CMP A, R6; @RW6+d8	AND A, R6; @RW6+d8	AND A, R6; @RW6+d8	OR A, R6; @RW6+d8	OR A, R6; @RW6+d8	XOR A, R6; @RW6+d8	XOR A, R6; @RW6+d8	DBNZ @R6, r; RW6+d8, r	DBNZ @R6, r; RW6+d8, r
+7	ADD A, R7; @RW7+d8	SUB A, R7; @RW7+d8	SUB A, R7; @RW7+d8	SUB A, R7; @RW7+d8	ADDC A, R7; @RW7+d8	ADDC A, R7; @RW7+d8	CMP A, R7; @RW7+d8	CMP A, R7; @RW7+d8	AND A, R7; @RW7+d8	AND A, R7; @RW7+d8	OR A, R7; @RW7+d8	OR A, R7; @RW7+d8	XOR A, R7; @RW7+d8	XOR A, R7; @RW7+d8	DBNZ @R7, r; RW7+d8, r	DBNZ @R7, r; RW7+d8, r
+8	ADD A, @RW0; @RW0+d16	SUB A, @RW0; @RW0+d16	SUB A, @RW0; @RW0+d16	SUB A, @RW0; @RW0+d16	ADDC A, @RW0; @RW0+d16	ADDC A, @RW0; @RW0+d16	CMP A, @RW0; @RW0+d16	CMP A, @RW0; @RW0+d16	AND A, @RW0; @RW0+d16	AND A, @RW0; @RW0+d16	OR A, @RW0; @RW0+d16	OR A, @RW0; @RW0+d16	XOR A, @RW0; @RW0+d16	XOR A, @RW0; @RW0+d16	DBNZ @R0, r; W0+d16, r	DBNZ @R0, r; W0+d16, r
+9	ADD A, @RW1; @RW1+d16	SUB A, @RW1; @RW1+d16	SUB A, @RW1; @RW1+d16	SUB A, @RW1; @RW1+d16	ADDC A, @RW1; @RW1+d16	ADDC A, @RW1; @RW1+d16	CMP A, @RW1; @RW1+d16	CMP A, @RW1; @RW1+d16	AND A, @RW1; @RW1+d16	AND A, @RW1; @RW1+d16	OR A, @RW1; @RW1+d16	OR A, @RW1; @RW1+d16	XOR A, @RW1; @RW1+d16	XOR A, @RW1; @RW1+d16	DBNZ @R1, r; W1+d16, r	DBNZ @R1, r; W1+d16, r
+A	ADD A, @RW2; @RW2+d16	SUB A, @RW2; @RW2+d16	SUB A, @RW2; @RW2+d16	SUB A, @RW2; @RW2+d16	ADDC A, @RW2; @RW2+d16	ADDC A, @RW2; @RW2+d16	CMP A, @RW2; @RW2+d16	CMP A, @RW2; @RW2+d16	AND A, @RW2; @RW2+d16	AND A, @RW2; @RW2+d16	OR A, @RW2; @RW2+d16	OR A, @RW2; @RW2+d16	XOR A, @RW2; @RW2+d16	XOR A, @RW2; @RW2+d16	DBNZ @R2, r; W2+d16, r	DBNZ @R2, r; W2+d16, r
+B	ADD A, @RW3; @RW3+d16	SUB A, @RW3; @RW3+d16	SUB A, @RW3; @RW3+d16	SUB A, @RW3; @RW3+d16	ADDC A, @RW3; @RW3+d16	ADDC A, @RW3; @RW3+d16	CMP A, @RW3; @RW3+d16	CMP A, @RW3; @RW3+d16	AND A, @RW3; @RW3+d16	AND A, @RW3; @RW3+d16	OR A, @RW3; @RW3+d16	OR A, @RW3; @RW3+d16	XOR A, @RW3; @RW3+d16	XOR A, @RW3; @RW3+d16	DBNZ @R3, r; W3+d16, r	DBNZ @R3, r; W3+d16, r
+C	ADD A, @RW0+; @RW0+RW7	SUB A, @RW0+; @RW0+RW7	SUB A, @RW0+; @RW0+RW7	SUB A, @RW0+; @RW0+RW7	ADDC A, @RW0+; @RW0+RW7	ADDC A, @RW0+; @RW0+RW7	CMP A, @RW0+; @RW0+RW7	CMP A, @RW0+; @RW0+RW7	AND A, @RW0+; @RW0+RW7	AND A, @RW0+; @RW0+RW7	OR A, @RW0+; @RW0+RW7	OR A, @RW0+; @RW0+RW7	XOR A, @RW0+; @RW0+RW7	XOR A, @RW0+; @RW0+RW7	DBNZ @R0, r; W0+RW7, r	DBNZ @R0, r; W0+RW7, r
+D	ADD A, @RW1+; @RW1+RW7	SUB A, @RW1+; @RW1+RW7	SUB A, @RW1+; @RW1+RW7	SUB A, @RW1+; @RW1+RW7	ADDC A, @RW1+; @RW1+RW7	ADDC A, @RW1+; @RW1+RW7	CMP A, @RW1+; @RW1+RW7	CMP A, @RW1+; @RW1+RW7	AND A, @RW1+; @RW1+RW7	AND A, @RW1+; @RW1+RW7	OR A, @RW1+; @RW1+RW7	OR A, @RW1+; @RW1+RW7	XOR A, @RW1+; @RW1+RW7	XOR A, @RW1+; @RW1+RW7	DBNZ @R1, r; W1+RW7, r	DBNZ @R1, r; W1+RW7, r
+E	ADD A, @RW2+; @PC+d16	SUB A, @RW2+; @PC+d16	SUB A, @RW2+; @PC+d16	SUB A, @RW2+; @PC+d16	ADDC A, @RW2+; @PC+d16	ADDC A, @RW2+; @PC+d16	CMP A, @RW2+; @PC+d16	CMP A, @RW2+; @PC+d16	AND A, @RW2+; @PC+d16	AND A, @RW2+; @PC+d16	OR A, @RW2+; @PC+d16	OR A, @RW2+; @PC+d16	XOR A, @RW2+; @PC+d16	XOR A, @RW2+; @PC+d16	DBNZ @R2, r; PC+d16, r	DBNZ @R2, r; PC+d16, r
+F	ADD A, @RW3+; A, addr16	SUB A, @RW3+; A, addr16	SUB A, @RW3+; A, addr16	SUB A, @RW3+; A, addr16	ADDC A, @RW3+; A, addr16	ADDC A, @RW3+; A, addr16	CMP A, @RW3+; A, addr16	CMP A, @RW3+; A, addr16	AND A, @RW3+; A, addr16	AND A, @RW3+; A, addr16	OR A, @RW3+; A, addr16	OR A, @RW3+; A, addr16	XOR A, @RW3+; A, addr16	XOR A, @RW3+; A, addr16	DBNZ @R3, r; A, addr16, r	DBNZ @R3, r; A, addr16, r

表 C.9-11 ea 系命令 その6 (第1バイト = 75_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	ADD R0, A, @RW0+d8, A	SUB R0, A, @RW0+d8, A	SUB R0, A, @RW0+d8, A	SUB R0, A, @RW0+d8, A	SUBC A, R0, @RW0+d8, A	SUBC A, R0, @RW0+d8, A	NEG R0, @RW0+d8, A	NEG A, R0, @RW0+d8, A	AND R0, A, @RW0+d8, A	AND R0, A, @RW0+d8, A	OR R0, A, @RW0+d8, A	OR R0, A, @RW0+d8, A	XOR R0, A, @RW0+d8, A	XOR R0, A, @RW0+d8, A	NOT R0, @RW0+d8, A	NOT R0, @RW0+d8, A
+1	ADD R1, A, @RW1+d8, A	SUB R1, A, @RW1+d8, A	SUB R1, A, @RW1+d8, A	SUB R1, A, @RW1+d8, A	SUBC A, R1, @RW1+d8, A	SUBC A, R1, @RW1+d8, A	NEG R1, @RW1+d8, A	NEG A, R1, @RW1+d8, A	AND R1, A, @RW1+d8, A	AND R1, A, @RW1+d8, A	OR R1, A, @RW1+d8, A	OR R1, A, @RW1+d8, A	XOR R1, A, @RW1+d8, A	XOR R1, A, @RW1+d8, A	NOT R1, @RW1+d8, A	NOT R1, @RW1+d8, A
+2	ADD R2, A, @RW2+d8, A	SUB R2, A, @RW2+d8, A	SUB R2, A, @RW2+d8, A	SUB R2, A, @RW2+d8, A	SUBC A, R2, @RW2+d8, A	SUBC A, R2, @RW2+d8, A	NEG R2, @RW2+d8, A	NEG A, R2, @RW2+d8, A	AND R2, A, @RW2+d8, A	AND R2, A, @RW2+d8, A	OR R2, A, @RW2+d8, A	OR R2, A, @RW2+d8, A	XOR R2, A, @RW2+d8, A	XOR R2, A, @RW2+d8, A	NOT R2, @RW2+d8, A	NOT R2, @RW2+d8, A
+3	ADD R3, A, @RW3+d8, A	SUB R3, A, @RW3+d8, A	SUB R3, A, @RW3+d8, A	SUB R3, A, @RW3+d8, A	SUBC A, R3, @RW3+d8, A	SUBC A, R3, @RW3+d8, A	NEG R3, @RW3+d8, A	NEG A, R3, @RW3+d8, A	AND R3, A, @RW3+d8, A	AND R3, A, @RW3+d8, A	OR R3, A, @RW3+d8, A	OR R3, A, @RW3+d8, A	XOR R3, A, @RW3+d8, A	XOR R3, A, @RW3+d8, A	NOT R3, @RW3+d8, A	NOT R3, @RW3+d8, A
+4	ADD R4, A, @RW4+d8, A	SUB R4, A, @RW4+d8, A	SUB R4, A, @RW4+d8, A	SUB R4, A, @RW4+d8, A	SUBC A, R4, @RW4+d8, A	SUBC A, R4, @RW4+d8, A	NEG R4, @RW4+d8, A	NEG A, R4, @RW4+d8, A	AND R4, A, @RW4+d8, A	AND R4, A, @RW4+d8, A	OR R4, A, @RW4+d8, A	OR R4, A, @RW4+d8, A	XOR R4, A, @RW4+d8, A	XOR R4, A, @RW4+d8, A	NOT R4, @RW4+d8, A	NOT R4, @RW4+d8, A
+5	ADD R5, A, @RW5+d8, A	SUB R5, A, @RW5+d8, A	SUB R5, A, @RW5+d8, A	SUB R5, A, @RW5+d8, A	SUBC A, R5, @RW5+d8, A	SUBC A, R5, @RW5+d8, A	NEG R5, @RW5+d8, A	NEG A, R5, @RW5+d8, A	AND R5, A, @RW5+d8, A	AND R5, A, @RW5+d8, A	OR R5, A, @RW5+d8, A	OR R5, A, @RW5+d8, A	XOR R5, A, @RW5+d8, A	XOR R5, A, @RW5+d8, A	NOT R5, @RW5+d8, A	NOT R5, @RW5+d8, A
+6	ADD R6, A, @RW6+d8, A	SUB R6, A, @RW6+d8, A	SUB R6, A, @RW6+d8, A	SUB R6, A, @RW6+d8, A	SUBC A, R6, @RW6+d8, A	SUBC A, R6, @RW6+d8, A	NEG R6, @RW6+d8, A	NEG A, R6, @RW6+d8, A	AND R6, A, @RW6+d8, A	AND R6, A, @RW6+d8, A	OR R6, A, @RW6+d8, A	OR R6, A, @RW6+d8, A	XOR R6, A, @RW6+d8, A	XOR R6, A, @RW6+d8, A	NOT R6, @RW6+d8, A	NOT R6, @RW6+d8, A
+7	ADD R7, A, @RW7+d8, A	SUB R7, A, @RW7+d8, A	SUB R7, A, @RW7+d8, A	SUB R7, A, @RW7+d8, A	SUBC A, R7, @RW7+d8, A	SUBC A, R7, @RW7+d8, A	NEG R7, @RW7+d8, A	NEG A, R7, @RW7+d8, A	AND R7, A, @RW7+d8, A	AND R7, A, @RW7+d8, A	OR R7, A, @RW7+d8, A	OR R7, A, @RW7+d8, A	XOR R7, A, @RW7+d8, A	XOR R7, A, @RW7+d8, A	NOT R7, @RW7+d8, A	NOT R7, @RW7+d8, A
+8	ADD @RW0, A, @RW0+d16, A	SUB @RW0, A, @RW0+d16, A	SUB @RW0, A, @RW0+d16, A	SUB @RW0, A, @RW0+d16, A	SUBC A, @RW0, @RW0+d16, A	SUBC A, @RW0, @RW0+d16, A	NEG @RW0, @RW0+d16, A	NEG A, @RW0, @RW0+d16, A	AND @RW0, A, @RW0+d16, A	AND @RW0, A, @RW0+d16, A	OR @RW0, A, @RW0+d16, A	OR @RW0, A, @RW0+d16, A	XOR @RW0, A, @RW0+d16, A	XOR @RW0, A, @RW0+d16, A	NOT @RW0, @RW0+d16, A	NOT @RW0, @RW0+d16, A
+9	ADD @RW1, A, @RW1+d16, A	SUB @RW1, A, @RW1+d16, A	SUB @RW1, A, @RW1+d16, A	SUB @RW1, A, @RW1+d16, A	SUBC A, @RW1, @RW1+d16, A	SUBC A, @RW1, @RW1+d16, A	NEG @RW1, @RW1+d16, A	NEG A, @RW1, @RW1+d16, A	AND @RW1, A, @RW1+d16, A	AND @RW1, A, @RW1+d16, A	OR @RW1, A, @RW1+d16, A	OR @RW1, A, @RW1+d16, A	XOR @RW1, A, @RW1+d16, A	XOR @RW1, A, @RW1+d16, A	NOT @RW1, @RW1+d16, A	NOT @RW1, @RW1+d16, A
+A	ADD @RW2, A, @RW2+d16, A	SUB @RW2, A, @RW2+d16, A	SUB @RW2, A, @RW2+d16, A	SUB @RW2, A, @RW2+d16, A	SUBC A, @RW2, @RW2+d16, A	SUBC A, @RW2, @RW2+d16, A	NEG @RW2, @RW2+d16, A	NEG A, @RW2, @RW2+d16, A	AND @RW2, A, @RW2+d16, A	AND @RW2, A, @RW2+d16, A	OR @RW2, A, @RW2+d16, A	OR @RW2, A, @RW2+d16, A	XOR @RW2, A, @RW2+d16, A	XOR @RW2, A, @RW2+d16, A	NOT @RW2, @RW2+d16, A	NOT @RW2, @RW2+d16, A
+B	ADD @RW3, A, @RW3+d16, A	SUB @RW3, A, @RW3+d16, A	SUB @RW3, A, @RW3+d16, A	SUB @RW3, A, @RW3+d16, A	SUBC A, @RW3, @RW3+d16, A	SUBC A, @RW3, @RW3+d16, A	NEG @RW3, @RW3+d16, A	NEG A, @RW3, @RW3+d16, A	AND @RW3, A, @RW3+d16, A	AND @RW3, A, @RW3+d16, A	OR @RW3, A, @RW3+d16, A	OR @RW3, A, @RW3+d16, A	XOR @RW3, A, @RW3+d16, A	XOR @RW3, A, @RW3+d16, A	NOT @RW3, @RW3+d16, A	NOT @RW3, @RW3+d16, A
+C	ADD @RW0+, A, @RW0+RW7, A	SUB @RW0+, A, @RW0+RW7, A	SUB @RW0+, A, @RW0+RW7, A	SUB @RW0+, A, @RW0+RW7, A	SUBC A, @RW0+, @RW0+RW7, A	SUBC A, @RW0+, @RW0+RW7, A	NEG @RW0+, @RW0+RW7, A	NEG A, @RW0+, @RW0+RW7, A	AND @RW0+, A, @RW0+RW7, A	AND @RW0+, A, @RW0+RW7, A	OR @RW0+, A, @RW0+RW7, A	OR @RW0+, A, @RW0+RW7, A	XOR @RW0+, A, @RW0+RW7, A	XOR @RW0+, A, @RW0+RW7, A	NOT @RW0+, @RW0+RW7, A	NOT @RW0+, @RW0+RW7, A
+D	ADD @RW1+, A, @RW1+RW7, A	SUB @RW1+, A, @RW1+RW7, A	SUB @RW1+, A, @RW1+RW7, A	SUB @RW1+, A, @RW1+RW7, A	SUBC A, @RW1+, @RW1+RW7, A	SUBC A, @RW1+, @RW1+RW7, A	NEG @RW1+, @RW1+RW7, A	NEG A, @RW1+, @RW1+RW7, A	AND @RW1+, A, @RW1+RW7, A	AND @RW1+, A, @RW1+RW7, A	OR @RW1+, A, @RW1+RW7, A	OR @RW1+, A, @RW1+RW7, A	XOR @RW1+, A, @RW1+RW7, A	XOR @RW1+, A, @RW1+RW7, A	NOT @RW1+, @RW1+RW7, A	NOT @RW1+, @RW1+RW7, A
+E	ADD @RW2+, A, @PC+d16, A	SUB @RW2+, A, @PC+d16, A	SUB @RW2+, A, @PC+d16, A	SUB @RW2+, A, @PC+d16, A	SUBC A, @RW2+, @PC+d16, A	SUBC A, @RW2+, @PC+d16, A	NEG @RW2+, @PC+d16, A	NEG A, @RW2+, @PC+d16, A	AND @RW2+, A, @PC+d16, A	AND @RW2+, A, @PC+d16, A	OR @RW2+, A, @PC+d16, A	OR @RW2+, A, @PC+d16, A	XOR @RW2+, A, @PC+d16, A	XOR @RW2+, A, @PC+d16, A	NOT @RW2+, @PC+d16, A	NOT @RW2+, @PC+d16, A
+F	ADD @RW3+, A, @addr16, A	SUB @RW3+, A, @addr16, A	SUB @RW3+, A, @addr16, A	SUB @RW3+, A, @addr16, A	SUBC A, @RW3+, @addr16, A	SUBC A, @RW3+, @addr16, A	NEG @RW3+, @addr16, A	NEG A, @RW3+, @addr16, A	AND @RW3+, A, @addr16, A	AND @RW3+, A, @addr16, A	OR @RW3+, A, @addr16, A	OR @RW3+, A, @addr16, A	XOR @RW3+, A, @addr16, A	XOR @RW3+, A, @addr16, A	NOT @RW3+, @addr16, A	NOT @RW3+, @addr16, A

表 C.9-12 ea 系命令 その 7 (第 1 バイト = 76_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	ADDW A, RW0, @RW0+d8	SUBW A, RW0, @RW0+d8	SUBW A, RW0, @RW0+d8	SUBW A, RW0, @RW0+d8	ADDCW A, RW0, @RW0+d8	ADDCW A, RW0, @RW0+d8	CMPW A, RW0, @RW0+d8	CMPW A, RW0, @RW0+d8	ANDW A, RW0, @RW0+d8	ANDW A, RW0, @RW0+d8	ORW A, RW0, @RW0+d8	ORW A, RW0, @RW0+d8	XORW A, RW0, @RW0+d8	XORW A, RW0, @RW0+d8	DWBZ RW0, r' @RW0+d8, r	DWBZ RW0, r' @RW0+d8, r
+1	ADDW A, RW1, @RW1+d8	SUBW A, RW1, @RW1+d8	SUBW A, RW1, @RW1+d8	SUBW A, RW1, @RW1+d8	ADDCW A, RW1, @RW1+d8	ADDCW A, RW1, @RW1+d8	CMPW A, RW1, @RW1+d8	CMPW A, RW1, @RW1+d8	ANDW A, RW1, @RW1+d8	ANDW A, RW1, @RW1+d8	ORW A, RW1, @RW1+d8	ORW A, RW1, @RW1+d8	XORW A, RW1, @RW1+d8	XORW A, RW1, @RW1+d8	DWBZ RW1, r' @RW1+d8, r	DWBZ RW1, r' @RW1+d8, r
+2	ADDW A, RW2, @RW2+d8	SUBW A, RW2, @RW2+d8	SUBW A, RW2, @RW2+d8	SUBW A, RW2, @RW2+d8	ADDCW A, RW2, @RW2+d8	ADDCW A, RW2, @RW2+d8	CMPW A, RW2, @RW2+d8	CMPW A, RW2, @RW2+d8	ANDW A, RW2, @RW2+d8	ANDW A, RW2, @RW2+d8	ORW A, RW2, @RW2+d8	ORW A, RW2, @RW2+d8	XORW A, RW2, @RW2+d8	XORW A, RW2, @RW2+d8	DWBZ RW2, r' @RW2+d8, r	DWBZ RW2, r' @RW2+d8, r
+3	ADDW A, RW3, @RW3+d8	SUBW A, RW3, @RW3+d8	SUBW A, RW3, @RW3+d8	SUBW A, RW3, @RW3+d8	ADDCW A, RW3, @RW3+d8	ADDCW A, RW3, @RW3+d8	CMPW A, RW3, @RW3+d8	CMPW A, RW3, @RW3+d8	ANDW A, RW3, @RW3+d8	ANDW A, RW3, @RW3+d8	ORW A, RW3, @RW3+d8	ORW A, RW3, @RW3+d8	XORW A, RW3, @RW3+d8	XORW A, RW3, @RW3+d8	DWBZ RW3, r' @RW3+d8, r	DWBZ RW3, r' @RW3+d8, r
+4	ADDW A, RW4, @RW4+d8	SUBW A, RW4, @RW4+d8	SUBW A, RW4, @RW4+d8	SUBW A, RW4, @RW4+d8	ADDCW A, RW4, @RW4+d8	ADDCW A, RW4, @RW4+d8	CMPW A, RW4, @RW4+d8	CMPW A, RW4, @RW4+d8	ANDW A, RW4, @RW4+d8	ANDW A, RW4, @RW4+d8	ORW A, RW4, @RW4+d8	ORW A, RW4, @RW4+d8	XORW A, RW4, @RW4+d8	XORW A, RW4, @RW4+d8	DWBZ RW4, r' @RW4+d8, r	DWBZ RW4, r' @RW4+d8, r
+5	ADDW A, RW5, @RW5+d8	SUBW A, RW5, @RW5+d8	SUBW A, RW5, @RW5+d8	SUBW A, RW5, @RW5+d8	ADDCW A, RW5, @RW5+d8	ADDCW A, RW5, @RW5+d8	CMPW A, RW5, @RW5+d8	CMPW A, RW5, @RW5+d8	ANDW A, RW5, @RW5+d8	ANDW A, RW5, @RW5+d8	ORW A, RW5, @RW5+d8	ORW A, RW5, @RW5+d8	XORW A, RW5, @RW5+d8	XORW A, RW5, @RW5+d8	DWBZ RW5, r' @RW5+d8, r	DWBZ RW5, r' @RW5+d8, r
+6	ADDW A, RW6, @RW6+d8	SUBW A, RW6, @RW6+d8	SUBW A, RW6, @RW6+d8	SUBW A, RW6, @RW6+d8	ADDCW A, RW6, @RW6+d8	ADDCW A, RW6, @RW6+d8	CMPW A, RW6, @RW6+d8	CMPW A, RW6, @RW6+d8	ANDW A, RW6, @RW6+d8	ANDW A, RW6, @RW6+d8	ORW A, RW6, @RW6+d8	ORW A, RW6, @RW6+d8	XORW A, RW6, @RW6+d8	XORW A, RW6, @RW6+d8	DWBZ RW6, r' @RW6+d8, r	DWBZ RW6, r' @RW6+d8, r
+7	ADDW A, RW7, @RW7+d8	SUBW A, RW7, @RW7+d8	SUBW A, RW7, @RW7+d8	SUBW A, RW7, @RW7+d8	ADDCW A, RW7, @RW7+d8	ADDCW A, RW7, @RW7+d8	CMPW A, RW7, @RW7+d8	CMPW A, RW7, @RW7+d8	ANDW A, RW7, @RW7+d8	ANDW A, RW7, @RW7+d8	ORW A, RW7, @RW7+d8	ORW A, RW7, @RW7+d8	XORW A, RW7, @RW7+d8	XORW A, RW7, @RW7+d8	DWBZ RW7, r' @RW7+d8, r	DWBZ RW7, r' @RW7+d8, r
+8	ADDW A, @RW0, @RW0+d16	SUBW A, @RW0, @RW0+d16	SUBW A, @RW0, @RW0+d16	SUBW A, @RW0, @RW0+d16	ADDCW A, @RW0, @RW0+d16	ADDCW A, @RW0, @RW0+d16	CMPW A, @RW0, @RW0+d16	CMPW A, @RW0, @RW0+d16	ANDW A, @RW0, @RW0+d16	ANDW A, @RW0, @RW0+d16	ORW A, @RW0, @RW0+d16	ORW A, @RW0, @RW0+d16	XORW A, @RW0, @RW0+d16	XORW A, @RW0, @RW0+d16	DWBZ @RW0, r' @RW0+d16, r	DWBZ @RW0, r' @RW0+d16, r
+9	ADDW A, @RW1, @RW1+d16	SUBW A, @RW1, @RW1+d16	SUBW A, @RW1, @RW1+d16	SUBW A, @RW1, @RW1+d16	ADDCW A, @RW1, @RW1+d16	ADDCW A, @RW1, @RW1+d16	CMPW A, @RW1, @RW1+d16	CMPW A, @RW1, @RW1+d16	ANDW A, @RW1, @RW1+d16	ANDW A, @RW1, @RW1+d16	ORW A, @RW1, @RW1+d16	ORW A, @RW1, @RW1+d16	XORW A, @RW1, @RW1+d16	XORW A, @RW1, @RW1+d16	DWBZ @RW1, r' @RW1+d16, r	DWBZ @RW1, r' @RW1+d16, r
+A	ADDW A, @RW2, @RW2+d16	SUBW A, @RW2, @RW2+d16	SUBW A, @RW2, @RW2+d16	SUBW A, @RW2, @RW2+d16	ADDCW A, @RW2, @RW2+d16	ADDCW A, @RW2, @RW2+d16	CMPW A, @RW2, @RW2+d16	CMPW A, @RW2, @RW2+d16	ANDW A, @RW2, @RW2+d16	ANDW A, @RW2, @RW2+d16	ORW A, @RW2, @RW2+d16	ORW A, @RW2, @RW2+d16	XORW A, @RW2, @RW2+d16	XORW A, @RW2, @RW2+d16	DWBZ @RW2, r' @RW2+d16, r	DWBZ @RW2, r' @RW2+d16, r
+B	ADDW A, @RW3, @RW3+d16	SUBW A, @RW3, @RW3+d16	SUBW A, @RW3, @RW3+d16	SUBW A, @RW3, @RW3+d16	ADDCW A, @RW3, @RW3+d16	ADDCW A, @RW3, @RW3+d16	CMPW A, @RW3, @RW3+d16	CMPW A, @RW3, @RW3+d16	ANDW A, @RW3, @RW3+d16	ANDW A, @RW3, @RW3+d16	ORW A, @RW3, @RW3+d16	ORW A, @RW3, @RW3+d16	XORW A, @RW3, @RW3+d16	XORW A, @RW3, @RW3+d16	DWBZ @RW3, r' @RW3+d16, r	DWBZ @RW3, r' @RW3+d16, r
+C	ADDW A, @RW0+, @RW0+RW7	SUBW A, @RW0+, @RW0+RW7	SUBW A, @RW0+, @RW0+RW7	SUBW A, @RW0+, @RW0+RW7	ADDCW A, @RW0+, @RW0+RW7	ADDCW A, @RW0+, @RW0+RW7	CMPW A, @RW0+, @RW0+RW7	CMPW A, @RW0+, @RW0+RW7	ANDW A, @RW0+, @RW0+RW7	ANDW A, @RW0+, @RW0+RW7	ORW A, @RW0+, @RW0+RW7	ORW A, @RW0+, @RW0+RW7	XORW A, @RW0+, @RW0+RW7	XORW A, @RW0+, @RW0+RW7	DWBZ @RW0+, r' @RW0+RW7, r	DWBZ @RW0+, r' @RW0+RW7, r
+D	ADDW A, @RW1+, @RW1+RW7	SUBW A, @RW1+, @RW1+RW7	SUBW A, @RW1+, @RW1+RW7	SUBW A, @RW1+, @RW1+RW7	ADDCW A, @RW1+, @RW1+RW7	ADDCW A, @RW1+, @RW1+RW7	CMPW A, @RW1+, @RW1+RW7	CMPW A, @RW1+, @RW1+RW7	ANDW A, @RW1+, @RW1+RW7	ANDW A, @RW1+, @RW1+RW7	ORW A, @RW1+, @RW1+RW7	ORW A, @RW1+, @RW1+RW7	XORW A, @RW1+, @RW1+RW7	XORW A, @RW1+, @RW1+RW7	DWBZ @RW1+, r' @RW1+RW7, r	DWBZ @RW1+, r' @RW1+RW7, r
+E	ADDW A, @RW2+, @PC+d16	SUBW A, @RW2+, @PC+d16	SUBW A, @RW2+, @PC+d16	SUBW A, @RW2+, @PC+d16	ADDCW A, @RW2+, @PC+d16	ADDCW A, @RW2+, @PC+d16	CMPW A, @RW2+, @PC+d16	CMPW A, @RW2+, @PC+d16	ANDW A, @RW2+, @PC+d16	ANDW A, @RW2+, @PC+d16	ORW A, @RW2+, @PC+d16	ORW A, @RW2+, @PC+d16	XORW A, @RW2+, @PC+d16	XORW A, @RW2+, @PC+d16	DWBZ @RW2+, r' @PC+d16, r	DWBZ @RW2+, r' @PC+d16, r
+F	ADDW A, @RW3+, addr 16	SUBW A, @RW3+, addr 16	SUBW A, @RW3+, addr 16	SUBW A, @RW3+, addr 16	ADDCW A, @RW3+, addr 16	ADDCW A, @RW3+, addr 16	CMPW A, @RW3+, addr 16	CMPW A, @RW3+, addr 16	ANDW A, @RW3+, addr 16	ANDW A, @RW3+, addr 16	ORW A, @RW3+, addr 16	ORW A, @RW3+, addr 16	XORW A, @RW3+, addr 16	XORW A, @RW3+, addr 16	DWBZ @RW3+, r' addr 16, r	DWBZ @RW3+, r' addr 16, r

表 C.9-13 ea 系命令 その 8 (第 1 バイト = 77_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	ADDW @RW0, A' @RW0+d8, A	SUBW @RW0, A' @RW0+d8, A	SUBW @RW0, A' @RW0+d8, A	SUBW @RW0+d8, A, A'	SUBCW A, @RW0+d8	SUBCW A, @RW0+d8	NEGW @RW0' @RW0+d8	NEGW @RW0' @RW0+d8	ANDW @RW0, A' @RW0+d8, A	ANDW @RW0+d8, A	ORW @RW0, A' @RW0+d8, A	ORW @RW0+d8, A	XORW @RW0, A' @RW0+d8, A	XORW @RW0+d8, A	NOTW @RW0' @RW0+d8	NOTW @RW0+d8
+1	ADDW @RW1, A' @RW1+d8, A	SUBW @RW1, A' @RW1+d8, A	SUBW @RW1, A' @RW1+d8, A	SUBW @RW1+d8, A, A'	SUBCW A, @RW1+d8	SUBCW A, @RW1+d8	NEGW @RW1' @RW1+d8	NEGW @RW1' @RW1+d8	ANDW @RW1, A' @RW1+d8, A	ANDW @RW1+d8, A	ORW @RW1, A' @RW1+d8, A	ORW @RW1+d8, A	XORW @RW1, A' @RW1+d8, A	XORW @RW1+d8, A	NOTW @RW1' @RW1+d8	NOTW @RW1+d8
+2	ADDW @RW2, A' @RW2+d8, A	SUBW @RW2, A' @RW2+d8, A	SUBW @RW2, A' @RW2+d8, A	SUBW @RW2+d8, A, A'	SUBCW A, @RW2+d8	SUBCW A, @RW2+d8	NEGW @RW2' @RW2+d8	NEGW @RW2' @RW2+d8	ANDW @RW2, A' @RW2+d8, A	ANDW @RW2+d8, A	ORW @RW2, A' @RW2+d8, A	ORW @RW2+d8, A	XORW @RW2, A' @RW2+d8, A	XORW @RW2+d8, A	NOTW @RW2' @RW2+d8	NOTW @RW2+d8
+3	ADDW @RW3, A' @RW3+d8, A	SUBW @RW3, A' @RW3+d8, A	SUBW @RW3, A' @RW3+d8, A	SUBW @RW3+d8, A, A'	SUBCW A, @RW3+d8	SUBCW A, @RW3+d8	NEGW @RW3' @RW3+d8	NEGW @RW3' @RW3+d8	ANDW @RW3, A' @RW3+d8, A	ANDW @RW3+d8, A	ORW @RW3, A' @RW3+d8, A	ORW @RW3+d8, A	XORW @RW3, A' @RW3+d8, A	XORW @RW3+d8, A	NOTW @RW3' @RW3+d8	NOTW @RW3+d8
+4	ADDW @RW4, A' @RW4+d8, A	SUBW @RW4, A' @RW4+d8, A	SUBW @RW4, A' @RW4+d8, A	SUBW @RW4+d8, A, A'	SUBCW A, @RW4+d8	SUBCW A, @RW4+d8	NEGW @RW4' @RW4+d8	NEGW @RW4' @RW4+d8	ANDW @RW4, A' @RW4+d8, A	ANDW @RW4+d8, A	ORW @RW4, A' @RW4+d8, A	ORW @RW4+d8, A	XORW @RW4, A' @RW4+d8, A	XORW @RW4+d8, A	NOTW @RW4' @RW4+d8	NOTW @RW4+d8
+5	ADDW @RW5, A' @RW5+d8, A	SUBW @RW5, A' @RW5+d8, A	SUBW @RW5, A' @RW5+d8, A	SUBW @RW5+d8, A, A'	SUBCW A, @RW5+d8	SUBCW A, @RW5+d8	NEGW @RW5' @RW5+d8	NEGW @RW5' @RW5+d8	ANDW @RW5, A' @RW5+d8, A	ANDW @RW5+d8, A	ORW @RW5, A' @RW5+d8, A	ORW @RW5+d8, A	XORW @RW5, A' @RW5+d8, A	XORW @RW5+d8, A	NOTW @RW5' @RW5+d8	NOTW @RW5+d8
+6	ADDW @RW6, A' @RW6+d8, A	SUBW @RW6, A' @RW6+d8, A	SUBW @RW6, A' @RW6+d8, A	SUBW @RW6+d8, A, A'	SUBCW A, @RW6+d8	SUBCW A, @RW6+d8	NEGW @RW6' @RW6+d8	NEGW @RW6' @RW6+d8	ANDW @RW6, A' @RW6+d8, A	ANDW @RW6+d8, A	ORW @RW6, A' @RW6+d8, A	ORW @RW6+d8, A	XORW @RW6, A' @RW6+d8, A	XORW @RW6+d8, A	NOTW @RW6' @RW6+d8	NOTW @RW6+d8
+7	ADDW @RW7, A' @RW7+d8, A	SUBW @RW7, A' @RW7+d8, A	SUBW @RW7, A' @RW7+d8, A	SUBW @RW7+d8, A, A'	SUBCW A, @RW7+d8	SUBCW A, @RW7+d8	NEGW @RW7' @RW7+d8	NEGW @RW7' @RW7+d8	ANDW @RW7, A' @RW7+d8, A	ANDW @RW7+d8, A	ORW @RW7, A' @RW7+d8, A	ORW @RW7+d8, A	XORW @RW7, A' @RW7+d8, A	XORW @RW7+d8, A	NOTW @RW7' @RW7+d8	NOTW @RW7+d8
+8	ADDW @RW0, A' @RW0+d16, A	SUBW @RW0, A' @RW0+d16, A	SUBW @RW0, A' @RW0+d16, A	SUBW @RW0+d16, A, A'	SUBCW A, @RW0+d16	SUBCW A, @RW0+d16	NEGW @RW0' @RW0+d16	NEGW @RW0' @RW0+d16	ANDW @RW0, A' @RW0+d16, A	ANDW @RW0+d16, A	ORW @RW0, A' @RW0+d16, A	ORW @RW0+d16, A	XORW @RW0, A' @RW0+d16, A	XORW @RW0+d16, A	NOTW @RW0' @RW0+d16	NOTW @RW0+d16
+9	ADDW @RW1, A' @RW1+d16, A	SUBW @RW1, A' @RW1+d16, A	SUBW @RW1, A' @RW1+d16, A	SUBW @RW1+d16, A, A'	SUBCW A, @RW1+d16	SUBCW A, @RW1+d16	NEGW @RW1' @RW1+d16	NEGW @RW1' @RW1+d16	ANDW @RW1, A' @RW1+d16, A	ANDW @RW1+d16, A	ORW @RW1, A' @RW1+d16, A	ORW @RW1+d16, A	XORW @RW1, A' @RW1+d16, A	XORW @RW1+d16, A	NOTW @RW1' @RW1+d16	NOTW @RW1+d16
+A	ADDW @RW2, A' @RW2+d16, A	SUBW @RW2, A' @RW2+d16, A	SUBW @RW2, A' @RW2+d16, A	SUBW @RW2+d16, A, A'	SUBCW A, @RW2+d16	SUBCW A, @RW2+d16	NEGW @RW2' @RW2+d16	NEGW @RW2' @RW2+d16	ANDW @RW2, A' @RW2+d16, A	ANDW @RW2+d16, A	ORW @RW2, A' @RW2+d16, A	ORW @RW2+d16, A	XORW @RW2, A' @RW2+d16, A	XORW @RW2+d16, A	NOTW @RW2' @RW2+d16	NOTW @RW2+d16
+B	ADDW @RW3, A' @RW3+d16, A	SUBW @RW3, A' @RW3+d16, A	SUBW @RW3, A' @RW3+d16, A	SUBW @RW3+d16, A, A'	SUBCW A, @RW3+d16	SUBCW A, @RW3+d16	NEGW @RW3' @RW3+d16	NEGW @RW3' @RW3+d16	ANDW @RW3, A' @RW3+d16, A	ANDW @RW3+d16, A	ORW @RW3, A' @RW3+d16, A	ORW @RW3+d16, A	XORW @RW3, A' @RW3+d16, A	XORW @RW3+d16, A	NOTW @RW3' @RW3+d16	NOTW @RW3+d16
+C	ADDW @RW0+, A' @RW0+RW7, A	SUBW @RW0+, A' @RW0+RW7, A	SUBW @RW0+, A' @RW0+RW7, A	SUBW @RW0+RW7, A, A'	SUBCW A, @RW0+RW7	SUBCW A, @RW0+RW7	NEGW @RW0' @RW0+RW7	NEGW @RW0' @RW0+RW7	ANDW @RW0+, A' @RW0+RW7, A	ANDW @RW0+RW7, A	ORW @RW0+, A' @RW0+RW7, A	ORW @RW0+RW7, A	XORW @RW0+, A' @RW0+RW7, A	XORW @RW0+RW7, A	NOTW @RW0' @RW0+RW7	NOTW @RW0+RW7
+D	ADDW @RW1+, A' @RW1+RW7, A	SUBW @RW1+, A' @RW1+RW7, A	SUBW @RW1+, A' @RW1+RW7, A	SUBW @RW1+RW7, A, A'	SUBCW A, @RW1+RW7	SUBCW A, @RW1+RW7	NEGW @RW1' @RW1+RW7	NEGW @RW1' @RW1+RW7	ANDW @RW1+, A' @RW1+RW7, A	ANDW @RW1+RW7, A	ORW @RW1+, A' @RW1+RW7, A	ORW @RW1+RW7, A	XORW @RW1+, A' @RW1+RW7, A	XORW @RW1+RW7, A	NOTW @RW1' @RW1+RW7	NOTW @RW1+RW7
+E	ADDW @RW2+, A' @PC+d16, A	SUBW @RW2+, A' @PC+d16, A	SUBW @RW2+, A' @PC+d16, A	SUBW @PC+d16, A, A'	SUBCW A, @RW2+PC	SUBCW A, @RW2+PC	NEGW @RW2' @PC+d16	NEGW @RW2' @PC+d16	ANDW @RW2+, A' @PC+d16, A	ANDW @PC+d16, A	ORW @RW2+, A' @PC+d16, A	ORW @PC+d16, A	XORW @RW2+, A' @PC+d16, A	XORW @PC+d16, A	NOTW @RW2' @PC+d16	NOTW @PC+d16
+F	ADDW @RW3+, A' addr16, A	SUBW @RW3+, A' addr16, A	SUBW @RW3+, A' addr16, A	SUBW addr16, A, A'	SUBCW A, @RW3+addr16	SUBCW A, @RW3+addr16	NEGW @RW3' addr16	NEGW @RW3' addr16	ANDW @RW3+, A' addr16, A	ANDW addr16, A	ORW @RW3+, A' addr16, A	ORW addr16, A	XORW @RW3+, A' addr16, A	XORW addr16, A	NOTW @RW3' @RW3+addr16	NOTW @RW3+addr16

表 C.9-14 ea 系命令 その9 (第1バイト = 78_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	MULU A, R0', @RW0+d8	MULU A, MULU A, R0', @RW0+d8	MULUW A, MULUW A, RW0', @RW0+d8	MULUW A, MULUW A, RW0', @RW0+d8	MUL A, R0', @RW0+d8	MUL A, R0', @RW0+d8	MULW A, MULW A, RW0', @RW0+d8	MULW A, MULW A, RW0', @RW0+d8	MULW A, MULW A, RW0', @RW0+d8	DIVU A, R0', @RW0+d8	DIVU A, R0', @RW0+d8	DIVUW A, DIVUW A, RW0', @RW0+d8	DIVUW A, DIVUW A, RW0', @RW0+d8	DIV A, R0', @RW0+d8	DIV A, R0', @RW0+d8	DIVW A, DIVW A, RW0', @RW0+d8
+1	MULU A, R1', @RW1+d8	MULU A, MULU A, R1', @RW1+d8	MULUW A, MULUW A, RW1', @RW1+d8	MULUW A, MULUW A, RW1', @RW1+d8	MUL A, R1', @RW1+d8	MUL A, R1', @RW1+d8	MULW A, MULW A, RW1', @RW1+d8	MULW A, MULW A, RW1', @RW1+d8	MULW A, MULW A, RW1', @RW1+d8	DIVU A, R1', @RW1+d8	DIVU A, R1', @RW1+d8	DIVUW A, DIVUW A, RW1', @RW1+d8	DIVUW A, DIVUW A, RW1', @RW1+d8	DIV A, R1', @RW1+d8	DIV A, R1', @RW1+d8	DIVW A, DIVW A, RW1', @RW1+d8
+2	MULU A, R2', @RW2+d8	MULU A, MULU A, R2', @RW2+d8	MULUW A, MULUW A, RW2', @RW2+d8	MULUW A, MULUW A, RW2', @RW2+d8	MUL A, R2', @RW2+d8	MUL A, R2', @RW2+d8	MULW A, MULW A, RW2', @RW2+d8	MULW A, MULW A, RW2', @RW2+d8	MULW A, MULW A, RW2', @RW2+d8	DIVU A, R2', @RW2+d8	DIVU A, R2', @RW2+d8	DIVUW A, DIVUW A, RW2', @RW2+d8	DIVUW A, DIVUW A, RW2', @RW2+d8	DIV A, R2', @RW2+d8	DIV A, R2', @RW2+d8	DIVW A, DIVW A, RW2', @RW2+d8
+3	MULU A, R3', @RW3+d8	MULU A, MULU A, R3', @RW3+d8	MULUW A, MULUW A, RW3', @RW3+d8	MULUW A, MULUW A, RW3', @RW3+d8	MUL A, R3', @RW3+d8	MUL A, R3', @RW3+d8	MULW A, MULW A, RW3', @RW3+d8	MULW A, MULW A, RW3', @RW3+d8	MULW A, MULW A, RW3', @RW3+d8	DIVU A, R3', @RW3+d8	DIVU A, R3', @RW3+d8	DIVUW A, DIVUW A, RW3', @RW3+d8	DIVUW A, DIVUW A, RW3', @RW3+d8	DIV A, R3', @RW3+d8	DIV A, R3', @RW3+d8	DIVW A, DIVW A, RW3', @RW3+d8
+4	MULU A, R4', @RW4+d8	MULU A, MULU A, R4', @RW4+d8	MULUW A, MULUW A, RW4', @RW4+d8	MULUW A, MULUW A, RW4', @RW4+d8	MUL A, R4', @RW4+d8	MUL A, R4', @RW4+d8	MULW A, MULW A, RW4', @RW4+d8	MULW A, MULW A, RW4', @RW4+d8	MULW A, MULW A, RW4', @RW4+d8	DIVU A, R4', @RW4+d8	DIVU A, R4', @RW4+d8	DIVUW A, DIVUW A, RW4', @RW4+d8	DIVUW A, DIVUW A, RW4', @RW4+d8	DIV A, R4', @RW4+d8	DIV A, R4', @RW4+d8	DIVW A, DIVW A, RW4', @RW4+d8
+5	MULU A, R5', @RW5+d8	MULU A, MULU A, R5', @RW5+d8	MULUW A, MULUW A, RW5', @RW5+d8	MULUW A, MULUW A, RW5', @RW5+d8	MUL A, R5', @RW5+d8	MUL A, R5', @RW5+d8	MULW A, MULW A, RW5', @RW5+d8	MULW A, MULW A, RW5', @RW5+d8	MULW A, MULW A, RW5', @RW5+d8	DIVU A, R5', @RW5+d8	DIVU A, R5', @RW5+d8	DIVUW A, DIVUW A, RW5', @RW5+d8	DIVUW A, DIVUW A, RW5', @RW5+d8	DIV A, R5', @RW5+d8	DIV A, R5', @RW5+d8	DIVW A, DIVW A, RW5', @RW5+d8
+6	MULU A, R6', @RW6+d8	MULU A, MULU A, R6', @RW6+d8	MULUW A, MULUW A, RW6', @RW6+d8	MULUW A, MULUW A, RW6', @RW6+d8	MUL A, R6', @RW6+d8	MUL A, R6', @RW6+d8	MULW A, MULW A, RW6', @RW6+d8	MULW A, MULW A, RW6', @RW6+d8	MULW A, MULW A, RW6', @RW6+d8	DIVU A, R6', @RW6+d8	DIVU A, R6', @RW6+d8	DIVUW A, DIVUW A, RW6', @RW6+d8	DIVUW A, DIVUW A, RW6', @RW6+d8	DIV A, R6', @RW6+d8	DIV A, R6', @RW6+d8	DIVW A, DIVW A, RW6', @RW6+d8
+7	MULU A, R7', @RW7+d8	MULU A, MULU A, R7', @RW7+d8	MULUW A, MULUW A, RW7', @RW7+d8	MULUW A, MULUW A, RW7', @RW7+d8	MUL A, R7', @RW7+d8	MUL A, R7', @RW7+d8	MULW A, MULW A, RW7', @RW7+d8	MULW A, MULW A, RW7', @RW7+d8	MULW A, MULW A, RW7', @RW7+d8	DIVU A, R7', @RW7+d8	DIVU A, R7', @RW7+d8	DIVUW A, DIVUW A, RW7', @RW7+d8	DIVUW A, DIVUW A, RW7', @RW7+d8	DIV A, R7', @RW7+d8	DIV A, R7', @RW7+d8	DIVW A, DIVW A, RW7', @RW7+d8
+8	MULU A, @RW0	MULU A, MULU A, @RW0	MULUW A, MULUW A, @RW0	MULUW A, MULUW A, @RW0	MUL A, @RW0	MUL A, @RW0	MULW A, MULW A, @RW0	MULW A, MULW A, @RW0	MULW A, MULW A, @RW0	DIVU A, @RW0	DIVU A, @RW0	DIVUW A, DIVUW A, @RW0	DIVUW A, DIVUW A, @RW0	DIV A, @RW0	DIV A, @RW0	DIVW A, DIVW A, @RW0
+9	MULU A, @RW1	MULU A, MULU A, @RW1	MULUW A, MULUW A, @RW1	MULUW A, MULUW A, @RW1	MUL A, @RW1	MUL A, @RW1	MULW A, MULW A, @RW1	MULW A, MULW A, @RW1	MULW A, MULW A, @RW1	DIVU A, @RW1	DIVU A, @RW1	DIVUW A, DIVUW A, @RW1	DIVUW A, DIVUW A, @RW1	DIV A, @RW1	DIV A, @RW1	DIVW A, DIVW A, @RW1
+A	MULU A, @RW2	MULU A, MULU A, @RW2	MULUW A, MULUW A, @RW2	MULUW A, MULUW A, @RW2	MUL A, @RW2	MUL A, @RW2	MULW A, MULW A, @RW2	MULW A, MULW A, @RW2	MULW A, MULW A, @RW2	DIVU A, @RW2	DIVU A, @RW2	DIVUW A, DIVUW A, @RW2	DIVUW A, DIVUW A, @RW2	DIV A, @RW2	DIV A, @RW2	DIVW A, DIVW A, @RW2
+B	MULU A, @RW3	MULU A, MULU A, @RW3	MULUW A, MULUW A, @RW3	MULUW A, MULUW A, @RW3	MUL A, @RW3	MUL A, @RW3	MULW A, MULW A, @RW3	MULW A, MULW A, @RW3	MULW A, MULW A, @RW3	DIVU A, @RW3	DIVU A, @RW3	DIVUW A, DIVUW A, @RW3	DIVUW A, DIVUW A, @RW3	DIV A, @RW3	DIV A, @RW3	DIVW A, DIVW A, @RW3
+C	MULU A, @RW0+	MULU A, MULU A, @RW0+	MULUW A, MULUW A, @RW0+	MULUW A, MULUW A, @RW0+	MUL A, @RW0+	MUL A, @RW0+	MULW A, MULW A, @RW0+	MULW A, MULW A, @RW0+	MULW A, MULW A, @RW0+	DIVU A, @RW0+	DIVU A, @RW0+	DIVUW A, DIVUW A, @RW0+	DIVUW A, DIVUW A, @RW0+	DIV A, @RW0+	DIV A, @RW0+	DIVW A, DIVW A, @RW0+
+D	MULU A, @RW1+	MULU A, MULU A, @RW1+	MULUW A, MULUW A, @RW1+	MULUW A, MULUW A, @RW1+	MUL A, @RW1+	MUL A, @RW1+	MULW A, MULW A, @RW1+	MULW A, MULW A, @RW1+	MULW A, MULW A, @RW1+	DIVU A, @RW1+	DIVU A, @RW1+	DIVUW A, DIVUW A, @RW1+	DIVUW A, DIVUW A, @RW1+	DIV A, @RW1+	DIV A, @RW1+	DIVW A, DIVW A, @RW1+
+E	MULU A, @RW2+	MULU A, MULU A, @RW2+	MULUW A, MULUW A, @RW2+	MULUW A, MULUW A, @RW2+	MUL A, @RW2+	MUL A, @RW2+	MULW A, MULW A, @RW2+	MULW A, MULW A, @RW2+	MULW A, MULW A, @RW2+	DIVU A, @RW2+	DIVU A, @RW2+	DIVUW A, DIVUW A, @RW2+	DIVUW A, DIVUW A, @RW2+	DIV A, @RW2+	DIV A, @RW2+	DIVW A, DIVW A, @RW2+
+F	MULU A, @RW3+	MULU A, MULU A, @RW3+	MULUW A, MULUW A, @RW3+	MULUW A, MULUW A, @RW3+	MUL A, @RW3+	MUL A, @RW3+	MULW A, MULW A, @RW3+	MULW A, MULW A, @RW3+	MULW A, MULW A, @RW3+	DIVU A, @RW3+	DIVU A, @RW3+	DIVUW A, DIVUW A, @RW3+	DIVUW A, DIVUW A, @RW3+	DIV A, @RW3+	DIV A, @RW3+	DIVW A, DIVW A, @RW3+

表 C.9-15 MOVEA RWi, ea 命令 (第1バイト=79_H)[illegible]

[illegible]

付録 C 付録 命令

表 C.9-17 MOVW RWi, ea 命令 (第1バイト=7B_H)[illegible]

表 C.9-18 MOV Ri, ea 命令 (第1バイト = 7C_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	MOV R0, R0, @RW0+d8, R0	MOV R0, R1, @RW0-d8, R1	MOV R0, R1, @RW0-d8, R1	MOV R0, R2, @RW0+d8, R2	MOV R0, R3, @RW0+d8, R3	MOV R0, R4, @RW0+d8, R4	MOV R0, R5, @RW0+d8, R5	MOV R0, R6, @RW0+d8, R6	MOV R0, R7, @RW0+d8, R7	MOV R0, R8, @RW0+d8, R8	MOV R0, R9, @RW0+d8, R9	MOV R0, R10, @RW0+d8, R10	MOV R0, R11, @RW0+d8, R11	MOV R0, R12, @RW0+d8, R12	MOV R0, R13, @RW0+d8, R13	MOV R0, R14, @RW0+d8, R14
+1	MOV R1, R0, @RW1+d8, R0	MOV R1, R1, @RW1+d8, R1	MOV R1, R2, @RW1+d8, R2	MOV R1, R3, @RW1+d8, R3	MOV R1, R4, @RW1+d8, R4	MOV R1, R5, @RW1+d8, R5	MOV R1, R6, @RW1+d8, R6	MOV R1, R7, @RW1+d8, R7	MOV R1, R8, @RW1+d8, R8	MOV R1, R9, @RW1+d8, R9	MOV R1, R10, @RW1+d8, R10	MOV R1, R11, @RW1+d8, R11	MOV R1, R12, @RW1+d8, R12	MOV R1, R13, @RW1+d8, R13	MOV R1, R14, @RW1+d8, R14	MOV R1, R15, @RW1+d8, R15
+2	MOV R2, R0, @RW2+d8, R0	MOV R2, R1, @RW2+d8, R1	MOV R2, R2, @RW2+d8, R2	MOV R2, R3, @RW2+d8, R3	MOV R2, R4, @RW2+d8, R4	MOV R2, R5, @RW2+d8, R5	MOV R2, R6, @RW2+d8, R6	MOV R2, R7, @RW2+d8, R7	MOV R2, R8, @RW2+d8, R8	MOV R2, R9, @RW2+d8, R9	MOV R2, R10, @RW2+d8, R10	MOV R2, R11, @RW2+d8, R11	MOV R2, R12, @RW2+d8, R12	MOV R2, R13, @RW2+d8, R13	MOV R2, R14, @RW2+d8, R14	MOV R2, R15, @RW2+d8, R15
+3	MOV R3, R0, @RW3+d8, R0	MOV R3, R1, @RW3+d8, R1	MOV R3, R2, @RW3+d8, R2	MOV R3, R3, @RW3+d8, R3	MOV R3, R4, @RW3+d8, R4	MOV R3, R5, @RW3+d8, R5	MOV R3, R6, @RW3+d8, R6	MOV R3, R7, @RW3+d8, R7	MOV R3, R8, @RW3+d8, R8	MOV R3, R9, @RW3+d8, R9	MOV R3, R10, @RW3+d8, R10	MOV R3, R11, @RW3+d8, R11	MOV R3, R12, @RW3+d8, R12	MOV R3, R13, @RW3+d8, R13	MOV R3, R14, @RW3+d8, R14	MOV R3, R15, @RW3+d8, R15
+4	MOV R4, R0, @RW4+d8, R0	MOV R4, R1, @RW4+d8, R1	MOV R4, R2, @RW4+d8, R2	MOV R4, R3, @RW4+d8, R3	MOV R4, R4, @RW4+d8, R4	MOV R4, R5, @RW4+d8, R5	MOV R4, R6, @RW4+d8, R6	MOV R4, R7, @RW4+d8, R7	MOV R4, R8, @RW4+d8, R8	MOV R4, R9, @RW4+d8, R9	MOV R4, R10, @RW4+d8, R10	MOV R4, R11, @RW4+d8, R11	MOV R4, R12, @RW4+d8, R12	MOV R4, R13, @RW4+d8, R13	MOV R4, R14, @RW4+d8, R14	MOV R4, R15, @RW4+d8, R15
+5	MOV R5, R0, @RW5+d8, R0	MOV R5, R1, @RW5+d8, R1	MOV R5, R2, @RW5+d8, R2	MOV R5, R3, @RW5+d8, R3	MOV R5, R4, @RW5+d8, R4	MOV R5, R5, @RW5+d8, R5	MOV R5, R6, @RW5+d8, R6	MOV R5, R7, @RW5+d8, R7	MOV R5, R8, @RW5+d8, R8	MOV R5, R9, @RW5+d8, R9	MOV R5, R10, @RW5+d8, R10	MOV R5, R11, @RW5+d8, R11	MOV R5, R12, @RW5+d8, R12	MOV R5, R13, @RW5+d8, R13	MOV R5, R14, @RW5+d8, R14	MOV R5, R15, @RW5+d8, R15
+6	MOV R6, R0, @RW6+d8, R0	MOV R6, R1, @RW6+d8, R1	MOV R6, R2, @RW6+d8, R2	MOV R6, R3, @RW6+d8, R3	MOV R6, R4, @RW6+d8, R4	MOV R6, R5, @RW6+d8, R5	MOV R6, R6, @RW6+d8, R6	MOV R6, R7, @RW6+d8, R7	MOV R6, R8, @RW6+d8, R8	MOV R6, R9, @RW6+d8, R9	MOV R6, R10, @RW6+d8, R10	MOV R6, R11, @RW6+d8, R11	MOV R6, R12, @RW6+d8, R12	MOV R6, R13, @RW6+d8, R13	MOV R6, R14, @RW6+d8, R14	MOV R6, R15, @RW6+d8, R15
+7	MOV R7, R0, @RW7+d8, R0	MOV R7, R1, @RW7+d8, R1	MOV R7, R2, @RW7+d8, R2	MOV R7, R3, @RW7+d8, R3	MOV R7, R4, @RW7+d8, R4	MOV R7, R5, @RW7+d8, R5	MOV R7, R6, @RW7+d8, R6	MOV R7, R7, @RW7+d8, R7	MOV R7, R8, @RW7+d8, R8	MOV R7, R9, @RW7+d8, R9	MOV R7, R10, @RW7+d8, R10	MOV R7, R11, @RW7+d8, R11	MOV R7, R12, @RW7+d8, R12	MOV R7, R13, @RW7+d8, R13	MOV R7, R14, @RW7+d8, R14	MOV R7, R15, @RW7+d8, R15
+8	MOV @RW0, R0, @RW0+d16, R0	MOV @RW0, R1, @RW0+d16, R1	MOV @RW0, R2, @RW0+d16, R2	MOV @RW0, R3, @RW0+d16, R3	MOV @RW0, R4, @RW0+d16, R4	MOV @RW0, R5, @RW0+d16, R5	MOV @RW0, R6, @RW0+d16, R6	MOV @RW0, R7, @RW0+d16, R7	MOV @RW0, R8, @RW0+d16, R8	MOV @RW0, R9, @RW0+d16, R9	MOV @RW0, R10, @RW0+d16, R10	MOV @RW0, R11, @RW0+d16, R11	MOV @RW0, R12, @RW0+d16, R12	MOV @RW0, R13, @RW0+d16, R13	MOV @RW0, R14, @RW0+d16, R14	MOV @RW0, R15, @RW0+d16, R15
+9	MOV @RW1, R0, @RW1+d16, R0	MOV @RW1, R1, @RW1+d16, R1	MOV @RW1, R2, @RW1+d16, R2	MOV @RW1, R3, @RW1+d16, R3	MOV @RW1, R4, @RW1+d16, R4	MOV @RW1, R5, @RW1+d16, R5	MOV @RW1, R6, @RW1+d16, R6	MOV @RW1, R7, @RW1+d16, R7	MOV @RW1, R8, @RW1+d16, R8	MOV @RW1, R9, @RW1+d16, R9	MOV @RW1, R10, @RW1+d16, R10	MOV @RW1, R11, @RW1+d16, R11	MOV @RW1, R12, @RW1+d16, R12	MOV @RW1, R13, @RW1+d16, R13	MOV @RW1, R14, @RW1+d16, R14	MOV @RW1, R15, @RW1+d16, R15
+A	MOV @RW2, R0, @RW2+d16, R0	MOV @RW2, R1, @RW2+d16, R1	MOV @RW2, R2, @RW2+d16, R2	MOV @RW2, R3, @RW2+d16, R3	MOV @RW2, R4, @RW2+d16, R4	MOV @RW2, R5, @RW2+d16, R5	MOV @RW2, R6, @RW2+d16, R6	MOV @RW2, R7, @RW2+d16, R7	MOV @RW2, R8, @RW2+d16, R8	MOV @RW2, R9, @RW2+d16, R9	MOV @RW2, R10, @RW2+d16, R10	MOV @RW2, R11, @RW2+d16, R11	MOV @RW2, R12, @RW2+d16, R12	MOV @RW2, R13, @RW2+d16, R13	MOV @RW2, R14, @RW2+d16, R14	MOV @RW2, R15, @RW2+d16, R15
+B	MOV @RW3, R0, @RW3+d16, R0	MOV @RW3, R1, @RW3+d16, R1	MOV @RW3, R2, @RW3+d16, R2	MOV @RW3, R3, @RW3+d16, R3	MOV @RW3, R4, @RW3+d16, R4	MOV @RW3, R5, @RW3+d16, R5	MOV @RW3, R6, @RW3+d16, R6	MOV @RW3, R7, @RW3+d16, R7	MOV @RW3, R8, @RW3+d16, R8	MOV @RW3, R9, @RW3+d16, R9	MOV @RW3, R10, @RW3+d16, R10	MOV @RW3, R11, @RW3+d16, R11	MOV @RW3, R12, @RW3+d16, R12	MOV @RW3, R13, @RW3+d16, R13	MOV @RW3, R14, @RW3+d16, R14	MOV @RW3, R15, @RW3+d16, R15
+C	MOV @RW0+, R0, @RW0+RW7, R0	MOV @RW0+, R1, @RW0+RW7, R1	MOV @RW0+, R2, @RW0+RW7, R2	MOV @RW0+, R3, @RW0+RW7, R3	MOV @RW0+, R4, @RW0+RW7, R4	MOV @RW0+, R5, @RW0+RW7, R5	MOV @RW0+, R6, @RW0+RW7, R6	MOV @RW0+, R7, @RW0+RW7, R7	MOV @RW0+, R8, @RW0+RW7, R8	MOV @RW0+, R9, @RW0+RW7, R9	MOV @RW0+, R10, @RW0+RW7, R10	MOV @RW0+, R11, @RW0+RW7, R11	MOV @RW0+, R12, @RW0+RW7, R12	MOV @RW0+, R13, @RW0+RW7, R13	MOV @RW0+, R14, @RW0+RW7, R14	MOV @RW0+, R15, @RW0+RW7, R15
+D	MOV @RW1+, R0, @RW1+RW7, R0	MOV @RW1+, R1, @RW1+RW7, R1	MOV @RW1+, R2, @RW1+RW7, R2	MOV @RW1+, R3, @RW1+RW7, R3	MOV @RW1+, R4, @RW1+RW7, R4	MOV @RW1+, R5, @RW1+RW7, R5	MOV @RW1+, R6, @RW1+RW7, R6	MOV @RW1+, R7, @RW1+RW7, R7	MOV @RW1+, R8, @RW1+RW7, R8	MOV @RW1+, R9, @RW1+RW7, R9	MOV @RW1+, R10, @RW1+RW7, R10	MOV @RW1+, R11, @RW1+RW7, R11	MOV @RW1+, R12, @RW1+RW7, R12	MOV @RW1+, R13, @RW1+RW7, R13	MOV @RW1+, R14, @RW1+RW7, R14	MOV @RW1+, R15, @RW1+RW7, R15
+E	MOV @RW2+, R0, @PC+d16, R0	MOV @RW2+, R1, @PC+d16, R1	MOV @RW2+, R2, @PC+d16, R2	MOV @RW2+, R3, @PC+d16, R3	MOV @RW2+, R4, @PC+d16, R4	MOV @RW2+, R5, @PC+d16, R5	MOV @RW2+, R6, @PC+d16, R6	MOV @RW2+, R7, @PC+d16, R7	MOV @RW2+, R8, @PC+d16, R8	MOV @RW2+, R9, @PC+d16, R9	MOV @RW2+, R10, @PC+d16, R10	MOV @RW2+, R11, @PC+d16, R11	MOV @RW2+, R12, @PC+d16, R12	MOV @RW2+, R13, @PC+d16, R13	MOV @RW2+, R14, @PC+d16, R14	MOV @RW2+, R15, @PC+d16, R15
+F	MOV @RW3+, R0, addr16, R0	MOV @RW3+, R1, addr16, R1	MOV @RW3+, R2, addr16, R2	MOV @RW3+, R3, addr16, R3	MOV @RW3+, R4, addr16, R4	MOV @RW3+, R5, addr16, R5	MOV @RW3+, R6, addr16, R6	MOV @RW3+, R7, addr16, R7	MOV @RW3+, R8, addr16, R8	MOV @RW3+, R9, addr16, R9	MOV @RW3+, R10, addr16, R10	MOV @RW3+, R11, addr16, R11	MOV @RW3+, R12, addr16, R12	MOV @RW3+, R13, addr16, R13	MOV @RW3+, R14, addr16, R14	MOV @RW3+, R15, addr16, R15

付録 C 付録 命令

表 C.9-19 MOVW RWi, ea 命令 (第1バイト = 7D_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	MOVW RW0, RW0, @RW0-d8, RW0	MOVW RW0, RW1, @RW0-d8, RW1	MOVW RW0, RW2, @RW0-d8, RW2	MOVW RW0, RW3, @RW0-d8, RW3	MOVW RW0, RW4, @RW0-d8, RW4	MOVW RW0, RW5, @RW0-d8, RW5	MOVW RW0, RW6, @RW0-d8, RW6	MOVW RW0, RW7, @RW0-d8, RW7	MOVW RW0, RW4, @RW0-d8, RW4	MOVW RW0, RW5, @RW0-d8, RW5	MOVW RW0, RW6, @RW0-d8, RW6	MOVW RW0, RW7, @RW0-d8, RW7	MOVW RW0, RW8, @RW0-d8, RW8	MOVW RW0, RW9, @RW0-d8, RW9	MOVW RW0, RW10, @RW0-d8, RW10	MOVW RW0, RW11, @RW0-d8, RW11
+1	MOVW RW1, RW0, @RW1-d8, RW0	MOVW RW1, RW1, @RW1-d8, RW1	MOVW RW1, RW2, @RW1-d8, RW2	MOVW RW1, RW3, @RW1-d8, RW3	MOVW RW1, RW4, @RW1-d8, RW4	MOVW RW1, RW5, @RW1-d8, RW5	MOVW RW1, RW6, @RW1-d8, RW6	MOVW RW1, RW7, @RW1-d8, RW7	MOVW RW1, RW4, @RW1-d8, RW4	MOVW RW1, RW5, @RW1-d8, RW5	MOVW RW1, RW6, @RW1-d8, RW6	MOVW RW1, RW7, @RW1-d8, RW7	MOVW RW1, RW8, @RW1-d8, RW8	MOVW RW1, RW9, @RW1-d8, RW9	MOVW RW1, RW10, @RW1-d8, RW10	MOVW RW1, RW11, @RW1-d8, RW11
+2	MOVW RW2, RW0, @RW2-d8, RW0	MOVW RW2, RW1, @RW2-d8, RW1	MOVW RW2, RW2, @RW2-d8, RW2	MOVW RW2, RW3, @RW2-d8, RW3	MOVW RW2, RW4, @RW2-d8, RW4	MOVW RW2, RW5, @RW2-d8, RW5	MOVW RW2, RW6, @RW2-d8, RW6	MOVW RW2, RW7, @RW2-d8, RW7	MOVW RW2, RW4, @RW2-d8, RW4	MOVW RW2, RW5, @RW2-d8, RW5	MOVW RW2, RW6, @RW2-d8, RW6	MOVW RW2, RW7, @RW2-d8, RW7	MOVW RW2, RW8, @RW2-d8, RW8	MOVW RW2, RW9, @RW2-d8, RW9	MOVW RW2, RW10, @RW2-d8, RW10	MOVW RW2, RW11, @RW2-d8, RW11
+3	MOVW RW3, RW0, @RW3-d8, RW0	MOVW RW3, RW1, @RW3-d8, RW1	MOVW RW3, RW2, @RW3-d8, RW2	MOVW RW3, RW3, @RW3-d8, RW3	MOVW RW3, RW4, @RW3-d8, RW4	MOVW RW3, RW5, @RW3-d8, RW5	MOVW RW3, RW6, @RW3-d8, RW6	MOVW RW3, RW7, @RW3-d8, RW7	MOVW RW3, RW4, @RW3-d8, RW4	MOVW RW3, RW5, @RW3-d8, RW5	MOVW RW3, RW6, @RW3-d8, RW6	MOVW RW3, RW7, @RW3-d8, RW7	MOVW RW3, RW8, @RW3-d8, RW8	MOVW RW3, RW9, @RW3-d8, RW9	MOVW RW3, RW10, @RW3-d8, RW10	MOVW RW3, RW11, @RW3-d8, RW11
+4	MOVW RW4, RW0, @RW4-d8, RW0	MOVW RW4, RW1, @RW4-d8, RW1	MOVW RW4, RW2, @RW4-d8, RW2	MOVW RW4, RW3, @RW4-d8, RW3	MOVW RW4, RW4, @RW4-d8, RW4	MOVW RW4, RW5, @RW4-d8, RW5	MOVW RW4, RW6, @RW4-d8, RW6	MOVW RW4, RW7, @RW4-d8, RW7	MOVW RW4, RW4, @RW4-d8, RW4	MOVW RW4, RW5, @RW4-d8, RW5	MOVW RW4, RW6, @RW4-d8, RW6	MOVW RW4, RW7, @RW4-d8, RW7	MOVW RW4, RW8, @RW4-d8, RW8	MOVW RW4, RW9, @RW4-d8, RW9	MOVW RW4, RW10, @RW4-d8, RW10	MOVW RW4, RW11, @RW4-d8, RW11
+5	MOVW RW5, RW0, @RW5-d8, RW0	MOVW RW5, RW1, @RW5-d8, RW1	MOVW RW5, RW2, @RW5-d8, RW2	MOVW RW5, RW3, @RW5-d8, RW3	MOVW RW5, RW4, @RW5-d8, RW4	MOVW RW5, RW5, @RW5-d8, RW5	MOVW RW5, RW6, @RW5-d8, RW6	MOVW RW5, RW7, @RW5-d8, RW7	MOVW RW5, RW4, @RW5-d8, RW4	MOVW RW5, RW5, @RW5-d8, RW5	MOVW RW5, RW6, @RW5-d8, RW6	MOVW RW5, RW7, @RW5-d8, RW7	MOVW RW5, RW8, @RW5-d8, RW8	MOVW RW5, RW9, @RW5-d8, RW9	MOVW RW5, RW10, @RW5-d8, RW10	MOVW RW5, RW11, @RW5-d8, RW11
+6	MOVW RW6, RW0, @RW6-d8, RW0	MOVW RW6, RW1, @RW6-d8, RW1	MOVW RW6, RW2, @RW6-d8, RW2	MOVW RW6, RW3, @RW6-d8, RW3	MOVW RW6, RW4, @RW6-d8, RW4	MOVW RW6, RW5, @RW6-d8, RW5	MOVW RW6, RW6, @RW6-d8, RW6	MOVW RW6, RW7, @RW6-d8, RW7	MOVW RW6, RW4, @RW6-d8, RW4	MOVW RW6, RW5, @RW6-d8, RW5	MOVW RW6, RW6, @RW6-d8, RW6	MOVW RW6, RW7, @RW6-d8, RW7	MOVW RW6, RW8, @RW6-d8, RW8	MOVW RW6, RW9, @RW6-d8, RW9	MOVW RW6, RW10, @RW6-d8, RW10	MOVW RW6, RW11, @RW6-d8, RW11
+7	MOVW RW7, RW0, @RW7-d8, RW0	MOVW RW7, RW1, @RW7-d8, RW1	MOVW RW7, RW2, @RW7-d8, RW2	MOVW RW7, RW3, @RW7-d8, RW3	MOVW RW7, RW4, @RW7-d8, RW4	MOVW RW7, RW5, @RW7-d8, RW5	MOVW RW7, RW6, @RW7-d8, RW6	MOVW RW7, RW7, @RW7-d8, RW7	MOVW RW7, RW4, @RW7-d8, RW4	MOVW RW7, RW5, @RW7-d8, RW5	MOVW RW7, RW6, @RW7-d8, RW6	MOVW RW7, RW7, @RW7-d8, RW7	MOVW RW7, RW8, @RW7-d8, RW8	MOVW RW7, RW9, @RW7-d8, RW9	MOVW RW7, RW10, @RW7-d8, RW10	MOVW RW7, RW11, @RW7-d8, RW11
+8	MOVW @RW0, RW0, @RW0-d16, RW0	MOVW @RW0, RW1, +d16, RW1	MOVW @RW0, RW2, +d16, RW2	MOVW @RW0, RW3, +d16, RW3	MOVW @RW0, RW4, +d16, RW4	MOVW @RW0, RW5, +d16, RW5	MOVW @RW0, RW6, +d16, RW6	MOVW @RW0, RW7, +d16, RW7	MOVW @RW0, RW4, +d16, RW4	MOVW @RW0, RW5, +d16, RW5	MOVW @RW0, RW6, +d16, RW6	MOVW @RW0, RW7, +d16, RW7	MOVW @RW0, RW8, +d16, RW8	MOVW @RW0, RW9, +d16, RW9	MOVW @RW0, RW10, +d16, RW10	MOVW @RW0, RW11, +d16, RW11
+9	MOVW @RW1, RW0, @RW1-d16, RW0	MOVW @RW1, RW1, +d16, RW1	MOVW @RW1, RW2, +d16, RW2	MOVW @RW1, RW3, +d16, RW3	MOVW @RW1, RW4, +d16, RW4	MOVW @RW1, RW5, +d16, RW5	MOVW @RW1, RW6, +d16, RW6	MOVW @RW1, RW7, +d16, RW7	MOVW @RW1, RW4, +d16, RW4	MOVW @RW1, RW5, +d16, RW5	MOVW @RW1, RW6, +d16, RW6	MOVW @RW1, RW7, +d16, RW7	MOVW @RW1, RW8, +d16, RW8	MOVW @RW1, RW9, +d16, RW9	MOVW @RW1, RW10, +d16, RW10	MOVW @RW1, RW11, +d16, RW11
+A	MOVW @RW2, RW0, +d16, RW0	MOVW @RW2, RW1, +d16, RW1	MOVW @RW2, RW2, +d16, RW2	MOVW @RW2, RW3, +d16, RW3	MOVW @RW2, RW4, +d16, RW4	MOVW @RW2, RW5, +d16, RW5	MOVW @RW2, RW6, +d16, RW6	MOVW @RW2, RW7, +d16, RW7	MOVW @RW2, RW4, +d16, RW4	MOVW @RW2, RW5, +d16, RW5	MOVW @RW2, RW6, +d16, RW6	MOVW @RW2, RW7, +d16, RW7	MOVW @RW2, RW8, +d16, RW8	MOVW @RW2, RW9, +d16, RW9	MOVW @RW2, RW10, +d16, RW10	MOVW @RW2, RW11, +d16, RW11
+B	MOVW @RW3, RW0, +d16, RW0	MOVW @RW3, RW1, +d16, RW1	MOVW @RW3, RW2, +d16, RW2	MOVW @RW3, RW3, +d16, RW3	MOVW @RW3, RW4, +d16, RW4	MOVW @RW3, RW5, +d16, RW5	MOVW @RW3, RW6, +d16, RW6	MOVW @RW3, RW7, +d16, RW7	MOVW @RW3, RW4, +d16, RW4	MOVW @RW3, RW5, +d16, RW5	MOVW @RW3, RW6, +d16, RW6	MOVW @RW3, RW7, +d16, RW7	MOVW @RW3, RW8, +d16, RW8	MOVW @RW3, RW9, +d16, RW9	MOVW @RW3, RW10, +d16, RW10	MOVW @RW3, RW11, +d16, RW11
+C	MOVW @RW0+, RW0, +RW7, RW0	MOVW @RW0+, RW1, +RW7, RW1	MOVW @RW0+, RW2, +RW7, RW2	MOVW @RW0+, RW3, +RW7, RW3	MOVW @RW0+, RW4, +RW7, RW4	MOVW @RW0+, RW5, +RW7, RW5	MOVW @RW0+, RW6, +RW7, RW6	MOVW @RW0+, RW7, +RW7, RW7	MOVW @RW0+, RW4, +RW7, RW4	MOVW @RW0+, RW5, +RW7, RW5	MOVW @RW0+, RW6, +RW7, RW6	MOVW @RW0+, RW7, +RW7, RW7	MOVW @RW0+, RW8, +RW7, RW8	MOVW @RW0+, RW9, +RW7, RW9	MOVW @RW0+, RW10, +RW7, RW10	MOVW @RW0+, RW11, +RW7, RW11
+D	MOVW @RW1+, RW0, +RW7, RW0	MOVW @RW1+, RW1, +RW7, RW1	MOVW @RW1+, RW2, +RW7, RW2	MOVW @RW1+, RW3, +RW7, RW3	MOVW @RW1+, RW4, +RW7, RW4	MOVW @RW1+, RW5, +RW7, RW5	MOVW @RW1+, RW6, +RW7, RW6	MOVW @RW1+, RW7, +RW7, RW7	MOVW @RW1+, RW4, +RW7, RW4	MOVW @RW1+, RW5, +RW7, RW5	MOVW @RW1+, RW6, +RW7, RW6	MOVW @RW1+, RW7, +RW7, RW7	MOVW @RW1+, RW8, +RW7, RW8	MOVW @RW1+, RW9, +RW7, RW9	MOVW @RW1+, RW10, +RW7, RW10	MOVW @RW1+, RW11, +RW7, RW11
+E	MOVW @RW2+, RW0, +d16, RW0	MOVW @RW2+, RW1, +d16, RW1	MOVW @RW2+, RW2, +d16, RW2	MOVW @RW2+, RW3, +d16, RW3	MOVW @RW2+, RW4, +d16, RW4	MOVW @RW2+, RW5, +d16, RW5	MOVW @RW2+, RW6, +d16, RW6	MOVW @RW2+, RW7, +d16, RW7	MOVW @RW2+, RW4, +d16, RW4	MOVW @RW2+, RW5, +d16, RW5	MOVW @RW2+, RW6, +d16, RW6	MOVW @RW2+, RW7, +d16, RW7	MOVW @RW2+, RW8, +d16, RW8	MOVW @RW2+, RW9, +d16, RW9	MOVW @RW2+, RW10, +d16, RW10	MOVW @RW2+, RW11, +d16, RW11
+F	MOVW @RW3+, RW0, +d16, RW0	MOVW @RW3+, RW1, +d16, RW1	MOVW @RW3+, RW2, +d16, RW2	MOVW @RW3+, RW3, +d16, RW3	MOVW @RW3+, RW4, +d16, RW4	MOVW @RW3+, RW5, +d16, RW5	MOVW @RW3+, RW6, +d16, RW6	MOVW @RW3+, RW7, +d16, RW7	MOVW @RW3+, RW4, +d16, RW4	MOVW @RW3+, RW5, +d16, RW5	MOVW @RW3+, RW6, +d16, RW6	MOVW @RW3+, RW7, +d16, RW7	MOVW @RW3+, RW8, +d16, RW8	MOVW @RW3+, RW9, +d16, RW9	MOVW @RW3+, RW10, +d16, RW10	MOVW @RW3+, RW11, +d16, RW11

[illegible]

付録 C 付録 命令

表 C.9-21 XCHW RWi, ea 命令 (第1バイト = 7FH)

[illegible]

索引

Numerics

16 ビット PPG タイマ	
16 ビット PPG タイマ (× 1)	339
16 ビット PPG タイマ (× 3)	316
16 ビット PPG タイマ使用上の注意	335
16 ビット PPG タイマ端子の ブロックダイアグラム	318
16 ビット PPG タイマの EI ² OS 機能	330
16 ビット PPG タイマの端子	318
16 ビット PPG タイマの ブロックダイアグラム	317
16 ビット PPG タイマのレジスタ	320
16 ビット PPG タイマの割込み	329
16 ビット PPG タイマ割込みと EI ² OS	330
16 ビットアウトプットコンペア	
16 ビットアウトプットコンペア (× 6)	338
16 ビットアウトプットコンペア タイミング	399
16 ビットアウトプットコンペアの使用上の 注意	414
16 ビットアウトプットコンペアの動作	395
16 ビットアウトプットコンペアの ブロックダイアグラム	342
16 ビットアウトプットコンペアの レジスタ	348
16 ビットアウトプットコンペア割込み	383
16 ビットアウトプットコンペア割込みと EI ² OS	383
16 ビットインプットキャプチャ	
16 ビットインプットキャプチャ (× 4)	339
16 ビットインプットキャプチャ使用上の 注意	415
16 ビットインプットキャプチャ入力 タイミング	402
16 ビットインプットキャプチャの動作	401
16 ビットインプットキャプチャの ブロックダイアグラム	342
16 ビットインプットキャプチャ割込み	384
16 ビットインプットキャプチャ割込みと EI ² OS	384
16 ビットタイマ制御レジスタ	
16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2)	376
16 ビットタイマ制御レジスタ (DTCR1)	378
16 ビットタイマレジスタ	
16 ビットタイマレジスタ (TMR0/TMR1)	266
16 ビットタイマレジスタ (TMRR0 ~ TMRR2)	375
16 ビットフリーランタイム	
16 ビットフリーランタイム (× 1)	338
16 ビットフリーランタイム使用上の注意	414
16 ビットフリーランタイムの ブロックダイアグラム	341
16 ビットフリーランタイムのレジスタ	347
16 ビットフリーランタイム割込み	382
16 ビットフリーランタイム割込みと EI ² OS	382
16 ビットリロードタイマ	
16 ビットリロードタイマの EI ² OS 機能	268
16 ビットリロードタイマの使用上の注意	279
16 ビットリロードタイマの設定	269
16 ビットリロードタイマの端子	260
16 ビットリロードタイマの動作モード	256
16 ビットリロードタイマの ブロックダイアグラム	258
16 ビットリロードタイマのレジスタ	261
16 ビットリロードタイマの割込み	268
16 ビットリロードタイマの割込みと EI ² OS	268
イベントカウントモード	277
イベントカウントモード (外部クロックモード)	257
カウンタ動作	257
カウンタの動作状態	270
内部クロックモード	256
内部タイマ (16 ビットリロードタイマ) による ポーレート	521
16 ビットリロードレジスタ	
16 ビットリロードレジスタ (TMRDL0/TMRDL1, TMRDH0/TMRDH1)	267
512K/1024K ビットフラッシュメモリ	
512K/1024K ビットフラッシュメモリの 概要	554
512K/1024K ビットフラッシュメモリの 特長	554
8/10 ビット A/D コンバータ	
8/10 ビット A/D コンバータ使用上の注意	477
8/10 ビット A/D コンバータの A/D 変換データ 保護機能の説明	474
8/10 ビット A/D コンバータの EI ² OS	465
8/10 ビット A/D コンバータの機能	444
8/10 ビット A/D コンバータの端子	449
8/10 ビット A/D コンバータの ブロックダイアグラム	446
8/10 ビット A/D コンバータの変換モード	445
8/10 ビット A/D コンバータのレジスタと 初期値の一覧	450
8/10 ビット A/D コンバータの割込みと EI ² OS	465

A

A

アキュムレータ (A)	38
A/D コンバータ	
A/D コンバータの割込み	465
A/D 制御ステータスレジスタ	
A/D 制御ステータスレジスタ下位 (ADCS0)	455
A/D 制御ステータスレジスタ上位 (ADCS1)	451
A/D セットアップレジスタ	
A/D セットアップレジスタ (ADSR0/ADSR1)	458
A/D データレジスタ	
A/D データレジスタ (ADCR0/ADCR1)	457
A/D 変換データ保護機能	
8/10 ビット A/D コンバータの A/D 変換データ 保護機能の説明	474
ADB	
バンクセレクトプリフィックス (PCB, DTB, ADB, SPB)	54
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	50
ADCR	
A/D データレジスタ (ADCR0/ADCR1)	457
ADCS	
A/D 制御ステータスレジスタ下位 (ADCS0)	455
A/D 制御ステータスレジスタ上位 (ADCS1)	451
単発変換モード (ADCS: MD1,MD0=00 _B または 01 _B)	466
停止変換モード (ADCS: MD1,MD0=11 _B)	466
連続変換モード (ADCS: MD1,MD0=10 _B)	466
ADER	
アナログ入力許可レジスタ (ADER0/ADER1)	463
ADSR	
A/D セットアップレジスタ (ADSR0/ADSR1)	458

B

BAP

バッファアドレスポインタ (BAP)	155
--------------------	-----

C

CCR

コンディションコードレジスタ (CCR) の 構成	44
------------------------------	----

CDCR

通信プリスケラ制御レジスタ (CDCR)	508
----------------------	-----

CKSCR

クロック選択レジスタ (CKSCR) の構成	79
------------------------	----

CMR

コモンレジスタバンクプリフィックス (CMR)	56
----------------------------	----

CPCLR

コンペアクリアレジスタ (CPCLR)	352
---------------------	-----

CPCLRB

コンペアクリアバッファレジスタ (CPCLRB)	351
-----------------------------	-----

CPU

CPU 間欠動作モード	101, 107
CPU 動作モードおよび消費電流	100
CPU の概要	24
CPU モード端子 (MD2 ~ MD0)	165

CSVCR

クロックスーパーバイザ制御レジスタ (CSVCR)	92
------------------------------	----

CT2 ~ CT0 ビット

コンペア時間の設定 (CT2 ~ CT0 ビット)	462
---------------------------	-----

D

D/A コンバータ

D/A コンバータの機能と動作	480
D/A コンバータの端子	482
D/A コンバータのブロックダイアグラム	481

D/A コンバータレジスタ

D/A コンバータレジスタ	483
D/A コンバータレジスタ 0(DAT0)	485
D/A コンバータレジスタ 1(DAT1)	484

D/A 制御レジスタ

D/A 制御レジスタ 0(DACR0)	487
D/A 制御レジスタ 1(DACR1)	486

DACR

D/A 制御レジスタ 0(DACR0)	487
D/A 制御レジスタ 1(DACR1)	486

DAT

D/A コンバータレジスタ 0(DAT0)	485
D/A コンバータレジスタ 1(DAT1)	484

DCT

データカウンタ (DCT)	153
---------------	-----

DIRR

遅延割込み発生モジュールレジスタ (DIRR)	419
----------------------------	-----

DIV

分周比制御レジスタ (DIV0/DIV1)	294
-----------------------	-----

DPR

ダイレクトページレジスタ (DPR)	49
--------------------	----

DQ3

セクタ消去の場合	568
----------	-----

DQ5

書込みとチップ消去 / セクタ消去の場合	567
----------------------	-----

DQ6

書込みとチップ消去 / セクタ消去の場合	566
セクタ消去一時停止の場合	566

DQ7

書込みの場合	564
セクタ消去一時停止の場合	564
チップ消去 / セクタ消去の場合	564

DTB

バンクセレクトプリフィックス (PCB, DTB, ADB, SPB)	54
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	50

MB90820B シリーズ

DTCR

16 ビットタイマ制御レジスタ (DTCR0 ~ DTCR2)	376
---------------------------------	-----

DTCR1

16 ビットタイマ制御レジスタ (DTCR1)	378
-------------------------	-----

DTP

DTP/ 外部割込みの設定	436
---------------	-----

DTP/ 外部割込み

DTP/ 外部割込み動作	437
DTP/ 外部割込みの端子	428
DTP/ 外部割込みのブロックダイアグラム	426

DTP/ 外部割込み機能

DTP/ 外部割込み機能	424
DTP/ 外部割込み機能および EI ² OS の割込み	425

DTP/ 外部割込み許可レジスタ

DTP/ 外部割込み許可レジスタ (ENIR)	432
-------------------------	-----

DTP/ 外部割込み要因レジスタ

DTP/ 外部割込み要因レジスタ (EIRR)	431
-------------------------	-----

DTP 機能

DTP 機能	440
--------	-----

DTTI 端子

DTTI 端子入力の動作	412
DTTI 端子ノイズキャンセル機能	413

DTTI 割込み

DTTI 割込み	413
----------	-----

E

E²PROM

E ² PROM メモリマップ	545
----------------------------	-----

EI²OS

16 ビット PPG タイマの EI ² OS 機能	330
16 ビット PPG タイマ割込みと EI ² OS	330
16 ビットアウトプットコンペア割込みと EI ² OS	383
16 ビットインプットキャプチャ割込みと EI ² OS	384
16 ビットフリーランタイム割込みと EI ² OS	382
16 ビットリロードタイマの EI ² OS 機能	268
16 ビットリロードタイマの割込みと EI ² OS	268
8/10 ビット A/D コンバータの EI ² OS	465
8/10 ビット A/D コンバータの割込みと EI ² OS	465
DTP/ 外部割込み機能および EI ² OS の割込み	425
EI ² OS 機能を使用した変換動作	473
PWC タイマの EI ² OS 機能	296
PWC タイマ割込みと EI ² OS	295
UART の EI ² OS 機能	511
UART の割込みと EI ² OS	511
UART 割込みおよび EI ² OS	491
拡張インテリジェント I/O サービス (EI ² OS)	150
拡張インテリジェント I/O サービス (EI ² OS) 処理時間 (1 回の転送時間)	158
拡張インテリジェント I/O サービス (EI ² OS) ステータスレジスタ (ISCS)	154
拡張インテリジェント I/O サービス (EI ² OS) ディスクリプタ (ISD) の構成	152

拡張インテリジェント I/O サービス (EI²OS)

の使用手順	157
-------	-----

拡張インテリジェント I/O サービス (EI²OS)

の処理手順	156
-------	-----

拡張インテリジェント I/O サービス (EI²OS)

の動作	151
-----	-----

タイムベースタイマの割込みと

EI ² OS の対応	240
------------------------	-----

多機能タイマの EI²OS 機能

	386
--	-----

波形ジェネレータ割込みと EI²OS

	385
--	-----

EIRR

DTP/ 外部割込み要因レジスタ (EIRR)	431
-------------------------	-----

ELVR

要求レベル設定レジスタ (ELVR)	434
--------------------	-----

ENIR

DTP/ 外部割込み許可レジスタ (ENIR)	432
-------------------------	-----

F

F²MC-16LX

F ² MC-16LX 命令一覧表	624
------------------------------	-----

FMCS

フラッシュメモリ制御ステータスレジスタ (FMCS)	558
----------------------------	-----

FPT-80

FPT-80P-M21/FPT-80P-M22 端子配列図	9
-------------------------------	---

FPT-80P-M06

FPT-80P-M06 端子配列図	8
-------------------	---

FPT-80P-M06 の外形寸法図	11
--------------------	----

FPT-80P-M21

FPT-80P-M21 の外形寸法図	10
--------------------	----

FPT-80P-M22

FPT-80P-M22 の外形寸法図	12
--------------------	----

G

GATE

RTO0 ~ RTO5 と GATE の出力状態	403
--------------------------	-----

GATE 信号

GTEN _x がアクティブであり各 RT _x が "H" であるときの GATE 信号生成	404
---	-----

GTEN _x がアクティブであるときの RT _x 立上りエッジから 16 ビットタイマ 0/1/2 アンダフローまでにおける GATE 信号の生成	405
---	-----

I

I/O

I/O 領域	26
--------	----

I/O アドレスポインタ

I/O アドレスポインタ (IOA)	153
--------------------	-----

I/O 端子

I/O 端子と端子機能	13
-------------	----

I/O ポート

I/O ポートのレジスター一覧	172
-----------------	-----

I/O マップ

I/O マップ	580
---------	-----

ICSH23

インプットキャプチャ状態制御レジスタ, 上位バイト (ICSH23)	368
------------------------------------	-----

ICSL23	
インプットキャプチャ状態制御レジスタ ,	
下位バイト (ICSL23)	369
ILM	
割込みレベルマスクレジスタ (ILM)	47
INT9	
INT9 割込み	546
IOA	
I/O アドレスポインタ (IOA)	153
IPCP0 ~ IPCP3	
インプットキャプチャレジスタ	
(IPCP0 ~ IPCP3)	367
ISCS	
拡張インテリジェント I/O サービス (EI ² OS)	
ステータスレジスタ (ISCS)	154
ISD	
拡張インテリジェント I/O サービス (EI ² OS)	
ディスクリプタ (ISD) の構成	152

L

LPMCR	
低消費電力モード制御レジスタ (LPMCR)	104

M

MB90820B シリーズ	
MB90820B シリーズの特長	2
MB90820B シリーズの品種構成	5
MB90820B シリーズのブロックダイアグラム	
.....	7
リソース	3
MD0	
CPU モード端子 (MD2 ~ MD0)	165
MD1	
CPU モード端子 (MD2 ~ MD0)	165
MD2	
CPU モード端子 (MD2 ~ MD0)	165

N

NCC	
フラグ変化抑止プリフィックス (NCC)	57

O

OCCP	
アウトプットコンペアレジスタ	
(OCCP0 ~ OCCP5)	361
OCCPB	
アウトプットコンペアバッファレジスタ	
(OCCPB0 ~ OCCPB)	360
OCS	
コンペア制御レジスタ, 下位バイト	
(OCS0/OCS2/OCS4)	365
コンペア制御レジスタ, 上位バイト	
(OCS1/OCS3/OCS5)	362

P

PACSR	
プログラムアドレス検出制御ステータスレジスタ	
(PACSR)	542
PADR0/PADR1	
プログラムアドレス検出レジスタ 0/1	
(PADR0/PADR1)	541
PC	
プログラムカウンタ (PC)	48
PCB	
バンクセレクトプリフィックス	
(PCB, DTB, ADB, SPB)	54
バンクレジスタ	
(PCB, DTB, USB, SSB, ADB)	50
PCKCR	
PLL クロック制御レジスタ (PCKCR) の構成	
.....	81
PCNTH	
PPG 状態制御レジスタ, 上位	
(PCNTH0 ~ PCNTH2)	325
PCNTL	
PPG 状態制御レジスタ, 下位	
(PCNTL1 ~ PCNTL3)	327
ワンショットモード	
(PCNTL: MDSE = 1)	332
PDCR	
PPG ダウンカウンタレジスタ	
(PDCR0 ~ PDCR2)	322
PICSH01	
PPG 出力制御 / インプットキャプチャ状態制御	
レジスタ, 上位バイト (PICSH01)	371
PICSL01	
インプットキャプチャ状態制御レジスタ,	
下位バイト (PICSL01)	373
PLL クロック	
PLL クロック逡倍率の選択	83
PLL クロック制御レジスタ	
PLL クロック制御レジスタ (PCKCR)	
の構成	81
PLL クロックモード	
メインクロックモード, PLL クロックモード	
.....	83
PPG	
ゲートトリガ	
(PPG チャンネル 0 の場合のみ)	333
PPG0	
PPG0 出力制御	404
ゲートトリガされた PPG0 出力	404
PPG0 出力パルス	
RT 立上りエッジから 16 ビットタイマアンダ	
フローまでにおける PPG0 出力パルス	407
PPG 出力制御 / インプットキャプチャ状態制御レジスタ	
PPG 出力制御 / インプットキャプチャ状態制御	
レジスタ, 上位バイト (PICSH01)	371
PPG 状態制御レジスタ	
PPG 状態制御レジスタ, 下位	
(PCNTL1 ~ PCNTL3)	327
PPG 状態制御レジスタ, 上位	
(PCNTH0 ~ PCNTH2)	325

PPG ダウンカウンタレジスタ	
PPG ダウンカウンタレジスタ	
(PDCR0 ~ PDCR2)	322
PPG 割込み	
PPG 割込み	334
PWC	
PWC データバッファレジスタ	
(PWC0/PWC1)	293
PWCSH	
PWC 状態制御レジスタ, 上位バイト	
(PWCSH0/PWCSH1)	288
PWCSL	
PWC 状態制御レジスタ, 下位バイト	
(PWCSL0/PWCSL1)	291
PWC 状態制御レジスタ	
PWC 状態制御レジスタ, 下位バイト	
(PWCSL0/PWCSL1)	291
PWC 状態制御レジスタ, 上位バイト	
(PWCSH0/PWCSH1)	288
PWC タイマ	
PWC タイマ	282
PWC タイマ使用上の注意	313
PWC タイマ端子のブロックダイアグラム	
.....	284
PWC タイマの EI ² OS 機能	296
PWC タイマの端子	284
PWC タイマの動作	282
PWC タイマのブロックダイアグラム	283
PWC タイマのレジスタ	287
PWC タイマの割込み	295
PWC タイマ割込みと EI ² OS	295
タイマ機能	297
タイマクリア	303
タイマ周期	305
パルス幅測定機能	298
PWC データバッファレジスタ	
PWC データバッファレジスタ (PWC0/PWC1)	
.....	293
PWM モード	
PWM モード (PCNTL: MDSE = 0)	331

R

RAM	
RAM 領域	26
ROM	
ROM 修正機能レジスタ	540
ROM 修正割込み	538
ROM 領域	26
ROMM	
ROM ミラー機能選択レジスタ (ROMM)	551
ROM 修正機能	
ROM 修正機能の動作説明	544
ROM 修正機能のブロックダイアグラム	539
ROM ミラー機能選択モジュール	
ROM ミラー機能選択モジュールのレジスタ	
.....	550
ROM ミラー機能選択モジュール	
ブロックダイアグラム	550
ROM ミラー機能選択レジスタ	
ROM ミラー機能選択レジスタ (ROMM)	551

RP	
汎用レジスタ領域とレジスタバンクポインタ	
(RP)	46
レジスタバンクポインタ (RP)	46
RTO	
RTO0 ~ RTO5 と GATE の出力状態	403

S

SCR	
シリアル制御レジスタ (SCR0/SCR1)	499
SIDR	
シリアル入力データレジスタ	
(SIDR/SIDR1)	506
SIGCR	
波形制御レジスタ (SIGCR)	380
SMR	
シリアルモードレジスタ (SMR0/SMR1)	501
SODR	
シリアル出力データレジスタ	
(SODR0/SODR1)	507
SPB	
バンクセレクトプリフィックス	
(PCB, DTB, ADB, SPB)	54
SSB	
バンクレジスタ	
(PCB, DTB, USB, SSB, ADB)	50
SSP	
システムスタックポインタ (SSP)	42
SSR	
シリアルステータスレジスタ	
(SSR0/SSR1)	503
ST2 ~ ST0 ビット	
サンプリング時間の設定	
(ST2 ~ ST0 ビット)	461

T

TBTC	
タイムベースタイマ制御レジスタ	
(TBTC)	238
TCCSH	
タイマ状態制御レジスタ, 上位バイト	
(TCCSH)	354
TCCSL	
タイマ状態制御レジスタ, 下位バイト	
(TCCSL)	357
TCDT	
タイマデータレジスタ (TCDT)	353
TMCSRH	
タイマ制御ステータスレジスタ上位ビットおよび	
bit7(TMCSRH0/TMCSRH1)	262
TMCSRL	
タイマ制御ステータスレジスタ下位	
(TMCSRL0/TMCSRL1)	264
TMR	
16 ビットタイマレジスタ (TMR0/TMR1)	266
TMRD	
16 ビットリロードレジスタ (TMRDL0/TMRDL1,	
TMRDH0/TMRDH1)	267

TMRR

16 ビットタイマレジスタ (TMRR0 ~ TMRR2)	375
-------------------------------	-----

U

UART

UART 使用上の注意	536
UART の EI ² OS 機能	511
UART の機能 (× 2)	490
UART の端子	495
UART の端子のブロックダイアグラム	496
UART の動作	524
UART のブロックダイアグラム	492
UART のボーレートの選択	516
UART のレジスタ	498
UART の割込み	510
UART の割込みと EI ² OS	511
UART 割込みおよび EI ² OS	491

USB

バンクレジスタ	
(PCB, DTB, USB, SSB, ADB)	50

USP

ユーザスタックポインタ (USP)	42
-------------------	----

W

WDTC

ウォッチドッグタイマ制御レジスタ	
(WDTC)	249

あ

アウトプットコンペアバッファレジスタ	
アウトプットコンペアバッファレジスタ	
(OCCPB0 ~ OCCPB5)	360
アウトプットコンペアレジスタ	
アウトプットコンペアレジスタ	
(OCCP0 ~ OCCP5)	361
アキュムレータ	
アキュムレータ (A)	38
アクセス空間	
バンクレジスタとアクセス空間	31
アドレス	
アドレス指定方法	571
アドレス引用指定	
32 ビット汎用レジスタの下位 24 ビットアドレス	
引用指定 (間接指定)	30
アドレス指定	
24 ビットオペランドによるリニア方式アドレス	
指定	30
アドレス指定方法	571
バンク方式によるアドレス指定と	
デフォルト空間	32
アドレッシング	
アドレッシング	601
間接アドレッシング	610
直接アドレッシング	603
アナログ入力許可レジスタ	
アナログ入力許可レジスタ	
(ADER0/ADER1)	463

い

一時停止	
セクタ消去一時停止の場合	564
フラッシュメモリのセクタ消去を	
一時停止する	576
イベントカウントモード	
イベントカウントモード	
(外部クロックモード)	257
インターバルタイマ	
インターバルタイマ機能	234
インターバルタイマ機能	
(タイムベースタイマ)	241
インプットキャプチャ	
インプットキャプチャのレジスタ	349
インプットキャプチャ状態制御レジスタ	
インプットキャプチャ状態制御レジスタ,	
下位バイト (PICSL01)	373
インプットキャプチャ状態制御レジスタ,	
下位バイト (ICSL23)	369
インプットキャプチャ状態制御レジスタ,	
上位バイト (ICSH23)	368
インプットキャプチャレジスタ	
インプットキャプチャレジスタ	
(IPCP0 ~ IPCP3)	367

う

ウォッチドッグタイマ	
ウォッチドッグタイマ機能	246
ウォッチドッグタイマ使用上の注意	253

MB90820B シリーズ

ウォッチドッグタイマの動作	251
ウォッチドッグタイマの ブロックダイアグラム	248
ウォッチドッグタイマ制御レジスタ ウォッチドッグタイマ制御レジスタ (WDTC)	249

か

外形寸法図	
FPT-80P-M06 の外形寸法図	11
FPT-80P-M21 の外形寸法図	10
FPT-80P-M22 の外形寸法図	12
外部カウントクロック	
選択された外部カウントクロック	393
外部クロック	
外部クロックによるポーレート	523
振動子と外部クロックの接続	86
外部クロックモード	
イベントカウントモード (外部クロックモード)	257
外部リセット端子	
外部リセット端子のブロックダイアグラム	66
外部割込み	
DTP/ 外部割込みの設定	436
外部割込み機能	439
カウントクロック周期	
カウントクロック周期と最大周期	305
書込み / 消去	
フラッシュメモリ書込み / 消去の詳細説明	569
拡張インテリジェント I/O サービス	
拡張インテリジェント I/O サービス (EI ² OS)	150
拡張インテリジェント I/O サービス (EI ² OS) 処理時間 (1 回の転送時間)	158
拡張インテリジェント I/O サービス (EI ² OS) ステータスレジスタ (ISCS)	154
拡張インテリジェント I/O サービス (EI ² OS) ディスクリプタ (ISD) の構成	152
拡張インテリジェント I/O サービス (EI ² OS) の使用手順	157
拡張インテリジェント I/O サービス (EI ² OS) の処理手順	156
拡張インテリジェント I/O サービス (EI ² OS) の動作	151
間接アドレッシング	
間接アドレッシング	610
間接指定	
32 ビット汎用レジスタの下位 24 ビットアドレス 引用指定 (間接指定)	30

き

機能	
8/10 ビット A/D コンバータの機能	444

く

クロック	
クロック供給機能	235
クロックの概要	74
クロックスーパバイザ	
クロックスーパバイザのブロック ダイアグラム	89
クロックスーパバイザの概要	88
クロックスーパバイザの動作	94
クロックスーパバイザのレジスタ一覧	91
クロックスーパバイザ制御レジスタ クロックスーパバイザ制御レジスタ (CSVCR)	92
クロック選択レジスタ	
クロック選択レジスタ	78
クロック選択レジスタ (CKSCR) の構成	79
クロック同期モード	
クロック同期モードでの動作	529
クロック発生部	
クロック発生部のブロックダイアグラム	76
クロックモード	
クロックモード	101
クロックモードの移行	83

け

ゲートトリガ	
ゲートトリガ (PPG チャネル 0 の場合のみ)	333
ゲートトリガされた PPG0 出力	404

こ

コマンドシーケンス	
コマンドシーケンス表	561
コモンレジスタバンクプリフィックス コモンレジスタバンクプリフィックス (CMR)	56
コンディションコードレジスタ	
コンディションコードレジスタ (CCR) の 構成	44
コンペアクリアパッファ	
コンペアクリアパッファ	390
コンペアクリアパッファレジスタ	
コンペアクリアパッファレジスタ (CPCLR)	351
コンペアクリアレジスタ	
コンペアクリアレジスタ (CPCLR)	352
コンペア時間	
コンペア時間の設定 (CT2 ~ CT0 ビット)	462
コンペア制御レジスタ	
コンペア制御レジスタ, 下位バイト (OCS0/OCS2/OCS4)	365
コンペア制御レジスタ, 上位バイト (OCS1/OCS3/OCS5)	362

さ

再開

フラッシュメモリのセクタ消去を 再開する	577
サンプリング時間 サンプリング時間の設定 (ST2 ~ ST0 ビット)	461

し

システム構成 システム構成	545
システムスタックポインタ システムスタックポインタ (SSP)	42
実効アドレス 実効アドレスフィールド	619
実効アドレスフィールド 実効アドレスフィールド	602
実行サイクル 実行サイクル数	617
実行サイクル数計算方法	617
周波数 発振クロック周波数とシリアルクロック 入力周波数	590
受信割込み要求 受信割込み要求の発生とフラグセットの タイミング	512
使用上の注意 16 ビット PPG タイマ使用上の注意	335
16 ビットアウトプットコンペアの使用上の 注意	414
16 ビットインプットキャプチャ使用上の 注意	415
16 ビットフリーランタイム使用上の注意	414
16 ビットリロードタイマの使用上の注意	279
PWC タイマ使用上の注意	313
使用上の注意	441
遅延割込み要求ラッチの使用上の注意	421
波形ジェネレータ使用上の注意	415
状態遷移図 状態遷移図	116
消費電流 CPU 動作モードおよび消費電流	100
初期状態 初期状態	545
初期値 8/10 ビット A/D コンバータのレジスタと初期値 の一覧	450
シリアルクロック 発振クロック周波数とシリアルクロック 入力周波数	590
シリアル出力データレジスタ シリアル出力データレジスタ (SODR0/SODR1)	507
シリアルステータスレジスタ シリアルステータスレジスタ (SSR0/SSR1)	503
シリアル制御レジスタ シリアル制御レジスタ (SCR0/SCR1)	499

シリアル入力データレジスタ

シリアル入力データレジスタ (SIDR0/SIDR1)	506
シリアルモードレジスタ シリアルモードレジスタ (SMR0/SMR1)	501
シングルチップモード シングルチップモード時の端子状態	119

す

スタック

スタックの設定	41
スタック領域	162
割込み処理開始時のスタック動作	161
割込み処理からの復帰時のスタック動作	161
スタンバイモード スタンバイモード	101
スタンバイモードの設定	121
スタンバイモードの動作状態	108
スタンバイモードへの移行と割込み	120
割込みによるスタンバイモードの解除	120

ストップモード

ストップモード解除	121
ストップモードの解除	114
ストップモードへの移行	114

スリープモード

スリープモードの解除	109
スリープモードへの遷移	109

スレーブ

マスタ / スレーブ型通信機能	533
-----------------------	-----

せ

セクタ

複数のセクタを指定する場合の注意	574
------------------------	-----

セクタ構成

セクタ構成	556
-------------	-----

セクタ消去

書込みとチップ消去 / セクタ消去 の場合	566, 567
セクタ消去一時停止の場合	564
セクタ消去の場合	568
チップ消去 / セクタ消去の場合	564
フラッシュメモリのセクタ消去手順	574
フラッシュメモリのセクタ消去を 一時停止する	576
フラッシュメモリのセクタ消去を 再開する	577
フラッシュメモリの任意のデータを消去する (セクタ消去)	574
セクタ消去一時停止 セクタ消去一時停止の場合	566
セクタ消去タイマフラグ セクタ消去の場合	568
専用ポーレートジェネレータ 専用ポーレートジェネレータによる ポーレート	518
専用レジスタ 専用レジスタと汎用レジスタ	35
専用レジスタの構成	36

そ

送信割込み要求の発生	
送信割込み要求の発生とフラグセットの	
タイミング	514
双方向通信機能	
双方向通信機能	531
測定結果	
測定結果データ	307
測定モードと測定動作	
測定モードと測定動作	310
ソフトウェア	
ソフトウェアプルアップ抵抗	119
ソフトウェア割込みからの復帰	148
ソフトウェア割込み動作	149
ソフトウェア割込みの起動	148

た

タイマクリア	
タイマクリア	388
タイマ状態制御レジスタ	
タイマ状態制御レジスタ, 下位バイト	
(TCCSL)	357
タイマ状態制御レジスタ, 上位バイト	
(TCCSH)	354
タイマ制御ステータスレジスタ	
タイマ制御ステータスレジスタ下位	
(TMCSRL0/TMCSRL1)	264
タイマ制御ステータスレジスタ上位ビットおよび	
bit7(TMCSRH0/TMCSRH1)	262
タイマ値	
タイマ値とリロード値	304
タイマデータレジスタ	
タイマデータレジスタ (TCDT)	353
タイマとパルス幅	
タイマとパルス幅測定の起動 / 停止	302
タイマモード	
タイマモード	389
タイマモード動作のフローチャート	306
タイマ割込み	
タイマ割込み	391
タイミングリミット超過フラグ	
書き込みとチップ消去 / セクタ消去	
の場合	567
タイムベースタイマ	
インターバルタイマ機能	
(タイムベースタイマ)	241
タイムベースタイマ使用上の注意	243
タイムベースタイマの動作	244
タイムベースタイマの	
ブロックダイアグラム	236
タイムベースタイマの割込み	240
タイムベースタイマの割込みと	
EI ² OS の対応	240
タイムベースタイマ制御レジスタ	
タイムベースタイマ制御レジスタ	
(TBTC)	238
タイムベースタイマモード	
タイムベースタイマモードの解除	112, 121
タイムベースタイマモードへの遷移	112

ダイレクトページレジスタ	
ダイレクトページレジスタ (DPR)	49
多機能タイマ	
多機能タイマの EI ² OS 機能	386
多機能タイマの端子	344
多機能タイマの動作	387
多機能タイマのブロックダイアグラム	340
多機能タイマ端子	
多機能タイマ端子のブロックダイアグラム	345
多バイト長	
多バイト長のアクセス	34
多バイト長オペランド	
多バイト長オペランドの格納状態	33
多バイト長データ	
スタック上の多バイト長データの格納状態	34
メモリ上の多バイト長データの格納状態	33
単一測定モード	
単一測定モードと連続測定モード	307
端子	
端子のブロックダイアグラム	429, 482
端子機能	
I/O 端子と端子機能	13
端子配列図	
FPT-80P-M06 端子配列図	8
FPT-80P-M21/FPT-80P-M22 端子配列図	9
単発変換モード	
単発変換モード (ADCS: MD1, MD0=00 _B または	
01 _B)	466
単発変換モードの設定	467
単発変換モードの動作と使用方法	468

ち

遅延割込み発生モジュール	
遅延割込み発生モジュールの動作	420
遅延割込み発生モジュールの	
ブロックダイアグラム	418
遅延割込み発生モジュールレジスタ	
遅延割込み発生モジュールレジスタ	
(DIRR)	419
遅延割込み要求ラッチ	
遅延割込み要求ラッチの使用上の注意	421
チップ消去	
書き込みとチップ消去 / セクタ消去の場合	566, 567
チップ消去 / セクタ消去の場合	564
フラッシュメモリのデータを消去する	
(チップ消去)	573
注意	
8/10 ビット A/D コンバータ使用上の注意	477
注意	576
データ書き込み上の注意	571
複数のセクタを指定する場合の注意	574
注意事項	
スタンバイモード遷移のための低消費電力モード	
制御レジスタ (LPMCR) へアクセスする際の	
注意事項	121
直接アドレッシング	
直接アドレッシング	603

つ

通信プリスケアラ制御レジスタ	
通信プリスケアラ制御レジスタ (CDCR)	508

て

停止変換モード	
停止変換モード	
(ADCS: MD1, MD0=11 _B)	466
停止変換モードの設定	471
停止変換モードの動作と使用方法	471
低消費電力制御回路	
低消費電力制御回路の	
ブロックダイアグラム	102
低消費電力モード	
低消費電力モード制御レジスタへのアクセス	
.....	106
低消費電力モードの動作状態	117
低消費電力モード制御レジスタ	
低消費電力モード制御レジスタ (LPMCR)	104
低消費電力モード制御レジスタへのアクセス	
.....	106
ディスクリプタ	
拡張インテリジェント I/O サービス (EI ² OS)	
ディスクリプタ (ISD) の構成	152
通倍率	
PLL クロック通倍率の選択	83
データ	
フラッシュメモリの任意のデータを消去する	
(セクタ消去)	574
フラッシュメモリへデータを書き込む	571
データカウンタ	
データカウンタ (DCT)	153
データ書込み	
データ書込み上の注意	571
データボーリングフラグ	
書込みの場合	564
セクタ消去一時停止の場合	564
チップ消去 / セクタ消去の場合	564
デバイスの取り扱いについて	
デバイスの取り扱いについて	21
デフォルト空間	
バンク方式によるアドレス指定とデフォルト空間	
.....	32

と

動作クロック	
動作クロック供給	242
動作説明	
ROM 修正機能の動作説明	544
動作モード	
16 ビットリロードタイマの動作モード	256
動作モード	164
動作モードの選択	300
トグルビットフラグ	
書込みとチップ消去 / セクタ消去の場合	566
セクタ消去一時停止の場合	566

な

内部クロックモード	
内部クロックモード (リロードモード)	
の動作	271
内部クロックモード	
(ワンショットモード)	274
内部タイマ	
内部タイマ (16 ビットリロードタイマ) による	
ポーレート	521

に

入出力回路	
入出力回路の種類	17
入出力ポート	
入出力ポート機能	170

の

ノイズキャンセル機能	
DTTI 端子ノイズキャンセル機能	413
ノンオーバーラップ信号	
通常極性の PPG によるノンオーバーラップ	
信号生成	410
通常極性の RT1/RT3/RT5 によるノンオーバ	
ラップ信号生成	408
反転極性の PPG によるノンオーバーラップ	
信号生成	411
反転極性の RT1/RT3/RT5 によるノンオーバ	
ラップ信号生成	409

は

ハードウェア	
ハードウェアシーケンスフラグ	562
ハードウェアの割込みの抑止	138
ハードウェア割込み	137
ハードウェア割込みからの復帰	140
ハードウェア割込み使用手順	143
ハードウェア割込みの起動	140
ハードウェア割込みの構造	138
ハードウェア割込みの処理時間	146
ハードウェア割込みの動作	141
波形ジェネレータ	
波形ジェネレータ	339
波形ジェネレータ使用上の注意	415
波形ジェネレータの	
ブロックダイアグラム	343
波形ジェネレータのレジスタ	350
波形ジェネレータ割込み	385
波形ジェネレータ割込みと EI ² OS	385
波形制御レジスタ	
波形制御レジスタ (SIGCR)	380
バスモード	
バスモード設定ビット	166
発振安定待ち	
発振安定待ちリセット状態	65
発振安定待ち時間	
発振安定待ち時間	85, 121
発振安定待ち時間用タイマの機能	241

MB90820B シリーズ

リセット要因と発振安定待ち時間	64
発振クロック	
発振クロック周波数とシリアルクロック	
入力周波数	590
バッファアドレスポインタ	
バッファアドレスポインタ (BAP)	155
パルス幅	
最小入力パルス幅	308
パルス幅 / 周期測定範囲	308
パルス幅 / 周期の算出	308
パルス幅測定機能	298
パルス幅測定動作のフローチャート	312
バンクセレクトプリフィックス	
バンクセレクトプリフィックス	
(PCB, DTB, ADB, SPB)	54
バンク方式	
バンク方式によるアドレス指定とデフォルト空間	
.....	32
リニア方式とバンク方式	29
バンクレジスタ	
バンクレジスタ	
(PCB, DTB, USB, SSB, ADB)	50
バンクレジスタとアクセス空間	31
汎用レジスタ	
32 ビット汎用レジスタの下位 24 ビットアドレス	
引用指定 (間接指定)	30
専用レジスタと汎用レジスタ	35
汎用レジスタの構成	51
汎用レジスタ領域	
汎用レジスタ領域とレジスタバンクポインタ	
(RP)	46

ひ

非同期モード	
非同期モードでの動作	526

ふ

フラグセットのタイミング	
受信割込み要求の発生とフラグセットの	
タイミング	512
送信割込み要求の発生とフラグセットの	
タイミング	514
フラグ変化抑止プリフィックス	
フラグ変化抑止プリフィックス (NCC)	57
フラッシュメモリ	
512K/1024K ビットフラッシュメモリの	
概要	554
512K/1024K ビットフラッシュメモリの	
特長	554
フラッシュメモリ書込み / 消去	554
フラッシュメモリ書込み / 消去の詳細説明	
.....	569
フラッシュメモリ書込み手順	571
フラッシュメモリのセクタ消去手順	574
フラッシュメモリのセクタ消去を	
一時停止する	576
フラッシュメモリのセクタ消去を	
再開する	577
フラッシュメモリのデータを消去する	
(チップ消去)	573

フラッシュメモリの任意のデータを消去する	
(セクタ消去)	574
フラッシュメモリのレジスタ	555
フラッシュメモリヘデータを書き込む	571
フラッシュメモリを読み出し / リセット状態	
にする	570
フラッシュメモリ制御ステータスレジスタ	
フラッシュメモリ制御ステータスレジスタ	
(FMCS)	558
プリフィックスコード	
プリフィックスコード	53
プリフィックスコードと割込み抑止命令	58
プリフィックスコードの連続	59
フローチャート	
タイマモード動作のフローチャート	306
パルス幅測定動作のフローチャート	312
プログラムアドレス検出制御ステータスレジスタ	
プログラムアドレス検出制御ステータスレジスタ	
(PACSR)	542
プログラムアドレス検出レジスタ (× 2)	
プログラムアドレス検出レジスタ (× 2)	538
プログラムアドレス検出レジスタ 0/1	
プログラムアドレス検出レジスタ 0/1	
(PADR0/PADR1)	541
プログラムエラー	
プログラムエラーが発生した場合	546
プログラムカウンタ	
プログラムカウンタ (PC)	48
プロセッサステータス	
プロセッサステータスの構成	43
ブロックダイヤグラム	
クロックスーパーバイザのブロック	
ダイヤグラム	89
16 ビット PPG タイマ端子の	
ブロックダイヤグラム	318
16 ビット PPG タイマの	
ブロックダイヤグラム	317
16 ビットアウトプットコンペアの	
ブロックダイヤグラム	342
16 ビットインプットキャプチャの	
ブロックダイヤグラム	342
16 ビットフリーランタイマの	
ブロックダイヤグラム	341
16 ビットリロードタイマの	
ブロックダイヤグラム	258
8/10 ビット A/D コンバータの	
ブロックダイヤグラム	446
D/A コンバータのブロックダイヤグラム	481
DTP/ 外部割込みのブロックダイヤグラム	
.....	426
MB90820B シリーズのブロック	
ダイヤグラム	7
PWC タイマ端子の	
ブロックダイヤグラム	284
PWC タイマのブロックダイヤグラム	283
ROM 修正機能のブロックダイヤグラム	539
ROM ミラー機能選択モジュール	
ブロックダイヤグラム	550
UART の端子のブロックダイヤグラム	496
UART のブロックダイヤグラム	492
ウォッチドッグタイマの	
ブロックダイヤグラム	248

外部リセット端子のブロック ダイアグラム	66
クロック発生部のブロックダイアグラム	76
タイムベースタイマの ブロックダイアグラム	236
多機能タイマ端子の ブロックダイアグラム	345
多機能タイマのブロックダイアグラム	340
端子のブロックダイアグラム	429, 482
遅延割込み発生モジュールの ブロックダイアグラム	418
低消費電力制御回路の ブロックダイアグラム	102
波形ジェネレータのブロックダイアグラム	343
ポート 0 の端子のブロックダイアグラム	174
ポート 1 の端子のブロックダイアグラム	181
ポート 2 の端子のブロックダイアグラム	188
ポート 3 の端子のブロックダイアグラム	195
ポート 4 の端子のブロックダイアグラム	202
ポート 5 の端子のブロックダイアグラム	208
ポート 6 の端子のブロックダイアグラム	214
ポート 7 の端子のブロックダイアグラム	220
ポート 8 の端子のブロックダイアグラム	228
分周比制御レジスタ 分周比制御レジスタ (DIV0/DIV1)	294

へ

変換動作 EI ² OS 機能を使用した変換動作	473
--	-----

ほ

ポート 0 ポート 0 の構成	173
ポート 0 の端子のブロックダイアグラム	174
ポート 0 の端子配列	173
ポート 0 の動作	178
ポート 0 のレジスタ	175
ポート 0 のレジスタの機能	176
ポート 1 ポート 1 の構成	180
ポート 1 の端子	180
ポート 1 の端子のブロックダイアグラム	181
ポート 1 の動作	185
ポート 1 のレジスタ	182
ポート 1 のレジスタの機能	183
ポート 2 ポート 2 の構成	187
ポート 2 の端子	187
ポート 2 の端子のブロックダイアグラム	188
ポート 2 の動作	192
ポート 2 のレジスタ	189
ポート 2 のレジスタの機能	190
ポート 3 ポート 3 の構成	194
ポート 3 の端子のブロックダイアグラム	195
ポート 3 の端子配列	194
ポート 3 の動作	199
ポート 3 のレジスタ	196
ポート 3 のレジスタの機能	197

ポート 4 ポート 4 の構成	201
ポート 4 の端子	201
ポート 4 の端子のブロックダイアグラム	202
ポート 4 の動作	205
ポート 4 のレジスタ	203
ポート 4 のレジスタの機能	204
ポート 5 ポート 5 の構成	207
ポート 5 の端子	207
ポート 5 の端子のブロックダイアグラム	208
ポート 5 の動作	211
ポート 5 のレジスタ	209
ポート 5 のレジスタの機能	210
ポート 6 ポート 6 の構成	213
ポート 6 の端子	213
ポート 6 の端子のブロックダイアグラム	214
ポート 6 の動作	217
ポート 6 のレジスタ	214
ポート 6 のレジスタの機能	215
ポート 7 ポート 7 の構成	219
ポート 7 の端子	219
ポート 7 の端子のブロックダイアグラム	220
ポート 7 の動作	225
ポート 7 のレジスタ	222
ポート 7 のレジスタの機能	223
ポート 8 ポート 8 の構成	227
ポート 8 の端子	227
ポート 8 の端子のブロックダイアグラム	228
ポート 8 の動作	231
ポート 8 のレジスタ	229
ポート 8 のレジスタの機能	230
ボーレート 外部クロックによるボーレート	523
専用ボーレートジェネレータによる ボーレート	518
内部タイマ (16 ビットリロードタイマ) による ボーレート	521

ま

マシクロック マシクロックの選択	83
マスク 割込みマスク機能	392
マスタ マスタ / スレーブ型通信機能	533

め

命令 命令の種類	600
命令マップの構造	638
命令一覧表 F ² MC-16LX 命令一覧表	624
メインクロックモード メインクロックモード, PLL クロックモード	83

メモリ空間	
メモリ空間	25
メモリマップ	
E ² PROM メモリマップ	545
メモリマップ	27

も

モード設定	
モード設定	164
モード端子	
CPU モード端子 (MD2 ~ MD0)	165
モード端子	67
モード端子とモードデータの関係	167
モードデータ	
モード端子とモードデータの関係	167
モードデータ	166
モードデータの取り込み	68
モードデータ読出し後の端子の状態	71

ゆ

ユーザスタックポインタ	
ユーザスタックポインタ (USP)	42

よ

要求レベル設定レジスタ	
要求レベル設定レジスタ (ELVR)	434
読出し / リセット状態	
フラッシュメモリを読出し / リセット状態 にする	570

り

リセット	
発振安定待ちリセット状態	65
リセット動作中の端子の状態	71
リセット動作の概要	67
リセット要因	62
リセット要因と発振安定待ち時間	64
リセット要因ビット	69
リセットシーケンス	
リセットシーケンス	546
リセット要因	
リセット要因フラグビットとリセット要因 の対応	70
リセット要因ビット	
リセット要因ビットの注意事項	70
リセット要因フラグビット	
リセット要因フラグビットとリセット要因 の対応	70
リニア方式	
24 ビットオペランドによるリニア方式アドレス 指定	30
リニア方式とバンク方式	29
リロード値	
タイマ値とリロード値	304
リロード動作	
リロード動作モード	304

リロードモード	
内部クロックモード (リロードモード) の動作	271

れ

例外処理	
例外処理	160
レジスタ	
16 ビット PPG タイマのレジスタ	320
16 ビットアウトプットコンペアの レジスタ	348
16 ビットリロードタイマのレジスタ	261
8/10 ビット A/D コンバータのレジスタと 初期値の一覧	450
A/D 制御ステータスレジスタ下位 (ADCS0)	455
A/D 制御ステータスレジスタ上位 (ADCS1)	451
A/D セットアップレジスタ (ADSR0/ADSR1)	458
A/D デ - タレジスタ (ADCR0/ADCR1)	457
D/A コンバータレジスタ 0(DAT0)	485
D/A コンバータレジスタ 1(DAT1)	484
D/A 制御レジスタ 0(DACR0)	487
D/A 制御レジスタ 1(DACR1)	486
DTP/ 外部割込み許可レジスタ (ENIR)	432
DTP/ 外部割込み要因レジスタ (EIRR)	431
PPG 出力制御 / インプットキャプチャ状態制御 レジスタ, 上位バイト (PICS01)	371
PPG 状態制御レジスタ, 下位 (PCNTL1 ~ PCNTL3)	327
PPG 状態制御レジスタ, 上位 (PCNTH0 ~ PCNTH2)	325
PPG ダウンカウンタレジスタ (PDCR0 ~ PDCR2)	322
PWC 状態制御レジスタ, 下位バイト (PWCSL0/PWCSL1)	291
ROM ミラー機能選択モジュールの レジスタ	550
アナログ入力許可レジスタ (ADER0/ADER1)	463
インプットキャプチャ状態制御レジスタ, 下位バイト (PICSL01)	373
インプットキャプチャのレジスタ	349
ウォッチドッグタイマ制御レジスタ (WDTC)	249
クロック選択レジスタ	78
シリアル出力データレジスタ (SODR0/SODR1)	507
シリアルステータスレジスタ (SSR0/SSR1)	503
シリアル制御レジスタ (SCR0/SCR1)	499
シリアル入力データレジスタ (SIDR0/SIDR1)	506
シリアルモードレジスタ (SMR0/SMR1)	501
タイマ状態制御レジスタ, 下位バイト (TCCSL)	357
タイマ状態制御レジスタ, 上位バイト (TCCSH)	354
タイマ制御ステータスレジスタ上位ビットおよび bit7(TMCSRH0/TMCSRH1)	262

プログラムアドレス検出レジスタ (× 2)	538
要求レベル設定レジスタ (ELVR)	434
割込み要因と割込みベクタ・割込み制御 レジスタ	128
レジスタ一覧	
I/O ポートのレジスタ一覧	172
レジスタバンク	
レジスタバンク	52
レジスタバンクポインタ	
汎用レジスタ領域とレジスタバンクポインタ (RP)	46
レジスタバンクポインタ (RP)	46
連続測定モード	
単一測定モードと連続測定モード	307
連続変換モード	
連続変換モード (ADCS: MD1,MD0=10 _B)	466
連続変換モードの設定	469
連続変換モードの動作と使用方法	469

わ

割込み

16 ビット PPG タイマの割込み	329
16 ビットアウトプットコンペア割込み	383
16 ビットアウトプットコンペア割込みと EI ² OS	383
16 ビットインプットキャプチャ割込み	384
16 ビットインプットキャプチャ割込みと EI ² OS	384
16 ビットフリーランタイム割込み	382
16 ビットフリーランタイム割込みと EI ² OS	382
16 ビットリロードタイマの割込み	268
8/10 ビット A/D コンバータの割込みと EI ² OS	465
A/D コンバータの割込み	465
DTTI 割込み	413
PWC タイマの割込み	295
PWC タイマ割込みと EI ² OS	295
UART の割込みと EI ² OS	511
UART 割込みおよび EI ² OS	491
スタンバイモードへの移行と割込み	120
ソフトウェア割込みからの復帰	148
ソフトウェア割込み動作	149
ソフトウェア割込みの起動	148
タイマ割込み	391
タイムベースタイマの割込み	240
タイムベースタイマの割込みと EI ² OS の対応	240

ハードウェアの割込みの抑止	138
ハードウェア割込み	137
ハードウェア割込みからの復帰	140
ハードウェア割込み使用手順	143
ハードウェア割込みの起動	140
ハードウェア割込みの構造	138
ハードウェア割込みの処理時間	146
ハードウェア割込みの動作	141
波形ジェネレータ割込み	385
波形ジェネレータ割込みと EI ² OS	385
複数の割込み	144
割込み処理開始時のスタック動作	161
割込み処理からの復帰時のスタック動作	161
割込み動作	125
割込み動作時の処理	142
割込みによるスタンバイモードの解除	120
割込みの種類と機能	124
割込みベクタ	126
割込みマスク機能	392
割込み要因と割込みベクタ・割込み制御 レジスタ	128
割込み制御レジスタ	
割込み制御レジスタ (ICR) のビット構成	134
割込み制御レジスタ (ICR00 ~ ICR15)	132
割込み制御レジスタ一覧	130
割込み制御レジスタ機能	130
割込み制御レジスタの機能	135
割込み要因と割込みベクタ・割込み制御 レジスタ	128
割込みベクタ	
割込み要因と割込みベクタ・割込み制御 レジスタ	128
割込み要因	
割込み要因と割込みベクタ・割込み制御 レジスタ	128
割込み要求	
割込み要求の生成	305, 309
割込み抑止命令	
プリフィックスコードと割込み抑止命令	58
割込みレベルマスクレジスタ	
割込みレベルマスクレジスタ (ILM)	47
ワンショット動作	
ワンショット動作モード	304
ワンショットモード	
内部クロックモード (ワンショットモード)	274
ワンショットモード (PCNTL: MDSE = 1)	332

CM44-10147-2

富士通マイクロエレクトロニクス・CONTROLLER MANUAL

F²MC-16LX

16 ビット・マイクロコントローラ

MB90820B Series ハードウェアマニュアル

2008 年 7 月 第 2 版発行

発行	富士通マイクロエレクトロニクス株式会社
編集	マーケティング統括部 ビジネス推進部
