



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

MB90330A 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

MB90330A 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社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、「デザインレビューシート」を参照してください。
「デザインレビューシート」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://edevicet.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顧を賜り厚くお礼申し上げます。

MB90330A シリーズは、パソコン周辺機器などの USB 通信が要求される用途向けに設計された 16 ビットマイクロコントローラです。USB 機能は、12 Mbps の Function 動作ばかりでなく、簡易的な Host 動作が可能です。ディスプレイやオーディオなどのパソコン周辺機器や USB 通信対応の携帯機器制御などに適した機能を搭載しています。

本書は、実際に MB90330A シリーズを使用して製品を開発される技術者を対象に、MB90330A シリーズの機能や動作について解説したものです。本書をご一読ください。なお、各種命令の詳細については、『インストラクション・マニュアル』をご参照ください。

■ 商標

F²MC は、FUJITSU Flexible Microcontroller の略で富士通マイクロエレクトロニクス株式会社の登録商標です。

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。

■ サンプルプログラム

F²MC-16LX にファミリの周辺機能を動作させるためのサンプルプログラムを無償で提供しております。当社マイコンの動作仕様や使用方法の確認などにお役立てください。

マイコンサポート情報

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

* サンプルプログラムについては、予告なしに変更することがあります。本ソフトは、標準的な動作や使い方を示したものですので、お客様のシステム上でご使用の際には十分評価の上でご使用ください。また、これらの使用に起因し生じた損害については、当社は一切その責任を負いません。

■ 本書の全体構成

このマニュアルは、以下に示す 27 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、MB90330A シリーズの特長、ブロックダイアグラム、機能概要など全体を知るための基本的なことがらについて説明します。

第 2 章 CPU

この章では、MB90330A シリーズの機能を知るために、仕様、メモリおよびレジスタの機能について説明します。

第 3 章 割込み

この章では、MB90330A シリーズの割込みと拡張インテリジェント I/O サービス (EI²OS)、ダイレクトメモリアクセスコントローラ (μDMAC) について説明します。

第 4 章 リセット

この章では、MB90330A シリーズのリセットについて説明します。

第 5 章 クロック

この章では、MB90330A シリーズのクロック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、MB90330A シリーズの低消費電力モードについて説明します。

第 7 章 モード設定

この章では、モード設定と外部メモリアクセスについて説明します。

第 8 章 I/O ポート

この章では、I/O ポートで使用するレジスタの構成および機能について説明します。

第 9 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 10 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 11 章 時計タイマ

この章では、時計タイマの概要、レジスタの構成 / 機能および時計タイマの動作について説明します。

第 12 章 16 ビット入出力タイマ

この章では、16 ビット入出力タイマの概要、レジスタの構成 / 機能および 16 ビット入出力タイマの動作について説明します。

第 13 章 USB ファンクション

この章では、USB ファンクションの機能と概要について説明します。

第 14 章 USB ホスト

この章では、USB ホストの機能と動作について示します。

第 15 章 PWC タイマ

この章では、PWC タイマの概要、レジスタの構成 / 機能、PWC タイマの動作および PWC タイマの使用上の注意について説明します。

第 16 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概要、レジスタの構成 / 機能および 16 ビットリロードタイマの動作について説明します。

第 17 章 8/16 ビット PPG タイマ

この章では、8/16 ビット PPG タイマの概要、レジスタの構成 / 機能および 8/16 ビット PPG タイマの動作について説明します。

第 18 章 DTP/ 外部割込み

この章では、DTP/ 外部割込みの概要、レジスタの構成 / 機能および DTP/ 外部割込みの動作について説明します。

第 19 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。

第 20 章 拡張 I/O シリアルインタフェース

この章では、拡張 I/O シリアルインタフェースの概要、レジスタの構成 / 機能および拡張 I/O シリアルインタフェースの動作について説明します。

第 21 章 UART

この章では、UART の機能と動作について説明します。

第 22 章 I²C インタフェース

この章では、I²C インタフェースの概要、レジスタの構成 / 機能および I²C インタフェースの動作について説明します。

第 23 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について説明します。

第 24 章 アドレス一致検出機能

この章では、アドレス一致検出機能の機能と動作について説明します。

第 25 章 フラッシュメモリ

この章では、3M/4M ビットフラッシュメモリの機能や動作について説明します。

第 26 章 シリアル書込み接続例（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ム）

この章では、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ムを用いた場合の、シリアル書込み接続例について説明します。

第 27 章 シリアル書込み接続例（富士通マイクロエレクトロニクス製 シリアルプログラム）

本章では、フラッシュメモリのシリアル書込みについて、富士通マイクロエレクトロニクス製シリアルプログラムを用いた場合の基本構成を説明します。

付録

付録では、メモリマップおよび F²MC-16LX に使用している命令について、説明します。

- 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に、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	MB90330A シリーズの特長	2
1.2	ブロックダイアグラム	6
1.3	外形寸法図	8
1.4	端子配列図	10
1.5	端子機能	11
1.6	入出力回路形式	18
1.7	デバイス使用上の注意	21
第2章	CPU	25
2.1	CPU の概要	26
2.2	メモリ空間	27
2.3	リニア方式によるアドレス指定	30
2.4	バンク方式によるアドレス指定	31
2.5	メモリ空間における多バイト長データ	33
2.6	レジスタ	34
2.6.1	アキュムレータ (A)	37
2.6.2	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)	38
2.6.3	プロセッサステータス (PS)	39
2.6.4	プログラムカウンタ (PC)	42
2.6.5	バンクレジスタ (PCB, DTB, USB, SSB, ADB)	43
2.6.6	ダイレクトページレジスタ (DPR)	44
2.7	レジスタバンク	45
2.8	プリフィックスコード	46
2.9	割込み抑止命令	49
第3章	割込み	51
3.1	割込みの概要	52
3.2	割込み要因と割込みベクタ	55
3.3	割込み制御レジスタと周辺機能	59
3.3.1	割込み制御レジスタ (ICR00 ~ ICR15)	61
3.3.2	割込み制御レジスタの機能	63
3.4	ハードウェア割込み	66
3.4.1	ハードウェア割込みの動作	69
3.4.2	ハードウェア割込みの動作フロー	71
3.4.3	ハードウェア割込みの使用手順	72
3.4.4	多重割込み	74
3.4.5	ハードウェア割込みの処理時間	76
3.5	ソフトウェア割込み	78
3.6	拡張インテリジェント I/O サービス (EI ² OS) による割込み	80
3.6.1	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)	82
3.6.2	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) の各レジスタ	83
3.6.3	拡張インテリジェント I/O サービス (EI ² OS) の動作	86
3.6.4	拡張インテリジェント I/O サービス (EI ² OS) の使用手順	87
3.6.5	拡張インテリジェント I/O サービス (EI ² OS) 処理時間	88

3.7	例外処理割込み	91
3.8	μDMAC による割込み.....	92
3.8.1	μDMAC の機能	93
3.8.2	μDMAC のレジスタ	94
3.8.2.1	DMA ディスクリプタチャネル指定レジスタ (DCSR)	95
3.8.2.2	DMA ステータスレジスタ (DSRH/DSRL)	97
3.8.2.3	DMA ストップステータスレジスタ (DSSR)	98
3.8.2.4	DMA 許可レジスタ (DERH/DERL).....	100
3.8.3	DMA ディスクリプタウィンドウレジスタ (DDWR).....	101
3.8.3.1	DMA データカウンタ (DDCTH/DDCTL)	102
3.8.3.2	DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL)	103
3.8.3.3	DMA コントロールレジスタ (DMACS)	104
3.8.3.4	DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)	106
3.8.4	μDMAC の動作説明.....	107
3.9	例外	109
3.10	割込み処理のスタック動作.....	110
3.11	割込み処理のプログラム例.....	112
3.12	遅延割込み発生モジュール.....	116
3.12.1	遅延割込み発生モジュールの動作.....	117
第 4 章	リセット	119
4.1	リセットの概要	120
4.2	リセット要因と発振安定待ち時間.....	122
4.3	外部リセット端子	124
4.4	リセット動作	125
4.5	リセット要因ビット	127
4.6	リセットによる各端子の状態.....	129
第 5 章	クロック	131
5.1	クロックの概要	132
5.2	クロック発生部のブロックダイアグラム	135
5.3	クロック選択レジスタ (CKSCR).....	137
5.4	クロックモード	140
5.5	発振安定待ち時間.....	144
5.6	振動子と外部クロックの接続.....	145
第 6 章	低消費電力モード	147
6.1	低消費電力モードの概要	148
6.2	低消費電力制御回路のブロックダイアグラム	151
6.3	低消費電力モード制御レジスタ (LPMCR)	153
6.4	CPU 間欠動作モード	156
6.5	スタンバイモード	157
6.5.1	スリープモード	158
6.5.2	タイムベースタイマモード	160
6.5.3	時計モード	162
6.5.4	ストップモード	164
6.6	状態遷移図.....	166
6.7	スタンバイモード, ホールド, リセット時の端子状態.....	168
6.8	低消費電力モード使用上の注意	177

第7章	モード設定	181
7.1	モード設定	182
7.2	モード端子 (MD2 ~ MD0)	183
7.3	モードデータ	184
7.4	外部メモリアクセス	188
7.4.1	自動レディ機能選択レジスタ (ARSR)	190
7.4.2	外部アドレス出力制御レジスタ (HACR)	191
7.4.3	バス制御信号選択レジスタ (EPCR)	192
7.5	モード設定の各モードにおける動作	194
7.5.1	外部メモリアクセス制御信号	195
7.5.2	レディ機能	198
7.5.3	ホールド機能	201
第8章	I/O ポート	205
8.1	I/O ポートの機能	206
8.2	I/O ポートのレジスタ	207
8.2.1	ポートデータレジスタ (PDR0 ~ PDRB)	208
8.2.2	ポート方向レジスタ (DDR0 ~ DDRB)	210
8.2.3	その他のレジスタ	212
第9章	タイムベースタイマ	215
9.1	タイムベースタイマの概要	216
9.2	タイムベースタイマの構成	218
9.3	タイムベースタイマ制御レジスタ (TBTC)	220
9.4	タイムベースタイマの割込み	222
9.5	タイムベースタイマの動作	223
9.6	タイムベースタイマ使用上の注意	225
9.7	タイムベースタイマのプログラム例	227
第10章	ウォッチドッグタイマ	229
10.1	ウォッチドッグタイマの概要	230
10.2	ウォッチドッグタイマ制御レジスタ (WDTC)	232
10.3	ウォッチドッグタイマの構成	234
10.4	ウォッチドッグタイマの動作	236
10.5	ウォッチドッグタイマ使用上の注意	238
10.6	ウォッチドッグタイマのプログラム例	239
第11章	時計タイマ	241
11.1	時計タイマの概要	242
11.2	時計タイマの構成	243
11.3	時計タイマ制御レジスタ (WTC)	244
11.4	時計タイマの動作	246
第12章	16 ビット入出力タイマ	249
12.1	16 ビット入出力タイマの概要	250
12.2	16 ビット入出力タイマのレジスタ	252
12.2.1	16 ビットフリーランタイマ	253
12.2.2	アウトプットコンペア	259
12.2.3	インプットキャプチャ	264

12.3	16 ビット入出力タイマの動作.....	268
12.3.1	16 ビットフリーランタイマの動作	269
12.3.2	16 ビットアウトプットコンペアの動作	270
12.3.3	16 ビットインプットキャプチャの動作	272
12.3.4	16 ビットフリーランタイマのタイミング	273
12.3.5	アウトプットコンペアのタイミング	274
12.3.6	インプットキャプチャの入力タイミング.....	275
第 13 章	USB ファンクション.....	277
13.1	USB ファンクションの概要	278
13.2	USB ファンクションのブロックダイアグラム.....	279
13.3	USB ファンクションのレジスタ	280
13.3.1	UDC 制御レジスタ (UDCC).....	284
13.3.2	EP0 制御レジスタ (EP0C).....	287
13.3.3	EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C).....	289
13.3.4	タイムスタンプレジスタ (TMSP).....	293
13.3.5	UDC ステータスレジスタ (UDCS).....	294
13.3.6	UDC 割込み許可レジスタ (UDCIE).....	297
13.3.7	EP0I ステータスレジスタ (EP0IS).....	299
13.3.8	EP0O ステータスレジスタ (EP0OS)	301
13.3.9	EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S)	304
13.3.10	EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT).....	308
13.4	USB ファンクションの動作説明	309
13.4.1	接続検出と切断検出	312
13.4.2	コマンド応答時の各レジスタ動作.....	314
13.4.3	STALL 応答と解除.....	316
13.4.4	サスペンド機能	321
13.4.5	ウェイクアップ機能	322
13.4.6	DMA 転送機能	323
13.4.7	NULL 転送機能	327
第 14 章	USB ホスト.....	329
14.1	USB ホストの特長.....	330
14.2	USB ホストの制限事項.....	331
14.3	USB ホストのブロックダイアグラム	332
14.4	USB ホストのレジスタ	334
14.4.1	ホストコントロールレジスタ 0, 1(HCNT0/HCNT1)	337
14.4.2	ホスト割込みレジスタ (HIRQ)	341
14.4.3	ホストエラーステータスレジスタ (HERR)	344
14.4.4	ホスト状態ステータスレジスタ (HSTATE).....	346
14.4.5	SOF 割込み FRAME 比較レジスタ (HFCOMP).....	348
14.4.6	リトライタイマ設定レジスタ (HRTIMER)	349
14.4.7	ホストアドレスレジスタ (HADR)	350
14.4.8	EOF 設定レジスタ (HEOF)	351
14.4.9	FRAME 設定レジスタ (HFRAME)	352
14.4.10	ホストトークンエンドポイントレジスタ (HTOKEN)	353
14.5	USB ホストの動作	355
14.5.1	デバイスの接続	356
14.5.2	USB バスのリセット	358

14.5.3	トークンパケット	359
14.5.4	データパケット	362
14.5.5	ハンドシェークパケット	363
14.5.6	リトライ機能	364
14.5.7	SOF 割込み	365
14.5.8	エラーステータス	367
14.5.9	パケット終了	368
14.5.10	サスペンド・レジューム	369
14.5.11	デバイスの切断	372
14.6	USB ホストの各トークンフローチャート	373
第 15 章 PWC タイマ		375
15.1	PWC タイマの概要	376
15.2	PWC タイマのレジスタ	378
15.2.1	PWC コントロールステータスレジスタ (PWCSR)	379
15.2.2	PWC データバッファレジスタ (PWCR)	384
15.2.3	PWC 分周比制御レジスタ (DIVR)	385
15.3	PWC タイマの動作	386
15.3.1	タイマ機能の動作	387
15.3.2	パルス幅測定機能の動作	388
15.3.3	カウントクロックおよび動作モードの選択	389
15.3.4	タイマ / パルス幅測定の起動と停止	391
15.3.5	タイマモードの動作	393
15.3.6	パルス幅測定モードの動作	396
15.4	PWC タイマの使用上の注意	401
第 16 章 16 ビットリロードタイマ		403
16.1	16 ビットリロードタイマの概要	404
16.1.1	16 ビットリロードタイマの機能	405
16.1.2	16 ビットリロードタイマのブロックダイアグラム	407
16.2	16 ビットリロードタイマのレジスタ	408
16.2.1	タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2)	409
16.2.2	16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/ 16 ビットリロードレジスタ 0 ~ 2(TMRLR0 ~ TMRLR2)	413
16.3	16 ビットリロードタイマの動作	415
16.3.1	カウンタ動作の状態遷移	416
16.3.2	内部クロックモード (リロードモード) の動作	417
16.3.3	内部クロックモード (ワンショットモード) の動作	419
16.3.4	イベントカウントモード	421
第 17 章 8/16 ビット PPG タイマ		423
17.1	8/16 ビット PPG タイマの概要	424
17.1.1	8/16 ビット PPG タイマのブロックダイアグラム	425
17.2	8/16 ビット PPG タイマのレジスタ	427
17.2.1	PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4)	428
17.2.2	PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5)	430
17.2.3	PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45)	433
17.2.4	PPG リロードレジスタ (PRLL0 ~ PRLL5, PRLH0 ~ PRLH5)	435
17.3	8/16 ビット PPG タイマの動作	436

第 18 章	DTP/ 外部割込み	441
18.1	DTP/ 外部割込みの概要	442
18.2	DTP/ 外部割込みのレジスタ	443
18.3	DTP/ 外部割込みの動作	446
18.4	DTP/ 外部割込み使用上の注意	448
第 19 章	8/10 ビット A/D コンバータ	451
19.1	8/10 ビット A/D コンバータの概要	452
19.2	8/10 ビット A/D コンバータの構成	453
19.3	8/10 ビット A/D コンバータのレジスタ	455
19.3.1	A/D 制御ステータスレジスタ上位 (ADCS1)	456
19.3.2	A/D 制御ステータスレジスタ下位 (ADCS0)	459
19.3.3	A/D 変換チャネル設定レジスタ (ADMR)	461
19.3.4	A/D データレジスタ (ADCR1/ADCR0)	463
19.4	8/10 ビット A/D コンバータの動作説明	465
19.4.1	μDMAC または EI ² OS を使用した変換動作	468
19.4.2	A/D 変換データ保護機能	469
19.5	8/10 ビット A/D コンバータ使用上の注意	471
19.6	8/10 ビット A/D コンバータのプログラム例 -1(単発モードの EI ² OS 起動例)	472
19.7	8/10 ビット A/D コンバータのプログラム例 -2(連続モードの EI ² OS 起動例)	475
19.8	8/10 ビット A/D コンバータのプログラム例 -3(停止モードの EI ² OS 起動例)	478
第 20 章	拡張 I/O シリアルインタフェース	481
20.1	拡張 I/O シリアルインタフェースの概要	482
20.2	拡張 I/O シリアルインタフェースのレジスタ	483
20.2.1	シリアルモードコントロールステータスレジスタ (SMCS)	484
20.2.2	シリアルデータレジスタ (SDR)	489
20.2.3	通信プリスケール制御レジスタ (SDCR)	490
20.3	拡張 I/O シリアルインタフェースの動作	491
20.3.1	シフトクロックモード	492
20.3.2	シリアル I/O の動作状態	493
20.3.3	シフト動作のスタート / ストップタイミングと入出力のタイミング	495
20.3.4	割込み機能	497
第 21 章	UART	499
21.1	UART の概要	500
21.2	UART のブロックダイアグラム	502
21.3	UART の端子	505
21.4	UART のレジスタ	506
21.4.1	シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3)	507
21.4.2	シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3)	509
21.4.3	シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3)	511
21.4.4	シリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3), シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3)	514
21.4.5	UART プリスケール制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3), UART プリスケールリロードレジスタ 0 ~ 3(UTRLR0 ~ UTRLR3)	516
21.5	UART の割込み	518
21.5.1	受信割込み発生とフラグセットのタイミング	520
21.5.2	送信割込み発生とフラグセットのタイミング	522

21.6	UART のボーレート	524
21.6.1	UART 内部クロックの専用ボーレートジェネレータによるボーレート	525
21.6.2	外部クロックの専用ボーレートジェネレータによるボーレート	526
21.6.3	外部クロック (1 対 1 モード) によるボーレート	527
21.7	UART の動作説明	528
21.7.1	非同期モード (動作モード 0, 1) 時の動作	530
21.7.2	同期モード (動作モード 2) 時の動作	533
21.7.3	双方向通信機能 (ノーマルモード)	536
21.7.4	マスタ/スレーブ型通信機能 (マルチプロセッサモード)	538
21.8	UART 使用上の注意	541
21.9	UART のプログラム例	542
第 22 章	I²C インタフェース	545
22.1	I ² C インタフェースの概要	546
22.2	I ² C インタフェースのレジスタ	548
22.2.1	I ² C バスステータスレジスタ 0 ~ 2 (IBSR0 ~ IBSR2)	549
22.2.2	I ² C バスコントロールレジスタ 0 ~ 2 (IBCR0 ~ IBCR2)	551
22.2.3	I ² C バスクロックコントロールレジスタ 0 ~ 2 (ICCR0 ~ ICCR2)	557
22.2.4	I ² C バスアドレスレジスタ 0 ~ 2 (IADR0 ~ IADR2)	559
22.2.5	I ² C バスデータレジスタ 0 ~ 2 (IDAR0 ~ IDAR2)	560
22.3	I ² C インタフェースの動作	561
22.3.1	I ² C インタフェースの転送フロー	564
22.3.2	I ² C インタフェースのモードフロー	566
22.3.3	I ² C インタフェースの動作フロー	567
第 23 章	ROM ミラー機能選択モジュール	569
23.1	ROM ミラー機能選択モジュールの概要	570
23.2	ROM ミラー機能選択レジスタ (ROMM)	571
第 24 章	アドレス一致検出機能	573
24.1	アドレス一致検出機能の概要	574
24.2	アドレス一致検出機能のブロックダイアグラム	575
24.3	アドレス一致検出機能のレジスタ構成	576
24.3.1	プログラムアドレス検出制御ステータスレジスタ (PACSR)	577
24.3.2	プログラムアドレス検出レジスタ (PADR0, PADR1)	579
24.4	アドレス一致検出機能の動作説明	581
24.4.1	アドレス一致検出機能の使用例	582
24.5	アドレス一致検出機能のプログラム例	586
第 25 章	フラッシュメモリ	589
25.1	フラッシュメモリの概要	590
25.2	フラッシュメモリのセクタ構成	591
25.3	フラッシュメモリコントロールステータスレジスタ (FMCS)	593
25.4	フラッシュメモリの自動アルゴリズム起動方法	595
25.5	自動アルゴリズム実行状態の確認	597
25.5.1	データポーリングフラグ (DQ7)	599
25.5.2	トグルビットフラグ (DQ6)	601
25.5.3	タイミングリミット超過フラグ (DQ5)	602
25.5.4	セクタ消去タイマフラグ (DQ3)	603

25.6	フラッシュメモリの書込み / 消去	604
25.6.1	フラッシュメモリの読出し / リセット状態	605
25.6.2	フラッシュメモリのデータ書込み	606
25.6.3	フラッシュメモリの全データ消去 (チップ消去)	608
25.6.4	フラッシュメモリの任意のデータ消去 (セクタ消去)	609
25.6.5	フラッシュメモリのセクタ消去を一時停止	611
25.6.6	フラッシュメモリのセクタ消去を再開	612
第 26 章	シリアル書込み接続例 (横河ディ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)	613
26.1	基本構成	614
26.2	発振クロック周波数とシリアルクロック入力周波数について	616
26.3	フラッシュマイコンプログラマのシステム構成	617
26.4	シリアル書込み接続例	618
26.4.1	シングルチップモード時の接続例 (ユーザ電源使用時)	619
26.4.2	フラッシュマイコンプログラマとの最小限の接続例 (ユーザ電源使用時)	621
第 27 章	シリアル書込み接続例 (富士通マイクロエレクトロニクス製 シリアルプログラマ)	623
27.1	富士通マイクロエレクトロニクス製シリアルプログラマ	624
27.1.1	使用する端子	628
付録	629
付録 A	メモリマップ	630
付録 B	命令	643
B.1	命令の種類	644
B.2	アドレッシング	645
B.3	直接アドレッシング	647
B.4	間接アドレッシング	654
B.5	実行サイクル数	661
B.6	実効アドレスフィールド	663
B.7	命令一覧表の読み方	665
B.8	F2MC-16LX 命令一覧表	668
B.9	命令マップ	682
索引	705

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。)	
-	-	レジスタ名を統一 インタラプト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)
-	-	USB を変更 USB Mini-Host USB ホスト
2, 4	第 1 章 概要 1.1 MB90330A シリーズの特長	最小命令実行時間を訂正 41.6 ns 41.7 ns
23	第 1 章 概要 1.7 デバイス使用上の注意	「●シリアル通信について」の項目を追加
26	第 2 章 CPU 2.1 CPU の概要 ■ CPU の概要	<p>「●最小命令実行時間」の説明を変更</p> <ul style="list-style-type: none"> ・41.7 ns / 6 MHz 原発振 4 通倍 (内部動作 24 MHz / 3.3 V ± 0.3 V 時) ・PLL クロック通倍方式 ・41.7 ns (マシンクロック 24 MHz のとき) ・マシンクロックの周波数は、品種によって異なります。 <p>「●CPU に依存しない自動転送機能」の説明を変更</p> <ul style="list-style-type: none"> ・最大 16 チャネルまでの μDMAC ・DMA 転送 (μDMAC) は、品種により内蔵していない場合があります。
41	第 2 章 CPU 2.6 レジスタ	表 2.6-1 の「許可される割込みレベル」を訂正
43		2.6.5 項のタイトルと内容を訂正
46	第 2 章 CPU 2.8 プリフィックス コード	表 2.8-1 の PC 空間のバンクセレクトプリフィックスを訂正 PCC PCB
94	第 3 章 割込み 3.8.2 μ DMAC のレジスタ	図 3.8-1 の DMA ストップステータスレジスタ (DSSR) を 訂正 bit8 ~ bit15 STP15 ~ STP8 bit0 ~ bit7 STP7 ~ STP0
99	第 3 章 割込み 3.8.2.3 DMA ストップス テータスレジスタ (DSSR)	< 注意事項 > を訂正 bit8 ~ bit15 STP15 ~ STP8 bit0 ~ bit7 STP7 ~ STP0
210	第 8 章 I/O ポート 8.2.2 ポート方向 レジスタ (DDR0 ~ DDRB)	図 8.2-2 のレジスタ名を訂正 PDR0, PDRA, PDRB DDR0 ~ DDR9, DDRA, DDRB
211		「●各端子がポートとして機能している場合」の説明を訂正

ページ	変更内容 (詳細は本文を参照してください。)	
331	第 14 章 USB ホスト 14.2 USB ホストの 制限事項	説明を変更 USB HOST との相違点 USB ホストの制限事項
433	第 17 章 8/16 ビット PPG タイマ 17.2 8/16 ビット PPG タイ マのレジスタ	[bit7 ~ bit5] PCS2 ~ PCS0 (カウントクロック選択) の表の 動作モード欄を訂正 41.6 ns 41.7 ns
434	■ PPG0 ~ PPG5 出力制御 レジスタ (PPG01/PPG23/ PPG45)	[bit4 ~ bit2] PCM2 ~ PCM0 (カウントクロック選択) の表の 動作モード欄を訂正 41.6 ns 41.7 ns
597, 598, 603	第 25 章 フラッシュ メモリ 25.5 自動アルゴリズム 実行状態の確認	「トグルビット 2 フラグ (DQ2)」の記述を削除
610	第 25 章 フラッシュ メモリ 25.6.4 フラッシュ メモリの任意のデータ 消去 (セクタ消去)	図 25.6-2 のフローチャートを訂正
649	B.3 直接アドレッシング ● I/O 直接 (io)	図 B.3-5 を変更。 MOVW A, i: 0C0H MOVW A, I:0C0H
		図 B.3-5 に (注意事項) を追加。
650	B.3 直接アドレッシング ● 短縮直接アドレス (dir)	図 B.3-6 に (注意事項) を追加。
651	B.3 直接アドレッシング ● I/O 直接ビットアドレ ス (io: bp)	図 B.3-8 を変更。 SETB i: 0C1H: 0 SETB I:0C1H:0
		図 B.3-8 に (注意事項) を追加。
	B.3 直接アドレッシング ● 短縮直接ビットアドレ ス (dir: bp)	図 B.3-9 に (注意事項) を追加。
657	B.4 間接アドレッシング ● プログラムカウンタ相 対分岐アドレス (rel)	説明を変更。 ディスプレースメント オフセット
		図 B.4-7 を変更。 BRA 10H BRA 3C32H 実行後 PC : 3C20 3C32
658	B.4 間接アドレッシング ● レジスタリスト (rlst)	図 B.4-9 を変更。 POPW, RW0, RW4 POPW RW0, RW4

ページ	変更内容 (詳細は本文を参照してください。)	
683	B.9 命令マップ ■ 命令マップの構造	表 B.9-1 の命令欄を変更。 @RW2+d8, #8, rel CBNE @RW2+d8, #8, rel
684		表 B.9-2 の E0 列 , +0 行 のオペランドを変更。 #4 #vct4
		表 B.9-2 の D0 列 , +0 行 のニーモニックを変更。 MOV MOVN
		表 B.9-2 の B0 列 , +0 行 のニーモニックを変更。 MOV MOVX
		表 B.9-2 の B0 列 , +8 行 のニーモニックを変更。 MOV MOVW
686		表 B.9-4 の E0 列 , +0 行 のニーモニックを変更。 FILSI FILSWI
687		表 B.9-5 の 70 列 , +A 行 のニーモニックを変更。 DIVU DIV
688		表 B.9-6 の F0 列 , +E, +F 行 のオペランドを変更。 ,#8, rel #8, rel
691		表 B.9-9 の 50 列 , +8 ~ +E 行 のオペランドを変更。 @@ @
		表 B.9-9 の 20 列 , +0 ~ +7 行 のオペランドを変更。 RWi @RWi
692		表 B.9-10 の E0 列 , F0 列 のオペランドを変更。 ,r ,rel
693		表 B.9-11 の 70 列 のオペランドを変更。 NEG A, NEG
694		表 B.9-12 の E0 列 , F0 列 のオペランドを変更。 ,r ,rel
702		表 B.9-20 XCH Ri, ea 命令 (第 1 バイト =7EH) を 変更。 (項目「A」を「A0」に変更 , +A の行の「W2+d16,A」を「@RW2+d16」に変更)

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

この章では、MB90330A シリーズの特長、ブロックダイアグラム、機能概要など全体を知るための基本的なことがらについて説明します。

- 1.1 MB90330A シリーズの特長
- 1.2 ブロックダイアグラム
- 1.3 外形寸法図
- 1.4 端子配列図
- 1.5 端子機能
- 1.6 入出力回路形式
- 1.7 デバイス使用上の注意

1.1 MB90330A シリーズの特長

MB90330A シリーズは、パソコン周辺機器などの USB 通信が要求される用途向けに設計された 16 ビットマイクロコントローラです。USB 機能は、12 Mbps の Function 動作ばかりでなく、簡易的な Host 動作が可能です。ディスプレイやオーディオなどのパソコン周辺機器や USB 通信対応の携帯機器制御などに適した機能を搭載しています。

■ MB90330A シリーズの特長

MB90330A シリーズには、以下の特長があります。

- PLL クロック逡倍回路内蔵
 - 原発振の 2 分周もしくは原発振の 1 逡倍、2 逡倍、4 逡倍
原発振 6 MHz 時、3 MHz ~ 24 MHz の動作クロック (PLL クロック) を選択可能
USB 用クロックは 48 MHz
 - 最小命令実行時間: 41.7 ns (原発振 6 MHz, PLL クロック 4 逡倍, V_{CC}=3.3 V 動作時)
- 最大メモリ空間: 16 M バイト
- コントロール用途に最適化された命令体系
 - 取扱い可能なデータタイプ: ビット / バイト / ワード / ロングワード
 - 標準アドレッシングモード: 23 種類
 - 32 ビットアキュムレータの採用による高精度演算の強化
 - 符号付き乗除算、拡張 RETI 命令
- 高級言語 (C 言語) マルチタスクに対応をとった命令体系
 - システムスタックポイントの採用
 - 命令セットの対称性とバレルシフト命令
- ノンマルチバス / マルチバス対応
- 実行速度の向上: 4 バイトのキュー
- 強力な割込み機能 (プライオリティがプログラマブルに 8 レベル設定可能): 外部割込み 8 本
- データ転送機能
 - μ DMAC: 最大 16 チャンネル
 - 拡張インテリジェント I/O サービス: 最大 16 チャンネル
- 内蔵 ROM 容量と ROM タイプ
 - マスク ROM: 256 K バイト、384 K バイト
 - フラッシュ ROM: 384 K バイト

- 内蔵 RAM
 - ・ 量産品 : 16 K バイト , 24 K バイト
 - ・ フラッシュ品 : 24 K バイト , 30K バイト
 - ・ エバリュエーションチップ : 28 K バイト
- プロセス : CMOS テクノロジ
- 低消費電力 (スタンバイ) モード
 - ・ スリープモード (CPU 動作クロックを停止するモード)
 - ・ ストップモード (原発振を停止するモード)
 - ・ CPU 間欠動作モード
- パッケージ
 - ・ LQFP-120 (FPT-120P-M24: 0.4 mm ピンピッチ)
 - ・ LQFP-120 (FPT-120P-M21: 0.5 mm ピンピッチ)
- 動作保証温度 : - 40 ~ + 85 (USB 使用時 : 0 ~ + 70)
- 汎用ポート : 最大 94 本
 - 汎用入出力 (CMOS) : 56 本
 - 汎用入出力 (入力プルアップ抵抗設定可能) : 16 本
 - 汎用入出力 (出力オープンドレイン / 5 V 耐圧 I/O) : 22 本
- タイマ : タイムベースタイマ / ウォッチドッグタイマ / 時計タイマ : 1 チャンネル
- 16 ビット入出力タイマ
 - ・ 16 ビットフリーランタイマ : 1 チャンネル
 - ・ インプットキャプチャ (ICU) : 4 チャンネル
 - ・ アウトプットコンペア (OCU) : 4 チャンネル
- 8/16 ビット PPG タイマ : 8 ビット × 6 チャンネルまたは 16 ビット × 3 チャンネル
- 16 ビットリロードタイマ : 3 チャンネル
- 16 ビット PWC タイマ : 1 チャンネル
- UART: 4 チャンネル
- I/O 拡張シリアルインタフェース : 1 チャンネル
- I²C インタフェース : 3 チャンネル
- 8/10 ビット A/D コンバータ (RC 逐次比較型) : 16 チャンネル
- DTP/ 外部割込み : 8 チャンネル
- USB
 - ・ USB ファンクション (USB Full Speed 対応) : 1 チャンネル
 - ・ USB ホスト : 1 チャンネル

■ 品種構成

表 1.1-1 MB90330A シリーズ品種構成一覧 (1 / 2)

品種名		MB90V330A*	MB90F334A	MB90F335A	MB90333A
分類		評価用品	フラッシュメモリ品		マスク ROM 品
ROM 容量		搭載なし	384 K バイト	512 K バイト	256 K バイト
RAM 容量		28 K バイト	24 K バイト	30 K バイト	16 K バイト
エミュレータ専用電源		あり			
CPU 機能		基本命令数 : 351 命令 命令ビット長 : 8 ビット, 16 ビット 最小命令実行時間 : 41.7 ns / 24 MHz アドレッシング種類 : 23 種類 最大メモリ空間 : 16 M バイト			
ポート		入出力ポート (CMOS) : 94 本			
16ビット 入出力 タイマ	16 ビット フリーランタイマ	チャンネル数 : 1 オーバフロー割込み			
	アウトプット コンペア (OCU)	チャンネル数 : 4 端子入力要因 : コンペアレジスタの一致信号による			
	インプット キャプチャ (ICU)	チャンネル数 : 4 端子入力 (立上り, 立下り, 両エッジ) によるレジスタの書換え			
8/16 ビット PPG タイマ		チャンネル数 : 8 ビット × 6 チャンネル, 16 ビット × 3 チャンネル モード切換え機能付き 8 ビットまたは 16 ビットの PPG 動作 任意周期, 任意デューティのパルス波出力可能			
16 ビット リロードタイマ		チャンネル数 : 3 16 ビットリロードタイマ動作 イベントカウンタ付き			
16 ビット PWC タイマ		チャンネル数 : 1 タイマ機能 (3 つの内部クロックからカウンタのクロックを選択) パルス幅測定機能 (3 つの内部クロックからカウンタのクロックを選択)			
UART		チャンネル数 : 4 クロック同期 / 非同期の選択可能 専用ボーレートジェネレータ内蔵 クロック同期 LSB, MSB 切換え可能			
I/O 拡張 シリアルインタフェース		チャンネル数 : 1 クロック同期転送 LSB ファースト / MSB ファースト			
I ² C バス通信		チャンネル数 : 3 Inter IC BUS をサポートするシリアル I/O			
8/10 ビット A/D コンバータ		10 ビット分解能アナログ入力 16 チャンネル			

表 1.1-1 MB90330A シリーズ品種構成一覧 (2 / 2)

品種名	MB90V330A*	MB90F334A	MB90F335A	MB90333A
DTP/ 外部割込み	入力本数 : 8 本 割込み要因 : 立上りエッジ / 立下りエッジ / "L" レベル / "H" レベル 選択可			
USB	USB ファンクション (USB Full Speed 対応) Full speed をサポート Endpoint を最大 6 つまで指定可能 転送タイプ : Control, Interrupt, Bulk, Isochronous 転送可能 デュアルポート RAM(FIFO モードもサポート) USB ホスト 機能			
μDMAC	対応			
外部バスインタフェース	あり (マルチ / ノンマルチ対応)			
その他	5 V 耐圧 I/O 22 本 (I ² C 兼用端子を含む)			
パッケージ	PGA299	LQFP120		
動作電圧	3.3 V ± 0.3 V			

*: 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。

< 注意事項 >

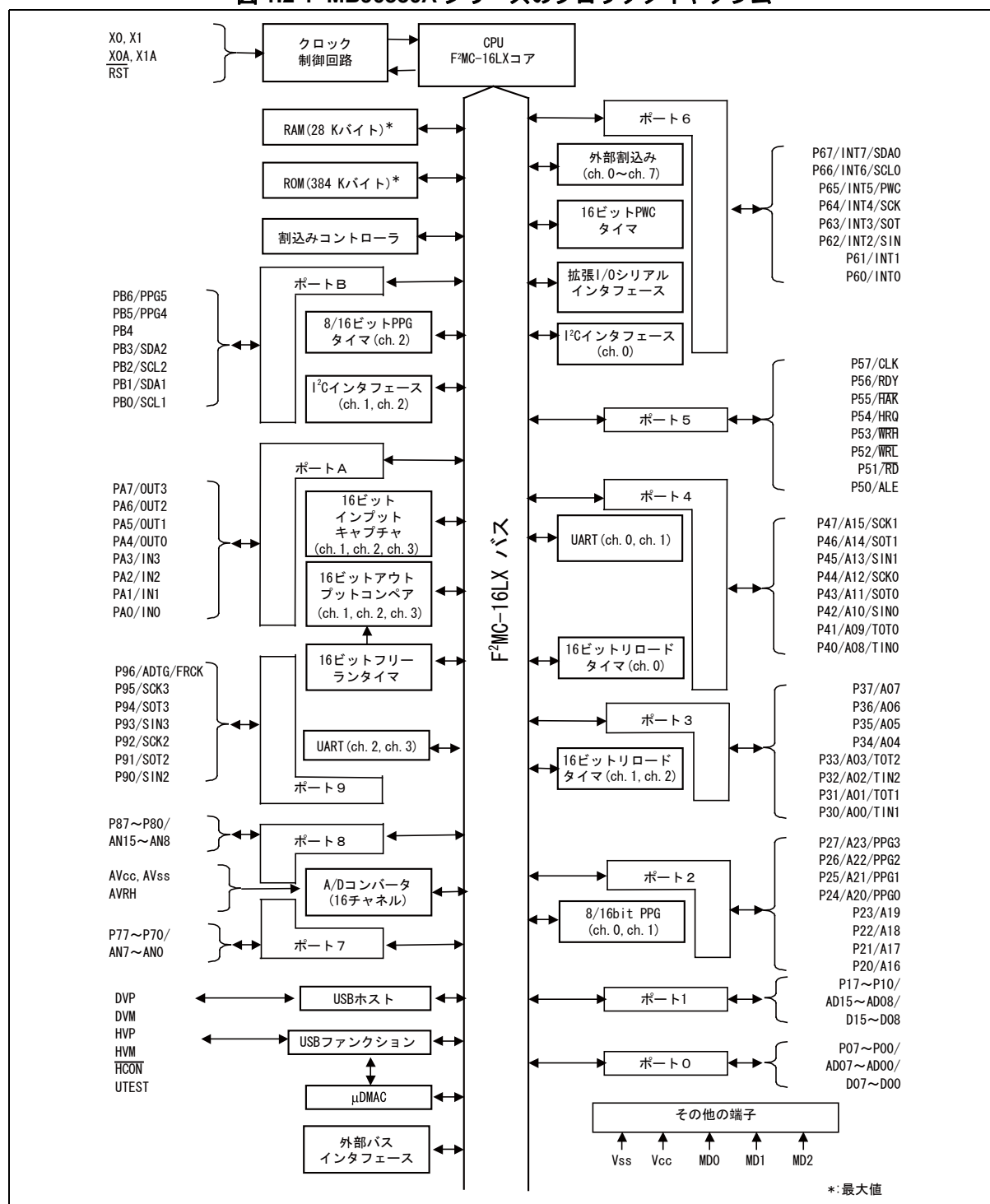
Flash への書込みは, 必ず, $V_{CC}=3.13\text{ V} \sim 3.60\text{ V}$ 以上で行ってください。

1.2 ブロックダイアグラム

図 1.2-1 に、MB90330A シリーズのブロックダイアグラムを示します。

■ MB90330A シリーズのブロックダイアグラム

図 1.2-1 MB90330A シリーズのブロックダイアグラム



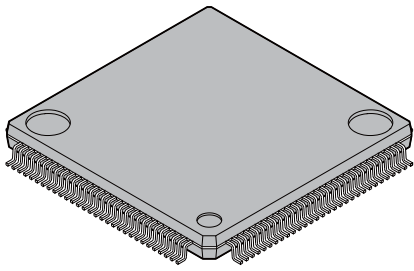
< 注意事項 >

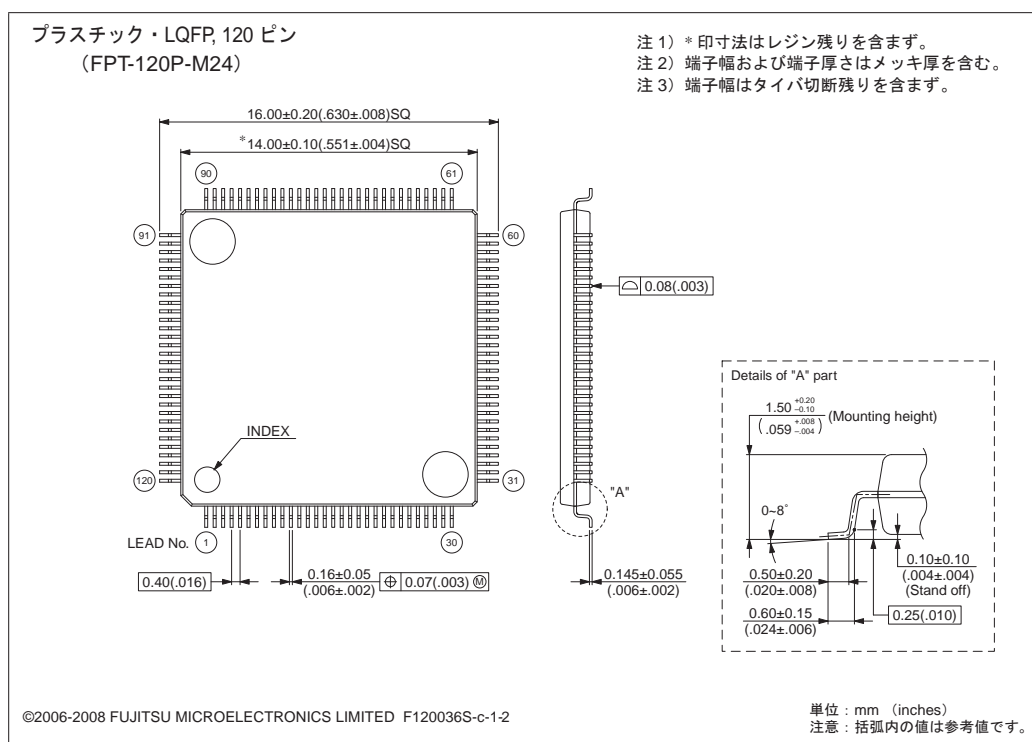
図 1.2-1 において，I/O ポートは内蔵の各機能ブロックと端子を共用しています。内蔵モジュールの端子として使用する場合は，I/O ポートとしては使えません。

1.3 外形寸法図

MB90330A シリーズには、2 種類のパッケージが用意されています。

■ 外形寸法図 (LQFP-120)

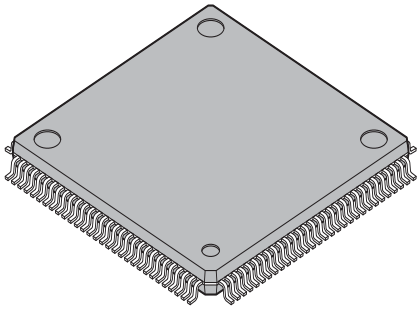
<p>プラスチック・LQFP, 120 ピン</p>  <p>(FPT-120P-M24)</p>	リードピッチ	0.40 mm
	パッケージ幅 × パッケージ長さ	14.0 mm × 14.0 mm
	リード形状	ガルウィング
	封止方法	プラスチックモールド
	取付け高さ	1.70 mm MAX
	コード (参考)	P-LQFP120-14×14-0.40

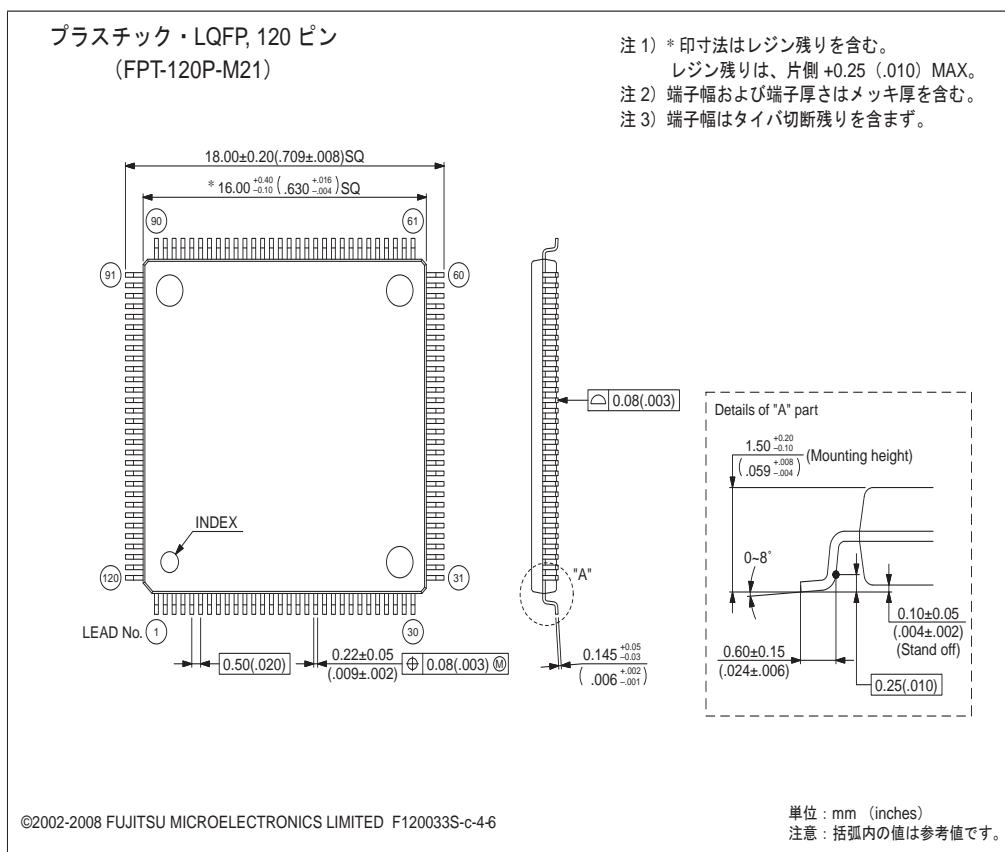


最新の外形寸法図については、下記 URL にてご確認ください。

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

■ 外形寸法図 (LQFP-120)

<p>プラスチック・LQFP, 120 ピン</p>  <p>(FPT-120P-M21)</p>	リードピッチ	0.50mm
	パッケージ幅× パッケージ長さ	16.0 × 16.0mm
	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	1.70mm MAX
	質量	0.88g
	コード (参考)	P-LFQFP120-16×16-0.50



最新の外形寸法図については、下記 URL にてご確認ください。

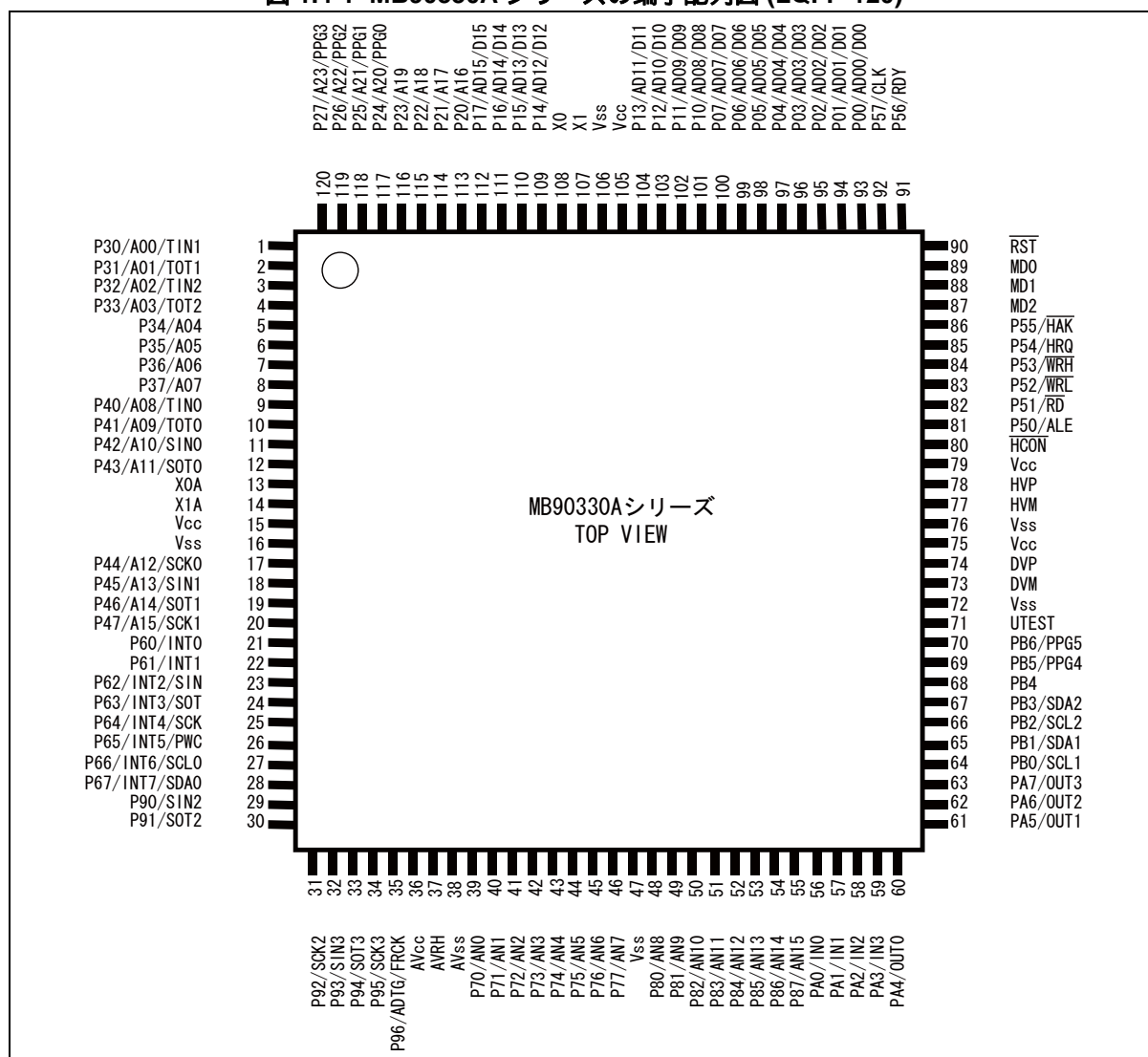
<http://edevic.fujitsu.com/package/jp-search/>

1.4 端子配列図

図 1.4-1 に、MB90330A シリーズの端子配列図を示します。

■ 端子配列図 (LQFP-120)

図 1.4-1 MB90330A シリーズの端子配列図 (LQFP-120)



1.5 端子機能

表 1.5-1 に、MB90330A シリーズの端子機能について説明します。

■ 端子機能

表 1.5-1 端子機能 (1 / 7)

端子番号	端子名	回路形式	機能説明
107	X1	A	発振端子です。
108	X0	A	発振端子です。
13	X0A	A	32 kHz 発振端子です。
14	X1A	A	32 kHz 発振端子です。
90	$\overline{\text{RST}}$	F	リセット入力端子です。
93 ~ 100	P00 ~ P07	H	汎用入出力ポートです。 プルアップ抵抗設定レジスタ (RDR0) により、プルアップ抵抗を付加 (RD00 ~ RD07=1) に設定できます (出力設定時は無効)。
	AD00 ~ AD07		マルチプレクスモード時、外部アドレス/データバス下位入出力端子として機能します。
	D00 ~ D07		ノンマルチプレクスモード時、外部データバス下位入出力端子として機能します。
101 ~ 104	P10 ~ P13	H	汎用入出力ポートです。 プルアップ抵抗設定レジスタ (RDR1) により、プルアップ抵抗を付加 (RD10 ~ RD13=1) に設定できます (出力設定時は無効)。
	AD08 ~ AD11		マルチプレクスモード時、外部アドレス/データバス上位入出力端子として機能します。
	D08 ~ D11		ノンマルチプレクスモード時、外部データバス上位入出力端子として機能します。
109 ~ 112	P14 ~ P17	H	汎用入出力ポートです。 プルアップ抵抗設定レジスタ (RDR1) により、プルアップ抵抗を付加 (RD14 ~ RD17=1) に設定できます (出力設定時は無効)。
	AD12 ~ AD15		マルチプレクスモード時、外部アドレス/データバス上位入出力端子として機能します。
	D12 ~ D15		ノンマルチプレクスモード時、外部データバス上位入出力端子として機能します。

表 1.5-1 端子機能 (2 / 7)

端子番号	端子名	回路形式	機能説明
113 ~ 116	P20 ~ P23	D	汎用入出力ポートです。外バスモード時は、外部アドレス出力制御レジスタ (HACR) の対応するビットが "1" のとき、汎用入出力ポートとして機能します。
	A16 ~ A19		マルチプレクスモード時、外部アドレス出力制御レジスタ (HACR) の対応するビットが "0" のとき、アドレス上位出力端子 (A16 ~ A19) として機能します。
	A16 ~ A19		ノンマルチプレクスモード時、外部アドレス出力制御レジスタ (HACR) の対応するビットが "0" のとき、アドレス上位出力端子 (A16 ~ A19) として機能します。
117 ~ 120	P24 ~ P27	D	汎用入出力ポートです。外バスモード時は、外部アドレス出力制御レジスタ (HACR) の対応するビットが "1" のとき、汎用入出力ポートとして機能します。
	A20 ~ A23		マルチプレクスモード時、外部アドレス出力制御レジスタ (HACR) の対応するビットが "0" のとき、アドレス上位出力端子 (A20 ~ A23) として機能します。
	A20 ~ A23		ノンマルチプレクスモード時、外部アドレス出力制御レジスタ (HACR) の対応するビットが "0" のとき、アドレス上位出力端子 (A20 ~ A23) として機能します。
	PPG0 ~ PPG3		PPG タイマ ch.0 ~ ch.3 出力端子として機能します。
1	P30	D	汎用入出力ポートです。
	A00		ノンマルチバスモード時、外部アドレス端子として機能します。
	TIN1		16 ビットリロードタイマ ch.1 のイベント入力端子として機能します。
2	P31	D	汎用入出力ポートです。
	A01		ノンマルチバスモード時、外部アドレス端子として機能します。
	TOT1		16 ビットリロードタイマ ch.1 の出力端子として機能します。
3	P32	D	汎用入出力ポートです。
	A02		ノンマルチバスモード時、外部アドレス端子として機能します。
	TIN2		16 ビットリロードタイマ ch.2 のイベント入力端子として機能します。
4	P33	D	汎用入出力ポートです。
	A03		ノンマルチバスモード時、外部アドレス端子として機能します。
	TOT2		16 ビットリロードタイマ ch.2 の出力端子として機能します。

表 1.5-1 端子機能 (3 / 7)

端子番号	端子名	回路形式	機能説明
5 ~ 8	P34 ~ P37	D	汎用入出力ポートです。
	A04 ~ A07		ノンマルチバスモード時，外部アドレス端子として機能します。
9	P40	G	汎用入出力ポートです。
	A08		ノンマルチバスモード時，外部アドレス端子として機能します。
	TIN0		16ビットリロードタイマ ch.0 のイベント入力端子として機能します。
10	P41	G	汎用入出力ポートです。
	A09		ノンマルチバスモード時，外部アドレス端子として機能します。
	TOT0		16ビットリロードタイマ ch.0 の出力端子として機能します。
11	P42	G	汎用入出力ポートです。
	A10		ノンマルチバスモード時，外部アドレス端子として機能します。
	SIN0		UART ch.0 データ入力端子として機能します。
12	P43	G	汎用入出力ポートです。
	A11		ノンマルチバスモード時，外部アドレス端子として機能します。
	SOT0		UART ch.0 データ出力端子として機能します。
17	P44	G	汎用入出力ポートです。
	A12		ノンマルチバスモード時，外部アドレス端子として機能します。
	SCK0		UART ch.0 クロック入出力端子として機能します。
18	P45	G	汎用入出力ポートです。
	A13		ノンマルチバスモード時，外部アドレス端子として機能します。
	SIN1		UART ch.1 データ入力端子として機能します。
19	P46	G	汎用入出力ポートです。
	A14		ノンマルチバスモード時，外部アドレス端子として機能します。
	SOT1		UART ch.1 データ出力端子として機能します。

表 1.5-1 端子機能 (4 / 7)

端子番号	端子名	回路形式	機能説明
20	P47	G	汎用入出力ポートです。
	A15		ノンマルチバスモード時，外部アドレス端子として機能します。
	SCK1		UART ch.1 クロック入出力端子として機能します。
81	P50	L	汎用入出力ポートです。
	ALE		外バスモード時，アドレス取込み許可信号 (ALE) 端子として機能します。
82	P51	L	汎用入出力ポートです。
	\overline{RD}		外バスモード時，リードストロブ出力 (\overline{RD}) 端子として機能します。
83	P52	L	汎用入出力ポートです。
	\overline{WRL}		外バスモード時，下位側データライトストロブ出力 (\overline{WRL}) 端子として機能します。EPCR レジスタの WRE ビットが "0" のとき，汎用入出力ポートとして機能します。
84	P53	L	汎用入出力ポートです。
	\overline{WRH}		バス幅 16 ビットの外バスモード時，上位側データライトストロブ出力 (\overline{WRH}) 端子として機能します。EPCR レジスタの WRE ビットが "0" のとき，汎用入出力ポートとして機能します。
85	P54	L	汎用入出力ポートです。
	HRQ		外バスモード時，ホールド要求入力 (HRQ) 端子として機能します。EPCR レジスタの HDE ビットが "0" のとき，汎用入出力ポートとして機能します。
86	P55	L	汎用入出力ポートです。
	\overline{HAK}		外バスモード時，ホールドアクノリッジ出力 (\overline{HAK}) 端子として機能します。EPCR レジスタの HDE ビットが "0" のとき，汎用入出力ポートとして機能します。
91	P56	L	汎用入出力ポートです。
	RDY		外バスモード時，外部レディ入力 (RDY) 端子として機能します。EPCR レジスタの RYE ビットが "0" のとき，汎用入出力ポートとして機能します。
92	P57	L	汎用入出力ポートです。
	CLK		外バスモード時，マシンサイクルクロック出力 (CLK) 端子として機能します。EPCR レジスタの CKE ビットが "0" のとき，汎用入出力ポートとして機能します。
21, 22	P60, P61	C	汎用入出力ポートです (5 V 耐圧)。
	INT0, INT1		外部割込み ch.0, ch.1 入力端子として機能します。

表 1.5-1 端子機能 (5 / 7)

端子番号	端子名	回路形式	機能説明
23	P62	C	汎用入出力ポートです (5 V 耐圧)。
	INT2		外部割込み ch.2 入力端子として機能します。
	SIN		拡張シリアル I/O データ入力端子です。
24	P63	C	汎用入出力ポートです (5 V 耐圧)。
	INT3		外部割込み ch.3 入力端子として機能します。
	SOT		拡張シリアル I/O データ出力端子です。
25	P64	C	汎用入出力ポートです (5 V 耐圧)。
	INT4		外部割込み ch.4 入力端子として機能します。
	SCK		拡張シリアル I/O クロック入出力端子です。
26	P65	C	汎用入出力ポートです (5 V 耐圧)。
	INT5		外部割込み ch.5 入力端子として機能します。
	PWC		PWC 入力端子として機能します。
27	P66	C	汎用入出力ポートです (5 V 耐圧)。
	INT6		外部割込み ch.6 入力端子として機能します。
	SCL0		I ² C インタフェース ch.0 クロック入出力端子として機能します。 I ² C インタフェース動作中は、ポート出力を Hi-Z としてください。
28	P67	C	汎用入出力ポートです (5 V 耐圧)。
	INT7		外部割込み ch.7 入力端子として機能します。
	SDA0		I ² C インタフェース ch.0 データ入出力端子として機能します。 I ² C インタフェース動作中は、ポート出力を Hi-Z としてください。
39 ~ 46	P70 ~ P77	I	汎用入出力ポートです。
	AN0 ~ AN7		アナログ ch.0 ~ ch.7 入力端子として機能します。
48 ~ 55	P80 ~ P87	I	汎用入出力ポートです。
	AN8 ~ AN15		アナログ ch.8 ~ ch.15 入力端子として機能します。
29	P90	D	汎用入出力ポートです。
	SIN2		UART ch.2 データ入力端子として機能します。
30	P91	D	汎用入出力ポートです。
	SOT2		UART ch.2 データ出力端子として機能します。

表 1.5-1 端子機能 (6 / 7)

端子番号	端子名	回路形式	機能説明
31	P92	D	汎用入出力ポートです。
	SCK2		UART ch.2 クロック入出力端子として機能します。
32	P93	D	汎用入出力ポートです。
	SIN3		UART ch.3 データ入力端子として機能します。
33	P94	D	汎用入出力ポートです。
	SOT3		UART ch.3 データ出力端子として機能します。
34	P95	D	汎用入出力ポートです。
	SCK3		UART ch.3 クロック入出力端子として機能します。
35	P96	C	汎用入出力ポートです (5 V 耐圧)。
	ADTG		A/D コンバータ使用時に外部トリガ入力端子として機能します。
	FRCK		フリーランタイム使用時の外部クロック入力端子として機能します。
56 ~ 59	PA0 ~ PA3	C	汎用入出力ポートです (5 V 耐圧)。
	IN0 ~ IN3		インプットキャプチャ ch.0 ~ ch.3 のトリガ入力として取り込まれます。
60 ~ 63	PA4 ~ PA7	C	汎用入出力ポートです (5 V 耐圧)。
	OUT0 ~ OUT3		アウトプットコンペア ch.0 ~ ch.3 イベント出力端子として機能します。
64	PB0	C	汎用入出力ポートです (5 V 耐圧)。
	SCL1		I ² C インタフェース ch.1 クロック入出力端子として機能します。I ² C インタフェース動作中は、ポート出力を Hi-Z としてください。
65	PB1	C	汎用入出力ポートです (5 V 耐圧)。
	SDA1		I ² C インタフェース ch.1 データ入出力端子として機能します。I ² C インタフェース動作中は、ポート出力を Hi-Z としてください。
66	PB2	C	汎用入出力ポートです (5 V 耐圧)。
	SCL2		I ² C インタフェース ch.2 クロック入出力端子として機能します。I ² C インタフェース動作中は、ポート出力を Hi-Z としてください。
67	PB3	C	汎用入出力ポートです (5 V 耐圧)。
	SDA2		I ² C インタフェース ch.2 データ入出力端子として機能します。I ² C インタフェース動作中は、ポート出力を Hi-Z としてください。

表 1.5-1 端子機能 (7 / 7)

端子番号	端子名	回路形式	機能説明
68	PB4	C	汎用入出力ポートです (5 V 耐圧)。
69	PB5	D	汎用入出力ポートです。
	PPG4		PPG タイマ ch.4 出力端子として機能します。
70	PB6	D	汎用入出力ポートです。
	PPG5		PPG タイマ ch.5 出力端子として機能します。
71	UTEST	C	USB テスト用端子です。通常使用時は、プルダウン接続願います。
73	DVM	K	USB ファンクション D - 端子です。
74	DVP	K	USB ファンクション D + 端子です。
77	HVM	K	USB ホスト D - 端子です。
78	HVP	K	USB ホスト D + 端子です。
80	$\overline{\text{HCON}}$	E	外付けプルアップ抵抗接続端子です。
36	AVcc	-	A/D コンバータ電源端子です。
37	AVRH	J	A/D コンバータ外部基準電源端子です。
38	AVss	-	A/D コンバータ電源端子です。
87 ~ 89	MD2 ~ MD0	B	動作モード指定用入力端子です。
15, 17, 79, 105	Vcc	-	電源端子です。
16, 47, 72, 76, 106	Vss	-	電源端子 (GND) です。

1.6 入出力回路形式

表 1.6-1 に , MB90330A 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 1.6-1 入出力回路形式 (1 / 3)

分類	回路	備考
A		<ul style="list-style-type: none"> ・ 発振帰還抵抗 : X1, X0 約 1 MΩ X1A, X0A 10 MΩ ・ スタンバイコントロール付き
B		<ul style="list-style-type: none"> ・ CMOS ヒステリシス入力
C		<ul style="list-style-type: none"> ・ CMOS ヒステリシス入力 ・ N-ch オープンドレイン出力
D		<ul style="list-style-type: none"> ・ CMOS 出力 ・ CMOS ヒステリシス入力 ・ スタンバイコントロール付き

表 1.6-1 入出力回路形式 (2 / 3)

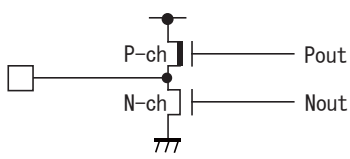

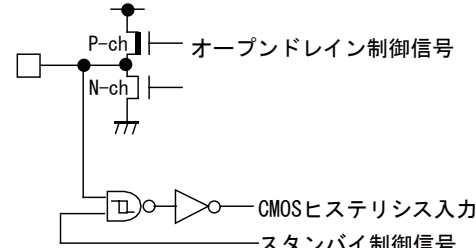
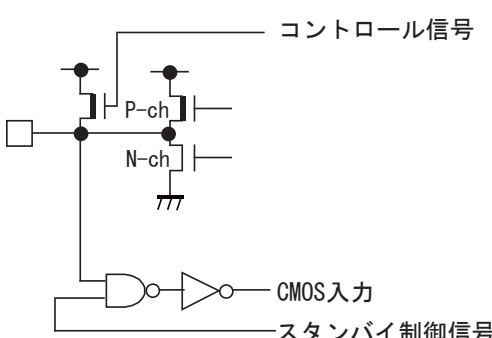
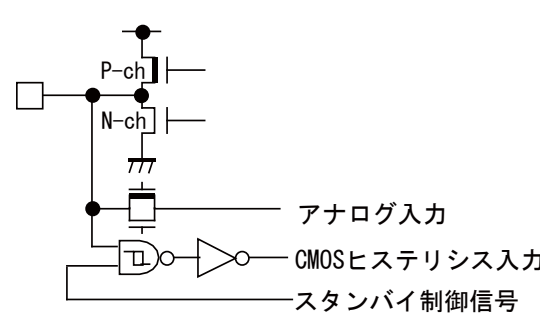
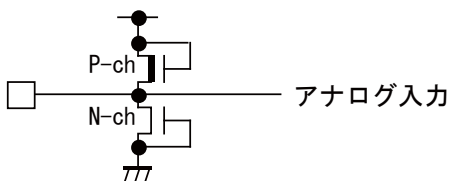
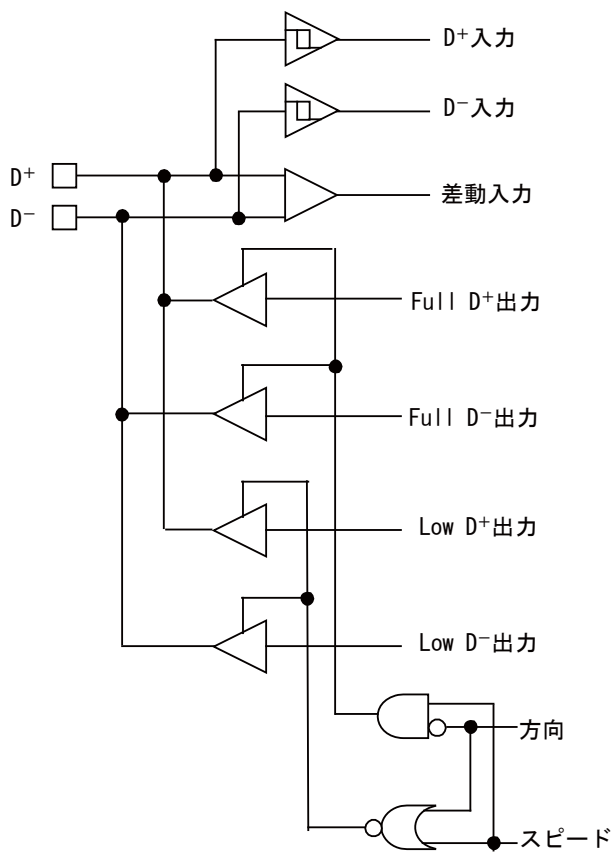
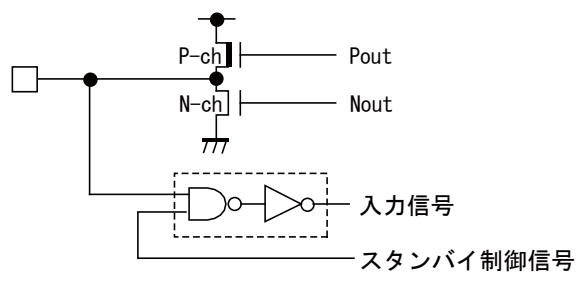
分類	回路	備考
E		<ul style="list-style-type: none"> CMOS 出力
F		<ul style="list-style-type: none"> プルアップ付きCMOSヒステリシス入力 抵抗 : 約 50 kΩ
G		<ul style="list-style-type: none"> CMOS 出力 CMOS ヒステリシス入力 オープンドレインコントロール付き スタンバイコントロール付き
H		<ul style="list-style-type: none"> CMOS 出力 CMOS 入力 入力プルアップ抵抗制御付き 抵抗 : 約 50 kΩ スタンバイコントロール付き
I		<ul style="list-style-type: none"> CMOS 出力 CMOS ヒステリシス入力 スタンバイコントロール付き アナログ入力

表 1.6-1 入出力回路形式 (3 / 3)

分類	回路	備考
J		<ul style="list-style-type: none"> ・ A/D コンバータ電圧入力端子
K		<ul style="list-style-type: none"> ・ USB 入出力端子
L		<ul style="list-style-type: none"> ・ CMOS 出力 ・ CMOS 入力 (スタンバイコントロール付き)

1.7 デバイス使用上の注意

デバイス使用上の注意について説明します。

■ デバイス使用上の注意

● ラッチアップの防止，電源投入について

CMOS IC では，次に示すような場合にラッチアップ現象を生じることがあります。

- 入力端子や出力端子に V_{CC} より高い電圧や V_{SS} より低い電圧が印加された場合
- V_{CC} 端子～ V_{SS} 端子間に定格を超える電圧が印加された場合
- V_{CC} の電圧より， AV_{CC} の電源が先に供給された場合

アナログ系の電圧印加は，必ず V_{CC} と同時に行うか，デジタル電源の投入後に行ってください（電源 OFF の場合は，アナログ系電源を先に落とすか，同時に落としてください）。

ラッチアップが起きると電源電流が激増し，素子の熱破壊に至ることがあります。使用に際しては，十分に注意してください。

● 未使用端子の処理

入力に用いる未使用端子を開放のままにしておくと，誤動作およびラッチアップによる永久破壊の原因になることがあります。使用していない入力端子は， $2k\Omega$ 以上の抵抗を介して，プルアップまたはプルダウンの処置をしてください。使用していない入出力端子は，出力状態に設定して開放とするか，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は開放としてください。

● A/D コンバータ未使用時の端子処理

A/D コンバータを使用しない場合は， $AV_{CC} = AVR_H = V_{CC}$ ， $AV_{SS} = V_{SS}$ となるように接続してください。

● 電源端子 (V_{CC}/V_{SS}) の取扱い

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ブ信号の誤動作の防止，総出力電流規格を遵守などのため，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，電源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} と V_{SS} の間に $0.1\mu F$ 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。

● 水晶発振回路

X0/X1 端子および X0A/X1A 端子の近くにノイズがあると、デバイスの誤動作の原因となる可能性があります。X0/X1, X0A/X1A, 水晶発振子（またはセラミック発振子）およびグラウンドへのバイパスコンデンサをできるだけデバイスの近くに配置するようにプリント板を設計してください。

グラウンドで X0/X1 端子および X0A/X1A 端子を取り囲むようにプリント板アートを設計すると、安定した動作が得られますので、この設計を強く推奨します。

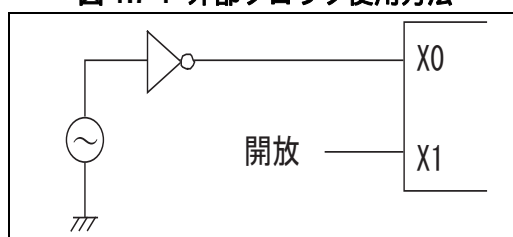
各量産品において、ご使用される発振子メーカーに発振評価依頼をしてください。

● 外部クロック使用時の注意

外部クロックをご使用になる場合は、外部端子を図 1.7-1 のように接続してください。

図 1.7-1 に外部クロック使用方法を示します ($f=7\text{ MHz}$ 以下)。

図 1.7-1 外部クロック使用方法



● 供給電源の安定化

V_{CC} 電源電圧の動作範囲内においても、電源電圧の急激な変化があると誤動作を生じることがあります。安定化の基準として、商用周波数 (50 Hz / 60 Hz) での V_{CC} リップル変動 (P-P 値) は、標準 V_{CC} 値の 10% 以下になるように、また電源の切換え時などの瞬時変化においては、過渡変動率が 0.1 V/ms 以下になるように電源電圧を安定させてください。

● 低電圧使用時の水晶発振回路

2.0 V 以下の電圧でご使用の場合には、電源 ON 時に外部の水晶発振子が発振しない場合があります。このため、外部クロックの使用を推奨致します。

● 2 系統品を 1 系統で使用する場合

2 系統である MB90330A シリーズを 1 系統で使用する場合、 $X0A=V_{SS}$: $X1A$ =開放として使用してください。

● フラッシュメモリへの書込みについて

フラッシュメモリへのシリアル書込みは、必ず、動作電圧: $V_{CC}=3.13\text{ V} \sim 3.6\text{ V}$ 内で行ってください。

フラッシュメモリへの通常書込みは、必ず、動作電圧: $V_{CC}=3.0\text{ V} \sim 3.6\text{ V}$ 内で行ってください。

● PLL クロックモード動作中の注意について

本マイコンで PLL クロックを選択しているときに発振子が外れたり、あるいはクロック入力が停止した場合、本マイコンは PLL 内部の自励発振回路の自走周波数で動作を継続する場合があります。この動作は保証外の動作です。

- シリアル通信について

シリアル通信においては、ノイズなどにより間違ったデータを受信する可能性があります。そのため、ノイズを抑えるボードの設計をしてください。

また、万が一ノイズなどの影響により誤ったデータを受信した場合を考慮し、最後にデータのチェックサムなどを付加してエラー検出を行ってください。エラーが検出された場合には、再送を行うなどの処理をしてください。

第2章

CPU

CPU の機能と動作について説明します。

- 2.1 CPU の概要
- 2.2 メモリ空間
- 2.3 リニア方式によるアドレス指定
- 2.4 バンク方式によるアドレス指定
- 2.5 メモリ空間における多バイト長データ
- 2.6 レジスタ
- 2.7 レジスタバンク
- 2.8 プリフィックスコード
- 2.9 割込み抑止命令

管理番号 : CM44-00101-2

固有箇所 : 26, 26

2.1 CPU の概要

F²MC-16LX CPU コアは、民生用・車載用機器などの高速リアルタイム処理が要求される用途向けに設計された 16 ビット CPU です。F²MC-16LX の命令セットはコントローラ用途向けに設計されており、各種制御の高速・高効率処理が可能です。

■ CPU の概要

F²MC-16LX CPU コアは 16 ビットデータ処理はもちろん、内部に 32 ビットアキュムレータを搭載しているため 32 ビットデータ処理も可能です。メモリ空間は最大 16M バイト（拡張可能）、リニア方式およびバンク方式のいずれかにてアクセス可能です。また、命令体系は F²MC-8L の A-T アーキテクチャをベースに、高級言語対応命令の追加・アドレッシングモードの拡張・乗除算命令の強化・ビット処理の充実化により命令が強化されています。

以下に、F²MC-16LX CPU の特長を示します。

- 最小命令実行時間

- 41.7 ns（マシクロック 24 MHz のとき）
- マシクロックの周波数は、品種によって異なります。

- 最大メモリ空間

16M バイト、リニア / バンク方式にてアクセス

- コントローラ用途に最適化された命令体系

- 豊富なデータタイプビット / バイト / ワード / ロングワード
- 拡張されたアドレッシングモード 23 種類
- 32 ビットアキュムレータの採用による高精度演算 (32 ビット長) の強化

- 強力な割込み機能

8 つの優先順位（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 (EI²OS)
- 最大 16 チャンネルまでの DMA 転送 (μDMAC)
- DMA 転送 (μDMAC) は、品種により内蔵していない場合があります。

- 高級言語 (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 / バレルシフト命令

- 実行速度の向上

4 バイトのキュー

2.2 メモリ空間

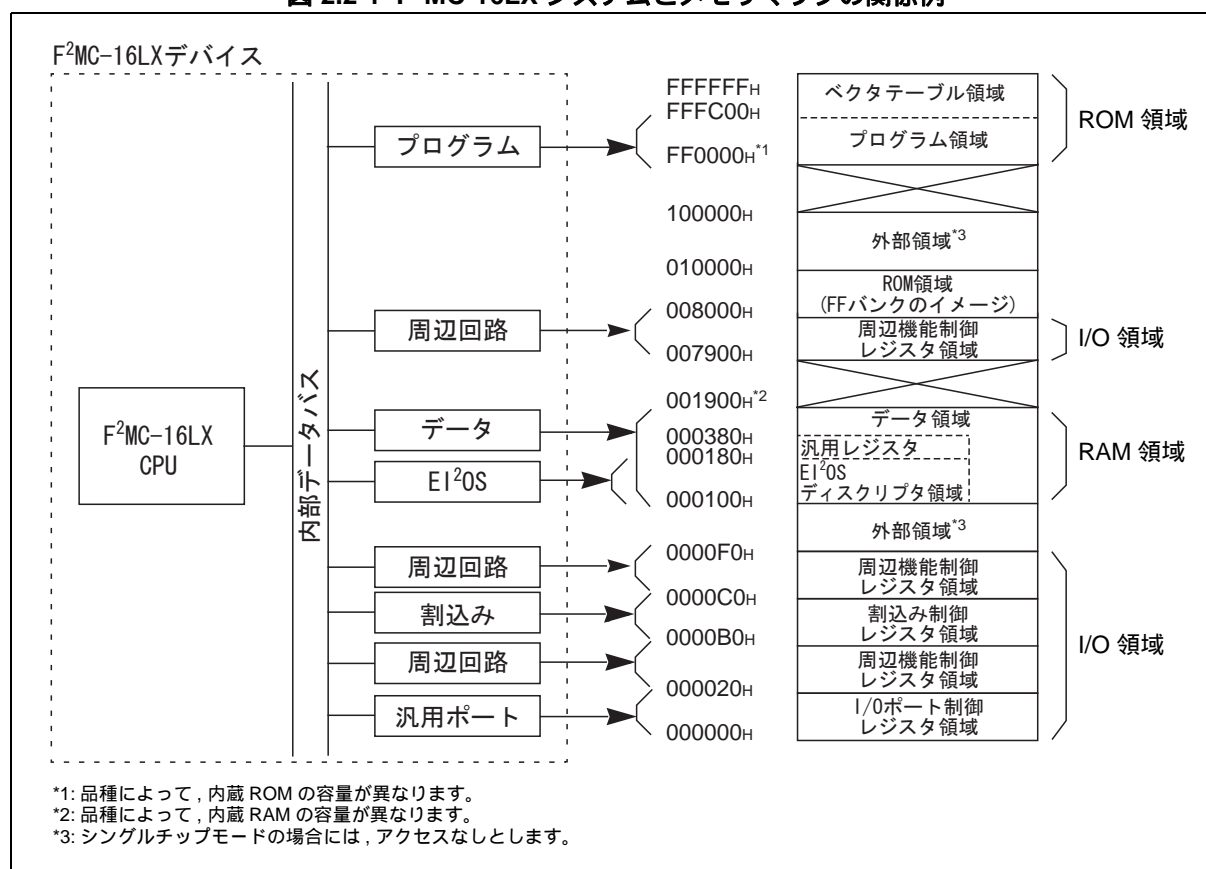
F²MC-16LX CPU は、16M バイトのメモリ空間を持ちます。

■ CPU メモリ空間の概要

F²MC-16LX CPU が管理するデータ・プログラム・I/O はすべて F²MC-16LX CPU の 16M バイトのメモリ空間のいずれかに配置されます。CPU は 24 ビットのアドレスバスでこれらのアドレスを示すことにより、各リソースにアクセスできます。

図 2.2-1 に、F²MC-16LX システムとメモリマップの関係例を示します。

図 2.2-1 F²MC-16LX システムとメモリマップの関係例



■ ROM 領域

- ベクタテーブル領域 (アドレス : FFFC00_H ~ FFFFFFF_H)
 - ・ リセットと割込みおよび CALLV ベクタに対応したベクタテーブルです。
 - ・ ROM 領域の最上位に割り当てられており、それぞれのベクタテーブルのアドレスに、対応する処理ルーチンの開始アドレスをデータとして設定します。
- プログラム領域 (アドレス : ~ FFFBFF_H)
 - ・ 内部プログラム領域として ROM が内蔵されています。
 - ・ 内部 ROM 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 : 000100_H ~ 0018FF_H (6 K バイトの場合))
 - ・ 内部データ領域として、スタティック RAM が内蔵されています。
 - ・ 内部 RAM 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 : 000180_H ~ 00037F_H)
 - ・ 8 ビット、16 ビット、32 ビットの演算や転送に使用する補助的なレジスタを配置します。
 - ・ RAM 領域の一部に割り当てられており、通常の RAM として使用することもできます。
 - ・ 汎用レジスタとして使用すると、汎用レジスタアドレッシングによって短い命令で高速にアクセスできます。
- 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 0000100_H ~ 00017F_H)
 - ・ 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保持します。
 - ・ RAM 領域の一部に割り当てられており、通常の RAM として使用することもできます。

■ I/O 領域

- 割込み制御レジスタ領域 (アドレス : 0000B0_H ~ 0000BF_H)

割込み制御レジスタ (ICR00 ~ ICR15) は、割込み機能を持つすべての周辺機能に対応し、割込みレベルの設定、および拡張インテリジェント I/O サービス (EI²OS) の制御を行います。
- 周辺機能制御レジスタ領域
(アドレス : 000020_H ~ 0000AF_H, 0000C0_H ~ 0000EF_H, 007900_H ~ 007FFF_H)

周辺機能およびデータの入出力を制御します。
- I/O ポート制御レジスタ領域 (アドレス : 000000_H ~ 00001F_H)

I/O ポートおよびデータの入出力を制御します。

■ アドレス作成の方式

F²MC-16LX のアドレス指定には、以下に示す 2 つの方式があります。

- リニア方式

24 ビットアドレスを命令により指定する方式

- バンク方式

アドレス上位 8 ビットを用途に応じたバンクレジスタで、アドレス下位 16 ビットを命令により指定する方式

2.3 リニア方式によるアドレス指定

リニア方式には以下に示す 2 つの方式があります。

- 24 ビットオペランド指定.....オペランドにて直接 24 ビットのアドレスを指定する方式
- 32 ビットレジスタ間接指定.....32 ビットの汎用レジスタの内容の下位 24 ビットをアドレスとして使用する方式

■ 24 ビットオペランド指定

図 2.3-1 にリニア方式 (24 ビットオペランド指定) の例を、図 2.3-2 にリニア方式 (32 ビットレジスタ間接指定) の例を示します。

図 2.3-1 リニア方式 (24 ビットオペランド指定) の例

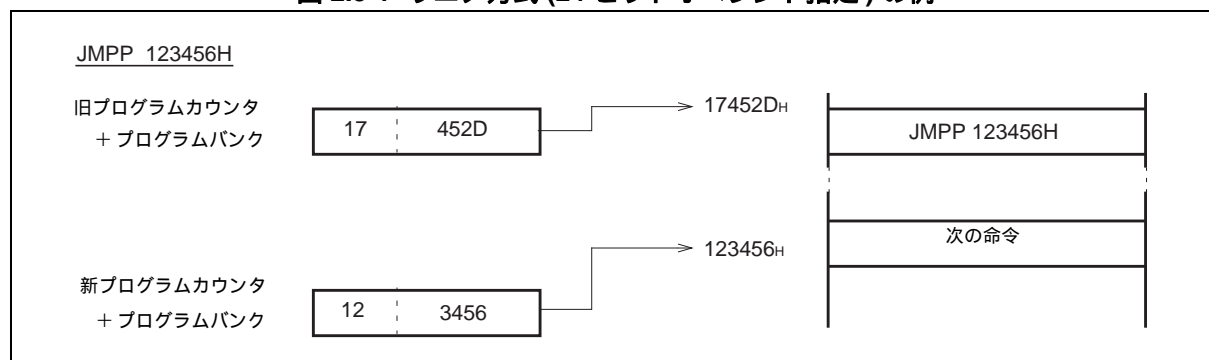
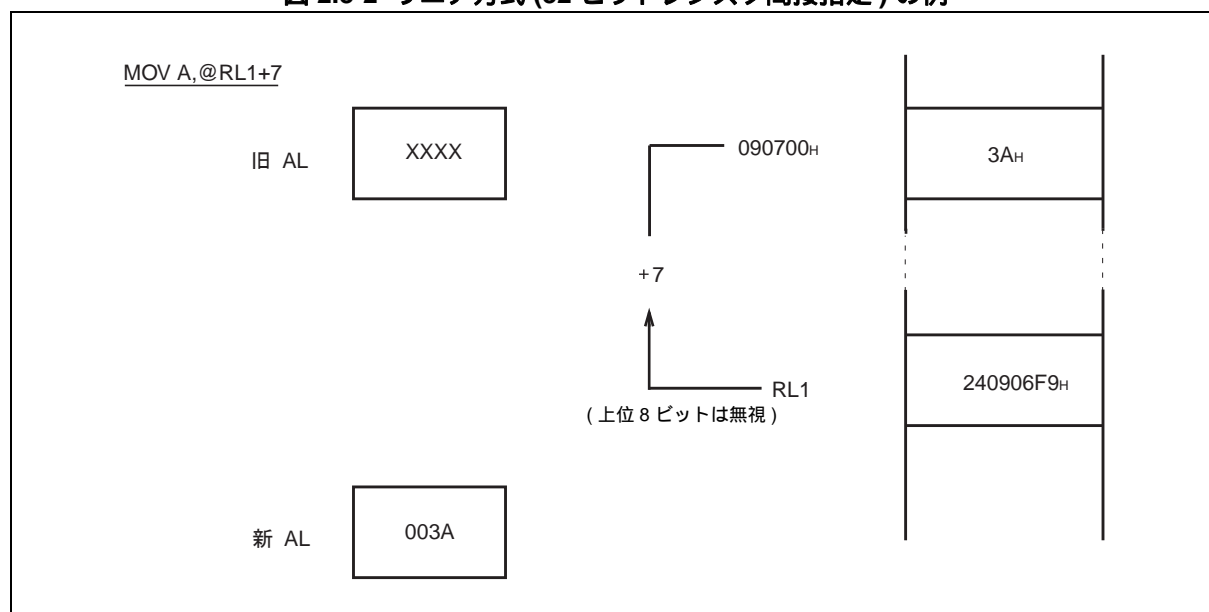


図 2.3-2 リニア方式 (32 ビットレジスタ間接指定) の例



2.4 バンク方式によるアドレス指定

バンク方式は 16M バイトの空間を 64 K バイトごとの 256 個のバンクに分割し、以下に示す 5 つのバンクレジスタで各空間に対応するバンクを指定します。

- プログラムカウンタバンクレジスタ (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)
- アディショナルデータバンクレジスタ (ADB)

■ バンク方式によるアドレス指定

● プログラムカウンタバンクレジスタ (PCB) による指定

プログラムカウンタバンクレジスタ (PCB) によって指定される 64 K バイトのバンクをプログラム (PC) 空間とよび、主として命令コードやベクタテーブル、即値データなどが存在します。

● データバンクレジスタ (DTB) による指定

データバンクレジスタ (DTB) によって指定される 64 K バイトのバンクをデータ (DT) 空間とよび、主として読み書き可能なデータや内外リソースの制御/データレジスタなどが存在します。

● 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) による指定

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) によって指定される 64 K バイトのバンクをスタック (SP) 空間とよび、プッシュ/ポップ命令や割込みのレジスタ退避などのときにスタックアクセスが発生したときにアクセスされる領域です。どちらの空間が使用されるかはコンディションコードレジスタ中の S フラグの値に依存します。

● アディショナルデータバンクレジスタ (ADB) による指定

アディショナルデータバンクレジスタ (ADB) によって指定される 64 K バイトのバンクをアディショナル (AD) 空間とよび、主として DT 空間に入りきらなかったデータなどが存在します。

表 2.4-1 に示すように、命令のコード効率を向上させるために各アドレッシングモードで使用されるデフォルト空間が事前に決められています。また、あるアドレッシングを使用したときにデフォルト以外の空間を使用したいときは、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指定することにより、そのプリフィックスコードに対応した任意のバンク空間をアクセスできます。

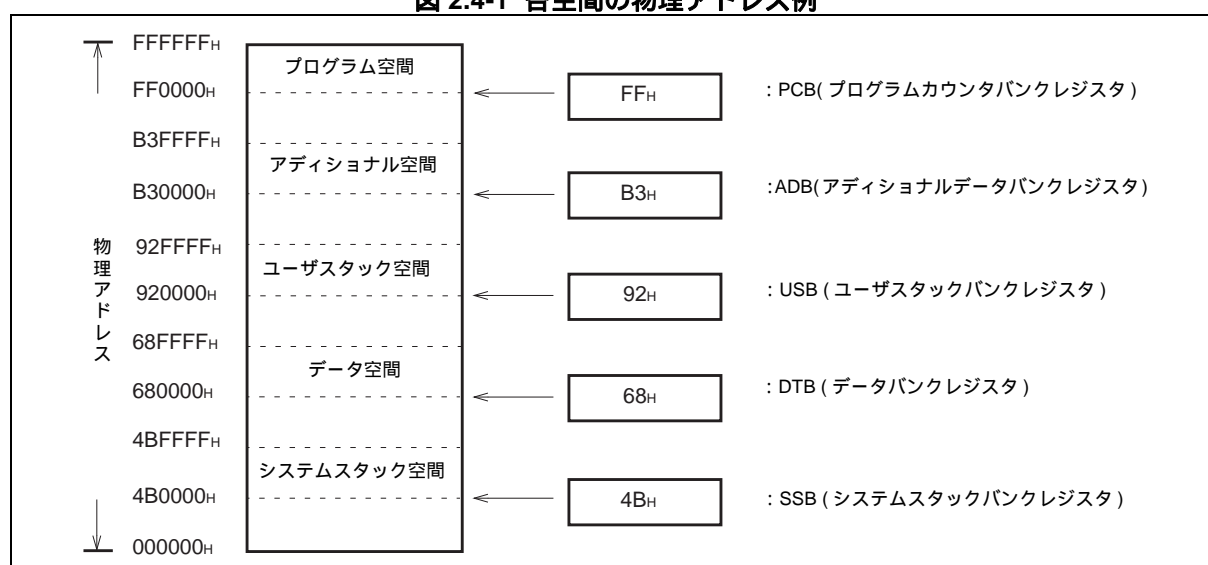
DTB, USB, SSB, ADB はリセットにより "00_H" に初期化され、PCB はリセットベクタにより指定された値に初期化されます。リセット後、DT, SP, AD の各空間はバンク "00_H" (000000_H ~ 00FFFF_H) に配置され、PC 空間はリセットベクタにより指定されたバンクに配置されます。

表 2.4-1 デフォルト空間

デフォルト空間	アドレッシング
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 を使用したアドレッシング , @A,addr16,dir
スタック空間	PUSHW, POPW,@RW3,@RW7 を使用したアドレッシング
アディショナル空間	@RW2,@RW6 を使用したアドレッシング

図 2.4-1 に , レジスタバンクに分割されたメモリ空間の物理アドレス例を示します。

図 2.4-1 各空間の物理アドレス例



2.5 メモリ空間における多バイト長データ

多バイト長のデータは、データの LSB 側のバイトから MSB 側のバイトの順番で、メモリ空間の下位アドレスから上位アドレスに配置されます。

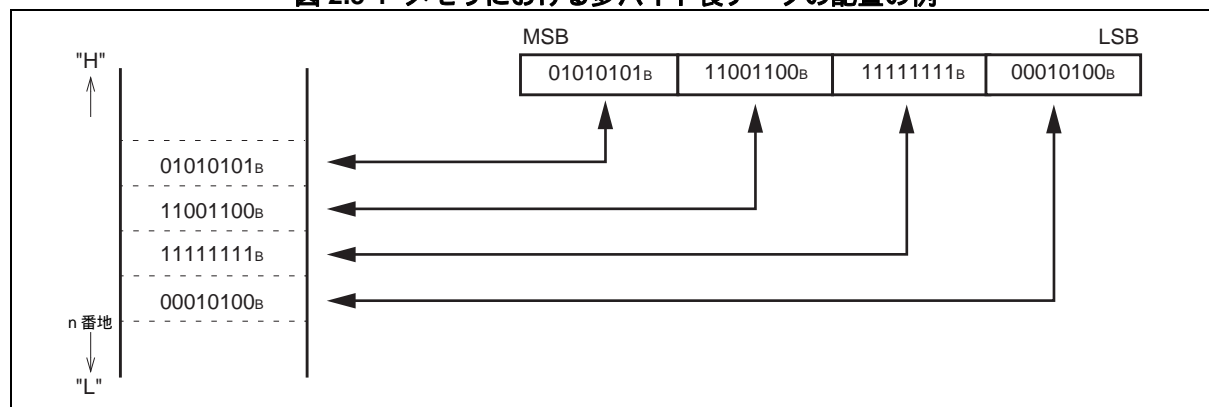
■ メモリ空間における多バイト長データの配置

メモリへの書込みはアドレスの低い方から順に行われます。したがって、32 ビット長データの場合には下位 16 ビットが先に転送され、続いて上位 16 ビットが転送されます。

なお、下位データの書込み直後にリセット信号を入力すると上位データが書き込まれないことがあります。

図 2.5-1 に、メモリにおける多バイト長データの構成を示します。データは下位 8 ビットが n 番地に、以下 $n+1$ 番地、 $n+2$ 番地、 $n+3$ 番地・・・の順に配置されます。

図 2.5-1 メモリにおける多バイト長データの配置の例

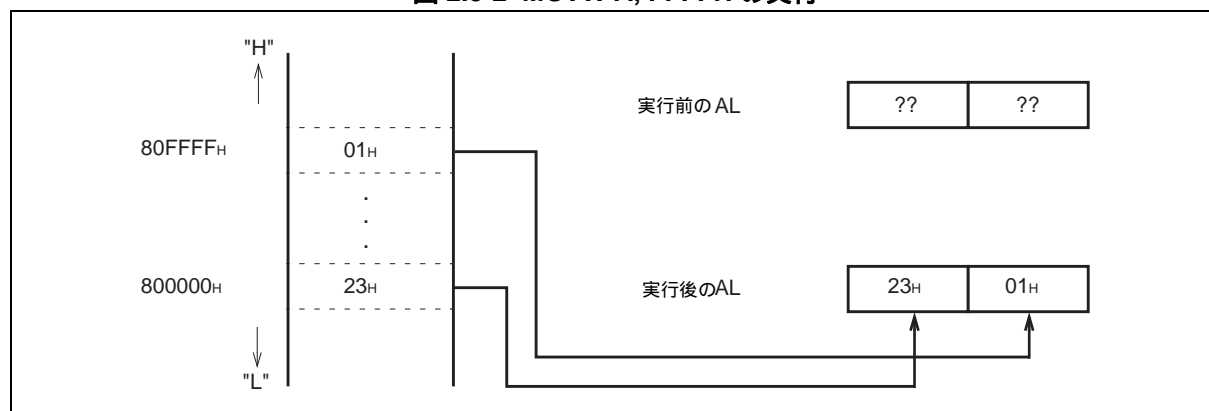


■ 多バイト長データのアクセス

アクセスはすべてバンク内を基本に行われますので、多バイト長のデータをアクセスする命令では、 FFFFH 番地の次のアドレスは同じバンクの 0000H 番地になります。

図 2.5-2 に、多バイト長データのアクセス命令の例を示します。

図 2.5-2 MOVW A, FFFFH の実行



2.6 レジスタ

F²MC-16LX のレジスタは大別して、専用レジスタと汎用レジスタの2種類に分けることができます。

専用レジスタは、CPU の内部に専用ハードウェアとして存在し、使用する用途が CPU のアーキテクチャ上で限定されています。

汎用レジスタは、通常のメモリと同じく、使用する用途をユーザが指定することができます。CPU のアドレス空間上に RAM と共存し、アドレスを指定しないでアクセスできるという点では専用レジスタと同じです。

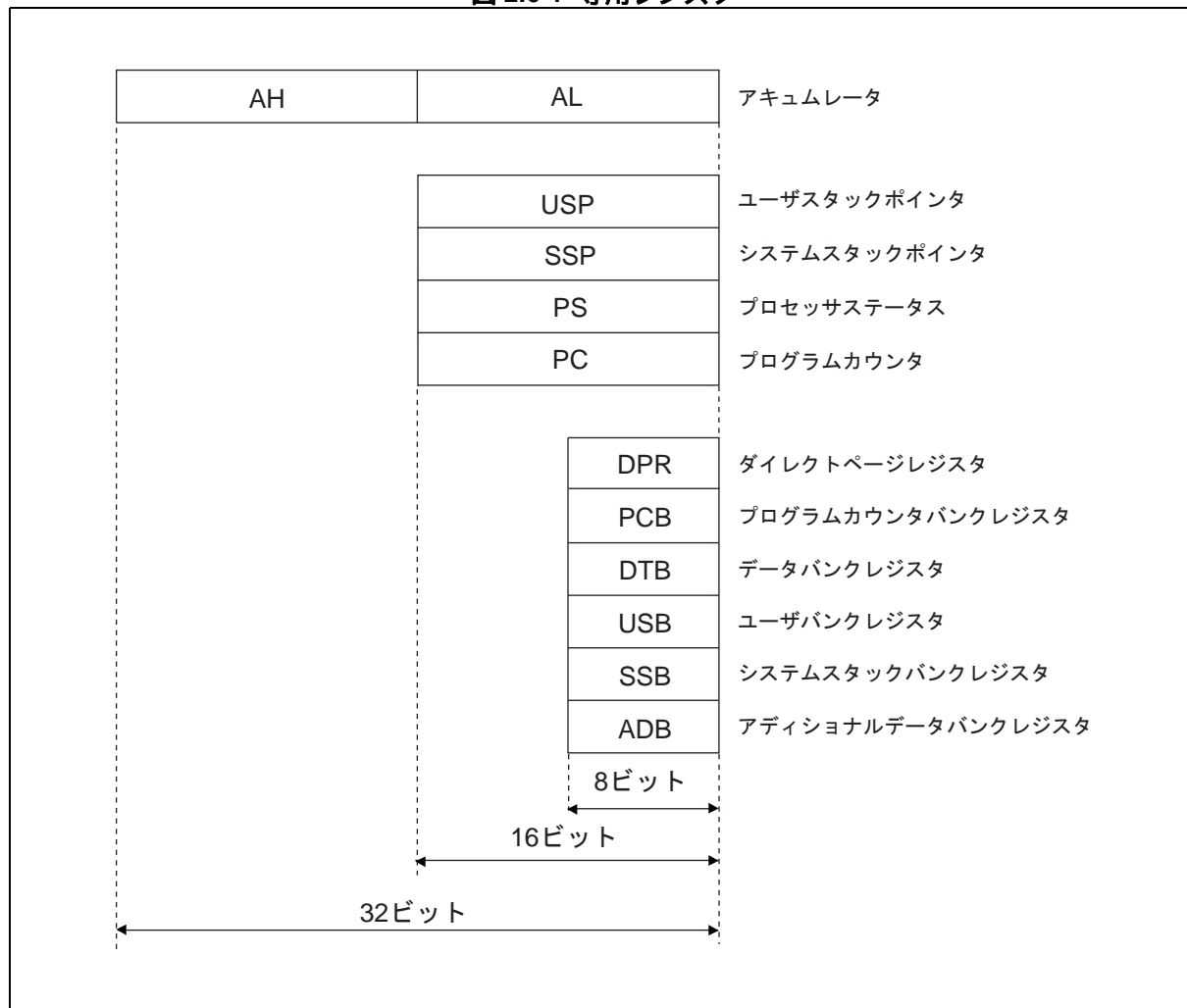
■ 専用レジスタ

F²MC-16LX CPU コアの専用レジスタは以下に示す 11 本があります。

- アキュムレータ (A=AH:AL)..... 16 ビット× 2 本のアキュムレータ
(合計32ビットのアキュムレータとしても使用可能)
- ユーザスタックポインタ (USP)..... ユーザスタック領域を示す 16 ビットのポインタ
- システムスタックポインタ (SSP) システムスタック領域を示す16ビットのポインタ
- プロセッサステータス (PS)..... システムの状態を示す 16 ビットのレジスタ
- プログラムカウンタ (PC)..... プログラムが格納されているアドレスを持つ 16 ビットのレジスタ
- プログラムカウンタバンクレジスタ (PCB) PC 空間を示す 8 ビットのレジスタ
- データバンクレジスタ (DTB)..... DT 空間を示す 8 ビットのレジスタ
- ユーザスタックバンクレジスタ (USB) ユーザスタック空間を示す8ビットのレジスタ
- システムスタックバンクレジスタ (SSB)..... システムスタック空間を示す8ビットのレジスタ
- アディショナルデータバンクレジスタ (ADB)..... AD 空間を示す 8 ビットのレジスタ
- ダイレクトページレジスタ (DPR) ダイレクトページを示す8ビットのレジスタ

図 2.6-1 に、専用レジスタの構成を示します。

図 2.6-1 専用レジスタ

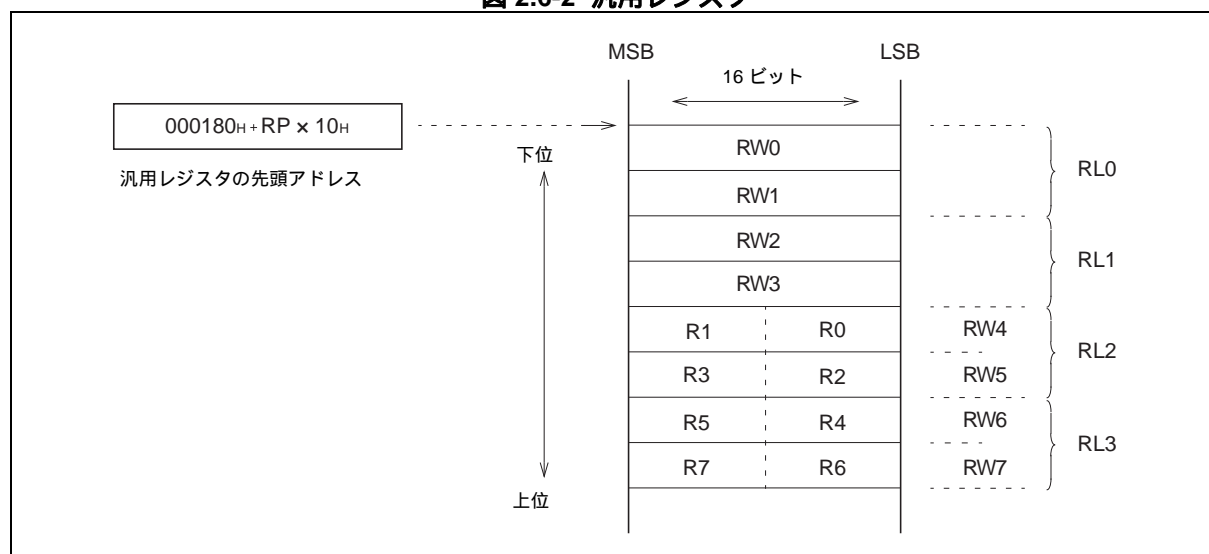


■ 汎用レジスタ

図 2.6-2 に示すように、F²MC-16LX の汎用レジスタは主記憶の 000180_H ~ 00037F_H (最大の場合) に存在し、レジスタバンクポインタ (RP) によって先に述べたアドレスのどの部分が現在使用中のレジスタバンクであるかを指定します。各バンクには以下に示す 3 種類のレジスタが存在します。これらは独立ではなく、以下に示すような関係があります。

- R0 ~ R7 :8 ビットの汎用レジスタ
- RW0 ~ RW7 :16 ビットの汎用レジスタ
- RL0 ~ RL3 :32 ビットの汎用レジスタ

図 2.6-2 汎用レジスタ



バイトレジスタとワードレジスタの上位 / 下位バイトの関係は、

$$RW_{(i+4)} = R_{(i \times 2 + 1)} \times 256 + R_{(i \times 2)} \quad [i=0 \sim 3]$$

という式で表すことができ、RLi の上位・下位と RW の関係は

$$RL_{(i)} = RW_{(i \times 2 + 1)} \times 65536 + RW_{(i \times 2)} \quad [i=0 \sim 3]$$

という式で表すことができます。

2.6.1 アキュムレータ (A)

アキュムレータ (A) レジスタは、2 つの 16 ビット長の演算用レジスタ AH および AL で構成され、演算などを行ったときの結果やデータ転送の一時記憶などに使用されます。

■ アキュムレータ (A)

アキュムレータ (A) レジスタでは、図 2.6-3 に示すように、32 ビットデータ処理時は AH と AL を連結して使用します。また、図 2.6-4 に示すように、16 ビットデータ処理のワード処理や 8 ビットデータ処理のバイト処理のときは AL のみが使用されます。A レジスタ中のデータはメモリ / レジスタ (R_i, R_{Wi}, R_{Li}) 中のデータと各種演算ができ、F²MC-8L と同様、F²MC-16LX でも基本的にワード長以下のデータを AL へ転送すると、転送前の AL 中のデータが自動的に AH に転送されます (データ保持機能)。したがって、データ保持機能と AL-AH 間演算において各種処理効率を上げることが可能になります。

AL へのバイト長以下のデータの転送時は、データは符号拡張またはゼロ拡張され 16 ビット長となり AL へ格納されます。AL 中のデータは、ワード長としてもバイト長としても扱えます。AL にバイト処理の算術演算命令を実行すると、演算前の AL の上位 8 ビットは無視されて演算結果の上位 8 ビットがすべて "0" になります。A レジスタは、リセットでは初期化されず、リセット直後は不定値になります。

図 2.6-3 32 ビットデータ転送例

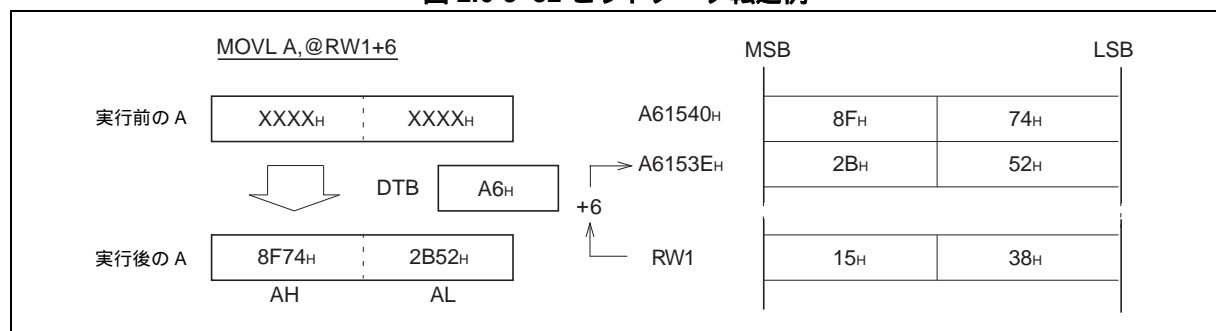
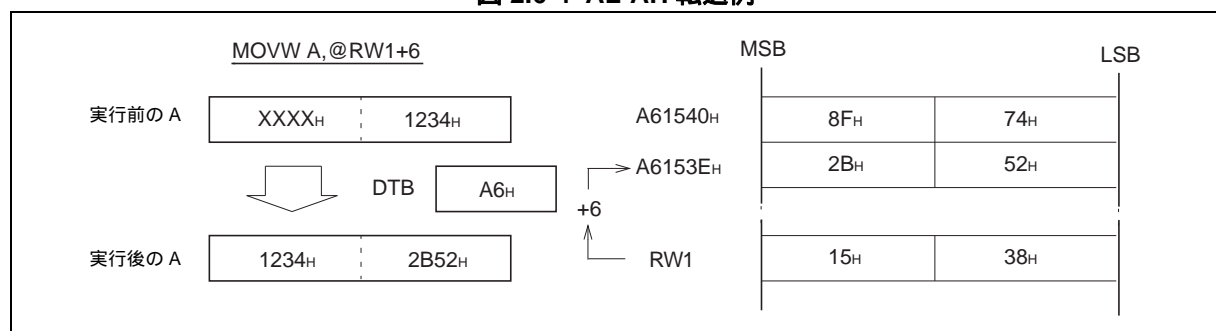


図 2.6-4 AL-AH 転送例



2.6.2 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)

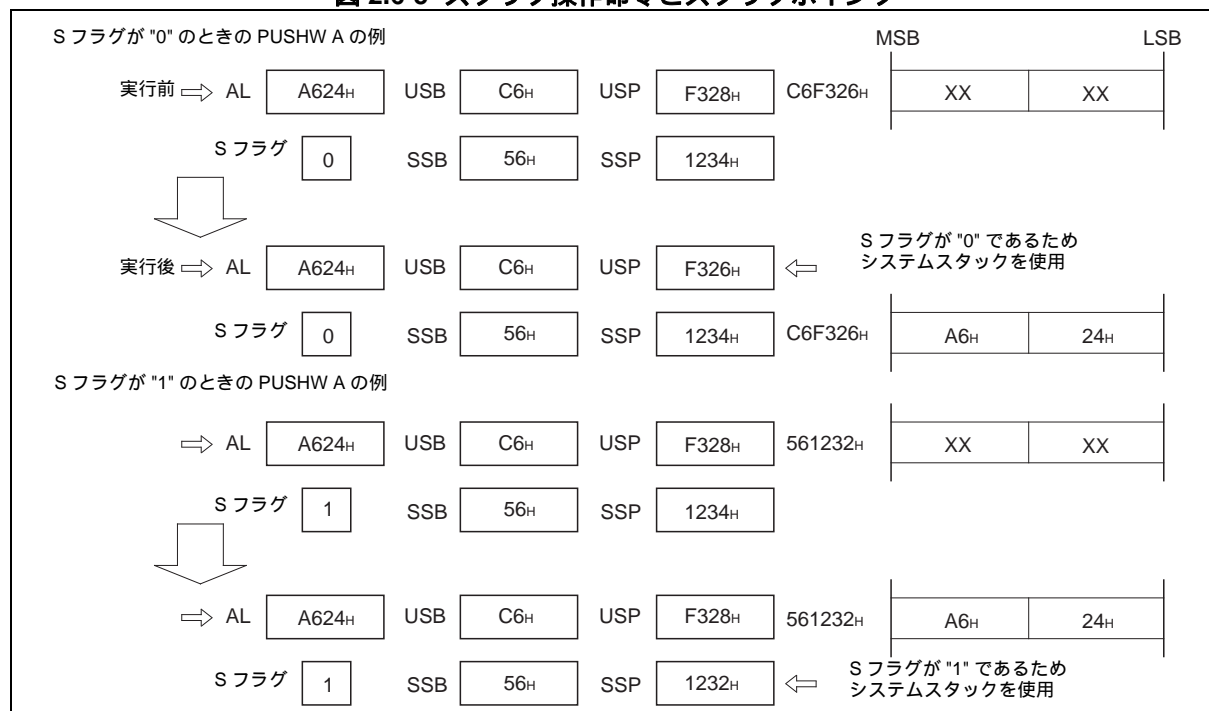
ユーザスタックポインタ (USP) およびシステムスタックポインタ (SSP) は、16 ビットのレジスタであり、プッシュ / ポップ命令およびサブルーチン実行時のデータ退避 / 復帰のメモリアドレスを示します。

■ 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)

図 2.6-5 に示すように、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) は、スタック系の命令により使用されますが、プロセッサステータス中の S フラグが "0" のときは USP レジスタが有効になり、S フラグが "1" のときは SSP レジスタが有効になります。また、割込みが受け付けられると S フラグがセットされるため、割込み時のレジスタ退避は必ず SSP の示すメモリ中に行われます。割込みルーチンでのスタック処理は SSP で、割込みルーチン以外のスタック処理には USP が使用されます。スタック空間を分ける必要のない場合には SSP だけを使用してください。

スタック時のアドレスの上位 8 ビットは、SSP SSB, USP USB により示されます。また、USP および SSP は、リセットでは初期化されず、不定値になります。

図 2.6-5 スタック操作命令とスタックポインタ



< 注意事項 >

スタックポインタに設定する値は、原則として偶数アドレスを使用してください。

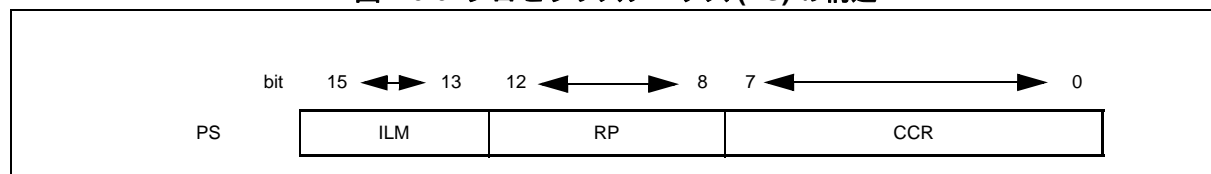
2.6.3 プロセッサステータス (PS)

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

■ プロセッサステータス (PS)

図 2.6-6 に示すように、プロセッサステータス (PS) の上位バイトはレジスタバンクの先頭アドレスを示す割り込みレベルマスクレジスタ (ILM) およびレジスタバンクポインタ (RP) より構成されます。PS の下位バイトは命令実行結果および割り込み発生などによりセット / リセットされる各種フラグで構成されているコンディションコードレジスタ (CCR) となります。

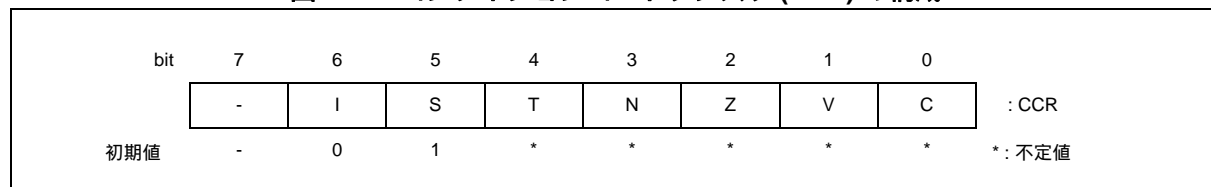
図 2.6-6 プロセッサステータス (PS) の構造



■ コンディションコードレジスタ (CCR)

図 2.6-7 に、コンディションコードレジスタ (CCR) の構造を示します。

図 2.6-7 コンディションコードレジスタ (CCR) の構成



● 割り込み許可フラグ (I)

ソフトウェア割り込み以外のすべての割り込み要求に対して I が "1" のときには割り込みが許可され、"0" のときには割り込みが禁止されます。リセットによって "0" にクリアされます。

● スタックフラグ (S)

S が "0" のときはスタック操作ポインタとして USP が有効、"1" のときは SSP が有効になります。割り込み受け時およびリセット時には "1" にセットされます。

● ステッキビットフラグ (T)

論理右/算術右シフト命令を実行後にキャリによってシフトアウトされたデータに 1 つ以上 "1" がある場合には "1"、それ以外は "0" となります。シフト量がゼロの場合も "0" となります。

● ネガティブフラグ (N)

演算結果の MSB が "1" の場合には N フラグは "1" にセットされ、それ以外は "0" にクリアされます。

● ゼロフラグ (Z)

演算結果がすべて "0" の場合には "1" にセットされ、それ以外は "0" にクリアされます。

● オーバフローフラグ (V)

演算の実行により符号付き数値としてオーバフローが発生すると "1" にセットされ、発生しないと "0" にクリアされます。

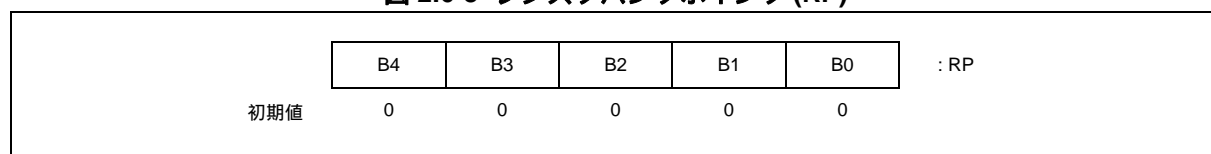
● キャリフラグ (C)

演算の実行により MSB より桁上り / 桁下りが発生すると "1" にセットされ、発生しないと "0" にクリアされます。

■ レジスタバンクポインタ (RP)

図 2.6-8 に示すように、レジスタバンクポインタ (RP) は、F²MC-16LX の汎用レジスタとそれが存在する内部 RAM のアドレスとの関係を示すレジスタで、現在使用中のレジスタバンクの先頭のメモリアドレスを $[000180_H + (RP) \times 10_H]$ という変換式で示します。RP は 5 ビットにより構成されており "00_H" ~ "1F_H" までの値をとることができ、000180_H ~ 00037F_H のメモリ中にレジスタバンクを配置できます。ただし、この範囲内であっても内部 RAM 以外の場合には汎用レジスタとして使用することはできません。RP はリセットによりすべて "0" に初期化されます。命令上では RP に 8 ビットの即値を転送できますが、実際に使用されるのはそのデータの下位 5 ビットのみです。

図 2.6-8 レジスタバンクポインタ (RP)



■ 割込みレベルマスクレジスタ (ILM)

図 2.6-9 に示すように、割込みレベルマスクレジスタ (ILM) は 3 ビットから構成されており、CPU の割込みマスクのレベルを示します。この 3 ビットにより示されるレベルより強いレベルの割込み要求のみが受け付けられます。図 2.6-1 に示すように、レベルの強弱は "0" が最強で、"7" が最弱と定義されています。したがって、割込みが受け付けられるためには、現状の ILM の保持値より小さい値の要求でなければなりません。割込みが受け付けられるとその割込みのレベル値が ILM に設定され、これ以降の同じかそれより低い優先順位の割込みは受け付けられなくなります。ILM はリセットによりすべて "0" に初期化されます。命令上では ILM に 8 ビットの即値を転送できますが、実際に使用されるのはそのデータの下位 3 ビットのみです。

図 2.6-9 割込みレベルマスクレジスタ (ILM)

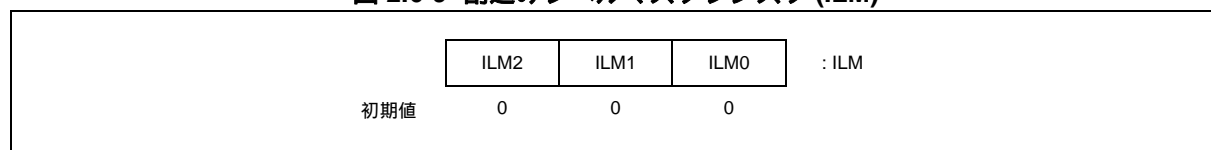


表 2.6-1 割込みレベルマスクレジスタ (ILM) で示されるレベルの強弱

ILM2	ILM1	ILM0	レベル値	許可される割込みレベル
0	0	0	0	割込み禁止
0	0	1	1	1 より小さい値のレベル (0 のみ)
0	1	0	2	2 より小さい値のレベル (0, 1)
0	1	1	3	3 より小さい値のレベル (0, 1, 2)
1	0	0	4	4 より小さい値のレベル (0, 1, 2, 3)
1	0	1	5	5 より小さい値のレベル (0, 1, 2, 3, 4)
1	1	0	6	6 より小さい値のレベル (0, 1, 2, 3, 4, 5)
1	1	1	7	7 より小さい値のレベル (0, 1, 2, 3, 4, 5, 6)

2.6.4 プログラムカウンタ (PC)

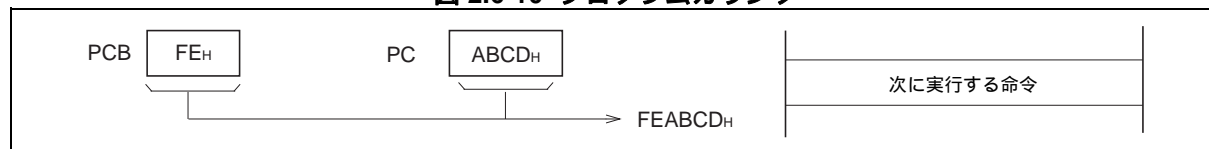
プログラムカウンタ (PC) は、CPU が実行すべき命令コードのメモリアドレスの下位 16 ビットを示します。

■ プログラムカウンタ (PC)

プログラムカウンタ (PC) は、16 ビットのカウンタであり、CPU が実行すべき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。上位 8 ビットアドレスは PCB により示されます。PC は条件分岐命令、サブルーチンコール命令、割込み、リセットなどにより内容が更新されます。また、オペランドアクセス時のベースポインタとして使用することもできます。

図 2.6-10 に、プログラムカウンタを示します。

図 2.6-10 プログラムカウンタ



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

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、アディショナル空間の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)

バンクレジスタには、以下に示す 5 つのレジスタがあります。

- プログラムカウンタバンクレジスタ (PCB) < 初期値 : リセットベクタ中の値 >
- データバンクレジスタ (DTB) < 初期値 : 00_H >
- ユーザスタックバンクレジスタ (USB) < 初期値 : 00_H >
- システムスタックバンクレジスタ (SSB) < 初期値 : 00_H >
- アディショナルデータバンクレジスタ (ADB) < 初期値 : 00_H >

各バンクレジスタは、PC, DT, SP(ユーザ), SP(システム), AD の各空間が配置されるメモリバンクを示します。

すべてのバンクレジスタは 1 バイト長であり、リセットにより PCB は "00_H" に初期化されます。PCB 以外のバンクレジスタは読出しのみができます。PCB も読出しできますが、書込みはできません。

16M バイト全空間に分岐する JMPP, CALLP, RETP, RETI, RETF 命令実行中、または割り込み発生時に PCB は更新されます。各レジスタの動作は、「2.2 メモリ空間」を参照してください。

2.6.6 ダイレクトページレジスタ (DPR)

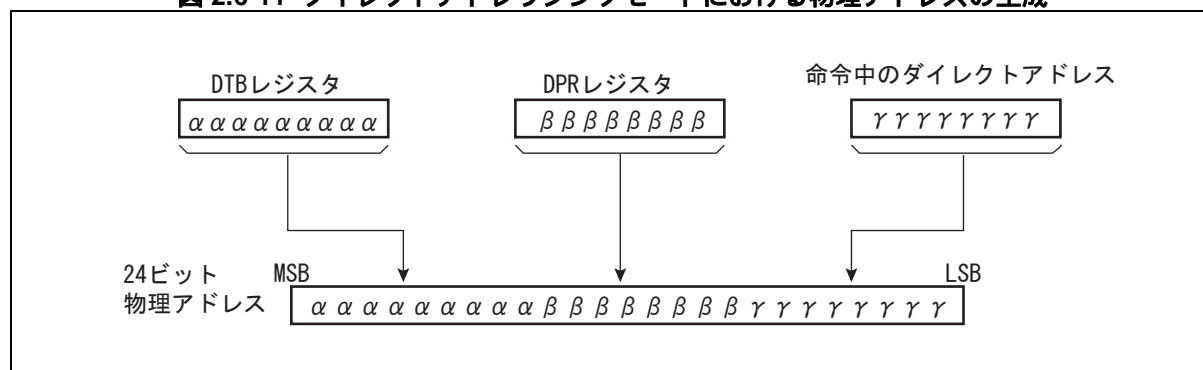
ダイレクトページレジスタ (DPR) は、ダイレクトアドレッシングモードのアドレス 8 ~ アドレス 15 を指定します。

■ ダイレクトページレジスタ (DPR) < 初期値 :01_H>

ダイレクトページレジスタ (DPR) は、図 2.6-11 に示すように、ダイレクトアドレッシングモードでの命令オペランドのアドレス 8 ~ アドレス 15 を指定します。DPR は 8 ビット長であり、リセットにより "01_H" に初期化されます。また、命令により読出し / 書込みができます。

図 2.6-11 に、ダイレクトアドレッシングモードにおける物理アドレスの生成を示します。

図 2.6-11 ダイレクトアドレッシングモードにおける物理アドレスの生成



2.7 レジスタバンク

レジスタバンクは8ワードで構成され、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) の汎用レジスタとして各種演算、ポインタとして各種命令に使用できます。(RL0 ~ RL3) は、メモリ空間の全空間を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。

■ レジスタバンク

表 2.7-1 に、各レジスタの機能を、表 2.7-2 に各レジスタの関係を示します。

レジスタバンクの内容は、通常の RAM 同様にリセットでは初期化されず、リセット前の状態が保持されます。ただし、パワーオン時は不定値になります。

表 2.7-1 各レジスタの機能

R0 ~ R7	各種命令のオペランドとして使用 (注意事項) R0 はパレルシフトのカウンタおよびノーマライズ命令のカウンタとしても使用
RW0 ~ RW7	ポインタ、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、各種命令のオペランドとして使用

表 2.7-2 各レジスタの関係

アドレス	バイトレジスタ	ワードレジスタ	ロングワードレジスタ
000180 _H + RP × 10 _H + 0		RW0	RL0
000180 _H + RP × 10 _H + 1		RW1	
000180 _H + RP × 10 _H + 2			RW2
000180 _H + RP × 10 _H + 3			
000180 _H + RP × 10 _H + 4			
000180 _H + RP × 10 _H + 5			
000180 _H + RP × 10 _H + 6		RW3	
000180 _H + RP × 10 _H + 7			
000180 _H + RP × 10 _H + 8	R0	RW4	RL2
000180 _H + RP × 10 _H + 9	R1		
000180 _H + RP × 10 _H + 10	R2	RW5	
000180 _H + RP × 10 _H + 11	R3		
000180 _H + RP × 10 _H + 12	R4	RW6	RL3
000180 _H + RP × 10 _H + 13	R5		
000180 _H + RP × 10 _H + 14	R6	RW7	
000180 _H + RP × 10 _H + 15	R7		

2.8 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置くことで、命令動作の一部を変更することができます。プリフィックスコードには、バンクセレクトプリフィックス、コモンレジスタバンクプリフィックス、フラグ変化抑止プリフィックスの3種類があります。

■ バンクセレクトプリフィックス

データアクセスのときに使用されるメモリ空間はアドレッシング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置くことで、命令によるデータアクセスのメモリ空間をアドレッシングモードとは無関係に任意に選択することができます。

表 2.8-1 に、バンクセレクトプリフィックスと選択されるメモリ空間を示します。

表 2.8-1 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	選択される空間
PCB	PC 空間
DTB	データ空間
ADB	AD 空間
SPB	そのときのスタックフラグの内容により SSP 空間、USP 空間のどちらかが使用されます。

ただし、以下に示す命令に対しては注意してください。

- スtring命令 [MOVS / MOVSW / SCEQ / SCWEQ / FILS / FILSW]

プリフィックスの有無にかかわらず、オペランド指定されたバンクレジスタを使用します。

- 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 フラグに応じて SSB または USB を使用します。

- I/O アクセス命令

MOV A,io	MOV io,A	MOVX A,io	MOVW A,io
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	WBTS	

プリフィックスの有無にかかわらず、バンクの I/O 空間を使用します。

- フラグ変更命令 [AND CCR,#imm8 / OR CCR,#imm8]

命令動作は正常ですが、プリフィックスの効果が次の命令まで及びます。

● POPW PS

プリフィックスの有無にかかわらず、S フラグに応じて SSB または USB を使用します。プリフィックスの効果が次の命令まで及びます。

● MOV ILM,#imm8

命令動作は正常ですが、プリフィックスの効果が次の命令まで及びます。

● RETI

プリフィックスの有無にかかわらず、SSB を使用します。

■ コモンレジスタバンクプリフィックス (CMR)

複数のタスク間でのデータ交換を容易にするには、RP がどのような値であっても同一のレジスタバンクを簡単にアクセスできる手段が必要です。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置くことで、現在の RP の値とは関係なくその命令のレジスタアクセスをすべて 000180_H ~ 00018F_H にあるコモンバンク (RP=0 のときに選択されるレジスタバンク) に変更することができます。ただし、以下に示す命令に対しては注意してください。

● スtring命令 [MOVS / MOVSW / SCEQ / SCWEQ / FILS / FILSW]

プリフィックスコードを付加したString命令実行中に割り込み要求が発生すると、割り込み復帰後のString命令に対しては、プリフィックスが無効であるため誤動作となります。上記のString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 / POPW PS]

命令動作は正常ですが、プリフィックスの効果が次の命令まで及びます。

● MOV ILM,#imm8

命令動作は正常ですが、プリフィックスの効果が次の命令まで及びます。

■ フラグ変化抑止プリフィックス (NCC)

フラグ変化を抑止するには、フラグ変化抑止プリフィックスコード (NCC) を使用します。不要フラグ変化を抑止する命令の前に置くことで、命令実行に伴うフラグ変化を抑止可能です。ただし、以下に示す命令に対しては注意してください。

● スtring命令 [MOVS / MOVSW / SCEQ / SCWEQ / FILS / FILSW]

プリフィックスコードを付加したString命令の実行中に割り込み要求が発生すると、割り込み復帰後のString命令に対してはプリフィックスが無効であるために誤動作となります。上記のString命令には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 / POPW PS]

命令動作は正常ですが、プリフィックスの効果が次の命令まで及びます。

● 割り込み命令 [INT #vct8 / INT9 / INT addr16 / INTP addr24 / RETI]

プリフィックスの有無にかかわらず、CCR は命令の仕様どおり変化します。

● JCTX @A

プリフィックスの有無にかかわらず,CCR は命令の仕様どおり変化します。

● MOV ILM,imm8

命令動作は正常ですが,プリフィックスの効果が次の命令まで及びます。

2.9 割り込み抑止命令

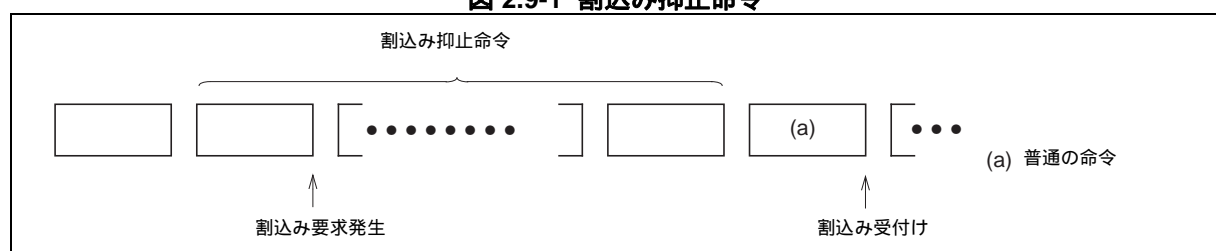
以下に示す 10 種類の命令については、割り込み要求が受け付けられません。

MOV ILM, #imm8	PCB	SPB	OR CCR, #imm8	NCC
AND CCR, #imm8	ADB	CMR	POPW PS	DTB

■ 割り込み抑止命令

図 2.9-1 に示すように、この命令実行中に有効なハードウェア割り込み要求が発生しても、割り込み処理が行われるのは、この命令以降にこの種類以外の命令が実行された後になります。

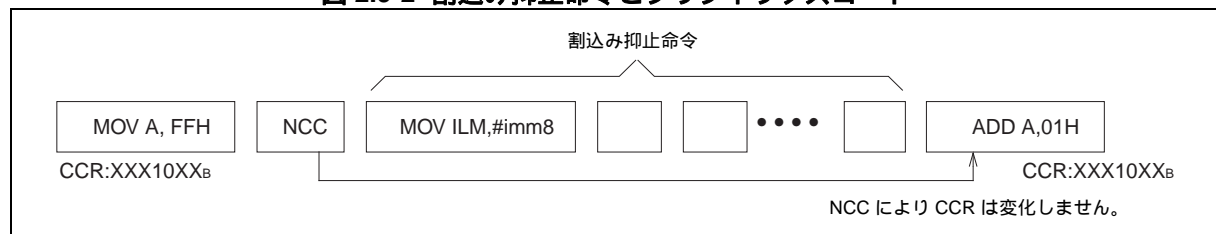
図 2.9-1 割り込み抑止命令



■ 割り込み抑止命令とプリフィックス命令に関する制約

図 2.9-2 に示すように、割り込み抑止命令の前にプリフィックスコードを付加した場合、プリフィックスコードの効果は、プリフィックスコード後の最初の〔割り込み抑止命令以外の命令〕まで及びます。

図 2.9-2 割り込み抑止命令とプリフィックスコード

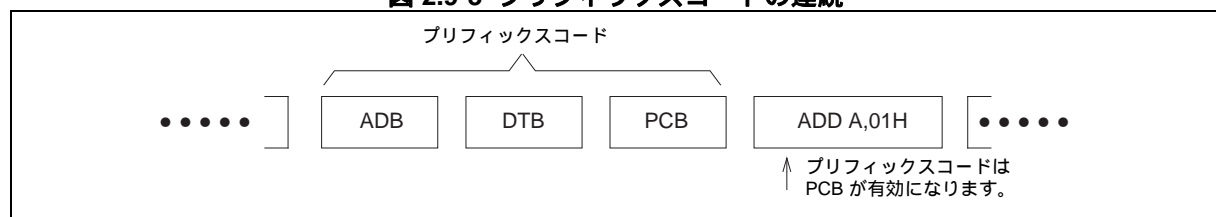


■ プリフィックスコードが連続している場合

図 2.9-3 に示すように、競合するプリフィックスコードが連続していた場合には後方のものが有効になります。

競合するプリフィックスコードとは PCB, ADB, DTB, SPB のことを意味します。

図 2.9-3 プリフィックスコードの連続



第3章

割込み

この章では、MB90330A シリーズの割込みと拡張インテリジェント I/O サービス (EI²OS), ダイレクトメモリアクセスコントローラ (μDMAC) について説明します。

- 3.1 割込みの概要
- 3.2 割込み要因と割込みベクタ
- 3.3 割込み制御レジスタと周辺機能
- 3.4 ハードウェア割込み
- 3.5 ソフトウェア割込み
- 3.6 拡張インテリジェント I/O サービス (EI²OS) による割込み
- 3.7 例外処理割込み
- 3.8 μDMAC による割込み
- 3.9 例外
- 3.10 割込み処理のスタック動作
- 3.11 割込み処理のプログラム例
- 3.12 遅延割込み発生モジュール

3.1 割込みの概要

F²MC-16LX にはイベントなどが発生したとき、現在実行中の処理を中断して、別に定義したプログラムへ制御を移す以下の5つの割込み機能があります。

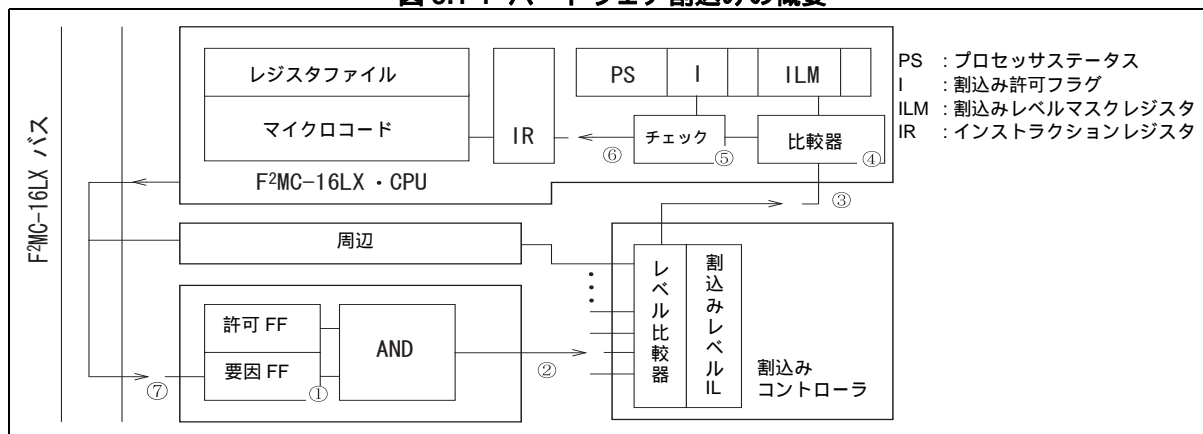
- ハードウェア割込み
- ソフトウェア割込み
- 拡張インテリジェント I/O サービス (EI²OS) による割込み
- μ DMAC による割込み
- 例外処理

■ 割込みの種類と機能

● ハードウェア割込み

周辺機能からの割込み要求に対して、ユーザの定義した割込み処理用プログラムへ制御を移行する機能です。

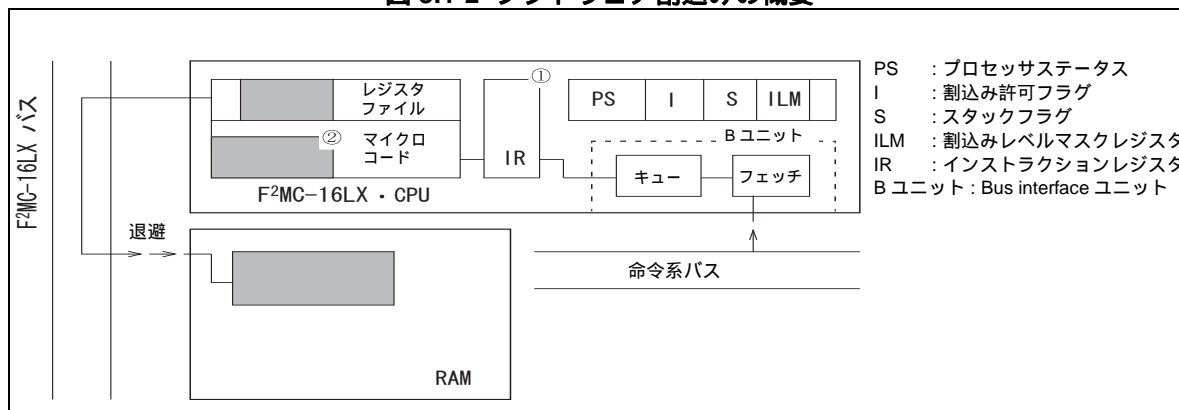
図 3.1-1 ハードウェア割込みの概要



● ソフトウェア割込み

ソフトウェア割込み専用の命令 (INT 命令など) の実行によって、ユーザの定義した割込み処理用プログラムへ制御を移行する機能です。

図 3.1-2 ソフトウェア割込みの概要

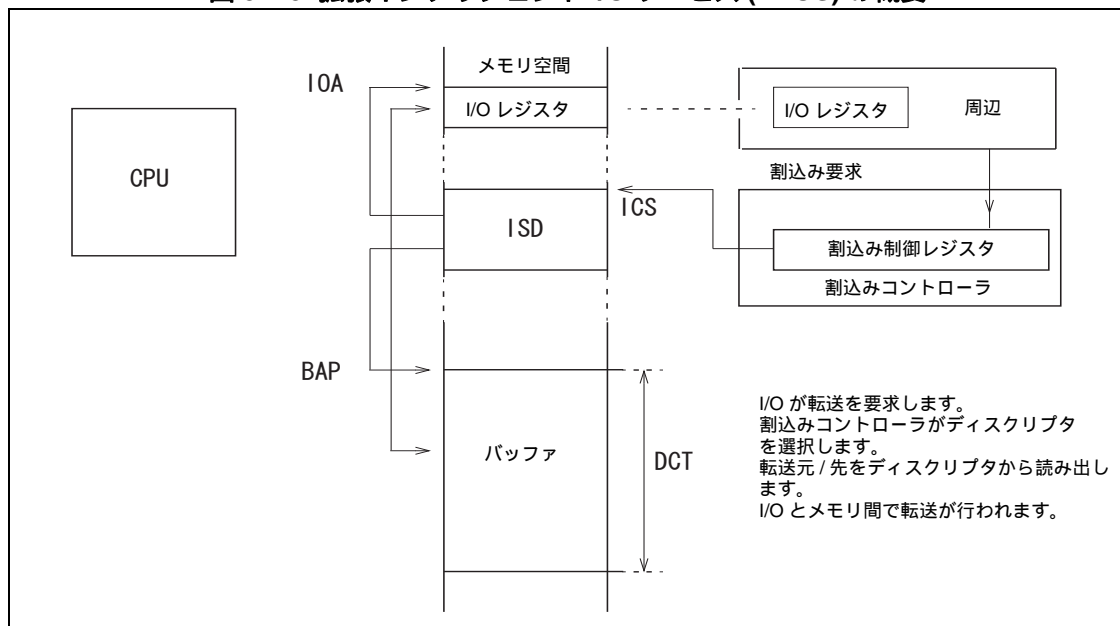


● 拡張インテリジェント I/O サービス (EI²OS) による割込み

El²OS は周辺機能とメモリ間の自動データ転送機能です。従来、割込み処理プログラムで行っていたデータ転送を DMA(ダイレクトメモリアクセス)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。

El²OS による割込みは、ハードウェア割込みの一種です。

図 3.1-3 拡張インテリジェント I/O サービス (EI²OS) の概要

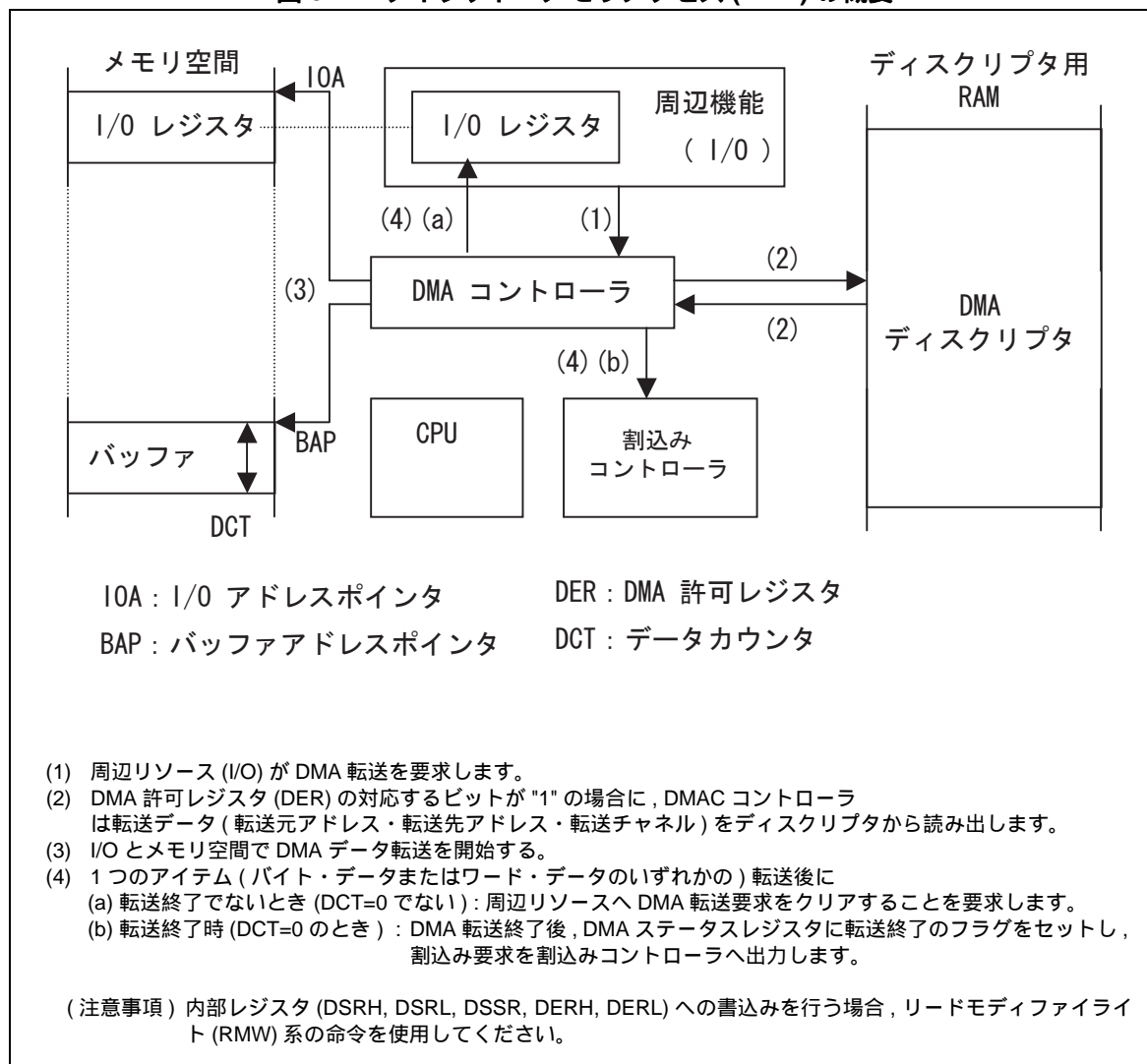


● μ DMAC による割込み

μDMAC は周辺機能とメモリ間の自動データ転送機能です。従来、割り込み処理プログラムで行っていたデータ転送を DMA 転送で行います。指定回数のデータ転送処理が終了すると、自動的に割り込み処理プログラムを実行します。

μDMAC による割込みはハードウェア割込みの一種です。

図 3.1-4 ダイレクト・メモリアクセス (DMA) の概要



● 例外処理

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

3.2 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、メモリの最上位アドレスに 256 組の割り込みベクタテーブルを割り当てています。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、一部の割り込みベクタはハードウェア割り込みや例外処理割り込みで共有されています。また、ハードウェア割り込みでは、各周辺機能に対し、割り込みベクタと割り込み制御レジスタ (ICR) が固定されています。

■ 割り込みベクタ

割り込み処理の際に参照す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 ~ FFFFFFF_H) に割り当てられています。また、割り込みベクタは、EI²OS, μ DMAC, ハードウェア割り込み、ソフトウェア割り込みおよび例外処理で同じ領域を共有しています。割り込み番号と割り込みベクタの割当てを表 3.2-1 に示します。

表 3.2-1 割り込みベクター一覧表

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 No	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	RESET 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 #0
INT12	FFFFCC _H	FFFFCD _H	FFFFCE _H	未使用	#12	ハードウェア 割り込み #1
INT13	FFFFC8 _H	FFFFC9 _H	FFFFCA _H	未使用	#13	ハードウェア 割り込み #2
INT14	FFFFC4 _H	FFFFC5 _H	FFFFC6 _H	未使用	#14	ハードウェア 割り込み #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 >

使われない割り込みベクタも、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 3.2-2 に、ソフトウェア割り込みを除く割り込み要因と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 3.2-2 割り込み要因と割り込みベクタ / 割り込み制御レジスタ (1 / 2)

割り込み要因	EI ² OS 対応	μDMAC 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		<div>高い</div> <div>↑</div>	
INT9 命令	×	×	#09	09 _H	FFFFD8 _H			
例外処理	×	×	#10	0A _H	FFFFD4 _H			
USB ファンクション 1	×	0, 1	#11	0B _H	FFFFD0 _H	ICR00		0000B0 _H *1
USB ファンクション 2	×	2 ~ 6 *3	#12	0C _H	FFFFCC _H			
USB ファンクション 3	×	×	#13	0D _H	FFFFC8 _H	ICR01		0000B1 _H *1
USB ファンクション 4	×	×	#14	0E _H	FFFFC4 _H			
USB ホスト 1	×	×	#15	0F _H	FFFFC0 _H	ICR02		0000B2 _H *1
USB ホスト 2	×	×	#16	10 _H	FFFFBC _H			
I ² C ch.0	×	×	#17	11 _H	FFFFB8 _H	ICR03		0000B3 _H *1
DTP/ 外部割り込み ch.0/ ch.1		×	#18	12 _H	FFFFB4 _H			
I ² C ch.1	×	×	#19	13 _H	FFFFB0 _H	ICR04		0000B4 _H *1
DTP/ 外部割り込み ch.2/ ch.3		×	#20	14 _H	FFFFAC _H			
I ² C ch.2	×	×	#21	15 _H	FFFFA8 _H	ICR05		0000B5 _H *1
DTP/ 外部割り込み ch.4/ ch.5		×	#22	16 _H	FFFFA4 _H			
PWC・リロードタイマ ch.0		14	#23	17 _H	FFFFA0 _H	ICR06		0000B6 _H *1
DTP/ 外部割り込み ch.6/ ch.7		×	#24	18 _H	FFFF9C _H			
インプットキャプチャ ch.0/ch.1		7	#25	19 _H	FFFF98 _H	ICR07	0000B7 _H *1	
リロードタイマ ch.1		×	#26	1A _H	FFFF94 _H			
インプットキャプチャ ch.2/ch.3		8	#27	1B _H	FFFF90 _H	ICR08	0000B8 _H *1	
リロードタイマ ch.2		×	#28	1C _H	FFFF8C _H			

表 3.2-2 割り込み要因と割り込みベクタ / 割り込み制御レジスタ (2 / 2)

割り込み要因	EI ² OS 対応	μDMAC 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	
アウトプットコンペア ch.0/ch.1		×	#29	1D _H	FFFF88 _H	ICR09	0000B9 _H ^{*1}	<div></div> <div>低い</div>
PPG ch.0/ch.1	×	×	#30	1E _H	FFFF84 _H			
アウトプットコンペア ch.2/ch.3		×	#31	1F _H	FFFF80 _H	ICR10	0000BA _H ^{*1}	
PPG ch.2/ch.3	×	×	#32	20 _H	FFFF7C _H			
UART 送信完了 ch.2/ch.3		11	#33	21 _H	FFFF78 _H	ICR11	0000BB _H ^{*1}	
PPG ch.4/ch.5	×	×	#34	22 _H	FFFF74 _H			
UART 受信完了 ch.2/ch.3		10	#35	23 _H	FFFF70 _H	ICR12	0000BC _H ^{*1}	
A/D 変換・フリーラン タイマ		15	#36	24 _H	FFFF6C _H			
UART 送信完了 ch.0/ch.1		13	#37	25 _H	FFFF68 _H	ICR13	0000BD _H ^{*1}	
拡張シリアル I/O	×	9	#38	26 _H	FFFF64 _H			
UART 受信完了 ch.0/ch.1		12	#39	27 _H	FFFF60 _H	ICR14	0000BE _H ^{*1}	
タイムベースタイマ・時計 タイマ	×	×	#40	28 _H	FFFF5C _H			
フラッシュメモリ書込み / 消去	×	×	#41	29 _H	FFFF58 _H	ICR15	0000BF _H ^{*1}	
遅延割り込み発生モジュール	×	×	#42	2A _H	FFFF54 _H			

: 使用可能 EI²OS 停止機能付 (割り込みクリア信号で、割り込み要求フラグはクリアされます。ストップ
要求あり)。

: 使用可能 (割り込みクリア信号で、割り込み要求フラグはクリアされます)。

: ICR を共有する割り込み要因を使用しない場合に使用可能

× : 使用不可

*1: ICR 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。

*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。

*3: ch.2, ch.3 は USB ホスト動作時でも使用できます。

< 注意事項 >

- 同一割込み番号に2つの割込み要因があった場合は、リソースは両方の割込み要求フラグが μ DMAC 割込みクリア信号でクリアされます。したがって、2 要因のどちらか1つが μ DMAC 機能を使用すると、もう1つの割込み機能は使用できません。該当リソースの割込み要求許可ビットを "0" にして、ソフトウェアポーリング処理で対処してください。
- 同一割込み制御レジスタ (ICR) に2つの割込み要因がある場合は EI²OS の使用を許可した場合、どちらかの割込み要因を検出すると EI²OS が起動されます。EI²OS 起動中は起動要因以外の割込みはマスク されますので、EI²OS 使用時はどちらかの割込み要求をマスクして使用されることを推奨致します。
- 同一割込み制御レジスタ (ICR) に2つの割込み要因があるリソースは、割込みフラグが EI²OS 割込みクリア信号でクリアされます。

■ USB 割込みの種類と機能

USB 割込み要因	詳細内容
USB ファンクション 1	End Point0-IN EndPoint0-OUT
USB ファンクション 2	End Point1 ~ 5*
USB ファンクション 3	SUSP SOF BRST WKUP CONF
USB ファンクション 4	SPK
USB ホスト 1	DIRQ CNNIRQ URIRQ RWKIRQ
USB ホスト 2	SOFIRQ CMPIRQ

* : End Point 1, 2 は、USB ホスト動作時でも使用できます。

3.3 割り込み制御レジスタと周辺機能

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周辺機能に対応して存在します。このレジスタは、割り込みと拡張インテリジェント I/O サービス (EI²OS) の制御を行います。

■ 割り込み制御レジスタ一覧

表 3.3-1 に、割り込み制御レジスタと対応する周辺機能の一覧を示します。

表 3.3-1 割り込み制御レジスタ一覧

アドレス	レジスタ	略称	対応する周辺機能
0000B0 _H	割り込み制御レジスタ 00	ICR00	USB ファンクション 1, USB ファンクション 2
0000B1 _H	割り込み制御レジスタ 01	ICR01	USB ファンクション 3, USB ファンクション 4
0000B2 _H	割り込み制御レジスタ 02	ICR02	USB ホスト 1, USB ホスト 2
0000B3 _H	割り込み制御レジスタ 03	ICR03	I ² C ch.0, DTP 外部割り込み ch.0/ch.1
0000B4 _H	割り込み制御レジスタ 04	ICR04	I ² C ch.1, DTP 外部割り込み ch.2/ch.3
0000B5 _H	割り込み制御レジスタ 05	ICR05	I ² C ch.2, DTP 外部割り込み ch.4/ch.5
0000B6 _H	割り込み制御レジスタ 06	ICR06	PWC, リロードタイマ ch.0, DTP 外部割り込み ch.6/ch.7
0000B7 _H	割り込み制御レジスタ 07	ICR07	インプットキャプチャ ch.0/ch.1, リロードタイマ ch.1
0000B8 _H	割り込み制御レジスタ 08	ICR08	インプットキャプチャ ch.2/ch.3, リロードタイマ ch.2
0000B9 _H	割り込み制御レジスタ 09	ICR09	アウトプットコンペア ch.01, PPGch.0/ch.1
0000BA _H	割り込み制御レジスタ 10	ICR10	アウトプットコンペア ch.2/ch.3, PPGch.2/ch.3
0000BB _H	割り込み制御レジスタ 11	ICR11	UART 送信 ch.2/ch.3, PPGch.4/ch.5
0000BC _H	割り込み制御レジスタ 12	ICR12	UART 受信 ch.2/ch.3, AD 変換, フリーランタイマ
0000BD _H	割り込み制御レジスタ 13	ICR13	UART 送信 ch.0/ch.1, 拡張シリアル I/O
0000BE _H	割り込み制御レジスタ 14	ICR14	UART受信ch.0/ch.1, タイマベースタイマ, 時計タイマ
0000BF _H	割り込み制御レジスタ 15	ICR15	フラッシュ書込み, 遅延割り込み発生モジュール

■ 割込み制御レジスタの機能

割込み制御レジスタ (ICR) には、それぞれ次に示す 4 つの機能があります。

- 対応する周辺機能の割込みレベルの設定
- 対応する周辺機能の割込みを通常割込みにするか、拡張インテリジェント I/O サービス (EI²OS) にするかを選択
- 拡張インテリジェント I/O サービス (EI²OS) のチャネルの選択
- 拡張インテリジェント I/O サービス (EI²OS) ステータスの表示

割込み制御レジスタ (ICR) は、図 3.3-1 と図 3.3-2 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 >

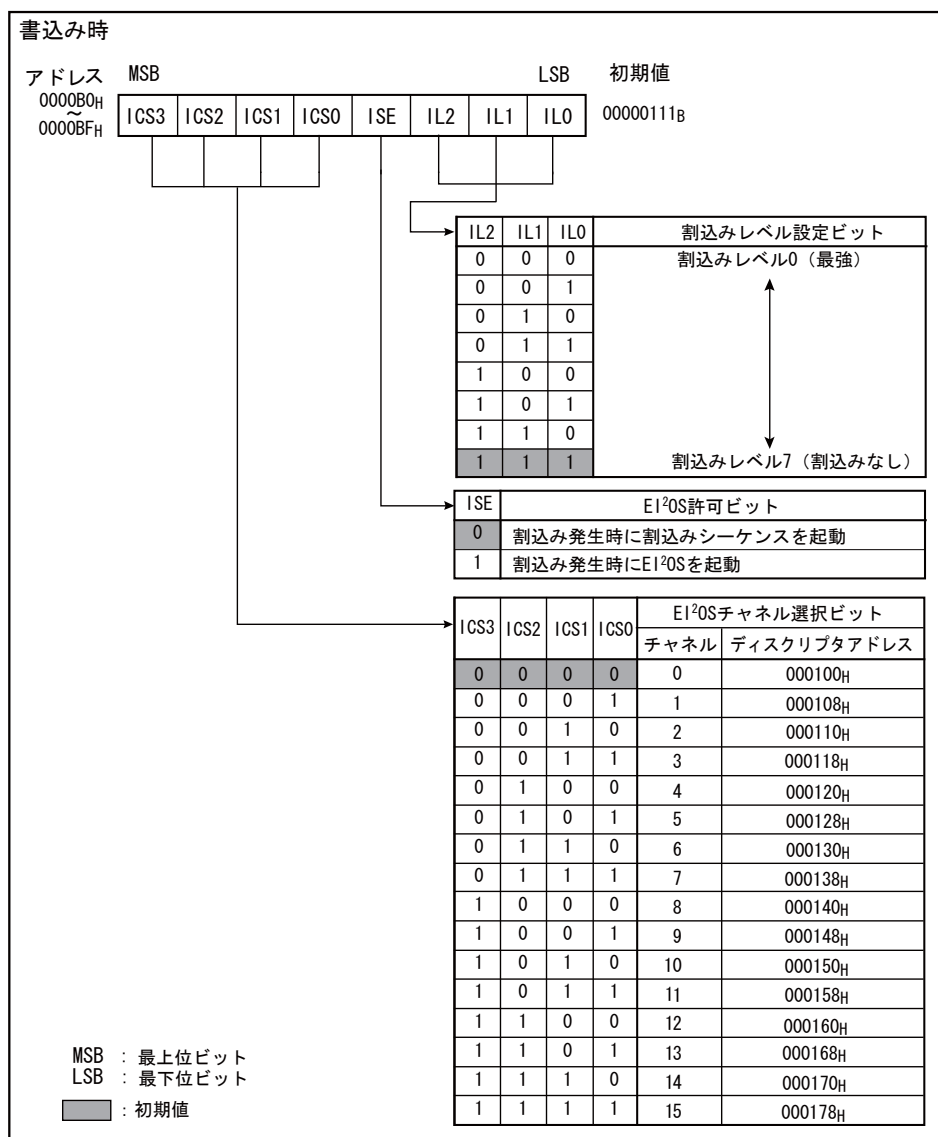
割込み制御レジスタ (ICR) に対するリードモディファイライト (RMV) 系命令でのアクセスは、誤動作を引き起こしますので行わないでください。

3.3.1 割り込み制御レジスタ (ICR00 ~ ICR15)

割り込み制御レジスタ (ICR00 ~ ICR15) は、割り込み機能を持つすべての周辺機能に対応し、割り込み要求発生時の処理を制御します。このレジスタは、書込み時と読出し時では、一部の機能が異なります。

■ 割り込み制御レジスタ (ICR00 ~ ICR15)

図 3.3-1 割り込み制御レジスタ (ICR00 ~ ICR15) 書込み時



読出し時
アドレス

MSB
0000B0H
~
0000BFH

LSB
初期値
- - 000111B

—	—	S1	S0	ISE	IL2	IL1	IL0
---	---	----	----	-----	-----	-----	-----

IL2 IL1 IL0

0	0	0	割り込みレベル設定ビット 割り込みレベル0（最強） ↑ ↓ 割り込みレベル7（割り込みなし）
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

ISE

EI ² OS許可ビット	
0	割り込み発生時に割り込みシーケンスを起動
1	割り込み発生時にEI ² OSを起動

S1 S0

EI ² OSステータス		
0	0	EI ² OS動作中あるいは非起動時
0	1	カウント終了による停止状態
1	0	予約
1	1	周辺機能からの要求による停止状態

MSB : 最上位ビット
LSB : 最下位ビット
— : 未定義
■ : 初期値

3.3.2 割り込み制御レジスタの機能

割り込み制御レジスタ (ICR00 ~ ICR15) は、次に示す 4 つの機能を持つビットで構成されています。

- 割り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 (S1, S0)

■ 割り込み制御レジスタ (ICR) の構成

図 3.3-3 に、割り込み制御レジスタ (ICR) のビット構成図を示します。

図 3.3-3 割り込み制御レジスタ (ICR) の構成

割り込み制御レジスタ (ICR) 書込み時									
アドレス	MSB								LSB 初期値
0000B0 _H ~ 0000BF _H	ICS3	ICS2	ICS1	ICS0	ISE	IL2	IL1	IL0	00000111 _B
割り込み制御レジスタ (ICR) 読出し時									
アドレス	MSB								LSB 初期値
0000B0 _H ~ 0000BF _H	-	-	S1	S0	ISE	IL2	IL1	IL0	--000111 _B
MSB : 最上位ビット									
LSB : 最下位ビット									
- : 未定義									

< 参考 >

- ICS3 ~ ICS0 ビットは、拡張インテリジェント I/O サービス (EI²OS) を起動する場合だけ有効となります。EI²OS を起動する場合は ISE ビットを "1" に設定し、起動しない場合は ISE 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 を設定しなくてもかまいません。
- ICS1 と ICS0 は書込み時だけ、S1 と S0 は読出し時だけ、それぞれ有効です。

< 注意事項 >

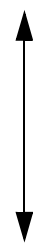
上位 2 ビットの読出し値は不定です。

■ 割込み制御レジスタの機能

● 割込みレベル設定ビット (IL2 ~ IL0)

対応する周辺機能の割込みレベルを指定します。リセットによりレベル7(割込みなし)に初期化されます。割込みレベル設定ビットと各割込みレベルとの関係を、表 3.3-2 に示します。

表 3.3-2 割込みレベル設定ビットと割込みレベルの対応

IL2	IL1	IL0	割込みレベル
0	0	0	0 (最強割込) 
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	6 (最弱割込)
1	1	1	7 (割込なし)

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

割込み要求の発生時に ISE ビットが "1" であると EI²OS が起動され, "0" のときは割込みシーケンスが起動されません。また, EI²OS の終了条件が成立 (S1, S0 ビットが "00_B" 以外) したとき, ISE ビットはクリアされます。対応する周辺機能に EI²OS の機能がないときには, ソフトウェアで ISE ビットを "0" にしておく必要があります。ISE ビットはリセットで "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) チャネル選択ビット (ICS3 ~ ICS0)

書込み専用のビットで, EI²OS のチャネルを指定します。ここで設定された値によって, EI²OS ディスクリプタのアドレスが決定します。ICS ビットはリセットで, "0000_B" に初期化されます。表 3.3-3 に EI²OS チャ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 3.3-3 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)

読出し専用のビットです。EI²OS 終了時にこの値を調べると、動作状態や終了状態を判別できます。リセットで "00_B" に初期化されます。表 3.3-4 に、S0, S1 ビットと EI²OS ステータスの関係について示します。

表 3.3-4 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 動作中あるいは非起動時
0	1	カウント終了による停止状態
1	0	予約
1	1	周辺機能からの要求による停止状態

3.4 ハードウェア割込み

ハードウェア割込みは、周辺機能からの割込み要求信号に対応して、CPU がそれまで実行していたプログラムの実行を一時中断し、ユーザの定義した割込み処理用プログラムに制御を移行する機能です。また、拡張インテリジェント I/O サービス (EI²OS) や μ DMAC、外部割込みなども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の機能

● ハードウェア割込みの機能

ハードウェア割込みは、周辺機能が出力する割込み要求信号の割込みレベルと、CPU のプロセッサステータス (PS) 内の割込みレベルマスクレジスタ (ILM) の比較およびプロセッサステータス (PS) 内の I フラグの内容をハードウェアで参照し、割込みが受け付けられるかどうかを判定します。

ハードウェア割込みが受け付けられると、自動的に CPU 内部のレジスタ類をシステムスタックへ退避し、割込みレベルマスクレジスタ (ILM) に現在要求している割込みレベルを格納したあと、対応する割込みベクタへ分岐します。

● 多重割込み

ハードウェア割込みは、多重に起動できます。

● 拡張インテリジェント I/O サービス (EI²OS)

EI²OS、メモリと I/O 間の自動転送機能ですが、転送完了時に、ハードウェア割込みが起動されます。なお、EI²OS は多重に起動されることはなく、ある EI²OS の処理中は、ほかの割込み要求および μ DMAC 要求はすべて保留されます。

● μ DMAC

μ DMAC は、メモリと I/O 間の自動転送機能ですが、転送完了時に、ハードウェア割込みが起動されます。なお、 μ DMAC は多重に起動されることはなく、ある μ DMAC の処理中は、ほかの割込み要求および 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を含む) は、周辺機能 (割込み要求検出回路) を通じて、ハードウェア割込みとして受け付けられます。

● 割込みベクタ

割込み処理の際に参照する割込みベクタテーブルは、メモリ FFFC00_H ~ FFFFFFF_H に割り当ててあり、ソフトウェア割込みと共用しています。割込み番号と割込みベクタの割当てについては、「3.2 割込み要因と割込みベクタ」を参照してください。

■ ハードウェア割込みの構造

ハードウェア割込みに関連する機構は、表 3.4-1 に示す 4 つの部分に分かれて存在します。ハードウェア割込みを使用する場合はあらかじめ、プログラムでこの 4 箇所を設定する必要があります。

表 3.4-1 ハードウェア割込みに関連する機構

	ハードウェア割込みに 関する機構	機能
周辺機能	割込み許可ビット, 割込み要求 ビット	周辺機能からの割込み要求の制御
割込みコントローラ	割込み制御レジスタ (ICR)	割込みレベルの設定
CPU	割込み許可フラグ (I)	割込み許可状態の識別
	割込みレベルマスクレジスタ (ILM)	要求割込みレベルと現割込みレベルの 比較
	マイクロコード	割込み処理ルーチンの実行
メモリ上の FFFC00 _H ~ FFFFFF _H	割込みベクタテーブル	割込み処理時の分岐先アドレスを格納

■ ハードウェア割込み抑止

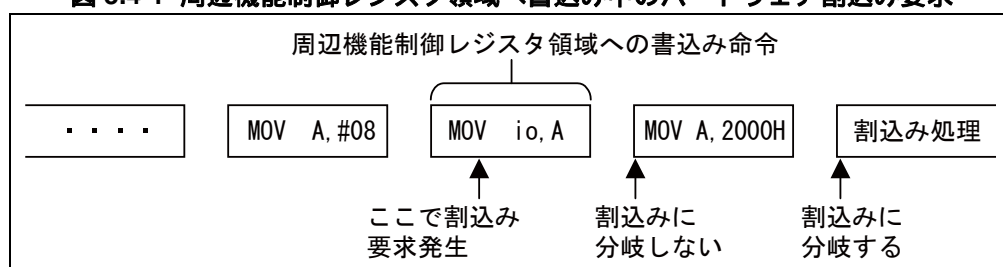
ハードウェア割込みは、以下の条件のとき割込み要求の受け付けが抑止されます。

● 周辺機能制御レジスタ領域への書き込み中のハードウェア割込み抑止

周辺機能制御レジスタ領域への書き込み中は、ハードウェア割込み要求を受け付けません。これは各リソースの割込み制御レジスタ関係の書換えを行っている最中の割込み要求に対して、CPU が割込み関係で誤動作を起こすことを避けるためです。周辺機能制御レジスタ領域とは、000000_H ~ 0000FF_H の I/O アドレッシング領域のことではなく、周辺機能制御レジスタの制御レジスタおよびデータレジスタに割り当てられている領域のことを示します。

図 3.4-1 に、周辺機能制御レジスタ領域へ書き込み中のハードウェア割込み動作について示します。

図 3.4-1 周辺機能制御レジスタ領域へ書き込み中のハードウェア割込み要求



● 割込み抑止命令のハードウェア割込み抑止

ハードウェア割込み抑止命令を表 3.4-2 に示します。ハードウェア割込み抑止命令を実行中にハードウェア割込み要求が発生した場合は、ハードウェア割込み抑止命令処理後、ほかの命令が実行された後に割込み処理が行われます。

表 3.4-2 ハードウェア割込み抑止命令

	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)
割込みやホールド要求を受け付けない命令	PCB DTB ADB SPB CMR NCC	MOV ILM, #imm8 OR CCR, #imm8 AND CCR, #imm8 POPW PS

● ソフトウェア割込み実行中のハードウェア割込み抑止

ソフトウェア割込みを起動すると、I フラグを "0" にクリアするために、ほかの割込み要求は受け付けられません。

3.4.1 ハードウェア割込みの動作

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動作について説明します。

■ ハードウェア割込みの起動

- 周辺機能の動作 (割込み要求の発生)

ハードウェア割込み要求の機能を持つ周辺機能は、割込みの要求のあり / なしを示す「割込み要求フラグ」と、CPU への割込み要求の許可 / 禁止を選択する「割込み許可フラグ」を持っています。割込み要求フラグは、周辺機能固有のイベントの発生によってセットされ、割込み許可フラグが「許可」であったときに、割込みコントローラへ割込み要求を発生します。

- 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) どうしを比較し、最も高いレベルの要求 (IL の値の最も小さいもの) を採択して CPU へ通知します。同一レベルの要求が複数あった場合には、割込み番号が小さいものを優先します。

- CPU の動作 (割込み要求の受け付けと割込み処理)

CPU は受け取った割込みレベル (ICR の IL2 ~ IL0) と割込みレベルマスクレジスタ (ILM) とを比較し、 $IL < ILM$ で、なおかつ割込みが許可 (CCR の I=1) されている場合に、現在実行中の命令が終了後、割込み処理マイクロコードを起動し、割込み処理を実行します。

割込み処理では、まず、システムスタック (SSB と SSP の示すシステムスタック空間) へ専用レジスタ (A, DPR, ADB, DTB, PCB, PC, PS の 12 バイト) の内容を退避します。次に、割込みベクタのプログラムカウンタ (PCB, PC) へのロード、ILM の更新、スタックフラグ (S) の設定 (CCR の S=1 とし、システムスタックを有効にする) を行います。

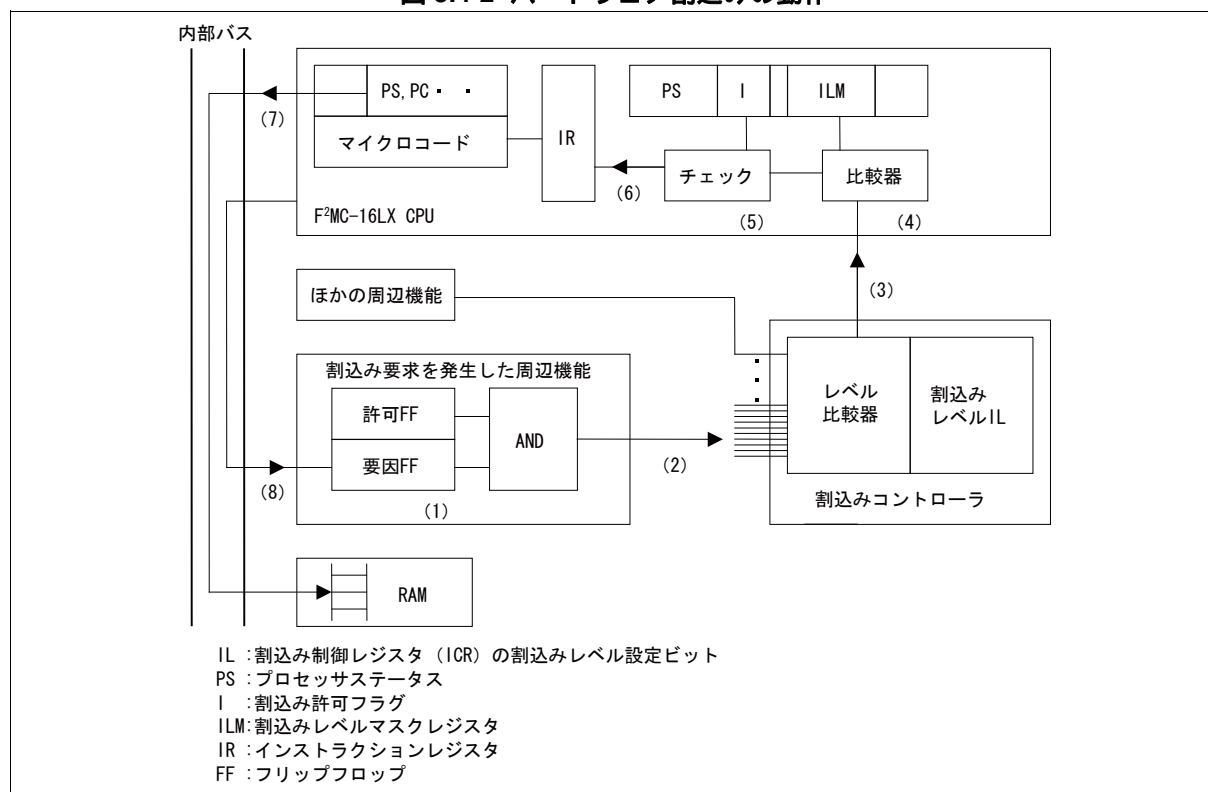
■ ハードウェア割込みからの復帰

割込み処理プログラムの中で、割込み要因となった周辺機能の割込み要求フラグをクリアして、RETI 命令を実行すると、システムスタックに退避している 12 バイトデータを専用レジスタに戻し、割込み分岐前に実行していた処理に復帰します。割込み要求フラグをクリアすることで、周辺機能が割込みコントローラへ出力していた割込み要求は自動的に取り下げられます。

■ ハードウェア割り込みの動作

図 3.4-2 に、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示します。

図 3.4-2 ハードウェア割り込みの動作



- (1) 周辺機能の内部で割り込み要因が発生します。
- (2) 周辺機能の割り込み許可ビットを参照し、割り込み許可になっていれば、周辺から割り込みコントローラへ割り込み要求を出力します。
- (3) 割り込み要求を受け取った割り込みコントローラは、同時に要求のあった割り込みの優先順位を判定したうえで、該当する割り込み要求に対応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から要求のあった割り込みレベル(IL)を、割り込みレベルマスクレジスタ(ILM)と比較します。
- (5) 比較の結果が現在の割り込み処理レベルより優先順位が高い場合、コンディションコードレジスタ(CCR)のIフラグの内容をチェックします。
- (6) (5)のチェックの結果、Iフラグが割り込み許可(I=1)の場合、現在実行中の命令の実行が終了するまで待ち、終了時点でILMに要求されたレベル(IL)を設定します。
- (7) レジスタを退避し、割り込み処理ルーチンへ分岐します。
- (8) 割り込み処理ルーチン内のソフトウェアによって、(1)で発生した割り込み要因をクリアし、RETI命令を実行することによって割り込み処理が終了します。

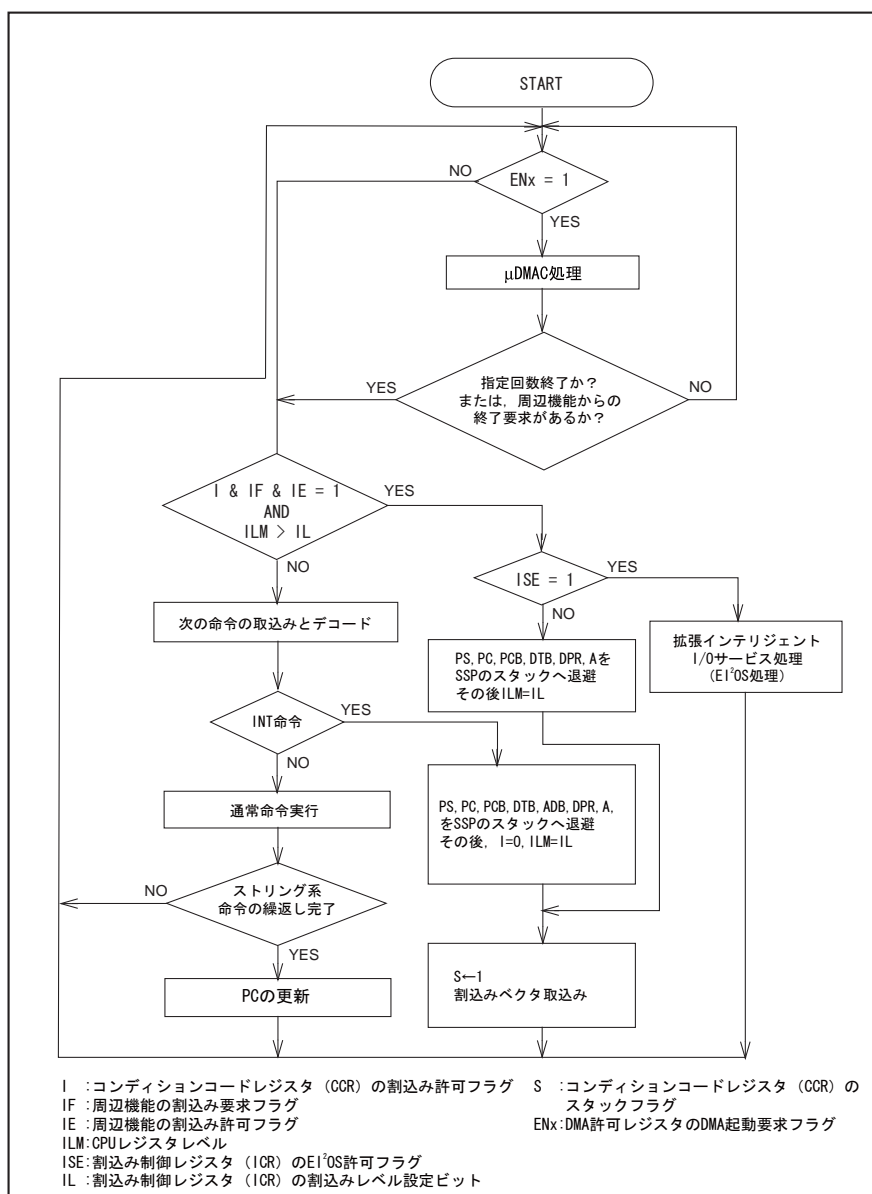
3.4.2 ハードウェア割込みの動作フロー

周辺機能から割込み要求が発生すると、割込みコントローラは割込みレベルを CPU に伝達し、CPU が割込みを受付けられる状態であれば、現在実行中の命令を一時中断して割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) μ DMA の起動を行います。また、INT 命令によるソフトウェア割込みが発生した場合は、CPU の状態にかかわらず割込み処理ルーチンを実行します。このとき、ハードウェア割込みは禁止されます。

■ ハードウェア割込みの動作フロー

図 3.4-3 に、割込み動作時の処理フローを示します。

図 3.4-3 割込み動作時の処理フロー



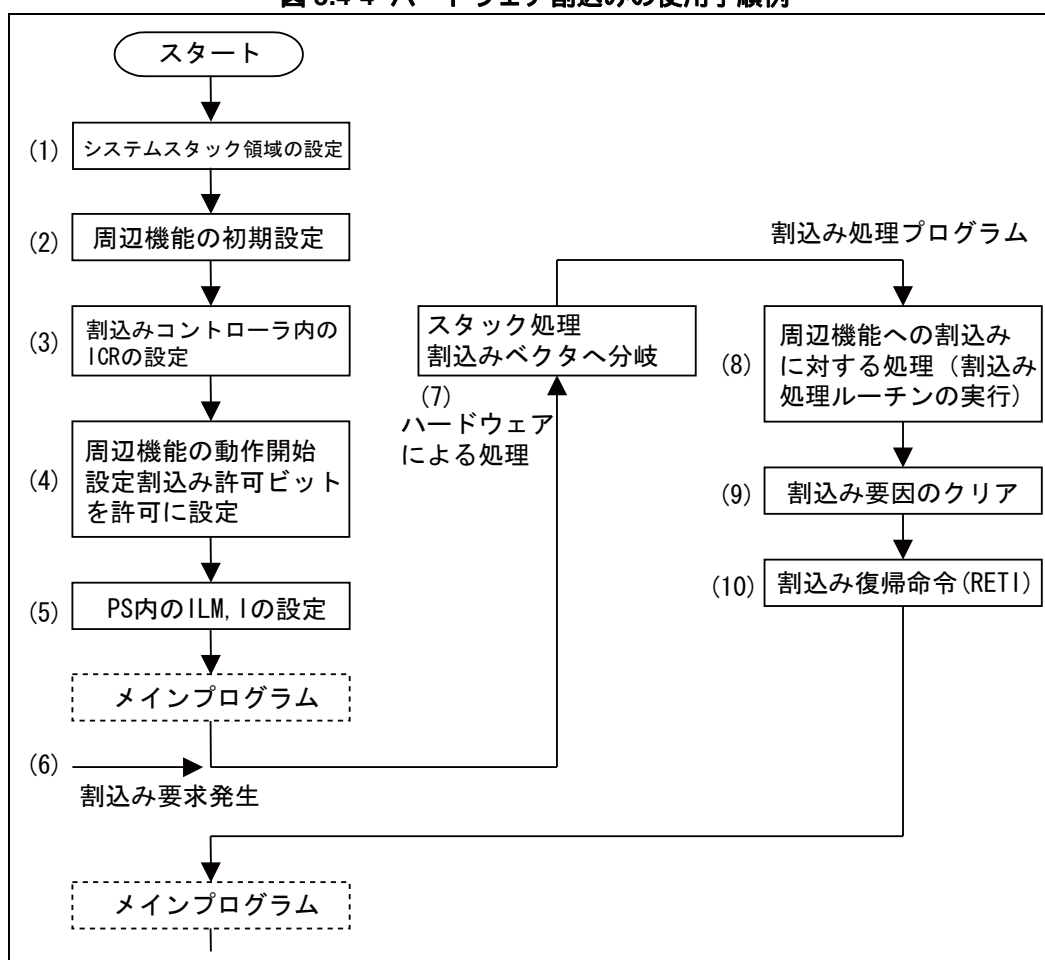
3.4.3 ハードウェア割り込みの使用手順

ハードウェア割り込みを使用するには、システムスタック領域、周辺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。

■ ハードウェア割り込みの使用手順

ハードウェア割り込みの使用手順の一例を、図 3.4-4 に示します。

図 3.4-4 ハードウェア割り込みの使用手順例



- (1) システムスタック領域を設定します。
- (2) 割り込み要求を発生可能な周辺機能の初期設定をします。
- (3) 割り込みコントローラ内の割り込み制御レジスタ (ICR) を設定します。
- (4) 周辺機能を動作開始状態にし、割り込み許可ビットを許可に設定します。
- (5) 割り込みレベルマスクレジスタ (ILM) および割り込み許可フラグ (I) を割り込み受け可能に設定します。
- (6) 周辺機能の割り込み発生で、ハードウェア割り込み要求が発生します。
- (7) 割り込み処理ハードウェアでレジスタの退避が行われ、割り込み処理プログラムに分岐します。

- (8) 割込み処理プログラムで、割込み発生に対する周辺機能への処理をします。
- (9) 周辺機能の割込み要求を解除します。
- (10) 割込み復帰命令を実行し、分岐前のプログラムに復帰します。

3.4.4 多重割込み

ハードウェア割込みでは、周辺機能からの複数の割込み要求に対して、割込み制御レジスタ (ICR) の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によって、多重割込みを実現できます。ただし、拡張インテリジェント I/O サービス (EI²OS), μ DMAC の多重起動はできません。

■ 多重割込み動作

割込み処理ルーチン実行中に、より高い割込みレベルの割込み要求が発生すると、現在の割込み処理を中断してより高い割込み要求を受付けます。高いレベルの割込みが終了すると、もとの割込み処理に戻ります。割込みレベルは0 ~ 7まで設定できますが、レベル7に設定した場合は、CPUは割込み要求を受付けません。

割込み処理実行中にその割込みと同等かより低いレベルの割込みが発生した場合、IフラグがILMを変更しないかぎり、現在の割込み終了するまで新しい割込み要求は保留されます。また、割込み処理ルーチン内で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 (CCRのI=0) にするか、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、割込み中の多重割込みの起動を一時的に禁止できます。

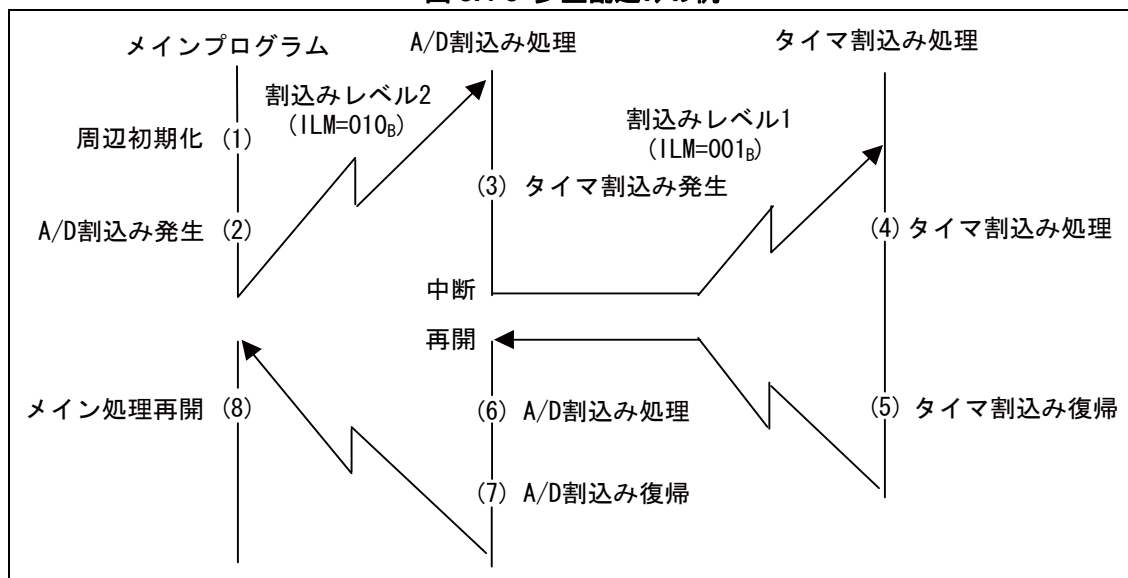
< 注意事項 >

拡張インテリジェント I/O サービス (EI²OS), μ DMAC は多重起動できません。拡張インテリジェント I/O サービス (EI²OS), μ DMAC 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 (EI²OS), μ DMAC 要求はすべて保留されます。

■ 多重割り込みの例

多重割り込み処理の例として、A/Dコンバータよりタイマ割り込みを優先させる場合を想定し、A/Dコンバータの割り込みレベルを2に、タイマ割り込みレベルを1に設定します。このとき、A/Dコンバータの割り込み処理中にタイマ割り込みが発生すると、図3.4-5のような処理を行います。

図 3.4-5 多重割り込みの例



● A/D 割り込み発生

A/Dコンバータ割り込み処理開始時に、割り込みレベルマスクレジスタ (ILM) が自動的にA/Dコンバータの割り込みレベル (ICRのIL2 ~ IL0) と同じ値 (例では2) になります。このとき、レベル1またはレベル0の割り込み要求が発生すると、この割り込み処理を優先して行います。

● 割り込み処理の終了

割り込み処理が終了して復帰命令 (RETI) を実行すると、スタック内に退避した専用レジスタ (A, DPR, ADB, DTB, PCB, PC, PS) の値が復帰し、割り込みレベルマスクレジスタ (ILM) は中断前の値となります。

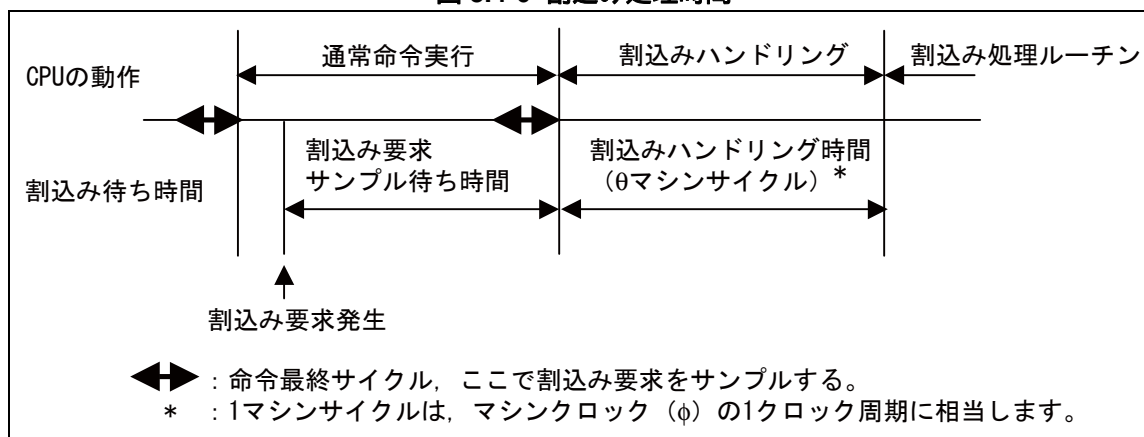
3.4.5 ハードウェア割込みの処理時間

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、現在実行中の命令が終了するまでの時間と、割込みハンドリング時間が必要です。

■ ハードウェア割込み処理時間

割込み要求が発生してから割込みが受け付けられて、割込み処理ルーチンが実行されるまでには、割込み要求サンプル待ち時間と割込みハンドリング時間(割込み処理準備に要する時間)が必要です。図 3.4-6 に、割込み処理時間を示します。

図 3.4-6 割込み処理時間



● 割込み要求サンプル待ち時間

割込み要求サンプル待ち時間とは、割込み要求が発生してから現在実行中の命令が終了するまでの時間をいいます。割込み要求が発生しているかどうかは、各命令の最後のサイクルで割込み要求をサンプリングして判断します。そのため、各命令の実行中は、CPU が割込み要求を認識できず、待ち時間が発生します。

なお、割込み要求サンプル待ち時間は、最も実行サイクルの長いPCPW, PW0, ...RW7命令(45 マシンサイクル)開始直後に割込み要求が発生した場合、最大となります。

● 割込みハンドリング時間 (θ マシンサイクル)

CPUは、割込み要求を受付けてから、専用レジスタのシステムスタックへの退避および割込みベクタの取込みなどを行うため、θマシンサイクルの割込みハンドリング時間を必要とします。割込みハンドリング時間は、以下の式によって求められます。

- 割込み起動時: $\theta = 24 + 6 \times Z$ マシンサイクル
- 割込み復帰時: $\theta = 11 + 6 \times Z$ マシンサイクル (RETI 命令)

割込みハンドリング時間は、スタックポインタの指しているアドレスによって異なります。表 3.4-3 に割込みハンドリング時間の補正值 (Z) を示します。

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。

表 3.4-3 割込みハンドリング時間の補正值 (Z)

スタックポインタが指しているアドレス	補正值 (Z)
外部 8 ビットの場合	+4
外部 偶数アドレスの場合	+1
外部 奇数アドレスの場合	+4
内部 偶数アドレスの場合	0
内部 奇数アドレスの場合	+2

3.5 ソフトウェア割込み

ソフトウェア割込みは、ソフトウェア割込み命令 (INT 命令) を実行すると、CPU がそれまで実行していたプログラムからユーザの定義した割込み処理用プログラムへ制御を移す機能です。

ソフトウェア割込みの実行中、ハードウェア割込みは停止します。

■ ソフトウェア割込みの起動

● ソフトウェア割込みの起動

ソフトウェア割込みの起動には、INT命令を使用します。ソフトウェア割込み要求には、割込み要求フラグおよび許可フラグはなく、INT命令を実行すると常に割込み要求が発生します。

● ハードウェア割込みの抑止

INT 命令には割込みレベルがないので、割込みレベルマスクレジスタ (ILM) の更新は行われません。INT 命令実行中は、コンディションコードレジスタ (CCR) の I フラグを "0" に設定し、ハードウェア割込みをマスクします。ソフトウェア割込み処理中でもハードウェア割込みを許可するときは、ソフトウェア割込み処理ルーチンの中で I フラグを "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 命令を取り込んで実行すると、ソフトウェア割込み処理用のマイクロコードを起動します。このマイクロコードにより、CPU 内部のレジスタ類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 (CCR の I=0) したあと、対応する割込みベクタへ分岐します。

割込み番号と割込みベクタの割当てについては、「3.2 割込み要因と割込みベクタ」を参照してください。

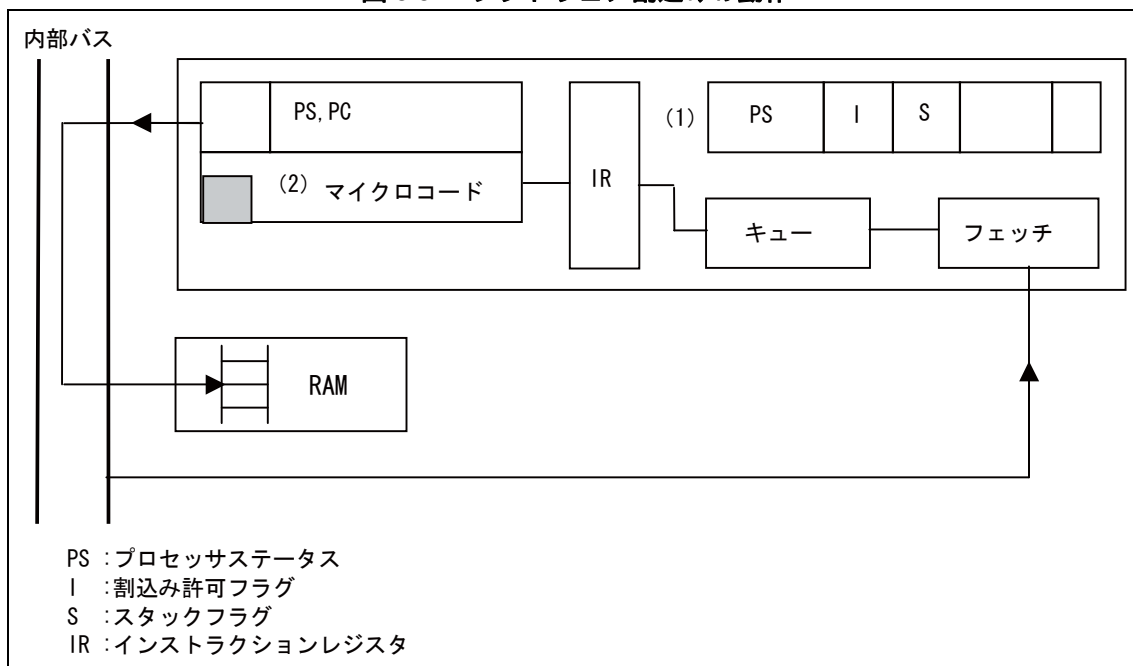
■ ソフトウェア割込みからの復帰

割込み処理プログラムの中で割込み復帰命令 (RETI 命令) を実行すると、システムスタックに退避している 12 バイトデータを専用レジスタに戻して、割込み分岐前に実行していた処理に復帰します。

■ ソフトウェア割込みの動作

図 3.5-1 に、ソフトウェア割込みの発生から割込み処理完了までの動作を示します。

図 3.5-1 ソフトウェア割込みの動作



- (1) ソフトウェア割込み命令を実行します。
- (2) ソフトウェア割込み命令に対応したマイクロコードに従って専用レジスタを退避するなどの必要な処理を行った後、分岐処理を行います。
- (3) ユーザの割込み処理ルーチン内の RETI 命令で割込み処理が終了します。

■ ソフトウェア割込みに関する注意事項

プログラムカウンタバンクレジスタ (PCB) が "FF_H" であるとき、CALLV 命令のベクタ領域は INT#vct8 命令のテーブルと重なります。ソフトウェアを作成する際に、CALLV 命令と INT#vct8 命令のアドレスの重複に注意してください。

3.6 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、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、データ転送終了時に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。これは、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、従来は割込み処理プログラムで行っていた周辺機能 (I/O) とのデータのやりとりを DMA (ダイレクト メモリ アクセス) のように行い、終了時に終了条件を設定した後、自動的に割込み処理ルーチンへ分岐します。ユーザは、EI²OS 起動時と終了時のみプログラムを作成することになり、途中のデータ転送プログラムは不要です。

● 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行っていたデータ転送と比べると、次のような利点があります。

- 転送のためのプログラムを記述する必要がないため、プログラムサイズを小さくできる。
- 周辺機能 (I/O) の状態によって転送を停止できるので、不要なデータの転送の必要がない。
- バッファアドレスのインクリメント、および更新なしを選択可能。
- I/O レジスタアドレスのインクリメント、および更新なしを選択可能。

● 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、終了条件を割込み制御レジスタ (ICR) の S1, S0 ビットに設定し自動的に割込み処理ルーチンへ分岐します。

割込み処理プログラムで、EI²OS ステータス (ICR:S1, S0) をチェックすることによって、EI²OS の終了要因を判断できます。

割込み番号や割込みベクタは、各周辺で固定されています。詳細は「3.2 割込み要因と割込みベクタ」を参照してください。

● 割込み制御レジスタ (ICR)

割込みコントローラ内にあり、EI²OS の起動、EI²OS のチャネル指定、および EI²OS 終了時の状態表示を行います。

● 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)

RAM 上の 000100_H ~ 00017F_H にあり、転送モード、I/O のアドレスや転送数、バッファアドレスを保持する 8 バイトデータで、16 チャネル分あります。割込み制御レジスタ (ICR) でチャネル指定します。

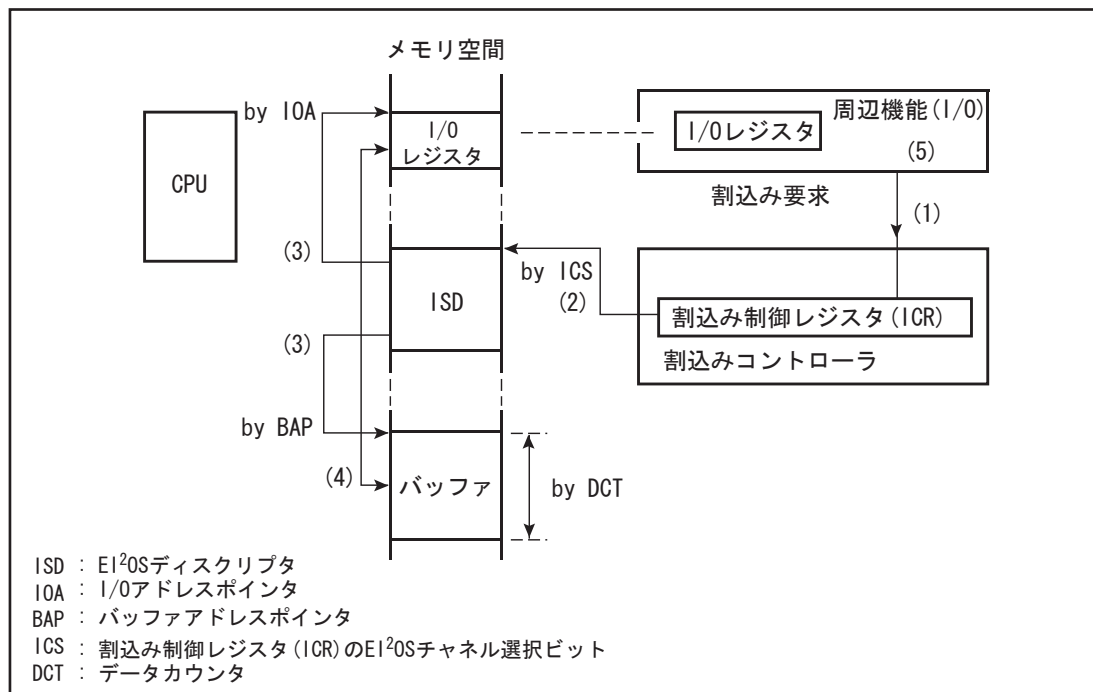
< 注意事項 >

拡張インテリジェント I/O サービス (EI²OS) が動作しているとき、CPU のプログラム実行は停止します。

■ 拡張インテリジェント I/O サービス (EI²OS) の動作

図 3.6-1 に、EI²OS の動作を示します。

図 3.6-1 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
- (2) 割り込みコントローラがディスクリプタを選択
- (3) 転送元や転送先をディスクリプタから読み出す
- (4) I/O とメモリ間での転送を実行
- (5) 割り込み要因は自動的にクリア

3.6.1 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) ディスクリプタ (ISD) は、内部 RAM の 000100_H ~ 00017F_H に存在し、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 3.6-2 のような構造になっています。チャンネル番号と ISD のアドレスの対応は、表 3.6-1 のとおりです。

図 3.6-2 EI²OS ディスクリプタ (ISD) の構成

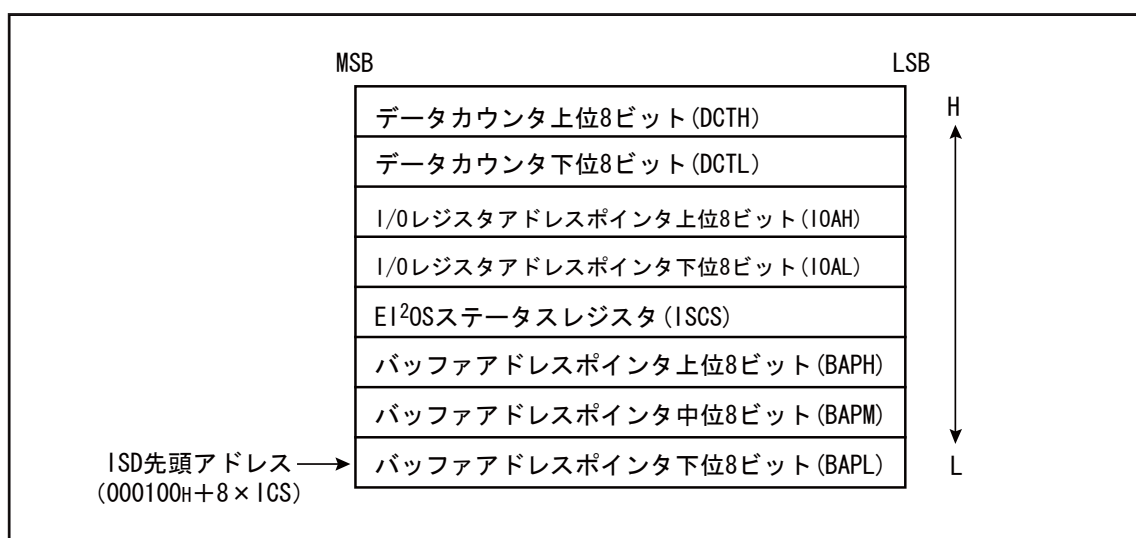


表 3.6-1 チャンネル番号とディスクリプタアドレスの対応

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

3.6.2 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) の各レジスタ

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) は、次のレジスタから構成されています。

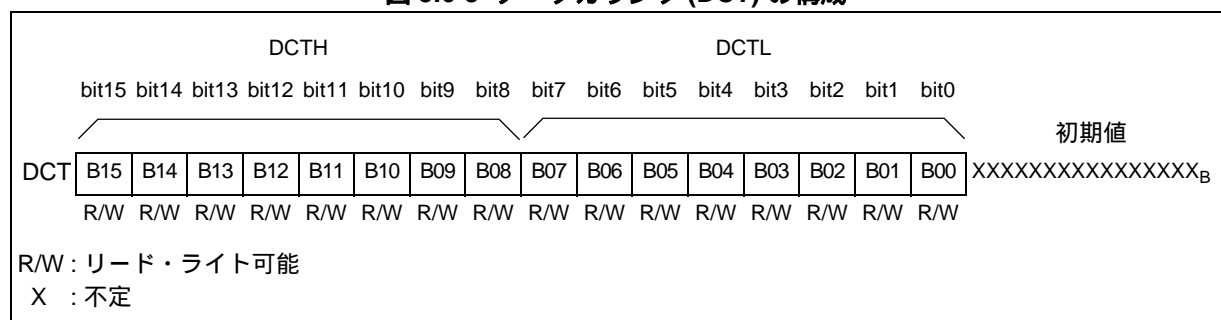
- データカウンタ (DCT)
- I/O レジスタアドレスポインタ (IOA)
- EI²OS ステータスレジスタ (ISCS)
- バッファアドレスポインタ (BAP)

各レジスタの初期値は、リセット時には不定となる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、16 ビット長のレジスタで、転送データ数に対応したカウンタとなります。各データの転送後に、カウンタは1デクリメント (減値) します。このカウンタが0になると EI²OS は終了します。図 3.6-3 に、DCT の構成を示します。

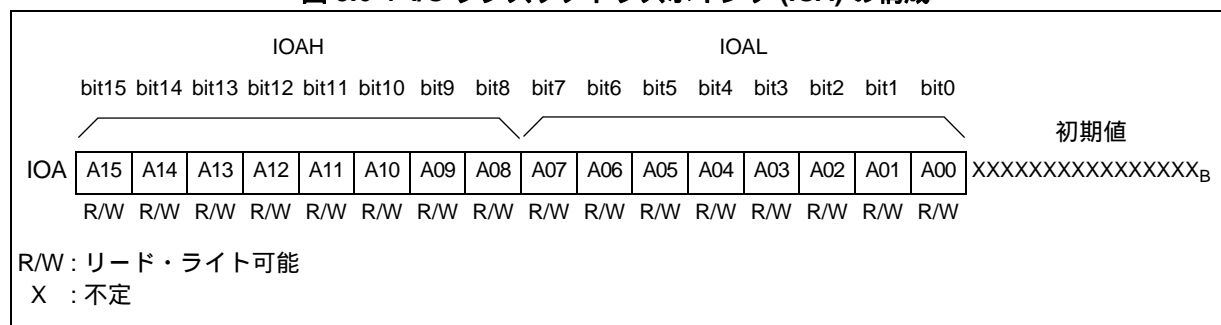
図 3.6-3 データカウンタ (DCT) の構成



■ I/O レジスタアドレスポインタ (IOA)

I/O レジスタアドレスポインタ (IOA) は、16 ビット長のレジスタで、バッファとのデータ転送を行う I/O レジスタの下位アドレス (A15 ~ A00) を示します。上位アドレス (A23 ~ A16) はすべて "0" であり、000000_H から 00FFFF_H 番地までの任意の I/O をアドレスで指定できます。図 3.6-4 に、IOA の構成を示します。

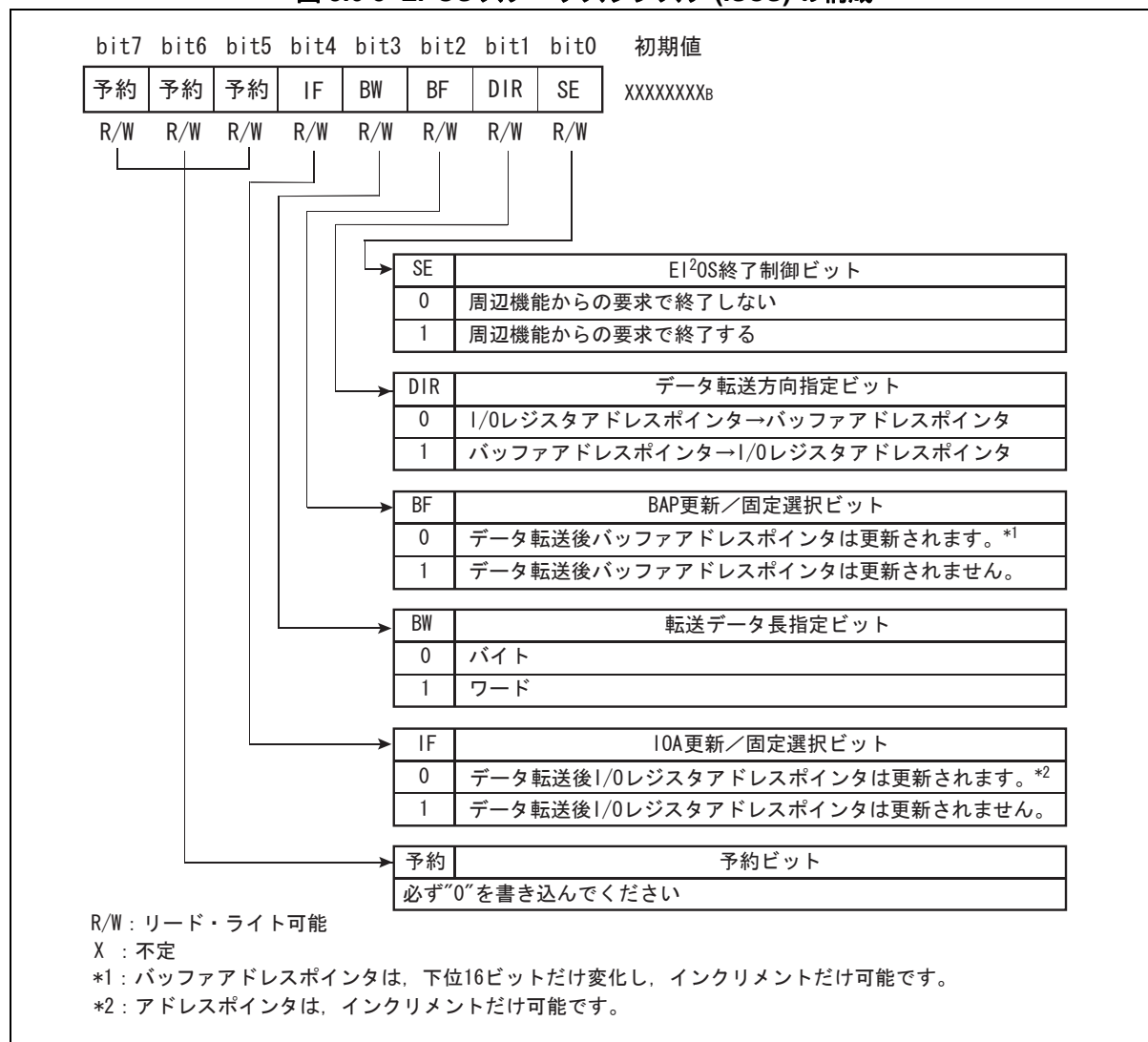
図 3.6-4 I/O レジスタアドレスポインタ (IOA) の構成



■ 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)

拡張インテリジェント I/O サービスステータスレジスタ (ISCS) は、8 ビット長で、バッファアドレスポインタと I/O レジスタアドレスポインタの更新 / 固定、転送データ形式 (バイト / ワード)、転送方向を示します。図 3.6-5 に、ISCS の構成を示します。

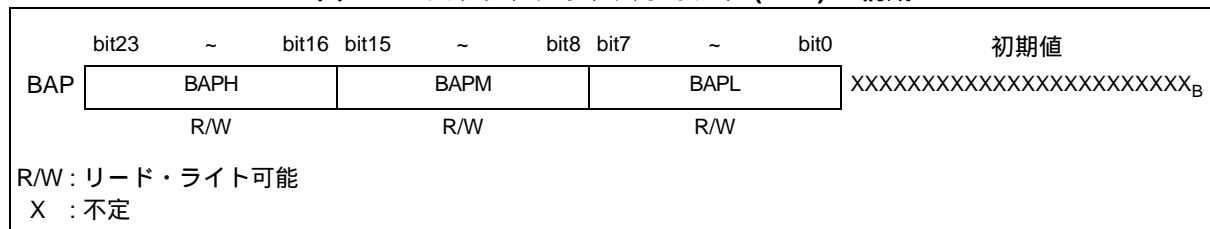
図 3.6-5 EI²OS ステータスレジスタ (ISCS) の構成



■ バッファアドレスポインタ (BAP)

バッファアドレスポインタ (BAP) は、24 ビットのレジスタで、EI²OS による次回の転送で使用するアドレスを保持します。BAP は、EI²OS の各チャネルに対してそれぞれ独立に存在するので、EI²OS の各チャネルは、16M バイトの任意のアドレスと I/O の間でデータを転送できます。EI²OS ステータスレジスタ (ISCS) の BF ビット (EI²OS ステータスレジスタの BAP 更新 / 固定選択ビット) で "更新あり" とした場合、BAP は下位 16 ビット (BAPM, BAPL) だけ変化し、上位 8 ビット (BAPH) は変化しません。図 3.6-6 に、バッファアドレスポインタ (BAP) の BAP 構成を示します。

図 3.6-6 バッファアドレスポインタ (BAP) の構成



< 注意事項 >

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 ~ FFFFFFF_H です。

データカウンタ (DCT) で指定できる最大転送数は, 65536 個 (64K バイト) です。

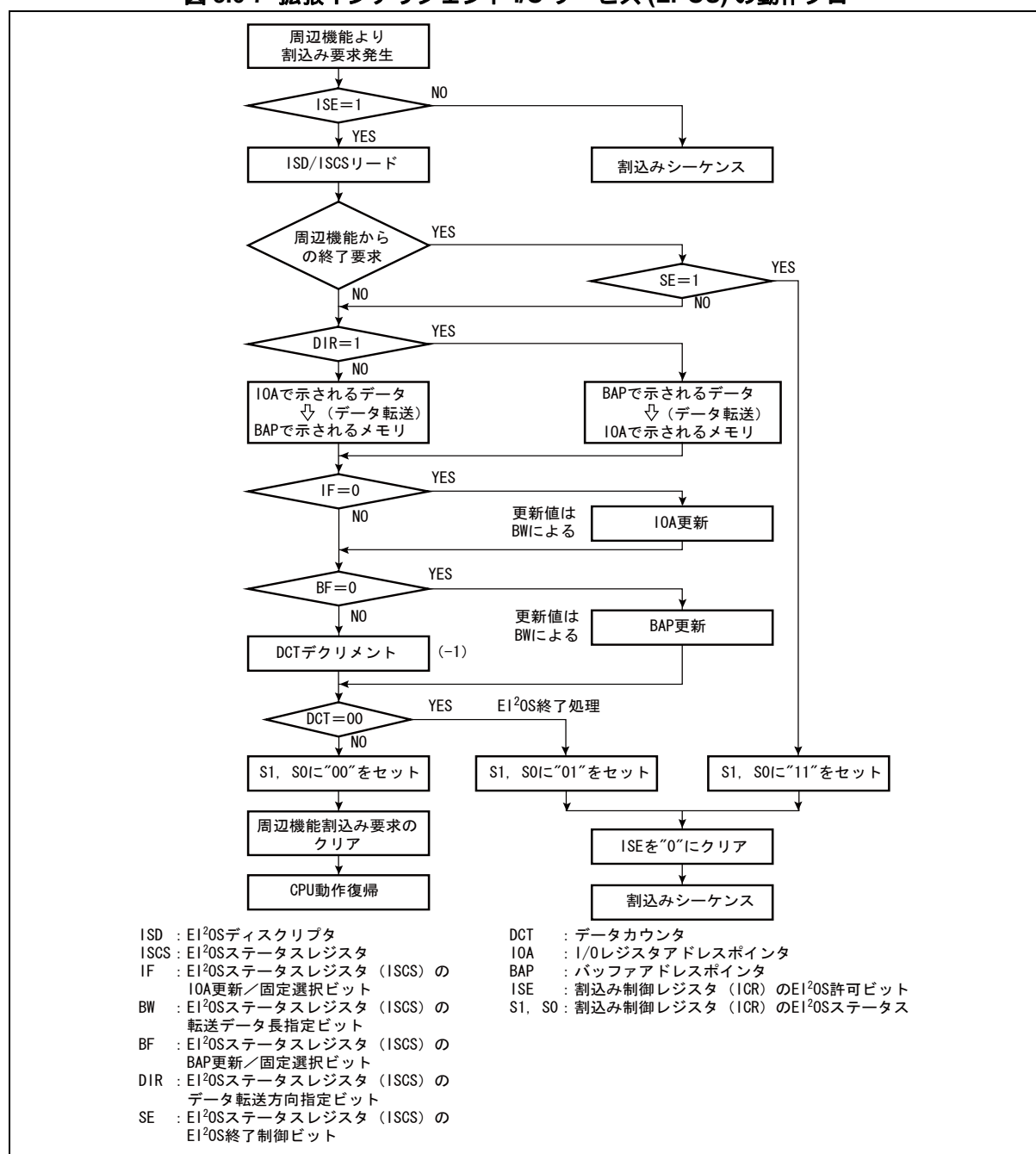
3.6.3 拡張インテリジェント I/O サービス (EI²OS) の動作

周辺機能から割り込み要求が発生し、対応する割り込み制御レジスタ (ICR) で EI²OS の起動が設定されていれば、CPU は EI²OS によるデータ転送を行います。指定回数のデータ転送が終了すれば、自動的にハードウェア割り込み処理を行います。

■ 拡張インテリジェント I/O サービス (EI²OS) の処理手順

図 3.6-7 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 3.6-7 拡張インテリジェント I/O サービス (EI²OS) の動作フロー

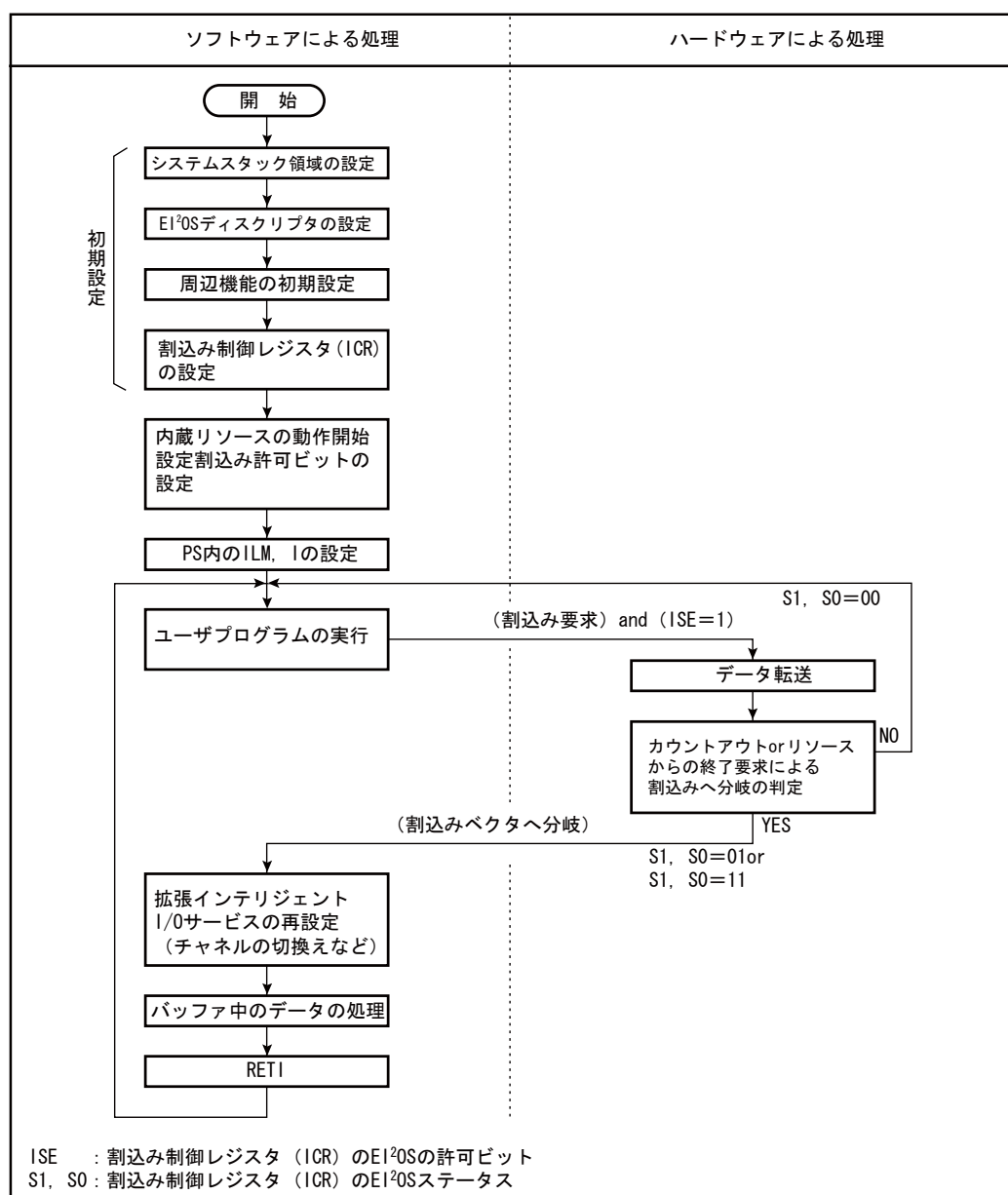


3.6.4 拡張インテリジェント I/O サービス (EI²OS) の使用手順

拡張インテリジェント I/O サービス (EI²OS) を使用するには、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、周辺機能、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 3.6-8 に、EI²OS のソフトウェアとハードウェアによる処理を示します。

図 3.6-8 拡張インテリジェント I/O サービス (EI²OS) 使用手順

3.6.5 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) の処理に必要な時間は次の要因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)
- 外部アクセス時の外部データバス幅
- 転送データのデータ長

また、EI²OS によるデータ転送終了時には、ハードウェア割込みが起動するため、割込みハンドリング時間が加算されます。

■ 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)

● データ転送継続時

データ転送継続時の EI²OS 処理時間は、EI²OS ステータスレジスタ (ISCS) の設定によって、表 3.6-2 のようになります。

表 3.6-2 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		周辺からの終了要求により終了		周辺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、マシクロック (φ) の 1 クロック周期に相当します)。

さらに、表 3.6-3 に示すように、EI²OS 実行時の条件によって補正が必要です。

表 3.6-3 EI²OS 実行時間のデータ転送の補正值

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	奇	B/ 偶	8/ 奇
バッファアドレス ポインタ	内部アクセス	B/ 偶	0	+ 2	+ 1	+ 4
		奇	+ 2	+ 4	+ 3	+ 6
	外部アクセス	B/ 偶	+ 1	+ 3	+ 2	+ 5
		8/ 奇	+ 4	+ 6	+ 5	+ 8

B : バイトデータ転送

8 : 外部バス幅 8 ビット・ワード転送

偶 : 偶数アドレス・ワード転送

奇 : 奇数アドレス・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 によるデータ転送終了時には、ハードウェア割込みが起動するため、割込みハンドリング時間が加算されます。カウント終了時の EI²OS の処理時間は、次の式で算出されます。

カウント終了時の EI²OS 処理時間 =

データ転送時の EI²OS 処理時間 + $(21+6 \times Z)$ マシンサイクル

割込みハンドリング時間

割込みハンドリング時間は、スタックポインタの指しているアドレスによって異なります。表 3.6-4 に、割込みハンドリング時間の補正值 (Z) を示します。

表 3.6-4 割込みハンドリング時間の補正值 (Z)

スタックポインタが指しているアドレス	補正值 (Z)
外部 8 ビットの時	+ 4
外部 偶数アドレスの時	+ 1
外部 奇数アドレスの時	+ 4
内部 偶数アドレスの時	0
内部 奇数アドレスの時	+ 2

● 周辺機能 (I/O) からの終了要求による終了時

周辺機能 (I/O) からの終了要求で、EI²OS によるデータ転送を途中で終了した場合 (ICR:S1, S0=11), データ転送は行わず、ハードウェア割込みを起動します。この場合の EI²OS 処理時間は、次の式で算出されます。式中の Z は、割込みハンドリング時間の補正值 (表 3.6-4 を参照) です。

途中終了時の EI²OS 処理時間 = $36 + 6 \times Z$ マシンサイクル

< 参考 >

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。

3.7 例外処理割込み

F²MC-16LX では、未定義命令の実行で例外処理が行われます。例外処理は、基本的には割込みと同じものであり、命令の境目で例外事項が発生したことを検出した段階で、通常処理を中断して例外処理を行います。

一般的に、例外処理は予想外の動作を行った結果によって発生するものであり、デバッグ時や緊急時の復旧ソフトウェアの起動などだけに使用することを推奨します。

■ 例外処理割込み

● 例外処理の動作

F²MC-16LX では、命令マップで定義していないコードはすべて未定義命令として扱います。未定義命令を実行すると、ソフトウェア割込み命令の "INT #10" と等価な処理を行います。

例外処理では、割込みルーチンへ分岐する前に、次の処理が行われます。

1. システムスタックへ A, DPR, ADB, DTB, PCB, PC, PS レジスタを退避
2. コンディションコードレジスタ (CCR) の I フラグを "0" にクリアし、ハードウェア割込みをマスク
3. コンディションコードレジスタ (CCR) の S フラグを "1" にセットし、システムスタックを有効にする。

スタックへ退避したプログラムカウンタ (PC) の値は、未定義命令を格納しているアドレスそのものです。2 バイト以上の命令コードでは、未定義であることが識別できたコードを格納しているアドレスになります。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

例外処理から RETI 命令で復帰させると、PC が未定義命令を指しているために、再度、例外処理に入ります。ソフトウェアリセットなどの対策をしてください。

3.8 μ DMAC による割込み

μ DMAC は、EI²OS と同等の機能を持った簡易 DMA です。

3.8.1 μ DMAC の機能

3.8.2 μ DMAC のレジスタ

3.8.2.1 DMA ディスクリプタチャンネル指定レジスタ (DCSR)

3.8.2.2 DMA ステータスレジスタ (DSRH/DSRL)

3.8.2.3 DMA ストップステータスレジスタ (DSSR)

3.8.2.4 DMA 許可レジスタ (DERH/DERL)

3.8.3 DMA ディスクリプタウィンドウレジスタ (DDWR)

3.8.3.1 DMA データカウンタ (DDCTH/DDCTL)

3.8.3.2 DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL)

3.8.3.3 DMA コントロールレジスタ (DMACS)

3.8.3.4 DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)

3.8.4 μ DMAC の動作説明

3.8.1 μ DMAC の機能

μ DMAC はディスクリプタ用レジスタを持つことによって、EI²OS に比べて高速にデータ転送を行うことを可能にしています。

■ μ DMAC の機能

μ DMAC は、以下の機能をもっています。

- 周辺リソース (I/O) とメモリ間で自動データ転送を行います。
- DMA 起動中は、CPU のプログラム実行は停止します。
- DMA 転送チャンネルは 16 チャンネルです (チャンネルの番号の小さい方が DMA 転送の優先順位が高い)。
- 転送元、転送先アドレスのインクリメントあり、またはインクリメントなしを選択可能です。
- DMA 転送は、周辺リソース (I/O) の割り込み要因により起動します。
- DMA 転送の制御は、(a) DMA 許可レジスタ (DERH/DERL), (b) DMA ストップステータスレジスタ (DSSR), (c) DMA ステータスレジスタ (DSRH/DSRL), (d) DMA ディスクリプタチャンネル指定レジスタ (DCSR), (e) ディスクリプタ (DMACS) で行います。
- リソースから DMA 転送を停止させる手段として STOP 要求があります。
- DMA 転送終了後、DMA ステータスレジスタ (DSRH/DSRL) の該当するビットにフラグがセットされ、割り込みコントローラへ割り込みが出力されます。

3.8.2 μ DMAC のレジスタ

μ DMAC は、DCSR, DSR, DSSR, DER という4つのレジスタを持っています。DMA 転送の設定に使用される DMA ディスクリプタについては「3.8.3 DMA ディスクリプタウィンドウレジスタ (DDWR)」で説明しています。

■ μ DMAC レジスタ一覧

図 3.8-1 μ DMAC レジスタ一覧

DMA ディスクリプタチャンネル指定レジスタ (DCSR)									
bit	15	14	13	12	11	10	9	8	
00009B _H	STP	予約	予約	予約	DCSR3	DCSR2	DCSR1	DCSR0	DCSR
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
DMA ステータスレジスタ (DSRH/DSRL)									
bit	15	14	13	12	11	10	9	8	
00009D _H	DTE15	DTE14	DTE13	DTE12	DTE11	DTE10	DTE9	DTE8	DSRH
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
bit	7	6	5	4	3	2	1	0	
00009C _H	DTE7	DTE6	DTE5	DTE4	DTE3	DTE2	DTE1	DTE0	DSRL
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
DMA ストップステータスレジスタ (DSSR)									
bit	7	6	5	4	3	2	1	0	
0000A4 _H	STP15	STP14	STP13	STP12	STP11	STP10	STP9	STP8	DSSR
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
bit	7	6	5	4	3	2	1	0	
0000A4 _H	STP7	STP6	STP5	STP4	STP3	STP2	STP1	STP0	DSSR
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
(DSSR は DCSR の STP ビットが "0" のとき STP15 ~ STP8 を, "1" のとき STP7 ~ STP0 を使用)									
DMA 許可レジスタ (DERH/DERL)									
bit	15	14	13	12	11	10	9	8	
0000AD _H	EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8	DERH
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
bit	7	6	5	4	3	2	1	0	
0000AC _H	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	DERL
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
R/W: リード・ライト可能									

3.8.2.1 DMA ディスクリプタチャネル指定レジスタ (DCSR)

DMA ディスクリプタチャネル指定レジスタ (DCSR) は、各チャネルのディスクリプタを切り換えるレジスタです。本レジスタでチャネルを指定してからディスクリプタを設定してください。

■ DMA ディスクリプタチャネル指定レジスタ (DCSR)

図 3.8-2 DMA ディスクリプタチャネル指定レジスタ (DCSR) のビット構成

bit	15	14	13	12	11	10	9	8	
00009B _H	STP	予約	予約	予約	DCSR3	DCSR2	DCSR1	DCSR0	DCSR
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
R/W: リード・ライト可能									

[bit11 ~ bit8] DCSRx :DMA ディスクリプタチャネル指定

表 3.8-1 DCSR と選択チャネルの関係

DCSR3 ~ DCSR0	選択チャネル	リソース割込み要求
0000 _B	0	USB ファンクション 1 (End Point 0-IN)
0001 _B	1	USB ファンクション 1 (End Point 0-OUT)
0010 _B	2	USB ファンクション 2 (End Point 1) *
0011 _B	3	USB ファンクション 2 (End Point 2) *
0100 _B	4	USB ファンクション 2 (End Point 3)
0101 _B	5	USB ファンクション 2 (End Point 4)
0110 _B	6	USB ファンクション 2 (End Point 5)
0111 _B	7	インプットキャプチャ 0, 1
1000 _B	8	インプットキャプチャ 2, 3
1001 _B	9	拡張シリアル I/O
1010 _B	10	UART2/UART3・受信
1011 _B	11	UART2/UART3・送信
1100 _B	12	UART0/UART1・受信
1101 _B	13	UART0/UART1・送信
1110 _B	14	PWC・リロードタイマ 0
1111 _B	15	A/D コンバータ・変換終了・フリーランタイマ

DCSR の設定により、16 チャンネル中 1 チャンネルのディスクリプタチャンネルが選択されます。詳細は「3.8.3 DMA ディスクリプタウィンドウレジスタ (DDWR)」を参照してください。

*: USB ホスト動作時でも使用できます。

[bit15] STP:STP 制御ビット

STP ビット	機能
0 [初期値]	DSSR として STP8 ~ STP15 を選択します。
1	DSSR として STP0 ~ STP7 を選択します。

[bit14 ~ bit12] 予約: (予約ビット)

bit14 ~ bit12 の 3 ビットは予約ビットです。

これらのビットの読出し値は常に "0" です。

これらのビットには必ず "0" 書き込んでください。

< 注意事項 >

DCSR レジスタにリードモディファイライト (RMW) 系命令を使用しないでください。

3.8.2.2 DMA ステータスレジスタ (DSRH/DSRL)

DMA ステータスレジスタ (DSRH/DSRL) は、DMA 転送が終了したことを示すレジスタです。本レジスタに "1" がセットされると同時に割り込みが発生します。

■ DMA ステータスレジスタ (DSRH/DSRL)

図 3.8-3 DMA ステータスレジスタ (DSRH/DSRL) のビット構成

bit	15	14	13	12	11	10	9	8	
00009D _H	DTE15	DTE14	DTE13	DTE12	DTE11	DTE10	DTE9	DTE8	DSRH
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
bit	7	6	5	4	3	2	1	0	
00009C _H	DTE7	DTE6	DTE5	DTE4	DTE3	DTE2	DTE1	DTE0	DSRL
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B

R/W: リード・ライト可能

[bit15 ~ bit0] DTE_x :DMA ステータス

DTE _x ビット	機能
0 [初期値]	DMA 転送の終了による割り込みは発生していません。 DTE _x が "0" のとき、必ず "0" を書き込んでください。
1	DMA 転送が終了して、割り込みを要求していることを示しています。最終転送を除いて STOP 要求による DMA 転送の停止では "1" にセットされません。 DTE _x が "1" のとき、"0" を書き込むと "0" にクリアされ、"1" を書き込むと前のデータを保持します。

< 注意事項 >

DSRH/DSRL に書き込みを行う場合、リードモディファイライト (RMW) 系の命令を使用してください。

< 注意事項 >

- DSSR は、DCSR の最上位ビット (STP) により制御されます。STP が "0" のときは、STP15 ~ STP8 が DSSR として選択され、"1" のときは、STP7 ~ STP0 が DSSR として選択されます。STP は初期値が "0" なので、最初は STP15 ~ STP8 が選択されています。
 - DSSR に書込みを行う場合、リードモディファイライト (RMW) 系の命令を使用してください。
-

3.8.2.4 DMA 許可レジスタ (DERH/DERL)

DMA 許可レジスタ (DERH/DERL) は DMA 転送を許可するレジスタです。本レジスタを "1" にセットし、対応するチャンネルに割り込み要求が発生すると DMA 転送要求とみなし、DMA 転送を開始します。

■ DMA 許可レジスタ (DERH/DERL)

図 3.8-5 DMA 許可レジスタ (DERH/DERL) のビット構成

bit	15	14	13	12	11	10	9	8	
0000AD _H	EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8	DERH
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B
bit	7	6	5	4	3	2	1	0	
0000AC _H	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	DERL
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0000000 _B

R/W: リード・ライト可能

[bit15 ~ bit0] ENx:DMA 許可

ENx ビット	機能
0 [初期値]	DMA 転送を実行しません。
1	リソースからの割り込み要求を DMA 起動要求とし、DMA 転送終了時点で割り込み要求を割り込みコントローラに出力します。 DMA 転送バイト数が0となるか、リソースからの STOP 要求で DMA 転送が停止された時点で、"0" にクリアされます。

< 注意事項 >

- DERH/DERL に書き込みを行う場合、リードモディファイライト (RMW) 系の命令を使用してください。
- スタンバイモード (スリープモード、ストップモード、時計モード、タイムベースタイマモード) または CPU 間欠動作モード (メインクロック間欠動作モード、PLL クロック間欠モード、サブクロック間欠モード) に移行する場合、必ず DMA 許可レジスタ (DERH/DERL) に "0" を設定してから各モードに移行してください。

3.8.3 DMA ディスクリプタウィンドウレジスタ (DDWR)

DMA ディスクリプタは 8 バイト \times 16 チャンネルで構成されており、DMA 転送の設定に使用されます。16 チャンネル中から指定された 1 チャンネルが DMA ディスクリプタウィンドウレジスタ (DDWR) にマッピングされアクセス可能となります。

DDWR のアドレスは 007920_H ~ 007927_H です。

■ DMA ディスクリプタウィンドウレジスタ (DDWR) の構成

DMA ディスクリプタは、8 バイト \times 16 チャンネルで構成されており、各チャンネルは図 3.8-6 に示す構造になっています。DMA ディスクリプタチャンネル指定レジスタ (DCSR) または割込み要求チャンネル番号によって選択されたチャンネルのディスクリプタが、DMA ディスクリプタウィンドウレジスタ (DDWR) 上にマッピングされます。DMA ディスクリプタチャンネル指定レジスタ (DCSR) と選択チャンネルの関係は、表 3.8-1 を参照してください。

図 3.8-6 DMA ディスクリプタウィンドウレジスタ (DDWR) の構成

アドレス	
007927 _H	DMA データカウンタ上位 8 ビット (DDCTH)
007926 _H	DMA データカウンタ下位 8 ビット (DDCTL)
007925 _H	DMA I/O レジスタアドレスポインタ上位 8 ビット (DIOAH)
007924 _H	DMA I/O レジスタアドレスポインタ下位 8 ビット (DIOAL)
007923 _H	DMA コントロールレジスタ (DMACS)
007922 _H	DMA バッファアドレスポインタ上位 8 ビット (DBAPH)
007921 _H	DMA バッファアドレスポインタ中位 8 ビット (DBAPM)
007920 _H	DMA バッファアドレスポインタ下位 8 ビット (DBAPL)

■ DMA ディスクリプタの各レジスタ

DMA ディスクリプタを構成する各レジスタについて次ページ以降に説明します。各レジスタの初期値は、リセット時には不定となるので必ず ENx に "1" を設定する前までに初期化してください。

< 注意事項 >

DMA ディスクリプタチャンネル指定レジスタ (DCSR) によって、チャンネルのディスクリプタを切り換えた場合、2 マシンサイクル以内は DMA ディスクリプタウィンドウレジスタ (DDWR) へのアクセスは禁止です。

3.8.3.1 DMA データカウンタ (DDCTH/DDCTL)

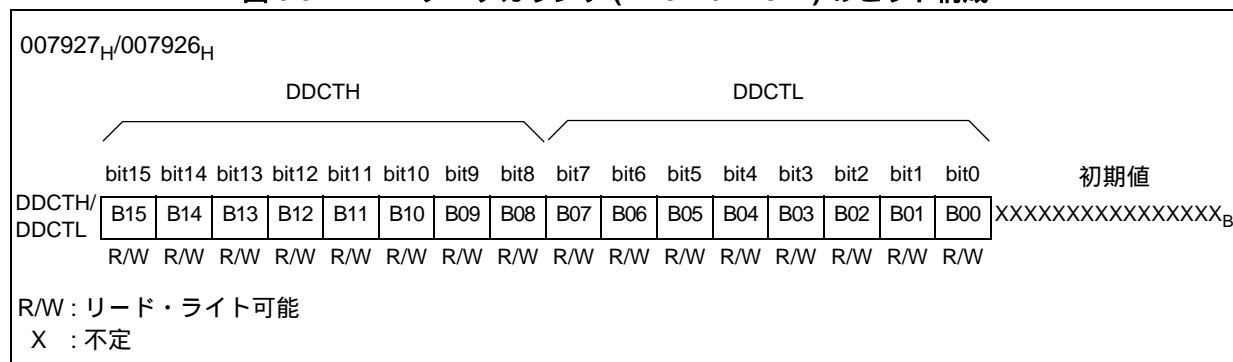
DMA データカウンタ (DDCTH/DDCTL) はデータ転送回数を設定するレジスタです。
DMA データカウンタ (DDCTH/DDCTL) が "0" になると DMA 転送は終了します。

■ DMA データカウンタ (DDCTH/DDCTL)

DMA データカウンタ (DDCTH/DDCTL) は、16 ビット長のレジスタで、転送回数に対応したカウンタとなります。各データの転送後に、カウンタはワード、バイト転送に関係なく1デクリメント(減値)します。このカウンタが"0"になるとDMA転送は終了します。図 3.8-7 に DDCT の構成を示します。

DDCT に "0" を設定すると、最大データ転送数 (65536 個) がセットされます。

図 3.8-7 DMA データカウンタ (DDCTH/DDCTL) のビット構成



■ DMA データカウンタ (DDCTH/DDCTL) の設定値について

転送バイト数と DMA データカウンタ (DDCTH/DDCTL) の関係は、表 3.8-2 のようになります。

表 3.8-2 DMA データカウンタ (DDCTH/DDCTL) の設定値

DMACS		DDCT
BW ビット	BYTEL ビット	
0	-	N
1	0	N/2
1	1	(N+1)/2

N: 転送バイト数

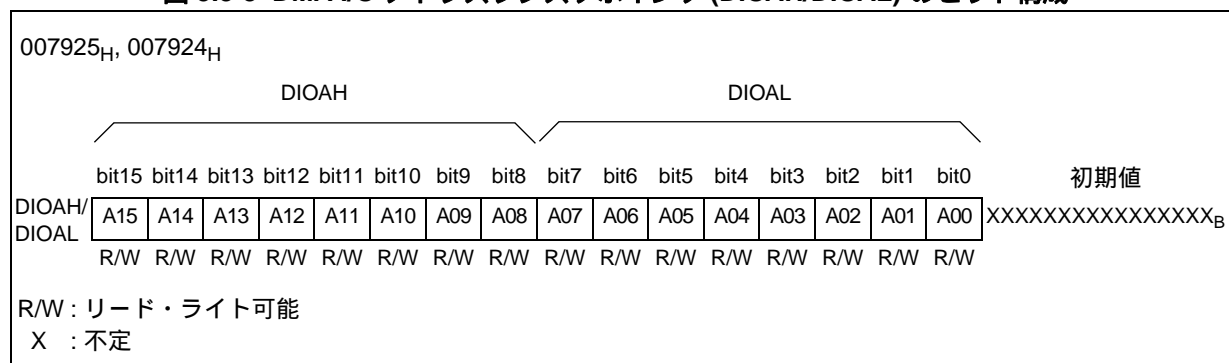
3.8.3.2 DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL)

DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL) は I/O アドレスポインタを設定するレジスタです。上位アドレス A23 ~ A16 は "00_H" 固定です。

■ DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL)

DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL) は、16 ビット長のレジスタで、DMA I/O レジスタアドレスの下位 16 ビット (A15 ~ A00) を示します。上位アドレス (A23 ~ A16) はすべて "0" であり、000000_H から 00FFFF_H 番地までの任意の I/O アドレス空間が指定できます。DMA コントロールレジスタ (DMACS) の IF ビット (DIOA 更新/固定選択ビット) で "更新あり" とした場合、DIOAH/DIOAL はバイト転送時には +1、ワード転送時には +2 変化し、"更新なし" とした場合、DIOAH/DIOAL は固定です。図 3.8-8 に DIOA の構成を示します。

図 3.8-8 DMA I/O アドレスレジスタポインタ (DIOAH/DIOAL) のビット構成



3.8.3.3 DMA コントロールレジスタ (DMACS)

DMA コントロールレジスタ (DMACS) は DMA 転送を制御するレジスタです。

DMA コントロールレジスタ (DMACS) によって

- 方向制御 (IOA BAP, BAP IOA)
- 転送ビット長 (バイト, ワード)
- アドレス更新 (あり, なし)
- 転送間隔
- ワード転送時の奇数バイト制御を制御できます。

■ DMA コントロールレジスタ (DMACS)

DMA コントロールレジスタ (DMACS) は, 8 ビット長で, DMA バッファアドレスポインタと DMA I/O レジスタアドレスポインタの更新・固定, 転送データ形式 (バイト / ワード), 転送方向, バイト転送指定, ウェイト指示を出します。図 3.8-9 に DMACS の構成を示します。

図 3.8-9 DMA コントロールレジスタ (DMACS) のビット構成

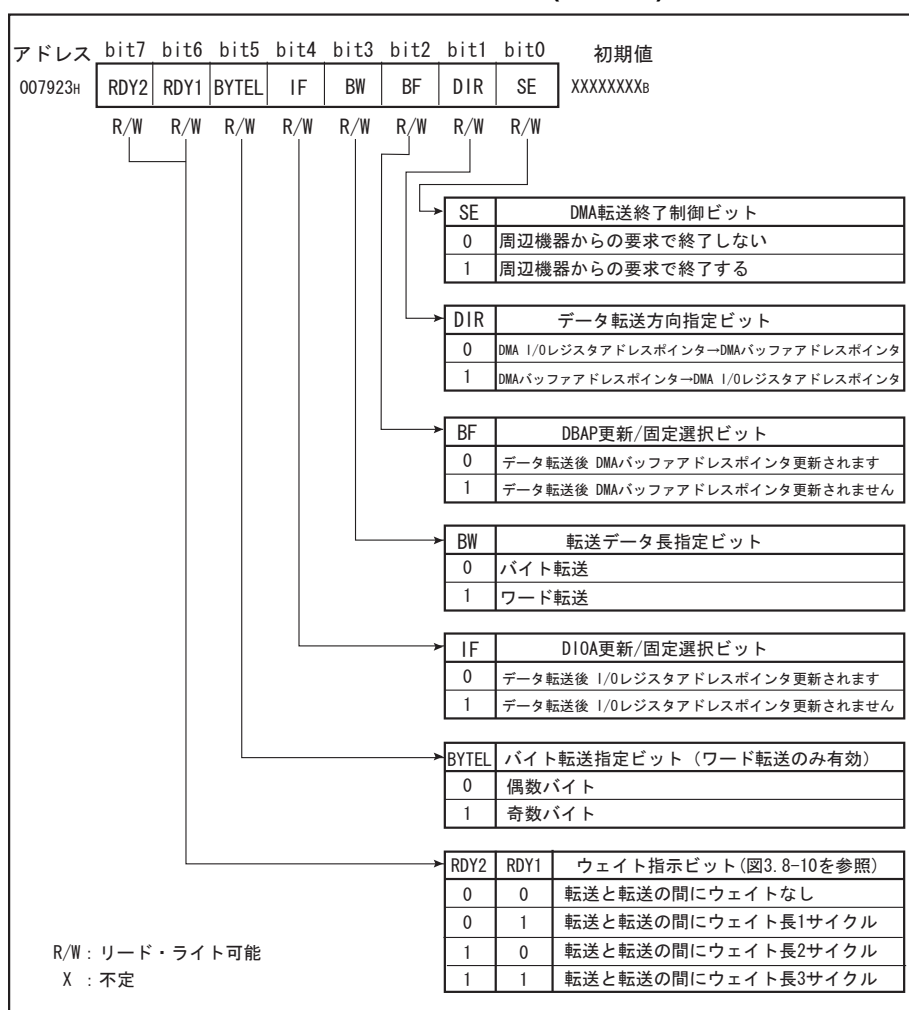
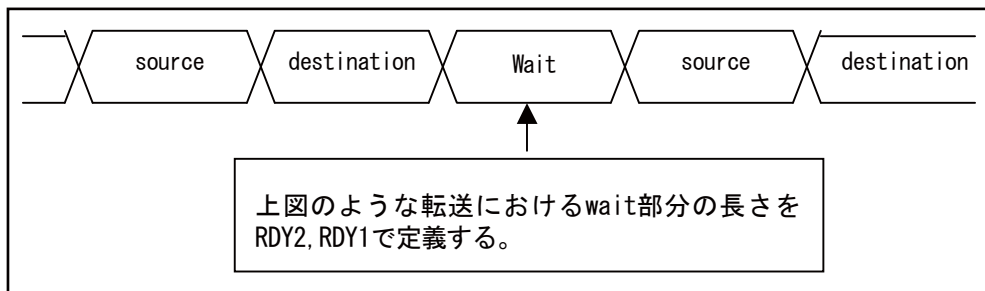


図 3.8-10 ウェイト指示ビット説明



< 注意事項 >

μ DMAC を使用して、UART へ送信データを書き込む場合、DMACS レジスタの RDY2, RDY1 ビットを (0, 0) に設定しないでください。

3.8.3.4 DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)

DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL) はバッファアドレスポインタを設定するレジスタです。DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL) は A23 ~ A00 まで設定可能です。

■ DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)

DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL) は、24ビットのレジスタで、DMA 転送で使用するアドレスを保管します。DBAP は、DMA の各チャネルに対してそれぞれ独立に存在するので、DMA の各チャネルは、16 M バイトの任意のアドレスと I/O の間でデータ転送できます。DMA コントロールレジスタ (DMACS) の BF ビット (DBAP 更新 / 固定選択ビット) を " 更新あり " と設定した場合、DBAP は下位 16 ビット (DBAPM、DBAPL) がバイト転送時には +1、ワード転送時には +2 変化し、上位 8 ビット (DBAPH) は変化しません。図 3.8-11 に、DMA バッファアドレスポインタ (DBAP) の構成を示します。

図 3.8-11 DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL) のビット構成

007922 _H /007921 _H /007920 _H										
	bit23	~	bit16	bit15	~	bit8	bit7	~	bit0	初期値
DBAPH/ DBAPM/ DBAPL	DBAPH				DBAPM				DBAPL	XXXXXXXXXXXXXXXXXXXXXXXXXXXX _B
	R/W				R/W				R/W	
R/W : リード・ライト可能										
X : 不定										

< 注意事項 >

- DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL) で指定できる領域は、000000_H ~ 00FFFF_H です。
- DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL) で指定できる領域は、000000_H ~ FFFFFFF_H です。
- DIOA と DBAP に μDMAC の内部レジスタ DCSR, DSRH, DSRL, DSSR, DERH, DERL, および DMA ディスクリプタウィンドウレジスタ DDWR のアドレスを指定しないでください。

3.8.4 μ DMAC の動作説明

This section describes the μ DMAC operation.

■ μ DMAC の動作

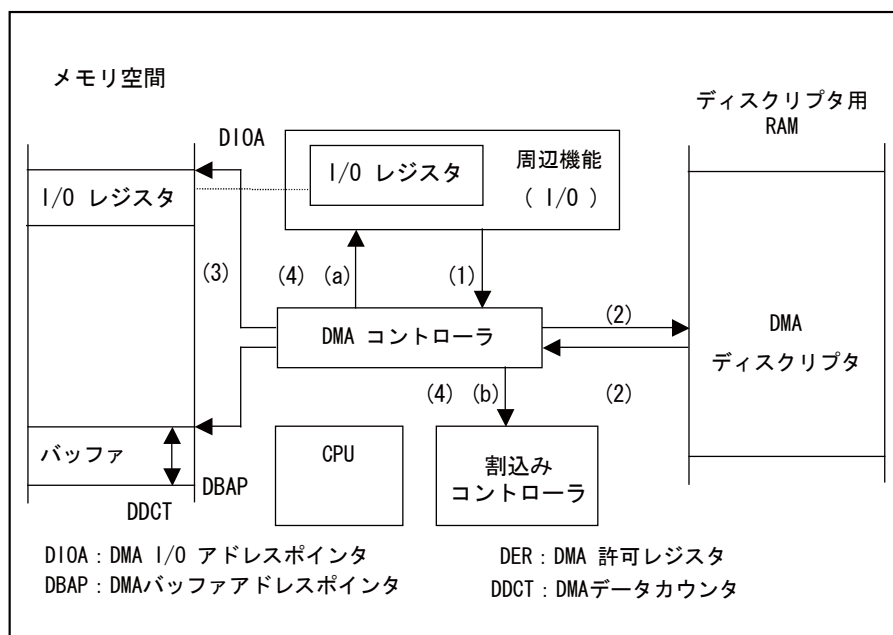
図 3.8-12 に DMA の動作の説明図を示します。

μ DMAC を使用したデータ転送は以下の順番で行われます。

1. 周辺リソース (I/O) が DMA 転送を要求する。
2. DMA 許可レジスタ (DERH/DERL) が "1" のとき、 μ DMAC が、指定されたチャンネルの転送元、転送先アドレスや転送回数等の転送データをディスクリプタから読み出す。
3. I/O とメモリ間で DMA データ転送を開始する。
4. 1 バイトもしくは 1 ワードの転送を実行後
 - (a) 転送終了でないとき (DMA データカウンタ DDCT は "0" になっていない)
周辺リソースに対して DMA 転送要求のクリアを要求する。
 - (b) 転送終了時 (DMA データカウンタ DDCT=0)
DMA 転送終了後、DMA ステータスレジスタに転送終了のフラグをセットし、割り込みコントローラへ割り込み要求を出力する。

< 注意事項 >

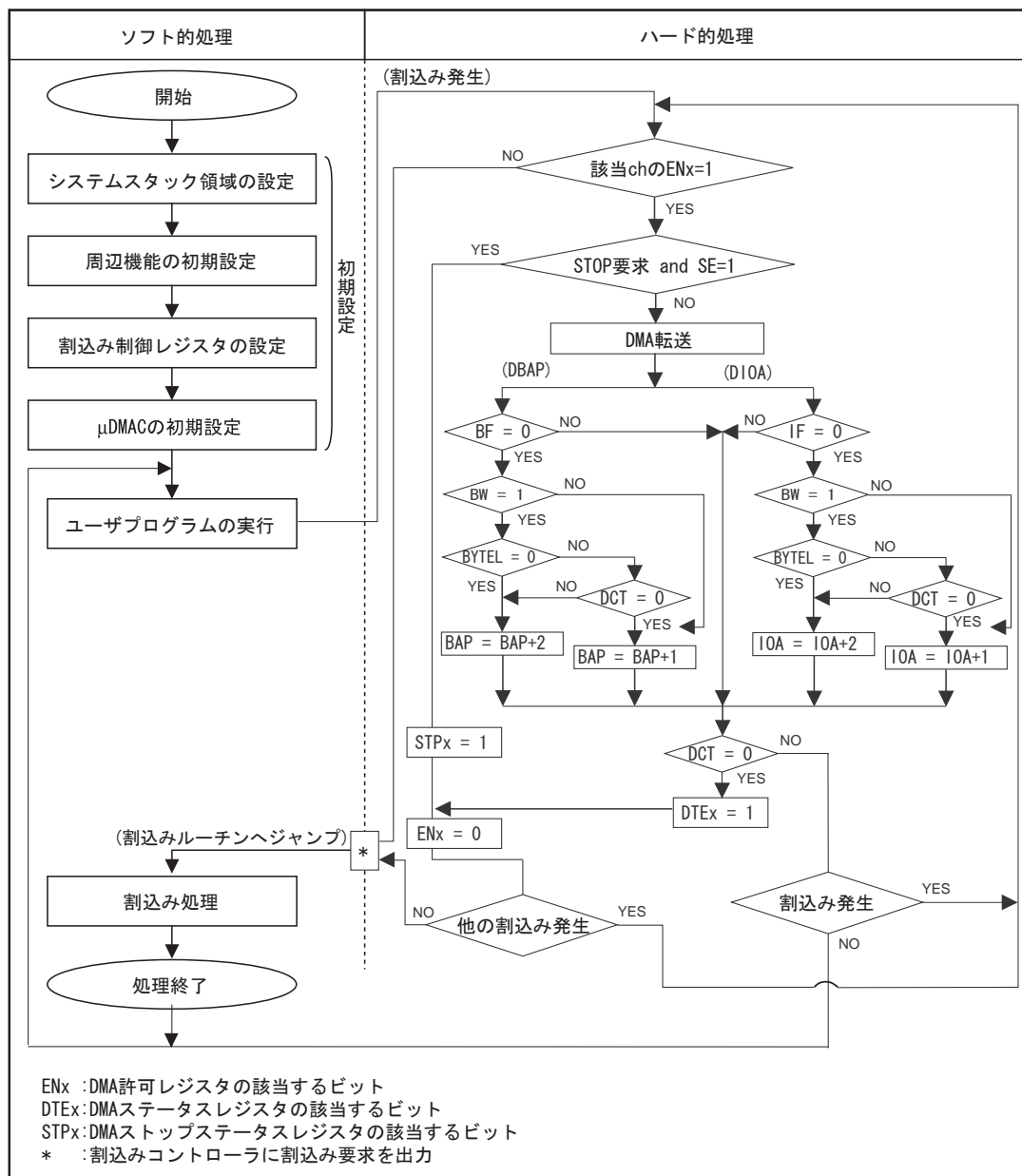
内部レジスタ DSRH, DSRL, DSSR, DERH, DERL へ書込みを行う場合、リードモディファイライト (RMW) 系の命令を使用してください。

図 3.8-12 μ DMAC の動作

■ μ DMAC 使用手順

図 3.8-13 に μ DMAC の使用手順を示します。

図 3.8-13 μ DMAC の使用手順



3.9 例外

F²MC-16LX では、以下の要因により例外が発生し、例外処理が行われます。

■ 未定義命令の実行

例外処理は、基本的には割込みと同じものであり、命令の境目で例外事項が発生したことを検出した段階で、通常処理から離れて例外処理を行います。一般的に、例外処理は予想外の動作を行った結果で発生するものですので、デバッグ時だけの使用や緊急時の復旧ソフトウェアの起動などに使用することをお勧め致します。

■ 未定義命令の実行による例外発生

F²MC-16LX では、命令マップで定義していないコードはすべて未定義命令として扱います。未定義命令を実行すると、ソフトウェア割込み命令の「INT10」と等価な処理を行います。すなわち、AL, AH, DPR, DTB, ADB, PCB, PC, PS の内容をシステムスタックに退避した後に、I フラグを "0" に設定し、S フラグを "1" に設定し、割込み番号 10 のベクタで示されるルーチンへ分岐します。スタックへ退避した PC の値は未定義命令を格納しているアドレスそのものです。2 バイト以上の命令コードでは、未定義であることが識別できたコードを格納しているアドレスになります。このため、RETI 命令で復帰させることは可能ですが、再び例外が発生するため、意味がありません。

3.10 割込み処理のスタック動作

割込みが受け付けられると、割込み処理に分岐する前に、専用レジスタの内容を自動的にシステムスタックに退避します。割込み処理終了時のスタックからの復帰も自動的に行われます。

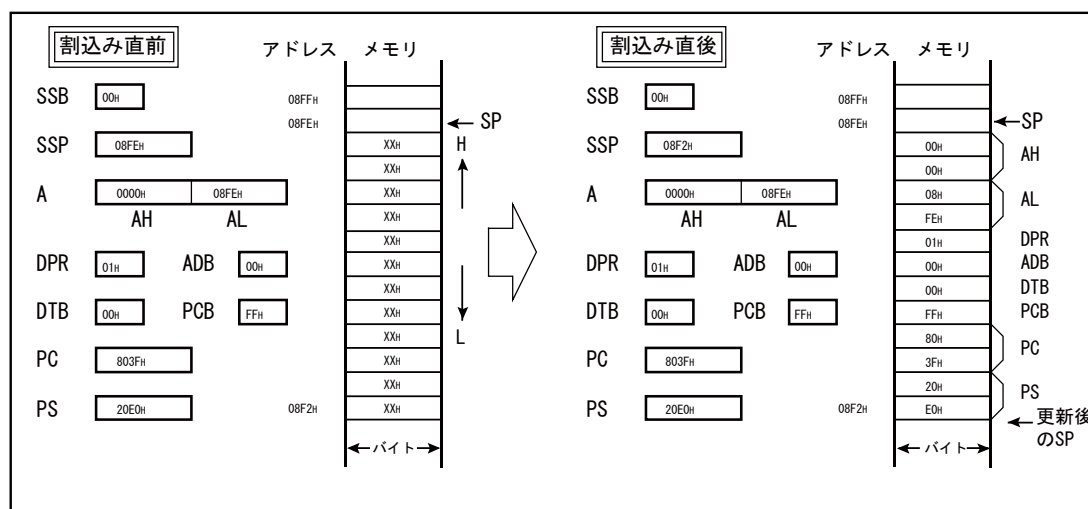
■ 割込み処理開始時のスタック動作

割込みが受け付けられると、CPUは現在の専用レジスタ類の内容を、次に示す順番で自動的にシステムスタックに退避させます。

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

図 3.10-1 に、割込み処理開始時のスタック動作を示します。

図 3.10-1 割込み処理開始時のスタック動作



■ 割込み処理復帰時のスタック動作

割込み処理終了時に割込み復帰命令 (RETI) を実行すると、割込み処理開始時と反対の順番で PS, PC, PCB, DTB, ADB, DPR, A の値がスタックから復帰し、専用レジスタは割込み開始直前の状態に戻ります。

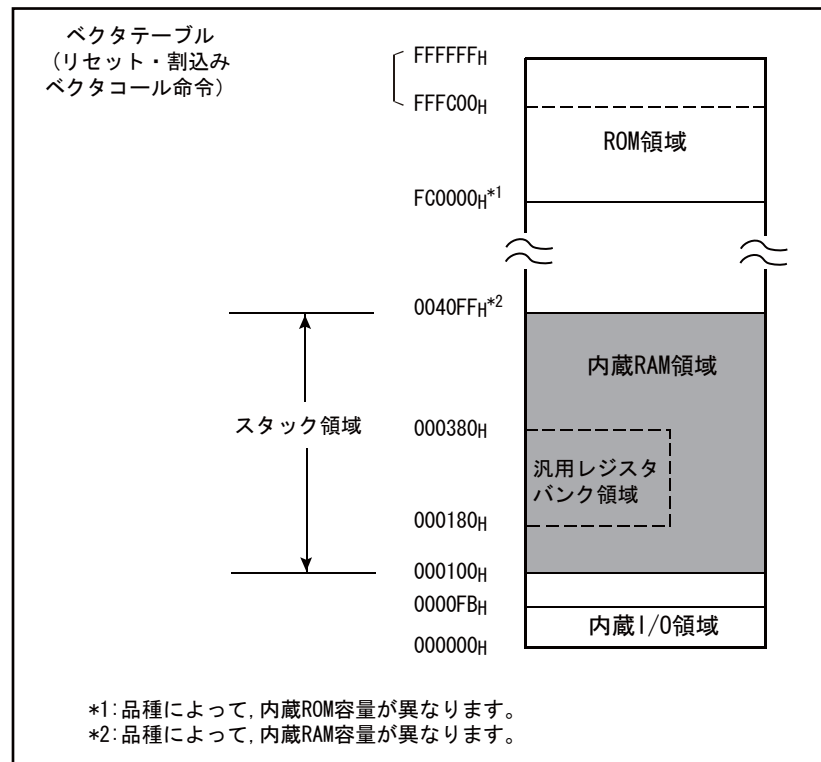
■ スタック領域

● スタック領域の確保

スタック領域は、割込み処理の他にサブルーチンコール命令 (CALL) やベクタコール命令 (CALLV) を実行するときのプログラムカウンタ (PC) の退避/復帰や、PUSHW, POPW 命令による一時的なレジスタ類の退避 / 復帰にも使われます。スタック領域は、データ領域とともに RAM 上に確保します。

図 3.10-2 にスタック領域を示します。

図 3.10-2 スタック領域



< 注意事項 >

- スタックポインタ (SSP, USP) にアドレスを設定する場合、原則として偶数アドレスを設定してください。
- システムスタック領域、ユーザスタック領域、およびデータ領域は、お互いに重なり合わないよう配置してください。

● システムスタックとユーザスタック

割込み処理では、システムスタック領域を使用します。割込み発生時にユーザスタック領域を使用していても、強制的にシステムスタックに切り換えられます。このため、ユーザスタック領域を主に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ておく必要があります。特に、スタック空間を分ける必要がなければ、システムスタックだけを使用してください。

3.11 割込み処理のプログラム例

割込み処理のプログラム例を示します。

■ 割込み処理のプログラム例

● 処理仕様

外部割込み 0(INT0) を利用した割込みプログラムの一例です。

● コーディング例

```
DDR6    EQU    000016H        ; ポート 6 方向レジスタ
ENIR     EQU    00003CH        ; 割込み /DTP 許可レジスタ
EIRR     EQU    00003DH        ; 割込み /DTP 要因レジスタ
ELVR     EQU    00003EH        ; 要求レベル設定レジスタ
ICR03    EQU    0000B3H        ; 割込み制御レジスタ 03
STACK    SSEG                  ; スタック
RW       100
STACK_T  RW       1
STACK    ENDS
;----- メインプログラム -----
CODE     CSEG                  ;
START:
        MOV    RP, #0          ; 汎用レジスタは先頭バンクを使用
        MOV    ILM, #07H       ; PS 内 ILM をレベル 7 に設定
        MOV    A, #!STACK_T    ; システムスタックの設定
        MOV    SSB, A
        MOVW   A, #STACK_T      ; スタックポインタの設定, このときは
        MOVW   SP, A            ; S フラグ =1 のため SSP にセットされる
        MOV    DDR6, #00000000B ; P60/INT0 端子を入力に設定
        OR     CCR, #40H        ; PS内CCRのIフラグをセットして割込み許可
        MOV    I:ICR03, #00H    ; 割込みレベル 0(最強) とする
        MOV    I:ELVR, #00000001B ; INT0 を "H" レベル要求とする
        MOV    I:EIRR, #00H     ; INT0 の割込み要因をクリア
        MOV    I:ENIR, #01H     ; INT0 の入力許可
        :
LOOP:    NOP                    ; ダミーループ
        NOP
        NOP
        NOP
```

```

        BRA    LOOP                ; 無条件ジャンプ
;----- 割込みプログラム -----
ED_INT1:
        MOV    I:EIRR, #00H        ; INT0 の新規受付禁止
        NOP
        NOP
        NOP
        NOP
        NOP
        NOP
        RETI                        ; 割込みから復帰
CODE    ENDS
;----- ベクタ設定 -----
VECT    CSEG    ABS=0FFH
        ORG    0FFB4H              ; 割込み #18(12H) にベクタを設定
        DSL    ED_INT1
        ORG    0FFDCH              ; リセットベクタ設定
        DSL    START
        DB     00H                 ; シングルチップモードに設定
VECT    ENDS
        END    START

```

● 拡張インテリジェント I/O サービス (EI²OS) のプログラム例処理仕様

1. INT0端子に入力される, 信号の"H"レベルを検出して拡張インテリジェントI/Oサービス (EI²OS) を起動
2. INT0 端子に "H" レベルが入力されると, EI²OS が起動されポート 0 のデータをメモリの 3000_H 番地に転送
3. 転送データバイト数は 100 バイトで, 100 バイト転送後, EI²OS 転送終了による割込みが発生

● コーディング例

```

DDR6    EQU    000016H            ; ポート 6 方向レジスタ
ENIR     EQU    00003CH            ; 割込み /DTP 許可レジスタ
EIRR     EQU    00003DH            ; 割込み /DTP 要因レジスタ
ELVR     EQU    00003EH            ; 要求レベル設定レジスタ
ICR03    EQU    0000B3H            ; 割込み制御レジスタ 03
BAPL     EQU    000100H            ; バッファアドレスポインタ下位
BAPM     EQU    000101H            ; バッファアドレスポインタ中位
BAPH     EQU    000102H            ; バッファアドレスポインタ上位
ISCS     EQU    000103H            ; EI2OS ステータス

```

```

IOAL    EQU    000104H    ; I/O アドレスポインタ下位
IOAH    EQU    000105H    ; I/O アドレスポインタ上位
DCTL    EQU    000106H    ; データカウンタ下位
DCTH    EQU    000107H    ; データカウンタ上位
ER0     EQU    EIRR:0     ; 外部割り込み要求フラグビットの定義
STACK   SSEG
        RW      100
STACK_T RW      1
STACK   ENDS
;----- メインプログラム -----
CODE    CSEG
START:

        AND     CCR, #0BFH    ; PS内CCRのIフラグをクリアして割り込み禁止
        MOV     RP, #00       ; レジスタバンクポインタを設定
        MOV     A, #!STACK_T   ; システムスタックの設定
        MOV     SSB, A
        MOVW    A, #STACK_T    ; スタックポインタの設定, このときは
        MOVW    SP, A          ; S フラグ =1 のため SSP にセットされる
        MOV     I:DDR6, #00000000B ; P60/INT0 端子を入力に設定
        MOV     BAPL, #00H     ; バッファアドレスをセット (003000H)
        MOV     BAPM, #30H
        MOV     BAPH, #00H
        MOV     ISCS, #00010001B ; I/O アドレス更新なし, バイト転送,
                                ; バッファアドレス更新あり
                                ; I/O バッファへ転送, 周辺機能による終了あり
        MOV     IOAL, #00H     ; 転送元アドレスをセット (ポート 0:000000H)
        MOV     IOAH, #00H
        MOV     DCTL, #64H     ; 転送バイト数をセット (100 バイト)
        MOV     DCTH, #00H
        MOV     I:ICR00, #00001000B ; EI2OS ch.0, EI2OS 許可, 割り込み
                                ; レベル 0 (最強)
        MOV     I:ELVR, #00000001B ; INT0 を "H" レベル要求とする
        MOV     I:EIRR, #00H    ; INT0 の割り込み要因クリア
        MOV     I:ENIR, #01H    ; INT0 の割り込み許可
        MOV     ILM, #07H       ; PS 内 ILM をレベル 7 に設定
        OR      CCR, #40H       ; PS内CCRのIフラグをセットして割り込み許可
        ;

```

```

LOOP:   BRA    LOOP           ;無限ループ
;----- 割込みプログラム -----
WARI    CLRB   ER0           ;割込み /DTP 要求フラグのクリア
      :
      ユーザ処理           ;EI2OS の終了要因をチェック,
      :                   ;バッファ中のデータの処理, EI2OS の再設定
                           など

RETI
CODE    ENDS
;----- ベクタ設定 -----
VECT    CSEG   ABS=0FFH
      ORG     0FFB4H         ;割込み #18(12H) にベクタを設定
      DSL     WARI
      ORG     0FFDCH         ;リセットベクタ設定
      DSL     START
      DB      00H           ;シングルチップモードに設定
VECT    ENDS
      END     START

```

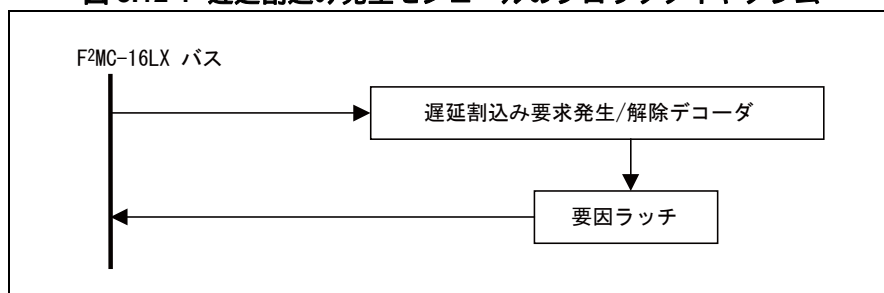
3.12 遅延割り込み発生モジュール

遅延割り込み発生モジュールは、タスク切換え用の割り込みを発生するためのモジュールです。本モジュールを使用することによってソフトウェアで F²MC-16LX CPU に対して割り込み要求の発生 / 取消しを行うことができます。

■ 遅延割り込み発生モジュールのブロックダイアグラム

図 3.12-1 に、遅延割り込み発生モジュールのブロックダイアグラムを示します。

図 3.12-1 遅延割り込み発生モジュールのブロックダイアグラム



■ 遅延割り込み発生モジュールのレジスタ一覧

遅延割り込み発生モジュール { 遅延割り込み要因発生 / 解除レジスタ (DIRR: Delayed Interrupt Request Register) } のビット構成は下図のとおりです。

図 3.12-2 遅延割り込み要因発生 / 解除レジスタ (DIRR) のビット構成

bit	15	14	13	12	11	10	9	8	初期値
00009F _H	-	-	-	-	-	-	-	R0	-----0 _B
	-	-	-	-	-	-	-	R/W	

R/W : リード・ライト可能

遅延割り込み要因発生 / 解除レジスタ (DIRR) は、遅延割り込み要因の発生 / 解除を制御するレジスタです。このレジスタに対して "1" を書き込むときには遅延割り込みの要求を発生させ、"0" を書き込むときには遅延割り込みの要求を解除します。リセット時には要因解除状態になります。予約ビット領域は "0", "1" どちらの書き込みでも結構ですが、将来の拡張を考慮してこのレジスタをアクセスする際には、セットビット、クリアビット命令を使用することをお勧めいたします。

3.12.1 遅延割り込み発生モジュールの動作

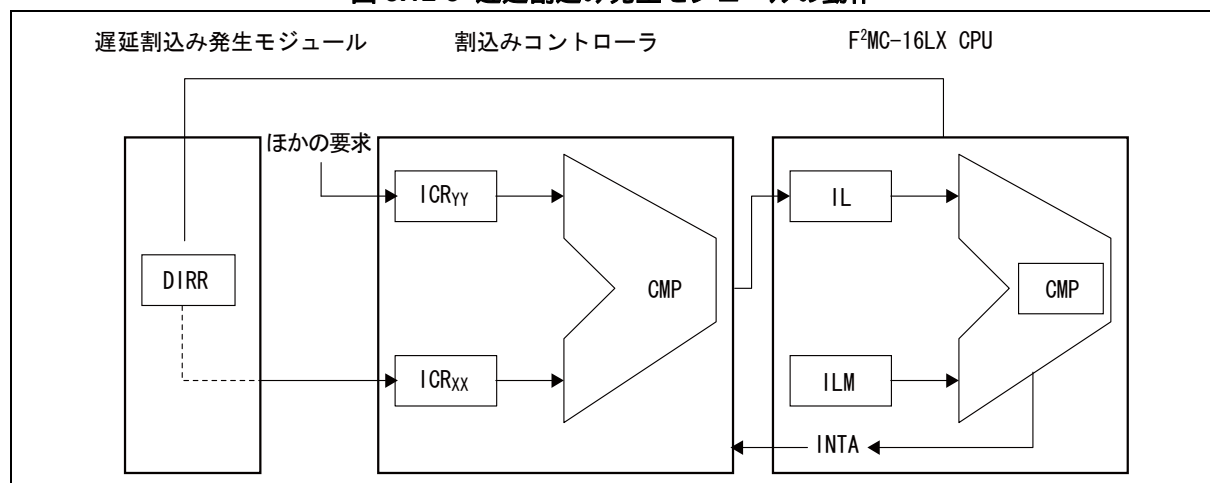
ソフトウェアで CPU が DIRR の該当ビットに "1" を書き込むと、遅延割り込み発生モジュールの中にある要求ラッチがセットされ、割り込みコントローラに割り込み要求を発生します。

■ 遅延割り込み発生モジュールの動作

ソフトウェアで CPU が DIRR の該当ビットに "1" を書き込むと、遅延割り込み発生モジュールの中にある要求ラッチがセットされ、割り込みコントローラに割り込み要求を発生します。ほかの割り込み要求が本割り込みより優先順位が低いまたはほかの割り込み要求がない場合に、割り込みコントローラは F²MC-16LX CPU に対して割り込みを発生します。F²MC-16LX CPU は自分の内部にある CCR レジスタ中の ILM ビットと割り込み要求を比較し、要求レベルが ILM ビットより高かったときに現在実行中の命令が終了し次第、ハードウェア割り込み処理マイクロプログラムを起動します。この結果、本割り込みに対する割り込みルーチンが実行されます。割り込み処理ルーチン内で DIRR の該当ビットに "0" を書き込むことによって本割り込み要因をクリアし、あわせてタスクの切換えを行います。

以上の動作の流れを、図 3.12-3 に示します。

図 3.12-3 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の使用上の注意 (遅延割り込み要求ラッチ)

このラッチは、DIRR の該当するビットに "1" を書き込むことによってセットされ、同じビットに "0" を書き込むことによってクリアされます。したがって、割り込み処理ルーチン内で、要因をクリアするようにソフトウェアを作成しておかないと、割り込み要因処理から復帰した途端に再割り込み処理を起動することになりますので、注意してください。

第4章

リセット

この章では、MB90330A シリーズのリセットについて説明します。

- 4.1 リセットの概要
- 4.2 リセット要因と発振安定待ち時間
- 4.3 外部リセット端子
- 4.4 リセット動作
- 4.5 リセット要因ビット
- 4.6 リセットによる各端子の状態

4.1 リセットの概要

リセット要因が発生すると、CPU は現在実行中の処理を直ちに中断し、リセット解除待ち状態になります。リセット解除後は、リセットベクタで示されたアドレスから処理を開始します。

リセットには、次の4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の発生
- ・ ソフトウェアリセット要求の発生

■ リセット要因

リセットの要因について表 4.1-1 に示します。

表 4.1-1 リセット要因

リセット	発生要因	マシンのクロック	ウォッチドッグタイマ	発振安定待ち
パワーオン	電源投入時	メインクロック (MCLK)	停止	あり
ウォッチドッグタイマ	ウォッチドッグタイマオーバフロー	メインクロック (MCLK)	停止	なし
外部端子	$\overline{\text{RST}}$ 端子への "L" レベル入力	メインクロック (MCLK)	停止	なし
ソフトウェア	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" を書き込む	メインクロック (MCLK)	停止	なし

メインクロック : 発振クロックの2分周クロック

● 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。評価品 / フラッシュ品の発振安定待ち時間は $2^{18}/\text{HCLK}$ (約 43.70 ms: 発振クロック 6 MHz 時)、マスク品の発振安定待ち時間は $2^{17}/\text{HCLK}$ (約 21.85 ms: 発振クロック 6 MHz 時) 固定です。発振安定待ち時間が経過した後、リセット動作を行います。

● ウォッチドッグリセット

ウォッチドッグリセットは、ウォッチドッグタイマの起動後、定められ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" の書き込みがない場合、ウォッチドッグタイマのオーバフローによってリセットを発生します。発振安定待ち時間はクロック選択レジスタ (CKSCR) で設定できます。

● 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" レベルを入力することによってリセットが発生します。 $\overline{\text{RST}}$ 端子への "L" レベル入力時間は、16 マシンサイクル ($16/\phi$) 以上必要です。

外部リセット ($\overline{\text{RST}}$ 端子入力リセット) では、発振安定待ち時間を取りません。

< 参考 >

$\overline{\text{RST}}$ 端子によるリセット要求が発生した場合に限り、書込み動作中 (転送系命令実行中 MOV 命令など) にリセット要因が発生した場合は、命令の終了後にリセット解除待ち状態になります。そのため、書込み中にリセットが入力されても、正常に書込みを終了します。

ただし、ストリング系命令 (MOVS 命令など) は指定したカウンタ分の転送が完了する前に、リセットを受け付けますので、すべてのデータが転送されることは保証されません。また、外部バスアクセス時に RDY 端子によるバスサイクルの延長が 16 マシンサイクル以上続いた場合もリセットを受け付けます。

● 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" を書き込むことによって、内部リセットが発生します。ソフトウェアリセットでは、発振安定待ち時間を取りません。

< 参考 >

クロックの定義

HCLK : 発振クロック (高速発振端子から供給されるクロック)

MCLK : メインクロック (HCLK の 2 分周クロック)

SCLK : サブクロック (低速発振端子から供給されるクロックの 4 分周)

ϕ : マシンクロック (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、「5.1 クロックの概要」を参照してください。

< 注意事項 >

ストップモードまたはサブクロックモードでリセットが発生した場合は、 $2^{17}/\text{HCLK}$ (約 21.85 ms: 発振クロック 6 MHz 時) の発振安定待ち時間を取ります。

クロックモードの詳細は、「5.4 クロックモード」を参照してください。

4.2 リセット要因と発振安定待ち時間

MB90330A シリーズには、4 種類のリセット要因があります。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それぞれ異なります。

■ リセット要因と発振安定待ち時間

表 4.2-1 に、リセット要因と発振安定待ち時間について示します。

表 4.2-1 リセット要因と発振安定待ち時間

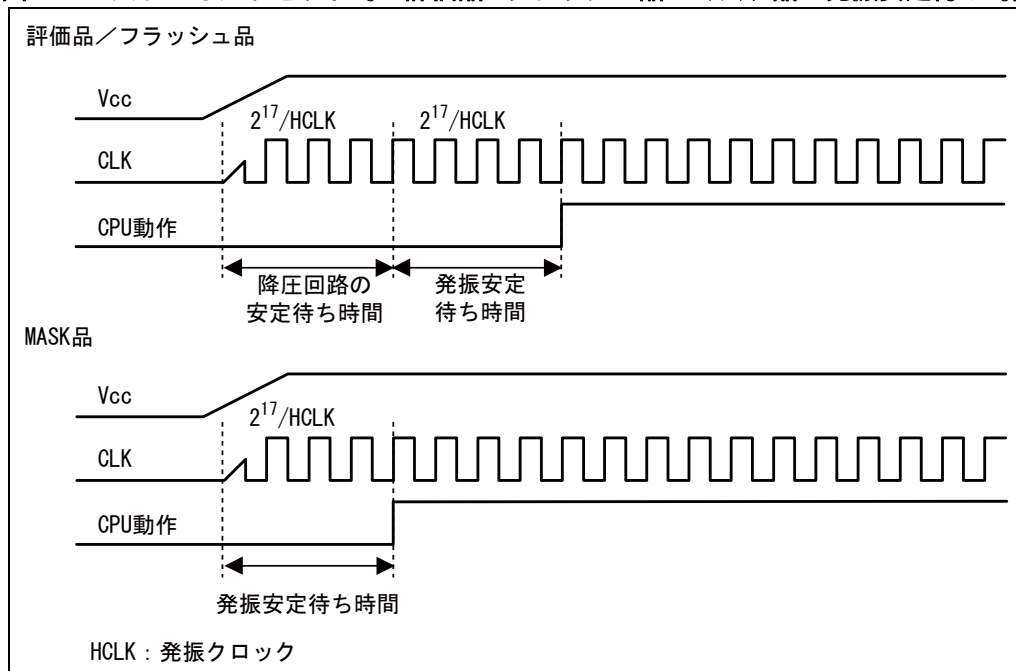
リセット要因	発振安定待ち時間 () 内は発振クロック 6 MHz 時
パワーオンリセット	評価品 / フラッシュ品 : $2^{18}/\text{HCLK}$ (約 43.70 ms) マスク品 : $2^{17}/\text{HCLK}$ (約 21.85 ms)
ウォッチドッグタイマ	なし : ただし、WS1, WS0 ビットは "11 _B " に初期化されます。
RST 端子からの外部リセット	なし : ただし、WS1, WS0 ビットは "11 _B " に初期化されます。
ソフトウェアリセット	なし : ただし、WS1, WS0 ビットは "11 _B " に初期化されます。

HCLK : 発振クロック

WS1, WS0: クロック選択レジスタ (CKSCR) の発振安定待ち時間選択ビット

図 4.2-1 に、パワーオンリセット時の評価品 / フラッシュ品とマスク品の発振安定待ち時間について示します。

図 4.2-1 パワーオンリセット時の評価品 / フラッシュ品とマスク品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、一般に数ミリ秒から十数ミリ秒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、「5.5 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。この場合、外部リセット入力解除されていなければ、外部リセット解除後にリセット動作を行います。

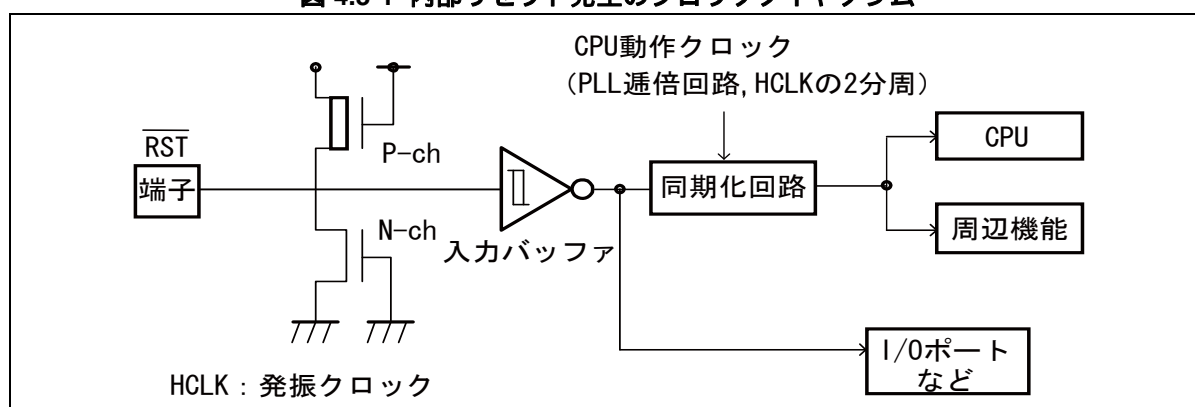
4.3 外部リセット端子

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。MB90330A シリーズは, CPU 動作クロックに同期してリセットが発生しますが, 外部端子 (ポートなど) のみ非同期でリセット状態に遷移します。

■ 外部リセット端子のブロックダイアグラム

図 4.3-1 に, 内部リセット発生 のブロックダイアグラムを示します。

図 4.3-1 内部リセット発生 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。

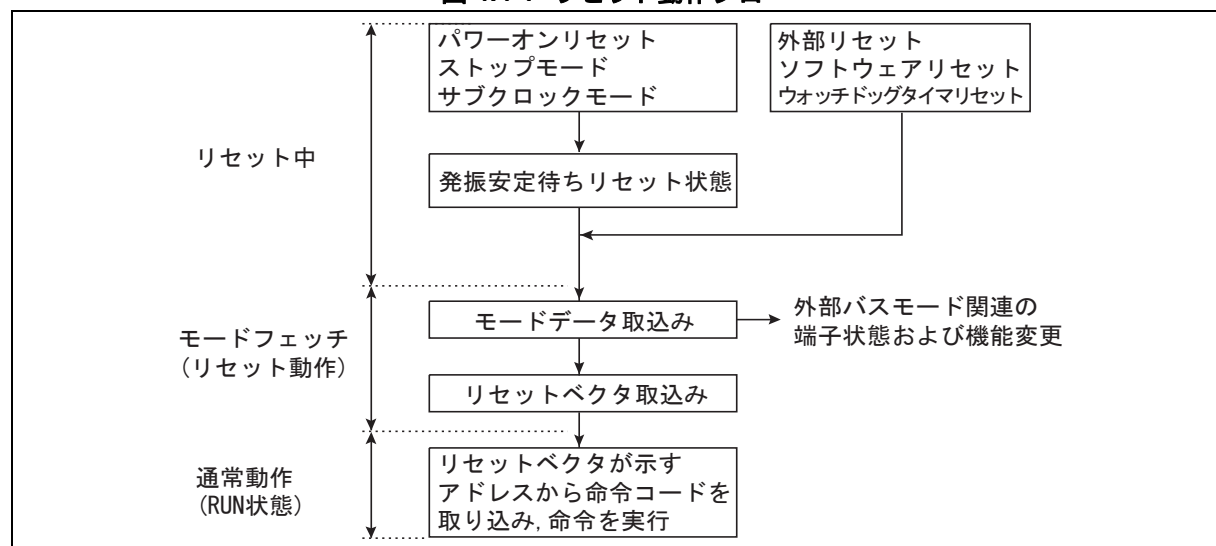
4.4 リセット動作

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。電源投入時および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

図 4.4-1 に、リセット動作フローを示します。

図 4.4-1 リセット動作フロー



■ モード端子

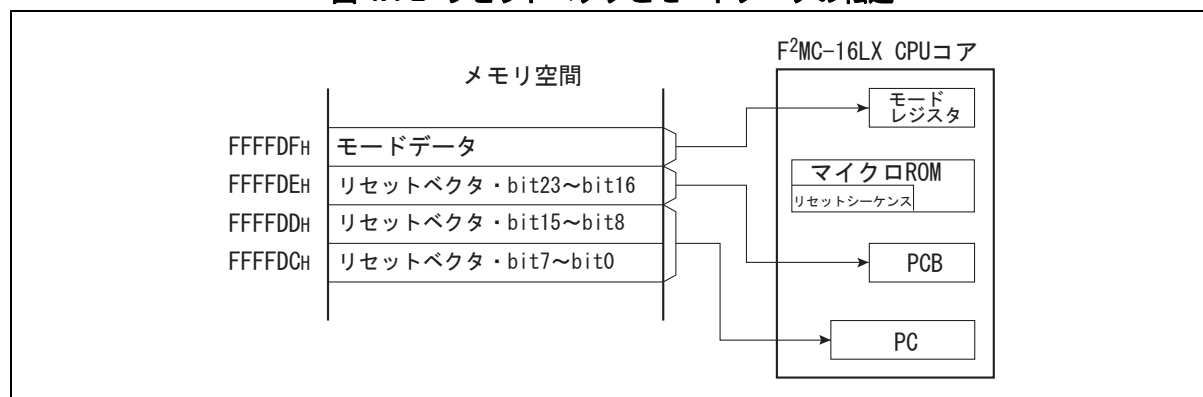
モード端子 (MD2 ~ MD0) は、リセットベクタとモードデータの取込み方法を指定します。リセットベクタとモードデータの取込みは、リセットシーケンスで行います。モード端子の詳細は、「7.2 モード端子 (MD2 ~ MD0)」を参照してください。

■ モードフェッチ

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 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 4.4-2 に、リセットベクタとモードデータの転送を示します。

図 4.4-2 リセットベクタとモードデータの転送



< 参考 >

リセットベクタとモードデータを、内部 ROM または外部メモリのどちらから読み出すかは、モード端子によって指定します。モード端子で外部ベクタモードを指定すると、内部 ROM ではなく、外部メモリからリセットベクタとモードデータを読み出しますので、シングルチップモードおよび内部 ROM 外部バスモードで 사용되는場合には、モード端子で内部ベクタモードを指定することを推奨します。

● 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、「7.3 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書き込んでおきます。この内容のアドレスから実行を開始します。

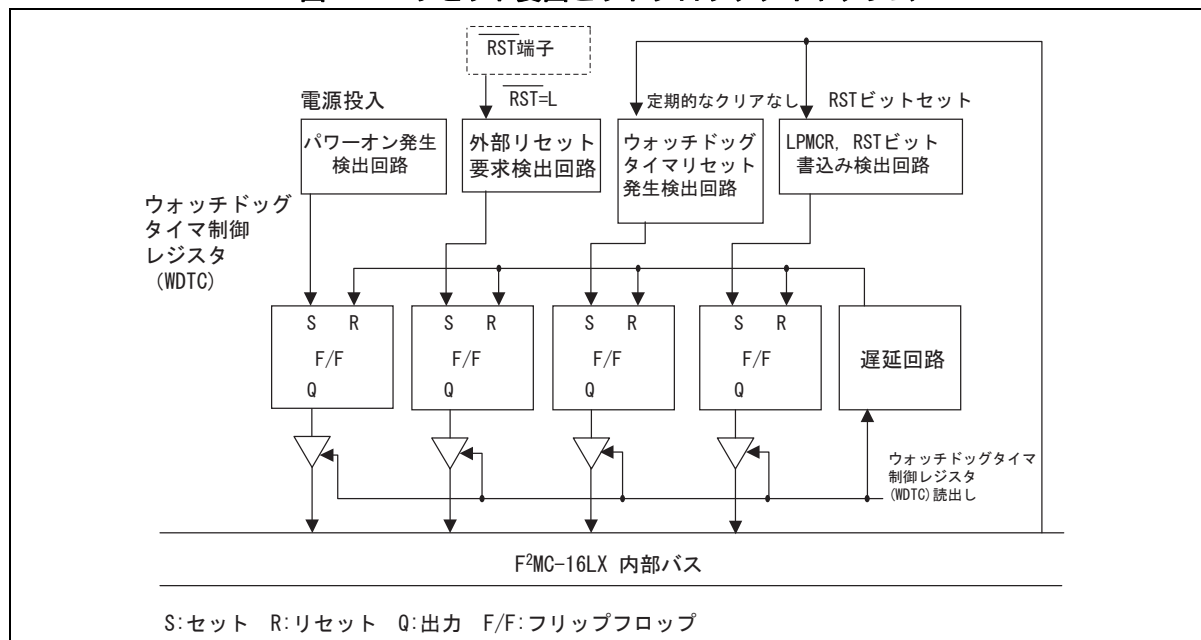
4.5 リセット要因ビット

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) を読み出すことによって識別できます。

■ リセット要因ビット

図 4.5-1 に示すように、それぞれのリセット要因には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、WDTC レジスタ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。

図 4.5-1 リセット要因ビットブロックダイアグラム



■ リセット要因ビットとリセット要因の対応

ウォッチドッグタイマ制御レジスタ(WDTC)のリセット要因ビットの構成を図 4.5-2に、リセット要因ビットの内容とリセット要因の対応を表 4.5-1 に示します。

詳細は、「10.2 ウォッチドッグタイマ制御レジスタ (WDTC)」のビット構成を参照してください。

図 4.5-2 リセット要因ビットの構成 (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)									
bit	15-8	7	6	5	4	3	2	1	0
0000A8 _H	(TBTC)	PONR	予約	WRST	ERST	SRST	WTE	WT1	WT0
		X	X	X	X	X	1	1	1
		R	-	R	R	R	W	W	W

初期値 R/W

R: リードオンリ W: ライトオンリ X: 不定

表 4.5-1 リセット要因ビットの内容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の発生	1	X	X	X
ウォッチドッグタイマのオーバフロー発生	*	1	*	*
RST 端子からの外部リセット要求発生	*	*	1	*
ソフトウェアリセット要求の発生	*	*	*	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、WDTC レジスタ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、リセット要因ビットのうち ERST ビットと WRST ビットが "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時には、リセット要因ビットのうち 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、WDTC レジスタを読み出した場合のみクリアされます。一度発生したリセット要因に対応するビットは、リセットが発生してもクリアされません ("1" のままです)。

< 注意事項 >

パワーオンリセットが発生しない条件で電源を立ち上げた場合、WDTC レジスタの値は保証されません。

4.6 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。

■ リセット中の端子の状態

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

- 内部ベクタモード設定時 (MD2 ~ MD0=011_B)

I/O 端子 (リソース端子) 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。

リセット中の各端子の状態については、「6.7 スタンバイモード、ホールド、リセット時の端子状態」を参照してください。

■ モードデータ読出し後の端子の状態

モードデータ読出し後の端子の状態は、モードデータ (M1, M0) によって決定されます。

- シングルチップモード選択時 (M1, M0=00_B)

I/O 端子 (リソース端子) 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場合に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。

第5章

クロック

この章では、MB90330A シリーズのクロックについて説明します。

- 5.1 クロックの概要
- 5.2 クロック発生部のブロックダイアグラム
- 5.3 クロック選択レジスタ (CKSCR)
- 5.4 クロックモード
- 5.5 発振安定待ち時間
- 5.6 振動子と外部クロックの接続

5.1 クロックの概要

クロック発生部は、CPU や周辺機能の動作クロックである内部クロックの動作を制御します。本書では、クロックの種類に応じて以下のようによびます。

- マシナクロック: 内部クロックのことをいいます。
- メインクロック: 発振クロックの2分周クロックをいいます。
- 発振クロック : 高速発振端子から供給されるクロックをいいます。
- PLL クロック : 内部の PLL 発振によるクロックをいいます。
- サブクロック : 低速発振端子から供給されるクロックの4分周をいいます。

■ クロックの概要

クロック発生部は発振回路を内蔵しており、外部に振動子を接続することによって、発振クロックおよびサブクロックを発生します。外部で生成したクロックを入力して、発振クロックにすることもできます。また、PLL クロック通倍回路を内蔵しており、発振クロックの通倍クロックを3種類発生できます。クロック発生部では、発振安定待ち時間の制御、PLL クロック通倍制御およびクロックセクタでのクロックの切換えによる、内部クロックの動作制御を行います。

● 発振クロック (HCLK)

高速発振端子に振動子を接続するか、外部クロックを入力して発生させたクロックです。

● サブクロック (SCLK)

時計タイマを動作させるクロックです。また、低速のマシナクロックとしても使用可能です。

低速発振端子に振動子を接続するか、外部クロックを入力して発生させたクロックの4分周です。

● 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とクロックセクタへの入力クロックとなります。

● PLL クロック (PCLK)

発振クロックを、内蔵の PLL クロック通倍回路 (PLL 発振回路) により通倍したクロックです。3種類のクロックを選択できるようになっています。

● マシナクロック (ϕ)

CPU と周辺機能の動作クロックです。このクロックの1周期を、マシナサイクル ($1/\phi$) としています。メインクロック (発振クロックの2分周クロック)、サブクロックと、3種類の通倍クロックの中から1種類を選択できるようになっています。

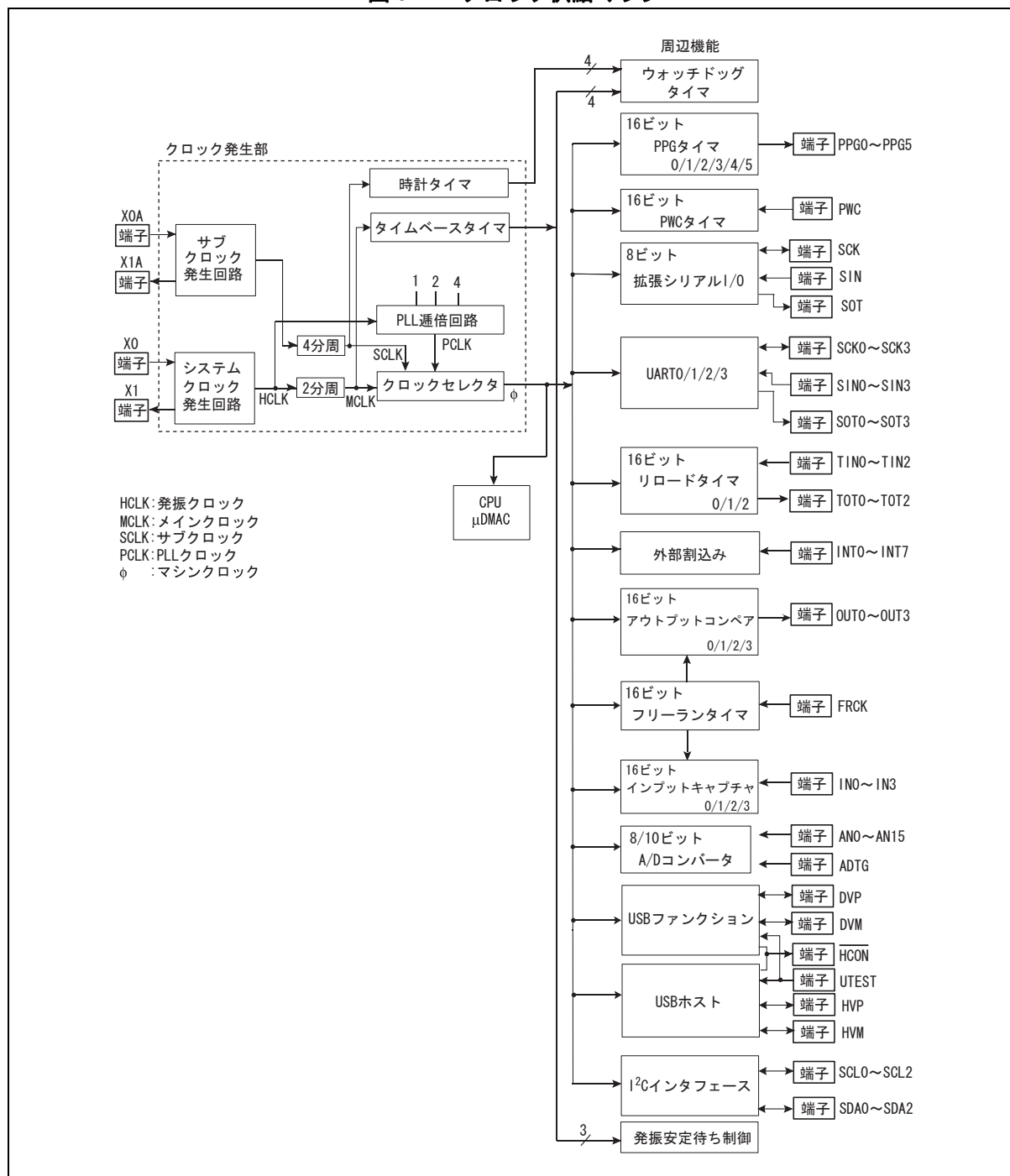
< 注意事項 >

- 発振クロックは 1 MHz ~ 7 MHz の発振が可能です。CPU および周辺機能は最大動作周波数が 24 MHz です。最大動作周波数を超える逡倍率を指定した場合、デバイスは正常に動作しません。例えば、原発振が 6 MHz で発振している場合には、最大 4 逡倍の指定が可能です。
 - USBホスト, USBファンクション使用時はこの6 MHz 4逡倍の設定が必要となります。
-

■ クロック供給マップ

CPU や周辺機能の動作クロックとして、クロック発生部で生成したマシクロックを供給しています。このため、CPU と周辺機能の動作は、メインクロックと PLL クロックの切換え（クロックモード）や PLL クロック逡倍率切換えの影響を受けます。一部の周辺機能へは、タイムベースタイマの分周出力が供給されており、各周辺で、その動作クロックを選択できるようになっています。図 5.1-1 に、クロック供給マップを示します。

図 5.1-1 クロック供給マップ



5.2 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6つのブロックで構成されています。

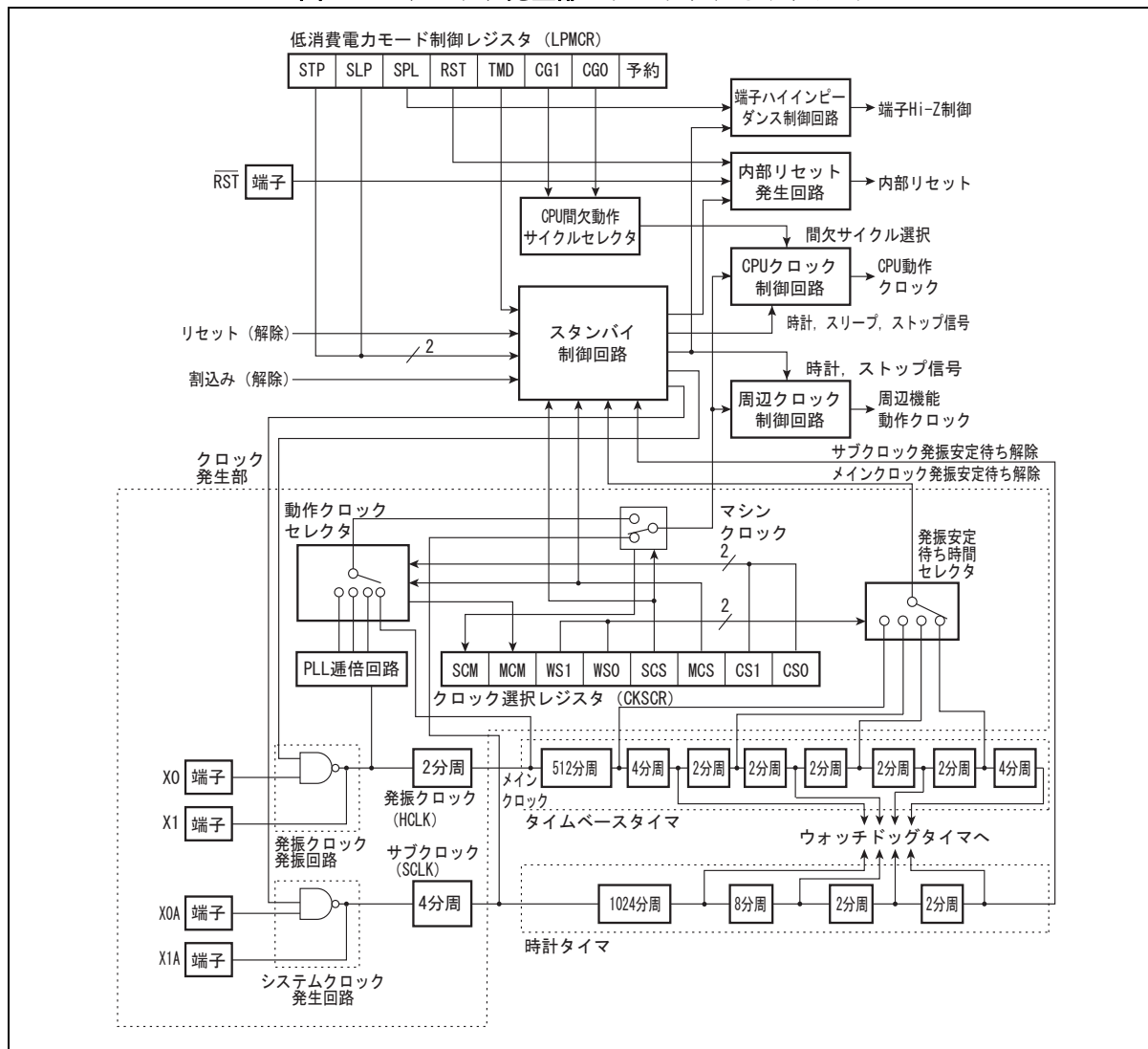
- システムクロック発生回路
- サブクロック発生回路
- PLL 逡倍回路
- クロックセクタ
- クロック選択レジスタ (CKSCR)
- 発振安定待ち時間セクタ

■ クロック発生部のブロックダイアグラム

図 5.2-1 に、クロック発生部のブロックダイアグラムを示します。

なお、図 5.2-1には、スタンバイ制御回路、タイムベースタイマの回路も含まれています。

図 5.2-1 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

高速発振端子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。外部クロックを入力することもできます。

● サブクロック発生回路

低速発振端子に接続した振動子によって、サブクロック (SCLK) を発生します。外部クロックを入力することもできます。

● PLL 通倍回路

PLL 発振で発振クロックを通倍し、CPU クロックセクタへ供給します。

● クロックセクタ

メインクロック、サブクロックと、3種類の PLL クロックから、CPU 系クロック制御回路および周辺系クロック制御回路へ供給されるクロックを選択します。

● クロック選択レジスタ (CKSCR)

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 クロックの通倍率の選択などを行います。

● 発振安定待ち時間セクタ

ストップモードの解除時およびウォッチドッグリセット時の、発振クロックの発振安定待ち時間を選択する回路です。4種類のタイムベースタイマ出力を選択します。

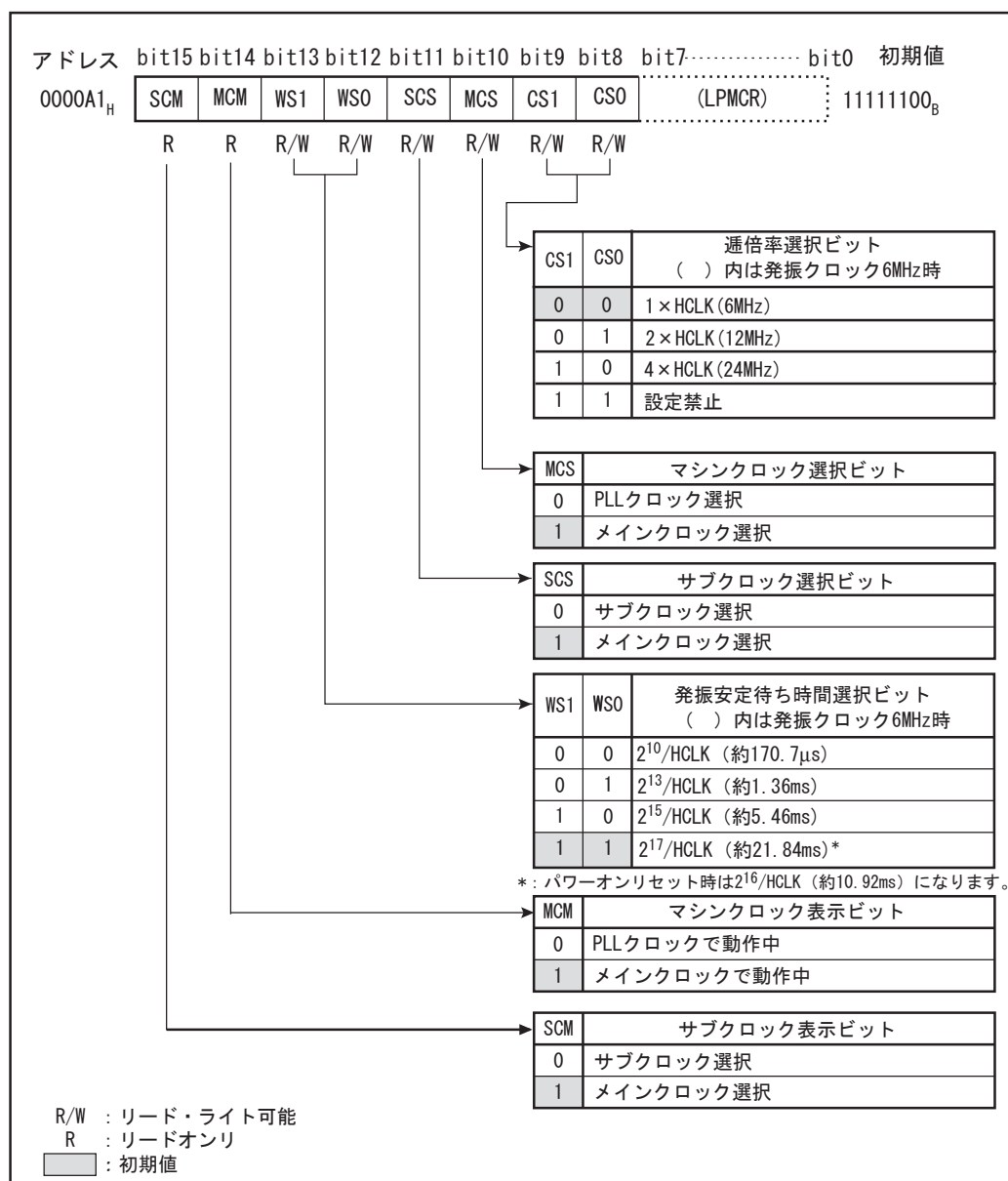
5.3 クロック選択レジスタ (CKSCR)

クロック選択レジスタ (CKSCR) は、メインクロック、サブクロックと 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。

■ クロック選択レジスタ (CKSCR) の構成

図 5.3-1 にクロック選択レジスタ (CKSCR) のビット構成を、表 5.3-1 にクロック選択レジスタの各ビットの機能について説明します。

図 5.3-1 クロック選択レジスタ (CKSCR) のビット構成



< 注意事項 >

マシナクロック選択ビット (MCS) は、リセットによってメインクロック選択に初期化されます。

表 5.3-1 クロック選択レジスタ (CKSCR) の各ビットの機能 (1 / 2)

ビット名		機能
bit15	SCM: サブクロック 表示ビット	<ul style="list-style-type: none"> ・マシナクロックとして、メインクロックまたはサブクロックのどちらが選択されているかを表示するビットです。 ・"0" の場合、サブクロックが選択されていることを示し、"1" の場合、メインクロックまたは PLL クロックが選択されていることを示します。 ・SCS=1 で、かつ SCM=0 ならば、メインクロック発振安定待ち時間中であることを示します。
bit14	MCM: PLL クロック 表示ビット	<ul style="list-style-type: none"> ・マシナクロックとして、メインクロックまたは PLL クロックのどちらが選択されているかを表示するビットです。 ・"0" の場合、PLL クロックが選択されていることを示し、"1" の場合、メインクロックまたはサブクロックが選択されていることを示します。 ・PLL クロック選択ビット (MCS)=0 で、かつ MCM=1 ならば、PLL クロック発振安定待ち時間中であることを示します。
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時、サブクロックモードからメインクロックモードへの移行時およびサブクロックモードから PLL クロックモードへの移行時の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 "11_B" に初期化されます。 <p>(注意事項) 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。「 4.2 リセット要因と発振安定待ち時間 」を参照してください。"00_B" の設定はメインクロックモード時のみ設定してください。</p> <p>(参考) 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。</p>

表 5.3-1 クロック選択レジスタ (CKSCR) の各ビットの機能 (2 / 2)

ビット名		機能
bit11	SCS: サブクロック 選択ビット	<ul style="list-style-type: none"> ・マシニングロックとして、メインクロックまたはサブクロックのどちらを選択するかを指定するビットです。 ・"0" のときサブクロックを選択し、"1" のときメインクロックを選択します。 ・"0" のときに "1" を書き込むと、メインクロックの発振安定待ち時間が発生するために、自動的にタイムベースタイマがクリアされます。 ・サブクロック選択時の動作クロックは、サブクロックを使用します (低速発振 32 kHz 時、マシニングロックは 8 kHz になります)。 ・SCS, MCS がともに "0" のときは、SCS が優先され、サブクロックが選択されます。 ・すべてのリセット要因で "1" に初期化されます。
bit10	MCS: PLL クロック 選択ビット	<ul style="list-style-type: none"> ・マシニングロックとして、メインクロックまたは PLL クロックのどちらを選択するかを指定するビットです。 ・"0" の場合には PLL クロックを選択し、"1" の場合にはメインクロックを選択します。 ・"1" の場合に "0" を書き込むと、PLL クロックの発振安定待ち時間が発生するために、自動的にタイムベースタイマがクリアされます。さらにタイムベースタイマ制御レジスタ (TBTC) の割込み要求フラグビット (TBOF) 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 6 MHz 時、発振安定待ち時間は約 2.73 ms になります)。 ・メインクロック選択時の動作クロックは、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 6 MHz 時、動作クロックは 3 MHz になります)。 ・すべてのリセット要因で "1" に初期化されます。 <p>(注意事項)</p> <p>MCS ビットが "1" の場合に "0" を書き込む場合には、TBTC レジスタの割込み要求許可ビット (TBIE) または割込みレベルマスクレジスタ (ILM) によって、タイムベースタイマ割込みがマスクされている状態で行うようにしてください。</p>
bit9, bit8	CS1, CS0: 逡倍率選択 ビット	<ul style="list-style-type: none"> ・PLL クロックの逡倍率を選択するビットです。 ・3 種類の逡倍率から選択できます。 ・すべてのリセット要因で "00_B" に初期化されます。 <p>(注意事項)</p> <p>MCS ビット、または MCM ビットが "0" の場合には、書込みが抑止されます。一度、MCS ビットを "1" に設定 (メインクロックモード) 後に、CS1, CS0 ビットを書き換えてください。"11_B" は設定禁止です。</p>

HCLK: 発振クロック

5.4 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、PLL クロックモード、サブクロックモードがあります。

■ メインクロックモード、PLL クロックモード、サブクロックモード

● メインクロックモード

メインクロックモードは、CPU および周辺リソースの動作クロックとして発振クロックを2分周したクロックを使用し、PLL クロックを停止させます。

● PLL クロックモード

PLL クロックモードは、CPU および周辺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) によって選択できます。

< 注意事項 >

USB ホスト、USB ファンクション使用時は PLL クロックモードで動作させる必要があります。

● サブクロックモード

サブクロックモードは、CPU および周辺リソースの動作クロックとしてサブクロックを使用し、メインクロックおよび PLL クロックを停止させます。

■ クロックモードの移行

CKSCR レジスタの PLL クロック選択ビット (MCS)、サブクロック選択ビット (SCS) への書き込み動作によって、クロックモードは、メインクロックモード、PLL クロックモード、サブクロックモードに移行します。

● 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、CKSCR レジスタの 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時間 ($2^{14}/HCLK$) 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、CKSCR レジスタの 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 ~ 8 PLL 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。

● メインクロックモードからサブ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、CKSCR レジスタの SCS ビットを "1" から "0" に書き換えた場合には、メインクロックからサブクロックに切り換わります。

● サブクロックモードからメインクロックモードへの移行

サブクロックモードの状態では、CKSCR レジスタの SCS ビットを "0" から "1" に書き換えた場合には、メインクロックの発振安定待ち時間後に、サブクロックからメインクロックに切り換わります。発振安定待ち時間は、CKSCR レジスタの発振安定待ち時間選択ビット (WS1, WS0) によって選択します。

● PLL クロックモードからサブ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) のサブクロック選択ビット (SCS) を "1" から "0" に書き換えた場合には、PLL クロックからサブクロックに切り換わります。

● サブクロックモードから PLL クロックモードへの移行

サブクロックモードの状態では、CKSCR レジスタの SCS ビットを "0" から "1" に書き換えた場合には、メインクロックの発振安定待ち時間後に、サブクロックから PLL クロックに切り換わります。発振安定待ち時間は、CKSCR レジスタの発振安定待ち時間選択ビット (WS1, WS0) によって選択します。

< 注意事項 >

CKSCR レジスタの PLL クロック選択ビット (MCS), SCS ビットを書き換えても即座にはマシナクロックは切り換えられません。マシナクロックに依存するリソースを操作する場合には、CKSCR レジスタの PLL クロック表示ビット (MCM) およびサブクロック表示ビット (SCM) を参照してマシナクロックの切り換えが行われたことを確認した後に、リソースの操作をしてください。

SCS ビット, MCS ビットがともに "0" の場合は、SCS が優先され、サブクロックモードになります。

クロックモードを切り換えた場合、切り換えが完了するまでは、ほかのクロックモードおよび低消費電力モードへの切り換えを行わないでください。切り換えの完了は、クロック選択レジスタ (CKSCR) の MCM ビットおよび SCM ビットを参照して確認してください。

■ PLL クロック逡倍率の選択

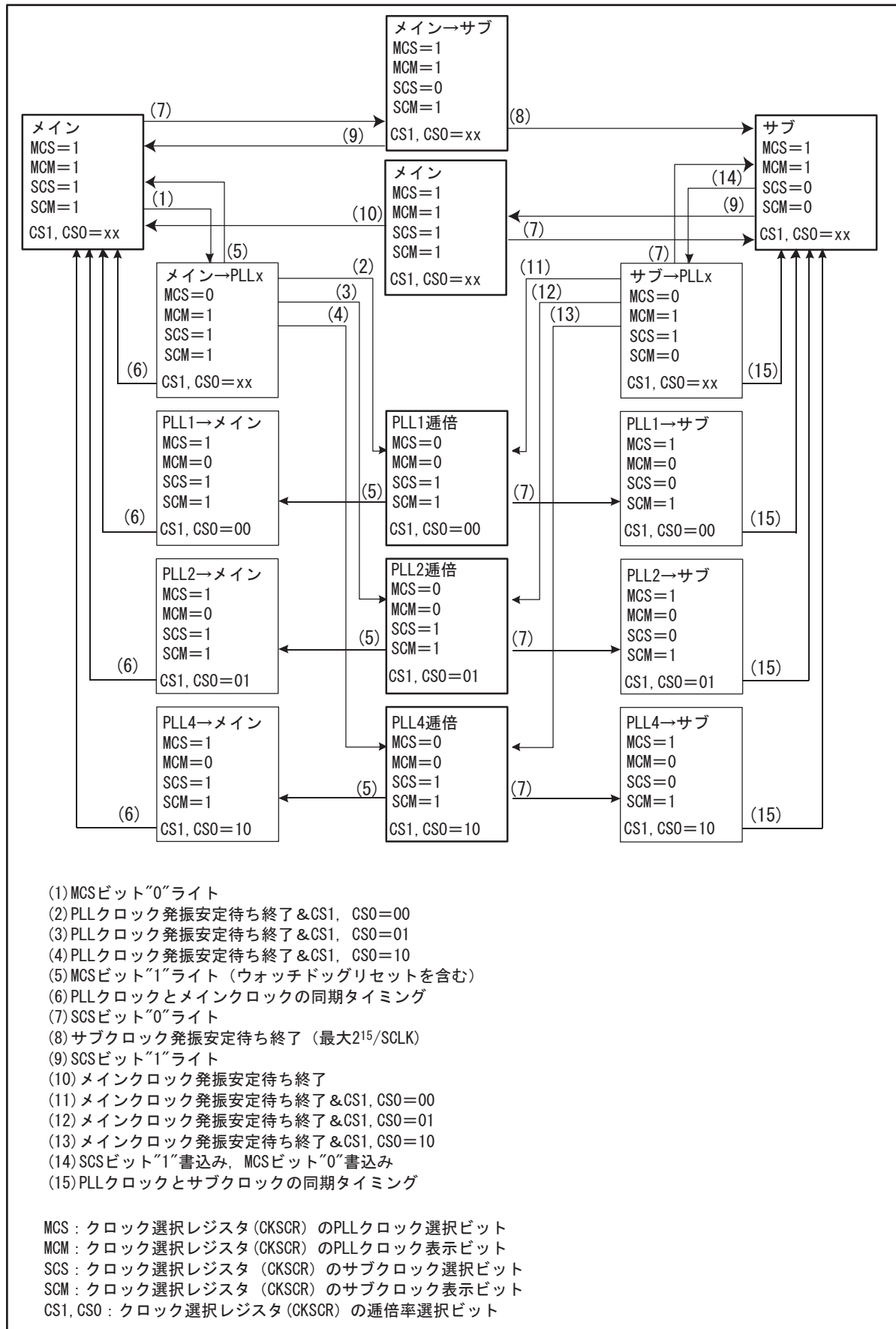
CKSCR レジスタの CS1, CS0 ビットに "00_B" ~ "10_B" を書き込むことによって、1 ~ 4 逡倍の 3 種類の PLL クロック逡倍率を選択できます。"11_B" は設定禁止です。

■ マシナクロック

PLL 逡倍回路から出力される PLL クロック、メインクロック、サブクロックが、マシナクロックとなります。このマシナクロックが CPU および周辺機能に供給されます。メインクロック、PLL クロック、サブクロックは、CKSCR レジスタの MCS ビットまたは SCS ビットへの書込みにより、いずれかを選択できます。

図 5.4-1 に、マシナクロック選択の状態遷移図を示します。

図 5.4-1 マシナクロック選択の状態遷移図



< 注意事項 >

マシナクロックの初期値はメインクロック (MCS=1, SCS=1) です。

SCS, MCS とともに "0" のときは SCS が優先され, サブクロックが選択されます。

サブクロックモードから PLL クロックモードへの切換え時は, CKSCR レジスタの発振安定待ち時間選択ビット (WS1, WS0) は "01_B", "10_B", "11_B" を設定してください。

5.5 発振安定待ち時間

電源投入時，ストップモード解除時，サブクロックからメインクロックまたは PLL クロックへの切換え時は，発振クロックの発振が停止しているため，発振開始後，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 クロックへの切換え時も，PLL 発振開始後の発振安定待ち時間が必要となります。

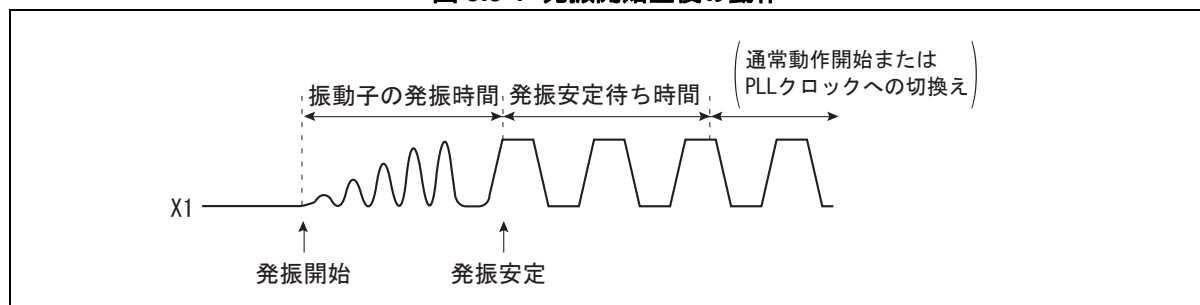
■ 発振安定待ち時間

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振動数（発振周波数）で安定して発振するまでに，一般的に数ミリ秒から数十ミリ秒の時間が必要です。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 クロックへの切換え時は，PLL 発振安定待ち時間の間，CPU はメインクロックで動作し，その後，PLL クロックに切り換わります。

図 5.5-1 に，発振開始直後の動作を示します。

図 5.5-1 発振開始直後の動作



5.6 振動子と外部クロックの接続

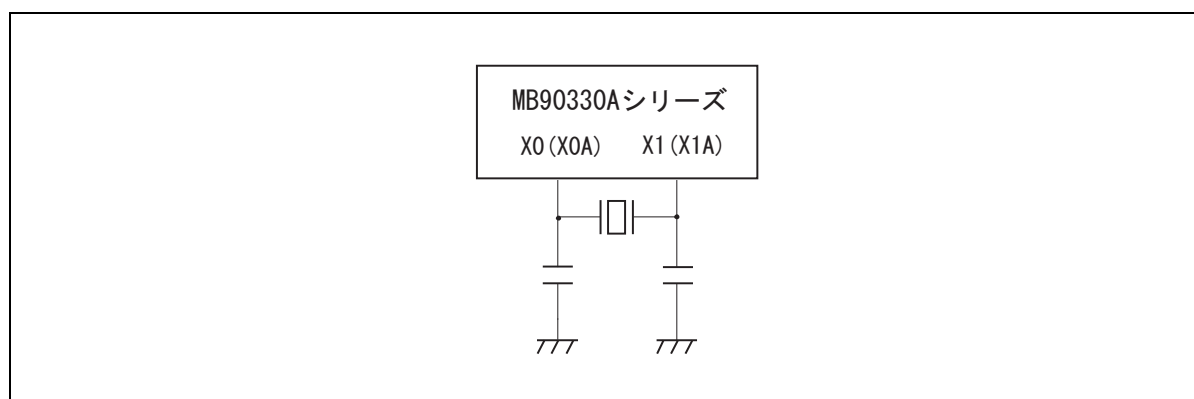
MB90330A シリーズは、システムクロック発生回路を内蔵しており、外部に振動子を接続してクロック発生させます。また、外部で生成したクロックを入力することもできます。

■ 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 5.6-1 の例で示すように接続してください。

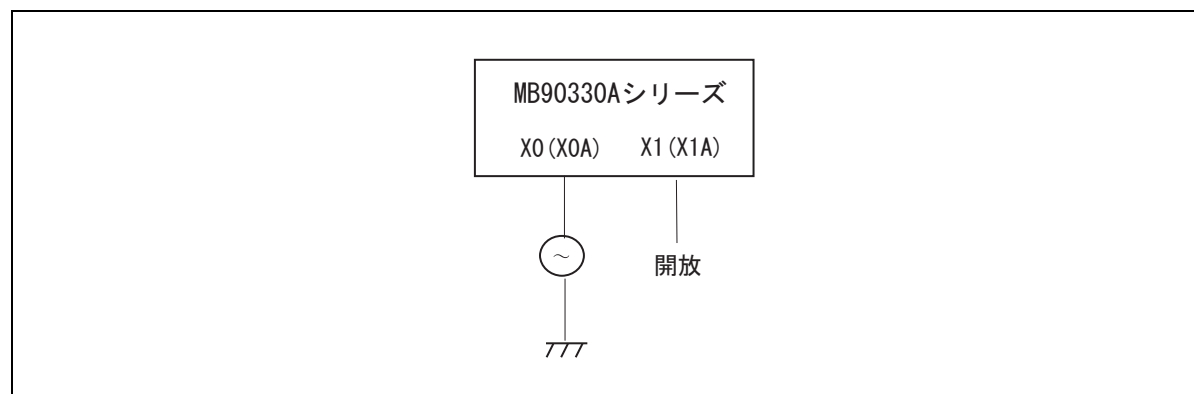
図 5.6-1 水晶振動子またはセラミック振動子の接続例



● 外部クロックの接続例

図 5.6-2 の例で示すように、外部クロックは X0 端子に接続し、X1 端子はオープン（開放）にしてください。

図 5.6-2 外部クロックの接続例



第6章

低消費電力モード

この章では、MB90330A シリーズの低消費電力モード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、ホールド、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

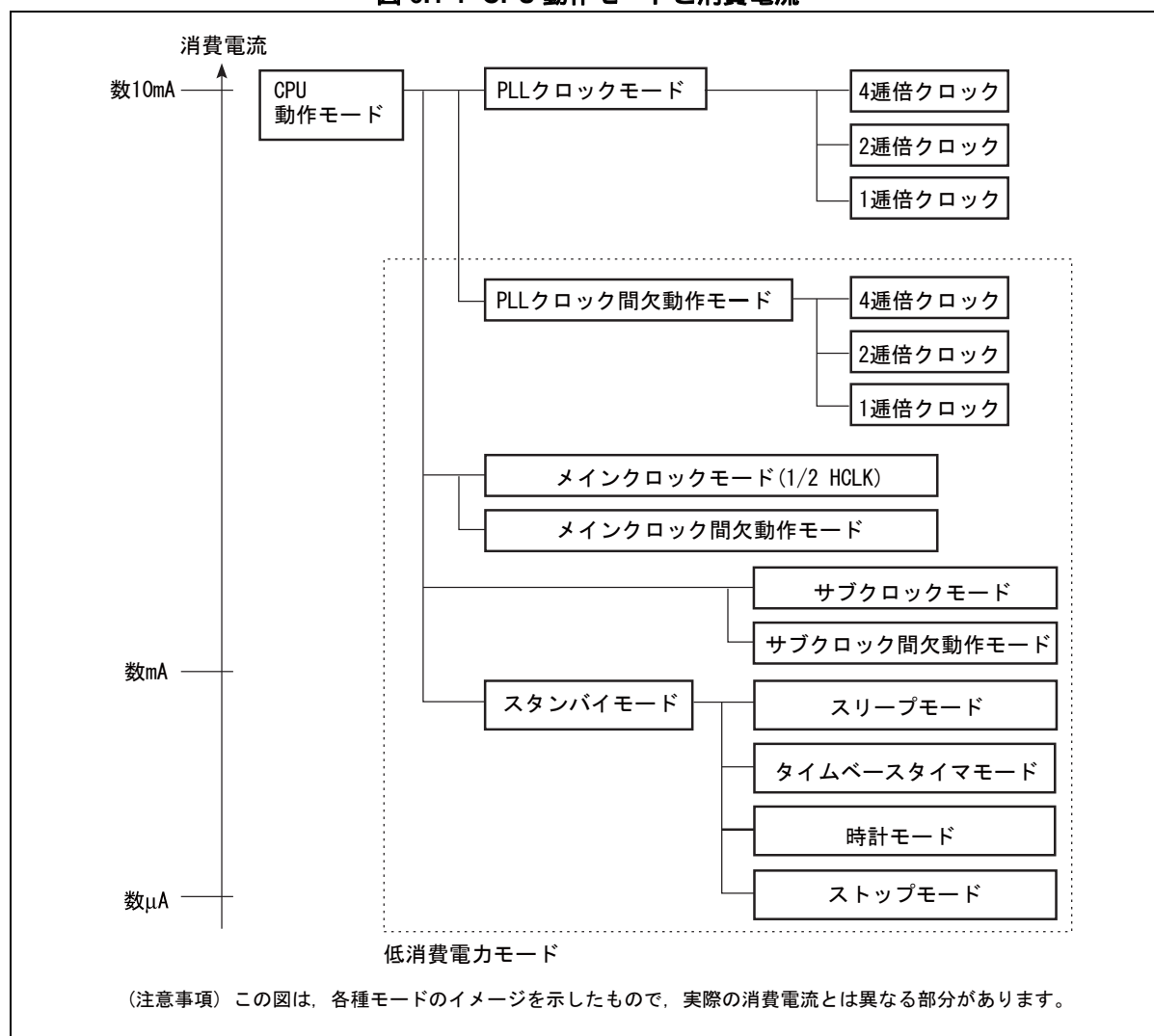
MB90330A シリーズは、動作クロックの選択とクロックの動作制御により、次のような CPU 動作モードを備えています。

- クロックモード
(PLL クロックモード、メインクロックモード、サブクロックモード)
- CPU 間欠動作モード
(PLL クロック間欠動作モード、メインクロック間欠動作モード、サブクロック間欠動作モード)
- スタンバイモード
(スリープモード、タイムベースタイマモード、ストップモード、時計モード)

■ CPU 動作モードと消費電流

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

発振クロック (HCLK) の PLL 通倍クロックで、CPU および周辺機能を動作させるモードです。

< 注意事項 >

USB ホスト、USB ファンクション使用時には、PLL クロックモードに設定する必要があります。

● メインクロックモード

発振クロック (HCLK) の 2 分周クロックで、CPU および周辺機能を動作させるモードです。メインクロックモード時には、PLL 通倍回路が停止します。

● サブクロックモード

サブクロック (SCLK) で、CPU および周辺機能を動作させるモードです。サブクロックモード時には、メインクロックおよび PLL 通倍回路が停止します。

< 参考 >

クロックモードについては、「5.4 クロックモード」を参照してください。

■ CPU 間欠動作モード

CPU 間欠動作モードとは、周辺機能には高速クロックを供給したまま、CPU を間欠動作させ、消費電力を低減するモードです。CPU 間欠動作モードは、CPU がレジスタ、内蔵メモリ、周辺機能および外部アクセスを行う場合、CPU にだけ間欠クロックを入力するモードです。

■ スタンバイモード

スタンバイモードは、低消費電力制御回路による CPU へのクロックの供給停止 (スリープモード) や、CPU と周辺機能へのクロックの供給停止 (タイムベースタイマモード) または発振クロックの停止 (ストップモード) によって消費電力を低減します。

● PLL スリープモード

PLL スリープモードは、PLL クロックモード時に CPU の動作クロックを停止させるモードで、CPU 以外は PLL クロックで動作します。

● メインスリープモード

メインスリープモードは、メインクロックモード時に CPU の動作クロックを停止させるモードで、CPU 以外はメインクロックで動作します。

● サブスリープモード

サブスリープモードは、サブクロックモード時に CPU の動作クロックを停止させるモードで、CPU 以外はサブクロックで動作します。

● 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、発振クロック、タイムベースタイマおよび時計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と時計タイマ以外の機能が停止します。

● 時計モード

時計モードは、時計タイマのみを動作させるモードです。サブクロックのみ動作し、メインクロックおよび PLL 通倍回路が停止します。

● ストップモード

ストップモードは、原発振を停止させるモードで、すべての機能が停止します。

< 注意事項 >

ストップモードでは、発振クロックが停止するため、最も低い消費電力でデータを保持できます。

クロックモードを切り換えた場合、切換えが完了するまでは、ほかのクロックモードおよび低消費電力モードへの切換えを行わないでください。切換えの完了は、クロック選択レジスタ (CKSCR) の MCM ビットおよび SCM ビットを参照して確認してください。USB 転送中はスタンバイモードへの遷移は禁止です。

6.2 低消費電力制御回路のブロックダイアグラム

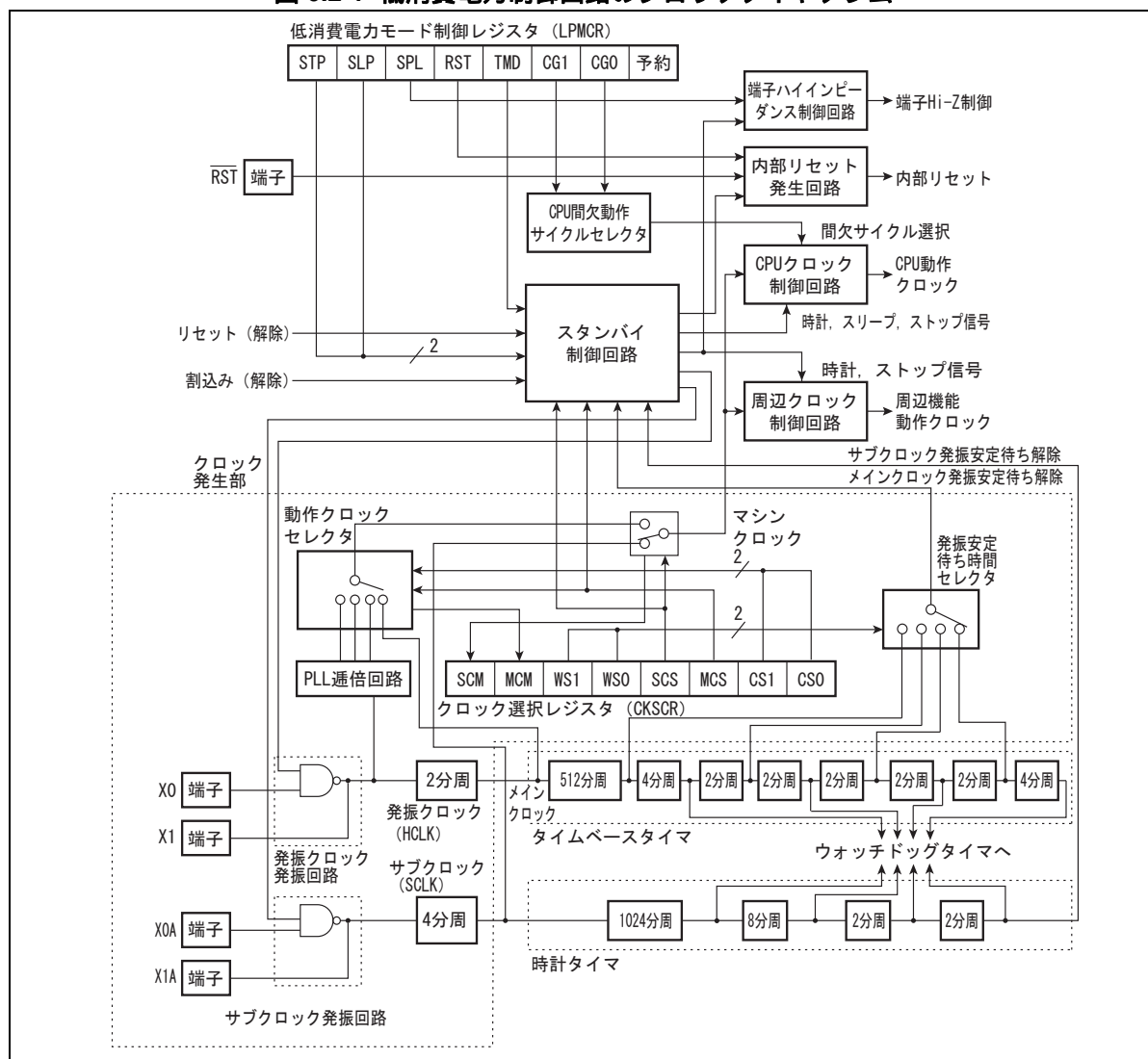
低消費電力制御回路は、次の7つのブロックで構成されています。

- CPU 間欠動作セレクト
- スタンバイ制御回路
- 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)

■ 低消費電力制御回路のブロックダイヤグラム

図 6.2-1 に、低消費電力制御回路のブロックダイヤグラムを示します。

図 6.2-1 低消費電力制御回路のブロックダイヤグラム



● CPU 間欠動作セレクト

CPU 間欠動作セレクトは、CPU 間欠動作モード時の一時停止クロック数を選択します。

● スタンバイ制御回路

スタンバイ制御回路は、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および解除を行います。

● CPU クロック制御回路

CPU クロック制御回路は、CPU に供給するクロックを制御する回路です。

● 周辺クロック制御回路

周辺クロック制御回路は、周辺機能に供給するクロックを制御する回路です。

● 端子ハイインピーダンス制御回路

端子ハイインピーダンス制御回路は、タイムベースタイマモードおよびストップモード時に、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に対しては、ストップモード時にプルアップ抵抗を切り離します。

● 内部リセット発生回路

内部リセット発生回路は、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)

低消費電力モード制御レジスタ (LPMCR) は、スタンバイモードへの移行 / 解除および 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移行 / 解除および CPU 間欠動作モード時の CPU クロッカー時停止サイクル数の設定などを行うレジスタです。

■ 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) のビット構成

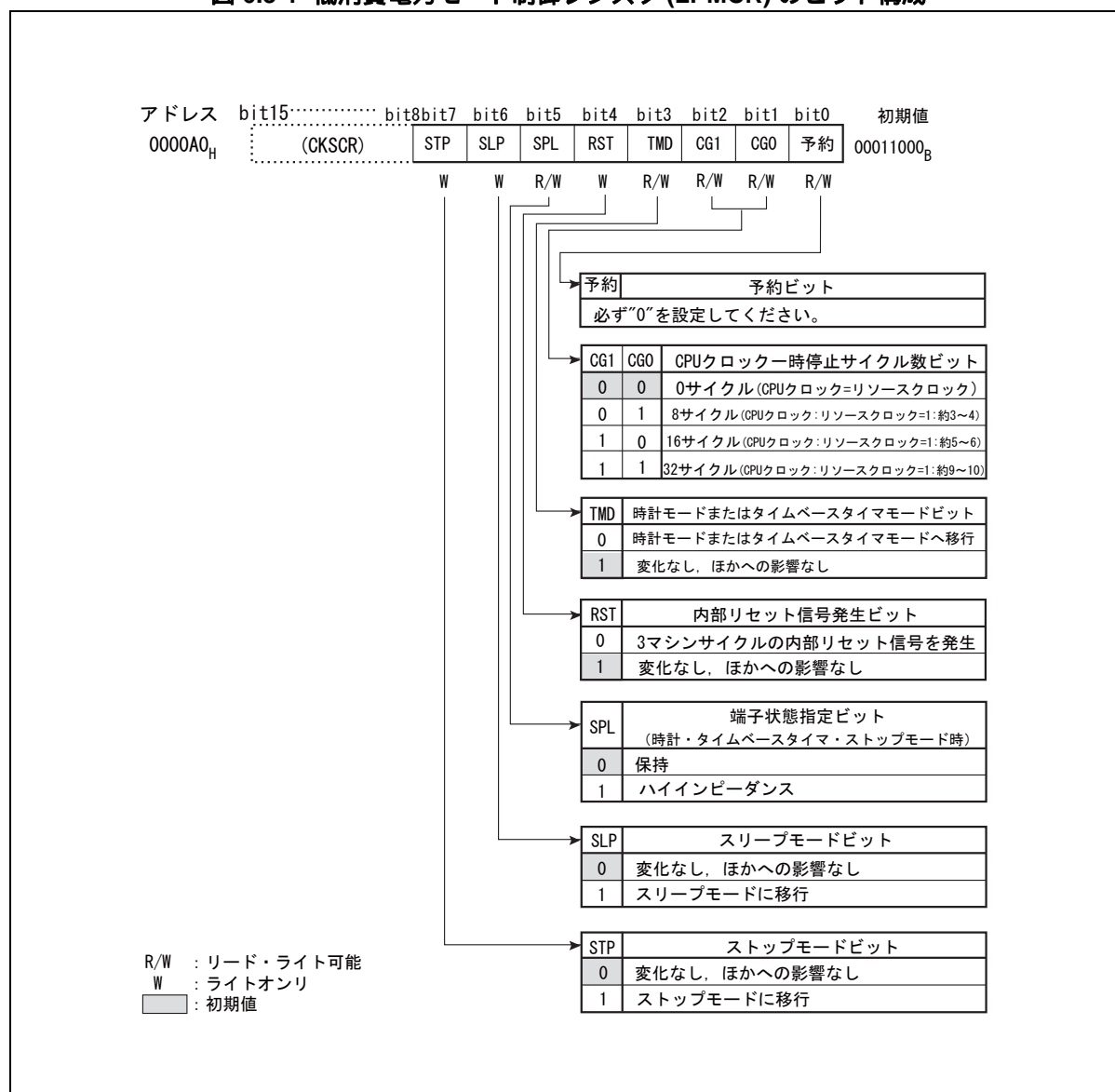


表 6.3-1 低消費電力モード制御レジスタ (LPMCR) の各ビットの機能

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移行を指示するビットです。 ・"1" を書き込むことにより、ストップモードへ移行します。 ・"0" を書き込んでも動作に影響を与えません。 ・リセットまたは割込み要求発生で "0" にクリアされます。 ・常に "0" が読み出されます。
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移行を指示するビットです。 ・"1" を書き込むことにより、スリープモードに移行します。 ・"0" を書き込んでも動作に影響を与えません。 ・リセットまたは割込み要求発生で "0" にクリアされます。 ・常に "0" が読み出されます。
bit5	SPL: 端子状態指定 ビット (時計、タイム ベースタイマ、 ストップモード 時)	<ul style="list-style-type: none"> ・時計モード、タイムベースタイマモード、ストップモードのときだけ有効です。 ・"0" の場合、外部端子のレベルを保持します。 ・"1" 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" を書き込むと、3 マシンサイクルの内部リセット信号を発生します。 ・"1" を書き込んでも動作に影響を与えません。 ・常に "1" が読み出されます。
bit3	TMD: 時計、タイム ベースタイマ モードビット	<ul style="list-style-type: none"> ・時計モード、タイムベースタイマモードへの移行を指示するビットです。 ・メインクロックモードまたは PLL クロックモード時に、"0" を書き込むことにより、タイムベースタイマモードに移行します。 ・サブクロックモード時に、"0" を書き込むことにより、時計モードに移行します。 ・リセットまたは割込み要求発生で "1" に初期化されます。 ・常に "1" が読み出されます。
bit2, bit1	CG1, CG0: CPU クロック 一時停止 サイクル数選択 ビット	<ul style="list-style-type: none"> ・CPU 間欠動作機能の CPU クロックの一時停止サイクル数を設定するビットです。 ・1 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。 ・リセットで "00_B" に初期化されます。
bit0	予約: 予約ビット	<ul style="list-style-type: none"> ・必ず "0" を設定してください。

■ 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）への遷移が行われますが、この場合に使用する命令は、表 6.3-2 の命令を使用するようにしてください。

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

「 」内の命令列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) への書込みを行う場合には、偶数アドレスで書込みが行われる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された場合には、誤動作の原因となる場合があります。

表 6.3-1 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	

■ STP, SLP, TMD ビットの優先順位

ストップモード要求、スリープモード要求および時計モード・タイムベースタイマモード要求を同時に行った場合は、以下の優先順位で要求を処理します。

ストップモード要求 > 時計モード・タイムベースタイマモード要求 > スリープモード要求

6.4 CPU 間欠動作モード

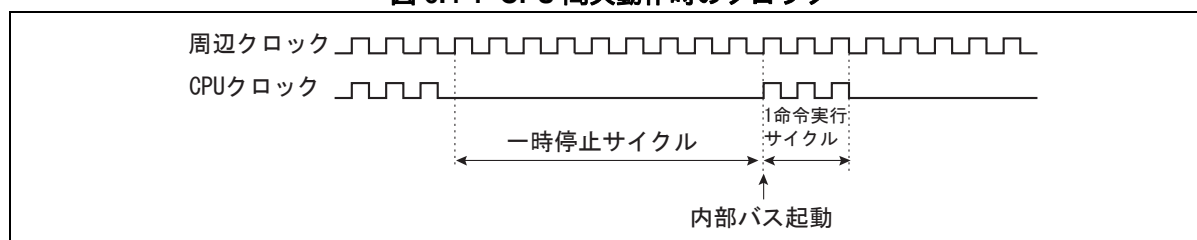
CPU 間欠動作モードは、外部バスや周辺機能を高速動作させたまま CPU を間欠動作させることで、消費電力を低減するモードです。

■ CPU 間欠動作モード

CPU 間欠動作モードは、レジスタ、内蔵メモリ (ROM, RAM), I/O, 周辺機能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 1 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。周辺機能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、低消費電力で処理が行えます。

- 低消費電力モード制御レジスタ (LPMCR) の CPU クロッカー一時停止サイクル数選択ビット (CG1, CG0) で、CPU に供給するクロックの一時停止サイクル数を選択します。
- 外部バス動作そのものは、周辺機能と同じクロックを使用します。
- CPU 間欠動作モードを使用した場合の命令実行時間の算出は、レジスタ、内蔵メモリ、内蔵周辺機能および外部バスアクセスを行う場合の命令実行回数に、一時停止サイクル数をかけた補正值を通常の実行時間に加えることで、求めることができます。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, サブスリープ), 時計, タイムベースタイマ, ストップの各モードがあります。

■ スタンバイモード時の動作状態

表 6.5-1 に、スタンバイモード時の動作状態を示します。

表 6.5-1 スタンバイモード時の動作状態

スタンバイモード		移行条件	メインクロック	サブクロック	マシクロック	CPU	周辺	端子	解除方法			
スリープモード	PLL スリープモード	SCS=1 MCS=0 SLP=1	動作	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	SCS=1 MCS=1 SLP=1										
	サブスリープモード	SCS=0 SLP=1	停止									
タイムベースタイマモード	タイムベースタイマモード (SPL=0)	SCS=1 TMD=0	動作	動作	停止		停止	停止 *1		保持		
	タイムベースタイマモード (SPL=1)	SCS=1 TMD=0								Hi-Z		
時計モード	時計モード (SPL=0)	SCS=0 TMD=0	停止					停止		停止	停止 *2	保持
	時計モード (SPL=1)	SCS=0 TMD=0										Hi-Z
ストップモード	ストップモード (SPL=0)	STP=1	停止	停止	停止						停止	保持
	ストップモード (SPL=1)	STP=1										Hi-Z

*1: タイムベースタイマ, 時計タイマは動作します。

*2: 時計タイマは動作します。

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

TMD: 低消費電力モード制御レジスタ (LPMCR) の時計, 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) のマシクロック選択ビット

SCS: クロック選択レジスタ (CKSCR) のマシクロック選択ビット (サブ)

Hi-Z: ハイインピーダンス

RST: 外部リセット端子

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移行を指示すると、PLL クロックモードを設定している場合は PLL スリープモードへ移行し、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移行し、サブクロックモードを設定しているときはサブスリープモードへ移行します。

■ スリープモードへの移行

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 (SLP) に "1", 時計・タイムベースタイマモードビット (TMD) に "1", ストップモードビット (STP) に "0" を書き込むと、スリープモードに移行します。このとき、クロック選択レジスタ (CKSCR) が PLL クロック選択ビット (MCS)=0, サブクロック選択ビット (SCS)=1 であれば PLL スリープモードに、MCS=1, SCS=1 であればメインスリープモードに、SCS=0 であればサブスリープモードに移行します。

< 注意事項 >

LPMCR の SLP ビットと STP ビットに同時に "1" を書き込むと、STP ビットを優先し、ストップモードへ移行します。

低消費電力モード制御レジスタの SLP ビットに "1", TMD ビットに "0" を同時に書き込むと、TMD ビットを優先し、タイムベースタイマモード、または時計モードに移行します。

● データ保持機能

スリープモードでは、アキュムレータなどの専用レジスタと内部 RAM の内容を保持します。

● ホールド機能

スリープモード中は、外部バスホールド機能は動作し、ホールド要求があるとホールド状態になります。

● 割込み要求発生中の動作

LPMCR の SLP ビットに "1" を書き込んだ場合に、割込み要求が発生している場合はスリープモードに移行しません。そのため、CPU は割込みを受け付けられない状態では次の命令を実行し、受け付ける状態であれば即座に割込み処理ルーチンへ分岐します。

● 端子状態

スリープモード中は、バス入出力またはバス制御として使用されている端子以外は、直前の状態を保持します。

■ 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

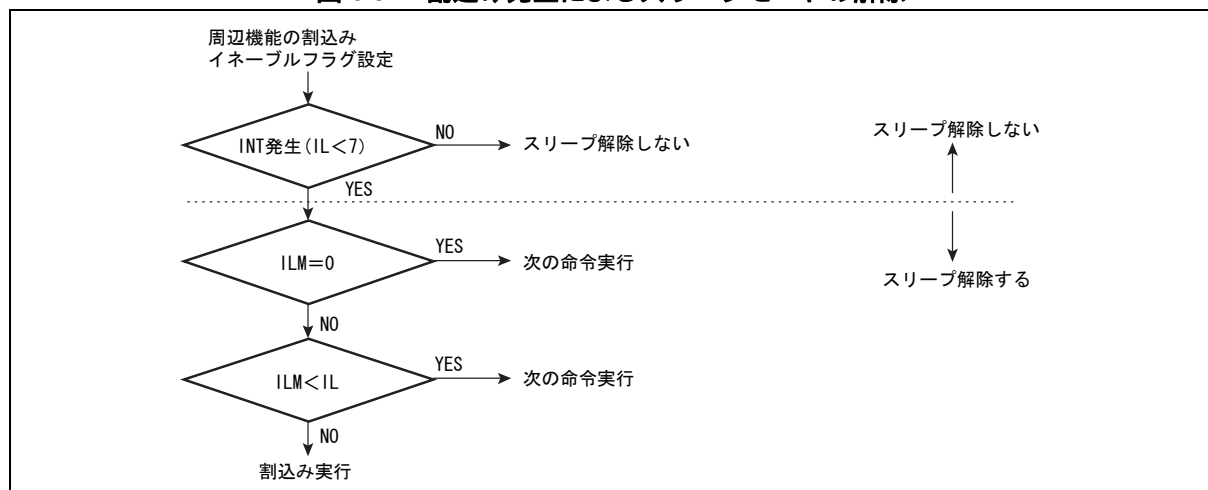
リセットによってメインクロックモードに初期化されます。

● 割込みによる復帰

スリープモード中に、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、スリープモードを解除します。スリープモードの解除後は、通常の割込み処理と同じ扱いとなります。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) の設定によって、割込みが受け付けられる場合は、CPU 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、スリープモードを指定した命令の次の命令から処理を続行します。

図 6.5-1 に、割込み発生によるスリープモードの解除を示します。

図 6.5-1 割込み発生によるスリープモードの解除



< 注意事項 >

割込み処理を実行する場合、通常はスリープモードを指定した命令の次の命令を実行した後に、割込み処理に移行します。ただし、スリープモードへの移行と外部バスホールド要求の受付が同時であった場合、次の命令を実行する前に割込み処理に移行することがあります。

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、時計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と時計タイマ以外は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移行

PLL クロックモードまたはメインクロックモード時 (クロック選択レジスタ (CKSCR) のサブクロック表示ビット (SCM)=1) に、低消費電力モード制御レジスタ (LPMCR) の時計、タイムベースタイマモードビット (TMD) に "0" を書き込むと、タイムベースタイマモードに移行します。

● データ保持機能

タイムベースタイマモードでは、アキュムレータなどの専用レジスタと内部RAMの内容を保持します。

● ホールド機能

タイムベースタイマモード中は、外部バスホールド機能は停止し、ホールド要求が入力されてもその要求を受け付けません。タイムベースタイマモードへの移行中にホールド要求が入力された場合、バスをハイインピーダンスにした状態で、HAK信号が"L"にならないことがあります。

● 割込み要求発生中の動作

低消費電力モード制御レジスタ (LPMCR) の TMD ビットに "0" を書き込んだ場合、割込み要求が発生しているときはタイムベースタイマモードに移行しません。

● 端子状態

タイムベースタイマモード中の外部端子を、直前の状態に保持するか、ハイインピーダンス状態にするかを、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) で制御できます。

■ 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要求の発生でタイムベースタイマモードを解除します。

● 外部リセットによる復帰

外部リセットによって、メインクロックモードに初期化されます。

● 割込みによる復帰

タイムベースタイマ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) の IL2, IL1, IL0=111_B 以外の場合)、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。コンディションコードレジスタ (CCR) の I フラグ、割込みレベルマスケレジスタ (ILM) および割込み制御レジスタ (ICR) の設定により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に入る前の次の命令から処理を続行します。

< 注意事項 >

割込み処理を実行する場合、通常はタイムベースタイマモードを指定した命令の次の命令を実行した後に、割込み処理に移行します。ただし、タイムベースタイマモードへの移行と外部バスホールド要求の受け付けが同時であった場合、次の命令を実行する前に割込み処理に移行することがあります。

6.5.3 時計モード

時計モードとは、サブクロックと時計タイマ以外の動作を止めるモードで、チップのほぼ全機能が停止します。

■ 時計モードへの遷移

サブクロックモード時（クロック選択レジスタ (CKSCR) のサブクロック表示ビット (SCS)=0）、低消費電力モード制御レジスタ (LPMCR) の時計、タイムベースタイマモードビット (TMD) に "0" を書き込むことによって、時計モードに移行します。

- データ保持機能

時計モードでは、アキュムレータなどの専用レジスタと内部 RAM の内容を保持します。

- ホールド機能

時計モード中は、外部バスホールド機能は停止し、ホールド要求が入力されても、その要求は受け付けません。時計モードへの遷移中に、ホールド要求が入力された場合、バスをハイインピーダンスにした状態で、HAK 信号が "L" にならないことがあります。

- 割込み要求発生中の動作

LPMCR の TMD ビットに "0" を設定したときに、割込み要求が発生している場合は時計モードに移行しません。

- 端子状態の設定

時計モード中の外部端子を、直前の状態に保持するか、ハイインピーダンス状態にするかを、LPMCR の端子状態指定ビット (SPL) で制御できます。

■ 時計モードの解除

低消費電力制御回路は、リセットの入力または割込み要求の発生によって時計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時計モードの解除の場合は、時計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実行されます。

- 割込みによる復帰

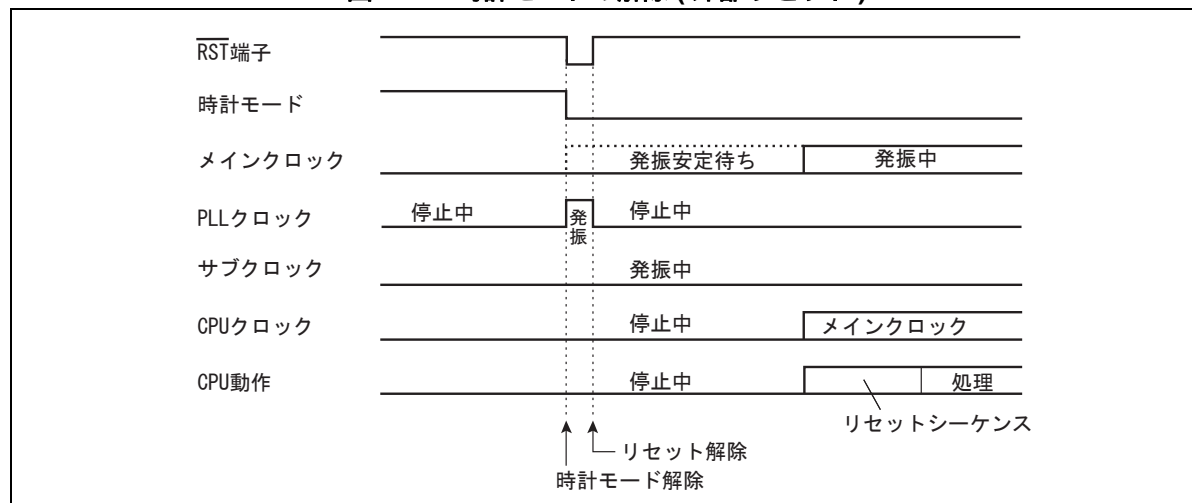
時計モード中に周辺回路などから割込みレベルが 7 より高い割込み要求が発生すると（割込み制御レジスタ (ICR) の IL2, IL1, IL0 が "111_B" 以外）、低消費電力制御回路は時計モードを解除し、即座にサブクロックモードに移行します。サブクロックモードに移行後は、通常の割込み処理と同じ扱いとなります。コンディションコードレジスタ (CCR) の I フラグ、割込みレベルマスクレジスタ (ILM) および割込み制御レジスタ (ICR) を設定して割込みが受け付けられる場合は、割込み処理を実行します。割込みが受け付けられない場合は、時計モードに入る前の次の命令から処理を続行します。

< 注意事項 >

割り込み処理を実行する場合、通常は時計モードを指定した命令の次の命令を実行した後に、割り込み処理に移行します。ただし、時計モードへの移行と外部バスホールド要求の受け付けが同時であった場合、次の命令を実行する前に割り込み処理に移行することがあります。

図 6.5-2 に、時計モードの解除動作を示します。

図 6.5-2 時計モードの解除 (外部リセット)



6.5.4 ストップモード

ストップモードは、原発振を停止させるモードで、すべての機能が停止します。したがって、最も低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) のストップモードビット (STP) に "1" を書き込むと、ストップモードに移行します。

- 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 の内容を保持します。

- ホールド機能

ストップモード中は、外部バスホールド機能は停止し、ホールド要求が入力されても、その要求を受け付けません。ストップモードへの遷移中に、ホールド要求が入力された場合、バスをハイインピーダンスにした状態で、HAK 信号が "L" にならないことがあります。

- 割込み要求発生中の動作

低消費電力モード制御レジスタ (LPMCR) の STP ビットに "1" を書き込んだ場合に、割込み要求が発生している場合はストップモードに移行しません。

- 端子状態の設定

ストップモード中の外部端子を、直前の状態に保持するか、ハイインピーダンス状態にするかを、低消費電力モード制御レジスタ (LPMCR) 端子状態指定ビット (SPL) で指定できます。

■ 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の復帰の場合、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、その後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実行されます。

- 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) の IL2, IL1, IL0=111_B 以外の場合)、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、クロック選択レジスタ (CKSCR) の発振安定待ち時間選択ビット (WS1, WS0) 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、通常の割込み処理と同じ扱いとなります。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

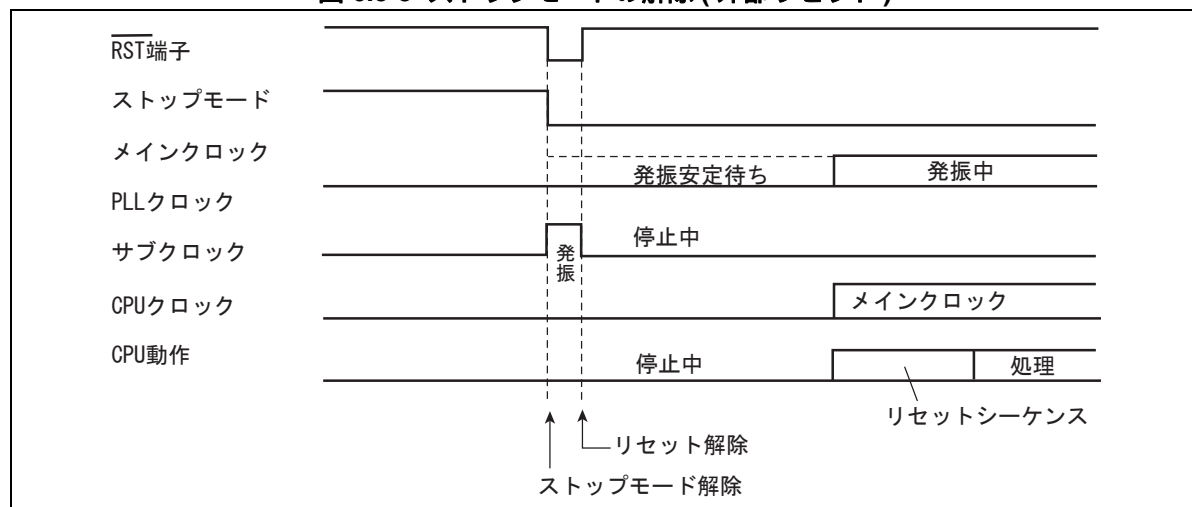
込み制御レジスタ (ICR) を設定して割込みが受け付けられる場合は、割込み処理を実行します。割込みが受け付けられない場合は、ストップモードに入る前の次の命令から処理を続行します。

< 注意事項 >

- 割込み処理を実行する場合、通常はストップモードを指定した命令の次の命令を実行した後に、割込み処理に移行します。ただし、ストップモードへの移行と外部バスホールド要求の受付けが同時であった場合、次の命令を実行する前に割込み処理に移行することがあります。
- PLL ストップモード中は、メインクロックおよび PLL 通倍回路が停止しているため、PLL ストップモードから復帰する場合は、メインクロック発振安定待ち時間および PLL クロック発振安定待ち時間を確保する必要があります。この場合の発振安定待ち時間は、クロック選択レジスタの発振安定待ち時間選択ビット (CKSCR:WS1, WS0) に設定された値に従い、メインクロック発振安定待ち時間および PLL クロック発振安定待ち時間を同時にカウントしますので、CKSCR:WS1, WS0 ビットは、発振安定待ち時間の長い方に合わせて値を設定してください。ただし、PLL クロック発振安定待ち時間は $2^{14}/HCLK$ 以上必要になりますので、CKSCR:WS1, WS0 ビットは "10_B" または "11_B" を設定してください。

図 6.5-3 に、ストップモードの解除動作を示します。

図 6.5-3 ストップモードの解除 (外部リセット)



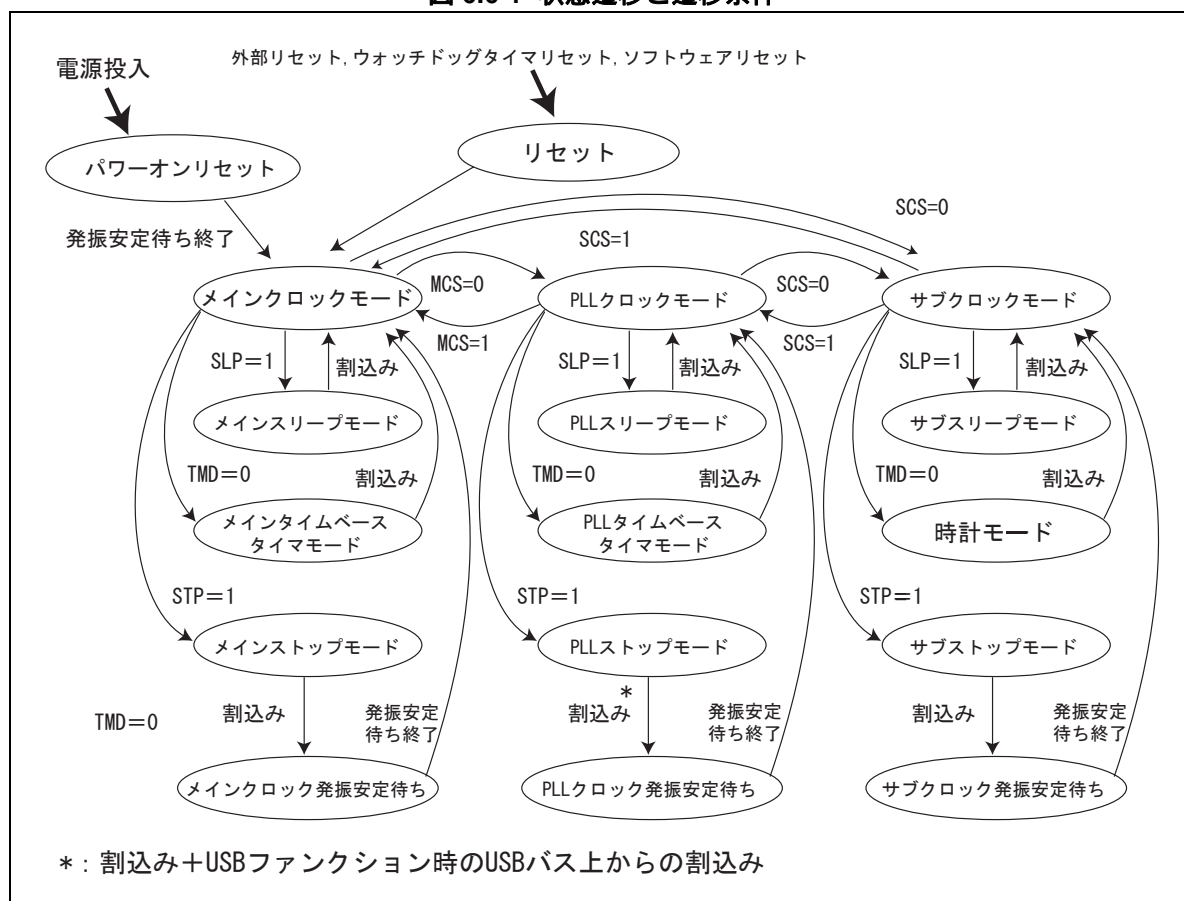
6.6 状態遷移図

MB90330A シリーズの動作状態の遷移と、遷移条件を示します。

■ 状態遷移図

図 6.6-1 に、MB90330A シリーズの動作状態の遷移と、遷移条件を示します。

図 6.6-1 状態遷移と遷移条件



■ 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 低消費電力モードの動作状態

動作状態	メイン クロック	サブ クロック	PLL クロック	CPU	周辺	時計	タイム ベース タイマ	クロック ソース
PLL	動作	動作	動作	動作	動作	動作	動作	PLL クロック
PLL スリープ				停止				
PLL タイムベース タイマ								
PLL ストップ	停止	停止	停止		停止	停止	停止	
PLL 発振安定待ち	動作	動作	動作	動作		動作		
メイン	動作	動作	停止	動作	動作	動作	動作	メイン クロック
メインスリープ				停止				
メインタイムベース タイマ								
メインストップ	停止	停止			停止	停止	停止	
メイン発振安定待ち	動作	動作		動作		動作		
サブ	停止	動作	停止	動作	動作	動作	停止	サブ クロック
サブスリープ				停止				
時計		停止			停止			
サブストップ						停止		
サブ発振安定待ち				動作				
パワーオンリセット	動作	動作	停止	停止	停止	動作	動作	メイン クロック
リセット			動作					

6.7 スタンバイモード，ホールド，リセット時の端子状態

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

■ 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端子状態を示します。

表 6.7-1 シングルチップモード時の端子状態

端子名	スリープ時	ストップ時		ホールド時	リセット時
		SPL=0	SPL=1		
P07 ~ P00	直前の状態を保持 *2	入力遮断 / 直前の状態を保持 *2, *3	入力遮断 / 出力 Hi-Z *3	本状態はない	入力不可 / 出力 Hi-Z
P17 ~ P10					
P27 ~ P20					
P37 ~ P30					
P47 ~ P40					
P57 ~ P50					
P77 ~ P70					
P87 ~ P80					
P96 ~ P90					
PA7 ~ PA0					
PB6 ~ PB0					
P67 ~ P60	入力可能 *1	入力可能 *1			入力不可
DVP	USB ポート入力	USB ポート入力 *4		*5	Hi-Z
DVM					
HVP					
HVM					
UTEST	プルダウン接続	プルダウン接続			プルダウン接続
HCON	直前の状態を保持	直前の状態を保持			"H" 出力

*1: 出力状態として使用している場合には，ほかのポートと同じです。入力可能とは，入力機能が使用可能な状態であることを意味するので，ブルアップ / ブルダウンまたは外部からの入力が必要です。

*2: 本モードになる直前に出力していた状態をそのまま出力します。または，入力であれば入力不可を意味します。入力不可とは，端子からすぐの入力ゲートの動作は許可状態であるが，内部回路が動作していないので，端子の内容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。

*4: USB 要因によるストップ時は USB ポート入力で，それ以外の要因によるストップ時は直前の状態のままです。

*5: USB 端子の動作に影響はありません。

< 注意事項 >

USB 転送中はスタンバイモードへの遷移は禁止です。

■ 外部バス 16 ビットデータバスモードおよびマルチプレクス 16 ビット外部バスモードの端子状態

表 6.7-2 に、外部バス 16 ビットデータバスモードおよびマルチプレクス 16 ビット外部バスモードの端子状態を示します。

表 6.7-2 外部バス 16 ビットデータバスモードおよびマルチプレクス 16 ビット外部バスモードの端子状態

端子名	スリープ時	ストップ時		ホールド時	リセット時	
		SPL=0	SPL=1			
P07 ~ P00 (AD07 ~ AD00)	入力不可 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z *6	入力不可 / 出力 Hi-Z	入力不可 / 出力 Hi-Z	
P17 ~ P10 (AD15 ~ AD08)						
P27 ~ P20 (A23 ~ A16)	出力状態 *1, *4	出力状態 *1, *4		入力不可 / 出力 Hi-Z *4	出力状態 *1	
P57(CLK)	入力不可 / 出力可能 *2, *4	入力不可 / 出力状態 *1, *4		入力不可 / 出力可能 *2, *4	入力不可 / 出力可能 *2	
P56(RDY)	直前の状態を保持 *5	入力遮断 / 直前の状態を保持 *5, *6		入力不可 *4	入力不可 / 出力 Hi-Z	
P55(HAK)				"L" 出力		
P54(HRQ)				"1" 入力		
P53(WRH)	"H" 出力 *4	"H" 出力 *4		入力不可 / 出力 Hi-Z *4	"H" 出力	
P52(WRL)						
P51(RD)	"H" 出力	"H" 出力		入力不可 / 出力 Hi-Z	出力可能 *2	
P50(ALE)	"L" 出力	"L" 出力				
P37 ~ P30	直前の状態を保持 *5	入力遮断 / 直前の状態を保持 *5	入力遮断 / 出力 Hi-Z *6	直前の状態を保持 *5	入力不可 / 出力 Hi-Z	
P47 ~ P40						
P77 ~ P70						
P87 ~ P80						
P96 ~ P90						
PA7 ~ PA0						
PB6 ~ PB0						
P67 ~ P60	入力可能 *3	入力可能 *3	入力可能 *3		入力不可	
DVP	USB ポート入力	USB ポート入力 *7	*8		Hi-Z	
DVM						
HVP						
HVM						
UTEST	プルダウン接続	プルダウン接続				プルダウン接続
HCON	直前の状態を保持	直前の状態を保持				"H" 出力

*1: 出力状態とは、端子駆動用トランジスタは駆動許可状態ですが、内部回路の動作が停止状態のため、"H" ないし "L" の固定値を出力することを意味します。内部周辺回路が動作中で、出力機能を使用している場合には、リセット時を除いて出力が変化します。リセット時の出力変化はありません。

*2: 出力可能とは、端子駆動用トランジスタを駆動状態にしてあり、内部回路の動作が許可状態のため、動作内容が端子に現れることを意味します。

*3: 出力状態として使用している場合には、ほかのポートと同じです。入力可能とは、入力機能が使用可能な状態であることを意味するので、プルアップ / プルダウンまたは外部からの入力が必要です。

- *4: 出力ポートとして使用している場合は、直前の値を保持します。
- *5: 本モードになる直前に出力していた状態をそのまま出力します。または、入力であれば入力不可を意味します。
入力不可とは、端子からすぐの入力ゲートの動作は許可状態ですが、内部回路が動作していないので、端子の内容が内部で受け付けられない状態を意味します。
- *6: 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。
- *7: USB 要因によるストップ時は USB ポート入力で、それ以外の要因によるストップ時は直前の状態のままです。
- *8: USB 端子の動作に影響はありません。

< 注意事項 >

USB 転送中はスタンバイモードへの遷移は禁止です。

■ 外部バス 8 ビットデータバスモードおよびマルチプレクス 8 ビット外部バスモードの端子状態

表 6.7-3 に，外部バス 8 ビットデータバスモードおよびマルチプレクス 8 ビット外部バスモードの端子状態を示します。

表 6.7-3 外部バス 8 ビットデータバスモードおよびマルチプレクス 8 ビット外部バスモードの端子状態

端子名	スリープ時	ストップ時		ホールド時	リセット時
		SPL=0	SPL=1		
P07 ~ P00 (AD07 ~ AD00)	入力不可 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z *6	入力不可 / 出力 Hi-Z	入力不可 / 出力 Hi-Z
P17 ~ P10 (A15 ~ A08)	出力状態 *1	出力状態 *1		入力不可 / 出力 Hi-Z *4	出力状態 *1
P27 ~ P20 (A23 ~ A16)	出力状態 *1, *4	出力状態 *1, *4		入力不可 / 出力 Hi-Z *4	
P57(CLK)	入力不可 / 出力可能 *2, *4	入力不可 / 出力状態 *1, *4		入力不可 / 出力可能 *2, *4	入力不可 / 出力可能 *2
P56(RDY)	直前の状態を保持 *5	入力遮断 / 直前の状態を 保持 *5		入力不可 *4	入力不可 / 出力 Hi-Z
P55(HAK)				"L" 出力	
P54(HRQ)				"I" 入力	
P53(WRH)				直前の状態を 保持 *5	
P52(WRL)	"H" 出力 *5	"H" 出力 *5		入力不可 / 出力 Hi-Z *4	"H" 出力
P51(RD)	"H" 出力	"H" 出力		入力不可 / 出力 Hi-Z	
P50(ALE)	"L" 出力	"L" 出力			"L" 出力
P37 ~ P30	直前の状態を保持 *5	入力遮断 / 直前の状態を 保持 *5		直前の状態を 保持 *5	入力不可 / 出力 Hi-Z
P47 ~ P40					
P77 ~ P70					
P87 ~ P80					
P96 ~ P90					
PA7 ~ PA0					
PB6 ~ PB0					
P67 ~ P60	入力可能 *3	入力可能 *3	入力可能 *3		入力不可
DVP	USB ポート 入力	USB ポート入力 *7		*8	Hi-Z
DVM					
HVP					
HVM					
UTEST	ブルダウン接続	ブルダウン接続			ブルダウン接続
HCON	直前の状態を 保持	直前の状態を保持			"H" 出力

*1: 出力状態とは，端子駆動用トランジスタは駆動許可状態ですが，内部回路の動作が停止状態のため，"H" ないし "L" の固定値を出力することを意味します。内部周辺回路が動作中で，出力機能を使用している場合には，リセット時を除いて出力が変化します。リセット時の出力変化はありません。

- *2: 出力可能とは、端子駆動用トランジスタを駆動状態にしてあり、内部回路の動作が許可状態のため、動作内容が端子に現れることを意味します。
- *3: 出力状態として使用している場合には、ほかのポートと同じです。入力可能とは、入力機能が使用可能な状態であることを意味するので、プルアップ / プルダウンまたは外部からの入力が必要です。
- *4: 出力ポートとして使用している場合は、直前の値を保持します。
- *5: 本モードになる直前に出力していた状態をそのまま出力します。または、入力であれば入力不可を意味します。
入力不可とは、端子からすぐの入力ゲートの動作は許可状態ですが、内部回路が動作していないため、端子の内容が内部で受け付けられない状態を意味します。
- *6: 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。
- *7: USB 要因によるストップ時は USB ポート入力で、それ以外の要因によるストップ時は直前の状態のままです。
- *8: USB 端子の動作に影響がありません。

< 注意事項 >

USB 転送中はスタンバイモードへの遷移は禁止です。

■ 外部バス 16 ビットデータバスモードおよびノンマルチプレクス 16 ビット外部バスモードの端子状態

表 6.7-4 に, 外部バス 16 ビットデータバスモードおよびノンマルチプレクス 16 ビット外部バスモードの端子状態を示します。

表 6.7-4 外部バス 16 ビットデータバスモードおよびノンマルチプレクス 16 ビット外部バスモードの端子状態

端子名	スリープ時	ストップ時		ホールド時	リセット時
		SPL=0	SPL=1		
P07 ~ P00 (D07 ~ D00)	入力不可 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z *6	入力不可 / 出力 Hi-Z	入力不可 / 出力 Hi-Z
P17 ~ P10 (D15 ~ D08)					
P37 ~ P30 (A07 ~ A00)	出力状態 *1	出力状態 *1			出力状態 *1
P47 ~ P40 (A15 ~ A08)					
P27 ~ P20 (A23 ~ A16)	出力状態 *1, *4	出力状態 *1, *4		入力不可 / 出力 Hi-Z *4	
P57(CLK)	入力不可 / 出力可能 *2, *4	入力不可 / 出力可能 *1, *4		入力不可 / 出力可能 *2, *4	入力不可 / 出力可能 *2
P56(RDY)	直前の状態を保持 *5	入力遮断 / 直前の状態を保持 *5		入力不可 *4	入力不可 / 出力 Hi-Z
P55(HAK)				"L" 出力	
P54(HRQ)				"1" 入力	
P53(WRH)	"H" 出力 *5	"H" 出力 *5		入力不可 / 出力 Hi-Z *4	"H" 出力
P52(WRL)					
P51(RD)	"H" 出力	"H" 出力		入力不可 / 出力 Hi-Z	出力可能 *2
P50(ALE)	"L" 出力	"L" 出力			
P77 ~ P70	直前の状態を保持 *5	入力遮断 / 直前の状態を保持 *5		直前の状態を保持 *5	入力不可 / 出力 Hi-Z
P87 ~ P80					
P96 ~ P90					
PA7 ~ PA0					
PB6 ~ PB0					
P67 ~ P60	入力可能 *3	入力可能 *3	入力可能 *3		入力不可
DVP	USB ポート入力	USB ポート入力 *7		*8	Hi-Z
DVM					
HVP					
HVM					
UTEST	プルダウン接続	プルダウン接続			プルダウン接続
HCON	直前の状態を保持	直前の状態を保持			"H" 出力

*1: 出力状態とは, 端子駆動用トランジスタは駆動許可状態ですが, 内部回路の動作が停止状態のため, "H" ないし "L" の固定値を出力することを意味します。内部周辺回路が動作中で, 出力機能を使用している場合には, リセット時を除いて出力が変化します。リセット時の出力変化はありません。

*2: 出力可能とは, 端子駆動用トランジスタを駆動状態にしてあり, 内部回路の動作が許可状態のため, 動作内容が

端子に現れることを意味します。

*3: 出力状態として使用している場合には、ほかのポートと同じです。入力可能とは、入力機能が使用可能な状態であることを意味するので、プルアップ / プルダウンまたは外部からの入力が必要です。

*4: 出力ポートとして使用している場合は、直前の値を保持します。

*5: 本モードになる直前に出力していた状態をそのまま出力します。または、入力であれば入力不可を意味します。入力不可とは、端子からすぐの入力ゲートの動作は許可状態ですが、内部回路が動作していないので、端子の内容が内部で受け付けられない状態を意味します。

*6: 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。

*7: USB 要因によるストップ時は USB ポート入力で、それ以外の要因によるストップ時は直前の状態のままです。

*8: USB 端子の動作に影響はありません。

< 注意事項 >

USB 転送中はスタンバイモードへの遷移は禁止です。

■ 外部バス 8 ビットデータバスモードおよびノンマルチプレクス 8 ビット外部バスモードの端子状態

表 6.7-5 に，外部バス 8 ビットデータバスモードおよびノンマルチプレクス 8 ビット外部バスモードの端子状態を示します。

表 6.7-5 外部バス 8 ビットデータバスモードおよびノンマルチプレクス 8 ビット外部バスモード

端子名	スリープ時	ストップ時		ホールド時	リセット時	
		SPL=0	SPL=1			
P07 ~ P00 (D07 ~ D00)	入力不可 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z	入力遮断 / 出力 Hi-Z *6	入力不可 / 出力 Hi-Z	入力不可 / 出力 Hi-Z	
P37 ~ P30 (A07 ~ A00)	出力状態 *1	出力状態 *1			入力不可 / 出力 Hi-Z *4	出力状態 *1
P47 ~ P40 (A15 ~ A08)						
P27 ~ P20 (A23 ~ A16)	出力状態 *1, *4	出力状態 *1, *4		入力不可 / 出力 Hi-Z *4		
P57(CLK)	入力不可 / 出力可能 *2, *4	入力不可 / 出力可能 *1, *4		入力不可 / 出力可能 *2, *4	入力不可 / 出力可能 *2	
P56(RDY)	直前の状態を保持 *5	入力遮断 / 直前の状態を保持 *5		入力不可 *4	入力不可 / 出力 Hi-Z	
P55(HAK)				"L" 出力		
P54(HRQ)				"1" 入力		
P53(WRH)				直前の状態を保持 *5		
P52(WRL)	"H" 出力 *4	"H" 出力 *4		入力不可 / 出力 Hi-Z *4	"H" 出力	
P51(RD)	"H" 出力	"H" 出力		入力不可 / 出力 Hi-Z	出力可能 *2	
P50(ALE)	"L" 出力	"L" 出力				
P17 ~ P10	直前の状態を保持 *5	入力遮断 / 直前の状態を保持 *5		直前の状態を保持 *5	入力不可 / 出力 Hi-Z	
P77 ~ P70						
P87 ~ P80						
P96 ~ P90						
PA7 ~ PA0						
PB6 ~ PB0						
P67 ~ P60	入力可能 *3	入力可能 *3	入力可能 *3		入力不可	
DVP	USB ポート入力	USB ポート入力 *7		*8	Hi-Z	
DVM						
HVP						
HVM						
UTEST	プルダウン接続	プルダウン接続			プルダウン接続	
HCON	直前の状態を保持	直前の状態を保持			"H" 出力	

*1: 出力状態とは，端子駆動用トランジスタは駆動許可状態にしてありますが，内部回路の動作が停止状態のため，"H" ないし "L" の固定値を出力することを意味します。内部周辺回路が動作中で，出力機能を使用している場合には，リセット時を除いて出力が変化します。リセット時の出力変化はありません。

- *2: 出力可能とは、端子駆動用トランジスタを駆動状態にしてあり、内部回路の動作が許可状態のため、動作内容が端子に現れることを意味します。
- *3: 出力状態として使用している場合には、ほかのポートと同じです。入力可能とは、入力機能が使用可能な状態であることを意味するので、プルアップ / プルダウンまたは外部からの入力が必要です。
- *4: 出力ポートとして使用している場合は、直前の値を保持します。
- *5: 本モードになる直前に出力していた状態をそのまま出力します。または、入力であれば入力不可を意味します。
入力不可とは、端子からすぐの入力ゲートの動作は許可状態ですが、内部回路が動作していないので、端子の内容が内部で受け付けられない状態を意味します。
- *6: 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。
- *7: USB 要因によるストップ時は USB ポート入力、それ以外の要因によるストップ時は直前の状態のままです。
- *8: USB 端子の動作に影響はありません。

< 注意事項 >

USB 転送中はスタンバイモードへの遷移は禁止です。

6.8 低消費電力モード使用上の注意

低消費電力モード使用時には、次に示す内容に関する注意が必要です。

- スタンバイモードへの移行と割込み
 - スタンバイモードの割込みによる解除
 - 発振安定待ち時間
 - クロックモードの切換え
 - 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 に対して割込み要求が発生していると、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) とスリープモードビット (SLP) を "1" または時計モードまたはタイムベースタイマモードビット (TMD) を "0" としても無視されるため、各スタンバイモードへは移行しません (割込み処理後にも、スタンバイモードへの移行はしません)。この場合、割込みレベルが 7 より高ければ、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。

また、CPU が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ければ、スタンバイモードへ移行できます。

■ スタンバイモードの割込みによる解除

スリープ、タイムベースタイマおよび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すると、スタンバイモードは解除されます。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。

割込みによるスタンバイモードの解除後は、通常の割込み動作として、割込み要求に対応する割込みレベル設定ビット (ICR レジスタの IL2, IL1, IL0 ビット) の優先度が割込みレベルマスクレジスタ (ILM) より高く、コンディションコードレジスタ (CCR) の I フラグによって割込みが許可 (I=1) されている場合は、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、スタンバイモードを指定した命令の次の命令から動作を再開します。

割込み処理を実行する場合、通常はスタンバイモードを指定した命令の次の命令を実行した後に、割込み処理に移行します。

ただし、スタンバイモードに移行した場合の条件によっては、次の命令を実行する前に割込み処理に移行することがあります。

< 注意事項 >

復帰直後に割込み処理ルーチンへ分岐させない場合は、スタンバイモード設定の前に割込み禁止をするなどの対策が必要です。

■ 発振安定待ち時間

● 発振クロック発振安定待ち時間

ストップモード中は、原発振用の発振器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。発振安定待ち時間としては、クロック選択レジスタ (CKSCR) の発振安定待ち時間選択ビット (WS1, WS0) で選択された時間をとります。

< 注意事項 >

CKSCR レジスタの発振安定待ち時間選択ビット (WS1, WS0) に "00_B" を設定する場合は、メインクロック時のみにしてください。

● 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し、PLL クロックが停止している状態から、CPU または周辺を PLL クロックで動作するモードに移行する場合は、PLL クロック発振安定待ち状態へ移行し、発振安定待ち中は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) 固定です。

サブクロックモード中は、メインクロックおよび PLL 通倍回路が停止しているため、PLL クロックモードに移行する場合は、メインクロック発振安定待ち時間および PLL クロック発振安定待ち時間を確保する必要があります。この場合の発振安定待ち時間は、クロック選択レジスタの発振安定待ち時間選択ビット (CKSCR: WS1, WS0) に設定された値に従い、メインクロック発振安定待ち時間および PLL クロック発振安定待ち時間を同時にカウントしますので、CKSCR: WS1, WS0 ビットには、発振安定待ち時間の長いほうに合わせて値を設定してください。ただし、PLL クロック発振安定待ち時間は $2^{14}/\text{HCLK}$ 以上必要ですので、CKSCR: WS1, WS0 ビットには "10_B" または "11_B" を設定してください。

PLL ストップモード中は、メインクロックおよび PLL 通倍回路が停止しているため、PLL ストップモードから復帰する場合は、メインクロック発振安定待ち時間および PLL クロック発振安定待ち時間を確保する必要があります。この場合の発振安定待ち時間は、クロック選択レジスタの発振安定待ち時間選択ビット (CKSCR: WS1, WS0) に設定された値に従い、メインクロック発振安定待ち時間および PLL クロック発振安定待ち時間を同時にカウントしますので、CKSCR: WS1, WS0 ビットには、発振安定待ち時間の長いほうに合わせて値を設定してください。ただし、PLL クロック発振安定待ち時間は $2^{14}/\text{HCLK}$ 以上必要ですので、CKSCR: WS1, WS0 ビットには "10_B" または "11_B" を設定してください。

■ クロックモードの切換え

クロックモードを切り換えた場合、切換えが完了するまでは、低消費電力モードおよびほかのクロックモードへの切換えを行わないようにしてください。切換えの完了は、クロック選択レジスタ (CKSCR) の MCM ビットおよび SCM ビットを参照して確認してください。切換えが完了する前に、ほかのクロックモードおよび低消費電力モードへの切換えを行った場合、切り換わらない場合があります。

■ 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

[] 内の命令列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

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_nop0` 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_L.LPMCR.byte = 0x10;      /* LPMCR の TMD ビットに 0 をセット */
    _wait_nop0;
    _wait_nop0;
}
```

- (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

モード設定

この章では、モード設定と外部メモリアクセスについて説明します。

- 7.1 モード設定
- 7.2 モード端子 (MD2 ~ MD0)
- 7.3 モードデータ
- 7.4 外部メモリアクセス
- 7.5 モード設定の各モードにおける動作

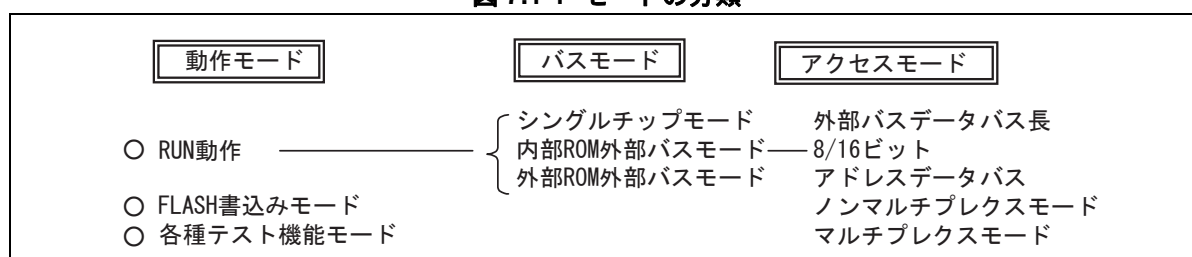
7.1 モード設定

F²MC-16LX には、アクセス方式、アクセス領域および試験の各々において各種のモードがあります。リセット時のモード端子の設定と、モードフェッチされたモードデータによって各モードが設定されます。

■ モード設定

F²MC-16LX では、アクセス方式、アクセス領域および試験の各々について各種のモードがあり、図 7.1-1 に示す分類になっています。

図 7.1-1 モードの分類



■ 動作モード

動作モードとは、デバイスの動作状態を制御するモードを示すもので、モード設定用端子 (MD_x) とモードデータ内の M_x ビットの内容で指定します。動作モードを選択することによって、通常動作 / 内部試験プログラムの起動 / 特殊試験機能の起動を行うことができます。

■ バスモード

バスモードとは、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を示すもので、モード設定用端子 (MD_x) とモードデータ内の M_x ビットの内容で指定します。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す。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。

■ アクセスモード

アクセスモードは、外部データバス幅を制御するモードを示すもので、モード設定用端子 (MD_x) とモードデータ内の S_x ビットで指定します。アクセスモードを選択することによって、外部データバスを 8 ビット長または 16 ビット長のいずれかを指定します。また、アドレスデータバスのノンマルチプレクスモードまたはマルチプレクスモードのいずれかを指定します。

7.2 モード端子 (MD2 ~ MD0)

モード端子は、(MD2 ~ MD0) の 3 本の外部端子で、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。

■ モード端子 (MD2 ~ MD0) の設定

モード端子 (MD2 ~ MD0) 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選択を行います。フラッシュROM内蔵品の場合は、内蔵ROMプログラムなどを書き込むための、フラッシュROM書き込みモードの指定もモード端子で行います。

モード端子の設定内容を、表 7.2-1 に示します。

表 7.2-1 モード端子の設定内容

P61	P60	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	外部ベクタモード 0	外部	マルチプレクスモード	リセットベクタ 8 ビットバス幅アクセス
-	-	0	0	1	外部ベクタモード 1	外部	マルチプレクスモード	リセットベクタ 16 ビットバス幅アクセス
-	-	0	1	0	外部ベクタモード 2	外部	ノンマルチプレクスモード	リセットベクタ 8 ビットバス幅アクセス
-	-	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	0	1	1	0	フラッシュシリアル書込み			
-	-	1	1	1	フラッシュライト書き込みモード	-	-	-

< 注意事項 >

MD2 ~ MD0:0= V_{SS} または 1= V_{CC} としてください。外部ベクタモード 2 は、データバス幅も 8 ビットが初期値です。データバス幅を 16 ビットにする場合は、モードデータをノンマルチプレクス外部データバス 16 ビットモードにすると LMBS の領域は 16 ビットサイズアクセスとなります。HMBS の領域を 16 ビットサイズアクセスにしたいときは、EPCR:HMBS ビットを設定し直してください。

7.3 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 コア内のモードレジスタに取り込みます。CPU は、この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。また、モードデータの設定は、リセットシーケンス以降に有効となります。

モードデータの構成を、図 7.3-1 に示します。

図 7.3-1 モードデータのビット構成



■ 各種モードの設定ビット (S1, S0)

S1, S0 ビットは、リセットシーケンス終了後のバスモードおよびアクセスモードを指定するビットです。

表 7.3-1 に、S1, S0 ビットの設定内容を示します。

表 7.3-1 S1, S0 ビットの設定内容

S1	S0	機能	
0	0	外部データバス 8 ビットモード	アドレスデータバス マルチプレクス
0	1	外部データバス 16 ビットモード	
1	0	外部データバス 8 ビットモード	アドレスデータバス ノンマルチプレクス
1	1	外部データバス 16 ビットモード	

■ バスモードの設定ビット (M1, M0)

M1, M0 ビットは、リセットシーケンス終了後の動作モードを指定するビットです。

表 7.3-2 に、M1, M0 ビットの設定内容を示します。

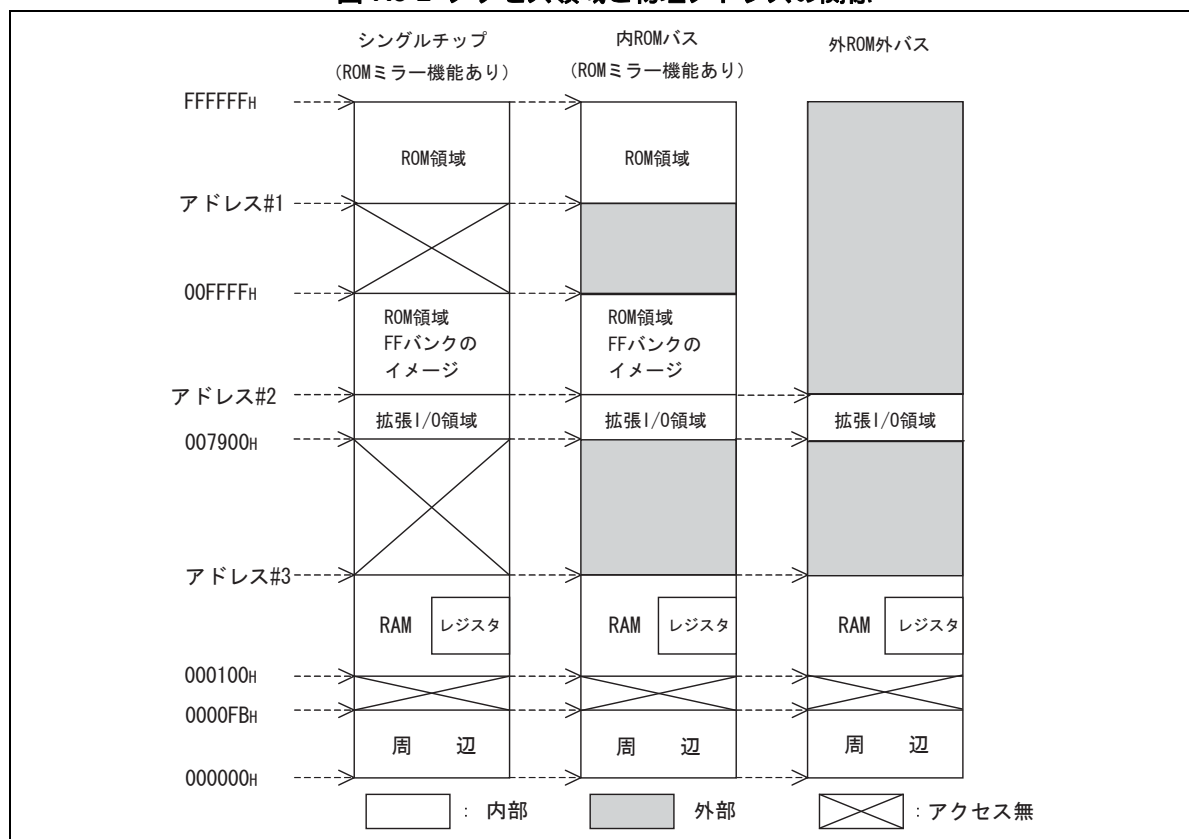
表 7.3-2 M1, M0 ビットの設定内容

M1	M0	機 能
0	0	シングルチップモード
0	1	内 ROM 外バスモード
1	0	外 ROM 外バスモード
1	1	(設定禁止)

■ アクセス領域と物理アドレスの関係

アクセス領域と物理アドレスの対応を、図 7.3-2 に示します。

図 7.3-2 アクセス領域と物理アドレスの関係



< 注意事項 >

「アドレス #X」は、各品種に依存して決まるアドレスとなります。詳細は、「付録 A メモリマップ」を参照してください。

■ モード端子とモードデータの関係 (推奨例)

表 7.3-3 に、モード端子とモードデータの関係を示します。

表 7.3-3 モード端子とモードデータの関係

モード	MD2	MD1	MD0	M1	M0	S1	S0
シングルチップ	0	1	1	0	0	X	X
内部 ROM 外部バスモード 8 ビット (アドレスデータマルチプレクス)	0	1	1	0	1	0	0
内部 ROM 外部バスモード 16 ビット (アドレスデータマルチプレクス)	0	1	1	0	1	0	1
内部 ROM 外部バスモード 8 ビット (アドレスデータノンマルチプレクス)	0	1	1	0	1	1	0
内部 ROM 外部バスモード 16 ビット (アドレスデータノンマルチプレクス)	0	1	1	0	1	1	1
外部 ROM 外部バスモード 16 ビットバス ベクタ 16 ビット幅 (アドレスデータマルチプレクス)	0	0	1	1	0	0	1
外部 ROM 外部バスモード 8 ビット (アドレスデータマルチプレクス)	0	0	0	1	0	0	0
外部 ROM 外部バスモード 8 ビット (アドレスデータノンマルチプレクス)	0	1	0	1	0	1	0

< 注意事項 >

上位アドレス A23 ~ A16 の出力を抑制した場合には、アクセスできる最大は 64 K バイトになります。

■ 各モードにおける外部端子の動作

表 7.3-4 に、ノンマルチプレクスモードおよびマルチプレクスモードにおける各外部端子の動作関係を示します。

表 7.3-4 各モードにおける外部端子の動作関係

	機能							
	ノンマルチプレクスモード				マルチプレクスモード			
	外部アドレス制御				外部アドレス制御			
	許可 (アドレス)		禁止 (アドレス)		許可 (アドレス)		禁止 (アドレス)	
	外部バス拡張		外部バス拡張		外部バス拡張		外部バス拡張	
	8 ビット	16 ビット	8 ビット	16 ビット	8 ビット	16 ビット	8 ビット	16 ビット
P07 ~ P00/ D07 ~ D00/ AD07 ~ AD00	D07 ~ D00				AD07 ~ AD00			
P17 ~ P10/ D15 ~ D08/ AD15 ~ AD08	ポート	D15 ~ D08	ポート	D15 ~ D08	A15 ~ A08	AD15 ~ AD08	A15 ~ A08	AD15 ~ AD08
P27 ~ P20	A23 ~ A16		ポート		A23 ~ A16		ポート	
P37 ~ P30	A07 ~ A00		A07 ~ A00		ポート			
P47 ~ P40	A15 ~ A08		A15 ~ A08					
ALE	ALE				ALE			
$\overline{\text{RD}}$	$\overline{\text{RD}}$				$\overline{\text{RD}}$			
P52/ $\overline{\text{WRL}}$	$\overline{\text{WRL}}$				$\overline{\text{WRL}}$			
P53/ $\overline{\text{WRH}}$	ポート	$\overline{\text{WRH}}$	ポート	$\overline{\text{WRH}}$	ポート	$\overline{\text{WRH}}$	ポート	$\overline{\text{WRH}}$
P54/HRQ	HRQ				HRQ			
P55/ $\overline{\text{HAK}}$	$\overline{\text{HAK}}$				$\overline{\text{HAK}}$			
P56/RDY	RDY				RDY			
P57/CLK	CLK				CLK			

- ・シングルチップモードは、すべてポートとして使用可能です。
- ・アドレス上位と $\overline{\text{WRL}}$, $\overline{\text{WRH}}$, $\overline{\text{HAK}}$, HRQ, RDY, CLK は、機能選択によってポートとして使用可能です。
- ・マルチプレクス使用時は、アドレス上位 A23 ~ A20 をポートまたは PPG3 ~ PPG0 として使用可能です。
- ・ノンマルチプレクス使用時には、リロードタイマ ch.0/1/2, UARTch.0/1 の使用はできません。アドレスとして機能します。

7.4 外部メモリアクセス

外部メモリアクセスのブロックダイアグラム, レジスタの構成 / 機能, および外部メモリアクセスの動作について説明します。

■ 外部メモリアクセスの入出力信号端子

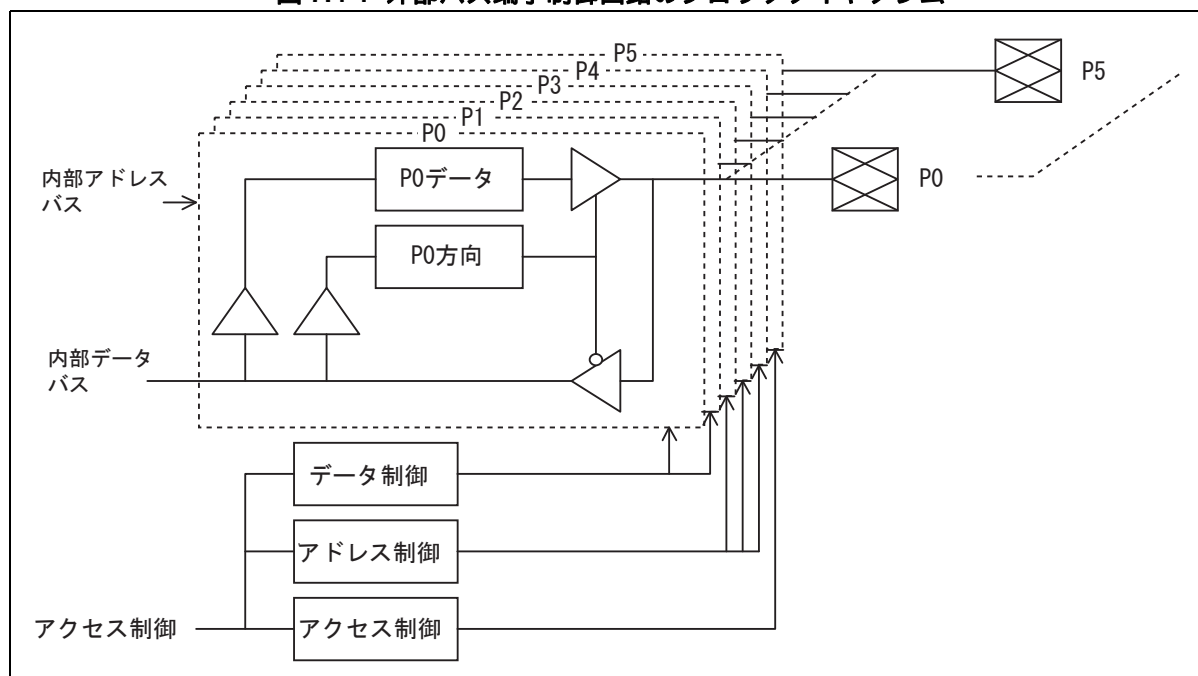
デバイス外部のメモリ / 周辺をアクセスするために, F²MC-16LX は以下のようなアドレス / データ / 制御信号を供給します。

- CLK(P57) : マシンサイクルクロック (KBP) を出力します。
- RDY(P56) : 外部レディ入力端子です。
- $\overline{\text{HAK}}$ (P55) : ホールドアクノリッジ出力端子です。
- HRQ(P54) : ホールドリクエスト入力端子です。
- $\overline{\text{WRH}}$ (P53) : データバス上位 8 ビットの書込み信号です。
- $\overline{\text{WRL}}$ (P52) : データバス下位 8 ビットの書込み信号です。
- $\overline{\text{RD}}$ (P51) : 読出し信号です。
- ALE(P50) : アドレスラッチ許可信号です (マルチプレクスモード時に有効)。

■ ブロックダイアグラム

図 7.4-1 に, 外部バス端子制御回路のブロックダイアグラムを示します。

図 7.4-1 外部バス端子制御回路のブロックダイアグラム



■ レジスタ一覧

図 7.4-2 に、外部バス端子制御回路のレジスタ一覧を示します。

図 7.4-2 外部バス端子制御回路のレジスタ一覧

0000A5 _H	bit	15	14	13	12	11	10	9	8	自動レディ 機能選択レジスタ (ARSR)
		予約	予約	HMR1	HMR0	-	-	LMR1	LMR0	
		(W) (0)	(W) (0)	(W) (1)	(W) (1)	(-) (-)	(-) (-)	(W) (0)	(W) (0)	
0000A6 _H	bit	7	6	5	4	3	2	1	0	外部アドレス 出力制御レジスタ (HACR)
		E23	E22	E21	E20	E19	E18	E17	E16	
		(W) (*)	(W) (*)	(W) (*)	(W) (*)	(W) (*)	(W) (*)	(W) (*)	(W) (*)	
0000A7 _H	bit	15	14	13	12	11	10	9	8	バス制御 信号選択レジスタ (EPCR)
		CKE	RYE	HDE	予約	HMBS	WRE	LMBS	-	
		(W) (1)	(W) (0)	(W) (0)	(W) (0)	(W) (*)	(W) (1)	(W) (0)	(-) (-)	

7.4.1 自動レディ機能選択レジスタ (ARSR)

自動レディ機能選択レジスタ (ARSR) の構成および機能について説明します。

■ 自動レディ機能選択レジスタ (ARSR)

自動レディ機能選択レジスタ (ARSR) のビット構成を、図 7.4-3 に示します。

図 7.4-3 自動レディ機能選択レジスタ (ARSR) のビット構成

bit	15	14	13	12	11	10	9	8	自動レディ 機能選択レジスタ (ARSR)
0000A5 _H	予約	予約	HMR1	HMR0	-	-	LMR1	LMR0	リード/ライト 初期値
	(W)	(W)	(W)	(W)	(-)	(-)	(W)	(W)	
	(0)	(0)	(1)	(1)	(-)	(-)	(0)	(0)	

以下に、自動レディ機能選択レジスタ (ARSR) の各ビットの機能を示します。

[bit15, bit14] 予約

予約ビットです。必ず "00_B" を書き込んでください。

[bit13, bit12] HMR1, HMR0

800000_H ~ FFFFFFF_H の領域に対する外部アクセスを行ったときの自動ウェイト機能を選択します。設定内容は、下表のとおりです。

HMR1	HMR0	設定
0	0	自動ウェイト禁止
0	1	外部アクセス時、1 マシンサイクルの自動ウェイト
1	0	外部アクセス時、2 マシンサイクルの自動ウェイト
1	1	外部アクセス時、3 マシンサイクルの自動ウェイト [初期値]

[bit11, bit10] 未定義ビット

ライトは動作に影響を与えません。

[bit9, bit8] LMR1, LMR0

007100_H ~ 7FFFFFF_H (ただし拡張 I/O 領域 007900_H ~ 007FFF_H は使用不可) の領域に対する外部アクセスを行ったときの自動ウェイト機能を選択します。設定内容は、下表のとおりです。

LMR1	LMR0	設定
0	0	自動ウェイト禁止 [初期値]
0	1	外部アクセス時、1 マシンサイクルの自動ウェイト
1	0	外部アクセス時、2 マシンサイクルの自動ウェイト
1	1	外部アクセス時、3 マシンサイクルの自動ウェイト

7.4.2 外部アドレス出力制御レジスタ (HACR)

外部アドレス出力制御レジスタの構成および機能について説明します。

■ 外部アドレス出力制御レジスタ (HACR)

外部アドレス出力制御レジスタのビット構成を、図 7.4-4 に示します。

図 7.4-4 外部アドレス出力制御レジスタ (HACR) のビット構成

bit	7	6	5	4	3	2	1	0	外部アドレス 出力制御レジスタ (HACR)
0000A6 _H	E23	E22	E21	E20	E19	E18	E17	E16	リード/ライト 初期値
	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
	(*)	(*)	(*)	(*)	(*)	(*)	(*)	(*)	

外部アドレス出力制御レジスタは、アドレス (A23 ~ A16) の外部への出力を制御するレジスタです。各ビットがそれぞれアドレス A23 ~ A16 に対応し、各アドレス出力端子を制御します。

0	対応する端子は、アドレス出力 (AXX) になります。[初期値]
1	対応する端子は I/O ポート (PXX) になります。

HACR レジスタは、デバイスがシングルチップモード時にはアクセスできません。その場合、HACR レジスタの値にかかわらず、全ポートが I/O ポートとして機能します。HACR レジスタの全ビットは書き込み専用で、読出し値は "1" となります。また、アドレス出力を選択してアドレスの出力を期待する場合は、DDR を "0" で使用してください。初期値は、内部ベクタモードで起動した場合のみ "1" となります。その他の場合は、"0" となります。

< 注意事項 >

PPG 使用時は、"1"(I/O ポート設定) として使用してください。

7.4.3 バス制御信号選択レジスタ (EPCR)

バス制御信号選択レジスタの構成および機能について説明します。

■ バス制御信号選択レジスタ (EPCR)

バス制御信号選択レジスタは、外部バスモード時のバス動作の制御機能を設定するレジスタです。

バス制御信号選択レジスタのビット構成を、図 7.4-5 に示します。

図 7.4-5 バス制御信号選択レジスタ (EPCR) のビット構成

bit	15	14	13	12	11	10	9	8	バス制御 信号選択レジスタ (EPCR)
0000A7 _H	CKE	RYE	HDE	予約	HMBS	WRE	LMBS	-	リード/ライト 初期値
	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(-)	
	(1)	(0)	(0)	(0)	(*)	(1)	(0)	(-)	

EPCR レジスタは、デバイスがシングルチップモード時には、アクセスできません。シングルチップモードの場合は、レジスタの値にかかわらず、全端子が I/O ポートとして機能します。EPCR レジスタの全ビットは書き込み専用で、読出し値は "1" となります。

以下に、バス制御信号選択レジスタの各ビットの機能を説明します。

[bit15] CKE

外部クロック (CLK) の出力を制御します。

0	I/O ポート (P57) 動作 (クロック禁止)
1	クロック信号 (CLK) 出力許可 [初期値]

[bit14] RYE

外部レディ (RDY) の入力を制御します。

0	I/O ポート (P56) 動作 (外部 RDY 入力禁止) [初期値]
1	外部レディ (RDY) 入力許可

[bit13] HDE

ホールド関係端子の入出力許可を指定するビットです。設定によって、ホールド要求入力 (HRQ) とホールドアクノリッジ出力 ($\overline{\text{HAK}}$) を制御します。

0	I/O ポート (P55, P54) 動作 (ホールド機能入出力禁止) [初期値]
1	ホールド要求 (HRQ) 入力 / ホールドアクノリッジ ($\overline{\text{HAK}}$) 出力許可

[bit12] 予約

予約ビットです。必ず "0" を書き込んでください。

[bit11] HMBS

外部データバス 16 ビットモード時に、800000_H ~ FFFFFFF_H の領域に対する外部バスをアクセスする場合のバス幅を指定します。

0	16 ビットバス幅アクセス [外部ベクタモード 1 の場合の初期値]
1	8 ビットバス幅アクセス [外部ベクタモード 0, 2 の場合の初期値]

[bit10] WRE

外部書込み信号 (外部データバス 16 ビットモードの場合には、 $\overline{\text{WRH}}/\overline{\text{WRL}}$ 両端子、外部データバス 8 ビットモードの場合は、 $\overline{\text{WRL}}$ 端子) の出力を制御します。

0	I/O ポート (P53, P52) 動作 (書込み信号出力禁止)
1	書込みストロブ信号 ($\overline{\text{WRH}}/\overline{\text{WRL}}$ または $\overline{\text{WRL}}$ のみ) の出力許可 [初期値]

[bit9] LMBS

外部データバス 16 ビットモード時に、007100_H ~ 7FFFFFF_H (ただし拡張 I/O 領域 007900_H ~ 007FFF_H は使用不可) の領域に対する外部バスをアクセスする場合のバス幅を指定します。

0	16 ビットバス幅アクセス [初期値]
1	8 ビットバス幅アクセス

< 注意事項 >

外部データバス 16 ビットモード時、WRE ビットで $\overline{\text{WRH}}/\overline{\text{WRL}}$ 機能を許可する場合は、P53/P52 を入力モードにしてください (DDR5 の bit3, bit2 は "0" にしてください)。

また、RYE, HDE ビットで RDY, HRQ 入力を許可した場合でも、そのポートの I/O ポート機能は有効になります。そのため、DDR5 中のそのポートに対応したビットは必ず "0" (入力モード) にしてください。

7.5 モード設定の各モードにおける動作

モード設定の各モードにおける動作について、タイミングチャートを示して説明します。

■ モードの種類

機能ごとに、以下の項目の動作を説明します。

- 外部メモリアクセス制御信号
 - 外部データバス 8 ビットモード (ノンマルチプレクスモード)
 - 外部データバス 8 ビットモード (マルチプレクスモード)
 - 外部データバス 16 ビットモード (ノンマルチプレクスモード)
 - 外部データバス 16 ビットモード (マルチプレクスモード)
- レディ機能
 - ノンマルチプレクスモード
 - マルチプレクスモード
- ホールド機能
 - ノンマルチプレクスモード
 - マルチプレクスモード

7.5.1 外部メモリアクセス制御信号

外部メモリへのアクセスは、レディ機能を使用しない場合に 3 サイクルで行います。

■ 外部メモリアクセス制御信号

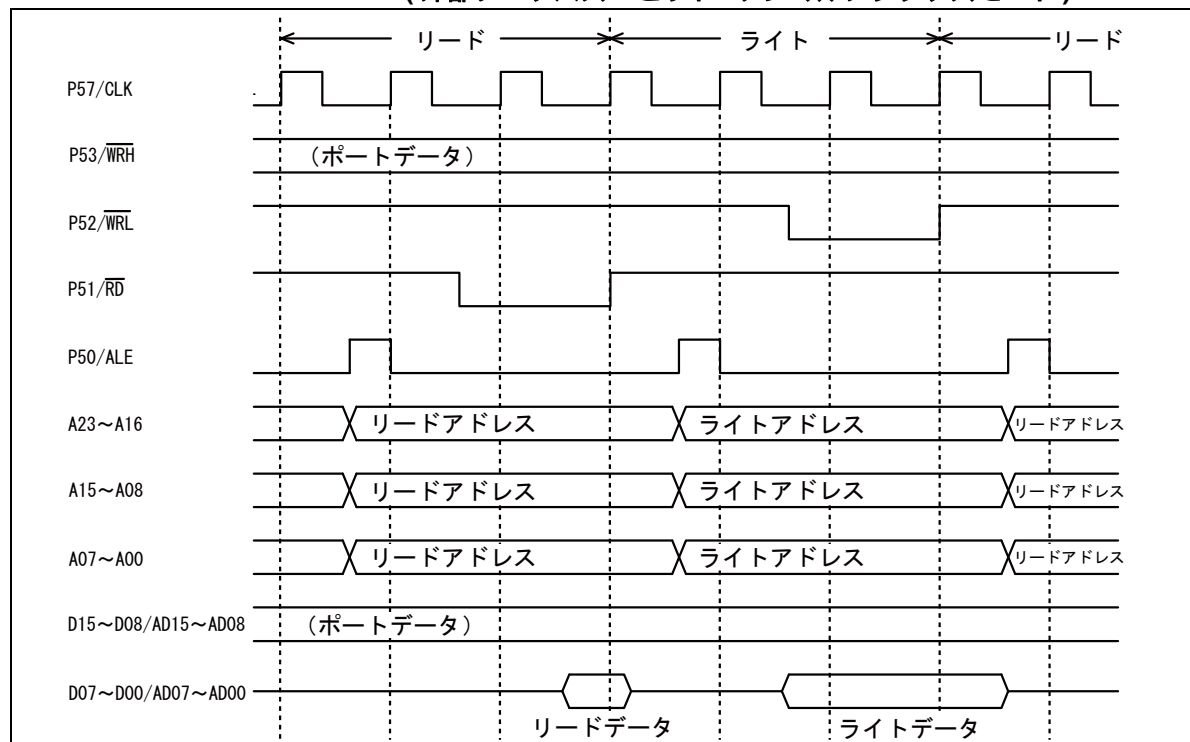
各モードにおける外部アクセスのタイミングチャートを、図 7.5-1 ~ 図 7.5-4 に示します。外部データバス 16 ビットモードでの 8 ビットバス幅アクセスは、8 ビット幅の周辺チップと 16 ビット幅の周辺チップを混在して外部バスに接続した場合に、8 ビット幅の周辺チップを読み出し / 書込みをするための機能です。8 ビットバス幅アクセスは、データバスの下位 8 ビットを使用して実行されるので、8 ビット幅の周辺チップはデータの下位 8 ビットに接続してください。外部データバス 16 ビットモードで 16 ビットバス幅アクセスを行うか、8 ビットバス幅アクセスを行うかは、EPCR の HMBS/LMBS ビットを指定することによります。なお、マルチプレクスモード時において、アドレス出力と ALE のアサート出力のみを行い、 $\overline{\text{RD}}/\overline{\text{WRL}}/\overline{\text{WRH}}$ をアサートしないことにより実際には、バス動作を行わない場合があります。

< 注意事項 >

ALE 信号のみで周辺チップへのアクセスを実行することがないようにしてください。

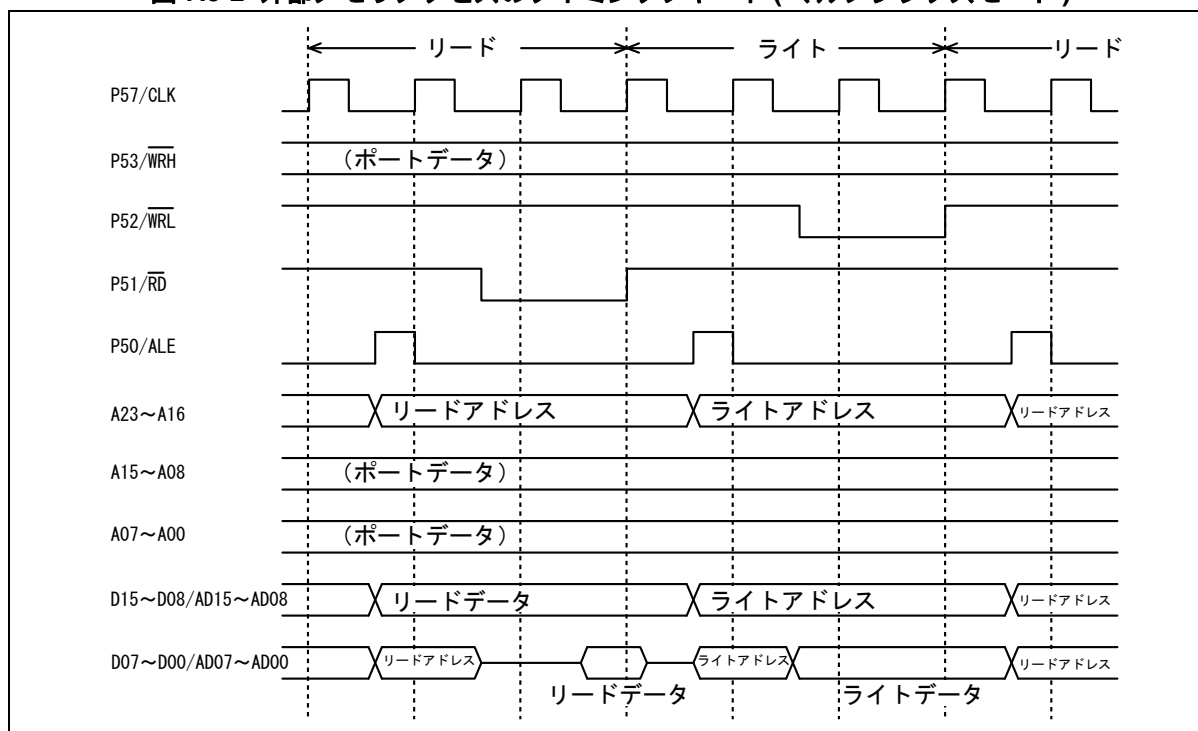
● 外部データバス 8 ビットモード (ノンマルチプレクスモード)

図 7.5-1 外部メモリアクセスのタイミングチャート
(外部データバス 8 ビット / ノンマルチプレクスモード)

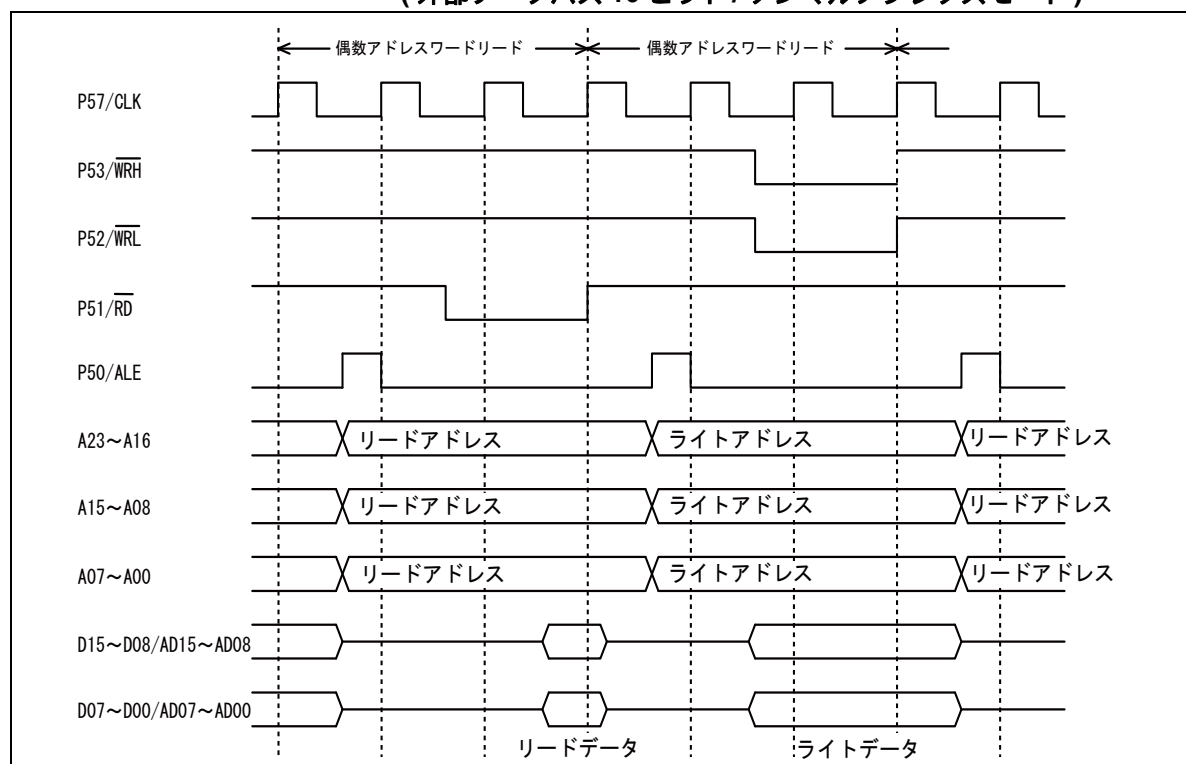


● 外部データバス 8 ビットモード (外部データバス 8 ビット / マルチプレクスモード)

図 7.5-2 外部メモリアクセスのタイミングチャート (マルチプレクスモード)

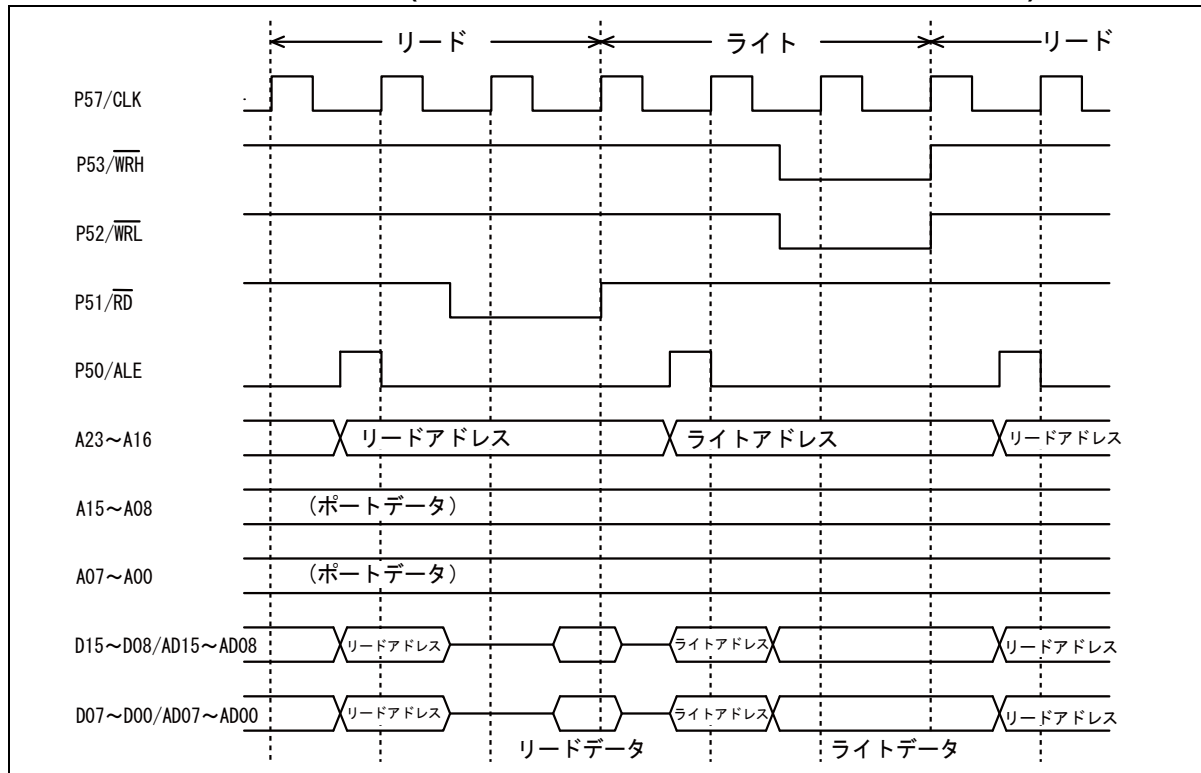


● 外部データバス 16 ビットモード (外部データバス 16 ビット / ノンマルチプレクスモード)

図 7.5-3 外部メモリアクセスのタイミングチャート
(外部データバス 16 ビット / ノンマルチプレクスモード)

■ 外部 16 ビットバスモード (外部データバス 16 ビット/マルチプレクスモード)

図 7.5-4 外部メモリアクセスのタイミングチャート
(外部データバス 16 ビット/マルチプレクスモード)



7.5.2 レディ機能

P56/RDY 端子, または自動レディ機能選択レジスタ (ARSR) の設定により, 低速のメモリや周辺回路へのアクセスが可能となります。バス制御信号選択レジスタ (EPCR) 中の RYE ビットが "1" に設定されている場合, 外部領域へのアクセス時に P56/RDY 端子に "L" レベルが入力されている期間ウェイトサイクルとなり, アクセスサイクルを延ばすことができます。

■ レディ機能

F²MC-16LX は, 外部メモリ用のオートレディ機能を 2 種類内蔵しています。オートレディ機能は, アドレス 007100_H ~ 7FFFFFF_H (ただし拡張 I/O 領域 007900_H ~ 007FFF_H は使用不可) 間に配置した下位アドレス外部領域へのアクセスが発生したとき, アドレス 800000_H ~ FFFFFFF_H 間に配置した上位アドレス外部領域へのアクセスが発生したときに, 外部回路なしで自動的に 1 ~ 3 サイクルのウェイトサイクルを挿入し, アクセスサイクルを延ばすことができる機能です。ARSR 中の LMR1/LMR0 ビット (下位アドレス外部領域) および ARSR 中の HMR1/HMR0 ビット (上位アドレス外部領域) の設定により起動します。

外部メモリ用, 外部 I/O 用のオートレディどちらでも EPCR 中の RYE ビットが "1" に設定されている場合, 上記オートレディによるウェイトサイクル終了後に P56/RDY 端子に "L" レベルが入力されていると, そのままウェイトサイクルを継続します。

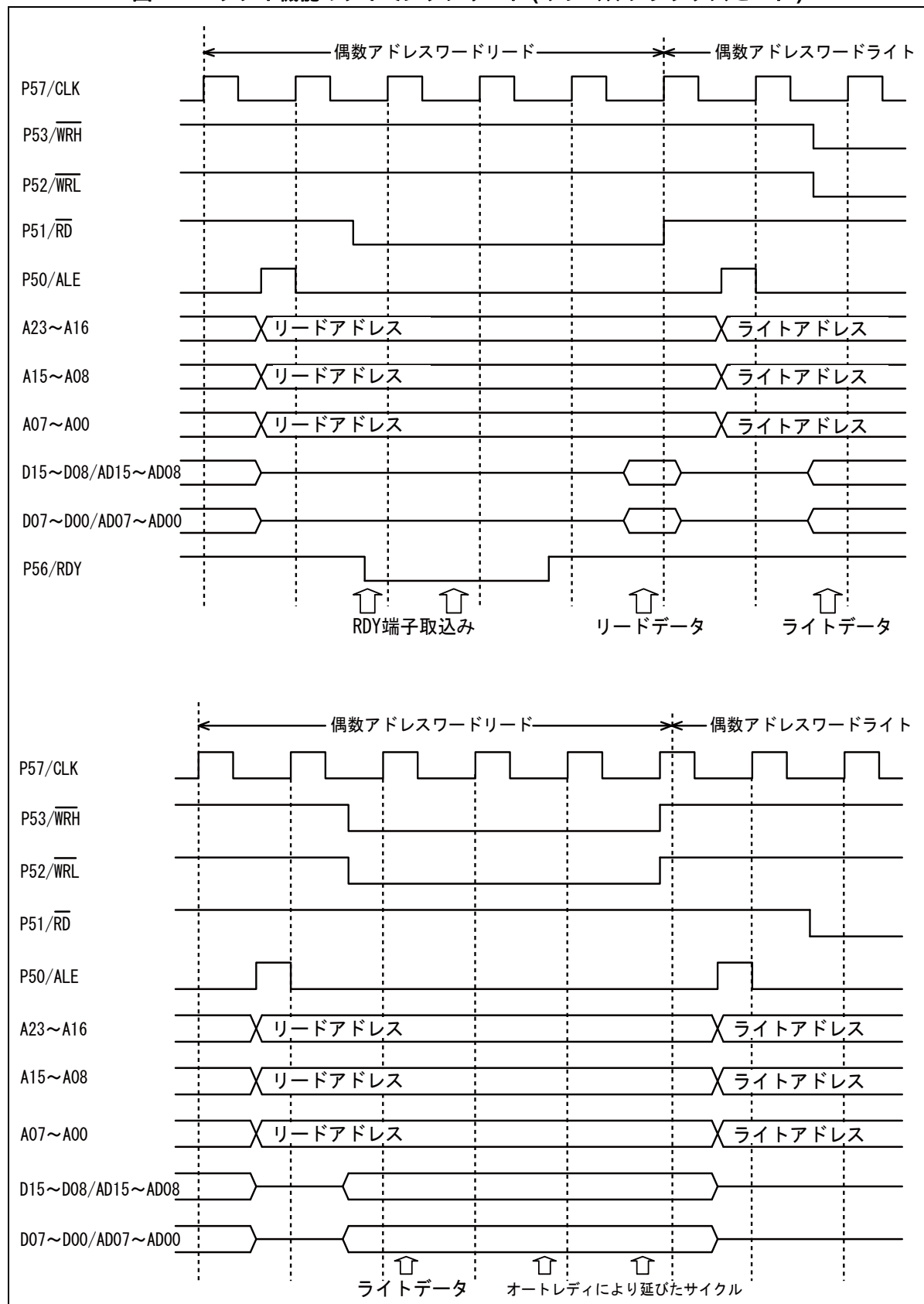
以下に, ノンマルチプレクスモードおよびマルチプレクスモードにおけるレディ機能のタイミングチャートを示します。どちらのモードも上の図がレディ機能を設定しない場合, 下の図がレディ機能を設定した場合です。

< 注意事項 >

RDY 端子から入力する場合, AC 規格を満たさないとき, デバイスは暴走する恐れがありますので, 注意してください。

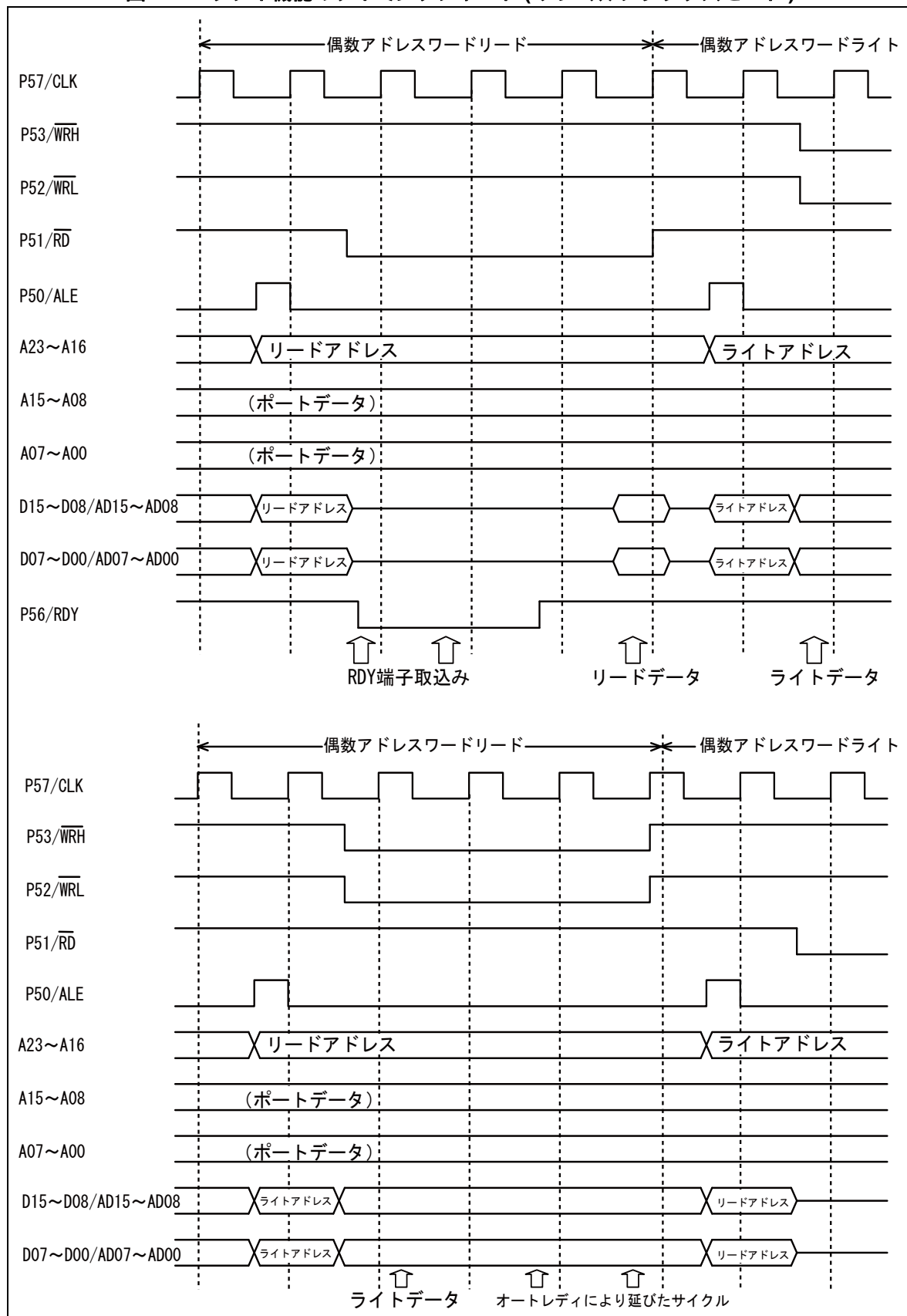
● ノンマルチプレクスモード

図 7.5-5 レディ機能のタイミングチャート (ノンマルチプレクスモード)



● マルチプレクスモード

図 7.5-6 レディ機能のタイミングチャート (ノンマルチプレクスモード)



7.5.3 ホールド機能

ホールド機能の動作を、タイミングチャートを示して説明します。

■ ホールド機能の動作

EPCR 中の HDE ビットが "1" に設定されている場合、P54/HRQ、P55/ $\overline{\text{HAK}}$ 両端子による外部バスのホールド機能が有効となります。P54/HRQ 端子に "H" レベルを入力すると、CPU の命令終了時（ストリング命令の場合は 1 エLEMENTデータの処理の終了後）にホールド状態となり、P55/ $\overline{\text{HAK}}$ より "L" レベルを出力して以下の端子をハイインピーダンス状態にします。

● ノンマルチプレクスモード

- アドレス出力 : A23 ~ A00
- データ入出力 : D15/AD15 ~ D00/AD00
- バス制御信号 : P51/ $\overline{\text{RD}}$, P52/ $\overline{\text{WRL}}$, P53/ $\overline{\text{WRH}}$

● マルチプレクスモード

アドレス出力 : A23 ~ A16

データ入出力 : D15/AD15 ~ D00/AD00

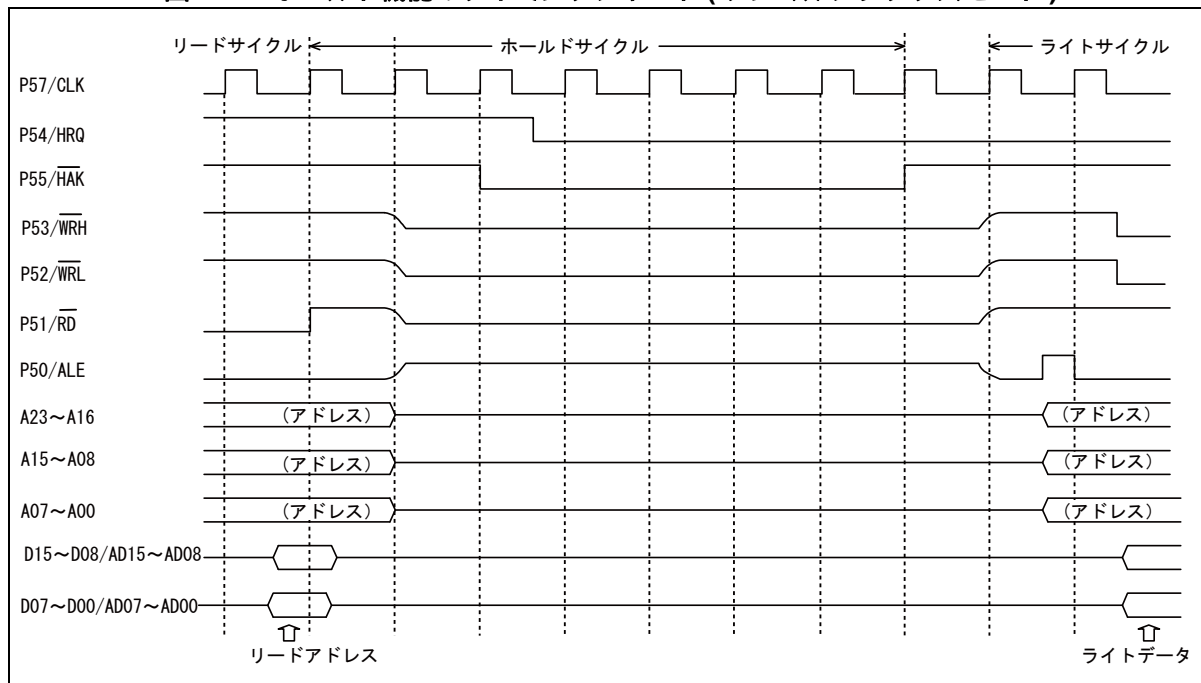
バス制御信号 : P51/ $\overline{\text{RD}}$, P52/ $\overline{\text{WRL}}$, P53/ $\overline{\text{WRH}}$

これにより、デバイス外部回路により外部バスを使用することが可能となります。P54/HRQ 端子に "L" レベルを入力すると、P55/ $\overline{\text{HAK}}$ 端子は "H" レベル出力となり、外部端子状態が復活して CPU は動作を再開します。STOP 状態においては、ホールド要求は受け付けません。

■ ノンマルチプレクスモード

図 7.5-7 に、外部データバス 16 ビットモードにおけるノンマルチプレクスモードのホールド機能のタイミングチャートを示します。

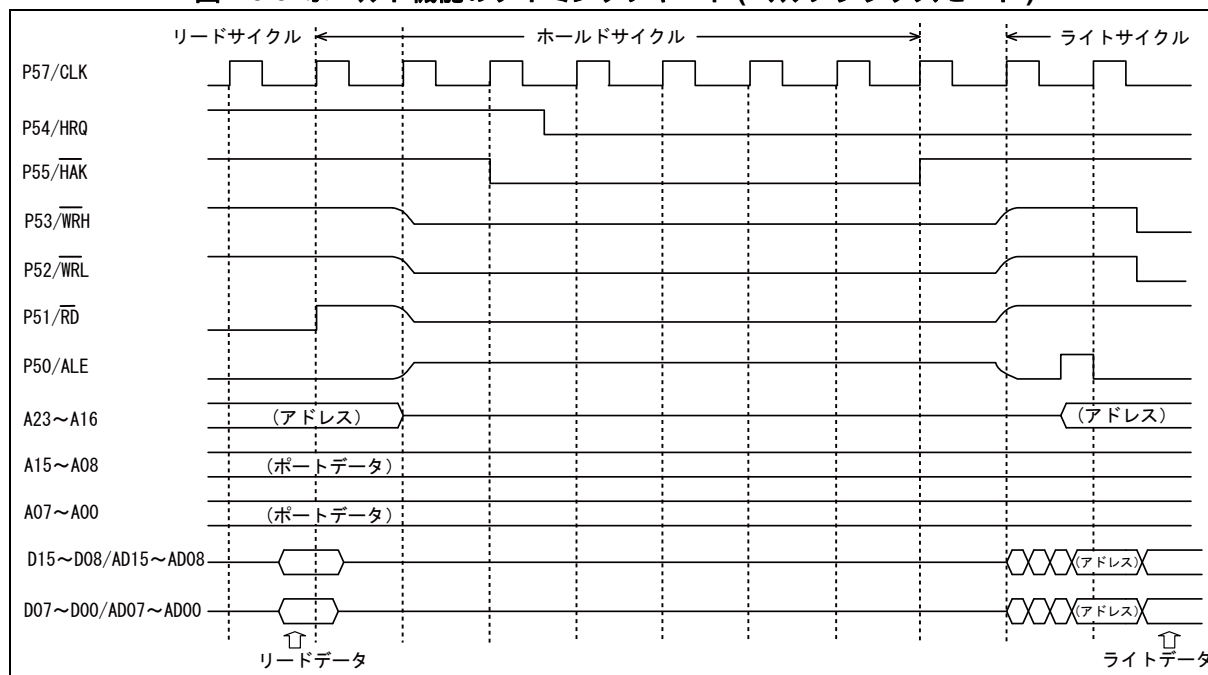
図 7.5-7 ホールド機能のタイミングチャート (ノンマルチプレクスモード)



■ マルチプレクスモード

図 7.5-8 に、外部データバス 16 ビットモードにおけるマルチプレクスモードのホールド機能のタイミングチャートを示します。

図 7.5-8 ホールド機能のタイミングチャート (マルチプレクスモード)



< 注意事項 >

P54/HRQ 端子に "H" レベルを入力した場合 , P55/HAK 端子が "L" レベルになるまで P54/HRQ 端子を "H" レベルに保持してください。

第8章

I/O ポート

この章では、I/O ポートで使用するレジスタの構成および機能について説明します。

8.1 I/O ポートの機能

8.2 I/O ポートのレジスタ

8.1 I/O ポートの機能

I/O ポートの機能概要を示します。

■ I/O ポートの機能

I/O ポートは、ポートデータレジスタ (PDR) によって、CPU からデータを I/O 端子に出力したり、I/O 端子に入力された信号を CPU に取込んだりする機能をもっています。また、ポートはポート方向レジスタ (DDR) によって、I/O 端子の入出力の方向をビット単位で任意に設定できます。

MB90330A シリーズには、72 本の入出力と、22 本のオープンドレインの出力端子があります。

P07 ~ P00, P17 ~ P10, P27 ~ P20, P37 ~ P30, P47 ~ P40, P57 ~ P50, P77 ~ P70,
P87 ~ P80, P95 ~ P90, PB6, PB5 が入出力ポートで、P67 ~ P60, P96, PA7 ~ PA0,
PB4 ~ PB0 がオープンドレイン端子となっています。

8.2 I/O ポートのレジスタ

I/O ポートで使用するレジスタの構成, および機能について説明します。

■ I/O ポートのレジスタ

I/O ポートには以下のレジスタがあります。

- ポートデータレジスタ (PDR0 ~ PDRB)
- ポート方向レジスタ (DDR0 ~ DDRB)
- 入力抵抗レジスタ (RDR0, RDR1)
- 出力端子レジスタ (ODR4)
- アナログ入力許可レジスタ (ADER0, ADER1)

8.2.1 ポートデータレジスタ (PDR0 ~ PDRB)

ポートデータレジスタ (PDR0 ~ PDRB) の構成および機能について説明します。

■ ポートデータレジスタ (PDR0 ~ PDRB)

図 8.2-1 に、ポートデータレジスタ (PDR0 ~ PDRB) の一覧を示します。

図 8.2-1 ポートデータレジスタ (PDR0 ~ PDRB) の一覧

PDR0	bit	7	6	5	4	3	2	1	0	初期値	アクセス
アドレス : 000000 _H		P07	P06	P05	P04	P03	P02	P01	P00	XXXXXXXX _B	R/W *
PDR1	bit	15	14	13	12	11	10	9	8		
アドレス : 000001 _H		P17	P16	P15	P14	P13	P12	P11	P10	XXXXXXXX _B	R/W *
PDR2	bit	7	6	5	4	3	2	1	0		
アドレス : 000002 _H		P27	P26	P25	P24	P23	P22	P21	P20	XXXXXXXX _B	R/W *
PDR3	bit	15	14	13	12	11	10	9	8		
アドレス : 000003 _H		P37	P36	P35	P34	P33	P32	P31	P30	XXXXXXXX _B	R/W *
PDR4	bit	7	6	5	4	3	2	1	0		
アドレス : 000004 _H		P47	P46	P45	P44	P43	P42	P41	P40	XXXXXXXX _B	R/W *
PDR5	bit	15	14	13	12	11	10	9	8		
アドレス : 000005 _H		P57	P56	P55	P54	P53	P52	P51	P50	XXXXXXXX _B	R/W *
PDR6	bit	7	6	5	4	3	2	1	0		
アドレス : 000006 _H		P67	P66	P65	P64	P63	P62	P61	P60	XXXXXXXX _B	R/W *
PDR7	bit	15	14	13	12	11	10	9	8		
アドレス : 000007 _H		P77	P76	P75	P74	P73	P72	P71	P70	XXXXXXXX _B	R/W *
PDR8	bit	7	6	5	4	3	2	1	0		
アドレス : 000008 _H		P87	P86	P85	P84	P83	P82	P81	P80	XXXXXXXX _B	R/W *
PDR9	bit	15	14	13	12	11	10	9	8		
アドレス : 000009 _H		-	P96	P95	P94	P93	P92	P91	P90	-XXXXXXXX _B	R/W *
PDRA	bit	7	6	5	4	3	2	1	0		
アドレス : 00000A _H		PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	XXXXXXXX _B	R/W *
PDRB	bit	7	6	5	4	3	2	1	0		
アドレス : 00000C _H		-	PB6	PB5	PB4	PB3	PB2	PB1	PB0	-XXXXXXXX _B	R/W *

*: 入出力ポートへの R/W アクセスは、メモリへの R/W アクセスとは多少動作が異なります。

以下のような動作をしますので、注意してください。

- 入力モード
 - 読出し時: 対応する端子のレベルが読み出されます。
 - 書込み時: 出力用のラッチに書き込まれます。
- 出力モード
 - 読出し時: データレジスタラッチの値が読み出されます。
 - 書込み時: 対応する端子に出力されます。

8.2.2 ポート方向レジスタ (DDR0 ~ DDRB)

ポート方向レジスタの構成および機能について説明します。

■ ポート方向レジスタ (DDR0 ~ DDRB)

図 8.2-2 に、ポート方向レジスタ (DDR0 ~ DDRB) の一覧を示します。

図 8.2-2 ポート方向レジスタ (DDR0 ~ DDRB) の一覧

DDR0	bit	7	6	5	4	3	2	1	0	初期値	アクセス
アドレス : 000010 _H		D07	D06	D05	D04	D03	D02	D01	D00	00000000	R/W
DDR1	bit	15	14	13	12	11	10	9	8		
アドレス : 000011 _H		D17	D16	D15	D14	D13	D12	D11	D10	00000000	R/W
DDR2	bit	7	6	5	4	3	2	1	0		
アドレス : 000012 _H		D27	D26	D25	D24	D23	D22	D21	D20	00000000	R/W
DDR3	bit	15	14	13	12	11	10	9	8		
アドレス : 000013 _H		D37	D36	D35	D34	D33	D32	D31	D30	00000000	R/W
DDR4	bit	7	6	5	4	3	2	1	0		
アドレス : 000014 _H		D47	D46	D45	D44	D43	D42	D41	D40	00000000	R/W
DDR5	bit	15	14	13	12	11	10	9	8		
アドレス : 000015 _H		D57	D56	D55	D54	D53	D52	D51	D50	00000000	R/W
DDR6	bit	7	6	5	4	3	2	1	0		
アドレス : 000016 _H		D67	D66	D65	D64	D63	D62	D61	D60	00000000	R/W
DDR7	bit	15	14	13	12	11	10	9	8		
アドレス : 000017 _H		D77	D76	D75	D74	D73	D72	D71	D70	00000000	R/W
DDR8	bit	7	6	5	4	3	2	1	0		
アドレス : 000018 _H		D87	D86	D85	D84	D83	D82	D81	D80	00000000	R/W
DDR9	bit	15	14	13	12	11	10	9	8		
アドレス : 000019 _H		-	D96	D95	D94	D93	D92	D91	D90	-00000000	R/W
DDRA	bit	7	6	5	4	3	2	1	0		
アドレス : 00001A _H		DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0	00000000	R/W
DDRB	bit	15	14	13	12	11	10	9	8		
アドレス : 00000D _H		-	DB6	DB5	DB4	DB3	DB2	DB1	DB0	-00000000	R/W

● 各端子がポートとして機能している場合

各端子がポートとして機能している場合、対応する各端子を以下のように制御します。

- 0: 入力モード
- 1: 出力モード

リセットで "0" になります。

< 注意事項 >

リードモディファイライト (RMW) 系の命令 (ビットセットなどの命令) を使用して DDR0 ~ DDRB レジスタをアクセスすると、命令で注目しているビットは所定の値になりますが、それ以外のビットで入力設定してあるものに対応する出力レジスタの内容は、そのときの端子の入力値に書き換わります。このため、入力として使用していた端子を出力に切り換えるときには、はじめに希望値を PDR に書き込んでから DDR をセットし、そのあとで出力に切り換えるようにしてください。

8.2.3 その他のレジスタ

ポートデータレジスタ (PDR0 ~ PDRB), ポート方向レジスタ (DDR0 ~ DDRB) 以外のレジスタの構成および機能について説明します。

■ ポート 0, 1 プルアップ抵抗レジスタ (RDR0, RDR1)

プルアップ抵抗レジスタ (RDR0, RDR1) のビット構成を図 8.2-3 に示します。

図 8.2-3 プルアップ抵抗レジスタ (RDR0, RDR1) のビット構成

PDR0	bit	7	6	5	4	3	2	1	0	初期値	アクセス
アドレス : 00001C _H		RD07	RD06	RD05	RD04	RD03	RD02	RD01	RD00	00000000	R/W
PDR1	bit	15	14	13	12	11	10	9	8		
アドレス : 00001D _H		RD17	RD16	RD15	RD14	RD13	RD12	RD11	RD10	00000000	R/W

プルアップ抵抗レジスタ (RDR0, RDR1) は、入力モード時にプルアップ抵抗をなしとするか、ありとするかを設定します。

- 0: 入力モード時プルアップ抵抗なし
- 1: 入力モード時プルアップ抵抗あり

RDR0, RDR1 レジスタは、出力モード時には機能をもちません (プルアップ抵抗なし)。

入出力モードは、方向レジスタ (DDR) の設定で決まります。

ストップ時 (SPL=1) のときはプルアップ抵抗なしになります (ハイインピーダンス)。

外部バスで使用する場合は、この機能は使用禁止です。RDR0, RDR1 レジスタへは書き込まないようにしてください。

■ ポート 4 出力端子レジスタ (ODR4)

出力端子レジスタ (ODR4) のビット構成を図 8.2-4 に示します。

図 8.2-4 出力端子レジスタ (ODR4) のビット構成

PDR0	bit	15	14	13	12	11	10	9	8	初期値	アクセス
アドレス : 00001B _H		OD47	OD46	OD45	OD44	OD43	OD42	OD41	OD40	00000000	R/W

出力端子レジスタ (ODR4) は、出力モード時のオープンドレイン制御を行います。

- 0: 出力モード時に標準出力ポートになります。
- 1: 出力モード時にオープンドレイン出力ポートになります。

出力端子レジスタ (ODR4) は、入力モード時は機能をもちません (出力ハイインピーダンス)。

入出力モードは、方向レジスタ (DDR) の設定で決まります。

外部バスで使用する場合は、この機能は使用禁止です。出力端子レジスタ (ODR4) へは書き込まないようにしてください。

■ アナログ入力許可レジスタ (ADER0, ADER1)

アナログ入力許可レジスタ (ADER0, ADER1) のビット構成を図 8.2-5 に示します。

図 8.2-5 アナログ入力許可レジスタ (ADER0, ADER1) のビット構成

PDR0	bit	7	6	5	4	3	2	1	0	初期値	アクセス
アドレス : 00001E _H		ADE07	ADE06	ADE05	ADE04	ADE03	ADE02	ADE01	ADE00	11111111	R/W
PDR1	bit	15	14	13	12	11	10	9	8		
アドレス : 00001F _H		ADE15	ADE14	ADE13	ADE12	ADE11	ADE10	ADE09	ADE08	11111111	R/W

アナログ入力許可レジスタ (ADER0, ADER1) は , ポート 7, 8 の各端子を以下のように制御します。

- 0: ポート入出力モードになります。
- 1: アナログ入力モードになります。リセットで "1" になります。

第9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。

- 9.1 タイムベースタイマの概要
- 9.2 タイムベースタイマの構成
- 9.3 タイムベースタイマ制御レジスタ (TBTC)
- 9.4 タイムベースタイマの割込み
- 9.5 タイムベースタイマの動作
- 9.6 タイムベースタイマ使用上の注意
- 9.7 タイムベースタイマのプログラム例

9.1 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（原発振の2分周）に同期してカウントアップする18ビットのフリーランカウンタ（タイムベースカウンタ）で、4種類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があります。また、発振安定待ち時間のタイマ出力やウォッチドッグタイマなどの動作クロックを供給する機能もあります。

■ インターバルタイマ機能

インターバルタイマ機能は、一定時間間隔で繰り返し割込み要求を発生する機能です。

- タイムベースカウンタのインターバルタイマ用ビットがオーバフローすると割込み要求を発生します。
- インターバルタイマ用のビット（インターバル時間）は、4種類の中から選択できます。

表 9.1-1 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 9.1-1 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(0.33 μs)	$2^{12}/\text{HCLK}$ (約 0.68 ms)
	$2^{14}/\text{HCLK}$ (約 2.7 ms)
	$2^{16}/\text{HCLK}$ (約 10.9 ms)
	$2^{19}/\text{HCLK}$ (約 87.4 ms)

HCLK: 発振クロック

() 内は発振クロック 6 MHz 動作時の値です。

■ 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や、一部の周辺機能に対する動作クロックを供給する機能です。表 9.1-2 に、タイムベースタイマから各周辺に供給されるクロックの周期を示します。

表 9.1-2 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 1.4 ms)	セラミック振動子用発振安定待ち時間
	$2^{15}/\text{HCLK}$ (約 5.5 ms)	水晶振動子用発振安定待ち時間
	$2^{17}/\text{HCLK}$ (約 21.8 ms)	
ウォッチドッグタイマ	$2^{12}/\text{HCLK}$ (約 0.68 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 2.7 ms)	
	$2^{16}/\text{HCLK}$ (約 10.9 ms)	
	$2^{19}/\text{HCLK}$ (約 87.4 ms)	

HCLK: 発振クロック

() 内は発振クロック 6 MHz 動作時の値です。

< 参考 >

発振開始直後は発振周期が不安定なため、発振安定待ち時間は目安となります。

9.2 タイムベースタイマの構成

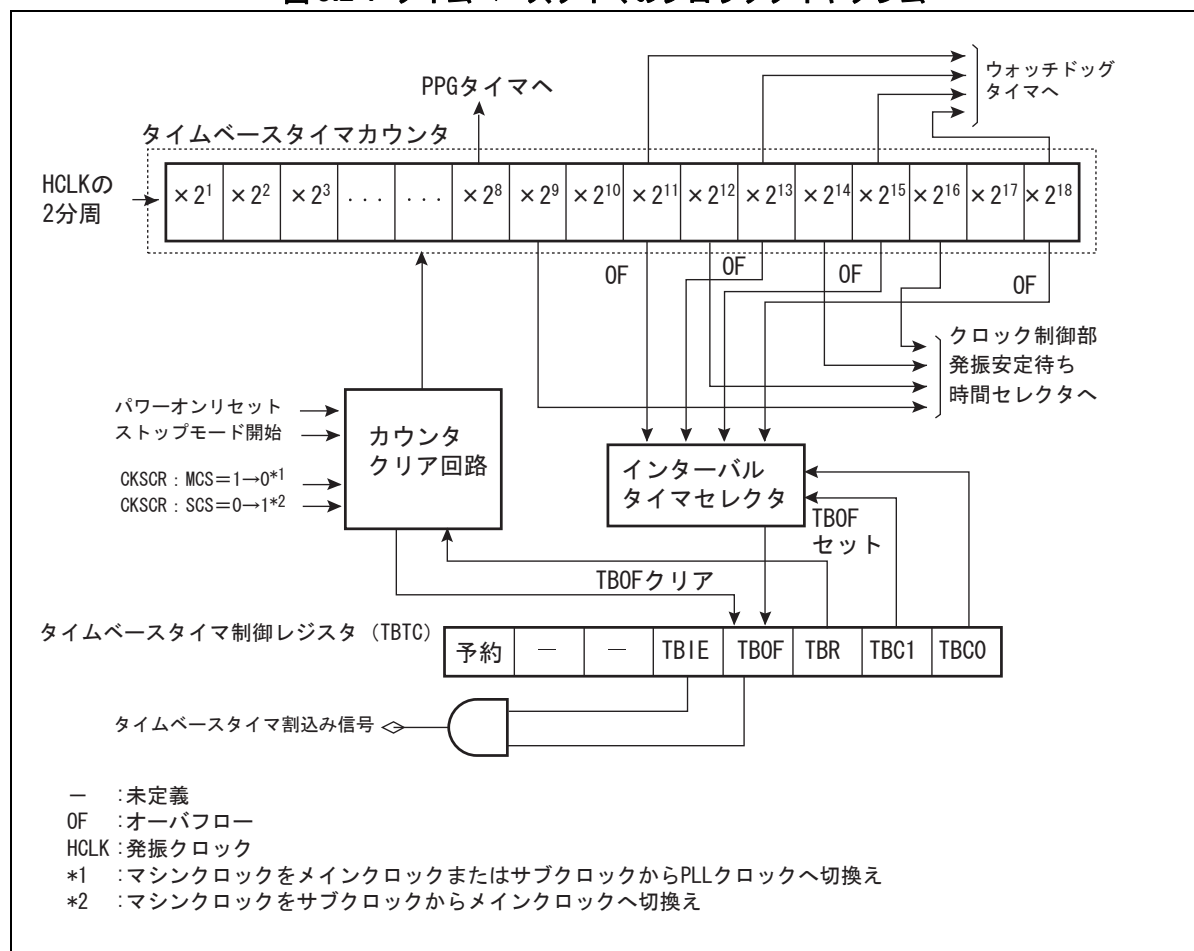
タイムベースタイマは、次の4つのブロックで構成されています。

- 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 9.2-1 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 9.2-1 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

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する、18ビットのアップカウンタです。

- カウンタクリア回路

タイムベースタイマ制御レジスタ (TBTC) のタイムベースタイマ初期化ビット (TBR) への "0" 書込み, パワーオンリセット, ストップモードへの移行, メインクロックモードまたはサブクロックから PLL クロックモードへの切換え, サブクロックからメインクロックモードへの切換えでカウンタをクリアする回路です。

- インターバルタイマセクタ

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, カウンタのクリア, 割込み要求の制御および状態の確認をします。

9.3 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) は、インターバル時間の選択、タイムベースタイマカウンタのクリア、割込み制御および状態の確認をするレジスタです。

■ タイムベースタイマ制御レジスタ (TBTC)

図 9.3-1 タイムベースタイマ制御レジスタ (TBTC) のビット構成

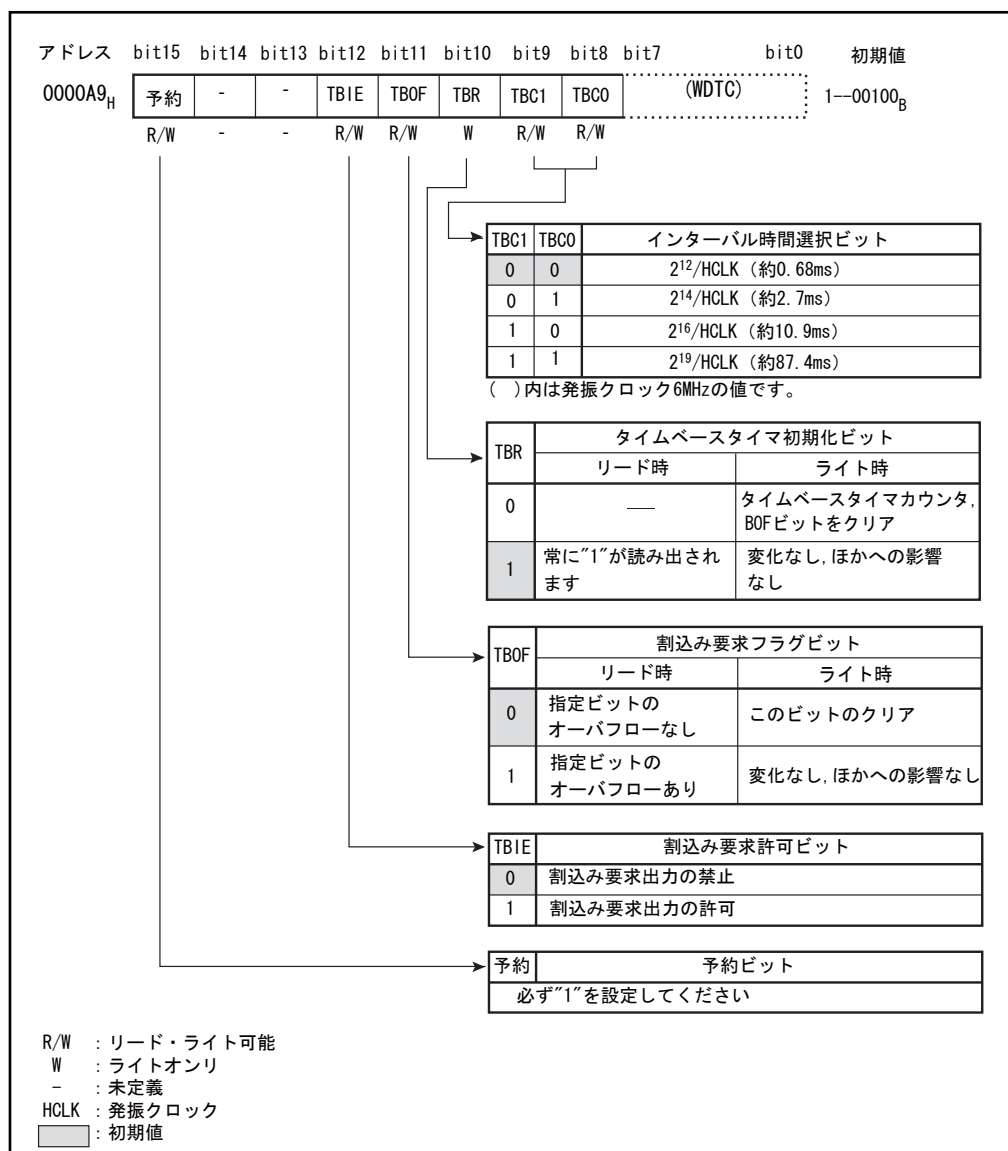


表 9.3-1 タイムベースタイマ制御レジスタ (TBTC) の各ビットの機能

ビット名		機能
bit15	予約: 予約ビット	(注意事項) 必ず "1" を書き込んでください。
bit14, bit13	未定義 ビット	・ 読出し時の値は不定です。 ・ 書込みは動作に影響を与えません。
bit12	TBIE: 割込み要求 許可ビット	・ CPU への割込み要求出力の許可 / 禁止をするビットです。 ・ TBIE ビットと割込み要求フラグビット (TBOF) が "1" の場合、割込み要求を出力します。
bit11	TBOF: 割込み要求 フラグビット	・ タイムベースタイマカウンタ指定ビットがオーバフローすると "1" にセットされます。 ・ TBOF ビットと割込み要求許可ビット (TBIE) が "1" の場合、割込み要求を出力します。 ・ 書込み時は "0" でクリアされ、"1" では変化せず、ほかへの影響はありません。 (注意事項) ・ TBOF ビットをクリアする場合には、割込み要求許可ビット (TBIE) 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) の指定で、タイムベースタイマ割込みが禁止されている状態にしてください。 ・ "0" の書込み、ストップモードへの遷移、サブクロックモードからメインクロックモードへの遷移、メインクロックモードから PLL クロックモードへの遷移およびタイムベースタイマ初期化ビット (TBR) への "0" 書込みおよびリセットにより "0" にクリアされます。
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するビットです。 ・ "0" を書き込むとカウンタがクリアされ、さらに TBOF ビットをクリアします。"1" では変化せず、ほかへの影響はありません。 (参考) 読出し値は、常に "1" です。
bit9, bit8	TBC1, TBC0: インターバル 時間選択 ビット	・ インターバルタイマの周期を選択するビットです。 ・ 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。

9.4 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の指定ビットのオーバフローで割込み要求を発生できます (インターバルタイマ機能)。

■ タイムベースタイマの割込み

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用のビットがオーバフローすると、タイムベースタイマ制御レジスタ (TBTC) の割込み要求フラグビット (TBOF) が "1" にセットされます。その場合、割込み要求許可ビット (TBIE) を "1" にして割込み要求を許可していると、CPU に割込み要求が発生します。割込み処理ルーチンで TBOF ビットに "0" を書き込んで、割込み要求をクリアしてください。なお、TBOF ビットは、割込み要求許可ビット (TBIE) の値に関係なく、指定したビットがオーバフローするとセットされます。

< 注意事項 >

タイムベースタイマ制御レジスタ (TBTC) の割込み要求フラグビット (TBOF) をクリアする場合には、割込み要求許可ビット (TBIE) 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) の設定で、タイムベースタイマ割込みが禁止されている状態で行ってください。

< 参考 >

- TBOF ビットが "1" の場合に、TBIE ビットを禁止から許可 (0 → 1) にすると直ちに割込み要求が発生します。
- EI²OS, μ DMAC は使用できません。

■ タイムベースタイマの割込みと EI²OS, μ DMAC

表 9.4-1 に、タイムベースタイマの割込みと EI²OS, μ DMAC を示します。

表 9.4-1 タイムベースタイマの割込みと EI²OS, μ DMAC

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	μ DMAC
	レジスタ名	アドレス	下位	上位	バンク		
#40	ICR14	0000BE _H	FFFF5C _H	FFFF5D _H	FFFF5E _H	×	×

× : 使用不可

< 注意事項 >

ICR14 は、タイムベースタイマ割込みと時計タイマ割込みと UART 受信完了 ch.0/ch.1 割込みの 3 つの用途に使用できますが、割込みレベルは同一になります。

9.5 タイムベースタイマの動作

タイムベースタイマには、インターバルタイマ機能および一部の周辺機能へのクロック供給機能があります。

■ インターバルタイマ機能 (タイムベースタイマ) の動作

インターバルタイマ機能は、任意のインターバル時間ごとに割り込み要求を発生します。インターバルタイマとして動作させるには、図 9.5-1 の設定が必要です。

図 9.5-1 タイムベースタイマの設定

アドレス		bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit0
0000A9 _H	TBTC	予約	-	-	TBIE	TBOF	TBR	TBC1	TBC0	(WDTC)	
		1	-	-		0	0				

: 使用ビット
 - : 未使用ビット
 0 : 0 を設定
 1 : 1 を設定

- ・ タイムベースタイマカウンタは、クロックが発振している限り、内部カウンタクロック（発振クロックの2分周）に同期して、カウントアップを続けます。
- ・ カウンタがクリアされると "0" からカウントアップを行い、インターバルタイマ用のビットがオーバーフローすると、割り込み要求フラグビット (TBOF) を "1" にセットします。この場合、割り込み要求出力を許可 (TBIE=1) しているとクリアされた時間を基準にして、選択されたインターバル時間ごとに割り込みを発生することになります。
- ・ インターバル時間は、タイムベースタイマのクリア動作で、設定時間よりも長くなることがあります。

■ 発振安定待ち時間用タイマ機能

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用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し、発振安定待ち時間用のビットがオーバーフローするまでの時間となります。ただし、タイムベースタイマモードから PLL クロックモードまたはメインクロックモード復帰時には、タイムベースタイマカウンタがクリアされないため、カウント途中からの時間になります。表 9.5-1 に、タイムベースカウンタのクリアと発振安定待ち時間について示します。

表 9.5-1 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 クリア	TBOF	発振安定待ち時間
タイムベースタイマ制御レジスタ (TBTC)のタイムベースタイマ初期化ビット (TBR) へ "0" 書込み			なし
パワーオンリセット			メインクロック発振安定待ち時間
ウォッチドッグリセット	×		
メインストップモードの解除			
PLL ストップモードの解除			
サブストップモードの解除	×	×	サブクロック発振安定待ち時間
メイン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
サブクロックモードから メインクロックモードへの移行 (MCS=1 1)			メイン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	×	×	なし
スリープモードの解除	×	×	なし

○ : あり
× : なし

■ クロック供給機能

タイムベースタイマは、ウォッチドッグタイマにクロックを供給しています。タイムベースカウンタのクリアで、ウォッチドッグタイマは、動作に影響を受けます。

9.6 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のクリアによる周辺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 (TBTC) の割込み要求フラグビット (TBOF) をクリアする場合には、割込み要求許可ビット (TBIE) 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) でタイムベースタイマ割込みがマスクされている状態で行うようにしてください。

● 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のカウンタのクリアで、次の動作は影響を受けます。

- ・ タイムベースタイマでインターバルタイマ機能（インターバル割込み）を使用している場合
- ・ ウォッチドッグタイマを使用している場合

● 発振安定待ち時間用タイマとしての使用

電源投入時、メインストップモード中などでは、発振クロックが停止しているため、発振器が動作を開始した後、タイムベースタイマが供給する動作クロックを使用して発振クロックの発振安定待ち時間をとります。高速発振端子に接続する振動子の種類によって、適切な発振安定待ち時間を選択する必要があります。詳細については、「5.5 発振安定待ち時間」を参照してください。

● タイムベースタイマからクロックを供給される周辺機能に対する注意

メインクロックが停止するモードでは、カウンタはクリアされ、タイムベースタイマは動作を停止します。また、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カウンタがクリアされると、初期状態からの供給となるため、“H” レベルが短くまたは “L” レベルが最大で 1/2 周期長くなることがあります。ウォッチドッグタイマ用のクロックも初期状態からの供給となりますが、ウォッチドッグタイマのカウンタが同時にクリアされるためウォッチドッグタイマは正常な周期で動作します。

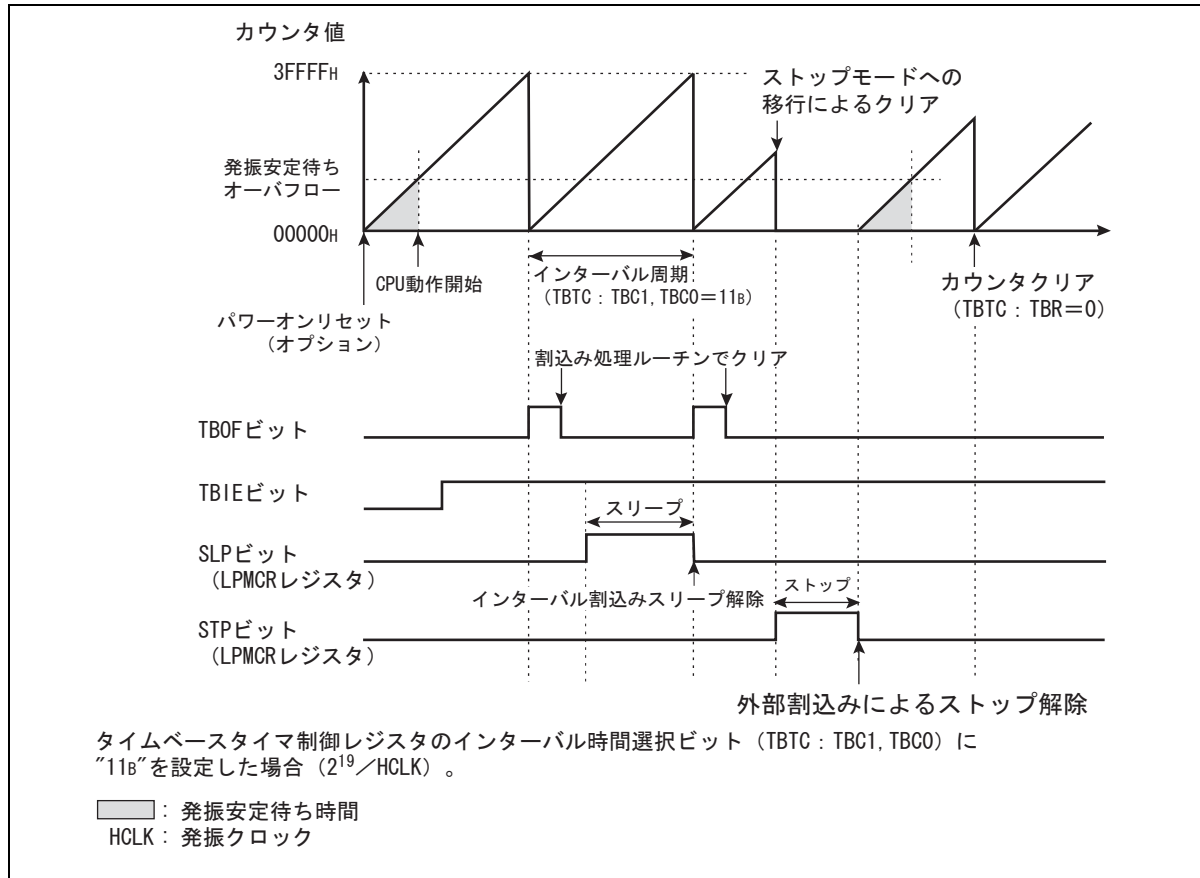
■ タイムベースタイマの動作

次の状態の動作を図 9.6-1 に示します。

- ・ パワーオンリセットが発生した場合
- ・ インターバルタイマ機能の動作中にスリープモードへ移行した場合
- ・ ストップモードへ移行した場合
- ・ カウンタクリアの要求があった場合

ストップモードへの移行でタイムベースタイマはクリアされ、動作を停止します。ストップモードからの復帰時にタイムベースタイマで発振安定待ち時間をカウントします。

図 9.6-1 タイムベースタイマの動作



9.7 タイムベースタイマのプログラム例

タイムベースタイマのプログラム例を示します。

■ タイムベースタイマのプログラム例

● 処理仕様

$2^{12}/\text{HCLK}$ (発振クロック) のインターバル割込みを繰り返し発生します。この場合のインターバル時間は、約 0.68 ms (6 MHz 動作時) となります。

● コーディング例

```

ICR14    EQU    0000BEH    ; タイムベースタイマ用割込み制御レジスタ
TBTC     EQU    0000A9H    ; タイムベースタイマ制御レジスタ
TBOF     EQU    TBTC:3     ; 割込み要求フラグビット
;----- メインプログラム -----
CODE     CSEG
START:
;           ;               ; スタックポインタ (SP) などは初期化済み
;           ;               ; とする

AND       CCR,      #0BFH   ; 割込み禁止
MOV       I:ICR14,  #00H    ; 割込みレベル 0 (最も高い)
MOV       I:TBTC,   #10010000B ; 上位 3 ビットは固定
;           ;               ; 割込み許可, TBOF クリア
;           ;               ; カウンタクリア
;           ;               ; インターバル時間  $2^{12}/\text{HCLK}$  選択

MOV       ILM,      #07H    ; PS 内 ILM をレベル 7 に設定
OR        CCR,      #40H    ; 割込み許可
LOOP:     MOV       A, #00H   ; 無限ループ
MOV       A, #01H
BRA       LOOP
;----- 割込みプログラム -----
WARI:
;           CLR ビット BOF    ; 割込み要求フラグをクリア
;           ;
;           ユーザ処理
;           ;
;           RETI              ; 割込みから復帰
CODE     ENDS
;----- ベクタ設定 -----

```

```
      VECT      CSEG      ABS=0FFH
                        ORG      0FF6CH      ; 割込みベクタを設定
                        DSL      WARI
                        ORG      0FFDCH      ; リセットベクタ設定
                        DSL      START
                        DB        00H        ; シングルチップモードに設定
VECT      ENDS
      END        START
```

第10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。

10.1 ウォッチドッグタイマの概要

10.2 ウォッチドッグタイマ制御レジスタ (WDTC)

10.3 ウォッチドッグタイマの構成

10.4 ウォッチドッグタイマの動作

10.5 ウォッチドッグタイマ使用上の注意

10.6 ウォッチドッグタイマのプログラム例

10.1 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または時計タイマの出力をカウントクロックとする 2 ビットのカウンタで、起動後、一定時間内にクリアされない場合 CPU をリセットします。

■ ウォッチドッグタイマ機能

ウォッチドッグタイマは、プログラム暴走対策用のカウンタです。一度起動すると一定時間内で定期的にクリアし続ける必要があります。プログラムが無限ループに陥るなどして、一定時間以上クリアされない場合、CPU に対してウォッチドッグリセットが発生します。ウォッチドッグタイマのインターバル時間は、表 10.1-1 に示すように、ウォッチドッグタイマ制御レジスタ (WDTC) の WT1, WT0 ビットで設定できます。ウォッチドッグタイマがクリアされない場合、最小時間～最大時間の間にウォッチドッグリセットが発生します。この表の最小時間内にカウンタをクリアしてください。

表 10.1-1 ウォッチドッグタイマのインターバル時間

WT1	WT0	WDCS & SCM	インターバル時間		クロックサイクル数
			最小 *	最大 *	
0	0	1	約 2.39 ms	約 3.07 ms	$(2^{14} \pm 2^{11}) / \text{HCLK サイクル}$
0	1	1	約 9.56 ms	約 12.29 ms	$(2^{16} \pm 2^{13}) / \text{HCLK サイクル}$
1	0	1	約 38.23 ms	約 49.15 ms	$(2^{18} \pm 2^{15}) / \text{HCLK サイクル}$
1	1	1	約 305.83 ms	約 393.22 ms	$(2^{21} \pm 2^{18}) / \text{HCLK サイクル}$
0	0	0	約 0.448 s	約 0.576 s	$(2^{12} \pm 2^9) / \text{SCLK サイクル}$
0	1	0	約 3.584 s	約 4.608 s	$(2^{15} \pm 2^{12}) / \text{SCLK サイクル}$
1	0	0	約 7.168 s	約 9.216 s	$(2^{16} \pm 2^{13}) / \text{SCLK サイクル}$
1	1	0	約 14.336 s	約 18.432 s	$(2^{17} \pm 2^{14}) / \text{SCLK サイクル}$

*: 発振クロック (HCLK) 6 MHz, サブクロック (SCLK) 32 kHz の 4 分周 (=8 kHz) 動作時の値です。ウォッチドッグタイマインターバル時間の最大, 最小および発振クロックサイクル数は、クリアのタイミングで決まります。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の供給クロック) 周期の 3.5 ~ 4.5 倍になります。ウォッチドッグタイマインターバル時間については、「10.4 ウォッチドッグタイマの動作」を参照してください。

< 注意事項 >

ウォッチドッグカウンタは、タイムベースタイマの桁上り信号をカウントする 2 ビットカウンタによって構成されています。したがって、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が、設定した時間より長くなることがあります。

< 参考 >

ウォッチドッグタイマは起動されると、パワーオン、ウォッチドッグによるリセットで初期化され、停止状態になります。また、外部端子によるリセット、ソフトウェアリセット、ウォッチドッグタイマ制御レジスタのウォッチドッグ制御ビット (WTE) への書込み、スリープモードへの遷移およびストップモードへの遷移、時計モードへの遷移でウォッチドッグカウンタはクリアされますが、ウォッチドッグタイマは起動したままです。

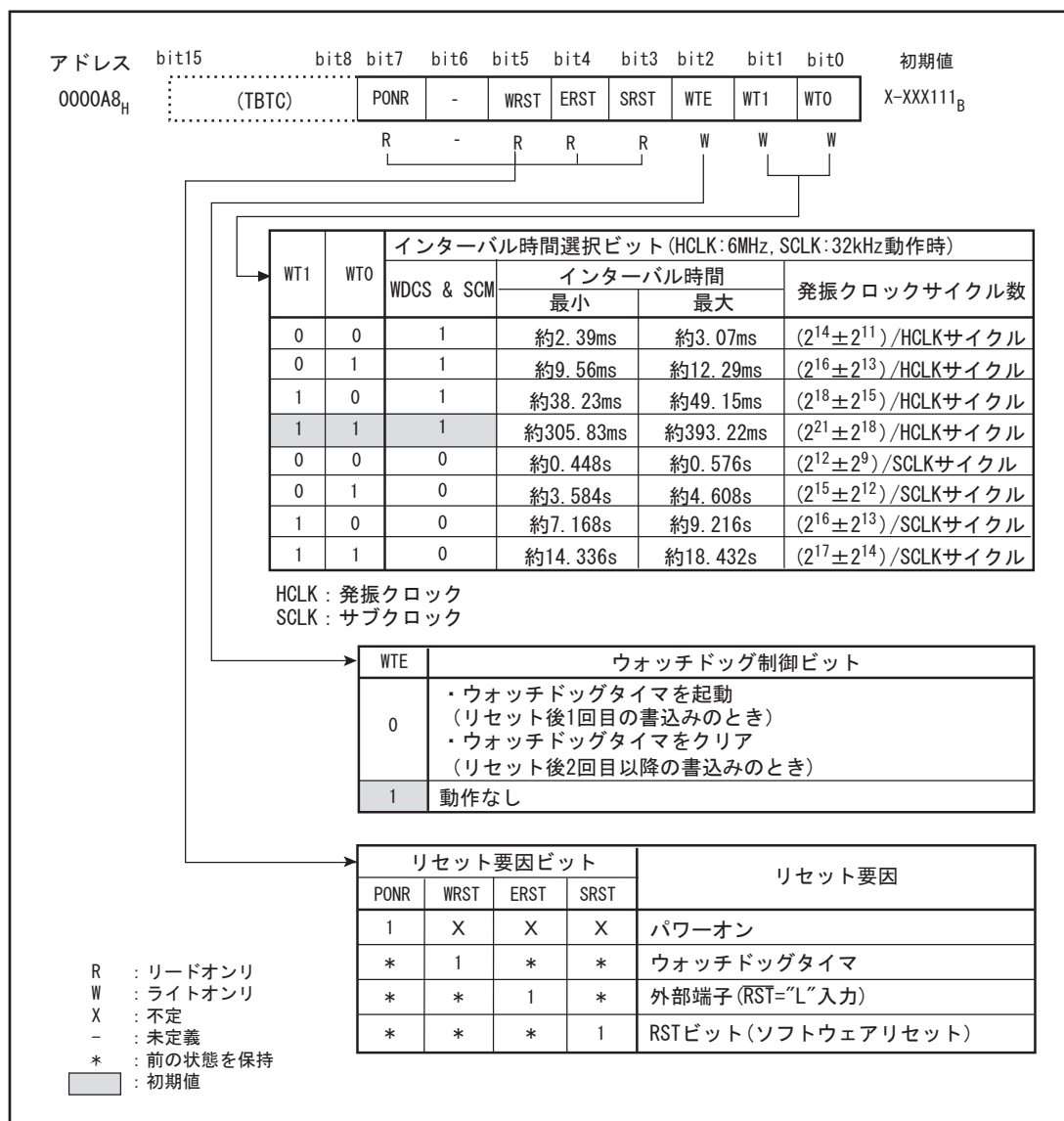
10.2 ウォッチドッグタイマ制御レジスタ (WDTC)

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うレジスタです。

■ ウォッチドッグタイマ制御レジスタ (WDTC)

図 10.2-1 にウォッチドッグタイマ制御レジスタ (WDTC) を、表 10.2-1 に WDTC レジスタの各ビットの機能について説明します。

図 10.2-1 ウォッチドッグタイマ制御レジスタ (WDTC) のビット構成



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) 周期の 3.5 ~ 4.5 倍となります。詳細は「10.4 ウォッチドッグタイマの動作」を参照してください。

表 10.2-1 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

ビット名			機能
bit7 , bit5 ~ 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読出し専用ビットです。各リセット要因が発生すると、これらのビットが"1"にセットされます。 ・PONR, WRST, ERST, SRST ビットは、WDTC レジスタの読出し動作後に、すべて "0" にクリアされます。 ・パワーオン時は、PONR ビット以外のビットの内容は保証されません。したがって、PONR ビットが "1" の場合はこれ以外のビットの内容を無視するようにしてください。
bit6	予約	予約	<ul style="list-style-type: none"> ・読出し値は不定です。 ・書込みは、動作に影響ありません。
bit2	WTE	ウォッチドッグ制御ビット	<ul style="list-style-type: none"> ・"0" を書き込むと、ウォッチドッグタイマを起動（リセット後 1 回目の書込み）または 2 ビットカウンタをクリア（リセット後 2 回目以降の書込み）します。 ・"1" の書込みでは、動作に影響はありません。
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するビットです。 ・クロックモードとしてサブクロックモードが選択されている（クロック選択レジスタ (CKSCR) のサブクロック表示ビット (SCM) が "0"）または時計タイマ制御レジスタ (WTC) により、ウォッチドッグタイマのクロックソースが時計タイマになっている場合（ウォッチドッグタイマクロックソース選択ビット (WDCS) が "0"）と、クロックモードとしてメインクロックモードまたは PLL クロックモードが選択されていて、かつ WTC の WDCS ビットが "1" の場合で、インターバル時間は図 10.2-1 のように異なります。 ・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時後の書込みデータは、無視されます。 ・WT1, WT0 ビットは、書込み専用です。

10.3 ウォッチドッグタイマの構成

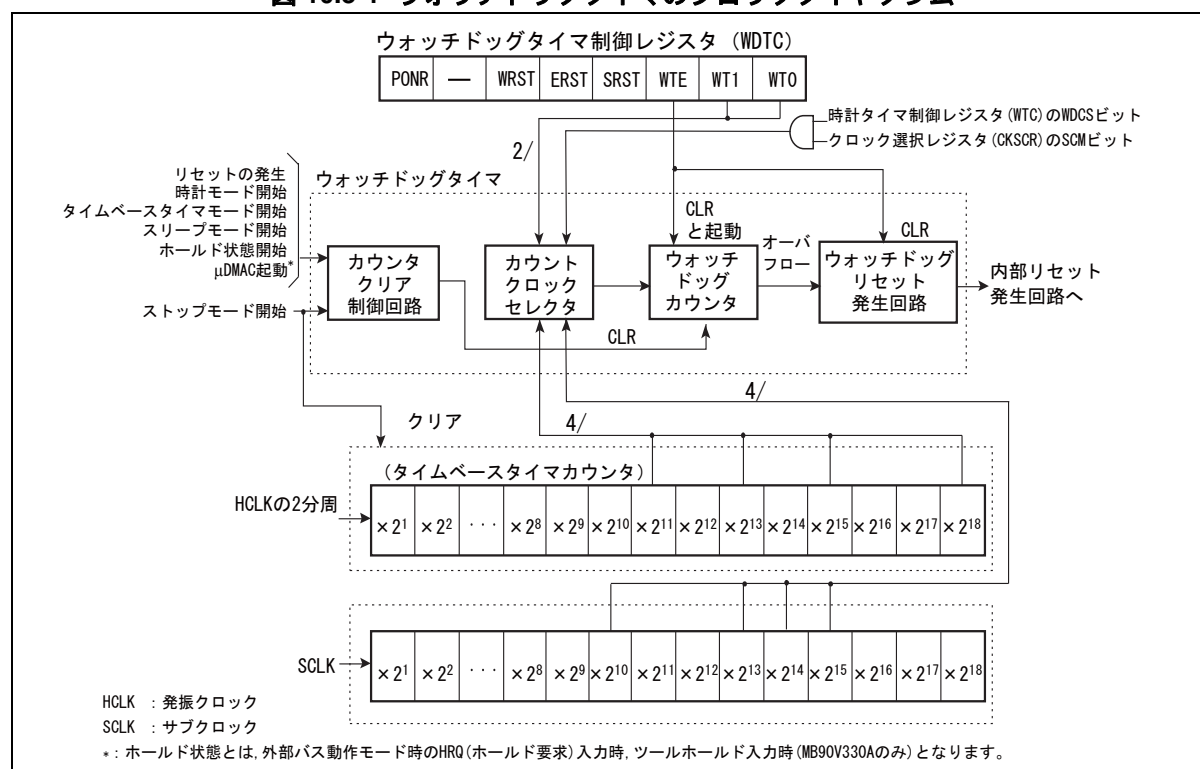
ウォッチドッグタイマは、次の 5 つのブロックで構成されています。

- 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 10.3-1 に、ウォッチドッグタイマのブロックダイアグラムを示します。

図 10.3-1 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

ウォッチドッグタイマのカウントクロックを、4 種類のタイムベースタイマ出力と 4 種類の時計タイマ出力から選択する回路です。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。

● ウォッチドッグカウンタ (2 ビットカウンタ)

タイムベースタイマ出力をカウントクロックとする 2 ビットのアップカウンタです。

● 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。

● カウンタクリア制御回路

ウォッチドッグカウンタのクリアと、カウンタの動作 / 停止を制御します。

表 10.3-1 ウォッチドッグタイマクリア条件

動作 モード	リセット	WDTC レジスタ WTE=0	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	ホールド	μDMAC
クリア	遷移時	書き込み時	遷移時	遷移時	遷移時	遷移時	遷移時	起動時
モード中の ウォッチ ドッグ タイマ状態	無効	-	停止 (クリア 状態を 保持)	動作 (クリア 直後から カウント 開始)	動作 (クリア 直後から カウント 開始)	動作 (クリア 直後から カウント 開始)	停止 (クリア状 態を保持)	停止 (クリア状 態を保持)
モード中の ウォッチ ドッグ リセット	発生 しない	-	発生 しない	発生 する	発生 する	発生 する	発生 しない	発生 しない
モード解除 / 復帰後の ウォッチ ドッグ タイマ状態	無効	動作	動作 (クリア 状態から カウント 再開)	動作 (カウント を継続)	動作 (カウント を継続)	動作 (カウント を継続)	動作 (クリア 状態から カウント 再開)	動作 (クリア 状態から カウント 再開)

● 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およびクリアと、リセット発生要因の保持を行います。

10.4 ウォッチドッグタイマの動作

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で、ウォッチドッグリセットが発生します。

■ ウォッチドッグタイマの動作

ウォッチドッグタイマの動作には、図 10.4-1 に示すような設定が必要です。

図 10.4-1 ウォッチドッグタイマの設定

アドレス	bit15 ... bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0000A8 _H WDTC	(TBT̄C̄)	PONR	-	WRST	ERST	SRST	WTE	WT1	WT0
							0		

：使用ビット
- ：未使用ビット
0 ："0" を設定

● ウォッチドッグタイマの起動

- ウォッチドッグタイマ制御レジスタ(WDTC)のウォッチドッグ制御ビット(WTE)に、リセット後1回目の"0"を書き込むとウォッチドッグタイマは起動します。この場合、WDTC レジスタのインターバル時間選択ビット (WT1, WT0) でインターバル時間を同時に指定します。
- ウォッチドッグタイマは起動を開始すると、パワーオン、ウォッチドッグによるリセット以外に止める方法はありません。

● ウォッチドッグタイマのクリア

- WTE ビットへの、2 回目以降の "0" の書込みで、ウォッチドッグタイマの 2 ビットカウンタをクリアします。インターバル時間内にカウンタがクリアされない場合、カウンタがオーバフローし、ウォッチドッグリセットが発生します。
- リセットの発生、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への遷移によって、ウォッチドッグカウンタはクリアされます。
- タイムベースタイマモードおよび時計モードへ遷移したときに、一度ウォッチドッグカウンタをクリアしますが、クリア後ウォッチドッグカウンタは停止しませんので注意してください。
- 時計モード使用時(サブクロック)は、ウォッチドッグタイマは使用しないようにしてください。

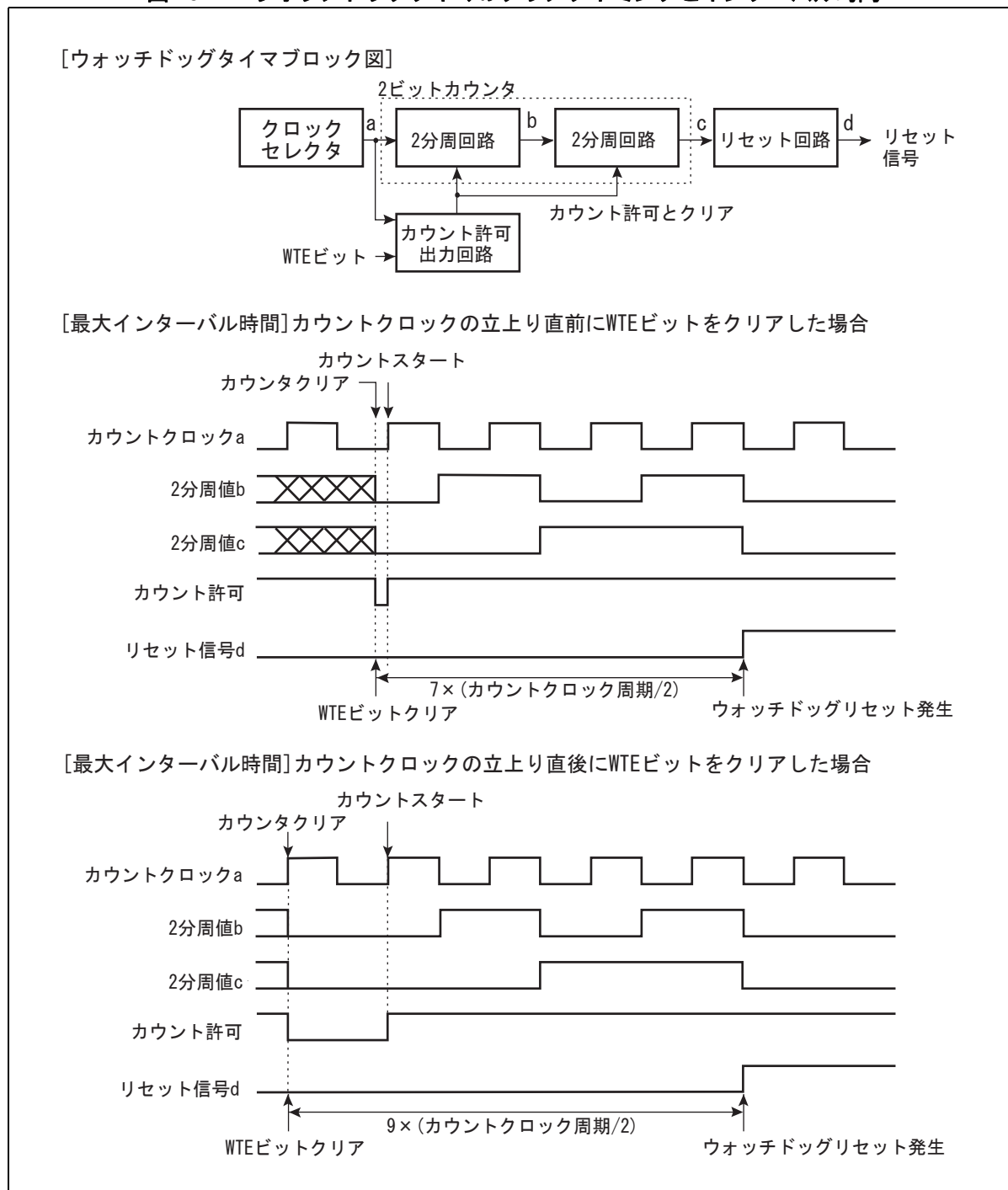
● ウォッチドッグタイマのインターバル時間

図 10.4-2 に、ウォッチドッグタイマのクリアのタイミングとインターバル時間の関係を示します。インターバル時間は、ウォッチドッグタイマをクリアするタイミングによって変化し、カウントクロック周期の 3.5 ~ 4.5 倍の時間を要します。

● リセット要因の確認

リセット後 WDTC レジスタのリセット要因ビット (PONR, WRST, ERST, SRST) をチェックすることで、リセット要因を知ることができます。

図 10.4-2 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



10.5 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使用時の注意点を示します。

■ 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

ウォッチドッグタイマは一度起動すると、パワーオン、ウォッチドッグ外部リセットが発生するまで停止できません。

- インターバル時間

インターバル時間は、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しているため、タイムベースタイマのクリアによって、ウォッチドッグタイマのインターバル時間が設定より長くなることがあります。

- インターバル時間の選択

インターバル時間は、ウォッチドッグタイマの起動時に設定が可能です。起動時以外の書込みデータは無視されます。

- プログラム作成上の注意

メインループの中で、繰り返しウォッチドッグタイマをクリアするようなプログラムを作成する場合、割込み処理を含めたメインループの処理時間が、ウォッチドッグタイマのインターバル時間の最小時間以下となる必要があります。

10.6 ウォッチドッグタイマのプログラム例

ウォッチドッグタイマのプログラム例を示します。

■ ウォッチドッグタイマのプログラム例

● 処理仕様

- ・ メインプログラムのループの中で毎回ウォッチドッグタイマをクリアします。
- ・ メインループは、ウォッチドッグタイマの最小インターバル時間内に 1 周する必要があります。

● コーディング例

```

WDTC EQU 0000A8H ;ウォッチドッグタイマ制御レジスタ
WTE EQU WDTC:2 ;ウォッチドッグ制御ビット
;----- メインプログラム -----
CODE CSEG
START:
;      : ;スタックポインタ (SP) などは初期化済み
;      : とする

WDG_START:
MOV WDTC, #00000011B ;ウォッチドッグタイマの起動
; ;インターバル時間 $2^{21} \pm 2^{18}$ サイクルを選択
;----- メインループ -----
MAIN: CLRB I:WTE ;ウォッチドッグタイマのクリア
;      : 定期的に 2 ビットのクリア
;      ユーザ処理
;      :
JMP MAIN ;ウォッチドッグタイマのインターバル時間
; ;より短い時間でループする
CODE ENDS
;----- ベクタ設定 -----
VECT CSEG ABS=0FFH
ORG 0FFDCH ;リセットベクタ設定
DSL START
DB 00H ;シングルチップモードに設定
VECT ENDS
END START

```


第11章

時計タイマ

この章では、時計タイマの概要、レジスタの構成 / 機能および時計タイマの動作について説明します。

- 11.1 時計タイマの概要
- 11.2 時計タイマの構成
- 11.3 時計タイマ制御レジスタ (WTC)
- 11.4 時計タイマの動作

11.1 時計タイマの概要

時計タイマは、サブクロックを使用した 15 ビットのタイマです。インターバルな割込みを発生させることができます。また、設定により、ウォッチドッグタイマのクロックソースとしても使用できます。

■ 時計タイマの機能

時計タイマは 15 ビットのタイマと、インターバルな割込みを制御する回路から構成されています。

時計タイマは、クロック選択レジスタ (CKSCR) の PLL クロック選択ビット (MCS) およびサブクロック選択ビット (SCS) と関係なく、サブクロックを使用します。

時計タイマのインターバル時間は表 11.1-1 に示すとおりです。

表 11.1-1 時計タイマのインターバル時間

WTC2	WTC1	WTC0	インターバル時間 *
0	0	0	31.25 ms
0	0	1	62.5 ms
0	1	0	125 ms
0	1	1	250 ms
1	0	0	500 ms
1	0	1	1.000 s
1	1	0	2.000 s
1	1	1	設定禁止

*: サブクロック 32 kHz の 4 分周 (=8 kHz)

11.2 時計タイマの構成

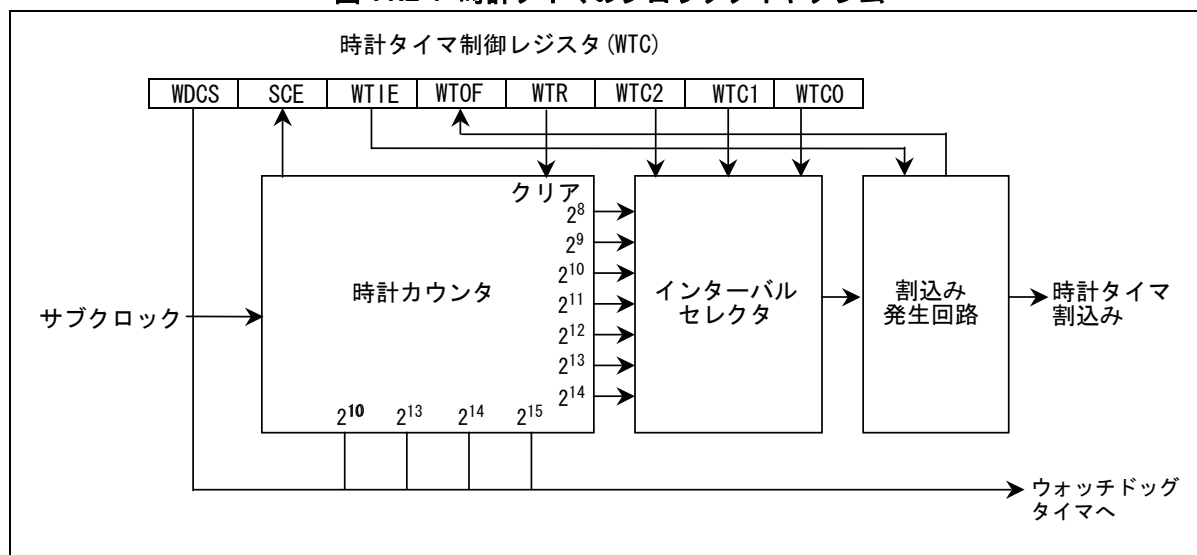
時計タイマは、次の 4 つのブロックから構成されています。

- インターバルセクタ
- 時計カウンタ
- 時計タイマ割込み発生回路
- 時計タイマ制御レジスタ (WTC)

■ 時計タイマのブロックダイアグラム

図 11.2-1 に、時計タイマのブロックダイアグラムを示します。

図 11.2-1 時計タイマのブロックダイアグラム



● 時計カウンタ

サブクロックをクロックソースとする 15 ビットのアップカウンタです。

● インターバルセクタ

時計タイマ割込みのインターバルを 7 種類から選択するセクタです。

● 割込み発生回路

時計タイマのインターバル割込みを発生させます。

● 時計タイマ制御レジスタ (WTC)

時計タイマの動作、時計タイマ割込みの制御およびウォッチドッグタイマのクロックソースを指定します。

11.3 時計タイマ制御レジスタ (WTC)

時計タイマ制御レジスタ (WTC) は、時計タイマの動作を制御します。また、インターバル割込みの時間を制御します。

■ 時計タイマ制御レジスタ (WTC) の構成

図 11.3-1 に時計タイマ制御レジスタ (WTC) の構成を、表 11.3-1 に時計タイマ制御レジスタ (WTC) の各ビットの機能について説明します。

図 11.3-1 時計タイマ制御レジスタ (WTC) のビット構成

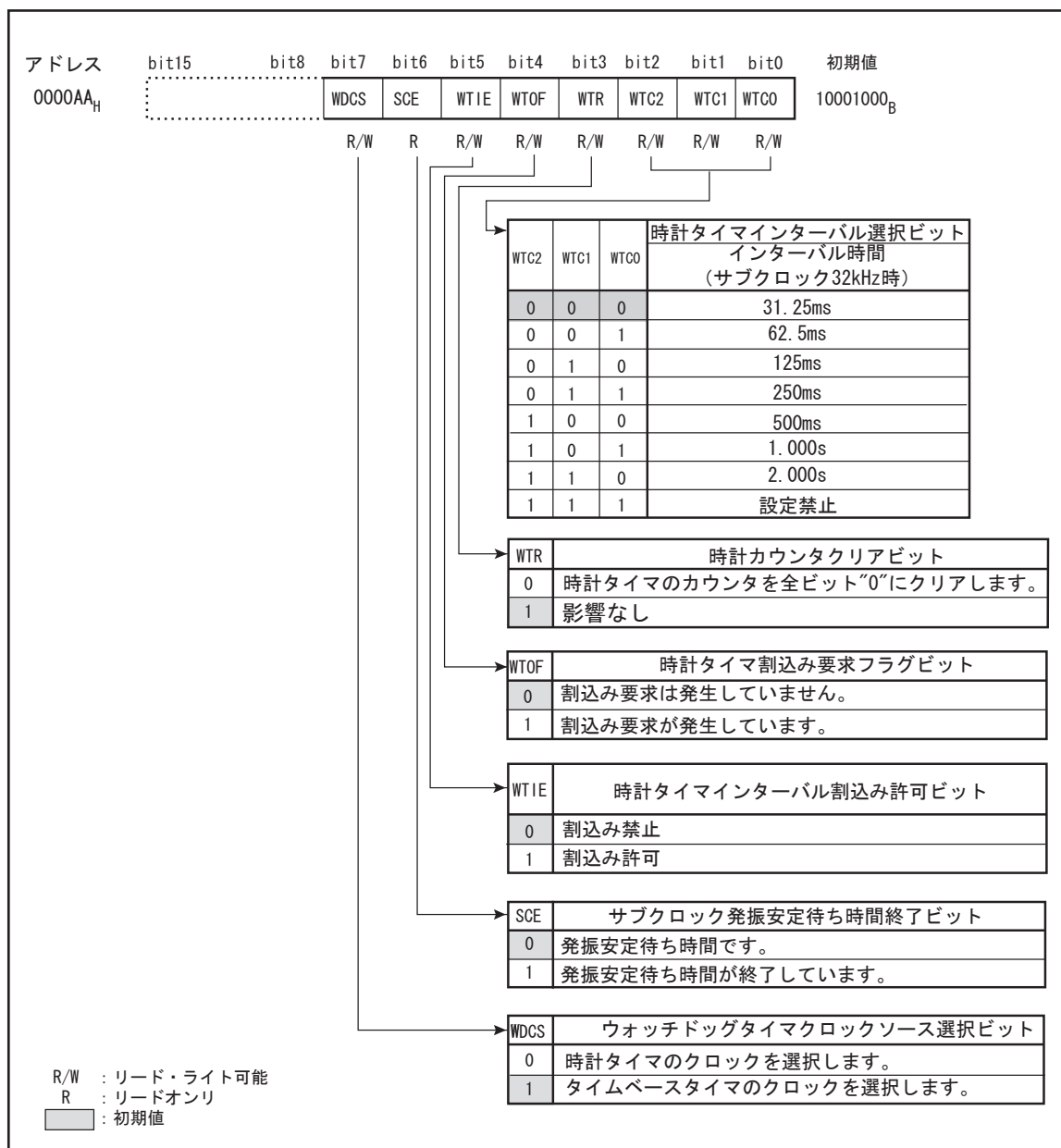


表 11.3-1 時計タイマ制御レジスタ (WTC) の各ビットの機能

ビット名		機能
bit7	WDCS: ウォッチドッグタイマ クロックソース 選択ビット	<ul style="list-style-type: none"> ・ウォッチドッグタイマのクロックソースを選択するビットです。 ・"0" のとき、時計タイマのクロックを選択し、"1" のとき、タイムベースタイマのクロックを選択します。 ・リセットにより "1" に初期化します。
bit6	SCE: サブクロック発振安定 待ち時間終了ビット	<ul style="list-style-type: none"> ・サブクロックの発振安定待ち時間が終了したことを示すビットです。 ・"0" のとき、発振安定待ち時間であることを示します。 ・サブクロックの発振安定待ち時間は、2^{14} サブクロックサイクル固定です。 ・パワーオンリセットおよびストップ時に "0" に初期化します。
bit5	WTIE: 時計タイマ インターバル 割込み許可ビット	<ul style="list-style-type: none"> ・時計タイマによるインターバル割込みを許可するビットです。 ・"1" のとき割込みを許可し、"0" のとき割込みを禁止します。 ・リセットにより "0" に初期化します。
bit4	WTOF: 時計タイマ割込み要求 フラグビット	<ul style="list-style-type: none"> ・時計タイマの割込み要求が発生していることを示すビットです。 ・WTIE ビットが "1" のときこのビットが "1" になると、割込み要求が発生します。 ・WTC2 ~ WTC0 ビットで設定されたインターバルごとに "1" にセットされます。 ・"0" の書込み、ストップモードへの遷移およびリセットにより "0" にクリアされます。 ・"1" 書込みは意味をもちません。
bit3	WTR: 時計カウンタ クリアビット	<ul style="list-style-type: none"> ・時計タイマのカウンタを全ビット "0" にクリアするビットです。 ・"0" を書き込むことにより、時計タイマカウンタを "0" にクリアします。 ・"1" 書込みは意味をもちません。 ・常に "1" が読み出されます。
bit2 ~ bit0	WTC2, WTC1, WTC0: 時計タイマ インターバル 選択ビット	<ul style="list-style-type: none"> ・時計タイマのインターバルを設定するビットです。 ・リセットにより "000_B" に初期化されます。 ・WTC2, WTC1, WTC0 ビットを変更するときは、WTOF ビットも同時にクリアしてください。

11.4 時計タイマの動作

時計タイマは、ウォッチドッグタイマのクロックソース、サブクロックの発振安定待ちのためのタイマ、一定周期で割込みを発生するインターバルタイマの機能があります。

■ 時計カウンタ

時計カウンタはサブクロックをカウントする 15 ビットのカウンタにより構成され、サブクロックが入力されている間、常にカウント動作を続けます。

● 時計カウンタのクリア

時計カウンタのクリアは、パワーオンリセット、ストップモードへの遷移および時計タイマ制御レジスタ (WTC) の時計カウンタクリアビット (WTR) への "0" 書込みにより行われます。

< 注意事項 >

- 時計タイマの出力を使用している、ウォッチドッグタイマとインターバル割込みは、時計カウンタのクリアにより動作に影響を与えます。
- 時計タイマ制御レジスタ (WTC) の WTR ビットへの "0" 書込みにより時計タイマをクリアする場合は、WTIE ビットに "0" を設定し、時計タイマの割込みを禁止した状態で行ってください。また、割込みを許可する前に、WTOF フラグへの "0" 書込みによる割込み要求のクリアを行ってください。

■ 時計タイマのインターバル割込み機能

時計カウンタの桁上り信号により一定周期で割込みを発生します。

● インターバル時間の指定

WTC レジスタの (WTC2, WTC1, WTC0) ビットにより、インターバル時間を指定できます。

● 時計タイマ割込みの発生

WTC2, WTC1, WTC0 ビットで設定されるインターバル時間ごとに時計タイマ割込み要求フラグビット (WTOF) をセットします。このとき、時計タイマインターバル割込み許可ビット (WTIE) が "1" に設定されていて、割込みが許可されていれば、時計タイマ割込みを発生します。

WTOF ビットのセットは、最後に時計タイマがクリアされた時間を基準にして行われます。

ストップモードに遷移すると、時計タイマは、サブクロックの発振安定時間待ちのタイマとして使用されるため、WTOF ビットはモード遷移と同時にクリアされます。

■ ウォッチドッグタイマのクロックソースの指定機能

WTC レジスタのウォッチドッグタイマクロックソース選択ビット (WDCS) により、ウォッチドッグタイマのクロックソースを指定できます。ただし、クロックモードがサブクロックモードの場合は、WDCS ビットによらず、時計タイマのカウンタ値が使用されます。

■ サブクロックの発振安定時間待ち機能

パワーオンリセット、ストップモードからの復帰時は、時計タイマは、サブクロックの発振安定時間待ちタイマとして機能します。サブクロックの発振安定待ち時間はサブクロックの 2^{14} サイクル固定です。

第 12 章

16 ビット入出力タイマ

この章では、16 ビット入出力タイマの概要、レジスタの構成 / 機能および 16 ビット入出力タイマの動作について説明します。

12.1 16 ビット入出力タイマの概要

12.2 16 ビット入出力タイマのレジスタ

12.3 16 ビット入出力タイマの動作

12.1 16 ビット入出力タイマの概要

16 ビット入出力タイマは、16 ビットフリーランタイム 1 本、アウトプットコンペア 4 本、インプットキャプチャ 4 本のモジュールから構成されています。本機能を用いることにより、16 ビットフリーランタイムをベースに 4 本の独立した波形出力が可能であり、入力パルス幅測定、外部クロック周期の測定が可能となります。

■ 16 ビット入出力タイマの構成および機能

16 ビット入出力タイマを構成している 16 ビットフリーランタイム、アウトプットコンペアおよびインプットキャプチャの機能は、以下のとおりです。

● 16 ビットフリーランタイム (× 1)

16 ビットフリーランタイムは 16 ビットのアップカウンタ、コントロールレジスタ、プリスケアラから構成されています。タイマカウンタの出力値は、インプットキャプチャ、アウトプットコンペアの基本時間 (ベースタイマ) として使用されます。カウンタ動作のためのクロックは、8 種類から選択できます (ϕ : マシンクロック)。

- 内部クロック 8 種類 (ϕ , $\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$, $\phi/32$, $\phi/64$, $\phi/128$)
- 基本になるマシンクロックは、内部クロックと外部クロック (FRCK) のいずれかを選択できます。

割込みは、カウンタ値のオーバフロー、コンペアクリアレジスタとのコンペア一致により発生できます (コンペア一致はモード設定が必要です)。

カウント値は、リセット、ソフトウェアクリア、コンペアクリアレジスタとのコンペア一致により "0000_H" に初期化できます。

● アウトプットコンペア (× 4)

アウトプットコンペアは、4 本の 16 ビットコンペアレジスタ、コンペア出力用ラッチおよびコントロールレジスタによって構成されています。16 ビットフリーランタイム値とコンペアレジスタ値が一致したとき出力レベルを反転するとともに、割込みを発生できます。

- 4 本のコンペアレジスタを独立して動作させます。各コンペアレジスタに対応した出力端子と割込みフラグがあります。
- 2 本のコンペアレジスタをペアにして出力端子を制御できます。
- 出力端子の初期値を設定できます。
- 割込みは、コンペア一致により発生可能です。

● インพุットキャプチャ (× 4)

インพุットキャプチャは、独立した4本の外部入力端子と対応したキャプチャレジスタ、コンペアレジスタから構成されています。外部入力端子より入力された信号の任意エッジを検出することにより、16 ビットフリーランタイム値をキャプチャレジスタに保持し、同時に割込みを発生できます。

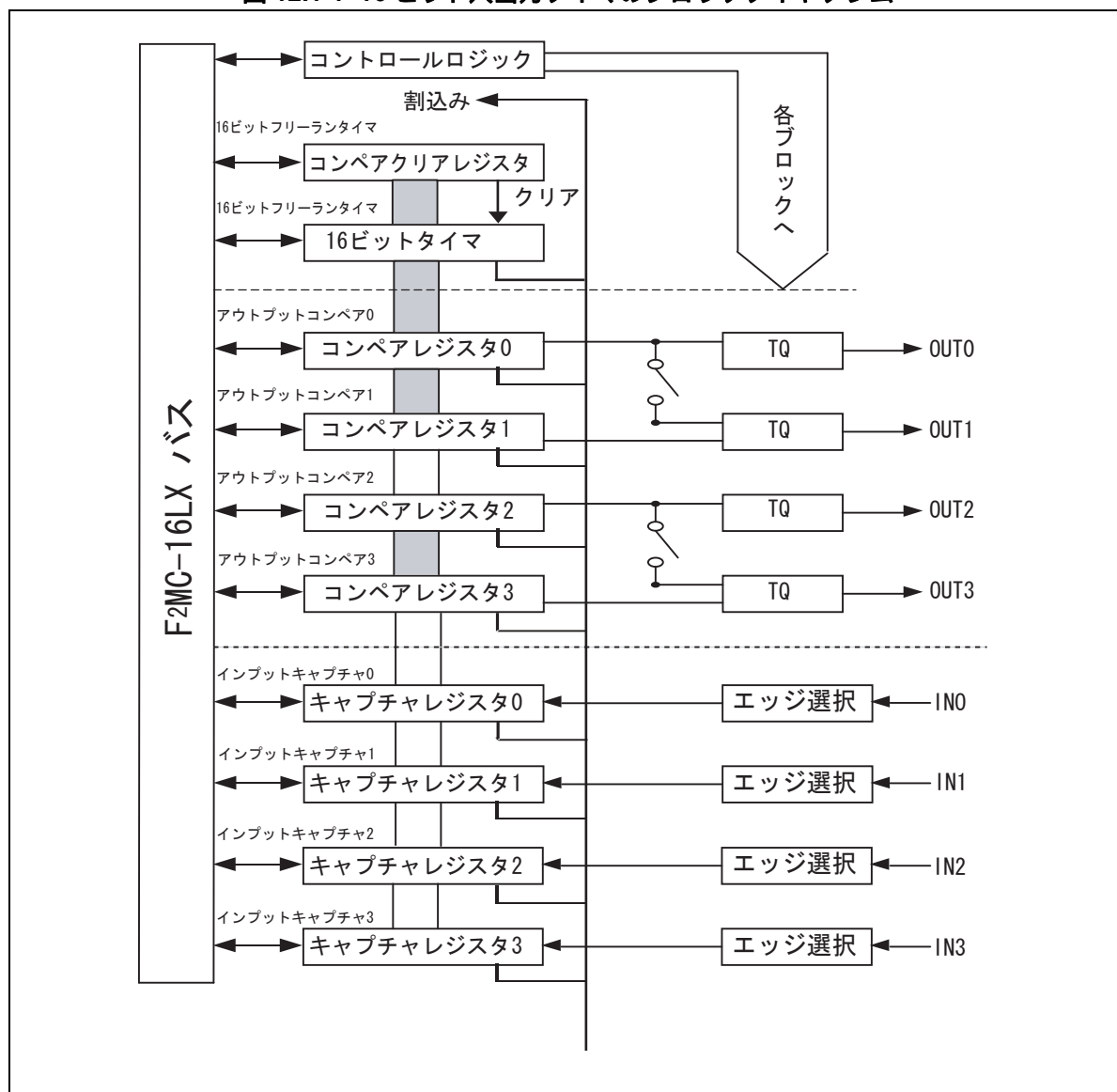
- ・ 外部入力信号のエッジを選択可能です。立上りエッジ、立下りエッジの両エッジから選択できます。
- ・ 4 本のインพุットキャプチャは独立して動作可能です。

割込みは、外部入力信号の有効エッジにより発生可能です。インพุットキャプチャは、割込みにより μ DMAC を起動できます。

■ ブロックダイアグラム

図 12.1-1 に、16 ビット入出力タイマのブロックダイアグラムを示します。

図 12.1-1 16 ビット入出力タイマのブロックダイアグラム



12.2 16 ビット入出力タイマのレジスタ

16 ビット入出力タイマのレジスタは、大きく以下のように分類されます。

- 16 ビットフリーランタイム
- 16 ビットアウトプットコンペア
- 16 ビットインプットキャプチャ

レジスタの構成および機能について説明します。

■ 16 ビット入出力タイマのレジスタ構成

以下に、16 ビット入出力タイマのレジスタ構成を示します。

● 16 ビットフリーランタイム

図 12.2-1 16 ビットフリーランタイムの構成

bit	15	0	
00008B _H , 00008A _H	CPCLR		コンペアクリアレジスタ
000087 _H , 000086 _H	TCDT		タイマカウンタデータレジスタ
000089 _H , 000088 _H	TCCS		タイマコントロールステータスレジスタ

● 16 ビットアウトプットコンペア

図 12.2-2 16 ビットアウトプットコンペアの構成

bit 15		0		
007919 _H , 007918 _H	OCCP0		アウトプットコンペアレジスタ	
00791B _H , 00791A _H	OCCP1			
00791D _H , 00791C _H	OCCP2			
00791F _H , 00791E _H	OCCP3			
000055 _H , 000054 _H	OCS1	OCS0	アウトプットコンペア コントロールレジスタ	
000057 _H , 000056 _H	OCS3	OCS2		

● 16 ビットインプットキャプチャ

図 12.2-3 16 ビットインプットキャプチャの構成

bit 15		0		
007911 _H , 007910 _H	IPCP0			インプットキャプチャデータレジスタ
007913 _H , 007912 _H	IPCP1			
007915 _H , 007914 _H	IPCP2			
007917 _H , 007916 _H	IPCP3			
000053 _H , 000052 _H		ICS23	ICS01	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

12.2.1 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、16 ビットアップダウンカウンタ、コントロールステータスレジスタから構成されています。

タイマのカウント値は、インプットキャプチャ、アウトプットコンペアの基本時間（ベースタイム）として使用されます。

- カウンタ動作のためのクロックは、8 種類から選択可能です。
- カウンタオーバフロー割込みを発生できます。
- モード設定により、アウトコンペアのコンペアクリアレジスタの値との一致によるカウンタの初期化が可能です。

■ 16 ビットフリーランタイムのレジスタ一覧

図 12.2-4 に、16 ビットフリーランタイムのレジスタ一覧を示します。

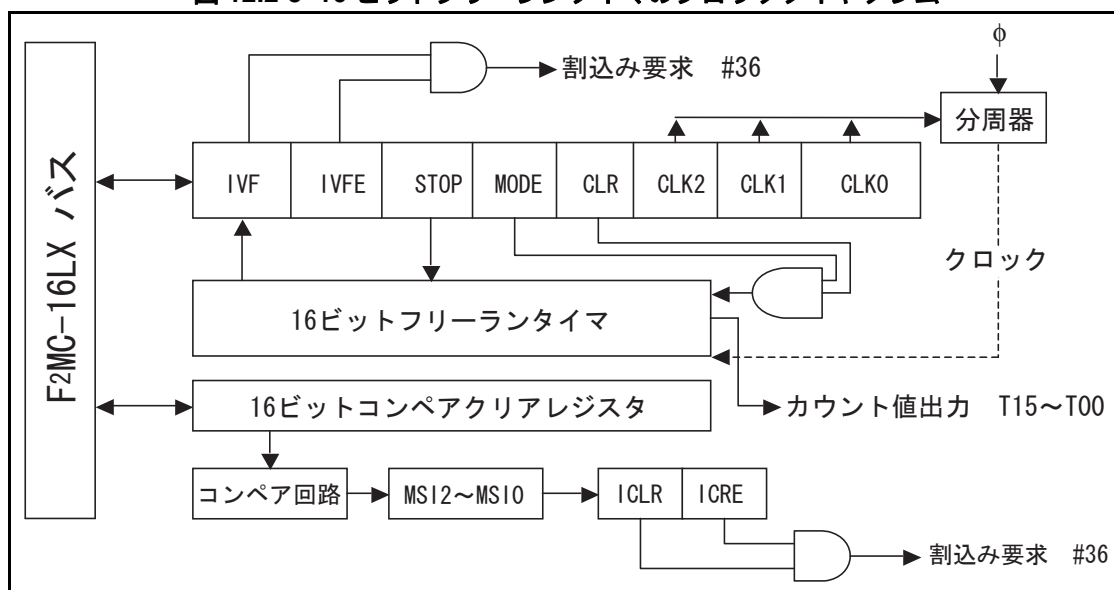
図 12.2-4 16 ビットフリーランタイムのレジスタ一覧

bit	15	14	13	12	11	10	9	8	CPCLR
00008B _H	CL15	CL14	CL13	CL12	CL11	CL10	CL09	CL08	コンペアクリアレジスタ上位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 XXXXXXXX _B
bit	7	6	5	4	3	2	1	0	CPCLR
00008A _H	CL07	CL06	CL05	CL04	CL03	CL02	CL01	CL00	コンペアクリアレジスタ下位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 XXXXXXXX _B
bit	15	14	13	12	11	10	9	8	TCDT
000087 _H	T15	T14	T13	T12	T11	T10	T09	T08	タイマカウンタデータレジスタ上位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B
bit	7	6	5	4	3	2	1	0	TCDT
000086 _H	T07	T06	T05	T04	T03	T02	T01	T00	タイマカウンタデータレジスタ下位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B
bit	15	14	13	12	11	10	9	8	TCCS
000089 _H	ECKE	-	-	MSI2	MSI1	MSI0	ICLR	ICRE	タイマコントロールステータスレジスタ上位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0XX00000 _B
bit	7	6	5	4	3	2	1	0	TCCS
000088 _H	IVF	IVFE	STOP	MODE	CLR	CLK2	CLK1	CLK0	タイマコントロールステータスレジスタ下位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B

R/W: リード・ライト可能
X: 不定

図 12.2-5 に、16 ビットフリーランタイムのブロックダイヤグラムを示します。

図 12.2-5 16 ビットフリーランタイムのブロックダイヤグラム



コンペアクリアレジスタ (CPCLR) のビット構成を、図 12.2-6 に示します。

図 12.2-6 コンペアクリアレジスタ (CPCLR) のビット構成

bit	15	14	13	12	11	10	9	8	CPCLR
00008B _H	CL15	CL14	CL13	CL12	CL11	CL10	CL09	CL08	コンペアクリアレジスタ上位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 XXXXXXXX _B

bit	7	6	5	4	3	2	1	0	CPCLR
00008A _H	CL07	CL06	CL05	CL04	CL03	CL02	CL01	CL00	コンペアクリアレジスタ下位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 XXXXXXXX _B

R/W: リード・ライト可能
X: 不定

コンペアクリアレジスタ (CPCLR) は、16 ビットフリーランタイムと比較する 16 ビット長のコンペアレジスタです。CPCLR レジスタ値は初期値不定ですので、動作許可する場合は値を設定してから動作を許可してください。また、CPCLR レジスタは、ワードアクセスしてください。

タイマコントロールステータスレジスタ (TCCS) の MODE ビットに "1" を設定している場合は、本レジスタ値と 16 ビットフリーランタイマ値が一致したとき、16 ビットフリーランタイマ値を "0000_H" にクリアします。また、レジスタ値と 16 ビットフリーランタイマ値が一致したとき、16 ビットフリーランタイマ値を "0000_H" に初期化してコンペアクリア割込みフラグをセットします。コンペアクリア割込みフラグが "1" のとき、割込み動作を許可している場合は、CPU に対して割込み要求を行います。

■ タイマカウンタデータレジスタ (TCDT)

タイマカウンタデータレジスタ (TCDT) のビット構成を、図 12.2-7 に示します。

図 12.2-7 タイマカウンタデータレジスタ (TCDT) のビット構成

bit	15	14	13	12	11	10	9	8	TCDT
000087 _H	T15	T14	T13	T12	T11	T10	T09	T08	タイマカウンタデータレジスタ 上位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B
bit	7	6	5	4	3	2	1	0	TCDT
000086 _H	T07	T06	T05	T04	T03	T02	T01	T00	タイマカウンタデータレジスタ 下位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B

R/W: リード・ライト可能

タイマカウンタデータレジスタ (TCDT) は、16 ビットフリーランタイマのカウンタ値を読み出すことができるレジスタです。カウンタ値はリセット時に "0000_H" にクリアされます。TCDT レジスタに書き込むことによって、タイマ値を設定できますが、必ず停止 (STOP=1) 状態で行ってください。

TCDT レジスタは、ワードアクセスしてください。16 ビットフリーランタイマの初期化は、次の要因で行われます。

- リセットによる初期化
- コントロールステータスのクリアビット (CLR) による初期化
- コンペアクリアレジスタとタイマカウンタ値の一致による初期化 (モード設定必要です)

■ タイマコントロールステータスレジスタ (TCCS)

タイマコントロールステータスレジスタ (TCCS) のビット構成を、図 12.2-8 に示します。

図 12.2-8 タイマコントロールステータスレジスタ (TCCS) のビット構成

bit	15	14	13	12	11	10	9	8	TCCS
000089 _H	ECKE	-	-	MSI2	MSI1	MSI0	ICLR	ICRE	タイマコントロールステータス レジスタ上位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 0XX00000 _B
bit	7	6	5	4	3	2	1	0	TCCS
000088 _H	IVF	IVFE	STOP	MODE	CLR	CLK2	CLK1	CLK0	タイマコントロールステータス レジスタ下位
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B

R/W: リード・ライト可能
X: 不定

以下に、タイマコントロールステータスレジスタ (TCCS) の各ビットの機能を説明します。

[bit15] ECKE (外部クロック入力許可ビット)

16 ビットフリーランタイムのカウントクロックソースが内部か外部を選択するビットです。ECKE ビットに書き込んだ後すぐにクロックは変更されますので、アウトプットコンペア、インプットキャプチャが停止状態のときに変更してください。

0	内部クロックソースを選択 [初期値]
1	外部端子 (FRCK) よりクロックを入力します。

[bit14, bit13] 未定義ビット

読出し値は不定です。書込みは動作に影響を与えません。

[bit12 ~ bit10] MSI2 ~ MSI0 (割込みマスク選択ビット)

コンペアクリア割込みをマスクする回数を設定するビットです。3 ビットのリロードカウンタで構成され、カウンタ値が "000_B" になるたびにカウント値をリロードします。また、MSI2, MSI1, MSI0 レジスタ書込み時にもカウント値をロードします。マスク回数 = 設定回数となります (例 : 2 回マスクし 3 回目で割込み処理をする場合 "010_B")。ただし、"000_B" を設定すると、割込み要因は、マスクされません。

[bit9] ICLR (コンペアクリア割込みフラグビット)

コンペアクリアの割込み要求フラグです。コンペアクリアレジスタ値と 16 ビットフリーランタイム値がコンペア一致したときにこのビットが "1" にセットされます。割込み要求許可ビット (bit8 の ICRE ビット) がセットされていると、割込みが発生します。ICLR ビットは、"0" を書き込むことによりクリアされます。"1" の書込みは意味を持ちません。リードモディファイライト (RMW) 系命令では、常に "1" が読み出されます。

0	割込み要求なし [初期値]
1	割込み要求あり

[bit8] ICRE (コンペアクリア割込み要求許可ビット)

コンペアクリアの割込み許可ビットです。"1" のとき、割込みフラグ (bit9 の ICLR) が "1" にセットされていると割込みが発生します。

0	割込み禁止 [初期値]
1	割込み許可

[bit7] IVF (タイマオーバーフロー発生フラグビット)

16 ビットフリーランタイムの割込み要求フラグです。

16 ビットフリーランタイムがオーバーフローを起こしたときまたはモード設定によりコンペアクリアレジスタと比較結果が一致してカウンタクリアされたとき、IVF ビットが "1" にセットされます。割込み要求許可ビット (bit5 の IVFE) がセットされていると、割込みが発生します。"0" 書込みによりクリアされます。"1" 書込みは意味を持ちません。リードモディファイライト (RMW) 系命令では、"1" が読み出されます。

0	割込み要求なし [初期値]
1	割込み要求あり

[bit6] IVFE (タイマオーバフロー割込み許可ビット)

16 ビットフリーランタイマの割込み許可ビットです。"1" のとき、書込みフラグ (bit5 の IVF) が "1" にセットされると割込みが発生します。

0	割込み禁止 [初期値]
1	割込み許可

[bit5] STOP (タイマ動作ストップビット)

16 ビットフリーランタイマのカウントを停止するためのビットです。"1" 書込み時にタイマのカウントを停止し、"0" 書込み時にタイマのカウントを開始します。

0	カウント許可 (動作) [初期値]
1	カウント禁止 (停止)

なお、16ビットフリーランタイマのカウントが停止すると、アウトプットコンペア動作も停止します。

[bit4] MODE (クリア条件選択ビット)

16 ビットフリーランタイマの初期化条件を設定します。

"0" のときは、リセットとクリアビット (bit2 の (CLR)) でカウント値を初期化できます。

"1" のときは、リセットとクリアビット (bit2 の (CLR)) のほかにコンペアクリアレジスタの値との一致により、カウンタ値を初期化できます。

0	リセット、クリアビットによる初期化 [初期値]
1	リセット、クリアビット、コンペアクリアレジスタによる初期化

なお、カウンタ値の初期化はカウント値の変化点で行われます。

[bit3] CLR (タイマクリアビット)

動作中の 16 ビットフリーランタイム値を "0000_H" に初期化するビットです。

"1" 書込み時にカウンタ値を "0000_H" に初期化します。"0" を書き込んでも意味を持ちません。読出し値は、常に "0" です。カウンタ値の初期化は、カウンタ値の変化点で行われます。

0	意味を持ちません。 [初期値]
1	カウンタ値を "0000 _H " に初期化します。

なお、タイマ停止中に初期化する場合は、データレジスタに "0000_H" を書き込んでください。

< 注意事項 >

"1"を書き込んだ後、次のカウントクロックまでに、このビットへの"0"書込みを行うと、カウンタ値の初期化は行われません。

[bit2, bit1, bit0] CLK2, CLK1, CLK0 (カウントクロック周期選択ビット)

16 ビットフリーランタイムのカウントクロックを選択します。CLK2, CLK1, CLK0 ビットに書込み後すぐにクロックは変更されますので、アウトプットコンペア、インプットキャプチャが停止状態のときに変更してください。

CLK2	CLK1	CLK0	カウントクロック	$\phi=24\text{ MHz}$	$\phi=12\text{ MHz}$	$\phi=6\text{ MHz}$	$\phi=3\text{ MHz}$
0	0	0	ϕ	41.7 ns	83.3 ns	0.17 μs	0.33 μs
0	0	1	$\phi/2$	83.3 ns	0.17 μs	0.33 μs	0.67 μs
0	1	0	$\phi/4$	0.17 μs	0.33 μs	0.67 μs	1.33 μs
0	1	1	$\phi/8$	0.33 μs	0.67 μs	1.33 μs	2.67 μs
1	0	0	$\phi/16$	0.67 μs	1.33 μs	2.67 μs	5.33 μs
1	0	1	$\phi/32$	1.33 μs	2.67 μs	5.33 μs	10.7 μs
1	1	0	$\phi/64$	2.67 μs	5.33 μs	10.7 μs	21.3 μs
1	1	1	$\phi/128$	5.33 μs	10.7 μs	21.3 μs	42.7 μs

12.2.2 アウトプットコンペア

アウトプットコンペアは、16 ビットのコンペアレジスタ、コンペア出力端子部、コントロールレジスタより構成されています。コンペアレジスタに設定された値と 16 ビットフリーランタイム値が一致すると、端子の出力レベルを反転するとともに割込みを発生できます。

- コンペアレジスタは全部で 4 本あり、独立して動作させることができます。
また、設定により 2 本のコンペアレジスタを使い出力を制御できます。
- 割込みは、コンペア一致により設定できます。

■ アウトプットコンペアのレジスタ一覧

図 12.2-9 に、アウトプットコンペアのレジスタ一覧を示します。

図 12.2-9 アウトプットコンペアのレジスタ一覧

bit	15	14	13	12	11	10	9	8	OCCP0 ~ OCCP3
ch.0 : 007919 _H	C15	C14	C13	C12	C11	C10	C09	C08	アウトプットコンペア レジスタ上位 初期値 XXXXXXXX _B
ch.1 : 00791B _H									
ch.2 : 00791D _H	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
ch.3 : 00791F _H									
bit	7	6	5	4	3	2	1	0	OCCP0 ~ OCCP3
ch.0 : 007918 _H	C07	C06	C05	C04	C03	C02	C01	C00	アウトプットコンペア レジスタ下位 初期値 XXXXXXXX _B
ch.1 : 00791A _H									
ch.2 : 00791C _H	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
ch.3 : 00791E _H									
bit	15	14	13	12	11	10	9	8	OCS1/OCS3
ch.1 : 000055 _H	-	-	-	CMOD	OTE1	OTE0	OTD1	OTD0	アウトプットコンペアコ ントロールレジスタ上位 初期値 ---00000 _B
ch.3 : 000057 _H									
	-	-	-	R/W	R/W	R/W	R/W	R/W	
bit	7	6	5	4	3	2	1	0	OCS0/OCS2
ch.0 : 000054 _H	ICP1	ICP0	ICE1	ICE0	-	-	CST1	CST0	アウトプットコンペアコ ントロールレジスタ下位 初期値 0000--00 _B
ch.2 : 000056 _H									
	R/W	R/W	R/W	R/W	-	-	R/W	R/W	

< 注意事項 >

コンペアレジスタを書き換える場合は、コンペア割込みのルーチン内で行うか、コンペア動作禁止の状態で行い、コンペア一致と書込みが同時に発生しないようにしてください。

■ アウトプットコンペアコントロールレジスタ (OCS0 ~ OCS3)

アウトプットコンペアコントロールレジスタ (OCS0 ~ OCS3) のビット構成を、図 12.2-12 に示します。

図 12.2-12 アウトプットコンペアコントロールレジスタ (OCS0 ~ OCS3) のビット構成

bit	15	14	13	12	11	10	9	8	OCS1/OCS3
ch.1 : 000055 _H ch.3 : 000057 _H	-	-	-	CMOD	OTE1	OTE0	OTD1	OTD0	アウトプットコンペアコントロールレジスタ上位
	-	-	-	R/W	R/W	R/W	R/W	R/W	初期値 ---00000 _B
bit	7	6	5	4	3	2	1	0	OCS0/OCS2
ch.0 : 000054 _H ch.2 : 000056 _H	ICP1	ICP0	ICE1	ICE0	-	-	CST1	CST0	アウトプットコンペアコントロールレジスタ下位
	R/W	R/W	R/W	R/W	-	-	R/W	R/W	初期値 0000--00 _B

以下に、アウトプットコンペアコントロールレジスタ (OCS0 ~ OCS3) の各ビットの機能を説明します。

[bit15, bit14, bit13] 未定義ビット

読出し値は不定です。書込みは動作に影響を与えません。

[bit12] CMOD (出力レベル反転モードビット)

端子出力を許可した場合 (OTE1=1 OR OTE0=1) のコンペア一致における端子出力レベル反転動作モードを切り換えます。

- CMOD=0 のとき (初期値) は、コンペアレジスタに対応したレベルを反転します。
 - OUT0/OUT2: コンペアレジスタ 0/2 の一致によりレベルを反転します。
 - OUT1/OUT3: コンペアレジスタ 1/3 の一致によりレベルを反転します。
- CMOD=1 のときは、コンペアレジスタ 0/2 は、CMOD=0 のときと同じく出力レベルを反転しますが、コンペアレジスタ 1/3 に対応した端子 (OUT1/OUT3) の出力レベルは、コンペアレジスタ 0/2 の一致とコンペアレジスタ 1/3 の一致の両方で出力レベルを反転します。コンペアレジスタ 0/2 と 1/3 が同値のときは、コンペアレジスタ 1 本のときと同じ動作をします。
 - OUT0/OUT2: コンペアレジスタ 0/2 の一致によりレベルを反転します。
 - OUT1/OUT3: コンペアレジスタ 1/3 または 0/2 の一致によりレベルを反転します。

[bit11, bit10] OTE1, OTE0 (出力許可ビット)

アウトプットコンペアの端子出力を許可するビットです。初期値は "0" です。

0	汎用ポートとして動作します。 (初期値)
1	アウトプットコンペア端子出力になります。

- OTE1: アウトプットコンペア 1/3 に対応
- OTE0: アウトプットコンペア 0/2 に対応

[bit9, bit8] OTD1, OTD0 (出力レベルビット)

アウトプットコンペアの端子出力を許可した場合の端子出力レベルを変更するとき
に使用します。コンペア端子出力の初期値は "0" となります。書込み時は、コン
ペア動作を停止してから行ってください。読出し時は、アウトプットコンペア端子
の出力値が読み出されます。

0	コンペア端子出力を "0" にします。 (初期値)
1	コンペア端子出力を "1" にします。

- OTD1: アウトプットコンペア 1/3 に対応
- OTD0: アウトプットコンペア 0/2 に対応

[bit7, bit6] ICP1, ICP0 (コンペアー一致割込みフラグビット)

アウトプットコンペアの割込みフラグです。コンペアレジスタと 16 ビットフリー
ランタイム値が一致した場合に "1" にセットされます。割込み要求ビット (ICE1,
ICE0) が許可されているとき ICP1, ICP0 ビットがセットされると、アウトプットコ
ンペア割込みが発生します。"0" 書込みによりクリアされ、"1" 書込みでは意味を持
ちません。リードモディファイ (RMW) 系では、"1" が読み出されます。

0	コンペアー一致なし (初期値)
1	コンペアー一致あり

- ICP1: アウトプットコンペア 1/3 に対応
- ICP0: アウトプットコンペア 0/2 に対応

[bit5, bit4] ICE1, ICE0 (コンペアー一致割込み許可ビット)

アウトプットコンペアの割込み許可ビットです。ICE1, ICE0 ビットが "1" のとき割
込みフラグ (ICP1, ICP0) がセットされると、アウトプットコンペア割込みが発生し
ます。

0	アウトプットコンペア割込み禁止 (初期値)
1	アウトプットコンペア割込み許可

- ICE1: アウトプットコンペア 1/3 に対応
- ICE0: アウトプットコンペア 0/2 に対応

[bit3, bit2] 未定義ビット

読出し値は不定です。書込みは動作に影響を与えません。

[bit1, bit0] CST1, CST0 (コンペア動作許可ビット)

16 ビットフリーランタイムとの一致動作を許可するビットです。

0	コンペア動作禁止 (初期値)
1	コンペア動作許可

- CST1: アウトプットコンペア 1/3 に対応
- CST0: アウトプットコンペア 0/2 に対応

コンペア動作を許可する前に、コンペアレジスタ値を設定してください。

< 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のクロックと同期させているため、16 ビットフリーランタイムを停止させるとコンペア動作も停止します。

12.2.3 インプットキャプチャ

本モジュールは、外部から入力された信号の立上りエッジ、立下りエッジまたは両エッジを検出して、そのときの 16 ビットフリーランタイム値をレジスタに保持する機能を持っています。また、エッジ検出時に割込みを発生できます。

■ インプットキャプチャ

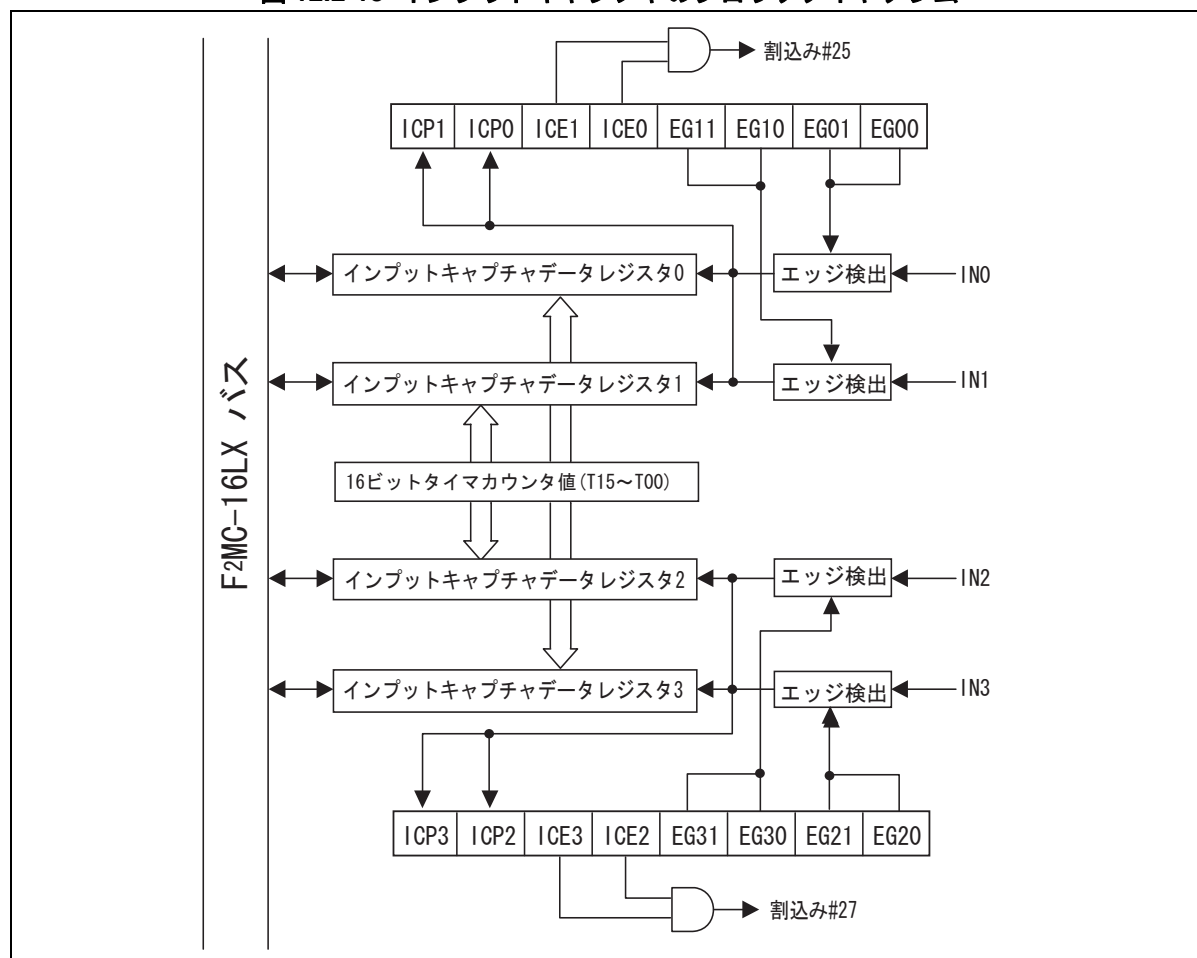
インプットキャプチャは、インプットキャプチャレジスタ、コントロールレジスタで構成されています。各インプットキャプチャはそれぞれ対応した外部入力端子をもっています。

- 外部入力の有効エッジを 3 種類から選択可能です。
立上りエッジ / 立下りエッジ / 両エッジ
- 外部入力の有効エッジを検出したときに割込みを発生できます。

■ インプットキャプチャのブロックダイアグラム

図 12.2-13 に、インプットキャプチャのブロックダイアグラムを示します。

図 12.2-13 インプットキャプチャのブロックダイアグラム



■ インพุットキャプチャのレジスタ一覧

図 12.2-14 に、インพุットキャプチャのレジスタ一覧を示します。

図 12.2-14 インพุットキャプチャのレジスタ一覧

bit	15	14	13	12	11	10	9	8	IPCP0 ~ IPCP3
ch.0 : 007911 _H	CP15	CP14	CP13	CP12	CP11	CP10	CP09	CP08	インพุットキャプチャ データレジスタ上位 初期値 XXXXXXXX _B
ch.1 : 007913 _H									
ch.2 : 007915 _H	R	R	R	R	R	R	R	R	
ch.3 : 007917 _H									
bit	7	6	5	4	3	2	1	0	IPCP0 ~ IPCP3
ch.0 : 007910 _H	CP07	CP06	CP05	CP04	CP03	CP02	CP01	CP00	インพุットキャプチャ データレジスタ下位 初期値 XXXXXXXX _B
ch.1 : 007912 _H									
ch.2 : 007914 _H	R	R	R	R	R	R	R	R	
ch.3 : 007916 _H									
bit	15	14	13	12	11	10	9	8	ICS23
000053 _H	ICP3	ICP2	ICE3	ICE2	EG31	EG30	EG21	EG20	インพุットキャプチャ コントロールステータス レジスタ 23 初期値 00000000 _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
bit	7	6	5	4	3	2	1	0	ICS01
000052 _H	ICP1	ICP0	ICE1	ICE0	EG11	EG10	EG01	EG00	インพุットキャプチャ コントロールステータス レジスタ 01 初期値 00000000 _B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

■ インพุットキャプチャデータレジスタ (IPCP0 ~ IPCP3)

インพุットキャプチャデータレジスタ (IPCP0 ~ IPCP3) のビット構成を、図 12.2-15 に示します。

図 12.2-15 インพุットキャプチャデータレジスタ (IPCP0 ~ IPCP3) のビット構成

bit	15	14	13	12	11	10	9	8	IPCP0 ~ IPCP3
ch.0 : 007911 _H	CP15	CP14	CP13	CP12	CP11	CP10	CP09	CP08	インพุットキャプチャ データレジスタ上位 初期値 XXXXXXXX _B
ch.1 : 007913 _H									
ch.2 : 007915 _H	R	R	R	R	R	R	R	R	
ch.3 : 007917 _H									
bit	7	6	5	4	3	2	1	0	IPCP0 ~ IPCP3
ch.0 : 007910 _H	CP07	CP06	CP05	CP04	CP03	CP02	CP01	CP00	インพุットキャプチャ データレジスタ下位 初期値 XXXXXXXX _B
ch.1 : 007912 _H									
ch.2 : 007914 _H	R	R	R	R	R	R	R	R	
ch.3 : 007916 _H									

インพุットキャプチャデータレジスタ (IPCP0 ~ IPCP3) は、対応した外部端子入力波形の有効エッジを検出したとき 16 ビットフリーランタイム値を保持するレジスタです。IPCP0 ~ IPCP3 レジスタは、ワードアクセスをしてください。IPCP0 ~ IPCP3 レジスタに書き込むことはできません。

■ インพุットキャプチャコントロールステータスレジスタ (ICS01, ICS23)

インพุットキャプチャコントロールステータスレジスタ (ICS01, ICS23) のビット構成を、図 12.2-16 に示します。

図 12.2-16 インพุットキャプチャコントロールステータスレジスタ (ICS01, ICS23) のビット構成

bit	15	14	13	12	11	10	9	8	ICS23
000053 _H	ICP3	ICP2	ICE3	ICE2	EG31	EG30	EG21	EG20	インพุットキャプチャ コントロールステータス レジスタ 23
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B
bit	7	6	5	4	3	2	1	0	ICS01
000052 _H	ICP1	ICP0	ICE1	ICE0	EG11	EG10	EG01	EG00	インพุットキャプチャ コントロールステータス レジスタ 01
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	初期値 00000000 _B

以下に、インพุットキャプチャコントロールステータスレジスタ (ICS01, ICS23) の各ビットの機能を説明します。

[bit15, bit14, bit7, bit6] ICP3, ICP2, ICP1, ICP0

(有効エッジ検出フラグビット)

インพุットキャプチャ割込みフラグです。外部入力端子の有効エッジを検出すると、ICP3 ~ ICP0 ビットを "1" にセットします。割込み許可ビット (ICE3, ICE2, ICE1, ICE0) がセットされていると、有効エッジを検出することにより割込みを発生させることができます。

"0" 書込みによりクリアされます。"1" 書込みは意味を持ちません。リードモディファイライト (RMW) 系命令では、"1" が読み出されます。

0	有効エッジ検出なし (初期値)
1	有効エッジ検出あり

ICPn: n の番号がインพุットキャプチャのチャンネル番号に対応します。

[bit13, bit12, bit5, bit4] ICE3, ICE2, ICE1, ICE0

(キャプチャ割込み許可ビット)

インพุットキャプチャ割込み許可ビットです。ICE3 ~ ICE0 ビットが "1" のとき割込みフラグ ICP3 ~ ICP0 がセットされると、インพุットキャプチャ割込みが発生します。

0	割込み禁止 (初期値)
1	割込み許可

ICE_n: n の番号がインพุットキャプチャのチャンネル番号に対応します。

[bit11 ~ bit8, bit3 ~ bit0]

EG31, EG30, EG21, EG20, EG11, EG10, EG01, EG00 (エッジ選択ビット)

外部入力の有効エッジ極性を指定します。インプットキャプチャ動作許可も兼用しています。

EGn1	EGn0	エッジ検出極性
0	0	エッジ検出なし (停止状態) (初期値)
0	1	立上りエッジ検出
1	0	立下りエッジ検出
1	1	両エッジ検出

EGn1/EGn0: n の番号がインプットキャプチャのチャンネル番号に対応します。

12.3 16 ビット入出力タイマの動作

16 ビット入出力タイマの動作およびタイミングについて説明します。

■ 16 ビット入出力タイマの動作，タイミング

16 ビット入出力タイマの動作，タイミングとして以下の項目について説明します。

- 16 ビットフリーランタイマの動作
- 16 ビットアウトプットコンペアの動作
- 16 ビットインプットキャプチャの動作
- 16 ビットフリーランタイマのタイミング
 - カウントタイミング
 - クリアタイミング
- アウトプットコンペアのタイミング
 - コンペア動作のタイミング
 - 割込みタイミング
 - 出力端子の変化タイミング
- インプットキャプチャの入力タイミング
入力信号に対するキャプチャタイミング

12.3.1 16 ビットフリーランタイマの動作

16 ビットフリーランタイマの動作およびタイミングについて説明します。

■ 16 ビットフリーランタイマの動作

16 ビットフリーランタイマは、リセット解除後にカウンタ値 "0000_H" よりカウントを開始します。このカウンタ値が、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なります。

カウンタ値は、次の条件でクリアされます。

- オーバフローが発生したとき
- コンペアクリアレジスタ値とコンペア一致したとき（モード設定必要）
- 動作中に TCCS レジスタの CLR ビットに "1" を書き込んだとき
- 動作中に TCDDT レジスタに "0000_H" を書き込んだとき
- リセット時

割込みはオーバフローが発生したとき、またはコンペアクリアレジスタ値と 16 ビットフリーランタイマのカウンタ値の比較結果が一致したとき発生します（比較結果一致割込みは、モード設定が必要です）。

図 12.3-1 にオーバフローによるカウンタクリアのタイミングチャートを、図 12.3-2 に比較結果一致によるカウンタクリアのタイミングチャートを示します。

図 12.3-1 オーバフローによるカウンタクリアのタイミングチャート

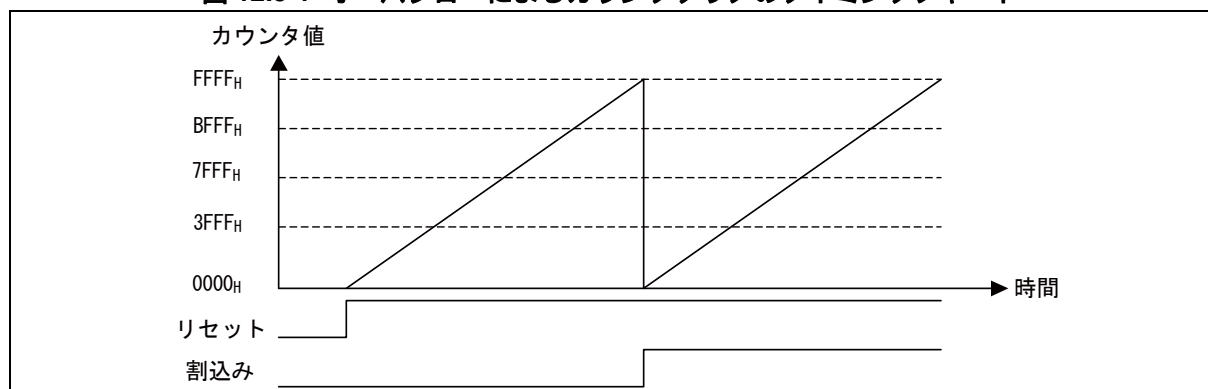
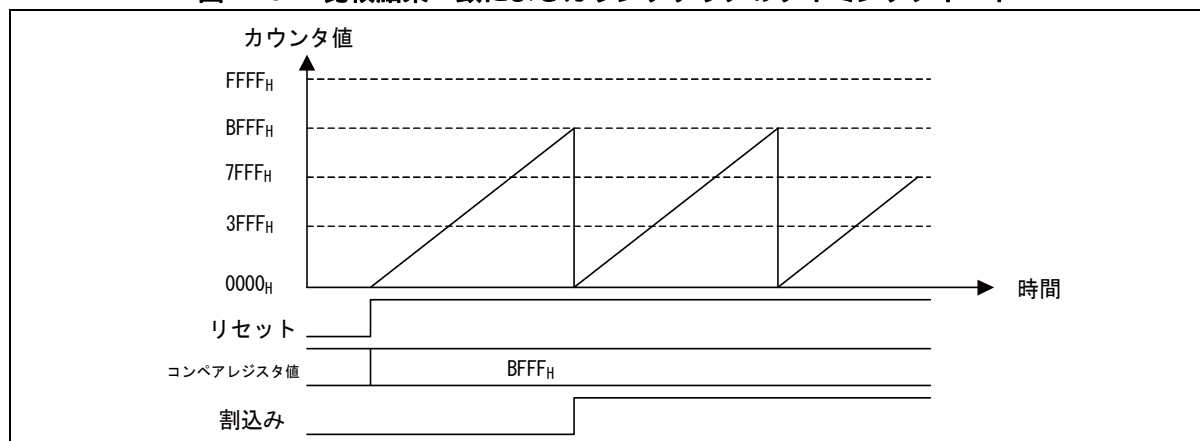


図 12.3-2 比較結果一致によるカウンタクリアのタイミングチャート



12.3.2 16 ビットアウトプットコンペアの動作

16 ビットアウトプットコンペアは、設定されたコンペアレジスタ値と 16 ビットフリーランタイムとの値を比較して、一致していたら割込み要求フラグをセットするとともに、出力レベルを反転します。

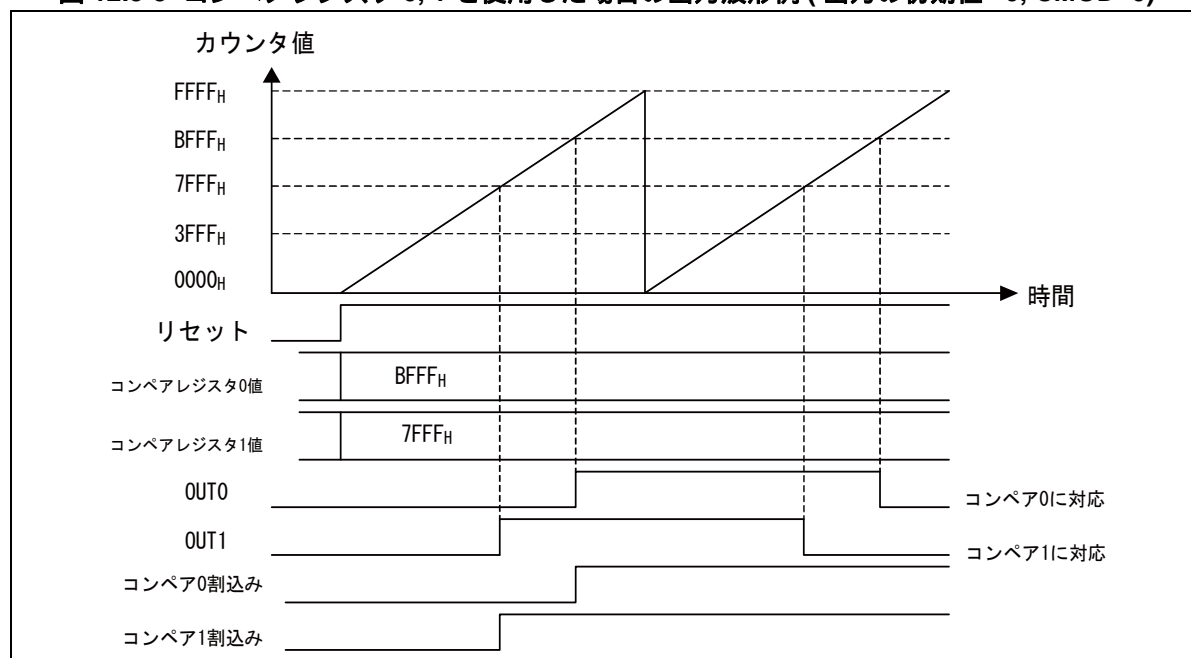
■ 出力波形例

以下に、出力波形例を示します。

● コンペアレジスタ 0, 1 を使用した場合の出力波形例

出力の初期値を "0" とした場合の出力波形例を、図 12.3-3 に示します。

図 12.3-3 コンペアレジスタ 0, 1 を使用した場合の出力波形例 (出力の初期値 =0, CMOD=0)

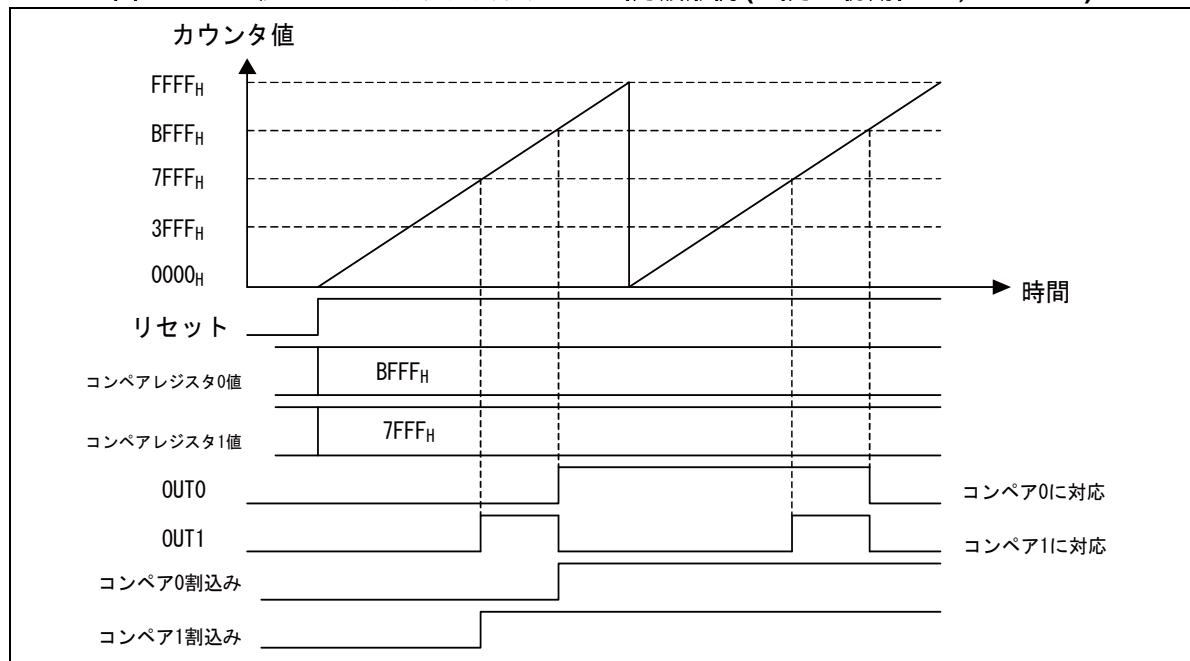


2 組のコンペアレジスタを使って出力レベルを変えることができます (CMOD=1 のとき)。

● 2 組のコンペアレジスタによる出力波形例

出力の初期値を "0" とした場合の出力波形例を、図 12.3-4 に示します。

図 12.3-4 2 組のコンペアレジスタによる出力波形例 (出力の初期値 =0, CMOD=1)



< 注意事項 >

コンペアレジスタを書き換える場合は コンペア割込みのルーチン内で行うか コンペア動作禁止の状態で行い、コンペアー一致と書込みが同時に発生しないようにしてください。

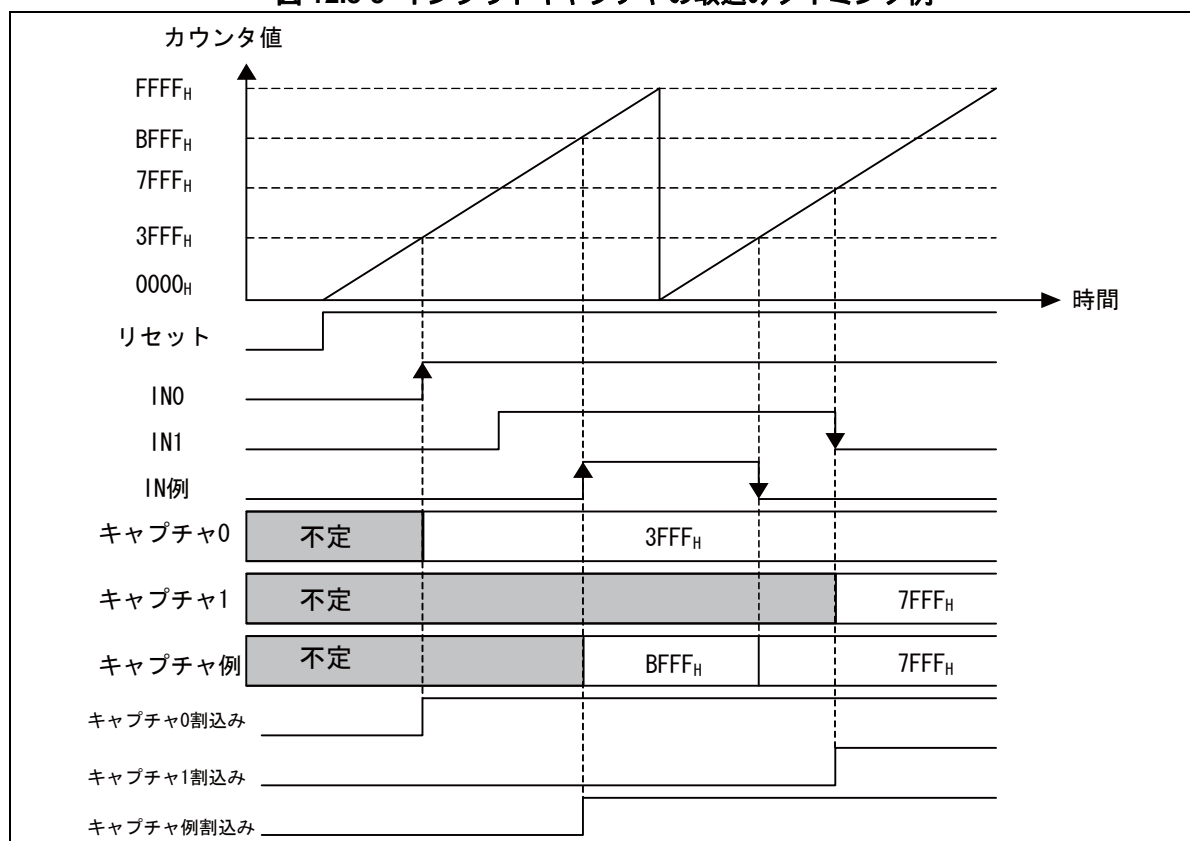
12.3.3 16 ビットインプットキャプチャの動作

16 ビットインプットキャプチャは、設定された有効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に取り込んで割込みを発生できます。

■ インプットキャプチャの取込みタイミング例

図 12.3-5 に、インプットキャプチャの取込みタイミング例を示します。

図 12.3-5 インプットキャプチャの取込みタイミング例



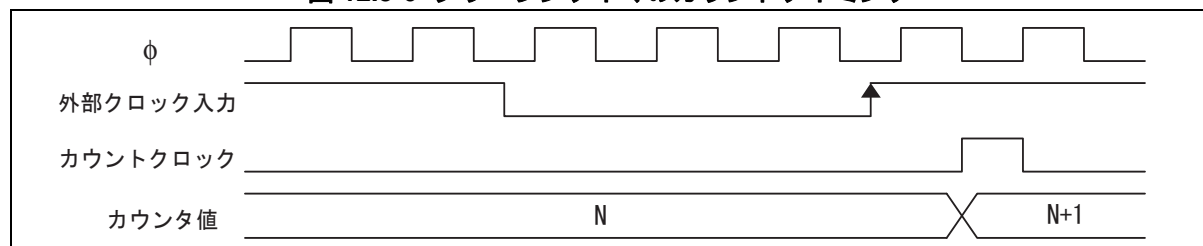
12.3.4 16 ビットフリーランタイマのタイミング

16 ビットフリーランタイマは、入力されたクロック（内部または外部クロック）によりカウントアップされます。外部クロック選択時は、立上りエッジでカウントされます。

■ フリーランタイマのカウントタイミング

図 12.3-6 に、フリーランタイマのカウントタイミングを示します。

図 12.3-6 フリーランタイマのカウントタイミング

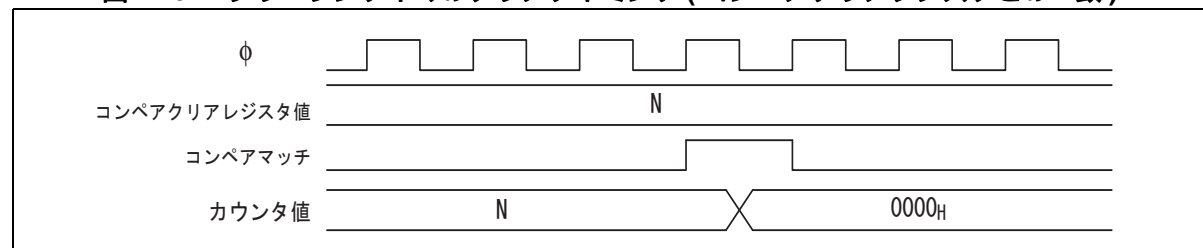


カウンタのクリアはリセット、ソフトウェアおよびコンペアクリアレジスタとの一致で行うことができます。リセットとソフトウェアでのカウンタクリアは、クリア発生とともに行われます。コンペアクリアレジスタとの一致によるカウンタクリアは、カウントタイミングに同期して行われます。

■ フリーランタイマのクリアタイミング (コンペアクリアレジスタとの一致)

図 12.3-7 に、コンペアクリアレジスタとの一致によるフリーランタイマのクリアタイミングを示します。

図 12.3-7 フリーランタイマのクリアタイミング (コンペアクリアレジスタとの一致)



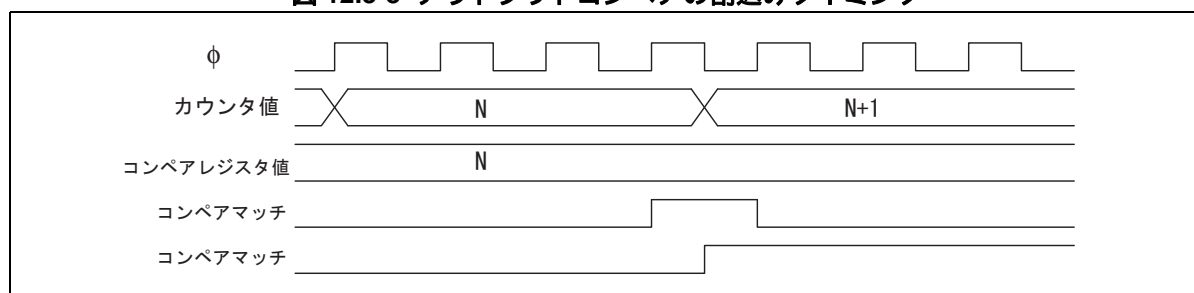
12.3.5 アウトプットコンペアのタイミング

アウトプットコンペアは、フリーランタイマと設定したコンペアレジスタの値が一致したときにコンペア一致信号が発生して出力値を反転するとともに割込みを発生できます。コンペア一致時の出力反転タイミングはカウンタタイミングに同期して行われます。

■ 割込みタイミング

図 12.3-8 に、アウトプットコンペアの割込みタイミングを示します。

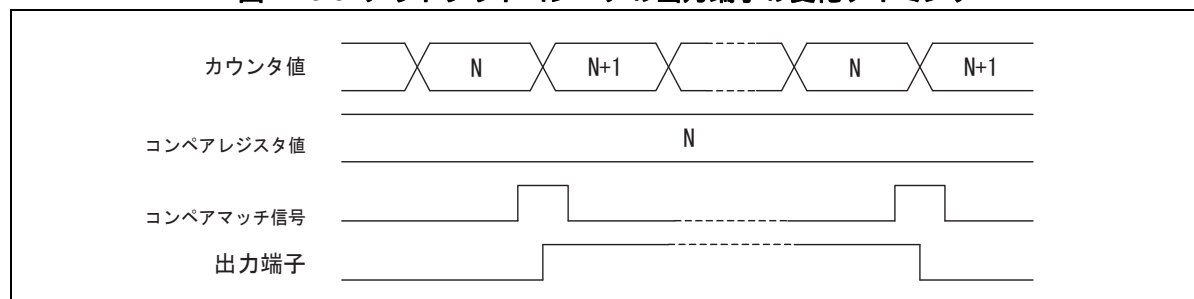
図 12.3-8 アウトプットコンペアの割込みタイミング



■ 出力端子の変化タイミング

図 12.3-9 に、アウトプットコンペアの出力端子の変化タイミングを示します。

図 12.3-9 アウトプットコンペアの出力端子の変化タイミング



< 注意事項 >

コンペアレジスタを書き換える場合は、コンペア割込みのルーチン内で行うか、コンペア動作禁止の状態で行い、コンペア一致と書込みが同時に発生しないようにしてください。

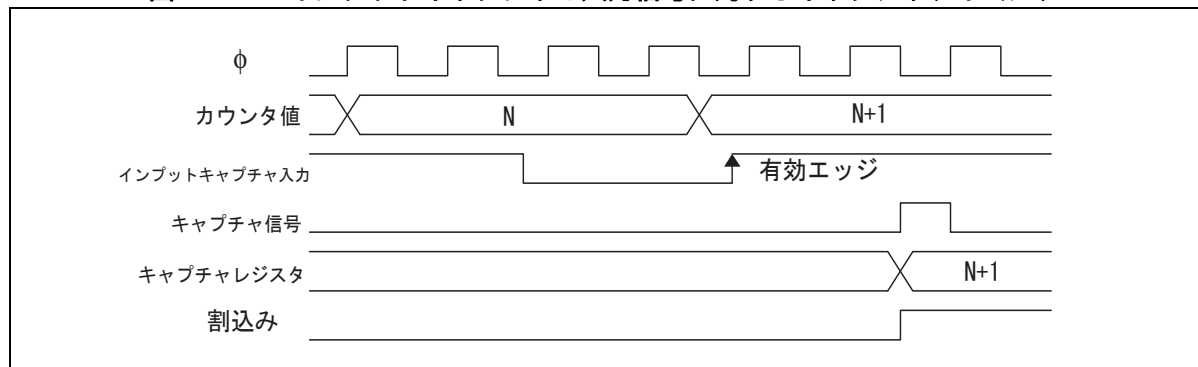
12.3.6 インプットキャプチャの入力タイミング

インプットキャプチャの入力信号に対するキャプチャタイミングを示します。

■ 入力信号に対するキャプチャタイミング

図 12.3-10 に、インプットキャプチャの入力信号に対するキャプチャタイミングを示します。

図 12.3-10 インプットキャプチャの入力信号に対するキャプチャタイミング



第13章

USB ファンクション

USB ファンクションの機能と概要について説明します。

13.1 USB ファンクションの概要

13.2 USB ファンクションのブロックダイアグラム

13.3 USB ファンクションのレジスタ

13.4 USB ファンクションの動作説明

13.1 USB ファンクションの概要

USB ファンクションは、USB(Universal Serial Bus) 通信プロトコルをサポートするインタフェースです。転送スピードは FULL(12Mbps) に対応して動作し、以下の特長があります。

■ USB ファンクションの特長

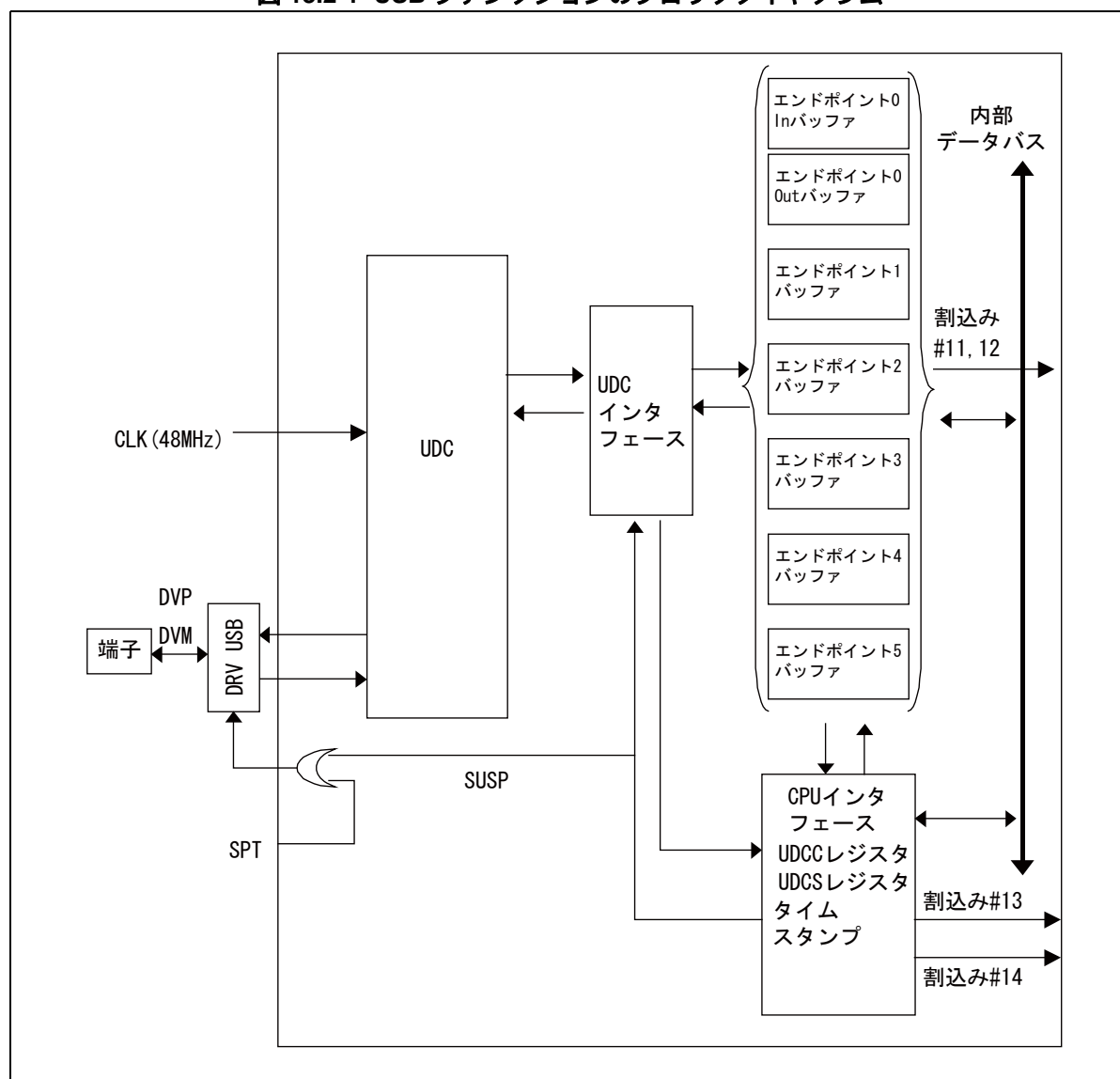
- FULL スピード (12Mbps) をサポート (USB Full Speed 対応)
- デバイスステータスは自動応答
- Bit Stripping, Bit Stuffing, CRC5, CRC16 の自動生成とチェック
- データ同期ビットによるトグルチェック
- Get/SetDescriptor, SynchFrame コマンドを除くすべての標準コマンドに自動応答 (前記 3 コマンドはクラス・ベンダーコマンドと同様の処理が可能)
- クラス・ベンダーコマンドはデータとして受信し、ファームによる応答が可能
- 最大 6 本のエンドポイントをサポート (エンドポイント 0 は control 転送に固定)
- 転送データバッファに各エンドポイントでバッファを 2 本ずつ内蔵 (エンドポイント 0 の場合は IN と OUT それぞれ専用に 1 本ずつ内蔵)
- DMA による転送データの自動転送モードをサポート (エンドポイント 0 のバッファ以外)

13.2 USB ファンクションのブロックダイアグラム

図 13.2-1 に , USB ファンクションのブロックダイアグラムを示します。

■ USB ファンクションのブロックダイアグラム

図 13.2-1 USB ファンクションのブロックダイアグラム



13.3 USB ファンクションのレジスタ

USB ファンクションで使用するレジスタの構成，および機能について説明します。

■ USB ファンクションのレジスタ一覧

図 13.3-1 USB ファンクションのレジスタ一覧

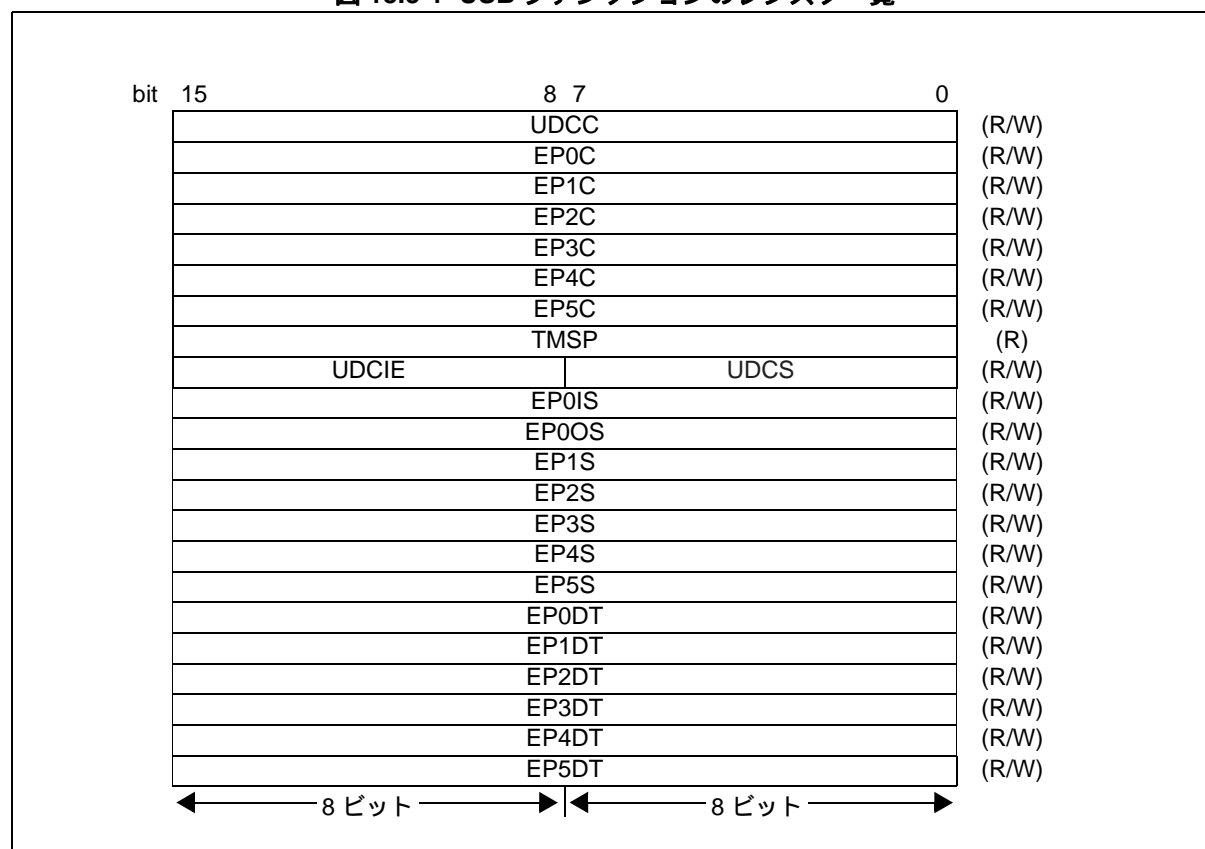


図 13.3-2 USB ファンクションレジスタのビット構成

bit	7	6	5	4	3	2	1	0	
アドレス :0000D0 _H	RST	RESUM	HCON	USTP	予約	予約	RFBK	PWC	UDC 制御レジスタ (UDCC)
bit	15	14	13	12	11	10	9	8	
アドレス :0000D1 _H	予約	予約	予約	予約	予約	予約	予約	予約	
bit	7	6	5	4	3	2	1	0	
アドレス :0000D2 _H	予約	PKS0							EP0 制御レジスタ (EP0C)
bit	15	14	13	12	11	10	9	8	
アドレス :0000D3 _H	-	-	-	-	予約	予約	STAL	予約	
bit	7	6	5	4	3	2	1	0	
アドレス :0000D4 _H	PKS1								EP1 制御レジスタ (EP1C)
bit	15	14	13	12	11	10	9	8	
アドレス :0000D5 _H	EPEN	TYPE		DIR	DMAE	NULE	STAL	PKS1	
bit	7	6	5	4	3	2	1	0	
アドレス :0000D6 _H	予約	PKS2							EP2 制御レジスタ (EP2C)
bit	15	14	13	12	11	10	9	8	
アドレス :0000D7 _H	EPEN	TYPE		DIR	DMAE	NULE	STAL	予約	
bit	7	6	5	4	3	2	1	0	
アドレス :0000D8 _H	予約	PKS3							EP3 制御レジスタ (EP3C)
bit	15	14	13	12	11	10	9	8	
アドレス :0000D9 _H	EPEN	TYPE		DIR	DMAE	NULE	STAL	予約	
bit	7	6	5	4	3	2	1	0	
アドレス :0000DA _H	予約	PKS4							EP4 制御レジスタ (EP4C)
bit	15	14	13	12	11	10	9	8	
アドレス :0000DB _H	EPEN	TYPE		DIR	DMAE	NULE	STAL	予約	
bit	7	6	5	4	3	2	1	0	
アドレス :0000DC _H	予約	PKS5							EP5 制御レジスタ (EP5C)
bit	15	14	13	12	11	10	9	8	
アドレス :0000DD _H	EPEN	TYPE		DIR	DMAE	NULE	STAL	予約	
bit	7	6	5	4	3	2	1	0	
アドレス :0000DE _H	TMSP								タイムスタンプレジスタ (TMSP)
bit	15	14	13	12	11	10	9	8	
アドレス :0000DF _H	-	-	-	-	-	TMSP			
bit	7	6	5	4	3	2	1	0	
アドレス :0000E0 _H	-	-	SUSP	SOF	BRST	WKUP	SETP	CONF	UDC ステータスレジスタ (UPCS)

(続く)

(続き)

アドレス :0000E1 _H	bit 15	14	13	12	11	10	9	8	UDC 割込み許可レジスタ (UPCIE)
	予約	予約	SUSPIE	SOFIE	BRSTIE	WKUPE	CONFN	CONFIE	
アドレス :0000E2 _H	bit 7	6	5	4	3	2	1	0	EP0I ステータスレジスタ (EP0IS)
	-	-	-	-	-	-	-	-	
アドレス :0000E3 _H	bit 15	14	13	12	11	10	9	8	
	BFINI	DRQIE	-	-	-	DRQI	-	-	
アドレス :0000E4 _H	bit 7	6	5	4	3	2	1	0	EP0O ステータスレジスタ (EP0OS)
	予約	SIZE							
アドレス :0000E5 _H	bit 15	14	13	12	11	10	9	8	
	BFINI	DRQIE	SPKIE	-	-	DRQO	SPK	予約	
アドレス :0000E6 _H	bit 7	6	5	4	3	2	1	0	EP1 ステータスレジスタ (EP1S)
	SIZE								
アドレス :0000E7 _H	bit 15	14	13	12	11	10	9	8	
	BFINI	DRQIE	SPKIE	予約	BUSY	DRQ	SPK	SIZE	
アドレス :0000E8 _H	bit 7	6	5	4	3	2	1	0	EP2 ステータスレジスタ (EP2S)
	予約	SIZE							
アドレス :0000E9 _H	bit 15	14	13	12	11	10	9	8	
	BFINI	DRQIE	SPKIE	予約	BUSY	DRQ	SPK	予約	
アドレス :0000EA _H	bit 7	6	5	4	3	2	1	0	EP3 ステータスレジスタ (EP3S)
	予約	SIZE							
アドレス :0000EB _H	bit 15	14	13	12	11	10	9	8	
	BFINI	DRQIE	SPKIE	予約	BUSY	DRQ	SPK	予約	
アドレス :0000EC _H	bit 7	6	5	4	3	2	1	0	EP4 ステータスレジスタ (EP4S)
	予約	SIZE							
アドレス :0000ED _H	bit 15	14	13	12	11	10	9	8	
	BFINI	DRQIE	SPKIE	予約	BUSY	DRQ	SPK	予約	
アドレス :0000EE _H	bit 7	6	5	4	3	2	1	0	EP5 ステータスレジスタ (EP5S)
	予約	SIZE							
アドレス :0000EF _H	bit 15	14	13	12	11	10	9	8	
	BFINI	DRQIE	SPKIE	予約	BUSY	DRQ	SPK	予約	
アドレス :0000F0 _H	bit 7	6	5	4	3	2	1	0	EP0 データレジスタ (EP0DT)
	BFDT								
アドレス :0000F1 _H	bit 15	14	13	12	11	10	9	8	
	BFDT								
アドレス :0000F2 _H	bit 7	6	5	4	3	2	1	0	EP1 データレジスタ (EP1DT)
	BFDT								

(続く)

(続き)

bit	15	14	13	12	11	10	9	8	
アドレス :0000F3 _H	BFD _T								
bit	7	6	5	4	3	2	1	0	
アドレス :0000F4 _H	BFD _T								EP2 データレジスタ (EP2DT)
bit	15	14	13	12	11	10	9	8	
アドレス :0000F5 _H	BFD _T								
bit	7	6	5	4	3	2	1	0	
アドレス :0000F6 _H	BFD _T								EP3 データレジスタ (EP3DT)
bit	15	14	13	12	11	10	9	8	
アドレス :0000F7 _H	BFD _T								
bit	7	6	5	4	3	2	1	0	
アドレス :0000F8 _H	BFD _T								EP4 データレジスタ (EP4DT)
bit	15	14	13	12	11	10	9	8	
アドレス :0000F9 _H	BFD _T								
bit	7	6	5	4	3	2	1	0	
アドレス :0000FA _H	BFD _T								EP5 データレジスタ (EP5DT)
bit	15	14	13	12	11	10	9	8	
アドレス :0000FB _H	BFD _T								

13.3.1 UDC 制御レジスタ (UDCC)

UDC 制御レジスタ (UDCC) は、UDC コア回路の制御を行います。

■ UDC 制御レジスタ (UDCC)

図 13.3-3 に、UDC 制御レジスタ (UDCC) のビット構成を示します。

図 13.3-3 UDC 制御レジスタ (UDCC) のビット構成

アドレス	bit	7	6	5	4	3	2	1	0	
0000D0 _H		RST	RESUM	HCON	USTP	予約	予約	RFBK	PWC	UDC 制御レジスタ
		1	0	1	0	0	0	0	0	初期値
		R/W	R/W	R/W	R/W	-	-	R/W	R/W	アクセス
アドレス	bit	15	14	13	12	11	10	9	8	
0000D1 _H		予約	予約	予約	予約	予約	予約	予約	予約	
		0	0	0	0	0	0	0	0	初期値
		-	-	-	-	-	-	-	-	アクセス

< 注意事項 >

UDC 制御レジスタ (UDCC) は、bit7 の RST=1 のときに設定し、USB 動作中に書き換えないようにしてください。ただし、bit6 の RESUM と bit4 の USTP は除きます。bit6 の RESUM は、USB のサスペンド状態において下記のコマンドによるリモートウェイクアップ許可状態でのみセット、リセットを行ってください。

ストップモード時は、ストップモード状態に入る前に bit4 の USTP へ "1" をセットしてください。

またストップモード解除時は UDCS の SUSP、UDCC の USTP の順に "0" に設定してください。

以下に、UDC 制御レジスタ (UDCC) の各ビットの機能を説明します。

[bit15 ~ bit8] 予約ビット

予約ビットです。必ず "0" を書き込んでください。常に "0" が読み出されます。

[bit7] RST (ファンクションリセットビット)

USB ファンクションにチップのシステムリセット相当の個別リセットを行います。HOST PC とのケーブル接続時に RST ビットにより USB ファンクションにリセットを行います。初期値は "1" でリセット状態ですので "0" ライトで解除を行ってください。

RST	動作モード
0	USB ファンクションのリセット解除
1	USB ファンクションをリセット

< 注意事項 >

RST ビットはタイムスタンプレジスタ，UDC ステータスレジスタ，割込み許可レジスタの該当ビットを同時に初期化します。また，EP0I, EP0O, EP1 ~ EP5 ステータスレジスタのBFINIも同時にセットしますので，初期設定の後，RST ビットのクリアを行い (BFINI はクリアされません)，使用するエンドポイントの BFINI クリアの順で行ってください。

[bit6] RESUM (リジューム設定ビット)

リモートウェイクアップ許可状態のとき (ホストより SET_FEATURE コマンドで DEVICE_REMOTE_WAKEUP ビットがセットされている)，かつサスペンド状態のとき，RESUM ビットに "1" ライトでリジュームの開始となります。リジューム指示は RESUM ビットに "1" 設定後 "0" ライトでクリアしてください。

RESUM	動作モード
0	USB リジューム 開始指示ビット解除
1	USB リジューム 開始指示

[bit5] $\overline{\text{HCON}}$ (ホスト接続ビット)

外付けプルアップ抵抗と USB データラインとの間のスイッチを制御し，HOST PC またはハブとの接続を認識させます。

$\overline{\text{HCON}}$	動作モード
0	HOST PC または HUB と接続
1	HOST PC または HUB と切断状態

< 注意事項 >

外付けプルアップ抵抗が ON 状態で HOST，または HUB から接続を認識された場合でも， $\overline{\text{HCON}}$ ビットが "1" の間は USB バスのバスリセット，コマンドは無視します。

[bit4] USTP (USB 動作クロック停止ビット)

USTP ビットのセットにより USB 動作部のクロックを停止させます。USB を動作させない場合に USTP ビットのセットにより消費電力を低減できます。

USTP	動作モード
0	通常モード
1	USB 動作部のクロック停止

< 注意事項 >

USTP ビットのセットはストップモードにしない場合、RST=1 にした後、リセットが確実にかかるように FULL スピード時には 3 サイクル後、LOW スピード (ホストモードの場合のみ対応) 時には 43 サイクル待った後に行ってください。USTP ビットのクリアは RST のクリアと同時に構いません。

[bit3, bit2] 予約ビット

予約ビットです。必ず "0" を書き込んでください。常に "0" が読み出されます。

[bit1] RFBK: データトグルモード選択ビット (レートフィードバックモード)

USB の割込み転送時のデータトグルモードの選択ビットです。

RFBK	動作モード
0	交代データトグルモードの選択 転送が問題なく完了したときにデータ PID をトグル
1	データトグルモードの選択 無条件にデータ PID をトグル

データトグルモードの選択は ISO 転送においてレートフィードバック情報に使用することができます。

[bit0] PWC (電源制御ビット)

USB ファンクションの動作電源モード (自己電源 , バス電源) を指定します。

(PWC ビットの設定は標準コマンド GetStatus に反映します)

PWC	動作モード
0	バス電源
1	自己電源

13.3.2 EP0 制御レジスタ (EP0C)

EP0 制御レジスタ (EP0C) は、エンドポイント 0 に関する制御を行います。

■ EP0 制御レジスタ (EP0C)

図 13.3-4 に、EP0 制御レジスタ (EP0C) のビット構成を示します。

図 13.3-4 EP0 制御レジスタ (EP0C) のビット構成

アドレス	bit	7	6	5	4	3	2	1	0	
0000D2 _H		予約	PKS0							EP0 制御レジスタ
		0	1	0	0	0	0	0	0	初期値
		-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	アクセス
アドレス	bit	15	14	13	12	11	10	9	8	
0000D3 _H		-	-	-	-	予約	予約	STAL	予約	
		x	x	x	x	0	0	0	0	初期値
		-	-	-	-	-	-	R/W	-	アクセス

< 注意事項 >

EP0 制御レジスタ (EP0C) は、bit9 の STAL を除き UDC 制御レジスタ (UDCC)bit7 の RST, EP0I/EP0O ステータスレジスタ (EP0IS/EP0OS)bit7 の BFINI が共に "1" のときに設定して、USB 動作中に書き換えないようにしてください。

以下に、EP0 制御レジスタ (EP0C) の各ビットの機能を説明します。

[bit15 ~ bit12] 未定義ビット

書込みは動作に影響しません。読出し時は不定です。

[bit11, bit10] 予約ビット

予約ビットです。必ず "00_B" を書き込んでください。

常に "00_B" が読み出されます。

[bit9] STAL (STALL エンドポイント 0 設定ビット)

STAL ビットのセットによりエンドポイント 0 を STALL 状態 (STALL 応答) にできます。

STAL	動作モード
0	STALL 状態の解除
1	STALL 状態 (STALL 応答) のセット

< 注意事項 >

STAL ビットのセット中はホストに対し、STALL 応答し続けます。STAL ビットの解除の後、正常な SETUP パケットを受信した時に STALL 状態から復帰します。

[bit8, bit7] 予約ビット

予約ビットです。必ず "00_B" を書き込んでください。

常に "00_B" が読み出されます。

[bit6 ~ bit0] PKS0 (パケットサイズ エンドポイント 0 設定ビット)

1 パケットでの最大転送バイト数を指定します。エンドポイント 0 の指定可能なパケットの最大転送バイトは 64 バイトで、IN, OUT 共通の設定となります。

< 例 > "08_H" 8 バイト, "40_H" 64 バイト (最大指定値)

< 注意事項 >

最大転送バイト数 (40_H) 以上の設定と "00_H" の設定は禁止です。

13.3.3 EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C)

EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) は、エンドポイント 1 ~ 5 に関する制御を行います。

■ EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C)

図 13.3-5 に EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) のビット構成を示します。

図 13.3-5 EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) のビット構成

アドレス	bit	7	6	5	4	3	2	1	0	
EP1C 0000D4 _H		PKS1								
		0	0	0	0	0	0	0	0	初期値
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	アクセス
アドレス	bit	7	6	5	4	3	2	1	0	
EP2C 0000D6 _H		予約	PKS5 ~ PKS2							
EP3C 0000D8 _H		0	1	0	0	0	0	0	0	初期値
EP4C 0000DA _H		-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	アクセス
EP5C 0000DC _H										
アドレス	bit	15	14	13	12	11	10	9	8	
EP1C 0000D5 _H		EPEN	TYPE		DIR	DMAE	NULE	STAL	PKS1	
		0	1	1	0	0	0	0	1	初期値
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	アクセス
アドレス	bit	15	14	13	12	11	10	9	8	
EP2C 0000D7 _H		EPEN	TYPE		DIR	DMAE	NULE	STAL	予約	
EP3C 0000D9 _H		0	1	1	0	0	0	0	0	初期値
EP4C 0000DB _H		R/W	R/W	R/W	R/W	R/W	R/W	R/W	-	アクセス
EP5C 0000DD _H										

< 注意事項 >

EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) は、DMAE, NULE, STAL を除き UDC 制御レジスタ (UDCC)bit7 の RST, EP0 ~ EP5 ステータスレジスタ (EP1S ~ EP5S)bit15 の BFINI が共に "1" のときに設定して、USB 動作中に書き換えないようにしてください。

以下に、EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) の各ビットの機能を説明します。

[bit15] EPEN (エンドポイント 1 ~ 5 許可ビット)

エンドポイントを有効にします。EPEN ビットの設定によりファンクションで使用するエンドポイントとしてホストから構成されます。EP1 ~ EP5 制御レジスタの TYPE, DIR, PKS が構成情報として有効となります。

EPEN	動作モード
0	エンドポイントは無効
1	エンドポイントを有効とする

[bit14, bit13] TYPE (エンドポイント転送タイプ選択ビット)

エンドポイントがサポートする転送タイプを指定します。

TYPE	動作モード
00	指定禁止
01	Isochronous 転送
10	Bulk 転送
11	Interrupt 転送

< 注意事項 >

USB2.0 の Isochronous 転送では、Alternate 値が "0" でかつ、最大パケット数が "0" の状態を初期設定とすることが仕様に追加されました。本 USB ファンクションの場合、ホストから Alternate interface setting コマンドで Alternate 値を "0" 以外に設定すると、ホストに対し自動的にSTALL応答してしまい、Alternate値を変更できません。したがって、Alternate 値の変更による最大パケット数の設定変更には対応できません。

[bit12] DIR (エンドポイント転送方向選択ビット)

エンドポイントがサポートする転送方向を指定します。

DIR	ファンクション動作モード	HOST 動作モード (EP1, EP2 のみ)
0	OUT エンドポイント	IN エンドポイント
1	IN エンドポイント	OUT エンドポイント

[bit11] DMAE (DMA 自動転送許可ビット)

転送データの送受信バッファへの書込みあるいは、読出しに DMA を使用し、DMA に設定したデータ転送数までホストからの IN, OUT データ要求に同期して自動で送受信データを転送するモード設定です。

DMAE	動作モード
0	自動バッファ転送モードの解除
1	自動バッファ転送モードのセット

< 注意事項 >

DMAE ビットのセット中は、CPU による送受信バッファへのアクセスは禁止です。OUT 方向のデータ転送時にはDMAの転送数の設定はEP1 ~ EP5制御レジスタ (EP1C ~ EP5C) の PKS 設定数の倍数としてください。

[bit10] NULE (NULL 自動転送許可ビット)

自動バッファ転送モードがセットされている状態 (DMAE=1) で、IN 方向のデータ転送要求がきたときに、最後のパケット転送を検出し、0 バイトのデータ転送を自動で送信するモードの設定を行います。

NULE	動作モード
0	NULL 自動転送モードの解除
1	NULL 自動転送モードのセット

< 注意事項 >

OUT 方向のデータ転送時や自動バッファ転送モードがセットされていないときには NULE ビットの設定は通信に影響しません。

[bit9] STAL (STALL 設定ビット)

STAL ビットのセットによりエンドポイントを STALL 状態 (STALL 応答) にできます。

STAL	動作モード
0	STALL 状態の解除
1	STALL 状態 (STALL 応答) のセット

< 注意事項 >

STAL ビットのセット中は HOST に対し、STALL 応答し続けます。STALL 状態からの復帰は STAL ビットの解除の後、ホストからの ClearFutcher コマンドにより可能です。

EP2 ~ EP5:[bit8, bit7] 予約ビット

このビットは、EP2 ~ EP5 の場合、予約ビットです。必ず、"0" を書き込んでください。常に "0" が読み出されます。

EP1:[bit8 ~ bit0] PKS (パケットサイズ設定ビット)

EP2 ~ EP5:[bit6 ~ bit0] PKS (パケットサイズ設定ビット)

1 パケットでの最大転送数を指定します。エンドポイント 1 ~ 5 の指定可能なパケットの最大転送数は下記になります。

EndPoint	最大転送数	設定可能範囲
1	256 バイト (奇数設定可能)	001 _H ~ 100 _H
2 ~ 5	64 バイト (奇数設定可能)	01 _H ~ 40 _H

< 注意事項 >

最大転送数 (100_H, 40_H) 以上の設定と "00_H" の設定は禁止です。エンドポイント 2 ~ 5 は bit8, bit7 に "00" 書込みをしてください。さらに, 自動バッファ転送モード (DMAE=1) を使用する場合は該当するエンドポイントでの 0 ~ 2 設定は禁止となります。

13.3.4 タイムスタンプレジスタ (TMSP)

タイムスタンプレジスタ (TMSP) は、SOF パケット受信時のフレーム番号の表示を行います。

■ タイムスタンプレジスタ (TMSP)

図 13.3-6 に、タイムスタンプレジスタ (TMSP) のビット構成を示します。

図 13.3-6 タイムスタンプレジスタ (TMSP) のビット構成

アドレス bit	7	6	5	4	3	2	1	0	
0000DE _H	TMSP								タイムスタンプレジスタ
	0	0	0	0	0	0	0	0	初期値
	0	0	0	0	0	0	0	0	RST リセット
	R	R	R	R	R	R	R	R	アクセス
アドレス bit	15	14	13	12	11	10	9	8	
0000DF _H	-	-	-	-	-	TMSP			タイムスタンプレジスタ
	X	X	X	X	X	0	0	0	初期値
	X	X	X	X	X	0	0	0	RST リセット
	-	-	-	-	-	R	R	R	アクセス

以下に、タイムスタンプレジスタ (TMSP) の各ビットの機能を説明します。

[bit15 ~ bit11] 未定義ビット

書込みは動作に影響しません。読出し時は不定です。

[bit10 ~ bit0] TMSP (タイムスタンプビット)

SOF パケットの受信によるフレーム番号を示します。フレーム番号は SOF パケットの受信時に更新されます。

13.3.5 UDC ステータスレジスタ (UDCS)

UDC ステータスレジスタ (UDCS) は、USB 通信上のバスの状態や、特定のコマンド受信状態を示すレジスタです。SETP を除く各ビットは割り込み要因となっており、対応する割り込み許可ビットが有効指定されていれば CPU へ割り込みが発生します。

■ UDC ステータスレジスタ (UDCS)

図 13.3-7 に、UDC ステータスレジスタ (UDCS) のビット構成を示します。

図 13.3-7 UDC ステータスレジスタ (UDCS) のビット構成

アドレス	bit	7	6	5	4	3	2	1	0	
0000E0 _H		-	-	SUSP	SOF	BRST	WKUP	SETP	CONF	UDC ステータスレジスタ
		X	X	0	0	0	0	0	0	初期値
		X	X	0	0	0	0	0	0	RST リセット
		-	-	R/W	R/W	R/W	R/W	R/W	R/W	アクセス

以下に、UDC ステータスレジスタ (UDCS) の各ビットの機能を説明します。

[bit7, bit6] 未定義ビット

書込みは動作に影響しません。読出し時は不定です。

[bit5] SUSP (サスペンド検出ビット)

USB ファンクションがサスペンド状態に移行したことを表示します。SUSP ビットは割り込み要因であり、"1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

SUSP	動作モード
0	Suspend 未検出・割り込み要因クリア
1	Suspend 検出

[bit4] SOF (SOF 受信検出ビット)

SOF パケットを受信したことを示し、タイムスタンプレジスタの値が更新されます。SOF ビットは割り込み要因であり、"1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

SOF	動作モード
0	SOF 未受信・割り込み要因クリア
1	SOF パケットを受信

[bit3] BRST (バスリセット検出ビット)

USB バスリセットの検出を表示します。BRST ビットは割込み要因であり, "1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

BRST	動作モード
0	USB バスリセットを未検出・割込み要因クリア
1	USB バスリセットを検出

< 注意事項 >

BRST ビットの検出時には UDCC レジスタの RST により USB ファンクションの初期化をして、レジスタの再設定を行ってください。

[bit2] WKUP (WakeUp 検出ビット)

USB ファンクションがサスペンド状態から復帰したことを表示します。復帰要因は RESUM ビットのセットによるリモートウェイクアップとホストからの要求によるウェイクアップがありますが, WKUP ビットはホストからの復帰要求のみで自動セットされます。WKUP ビットは割込み要因であり, "1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

WKUP	動作モード
0	HOST 要因 RESUME を未検出・割込み要因クリア
1	HOST 要因 RESUME を検出

< 注意事項 >

ホスト要求によるウェイクアップ時でも UDCC レジスタの RESUM ビットがセットされている場合, WKUP ビットはセットされません。

[bit1] SETP (SETUP ステージ検出ビット)

受信したデータが USB コントロール転送の Setup ステージであることを示します。
"1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

SETP	動作モード
0	SETUP 未受信・要因クリア
1	コントロール転送 SETUP ステージを受信

< 注意事項 >

標準コマンドの自動応答時にはセットされません。SETP ビットは割込み要因ではありません。

[bit0] CONF (コンフィグレーション検出ビット)

USB ファンクションがコンフィグレーション済みであることを表示します。CONF ビットは USB コマンドの SetConfig を正常受信したときにセットされます。CONF ビットは割込み要因であり、"1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

CONF	動作モード
0	SetConfig 未検出・割込み要因クリア
1	SetConfig を検出

13.3.6 UDC 割込み許可レジスタ (UDCIE)

UDC 割込み許可レジスタ (UDCIE) は、UDC ステータスレジスタの各割込み要因による割込み CONFN を除き、ビットごとに許可するレジスタです。

■ UDC 割込み許可レジスタ (UDCIE)

図 13.3-8 に、UDC 割込み許可レジスタ (UDCIE) のビット構成を示します。

図 13.3-8 UDC 割込み許可レジスタ (UDCIE) のビット構成

アドレス bit	15	14	13	12	11	10	9	8	
0000E1 _H	予約	予約	SUSPIE	SOFIE	BRSTIE	WKUPIE	CONFN	CONFIE	UDC 割込み許可レジスタ
	0	0	0	0	0	0	0	0	初期値
	0	0	0	0	0	0	0	0	RST リセット
	-	-	R/W	R/W	R/W	R/W	R	R/W	アクセス

以下に、UDC 割込み許可レジスタ (UDCIE) の各ビットの機能を説明します。

[bit15, bit14] 予約ビット

予約ビットです。必ず "00_B" を書き込んでください。常に "00_B" が読み出されます。

[bit13] SUSPIE (サスペンド割込み許可ビット)

UDC ステータスレジスタ "SUSP" の割込み要因による割込みを許可します。

SUSPIE	動作モード
0	SUSP 要因による割込み禁止
1	SUSP 要因による割込み許可

[bit12] SOFIE (SOF 受信割込み許可ビット)

UDC ステータスレジスタ "SOF" の割込み要因による割込みを許可します。

SOFIE	動作モード
0	SOF 要因による割込み禁止
1	SOF 要因による割込み許可

[bit11] BRSTIE (バスリセット割込み許可ビット)

UDC ステータスレジスタ "BRST" の割込み要因による割込みを許可します。

BRSTIE	動作モード
0	BRST 要因による割込み禁止
1	BRST 要因による割込み許可

[bit10] WKUPIE (WakeUp 割込み許可ビット)

UDC ステータスレジスタ "WKUP" の割込み要因による割込みを許可します。

WKUPIE	動作モード
0	WKUP 要因による割込み禁止
1	WKUP 要因による割込み許可

[bit9] CONFN (コンフィグレーション番号表示ビット)

コンフィグレーション番号を表示します。UDC ステータスレジスタ "CONF" の割込み要因セット時に更新します。

CONFN	動作モード
0	CONFIG 番号 0
1	CONFIG 番号 1

[bit8] CONFIE (コンフィグレーション割込み許可ビット)

UDC ステータスレジスタ "CONF" の割込み要因による割込みを許可します。

CONFIE	動作モード
0	CONF 要因による割込み禁止
1	CONF 要因による割込み許可

13.3.7 EP0I ステータスレジスタ (EP0IS)

EP0I ステータスレジスタ (EP0IS) は、エンドポイント 0 の In 方向転送に関するステータス表示を行います。

■ EP0I ステータスレジスタ (EP0IS)

図 13.3-9 に、EP0I ステータスレジスタ (EP0IS) のビット構成を示します。

図 13.3-9 EP0I ステータスレジスタ (EP0IS) のビット構成

アドレス bit	7	6	5	4	3	2	1	0	
0000E2 _H	-	-	-	-	-	-	-	-	EP0I ステータスレジスタ
	X	X	X	X	X	X	X	X	初期値
	X	X	X	X	X	X	X	X	BFINI リセット
	-	-	-	-	-	-	-	-	アクセス
アドレス bit	15	14	13	12	11	10	9	8	
0000E3 _H	BFINI	DRQIIE	-	-	-	DRQI	-	-	
	1	0	X	X	X	1	X	X	初期値
	1	無関係	X	X	X	1	X	X	BFINI リセット
	R/W	R/W	-	-	-	R/W	-	-	アクセス

以下に、EP0I ステータスレジスタ (EP0IS) の各ビットの機能を説明します。

[bit15] BFINI (送信バッファ初期化ビット)

転送データの送信バッファの初期化をします。また、BFINI ビットは UDC 制御レジスタ (UDCC) RST ビットのセットでも自動でセットされるようになっています。したがって、RST ビットでリセットされている場合、BFINI ビットのクリアは RST ビットのクリア後に行ってください。

BFINI	動作モード
0	初期化の解除
1	送信バッファの初期化

< 注意事項 >

BFINI ビットによる初期化ではバッファ、DRQI ビットが初期化されます。バッファの初期化をする場合、DRQI または DRQO ビットがセットされ HOST からのアクセスがないことを確認した後、必要に応じて STAL ビットをセットしてから初期化を行ってください。

[bit14] DRQIE (送信データ割込み許可ビット)

EP0I ステータスレジスタ "DRQI" の割込み要因による割込みを許可します。

DRQIE	動作モード
0	DRQI 要因による割込み禁止
1	DRQI 要因による割込み許可

[bit13 ~ bit11] 未定義ビット

書込みは動作に影響しません。読出し時は不定です。

[bit10] DRQI (送信データ割込み要求ビット)

EP0 のホストからの IN パケット転送が正常に終了し、送信バッファからデータが読み出され次の送信データの書込みが可能であることを示します。DRQI ビットは割込み要因であり、"1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

DRQI	動作モード
0	割込み要因クリア
1	送信データの書込み可能状態

< 注意事項 >

送信バッファのデータライト処理の完了後、DRQI をクリアする必要があります。また、DRQI がセットされていないときの "0" 書込みは禁止です。DRQI が "1" のとき、送信バッファにデータ書込みが可能です。また、クリアした時点で送信バッファにデータ設定が完了したことを意味します。したがって、DRQI ビットが "1" の状態で IN パケット要求があった場合は、自動で HOST に NAK 応答します。

[bit9 ~ bit0] 未定義ビット

書込みは動作に影響しません。読出し時は不定です。

13.3.8 EP0O ステータスレジスタ (EP0OS)

EP0O ステータスレジスタ (EP0OS) は、エンドポイント 0 の Out 方向転送に関するステータス表示を行います。

■ EP0O ステータスレジスタ (EP0OS)

図 13.3-10 に、EP0O ステータスレジスタ (EP0OS) のビット構成を示します。

図 13.3-10 EP0O ステータスレジスタ (EP0OS) のビット構成

アドレス bit	7	6	5	4	3	2	1	0	
0000E4 _H	予約	SIZE							EP0O ステータスレジスタ
	0	X	X	X	X	X	X	X	初期値
	0	X	X	X	X	X	X	X	BFINI リセット
	-	R	R	R	R	R	R	R	アクセス
アドレス bit	15	14	13	12	11	10	9	8	
0000E5 _H	BFINI	DRQOIE	SPKIE	-	-	DRQO	SPK	予約	
	1	0	0	X	X	0	0	0	初期値
	1	無関係	無関係	X	X	0	0	0	BFINI リセット
	R/W	R/W	R/W	-	-	R/W	R/W	-	アクセス

以下に、EP0O ステータスレジスタ (EP0OS) の各ビットの機能を説明します。

[bit15] BFINI (受信バッファ初期化ビット)

転送データの受信バッファの初期化をします。また、BFINI ビットは UDC 制御レジスタ (UDCC) RST ビットのセットでも自動でセットされるようになっています。したがって、RST ビットでリセットされている場合、BFINI ビットのクリアは RST ビットのクリア後に行ってください。

BFINI	動作モード
0	初期化の解除
1	受信バッファの初期化

< 注意事項 >

BFINI ビットによる初期化ではバッファ、DRQO、SPK ビットが初期化されます。バッファの初期化をする場合、DRQI または DRQO ビットがセットされ HOST からのアクセスがないことを確認した後、必要に応じて STAL ビットをセットしてから初期化を行ってください。

[bit14] DRQOIE (受信データ割込み許可ビット)

EP0O ステータスレジスタ "DRQO" の割込み要因による割込みを許可します。

DRQOIE	動作モード
0	DRQO 要因による割込み禁止
1	DRQO 要因による割込み許可

[bit13] SPKIE (ショートパケット割込み許可ビット)

EP0O ステータスレジスタ "SPK" の割込み要因による割込みを許可します。

SPKIE	動作モード
0	SPK 要因による割込み禁止
1	SPK 要因による割込み許可

[bit12, bit11] 未定義ビット

書込みは動作に影響しません。読出し時は不定です。

[bit10] DRQO (受信データ割込み要求ビット)

EP0 のホストからの OUT パケット転送が正常に終了し、受信バッファにデータが書き込まれ受信データを読み出すことが可能であることを示します。DRQO ビットは割込み要因であり、"1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

DRQO	動作モード
0	割込み要因クリア
1	受信データの読出し可能状態

< 注意事項 >

受信バッファのデータリード処理完了後、DRQO をクリアする必要があります。また、DRQO がセットされていないときの "0" 書込みは禁止です。DRQO が "1" のとき、受信バッファは更新されません。クリアした時点で更新許可となります。DRQO が "1" の状態で OUT パケット要求があった場合は、自動で HOST に NAK 応答します。

[bit9] SPK (ショートパケット割込み要求ビット)

ホストからの転送データ数が正常受信時に EP0 制御レジスタ (EP0C) の PKS で設定したマックスパケット数に満たない場合 (0 バイトを含みます) を示します。SPK ビットは割込み要因であり , "1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

SPK	動作モード
0	マックスパケット転送数分を受信
1	マックスパケット転送数未満を受信

[bit8, bit7] 予約ビット

これらのビットは , 予約ビットです。書込みは動作に影響しません。読出し時には "0" が読み出せます。

[bit6 ~ bit0] SIZE (パケットサイズ表示ビット)

EP0 の OUT パケット転送終了後に受信バッファに書き込まれたデータバイト数が表示されます。SIZE ビットは EP0O ステータスレジスタ (EP0OS) の DRQO の割込み要因がセットされたときに有効な値に更新されます。

< 例 > 8 バイト "08_H", 64 バイト "40_H" (最大値)

13.3.9 EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S)

EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S) は、エンドポイント 1 ~ 5 に関するステータス表示を行います。

■ EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S)

図 13.3-11 に、EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S) のビット構成を示します。

図 13.3-11 EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S) のビット構成

アドレス bit	7	6	5	4	3	2	1	0	
EP1S 0000E6 _H	SIZE								
	X	X	X	X	X	X	X	X	初期値
	X	X	X	X	X	X	X	X	BFINI リセット
	R	R	R	R	R	R	R	R	アクセス
アドレス bit	7	6	5	4	3	2	1	0	
EP2S 0000E8 _H	予約	SIZE							
EP3S 0000EA _H	0	X	X	X	X	X	X	X	初期値
EP4S 0000EC _H	0	X	X	X	X	X	X	X	BFINI リセット
EP5S 0000EE _H	-	R	R	R	R	R	R	R	アクセス
アドレス bit	15	14	13	12	11	10	9	8	
EP1S 0000E7 _H	BFINI	DRQIE	SPKIE	予約	BUSY	DRQ	SPK	SIZE	
	1	0	0	-	0	0	0	X	初期値
	1	無関係	無関係	-	無関係	0	0	X	BFINI リセット
	R/W	R/W	R/W	-	R	R/W	R/W	R	アクセス
アドレス bit	15	14	13	12	11	10	9	8	
EP2S 0000E9 _H	BFINI	DRQIE	SPKIE	予約	BUSY	DRQ	SPK	予約	
EP3S 0000EB _H	1	0	0	-	0	0	0	0	初期値
EP4S 0000ED _H	1	無関係	無関係	-	無関係	0	0	0	BFINI リセット
EP5S 0000EF _H	R/W	R/W	R/W	-	R	R/W	R/W	-	アクセス

以下に、EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S) の各ビットの機能を説明します。

[bit15] BFINI (送受信バッファ初期化ビット)

転送データの送受信バッファの初期化をします。また、BFINI ビットは UDC 制御レジスタ (UDCC) RST ビットのセットでも自動でセットされるようになっています。したがって、RST ビットでリセットされている場合、BFINI ビットのクリアは RST ビットのクリア後に行ってください。

BFINI	動作モード
0	初期化の解除
1	送受信バッファの初期化

< 注意事項 >

EP1 ~ EP5 の送受信バッファはダブルバッファ構成で、BFINI ビットによる初期化ではダブルバッファ同時に初期化が行われ、DRQ、SPK ビットも初期化されます。バッファの初期化をする場合、DRQ ビットがセットされ BUSY ビットにより HOST からのアクセスがないことを確認した後、STAL ビットをセットしてから初期化を行ってください。

[bit14] DRQIE (パケット転送割込み許可ビット)

EP1 ~ EP5 ステータスレジスタ "DRQ" の割込み要因による割込みを許可します。

DRQIE	動作モード
0	DRQ 要因による割込み禁止
1	DRQ 要因による割込み許可

< 注意事項 >

自動バッファ転送モード (DMAE=1) を使用する場合は DMA の設定、転送を許可してから DRQIE ビットの許可をするようにしてください。

[bit13] SPKIE (ショートパケット割込み許可ビット)

EP1 ~ EP5 ステータスレジスタ "SPK" の割込み要因による割込みを許可します。

SPKIE	動作モード
0	SPK 要因による割込み禁止
1	SPK 要因による割込み許可

[bit12] 予約ビット

このビットは、予約ビットです。書込みは動作に影響しません。読出し時は不定です。

[bit11] BUSY (ビジーフラグビット)

HOST から送受信バッファへの書込み，または読出しアクセス中であることを示します。BUSY ビットは自動でセット，リセットされます。

BUSY	動作モード
0	HOST からのアクセスなし
1	HOST からの書込みまたは読出し動作中

< 注意事項 >

DRQ ビットがセットされた状態で BUSY ビットもセットされている場合，ダブルバッファのうち CPU または DMA アクセスしているバッファとは別のバッファが HOST からのアクセス中であることを意味します。また，通常であれば BUSY ビットによる制御は必要ありませんが，BFINI のセットによるバッファの初期化をする場合，DRQ ビットがセットされ BUSY ビットにより HOST からのアクセスがないことを確認した後，STAL ビットをセットしてから初期化を行ってください。

[bit10] DRQ (パケット転送割込み要求ビット)

EP1 ~ EP5 のパケット転送が正常に終了し，データ処理が必要であることを示します。DRQ ビットは割込み要因であり，"1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

DRQ	動作モード
0	割込み要因クリア
1	パケット転送が正常に終了

< 注意事項 >

自動バッファ転送モード (DMAE=1) を使用しない場合は，送受信バッファのデータリードまたはデータライト処理が済んだら DRQ ビットに "0" 書込みする必要があります。DRQ ビットをクリアした時点でアクセスバッファを切り換えます。転送方向が IN 方向の設定の場合，DRQ ビットが "1" でバッファにデータ書込みせずにクリアした場合，0 バイトデータを設定したことになります。初期設定において EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) の DIR を "1" にセットした場合，対応するエンドポイントの DRQ ビットも同時にセットされます。また，DRQ ビットがセットされていないときの "0" 書込みは禁止です。

[bit9] SPK (ショートパケット割込み要求ビット)

ホストからの転送データ数が正常受信時に EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) の PKS で設定したマックスパケット数に満たない場合 (0 バイトを含みます) を示します。SPK ビットは割込み要因であり, "1" 書込みは無視します。"0" 書込みでクリアしてください。リードモディファイライト時は "1" が読み出されます。

SPK	動作モード
0	マックスパケット転送数分を受信
1	マックスパケット転送数未満を受信

< 注意事項 >

IN 方向のデータ転送時に SPK ビットはセットされません。

EP2 ~ EP5:[bit8, bit7] 予約ビット

EP2 ~ EP5 の場合, これらのビットは, 予約ビットです。書込みは動作に影響しません。読み出し時には "0" が読み出せます。

EP1:[bit8 ~ bit0]SIZE (パケットサイズ表示ビット)**EP2 ~ EP5:[bit6 ~ bit0] SIZE (パケットサイズ表示ビット)**

EP1 ~ EP5 の OUT パケット転送終了時に受信バッファに書き込まれたデータバイト数が表示されます。SIZE ビットは EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S) の DRQ の割込み要因がセットされたときに有効な値に更新されます。

EndPoint1 ~ EndPoint5 の最大転送数は下記のとおりになります。

EndPoint	最大転送数	表示範囲
1	256 バイト	000 _H ~ 100 _H
2 ~ 5	64 バイト	00 _H ~ 40 _H

< 注意事項 >

SIZE ビットは OUT 方向転送時での HOST からのバッファ書込みデータ数がセットされるので, IN 方向時に読み出された値は意味をもちません。

13.3.10 EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT)

EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT) は、エンドポイント 0 ~ 5 に関する転送データの送受信バッファへのリードまたはライトのアクセスレジスタになります。

■ EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT)

図 13.3-12 に、EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT) のビット構成を示します。

図 13.3-12 EP0~5 データレジスタ (EP0DT ~ EP5DT) のビット構成

アドレス	bit	7	6	5	4	3	2	1	0	
EP0DT 0000F0H		BFD								
EP1DT 0000F2H		X	X	X	X	X	X	X	X	初期値
EP2DT 0000F4H		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	アクセス
EP3DT 0000F6H										
EP4DT 0000F8H										
EP5DT 0000FAH										
アドレス	bit	15	14	13	12	11	10	9	8	
EP0DT 0000F1H		BFD								
EP1DT 0000F3H		X	X	X	X	X	X	X	X	初期値
EP2DT 0000F5H		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	アクセス
EP3DT 0000F7H										
EP4DT 0000F9H										
EP5DT 0000FBH										

以下に、EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT) の各ビットの機能を説明します。

[bit15 ~ bit0] BFD: エンドポイント用送受信バッファデータビット

各エンドポイント用送受信バッファへのデータリード、ライトレジスタです。DMA 転送による BFD レジスタへのアクセスはワードアクセスのみ対応しています。奇数分のデータ数を転送する場合で DMA 転送の場合、最後のデータ転送をバイト転送に設定することで奇数分の DMA 転送が可能です。CPU アクセスでのワード転送の場合でも、DMA 転送と同様に最後にバイト転送を行います。

< 注意事項 >

EP0DT ~ EP5DT レジスタへの CPU アクセスはバイト、ワード共に可能で、バイトの場合は最初に bit7 ~ bit0 へのアクセスとし、次は bit15 ~ bit8 へのアクセスとして、以後上位、下位を交互にアクセスしていただきます。ビット命令による EP0DT ~ EP5DT レジスタへのアクセスは禁止です。

13.4 USB ファンクションの動作説明

USB ファンクションは、USB(Universal Serial Bus) 通信プロトコルに対応し、基本的なプロトコル動作(ハンドシェーク)をハードウェアがサポートしています。したがって、通信データのみを処理することで USB 通信が実現できます。

■ USB ファンクションの動作

USB ファンクションは USB プロトコルをサポートするホストコントローラと双方向の packets 転送を行います。ホスト PC とデバイスの接続、構成はエニューメレーションにより実施されます。そのあとにデバイスドライバを使用した各種の転送タイプでの通信が行われます。

エニューメレーションを例にホスト PC とデバイスの USB 通信の動作について説明します。

全体の処理内容を理解するためのレジスタおよび USB パケットの動きを示します。

● エニューメレーション処理

USB が動作する上で一番初めにホスト PC とデバイスとの接続を確立する処理です。USB のバス上にどのようなデバイスが接続されているかホスト PC が USB コントロール転送(USB の転送タイプ)を使って調査します。(USB 仕様で規定)これは 6 本あるエンドポイントのうち EP0(EndPoint0)を使用します(USB 仕様)。

EP1 ~ EP5 を使用するためには USB バス上で

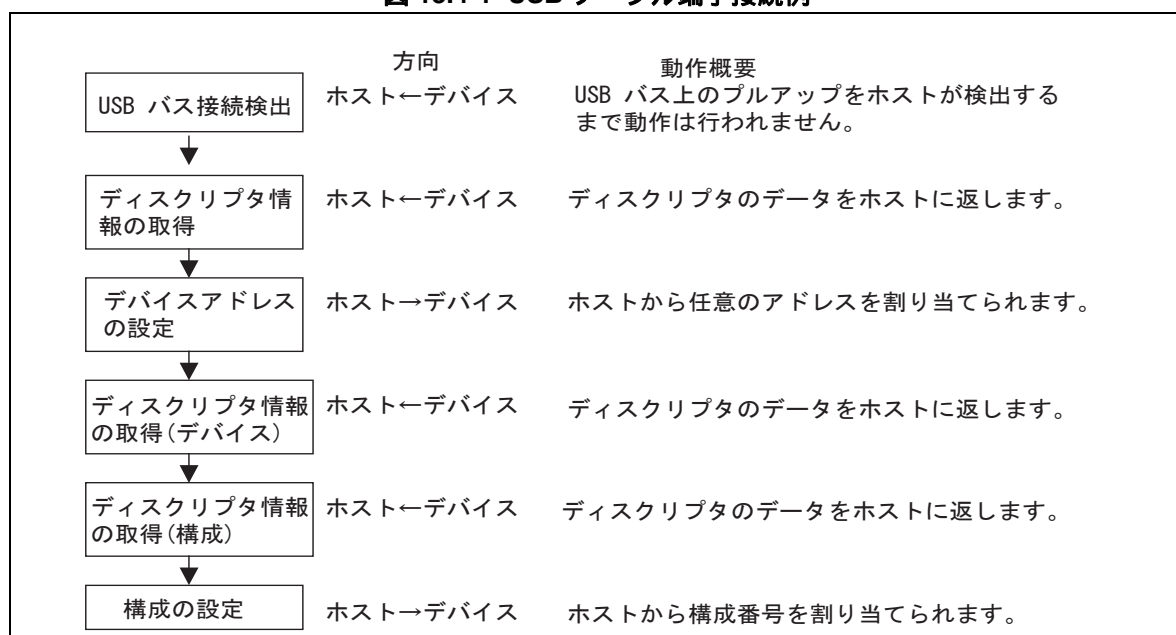
USB バスリセット

SET_Address によるアドレスセット

SET_Config による構成セット

を受信しないと動作しません。

図 13.4-1 USB ケーブル端子接続例



● 接続検出

デバイスからホスト PC に通知します。

ホストは USB バスの 2 本の信号線 (D+, D-) を監視し、どちらかの信号が "H" レベルになることによりデバイスが接続されることを認識します。

自己電源デバイスとして使用する場合は「13.4.1 接続検出と切断検出」を参照してください。バス電源デバイスとして使用する場合は「●レジスタ初期設定例と動作開始手順例」にしたがって処理してください。

● レジスタ初期設定と動作開始手順例

USB ファンクションのレジスタ初期設定例を示します。

EP0C レジスタによる EP0 の設定 (パケットサイズなど)

EP1C ~ EP5C レジスタによる各 EP の EPEN, DIR, TYPE などを設定

UDCC レジスタの RST ビットのクリア

EP0IS, EP0OS, EP1S ~ EP5S レジスタの BFINI をクリア

UDCC レジスタの $\overline{\text{HCON}}$ ビットのクリア

● USB バスリセット

ホスト PC からデバイスにバスリセットがかかり USB デバイスコアが初期化されます。デバイスは以下の順序で処理が必要です (USB 接続後の最初のバスリセットでは処理の必要はありません)。

UDCC レジスタの RST で USB ファンクションを初期化

使用する送受信バッファ、関係する制御レジスタを再設定

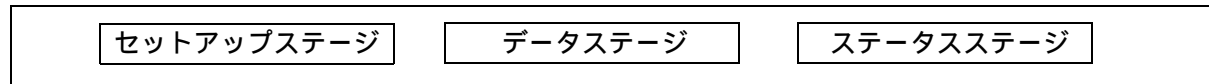
ファームの制御をエニュメレーション前に戻します。

● ディスクリプタの取得

ホスト PC からデバイスに要求がありデータをホストに通知します。

詳細には以下の 3 つのステージに分かれて通信されます。

図 13.4-2 通信ステージ



セットアップステージでは、ホスト PC から正常にパケットが受信されたか確認し、そのコマンドが何かをデコードします。また、次のデータステージで返すディスクリプタの情報を送信バッファに用意します。データステージでは単にホスト PC からデータが正常に送信されたかを確認します。ステータスステージでは、ホスト PC がデータなしパケットの転送をしますので終了処理をします。

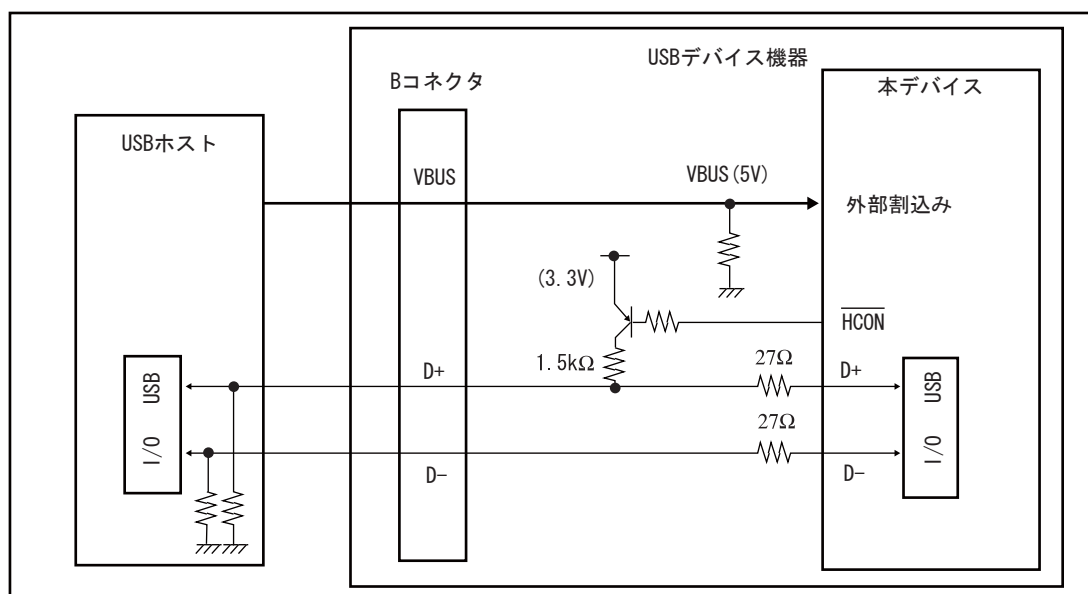
13.4.1 接続検出と切断検出

USB ホストとの接続検出と切断検出について説明します。

■ USB システムの接続例

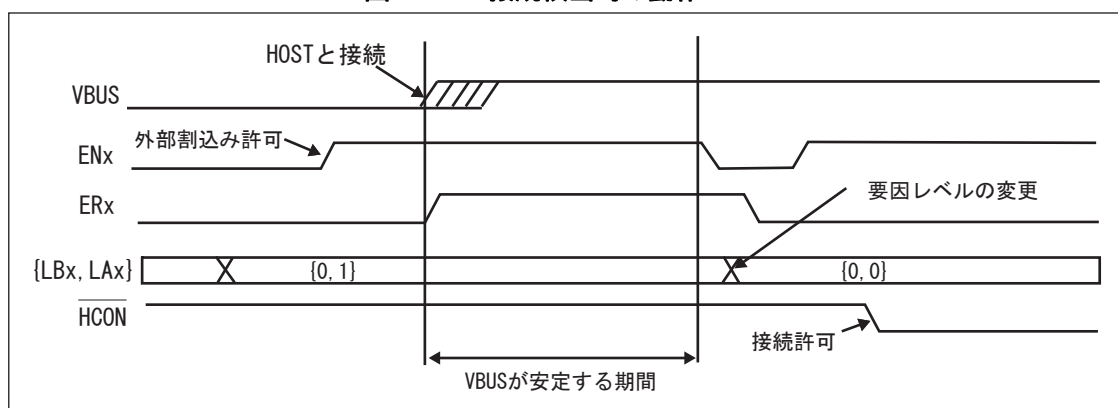
外部割込み端子を USB コネクタの VBUS ピンに接続し、プルダウン抵抗を接続することで USB ホストとの接続と切断を検出することができます。図 13.4-3 に、USB コネクタの D+, D-, VBUS との接続例を示します。

図 13.4-3 USB システム接続例



● 接続検出

図 13.4-4 接続検出時の動作



デバイスは以下の順序によりホスト PC との接続を認識し、処理します。

VBUS に接続した外部割込みの要因レベルを "H" レベル検出に設定し、割込み許可します。

外部割込み端子の "H" レベル検出で USB ホストが接続されたことを認識して VBUS が安定する期間を待ちます。

外部割込みをいったん禁止にします。外部割込み要因レベルを "L" レベル検出に設定変更し、割込み要因のクリアして再び外部割込みを許可します。

初期設定 (USB ファンクションレジスタを含むすべての初期化)「レジスタ初期設定例と動作開始手順例」参照してください。

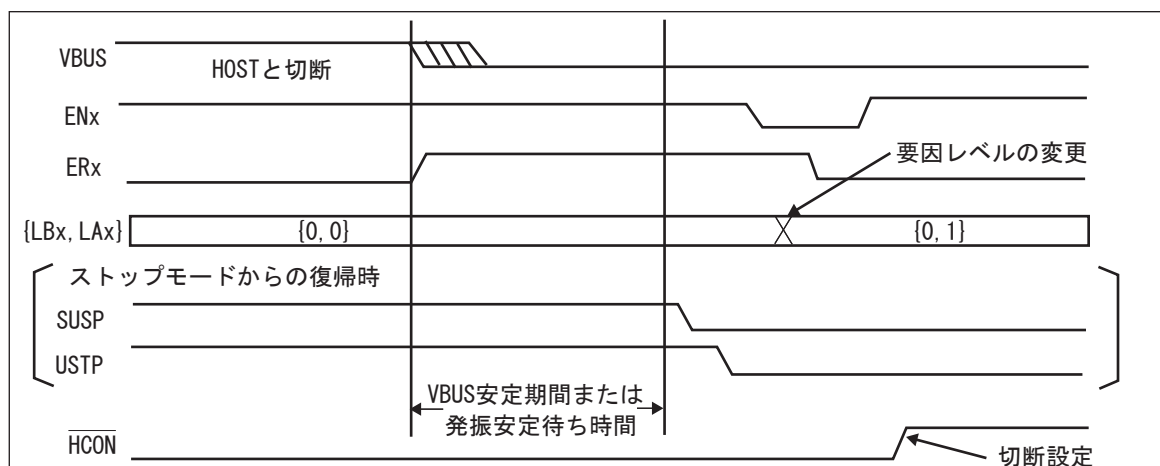
UDCC レジスタの $\overline{\text{HCON}}$ ビットをクリアし、D+ のプルアップ抵抗を接続します (プルアップ抵抗の制御をしていない場合でも $\overline{\text{HCON}}$ ビットもクリアします)。

< 注意事項 >

外部割込み端子にノイズフィルタを外付けした場合は上記の VBUS 安定期間をプログラムでとる必要はありません。

● 切断検出

図 13.4-5 切断検出時の動作



デバイスは以下の順序によりホスト PC との切断を認識し、処理します。

VBUS に接続の外部割込み端子の "L" レベル検出で USB ホストが切断されたことを認識します。

ストップモードからの復帰時：

発振安定待ち時間後に UDCC レジスタ SUSP、UDCC レジスタ USTP の順にクリアします。

ストップモード以外のとき：

VBUS が安定する期間を待ちます。

いったん外部割込みを禁止します。外部割込み要因レベルを "H" レベル検出に設定変更し、外部割込み要因のクリアして再び外部割込みを許可します。

UDCC レジスタの $\overline{\text{HCON}}$ ビットをセットして、D+ のプルアップ抵抗を切断します。(プルアップ抵抗の制御をしていない場合でも UDCC レジスタの $\overline{\text{HCON}}$ ビットをセットします。)

< 注意事項 >

外部割込み端子にノイズフィルタを外付けした場合は上記の VBUS 安定期間をプログラムでとる必要はありません。

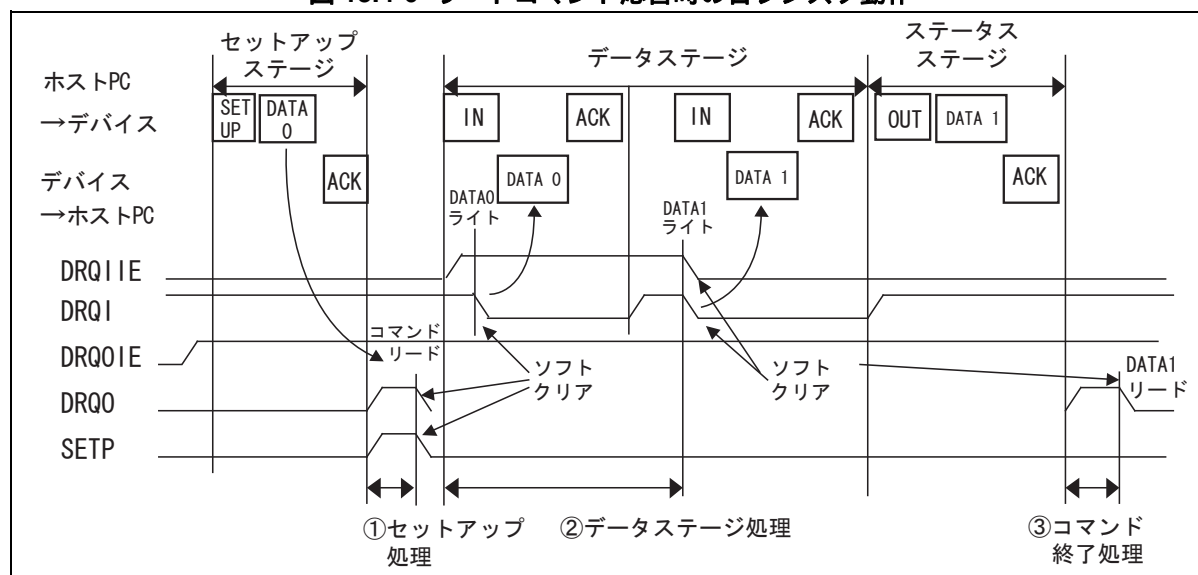
13.4.2 コマンド応答時の各レジスタ動作

基本となるレジスタの動作および制御より USB パケットの処理方法 (アーキテクチャ) を説明します。CPU 割込みによるファームウェアの処理はハンドシェイクごとに処理します。これは各パケットのステージ単位に処理することに等しくなります。

■ リードコマンド応答時の各レジスタ動作

GetDescriptor, SynchFrame, クラスベンダーコマンドの場合

図 13.4-6 リードコマンド応答時の各レジスタ動作



● セットアップ処理

セットアップステージを受信したときに DRQO がセットされます。DRQO がセットされた時点で CPU 割込みに入り SETP フラグを確認します。セットされていれば受信バッファにあるコマンドを必要な分読み出して (必ずしも 8 バイトすべて読み出す必要はありません) コマンドをデコードして各種設定処理し, SETP フラグ, DRQO 割込み要因をクリアし戻ります。

● データステージ処理

コマンドデコードの結果, データステージが IN 方向の場合, DRQIE を許可し (割込み要因 DRQI は初期値 "1" なので割込み許可をセットするだけ), CPU 割込みにて送信データを送信バッファに転送します。転送終了後, 割込み要因 DRQI をクリアしてから戻ります。

IN 方向のデータパケットが終了すると DRQI がセットされます。DRQI がセットされた時点で CPU 割込みに入り, 次のデータパケットに備え送信データを送信バッファに転送します。転送終了後, 割込み要因である DRQI をクリアして戻ります。

< 注意事項 >

この USB ファンクションは, USB 2.0 で新たに追加されたコマンドには対応していないため, Get Descriptor コマンドに対して USB リビジョンは 1.1 で応答してください。

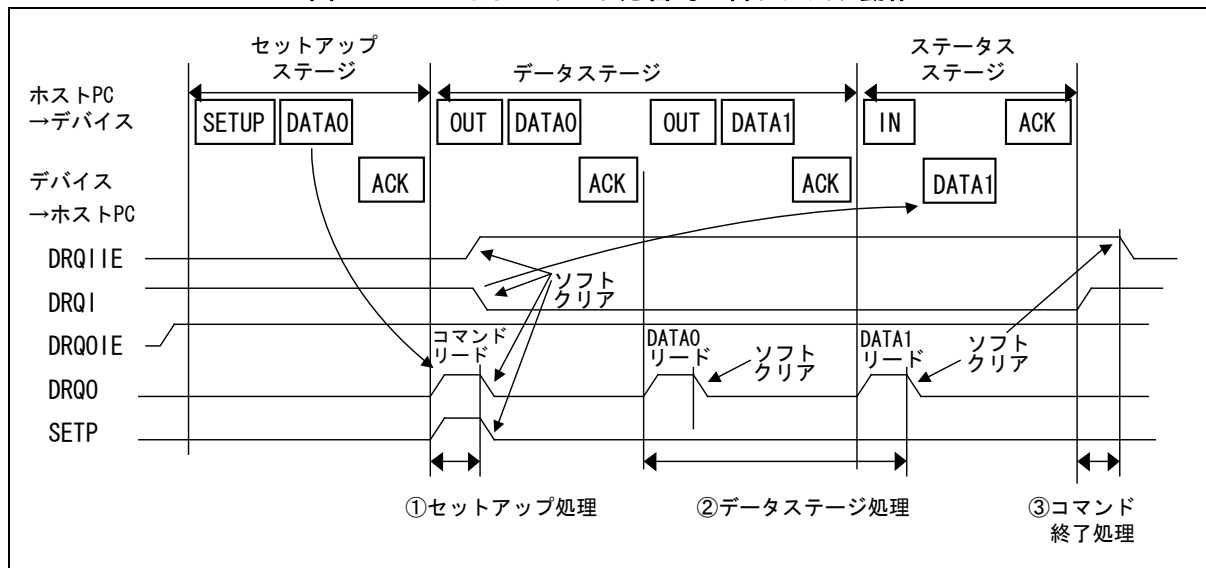
● コマンド終了処理

OUT 方向のステータスステージが終了すると DRQO がセットされます。DRQO がセットされた時点で CPU 割込みに入り、受信データ数 0 を確認し、次のセットアップステージに備え、割込み要因である DRQO をクリアして戻ります。

■ ライトコマンド応答時の各レジスタ動作

SetDescriptor, クラスベンダーコマンドの場合

図 13.4-7 ライトコマンド応答時の各レジスタ動作



● セットアップ処理

セットアップステージを受信したときに DRQO がセットされます。DRQO がセットされた時点で CPU 割込みに入り SETP フラグを確認します。セットされていれば受信バッファにあるコマンドを必要な分読み出して（必ずしも 8 バイトすべて読み出す必要はありません）デコードし各種設定処理をします。ステータスステージの 0 バイト応答に備え、送信バッファへデータをライトせずに、DRQI (割込み要因 DRQI は初期値 "1" のため) をクリアし、ステータスステージの正常終了確認用に DRQIE をセットしておきます。また SETP フラグ、DRQO 割込み要因をクリアし割込みから復帰します。

● データステージ処理

OUT 方向のデータステージが終了すると DRQO がセットされます。DRQO がセットされた時点で CPU 割込みに入ります。EP0 ステータスレジスタの SIZE を確認し、受信したデータ数分だけ DMA を起動するか、CPU リードにより受信バッファからデータを読み出します。その後、割込み要因である DRQO をクリアして割込みから復帰します。

● コマンド終了処理

IN 方向のステータスステージが終了すると DRQI がセットされます。DRQI がセットされた時点で CPU 割込みに入りステータスステージが正常終了したことを確認できます。その後、割込み許可である DRQIE をクリアして戻ります。

13.4.3 STALL 応答と解除

エンドポイント 0 の場合と、エンドポイント 1 ~ エンドポイント 5 の場合それぞれに対して STALL 応答と解除方法について説明します。

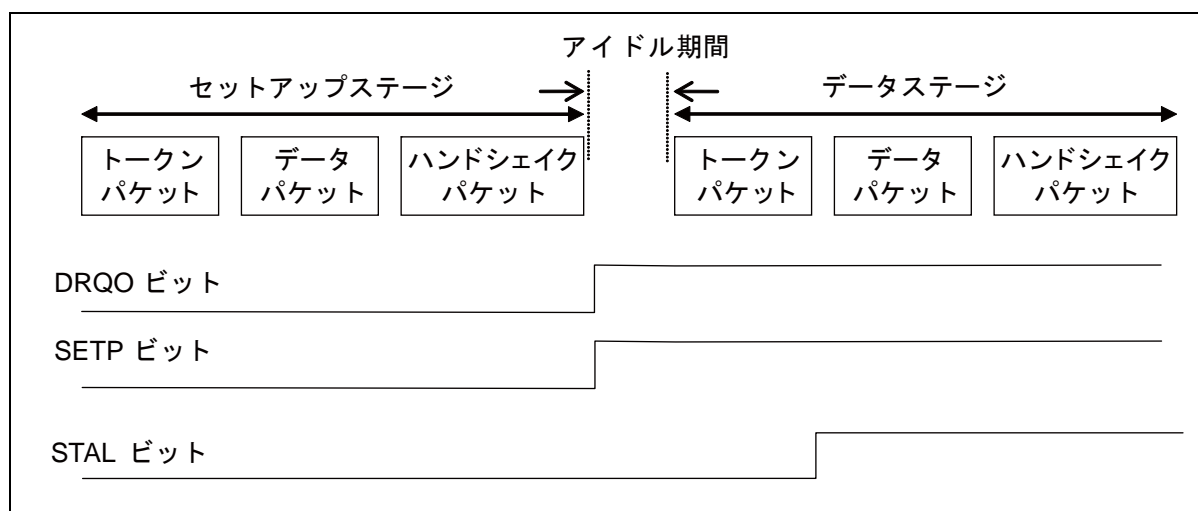
■ エンドポイント 0 の STALL 応答 / 解除について

エンドポイント 0 の STALL 応答 / 解除制御は、EP0 制御レジスタ (EP0C) の STAL ビットにて行います。

- STAL ビットのセットタイミング

STALL 応答は、コントロール転送のセットアップステージである事を示す SETP ビット =1 検出 (DRQO ビット =1 割込み) にてコマンドを解釈し、STALL 応答が必要な場合に STAL ビットをセットします (図 13.4-8 参照)。STAL ビットセット後に割込み要因 (DRQO ビット) をクリアしてください。

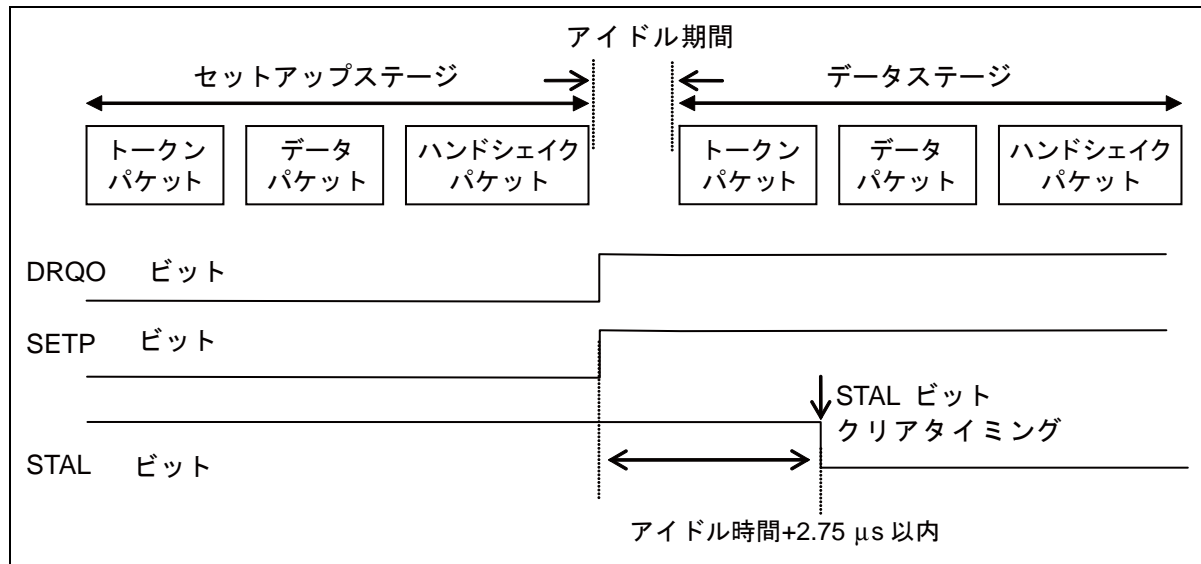
図 13.4-8 STAL ビットセットのタイミング



- STAL ビットのクリアタイミング

STALL 解除は、コントロール転送のセットアップステージであることを示す SETP ビット =1 検出 (DRQO ビット =1 割込み) にて STAL ビットクリアします (図 13.4-9 を参照してください)。

図 13.4-9 STAL ビットクリアのタイミング



STALL 応答解除 (STAL ビットクリア) は、SETP ビット =1 (DRQO ビット =1 割込み) を検出してから、次のデータステージのデータパケット送受信開始前に、STAL ビットクリアしてください。DRQO ビット =1 から、STAL ビットクリアまでの時間は下記のとおりです (転送速度: Full Speed 12Mbps 時)。下記時間内に STAL ビットがクリアされない場合、データステージのハンドシェイクパケットにて STAL 応答します。

DRQO ビット =1 検出から STAL ビットクリアまでの時間: アイドル時間 + 2.75μs 以内

アイドル期間が最小の 2 ビット転送時間である場合は約 2.9μs 以内となります。

STAL ビットクリアが、上記時間内で対応できない場合には、USB ホスト側のドライバソフトでアイドル時間を長くする等の対応を行ってください。

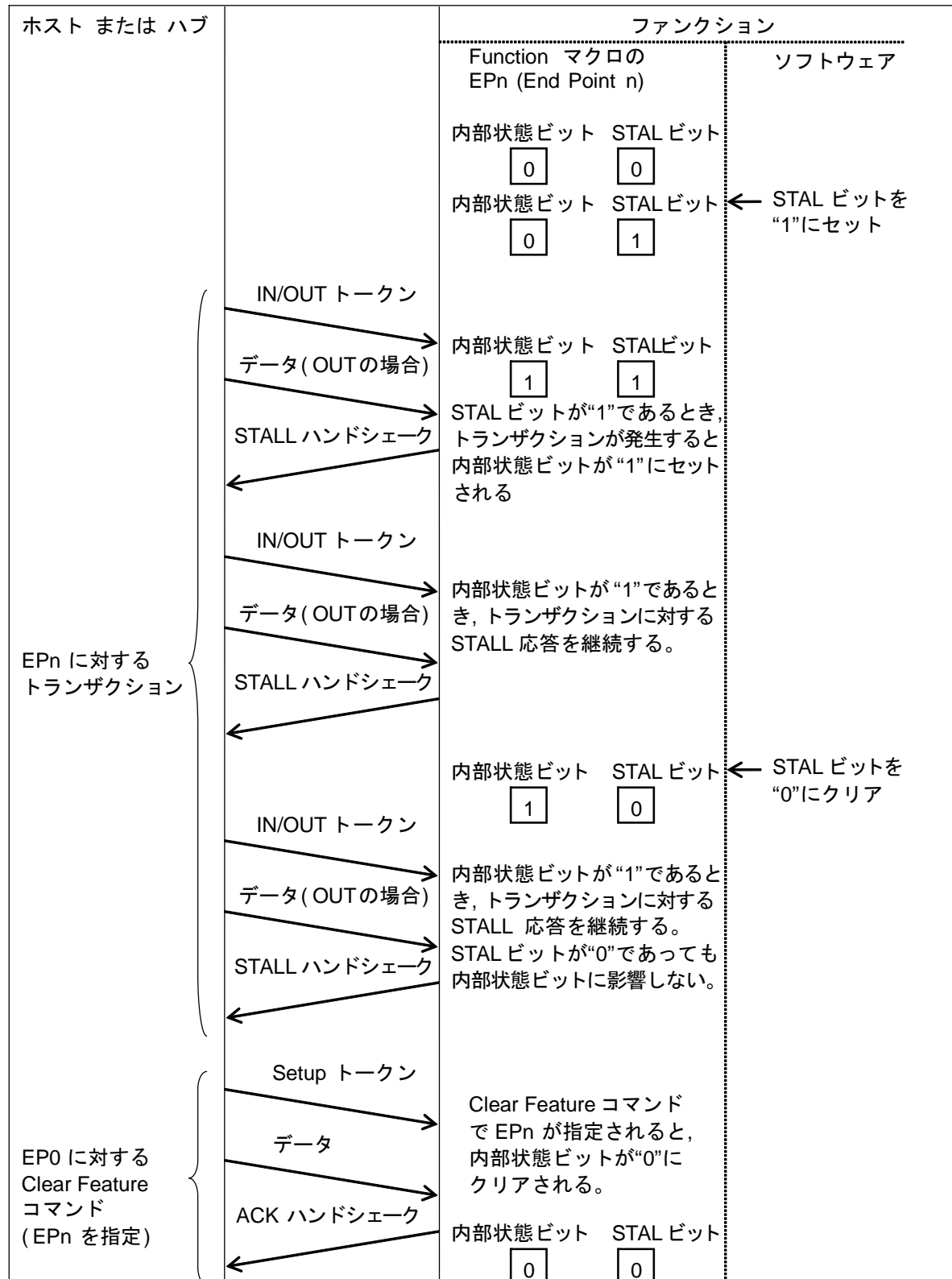
■ エンドポイント 1 ~ エンドポイント 5 の STALL 応答 / 解除について

エンドポイント 1 ~ エンドポイント 5 の STALL 応答 / 解除制御は、EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C) の STAL ビットと内部状態ビットで行われます。

• ソフト処理にて STALL 応答したい場合

ソフト処理で STALL 応答する場合の手順を図 13.4-10 に示します。STALL 応答する場合、該当するエンドポイントの STAL ビットをソフトでセットします。このとき、内部状態ビットは変化しません。次に、ホストから STAL ビットがセットされているエンドポイントに対してトランザクションが発生したとき、ハードが自動的に該当エンドポイントの内部状態ビットをセットし、ホストに対して STALL 応答します。一度、内部状態ビットがセットされた後は、STAL ビットをクリアしても、ホストから Clear Feature コマンドが発行されるまで、内部状態ビットはセットされたままとなり、STALL 応答を継続します。また、Clear Feature コマンドで内部状態ビットがクリアされても、STAL ビットがセットされている場合は、該当するエンドポイントに対するトランザクションが発生するたびに内部状態ビットがセットされるため STALL 応答を継続します。したがって、STALL 応答を解除するためには、STAL ビットをクリアし、さらに Clear Feature コマンドで内部状態ビットをクリアする必要があります。

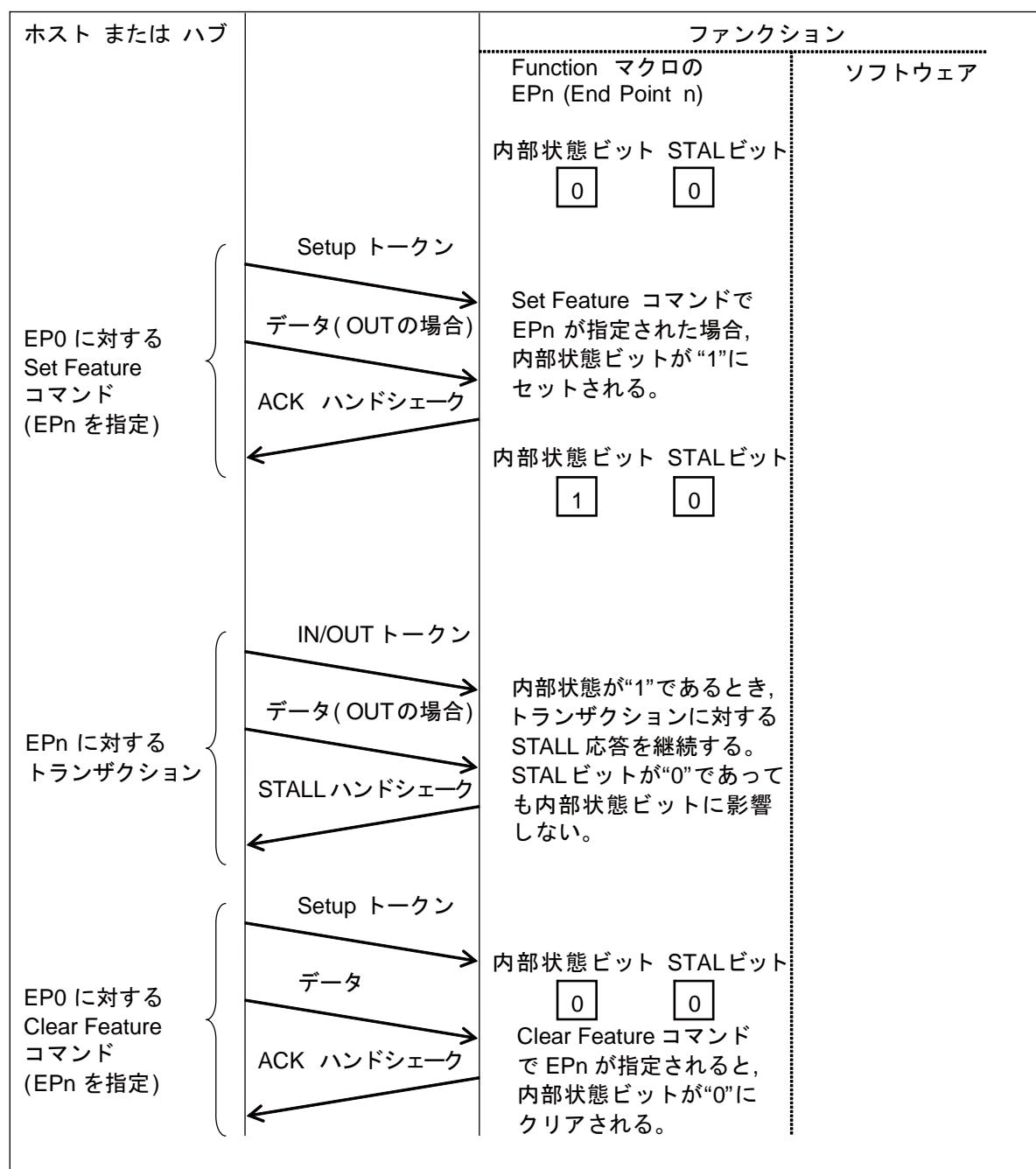
図 13.4-10 ソフト処理にて STALL 応答したい場合



・ハードが自動で STALL 応答する場合

ハードが自動で STALL 応答する場合の手順を図 13.4-11 に示します。Set Feature コマンドで STALL 応答設定された場合、STAL ビットに関係なく、ハードが自動的に該当エンドポイントの内部状態ビットをセットし、STALL 応答します。一度、内部状態ビットがセットされた後は、STAL ビットに関係なく、ホストから Clear Feature コマンドでクリアされるまで、内部状態ビットは保持されます。Clear Feature コマンドで該当ビットがクリアされた後は、STAL ビットを参照ようになります。したがって、STALL 応答を解除するには、Clear Feature コマンドで内部状態ビットをクリアする必要があります。

図 13.4-11 ハードが自動で STALL 応答する場合



13.4.4 サスペンド機能

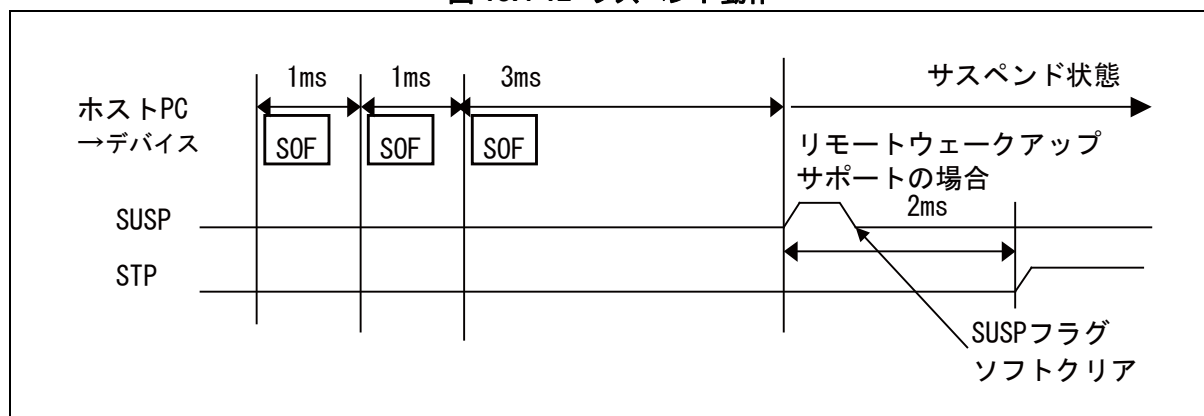
USB デバイスはバス電源の構成ではサスペンド状態において $500\ \mu\text{A}$ 以下に消費電力を落とす必要があります。ここではデバイスがサスペンド状態に移行し、STOP モードに入れるまでを説明します。

■ サスペンド処理

USB デバイスコアがサスペンド状態を検出した場合に UDCS レジスタの SUSP が有効にセットされます。

以下にサスペンド処理する例を示します。

図 13.4-12 サスペンド動作



● サスペンド処理

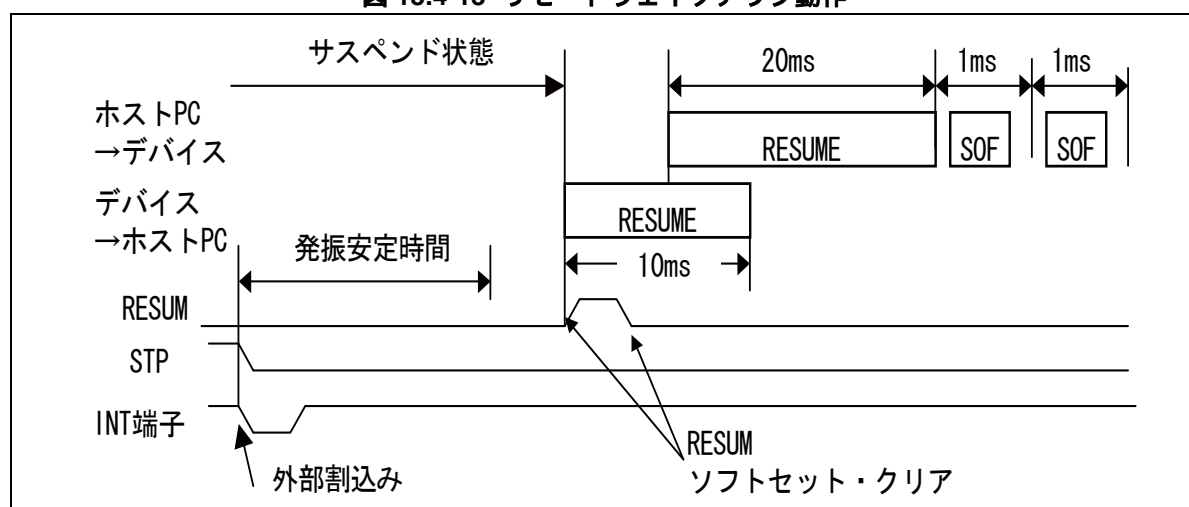
USB バス上に $3\ \text{ms}$ 以上動作がない場合、USB ファンクションはサスペンドを検出し、UDCS レジスタの SUSP 割込み要因がセットされます。リモートウェイクアップをサポートするデバイスの場合はここからさらに $2\ \text{ms}$ 待ち（これはこの時間リモートウェイクアップさせないためです）、ストップモードに設定します。

13.4.5 ウェイクアップ機能

- デバイスからのリモートウェイクアップ
- ホスト PC からのウェイクアップ

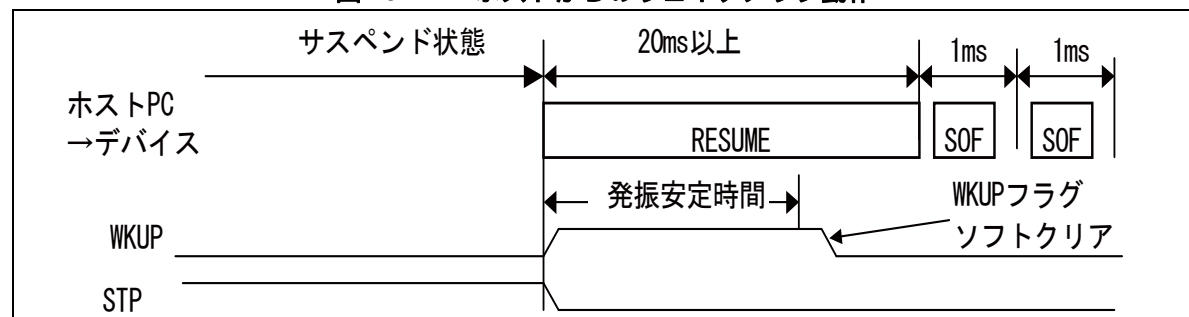
各々について説明します。

図 13.4-13 リモートウェイクアップ動作



UDCC レジスタの RESUM をクリアします。

図 13.4-14 ホストからのウェイクアップ動作



WKUP 要因による割込みに入り、割込み要因である UDCS の WKUP をクリアし割込みから復帰します。

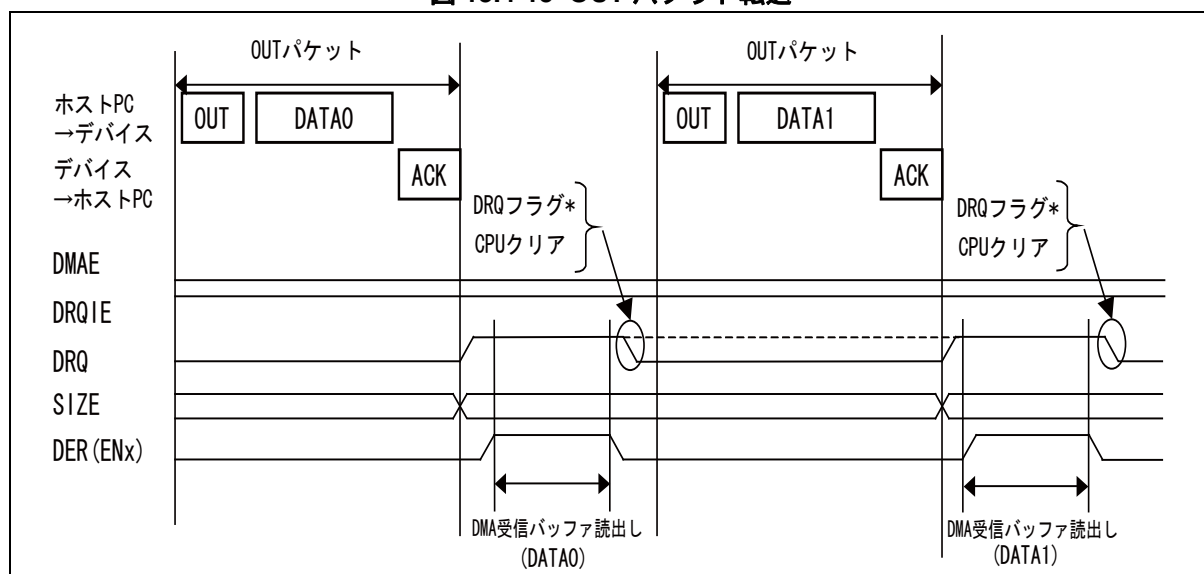
13.4.6 DMA 転送機能

USB ファンクションで通信するデータを送受信バッファと内蔵 RAM との間で DMA 転送することが可能です。DMA 転送には 1 パケット単位ごとに転送数を設定し転送するパケット転送モードと、設定した転送データ数分を 1 回の設定で転送するデータ数自動転送モードの 2 つが選択できます。各 DMA 転送モードについて説明します。

■ パケット転送モード

- 1 パケット単位ごとに転送数を DMA に設定し転送終了後、割り込み要因をクリアして転送するパケット転送モードです。転送モードは各エンドポイントに対するどのバッファへのアクセスも可能です。
- OUT 方向, IN 方向のそれぞれでバッファをアクセスするタイミングを次に示します。
- OUT 方向 (ホスト PC → デバイス) 転送

図 13.4-15 OUT パケット転送



OUT 方向転送ではデバイスでは以下の順序で処理が必要です。

DRQ フラグがセットされ、割り込み処理に入ったら転送データ数を確認します。

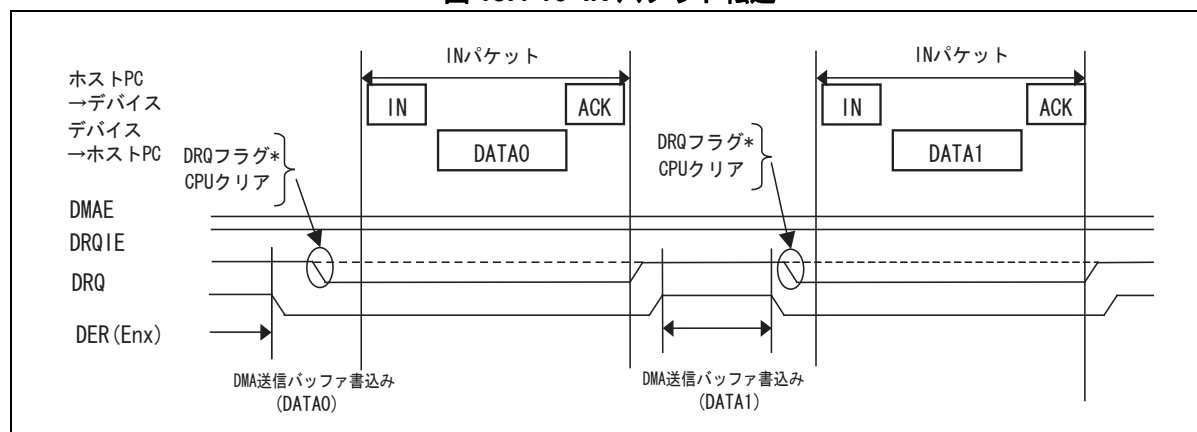
転送データ数を DMA のデータカウンタレジスタ DDCT に設定し DER レジスタで DMA を許可して転送開始します。

転送後, EP1S ~ EP5S レジスタの該当する DRQ フラグと μ DMAC の DSR レジスタの該当する割り込み要因フラグをクリアして割り込み処理から復帰します。

*: EP1 ~ EP5 はダブルバッファ構成となっており、アクセスしていないバッファが空の状態アクセスしているバッファが読み出されたときにのみクリアでき、アクセスしていない方のバッファに読み出されていないデータが残っていた場合は "0" をライトしてもクリアされません (破線状態)。連続して DRQ による割り込み処理に入ることになります。

● IN 方向 (ホスト PC デバイス) 転送

図 13.4-16 IN パケット転送



IN 方向転送ではデバイスは以下の順序で処理が必要です。

DRQ フラグがセットされ、割り込み処理に入ったら、次 IN パケットで転送する転送データ数を DMA のデータカウンタレジスタ DDCT に設定し DER レジスタで DMA を許可して転送開始します。

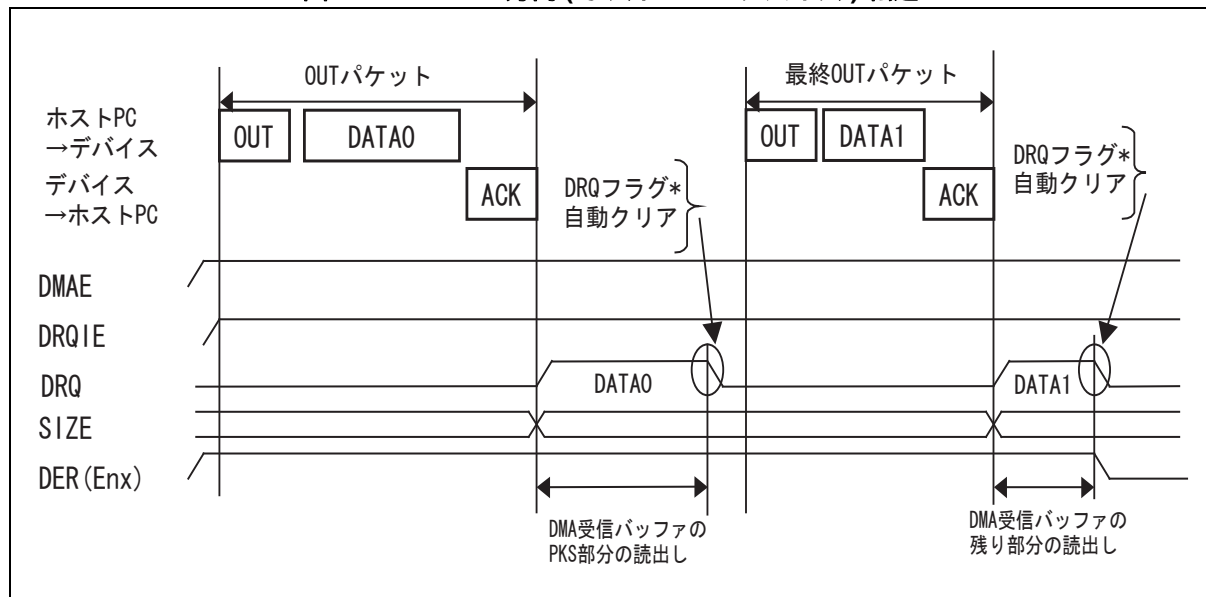
DMA 転送後、EP1S ~ EP5S レジスタの該当する DRQ フラグと μ DMAC の DSR レジスタの該当する割り込み要因フラグをクリアして割り込み処理から復帰します。

*: EP1 ~ EP5 はダブルバッファ構成となっており、アクセスしていないバッファに既にデータが書き込まれている状態で、アクセスしているバッファにデータが書き込まれたときにのみクリアでき、アクセスしていない方のバッファが空の場合は "0" をライトしてもクリアされません (破線状態)。連続して DRQ による割り込み処理に入ることになります。

■ データ数自動転送モード

DMA にあらかじめ転送する総データ数を設定し転送許可ビットもセットしておきます。DMAE が許可されていてホストからの転送後 DRQ がセットされると、EP1 ~ EP5 制御レジスタ (EPxC) レジスタの PKS 分のデータ数を転送した後に自動で割り込み要因をクリアします (DRQ フラグが実際にクリアされるかはダブルバッファが共に空になるか、フルになるかによります)。以後、ホストからの転送後に同様の処理をあらかじめ DMA に設定した転送データ数分まで繰り返し行います。その間 CPU による設定は一切必要なく 1 回の設定で転送する自動転送モードです。次の転送を行う場合は、ラストデータ転送後に CPU 割り込みに入るのでそこで μ DMAC の再設定を行い、DMA を許可して割り込み復帰します。データ数自動転送モードは DMAE=1 として使用するのでエンドポイント 1 から 5 に対するバッファアクセスのみ有効となります。OUT 方向、IN 方向のそれぞれでバッファをアクセスするタイミングを次に示します。

図 13.4-17 OUT 方向 (ホスト PC デバイス) 転送



デバイスは OUT 方向転送, IN 方向転送共に以下の順序で処理が必要です。

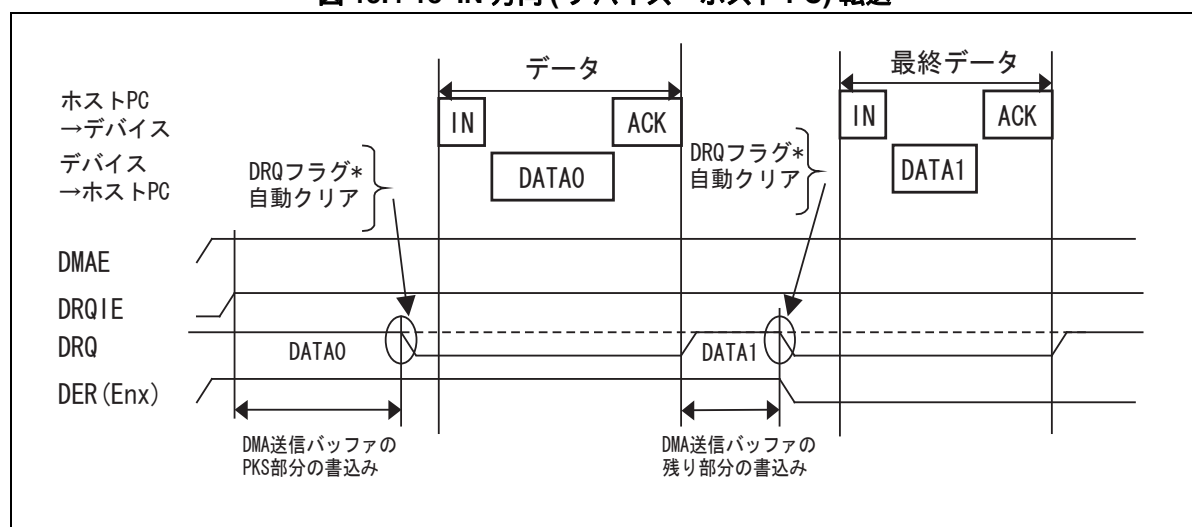
転送する総データ数を DMA のデータカウンタレジスタ DDCT に設定し, DER レジスタで DMA を許可します。

DMAE, DRQIE を許可設定します。

転送後, μ DMAC の DSR レジスタの該当する割込み要因による割込みで必要に応じて μ DMAC の再設定を行い, フラグをクリアして割込み処理から復帰します。

*: EP1 ~ EP5 はダブルバッファ構成となっており, アクセスしていないバッファが空の状態アクセスしているバッファが読み出されたときにのみクリアされ (自動リセット), アクセスしていない方のバッファに読み出されていないデータが残っていた場合はクリアされません。連続して DRQ による割込み処理に入ることになります。

図 13.4-18 IN 方向 (デバイス ホスト PC) 転送



デバイス は OUT 方向転送 , IN 方向転送共に以下の順序で処理が必要です。

転送する総データ数を DMA のデータカウンタレジスタ DDCT に設定し DER レジスタで DMA を許可します。

DMAE, DRQIE を許可設定します。

転送後 , μ DMAC の DSR レジスタの該当する割込み要因による割込みで必要に応じて μ DMAC の再設定を行い , フラグをクリアして割込み処理から復帰します。

*: EP1 ~ EP5 はダブルバッファ構成となっており , アクセスしていないバッファに既にデータが書き込まれている状態でアクセスしているバッファにデータが書き込まれたときにのみクリアされ , アクセスしていない方のバッファが空の場合はクリアされません。連続して DRQ による割込み処理に入ることになります。

13.4.7 NULL 転送機能

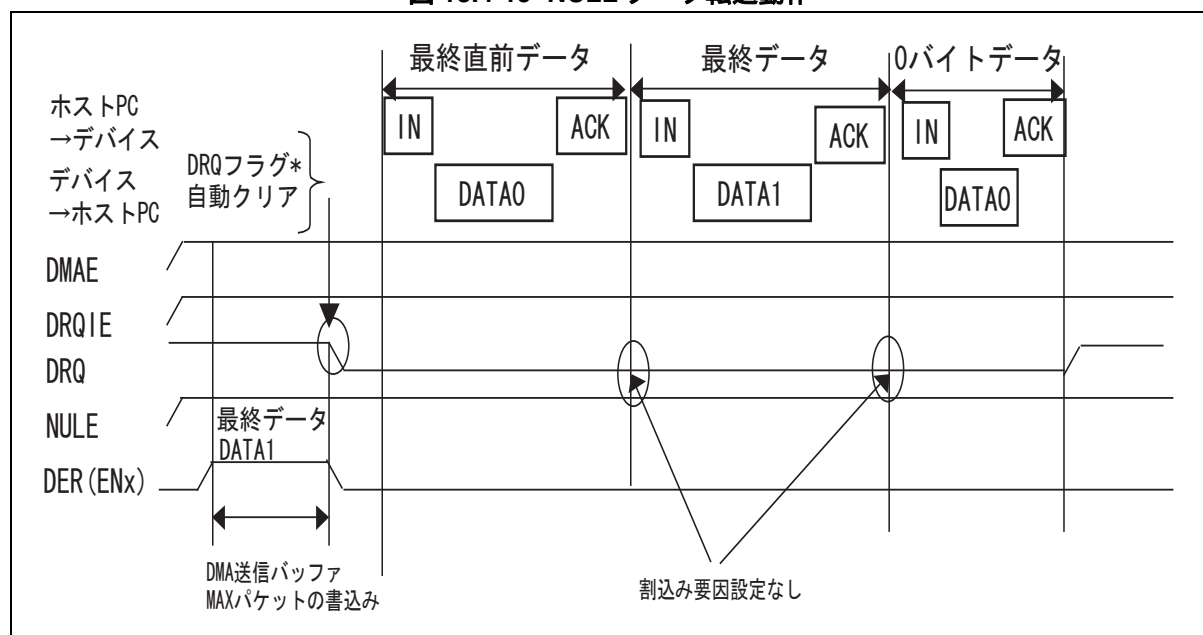
USB ファンクションから送信するデータがラストパケットで最大パケット数のとき、次パケットの転送で 0 バイトの転送を自動送信することが可能です。NULL 転送機能は DMAE を許可する必要があり、IN 転送時のみ有効な機能です。

■ NULL 転送モード

自動バッファ転送モードがセットされている状態 (DMAE=1) で、IN 方向のデータ転送要求がきたときに、最大パケット数の DMA 書込みが行われ、かつ最後のデータ書込みで DMA カウントデータ数が 0 になる場合、HOST から IN 方向のラストのデータ転送要求がきたときに 0 バイトのデータ転送を自動で設定し、次の IN 方向のデータ転送要求で 0 バイト送信するモードです。DMA でラストのデータがバッファに書き込まれた後、HOST から 0 バイトのデータが読み出されるまで DRQ の割込みフラグはセットされません。バッファをアクセスするタイミングを次に示します。

- IN 方向 (デバイス → ホスト PC) 転送のみ

図 13.4-19 NULL データ転送動作



デバイスは以下の処理が必要です。

DMAE, DRQIE, NULE を許可設定します。

第14章

USB ホスト

この章では、USB ホストの機能と動作について示します。

- 14.1 USB ホストの特長
- 14.2 USB ホストの制限事項
- 14.3 USB ホストのブロックダイアグラム
- 14.4 USB ホストのレジスタ
- 14.5 USB ホストの動作
- 14.6 USB ホストの各トークンフローチャート

14.1 USB ホストの特長

USB ホストは、必要最低限のホスト動作を実現し、PC が介在することなくデバイスとのデータ転送ができる機能です。

■ USB ホストの特長

USB ホストは、オリジナルの USB ホスト機能です。動作モードを切り替えることで、USB ファンクションとしても動作可能です。

USB ホストには以下の特長があります。

- Low Speed/Full Speed 転送の自動検出
- Low Speed/Full Speed 転送サポート
- デバイスの接続および切断の自動検出
- USB バスへのリセット送出機能サポート
- IN/OUT/SETUP/SOF トークンのサポート
- IN トークン時のハンドシェイクパケット自動送出 (STALL は除く)
- OUT トークン時のハンドシェイクパケット自動検出
- 最大パケット長 256 バイトまでサポート
- 各種エラー (CRC エラー / トグルエラー / タイムアウト) サポート
- ウェイクアップ機能サポート

14.2 USB ホストの制限事項

USB ホストの制限事項を示します。

■ USB ホストの制限事項

表 14.2-1 USB ホストの制限事項

		ホスト
ハブのサポート		*
転送	バルク転送	
	コントロール転送	
	インタラプト転送	
	アイソクロナス転送	×
転送スピード	Low Speed	
	Full Speed	
PRE パケットサポート		×
SOF パケットサポート		
エラー	CRC エラー	
	トグルエラー	
	タイムアウト	
	最大パケット < 受信データ	
デバイスの接続・切断検出		
転送スピード検出		

：サポートしている

×：サポートしていない

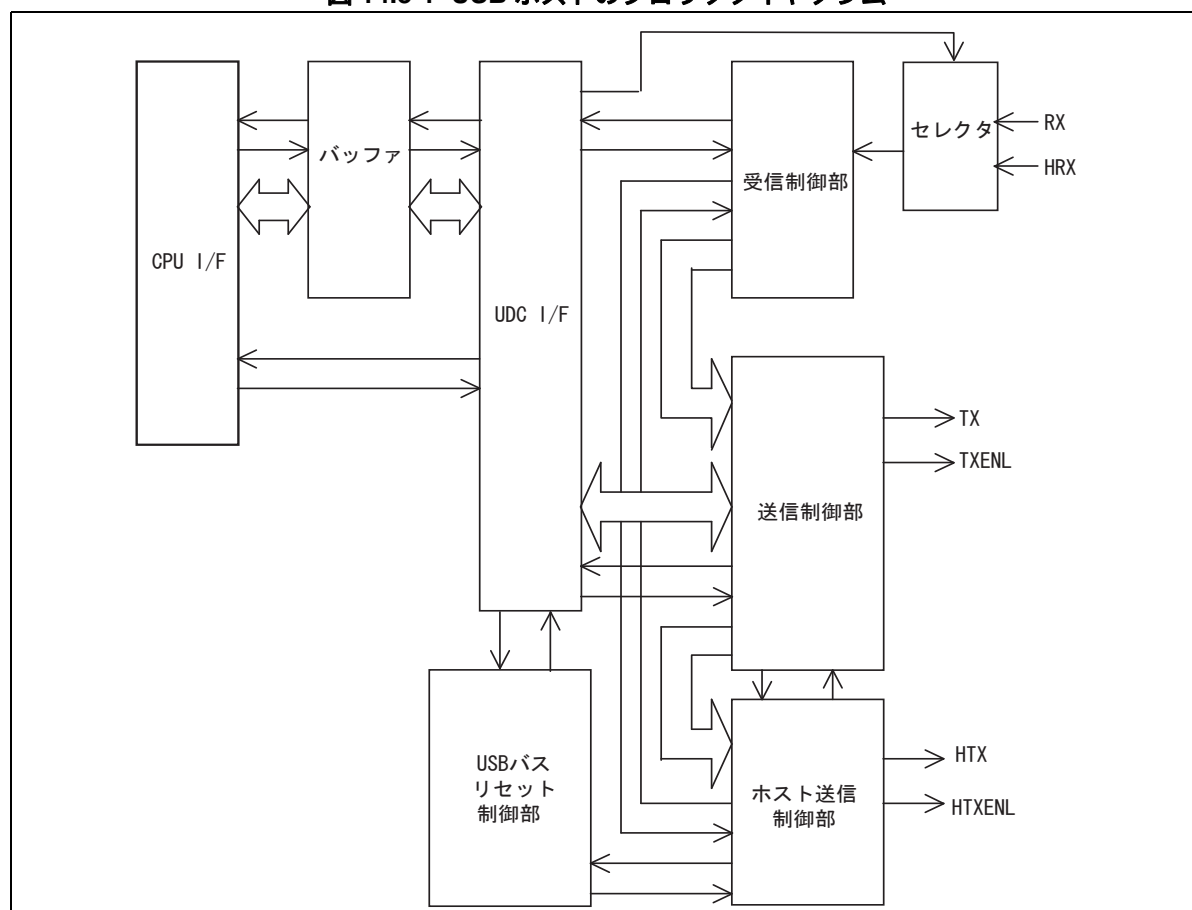
*：Full Speed のみ対応し，HUB は 1 段までのサポートです。

14.3 USB ホストのブロックダイアグラム

図 14.3-1 に、USB ホストのブロックダイアグラムを示します。

■ USB ホストのブロックダイアグラム

図 14.3-1 USB ホストのブロックダイアグラム



ブロックの説明

CPU I/F	: CPU とのインタフェース回路ブロック
バッファ	: バッファおよびバッファコントロール回路ブロック
UDC I/F	: USB とのインタフェース回路ブロック
USB バスリセット制御部	: USB バスリセット・接続制御ブロック
受信制御部	: 受信データのシリアル・パラレル変換回路および受信制御回路ブロック
送信制御部	: ファンクション時の送信データのシリアル・パラレル変換回路および送信制御回路ブロック
ホスト送信制御部	: ホスト時の送信データのシリアル・パラレル変換回路および送信制御回路ブロック
TX	: ファンクションの送信データ信号

RX	: ファクションの受信データ信号
TXENL	: ファクションの送受信方向信号
HTX	: ホストの送信データ信号
HRX	: ホストの受信データ信号
HTXENL	: ホストの送受信方向信号

14.4 USB ホストのレジスタ

USB ホストには、次の 10 種類のレジスタがあります。

- ホストコントロールレジスタ 0, 1(HCNT0/HCNT1)
- ホスト割込みレジスタ (HIRQ)
- ホストエラーステータスレジスタ (HERR)
- ホスト状態ステータスレジスタ (HSTATE)
- SOF 割込み FRAME 比較レジスタ (HFCOMP)
- リトライタイマ設定レジスタ (HRTIMER)
- ホストアドレスレジスタ (HADR)
- EOF 設定レジスタ (HEOF)
- FRAME 設定レジスタ (HFRAME)
- ホストトークンエンドポイントレジスタ (HTOKEN)

■ USB ホストのレジスタ

ホストコントロールレジスタ 0										
	bit	7	6	5	4	3	2	1	0	
アドレス : 0000C0 _H		RWKIRE	URIRE	CMPIRE	CNNIRE	DIRE	SOFIRE	URST	HOST	HCNT0
リード / ライト		(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値		(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

ホストコントロールレジスタ 1										
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000C1 _H		予約	予約	予約	予約	予約	SOFSTEP	CANCEL	RETRY	HCNT1
リード / ライト		(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値		(0)	(0)	(0)	(0)	(0)	(0)	(0)	(1)	

ホスト割込みレジスタ										
	bit	7	6	5	4	3	2	1	0	
アドレス : 0000C2 _H		TCAN	予約	RWKIRQ	URIRQ	CMPIRQ	CNNIRQ	DIRQ	SOFIRQ	HIRQ
リード / ライト		(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値		(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

ホストエラーステータスレジスタ										
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000C3 _H		LSTSOF	RERR	TOUT	CRC	TGERR	STUFF	HS		HERR
リード / ライト		(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)		
初期値		(0)	(0)	(0)	(0)	(0)	(0)	(11 _B)		

ホスト状態ステータスレジスタ										
	bit	7	6	5	4	3	2	1	0	
アドレス : 0000C4 _H		予約		ALIVE	CLKSEL	SOFBUSY	SUSP	TMODE	CSTAT	HSTATE
リード / ライト		(-)		(R/W)	(R/W)	(R/W)	(R/W)	(R)	(R)	
初期値		(X)		(0)	(1)	(0)	(0)	(1)	(0)	

(続く)

(続き)

SOF 割込み FRAME 比較レジスタ										
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000C5 _H		FRAMECOMP								HFCOMP
リード / ライト		(R/W)								
初期値		(00000000 _B)								
リトライタイマ設定レジスタ										
	bit	7	6	5	4	3	2	1	0	
アドレス : 0000C6 _H		RTIMER0								HRTIMER
リード / ライト		(R/W)								
初期値		(00000000 _B)								
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000C7 _H		RTIMER1								HRTIMER
リード / ライト		(R/W)								
初期値		(00000000 _B)								
	bit	7(23)	6(22)	5(21)	4(20)	3(19)	2(18)	1(17)	0(16)	
アドレス : 0000C8 _H		予約						RTIMER2		HRTIMER
リード / ライト		(-)						(R/W)		
初期値		(X)						(00 _B)		
ホストアドレスレジスタ										
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000C9 _H		予約	Address							HADR
リード / ライト		(-)	(R/W)							
初期値		(X)	(00000000 _B)							
EOF 設定レジスタ										
	bit	7	6	5	4	3	2	1	0	
アドレス : 0000CA _H		EOF0								HEOF
リード / ライト		(R/W)								
初期値		(00000000 _B)								
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000CB _H		予約	EOF1							HEOF
リード / ライト		(-)	(R/W)							
初期値		(X)	(0000000 _B)							
FRAME 設定レジスタ										
	bit	7	6	5	4	3	2	1	0	
アドレス : 0000CC _H		FRAME0								HFRAME
リード / ライト		(R/W)								
初期値		(00000000 _B)								
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000CD _H		予約						FRAME1		HFRAME
リード / ライト		(-)						(R/W)		
初期値		(X)						(000 _B)		

(続き)

ホストトークンエンドポイントレジスタ									
	bit	7	6	5	4	3	2	1	0
アドレス : 0000CE _H		TGGL		TKNEN			ENDPT		
リード / ライト		(R/W)		(R/W)			(R/W)		
初期値		(0)		(000 _B)			(0000 _B)		

14.4.1 ホストコントロールレジスタ 0, 1(HCNT0/HCNT1)

ホストコントロールレジスタ 0, 1(HCNT0/HCNT1) は、USB の動作モードおよび割込みの設定を指定します。

■ ホストコントロールレジスタ 0, 1(HCNT0/HCNT1)

図 14.4-1 ホストコントロールレジスタ 0, 1 (HCNT0/1) のビット構成

ホストコントロールレジスタ 0									
bit	7	6	5	4	3	2	1	0	
アドレス : 0000C0 _H	RWKIRE	URIRE	CMPIRE	CNNIRE	DIRE	SOFIRE	URST	HOST	HCNT0
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
UDCC RST ビット でのリセット可否	(X)	(X)	(X)	(X)	(X)	(X)	()	(X)	

ホストコントロールレジスタ 1									
bit	15	14	13	12	11	10	9	8	
アドレス : 0000C1 _H	予約	予約	予約	予約	予約	SOFSTEP	CANCEL	RETRY	HCNT1
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(1)	
UDCC RST ビット でのリセット可否	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

[bit15 ~ bit11] 予約

予約ビットです。

必ず "0" を設定してください。

[bit10] SOFSTEP (SOF 割込み条件選択)

SOF 実行ごとに SOF による割込みを発生するかどうかを設定します。ホストコントロールレジスタ 0(HCNT0) の SOFIRE ビットが "1" のとき有効となります。

"0" の時、SOF 割込み FRAME 比較レジスタ (HFCOMP) の設定により、割込みを発生し、"1" のときは無条件で SOF 実行時に割込みが発生します。ただし、最初の SOF トークン時、割込みは発生しません。UDC 制御レジスタ (UDCC) の RST ビットで初期化されません。

SOFSTEP	動作モード
0	HFCOMP の設定により割込み発生
1	割込み発生

[bit9] CANCEL (トークンキャンセル許可)

ホスト割込みレジスタ (HIRQ) の SOFIRQ ビットが "1" になるとき、一度もトークンが実行されずにトークンが待ち状態であるとき (EOF 領域でトークンを発行)、そのトークンを中止するかどうかを設定するビットです。UDC 制御レジスタ (UDCC) の RST ビットで初期化されません。

CANCEL	動作モード
0	トークン継続
1	トークン中止

[bit8] RETRY (リトライ許可)

NAK および CRC エラー等が発生した場合、リトライするかどうかを設定するビットです。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

RETRY	動作モード
0	リトライする
1	リトライしない

[bit7] RWKIRE (再起動割込み要求許可)

レジューム終了後、ホスト機能の動作が可能となったときに割込みを発生するかどうかを設定するビットです。ホストモード時、サスペンド状態にするにはホスト状態ステータスレジスタの SUSP ビットに "1" をライトしてください。ホストモード時のみ有効です。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

RWKIRE	動作モード
0	再起動後割込み禁止
1	再起動後割込み許可

[bit6] URIRE (USB バスリセット時割込み要求許可)

USB バスへのリセット動作が終了したときに割込みを発生するかどうかを設定するビットです。ホストモード時のみ有効です。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

URIRE	動作モード
0	USB バスリセット後割込み禁止
1	USB バスリセット後割込み許可

[bit5] CMPIRE (完了時割込み要求許可)

トークンが完了したときに割込みを発生するかどうかを設定します。ホストモード時のみ有効です。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

CMPIRE	動作モード
0	完了時割込み禁止
1	完了時割込み許可

[bit4] CNNIRE (接続時割込み要求許可)

デバイスの接続時に割込みを発生するかどうかを設定するビットです。ホストモード時のみ有効です。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

CNNIRE	動作モード
0	デバイス接続時割込み禁止
1	デバイス接続時割込み許可

[bit3] DIRE (切断時割込み要求許可)

デバイスの切断時に割込みを発生するかどうかを設定するビットです。ホストモード時のみ有効です。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

DIRE	動作モード
0	デバイス切断時割込み禁止
1	デバイス切断時割込み許可

[bit2] SOFIRE (SOF 割込み要求許可)

SOF 送信時に割込みを発生するかどうかを設定するビットです。ホストモード時のみ有効です。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

SOFIRE	動作モード
0	SOF 送信時に割込み禁止
1	SOF 送信時に割込み許可

[bit1] URST (USB バスリセット)

USB バスに対し、リセットを発生するかどうかを設定します。USB バスへのリセット動作中は "1" を示し、リセットが終了すると "0" になります。ホスト状態ステータスレジスタ (HSTATE) の SUSP ビットが "1" またはトークン実行中に "1" を設定することは禁止です。また、"1" の間、ホストコントロールレジスタ (HCNT0, HCNT1) の更新は禁止です。ホストモード時のみ有効です。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

URST	動作モード
0	USB バス状態保持
1	バスリセット

[bit0] HOST (ホストモード)

LSI がファンクションかホストかを設定します。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されませんので UDC 制御レジスタ (UDCC) の RST ビットが "1" のときに変更してください。

また , ファンクションモードからホストモードにする場合には UDC 制御レジスタ (UDCC) の $\overline{\text{HCON}}$ ビットを "1" にしてホスト PC または HUB と切断状態にしてください。ホストモードからファンクションモードにする場合 , ホストステータスレジスタ (HSTATE) の SOFBUSY ビットを "0" にし , ホストトークンエンドポイントレジスタ (HTOKEN) の TKNEN ビットが 000_B になっていることを確認して変更してください。

HOST	動作モード
0	ファンクションモード
1	ホストモード

14.4.2 ホスト割込みレジスタ (HIRQ)

ホスト割込みレジスタ (HIRQ) は、USB ホストの割込み要求フラグを示します。TCAN ビットを除いてホストコントロールレジスタ (HCNT0/HCNT1) の割込み許可ビットの設定により割込みを発生させることができます。

■ ホスト割込みレジスタ (HIRQ)

図 14.4-2 ホスト割込みレジスタ (HIRQ) のビット構成

ホスト割込みレジスタ								
	bit 7	6	5	4	3	2	1	0
アドレス : 0000C2 _H	TCAN	予約	RWKIRQ	URIRQ	CMPIRQ	CNNIRQ	DIRQ	SOFIRQ
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)
UDCC RST ビットでのリセット可否	()	()	()	()	()	(X)	(X)	()

[bit7] TCAN (トークンキャンセルフラグ)

ホスト割込みレジスタ (HIRQ) の SOFIRQ ビットが "1" になるとき、一度もトークンが実行されずにトークンが中止されたことを示します。SOF による割込みと組み合わせるために割込みは発生しません。"0" にするためには "0" を書き込んでください。

更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

TCAN	動作モード
0	トークン中止なし
1	トークン中止あり

[bit6] 予約

予約ビットです。

必ず "0" を設定してください。

[bit5] RWKIRQ (再起動割込み要求)

レジュームが終了したことを示します。"1" となった場合、"0" を書き込むと "0" となります。"1" を書き込んだ場合には、現在の状態を保持します。ホストコントロールレジスタ 0 (HCNT0) の RWKIRE ビットが "1" の場合、"1" になると割込みが発生し、"0" にクリアすると割込み信号もクリアされます。

更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

RWKIRQ	動作モード
0	再起動による割込み要求なし
1	再起動による割込み要求あり

[bit4] URIRQ (USB バス割込み要求)

USB バスへのリセットが終了したことを示します。"1" となった場合, "0" を書き込むと "0" となります。"1" を書き込んだ場合には, 現在の状態を保持します。ホストコントロールレジスタ 0(HCNT0) の URIRE ビットが "1" の場合, "1" になると割込みが発生し, "0" をクリアすると割込み信号もクリアされます。

更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

URIRQ	動作モード
0	USB バスリセットによる割込み要求なし
1	USB バスリセットによる割込み要求あり

[bit3] CMPIRQ (完了割込み要求)

トークンが完了したことを示します。ホスト割込みレジスタ (HIRQ) の TCAN ビットが "1" となる場合には, "1" にセットされません。"1" となった場合, "0" を書き込むと "0" となります。"1" を書き込んだ場合には, 現在の状態を保持します。ホストコントロールレジスタ 0(HCNT0) の CMPIRE ビットが "1" の場合, "1" になると割込みが発生し, "0" にクリアすると割込み信号もクリアされます。

更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

CMPIRQ	動作モード
0	トークン完了による割込み要求なし
1	トークン完了による割込み要求あり

[bit2] CNNIRQ (接続割込み要求)

デバイスの接続を検出したことを示します。"1" となった場合, "0" を書き込むと "0" となります。"1" を書き込んだ場合には, 現在の状態を保持します。ホストコントロールレジスタ 0(HCNT0) の CNNIRE ビットが "1" の場合, "1" になると割込みが発生し, "0" にクリアすると割込み信号もクリアされます。

UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

CNNIRQ	動作モード
0	デバイス接続検出による割込み要求なし
1	デバイス接続検出による割込み要求あり

[bit1] DIRQ (切断割込み要求)

デバイスの切断を検出したことを示します。"1" となった場合, "0" を書き込むと "0" となります。"1" をライトした場合には, 現在の状態を保持します。ホストコントロールレジスタ 0(HCNT0) の DIRE ビットが "1" の場合, "1" になると割込みが発生し, "0" にクリアすると割込み信号もクリアされます。

UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

DIRQ	動作モード
0	デバイス切断検出による割込み要求なし
1	デバイス切断検出による割込み要求あり

[bit0] SOFIRQ (SOF 割込み要求)

SOF トークンを開始したかを示します。"1" となった場合, "0" をライトすると "0" となります。"1" を書き込んだ場合には, 現在の状態を保持します。ホストコントロールレジスタ 0(HCNT0) の SOFIRE ビットが "1" の場合, "1" になると割込みが発生し, "0" にクリアすると割込み信号もクリアされます。

更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

SOFIRQ	動作モード
0	SOF トークン開始による割込み要求なし
1	SOF トークン開始による割込み要求あり

14.4.3 ホストエラーステータスレジスタ (HERR)

ホストエラーステータスレジスタ (HERR) は、ホストモード時のデータ送信および受信中にエラーが発生したかどうかを示すレジスタです。

■ ホストエラーステータスレジスタ (HERR)

図 14.4-3 ホストエラーステータスレジスタ (HERR) のビット構成

ホストエラーステータスレジスタ							
	bit 15	14	13	12	11	10	9 8
アドレス : 0000C3 _H	LSTSOF	RERR	TOUT	CRC	TGERR	STUFF	HS
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(11 _B)
UDCC RST ビット でのリセット可否	()	()	()	()	()	()	()

[bit15] LSTSOF (SOF 実行エラー)

ホストモードで SOF トークンを実行しようとしたとき、他のトークンを実行していて SOF トークンが実行できなかったことを示します。"1" をクリアするには "0" を書き込んでください。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

LSTSOF	動作モード
0	SOF 実行
1	SOF 実行エラー

[bit14] RERR (受信エラー)

ホストモード時、設定された最大パケット数より多いデータを受信したかどうかを示します。受信エラーが発生した場合には、TOUT も "1" にセットされます。

"1" をクリアするには "0" を書き込んでください。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

RERR	動作モード
0	受信エラーなし
1	最大パケット受信エラー

[bit13] TOUT (タイムアウト)

ホストモード時、トークンに対し、所定時間内に応答がないとき、タイムアウトを検出します。そのタイムアウトが発生したかどうかを示します。"1" をクリアするには "0" を書き込んでください。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

TOUT	動作モード
0	タイムアウトなし
1	タイムアウトあり

[bit12] CRC (CRC エラー)

ホストモード時, CRC エラーが発生したかどうかを示します。CRC エラーが発生した場合には, TOUT も "1" にセットされます。

"1" をクリアするには "0" を書き込んでください。

更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

CRC	動作モード
0	CRC エラーなし
1	CRC エラーあり

[bit11] TGERR (トグルエラー)

ホストモード時, トグルエラーが発生したかどうかを示します。"1" をクリアするには "0" を書き込んでください。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

TGERR	動作モード
0	トグルエラーなし
1	トグルエラーあり

[bit10] STUFF (スタUFFイングエラー)

ホストモード時, スタッフィングエラーが発生したかどうかを示します。スタッフィングエラーが発生した場合には, TOUT も "1" にセットされます。

"1" をクリアするには "0" を書き込んでください。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" にしてから行ってください。

[bit9, bit8] HS (ハンドシェイクステータス)

ホストモード時, 送受信のハンドシェイク状態を示します。エラー等でハンドシェイクが発生しない場合および SOF トークンを終了した場合には NULL を示します。送受信終了時に更新します。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

HS		ハンドシェイク
bit9	bit8	
0	0	ACK
0	1	NAK
1	0	STALL
1	1	NULL

14.4.4 ホスト状態ステータスレジスタ (HSTATE)

ホスト状態ステータスレジスタ (HSTATE) は、デバイスの接続、転送モード等を USB 回路の状態を示すレジスタです。CLKSEL ビットは、ファンクションモードでも有効ですので注意してください。

■ ホスト状態ステータスレジスタ (HSTATE)

図 14.4-4 ホスト状態ステータスレジスタ (HSTATE) のビット構成

ホスト状態ステータスレジスタ									HSTATE
	bit 7	6	5	4	3	2	1	0	
アドレス : 0000C4 _H	予約	予約	ALIVE	CLKSEL	SOFBUSY	SUSP	TMODE	CSTAT	
リード/ライト	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R)	(R)	
初期値	(X)	(X)	(0)	(1)	(0)	(0)	(1)	(0)	
UDCC RST ビット でのリセット可否	(-)	(-)	(X)	(X)	()	()	(X)	(X)	

[bit7, bit6] 予約

予約ビットです。リードは不定です。ライトは動作に影響しません。

[bit5] ALIVE (Keep-Alive 機能設定)

Low Speed 時、Keep-Alive 機能を設定するビットです。ホスト状態ステータスレジスタ (HSTATE) の CLKSEL ビットが "0" のとき、"1" に設定すると SOF の代わりに SE0 を出力します。ホスト状態レジスタの CLKSEL ビットが "0" のとき有効で、CLKSEL ビットが "1" の場合には ALIVE ビットの設定に関係なく SOF を出力します。

ALIVE	動作モード
0	SOF 出力
1	SE0 出力 (Keep-Alive)

[bit4] CLKSEL (クロック選択)

USB の動作クロックを選択します。Full Speed 時には "1", Low Speed 時には "0" を設定しなければなりません。

ファンクションモード、ホストモードともに有効で、クロックの切換えは UDC 制御レジスタ (UDCC) の RST ビットが "1" のときに実施してください。ファンクションモード時の "0" 設定は禁止です。

CLKSEL	動作モード
0	Low Speed 用クロック
1	Full Speed 用クロック

[bit3] SOFBUSY (SOF タイマ動作)

ホストモード時、SOF タイマが動作中かどうかを示します。"0" を書き込むと SOF の送出は停止します。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

SOFBUSY	動作モード
0	SOF タイマ停止
1	SOF タイマ動作中

[bit2] SUSP (サスペンド)

ホストモード時、サスペンド状態を設定するビットです。"1" を書き込むとサスペンド状態にします。"1" の状態から "0" を書き込んだ場合、または USB バスが k-state 状態に変化するとサスペンド状態を解除し、ホスト割込みレジスタ (HIRQ) の RWKIRQ ビットが "1" となります。USB が動作中 (USB バスへのリセット、データの送受信)、"1" を設定することは禁止です。

ホストモード時、サスペンド状態でも USB 用クロックを停止することは禁止です。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行い、ファンクションモードで "1" をセットすることは禁止です。また、ホストモード時からファンクションモードに変更する時に "1" の場合は "0" を書き込みサスペンド状態から抜けた状態で変更してください。

Table 14.4-1 サスペンドの設定

SUSP	動作モード
"1" 書込み	サスペンド
"1" 状態で "0" 書込み	レジューム
その他	状態保持

[bit1] TMODE (転送モード)

ホストモード時の転送モードを示します。書き込んだ時は "1" を書き込んでください。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。CPU クロックは 24 MHz で使用してください。

TMODE	動作モード
0	Low Speed
1	Full Speed

[bit0] CSTAT (接続状態)

デバイスが接続されているかを示します。ホスト用端子が対象となります。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。

CSTAT	動作モード
0	デバイス切断
1	デバイス接続

14.4.5 SOF 割込み FRAME 比較レジスタ (HFCOMP)

SOF 割込み FRAME 比較レジスタ (HFCOMP) は、SOF トークン時の FRAME Number の下位 8 ビットと比較するデータを設定するレジスタです。HFCOMP レジスタと FRAME Number の下位 8 ビットを比較し、ホストコントロールレジスタ 0 (HCNT0) の SOFIRE ビットが "1" で比較結果が一致した場合には、SOF 送信開始時にホスト割込みレジスタ (HIRQ) の SOFIRQ ビットを "1" にして割込みが発生します。

■ SOF 割込み FRAME 比較レジスタ (HFCOMP)

図 14.4-5 SOF 割込み FRAME 比較レジスタ (HFCOMP) のビット構成

SOF 割込み FRAME 比較レジスタ								
bit	15	14	13	12	11	10	9	8
アドレス : 0000C5 _H	FRAMECOMP							HFCOMP
リード/ライト	(R/W)							
初期値	(00000000 _B)							
UDCC RST ビット	(X)							
でのリセット可否								

[bit15 ~ bit8] FRAMECOMP

Frame Number の下位 8 ビットと比較するデータを設定します。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

14.4.6 リトライタイマ設定レジスタ (HRTIMER)

リトライタイマ設定レジスタ (HRTIMER) は、トークンのリトライ時間を設定するレジスタです。

■ リトライタイマ設定レジスタ (HRTIMER)

図 14.4-6 リトライタイマ設定レジスタ (HRTIMER) のビット構成

リトライタイマ設定レジスタ								
bit	7	6	5	4	3	2	1	0
アドレス : 0000C6 _H	RTIMER0							HRTIMER
リード/ライト	(R/W)							
初期値	(00000000 _B)							
UDCC RST ビット でのリセット可否	(X)							
bit	15	14	13	12	11	10	9	8
アドレス : 0000C7 _H	RTIMER1							HRTIMER
リード/ライト	(R/W)							
初期値	(00000000 _B)							
UDCC RST ビット でのリセット可否	(X)							
bit	7(23)	6(22)	5(21)	4(20)	3(19)	2(18)	1(17)	0(16)
アドレス : 0000C8 _H	予約						RTIMER2	HRTIMER
リード/ライト	(-)						(R/W)	
初期値	(X)						(00 _B)	
UDCC RST ビット でのリセット可否	(-)						(X)	

[bit23 ~ bit18] 予約

予約ビットです。

読出しは不定です。書込みは動作に影響しません。

[bit17 ~ bit0] RTIMER0, HRTIMER1, HRTIMER2

トークンをリトライする時間を設定します。ホストコントロールレジスタ (HCNT1) の RETRY ビットが 1 のとき、トークンの実行が開始するとリトライタイマが起動され、1 ビットの転送クロック (Full Speed の場合、12 MHz) によりタイマ値が -1 されます。リトライタイマが "0" になったときにトークンのリトライは実施しません。

トークンのリトライが EOF 領域で発生した場合には SOF の実行が終了するまでリトライタイマは停止します。SOF の実行終了後、停止したタイマ値から -1 します。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

14.4.7 ホストアドレスレジスタ (HADR)

ホストアドレスレジスタ (HADR) は、トークンを送信する際のアドレスフィールドに使用されるレジスタです。

■ ホストアドレスレジスタ (HADR)

図 14.4-7 ホストアドレスレジスタ (HADR) のビット構成

ホストアドレスレジスタ										
	bit	15	14	13	12	11	10	9	8	
アドレス : 0000C9 _H	予約	Address								HADR
リード / ライト	(-)	(R/W)								
初期値	(X)	(00000000 _B)								
UDCC RST ビット でのリセット可否	(-)	(X)								

[bit15] 予約

予約ビットです。読出しは不定です。書込みは動作に影響しません。

[bit14 ~ bit8] Address (アドレス)

トークンのアドレスを設定します。UDC 制御レジスタ (UDCC) の RST ビットにより初期化されません。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

14.4.10 ホストトークンエンドポイントレジスタ (HTOKEN)

ホストトークンエンドポイントレジスタ (HTOKEN) は、トグル、エンドポイント、トークンを設定するレジスタです。

■ ホストトークンエンドポイントレジスタ (HTOKEN)

図 14.4-10 ホストトークンエンドポイントレジスタ (HTOKEN) のビット構成

ホストトークンエンドポイントレジスタ							
	bit 7	6	5	4	3	2	1 0
アドレス : 0000CE _H	TGGL		TKNEN		ENDPT		
リード / ライト	(R/W)		(R/W)		(R/W)		
初期値	(0)		(000 _B)		(0000 _B)		
UDCC RST ビット でのリセット可否	()		()		()		

[bit7] TGGL: トグル

データのトグルを設定します。送信時は TGGL ビットに従ってトグルデータを送出し、受信時は受信したトグルデータと TGGL ビットが示すトグルデータを比較し、エラー検出に使用します。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" で、TKNEN ビットが 000_B のときに設定してから行ってください。

TGGL	動作モード
0	データ 0
1	データ 1

[bit6 ~ bit4] TKNEN (トークン許可)

設定に従ってトークンを送出します。動作終了後 TKNEN = 000_B になり、ホスト割込みレジスタ (HIRQ) の CMPIRQ ビットを "1" にします。そのとき、ホストコントロールレジスタ 0 (HCNT0) の CMPIRE ビットが "1" であれば、割込みが発生します。SOF トークン時、TGGL ビット、ENDPT ビットは無視します。TKNEN ビットは UDC 制御レジスタ (UDCC) の RST ビットを "0" にし、ホストモードにしてから書込みを行ってください。また、トークンによる割込みが発生して再度トークンを発行する場合には USB 転送クロック (Full Speed 時 : 12 MHz, Low Speed 時 : 1.5 MHz) で 3 サイクル以上待って TKNEN ビットに書き込んでください。切断状態 (HSTATE の CSTAT=0) では、TKNEN ビットに書き込んでもトークンは実行されませんので注意してください。

表 14.4-2 トークン設定

TKNEN			動作モード
bit6	bit5	bit4	
0	0	0	送出しない
0	0	1	SETUP を送出
0	1	0	IN を送出
0	1	1	OUT を送出
1	0	0	SOF を送出
1	0	1	予約 (設定禁止)
1	1	0	予約 (設定禁止)
1	1	1	予約 (設定禁止)

< 注意事項 >

PRE パケットについてはサポートしていません。

ホスト状態ステータスレジスタ (HSTATE) の SOFBUSY ビットが "1" のときに TKNEN = 100_B の設定は禁止です。

[bit3 ~ bit0] ENDPT (エンドポイント)

デバイスへの送受信するエンドポイントを設定します。更新は UDC 制御レジスタ (UDCC) の RST ビットを "0" に設定してから行ってください。

14.5 USB ホストの動作

USB ホストの動作について説明します。

■ USB ホストの動作

- デバイスの接続

外部の USB 装置が接続されたことをソフトウェアで検出します。

- USB バスのリセット

USB バスをリセットします。

- トークンパケット

ホストモード時, 3 種類のトークンを選択できます。

- データパケット

データパケットの送信, 受信を行います。

- ハンドシェークパケット

ハンドシェークパケットにより, 送受信相手に状態を通知します。

- リトライ機能

パケット終了時, エラー等が発生した場合, リトライし続けます。

- SOF 割込み

割込みを発生します。

- エラーステータス

各種エラーを表示します。

- パケット終了

パケット終了すると割込みを発生します。

- サスペンド・レジューム

USB 回路をサスペンド・レジューム状態にします。

- デバイスの切断

ホスト用端子の状態検出により切断されたことの判断をします。

14.5.1 デバイスの接続

外部の USB 装置が接続されたことをソフトウェアで検出する方法を示します。

■ ホスト機能の設定

USB 装置のホストとして動作させるためには、ホストコントロールレジスタ 0(HCNT0) の HOST ビットを "1" に設定します。

■ 外部 USB 装置が非接続の状態、接続の状態

外部 USB 装置が接続されていないときはプルダウン抵抗によりホスト用端子 D+, D- ともに "L" になっています。このとき、ホスト状態ステータスレジスタ (HSTATE) の CSTAT ビットは "0", TMODE ビットは不定となります。外部 USB 装置が接続されると、ホスト状態ステータスレジスタ (HSTATE) の CSTAT ビットは "1" になります。

■ 外部 USB 装置の接続検出

外部 USB 装置が接続されたことを検出するためには、ホストコントロールレジスタ 0 (HCNT0) の CNNIRE ビットを "1" に設定します。ホスト割込みレジスタ (HIRQ) の CNNIRQ ビットが "1" となり、デバイス接続割込みが発生します。本割込みをクリアする場合には、ホスト割込みレジスタ (HIRQ) の CNNIRQ ビットに "0" を書き込んでください。割込みではなくポーリングでデバイスの接続を検出する場合は、ホストコントロールレジスタ 0(HCNT0)のCNNIRE ビットを "0" にし、ホスト割込みレジスタ (HIRQ) の CNNIRQ が "1" になることを確認するようにプログラムを作成してください。

■ 相手先 USB 装置の転送速度の取得とクロック選択

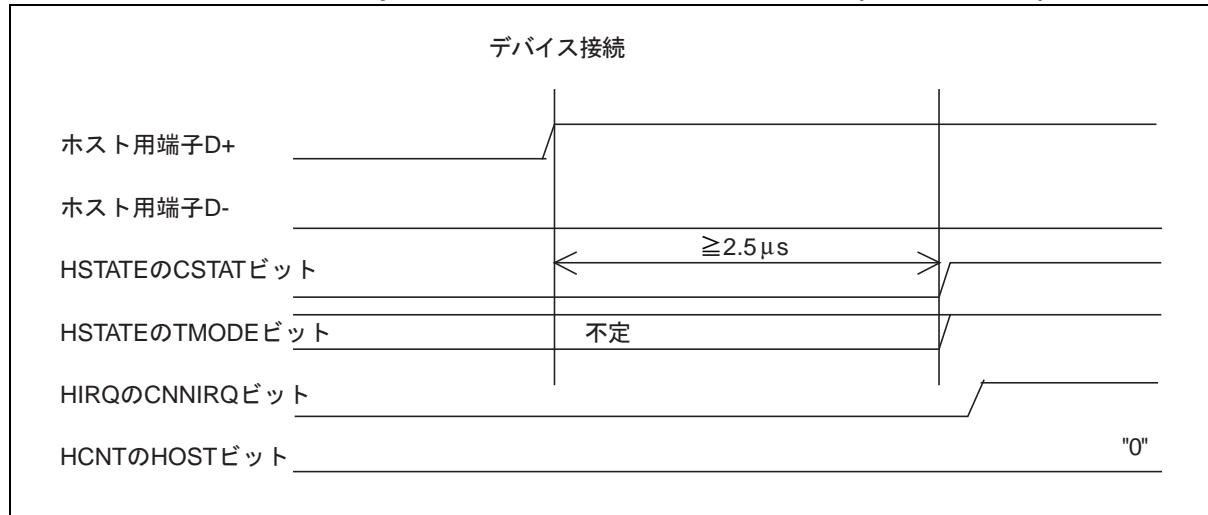
接続が検出された後、相手先 USB の転送可能速度を取得するためには、ホスト状態ステータスレジスタ (HSTATE) の TMODE の値を参照します。転送速度とホスト状態ステータスレジスタ (HSTATE) の TMODE ビットの関係は以下のようになります。

- ・ 接続先が Full Speed 対応装置 TMODE=1
- ・ 接続先が Low Speed 対応装置 TMODE=0

外部 USB 装置の転送速度を取得した後、UDC 制御レジスタ (UDCC) の RST ビットが "1" のとき、取得した転送速度に従ってホスト状態ステータスレジスタ (HSTATE) の CLKSEL ビットを更新してください。ホストステータスレジスタ (HSTATE) の TMODE ビットと CLKSEL ビットの関係は以下のとおりです。

- ・ TMODE=1 CLKSEL ビットに "1" を設定
- ・ TMODE=0 CLKSEL ビットに "0" を設定

図 14.5-1 Full Speed デバイスの接続検出タイミング例 (HCNT0 bit0=0)



< 注意事項 >

外部の USB 装置が接続されてから 2.5 μs 後ホスト状態ステータスレジスタ (HSTATE) の CSTAT ビットが "1" になります。

ホスト状態ステータスレジスタ (HSTATE) の TMODE ビット, CSTAT ビットはホストコントロールレジスタ 0(HCNT0) の HOST ビットの設定にかかわらず更新されます。

14.5.2 USB バスのリセット

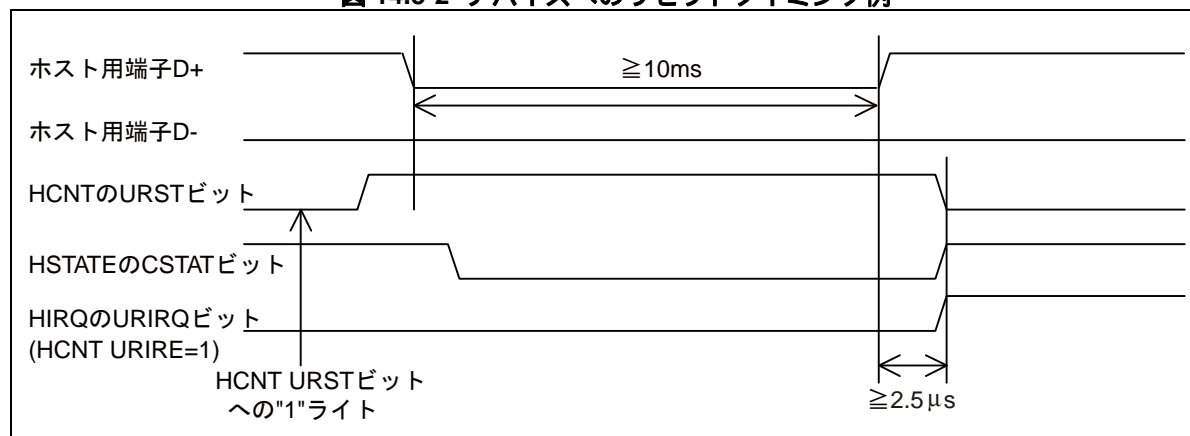
ホストモード時，ホストコントロールレジスタ 0(HCNT0) の URST ビットに "1" を設定すると，10 ms 以上の SE0 を送出し，USB バスをリセットします。USB バスのリセットが完了するとホストコントロールレジスタ 0(HCNT0) の URST ビットを "0" にしてホストコントロールレジスタ 0(HCNT0) の URIRE ビットが "1" ならば割込みを発生してホスト割込みレジスタ (HIRQ) の URIRQ ビットが "1" になります。割込みをクリアする場合にはホスト割込みレジスタ (HIRQ) の URIRQ ビットに "0" を書き込んでください。

■ USB バスのリセット前後の注意点

USB バスのリセットについて以下の点に注意してください。

1. USB バスへのリセット前にデバイスが接続されていることをホスト状態ステータスレジスタ (HSTATE) の CSTAT ビットが "1" になっていることで確認。
2. USB バスのリセットを実行するとホスト状態ステータスレジスタ (HSTATE) の CSTAT ビットは "0" になり，切断状態になります。そのとき，ホスト割込みレジスタ (HIRQ) の DIRQ ビットは "1" になりません。
3. USB バスへのリセットが終了後，ホスト状態ステータスレジスタ (HSTATE) の CLKSEL ビットと TMODE ビットを比較し，不一致であれば一致するように CLKSEL ビットを更新してください。更新するときには UDC 制御レジスタ (UDCC) の RST ビットが "1" のときに実施してください。

図 14.5-2 デバイスへのリセットタイミング例



14.5.3 トークンパケット

ホストモード時、IN トークン、OUT トークン、SETUP トークンのいずれかを実行する場合、ホストアドレスレジスタ (HADR)、EP1 制御レジスタの DIR ビットに従って EP1 制御レジスタ (EP1C) または EP2 制御レジスタ (EP2C) の PKS ビットを設定後、ホストトークンレジスタ (HTOKEN) に必要データを設定するとトークンパケットから開始します。SOF トークン時は、FRAME 設定レジスタ (HFRAME)、EOF 設定レジスタ (HEOF) 設定後、ホストトークンレジスタ (HTOKEN) に必要データを設定してください。各種レジスタ (HADR、EP1C、EP2C、HFRAME、HEOF) に変更がない場合には、設定は不要です。

■ トークンパケットの設定

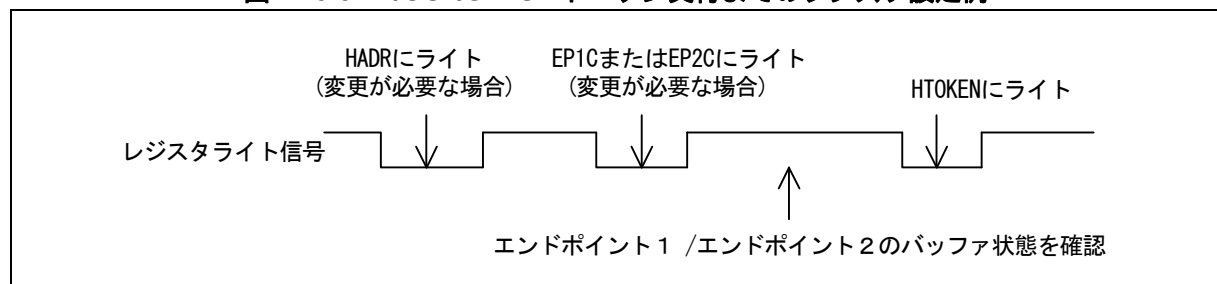
IN トークン、OUT トークン、SETUP トークンの場合、目的のアドレスをホストアドレスレジスタ (HADR) に、1 パケットの最大バイト数を実行しようとするトークンおよび EP1 制御レジスタの DIR ビットに従って EP1 制御レジスタ (EP1C) または EP2 制御レジスタの PKS ビットにセットします。

EP1 制御レジスタ (EP1C) の DIR ビットが "1" であれば、エンドポイント 1 のバッファが OUT 方向のバッファとして使用され、エンドポイント 2 のバッファが IN 方向のバッファとして使用されます。このとき、EP2 制御レジスタ (EP2C) の DIR ビットに "0" をセットしてください。

EP1 制御レジスタ (EP1C) の DIR ビットが "0" であれば、エンドポイント 1 のバッファが IN 方向のバッファとして使用され、エンドポイント 2 のバッファが OUT 方向のバッファとして使用されます。このとき、EP2 制御レジスタ (EP2C) の DIR ビットに "1" をセットしてください。

エンドポイント 1 のバッファを使用する場合、EP1 ステータスレジスタ (EP1S) の DRQ ビットに "0"、エンドポイント 2 のバッファを使用する場合、EP2 ステータスレジスタ (EP2S) の DRQ ビットに "0" がセットされていることを確認してからホストトークンエンドポイントレジスタ (HTOKEN) に目的のエンドポイント、トークンおよびトグルデータを設定します。USB 回路はその設定されたトークンに従って Sync、トークン、アドレス、エンドポイント、CRC5、EOP の順でトークンパケットを送出します (Sync、CRC5、EOP は自動)。1 パケット終了後、ホスト割込みレジスタ (HIRQ) の CMPIRQ ビットが "1" となり、ホストトークンエンドポイントレジスタ (HTOKEN) の TKNEN を 000_B にします (「14.5.7 SOF 割込み」参照)。そのとき、ホストコントロールレジスタ 0 (HCNT0) の CMPIRE ビットが "1" であれば、割込みが発生します。割込みをクリアするにはホスト割込みレジスタ (HIRQ) の CMPIRQ ビットに "0" を書き込んでください。

図 14.5-3 IN/OUT/SETUP トークン実行までのレジスタ設定例



SOFトークンの場合、EOF設定レジスタ(HEOF)およびFRAME設定レジスタ(HFRAME)に EOF 時間および FRAME 番号をセットし、ホストトークンエンドポイントレジスタ(HTOKEN)の TKNEN ビットに SOF トークンのコードを書き込むと Sync, SOF トークン, FRAME 番号, CRC5, EOP を送出し、ホスト状態ステータスレジスタ(HSTATE)の SOFBUSY ビットを "1" にし、FRAME 設定レジスタ(HFRAME)を +1 します。このとき、ホスト割込みレジスタ(HIRQ)の CMPIRQ も "1" にセットされ、ホストトークンエンドポイントレジスタ(HTOKEN)の TKNEN ビットが 000_B にクリアされます。ホストコントロールレジスタ 0(HCNT0)の CMPIRQ による割込みは発生しません。その後、自動的に発生する SOF の場合には、CMPIRQ による割込みは発生しません。トークン完了の割込みをクリアするにはホスト割込みレジスタ(HIRQ)の CMPIRQ に "0" を書き込んでください。

ホスト状態ステータスレジスタ(HSTATE)の SOFBUSY ビットが "1" の間、SOF は 1 ms ごとに自動的に送出されます。ホスト状態ステータスレジスタ(HSTATE)の SOFBUSY ビットが "0" となる条件(SOF の停止条件)は、以下のとおりです。

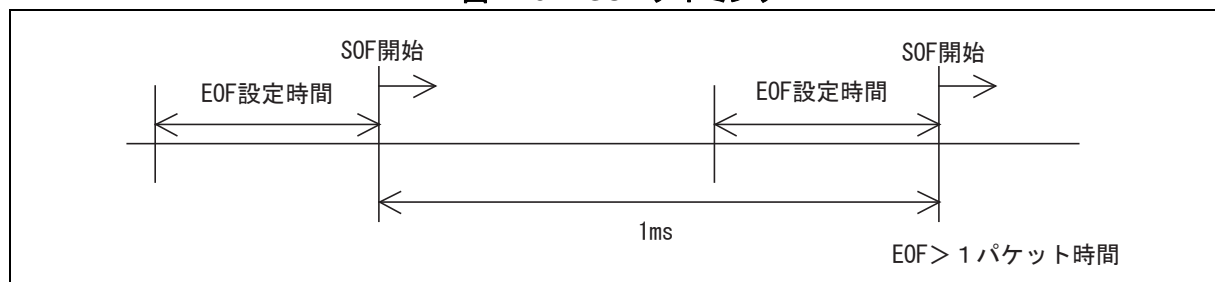
- ・ホスト状態レジスタ(HSTATE)の SOFBUSY ビットへの "0" 書き込み
- ・USB バスへのリセット(HCNT の URST ビットへの "1" 書き込み)
- ・ホスト状態ステータスレジスタ(HSTATE)の SUSP ビットへの "1" 書き込み
- ・デバイスの切断(HSTATE の CSTAT ビットが "0" の場合)

ホストモードからファンクションモードへ切り換える場合には、ホスト状態レジスタ(HSTATE)の SOFBUSY ビットへ "0" を書き込み、ホスト状態レジスタ(HSTATE)の SOFBUSY ビットが "0" になったことを確認してから切り換えてください。

ホスト状態ステータスレジスタ(HSTATE)の SOFBUSY ビットを再び "1" にするには SOF トークンをもう 1 度実行する必要があります。

EOF 設定レジスタは、SOF と他のトークンの同時実行を防止します。EOF 設定時間から SOF 開始時間までの間にホストトークンエンドポイントレジスタ(HTOKEN)の TKNEN ビットに書き込んだ場合、設定したトークンをウェイトされ、SOF 実行後にウェイトしているトークンを実行します。EOF 設定レジスタの時間単位は 1 ビット時間で、例えば EOF 設定レジスタに 10_H と設定した場合、Full Speed モードでは $16 \times 1/12 \text{ MHz} = 1333.3 \text{ ns}$ 、Low Speed モードでは $16 \times 1/1.5 \text{ MHz} = 10666.6 \text{ ns}$ という時間になります。EOF 設定時間の設定が 1 パケット時間より短く設定すると、他のトークン実行中に SOF 実行が重なる場合があります。この場合、ホストエラーステータスレジスタ(HERR)の LSTSOF ビットが "1" にセットされ SOF は実行されません。ホストエラーステータスレジスタ(HERR)の LSTSOF に "1" が設定された場合には、EOF 設定レジスタのデータを大きくする必要があります(「14.4.8 EOF 設定レジスタ(HEOF)」参照)。

図 14.5-4 SOF タイミング



14.5.4 データパケット

トークンパケット送出後，データパケットの送信の場合，ホストトークンエンドポイントレジスタ (HTOKEN) の TGGL ビットに従ってトグルデータを送出し，EP1 制御レジスタ (EP1C) の DIR ビットによってエンドポイント 1 またはエンドポイント 2 のバッファデータ，CRC16 データ，EOP を送出します。データパケットの受信の場合，ホストトークンエンドポイントレジスタ (HTOKEN) の TGGL ビットと受信したトグルデータを比較し，一致した場合には受信データを EP1 制御レジスタ (EP1C) の DIR ビットによってエンドポイント 1 またはエンドポイント 2 のバッファに書き込み，CRC16 のエラーを検査します。

■ データパケット

トークンパケット送出後，データパケットを以下の手順で実行します。

● 送信の場合

- (1) Sync の自動送出
- (2) ホストトークンエンドポイントレジスタ (HTOKEN) の TGGL ビットが "0" であれば DATA0, TGGL ビットが "1" であれば DATA1 を送出
- (3) EP1 制御レジスタ (EP1C) の DIR ビットが "1" であればエンドポイント 1 のバッファ，EP1 制御レジスタ (EP1C) の DIR ビットが "0" であればエンドポイント 2 のバッファを選択し，送信データをすべて送出
- (4) CRC 16 ビットを送出
- (5) EOP 2 ビットを送出
- (6) J State 1 ビットを送出

● 受信の場合

- (1) Sync の受信
- (2) トグルデータを受信し，ホストトークンエンドポイントレジスタ (HTOKEN) の TGGL ビットと比較します。
- (3) トグルデータを比較した結果，一致すれば EP1 制御レジスタ (EP1C) の DIR ビットが "1" であればエンドポイント 2 のバッファ，EP1 制御レジスタ (EP1C) の DIR ビットが "0" であればエンドポイント 1 のバッファを選択し，受信データをライトしていきます。
- (4) EOF を受信したとき，CRC 16 ビットを検査します。

ホストコントロールレジスタ 0 (HCNT0) の HOST ビットが "1" のときには，EP1 制御レジスタ (EP1C) の DIR ビットと EP2 制御レジスタ (EP2C) の DIR ビットは反転させたデータをセットしてください。例えば，EP1 制御レジスタ (EP1C) の DIR ビットが "0" であれば，EP2 制御レジスタ (EP2C) の DIR ビットには "1" をセットします。

14.5.5 ハンドシェークパケット

ハンドシェークパケットにより、送受信相手に自分の状態を通知します。

■ ハンドシェークパケット

ハンドシェークパケットは、データ受信に対し正常に受信できる状態にあるか、またエンドポイントがサポートしているか等を判断して受信側が ACK, NAK, STALL のいずれかを送出します。そのとき、USB 回路はハンドシェークパケットを受信した場合は、受信したハンドシェークパケットをホストエラーステータスレジスタ (HERR) の HS ビットにセットします。ハンドシェークパケットを送信した場合は送信したハンドシェークパケットをホストエラーステータスレジスタ (HERR) の HS ビットにセットします。

14.5.6 リトライ機能

パケット終了時, NAK および CRC エラー等のエラーが発生し, ホストコントロールレジスタ 1(HCNT1) の RETRY ビットが "1" のとき, リトライタイマレジスタ (HRTIMER) で設定された時間, リトライし続けます。

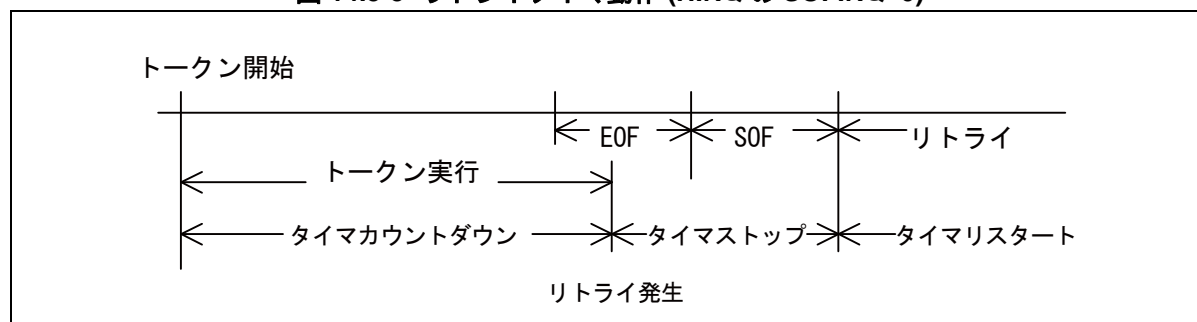
■ リトライ機能

STALL およびデバイスの切断以外のエラーが発生した場合, ホストコントロールレジスタ 1(HCNT1) の RETRY ビットが "1" であれば, そのトークンをリトライします。リトライの終了条件は, 以下のとおりです。

- ・ ホストコントロールレジスタ 1(HCNT1) の RETRY ビットの "0" 設定
- ・ リトライタイマの 0 検出
- ・ SOF による割込みの発生 (HCNT0 の SOFIRE=1, HIRQ の SOFIRQ=1)
- ・ ACK の検出
- ・ デバイスが切断されたことの検出

リトライタイマはトークン開始時に起動され, 1 ビットの転送クロックにてカウントダウンされますが, EOF 領域内でリトライが発生した場合にはカウントは停止します。HIRQ の SOFIRQ ビットが "0" で SOF トークンが終了した場合には, 停止したタイマ値からリスタートし, リトライタイマが 0 となり, パケットが終了した時点でリトライ要求がきてもリトライ要求はキャンセルし, パケットを終了します。

図 14.5-5 リトライタイマ動作 (HIRQ の SOFIRQ=0)



リトライが終了したとき, その終了パケットの終了情報が各レジスタにセットされます。

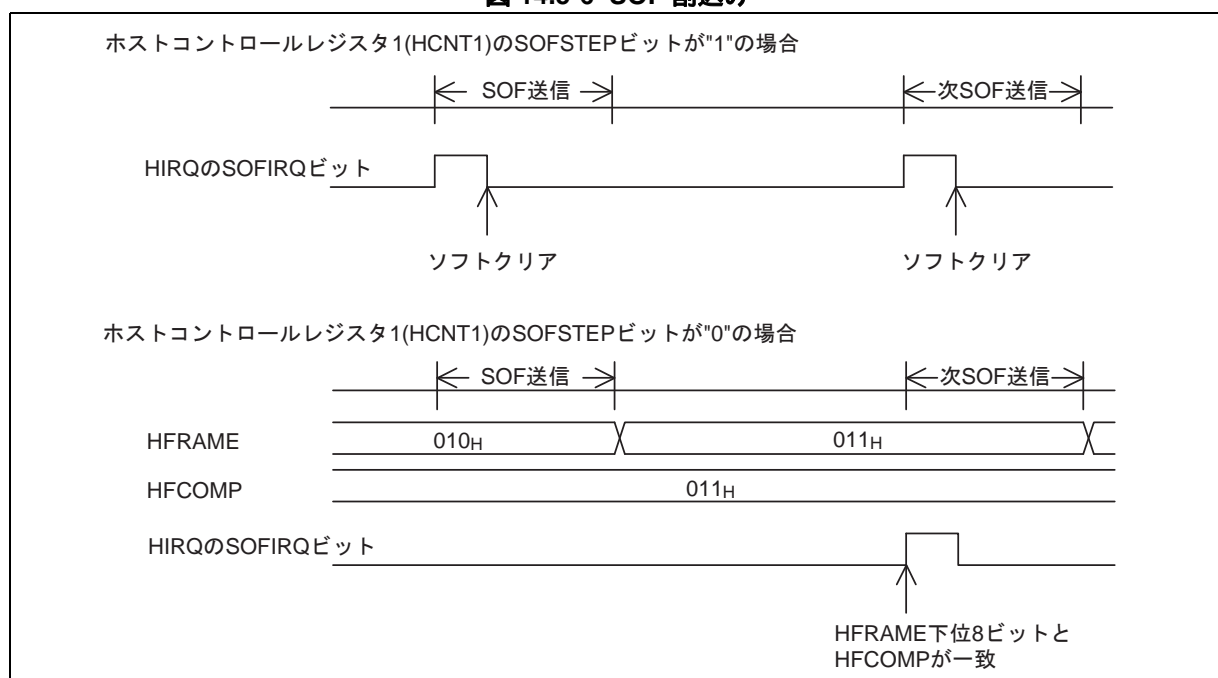
14.5.7 SOF 割込み

ホストコントロールレジスタ 0(HCNT0) の SOFIRE ビットに "1" を設定するとホストコントロールレジスタ 1(HCNT1) の SOFSTEP ビットおよび SOF 割込み FRAME 比較レジスタ (HFCOMP) によって SOF 開始時, ホスト割込みレジスタ (HIRQ) の SOFIRQ ビットを "1" にし, 割込みを発生します。ホストトークンエンドポイントレジスタ (HTOKEN) による SOF 実行ではホスト割込みレジスタ (HIRQ) の SOFIRQ ビットは "1" になりません。

■ SOF 割込み

ホストコントロールレジスタ 0(HCNT0) の SOFIRE ビットに "1" を設定すると SOF を送出する時に割込みを発生し, ホスト割込みレジスタ (HIRQ) の SOFIRQ ビットに "1" をセットします。SOF による割込み条件には, ホストコントロールレジスタ 1(HCNT1) の SOFSTEP ビットの設定によって SOF が送出されるごとか, SOF 割込み FRAME 比較レジスタ (HFCOMP) の示す下位 8 ビットの Frame 番号かを選択できます。

図 14.5-6 SOF 割込み



ホストコントロールレジスタ 1(HCNT1) の CANCEL ビットを "1" に設定すると, EOF 領域内でホストトークンエンドポイントレジスタ (HTOKEN) に SOF トークン以外のトークンを設定し, 次の SOF でホスト割込みレジスタ (HIRQ) の SOFIRQ ビットが "1" になる場合, そのトークンは実行せずにホストトークンエンドポイントレジスタ (HTOKEN) の TKNEN ビットは 000_B にクリアされません。そのとき, ホスト割込みレジスタ (HIRQ) の CMPIRQ ビットは "1" になりません。トークンがキャンセルされたことは, SOFIRQ ビットが "1" になったときにホスト割込みレジスタ (HIRQ) の TCAN ビットによって知ることができます。再度, トークンを実行したい場合には, ホスト割込みレジスタ (HIRQ) の TCAN ビットに "0" を書き込み, ホストトークンエンドポイントレ

ジスタ (HTOKEN) の TKNEN ビットに実行させたいトークンを書き込んでください。
ホストコントロールレジスタ 1 (HCNT1) の CANCEL ビットを "0" にした場合には SOF
送信後、ホストトークンエンドポイントレジスタ (HTOKEN) に設定されたトークンを実行します。

図 14.5-7 HCNT1 の CANCEL ビット =1 のトークンキャンセル動作例

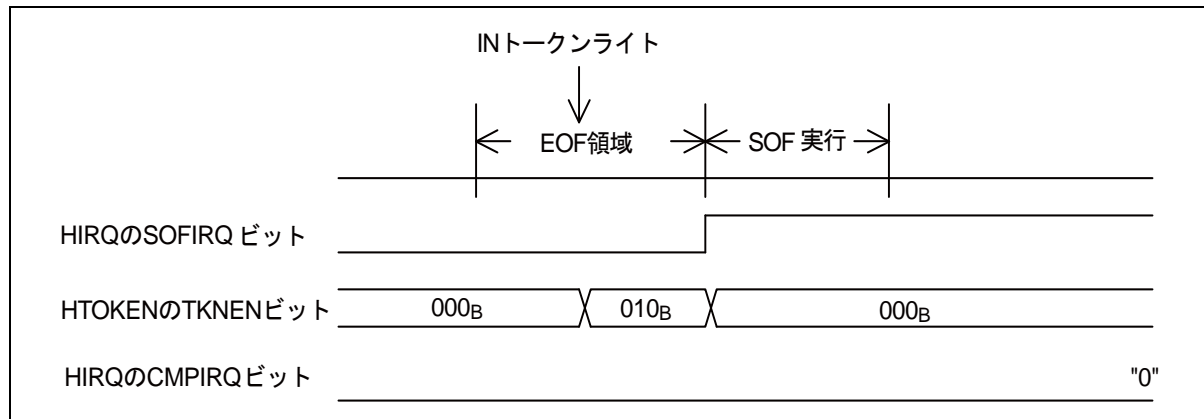
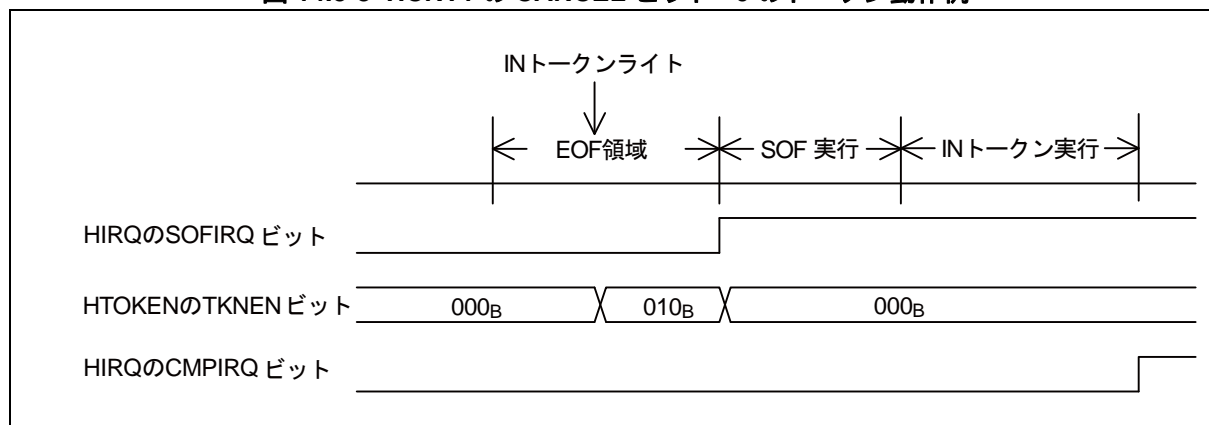


図 14.5-8 HCNT1 の CANCEL ビット =0 のトークン動作例



14.5.8 エラーステータス

USB ホストは各種エラー情報をサポートしています。

■ エラーステータス

● スタッフィングエラー

6 ビット連続 "1" が発生した場合, "0" を 1 ビット挿入することになっていますが, 7 ビット連続 "1" を検出するとスタッフィングエラーとしてホストエラーステータスレジスタ (HERR) の STUFF ビットを "1" にセットします。これをクリアするには STUFF ビットに "0" を書き込んでください。STUFF ビットをクリアせずに次のトークンを実行した場合には次のトークンの終了時に更新されます。

● トグルエラー

IN トークン時, データパケットのトグルデータとホストトークンエンドポイントレジスタ (HTOKEN) の TGGL ビットを比較し, 一致しなかった場合ホストエラーレジスタ (HERR) の TGERR ビットに "1" をセットします。TGERR ビットをクリアするには, ホストエラーレジスタ (HERR) の TGERR ビットに "0" を書き込んでください。TGERR ビットをクリアせずに次のトークンを実行した場合には次のトークンの終了時に更新されます。

● CRC エラー

IN トークン時, 受信したデータパケットのデータおよび CRC を CRC の多項式 $G(X)=X^{16}+X^{15}+X^2+1$ で計算し, 剰余が $800D_H$ でなければ CRC エラーが発生したことになり, ホストエラーレジスタ (HERR) の CRC ビットに "1" をセットします。CRC ビットをクリアするには, ホストエラーレジスタ (HERR) の CRC ビットに "0" を書き込んでください。CRC ビットをクリアせずに次のトークンを実行した場合には次のトークンの終了時に更新されます。

● タイムアウトエラー

所定時間内にデータパケットやハンドシェークが入力されなかった, 受信データ中に SE0 が検出された, スタッフィングエラーが検出された場合等, ホストエラーステータスレジスタ (HERR) の TOUT ビットに "1" がセットされます。TOUT ビットをクリアするには, ホストエラーレジスタ (HERR) の TOUT ビットに "0" を書き込んでください。TOUT ビットをクリアせずに次のトークンを実行した場合には次のトークンの終了時に更新されます。

● 受信エラー

受信バッファとして EP1 が使用されている場合, EP1 制御レジスタ (EP1C) の PKS ビット, EP2 が使用されている場合, EP2 制御レジスタ (EP2C) の PKS ビットが受信パケットサイズとなります。その受信パケットサイズより多く受信データを受信した場合, ホストエラーステータスレジスタ (HERR) の RERR ビットに "1" がセットされます。RERR ビットをクリアするには, ホストエラーレジスタ (HERR) の RERR ビットに "0" を書き込んでください。RERR ビットをクリアせずに次のトークンを実行した場合には次のトークンの終了時に更新されます。

14.5.9 パケット終了

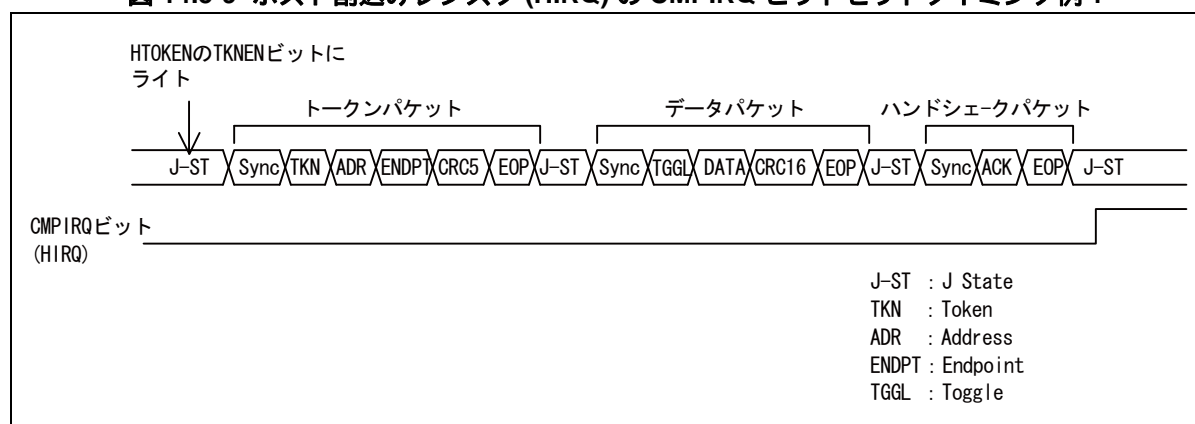
USB ホストにおいて 1 パケット終了するとホストコントロールレジスタ 0 (HCNT0) の CMPIRE ビットが "1" のとき、割込みが発生しホスト割込みレジスタ (HIRQ) の CMPIRQ ビットを "1" にします。

■ パケット終了タイミング

1 パケット終了すると以下のタイミングで割込みが発生します。

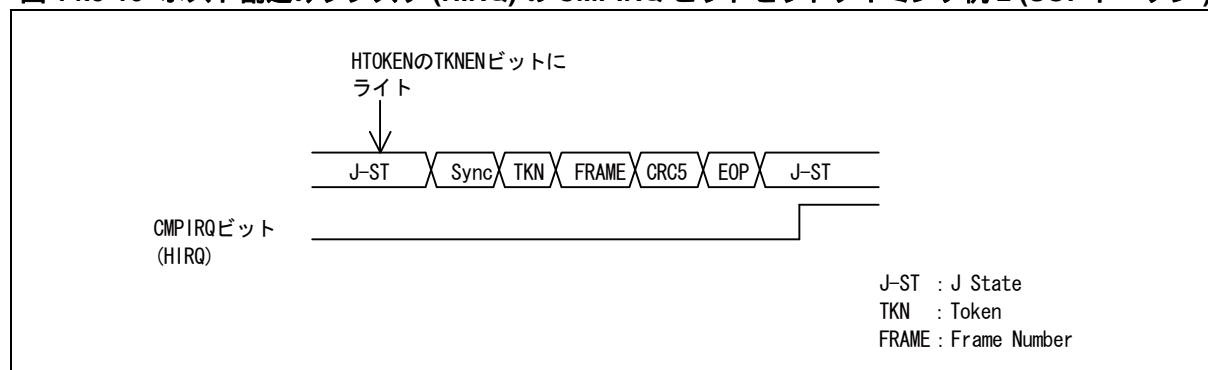
- ホストトークンエンドポイントレジスタ (HTOKEN) の TKNEN ビットが 001_B, 010_B, 011_B のとき (SETUP トークン, IN トークン, OUT トークン)

図 14.5-9 ホスト割込みレジスタ (HIRQ) の CMPIRQ ビットセットタイミング例 1



- ホストトークンエンドポイントレジスタ (HTOKEN) の TKNEN ビットが 100_B のとき (SOF トークン)

図 14.5-10 ホスト割込みレジスタ (HIRQ) の CMPIRQ ビットセットタイミング例 2 (SOF トークン)



14.5.10 サスペンド・レジューム

USB ホストにおいてサスペンドおよびレジュームをサポートしています。

■ サスペンド動作

ホスト状態ステータスレジスタ (HSTATE) の SUSP ビットに "1" を書き込むと、

- ・ USB バスのハイインピーダンス状態
- ・ クロックが必要ない回路ブロックの停止

の順に実行し、USB回路をサスペンド状態にします。サスペンド状態になった場合には、ホスト状態ステータスレジスタ (HSTATE) の SUSP ビットを "1" にします。

USB バスへのリセット中、HSTATE の SOF BUSY が "1" のとき、およびデータの送受信中にサスペンドに設定したり、サスペンド中 USB へ供給しているクロックを停止したりすることは禁止です。

■ レジューム動作

サスペンド状態からレジュームの動作を開始するには、以下のいずれかの条件が成立したときです。

- (1) ホスト状態ステータスレジスタ (HSTATE) の SUSP ビットに "0" を書き込む
- (2) 簡易ホスト用端子 D+, 簡易ホスト用端子 D- が k-state になったことを検出
- (3) デバイスが切断されたことを検出
- (4) デバイスが接続されたことを検出

ホスト割込みレジスタ (HIRQ) の RWKIRQ ビットが "1" にセットされた後トークンの発行が可能となります。各条件による動作タイミングは以下のとおりです。

図 14.5-11 レジスタによるレジューム動作 (Full Speed モード時)

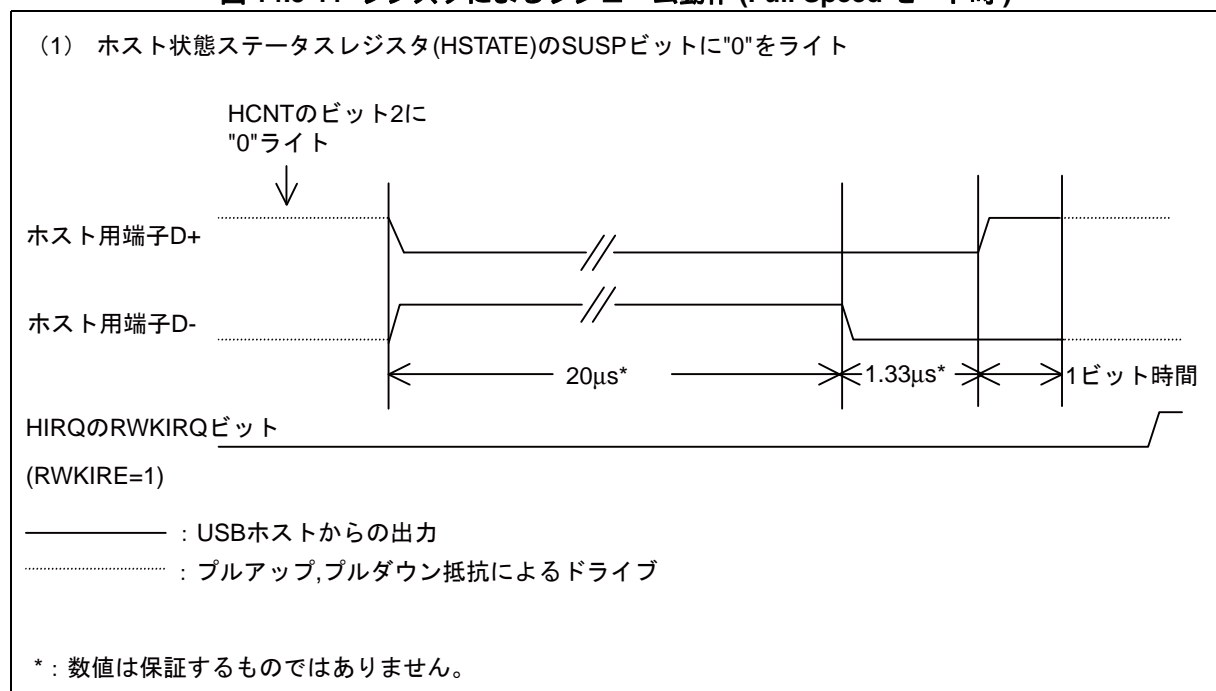


図 14.5-12 デバイスからのレジューム動作 (Full Speed モード)

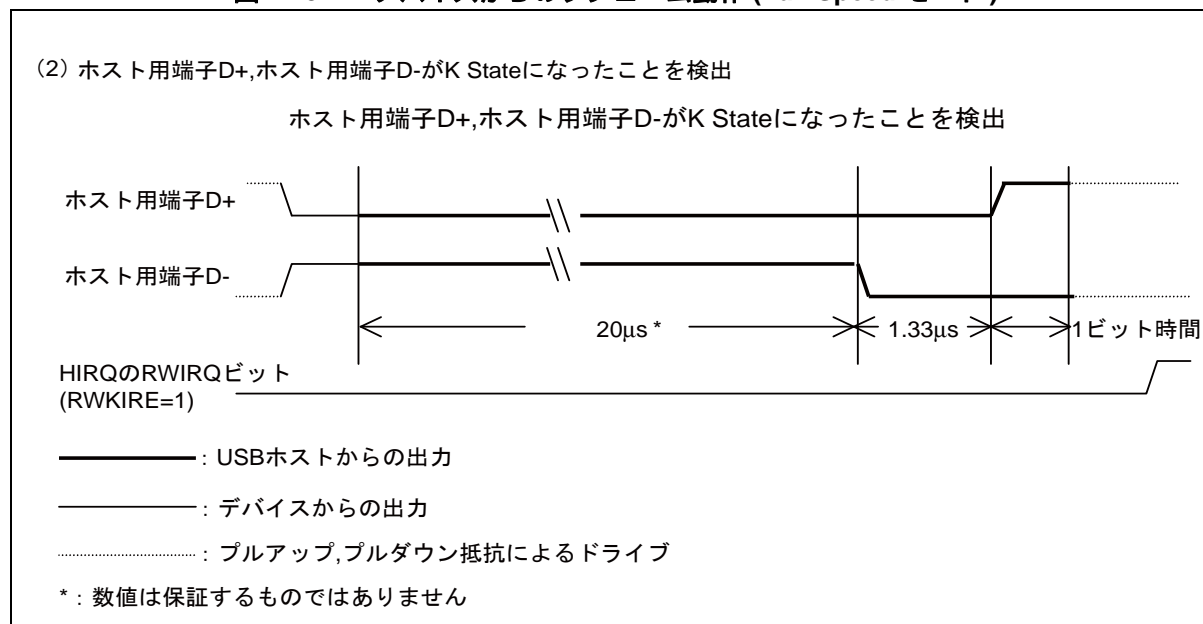


図 14.5-13 デバイス切断によるレジューム動作

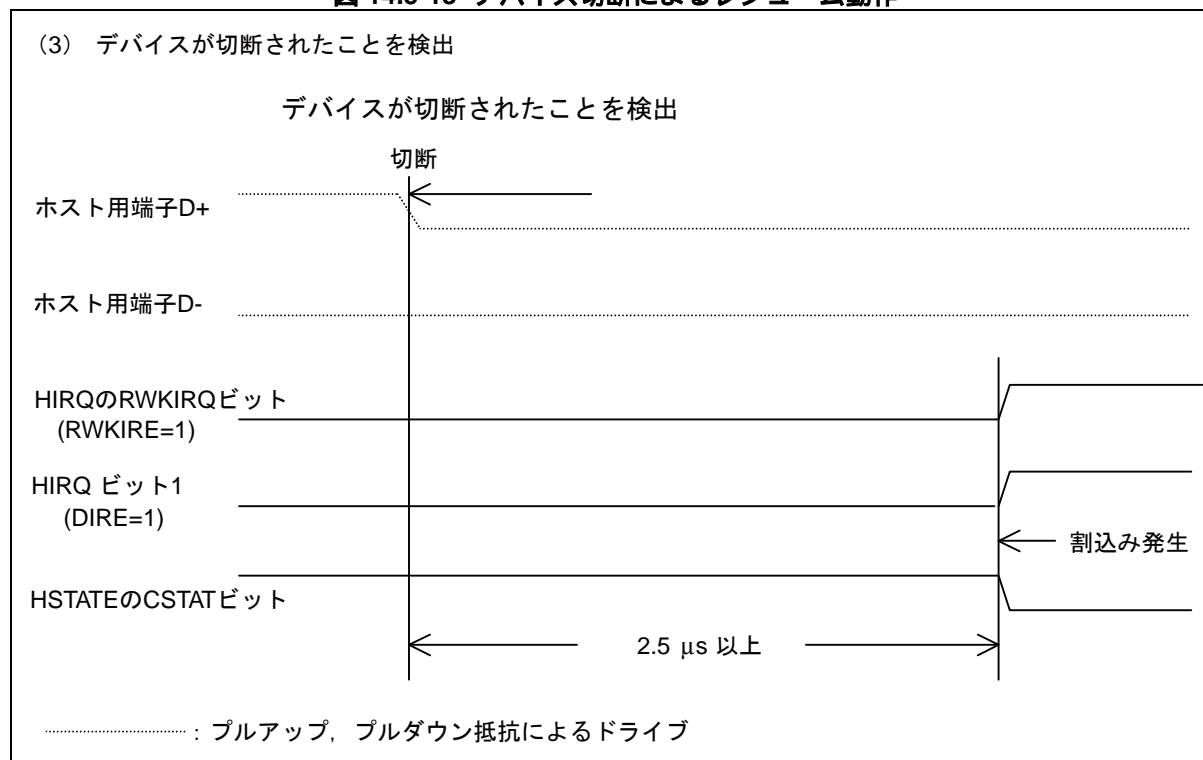
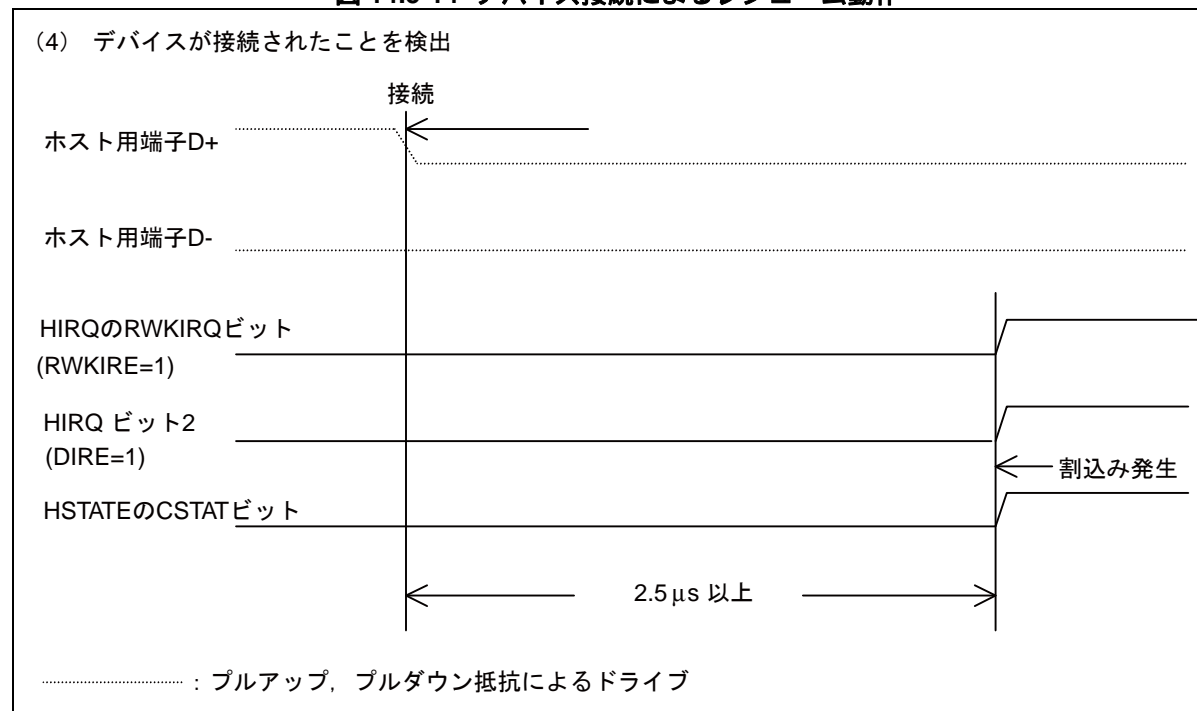


図 14.5-14 デバイス接続によるレジューム動作



14.5.11 デバイスの切断

ホスト用端子 D+, D- とともに "L" になると切断タイマが起動され, 2.5 μ s 以上 "L" を検出した場合, ホスト状態ステータスレジスタ (HSTATE) の CSTAT ビットを "0" にします。

■ デバイスの切断

ホストモード, ファンクションモードに関係なくホスト用端子 D+, ホスト用端子 D- が両方とも 2.5 μ s 以上 "L" を検出するとデバイスが切断されたと判断し, ホスト状態ステータスレジスタ (HSTATE) の CSTAT ビットが "0" となり, ホスト割込みレジスタ (HIRQ) の DIRQ ビットは "1" にセットされます。ホストコントロールレジスタ 0 (HCNT0) の DIRE ビットが "1" であれば割込みが発生します。その割込みをクリアしたい場合には, ホスト割込みレジスタ (HIRQ) の DIRQ ビットに "0" をライトしてください。

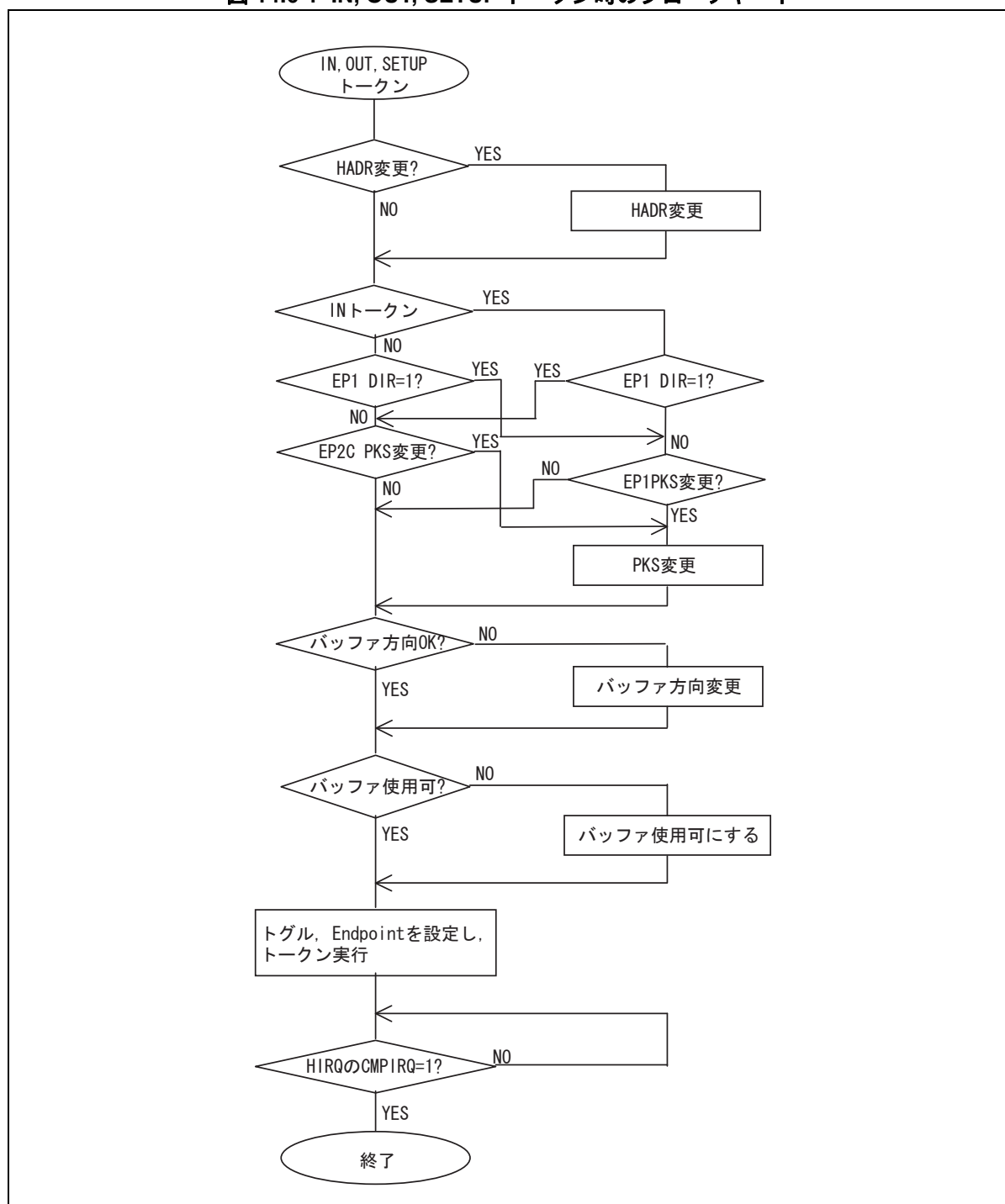
USB バスへのリセットを実行すると切断されたと判断し, ホスト状態レジスタ (HSTATE) の CSTAT ビットを "0" にしますが, ホスト割込みレジスタ (HIRQ) の DIRQ ビットは "1" になりません。

14.6 USB ホストの各トークンフローチャート

USB ホストの各トークンのフローチャートは以下のとおりです。

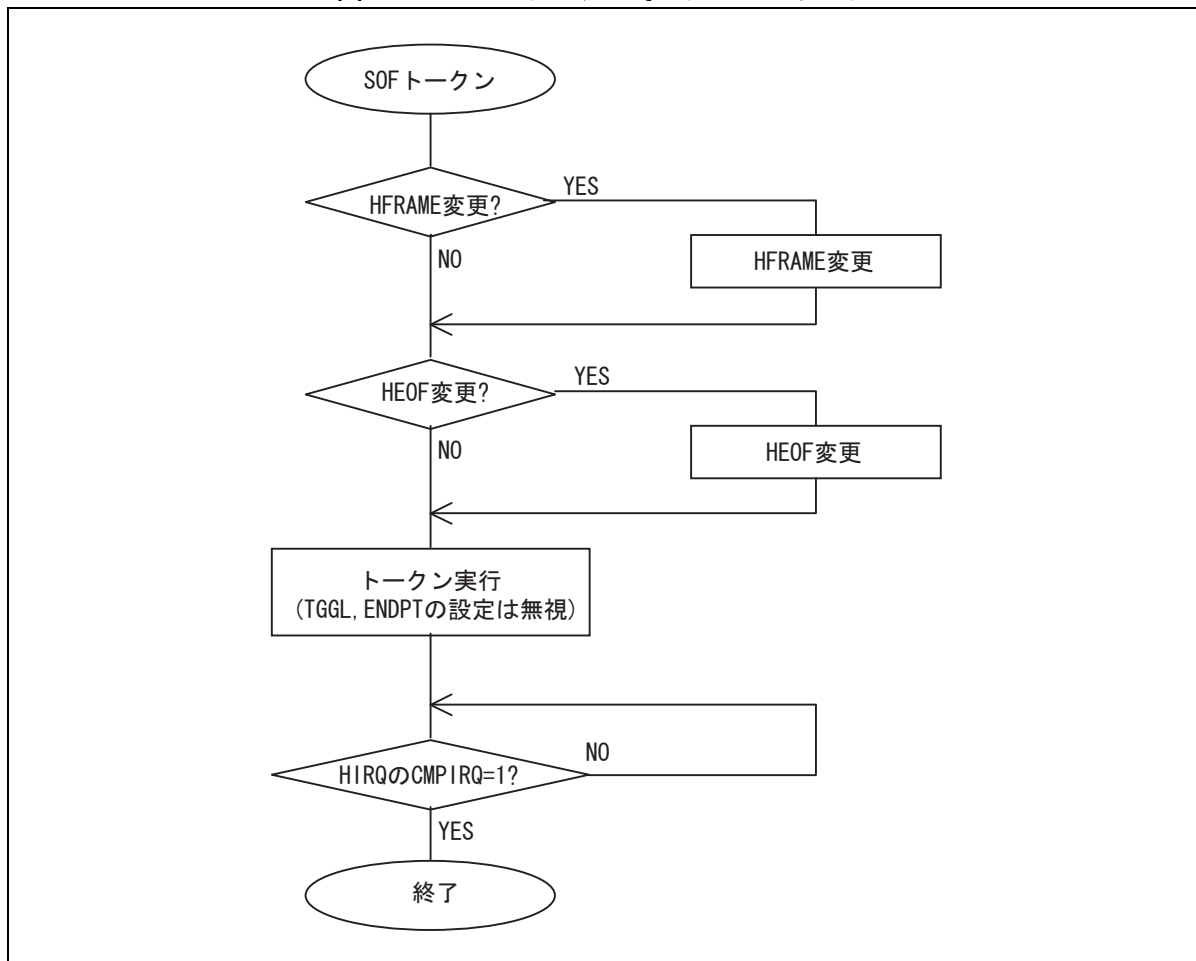
■ IN, OUT, SETUP トークン

図 14.6-1 IN, OUT, SETUP トークン時のフローチャート



■ SOF トークン

図 14.6-2 SOF トークン時のフローチャート



第15章

PWC タイマ

この章では、PWC タイマの概要、レジスタの構成 / 機能、PWC タイマの動作および PWC タイマの使用上の注意について説明します。

- 15.1 PWC タイマの概要
- 15.2 PWC タイマのレジスタ
- 15.3 PWC タイマの動作
- 15.4 PWC タイマの使用上の注意

15.1 PWC タイマの概要

PWC タイマは、入力信号のパルス幅測定機能を持つ、16 ビットの多機能アップカウントタイマです。

PWC: Pulse Width Count(パルス幅測定)

■ PWC タイマの機能

ハードウェアとして、16 ビットアップカウントタイマ 1 個、入力パルス分周器 & 分周比制御レジスタ 1 個、測定入力端子 1 本、16 ビット制御レジスタ 1 個を 1 チャンネルとして、合計 1 チャンネルを有し、これらによって以下の機能を実現します。

● タイマ機能

- 設定された時間ごとに割込み要求を発生できます。
- 基準となる内部クロックは、3 種類より選択できます。

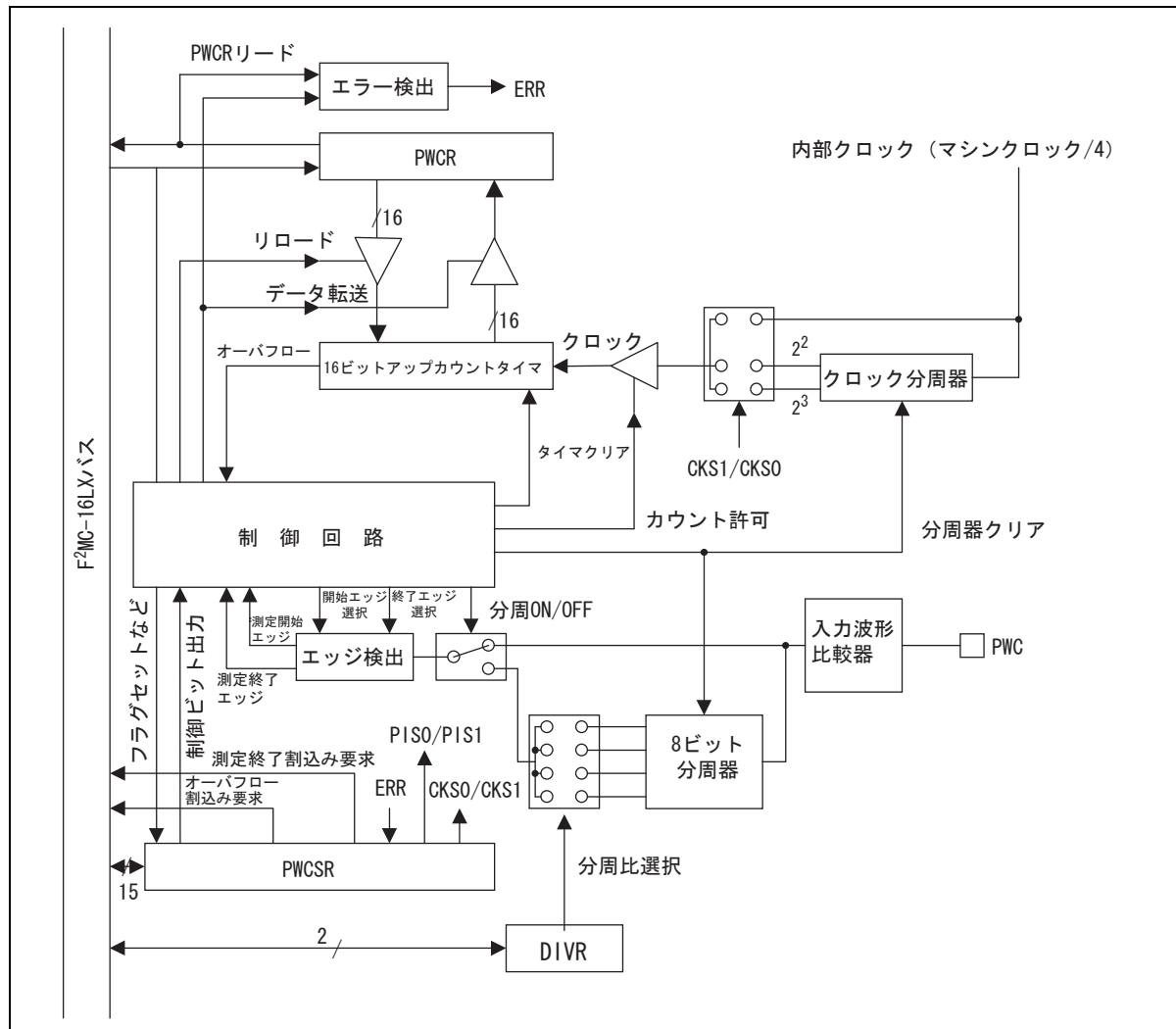
● パルス幅測定機能

- 外部からのパルス入力の任意イベント間の時間を測定します。
- 基準となる内部クロックは、3 種類より選択できます。
- 各種測定モード
 - "H" パルス幅 (~)/"L" パルス幅 (~)
 - 立上り周期 (~)/立下り周期 (~)
 - エッジ間測定 (or ~ or)
- 8 ビット入力分周器で、入力パルスを 2^{2n} 分周 ($n=1, 2, 3, 4$) して分周測定を行うことができます。
- 測定終了時に割込みを発生させることができます。
- 1 回のみの測定か、連続測定かを選択できます。

■ PWC タイマのブロックダイアグラム

図 15.1-1 に、PWC タイマのブロックダイアグラムを示します。

図 15.1-1 PWC タイマのブロックダイアグラム



15.2.1 PWC コントロールステータスレジスタ (PWCSR)

PWC コントロールステータスレジスタ (PWCSR) の構成および機能について説明します。

■ PWC コントロールステータスレジスタ (PWCSR)

図 15.2-2に、PWCコントロールステータスレジスタ(PWCSR)のビット構成を示します。

図 15.2-2 PWC コントロールステータスレジスタ (PWCSR) のビット構成

bit	15	14	13	12	11	10	9	8	PWCSR
00005D _H	STRT	STOP	EDIR	EDIE	OVIR	OVIE	ERR	予約	PWC コントロール ステータスレジスタ
	(R/W)	(R/W)	(R)	(R/W)	(R/W)	(R/W)	(R)	(R/W)	初期値 0000000X _B
bit	7	6	5	4	3	2	1	0	PWCSR
00005C _H	CKS1	CKS0	PIS1	PIS0	S/C	MOD2	MOD1	MOD0	PWC コントロール ステータスレジスタ
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	初期値 00000000 _B

R/W : リード・ライト可能
R : リードオンリ
- : 未定義

以下に、PWC コントロールステータスレジスタ (PWCSR) の各ビットの機能を説明します。

[bit15, bit14] STRT, STOP(タイマスタートビット, タイマストップビット)

16 ビットアップカウントタイマの起動 / 再起動 / 停止を制御するビットです。読出し時には、タイマの動作状態を表示します。

STRT および STOP ビットの機能を示します。

表 15.2-1 書き込み時機能 (16 ビットアップカウントタイマの動作制御)

STRT	STOP	動作制御機能
0	0	機能なし / 動作に影響なし
0	1	タイマ起動 / 再起動 (カウント許可) *
1	0	タイマ動作強制停止 (カウント禁止) *
1	1	機能なし / 動作に影響なし

*: クリアビット命令使用可能

表 15.2-2 読出し時機能 (16 ビットアップカウントタイマの動作状態表示)

STRT	STOP	状態表示機能
0	0	タイマ停止中 (起動されていないかまたは測定終了) [初期値]
1	1	タイマカウント動作中 (測定中)

- リセット時 "00_B" に初期化されます。
- 読出し / 書込みが可能です。ただし、書込み時と読出し時では、表 15.2-1、表 15.2-2 に示すように意味が異なります。
- リードモディファイライト (RMW) 系命令における読出し値は、ビット値にかかわらず "11_B" です。
- タイマの起動 / 停止のための STRT/STOP ビットへの書込みは、それぞれのビットに対するビット命令 (ビットクリア) を用いることが可能ですが、動作状態の読出しには、ビット命令は使用できませんので注意してください (読み出すと必ず動作中となります)。

[bit13] EDIR (測定終了割込み要求フラグ)

パルス幅測定時、測定が終了したことを示すフラグです。測定終了割込み要因が許可されているとき (bit12:EDIE=1) に、このビットがセットされると、測定終了割込み要求が発生します。

EDIR	動作モード
セット要因	パルス幅測定が終了するとセット (PWCR に測定結果が収納されている)
クリア要因	PWCR (測定結果) を読み出すことによってクリア

- リセット時 "0" に初期化されます。
- 読出しのみ可能です。
- 書き込んでもビット値は変化しません。

[bit12] EDIE (測定終了割込み許可)

パルス幅測定時の、測定終了割込み要求を下表のように制御します。

EDIE	動作モード
0	測定終了割込み要求出力禁止 (EDIR がセットされていても割込み発生せず) [初期値]
1	測定終了割込み要求出力許可 (EDIR がセットされると割込みが発生する)

- リセット時 "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit11] OVIR(タイマオーバーフロー割込み要求フラグ)

16 ビットアップカウントタイマが FFFF_H から 0000_H へオーバーフローしたことを示すフラグです。タイマオーバーフロー割込み要求が許可されているとき (bit10:OVIE=1 のとき) にこのビットがセットされると、タイマオーバーフロー割込み要求が発生します。

OVIR	動作モード
セット要因	タイムオーバーフローが発生するとセット (FFFF _H から 0000 _H)
クリア要因	"0" 書込み, または μ DMAC によりクリア

- リセット時 "0" に初期化されます。
- 読出し / 書込みが可能です。ただし、書込みは "0" のみ可能で、"1" を書き込んでもビット値は変化しません。
- リードモディファイライト (RMW) 系命令における読出しは、ビット値にかかわらず "1" です。

[bit10] OVIE(タイマオーバーフロー割込み要求許可)

パルス幅測定時の測定終了割込み要求を制御します。

OVIE	動作モード
0	オーバーフロー割込み要求出力禁止 (OVIR がセットされていても割込み発生せず) [初期値]
1	オーバーフロー割込み要求出力許可 (OVIR がセットされると割込みが発生する)

- リセット時 "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit9] ERR(エラーフラグ)

パルス幅測定時の連続測定モード時において、PWCR 内の測定結果を読み出さないうちに、次の測定が終了されてしまったことを示すフラグです。この際、PWCR の値は新しい測定結果に更新され、1 つ前の測定結果は消失します。測定は、ビット値に関係なく続行されます。

ERR	動作モード
セット要因	読み出していない測定結果が次の結果により消失するとセット
クリア要因	PWCR(測定結果) を読み出すことによりクリア

- リセット時 "0" に初期化されます。
- 読出しのみ可能です。ただし、書き込んでもビット値は変化しません。

[bit8] 予約ビット

予約ビットです。必ず "0" を書き込んでください。

[bit7, bit6] CKS1, CKS0 (カウントクロック選択)

内部カウントクロックは、3 種類の中からいずれかを選択できます。

表 15.2-3 16 ビットアップカウントタイマのカウントクロック

CKS1	CKS0	動作モード
0	0	マシンのクロックの 4 分周クロック (マシンのクロック =24 MHz 時 0.17 μ s) [初期値]
0	1	マシンのクロックの 16 分周クロック (マシンのクロック =24 MHz 時 0.67 μ s)
1	0	マシンのクロックの 32 分周クロック (マシンのクロック =24 MHz 時 1.33 μ s)
1	1	設定禁止 (未定義)

- リセット時, "00_B" に初期化されます。
- 読出し / 書込みが可能です。ただし, "11_B" の設定は禁止です。

< 注意事項 >

起動後の書換えは、禁止です。必ず起動前か停止後に書き込んでください。

[bit5, bit4] PIS1, PIS0 (パルス幅測定入力端子選択)

パルス幅測定入力端子を選択します。

表 15.2-4 パルス幅測定入力端子の選択

PIS1	PIS0	動作モード
0	0	(PWC 端子を選択) [初期値]
0	1	設定禁止
1	0	設定禁止
1	1	設定禁止 (未定義)

- リセット時, "00_B" に初期化されます。
- 読出し / 書込みが可能です。ただし "01_B", "10_B", "11_B" は設定してはいけません。

< 注意事項 >

起動後の書換えは、禁止です。必ず起動前か停止後に書き込んでください。

[bit3] S/C(測定モード (単発 / 連続) の選択)

測定モードを選択します。

表 15.2-5 16 ビットアップカウントタイマの測定モードの選択

S/C	測定モード選択	タイマモード時	パルス幅
0	単発測定モード [初期値]	リロードなし (ワンショット)	1 回測定後停止
1	連続測定モード	リロードあり (リロードタイマ)	連続測定 : バッファレジスタ有効

- リセット時 , "0" に初期化されます。
- 読出し / 書込みが可能です。

< 注意事項 >

起動後の書換えは禁止です。必ず起動前か停止後に書き込んでください。

[bit2 ~ bit0] MOD2 ~ MOD0(動作モード / 測定エッジ選択)

動作モードおよび幅測定を行うエッジを選択します。

表 15.2-6 16 ビットアップカウントタイマの動作モード / 測定エッジの選択

MOD2	MOD1	MOD0	動作モード / 測定エッジ選択
0	0	0	タイマモード [初期値]
0	0	1	タイマモード (リロードモードのみ)
0	1	0	全エッジ間パルス幅測定モード (or ~ or)
0	1	1	分周周期測定モード (入力分周有効)
1	0	0	立上りエッジ間周期測定モード (~)
1	0	1	"H" パルス幅測定モード (~)
1	1	0	"L" パルス幅測定モード (~)
1	1	1	立下りエッジ間周期測定モード (~)

- リセット時 , "00_B" に初期化されます。
- 読出し / 書込みが可能です。

< 注意事項 >

起動後の書換えは , 禁止です。必ず起動前か停止後に書き込んでください。

15.2.2 PWC データバッファレジスタ (PWCR)

PWC データバッファレジスタ (PWCR) の構成および機能について説明します。

■ PWC データバッファレジスタ (PWCR)

図 15.2-3 に、PWC データバッファレジスタ (PWCR) のビット構成を示します。

図 15.2-3 PWC データバッファレジスタ (PWCR) のビット構成

bit	15	14	13	12	11	10	9	8	PWCR
00005F _H	D15	D14	D13	D12	D11	D10	D9	D8	PWC データバッファレジスタ
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	初期値 00000000 _B
bit	7	6	5	4	3	2	1	0	PWCR
00005E _H	D7	D6	D5	D4	D3	D2	D1	D0	PWC データバッファレジスタ
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	初期値 00000000 _B

R/W : リード・ライト可能

PWC データバッファレジスタ (PWCR) は、PWCSR の bit2 ~ bit0(MOD2 ~ MOD0) の設定によるタイマモード時と、パルス幅設定モード時で機能が異なります。

● タイマモード時 (読出し / 書込み可能)

リロードタイマ動作時 (PWCSR の bit3:S/C=1) は、リロードデータを保持するリロードレジスタとなります。この場合、読出し / 書込みともに可能です。

ワンショットタイマ動作モード時 (PWCSR のビット 3:S/C=0) は、アップカウントタイマを直接アクセスする窓口になります。この場合も、読出し / 書込みともに可能ですが、書込みはタイマ停止中に行ってください。読出しは随時可能で、カウント中のタイマ値を読み出すことができます。

● パルス幅測定モード時 (読出しのみ可能)

連続測定モード時 (PWCSR の bit3:S/C=1) は、前回の測定結果を保持するバッファレジスタとなります。

この場合は読出しのみ可能で、書き込んでもレジスタ値は変化しません。

単発モード時 (PWCSR の bit3:S/C=0) は、アップカウントタイマを直接アクセスする窓口となります。

この場合も読出しのみ可能で、書き込んでもレジスタ値は変化しません。読出しは随時可能で、カウント中のタイマ値が得られます。測定終了後は、測定結果を保持します。

< 注意事項 >

PWCR のアクセスは、必ずワード転送命令で行ってください。リセット時 "00_B" に初期化されます。

15.2.3 PWC 分周比制御レジスタ (DIVR)

PWC 分周比制御レジスタ (DIVR) の構成および機能について説明します。

■ PWC 分周比制御レジスタ (DIVR)

図 15.2-4 に、PWC 分周比制御レジスタ (DIVR) のビット構成を示します。

図 15.2-4 PWC 分周比制御レジスタ (DIVR) のビット構成

bit	7	6	5	4	3	2	1	0	DIVR
000060 _H	-	-	-	-	-	-	DIV1	DIV0	分周比制御レジスタ
	(-)	(-)	(-)	(-)	(-)	(-)	(R/W)	(R/W)	初期値 -----00 _B
R/W : リード・ライト可能									
- : 未定義									

[bit7 ~ bit2] 未定義ビット

読出し値は不定です。書込みは動作に影響を与えません。

[bit1, bit0] DIV1, DIV0(分周比選択)

分周周期測定モード (PWCSR の bit2 ~ bit0:MOD2, MOD1, MOD0=001_B) において使用するレジスタで、ほかのモードでは意味をもちません。

分周周期測定モードには、DIVR によって設定された分周比だけ測定端子に入力されたパルスを分周し、分周後の 1 周期幅を測定します。

DIV1	DIV0	カウントクロック選択
0	0	4 分周 [初期値]
0	1	16 分周
1	0	64 分周
1	1	256 分周

- リセット時, "00_B" に初期化されます。
- 読出し / 書込みが可能です。

< 注意事項 >

起動後の書換えは、禁止です。必ず起動前か停止後に書き込んでください。

15.3 PWC タイマの動作

PWC タイマの動作について説明します。

■ PWC タイマ動作概略

PWC タイマは、16 ビットアップカウントタイマを基本にした多機能タイマで、測定入力端子と 8 ビット入力分周などを組み込んでいます。PWC タイマには、次の 2 つの主要機能があります。

- タイマ機能
- パルス幅カウント機能

どちらの機能でも 3 種類のカウンタクロックを選択できます。以下に、各機能における基本性能 / 動作について説明します。

15.3.1 タイマ機能の動作

リロード動作とワンショット動作が可能なアップカウントタイマです。

■ タイマ機能の動作

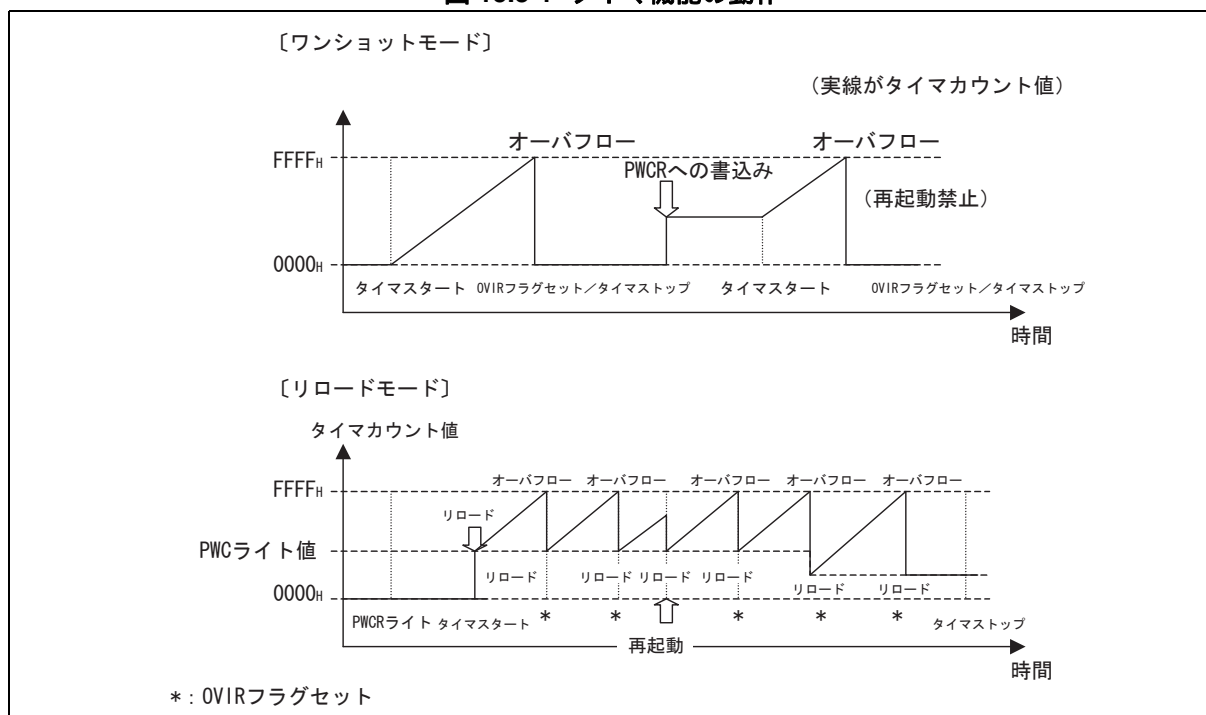
タイマ起動後, カウントクロックごとにカウントアップを行います。0000_H FFFF_H の範囲でオーバーフローが発生すると, 割込み要求が発生するおそれがあります。

オーバーフロー発生した場合, モードによって以下のように動作します。

- ・ワンショットモード : カウントを停止します。
- ・リロードモード : タイマにリロードレジスタの内容をリロードし, 再びカウントを開始します。

図 15.3-1 に, タイマ機能のワンショットモードおよびリロードモードの動作を示します。

図 15.3-1 タイマ機能の動作



15.3.2 パルス幅測定機能の動作

入力パルスの任意イベント間の時間周期をタイマで測定できます。

■ パルス幅測定機能の動作

パルス幅測定機能は、起動後、設定した測定開始エッジが入力されるまでカウントを開始しません。開始エッジを検出するとタイマを 0000_H にクリア後カウントアップを開始し、停止エッジを検出するとカウントを停止します。この間のカウント値がパルス幅としてレジスタに保存されます。測定終了時に割込みを検出します。

測定終了後は、測定モードに応じて以下のように動作します。

- ・ 単発測定モード：動作を中断します。
- ・ 連続測定モード：タイマ値はバッファレジスタに転送され、次の開始エッジが入力されるまで測定が中断されます。

図 15.3-2 に単発測定モードの動作を、図 15.3-3 に連続測定モードの動作を示します。

図 15.3-2 パルス幅測定動作 (単発測定モード/"H" 幅測定)

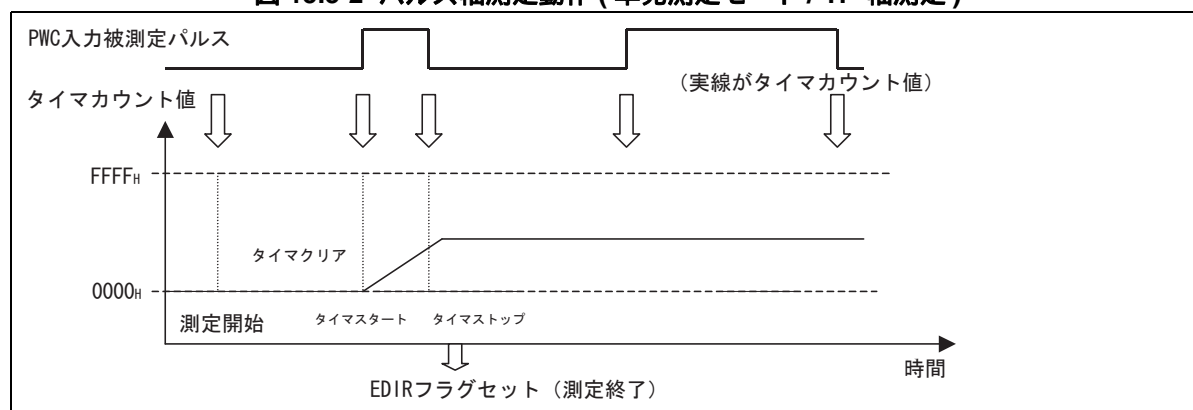
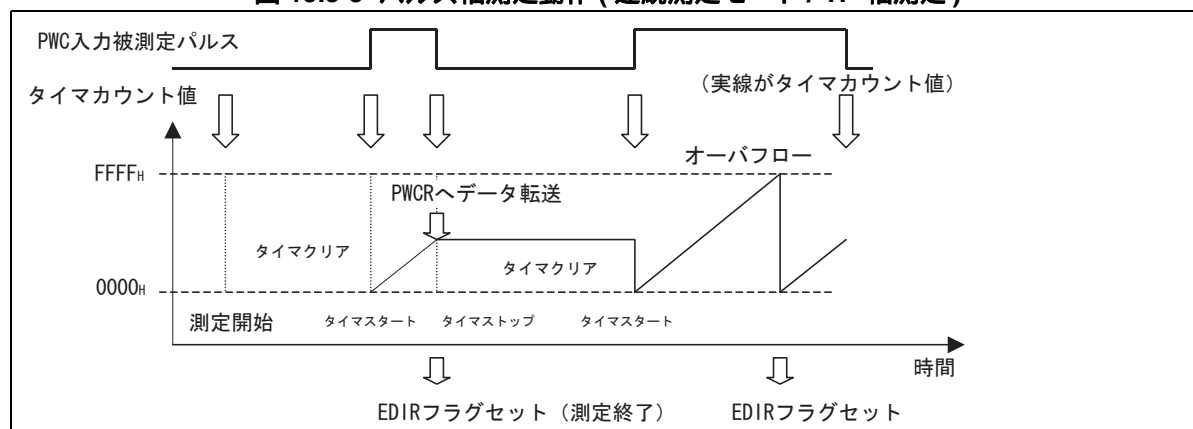


図 15.3-3 パルス幅測定動作 (連続測定モード/"H" 幅測定)



15.3.3 カウントクロックおよび動作モードの選択

カウントクロックの選択および動作モードの選択について説明します。

■ カウントクロックの選択

タイマのカウントクロックは、PWCSR の bit7(CKS1) および bit6(CKS0) の設定によって内部クロックソース 3 種類から選択できます。

表 15.3-1 に、カウントクロックの選択内容を示します。

表 15.3-1 カウントクロックの選択内容

PWCSR/bit7, bit6: CKS1, CKS0	選択される内部カウントクロック
00 _B	マシンのクロックの 4 分の 1 (24 MHz のマシンのクロックの場合は 0.17 μs) [初期値]
01 _B	マシンのクロックの 16 分の 1 (24 MHz のマシンのクロックの場合は 0.67 μs)
10 _B	マシンのクロックの 32 分の 1 (24 MHz のマシンのクロックの場合は 1.33 μs)

リセット後の初期値では、マシンのクロックの 4 分周クロックが選択されます。

< 注意事項 >

カウントクロックの選択は、必ずタイマ起動前に行ってください。

■ 動作モードの選択

各動作モード / 測定モードの選択は、PWCSR ビットの設定によって行います。

- 動作モードの選択：PWCSR の bit2 ~ bit0(MOD2, MOD1, MOD0 ビット) タイマモード / パルス幅測定モードの選択，測定エッジの決定など
- 測定モードの設定：PWCSR の bit3(S/C ビット) 単発測定 / 連続測定またはリロード / ワンショットの選択

表 15.3-2 に、動作モード / 測定モードの設定内容を示します。

表 15.3-2 動作モード / 測定モードの設定内容

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
パルス幅測定	or ~ or 全エッジ間測定	単発測定：バッファ無効	0	0	1	0	
		連続測定：バッファ有効	1	0	1	0	
	分周周期測定 (1 ~ 256 分周)	単発測定：バッファ無効	0	0	1	1	
		連続測定：バッファ有効	1	0	1	1	
	~ 立上り間周期測定	単発測定：バッファ無効	0	1	0	0	
		連続測定：バッファ有効	1	1	0	0	
	~ "H" パルス幅測定	単発測定：バッファ無効	0	1	0	1	
		連続測定：バッファ有効	1	1	0	1	
	~ "L" パルス幅測定	単発測定：バッファ無効	0	1	1	0	
		連続測定：バッファ有効	1	1	1	0	
	~ 立下り間周期測定	単発測定：バッファ無効	0	1	1	1	
		連続測定：バッファ有効	1	1	1	1	

リセット後の初期設定では、ワンショットタイマが選択されています。

< 注意事項 >

動作モードの選択は、必ずタイマ起動前に行ってください。

15.3.4 タイマ / パルス幅測定 of 起動と停止

各動作の起動 / 再起動 / 停止 / 強制停止は、PWCSR の bit15, bit14(STRT, STOP ビット) によって行います。

■ タイマ / パルス幅測定 of 起動と停止

タイマ / パルス幅測定 of 起動 / 再起動は STRT ビット、強制停止は STOP ビットと機能が分かれており、それぞれ "0" を書き込むことによって機能します。しかし、このときに両ビットに書き込む値が排他でないと機能しません。ビット操作命令以外の命令 (バイト以上の命令) で書き込む場合は、必ず表 15.3-3 に示すビットの組合せに限られます。

表 15.3-3 STRT ビットと STOP ビットの機能

STRT	STOP	機能
0	1	タイマ / パルス幅測定 of 起動 / 再起動
1	0	タイマ / パルス幅測定 of 強制停止

ビット操作命令 (クリアビット命令) を用いる場合、ハードウェアによって自動的に表 15.3-3 に示す組合せで書き込まれるため、特に意識する必要はありません。

■ 起動後の動作

タイマモードおよびパルス幅測定モード of 起動後の動作は、以下のとおりです。

● タイマモード

直ちにカウント動作を開始します。

● パルス幅測定モード

測定開始エッジが入力されるまでカウントは行いません。測定開始エッジ検出後、16 ビットアップカウントタイマを "0000_B" にクリアし、カウントを開始します。

■ 再起動

タイマ / パルス幅測定モード of 起動後、動作中に起動をかける (STRT ビットに "0" を書き込む) ことを再起動とよびます。

再起動をかけると、モードによって、以下の動作が行われます。

● ワンショットタイマモード

動作に影響はありません。

● リロードタイマモード

リロード動作を行い、動作を続行します。オーバフロー発生タイミングと同時に再起動をかけた場合、オーバフローフラグ (OVR) はセットされます。

● パルス幅測定モード

測定開始エッジ待ち状態の場合、動作に影響はありません。測定中の場合は、カウントを停止し再度測定開始エッジ待ち状態となります。この際、測定終了エッジ検出と再起動が同時になると、測定終了フラグ (EDIR) がセットされ、連続測定モード時は結果が PWCR に転送されます。

■ 停止

ワンショットタイマモード/単発測定モードでは、タイマのオーバフローまたは測定終了より自動的にカウントを停止しますので、特に意識して停止する必要はありません。ただし、ほかのモードでは、タイマを強制的に停止させる必要があります。タイマが自動的に停止する前にタイマを停止させたい場合にも同様です。

■ 動作状態の確認

前述の STRT, STOP ビットは、読出し時には、動作状態表示ビットとして機能します。表 15.3-4 に、動作状態表示ビットの機能を示します。

表 15.3-4 動作状態表示ビットの機能

STRT	STOP	動作状態
0	0	タイマ停止 (測定開始エッジ待ち状態を除く): 起動されていないかまたは測定が終了したことを示します。
1	1	タイマカウント動作中または測定開始エッジ状態待ち

< 注意事項 >

STRT, STOP どちらのビットを読み出しても同じ値となります。ただし、リードモディファイライト (RMW) 系命令 (ビット処理命令など) で STRT, STOP ビットを読み出すと常に "11_B" になるため、これらの命令を使用して読み出さないでください。

15.3.5 タイマモードの動作

タイマモードの動作について説明します。

■ タイマのクリア

次の場合には、16 ビットアップカウントタイマが "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、測定開始エッジを検出しカウントを開始する場合

■ ワンショット動作モード

ワンショット動作モードでは、タイマ起動後カウントクロックごとにカウントアップを行い、"FFFF_H" から "0000_H" へカウントする際のオーバフロー発生によって自動的に停止します。タイマ起動前に PWCR に値を設定しておく、その値からカウントが開始されます。この場合、設定した値は保持されず、PWCR の値は現行のカウント値を示します。

■ リロード動作モード

リロード動作モードでは、タイマ起動後 PWCR 中のリロード値をタイマに設定してからカウントクロックごとにカウントアップを行います。"FFFF_H" から "0000_H" へカウントする際のオーバフロー発生によって、再度 PWCR 中のリロード値をタイマに設定し（リロード動作）、カウント動作を繰り返します。PWCSR の STOP ビットへの書き込みによる強制停止されるかまたはリセットされるまで、タイマは停止しません。タイマ起動前に PWCR に設定した値は、リロード値としてカウント中も保持され、起動 / 再起動時およびオーバフローが生じる際には必ずタイマに設定されます。カウント中に設定値を変更する場合は、次のオーバフロー発生時またはタイマの再起動の際に、変更後の新規リロード値を使用します。

■ タイマ値とリロード値

ワンショット動作モードでの PWCR は、直接アップカウントタイマにアクセスします。PWCR に値を書き込むとそのままタイマに書き込まれ、タイマが動作中に PWCR の値を読み出すとカウント中のタイマ値が得られます。タイマ起動前に PWCR に任意の値を書き込んでから起動すると、この指定された値からカウントが開始されます。リロード動作モードでは、アップカウントタイマにアクセスすることが不可能です。PWCR は、リロード値を保持するリロードレジスタとして機能します。タイマの起動 / 再起動 / オーバフローが生じる際には必ず、PWCR に書き込まれた値がタイマに設定されます。PWCR を読み出すと、格納されたリロード値が読み出されます。

PWCR の値とタイマの値が定まらないのは、リロード動作モードを強制的に中断させた後にタイマをワンショット動作モードに設定した場合です。したがって、タイマの使用前に、必ず値を書き込んでから使用してください。

■ 割込み発生要求

タイマモード動作においては、タイマのオーバフローによる割込み要求が発生することが可能です。カウントアップによってオーバフローが発生するとオーバフローフラグがセットされ、オーバフロー割込み要求が許可されていると割込み要求が発生します。

■ タイマ周期

ワンショット動作モードで PWCR に "0000_H" を設定してタイマを起動すると、65536 回カウントアップ後にオーバフローを発生し、カウントを停止します。起動から停止までの時間は、次の式によって計算されます。

$T_1 = (65536 - n_1) \times t$

- T₁: 起動から停止までの時間 [μs]
- n₁: 起動時に PWCR に書き込まれていたタイマ値
- t: カウントクロックの周期 [μs]

リロード動作モードで PWCR に "0000_H" を設定してタイマを起動すると、65536 回カウントアップごとにオーバフローを発生します。リロード周期の時間は、次の式によって計算されます。

$T_R = (65536 - n_R) \times t$

- T_R: リロード周期 (オーバフロー周期) [μs]
- n_R: PWCR に保持されているリロード値
- t: カウントクロックの周期 [μs]

■ カウントクロックと最大周期

タイマモードの場合、PWCR に設定する値が "0000_H" のときに最大周期となります。
マシナクロック (以下 φ とする) = 24 MHz 時のカウントクロックの周期とタイマの最大周期を表 15.3-5 に示します。

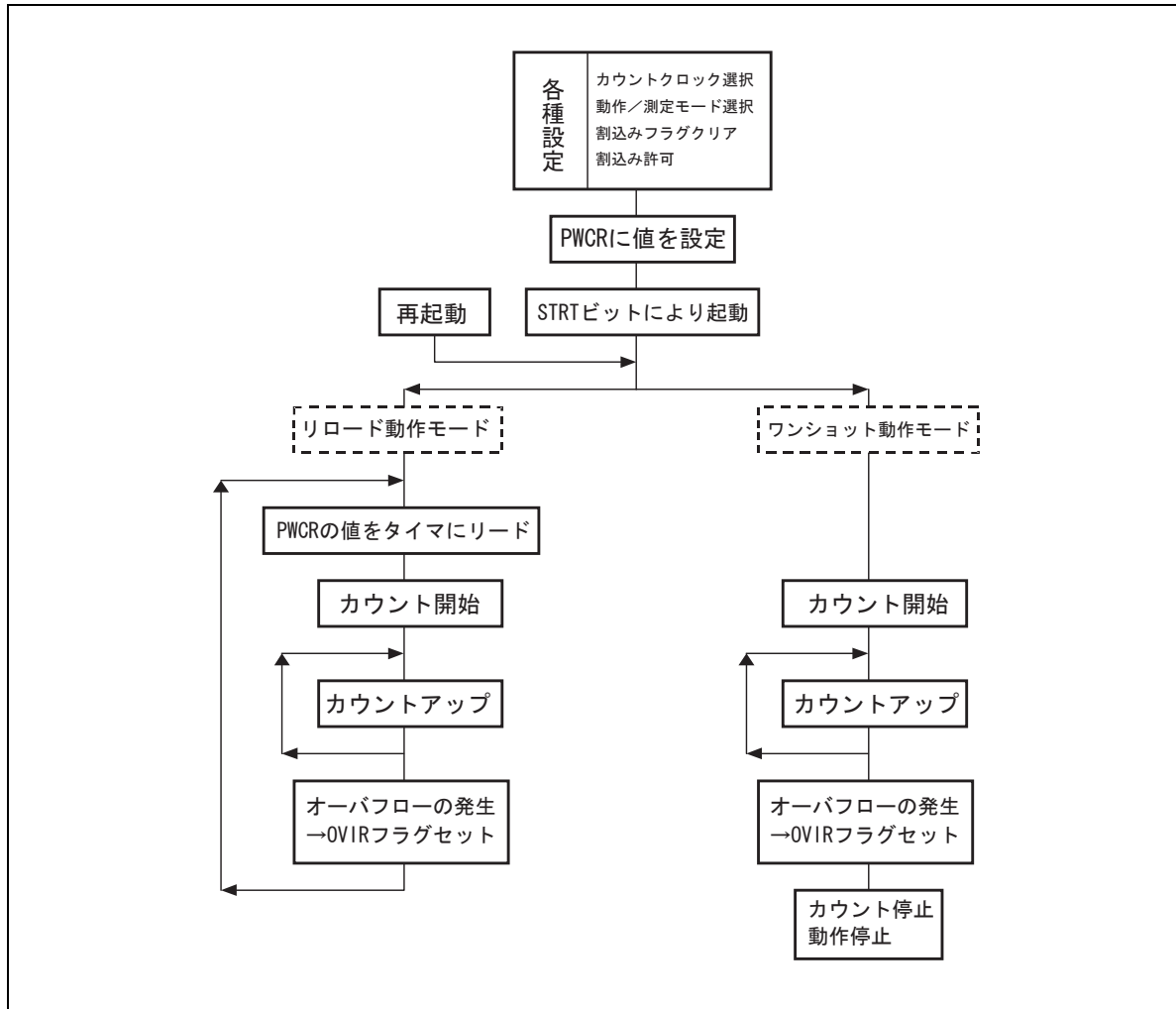
表 15.3-5 カウントクロックと周期

カウントクロック選択	CSK1, CSK0=00: (φ/4) 時	CSK1, CSK0=00: (φ/16) 時	CSK1, CSK0=00: (φ/32) 時
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	43.7 ms	87.4 ms

■ タイマの動作フロー

図 15.3-4 に、タイマの動作フローを示します。

図 15.3-4 タイマの動作フロー



15.3.6 パルス幅測定モードの動作

パルス幅測定モードの動作について説明します。

■ 単発測定と連続測定

パルス幅測定モードには、1 回のみ測定を行うモードと、連続して測定を行うモードがあります。各モードは、PWCSR の S/C ビットによって選択します（「15.3.3 カウントクロックおよび動作モードの選択」を参照してください）。

● 単発測定モード

1 回目の測定終了エッジが入力されると、タイマのカウンタが停止して PWCSR 中の測定終了フラグ (EDIR) がセットされ、以降は測定を行いません（ただし、同時に再起動がかかった場合は、測定開始待ちとなります）。

● 連続測定モード

測定終了エッジが入力されると、タイマのカウンタが停止して PWCSR 中の測定終了フラグ (EDIR) がセットされ、再度測定開始エッジが入力されるまでカウントを停止します。再度、測定開始エッジが入力されると、タイマを "0000_H" にクリアした後測定を開始します。測定終了時タイマ内の測定結果は PWCR に転送されます。

< 注意事項 >

測定モードの選択変更は、必ずタイマの停止中に行ってください。

■ 測定結果のデータ

単発測定モードと連続測定モードでは、測定結果とタイマ値の扱い、PWCR の機能に違いがあります。両モードにおける測定結果は以下のとおりです。

● 単発測定モード

- 動作中に PWCR を読み出すと、測定中のタイマ値が得られます。
- 測定終了後に PWCR を読み出すと、測定結果データが得られます。

● 連続測定モード

- 測定終了時、タイマ内の測定結果は PWCR に転送されます。
- PWCR を読み出すと直前の測定結果が得られ、測定動作中も前回の測定結果を保持しています。測定中のタイマ値は読み出せません。

連続測定モードで、測定結果を読み出さないうちに次の測定が終了してしまった場合は、前回の測定結果が新しい測定結果に消されてしまいます。この際、PWCSR 中のエラーフラグ (ERR) がセットされます。エラーフラグ (ERR) は PWCR を読み出すと自動的にクリアされます。

■ 測定モードとカウンタ動作

入力されたパルスのどこを測定するかによって、測定モードは6種類のうちから選択できます。さらに、高い周波数のパルスの幅を精度よく測定できるように、入力されたパルスを任意分周して周期を測定するモードも用意されています。表 15.3-6 に、測定モード一覧を示します。

表 15.3-6 測定モード一覧 (1 / 2)

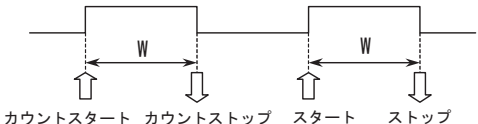
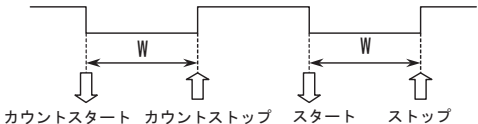
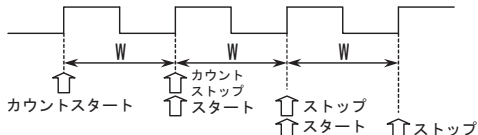
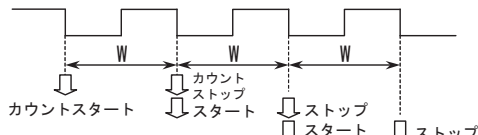
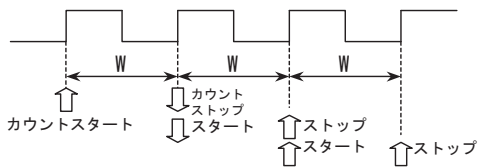
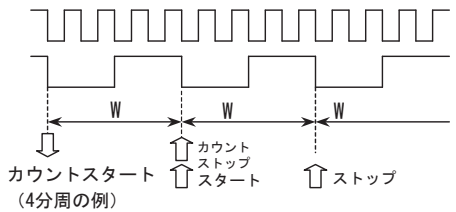
測定モード	MOD2	MOD1	MOD0	測定モード測定内容 (W: 測定するパルス幅)
"H" パルス幅測定	1	0	1	 <p>"H" 期間の幅を測定します。 カウント (測定) 開始: 立上りエッジ検出時 カウント (測定) 終了: 立下りエッジ検出時</p>
"L" パルス幅測定	1	1	0	 <p>"L" 期間の幅を測定します。 カウント (測定) 開始: 立下りエッジ検出時 カウント (測定) 終了: 立上りエッジ検出時</p>
立上りエッジ間 周期測定	1	0	0	 <p>立上りエッジ間の周期を測定します。 カウント (測定) 開始: 立上りエッジ検出時 カウント (測定) 終了: 立上りエッジ検出時</p>
立下りエッジ間 周期測定	1	1	1	 <p>立下りエッジ間の周期を測定します。 カウント (測定) 開始: 立下りエッジ検出時 カウント (測定) 終了: 立下りエッジ検出時</p>

表 15.3-6 測定モード一覧 (2 / 2)

測定モード	MOD2	MOD1	MOD0	測定モード測定内容 (W: 測定するパルス幅)
全エッジ間 パルス幅測定	0	1	0	 <p>連続して入力されるエッジ間の幅を測定します。 カウント (測定) 開始: エッジ検出時 カウント (測定) 終了: エッジ検出時</p>
分周周期測定	0	1	1	 <p>分周比設定レジスタ DIVR で選択した分周比だけ 入力パルスを分周してその周期を測定します。 カウント (測定) 開始: 起動直後の立上りエッジ検出時 カウント (測定) 終了: 分周後の 1 周期終了時</p>

どのモードでも、カウントの開始からカウント開始エッジの入力までの間はタイマによるカウントは行われません。カウント開始エッジ入力後に、タイマは "0000_H" にクリアされ、カウント終了エッジが入力されるまでは、カウントクロックのたびにタイマのカウントが行われます。カウント終了エッジの入力時には、次の動作を実行します。

1. PWCSR 中の測定終了フラグ (EDIR) がセットされます。
2. タイマのカウント動作が停止します (再起動と同時であった場合を除く)。
3. 連続測定モード: タイマの値 (測定結果) は PWCR に転送され、次の測定開始エッジが入力されるまでカウントを中断したままとなります。
4. 単発測定モード: 測定を終了します (再起動と同時であった場合を除く)。

連続測定モードの場合で、全エッジ間パルス幅測定や分周測定などを行った場合、終了エッジが次の測定エッジとなります。

■ 最小入力パルス幅

パルス幅測定入力端子 (PWC) に入力できるパルスには、以下の制限があります。

パルス幅は、4 マシンサイクル (24 MHz のマシクロックの場合は、0.17 μ s) 以上とすること。

■ パルス幅 / 周期算出方法

測定するパルスの幅 / 周期は、次の式で算出できます。

$$T_W = n \times t \div D_{IV} [\mu s]$$

T_W : 被測定パルス幅 / 周期 [μs]

n : PWCR 内の測定結果データ

t : カウントクロックの周期 [μs]

D_{IV} : 分周比レジスタ DIVR で選択した分周比

(分周周波数測定モード以外は 1 を代入)

■ パルス幅 / 周期カウントの範囲

カウントクロックおよび入力分周器の分周比の選択の組合せにより、測定可能なパルス幅 / 周期のレンジが変化します。

表 15.3-7 に、マシニングロック (以下 ϕ とする) = 24 MHz 時の測定レンジ一覧を示します。

表 15.3-7 パルス幅測定レンジ一覧

分周比	DIV1	DIV0	CKS1, CKS0=00 ($\phi/4$) 時	CKS1, CKS0=01 ($\phi/16$) 時	CKS1, CKS0=10 ($\phi/32$) 時
分周なし			0.17 μs ~ 10.92 ms [0.25 μs]	0.17 μs ~ 43.7 ms [1.6 μs]	0.17 μs ~ 87.4 ms [3.2 μs]
4 分周	0	0	0.17 μs ~ 2.73 ms [6.25 μs]	0.17 μs ~ 10.92 ms [0.4 μs]	0.17 μs ~ 21.85 ms [0.8 μs]
16 分周	0	1	0.17 μs ~ 683 μs [15.6 ns]	0.17 μs ~ 2.73 ms [1.6 μs]	0.17 μs ~ 5.46 ms [0.2 μs]
64 分周	1	0	0.17 μs ~ 170 μs [3.91 μs]	0.17 μs ~ 683 μs [25 μs]	0.17 μs ~ 1.36 ms [50 ns]
256 分周	1	1	0.17 μs ~ 43 μs [0.98 μs]	0.17 μs ~ 171 μs [6.25 μs]	0.17 μs ~ 341 μs [12.5 ns]

[] 内は、1 ビット当たりの分解能を示します。

■ 割込み要求発生

パルス幅測定モードにおいては、以下の 2 つの割込み要求を発生できます。

● タイマのオーバフローによる割込み要求

測定中、カウントアップによってオーバフローが発生すると、オーバフローフラグがセットされます。オーバフロー割込み要求が許可されていると、割込み要求が発生します。

● 測定終了による割込み要求

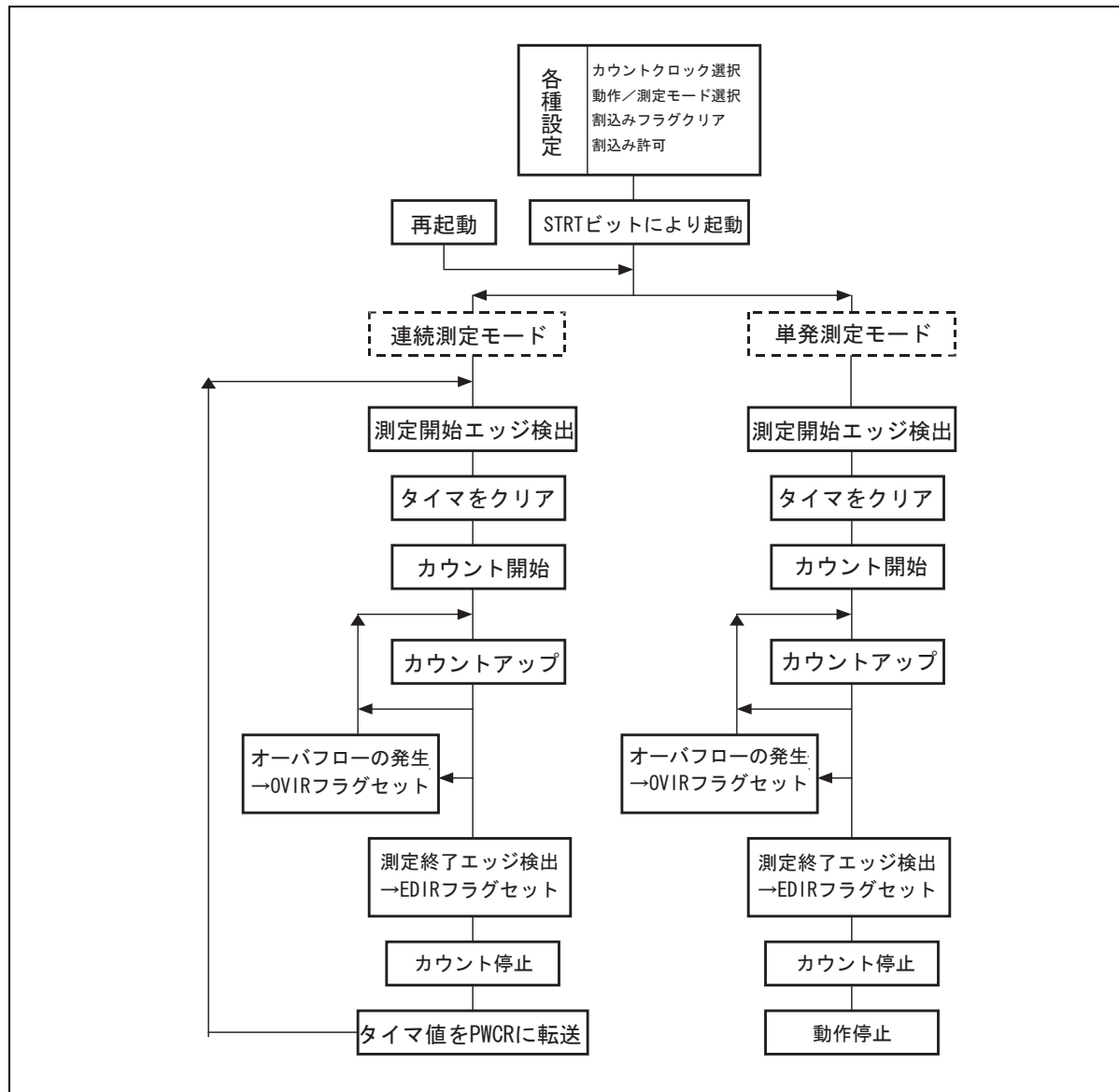
測定終了エッジを検出すると PWCSR 中の測定終了フラグ (EDIR) がセットされ、測定終了割込み要求が許可されていると割込み要求が発生します。

測定終了フラグ (EDIR) は、測定結果 PWCR を読み出すと同時に自動的にクリアされます。

■ パルス幅測定 of 動作フロー

図 15.3-5 に、パルス幅測定 of 動作フローを示します。

図 15.3-5 パルス幅測定 of 動作フロー



15.4 PWC タイマの使用上の注意

PWC タイマを使用する上で注意すべきことについて説明します。

■ PWC タイマの使用上の注意

● レジスタの書換えに関する注意事項

PWCSR 中、以下に示すビットは動作中に書き換えることを禁止します。書換えは、必ず起動前か停止後に行ってください。

- [bit7, bit6] CKS1, CKS0(クロック選択)
- [bit5, bit4] PIS1, PIS0(入力信号選択)
- [bit3] S/C (測定モード (単発 / 連続) の選択)
- [bit2 ~ bit0] MOD2, MOD1, MOD0(動作モード / 測定エッジ選択)

DIVR は、動作中に書き換えることを禁止します。書換えは必ず起動前か停止後に行ってください。

● タイマモードの測定終了フラグの扱い

PWCSR 中の測定終了割込み要求フラグ (EDIR) の値は、タイマモードでは、意味がありません。このため、PWCSR 中の測定終了割込み要求ビット (EDIE) は、タイマモードで使用するときは必ず "0" を設定してください。

● PWCSR 中の STRT, STOP ビットの扱い

両ビットともに、書込み時と読出し時では、意味が異なるので注意してください (「15.2.1 PWC コントロールステータスレジスタ (PWCSR)」を参照)。また、リードモディファイライト (RMW) 系命令における読出し値は、ビット値にかかわらず "11_B" です。このため、動作状態の読出しには、ビット処理命令は使用できません (読み出すと必ず動作中となります) ので注意してください。タイマの起動 / 停止のための STRT, STOP ビットへの書込みは、それぞれのビットに対するビット処理命令 (ビットクリア命令など) を用いることができます。

● タイマのクリア

パルス幅測定モードの場合、測定開始エッジでタイマがクリアされますので、起動前にタイマ中のデータは無効となります。

● モード変更時の PWCR とタイマの値

- リロードタイマモードで使用した後、タイマを強制停止してからワンショットタイマモードにした場合、PWCR に保持された値およびタイマの値は不定となりますので、必ず値を書き込んでから使用してください。
- ワンショットタイマモードで使用した場合、PWCR の値は不定となりますので、必ず値を書き込んでから使用してください。
- パルス幅測定モードから、タイマモードへ切り替える場合は、起動前に必ず PWCR に値を再設定してから起動してください。

● 最小パルス幅

パルス幅測定入力端子に入力できるパルスには、以下の制限があります。

- 最小パルス幅：マシクロックの 2 分周 (16 MHz マシクロック時, 0.25 μ s 以上)
 - 最小入力周波数：マシクロックの 4 分周 (16 MHz マシクロック時, 4 MHz 以下)
- 上記パルスより小さい幅や高い周波数のパルスを入力した場合、動作は保証できません。入力信号にそのようなノイズがのる可能性がある場合は、チップ外部でフィルタなどを通して除去してから入力してください。

● 分周周波数測定モード

パルス幅測定モードのうちの分周周期測定モードでは、入力パルスを分周するため、測定結果より算出して得られるパルス幅は、平均値となりますので注意してください。

● クロック選択ビットの扱い

PWCSR 中の [bit7, bit6] CKS1, CKS0(クロック選択) には, "11_B" は設定しないでください。

● 動作中の再起動

カウント動作を開始した後に再起動をする場合は、そのタイミングによっては以下に示すようなことが起こり得ます。

- リロードタイマモード時、オーバフロー発生と同時タイミングであった場合、再起動を行いますが、オーバフローフラグ (OVIR) はセットされます。
- パルス幅単発測定モードにおいて、測定終了エッジと同時であった場合、再起動を行い測定開始エッジ待ち状態となりますが、測定終了エッジ (EDIR) はセットされず。
- パルス幅連続測定モードにおいて、測定終了エッジと同時であった場合、再起動を行い測定開始エッジ待ち状態となりますが、測定終了エッジ (EDIR) はセットされ、その時点での測定結果は PWCR に転送されます。

● PWC タイマを「"H" パルス幅または "L" パルス幅測定モードを連続測定モード」で使用している場合

パルス幅測定が完了し、次のパルス幅測定開始待ち状態になっている間、本来停止すべきタイマが動作し続け、次のパルス幅測定開始前にタイマのオーバフローフラグ (OVIR) がセットされてしまうことがあるため、次回パルス幅測定終了時にオーバフローしていない場合でもオーバフローフラグがセットされている場合があります。したがって、PWC タイマを "H" パルス幅または "L" パルス幅測定モードを連続測定モードでご使用になる場合には、オーバフローフラグを使用しないでください。

第 16 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概要、レジスタの構成 / 機能および 16 ビットリロードタイマの動作について説明します。

16.1 16 ビットリロードタイマの概要

16.2 16 ビットリロードタイマのレジスタ

16.3 16 ビットリロードタイマの動作

16.1 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は、3 種類の内部クロックに同期してカウントダウンする内部クロックと、外部端子に入力されたパルスの任意エッジを検出してカウントダウンするイベントカウントモードがあり、どちらかの機能を選択できます。

■ 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は、カウント値が "0000_H" ~ "FFFF_H" になるときをアンダフローと定義しています。したがって、[リロードレジスタの設定値 +1] カウントでアンダフローが発生します。カウント動作としては、アンダフローで、カウント設定値を再ロードしてカウントを繰り返すリロードモードと、アンダフローでカウントを停止するワンショットモードを選択できます。カウンタのアンダフローで割込みを発生させることもでき、DTC にも対応しています。

16.1.1 16 ビットリロードタイマの機能

16 ビットリロードタイマの機能概要を説明します。

■ 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	16 ビットリロードタイマ動作
内部クロック	リロードモード	ソフトウェアトリガ動作 外部トリガ動作 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

カウントクロックとして 3 種類の内部クロックから 1 種類を選択できます。

● 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 ~ TMCSR2) の TRG ビットに "1" を書き込むと、カウンタ動作を開始します。TRG ビットによるトリガ入力は、外部トリガ入力時および外部ゲート入力時にも有効です。

● 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/TIN2 端子に入力されたとき、カウント動作を開始します。

● 外部ゲート入力動作

選択された信号 ("L" または "H") が TIN0/TIN1/TIN2 に入力されている間、カウント動作を続行します。

■ イベントカウントモード (外部クロックモード)

選択されたエッジ (立上り / 立下り / 両エッジ) が TIN0/TIN1/TIN2 端子に入力されると、そのエッジでカウントダウンする機能です。

一定周期の外部クロックを使用した場合は、インターバルタイマとしても使用できます。

■ カウンタ動作のモード

● 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、カウント設定値が再ロードされてカウント動作を継続します。アンダフローで、割込み要求を発生させることができるので、インターバルタイマとして使用できます。また、アンダフローごとに、反転するトグル波形を TOT0/TOT1/TOT2 端子から出力できます。

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
内部クロック	$2^1/\phi$ (0.083 μ s)	0.083 μ s ~ 5.461 ms
	$2^3/\phi$ (0.33 μ s)	0.33 μ s ~ 21.845 ms
	$2^5/\phi$ (1.33 μ s)	1.33 μ s ~ 87.38 ms
外部クロック	$2^3/\phi$ (0.5 μ s)	0.5 μ s 以上

ϕ : マシンクロック () 内はマシンクロック 24 MHz 動作時の値です。

● ワンショットモード

カウントダウンで、アンダフロー(0000_H ~ FFFF_H)が発生すると、カウント動作を停止します。

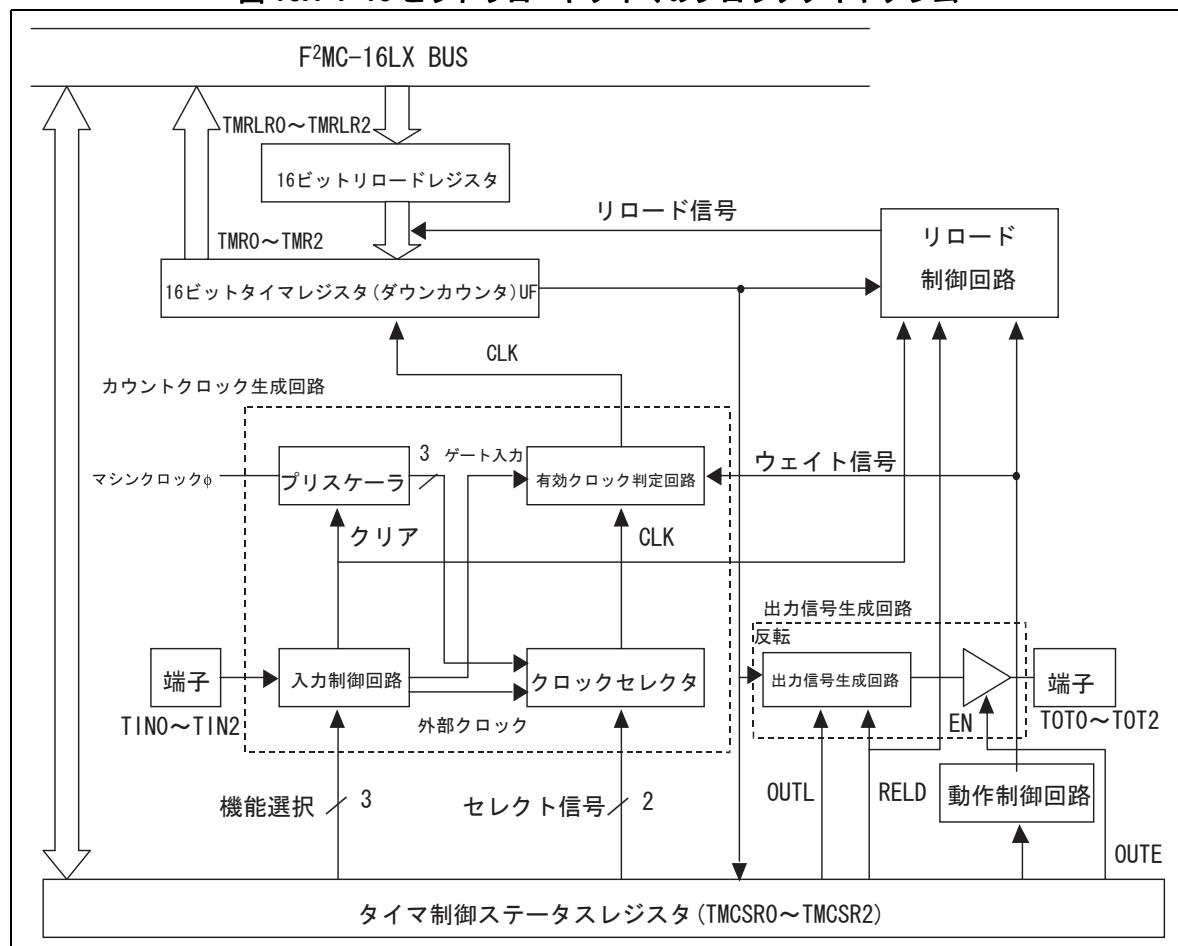
アンダフローで割込みを発生させることができます。また、カウントの動作中は、カウント中を示す短波形を TOT0/TOT1/TOT2 端子から出力できます。

16.1.2 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のブロックダイアグラムを示します。

■ 16 ビットリロードタイマのブロックダイアグラム

図 16.1-1 16 ビットリロードタイマのブロックダイアグラム



16.2 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で使用するレジスタの構成および機能について説明します。

■ 16 ビットリロードタイマレジスタ一覧

図 16.2-1 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 16.2-1 16 ビットリロードタイマのレジスタ一覧

bit	15	14	13	12	11	10	9	8	TMCSR0 ~ TMCSR2
ch.0 : 000063 _H	-	-	-	-	CSL1	CSL0	MOD2	MOD1	タイマ制御ステータスレジスタ (上位)
ch.1 : 000067 _H	(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	読出し / 書込み
ch.2 : 00006B _H	(X)	(X)	(X)	(X)	(0)	(0)	(0)	(0)	初期値
bit	7	6	5	4	3	2	1	0	TMCSR0 ~ TMCSR2
ch.0 : 000062 _H	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG	タイマ制御ステータスレジスタ (下位)
ch.1 : 000066 _H	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	読出し / 書込み
ch.2 : 00006A _H	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	初期値
bit	15	14	13	12	11	10	9	8	TMR0~TMR2/ TMRLR0~TMRLR2
ch.0 : 000065 _H	D15	D14	D13	D12	D11	D10	D09	D08	16 ビットタイマレジスタ / 16 ビットリロードレジスタ (上位)*
ch.1 : 000069 _H	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	読出し (TMR0~TMR2)
ch.2 : 00006D _H	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	書込み (TMRLR0~TMRLR2)
	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	初期値
bit	7	6	5	4	3	2	1	0	TMR0~TMR2/ TMRLR0~TMRLR2
ch.0 : 000064 _H	D07	D06	D05	D04	D03	D02	D01	D00	16 ビットタイマレジスタ / 16 ビットリロードレジスタ (下位)*
ch.1 : 000068 _H	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	読出し (TMR0~TMR2)
ch.2 : 00006C _H	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	書込み (TMRLR0~TMRLR2)
	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	初期値

* : 読出し時は 16 ビットタイマレジスタ (TMR0 ~ TMR2) として機能し、
書込み時は 16 ビットリロードレジスタ (TMRLR0 ~ TMRLR2) として機能します。

16.2.1 タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2)

タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2) の構成および機能について説明します。

■ タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2)

タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2) は、16 ビットリロードタイマの動作モードおよび割込みを制御します。CNTE=0 のとき、UF/CNTE/TRG 以外のビットをモディファイします。

図 16.2-2 に、タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2) のビット構成を示します。

図 16.2-2 タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2) のビット構成

bit	15	14	13	12	11	10	9	8	TMCSR0 ~ TMCSR2
ch.0 : 000063 _H	-	-	-	-	CSL1	CSL0	MOD2	MOD1	タイマ制御ステータスレジスタ (上位)
ch.1 : 000067 _H	(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
ch.2 : 00006B _H	(X)	(X)	(X)	(X)	(0)	(0)	(0)	(0)	読出し / 書込み初期値
bit	7	6	5	4	3	2	1	0	TMCSR0 ~ TMCSR2
ch.0 : 000062 _H	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG	タイマ制御ステータスレジスタ (下位)
ch.1 : 000066 _H	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
ch.2 : 00006A _H	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	読出し / 書込み初期値

以下に、タイマ制御ステータスレジスタ 0 ~ 2(TMCSR0 ~ TMCSR2) の各ビットの機能を説明します。

[bit15 ~ bit12] 未定義ビット

読出し値は不定です。書込みは動作に影響を与えません。

[bit11, bit10] CSL1, CSL0(クロック選択)

カウントクロック選択ビットで、クロックソースを選択します。

CSL1	CSL0	クロックソース (マシクロック $\phi=24$ MHz 時)
0	0	$\phi/2^1(0.083 \mu\text{s})$ [初期値]
0	1	$\phi/2^3(0.33 \mu\text{s})$
1	0	$\phi/2^5(1.33 \mu\text{s})$
1	1	イベントカウントモード

[bit9 ~ bit7] MOD2 ~ MOD0

動作モードと入出力端子の機能を設定するビットです。MOD2=0 のとき、入力端子はトリガとして機能します。アクティブエッジが入力端子に入力されカウント動作が進行すると、リロードレジスタの内容がカウンタにロードされます。MOD2=1 のときは、タイマはゲートカウントモードで動作し、入力端子はゲート入力として機能します。このモードでは、カウントはアクティブレベルが入力端子に入力されている間だけカウントします。

MOD2 ~ MOD0 ビットの組合せによって、内部クロックモードおよびイベントカウントモードを表 16.2-1 および表 16.2-2 に示すモードから選択します。

表 16.2-1 内部クロックモード (CSL0, CSL1=00, 01 または 10)

MOD2	MOD1	MOD0	入力端子機能	アクティブエッジまたはレベル
0	0	0	トリガ無効	- [初期値]
0	0	1	トリガ入力	立上りエッジ
0	1	0		立下りエッジ
0	1	1		両エッジ
1	X	0	ゲート入力	"L" レベル
1	X	1		"H" レベル

表 16.2-2 イベントカウントモード (CSL0, CSL1=11)

MOD2	MOD1	MOD0	入力端子機能	アクティブエッジまたはレベル
X	0	0	トリガ無効	- [初期値]
X	0	1	トリガ入力	立上りエッジ
X	1	0		立下りエッジ
X	1	1		両エッジ

[bit6] OUTE(出力許可)

出力許可を制御します。

TOT0 ~ TOT2 端子は、"0" のとき汎用ポート、"1" のときタイマ出力端子として機能します。リロードモードでは、出力波形はトグル波形となります。ワンショットモードでは、TOT0 ~ TOT2 端子は、カウントが進行中であることを示す矩形波を出力します。

OUTE	機能
0	汎用ポート [初期値]
1	タイマ出力

[bit5] OUTL(出力レベルの設定)

TOT0 ~ TOT2 端子の出力レベルを設定するビットです。OUTL が、0/1 では出力端子レベルは逆になります。

OUTL	ワンショットモード時(RELD=0)	リロードモード時 (RELD=1)
0	カウント中 "H" の短形波	0 [初期値]
1	カウント中 "L" の短形波	1
X	1	0
X	1	1

[bit4] RELD(リロード動作許可)

リロード動作を許可するビットです。RELD が "1" のときは、タイマはリロードモード動作で動作します。このモードでは、タイマは、リロードレジスタの内容をカウンタにロードし、アンダフローが発生しても (カウンタ値 0000_H ~ FFFF_H に変化するとき) カウンタを継続します。RELD が "0" のときは、タイマはワンショットモードで動作します。このモードでは、カウンタ値が 0000_H ~ FFFF_H に変化するときアンダフローが発生するとカウンタ動作を停止します。

RELD	機能
0	ワンショットモード [初期値]
1	リロードモード

[bit3] INTE(タイマ割込み要求許可)

タイマ割込み要求を許可するビットです。INTE=0 のときは、UF が "1" になっても、割込み要求を発生しません。

INTE	機能
0	割込み要求出力禁止 [初期値]
1	割込み要求出力許可

[bit2] UF(タイマ割込み要求フラグ)

タイマ割込み要求フラグです。アンダフローが発生すると UF は "1" に設定されます。"0" を書き込むか、μDMAC によってクリアされます。"1" を書き込んでも意味をもちません。リードモディファイライト命令での読出し値は "1" です。

UF	読出し時	書込み時
0	カウンタのアンダフローなし (初期値)	このビットのクリア [初期値]
1	カウンタのアンダフローあり	変化なし (ほかへの影響なし)

[bit1] CNTE(タイマカウンタ許可)

タイマカウンタを許可するビットです。

CNTE	機能
0	カウンタ停止 [初期値]
1	カウンタ許可 (起動トリガ待ち)

[bit0] TRG(ソフトウェアトリガ)

ソフトウェアトリガビットです。TRG=1 の書込みで、ソフトウェアトリガを適用してタイマのリロードレジスタの内容をカウンタロードさせ、カウントを開始させます。"0" を書き込んでも何の意味ももちません。読出しは常に "0" です。CNTE=1 のときのみ、動作モードにかかわらず常に有効になります。

TRG	機能
0	変化なし (ほかへの影響なし) [初期値]
1	カウント動作開始

16.2.2 16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/ 16 ビットリロードレジスタ 0 ~ 2(TMRLR0 ~ TMRLR2)

16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2) の構成および機能について説明します。

■ 16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2)

図 16.2-3 に、16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/16 ビットリロードレジスタ (TMRLR0 ~ TMRLR2) のビット構成を示します。

図 16.2-3 16 ビットタイマレジスタ 0 ~ 2 (TMR0 ~ TMR2)/
16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2) のビット構成

bit	15	14	13	12	11	10	9	8	TMR0~TMR2/ TMRLR0~TMRLR2
ch.0 : 000065 _H	D15	D14	D13	D12	D11	D10	D09	D08	16 ビットタイマレジスタ /16 ビットリロードレジスタ (上位)*
ch.1 : 000069 _H	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
ch.2 : 00006D _H	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	初期値
bit	7	6	5	4	3	2	1	0	TMR0~TMR2/ TMRLR0~TMRLR2
ch.0 : 000064 _H	D07	D06	D05	D04	D03	D02	D01	D00	16 ビットタイマレジスタ /16 ビットリロードレジスタ (下位)*
ch.1 : 000068 _H	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
ch.2 : 00006C _H	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	初期値

* : 読出し時は 16 ビットタイマレジスタ (TMR0 ~ TMR2) として機能し、
書込み時は 16 ビットリロードレジスタ (TMRLR0 ~ TMRLR2) として機能します。

■ 16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)

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るレジスタです。カウンタ動作を許可 (TMCSR0 ~ TMCSR2 の CNTE=1) し、カウントを開始すると、16 ビットリロードレジスタに書き込まれた値が TMR0 ~ TMR2 にロードされ、カウントダウンが開始されます。カウント停止状態 (TMCSR0 ~ TMCSR2 の CNTE=0) では、TMR0 ~ TMR2 レジスタの値が保持されます。

< 注意事項 >

TMR0 ~ TMR2 の読出しはカウンタ動作中にもできますが、必ずワード転送命令 (MOVW A 003AH など) を使用してください。

16 ビットタイマレジスタ (TMR0 ~ TMR2) は、機能的には、読出し専用ですが、書込み専用の 16 ビットリロードレジスタ (TMRLR0 ~ TMRLR2) と同一アドレスに配置されています。したがって、書込みをしても TMR0 ~ TMR2 の値は影響を受けませんが、TMRLR0 ~ TMRLR2 に対して書込みが行われます。

■ 16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2)

16 ビットリロードタイマの動作に関係なく、カウンタ動作を禁止 (TMCSR0 ~ TMCSR2 の CNTE=0) している状態で、TMRLR0 ~ TMRLR2 にカウンタの初期値を設定します。カウンタ動作を許可 (TMCSR0 ~ TMCSR2 の CNTE=1) し、カウンタを起動すると、TMRLR0 ~ TMRLR2 に書き込まれた値からカウントダウンを開始します。TMRLR0 ~ TMRLR2 に設定した値は、リロードモードのとき、アンダフローの発生でカウンタにリロードされ、カウントダウンを続行します。ワンショットモードのときは、アンダフローの発生でカウンタは "FFFF_H" で停止します。

< 注意事項 >

TMRLR0 ~ TMRLR2 の書込みは、カウンタ停止 (TMCSR0 ~ TMCSR2 の CNTE=0) の状態で行ってください。

また、必ずワード転送命令 (MOVW A 003AH など) で書き込んでください。

16 ビットリロードレジスタ (TMRLR0 ~ TMRLR2) は、機能的には書込み専用ですが、読出し専用の 16 ビットタイマレジスタ (TMR0 ~ TMR2) 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 ~ TMR2 の値となるため、INC/DEC 命令などの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。

16.3 16 ビットリロードタイマの動作

16 ビットリロードタイマの設定と , カウンタ動作の状態遷移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 16.3-1 に示す設定が必要です。

図 16.3-1 内部クロックモードの設定

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMCSR0	-	-	-	-	CSL1	CSL0	MOD2	MOD1	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG
TMCSR2					"11" 以外		⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	1	⊙
TMRLR0 ~ TMRLR2	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

イベントカウントモードとして動作させるには , 図 16.3-2 に示す設定が必要です。

図 16.3-2 イベントカウントモードの設定

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMCSR0	-	-	-	-	CSL1	CSL0	MOD2	MOD1	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG
TMCSR2					1	1	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	1	⊙

TMRLR0
~
TMRLR2

カウンタの初期値 (リロード値) を設定

DDR3

D37D36D35D34D33D32D31D30

△△

DDR4

D47D46D45D44D43D42D41D40

△

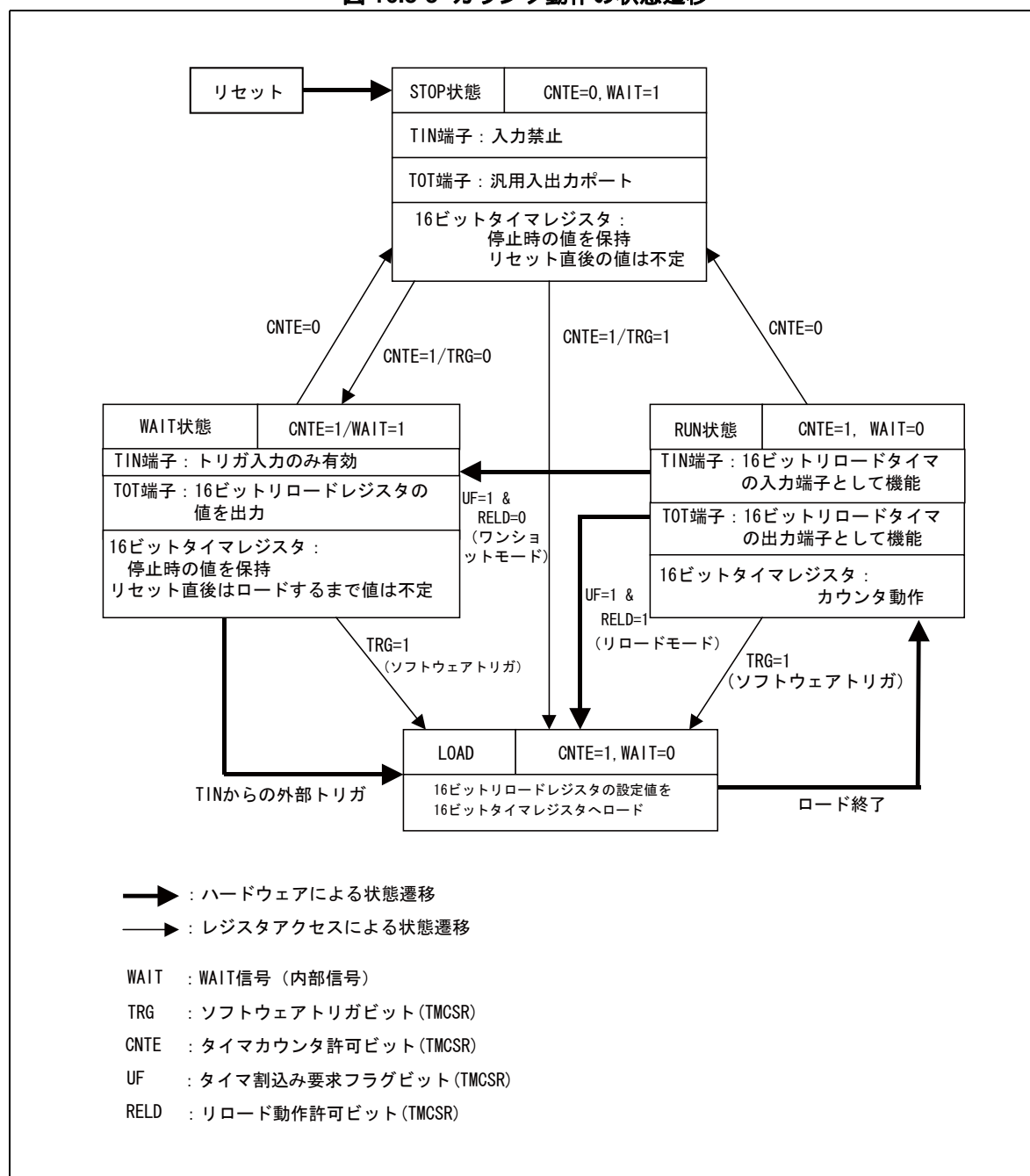
⊙ : 使用ビット
1 : "1" を設定
△ : 使用する端子に対応するビットに "0" を設定

16.3.1 カウンタ動作の状態遷移

カウンタ動作の状態遷移を示します。

■ カウンタ動作の状態遷移

図 16.3-3 カウンタ動作の状態遷移



16.3.2 内部クロックモード (リロードモード) の動作

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で CPU に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

カウント動作を許可 (TMCSR の CNTE=1) し、ソフトウェアトリガビット (TMCSR の TRG)、または外部トリガでタイマを起動すると、リロードレジスタ (TMRLR) の値をカウンタにロードして、カウンタ動作を開始します。

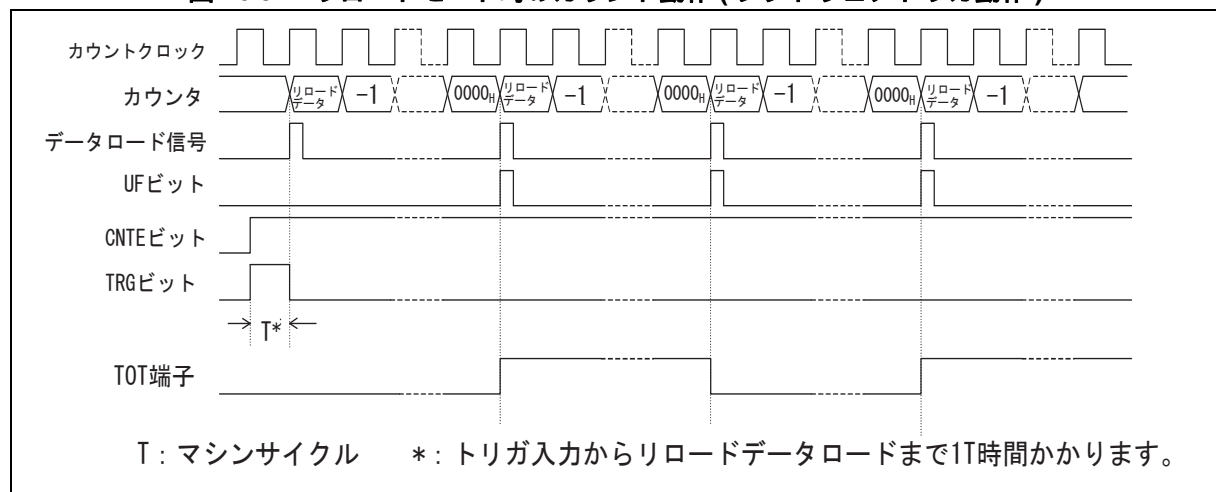
カウンタ許可ビットとソフトウェアトリガビットを同時に "1" にセットすると、カウンタ許可と同時にカウントを開始します。カウンタ値がアンダフロー ("0000_H" "FFFF_H") すると、16 ビットリロードレジスタ (TMRLR) の値をカウンタにロードして、カウント動作を継続します。このとき、アンダフロー割込み要求フラグビット (UF) を "1" にセットし、割込み要求許可ビット (INTE) が "1" であれば割込み要求を発生します。また、TOT 端子からは、アンダフローごとに反転するトグル波形を出力できます。

● ソフトウェアトリガの動作

タイマ制御ステータスレジスタ (TMCSR) の TRG ビットに "1" を書き込むと、カウンタを起動します。

図 16.3-4 にリロード時のソフトウェアトリガの動作を示します。

図 16.3-4 リロードモード時のカウント動作 (ソフトウェアトリガ動作)

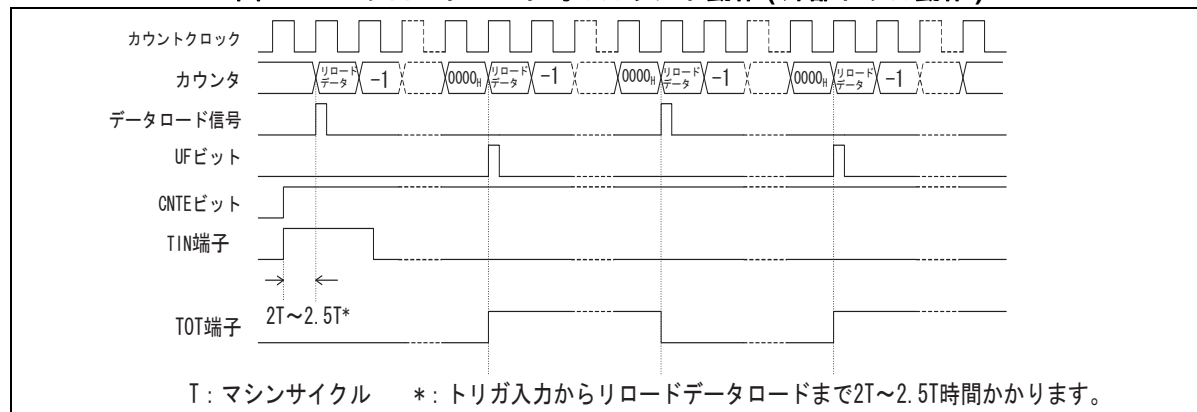


● 外部トリガの動作

有効エッジ（立上り，立下り，両エッジ選択可能）が TIN 端子に入力されると，カウンタを起動します。

図 16.3-5 にリロードモード時の外部トリガの動作を示します。

図 16.3-5 リロードモード時のカウント動作（外部トリガ動作）



< 注意事項 >

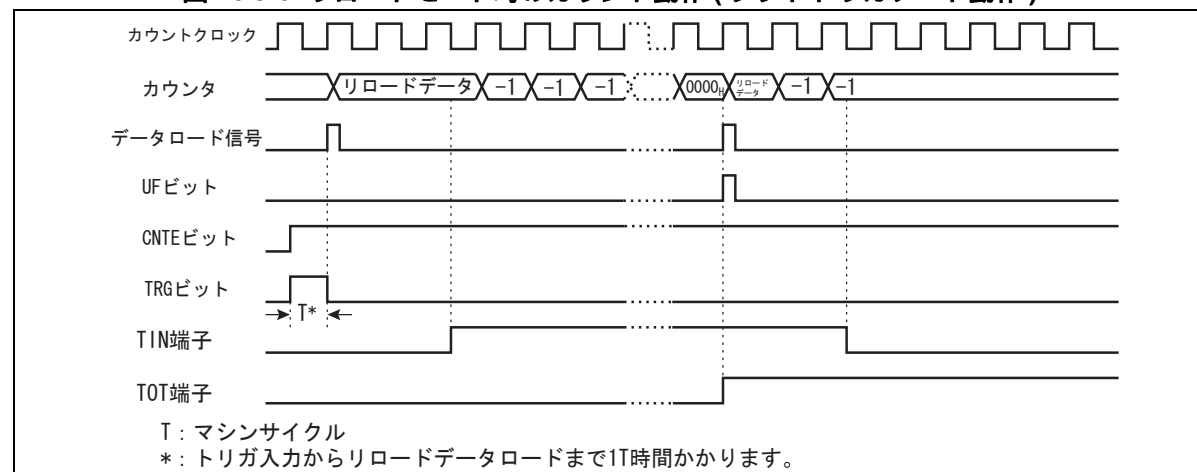
TIN 端子へ入力するトリガパルス幅は， $2/\phi$ 以上としてください。

● ゲート入力の動作

タイマコントロールステータスレジスタ (TMCSR) のカウント許可ビット (CNTE) に "1" を設定している場合に，ソフトウェアトリガビット (TRG) に "1" を設定すると，カウント動作を開始します。

動作モード設定ビット (MOD2, MOD1, MOD0) で設定しているゲート入力の有効レベル ("L", "H" から設定可能) が TIN 端子に入力されている間，カウント動作を行います。

図 16.3-6 リロードモード時のカウント動作（ソフトトリガゲート動作）



< 注意事項 >

TIN 端子へ入力するトリガパルス幅は， $2/\phi$ 以上としてください。

16.3.3 内部クロックモード (ワンショットモード) の動作

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で CPU に割り込み要求を発生します。また、TOT 端子から、カウント中を示す矩形波を出力できます。

■ 内部クロックモード (ワンショットモード)

カウント動作を許可 (TMCSR の CNTE=1) し、ソフトウェアトリガビット (TMCSR の TRG) または外部トリガで、タイマ起動すると、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を同時に "1" にセットすると、カウント許可と同時にカウントを開始します。カウンタ値がアンダフロー ("0000_H" "FFFF_H") すると、"FFFF_H" の状態で、カウンタを停止します。このとき、アンダフロー割り込み要求フラグビット (UF) を "1" にセットし、割り込み要求許可ビット (INTE) が "1" であれば割り込み要求を発生します。

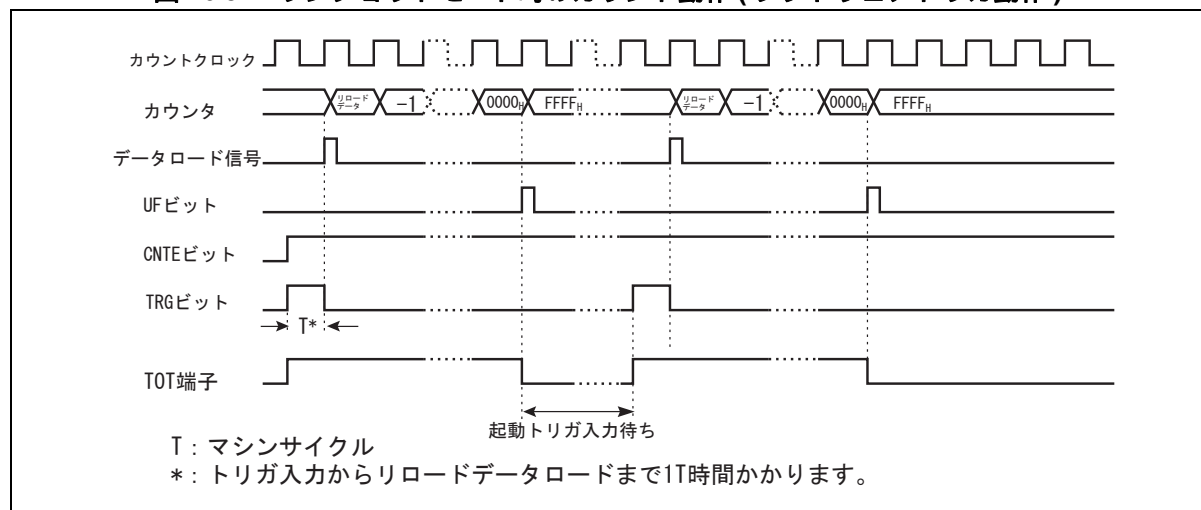
また、TOT 端子からはカウント中を示す矩形波を出力できます。

● ソフトウェアトリガの動作

タイマ制御ステータスレジスタ (TMCSR) の TRG ビットに "1" を書き込むことで、カウンタを起動します。

図 16.3-7 に、ワンショットモード時のソフトウェアトリガ動作を示します。

図 16.3-7 ワンショットモード時のカウント動作 (ソフトウェアトリガ動作)

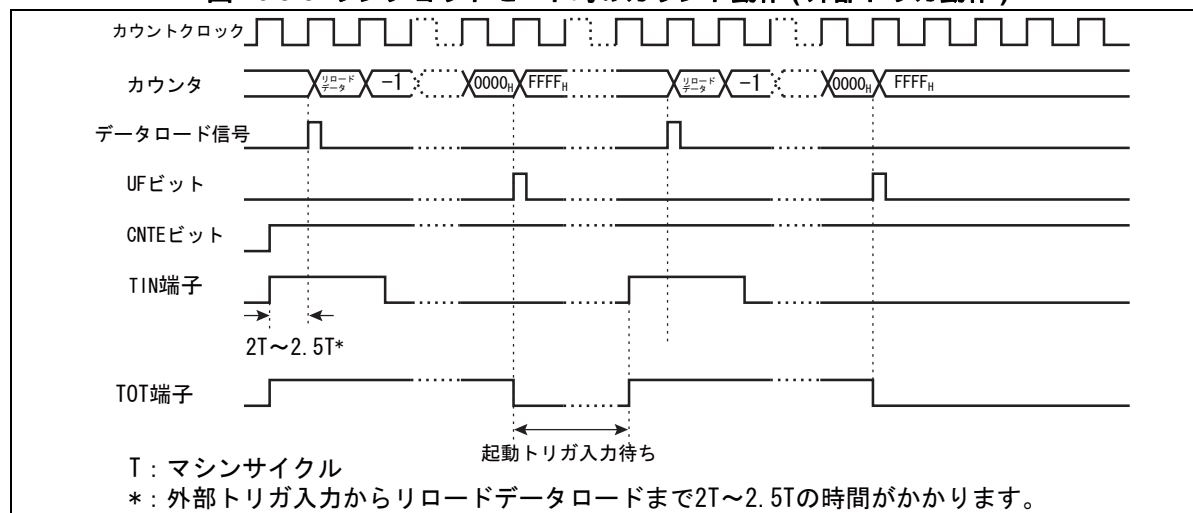


● 外部トリガの動作

有効エッジ（立上り、立下り、両エッジ選択可能）が TIN 端子に入力されると、カウンタを起動します。

図 16.3-8 に、ワンショットモード時の外部トリガ動作を示します。

図 16.3-8 ワンショットモード時のカウント動作（外部トリガ動作）



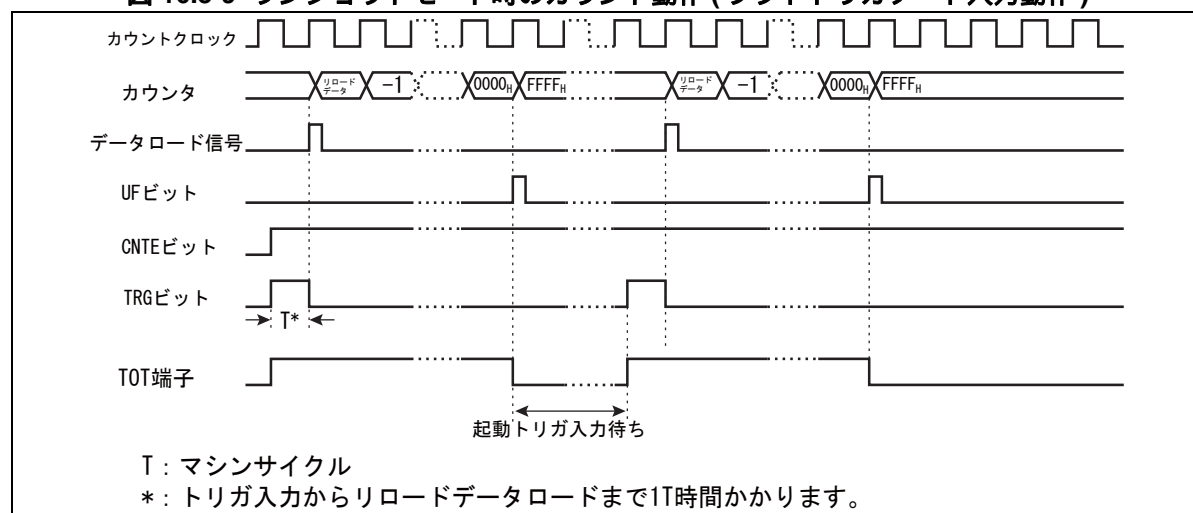
< 注意事項 >

TIN 端子へ入力するトリガパルス幅は、 $2/\phi$ 以上としてください。

● ゲート入力の動作

有効レベル（"H" レベル、"L" レベル選択可能）が TIN 端子に入力されている間、カウント動作を行います。図 16.3-9 に、ワンショットモード時のゲート入力動作を示します。

図 16.3-9 ワンショットモード時のカウント動作（ソフトトリガゲート入力動作）



< 注意事項 >

TIN 端子へ入力するゲート入力のパルス幅は、 $2/\phi$ 以上としてください。

16.3.4 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 ビットカウンタをカウントダウンし、カウンタのアンダフローで、CPU に割込み要求を発生します。また、TOT 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

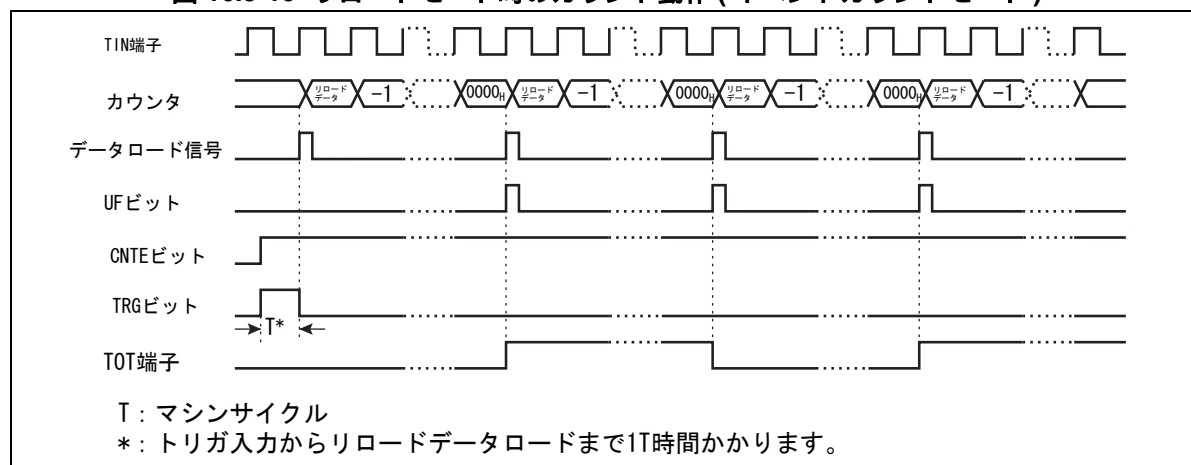
カウント動作を許可 (TMCSR の CNTE=1) し、カウンタを起動 (TMCSR の TRG=1) すると、16 ビットリロードレジスタ (TMRLR) の値がカウンタにロードされ、TIN 端子に入力されたパルス (外部カウントクロック) の、有効エッジ (立上り、立下り、両エッジ選択可能) を検出するたびにカウントダウンします。カウント許可ビットとソフトウェアトリガビットを同時に "1" にセットすれば、カウント許可と同時にカウントを開始します。

● リロードモード時の動作

カウンタ値がアンダフロー ("0000_H" "FFFF_H") すると、16 ビットリロードレジスタ (TMRLR) の値をカウンタにロードして、カウント動作を継続します。このとき、アンダフロー割込み要求フラグビット (UF) を "1" にセットし、割込み要求許可ビット (TMCSR の INTE) が "1" であれば割込み要求を発生します。また、TOT 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 16.3-10 に、リロードモードのカウント動作を示します。

図 16.3-10 リロードモード時のカウント動作 (イベントカウントモード)



< 注意事項 >

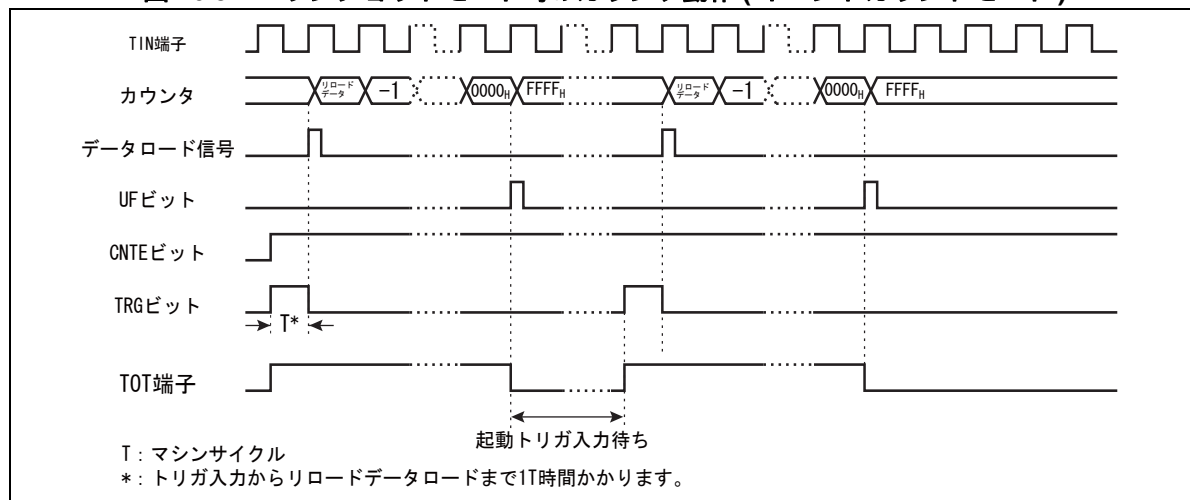
TIN 端子へ入力されるクロックの "H" 幅および "L" 幅は、4/φ 以上としてください。

● ワンショットモード時の動作

カウンタ値がアンダフロー ("0000_H" "FFFF_H") すると, "FFFF_H" の状態で, カウンタを停止します。このとき, アンダフロー要求フラグビット (UF) が "1" にセットされ, 割込み要求出力許可ビット (INTE) が "1" であれば割込み要求を発生します。また, TOT 端子からは, カウント中を示す矩形波を出力できます。

図 16.3-11 に, ワンショットモードのカウンタ動作を示します。

図 16.3-11 ワンショットモード時のカウンタ動作 (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, $4/\phi$ 以上としてください。

第17章

8/16 ビット PPG タイマ

この章では、8/16 ビット PPG タイマの概要、レジスタの構成 / 機能および 8/16 ビット PPG タイマの動作について説明します。

17.1 8/16 ビット PPG タイマの概要

17.2 8/16 ビット PPG タイマのレジスタ

17.3 8/16 ビット PPG タイマの動作

17.1 8/16 ビット PPG タイマの概要

8/16 ビット PPG タイマは、8 ビットのリロードタイマモジュールで、タイマ動作に応じたパルス出力により、PPG 出力を行います。

ハードウェアとして以下を有しています。

- 8 ビットダウンカウンタ 6 個
- 8 ビットリロードタイマ 12 個
- 16 ビット制御レジスタ 3 個
- 外部パルス出力端子 6 本
- 割込み出力 6 本

なお、MB90330A シリーズには、8 ビット PPG として 6 チャンネルあり、このうち PPG0+PPG1/PPG2+PPG3/ PPG4+PPG5 の組合せで 16 ビット PPG(3 チャンネル)としても動作します。

■ 8/16 ビット PPG タイマの概略仕様

以下に、8/16 ビット PPG タイマの機能概略を示します。

- 8 ビット PPG 出力 6 チャンネル独立動作モード
6 チャンネルの独立した PPG 出力動作が可能です。
 - 16 ビット PPG 出力動作モード
3 チャンネルの 16 ビットの PPG 出力動作が可能です。
PPG0+PPG1, PPG2+PPG3, PPG4+PPG5 の組合せで使用します。
 - 8+8 ビット PPG 出力動作モード
PPG0(PPG2/PPG4) の出力を PPG1(PPG3/PPG5) のクロック入力にすることによって、任意周期の 8 ビット PPG 出力が可能です。
 - PPG 出力動作
任意周期およびデューティ比のパルス波を出力します。
-

< 参考 >

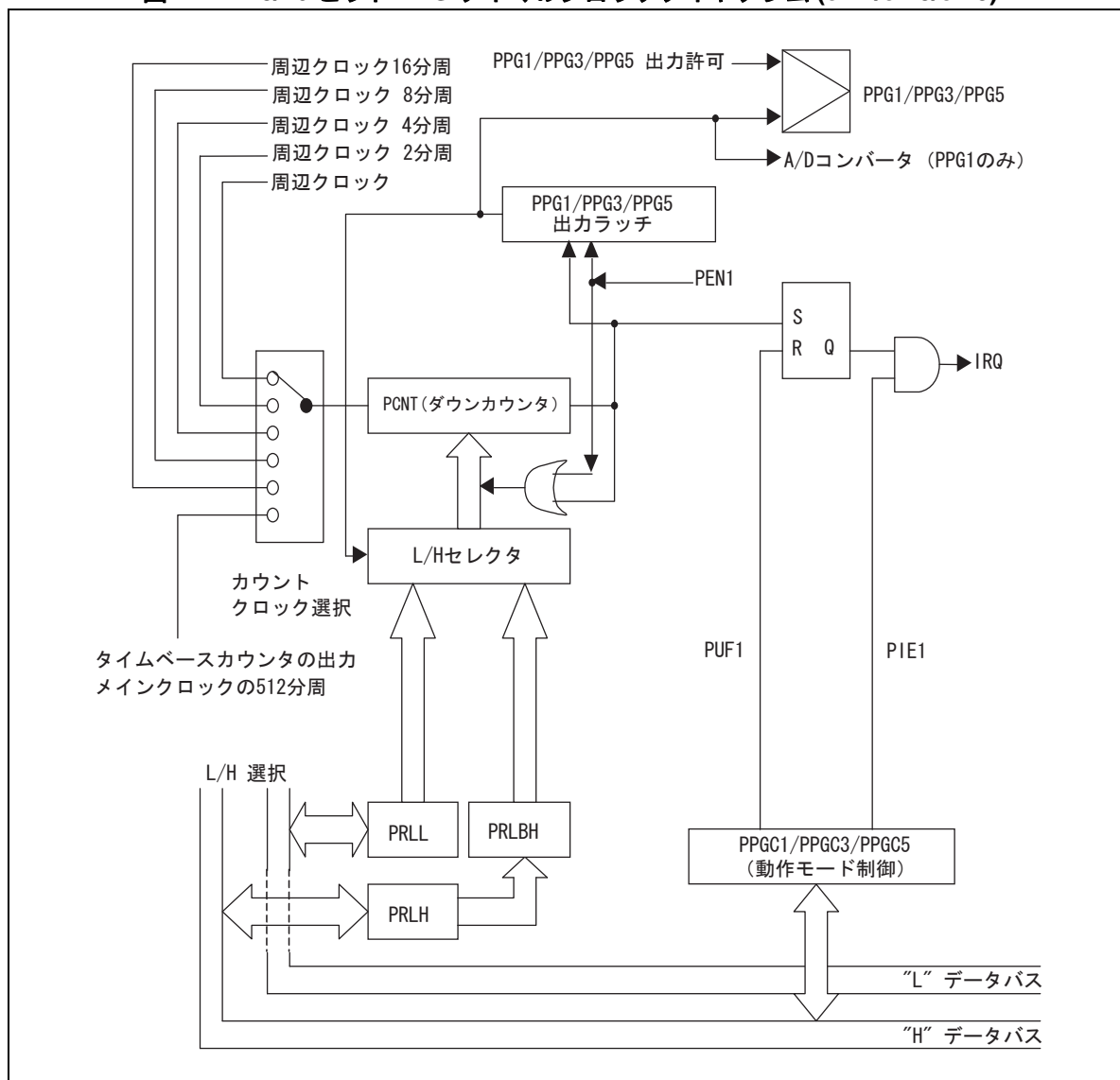
PPG1 のみ A/D コンバータの割込みトリガに使用できます。

8/16 ビット PPG タイマの ch.0/ch.2/ch.4 および ch.1/ch.3/ch.5 のブロックダイアグラムを示します。

図 17.1-1 に ch.0/ch.2/ch.4 のブロックダイアグラムを、図 17.1-2 に ch.1/ch.3/ch.5 のブロックダイアグラムを示します。

[illegible]

図 17.1-2 8/16 ビット PPG タイマのブロックダイアグラム (ch.1/ch.3/ch.5)



17.2 8/16 ビット PPG タイマのレジスタ

8/16 ビット PPG タイマで使用するレジスタの構成および機能について説明します。

■ 8/16 ビット PPG タイマのレジスタ一覧

図 17.2-1 に、8/16 ビット PPG タイマのレジスタ一覧を示します。

図 17.2-1 8/16 ビット PPG タイマのレジスタ一覧

		bit	7	6	5	4	3	2	1	0	PPGC0/PPGC2/PPGC4
ch.0 : 000046 _H			PEN0	-	PE00	PIE0	PUF0	-	-	予約	PPG 動作モード
ch.2 : 000048 _H			(R/W)	(-)	(R/W)	(R/W)	(R/W)	(-)	(-)	(R/W)	制御レジスタ
ch.4 : 00704A _H			(0)	(X)	(0)	(0)	(0)	(X)	(X)	(1)	リード/ライト 初期値
		bit	15	14	13	12	11	10	9	8	PPGC1/PPGC3/PPGC5
ch.1 : 000047 _H			PEN1	-	PE10	PIE1	PUF1	MD1	MD0	予約	PPG 動作モード
ch.3 : 000049 _H			(R/W)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	制御レジスタ
ch.5 : 00004B _H			(0)	(X)	(0)	(0)	(0)	(0)	(0)	(1)	リード/ライト 初期値
		bit	7	6	5	4	3	2	1	0	PPG01/PPG23/PPG45
ch.0,1 : 00004C _H			PCS2	PCS1	PCS0	PCM2	PCM1	PCM0	予約	予約	PPG 出力制御レジスタ
ch.2,3 : 00004E _H			(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	リード/ライト
ch.4,5 : 000050 _H			(0)	(0)	(0)	(0)	(0)	(0)	(X)	(X)	初期値
		bit	7	6	5	4	3	2	1	0	PRLL0~PRLL5
ch.0 : 007900 _H			D07	D06	D05	D04	D03	D02	D01	D00	PPG リロードレジスタ下位
ch.1 : 007902 _H			(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	リード/ライト
ch.2 : 007904 _H			(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	初期値
ch.3 : 007906 _H											
ch.4 : 007908 _H											
ch.5 : 00790A _H											
		bit	15	14	13	12	11	10	9	8	PRLH0~PRLH5
ch.0 : 007901 _H			D15	D14	D13	D12	D11	D10	D09	D08	PPG リロードレジスタ上位
ch.1 : 007903 _H			(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	リード/ライト
ch.2 : 007905 _H			(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	初期値
ch.3 : 007907 _H											
ch.4 : 007909 _H											
ch.5 : 00790B _H											

17.2.1 PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4)

PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4) の構成および機能について説明します。

■ PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4)

PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4) は、ch.0/ch.2/ch.4 の動作モード選択、端子出力制御、カウントクロック選択およびトリガの制御を行います。

図 17.2-2 に、PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4) のビット構成を示します。

図 17.2-2 PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4) のビット構成

	bit	7	6	5	4	3	2	1	0	PPGC0/PPGC2/PPGC4
ch.0 : 000046 _H		PEN0	-	PE00	PIE0	PUF0	-	-	予約	PPG 動作モード 制御レジスタ リード / ライト 初期値
ch.2 : 000048 _H		(R/W)	(-)	(R/W)	(R/W)	(R/W)	(-)	(-)	(R/W)	
ch.4 : 00704A _H		(0)	(X)	(0)	(0)	(0)	(X)	(X)	(1)	

以下に、PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4) の各ビットの機能を説明します。

[bit7] PEN0 (動作許可)

PPG0/PPG2/PPG4 の動作開始および動作モードを選択します。

PEN0	動作状態
0	動作停止 ("L" レベル出力保持)
1	PPG 動作許可

- このビットに "1" を書き込むことによって、PPG はカウントを開始します。
- このビットは、リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit6] 未定義ビット

リード値は不定です。ライトは動作に影響を与えません。

[bit5] PE00 (PPG0/PPG2/PPG4 出力端子許可)

パルス出力外部端子 PPG0/PPG2/PPG4 へのパルス出力の禁止 / 許可を制御します。

PE00	動作状態
0	汎用ポート端子 (パルス出力禁止)
1	PPG0/PPG2/PPG4 パルス出力 (パルス出力許可)

- このビットは、リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit4] PIE0 (PPG0/PPG2/PPG4 の割込み許可)

PPG0/PPG2/PPG4 の割込みの禁止 / 許可を制御します。

PIE0	動作状態
0	割込み禁止
1	割込み許可

- このビットが "1" のとき, PUF0 が "1" になると割込み要求を発生します。このビットが "0" のときは, 割込みを発生しません。
- このビットは, リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit3] PUF0 (PPG0/PPG2/PPG4 カウンタアンダフロー)

PPG0/PPG2/PPG4 のカウンタアンダフローの検出結果を表示します。

PUF0	動作状態
0	PPG カウンタアンダフローが検出していません。
1	PPG カウンタアンダフローを検出しました。

8 ビット PPG6 チャンネルモード (PPG0, PPG1/PPG2, PPG3/PPG4, PPG5) および 8 ビット プリスケアラ +8 ビット PPG モード時には, ch.0, ch.2, ch.4 のカウンタの値が "00_H" ~ "FF_H" になったときのアンダフローにより "1" にセットされます。16 ビット PPG3 チャンネルモード (PPG0, PPG1/PPG2, PPG3/PPG4, PPG5) 時には, ch.1, ch.3, ch.5/ch.0, ch.2, ch.4 のカウンタの値が "0000_H" ~ "FFFF_H" になったときのアンダフローによって "1" にセットされます。"0" の書込みにより "0" になります。PUF0 ビットへの "1" 書込みは意味がありません。リードモディファイライト (RMW) 系命令の読出しに対しては, "1" が読み出されます。

- このビットは, リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit2, bit1] 未定義ビット

読出し値は不定です。書込みは動作に影響を与えません。

[bit0] 予約ビット

予約ビットです。必ず "1" に設定してください。

17.2.2 PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5)

PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5) の構成および機能について説明します。

■ PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5)

PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5) は、ch.1/ch.3/ch.5 の動作モード選択、端子出力制御、カウントクロック選択およびトリガの制御を行います。

図 17.2-3 に、PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5) のビット構成を示します。

図 17.2-3 PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5) のビット構成

	bit 15	14	13	12	11	10	9	8	PPGC1/PPGC3/PPGC5
ch.1 : 000047 _H	PEN1	-	PE10	PIE1	PUF1	MD1	MD0	予約	PPG 動作モード 制御レジスタ リード / ライト 初期値
ch.3 : 000049 _H	(R/W)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
ch.5 : 00004B _H	(0)	(X)	(0)	(0)	(0)	(0)	(0)	(1)	

以下に、PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5) の各ビットの機能を説明します。

[bit15] PEN1 (動作許可)

PPG1/PPG3/PPG5 の動作開始および動作モードを選択します。

PEN1	動作状態
0	動作停止 ("L" レベル出力保持)
1	PPG 動作許可

- このビットに "1" を書き込むことによって、PPG はカウントを開始します。
- このビットは、リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit14] 未定義ビット

リード値は不定です。ライトは動作に影響を与えません。

[bit13] PE10 (PPG1/PPG3/PPG5 出力端子許可)

パルス出力外部端子 PPG1/PPG3/PPG5 へのパルス出力の禁止 / 許可を制御します。

PE10	動作状態
0	汎用ポート端子 (パルス出力禁止)
1	PPG1/PPG3/PPG5 パルス出力 (パルス出力許可)

- このビットは、リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit12] PIE1 (PPG1/PPG3/PPG5 の割込み許可)

PPG1/PPG3/PPG5 の割込みの禁止 / 許可を制御します。

PIE1	動作状態
0	割込み禁止
1	割込み許可

- このビットが "1" のとき、PUF1 が "1" になると割込み要求を発生します。このビットが "0" のときは、割込みを発生しません。
- このビットは、リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit11] PUF1 (PPG1/PPG3/PPG5 カウンタアンダフロー)

PPG1/PPG3/PPG5 のカウンタアンダフローの検出結果を表示します。

PUF1	動作状態
0	PPG カウンタアンダフローを検出していません。
1	PPG カウンタアンダフローを検出しました。

8 ビット PPG6 チャンネルモード (PPG0, PPG1/PPG2, PPG3/PPG4, PPG5) および 8 ビットプリスケラ +8 ビット PPG モード時には、ch.1, ch.3, ch.5 のカウンタの値が "00_H" ~ "FF_H" になったときのアンダフローにより "1" にセットされます。16 ビット PPG3 チャンネルモード (PPG0, PPG1/PPG2, PPG3/PPG4, PPG5) 時には、ch.1, ch.3, ch.5/ch.0, ch.2, ch.4 のカウンタの値が "0000_H" ~ "FFFF_H" になったときのアンダフローによって "1" にセットされます。"0" の書込みにより "0" になります。PUF1 ビットへの "1" 書込みは意味がありません。リードモディファイライト (RMW) 系命令系の読出しに対しては、"1" が読み出されます。

- このビットは、リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

[bit10, bit9] MD1, MD0 (動作モード選択)

PPG タイマの動作モードを選択します。

MD1	MD0	動作モード
0	0	8 ビット PPG 2 チャンネル独立モード (× 3 あり)
0	1	8 ビットプリスケラ +8 ビット PPG 1 チャンネル
1	0	予約 (設定禁止)
1	1	16 ビット PPG 1 チャンネルモード (× 3 あり)

- このビットは、リセットによって "0" に初期化されます。
- 読出し / 書込みが可能です。

< 注意事項 >

- MD1, MD0 ビットを "10_B" に設定しないでください。
 - MD1, MD0 ビットを "01_B" に設定する場合には、PPGC0 の PEN0 ビット /PPGC1 の PEN1 ビットを "01_B" に設定しないでください。また、PEN0 ビットおよび PEN1 ビットを同時に "11" または "00_B" にセットすることを推奨します。
 - MD1, MD0 ビットを "11_B" に設定する場合には、PPGC0/PPGC1 をワード転送で書き換え、PEN0/PEN1 を同時に "11_B" または "00_B" にセットしてください。
-

[bit8] 予約ビット

予約ビットです。必ず "1" に設定してください。

17.2.3 PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45)

PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45) の構成および機能について説明します。

■ PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45)

図 17.2-4 に、PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45) のビット構成を示します。

図 17.2-4 PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45)

bit	7	6	5	4	3	2	1	0	PPG01/PPG23/PPG45
ch.0,ch.1 : 00004C _H	PCS2	PCS1	PCS0	PCM2	PCM1	PCM0	予約	予約	PPG 出力制御レジスタ
ch.2,ch.3 : 00004E _H	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
ch.4,ch.5 : 000050 _H	(0)	(0)	(0)	(0)	(0)	(0)	(X)	(X)	
									リード / ライト 初期値

以下に、PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45) の各ビットの機能を説明します。

[bit7 ~ bit5] PCS2 ~ PCS0 (カウントクロック選択)

ch.1, ch.3, ch.5 のダウンカウンタの動作クロックを選択します。

PCS2	PCS1	PCS0	動作モード
0	0	0	周辺クロック (41.7 ns マシンクロック 24 MHz 時)
0	0	1	周辺クロック /2 (83.3 ns マシンクロック 24 MHz 時)
0	1	0	周辺クロック /4 (167 ns マシンクロック 24 MHz 時)
0	1	1	周辺クロック /8 (333 ns マシンクロック 24 MHz 時)
1	0	0	周辺クロック /16 (667 ns マシンクロック 24 MHz 時)
1	1	1	タイムベースカウンタからの入力クロック (2 ⁹ × 167 ns=85 μs 原発振 6 MHz 時)

- このビットは、リセットによって "000_B" に初期化されます。
- 読出し / 書込みが可能です。

< 注意事項 >

8 ビットプリスケアラ +8 ビット PPG モードおよび 16 ビット PPG モード時には、ch.1, ch.3, ch.5のPPGはch.0, ch.2, ch.4からカウントクロックを受けて動作するため、PCS2 ~ PCS0 ビットの指定は無効になります。

[bit4 ~ bit2] PCM2 ~ PCM0 (カウントクロック選択)

ch.0, ch.2, ch.4 のダウンカウンタの動作クロックを選択します。

PCM2	PCM1	PCM0	動作モード
0	0	0	周辺クロック (41.7 ns マシンクロック 24 MHz 時)
0	0	1	周辺クロック /2 (83.3 ns マシンクロック 24 MHz 時)
0	1	0	周辺クロック /4 (167 ns マシンクロック 24 MHz 時)
0	1	1	周辺クロック /8 (333 ns マシンクロック 24 MHz 時)
1	0	0	周辺クロック /16 (667 ns マシンクロック 24 MHz 時)
1	1	1	タイムベースカウンタからの入力クロック ($2^9 \times 167 \text{ ns} = 85 \mu\text{s}$ 原発振 6 MHz 時)

- このビットは , リセットによって "000_B" に初期化されます。
- 読出し / 書込みが可能です。

[bit1, bit0] 予約ビット

予約ビットです。必ず "00_B" に設定してください。

17.2.4 PPG リロードレジスタ (PRLL0 ~ PRLL5, PRLH0 ~ PRLH5)

PPG リロードレジスタ (PRLL0 ~ PRLL5, PRLH0 ~ PRLH5) の構成および機能について説明します。

■ PPG リロードレジスタ (PRLL0 ~ PRLL5, PRLH0 ~ PRLH5)

図 17.2-5 に、PPG リロードレジスタ (PRLL0 ~ PRLL5, PRLH0 ~ PRLH5) のビット構成を示します。

図 17.2-5 PPG リロードレジスタ (PRLL0 ~ PRLL5, PRLH0 ~ PRLH5)

bit	7	6	5	4	3	2	1	0	PRLL0~PRLL5
ch.0 : 007900 _H	D07	D06	D05	D04	D03	D02	D01	D00	PPG リロードレジスタ下位 リード / ライト 初期値
ch.1 : 007902 _H	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
ch.2 : 007904 _H	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
ch.3 : 007906 _H									
ch.4 : 007908 _H									
ch.5 : 00790A _H									
bit	15	14	13	12	11	10	9	8	PRLH0~PRLH5
ch.0 : 007901 _H	D15	D14	D13	D12	D11	D10	D09	D08	PPG リロードレジスタ上位 リード / ライト 初期値
ch.1 : 007903 _H	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
ch.2 : 007905 _H	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
ch.3 : 007907 _H									
ch.4 : 007909 _H									
ch.5 : 00790B _H									

PPG リロードレジスタ (PRLL0 ~ PRLL5, PRLH0 ~ PRLH5) は、ダウンカウンタ (PCNT) へのリロード値を保持する、各 8 ビットのレジスタです。それぞれ次の役割を持っています。

レジスタ名	機能
PRLL	"L" 側リロード値保持
PRLH	"H" 側リロード値保持

どちらのレジスタも、読出し / 書込み可能です。

< 注意事項 >

8 ビットプリスケラ +8 ビット PPG モードで使用する場合には、ch.0, ch.2, ch.4 の PRLL と PRLH に異なる値を設定すると、ch.1, ch.3, ch.5 の PPG 波形がサイクルごとに異なる場合があります。したがって、ch.0, ch.2, ch.4 の PRLL と PRLH は、同じ値に設定することを推奨します。

17.3 8/16 ビット PPG タイマの動作

8/16 ビット PPG タイマには、8 ビット長の PPG ユニットが 6 チャンネル (PPG0, PPG1/PPG2, PPG3/PPG4, PPG5) あり、それぞれが独立モード以外にも、直結 (PPG0+PPG1/PPG2+PPG3/PPG4+PPG5) 動作をさせることによって、8 ビットプリスケアラ +8 ビット PPG モードと、16 ビット PPG モードの計 3 種類の動作を行うことができます。

■ 8/16 ビット PPG タイマの動作概略

8 ビット長の PPG ユニットそれぞれは、8 ビット長のリロードレジスタが "L" 側と "H" 側の 2 本あります (PRLH, PRLH)。

PRLH, PRLH レジスタに書き込まれた値が、8 ビットダウンカウンタ (PCNT) に "L" 側 / "H" 側交互にリロードされてカウントクロックごとにダウンカウントされ、カウントのボロー発生によるリロード時に、出力端子の値を反転させます。この動作によって、端子出力はリロードレジスタ値に対応した "L" 幅 / "H" 幅をもつパルス出力となります。

動作開始 / 再スタートは、レジスタのビット書込みによります。

リロード動作とパルス出力の関係を示します。

リロード動作	端子出力変化
PRLH PCNT	PPG0/PPG1[0 1] 立上り

また、PPGC0/PPGC2/PPGC4 中の bit4(PIE0) が "1" のときおよび PPGC1/PPGC3/PPGC5 中の bit2(PIE1) が "1" のとき、それぞれのカウンタの "00_H FF_H" へのボロー (16 ビット PPG モードの場合は、"0000_H FFFF_H" へのボロー) によって割込み要求が出力されます。

■ 動作モード

8/16 ビット PPG タイマは、2 チャンネル独立モードと 8 ビットプリスケアラ +8 ビット PPG モードと、16 ビット PPG モードの計 3 種類の動作モードがあります (MB90330A シリーズはモードごとに × 3 チャンネルあります)。

2 チャンネル独立モードは、8 ビット PPG として 2 チャンネル独立に動作させるモードです。PPG0 端子には、ch.0 の PPG 出力が接続され、PPG1 には ch.1 の PPG 出力が接続されます (PPG2 ~ PPG5 ch.2 ~ ch.5 に対応)。

8 ビットプリスケアラ +8 ビット PPG モードは、ch.0(ch.2/ch.4) を 8 ビットプリスケアラとして動作させ、ch.1(ch.3/ch.5) を ch.0(ch.2/ch.4) のボロー出力でカウントすることによって、任意周期の 8 ビット PPG 波形を出力できるようにする動作モードです。PPG0(PPG2/PPG4) 端子は、ch.0(ch.2/ch.4) のプリスケアラ出力が接続され、PPG1 端子は、ch.1(ch.3/ch.5) の PPG の出力が接続されます。

16 ビット PPG チャンネルモード (MB90330A シリーズは × 3 チャンネルあります) は、ch.0 と ch.1 を直結させて (ch.2, ch.3/ch.4, ch.5 直結) 16 ビットの PPG として動作させる動作モードです。PPG0, PPG1 は両方ともに 16 ビット PPG 出力が接続されます。

■ PPG 出力動作

8/16 ビット PPG タイマは、ch.0(ch.2/ch.4) の PPG については、PPGC0 の bit7(PEN0) を、ch.1(ch.3/ch.5) の PPG については、PPGC1 の bit15(PEN1) を "1" にセットすることによって起動され、カウントを開始します。動作を開始した後は、PPGC0 の bit7(PEN0) または PPGC1 の bit15(PEN1) に "0" を書き込むことによってカウント動作を停止し、停止した後にパルス出力は "L" レベルを保持します。

8 ビットプリスケアラ +8 ビット PPG モードのときには、ch.0(ch.2/ch.4) を停止状態で、ch.1(ch.3/ch.5) を動作状態に設定しないでください。

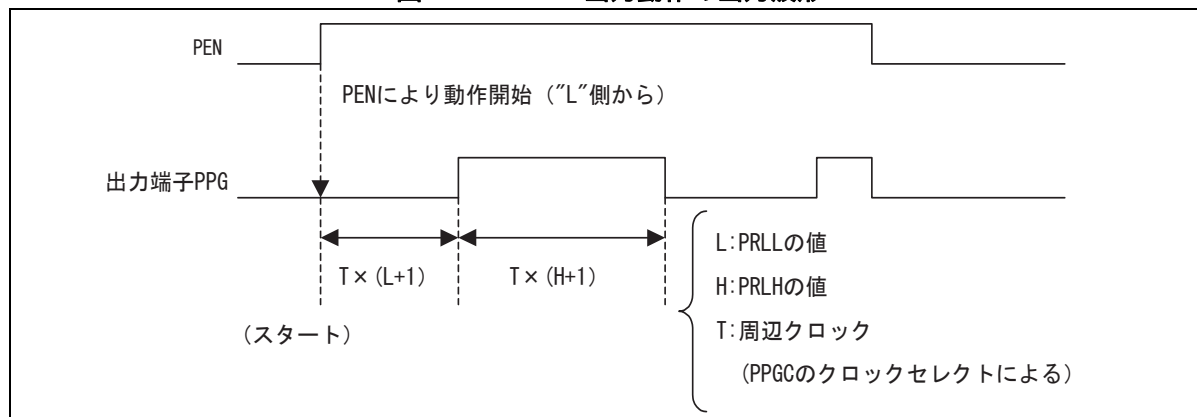
16 ビット PPG モードのときには、PPGC0 中の bit7(PEN0) と PPGC1 中の bit15(PEN1) は、同時に開始 / 停止の制御を行ってください。

以下に、PPG 出力動作を説明します。

PPG 動作時は、任意周期任意デューティ比 (パルス波の "H" レベル期間と "L" レベル期間の比) のパルス波出力を連続して出力します。PPG はパルス波出力を開始し、動作停止を設定するまで停止しません。

図 17.3-1 に、PPG 出力動作の出力波形を示します。

図 17.3-1 PPG 出力動作の出力波形



■ リロード値とパルス幅の関係

リロードレジスタに書かれた値に+1した値に、カウントクロックの周期をかけた値が、出力されるパルスの幅となります。つまり、8 ビット PPG 動作時のリロードレジスタ値は 00_H のときおよび 16 ビット PPG 動作時のリロードレジスタ値が 0000_H のときは、カウントクロック 1 周期分のパルス幅を持つことになりますので注意してください。また、8 ビット PPG 動作時のリロードレジスタ値が FF_H のときは、カウントクロック 256 周期分のパルス幅をもつことになり、16 ビット PPG 動作時のリロードレジスタ値が $FFFF_H$ のときは、カウントクロック 65536 分周分のパルス幅をもつことになりますので注意してください。パルス幅の計算式を以下に示します。

$$P_L = T \times (L+1)$$

$$P_H = T \times (H+1)$$

P_L : "L" パルスの幅

P_H : "H" パルスの幅

T : 入力クロック周期

L : PRLH 値

H : PRLH 値

■ カウントクロックの選択

8/16 ビット PPG タイマの動作に使用するカウントクロックは、周辺クロックおよびタイムベースカウンタの入力を使用しており、6種類のカウントクロック入力を選択できます。

PPG01/PPG23/PPG45 の bit4 ~ bit2(PCM2 ~ PCM0) で ch.0(ch.2/ch.4) のクロックを、PPG01/PPG23/PPG45 の bit7 ~ bit5(PCS2 ~ PCS0) で ch.1(ch.3/ch.5) のクロックを選択します。

クロックの選択は、マシニングクロックより 1/16 ~ 1 倍およびタイムベースカウンタの入力を選択します。

< 注意事項 >

- 8 ビットプリスケアラ +8 ビット PPG モードおよび 16 ビット PPG モードのときは、PPGC1 の bit14(PCS1) の値は無効となります。
- タイムベースタイマの入力を使用した場合、トリガまたは停止後の最初のカウントサイクルがずれる可能性があります。また、8/16 ビット PPG タイマが動作中にタイムベースカウンタの初期化を行うとサイクルがずれることがあります。
- 8 ビットプリスケアラ +8 ビット PPG モードで、ch.0/ch.2/ch.4 が動作状態で ch.1/ch.3/ch.5 が停止状態のときに ch.1(ch.3/ch.5) を起動すると、最初のカウントサイクルがずれる可能性があります。

■ パルスの端子出力制御

8/16 ビット PPG タイマによって生成したパルスは、外部端子 (PPG0 ~ PPG5) から出力させることができます。外部端子からパルスを出力するには、各端子に対応するビットに "1" を書き込みます。PPG0/PPG2/PPG4 端子に対しては PPGC0 の bitPE0) を、PPG1/PPG3/PPG5 に対しては PPGC1 の bit3(PE1) を使用します。このビットに "0" が書き込まれているとき (初期値) は、パルスは外部端子から出力されずに、汎用ポートとして機能します。

16 ビット PPG モードでは、PPG0 ~ PPG5 からは同じ波形が出力されるので、どちらの外部端子を有効にしても同じ出力を得ることができます。

8 ビットプリスケアラ +8 ビット PPG モードでは、PPG0/PPG2/PPG4 からは 8 ビットプリスケアラのトグル波形が出力され、PPG1/PPG3/PPG5 からは 8 ビット PPG の波形が出力されます。

このモードにおける出力波形を図 17.3-2 に示します。

図 17.3-2 8+8PPG 出力動作の波形

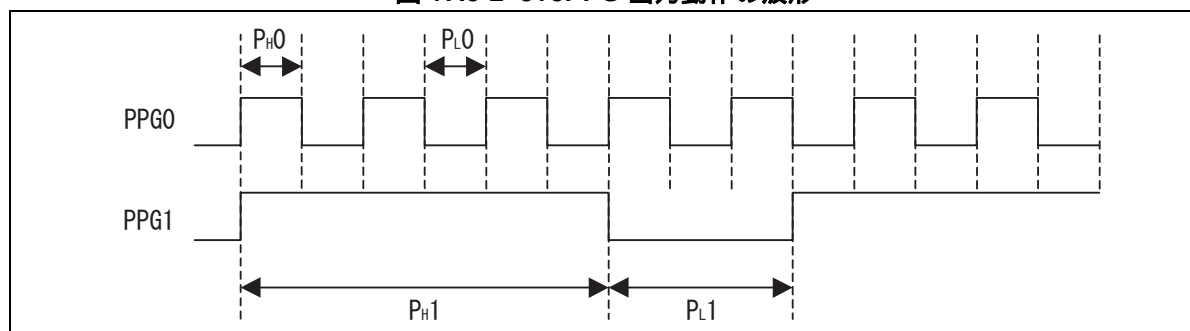


図 17.3-2 におけるパルス幅は、以下の式によって計算できます。

$$P_{L0} = T \times (L0 + 1)$$

$$P_{H0} = T \times (L0 + 1)$$

$$P_{L1} = T \times (L0 + 1) \times (L1 + 1)$$

$$P_{H1} = T \times (L0 + 1) \times (H1 + 1)$$

L0 : ch.0 の PRLH の値および ch.1 の PRLH の値

L1 : ch.1 の PRLH の値

H1 : ch.1 の PRLH の値

T : 入力クロックサイクル

P_{H0}: PPG0 の "H" パルスの幅

P_{L0}: PPG0 の "L" パルスの幅

P_{H1}: PPG1 の "H" パルスの幅

P_{L1}: PPG1 の "L" パルスの幅

< 注意事項 >

ch.0 の PRLH と ch.1 の PRLH は同じ値を設定します。

■ 8/16 ビット PPG タイマの割込み

8/16 ビット PPG タイマの割込みは、リロード値がカウントアウトしてボローが発生したときにアクティブになります。8 ビット PPG2 チャンネルモードまたは 8 ビットプリスケラ +9 ビット PPG モード (MB90330A シリーズにはそれぞれ × 3 チャンネル分あります) のときは、それぞれのボローによってそれぞれの割込み要求が行われます。しかし、16 ビット PPG モードでは、16 ビットのカウンタのボローによって PUF0 と PUF1 が同時に設定されます。このため、割込み要因を 1 本化するために、PIE0 または PIE1 のどちらか一方のみを許可します。また、割込み要因のクリアも PUF0 と PUF1 を同時に行います。

■ ハードウェアコンポーネントの初期値

8/16 ビット PPG タイマのハードウェアコンポーネントは、リセット時に次の値に初期化されます。

< レジスタ >	PPG0	0X000001 _B
	PPG1	00000001 _B
	PPG01	XXXXXX00 _B
< パルス出力 >	PPG0	"L"
	PPG1	"L"
	PE0	PPG0 出力禁止
	PE1	PPG1 出力禁止
< 割込み要求 >	IRQ0	"L"
	IRQ1	"L"

上記以外のハードウェアコンポーネントは初期化されません。

■ リロードレジスタへの書き込みタイミング

16 ビット PPG モード以外のモードでは、リロードレジスタ PRLH, PRLH へのデータ書き込みはワード転送命令を使用することを推奨します。データ項目をバイト転送命令 2 回でレジスタに書き込んだ場合、タイミングによっては予想しないパルス幅の出力が発生する可能性があります。

図 17.3-3 に、リロードレジスタへの書き込みタイミングを示します。

図 17.3-3 リロードレジスタへの書き込みタイミングチャート

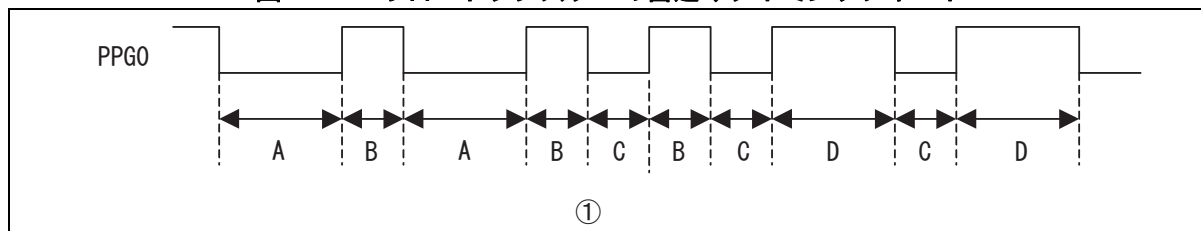
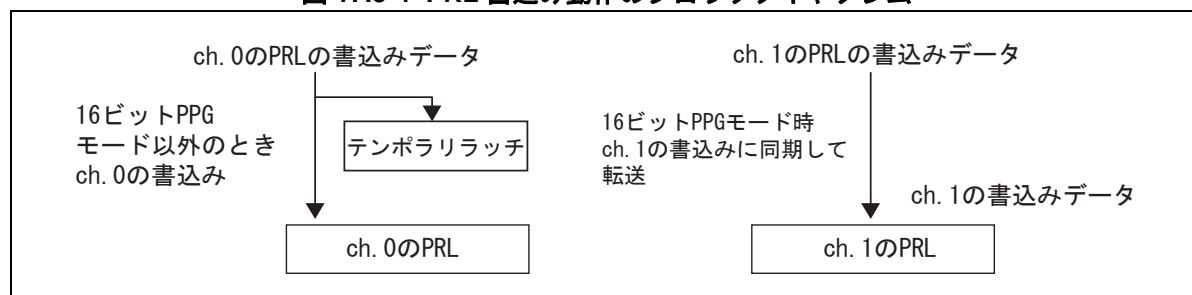


図 17.3-3 において、のタイミングの前で PRLH を A から C に書き換えて、の後に PRLH の値を B から D に書き換えたとすると、のタイミングにおける PRL の値は、 $PRLH=C$, $PRLH=B$ のため 1 回だけ L 側のカウンタ数 C, H 側のカウンタ数 B のパルスが発生します。同様に 16 ビット PPG モードで ch.0/ch.2/ch.4 と ch.1/ch.3/ch.5 の PRL にデータを書き込むには、ロングワード転送命令を使用するかまたは ch.0 ch.1(ch.2 ch.3/ch.4 ch.5) の PRL に順にワード転送命令を使用します。このモードのときは、データは、ch.0/ch.2/ch.4 から PRL へ一時的に書き込まれ、その後 ch.1/ch.3/ch.5 の PRL に書き込んだ時点で、実際に ch.0 の PRL へ書き込まれます。

16 ビット PPG モード以外では、ch.0/ch.2/ch.4 と ch.1/ch.3/ch.5 の書き込みは独立に行われます。

図 17.3-4 に、PRL 書き込み動作のブロックダイヤグラムを示します。

図 17.3-4 PRL 書き込み動作のブロックダイヤグラム



第 18 章

DTP/ 外部割込み

この章では、DTP/ 外部割込みの概要、レジスタの構成 / 機能および DTP/ 外部割込みの動作について説明します。

18.1 DTP/ 外部割込みの概要

18.2 DTP/ 外部割込みのレジスタ

18.3 DTP/ 外部割込みの動作

18.4 DTP/ 外部割込み使用上の注意

18.1 DTP/ 外部割込みの概要

DTP(Data Transfer Peripheral) は、デバイス外部に存在する周辺装置と F²MC-16LX CPU との間であって、外部周辺装置が発生する DMA 要求または割込み要求を受け取り、これを F²MC-16LX CPU に伝えて μ DMAC または割込み処理を起動させるための周辺制御部です。

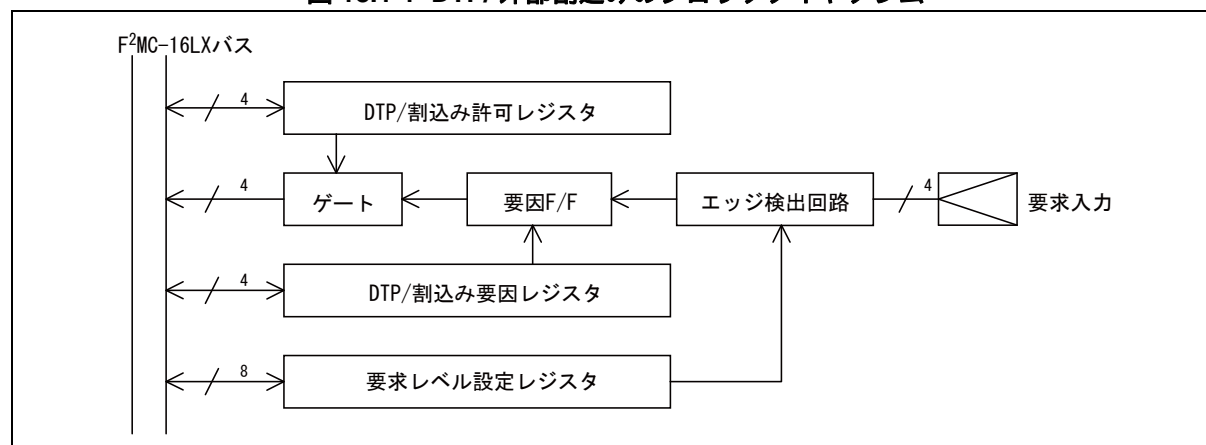
■ DTP/ 外部割込みの概要

要求レベルとして、 μ DMAC の場合は "H", "L" の 2 種類が、外部割込み要求の場合は "H", "L" のほかに立上りエッジと立下りエッジの計 4 種類が選択できます。

■ DTP/ 外部割込みのブロックダイアグラム

図 18.1-1 に、DTP/ 外部割込みのブロックダイアグラムを示します。

図 18.1-1 DTP/ 外部割込みのブロックダイアグラム



18.2 DTP/ 外部割込みのレジスタ

DTP/ 外部割込みで使用するレジスタの構成および機能について説明します。

■ DTP/ 外部割込みのレジスタ一覧

図 18.2-1 に、DTP/ 外部割込みのレジスタ一覧を示します。

図 18.2-1 DTP/ 外部割込みのレジスタ一覧

bit	7	6	5	4	3	2	1	0	
アドレス : 00003C _H	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	DTP/ 割込み許可レジスタ (ENIR)
bit	15	14	13	12	11	10	9	8	
アドレス : 00003D _H	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	DTP/ 割込み要因レジスタ (EIRR)
bit	7	6	5	4	3	2	1	0	
アドレス : 00003E _H	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	要求レベル設定レジスタ (ELVR)
bit	15	14	13	12	11	10	9	8	
アドレス : 00003F _H	LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	要求レベル設定レジスタ (ELVR)

■ DTP/ 割込み許可レジスタ (ENIR:Enable Interrupt Request Register)

図 18.2-2 に、DTP/ 割込み許可レジスタ (ENIR) のビット構成を示します。

図 18.2-2 DTP/ 割込み許可レジスタ (ENIR) のビット構成

ENIR									
bit	7	6	5	4	3	2	1	0	初期値
アドレス : 00003C _H	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	

DTP/ 割込み許可レジスタ (ENIR) は、デバイス端子を外部 DTP/ 割込み要求入力として使用し、割込みコントローラに対して要求を発生させる機能を動作させることを決定するレジスタです。ENIR レジスタに "1" が書かれたビットに対応する端子は、外部 DTP/ 割込み要求入力として使用され、割込みコントローラに対して要求を発生する機能を動作させます。"0" が書き込まれたビットに対応する端子は、DTP/ 外部割込み要求入力要因は保持しますが、割込みコントローラに対しては要求を発生しません。

■ DTP/ 割込み要因レジスタ (EIRR:External Interrupt Request Register)

図 18.2-3 に、DTP/ 割込み要因レジスタ (EIRR) のビット構成を示します。

図 18.2-3 DTP/ 割込み要因レジスタ (EIRR) のビット構成

EIRR									初期値
アドレス : 00003D _H	bit 15	14	13	12	11	10	9	8	00000000 _B
	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	(ただし両者で対象が異なる)

DTP/ 割込み要因レジスタ (EIRR) は、読出し時には対応する DTP/ 外部割込み要求があることを示し、書込み時にはこの要求を示すフリップフロップの内容をクリアします。EIRR レジスタを読み出したときに "1" であった場合は、ER_x ビットに対応する端子に外部割込み /DTP 要求があることを示します。また、EIRR レジスタに "0" を書き込むと対応するビットの要求フリップフロップはクリアされます。"1" を書き込んでも何の操作も行いません。リードモディファイライト (RMW) 系命令の読出し時には "1" が読み出されます。

< 注意事項 >

- 初期値は "00_H" ですが、外部割込み兼用端子の状態によってリセット解除後の値は変化します。
- 複数の外部割込み要求出力が許可 (ENIR:EN7 ~ EN0=1) されている場合、CPU が割込みを受付けたビット (ER7 ~ ER0 の "1" にセットされているビット) だけを "0" にクリアするようにしてください。それ以外のビットを無条件にクリアすることは避けてください。
- DTP/ 外部割込み要因ビット (EIRR:ER) の値は、対応する DTP/ 外部割込み許可ビット (ENIR:EN) が "1" に設定されているときのみ有効です。
DTP/ 外部割込みが許可されてない状態 (ENIR:EN=0) では、DTP/ 外部割込み要因の有無にかかわらず DTP/ 外部割込み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/ 外部割込み要因ビット (EIRR:ER) をクリアしてください。

■ 要求レベル設定レジスタ (ELVR:External Level Register)

図 18.2-4 に、要求レベル設定レジスタ (ELVR) のビット構成を示します。

図 18.2-4 要求レベル設定レジスタ (ELVR)

ELVR									初期値
アドレス : 00003E _H	bit 7	6	5	4	3	2	1	0	00000000 _B
	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
アドレス : 00003F _H	bit 15	14	13	12	11	10	9	8	初期値 00000000 _B
	LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	
	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 (ELVR) は、要求検出レベルを選択するレジスタです。1 つの端子あたり 2 ビットが割り当てられていて、表 18.2-1 に示す対応となっています。要求入力レベルの場合、クリアしても入力がアクティブならば再びセットされます。

表 18.2-1 ELVR 割当て (LA0 ~ LA7, LB0 ~ LB7)

LBx	LAx	動作
0	0	"L" レベルで要求あり
0	1	"H" レベルで要求あり
1	0	立上りエッジで要求あり
1	1	立下りエッジで要求あり

18.3 DTP/ 外部割込みの動作

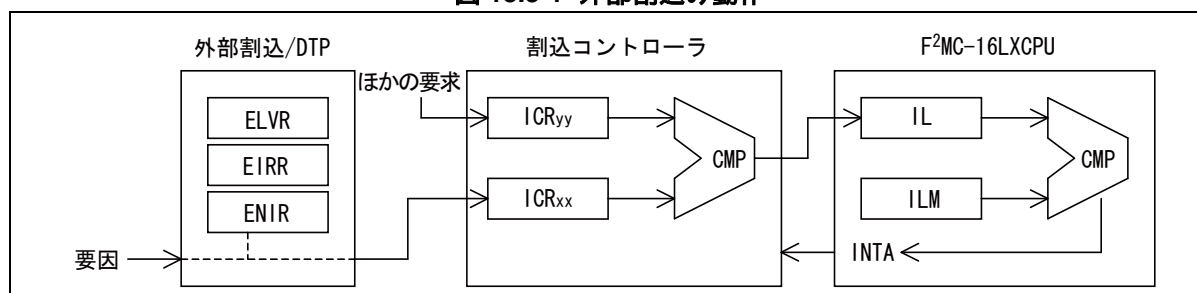
DTP/ 外部割込みの動作について説明します。

■ 外部割込みの動作

外部割込み要求の設定の後，対応する端子に ELVR で設定された要求が入力されると，本リソースは割込みコントローラに対して割込み要求信号を発生します。割込みコントローラ内で同時発生した割込みの優先順位を識別した結果，本リソースからの割込みが最も優先順位が高かったときに，割込みコントローラは F²MC-16LX CPU に対して割込み要求を発生します。F²MC-16LX CPU は，プロセッサステータス (PS) 内の割込みレベルマスクレジスタ (ILM) と割込み要求を比較し，要求レベルが ILM ビットより高かったときに現在実行中の命令が終了し次第，ハードウェア割込み処理マイクロプログラムを起動します。

図 18.3-1 に，外部割込み動作のフローを示します。

図 18.3-1 外部割込み動作



割込み処理マイクロプログラムでは割込みベクタ領域の読出しと，割込みコントローラへの割込みアクノリッジを発生し，ベクタから生成したマクロ命令のジャンプ先アドレスをプログラムカウンタへ転送した上で，ユーザの割込み処理プログラムを実行します。

■ DTP の動作

初期化としてユーザプログラム内では μ DMAC を起動するにあたって， μ DMAC ディスクリプタ内の I/O アドレスポインタに 000000_H から 0000FF_H に割り付けられているレジスタのアドレスを設定し，バッファアドレスポインタにメモリバッファの先頭アドレスを設定します。

DTP の動作シーケンスは外部割込みの場合とほぼ同じで，CPU がハードウェア割込み処理マイクロプログラムを起動するまでは全く同じです。 μ DMAC が起動されると，アドレスリングされている外部周辺装置に読出しまたは書込み信号が送られ，本チップとの間で転送が行われます。外部周辺装置は，その転送が行われてから 3 マシンサイクル以内に本チップに対する割込み要求を取り下げてください。転送が終了するとディスクリプタの更新などが行われ，その後転送要因をクリアする信号を割込みコントローラに発生させます。転送要因をクリアする信号を受け取った本リソースは，要因を保持しているフリップフロップをクリアして，端子からの次の要求に備えます。

図 18.3-2 に，DTP 動作終了時の外部割込み要求を取り下げるタイミングを示します。

また，図 18.3-3 に，外部周辺装置とのインタフェース例を示します。

図 18.3-2 DTP 動作終了時の外部割り込み要求の取下げタイミング

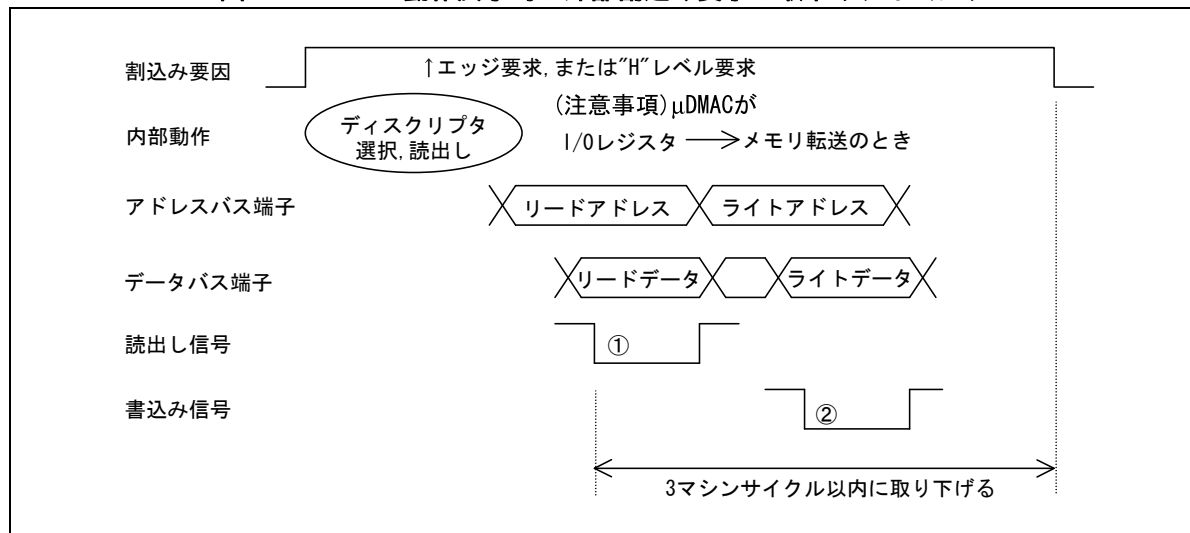
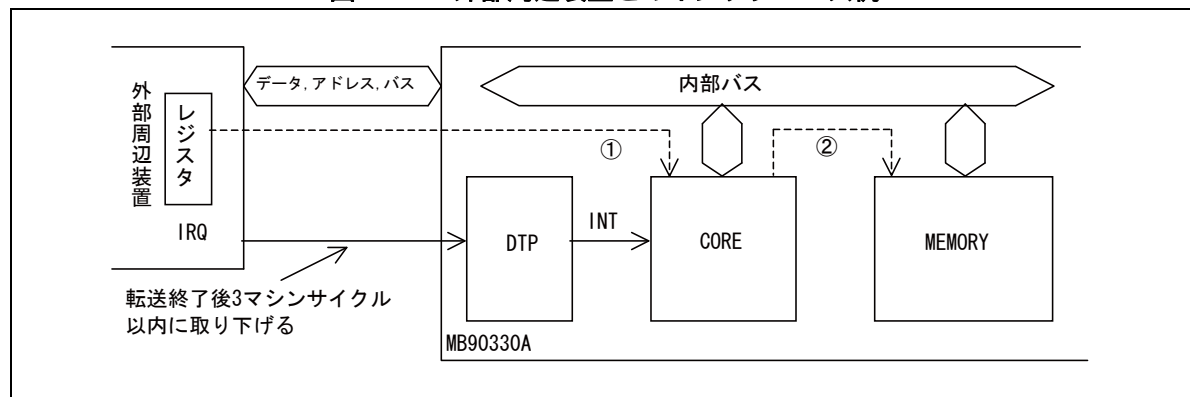


図 18.3-3 外部周辺装置とのインタフェース例



18.4 DTP/ 外部割込み使用上の注意

DTP/ 外部割込みを使用する上での注意事項について説明します。

■ 外部に接続する周辺装置の条件

DTP がサポートできる外部周辺装置は、転送が行われたことにより自動的に要求をクリアするものでなければなりません。また、転送動作を開始してから 3 マシンサイクル以内（暫定値）で転送要求を取り下げようになっていないと、本リソース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。

■ DTP/ 外部割込みの動作手順

DTP/ 外部割込み内に存在するレジスタの設定は、次の手順で行ってください。

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する。
 2. 許可レジスタの対象となるビットを禁止状態にする。
 3. 要求レベル設定レジスタの対象となるビットを設定する。
 4. 要因レジスタの対象となるビットをクリアする。
 5. 許可レジスタの対象となるビットを許可状態にする。
4. と 5. はワード指定による同時書き込みが可能です。

本リソース内のレジスタを設定するときには、必ず許可レジスタを禁止状態に設定しておかなくてはなりません。また、許可レジスタを許可状態にする前に必ず要因レジスタをクリアしておく必要があります。これは、レジスタ設定時や割込み許可状態時に誤って割込み要因が発生してしまうことを避けるためです。

■ 外部割込み要求レベル

- 要求レベルがエッジ要求の場合、エッジがあったことを検出するためには、パルス幅は最小 3 マシンサイクル必要です。
- 要求入力レベルがレベル設定の場合、外部から要求が入力され、その後取り下げられても、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のままです。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) をクリアする必要があります。

図 18.4-1 レベル設定時の割込み要求フラグビット (EIRR:ER) のクリア

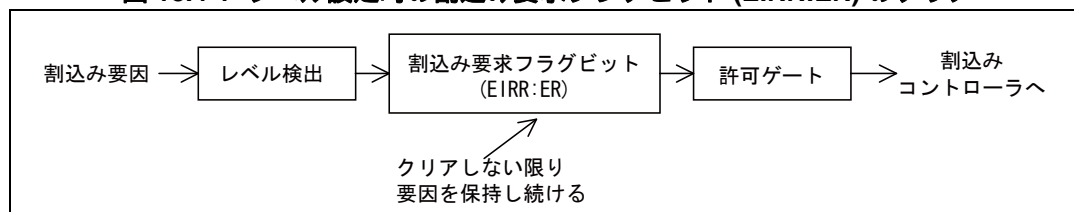
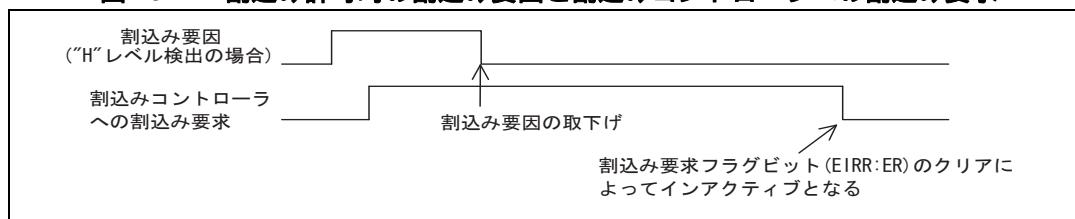


図 18.4-2 割込み許可時の割込み要因と割込みコントローラへの割込み要求



< 注意事項 >

時計モードからのエッジ検出による復帰はできません。

第19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。

- 19.1 8/10 ビット A/D コンバータの概要
- 19.2 8/10 ビット A/D コンバータの構成
- 19.3 8/10 ビット A/D コンバータのレジスタ
- 19.4 8/10 ビット A/D コンバータの動作説明
- 19.5 8/10 ビット A/D コンバータ使用上の注意
- 19.6 8/10 ビット A/D コンバータのプログラム例 -1
(単発モードの EI²OS 起動例)
- 19.7 8/10 ビット A/D コンバータのプログラム例 -2
(連続モードの EI²OS 起動例)
- 19.8 8/10 ビット A/D コンバータのプログラム例 -3
(停止モードの EI²OS 起動例)

19.1 8/10 ビット A/D コンバータの概要

8/10 ビット A/D コンバータには、RC 逐次比較変換方式でアナログ入力電圧を 8 ビットもしくは 10 ビットのデジタル値に変換する機能があります。入力信号は、16 チャンネルのアナログ入力端子から選択し、変換起動は、ソフトウェア、PPG タイマ 1 出力および外部端子からのトリガ入力起動の 3 種類から選択できます。

■ 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）をデジタル値に A/D 変換する機能があり、次の特長をもっています。

- 変換時間は、最小 7.16 μ s(マシクロック 24 MHz 時、サンプリング時間を含む)です。
- サンプリング時間は、最小 3.33 μ s(マシクロック 6 MHz 時)です。
- 変換方式は、サンプルホールド回路付 RC 逐次変換比較方式です。
- アナログ入力端子は 16 チャンネルからプログラムで選択可能です。
- A/D 変換終了時に割込み要求を発生でき、EI²OS を起動することもできます。
割込み許可の状態では変換データ保護機能が働くため連続変換してもデータの欠落がありません。
- 変換の起動要因は、ソフトウェア、PPG タイマ 1 出力（立上りエッジ）、外部からのトリガ入力（立下りエッジ）から選択できます。

変換モードは、表 19.1-1 に示すように 3 種類あります。

表 19.1-1 8/10 ビット A/D コンバータの変換モード

変換モード	シングル変換動作	スキャン変換動作
単発変換モード	指定したチャンネル (1 チャンネルのみ) を 1 回変換して終了	連続した複数のチャンネル (最大 16 チャンネルまで指定可能) を 1 回変換して終了
連続変換モード	指定したチャンネル (1 チャンネルのみ) を繰り返し変換	連続した複数のチャンネル (最大 16 チャンネルまで指定可能) を繰り返し変換
停止変換モード	指定したチャンネル (1 チャンネルのみ) を 1 回変換したら一時停止し、次の起動がかかるまで待機	連続した複数のチャンネル (最大 16 チャンネルまで指定可能) を変換 ただし、1 チャンネル変換ごとに一時停止し、次の起動がかかるまで待機

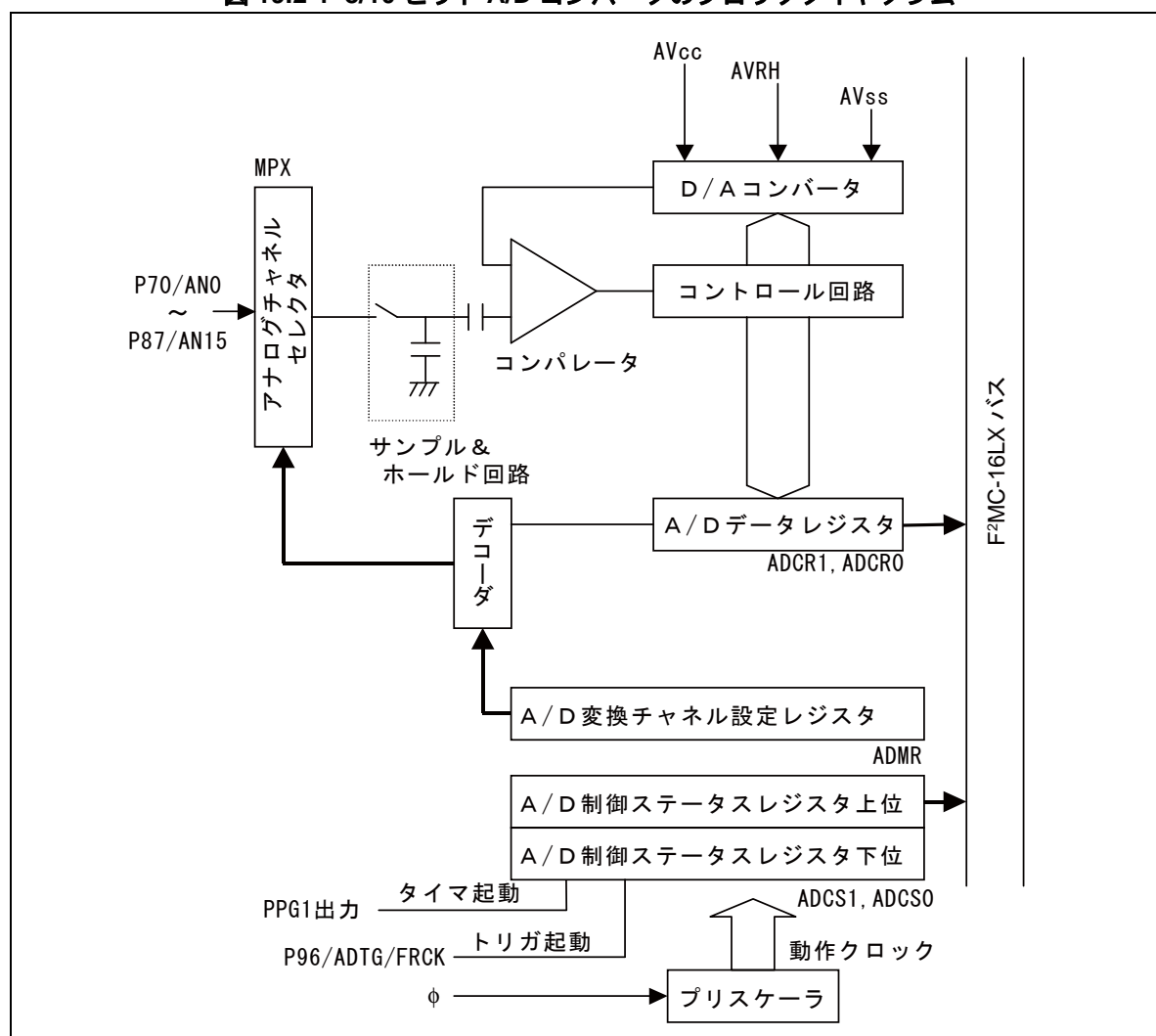
19.2 8/10 ビット A/D コンバータの構成

8/10 ビット A/D コンバータは、次の 9 つのブロックで構成されています。

- A/D 制御ステータスレジスタ (ADCS)
- A/D データレジスタ (ADCR)
- A/D 変換チャンネル設定レジスタ (ADMR)
- デコーダ
- アナログチャンネルセクタ
- サンプルホールド回路
- D/A コンバータ
- コンパレータ
- コントロール回路 (逐次比較レジスタ)

■ 8/10 ビット A/D コンバータのブロックダイアグラム

図 19.2-1 8/10 ビット A/D コンバータのブロックダイアグラム



● A/D 制御ステータスレジスタ (ADCS)

ソフトウェアによる起動、起動トリガの選択、変換モードの選択、A/D 変換チャネルの選択、割込み要求の許可 / 禁止、割込み要求の状態の確認、一時停止中および変換中の表示をします。

● A/D データレジスタ (ADCR)

A/D 変換結果を格納するレジスタであり、A/D 変換の分解能を選択する機能もあります。

● A/D 変換チャネル設定レジスタ (ADMR)

A/D 変換チャネルの選択をします。

● デコーダ

A/D 変換チャネル設定レジスタ (ADMR) の ANE0 ~ ANE3, ANS0 ~ ANS3 ビットの設定から、使用するアナログ入力端子を選択する回路です。

● アナログチャネルセクタ

16 本のアナログ入力端子の中から使用する端子を選択する回路です。

● 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する回路です。A/D 変換を起動した直後の入力電圧をサンプルホールドすることで、A/D 変換中 (比較中) 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

サンプルホールドされた入力電圧と、D/A コンバータの出力電圧を比較し、大小を判定します。

● コントロール回路 (逐次比較レジスタ)

コンパレータからの大小信号で、A/D 変換値を決定します。A/D 変換が終了すると、変換結果を A/D データレジスタ (ADCR) に格納し、割込み要求を発生します。

19.3 8/10 ビット A/D コンバータのレジスタ

8/10 ビット A/D コンバータのレジスタ一覧を示します。

■ 8/10 ビット A/D コンバータのレジスタ一覧

図 19.3-1 8/10 ビット A/D コンバータのレジスタ一覧

アドレス: 000040 _H	bit 7	6	5	4	3	2	1	0	A/D 制御ステータスレジスタ下位 (ADCS0) 初期値 00-----0 _B
	MD1 (R/W)	MD0 (R/W)	- (-)	- (-)	- (-)	- (-)	- (-)	予約 (R/W)	
アドレス: 000041 _H	bit 15	14	13	12	11	10	9	8	A/D 制御ステータスレジスタ上位 (ADCS1) 初期値 00000000 _B
	BUSY (R/W)	INT (R/W)	INTE (R/W)	PAUS (R/W)	STS1 (R/W)	STS0 (R/W)	STRT (W)	予約 (R/W)	
アドレス: 000042 _H	bit 7	6	5	4	3	2	1	0	A/D データレジスタ下位 (ADCR0) 初期値 XXXXXXXX _B
	D7 (R)	D6 (R)	D5 (R)	D4 (R)	D3 (R)	D2 (R)	D1 (R)	D0 (R)	
アドレス: 000043 _H	bit 15	14	13	12	11	10	9	8	A/D データレジスタ上位 (ADCR1) 初期値 00101XXX _B
	S10 (R/W)	ST1 (W)	ST0 (W)	CT1 (W)	CT0 (W)	- (-)	D9 (R)	D8 (R)	
アドレス: 000045 _H	bit 15	14	13	12	11	10	9	8	A/D 変換チャネル選択レジスタ (ADMR) 初期値 00000000 _B
	ANS3 (R/W)	ANS2 (R/W)	ANS1 (R/W)	ANS0 (R/W)	ANE3 (R/W)	ANE2 (R/W)	ANE1 (R/W)	ANE0 (R/W)	

19.3.1 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) は、ソフトウェアによる起動、起動トリガの選択、割り込み要求の許可 / 禁止、割り込み要求の状態の確認、一時停止中および変換中の確認をする機能があります。

■ A/D 制御ステータスレジスタ上位 (ADCS1)

図 19.3-2 A/D 制御ステータスレジスタ上位 (ADCS1) のビット構成

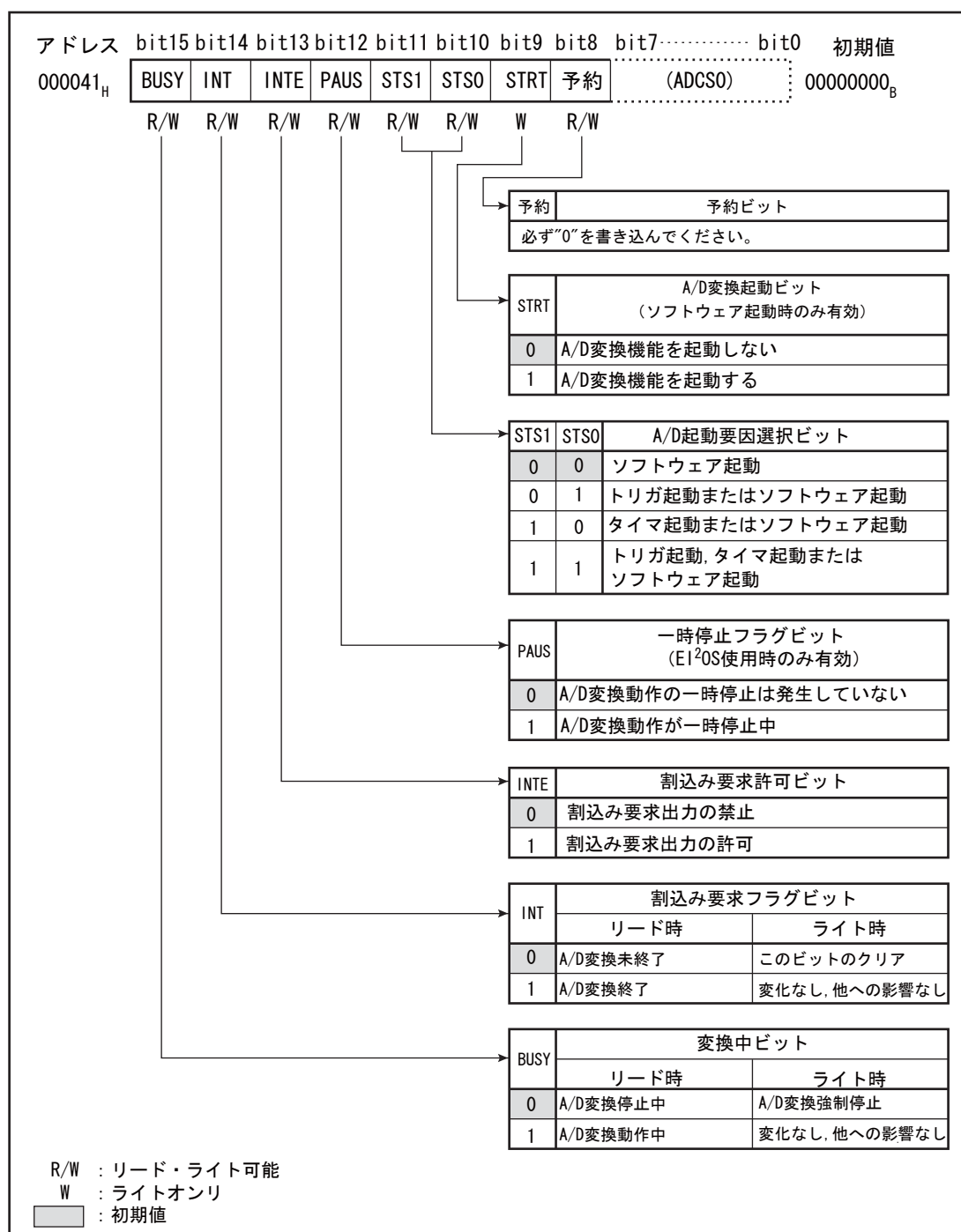


表 19.3-1 A/D 制御ステータスレジスタ上位 (ADCS1) の各ビットの機能説明 (1 / 2)

ビット名		機 能
bit15	BUSY : 変換中ビット	<ul style="list-style-type: none"> ・ A/D コンバータの動作表示ビットです。 ・ リード時, BUSY ビットが "0" であれば A/D 変換停止中であることを示し, "1" であれば A/D 変換動作中であることを示します。 ・ ライト時, BUSY ビットへの "0" の書込みによって A/D 変換動作は強制的に停止されます。"1" の書込みでは, 変化せず他への影響はありません。 <p>(注意事項) 強制停止とソフトウェア起動 (BUSY = 0, STRT = 1) を同時にしないでください。</p>
bit14	INT : 割込み要求 フラグビット	<ul style="list-style-type: none"> ・ A/D 変換によって A/D データレジスタにデータがセットされれば, INT ビットは "1" にセットされます。 ・ INT ビットと割込み要求許可ビット (ADCS1 : INTE) が "1" の場合, 割込み要求を発生します。EI²OS が許可されていれば EI²OS が起動されます。 ・ 書込み時は, "0" で INT ビットがクリアされ, "1" では変化せず他への影響はありません。 ・ EI²OS の起動で INT ビットはクリアされます。 <p>(注意事項) INT ビットの "0" 書込みによるクリアは, A/D 停止中に行ってください。</p>
bit13	INTE : 割込み要求 許可ビット	<ul style="list-style-type: none"> ・ CPU への割込み出力の許可 / 禁止をするビットです。 ・ INTE ビットと, 割込み要求フラグビット (ADCS1 : INT) が "1" の場合, 割込み要求を発生します。 ・ EI²OS 使用時は "1" にセットしてください。
bit12	PAUS : 一時停止 フラグビット	<ul style="list-style-type: none"> ・ A/D 変換動作が一時停止した場合に "1" にセットされます。 ・ この A/D コンバータには, A/D データレジスタが 1 つしかないため, 連続変換モードを使用した場合に旧変換結果の CPU による読み出しが完了していなければ, 新しい変換結果の書込みにより旧変換データは失われてしまいます。したがって, 連続変換モードを使用する場合は, 基本的には EI²OS を用いて変換終了ごとに変換結果を自動的にメモリに転送するように設定しておく必要があります。ただし, 多重割込みなどで変換データの転送が次の変換の間に合わない場合が想定できます。PAUS ビットはその場合の対処として考えられた機能で, 変換終了後にデータレジスタの内容を EI²OS で転送するまでの間, PAUS ビットを "1" にセットし, その間は A/D 変換は停止し, 次の変換データを格納しないようになっています。その後, EI²OS で転送を終了すると A/D コンバータは自動的に変換を再開します。 <p>(注意事項) PAUS ビットは, EI²OS を使用した場合のみ有効です。</p>

表 19.3-1 A/D 制御ステータスレジスタ上位 (ADCS1) の各ビットの機能説明 (2 / 2)

ビット名		機 能
bit11, bit10	STS1, STS0 : A/D 起動要因 選択ビット	<ul style="list-style-type: none"> ・ A/D 変換の起動要因の選択を行います。 ・ 起動要因が兼用になっている場合には , 最初に発生した起動要因で起動します。 (注意事項) 起動要因は , 書換えと同時に変更されますので , A/D 変換動作中に書き換える場合には , 目的とする起動要因がない状態で切り換えてください。
bit9	STRT : A/D 変換起動 ビット	<ul style="list-style-type: none"> ・ A/D 変換動作をソフトウェア的に起動するビットです。 ・ STRT ビットに "1" を書き込むと A/D 変換が起動します。 ・ 停止変換モード時は , STRT ビットによる再起動はかかりません。 (注意事項) 強制停止とソフトウェア起動 (BUSY=0, STRT=1) を同時にしないでください。
bit8	予約 : 予約ビット	(注意事項) 必ず "0" を書き込んでください。

19.3.2 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) は、変換モードの選択をする機能があります。

■ A/D 制御ステータスレジスタ下位 (ADCS0)

図 19.3-3 A/D 制御ステータスレジスタ下位 (ADCS0) のビット構成

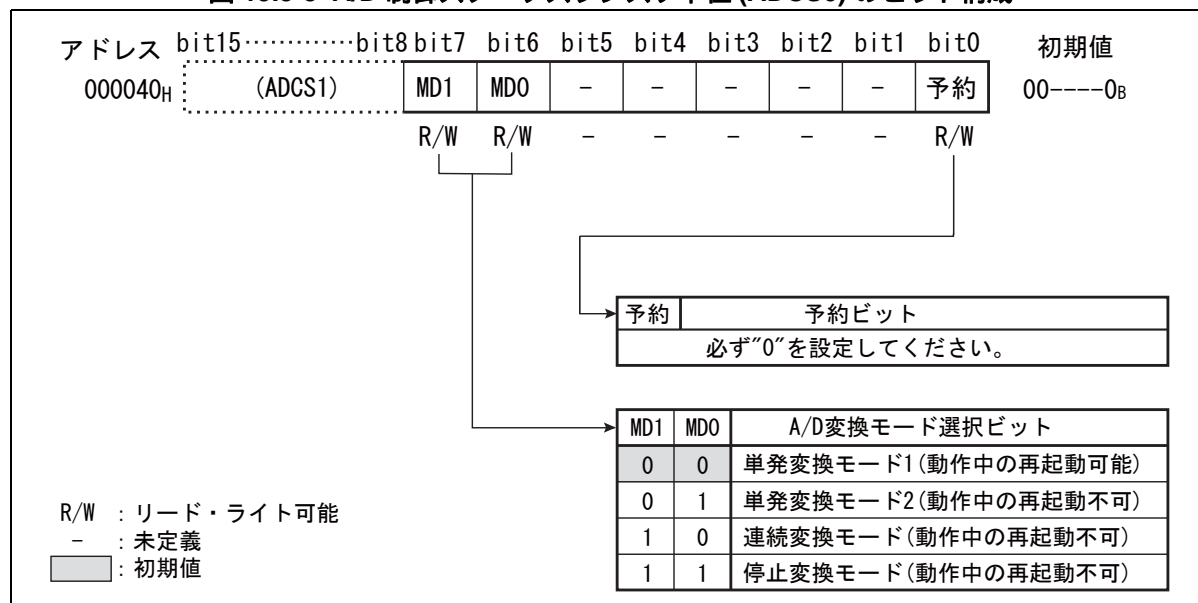


表 19.3-2 A/D 制御ステータスレジスタ下位 (ADCS0) の各ビットの機能説明

ビット名		機 能
bit7, bit6	MD1, MD0: A/D 変換 モード選択 ビット	<ul style="list-style-type: none"> ・ A/D 変換機能時の変換モードを選択するビットです。 ・ MD1, MD0 の 2 ビット値によって、単発変換モード 1, 単発変換モード 2, 連続変換モード, 停止変換モードのいずれかが選択されます。 ・ それぞれのモードの意味は、以下のとおりです。 <p>単発変換モード 1 : ANS3 ~ ANS0 の設定チャンネルから ANE3 ~ ANE0 の設定チャンネルまでの A/D 変換を連続して、1 度だけします。動作中の再起動が可能です。</p> <p>単発変換モード 2 : ANS3 ~ ANS0 の設定チャンネルから ANE3 ~ ANE0 の設定チャンネルまでの A/D 変換を連続して、1 度だけします。動作中の再起動はできません。</p> <p>連続変換モード : ANS3 ~ ANS0 の設定チャンネルから ANE3 ~ ANE0 の設定チャンネルまでの A/D 変換を連続して、BUSY ビットで強制停止するまで繰り返し行います。動作中の再起動はできません。</p> <p>停止変換モード : ANS3 ~ ANS0 の設定チャンネルから ANE3 ~ ANE0 の設定チャンネルまでの A/D 変換を 1 チャンネルごとに一時停止しながら、BUSY ビットで強制停止するまで繰り返します。動作中の再起動はできません。一時停止中の再起動は、STS1, STS0 ビットで選択した起動要因の発生によります。</p> <p>(注意事項) 単発、連続、停止の各変換モードの再起動不可は外部トリガ、ソフトウェアの起動に適用されます。</p>
bit5 ~ bit1	未定義 ビット	読出し値は不定です。書込みは動作に影響を与えません。
bit0	予約ビット	必ず "0" を設定してください。

19.3.3 A/D 変換チャネル設定レジスタ (ADMR)

A/D 変換チャネル設定レジスタ (ADMR) は、A/D 変換チャネルの選択をする機能があります。

■ A/D 変換チャネル設定レジスタ (ADMR)

図 19.3-4 A/D 変換チャネル設定レジスタ (ADMR) のビット構成

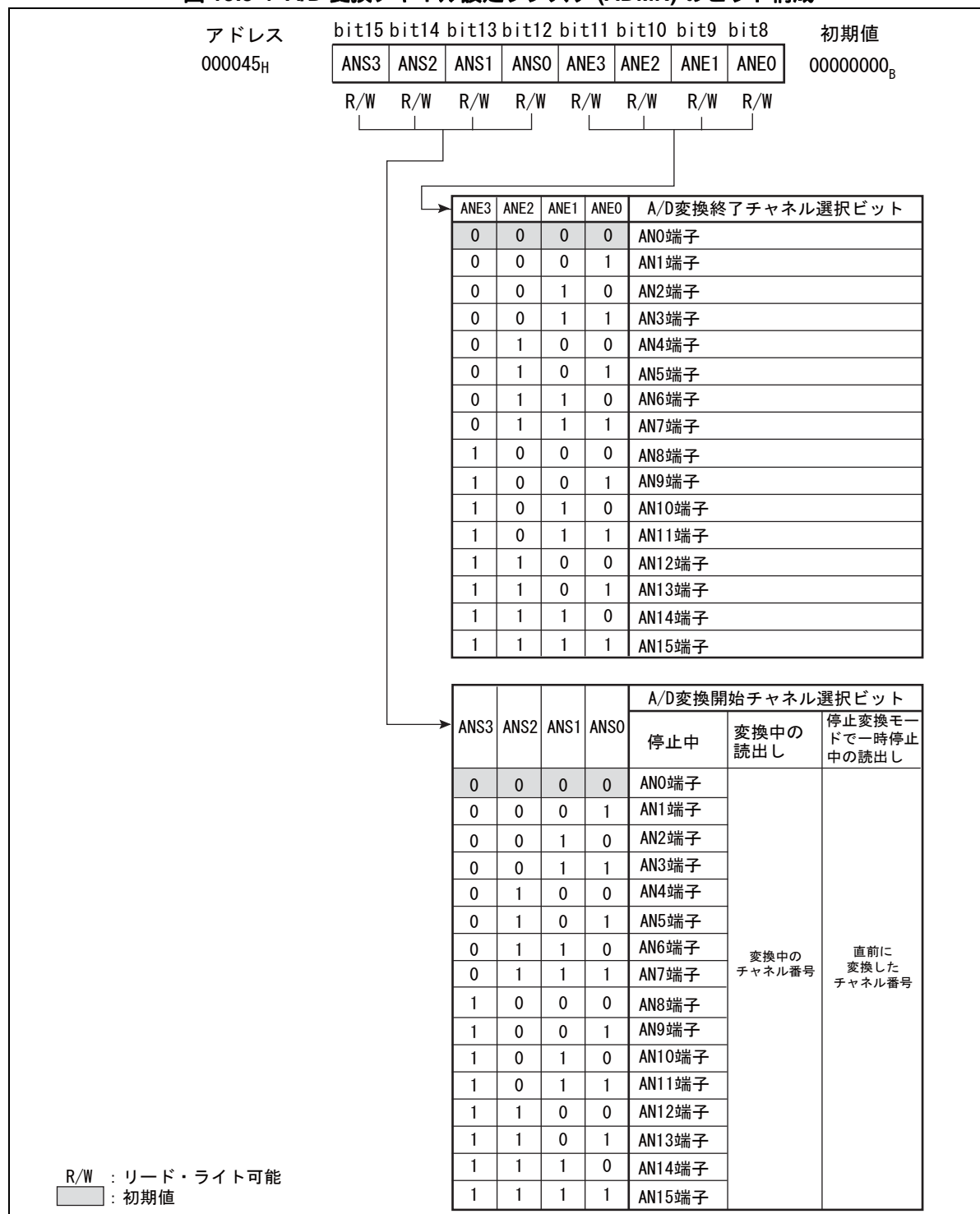


表 19.3-3 A/D 変換チャンネル設定レジスタ (ADMR) の各ビットの機能説明

ビット名		機 能
bit15 ~ bit12	ANS3, ANS2 ANS1, ANS0 : A/D 変換開始 チャンネル 選択ビット	<ul style="list-style-type: none"> ・ A/D 変換の開始チャンネルの設定および変換中チャンネル番号の確認を行うビットです。 ・ A/D 変換を起動するとこれらのビットに書き込まれたチャンネルから A/D 変換を開始します。 ・ A/D 変換中は、変換中のチャンネル番号が読み出せます。停止変換モードでの一時停止中は、直前に変換したチャンネルの番号が読み出せます。
bit11 ~ bit8	ANE3, ANE2 ANE1, ANE0 : A/D 変換終了 チャンネル 選択ビット	<ul style="list-style-type: none"> ・ A/D 変換の終了チャンネルの設定を行うビットです。 ・ A/D 変換を起動するとこれらのビットに書き込まれたチャンネルまで A/D 変換を行います。 ・ ANS3 ~ ANS0 と同じチャンネルを設定すると、そのチャンネルのみ変換を行います。また、連続変換モードまたは、停止変換モードを設定している場合は、これらのビットで設定されたチャンネルまでの変換が終わると ANS3 ~ ANS0 で設定された開始チャンネルに戻ります。 <p>(注意事項)</p> <p>A/D 変換開始チャンネル選択ビット (ANS3, ANS2, ANS1, ANS0) に開始チャンネルを設定したあとに、A/D 変換モード設定ビット (MD1, MD0) 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 (RMW) 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 ビットに開始チャンネルを設定したあとに、MD1, MD0 ビットおよび ANE3, ANE2, ANE1, ANE0 ビットをリードモディファイライト (RMW) 系命令で設定した場合、ANE3, ANE2, ANE1, ANE0 ビットの値が書き換わる可能性があります。</p>

< 注意事項 >

A/D 変換終了チャンネルが A/D 変換開始チャンネルよりも小さい設定 (ANS > ANE) はできません。A/D 変換チャンネル設定レジスタ (ADMR) は、バイトアクセスしてください。

19.3.4 A/D データレジスタ (ADCR1/ADCR0)

A/D データレジスタ (ADCR1/ADCR0) は、A/D 変換結果を格納するレジスタであり、A/D 変換の分解能を選択する機能もあります。

■ A/D データレジスタ (ADCR1/ADCR0)

図 19.3-5 A/D データレジスタ (ADCR1/ADCR0) のビット構成

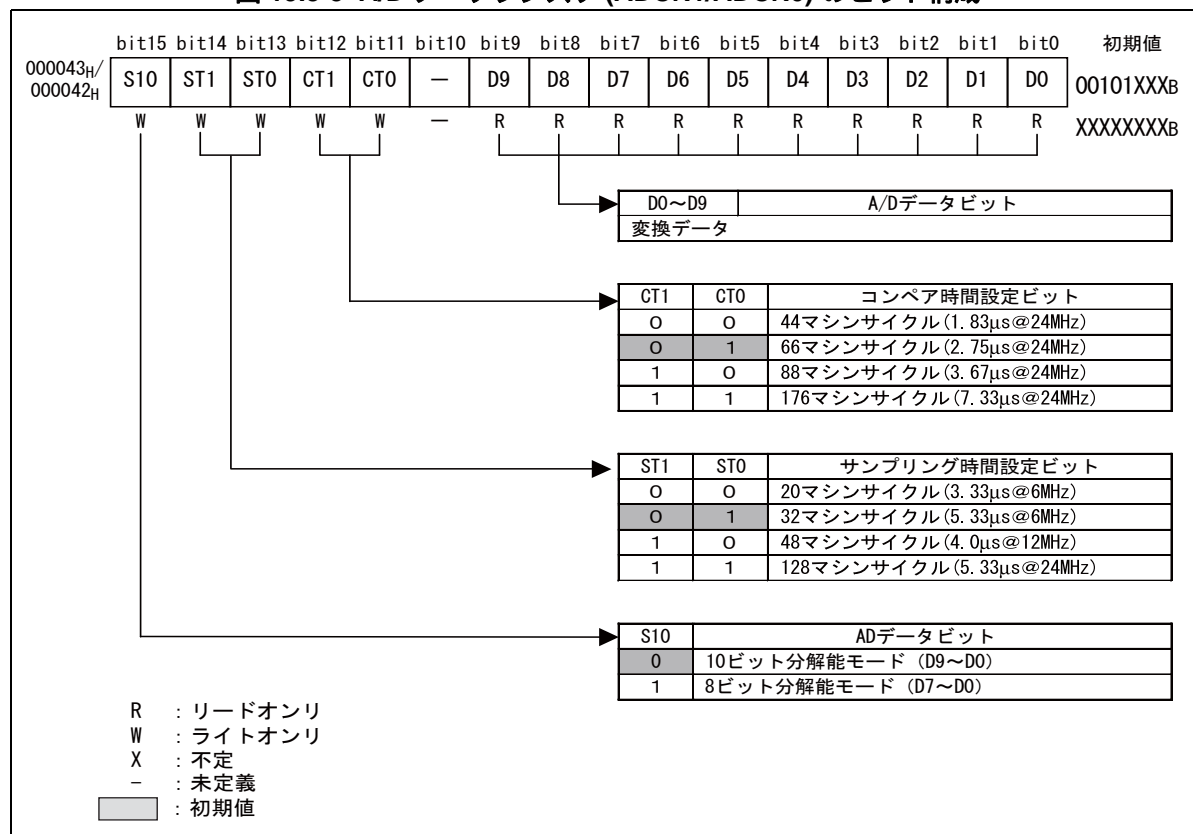


表 19.3-4 A/D データレジスタ (ADCR1/ADCR0) の各ビットの機能説明

ビット名		機 能
bit15	S10 : A/D 変換 分解能選択 ビット	<ul style="list-style-type: none"> ・ A/D 変換の分解能を選択するビットです。 ・ S10 ビットに "0" を書き込むと 10 ビット分解能が選択され , "1" を書き込むと 8 ビット分解能が選択されます。 (注意事項) 分解能によって , 使用されるデータビットが異なります。
bit14, bit13	ST1, ST0 : サンプリング 時間設定ビット	<ul style="list-style-type: none"> ・ A/D 変換時のサンプリング時間を選択するビットです。 ・ A/D が起動されると , ST1, ST0 ビットに設定された時間 , アナログ入力を取り込まれます。 (注意事項) 24 MHz 動作時に "00" 6 MHz 用の設定を行うと正常なアナログ電圧を取り込めない場合があります。
bit12, bit11	CT1, CT0 : コンペア時間 設定ビット	<ul style="list-style-type: none"> ・ A/D 変換時のコンペア時間を選択するビットです。 ・ アナログ入力を取り込まれた (サンプリング時間経過) 後 , CT1, CT0 ビットに設定された時間後に変換結果のデータが確定し , このレジスタの bit9 ~ bit0 に格納されます。 (注意事項) 24 MHz 動作時に "00" 6 MHz 用の設定を行うと正常なアナログ変換値が得られない場合があります。
bit10	未定義ビット	読出し値は不定です。書込みは動作に影響を与えません。
bit9 ~ bit0	D9 ~ D0	<ul style="list-style-type: none"> ・ A/D 変換の結果が格納され , レジスタは 1 回の変換終了ごとに書き換えられます。 ・ 通常は , 最終変換値が格納されます。 ・ ADCR レジスタの初期値は不定です。 (注意事項) 変換データ保護機能があります。(「19.4 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) A/D 変換中に D9 ~ D0 ビットにデータを書き込まないようにしてください。

< 注意事項 >

- ・ S10 ビットの書き換えは必ず変換動作前の A/D 動作が停止の状態で行ってください。変換後の書き換えをした場合 , ADCR の内容は不定となります。
- ・ ADCR レジスタの読出しは , 10 ビットモードを指定した場合は必ずワード転送命令 (MOVW A, 0042H など) を使用してください。

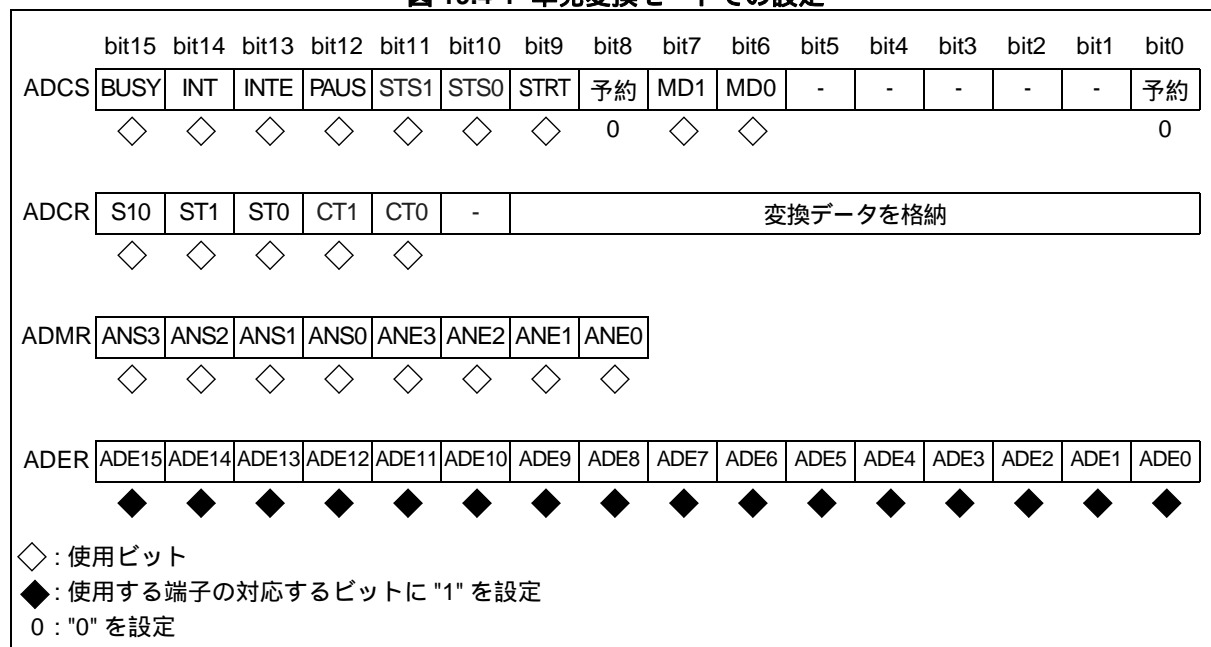
19.4 8/10 ビット A/D コンバータの動作説明

8/10 ビット A/D コンバータには、単発変換モード、連続変換モード、停止変換モードの 3 種類のモードがあります。各モードでの動作説明をします。

■ 単発変換モードの動作

単発変換モードは、ANS ビットと ANE ビットで設定されたアナログ入力を順次変換していき、ANE ビットで設定された終了チャンネルまで変換が終わると A/D 変換は停止します。開始チャンネルと終了チャンネルが同じ (ANS=ANE) 場合は ANS ビットで指定した 1 チャンネルだけの変換となります。単発変換モードで動作させるには、図 19.4-1 の設定が必要です。

図 19.4-1 単発変換モードでの設定



単発変換モードでの変換順序の例を以下に示します。

ANS=000_B, ANE=011_B の場合 : AN0 AN1 AN2 AN3 終了

ANS=011_B, ANE=011_B の場合 : AN3 終了

■ 連続変換モードの動作

連続変換モードは、ANS ビットと ANE ビットで設定されたアナログ入力を順次変換していき、ANE ビットで設定された終了チャンネルまで変換が終わると ANS ビットで設定されたアナログ入力に戻り、A/D 変換動作を続けます。開始チャンネルと終了チャンネルが同じ場合 (ANS=ANE) は ANS で指定したチャンネルだけの変換を繰り返します。連続変換モードで動作させるには、図 19.4-2 に示す設定が必要です。

図 19.4-2 連続変換モードでの設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ADCS	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	予約	MD1	MD0	-	-	-	-	-	予約
	◇	◇	◇	◇	◇	◇	◇	0	1	0						0
ADCR	S10	ST1	ST0	CT1	CT0	-	変換データを格納									
	◇	◇	◇	◇	◇											
ADMR	ANS3	ANS2	ANS1	ANS0	ANE3	ANE2	ANE1	ANE0								
	◇	◇	◇	◇	◇	◇	◇	◇								
ADER	ADE15	ADE14	ADE13	ADE12	ADE11	ADE10	ADE9	ADE8	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0
	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆

◇ : 使用ビット
◆ : 使用する端子の対応するビットに "1" を設定
1 : "1" を設定
0 : "0" を設定

連続変換モードでの変換順序の例を以下に示します。

ANS=000_B, ANE=011_B の場合 : AN0 AN1 AN2 AN3 AN0 繰返し

ANS=011_B, ANE=011_B の場合 : AN3 AN3 繰返し

■ 停止変換モードの動作

停止変換モードは、ANS ビットと ANE ビットで設定されたアナログ入力を 1 チャンネルごとに一時停止しながら変換していき、ANE ビットで設定された終了チャンネルまで変換が終わると ANS ビットで設定されたアナログ入力に戻り、A/D 変換と一時停止の動作を続けます。開始チャンネルと終了チャンネルが同じ場合 (ANS=ANE) は ANS ビットで指定したチャンネルだけの変換を繰り返します。一時停止時の変換の再起動は、STS1, STS0 ビットで指定した起動要因を発生します。停止変換モードで動作させるには、図 19.4-3 の設定が必要です。

図 19.4-3 停止変換モードでの設定

図 15-10 付立交換モードの設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ADCS	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	予約	MD1	MD0	--	--	--	--	--	--
	◇	◇	◇	◇	◇	◇	◇	0	1	1						0

ADCR	S10	ST1	ST0	CT1	CT0	-	変換データを格納										
	◇	◇	◇	◇	◇												

ADMR	ANS3	ANS2	ANS1	ANS0	ANE3	ANE2	ANE1	ANE0
	◇	◇	◇	◇	◇	◇	◇	◇

ADER	ADE15	ADE14	ADE13	ADE12	ADE11	ADE10	ADE9	ADE8	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0
	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆

◇ : 使用ビット
 ◆ : 使用する端子の対応するビットに "1" を設定
 1 : "1" を設定
 0 : "0" を設定

停止変換モードでの変換順序の例を以下に示します。

ANS=000_B, ANE=011_B の場合： AN0 一時停止 AN1 一時停止 AN2
 一時停止 AN0 繰返し

ANS=011_B, ANE=011_B の場合: AN3 一時停止 AN3 一時停止 繰返し

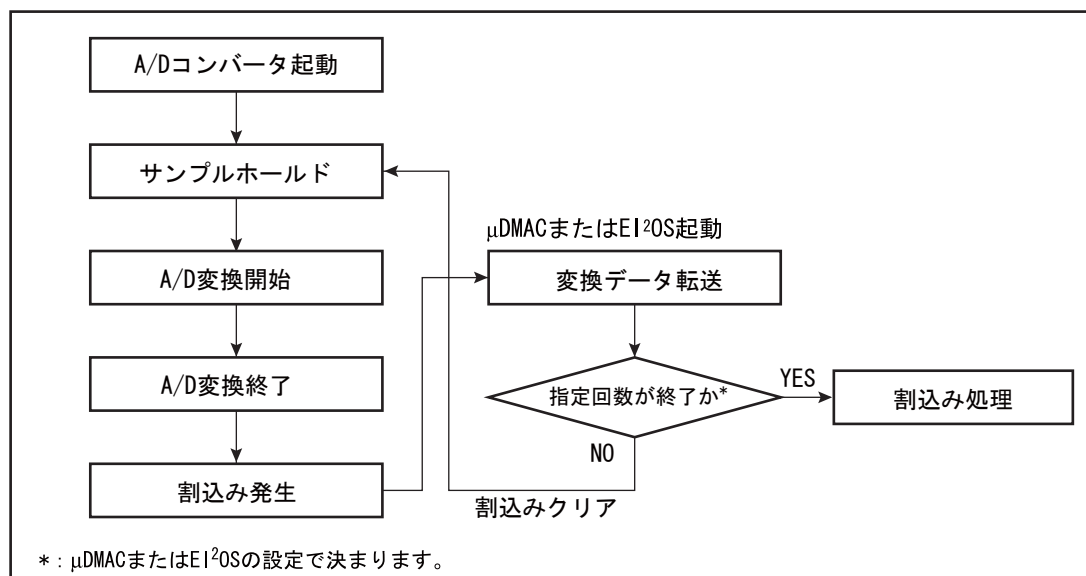
19.4.1 μ DMAC または EI²OS を使用した変換動作

10 ビット A/D コンバータは、 μ DMAC または EI²OS を使用して、A/D 変換結果をメモリに転送できます。

■ μ DMAC または EI²OS を使用した変換動作

μ DMAC または EI²OS 使用時の動作フローを図 19.4-4 に示します。

図 19.4-4 μ DMAC または EI²OS 使用時の動作フローチャート例



μ DMAC または EI²OS を利用すると変換データ保護機能で、連続変換時にもデータ欠落を起こすことなく、複数のデータを確実にメモリに転送できます。

19.4.2 A/D 変換データ保護機能

割込み許可状態で A/D 変換を実行すると、変換データ保護機能が働きます。

■ A/D 変換データ保護機能

この A/D コンバータでは、変換データ格納用のデータレジスタが 1 つしかないので、A/D 変換をすると、変換終了時に、データレジスタ内の格納データを書き換えます。そのため、変換データのメモリへの転送が間に合わないで前回のデータが一部欠落することになります。この対策として、割込み許可 (INTE=1) の場合は、以下のようにデータ保護機能が働くようになっています。

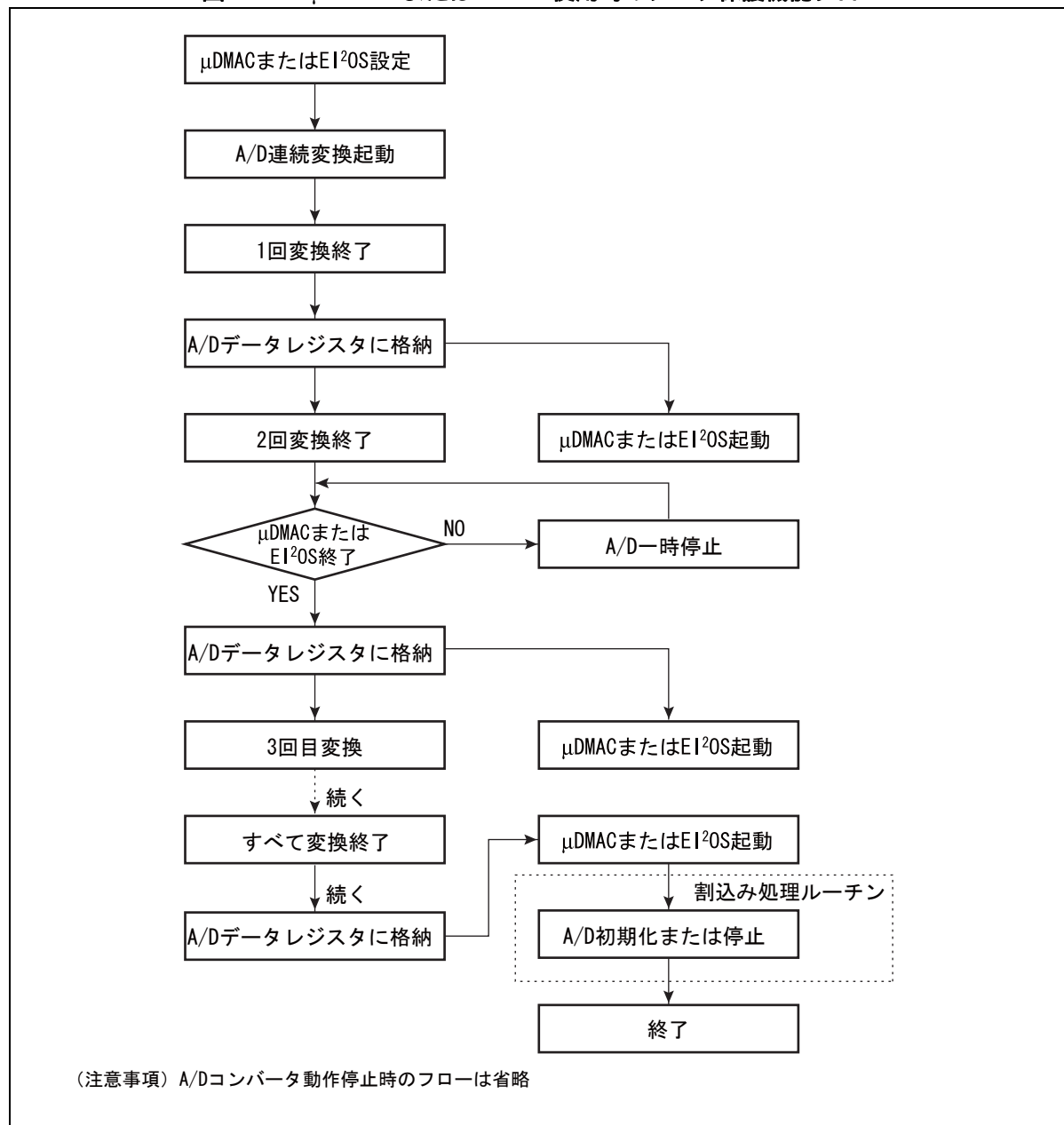
● μ DMAC または EI²OS を使用しない場合のデータ保護機能

変換データが 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上位 (ADCS1) の INT ビットが "1" にセットされます。この INT ビットが "1" の間、A/D 変換は一時停止状態になります。割込みルーチン内で、A/D データレジスタ (ADCR) をメモリなどに転送した後、INT ビットをクリアすると停止状態が解除されます。

● μ DMAC または EI²OS を使用する場合のデータ保護機能

μ DMAC または EI²OS を使って連続変換を指示すると、変換終了から μ DMAC または EI²OS でデータレジスタの変換データをメモリに転送完了するまで、A/D 制御ステータスレジスタ上位 (ADCS1) の PAUS ビットを "1" にセットし、この間、A/D 変換動作は停止し、次の変換データを格納しないようになっています。メモリへのデータ転送が完了すると、PAUS ビットは "0" にクリアされ、変換動作が再開します。図 19.4-5 に μ DMAC または EI²OS 使用時のデータ保護機能フローを示します。

図 19.4-5 μ DMAC または EI²OS 使用時のデータ保護機能フロー



< 注意事項 >

- 変換データ保護機能は、割り込み許可 (ADCS1:INTE=1) 状態でしか動作しません。
- μ DMAC または EI²OS 動作中で、A/D 変換が一時停止している場合に、割り込みを禁止すると A/D 変換が動作し、旧データの転送前に新データが書き込まれる場合があります。また、一時停止中に再起動がかかった場合も、旧データは破壊されます。
- 一時停止中に再起動をかけると待機データが壊れます。

19.5 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の注意点を示します。

■ 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

A/D 入力端子はポート 7, 8 の入出力端子と兼用になっており、ポート 7, 8 方向レジスタ (DDR7, DDR8) とアナログ入力許可レジスタ (ADER) で切り替えて使用できるようになっています。アナログ入力として使用する端子では、DDR7, DDR8 の対応するビットに "0" を書き込んでポート設定を入力にした上で、ADER でアナログ入力モード (ADE_x=1) に設定して、ポート側の入力ゲートを固定してください。ポート入力モード (ADE_x=0) の状態では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。

● 内部タイマ (16 ビットリロードタイマ 1) で使用する場合の注意

A/D コンバータを内部タイマで起動する場合、A/D 制御ステータスレジスタ上位 (ADCS1) の STS1, STS0 ビットで設定しますが、このとき、内部タイマの入力値は、インアクティブ側 (内部タイマの場合は "L") にしておいてください。アクティブ側にしておくと、ADCS1 レジスタへの書き込みと同時に動作し始める場合があります。

STS1, STS0 設定時は、ADTG=1 入力、内部タイマ (PPG1)=0 出力の状態で行ってください。

● A/D コンバータの電源・アナログ入力の投入順序

A/D コンバータの電源 (AV_{cc}, AVR_H) および、アナログ入力 (AN0 ~ AN15) への印加は、必ずデジタル電源 (V_{cc}) の投入後かまたは同時に投入してください。また、電源切断時は、A/D コンバータの電源およびアナログ入力の切断後にデジタル電源 (V_{cc}) を切断するかまたは同時に切断してください。

● 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、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してください。

19.6 8/10 ビット A/D コンバータのプログラム例 -1 (単発モードの EI²OS 起動例)

単発モードの EI²OS 起動による A/D 変換処理のプログラムを示します。

■ 単発モードの EI²OS 起動プログラム例

● 処理仕様

アナログ入力 AN1 ~ AN3 まで変換して終了します。

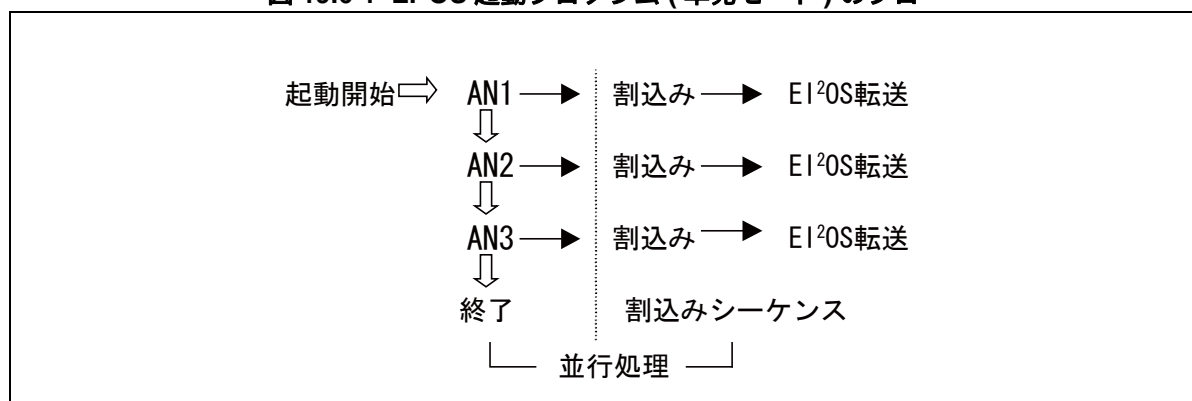
変換データは 200_H ~ 205_H 番地に順に転送します。

分解能は 10 ビットとします。

起動はソフトで行います。

図 19.6-1 に、EI²OS 起動プログラム (単発モード) のフローを示します。

図 19.6-1 EI²OS 起動プログラム (単発モード) のフロー



● コーディング例

BAPL	EQU	000100H	; バッファアドレスポインタ下位
BAPM	EQU	000101H	; バッファアドレスポインタ中位
BAPH	EQU	000102H	; バッファアドレスポインタ上位
ISCS	EQU	000103H	; EI ² OS ステータスレジスタ
IOAL	EQU	000104H	; I/O アドレスレジスタ下位
IOAH	EQU	000105H	; I/O アドレスレジスタ上位
DCTL	EQU	000106H	; データカウンタ下位
DCTH	EQU	000107H	; データカウンタ上位
DDR7	EQU	000017H	; ポート 7 方向レジスタ
ADER0	EQU	00001EH	; アナログ入力許可レジスタ 0
ICR12	EQU	0000BCH	; A/DC 用割込み制御レジスタ
ADMR	EQU	000045H	; A/D 変換チャネル設定レジスタ

```

ADCS0    EQU    000040H          ; A/D 制御ステータスレジスタ
ADCS1    EQU    000041H          ;
ADCR0     EQU    000042H          ; A/D データレジスタ
ADCR1     EQU    000043H          ;
;----- メインプログラム -----
CODE      CSEG
START:
; スタックポインタ (SP) などは初期化済みとする
        AND     CCR, #0BFH        ; 割込み禁止
        MOV     ICR12, #00H        ; 割込みレベル 0 (最強)
        MOV     BAPL, #00H        ; 変換データの格納先アドレスの設定
        MOV     BAPM, #02H        ; (200H ~ 205H を使用)
        MOV     BAPH, #00H        ;
        MOV     ISCS, #18H        ; ワードデータ転送, 転送後アドレス +1,
                                   ; I/O メモリに転送
        MOV     IOAL, #42H        ; 転送元アドレスポインタとして
        MOV     IOAH, #00H        ; アナログデータレジスタのアドレスを
                                   ; セット
        MOV     DCTL, #03H        ; EI2OS 転送を 3 回, 変換回数と同じに
                                   ; する
        MOV     DDR7, #11110001B  ; P71 ~ P73 を入力に設定
        MOV     DCTH, #00H        ;
        MOV     ADER0, #00001110B ; P71/AN1 ~ P73/AN3 をアナログ入力に
                                   ; 設定
        MOV     ADMR, #013H        ; AN1 ~ AN3 CH を変換
        MOV     ADCSL, #000H        ; 単発起動
        MOV     ADCSH, #0A2H        ; ソフト起動, A/D 変換開始, 割込み許可
        MOV     ILM, #07H          ; PS 内 ILM をレベル 7 に設定
        OR      CCR, #40H          ; 割込み許可
LOOP:    MOV     A, #00H            ; 無限ループ
        MOV     A, #01H
        BRA     LOOP
;----- 割込みプログラム -----
ED_INT1:
        MOV     I:ADCS1, #00H      ; A/D 停止, 割込みのフラグクリアと禁止
        RETI                        ; 割込みから復帰
CODE      ENDS

```

```
----- ベクタ設定 -----  
VECT      CSEG      ABS=0FFH  
           ORG        0FF6CH          ; 割込み #36(24H) にベクタを設定  
           DSL        ED_INT1  
           ORG        0FFDCH          ; リセットベクタ設定  
           DSL        START  
           DB          00H            ; シングルチップモードに設定  
VECT      ENDS  
          END        START
```

19.7 8/10 ビット A/D コンバータのプログラム例 -2 (連続モードの EI²OS 起動例)

連続モードの EI²OS 起動による A/D 変換処理のプログラムを示します。

■ 連続モードの EI²OS 起動プログラム例

● 処理仕様

アナログ入力 AN3 ~ AN5 の変換を 2 度行い, 各チャネルの変換データを 2 つ取得します。

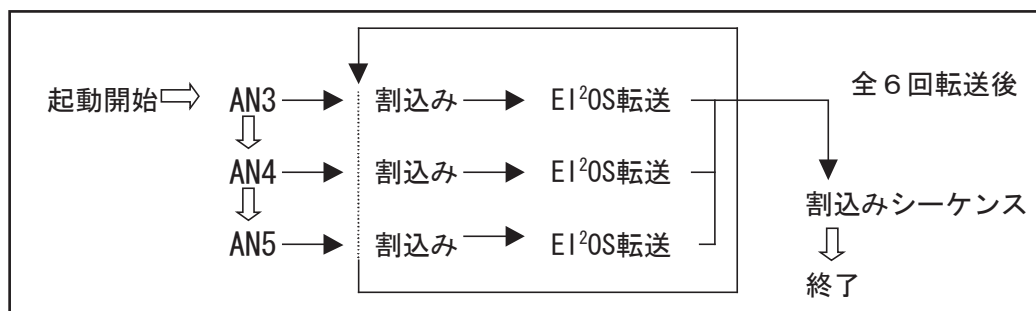
変換データは 600_H ~ 60B_H 番地に順に転送します。

分解能は 10 ビットとします。

起動は 16 ビットリロードタイマ 0 で行います。

図 19.7-1 に, EI²OS 起動プログラム (連続モード) のフローを示します。

図 19.7-1 EI²OS 起動プログラム (連続モード) のフロー



● コーディング例

BAPL	EQU	000100H	; バッファアドレスポインタ下位
BAPM	EQU	000101H	; バッファアドレスポインタ中位
BAPH	EQU	000102H	; バッファアドレスポインタ上位
ISCS	EQU	000103H	; EI ² OS ステータスレジスタ
IOAL	EQU	000104H	; I/O アドレスレジスタ下位
IOAH	EQU	000105H	; I/O アドレスレジスタ上位
DCTL	EQU	000106H	; データカウンタカウンタ下位
DCTH	EQU	000107H	; データカウンタ上位
DDR7	EQU	000017H	; ポート 7 方向レジスタ
ADER0	EQU	00001EH	; アナログ入力許可レジスタ 0
ICR12	EQU	0000BCH	; A/D 用割込み制御レジスタ
ADMR	EQU	000045H	; A/D 変換チャネル設定レジスタ
ADCS0	EQU	000040H	; A/D 制御ステータスレジスタ
ADCS1	EQU	000041H	;

```

ADCR0    EQU    000042H          ; A/D データレジスタ
ADCR1    EQU    000043H          ;
TMCSR1L  EQU    000062H          ; タイマコントロールステータスレジ
                                ; スタ 0 下位
TMCSR0H  EQU    000063H          ;
TMRLR0L  EQU    000064H          ; リロードレジスタ 0
TMRLR0H  EQU    000065H          ;
;----- メインプログラム -----
CODE      CSEG
START:
                                ; スタックポインタ (SP) などは初期化
                                ; 済みとする

                                AND      CCR, #0BFH          ; 割込み禁止
                                MOV      ICR12, #08H         ; 割込みレベル 0( 最強 ), 割込み許可
                                MOV      BAPL, #00H          ; 変換データの格納先アドレスの設定
                                MOV      BAPM, #06H          ; (600H ~ 60BH を使用 )
                                MOV      BAPH, #00H          ;
                                MOV      ISCS, #18H          ; ワードデータ転送, 転送後アドレス +1,
                                ; I/O   メモリに転送
                                MOV      IOAL, #42H          ; 転送元アドレスポインタとして
                                MOV      IOAH, #00H          ; アナログデータレジスタのアドレスを
                                ; セット
                                MOV      DCTL, #06H          ; EI2OS 転送を 6 回 , 3 チャネル × 2 回分
                                ; の転送
                                MOV      DCTH, #00H
                                MOV      DDR7, #00000000B    ; P70 ~ P77 を入力に設定
                                MOV      ADER0, #00111000B    ; P73/AN3 ~ P75/AN5 をアナログ入力に
                                ; 設定
                                MOV      ADMR, #035H         ; AN3 ~ AN5 CH を変換
                                MOV      ADCS0, #080H         ; 連続モード
                                MOV      ADCS1, #0A8H         ; 16 ビットタイマ起動 , A/D 変換開始 ,
                                ; 割込み許可
                                MOVW     TMRLR0L, #0320H      ; タイマ値を設定 800(320H) 66 μs
                                MOV      TMCSR0H, #00H        ; クロックソースを 83ns に設定 ,
                                ; 外部トリガ禁止
                                MOV      TMCSR0L, #12H        ; タイマ出力禁止, 割込み禁止, リロード
                                ; 許可
                                MOV      TMCSR0L, #13H        ; 16 ビットタイマ起動
                                MOV      ILM, #07H            ; PS 内 ILM をレベル 7 に設定
                                OR       CCR, #40H            ; 割込み許可

```

```
LOOP:    MOV    A, #00H           ; 無限ループ
          MOV    A, #01H
          BRA    LOOP
;----- 割込みプログラム -----
ED_INT1:
          MOV    I:ADCS1, #80H    ; A/Dは停止しない, 割込みのフラグクリアと禁止
          RETI                    ; 割込みから復帰
CODE     ENDS
;----- ベクタ設定 -----
VECT     CSEG    ABS=0FFH
          ORG     0FF7CH          ; 割込み #36(24H) にベクタを設定
          DSL     ED_INT1
          ORG     0FFDCH          ; リセットベクタ設定
          DSL     START
          DB      00H            ; シングルチップモードに設定
VECT     ENDS
          END     START
```


19.8 8/10 ビット A/D コンバータのプログラム例 -3 (停止モードの EI²OS 起動例)

停止モードの EI²OS 起動による A/D 変換処理のプログラムを示します。

■ 停止モードの EI²OS 起動プログラム例

● 処理仕様

アナログ入力 AN3 を一定期間で 12 回変換します。

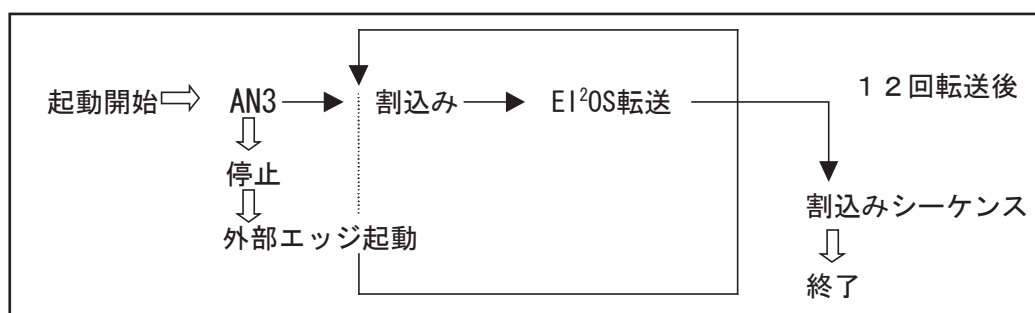
変換データは 600_H ~ 617_H 番地に順に転送します。

分解能は 10 ビットとします。

起動は 16 ビットリロードタイマ 0で行います。

図 19.8-1 に、EI²OS 起動プログラム (停止モード) のフローを示します。

図 19.8-1 EI²OS 起動プログラム (停止モード) のフロー



● コーディング例

BAPL	EQU	000100H	; バッファアドレスポインタ下位
BAPM	EQU	000101H	; バッファアドレスポインタ中位
BAPH	EQU	000102H	; バッファアドレスポインタ上位
ISCS	EQU	000103H	; EI ² OS ステータスレジスタ
IOAL	EQU	000104H	; I/O アドレスレジスタ下位
IOAH	EQU	000105H	; I/O アドレスレジスタ上位
DCTL	EQU	000106H	; データカウンタ下位
DCTH	EQU	000107H	; データカウンタ上位
DDR7	EQU	000017H	; ポート 7 方向レジスタ
ADER0	EQU	00001EH	; アナログ入力許可レジスタ 0
ICR12	EQU	0000BCH	; A/D 用割込み制御レジスタ
ADMR	EQU	000045H	; A/D 変換チャンネル設定レジスタ
ADCS0	EQU	000040H	; A/D 制御ステータスレジスタ

```

ADCS1    EQU    000041H    ;
ADCR0     EQU    000042H    ; A/D データレジスタ
ADCR1     EQU    000043H    ;
TMCSR0L   EQU    000062H    ; タイマコントロールステータスレジスタ 0 下位
TMCSR0H   EQU    000063H    ;
TMRLR0L   EQU    000064H    ; リロードレジスタ 0
TMRLR0H   EQU    000065H    ;
;----- メインプログラム -----
CODE      CSEG
START:
; スタックポインタ (SP) などは初期化済みとする

ANDCCR, #0BFH; 割込み禁止

        MOV     ICR12, #08H    ; 割込みレベル 0( 最強 )
        MOV     BAPL, #00H    ; 変換データの格納先アドレスの設定
        MOV     BAPM, #06H    ; (600H ~ 617H を使用 )
        MOV     BAPH, #00H    ;
        MOV     ISCS, #19H    ; ワードデータ転送, 転送後アドレス +1,
                                ; I/O メモリに転送, リソースの要求で終了
        MOV     IOAL, #42H    ; 転送元アドレスポインタとして
        MOV     IOAH, #00H    ; アナログデータレジスタのアドレスをセット
        MOV     DCTL, #0CH    ; EI2OS 転送を 12 回, 3 チャネルのみ
        MOV     DCTH, #00H    ;
        MOV     DDR7, #00000000B ; P70 ~ P77 を入力に設定
        MOV     ADER, #00001000B ; P73/AN3 をアナログ入力に設定
        MOV     ADMR, #033H    ; AN3 チャネルを変換
        MOV     ADCS0, #0C0H    ; 停止モード
        MOV     ADCS1, #0A8H    ; 16 ビットタイマ起動, A/D 変換開始, 割込み許可
        MOVW    TMRLR0L, #0320H ; タイマ値を設定 800(320H) 66 μs
        MOV     TMCSR0H, #00H    ; クロックソースを 83 ns に設定, 外部トリガ禁止
        MOV     TMCSR0L, #12H    ; タイマ出力禁止, 割込み禁止, リロード許可
        MOV     TMCSR0L, #13H    ; 16 ビットタイマ起動
        MOV     ILM, #07H    ; PS 内 ILM をレベル 7 に設定
        OR      CCR, #40H    ; 割込み許可

```

```
LOOP:    MOV     A, #00H           ; 無限ループ
          MOV     A, #01H
          BRA     LOOP

;----- 割込みプログラム -----
ED_INT1:
          MOV     I:ADCS1, #80H    ; A/Dは停止しない, 割込みのフラグクリアと禁止

          RETI                     ; 割込みから復帰
CODE      ENDS

;----- ベクタ設定 -----
VECT      CSEG      ABS=0FFH
          ORG      0FF6CH          ; 割込み #36(24H) にベクタを設定
          DSL      ED_INT1
          ORG      0FFDCH          ; リセットベクタ設定
          DSL      START
          DB       00H             ; シングルチップモードに設定
VECT      ENDS
          END      START
```

第20章

拡張 I/O シリアルインタ フェース

この章では、拡張 I/O シリアルインタフェースの概要、レジスタの構成 / 機能および拡張 I/O シリアルインタフェースの動作について説明します。

20.1 拡張 I/O シリアルインタフェースの概要

20.2 拡張 I/O シリアルインタフェースのレジスタ

20.3 拡張 I/O シリアルインタフェースの動作

20.1 拡張 I/O シリアルインタフェースの概要

拡張 I/O シリアルインタフェースは、8 ビット×1 チャンネル構成のクロック同期式によるデータ転送可能なシリアル I/O インタフェースです。また、データ転送において、LSB ファースト /MSB ファーストの選択が可能です。

■ 拡張 I/O シリアルインタフェースの概要

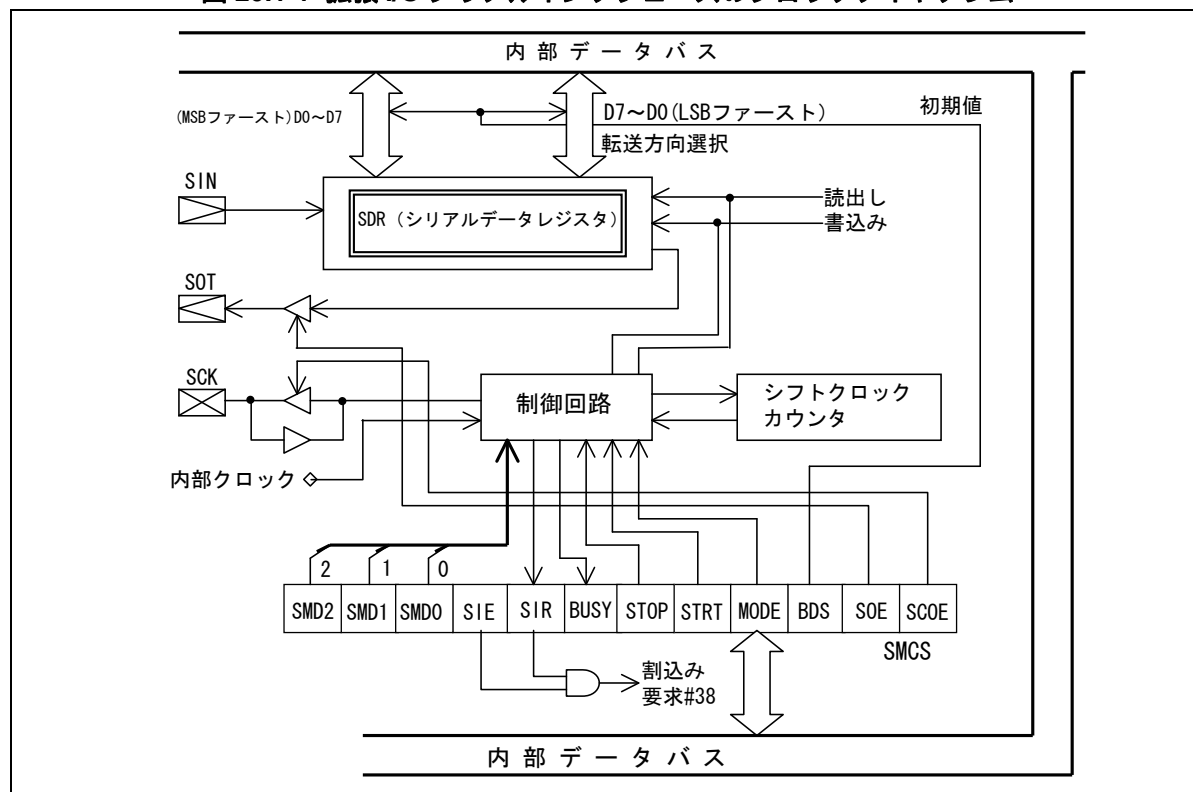
拡張 I/O シリアルインタフェースには、以下の 2 種類の動作モードがあります。

- ・ 内部シフトクロックモード：内部クロックに同期してデータ転送を行います。
- ・ 外部シフトクロックモード：外部端子（SCK）から入力されるクロックに同期してデータ転送を行います。このモードで外部端子（SCK）を共有している汎用ポートを操作することにより、CPU の命令による転送動作も可能です。

■ 拡張 I/O シリアルインタフェースのブロックダイアグラム

図 20.1-1 に、拡張 I/O シリアルインタフェースのブロックダイアグラムを示します。

図 20.1-1 拡張 I/O シリアルインタフェースのブロックダイアグラム



20.2 拡張 I/O シリアルインタフェースのレジスタ

拡張 I/O シリアルインタフェースで使用するレジスタの構成および機能について説明します。

■ 拡張 I/O シリアルインタフェースのレジスタ一覧

図 20.2-1 に、拡張 I/O シリアルインタフェースのレジスタ一覧を示します。

図 20.2-1 拡張 I/O シリアルインタフェースのレジスタ一覧

SMCS											
		bit	15	14	13	12	11	10	9	8	シリアルモード コントロール ステータスレジスタ (SMCS) 初期値 00000010 _B
アドレス：000059 _H			SMD2	SMD1	SMD0	SIE	SIR	BUSY	STOP	STRT	
			(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R)	(R/W)	(R/W)	
SMCS											
		bit	7	6	5	4	3	2	1	0	シリアルモード コントロール ステータスレジスタ (SMCS) 初期値 XXXX0000 _B
アドレス：000058 _H			-	-	-	-	MODE	BDS	SOE	SCOE	
			(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
SDR											
		bit	7	6	5	4	3	2	1	0	シリアルデータ レジスタ (SDR) 初期値 XXXXXXXX _B
アドレス：00005A _H			D7	D6	D5	D4	D3	D2	D1	D0	
			(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
SDCR											
		bit	15	14	13	12	11	10	9	8	通信プリスケアラ 制御レジスタ (SDCR) 初期値 0XXX0000 _B
アドレス：00005B _H			MD	-	-	-	DIV3	DIV2	DIV1	DIV0	
			(R/W)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
R/W：リード・ライト可能											
-：未定義											
X：不定											

20.2.1 シリアルモードコントロールステータスレジスタ (SMCS)

シリアルモードコントロールステータスレジスタ (SMCS) の構成および機能について説明します。

■ シリアルモードコントロールステータスレジスタ (SMCS)

シリアルモードコントロールステータスレジスタ (SMCS) は、シリアル I/O の転送動作モードを制御するレジスタです。

図 20.2-2 に、シリアルモードコントロールステータスレジスタ (SMCS) のビット構成を示します。

図 20.2-2 シリアルモードコントロールステータスレジスタ (SMCS) のビット構成

SMCS									
	bit	15	14	13	12	11	10	9	8
アドレス: 000059 _H		SMD2	SMD1	SMD0	SIE	SIR	BUSY	STOP	STRT
		(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)
									シリアルモード コントロール ステータスレジスタ 初期値 00000010 _B
SMCS									
	bit	7	6	5	4	3	2	1	0
アドレス: 000058 _H		-	-	-	-	MODE	BDS	SOE	SCOE
		(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)
									シリアルモード コントロール ステータスレジスタ 初期値 XXXX0000 _B
R/W: リード・ライト可能									
- : 未定義									

以下に、シリアルモードコントロールステータスレジスタ (SMCS) の各ビットの機能について説明します。

[bit15 ~ bit13] SMD2 ~ SMD0 (シフトクロック選択)

シリアルシフトクロックモードを選択します。

シリアルシフトクロックモードの設定内容を、表 20.2-1, 表 20.2-2 に示します。

表 20.2-1 シリアルシフトクロックモードの選択

SMD2	SMD1	SMD0	$\phi=24$ MHz div=8	$\phi=12$ MHz div=4	$\phi=6$ MHz div=6	分周値
0	0	0	1.5 MHz	1.5 MHz	500 kHz	2
0	0	1	750 kHz	750 kHz	250 kHz	4
0	1	0	188 kHz	188 kHz	62.5 kHz	16
0	1	1	93.4 kHz	93.4 kHz	31.2 kHz	32
1	0	0	46.9 kHz	46.9 kHz	15.6 kHz	64
1	0	1	外部シフトクロックモード			
1	1	0	設定禁止			
1	1	1	設定禁止			

表 20.2-2 通信プリスケラ (SDCR) の設定例

Div	(マシンクロック)					マシンサイクル (推奨設定)
	MD	DIV3	DIV2	DIV1	DIV0	
1	1	0	0	0	0	3 MHz
2	1	0	0	0	1	6 MHz
4	1	0	0	1	1	12 MHz
8	1	0	1	1	1	24 MHz

リセットにより "000_B" に初期化されます。転送中の書換えは禁止です。

シフトクロックは、内部シフトクロック5種類と、外部シフトクロックが選択できます。外部シフトします。SMD2, SMD1, SMD0=110_B, 111_B は設定しないでください。

クロック選択で SCOE=0 とし、SCK 端子を共有しているポートを操作することによって、命令ごとにシフト動作させることもできます。

[bit12] SIE (シリアル I/O 割込み許可)

シリアル I/O の割込み要求を制御します。

SIE	動作
0	シリアル I/O 割込み禁止 [初期値]
1	シリアル I/O 割込み許可

- このリセットにより "0" に初期化されます。
- このビットは、読出し / 書込みが可能です。

[bit11] SIR (シリアル I/O 割込み要求)

シリアルデータの転送が終了すると "1" にセットされ、割込み許可時 (SIE=1) にこのビットが "1" になると、CPU へ割込み要求が発生します。クリア条件は、MODE ビットによって異なります。

- MODE ビットが "0" のとき、SIR ビットへの "0" 書込みによってクリアされます。
- MODE ビットが "1" のとき、SDR の読出しまたは書込み動作によってクリアされます。
- MODE ビットの値にかかわらずリセットまたは STOP ビットへの "1" 書込み動作によってクリアされます。
- "1" を書き込んでも意味がありません。
- リードモディファイライト(RMW)系命令の読出し時は、常に、"1"が読み出されます。

[bit10] BUSY(転送状態表示)

シリアル転送が実行中か否かを示すビットです。

BUSY	動作
0	停止、またはシリアルデータレジスタリード / ライト待機状態 [初期値]
1	シリアル転送状態

- このリセットにより "0" に初期化されます。
- このビットは、読出しのみ可能です。

[bit9] STOP(ストップビット)

シリアル転送を強制的に中断させるビットです。"1" にすると、STOP=1 による停止状態になります。

STOP	動作
0	通常動作
1	STOP=1 による転送停止 [初期値]

- このリセットにより "1" に初期化されます。
- このビットは、読出し / 書込みが可能です。

[bit8] STRT (スタートビット)

シリアル転送を起動するビットです。停止状態で "1" を書き込むことによって転送が開始されます。シリアル転送動作中およびシリアルシフトレジスタ R/W 待機中の "1" のラインは無視されます。

- "0" の書込みは意味がありません。
- 常に "0" が読み出されます。

[bit7 ~ bit4] 未定義ビット

リード値は不定です。ライトは動作に影響を与えません。

[bit3] MODE(シリアルモード選択)

停止状態からの起動条件を選択します。ただし、動作中の書換えは禁止です。

MODE	動作
0	STRT=1 にすることにより起動します。 [初期値]
1	シリアルデータレジスタの読出し / 書込みにより起動します。

- このリセットにより "0" に初期化されます。
- このビットは、読出し / 書込みが可能です。
- μ DMAC を起動する際には、"1" にしておいてください。

[bit2] BDS (転送方向選択)

シリアルデータ入出力時に、最下位ビット側から先に転送するか (LSB ファースト)、最上位ビット側から先に転送するか (MSB ファースト) を選択します。

BDS	動作
0	LSB ファースト [初期値]
1	MSB ファースト

- このリセットにより "0" に初期化されます。
- このビットは、読出し / 書込みが可能です。

< 注意事項 >

転送方向の選択は、SDR にデータを書き込む前に、設定しておいてください。

[bit1] SOE (シリアル出力許可)

シリアル I/O 用出力外部端子 (SOT) の出力を制御します。

SOE	動作
0	汎用ポート端子 [初期値]
1	シリアルデータ出力

- このリセットにより "0" に初期化されます。
- このビットは、読出し / 書込みが可能です。

[bit0] SCOE (シフトクロック出力許可)

シフトクロック用入出力外部端子 (SCK1, 2) の出力を制御します。

SCOE	動作
0	汎用ポート端子, 命令ごと転送時 [初期値]
1	シフトクロック出力端子

- このリセットにより "0" に初期化されます。
- このビットは, 読出し / 書込みが可能です。

外部シフトクロックモードで命令ごとに転送を行うときは, "0" に設定してください。

20.2.2 シリアルデータレジスタ (SDR)

シリアルデータレジスタ (SDR) の構成および機能について説明します。

■ シリアルデータレジスタ (SDR)

図 20.2-3 に、シリアルデータレジスタ (SDR) のビット構成を示します。

図 20.2-3 シリアルデータレジスタ (SDR) のビット構成

SDR										
	bit	7	6	5	4	3	2	1	0	シリアルデータ レジスタ 初期値 XXXXXXXX _B
アドレス: 00005A _H		D7	D6	D5	D4	D3	D2	D1	D0	
		(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										

シリアルデータレジスタ (SDR) は、シリアル I/O の転送データを保持するシリアルデータレジスタです。

転送中の SDR への書込みおよび読出しは禁止です。

20.2.3 通信プリスケアラ制御レジスタ (SDCR)

通信プリスケアラ制御レジスタ (SDCR) の構成および機能について説明します。

■ 通信プリスケアラ制御レジスタ (SDCR)

図 20.2-4 に、通信プリスケアラ制御レジスタ (SDCR) のビット構成を示します。

図 20.2-4 通信プリスケアラ制御レジスタ (SDCR) のビット構成

SDCR								
	bit 15	14	13	12	11	10	9	8
アドレス: 00005B _H	MD	-	-	-	DIV3	DIV2	DIV1	DIV0
	(R/W)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)
								通信プリスケアラ 制御レジスタ 初期値 0XXX0000 _B
R/W: リード・ライト可能								
- : 未定義								

以下に、通信プリスケアラ制御レジスタ (SDCR) の各ビットの機能を説明します。

[bit15] MD

通信プリスケアラの動作を許可するビットです。

MD	動作
0	通信プリスケアラは停止します。
1	通信プリスケアラは動作します。

[bit11 ~ bit8] DIV3 ~ DIV0:DIVide3 ~ DIVide0

マシクロックの分周比を決定します

DIV3 ~ DIV0	分周比
0000 _B	1 分周
0001 _B	2 分周
0010 _B	3 分周
0011 _B	4 分周
0100 _B	5 分周
0101 _B	6 分周
0110 _B	7 分周
0111 _B	8 分周

< 注意事項 >

分周比を変えた場合には、クロックの安定時間として 2 分周分の時間をおいてから通信を行ってください。

20.3 拡張 I/O シリアルインタフェースの動作

拡張 I/O シリアルインタフェースは、シリアルモードコントロールステータスレジスタ (SMCS) およびシリアルデータレジスタ (SDR) によって構成され、8 ビットのシリアルデータの入出力に使用します。

拡張 I/O シリアルインタフェースの動作について説明します。

■ 拡張 I/O シリアルインタフェースの動作概要

シリアルデータの入出力は、入力、出力それぞれが以下のように動作します。

● シリアルデータの入力

シリアルデータレジスタの内容がシリアルシフトクロック (外部クロック、内部クロック) の立下りに同期してビット直列にシリアル出力端子 (SOT 端子) に出力されます。

● シリアルデータの出力

シリアルシフトクロック (外部クロック、内部クロック) の立上りに同期してシリアル入力端子 (SIN 端子) からビット直列に SDR (シリアルデータレジスタ) に入力されます。シフトの方向 (MSB からの転送または LSB からの転送) は、シリアルモードコントロールステータスレジスタ (SMCS) の方向指定ビット (BDS) により指定できます。

転送が終了すると、シリアルモードコントロールステータスレジスタ (SMCS) の MODE ビットによって停止状態またはシリアルデータレジスタ R/W 待機状態に入ります。各々の状態から転送状態に移るには、それぞれ次のことを行います。

- 停止状態から復帰するときは、STOP ビットに "0"、STRT ビットに "1" を書き込みます (STOP と STRT は同時設定が可能です)。
- シリアルデータレジスタ R/W 待機状態から復帰するときは、データレジスタを読出しまたは書込みをします。

20.3.1 シフトクロックモード

シフトクロックには、内部シフトクロックモードおよび外部シフトクロックモードの 2 種類があり、SMCS の設定で指定します。モードの切換えは、シリアル I/O が停止した状態で行ってください。停止状態の確認は、BUSY ビットを読み出すことによって可能となります。

■ 内部シフトクロックモード

内部クロックによって動作し、同期タイミング出力としてデューティ比 50% のシフトクロックが SCK 端子より出力されます。

データが、1 クロックごとに 1 ビット転送されます。

転送の速度は、次式によって算出できます。

$$\text{転送速度 (S)} = \frac{A}{\text{内部のクロックマシンサイクル (A)}}$$

A は、SMCS の SMD ビットで示す分周比です。

$$(\phi \div \text{div})/2, (\phi \div \text{div})/2^2, (\phi \div \text{div})/2^4, (\phi \div \text{div})/2^5, (\phi \div \text{div})/2^6$$

■ 外部シフトクロックモード

SCK 端子から入力される外部シフトクロックに同期して、データが 1 クロックごとに 1 ビット転送されます。

転送速度は、DC から 1/(8 マシンサイクル) まで可能です。例えば、1 マシンサイクル = 62.5 ns のとき、2 MHz まで可能です。

命令ごとに転送することもでき、以下の設定をすることによって実現します。

- 外部シフトクロックモードを選択し、SMCS の SCOE ビットを "0" に設定します。
- SCK 端子を共有しているポートの方向レジスタに "1" を書き込み、ポートを出力モードに設定します。

以上の設定をした後、ポートのデータレジスタ (PDR) に "1", "0" を書き込むと、SCK 端子に出力されるポートの値を外部クロックとして取り込み、転送動作を行います。シフトクロックは、"H" から開始させてください。

< 注意事項 >

シリアル I/O 動作中の SMCS.SDR への書込みは禁止です。

20.3.2 シリアル I/O の動作状態

シリアル I/O の動作状態として、STOP 状態、停止状態、SDR の R/W 待機状態および転送状態の 4 種類の状態があります。

■ シリアル I/O の動作状態

● STOP 状態

リセット時または SMCS の STOP ビットに "1" を書き込んだときの状態でシフトカウンタは初期化され、SIR=0 となります。

STOP 状態からの復帰は、STOP=0, STRT=1(同時設定可)とすることによって行われます。STOP ビットは STRT ビットより優先ビットですので、STOP=1 のとき STRT=1 としても転送動作は行われません。

● 停止状態

MODE ビットが "0" のとき、転送が終了したことによって SMCS が BUSY=0, SIR=1 となり、カウンタが初期化されて停止状態に入ります。停止状態からの復帰は、STRT=1 にすると転送動作が再開されます。

● シリアルデータレジスタ R/W 待機状態

SMCS の MODE ビットが "1" のとき、シリアル転送が終了すると、BUSY=0, SIR=1 となりシリアルデータレジスタ R/W 待機状態に入ります。割込み許可レジスタが許可状態ならば、本ブロックより割込み信号を出します。

R/W 待機状態からの復帰は、シリアルデータレジスタが読み出されたり、書き込まれたりすると、BUSY=1 となり、転送動作が再開されます。

● 転送状態

BUSY=1 でシリアル転送を行っている状態です。MODE ビットにより、それぞれ停止状態および R/W 待機状態に遷移します。

図 20.3-1 に各状態の動作遷移図を、図 20.3-2 にシリアルデータレジスタの読出し / 書込みの概念図を示します。

図 20.3-1 拡張 I/O シリアルインタフェースの動作遷移図

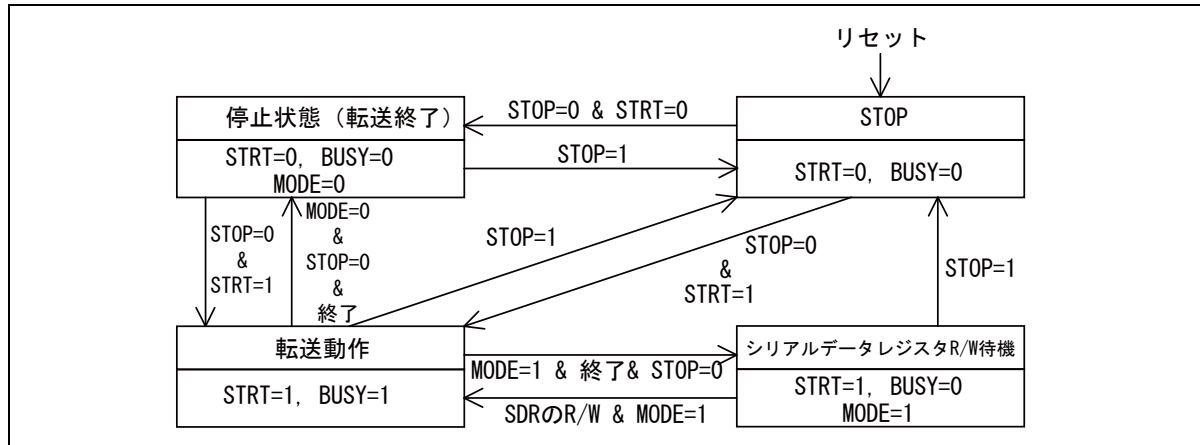
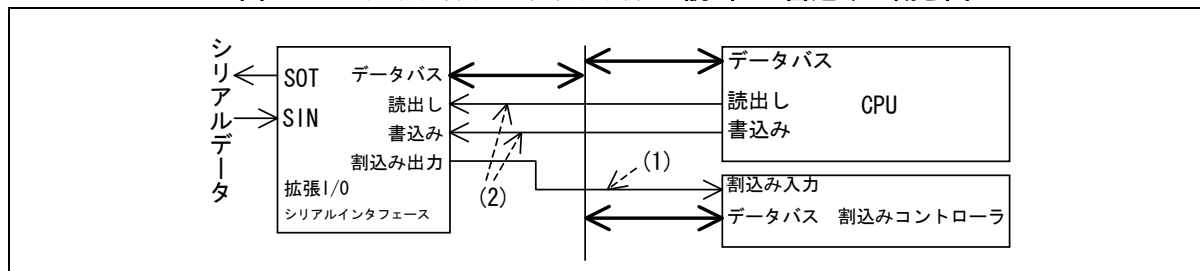


図 20.3-2 シリアルデータレジスタの読出し / 書込みの概念図



- (1) MODE=1 のとき、シフトクロックカウンタによって転送終了し、SIR=1 となって読出し / 書込み待機状態に入ります。SIE ビットが "1" ならば、割込み信号を発生します。ただし、SIE がインアクティブのときや STOP へ "1" を書き込むことによる転送中断のときは、割込み信号は発生しません。
- (2) シリアルデータレジスタが読出し / 書込みされると、割込み要求はクリアされ、シリアル転送を開始します。

20.3.3 シフト動作のスタート/ストップタイミングと入出力のタイミング

シフト動作のスタート/ストップタイミングと入出力のタイミングについて説明します。

■ シフト動作のスタート/ストップタイミングと入出力のタイミング

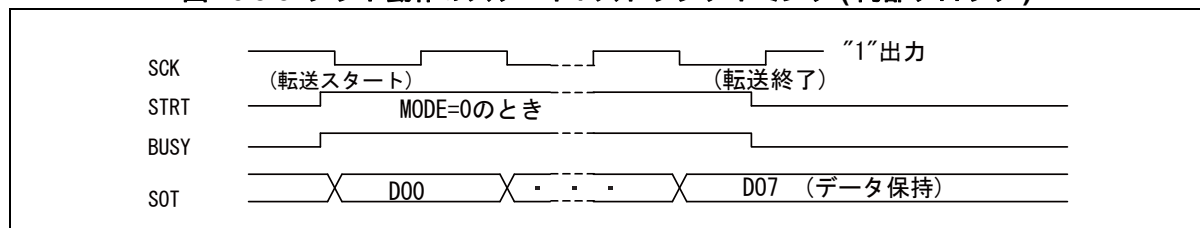
- スタート
SMCS の STOP ビットを "0", STRT ビット "1" に設定します。
- ストップ
転送終了によって停止する場合と, STOP=1 によって停止する場合があります。
 - STOP=1 によって停止 : MODE ビットにかかわらず SIR=0 のまま停止します。
 - 転送終了で停止 : MODE ビットにかかわらず, SIR=1 となり停止します。

BUSY ビットは, MODE ビットにかかわらずシリアル転送状態の場合 "1" となり, 停止状態または R/W 待機状態の場合 "0" となります。転送状態を確認するときは, このビットを読み出してください。

以下に, 各モードの動作および停止動作におけるタイミングチャートを示します。図中の D07 ~ D00 は, 出力データを示します。

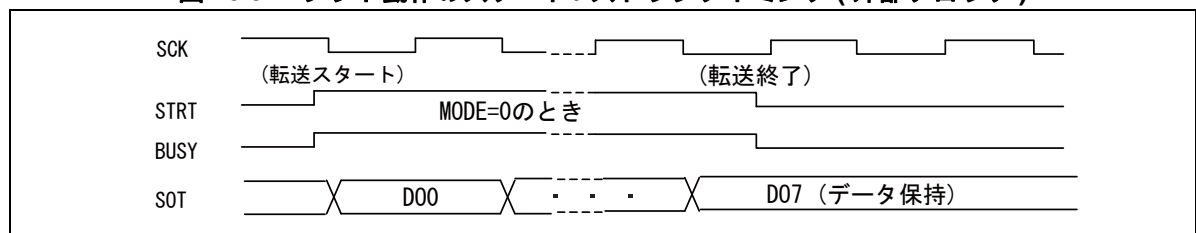
● 内部シフトクロックモード (LSB ファースト)

図 20.3-3 シフト動作のスタート/ストップタイミング (内部クロック)



● 外部シフトクロックモード (LSB ファースト)

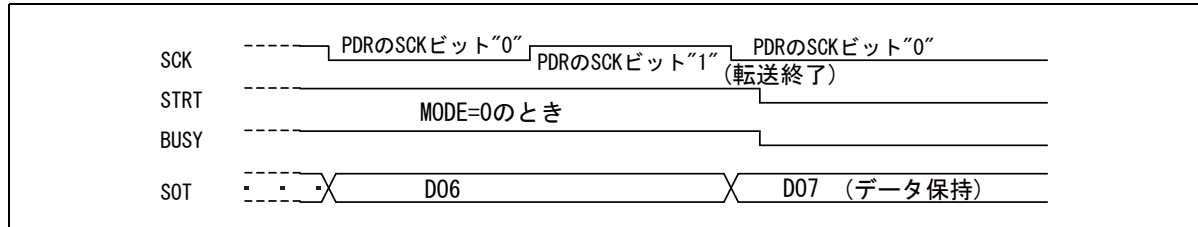
図 20.3-4 シフト動作のスタート/ストップタイミング (外部クロック)



● 外部シフトクロックモードで命令シフトを行った場合 (LSB ファースト)

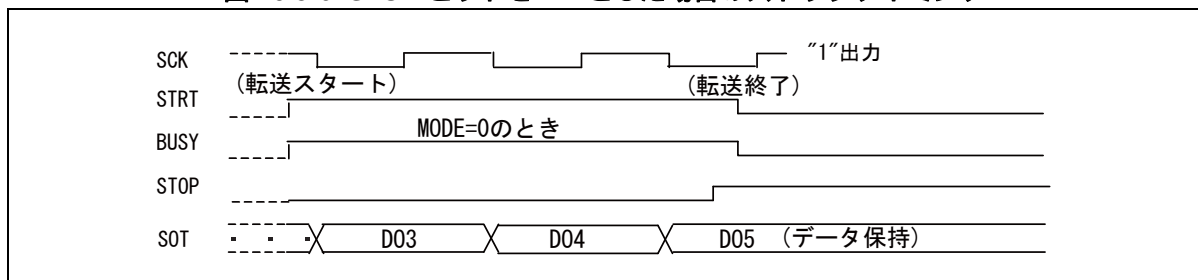
命令シフトでは, PDR の SCK に対応するビットに "1" が書き込まれたとき "H" が出力され, "0" が書き込まれたとき "L" が出力されます (ただし, 外部シフトクロックモード選択で, SCOE=0 のとき)。

図 20.3-5 外部シフトクロックモードで命令シフトを行った場合



● STOP=1 による停止 (LSB ファースト, 内部クロック時)

図 20.3-6 STOP ビットを "1" とした場合のストップタイミング

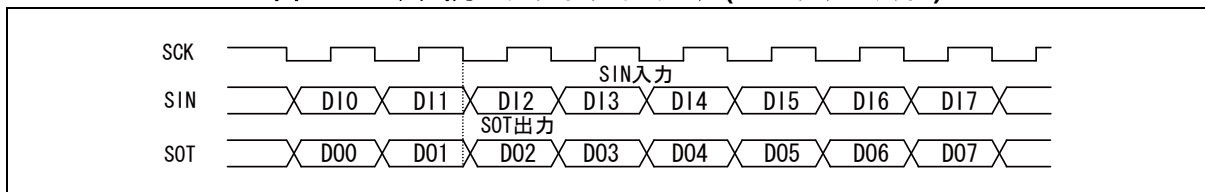


■ シリアルデータ転送中の動作

シリアルデータの転送中は, シフトクロックの立下りでシリアル出力端子 (SOT) からのデータが出力され, 立上りでシリアル入力端子 (SIN) のデータが入力されます。

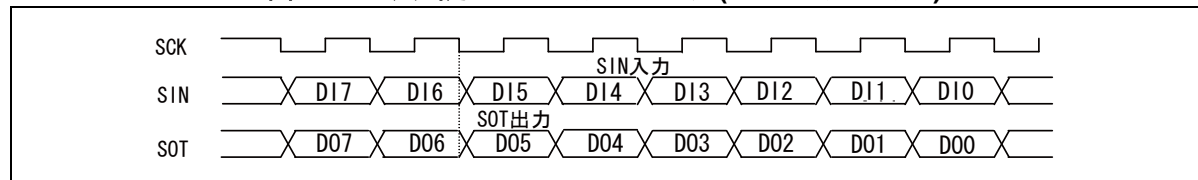
● LSB ファースト (BDS ビットが "0" のとき)

図 20.3-7 入出力のシフトタイミング (LSB ファースト)



● MSB ファースト (BDS ビットが "1" のとき)

図 20.3-8 入出力のシフトタイミング (MSB ファースト)



20.3.4 割込み機能

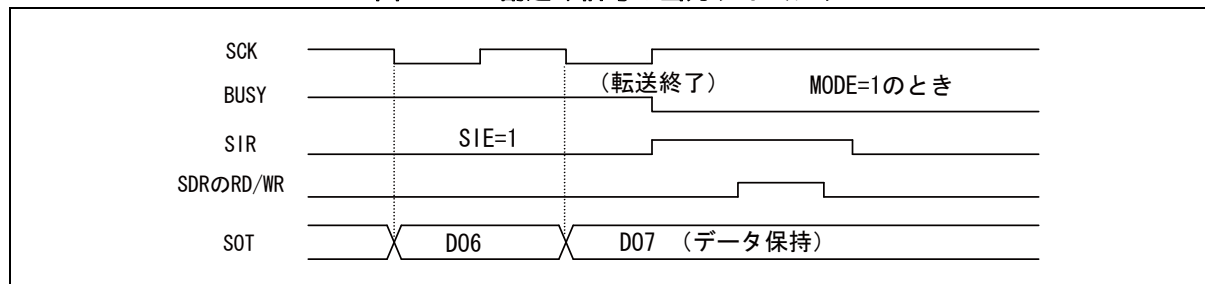
拡張 I/O シリアルインタフェースは、CPU に対し割込み要求を発生できます。

■ 拡張 I/O シリアルインタフェースの割込み機能

データの転送終了時に割込みフラグである SIR ビットがセットされ、割込みを許可する SMCS の SIE ビットが "1" のとき、CPU へ割込み要求を出力します。

図 20.3-9 に、割込み信号の出力タイミングを示します。

図 20.3-9 割込み信号の出力タイミング



第21章

UART

この章では、UART の機能と動作について説明します。

- 21.1 UART の概要
- 21.2 UART のブロックダイアグラム
- 21.3 UART の端子
- 21.4 UART のレジスタ
- 21.5 UART の割込み
- 21.6 UART のボーレート
- 21.7 UART の動作説明
- 21.8 UART 使用上の注意
- 21.9 UART のプログラム例

21.1 UART の概要

UART は、外部装置と同期通信もしくは非同期通信 (調歩同期) をするための、汎用のシリアルデータ通信インタフェースです。通常の双方向通信機能 (ノーマルモード) だけではなく、マスタ / スレーブ型通信機能 (マルチプロセッサモード : マスタ側だけサポート) があります。

■ UART の機能

● UART の機能

UART は、他の CPU や周辺装置とシリアルデータの送受信をする汎用シリアルデータ通信インタフェースで、表 21.1-1 に示す機能をもっています。

表 21.1-1 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	・クロック同期 (スタート / ストップビットなし) ・クロック非同期 (調歩同期)
ボーレート	・専用ボーレートジェネレータあり ・外部クロック入力可能
データ長	・8 ビット、または 7 ビット (非同期 ノーマルモード時のみ) ・1 ~ 8 ビット (同期モードのみ)
信号方式	NRZ (Non Return to Zero) 方式
受信エラー検出	・フレーミングエラー ・オーバランエラー ・パリティエラー (動作モード 1 では不可)
割込み要求	・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 (EI ² OS)、 μ DMAC 転送の対応あり
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，データのみ転送されます。

表 21.1-2 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 ビット + 1 ^{*1}		非同期	
2	ノーマルモード	1 ~ 8 ビット		同期	なし

: 設定不可

*1: " + 1" は通信制御用に使用されるアドレス / データ選択ビット (A/D) です。

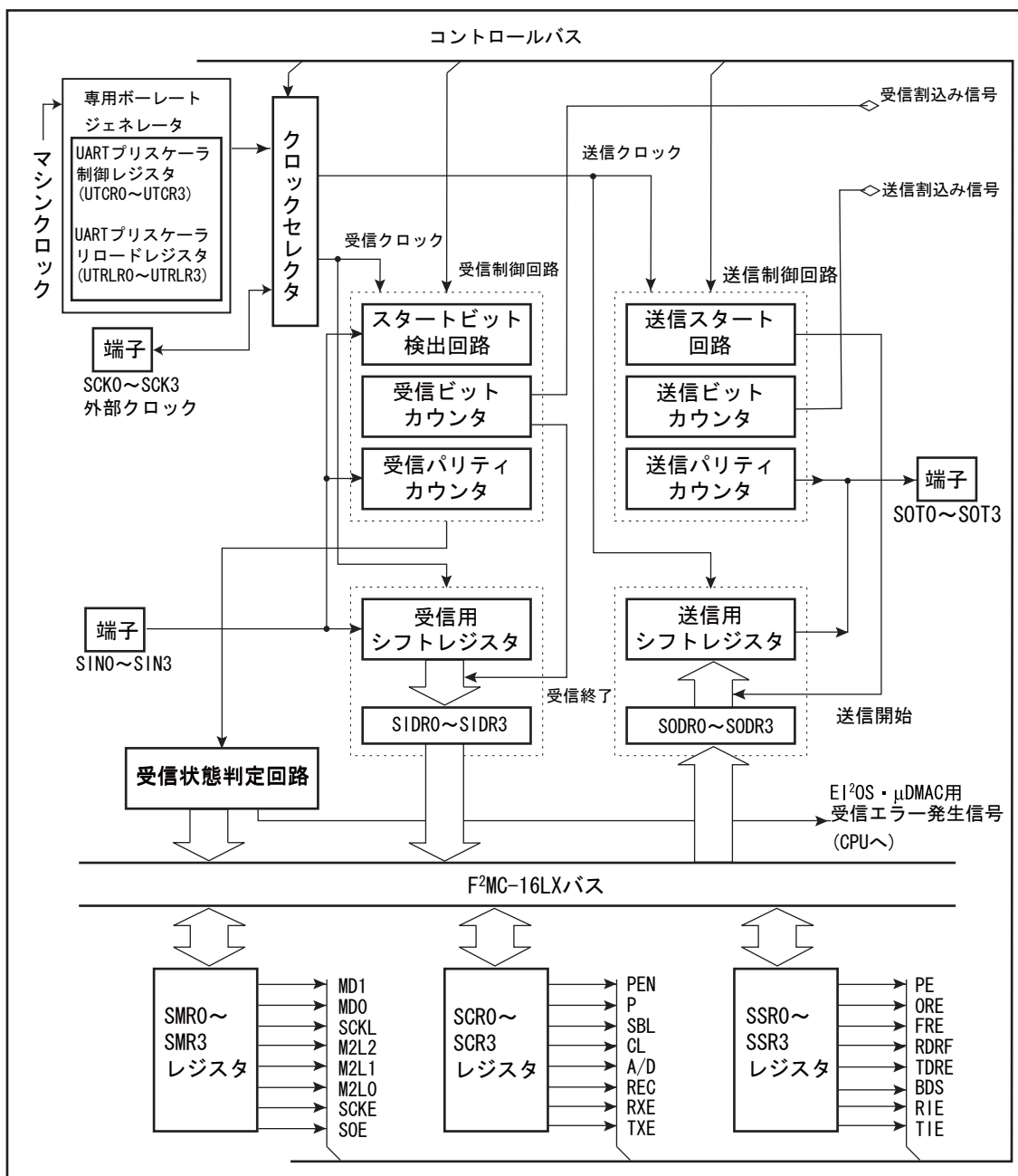
*2: 受信時のストップビットは 1 ビットのみ検出

21.2 UART のブロックダイアグラム

UART は、次のブロックで構成されています。

■ UART のブロックダイアグラム

図 21.2-1 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, 外部入力クロックから送受信クロックを選択します。

● 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, および受信パリティカウンタで構成されています。受信ビットカウンタは受信データのカウンタをして, 設定したデータ長に応じて, 1 データの受信を完了すれば, 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する回路で, スタートビットを検出すると設定された転送速度に応じてシフトしながらシリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3) にデータを書き込みます。受信パリティカウンタは, 受信データのパリティを計算します。

● 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, および送信パリティカウンタで構成されています。送信ビットカウンタは送信データのカウンタをして, 設定したデータ長に応じて, 1 データの送信を完了すれば, 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 0 ~ 3(SODR0 ~ SODR3) の書き込みで送信動作を開始します。送信パリティカウンタは, パリティありの場合送信するデータのパリティビットを生成します。

● 受信用シフトレジスタ

SIN0 ~ SIN3 端子から入力された受信データを, 1 ビットずつビットシフトしながら取り込み, 受信終了すればシリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3) に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 0 ~ 3(SODR0 ~ SODR3) に書き込まれたデータを送信用シフトレジスタに転送し, 1 ビットずつシフトしながら SOT 端子に出力します。

● シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3)

動作モードの選択, シリアルデータの端子への出力許可 / 禁止, クロックの端子への出力許可 / 禁止の設定および, 同期通信モード時に, 1 ~ 8 ビットの任意の転送数の設定, 非動作時のシリアルクロック出力レベル ("L" 固定, "H" 固定) の設定をします。

● シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3)

パリティの有無の設定, パリティの選択, ストップビット長の設定, データ長の設定, モード 1 でのフレームデータ形式の選択, フラグのクリア, 送信の許可 / 禁止, 受信の許可 / 禁止の設定をします。

● シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3)

送受信やエラーの状態の確認と送受信割込み要求の許可 / 禁止の設定をします。

● シリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3)

受信データを保持するレジスタです。シリアル入力に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 0 ~ 3(SODR0 ~ SODR3)

送信データを設定するレジスタです。このレジスタに書き込まれたデータが、シリアル変換されて出力されます。

- UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3)

通信プリスケアラの起動 / 停止 , UART 強制リセット , クロックソースの選択 , マシンクロックの分周比を設定します。

- UART プリスケアラリロードレジスタ 0 ~ 3(UTRLR0 ~ UTRLR3)

マシンクロックの分周比を設定します。

21.3 UART の端子

UART の端子を示します。

■ UART の端子

UART の端子は、汎用ポートと兼用になっています。表 21.3-1 に端子の機能、入出力形式、および UART 使用時の設定などを示します。

表 21.3-1 UART の端子

端子名	端子機能	入出力形式	プルアップ 選択	スタンバイ 制御	端子の使用に必要な設定
P42/A10/SIN0 P45/A13/SIN1 P90/SIN2 P93/SIN3	ポート 4, 9 入出力 / シリアル データ入力	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 : bit2=0) (DDR4 : bit5=0) (DDR9 : bit0=0) (DDR9 : bit3=0)
P43/A11/SOT0 P46/A14/SOT1 P91/SOT2 P94/SOT3	ポート 4, 9 入出力 / シリアル データ出力				出力許可に設定 (SMR0 ~ SMR3 : SOE=1)
P44/A12/SCK0 P47/A15/SCK1 P92/SCK2 P95/SCK3	ポート 4, 9 入出力 / シ リ ア ル ク ロック入出力				クロック入力時 入力ポートに設定 (DDR4 : bit4=0) (DDR4 : bit7=0) (DDR9 : bit2=0) (DDR9 : bit5=0)
		クロック出力時出力許可 に設定 (SMR0 ~ SMR3 : SCKE=1)			

21.4 UART のレジスタ

UART のレジスタ一覧を示します。

■ UART のレジスタ一覧

図 21.4-1 UART のレジスタ一覧

	bit15-----bit8	bit7-----bit0
ch.0 : 000021 _H , 20 _H ch.1 : 000027 _H , 26 _H ch.2 : 00002D _H , 2C _H ch.3 : 000033 _H , 32 _H	SCR0~SCR3 (シリアル制御レジスタ 0~3)	SMR0~SMR3 (シリアルモードレジスタ 0~3)
ch.0 : 000023 _H , 22 _H ch.1 : 000029 _H , 28 _H ch.2 : 00002F _H , 2E _H ch.3 : 000035 _H , 34 _H	SSR0~SSR3 (シリアルステータスレジスタ 0~3)	SIDR0~SIDR3・SODR0~SODR3 (シリアル入力・出力データ レジスタ 0~3)
ch.0 : 000025 _H , 24 _H ch.1 : 00002B _H , 2A _H ch.2 : 000031 _H , 30 _H ch.3 : 000037 _H , 36 _H	UTCRO~UTC3 (UART プリスケアラ制御 レジスタ 0~3)	UTRLR0~UTRLR3 (UART プリスケアラリロード レジスタ 0~3)

< 注意事項 >

通信モードの設定は、動作停止中に行ってください。モード設定時に送受信したデータは保証されません。

21.4.1 シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3)

シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3) は、パリティの設定、ストップビット長やデータ長の選択、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、送受信動作の許可または禁止の設定を行います。

■ シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3)

図 21.4-2 シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3) のビット構成

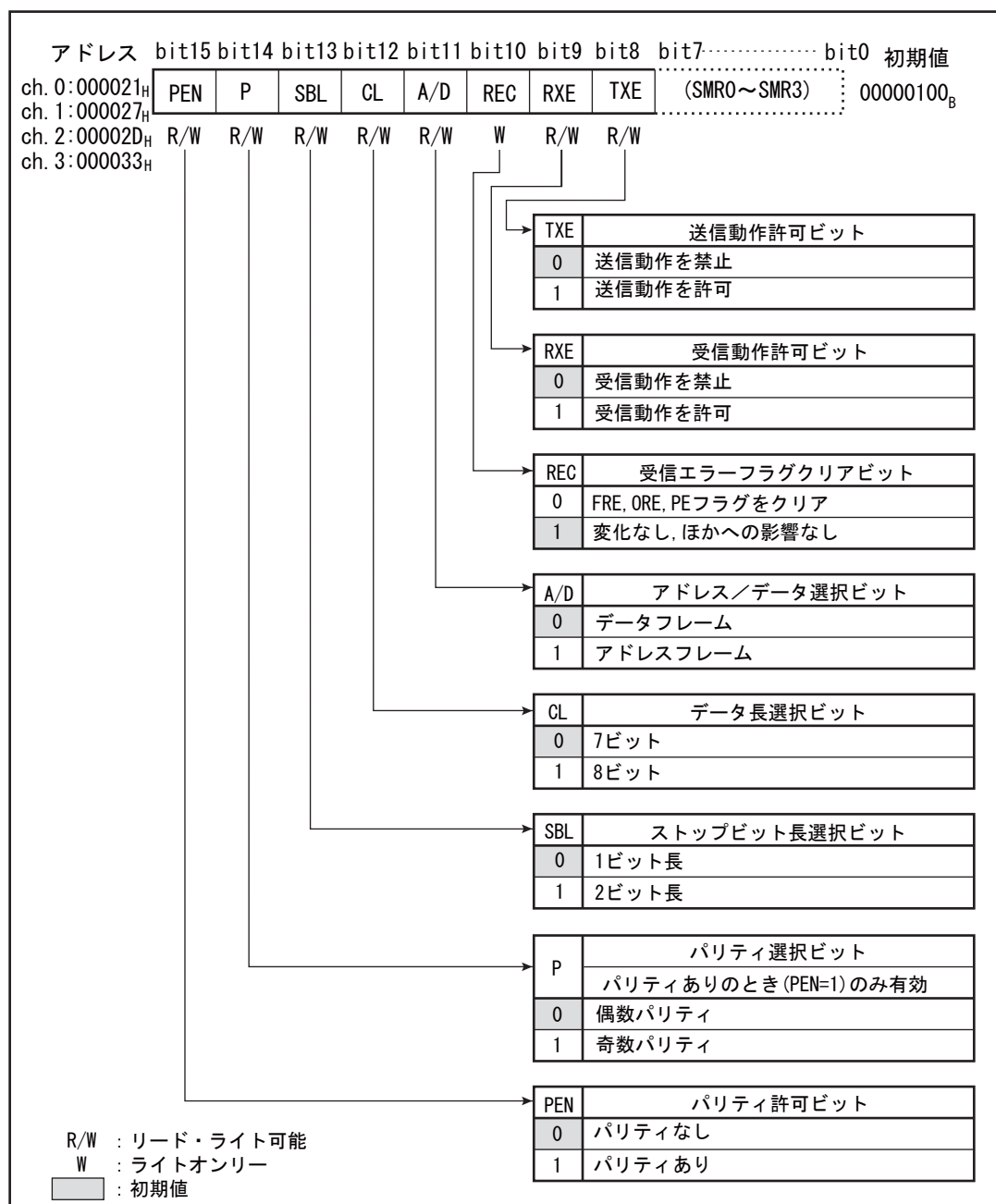


表 21.4-1 シリアル制御レジスタ 0 ~ 3(SCR0 ~ 3) の各ビットの機能説明

ビット名		機 能
bit15	PEN : パリティ 許可ビット	パリティビットの付加 (送信時) と検出 (受信時) を行うかどうかを設定します。 (注意事項) 動作モード 1, 2 の場合, 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 (SCR0 ~ SCR3 : 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に設定します。
bit13	SBL : ストップ ビット長 選択ビット	動作モード 0, 1 (非同期) の場合のストップビット (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 ビット目だけを検出します。
bit12	CL : データ長 選択ビット	送受信データのデータ長を指定します。 (注意事項) 7 ビットを選択できるのは, 動作モード 0 の場合のみです。動作モード 1, 2 の場合は, 必ず 8 ビットに設定してください。
bit11	A/D : アドレス / データ 選択ビット	・動作モード 1 の場合に, 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されます。 ・"1" に設定した場合: アドレスデータのフレームに設定されます。
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 ~ SSR3: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。 ・リードした場合: 常に "1" が読み出されます。 (注意事項) 受信割込みを許可に設定していれば (SSR0 ~ SSR3 : RIE=1), FRE, ORE, PE フラグのいずれかが "1" にセットされている場合のみ, 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には, 受信中のデータをシリアル入力データレジスタに格納した後に受信動作が停止します。
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のデータの送信が完了した後に送信動作が停止します。

21.4.2 シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3)

シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3) は、動作モードの選択、シリアルデータとクロックの端子への出力許可または禁止の設定、および同期通信モード時に、1 ~ 8 ビットの任意の転送数の設定、非動作時のシリアルクロック出力レベル("L" 固定, "H" 固定)の設定を行います。

■ シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3)

図 21.4-3 シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3) のビット構成

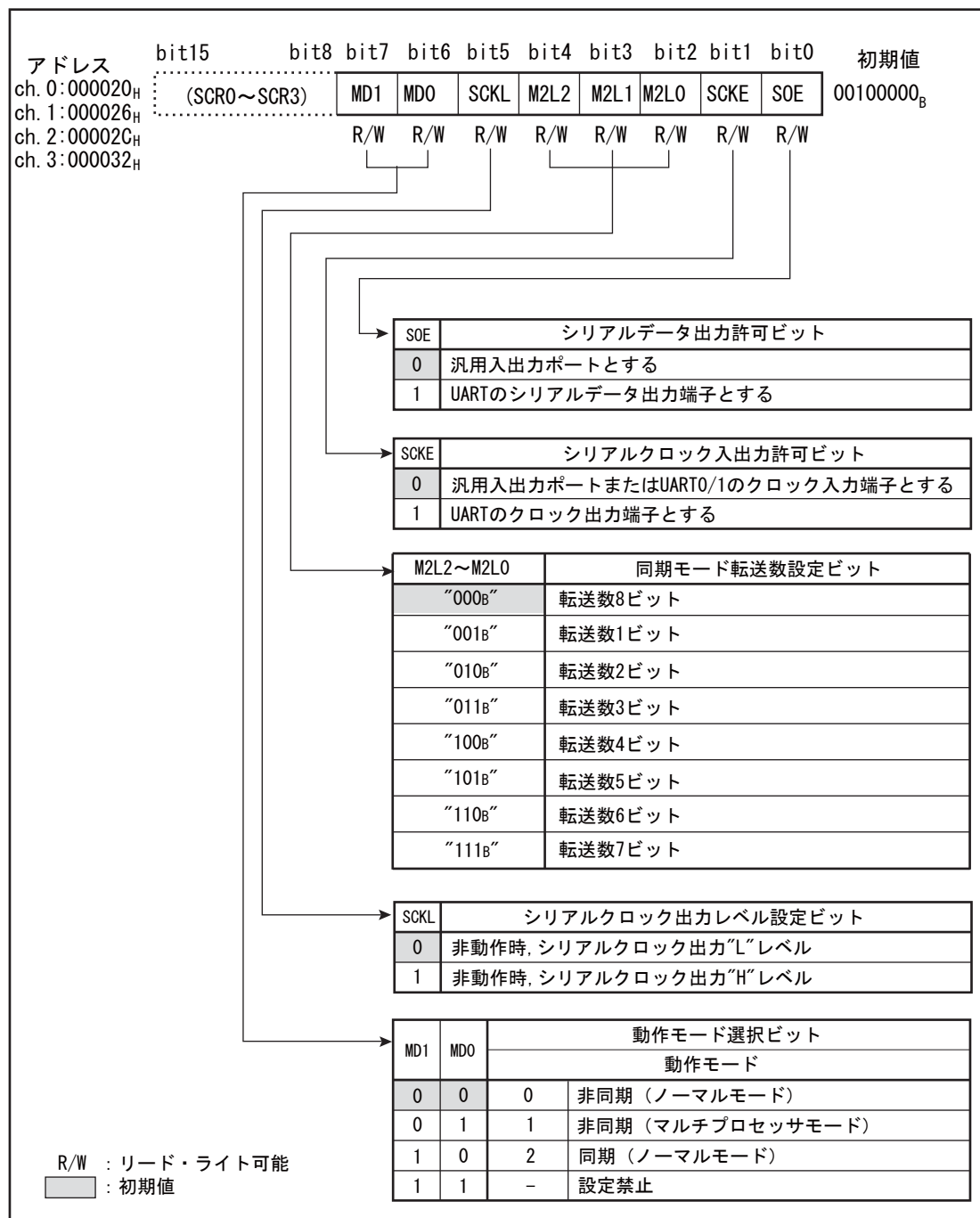


表 21.4-2 シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3) の各ビットの機能説明

ビット名		機 能
bit7, bit6	MD1, MD0 : 動作モード 選択ビット	<ul style="list-style-type: none"> 動作モードを設定します。 (注意事項) 動作モード 1 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。動作モード 1 は 9 ビット目のアドレス / データビットが受信できないのでスレーブとしては使用できません。 動作モード 1 ではパリティチェック機能は使用できないので、パリティ付加許可ビットはパリティなし (SCR0 ~ SCR3 : PEN=0) に設定してください。
bit5	SCKL : シリアルクロック出 力レベル設定ビット	<ul style="list-style-type: none"> クロック同期モードで通信を行う場合、非動作時のシリアルクロック出力レベルを設定します。 "0" に設定した場合：シリアルクロック出力は "L" レベルです。 "1" に設定した場合：シリアルクロック出力は "H" レベルです。
bit4 ~ bit2	M2L2 ~ M2L0 : 同期モード転送数設 定ビット	<ul style="list-style-type: none"> 同期通信モード時の転送数を設定します。 非同期通信モード時は無効です。 "000_B" に設定した場合：転送数は 8 ビットです。 "001_B" に設定した場合：転送数は 1 ビットです。 "010_B" に設定した場合：転送数は 2 ビットです。 "011_B" に設定した場合：転送数は 3 ビットです。 "100_B" に設定した場合：転送数は 4 ビットです。 "101_B" に設定した場合：転送数は 5 ビットです。 "110_B" に設定した場合：転送数は 6 ビットです。 "111_B" に設定した場合：転送数は 7 ビットです。
bit1	SCKE : シリアル クロック 入出力許可 ビット	<ul style="list-style-type: none"> シリアルクロックの入出力を切り換えます。 "0" に設定した場合：汎用入出力ポート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) SCK0 ~ SCK3 端子をシリアルクロック入力 (SCKE=0) として使用する場合は、ポート方向レジスタ (DDR) で端子を入力ポートに設定してください。また、クロック入力ソース選択ビットによって外部クロックを選択 (UTCR0 ~ UTCR3 : CKS=1B) してください。 シリアルクロック出力として使用する場合は、クロック入力ソース選択ビットを専用ポーレートジェネレータ (UTCR0 ~ UTCR3 : CKS=0B) に設定してください。 SCK0 ~ SCK3 端子がシリアルクロック出力に設定されている場合は、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	<ul style="list-style-type: none"> シリアルデータの出力を許可または禁止します。 "0" に設定した場合：汎用入出力ポートに設定されます。 "1" に設定した場合：シリアルデータ出力端子に設定されます。 SOT0 ~ SOT3 端子がシリアルデータ出力に設定されている場合は、汎用入出力ポートの設定にかかわらずシリアルデータ出力端子として機能します。

21.4.3 シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3)

シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3) は、送受信やエラーの状態の確認、割込みの許可または禁止の設定を行います。

■ シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3)

図 21.4-4 シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3) のビット構成

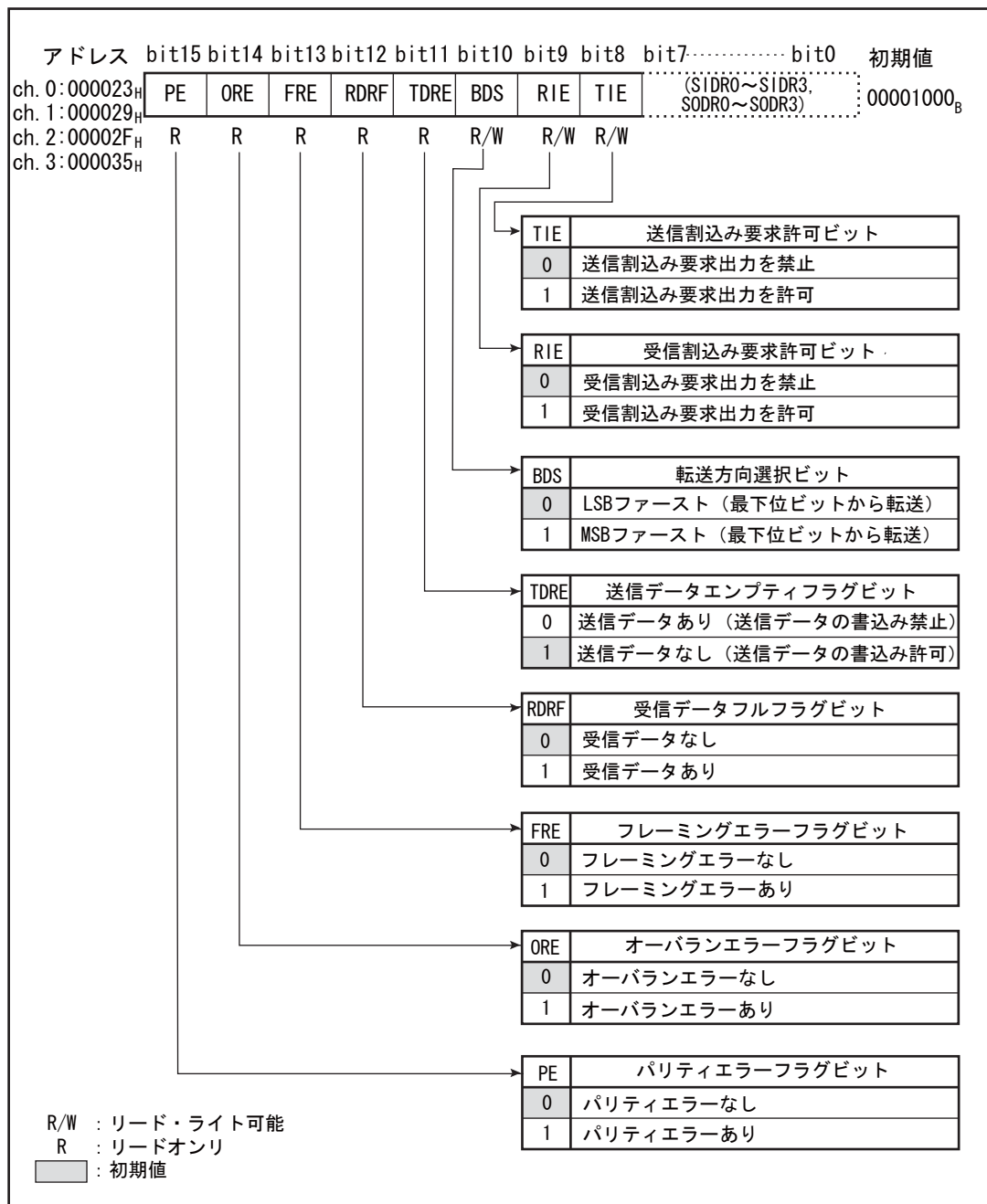


表 21.4-3 シリアルステータスレジスタ 0 ~ 3 (SSR0 ~ SSR3) の各ビットの機能説明 (1 / 2)

ビット名		機 能
bit15	PE : パリティ エラーフラグ ビット	<ul style="list-style-type: none"> ・受信データのパリティエラーを検出します。 ・パリティエラーが発生すると "1" がセットされます。 ・受信エラーフラグクリアビット (SCR0 ~ SCR3 : REC) に "0" を書き込むとクリアされます。 ・受信割込みが許可されている場合は (SCR0 ~ SCR3 : RIE=1), パリティエラーが発生すると受信割込み要求が発生します。 ・パリティエラーフラグビットがセットされた場合は (SSR0 ~ SSR3 : PE=1), シリアル入力データレジスタ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	<ul style="list-style-type: none"> ・受信時のオーバランエラーを検出します。 ・オーバランエラーが発生すると "1" がセットされます。 ・受信エラーフラグクリアビット (SSR0 ~ SSR3 : REC) に "0" を書き込むとクリアされます。 ・受信割込みが許可されている場合は (SSR0 ~ SSR3 : RIE=1), オーバランエラーが発生すると受信割込み要求が発生します。 ・オーバランエラーフラグビットがセットされた場合は (SSR0 ~ SSR3 : ORE=1), シリアル入力データレジスタ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	<ul style="list-style-type: none"> ・受信データのフレーミングエラーを検出します。 ・フレーミングエラーが発生すると "1" がセットされます。 ・受信エラーフラグクリアビット (SCR0 ~ SCR3 : REC) に "0" を書き込むとクリアされます。 ・受信割込みが許可されている場合は (SSR0 ~ SSR3 : RIE=1), フレーミングエラーが発生すると受信割込み要求が発生します。 ・フレーミングエラーフラグビットがセットされた場合は (SSR0 ~ SSR3 : FRE=1), シリアル入力データレジスタのデータは無効です。
bit12	RDRF : 受信データ フルフラグ ビット	<ul style="list-style-type: none"> ・シリアル入力データレジスタの状態を示します。 ・シリアル入力データレジスタ 0 ~ 3 (SIDR0 ~ SIDR3) に受信データがロードされると, "1" がセットされます。 ・シリアル入力データレジスタ 0 ~ 3 (SIDR0 ~ SIDR3) のデータを読み出すと "0" にクリアされます。 ・受信割込みが許可されている場合は (SSR0 ~ SSR3 : RIE=1), 受信データがシリアル入力データレジスタ (SIDR0 ~ SIDR3) にロードされると受信割込み要求が発生します。
bit11	TDRE : 送信データ エンプティ フラグビット	<ul style="list-style-type: none"> ・シリアル出力データレジスタ 0 ~ 3 (SODR0 ~ SODR3) の状態を示します。 ・シリアル出力データレジスタ 0 ~ 3 (SODR0 ~ SODR3) に送信データを書き込むと, "0" にクリアされます。 ・送信用シフトレジスタにデータがロードされ送信が開始されると "1" がセットされます。 ・送信割込みが許可されている場合は (SSR0 ~ SSR3 : TIE=1), シリアル出力データレジスタ 0 ~ 3 (SODR0 ~ SODR3) に書き込まれたデータが送信シフトレジスタに転送されると送信割込み要求が発生します。

表 21.4-3 シリアルステータスレジスタ 0 ~ 3 (SSR0 ~ SSR3) の各ビットの機能説明 (2 / 2)

ビット名		機 能
bit10	BDS : 転送方向選択 ビット	<ul style="list-style-type: none"> ・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 : シリアルデータを最下位ビット側から先に転送します (LSB ファースト)。 ・"1" に設定した場合 : シリアルデータを最上位ビット側から先に転送します (MSB ファースト)。 <p>(注意事項) シリアル入力データレジスタへの読み出し時およびシリアル出力データレジスタへの書き込み時にデータの上位側と下位側を入れ替えるため、レジスタへのアクセス完了後に BDS ビットを書き換えた場合はそのデータは無効になります。</p>
bit9	RIE : 受信割込み 要求許可 ビット	<ul style="list-style-type: none"> ・受信割込みを許可または禁止します。 ・"1" に設定した場合 : シリアル入力データレジスタ 0 ~ 3 に受信データがロードされるか (SSR0 ~ SSR3: RDRF=1), 受信エラーが発生すると (SSR0 ~ SSR3: PE=1, または ORE=1, または FRE=1), 受信割込み要求を発生します。
bit8	TIE : 送信割込み 要求許可 ビット	<ul style="list-style-type: none"> ・送信割込みを許可または禁止します。 ・"1" に設定した場合 : シリアル出力データレジスタ 0 ~ 3 に書き込まれたデータが送信用シフトレジスタに送信されると (SSR0 ~ SSR3: TDRE=1), 送信割込み要求を発生します。

21.4.4 シリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3), シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3)

シリアル入力データとシリアル出力データレジスタは同一アドレスに配置されています。リードした場合は、シリアル入力データレジスタとして機能し、ライトした場合はシリアル出力データレジスタとして機能します。

■ シリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3)

図 21.4-5 に、シリアル入力データレジスタのビット構成を示します。

図 21.4-5 シリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3) のビット構成

アドレス	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値
ch.0 : 000022 _H	D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX _B
ch.1 : 000028 _H	R	R	R	R	R	R	R	R	
ch.2 : 00002E _H									
ch.3 : 000034 _H									

R : リードオンリ
X : 不定

シリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3) 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。

- ・シリアル入力端子(SIN0 ~ SIN3)に送られてきたシリアルデータ信号がシフトレジスタで変換されて、シリアル入力データレジスタ0~3(SIDR0~SIDR3)に格納されます。
- ・データ長が7ビットの場合は、上位1ビット(SIDR0 ~ SIDR3:D7)は無効データとなります。
- ・受信データが、シリアル入力データレジスタ0 ~ 3(SIDR0 ~ SIDR3)に格納されると、受信データフルフラグビット(SSR0 ~ SSR3:RDRF)が"1"にセットされます。受信割込みが許可されている場合は(SSR0 ~ SSR3:RIE=1)、受信割込み要求が発生します。
- ・シリアル入力データレジスタ0 ~ 3(SIDR0 ~ SIDR3)は、受信データフルフラグビット(SSR0 ~ SSR3:RDRF)が"1"の状態を読み出してください。受信データフルフラグビット(SSR0 ~ SSR3:RDRF)は、シリアル入力データレジスタ0 ~ 3(SIDR0 ~ SIDR3)を読み出すと自動的に"0"にクリアされます。
- ・受信エラーが発生(SSR0 ~ SSR3:PE, ORE, FREのいずれかが"1")した場合、シリアル入力データレジスタ0 ~ 3(SIDR0 ~ SIDR3)のデータは無効となります。

■ シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3)

図 21.4-6 に、シリアル出力データレジスタのビット構成を示します。

図 21.4-6 シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) のビット構成

アドレス	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値
ch.0 : 000022 _H	D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX _B
ch.1 : 000028 _H	W	W	W	W	W	W	W	W	
ch.2 : 00002E _H									
ch.3 : 000034 _H									

W : ライトオンリ
X : 不定

シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) 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。

- ・送信動作が許可されている場合に (SCR0 ~ SCR3:TXE=1), 送信するデータをシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に書き込むと、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に変換されて、シリアルデータ出力端子 (SOT0 ~ SOT3) から送出されます。
- ・データ長が 7 ビットの場合、上位 1 ビット (SODR0 ~ SODR3:D7) は無効データとなります。
- ・送信データエンプティフラグ (SSR0 ~ SSR3:TDRE) は、送信データがシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に書き込まれると、"0" にクリアされます。
- ・送信データエンプティフラグ (SSR0 ~ SSR3:TDRE) は、送信用シフトレジスタへの転送が終了すると、"1" にセットされます。
- ・送信データエンプティフラグ (SSR0 ~ SSR3:TDRE) が "1" の場合は、次の送信用データを書き込むことができます。送信割込みが許可されている場合には送信割込みが発生します。次の送信データの書き込みは、送信データエンプティフラグ (SSR0 ~ SSR3:TDRE) 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き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出し専用のレジスタです。2 つのレジスタは同一アドレスに配置されているため書き込み値と読み出し値が異なります。したがって、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。

21.4.5 UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3), UART プリスケアラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3)

UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3) は、プリスケアラの起動 / 停止、強制リセット、クロックソース選択を設定します。また、UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3) の下位 3 ビットと UART プリスケアラリロードレジスタ (0 ~ 3) との組合せでマシンのクロックの分周比を設定します。

■ UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3), UART プリスケアラリロードレジスタ 0 ~ 3(UTRLR0 ~ UTRLR3)

UART の動作クロックは、マシンのクロックを分周することにより得られます。このプリスケアラによって、種々のマシンサイクルに対して一定のボーレートが得られように設計されています。UTCR0 ~ UTCR3 のビット構成を図 21.4-7 に示します。

図 21.4-7 UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3), UART プリスケアラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3) のビット構成

UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3)									
アドレス	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	初期値
ch.0 : 000025 _H	MD	SRST	CKS	予約	-	D10	D9	D8	0000-000 _B
ch.1 : 00002B _H	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	
ch.2 : 000031 _H									
ch.3 : 000037 _H									
UART プリスケアラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3)									
アドレス	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	初期値
ch.0 : 000024 _H	D7	D6	D5	D4	D3	D2	D1	D0	00000000 _B
ch.1 : 00002A _H	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
ch.2 : 000030 _H									
ch.3 : 000036 _H									
R/W : リード・ライト可能									
- : 未定義									

[bit15] MD (プリスケアラ許可ビット)

プリスケアラの動作許可ビットです。

0: プリスケアラは停止します。

1: プリスケアラは動作します。

[bit14] SRST (UART 強制リセットビット)

UART 用内部リセットビットです。強制的に UART をリセットし、初期状態にします。初期化後は自動的に "0" になります。

0: 状態保持

1: 強制リセット

[bit13] CKS (クロックソース選択ビット)

クロックソースを選択します。

0: 専用ポーレートジェネレータ

1: 外部クロック

[bit12] 予約 (予約ビット)

予約ビットです。

必ず "0" を設定してください。

[bit11] 未定義ビット

リード時の値は不定です。ライトは動作に影響を与えません。

[bit10 ~ bit0] D10 ~ D0 (基本周期設定ビット)**<非同期モード>**

シリアルクロックの 2 サイクル分の周期を決定します。

UART 側でシリアルクロックを 8 分周しますので、ポーレートは以下のようになります。

ポーレート = $\phi/4n$ (bps)

ϕ : マシンクロック n: D10 ~ D0 (基本周期設定ビット)

ただし, n=1, 2, 3 は設定しないでください。

<クロック同期モード>

ポーレート = $2\phi/n$ (bps)

ϕ : マシンクロック n: D10 ~ D0 (基本周期設定ビット)

ただし, n=16 以上,

(D1, D0) n は "00_B" を設定してください。

<注意事項>

- 基本周期設定ビットの D1, D0 (UTRLR0 ~ UTRLR3 のビット 1, 0) に "01_B", "11_B" を設定しますとデューティ比が異なるシリアルクロックを生成しますので、クロック同期モード時には、D1, D0 (UTRLR0 ~ UTRLR3 のビット 1, 0) を "00_B", "11_B" に設定してください。
 - UTCR0 ~ UTCR3, UTRLR0 ~ UTRLR3 へはワード転送命令でアクセスしてください。
 - クロックソースの切換えはプリスケアラの動作を停止 (MD=0, CKS=0) した後、切り換えてください。
 - 外部クロック選択モード時にリロード値を変更する場合は、プリスケアラの動作を停止 (MD=0, CKS=0) した後、リロード値を変更してください。
 - リロード値 n=1, 2, 3 は設定しないでください。
 - 外部クロックによる、非同期モードの設定はしないでください。
-

21.5 UART の割り込み

UART には、受信割り込みと送信割り込みがあり、次に示す要因で割り込み要求を発生させることができます。

- 受信データがシリアル入力データレジスタ 0 ~ 3 (SIDR0 ~ SIDR3) にセットされた場合、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 0 ~ 3 (SODR0 ~ SODR3) から送信用シフトレジスタに転送された場合

また、それぞれ拡張インテリジェント I/O サービス (EI²OS), μ DMAC にも対応しています。

■ UART の割り込み

UART の割り込み制御ビットと割り込み要因は表 21.5-1 のようになっています。

表 21.5-1 UART の割り込み制御ビットと割り込み要因

送受信	シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3)					
	割り込み フラグ ビット	動作モード			割り込み要因	割り込み 許可 ビット
		0	1	2		
受信	RDRF				受信データをバッファ (SIDR0 ~ SIDR3) へロード	RIE
	ORE				オーバランエラー発生	
	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				送信バッファ (SODR0 ~ SODR3) が空	TIE

：使用ビット

×：未使用ビット

● 受信割り込み

受信割り込みが許可されている場合 (SSR0 ~ SSR3:RIE=1) に、データ受信完了 (SSR0 ~ SSR3:RDRF=1)、オーバランエラー (SSR0 ~ SSR3:ORE=1)、フレーミングエラー (SSR0 ~ SSR3:FRE=1)、パリティエラー (SSR0 ~ SSR3:PE=1) のいずれかが発生すると受信割り込み要求を発生します。

受信データフルフラグ (SSR0 ~ SSR3:RDRF) は、シリアル入力データレジスタ 0 ~ 3 (SIDR0 ~ SIDR3) 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 ~ SSR3:PE, ORE, FRE) は受信エラーフラグクリアビット (SCR0 ~ SCR3:REC) に "0" を書き込めば、すべて "0" にクリアされます。

受信エラー (パリティエラー、オーバランエラー、フレーミングエラー) が発生した場合は、必要に応じてエラー処理を行った後、受信エラーフラグクリアビット (SCR0 ~ SCR3:REC) に "0" を書き込んで、各受信エラーフラグをクリアしてください。

● 送信割り込み

送信データがシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) から送信用シフトレジスタに送信されると、送信データエンプティフラグビット (SSR0 ~ SSR3:TDRE) に "1" がセットされます。送信割り込みが許可されている場合は (SSR0 ~ SSR3:TIE=1), 送信割り込み要求が発生します。

■ UART の割り込みと EI²OS および μ DMAC表 21.5-2 UART の割り込みと EI²OS および μ DMAC

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	μ DMAC チャンネル番号
		レジスタ名	アドレス	下位	上位	バンク		
UART3 受信割り込み	#35(23 _H)	ICR12	0000BC _H	FFFF70 _H	FFFF71 _H	FFFF72 _H		10
UART3 送信割り込み	#33(21 _H)	ICR11	0000BB _H	FFFF78 _H	FFFF79 _H	FFFF7A _H		11
UART2 受信割り込み	#35(23 _H)	ICR12	0000BC _H	FFFF70 _H	FFFF71 _H	FFFF72 _H		10
UART2 送信割り込み	#33(21 _H)	ICR11	0000BB _H	FFFF78 _H	FFFF79 _H	FFFF7A _H		11
UART1 受信割り込み	#39(27 _H)	ICR14	0000BE _H	FFFF60 _H	FFFF61 _H	FFFF62 _H		12
UART1 送信割り込み	#37(25 _H)	ICR13	0000BD _H	FFFF68 _H	FFFF69 _H	FFFF6A _H		13
UART0 受信割り込み	#39(27 _H)	ICR14	0000BE _H	FFFF60 _H	FFFF61 _H	FFFF62 _H		12
UART0 送信割り込み	#37(25 _H)	ICR13	0000BD _H	FFFF68 _H	FFFF69 _H	FFFF6A _H		13

: 使用可能。UART の受信エラー検出による EI²OS 停止機能付

: 使用可能

■ UART の EI²OS 機能

UART は EI²OS 対応の回路を有しています。このため受信 / 送信の各割り込みで別々に EI²OS を起動できます。

● 送受信時

送受信時は、他の周辺機能の状態に関係なく、EI²OS を使用できます。

21.5.1 受信割込み発生とフラグセットのタイミング

受信時の割込みとしては、受信完了 (SSR0 ~ SSR3:RDRF), および受信エラーの発生 (SSR0 ~ SSR3:PE, ORE, FRE) があります。

■ 受信割込み発生とフラグセットのタイミング

データを受信する場合は、ストップビットの検出 (動作モード 0, 1) または、データの最終ビット (SIDR0 ~ SIDR3:D7) の検出 (動作モード 2) によって、受信データがシリアル入力データレジスタ 0 ~ 3 (SIDR0 ~ SIDR3) に格納されます。受信エラーが発生した場合はエラーフラグ (SSR0 ~ SSR3:PE, ORE, FRE) がセットされ、受信データフルフラグ (SSR0 ~ SSR3:RDRF) がセットされます。各動作モードともいずれかのエラーフラグがセットされた場合は、受信したシリアル入力データレジスタ 0 ~ 3 (SIDR0 ~ SIDR3) 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)

ストップビット検出時に受信データフルフラグ (SSR0 ~ SSR3:RDRF) にセットされます。受信エラーが発生すると、エラーフラグ (SSR0 ~ SSR3:PE, ORE, FRE) がセットされます。

● 動作モード 1 (非同期マルチプロセッサモード)

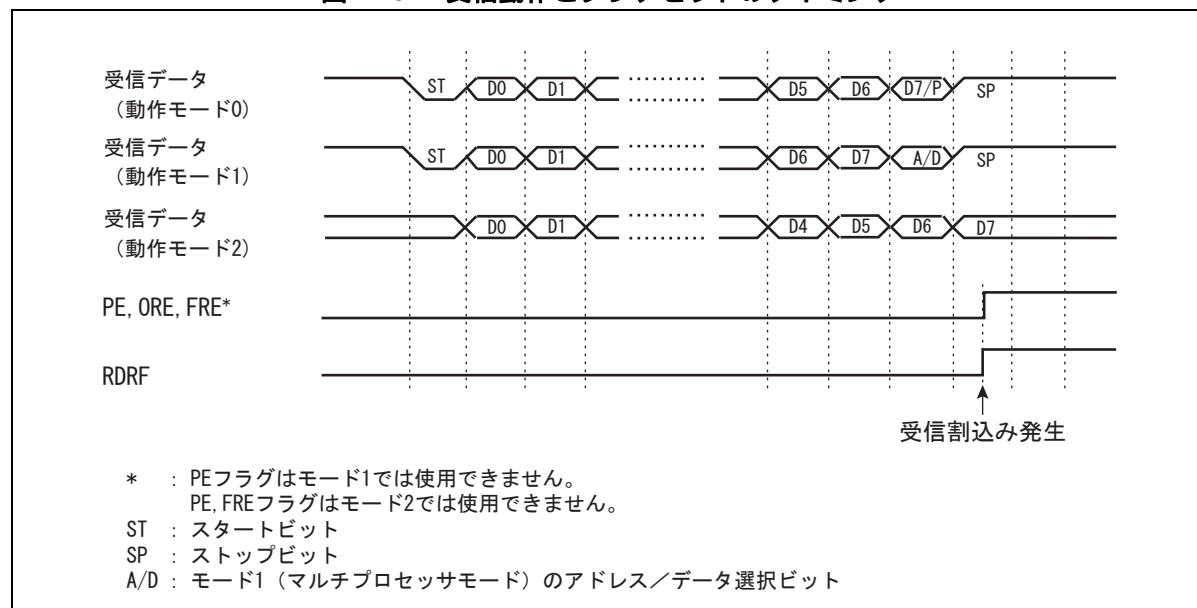
ストップビットを検出すると受信データフルフラグビット (SSR0 ~ SSR3:RDRF) がセットされます。受信エラーが発生すると、エラーフラグ (SSR0 ~ SSR3:ORE, FRE) がセットされます。パリティエラー (SSR0 ~ SSR3:PE) は検出できません。

● 動作モード 2 (クロック同期ノーマルモード)

受信データの最終ビット (SIDR0 ~ SIDR3:D7) を検出すると受信データフルフラグビット (SSR0 ~ SSR3:RDRF) がセットされます。受信エラーが発生すると、エラーフラグ (SSR0 ~ SSR3:ORE) がセットされます。パリティエラー (SSR0 ~ SSR3:PE), およびフレーミングエラー (SSR0 ~ SSR3:FRE) は検出できません。

図 21.5-1 に、受信動作とフラグセットのタイミングを示します。

図 21.5-1 受信動作とフラグセットのタイミング



● 受信割り込み発生のタイミング

受信割り込みが許可されている場合に (SSR0 ~ SSR3:RIE=1), 受信データフルフラグ (SSR0 ~ SSR3:RDRF), パリティエラーフラグ (SSR0 ~ SSR3:PE), オーバランエラーフラグ (SSR0 ~ SSR3:ORE), フレーミングエラーフラグ (SSR0 ~ SSR3:FRE) のいずれかが "1" にセットされると, 受信割り込み要求が発生します。

21.5.2 送信割込み発生とフラグセットのタイミング

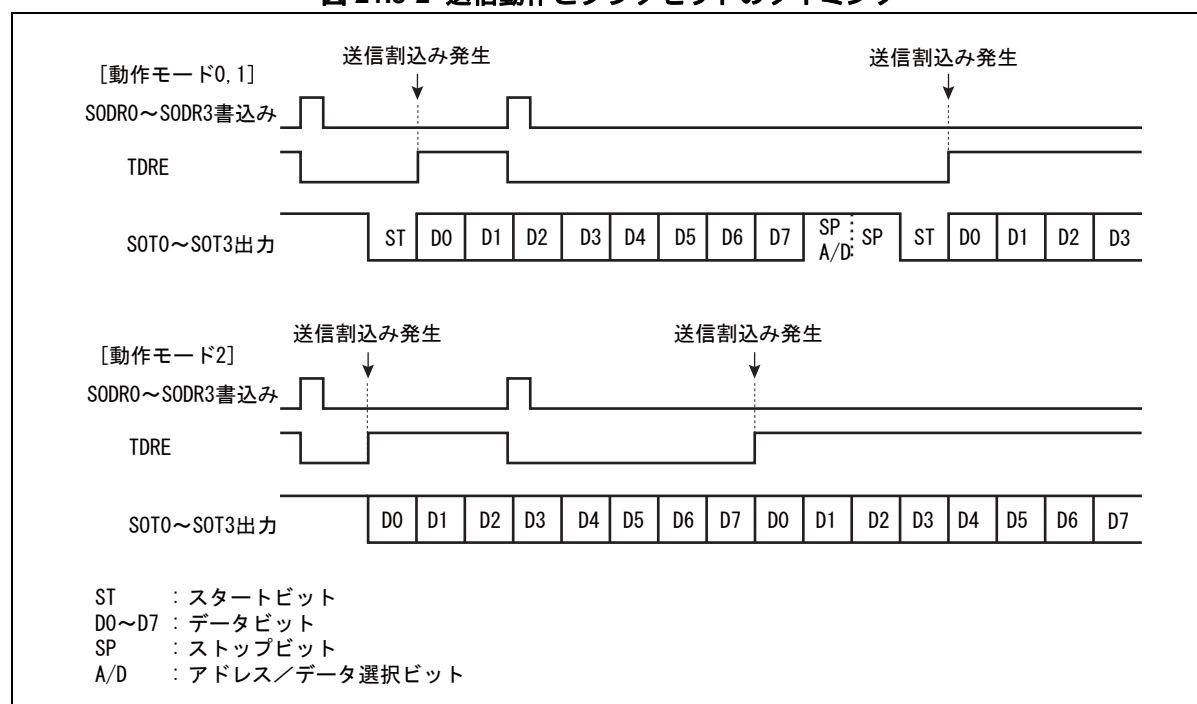
送信時の割込みは、シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) が空になり、次の送信データの書込みが可能な状態になると発生します。

■ 送信割込み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ビット (SSR0 ~ SSR3:TDRE) は、シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に書き込まれた送信データが、送信用シフトレジスタに転送され、次のデータが書込み可能な状態になると "1" にセットされます。シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に次の送信データを書き込むと送信データエンプティフラグビット (SSR0 ~ SSR3:TDRE) は "0" にクリアされます。

図 21.5-2 に送信動作とフラグセットのタイミングについて示します。

図 21.5-2 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タイミング

送信割込みが許可 (SSR0 ~ SSR3:TIE=1) されている場合に、送信データエンプティフラグビット(SSR0 ~ SSR3:TDRE)が"1"にセットされると送信割込み要求が発生します。

< 注意事項 >

送信動作中に送信動作を禁止 (SCR0 ~ SCR3:TXE=0, 動作モード 1 の場合、受信動作を禁止 RXE も含む) すると、送信データエンプティフラグビットがセットされ (SSR0 ~ SSR3:TDRE=1), 送信用シフトレジスタのシフト動作が停止してから UART の通信動作が禁止されます。送信が停止する前にシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に書き込まれていた送信データは送信されます。

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 0 ~ 3(SODR0 ~ SODR3) 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ので、送信割込み許可のタイミングに注意してください。

21.6 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できます。

- 内部クロック (リロードカウンタ)
 - 外部クロック (リロードカウンタ)
 - 外部クロック (SCK 端子入力のクロック)
-

■ UART ボーレート選択

ボーレートの選択回路は次のようになっており、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。

● 内部クロックの専用ボーレートジェネレータによるボーレート

UART プリスケアラ制御レジスタ 0 ~ 3 (UTCR0 ~ UTCR3), および UART プリスケアラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3) の設定で 11 ビットのリロード値を設定することにより、ボーレートを選択できます。リロードカウンタは、設定された値でマシンクロックを分周します。非同期モード、同期モード時に使用します。

● 外部クロックの専用ボーレートジェネレータによるボーレート

リロードカウンタのクロックソースに外部クロックを使用します。UART プリスケアラ制御レジスタ 0 ~ 3 (UTCR0 ~ UTCR3), および UART プリスケアラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3) の設定で 11 ビットのリロード値を設定することにより、ボーレートを選択できます。リロードカウンタは、設定された値で外部クロックを分周します。非同期モード時に使用します。

● 外部クロック (1 対 1 モード) によるボーレート

UART のクロック入力端子 (SCK0 ~ SCK3) から入力されたクロックを、そのままボーレートとして使用します。同期モードに使用します。

21.6.1 UART 内部クロックの専用ボーレートジェネレータによるボーレート

UART の転送クロックとして、専用ボーレートジェネレータを選択した場合の設定可能なボーレートを示します。

■ 内部クロックの専用ボーレートジェネレータによるボーレート

UART プリスケアラ制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3), および UART プリスケアラリロードレジスタ 0 ~ 3(UTRLR0 ~ UTRLR3) で 11 ビットのリロード値を設定することにより、マシニングロックを分周します。

< 非同期モード >

ボーレート = $\phi/4n$ (bps)

ϕ : マシニングロック周波数 n:D10 ~ 0(UTCR0 ~ UTCR3, UTRLR0 ~ UTRLR3) の値
ただし, n=1, 2, 3 は設定しないでください。

< 同期モード >

ボーレート = $2\phi/n$ (bps)

ϕ : マシニングロック周波数 n:D10 ~ 0(UTCR0 ~ UTCR3, UTRLR0 ~ UTRLR3) の値
ただし, n=16 以上 (D1, D0) には "00_B" を設定してください。

< 注意事項 >

基本周期設定ビットの D1, D0(UTRLR0 ~ UTRLR3 のビット 1, 0) に "01_B", "10_B", "11_B" を設定しますとデューティ比が異なるシリアルクロックを生成しますので、クロック同期モード時には、D1, D0(UTRLR0 ~ UTRLR3 のビット 1, 0) を "00_B" に設定してください。

21.6.2 外部クロックの専用ボーレートジェネレータによるボーレート

UART の転送クロックとして、外部クロックの専用ボーレートジェネレータを選択した場合の設定可能なボーレートを示します。非同期モード時に使用します。

■ 外部クロックの専用ボーレートジェネレータによるボーレート

UART プリスケール制御レジスタ 0 ~ 3(UTCR0 ~ UTCR3), および UART プリスケールリロードレジスタ 0 ~ 3(UTRLR0 ~ UTRLR3) で 11 ビットのリロード値を設定することにより、外部クロックを分周します。

ボーレートは外部クロックの周波数を f とすると次のようになります。

・ UTCR0 ~ UTCR3:MD=1 に設定した場合

$$b = f/4n \text{ (bps)}, f = \phi \cdot f/4n$$

f: 外部クロック周波数

n: D10 ~ 0 (UTCR0 ~ UTCR3, UTRLR0 ~ UTRLR3) の値

b: ボーレート

ただし、 $n=1, 2, 3$ は設定しないでください。

f の最大周波数は 24 MHz です。

< 注意事項 >

外部クロックにより生成されたボーレートはマシクロックで同期化されるために、ボーレートの計算値に誤差が発生します。

21.6.3 外部クロック (1 対 1 モード) によるボーレート

UART の転送クロックとして、外部クロックを選択する場合の設定とボーレートの計算式を示します。クロック同期モード時に使用します。

■ 外部クロック (1 対 1 モード) によるボーレート

外部クロック (1 対 1 モード) によるボーレートを選択するには、次の 3 つの設定が必要です。

UART プリスケアラ制御レジスタ 0 ~ 3 (UTCR0 ~ UTCR3) の MD ビットに "0", CKS ビットに "1" を書き込んで、外部クロック (1 対 1 モード) によるボーレートを選択します。

SCK0 ~ SCK3 端子を入力に設定します (DDR4:bit4=0, bit7=0, DDR9:bit2=0, bit5=0)。

シリアルモードレジスタ 0 ~ 3 (SMR0 ~ SMR3) の SCKE ビットに "0" を書き込んで端子を外部クロック入力端子とします。

クロック同期モード時に使用します。

● ボーレート計算式

外部クロック (1 対 1 モード) によるボーレートを選択した場合 (UTCR0 ~ UTCR3:MD=0, CKS=1) のボーレートは、外部クロックの周波数を f とすると次のようになります。

同期 $= f$ (bps)

f : 外部クロック周波数

ただし、 f の最大はマシンのクロックの 1/8 までです。

21.7 UART の動作説明

UART には、通常の双方向シリアル通信機能 (動作モード 0, 動作モード 2) の他に、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 は動作モード 0 ~ 動作モード 2 の 3 種類があり、表 21.7-1 に示すように CPU 間の接続方式やデータ転送方式で選択できます。

表 21.7-1 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 ^{*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" は通信制御用に使用されるアドレス / データ選択ビット (A/D) です。

*2: 受信時のストップビットは 1 ビットのみ検出可能

< 注意事項 >

UART の動作モード 1 は、マスタ・スレーブ型接続時のマスタ時だけに使用されます。

● CPU 間接続方式

1 対 1 接続とマスタ・スレーブ型接続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。動作モードは次のように選択します。

1 対 1 接続の場合: 2 つの CPU で動作モード 0, 動作モード 2 のいずれかの同じ方式を採用する必要があります。非同期方式では動作モード 0 を、同期方式では動作モード 2 を選択してください。

マスタ・スレーブ型接続の場合: 動作モード 1 を設定します。動作モード 1 を選択した場合は、マスタとして使用してください。なお、この接続ではパリティなし、データ長 8 ビットに設定してください。

● 同期方式

非同期方式 (調歩同期) またはクロック同期方式を選択できます。

- 信号方式

NRZ(Non Return to Zero) 形式のデータだけを扱えます。

- 送受信動作の開始

シリアル制御レジスタの送信動作許可ビット (SCR0 ~ SCR3:TXE) を "1" に設定すると、送信動作を開始します。

シリアル制御レジスタの受信動作許可ビット (SCR0 ~ SCR3:RXE) を "1" に設定すると、受信動作を開始します。

- 送受信動作の停止

シリアル制御レジスタの送信動作許可ビット (SCR0 ~ SCR3:TXE) を "0" に設定すると、送信動作を停止します。

シリアル制御レジスタの受信動作許可ビット (SCR0 ~ SCR3:RXE) を "0" に設定すると、受信動作を停止します。

- 送受信中の停止

受信 (受信用シフトレジスタへのデータ入力中) に受信動作を禁止 (SCR0 ~ SCR3:RXE=0) した場合には、受信中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 0 ~ 3 (SIDR0 ~ SIDR3) に受信データを格納してから受信動作を停止します。

送信 (送信用シフトレジスタからデータが出力中) に送信動作を禁止 (SCR0 ~ SCR3:TXE=0) した場合は、シリアル出力データレジスタ 0 ~ 3 (SODR0 ~ SODR3) から送信用シフトレジスタに 1 フレームの転送が完了してから送信動作を停止します。

21.7.1 非同期モード (動作モード 0, 1) 時の動作

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, 転送方式は非同期となります。

■ 非同期モード時の動作

● 送受信データフォーマット

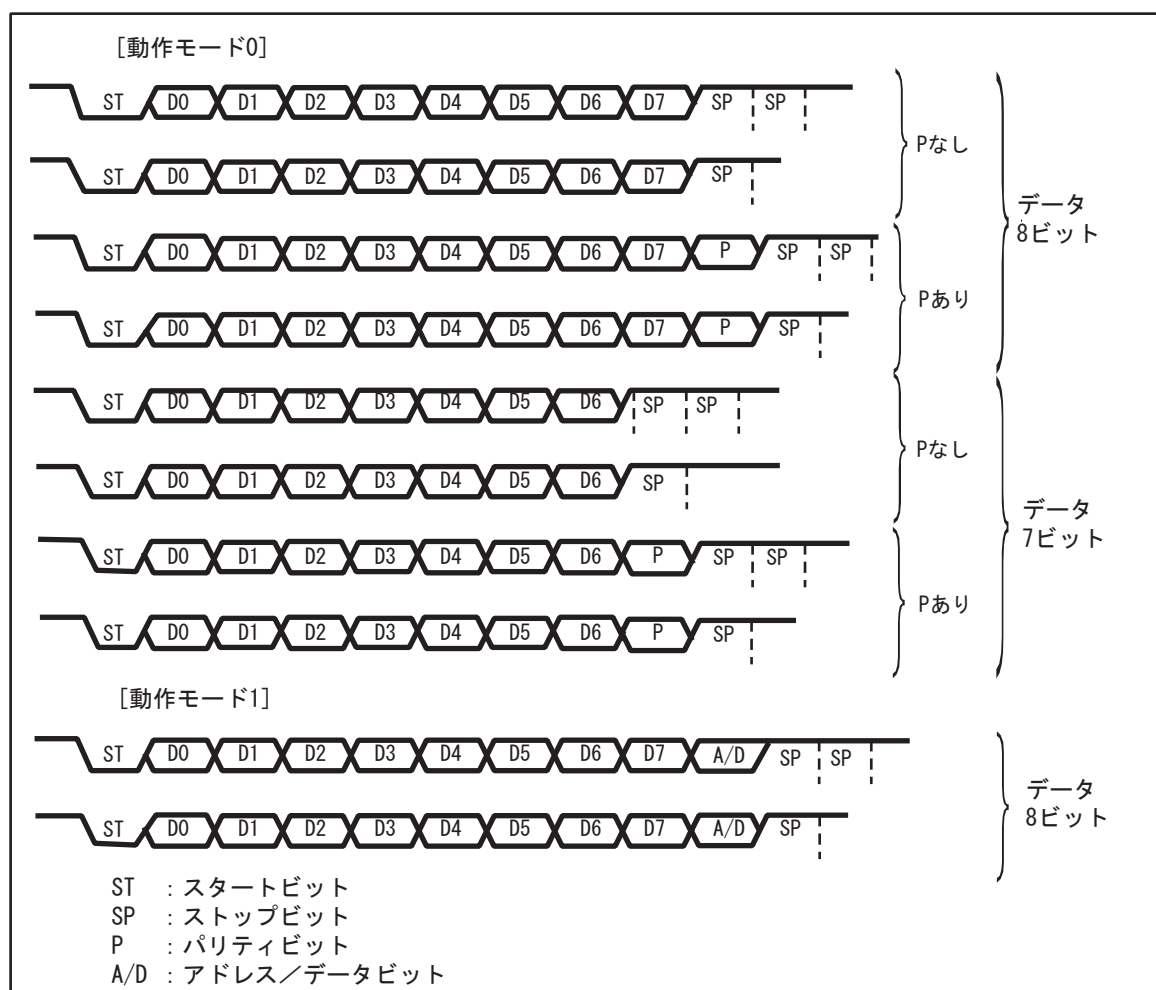
送受信は, 必ずスタートビット ("L" レベル) から始まり, 指定されたデータビット長の送受信が行われ, ストップビット ("H" レベル) で終了します。

動作モード 0 では, データ長は 7 ビットまたは 8 ビットを選択します。パリティビットは, あり / なしの選択ができます。

動作モード 1 では, 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。9 ビット目に, アドレス / データビット (SCR0 ~ SCR3:A/D) 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 21.7-1 に示します。

図 21.7-1 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ビット (SSR0 ~ SSR3:TDRE) に "1" がセットされている状態で、シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に書き込みます。
- ・ 送信データを書込み、シリアル制御レジスタの送信動作許可ビット (SCR0 ~ SCR3:TXE) を "1" に設定すると、送信が開始されます。
- ・ 送信データをシリアル出力データレジスタに書き込むと、送信データエンプティフラグビット (SSR0 ~ SSR3:TDRE) は、いったん "0" にクリアされます。
- ・ 送信データをシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) から送信用シフトレジスタに転送すると、送信データエンプティフラグビット (SSR0 ~ SSR3:TDRE) は再度 "1" にセットされます。
- ・ 送信割込み許可ビット (SSR0 ~ SSR3:TIE) を "1" に設定していた場合は、送信データエンプティフラグビット (SSR0 ~ SSR3:TDRE) が "1" にセットされると、送信割込み要求が発生します。割込み処理において、次の送信データをシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に書き込みます。

● 受信動作

- ・ 受信動作は、受信動作の許可が設定されている場合では (SCR0 ~ SCR3:RXE=1), 常に行われます。
- ・ 受信データのスタートビットを検出すると、シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3) に設定しているデータフォーマットに従って、1 フレームのデータをシリアル入力データレジスタ 0 ~ 3(SIDR0 ~ SIDR3) に受信します。
- ・ 1 フレームのデータ受信が完了すると、受信データフルフラグビット (SSR0 ~ SSR3:RDRF) が "1" にセットされます。
- ・ 受信データを読み出す場合は、1 フレームデータの受信完了後に、シリアルステータスレジスタ 0 ~ 3(SSR0 ~ SSR3) のエラーフラグの状態を確認し、正常に受信が行われていれば、シリアル入力データレジスタから受信データを読み出してください。受信エラーが発生している場合は、エラー処理を行ってください
- ・ 受信データの読出しを行うと、受信データフルフラグビット (SSR0 ~ SSR3:RDRF) は "0" にクリアされます。

● ストップビット

1 ビットまたは 2 ビット長を選択できます。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、フレーミングエラーが検出できます。

動作モード 1 では、オーバラン、フレーミングエラーが検出できます。パリティエラーは検出できません。

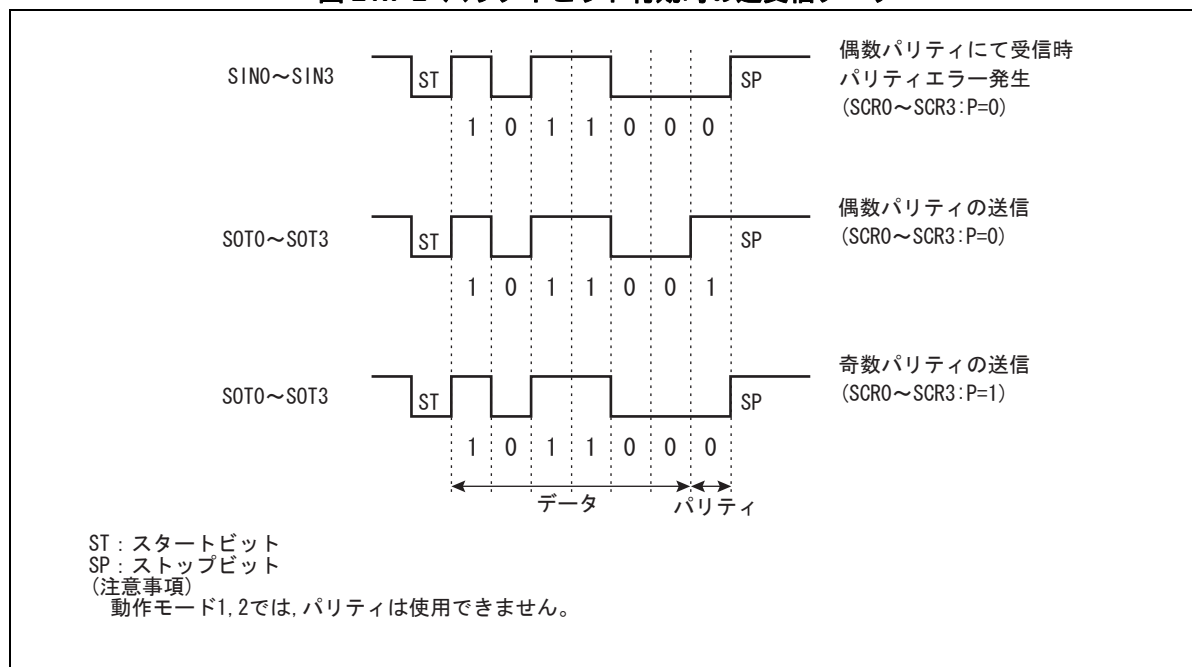
● パリティビット

パリティビットの付加は、動作モード 0 の場合のみ設定できます。パリティ付加許可ビット(SCR0 ~ SCR3:PEN)でパリティの有無を、パリティ選択ビット(SCR0 ~ SCR3:P)で偶数パリティ / 奇数パリティを設定できます。

動作モード 1, 動作モード 2 では、パリティの付加はできません。

パリティ有効時の送受信データを図 21.7-2 に示します。

図 21.7-2 パリティビット有効時の送受信データ



21.7.2 同期モード (動作モード 2) 時の動作

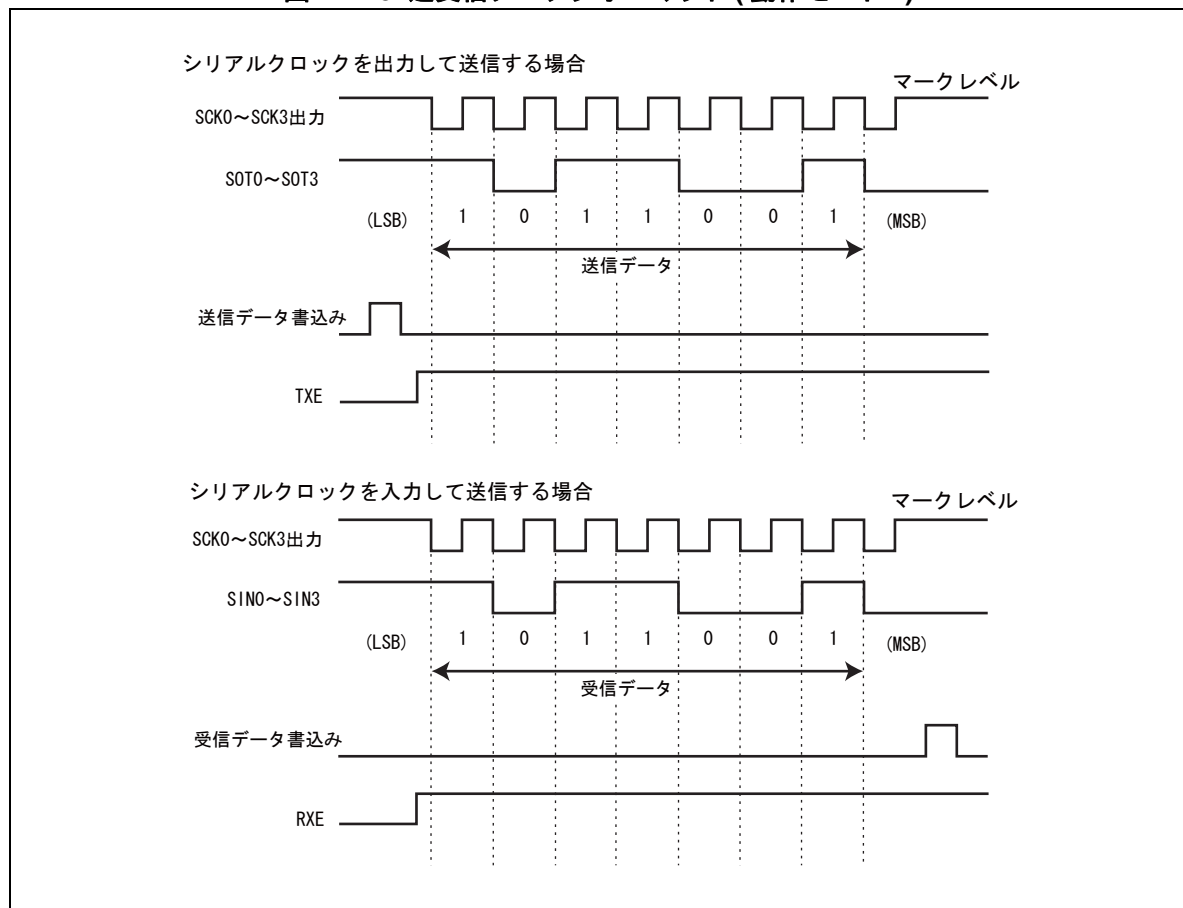
UART 動作モード 2(ノーマルモード) で使用する場合 , 転送方式はクロック同期となります。

■ 同期モード (動作モード 2) 時の動作

● 送受信データフォーマット

同期モードでは , 1 ~ 8 ビットデータを転送し , スタートビット , ストップビットは付加されません。図 21.7-3 にクロック同期モード時の送受信データフォーマットを示します。

図 21.7-3 送受信データフォーマット (動作モード 2)



● クロック供給

クロック同期方式では , 送受信を行うフレームの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。

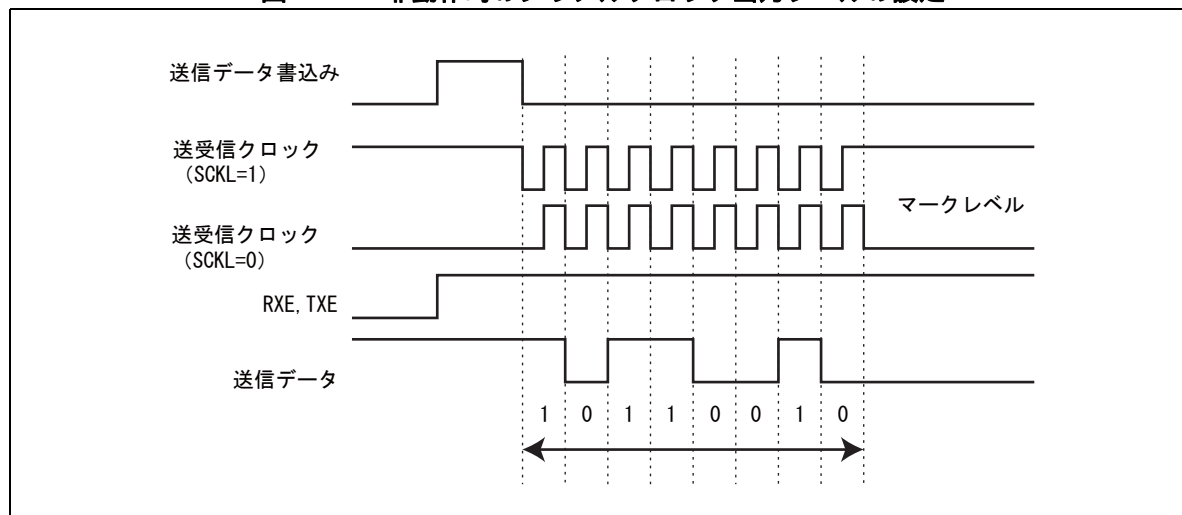
内部クロック (専用ボーレートジェネレータ) を選択している場合は , データを送信するとデータ受信用同期クロックが自動的に生成されます。

外部クロックを選択している場合は , シリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に送信データがあること (SSR0 ~ SSR3:TDRE=0) を確認してから , 正確に 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。また , 送信前と終了後は , 必ずマークレベル ("H") にする必要があります。

● 非動作時のシリアルクロック出力レベルの指定

クロック同期モード時, 非動作時のシリアルクロック出力レベルの設定(SMR:SCKL)ができます。

図 21.7-4 非動作時のシリアルクロック出力レベルの設定



● 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の, 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 0 ~ 3(SMR0 ~ SMR3)]

MD1, MD0 : "10_B"

SCKL : 非動作時のシリアルクロック出力レベルを "H" にする場合は "1" に設定

: 非動作時のシリアルクロック出力レベルを "L" にする場合は "0" に設定

M2L2 ~ M2L0 : 転送数 8 ビットの場合は, "000_B" に設定

: 転送数 7 ビットの場合は, "111_B" に設定

: 転送数 6 ビットの場合は, "110_B" に設定

: 転送数 5 ビットの場合は, "101_B" に設定

: 転送数 4 ビットの場合は, "100_B" に設定

: 転送数 3 ビットの場合は, "011_B" に設定

: 転送数 2 ビットの場合は, "010_B" に設定

: 転送数 1 ビットの場合は, "001_B" に設定

SCKE : 専用ボーレートジェネレータの場合 "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 0 ~ 3(SCR0 ~ SCR3)]

PEN : "0"

P, SBL, A/D : 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 0 ~ 3(SSR0 ~ SSR3)]

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 0 ~ 3(SODR0 ~ SODR3) に送信データを書き込むと , 通信が開始されます。受信のみ行う場合でも通信を開始する場合は , 必ず仮の送信データをシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に書き込む必要があります。

● 通信終了

1 フレームのデータ送受信が終了すると , 受信データフルフラグビット (SSR0 ~ SSR3:RDRF) が "1" にセットされます。データを受信した場合には , オーバランエラーフラグビット (SSR0 ~ SSR3:ORE) をチェックし , 通信が正常に行われたかどうかを判断してください。

21.7.3 双方向通信機能 (ノーマルモード)

動作モード 0, 2 で, 1 対 1 接続の通常のシリアル双方向通信ができます。同期方式は, 動作モード 0 の場合は非同期, 動作モード 2 の場合は同期となります。

■ 双方向通信機能

UART をノーマルモード (動作モード 0, 2) で動作させるためには, 図 21.7-5 の設定が必要です。

図 21.7-5 UART の動作モード 0 の設定

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SCR3~SCR0, SMR3~SMR30	PEN	P	SBL	CL	A/D	REC	RXE	TXE	MD1	MD0	SCKL	M2L2	M2L1	M2L0	SCKE	SOE
モード 0 →	⊙	⊙	⊙	⊙	X	0	⊙	⊙	0	0	X	X	X	X	⊙	⊙
モード 2 →	0	X	X	1	X	0	⊙	⊙	1	0	⊙	⊙	⊙	⊙	⊙	⊙

	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE	
SSR3~SSR0, SIDR3~SIDR0/ SODR3~SODR0									送信データを設定 (ライト時) / 受信データを保持 (リード時)
モード 0 →	⊙	⊙	⊙	⊙	⊙	⊙	⊙	⊙	
モード 2 →	X	⊙	X	⊙	⊙	⊙	⊙	⊙	

⊙ : 使用ビット
X : 未使用ビット
1 : "1" を設定
0 : "0" を設定
△ : 端子の入力を使用する場合 "0" を設定

DDR4

D47	D46	D45	D44	D43	D42	D41	D40
△		△	△		△		

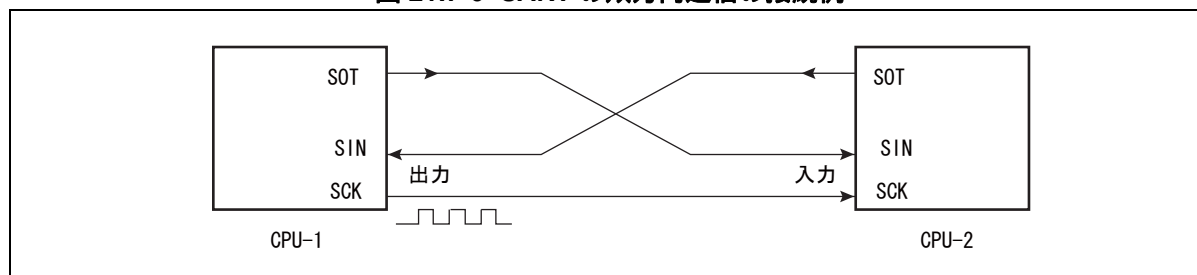
DDR9

	D96	D95	D94	D93	D92	D91	D90
		△		△	△		△

● CPU 間接続

図 21.7-6 に示すように, 2 つの CPU を相互に接続します。

図 21.7-6 UART の双方向通信の接続例

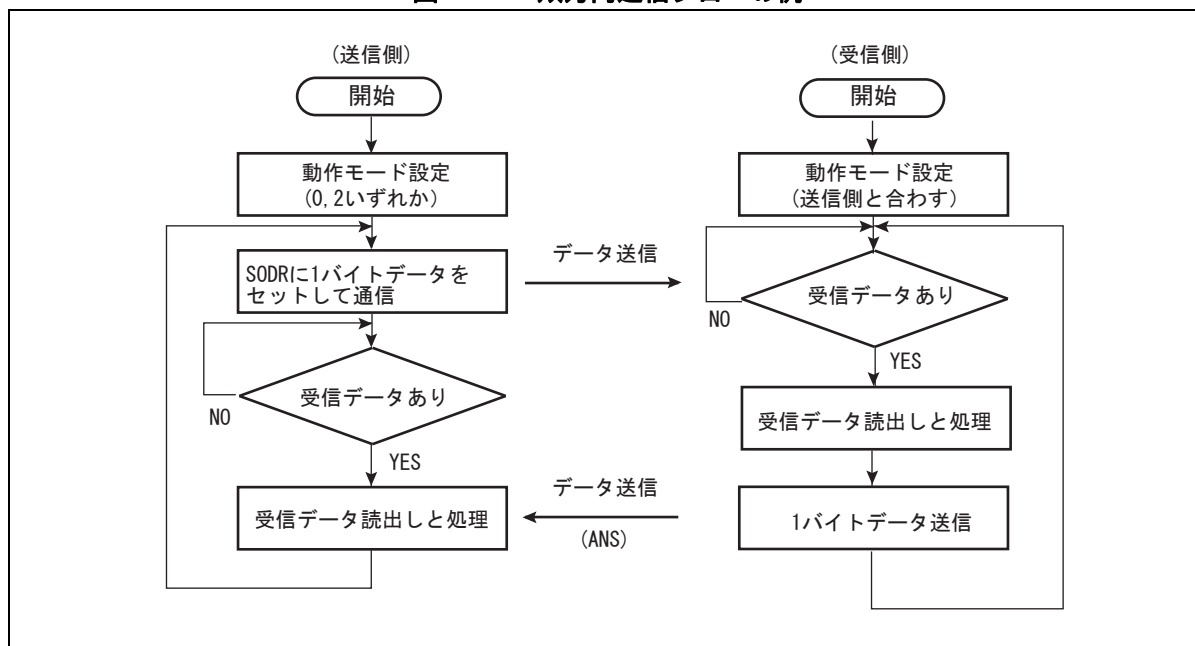


● 通信手順

通信は、送信データが準備できたときに任意のタイミングで送信側から開始します。送信側はシリアル出力データレジスタ 0 ~ 3(SODR0 ~ SODR3) に送信データを設定し、シリアル制御レジスタの送信動作許可ビット (SCR0 ~ SCR3:TXE) を "1" に設定して、送信を開始してください。

正常に受信したことを送信側に示すために、受信データを送信側に転送する例を図 21.7-7 に示します。受信側で送信データを受け取ると定期的に、ANS(例では 1 バイトごと) を返します。

図 21.7-7 双方向通信フローの例



21.7.4 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 CPU のマスタ/スレーブ型接続による通信が可能です。ただし、マスタとしてのみ使用できます。

■ マスタ/スレーブ型通信機能

UART をマルチプロセッサモード (動作モード 1) で動作させるためには、図 21.7-8 の設定が必要です。

図 21.7-8 UART の動作モード 1 の設定

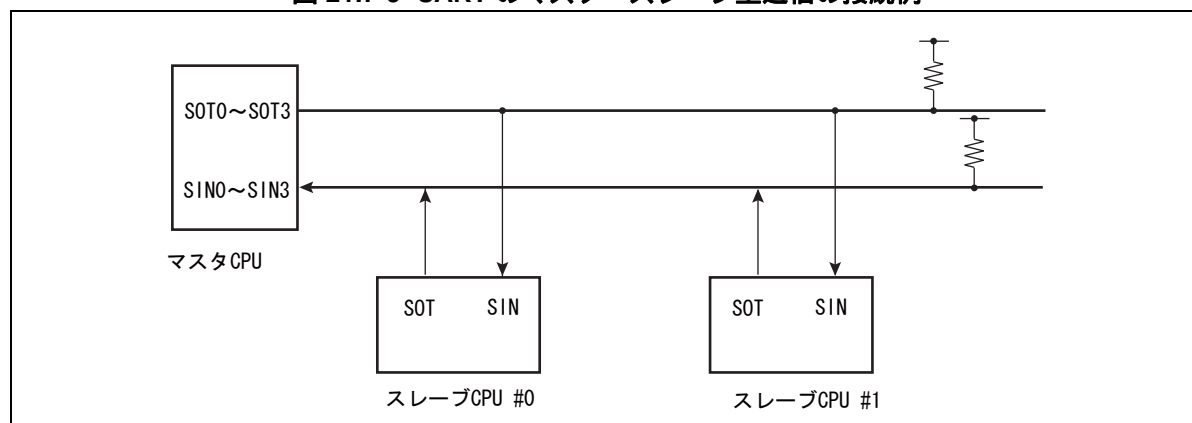
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SCR3~SCR0, SMR3~SMR0	PEN	P	SBL	CL	A/D	REC	RXE	TXE	MD1	MD0	SCKL	M2L2	M2L1	M2L0	SCKE	SOE
	0	X	⊙	1	⊙	0	⊙	⊙	0	1	X	X	X	X	⊙	⊙
SSR3~SSR0, SIDR3~SIDR0/ SODR3~SODR0	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE	送信データを設定 (ライト時) / 受信データを保持 (リード時)							
	X	⊙	⊙	⊙	⊙	⊙	⊙	⊙								
DDR4	D47	D46	D45	D44	D43	D42	D41	D40								
	△		△	△		△										
DDR9		D96	D95	D94	D93	D92	D91	D90								
			△		△	△		△								

⊙ : 使用ビット
 X : 未使用ビット
 1 : "1" を設定
 0 : "0" を設定
 △ : 端子の入力を使用する場合 "0" を設定

● CPU 間接続

図 21.7-9 に示すように、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 を接続して、通信システムを構成します。UART はマスタ CPU としてのみ使用できます。

図 21.7-9 UART のマスタ・スレーブ型通信の接続例



● 機能選択

マスタ / スレーブ型通信では、表 21.7-2 のように、動作モードとデータ転送方式を選択してください。動作モード 1 では、パリティチェック機能は使用できないので、パリティ許可ビット (SCR0 ~ SCR3:PEN) は "0" に設定してください。

表 21.7-2 マスタ / スレーブ型通信機能の選択

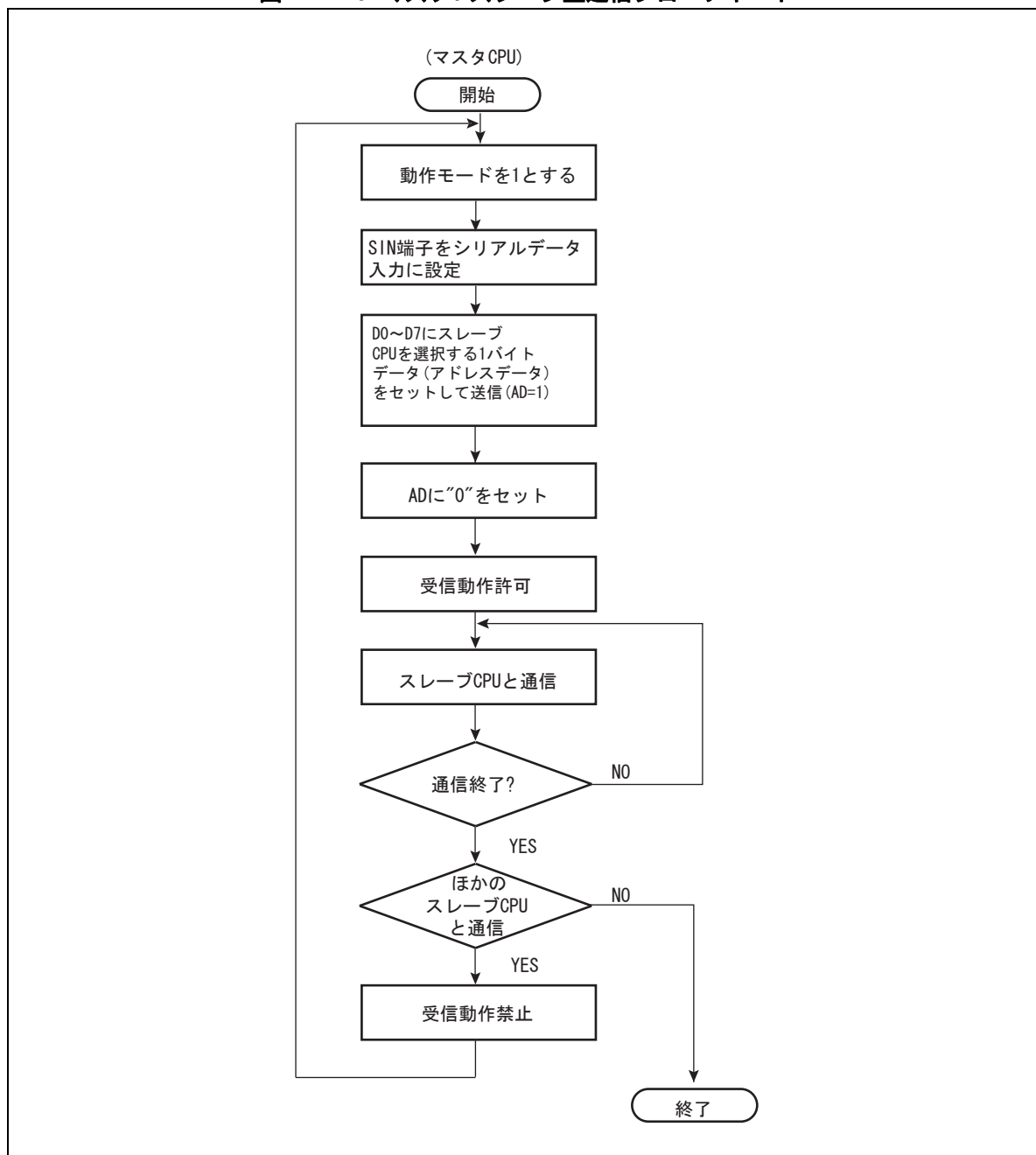
	動作モード		データ	パリティ	同期方式	ストップビット
	マスタ CPU	スレーブ CPU				
アドレス送受信	モード 1		AD= 1 + 8 ビット アドレス	なし	非同期	1 ビット または 2 ビット
データ送受信			AD= 0 + 8 ビット データ			

● 通信手順

通信はマスタ CPU がアドレスデータを送信することで始まります。アドレスデータとは、AD ビットを "1" としたデータで、通信先となるスレーブ CPU を選択します。各スレーブ CPU はプログラムで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、マスタ CPU との通信 (通常データ) をします。

図 21.7-10 に、マスタ / スレーブ型通信 (マルチプロセッサモード) のフローチャートを示します。

図 21.7-10 マスタ/スレーブ型通信フローチャート



21.8 UART 使用上の注意

UART を使用する場合は以下の点に注意してください。

■ UART 使用上の注意

● 送受信の動作許可

- ・送信, 受信のそれぞれに対して送信動作許可ビット (SCR0 ~ SCR3:TXE), 受信動作許可ビット (SCR0 ~ SCR3:RXE) があります。
- ・リセット後の初期状態では, 送受信とも動作禁止に設定 (SCR0 ~ SCR3:TXE=0, RXE=0) されているので, 送受信を行う前に動作を許可する必要があります。
- ・送受信動作を禁止 (SCR0 ~ SCR3:TXE=0, RXE=0) して, 送受信を中止できます。

● 動作モードの設定について

動作モードは, 送受信動作を禁止 (SCR0 ~ SCR3:TXE=0, RXE=0) してから, 設定してください。送受信中に動作モードを切り換えた場合, 送受信されたデータは保証されません。

● クロック同期モードについて

動作モード 2 は, クロック同期方式として設定されます。送受信データにスタートビット, ストップビットは付加されません。

● 送信割込み許可のタイミング

送信データエンプティフラグビット (SSR0 ~ SSR3:TDRE) は, リセット後の初期値が "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 ~ SSR3: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 ~ SSR3:TIE=1) してください。

● マルチプロセッサモードの受信

マルチプロセッサモードにおいては, 9 ビット受信での受信動作はできません。

21.9 UART のプログラム例

UART のプログラム例を示します。

■ UART のプログラム例

● 処理仕様

UART の双方向通信機能 (ノーマルモード) を使用して , シリアル送受信をします。
動作モード 0, 非同期 , データ長 8 ビット , ストップビット長 2 ビット , パリティなしと
します。

通信には , P42/A10/SIN0, P43/A11/SOT0 端子を使用します。

専用ボーレートジェネレータを使用し , ボーレートの設定は , 約 9600bps にします。

SOT0 端子からキャラクタ "13_H" を送出し , 割込みで受信します。

マシクロック (ϕ) は , 16 MHz とします。

● コーディング例

```
ICR14    EQU    0000BEH        ; UART 送受信割込み用制御レジスタ
DDR4     EQU    000014H        ; ポート 4 方向レジスタ
SMR0     EQU    000020H        ; シリアルモードレジスタ 0
SCR0     EQU    000021H        ; シリアル制御レジスタ 0
SIDR0    EQU    000022H        ; シリアル入力データレジスタ 0
SODR0    EQU    000022H        ; シリアル出力データレジスタ 0
SSR0     EQU    000023H        ; シリアルステータスレジスタ 0
UTCR0    EQU    000025H        ; UART プリスケラ制御レジスタ 0
UTRLR0   EQU    000024H        ; UARTプリスケラリロードレジスタ0
REC      EQU    SCR:2          ; 受信エラーフラグクリアビット

;----- メインプログラム -----
CODE     CSEG    ABS = 0FFH
START:

;          :          ; スタックポインタ (SP) などは初期化
;          :          ; 済みとする

          AND     CCR, #0BFH    ; 割込み禁止
          MOV     I:ICR14, #00H ; 割込みレベル 0( 最強 )
          MOV     I:DDR4, #00000000B ; SIN0 端子を入力に設定
          MOV     I:SMR0, #00000001B ; 動作モード 0( 非同期 )
          ;          ; クロック出力禁止 , データ出力許可
          MOVW    I:UTRL0, #81A0H ; 専用ボーレートジェネレータ使用
          ;          ; (9615bps 選択 )
```

```

MOV      I:SCR0, #00110011B    ; パリティなし, ストップビット2ビット
                                   ; データ長8ビット, 受信エラーフラグク
                                   ; リア
                                   ; 送受信動作許可

MOV      I:SSR0, #00000010B    ; 送信割込み禁止, 受信割込み許可
MOV      I:SODR0, #13H         ; 送信データ書込み
MOV      ILM, #07H             ; PS 内 ILM をレベル7 に設定
OR       CCR, #40H             ; 割込み許可
LOOP:    MOV      A, #00H        ; 無限ループ
MOV      A, #01H
BRA      LOOP

;----- 割込みプログラム -----
WARI:
MOV      A, SIDR0               ; 受信データの読出し
CLRB     I:REC                 ; 受信割込み要求フラグをクリア
;
;      ユーザ処理
;
RETI                      ; 割込みから復帰
CODE     ENDS

;----- ベクタ設定 -----
VECT     CSEG      ABS=0FFH
ORG      0FF60H      ; 割込み #39(27H) にベクタを設定
DSL      WARI
ORG      0FFDCH      ; リセットベクタ設定
DSL      START
DB       00H         ; シングルチップモードに設定
VECT     ENDS

```


第22章

I²C インタフェース

この章では、I²C インタフェースの概要、レジスタの構成 / 機能および I²C インタフェースの動作について説明します。

22.1 I²C インタフェースの概要

22.2 I²C インタフェースのレジスタ

22.3 I²C インタフェースの動作

22.1 I²C インタフェースの概要

I²C インタフェースは、Inter IC BUS をサポートするシリアル I/O ポートです。I²C バス上のマスタ/スレーブデバイスとして動作します。

■ I²C インタフェースの機能

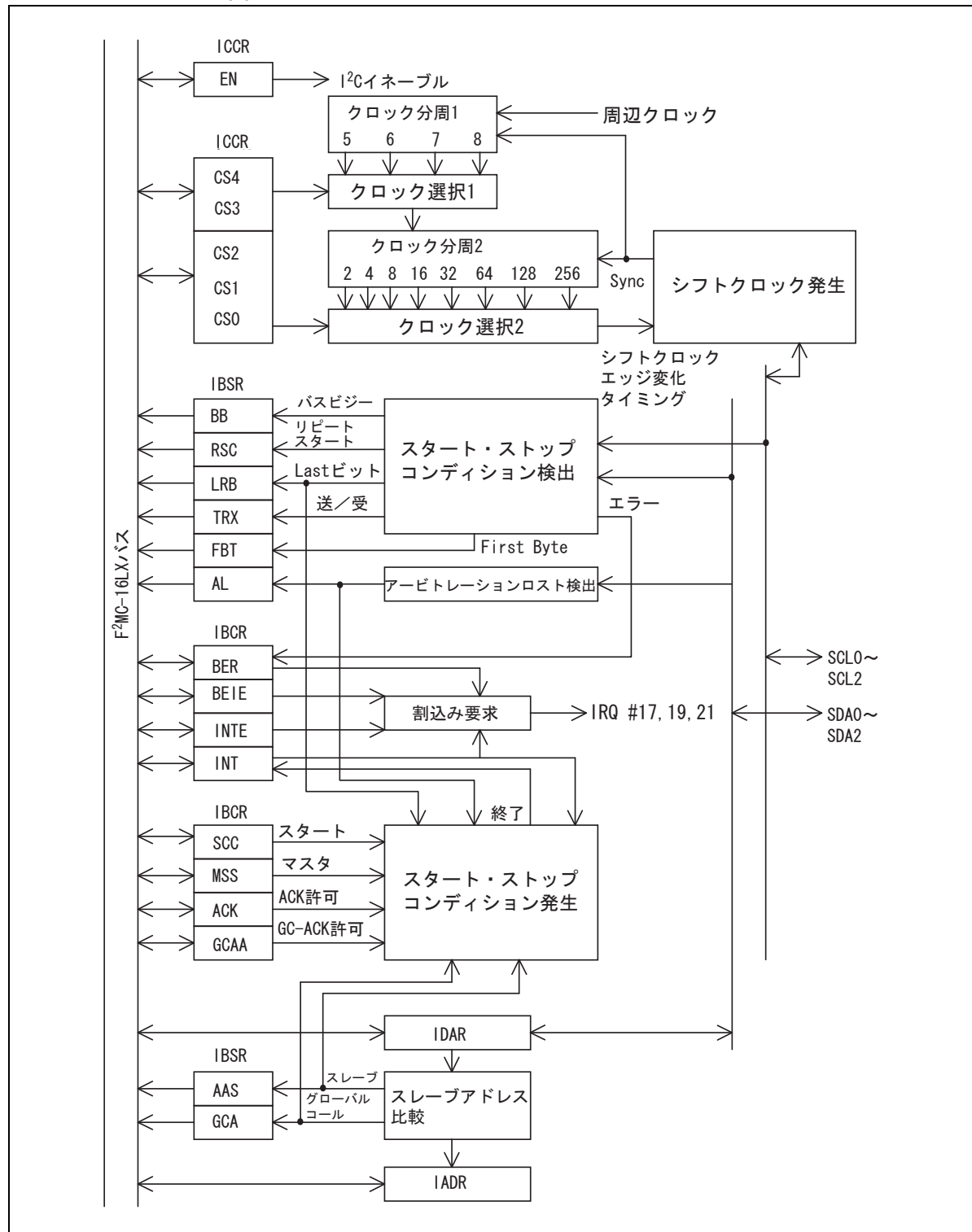
I²C インタフェースは、以下の機能をもっています。

- マスタ/スレーブ送受信
- アービトレーション機能
- クロック同期化機能
- スレーブアドレス/ゼネラルコールアドレス検出機能
- 転送方向検出機能
- スタートコンディションの繰返し発生および検出機能
- バスエラー検出機能

■ I²C インタフェースのブロックダイアグラム

図 22.1-1 に、I²C インタフェースのブロックダイアグラムを示します。

図 22.1-1 I²C インタフェースのブロックダイアグラム



22.2 I²C インタフェースのレジスタ

I²C インタフェースで使用するレジスタの構成および機能について説明します。

■ I²C インタフェースのレジスタ一覧

図 22.2-1 I²C インタフェースのレジスタ一覧

bit 7 6 5 4 3 2 1 0 IBSR0 ~ IBSR2								
ch.0 : 000070 _H	BB	RSC	AL	LRB	TRX	AAS	GCA	FBT
ch.1 : 000076 _H								
ch.2 : 00007C _H								
	R	R	R	R	R	R	R	R
I ² C バスステータスレジスタ 初期値 00000000 _B 読出し / 書込み								
bit 15 14 13 12 11 10 9 8 IBCR0 ~ IBCR2								
ch.0 : 000071 _H	BER	BEIE	SCC	MSS	ACK	GCAA	INTE	INT
ch.1 : 000077 _H								
ch.2 : 00007D _H								
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
I ² C バスコントロールレジスタ 初期値 00000000 _B 読出し / 書込み								
bit 7 6 5 4 3 2 1 0 ICCR0 ~ ICCR2								
ch.0 : 000072 _H	-	-	EN	CS4	CS3	CS2	CS1	CS0
ch.1 : 000078 _H								
ch.2 : 00007E _H								
	-	-	R/W	R/W	R/W	R/W	R/W	R/W
I ² C クロックコントロールレジスタ 初期値 --0XXXXX _B 読出し / 書込み								
bit 7 14 13 12 11 10 9 8 IADR0 ~ IADR2								
ch.0 : 000073 _H	-	A6	A5	A4	A3	A2	A1	A0
ch.1 : 000079 _H								
ch.2 : 00007F _H								
	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
I ² C バスアドレスレジスタ 初期値 -XXXXXXXX _B 読出し / 書込み								
bit 7 6 5 4 3 2 1 0 IDAR0 ~ IDAR2								
ch.0 : 000074 _H	D7	D6	D5	D4	D3	D2	D1	D0
ch.1 : 00007A _H								
ch.2 : 000080 _H								
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
I ² C バスデータレジスタ 初期値 XXXXXXXX _B 読出し / 書込み								

22.2.1 I²C バスステータスレジスタ 0 ~ 2(IBSR0 ~ IBSR2)

I²C バスステータスレジスタ 0 ~ 2(IBSR0 ~ IBSR2) の構成および機能について説明します。

■ I²C バスステータスレジスタ 0 ~ 2(IBSR0 ~ IBSR2)

図 22.2-2 に、I²C バスステータスレジスタ 0 ~ 2(IBSR0 ~ IBSR2) のビット構成を示します。

図 22.2-2 I²C バスステータスレジスタ 0 ~ 2(IBSR0 ~ IBSR2) のビット構成

	bit 7	6	5	4	3	2	1	0	IBSR0 ~ IBSR2
ch.0 : 000070 _H									I ² C バスステータス レジスタ 初期値 00000000 _B 読出し / 書込み
ch.1 : 000076 _H	BB	RSC	AL	LRB	TRX	AAS	GCA	FBT	
ch.2 : 00007C _H	R	R	R	R	R	R	R	R	

以下に、I²C バスステータスレジスタ 0 ~ 2(IBSR0 ~ IBSR2) の各ビットの機能を説明します。

[bit7] BB: Bus Busy

I²C バスの状態を示すビットです。

0	ストップコンディションを検出した。
1	スタートコンディションを検出した (バスは使用されている)。

[bit6] RSC :Repeated Start Condition

繰返しスタートコンディション検出ビットです。

0	繰返しスタートコンディションが検出されていない。
1	バス使用中に、スタートコンディションを検出した。

INT ビットへの "0" 書込み、スレープ時にアドレスされなかった場合、バス停止中のスタートコンディション検出またはストップコンディション検出でクリアされます。

[bit5] AL :Arbitration Lost

アービトレーションロスト検出ビットです。

0	アービトレーションロスト検出されていない。
1	マスタ送信中にアービトレーションロストが発生、またはほかのシステムがバスを使用中、MSS ビットに "1" を書き込んだ。

INT ビットへの "0" 書込みによってクリアされます。

[bit4] LRB :Last Recieved Bit

アクノリッジ格納ビットです。受信側からのアクノリッジを格納します。

0	受信を確認した。
1	送信が確認されない。

スタートコンディションまたはストップコンディションの検出によってクリアされます。

[bit3] TRX :Transfer/Recieve

データ転送の送受信を示すビットです。

0	受信状態。
1	送信状態。

[bit2] AAS :Addressed As Slave

アドレッシング検出ビットです。

0	スレーブ時に、アドレッシングされていない。
1	スレーブ時に、アドレッシングされた。

スタートコンディションまたはストップコンディションの検出でクリアされます。

[bit1] GCA :General Call Address

ゼネラルコールアドレス (00_H) 検出ビットです。

0	スレーブ時に、ゼネラルコールアドレスは受信されていない。
1	スレーブ時に、ゼネラルコールアドレスを受信した。

スタートコンディションまたはストップコンディションの検出でクリアされます。

[bit0] FBT:First Byte Transfer

第 1 バイト検出ビットです。

0	受信データは第 1 バイト以外である。
1	受信データが第 1 バイト (アドレスデータ) である。

スタートコンディションの検出で "1" にセットされても、INT ビットの "0" 書込みまたはスレーブ時にアドレスされなかった場合にクリアされます。

22.2.2 I²C バスコントロールレジスタ 0 ~ 2(BCR0 ~ IBCR2)

I²C バスコントロールレジスタ 0 ~ 2(BCR0 ~ IBCR2) の構成および機能について説明します。

■ I²C バスコントロールレジスタ 0 ~ 2(BCR0 ~ IBCR2)

図 22.2-3に、バスコントロールレジスタ0~2(BCR0~IBCR2)のビット構成を示します。

図 22.2-3 I²C バスコントロールレジスタ 0 ~ 2(BCR0 ~ IBCR2) のビット構成

bit	15	14	13	12	11	10	9	8	BCR0 ~ IBCR 2
ch.0 : 000071 _H ch.1 : 000077 _H ch.2 : 00007D _H	BER	BEIE	SCC	MSS	ACK	GCAA	INTE	INT	I ² C バスコントロール レジスタ 初期値 00000000 _B 読出し / 書込み
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

以下に、バスコントロールレジスタ 0 ~ 2(BCR0 ~ IBCR2) の各ビットの機能を説明します。

[bit15] BER: Bus Error

バスエラー割込み要求フラグです。書込み時と読出し時で機能が異なります。
(書込み時)

0	バスエラー割込み要求フラグをクリアする。
1	動作に影響を与えません。

(読出し時)

0	バスエラーは検出されていない。
1	データ転送中に、不正なスタート、ストップコンディションを検出した。

BER ビットがセットされた場合、ICCR の EN ビットはクリアされて I²C インタフェースは停止状態になり、データ転送は中断されます。

[bit14] BEIE :Bus Error Interrupt Enable

バスエラー割込み許可ビットです。

0	バスエラー割込み禁止。
1	バスエラー割込み許可。

"1" のとき BER ビットが "1" ならば、割込みを発生します。

[bit13] SCC :Start Condition Continue

スタートコンディション発生ビットです。
(書込み時)

0	動作に影響を与えません。
1	マスタ転送時に再びスタートコンディションを発生させる。

読出し値は、常に "0" です。

[bit12] MSS :Master Slave Select

マスタ / スレーブの選択ビットです。

0	ストップコンディションを発生し、転送終了後スレーブモードとなる。
1	マスタモードとなり、スタートコンディションを発生し転送を開始する。

マスタ送信中にアービトレーションロストが発生した場合にクリアされ、スレーブモードになります。

< 注意事項 >

下記の条件で使用する場合、スレーブとして受信できませんのでゼネラルコールアドレスの送信は禁止です。

- ・本 LSI のほかにマスタモードとなるほかの LSI がバス上に存在し、本 LSI がマスタとしてゼネラルコールアドレスを送信し、第 2 バイト目以降でアービトレーションロストが発生する場合

[bit11] ACK :Acknowledge

データを受信した場合のアクノリッジ発生許可ビットです。

0	アクノリッジ発生禁止。
1	アクノリッジ発生許可。

スレーブにおけるアドレスデータ受信時には、無効となります。

[bit10] GCAA :General Call Address Acknowledge

ゼネラルコールアドレスを受信した場合のアクノリッジ発生許可ビットです。

0	アクノリッジ発生禁止。
1	アクノリッジ発生許可。

[bit9] INTE :INTerrupt Enable

割込み許可ビットです。

0	割込み禁止。
1	割込み許可。

"1" のとき INT ビットが "1" ならば、割込みを発生します。

[bit8] INT :INTerrupt

転送終了割込み要求フラグビットです。

(書込み時)

0	転送終了割込み要求フラグをクリアする。
1	動作に影響を与えません。

(読出し時)

0	転送が終了していない。
1	アクリッジビットを含めた 1 バイト転送が終了したときに、次の条件に該当する場合にセット ・バスマスタである。 ・アドレスされたスレーブである。 ・ゼネラルコールアドレスを受信した。 ・アービトレーションロストを発生した。 ・ほかのシステムがバス使用中にスタートコンディションを発生しようとした。

"1" のとき、SCL ラインは "L" レベルに保たれます。"0" を書き込むことによってクリアされ、SCL ラインを開放して次バイトの転送を行います。また、マスタ時にスタートコンディションまたはストップコンディションの発生によって "0" にリセットされます。

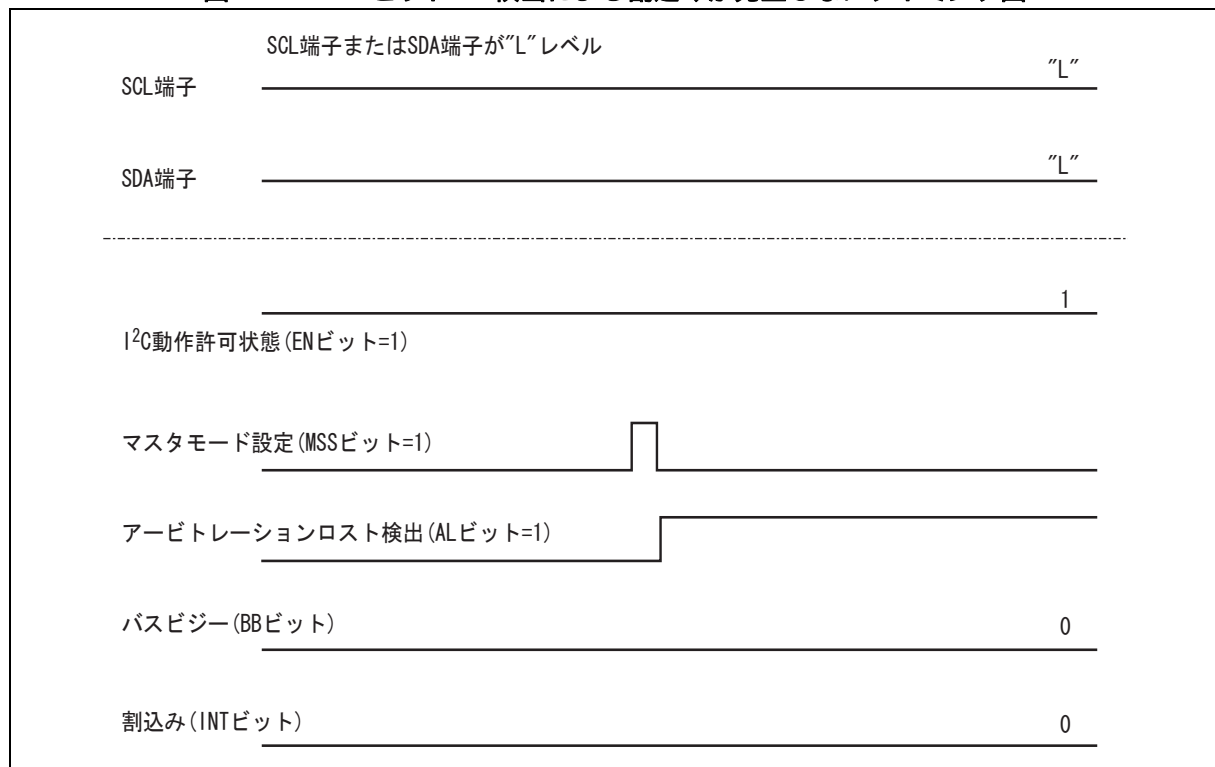
< 注意事項 >

図 22.2-4 と図 22.2-5 に示したタイミングで、スタートコンディションを発生させる命令を実行 (MSS ビットに "1" を設定) すると、アービトレーションロスト検出 (AL ビット = 1) による割込み (INT ビット = 1) が発生しません。

- ・ AL ビット = 1 検出による割込み (INT ビット = 1) が発生しない条件 1

スタートコンディション未検出 (BB ビット = 0) 状態で、SDA 端子または SCL 端子レベルが "L" のときに、スタートコンディションを発生させる命令を実行 (IBCR レジスタの MSS ビットに "1" を設定) した場合。

図 22.2-4 AL ビット = 1 検出による割込みが発生しないタイミング図

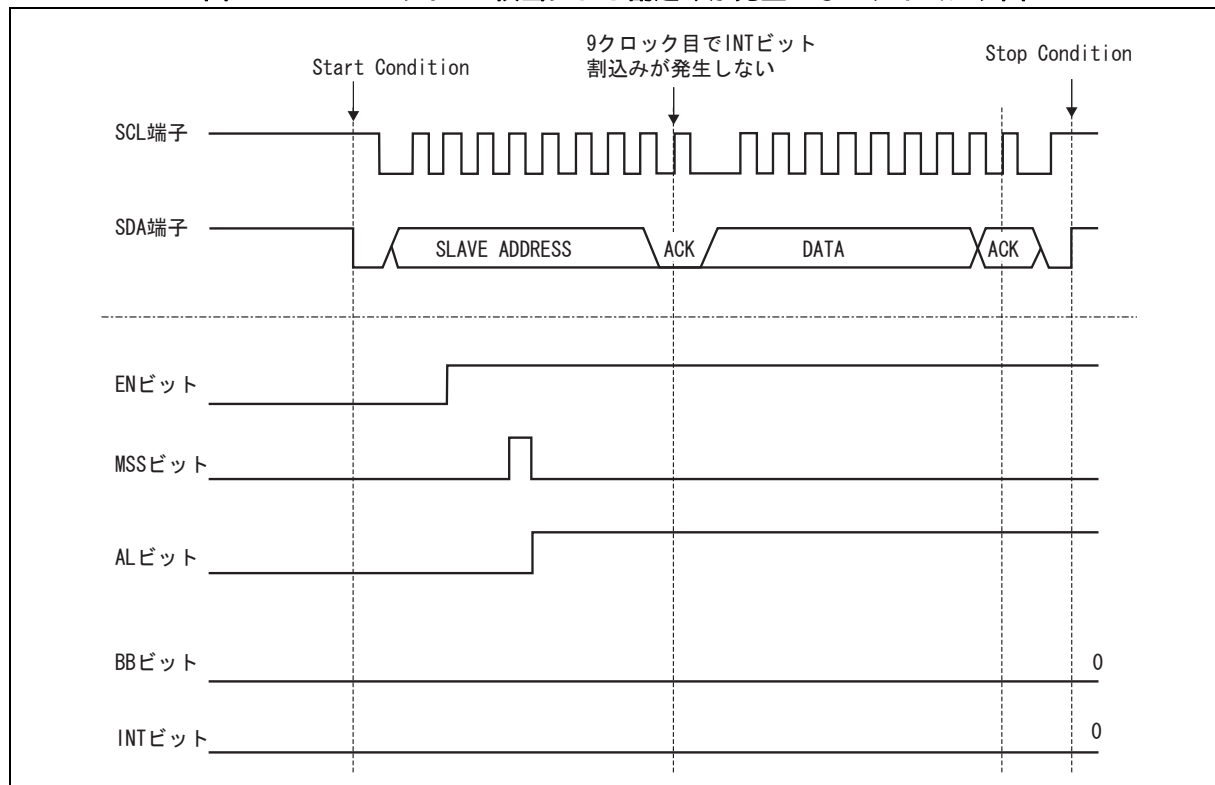


- ・ AL ビット = 1 検出による割込み (INT ビット = 1) が発生しない条件 2

ほかのマスタに I²C バスが占有されている状態で、I²C 動作許可 (EN ビット = 1) してスタートコンディションを発生させる命令を実行 (IBCR レジスタの MSS ビットに "1" を設定) した場合。

図 22.2-5 に示したとおり、I²C が動作禁止 (EN ビット = 0) の状態で、I²C バス上のほかのマスタが通信を開始すると、スタートコンディション未検出 (BB ビット = 0) で I²C バスが占有されている状態になるためです。

図 22.2-5 AL ビット =1 検出による割込みが発生しないタイミング図



上記のような現象が発生する可能性がある場合には、以下の手順でソフト処理をお願いします。

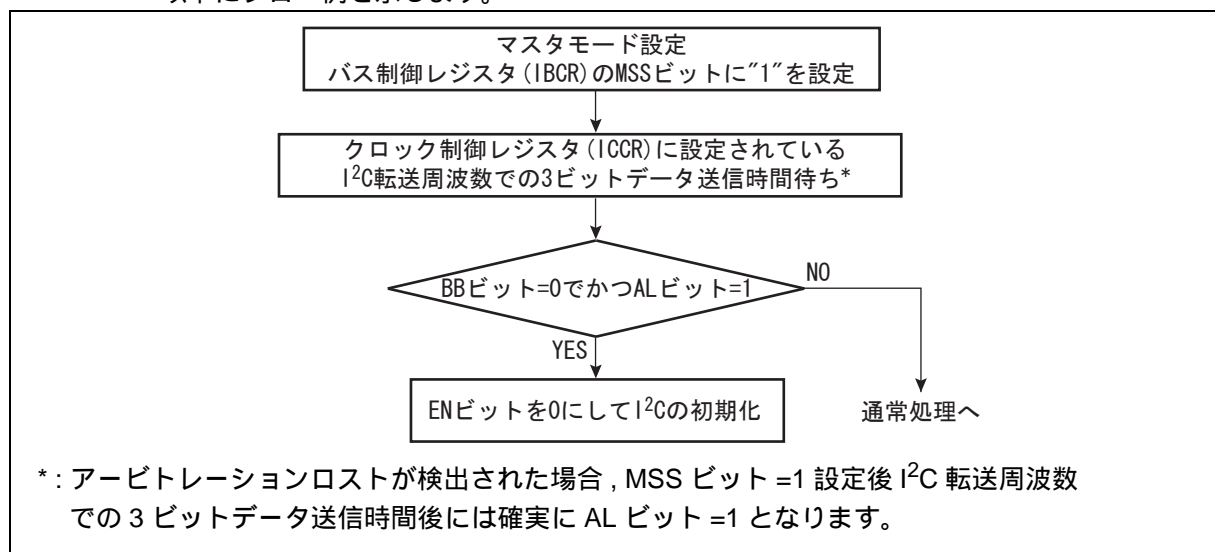
- 1) スタートコンディションを発生させる命令を実行 (MSS ビットに "1" を設定)。
- 2) タイマ機能などを用いて、ICCR レジスタに設定されている I²C 転送周波数での 3 ビットデータ送信時間を待つ。*

例 : I²C 転送周波数 100 kHz の場合

$$3 \text{ ビットデータ送信時間} = \{1/(100 \times 10^3)\} \times 3 = 30 \mu\text{s}$$

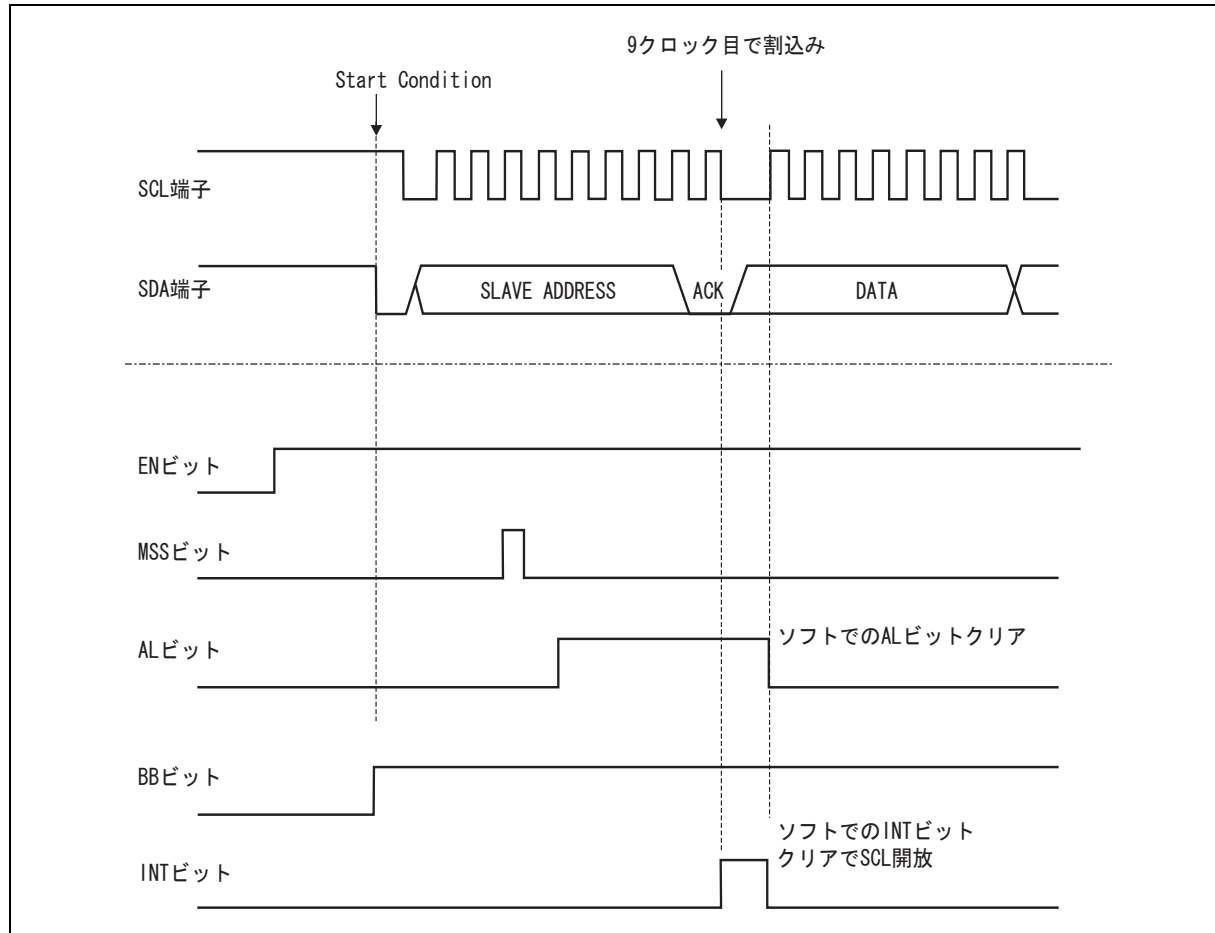
- 3) IBSR レジスタの AL ビットと BB ビットを確認し、AL ビット =1, BB ビット =0 の場合には、ICCR レジスタの EN ビットを "0" にして I²C を初期化する。AL ビットと BB ビットがその他の状態のときは通常処理を行う。

以下にフロー例を示します。



- ・ AL ビット =1 検出による割込み (INT ビット =1) が発生する例
- バスビジー検出 (BB ビット =1) されている状態で、スタートコンディションを発生させる命令を実行 (MSS ビットに "1" を設定) しアービトレーションロストした場合には、AL ビット =1 検出時に INT ビット割込みが発生します。

図 22.2-6 AL ビット =1 時に割込みが発生するタイミング図



■ I²C バスコントロールレジスタ 0 ~ 2 (IBCR0 ~ IBCR2) 使用上の注意

SCC, MSS, INT ビットの競合について、以下の注意が必要です。

SCC, MSS, INT ビットの同時書込みにより、次バイト転送、スタートコンディション発生およびストップコンディション発生の競合が起こります。この場合の優先度は、次のようになります。

- 次のバイト転送とストップコンディション発生
INT ビットに "0", MSS ビットに "0" を書き込むと、MSS ビットの "0" 書込みが優先され、ストップコンディションが発生します。
- 次のバイト転送とスタートコンディション発生
INT ビットに "0", SCC ビットに "1" を書き込むと、SCC ビットの "1" 書込みが優先され、スタートコンディションが発生します。
- スタートコンディション発生とストップコンディション発生
SCC ビットに "1", MSS ビットに "0" の同時書込みは、禁止です。

22.2.3 I²C バスクロックコントロールレジスタ 0 ~ 2 (ICCR0 ~ ICCR2)

I²C バスクロックコントロールレジスタ 0 ~ 2(ICCR0 ~ ICCR2) の構成および機能について説明します。

■ I²C バスクロックコントロールレジスタ 0 ~ 2(ICCR0 ~ ICCR2)

図 22.2-7 に、I²C バスクロックコントロールレジスタ 0 ~ 2(ICCR0 ~ ICCR2) のビット構成を示します。

図 22.2-7 I²C バスクロックコントロールレジスタ 0 ~ 2(ICCR0 ~ ICCR2) のビット構成

	bit 7	6	5	4	3	2	1	0	ICCR0 ~ ICCR2
ch.0 : 000072 _H ch.1 : 000078 _H ch.2 : 00007E _H	-	-	EN	CS4	CS3	CS2	CS1	CS0	I ² C クロックコントロールレジスタ 初期値 XX0XXXX _B 読出し / 書込み
	-	-	R/W	R/W	R/W	R/W	R/W	R/W	

以下に、I²C バスクロックコントロールレジスタ 0 ~ 2(ICCR0 ~ ICCR2) の機能を説明します。

[bit7, bit6] 未定義ビット

読出し値は不定です。書込みは動作に影響を与えません。

[bit5] EN :ENable

I²C インタフェースの動作許可ビットです。

0	動作禁止
1	動作許可

- "0" のとき、IBSR および IBCR(BER, BEIE ビットを除く) の各ビットがクリアされます。
- BER ビットがセットされると、クリアされます。

< 注意事項 >

- I²C インタフェースの動作が禁止されると、送受信は直ちに停止されます。
- MSS ビットに "0" を書き込んでストップコンディションを発生させた後に I²C インタフェースの動作を禁止する場合は、ストップコンディションの発生を確認 (IBSR の BB=0) した後、動作を禁止 (ICCR の EN=0) してください。

[bit4 ~ bit0] CS4 ~ CS0 :Clock Period Select4 ~ Clock Period Select0

シリアルクロックの周波数を設定するビットです。シフトクロックの周波数 f_{sck} は、次式のように設定されます。

$$f_{sck} = \frac{\phi}{m \times n + 4} \quad \phi: \text{マシンのクロック}$$

< 注意事項 >

+4のサイクルは、SCL端子の出力レベルが変化したことをチェックするための最小のオーバーヘッドです。SCL 端子の立上りのディレイが大きい場合や、スレーブデバイスでクロックを引き延ばしている場合には、この値より大きくなります。なお、シリアルクロックの周波数は 100 kHz 以上に設定しないでください。

m, n は、CS4 ~ CS0 に対して表 22.2-1 に示すようになります。

表 22.2-1 シリアルクロック周波数設定

m	CS4	CS3
5	0	0
6	0	1
7	1	0
8	1	1

n	CS2	CS1	CS0
4	0	0	0
8	0	0	1
16	0	1	0
32	0	1	1
64	1	0	0
128	1	0	1
256	1	1	0
512	1	1	1

22.2.4 I²C バスアドレスレジスタ 0 ~ 2(IADR0 ~ IADR2)

I²C バスアドレスレジスタ 0 ~ 2(IADR0 ~ IADR2) の構成および機能について説明します。

■ I²C バスアドレスレジスタ 0 ~ 2 (IADR0 ~ IADR2)

図 22.2-8に、I²Cバスアドレスレジスタ0 ~ 2(IADR0 ~ IADR2)のビット構成を示します。

図 22.2-8 I²C バスアドレスレジスタ 0 ~ 2(IADR0 ~ IADR2) のビット構成

	bit 15	14	13	12	11	10	9	8	IADR0 ~ IADR2
ch.0 : 000073 _H	-	A6	A5	A4	A3	A2	A1	A0	I ² C バスアドレスレジスタ 初期値 XXXXXXXX _B 読出し / 書込み
ch.1 : 000079 _H	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
ch.2 : 00007F _H	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

[bit14 ~ bit8] A6 ~ A0

スレーブアドレスビットです。スレーブアドレスを指定するレジスタです。スレーブ時、アドレスデータ受信後に IDAR レジスタとの比較が行われ、一致している場合はマスタに対してアクノリッジを送信します。

22.2.5 I²C バスデータレジスタ 0 ~ 2(IDAR0 ~ IDAR2)

I²C バスデータレジスタ 0 ~ 2(IDAR0 ~ IDAR2) の構成および機能について説明します。

■ I²C バスデータレジスタ 0 ~ 2(IDAR0 ~ IDAR2)

図 22.2-9 に、I²C バスデータレジスタ 0 ~ 2(IDAR0 ~ IDAR2) のビット構成を示します。

図 22.2-9 I²C バスデータレジスタ 0 ~ 2(IDAR0 ~ IDAR2) のビット構成

	bit	7	6	5	4	3	2	1	0	IDAR0 ~ IDAR2
ch.0 : 000074 _H		D7	D6	D5	D4	D3	D2	D1	D0	I ² C バスデータレジスタ 初期値 XXXXXXXX _B 読出し / 書込み
ch.1 : 00007A _H		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
ch.2 : 000080 _H		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

[bit7 ~ bit0] D7 ~ D0

データビットです。

シリアル転送に使用されるデータレジスタであり、MSB から転送されます。データ受信時 (TRX=0) は、データ出力値は "1" になります。

IDAR レジスタの書込み側はダブルバッファになっており、バスが使用中 (BB=1) である場合、書込みデータは各バイト転送時にシリアル転送用のレジスタにロードされます。読出し時は、シリアル転送用のレジスタを直接読み出すため、受信データは INT ビットがセットされている場合のみ有効です。

22.3 I²C インタフェースの動作

I²C バスは、1 本のシリアルデータライン (SDA) と 1 本のシリアルクロックライン (SCL) の 2 本の双方向バスラインによって通信が行われます。I²C インタフェースは、それに対して 2 本のオープンドレイン入出力端子 (SDA, SCL) を有し、ワイヤード論理を可能にします。

■ スタートコンディション

バスが開放されている状態 (BB=0, MSS=0) で MSS ビットに "1" を書き込むと、I²C インタフェースはマスタモードとなり、同時にスタートコンディションを発生します。マスタモードでは、バスが使用状態 (BB=1) であっても、SCC ビットに "1" を書き込むことによって、再びスタートコンディションを発生させることができます。スタートコンディションを発生させる条件として、次の 2 とおりがあります。

- バスが使用されていない状態 (MSS=0 * BB=0 * INT=0 * AL=0) での MSS ビットへの "1" 書込み
- バスマスタ時の割込み状態 (MSS=1 * BB=1 * INT=1 * AL=0) での SCC ビットへの "1" 書込み

ほかのシステム (アイドル状態中) がバス使用中に、MSS ビットへ "1" を書き込むと、AL ビットが "1" にセットされます。上記以外での MSS ビット、SCC ビットへの "1" の書込みは無視されます。

■ ストップコンディション

マスタモード (MSS=1) 時に、MSS ビットに "0" を書き込むと、ストップコンディションを発生し、スレーブモードになります。ストップコンディションを発生させる条件は、次のとおりです。

- バスマスタ時の割込み状態 (MSS=1 * BB=1 * INT=1 * AL=0) での MSS ビットへの "0" 書込み

これ以外での、MSS ビットへの "0" の書込みは無視されます。

■ アドレッシング

マスタモードでは、スタートコンディション発生後、BB=1, TRX=1 にセットされ、IDAR の内容を MSB から出力します。アドレスデータ送信後、スレーブからアクノリッジを受信すると、送信データの bit0(送信後の IDAR の bit0) を反転して、TRX ビットへ格納します。

スレーブモードでは、スタートコンディション発生後、BB=1, TRX=0 にセットされ、マスタからの送信データを IDAR へ受信します。アドレスデータ受信後、IDAR と IADR との比較が行われ、一致している場合、AAS=1 にセットしてマスタに対してアクノリッジを送信します。その後、受信データの bit0(受信後の IDAR の bit0) を TRX ビットへ格納します。

■ アービトレーション

マスタ送信時に、ほかのマスタも同時にデータを送信している場合、アービトレーションが起こります。自分の送信データが "1", SDA ライン上のデータが "L" レベルの場合、自分はアービトレーションを失ったと見なし、AL=1 にセットします。また、前述のようにバスが使用状態のときに、スタートコンディションを発生させようとした場合も AL=1 にセットされます。AL=1 にセットされると、MSS=0, TRX=0 となり、スレーブ受信モードとなります。

■ アクノリッジ

アクノリッジは、受信側が送信側に対して送信します。データ受信時は、ACK ビットによってアクノリッジの有無を選択できます。データ送信時は、受信側からのアクノリッジが LRB ビットに格納されます。

スレーブ送信側に、マスタ受信側からアクノリッジを受信しなかった場合、TRX=0 となりスレーブ受信モードになります。これによって、マスタはスレーブが SCL ラインを開放したときに、ストップコンディションを発生できます。

■ バスエラー

以下の条件が成立した場合はバスエラーと判断され、I²C インタフェースは停止状態になります。

- データ転送中 (ACK ビットを含む) の I²C バス上の基本規定違反の検出
- マスタ時のストップコンディション検出
- バスアイドルにおける I²C バス上の基本規定違反の検出

■ その他

● アービトレーションロスト検出後の処理

アービトレーションロストした後、自分がアドレッシングされている場合と、アドレッシングされていなかった場合をソフトウェアで判断しなければなりません。

アービトレーションロストするとハードウェア的にスレーブになりますが、1 バイト転送終了後、CLK ライン、DATA ライン双方を "L" に引っ張ります。このため、アドレッシングされていなければ、即座に CLK ライン、DATA ラインを解放し、アドレッシングされていた場合には、スレーブ送信、またはスレーブ受信の準備をしてから、CLK ライン、DATA ラインを開放する必要があります (すべてソフトウェアで処理する必要があります)。

● アービトレーションロスト検出時の割込み要因

アービトレーションロストを検出した場合、即時に割込み要因が立つのではなく、1 バイト転送終了した時点で割込み要因が立ちます。

アービトレーションロストを検出するとハードウェア的にスレーブになりますが、スレーブになっても、割込み要因を立てるために、トータル 9 クロック出力します。したがって、即時に割込み要因が立たないため、アービトレーションロスト後に処理を行うことはできません。

● 割込み条件

I²C バスには 1 つの割込みしかなく、1 バイト転送終了した時点で、または割込み条件に該当していた場合、割込み要因が立つ仕様になっています。

1 つの割込みで、複数の割込み条件を判断しなければならないため、割込みルーチン内で、各フラグをチェックしなければなりません。1 バイト転送終了時点での割込み条件を下記に示します。

- バスマスタであった場合
- アドレスされたスレーブであった場合
- ゼネラルコールアドレスを受信した場合
- アービトレーションロストが生じた場合

● 転送速度

I²C バスは、最大でシリアルクロック周波数 100 kHz の転送までしか行えませんので、注意してください。

22.3.1 I²C インタフェースの転送フロー

図 22.3-1 に、マスタからスレーブへの 1 バイト転送フローを、図 22.3-2 に、スレーブからマスタへの 1 バイト転送フローを示します。

■ I²C インタフェースの転送フロー

図 22.3-1 マスタからスレーブへの 1 バイト転送フロー

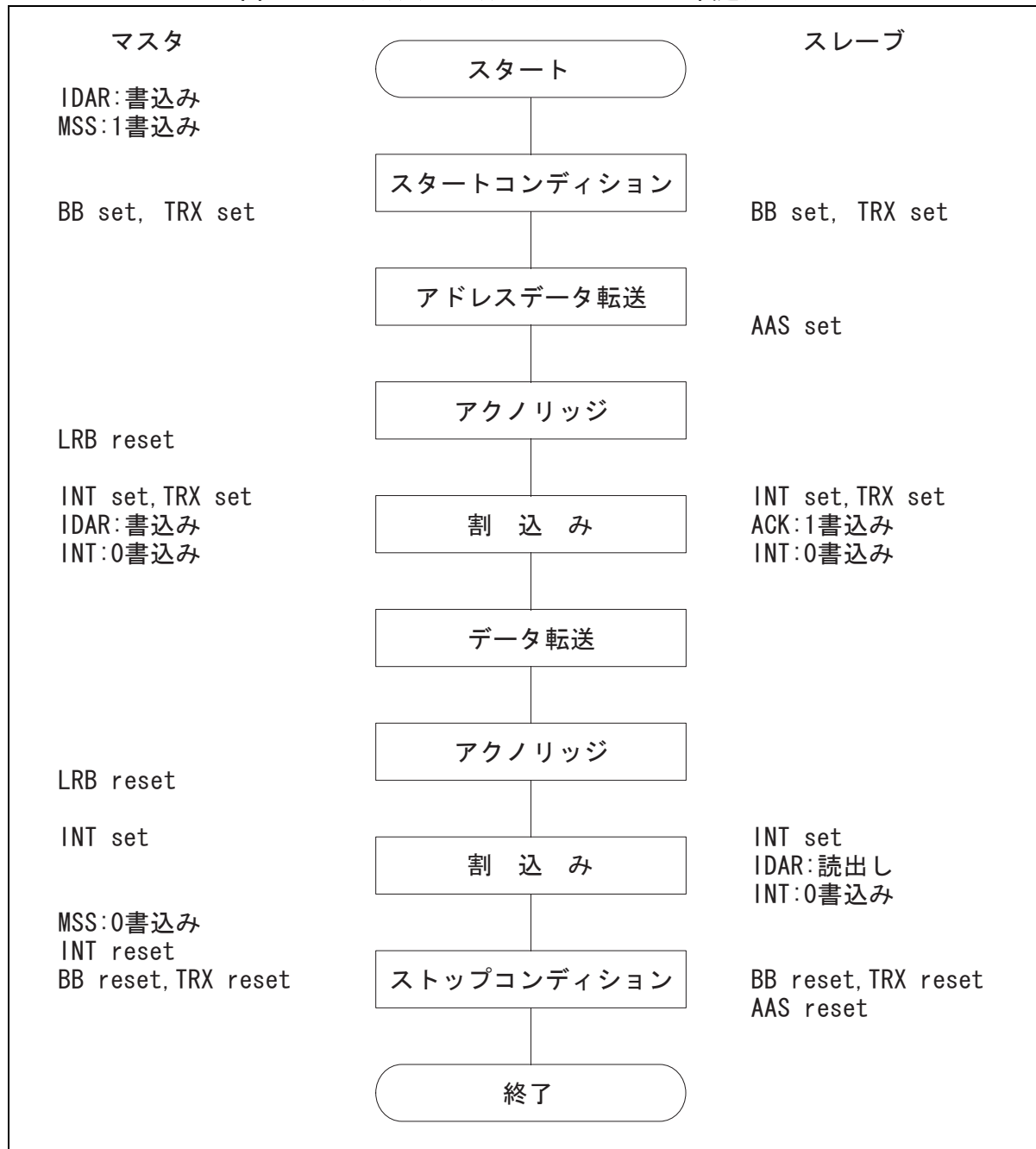
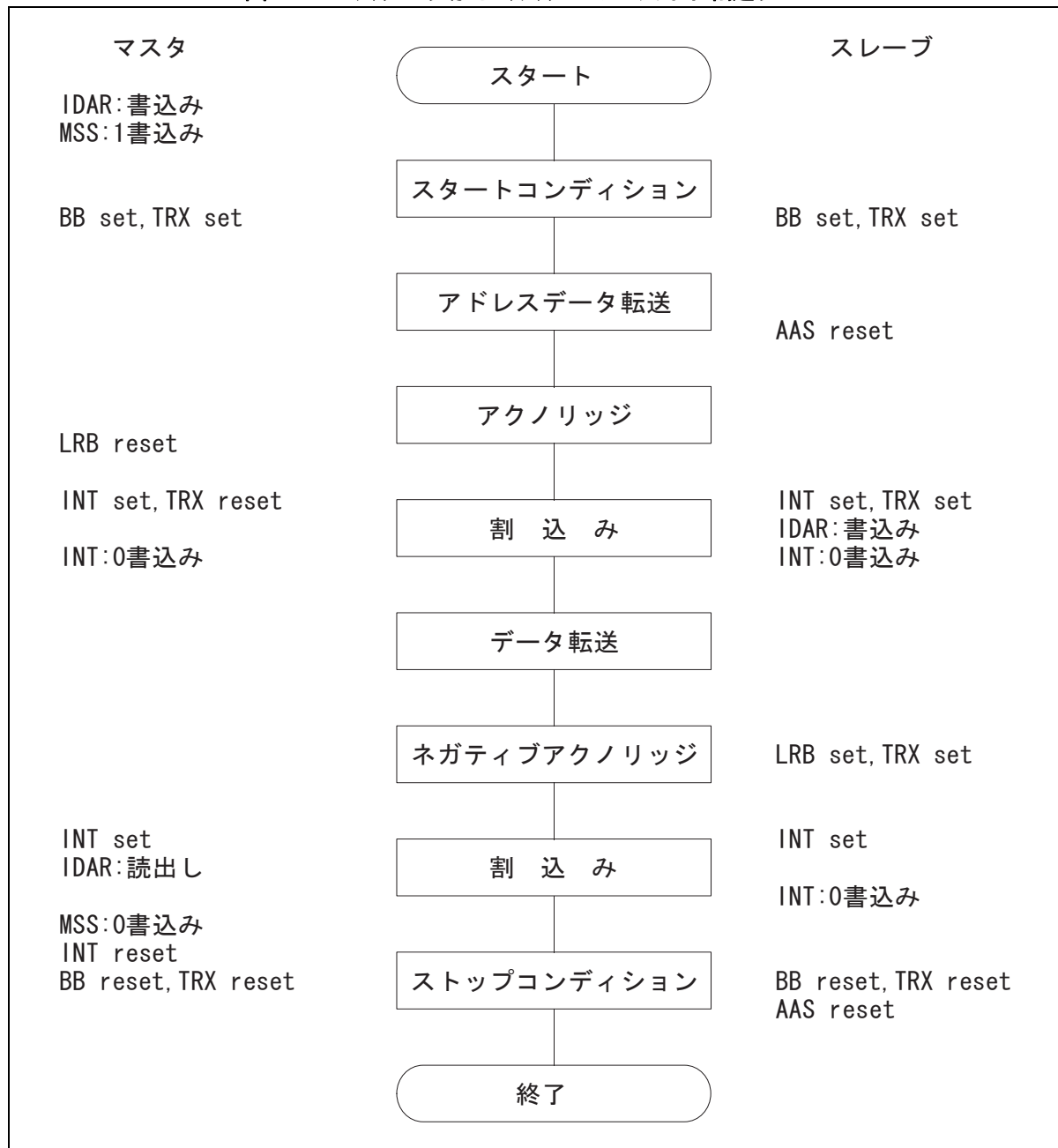


図 22.3-2 スレーブからマスタへの 1 バイト転送フロー

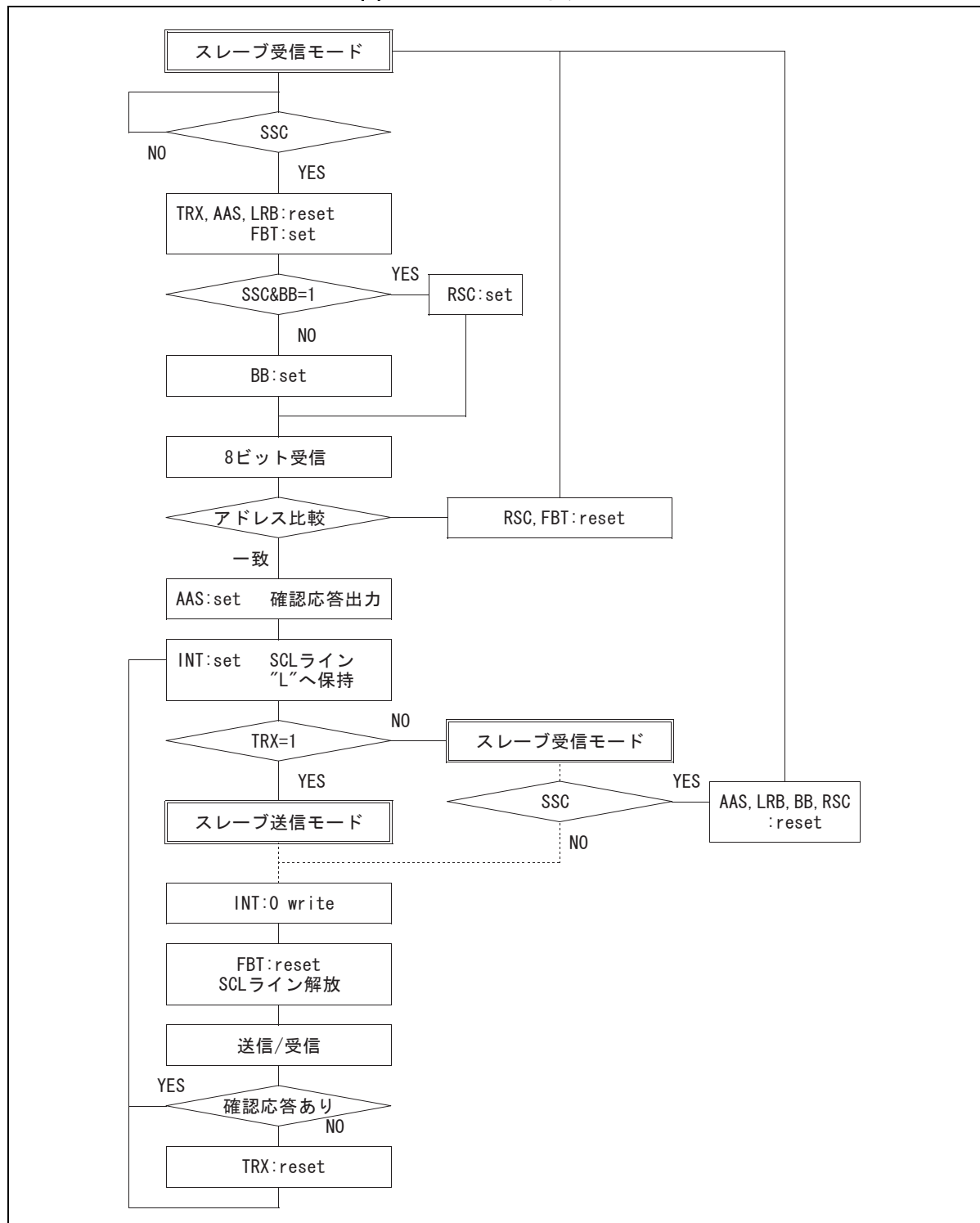


22.3.2 I²C インタフェースのモードフロー

図 22.3-3 に、I²C インタフェースのモードフローを示します。

■ I²C インタフェースのモードフロー

図 22.3-3 I²C モードフロー



22.3.3 I²C インタフェースの動作フロー

図 22.3-4 に I²C インタフェースのマスタ送受信プログラムフロー (割込み使用) を ,
また , 図 22.3-5 に I²C インタフェースのスレーブプログラムフロー (割込み使用) を
示します。

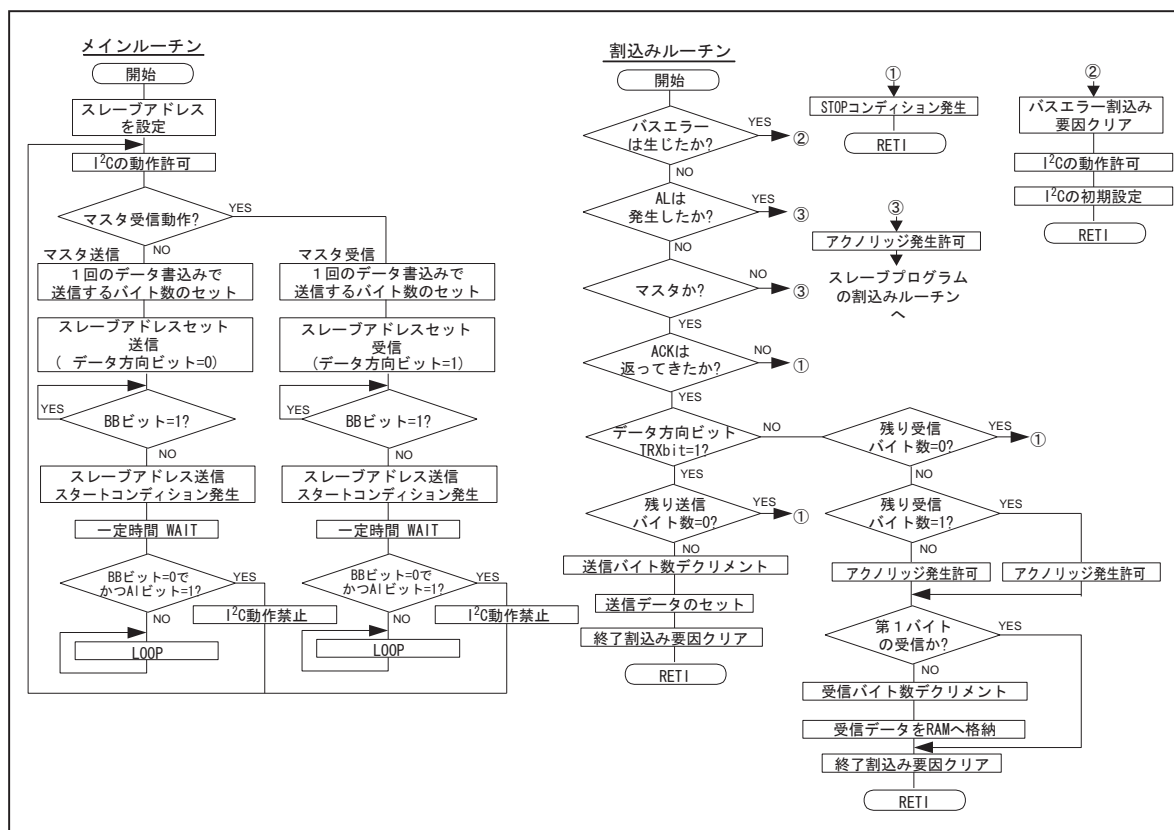
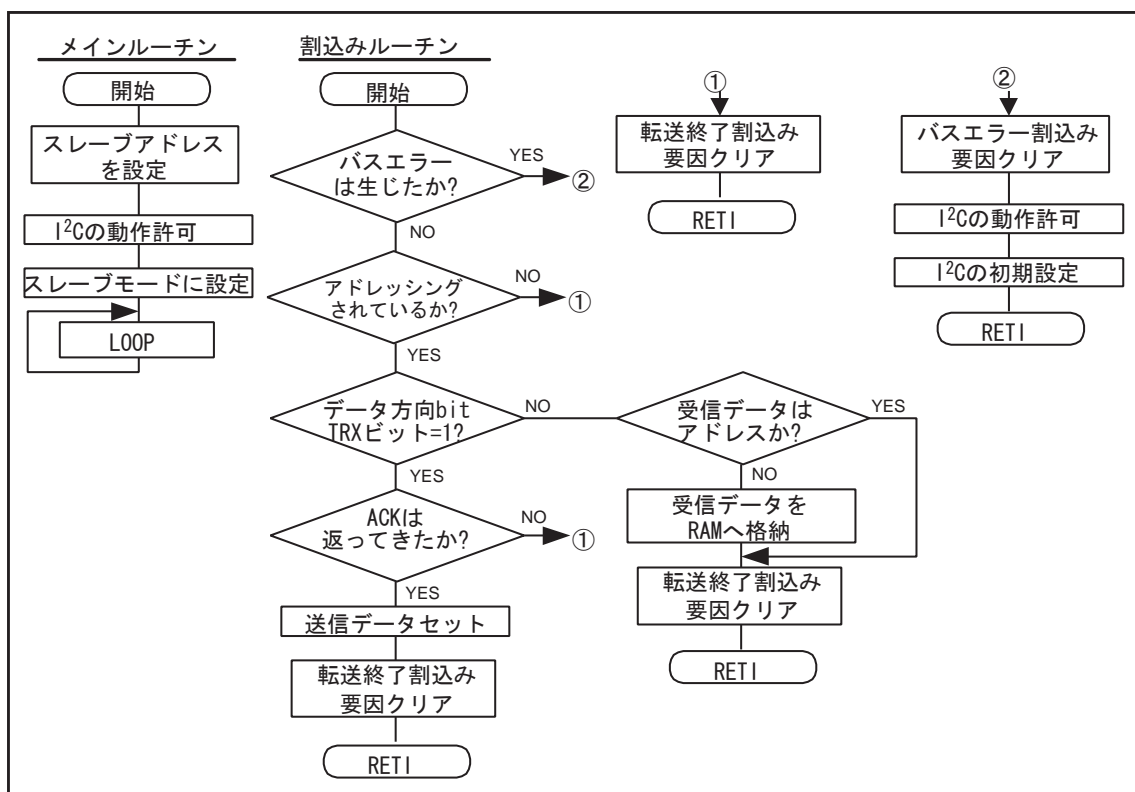
■ I²C インタフェースの動作フロー図 22.3-4 I²C マスタ送受信プログラムフローチャート (割込み使用)

図 22.3-5 I²C スレーブプログラムフローチャート (割込み使用)



第23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について説明します。

23.1 ROM ミラー機能選択モジュールの概要

23.2 ROM ミラー機能選択レジスタ (ROMM)

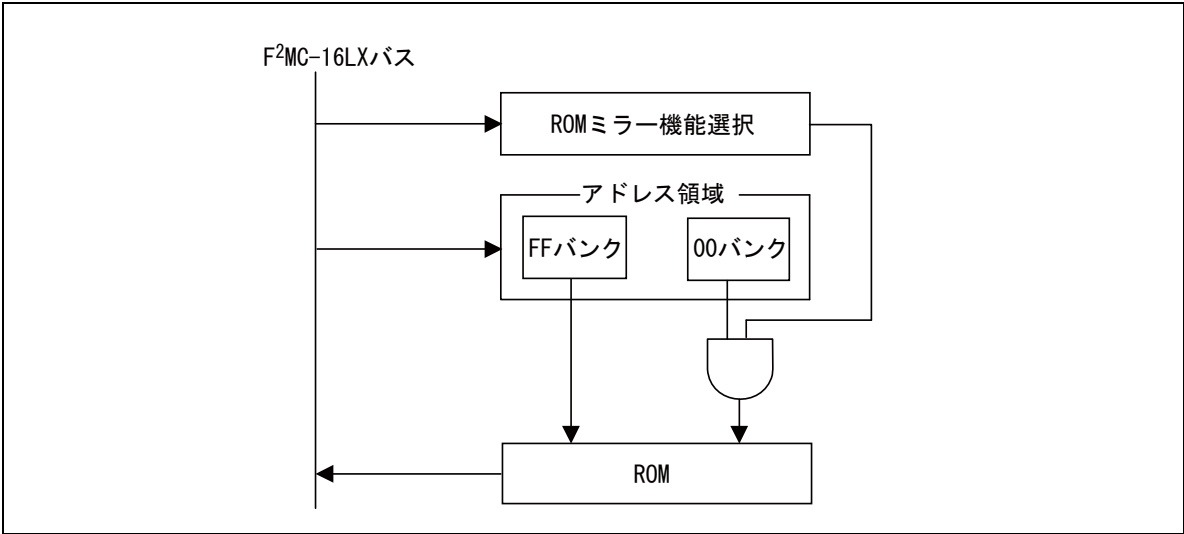
23.1 ROM ミラー機能選択モジュールの概要

ROM ミラー機能選択モジュールは、FF バンクに配置されている ROM 内のデータを、00 バンクへのアクセスで読み出せるように設定します。

■ ROM ミラー機能選択モジュールのブロックダイアグラム

図 23.1-1 に、ROM ミラー機能選択モジュールのブロックダイアグラムを示します。

図 23.1-1 ROM ミラー機能選択モジュールのブロックダイアグラム



■ ROM ミラー機能選択モジュールのレジスタ

図 23.1-2 に、ROM ミラー機能選択モジュールの構成を示します。

図 23.1-2 ROM ミラー機能選択モジュールの構成

	bit	15	14	13	12	11	10	9	8	初期値
ROMM アドレス : 00006F _H		-	-	-	-	-	-	予約 R/W	MI W	-----11 _B

23.2 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) の構成および機能について説明します。

■ ROMM(ROM ミラー機能選択レジスタ)

下図に、ROM ミラー機能選択レジスタ (ROMM) のビット構成を示します。

図 23.2-1 ROM ミラー機能選択レジスタ (ROMM) のビット構成

bit	15	14	13	12	11	10	9	8	初期値
ROMM アドレス : 00006F _H	-	-	-	-	-	-	予約 R/W	MI W	-----11 _B

[bit9] 予約ビット

予約ビットです。必ず "1" を書き込んでください。

[bit8] MI

- "1" の書込み時には、FF バンクの ROM のデータが 00 バンクでも読み出せるようになります。
- "0" の書込み時には 00 バンクでこの機能は働きません。
- 書込みのみ可能です。

< 注意事項 >

- ROMM レジスタは、アドレス 008000_H ~ 00FFFF_H 番地の動作中にアクセスしないでください。
- ROM ミラー機能を起動している場合、00 バンクの 008000_H ~ 00FFFF_H 番地へ FF8000_H ~ FFFFFFF_H 番地がミラーされますので、FF7FFF_H 番地以下の ROM アドレスでは ROM のミラー機能を設定しても 00 バンクにミラーされません。

第24章

アドレス一致検出機能

この章では、アドレス一致検出機能の機能と動作について説明します。

- 24.1 アドレス一致検出機能の概要
- 24.2 アドレス一致検出機能のブロックダイアグラム
- 24.3 アドレス一致検出機能のレジスタ構成
- 24.4 アドレス一致検出機能の動作説明
- 24.5 アドレス一致検出機能のプログラム例

24.1 アドレス一致検出機能の概要

アドレス一致検出機能は、プログラムが現在処理している命令の次に処理される命令のアドレスが、プログラムアドレス検出レジスタに設定したアドレスと一致した場合に、プログラムで次に処理される命令を強制的に INT9 命令に置き換え、割り込み処理プログラムへ分岐する機能です。INT9 割り込みを利用できるので、本機能はプログラムのパッチ処理による修正に利用できます。

■ アドレス一致検出機能の概要

- プログラムが現在処理している命令の次に処理される命令のアドレスは、内部データバスを通じて常にアドレスラッチに保持されます。アドレス一致検出機能の働きにより、アドレスラッチに保持されたアドレスの値と検出アドレス設定レジスタに設定したアドレスの値は常に比較されます。比較されたアドレスの値が一致した場合は、CPU が次に実行する命令が強制的に INT9 命令に置き換えられ、割り込み処理プログラムが実行されます。
- プログラムアドレス検出レジスタ (PADR0, PADR1) は 2 つあり、各レジスタに割り込み許可ビットが用意されています。アドレスラッチに保持されたアドレスと検出アドレス設定レジスタに設定したアドレスの一致による割り込みの発生を、レジスタごとに許可または禁止できます。

24.2 アドレス一致検出機能のブロックダイアグラム

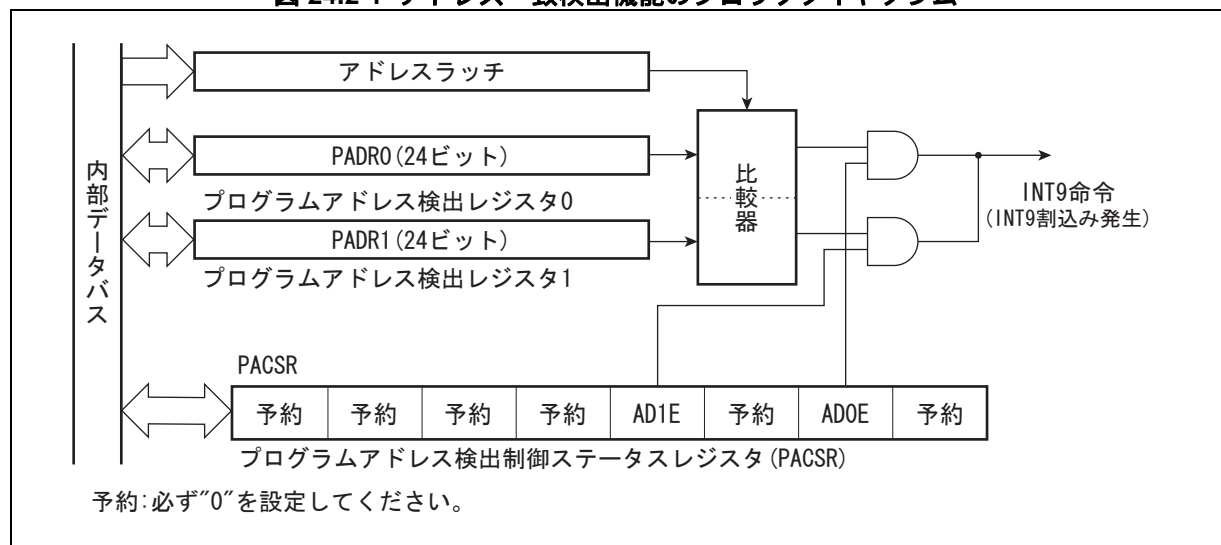
アドレス一致検出モジュールは、以下のブロックで構成されています。

- アドレスラッチ
- 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レジスタ (RADR0, RADR1)

■ アドレス一致検出機能のブロックダイアグラム

アドレス一致検出機能のブロックダイアグラムを図 24.2-1 に示します。

図 24.2-1 アドレス一致検出機能のブロックダイアグラム



● アドレスラッチ

内部データバスへ出力されたアドレス値を保持します。

● プログラムアドレス検出制御ステータスレジスタ (PACSR)

アドレスが一致した場合の割込み出力の許可または禁止を設定します。

● プログラムアドレス検出レジスタ (PADR0, PADR1)

アドレスラッチの値と比較するためのアドレスを設定します。

< 注意事項 >

プログラムアドレス検出レジスタは、1FF0_H ~ 1FF5_Hにレジスタをもっているため、RAM領域と重なりますので、本機能をご使用の場合は、RAMアクセスを行わないようにしてください。

24.3 アドレス一致検出機能のレジスタ構成

アドレス一致検出機能で使用するレジスタの一覧と詳細を記述します。

■ アドレス一致検出機能のレジスタと初期値の一覧

図 24.3-1 アドレス一致検出機能のレジスタと初期値の一覧

プログラムアドレス検出制御ステータスレジスタ (PACSR)							
bit	7	6	5	4	3	2	1 0
アドレス : 009E _H	0	0	0	0	0	0	0
プログラムアドレス検出レジスタ 0 (PADR0): 上位							
bit	7	6	5	4	3	2	1 0
アドレス : 1FF2 _H	X	X	X	X	X	X	X
プログラムアドレス検出レジスタ 0 (PADR0): 中位							
bit	15	14	13	12	11	10	9 8
アドレス : 1FF1 _H	X	X	X	X	X	X	X
プログラムアドレス検出レジスタ 0 (PADR0): 下位							
bit	7	6	5	4	3	2	1 0
アドレス : 1FF0 _H	X	X	X	X	X	X	X
プログラムアドレス検出レジスタ 1 (PADR1): 上位							
bit	7	6	5	4	3	2	1 0
アドレス : 1FF5 _H	X	X	X	X	X	X	X
プログラムアドレス検出レジスタ 1 (PADR1): 中位							
bit	15	14	13	12	11	10	9 8
アドレス : 1FF4 _H	X	X	X	X	X	X	X
プログラムアドレス検出レジスタ 1 (PADR1): 下位							
bit	7	6	5	4	3	2	1 0
アドレス : 1FF3 _H	X	X	X	X	X	X	X
X : 不定							

24.3.1 プログラムアドレス検出制御ステータスレジスタ (PACSR)

アドレス一致による割込み出力の許可または禁止を設定します。アドレス一致による割込み出力を許可した場合にアドレス一致が検出されると、INT9 の割込みが出力されます。

■ プログラムアドレス検出制御ステータスレジスタ (PACSR)

図 24.3-2 プログラムアドレス検出制御ステータスレジスタ (PACSR)

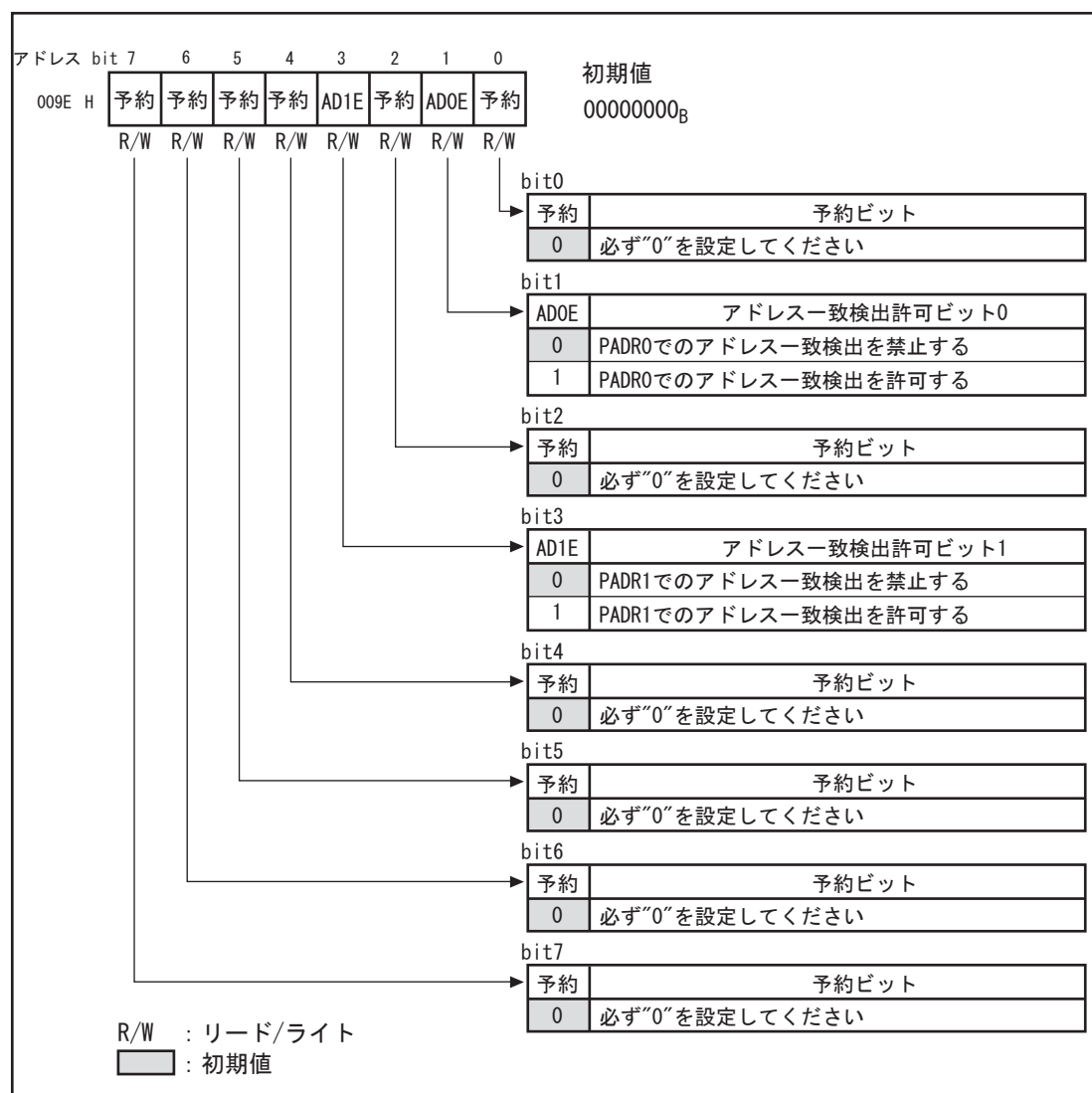


表 24.3-1 プログラムアドレス検出制御ステータスレジスタ (PACSR) の機能

ビット名		機能
bit7 ~ bit4	予約ビット	必ず "0" を設定してください。
bit3	AD1E: アドレス一致 検出許可ビット 1	<p>プログラムアドレス検出レジスタ 1 (PADR1) とのアドレス一致検出 動作を許可または禁止します。</p> <p>このビットに "0" を設定した場合：アドレス一致検出動作を 禁止します。</p> <p>このビットに "1" を設定した場合：アドレス一致検出動作を 許可します。</p> <p>アドレス一致検出動作を許可した場合に (AD1E=1), アドレ スラッチの値とプログラムアドレス検出レジスタ 1 (PADR1) の値が一致した場 合は, 直ちに INT9 命令が実行されます。</p>
bit2	予約ビット	必ず "0" を設定してください
bit1	AD0E: アドレス一致 検出許可ビット 0	<p>プログラムアドレス検出レジスタ 0 (PADR0) とのアドレス一致検出 動作を許可または禁止します。</p> <p>このビットに "0" を設定した場合：アドレス一致検出動作を 禁止します。</p> <p>このビットに "1" を設定した場合：アドレス一致検出動作を 許可します。</p> <p>アドレス一致検出動作を許可した場合に (AD0E=1), アドレ スラッチの値とプログラムアドレス検出レジスタ 0 (PADR0) の値が一致した場 合は, 直ちに INT9 命令が実行されます。</p>
bit0	予約ビット	必ず "0" を設定してください。

24.3.2 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には、検出するアドレス値を設定します。プログラムで実行されている命令の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アドレスと一致すると、次に実行される命令は強制的に INT9 命令に置き換えられ、割込み処理プログラムが実行されます。

■ プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L)

図 24.3-3 プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L)

PADR0, PADR1 : 上位								
bit	7	6	5	4	3	2	1	0
アドレス : 1FF2 _H 1FF5 _H	D23	D22	D21	D20	D19	D18	D17	D16
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
								初期値 XXXXXXXX _B
PADR0, PADR1 : 中位								
bit	15	14	13	12	11	10	9	8
アドレス : 1FF1 _H 1FF4 _H	D15	D14	D13	D12	D11	D10	D9	D8
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
								初期値 XXXXXXXX _B
PADR0, PADR1 : 下位								
bit	7	6	5	4	3	2	1	0
アドレス : 1FF0 _H 1FF3 _H	D7	D6	D5	D4	D3	D2	D1	D0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
								初期値 XXXXXXXX _B
R/W : リード・ライト可能								
X : 不定								

■ プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L)

プログラムアドレス検出レジスタは 2 本 (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L) 用意されており, それぞれ上位, 中位, 下位の 3 バイト, 合計 24 ビットで構成されています。

表 24.3-2 プログラムアドレス検出レジスタのアドレス設定

レジスタ名	割込み出力許可	アドレス設定	
プログラムアドレス検出レジスタ 0 (PADR0H, PADR0M, PADR0L)	PACSR:ADE0	上位	プログラムアドレス 0 の上位 8 ビットを設定 (バンク)
		中位	プログラムアドレス 0 の中位 8 ビットを設定
		下位	プログラムアドレス 0 の下位 8 ビットを設定
プログラムアドレス検出レジスタ 1 (PADR1H, PADR1M, PADR1L)	PACSR:ADE1	上位	プログラムアドレス 1 の上位 8 ビットを設定 (バンク)
		中位	プログラムアドレス 1 の中位 8 ビットを設定
		下位	プログラムアドレス 1 の下位 8 ビットを設定

プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L) には, INT9 命令と置き換える命令の先頭アドレス (1 バイト目) を設定する必要があります。

図 24.3-4 INT9 命令と置き換える命令コードの先頭アドレスの設定

アドレス	命令コード	検出アドレスに設定 (上位:FF _H , 中位:00 _H , 下位:1F _H)	
		ニーモニック	
FF001C :	A8 00 00	MOVW	RW0, #0000
FF001F :	4A 00 00	MOVW	A, #0000
FF0022 :	4A 80 08	MOVW	A, #0880

< 注意事項 >

プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L) に, 1 バイト目以外のアドレスを設定した場合は, 命令コードが INT9 命令に置き換えられず, 割込み処理プログラムは実行されません。また, 2 バイト目以降に設定した場合は, 命令コードで指定した番地が "01_H" (INT9 命令コード) に置き換わってしまい, 誤作動を起こす原因になる場合があるので注意してください。

プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L) は, 対応するプログラムアドレス一致制御レジスタのアドレス一致検出動作を禁止してから (PACSR レジスタの AD0E:bit1=0 または AD1E:bit1=0) 設定してください。アドレス一致検出動作を禁止せずにプログラムアドレス検出レジスタを変更した場合, アドレスの書き込み中にアドレスが一致すると直ちにアドレス一致検出機能が働いてしまい, 誤動作を起こす原因になります。

アドレス一致検出機能は, 内部 ROM のアドレスに対してのみ使用できます。外部メモリ領域のアドレスを設定してもアドレス一致検出機能は働かず, INT9 命令は実行されません。

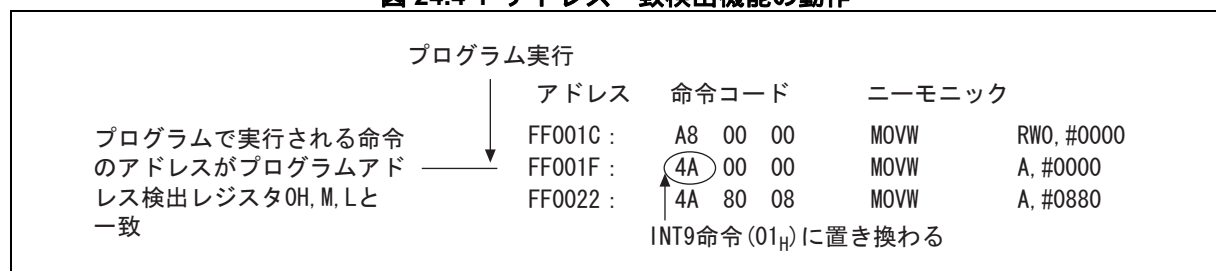
24.4 アドレス一致検出機能の動作説明

アドレス一致検出機能は、プログラムで実行される命令のアドレスが、プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L) で設定したアドレスと一致した場合に、CPU で実行される先頭の命令コードを INT9 (01_H) 命令に置き換え、割込み処理プログラムへ分岐します。

■ アドレス一致検出機能の動作

検出アドレスの設定とアドレス一致が検出された場合の動作を、図 24.4-1 に示します。

図 24.4-1 アドレス一致検出機能の動作



■ 検出アドレスの設定

検出アドレスを設定するプログラムアドレス検出レジスタ 0 (PADR0H, PADR0M, PADR0L) のアドレス一致検出動作を禁止してください (PACSR レジスタの AD0E:bit1=0)。

プログラムアドレス検出レジスタ 0 (PADR0H, PADR0M, PADR0L) に、検出するアドレスを設定します。プログラムアドレス検出レジスタ 0 (PADR0H, PADR0M, PADR0L) の上位に "FF_H", 中位に "00_H", 下位に "1F_H" を設定してください。

検出アドレスを設定するプログラムアドレス検出レジスタ 0 (PADR0H, PADR0M, PADR0L) のアドレス一致検出動作を許可してください (PACSR レジスタの AD0E:bit1=1)。

■ プログラムの実行

プログラムで実行される命令のアドレスが、設定した検出アドレスと一致すると、一致したアドレスの先頭の命令コードが INT9 命令コード ("01_H") に置き換わります。

INT9 命令が実行されます。INT9 割込みが発生し、割込み処理プログラムが実行されます。

表 25.5-2 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。

24.4.1 アドレス一致検出機能の使用例

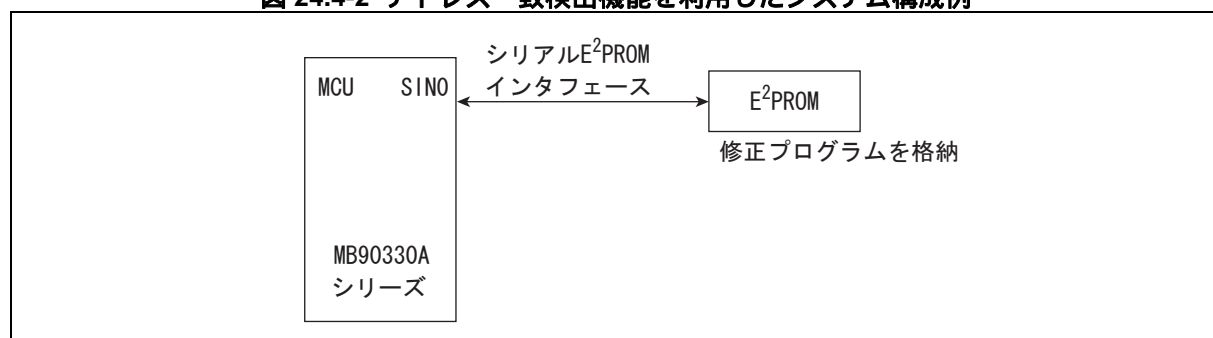
アドレス一致検出機能を利用した，プログラム修正のパッチ処理の例を示します。

■ システム構成と E²PROM のメモリ構成

● システム構成

アドレス一致検出機能を利用したシステム構成例を図 24.4-2 に示します。

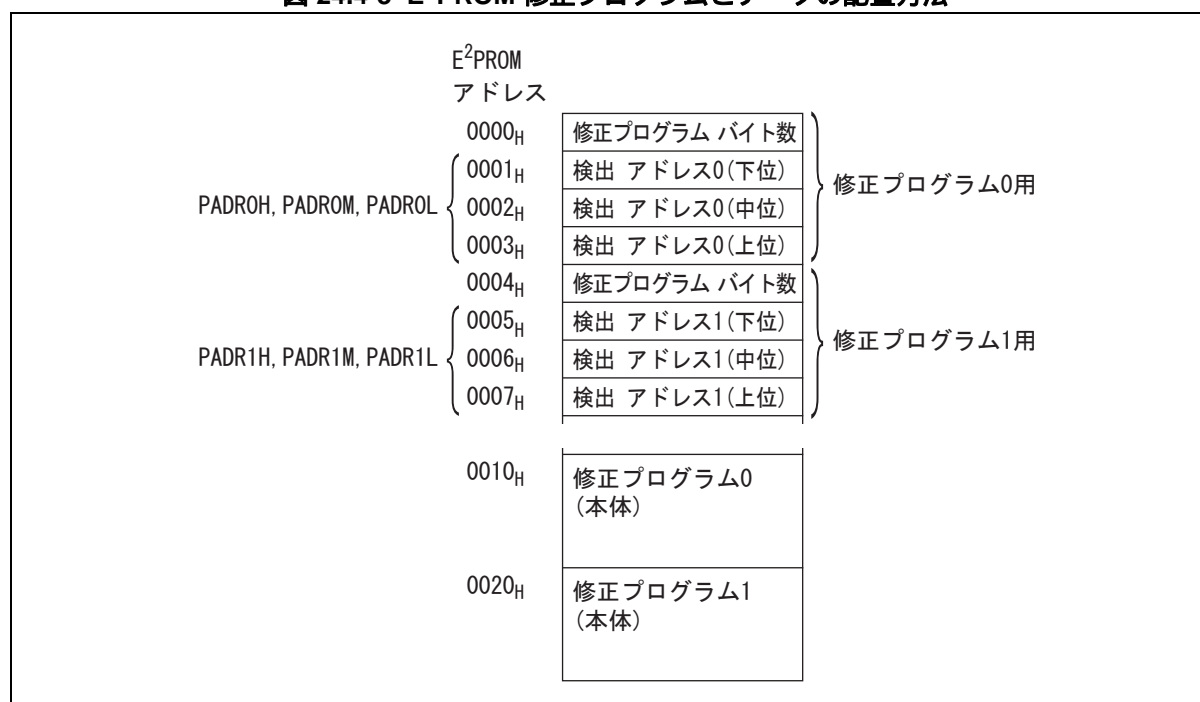
図 24.4-2 アドレス一致検出機能を利用したシステム構成例



■ E²PROM のメモリマップ

修正プログラムを E²PROM に格納する場合の修正プログラムとデータの配置方法を図 24.4-3 に示します。

図 24.4-3 E²PROM 修正プログラムとデータの配置方法



● 修正プログラムバイト数

修正プログラム (本体) の総バイト数を格納します。バイト数が "00_H" の場合は , " 修正プログラムは存在しない " ことを示します。

● 検出アドレス (24 ビット)

プログラムミスにより INT9 命令に置き換える箇所のアドレスを格納します。このアドレスはプログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L) に設定するアドレスです。

● 修正プログラム (本体)

プログラムのアドレスと検出アドレスが一致した場合の , INT9 割込みによって実行するプログラムを格納します。修正プログラム 0 は , 任意に決めたアドレスから配置します。修正プログラム 1 は , < 修正プログラム 0 の先頭アドレス + 修正プログラム 0 の総バイト数 > のアドレスから配置します。

■ 設定と動作状態

● 初期設定

E²PROM のデータはすべて "00_H" にクリアします。

● プログラムミスが発生した場合

コネクタ接続 (UART) を使用して , 外部から MCU (MB90330A シリーズ) へ , E²PROM 修正プログラムとデータの配置方法に従って , 修正プログラムの情報を送ります。

MCU (MB90330A シリーズ) 側では , 外部から受け取った修正プログラムの情報を E²PROM へ格納します。

● リセットシーケンス

リセット後 , MCU (MB90330A シリーズ) は , E²PROM 修正プログラムのバイト数を読み出して , 修正プログラムの有無を確認します。リセット後 , MCU (MB90330A シリーズ) は , E²PROM 修正プログラムのバイト数を読み出して , 修正プログラムの有無を確認します。

修正プログラムのバイト数が "00_H" でない場合は , 検出アドレス 0, 1 の上位 , 中位 , 下位を読み出してプログラムアドレス検出レジスタ 0, 1 (PADR0H, PADR0M, PADR0L, PADR1H, PADR1M, PADR1L) へ読み出したデータを設定します。また , 修正プログラムバイト数に従って , 修正プログラム (本体) を読み出して , MCU (MB90330A シリーズ) の RAM へ書き込みます。

修正プログラム (本体) は , アドレス一致検出機能によって INT9 割込み処理で実行するアドレスに配置します。

アドレス一致検出動作を許可します (PACSR レジスタの ADE0:bit1=1, AD1E:bit3=1)。

● INT9 割込み処理

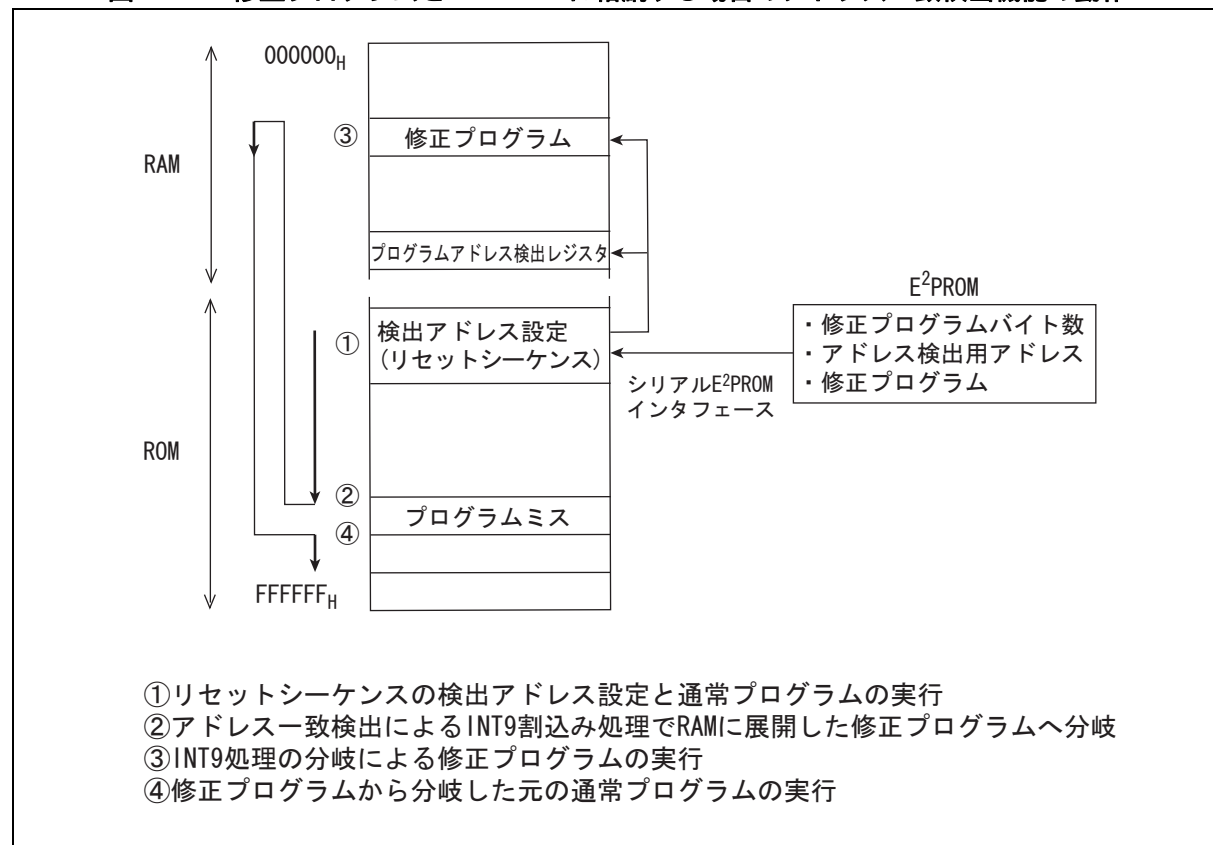
INT9 命令によって割込み処理が実行されます。MB90330A シリーズには , アドレス一致検出による割込み要求フラグがありません。したがって , プログラムカウンタのスタック情報が破棄されると検出アドレスを確認できません。検出アドレスを確認する場合は , 割込み処理ルーチンでスタックされているプログラムカウンタの値を確認してください。

修正プログラムの実行後に通常のプログラムに分岐します。

■ 修正プログラムを E²PROM に格納する場合のアドレス一致検出機能の動作

正プログラムを E²PROM に格納する場合のアドレス一致検出機能の動作を、図 24.4-4 に示します。

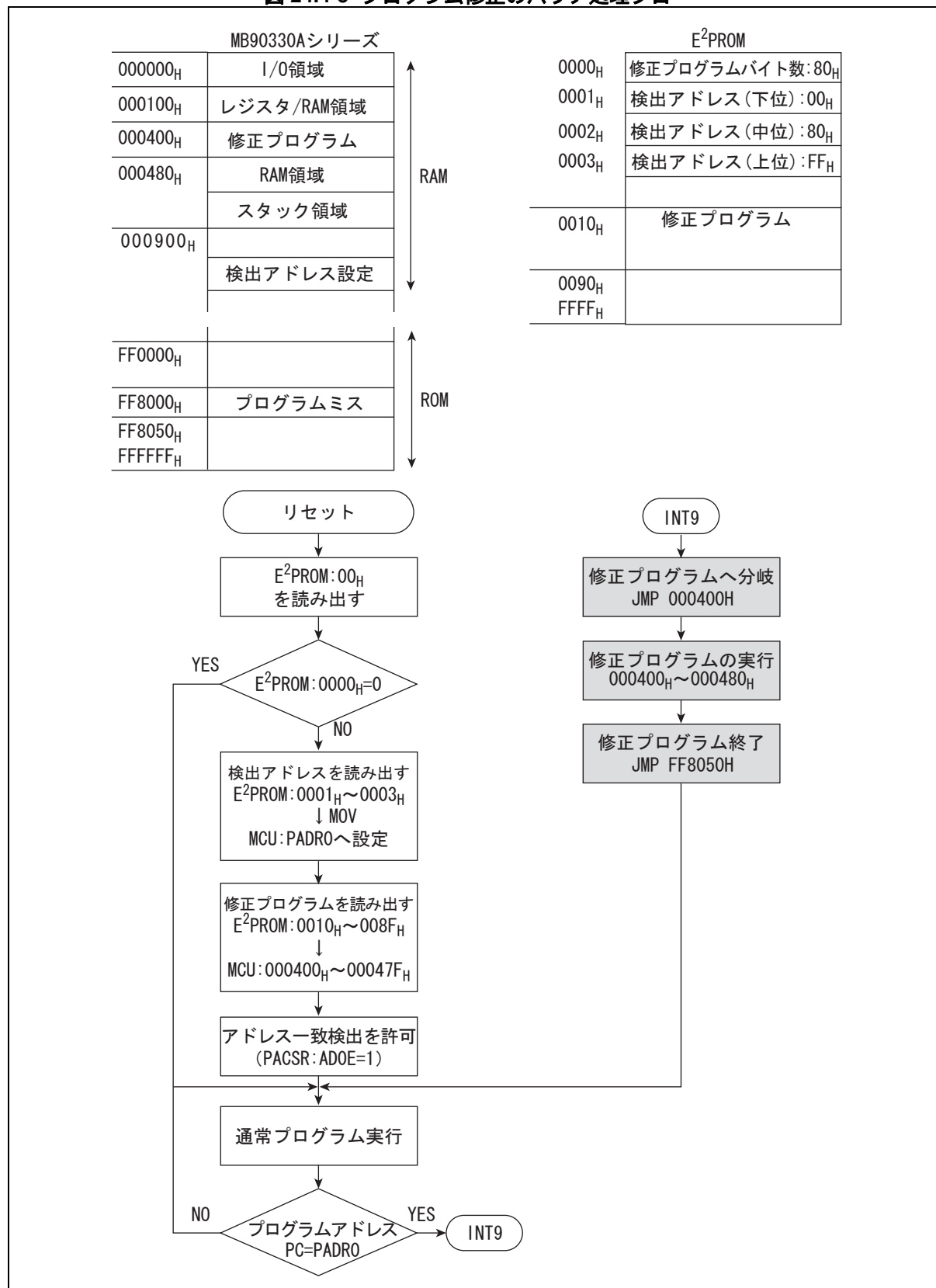
図 24.4-4 修正プログラムを E²PROM に格納する場合のアドレス一致検出機能の動作



■ プログラム修正のパッチ処理フロー

アドレス一致検出機能を利用したプログラム修正のパッチ処理のフローを図 24.4-5 に示します。

図 24.4-5 プログラム修正のパッチ処理フロー



24.5 アドレス一致検出機能のプログラム例

アドレス一致検出機能のプログラム例を示します。

■ アドレス一致検出機能のプログラム例

● 処理仕様

プログラムで実行される命令のアドレスと、プログラムアドレス検出レジスタ (PADR0H, PADR0M, PADR0L) に設定したアドレスが一致した場合に、INT9 命令が実行されます。

● コーディング例

```
PACSR EQU 00009EH          ; プログラムアドレス検出制御
                             ; ステータスレジスタ
PADRL EQU 000001H          ; プログラムアドレス検出レジスタ 0 下位
PADRM EQU 000002H          ; プログラムアドレス検出レジスタ 0 中位
PADRH EQU 000003H          ; プログラムアドレス検出レジスタ 0 上位
;
;----- メインプログラム -----
CODE CSEG
START:
                             ; スタックポインタ (SP) などは
                             ; 初期化済みとする
MOV PADRL, #00H             ; プログラムアドレス検出レジスタ 0 下位設定
MOV PADRM, #00H             ; プログラムアドレス検出レジスタ 0 中位設定
MOV PADRH, #00H             ; プログラムアドレス検出レジスタ 0 上位設定
;
MOV I:PACSR, #00000010B     ; アドレス一致検出の許可
.
ユーザ処理
.
LOOP:
.
ユーザ処理
.
BRA LOOP
;----- 割込みプログラム -----
WARI:
.
ユーザ処理
```

```

      .
      RETI                                ; 割込み処理からの復帰
CODE  ENDS
;----- ベクタ設定 -----
VECT  CSEG ABS=0FFH
      ORG 00FFDCH                        ; リセットベクタ設定
      DSL START
      DB 00H                            ; シングルチップモードに設定
VECT  ENDS
      END START
```


第25章

フラッシュメモリ

この章では、3M/4M ビットフラッシュメモリの機能や動作について説明します。

フラッシュメモリへのデータの書込み / 消去の方法には、下記の3とおりがあります。

- プログラム実行による書込み / 消去
- シリアルライターによる書込み
- フラッシュメモリライター

ここでは、「プログラム実行による書込み / 消去」について説明します。

25.1 フラッシュメモリの概要

25.2 フラッシュメモリのセクタ構成

25.3 フラッシュメモリコントロールステータスレジスタ (FMCS)

25.4 フラッシュメモリの自動アルゴリズム起動方法

25.5 自動アルゴリズム実行状態の確認

25.6 フラッシュメモリの書込み / 消去

25.1 フラッシュメモリの概要

フラッシュメモリは、CPU メモリマップ上の F9 ~ FF バンクに配置されています。フラッシュメモリインタフェース回路の機能によって、マスク ROM と同様に CPU からの読出しアクセスおよびプログラムアクセスが可能です。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 からの命令動作でできます。このため、内蔵 CPU の制御による実装状態での書き換えが可能となり、プログラム、およびデータの改善が効率よくできます。

■ フラッシュメモリの特長

フラッシュメモリは、以下の特長をもっています。

- 自動プログラムアルゴリズム (Embedded Alogrithm: MBM29LV400TC 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 / 消去の完了検出
- CPU 割込みによる書込み / 消去の完了検出
- セクタごとの消去が可能 (セクタの組合せ自由)
- 書込み / 消去回数 (最小) 10000 回
- セクタプロテクト機能 (推奨パラレルライタにて設定可能)

■ フラッシュメモリの容量と品種

フラッシュメモリは品種により、3M ビットフラッシュメモリ、4M ビットフラッシュメモリのどちらかが搭載されています。

● 3M ビットフラッシュメモリ

- 搭載品種：MB90F334A
- 容量：384 K バイト / 192 K ワード
- セクタ構成：64 K × 5 + 32 K + 8 K × 2 + 16 K

● 4M ビットフラッシュメモリ

- 搭載品種：MB90F335A
- 容量：512 K バイト / 256 K ワード
- セクタ構成：64 K × 6 + 32 K × 2 + 8 K × 4 + 16 K × 2

■ フラッシュメモリの書込み / 消去の方法

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。つまり、フラッシュメモリにデータ書込み / 消去の動作をさせる際には、フラッシュメモリ上にあるプログラムを RAM に一度コピーし、RAM 実行することによって、フラッシュメモリからプログラムアクセスをせずに書込み動作だけを行うことが可能となります。

25.2 フラッシュメモリのセクタ構成

フラッシュメモリのセクタ構成を示します。

■ セクタ構成

図 25.2-1 に、3M ビットフラッシュメモリのセクタ構成を、図 25.2-2 に4M ビットフラッシュメモリのセクタ構成を示します。図中のアドレスには、各セクタの上位アドレスと下位アドレスを示します。

図 25.2-1 3M 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
SA8 (16Kバイト)	FFFFFF _H	7FFFF _H
	FFC000 _H	7C000 _H
SA7 (8Kバイト)	FFBFFF _H	7BFFF _H
	FFA000 _H	7A000 _H
SA6 (8Kバイト)	FF9FFF _H	79FFF _H
	FF8000 _H	78000 _H
SA5 (32Kバイト)	FF7FFF _H	77FFF _H
	FF0000 _H	70000 _H
SA4 (64Kバイト)	FEFFFF _H	6FFFF _H
	FE0000 _H	60000 _H
SA3 (64Kバイト)	FDFFFF _H	5FFFF _H
	FD0000 _H	50000 _H
使用禁止	FCFFFF _H	4FFFF _H
	FC0000 _H	40000 _H
SA2 (64Kバイト)	FBFFFF _H	3FFFF _H
	FB0000 _H	30000 _H
SA1 (64Kバイト)	FAFFFF _H	2FFFF _H
	FA0000 _H	20000 _H
SA0 (64Kバイト)	F9FFFF _H	1FFFF _H
	F90000 _H	10000 _H
使用禁止	F8FFFF _H	0FFFF _H
	F80000 _H	00000 _H

CPUからアクセスする場合、F9バンクレジスタにSA0、FAバンクレジスタにSA1、FBバンクレジスタにSA2、FDバンクレジスタにSA3、FEバンクレジスタにSA4、FFバンクレジスタにSA5～SA8が配置されています。

図 25.2-2 4M 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
SA13 (16Kバイト)	FFFFFF _H	7FFFF _H
	FFC000 _H	7C000 _H
SA12 (8Kバイト)	FFBFFF _H	7BFFF _H
	FFA000 _H	7A000 _H
SA11 (8Kバイト)	FF9FFF _H	79FFF _H
	FF8000 _H	78000 _H
SA10 (32Kバイト)	FF7FFF _H	77FFF _H
	FF0000 _H	70000 _H
SA9 (64Kバイト)	FEFFFF _H	6FFFF _H
	FE0000 _H	60000 _H
SA8 (64Kバイト)	FDFFFF _H	5FFFF _H
	FD0000 _H	50000 _H
SA7 (64Kバイト)	FCFFFF _H	4FFFF _H
	FC0000 _H	40000 _H
SA6 (16Kバイト)	FBFFFF _H	3FFFF _H
	FBC000 _H	3C000 _H
SA5 (8Kバイト)	FBBFFF _H	3BFFF _H
	FBA000 _H	3A000 _H
SA4 (8Kバイト)	FB9FFF _H	39FFF _H
	FB8000 _H	38000 _H
SA3 (32Kバイト)	FB7FFF _H	37FFF _H
	FB0000 _H	30000 _H
SA2 (64Kバイト)	FAFFFF _H	2FFFF _H
	FA0000 _H	20000 _H
SA1 (64Kバイト)	F9FFF _H	1FFFF _H
	F90000 _H	10000 _H
SA0 (64Kバイト)	F8FFF _H	0FFFF _H
	F80000 _H	00000 _H

CPUからアクセスする場合、F8バンクレジスタにSA0、F9バンクレジスタにSA1、FAバンクレジスタにSA2、FBバンクレジスタにSA3～SA6、FCバンクレジスタにSA7、FDバンクレジスタにSA8、FE バンクレジスタにSA9、FFバンクレジスタにSA10 ～ SA13が配置されています。

*: ライターアドレス

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を書き込む際、CPUアドレスに対して相当するアドレスです。汎用ライターを使用して書込み / 消去を行う際には、このアドレスで書込み / 消去を実行します。

25.3 フラッシュメモリコントロールステータスレジスタ (FMCS)

フラッシュメモリコントロールステータスレジスタ (FMCS) は、フラッシュメモリインタフェース回路にあるレジスタで、フラッシュメモリの書込み / 消去の際に使用します。

■ フラッシュメモリコントロールステータスレジスタ (FMCS)

図 25.3-1 に、コントロールステータスレジスタ (FMCS) のビット構成を示します。

図 25.3-1 フラッシュメモリコントロールステータスレジスタ (FMCS) のビット構成

FMCS								
	bit 7	6	5	4	3	2	1	0
初期値								000X0000 _B
アドレス : 0000AE _H	INTE	RDYINT	WE	RDY	予約	予約	予約	予約
リード / ライト	R/W	R/W	R/W	R	W	W	W	W

以下に、コントロールステータスレジスタ (FMCS) の各ビットの機能を説明します。

[bit7] INTE

フラッシュメモリの書込み / 消去の終了によって、CPU に割り込みを発生させます。

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、割り込みは発生しません。

0	書込み / 消去終了での割り込み禁止
1	書込み / 消去終了での割り込み許可

[bit6] RDYINT

フラッシュメモリの動作状態を表すビットです。

フラッシュメモリの書込み / 消去が終わって "1" となります。フラッシュメモリ書込み / 消去後、RDYINT 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" になった後は、フラッシュメモリへの書込み / 消去が可能になります。

"0" の書込みによって "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 (「25.4 フラッシュメモリの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) 命令使用時は、必ず "1" が読み出されます。

0	書込み / 消去動作実行中
1	書込み / 消去動作終了 (割り込み要求発生)

[bit5] WE

フラッシュメモリ領域への書込み許可ビットです。

"1" のとき , F9 ~ FB, FD ~ FF バンクへのコマンドシーケンス (「 25.4 フラッシュメモリの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。"0" のとき , 書込み / 消去の信号は発生されません。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。

書込み / 消去を行わないときは , 誤ってフラッシュメモリにデータを書き込まないように , 常に "0" に設定することを推奨します。

0	フラッシュメモリ書込み / 消去禁止
1	フラッシュメモリ書込み / 消去許可

[bit4] RDY

フラッシュメモリの書込み / 消去許可ビットです。

"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。なお , この状態でも読み出し / 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られます。

0	書込み / 消去動作実行中
1	書込み / 消去動作終了 (次データ書込み / 消去許可)

[bit3 ~ bit0] 予約ビット

予約ビットです。必ず "0" に設定してください。

< 注意事項 >

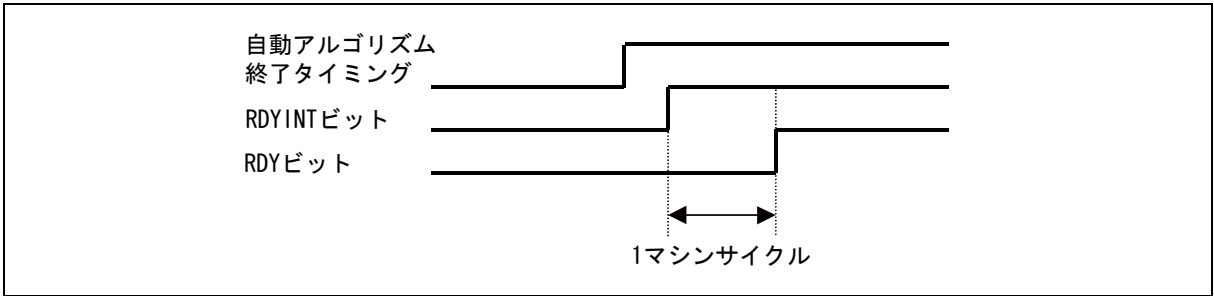
FMCS はバイトアクセスのみ可能です。

■ 自動アルゴリズム終了タイミング

図 25.3-2 に , 自動アルゴリズム終了タイミングと RDYINT ビットおよび RDY ビットの関係を示します。

RDYINT ビットと RDY ビットは同時には変化しません。どちらかのビットで判定するようにプログラムを作成してください。

図 25.3-2 自動アルゴリズム終了タイミングと RDYINT ビットおよび RDY ビットの関係



25.4 フラッシュメモリの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、セクタ消去、チップ消去の 4 種類があります。セクタ消去については、一時停止と再開の制御が可能です。

■ コマンドシーケンス表

表 25.4-1 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書き込むデータはすべてバイトですが、ワードアクセスで書き込むようにしてください。このときの上位バイト分のデータは無視されます。

表 25.4-1 コマンドシーケンス表

コマンド シーケンス	バス書込み サイクル	1st バス書込み サイクル		2nd バス書込み サイクル		3rd バス書込み サイクル		4th バス書込み サイクル		5th バス書込み サイクル		6th バス書込み 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	FxXXXX	XXF0	-	-			-	-	-	-	-	-
読出し / リセット *	4	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XXF0	RA	RD	-	-	-	-
書込み	4	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XXA0	PA (even)	PD (word)	-	-	-	-
チップ消去	6	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XX80	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XX10
セクタ消去	6	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XX80	FxAAAA	XXAA	Fx5554	XX55	SA (even)	XX30
セクタ消去一時停止		Address "FxXXXX" データ (xxB0 _H) の入力で、セクタ消去中の消去一時停止											
セクタ消去再開		Address "FxXXXX" データ (xx30 _H) の入力で、セクタ消去一時停止後、消去再開											
AutoSelect	3	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XX90	-	-	-	-	-	-

RA: 読出しアドレス

PA: 書込みアドレス、偶数アドレスのみ指定可能

SA: セクタアドレス (「25.2 フラッシュメモリのセクタ構成」を参照してください。)

RD: 読出しデータ

PD: 書込みデータ、ワードデータのみ指定可能

*: 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できます。

< 注意事項 >

- 表中のアドレス Fx は、FF, FE, FD, FB, FA, F9 を意味します。それぞれの操作時には、アクセス対象バンクの値としてください。
- 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 進数表記しています。ただし、"X" は任意の値です。

表 25.4-1 における AutoSelect は、セクタ保護の状態を知るためのコマンドです。
実際には、前記コマンドと共に下記のようにアドレスを設定する必要があります。

表 25.4-2 AutoSelect 時のアドレス設定

	AQ13 ~ AQ17	AQ7	AQ2	AQ1	AQ0	DQ7 ~ DQ0
セクタ保護	セクタアドレス	L	H	L	L	CODE *

*: 保護されたセクタアドレスでの出力は "01_H", 保護されていないセクタアドレスでの出力は "00_H"

25.5 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が完了したこと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。これによって、書込み / チップセクタ消去終了、消去コード書込みが有効かの確認ができます。

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 (表 25.4-1 を参照) 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読出しアクセスすることで参照できます。表 25.5-1 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 25.5-1 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であるかの判断は、ハードウェアシーケンスフラグを確認するか、フラッシュメモリコントロールレジスタ (FMCS) の RDY ビットを確認することで、書込みが終了しているかを知ることができます。書込み / 消去終了後は、読出し / リセット状態に戻ります。実際にプログラムを作成する際には、いずれかのフラグで自動書込み / 消去終了を確認後に、データの読出しなどの次処理を行ってください。また、2 回目以降のセクタ消去コード書込みが有効であるかも、ハードウェアシーケンスフラグによって確認できます。次項に各ハードウェアシーケンスフラグについて説明します。

表 25.5-2 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 25.5-2 ハードウェアシーケンスフラグの機能一覧

状態		DQ7	DQ6	DQ5	DQ3
正常動作時の 状態変化	書込み動作 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
	チップセクタ消去動作 消去完了	0 1	Toggle Stop	1	0 1
	セクタ消去ウェイト 消去開始	0	Toggle	0 1	0
	消去動作 セクタ消去一時停止 (消去中のセクタ)	0 1	Toggle 1	1 0	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 1	0
	セクタ消去一時停止中 (消去中でないセクタ)	DATA:7	DATA:6	DATA:5	DATA:3
異常動作	書込み動作	$\overline{\text{DQ7}}$	Toggle	0	1
	チップセクタ消去動作	0	Toggle	1	1

25.5.1 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また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ための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 (DQ7) の状態遷移

表 25.5-3 と表 25.5-4 に、データポーリングフラグの状態遷移を示します。

表 25.5-3 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	書き込み動作 完了	チップセクタ 消去 完了	セクタ消去 ウェイト 開 始	セクタ消去 消去一時停止 消去中のセク タ	セクタ消去一 時停止 再開 消去中のセク タ	セクタ消去 一時停止中 消去中でない セクタ
DQ7	$\overline{\text{DQ7}}$ DATA:7	0 1	0	0 1	1 0	DATA:7

表 25.5-4 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	書き込み動作	チップセクタ 消去動作
DQ7	$\overline{\text{DQ7}}$	0

■ 書き込み動作時

自動書き込みアルゴリズム実行中に読出しアクセスすると、フラッシュメモリはアドレスの指し示す番地によらず、最後に関与したデータの bit7 の反転データを出力します。自動書き込みアルゴリズム終了時に読出しアクセスすると、フラッシュメモリはアドレスの指し示す番地の読出し値の bit7 を出力します。

■ チップ / セクタ消去動作時

セクタ消去の自動アルゴリズム実行中に、現在消去しているセクタから読出しアクセスすると、フラッシュメモリは "0" を出力します。チップ消去の自動アルゴリズム実行中は、アドレスの指し示す番地に関係なく、読出しアクセスすると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時

セクタ消去一時停止時に読出しアクセスすると、フラッシュメモリはアドレスの指し示す番地が消去中のセクタであれば "1" を出力し、消去中のセクタでなければアドレスの指し示す番地の読出し値の 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、現在セクタ一時停止状態であるか、どのセクタが消去中であるかの判定ができます。

< 注意事項 >

自動アルゴリズム起動時は、指定したアドレスへの読出しアクセスは無視されます。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。このため、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読出しアクセスの次に行うようにしてください。

25.5.2 トグルビットフラグ (DQ6)

トグルビットフラグ (DQ6) は、データポーリングフラグ (DQ7) と同様に、主に自動アルゴリズム実行が進行中また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。

■ トグルビットフラグ (DQ6) の状態遷移

表 25.5-5 と表 25.5-6 に、トグルビットフラグの状態遷移を示します。

表 25.5-5 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	書き込み動作 完了	チップセクタ 消去 完了	セクタ消去 ウェイト 開 始	セクタ消去 消去一時停止 消去中のセク タ	セクタ消去一 時停止 再開 消去中のセク タ	セクタ消去 一時停止中 消去中でない セクタ
DQ6	Toggle DATA:6	Toggle Stop	Toggle	Toggle 1	1 Toggle	DATA:6

表 25.5-6 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	書き込み動作	チップセクタ 消去動作
DQ6	Toggle	Toggle

■ 書き込み / チップセクタ消去時

自動書き込みアルゴリズムおよびチップセクタ消去の自動アルゴリズム実行中に、連続した読出しアクセスを行うと、フラッシュメモリはアドレスの指し示す番地によらず、読出しごとに "1" と "0" を交互に出力するトグル状態を出力します。自動書き込みアルゴリズムおよびチップ / セクタ消去の自動アルゴリズム終了時に連続した読出しアクセスを行うと、フラッシュメモリは bit6 のトグル動作を止め、アドレスの指し示す番地の読出し値の bit6(DATA:6) を出力します。

■ セクタ消去一時停止時

セクタ消去一時停止時に読出しアクセスすると、フラッシュメモリはアドレスの指し示す番地が消去中のセクタに属するならば、"1" を出力します。消去中のセクタに属しないのであれば、アドレスの指し示す番地の読出し値の bit6(DATA:6) を出力します。

< 参考 >

書き込みの際、書き込もうとしているセクタが書換え保護されているセクタの場合は、約 2 μ s のトグル動作をした後、データを書き換えることなくトグル動作を終わります。
消去の際、選択されたすべてのセクタが書換え保護されている場合、トグルビットは約 100 μ s のトグル動作をし、その後データを書き換えなくて読出し/リセット状態に戻ります。

25.5.3 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された時間 (内部パルス回数) 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 (DQ5) の状態遷移

表 25.5-7 と表 25.5-8 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 25.5-7 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	書込み動作 完了	チップセクタ 消去 完了	セクタ消去 ウェイト 開 始	セクタ消去 消去一時停止 消去中のセク タ	セクタ消去一 時停止 再開 消去中のセク タ	セクタ消去 一時停止中 消去中でない セクタ
DQ5	0 DATA:5	0 1	0	0	0	DATA:5

表 25.5-8 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	書込み動作	チップセクタ 消去動作
DQ5	1	1

■ 書込み / チップセクタ消去時

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読み出しアクセスすると、規定時間 (書込み / 消去に要する時間) 内であれば "0" を、規定時間を超えてしまっている場合は "1" を出力します。これは、自動アルゴリズムが実行中か終了状態にあるかに無関係ですので、書込み / 消去が成功したか失敗したかの判定が可能です。すなわち、このフラグが "1" を出力したとき、データポーリング機能またはトグルビット機能によって自動アルゴリズムがまだ実行中であれば、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に "1" を書き込もうとすると、フェイルが発生します。この場合、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しません。まれに "1" が書き込めたように、正常終了する場合があります。したがって、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) はトグル動作を止めず、タイムリミットを超えて、タイミングリミット超過フラグ (DQ5) は "1" を出力します。この状態は、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ということを表しています。この状態が発生したときは、リセットコマンドを実行してください。

25.5.4 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 (DQ3) の状態遷移

表 25.5-9 と表 25.5-10 に、セクタ消去タイマフラグの状態遷移を示します。

表 25.5-9 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	書込み動作 完了	チップセクタ 消去 完了	セクタ消去 ウェイト 開始	セクタ消去 消去一時停止 消去中のセク タ	セクタ消去一 時停止 再開 消去中のセク タ	セクタ消去 一時停止中 消去中でない セクタ
DQ3	0 DATA:3	1	0 1	1 0	0 1	DATA:3

表 25.5-10 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	書込み動作	チップセクタ 消去動作
DQ3	0	1

■ セクタ消去動作時

セクタ消去コマンド起動後に読出しアクセスすると、フラッシュメモリはコマンドを発行したセクタのアドレス信号の指し示す番地によらずに、セクタ消去ウェイト期間中であれば "0" を、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って消去アルゴリズムが実行中を示している場合、このフラグが "1" であれば、内部で制御される消去が始まっています。続けてのセクタ消去コードの書込みまたは消去一時停止以外のコマンドは、消去が終了されるまで無視されます。

このフラグが "0" であれば、フラッシュメモリは、追加のセクタ消去コードの書込みを受け付けます。このことを確認するために、引き続きセクタ消去コードの書込みに先立って、このフラグの状態をチェックすることを推奨します。もし、2 回目の状態チェックで "1" であったら、追加セクタの消去コードは受け付けられていない可能性があります。

■ セクタ消去一時停止時

セクタ消去一時停止時に読出しアクセスすると、フラッシュメモリはアドレスの指し示す番地が消去中のセクタであれば "1" を出力し、消去中のセクタでなければアドレスの指し示す番地の読出し値の bit3(DATA:3) を出力します。

25.6 フラッシュメモリの書込み / 消去

自動アルゴリズムを起動するコマンドを発行し、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およびセクタ消去再開のそれぞれの動作手順について説明します。

■ フラッシュメモリの書込み / 消去

フラッシュメモリは、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および消去再開の動作がコマンドシーケンス (表 25.4-1 を参照) のバスへの書込みサイクルを行うことで自動アルゴリズムを実行できます。それぞれのバスへの書込みサイクルは、続けて行うことが必要です。また、自動アルゴリズムはデータポーリング機能などで終了時を知ることができます。正常終了後は、読出し / リセット状態に戻ります。

以下の項で、フラッシュメモリの書込み / 消去に関する下記項目について説明します。

- 読出し / リセット状態にする
- データを書き込む
- 全データを消去する (チップ全消去)
- 任意のデータを消去する (セクタ消去)
- セクタ消去を一時停止する
- セクタ消去を再開する

25.6.1 フラッシュメモリの読出し/リセット状態

読出し/リセットコマンドを発行し、フラッシュメモリを読出し/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読出し/リセット状態にする

フラッシュメモリを読出し/リセット状態にするには、コマンドシーケンス表(表 25.4-1 を参照) の読出し/リセットコマンドを、フラッシュメモリ内の対象セクタに続けて送ることで実行できます。

読出し/リセットコマンドには、1 回と 3 回のバス動作を行う 2 とおりのコマンドシーケンスがありますが、これらの本質的な違いはありません。

読出し/リセット状態は、フラッシュメモリの初期状態であり、電源投入時、コマンドの正常終了時は常に読出し/リセット状態になります。読出し/リセット状態は、ほかのコマンドの入力待ち状態です。

読出し / リセット状態では、通常の読出しアクセスで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可能です。通常読出しでのデータ読出しに、このコマンドは必要ありません、何らかの理由でコマンドが正常に終了しなかったときなど、自動アルゴリズムを初期化する場合に、主にこのコマンドを使用します。

25.6.2 フラッシュメモリのデータ書込み

書込みコマンドを発行し、フラッシュメモリにデータを書き込む手順について説明します。

■ フラッシュメモリへのデータ書込み

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には、コマンドシーケンス表 (表 25.4-1 を参照) の書込みコマンドを、フラッシュメモリ内の対象セクタに続けて送ることで実行できます。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時点で、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の中で指定する書込みアドレスは、偶数アドレスのみが可能です。奇数アドレスを指定すると正しく書き込むことができません。つまり、偶数アドレスへのワードデータ単位での書込みが必要となります。

書込みはどのようなアドレスの順番でも、また、セクタの境界を超えて可能ですが、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、データ 0 をデータ 1 に戻すことはできません。データ "0" にデータ "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) が終了せずに、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タイミングリミット超過フラグ (DQ6) がエラーと判定するかまたは見かけ上データ "1" が書き込まれたように見えるかのどちらかとなります。しかし、読出し / リセット状態でデータを読み出すとデータは "0" のままです。消去動作でのみ "0" のデータを "1" にすることができます。

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ると、書き込んでいるアドレスのデータは保証されませんので、注意が必要です。

■ フラッシュメモリの書込み手順

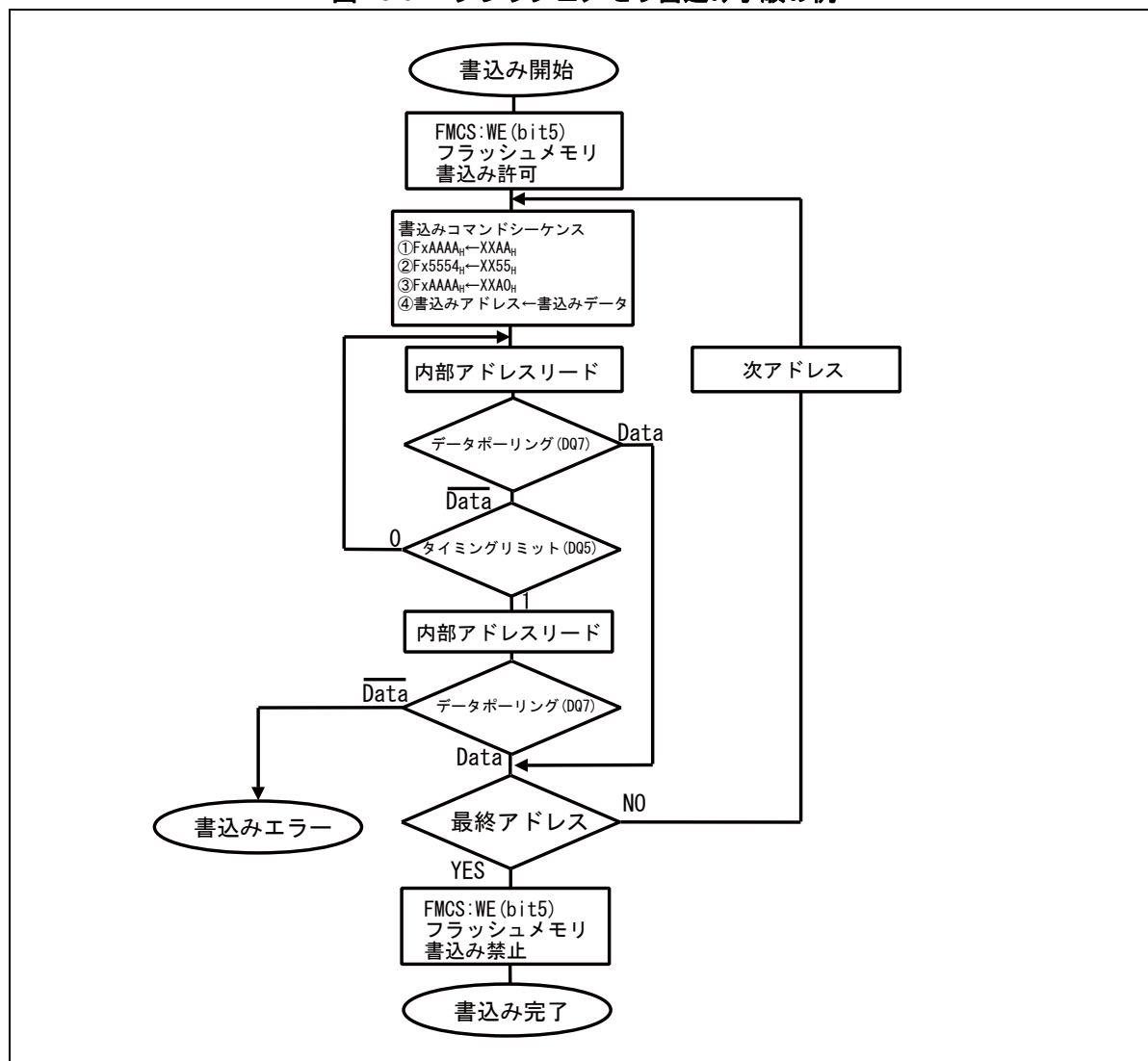
図 25.6-1 に、フラッシュメモリ書込み手順の例を示します。ハードウェアシーケンスフラグ (「25.5 自動アルゴリズム実行状態の確認」を参照) を用いることでフラッシュメモリ内部の自動アルゴリズムの状態判定ができます。ここでは、書込み終了の確認に、データポーリングフラグ (DQ7) を用いています。

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) と同時に変わるので、たとえタイミングリミット超過フラグ (DQ5) が "1" であっても、データポーリングフラグビット (DQ7) は再チェックする必要があります。

トグルビットフラグ (DQ6) でも同様に、タイミングリミット超過フラグビット (DQ5) が "1" に変わると同時にトグル動作を止めるので、トグルビットフラグ (DQ6) を再チェックする必要があります。

図 25.6-1 フラッシュメモリ書き込み手順の例



25.6.3 フラッシュメモリの全データ消去 (チップ消去)

チップ消去コマンドを発行し、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全データを消去 (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には、コマンドシーケンス (表 25.4-1 を参照) のチップ消去コマンドをフラッシュメモリ内の対象セクタに送ることで実行できます。

チップ消去コマンドは、6 回のバス動作で行われます。6 サイクル目の書込みが完了した時点でチップ消去動作が開始します。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は、フラッシュメモリは自動的にすべてのセルを消去する前に "0" を書き込んで検証します。

25.6.4 フラッシュメモリの任意のデータ消去 (セクタ消去)

セクタ消去コマンドを発行し、フラッシュメモリの任意のセクタを消去する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 (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(表 25.4-1 を参照)のセクタ消去コマンドをフラッシュメモリ内の対象セクタに続けて送ることで実行できます。

● 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 サイクル目に目的のセクタ内のアクセス可能な任意の偶数の 1 アドレスへ、セクタ消去コード (30_H) を書き込むことによって最小 50 μ s のセクタ消去ウェイトが開始します。複数のセクタを消去する場合は、上記の処理に引き続き、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最小 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。つまり、複数のセクタを同時に消去する場合は、次の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) をそれぞれ 50 μ s 以内に入力する必要があるため、それ以降では受け付けられないことがあります。引き続きセクタ消去コードの書込みが有効かどうかは、セクタ消去タイマ (ハードウェアシーケンスフラグ : DQ3) によって調べることができます。このとき、セクタ消去タイマを読み出すアドレスは、消去しようとしているセクタを指すようにします。

■ セクタ消去の手順

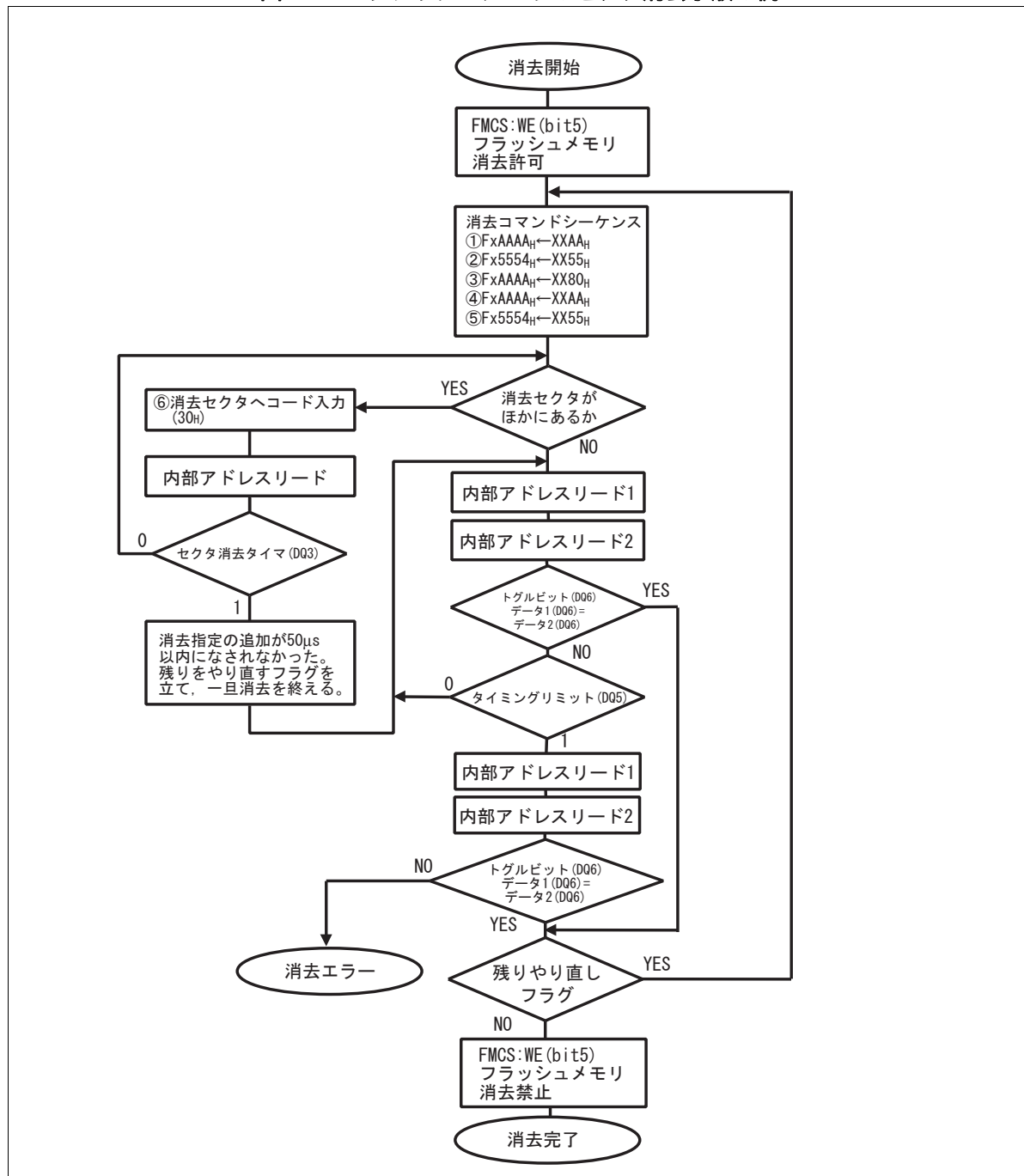
ハードウェアシーケンスフラグ (「25.5 自動アルゴリズム実行状態の確認」を参照) を用いることによってフラッシュメモリ内部の自動アルゴリズムの状態判定ができます。図 25.6-2 に、フラッシュメモリのセクタ消去手順の例を示します。ここでは、消去終了の確認にトグルビットフラグ (DQ6) を用いています。

フラグチェックのために読み込むデータは、消去しようとしているセクタからの読み込みとなりますので、注意が必要です。

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" に変わるのと同時にトグル動作を止めるので、たとえタイミングリミット超過フラグ (DQ5) が "1" であっても、トグルビットフラグ (DQ6) は再チェックする必要があります。

データポーリングフラグ (DQ7) でも同様に、タイミングリミット超過フラグ (DQ5) と同時に変わるので、データポーリングフラグ (DQ7) を再チェックする必要があります。

図 25.6-2 フラッシュメモリのセクタ消去手順の例



25.6.5 フラッシュメモリのセクタ消去を一時停止

セクタ消去一時停止コマンドを発行し、フラッシュメモリのセクタ消去を一時停止する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できます。

■ フラッシュメモリのセクタ消去を一時停止

フラッシュメモリのセクタ消去を一時停止するには、コマンドシーケンス表 (表 25.4-1 を参照) のセクタ消去一時停止コマンドを、フラッシュメモリ内に送ることで実行できます。

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のデータ読出しを可能にするものです。この状態では、読出しのみが可能で書込みはできません。このコマンドは、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動作中は無視されます。

消去一時停止コード (B0_H) の書込みを行うことで実施されますが、このときアドレスはフラッシュメモリ内の任意のアドレスを指すようにします。消去一時停止での再度の消去一時停止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が入力されると、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動作中に消去一時停止コマンドが入力されると、最大 20 μ s の時間の後、消去一時停止状態に入ります。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。

25.6.6 フラッシュメモリのセクタ消去を再開

セクタ消去再開コマンドを発行し、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

一時停止したセクタ消去を再開させるには、コマンドシーケンス表（表 25.4-1 を参照）のセクタ消去再開コマンドを、フラッシュメモリ内に続けて送ることで実行できます。

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。この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施されますが、このときのアドレスはフラッシュメモリ領域内の任意のアドレスを指すようにします。

なお、セクタ消去中のセクタ消去再開コマンドの発行は無視されます。

第26章

シリアル書込み接続例 (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)

この章では、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用いた場合の、シリアル書込み接続例について説明します。

26.1 基本構成

26.2 発振クロック周波数とシリアルクロック入力周波数について

26.3 フラッシュマイコンプログラマのシステム構成

26.4 シリアル書込み接続例

26.1 基本構成

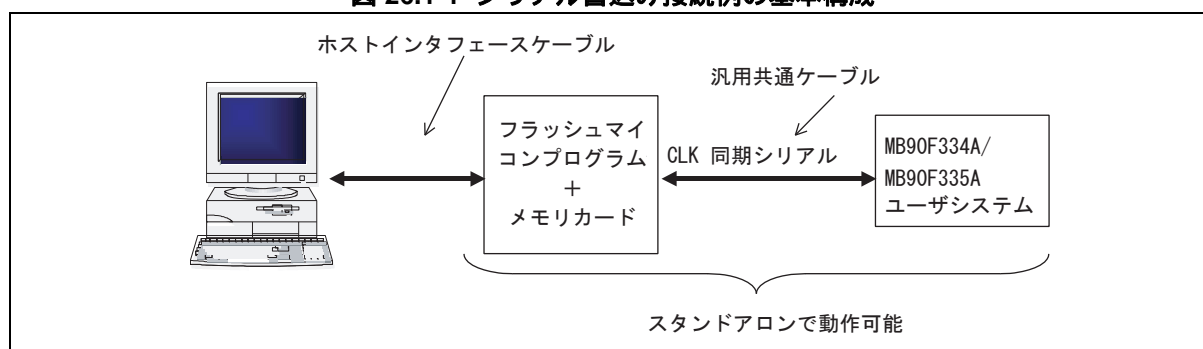
MB90F334A, MB90F335A ではフラッシュ ROM のシリアルオンボード書込み (富士通マイクロエレクトロニクス標準) をサポートしています。その仕様について以下に解説します。

■ 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。シングルチップモードで動作するプログラムあるいは内 ROM 外バスモードで動作するプログラムのどちらかを選択して書き込むことができます。

図 26.1-1 に、シリアル書込み接続例の基本構成を示します。

図 26.1-1 シリアル書込み接続例の基本構成



< 注意事項 >

フラッシュマイコンプログラマの機能、操作方法および接続用汎用共通ケーブル、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殿にお問い合わせください。

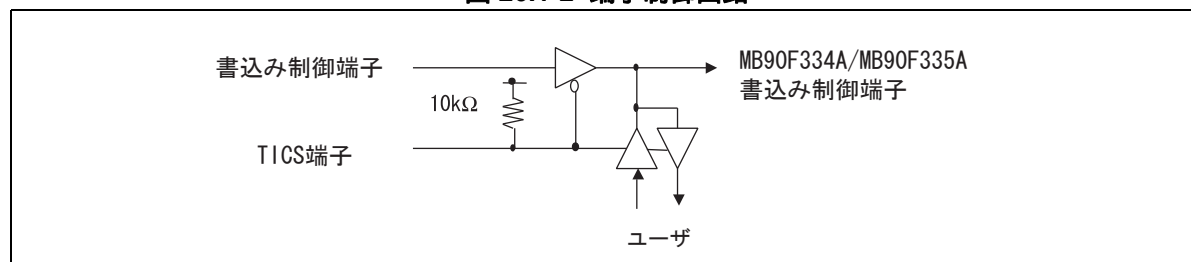
■ 富士通標準シリアルオンボード書込みに使用する端子

表 26.1-1 に、富士通標準シリアルオンボード書込みに使用する端子の機能を示します。

表 26.1-1 使用する端子の機能

端子	機能	補足説明
MD2, MD1, MD0	モード端子	MD2=1, MD1=1, MD0=0 に設定することで、シリアル書込みモードになります。
X0, X1	発振用端子	シリアル書込みモード時に、CPU 内部動作クロックは PLL クロック 1 週倍となりますので、内部動作クロック周波数は、発振クロック周波数と同様になります。シリアル書込みを行う場合、高速発振入力端子に入力可能な周波数は 6 MHz 固定となります。
P60, P61	書込みプログラム起動端子	P60 に "L" レベル, P61 に "H" レベルを入力してください。
$\overline{\text{RST}}$	リセット	-
SIN0	シリアルデータ入力	UART0 を CLK 同期モードとして使用します。
SOT0	シリアルデータ出力	
SCK0	シリアルクロック入力	
V _{CC}	電源電圧供給	書込み電圧 (V _{CC} =3.3 V \pm 0.3 V)
V _{SS}	GND	フラッシュマイコンプログラムの GND と共通にします。

図 26.1-2 端子制御回路



< 注意事項 >

- P60, P61, SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には、図 26.1-2 に示す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。「26.4 シリアル書込み接続例」をご参照ください。

26.2 発振クロック周波数とシリアルクロック入力周波数について

MB90F334A, MB90F335A の入力可能なシリアルクロック周波数は、以下の計算式により求められます。したがって、使用する発振クロック周波数に対応した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ムの設定により、変更してください。

■ 発振クロック周波数とシリアルクロック入力周波数

入力可能なシリアルクロック周波数は以下の計算式により求められます。

$$\text{入力可能なシリアルクロック周波数} = 0.125 \times \text{発振クロック周波数}$$

表 26.2-1 に、入力可能なシリアルクロック周波数を示します。

表 26.2-1 入力可能なシリアルクロック周波数

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	設定可能な最大シリアルクロック周波数
6 MHz 時	750 kHz	500 kHz

26.3 フラッシュマイコンプログラムのシステム構成

フラッシュマイコンプログラムのシステム構成を示します。

■ フラッシュマイコンプログラマシステム

表 26.3-1 フラッシュマイコンプログラムのシステム構成

名称	型格	仕様
フラッシュマイコン プログラマ本体	AF420/AC4P	<ul style="list-style-type: none"> ・ FULL KEY モデル ・ 100BASE-TX ホストインタフェース付き
	AF620/AC4P	<ul style="list-style-type: none"> ・ FULL KEY モデル ・ CAN インタフェース付き ・ 100BASE-TX ホストインタフェース付き
	AF320/AC4P	<ul style="list-style-type: none"> ・ 単 KEY モデル ・ 100BASE-TX ホストインタフェース付き
	AF520/AC4P	<ul style="list-style-type: none"> ・ 単 KEY モデル ・ CAN インタフェース付き ・ 100BASE-TX ホストインタフェース付き
汎用共通ケーブル	AZ410	標準ターゲットプローブ (a) 長さ : 1 m
コンパクトモジュール	FF801	富士通マイクロエレクトロニクス製マイコン用 コントロールモジュール
リモートコントローラ	AZ490	リモートコントローラ
メモリカード	-	PC Card

< 問合せ先 > 横河デジタルコンピュータ株式会社 機器事業センター 営業部 TEL(042)333-6224

< 注意事項 >

AF2xx/AF1xx シリーズは 2007 年 3 月末日をもって販売終了しています (製品の保守 (修理) は販売終了時点より 5 年間継続)。

今後、新規もしくは追加購入をご希望のお客様は、後継機種 (AF400/300, AF600/500 シリーズ) のご検討をお願いいたします。

詳細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。

なお、後継機種におきましても、富士通マイコンのサポートは引き続き行ってまいります。

26.4 シリアル書込み接続例

シリアル書込み接続例を示します。

■ シリアル書込み接続例

シリアル書込み接続例には、以下の 2 つがあります。

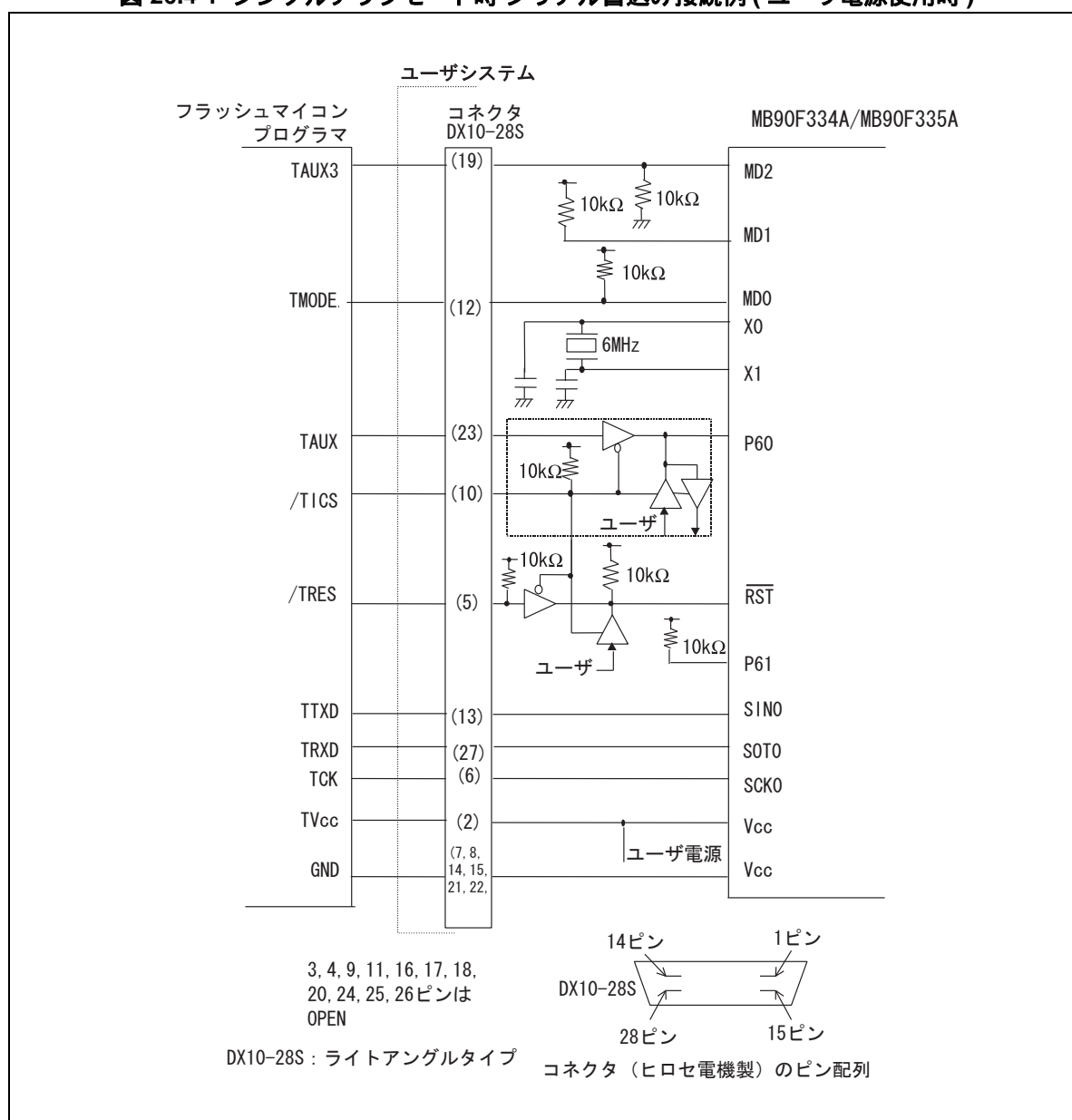
- シングルチップモード時の接続例（ユーザ電源使用時）
- フラッシュマイコンプログラマとの最小限の接続例（ユーザ電源使用時）

26.4.1 シングルチップモード時の接続例
(ユーザ電源使用時)

ユーザシステムにて、シングルチップモードに設定されているモード端子 MD2, MD0 には、フラッシュマイコンプログラムの TAUX3, TMODE より MD2=1, MD0=0 が入力され、シリアル書込みモードになります。
(シリアル書込みモード : MD2, MD1, MD0=110)

■ シングルチップモード時の接続例 (ユーザ電源使用時)

図 26.4-1 シングルチップモード時 シリアル書込み接続例 (ユーザ電源使用時)



< 注意事項 >

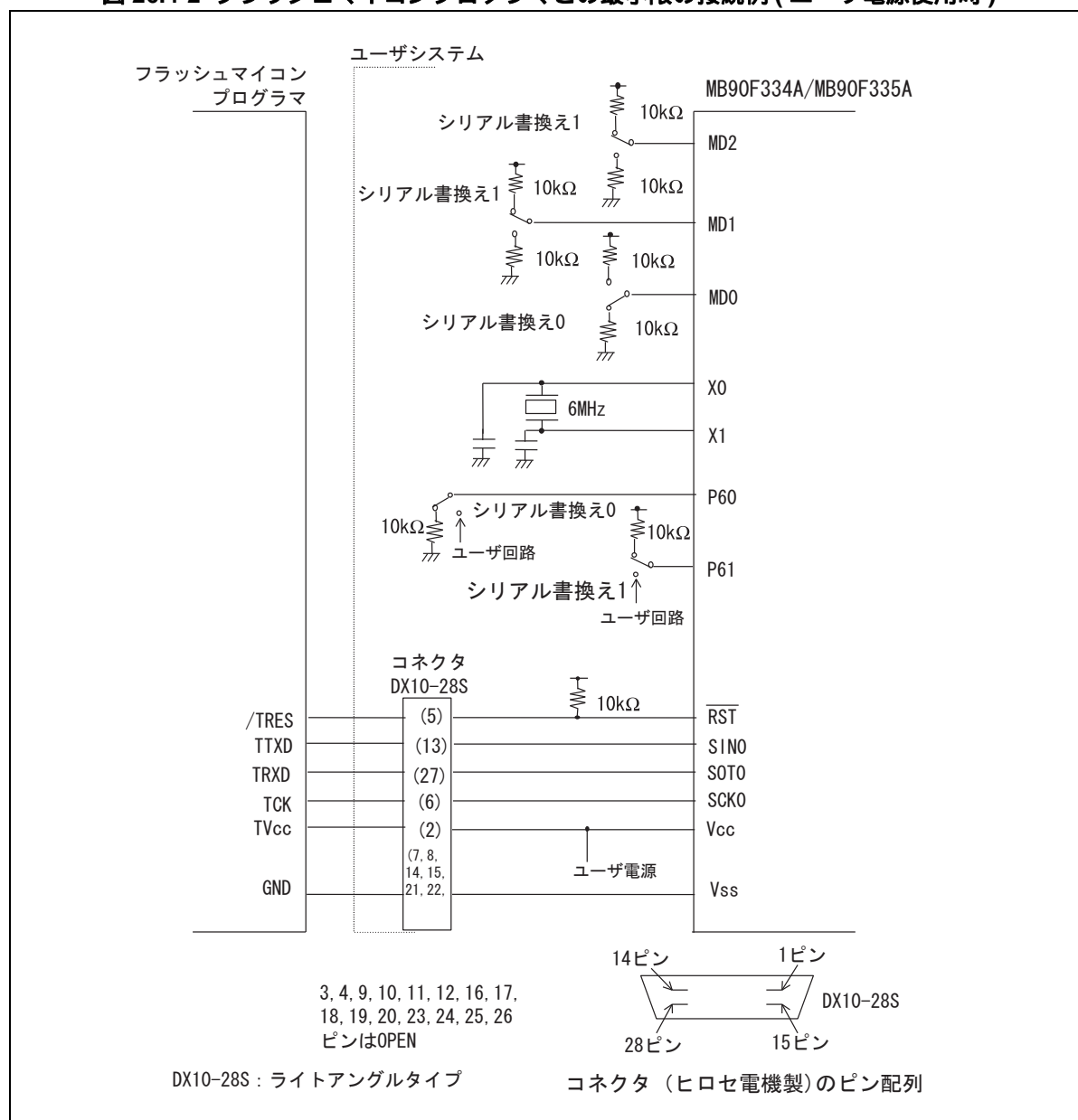
- SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には, P60 と同様, 図 26.1-2 に示す制御回路が必要となります (フラッシュマイコンプログラムの /TICS 信号により, シリアル書込み中はユーザ回路を切り離すことができます)。
 - フラッシュマイコンプログラマとの接続は, ユーザ電源が OFF の状態で行ってください。
-

26.4.2 フラッシュマイコンプログラマとの最小限の接続例
(ユーザ電源使用時)

シリアル書込み時に , 各端子 (MD2, MD0, P60) を図 26.4-2 に示すように設定した場合は , MD2, MD0, P60 とフラッシュマイコンプログラマと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との最小限の接続例 (ユーザ電源使用時)

図 26.4-2 フラッシュマイコンプログラマとの最小限の接続例 (ユーザ電源使用時)



< 注意事項 >

- SIN0, SOT0, SCK0 端子をユーザシステムでも使用する場合には、図 26.1-2 に示す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。
-

第27章

シリアル書込み接続例 (富士通マイクロエレクトロニクス製シリアルプログラマ)

MB90F334A/335A は、フラッシュメモリのシリアルオンボード書込み (富士通マイクロエレクトロニクス標準) に対応しています。

本章では、フラッシュメモリのシリアル書込みについて、富士通マイクロエレクトロニクス製シリアルプログラマを用いた場合の基本構成を説明します。

27.1 富士通マイクロエレクトロニクス製 シリアルプログラマ

27.1 富士通マイクロエレクトロニクス製 シリアルプログラマ

富士通マイクロエレクトロニクス製シリアルプログラマ(ソフトウェア)は、富士通マイクロエレクトロニクス製のフラッシュメモリ内蔵マイコン全製品を対象にしたオンボードプログラミングツールです。

本シリアルプログラマは、PC との I/F (RS-232C, USB) により、2 種類のツールを用意していますので、環境に合わせてご利用いただけます。

■ FUJITSU MICROELECTRONICS MCU Programmer(クロック非同期 シリアル書込み) の基本構成

FUJITSU MICROELECTRONICS MCU Programmer は、パソコンとマイコンを RS-232C ケーブルで接続し、クロック非同期シリアル通信によりユーザシステムに実装されているマイコン内蔵フラッシュメモリヘータの書込みを行います。

FUJITSU MICROELECTRONICS MCU Programmer の基本構成を、システム構成を示します。

図 27.1-1 FUJITSU MICROELECTRONICS MCU Programmer の基本構成

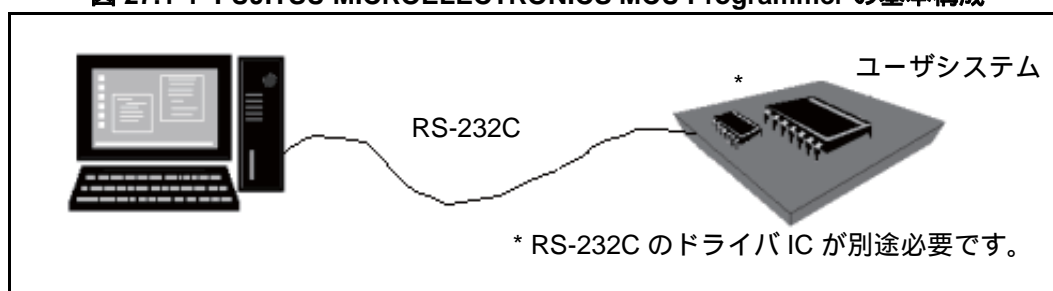


表 27.1-1 FUJITSU MICROELECTRONICS MCU Programmer のシステム構成

名称	型格	仕様
FUJITSU MICROELECTRONICS MCU Programmer	-	ソフト (Web よりダウンロード可能 (登録制)) *

* 登録は営業部門までお問い合わせください。

接続例を図 27.1-2 に示します。

図 27.1-2 FUJITSU MICROELECTRONICS MCU Programmer の接続例

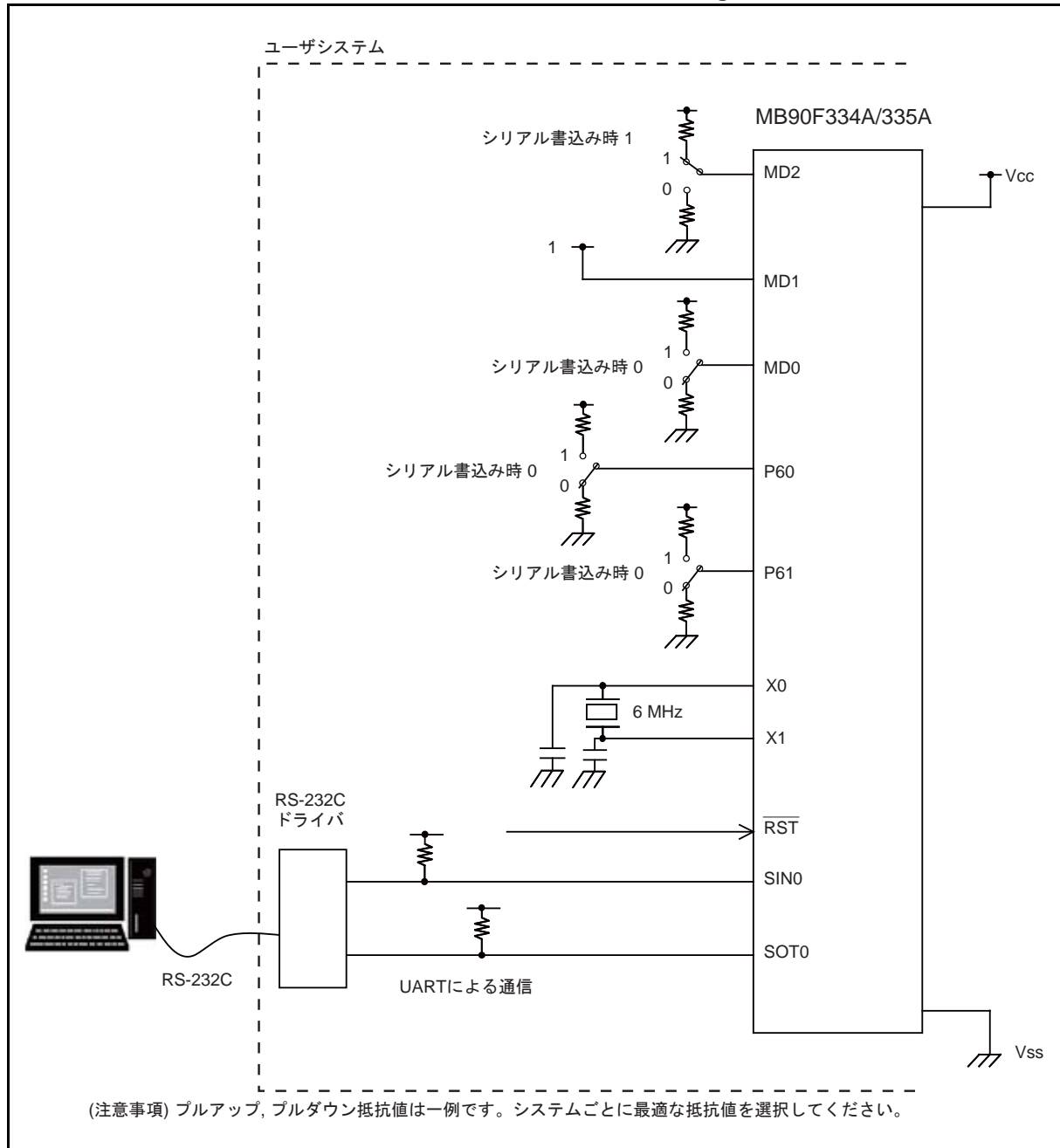


表 27.1-2 クロック非同期シリアル通信時の入力可能な発振周波数と通信ボーレート

原発振周波数	通信ボーレート
6MHz	19200bps

■ FUJITSU MICROELECTRONICS USB Programmer (クロック同期シリアル書き込み) の基本構成

FUJITSU MICROELECTRONICS USB Programmer は、パソコンとマイコンをアダプタ (MB2146-09A-E) で接続し、クロック同期シリアル通信によりマイコン内蔵フラッシュメモリへデータの書き込みを行います。

FUJITSU MICROELECTRONICS USB Programmer の基本構成を図 27.1-2, システム構成を示します。

図 27.1-3 FUJITSU MICROELECTRONICS USB Programmer の基本構成

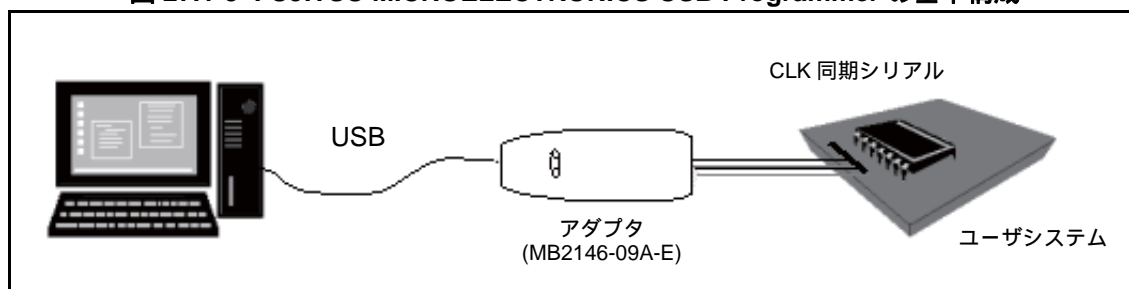


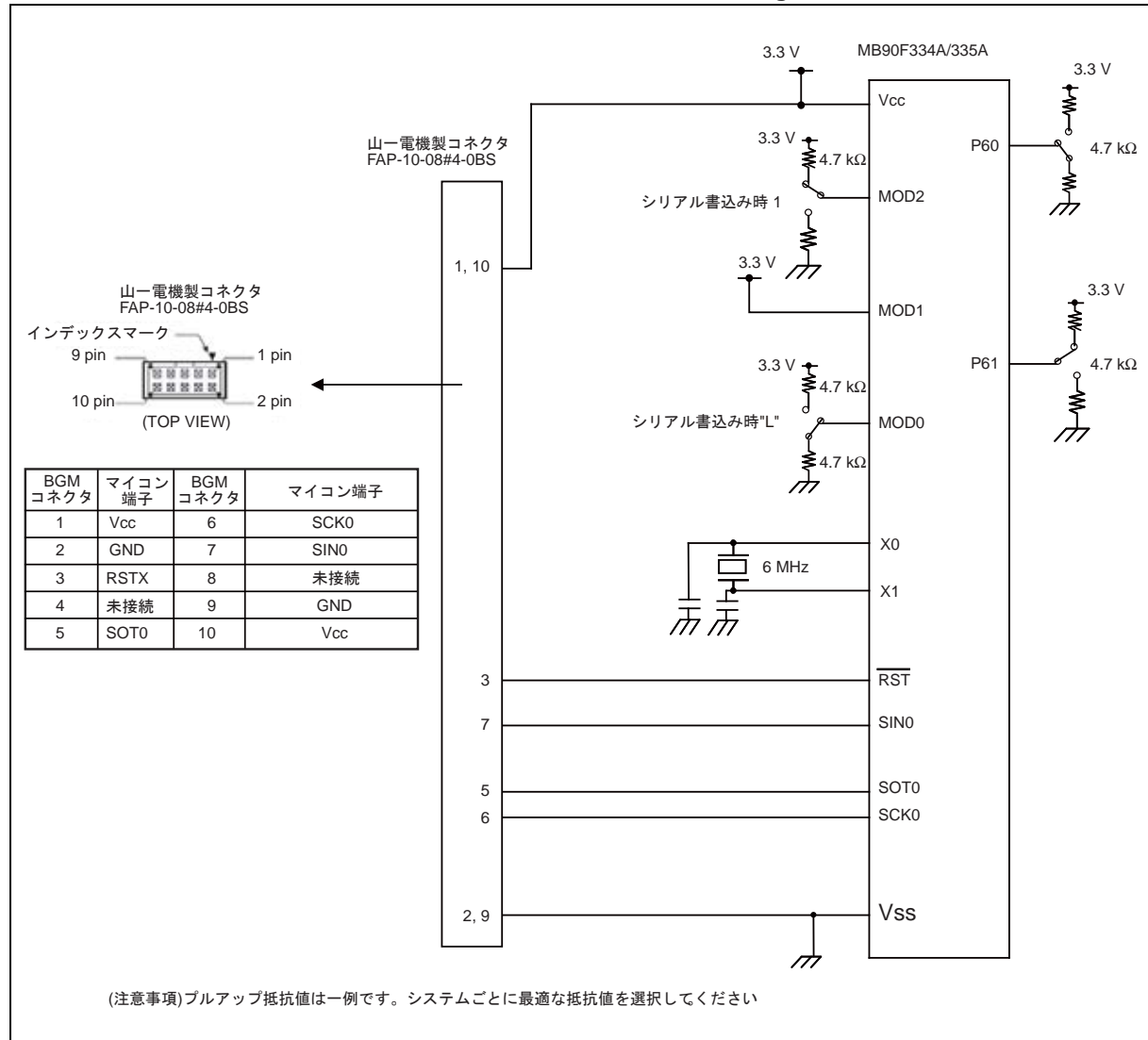
表 27.1-3 FUJITSU MICROELECTRONICS USB Programmer のシステム構成

名称	型格	仕様
FUJITSU MICROELECTRONICS USB Programmer	-	ソフト (Webよりダウンロード可能(登録制))*
アダプタ	MB2146-09A-E	F ² MC ファミリ BGM アダプタ (付属品: USB ケーブル)

* 登録は営業部門までお問い合わせください。

接続例を図 27.1-2 に示します。

図 27.1-4 FUJITSU MICROELECTRONICS USB Programmer の接続例



27.1.1 使用する端子

表 27.1-4 使用する端子の機能

端子	機能	補足説明
MD2, MD1, MD0	モード端子	MD2=1, MD1=1, MD0=0 に設定することで, シリアル書込みモードになります。
X0, X1	発振用端子	シリアル書込みモード時に, CPU 内部動作クロックは PLL クロック 1 逓倍となりますので, 内部動作クロック周波数は, 発振クロック周波数と同様になります。シリアル書込みを行う場合, 高速発振入力端子に入力可能な周波数は 6 MHz 固定となります。
P60, P61	書込みプログラム起動端子	同期シリアル書込み時は, P60 に "L" レベル, P61 に "H" レベルを入力してください。 非同期シリアル書込み時は, P60 に "L" レベル, P61 に "L" レベルを入力してください。
$\overline{\text{RST}}$	リセット	-
SIN0	シリアルデータ入力	UART0 を CLK 同期モード, 非同期モードで使用します。 (注意事項) CLK 非同期モード時は, SCK0 は使用しません。
SOT0	シリアルデータ出力	
SCK0	シリアルクロック入力	
V _{CC}	電源電圧供給	書込み電圧 (VCC=3.3 V \pm 0.3 V)
V _{SS}	GND	フラッシュマイコンプログラマの GND と共通にします。

付録では、メモリマップおよび F²MC-16LX に使用している命令について、説明します。

付録 A メモリマップ

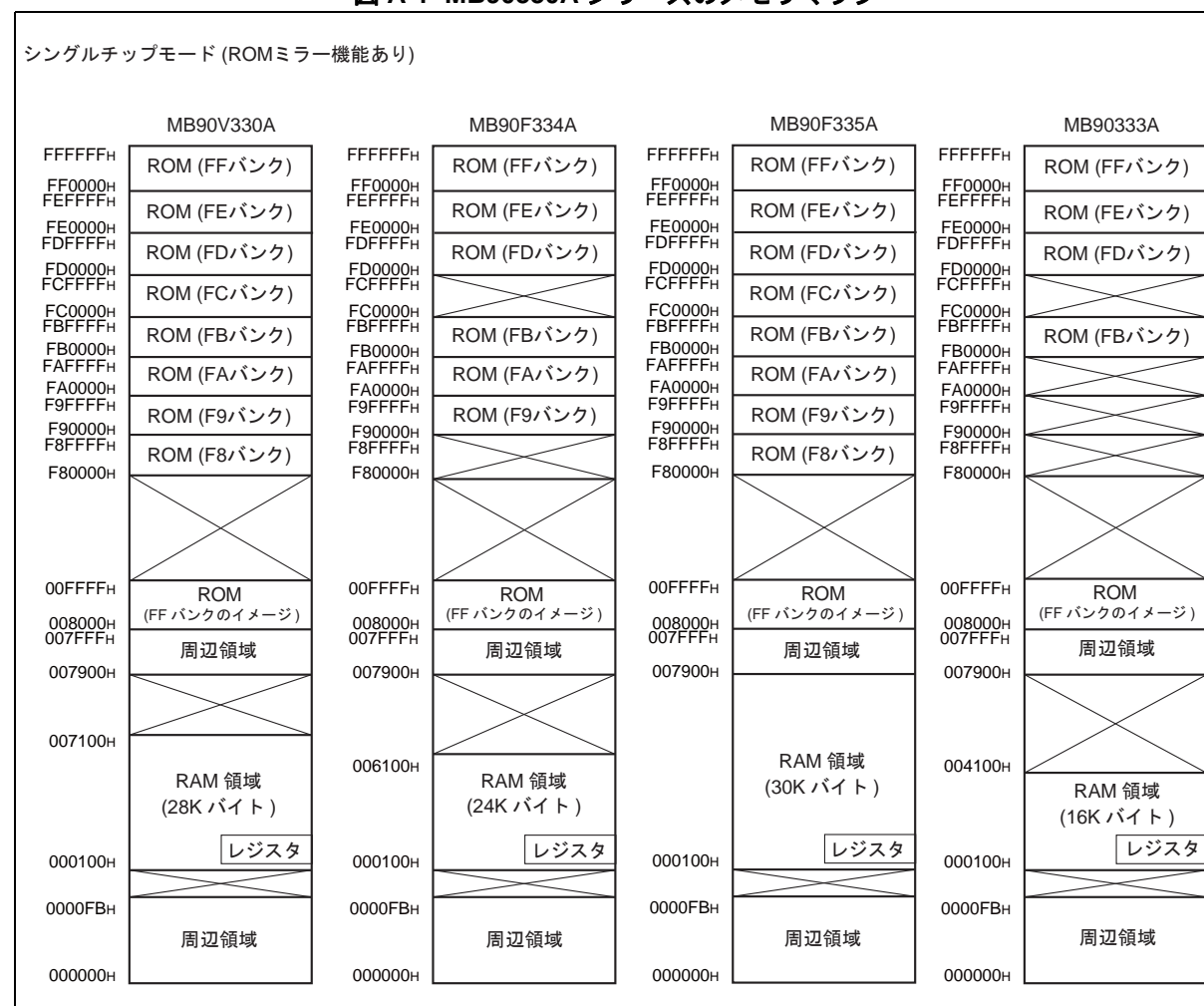
付録 B 命令

付録 A メモリマップ

メモリ空間は、3つのモードに分かれます。

■ メモリ空間

図 A-1 MB90330A シリーズのメモリマップ



< 注意事項 >

- ROM ミラー機能の設定は、「第 23 章 ROM ミラー機能選択モジュール」を参照してください。
 - ROM ミラー機能レジスタの設定を行った場合、00 バンクの上位側 ("008000_H ~ 00FFFF_H") に FF バンクの上位側 ("FF8000_H ~ FFFFFFFF_H") のデータがミラーイメージで見えるようになります。
 - ROM ミラー機能は、C コンパイラのスモールモデルを有効に生かすことができます。
 - FF バンクの下位 16 ビットアドレスは、00 バンクの下位 16 ビットアドレスと同じになります。ただし、FF バンクの ROM 領域は 48K バイトを超えますので、00 バンクには、ROM 領域の全データをミラーイメージで見せることはできません。
 - C コンパイラのスモールモデルをご使用の場合は、データテーブルを "FF8000_H ~ FFFFFFFF_H" に格納しておくことで、"008000_H ~ 00FFFF_H" のミラーイメージとしてデータテーブルを見せることができます。したがって、ポインタで far 指定を宣言することなく、ROM 領域内のデータテーブルを参照できます。
 - MB90F335A は、搭載 RAM 領域が MB90V330A よりも大きいため、007100_H より大きい領域のエミュレーションは、ツール側でエミュレーションメモリエリアの設定を行う必要があります。設定については、下記 URL より「アプリケーションノート」「MB90330A series デバッグ環境設定に関する注意事項」を参照してください。
<http://edevic.fujitsu.com/micom/jp-support/>
 - エミュレーションメモリ領域 (007100_H ~ 0078FF_H) へのアクセスは、搭載 RAM 領域へのアクセスより 1 サイクル多く、3 サイクルとなります。
-

図 A-2 MB90330A シリーズのメモリマップ

内ROM外バスモード (ROMミラー機能あり)							
MB90V330A		MB90F334A		MB90F335A		MB90333A	
FFFFFH	ROM (FFバンク)	FFFFFH	ROM (FFバンク)	FFFFFH	ROM (FFバンク)	FFFFFH	ROM (FFバンク)
FF0000H FEFFFFH	ROM (FEバンク)	FF0000H FEFFFFH	ROM (FEバンク)	FF0000H FEFFFFH	ROM (FEバンク)	FF0000H FEFFFFH	ROM (FEバンク)
FE0000H FDFFFFH	ROM (FDバンク)	FE0000H FDFFFFH	ROM (FDバンク)	FE0000H FDFFFFH	ROM (FDバンク)	FE0000H FDFFFFH	ROM (FDバンク)
FD0000H FCFFFFH	ROM (FCバンク)	FD0000H FCFFFFH	ROM (FCバンク)	FD0000H FCFFFFH	ROM (FCバンク)	FD0000H FCFFFFH	ROM (FCバンク)
FC0000H FBFFFFH	ROM (FBバンク)	FC0000H FBFFFFH	ROM (FBバンク)	FC0000H FBFFFFH	ROM (FBバンク)	FC0000H FBFFFFH	ROM (FBバンク)
FB0000H FAFFFFH	ROM (FAバンク)	FB0000H FAFFFFH	ROM (FAバンク)	FB0000H FAFFFFH	ROM (FAバンク)	FB0000H FAFFFFH	ROM (FAバンク)
FA0000H F9FFFFH	ROM (F9バンク)	FA0000H F9FFFFH	ROM (F9バンク)	FA0000H F9FFFFH	ROM (F9バンク)	FA0000H F9FFFFH	外部領域
F90000H F8FFFFH	ROM (F8バンク)	F90000H F8FFFFH	ROM (F8バンク)	F90000H F8FFFFH	ROM (F8バンク)	F90000H F8FFFFH	外部領域
F80000H	外部領域	F80000H	外部領域	F80000H	外部領域	F80000H	外部領域
00FFFFH	ROM (FFバンクのイメージ)	00FFFFH	ROM (FFバンクのイメージ)	00FFFFH	ROM (FFバンクのイメージ)	00FFFFH	ROM (FFバンクのイメージ)
008000H 007FFFH	周辺領域	008000H 007FFFH	周辺領域	008000H 007FFFH	周辺領域	008000H 007FFFH	周辺領域
007900H	外部領域	007900H	外部領域	007900H	外部領域	007900H	外部領域
007100H	RAM 領域 (28K バイト)	006100H	RAM 領域 (24K バイト)	007100H	RAM 領域 (30K バイト)	004100H	RAM 領域 (16K バイト)
000100H	レジスタ	000100H	レジスタ	000100H	レジスタ	000100H	レジスタ
0000FBH	周辺領域	0000FBH	周辺領域	0000FBH	周辺領域	0000FBH	周辺領域
000000H		000000H		000000H		000000H	

*1: MB90F334A では、F80000_H ~ F8FFFF_H、FC0000_H ~ FCFFFF_H の領域のリード動作時には "1" がリードされます。

*2: MB90333A では、FA0000_H ~ FAFFFF_H、FC0000_H ~ FCFFFF_H の領域のリード動作時には "1" がリードされます。

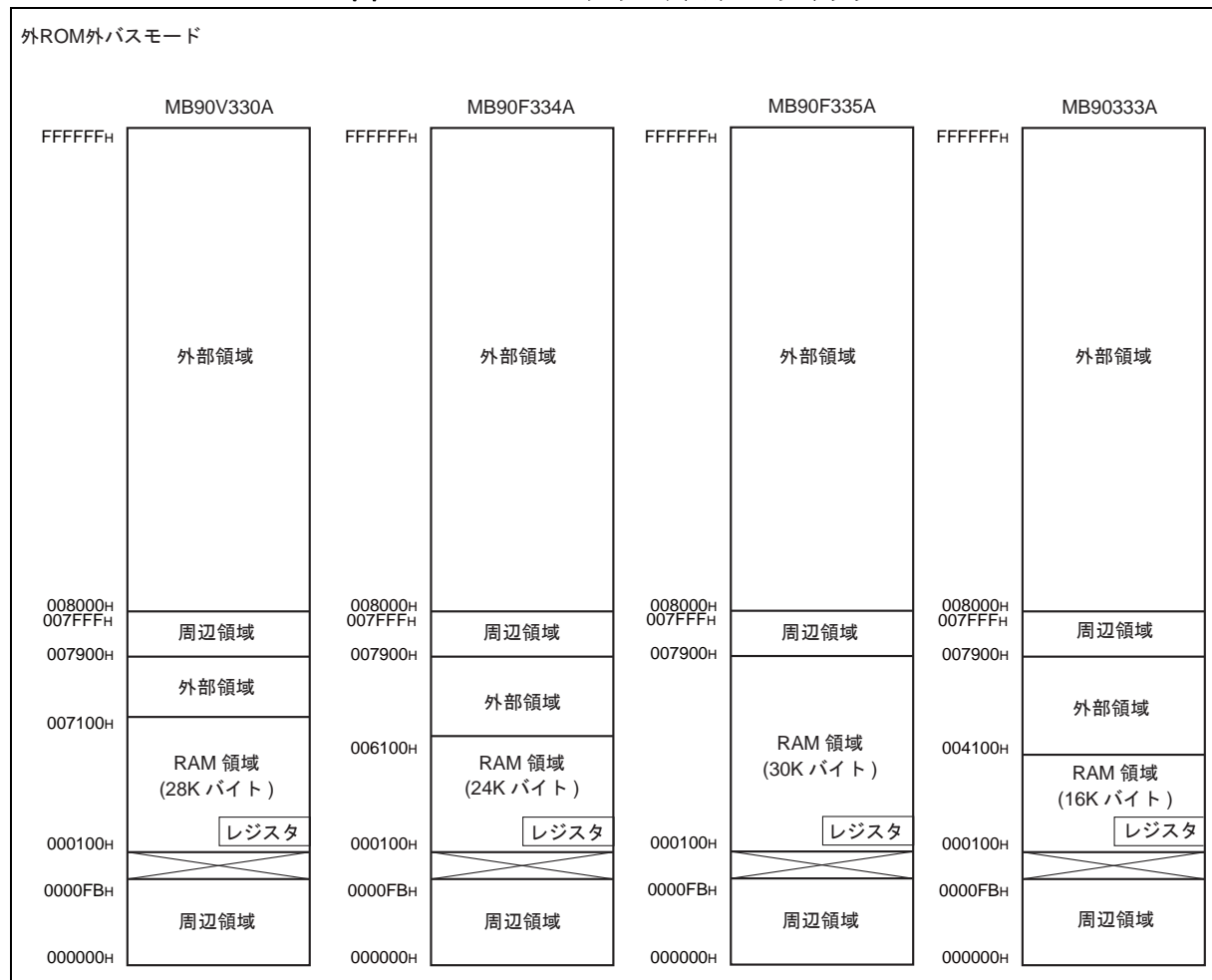
< 注意事項 >

- ROM ミラー機能レジスタの設定を行った場合、00 バンクの上位側 ("008000_H ~ 00FFFF_H") に FF バンクの上位側 ("FF8000_H ~ FFFFFFF_H") のデータがミラーイメージで見えるようになります。
- ROM ミラー機能は、C コンパイラのスモールモデルを有効に生かすことができます。
- FF バンクの下位 16 ビットアドレスは、00 バンクの下位 16 ビットアドレスと同じになります。ただし、FF バンクの ROM 領域は 48K バイトを超えますので、00 バンクには、ROM 領域の全データをミラーイメージで見せることはできません。
- C コンパイラのスモールモデルをご使用の場合は、データテーブルを "FF8000_H ~ FFFFFFF_H" に格納しておくことで、"008000_H ~ 00FFFF_H" のミラーイメージとしてデータテーブルを見せることができます。したがって、ポインタで far 指定を宣言することなく、ROM 領域内のデータテーブルを参照できます。
- MB90F335A は、搭載 RAM 領域が MB90V330A よりも大きいため、007100_H より大きい領域のエミュレーションは、ツール側でエミュレーションメモリエリアの設定を行う必要があります。設定については、下記 URL より「アプリケーションノート」「MB90330A series デバッグ環境設定に関する注意事項」を参照してください。

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

- エミュレーションメモリ領域 (007100_H ~ 007FFF_H) へのアクセスは、搭載 RAM 領域へのアクセスより 1 サイクル多く、3 サイクルとなります。

図 A-3 MB90330A シリーズのメモリマップ



< 注意事項 >

- ROM ミラー機能レジスタの設定を行った場合、00 バンクの上位側 ("008000_H ~ 00FFFF_H") に FF バンクの上位側 ("FF8000_H ~ FFFFFFF_H") のデータがミラーイメージで見えるようになります。
- ROM ミラー機能は、C コンパイラのスモールモデルを有効に生かすことができます。
- FF バンクの下位 16 ビットアドレスは、00 バンクの下位 16 ビットアドレスと同じになります。ただし、FF バンクの ROM 領域は 48K バイトを超えますので、00 バンクには、ROM 領域の全データをミラーイメージで見せることはできません。
- C コンパイラのスモールモデルをご使用の場合は、データテーブルを "FF8000_H ~ FFFFFFF_H" に格納しておくことで、"008000_H ~ 00FFFF_H" のミラーイメージとしてデータテーブルを見せることができます。したがって、ポインタで far 指定を宣言することなく、ROM 領域内のデータテーブルを参照できます。
- MB90F335A は、搭載 RAM 領域が MB90V330A よりも大きいいため、007100_H より大きい領域のエミュレーションは、ツール側でエミュレーションメモリエリアの設定を行う必要があります。設定については、下記 URL より「アプリケーションノート」「MB90330A series デバッグ環境設定に関する注意事項」を参照してください。
<http://edevic.fujitsu.com/micom/jp-support/>
- エミュレーションメモリ領域 (007100_H ~ 007FFF_H) へのアクセスは、搭載 RAM 領域へのアクセスより 1 サイクル多く、3 サイクルとなります。

■ I/O マップ

表 A-1 に、各周辺機能のレジスタに割り当てられるアドレスを示します。

表 A-1 I/O マップ (1/7)

アドレス	レジスタ	略称	アクセス	リソース	初期値
000000 _H	ポート 0 データレジスタ	PDR0	R/W	ポート 0	XXXXXXXXXX
000001 _H	ポート 1 データレジスタ	PDR1	R/W	ポート 1	XXXXXXXXXX
000002 _H	ポート 2 データレジスタ	PDR2	R/W	ポート 2	XXXXXXXXXX
000003 _H	ポート 3 データレジスタ	PDR3	R/W	ポート 3	XXXXXXXXXX
000004 _H	ポート 4 データレジスタ	PDR4	R/W	ポート 4	XXXXXXXXXX
000005 _H	ポート 5 データレジスタ	PDR5	R/W	ポート 5	XXXXXXXXXX
000006 _H	ポート 6 データレジスタ	PDR6	R/W	ポート 6	XXXXXXXXXX
000007 _H	ポート 7 データレジスタ	PDR7	R/W	ポート 7	XXXXXXXXXX
000008 _H	ポート 8 データレジスタ	PDR8	R/W	ポート 8	XXXXXXXXXX
000009 _H	ポート 9 データレジスタ	PDR9	R/W	ポート 9	-XXXXXXXXX
00000A _H	ポート A データレジスタ	PDRA	R/W	ポート A	XXXXXXXXXX
00000B _H	使用禁止				
00000C _H	ポート B データレジスタ	PDRB	R/W	ポート B	-XXXXXXXXX
00000D _H	ポート B 方向レジスタ	DDRB	R/W	ポート B	-0000000
00000E _H	使用禁止				
00000F _H					
000010 _H	ポート 0 方向レジスタ	DDR0	R/W	ポート 0	00000000
000011 _H	ポート 1 方向レジスタ	DDR1	R/W	ポート 1	00000000
000012 _H	ポート 2 方向レジスタ	DDR2	R/W	ポート 2	00000000
000013 _H	ポート 3 方向レジスタ	DDR3	R/W	ポート 3	00000000
000014 _H	ポート 4 方向レジスタ	DDR4	R/W	ポート 4	00000000
000015 _H	ポート 5 方向レジスタ	DDR5	R/W	ポート 5	00000000
000016 _H	ポート 6 方向レジスタ	DDR6	R/W	ポート 6	00000000
000017 _H	ポート 7 方向レジスタ	DDR7	R/W	ポート 7	00000000
000018 _H	ポート 8 方向レジスタ	DDR8	R/W	ポート 8	00000000
000019 _H	ポート 9 方向レジスタ	DDR9	R/W	ポート 9	-0000000
00001A _H	ポート A 方向レジスタ	DDRA	R/W	ポート A	00000000
00001B _H	ポート 4 出力端子レジスタ	ODR4	R/W	ポート 4(OD 制御)	00000000
00001C _H	ポート 0 プルアップ抵抗レジスタ	RDR0	R/W	ポート 0(PULLUP)	00000000
00001D _H	ポート 1 プルアップ抵抗レジスタ	RDR1	R/W	ポート 1(PULLUP)	00000000
00001E _H	アナログ入力許可レジスタ 0	ADER0	R/W	ポート 7, A/D	11111111
00001F _H	アナログ入力許可レジスタ 1	ADER1	R/W	ポート 8, A/D	11111111
000020 _H	シリアルモードレジスタ 0	SMR0	R/W	UART0	00100000
000021 _H	シリアル制御レジスタ 0	SCR0	R/W,W		00000100
000022 _H	シリアル入力データレジスタ 0	SIDR0	R		XXXXXXXXXX
	シリアル出力データレジスタ 0	SODR0	W		
000023 _H	シリアルステータスレジスタ 0	SSR0	R/W,R		00001000
000024 _H	UART プリスケアラリロードレジスタ 0	UTRLR0	R/W	通信プリスケアラ (UART0)	00000000
000025 _H	UART プリスケアラ制御レジスタ 0	UTCR0	R/W		0000-000
000026 _H	シリアルモードレジスタ 1	SMR1	R/W	UART1	00100000
000027 _H	シリアル制御レジスタ 1	SCR1	R/W,W		00000100
000028 _H	シリアル入力データレジスタ 1	SIDR1	R		XXXXXXXXXX
	シリアル出力データレジスタ 1	SODR1	W		
000029 _H	シリアルステータスレジスタ 1	SSR1	R/W,R		00001000

表 A-1 I/O マップ (2/7)

アドレス	レジスタ	略称	アクセス	リソース	初期値
00002A _H	UART プリスケーラリロードレジスタ 1	UTRLR1	R/W	通信プリスケーラ (UART1)	00000000
00002B _H	UART プリスケーラ制御レジスタ 1	UTCR1	R/W		0000-000
00002C _H	シリアルモードレジスタ 2	SMR2	R/W	UART2	00100000
00002D _H	シリアル制御レジスタ 2	SCR2	R/W,W		00000100
00002E _H	シリアル入力データレジスタ 2	SIDR2	R		XXXXXXXXXX
	シリアル出力データレジスタ 2	SODR2	W		
00002F _H	シリアルステータスレジスタ 2	SSR2	R/W,R		00001000
000030 _H	UART プリスケーラリロードレジスタ 2	UTRLR2	R/W	通信プリスケーラ (UART2)	00000000
000031 _H	UART プリスケーラ制御レジスタ 2	UTCR2	R/W		0000-000
000032 _H	シリアルモードレジスタ 3	SMR3	R/W	UART3	00100000
000033 _H	シリアル制御レジスタ 3	SCR3	R/W,W		00000100
000034 _H	シリアル入力データレジスタ 3	SIDR3	R		XXXXXXXXXX
	シリアル出力データレジスタ 3	SODR3	W		
000035 _H	シリアルステータスレジスタ 3	SSR3	R/W,R		00001000
000036 _H	UART プリスケーラリロードレジスタ 3	UTRLR3	R/W	通信 プリスケーラ (UART3)	00000000
000037 _H	UART プリスケーラ制御レジスタ 3	UTCR3	R/W		0000-000
000038 _H ~ 00003B _H	使用禁止				
00003C _H	DTP/ 割り込み許可レジスタ	ENIR	R/W	DTP/ 外部割り込み	00000000
00003D _H	DTP/ 割り込み要因レジスタ	EIRR	R/W		00000000
00003E _H	要求レベル設定レジスタ下位	ELVR	R/W		00000000
00003F _H	要求レベル設定レジスタ上位		R/W		00000000
000040 _H	A/D 制御ステータスレジスタ下位	ADCS0	R/W	8/10 ビット A/D コンバータ	00----0
000041 _H	A/D 制御ステータスレジスタ上位	ADCS1	R/W,W		00000000
000042 _H	A/D データレジスタ下位	ADCR0	R		XXXXXXXXXX
000043 _H	A/D データレジスタ上位	ADCR1	W, R		00101XXX
000044 _H	使用禁止				
000045 _H	A/D 変換チャネル選択レジスタ	ADMR	R/W	8/10 ビット A/D コンバータ	00000000
000046 _H	PPG0 動作モード制御レジスタ	PPGC0	R/W	PPG ch.0	0X000XX1
000047 _H	PPG1 動作モード制御レジスタ	PPGC1	R/W	PPG ch.1	0X000001
000048 _H	PPG2 動作モード制御レジスタ	PPGC2	R/W	PPG ch.2	0X000XX1
000049 _H	PPG3 動作モード制御レジスタ	PPGC3	R/W	PPG ch.3	0X000001
00004A _H	PPG4 動作モード制御レジスタ	PPGC4	R/W	PPG ch.4	0X000XX1
00004B _H	PPG5 動作モード制御レジスタ	PPGC5	R/W	PPG ch.5	0X000001
00004C _H	PPG0, 1 出力制御レジスタ	PPG01	R/W	PPG ch.0/1	000000XX
00004D _H	使用禁止				
00004E _H	PPG2, 3 出力制御レジスタ	PPG23	R/W	PPG ch.2/3	000000XX
00004F _H	使用禁止				
000050 _H	PPG4, 5 出力制御レジスタ	PPG45	R/W	PPG ch.4/5	000000XX
000051 _H	使用禁止				
000052 _H	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 01	ICS01	R/W	インプット キャプチャ 0/1	00000000
000053 _H	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 23	ICS23	R/W	インプット キャプチャ 2/3	00000000
000054 _H	アウトプットコンペアコントロールレジスタ ch.0 下位	OCS0	R/W	アウトプット コンペア ch.0/1	0000--00
000055 _H	アウトプットコンペアコントロールレジスタ ch.1 上位	OCS1	R/W		---00000
000056 _H	アウトプットコンペアコントロールレジスタ ch.2 下位	OCS2	R/W	アウトプット コンペア ch.2/3	0000--00
000057 _H	アウトプットコンペアコントロールレジスタ ch.3 上位	OCS3	R/W		---00000

表 A-1 I/O マップ (3 / 7)

アドレス	レジスタ	略称	アクセス	リソース	初期値
000058 _H	シリアルモードコントロール ステータスレジスタ	SMCS	R/W	拡張シリアル I/O	XXXX0000
000059 _H					00000010
00005A _H	シリアルデータレジスタ	SDR	R/W		XXXXXXXXXX
00005B _H	通信プリスケアラ制御レジスタ	SDCR	R/W	通信プリスケアラ	0XXX0000
00005C _H	PWC コントロールステータスレジスタ	PWCSR	R/W,R	16 ビット PWC タイマ	00000000
00005D _H					0000000X
00005E _H	PWC データバッファレジスタ	PWCR	R/W		00000000
00005F _H					00000000
000060 _H	PWC 分周比制御レジスタ	DIVR	R/W		-----00
000061 _H	使用禁止				
000062 _H	タイマ制御ステータスレジスタ 0	TMCSR0	R/W	16 ビット リロードタイマ ch.0	00000000
000063 _H					XXXX0000
000064 _H	16 ビットタイマレジスタ 0 下位	TMR0	R		XXXXXXXXXX
	16 ビットリロードレジスタ 0 下位	TMRLR0	W		XXXXXXXXXX
000065 _H	16 ビットタイマレジスタ 0 上位	TMR0	R		XXXXXXXXXX
	16 ビットリロードレジスタ 0 上位	TMRLR0	W	XXXXXXXXXX	
000066 _H	タイマ制御ステータスレジスタ 1	TMCSR1	R/W	16 ビット リロードタイマ ch.1	00000000
000067 _H					XXXX0000
000068 _H	16 ビットタイマレジスタ 1 下位	TMR1	R		XXXXXXXXXX
	16 ビットリロードレジスタ 1 下位	TMRLR1	W		XXXXXXXXXX
000069 _H	16 ビットタイマレジスタ 1 上位	TMR1	R		XXXXXXXXXX
	16 ビットリロードレジスタ 1 上位	TMRLR1	W	XXXXXXXXXX	
00006A _H	タイマ制御ステータスレジスタ 2	TMCSR2	R/W	16 ビット リロードタイマ ch.2	00000000
00006B _H					XXXX0000
00006C _H	16 ビットタイマレジスタ 2 下位	TMR2	R		XXXXXXXXXX
	16 ビットリロードレジスタ 2 下位	TMRLR2	W		XXXXXXXXXX
00006D _H	16 ビットタイマレジスタ 2 上位	TMR2	R		XXXXXXXXXX
	16 ビットリロードレジスタ 2 上位	TMRLR2	W	XXXXXXXXXX	
00006E _H	使用禁止				
00006F _H	ROM ミラー機能選択レジスタ	ROMM	R/W,W	ROM ミラー機能	-----11
000070 _H	I ² C バスステータスレジスタ 0	IBSR0	R	I ² C バスインタフェース ch.0	00000000
000071 _H	I ² C バスコントロールレジスタ 0	IBCR0	R/W		00000000
000072 _H	I ² C バスクロックコントロールレジスタ 0	ICCR0	R/W		XX0XXXXX
000073 _H	I ² C バスアドレスレジスタ 0	IADR0	R/W		XXXXXXXXXX
000074 _H	I ² C バスデータレジスタ 0	IDAR0	R/W		XXXXXXXXXX
000075 _H	使用禁止				
000076 _H	I ² C バスステータスレジスタ 1	IBSR1	R	I ² C バスインタフェース ch.1	00000000
000077 _H	I ² C バスコントロールレジスタ 1	IBCR1	R/W		00000000
000078 _H	I ² C バスクロックコントロールレジスタ 1	ICCR1	R/W		XX0XXXXX
000079 _H	I ² C バスアドレスレジスタ 1	IADR1	R/W		XXXXXXXXXX
00007A _H	I ² C バスデータレジスタ 1	IDAR1	R/W		XXXXXXXXXX
00007B _H	使用禁止				
00007C _H	I ² C バスステータスレジスタ 2	IBSR2	R	I ² C バスインタフェース ch.2	00000000
00007D _H	I ² C バスコントロールレジスタ 2	IBCR2	R/W		00000000
00007E _H	I ² C バスクロックコントロールレジスタ 2	ICCR2	R/W		XX0XXXXX
00007F _H	I ² C バスアドレスレジスタ 2	IADR2	R/W		XXXXXXXXXX
000080 _H	I ² C バスデータレジスタ 2	IDAR2	R/W		XXXXXXXXXX

表 A-1 I/O マップ (4/7)

アドレス	レジスタ	略称	アクセス	リソース	初期値
000081 _H ~ 000085 _H	使用禁止				
000086 _H	タイマカウンタデータレジスタ下位	TCDT	R/W	16 ビット フリーランタイム	00000000
000087 _H	タイマカウンタデータレジスタ上位		R/W		00000000
000088 _H	タイマコントロールステータスレジスタ下位	TCCS	R/W		00000000
000089 _H	タイマコントロールステータスレジスタ上位		R/W		0XX00000
00008A _H	コンペアクリアレジスタ下位	CPCLR	R/W		XXXXXXXX
00008B _H	コンペアクリアレジスタ上位		R/W		XXXXXXXX
00008C _H ~ 00009A _H	使用禁止				
00009B _H	DMA ディスクリプタチャンネル指定レジスタ	DCSR	R/W	μDMAC	00000000
00009C _H	DMA ステータスレジスタ下位	DSRL	R/W		00000000
00009D _H	DMA ステータスレジスタ上位	DSRH	R/W		00000000
00009E _H	プログラムアドレス検出 コントロール ステータスレジスタ	PACSR	R/W	アドレス一致検出	00000000
00009F _H	遅延割込み要因発生 / 解除レジスタ	DIRR	R/W	遅延割込み	-----0
0000A0 _H	低消費電力モード制御レジスタ	LPMCR	R/W,W	低消費電力	00011000
0000A1 _H	クロック選択レジスタ	CKSCR	R/W,R	クロック	11111100
0000A2 _H , 0000A3 _H	使用禁止				
0000A4 _H	DMA ストップステータスレジスタ	DSSR	R/W	μDMAC	00000000
0000A5 _H	自動レディ機能選択レジスタ	ARSR	W	外部端子	0011--00
0000A6 _H	外部アドレス出力制御レジスタ	HACR	W		*****
0000A7 _H	バス制御信号選択レジスタ	EPCR	W		1000*10-
0000A8 _H	ウォッチドッグタイマ制御レジスタ	WDTC	R, W	ウォッチ ドッグタイマ	X-XXX111
0000A9 _H	タイムベースタイマ制御レジスタ	TBTC	R/W,W	タイムベース タイマ	1--00100
0000AA _H	時計タイマ制御レジスタ	WTC	R/W,R	時計タイマ	10001000
0000AB _H	使用禁止				
0000AC _H	DMA 許可レジスタ下位	DERL	R/W	μDMAC	00000000
0000AD _H	DMA 許可レジスタ上位	DERH	R/W		00000000
0000AE _H	フラッシュメモリコントロールステータスレジスタ	FMCS	R/W,R,W	フラッシュメモリ I/F	000X0000
0000AF _H	使用禁止				
0000B0 _H	割込み制御レジスタ 00	ICR00	R/W	割込みコントローラ	00000111
0000B1 _H	割込み制御レジスタ 01	ICR01	R/W		00000111
0000B2 _H	割込み制御レジスタ 02	ICR02	R/W		00000111
0000B3 _H	割込み制御レジスタ 03	ICR03	R/W		00000111
0000B4 _H	割込み制御レジスタ 04	ICR04	R/W		00000111
0000B5 _H	割込み制御レジスタ 05	ICR05	R/W		00000111
0000B6 _H	割込み制御レジスタ 06	ICR06	R/W		00000111
0000B7 _H	割込み制御レジスタ 07	ICR07	R/W		00000111
0000B8 _H	割込み制御レジスタ 08	ICR08	R/W		00000111
0000B9 _H	割込み制御レジスタ 09	ICR09	R/W		00000111
0000BA _H	割込み制御レジスタ 10	ICR10	R/W		00000111
0000BB _H	割込み制御レジスタ 11	ICR11	R/W		00000111
0000BC _H	割込み制御レジスタ 12	ICR12	R/W		00000111
0000BD _H	割込み制御レジスタ 13	ICR13	R/W		00000111
0000BE _H	割込み制御レジスタ 14	ICR14	R/W		00000111
0000BF _H	割込み制御レジスタ 15	ICR15	R/W		00000111

表 A-1 I/O マップ (5 / 7)

アドレス	レジスタ	略称	アクセス	リソース	初期値
0000C0 _H	ホストコントロールレジスタ 0	HCNT0	R/W	USB ホスト	00000000
0000C1 _H	ホストコントロールレジスタ 1	HCNT1	R/W		00000001
0000C2 _H	ホスト割込みレジスタ	HIRQ	R/W		00000000
0000C3 _H	ホストエラーステータスレジスタ	HERR	R/W		00000011
0000C4 _H	ホスト状態ステータスレジスタ	HSTATE	R/W,R		XX010010
0000C5 _H	SOF 割込み FRAME 比較レジスタ	HFCOMP	R/W		00000000
0000C6 _H	リトライタイム設定レジスタ	HRTIMER	R/W		00000000
0000C7 _H					00000000
0000C8 _H					XXXXXXXX00
0000C9 _H	ホストアドレスレジスタ	HADR	R/W		X0000000
0000CA _H	EOF 設定レジスタ	HEOF	R/W		00000000
0000CB _H					XX000000
0000CC _H	FRAME 設定レジスタ	HFRAME	R/W		00000000
0000CD _H					XXXXXX00
0000CE _H	ホストトークンエンドポイントレジスタ	HTOKEN	R/W		00000000
0000CF _H	使用禁止				
0000D0 _H	UDC 制御レジスタ	UDCC	R/W	USB ファンクション	10100000
0000D1 _H					00000000
0000D2 _H	EP0 制御レジスタ	EP0C	R/W		01000000
0000D3 _H			R/W		XXXXX000
0000D4 _H	EP1 制御レジスタ	EP1C	R/W		00000000
0000D5 _H			R/W		01100001
0000D6 _H	EP2 制御レジスタ	EP2C	R/W		01000000
0000D7 _H			R/W		01100000
0000D8 _H	EP3 制御レジスタ	EP3C	R/W		01000000
0000D9 _H			R/W		01100000
0000DA _H	EP4 制御レジスタ	EP4C	R/W		01000000
0000DB _H			R/W		01100000
0000DC _H	EP5 制御レジスタ	EP5C	R/W		01000000
0000DD _H			R/W		01100000
0000DE _H	タイムスタンプレジスタ	TMSP	R		00000000
0000DF _H			R		XXXXXX00
0000E0 _H	UDC ステータスレジスタ	UDCS	R/W		XX000000
0000E1 _H	UDC 割込み許可レジスタ	UDCIE	R/W,R		00000000
0000E2 _H	EP0I ステータスレジスタ	EP0IS	R/W		XXXXXXXXXX
0000E3 _H			R/W		10XXX1XX
0000E4 _H	EP0O ステータスレジスタ	EP0OS	R/W,R		0XXXXXXXXX
0000E5 _H			R/W		100XX000
0000E6 _H	EP1 ステータスレジスタ	EP1S	R		XXXXXXXXXX
0000E7 _H			R/W,R		1000000X
0000E8 _H	EP2 ステータスレジスタ	EP2S	R		0XXXXXXXXX
0000E9 _H			R/W,R		10000000
0000EA _H	EP3 ステータスレジスタ	EP3S	R		0XXXXXXXXX
0000EB _H			R/W,R		10000000
0000EC _H	EP4 ステータスレジスタ	EP4S	R		0XXXXXXXXX
0000ED _H			R/W,R		10000000
0000EE _H	EP5 ステータスレジスタ	EP5S	R		0XXXXXXXXX
0000EF _H			R/W,R		10000000

表 A-1 I/O マップ (6 / 7)

アドレス	レジスタ	略称	アクセス	リソース	初期値	
0000F0 _H	EP0 データレジスタ	EP0DT	R/W	USB ファンクション	XXXXXXXX	
0000F1 _H			R/W		XXXXXXXX	
0000F2 _H	EP1 データレジスタ	EP1DT	R/W		XXXXXXXX	
0000F3 _H			R/W		XXXXXXXX	
0000F4 _H	EP2 データレジスタ	EP2DT	R/W		XXXXXXXX	
0000F5 _H			R/W		XXXXXXXX	
0000F6 _H	EP3 データレジスタ	EP3DT	R/W		XXXXXXXX	
0000F7 _H			R/W		XXXXXXXX	
0000F8 _H	EP4 データレジスタ	EP4DT	R/W		XXXXXXXX	
0000F9 _H			R/W		XXXXXXXX	
0000FA _H	EP5 データレジスタ	EP5DT	R/W		XXXXXXXX	
0000FB _H			R/W		XXXXXXXX	
0000FC _H ~ 0000FF _H	使用禁止					
000100 _H ~ # _H	RAM 領域					
001FF0 _H	プログラムアドレス検出レジスタ ch.0 下位	PADR0	R/W	アドレス一致検出	XXXXXXXX	
001FF1 _H	プログラムアドレス検出レジスタ ch.0 中位		R/W		XXXXXXXX	
001FF2 _H	プログラムアドレス検出レジスタ ch.0 上位		R/W		XXXXXXXX	
001FF3 _H	プログラムアドレス検出レジスタ ch.1 下位	PADR1	R/W		XXXXXXXX	
001FF4 _H	プログラムアドレス検出レジスタ ch.1 中位		R/W		XXXXXXXX	
001FF5 _H	プログラムアドレス検出レジスタ ch.1 上位		R/W		XXXXXXXX	
# _H ~ 0078FF _H	未使用領域					
007900 _H	PPG リロードレジスタ下位 ch.0	PRL0	R/W	PPG ch.0	XXXXXXXX	
007901 _H	PPG リロードレジスタ上位 ch.0	PRLH0	R/W		XXXXXXXX	
007902 _H	PPG リロードレジスタ下位 ch.1	PRL1	R/W	PPG ch.1	XXXXXXXX	
007903 _H	PPG リロードレジスタ上位 ch.1	PRLH1	R/W		XXXXXXXX	
007904 _H	PPG リロードレジスタ下位 ch.2	PRL2	R/W	PPG ch.2	XXXXXXXX	
007905 _H	PPG リロードレジスタ上位 ch.2	PRLH2	R/W		XXXXXXXX	
007906 _H	PPG リロードレジスタ下位 ch.3	PRL3	R/W	PPG ch.3	XXXXXXXX	
007907 _H	PPG リロードレジスタ上位 ch.3	PRLH3	R/W		XXXXXXXX	
007908 _H	PPG リロードレジスタ下位 ch.4	PRL4	R/W	PPG ch.4	XXXXXXXX	
007909 _H	PPG リロードレジスタ上位 ch.4	PRLH4	R/W		XXXXXXXX	
00790A _H	PPG リロードレジスタ下位 ch.5	PRL5	R/W	PPG ch.5	XXXXXXXX	
00790B _H	PPG リロードレジスタ上位 ch.5	PRLH5	R/W		XXXXXXXX	
00790C _H ~ 00790F _H	使用禁止					
007910 _H	インプットキャプチャデータレジスタ下位 ch.0	IPCP0	R	インプット キャプチャ ch.0/1	XXXXXXXX	
007911 _H	インプットキャプチャデータレジスタ上位 ch.0		R		XXXXXXXX	
007912 _H	インプットキャプチャデータレジスタ下位 ch.1	IPCP1	R		XXXXXXXX	
007913 _H	インプットキャプチャデータレジスタ上位 ch.1		R		XXXXXXXX	
007914 _H	インプットキャプチャデータレジスタ下位 ch.2	IPCP2	R	インプット キャプチャ ch.2/3	XXXXXXXX	
007915 _H	インプットキャプチャデータレジスタ上位 ch.2		R		XXXXXXXX	
007916 _H	インプットキャプチャデータレジスタ下位 ch.3	IPCP3	R		XXXXXXXX	
007917 _H	インプットキャプチャデータレジスタ上位 ch.3		R		XXXXXXXX	

表 A-1 I/O マップ (7/7)

アドレス	レジスタ	略称	アクセス	リソース	初期値
007918 _H	アウトプットコンペアレジスタ下位 ch.0	OCCP0	R/W	アウトプット コンペア ch.0/1	XXXXXXXX
007919 _H	アウトプットコンペアレジスタ上位 ch.0		R/W		XXXXXXXX
00791A _H	アウトプットコンペアレジスタ下位 ch.1	OCCP1	R/W		XXXXXXXX
00791B _H	アウトプットコンペアレジスタ上位 ch.1		R/W		XXXXXXXX
00791C _H	アウトプットコンペアレジスタ下位 ch.2	OCCP2	R/W	アウトプット コンペア ch.2/3	XXXXXXXX
00791D _H	アウトプットコンペアレジスタ上位 ch.2		R/W		XXXXXXXX
00791E _H	アウトプットコンペアレジスタ下位 ch.3	OCCP3	R/W		XXXXXXXX
00791F _H	アウトプットコンペアレジスタ上位 ch.3		R/W		XXXXXXXX
007920 _H	DMA バッファアドレスポインタ 下位 8 ビット	DBAPL	R/W	μDMAC	XXXXXXXX
007921 _H	DMA バッファアドレスポインタ 中位 8 ビット	DBAPM	R/W		XXXXXXXX
007922 _H	DMA バッファアドレスポインタ 上位 8 ビット	DBAPH	R/W		XXXXXXXX
007923 _H	DMA コントロールレジスタ	DMACS	R/W		XXXXXXXX
007924 _H	DMA I/O レジスタアドレスポインタ下位 8 ビット	DIOAL	R/W		XXXXXXXX
007925 _H	DMA I/O レジスタアドレスポインタ上位 8 ビット	DIOAH	R/W		XXXXXXXX
007926 _H	DMA データカウンタ下位 8 ビット	DDCTL	R/W		XXXXXXXX
007927 _H	DMA データカウンタ上位 8 ビット	DDCTH	R/W		XXXXXXXX
007928 _H ~ 007FFF _H	使用禁止				

・ 読出し / 書込みについての説明

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。

-: 未定義ビットです。初期値は不定です。

*: 初期値は "1" または "0" です。

< 注意事項 >

7900_H ~ 7FFF_H に配置されているレジスタについては、I/O 系命令は使用できません。

■ 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 A-2 に、割り込み要因と割り込みベクタおよび割り込み制御レジスタとの対応を示します。

表 A-2 割り込み要因と割り込みベクタおよび割り込み制御レジスタとの対応

割り込み要因	EI ² OS クリア	μDMAC チャンネル番号	割り込みベクタ		割り込み制御レジスタ	
			番号	アドレス	ICR	アドレス
リセット	×	×	#08	FFFFDC _H	-	-
INT9 命令	×	×	#09	FFFFD8 _H	-	-
例外処理	×	×	#10	FFFFD4 _H	-	-
USB ファンクション 1	×	0, 1	#11	FFFFD0 _H	ICR00	0000B0 _H
USB ファンクション 2	×	2 ~ 6 *	#12	FFFFCC _H		
USB ファンクション 3	×	×	#13	FFFFC8 _H	ICR01	0000B1 _H
USB ファンクション 4	×	×	#14	FFFFC4 _H		
USB ホスト 1	×	×	#15	FFFFC0 _H	ICR02	0000B2 _H
USB ホスト 2	×	×	#16	FFFFBC _H		
I ² C ch.0	×	×	#17	FFFFB8 _H	ICR03	0000B3 _H
DTP/ 外部割り込み ch.0/ch.1		×	#18	FFFFB4 _H		
I ² C ch.1	×	×	#19	FFFFB0 _H	ICR04	0000B4 _H
DTP/ 外部割り込み ch.2/ch.3		×	#20	FFFFAC _H		
I ² C ch.2	×	×	#21	FFFFA8 _H	ICR05	0000B5 _H
DTP/ 外部割り込み ch.4/ch.5		×	#22	FFFFA4 _H		
PWC/ リロードタイマ ch.0		14	#23	FFFFA0 _H	ICR06	0000B6 _H
DTP/ 外部割り込み ch.6/7		×	#24	FFFF9C _H		
インプットキャプチャ ch.0/ch.1		7	#25	FFFF98 _H	ICR07	0000B7 _H
リロードタイマ ch.1		×	#26	FFFF94 _H		
インプットキャプチャ ch.2/ch.3		8	#27	FFFF90 _H	ICR08	0000B8 _H
リロードタイマ ch.2		×	#28	FFFF8C _H		
アウトプットコンペア ch.0/ch.1		×	#29	FFFF88 _H	ICR09	0000B9 _H
PPG ch.0/ch.1	×	×	#30	FFFF84 _H		
アウトプットコンペア ch.2/ch.3		×	#31	FFFF80 _H	ICR10	0000BA _H
PPG ch.2/ch.3	×	×	#32	FFFF7C _H		
UART 送信完了 ch.2/ch.3		11	#33	FFFF78 _H	ICR11	0000BB _H
PPG ch.4/ch.5	×	×	#34	FFFF74 _H		
UART 受信完了 ch.2/ch.3		10	#35	FFFF70 _H	ICR12	0000BC _H
AD 変換 / フリーランタイム		15	#36	FFFF6C _H		
UART 送信完了 ch.0/ch.1		13	#37	FFFF68 _H	ICR13	0000BD _H
拡張シリアル I/O	×	9	#38	FFFF64 _H		
UART 受信完了 ch.0/ch.1		12	#39	FFFF60 _H	ICR14	0000BE _H
タイムベースタイマ / 時計タイマ	×	×	#40	FFFF5C _H		
フラッシュ書き込み / 消去	×	×	#41	FFFF58 _H	ICR15	0000BF _H
遅延割り込み発生モジュール	×	×	#42	FFFF54 _H		

：使用可能。EI²OS 停止機能付（割り込みクリア信号で、割り込み要求フラグはクリアされます。ストップ要求あり）

：使用可能。（割り込みクリア信号で、割り込み要求フラグはクリアされます。）

：ICR を共有する割り込み要因を使用しない場合に使用可能

×：使用不可能

*：チャンネル 2, 3 は、USB ホスト動作時でも使用できます。

< 注意事項 >

- 同一割込み制御レジスタ (ICR) に 2 つの割込み要因がある場合は EI²OS の使用を許可した場合、どちらかの割込み要因を検出すると EI²OS が起動されます。EI²OS 起動中は起動要因以外の割込みはマスクされますので、EI²OS 使用時は、どちらかの割込み要求をマスクして使用されることを推奨します。
- 同一割込み制御レジスタ (ICR) に 2 つの割込み要因があるリソースは、割込みフラグが EI²OS 割込みクリア信号でクリアされます。
- 同一割込み番号に 2 つの割込み要因があった場合は、リソースは両方の割込み要求フラグが μ DMAC 割込みクリア信号でクリアされます。したがって、2 つの要因のどちらか 1 つを DMAC 機能に使用すると、もう 1 つの割込み機能は使用できません。該当リソースの割込み要求許可ビットを "0" にして、ソフトウェアポーリング処理で対処してください。

■ USB 割込みの種類と機能

USB 割込み要因	詳細内容
USB ファンクション 1	End Point0-IN EndPoint0-OUT
USB ファンクション 2	End Point1-5 *
USB ファンクション 3	SUP SOF BRST WKUP CONF
USB ファンクション 4	SPK
USB ホスト 1	DIRQ CNNIRQ URIEQ RWKIRQ
USB ホスト 2	SOFIRQ CMPIRQ

* : End Point 1, 2 は、USB ホスト動作時でも使用できます。

付録 B 命令

F²MC-16LX に使用している命令について説明します。

- B.1 命令の種類
- B.2 アドレッシング
- B.3 直接アドレッシング
- B.4 間接アドレッシング
- B.5 実行サイクル数
- B.6 実効アドレスフィールド
- B.7 命令一覧表の読み方
- B.8 F²MC-16LX 命令一覧表
- B.9 命令マップ

B.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

B.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 B.2-1 に示します。

表 B.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

B.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 B.3-1 に例を示します。

図 B.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 B.3-1 に示します。

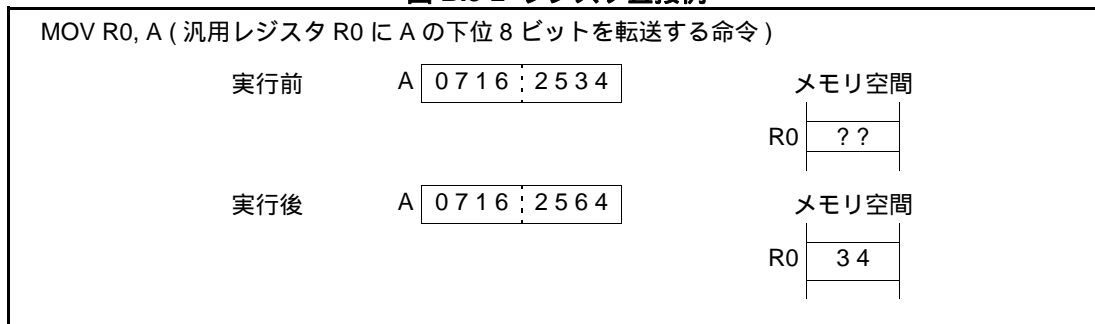
表 B.3-1 レジスタ直接

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 B.3-2 に例を示します。

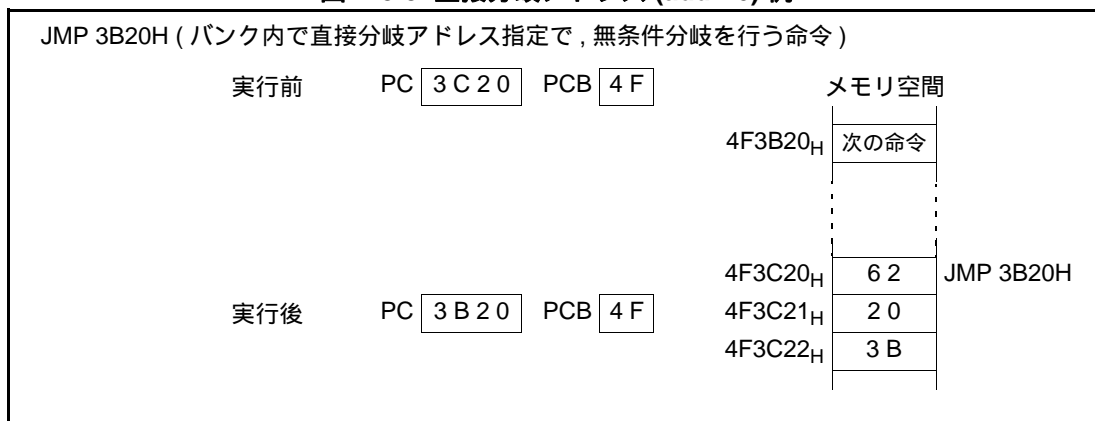
図 B.3-2 レジスタ直接例



● 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 B.3-3 に例を示します。

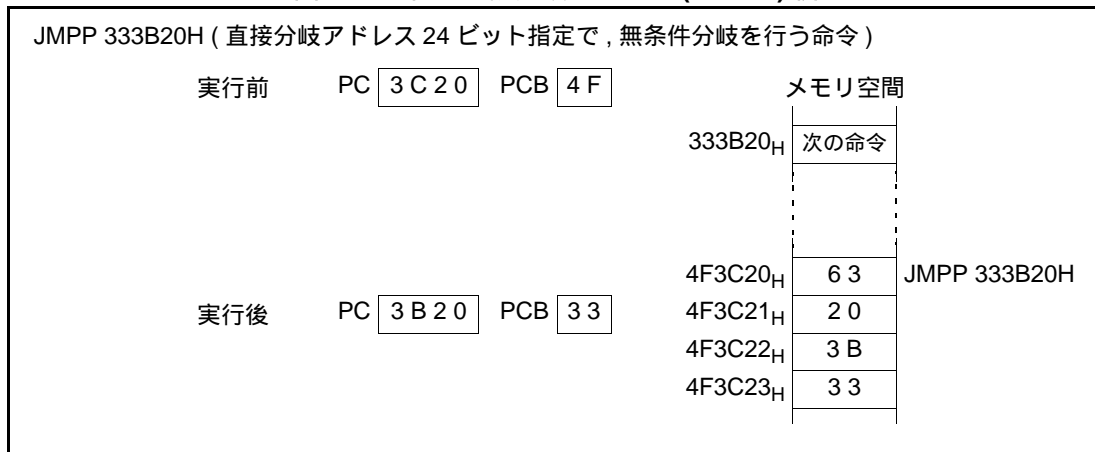
図 B.3-3 直接分岐アドレス (addr16) 例



● 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 B.3-4 に例を示します。

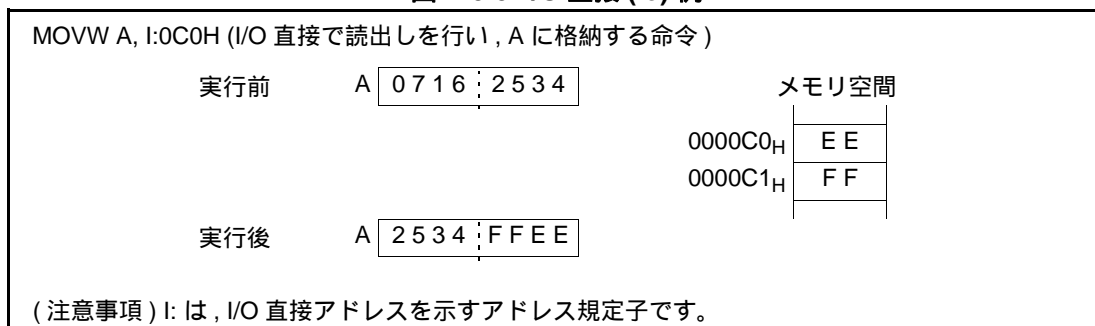
図 B.3-4 物理直接分岐アドレス (addr24) 例



● 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 B.3-5 に例を示します。

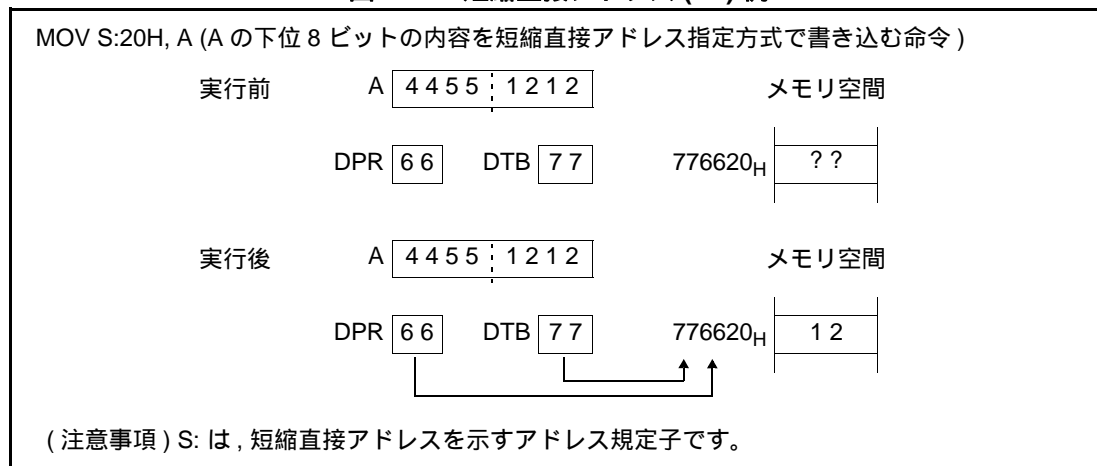
図 B.3-5 I/O 直接 (io) 例



● 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 B.3-6 に例を示します。

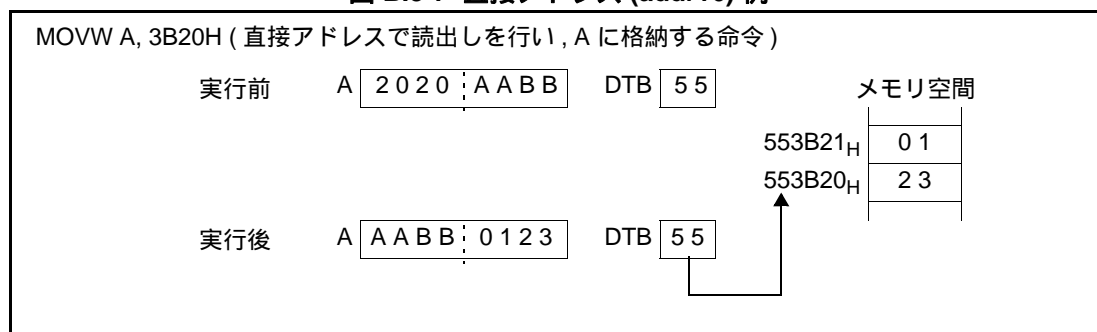
図 B.3-6 短縮直接アドレス (dir) 例



● 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 B.3-7 に例を示します。

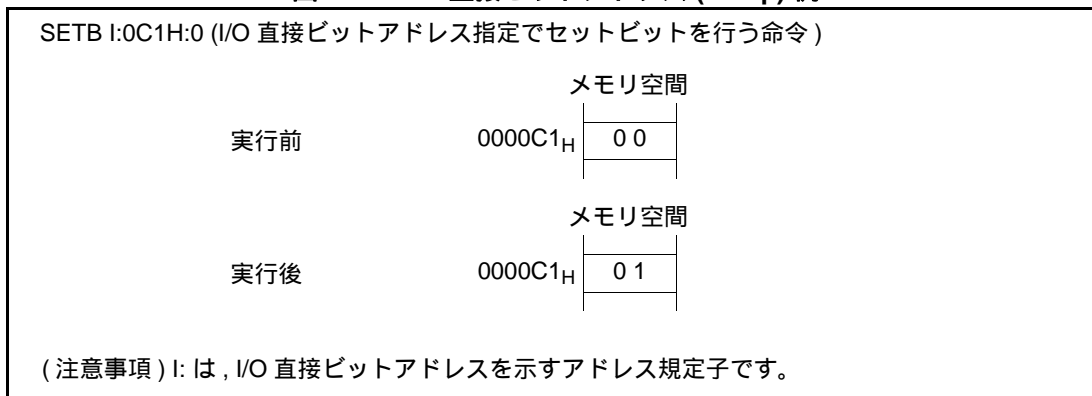
図 B.3-7 直接アドレス (addr16) 例



● 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 B.3-8 に例を示します。

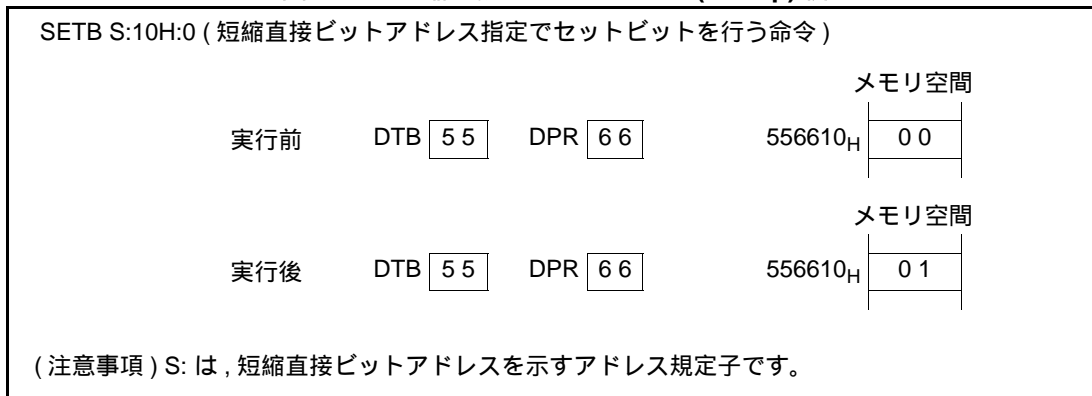
図 B.3-8 I/O 直接ビットアドレス (io: bp) 例



● 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 B.3-9 に例を示します。

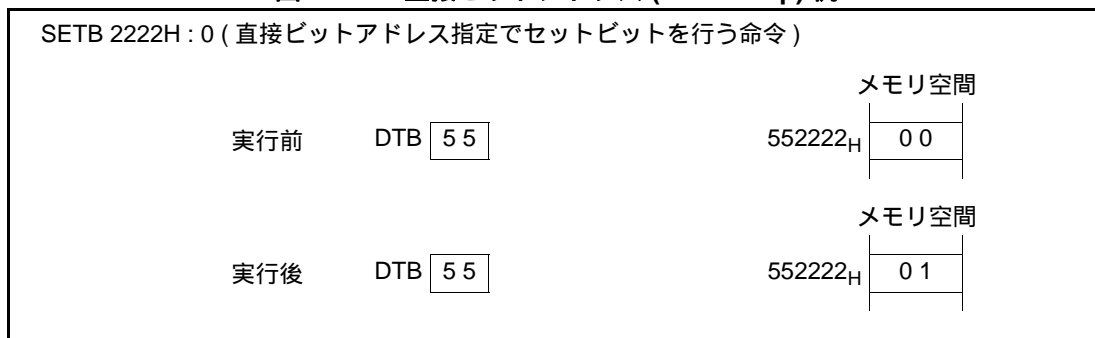
図 B.3-9 短縮直接ビットアドレス (dir: bp) 例



● 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 B.3-10 に例を示します。

図 B.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 B.3-11 に例を示します。

図 B.3-11 ベクタアドレス (#vct) 例

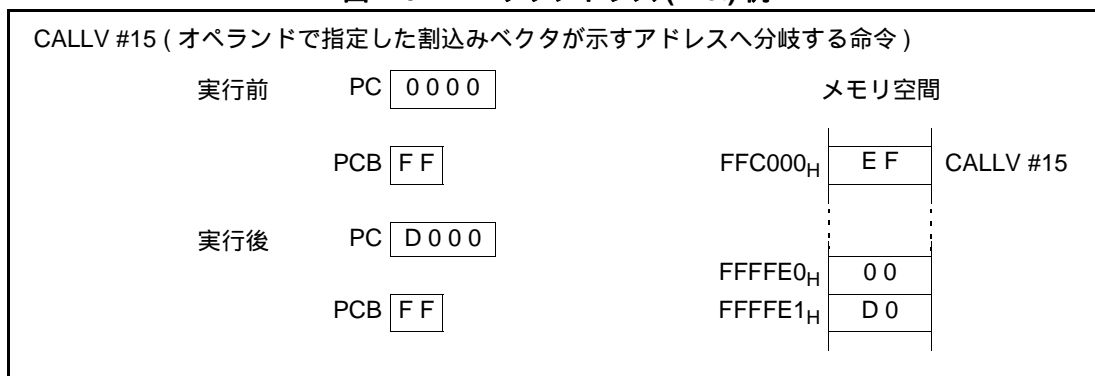


表 B.3-2 CALLV ベクター一覧表

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 B.3-2 を参照してください)

B.4 間接アドレッシング

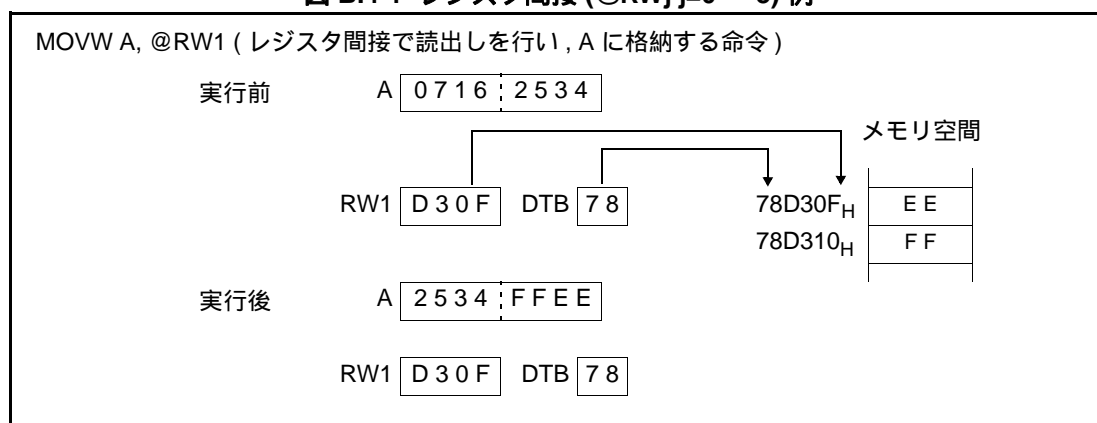
間接アドレッシングでは、記述したオペランドが示すアドレスのデータで、間接的にアドレスを指定します。

■ 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 B.4-1 に例を示します。

図 B.4-1 レジスタ間接 (@RWj j=0 ~ 3) 例



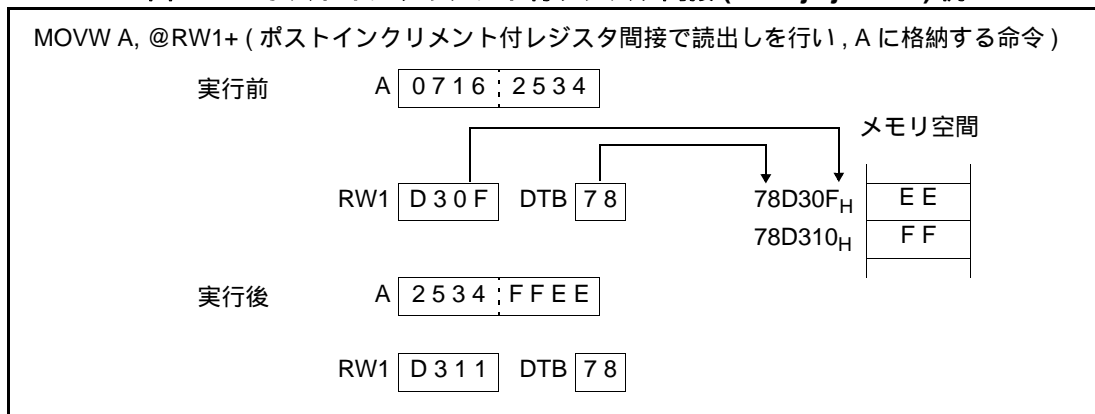
● ポストインクリメント付レジスタ間接 (@RWj+ j=0 ~ 3)

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、命令が書込みだった場合は命令による書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込みデータとなります。

図 B.4-2 に例を示します。

図 B.4-2 ポストインクリメント付レジスタ間接 (@RWj+ j=0 ~ 3) 例

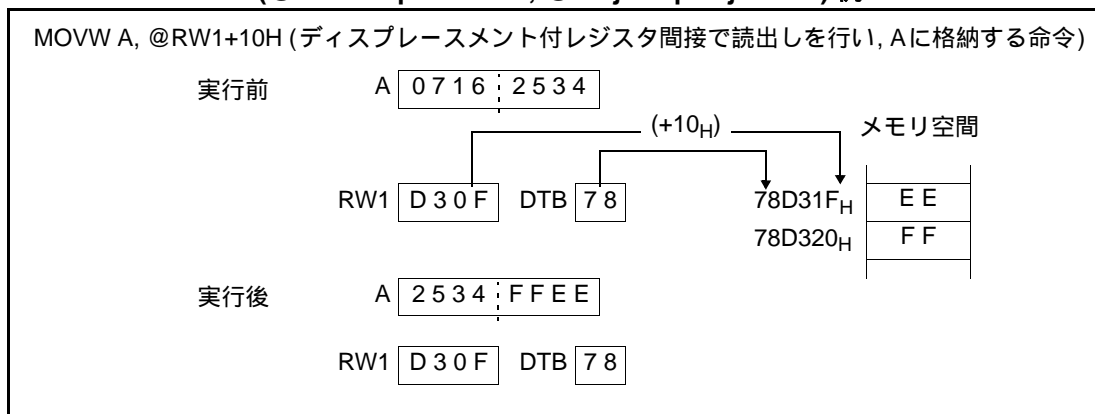


● ディスプレースメント付レジスタ間接

(@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 B.4-3 に例を示します。

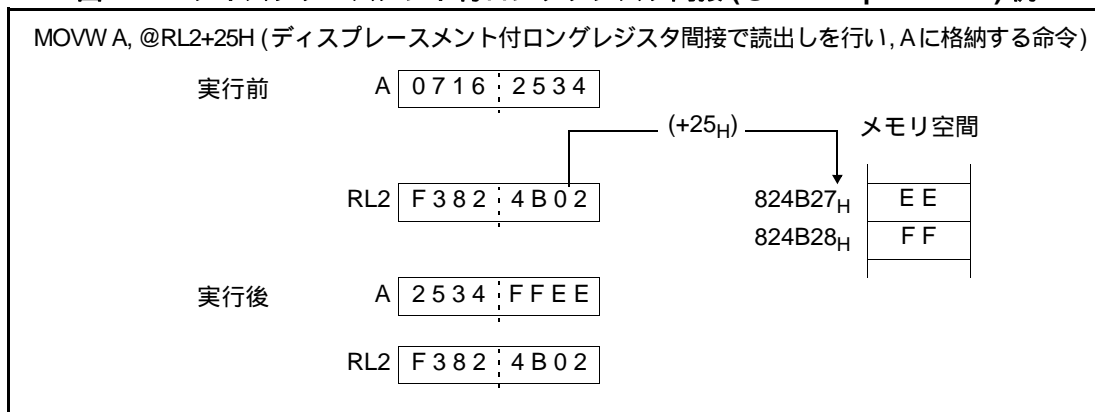
図 B.4-3 ディスプレースメント付レジスタ間接
(@RWi+disp8 i=0 ~ 7, @RWj+disp16 j=0 ~ 3) 例



● 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 B.4-4 に例を示します。

図 B.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



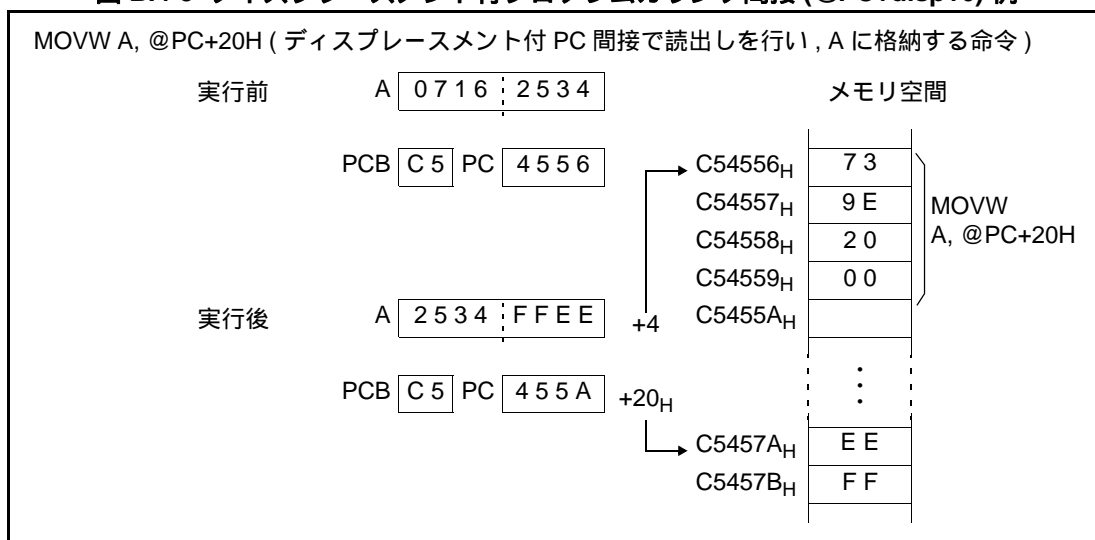
● ディスプレースメント付プログラムカウンタ間接 (@PC+disp16)

(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 B.4-5 に例を示します。

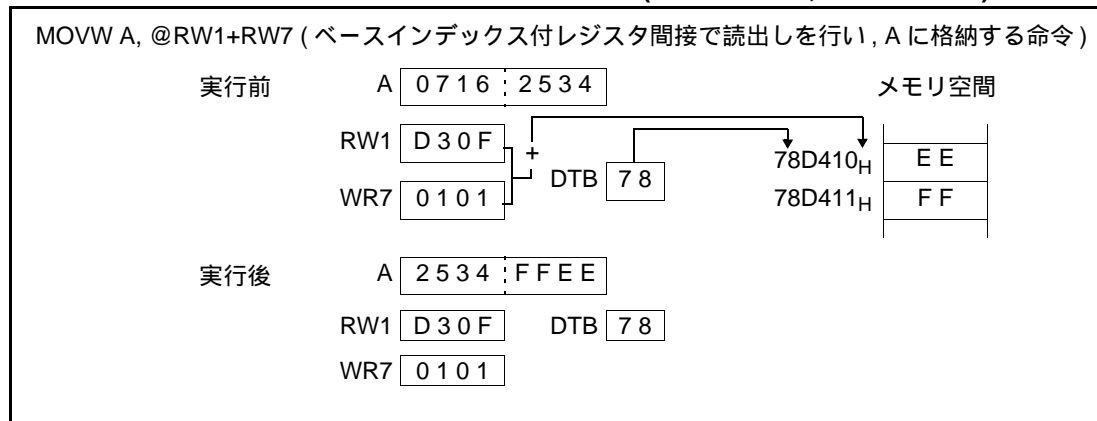
図 B.4-5 ディスプレースメント付プログラムカウンタ間接 (@PC+disp16) 例



● 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 B.4-6 に例を示します。

図 B.4-6 ベースインデックス付レジスタ間接 (@RW0+RW7, @RW1+RW7) 例

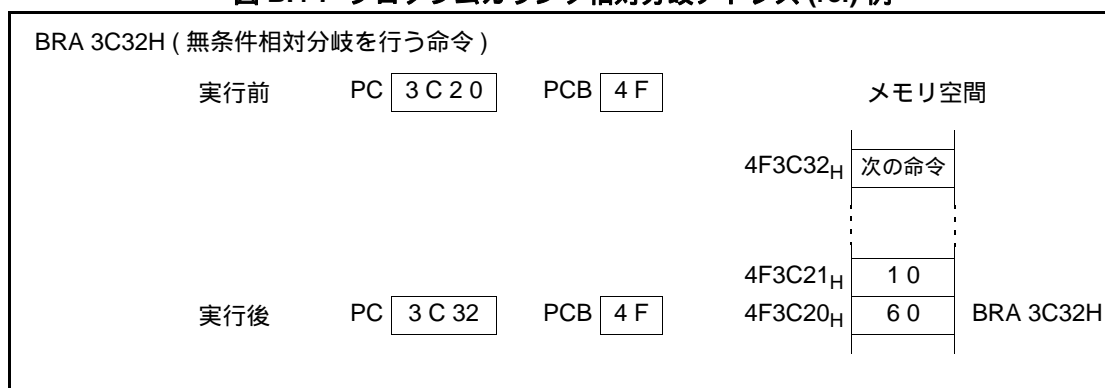


● プログラムカウンタ相対分岐アドレス (rel)

分岐先のアドレスはプログラムカウンタ (PC) の値と 8 ビットのオフセッ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リメントをせずに, 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 B.4-7 に例を示します。

図 B.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 B.4-8 にレジスタリストの構成、図 B.4-9 に例を示します。

図 B.4-8 レジスタリストの構成

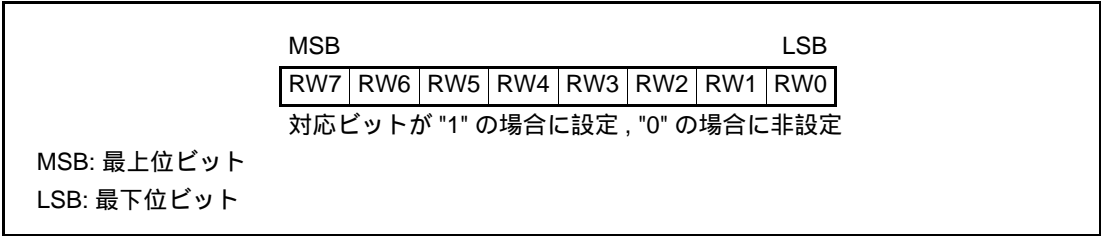
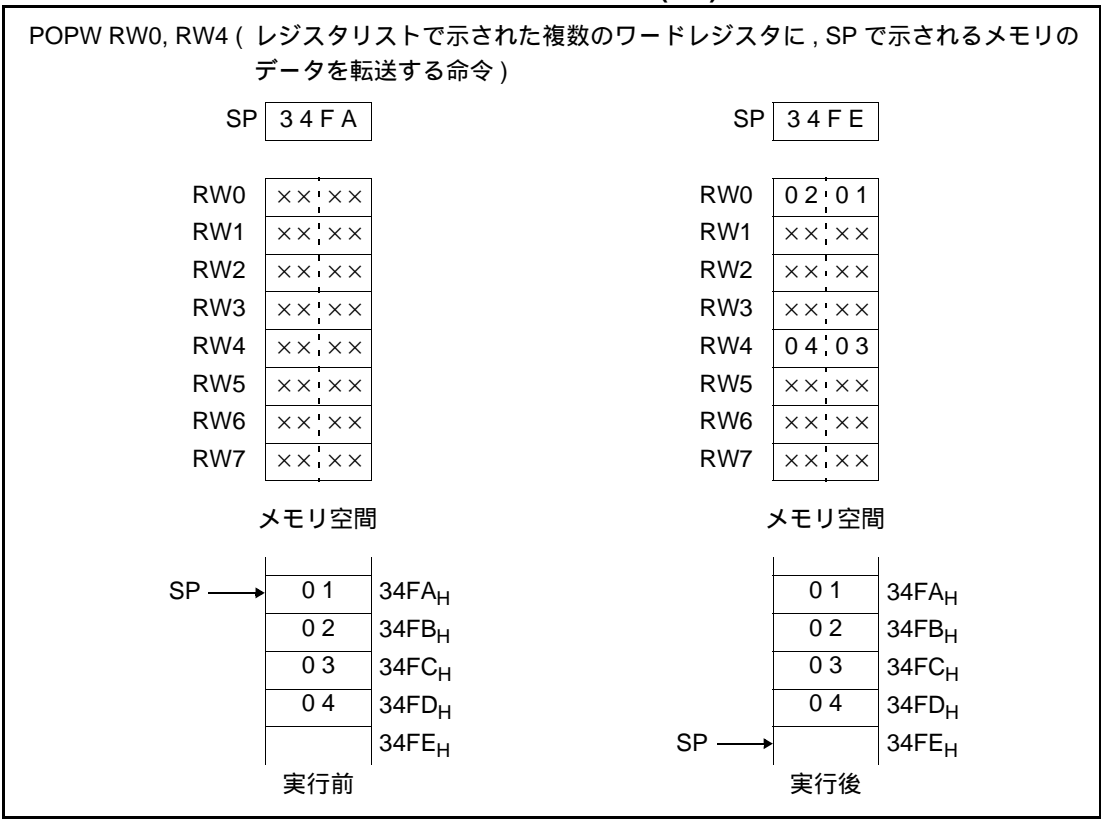


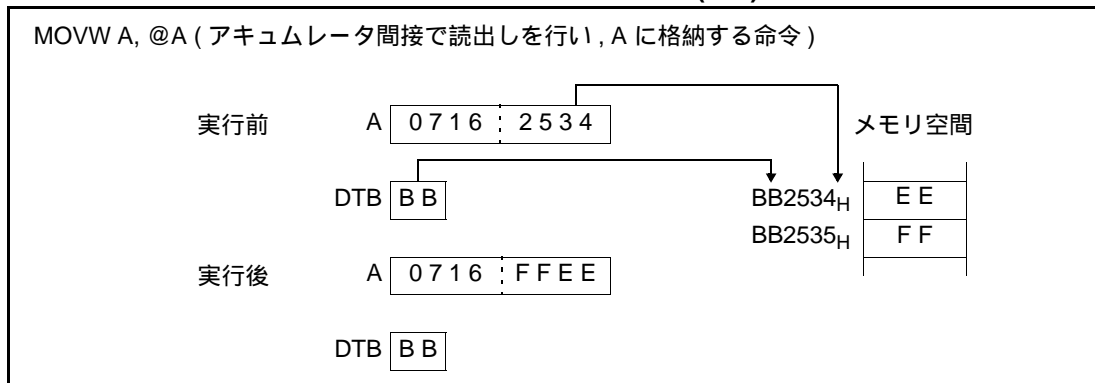
図 B.4-9 レジスタリスト (rlst) 例



● 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 B.4-10 に例を示します。

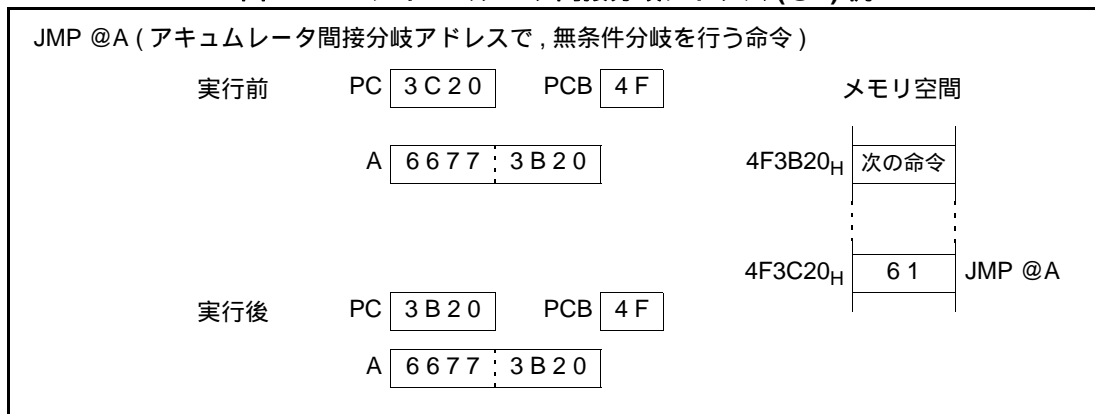
図 B.4-10 アキュムレータ間接 (@A) 例



● 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 B.4-11 に例を示します。

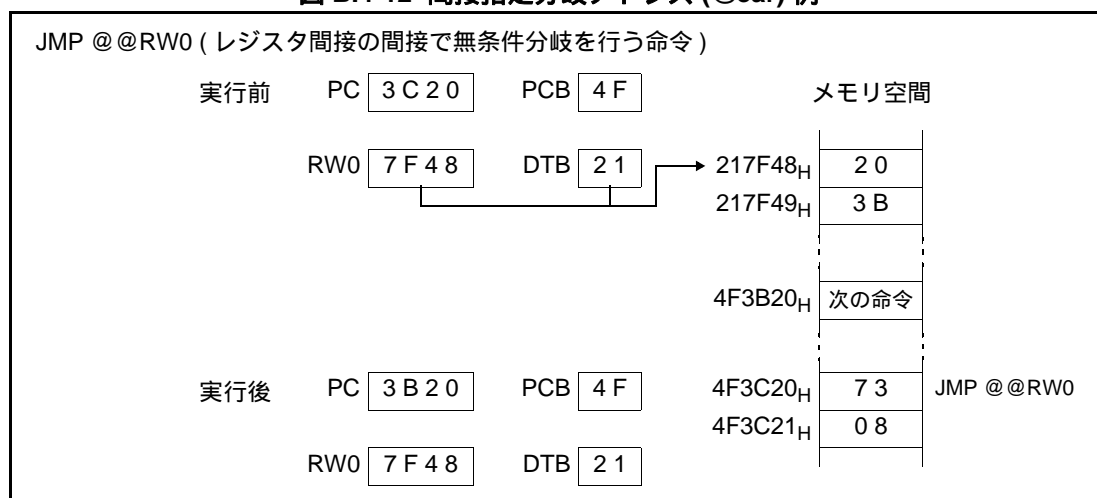
図 B.4-11 アキュムレータ間接分岐アドレス (@A) 例



● 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 B.4-12 に例を示します。

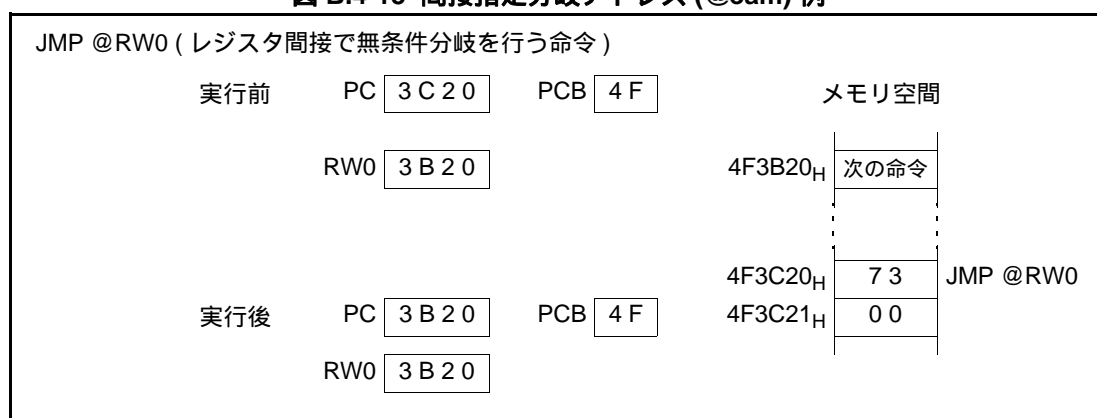
図 B.4-12 間接指定分岐アドレス (@ear) 例



● 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 B.4-13 に例を示します。

図 B.4-13 間接指定分岐アドレス (@eam) 例



B.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 B.5-1、表 B.5-2、表 B.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 B.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「B.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「B.7 命令一覧表の読み方」を参照してください。

表 B.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「B.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 B.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。

B.6 実効アドレスフィールド

表 B.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 B.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 B.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、「B.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「B.7 命令一覧表の読み方」を参照してください。

B.7 命令一覧表の読み方

「B.8 F2MC-16LX 命令一覧表」で使用している項目の説明を表 B.7-1 に、記号の説明を表 B.7-2 に示します。

■ 命令の表示記号の説明

表 B.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 B.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 B.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 _H ~ 07 _H)
eam	実効アドレス指定 (コード 08 _H ~ 1F _H)
rlst	レジスタ並び

B.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 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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込み (word)

注 2) W : スタックへの退避 (word), R : 分岐先アドレスの読み込み (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込み (long)

注 5) スタックへの退避 (long)

(注意事項) 表中の (a) ~ (d) は , 表 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.5-1 および表 B.5-2 を参照してください。

表 B.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 B.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 B.5-1 および表 B.5-2 を参照してください。

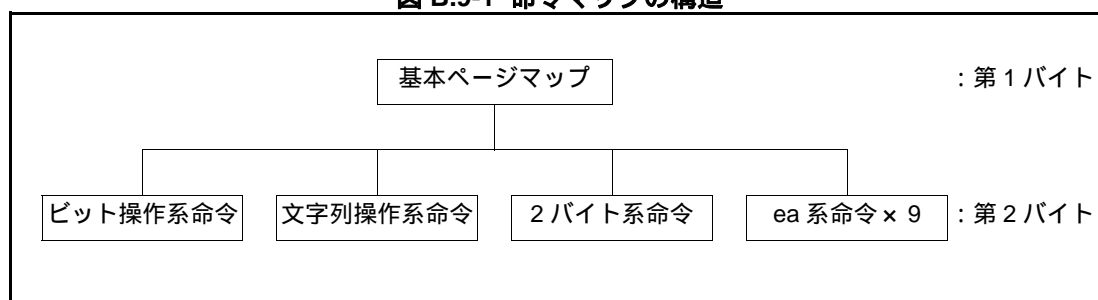
B.9 命令マップ

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

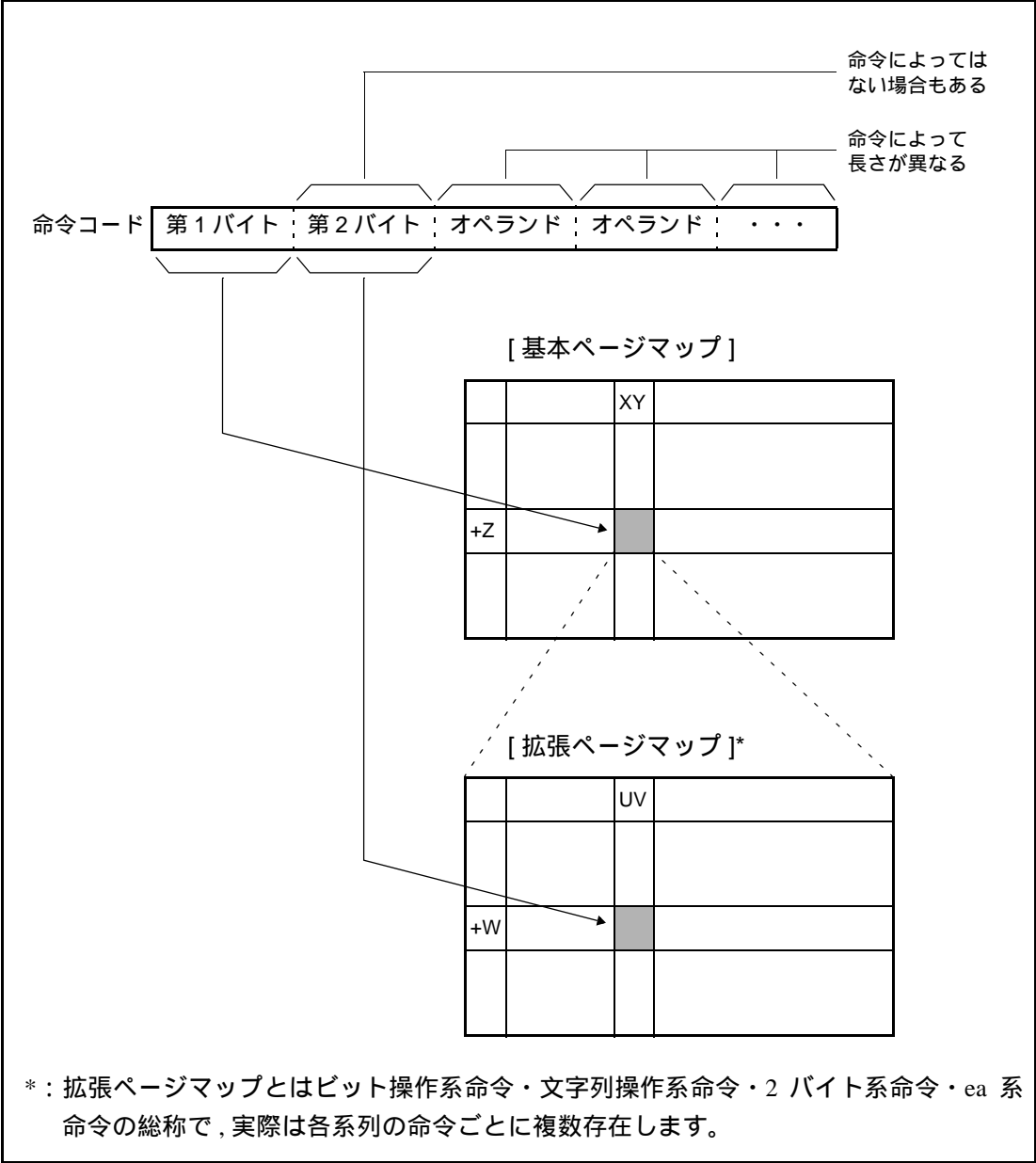
図 B.9-1 命令マップの構造



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 B.9-2 に示します。

図 B.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 B.9-1 に示します。

表 B.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
CBNE @RW2+d8, #8, rel	70 + 0=70	F0 + 2=F2

表 B.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	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	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	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BNZ/BNE rel
+2	ADDCA	SUBDC A	ADDC A	SUBC A	MOV A, #8	MOV A, addr16	JMP addr16	ea 系命令 その 3	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	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	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	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	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	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	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BP rel
+6	ADB	SWAP	DIVU A	XOR A, #8	MOVW A, SP	MOVW io, #16	RETP	ea 系命令 その 7	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BV rel
+7	SPB	ADDSP #8	MULU A	NOT A	MOVW SP, A	MOVX A, addr16	RET	ea 系命令 その 8	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	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	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BT rel
+9	UNLINK	SUBL A, #32	SUBW A	SUBW A, #16	MOVW dir, A	MOVW io, A	INT	MOVEA RWI, ea	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	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	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BLT rel
+B	NEGW A	CMPL A, #32	CMPW A	CMPW A, #16	MOVL A, #32	MOVW addr16, A	RETI	MOVW RWI, ea	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	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	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BLE rel
+D		ZEXTW	ORW A	ORW A, #16	PUSHW AH	POPW AH		MOVW ea, RWI	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BGT rel
+E	ASRW A	SWAPW A	XORW A	XORW A, #16	PUSHW PS	POPW PS	文字列操 作系命令	XCH Ri, ea	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BLS rel
+F	LSRW A	ADDSP A	MULW A	NOTW A	PUSHW r1st	POPW r1st	2 バイト 系命令	XCHW RWI, ea	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOVX A, Ri	MOVX A, @RWI+dB	MOVN A, #4	CALL #vct4	BHI rel

表 B.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															
+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 B.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		FILSWI PCB	
+1	PCB, DTB								DTB	DTB	DTB	DTB	DTB		DTB	
+2	PCB, ADB								ADB	ADB	ADB	ADB	ADB		ADB	
+3	PCB, SPB								SPB	SPB	SPB	SPB	SPB		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 B.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			DIV 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 B.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	#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	#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	#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	#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	#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	#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	#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	#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	#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	#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	#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	#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	#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	@RW1+RW7 #8, rel	#8, rel
+E	ADDL A, A, @RW2+PC+d16	SUBL A, A, @RW2+PC+d16	SUBL A, A, @RW2+PC+d16	SUBL A, A, @RW2+PC+d16	@RW2+PC+d16 #16, rel	CMPL A, A, @RW2+PC+d16	CMPL A, A, @RW2+PC+d16	CMPL A, A, @RW2+PC+d16	ANDL A, A, @RW2+PC+d16	ANDL A, A, @RW2+PC+d16	ORL A, A, @RW2+PC+d16	ORL A, A, @RW2+PC+d16	XORL A, A, @RW2+PC+d16	XORL A, A, @RW2+PC+d16	@PC+d16 #8, rel	#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	addr16 #8, rel	addr16, #8, rel

表 B.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 RL0, A, @RW0+d8	MOVL RL0, A, @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 RL0, A, @RW1+d8	MOVL RL0, A, @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 RL1, A, @RW2+d8	MOVL RL1, A, @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 RL1, A, @RW3+d8	MOVL RL1, A, @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 RL2, A, @RW4+d8	MOVL RL2, A, @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 RL2, A, @RW5+d8	MOVL RL2, A, @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 RL3, A, @RW6+d8	MOVL RL3, A, @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 RL3, A, @RW7+d8	MOVL RL3, A, @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 @RW0, A, @RW0+d16	MOVL @RW0, A, @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 @RW1, A, @RW1+d16	MOVL @RW1, A, @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 @RW2, A, @RW2+d16	MOVL @RW2, A, @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 @RW3, A, @RW3+d16	MOVL @RW3, A, @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 @RW0+, A, @RW0+RW7	MOVL @RW0+, A, @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 @RW1+, A, @RW1+RW7	MOVL @RW1+, A, @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 @RW2+, A, @PC+d16, A	MOVL @RW2+, A, @PC+d16, A	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 @RW3+, A, @addr16, A	MOVL @RW3+, A, @addr16, A	MOV @RW3+, #8, @addr16, #8	MOV @RW3+, #8, @addr16, #8	MOVEA A, @RW3+, @addr16	MOVEA A, @RW3+, @addr16

表 B.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 B.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 @PC+, @PC+d16	JMP @PC+, @PC+d16	CALL @PC+, @PC+d16	CALL @PC+, @PC+d16	INCW @PC+, @PC+d16	INCW @PC+, @PC+d16	DECW @PC+, @PC+d16	DECW @PC+, @PC+d16	MOVW A, @PC+, @PC+d16	MOVW A, @PC+, @PC+d16	MOVW @PC+, #16, @PC+d16, #16	MOVW @PC+, #16, @PC+d16, #16	MOVW @PC+, #16, @PC+d16, #16	MOVW @PC+, #16, @PC+d16, #16	XCHW A, @PC+, @PC+d16	XCHW A, @PC+, @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 B.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	ADD 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, rel, +d8, rel	DBNZ R0, rel, +d8, rel
+1	ADD A, R1, @RW1+d8	ADD 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, rel, +d8, rel	DBNZ R1, rel, +d8, rel
+2	ADD A, R2, @RW2+d8	ADD 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, rel, +d8, rel	DBNZ R2, rel, +d8, rel
+3	ADD A, R3, @RW3+d8	ADD 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, rel, +d8, rel	DBNZ R3, rel, +d8, rel
+4	ADD A, R4, @RW4+d8	ADD 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, rel, +d8, rel	DBNZ R4, rel, +d8, rel
+5	ADD A, R5, @RW5+d8	ADD 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, rel, +d8, rel	DBNZ R5, rel, +d8, rel
+6	ADD A, R6, @RW6+d8	ADD 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, rel, +d8, rel	DBNZ R6, rel, +d8, rel
+7	ADD A, R7, @RW7+d8	ADD 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, rel, +d8, rel	DBNZ R7, rel, +d8, rel
+8	ADD A, @RW0, @RW0+d16	ADD 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 @RW0, rel, +d16, rel	DBNZ @RW0, rel, +d16, rel
+9	ADD A, @RW1, @RW1+d16	ADD 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 @RW1, rel, +d16, rel	DBNZ @RW1, rel, +d16, rel
+A	ADD A, @RW2, @RW2+d16	ADD 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 @RW2, rel, +d16, rel	DBNZ @RW2, rel, +d16, rel
+B	ADD A, @RW3, @RW3+d16	ADD 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 @RW3, rel, +d16, rel	DBNZ @RW3, rel, +d16, rel
+C	ADD A, @RW0+, @RW0+RW7	ADD 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 @RW0+, rel, +RW7, rel	DBNZ @RW0+, rel, +RW7, rel
+D	ADD A, @RW1+, @RW1+RW7	ADD 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 @RW1+, rel, +RW7, rel	DBNZ @RW1+, rel, +RW7, rel
+E	ADD A, @RW2+, @PC+d16	ADD 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 @RW2+, rel, +d16, rel	DBNZ @RW2+, rel, +d16, rel
+F	ADD A, @RW3+, A, addr16	ADD 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 @RW3+, rel, +addr16, rel	DBNZ @RW3+, rel, +addr16, rel

表 B.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	SUBC A, R0', @RW0+d8	NEG R0', @RW0+d8	NEG R0', @RW0+d8	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	NOT R0', @RW0+d8
+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	SUBC A, R1', @RW1+d8	NEG R1', @RW1+d8	NEG R1', @RW1+d8	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	NOT R1', @RW1+d8
+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	SUBC A, R2', @RW2+d8	NEG R2', @RW2+d8	NEG R2', @RW2+d8	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	NOT R2', @RW2+d8
+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	SUBC A, R3', @RW3+d8	NEG R3', @RW3+d8	NEG R3', @RW3+d8	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	NOT R3', @RW3+d8
+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	SUBC A, R4', @RW4+d8	NEG R4', @RW4+d8	NEG R4', @RW4+d8	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	NOT R4', @RW4+d8
+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	SUBC A, R5', @RW5+d8	NEG R5', @RW5+d8	NEG R5', @RW5+d8	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	NOT R5', @RW5+d8
+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	SUBC A, R6', @RW6+d8	NEG R6', @RW6+d8	NEG R6', @RW6+d8	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	NOT R6', @RW6+d8
+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	SUBC A, R7', @RW7+d8	NEG R7', @RW7+d8	NEG R7', @RW7+d8	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	NOT R7', @RW7+d8
+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	SUBC A, @RW0', @RW0+d16	NEG @RW0', @RW0+d16	NEG @RW0', @RW0+d16	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	NOT @RW0', @RW0+d16
+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	SUBC A, @RW1', @RW1+d16	NEG @RW1', @RW1+d16	NEG @RW1', @RW1+d16	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	NOT @RW1', @RW1+d16
+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	SUBC A, @RW2', @RW2+d16	NEG @RW2', @RW2+d16	NEG @RW2', @RW2+d16	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	NOT @RW2', @RW2+d16
+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	SUBC A, @RW3', @RW3+d16	NEG @RW3', @RW3+d16	NEG @RW3', @RW3+d16	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	NOT @RW3', @RW3+d16
+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	SUBC A, @RW0+', @RW0+RW7	NEG @RW0+', @RW0+RW7	NEG @RW0+', @RW0+RW7	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	NOT @RW0+', @RW0+RW7
+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	SUBC A, @RW1+', @RW1+RW7	NEG @RW1+', @RW1+RW7	NEG @RW1+', @RW1+RW7	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	NOT @RW1+', @RW1+RW7
+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	SUBC A, @RW2+', @PC+d16	NEG @RW2+', @PC+d16	NEG @RW2+', @PC+d16	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	NOT @RW2+', @PC+d16
+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	SUBC A, @RW3+', addr16	NEG @RW3+', addr16	NEG @RW3+', addr16	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	NOT @RW3+', addr16

表 B.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	ADDW 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, rel, @RW0+d8, rel	DWBZ RW0, rel, @RW0+d8, rel
+1	ADDW A, RW1, @RW1+d8	ADDW 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, rel, @RW1+d8, rel	DWBZ RW1, rel, @RW1+d8, rel
+2	ADDW A, RW2, @RW2+d8	ADDW 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, rel, @RW2+d8, rel	DWBZ RW2, rel, @RW2+d8, rel
+3	ADDW A, RW3, @RW3+d8	ADDW 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, rel, @RW3+d8, rel	DWBZ RW3, rel, @RW3+d8, rel
+4	ADDW A, RW4, @RW4+d8	ADDW 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, rel, @RW4+d8, rel	DWBZ RW4, rel, @RW4+d8, rel
+5	ADDW A, RW5, @RW5+d8	ADDW 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, rel, @RW5+d8, rel	DWBZ RW5, rel, @RW5+d8, rel
+6	ADDW A, RW6, @RW6+d8	ADDW 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, rel, @RW6+d8, rel	DWBZ RW6, rel, @RW6+d8, rel
+7	ADDW A, RW7, @RW7+d8	ADDW 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, rel, @RW7+d8, rel	DWBZ RW7, rel, @RW7+d8, rel
+8	ADDW A, @RW0, @RW0+d16	ADDW 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, rel, @RW0+d16, rel	DWBZ @RW0, rel, @RW0+d16, rel
+9	ADDW A, @RW1, @RW1+d16	ADDW 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, rel, @RW1+d16, rel	DWBZ @RW1, rel, @RW1+d16, rel
+A	ADDW A, @RW2, @RW2+d16	ADDW 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, rel, @RW2+d16, rel	DWBZ @RW2, rel, @RW2+d16, rel
+B	ADDW A, @RW3, @RW3+d16	ADDW 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, rel, @RW3+d16, rel	DWBZ @RW3, rel, @RW3+d16, rel
+C	ADDW A, @RW0+, @RW0+RW7	ADDW 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+, rel, @RW0+RW7, rel	DWBZ @RW0+, rel, @RW0+RW7, rel
+D	ADDW A, @RW1+, @RW1+RW7	ADDW 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+, rel, @RW1+RW7, rel	DWBZ @RW1+, rel, @RW1+RW7, rel
+E	ADDW A, @RW2+, @PC+d16	ADDW 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+, rel, @PC+d16, rel	DWBZ @RW2+, rel, @PC+d16, rel
+F	ADDW A, @RW3+, addr16	ADDW A, @RW3+, addr16	SUBW A, @RW3+, addr16	SUBW A, @RW3+, addr16	ADDCW A, @RW3+, addr16	ADDCW A, @RW3+, addr16	CMPW A, @RW3+, addr16	CMPW A, @RW3+, addr16	ANDW A, @RW3+, addr16	ANDW A, @RW3+, addr16	ORW A, @RW3+, addr16	ORW A, @RW3+, addr16	XORW A, @RW3+, addr16	XORW A, @RW3+, addr16	DWBZ @RW3+, rel, addr16, rel	DWBZ @RW3+, rel, addr16, rel

表 B.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, A', @RW0+d8, A	SUBCW A, RW0', @RW0+d8	SUBCW A, RW0', @RW0+d8	NEG RW0', @RW0+d8	NEG RW0', @RW0+d8	ANDW RW0, A', @RW0+d8, A	ANDW RW0, A', @RW0+d8, A	ORW RW0, A', @RW0+d8, A	ORW RW0, A', @RW0+d8, A	XORW RW0, A', @RW0+d8, A	XORW RW0, A', @RW0+d8, A	NOTW RW0', @RW0+d8	NOTW RW0', @RW0+d8
+1	ADDW RW1, A', @RW1+d8, A	SUBW RW1, A', @RW1+d8, A	SUBW RW1, A', @RW1+d8, A	SUBW RW1, A', @RW1+d8, A	SUBCW A, RW1', @RW1+d8	SUBCW A, RW1', @RW1+d8	NEG RW1', @RW1+d8	NEG RW1', @RW1+d8	ANDW RW1, A', @RW1+d8, A	ANDW RW1, A', @RW1+d8, A	ORW RW1, A', @RW1+d8, A	ORW RW1, A', @RW1+d8, A	XORW RW1, A', @RW1+d8, A	XORW RW1, A', @RW1+d8, A	NOTW RW1', @RW1+d8	NOTW RW1', @RW1+d8
+2	ADDW RW2, A', @RW2+d8, A	SUBW RW2, A', @RW2+d8, A	SUBW RW2, A', @RW2+d8, A	SUBW RW2, A', @RW2+d8, A	SUBCW A, RW2', @RW2+d8	SUBCW A, RW2', @RW2+d8	NEG RW2', @RW2+d8	NEG RW2', @RW2+d8	ANDW RW2, A', @RW2+d8, A	ANDW RW2, A', @RW2+d8, A	ORW RW2, A', @RW2+d8, A	ORW RW2, A', @RW2+d8, A	XORW RW2, A', @RW2+d8, A	XORW RW2, A', @RW2+d8, A	NOTW RW2', @RW2+d8	NOTW RW2', @RW2+d8
+3	ADDW RW3, A', @RW3+d8, A	SUBW RW3, A', @RW3+d8, A	SUBW RW3, A', @RW3+d8, A	SUBW RW3, A', @RW3+d8, A	SUBCW A, RW3', @RW3+d8	SUBCW A, RW3', @RW3+d8	NEG RW3', @RW3+d8	NEG RW3', @RW3+d8	ANDW RW3, A', @RW3+d8, A	ANDW RW3, A', @RW3+d8, A	ORW RW3, A', @RW3+d8, A	ORW RW3, A', @RW3+d8, A	XORW RW3, A', @RW3+d8, A	XORW RW3, A', @RW3+d8, A	NOTW RW3', @RW3+d8	NOTW RW3', @RW3+d8
+4	ADDW RW4, A', @RW4+d8, A	SUBW RW4, A', @RW4+d8, A	SUBW RW4, A', @RW4+d8, A	SUBW RW4, A', @RW4+d8, A	SUBCW A, RW4', @RW4+d8	SUBCW A, RW4', @RW4+d8	NEG RW4', @RW4+d8	NEG RW4', @RW4+d8	ANDW RW4, A', @RW4+d8, A	ANDW RW4, A', @RW4+d8, A	ORW RW4, A', @RW4+d8, A	ORW RW4, A', @RW4+d8, A	XORW RW4, A', @RW4+d8, A	XORW RW4, A', @RW4+d8, A	NOTW RW4', @RW4+d8	NOTW RW4', @RW4+d8
+5	ADDW RW5, A', @RW5+d8, A	SUBW RW5, A', @RW5+d8, A	SUBW RW5, A', @RW5+d8, A	SUBW RW5, A', @RW5+d8, A	SUBCW A, RW5', @RW5+d8	SUBCW A, RW5', @RW5+d8	NEG RW5', @RW5+d8	NEG RW5', @RW5+d8	ANDW RW5, A', @RW5+d8, A	ANDW RW5, A', @RW5+d8, A	ORW RW5, A', @RW5+d8, A	ORW RW5, A', @RW5+d8, A	XORW RW5, A', @RW5+d8, A	XORW RW5, A', @RW5+d8, A	NOTW RW5', @RW5+d8	NOTW RW5', @RW5+d8
+6	ADDW RW6, A', @RW6+d8, A	SUBW RW6, A', @RW6+d8, A	SUBW RW6, A', @RW6+d8, A	SUBW RW6, A', @RW6+d8, A	SUBCW A, RW6', @RW6+d8	SUBCW A, RW6', @RW6+d8	NEG RW6', @RW6+d8	NEG RW6', @RW6+d8	ANDW RW6, A', @RW6+d8, A	ANDW RW6, A', @RW6+d8, A	ORW RW6, A', @RW6+d8, A	ORW RW6, A', @RW6+d8, A	XORW RW6, A', @RW6+d8, A	XORW RW6, A', @RW6+d8, A	NOTW RW6', @RW6+d8	NOTW RW6', @RW6+d8
+7	ADDW RW7, A', @RW7+d8, A	SUBW RW7, A', @RW7+d8, A	SUBW RW7, A', @RW7+d8, A	SUBW RW7, A', @RW7+d8, A	SUBCW A, RW7', @RW7+d8	SUBCW A, RW7', @RW7+d8	NEG RW7', @RW7+d8	NEG RW7', @RW7+d8	ANDW RW7, A', @RW7+d8, A	ANDW RW7, A', @RW7+d8, A	ORW RW7, A', @RW7+d8, A	ORW RW7, A', @RW7+d8, A	XORW RW7, A', @RW7+d8, A	XORW RW7, A', @RW7+d8, A	NOTW RW7', @RW7+d8	NOTW RW7', @RW7+d8
+8	ADDW @RW0, A', @RW0+d16, A	SUBW @RW0, A', @RW0+d16, A	SUBW @RW0, A', @RW0+d16, A	SUBW @RW0, A', @RW0+d16, A	SUBCW A, @RW0', @RW0+d16	SUBCW A, @RW0', @RW0+d16	NEG @RW0', @RW0+d16	NEG @RW0', @RW0+d16	ANDW @RW0, A', @RW0+d16, A	ANDW @RW0, A', @RW0+d16, A	ORW @RW0, A', @RW0+d16, A	ORW @RW0, A', @RW0+d16, A	XORW @RW0, A', @RW0+d16, A	XORW @RW0, A', @RW0+d16, A	NOTW @RW0', @RW0+d16	NOTW @RW0', @RW0+d16
+9	ADDW @RW1, A', @RW1+d16, A	SUBW @RW1, A', @RW1+d16, A	SUBW @RW1, A', @RW1+d16, A	SUBW @RW1, A', @RW1+d16, A	SUBCW A, @RW1', @RW1+d16	SUBCW A, @RW1', @RW1+d16	NEG @RW1', @RW1+d16	NEG @RW1', @RW1+d16	ANDW @RW1, A', @RW1+d16, A	ANDW @RW1, A', @RW1+d16, A	ORW @RW1, A', @RW1+d16, A	ORW @RW1, A', @RW1+d16, A	XORW @RW1, A', @RW1+d16, A	XORW @RW1, A', @RW1+d16, A	NOTW @RW1', @RW1+d16	NOTW @RW1', @RW1+d16
+A	ADDW @RW2, A', @RW2+d16, A	SUBW @RW2, A', @RW2+d16, A	SUBW @RW2, A', @RW2+d16, A	SUBW @RW2, A', @RW2+d16, A	SUBCW A, @RW2', @RW2+d16	SUBCW A, @RW2', @RW2+d16	NEG @RW2', @RW2+d16	NEG @RW2', @RW2+d16	ANDW @RW2, A', @RW2+d16, A	ANDW @RW2, A', @RW2+d16, A	ORW @RW2, A', @RW2+d16, A	ORW @RW2, A', @RW2+d16, A	XORW @RW2, A', @RW2+d16, A	XORW @RW2, A', @RW2+d16, A	NOTW @RW2', @RW2+d16	NOTW @RW2', @RW2+d16
+B	ADDW @RW3, A', @RW3+d16, A	SUBW @RW3, A', @RW3+d16, A	SUBW @RW3, A', @RW3+d16, A	SUBW @RW3, A', @RW3+d16, A	SUBCW A, @RW3', @RW3+d16	SUBCW A, @RW3', @RW3+d16	NEG @RW3', @RW3+d16	NEG @RW3', @RW3+d16	ANDW @RW3, A', @RW3+d16, A	ANDW @RW3, A', @RW3+d16, A	ORW @RW3, A', @RW3+d16, A	ORW @RW3, A', @RW3+d16, A	XORW @RW3, A', @RW3+d16, A	XORW @RW3, A', @RW3+d16, A	NOTW @RW3', @RW3+d16	NOTW @RW3', @RW3+d16
+C	ADDW @RW0+, A', @RW0+RW7, A	SUBW @RW0+, A', @RW0+RW7, A	SUBW @RW0+, A', @RW0+RW7, A	SUBW @RW0+, A', @RW0+RW7, A	SUBCW A, @RW0+', @RW0+RW7	SUBCW A, @RW0+', @RW0+RW7	NEG @RW0+', @RW0+RW7	NEG @RW0+', @RW0+RW7	ANDW @RW0+, A', @RW0+RW7, A	ANDW @RW0+, A', @RW0+RW7, A	ORW @RW0+, A', @RW0+RW7, A	ORW @RW0+, A', @RW0+RW7, A	XORW @RW0+, A', @RW0+RW7, A	XORW @RW0+, A', @RW0+RW7, A	NOTW @RW0+', @RW0+RW7	NOTW @RW0+', @RW0+RW7
+D	ADDW @RW1+, A', @RW1+RW7, A	SUBW @RW1+, A', @RW1+RW7, A	SUBW @RW1+, A', @RW1+RW7, A	SUBW @RW1+, A', @RW1+RW7, A	SUBCW A, @RW1+', @RW1+RW7	SUBCW A, @RW1+', @RW1+RW7	NEG @RW1+', @RW1+RW7	NEG @RW1+', @RW1+RW7	ANDW @RW1+, A', @RW1+RW7, A	ANDW @RW1+, A', @RW1+RW7, A	ORW @RW1+, A', @RW1+RW7, A	ORW @RW1+, A', @RW1+RW7, A	XORW @RW1+, A', @RW1+RW7, A	XORW @RW1+, A', @RW1+RW7, A	NOTW @RW1+', @RW1+RW7	NOTW @RW1+', @RW1+RW7
+E	ADDW @RW2+, A', @PC+d16, A	SUBW @RW2+, A', @PC+d16, A	SUBW @RW2+, A', @PC+d16, A	SUBW @RW2+, A', @PC+d16, A	SUBCW A, @RW2+', @PC+d16	SUBCW A, @RW2+', @PC+d16	NEG @RW2+', @PC+d16	NEG @RW2+', @PC+d16	ANDW @RW2+, A', @PC+d16, A	ANDW @RW2+, A', @PC+d16, A	ORW @RW2+, A', @PC+d16, A	ORW @RW2+, A', @PC+d16, A	XORW @RW2+, A', @PC+d16, A	XORW @RW2+, A', @PC+d16, A	NOTW @RW2+', @PC+d16	NOTW @RW2+', @PC+d16
+F	ADDW @RW3+, A', addr16, A	SUBW @RW3+, A', addr16, A	SUBW @RW3+, A', addr16, A	SUBW @RW3+, A', addr16, A	SUBCW A, @RW3+', addr16	SUBCW A, @RW3+', addr16	NEG @RW3+', addr16	NEG @RW3+', addr16	ANDW @RW3+, A', addr16, A	ANDW @RW3+, A', addr16, A	ORW @RW3+, A', addr16, A	ORW @RW3+, A', addr16, A	XORW @RW3+, A', addr16, A	XORW @RW3+, A', addr16, A	NOTW @RW3+', addr16	NOTW @RW3+', addr16

表 B.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, @RW0+d8	MULUW A, MULUW A, @RW0+d8	MULUW A, MULUW A, @RW0+d8	MUL A, R0', @RW0+d8	MUL A, @RW0+d8	MULW A, MULW A, @RW0+d8	MULW A, MULW A, @RW0+d8	DIVU A, R0', @RW0+d8	DIVU A, @RW0+d8	DIVUW A, DIVUW A, @RW0+d8	DIVUW A, DIVUW A, @RW0+d8	DIV A, R0', @RW0+d8	DIV A, @RW0+d8	DIVW A, DIVW A, @RW0+d8	DIVW A, DIVW A, @RW0+d8
+1	MULU A, R1', @RW1+d8	MULU A, MULU A, @RW1+d8	MULUW A, MULUW A, @RW1+d8	MULUW A, MULUW A, @RW1+d8	MUL A, R1', @RW1+d8	MUL A, @RW1+d8	MULW A, MULW A, @RW1+d8	MULW A, MULW A, @RW1+d8	DIVU A, R1', @RW1+d8	DIVU A, @RW1+d8	DIVUW A, DIVUW A, @RW1+d8	DIVUW A, DIVUW A, @RW1+d8	DIV A, R1', @RW1+d8	DIV A, @RW1+d8	DIVW A, DIVW A, @RW1+d8	DIVW A, DIVW A, @RW1+d8
+2	MULU A, R2', @RW2+d8	MULU A, MULU A, @RW2+d8	MULUW A, MULUW A, @RW2+d8	MULUW A, MULUW A, @RW2+d8	MUL A, R2', @RW2+d8	MUL A, @RW2+d8	MULW A, MULW A, @RW2+d8	MULW A, MULW A, @RW2+d8	DIVU A, R2', @RW2+d8	DIVU A, @RW2+d8	DIVUW A, DIVUW A, @RW2+d8	DIVUW A, DIVUW A, @RW2+d8	DIV A, R2', @RW2+d8	DIV A, @RW2+d8	DIVW A, DIVW A, @RW2+d8	DIVW A, DIVW A, @RW2+d8
+3	MULU A, R3', @RW3+d8	MULU A, MULU A, @RW3+d8	MULUW A, MULUW A, @RW3+d8	MULUW A, MULUW A, @RW3+d8	MUL A, R3', @RW3+d8	MUL A, @RW3+d8	MULW A, MULW A, @RW3+d8	MULW A, MULW A, @RW3+d8	DIVU A, R3', @RW3+d8	DIVU A, @RW3+d8	DIVUW A, DIVUW A, @RW3+d8	DIVUW A, DIVUW A, @RW3+d8	DIV A, R3', @RW3+d8	DIV A, @RW3+d8	DIVW A, DIVW A, @RW3+d8	DIVW A, DIVW A, @RW3+d8
+4	MULU A, R4', @RW4+d8	MULU A, MULU A, @RW4+d8	MULUW A, MULUW A, @RW4+d8	MULUW A, MULUW A, @RW4+d8	MUL A, R4', @RW4+d8	MUL A, @RW4+d8	MULW A, MULW A, @RW4+d8	MULW A, MULW A, @RW4+d8	DIVU A, R4', @RW4+d8	DIVU A, @RW4+d8	DIVUW A, DIVUW A, @RW4+d8	DIVUW A, DIVUW A, @RW4+d8	DIV A, R4', @RW4+d8	DIV A, @RW4+d8	DIVW A, DIVW A, @RW4+d8	DIVW A, DIVW A, @RW4+d8
+5	MULU A, R5', @RW5+d8	MULU A, MULU A, @RW5+d8	MULUW A, MULUW A, @RW5+d8	MULUW A, MULUW A, @RW5+d8	MUL A, R5', @RW5+d8	MUL A, @RW5+d8	MULW A, MULW A, @RW5+d8	MULW A, MULW A, @RW5+d8	DIVU A, R5', @RW5+d8	DIVU A, @RW5+d8	DIVUW A, DIVUW A, @RW5+d8	DIVUW A, DIVUW A, @RW5+d8	DIV A, R5', @RW5+d8	DIV A, @RW5+d8	DIVW A, DIVW A, @RW5+d8	DIVW A, DIVW A, @RW5+d8
+6	MULU A, R6', @RW6+d8	MULU A, MULU A, @RW6+d8	MULUW A, MULUW A, @RW6+d8	MULUW A, MULUW A, @RW6+d8	MUL A, R6', @RW6+d8	MUL A, @RW6+d8	MULW A, MULW A, @RW6+d8	MULW A, MULW A, @RW6+d8	DIVU A, R6', @RW6+d8	DIVU A, @RW6+d8	DIVUW A, DIVUW A, @RW6+d8	DIVUW A, DIVUW A, @RW6+d8	DIV A, R6', @RW6+d8	DIV A, @RW6+d8	DIVW A, DIVW A, @RW6+d8	DIVW A, DIVW A, @RW6+d8
+7	MULU A, R7', @RW7+d8	MULU A, MULU A, @RW7+d8	MULUW A, MULUW A, @RW7+d8	MULUW A, MULUW A, @RW7+d8	MUL A, R7', @RW7+d8	MUL A, @RW7+d8	MULW A, MULW A, @RW7+d8	MULW A, MULW A, @RW7+d8	DIVU A, R7', @RW7+d8	DIVU A, @RW7+d8	DIVUW A, DIVUW A, @RW7+d8	DIVUW A, DIVUW A, @RW7+d8	DIV A, R7', @RW7+d8	DIV A, @RW7+d8	DIVW A, DIVW A, @RW7+d8	DIVW A, DIVW A, @RW7+d8
+8	MULU A, @RW0, @RW0+d16	MULU A, MULU A, @RW0, @RW0+d16	MULUW A, MULUW A, @RW0, @RW0+d16	MULUW A, MULUW A, @RW0, @RW0+d16	MUL A, @RW0, @RW0+d16	MUL A, @RW0+d16	MULW A, MULW A, @RW0, @RW0+d16	MULW A, MULW A, @RW0, @RW0+d16	DIVU A, @RW0, @RW0+d16	DIVU A, @RW0+d16	DIVUW A, DIVUW A, @RW0, @RW0+d16	DIVUW A, DIVUW A, @RW0, @RW0+d16	DIV A, @RW0, @RW0+d16	DIV A, @RW0+d16	DIVW A, DIVW A, @RW0, @RW0+d16	DIVW A, DIVW A, @RW0, @RW0+d16
+9	MULU A, @RW1, @RW1+d16	MULU A, MULU A, @RW1, @RW1+d16	MULUW A, MULUW A, @RW1, @RW1+d16	MULUW A, MULUW A, @RW1, @RW1+d16	MUL A, @RW1, @RW1+d16	MUL A, @RW1+d16	MULW A, MULW A, @RW1, @RW1+d16	MULW A, MULW A, @RW1, @RW1+d16	DIVU A, @RW1, @RW1+d16	DIVU A, @RW1+d16	DIVUW A, DIVUW A, @RW1, @RW1+d16	DIVUW A, DIVUW A, @RW1, @RW1+d16	DIV A, @RW1, @RW1+d16	DIV A, @RW1+d16	DIVW A, DIVW A, @RW1, @RW1+d16	DIVW A, DIVW A, @RW1, @RW1+d16
+A	MULU A, @RW2, @RW2+d16	MULU A, MULU A, @RW2, @RW2+d16	MULUW A, MULUW A, @RW2, @RW2+d16	MULUW A, MULUW A, @RW2, @RW2+d16	MUL A, @RW2, @RW2+d16	MUL A, @RW2+d16	MULW A, MULW A, @RW2, @RW2+d16	MULW A, MULW A, @RW2, @RW2+d16	DIVU A, @RW2, @RW2+d16	DIVU A, @RW2+d16	DIVUW A, DIVUW A, @RW2, @RW2+d16	DIVUW A, DIVUW A, @RW2, @RW2+d16	DIV A, @RW2, @RW2+d16	DIV A, @RW2+d16	DIVW A, DIVW A, @RW2, @RW2+d16	DIVW A, DIVW A, @RW2, @RW2+d16
+B	MULU A, @RW3, @RW3+d16	MULU A, MULU A, @RW3, @RW3+d16	MULUW A, MULUW A, @RW3, @RW3+d16	MULUW A, MULUW A, @RW3, @RW3+d16	MUL A, @RW3, @RW3+d16	MUL A, @RW3+d16	MULW A, MULW A, @RW3, @RW3+d16	MULW A, MULW A, @RW3, @RW3+d16	DIVU A, @RW3, @RW3+d16	DIVU A, @RW3+d16	DIVUW A, DIVUW A, @RW3, @RW3+d16	DIVUW A, DIVUW A, @RW3, @RW3+d16	DIV A, @RW3, @RW3+d16	DIV A, @RW3+d16	DIVW A, DIVW A, @RW3, @RW3+d16	DIVW A, DIVW A, @RW3, @RW3+d16
+C	MULU A, @RW0+, @RW0+RW7	MULU A, MULU A, @RW0+, @RW0+RW7	MULUW A, MULUW A, @RW0+, @RW0+RW7	MULUW A, MULUW A, @RW0+, @RW0+RW7	MUL A, @RW0+, @RW0+RW7	MUL A, @RW0+RW7	MULW A, MULW A, @RW0+, @RW0+RW7	MULW A, MULW A, @RW0+, @RW0+RW7	DIVU A, @RW0+, @RW0+RW7	DIVU A, @RW0+RW7	DIVUW A, DIVUW A, @RW0+, @RW0+RW7	DIVUW A, DIVUW A, @RW0+, @RW0+RW7	DIV A, @RW0+, @RW0+RW7	DIV A, @RW0+RW7	DIVW A, DIVW A, @RW0+, @RW0+RW7	DIVW A, DIVW A, @RW0+, @RW0+RW7
+D	MULU A, @RW1+, @RW1+RW7	MULU A, MULU A, @RW1+, @RW1+RW7	MULUW A, MULUW A, @RW1+, @RW1+RW7	MULUW A, MULUW A, @RW1+, @RW1+RW7	MUL A, @RW1+, @RW1+RW7	MUL A, @RW1+RW7	MULW A, MULW A, @RW1+, @RW1+RW7	MULW A, MULW A, @RW1+, @RW1+RW7	DIVU A, @RW1+, @RW1+RW7	DIVU A, @RW1+RW7	DIVUW A, DIVUW A, @RW1+, @RW1+RW7	DIVUW A, DIVUW A, @RW1+, @RW1+RW7	DIV A, @RW1+, @RW1+RW7	DIV A, @RW1+RW7	DIVW A, DIVW A, @RW1+, @RW1+RW7	DIVW A, DIVW A, @RW1+, @RW1+RW7
+E	MULU A, @RW2+, @PC+d16	MULU A, MULU A, @RW2+, @PC+d16	MULUW A, MULUW A, @RW2+, @PC+d16	MULUW A, MULUW A, @RW2+, @PC+d16	MUL A, @RW2+, @PC+d16	MUL A, @PC+d16	MULW A, MULW A, @RW2+, @PC+d16	MULW A, MULW A, @RW2+, @PC+d16	DIVU A, @RW2+, @PC+d16	DIVU A, @PC+d16	DIVUW A, DIVUW A, @RW2+, @PC+d16	DIVUW A, DIVUW A, @RW2+, @PC+d16	DIV A, @RW2+, @PC+d16	DIV A, @PC+d16	DIVW A, DIVW A, @RW2+, @PC+d16	DIVW A, DIVW A, @RW2+, @PC+d16
+F	MULU A, @RW3+, addr16	MULU A, MULU A, @RW3+, addr16	MULUW A, MULUW A, @RW3+, addr16	MULUW A, MULUW A, @RW3+, addr16	MUL A, @RW3+, addr16	MUL A, addr16	MULW A, MULW A, @RW3+, addr16	MULW A, MULW A, @RW3+, addr16	DIVU A, @RW3+, addr16	DIVU A, addr16	DIVUW A, DIVUW A, @RW3+, addr16	DIVUW A, DIVUW A, @RW3+, addr16	DIV A, @RW3+, addr16	DIV A, addr16	DIVW A, DIVW A, @RW3+, addr16	DIVW A, DIVW A, @RW3+, addr16

付録 B 命令

表 B.9-15 MOVEA RWi, ea 命令 (第1バイト=79_H)[illegible]

[illegible]

付録 B 命令

表 B.9-17 MOVW RWi, ea 命令 (第1バイト=7B_H)[illegible]

表 B.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, 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	MOV R0, R15, @RW0+d8, R15
+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

表 B.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, RW8, @RW0+d8, RW8	MOVW RW0, RW9, @RW0+d8, RW9	MOVW RW0, RW10, @RW0+d8, RW10	MOVW RW0, RW11, @RW0+d8, RW11	MOVW RW0, RW12, @RW0+d8, RW12	MOVW RW0, RW13, @RW0+d8, RW13	MOVW RW0, RW14, @RW0+d8, RW14	MOVW RW0, RW15, @RW0+d8, RW15
+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, RW8, @RW1+d8, RW8	MOVW RW1, RW9, @RW1+d8, RW9	MOVW RW1, RW10, @RW1+d8, RW10	MOVW RW1, RW11, @RW1+d8, RW11	MOVW RW1, RW12, @RW1+d8, RW12	MOVW RW1, RW13, @RW1+d8, RW13	MOVW RW1, RW14, @RW1+d8, RW14	MOVW RW1, RW15, @RW1+d8, RW15
+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, RW8, @RW2+d8, RW8	MOVW RW2, RW9, @RW2+d8, RW9	MOVW RW2, RW10, @RW2+d8, RW10	MOVW RW2, RW11, @RW2+d8, RW11	MOVW RW2, RW12, @RW2+d8, RW12	MOVW RW2, RW13, @RW2+d8, RW13	MOVW RW2, RW14, @RW2+d8, RW14	MOVW RW2, RW15, @RW2+d8, RW15
+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, RW8, @RW3+d8, RW8	MOVW RW3, RW9, @RW3+d8, RW9	MOVW RW3, RW10, @RW3+d8, RW10	MOVW RW3, RW11, @RW3+d8, RW11	MOVW RW3, RW12, @RW3+d8, RW12	MOVW RW3, RW13, @RW3+d8, RW13	MOVW RW3, RW14, @RW3+d8, RW14	MOVW RW3, RW15, @RW3+d8, RW15
+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, RW8, @RW4+d8, RW8	MOVW RW4, RW9, @RW4+d8, RW9	MOVW RW4, RW10, @RW4+d8, RW10	MOVW RW4, RW11, @RW4+d8, RW11	MOVW RW4, RW12, @RW4+d8, RW12	MOVW RW4, RW13, @RW4+d8, RW13	MOVW RW4, RW14, @RW4+d8, RW14	MOVW RW4, RW15, @RW4+d8, RW15
+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, RW8, @RW5+d8, RW8	MOVW RW5, RW9, @RW5+d8, RW9	MOVW RW5, RW10, @RW5+d8, RW10	MOVW RW5, RW11, @RW5+d8, RW11	MOVW RW5, RW12, @RW5+d8, RW12	MOVW RW5, RW13, @RW5+d8, RW13	MOVW RW5, RW14, @RW5+d8, RW14	MOVW RW5, RW15, @RW5+d8, RW15
+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, RW8, @RW6+d8, RW8	MOVW RW6, RW9, @RW6+d8, RW9	MOVW RW6, RW10, @RW6+d8, RW10	MOVW RW6, RW11, @RW6+d8, RW11	MOVW RW6, RW12, @RW6+d8, RW12	MOVW RW6, RW13, @RW6+d8, RW13	MOVW RW6, RW14, @RW6+d8, RW14	MOVW RW6, RW15, @RW6+d8, RW15
+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, RW8, @RW7+d8, RW8	MOVW RW7, RW9, @RW7+d8, RW9	MOVW RW7, RW10, @RW7+d8, RW10	MOVW RW7, RW11, @RW7+d8, RW11	MOVW RW7, RW12, @RW7+d8, RW12	MOVW RW7, RW13, @RW7+d8, RW13	MOVW RW7, RW14, @RW7+d8, RW14	MOVW RW7, RW15, @RW7+d8, RW15
+8	MOVW @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, RW8, +d16, RW8	MOVW @RW0, RW9, +d16, RW9	MOVW @RW0, RW10, +d16, RW10	MOVW @RW0, RW11, +d16, RW11	MOVW @RW0, RW12, +d16, RW12	MOVW @RW0, RW13, +d16, RW13	MOVW @RW0, RW14, +d16, RW14	MOVW @RW0, RW15, +d16, RW15
+9	MOVW @RW1, RW0, +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, RW8, +d16, RW8	MOVW @RW1, RW9, +d16, RW9	MOVW @RW1, RW10, +d16, RW10	MOVW @RW1, RW11, +d16, RW11	MOVW @RW1, RW12, +d16, RW12	MOVW @RW1, RW13, +d16, RW13	MOVW @RW1, RW14, +d16, RW14	MOVW @RW1, RW15, +d16, RW15
+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, RW8, +d16, RW8	MOVW @RW2, RW9, +d16, RW9	MOVW @RW2, RW10, +d16, RW10	MOVW @RW2, RW11, +d16, RW11	MOVW @RW2, RW12, +d16, RW12	MOVW @RW2, RW13, +d16, RW13	MOVW @RW2, RW14, +d16, RW14	MOVW @RW2, RW15, +d16, RW15
+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, RW8, +d16, RW8	MOVW @RW3, RW9, +d16, RW9	MOVW @RW3, RW10, +d16, RW10	MOVW @RW3, RW11, +d16, RW11	MOVW @RW3, RW12, +d16, RW12	MOVW @RW3, RW13, +d16, RW13	MOVW @RW3, RW14, +d16, RW14	MOVW @RW3, RW15, +d16, RW15
+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+, RW8, +RW7, RW8	MOVW @RW0+, RW9, +RW7, RW9	MOVW @RW0+, RW10, +RW7, RW10	MOVW @RW0+, RW11, +RW7, RW11	MOVW @RW0+, RW12, +RW7, RW12	MOVW @RW0+, RW13, +RW7, RW13	MOVW @RW0+, RW14, +RW7, RW14	MOVW @RW0+, RW15, +RW7, RW15
+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+, RW8, +RW7, RW8	MOVW @RW1+, RW9, +RW7, RW9	MOVW @RW1+, RW10, +RW7, RW10	MOVW @RW1+, RW11, +RW7, RW11	MOVW @RW1+, RW12, +RW7, RW12	MOVW @RW1+, RW13, +RW7, RW13	MOVW @RW1+, RW14, +RW7, RW14	MOVW @RW1+, RW15, +RW7, RW15
+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+, RW8, +d16, RW8	MOVW @RW2+, RW9, +d16, RW9	MOVW @RW2+, RW10, +d16, RW10	MOVW @RW2+, RW11, +d16, RW11	MOVW @RW2+, RW12, +d16, RW12	MOVW @RW2+, RW13, +d16, RW13	MOVW @RW2+, RW14, +d16, RW14	MOVW @RW2+, RW15, +d16, RW15
+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+, RW8, +d16, RW8	MOVW @RW3+, RW9, +d16, RW9	MOVW @RW3+, RW10, +d16, RW10	MOVW @RW3+, RW11, +d16, RW11	MOVW @RW3+, RW12, +d16, RW12	MOVW @RW3+, RW13, +d16, RW13	MOVW @RW3+, RW14, +d16, RW14	MOVW @RW3+, RW15, +d16, RW15

[illegible]

付録 B 命令

表 B.9-21 XCHW RWi, ea 命令 (第1バイト = 7FH)

[illegible]

索引

Numerics

16 ビットタイマレジスタ	
16 ビットタイマレジスタ 0 ~ 2	
(TMR0 ~ TMR2)	413
16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/	
16 ビットリロードレジスタ 0 ~ 2	
(TMRLR0 ~ TMRLR2)	413
16 ビットデータバス	
外部バス 16 ビットデータバスモードおよび	
ノンマルチプレクス 16 ビット外部バス	
モードの端子状態	173
外部バス 16 ビットデータバスモードおよび	
マルチプレクス 16 ビット外部バスモードの	
端子状態	169
16 ビット入出力タイマ	
16 ビット入出力タイマの構成および機能	250
16 ビット入出力タイマの動作, タイミング	
.....	268
16 ビット入出力タイマのレジスタ構成	252
16 ビットバス	
外部 16 ビットバスモード (外部データバス	
16 ビット / マルチプレクスモード)	197
16 ビットフリーランタイム	
16 ビットフリーランタイムの動作	269
16 ビットフリーランタイムの	
ブロックダイアグラム	254
16 ビットフリーランタイムのレジスタ一覧	
.....	253
16 ビットリロードタイマ	
16 ビットリロードタイマの概要	404
16 ビットリロードタイマの設定	415
16 ビットリロードタイマの動作モード	405
16 ビットリロードタイマの	
ブロックダイアグラム	407
16 ビットリロードタイマレジスタ一覧	408
16 ビットリロードレジスタ	
16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/	
16 ビットリロードレジスタ 0 ~ 2	
(TMRLR0 ~ TMRLR2)	413
16 ビットリロードレジスタ 0 ~ 2	
(TMRLR0 ~ TMRLR2)	414
1 対 1 モード	
外部クロック (1 対 1 モード) によるポーレート	
.....	527
24 ビット	
24 ビットオペランド指定	30
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	471
8/10 ビット A/D コンバータの機能	452
8/10 ビット A/D コンバータの	
ブロックダイアグラム	453
8/10 ビット A/D コンバータのレジスタ一覧	
.....	455
8/16 ビット PPG タイマ	
8/16 ビット PPG タイマの概略仕様	424
8/16 ビット PPG タイマの動作概略	436

8/16 ビット PPG タイマのブロックダイアグラム	
.....	425
8/16 ビット PPG タイマのレジスタ一覧	427
8/16 ビット PPG タイマの割込み	439
8 ビットデータバス	
外部バス 8 ビットデータバスモードおよび	
ノンマルチプレクス 8 ビット外部バスモード	
の端子状態	175
外部バス 8 ビットデータバスモードおよび	
マルチプレクス 8 ビット外部バスモードの	
端子状態	171

A

A

アキュムレータ (A)	37
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	471
8/10 ビット A/D コンバータの機能	452
8/10 ビット A/D コンバータの ブロックダイアグラム	453
8/10 ビット A/D コンバータのレジスター一覧	455
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)	459
A/D 制御ステータスレジスタ上位 (ADCS1)	456
A/D データレジスタ	
A/D データレジスタ (ADCR1/ADCR0)	463
A/D 変換チャンネル設定レジスタ	
A/D 変換チャンネル設定レジスタ (ADMR)	461
A/D 変換データ	
A/D 変換データ保護機能	469
ADB	
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	43
ADCR	
A/D データレジスタ (ADCR1/ADCR0)	463
ADCS	
A/D 制御ステータスレジスタ下位 (ADCS0)	459
A/D 制御ステータスレジスタ上位 (ADCS1)	456
ADER	
アナログ入力許可レジスタ (ADER0,ADER1)	213
ADMR	
A/D 変換チャンネル設定レジスタ (ADMR)	461
ARSR	
自動レディ機能選択レジスタ (ARSR)	190

B

BAP

バッファアドレスポインタ (BAP)	84
--------------------	----

C

CCR

コンディションコードレジスタ (CCR)	39
----------------------	----

CKSCR

クロック選択レジスタ (CKSCR) の構成	137
------------------------	-----

CMR

コモンレジスタバンクプリフィックス (CMR)	47
----------------------------	----

CPCLR

コンペアクリアレジスタ (CPCLR)	254
---------------------	-----

CPU

CPU 動作モードと消費電流	148
CPU の概要	26
CPU メモリ空間の概要	27
CPU 間欠動作	
CPU 間欠動作モード	149, 156

D

DBAPH

DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)	106
---	-----

DBAPL

DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)	106
---	-----

DBAPM

DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)	106
---	-----

DCSR

DMA ディスクリプタチャンネル指定レジスタ (DCSR)	95
----------------------------------	----

DCT

データカウンタ (DCT)	83
---------------	----

DDCTH

DMA データカウンタ (DDCTH/DDCTL)	102
DMA データカウンタ (DDCTH/DDCTL) の設定値 について	102

DDCTL

DMA データカウンタ (DDCTH/DDCTL)	102
DMA データカウンタ (DDCTH/DDCTL) の設定値 について	102

DDR

ポート方向レジスタ (DDR0 ~ DDRB)	210
-------------------------	-----

DDWR

DMA ディスクリプタウィンドウレジスタ (DDWR) の構成	101
------------------------------------	-----

DERH

DMA イネーブルレジスタ (DERH/DERL)	100
---------------------------	-----

DERL

DMA イネーブルレジスタ (DERH/DERL)	100
---------------------------	-----

DIOAH

DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL)	103
---------------------------------------	-----

DIOAL

DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL)	103
---------------------------------------	-----

DIVR

PWC 分周比制御レジスタ (DIVR)	385
----------------------	-----

DMA I/O レジスタアドレスポインタ

DMA I/O レジスタアドレスポインタ (DIOAH/DIOAL)	103
---------------------------------------	-----

DMACS

DMA コントロールレジスタ (DMACS)	104
------------------------	-----

DMA イネーブルレジスタ

DMA イネーブルレジスタ (DERH/DERL)	100
---------------------------	-----

DMA コントロールレジスタ

DMA コントロールレジスタ (DMACS)	104
------------------------	-----

DMA ステータスレジスタ

DMA ステータスレジスタ (DSRH/DSRL)	97
---------------------------	----

DMA ストップステータスレジスタ

DMA ストップステータスレジスタ (DSSR)	98
--------------------------	----

DMA ディスクリプタ

DMA ディスクリプタの各レジスタ	101
-------------------	-----

DMA ディスクリプタウィンドウレジスタ

DMA ディスクリプタウィンドウレジスタ (DDWR) の構成	101
------------------------------------	-----

DMA ディスクリプタチャンネル指定レジスタ

DMA ディスクリプタチャンネル指定レジスタ (DCSR)	95
----------------------------------	----

DMA データカウンタ	
DMA データカウンタ (DDCTH/DDCTL)	102
DMA データカウンタ (DDCTH/DDCTL) の設定値について	102
DMA バッファアドレスポインタ	
DMA バッファアドレスポインタ (DBAPH/DBAPM/DBAPL)	106
DPR	
ダイレクトページレジスタ (DPR)< 初期値 :01 _H >	44
DQ3	
セクタ消去タイマフラグ (DQ3) の状態遷移	603
DQ5	
タイミングリミット超過フラグ (DQ5) の状態遷移	602
DQ6	
トグルビットフラグ (DQ6) の状態遷移	601
DQ7	
データポーリングフラグ (DQ7) の状態遷移	599
DSRH	
DMA ステータスレジスタ (DSRH/DSRL)	97
DSRL	
DMA ステータスレジスタ (DSRH/DSRL)	97
DSSR	
DMA ストップステータスレジスタ (DSSR)	98
DTB	
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	43
DTP	
DTP/ 外部割込みの概要	442
DTP/ 外部割込みの動作手順	448
DTP/ 外部割込みのブロックダイヤグラム	442
DTP/ 外部割込みのレジスタ一覧	443
DTP の動作	446
DTP/ 割込み許可レジスタ	
DTP/ 割込み許可レジスタ (ENIR:Enable Interrupt Request Register)	443
DTP/ 割込み要因レジスタ	
DTP/ 割込み要因レジスタ (EIRR:External Interrupt Request Register)	444
E	
E ² PROM	
E ² PROM のメモリマップ	582
システム構成と E ² PROM のメモリ構成	582
修正プログラムを E ² PROM に格納する場合のアドレス一致検出機能の動作	584
EI ² OS	
μDMAC または EI ² OS を使用した変換動作	468
UART の EI ² OS 機能	519
UART の割込みと EI ² OS および μDMAC	519
拡張インテリジェント I/O サービス (EI ² OS)	80
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)	88

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)	84
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	82
拡張インテリジェント I/O サービス (EI ² OS) の使用手順	87
拡張インテリジェント I/O サービス (EI ² OS) の処理手順	86
拡張インテリジェント I/O サービス (EI ² OS) の動作	81
タイムベースタイマの割込みと EI ² OS, μDMAC	222
単発モードの EI ² OS 起動プログラム例	472
停止モードの EI ² OS 起動プログラム例	478
連続モードの EI ² OS 起動プログラム例	475
EIRR	
DTP/ 割込み要因レジスタ (EIRR:External Interrupt Request Register)	444
ELVR	
要求レベル設定レジスタ (ELVR:External Level Register)	444
ENIR	
DTP/ 割込み許可レジスタ (ENIR:Enable Interrupt Request Register)	443
EOF 設定レジスタ	
EOF 設定レジスタ (HEOF)	351
EP0C	
EP0 制御レジスタ (EP0C)	287
EP0DT	
EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT)	308
EP0IS	
EP0I ステータスレジスタ (EP0IS)	299
EP0I ステータスレジスタ	
EP0I ステータスレジスタ (EP0IS)	299
EP0OS	
EP0O ステータスレジスタ (EP0OS)	301
EP0O ステータスレジスタ	
EP0O ステータスレジスタ (EP0OS)	301
EP0 制御レジスタ	
EP0 制御レジスタ (EP0C)	287
EP0 ~ EP5 データレジスタ	
EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT)	308
EP1C	
EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C)	289
EP1S	
EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S)	304
EP1 ~ EP5 ステータスレジスタ	
EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S)	304
EP1 ~ EP5 制御レジスタ	
EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C)	289
EP5C	
EP1 ~ EP5 制御レジスタ (EP1C ~ EP5C)	289
EP5DT	
EP0 ~ EP5 データレジスタ (EP0DT ~ EP5DT)	308
EP5S	
EP1 ~ EP5 ステータスレジスタ (EP1S ~ EP5S)	304

EPCR	
バス制御信号選択レジスタ (EPCR)	192

F

F ² MC-16LX	
F ² MC-16LX 命令一覧表	668
FMCS	
フラッシュメモリコントロールステータス レジスタ (FMCS)	593
FRAME 設定レジスタ	
FRAME 設定レジスタ (HFRAME)	352

H

HACR	
外部アドレス出力制御レジスタ (HACR)	191
HADR	
ホストアドレスレジスタ (HADR)	350
HCNT	
ホストコントロールレジスタ 0,1(HCNT0/HCNT1)	337
HEOF	
EOF 設定レジスタ (HEOF)	351
HERR	
ホストエラーステータスレジスタ (HERR)	344
HFCOMP	
SOF 割込み FRAME 比較レジスタ (HFCOMP)	348
HFRAME	
FRAME 設定レジスタ (HFRAME)	352
HIRQ	
ホスト割込みレジスタ (HIRQ)	341
HRTIMER	
リトライタイマ設定レジスタ (HRTIMER)	349
HSTATE	
ホスト状態ステータスレジスタ (HSTATE)	346
HTOKEN	
ホストトークンエンドポイントレジスタ (HTOKEN)	353

I

I/O	
I/O 領域	28
I/O ポート	
I/O ポートの機能	206
I/O ポートのレジスタ	207
I/O レジスタアドレスポインタ	
I/O レジスタアドレスポインタ (IOA)	83
I ² C	
I ² C インタフェースの機能	546
I ² C インタフェースの転送フロー	564
I ² C インタフェースの動作フロー	567
I ² C インタフェースのブロックダイアグラム	547
I ² C インタフェースのモードフロー	566
I ² C インタフェースのレジスタ一覧	548

I ² C バスアドレスレジスタ	
I ² C バスアドレスレジスタ 0 ~ 2 (IADR0 ~ IADR2)	559
I ² C バスクロックコントロールレジスタ	
I ² C バスクロックコントロールレジスタ 0 ~ 2 (ICCR0 ~ ICCR2)	557
I ² C バスコントロールレジスタ	
I ² C バスコントロールレジスタ 0 ~ 2 (IBCR0 ~ IBCR2)	551
I ² C バスコントロールレジスタ 0 ~ 2 (IBCR0 ~ IBCR2) 使用上の注意	556
I ² C バスステータスレジスタ	
I ² C バスステータスレジスタ 0 ~ 2 (IBSR0 ~ IBSR2)	549
I ² C バスデータレジスタ	
I ² C バスデータレジスタ 0 ~ 2(IDAR0 ~ IDAR2)	560
IADR	
I ² C バスアドレスレジスタ 0 ~ 2 (IADR0 ~ IADR2)	559
IBCR	
I ² C バスコントロールレジスタ 0 ~ 2 (IBCR0 ~ IBCR2)	551
I ² C バスコントロールレジスタ 0 ~ 2 (IBCR0 ~ IBCR2) 使用上の注意	556
IBSR	
I ² C バスステータスレジスタ 0 ~ 2 (IBSR0 ~ IBSR2)	549
ICCR	
I ² C バスクロックコントロールレジスタ 0 ~ 2 (ICCR0 ~ ICCR2)	557
ICR	
割込み制御レジスタ (ICR) の構成	63
割込み制御レジスタ (ICR00 ~ ICR15)	61
ICS	
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 (ICS01,ICS23)	266
IDAR	
I ² C バスデータレジスタ 0 ~ 2(IDAR0 ~ IDAR2)	560
ILM	
割込みレベルマスキングレジスタ (ILM)	40
IN	
IN,OUT,SETUP トークン	373
IOA	
I/O レジスタアドレスポインタ (IOA)	83
IPCP	
インプットキャプチャデータレジスタ (IPCP0 ~ IPCP3)	265
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)	84
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	82
L	
LPMCR	
低消費電力モード制御レジスタ (LPMCR)	153

LQFP-120	
外形寸法図 (LQFP-120)	8, 9
端子配列図 (LQFP-120)	10

M

M0	
バスモードの設定ビット (M1,M0)	185
M1	
バスモードの設定ビット (M1,M0)	185
MB90330A シリーズ	
MB90330A シリーズの特長	2
MB90330A シリーズのブロックダイアグラム	6
MD	
モード端子 (MD2 ~ MD0) の設定	183
μDMAC	
μDMAC 使用手順	108
μDMAC の機能	93
μDMAC の動作	107
μDMAC または EI ² OS を使用した変換動作	468
μDMAC レジスタ一覧	94
UART の割込みと EI ² OS および μDMAC	519
タイムベースタイマの割込みと EI ² OS, μDMAC	222

N

NCC	
フラグ変化抑止ブリフィックス (NCC)	47
NULL	
NULL 転送モード	327

O

OCCP	
アウトプットコンペアレジスタ	
(OCCP0 ~ OCCP3)	260
OCS	
アウトプットコンペアコントロールレジスタ	
(OCS0 ~ OCS3)	261
ODR	
ポート 4 出力端子レジスタ (ODR4)	212
OUT	
IN,OUT,SETUP トークン	373

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)	577
PADR	
プログラムアドレス検出レジスタ	
(PADR0H,PADR0M,PADR0L,PADR1H,	
PADR1M, PADR1L)	579, 580
PC	
プログラムカウンタ (PC)	42
PCB	
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	43

PDR	
ポートデータレジスタ (PDR0 ~ PDRB)	208
PLL クロック	
PLL クロック通倍率の選択	141
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,	
サブクロックモード	140
PPG	
PPG0/PPG2/PPG4 動作モード制御レジスタ	
(PPGC0/PPGC2/PPGC4)	428
PPG0 ~ PPG5 出力制御レジスタ	
(PPG01/PPG23/PPG45)	433
PPG1/PPG3/PPG5 動作モード制御レジスタ	
(PPGC1/PPGC3/PPGC5)	430
PPG 出力動作	437
PPGC	
PPG0/PPG2/PPG4 動作モード制御レジスタ	
(PPGC0/PPGC2/PPGC4)	428
PPG1/PPG3/PPG5 動作モード制御レジスタ	
(PPGC1/PPGC3/PPGC5)	430
PPG タイマ	
8/16 ビット PPG タイマの概略仕様	424
8/16 ビット PPG タイマの動作概略	436
8/16 ビット PPG タイマのブロックダイアグラム	425
8/16 ビット PPG タイマのレジスタ一覧	427
8/16 ビット PPG タイマの割込み	439
PPG リロードレジスタ	
PPG リロードレジスタ	
(PRL0 ~ PRL5, PRLH0 ~ PRLH5)	435
PRLH	
PPG リロードレジスタ	
(PRL0 ~ PRL5, PRLH0 ~ PRLH5)	435
PRL	
PPG リロードレジスタ	
(PRL0 ~ PRL5, PRLH0 ~ PRLH5)	435
PS	
プロセッサステータス (PS)	39
PWC	
PWC 分周比制御レジスタ (DIVR)	385
PWCR	
PWC データバッファレジスタ (PWCR)	384
PWCSR	
PWC コントロールステータスレジスタ (PWCSR)	379
PWC コントロールステータスレジスタ	
PWC コントロールステータスレジスタ (PWCSR)	379
PWC タイマ	
PWC タイマ動作概略	386
PWC タイマの機能	376
PWC タイマの使用上の注意	401
PWC タイマのブロックダイアグラム	377
PWC タイマのレジスタ一覧	378
PWC データバッファレジスタ	
PWC データバッファレジスタ (PWCR)	384
PWC 分周比制御レジスタ	
PWC 分周比制御レジスタ (DIVR)	385

R

RAM	
RAM 領域	28
RDR	
ポート 0,1 プルアップ抵抗レジスタ (RDR0,RDR1)	212
ROM	
ROM 領域	28
ROMM	
ROMM(ROM ミラー機能選択レジスタ)	571
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 ブロックダイアグラム	570
ROM ミラー機能選択モジュールのレジスタ	570
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)	571
RP	
レジスタバンクポインタ (RP)	40

S

S0	
各種モードの設定ビット (S1,S0)	184
S1	
各種モードの設定ビット (S1,S0)	184
SCR	
シリアル制御レジスタ 0 ~ 3(SCR0 ~ SCR3)	507
SDCR	
通信プリスケラ制御レジスタ (SDCR)	490
SDR	
シリアルデータレジスタ (SDR)	489
SETUP	
IN,OUT,SETUP トークン	373
SIDR	
シリアル入力データレジスタ 0 ~ 3 (SIDR0 ~ SIDR3)	514
SLP	
STP,SLP,TMD ビットの優先順位	155
SMCS	
シリアルモードコントロールステータスレジスタ (SMCS)	484
SMR	
シリアルモードレジスタ 0 ~ 3(SMR0 ~ SMR3)	509
SODR	
シリアル出力データレジスタ 0 ~ 3 (SODR0 ~ SODR3)	515
SOF トークン	
SOF トークン	374
SOF 割込み	
SOF 割込み	365
SOF 割込み FRAME 比較レジスタ	
SOF 割込み FRAME 比較レジスタ (HFCOMP)	348
SSB	
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	43

SSP

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)	38
---	----

SSR

シリアルステータスレジスタ 0 ~ 3 (SSR0 ~ SSR3)	511
--	-----

STP

STP,SLP,TMD ビットの優先順位	155
----------------------------	-----

T

TBTC

タイムベースタイマ制御レジスタ (TBTC)	220
------------------------------	-----

TCCS

タイマコントロールステータスレジスタ (TCCS)	255
---------------------------------	-----

TCDT

タイマカウンタデータレジスタ (TCDT)	255
-----------------------------	-----

TMCSR

タイマ制御ステータスレジスタ 0 ~ 2 (TMCSR0 ~ TMCSR2)	409
---	-----

TMD

STP,SLP,TMD ビットの優先順位	155
----------------------------	-----

TMR

16 ビットタイマレジスタ 0 ~ 2 (TMR0 ~ TMR2)	413
16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/ 16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2)	413

TMRLR

16 ビットタイマレジスタ 0 ~ 2(TMR0 ~ TMR2)/ 16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2)	413
16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2)	414

TMSP

タイムスタンプレジスタ (TMSP)	293
--------------------------	-----

U

UART

UART 使用上の注意	541
UART の EI ² OS 機能	519
UART の機能	500
UART の端子	505
UART の動作	528
UART のプログラム例	542
UART のブロックダイアグラム	502
UART のレジスタ一覧	506
UART の割込み	518
UART の割込みと EI ² OS および μ DMAC	519
UART ボーレート選択	524
UART プリスケラ制御レジスタ	
UART プリスケラ制御レジスタ 0 ~ 3 (UTCR0 ~ UTCR3), UART プリスケラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3)	516
UART プリスケラリロードレジスタ	
UART プリスケラ制御レジスタ 0 ~ 3 (UTCR0 ~ UTCR3), UART プリスケラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3)	516

UDCC	
UDC 制御レジスタ (UDCC)	284
UDCIE	
UDC 割込み許可レジスタ (UDCIE)	297
UDCS	
UDC ステータスレジスタ (UDCS)	294
UDC ステータスレジスタ	
UDC ステータスレジスタ (UDCS)	294
UDC 制御レジスタ	
UDC 制御レジスタ (UDCC)	284
UDC 割込み許可レジスタ	
UDC 割込み許可レジスタ (UDCIE)	297
USB	
USB ホストの制限事項	331
相手先 USB 装置の転送速度の取得と クロック選択	356
外部 USB 装置が非接続の状態、 接続の状態	356
外部 USB 装置の接続検出	356
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	43
USB バス	
USB バスのリセット前後の注意点	358
USB ファンクション	
USB ファンクションの動作	309
USB ファンクションの特長	278
USB ファンクションのブロックダイアグラム	279
USB ファンクションのレジスター一覧	280
USB ホスト	
USB ホストの特長	330
USB ホストの動作	355
USB ホストのブロックダイアグラム	332
USB ホストのレジスタ	334
USB 割込み	
USB 割込みの種類と機能	58, 642
USP	
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)	38
UTCR	
UART プリスケアラ制御レジスタ 0 ~ 3 (UTCR0 ~ UTCR3), UART プリスケアラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3)	516
UTRLR	
UART プリスケアラ制御レジスタ 0 ~ 3 (UTCR0 ~ UTCR3), UART プリスケアラリロードレジスタ 0 ~ 3 (UTRLR0 ~ UTRLR3)	516
W	
WDTC	
ウォッチドッグタイマ制御レジスタ (WDTC)	232
WTC	
時計タイマ制御レジスタ (WTC) の構成	244

あ

アービトレーション	
アービトレーション	562
相手先 USB 装置	
相手先 USB 装置の転送速度の取得と クロック選択	356
アウトプットコンペア	
アウトプットコンペアのブロックダイアグラム	260
アウトプットコンペアのレジスター一覧	259
アウトプットコンペアコントロールレジスタ アウトプットコンペアコントロールレジスタ (OCS0 ~ OCS3)	261
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 ~ OCCP3)	260
アキュムレータ	
アキュムレータ (A)	37
アクセスモード	
アクセスモード	182
アクセス領域	
アクセス領域と物理アドレスの関係	185
アクノリッジ	
アクノリッジ	562
アドレス	
アドレス作成の方式	29
アドレス一致検出	
アドレス一致検出機能のプログラム例	586
アドレス一致検出機能	
アドレス一致検出機能の概要	574
アドレス一致検出機能の動作	581
アドレス一致検出機能のブロックダイアグラム	575
アドレス一致検出機能のレジスタと 初期値の一覧	576
修正プログラムを E ² PROM に格納する場合の アドレス一致検出機能の動作	584
アドレッシング	
アドレッシング	561, 645
間接アドレッシング	654
直接アドレッシング	647
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)	213

い

一時停止	
フラッシュメモリのセクタ消去を一時停止	611
イベントカウントモード	
イベントカウントモード	421
イベントカウントモード (外部クロックモード)	405
インターバルタイマ	
インターバルタイマ機能 (タイムベースタイマ) の動作	223
インターバルタイマ機能	
インターバルタイマ機能	216
インターバル割込み	
時計タイマのインターバル割込み機能	246

インプットキャプチャ	
インプットキャプチャ	264
インプットキャプチャの取込みタイミング例	272
インプットキャプチャのブロックダイアグラム	264
インプットキャプチャのレジスター一覧	265
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	
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 (ICS01, ICS23)	266
インプットキャプチャデータレジスタ	
インプットキャプチャデータレジスタ (IPCP0 ~ IPCP3)	265

う

ウェイクアップ	
ホストからウェイクアップ	322
ウォッチドッグタイマ	
ウォッチドッグタイマ機能	230
ウォッチドッグタイマ使用上の注意	238
ウォッチドッグタイマのクロックソースの指定機能	247
ウォッチドッグタイマの動作	236
ウォッチドッグタイマのプログラム例	239
ウォッチドッグタイマのブロックダイアグラム	234
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)	232

え

エラーステータス	
エラーステータス	367

お

オペランド	
24 ビットオペランド指定	30

か

外形寸法図	
外形寸法図 (LQFP-120)	8, 9
外部 16 ビットバス	
外部 16 ビットバスモード (外部データバス 16 ビット / マルチプレクスモード)	197
外部 USB 装置	
外部 USB 装置が非接続の状態, 接続の状態	356
外部 USB 装置の接続検出	356
外部アドレス出力制御レジスタ	
外部アドレス出力制御レジスタ (HACR)	191
外部クロック	
イベントカウントモード (外部クロックモード)	405
外部クロック (1 対 1 モード) によるボーレート	527

外部クロックの専用ボーレートジェネレータによるボーレート	526
振動子と外部クロックの接続	145
外部シフトクロック	
外部シフトクロックモード	492
外部端子	
各モードにおける外部端子の動作	187
外部データバス 16 ビット / マルチプレクス	
外部 16 ビットバスモード (外部データバス 16 ビット / マルチプレクスモード)	197
外部バス 16 ビットデータバス	
外部バス 16 ビットデータバスモードおよびノンマルチプレクス 16 ビット外部バスモードの端子状態	173
外部バス 16 ビットデータバスモード	
外部バス 16 ビットデータバスモードおよびマルチプレクス 16 ビット外部バスモードの端子状態	169
外部バス 8 ビットデータバス	
外部バス 8 ビットデータバスモードおよびノンマルチプレクス 8 ビット外部バスモードの端子状態	175
外部バス 8 ビットデータバスモード	
外部バス 8 ビットデータバスモードおよびマルチプレクス 8 ビット外部バスモードの端子状態	171
外部メモリ	
外部メモリアクセス制御信号	195
外部メモリアクセスの入出力信号端子	188
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	124
外部割込み	
DTP/ 外部割込みの概要	442
DTP/ 外部割込みの動作手順	448
DTP/ 外部割込みのブロックダイアグラム	442
DTP/ 外部割込みのレジスター一覧	443
外部割込みの動作	446
外部割込み要求レベル	448
カウンタ動作	
カウンタ動作の状態遷移	416
カウンタ動作のモード	406
測定モードとカウンタ動作	397
カウントクロック	
カウントクロックと最大周期	394
カウントクロックの選択	389, 438
カウントタイミング	
フリーランタイムのカウントタイミング	273
書込み	
書込み / チップセクタ消去時	601, 602
書込み動作時	599
シリアルオンボード書込み基本構成	614
シリアル書込み接続例	618
富士通標準シリアルオンボード書込みに使用する端子	615
フラッシュメモリの書込み / 消去	604
フラッシュメモリの書込み / 消去の方法	590
フラッシュメモリの書込み手順	606
フラッシュメモリへのデータ書込み	606
書込みタイミング	
リロードレジスタへの書込みタイミング	440
拡張 I/O	
拡張 I/O シリアルインタフェースの概要	482

拡張 I/O シリアルインタフェースの動作概要	491
拡張 I/O シリアルインタフェースの ブロックダイアグラム	482
拡張 I/O シリアルインタフェースのレジスタ一覧	483
拡張 I/O シリアルインタフェースの割込み機能	497
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)	80
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)	88
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)	84
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	82
拡張インテリジェント I/O サービス (EI ² OS) の 使用手順	87
拡張インテリジェント I/O サービス (EI ² OS) の 処理手順	86
拡張インテリジェント I/O サービス (EI ² OS) の 動作	81
間接アドレッシング 間接アドレッシング	654
き	
起動プログラム 単発モードの EI ² OS 起動プログラム例	472
停止モードの EI ² OS 起動プログラム例	478
連続モードの EI ² OS 起動プログラム例	475
キャプチャタイミング 入力信号に対するキャプチャタイミング	275
く	
クリアタイミング フリーランタイムのクリアタイミング (コンペアクリアレジスタとの一致)	273
クロック 相手先 USB 装置の転送速度の取得と クロック選択	356
イベントカウントモード (外部クロックモード)	405
外部クロックの専用ポーレートジェネレータに よるポーレート	526
クロック供給マップ	133
クロックの概要	132
振動子と外部クロックの接続	145
内部クロックの専用ポーレートジェネレータに よるポーレート	525
内部クロックモード	405
発振クロック周波数とシリアルクロック入力 周波数	616
クロック供給 クロック供給機能	217, 224
クロック供給マップ	133
クロック選択レジスタ クロック選択レジスタ (CKSCR) の構成	137

クロックソース ウォッチドッグタイマのクロックソースの 指定機能	247
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	135
クロックモード クロックモード	149
クロックモードの移行	140
クロックモードの切替え	178

け

検出アドレス 検出アドレスの設定	581
---------------------	-----

こ

コマンドシーケンス コマンドシーケンス表	595
コモン コモンレジスタバンクプリフィックス (CMR)	47
コンディションコード コンディションコードレジスタ (CCR)	39
コンペアクリアレジスタ コンペアクリアレジスタ (CPCLR)	254
フリーランタイムのクリアタイミング (コンペアクリアレジスタとの一致)	273

さ

再開 フラッシュメモリのセクタ消去を再開	612
再起動 再起動	391
最小限の接続例 フラッシュマイコンプログラマとの最小限の 接続例 (ユーザ電源使用時)	621
最小入力パルス幅 最小入力パルス幅	398
最大周期 カウントクロックと最大周期	394
サスペンド サスペンド処理	321
サスペンド動作 サスペンド動作	369
サブクロック サブクロックの発振安定時間待ち機能	247
サブクロックモード メインクロックモード、PLL クロックモード、 サブクロックモード	140

し

システム構成 システム構成と E ² PROM のメモリ構成	582
システムスタックポインタ 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)	38
実効アドレス 実効アドレスフィールド	663

実効アドレスフィールド	
実効アドレスフィールド	646
実行サイクル	
実行サイクル数	661
実行サイクル数計算方法	661
自動アルゴリズム	
自動アルゴリズム終了タイミング	594
自動レディ機能選択レジスタ	
自動レディ機能選択レジスタ (ARSR)	190
シフトクロック	
外部シフトクロックモード	492
内部シフトクロックモード	492
シフト動作	
シフト動作のスタート/ストップタイミングと 入出力のタイミング	495
周期	
タイマ周期	394
周期カウンタ	
パルス幅/周期カウンタの範囲	399
周期算出方法	
パルス幅/周期算出方法	399
修正プログラム	
修正プログラムを E ² PROM に格納する場合の アドレス一致検出機能の動作	584
周辺装置	
外部に接続する周辺装置の条件	448
終了タイミング	
自動アルゴリズム終了タイミング	594
受信割込み	
受信割込み発生とフラグセットのタイミング	520
出力制御レジスタ	
PPG0 ~ PPG5 出力制御レジスタ (PPG01/PPG23/PPG45)	433
出力波形例	
出力波形例	270
消去	
フラッシュメモリの書込み/消去	604
フラッシュメモリの書込み/消去の方法	590
フラッシュメモリの全データを消去 (チップ消去)	608
フラッシュメモリの任意のデータを消去 (セクタ消去)	609
状態遷移	
カウンタ動作の状態遷移	416
セクタ消去タイマフラグ (DQ3) の状態遷移	603
タイミングリミット超過フラグ (DQ5) の 状態遷移	602
データポーリングフラグ (DQ7) の状態遷移	599
トグルビットフラグ (DQ6) の状態遷移	601
状態遷移図	
状態遷移図	166
消費電流	
CPU 動作モードと消費電流	148
初期値	
アドレス一致検出機能のレジスタと 初期値の一覧	576
ハードウェアコンポーネントの初期値	439
処理時間	
ハードウェア割込み処理時間	76

シリアル I/O	
シリアル I/O の動作状態	493
シリアルインタフェース	
拡張 I/O シリアルインタフェースの概要	482
拡張 I/O シリアルインタフェースの動作概要	491
拡張 I/O シリアルインタフェースの ブロックダイヤグラム	482
拡張 I/O シリアルインタフェースのレジスタ一覧	483
拡張 I/O シリアルインタフェースの割込み機能	497
シリアルオンボード	
シリアルオンボード書込み基本構成	614
富士通標準シリアルオンボード書込みに使用する 端子	615
シリアル書込み	
シリアル書込み接続例	618
シリアル書込み接続	
富士通マイクロエレクトロニクス製シリアル プログラマ	624
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	616
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 0 ~ 3 (SODR0 ~ SODR3)	515
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 0 ~ 3 (SSR0 ~ SSR3)	511
シリアル制御レジスタ	
シリアル制御レジスタ 0 ~ 3 (SCR0 ~ SCR3)	507
シリアルデータ	
シリアルデータ転送中の動作	496
シリアルデータレジスタ	
シリアルデータレジスタ (SDR)	489
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 0 ~ 3 (SIDR0 ~ SIDR3)	514
シリアルモードコントロールステータスレジスタ	
シリアルモードコントロールステータスレジスタ (SMCS)	484
シリアルモードレジスタ	
シリアルモードレジスタ 0 ~ 3 (SMR0 ~ SMR3)	509
シングルチップモード	
シングルチップモード時の接続例 (ユーザ電源使用時)	619
シングルチップモード時の端子状態	168
振動子	
振動子と外部クロックの接続	145

す

推奨例	
モード端子とモードデータの関係 (推奨例)	186
スタート/ストップタイミング	
シフト動作のスタート/ストップタイミングと 入出力のタイミング	495

スタートコンディション	
スタートコンディション	561
スタック	
スタック領域	111
割込み処理開始時のスタック動作	110
割込み処理復帰時のスタック動作	110
スタンバイモード	
スタンバイモード	149
スタンバイモード時の動作状態	157
スタンバイモードの割込みによる解除	177
スタンバイモードへの移行と割込み	177
ステータスレジスタ	
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)	84
ストップコンディション	
ストップコンディション	561
ストップモード	
ストップモードの解除	164
ストップモードへの移行	164
スリープモード	
スリープモードの解除	159
スリープモードへの移行	158

せ

制限事項	
USB ホストの制限事項	331
セクタ	
セクタ消去一時停止時	599, 601, 603
セクタ消去動作時	603
チップ / セクタ消去動作時	599
セクタ構成	
セクタ構成	591
セクタ消去	
セクタ消去の手順	609
フラッシュメモリのセクタ消去を一時停止	611
フラッシュメモリのセクタ消去を再開	612
フラッシュメモリの任意のデータを消去 (セクタ消去)	609
セクタ消去タイマフラグ	
セクタ消去タイマフラグ (DQ3) の状態遷移	603
接続	
外部 USB 装置が非接続の状態、接続の状態	356
接続検出	
外部 USB 装置の接続検出	356
接続例	
シリアル書込み接続例	618
シングルチップモード時の接続例 (ユーザ電源使用時)	619
フラッシュマイコンプログラムの最小限の接続例 (ユーザ電源使用時)	621
設定ビット	
各種モードの設定ビット (S1,S0)	184
バスモードの設定ビット (M1,M0)	185
専用ボーレートジェネレータ	
外部クロックの専用ボーレートジェネレータによるボーレート	526
内部クロックの専用ボーレートジェネレータによるボーレート	525

専用レジスタ	
専用レジスタ	34

そ

送信割込み	
送信割込み発生とフラグセットのタイミング	522
双方向通信	
双方向通信機能	536
測定結果	
測定結果のデータ	396
測定モード	
測定モードとカウンタ動作	397
ソフトウェア割込み	
ソフトウェア割込みからの復帰	78
ソフトウェア割込みに関する注意事項	79
ソフトウェア割込みの起動	78
ソフトウェア割込みの動作	79

た

タイマ	
タイマ / パルス幅測定の起動と停止	391
タイマ周期	394
タイマのクリア	393
タイマの動作フロー	395
タイマカウンタデータレジスタ	
タイマカウンタデータレジスタ (TCDT)	255
タイマ機能	
タイマ機能の動作	387
発振安定待ち時間用タイマ機能	223
タイマコントロールステータスレジスタ	
タイマコントロールステータスレジスタ (TCCS)	255
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 0 ~ 2 (TMCSR0 ~ TMCSR2)	409
タイマ値	
タイマ値とリロード値	393
タイマレジスタ	
16 ビットタイマレジスタ 0 ~ 2 (TMR0 ~ TMR2)	413
タイミングリミット超過フラグ	
タイミングリミット超過フラグ (DQ5) の状態遷移	602
タイムスタンプレジスタ	
タイムスタンプレジスタ (TMSP)	293
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) の動作	223
タイムベースタイマ使用上の注意	225
タイムベースタイマの動作	225
タイムベースタイマのプログラム例	227
タイムベースタイマのブロックダイアグラム	218
タイムベースタイマの割込み	222
タイムベースタイマの割込みと EI ² OS, μ DMAC	222
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)	220

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	160
タイムベースタイマモードへの移行	160
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) < 初期値 :01 _H >	44
多重割込み	
多重割込み動作	74
多重割込みの例	74
多バイト長	
多バイト長データのアクセス	33
メモリ空間における多バイト長データの配置	33
端子	
端子機能	11
端子状態	
シングルチップモード時の端子状態	168
端子配列図	
端子配列図 (LQFP-120)	10
単発測定	
単発測定と連続測定	396
単発変換	
単発変換モードの動作	465
単発モード	
単発モードの EI ² OS 起動プログラム例	472

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使用上の注意 (遅延割込み要求ラッチ)	117
遅延割込み発生モジュールの動作	117
遅延割込み発生モジュールのブロックダイヤグラム	116
遅延割込み発生モジュールのレジスタ一覧	116
遅延割込み要求ラッチ	
遅延割込み発生モジュールの使用上の注意 (遅延割込み要求ラッチ)	117
チップ	
チップ / セクタ消去動作時	599
チップ消去	
フラッシュメモリの全データを消去 (チップ消去)	608
チップセクタ	
書き込み / チップセクタ消去時	601, 602
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 (LPCMR) へアクセスする際の注意事項	179
直接アドレッシング	
直接アドレッシング	647

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 (SDCR)	490

て

停止	
停止	392
停止変換	
停止変換モードの動作	466
停止モード	
停止モードの EI ² OS 起動プログラム例	478
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ヤグラム	151
低消費電力モード	
低消費電力モードの動作状態	167
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)	153
低消費電力モード制御レジスタへのアクセス	155
ディスクリプタ	
DMA ディスクリプタの各レジスタ	101
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	82
通倍率	
PLL クロック通倍率の選択	141
データカウンタ	
データカウンタ (DCT)	83
データ数自動転送モード	
データ数自動転送モード	324
データパケット	
データパケット	362
データバス	
外部バス 16 ビットデータバスモードおよびノンマルチプレクス 16 ビット外部バスモードの端子状態	173
外部バス 16 ビットデータバスモードおよびマルチプレクス 16 ビット外部バスモードの端子状態	169
外部バス 8 ビットデータバスモードおよびノンマルチプレクス 8 ビット外部バスモードの端子状態	175
外部バス 8 ビットデータバスモードおよびマルチプレクス 8 ビット外部バスモードの端子状態	171
データポーリングフラグ	
データポーリングフラグ (DQ7) の状態遷移	599
デバイス	
デバイス使用上の注意	21
デバイスの切断	372
転送時間	
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)	88
転送速度	
相手先 USB 装置の転送速度の取得とクロック選択	356
転送モード	
NULL 転送モード	327
パケット転送モード	323

と

同期モード	
同期モード (動作モード 2) 時の動作	533

動作状態	
シリアル I/O の動作状態	493
設定と動作状態	583
低消費電力モードの動作状態	167
動作状態の確認	392
動作モード	
16 ビットリロードタイマの動作モード	405
CPU 動作モードと消費電流	148
同期モード (動作モード 2) 時の動作	533
動作モード	182, 436
動作モードの選択	390
動作モード制御レジスタ	
PPG0/PPG2/PPG4 動作モード制御レジスタ (PPGC0/PPGC2/PPGC4)	428
PPG1/PPG3/PPG5 動作モード制御レジスタ (PPGC1/PPGC3/PPGC5)	430
トークン	
IN,OUT,SETUP トークン	373
SOF トークン	374
トークンパケット	
トークンパケットの設定	359
トグルビットフラグ	
トグルビットフラグ (DQ6) の状態遷移	601
時計カウンタ	
時計カウンタ	246
時計タイマ	
時計タイマのインターバル割込み機能	246
時計タイマの機能	242
時計タイマのブロックダイアグラム	243
時計タイマ制御レジスタ	
時計タイマ制御レジスタ (WTC) の構成	244
時計モード	
時計モードの解除	162
時計モードへの遷移	162

な

内部クロック	
内部クロックの専用ポーレートジェネレータによるポーレート	525
内部クロックモード	405
内部クロックモード (リロードモード) の動作	417
内部クロックモード (ワンショットモード)	419
内部シフトクロック	
内部シフトクロックモード	492

に

入出力回路形式	
入出力回路形式	18
入出力タイマ	
16 ビット入出力タイマの構成および機能	250
16 ビット入出力タイマの動作, タイミング	268
16 ビット入出力タイマのレジスタ構成	252
入出力のタイミング	
シフト動作のスタート / ストップタイミングと入出力のタイミング	495

の

ノンマルチプレクス	
外部バス 16 ビットデータバスモードおよびノンマルチプレクス 16 ビット外部バスモードの端子状態	173
外部バス 8 ビットデータバスモードおよびノンマルチプレクス 8 ビット外部バスモードの端子状態	175
ノンマルチプレクスモード	202

は

ハードウェアコンポーネント	
ハードウェアコンポーネントの初期値	439
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	597
ハードウェア割込み	
ハードウェア割込みからの復帰	69
ハードウェア割込み処理時間	76
ハードウェア割込みの起動	69
ハードウェア割込みの機能	66
ハードウェア割込みの構造	67
ハードウェア割込みの使用手順	72
ハードウェア割込みの動作	70
ハードウェア割込みの動作フロー	71
ハードウェア割込み抑止	67
パケット	
パケット転送モード	323
パケット終了	
パケット終了タイミング	368
バスエラー	
バスエラー	562
バス制御信号選択レジスタ	
バス制御信号選択レジスタ (EPCR)	192
バスモード	
バスモード	182
バスモードの設定ビット (M1,M0)	185
発振安定時間	
サブクロックの発振安定時間待ち機能	247
発振安定待ち	
発振安定待ち時間用タイマ機能	223
発振安定待ちリセット状態	123
発振安定待ち時間	
発振安定待ち時間	144, 178
リセット要因と発振安定待ち時間	122
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	616
パッチ処理	
プログラム修正のパッチ処理フロー	585
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)	84
パルス	
パルスの端子出力制御	438
パルス幅	
最小入力パルス幅	398
パルス幅 / 周期カウンタの範囲	399
パルス幅 / 周期算出方法	399
リロード値とパルス幅の関係	437
パルス幅測定	
タイマ / パルス幅測定の起動と停止	391

パルス幅測定機能の動作	388
パルス幅測定の動作フロー	400
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	46
バンク方式	
バンク方式によるアドレス指定	31
バンクレジスタ	
バンクレジスタ (PCB, DTB, USB, SSB, ADB)	
.....	43
ハンドシェークパケット	
ハンドシェークパケット	363
汎用レジスタ	
汎用レジスタ	36

ひ

非接続	
外部 USB 装置が非接続の状態，	
接続の状態	356
非同期モード	
非同期モード時の動作	530
品種	
品種構成	4

ふ

富士通標準	
富士通標準シリアルオンボード書込みに使用する	
端子	615
復帰	
ソフトウェア割込みからの復帰	78
ハードウェア割込みからの復帰	69
物理アドレス	
アクセス領域と物理アドレスの関係	185
フラグ	
フラグ変化抑止プリフィックス (NCC)	47
フラグセット	
受信割込み発生とフラグセットのタイミング	
.....	520
送信割込み発生とフラグセットのタイミング	
.....	522
フラッシュマイコンプログラマ	
フラッシュマイコンプログラマとの最小限の	
接続例 (ユーザ電源使用時)	621
フラッシュマイコンプログラマシステム	
フラッシュマイコンプログラマシステム	617
フラッシュメモリ	
フラッシュメモリの書込み / 消去	604
フラッシュメモリの書込み / 消去の方法	590
フラッシュメモリの書込み手順	606
フラッシュメモリのセクタ消去を一時停止	
.....	611
フラッシュメモリのセクタ消去を再開	612
フラッシュメモリの全データを消去	
(チップ消去)	608
フラッシュメモリの特長	590
フラッシュメモリの任意のデータを消去	
(セクタ消去)	609
フラッシュメモリの容量と品種	590
フラッシュメモリへのデータ書込み	606
フラッシュメモリを読み出し / リセット状態にする	
.....	605

フラッシュメモリコントロールステータスレジスタ	
フラッシュメモリコントロールステータス	
レジスタ (FMCS)	593
フリーランタイマ	
16 ビットフリーランタイマの動作	269
16 ビットフリーランタイマのブロック	
ダイヤグラム	254
16 ビットフリーランタイマのレジスタ一覧	
.....	253
フリーランタイマのカウントタイミング	273
フリーランタイマのクリアタイミング	
(コンペアクリアレジスタとの一致)	273
プリフィックス	
割込み抑止命令とプリフィックス命令に関する	
制約	49
プリフィックスコード	
プリフィックスコードが連続している場合	49
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)	577
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	
(PADROH, PADROM, PADROL, PADRIH,	
PADRIM, PADRIL)	579, 580
プログラムカウンタ	
プログラムカウンタ (PC)	42
プログラム修正	
プログラム修正のパッチ処理フロー	585
プログラムの実行	
プログラムの実行	581
プログラム例	
UART のプログラム例	542
アドレス一致検出機能のプログラム例	586
ウォッチドッグタイマのプログラム例	239
タイムベースタイマのプログラム例	227
割込み処理のプログラム例	112
プロセッサステータス	
プロセッサステータス (PS)	39
ブロックダイヤグラム	
16 ビットフリーランタイマの	
ブロックダイヤグラム	254
16 ビットリロードタイマの	
ブロックダイヤグラム	407
8/10 ビット A/D コンバータの	
ブロックダイヤグラム	453
8/16 ビット PPG タイマのブロックダイヤグラム	
.....	425
DTP/ 外部割込みのブロックダイヤグラム	442
I ² C インタフェースのブロックダイヤグラム	
.....	547
MB90330A シリーズのブロックダイヤグラム	
.....	6
PWC タイマのブロックダイヤグラム	377
ROM ミラー機能選択モジュールの	
ブロックダイヤグラム	570
UART のブロックダイヤグラム	502
USB ファンクションのブロックダイヤグラム	
.....	279
USB ホストのブロックダイヤグラム	332
アウトプットコンペアのブロックダイヤグラム	
.....	260
アドレス一致検出機能のブロックダイヤグラム	
.....	575

インプットキャプチャのブロックダイアグラム	264
ウォッチドッグタイマのブロックダイアグラム	234
外部リセット端子のブロックダイアグラム	124
拡張 I/O シリアルインタフェースの ブロックダイアグラム	482
クロック発生部のブロックダイアグラム	135
タイムベースタイマのブロックダイアグラム	218
遅延割込み発生モジュールの ブロックダイアグラム	116
低消費電力制御回路のブロックダイアグラム	151
時計タイマのブロックダイアグラム	243
ブロックダイアグラム	188, 251

へ

変化タイミング	
出力端子の変化タイミング	274
変換動作	
μDMAC または EI ² OS を使用した変換動作	468

ほ

ポート 0	
ポート 0,1 プルアップ抵抗レジスタ (RDR0,RDR1)	212
ポート 0 プルアップ抵抗レジスタ	
ポート 0,1 プルアップ抵抗レジスタ (RDR0,RDR1)	212
ポート 1	
ポート 0,1 プルアップ抵抗レジスタ (RDR0,RDR1)	212
ポート 1 プルアップ抵抗レジスタ	
ポート 0,1 プルアップ抵抗レジスタ (RDR0,RDR1)	212
ポート 4	
ポート 4 出力端子レジスタ (ODR4)	212
ポート 4 出力端子レジスタ	
ポート 4 出力端子レジスタ (ODR4)	212
ポートデータレジスタ	
ポートデータレジスタ (PDR0 ~ PDRB)	208
ポート方向レジスタ	
ポート方向レジスタ (DDR0 ~ DDRB)	210
ホールド機能	
ホールド機能の動作	201
ボーレート	
UART ボーレート選択	524
外部クロック (1 対 1 モード) によるボーレート	527
外部クロックの専用ボーレートジェネレータに よるボーレート	526
内部クロックの専用ボーレートジェネレータに よるボーレート	525
ボーレートジェネレータ	
外部クロックの専用ボーレートジェネレータに よるボーレート	526

内部クロックの専用ボーレートジェネレータに よるボーレート	525
保護機能	
A/D 変換データ保護機能	469
ホスト	
USB ホストの特長	330
USB ホストのブロックダイアグラム	332
USB ホストの制限事項	331
USB ホストの動作	355
USB ホストのレジスタ	334
ホストからウェイクアップ	322
ホスト機能の設定	356
ホストアドレスレジスタ	
ホストアドレスレジスタ (HADR)	350
ホストエラーステータスレジスタ	
ホストエラーステータスレジスタ (HERR)	344
ホストコントロールレジスタ	
ホストコントロールレジスタ 0,1 (HCNT0/HCNT1)	337
ホスト状態ステータスレジスタ	
ホスト状態ステータスレジスタ (HSTATE)	346
ホストトークンエンドポイントレジスタ	
ホストトークンエンドポイントレジスタ (HTOKEN)	353
ホスト割込みレジスタ	
ホスト割込みレジスタ (HIRQ)	341

ま

マシニングロック	
マシニングロック	141
マスタ / スレーブ型通信	
マスタ / スレーブ型通信機能	538
マルチプレクス	
外部 16 ビットバスモード (外部データバス 16 ビット / マルチプレクスモード)	197
外部バス 16 ビットデータバスモードおよび マルチプレクス 16 ビット外部バスモードの 端子状態	169
外部バス 8 ビットデータバスモードおよび マルチプレクス 8 ビット外部バスモードの 端子状態	171
マルチプレクスモード	202

み

未定義命令	
未定義命令の実行	109
未定義命令の実行による例外発生	109

め

命令	
未定義命令の実行	109
未定義命令の実行による例外発生	109
命令の種類	644
命令マップの構造	682
命令一覧表	
F ² MC-16LX 命令一覧表	668

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、サブクロックモード	140
メモリ空間	
CPU メモリ空間の概要	27
メモリ空間	630
メモリ空間における多バイト長データの配置	33
メモリ構成	
システム構成と E ² PROM のメモリ構成	582
メモリマップ	
E ² PROM のメモリマップ	582

も

モード	
外部バス 8 ビットデータバスモードおよびノンマルチプレクス 8 ビット外部バスモードの端子状態	175
モード設定	
モード設定	182
モード端子	
モード端子	125
モード端子 (MD2 ~ MD0) の設定	183
モード端子とモードデータの関係 (推奨例)	186
モードデータ	
モード端子とモードデータの関係 (推奨例)	186
モードデータ	184
モードデータ読出し後の端子の状態	129
モードの種類	
モードの種類	194
モードフェッチ	
モードフェッチ	126

ゆ

ユーザスタックポインタ	
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)	38
ユーザ電源使用時	
シングルチップモード時の接続例 (ユーザ電源使用時)	619
フラッシュマイコンプログラムの最小限の接続例 (ユーザ電源使用時)	621
優先順位	
STP,SLP,TMD ビットの優先順位	155

よ

要求レベル設定レジスタ	
要求レベル設定レジスタ (ELVR:External Level Register)	444
読出し	
フラッシュメモリを読出し / リセット状態にする	605

ら

ライトコマンド	
ライトコマンド応答時の各レジスタ動作	315

り

リードコマンド	
リードコマンド応答時の各レジスタ動作	314
リセット	
USB バスのリセット前後の注意点	358
外部リセット端子のブロックダイアグラム	124
発振安定待ちリセット状態	123
フラッシュメモリを読出し / リセット状態にする	605
リセット中の端子の状態	129
リセット動作	
リセット動作の概要	125
リセット要因	
リセット要因	120
リセット要因と発振安定待ち時間	122
リセット要因ビット	127
リセット要因ビットとリセット要因の対応	128
リセット要因ビットの注意事項	128
リトライ	
リトライ機能	364
リトライタイマ設定レジスタ	
リトライタイマ設定レジスタ (HRTIMER)	349
リモートウェイクアップ	
リモートウェイクアップ	322
リロードタイマ	
16 ビットリロードタイマの概要	404
16 ビットリロードタイマの設定	415
16 ビットリロードタイマの動作モード	405
16 ビットリロードタイマのブロックダイアグラム	407
16 ビットリロードタイマレジスタ一覧	408
リロード値	
タイマ値とリロード値	393
リロード値とパルス幅の関係	437
リロード動作	
リロード動作モード	393
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	417
リロードレジスタ	
16 ビットタイマレジスタ 0 ~ 2 (TMR0 ~ TMR2) / 16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2)	413
16 ビットリロードレジスタ 0 ~ 2 (TMRLR0 ~ TMRLR2)	414
リロードレジスタへの書き込みタイミング	440

れ

例外処理割込み	
例外処理割込み	91
例外発生	
未定義命令の実行による例外発生	109

レジスタバンク	
レジスタバンク	45
レジスタバンクポインタ (RP)	40
レジューム動作	
レジューム動作	369
レディ機能	
レディ機能	198
連続測定	
単発測定と連続測定	396
連続変換	
連続変換モードの動作	466
連続モード	
連続モードの EI ² OS 起動プログラム例	475

わ

割り込み

8/16 ビット PPG タイマの割り込み	439
DTP/ 外部割り込みの概要	442
DTP/ 外部割り込みの動作手順	448
DTP/ 外部割り込みのブロックダイヤグラム	442
SOF 割り込み	365
UART の割り込み	518
UART の割り込みと EI ² OS および μ DMAC	519
USB 割り込みの種類と機能	58
外部割り込みの動作	446
外部割り込み要求レベル	448
拡張 I/O シリアルインタフェースの割り込み機能	497
受信割り込み発生とフラグセットのタイミング	520
スタンバイモードの割り込みによる解除	177
スタンバイモードへの移行と割り込み	177
送信割り込み発生とフラグセットのタイミング	522
ソフトウェア割り込みからの復帰	78
ソフトウェア割り込みに関する注意事項	79
ソフトウェア割り込みの起動	78
ソフトウェア割り込みの動作	79
タイムベースタイマの割り込み	222
タイムベースタイマの割り込みと EI ² OS, μ DMAC	222
多重割り込み動作	74
多重割り込みの例	74
時計タイマのインターバル割り込み機能	246
ハードウェア割り込みからの復帰	69
ハードウェア割り込み処理時間	76
ハードウェア割り込みの起動	69
ハードウェア割り込みの機能	66

ハードウェア割り込みの構造	67
ハードウェア割り込みの使用手順	72
ハードウェア割り込みの動作	70
ハードウェア割り込みの動作フロー	71
ハードウェア割り込み抑止	67
例外処理割り込み	91
割り込み処理開始時のスタック動作	110
割り込み処理のプログラム例	112
割り込み処理復帰時のスタック動作	110
割り込みタイミング	274
割り込みの種類と機能	52
割り込み発生要求	393
割り込み要因と割り込みベクタ, 割り込み制御レジスタ	641
割り込み要求発生	399
割り込みレベルマスクレジスタ (ILM)	40
割り込み制御レジスタ	
割り込み制御レジスタ (ICR) の構成	63
割り込み制御レジスタ (ICR00 ~ ICR15)	61
割り込み制御レジスタ一覧	59
割り込み制御レジスタの機能	60, 64
割り込み要因と割り込みベクタ, 割り込み制御レジスタ	641
割り込み要因と割り込みベクタ / 割り込み制御レジスタ	56
割り込みベクタ	
割り込みベクタ	55
割り込み要因と割り込みベクタ, 割り込み制御レジスタ	641
割り込み要因と割り込みベクタ / 割り込み制御レジスタ	56
割り込み要因	
割り込み要因と割り込みベクタ, 割り込み制御レジスタ	641
割り込み要因と割り込みベクタ / 割り込み制御レジスタ	56
割り込み要求	
外部割り込み要求レベル	448
割り込み抑止	
ハードウェア割り込み抑止	67
割り込み抑止命令	49
割り込み抑止命令とプリフィックス命令に関する制約	49
ワンショット動作	
ワンショット動作モード	393
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)	419

CM44-10129-6

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

MB90330A Series

ハードウェアマニュアル

2009 年 10 月 第 6 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 プロモーション推進部
