



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

MB90540/545 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

MB90540/545 Series

ハードウェアマニュアル

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

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

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

富士通マイクロエレクトロニクス株式会社

はじめに

■ 本書の目的と対象読者

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

MB90540/545 シリーズは、ASIC(Application Specific IC)対応が可能なオリジナル16ビット・ワンチップマイクロコントローラである F²MC[®]-16LX ファミリの汎用品の1つとして開発された製品です。

本書は、実際にこの半導体を使って製品を設計する技術者の方を対象に、MB90540/545 シリーズの機能や動作について記載しています。本書をご一読ください。

■ 商標

F²MCは、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商標です。

■ 本書の全体構成

本書は、以下に示す25の章および付録から構成されています。

第1章 概要

MB90540/545 シリーズの特長や基本的な仕様について示します。

第2章 CPU

CPU について説明します。

第3章 割込み

割込みの機能と動作について説明します。

第4章 クロックとリセット

クロックとリセットの機能と動作について説明します。

第5章 低消費電力制御回路

低消費電力制御回路の機能と動作について説明します。

第6章 低消費電力モード

低消費電力モードの機能と動作について説明します。

第7章 メモリアクセスモード

メモリアクセスモードの機能と動作について説明します。

第8章 I/O ポート

I/O ポートの機能と動作について説明します。

第9章 タイムベースタイマ

タイムベースタイマの機能と動作について説明します。

第10章 ウォッチドッグタイマ

ウォッチドッグタイマの機能と動作について説明します。

第11章 時計タイマ

時計タイマの機能と動作について説明します。

第 12 章 16 ビット入出力タイマ

16 ビット入出力タイマの機能と動作について説明します。

第 13 章 16 ビットリロードタイマ (イベントカウント機能付)

イベントカウント機能付の 16 ビットリロードタイマ (イベントカウント機能付) の機能と動作について説明します。

第 14 章 8/16 ビット PPG

8/16 ビット PPG の機能と動作について説明します。

第 15 章 遅延割込み

遅延割込みの機能と動作について説明します。

第 16 章 DTP/ 外部割込み

DTP/ 外部割込みの機能と動作について説明します。

第 17 章 A/D コンバータ

A/D コンバータの機能と動作について説明します。

第 18 章 UART0

UART0 の機能と動作について説明します。

第 19 章 UART1(SCI)

UART1 の機能と動作について説明します。

第 20 章 シリアル I/O

シリアル I/O の機能と動作について説明します。

第 21 章 CAN コントローラ

CAN コントローラの機能と動作について説明します。

第 22 章 アドレス一致検出機能

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 23 章 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機能と動作について説明します。

第 24 章 1M/2M ビットフラッシュメモリ

1M/2M ビットフラッシュメモリの機能と動作について説明します。

第 25 章 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S)

シリアル書込み接続例

MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) のシリアル書込みの接続例について説明します。

付録

I/O マップ, 命令概要について示します。

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

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。）
452 ～ 512	「付録 B 命令」全体を変更

変更箇所は、本文中のページ左側の によって示しています。

< 参考 > 4 版 5 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。）
-	用語を統一 (受容 アクセプタンス)
-	レジスタ名を変更 (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)) (プログラムバンクレジスタ (PCB) プログラムカウンタバンクレジスタ (PCB)) (低消費電力モードレジスタ 低消費電力モード制御レジスタ) (コンペアレジスタ 0,1 アウトプットコンペアレジスタ 0,1) (コントロールステータスレジスタ 制御ステータスレジスタ (OCS0/1)) (コンペアコントロールステータスレジスタ 0,1 アウトプットコンペア制御ステータスレジスタ 0,1) (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制御ステータスレジスタ (ICS01)) (タイマコントロールステータスレジスタ タイマカウンタ制御ステータスレジスタ) (DTP/ 割込み許可レジスタ DTP/ 外部割込み許可レジスタ) (DTP/ 割込み要因レジスタ DTP/ 外部割込み要因レジスタ) (A/D コントロールステータスレジスタ (ADCS1) A/D 制御ステータスレジスタ 1 (ADCS1)) (データレジスタ下位 A/D データレジスタ 0) (データレジスタ上位 A/D データレジスタ 1) (A/D 制御レジスタ 0 A/D 制御ステータスレジスタ 0) (A/D 制御レジスタ 1 A/D 制御ステータスレジスタ 1) (データレジスタ A/D データレジスタ) (変換データレジスタ A/D データレジスタ) (ステータスレジスタ 0(USR0) シリアルステータスレジスタ 0(USR0)) (UIDR0(インプットデータレジスタ) UIDR0(シリアルインプットデータレジスタ)) (アウトプットレジスタ 0(UODR0) シリアルアウトプットデータレジスタ 0(UODR0)) (シリアルコントロールレジスタ シリアル制御レジスタ) (通信プリスケラコントロールレジスタ UART1 通信プリスケラ制御レジスタ) (UART1 通信プリスケラコントロールレジスタ UART1 通信プリスケラ制御レジスタ) (SDR(シリアルデータレジスタ) SDR(シリアルシフトデータレジスタ)) (シリアルモードコントロールステータスレジスタ (SMCS) シリアルモード制御ステータスレジスタ (SMCS)) (シフトレジスタ (SDR) シリアルシフトデータレジスタ (SDR)) (RTR 送信レジスタ 送信 RTR レジスタ) (受容マスク選択レジスタ アクセプタンスマスク選択レジスタ) (受容マスクレジスタ アクセプタンスマスクレジスタ) (全体コントロールレジスタ 全体制御レジスタ) (メッセージバッファコントロールレジスタ メッセージバッファ制御レジスタ) (CAN コントロールステータスレジスタ (CSR) 制御ステータスレジスタ (CSR)) (受容コード ID レジスタ (IDRx) アクセプタンスコード ID レジスタ (IDRx)) (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 0/1 (PADR0/PADR1)) (フラッシュメモリコントロールステータスレジスタ (FMCS) フラッシュメモリ制御ステータスレジスタ (FMCS))

< 参考 > 4 版 5 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。)
-	(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 (ECSR)) (PPG01 レジスタ PPG ユニット 0 クロック選択レジスタ) (UART1 通信プリスケアラコントロールレジスタ (CDCR) UART1 通信プリスケアラ制御レジスタ (CDCR)) (通信プリスケアラコントロールレジスタ (CDCR) UART1 通信プリスケアラ制御レジスタ (CDCR)) (シリアルステータスレジスタ (SSR) シリアルステータスレジスタ 1(SSR1) (シリアルアウトプットデータレジスタ (SODR) シリアルアウトプットデータレジスタ 1(SODR1) (シリアルコントロールレジスタ (SCR) シリアル制御レジスタ 1(SCR1))
3	表 1.2-1 MB90540/545 シリーズの特長 を変更 (8 ビットリロードカウンタ× 8 本 8 ビットリロードカウンタ× 2 本× 4 ユニット) (L パルス幅用の 8 ビットリロードレジスタ× 8 本 "L" パルス幅用の 8 ビットリロードレジスタ× 2 本× 4 ユニット) (H パルス幅用の 8 ビットリロードレジスタ× 8 本 "H" パルス幅用の 8 ビットリロードレジスタ× 2 本× 4 ユニット)
4	表 1.2-1 MB90540/545 シリーズの特長 (2 / 2) を変更 (受入れフィルタの柔軟な構成 : アクセプタンスフィルタの柔軟な構成 :) (ほとんど , 全外部端子を汎用入出力として使用可能です。 汎用入出力ポート 81 本)
10 ~ 16	表 1.6-1 端子機能説明 を変更
17 ~ 19	表 1.7-1 入出力回路形式 を変更 (ヒステリシス入力 CMOS ヒステリシス入力)
21	電源端子について (V_{CC}/V_{SS}) を変更 図 1.8-2 電源端子 (V_{CC}/V_{SS}) の取扱いを追加 水晶発振回路について を変更 (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。 の文を追加)
33	専用レジスタ を変更 (F ² MC-16LXCPU コアの専用レジスタは以下に示す 13 本があります。 F ² MC-16LXCPU コアの専用レジスタは以下に示す 11 本があります。)
41	表 2.8-1 各レジスタの機能 を変更 (RO R0) (RWO RW0)
51	表 3.2-1 割込み要因と割込みベクタ・割込み制御レジスタ を変更 (PPG 0/1 PPG(ch0, ch1) ユニット 0) (PPG 2/3 PPG (ch2, ch3) ユニット 1) (PPG 4/5 PPG (ch4, ch5) ユニット 2) (PPG 6/7 PPG (ch6, ch7) ユニット 3)
53	表 3.3-1 割込みベクター一覧表 を変更 (PPG 0/1 PPG (ch0, ch1) ユニット 0) (PPG 2/3 PPG (ch2, ch3) ユニット 1) (PPG 4/5 PPG (ch4, ch5) ユニット 2) (PPG 6/7 PPG (ch6, ch7) ユニット 3)

< 参考 > 4 版 5 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。）
55	3.4 ハードウェア割込み の要約を変更 (内蔵リソース 周辺機能) CPU を変更 (内蔵リソース 周辺機能)
60	割込みハンドリング時間 (マシンサイクル) を変更 ()
62	ソフトウェア割込みの概要 を変更 (PS レジスタ内に I を設定します。 PS : I フラグを "0" に設定します。)
63	図 3.5-1 ソフトウェア割込みの発生と解除まで を変更 (を追加)
65	図 3.6-1 拡張インテリジェント I/O サービスの概要を変更 (を追加)
70	I/O レジスタアドレスポインタ (IOA) を変更 ((A15 ~ A0) (A15 ~ A00)) (IQA I/O レジスタアドレスポインタ (IOA))
72	図 3.6-7 拡張インテリジェント I/O サービス (EI ² OS) 動作フロー を変更 (内蔵ソースにより割込み要求発生 周辺機能により割込み要求発生) (内蔵ソースに割込要求のクリア 周辺機能 割込み要求のクリア)
73	図 3.6-8 拡張インテリジェント I/O サービス (EI ² OS) の使用手順フロー を変更 (内蔵リソース 周辺機能)
83	< 注意事項 > を変更 (また, サブクロックモード時のハードウェアスタンバイリセットの発振安定待ち時間は, 原発振 2 ¹⁷ サイクル固定です。 の文を追加)
92	表 5.3-1 CG ビット設定 を変更 (OCG ビット設定を変更 CG ビット設定を変更)
93	クロック選択レジスタ (CKSCR) を変更 (書込みは, 動作に影響しません。 の文を追加)
96	図 5.4-1 クロック選択の状態遷移図 1(2 系統品その 1) を変更 (矢印)
100	6.1 低消費電力モード の要約を変更 (・ CPU 間欠動作 の記述を追加)
101	CPU 間欠動作機能 を変更 (PLL クロック逡倍率は CS1, 0 ビットを使用して設定され, クロック信号供給の 2, 4, 6, 8 倍のいずれかになります。 このクロック信号は, 2 分周してマシクロック信号として使用されることに注意してください。 の文を削除)
103	表 6.1-2 低消費電力モード動作状態 (1 系統品) を変更 (*4 *1) (*5 *2)
111 ~ 113	表 6.2-1 2 系統品の遷移条件一覧表 を変更 (発振安定時間待ち終了, 発振安定待ち終了, 発振安定終了 発振安定待ち時間終了)
114	表 6.2-2 1 系統品の状態遷移条件一覧表 を変更 (発振安定時間待ち終了, 発振安定待ち終了, 発振安定終了 発振安定待ち時間終了)

< 参考 > 4 版 5 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。)
124	表 7.1-1 モード端子とモード を変更 (テスト機能 の行を削除) 動作モード を変更 (動作モードを選択することで、通常動作を行ったり、内部テストプログラムや特別テスト機能をアクティブにすることができます。 動作モードを選択することで、通常動作を行ったり、内部テストプログラムができます。)
137	外部メモリアクセス制御信号 を変更 (EPCR ECSR)
138	図 7.3-2 外部データバス 16 ビットモードのアクセスタイミングチャート (16 ビットバス幅アクセス, 8 ビットバス幅アクセス時) を変更 (P36/RDY 端子 P 36/RDY 端子)
140	レディ機能 を変更 (EPCR 中の RYE ビットが "1" に設定されている場合, ECSR 中の RYE ビットが "1" に設定されている場合,)
141	7.3.2 ホールド機能の要約 を変更 (バス制御信号選択レジスタ (EPCR) 中の HDE ビットが "1" に バス制御信号選択レジスタ (ECSR) 中の HDE ビットが "1" に) ホールド機能 を変更 (・アドレス出力 :P23/A19 ~ P20/A16 ・アドレス出力 :P27/A23 ~ P20/A16) (・データ入出力 :P17/D15 ~ P00/D00 ・データ入出力 :P17/AD15 ~ P00/AD00)
141	図 7.3-4 ホールドタイミング を変更 (P35/HAK P35/HAK (上線を追加)) P27/A23 ~ P20/A16
145	図 8.2-1 I/O ポートのレジスタ一覧 を変更 ((DDR2) (PDR2))
146	図 8.2-2 ポートデータレジスタ (PDR0 ~ PDRA) (ポート 0 ~ ポート A 用) の PDR7 のアドレス を変更 (00000B _H 000007 _H)
154	図 9.1-1 タイムベースタイマのレジスタ一覧 の初期値 を変更 (初期値 (1) (-) (-) (0) (0) (0) (0) (0) (1) (-) (-) (0) (0) (1) (0) (0)) 図 9.1-2 タイムベースタイマのブロックダイアグラム を変更 (*を削除)
155	図 9.2-1 タイムベースタイマ制御レジスタ (TBTC) の初期値 を変更 (初期値 (1) (-) (-) (0) (0) (0) (0) (0) (1) (-) (-) (0) (0) (1) (0) (0))
172	図 12.2-1 16 ビット入出力タイマのレジスタ のアドレス を変更 (003920 _H 003918 _H) (003922 _H 00391A _H)
187	図 12.5-3 インプットキャプチャ制御ステータスレジスタ (ICS01) を変更 (ICS0/1 ICS01)
192	図 13.1-1 16 ビットリロードタイマのブロックダイアグラム を変更 (有効クロック判定回路 の記述を追加) (動作制御回路 の記述を追加)
193	16 ビットリロードタイマの端子名 を追加

< 参考 > 4 版 5 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。)
194	図 13.2-1 16 ビットリロードタイマのレジスタ を変更 (TMCSR TMCSR0/1) (TMR/TMRLR TMR0/TMRLR0 TMR1/TMRLR1)
195	図 13.2-2 タイマ制御ステータスレジスタ (TMCSR) を変更 (TMCSR TMCSR0/1)
198	図 13.2-3 16 ビットタイマレジスタ (TMR)/16 ビットタイマリロードレジスタ (TMRLR) を変更 (TMR/TMRLR TMR0/TMRLR0 TMR1/TMRLR1)
203	図 13.6-1 カウンタの状態遷移 を変更 (TOT 端子 : 入力禁止 TIN 端子 : 入力禁止) (TOT 端子 :TOUT 端子として機能 TOT 端子 :TOT 端子として機能)
206	8/16 ビット PPG の機能 を変更 (8+8 ビット PPG 出力動作モード 8 ビットプリスケアラ +8 ビット PPG 出力動作モード)
206	PPG 出力動作 を変更 (PPG の ch0 と ch1 を合わせて 1 ユニットとよびます。の文を追加) (4 本の PPG が PPG が 4 ユニット) (PPG0/1 PPG の ch0 と ch1) (ch0 の PPG の出力信号は、どの外部端子にも接続されていません。 PPG の ch0 を PPG (ch0), ch1 を PPG (ch1) と表します。)
207	図 14.2-1 8 ビット PPG(ch0) のブロックダイアグラム を変更 (一部の図および文 を削除) (チャンネル 1 PPG(ch1)) (ch0 の PPG 出力信号は外部端子に接続していません。の文を追加)
208	図 14.2-2 8 ビット PPG(ch1) のブロックダイアグラム を変更 (PPG10 出力許可 PPG0 端子出力許可) (PPG10 PPG0 端子) (チャンネル 0 PPG(ch0))
208	図 14.2-3 PPG モジュールとユニット番号 , および外部端子の関係 を変更
210	14.3.1 PPG0 動作モード制御レジスタ (PPGC0) の要約を変更 (PPG(ch0) を制御します。の文を追加) PPG0 動作モード制御レジスタ (PPGC0) を変更 (PPG00 PPG0) 図 14.3-2 PPG0 動作モード制御レジスタ (PPGC0) を変更 (リード / ライト (R/W) (-) (R/W) (R/W) (R/W) (-) (-) (R/W) (R/W) (-) (R/W) (R/W) (R/W) (-) (-) (W))
210, 211	PPG0 動作モード制御レジスタ (PPGC0) を変更 (PPG PPG(ch0))
212	14.3.2 PPG1 動作モード制御レジスタ (PPGC1) の要約を変更 (PPG(ch1) の制御と PPG ユニット 0 の動作モードを選択します。の文を追加)
212, 213	PPG1 動作モード制御レジスタ (PPGC1) を変更 (PPG PPG(ch1)) (PPG10 PPG0) (PPG タイマの動作モード PPG ユニットの動作モード)

< 参考 > 4 版 5 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。)
214	14.3.3 PPG ユニット 0 クロック選択レジスタ (PPG01) を変更 (PPG0/1 PPG01)
215	リロードレジスタ (PRL, PRLH) の表を変更 ("L" のパルス幅を設定します。の文を追加) ("H" のパルス幅を設定します。の文を追加)
216	表 14.4-1 リロード動作とパルス出力の関係を変更 (PPG00/10 PPG0 出力端子) 8/16 ビット PPG の動作 を変更 (端子出力 (PPG) PPG0 出力端子) 2 チャンネル独立モード を変更 (PPG00 端子は , ch0 の PPG 出力が接続され , PPG10 端子は , ch1 の PPG 出力が接続され ます。の文を削除) 8 ビットプリスケアラ +8 ビット PPG モード を変更 (PPG00 端子は , ch0 のプリスケアラ出力が接続され , PPG10 端子は , ch1 の PPG 出力が接 続されます。の文を削除) 16 ビット PPG1 チャンネルモード を変更 (PPG00 端子と PPG10 端子は , 両方とも 16 ビット PPG 出力が接続されます。の文を削除) (MB90540/545 シリーズについては , ch0 の PPG からの出力信号は外部端子には接続して いません。の文を削除)
217	図 14.4-1 PPG 出力動作の出力波形を変更 (PPG00/10 出力端子 PPG0 出力端子)
219	14.6 8/16 ビット PPG のパルスの端子出力の制御 の要約を変更 (外部端子 PPG00/PPG10 外部端子 PPG0) 8/16 ビット PPG のパルスの端子出力の制御 を変更 (本ビット PPG1 動作モード制御レジスタ (PPG1) の bit13 (PE10)) (PPG0 PPG(ch0)) (PPG10 PPG(ch1)) 図 14.6-1 8 ビットプリスケアラ +8 ビット PPG 出力動作の出力波形 を変更 (PPG00 PPG(ch0) の出力 (内部信号)) (PPG10 PPG(ch1) の出力 (PPG0 端子波形)) (PPG00 PPG(ch0)) (PPG10 PPG(ch1))
219	図 14.6-1 8 ビットプリスケアラ +8 ビット PPG 出力動作の出力波形を変更 (PPG00 PPG(ch0) の出力 (内部信号)) (PPG10 PPG(ch1) の出力 (PPG0 端子波形))
220	14.7 8/16 ビット PPG の割込み の要約を変更 (8/16 ビット PPG の割込みは , リロード値がカウントアウトし , ボローが発生したときに アクティブになります。 8/16 ビット PPG は , リロード値がカウントアウトし , ボロー が発生したときに割込み要求を出力します。)

< 参考 > 4 版 5 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。)
221	<p>レジスタ を変更 (0X000XX1_B 0-000--1_B) (0X000001_B 0-000001_B) (PPG10 XXXXXX00_B PPG01 000000--_B)</p> <p>パルス出力 を変更 (PPG0 端子は出力禁止となります。出力を許可した場合は "L" 出力となります。に文を変更)</p> <p>割込み要求 を変更 (割込み禁止となります。なお , リロード値は保持されます。に文を変更)</p>
230	<p>< 注意事項 > を変更 ((ENIR:EN7 ~ EN0=1) (ENIR:EN7 ~ EN0=1))</p>
235	<p>DTP/ 外部割込みの動作手順 を変更 (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。の文を追加)</p>
244	<p>< 注意事項 > を変更 (・ 設定チャンネルが ANS > ANE の場合は , ANS より変換が始まり 7ch まで変換したら 0ch に戻り ANE まで変換します。 設定チャンネルが ANS > ANE の場合は , ANS より変換が始まり AN7 まで変換したら AN0 に戻り ANE まで変換します。) (例 : チャンネル設定 ANS = AN6, ANE = AN3 で単発モードのとき変換は次の順序で行われます。) (6ch 7ch 0ch 1ch 2ch 3ch AN6 AN7 AN0 AN1 AN2 AN3)</p>
253	<p>表 17.5-1 単発モード時の EI²OS の起動例を変更 (転送元アドレスの文を追加)</p>
257	<p>変換データ保護機能の使用上の注意 を変更 (ADCS2 レジスタ ADCS1 レジスタ)</p>
275	<p>転送データフォーマット を変更 (SOUT0 SOT0)</p>
289	<p>< 注意事項 > を変更 (UMC1 レジスタ SCR1 レジスタ)</p>
293	<p>シリアルインプットデータレジスタ 1(SIDR1)/ シリアルアウトプットデータレジスタ 1 (SODR1) を変更 (DTRE TDRE)</p>
300	<p>非同期 (調歩同期) モードの受信動作 を変更 (SCR SCR1) (SSR SSR1) (SIDR SIDR1) (シリアルステータスレジスタ (SSR) 内の RDRF が シリアルステータスレジスタ (SSR1) 内の RDRF が)</p>
301	<p>非同期 (調歩同期) モードの送信動作 を変更 (SSR SSR1) (SODR SODR1) (SCR SCR1)</p>
302	<p>図 19.4-2 CLK 同期モードの転送データフォーマット (モード 2) を変更 (SODR 書込み SODR1 書込み) (SCLK SCK1)</p>

< 参考 > 4 版 5 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。)
306	図 19.6-4 TDRE のセットタイミング (モード 0, 1) を変更 (SODR 書込み SODR1 書込み)
307	図 19.6-5 TDRE のセットタイミング (モード 2) を変更 (SODR 書込み SODR1 書込み)
308	図 19.7-1 シリアルエッジセレクトレジスタ (SES1) を変更 (SES2 SES1) (リード / ライト (R/W) (R/W) (R/W) (R/W) (R/W) (R/W) (R/W) (R/W) (-) (-) (-) (-) (-) (-) (-) (R/W))
313	図 20.2-1 シリアル I/O のレジスタ を変更 (MI MD)
319	図 20.2-4 シリアル I/O プリスケラ (SCDCR) を変更 (MI MD)
341	< 注意事項 > を変更 (switch (10_CANCTO.CSE.bit.NS) switch (IO_CANCTO.CSE.bit.NS))
347	図 21.6-5 ビットタイミングレジスタ (BTR) の初期値を変更 ((-) (1) (1) (1) (1) (1) (0) (0) (-) (1) (1) (1) (1) (1) (1) (1))
348	ビットタイミングレジスタ (BTR) を変更 (TS1= TS1.3 ~ TS1.0, TS2 = TS2.2 ~ TS1.0, TS1= TS1.3 ~ TS1.0, TS2 = TS2.2 ~ TS2.0,)
351	図 21.6-10 送信要求レジスタ (TREQR) の初期値を変更 ((x) (x) (x) (x) (x) (x) (x) (x) (0) (0) (0) (0) (0) (0) (0) (0))
353	図 21.6-12 リモートフレーム受信待機レジスタ (RFWTR) の RFWTR アドレスを変更 (003DD1 _H (CAN1) 003D0D _H (CAN1)) 図 21.6-12 リモートフレーム受信待機レジスタ (RFWTR) の初期値を変更 ((0) (0) (0) (0) (0) (0) (0) (0) (X) (X) (X) (X) (X) (X) (X) (X))
389	プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1) を変更 (ADCSR PACSR)
390	【bit 3】AD1E(Address Detect register 1 Enable) を変更 (ADR1 PADR1)
403	表 24.3-1 フラッシュ制御信号 を変更 (RY/B RY/BY)
405	【bit 2, bit 0】LPM1, LPM0(Low Power Mode) を変更 (01: 低電力消費モード (内部動作周波数 8 MHz 以下で動作) 10: 低電力消費モード (内部動作周波数 8 MHz 以下で動作))
406	24.5 フラッシュメモリ自動アルゴリズム起動方法 の要約を変更 (読出し / リセット , 書込み , チップ消去 の 4 種類 読出し / リセット , 書込み , チップ消去 , セクタ消去 の 4 種類)
408	表 24.6-2 ハードウェアシーケンスフラグ機能一覧 を変更 (DQ5 と DQ3 の内容の入替え)
423	フラッシュメモリのセクタ消去の一時停止 を変更 (最大 15 μs 後 , 最大 20 μs 後 ,) (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 μs 以上後に行ってください。 の文を追加)

< 参考 > 4 版 5 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。)
425	ハードウェアリセット (RST) の入力を変更 (最低 50 ns 最低 500 ns) (消去中にハードウェアリセットすると、消去中のハードウェアリセットや電源切れにより、) ホールド機能 を変更 ((EPCR の HDE ビットが "1" をセット) (ECSR の HDE ビットが "1" をセット)) (コントロールステータスレジスタ (FMCS) フラッシュメモリ制御ステータスレジスタ (FMCS))
425	V _{ID} の適用 を変更 (セクタプロテクト動作に必要な V _{ID} の適用は、供給電源が ON のとき常に開始し、終了する必要があります。 セクタプロテクト動作に必要な V _{ID} の印加は、電源が ON のとき開始し、終了する必要があります。)
435	表 25.1-1 富士通標準シリアルオンボード書込みに使用する端子 を変更 (P00 に "L" レベルを、P01 に "H" レベルを入力してください。の文を追加)
437	図 25.2-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) 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) を変更 (TICS /TICS) (TRES /TRES)
439	図 25.3-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) シングルチップモード時シリアル書込み接続例 (ライタから電源供給時) を変更 (TICS /TICS) (TRES /TRES)
441	図 25.4-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) 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) を変更 (TRES /TRES)
443	図 25.5-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) フラッシュマイコンプログラマとの最小限の接続例 (ライタから電源供給時) を変更 (TRES /TRES)
446 ~ 450	付表 A-1 I/O マップ の以下のレジスタ について、レジスタを変更 (PDR0 ~ PDRA, DDR0 ~ DDRA, ADER, PUCR0 ~ PUCR3, UMC0, URD0, SCR1, SSR1, CDCR, SMCS, SDR, ENIR, EIRR, ADCS0, ADCS1, ICS01, ICS23, ICS45, ICS67, TMCSR0, TMR0/TMRLR0, TMCSR1, TMR1/TMRLR1, OCS0 ~ OCS3, TCCS, PACSR, FMCS, IPCP0 ~ IPCP7)
447, 450	付表 A-1 I/O マップ の以下のレジスタ について、リソースを変更 (PPGC0, PPGC1, PPG01, PPGC2, PPGC3, PPG23, PPGC4, PPGC5, PPG45, PPGC6, PPGC7, PPG67, PRL0 ~ PRL7, PRLH0 ~ PRLH7)
446, 447	付表 A-1 I/O マップ の以下のレジスタ について、アクセス単位を変更 (UMC0, USR0, SCR1, SSR1, ADCR1, PPGC0 ~ PPGC7,)
448	付表 A-1 I/O マップ の以下のレジスタについて、アクセス単位を変更 (ROMM)
449	付表 A-1 I/O マップ の以下のレジスタについて、初期値を変更 (ICR00 ~ ICR15)
450	付表 A-1 I/O マップ の以下のレジスタについて、略称を変更 (OCCP0 OCCP0)

< 参考 > 4 版 5 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。）
459	図 B.3-7 直接アドレス (addr16) 例 を変更
489	表 B.8-17 アキュムレータ操作命令 (バイト・ワード) 6 命令 を変更 (SWAPW / XCHW A, T SWAPW)

目次

第1章	概要	1
1.1	製品概要	2
1.2	特長	3
1.3	ブロックダイヤグラム	5
1.4	パッケージ外形寸法図	6
1.5	端子配列図	8
1.6	端子機能説明	10
1.7	入出力回路形式	17
1.8	デバイスの取扱いに関する注意事項	20
第2章	CPU	25
2.1	CPU の概要	26
2.2	メモリ空間	27
2.3	メモリ空間マップ	28
2.4	リニア方式によるアドレス指定	29
2.5	バンク方式によるアドレス指定	30
2.6	メモリ空間における多バイト長データ	32
2.7	レジスタ	33
2.7.1	アキュムレータ (A)	35
2.7.2	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)	36
2.7.3	プロセッサステータス (PS)	37
2.7.4	プログラムカウンタ (PC)	40
2.8	レジスタバンク	41
2.9	プリフィックスコード	43
2.10	割込み抑止命令	45
2.11	「DIV A,Ri」, 「DIVW A,RWi」命令の使用上の注意	47
第3章	割込み	49
3.1	割込みの概要	50
3.2	割込み要因	51
3.3	割込みベクタ	53
3.4	ハードウェア割込み	55
3.4.1	ハードウェア割込みの動作	57
3.4.2	ハードウェア割込みの動作フロー	59
3.4.3	割込み処理の開始に要する時間	60
3.5	ソフトウェア割込み	62
3.6	拡張インテリジェント I/O サービス (EI ² OS)	64
3.6.1	割込み制御レジスタ (ICR)	66
3.6.2	拡張インテリジェント I/O サービスディスクリプタ (ISD)	69
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	72
3.6.4	拡張インテリジェント I/O サービス (EI ² OS) の実行時間	74
3.7	未定義命令の実行による例外発生	75

第4章	クロックとリセット	77
4.1	クロック発生部	78
4.2	リセット要因の発生	79
4.3	リセット要因	83
第5章	低消費電力制御回路	85
5.1	低消費電力制御回路の概要	86
5.2	低消費電力制御回路のブロックダイアグラム	89
5.3	低消費電力制御回路のレジスタ	90
5.3.1	低消費電力モード制御レジスタ (LPMCR)	91
5.3.2	クロック選択レジスタ (CKSCR)	93
5.4	クロック選択の状態遷移	96
第6章	低消費電力モード	99
6.1	低消費電力モード	100
6.1.1	スリープモード	104
6.1.2	擬似時計モード	105
6.1.3	時計モード	106
6.1.4	ストップモード	107
6.1.5	ハードウェアスタンバイモード	109
6.1.6	CPU 間欠動作	110
6.2	低消費電力モードの状態遷移	111
6.3	低消費電力モード状態遷移図	115
第7章	メモリアクセスモード	123
7.1	メモリアクセスモードの概要	124
7.1.1	モード端子	125
7.1.2	モードデータ	126
7.1.3	バスモード別メモリ空間	127
7.2	外部メモリアクセス (バス端子制御回路)	129
7.2.1	外部メモリアクセス (外部バス端子制御回路) のレジスタ	130
7.2.2	自動レディ機能選択レジスタ (ARSR)	131
7.2.3	外部アドレス出力制御レジスタ (HACR)	133
7.2.4	バス制御信号選択レジスタ (ECSR)	134
7.3	外部メモリアクセス制御信号の動作	137
7.3.1	レディ機能	139
7.3.2	ホールド機能	141
第8章	I/O ポート	143
8.1	I/O ポート	144
8.2	I/O ポートのレジスタ一覧	145
8.2.1	ポートデータレジスタ (PDR0 ~ PDRA) (ポート0 ~ ポートA用)	146
8.2.2	ポート方向レジスタ (DDR0 ~ DDRA) (ポート0 ~ ポートA用)	148
8.2.3	プルアップ制御レジスタ (PUCR0 ~ PUCR3) (ポート0 ~ ポート3用)	150
8.2.4	アナログ入力許可レジスタ (ADER)	152
第9章	タイムベースタイマ	153
9.1	タイムベースタイマの概要	154
9.2	タイムベースタイマ制御レジスタ (TBTC)	155

9.3	タイムベースタイマの動作.....	156
第 10 章	ウォッチドッグタイマ	157
10.1	ウォッチドッグタイマの概要.....	158
10.2	ウォッチドッグタイマ制御レジスタ (WDTC).....	159
10.3	ウォッチドッグタイマの動作.....	161
第 11 章	時計タイマ	163
11.1	時計タイマの概要.....	164
11.2	時計タイマ制御レジスタ (WTC).....	165
11.3	時計タイマの動作.....	167
第 12 章	16 ビット入出力タイマ	169
12.1	16 ビット入出力タイマの概要.....	170
12.2	16 ビット入出力タイマのレジスタ	172
12.3	16 ビットフリーランタイマ	173
12.3.1	16 ビットフリーランタイマのレジスタ	174
12.3.2	タイマカウンタ制御ステータスレジスタ (TCCS).....	175
12.3.3	16 ビットフリーランタイマの動作	177
12.4	アウトプットコンペア	179
12.4.1	アウトプットコンペアのレジスタ.....	180
12.4.2	アウトプットコンペアの制御ステータスレジスタ (OCS0/1)	181
12.4.3	16 ビットアウトプットコンペアの動作	184
12.5	インプットキャプチャ.....	186
12.5.1	インプットキャプチャのレジスタ.....	187
12.5.2	16 ビットインプットキャプチャの動作	189
第 13 章	16 ビットリロードタイマ (イベントカウンタ機能付き)	191
13.1	16 ビットリロードタイマ (イベントカウンタ機能付き) の概要	192
13.2	16 ビットリロードタイマのレジスタ.....	194
13.2.1	タイマ制御ステータスレジスタ (TMCSR).....	195
13.2.2	16 ビットタイマレジスタ (TMR)/16 ビットタイマリロードレジスタ (TMRLR) の レジスタ配置1.....	98
13.3	16 ビットリロードタイマの内部クロック動作と外部クロック動作	199
13.4	16 ビットリロードタイマのアンダフロー動作	201
13.5	16 ビットリロードタイマの出力端子機能.....	202
13.6	カウンタの動作状態	203
第 14 章	8/16 ビット PPG	205
14.1	8/16 ビット PPG の概要.....	206
14.2	8/16 ビット PPG のブロックダイアグラム	207
14.3	8/16 ビット PPG のレジスタ.....	209
14.3.1	PPG0 動作モード制御レジスタ (PPGC0)	210
14.3.2	PPG1 動作モード制御レジスタ (PPGC1)	212
14.3.3	PPG ユニット 0 クロック選択レジスタ (PPG01)	214
14.3.4	リロードレジスタ (PRL, PRLH).....	215
14.4	8/16 ビット PPG の動作	216
14.5	8/16 ビット PPG のカウントクロックの選択	218
14.6	8/16 ビット PPG のパルスの端子出力の制御	219

14.7	8/16 ビット PPG の割込み	220
14.8	8/16 ビット PPG の各ハードウェアの初期値	221
第 15 章	遅延割込み	223
15.1	遅延割込みモジュールの概要	224
15.2	遅延割込みのレジスタ	225
15.3	遅延割込みの動作	226
第 16 章	DTP/ 外部割込み	227
16.1	DTP/ 外部割込みの概要	228
16.2	DTP/ 外部割込みのレジスタ	230
16.3	DTP/ 外部割込みの動作	232
16.4	DTP 要求と外部割込み要求の切換え	234
16.5	DTP/ 外部割込みの使用上の注意	235
第 17 章	A/D コンバータ	237
17.1	A/D コンバータの特長	238
17.2	A/D コンバータのブロックダイアグラム	240
17.3	A/D コンバータのレジスタ一覧	241
17.3.1	A/D 制御ステータスレジスタ 0 (ADCS0)	242
17.3.2	A/D 制御ステータスレジスタ 1 (ADCS1)	245
17.3.3	A/D データレジスタ 0/1 (ADCR0, ADCR1)	248
17.4	A/D コンバータの動作	250
17.5	EI ² OS を使った変換動作	252
17.5.1	単発モード時の EI ² OS の起動例	253
17.5.2	連続モード時の EI ² OS の起動例	254
17.5.3	停止モード時の EI ² OS の起動例	255
17.6	変換データ保護機能	256
第 18 章	UART0	259
18.1	UART0 の特長	260
18.2	UART0 のブロックダイアグラム	261
18.3	UART0 のレジスタ	262
18.3.1	シリアルモード制御レジスタ 0(UMC0)	263
18.3.2	シリアルステータスレジスタ 0(USR0)	265
18.3.3	シリアルインプットデータレジスタ 0(UIDR0) と シリアルアウトプットデータレジスタ 0(UODR0)	267
18.3.4	レート・データレジスタ 0(URD0)	268
18.4	UART0 の動作	270
18.5	ボーレート	271
18.6	内部クロックおよび外部クロック	274
18.7	転送データフォーマット	275
18.8	パリティビット	276
18.9	割込み生成とフラグのセットタイミング	277
18.9.1	受信動作時 (モード 0, モード 1, モード 3) のフラグのセットタイミング	278
18.9.2	受信動作時 (モード 2) のフラグのセットタイミング	279
18.9.3	送信動作時のフラグのセットタイミング	280
18.9.4	送受信動作時のステータスフラグ	281
18.10	UART0 の応用例	282

第 19 章	UART1(SCI)	285
19.1	UART1 の特長	286
19.2	UART1 のブロックダイアグラム	287
19.3	UART1 のレジスタ一覧	288
19.3.1	シリアルモードレジスタ 1(SMR1)	289
19.3.2	シリアル制御レジスタ 1 (SCR1)	291
19.3.3	シリアルインプットデータレジスタ 1(SIDR1)/ シリアルアウトプットデータレジスタ 1(SODR1)	293
19.3.4	シリアルステータスレジスタ 1(SSR1)	294
19.3.5	UART1 通信プリスケアラ制御レジスタ (CDCR)	296
19.4	UART1 の動作モードとクロック選択	297
19.4.1	非同期 (調歩同期) モード	300
19.4.2	CLK 同期モード	302
19.5	UART1 のフラグと割込み発生要因	304
19.6	UART1 の割込みとフラグのセットタイミング	305
19.7	ネガティブクロック動作	308
19.8	UART1 の応用例と使用上の注意	309
第 20 章	シリアル I/O	311
20.1	シリアル I/O の概要	312
20.2	シリアル I/O のレジスタ	313
20.2.1	シリアルモード制御ステータスレジスタ (SMCS)	314
20.2.2	シリアルシフトデータレジスタ (SDR)	318
20.2.3	シリアル I/O プリスケアラ (SCDCR)	319
20.3	シリアル I/O の動作	320
20.3.1	シフトクロック	321
20.3.2	シリアル I/O の動作状態	322
20.3.3	シフト動作のスタート / ストップタイミング	324
20.3.4	シリアル I/O の割込み機能	327
20.4	ネガティブクロック動作	328
第 21 章	CAN コントローラ	329
21.1	CAN コントローラの特長	330
21.2	CAN コントローラのブロックダイアグラム	331
21.3	全体制御レジスタの一覧	332
21.4	メッセージバッファ (ID レジスタ) の一覧	334
21.5	メッセージバッファ (DLC レジスタ / データレジスタ) の一覧	336
21.6	CAN コントローラのレジスタ分類	338
21.6.1	制御ステータスレジスタ (CSR)	339
21.6.2	バス動作停止ビット (HALT=1)	342
21.6.3	ラストイベント表示レジスタ (LEIR)	344
21.6.4	受信および送信エラーカウンタ (RTEC)	346
21.6.5	ビットタイミングレジスタ (BTR)	347
21.6.6	メッセージバッファ有効レジスタ (BVALR)	349
21.6.7	IDE レジスタ (IDER)	350
21.6.8	送信要求レジスタ (TREQR)	351
21.6.9	送信 RTR レジスタ (TRTRR)	352
21.6.10	リモートフレーム受信待機レジスタ (RFWTR)	353
21.6.11	送信解除レジスタ (TCANR)	354

21.6.12 送信完了レジスタ (TCR)	355
21.6.13 送信割込み許可レジスタ (TIER).....	356
21.6.14 受信完了レジスタ (RCR).....	357
21.6.15 リモート要求受信レジスタ (RRTRR).....	358
21.6.16 受信オーバランレジスタ (ROVRR).....	359
21.6.17 受信割込み許可レジスタ (RIER)	360
21.6.18 アクセプタンスマスク選択レジスタ (AMSR)	361
21.6.19 アクセプタンスマスクレジスタ 0/1(AMR0/AMR1)	363
21.6.20 メッセージバッファ	365
21.6.21 ID レジスタ x(x=0 ~ 15)(IDRx).....	366
21.6.22 DLC レジスタ x(x=0 ~ 15)(DLCRx).....	368
21.6.23 データレジスタ x(x=0 ~ 15)(DTRx)	369
21.7 CAN コントローラの送信	371
21.8 CAN コントローラの受信	373
21.9 CAN コントローラの受信フローチャート	376
21.10 CAN コントローラの使用法	377
21.11 メッセージバッファ (x) による送信方法	379
21.12 メッセージバッファ (x) による受信方法	381
21.13 マルチレベルメッセージバッファの構成の決定	383
21.14 CAN コントローラの使用上の注意	385
第 22 章 アドレス一致検出機能	387
22.1 アドレス一致検出機能の概要	388
22.2 アドレス一致検出機能のレジスタ	389
22.3 アドレス一致検出機能の動作	391
22.4 アドレス一致検出機能の使用例	392
第 23 章 ROM ミラー機能選択モジュール	395
23.1 ROM ミラー機能選択モジュールの概要	396
23.2 ROM ミラー機能選択レジスタ (ROMM)	397
第 24 章 1M/2M ビットフラッシュメモリ	399
24.1 1M/2M ビットフラッシュメモリの概要	400
24.2 フラッシュメモリのセクタ構成	401
24.3 書込み / 消去モード	402
24.4 フラッシュメモリ制御ステータスレジスタ (FMCS)	404
24.5 フラッシュメモリ自動アルゴリズム起動方法	406
24.6 自動アルゴリズム実行状態の確認	407
24.6.1 データボーリングフラグ (DQ7)	409
24.6.2 トグルビットフラグ (DQ6).....	411
24.6.3 タイミングリミット超過フラグ (DQ5).....	412
24.6.4 セクタ消去タイマフラグ (DQ3)	413
24.6.5 トグルビット 2 フラグ (DQ2).....	414
24.7 フラッシュメモリ書込み / 消去の詳細説明	416
24.7.1 フラッシュメモリの読出し / リセット状態	417
24.7.2 フラッシュメモリへのデータ書込み	418
24.7.3 フラッシュメモリの全データの消去 (チップ消去)	420
24.7.4 フラッシュメモリの任意データの消去 (セクタ消去)	421
24.7.5 フラッシュメモリのセクタ消去の一時停止	423

24.7.6	フラッシュメモリのセクタ消去の再開	424
24.8	1M/2M ビットフラッシュメモリ使用上の注意.....	425
24.9	フラッシュセキュリティの特長	426
24.10	1M/2M ビットフラッシュメモリのプログラム例	427
第 25 章	MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S)	
	シリアル書込み接続例.....	433
25.1	MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) シリアル書込み接続の基本構成	434
25.2	シリアル書込み接続例 (ユーザ電源使用時).....	437
25.3	シリアル書込み接続例 (ライタから電源供給時).....	439
25.4	フラッシュマイコンプログラマとの最小限の接続例 (ユーザ電源使用時).....	441
25.5	フラッシュマイコンプログラマとの最小限の接続例 (ライタから電源供給時).....	443
付録	445
付録 A	I/O マップ	446
付録 B	命令	452
索引	513

第1章

概要

MB90540/MB90545 シリーズの特長と基本的な仕様について説明します。

- 1.1 製品概要
- 1.2 特長
- 1.3 ブロックダイヤグラム
- 1.4 パッケージ外形寸法図
- 1.5 端子配列図
- 1.6 端子機能説明
- 1.7 入出力回路形式
- 1.8 デバイスの取扱いに関する注意事項

1.1 製品概要

MB90540/545 シリーズの製品概要を示します。

■ MB90540/545 シリーズの製品概要

表 1.1-1 製品概要

特長	MB90V540/V540G	MB90F543/F549/F543G(S)/F548G(S) / F549G(S)/F546G(S)/F548GL(S)	MB90543G(S)/547G(S) / 548G(S)/549G(S)
CPU	F ² MC-16LX CPU		
システム クロック	オンチップ PLL クロック通倍方式 (× 1, × 2, × 3, × 4, PLL 停止時 1/2) 最小命令実行時間 62.5ns(4MHz 原発振 4 通倍)		
ROM 容量	外部	フラッシュメモリ MB90F543/F543G(S)/F548G(S)/F548GL(S): 128K バイト MB90F549/F549G(S)/F546G(S):256K バイト	MASK ROM MB90547G(S):64K バイト MB90543G(S)/548G(S): 128K バイト MB90549G(S):256K バイト
RAM 容量	8K バイト	MB90F548G(S)/F548GL(S):4K バイト MB90F543/F549/F543G(S)/F549G(S): 6K バイト MB90F546G(S):8K バイト	MB90547G(S):2K バイト MB90548G(S):4K バイト MB90543G(S)/549G(S): 6K バイト
クロック	2 系統品 *1	MB90F543/F549/F543G/F548G/F549G/F546G/ F548GL:2 系統品 MB90F543GS/F548GS/F549GS/F546GS/ F548GLS:1 系統品	MB90543G/547G/548G/549G: 2 系統品 MB90543GS/547GS/548GS/ 549GS :1 系統品
パッケージ	PGA-256	QFP100/LQFP100	
エミュレータ 専用電源 *2	なし	-	

*1: 1 系統で使用する場合は、ツール側より X0A, X1A にクロックを供給してください。

*2: エミュレーションポッド MB2145-507 を使用する際のディップスイッチ S2 の設定です。

詳細については MB2145-507 ハードウェアマニュアル (「2.7 エミュレータ専用電源」) を参照してください。

< 注意事項 >

型格に G サフィックスの付いた製品については、CAN コントローラの機能が拡張されています。

「第 21 章 CAN コントローラ」のビットタイミングレジスタを参照してください。

1.2 特長

表 1.2-1 に , MB90540/545 シリーズの特長を示します。

■ 特長

表 1.2-1 MB90540/545 シリーズの特長 (1 / 2)

機能	特長
UART0	全二重ダブルバッファ方式 非同期 / 同期転送 (スタート / ストップビット付き) をサポートします。 ボーレート 4808/5208/9615/10417/19230/38460/62500/500000bps(非同期) システムクロック =16MHz 時 500K/1K/2Mbps(同期)
UART1 (SCI)	全二重ダブルバッファ方式 非同期 (スタート / ストップ同期) および CLK 同期通信 ボーレート 1202/2404/4808/9615/19230/31250/38460/62500bps(非同期) 6, 8, 10, 12, 16MHz 時 62.5K/125K/250K/500K/1Mbps(同期)
シリアル I/O	MBS または LSB から転送開始可能です。 内部クロック同期転送および外部クロック同期転送をサポートします。 ポジティブエッジとネガティブエッジクロック同期をサポートします。 ボーレート システムクロック =16MHz 時 31.25K/62.5K/125K/500K/1M/2Mbps
A/D コンバータ	10 ビットまたは 8 ビットの分解能 8 本の入力チャネル 変換時間 :26.3 μ s(1 チャネルあたり)
16 ビット リロードタイマ (2 チャネル)	動作クロック周波数 $f_{sys}/2^1$, $f_{sys}/2^3$, $f_{sys}/2^5$ (f_{sys} = システムクロック周波数) 外部イベントカウント機能をサポートします。
16 ビット 入出力タイマ	オーバフロー時に割込み信号を出力します。 アウトプットコンペア (ch0) との照合時に , タイマクリアをサポートします。 動作クロック周波数 $f_{sys}/2^2$, $f_{sys}/2^4$, $f_{sys}/2^6$, $f_{sys}/2^8$ (f_{sys} = システムクロック周波数)
16 ビット アウトプットコンペア (4 チャネル)	16 ビット入出力タイマとの照合時に , 割込み信号を出力します。 16 ビットキャプチャレジスタ \times 4 本 一対のコンペアレジスタを出力信号生成に使用可能です。
16 ビット インプットキャプチャ (8 チャネル)	立上りエッジ , 立下りエッジまたはその両方を検出します。 16 ビットキャプチャレジスタ \times 4 本 外部イベント時に割込み信号を出力します。
8/16 ビット PPG (プログラマブルパルス発 生部)(4 チャネル)	8 ビットと 16 ビットの動作モードをサポートします。 8 ビットリロードカウンタ \times 2 本 \times 4 ユニット "L" パルス幅用の 8 ビットリロードレジスタ \times 2 本 \times 4 ユニット "H" パルス幅用の 8 ビットリロードレジスタ \times 2 本 \times 4 ユニット 一対の 8 ビットリロードカウンタを 1 本の 16 ビットリロードカウンタとして , ま たは 8 ビットプリスケアラと 8 ビットリロードカウンタとして構成可能です。 出力端子 \times 4 本 動作クロック周波数 f_{sys} , $f_{sys}/2^1$, $f_{sys}/2^2$, $f_{sys}/2^3$, $f_{sys}/2^4$ または 128 μ s f_{osc} =4MHz(f_{sys} = システムクロック周波数 , f_{osc} = 発振クロック周波数)

表 1.2-1 MB90540/545 シリーズの特長 (2 / 2)

機能	特長
CAN インタフェース MB90540 シリーズ : 2 チャンネル MB90545 シリーズ : 1 チャンネル	CAN 仕様バージョン 2.0 パート A およびパート B に準拠します。 エラー時に自動再送します。 リモートフレームに応答して自動伝送します。 データおよび ID 用の順位付けされた 16 個のメッセージバッファ複数メッセージをサポートします。 アクセプタンスフィルタの柔軟な構成 : 全ビットコンペア / 全ビットマスク / 2 個の部分ビットマスク 最高 1Mbps までサポートします。
外部割込み	エッジ検出とレベル検出が設定可能です。
外部バス インタフェース	選択可能な 8 ビットバスまたは 16 ビットバスを使用した外部アクセスが可能です (外部バスモード)。
入出力ポート	汎用入出力ポート 81 本 すべてのプッシュプル出力とシュミットトリガ入力 入力 / 出力または周辺信号として、ビット単位でプログラム可能です。
32kHz サブクロック	低電力動作のサブクロック。
フラッシュメモリ	自動プログラミング, Embedded Algorithm TM *, 書込み / 消去 / 消去一時停止 / 消去再開コマンドをサポートします。 アルゴリズムの完了を示すフラグ 消去サイクル数 :10,000 回 データ保持期間 :10 年間 ブートブロック構成 各ブロックで消去を実行可能です。 外部プログラミング電圧によるブロック保護

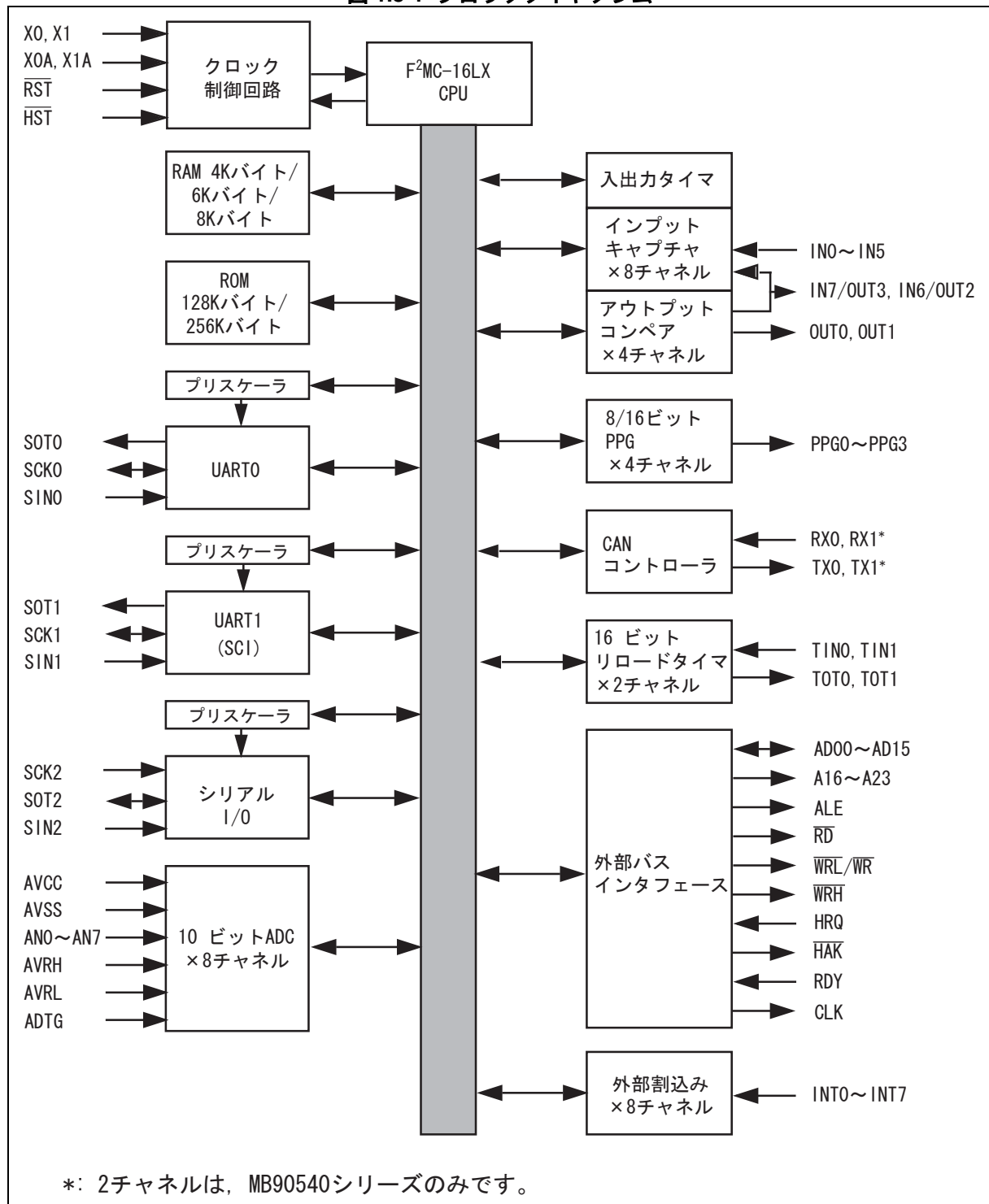
*:Embedded Algorithm は Advanced Micro devices Inc. の登録商標です。

1.3 ブロックダイアグラム

図 1.3-1 に , MB90540/545 シリーズのブロックダイアグラムを示します。

■ ブロックダイアグラム

図 1.3-1 ブロックダイアグラム



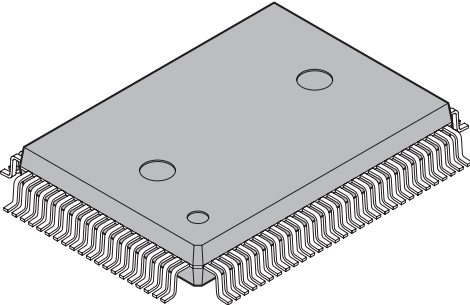
1.4 パッケージ外形寸法図

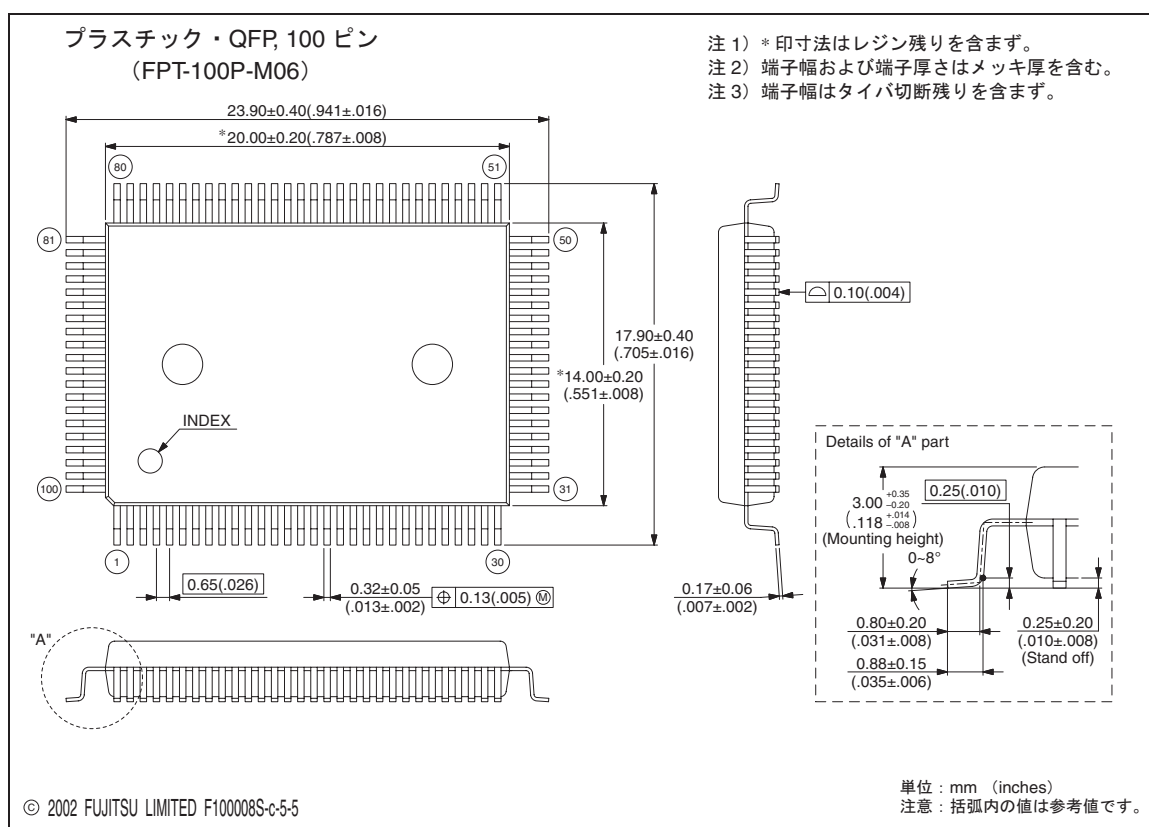
図 1.4-1 に FPT-100P-M06(QFP-100) のパッケージ外形寸法図を，図 1.4-2 に FPT-100P-M05(LQFP-100) のパッケージ外形寸法図を示します。

なお，本外形寸法図は参考用です。正式版につきましては別途ご相談ください。

■ FPT-100P-M06 パッケージ外形寸法図

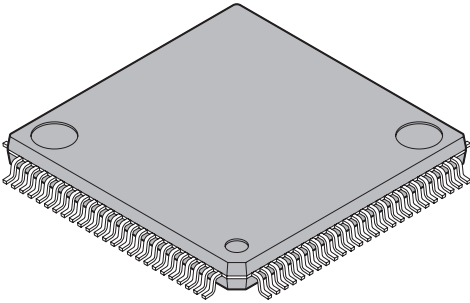
図 1.4-1 FPT-100P-M06(QFP-100) パッケージ外形寸法図

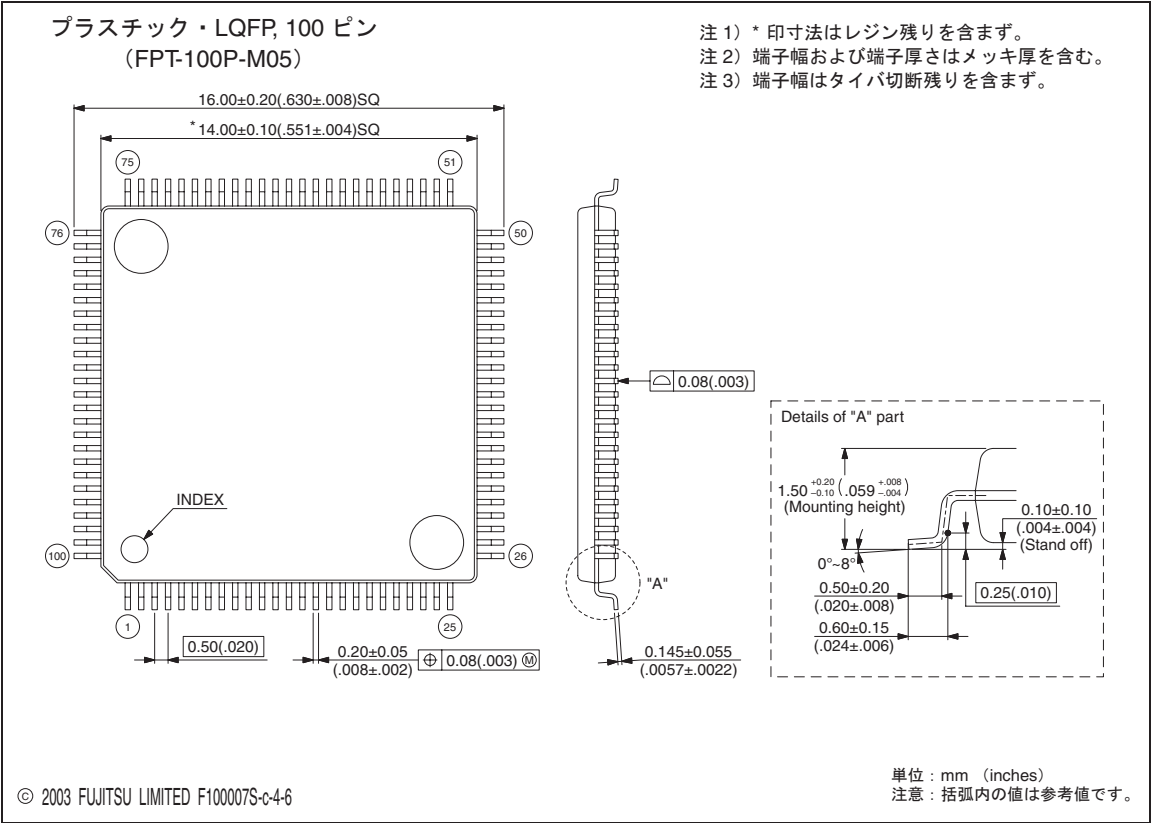
<p>プラスチック・QFP, 100 ピン</p>  <p>(FPT-100P-M06)</p>	リードピッチ	0.65mm
	パッケージ幅× パッケージ長さ	14.00 × 20.00mm
	リード形状	ガルウィング
	封止方法	プラスチックモールド
	取付け高さ	3.35mm MAX
	コード (参考)	P-QFP100-14×20-0.65



■ FPT-100P-M05 パッケージ外形寸法図

図 1.4-2 FPT-100P-M05(LQFP-100) パッケージ外形寸法図

<p>プラスチック・LQFP, 100 ピン</p>  <p>(FPT-100P-M05)</p>	リードピッチ	0.50mm
	パッケージ幅× パッケージ長さ	14.0 × 14.0mm
	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	0.65g
	コード (参考)	P-LFQFP100-14×14-0.50

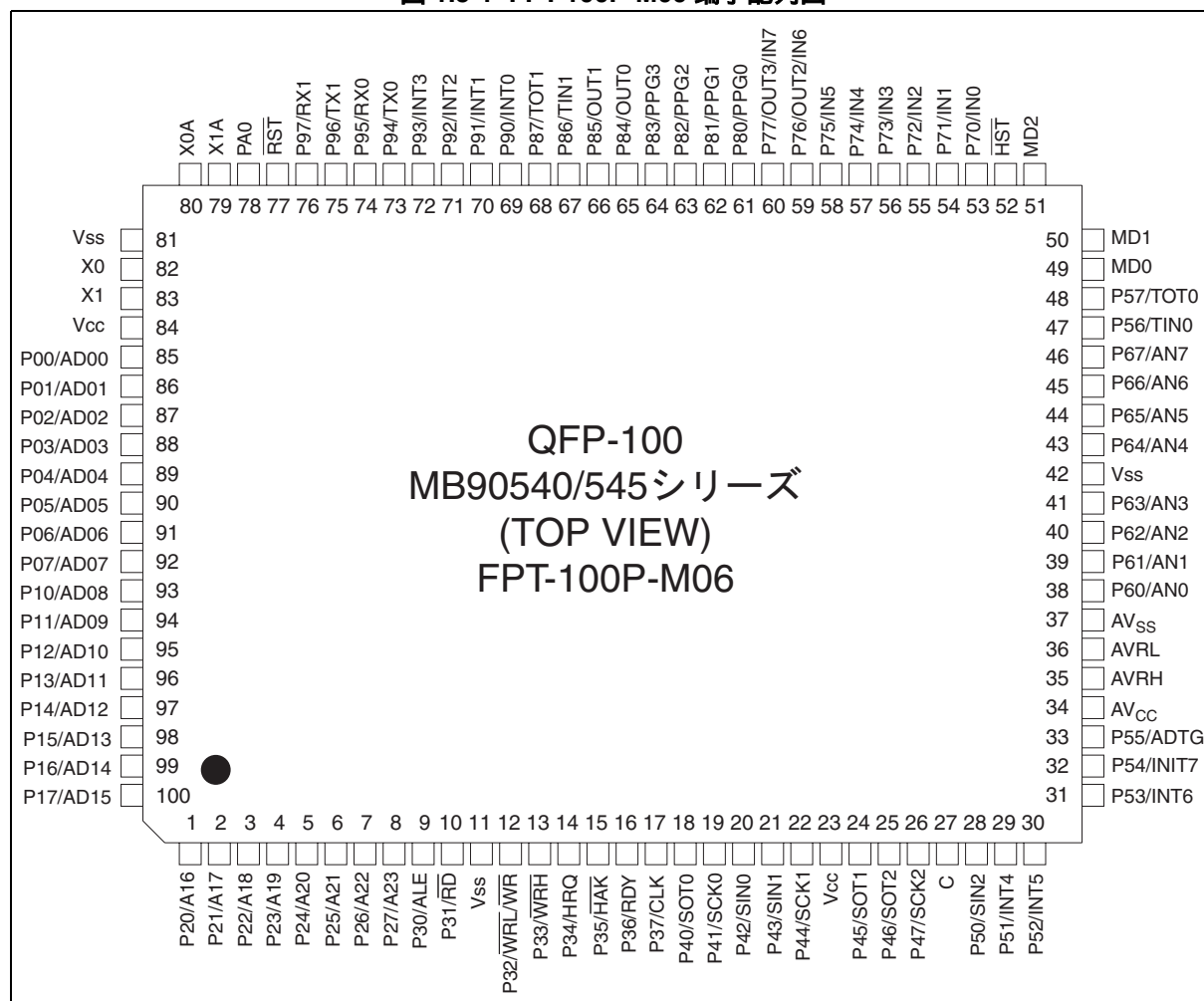


1.5 端子配列図

図 1.5-1 に FPT-100P-M06 の端子配列図を，図 1.5-2 に FPT-100P-M05 の端子配列図を示します。

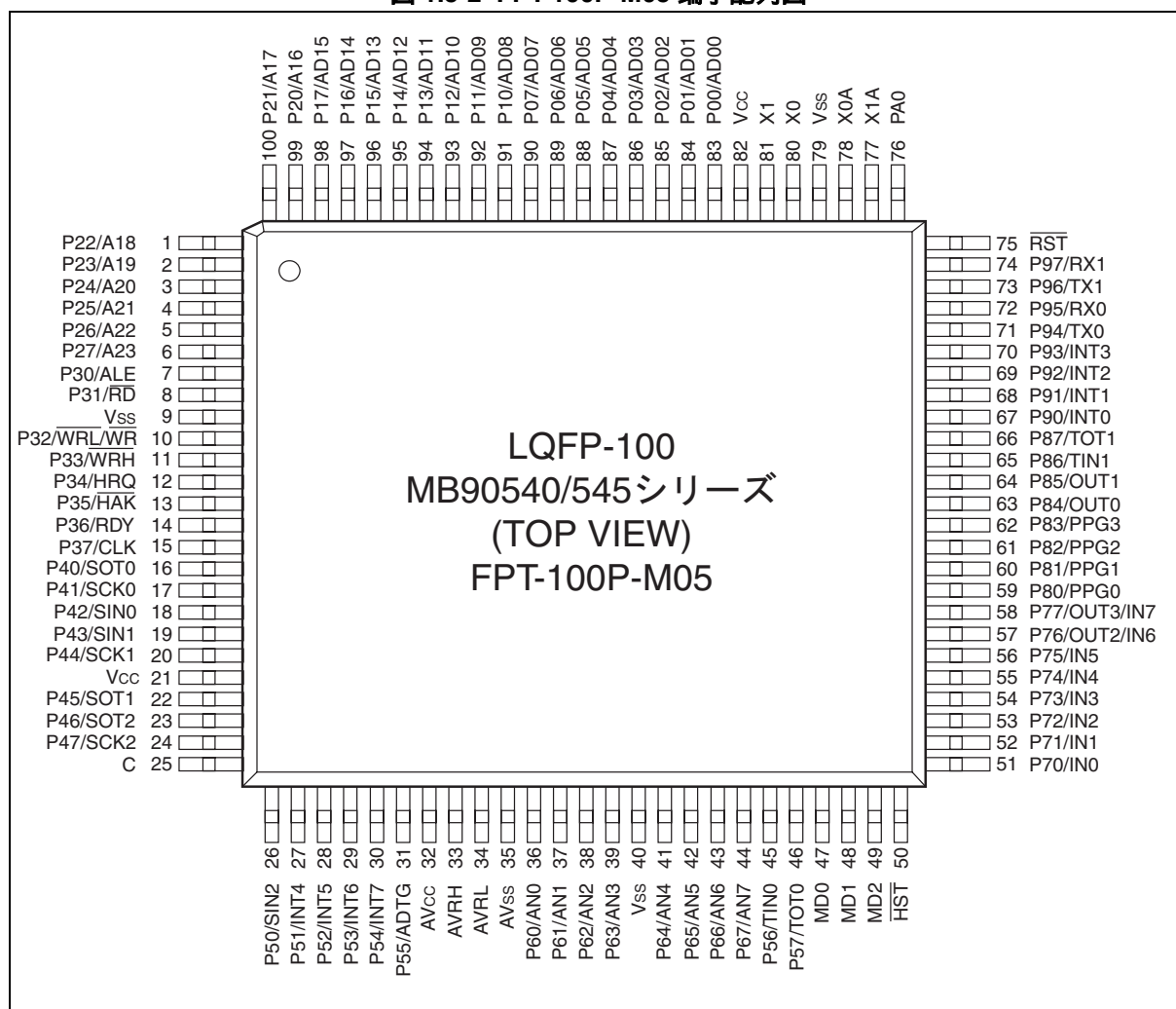
■ FPT-100P-M06 端子配列図

図 1.5-1 FPT-100P-M06 端子配列図



■ FPT-100P-M05 端子配列図

図 1.5-2 FPT-100P-M05 端子配列図



1.6 端子機能説明

表 1.6-1 に、端子名、回路形式、機能説明について示します。

■ 端子機能説明

表 1.6-1 端子機能説明 (1 / 7)

端子番号		端子名	入出力 回路 形式 *3	機能
LQFP*2	QFP*1			
80 81	82 83	X0 X1	A (発振)	高速水晶発振用端子です。
78	80	X0A	A (発振)	低速水晶発振用端子です。 1 系統品は外部でプルダウンの処理をしてください。
77	79	X1A		低速水晶発振用端子です。 1 系統品はオープンにしてください。
75	77	$\overline{\text{RST}}$	B	外部リセット要求入力端子です。
50	52	$\overline{\text{HST}}$	C	ハードウェアスタンバイ入力端子です。
83 ~ 90	85 ~ 92	P00 ~ P07	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードのときに有効となります。
		AD00 ~ AD07		外部アドレス・データバス下位 8 ビットの入出力端子です。 この機能は外部バスが有効なモードのときに有効となります。
91 ~ 98	93 ~ 100	P10 ~ P17	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードのときに有効となります。
		AD08 ~ AD15		外部アドレス・データバス上位 8 ビットの入出力端子です。 この機能は外部バスが有効なモードのときに有効となります。
99 ~ 6	1 ~ 8	P20 ~ P27	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 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" の場合に有効となります。
		A16 ~ A23		外部アドレス・データバス A16 ~ A23 の 8 ビットの出力端子です。 外バスモード時に、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" の場合に有効となります。

*1：FPT-100P-M06

*2：FPT-100P-M05

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

表 1.6-1 端子機能説明 (2 / 7)

端子番号		端子名	入出力回路形式 ^{*3}	機能
LQFP ^{*2}	QFP ^{*1}			
7	9	P30	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードのときに有効となります。
		ALE		アドレスラッチイネーブル出力端子です。 この機能は外部バスが有効なモードのときに有効となります。
8	10	P31	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードのときに有効となります。
		RD		データバスに対するリードストローブ出力端子です。 この機能は外部バスが有効なモードのときに有効となります。
10	12	P32	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードあるいは $\overline{\text{WR}}/\overline{\text{WRL}}$ 端子出力が禁止のときに有効となります。
		$\overline{\text{WRL}}$		データバスのライトストローブ出力端子です。 この機能は外部バスが有効なモードかつ $\overline{\text{WR}}/\overline{\text{WRL}}$ 端子出力が許可のときに有効となります。 $\overline{\text{WRL}}$ は、外バスモード・16 ビットバス時のデータバス下位 8 ビットに対するライトストローブ出力端子です。 $\overline{\text{WR}}$ は、外バスモード・8 ビットバス時のデータバス 8 ビットに対するライトストローブ出力端子です。
		$\overline{\text{WR}}$		
11	13	P33	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードあるいは外部バス 8 ビットモードあるいは $\overline{\text{WRH}}$ 端子出力が禁止のときに有効となります。
		$\overline{\text{WRH}}$		データバス上位 8 ビットに対するライトストローブ出力端子です。 この機能は外部バスが有効なモードかつ外部バス 16 ビットモードで、 $\overline{\text{WRH}}$ 出力端子が許可のときに有効となります。
12	14	P34	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードとホールド機能が禁止のときに有効となります。
		HRQ		ホールドリクエスト入力端子です。 この機能は外部バスが有効なモードでホールド機能が許可のときに有効となります。

*1 : FPT-100P-M06

*2 : FPT-100P-M05

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

表 1.6-1 端子機能説明 (3 / 7)

端子番号		端子名	入出力回路形式 *3	機能
LQFP *2	QFP *1			
13	15	P35	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードとホールド機能が禁止のときに有効となります。
		$\overline{\text{HAK}}$		ホールドアクノリッジ出力端子です。 この機能は外部バスが有効なモードでホールド機能が許可のときに有効となります。
14	16	P36	I	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードと外部レディ機能が禁止のときに有効となります。
		RDY		レディ入力端子です。 この機能は外部バスが有効なモードで外部レディ機能が許可のときに有効となります。
15	17	P37	H	汎用の入出力ポートです。 レジスタの設定により、プルアップ抵抗の有無が設定できます。 この機能はシングルチップモードとクロック出力指定が禁止のときに有効となります。
		CLK		クロック 出力端子です。 この機能は外部バスが有効なモードでクロック出力指定が許可のときに有効となります。
16	18	P40	G	汎用の入出力ポートです。 この機能は UART0 のシリアルデータ出力指定が禁止のときに有効となります。
		SOT0		UART0 のシリアルデータ出力端子です。 この機能は UART0 のシリアルデータ出力指定が許可のときに有効となります。
17	19	P41	G	汎用の入出力ポートです。 この機能は UART0 のシリアルクロック出力指定が禁止のときに有効となります。
		SCK0		UART0 のシリアルクロック入出力端子です。 この機能は UART0 のシリアルクロック出力指定が許可のときに有効となります。
18	20	P42	G	汎用の入出力ポートです。 この機能は常時有効です。
		SIN0		UART0 のシリアルデータ入力端子です。 UART0 が入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。

*1 : FPT-100P-M06

*2 : FPT-100P-M05

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

表 1.6-1 端子機能説明 (4 / 7)

端子番号		端子名	入出力回路形式 ^{*3}	機能
LQFP ^{*2}	QFP ^{*1}			
19	21	P43	G	汎用の入出力ポートです。 この機能は常時有効です。
		SIN1		UART1 のシリアルデータ入力端子です。 UART1 が入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。
20	22	P44	G	汎用の入出力ポートです。 この機能は UART1 のシリアルクロック出力指定が禁止のときに有効となります。
		SCK1		UART1 のシリアルクロック入出力端子です。 この機能は UART1 のシリアルクロック出力指定が許可のときに有効となります。
22	24	P45	G	汎用の入出力ポートです。 この機能は UART1 のシリアルデータ出力指定が禁止のときに有効となります。
		SOT1		UART1 のシリアルデータ出力端子です。 この機能は UART1 のシリアルデータ出力指定が許可のときに有効となります。
23	25	P46	G	汎用の入出力ポートです。 この機能は I/O 拡張シリアルインタフェースのシリアルデータ出力指定が禁止のときに有効となります。
		SOT2		I/O 拡張シリアルインタフェースのデータ出力端子です。 この機能は I/O 拡張シリアルインタフェースのシリアルデータ出力指定が許可のときに有効となります。
24	26	P47	G	汎用の入出力ポートです。 この機能は I/O 拡張シリアルインタフェースのシリアルクロック出力指定が禁止のときに有効となります。
		SCK2		I/O 拡張シリアルインタフェースのシリアルクロック入出力端子です。 この機能は I/O 拡張シリアルインタフェースのシリアルクロック出力指定が許可のときに有効となります。
26	28	P50	D	汎用の入出力ポートです。 この機能は常時有効です。
		SIN2		I/O 拡張シリアルインタフェースのデータ入力端子です。 シリアルデータが入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。
27 ~ 30	29 ~ 32	P51 ~ P54	D	汎用の入出力ポートです。 この機能は常時有効です。
		INT4 ~ INT7		INT4 ~ INT7 の外部割込み要求入力端子です。 外部割込み入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。

*1 : FPT-100P-M06

*2 : FPT-100P-M05

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

表 1.6-1 端子機能説明 (5 / 7)

端子番号		端子名	入出力回路形式 ^{*3}	機能
LQFP ^{*2}	QFP ^{*1}			
31	33	P55	D	汎用の入出力ポートです。 この機能は常時有効です。
		ADTG		8/10 ビット A/D コンバータの外部トリガ入力端子です。 8/10 ビット A/D コンバータが入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。
36 ~ 39	38 ~ 41	P60 ~ P63	E	汎用の入出力ポートです。 この機能はアナログ入力許可レジスタの指定がポートのときに有効となります。
		AN0 ~ AN3		8/10 ビット A/D コンバータのアナログ入力端子です。 この機能はアナログ入力許可レジスタの指定が許可のときに有効となります。
41 ~ 44	43 ~ 46	P64 ~ P67	E	汎用の入出力ポートです。 入力機能はアナログ入力許可レジスタの指定がポートのときに有効となります。
		AN4 ~ AN7		8/10 ビット A/D コンバータのアナログ入力端子です。 この機能はアナログ入力許可レジスタの指定が許可のときに有効となります。
45	47	P56	D	汎用の入出力ポートです。 この機能は常時有効です。
		TIN0		16 ビットリロードタイマ 0 のイベント入力端子です。 16 ビットリロードタイマ 0 が入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。
46	48	P57	D	汎用の入出力ポートです。 この機能は 16 ビットリロードタイマ 0 の出力指定が禁止のときに有効となります。
		TOT0		16 ビットリロードタイマ 0 の出力端子です。 この機能は 16 ビットリロードタイマ 0 の出力指定が許可のときに有効となります。
51 ~ 56	53 ~ 58	P70 ~ P75	D	汎用の入出力ポートです。 この機能は常時有効です。
		IN0 ~ IN5		インプットキャプチャ ICU0 ~ ICU5 のトリガ入力端子です。 インプットキャプチャが入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。

*1 : FPT-100P-M06

*2 : FPT-100P-M05

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

表 1.6-1 端子機能説明 (6 / 7)

端子番号		端子名	入出力回路形式 ^{*3}	機能
LQFP ^{*2}	QFP ^{*1}			
57, 58	59, 60	P76, P77	D	汎用の入出力ポートです。 この機能はアウトプットコンペア (OCU) の出力指定が禁止のときに有効となります。
		OUT2, OUT3		アウトプットコンペア OCU2 と OCU3 のイベント出力端子です。 この機能はアウトプットコンペア (OCU) の出力指定が許可のときに有効となります。
		IN6, IN7		インプットキャプチャ ICU6 および ICU7 のトリガ入力端子です。 インプットキャプチャが入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定し、アウトプットコンペア出力を禁止してください。
59 ~ 62	61 ~ 64	P80 ~ P83	D	汎用の入出力ポートです。 この機能は 8/16 ビット PPG タイマの波形出力指定が禁止のときに有効となります。
		PPG0 ~ PPG3		8/16 ビット PPG タイマの出力端子です。 この機能は 8/16 ビット PPG タイマの波形出力指定が許可のときに有効となります。
63, 64	65, 66	P84, P85	D	汎用の入出力ポートです。 この機能はアウトプットコンペア (OCU) の出力指定が禁止のときに有効となります。
		OUT0, OUT1		アウトプットコンペア OCU0 と OCU1 のイベント出力端子です。 この機能はアウトプットコンペア (OCU) の出力指定が許可のときに有効となります。
65	67	P86	D	汎用の入出力ポートです。 この機能は常時有効です。
		TIN1		16 ビットリロードタイマ 1 の入力端子です。 16 ビットリロードタイマが入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。
66	68	P87	D	汎用の入出力ポートです。 この機能は 16 ビットリロードタイマ 0 の出力指定が禁止のときに有効となります。
		TOT1		16 ビットリロードタイマ 1 の出力端子です。 この機能は 16 ビットリロードタイマ 1 の出力指定が許可のときに有効となります。
67 ~ 70	69 ~ 72	P90 ~ P93	D	汎用の入出力ポートです。 この機能は常時有効です。
		INT0 ~ INT3		INT0 ~ INT3 の外部割込み要求入力端子です。 外部割込み入力動作をしている間に対応するポート方向レジスタ (DDR0 ~ DDRA) を入力に設定してください。

*1 : FPT-100P-M06

*2 : FPT-100P-M05

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

表 1.6-1 端子機能説明 (7 / 7)

端子番号		端子名	入出力回路形式 ^{*3}	機能
LQFP ^{*2}	QFP ^{*1}			
71	73	P94	D	汎用の入出力ポートです。 この機能は CAN0 の出力指定が禁止のときに有効となります。
		TX0		CAN0 用の TX 出力端子です。 この機能は CAN0 の出力指定が許可のときに有効となります。
72	74	P95	D	汎用の入出力ポートです。 この機能は常時有効です。
		RX0		CAN0 インタフェース用の RX 入力端子です。 CAN 機能使用時は、ほかの機能による出力を停止させておく必要があります。
73	75	P96	D	汎用の入出力ポートです。 この機能は CAN1 の出力指定が禁止のときに有効となります。
		TX1		CAN1 用の TX 出力端子です。 この機能は CAN1 の出力指定が許可のときに有効となります (MB90540 シリーズのみ)。
74	76	P97	D	汎用の入出力ポートです。 この機能は常時有効です。
		RX1		CAN1 インタフェース用の RX 入力端子です。 CAN 機能使用時は、ほかの機能による出力を停止させておく必要があります (MB90540 シリーズのみ)。
76	78	PA0	D	汎用の入出力ポートです。 この機能は常時有効です。
32	34	AV _{CC}	電源	A/D コンバータの電源端子です。 この電源の投入 / 切断は必ず V _{CC} に AV _{CC} 以上の電位が印加してある状態で行ってください。
35	37	AV _{SS}	電源	A/D コンバータの電源端子です。
33	35	AVRH	電源	A/D コンバータの外部基準電源端子です。 この端子の投入 / 切断は必ず AV _{CC} に AVRH 以上の電位が印加してある状態で行ってください。
34	36	AVRL	電源	A/D コンバータの外部基準電源端子です。
47, 48	49, 50	MD0, MD1	C	動作モード指定用入力端子です。 V _{CC} あるいは V _{SS} に直接つないで使用してください。
49	51	MD2	F	動作モード指定用入力端子です。 V _{CC} あるいは V _{SS} に直接つないで使用してください。
25	27	C	—	電源安定化の容量端子です。 外部に 0.1 μF 程度のセラミックコンデンサを接続してください。
21, 82	23, 84	V _{CC}	電源	電源 (5.0 V) 入力端子です。
9, 40, 79	11, 42, 81	V _{SS}	電源	電源 (0.0 V) 入力端子です。

*1 : FPT-100P-M06

*2 : FPT-100P-M05

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

1.7 入出力回路形式

表 1.7-1 に、各端子の回路形式について示します。

■ 入出力回路形式

表 1.7-1 入出力回路形式 (1/3)

分類	回 路	備 考
A	<p>スタンバイ制御信号</p>	<ul style="list-style-type: none"> ・ 発振帰還抵抗 約1MΩ (高速発振) 約10MΩ (低速発振)
B	<p>R (プルアップ) R CMOS ヒステリシス入力</p>	<ul style="list-style-type: none"> ・ CMOSヒステリシス入力 ・ プルアップ抵抗値 約50kΩ
C	<p>R CMOS ヒステリシス入力</p>	<ul style="list-style-type: none"> ・ CMOSヒステリシス入力
D	<p>P-ch N-ch R CMOS ヒステリシス入力</p>	<ul style="list-style-type: none"> ・ CMOSレベル出力 ・ CMOSヒステリシス入力

表 1.7-1 入出力回路形式 (2/3)

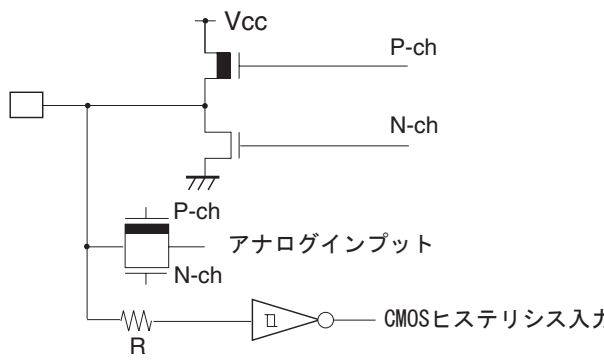
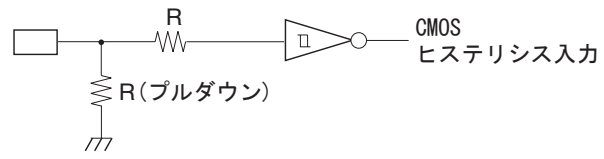
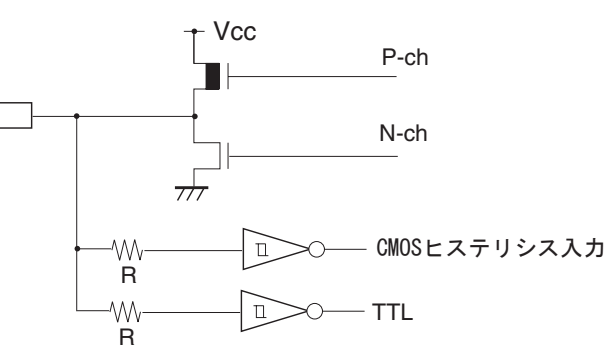
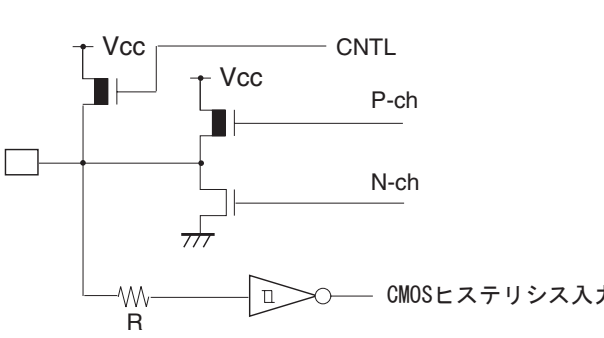
分類	回 路	備 考
E		<ul style="list-style-type: none"> ・ CMOS出力 ・ CMOSヒステリシス入力 ・ アナログ入力
F		<ul style="list-style-type: none"> ・ CMOSヒステリシス入力 ・ プルダウン抵抗 約50kΩ (フラッシュ品を除く)
G		<ul style="list-style-type: none"> ・ CMOS出力 ・ CMOSヒステリシス入力 ・ TTL入力 (フラッシュ書込みモードのフラッシュデバイスのみ)
H		<ul style="list-style-type: none"> ・ CMOS出力 ・ CMOSヒステリシス入力 ・ プログラマブルプルアップ抵抗 約50kΩ

表 1.7-1 入出力回路形式 (3/3)

分類	回 路	備 考
I		<ul style="list-style-type: none">・ CMOS出力・ CMOSヒステリシス入力・ TTL入力（フラッシュ書込みモードのフラッシュデバイス品のみ）・ プログラマブルプルアップ抵抗 約50kΩ

1.8 デバイスの取扱いに関する注意事項

デバイスを取り扱う際に注意が必要な事項について説明します。

■ デバイスの取扱いに関する注意事項

● 供給電圧の安定化

V_{CC} 電源電圧の動作保証内においても、電源電圧の急激な変化があると誤動作を起こす場合がありますので、 V_{CC} 電源電圧を安定させてください。

安定化の基準としては、商用周波数 (50Hz ~ 60Hz) での V_{CC} リプル変動 (peak to peak 値) は、標準 V_{CC} 電源電圧値の 10 % 以下に、また電源の切換え時などの瞬時変化においては、過度変動率が 0.1V/ms 以下になるように電源電圧を抑えてください。

● 最大定格電圧の厳守 (ラッチアップの防止)

最大定格を超えることのないよう注意してください (ラッチアップの防止)。

CMOS IC では、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または V_{CC} 端子と V_{SS} 端子との間に定格を超える電圧が印加された場合に、ラッチアップ現象を生じることがあります。

ラッチアップ現象が起きると電源電流が激増し、素子の熱破壊に至る場合がありますので、使用に際しては、最大定格を超えることのないよう十分注意してください。

また、アナログ電源投入時および切断時においてもアナログ電源電圧 (AV_{CC} , AV_{RH}) とアナログ入力電圧は、デジタル電源電圧 (V_{CC}) を超えることのないよう十分注意してください。

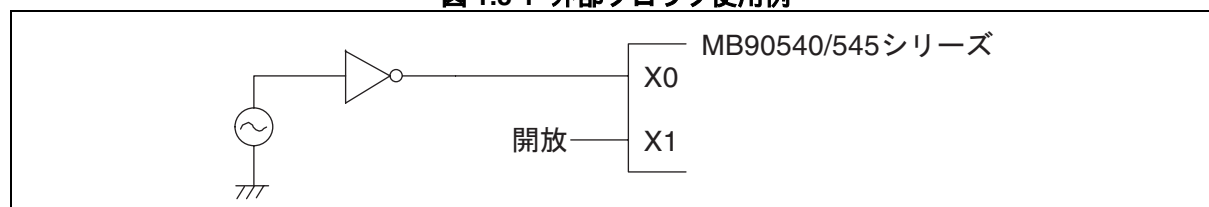
● 未使用端子の処理について

使用していない入力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損の原因となることがありますので、2k Ω 以上の抵抗を介して、プルアップまたはプルダウンの処理をしてください。また、使用していない入出力端子については、出力状態にして開放とするか、入力状態の場合は入力端子と同じ処置をしてください。

● 外部クロック使用時の注意について

外部クロックを使用する場合は、X0 端子のみを駆動し、X1 端子は開放してください。

図 1.8-1 外部クロック使用例



● 電源端子について (V_{CC}/V_{SS})

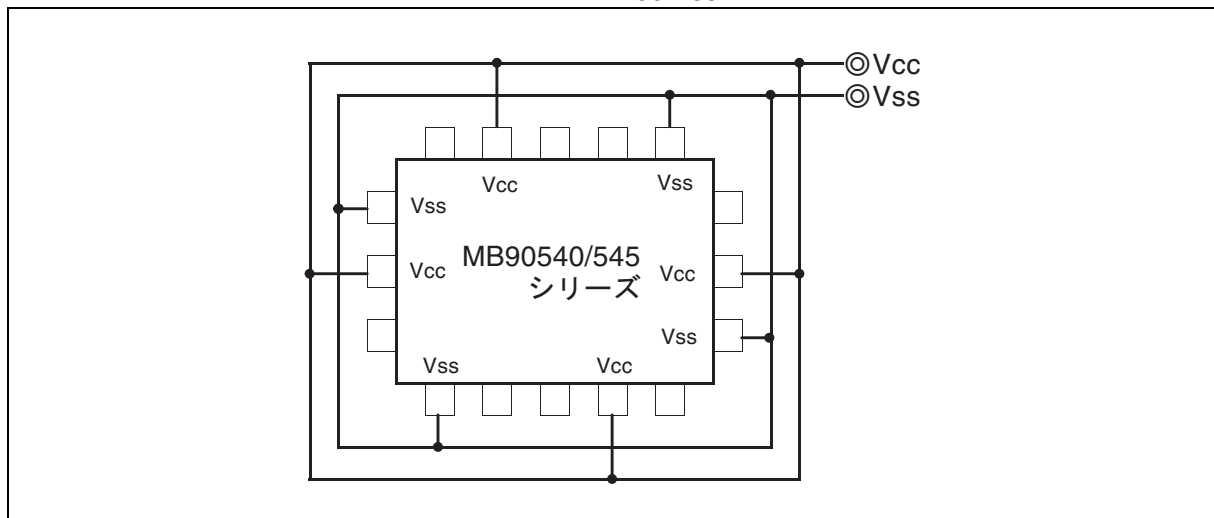
- V_{CC} , V_{SS} 端子が複数ある場合, デバイス設計上はラッチアップなどの誤動作を防止するために, 同電位にすべき端子はデバイス内部で接続してありますが, 不要輻射の低減, グラウンドレベルの上昇によるストローク信号の誤動作防止, 総出力電流規格を守るなどのために, 必ず V_{CC} , V_{SS} 端子を外部で電源およびグラウンドに接続してください (下図を参照)。

また, 電流供給源からできる限り低インピーダンスでこのデバイスの V_{CC} , V_{SS} に接続するように配慮してください。

さらに, このデバイスの近くで, V_{CC} と V_{SS} の間に $0.1\ \mu\text{F}$ 程度の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薦めします。

- 電源ノイズ対策として, このデバイスの近くで, V_{CC} と V_{SS} 端子の近くで, $0.1\ \mu\text{F}$ 程度の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薦めします。

図 1.8-2 電源端子 (V_{CC}/V_{SS}) の取扱い



● プルアップ / プルダウン抵抗器について

MB90540/545 シリーズは内蔵のプルアップ / プルダウン抵抗器をサポートしていません (ポート 0 ~ ポート 3 のプルアップ抵抗を除く) ので, 必要に応じて外付けしてください。

● 水晶発振回路について

X0, X1 端子の近辺のノイズはこのデバイスの誤動作の原因となります。X0, X1 端子および水晶振動子 (あるいはセラミック振動子) さらにグラウンドへのバイパスコンデンサはできる限り近くなるように, またその配線は, ほかの配線とできる限り交差しないようにプリント基板を設計してください。

また, X0, X1 端子の回りをグラウンドで囲むようなプリント基板アートワークは, 安定した動作を期待できますので, 強くお薦めします。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。

● A/D コンバータの電源, アナログ入力の投入順序について

A/D コンバータ, D/A コンバータの電源 (AV_{CC} , $AVRH$, $AVRL$) およびアナログ入力 ($AN0 \sim AN7$) の印加は, 必ずデジタル電源 (V_{CC}) の投入後に行ってください。

また, 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遮断の後で, デジタル

電源の遮断を行ってください。その際、 $AVRH$, AV_{CC} を超えないように投入・切断を行ってください(アナログ電源とデジタル電源を同時に投入、遮断することは問題ありません)。

● A/D コンバータ未使用時の端子処理について

A/D コンバータおよび D/A コンバータを使用しないときは、 $AV_{CC}=V_{CC}$, $AV_{SS}=AVRH=V_{SS}$ に接続してください。

● N.C. 端子の処理について

N.C.(内部接続)端子は、必ず開放にして使用してください。

● 電源投入時の注意点

内部に内蔵している降圧回路の誤動作を防ぐために、電源投入時における電圧の立上り時間は、 $50\mu s(0.2V \sim 2.7V \text{ 間})$ 以上を確保してください。

● サブクロック使用時の注意について

サブクロックを使用しない場合は、1 系統品をご使用ください。そのとき、 $X0A$ 端子にプルダウンの処理をして、 $X1A$ 端子は開放してください。2 系統品は必ず $X0A$, $X1A$ 端子に $32kHz$ 以下の振動子を接続してください。

● ポート 0, 1 からの出力が不定となる場合 (MB90F543/F549/V540/V540G のみ)

電源を投入後、降圧回路の発振安定待ち時間(パワーオンリセット中)の \overline{RST} 端子が "H" の場合、ポート 0, 1 から不定を出力します。 \overline{RST} 端子が "L" の場合、ポート 0, 1 はハイインピーダンス状態になります。タイミングは図 1.8-3, 図 1.8-4 のようになりますので注意してください。

図 1.8-3 ポート 0, 1 が不定出力になるタイミングチャート (\overline{RST} 端子が "H" の場合)

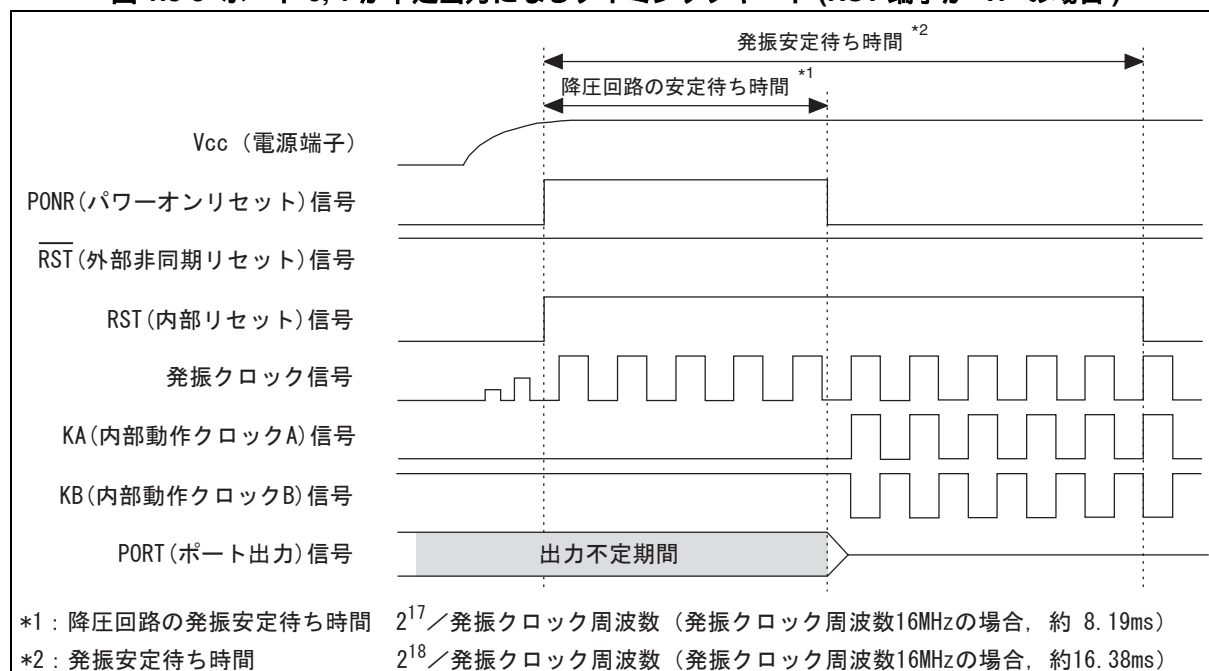
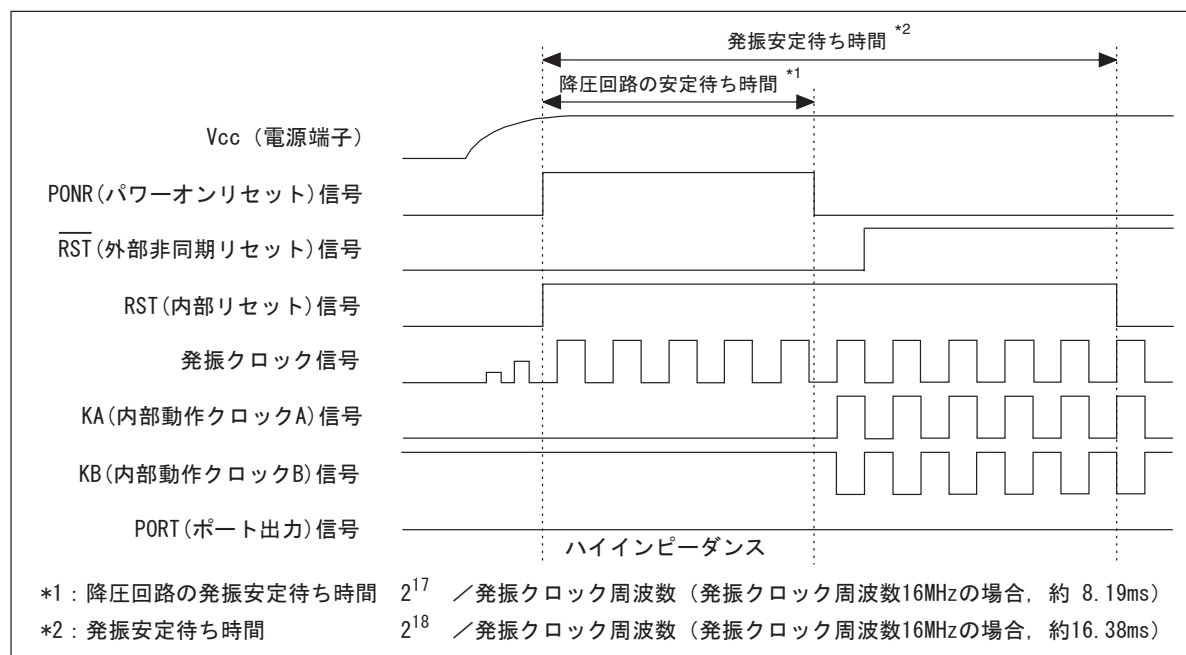


図 1.8-4 ポート 0, 1 がハイインピーダンス状態になるタイミングチャート
($\overline{\text{RST}}$ 端子が "L" の場合)



● 初期化について

デバイス内には、パワーオンリセットによってのみ初期化される内蔵レジスタ類があります。これらの初期化を期待する場合は電源の再投入を行ってください。

● 「DIV A,Ri」, 「DIVW A,RWi」 命令の使用上の注意

符号付乗除計算命令「DIV A,Ri」, 「DIVW A,RWi」 命令は, 対応するバンクレジスタ (DTB, ADB, USB, SSB) の値を "00_H" に設定し, 使用してください。

対応するバンクレジスタ (DTB, ADB, USB, SSB) の値を "00_H" 以外に設定した場合, 命令実行結果により得られる余りは, 命令オペランドのレジスタに格納されません。

詳細は, 「2.11 「DIV A,Ri」, 「DIVW A,RWi」 命令の使用上の注意」を参照してください。

● REALOS を使用する場合

REALOS を使用する場合は, 拡張インテリジェント I/O サービス (EI²OS) が使用できません。

● 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し続ける場合があります。この動作は, 保証外の動作です。

第2章

CPU

CPU について説明します。

- 2.1 CPU の概要
- 2.2 メモリ空間
- 2.3 メモリ空間マップ
- 2.4 リニア方式によるアドレス指定
- 2.5 バンク方式によるアドレス指定
- 2.6 メモリ空間における多バイト長データ
- 2.7 レジスタ
- 2.8 レジスタバンク
- 2.9 プリフィックスコード
- 2.10 割込み抑止命令
- 2.11 「DIV A,Ri」, 「DIVW A,RWi」命令の使用上の注意

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勿論、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 (32 ビットデータ処理にはいくつかの命令が使用されます)。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
62.5ns (4MHz 発振, 4 通倍)
- 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)
- 強力な割込み機能
8 つのプライオリティレベル (プログラマブル)
- CPU に依存しない自動転送機能
最大 16 チャネルまでの拡張インテリジェント I/O サービス
- 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 / 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

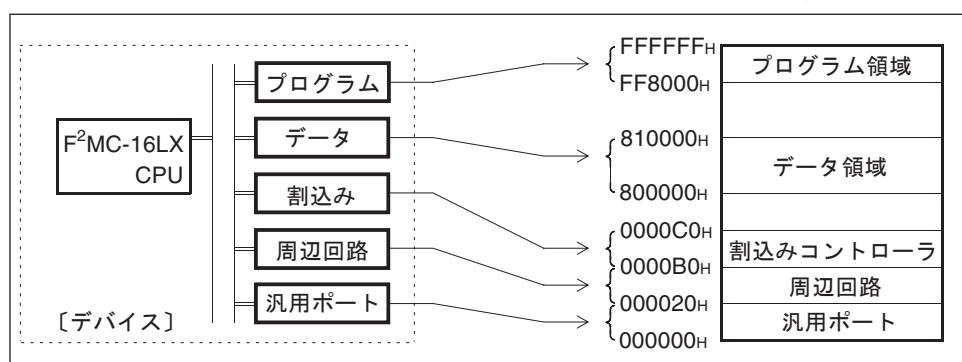
2.2 メモリ空間

F²MC-16LX CPU には 16M バイトのメモリ空間があります。F²MC-16LX CPU が管理するデータ・プログラム・I/O はすべてこの 16M バイトのメモリ空間に配置されます。CPU は 24 ビットのアドレスバスでこれらのアドレスを示すことにより、各リソースをアクセスすることができます。

■ CPU メモリ空間の概要

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



■ アドレス指定の方式

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 ビットアドレスすべて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 メモリ空間マップ

図 2.3-1 に , MB90540/545 シリーズのメモリ空間マップを示します。

■ メモリ空間マップ

図 2.3-1 に示すように , バンク 00 の上位部分は FF バンク ROM のイメージであり , C コンパイラのスモールモデルを有効にします。下位 16 ビットも同様なため , ポインタ宣言内の far 仕様を使用せずに ROM 内のテーブルを参照できます。例えば , "00C000_H" にアクセスしようとする , ROM 内の "FFC000_H" の値にアクセスすることになります。バンク FF 内の ROM 領域は 48K バイトを超え , バンク 00 内にその全体イメージを表示することはできません。

"FF4000_H" ~ "FFFFFF_H" の間のイメージはバンク 00 内で可視ですが , "FF0000_H" ~ "FF3FFF_H" の間のイメージはバンク FF 内でしか可視ではありません。

図 2.3-1 メモリ空間マップ

MB90V540/V540G/ F546G(S)		MB90543G(S)/F543/ F543G(S)		MB90F548G(S)/ F548GL(S)/ 548G(S)		MB90F549/549G(S)/ F549G(S)		MB90547G(S)	
FFFFFF _H	ROM (FFバンク)	FFFFFF _H	ROM (FFバンク)	FFFFFF _H	ROM (FFバンク)	FFFFFF _H	ROM (FFバンク)	FFFFFF _H	ROM (FFバンク)
FF0000 _H		FF0000 _H		FF0000 _H		FF0000 _H		FF0000 _H	
FEFFFF _H	ROM (FEバンク)	FEFFFF _H	ROM (FEバンク)	FEFFFF _H	ROM (FEバンク)	FEFFFF _H	ROM (FEバンク)		
FE0000 _H		FE0000 _H		FE0000 _H		FE0000 _H			
FDFFFF _H	ROM (FDバンク)					FDFFFF _H	ROM (FDバンク)		
FD0000 _H						FD0000 _H			
FCFFFF _H	ROM (FCバンク)					FCFFFF _H	ROM (FCバンク)		
FC0000 _H	外部アクセス メモリ		外部アクセス メモリ		外部アクセス メモリ	FC0000 _H	外部アクセス メモリ		外部アクセス メモリ
00FFFF _H	ROM (FFバンクの イメージ)	00FFFF _H	ROM (FFバンクの イメージ)	00FFFF _H	ROM (FFバンクの イメージ)	00FFFF _H	ROM (FFバンクの イメージ)	00FFFF _H	ROM (FFバンクの イメージ)
004000 _H		004000 _H		004000 _H		004000 _H		004000 _H	
003FFF _H	周辺	003FFF _H	周辺	003FFF _H	周辺	003FFF _H	周辺	003FFF _H	周辺
003900 _H	外部アクセスメモリ	003900 _H	外部アクセス メモリ	003900 _H	外部アクセス メモリ	003900 _H	外部アクセス メモリ	003900 _H	外部アクセス メモリ
0020FF _H		002000 _H		002000 _H		002100 _H		002000 _H	
001FF5 _H	ROM補正								
001FF0 _H		0018FF _H		0010FF _H		0018FF _H			
	RAM 8Kバイト		RAM 6Kバイト		RAM 4Kバイト		RAM 6Kバイト	0008FF _H	RAM 2Kバイト
000100 _H	外部アクセスメモリ	000100 _H	外部アクセスメモリ	000100 _H	外部アクセスメモリ	000100 _H	外部アクセスメモリ	000100 _H	外部アクセスメモリ
0000BF _H		0000BF _H		0000BF _H		0000BF _H		0000BF _H	
000000 _H	周辺	000000 _H	周辺	000000 _H	周辺	000000 _H	周辺	000000 _H	周辺

*:MB90F549 は "002000_H" となります。

2.4 リニア方式によるアドレス指定

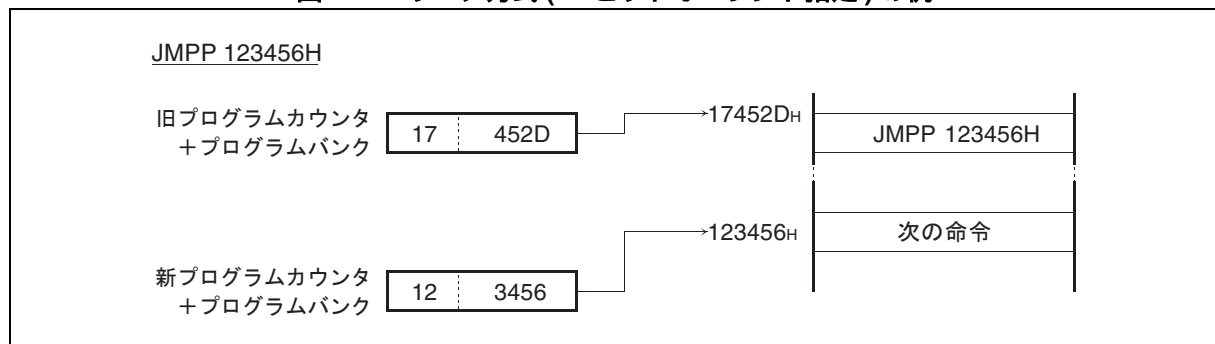
リニア方式には以下に示す2つの方式があります。

- 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引用する方式

■ 24ビットオペランド指定

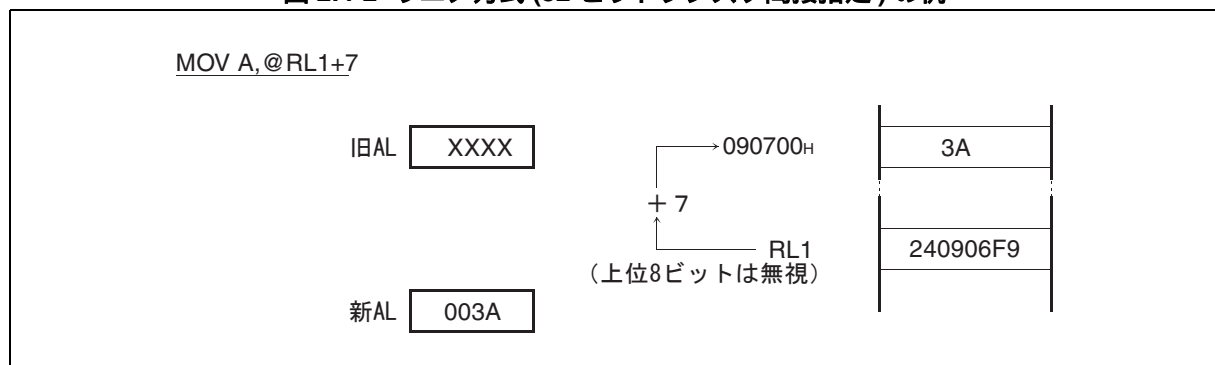
図2.4-1に、24ビットオペランド指定の例を、図2.4-2に、32ビットレジスタ間接指定の例を示します。

図2.4-1 リニア方式(24ビットオペランド指定)の例



■ 32ビットレジスタ間接指定

図2.4-2 リニア方式(32ビットレジスタ間接指定)の例



2.5 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 64K バイトごとの 256 個のバンクに分割し、以下に示す 5 個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 64K 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 64K 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)あるい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 64K 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際にスタックアクセスが生じ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 64K 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 2.5-1 に示すように、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、命令のコード効率を向上するために事前に決められています。また、あるアドレッシングを用い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することができます。

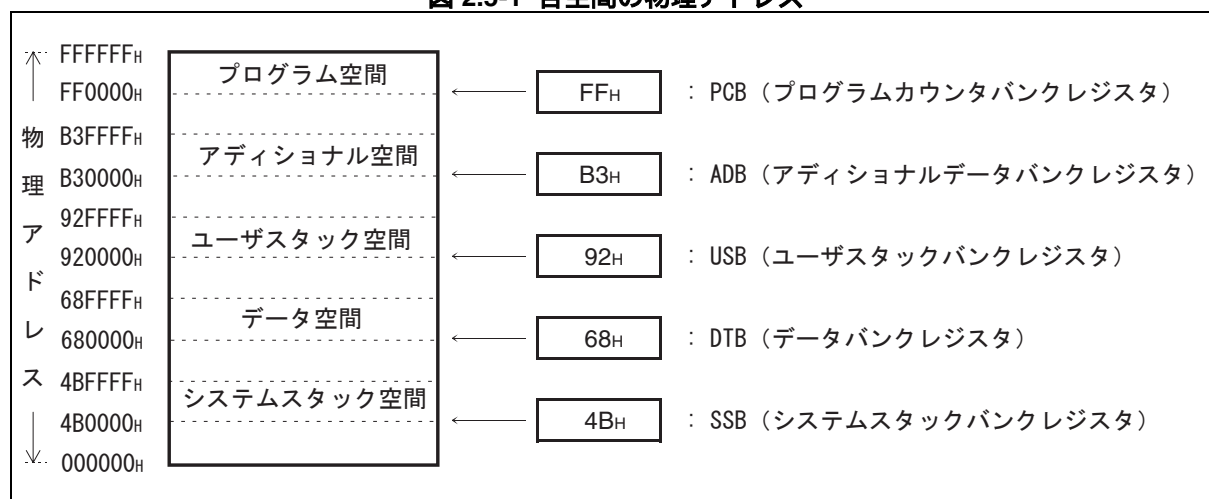
DTB, USB, SSB, ADB はリセットにより "00_H" に初期化され、PCB はリセットベクタにより指定された値に初期化されます。リセット後、DT, SP, AD の各空間はバンク "00_H" (000000_H ~ 00FFFF_H) に配置され、PC 空間はリセットベクタにより指定されたバンクに配置されます。

表 2.5-1 デフォルト空間

デフォルト空間	アドレッシングモード
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 を用いたアドレッシングモード, @A, addr16, dir
スタック空間	PUSHW, POPW, @RW3,@RW7 を用いたアドレッシングモード
アディショナル空間	@RW2,@RW6 を用いたアドレッシングモード

図 2.5-1 に, レジスタバンクに分割されたメモリ空間の例を示します。

図 2.5-1 各空間の物理アドレス



2.6 メモリ空間における多バイト長データ

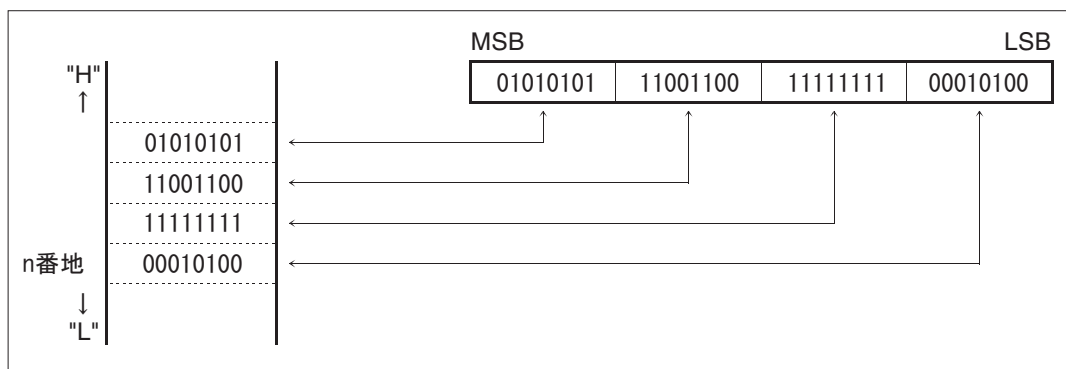
メモリへの書込みはアドレスの低い方から順に行われます。したがって、32 ビット長データであれば下位 16 ビットが先に転送され、続いて上位 16 ビットが転送されます。

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。

■ メモリ空間における多バイト長データの配置

図 2.6-1 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。

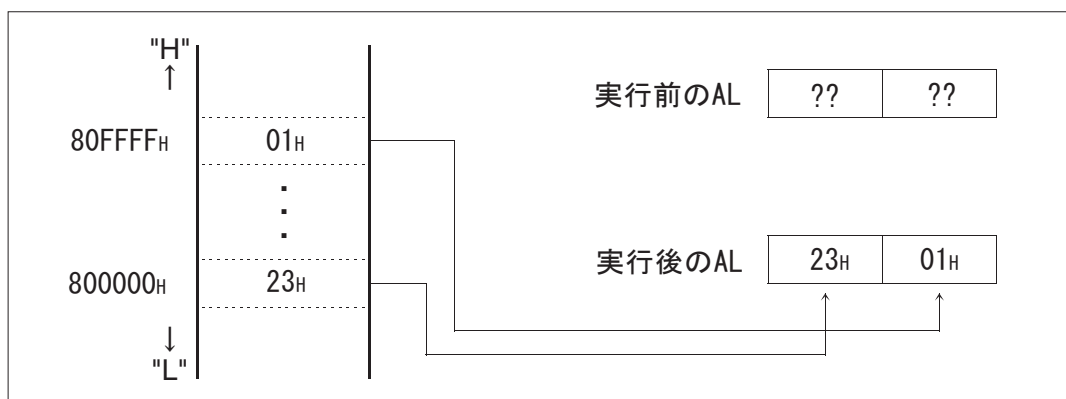
図 2.6-1 メモリにおける多バイト長データの配置の例



■ 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、"FFFF_H" 番地の次のアドレスは同じバンクの "0000_H" 番地になります。図 2.6-2 に、多バイト長データのアクセス命令の例を示します。

図 2.6-2 MOVW A, 080FFFFH の実行



2.7 レジスタ

F²MC-16LX のレジスタは大別して、CPU 内の専用レジスタと、メモリ内の汎用レジスタの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、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。

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。

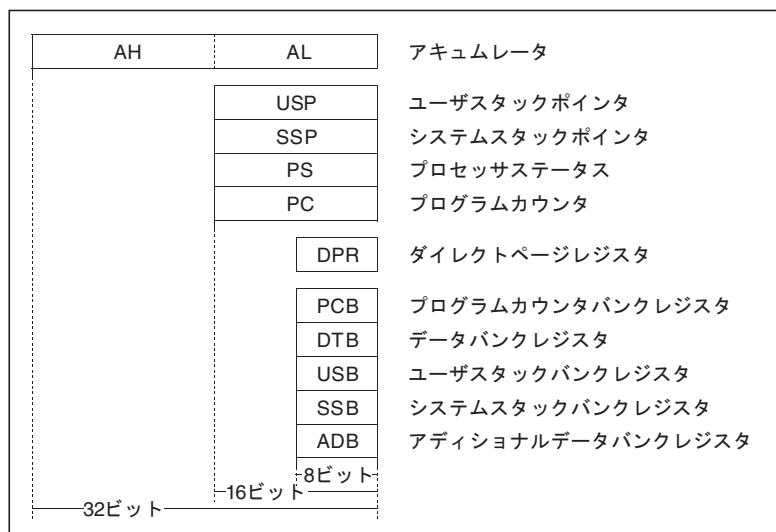
■ 専用レジスタ

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): A/D 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 2.7-1 に、専用レジスタの構成を示します。

図 2.7-1 専用レジスタ

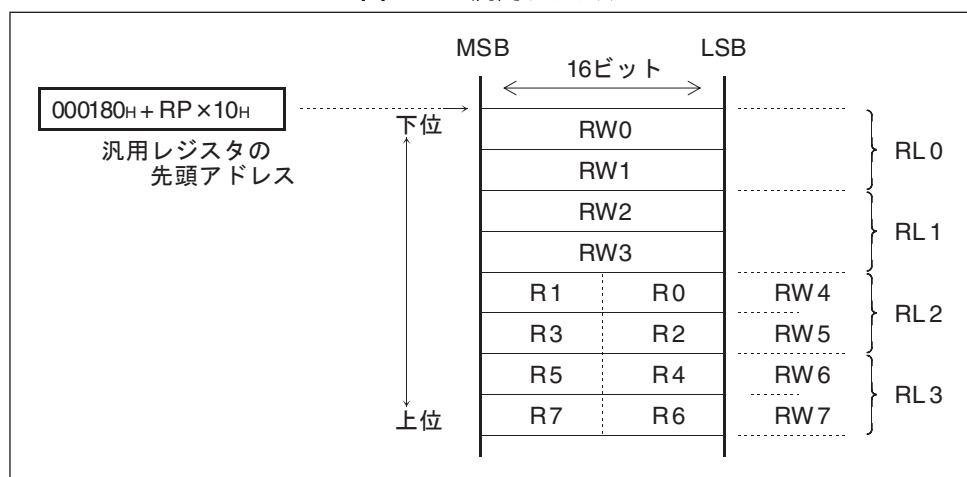


■ 汎用レジスタ

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。これらのレジスタは、図 2.7-2 に示すように互いに依存しています。

- 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 2.7-2 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、RL_i 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。

2.7.1 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、演算などを行った際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)

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、16ビットデータ処理モードのワード処理や8ビットデータ処理モードのバイト処理のときはALのみが使用されます(図2.7-3と図2.7-4を参照)。Aレジスタ中のデータはメモリ/レジスタ(Ri, RWi, RLi)中のデータと各種演算ができ、F²MC-8Lのときと同様、ワード長以下のデータをALへ転送しますと、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)ので、データ保持機能とAL-AH間演算に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ビットは無視され演算結果の上位8ビットはすべて"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 2.7-3 32 ビットデータ転送例

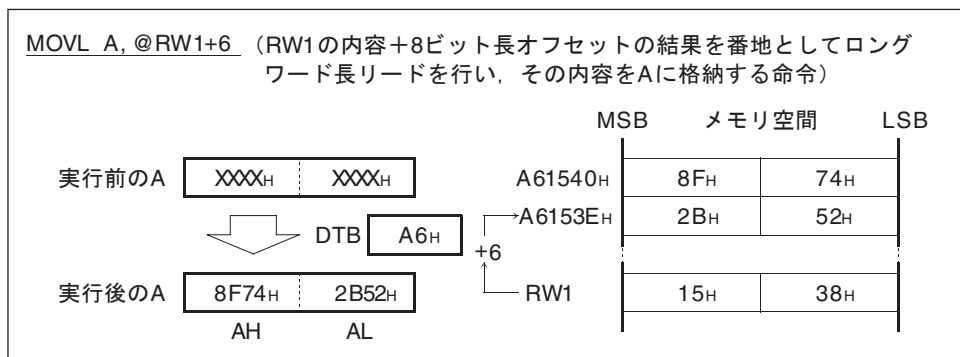
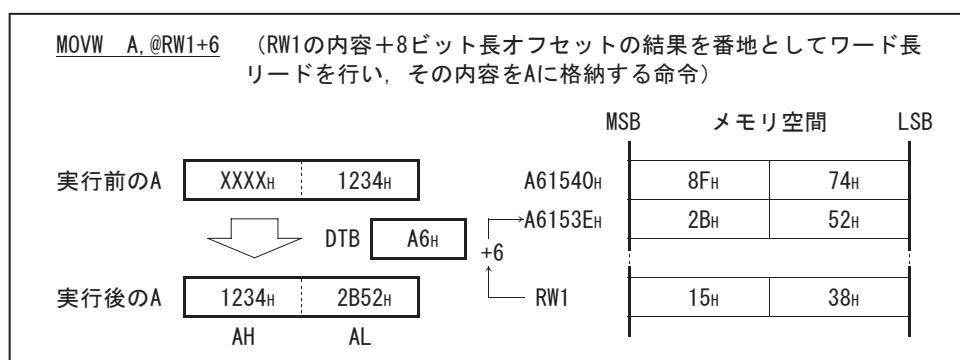


図 2.7-4 AL-AH 転送例



2.7.2 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)

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レジスタ中の 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 (図 2.7-5 を参照)。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は USP が用いられます。スタック空間を分ける必要のない場合は SSP だけをお使いください。

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 2.7-5 スタック操作命令とスタックポインタ (S フラグが "0" の場合の PUSHW A の例)

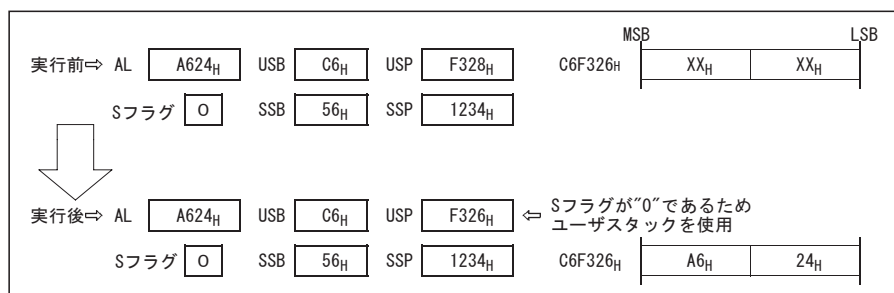
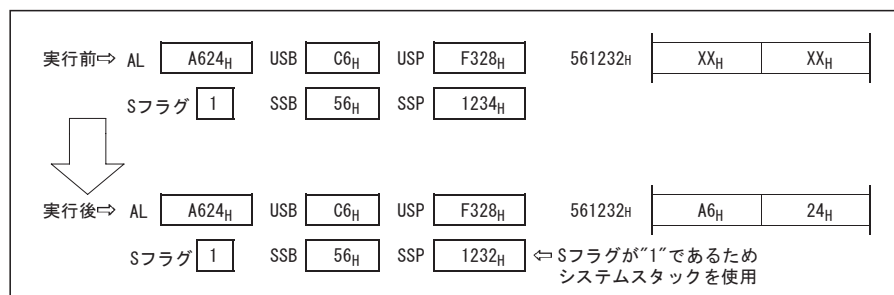


図 2.7-6 スタック操作命令とスタックポインタ (S フラグが "1" の場合の PUSHW A の例)



< 注意事項 >

スタックポインタに設定すべき値は、原則として偶数アドレスを使用してください。

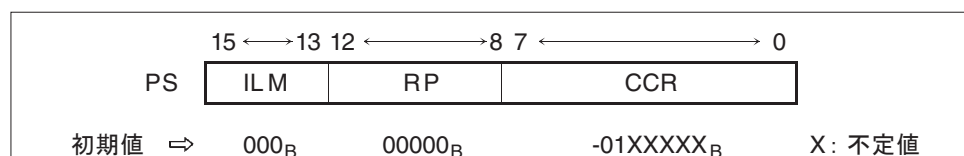
2.7.3 プロセッサステータス (PS)

PS レジスタは、CPU の動作制御を行うビットと 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 2.7-7 に示すように、PS レジスタはレジスタバンクポインタ (RP) および割込みレベルマスクレジスタ (ILM) により構成されています。RP はレジスタバンクの先頭アドレスを示します。PS レジスタの下位バイトは命令実行結果および割込み発生などによりセット / リセットされる各種フラグより構成されているコンディションコードレジスタ (CCR) より構成されています。

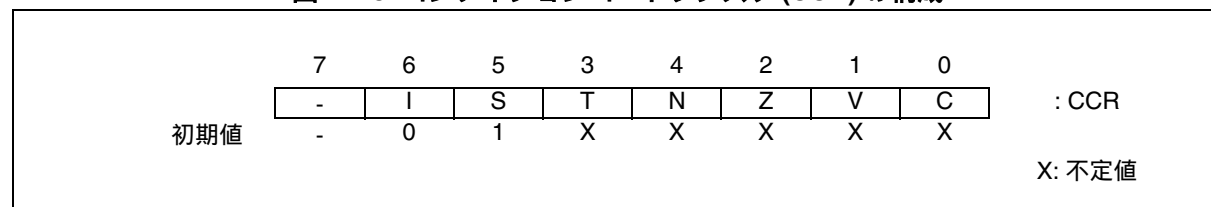
図 2.7-7 プロセッサステータス (PS) の構造



■ コンディションコードレジスタ (CCR)

コンディションコードレジスタ (CCR) のビット構成を図 2.7-8 に示します。

図 2.7-8 コンディションコードレジスタ (CCR) の構成



● 割込み許可フラグ (I)

ソフトウェアインタラプト以外のすべての割込みに対し I フラグが "1" の場合は割込みが許可, "0" の場合、割込みはマスクされます。リセット時はクリアされます。

● スタックフラグ (S)

S フラグが "0" の場合はスタック操作用ポインタとして USP が有効, "1" の場合は SSP が有効になります。割込み受け付け時またはリセット時にセットされます。

● スティックイビットフラグ (T)

論理右 / 算術右シフト命令を実行後にキャリよりシフトアウトされたデータに 1 つ以上 "1" があったら "1", それ以外は "0" になります。シフト量がゼロの場合も "0" になります。

● ネガティブフラグ (N)

演算結果の MSB が "1" なら N フラグはセット, それ以外はクリアされます。

● ゼロフラグ (Z)

演算結果がすべて "0" ならセット, それ以外はクリアされます。

● 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生じるとセット,生じないとクリアされます。

● キャリフラグ (C)

演算の実行により, MSB より桁上り/桁下りが生じるとセット,生じないとクリアされます。

■ レジスタバンクポインタ (RP)

レジスタバンクポインタ (RP) は, F²MC-16LX の持つ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 (図 2.7-9 参照)。RP レジスタは 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, メモリ中の "000180_H" ~ "00037F_H" のアドレス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 でなければ汎用レジスタとして使用することはできません。RP レジスタ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 レジスタに 8 ビットの即値が転送できますが, 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。

レジスタバンクポインタ (RP) のビット構成を図 2.7-9 に示します。

図 2.7-9 レジスタバンクポインタ (RP)

	B4	B3	B2	B1	B0	: RP
初期値	0	0	0	0	0	

■ 割込みレベルマスクレジスタ (ILM)

割込みレベルマスクレジスタ (ILM) 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。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 (表 2.7-1 参照)。したがって、割込みが受け付けられるには、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 にセットされ、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 レジスタに 8 ビットの即値が転送できますが、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。

割込みレベルマスクレジスタ (ILM) のビット構成を図 2.7-10 に示します。

図 2.7-10 割込みレベルマスクレジスタ

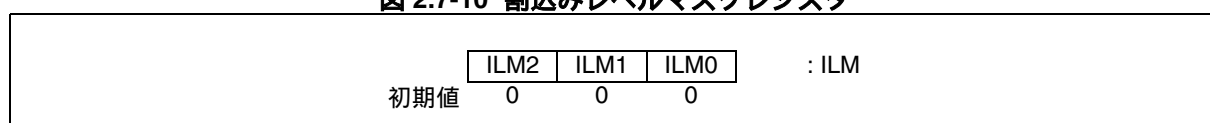


表 2.7-1 割込みレベルマスク (ILM) レジスタで示されるレベル

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	"0" のみ
0	1	0	2	"1" より小さい値のレベル
0	1	1	3	"2" より小さい値のレベル
1	0	0	4	"3" より小さい値のレベル
1	0	1	5	"4" より小さい値のレベル
1	1	0	6	"5" より小さい値のレベル
1	1	1	7	"6" より小さい値のレベル

2.7.4 プログラムカウンタ (PC)

プログラムカウンタレジスタは、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 レジスタは条件分岐命令、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。

■ プログラムカウンタ (PC)

図 2.7-11 に、プログラムカウンタを示します。

図 2.7-11 プログラムカウンタ



2.8 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。

■ レジスタバンク

表 2.8-1 に、各レジスタの機能を、表 2.8-2 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 領域と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 2.8-1 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または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

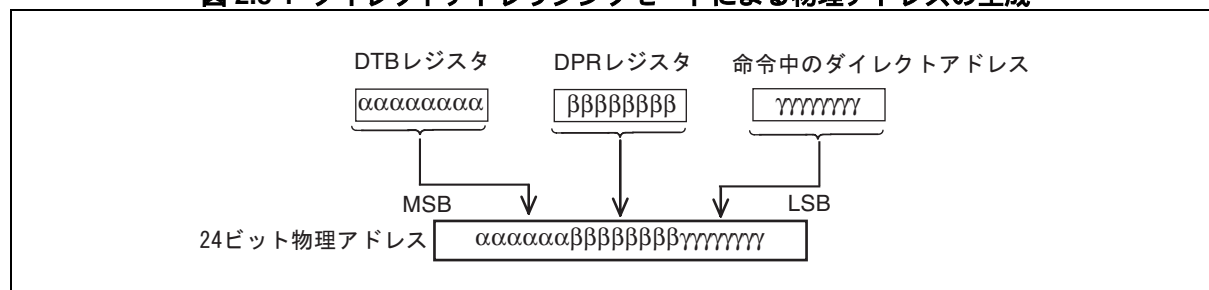
表 2.8-2 各レジスタの関係

	RW0	RL0
	RW1	
	RW2	RL1
	RW3	
R0	RW4	RL2
R1		
R2	RW5	
R3		
R4	RW6	RL3
R5		
R6	RW7	
R7		

● 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 2.8-1 に示すように、ダイレクトアドレッシングモードの命令時のオペランドの addr8 ~ addr15 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。DPR は命令により読出しや書込みができます。

図 2.8-1 ダイレクトアドレッシングモードによる物理アドレスの生成



- 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 >

各々のバンクレジスタはそれぞれ PC 空間, DT 空間, SP 空間 (ユーザ), SP 空間 (システム), AD 空間の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バイト長であり, リセットにより PCB はリセットベクタの値に初期化されます。PCB 以外のバンクレジスタは読出し / 書込みができます。PCB は, 読出しは可能ですが書き込むことはできません。

16M バイト全空間に分岐する JMPP, CALLP, RETP, RETI, RETF 命令実行時および割込み時に PCB は書き換わります。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。

2.9 プリフィックスコード

命令の前にプリフィックスコードを置くことで、命令動作の一部を変更することが可能です。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

データアクセスの際に使用されるメモリ空間は各アドレッシングモード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無関係に任意に選択することが可能となります。

表 2.9-1 に、バンクセレクトプリフィックスと対応するメモリ空間を示します。

表 2.9-1 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 空間のどちらかが用いられます。

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〔MOV S/MOV SW/SCEQ/SCWEQ/FIL S/FIL SW〕

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てください。

- スタック操作命令〔PUSH W/POP W〕

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 を使用してください。

- I/O アクセス命令〔MOV A,io/MOV io,A/MOV X A,io/MOV W A,io/MOV W io,A/MOV io,#imm8/MOV W io,#imm16/MOV B A,io:bp/MOV B io:bp,A/SET B io:bp/CLR B 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 空間が使用されます。

- 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。

- POP W 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 が使用されます。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

プリフィックス有無にかかわらず、SSB が使用されます。

■ 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ためには、その場合の RP がどのような値でも、比較的簡単に定められた同一のレジスタバンクをアクセスする手段が必要です。コモンレジスタバンクプリフィックス (CMR) を、レジスタバンクをアクセスする命令の前に置くことで、現在の RP の値に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 の場合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。

コモンレジスタバンクプリフィックス (CMR) を使用する際には、以下に示す命令に対して注意してください。

- スtring命令〔 MOV S/MOV S W/SCEQ/SCWEQ/FIL S/FIL S W 〕

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/POP W 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ためには、フラグ変化抑止プリフィックスコード (NCC) を用います。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。

フラグ変化抑止プリフィックス (NCC) を使用する際には、以下に示す命令に対して注意してください。

- スtring命令〔 MOV S/MOV S W/SCEQ/SCWEQ/FIL S/FIL S W 〕

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/POP W 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/RET I 〕

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。

2.10 割込み抑止命令

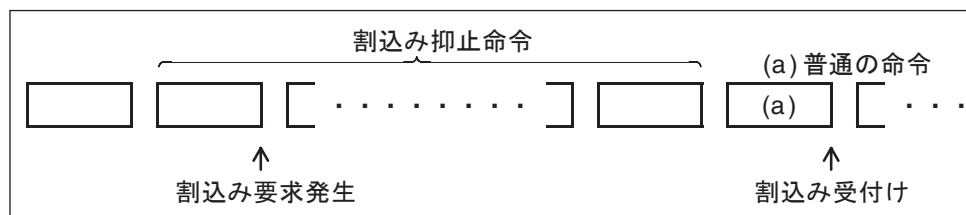
以下に示す 10 種類の命令については、割込み要求はサンプリングされません。

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

■ 割込み抑止命令

上記の命令実行中に有効なハードウェア割込み要求が発生しても、割込み処理が行われるのは、上記以外の命令が実行されたときに限られます。詳しくは図 2.10-1 を参照してください。

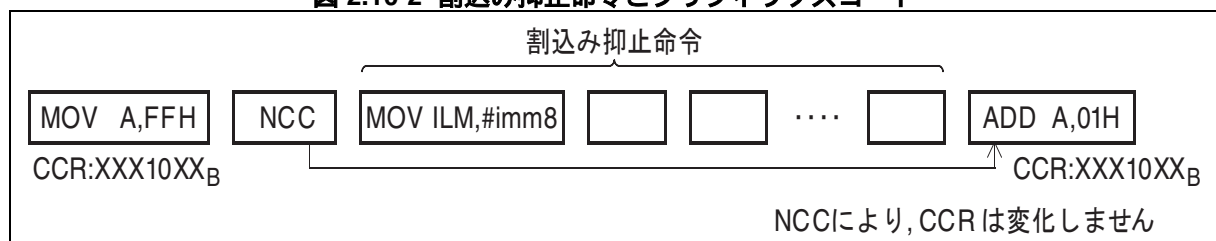
図 2.10-1 割込み抑止命令



■ 割込み抑止命令とプリフィックス命令に関する制約

割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、割込み抑止命令以外のコード後の最初の〔命令〕まで及びます。

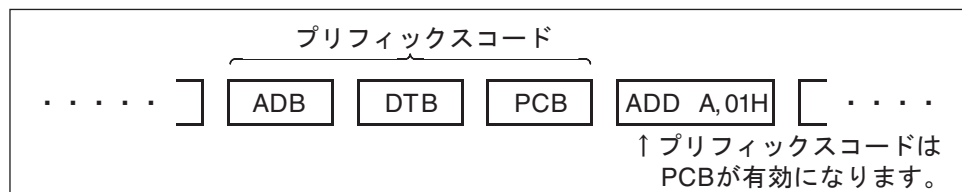
図 2.10-2 割込み抑止命令とプリフィックスコード



■ プリフィックスコードが連続している場合

競合するプリフィックスコードが連続していた場合、後方のものが有効になります。
なお、競合するプリフィックスコードとはここでいう PCB, ADB, DTB, SPB のことを意味します。

図 2.10-3 プリフィックスコードの連続



2.11 「DIV A,Ri」, 「DIVW A,RWi」命令の使用上の注意

「DIV A,Ri」, 「DIVW A,RWi」命令を使用する場合は, バンクレジスタを "00_H" に設定してください。

■「DIV A,Ri」, 「DIVW A,RWi」命令の使用上の注意

表 2.11-1 「DIV A,Ri」, 「DIVW A,RWi」命令の使用上の注意 (i=1 ~ 7)

命令	左記命令実行時に影響を受けるバンクレジスタ名	余りが格納されるアドレス
DIV A,R0	DTB	(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +8 _H : 下位 16 ビット)
DIV A,R1		(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +9 _H : 下位 16 ビット)
DIV A,R4		(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +C _H : 下位 16 ビット)
DIV A,R5		(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +D _H : 下位 16 ビット)
DIVW A,RW0		(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +0 _H : 下位 16 ビット)
DIVW A,RW1		(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +2 _H : 下位 16 ビット)
DIVW A,RW4		(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +8 _H : 下位 16 ビット)
DIVW A,RW5		(DTB: 上位 8 ビット)+(0180 _H +RP × 10 _H +A _H : 下位 16 ビット)
DIV A,R2	ADB	(ADB: 上位 8 ビット)+(0180 _H +RP × 10 _H +A _H : 下位 16 ビット)
DIV A,R6		(ADB: 上位 8 ビット)+(0180 _H +RP × 10 _H +E _H : 下位 16 ビット)
DIVW A,RW2		(ADB: 上位 8 ビット)+(0180 _H +RP × 10 _H +4 _H : 下位 16 ビット)
DIVW A,RW6		(ADB: 上位 8 ビット)+(0180 _H +RP × 10 _H +E _H : 下位 16 ビット)
DIV A,R3	USB SSB ^{*1}	(USB ^{*2} : 上位 8 ビット)+(0180 _H +RP × 10 _H +B _H : 下位 16 ビット)
DIV A,R7		(USB ^{*2} : 上位 8 ビット)+(0180 _H +RP × 10 _H +F _H : 下位 16 ビット)
DIVW A,RW3		(USB ^{*2} : 上位 8 ビット)+(0180 _H +RP × 10 _H +6 _H : 下位 16 ビット)
DIVW A,RW7		(USB ^{*2} : 上位 8 ビット)+(0180 _H +RP × 10 _H +E _H : 下位 16 ビット)

*1:CCR レジスタの S ビットによる

*2:CCR レジスタの S ビットが "0" の場合

バンクレジスタ (DTB, ADB, USB, SSB) の値が "00_H" の場合は, 除算結果の余りが命令オペランドのレジスタに格納されます。バンクレジスタの値が "00_H" 以外の場合は, 上位 8 ビットアドレスは命令オペランドのレジスタに対応したバンクレジスタにより指定され, 下位 16 ビットアドレスは命令オペランドのレジスタのアドレスと同じアドレスとなり, 上位 8 ビットで指定されたバンクのレジスタに余りが格納されます。

【例】

DTB=053_H, RP=03_H の場合に「DIV A,R0」を実行した場合は, R0 のアドレスが 0180_H+RP(03_H) × 10_H+08_H(R0 相当アドレス)=001B8_H となります。

ここで, 「DIV A,R0」で指定されるバンクレジスタはデータバンクレジスタ (DTB) です。バンクアドレス "053_H" を付加したアドレス "05301B8_H" に余りが格納されます。

< 参考 >

バンクレジスタおよび Ri, RWi のレジスタについては, 「 2.7 レジスタ 」を参照してください。

■ 注意事項の回避について

「DIV A,Ri」, 「DIVW A,RWi」命令の使用上の注意事項を回避してプログラムを開発していただくために, コンパイラでは表 2.11-1 の命令を生成しないように変更し, アセンブラでは表 2.11-1 の命令を同等の命令列に置き換える機能を追加したものが用意されています。MB90540/545 シリーズのコンパイラおよびアセンブラは, 以下のものを使用してください。

- コンパイラ
cc907 の V02L06 以降のバージョンおよび fcc907s の V30L02 以降のバージョン
- アセンブラ
asm907a の V03L04 以降のバージョンおよび fasm907s の V30L04 (Rev.300004) 以降のバージョン

第3章

割込み

割込みの機能と動作について説明します。

- 3.1 割込みの概要
- 3.2 割込み要因
- 3.3 割込みベクタ
- 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)
- 3.7 未定義命令の実行による例外発生

3.1 割込みの概要

F²MC-16LX はイベントなどの発生により、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割込み機能があります。

■ 割込みの概要

割込み機能は以下の 4 つに分けることができます。

- ハードウェア割込み：内蔵リソースのイベント発生による割込み処理
- ソフトウェア割込み：ソフトウェアのイベント発生命令による割込み処理
- 拡張インテリジェント I/O サービス (EI²OS):
内蔵リソースのイベント発生による転送処理
- 例外：動作例外事項の発生による終了処理

この章では、これら 4 種類の割込みについて説明します。

3.2 割り込み要因

表 3.2-1 に、MB90540/545 シリーズでの割り込み要因と割り込みベクタ・割り込み制御レジスタを示します。

■ 割り込み要因

表 3.2-1 割り込み要因と割り込みベクタ・割り込み制御レジスタ

割り込み要因	EI ² OS クリア	割り込みベクタ		割り込み制御レジスタ	
		番号	アドレス	番号	アドレス
リセット	×	# 08	FFFFDC _H	-	-
INT9 命令	×	# 09	FFFFD8 _H	-	-
例外	×	# 10	FFFFD4 _H	-	-
CAN 0 RX	×	# 11	FFFFD0 _H	ICR00	0000B0 _H
CAN 0 TX/NS	×	# 12	FFFFCC _H		
CAN 1 RX	×	# 13	FFFFC8 _H	ICR01	0000B1 _H
CAN 1 TX/NS	×	# 14	FFFFC4 _H		
外部割り込み INT0/INT1		# 15	FFFFC0 _H	ICR02	0000B2 _H
タイムベースタイマ	×	# 16	FFFFBC _H		
16 ビットリロードタイマ 0		# 17	FFFFB8 _H	ICR03	0000B3 _H
A/D コンバータ		# 18	FFFFB4 _H		
入出力タイマ	×	# 19	FFFFB0 _H	ICR04	0000B4 _H
外部割り込み INT2/INT3		# 20	FFFFAC _H		
シリアル I/O		# 21	FFFFA8 _H	ICR05	0000B5 _H
PPG(ch0, ch1) ユニット 0	×	# 22	FFFFA4 _H		
インプットキャプチャ 0		# 23	FFFFA0 _H	ICR06	0000B6 _H
外部割り込み INT4/INT5		# 24	FFFF9C _H		
インプットキャプチャ 1		# 25	FFFF98 _H	ICR07	0000B7 _H
PPG (ch2, ch3) ユニット 1	×	# 26	FFFF94 _H		
外部割り込み INT6/INT7		# 27	FFFF90 _H	ICR08	0000B8 _H
監視タイマ	×	# 28	FFFF8C _H		
PPG (ch4, ch5) ユニット 2	×	# 29	FFFF88 _H	ICR09	0000B9 _H
インプットキャプチャ 2/3		# 30	FFFF84 _H		
PPG (ch6, ch7) ユニット 3	×	# 31	FFFF80 _H	ICR10	0000BA _H
アウトプットコンペア 0		# 32	FFFF7C _H		
アウトプットコンペア 1		# 33	FFFF78 _H	ICR11	0000BB _H
インプットキャプチャ 4/5		# 34	FFFF74 _H		
アウトプットコンペア 2/3 インプットキャプチャ 6/7		# 35	FFFF70 _H	ICR12	0000BC _H
16 ビットリロードタイマ 1		# 36	FFFF6C _H		
UART 0 RX		# 37	FFFF68 _H	ICR13	0000BD _H
UART 0 TX		# 38	FFFF64 _H		
UART 1 RX		# 39	FFFF60 _H	ICR14	0000BE _H
UART 1 TX		# 40	FFFF5C _H		
フラッシュメモリ	×	# 41	FFFF58 _H	ICR15	0000BF _H
遅延割り込み	×	# 42	FFFF54 _H		

:EI²OS 割り込みクリア信号にて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。

:EI²OS 割り込みクリア信号にて割り込み要求フラグはクリアされます。

× :EI²OS 割り込みクリア信号にて割り込み要求フラグはクリアされません。

< 注意事項 >

同一割込み番号に 2 つの割込み要因がある周辺モジュールは、両方の割込み要求フラグが EI²OS 割込みクリア信号でクリアされます。

EI²OS の終了時には、同一の割込み番号に割り当てられたすべての割込みフラグに対して、EI²OS クリア信号が発行されます。1 つの割込みフラグが EI²OS を開始し、その間にハードウェアイベントによって別の割込みフラグが設定された場合、最初のイベントによって発行された EI²OS クリア信号によってフラグがクリアされるので、後から発生したイベントは失われます。したがって、この割込み番号には EI²OS は使用しないでください。

EI²OS が有効な場合、同一の割込み制御レジスタ (ICR) にある 2 つの割込み信号のうち 1 つが発行されると EI²OS が起動されます。各割込みソースには独自の EI²OS ディスクリプタが用意されるべきですが、実際には異なる割込みソースが同一の EI²OS ディスクリプタを共用します。したがって、1 つの割込みソースが EI²OS を使用している間は、ほかの割込みを無効にしておく必要があります。

3.3 割り込みベクタ

表 3.3-1 に、MB90540/545 シリーズでの割り込みベクター一覧表を示します。

■ 割り込みベクタ

表 3.3-1 割り込みベクター一覧表 (1 / 2)

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 レジスタ	割り込み No	ハードウェア割り込み
INT 0	FFFFFC _H	FFFFFD _H	FFFFFE _H	未使用	#0	なし
:	:	:	:	:	:	:
INT 7	FFFFE0 _H	FFFFE1 _H	FFFFE2 _H	未使用	#7	なし
INT 8	FFFFDC _H	FFFFDD _H	FFFFDE _H	FFFFDF _H	#8	(RESET ベクタ)
INT 9	FFFFD8 _H	FFFFD9 _H	FFFFDA _H	未使用	#9	ROM 訂正
INT 10	FFFFD4 _H	FFFFD5 _H	FFFFD6 _H	未使用	#10	< 例外 >
INT 11	FFFFD0 _H	FFFFD1 _H	FFFFD2 _H	未使用	#11	CAN 0 RX
INT 12	FFFFC _H	FFFFCD _H	FFFFCE _H	未使用	#12	CAN 0 TX/NS
INT 13	FFFFC8 _H	FFFFC9 _H	FFFFCA _H	未使用	#13	CAN 1 RX
INT 14	FFFFC4 _H	FFFFC5 _H	FFFFC6 _H	未使用	#14	CAN 1 TX/NS
INT 15	FFFFC0 _H	FFFFC1 _H	FFFFC2 _H	未使用	#15	外部割り込み INT0/INT1
INT 16	FFFFBC _H	FFFFBD _H	FFFFBE _H	未使用	#16	タイムベースタイマ
INT 17	FFFFB8 _H	FFFFB9 _H	FFFFBA _H	未使用	#17	16 ビットリロードタイマ 0
INT 18	FFFFB4 _H	FFFFB5 _H	FFFFB6 _H	未使用	#18	A/D コンバータ
INT 19	FFFFB0 _H	FFFFB1 _H	FFFFB2 _H	未使用	#19	入出力タイマ
INT 20	FFFFAC _H	FFFFAD _H	FFFFAE _H	未使用	#20	外部割り込み INT2/INT3
INT 21	FFFFA8 _H	FFFFA9 _H	FFFFAA _H	未使用	#21	シリアル I/O
INT 22	FFFFA4 _H	FFFFA5 _H	FFFFA6 _H	未使用	#22	PPG(ch0, ch1) ユニット 0
INT 23	FFFFA0 _H	FFFFA1 _H	FFFFA2 _H	未使用	#23	インプットキャプチャ 0
INT 24	FFFF9C _H	FFFF9D _H	FFFF9E _H	未使用	#24	外部割り込み INT4/INT5
INT 25	FFFF98 _H	FFFF99 _H	FFFF9A _H	未使用	#25	インプットキャプチャ 1
INT 26	FFFF94 _H	FFFF95 _H	FFFF96 _H	未使用	#26	PPG(ch 2, ch3) ユニット 1
INT 27	FFFF90 _H	FFFF91 _H	FFFF92 _H	未使用	#27	外部割り込み INT6/INT7
INT 28	FFFF8C _H	FFFF8D _H	FFFF8E _H	未使用	#28	監視タイマ
INT 29	FFFF88 _H	FFFF89 _H	FFFF8A _H	未使用	#29	PPG(ch4, ch5) ユニット 2
INT 30	FFFF84 _H	FFFF85 _H	FFFF86 _H	未使用	#30	インプットキャプチャ 2/3
INT 31	FFFF80 _H	FFFF81 _H	FFFF82 _H	未使用	#31	PPG(ch6, ch7) ユニット 3
INT 32	FFFF7C _H	FFFF7D _H	FFFF7E _H	未使用	#32	アウトプットコンペア 0
INT 33	FFFF78 _H	FFFF79 _H	FFFF7A _H	未使用	#33	アウトプットコンペア 1
INT 34	FFFF74 _H	FFFF75 _H	FFFF76 _H	未使用	#34	インプットキャプチャ 4/5
INT 35	FFFF70 _H	FFFF71 _H	FFFF72 _H	未使用	#35	アウトプットコンペア 2/3 インプットキャプチャ 6/7
INT 36	FFFF6C _H	FFFF6D _H	FFFF6E _H	未使用	#36	16 ビットリロードタイマ 1
INT 37	FFFF68 _H	FFFF69 _H	FFFF6A _H	未使用	#37	UART 0 RX

第 3 章 割り込み

表 3.3-1 割り込みベクター一覧表 (2 / 2)

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 レジスタ	割り込み No	ハードウェア割り込み
INT 38	FFFF64 _H	FFFF65 _H	FFFF66 _H	未使用	#38	UART 0 TX
INT 39	FFFF60 _H	FFFF61 _H	FFFF62 _H	未使用	#39	UART 1 RX
INT 40	FFFF5C _H	FFFF5D _H	FFFF5E _H	未使用	#40	UART 1 TX
INT 41	FFFF58 _H	FFFF59 _H	FFFF5A _H	未使用	#41	フラッシュメモリ
INT 42	FFFF54 _H	FFFF55 _H	FFFF56 _H	未使用	#42	遅延割り込み
INT 43	FFFF50 _H	FFFF51 _H	FFFF52 _H	未使用	#43	なし
⋮	⋮	⋮	⋮	⋮	⋮	⋮
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

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へ制御を移行する機能です。

■ ハードウェア割込みの概要

ハードウェア割込みの起動は、割込み要求がもつ割込みレベルと、CPU の PS がもつ割込みレベルマスクレジスタ (ILM) の比較および PS 内の I フラグの内容をハードウェアで参照したのち、発生条件に合致すれば発生します。

ハードウェア割込みが発生するときに CPU が行う処理は以下のものがあります。

- CPU 内部の PC, PS, AH, AL, PCB, DTB, ADB, DPR レジスタのシステムスタックへの退避
- PS レジスタ内の ILM を設定。現在要求している割込みレベルと自動的に同じになる
- 対応する割込みベクタの値を取り込み、その値が指示する処理へ分岐

■ 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構造は、以下の 3 つの部分に分かれて存在します。

● 周辺機能

割込み許可ビット、割込み要求ビット：リソースからの割込み要求の制御

● 割込みコントローラ

ICR: 割込みのレベル付け、同時要求割込みの優先度判定

● CPU

I, ILM: 要求割込みレベルと現レベルの比較、割込み許可状態の識別

マイクロコード：割込み処理用ステップ

上記 3 つの区分の状態は、周辺機能ではリソース制御レジスタに、割込みコントローラでは ICR に、CPU では CCR の値によって指示されます。ハードウェア割込みを使用する場合には、あらかじめソフトウェアでこの 3 つの区分を事前に設定します。

割込み処理の際に参照する割込みベクタのテーブルはメモリ領域の "FFFC00_H" ~ "FFFFFF_H" に割り当ててあり、ソフトウェア割込みと共通で使用しています。

■ 入出力領域への書き込み中のハードウェア割込み要求

入出力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。

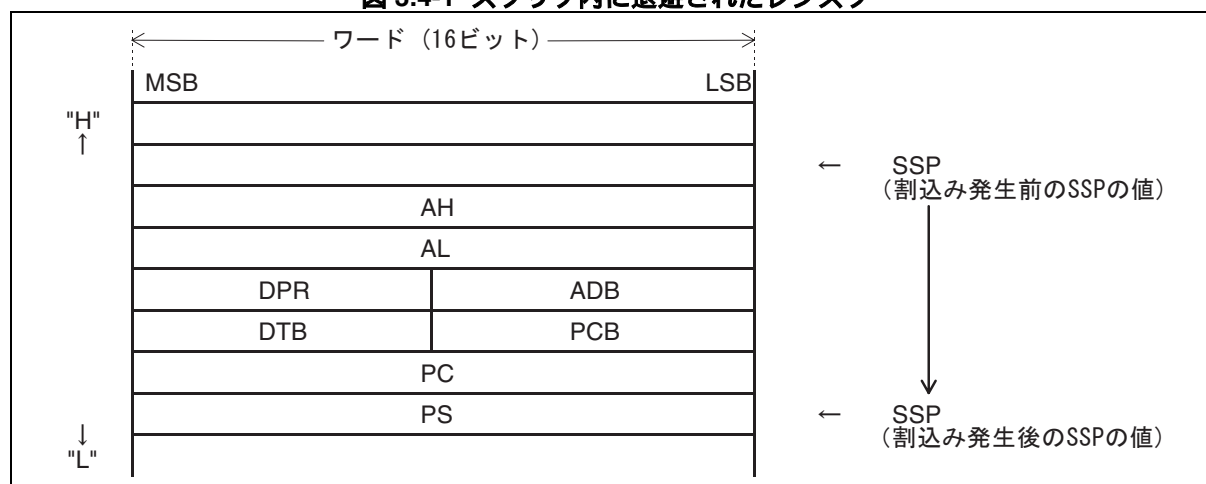
■ 多重割込み

F²MC-16LX CPU は多重割込みをサポートしています。そのため、割込み処理の実行中にその割込みのもつ割込みレベルより高いレベルの割込みが発生すると、現在実行中の命令を終了した後で、制御は高いレベルの割込みへ転送されます。高いレベルでの割込みが終了すると、元の割込み処理に戻ります。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処理が発生した場合は、ILM の内容や I フラグの命令によって変更しないかぎり、現在の割込み処理の終了まで、新しい割込み要求は保留になります。なお、拡張インテリジェント I/O サービスは多重に起動されることはなく、1 つの拡張インテリジェント I/O サービスの処理中は、ほかの割込み要求および拡張インテリジェント I/O サービス要求はすべて保留になります。

■ スタック内へのレジスタの退避

図 3.4-1 に、スタック内に退避されたレジスタの順序を示します。

図 3.4-1 スタック内に退避されたレジスタ



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の機能を備えた内部リソースには、割込み要求フラグと割込み許可フラグが存在します。割込み要求フラグは、割込み要求の有無を指示します。割込み許可フラグは、該当する内部リソースによる CPU への割込み要求の有無を指示します。割込み要求フラグがセットされるのは、内部リソースに特有のイベントが発生する場合です。割込み許可フラグが "許可" を指示すると、リソースは割込みコントローラへ割込み要求を発生します。

■ ハードウェア割込みの動作

同時に複数の割込み要求を受け取った場合、割込みコントローラは、各割込み要求に対する ICR 内の割込みレベル (IL) どうしを比較し、もっとも高いレベルの要求 (IL 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。各割込み要求と各 ICR の関係はハードウェアで決まっています。

CPU は受け取った割込みレベル (IL) と PS レジスタ内の ILM とを比較し、割込みレベル (IL) < (ILM) であり、PS レジスタ内の I ビットが "1" に設定されている場合に、現在実行中の命令が終了した後で、割込み処理用マイクロコードを起動します。割込み処理マイクロコードの先頭で割込みコントローラの ICR 中の ISE ビットを参照し、ISE ビットが "0" (すなわち割込み) であることを確認して割込み処理の本体を起動します。

割込み処理の本体では SSB と SSP の示すメモリ中へ PS と PC, PCB, DTB, ADB, DPR, A の 12 バイトを退避した後、割込みベクタの中から、3 バイトを取り出して PC と PCB へロードします。PS 中の ILM を、受け付けた割込み要求のレベル値に更新、S フラグを "1" に設定して分岐処理を行います。その結果、次に実行する命令はユーザの定義した割込み処理プログラムになります。

ハードウェア割込みが発生してから、割込み処理プログラム内に割込み要求がなくなるまでの処理の流れを図 3.4-2 に示します。

図 3.4-2 ハードウェア割込みの発生と解除まで

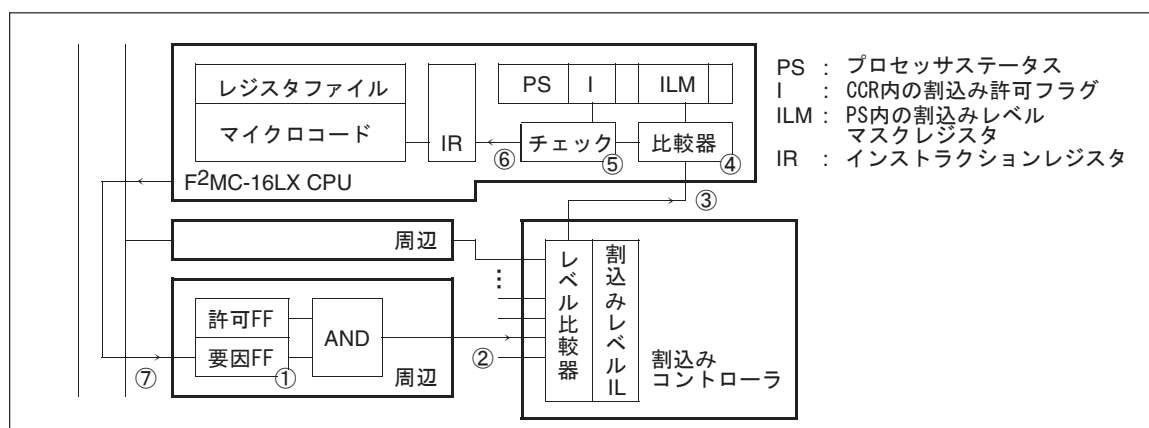


図 3.4-2 の図中における、 ～ の内容を以下に説明します。

周辺の内部で割込み要因が発生します。

周辺内の割込み許可ビットを参照し割込み許可になっていれば、周辺から割込みコントローラへ割込み要求が発生します。

割込み要求を受け取った割込みコントローラは、同時に要求のあった割込みの優先順位を判定した上で該当する割込みに対応する割込みレベルを CPU へ転送します。

CPU は割込みコントローラから要求のあった割込みレベルをプロセッサステータスレジスタ内の ILM ビットと比較します。

比較の結果が現在の割込み処理レベルより優先順位が高い場合のみ、同じプロセッサステータスレジスタ内の I フラグの内容をチェックします。

のチェックの結果 I フラグが割込み許可状態である場合のみ、ILM ビットの内容を要求されたレベルに設定し、現在実行中の命令の実行が終了し次第割込み処理を行い、制御を割込み処理ルーチンへ移します。

ユーザの割込み処理ルーチン内のソフトウェアで で発生した割込み要因をクリアすることで割込み要求が終了します。

および で、CPU が行う割込み処理の実行時間を以下に示します。

割込み起動 :24+6 × 表 3.4-1 マシンサイクル

割込み復帰 :15+6 × 表 3.4-1 マシンサイクル (RETI 命令)

表 3.4-1 割込み処理時のサイクル数の補正值

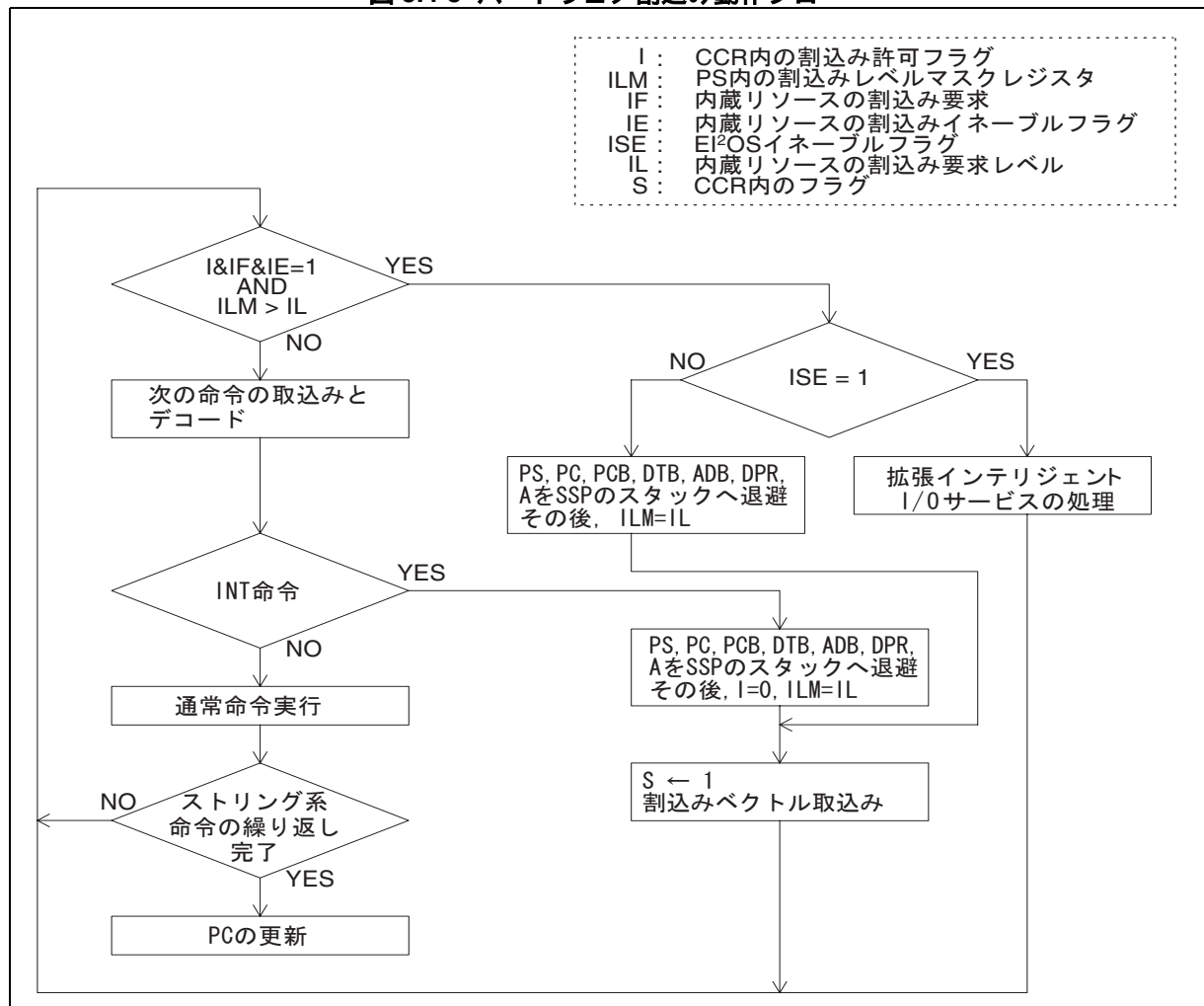
スタックポインタが示しているアドレス	サイクル数の補正值
外部領域 8 ビットデータバスの場合	+4
外部領域 偶数アドレスの場合	+1
外部領域 奇数アドレスの場合	+4
内部領域 偶数アドレスの場合	0
内部領域 奇数アドレスの場合	+2

3.4.2 ハードウェア割り込みの動作フロー

図 3.4-3 に、ハードウェア割り込みの動作フローを示します。

■ ハードウェア割り込みの動作フロー

図 3.4-3 ハードウェア割り込み動作フロー



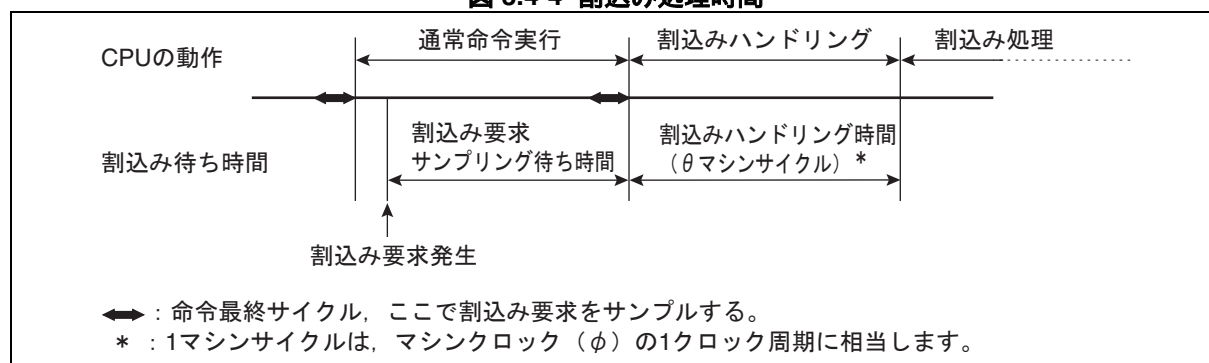
3.4.3 割り込み処理の開始に要する時間

ハードウェア割り込み要求が発生して割り込み処理が実行されるまでの間には、現在実行中の命令が終了するまでの時間と割り込みハンドリング時間が必要です。

■ 割り込み処理の開始に要する時間

割り込み要求が発生してから割り込みが受け付けられて、割り込み処理が実行されるまでの間には、割り込み要求サンプリング待ち時間と割り込みハンドリング時間（割り込み処理準備に要する時間）が必要です。割り込み処理時間を図 3.4-4 に示します。

図 3.4-4 割り込み処理時間



● 割り込み要求サンプリング待ち時間

割り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です。

割り込み要求の発生の有無は、各命令の最後のサイクルで割り込み要求をサンプリングして判断されます。各命令の実行中は CPU は割り込み要求を認識することができず、待ち時間が発生します。

< 参考 >

割り込み要求サンプリング待ち時間は、最も実行サイクルの長い POPW RW0, ...RW7 命令 (45 サイクル) 開始直後に割り込み要求が発生した場合、最大となります。

● 割り込みハンドリング時間 (マシンサイクル)

CPU は、割り込み要求を受け付けてから、専用レジスタのシステムスタックへの退避、および割り込みベクタテーブルアドレスの取込みなどを行うため、マシンサイクルの割り込みハンドリング時間を必要とします。割り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。

$$= 24 + 6 \times Z \text{ マシンサイクル (Z: 割り込みハンドリング時間の補正值)}$$

割り込みハンドリング時間は、スタックポインタに設定されているアドレスによって異なります。割り込みハンドリング時間の補正值 (Z) を表 3.4-2 に示します。

表 3.4-2 割込みハンドリング時間の補正值 (Z)

スタックポインタで設定されているアドレス	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

< 参考 >

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

3.5 ソフトウェア割込み

ソフトウェア割込みは、専用の命令の実行に対応して、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行する機能です。

■ ソフトウェア割込みの概要

ソフトウェア割込みの起動は、ソフトウェア割込み命令の実行で常に発生します。ソフトウェア割込みが発生するときに CPU が行う処理は、以下のものがあります。

- CPU 内部の PC, PS, AH, AL, PCB, DTB, ADB, DPR レジスタの値をシステムスタックへ退避させます。
- PS:I フラグを "0" に設定します。割込みは、自動的に禁止されます。
- 対応する割込みベクタの値を取り出し、次に値が指示する処理へ分岐します。

INT 命令から出されたソフトウェア割込み要求には、割込み要求フラグや許可フラグは存在しません。ソフトウェア割込み要求が出されるのは、必ず INT 命令の実行によります。

INT 命令には割込みレベルが存在しません。したがって、INT 命令は、ILM を更新しません。INT 命令は、I フラグをクリアして後続の割込み要求を保留にします。

■ ソフトウェア割込みの構造

ソフトウェア割込みは、すべて CPU の内部で取り扱われます。

● CPU

- マイクロコード：割込み処理用ステップ

表 3.3-1 に示すように、ソフトウェア割込みは、ハードウェア割込みと同じ割込みベクタ領域を共用します。例えば、割込み要求番号の INT 15 は、ハードウェア割込みの外部割込み #0 に使用されると同時に、ソフトウェア割込みの INT #15 に使用されます。したがって、外部割込み #0 と INT #15 は、同じ割込み処理ルーチンを呼び出します。

■ ソフトウェア割込みの動作

CPU がソフトウェア割込み命令を取り込み実行すると、ソフトウェア割込み処理用マイクロコードを起動します。ソフトウェア割込み処理用マイクロコードは SSB と SSP が指示するメモリ領域に 12 バイト (PS と PC, PCB, DTB, ADB, DPR, A) を退避させます。次に、マイクロコードは割込みベクタの中から、3 バイトを取り出して PC と PCB へ格納します。I フラグをリセット (0) して、S フラグをセット (1) します。その結果、ユーザアプリケーションプログラムが定義した割込み処理プログラムを次に実行します。

ソフトウェア割込みが発生してから、割込み処理プログラム内に割込み要求がなくなるまでの処理の流れを図 3.5-1 に示します。

図 3.5-1 ソフトウェア割込みの発生と解除まで

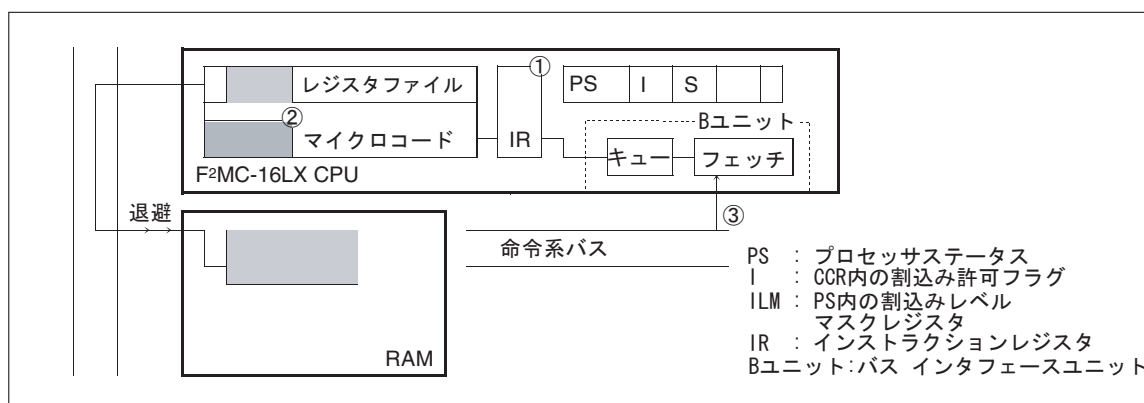


図 3.5-1 の図中における、①～③のソフトウェア割込みの発生と解除までの流れを以下に示します。

ソフトウェア割込み命令を実行します。

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レジスタファイル中の CPU 内専用レジスタが退避されます。

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" の場合、CALLV 命令のベクタ領域は INT #vct8 命令のテーブルと重複します。ソフトウェアを設計する際は、必ず CALLV 命令が INT #vct8 命令と同じアドレスを使用しないよう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)

拡張インテリジェント I/O サービス (EI²OS) とは、ハードウェア割込み動作の一種で、I/O とメモリとの間でデータを自動的に転送します。従来は、割込み処理プログラムにより、I/O とメモリの間でデータを転送していましたが、EI²OS では、DMA モードのようなデータの転送が可能になります。

■ 拡張インテリジェント I/O サービス (EI²OS) の概要

拡張インテリジェント I/O サービス (EI²OS) は、従来の割込み処理で行っていた方式と比べると以下に示すような利点があります。

- 転送プログラムの作成が不要となるため、プログラム全体のサイズを小さくすることができます。
- 転送の際に内部レジスタを使用しないので、レジスタの退避が不要になり転送速度が速くなります。
- I/O によりいつでも転送を停止することができ、不要なデータを転送しません。
- バッファアドレスのインクリメント、デクリメントまたは未更新が選択できます。
- I/O レジスタアドレスのインクリメント、デクリメントまたは未更新が選択できます (バッファアドレスを更新する場合)。

EI²OS 処理の終了時には、終了条件を設定した後に、CPU が自動的に割込み処理ルーチンへ分岐します。したがって、ユーザは終了条件の種類を判別することができます。

< 注意事項 >

REALOSを使用する場合は、拡張インテリジェントI/Oサービス(EI²OS)が使用できません。

EI²OS の概要を図 3.6-1 に示します。

図 3.6-1 拡張インテリジェント I/O サービスの概要

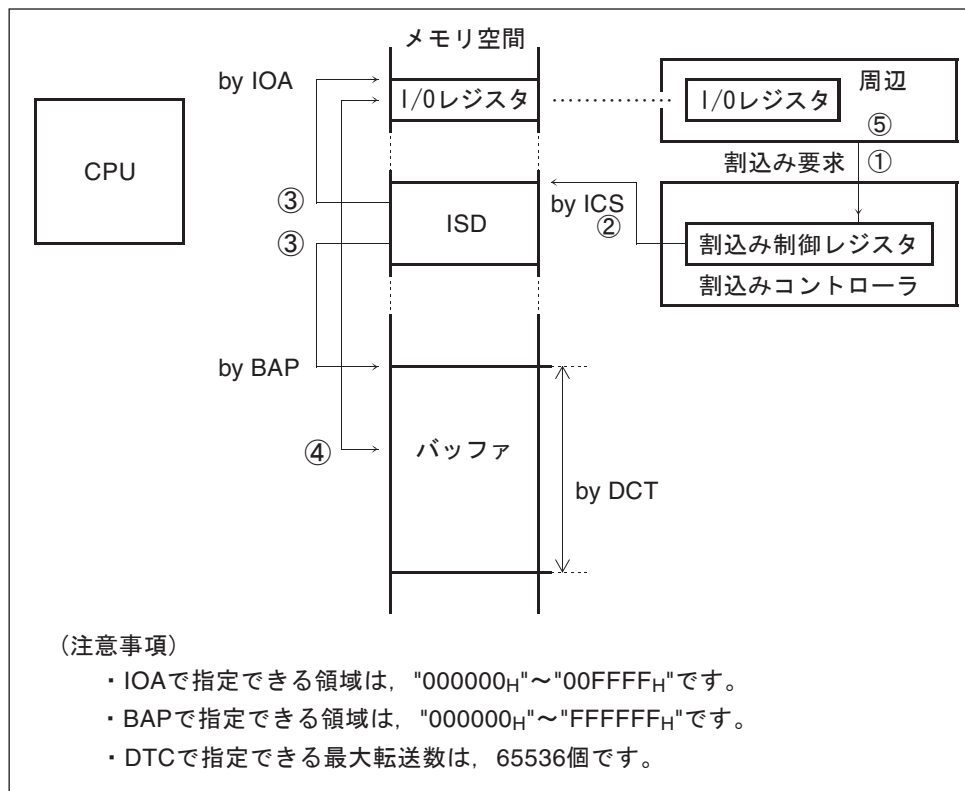


図 3.6-1 中の ① ~ ⑤ の説明を以下に示します。

I/O が転送を要求します。

割り込みコントローラがディスクリプタを選択します。

転送元 / 転送先をディスクリプタから読み出します。

I/O とメモリ間で転送が行われます。

割り込み要因は自動的にクリアされます。

■ 拡張インテリジェント I/O サービス (EI²OS) の構造

EI²OS に関連する機構は, 以下の 4 つの部分に分かれて存在します。

● 内蔵リソース

割り込み許可ビット, 割り込み要求ビット: リソースからの割り込み要求の制御

● 割り込みコントローラ

ICR: 割り込みのレベル付け, 同時要求割り込みの優先度判定, EI²OS 動作の選択

● CPU

- ・ I, ILM: 要求割り込みレベルと現レベルの比較, 割り込み許可状態の識別
- ・ マイクロコード: EI²OS 処理用ステップ

● RAM

ディスクリプタ: EI²OS の転送情報を記述する

3.6.1 割込み制御レジスタ (ICR)

割込み制御レジスタは割込みコントローラ内にあり、割込み機能を持つすべての I/O に対応して存在します。このレジスタには以下に示す 3 つの機能があります。

- 対応する周辺の割込みレベルの設定
- 対応する周辺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 サービスにするかの選択
- 拡張インテリジェント I/O サービスのチャンネルの選択

このレジスタに対するリードモディファイライト系の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。

■ 割込み制御レジスタ (ICR)

割込み制御レジスタ (ICR) のビット構成を図 3.6-2 に示します。

図 3.6-2 割込み制御レジスタ (ICR)

	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0	ビット No.
アドレス： 0000B0 _H ~ 0000BF _H	ICS3	ICS2	ICS1	ICS0	ISE	IL2	IL1	IL0	書き込み時
リード / ライト	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
初期値	(0)	(0)	(0)	(0)	(0)	(1)	(1)	(1)	
	15/7	14/6	13/5	12/4	11/3	10/2	9/1	8/0	ビット No.
アドレス： 0000B0 _H ~ 0000BF _H			S1	S0	ISE	IL2	IL1	IL0	読出し時
リード / ライト	(-)	(-)	(R)	(R)	(R)	(R)	(R)	(R)	
初期値	(-)	(-)	(0)	(0)	(0)	(1)	(1)	(1)	

< 注意事項 >

ICS3 ~ ICS0 は、EI²OS を起動する場合のみ有効となります。EI²OS を起動する場合は ISE を "1" に設定し、起動しない場合は "0" に設定してください。EI²OS を起動しない場合、ICS3 ~ ICS0 は何を設定してもかまいません。

ICS3 と ICS2 には、常に "1" が読み込まれます。

ICS1 と ICS0 は書き込みのみ有効で、S1 と S0 は読出しのみ有効です。

【bit 15 ~ bit 12, bit 7 ~ bit 4】ICS3 ~ ICS0

ICS3 ~ ICS0 ビットは、EI²OS チャンネル選択ビットです。

書き込み専用のビットです。ここで設定された値によりメモリ上の拡張インテリジェント I/O サービスディスクリプタのアドレスが決定します。ICS はリセットにより初期化されます。

ICS ビット、チャンネル番号およびディスクリプタアドレスの対応を表 3.6-1 に示します。

表 3.6-1 ICS ビット, チャネル番号およびディスクリプタアドレス

ICS3	ICS2	ICS1	ICS0	セレクトされるチャネル	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

【bit 13, bit 12, bit 5, bit 4】S0, S1

S0, S1 は, EI²OS 終了ステータスビットです。

読出し専用のビットで, EI²OS 終了時にこのビットの値を調べることで, 終了条件が何であったかを判別することができます。リセットにより "00_B" に初期化されます。

S ビットと終了条件との関係を表 3.6-2 に示します。

表 3.6-2 S ビットと終了条件

S1	S0	終了条件
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	内蔵リソースからの要求による停止状態

【bit 11, bit 3】ISE

ISE ビットは, EI²OS を許可します。読み書き可能なビットです。

割込み要求の発生時にこのビットが "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 の終了条件が満たされると (S1 ビットと S0 ビットが "00_B" でない), ISE ビットは "0" になります。対応する周辺に EI²OS の機能がないときには, ソフトウェアで ISE を "0" にしておく必要があります。リセットにより "0" に初期化されます。

第 3 章 割込み

【bit 10 ~ bit 8, bit 2 ~ bit 0】IL0, IL1, IL2

IL0, IL1, IL2 ビットは、割込みレベルを設定します。

対応する内蔵リソースの割込みレベルを指定します。読出しと書込みができます。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割込みレベルとの関係を表 3.6-3 に示します。

表 3.6-3 割込みレベル設定ビットのレベル値

ILM2	ILM1	ILM0	割込みレベル値
0	0	0	0(最強割込み)
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6(最弱割込み)
1	1	1	7(割込みなし)

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

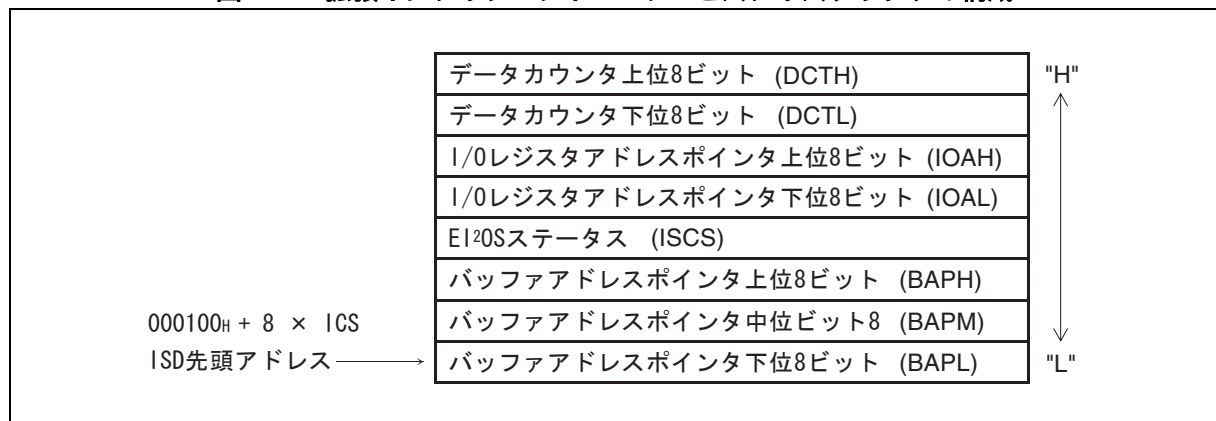
拡張インテリジェント I/O サービスディスクリプタは、内部 RAM 中の "000100_H" ~ "00017F_H" に存在し、以下に示すものから構成されます。

- データ転送の各種制御データ
- ステータスデータ
- バッファアドレスポインタ

■ 拡張インテリジェント I/O サービスディスクリプタ (ISD)

図 3.6-3 に、拡張インテリジェント I/O サービスディスクリプタの構成を示します。

図 3.6-3 拡張インテリジェント I/O サービスディスクリプタの構成



■ 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、転送されたデータ要素の数に対応したカウンタとして機能します。データ転送の前に、このカウンタは1個デクリメントされます。このカウンタがゼロになると EI²OS は終了します。

データカウンタ (DCT) のビット構成を図 3.6-4 に示します。

図 3.6-4 データカウンタ (DCT) の構成

データカウンタ上位									
	15	14	13	12	11	10	9	8	ビット No.
	B15	B14	B13	B12	B11	B10	B09	B08	DCTH
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
データカウンタ下位									
	7	6	5	4	3	2	1	0	ビット No.
	B07	B06	B05	B04	B03	B02	B01	B00	DCTL
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

■ 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、データ転送に使用されるバッファと 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" ~ "00FFFF_H" 番地までの任意の I/O を指定できます。I/O レジスタアドレスポインタ (IOA) のビット構成を図 3.6-5 に示します。

図 3.6-5 I/O レジスタアドレスポインタ (IOA) の構成

I/O レジスタアドレスポインタ上位									ビット No.
	15	14	13	12	11	10	9	8	
	A15	A14	A13	A12	A11	A10	A09	A08	IOAH
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
I/O レジスタアドレスポインタ下位									ビット No.
	7	6	5	4	3	2	1	0	
	A07	A06	A05	A04	A03	A02	A01	A00	IOAL
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

■ EI²OS ステータスレジスタ (ISCS)

EI²OS ステータスレジスタ (ISCS) は、8 ビット長のレジスタで、更新の向き (インクリメント / デクリメント)、転送データ形式 (バイト / ワード)、バッファアドレスポインタおよび I/O レジスタアドレスポインタの転送方向を指示します。また、このレジスタは、バッファアドレスポインタまたは I/O レジスタアドレスポインタが更新または固定されているかどうかを指示します。EI²OS ステータスレジスタ (ISCS) のビット構成を図 3.6-6 に示します。

図 3.6-6 EI²OS ステータスレジスタ (ISCS) の構成

	7	6	5	4	3	2	1	0	ビット No.
	予約	予約	予約	IF	BW	BF	DIR	SE	
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

各ビットについて、以下に説明します。

【bit 4】IF

IF ビットは、I/O レジスタアドレスポインタの更新 / 固定を指定します。

表 3.6-4 I/O レジスタアドレスポインタの更新 / 固定ビット (IF)

IF	機能
0	データ転送後 I/O レジスタアドレスポインタは更新される。
1	データ転送後 I/O レジスタアドレスポインタは更新されない。

【bit 3】BW

BW ビットは、転送データ長を指定します。

表 3.6-5 転送データ長を指定ビット (BW)

BW	機能
0	バイト
1	ワード

【bit 2】BF

BF ビットは、バッファアドレスポインタの更新 / 固定を指定します。

表 3.6-6 バッファアドレスポインタの更新 / 固定指定ビット (BF)

BF	機能
0	データ転送後バッファアドレスポインタは更新される。
1	データ転送後バッファアドレスポインタは更新されない。

< 注意事項 >

更新する場合、バッファアドレスポインタは下位 16 ビットのみ変化します。許可されるのはインクリメントのみです。

【bit 1】:DIR

DIR ビットは、データの転送方向を指定します。

表 3.6-7 データの転送方向指定ビット (DIR) の設定

DIR	設定
0	I/O バッファ
1	バッファ I/O

【bit 0】SE

SE ビットは、リソースからの要求に応じて、拡張インテリジェント I/O サービスの終了を制御します。

表 3.6-8 EI²OS 終了制御ビット

SE	設定
0	リソースからの要求により終了しない。
1	リソースからの要求により終了する。

■ バッファアドレスポインタ (BAP)

24 ビットのレジスタで次に EI²OS による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しますので、EI²OS の各チャネルは 16 M バイトの任意の空間を転送することができます。

< 注意事項 >

ISCS の BF ビットを "0" (更新あり) にした場合、BAP は下位 16 ビットのみ変化し、BAPH は変化しません。

3.6.3 拡張インテリジェント I/O サービス (EI²OS) の動作

図 3.6-7 に拡張インテリジェント I/O サービス (EI²OS) の動作フローを、図 3.6-8 に拡張インテリジェント I/O サービス (EI²OS) の使用手順フローを示します。

■ 拡張インテリジェント I/O サービス (EI²OS) の動作フロー

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

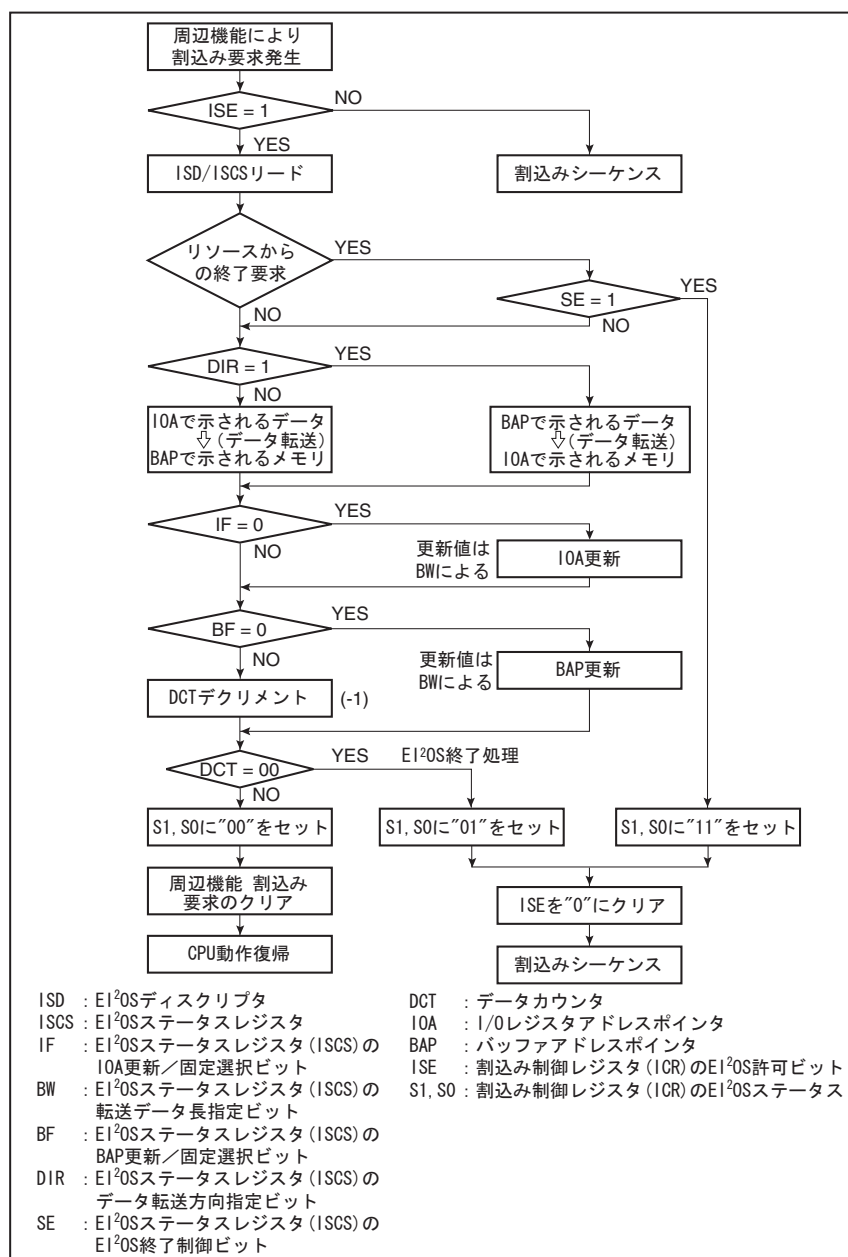
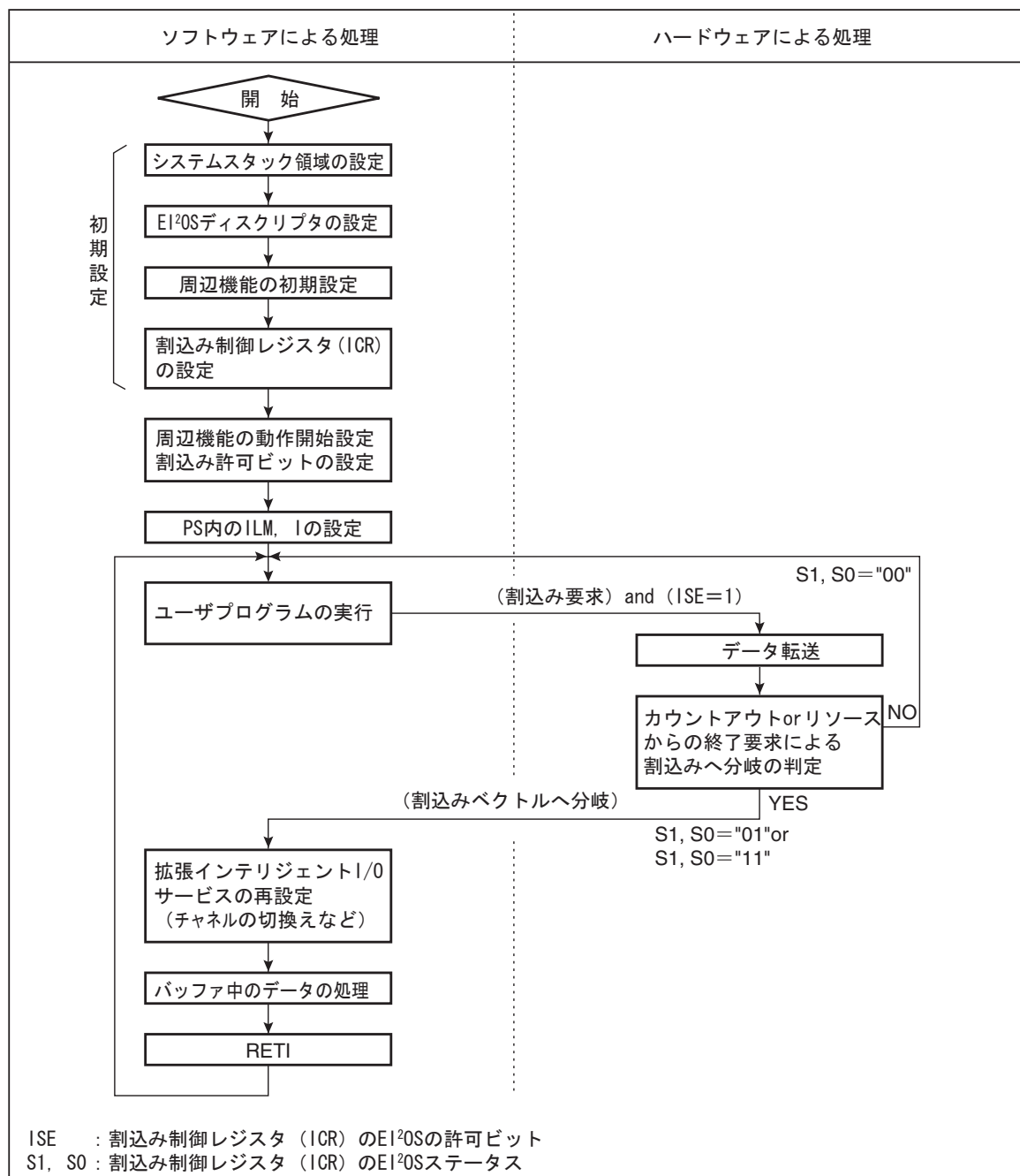


図 3.6-8 拡張インテリジェント I/O サービス (EI²OS) の使用手順フロー

3.6.4 拡張インテリジェント I/O サービス (EI²OS) の実行時間

拡張インテリジェント I/O サービス (EI²OS) の実行時間は次の 3 つの場合があります。

- データ転送継続時 (停止条件が成立しない場合)
- リソースからの停止要求時
- カウント終了時

■ 拡張インテリジェント I/O サービス (EI²OS) の実行時間

- データ転送継続時 (停止条件が成立しない場合)

(表 3.6-9 + 表 3.6-10) マシンサイクル

表 3.6-9 拡張 EI²OS の継続時の実行時間

ISCS・SE ビット		"0" に設定		"1" に設定	
I/O レジスタアドレスポインタ		固定	更新	固定	更新
バッファアドレスポインタ	固定	32	34	33	35
	更新	34	36	35	37

- リソースからの停止要求時

(36+6 × 表 3.4-1) マシンサイクル

- カウント終了時

(表 3.6-9 + 表 3.6-10 +(21+6 × 表 3.4-1)) マシンサイクル

表 3.6-10 拡張 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

3.7 未定義命令の実行による例外発生

F²MC-16LX では、未定義命令の実行により例外が発生し、例外処理が行われます。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のみに使用することをお勧めします。

■ 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、ソフトウェア割り込み命令の「INT 10」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、割り込み番号 10 のベクタで示されるルーチンへ分岐します。さらに、I フラグをクリアし、S フラグを設定します。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。RETI 命令で元の処理に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。

第4章

クロックとリセット

クロックとリセットの機能と動作について説明します。

4.1 クロック発生部

4.2 リセット要因の発生

4.3 リセット要因

4.1 クロック発生部

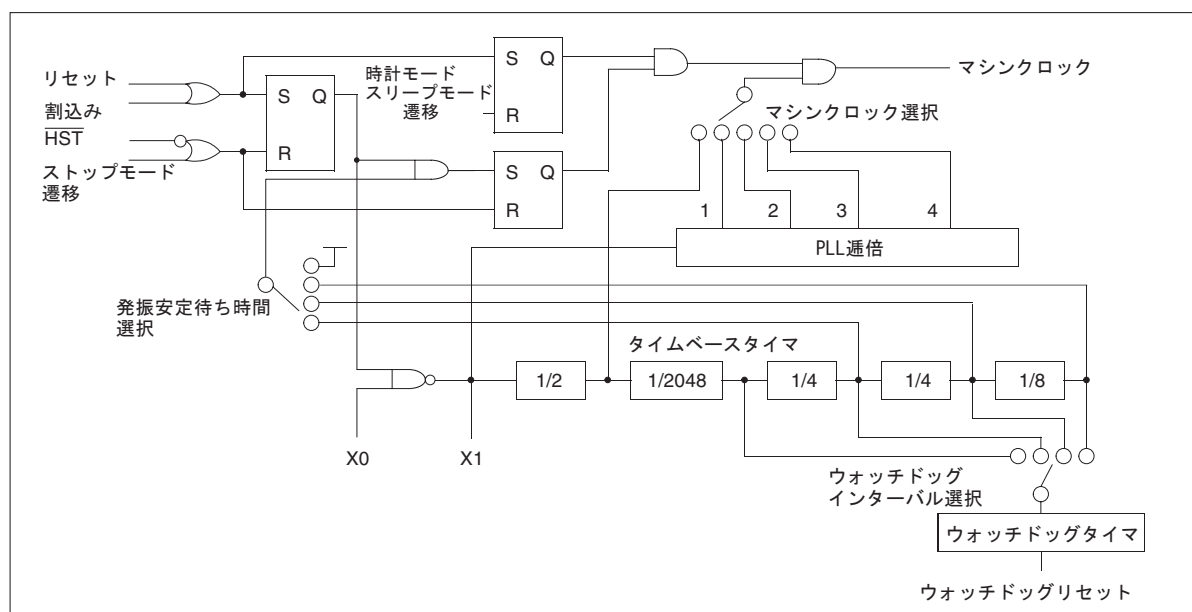
クロック発生部により，スリープ，時計，ストップおよびPLL クロック逡倍機能など内部クロックの動作制御を行います。この内部クロックのことをマシナクロックとよび，その1周期をマシナサイクルとしています。また，原発振によるクロックをメインクロックとよび，内部のVCO 発振によるクロックをPLL クロックとしています。

■ クロック発生部の注意

動作電圧が5V のとき，3 MHz ～ 16 MHz の発振が可能です。しかし，CPU および周辺リソース回路の最大動作周波数が16 MHz なので，16 MHz を超える逡倍係数を指定した場合には，正常な動作は保証されません。例えば，外部クロックが16 MHz で発振している場合，逡倍係数として"1"しか指定できません。

また，VCO 発振の最低動作周波数は4MHz であり，この周波数以下の発振は指定することができません。

図 4.1-1 クロック発生回路ブロックダイアグラム



4.2 リセット要因の発生

リセット要因が発生すると、F²MC-16LX は直ちに現在実行中の処理を中断し、リセット解除待ち状態になります。

■ リセット要因の発生

リセットは以下の5種類の要因で発生します。

- パワーオンリセットの発生
- ハードウェアスタンバイ状態の解除
- ウォッチドッグタイマのオーバフロー
- $\overline{\text{RST}}$ 端子による外部リセット要求の発生
- ソフトウェアによるリセット要求の発生

ストップモードまたはパワーオンリセットから出ると、動作を再開する前に発振安定間隔が挿入されます。

リセット要因が発生すると、F²MC-16LX はすぐに現在実行されている動作を停止し、リセット解除スタンバイモードに移ります。

マシクロックとウォッチドッグ機能はリセット要因に合わせて初期化されます。

ウォッチドッグタイマ制御レジスタのリセット要因ビットでは、リセット要因を識別できます。

< 注意事項 >

ストップモード以外のモードでは、外部リセット入力クロック回路により内部的にサンプルされますので、外部クロック信号供給が使用されるときにリセット信号入力は受信されません。

外部バスが使用されるとき、リセット要因が発生するデバイスによって未定義のアドレスが生成されます。 $\overline{\text{RD}}$ 、 $\overline{\text{WR}}$ 信号などの外部バスアクセスに使用される信号は、アクティブではありません。

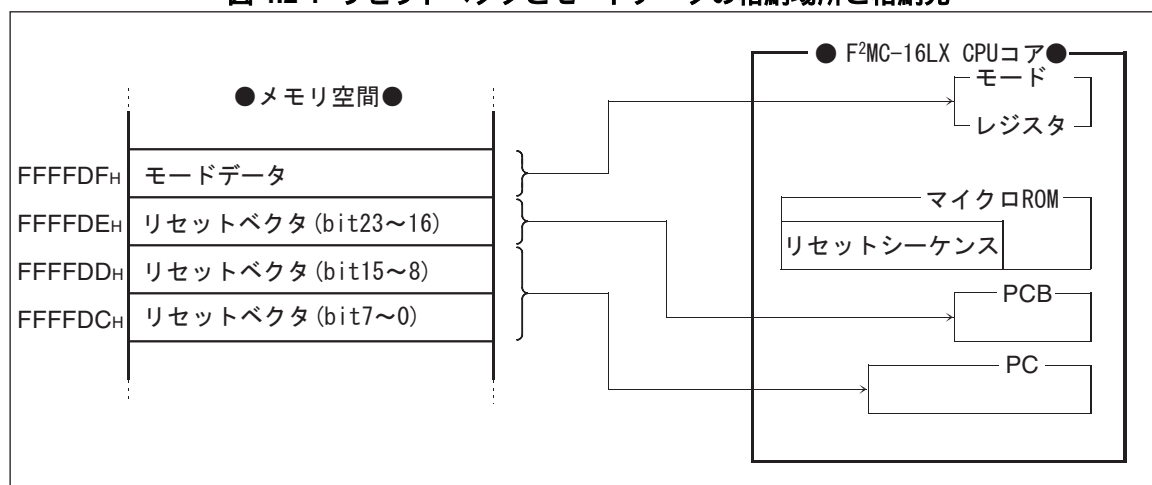
■ リセット解除後の動作

リセットが取り除かれると、F²MC-16LX は直ちにリセットベクタを格納してあるアドレスを出力し、リセットベクタとモードデータの取込みを行います。リセットベクタとモードデータは "FFFFDC_H" ~ "FFFFDF_H" の4バイトに割り当てられています。リセットベクタとモードデータは、図 4.2-1 で示すように、リセット解除後にレジスタにハードウェアで転送されます。

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を指定するのは、モード端子により行います。モード端子で外部ベクタモードを指定すると、外部メモリにリセットベクタとモードデータを読み出しにいきますので、シングルチップモードおよび内 ROM 外バスモードで使用される場合には、内部ベクタモードを指定することを推奨します。

リセットベクタとモードデータを読み出した後のバスモードは、モードデータにより指定します。

図 4.2-1 リセットベクタとモードデータの格納場所と格納先



< 注意事項 >

上記ダイアグラムのモードレジスタは、リセット直後は定義されていません。モードデータをメモリ空間に格納して値がこの領域に書き込まれるようにしてください。

■ リセット入力によって初期化されないレジスタ

本製品には、パワーオンリセットでしか初期化されないレジスタがあります。表 4.2-1 に各リセット要因によって初期化されないレジスタを示します。

表 4.2-1 リセット入力によって初期化されないレジスタ

リセットの種類		CKSCR					WTC	LPMCR	
		WS1	WS0	MCS	CS1	CS0	WDSCS	CG1	CG0
RST のみ使用 ソフトウェアリセット		×	×	×	×	×	×	×	×
ウォッチドッグリセット		×	×		×	×	×		
パワーオンリセット									
ハードウェアスタンバイ (G サフィックスなしの製品)		×	×		×	×	×		
ハードウェアスタンバイ (G サフィックス付きの製品)	メイン モード	×	×		×	×	×		
	サブ モード								

WS1, WS0: メインクロック発振安定待ち時間設定

MCS: マシンクロック ("0"=PLL クロック, "1"= メインクロック)

CS1, CS0: PLL クロック逡倍設定

WDSCS: ウォッチドッグ入力クロック "0"= 時計タイマ, "1" = タイムベースタイマ

: 初期化する

×: 初期化しない

MCS ビットについては、マシンのクロック設定ビットですので、特に注意が必要です。例えば、電源投入がパワーオンリセット規格を満足しないときは、パワーオンリセットが発生しません。そのため、MCS が初期化されないことにより、内部動作周波数が動作保証範囲外になり、正常動作しなくなることが考えられます。

また、何らかの原因により CPU が暴走し、MCS、CS1、CS0 が書き換わった際にも内部動作周波数が動作保証範囲外になり、その状態から $\overline{\text{RST}}$ 入力のみでは正常復帰しなくなることが考えられます（ただし、内部ウォッチドッグが発生した場合には、MCS が初期化され正常動作します）。

上記のようなケースが考えられる場合には、 $\overline{\text{HST}}+\overline{\text{RST}}$ ($\overline{\text{HST}}$ と $\overline{\text{RST}}$ をショート) として使用することを推奨します。

表 4.2-2 に、 $\overline{\text{HST}}+\overline{\text{RST}}$ を行った場合のリセット入力による初期化されないレジスタについて示します。表 4.2-2 に示すように、リセット解除時の動作状態が、 $\overline{\text{HST}}+\overline{\text{RST}}$ のリセット入力と、 $\overline{\text{RST}}$ のみの入力とで異なるので、注意が必要です。

表 4.2-2 リセット入力によって初期化されないレジスタ

リセットの種類		CKSCR					WTC	LPMCR	
		WS1	WS0	MCS	CS1	CS0	WDCS	CG1	CG0
$\overline{\text{HST}}+\overline{\text{RST}}$ (G サフィックスなしの製品)		×	×		×	×	×		
$\overline{\text{HST}}+\overline{\text{RST}}$ (G サフィックス付きの製品)	メインモード	×	×		×	×	×		
	サブモード*								

：初期化する

×：初期化しない

*：サブモード移行期間中を含みます。

図 4.2-2 リセット入力による動作遷移

[リセット入力による動作遷移]



A. 発振状態

	発振		状態	
RSTのみ使用 (HST="H")	メイン		発振	
	サブ		発振	
HST+RSTを 使用	メイン	発振	停止	メイン発振 安定待ち
	サブ			サブ発振安定待ち
				メインラン実行可能
				サブラン実行可能

B. 命令実行タイミング("L"停止, "H"開始)

- ・ RSTのみ使用 (HST="H")

- ・ HST+RSTを使用

メインモード

サブモード

- ・ パワーオンリセット時

V_{CC}(電源)

	発振		状態	
パワーオン リセット	メイン	発振	停止	メイン発振 安定待ち
	サブ			サブ発振安定待ち
				メインラン実行可能
				サブラン実行可能

メインモード

サブモード

サブ発振 (32kHz) の2¹⁶カウント=約2s

4.3 リセット要因

表 4.3-1 に、5 種類のリセット要因を示します。リセット要因によりマシナクロックとウォッチドッグ機能の初期化状態が異なります。
リセット要因レジスタはリセット要因を表示します。

■ リセット要因

表 4.3-1 リセット要因

リセット	発生要因	マシナクロック		ウォッチドッグ タイマ	発振安定 待ち
		サブクロック時	PLL クロック時		
パワーオン	電源立上げ時	メインクロック *	メインクロック *	停止	あり
ハードウェア スタンバイ	$\overline{\text{HST}}$ 端子への "L" レベル入力	メインクロック *	メインクロック *	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 *	メインクロック *	停止	あり
外部端子	$\overline{\text{RST}}$ 端子への "L" レベル入力	メインクロック * または PLL クロック	PLL クロック	前の状態を保持	なし
ソフトウェア	LPMCR レジスタ中の RST ビットに "0" をライト	メインクロック * または PLL クロック	PLL クロック	前の状態を保持	なし

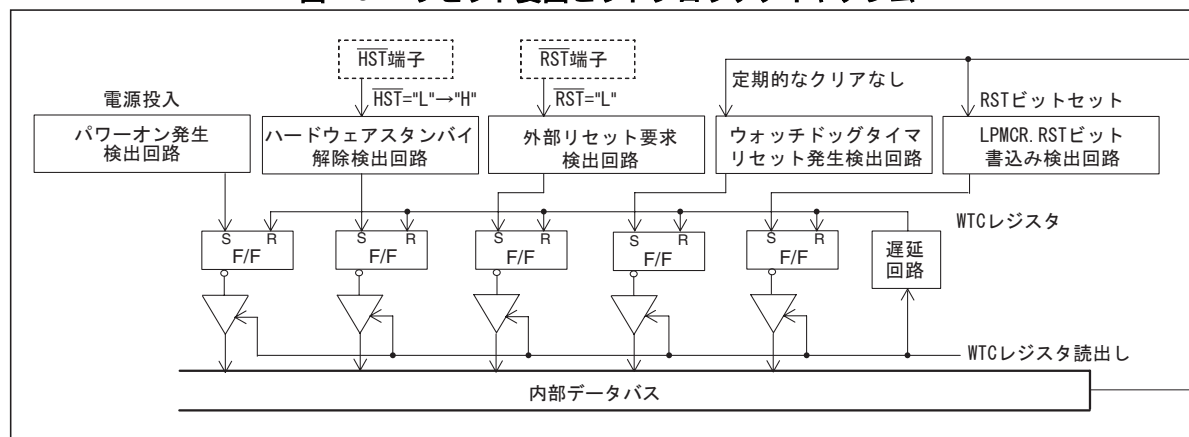
※:fosc/2(fosc: 原発振)

< 注意事項 >

- ストップモードで外部端子リセット信号が入力されると、リセット要因に関係なく発振安定待ち時間を確保します。
- パワーオンリセットの発振安定待ち時間は、原発振 2^{18} サイクル固定です。また、サブクロックモード時のハードウェアスタンバイリセットの発振安定待ち時間は、原発振 2^{17} サイクル固定です。また、サブクロックモード時のハードウェアスタンバイリセットの発振安定待ち時間は、原発振 2^{17} サイクル固定です。それ以外の発振安定待ち時間は、クロック選択レジスタの WS1/WS0 により決定されます。

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を読み出すことで得ることができますので、リセット解除後にリセット発生要因を識別する必要がある場合には、ウォッチドッグタイマ制御レジスタを読み出した値をソフトウェアで処理した上で、適切なアプリケーションプログラムへ分岐するようにしてください。

図 4.3-1 リセット要因ビットブロックダイヤグラム



リセット要因が複数発生する場合でも、ウォッチドッグタイマ制御レジスタの対応するそれぞれのリセット要因ビットを立てるようになっていきます。したがって、外部リセット要求とウォッチドッグリセットが同時に発生した場合でもERSTビットとWRSTビットの両方が"1"になります。

ただし、パワーオンリセットの場合だけは別で、PONR ビットが "1" であるときはそれ以外のビットの内容は正常なりセット要因を示しておりません。このため、PONR ビットが "1" の場合はそれ以外のリセット要因ビットの内容は無視するようにソフトウェアを作成してください。

表 4.3-2 リセット要因ビット値とリセット要因

リセット要因	PONR	STBR	WRST	ERST	SRST
パワーオン	1	-	-	-	-
ハードウェアスタンバイ	*	1	*	*	*
ウォッチドッグタイマ	*	*	1	*	*
外部端子	*	*	*	1	*
RST ビット	*	*	*	*	1

*: 前の値を保持

リセット要因ビットのクリアはウォッチドッグタイマ制御レジスタの読出しだけです
ので、一度発生したリセット要因に対応するリセット要因ビットは、それ以外のリセッ
ト要因が発生しても "1" になったままとなっています。

ウォッチドッグタイマ制御レジスタの構成とリセット要因ビットについての詳細は、「第9章 タイムベースタイマ」「第10章 ウォッチドッグタイマ」「第11章 時計タイマ」を参照してください。

第5章

低消費電力制御回路

低消費電力制御回路の機能と動作について説明します。

- 5.1 低消費電力制御回路の概要
- 5.2 低消費電力制御回路のブロックダイアグラム
- 5.3 低消費電力制御回路のレジスタ
- 5.4 クロック選択の状態遷移

5.1 低消費電力制御回路の概要

低消費電力制御回路は、主に低消費電力の動作モードで用いられます。また、レジスタのビット設定により CPU 間欠動作機能や発振安定待ち時間の設定を行うことができます。

全体のブロックダイアグラム中では、クロック制御回路の一部となります (「1.3 ブロックダイアグラム」を参照)。

■ 低消費電力制御回路の動作モード

MB90540/545 では次の動作モードをサポートしています。動作モードとして、PLL クロックモード・PLL スリープモード・PLL 時計モード・疑似時計モード・メインクロックモード・メインスリープモード・メイン時計モード・メインストップモード・サブクロックモード・サブスリープモード・サブ時計モード・サブストップモード・ハードウェアスタンバイモードがあり、PLL クロックモード以外の動作モードが低消費電力モードになります。

■ CPU 間欠動作機能

CPU 間欠動作機能は、内蔵レジスタ、内蔵メモリ (ROM, RAM, I/O, リソースメモリ)、外部バスアクセスを行うときに CPU に供給するクロックを一定期間停止させ、内蔵バスサイクルの起動を遅らせる機能であり、内蔵リソースに高速クロックを供給したまま CPU の実行速度を下げることで、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) 内の CG1, CG0 ビットにより、CPU に供給するクロックの一時停止サイクル数の選択を行います。

外部バス動作では周辺リソースと同じクロック信号が引き続き使用されることに注意してください。

また、CPU 間欠動作機能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、内蔵リソースにアクセスを行う回数に、一時停止サイクル数を掛けた補正値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。

■ メインクロック発振安定待ち時間設定

クロック選択レジスタ (CKSCR) 内の WS1, WS0 ビットにより、ストップモードの解除時のメインクロック発振安定待ち時間の選択を行います。X0, X1 端子に接続する発振回路、発振素子の種類や特性に応じて発振安定待ち時間を選択してください。

パワーオンリセットハードウェアスタンバイ以外のリセットは本ビットを初期化しません。パワーオンリセット発生後およびハードウェアスタンバイモード解除後は本ビットを "11_B" に初期化します。そのため、パワーオンハードウェアスタンバイ後のメインクロック発振安定待ち時間は、原発振の約 2^{18} カウントになります。

< 注意事項 >

G サフィックスなしの製品では、ハードウェアスタンバイモード時の発振安定待ち時間は、WS1, WS0 により決定され、WS1, WS0 ビットはハードウェアスタンバイでは初期化されません。

■ マシニングクロックの切換え

● メインクロック・PLL クロックの切換え

クロック選択レジスタ (CKSCR) 内の MCS ビットへの書き込み動作により、メインクロックと PLL クロックの切換えを行います。

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^{13} マシニングクロックサイクル) 後、メインクロックから PLL クロックに切り換わります。

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信号のエッジが次に一致するタイミング (1 ~ 8PLL クロックサイクル以内) で PLL クロックからメインクロックに切り換わります。

MCS ビットを書き換えても即座にマシニングクロックの切換えが行われないので、マシニングクロックに依存するリソースを操作する場合には、MCM ビットを参照してマシニングクロックスピードの切換えが行われたことを確認した後に、リソースの操作を行ってください。

● メインクロック・サブクロックの切換え

2 系統品ではクロック選択レジスタ (CKSCR) 内の SCS ビットへの書き込み動作により、メインクロックとサブクロックの切換えを行います。

SCS ビットを "1" から "0" に書き換えた場合には、サブクロックに同期をとって (約 130 μ s)、動作はメインクロックからサブクロックに切り換わります。

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、動作はサブクロックからメインクロックに切り換わります。

SCS ビットを書き換えても即座にマシニングクロックの切換えが行われないので、マシニングクロックの動作を確認した後に、リソースの操作を行ってください。

なお、サブクロックは 1 系統品では使用できません。

< 注意事項 >

サブクロックモードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、サブクロックの 2 マシンサイクル以上 "L" レベルを入力してください。

● マシニングクロック初期化

MCS ビットは、外部端子、RST ビットによるリセットでは、初期化されません。そのほかのリセットでは、"1" に初期化されます。

< 注意事項 >

電源を投入した場合またはハードウェアスタンバイモード、ストップモードを解除した場合は、サブクロック発振安定待ち時間 (約 2 秒) が発生します。この間に、メインクロックモードからサブクロックモードに切り換えた場合は、発振安定待ち時間が発生します。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。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。

■ PLL クロック逡倍機能

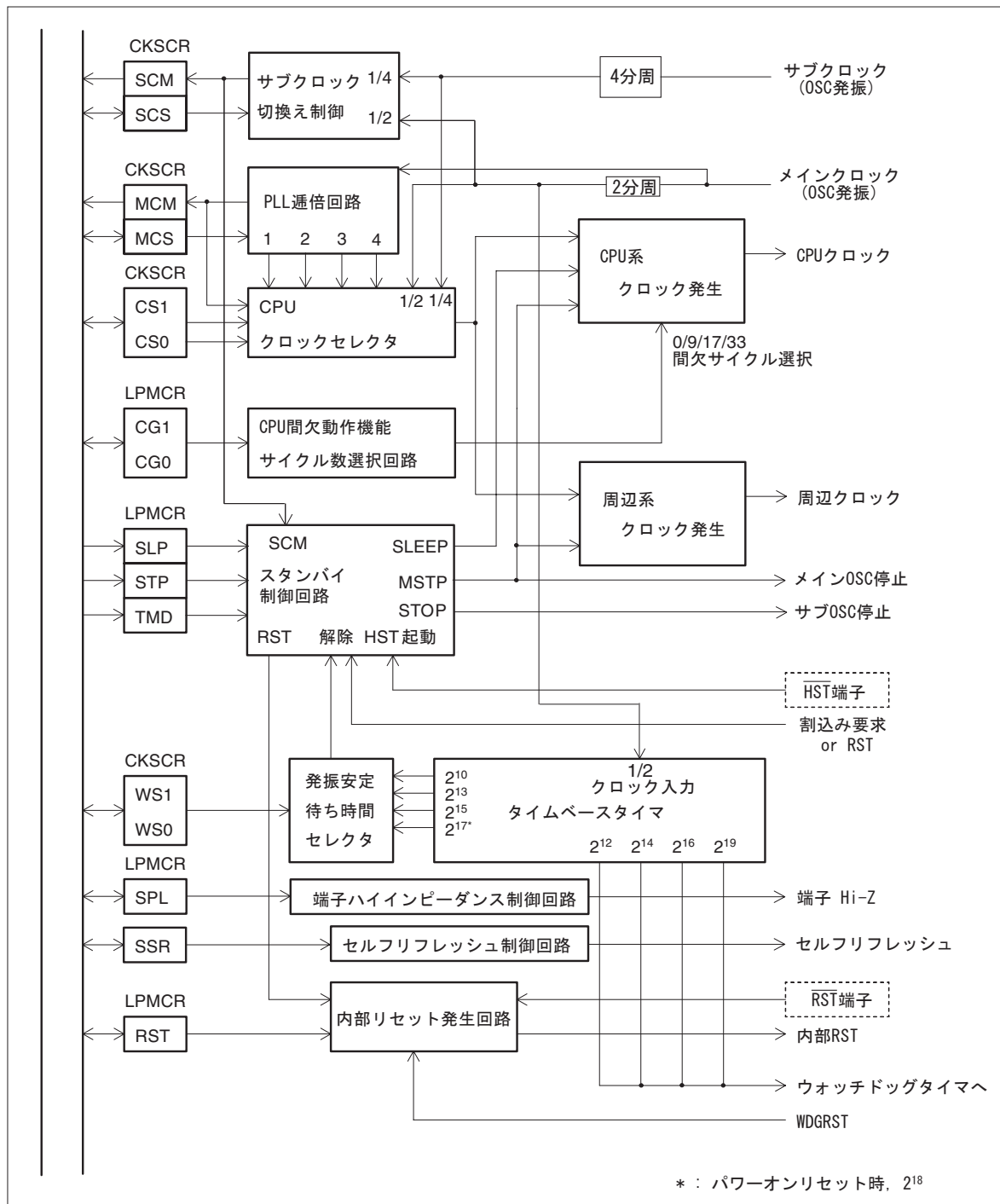
PLL クロックの逡倍率は CS1, CS0 ビットの設定により , 1, 2, 3, 4 逡倍の中から選択することができます。

5.2 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のブロックダイアグラムを示します。

■ 低消費電力制御回路のブロックダイアグラム

図 5.2-1 低消費電力制御回路とクロック発生部のブロックダイアグラム



5.3 低消費電力制御回路のレジスタ

低消費電力制御回路のレジスタには、次の 2 種類があります。

- 低消費電力モード制御レジスタ
- クロック選択レジスタ

■ 低消費電力制御回路のレジスタ

低消費電力制御回路のレジスタのビット構成を図 5.3-1 に示します。

図 5.3-1 低消費電力制御回路のレジスタ

図 3-3-1 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									
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000A0 _H	STP	SLP	SPL	RST	TMD	CG1	CG0	予約	LPMCR
リード / ライト	(W)	(W)	(R/W)	(W)	(W)	(R/W)	(R/W)	(-)	
初期値	(0)	(0)	(0)	(1)	(1)	(0)	(0)	(0)	

クロック選択レジスタ									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 0000A1 _H	SCM	MCM	WS1	WS0	SCS	MCS	CS1	CS0	CKSCR
リード / ライト	(R)	(R)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(1)	(1)	(1)	(1)	(1)	(1)	(0)	(0)	

5.3.1 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) の構成やビットの機能について説明します。

■ 低消費電力モード制御レジスタ (LPMCR)

消費電力モード制御レジスタ (LPMCR) のビット構成を図 5.3-2 に示します。

図 5.3-2 低消費電力モード制御レジスタ (LPMCR)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000A0 _H	STP	SLP	SPL	RST	TMD	CG1	CG0	予約	LPMCR
リード / ライト	(W)	(W)	(R/W)	(W)	(W)	(R/W)	(R/W)	(-)	
初期値	(0)	(0)	(0)	(1)	(1)	(0)	(0)	(0)	

【bit 7】 STP

"1" を書き込むことにより疑似時計モード (CKSCR.MCS=0 および SCS=1) またはストップモード (CKSCR.MCS=1 または SCS=0) に遷移します。"0" の書込みでは動作に影響はありません。リセット、時計モード解除またはストップモード解除で "0" にクリアされます。書込みのみ可能なビットです。読出し値は、常に "0" です。

【bit 6】 SLP

"1" を書き込むことによりスリープモードに遷移します。"0" の書込みでは動作に影響はありません。リセット、スリープモード解除またはストップモード解除で "0" にクリアされます。

STP ビットと SLP ビットに同時に "1" を書き込んだ場合、時計モードまたは疑似時計モードに遷移します。書込みのみ可能なビットです。読出し値は、常に "0" です。

【bit 5】 SPL

"0" の場合、時計モード疑似時計モードまたはストップモード時の外部端子のレベルを保持します。"1" の場合時計モード疑似時計モードまたはストップモード時の外部端子をハイインピーダンスにします。リセットで "0" にクリアされます。

【bit 4】 RST

"0" を書き込むことにより 3 マシンサイクルの内部リセット信号を発生します。"1" の書込みでは動作に影響はありません。読出し値は常に "1" です。

【bit 3】 TMD

2 系統品 :

"0" を書き込むことにより時計モードに遷移します。"1" の書込みでは動作に影響はありません。リセット、時計解除またはストップ解除で "1" にクリアされます。書込みのみ可能なビットです。読出し値は、常に "1" です。

1 系統品 :

常に "1" を書き込んでください。

【bit 2, bit1】CG1, CG0

CPU 間欠動作機能のクロック一時停止サイクル数を設定します。

パワーオン, ハードウェアスタンバイ, ウォッチドッグによるリセットにより "00_B" に初期化されます。そのほかのリセット要因によるリセットでは初期化されません。

表 5.3-1 に, CG ビット設定を示します。

表 5.3-1 CG ビット設定

CG1	CG0	CPU クロック一時停止サイクル数
0	0	0 サイクル (CPU クロック = リソースクロック)
0	1	9 サイクル (CPU クロック : リソースクロック = 1: 約 3 ~ 4)
1	0	17 サイクル (CPU クロック : リソースクロック = 1: 約 5 ~ 6)
1	1	33 サイクル (CPU クロック : リソースクロック = 1: 約 9 ~ 10)

【bit 0】予約ビット

本ビットには, 必ず "0" を書き込んでください。

■ 低消費電力モード制御レジスタのアクセス

ワード長で低消費電力モード制御レジスタ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。

低消費電力モード制御レジスタにデータを書き込むと, 低消費電力モードが起動します (ストップモードまたはスリープモード)。この場合には, 表 5.3-2 に示す命令を使用してください。そのほかの命令を使用して低消費電力モードを起動すると, 誤作動の原因となる場合があります。低消費電力モード制御レジスタから低消費電力モードへの遷移以外の機能を制御するには, どの命令でも使用できます。

ワード長で低消費電力モード制御レジスタにデータを書き込むには, データを偶数アドレスで書き込むようにしてください。奇数アドレスでデータを書き込んで低消費電力モードを起動すると, 誤作動の原因となる場合があります。

表 5.3-2 低消費電力モードへの遷移に使用される命令のリスト

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	

5.3.2 クロック選択レジスタ (CKSCR)

クロック選択レジスタ (CKSCR) のレジスタ配置とビットの機能について説明します。

■ クロック選択レジスタ (CKSCR)

クロック選択レジスタ (CKSCR) のビット構成を図 5.3-3 に示します。

図 5.3-3 クロック選択レジスタ (CKSCR)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 0000A1 _H	SCM	MCM	WS1	WS0	SCS	MCS	CS1	CS0	CKSCR
リード / ライト	(R)	(R)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(1)	(1)	(1)	(1)	(1)	(1)	(0)	(0)	

【bit 15】SCM

マシンのクロックとして、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。書込みは、動作に影響しません。

【bit 14】MCM

マシンのクロックとして、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、メインクロックが選択されていることを示します。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。なお、PLL クロックの発振安定待ち時間は、 2^{13} メインクロックサイクル固定です。書込みは、動作に影響しません。

【bit 13, bit12】WS1, WS0

ストップモードが解除されているときは、WS1 と WS0 を使用して発振安定待ち時間を指定します。X0 および X1 端子に接続されている発振回路と発振デバイスの種類と特性に応じて発振安定待ち時間を指定してください。

これらのビットはパワーオンリセット* 以外のリセットでは初期化されません。パワーオンリセット時に "11_B" に初期化されます。したがって、パワーオンリセット時には、発振安定待ち時間は原発振の約 2^{18} カウントです。読み書き可能なビットです。

*:G サフィックス付き製品は、サブクロックモード時のハードウェアスタンバイによるリセットでも初期化されます。

表 5.3-3 に、WS ビット設定を示します。

表 5.3-3 WS ビット設定

WS1	WS0	発振安定待ち時間 (原発振 4 MHz 時)
0	0	約 256 μ s (発振 2^{10} カウント)
0	1	約 2.05 ms (発振 2^{13} カウント)
1	0	約 8.19 ms (発振 2^{15} カウント)
1	1	約 32.77 ms (発振 2^{17} カウント) 約 65.54 ms (原発振 2^{18} カウント) パワーオンリセットのみ

【bit 11】SCS

2 系統品：

マシクロックとして、メインクロックまたはサブクロックのどちらかを選択するかを指示するビットです。"0" を書き込むことにより、サブクロックを選択します。"1" を書き込むことにより、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、サブクロックに同期をとって (約 130 μ s) サブクロックモードに切り換わります。"0" の場合に、"1" を書き込むと、メインクロックの発振安定待ち時間を発生した後、メインクロックモードに切り換わり自動的にタイムベースタイマがクリアされます。SCS ビットおよび MCS ビットがともに "0" の場合には、SCS ビットが優先され、サブクロックが選択されます。

1 系統品：

常に "1" を書き込んでください。

【bit10】MCS

マシクロックとして、メインクロックまたは PLL クロックのどちらを選択するかを指示するビットです。"0" を書き込むことにより、PLL クロックを選択します。"1" を書き込むことにより、メインクロックを選択します。"1" の場合に、"0" を上書きすると、PLL クロックの発振安定待ち時間を発生するために、自動的にタイムベースタイマがクリアされます。なお、PLL クロックの発振安定待ち時間は、 2^{13} メインクロックサイクル固定です。

また、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 (原発振 4 MHz 時、動作クロック速度は 2 MHz になります)。

MCS ビットが "0" から "1" に変更されると、メインクロックは PLL のエッジと一致したときに PLL クロックを引き継ぎます (およそ 1 ~ 8PLL クロックサイクル後)。MCS ビットに書き込んでも、すぐにはマシクロックの変更は行われません。マシクロックに依存するリソースを使用するには、常に前もって MCS ビットを参照してマシクロックが変更されているかを確認してください。

< 注意事項 >

パワーオン、ハードウェアスタンバイ、ウォッチドッグによるリセットにより "1" に初期化されます。MCS は外部端子または RST ビットを使用したリセットでは初期化されません。

【bit9, bit8】CS1, CS0

PLL クロックの逡倍率を決定するビットです。外部端子, RST ビット, ウォッチドッグタイマおよびハードウェアスタンバイ* によるリセットでは初期化されません。パワーオンによるリセットでのみ "00_B" に初期化されます。

※:G サフィックス付き製品はサブクロックモード時のハードウェアスタンバイによるリセットでも初期化されます。

MCS 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" にした(メインクロックモード)後に, CS ビットを書き換えてください。読み書き可能なビットです。

表 5.3-4 に, CS ビットの設定を示します。

表 5.3-4 CS ビットの設定

CS1	CS0	マシンクロック (原発振 4MHz 時)
0	0	4 MHz (動作周波数 =OSC 発振クロック)
0	1	8 MHz (動作周波数 =OSC 発振クロック × 2)
1	0	12 MHz (動作周波数 =OSC 発振クロック × 3)
1	1	16 MHz (動作周波数 =OSC 発振クロック × 4)

< 注意事項 >

動作電圧が 5 V のとき, 原発振は 3 MHz ~ 16 MHz の間になります。CPU と周辺リソース回路の最大動作周波数は 16 MHz なので, 16 MHz を超える逡倍係数が指定された場合は正常な動作が保証されません。例えば外部クロックが 16MHz の場合, 逡倍係数には "1" しか指定できません。

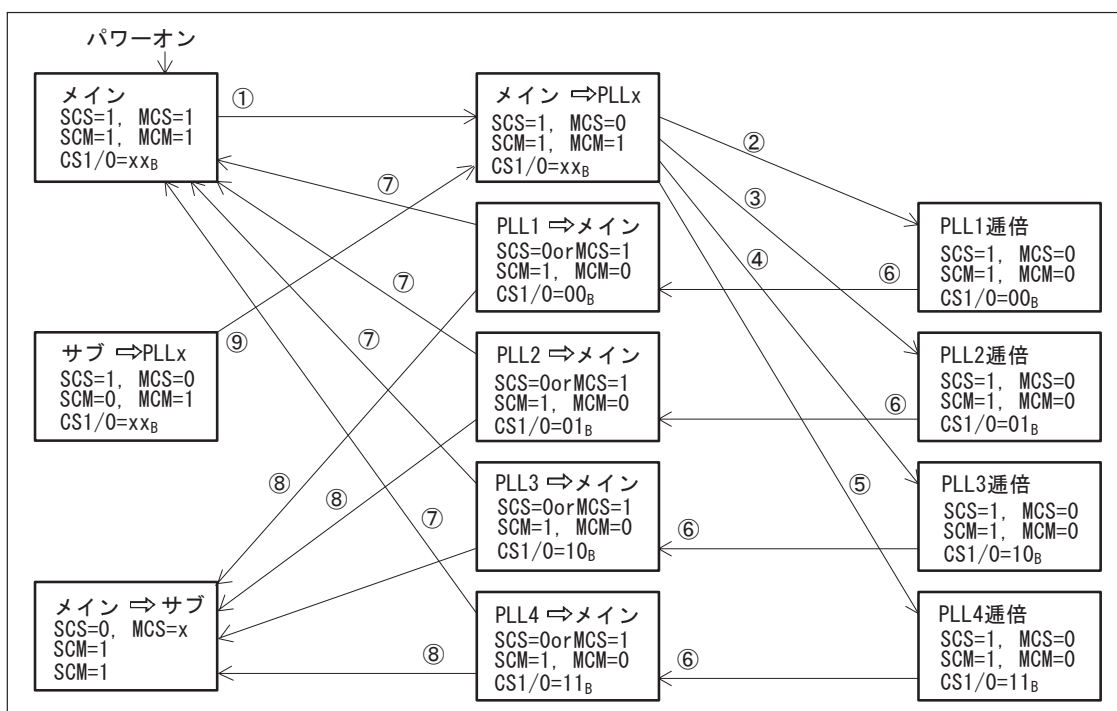
VCO 発振の最小動作周波数は 4 MHz です。4 MHz を下回る発振を指定しないでください。

5.4 クロック選択の状態遷移

図 5.4-1 , 図 5.4-2 および図 5.4-3 にクロック選択の状態遷移を示します。

■ クロック選択の状態遷移

図 5.4-1 クロック選択の状態遷移図 1(2 系統品その 1)



MCS ビットクリアおよび SCS ビットセット

PLL クロック発振安定待ち時間終了および CS1/0=00_B

PLL クロック発振安定待ち時間終了および CS1/0=01_B

PLL クロック発振安定待ち時間終了および CS1/0=10_B

PLL クロック発振安定待ち時間終了および CS1/0=11_B

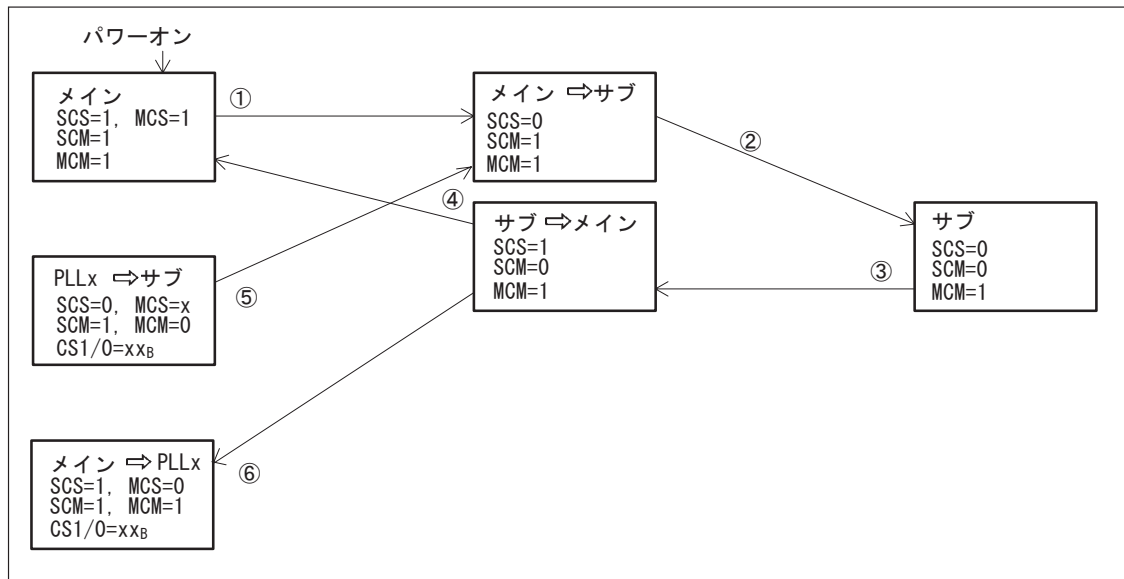
MCS ビットセットまたは SCS ビットクリア

PLL クロックとメインクロックの同期タイミングおよび SCS=1

PLL クロックとメインクロックの同期タイミングおよび SCS=0

メインクロック発振安定待ち時間終了および MCS=0

図 5.4-2 クロック選択モードの状態遷移図 2(2 系統品その 2)



SCS ビットクリア

サブクロックのエッジ検出タイミング

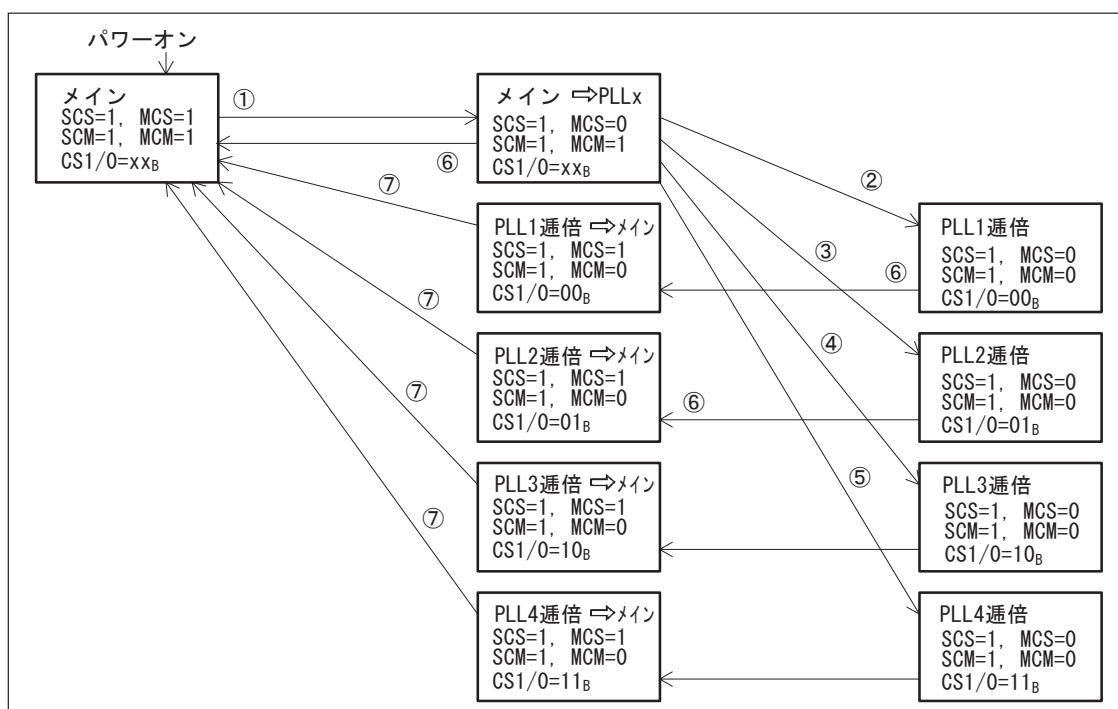
SCS ビットセット

メインクロック発振安定待ち時間終了および MCS=1

PLL クロックとメインクロックの同期タイミングおよび SCS=0

メインクロック発振安定待ち時間終了および MCS=0

図 5.4-3 クロック選択の状態遷移図 3(1 系統品)



MCS ビットクリアおよび SCS ビットクリア

PLL クロック発振安定待ち時間終了および CS1/0=00_B

PLL クロック発振安定待ち時間終了および CS1/0=01_B

PLL クロック発振安定待ち時間終了および CS1/0=10_B

PLL クロック発振安定待ち時間終了および CS1/0=11_B

MCS ビットセット

PLL クロックとメインクロックの同期タイミングおよび SCS=1

< 注意事項 >

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。切り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り換えを行った場合、切り換わらない場合があります。

第6章

低消費電力モード

低消費電力モードの機能と動作について説明します。

6.1 低消費電力モード

6.2 低消費電力モードの状態遷移

6.3 低消費電力モード状態遷移図

6.1 低消費電力モード

MB90540/545 では次の動作モードをサポートしています。

- PLL クロックモード
- PLL スリープモード
- PLL 時計モード
- 擬似時計モード
- メインクロックモード
- メインスリープモード
- メイン時計モード
- メインストップモード
- サブクロックモード
- サブスリープモード
- サブ時計モード
- サブストップモード
- ハードウェアスタンバイモード
- CPU 間欠動作

PLL クロックモード以外の動作モードが低消費電力モードに分類されています。

■ 低消費電力モード

● メインクロックモード・メインスリープモード

メインクロック (メイン OSC 発振クロック) とサブクロック (サブ OSC 発振クロック) で動作させるモードです。動作クロック信号はメインクロック信号を 2 分周したもので、サブクロック信号 (サブ OSC 発振クロック) は PLL クロック (VCO 発振クロック) が停止している間、時計クロック信号として使用されます。

● サブクロックモード・サブスリープモード

サブクロックのみで動作させるモードです。サブクロックを 4 分周したものが動作クロックとなり、メインクロックと PLL クロックは停止します。

● PLL スリープモード・メインスリープモード

CPU の動作クロックのみ停止させるモードで、CPU クロック以外は動作しています。

● 擬似時計モード

時計タイマおよびタイムベースタイマのみを動作させるモードです。

- PLL 時計モード・メイン時計モード・サブ時計モード

時計タイマのみを動作させるモードです。サブクロック信号のみで動作させるモードで、メインクロックおよび PLL クロックを停止させるモードです。なお、PLL 時計モード・メイン時計モード・サブ時計モードの違いは、割込みによる復帰時の動作モードがそれぞれ 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であることであり、時計モードの動作は違いありません。

- メインストップモード・サブストップモードとハードウェアスタンバイモード

発振を停止させるモードであり、もっとも低消費電力でデータを保持できます。なお、メインストップモード・サブストップモードの違いは、割込みによる復帰時の動作モードがそれぞれメインクロックモード・サブクロックモードであることであり、ストップモードの動作は違いありません。

- CPU 間欠動作機能

内蔵レジスタ、内蔵メモリ、内蔵リソースおよび外部バスアクセスを行うときに、CPU に供給するクロックを間欠動作させる機能であり、内蔵リソースに高速クロックを供給したまま CPU の実行速度を下げることで、低消費電力で処理が行えます。

< 注意事項 >

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。切り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り換えを行った場合、切り換わらない場合があります。

■ 低消費電力モード動作状態

表 6.1-1, 表 6.1-2 に, 各動作モードにおけるチップの動作状態を示します。

表 6.1-1 低消費電力モード動作状態 (2 系統品)

状態	遷移条件	サブ発振	メイン発振	マシニングクロック	CPU	周辺	端子	解除方法
サブクロック	SCS=0 MCS=x	動作	停止	動作	動作	動作	動作	外部リセット
サブスリープ	SCS=0 MCS=x SLP=1	動作	停止	動作	停止	動作	動作	外部リセット 割込み
メインスリープ	SCS=1 MCS=1 SLP=1	動作	動作	動作	停止	動作	動作	外部リセット 割込み
PLLスリープ	SCS=1 MCS=0 SLP=1	動作	動作	動作	停止	動作	動作	外部リセット 割込み
擬似時計 (SPL=0)	SCS=1 MCS=0 STP=1	動作	動作	停止	停止	停止	前の状態を保持	外部リセット 割込み ^{*1}
擬似時計 (SPL=1)	SCS=1 MCS=0 STP=1	動作	動作	停止	停止	停止	Hi-Z	外部リセット 割込み ^{*1}
時計 (SPL=0)	SCS=x MCS=x TMD=0	動作	停止	停止	停止	停止	前の状態を保持	外部リセット 割込み ^{*2}
時計 (SPL=1)	SCS=x MCS=x TMD=0	動作	停止	停止	停止	停止	Hi-Z	外部リセット 割込み ^{*2}
ストップ (SPL=0)	MCS=1 or SCS=0 STP=1	停止	停止	停止	停止	停止	前の状態を保持	外部リセット 割込み ^{*3}
ストップ (SPL=1)	MCS=1 or SCS=0 STP=1	停止	停止	停止	停止	停止	Hi-Z	外部リセット 割込み ^{*3}
ハードウェアスタンバイ	HST=L	停止	停止	停止	停止	停止	Hi-Z	HST="H"

*1: 時計タイマ, タイムベースタイマおよび外部割込み

*2: 時計タイマおよび外部割込み

*3: 外部割込み

表 6.1-2 低消費電力モード動作状態 (1 系統品)

	遷移条件	サブ発振	メイン 発振	マシン クロック	CPU	周辺	端子	解除方法
メイン スリープ	SCS=1 MCS=1 SLP=1	-	動作	動作	停止	動作	動作	外部 リセット 割込み
PLL スリープ	SCS=1 MCS=0 SLP=1	-	動作	動作	停止	動作	動作	外部 リセット 割込み
擬似時計 (SPL=0)	SCS=1 MCS=0 STP=1	-	動作	停止	停止	停止	保持	外部 リセット 割込み ^{*1}
擬似時計 (SPL=1)	SCS=1 MCS=0 STP=1	-	動作	停止	停止	停止	Hi-Z	外部 リセット 割込み ^{*1}
ストップ (SPL=0)	SCS=1 MCS=1 STP=1	-	停止	停止	停止	停止	保持	外部 リセット 割込み ^{*2}
ストップ (SPL=1)	SCS=1 MCS=1 STP=1	-	停止	停止	停止	停止	Hi-Z	外部 リセット 割込み ^{*2}
ハードウェア スタンバイ	$\overline{\text{HST}}=\text{L}$	-	停止	停止	停止	停止	Hi-Z	$\overline{\text{HST}}=\text{"H"}$

*1: タイムベースタイマおよび外部割込み

*2: 外部割込み

6.1.1 スリープモード

スリープモードとは、CPU に供給するクロックのみを停止するモードで、CPU は停止し、内部周辺リソース回路は動作を続けます。

■ スリープモードへの遷移

低消費電力モード制御レジスタ (LPMCR) 内の SLP ビットに "1", TMD ビットに "1", STP ビットに "0" を書き込むことによりスリープモードへの遷移が始まります。

SLP ビットに "1" を書き込んだときに割込み要求が発生しているとスタンバイ制御回路はスリープモードへ遷移しません。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の分岐を行います。

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。

■ スリープモードの解除

スタンバイ制御回路は、リセット信号入力または割込み要求の発生によりスリープモードを解除します。リセット要因によりスリープモードを解除した場合は、スリープモードを解除した上でリセット状態になります。

スリープモード中に周辺回路および内蔵リソース回路から割込みレベルが 7 より強い割込み要求が発生すると、スタンバイ制御回路は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) 内の I フラグ、割込みレベルマスクレジスタ (ILM) と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とき、CPU はスタンバイ書込み保留でない命令を実行後に割込み処理を実行します。割込みが受け付けられないときは、スリープモードに入れた命令の次の命令から処理を続行します。

< 注意事項 >

サブスリープモードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、サブクロックの 2 マシンサイクル以上 "L" レベルを入力してください。

6.1.2 擬似時計モード

擬似時計モードとは、原発振（メインおよびサブ）と時計タイマとタイムベースタイマ以外の動作を止めるモードであり、MB90540/545 のほぼ全機能が停止します。

■ 擬似時計モードへの遷移

クロック選択レジスタ (CKSCR) の SCS ビットに "1", MCS ビットに "0", 低消費電力モード制御レジスタ (LPMCR) 内の TMD ビットに "1", STP ビットに "1" を書き込むことにより擬似時計モードへの遷移が始まります。

また、擬似時計モード中の I/O 端子を直前の状態に保持するのか、ハイインピーダンス状態にするのか、低消費電力モード制御レジスタ (LPMCR) 内の SPL ビットにより制御できます。

STP ビットに "1" を書き込んだときに割込み要求が発生している場合は、スタンバイ制御回路は擬似時計モードに遷移しません。

擬似時計モードでは、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。

■ 擬似時計モードの解除

スタンバイ制御回路は、リセット信号入力または割込み要求の発生により擬似時計モードを解除します。リセット要因により擬似時計モードを解除した場合は、擬似時計モードを解除した上で、リセット状態になります。

擬似時計モードからの復帰の場合、スタンバイ制御回路はまず擬似時計モードを解除し、その後 PLL クロック発振安定待ち状態に遷移します。擬似時計モードからの解除が割込み要求で始まったときには、リセットシーケンスはメインクロックを使用して行われます。

周辺リソース回路から割込みレベルが 7 より強い割込み要求が発生すると、スタンバイ制御回路は擬似時計モードを解除します。擬似時計モードの解除後は、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) と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ときは、CPU はスタンバイ書込み命令の次の書込み保留でない命令を実行後に割込み処理を実行します。割込みが受け付けられないときは擬似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 > (MB90543G(S)/547G(S)/548G(S) のみ)

- 擬似時計モードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、100 μs 以上 "L" レベルを入力してください。
- 割込みにより擬似時計モードから復帰する場合は、割込み要求受付け後最大 80 μs 経過後割込み処理が行われます。

6.1.3 時計モード

時計モードとは、サブ原発振と時計タイマ以外の動作を止めるモードであり、MB90540/545 のほぼ全機能が停止します。

なお、時計モードは 1 系統品では使用できません。

■ 時計モードへの遷移

クロック選択レジスタ (CKSCR) 内の TMD ビットに "0" を書き込むことにより時計モードへの遷移が始まります。

また、時計モード中の I/O 端子を直前の状態に保持するのか、ハイインピーダンス状態にするのかを、低消費電力モード制御レジスタ (LPMCR) 内の SPL ビットにより制御できます。

TMD ビットに "1" を書き込んだときに割込み要求が発生している場合は、スタンバイ制御回路は時計モードに遷移しません。

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。

■ 時計モードの解除

スタンバイ制御回路は、リセット信号入力または割込みの発生により時計モードを解除します。リセット要因により時計モードを解除した場合は、時計モードを解除した上で、リセット状態になります。

サブ時計モードからの復帰の場合、スタンバイ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遷移します。サブ時計モードからの解除がリセット要因であったときには、リセットシーケンスはサブクロック信号を使用して行われます。

メイン時計モードおよび PLL 時計モードからの復帰の場合、スタンバイ制御回路はまず時計モードを解除し、その後メインクロック発振安定待ち状態に遷移します。時計モードからの解除がリセット要因であったときには、リセットシーケンスはサブクロック信号を使用して行われます。

周辺リソース回路などから割込みレベルが "7" より強い割込み要求が発生すると、スタンバイ制御回路は時計モードを解除します。時計モードの解除後は、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) 内の I フラグ、割込みレベルマスクレジスタ (ILM) と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ときは、CPU はスタンバイ書込み命令の次の書込み保留でない命令を実行後に割込み処理を実行します。割込みが受け付けられないとき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 >

時計モードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、サブクロックの 2 マシンサイクル以上 "L" レベルを入力してください。

6.1.4 ストップモード

ストップモードとは、原発振（メインおよびサブ）を止めるモードであり、MB90540/545 の全デバイス機能が停止します。したがって、もっとも低消費電力でデータを保持することができます。

■ ストップモードへの遷移

クロック制御レジスタ内の SCS ビットに "0" または MCS ビットに "1", 低消費電力モード制御レジスタ (LPMCR) 内の STP ビットに "1" を書き込むことによりスタンバイ制御回路をストップモードに遷移します。

また、I/O 端子を直前の状態に保持するのか、ハイインピーダンス状態にするのか、低消費電力モード制御レジスタ (LPMCR) 内の SPL ビットにより制御できます。

STP ビットに "1" を書き込んだときに割込み要求が発生している場合は、スタンバイ制御回路はストップモードに遷移しません。

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。

■ ストップモードの解除

スタンバイ制御回路は、リセット信号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。リセット要因によりストップモードを解除した場合は、ストップモードを解除した上で、リセット状態になります。

サブストップモードからの復帰の場合、スタンバイ制御回路はまずサブクロック発振安定待ちモードに遷移し、その後ストップモードを解除します。ストップモードからの解除がリセット要因であったときには、リセットシーケンスはサブクロック発振安定待ち時間経過後実行されます。

メインストップモードからの復帰の場合、スタンバイ制御回路はまずメインクロック発振安定待ちモードに遷移し、その後ストップモードを解除します。ストップモードからの解除がリセット要因であったときには、リセットシーケンスはメインクロック発振安定待ち時間経過後実行されます。

ストップモード中に周辺回路またはほかのソースから割込みレベルが 7 より強い割込み要求が発生すると、ストップモードから MB90540/545 を解除します。サブストップモードの解除後は、サブクロックの発振安定待ち時間が適用され、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) 内の I フラグ、割込みレベルマスクレジスタ (ILM) と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ときは、CPU はスタンバイ書込み命令の次の書込み保留でない命令を実行後に割込み処理を実行します。割込みが受け付けられないとき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。

メインストップモードの解除後は、クロック選択レジスタ (CKSCR) 内の WS1, 0 ビットで指定された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) 内の I フラグ、割込みレベルマスクレジスタ (ILM) と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ときは、CPU はスタンバイ書込み命令の次の書込み保留でない命令を実行後に割込み処理に分岐します。割込みが受け付けられないとき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 >

ストップモードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、振動子の発振時間^{*}+4 マシンサイクル以上 "L" レベルを入力してください。

*：振動子の発振時間は、振幅が 90 % に達する時間です。水晶発振子は数 ms ~ 数十 ms, FAR/ セラミック発振子は数百 μs ~ 数 ms, 外部クロックは 0 ms となります。

6.1.5 ハードウェアスタンバイモード

ハードウェアスタンバイモードとは、リセットを含むほかの状態に関係なく $\overline{\text{HST}}$ 端子が "L" レベルの間、発振を停止させ、I/O 端子をすべてハイインピーダンス状態にするモードです。

■ ハードウェアスタンバイモードへの遷移

$\overline{\text{HST}}$ 端子を "L" レベルにすることにより、どんなモードからでもスタンバイ制御回路をハードウェアスタンバイモードに遷移することができます。

ハードウェアスタンバイモードでは内部 RAM の内容は保持されますが、アキュムレータなどの専用レジスタは初期値にリセットされます。

■ ハードウェアスタンバイモードの解除

ハードウェアスタンバイモードの解除は、 $\overline{\text{HST}}$ 端子によってのみ解除することができます。 $\overline{\text{HST}}$ 端子を "H" レベルにすると、スタンバイ制御回路をハードウェアスタンバイモードから解除し、内部リセット信号を有効にした後、発振安定待ちモードに遷移します。さらに、メインクロック発振安定待ち時間の経過後、スタンバイ制御回路は内部リセットを解除し、その結果、CPU はリセットシーケンスから実行を開始します。

ハードウェアスタンバイモードの発振安定待ち時間は、G サフィックス付きの製品では、原発振の 2^{18} カウント固定であり、G サフィックスなしの製品ではクロック選択レジスタ (CKSCR) の WS1, WS0 ビットにより決定されます。

6.1.6 CPU 間欠動作

CPU 間欠動作機能は、レジスタ、内部メモリ (ROM, RAM, I/O, リソース) または外部バスにアクセスしたときに、内部バスサイクルの活動を遅らせて CPU に供給されるクロックを停止させます。高速度クロックが内部リソースに供給されている間、CPU 実行速度は落ちるので、低消費電力での処理が可能になります。クロック停止のサイクルカウントは CG1 および CG0 ビットで指定されます。外部バス動作自体は、リソースと同じクロックを使用して行われます。

■ CPU 間欠動作

CPU 間欠動作機能を使用時の命令実行時間は、通常の実行時間に補正値を加えて求めます。補正値はレジスタ、内部メモリ、内部リソース、外部バスへのアクセス数に停止のサイクルカウント数を乗じて求めます。

6.2 低消費電力モードの状態遷移

低消費電力モードでは、クロック選択レジスタや低消費電力モード制御レジスタの設定条件によって各状態が遷移します。

■ 低消費電力モードの遷移条件

以下に、表と図で使用されている記号の意味を説明します。

- MCS:MCS ビット (クロック選択レジスタ)(MCS=0 で PLL クロックモード選択)
- SCS:SCS ビット (クロック選択レジスタ)(SCS=0 でサブクロックモード選択)
- STP:STP ビット (低消費電力モード制御レジスタ)(STP=1 でストップモード選択)
- SLP:SLP ビット (低消費電力モード制御レジスタ)(SLP=1 でスリープモード選択)
- TMD:TMD ビット (低消費電力モード制御レジスタ)(TMD=0 で時計モード選択)
- MCM:MCM ビット (クロック選択レジスタ)(MCM=0 で PLL クロック使用中)
- SCM:SCM ビット (クロック選択レジスタ)(SCM=0 でサブクロック使用中)
- SCD: サブクロック発振停止 (SCD=1 で、サブクロック発振停止)
- MCD: メインクロック発振停止 (MCD=1 で、メインクロック発振停止)
- PCD:PLL クロック発振停止 (PCD=1 で、PLL クロック発振停止)

表 6.2-1 2 系統品の遷移条件一覧表 (1 / 3)

遷移前の状態	遷移条件	遷移後の状態
パワーオン	01 メイン発振安定待ち時間終了	メインモード
メイン発振安定	05 メイン発振安定待ち時間終了	メインモード
メインモード	06 SCS=0 書込み	MS 遷移モード
	07 SCS=1 ・ MCS=0 書込み	MP 遷移モード
	31 TMD=1 ・ STP=0 ・ SLP=1 書込み	メインスリープモード
	32 TMD=0 書込み	メイン時計遷移モード
	33 TMD=1 ・ STP=1 書込み	メインストップ
PLL モード	21 SCS=0 書込み	PS 遷移モード
	20 SCS=1 ・ MCS=1 書込み	PM 遷移モード
	59 TMD=1 ・ STP=0 ・ SLP=1 書込み	PLL スリープモード
	58 TMD=0 書込み	PLL 時計遷移 P
	57 TMD=1 ・ STP=1 書込み	擬似時計遷移モード
サブモード	10 SCS=1 ・ MCS=1 書込み	SM 遷移モード
	12 SCS=1 ・ MCS=0 書込み	SP 遷移モード
	11 リセット起動	メイン発振安定
	42 TMD=1 ・ STP=0 ・ SLP=1 書込み	サブスリープモード
	43 TMD=0 書込み	サブ時計モード
	44 TMD=1 ・ STP=1 書込み	サブストップモード

表 6.2-1 2 系統品の遷移条件一覧表 (2 / 3)

遷移前の状態	遷移条件	遷移後の状態
PM 遷移モード	13 PLL メイン切換え待ち終了	メインモード
	38 TMD=1 ・ STP=0 ・ SLP=1 書込み	PM 遷移スリープモード
	39 TMD=0 書込みおよび PLL メイン切換え待ち終了	メイン時計遷移モード
	40 TMD=1 ・ STP=1 書込みおよび PLL メイン切換え待ち終了	メインストップモード
SM 遷移モード	02 メイン発振安定待ち時間終了	メインモード
	03 リセット起動または割込み	メイン発振安定
	04 SCS=0 書込み	サブモード
	27 TMD=1 ・ STP=0 ・ SLP=1 書込み	SM 遷移スリープモード
	28 TMD=0 書込みおよびメイン発振安定待ち時間終了	メイン時計モード
	29 TMD=1 ・ STP=1 書込みおよびメイン発振安定待ち時間終了	メインストップモード
MP 遷移モード	16 PLL 発振安定待ち時間終了	PLL モード
	14 SCS=1 ・ MCS=1 書込み	メインモード
	15 SCS=0 書込み	MS 遷移モード
	68 TMD=1 ・ STP=0 ・ SLP=1 書込み	MP 遷移スリープモード
	70 TMD=0 書込み	PLL 時計遷移モード
	69 TMD=1 ・ STP=1 書込み	擬似時計モード
SP 遷移モード	17 メイン発振安定待ち時間終了	MP 遷移モード
	18 MCS=1 書込み	SM 遷移モード
	19 リセット起動	メイン発振安定
	75 TMD=1 ・ STP=0 ・ SLP=1 書込み	SP 遷移スリープモード
	76 TMD=0 書込み	PLL 時計モード
	78 TMD=1 ・ STP=1 書込みおよびメイン発振安定待ち時間終了	擬似時計モード
MS 遷移モード	09 メイン サブクロック切換え待ち終了	サブモード
	08 リセット起動	メインモード
	51 TMD=1 ・ STP=0 ・ SLP=1 書込み	MS 遷移スリープモード
	52 TMD=0 書込みおよびメイン サブ切換え待ち終了	サブ時計モード
	53 TMD=1 ・ STP=1 書込みおよびメイン サブ切換え待ち終了	サブストップモード
PS 遷移モード	23 PLL メインクロック切換え待ち終了	MS 遷移モード
	22 SCS=1 書込み	PM 遷移モード
	56 TMD=1 ・ STP=0 ・ SLP=1 書込み	PS 遷移スリープモード
メインスリープ	26 割込みまたはリセット起動	メインモード
SM 遷移スリープ	24 メイン発振安定待ち時間終了	メインスリープ
	25 割込みまたはリセット起動	SM 遷移モード
PM 遷移スリープ	34 PLL メインクロック切換え待ち終了	メインスリープモード
	35 割込みまたはリセット起動	PM 遷移モード
PLL スリープモード	63 割込みまたはリセット起動	PLL モード
MP 遷移スリープ	66 PLL 発振安定待ち時間終了	PLL スリープモード
	67 割込みまたはリセット起動	MP 遷移モード
SP 遷移スリープ	73 メイン発振安定待ち時間終了	MP 遷移スリープ
	74 割込みまたはリセット起動	SP 遷移モード

表 6.2-1 2 系統品の遷移条件一覧表 (3 / 3)

遷移前の状態	遷移条件	遷移後の状態
サブスリープ	46 割込みまたはリセット起動	サブモード
MS 遷移スリープ	49 メイン サブクロック切換え待ち終了	サブスリープモード
	50 割込みまたはリセット起動	MS 遷移モード
PS 遷移スリープ	54 PLL メインクロック切換え待ち終了	MS 遷移スリープモード
	55 割込みまたはリセット起動	PS 遷移モード
メイン時計	30 割込みまたはリセット起動	SM 遷移モード
メイン時計遷移	36 メイン サブクロック切換え待ち終了	メイン時計
	37 割込みまたはリセット起動	メインモード
PLL 時計	77 割込みまたはリセット起動	SP 遷移モード
PLL 時計遷移 M	72 メイン サブクロック切換え待ち終了	PLL 時計
	71 割込みまたはリセット起動	MP 遷移モード
PLL 時計遷移 P	65 PLL メインクロック切換え待ち終了	PLL 時計遷移 M
	64 割込みまたはリセット起動	PLL モード
サブ時計	47 割込みまたはリセット起動	サブモード
メインストップ	41 割込みまたはリセット起動	メイン発振安定
擬似時計	62 割込みまたはリセット起動	MP 遷移モード
擬似時計遷移	61 PLL メインクロック切換え待ち終了	擬似時計モード
	60 割込みまたはリセット起動	PLL モード
サブストップ	48 割込み	サブ発振安定
	79 リセット起動	メイン発振安定
サブ発振安定	45 サブクロック発振安定待ち時間終了	サブモード
	80 リセット起動	メイン発振安定

< 注意事項 >

- ストップモードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、振動子の発振時間^{*}+4 マシンサイクル以上 "L" レベルを入力してください。
* : 振動子の発振時間は、振幅が 90% に達する時間です。水晶発振子は数 ms ~ 数十 ms, FAR/ セラミック発振子は数百 μs ~ 数 ms, 外部クロックは 0 ms となります。
- サブクロックモード, サブスリープモードおよび時計モードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、サブクロックの 2 マシンサイクル以上 "L" レベルを入力してください。

表 6.2-2 1 系統品の状態遷移条件一覧表

遷移前の状態	遷移条件	遷移後の状態
パワーオン	01 メイン発振安定待ち時間終了	メインモード
メイン発振安定	05 メイン発振安定待ち時間終了	メインモード
メインモード	07 SCS=1・MCS=0 書込み	MP 遷移モード
	31 TMD=1・STP=0・SLP=1 書込み	メインスリープ
	33 TMD=1・STP=1 書込み	メインストップ
PLL モード	20 SCS=1・MCS=1 書込み	PM 遷移モード
	59 TMD=1・STP=0・SLP=1 書込み	PLL スリープ
PM 遷移モード	13 PLL メイン切換えタイミグ待ち終了	メインモード
	38 TMD=1・STP=0・SLP=1 書込み	PM 遷移スリープ
	40 TMD=1・STP=1 書込み & PLL メイン切換えタイミグ待ち終了	メインストップ
MP 遷移モード	16 PLL 発振安定待ち時間終了	PLL モード
	14 SCS=1・MCS=1 書込み	メインモード
	68 TMD=1・STP=0・SLP=1 書込み	MP 遷移スリープ
	69 TMD=1・STP=1 書込み	擬似時計モード
メインスリープ	26 割込みまたはリセット起動	メインモード
PM 遷移スリープ	34 PLL メインクロック切換えタイミグ待ち終了	メインスリープ
	35 割込みまたはリセット起動	PM 遷移モード
PLL スリープ	63 割込みまたはリセット起動	PLL モード
MP 遷移スリープ	66 PLL 発振安定待ち時間終了	PLL スリープ
	67 割込みまたはリセット起動	MP 遷移モード
メインストップ	41 割込みまたはリセット起動	メイン発振安定
擬似時計	62 割込みまたはリセット起動	MP 遷移モード
擬似時計遷移	61 PLL メインクロック切換えタイミグ待ち終了	擬似時計モード
	60 割込みまたはリセット起動	PLL モード

< 注意事項 >

ストップモードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、振動子の発振時間*+4 マシンサイクル以上 "L" レベルを入力してください。

*：振動子の発振時間は、振幅が 90% に達する時間です。水晶発振子は数 ms ~ 数十 ms, FAR/セラミック発振子は数百 μs ~ 数 ms, 外部クロックは 0 ms となります。

6.3 低消費電力モード状態遷移図

図 6.3-1 ~ 図 6.3-7 に状態遷移図を示します。状態遷移図では、複雑になるのを避けるために、同時に発生した事象を段階的に遷移する図になっています。実際には、即座に状態の遷移は行われます。

■ 低消費電力モード状態遷移図 (2 系統品)

簡潔にするため、状態遷移図では、同時に発生しているイベントを段階的な遷移として表しています。しかし実際には、状態遷移は即座に行われます。

状態遷移図を例にすると、PLL クロックモードで MCS=1, SLP=1 を同時に設定した場合に、状態遷移図では、一度 PM 遷移モードに遷移した後に、PM 遷移スリープに遷移することになっていますが、実際には、即座に PLL クロックモードから PM 遷移スリープに遷移します。また、サブスリープモードで、リセットが起動された場合に、一度サブモードに遷移した後に、メイン発振安定時間に遷移することになっていますが、実際には、即座にサブスリープモードからメイン発振安定時間に遷移します。

< 注意事項 >

- ストップモードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、振動子の発振時間 * +4 マシンサイクル以上 "L" レベルを入力してください。
* : 振動子の発振時間は、振幅が 90% に達する時間です。水晶発振子は数 ms ~ 数十 ms, FAR/ セラミック発振子は数百 μs ~ 数 ms, 外部クロックは 0 ms となります。
- サブクロックモード、サブスリープモードおよび時計モードから外部リセット端子 ($\overline{\text{RST}}$ 端子) にてメインクロックモードへ復帰する場合は、サブクロックの 2 マシンサイクル以上 "L" レベルを入力してください。

図 6.3-1 低消費電力モード状態遷移図 (2 系統品)A

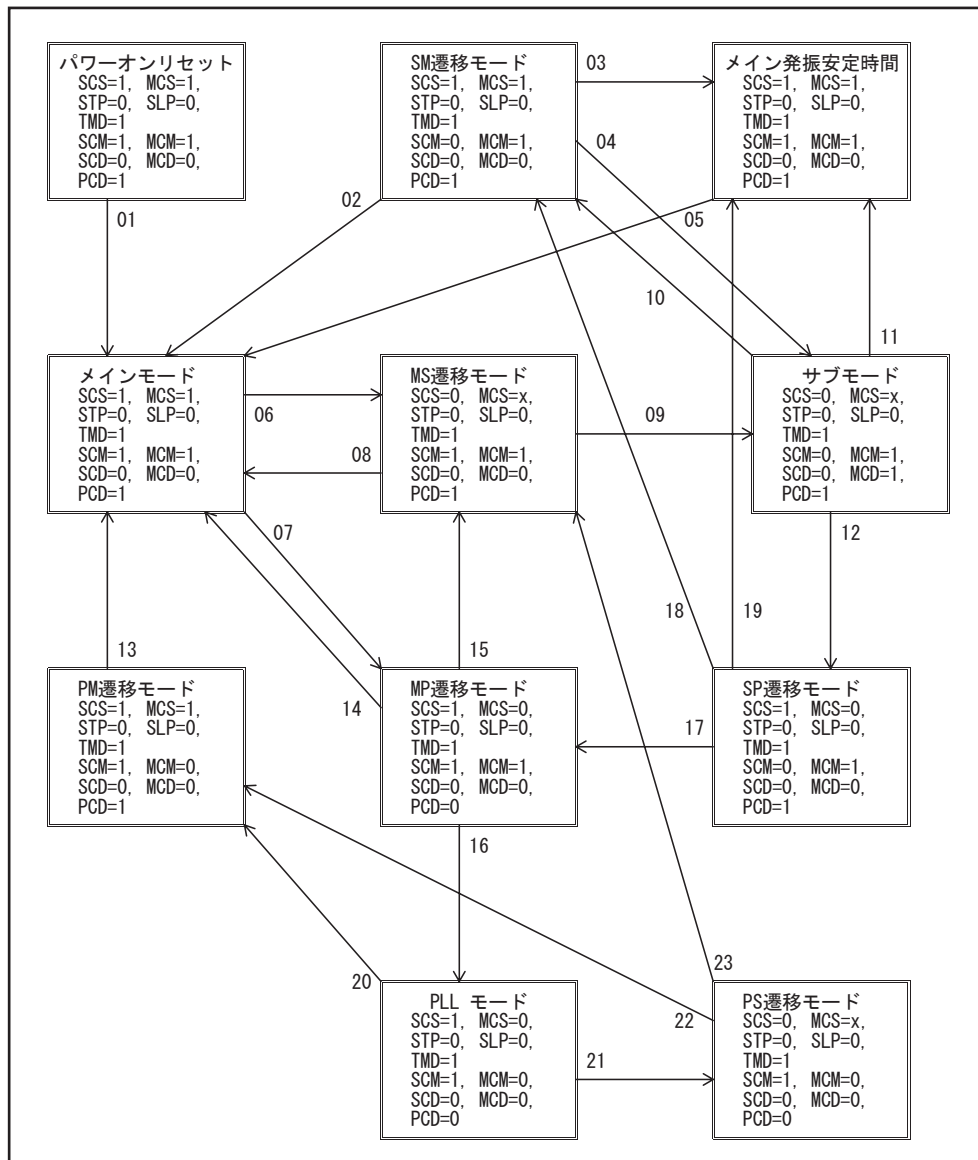


図 6.3-2 低消費電力モード状態遷移図 (2 系統品) B

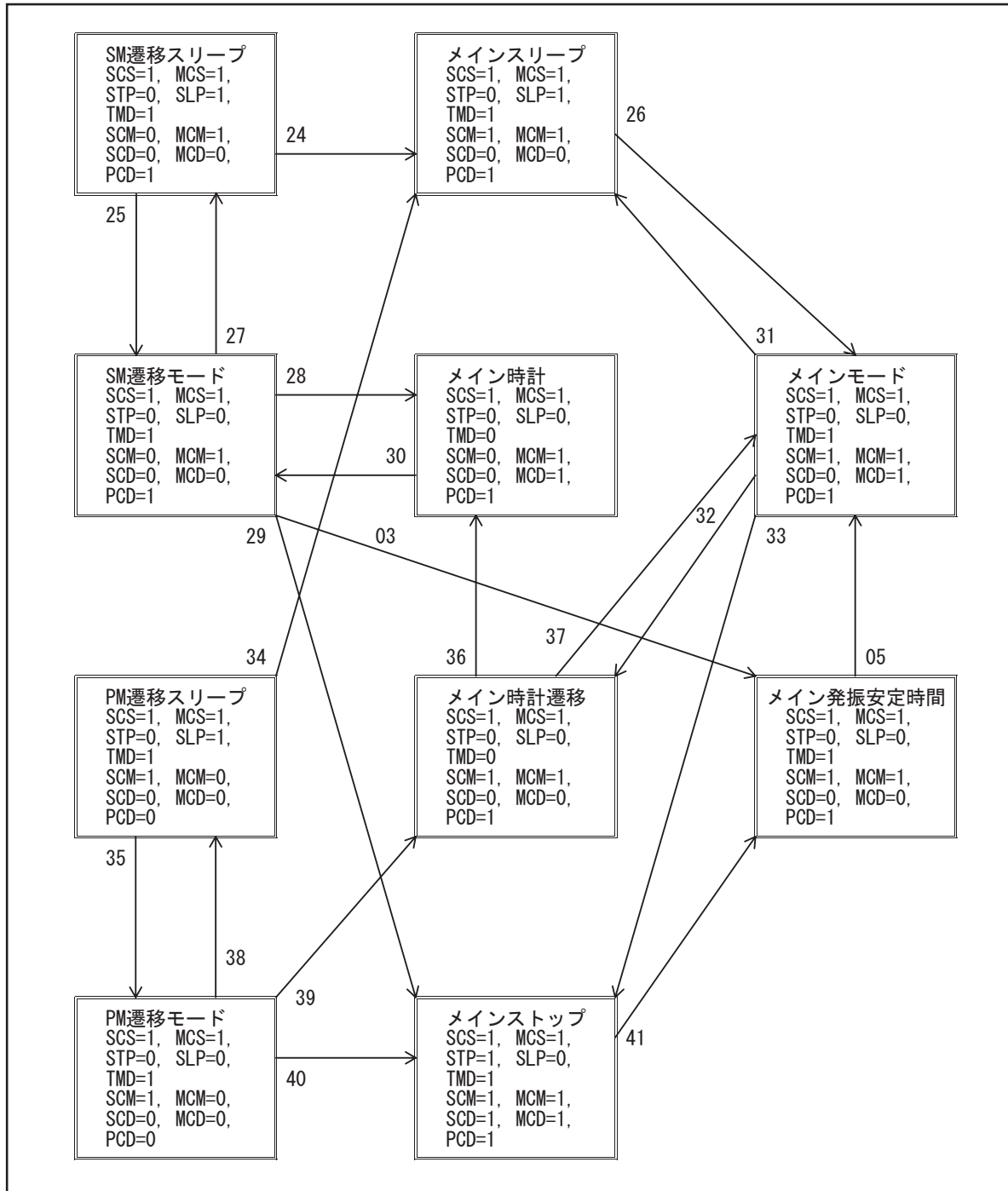


図 6.3-3 低消費電力モード状態遷移図 (2 系統品)C

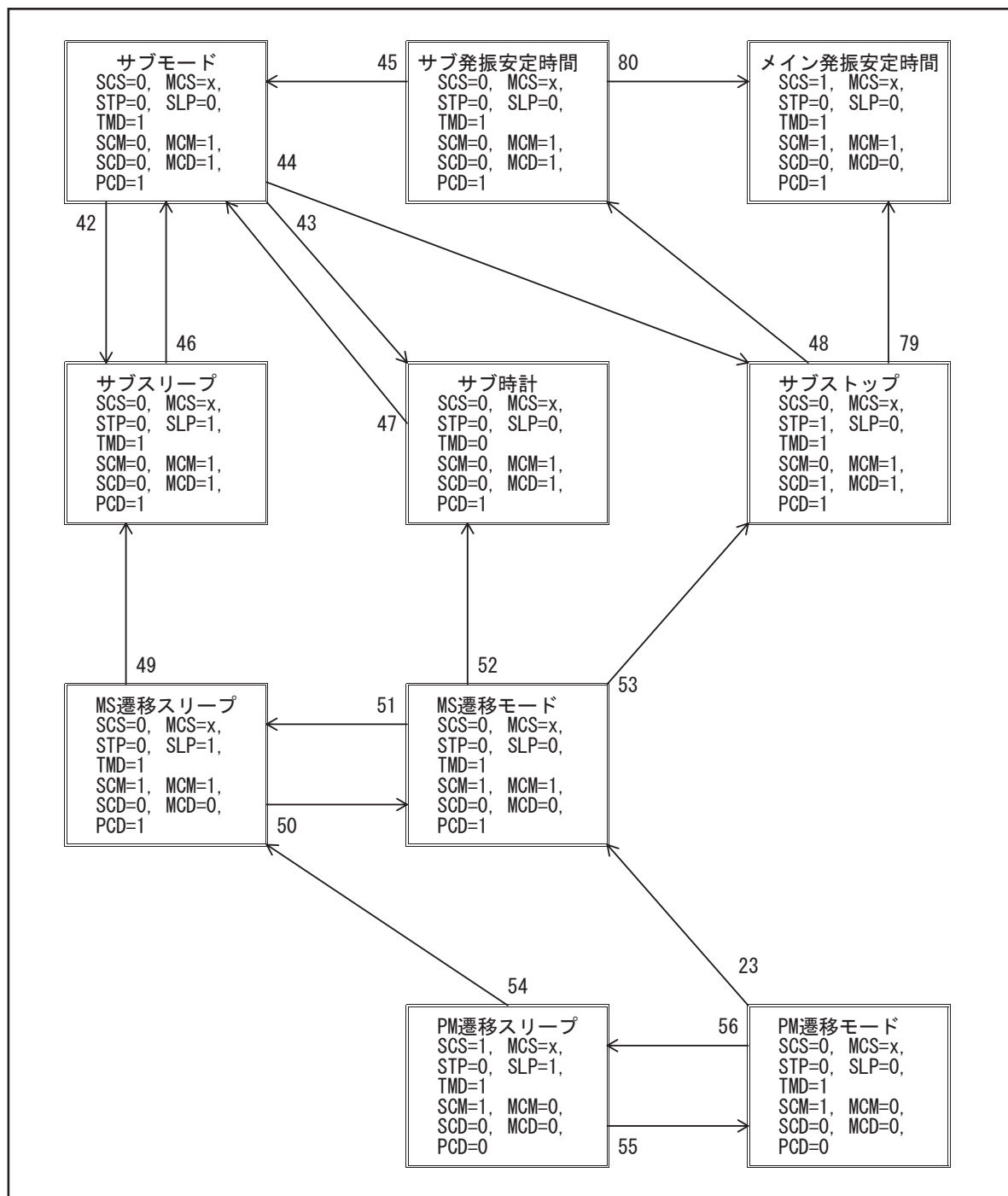
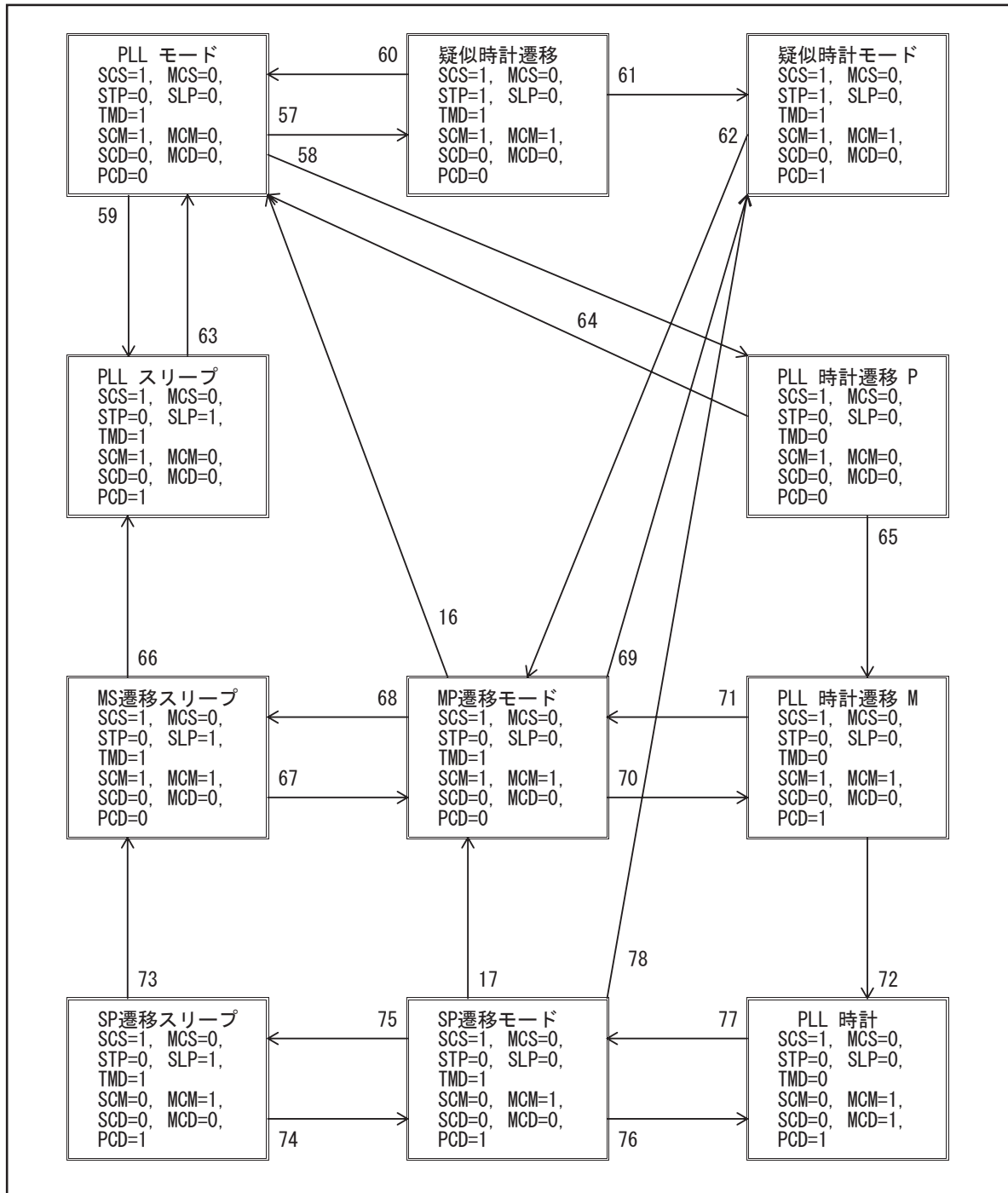


図 6.3-4 低消費電力モード状態遷移図 (2 系統品)D



< 注意事項 >

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。切り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り換えを行った場合、切り換わらない場合があります。

■ 低消費電力モード状態遷移図 (1 系統品)

図 6.3-5 低消費電力モード状態遷移図 (1 系統品) 1

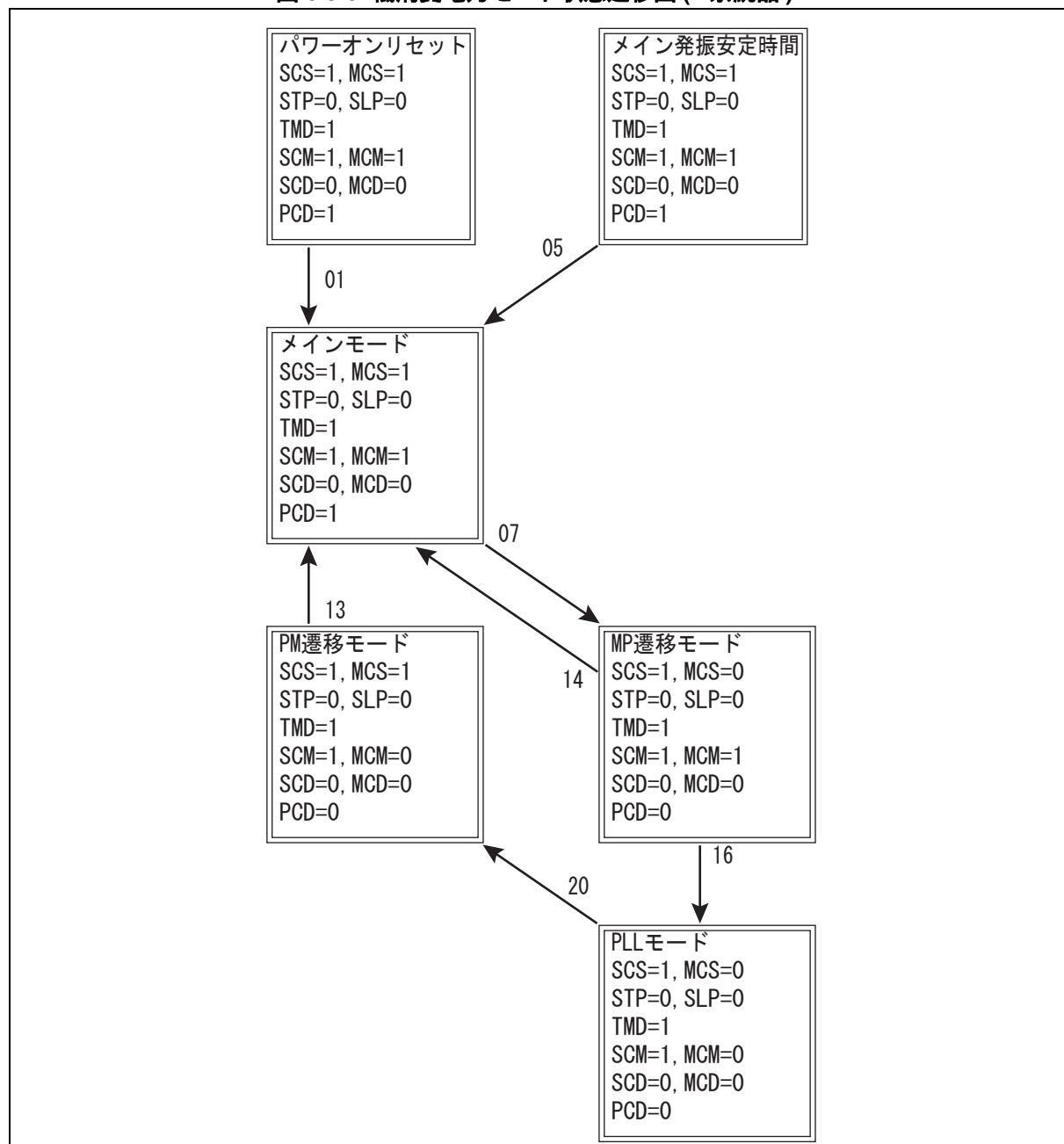


図 6.3-6 低消費電力モード状態遷移図 (1 系統品) 2

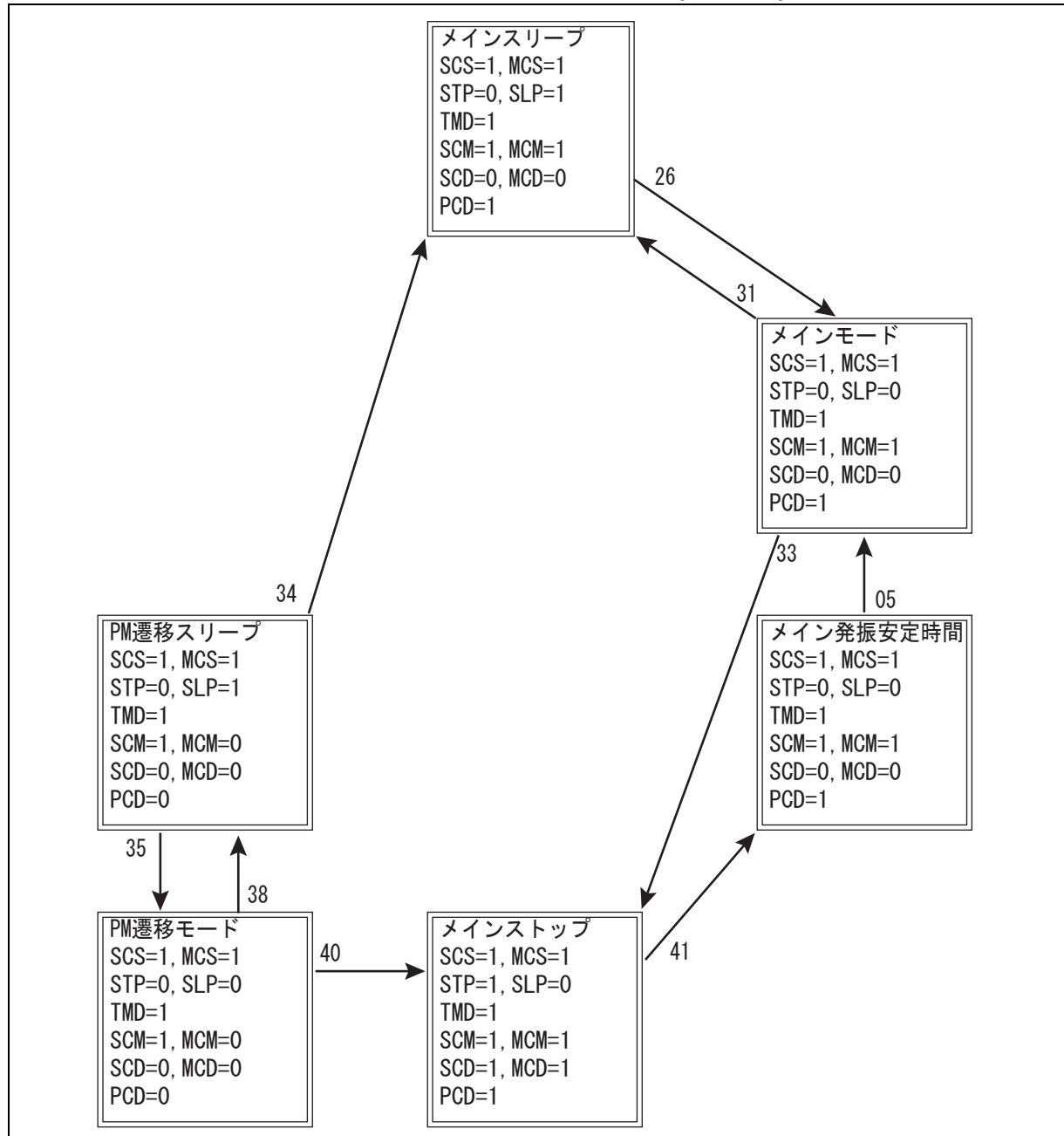
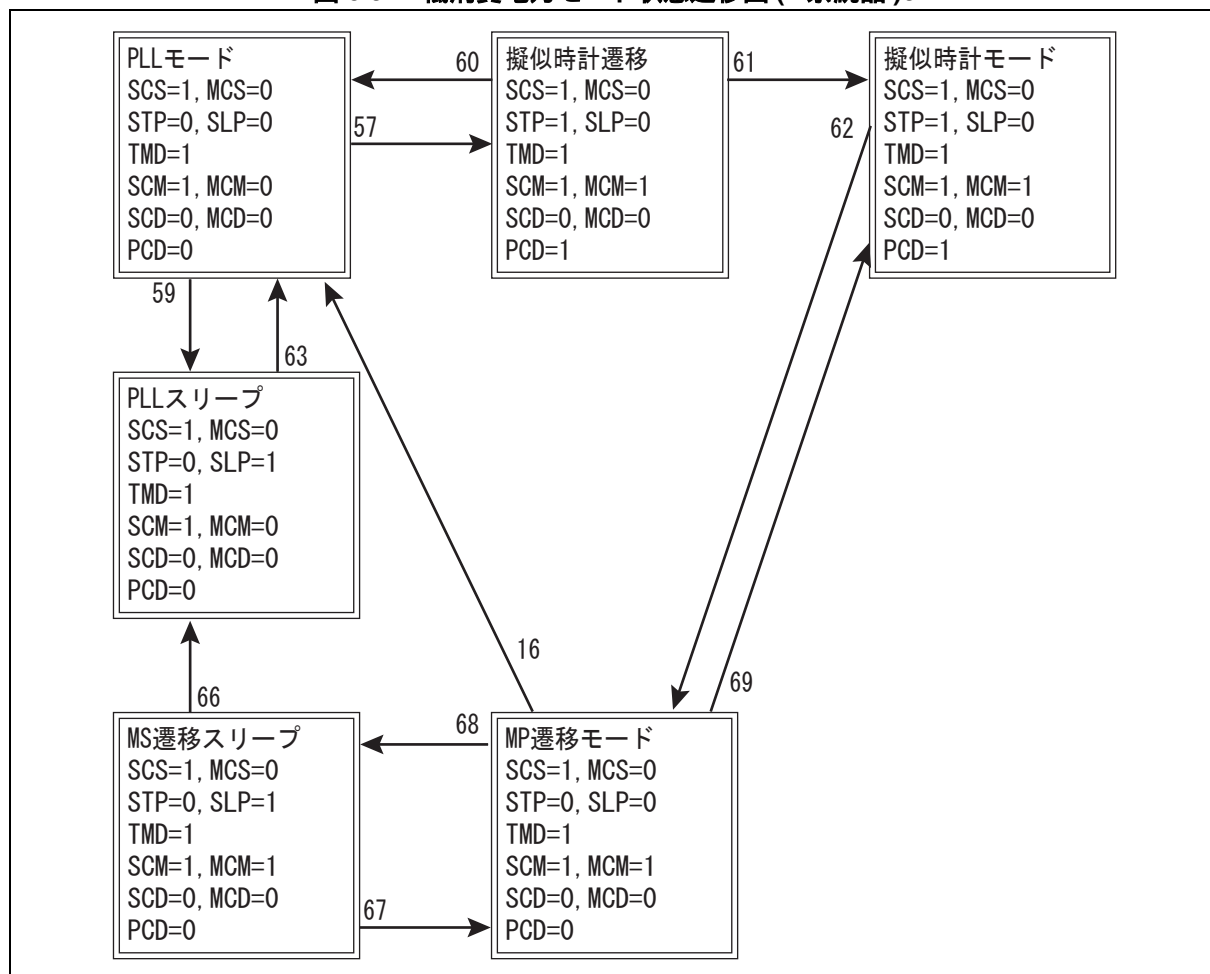


図 6.3-7 低消費電力モード状態遷移図 (1 系統品) 3



< 注意事項 >

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。切り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り換えを行った場合、切り換わらない場合があります。

第7章

メモリアクセスモード

メモリアクセスモードの機能と動作について説明します。

7.1 メモリアクセスモードの概要

7.2 外部メモリアクセス (バス端子制御回路)

7.3 外部メモリアクセス制御信号の動作

7.1 メモリアクセスモードの概要

F²MC-16LX では、アクセス方式、アクセス領域、テスト方式について、各種のモードがあります。このモジュールでは以下の分類が適用されます。

■ メモリアクセスモードの概要

表 7.1-1 モード端子とモード

動作モード	バスモード	アクセスモード
RUN	シングルチップ	-
	内 ROM 外バス	8 ビット
		16 ビット
	外 ROM 外バス	8 ビット
		16 ビット
フラッシュプログラミング	-	-

● 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、MD_x モード設定用端子とモードデータ内の Ex ビットで指定します。動作モードを選択することで、通常動作を行ったり、内部テストプログラムができます。

● 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、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

アクセスモードとは、外部データバス幅を制御するモードを示すもので、MD_x モード設定用端子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で、外部データバスを 8 ビット長あるいは 16 ビット長を指定します。

7.1.1 モード端子

MD2 ~ MD0 の3本の外部端子を組み合わせて、表 7.1-2 の動作を指定できます。

■ モード端子

表 7.1-2 モード端子とモード

モード端子設定			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
MD2	MD1	MD0				
0	0	0	外部ベクタモード 0	外部	8 ビット	
0	0	1	外部ベクタモード 1	外部	16 ビット	リセットベクタ 16 ビット バス幅アクセス
0	1	0	指定禁止			
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	指定禁止			
1	0	1				
1	1	0	フラッシュシリアル書込み*	-	-	-
1	1	1	フラッシュメモリ	-	-	パラレルライタ使用時の モード

*: フラッシュメモリのシリアル書込みは、モード端子の設定だけでは書き込みできません。ほかの端子設定も必要です。詳細は、「第 25 章 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) シリアル書込み接続例」を参照してください。

7.1.2 モードデータ

モードデータは、主記憶 "FFFFDF_H" に置かれ、CPU の動作を制御するために使用します。リセットシーケンス実行中にこのデータを取り込み、デバイス内部のモードレジスタに格納します。モードレジスタの値を変更できるのはリセットシーケンスだけです。

本レジスタによる設定はリセットシーケンス以降に有効となります。

予約ビットは、必ず "0" を設定してください。

■ モードデータ

モードデータのビット構成を図 7.1-1 に示します。

図 7.1-1 モードデータ構成

	7	6	5	4	3	2	1	0	ビット No.
アドレス : FFFFDF _H	M1	M0	予約	予約	S0	予約	予約	予約	

【bit 7, bit 6】M1, M0(バスモード設定ビット)

M1, M0 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。M1, M0 ビットと、機能の関係は、表 7.1-3 のようになっています。

表 7.1-3 M1, M0(バスモード設定ビット) の機能

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

【bit 3】S0(モード設定ビット)

S0 は、リセットシーケンス終了後のバスモード、アクセスモードを指定するビットです。S0 ビットと機能の関係は、表 7.1-4 のようになっています。

表 7.1-4 S0(モード設定ビット) の機能

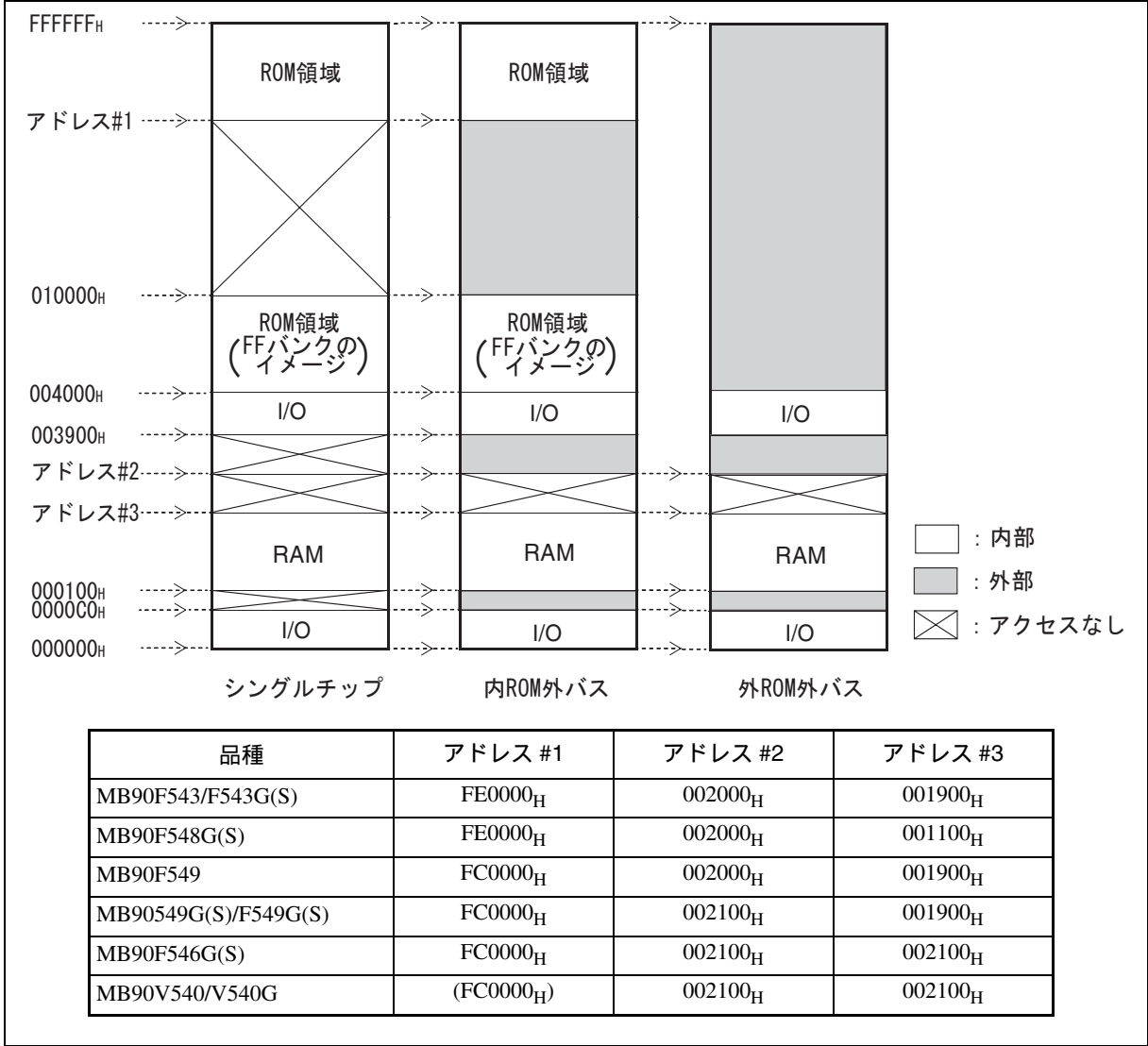
S0	機能
0	外部データバス 8 ビットモード
1	外部データバス 16 ビットモード

7.1.3 バスモード別メモリ空間

バスモードの指定による，アクセス領域と物理アドレスの対応を図 7.1-2 に示します。

■ バスモード別メモリ空間

図 7.1-2 各バスモードにおけるアクセス領域と物理アドレスの関係



■ 推奨設定

表 7.1-5 に、モード端子とモードデータの推奨設定例を示します。

表 7.1-5 モード端子とモードデータの推奨設定例

設定例	MD2	MD1	MD0	M1	M0	S0
シングルチップ	0	1	1	0	0	x
内 ROM 外バスモード・16 ビットバス	0	1	1	0	1	1
内 ROM 外バスモード・8 ビットバス	0	1	1	0	1	0
外 ROM 外バスモード・16 ビットバス・ベクタ 16 バス幅	0	0	1	1	0	1
外 ROM 外バスモード・8 ビットバス	0	0	0	1	0	0

外部端子には各種モードに異なる信号機能があります。

表 7.1-6 各種モードの外部端子の機能

端子名	機能			
	シングルチップ	外部バス拡張		フラッシュ プログラミング
		8 ビット	16 ビット	
P07 ~ P00	ポート	AD07 ~ AD00		D07 ~ D00
P17 ~ P10		A15 ~ A08	AD15 ~ AD08	A15 ~ A08
P27 ~ P20		A23 ~ A16*		A07 ~ A00
P30		ALE		A16
P31		\overline{RD}^*		\overline{CE}
P32		\overline{WR}^*	\overline{WRL}^*	\overline{OE}
P33		ポート	\overline{WRH}^*	\overline{PGM}
P34		HRQ*		未使用
P35		\overline{HAK}^*		
P36		RDY*		
P37		CLK*		

*：アドレス上位出力端子と、 $\overline{WRL}/\overline{WR}$ 、 \overline{WRH} 、HRQ、 \overline{HAK} 、RDY、CLK 端子は、機能選択によりポートとして使用することが可能です。
詳細は、「7.2 外部メモリアクセス（バス端子制御回路）」を参照してください。

7.2 外部メモリアクセス (バス端子制御回路)

外部バス端子制御回路は、CPU のアドレス / データバスを外部に拡張するための外部バス端子を制御します。

■ 外部メモリアクセス (外部バス端子制御回路)

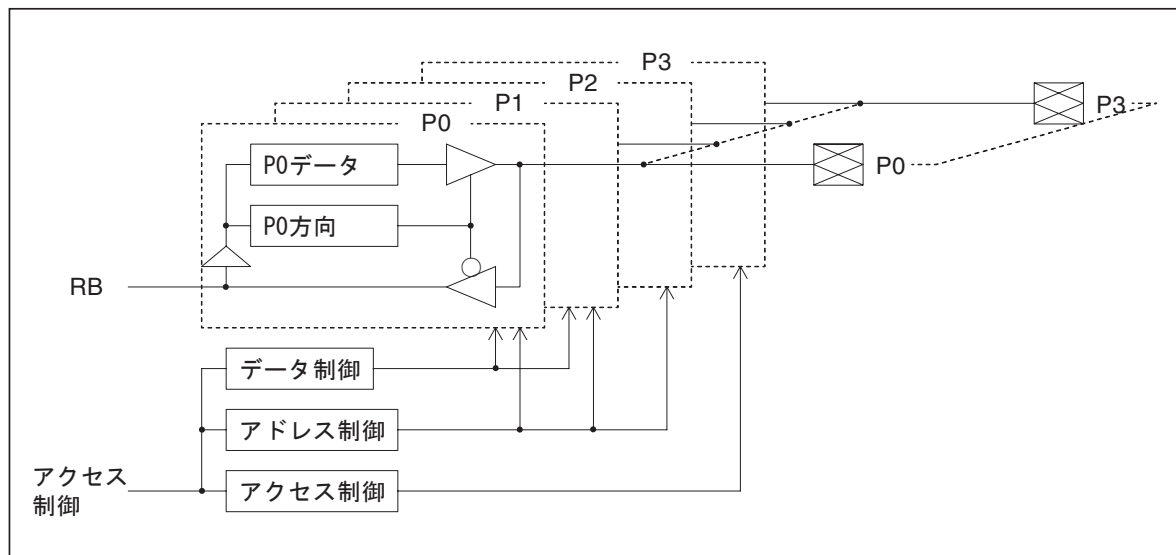
MB90540/545 デバイス外部のメモリ / 周辺をアクセスするためには、以下に示すようなアドレス / データ / 制御信号を使用します。

- CLK(P37) : マシンサイクルクロック (KBP) 出力端子です。
- RDY(P36) : 外部レディ入力端子です。
- $\overline{\text{WRH}}$ (P33) : データバス上位 8 ビットのライト信号です。
- $\overline{\text{WRL}}/\overline{\text{WR}}$ (P32) : データバス下位 8 ビットのライト信号または 8 ビットアクセス時のデータバス 8 ビットのライト信号です。
- $\overline{\text{RD}}$ (P31) : リード信号です。
- ALE(P30) : アドレスラッチイネーブル信号です。

外部バス端子制御回路は外部バス端子の制御に使用され、CPU アドレス / データバスの外部拡張を可能にします。

■ 外部メモリアクセスのブロックダイアグラム

図 7.2-1 外部バス制御器



7.2.1 外部メモリアクセス (外部バス端子制御回路) のレジスタ

外部メモリアクセス (外部バス端子制御回路) には , 次の 3 種類のレジスタがあります。

- 自動レディ機能選択レジスタ
- 外部アドレス出力制御レジスタ
- バス制御信号選択レジスタ

■ 外部メモリアクセスのレジスタ一覧

外部メモリアクセスのビット構成を図 7.2-2 に示します。

図 7.2-2 外部メモリアクセス (外部バス端子制御回路) のレジスタ一覧

自動レディ機能選択レジスタ									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 0000A5 _H	IOR1	IOR0	HMR1	HMR0			LMR1	LMR0	ARSR
リード / ライト	(W)	(W)	(W)	(W)	(-)	(-)	(W)	(W)	
初期値	(0)	(0)	(1)	(1)	(-)	(-)	(0)	(0)	
外部アドレス出力制御レジスタ									
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000A6 _H	E23	E22	E21	E20	E19	E18	E17	E16	HACR
リード / ライト	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
バス制御信号選択レジスタ									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 0000A7 _H	CKE	RYE	HDE	IOBS	HMBS	WRE	LMBS		ECSR
リード / ライト	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(-)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(-)	

7.2.2 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) は、外部アクセス時、領域ごとのメモリアクセスの自動ウェイト時間を設定します。

■ 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 7.2-3 に示します。

図 7.2-3 自動レディ機能選択レジスタ (ARSR) の構成

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 0000A5 _H	IOR1	IOR0	HMR1	HMR0			LMR1	LMR0	ARSR
リード/ライト	(W)	(W)	(W)	(W)	(-)	(-)	(W)	(W)	
初期値	(0)	(0)	(1)	(1)	(-)	(-)	(0)	(0)	

【bit 15, bit 14】 IOR1, IOR0

IOR1, IOR0 ビットでは, "0000C0_H" ~ "0000FF_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指定します。IOR1, IOR0 ビットの組合せで, 表 7.2-1 に示すような設定になります。

表 7.2-1 IOR1, IOR0(自動ウェイト機能指定ビット) の機能

IOR1	IOR0	機能
0	0	自動ウェイト禁止 [初期値]
0	1	外部アクセス時, 1 サイクルの自動ウェイトが入る
1	0	外部アクセス時, 2 サイクルの自動ウェイトが入る
1	1	外部アクセス時, 3 サイクルの自動ウェイトが入る

【bit 13, bit 12】 HMR1, HMR0

HMR1, HMR0 では, "800000_H" ~ "FFFFFF_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指定します。HMR1, HMR0 ビットの組合せで, 表 7.2-2 に示すような設定になります。

表 7.2-2 HMR1, HMR0(自動ウェイト機能指定ビット) の機能

HMR1	HMR0	機能
0	0	自動ウェイト禁止
0	1	外部アクセス時, 1 サイクルの自動ウェイトが入る
1	0	外部アクセス時, 2 サイクルの自動ウェイトが入る
1	1	外部アクセス時, 3 サイクルの自動ウェイトが入る [初期値]

【bit 9, bit 8】 LMR1, LMR0

LMR1, LMR0 では, "002000_H" ~ "7FFFFFF_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指定します。LMR1, LMR0 ビットの組合せで, 表 7.2-3 に示すような設定になります。

表 7.2-3 LMR1, LMR0(自動ウェイト機能指定ビット)の機能

LMR1	LMR0	機能 [初期値]
0	0	自動ウェイト禁止 [初期値]
0	1	外部アクセス時, 1 サイクルの自動ウェイトが入る
1	0	外部アクセス時, 2 サイクルの自動ウェイトが入る
1	1	外部アクセス時, 3 サイクルの自動ウェイトが入る

7.2.3 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 (HACR) 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図 7.2-4 に示すように制御します。

■ 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 (HACR) のビット構成を図 7.2-4 に示します。

図 7.2-4 外部アドレス出力制御レジスタ (HACR) の構成

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000A6 _H	E23	E22	E21	E20	E19	E18	E17	E16	HACR
リード / ライト	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

【bit 7 ~ bit 0】E23 ~ E16

HACR レジスタは外部回路へのアドレス (A23 ~ A16) の出力を制御します。8 ビットが A23 ~ A16 のアドレスビットに対応し、下記のようにアドレス出力端子を制御します。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することができません。その場合、本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。このレジスタの全ビットは、すべて書込み専用で、読出しでは "1" となります。

表 7.2-4 外部アドレス出力制御レジスタ (E23 ~ E16 ビット) の機能

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。

7.2.4 バス制御信号選択レジスタ (ECSR)

バス制御信号選択レジスタ (ECSR) 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。デバイスがシングルチップモード時にはアクセスすることができません。その場合、本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。バス制御信号選択レジスタのビットは、すべて書込み専用で、読出しでは "1" となります。

■ バス制御信号選択レジスタ (ECSR)

バス制御信号選択レジスタ (ECSR) のビット構成を図 7.2-5 に示します。

図 7.2-5 バス制御信号選択レジスタ (ECSR) の構成

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 0000A7 _H	CKE	RYE	HDE	IOBS	HMBS	WRE	LMBS		ECSR
リード / ライト	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(-)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(-)	

【bit 15】CKE

CKE ビットでは、外部クロック信号端子 (CLK) の出力を表 7.2-5 に示すように制御します。

表 7.2-5 CKE(外部クロック (CLK) 出力制御ビット) の機能

0	I/O ポート (P37) 動作 (クロック出力禁止) [初期値]
1	クロック信号 (CLK) 出力許可

【bit 14】RYE

RYE ビットでは、外部レディ (RDY) 信号端子の入力を、表 7.2-6 に示すように制御します。

表 7.2-6 RYE(外部レディ (RDY) 入力制御ビット) の機能

0	I/O ポート (P36) 動作 (外部 RDY 入力禁止) [初期値]
1	外部レディ (RDY) 入力許可

【bit 13】HDE

HDE ビットは、ホールド関係の信号の入出力許可を指定します。HDE ビットの設定により、ホールド要求入力信号 (HRQ) とホールドアクノリッジ出力信号 ($\overline{\text{HAK}}$) の 2 本を表 7.2-7 に示すように制御します。

表 7.2-7 HDE(ホールド関連端子の入出力許可指定ビット) の機能

0	I/O ポート (P35, P34) 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

【bit 12】IOBS

IOBS ビットでは、外部データバス 16 ビットモード時に、"0000C0_H" ~ "0000FF_H" の領域に対する外部アクセスを行ったときのバス幅を指定します。本ビットの設定により、表 7.2-8 に示すように制御します。

表 7.2-8 IOBS(バス幅指定ビット)

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

【bit 11】HMBS

HMBS ビットは、外部データバス 16 ビットモード時に、"800000_H" ~ "FFFFFF_H" の領域に対する外部アクセスを行ったときのバス幅を指定します。本ビットの設定により、表 7.2-9 に示すように制御します。

表 7.2-9 HMBS(バス幅指定ビット)の機能

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

【bit 10】WRE

WRE ビットは、外部ライト信号 (外部データバス 16 ビットモードの場合は $\overline{\text{WRH}}/\overline{\text{WRL}}$ 両端子、外部データバス 8 ビットモードの場合は $\overline{\text{WR}}$ 端子) の出力を、表 7.2-10 に示すように制御します。

外部データバス 8 ビットモードの場合は、本ビットの設定値に関係なく、P33 は I/O ポートとして機能します。

表 7.2-10 WRE(外部ライト信号出力制御ビット)の機能

0	I/O ポート (P33, P32) 動作 (ライト信号出力禁止) [初期値]
1	ライトストローブ信号 ($\overline{\text{WRH}}/\overline{\text{WRL}}$ または $\overline{\text{WR}}$ のみ) の出力許可

【bit 9】LMBS

LMBS ビットは、外部データバス 16 ビットモード時に、"002000_H" ~ "7FFFFFF_H" の領域に対する外部アクセスを行ったときのバス幅を指定します。本ビットの設定により、表 7.2-11 のように制御します。

表 7.2-11 LMBS(バス幅指定ビット)の機能

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 ビットにて \overline{WR} , \overline{WRH} , \overline{WRL} 機能を許可する場合は, P33, P32 を入力モードにしてください (DDR3 レジスタの bit3, bit2 は "0" にしてください)。
 - 外部データバス 8 ビットモード時, WRE ビットにて \overline{WR} 機能を許可する場合は, P32 を入力モードにしてください (DDR3 レジスタの bit2 は "0" にしてください)。
 - また, RYE, HDE ビットにて RDY, HRQ 信号入力を許可した場合でも, そのポートの I/O ポート機能は有効となります。そのため, そのポートに対応する DDR3 レジスタに, 必ず "0" (入力モード) を書き込みます。
-

7.3 外部メモリアクセス制御信号の動作

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は3サイクルで行います。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リード/ライトするための機能です。

■ 外部メモリアクセス制御信号

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、ECSR レジスタの HMBS/LMBS/IOBS ビットで指定することになります。

なお、アドレス出力と ALE 信号のアサート出力のみを行い、 $\overline{\text{RD}}/\overline{\text{WRL}}/\overline{\text{WRH}}/\overline{\text{WR}}$ 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。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。

図 7.3-1 外部データバス 8 ビットモードのアクセスタイミングチャート

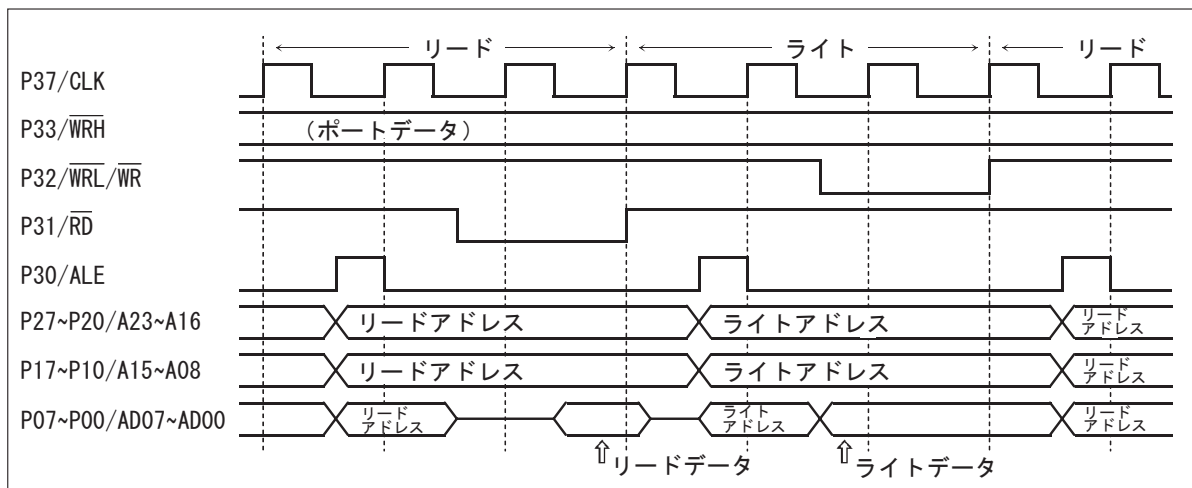
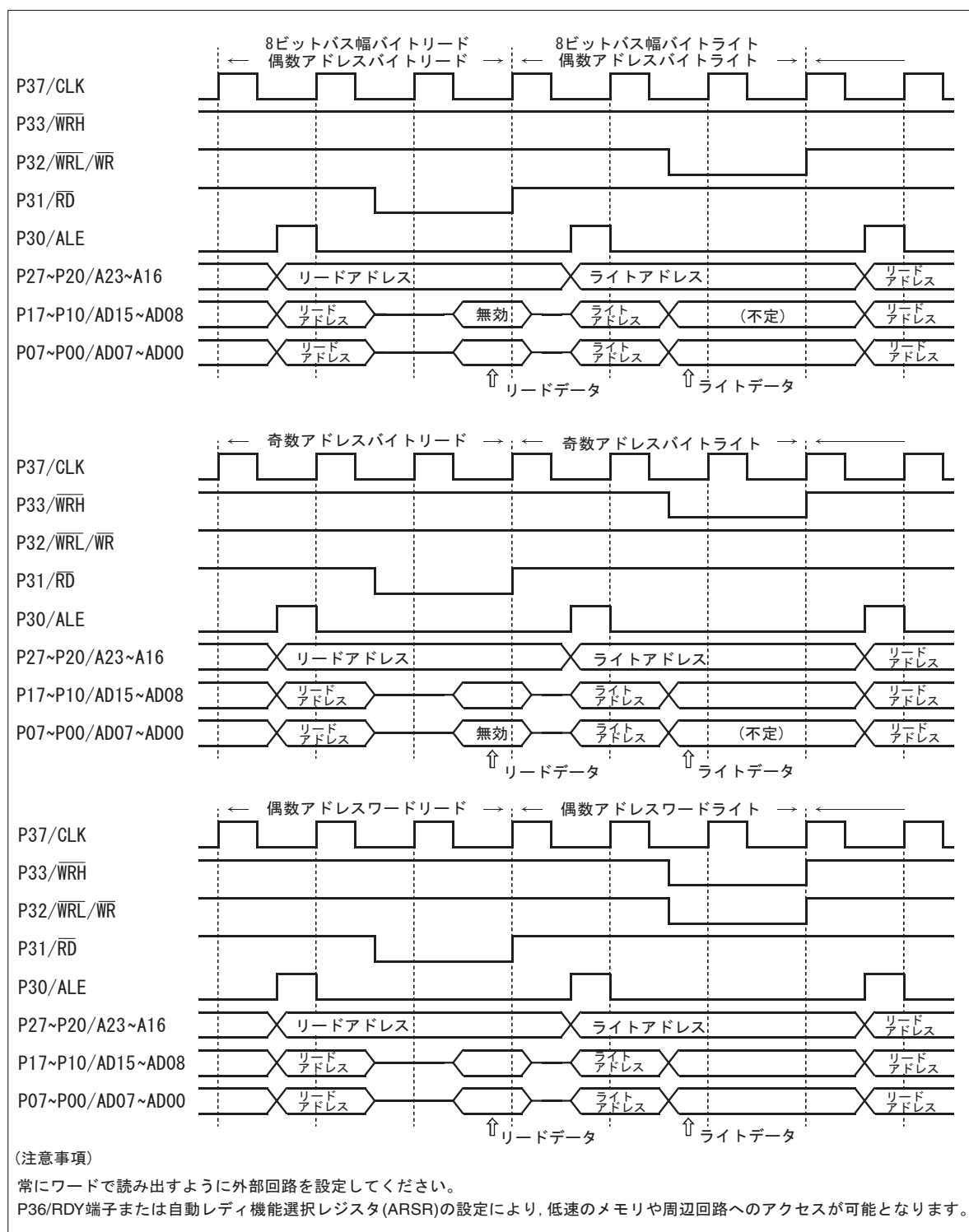


図 7.3-2 外部データバス 16 ビットモードのアクセスタイミングチャート
(16 ビットバス幅アクセス, 8 ビットバス幅アクセス時)



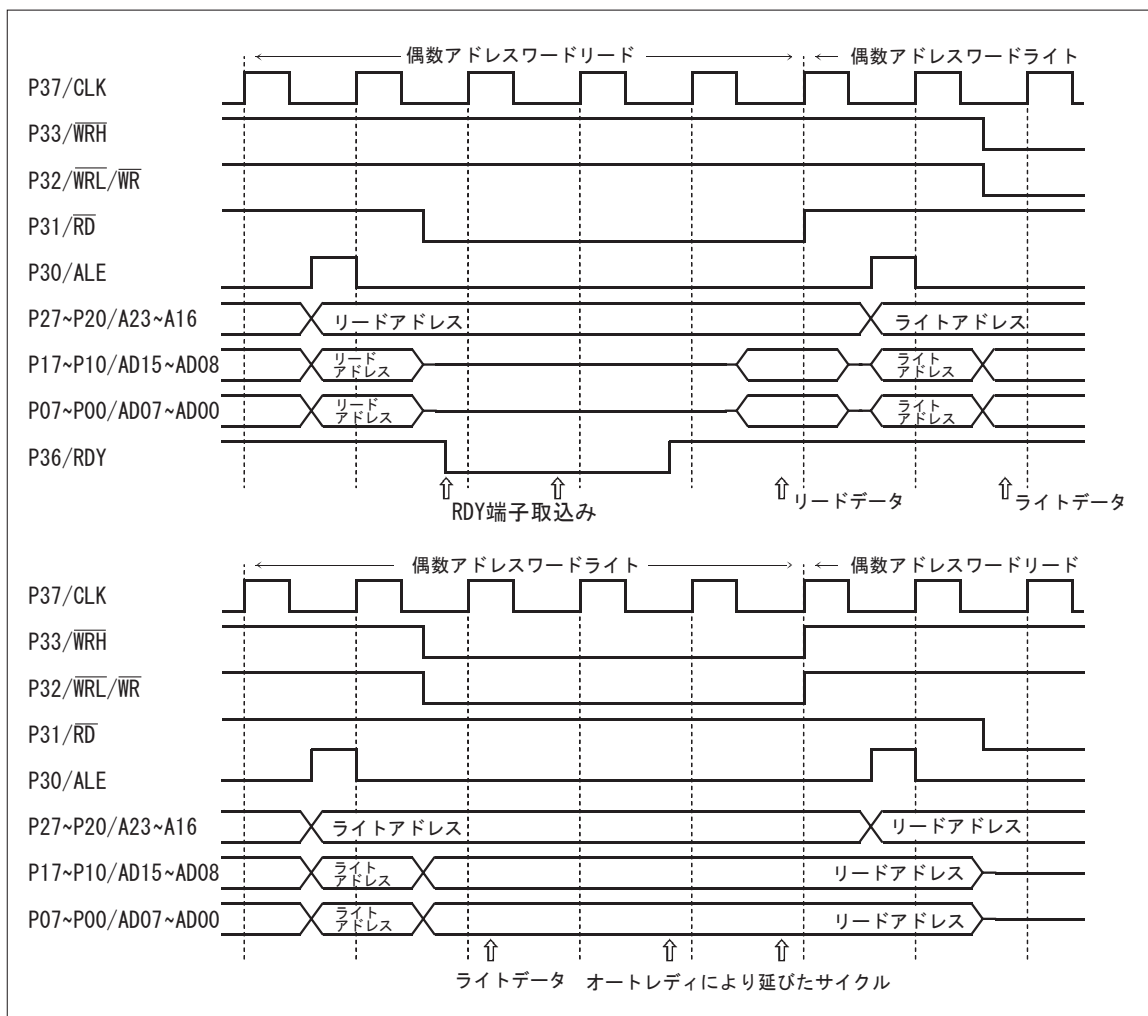
7.3.1 レディ機能

P36/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 (ECSR) 中の RYE ビットが "1" に設定されている場合、外部回路へのアクセス時に P36/RDY 信号に "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可能です。

■ レディ機能

図 7.3-3 レディ機能タイミングチャート



MB90540/545 は、外部メモリアクセス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 "002000_H" ~ "7FFFFF_H" 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と、アドレス "800000_H" ~ "FFFFFF_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、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。

さらに、MB90540/545 は I/O 用のオートレディ機能をメモリ用とは独立に内蔵しています。ARSR レジスタの IOR1/IOR0 ビットを "0" に設定すると、アドレス "0000C0_H" ~ "0000FF_H" 間の外部領域へのアクセス時に、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 ECSR 中の RYE ビットが "1" に設定されている場合、上記オートレディ機能によるウェイトサイクル終了後に P36/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。

7.3.2 ホールド機能

バス制御信号選択レジスタ (ECSR) 中の HDE ビットが "1" に設定されている場合，P34/HRQ, P35/ $\overline{\text{HAK}}$ 両端子による外部アドレスホールド機能が有効となります。

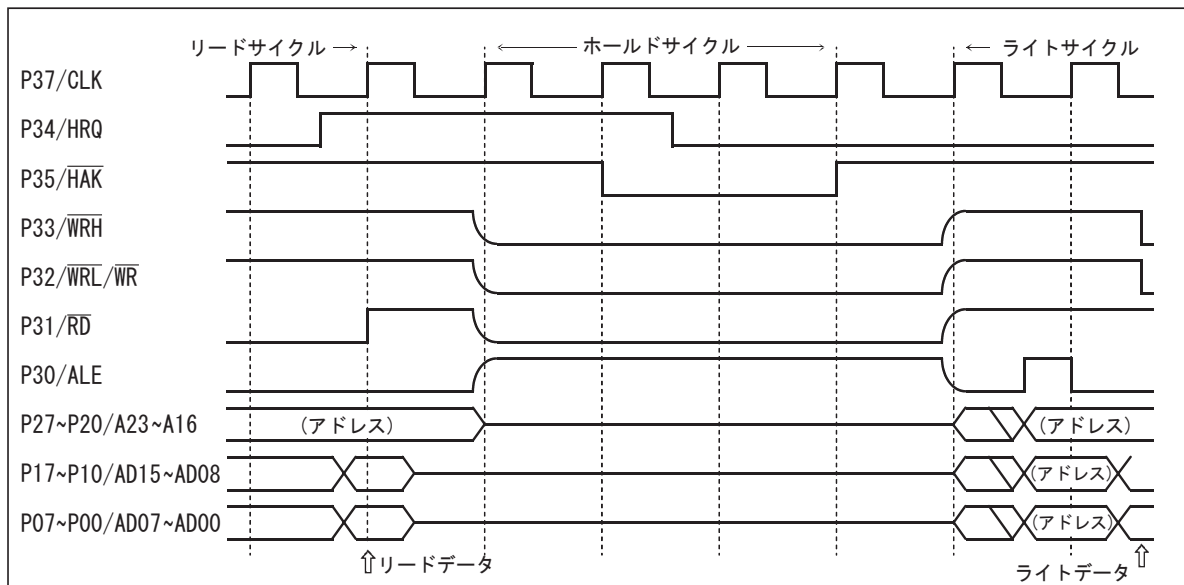
■ ホールド機能

P34/HRQ 端子に "H" レベルを入力すると，CPU の命令の終了時（ストリング命令の場合は "1" エLEMENTデータの処理の終了時）にてホールド状態となり，P35/ $\overline{\text{HAK}}$ 端子より "L" レベル信号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。

- アドレス出力 : P27/A23 ~ P20/A16
- データ入出力 : P17/AD15 ~ P00/AD00
- バス制御信号 : P30/ALE, P31/ $\overline{\text{RD}}$, P32/ $\overline{\text{WRL}}/\overline{\text{WR}}$, P33/ $\overline{\text{WRH}}$

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。P34/HRQ 端子に "L" レベル信号を入力すると，P35/ $\overline{\text{HAK}}$ 端子は "H" レベル出力となり，外部端子状態が復活してバスは動作を再開します。STOP 状態においては，ホールド要求入力を受け付けません。

図 7.3-4 ホールドタイミング



第8章

I/O ポート

I/O ポートの機能について説明します。

8.1 I/O ポート

8.2 I/O ポートのレジスター一覧

8.1 I/O ポート

それぞれのポートの各端子は、対応する周辺が端子を用いない設定となっているとき、ポート方向レジスタ (DDR0 ~ DDRA) の設定により入力、出力の指定を行うことができます。

■ I/O ポートの概要

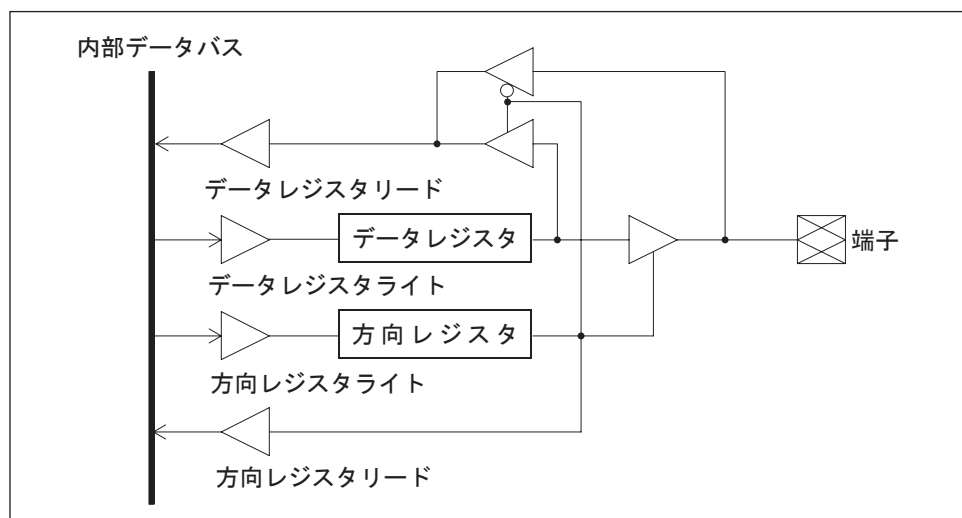
端子が入力と指定されている場合は、端子の論理レベルが読み出されます。端子が出力と指定されている場合は、データレジスタ値が読み出されます。これは、リードモディファイライト系命令の読み出しでも同じです。

端子がほかの周辺機能の出力として使用されている場合は、データレジスタの値に関係なく、端子の論理レベルが読み出されます。

通常は、ポートを出力に設定するのに先立って、リードモディファイライト系の命令をデータレジスタの設定に使用しない、および周辺機能の出力を禁止することをお勧めします。これは、この場合のリードモディファイライト系命令がレジスタ値ではなく、ポートの論理レベルを読み出すことになるからです。

図 8.1-1 に、I/O ポートのブロックダイアグラムを示します。

図 8.1-1 I/O ポートのブロックダイアグラム



8.2 I/O ポートのレジスタ一覧

I/O ポートの各レジスタのビット構成を、図 8.2-1 に示します。

■ I/O ポートのレジスタ一覧

I/O ポートのレジスタのビット構成を図 8.2-1 に示します。

図 8.2-1 I/O ポートのレジスタ一覧

ビット No.	7	6	5	4	3	2	1	0	
アドレス：000000 _H	P07	P06	P05	P04	P03	P02	P01	P00	ポートデータレジスタ (PDR0) (ポート 0 用)
アドレス：000001 _H	P17	P16	P15	P14	P13	P12	P11	P10	ポートデータレジスタ (PDR1) (ポート 1 用)
アドレス：000002 _H	P27	P26	P25	P24	P23	P22	P21	P20	ポートデータレジスタ (PDR2) (ポート 2 用)
アドレス：000003 _H	P37	P36	P35	P34	P33	P32	P31	P30	ポートデータレジスタ (PDR3) (ポート 3 用)
アドレス：000004 _H	P47	P46	P45	P44	P43	P42	P41	P40	ポートデータレジスタ (PDR4) (ポート 4 用)
アドレス：000005 _H	P57	P56	P55	P54	P53	P52	P51	P50	ポートデータレジスタ (PDR5) (ポート 5 用)
アドレス：000006 _H	P67	P66	P65	P64	P63	P62	P61	P60	ポートデータレジスタ (PDR6) (ポート 6 用)
アドレス：000007 _H	P77	P76	P75	P74	P73	P72	P71	P70	ポートデータレジスタ (PDR7) (ポート 7 用)
アドレス：000008 _H	P87	P86	P85	P84	P83	P82	P81	P80	ポートデータレジスタ (PDR8) (ポート 8 用)
アドレス：000009 _H	P97	P96	P95	P94	P93	P92	P91	P90	ポートデータレジスタ (PDR9) (ポート 9 用)
アドレス：00000A _H								PA0	ポートデータレジスタ (PDRA) (ポート A 用)
ビット No.	7	6	5	4	3	2	1	0	
アドレス：000010 _H	D07	D06	D05	D04	D03	D02	D01	D00	ポート方向レジスタ (DDR0) (ポート 0 用)
アドレス：000011 _H	D17	D16	D15	D14	D13	D12	D11	D10	ポート方向レジスタ (DDR1) (ポート 1 用)
アドレス：000012 _H	D27	D26	D25	D24	D23	D22	D21	D20	ポート方向レジスタ (DDR2) (ポート 2 用)
アドレス：000013 _H	D37	D36	D35	D34	D33	D32	D31	D30	ポート方向レジスタ (DDR3) (ポート 3 用)
アドレス：000014 _H	D47	D46	D45	D44	D43	D42	D41	D40	ポート方向レジスタ (DDR4) (ポート 4 用)
アドレス：000015 _H	D57	D56	D55	D54	D53	D52	D51	D50	ポート方向レジスタ (DDR5) (ポート 5 用)
アドレス：000016 _H	D67	D66	D65	D64	D63	D62	D61	D60	ポート方向レジスタ (DDR6) (ポート 6 用)
アドレス：000017 _H	D77	D76	D75	D74	D73	D72	D71	D70	ポート方向レジスタ (DDR7) (ポート 7 用)
アドレス：000018 _H	D87	D86	D85	D84	D83	D82	D81	D80	ポート方向レジスタ (DDR8) (ポート 8 用)
アドレス：000019 _H	D97	D96	D95	D94	D93	D92	D91	D90	ポート方向レジスタ (DDR9) (ポート 9 用)
アドレス：00001A _H								DA0	ポート方向レジスタ (DDRA) (ポート A 用)
ビット No.	7	6	5	4	3	2	1	0	
アドレス：00001C _H	PU07	PU06	PU05	PU04	PU03	PU02	PU01	PU00	ブルアップ制御レジスタ (PUCR0) (ポート 0 用)
アドレス：00001D _H	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10	ブルアップ制御レジスタ (PUCR1) (ポート 1 用)
アドレス：00001E _H	PU27	PU26	PU25	PU24	PU23	PU22	PU21	PU20	ブルアップ制御レジスタ (PUCR2) (ポート 2 用)
アドレス：00001F _H	PU37	PU36	PU35	PU34	PU33	PU32	PU31	PU30	ブルアップ制御レジスタ (PUCR3) (ポート 3 用)
ビット No.	7	6	5	4	3	2	1	0	
アドレス：00001B _H	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0	アナログ入力許可レジスタ (ADER)

8.2.1 ポートデータレジスタ (PDR0 ~ PDRA) (ポート0 ~ ポートA用)

ポートデータレジスタ (PDR0 ~ PDRA) (ポート0 ~ ポートA用) のビット構成の詳細を、図 8.2-2 に示します。

■ ポートデータレジスタ (PDR0 ~ PDRA) (ポート0 ~ ポートA用)

ポートデータレジスタ (PDR0 ~ PDRA) (ポート0 ~ ポートA用) のビット構成を図 8.2-2 に示します。

図 8.2-2 ポートデータレジスタ (PDR0 ~ PDRA) (ポート0 ~ ポートA用)

PDR0 (ポート0用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000000 _H	P07	P06	P05	P04	P03	P02	P01	P00	XXXXXXXX _B R/W*
PDR1 (ポート1用)									
ビット No.	7	6	5	4	3	2	1	0	
アドレス: 000001 _H	P17	P16	P15	P14	P13	P12	P11	P10	XXXXXXXX _B R/W*
PDR2 (ポート2用)									
ビット No.	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 (ポート3用)									
ビット No.	7	6	5	4	3	2	1	0	
アドレス: 000003 _H	P37	P36	P35	P34	P33	P32	P31	P30	XXXXXXXX _B R/W*
PDR4 (ポート4用)									
ビット No.	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 (ポート5用)									
ビット No.	7	6	5	4	3	2	1	0	
アドレス: 000005 _H	P57	P56	P55	P54	P53	P52	P51	P50	XXXXXXXX _B R/W*
PDR6 (ポート6用)									
ビット No.	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 (ポート7用)									
ビット No.	7	6	5	4	3	2	1	0	
アドレス: 000007 _H	P77	P76	P75	P74	P73	P72	P71	P70	XXXXXXXX _B R/W*
PDR8 (ポート8用)									
ビット No.	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 (ポート9用)									
ビット No.	7	6	5	4	3	2	1	0	
アドレス: 000009 _H	P97	P96	P95	P94	P93	P92	P91	P90	XXXXXXXX _B R/W*

(続く)

(続き)

PDRA (ポート A 用)

ビット No.	7	6	5	4	3	2	1	0	
アドレス : 00000A _H								PA0	-----X _B R/W *

* : 入出力サポートの R/W は , メモリへの R/W とは多少動作が異なり , 次のとおりになりますので注意してください。

- 入力モード
リード時 : 対応する端子のレベルが読み出されます。
ライト時 : 出力用のラッチに書き込まれます。
- 出力モード
リード時 : データレジスタラッチの値が読み出されます。
ライト時 : 出力用のラッチに書き込まれ , 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 ~ DDRA) (ポート0 ~ ポートA用)

ポート方向レジスタ (DDR0 ~ DDRA) (ポート0 ~ ポートA用) のビット構成を図 8.2-3 に示します。

■ ポート方向レジスタ (DDR0 ~ DDRA) (ポート0 ~ ポートA用)

ポート方向レジスタ (DDR0 ~ DDRA) (ポート0 ~ ポートA用) のビット構成を図 8.2-3 に示します。

図 8.2-3 ポート方向レジスタ (DDR0 ~ DDRA) (ポート0 ~ ポートA用)

DDR0 (ポート0用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000010 _H	D07	D06	D05	D04	D03	D02	D01	D00	00000000 _B
									R/W
DDR1 (ポート1用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000011 _H	D17	D16	D15	D14	D13	D12	D11	D10	00000000 _B
									R/W
DDR2 (ポート2用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000012 _H	D27	D26	D25	D24	D23	D22	D21	D20	00000000 _B
									R/W
DDR3 (ポート3用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000013 _H	D37	D36	D35	D34	D33	D32	D31	D30	00000000 _B
									R/W
DDR4 (ポート4用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000014 _H	D47	D46	D45	D44	D43	D42	D41	D40	00000000 _B
									R/W
DDR5 (ポート5用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000015 _H	D57	D56	D55	D54	D53	D52	D51	D50	00000000 _B
									R/W
DDR6 (ポート6用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000016 _H	D67	D66	D65	D64	D63	D62	D61	D60	00000000 _B
									R/W
DDR7 (ポート7用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000017 _H	D77	D76	D75	D74	D73	D72	D71	D70	00000000 _B
									R/W
DDR8 (ポート8用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000018 _H	D87	D86	D85	D84	D83	D82	D81	D80	00000000 _B
									R/W
DDR9 (ポート9用)									
ビット No.	7	6	5	4	3	2	1	0	初期値
アドレス: 000019 _H	D97	D96	D95	D94	D93	D92	D91	D90	00000000 _B
									R/W

(続く)

(続き)

DDRA (ポート A 用)

ビット No.

7

6

5

4

3

2

1

0

アドレス : 00001A_H

DA0

-----0_B

R/W

各端子がポートとして使用しているときは , 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。

149

8.2.3 ブルアップ制御レジスタ (PUCR0 ~ PUCR3) (ポート 0 ~ ポート 3 用)

ブルアップ制御レジスタ (PUCR0 ~ PUCR3) (ポート 0 ~ ポート 3 用) のビット構成を図 8.2-4 に、ブロックダイアグラムを図 8.2-5 に示します。

■ ブルアップ制御レジスタ (PUCR0 ~ PUCR3) (ポート 0 ~ ポート 3 用)

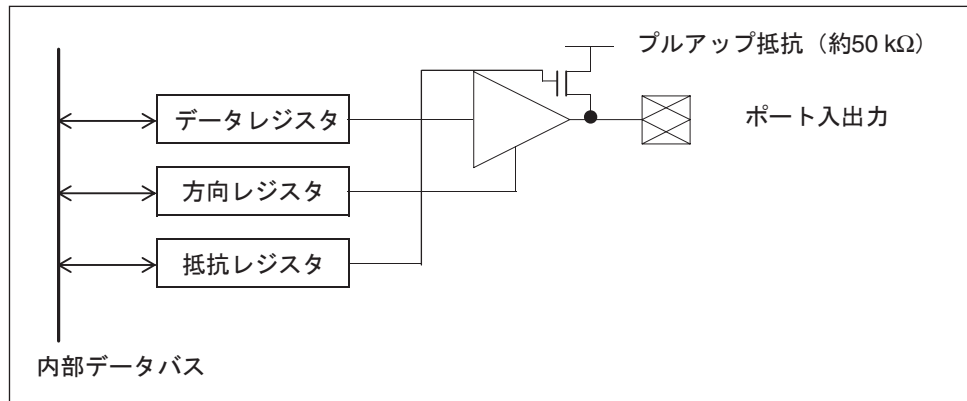
ブルアップ制御レジスタ (PUCR0 ~ PUCR3) (ポート 0 ~ ポート 3 用) のビット構成を図 8.2-4 に示します。

図 8.2-4 ブルアップ制御レジスタ (PUCR0 ~ PUCR3) (ポート 0 ~ ポート 3 用) のビット構成

アドレス : 00001C _H	7	6	5	4	3	2	1	0	ビット No.
	PU07	PU06	PU05	PU04	PU03	PU02	PU01	PU00	PUCR0
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(ポート 0 用)
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
アドレス : 00001D _H	7	6	5	4	3	2	1	0	ビット No.
	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10	PUCR1
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(ポート 1 用)
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
アドレス : 00001E _H	7	6	5	4	3	2	1	0	ビット No.
	PU27	PU26	PU25	PU24	PU23	PU22	PU21	PU20	PUCR2
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(ポート 2 用)
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
アドレス : 00001F _H	7	6	5	4	3	2	1	0	ビット No.
	PU37	PU36	PU35	PU34	PU33	PU32	PU31	PU30	PUCR3
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(ポート 3 用)
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

■ プルアップ制御レジスタ(PUCR0 ~ PUCR3) (ポート0 ~ ポート3用)のブロックダイヤグラム

図 8.2-5 プルアップ制御レジスタ (PUCR0 ~ PUCR3) (ポート 0 ~ ポート 3 用) のブロックダイヤグラム



< 注意事項 >

- 入力モード時のプルアップ抵抗制御を行います。
 - 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。
- 出力モード時は意味を持ちません (プルアップ抵抗なし)。
- 入出力モードは方向レジスタ (DDR) で決まります。
- ハードウェアスタンバイ時とストップモード時 (SPL=1) の場合は、プルアップ抵抗なしになります (ハイインピーダンス)。
- ポートが外部バスとして使用される場合、この機能は使用禁止になり、レジスタにデータは書き込まれません。

8.2.4 アナログ入力許可レジスタ (ADER)

アナログ入力許可レジスタ (ADER) のビット構成を , 図 8.2-6 に示します。

■ アナログ入力許可レジスタ (ADER)

アナログ入力許可レジスタ (ADER) のビット構成を図 8.2-6 に示します。

図 8.2-6 アナログ入力許可レジスタ (ADER) のビット構成

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00001B _H	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0	ADER
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	(1)	(1)	(1)	(1)	(1)	(1)	(1)	(1)	

アナログ入力許可レジスタ (ADER) は , ポート 6 の各端子を次のように制御します。

- 0: ポート入力モード
- 1: アナログ入力モード

外部端子が A/D 変換器のアナログ入力として使用される場合 , 対応するビットは "1" に設定してください。

第9章

タイムベースタイマ

タイムベースタイマの機能と動作について説明します。

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制御レジスタ (TBTC)

9.3 タイムベースタイマの動作

9.1 タイムベースタイマの概要

タイムベースタイマは、18ビットのタイマのほかに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。なお、タイムベースタイマは、クロック選択レジスタ (CKSCR) 中の MCS ビットおよび SCS ビットに関係なく、メインクロック信号を使用します。

■ タイムベースタイマのレジスタ一覧

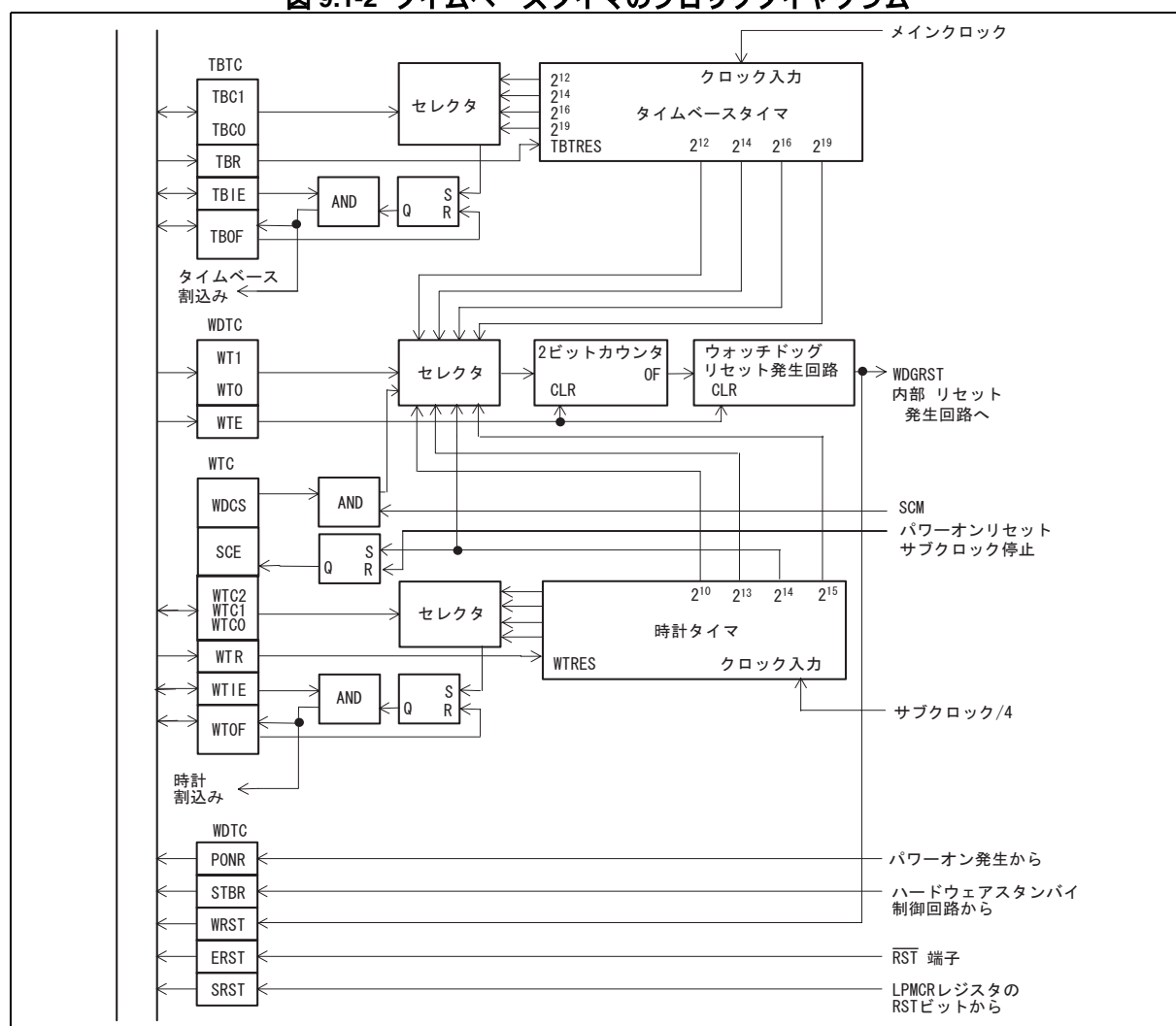
タイムベースタイマのレジスタのビット構成を図 9.1-1 に示します。

図 9.1-1 タイムベースタイマのレジスタ一覧

	15	14	13	12	11	10	9	8	ビット No.
アドレス: 0000A9 _H	予約			TBIE	TBOF	TBR	TBC1	TBC0	TBTC
リード/ライト	(R/W)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(1)	(-)	(-)	(0)	(0)	(1)	(0)	(0)	

■ タイムベースタイマのブロックダイアグラム

図 9.1-2 タイムベースタイマのブロックダイアグラム



9.2 タイムベースタイマ制御レジスタ (TBTC)

タイムベースタイマ制御レジスタ (TBTC) では、タイムベースタイマの動作を制御し、また、インターバル割込みの時間を制御します。

■ 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) のビット構成を図 9.2-1 に示します。

図 9.2-1 タイムベースタイマ制御レジスタ (TBTC)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 0000A9 _H	予約			TBIE	TBOF	TBR	TBC1	TBC0	TBTC
リード/ライト	(R/W)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(1)	(-)	(-)	(0)	(0)	(1)	(0)	(0)	

【bit 15】予約ビット

ライトの値は常に "1" にしてください。

【bit 14, bit 13】未使用ビット

bit 14, bit 13 は、空きビットです。

【bit 12】TBIE

タイムベース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" のとき割込み許可, "0" のとき割込みを禁止します。リセットにて "0" に初期化します。読み書き可能なビットです。

【bit 11】TBOF

タイムベースタイマの割込み要求フラグです。TBIE ビットが "1" のとき TBOF が "1" になると割込み要求を発生します。TBC1, TBC0 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、ハードウェアスタンバイモードへの遷移およびリセットによりクリアされます。"1" の書込みは意味がありません。リードモディファイライト系命令におけるリード時には "1" が読み出されます。

【bit 10】TBR

タイムベース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タイムベースタイマをクリアします。"1" の書込みは動作に影響はありません。リード時は "1" が読めます。

【bit 9, bit 8】TBC1, TBC0

タイムベースタイマのインターバル時間を設定するビットです。リセットにより "00_B" に初期化されます。読み書き可能なビットです。

表 9.2-1 に、タイムベースタイマ インターバル時間の設定を示します。

表 9.2-1 タイムベースタイマ インターバル時間の設定

TBC1	TBC0	原発振 4 MHz 時のインターバル時間
0	0	1.024 ms
0	1	4.096 ms
1	0	16.384 ms
1	1	131.072 ms

9.3 タイムベースタイマの動作

タイムベースタイマは、ウォッチドッグカウンタのクロックソース、メインクロックと PLL クロック発振安定待ち時間のタイマ、一定周期ごとに割込みを発生するインターバルタイマとしての機能があります。

■ タイムベースタイマ

タイムベースタイマは、マシニングクロックを作成する基になる発振クロックをカウントする 18 ビットのカウンタにより構成され、発振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。

タイムベースタイマのクリアは、パワーオンリセット、ストップモード、ハードウェアスタンバイモードへの遷移、クロック選択レジスタ (CKSCR) 内の MCS ビット書込みによるメインクロックから PLL クロックへの遷移、クロック選択レジスタ (CKSCR) 内の SCS ビット書込みによるメインクロックからサブクロックへの遷移およびタイムベースタイマ制御レジスタ (TBTC) 内の TBR ビットへの "0" 書込みにより行われます。

タイムベースタイマの出力を使用しているウォッチドッグカウンタとインターバル割込み機能は、タイムベースカウンタのクリアにより影響を受けます。

■ タイムベースタイマのインターバル割込み機能

タイムベースカウンタの桁上り信号により一定周期ごとに割込みを発生します。タイムベースタイマ制御レジスタ (TBTC) 内の TBC1, TBC0 ビットで設定されるインターバル時間ごとに TBOF フラグをセットします。このフラグのセットは、最後にタイムベースタイマがクリアされた時間を基準にして行われます。

メインクロックモードから PLL クロックモードに遷移すると、タイムベースタイマは PLL クロックの発振安定待ちのタイマとして使用するために、タイムベースタイマのクリアが行われます。

また、発振クロックモードからサブクロックモードに遷移すると、タイムベースタイマは発振クロックの発振安定待ちのタイマとして使用するために、タイムベースタイマがクリアされます。

ストップモードおよびハードウェアスタンバイモードに遷移すると、タイムベースタイマは復帰時の発振安定待ち時間のタイマとして使用されるので、TBOF フラグはモード遷移と同時にクリアされます。

第10章

ウォッチドッグタイマ

ウォッチドッグタイマの機能と動作について説明します。

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動作

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、18 ビットのタイムベースタイマまたは 15 ビットの時計タイマの桁上り信号をクロックソースとする 2 ビットのウォッチドッグカウンタ、制御レジスタ、ウォッチドッグリセット制御部により構成されています。

■ ウォッチドッグタイマのレジスタ一覧

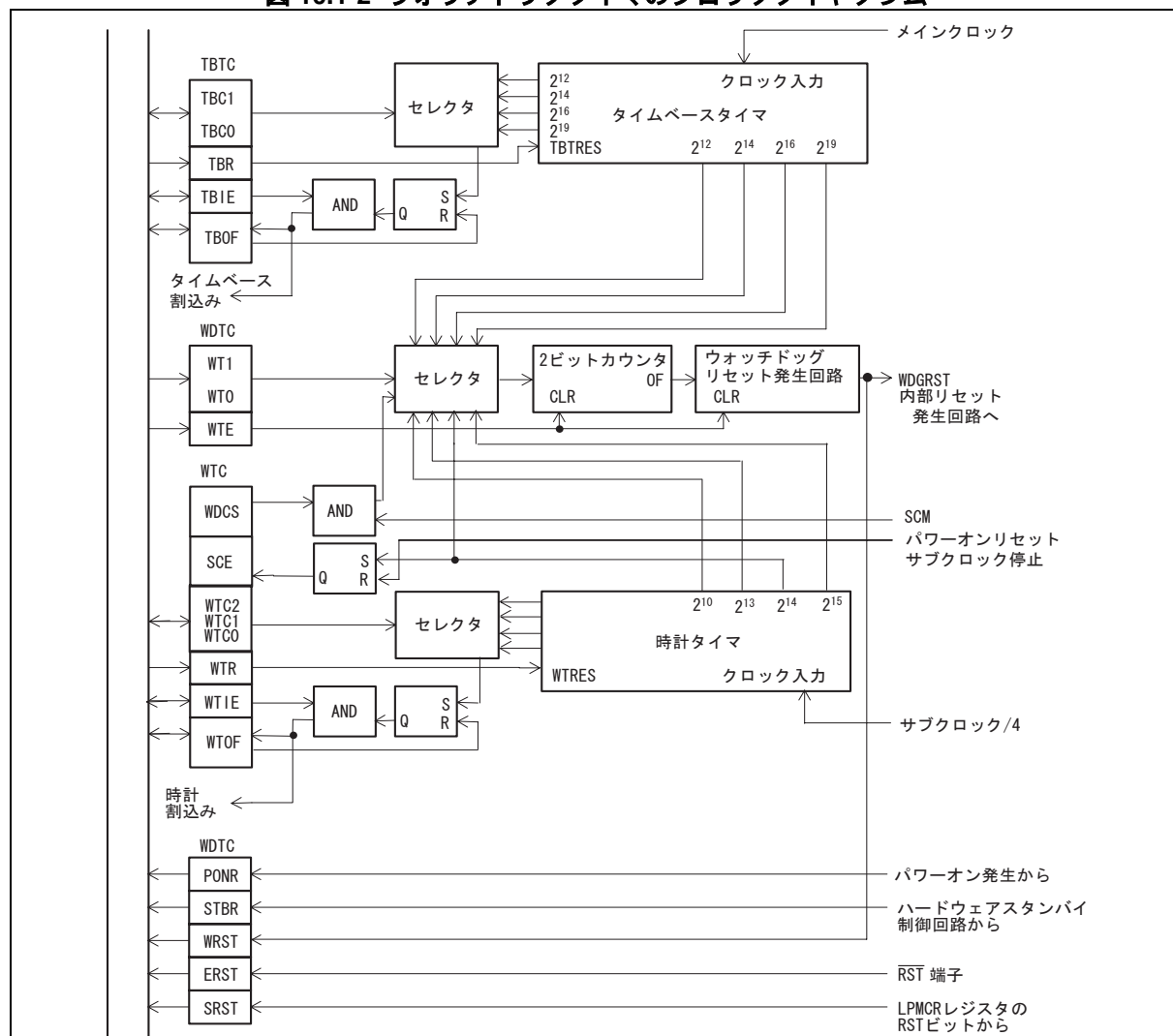
ウォッチドッグタイマのレジスタのビット構成を図 10.1-1 に示します。

図 10.1-1 ウォッチドッグタイマのレジスタ一覧

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000A8 _H	PONR	STBR	WRST	ERST	SRST	WTE	WT1	WT0	WDTC
リード/ライト	(R)	(R)	(R)	(R)	(R)	(W)	(W)	(W)	
初期値	(X)	(X)	(X)	(X)	(X)	(1)	(1)	(1)	

■ ウォッチドッグタイマのブロックダイアグラム

図 10.1-2 ウォッチドッグタイマのブロックダイアグラム



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関連の各種制御を行うビットと各種リセット要因を識別するビットで構成しています。

■ ウォッチドッグタイマ制御レジスタ (WDTC)

ウォッチドッグタイマ制御レジスタ (WDTC) のビット構成を図 10.2-1 に示します。

図 10.2-1 ウォッチドッグタイマ制御レジスタ (WDTC)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000A8 _H	PONR	STBR	WRST	ERST	SRST	WTE	WT1	WT0	WDTC
リード/ライト	(R)	(R)	(R)	(R)	(R)	(W)	(W)	(W)	
初期値	(X)	(X)	(X)	(X)	(X)	(1)	(1)	(1)	

【bit 7 ~ bit 3】PONR, STBR, WRST, ERST, SRST

リセットの要因を示すフラグです。各種リセット要因が発生すると、本ビットは表 10.2-1 に示すように設定されます。これらのビットはすべて WDTC レジスタのリード動作後に "0" にクリアされる読出し専用のレジスタです。なお、パワーオン時は、PONR ビット以外のリセット要因ビットの内容は定義されません。したがって、PONR ビットが "1" のとき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。

表 10.2-1 リセット要因ビットとリセット要因

リセット要因	PONR	STBR	WRST	ERST	SRST
パワーオン	1	-	-	-	-
ハードウェアスタンバイ	*	1	*	*	*
ウォッチドッグタイマ	*	*	1	*	*
外部端子	*	*	*	1	*
RST ビット	*	*	*	*	1

※: 前の値を保持します。

【bit 2】WTE

ウォッチドッグタイマが停止状態のときに本ビットに "0" を書き込むことによりウォッチドッグタイマを動作状態にします。2 回目以降の "0" の書込みではウォッチドッグタイマのカウンタをクリアします。"1" の書込みは動作に影響はありません。

ウォッチドッグタイマは、パワーオン、ハードウェアスタンバイ、ウォッチドッグタイマによるリセットにより停止状態となります。読出し時は "1" が読み出されます。

【bit 1, bit 0】WT1, WT0

ウォッチドッグタイマのインターバル時間を選択します。

ウォッチドッグタイマ起動時に書き込まれたデータのみが有効です。ウォッチドッグ起動時以外の書き込みデータは無視されます。なお、2 系統品では、時計タイマ制御レジスタ (WTC) 内の WDCS ビットによりウォッチドッグタイマに入力するクロック信号の選択が行われます (タイムベースタイマ, 時計タイマ)。

WT1, WT0 ビットによるインターバル時間の設定を、表 10.2-2 に示します。

表 10.2-2 WT1, WT0 へのアクセス (読出し専用)

WDCS	WT1	WT0	インターバル時間 *	
			最小	最大
1	0	0	約 3.58 ms	約 4.61 ms
1	0	1	約 14.33 ms	約 18.43 ms
1	1	0	約 57.23 ms	約 73.73 ms
1	1	1	約 458.75 ms	約 589.82 ms
0	0	0	約 0.457 s	約 0.576 s
0	0	1	約 3.584 s	約 4.608 s
0	1	0	約 7.168 s	約 9.216 s
0	1	1	約 14.336 s	約 18.432 s

* : 原発振 4 MHz, サブクロック 32 kHz の場合

< 注意事項 >

インターバル時間は、タイムベースタイマまたは時計タイマの桁上り信号をカウントクロックにしているため、タイムベースタイマまたは時計タイマがクリアされると、ウォッチドッグタイマのインターバル時間が長くなる場合があります。
タイムベースタイマはタイムベースタイマ制御レジスタ (TBTC) の TBR ビットへの "0" 書き込みのほか、メインクロックモードから PLL クロックモードへの遷移、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、サブクロックモードから PLL クロックモードへの遷移時にもクリアされますので、ご注意ください。

10.3 ウォッチドッグタイマの動作

ウォッチドッグタイマの機能は、プログラムの暴走を検出することができます。プログラムの暴走などにより、ウォッチドッグタイマの WTE ビットに "0" 書き込みが定められた時間内に行われなかった場合、ウォッチドッグタイマよりウォッチドッグリセット要求が発生します。

■ ウォッチドッグタイマの起動方法

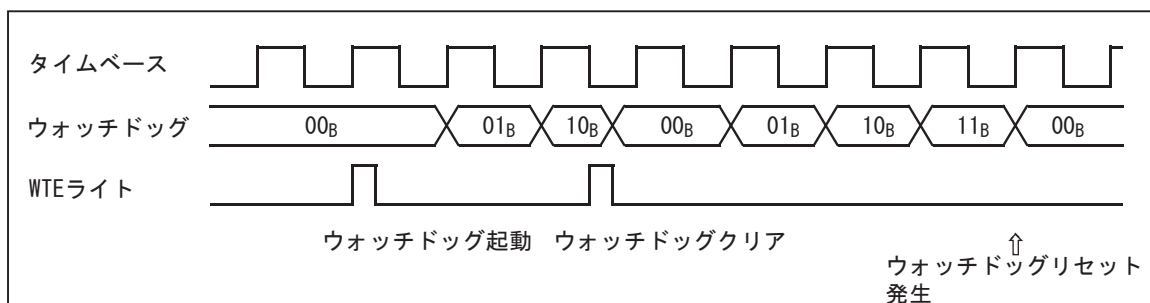
ウォッチドッグタイマは、ウォッチドッグタイマの停止中にウォッチドッグタイマ制御レジスタ (WDTC) 内の WTE ビットに "0" を書き込むことによって起動します。このとき、同時にウォッチドッグタイマのインターバル時間を WT1, WT0 ビットにより設定します。インターバルの設定は、この起動のときのデータのみが有効となります。

■ ウォッチドッグタイマのリセット

ウォッチドッグタイマが起動されると、プログラムで定期的に2ビットのウォッチドッグカウンタをクリアする必要があります。具体的には、ウォッチドッグタイマ制御レジスタ (WDTC) 内の WTE ビットに定期的に "0" を書き込む必要があります。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クロックソースとする2ビットのカウンタにより構成されています。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インターバルが設定より長くなることがあります。

ウォッチドッグタイマの動作を、図 10.3-1 に示します。

図 10.3-1 ウォッチドッグタイマ動作



■ ウォッチドッグカウンタの停止

ウォッチドッグタイマは一度起動すると、パワーオン、ハードウェアスタンバイ、ウォッチドッグによるリセットでのみ初期化され停止状態となります。

外部端子、ソフトウェアによるリセットでは、ウォッチドッグカウンタをクリアしますが、ウォッチドッグ機能は停止されません。

■ ウォッチドッグカウンタのクリア

ウォッチドッグカウンタは、ウォッチドッグタイマ制御レジスタ (WDTC) 内の WTE ビットに "0" を書き込むほかに、リセットの発生、スリープモード、ストップモードまたはホールドアクノリッジ信号への遷移によってクリアされます。

第11章

時計タイマ

時計タイマの機能と動作について説明します。
時計タイマは1系統品では使用できません。

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

11.1 時計タイマの概要

時計タイマは、15 ビットのタイマのほかに、インターバル割込みを制御する回路から構成されています。なお、時計タイマは、クロック選択レジスタ (CKSCR) 内の MCS ビットおよび SCS ビットに関係なく、サブクロック信号を使用します。

■ 時計タイマのレジスタ

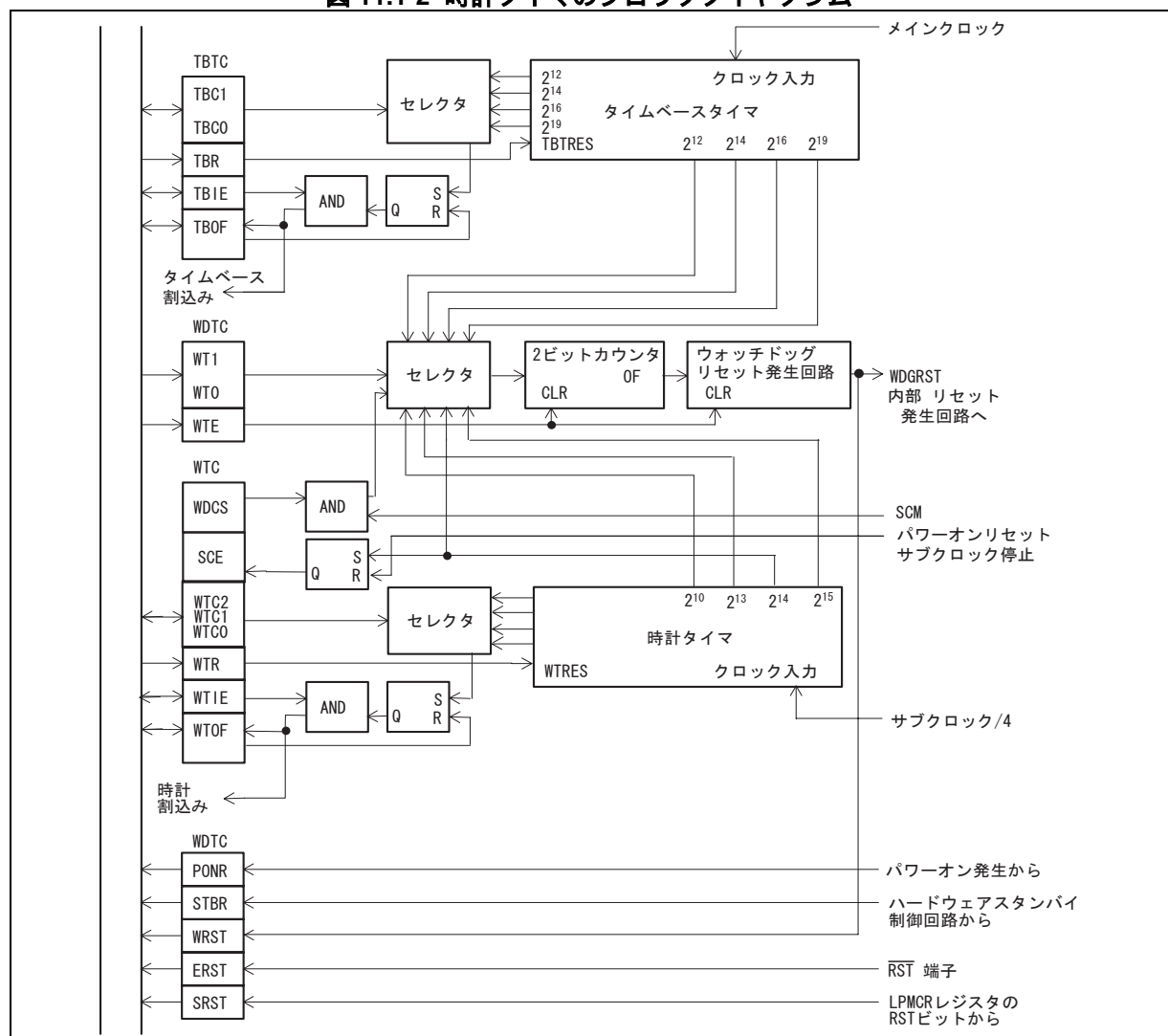
時計タイマのレジスタのビット構成を図 11.1-1 に示します。

図 11.1-1 時計タイマのレジスタ一覧

	7	6	5	4	3	2	1	0	ビット No.
アドレス: 0000AA _H	WDCS	SCE	WTIE	WTOF	WTR	WTC2	WTC1	WTC0	WTC
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	(1)	(X)	(0)	(0)	(0)	(0)	(0)	(0)	

■ 時計タイマのブロックダイアグラム

図 11.1-2 時計タイマのブロックダイアグラム



11.2 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)

時計タイマ制御レジスタ (WTC) のビット構成を図 11.2-1 に示します。

図 11.2-1 時計タイマ制御レジスタ (WTC)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000AA _H	WDCS	SCE	WTIE	WTOF	WTR	WTC2	WTC1	WTC0	WTC
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	(1)	(X)	(0)	(0)	(0)	(0)	(0)	(0)	

【bit 7】WDCS

ウォッチドッグタイマの入力クロックとして、時計タイマのクロック信号を使用するかまたはタイムベースタイマのクロック信号を使用するかを選択するビットです。

<メインクロックモードまたは PLL クロックモード時>

WDCS 1 : タイムベースタイマ

0 : 時計タイマ

<サブクロックモード時>

WDCS 必ず "0" を書き込んでください。

パワーオンリセットにより、本ビットは "1" に初期化されます。

< 注意事項 >

WDCS を "1" に設定した場合には、タイムベースタイマの出力と時計タイマの出力は非同期であるため、ウォッチドッグタイマのカウンタが進む可能性があります。このため、WDCS を "1" に設定した場合には、クロックモードを変更する前後で、ウォッチドッグタイマをクリアする必要があります。

サブクロックモード時にウォッチドッグタイマの入力としてタイムベースタイマを選択 (WDCS=1) した場合にはウォッチドッグタイマは停止します。

サブクロックモード時にウォッチドッグタイマを使用する場合は必ず時計タイマを選択 (WDCS=0) してください。

【bit 6】SCE

サブクロックの発振安定待ち時間が経過したことを示すビットです。本ビットが "0" のとき、発振安定待ち時間が現在進行中であることを示しています。発振安定待ち時間は、2¹⁴ サイクル (サブクロック) 固定です。パワーオンリセット時およびストップ時に、本ビットは "0" に初期化されます。

【bit 5】 WTIE

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。本ビットが "1" のときに割込みを許可し, "0" のときに割込みを禁止します。リセットにより本ビットは "0" に初期化されます。読み書き可能なビットです。

【bit 4】 WTOF

時計タイマの割込みフラグです。WTIE ビットが "1" のときに, WTOF を "1" に設定すると割込み要求が発生します。本ビットは, WTC1, WTC0 ビットで設定したインターバル時間ごとに "1" に設定されます。本ビットは, "0" の書込み, ストップモード, ハードウェアスタンバイモードへの遷移およびリセットによりクリアされます。"1" の書込みは動作に影響はありません。

リードモディファイライト系命令によるリード時には, 常に "1" が読み出されます。

【bit 3】 WTR

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。本ビットに "0" を書き込むことにより, 時計カウンタをクリアします。"1" の書込みは動作に影響はありません。常に "1" が読み出されます。

【bit 2 ~ bit 0】 WTC2 ~ WTC0

時計タイマのインターバル時間を設定するビットです。表 11.2-1 にインターバル時間の設定を示します。リセットにより本ビットは "000_B" に初期化されます。読み書き可能なビットです。

本ビットに書き込むときには, ビット 4(WTOF) もクリアしてください。

表 11.2-1 時計タイマ インターバル時間の設定

WTC2	WTC1	WTC0	インターバル時間 (サブクロック 32 kHz)
0	0	0	62.5 ms
0	0	1	125 ms
0	1	0	250 ms
0	1	1	500 ms
1	0	0	1.000 s
1	0	1	2.000 s
1	1	0	4.000 s
1	1	1	-

11.3 時計タイマの動作

時計タイマは、ウォッチドッグカウンタのクロックソース、サブクロックの発振安定待ち時間のためのタイマ、一定周期ごとに割込みを発生するインターバルタイマとしての機能があります。

■ 時計タイマ

時計タイマは、サブクロックにより発生する発振入力のカウントする 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、時計タイマはカウント動作を続けます。時計タイマのクリアは、パワーオンリセットおよび時計タイマ制御レジスタ (WTC) 内の WTR ビットへの "0" の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カウンタとインターバル割込みは、時計カウンタのクリアにより影響を受けます。
 - 時計タイマ制御レジスタ (WTC) の WTR ビットに "0" を設定して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 ビットへの "0" 設定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

インターバル割込み機能は、時計カウンタの桁上り信号により一定周期ごとに割込みを発生します。時計タイマ制御レジスタ (WTC) 内の WTC1, WTC0 ビットで設定されるインターバル時間ごとに WTOF フラグをセットします。このフラグ設定のタイミングは、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。

ストップモードおよびハードウェアスタンバイモードに遷移すると、時計タイマは復帰時のサブクロックの発振安定待ち時間のタイマとして使用されるので、WTOF フラグ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動作クロックの設定

時計タイマ制御レジスタ (WTC) のウォッチドッグクロック選択ビット (WDCS) によって、ウォッチドッグタイマのクロックソースを設定できます。マシクロックとしてサブクロックを使用する場合、必ず WDCS ビットを "0" に設定して、時計タイマの出力を選択してください。WDCS ビットを "1" に設定した状態で、サブクロックモードへ移行した場合、ウォッチドッグタイマは停止します。

第12章

16 ビット入出力タイマ

16 ビット入出力タイマの機能と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 ビットフリーランタイマ

12.4 アウトプットコンペア

12.5 インプットキャプチャ

12.1 16 ビット入出力タイマの概要

MB90540/545 シリーズは、16 ビットフリーランタイムモジュール 1 本、アウトプットコンペアモジュール 2 本、インプットキャプチャモジュール 4 本のモジュールから構成され、8 個の入力チャンネルおよび 4 個の出力チャンネルをサポートしています。ここでは、16 ビットフリーランタイム、アウトプットコンペア 0/1 およびインプットキャプチャ 0/1 について説明します。残りのモジュールの機能も同様であり、レジスタアドレスに関しては、「付録 A I/O マップ」を参照してください。

■ 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 ビットのアップカウンタ、制御レジスタ、プリスケアラより構成されています。本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。

- カウンタクロックは 4 種類から選択可能
内部クロック 4 種類 ($\phi/4$, $\phi/16$, $\phi/64$, $\phi/256$)
- 割込みはカウンタ値のオーバフロー、コンペアレジスタ 0 とのコンペアマッチにより発生可能
- カウンタ値はリセット、ソフトクリア、コンペアレジスタ 0 とのコンペアマッチにより "0000_H" に初期化可能

■ アウトプットコンペア (モジュールごとに 2 チャンネル)

アウトプットコンペアモジュールは 2 本の 16 ビットコンペアレジスタ、コンペア出力用ラッチ、制御レジスタより構成されています。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共に割込みを発生することができます。

- 2 本のコンペアレジスタを独立して動作
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
- 2 本のコンペアレジスタをペアにして出力端子を制御可能
コンペアレジスタ 2 本を使用して出力端子を反転
- 出力端子の初期値を設定することが可能
- 割込みはコンペア一致により発生可能

■ インプットキャプチャ (モジュールごとに 2 チャンネル)

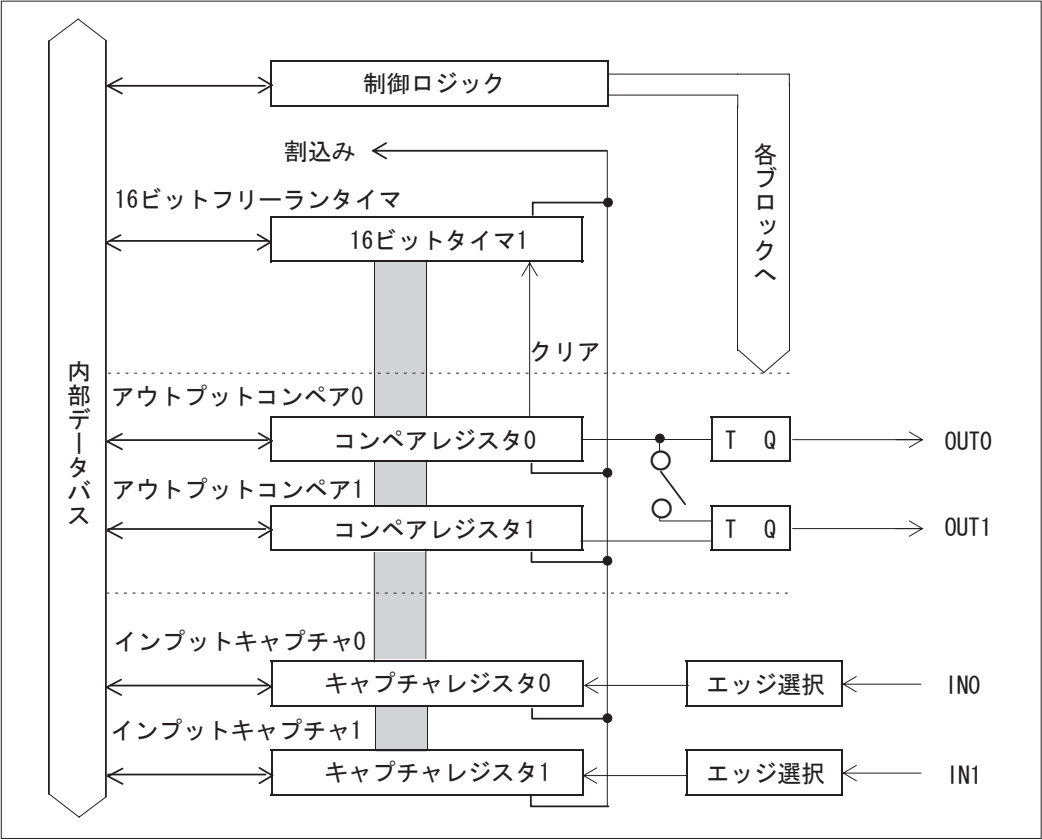
インプットキャプチャモジュールは独立した 2 本の外部入力端子と対応した 16 ビットキャプチャレジスタ、制御レジスタにより構成されています。外部入力端子より入力された信号の任意エッジを検出することにより 16 ビットフリーランタイム値をキャプチャレジスタに保持し、同時に割込みを発生することができます。

- 外部入力信号の検出エッジを選択可能
立上りエッジ、立下りエッジ、両エッジから選択可能
- 2 本のインプットチャンネルは独立して動作可能
- 割込みは外部入力信号の有効エッジにより発生可能
インプットキャプチャの割込みによりインテリジェント I/O サービスを起動可能

■ 16 ビット入出力タイマの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

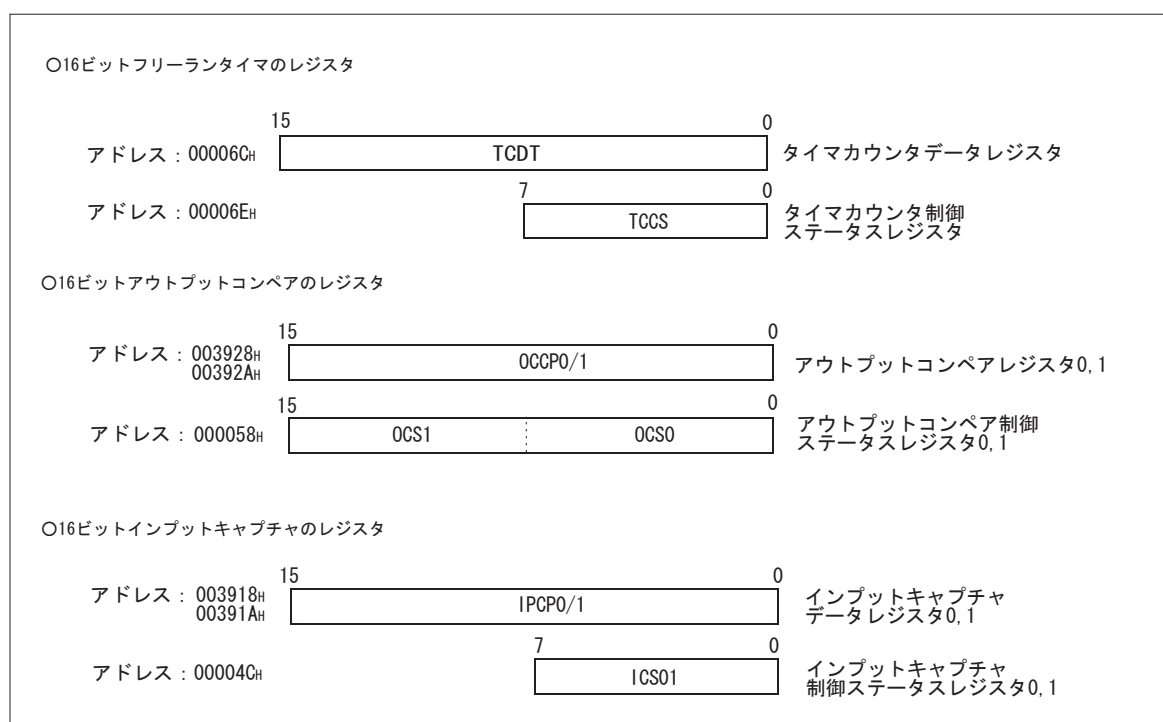
16 ビット入出力タイマのレジスタには、次の 3 つがあります。

- 16 ビットフリーランタイマのレジスタ
- 16 ビットアウトプットコンペアのレジスタ
- 16 ビットインプットキャプチャのレジスタ

■ 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のビット構成を図 12.2-1 に示します。

図 12.2-1 16 ビット入出力タイマのレジスタ



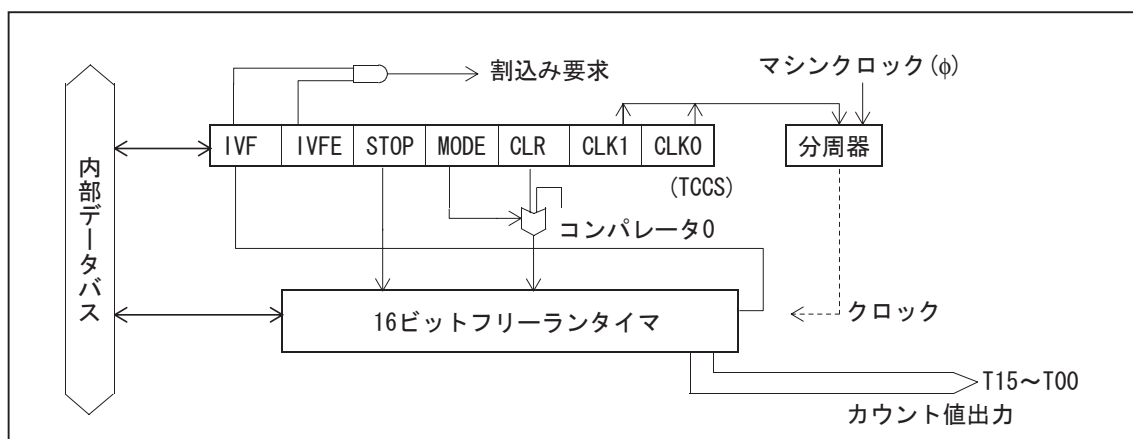
12.3 16 ビットフリーランタイム

16 ビットフリーランタイムは、16 ビットのアップカウンタと制御ステータスレジスタから構成されています。カウント値はアウトプットコンペア、インプットキャプチャの基本時間（ベースタイム）として使用されます。

- カウントクロックは 4 種類から選択可能です。
- カウンタオーバーフロー割込みを発生することができます。
- モード設定によりアウトプットコンペアのコンペアレジスタ 0 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ブロックダイアグラム

図 12.3-1 16 ビットフリーランタイムのブロックダイアグラム



12.3.1 16 ビットフリーランタイマのレジスタ

データレジスタは、16 ビットフリーランタイマのカウント値を読み出す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_B" にクリアされます。このレジスタに書き込むことで、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。このレジスタはワードアクセスしてください。

■ 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 12.3-2 に示します。

図 12.3-2 タイマカウンタデータレジスタ (TCDT)

アドレス : 00006D _H	15	14	13	12	11	10	9	8	ビット No.
	T15	T14	T13	T12	T11	T10	T09	T08	
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
リード / ライト 初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
アドレス : 00006C _H	7	6	5	4	3	2	1	0	ビット No. TCDT
	T07	T06	T05	T04	T03	T02	T01	T00	
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
リード / ライト 初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

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
- タイマ制御ステータスレジスタのクリアビット (CLR) による初期化
- コンペアのコンペアレジスタ 0 とタイマカウンタ値の一致による初期化
(モードの設定が必要です。)

12.3.2 タイマカウンタ制御ステータスレジスタ (TCCS)

タイマカウンタ制御ステータスレジスタ (TCCS) は、16 ビットフリーランタイムの動作モードの設定、起動・停止や割込みの制御を行います。

■ タイマカウンタ制御ステータスレジスタ (TCCS)

タイマカウンタ制御ステータスレジスタ (TCCS) のビット構成を図 12.3-3 に示します。

図 12.3-3 タイマカウンタ制御ステータスレジスタ (TCCS)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00006E _H	予約	IVF	IVFE	STOP	MODE	CLR	CLK1	CLK0	TCCS
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)	

【bit 7】予約ビット

本ビットには必ず "0" を書き込んでください。

【bit 6】IVF

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レジスタ0とコンペアマッチしてカウンタクリアされたときに本ビットは"1"にセットされます。

割込み要求許可ビット (bit5 : IVFE) がセットされていると割込みが発生します。

本ビットは "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系命令では "1" が読めます。

0	割込み要求なし (初期値)
1	割込み要求あり

【bit 5】IVFE

16 ビットフリーランタイムの割込み許可ビットです。本ビットが "1" のとき、割込みフラグ (bit5: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

【bit 4】STOP

16 ビットフリーランタイム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停止。"0" 書込み時にタイマのカウントを開始。

0	カウント許可 (動作) (初期値)
1	カウント禁止 (停止)

< 注意事項 >

16 ビットフリーランタイマのカウンタが停止すると、アウトプットコンペア動作も停止します。

【bit 3】MODE

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 : CLR) でカウンタ値を初期化可能です。

"1" のときはリセットとクリアビット (bit2 : CLR) のほかにアウトプットコンペアのコンペアレジスタ 0 の値との一致によりカウンタ値を初期化することができます。

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、コンペアレジスタ 0 による初期化

< 注意事項 >

クリアビットとコンペアレジスタとの一致により、タイマ値の変化点でタイマの初期値が行われます。

【bit 2】CLR

動作中の 16 ビットフリーランタイマ値を "0000_B" に初期化するビットです。

"1" 書込み時にカウンタ値を "0000_B" に初期化します。"0" を書き込んでも意味を持ちません。リード値は常に "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 _B " に初期化します

< 注意事項 >

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。

【bit 1, bit 0】CLK1, CLK0

16 ビットフリーランタイマのカウントクロックを選択します。本ビットに書き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。

CLK1	CLK0	カウントクロック 分周比	φ=16 MHz	φ= 8 MHz	φ= 4 MHz	φ= 2 MHz
0	0	φ/4	0.25 μs	0.5 μs	1 μs	2 μs
0	1	φ/16	1 μs	2 μs	4 μs	8 μs
1	0	φ/64	4 μs	8 μs	16 μs	32 μs
1	1	φ/256	16 μs	32 μs	64 μs	128 μs

(注意事項) φ マシンクロック

12.3.3 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。

■ 16 ビットフリーランタイマの動作

カウンタ値は、次の 5 つの条件でクリアされます。

- ・ オーバフローが発生したとき
- ・ アウトプットコンペアレジスタ 0 値とコンペアマッチしたとき (モード設定が必要)
- ・ 動作中に TCCS レジスタの CLR ビットに "1" を書き込んだとき
- ・ 停止中に TCDT 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と、コンペアレジスタ 0 値とコンペアマッチしてカウンタクリアされたときに発生することができます (コンペアマッチ割込みはモード設定が必要です)。

図 12.3-4 オーバフローによるカウンタクリア

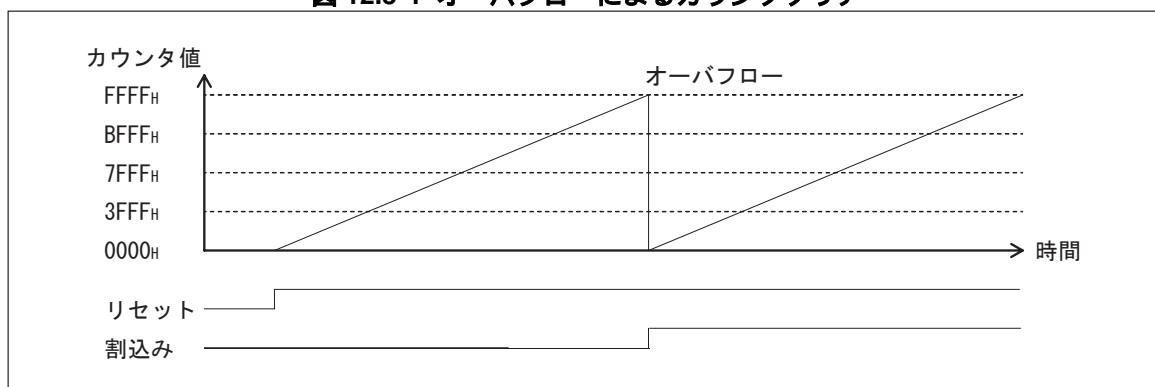
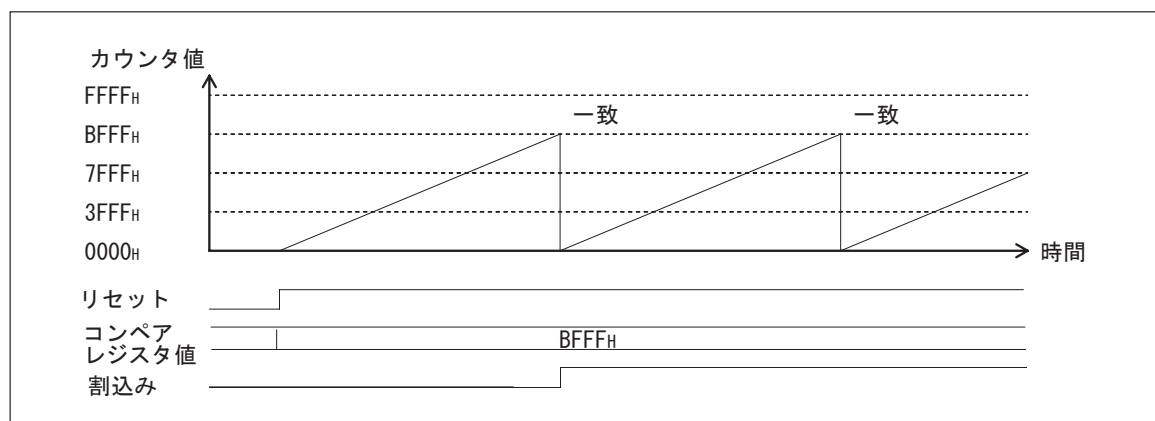


図 12.3-5 アウトプットコンペアレジスタ 0 値とコンペアマッチしたときのカウンタクリア



■ 16 ビットフリーランタイマのタイミング

図 12.3-6 に示すように、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。

図 12.3-6 16 ビットフリーランタイマのカウントタイミング

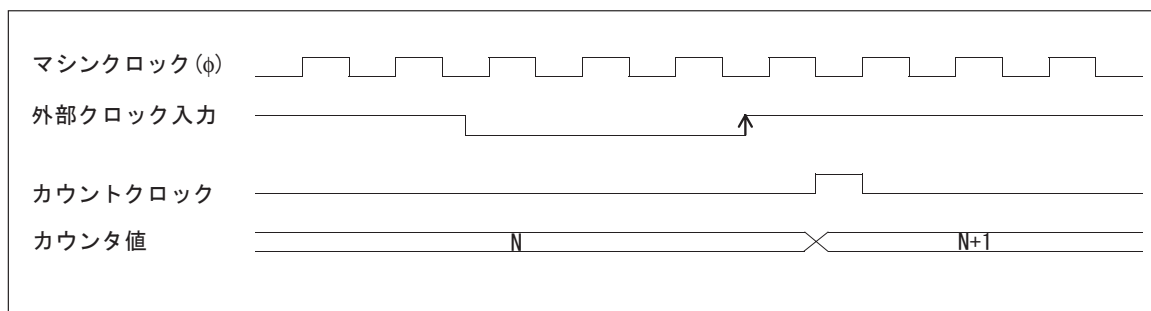
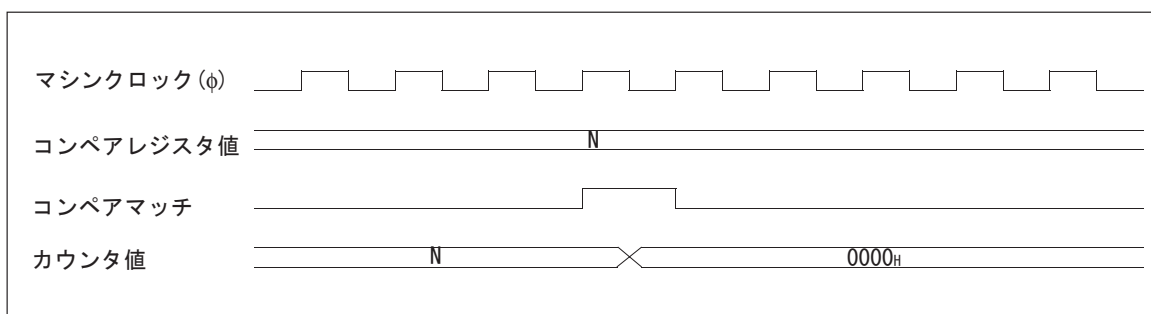


図 12.3-7 に示すように、カウンタのクリアはリセット、ソフトウェアのクリア、コンペアレジスタ 0 との一致で行うことができます。リセットとソフトウェアのクリアでのカウンタクリアはクリア発生とともに行われますが、コンペアレジスタ 0 との一致によるカウンタクリアはカウントタイミングに同期して行われます。

図 12.3-7 16 ビットフリーランタイマのクリアタイミング (コンペアレジスタ 0 との一致)



12.4 アウトプットコンペア

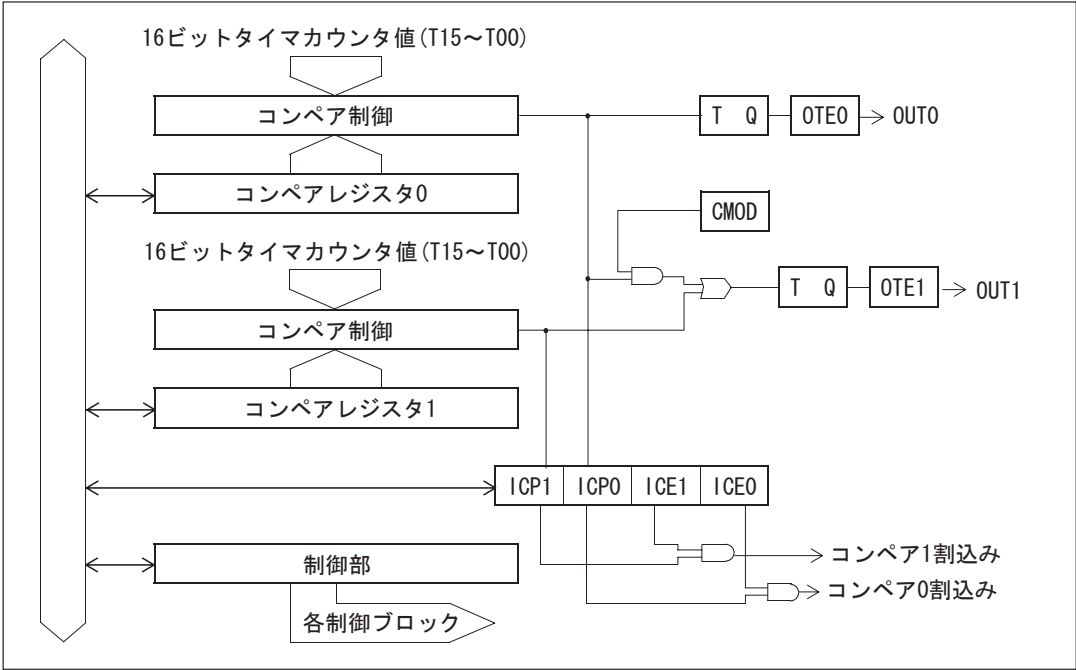
アウトプットコンペアモジュールは、2 つの 16 ビットコンペアレジスタ、2 本のコンペア出力端子および制御レジスタから構成されています。コンペアレジスタに設定された値と 16 ビットフリーランタイム値が一致したら、端子の出力レベルを反転すると共に割込みを発生することができます。

■ アウトプットコンペア

- コンペアレジスタは全部で 2 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端子の出力を制御可能です。
- 端子の出力初期値を設定することができます。
- 割込みはコンペア一致により発生可能です。

■ アウトプットコンペアのブロックダイアグラム

図 12.4-1 アウトプットコンペアのブロックダイアグラム



12.4.1 アウトプットコンペアのレジスタ

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。レジスタ値は初期値不定ですので設定した後，起動許可してください。本レジスタはワードアクセスしてください。本レジスタ値と 16 ビットフリーランタイム値が一致したときコンペア信号が発生してアウトプットコンペア割込みフラグをセットします。また，出力許可をしている場合はコンペアレジスタに対応した出力レベルを反転します。

■ アウトプットコンペアのコンペアレジスタ (OCCP)

コンペアレジスタ (OCCP) のビット構成を図 12.4-2 に示します。

図 12.4-2 コンペアレジスタ (OCCP)

アドレス : 003929 _H 00392B _H	15	14	13	12	11	10	9	8	ビット No.
	C15	C14	C13	C12	C11	C10	C09	C08	
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003928 _H 00392A _H	7	6	5	4	3	2	1	0	ビット No. OCCP0, OCCP1
	C07	C06	C05	C04	C03	C02	C01	C00	
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

12.4.2 アウトプットコンペアの制御ステータスレジスタ (OCS0/1)

制御ステータスレジスタ (OCS0/1) は, アウトプットコンペアの動作モードの設定, 起動・停止や割込みの制御, 外部出力端子の設定を行います。

■ アウトプットコンペアの制御ステータスレジスタ (OCS0/1)

制御ステータスレジスタ (OCS0/1) のビット構成を図 12.4-3 に示します。

図 12.4-3 制御ステータスレジスタ (OCS0/1)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000059 _H				CMOD	OTE1	OTE0	OTD1	OTD0	
リード/ライト	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(-)	(-)	(-)	(0)	(0)	(0)	(0)	(0)	
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000058 _H	ICP1	ICP0	ICE1	ICE0			CST1	CST0	OCS0/1
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(-)	(-)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(-)	(-)	(0)	(0)	

【bit 15 ~ bit 13】未使用ビット

【bit 12】CMOD

端子出力を許可した場合 (OTE1=1 または 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端子の出力レベルを反転します。
 - OUT0: コンペアレジスタ 0 の一致によりレベルを反転します。
 - OUT1: コンペアレジスタ 1 の一致によりレベルを反転します。
- CMOD=1 のときコンペアレジスタ 0 は, CMOD=0 時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 に対応した端子 (OUT1) の出力レベルは, コンペアレジスタ 0 の一致とコンペアレジスタ 1 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 と 1 が同値のときはコンペアレジスタ 1 本のとおり動作をします。
 - OUT0: コンペアレジスタ 0 の一致によりレベルを反転します。
 - OUT1: コンペアレジスタ 0 または 1 の一致によりレベルを反転します。

【bit 11, bit 10】OTE1, OTE0

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。本ビットは初期値 "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。

< 注意事項 >

OTE1: アウトプットコンペア 1(OUT1) に対応

OTE0: アウトプットコンペア 0(OUT0) に対応

【bit 9, bit 8】OTD1, OTD0

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出力値が読めます。

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。

< 注意事項 >

OTD1: アウトプットコンペア 1 に対応

OTD0: アウトプットコンペア 0 に対応

【bit 7, bit 6】ICP1, ICP0

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に本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。本ビットは "0" 書込みによりクリアされ, "1" 書込みでは意味をもちません。リードモディファイライト系命令では "1" が読めます。

0	コンペア一致なし (初期値)
1	コンペア一致あり

< 注意事項 >

ICP1: アウトプットコンペア 1 に対応

ICP0: アウトプットコンペア 0 に対応

【bit5, bit4】ICE1, ICE0

アウトプットコンペアの割込み許可ビットです。本ビットが "1" のとき割込みフラグ (ICP0, ICP1) 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

< 注意事項 >

ICE1: アウトプットコンペア 1 に対応

ICE0: アウトプットコンペア 0 に対応

【bit 3, bit 2】未使用ビット

【bit 1, bit 0】CST1, CST0

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

コンペア動作許可を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 >

CST1: アウトプットコンペア 1 に対応

CST0: アウトプットコンペア 0 に対応

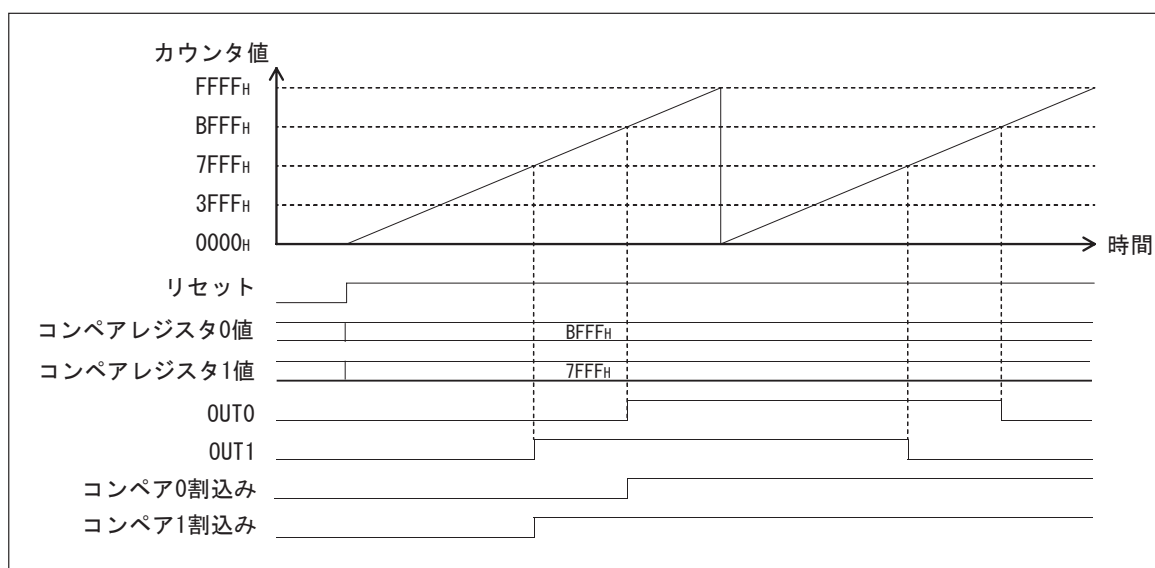
アウトプットコンペアは16ビットフリーランタイムのクロックと同期させているため16ビットフリーランタイムを停止させるとコンペア動作も停止します。

12.4.3 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、出力レベルを反転することができます。

■ コンペアレジスタ 0, 1 を使用したときの出力波形例 (出力の初期値は "0" とする)

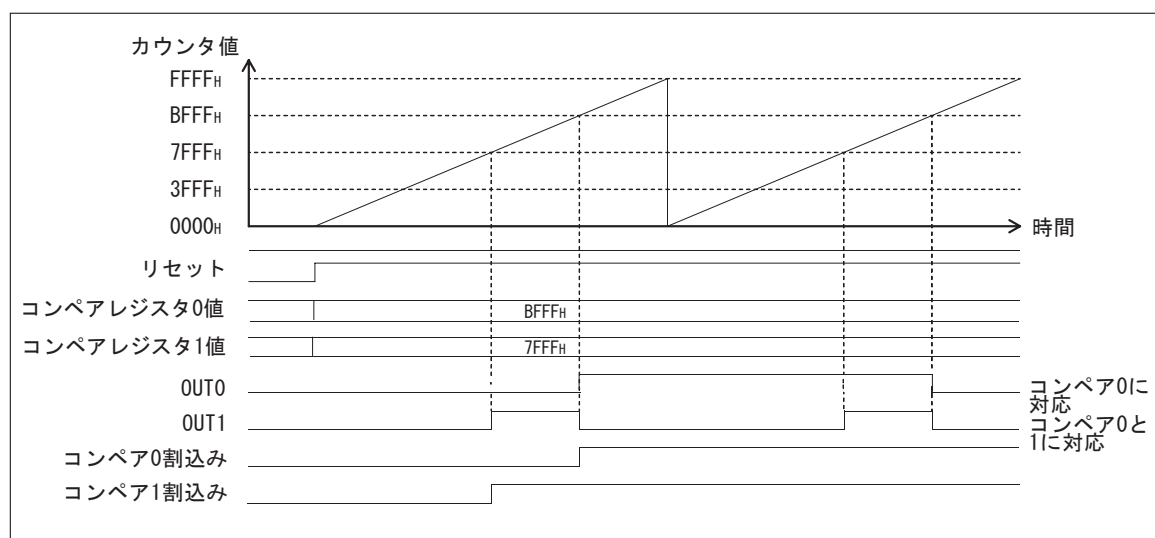
図 12.4-4 コンペアレジスタ 0, 1 を使用したときの出力波形例 (出力の初期値は "0" とする)



2組のコンペアレジスタを使い、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 (出力の初期値は "0" とする)

図 12.4-5 2 組のコンペアレジスタによる出力波形例 (出力の初期値は "0" とする)



■ 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、フリーランタイム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マッチ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することができます。コンペアマッチ時の出力反転タイミングはカウンタのカウントタイミングに同期して行われます。

図 12.4-6 に示すように、コンペアレジスタ書換え時のカウンタ値とはコンペアしません。

図 12.4-6 コンペアレジスタ書換え時のコンペア動作

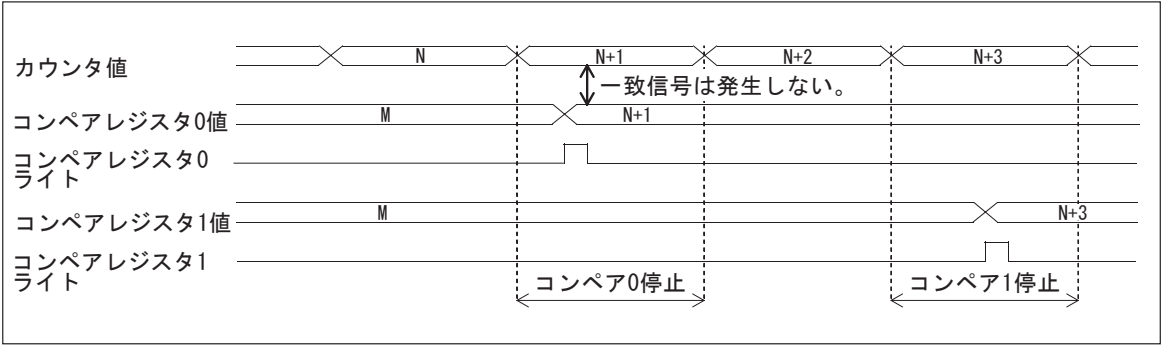


図 12.4-7 にアウトプットコンペアの割込みタイミングを、図 12.4-8 にアウトプットコンペアの出力端子変化タイミングを示します。

図 12.4-7 アウトプットコンペアの割込みタイミング

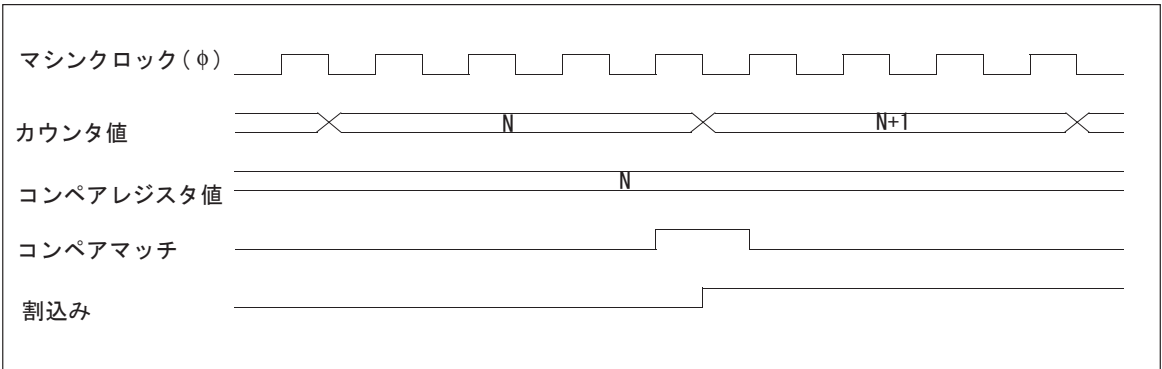
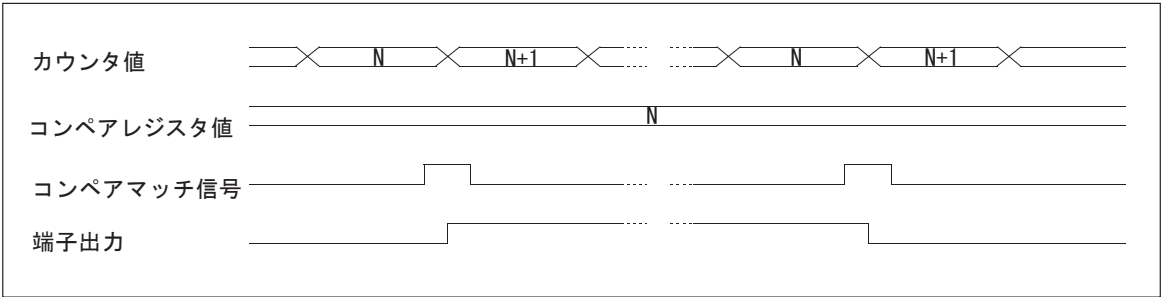


図 12.4-8 アウトプットコンペアの出力端子変化タイミング



12.5 インプットキャプチャ

インプットキャプチャ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することができます。インプットキャプチャは，インプットキャプチャデータレジスタ，制御ステータスレジスタで構成されています。

■ インプットキャプチャ

各インプットキャプチャは，それぞれに対応した外部入力端子を持っています。

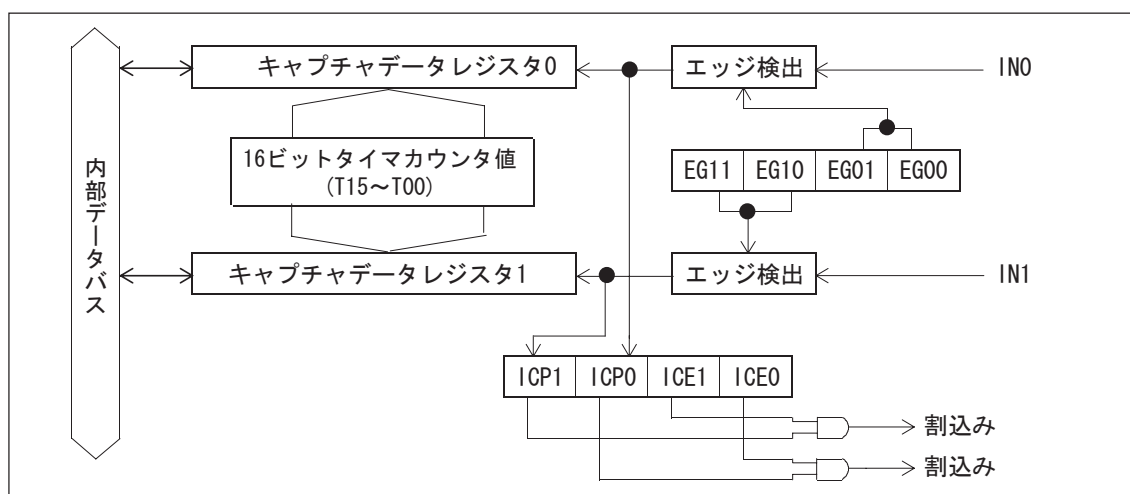
- 外部入力の検出エッジを 3 種類から選択可能です。

- 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することができます。

■ インプットキャプチャのブロックダイアグラム

図 12.5-1 インプットキャプチャの全体ブロックダイアグラム



12.5.1 インプットキャプチャのレジスタ

インプットキャプチャのレジスタは、次の 2 つがあり、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 ビットタイマ値を保持するレジスタです (ワードアクセスしてください。書込みはできません)。

■ インプットキャプチャデータレジスタ (IPCP0/1)

インプットキャプチャデータレジスタ (IPCP0/1) のビット構成を図 12.5-2 に示します。

図 12.5-2 インプットキャプチャデータレジスタ (IPCP0/1)

アドレス : 003919 _H	15	14	13	12	11	10	9	8	ビット No.
00391B _H	CP15	CP14	CP13	CP12	CP11	CP10	CP09	CP08	
リード / ライト	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

アドレス : 003918 _H	7	6	5	4	3	2	1	0	ビット No.
00391A _H	CP07	CP06	CP05	CP04	CP03	CP02	CP01	CP00	IPCP0/1
リード / ライト	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

■ インプットキャプチャ制御ステータスレジスタ (ICS01)

インプットキャプチャ制御ステータスレジスタ (ICS01) のビット構成を図 12.5-3 に示します。

図 12.5-3 インプットキャプチャ制御ステータスレジスタ (ICS01)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00004C _H	ICP1	ICP0	ICE1	ICE0	EG11	EG10	EG01	EG00	ICS01
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)	

【bit 7, bit 6】ICP1, ICP0

インプ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本ビットを "1" にセットします。割込み許可ビット (ICE0, ICE1) 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することができます。

本ビットは "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系命令では "1" が読めます。

0	有効エッジ検出なし (初期値)
1	有効エッジ検出あり

< 注意事項 >

ICP0: インพุットキャプチャ 0 に対応
ICP1: インพุットキャプチャ 1 に対応

【bit 5, bit 4】ICE1, ICE0

インพุットキャプチャ割込み許可ビットです。本ビットが "1" のとき割込みフラグ (ICP0, ICP1) 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

< 注意事項 >

ICE0: インพุットキャプチャ 0 に対応
ICE1: インพุットキャプチャ 1 に対応

【bit 3 ~ bit 0】EG11, EG10, EG01, EG00

外部入力の有効エッジ極性を指定します。インพุットキャプチャ動作許可も兼用しています。

EG11 EG01	EG10 EG00	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

< 注意事項 >

EG01, EG00: インพุットキャプチャ 0 に対応
EG11, EG10: インพุットキャプチャ 1 に対応

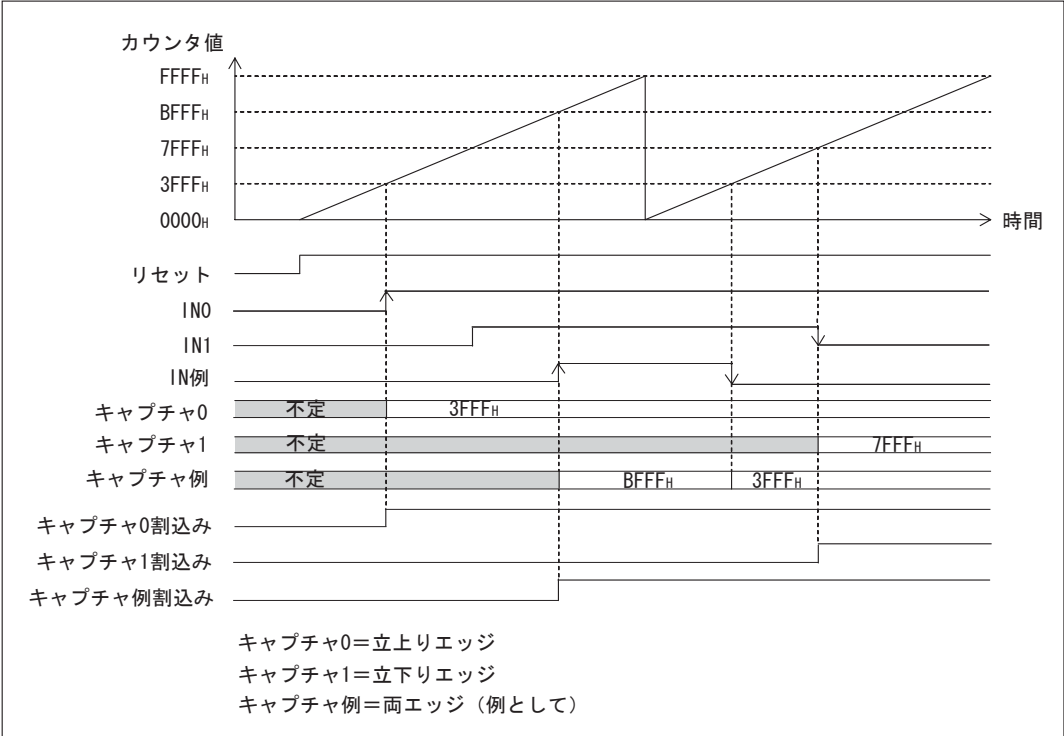
12.5.2 16 ビットインプットキャプチャの動作

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することができます。

■ インプットキャプチャの取込みタイミング例

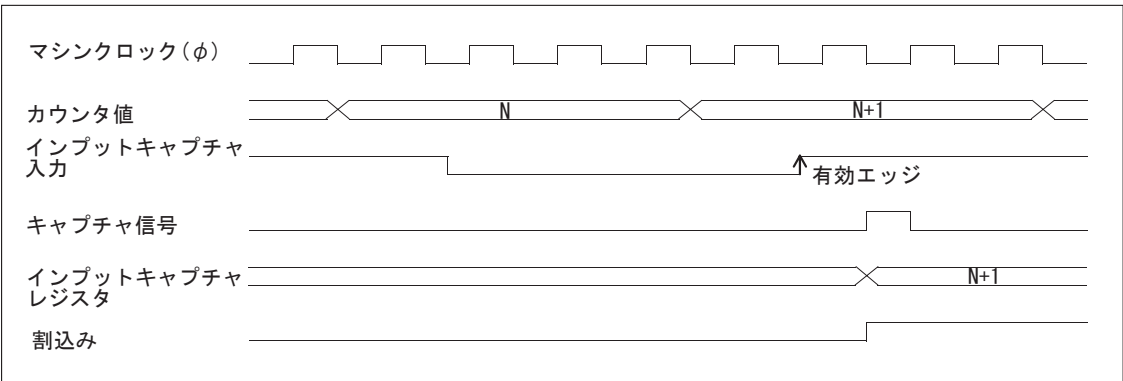
- キャプチャ 0= 立上りエッジ
- キャプチャ 1= 立下りエッジ
- キャプチャ例 = 両エッジ (例として)

図 12.5-4 インプットキャプチャの取込みタイミング例



■ インプットキャプチャ入力タイミング

図 12.5-5 入力信号に対するキャプチャタイミング



第13章

16 ビットリロードタイマ (イベントカウンタ機能付き)

16 ビットリロードタイマ (イベントカウンタ機能付き) の機能と動作について説明します。

- 13.1 16 ビットリロードタイマ (イベントカウンタ機能付き) の概要
- 13.2 16 ビットリロードタイマのレジスタ
- 13.3 16 ビットリロードタイマの内部クロック動作と外部クロック動作
- 13.4 16 ビットリロードタイマのアンダフロー動作
- 13.5 16 ビットリロードタイマの出力端子機能
- 13.6 カウンタの動作状態

13.1 16ビットリロードタイマ(イベントカウント機能付き) の概要

16 ビットリロードタイマは、16 ビットのダウンカウンタ、16 ビットのリロードレジスタ、入力端子 1 本 (TIN)、出力端子 1 本 (TOT)、制御レジスタで構成されています。入力クロックとして内部クロック 3 種類と外部クロックを選択できます。

■ 16 ビットリロードタイマ (イベントカウント機能付き) の概要

出力端子 (TOT) には、リロードモード時にはトグル出力波形を出力し、ワンショットモード時にはカウント中を示す矩形波を出力します。入力端子 (TIN) は、イベントカウントモード時にイベント入力となり、内部クロックモード時にはトリガ入力、またはゲート入力に使用することができます。

MB90540/545 シリーズの 16 ビットリロードタイマは 2 つです。

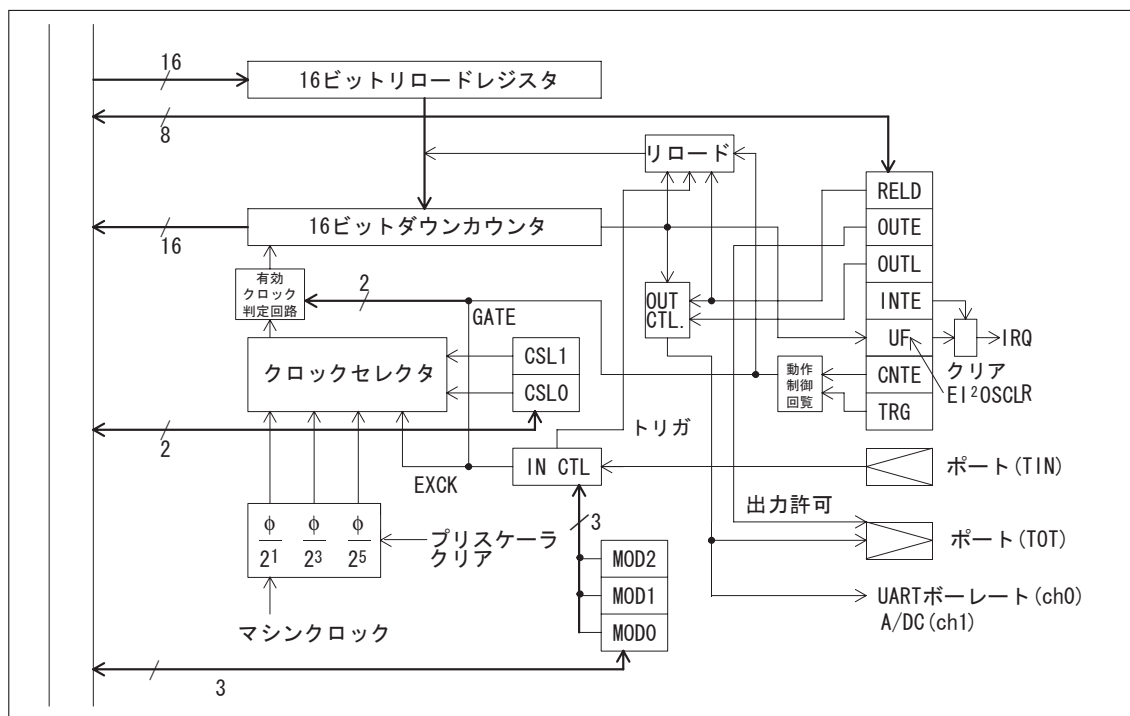
■ インテリジェント I/O サービス (EI²OS) 機能と割込み

本タイマは EI²OS 対応の回路を有しています。このため、本タイマのアンダフローにより EI²OS を起動させることができます。

■ 16 ビットリロードタイマのブロックダイヤグラム

図 13.1-1 に、16 ビットリロードタイマのブロックダイヤグラムを示します。

図 13.1-1 16 ビットリロードタイマのブロックダイヤグラム



■ 16 ビットリロードタイマの端子名

本タイマは2つ内蔵されており, 入力端子名 (TIN) と出力端子名 (TOT) は以下のようになります。

	入力端子名 (TIN)	出力端子名 (TOT)
タイマ 0	TIN0	TOT0
タイマ 1	TIN1	TOT1

13.2 16 ビットリロードタイマのレジスタ

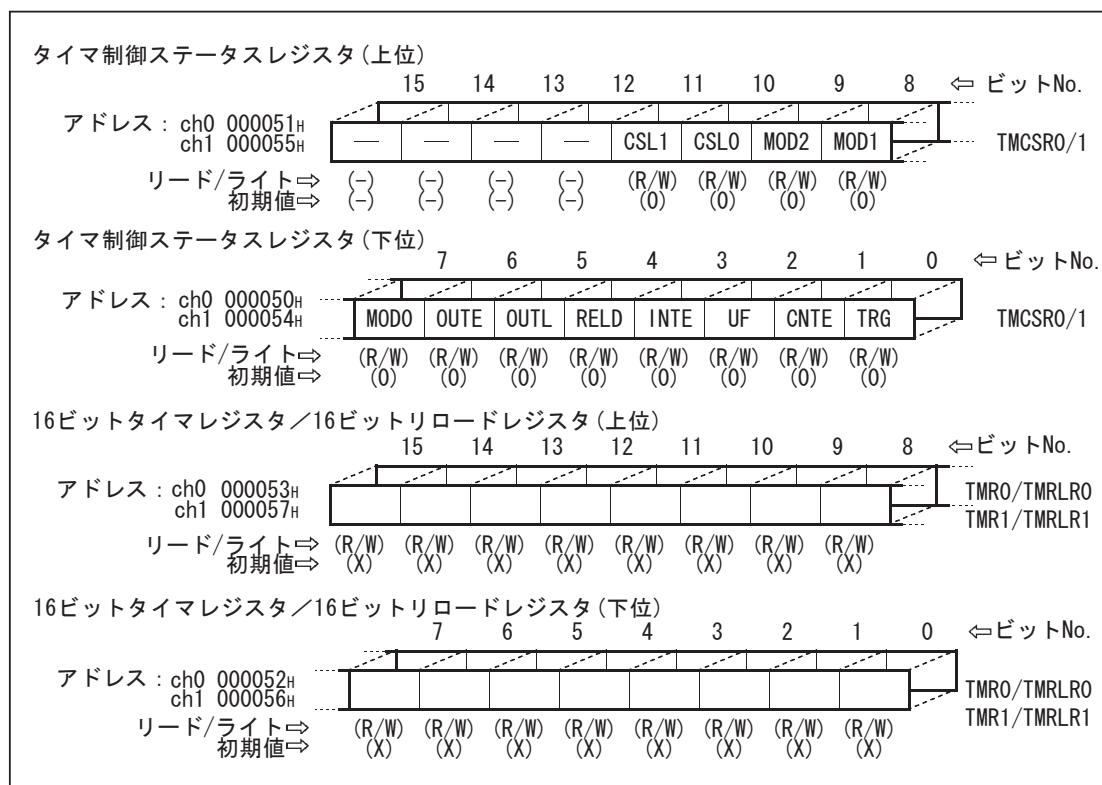
16 ビットリロードタイマのレジスタには、次の 2 種類があります。

- タイマ制御ステータスレジスタ
- 16 ビットタイマレジスタ / 16 ビットリロードレジスタ

■ 16 ビットリロードタイマのレジスタ

16 ビットリロードタイマのレジスタのビット構成を図 13.2-1 に示します。

図 13.2-1 16 ビットリロードタイマのレジスタ



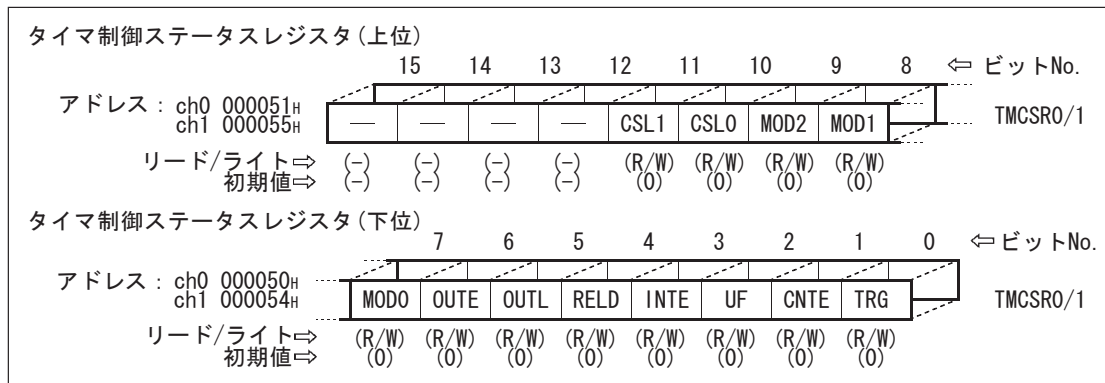
13.2.1 タイマ制御ステータスレジスタ (TMCSR)

16 ビットタイマの動作モードおよび割込みの制御をします。UF, CNTE, TRG ビット以外のビットの書換えは, CNTE=0 のときに行うようにしてください。

■ タイマ制御ステータスレジスタ (TMCSR)

タイマ制御ステータスレジスタ (TMCSR) のビット構成を図 13.2-2 に示します。

図 13.2-2 タイマ制御ステータスレジスタ (TMCSR)



【bit 11, bit 10】CSL1, CSL0(Clock SeLect1, 0)

カウントクロックセレクトビットです。選択されるクロックソースを, 表 13.2-1 に示します。

表 13.2-1 CSL ビット設定クロックソース

CSL1	CSL0	クロックソース (マシンサイクル $\phi=16$ MHz)
0	0	$\phi/2^1$ (0.125 μ s)
0	1	$\phi/2^3$ (0.5 μ s)
1	0	$\phi/2^5$ (2.0 μ s)
1	1	外部イベントカウントモード

【bit 9 ~ bit 7】MOD2, MOD1, MOD0

動作モードおよび入出力端子の機能を設定するビットです。

MOD2 ビットは, 入出力の機能を選択するビットです。"0" の場合, 入力端子はトリガ入力端子となり, 有効エッジが入力されるとリロードレジスタの内容をカウンタへロードし, カウント動作を継続します。"1" の場合, ゲートカウンタモードになり, 入力端子はゲート入力となり, 有効レベルが入力されている間のみカウントをします。

MOD1, MOD0 ビットは, 各モードにおける端子の機能の設定を行います。MOD2, MOD1, MOD0 ビットの設定を表 13.2-2 および表 13.2-3 に示します。

表 13.2-2 MOD2, MOD1, MOD0 ビットの設定方法 (1)

MOD2	MOD1	MOD0	入力端子機能	有効エッジ, レベル
0	0	0	トリガ禁止	
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	×	0	ゲート入力	"L" レベル
1	×	1		"H" レベル

内部クロックモード時 (CSL0, CSL1=00_B, 01_B, 10_B)

表 13.2-3 MOD2, MOD1, MOD0 ビットの設定方法 (2)

MOD2	MOD1	MOD0	入力端子機能	有効エッジ, レベル
×	0	0		
	0	1	トリガ入力	立上りエッジ
	1	0		立下りエッジ
	1	1		両エッジ

・ イベントカウントモード時 (CSL0, CSL1=11_B)

・ 表中の × は, 任意の値

【bit 6】 OUTE

出力許可ビットです。"0" のとき TOT 端子は汎用ポートになり, "1" のとき TOT 端子はタイマ出力端子になります。出力波形はリロードモード時トグル出力となり, ワンショットモード時はカウンタ中を示す矩形波出力になります。

【bit 5】 OUTL

TOT 端子の出力レベルを設定するビットです。

表 13.2-4 OUTE, RELD, OUTL の設定方法

OUTE	RELD	OUTL	出力波形
0	×	×	汎用ポート
1	0	0	カウント中 "H" の矩形波
1	0	1	カウント中 "L" の矩形波
1	1	0	カウントスタート時 "L" のトグル出力
1	1	1	カウントスタート時 "H" のトグル出力

・ 表中の × は, 任意の値

【bit 4】 RELD (REload)

リロード動作を許可するビットです。"1" のときはリロードモードになり, カウンタの値が "0000_H" "FFFF_H" へのアンダフローと同時にリロードレジスタの内容をカウンタへロードしてカウント動作を続けます。"0" のときは単発動作モードとなり, カウンタの値が "0000_H" "FFFF_H" へのアンダフローによりカウント動作を停止します。

【bit 3】INTE (INTerrupt Enable)

タイマ割込み要求許可ビットです。"1" のとき UF ビットが "1" になると割込み要求を発生します。"0" のときは UF ビットが "1" であっても割込み要求を発生しません。

【bit 2】UF (UnderFlow)

タイマ割込み要求フラグです。カウンタの値が, "0000_H" "FFFF_H" へのアンダフローにより "1" にセットされます。"0" の書込みまたはインテリジェント I/O サービスによってクリアされます。このビットへの "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系命令における読出し時には, "1" が読み出されます。

【bit 1】CNTE (CouNT Enable)

タイマのカウントイネーブルビットです。このビットに "1" を書き込むと, 起動トリガ待ち状態になります。"0" 書込みによりカウント動作は停止します。

【bit 0】TRG (Trigger)

ソフトウェアトリガビットです。"1" 書込みによりソフトウェアトリガがかかり, リロードレジスタの内容をカウンタへロードしてカウント動作を開始します。"0" 書込み動作は, 意味を持ちません。読出し値は, 常に "0" です。このレジスタによるトリガ入力, CNTE=1 のときのみ有効となります。CNTE=0 のときには, 動作に影響を与えません。

13.2.2 16 ビットタイマレジスタ (TMR)/16 ビットタイマリロードレジスタ (TMRLR) のレジスタ配置

TMR 内容 (リード時):

6 ビット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。初期値は不定です。このレジスタの読出しは、必ずワード転送命令で行ってください。

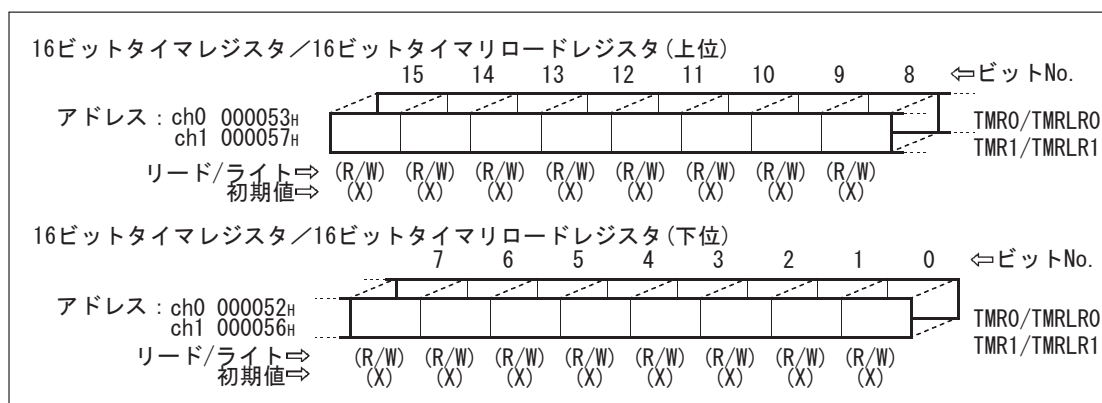
TMRLR 内容 (ライト時):

16 ビットタイマリロードレジスタは、カウンタの初期値を保持しておくレジスタです。初期値は不定です。このレジスタへの書込みは、必ずワード転送命令で行ってください。

■ 16 ビットタイマレジスタ (TMR)/16 ビットタイマリロードレジスタ (TMRLR)

16 ビットタイマレジスタ (TMR) と 16 ビットタイマリロードレジスタ (TMRLR) のビット構成を図 13.2-3 に示します。

図 13.2-3 16 ビットタイマレジスタ (TMR)/16 ビットタイマリロードレジスタ (TMRLR)



13.3 16 ビットリロードタイマの内部クロック動作と外部クロック動作

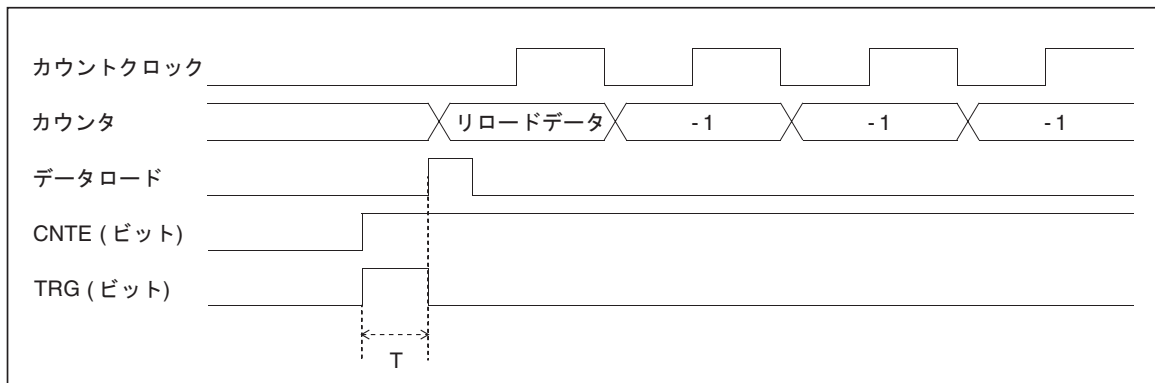
内部クロックの分周クロックでタイマを動作させる場合、クロックソースとしてマシンクロックの 2^1 , 2^3 , 2^5 分周のクロックから選択することができます。外部入力端子は、レジスタの設定により、トリガ入力またはゲート入力にすることができます。外部クロックをセレクトすると TIN 端子は、外部イベント入力端子となりレジスタで設定された有効エッジをカウントします。

■ 16 ビットリロードタイマの内部クロック動作

カウント許可と同時にカウント動作を開始したい場合は、制御レジスタの CNTE ビットと TRG ビットの両方に "1" を書き込んでください。TRG ビットによるトリガ入力は、タイマが起動状態のとき (CNTE=1) 動作モードにかかわらず常に有効です。

図 13.3-1 に、カウンタの起動およびカウンタの動作について示します。カウンタスタートのトリガが入力されてからリロードレジスタのデータがカウンタへロードされるまでに、 T (T: マシンサイクル) の時間がかかります。

図 13.3-1 16 ビットリロードタイマのカウンタの起動および動作

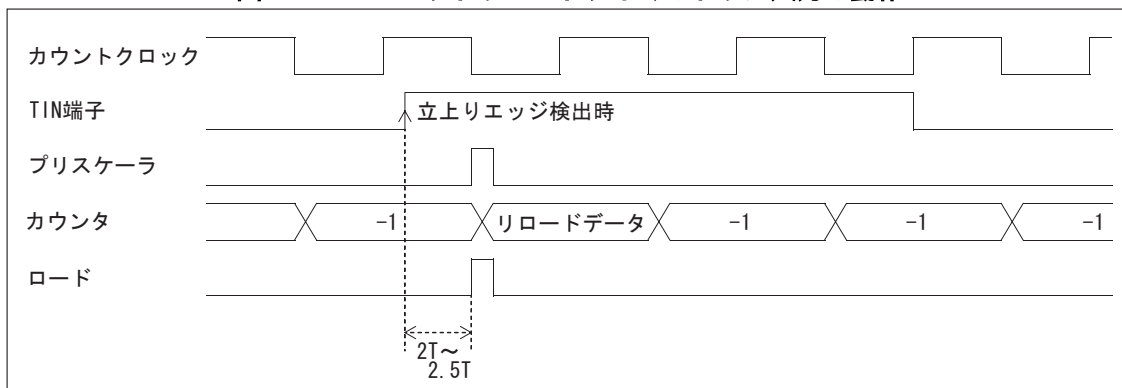


■ 16 ビットリロードタイマの入力端子機能 (内部クロックモード時)

クロックソースとして、内部クロックを選択した場合、TIN 端子はトリガ入力またはゲート入力として使用することができます。トリガ入力として使用した場合、有効エッジが入力されるとリロードレジスタの内容をカウンタにロードして、内部プリスケアラをクリアした後、カウント動作を開始します。

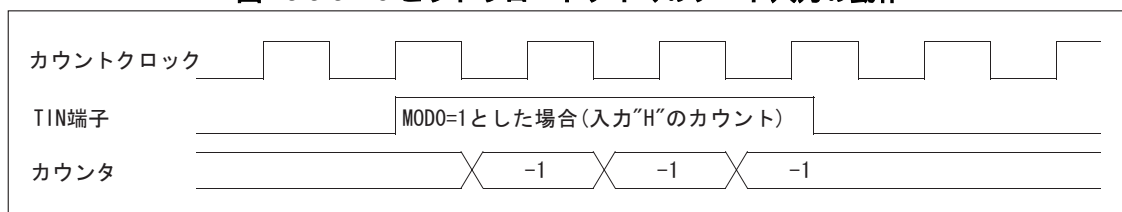
TIN 端子は、 $2T$ (T はマシンサイクル) 以上のパルスを入力してください。図 13.3-2 に、トリガ入力の動作を示します。

図 13.3-2 16 ビットリロードタイマのトリガ入力の動作



ゲート入力として使用する場合、制御レジスタの MOD0 ビットにより設定される有効レベルが、TIN端子から入力されている間のみカウントをします。このときカウントクロックは、止まらずに動き続けます。ゲートモード時のソフトウェアトリガは、ゲートレベルにかかわらず可能です。TIN 端子のパルス幅は、 $2 \cdot T$ (T はマシンサイクル) 以上にしてください。図 13.3-3 に、ゲート入力の動作を示します。

図 13.3-3 16 ビットリロードタイマのゲート入力の動作



■ 外部イベントカウント

外部クロックをセレクトすると TIN 端子は、外部イベント入力端子となりレジスタで設定された有効エッジをカウントします。TIN 端子のパルス幅は、 $4T$ (T はマシンサイクル) 以上にしてください。

13.4 16 ビットリロードタイマのアンダフロー動作

16 ビットリロードタイマにより, カウンタ値が "0000_H" から "FFFF_H" に変化したときにアンダフローが定義されますので, (リロードレジスタ設定 +1) カウントの後でアンダフローが生じます。

■ 16 ビットリロードタイマのアンダフロー動作

アンダフロー発生時, 制御レジスタの RELD ビットが "1" のときリロードレジスタの内容をカウンタへロードしてカウント動作を継続し, "0" のときカウンタを "FFFF_H" で停止させます。

アンダフローが発生すると, 制御レジスタの UF ビットをセットし, このときに INTE ビットが "1" であると割込み要求が発生します。

図 13.4-1 と図 13.4-2 に, アンダフロー発生時の動作を示します。

図 13.4-1 16 ビットリロードタイマのアンダフロー動作 [RELD=1 の場合]

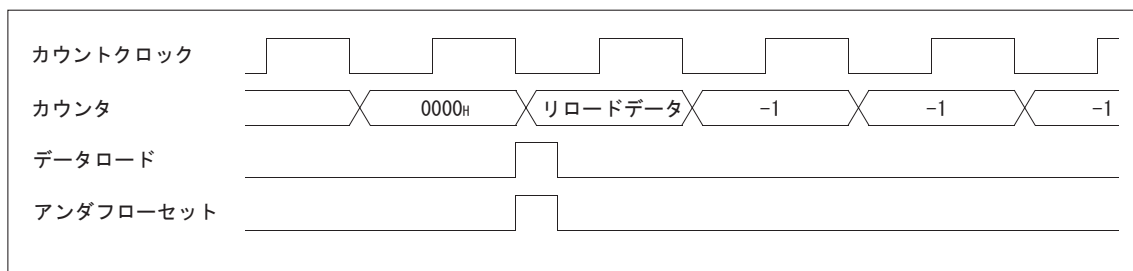
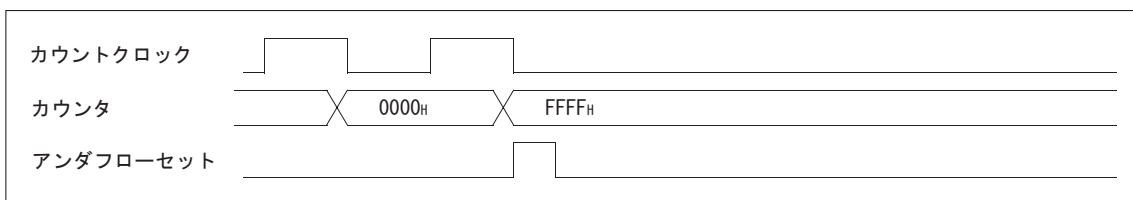


図 13.4-2 16 ビットリロードタイマのアンダフロー動作 [RELD=0 の場合]



13.5 16 ビットリロードタイマの出力端子機能

TOT 端子はリロードモード時には, アンダフローにより反転するトグル出力として, ワンショットモード時には, カウント中を示すパルス出力として機能します。

■ 16 ビットリロードタイマの出力端子機能

16 ビットリロードタイマの出力極性は, レジスタの OUTL ビットにより設定でき OUTL=0 のときトグル出力は, 初期値が "0" で, ワンショットパルス出力は, カウント中 "1" を出力します。OUTL=1 にすると出力波形は, 反転します。図 13.5-1 と図 13.5-2 に出力端子機能を示します。

図 13.5-1 16 ビットリロードタイマの出力端子機能 (RELD=1, OUTL=0 の場合)

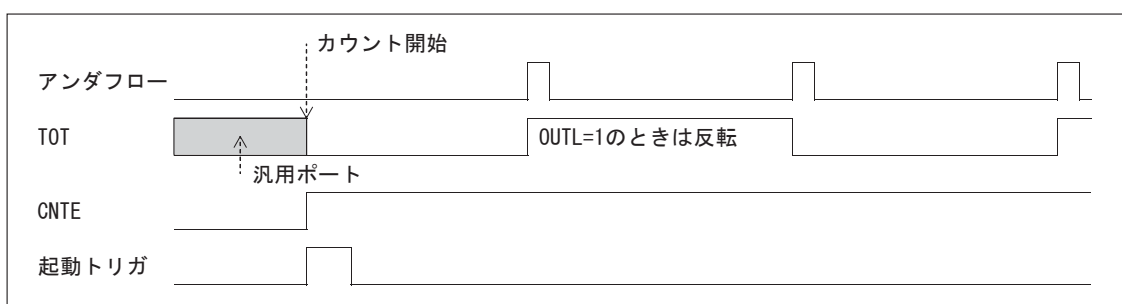
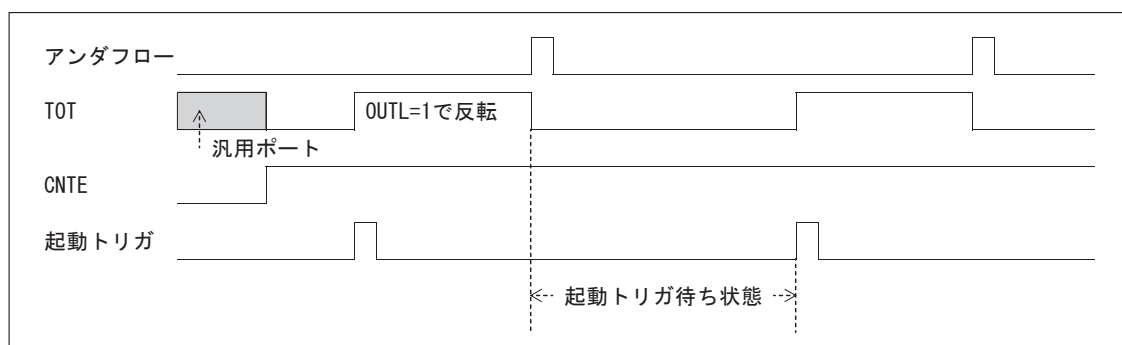


図 13.5-2 16 ビットリロードタイマの出力端子機能 (RELD=0, OUTL=0 の場合)



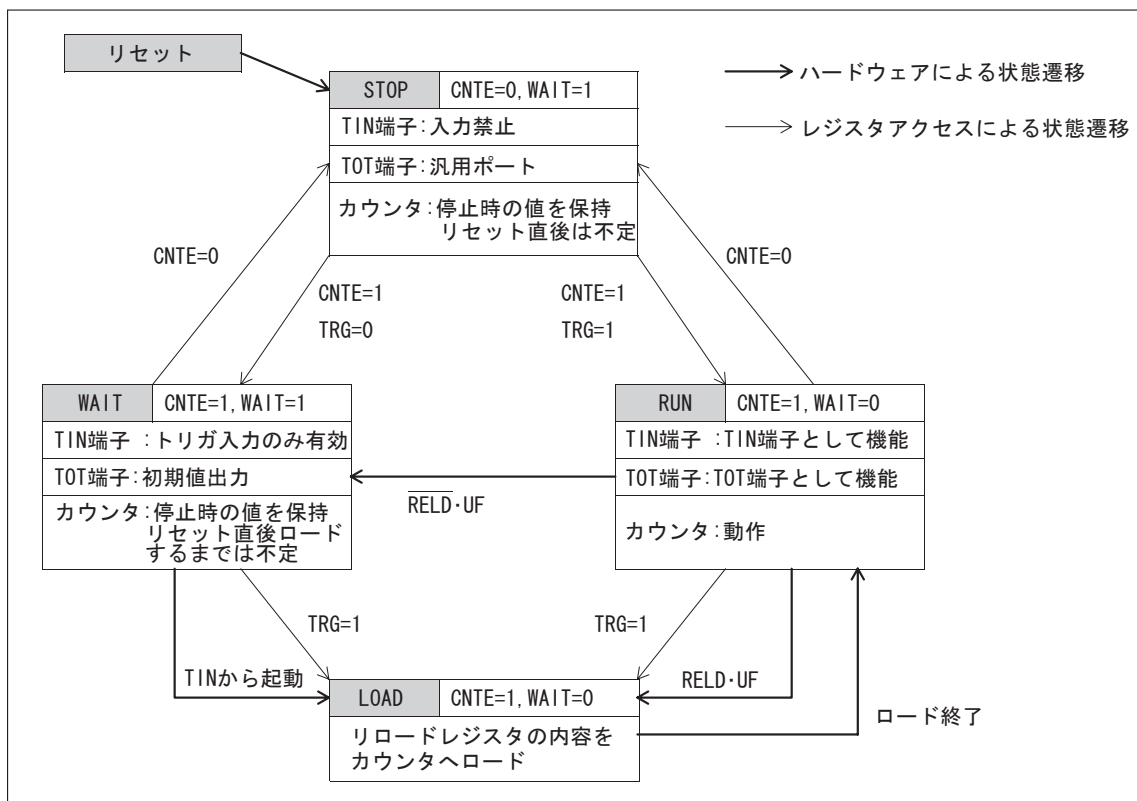
13.6 カウンタの動作状態

カウンタの状態は、制御レジスタの CNTE ビットと内部信号の WAIT 信号によって決まります。設定可能な状態として CNTE=0, WAIT=1 の停止状態 (STOP 状態), CNTE=1, WAIT=1 の起動トリガ待ち状態 (WAIT 状態), CNTE=1, WAIT=0 の動作状態 (RUN 状態) があります。

■ カウンタの動作状態

図 13.6-1 に、各状態の遷移を示します。

図 13.6-1 カウンタの状態遷移



第14章

8/16 ビット PPG

8/16 ビット PPG の機能と動作について説明します。

14.1 8/16 ビット PPG の概要

14.2 8/16 ビット PPG のブロックダイアグラム

14.3 8/16 ビット PPG のレジスタ

14.4 8/16 ビット PPG の動作

14.5 8/16 ビット PPG のカウントクロックの選択

14.6 8/16 ビット PPG のパルスの端子出力の制御

14.7 8/16 ビット PPG の割込み

14.8 8/16 ビット PPG の各ハードウェアの初期値

14.1 8/16 ビット PPG の概要

8/16 ビット PPG は、ハードウェアとして、8 ビットダウンカウンタ 2 個、8 ビットリロードレジスタ 4 個、16 ビット制御レジスタ 1 個、外部パルス出力端子 2 本、割込み出力 2 本を有し、これらにより以下の機能を実現します。

■ 8/16 ビット PPG の機能

- 8 ビット PPG 出力 2 チャンネル独立動作モード

2 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

1 チャンネルの 16 ビットの PPG 出力動作が可能です。

- 8 ビットプリスケアラ +8 ビット PPG 出力動作モード

ch0 の出力を ch1 のクロック入力とすることにより、任意周期の 8 ビット PPG 出力動作が可能です。

- PPG 出力動作

任意周期・デューティ比のパルス波を出力します。外付け回路により D/A コンバータとしても使用可能です。

PPG の ch0 と ch1 を合わせて 1 ユニットとよびます。

MB90540/545 シリーズには、PPG が 4 ユニットあります。次節では PPG の ch0 と ch1 の機能についてのみ記述します。残りの PPG も同じ機能を持ちますが、レジスタアドレスについては、「付録 A I/O マップ」を参照してください。

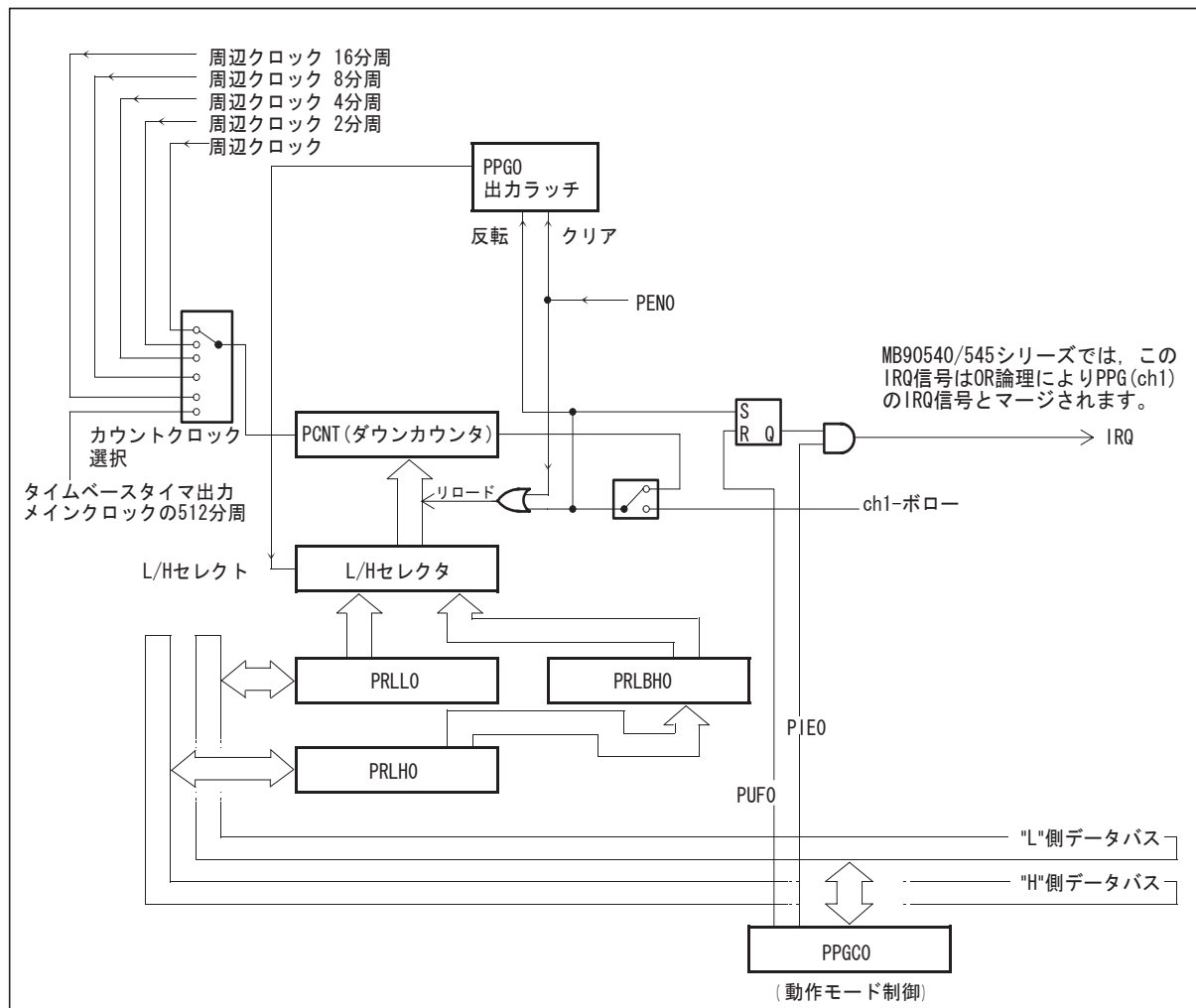
PPG の ch0 を PPG (ch0)、ch1 を PPG (ch1) と表します。

14.2 8/16 ビット PPG のブロックダイアグラム

図 14.2-1 に、8/16 ビット PPG(ch0) のブロックダイアグラムを、図 14.2-2 に 8/16 ビット PPG(ch1) のブロックダイアグラムを示します。

■ 8/16 ビット PPG のブロックダイアグラム

図 14.2-1 8 ビット PPG(ch0) のブロックダイアグラム



ch0 の PPG 出力信号は外部端子に接続していません。

図 14.2-2 8 ビット PPG(ch1) のブロックダイアグラム

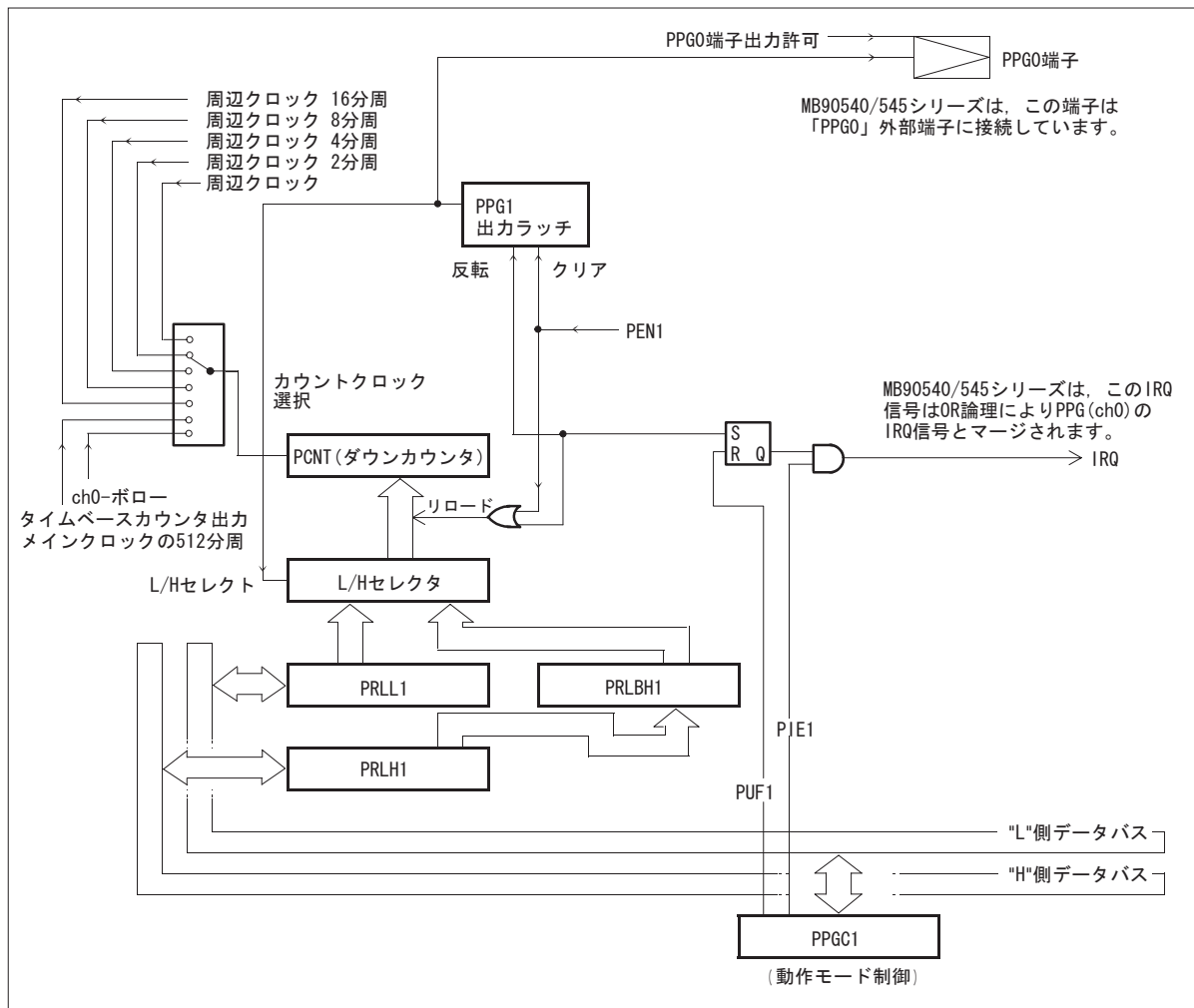
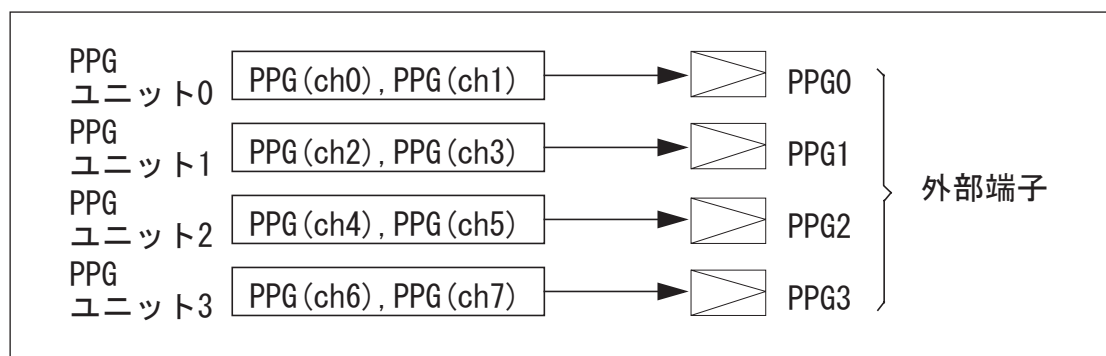


図 14.2-3 PPG モジュールとユニット番号、および外部端子の関係



14.3 8/16 ビット PPG のレジスタ

8/16 ビット PPG のレジスタには、次の 5 つがあります。

- PPG0 動作モード制御レジスタ
- PPG1 動作モード制御レジスタ
- PPG ユニット 0 クロック選択レジスタ
- リロードレジスタ H
- リロードレジスタ L

■ 8/16 ビット PPG のレジスタ

8/16 ビット PPG のレジスタのビット構成を図 14.3-1 に示します。

図 14.3-1 8/16 ビット PPG のレジスタ

PPG0動作モード制御レジスタ		7	6	5	4	3	2	1	0	↔ビットNo.
アドレス : ch0 000038H		PEN0	—	PE00	PIE0	PUF0	—	—	予約	PPG0
リード/ライト ⇒		(R/W)	(—)	(R/W)	(R/W)	(R/W)	(—)	(—)	(W)	
初期値 ⇒		(0)	(—)	(0)	(0)	(0)	(—)	(—)	(1)	

PPG1動作モード制御レジスタ		15	14	13	12	11	10	9	8	ビットNo.
アドレス : ch1 000039H		PEN1	—	PE10	PIE1	PUF1	MD1	MD0	予約	PPG1
リード/ライト ⇒		(R/W)	(—)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(W)	
初期値 ⇒		(0)	(—)	(0)	(0)	(0)	(0)	(0)	(1)	

PPGユニット0クロック選択レジスタ		7	6	5	4	3	2	1	0	↔ビットNo.
アドレス : ch0/1 00003AH		PCS2	PCS1	PCS0	PCM2	PCM1	PCM0	—	—	PPG0/PPG1
リード/ライト ⇒		(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(—)	(—)	
初期値 ⇒		(0)	(0)	(0)	(0)	(0)	(0)	(—)	(—)	

リロードレジスタH		15	14	13	12	11	10	9	8	↔ビットNo.
アドレス : ch0 003901H ch1 003903H										PRLH0/PRLH1
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 ⇒		(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

リロードレジスタL		7	6	5	4	3	2	1	0	↔ビットNo.
アドレス : ch0 003900H ch1 003902H										PRLLO/PRLLO1
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 ⇒		(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

14.3.1 PPG0 動作モード制御レジスタ (PPGC0)

動作モード制御レジスタ (PPGC0) は、ブロックの動作モードの選択，端子出力制御，カウントクロック選択，トリガの制御を行う 5 ビットの制御レジスタです。PPG(ch0) を制御します。

■ PPG0 動作モード制御レジスタ (PPGC0)

PPG0 動作モード制御レジスタ (PPGC0) のビット構成を図 14.3-2 に示します。

図 14.3-2 PPG0 動作モード制御レジスタ (PPGC0)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : ch0 000038 _H	PEN0		PE00	PIE0	PUF0			予約	PPGC0
リード / ライト	(R/W)	(-)	(R/W)	(R/W)	(R/W)	(-)	(-)	(W)	
初期値	(0)	(-)	(0)	(0)	(0)	(-)	(-)	(1)	

【bit 7】PEN0(PPG enable): 動作許可ビット

PPG(ch0) のカウンタ動作を許可します。

PEN0	動作状態
0	動作停止 ("L" レベル出力保持)
1	PPG(ch0) 動作許可

本ビットに "1" を書き込むことで，PPG(ch0) はカウントを開始します。

リセットにより，本ビットは "0" に初期化されます。リード / ライトが可能です。

【bit 5】PE00 (PPG output enable 00): PPG0 端子出力許可ビット

パルス出力外部端子 PPG0 を以下のように制御します。

PE00	動作状態
0	汎用ポート端子 (パルス出力禁止)
1	PPG0= パルス出力端子 (パルス出力許可)

リセットにより，本ビットは "0" に初期化されます。本ビットは，リード / ライトが可能です。

MB90540/545 シリーズについては，本ビットは常に "0" に設定してください。

【bit 4】PIE0 (PPG Interrupt enable): PPG の割込み許可ビット

PPG(ch0) の割込み許可を以下のように制御します。

PIE0	動作状態
0	割込み禁止
1	割込み許可

本ビットが "1" のとき，PUF0 が "1" になると割込み要求を発生します。本ビットが "0" のときは割込み要求を発生しません。

リセットにより，本ビットは "0" に初期化されます。本ビットは，リード / ライトが可能です。

【bit 3】PUF0 (PPG underflow flag):PPG カウンタアンダフロービット

PPG(ch0) のカウンタアンダフロービットを以下のように制御します。

PUF0	動作状態
0	PPG(ch0) のカウンタアンダフローが検出していません
1	PPG(ch0) のカウンタアンダフローを検出しました

8 ビット PPG 2 チャンネルモードおよび 8 ビットプリスケアラ +8 ビット PPG モードのときには, ch0 のカウンタの値が "00_H" ~ "FF_H" へ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 PPG モードのときには, ch1/ch0 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り "1" にセットされます。"0" の書込みにより "0" になります。このビットへの "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系命令のリード時には "1" が読まれます。

リセットにより, 本ビットは "0" に初期化されます。本ビットは, リード / ライトが可能です。

【bit 0】

予約ビットです。必ず "1" に設定してください。

本ビットは常に "1" が読まれます。

14.3.2 PPG1 動作モード制御レジスタ (PPGC1)

PPG1 動作モード制御レジスタ (PPGC1) は、ブロックの動作モードの選択、端子出力制御、カウントクロック選択、トリガの制御を行う 7 ビットの制御レジスタです。PPG(ch1) の制御と PPG ユニット 0 の動作モードを選択します。

■ PPG1 動作モード制御レジスタ (PPGC1)

PPG1 動作モード制御レジスタ (PPGC1) のビット構成を図 14.3-3 に示します。

図 14.3-3 PPG1 動作モード制御レジスタ (PPGC1)

アドレス :	15	14	13	12	11	10	9	8	ビット No.
ch1 000039 _H	PEN1		PE10	PIE1	PUF1	MD1	MD0	予約	PPGC1
リード / ライト	(R/W)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(W)	
初期値	(0)	(-)	(0)	(0)	(0)	(0)	(0)	(1)	

【bit 15】PEN1(PPG enable): 動作許可ビット

PPG(ch1) のカウンタ動作を許可します。

PEN1	動作状態
0	動作停止 ("L" レベル出力保持)
1	PPG(ch1) 動作イネーブル

本ビットを "1" に設定することでカウンタ動作を許可します。

リセットにより、本ビットは "0" に初期化されます。リード / ライトが可能です。

【bit 13】PE10 (PPG output enable 10): PPG0 端子出力許可ビット

パルス出力外部端子 PPG0 を以下のように制御します。

PE10	動作状態
0	汎用ポート端子 (パルス出力禁止)
1	PPG0= パルス出力端子 (パルス出力許可)

リセットにより、本ビットは "0" に初期化されます。本ビットは、リード / ライトが可能です。

【bit 12】PIE1 (PPG Interrupt enable): PPG の割込み許可ビット

PPG(ch1) の割込み許可を以下のように制御します。

PIE1	動作状態
0	割込み禁止
1	割込み許可

本ビットが "1" のとき、PUF1 が "1" になると割込み要求を発生します。本ビットが "0" のときは割込み要求を発生しません。

リセットにより、本ビットは "0" に初期化されます。本ビットは、リード / ライトが可能です。

【bit 11】PUF1 (PPG underflow flag):PPG カウンタアンダフロービット

PPG(ch1) のカウンタアンダフロービットを以下のように制御します。

PUF1	動作状態
0	PPG(ch1) のカウンタアンダフローを検出していない
1	PPG(ch1) のカウンタアンダフローを検出した

8 ビット PPG 2 チャンネルモードおよび 8 ビットプリスケラ +8 ビット PPG モードのときには, ch1 のカウンタの値が "00_H" ~ "FF_H" へ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 PPG モードのときには, ch1/ch0 のカウンタの値が "0000_H" ~ "FFFF_H" へなったときのアンダフローにより "1" にセットされます。"0" の書込みにより "0" になります。このビットへの "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系命令のリード時には "1" が読めます。

リセットにより, 本ビットは "0" に初期化されます。本ビットは, リード / ライトが可能です。

【bit 10, bit 9】MD1, MD0 (PPG count mode): 動作モード選択ビット

PPG ユニットの動作モードを以下のように選択します。

MD1	MD0	動作モード
0	0	8 ビット PPG2 チャンネル独立モード
0	1	8 ビットプリスケラ +8 ビット PPG1 チャンネルモード
1	0	予約 (設定禁止)
1	1	16 ビット PPG1 チャンネルモード

リセットにより, 本ビットは "00" に初期化されます。本ビットは, リード / ライトが可能です。

< 注意事項 >

- ビットを "10_B" に設定しないでください。
- 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_B" または "00_B" にセットすることを推奨します。
- 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 ビットおよび PEN1 ビットを同時に "11_B" または "00_B" にセットしてください。

【bit 8】予約ビット

予約ビットです。PPGC1 を設定するときは必ず "1" に設定してください。

本ビットは常に "1" が読めます。

14.3.3 PPG ユニット 0 クロック選択レジスタ (PPG01)

PPG ユニット 0 クロック選択レジスタ (PPG01) は、PPG の動作クロックの制御を行う 8 ビットの制御レジスタです。

■ PPG ユニット 0 クロック選択レジスタ (PPG01)

PPG ユニット 0 クロック選択レジスタ (PPG01) のビット構成を図 14.3-4 に示します。

図 14.3-4 PPG ユニット 0 クロック選択レジスタ (PPG01)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : ch0/1 00003A _H	PCS2	PCS1	PCS0	PCM2	PCM1	PCM0			PPG01
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	()	()	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	()	()	

【bit 7 ~ bit 5】PCS2 ~ PCS0 (PPG count select): カウントクロック選択ビット
ch1 のダウンカウンタの動作クロックを、次のように選択します。

PCS2	PCS1	PCS0	動作モード
0	0	0	周辺クロック (62.5 ns マシニングクロック 16 MHz 時)
0	0	1	周辺クロック /2(125 ns マシニングクロック 16 MHz 時)
0	1	0	周辺クロック /4(250 ns マシニングクロック 16 MHz 時)
0	1	1	周辺クロック /8(500 ns マシニングクロック 16 MHz 時)
1	0	0	周辺クロック /16(1 μs マシニングクロック 16 MHz 時)
1	1	1	タイムベースタイマからの入力クロック (128 μs 原発振 4 MHz 時)

リセットにより本ビットは "000_B" に初期化されます。本ビットは、リード / ライトが可能です。

< 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 モードのときには、ch1 の PPG は、ch0 からカウントクロックを受けて動作するため、これらのビットの設定は、無効になります。

【bit 4 ~ bit 2】PCM2 ~ PCM0 (PPG Count Mode): カウントクロック選択ビット
ch0 のダウンカウンタの動作クロックを、次のように選択します。

PCM2	PCM1	PCM0	動作モード
0	0	0	周辺クロック (62.5 ns マシニングクロック 16 MHz 時)
0	0	1	周辺クロック /2(125 ns マシニングクロック 16 MHz 時)
0	1	0	周辺クロック /4(250 ns マシニングクロック 16 MHz 時)
0	1	1	周辺クロック /8(500 ns マシニングクロック 16 MHz 時)
1	0	0	周辺クロック /16(1 μs マシニングクロック 16 MHz 時)
1	1	1	タイムベースタイマからの入力クロック (128 μs 原発振 4 MHz 時)

リセットにより本ビットは "000_B" に初期化されます。本ビットは、リード / ライトが可能です。

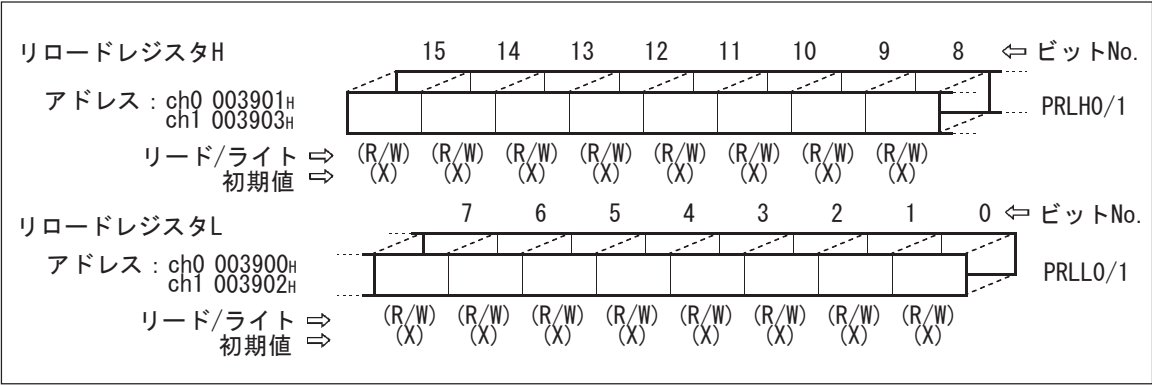
14.3.4 リロードレジスタ (PRLH, PRLH)

リロードレジスタ (PRLH, PRLH) 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。PRLH, PRLH レジスタは、どちらもリード/ライトが可能です。

■ リロードレジスタ (PRLH, PRLH)

リロードレジスタ (PRLH, PRLH) のビット構成を図 14.3-5 に示します。

図 14.3-5 リロードレジスタ (PRLH, PRLH)



レジスタ名	機能
PRLH	"L" 側リロード値保持。"L" のパルス幅を設定します。
PRLH	"H" 側リロード値保持。"H" のパルス幅を設定します。

< 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、ch0 の PRLH と PRLH に異なる値を設定すると、ch1 の PPG 波形がサイクルごとに異なる場合があるので、ch0 の PRLH と PRLH は、同じ値に設定することを推奨します。

14.4 8/16 ビット PPG の動作

8/16 ビット PPG には、8 ビット長の PPG ユニットが 2 チャンネル有り、2 チャンネル独立モード以外に、8 ビットプリスケアラ +8 ビット PPG モードと、16 ビット PPG 1 チャンネルモードの計 3 種類の動作を行うことができます。

■ 8/16 ビット PPG の動作

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 (PRL, PRLH)。この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、カウンタのボロー発生によるリロード時に、PPG0 出力端子の値を反転させます。この動作により、PPG0 出力端子はリロードレジスタ値に対応した "L" 幅 / "H" 幅をもつパルスを出力します。

表 14.4-1 に、リロード動作とパルス出力の関係を示します。

表 14.4-1 リロード動作とパルス出力の関係

リロード動作		端子出力変化	
PRLH	PCNT	PPG0 出力端子 [0 1]	立上り
PRL	PCNT	PPG0 出力端子 [1 0]	立下り

また、PPGC0 中の bit4 (PIE0) が "1" のときおよび PPGC1 中の bit12 (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 モードの場合には、"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。

■ 8/16 ビット PPG の動作モード

8/16 ビット PPG の動作モードには、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 1 チャンネルモードの計 3 種類の動作モードがあります。

● 2 チャンネル独立モード

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動作モードです。

● 8 ビットプリスケアラ +8 ビット PPG モード

8 ビットプリスケアラ +8 ビット PPG モードは、ch0 を 8 ビットプリスケアラとして動作させ、ch1 を ch0 のボロー出力でカウントすることにより、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。

● 16 ビット PPG1 チャンネルモード

16 ビット PPG 1 チャンネルモードは、ch0 と ch1 を連結させて、16 ビットの PPG として動作させる動作モードです。

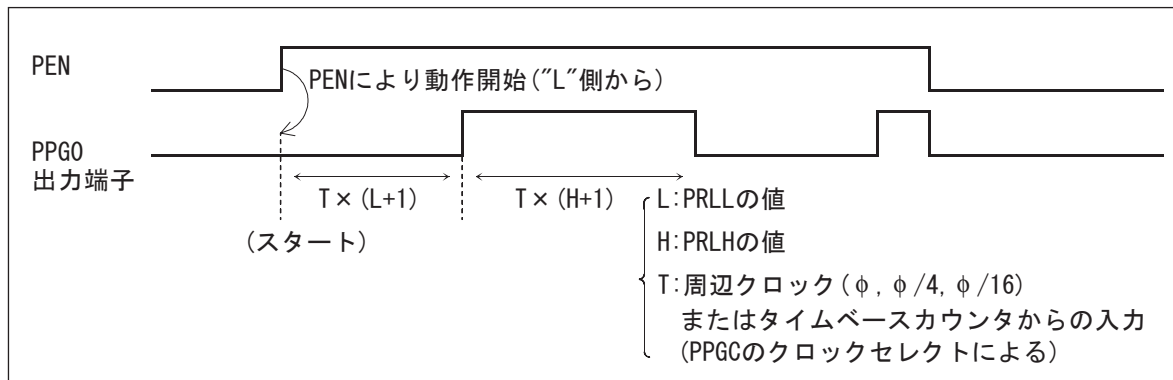
■ 8/16 ビット PPG の出力動作

8/16 ビット PPG は、ch0 の PPG については、PPGC0 レジスタの bit7 (PEN0) を、ch1 の PPG については、PPGC1 レジスタの 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、パルス出力は "L" レベルを保持します。MB90540/545 シリーズについては、ch0 の PPG からの出力信号は外部端子には接続していません。

8 ビットプリスケラ +8 ビット PPG モードのときには、ch0 を停止状態で、ch1 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、PPGC0 レジスタの bit7 (PEN0) と PPGC1 レジスタの bit15 (PEN1) が同時に開始または停止していることを確認してください。次の図は、PPG 出力動作を示しています。PPG の動作中、パルス波は振動数と効率の割合 ("H" レベルと "L" レベルのパルス波である時間の割合) で出力し続け、PPG は停止が指定されるまで動作します。

図 14.4-1 PPG 出力動作の出力波形



■ 8/16 ビット PPG の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掛けた値が、出力されるパルスの幅となります。

8 ビット PPG 動作時のリロードレジスタ値が "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もつことと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、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 周期分のパルス幅をもつこととなりますので注意してください。パルス幅の計算例を次に示します。

$$\begin{aligned}
 P_L &= T \times (L+1) \\
 P_H &= T \times (H+1)
 \end{aligned}
 \quad \left\{ \begin{array}{l}
 L : \text{PRL の値} \\
 H : \text{PRLH の値} \\
 T : \text{入力クロック周期} \\
 P_H : \text{"H"パルスの幅} \\
 P_L : \text{"L"パルスの幅}
 \end{array} \right.$$

14.5 8/16 ビット PPG のカウントクロックの選択

8/16 ビット PPG 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。

■ 8/16 ビット PPG のカウントクロックの選択

PPG ユニット 0 クロック選択レジスタの bit4 ~ bit2 (PCM2 ~ PCM0) で ch0 のクロックを、bit7 ~ bit5 (PCS2 ~ PCS0) で ch1 のクロックを選択します。

クロックは、マシンのクロックの 1/16 ~ 1 倍の周辺クロックとタイムベースカウンタからの入力クロックより選択できます。

しかし、8 ビットプリスケアラ +8 ビット PPG モードまたは 16 ビット PPG モードのときには、PCS2 に "0" を設定しても意味はありません。

タイムベースカウンタの入力を使用した場合、トリガにより起動がかかる最初およびストップ後の最初のカウント周期がずれる可能性がありますので注意してください。また、本モジュール動作中にタイムベースカウンタのクリアを行うと周期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 モードでは ch0 が動作状態、ch1 が停止状態であるときに ch1 の起動を行うと、最初のカウント周期がずれる可能性がありますので注意してください。

14.6 8/16 ビット PPG のパルスの端子出力の制御

本モジュールの動作によって生成されたパルス出力は、外部端子 PPG0 より出力させることができます。

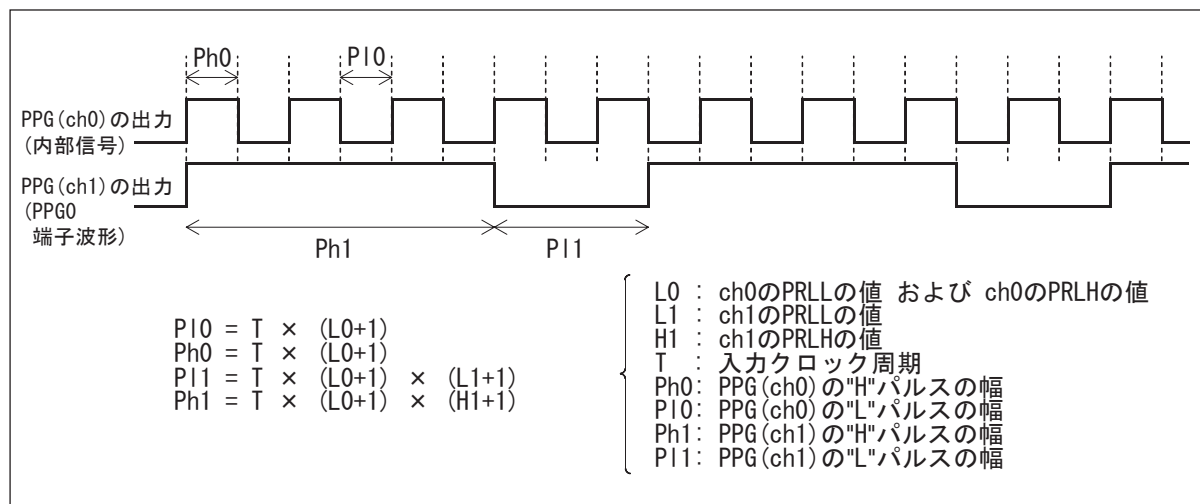
■ 8/16 ビット PPG のパルスの端子出力の制御

PPG1 動作モード制御レジスタ (PPGC1) の bit13 (PE10) が "0" のとき (初期値) は、パルス出力は外部端子より出力されずに、汎用ポートとして機能します。本ビットに "1" を設定すると、パルス出力が外部端子より出力されます。

8 ビットプリスケアラ +8 ビット PPG モードでは、PPG(ch0) の 8 ビットプリスケアラのトグル波形が出力され、PPG(ch1) の 8 ビット PPG の波形が出力されます。このモードのときの出力波形の例を図 14.6-1 に示します。

MB90540/545 シリーズについては、ch0 の PPG からの出力信号は外部端子には接続していません。

図 14.6-1 8 ビットプリスケアラ +8 ビット PPG 出力動作の出力波形



< 注意事項 >

ch0 の PRLL と ch0 の PRLH は、同じ値を設定することを推奨します。

14.7 8/16 ビット PPG の割込み

8/16 ビット PPG は、リロード値がカウントアウトし、ポローが発生したときに割込み要求を出力します。

■ 8/16 ビット PPG の割込み

8 ビット PPG 2 チャンネルモードおよび 8 ビットプリスケアラ +8 ビット PPG モードのときには、それぞれのカウンタのポローにより、それぞれの割込みの要求が行われますが、16 ビット PPG モードでは、16 ビットカウンタのポローにより、PUF0 と PUF1 が同時にセットされます。このため、割込み要因を一本化するために、PIE0 または PIE1 のどちらか一方のみを許可にすることを推奨します。また、割込みフラグのクリアも PUF0 と PUF1 を同時に行うことを推奨します。

14.8 8/16 ビット PPG の各ハードウェアの初期値

8/16 ビット PPG の各ハードウェアは、リセット時に以下のように初期化されます。

■ 8/16 ビット PPG の各ハードウェアの初期値

● レジスタ

- PPGC0 0-000--1_B
- PPGC1 0-000001_B
- PPG01 000000--_B

● パルス出力

PPG0 端子は出力禁止となります。

出力を許可した場合は "L" 出力となります。

● 割込み要求

割込み禁止となります。なお、リロード値は保持されます。

< 注意事項 >

8/16 ビット PPG のリロードレジスタ (PRL, PRLH) へのライトタイミング 16 ビット PPG モード以外のモードの場合には、リロードレジスタ (PRL, PRLH) への書込みはワード転送命令を使用することを推奨します。なお、バイト転送命令 2 回にて書き込んだ場合、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ので注意してください。

図 14.8-1 8/16 ビット PPG のリロードレジスタ (PRL, PRLH) へのライトタイミングチャート

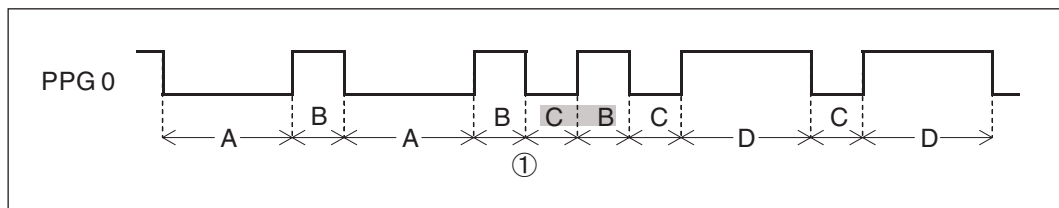
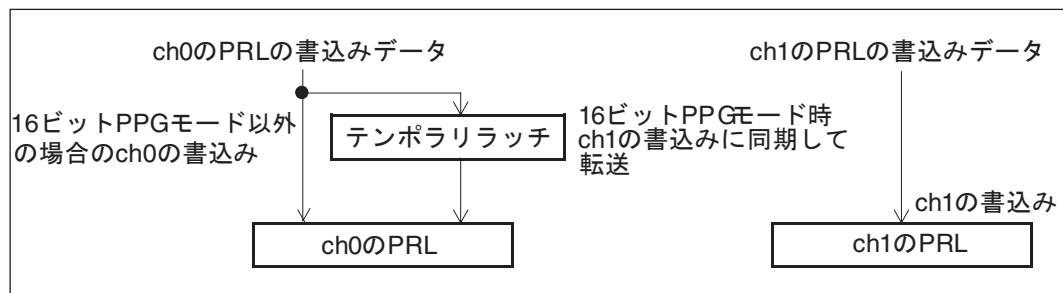


図 14.8-1 のタイムチャートにおいて、①のタイミングの前で PRL を A から C に書き換えて、②の後 PRLH の値を B から D に書き換えた場合、③のタイミングでの PRL の値は PRL=C, PRLH=B のため 1 回だけ "L" 側のカウンタ数 C, "H" 側のカウンタ数 B のパルスが発生します。

同様に、16 ビット PPG モードで使用する場合には、ch0 と ch1 の PRL はロングワード転送で書き込むかまたは ch0 → ch1 の順にワード転送で書き込んでください。このモードのときには、ch0 の PRL への書込みは、一時的に書き込まれ、その後 ch1 の PRL に書込みを行った時点で、実際に ch0 の PRL への書込みが行われます。

なお、図 14.8-2 に示すように、16 ビット PPG モード以外では、ch0 と ch1 の PRL の書込みは独立して行うことができます。

図 14.8-2 PRL 書き込み部分のブロックダイヤグラム



第15章

遅延割込み

遅延割込みの機能と動作について説明します。

15.1 遅延割込みモジュールの概要

15.2 遅延割込みのレジスタ

15.3 遅延割込みの動作

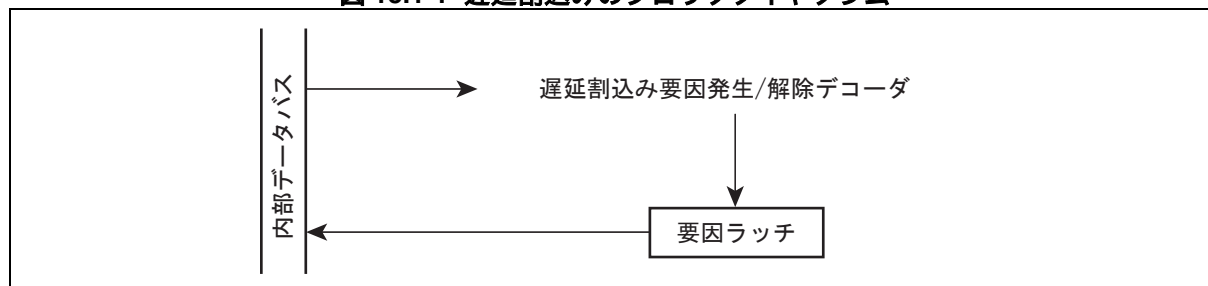
15.1 遅延割込みモジュールの概要

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

■ 遅延割込みのブロックダイアグラム

図 15.1-1 に、遅延割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 15.1-1 遅延割込みのブロックダイアグラム



■ 遅延割込み要求ロックの使用上の注意

このロックは DIRR の該当するビットに "1" を書き込むことでセットされ、同じビットに "0" を書き込むことでクリアされます。したがって、割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込み処理から復帰した途端に再割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。

15.2 遅延割込みのレジスタ

DIRR は遅延割込み要求の発生 / 解除を制御するレジスタで、このレジスタに対して "1" を書き込むときには遅延割込みの要求を発生させ、"0" を書き込むときには遅延割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。

■ 遅延割込み要因発生 / 解除レジスタ (DIRR)

遅延割込み要因発生 / 解除レジスタ (DIRR) では、予約ビット領域は "0", "1" どちらの書き込みも結構ですが、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します。

遅延割込み要因発生 / 解除レジスタ (DIRR) のビット構成を図 15.2-1 に示します。

図 15.2-1 遅延割込み要因発生 / 解除レジスタ (DIRR)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 00009F _H								R0	DIRR
リード / ライト	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(R/W)	...
初期値	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(0)	...

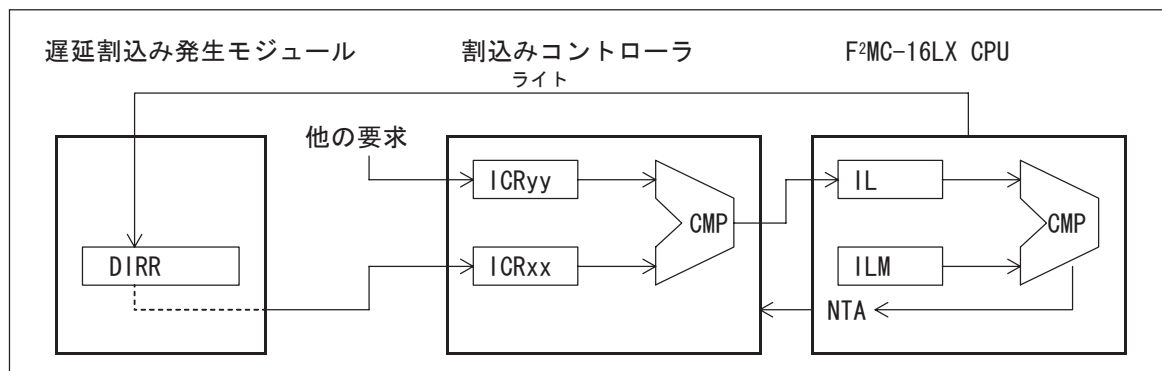
15.3 遅延割込みの動作

ソフトウェアにて CPU が DIRR の該当ビットに "1" を書き込むと、遅延割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、割込みコントローラに割込み要求が発生します。

■ 遅延割込み発生

ほかの割込み要求が本割込みより優先順位が低いあるいは、ほかの割込み要求がない場合に、割込みコントローラは F²MC-16LX CPU に対して割込み要求が発生します。図 15.3-1 に示すように、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。この結果、本割込みに対する割込み処理ルーチンが実行されます。

図 15.3-1 遅延割込み発生



割込み処理ルーチン内で DIRR の該当ビットの "0" を書き込むことで本割込み要因をクリアし、合わせてタスクの切換えを行います。

第16章

DTP/ 外部割込み

DTP/ 外部割込みの機能と動作について説明します。

16.1 DTP/ 外部割込みの概要

16.2 DTP/ 外部割込みのレジスタ

16.3 DTP/ 外部割込みの動作

16.4 DTP 要求と外部割込み要求の切換え

16.5 DTP/ 外部割込みの使用上の注意

16.1 DTP/ 外部割込みの概要

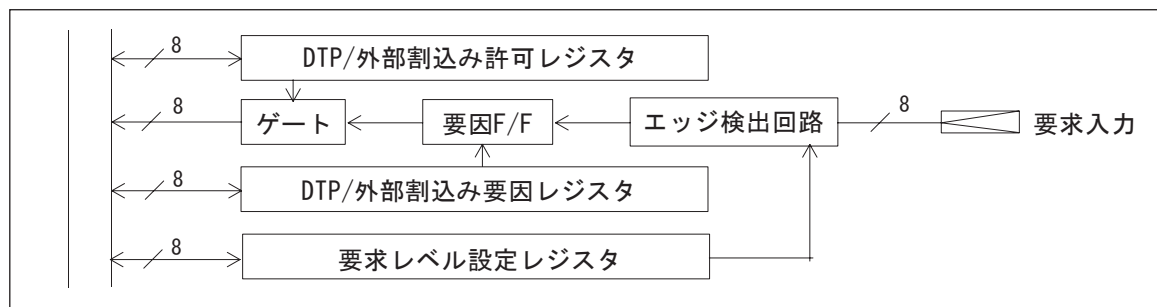
DTP(Data Transfer Peripheral) はデバイス外部に存在するペリフェラルと F²MC-16LX CPU との間において、外部ペリフェラルが発生する DMA 要求あるいは割込み要求を受け取り、これを F²MC-16LX CPU に伝えてインテリジェント I/O サービスあるいは割込み処理を起動させるための周辺です。

■ DTP/ 外部割込みの概要

インテリジェント I/O 機能に関しては "H" または "L" 要求レベルを使用することができます。外部割込み要求に対しては4種類の要求レベルを使用することができます("H", "L" の立上りエッジおよび立下りエッジ)。

■ DTP/ 外部割込みのブロックダイアグラム

図 16.1-1 DTP/ 外部割込みのブロックダイアグラム



■ DTP/ 外部割込みのレジスタ

DTP/ 外部割込みのレジスタのビット構成を図 16.1-2 に示します。

図 16.1-2 DTP/ 外部割込みのレジスタ一覧

DTP/ 外部割込み許可レジスタ									
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000030 _H	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	ENIR
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)	
DTP/ 外部割込み要因レジスタ									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000031 _H	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	EIRR
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
要求レベル設定レジスタ									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000033 _H	LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	
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)	
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000032 _H	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	ELVR
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)	

16.2 DTP/ 外部割込みのレジスタ

DTP/ 外部割込みのレジスタには、次の 3 つがあります。

- DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)
- DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)
- 要求レベル設定レジスタ (ELVR:External level register)

■ DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)

DTP/ 外部割込み許可レジスタ (ENIR) のビット構成を図 16.2-1 に示します。

図 16.2-1 DTP/ 外部割込み許可レジスタ (ENIR)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000030 _H	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	ENIR
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)	

DTP/ 外部割込み許可レジスタ (ENIR) は、デバイス端子を DTP/ 外部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。このレジスタの "1" が書かれたビットと対応する端子は DTP/ 外部割込み要求入力として使用されます。"0" が書かれたビットと対応する端子は 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/ 外部割込み要因レジスタ (EIRR:External interrupt request register)

DTP/ 外部割込み要因レジスタ (EIRR) のビット構成を図 16.2-2 に示します。

図 16.2-2 DTP/ 外部割込み要因レジスタ (EIRR)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000031 _H	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	EIRR
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

EIRR は、このレジスタの "1" となっているビットに対応する端子において DTP/ 外部割込み要求が存在することを表します。このレジスタのビットに "0" を書き込むと対応する要求フラグがクリアされます。"1" を書き込んでも効果はありません。リードモディファイライト系命令によりこのレジスタからは常に "1" が読み出されます。

< 注意事項 >

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 が割込みを受け付けたビット (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。

■ 要求レベル設定レジスタ (ELVR:External level register)

要求レベル設定レジスタ (ELVR) のビット構成を図 16.2-3 に示します。

図 16.2-3 要求レベル設定レジスタ (ELVR)

アドレス : 000033 _H	15	14	13	12	11	10	9	8	ビット No.
	LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
リード/ライト 初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
アドレス : 000032 _H	7	6	5	4	3	2	1	0	ビット No. ELVR
	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	
	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
リード/ライト 初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

ELVR は外部端子における要求イベントを定義するものです。表 16.2-1 に示すように、各端子には 2 ビットが割り当てられます。入力レベルにより要求が検出された場合、フラグがソフトウェアによりリセットされた後でも入力が指定されたレベルを保持していれば、割込みフラグはセットされます。

表 16.2-1 LBx と LAx 端子の割込み要求検出要因

LBx	LAx	割込み要求検出要因
0	0	"L" レベルが端子に入力
0	1	"H" レベルが端子に入力
1	0	立上りエッジが端子に入力
1	1	立下りエッジが端子に入力

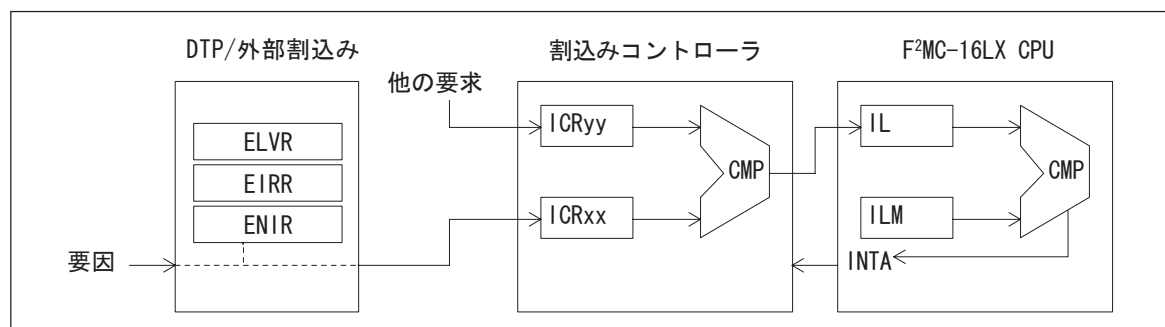
16.3 DTP/ 外部割込みの動作

割込みフラグがセットされたときにこのブロックは割込みコントローラへ割込み信号を送ります。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 は自分の内部にある CCR レジスタ中の ILM ビット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。

■ 外部割込みの動作

図 16.3-1 に示すように、ハードウェア割込み処理マイクロプログラムにおいて、CPU は割込みコントローラから ISE ビットの情報を読み出し、これにより当該要求が割込み処理であることを識別し、割込み処理マイクロプログラムへ分岐します。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。

図 16.3-1 外部割込みの動作



■ DTP の動作

初期化としてユーザプログラム内ではインテリジェント I/O サービスを起動するに当たって、インテリジェント I/O サービスディスクリプタ内の I/O レジスタアドレスポインタに "000000_H" から "0000FF_H" に割りつ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。DTP の場合は CPU がハードウェア割込み処理マイクロプログラム内で読み出す ISE ビットの内容が DTP を示しているので、インテリジェント I/O サービス処理マイクロプログラムへ制御を移します。インテリジェント I/O サービスが起動されると、アドレッシングされている外部ペリフェラルにリードまたはライト信号が送られ本チップとの転送が行われます。外部ペリフェラル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。インテリジェント I/O 機能処理の詳細については、『MB90500 プログラミングマニュアル』を参照してください。

図 16.3-2 DTP 動作終了時の外部割込み取り下げタイミング

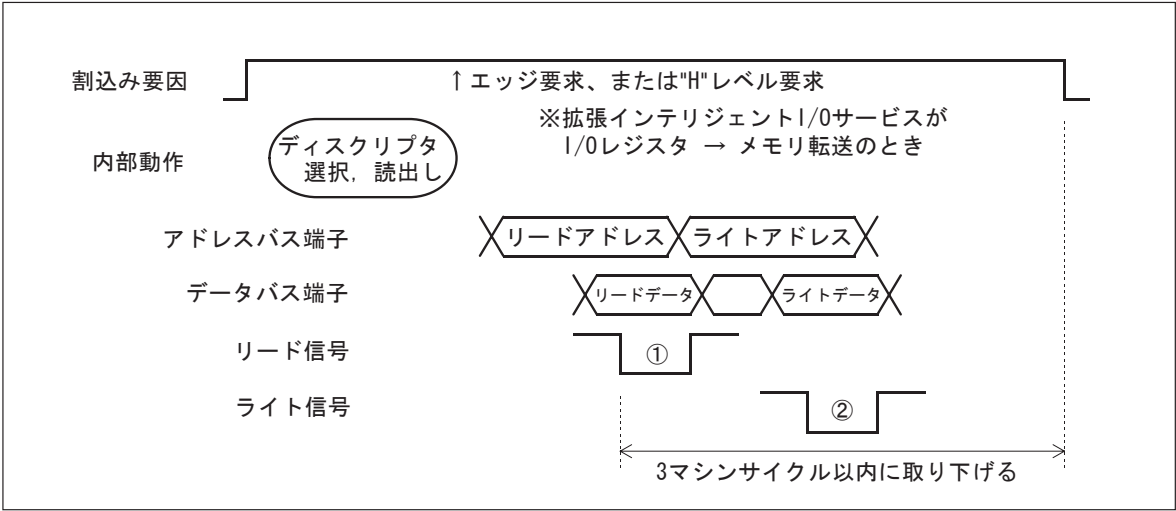
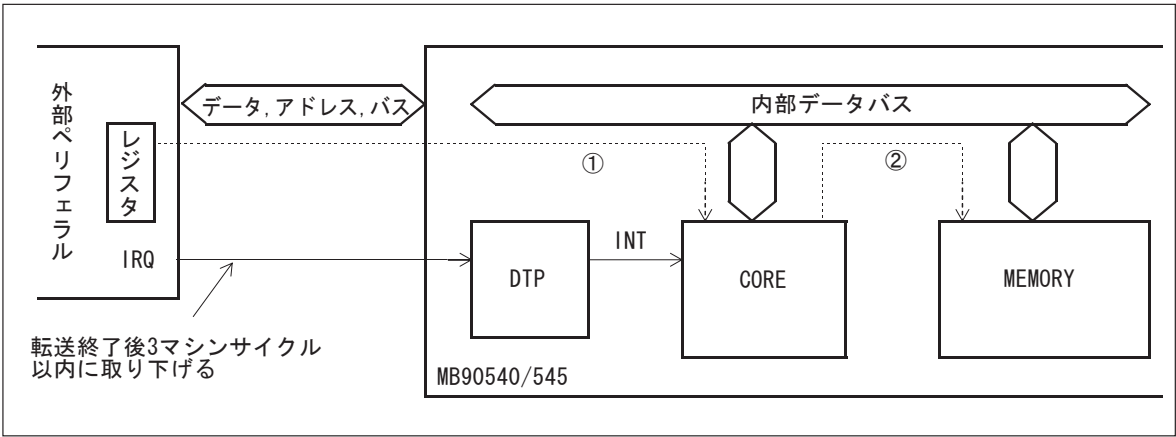


図 16.3-3 DTP と外部ペリフェラルとのインタフェース簡略例

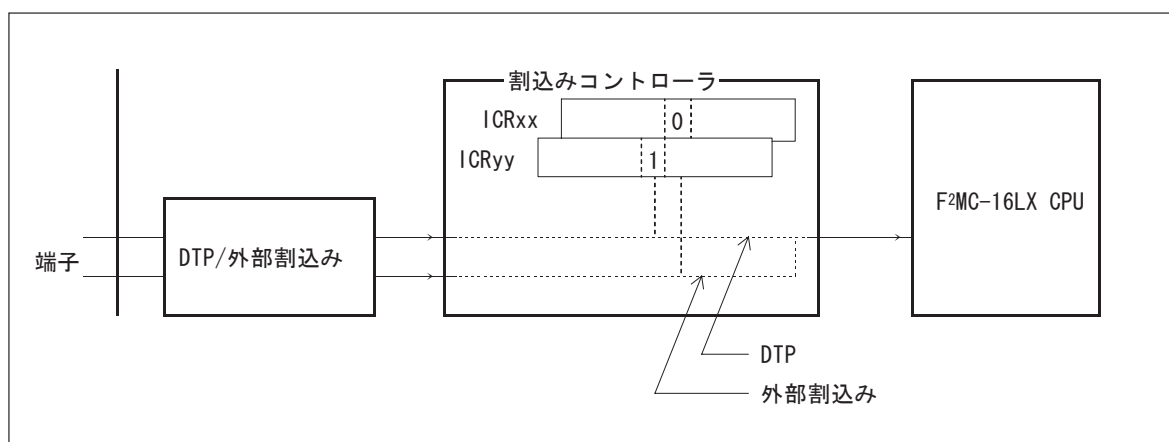


16.4 DTP 要求と外部割込み要求の切換え

外部割込み要求と DTP 要求の切換えは、割込みコントローラの中にある、本リソースに対応する ICR レジスタ中の ISE ビットの設定によって行います。各端子に対応して個々に ICR が割り当てられていますので、対応する ICR の ISE ビットに "1" が書かれた端子が DTP 要求として、また、"0" が書かれた場合には外部割込み要求として、それぞれ動作することになります。

■ DTP 要求と外部割込み要求の切換え

図 16.4-1 DTP 要求と外部割込み要求の切換え



16.5 DTP/ 外部割込みの使用上の注意

DTP/ 外部割込みを使用するには、次の点について特に注意が必要です。

- DTP を用いたときの外部に接続するペリフェラルの条件
- DTP/ 外部割込みの動作手順
- 外部割込み要求レベル

■ DTP/ 外部割込みの使用上の注意

● DTP を用いたときの外部に接続するペリフェラルの条件

DTP がサポートできる外部ペリフェラルは、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、転送動作が開始した後 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を行う際、次の手順で設定してください。

- 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。
（ただし、4）と 5）はワード指定による同時書込み可）。

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起こっ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とき、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とします。

図 16.5-1 に示すように、要求入力レベルがレベル設定のとき、外部より要求入力が入って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動作中のままです。図 16.5-2 に示すように、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 16.5-1 レベル設定時の割込み要求フラグビット (EIRR:ER) のクリア

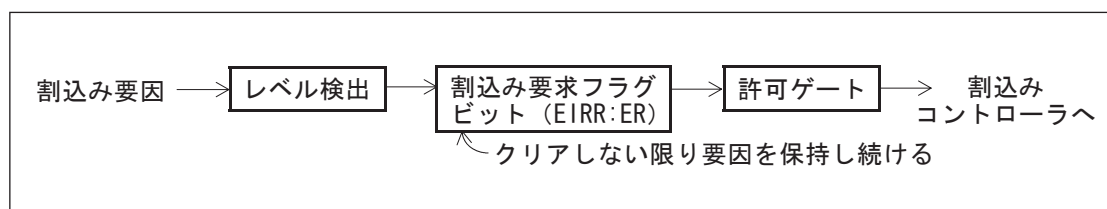
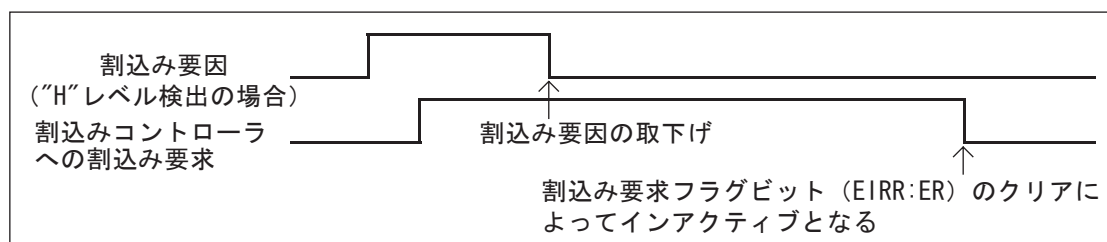


図 16.5-2 割込み許可時の割込み要因と割込みコントローラへの割込み要求



第17章

A/D コンバータ

A/D コンバータの機能と動作について説明します。

- 17.1 A/D コンバータの特長
- 17.2 A/D コンバータのブロックダイアグラム
- 17.3 A/D コンバータのレジスタ一覧
- 17.4 A/D コンバータの動作
- 17.5 I²C を使った変換動作
- 17.6 変換データ保護機能

17.1 A/D コンバータの特長

A/D コンバータは、アナログ入力電圧をデジタル値に変換するものです。A/D コンバータの特長は次のとおりです。

■ A/D コンバータの特長

● 変換時間

1 チャンネル当たり最小 26.3 μ s(16MHz のマシクロック)

● サンプル & ホールド回路付き RC 型逐次比較変換方式を採用

● 10 ビットまたは 8 ビットの分解能

● アナログ入力は 8 チャンネルからプログラムで選択

- ・ シングル変換モード : 1 チャンネルを選択変換
- ・ スキャン変換モード : 連続した複数のチャンネルを変換。最大 8 チャンネルプログラム可能
- ・ 連続変換モード : 指定チャンネルを繰り返し変換
- ・ 停止変換モード : 指定チャンネルを変換したら一時停止して次の起動が掛かるまで待機 (変換開始の同期が可能)

● 割込み要求

A/D 変換終了時には、CPU に対して A/D 変換終了の割込み要求を発生させることができます。この割込み発生で EI²OS を起動することができ A/D 変換結果データをメモリに自動的に転送できますので連続処理に適しています。

● 起動要因を選択可能

起動要因は、ソフト、外部トリガ (立下りエッジ) またはタイマ (立上りエッジ) があります。

■ アナログ入力許可レジスタ

アナログ入力に使用される端子に対応する ADER ビットに常に "1" を書き込んでください。

アナログ入力許可レジスタ (ADER) のビット構成を図 17.1-1 に示します。

図 17.1-1 アナログ入力許可レジスタ

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00001B _H	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0	ADER
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	(1)	(1)	(1)	(1)	(1)	(1)	(1)	(1)	

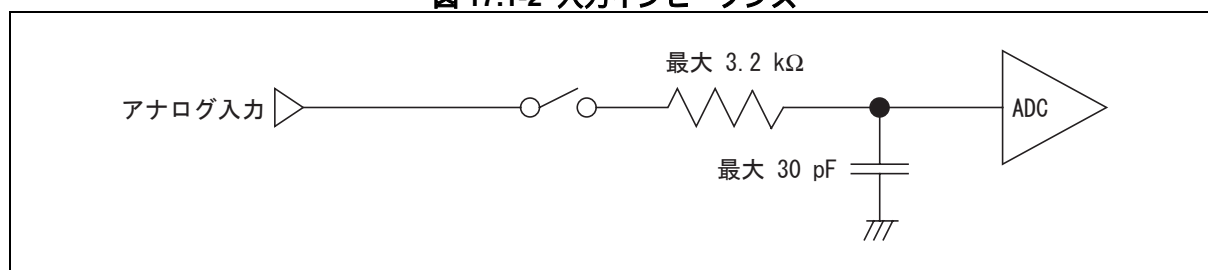
ポート 6 端子は次のように制御されます。

- ・ 0: ポート入力 / 出力モード
- ・ 1: アナログ入力モード
- ・ リセット時には "1" がセットされます。

■ 入力インピーダンス

A/D コンバータのサンプリング回路は次の等価回路で表されます。

図 17.1-2 入力インピーダンス



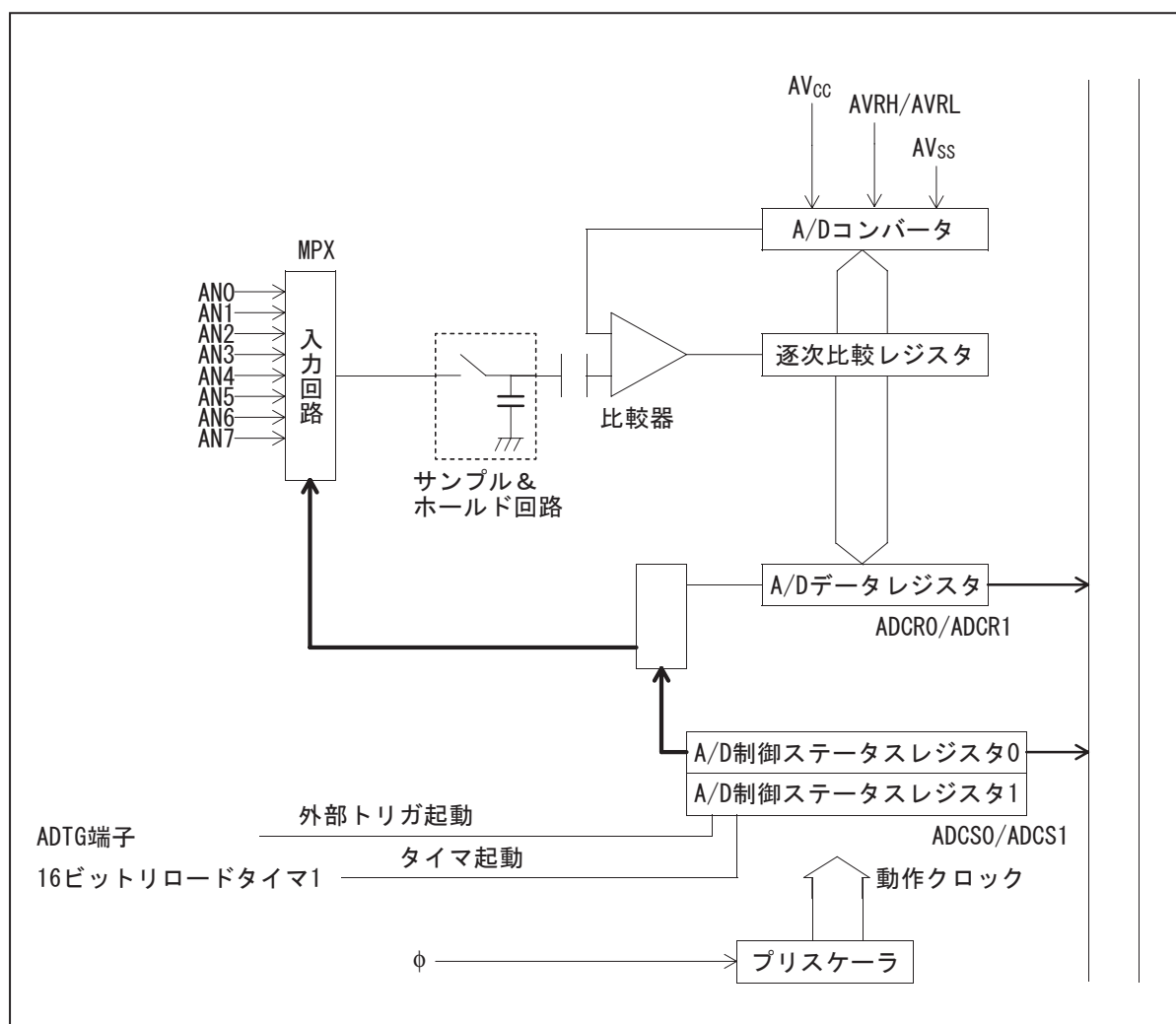
サンプリング時間が $4\ \mu\text{s}$ (16 MHz のマシクロックで $ST = 0$ および $ST0 = 0$) にセットされたとき、アナログ入力への駆動インピーダンスは $15.5\ \text{k}\Omega$ 以下でなければなりません。そうしなければ変換の精度は低下します。この場合、サンプリング時間をさらに長く設定する ($ST1 = 1$ および $ST0 = 1$) か、外部コンデンサを付加して駆動インピーダンスを補償してください。

17.2 A/D コンバータのブロックダイアグラム

図 17.2-1 に , A/D コンバータのブロックダイアグラムを示します

■ A/D コンバータのブロックダイアグラム

図 17.2-1 A/D コンバータのブロックダイアグラム



17.3 A/D コンバータのレジスタ一覧

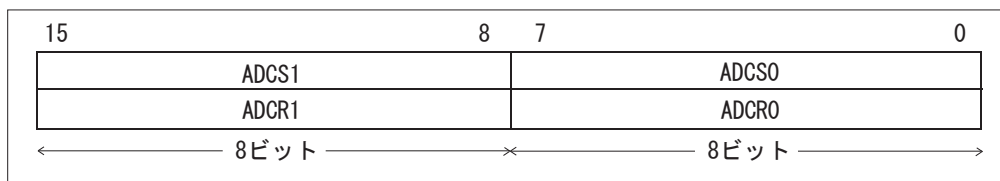
A/D コンバータのレジスタには次の 2 つがあります。

- A/D 制御ステータスレジスタ 0/1
- A/D データレジスタ 0/1

■ A/D コンバータのレジスタ一覧

A/D コンバータには次のレジスタがあります。

図 17.3-1 A/D コンバータのレジスタ配置



A/D コンバータのレジスタのビット構成を図 17.3-2 に示します。

図 17.3-2 A/D コンバータのレジスタ

A/D 制御ステータスレジスタ 0									
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000034 _H	MD1	MD0	ANS2	ANS1	ANS0	ANE2	ANE1	ANE0	ADCS0
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)	
A/D 制御ステータスレジスタ 1									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000035 _H	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	予約	ADCS1
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
A/D データレジスタ 0									
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000036 _H	D7	D6	D5	D4	D3	D2	D1	D0	ADCR0
リード/ライト	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
A/D データレジスタ 1									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000037 _H	SI0	ST1	ST0	CT1	CT0		D9	D8	ADCR1
リード/ライト	(W)	(W)	(W)	(W)	(W)	(-)	(R)	(R)	
初期値	(0)	(0)	(0)	(0)	(1)	(-)	(X)	(X)	

17.3.1 A/D 制御ステータスレジスタ 0 (ADCS0)

A/D 制御ステータスレジスタ 0 (ADCS0) は、A/D コンバータの制御およびステータス表示を行います。ADCS0 は A/D 変換動作中に書き換えないようにしてください。

■ A/D 制御ステータスレジスタ 0 (ADCS0)

A/D 制御ステータスレジスタ 0 (ADCS0) のビット構成を図 17.3-3 に示します。

図 17.3-3 A/D 制御ステータスレジスタ 0 (ADCS0)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000034 _H	MD1	MD0	ANS2	ANS1	ANS0	ANE2	ANE1	ANE0	ADCS0
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)	

【bit 7, bit 6】MD1, MD0 (A/D converter mode set)

MD1, MD0 ビットは、動作モードを設定します。

MD1	MD0	動作モード
0	0	単発モード、動作中の再起動はすべて可能
0	1	単発モード、動作中の再起動不可能
1	0	連続モード、動作中の再起動不可能
1	1	停止モード、動作中の再起動不可能

● 単発モード

ANS2 ~ ANS0 の設定チャンネルから ANE2 ~ ANE0 の設定チャンネルまで A/D 変換を連続して行いすべてのチャンネルの変換が終了したら停止する。

● 連続モード

ANS2 ~ ANS0 の設定チャンネルから ANE2 ~ ANE0 の設定チャンネルまで A/D 変換を繰り返し行います。

● 停止モード

ANS2 ~ ANS0 の設定チャンネルから ANE2 ~ ANE0 の設定チャンネルまで 1 チャンネルごとに A/D 変換を行い一時停止する。変換再開は起動要因発生によって行われます。

リセット時には本ビット群は "00_B" に初期化されます。

< 注意事項 >

- ・ 連続モード、停止モードで A/D 変換を起動すると、BUSY ビットで停止するまで変換動作を続けます。
- ・ 停止は BUSY ビットに "0" を書き込むことにより行われます。
- ・ 単発、連続、停止の各モードの再起動の不可能はタイマ、外部トリガ、ソフトすべての起動に適用されます。

【bit 5 ~ bit 3】ANS2, ANS1, ANS0 (Analog start channel set)

本ビット群より A/D 変換の開始チャンネルを設定します。

A/D コンバータを起動すると、このビットで選択されたチャンネルから A/D 変換を開始します。

ANS2	ANS1	ANS0	開始チャンネル
0	0	0	AN0
0	0	1	AN1
0	1	0	AN2
0	1	1	AN3
1	0	0	AN4
1	0	1	AN5
1	1	0	AN6
1	1	1	AN7

< 注意事項 >

- リード
A/D 変換の間、現在の変換チャンネルが本ビット群から読み込まれます。もしシステムが停止モードで停止すれば最後の変換チャンネルが読み込まれます。また、本ビットに値を設定した場合でも、A/D 変換が開始されるまでは、設定した値ではなく、前回に A/D 変換したチャンネル番号が読み出されます。
- リセット時に本ビット群は "000_B" に初期化されます。

【bit 2 ~ bit0】ANE2, ANE1, ANE0 (Analog end channel set)

本ビット群より A/D 変換の終了チャンネルを設定します。

ANE2	ANE1	ANE0	終了チャンネル
0	0	0	AN0
0	0	1	AN1
0	1	0	AN2
0	1	1	AN3
1	0	0	AN4
1	0	1	AN5
1	1	0	AN6
1	1	1	AN7

< 注意事項 >

- 同一のチャンネルが ANE2 ~ ANE0 および ANS2 ~ ANS0 に書き込まれると、変換は 1 つのチャンネルに対してのみ行われます (シングル変換)。
- 連続モード, 停止モードを設定している場合は本ビット群で設定されたチャンネルの変換が終わると ANS2 ~ ANS0 で設定された開始チャンネルに戻ります。
- 設定チャンネルが ANS > ANE の場合は, ANS より変換が始まり AN7 まで変換したら AN0 に戻り ANE まで変換します。

〔例：チャンネル設定 ANS = AN6, ANE = AN3 で単発モードのとき変換は次の順序で行われます。〕

AN6	AN7	AN0	AN1	AN2	AN3
-----	-----	-----	-----	-----	-----

- リセット時に本ビット群は "000_B" に初期化されます。
-

17.3.2 A/D 制御ステータスレジスタ 1 (ADCS1)

A/D 制御ステータスレジスタ 1 (ADCS1) は、A/D コンバータの制御およびステータス表示を行います。

■ A/D 制御ステータスレジスタ 1 (ADCS1)

A/D 制御ステータスレジスタ 1 (ADCS1) のビット構成を図 17.3-4 に示します。

図 17.3-4 A/D 制御ステータスレジスタ 1 (ADCS1)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000035 _H	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	予約	ADCS1
リード/ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

【bit 15】BUSY (busy flag and stop)

- リード時
 - A/D コンバータ動作表示用のビットです。
 - A/D 変換起動でセットされ終了でクリアされます。
- ライト時
 - A/D 動作中に本ビットに "0" を書き込むと強制的に動作を停止します。
 - 連続、停止モード時の強制停止に利用します。

動作表示用のビットに "1" を書き込むことはできません。リードモディファイライト系命令では "1" が読めます。単発モードでは A/D 変換終了でクリアされます。

連発、停止モードでは "0" 書込みで停止するまでクリアされません。

リセット時に本ビットは "0" に初期化されます。

強制停止とソフト起動を同時に行わないでください (BUSY=0, STRT=1)。

【bit 14】INT (interrupt)

変換データが ADCR に書き込まれるとセットされます。

bit5(INTE) が "1" のときに本ビットがセットされると割込み要求が発生します。また、EI²OS 起動を許可している場合 EI²OS が起動されます。"1" 書込みは意味を持ちません。

クリアは "0" 書込みと EI²OS 割込みクリア信号で行われます。

< 注意事項 >

本ビットの "0" 書込みクリアは A/D 停止中に行ってください。

リセット時に本ビットは "0" に初期化されます。

【bit 13】INTE (Interrupt enable)

本ビットは変換終了による割込みの許可・禁止を指定します。

- 0: 割込みの禁止
- 1: 割込みの許可

EI²OS を使用するときは本ビットをセットしてください。EI²OS は割込み要求発生で起動するようになっています

リセット時に本ビットは "0" に初期化されます。

【bit 12】PAUS (A/D converter pause)

A/D 変換動作が一時的に停止した場合にセットされます。

A/D 変換結果を格納するレジスタが 1 つのため連続で変換した場合、変換結果を EI²OS で転送しなければ前データが壊れてしまいます。

これを保護するためデータレジスタの内容を EI²OS で転送しなければ次の変換データが格納されないようになっています。この間 A/D 変換動作は停止します。EI²OS で転送を終了すると A/D は変換を再開します。

このビットは EI²OS を使用したときのみ有効です。

< 注意事項 >

- 変換データの保護機能に関しては「17.4 A/D コンバータの動作」を参照してください。
- リセット時に本ビットは "0" に初期化されます。

【bit 11, bit 10】STS1, STS0 (Start source select)

リセット時に本ビット群は "00_B" に初期化されます。

本ビットの設定により A/D 起動要因を選択します。

STS1	STS0	機能
0	0	ソフト起動
0	1	外部端子トリガでの起動とソフト起動
1	0	タイマでの起動とソフト起動
1	1	外部端子トリガ、タイマでの起動とソフト起動

2 つ以上の起動要因が許可されているモードでは最初に生じた原因により A/D 変換が起動されます。

起動源の設定は書き換えられると直ちに変わりますので、A/D 変換中に書き換えるときには注意してください。

< 注意事項 >

- 外部端子トリガは立下りエッジを検出します。外部トリガ入力レベルが "L" のときに本ビットを書き換えて外部端子トリガ起動に設定すると A/D が起動する場合があります。
- タイマ選択時は、16 ビットリロードタイマ 1 が選択されます。

【bit 9】STRT (Start)

本ビットに "1" を書き込むことにより A/D を起動します。

再起動をかけるときは、再び "1" を書き込んでください。

リセット時に本ビットは "0" に初期化されます。

停止モード時は、動作機能上再起動はかかりません。"1" を書き込む前に BUSY ビットをチェックしてください。

強制停止とソフト起動を同時に行わないでください (BUSY=0, STRT=1)。

バイト / ワード命令では "1" が読み出されます。リードモディファイライト系命令では "0" が読み出されます。

【bit 8】予約ビット

本ビットには常に "0" を書き込んでください。

17.3.3 A/D データレジスタ 0/1 (ADCR0, ADCR1)

A/D データレジスタ 0/1 (ADCR0, ADCR1) は、変換の結果として生成されたデジタル値を格納するために使用されるものです。ADCR0 は下位 8 ビットを格納し、ADCR1 は変換結果の最上位の 2 ビットを格納します。これらのレジスタ値は変換が完了するたびに書き換えられます。通常は最後の変換値が本ビット群に格納されます。

■ A/D データレジスタ 0/1(ADCR0, ADCR1)

A/D データレジスタ 0/1 (ADCR0, ADCR1) のビット構成を図 17.3-5 に示します。

図 17.3-5 A/D データレジスタ 0/1 (ADCR0, ADCR1)

A/D データレジスタ 0									ビット No. ADCR0
	7	6	5	4	3	2	1	0	
アドレス : 000036 _H	D7	D6	D5	D4	D3	D2	D1	D0	
リード/ライト	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
初期値	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
A/D データレジスタ 1									ビット No. ADCR1
	15	14	13	12	11	10	9	8	
アドレス : 000037 _H	SI0	ST1	ST0	CT1	CT0		D9	D8	
リード/ライト	(W)	(W)	(W)	(W)	(W)	(-)	(R)	(R)	
初期値	(0)	(0)	(0)	(0)	(1)	(-)	(X)	(X)	

ADCR1 の bit10 ~ bit15 からは常に "0" が読み込まれます。

変換データの保護機能を利用することができます。「17.4 A/D コンバータの動作」を参照してください。

A/D 変換の間はこれらのレジスタにデータを絶対に書き込まないでください。

【bit 15】SI0

本ビットは変換の分解能を指定します。"0" にセットされると 10 ビットの A/D 変換が行われます。そのほかの場合には 8 ビットの A/D 変換が行われてその結果は D7 ~ D0 に格納されます。

本ビットを読み出すと常に "0" が戻されます。

【bit 14, bit 13】ST1, ST0 (Sampling time)

ST1	ST0	機能
0	0	64 マシンサイクル (16 MHz で 4 μs)
0	1	予約
1	0	予約
1	1	4096 マシンサイクル (16 MHz で 256 μs)

本ビット群は入力時における電圧サンプリング時間の持続時間を決定するものです。

本ビット群を読み出すと常に "00_B" が戻されます。

【bit12, bit 11】CT1, CT0 (Compare time)

CT1	CT0	機能
0	0	176 マシンサイクル (8 MHz で 22 μ s)
0	1	352 マシンサイクル (16 MHz で 22 μ s)
1	0	予約
1	1	予約

本ビット群は比較操作時間の持続期間を決定するものです。

本ビットを "00_B" に設定するときはマシクロックが 8 MHz 以下のときにしてください。8 MHz を超える場合には変換精度が保証されません。

本ビット群を読み出すと常に "00_B" が戻されます。

17.4 A/D コンバータの動作

A/D コンバータは逐次比較方式で動作し、分解能は 10 ビットまたは 8 ビットを選択することができます。

この A/D コンバータは変換結果格納用のレジスタが 1 つ (16 ビット) であるため、A/D データレジスタ 0/1 (ADCR0 および ADCR1) は変換が完了するたびに書き換えられます。

このため、A/D コンバータ単独では連続変換処理には適しませんので F²MC-16 インテリジェント I/O 機能 (EI²OS) を使って変換データをメモリに転送しながら変換することを推奨いたします。動作モードを次に説明します。

■ 単発モード

単発モードは、ANS ビットと ANE ビットで設定されたアナログ入力を順に変換していき 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 は動作を停止します。開始チャンネルと終了チャンネルが同じとき (ANS=ANE) は一方のチャンネルのみの変換動作になります。

【例】

ANS = 000_B, ANE = 011_B

開始 AN0 AN1 AN2 AN3 終了

ANS = 010_B, ANE = 010_B

開始 AN2 終了

■ 連続モード

連続モードは、ANS ビットと ANE ビットで設定されたアナログ入力を順に変換していき 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 のアナログ入力に戻り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とき (ANS = ANE) は一方のチャンネルのみの変換を続けます。

【例】

ANS = 000_B, ANE = 011_B

開始 AN0 AN1 AN2 AN3 AN0 -- 繰返し

ANS = 010_B, ANE = 010_B

開始 AN2 AN2 AN2 -- 繰返し

連続モードで変換させると BUSY ビットに "0" を書き込むまで変換を繰り返し続けます (BUSY ビットに "0" を書き込む 強制動作停止)。強制動作停止を行うと変換中のものは途中で止まってしまうため、注意してください (強制動作停止した場合は、変換レジスタは変換完了した前データが格納されています)。

■ 停止モード

停止モードは、ANS ビットと ANE ビットで設定されたアナログ入力を順に変換していきますが 1 チャンネル変換するたびに変換動作を一時停止します。一時停止を解除するにはもう一度起動をかけることにより行われます。

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 のアナログ入力に戻り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 (ANS = ANE) であれば変換は一方のチャンネルに対してのみ実行されます。

【例】

ANS = 000_B, ANE = 011_B

開始 AN0 停止 起動 AN1 停止 起動 AN2 停止 起動
 AN3 停止 起動 AN0 -- 繰返し

ANS = 010_B, ANE = 010_B

開始 AN2 停止 起動 AN2 停止 起動 AN2 -- 繰返し

このときの起動要因は STS1, STS0 で設定されたものだけです。

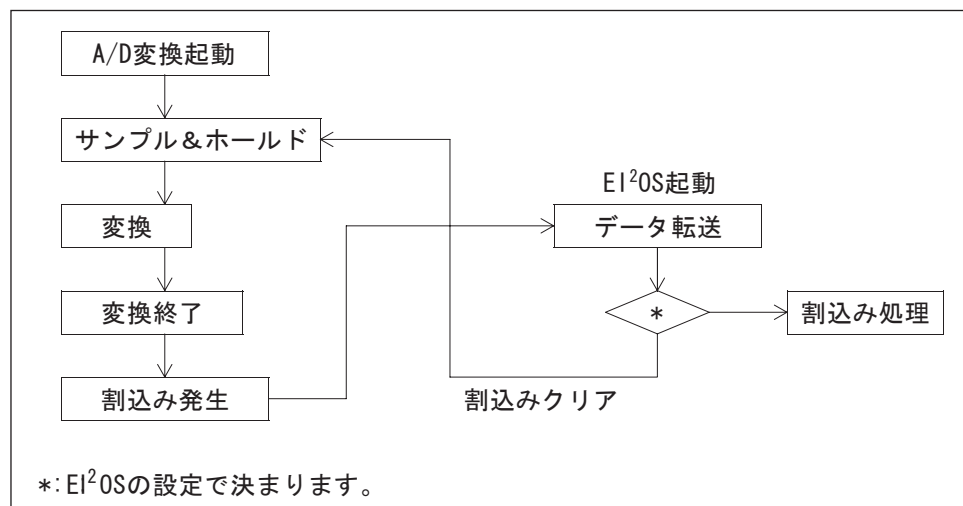
このモードを使用することにより変換開始の同期を掛けることが可能です。

17.5 EI²OS を使った変換動作

図 17.5-1 に , A/D 変換起動から変換データ転送までのフロー例 (連続モード) を示します。

■ EI²OS を使った変換動作

図 17.5-1 A/D 変換起動から変換データ転送までのフロー例 (連続モード)



17.5.1 単発モード時の EI²OS の起動例

単発モード時の EI²OS は次の手順で起動します。

- アナログ入力 AN1 ~ AN3 の変換をして終了する。
- 変換データは "200_H" ~ "205_H" 番地に順に転送する。
- ソフトウェアで変換を開始する。
- 最高の割込みレベルを使用する。

■ 単発モード時の EI²OS の起動例

表 17.5-1 単発モード時の EI²OS の起動例

設定項目	プログラム例	動作説明
EI ² OS の設定	MOV ICR3, #08H	最強割込みレベル, 割込み時 EI ² OS 起動, ディスクリプタアドレスの設定
	MOV BAPL, #00H	変換データの転送先番地
	MOV BAPM, #02H	
	MOV BAPH, #00H	
	MOV ISCS, #18H	ワードデータ転送, 転送後転送先番地をインクリメントする。I/O からメモリへ転送。転送はリソースからの要求に応じて終了しない。
	MOV I/OA, #36H	転送元アドレス
	MOV DCT, #03H	EI ² OS 転送を 3 回行う。変換回数と同じにする。
A/D コンバータ設定	MOV ADCS0 #0BH	単発モード, 開始チャネル AN1, 終了チャネル AN3
	MOV ADCS1 #A2H	ソフト起動, A/D 変換開始
割込みシーケンス	RET	割込みからの復帰

ICR3 : 割込み制御レジスタ

BAPL : バッファアドレスポインタ下位

BAPM : バッファアドレスポインタ中位

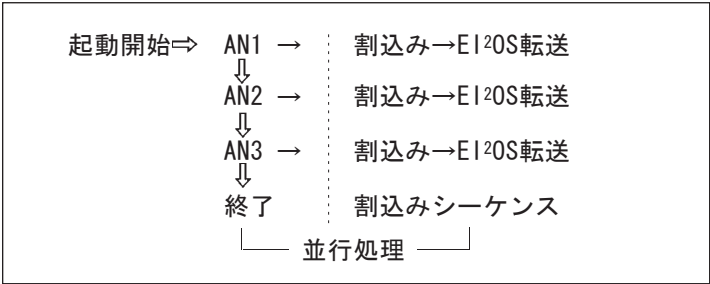
BAPH : バッファアドレスポインタ上位

ISCS : EI²OS ステータスレジスタ

I/OA : I/O アドレスカウンタ

DCT : データカウンタ

図 17.5-2 単発モード時の EI²OS の起動例



17.5.2 連続モード時の EI²OS の起動例

連続モード時の EI²OS の起動例は次のとおりです。

- アナログ入力 AN3 ~ AN5 の変換をして各チャンネルの変換データを 2 つ取得する。
- 変換データは "600_H" ~ "60B_H" 番地に順に転送する。
- 起動は外部エッジ入力で行う。
- 割込みレベル最強

■ 連続モード時の EI²OS の起動例

表 17.5-2 連続モード時の EI²OS の起動例

設定項目	プログラム例	動作説明
EI ² OS の設定	MOV ICR3 #08H	最強割込みレベル、割込み時 EI ² OS 起動、ディスクリプタアドレスの設定
	MOV BAPL, #00H	変換データの転送先番地
	MOV BAPM, #06H	
	MOV BAPH, #00H	
	MOV ISCS, #18H	ワードデータ転送、転送後転送先番地をインクリメントする。I/O からメモリへ転送。リソースからの要求で終了しない。
	MOV I/OA, #36H	転送元アドレス
	MOV DCT, #06H	EI ² OS 転送を 6 回行う。3 チャンネル×2 分のデータ転送を行う。
A/D コンバータ設定	MOV ADCS0 #9DH	連続モード、開始チャンネル AN3、終了チャンネル AN5
	MOV ADCS1 #A4H	外部エッジ起動、A/D 変換開始
割込みシーケンス	MOV ADCS1 #00H	割込みからの復帰
	RET	

ICR3 : 割込み制御レジスタ

BAPL : バッファアドレスポインタ下位

BAPM : バッファアドレスポインタ中位

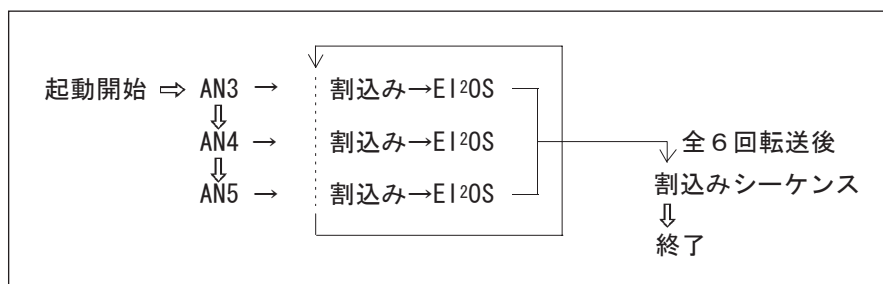
BAPH : バッファアドレスポインタ上位

ISCS : EI²OS ステータスレジスタ

I/OA : I/O アドレスカウンタ

DCT : データカウンタ

図 17.5-3 連続モード時の EI²OS の起動例



17.5.3 停止モード時の EI²OS の起動例

停止モード時の EI²OS の起動例は、次のとおりです。

- アナログ入力 AN3 を一定期間おいて 12 回変換する。
- 変換データは 600_H ~ 617_H 番地に順に転送する。
- 起動は外部エッジ入力で行う。
- 割込みレベル最強

■ 停止モード時の EI²OS の起動例

表 17.5-3 停止モード時の EI²OS の起動例

設定項目	プログラム例	動作説明
EI ² OS の設定	MOV ICR3 #08H	最強割込みレベル、割込み時 EI ² OS 起動、ディスクリプタアドレスの設定
	MOV BAPL, #00H	変換データの転送先番地
	MOV BAPM, #06H	
	MOV BAPH, #00H	
	MOV ISCS, #18H	ワードデータ転送、転送後転送先番地をインクリメントする。I/O からメモリへ転送。リソースからの要求で終了しない。
	MOV I/OA, #36H	転送元アドレス
	MOV DCT, #0CH	EI ² OS 転送を 12 回行う。
A/D コンバータ 設定	MOV ADCS0 #DBH	停止モード、開始チャネル AN3、終了チャネル AN3 (1 チャネル変換)
	MOV ADCS1 #A4H	外部エッジ起動、A/D 変換開始
EI ² OS 終了割込み シーケンス	MOV ADCS1 #00H	割込みからの復帰
	RET	

ICR3 : 割込み制御レジスタ

BAPL : バッファアドレスポインタ下位

BAPM : バッファアドレスポインタ中位

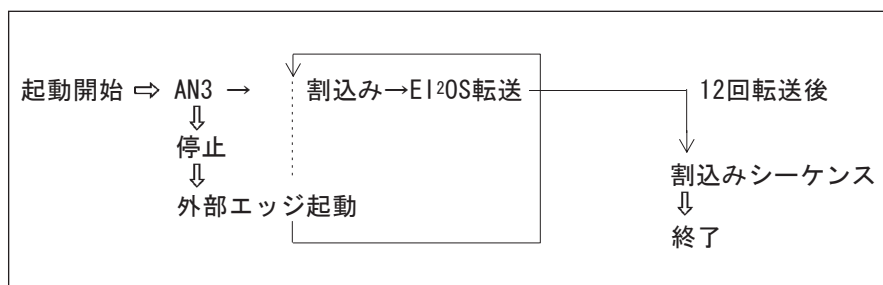
BAPH : バッファアドレスポインタ上位

ISCS : EI²OS ステータスレジスタ

I/OA : I/O アドレスカウンタ

DCT : データカウンタ

図 17.5-4 停止モード時の EI²OS の起動例



17.6 変換データ保護機能

A/D コンバータは、変換データ保護機能を持ち EI²OS を使って連続変換と複数のデータを確保できることを特長としています。

A/D データレジスタは 1 つで、その値は変換ごとに書き換えられています。そのため、連続で A/D 変換をすると 1 回の変換終了と共に変換データが格納されて前データが失われます。これを保護するために、本 A/D コンバータは変換が終了しても前データが EI²OS を使ってメモリへ転送されていないと変換データはレジスタに格納されず A/D は一時停止する機能を持っています。

■ 変換データ保護機能

一時停止の解除は EI²OS でメモリへ転送された後、行われます。

前データがメモリへ転送されていれば一時停止することなく A/D は連続して変換します。

< 注意事項 >

この機能は ADCS1 の INT, INTE ビットに関係しています。

変換データ保護機能は割込み許可 (INTE = 1) 状態でしか動作しないようになっています。

割込み禁止 (INTE = 0) の場合は、本機能は動作せず、連続で A/D 変換を行った場合は、次々に変換データはレジスタに格納されて旧データは失われます。

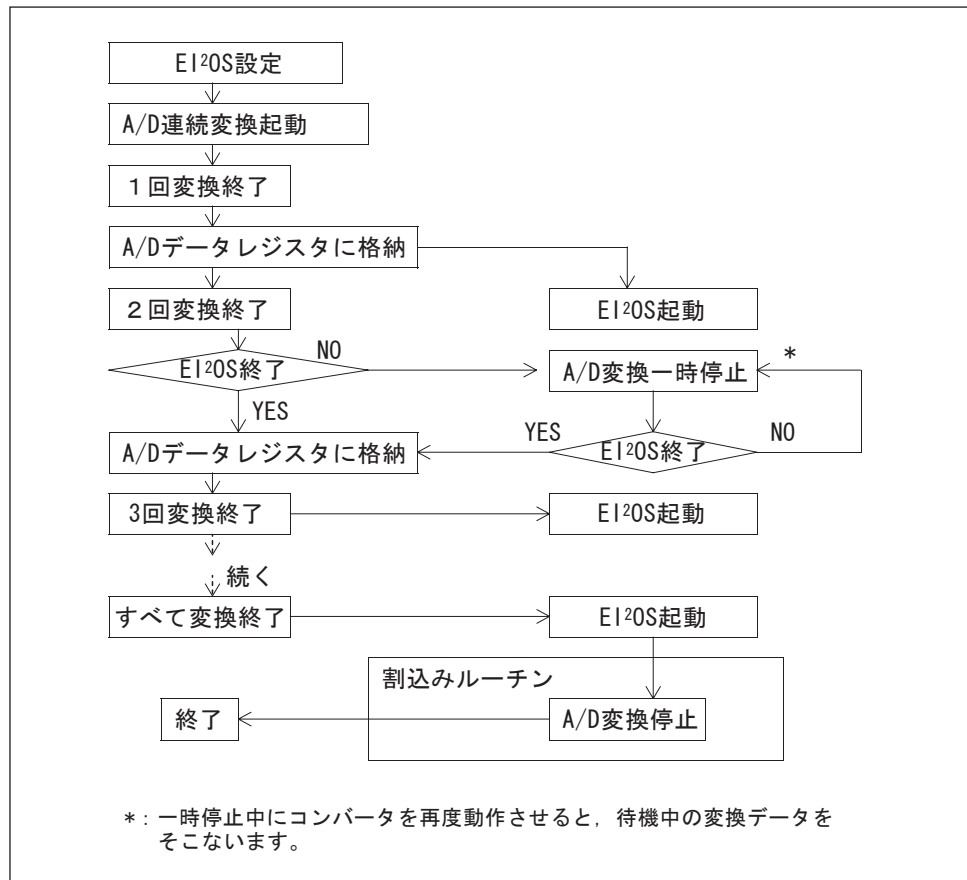
また、割込み許可 (INTE = 1) 状態で EI²OS を使わない場合は、INT ビットはクリアされないため、データ保護機能が働き、A/D は変換を一時停止状態にします。この場合は割込みシーケンスで INT ビットをクリアすると停止状態が解除されます。

EI²OS 動作中で、A/D コンバータが一時停止をしているときに割込みを禁止にすると、A/D コンバータが再動作し、転送する前に A/D データレジスタの内容が変わることがあります。

また、一時停止中に再起動をかけると待機データが壊れます。

■ 変換データ保護機能のフロー例 (EI²OS の場合)

図 17.6-1 変換データ保護機能のフロー例 (EI²OS の場合)



■ 変換データ保護機能の使用上の注意

A/D コンバータを外部トリガまたは内部タイマを使って起動する場合、ADCS1 レジスタの A/D 起動要因ビット STS1, STS0 で設定しますが、このときに外部トリガおよび内部タイマの入力値はインアクティブ側の状態で設定してください。アクティブ側にしておくと動作し始める場合があります。

STS1 および STS0 をセットする場合には必ず、ADTG に対して "1"(入力)を指定し内部タイマ(タイマ 2)に対しては "0"(出力)を指定してください。

第18章

UART0

UART0 の機能と動作について示します。

- 18.1 UART0 の特長
- 18.2 UART0 のブロックダイアグラム
- 18.3 UART0 のレジスタ
- 18.4 UART0 の動作
- 18.5 ボーレート
- 18.6 内部クロックおよび外部クロック
- 18.7 転送データフォーマット
- 18.8 パリティビット
- 18.9 割込み生成とフラグのセットタイミング
- 18.10 UART0 の応用例

18.1 UART0 の特長

UART0 は、外部デバイスと非同期 (調歩同期) 通信または CLK 同期通信を行うためのシリアル I/O ポートです。

■ UART0 の特長

UART0 には以下の特長があります。

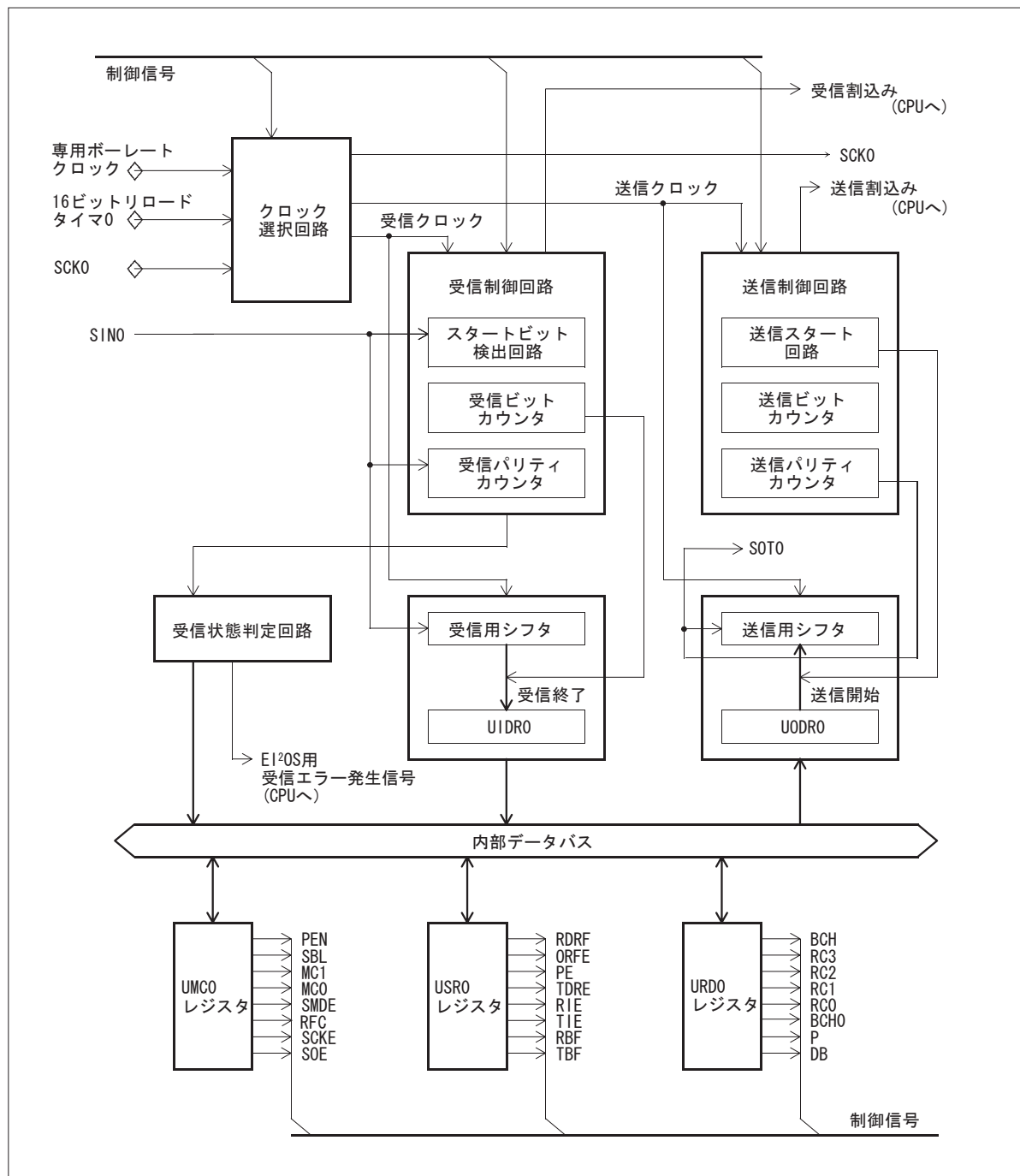
- 全二重ダブルバッファ
- CLK 同期と CLK 非同期調歩式データ転送のサポート
- マルチプロセッサモードのサポート (モード 2)
- 専用ボーレートジェネレータの内蔵 (12 種類)
- 外部クロック入力または内部タイマによる自由なボーレート設定のサポート
- 可変長データ (7 ~ 9 ビット [パリティなし], 6 ~ 8 ビット [パリティあり])
- エラー検出機能 (パリティ , フレーミング , オーバラン)
- 割込み機能 (受信割込みと送信割込み)
- NRZ 形式の転送フォーマット

18.2 UART0 のブロックダイアグラム

図 18.2-1 に , UART0 のブロックダイアグラムを示します。

■ UART0 のブロックダイアグラム

図 18.2-1 全体のブロックダイアグラム



18.3.1 シリアルモード制御レジスタ 0(UMC0)

シリアルモード制御レジスタ 0(UMC0) は, UART0 の動作モードを指定します。動作が停止中に動作モードを設定します。ただし, RFC ビットは動作中もアクセス可能です。

■ シリアルモード制御レジスタ 0 (UMC0)

シリアルモード制御レジスタ 0 (UMC0) のビット構成を図 18.3-2 に示します。

図 18.3-2 シリアルモード制御レジスタ 0(UMC0)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000020 _H	PEN	SBL	MC1	MC0	SMDE	RFC	SCKE	SOE	UMC0
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(W)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	

【bit 7】PEN(Parity ENable)

シリアルデータ I/O でパリティビットの追加 (送信時) または検出 (受信時) を指定します。モード 2 では "0" に設定します。

0: パリティなし

1: パリティあり

【bit 6】SBL(Stop Bit Length)

送信データのストップビット数を指定します。受信データについては, 最初のストップビットだけが認識され, それ以降のストップビットは無視されます。

0: 1 ビット長

1: 2 ビット長

【bit 5, bit 4】MC1, MC0 (Mode Control)

転送データの長さを制御します。表 18.3-1 に, 本ビットで選択可能な 4 つの転送モード (データ長) を示します。

表 18.3-1 UART0 動作モード

モード	MC1	MC0	データ長 ^{*1}
0	0	0	7 (6)
1	0	1	8 (7)
2 ^{*2}	1	0	8 + 1
3	1	1	9 (8)

*1: 括弧内の数字は, パリティ付きのデータ長を示します。

*2: いくつかのスレーブ CPU が単一ホスト CPU に接続している場合, モード 2 が使用されます。受信パリティチェック機能は使用不可のため, UMC0 レジスタ中の PEN を "0" に設定してください (詳細については, 「18.4 UART0 の動作」を参照)。送信データ長は 9 ビットで, パリティビットは追加できません。

【bit 3】SMDE (Synchro MoDe Enable)

SMDE は、転送方法を選択します。

0: 調歩式 CLK 同期転送 (調歩ビットによるクロック選択の同期転送)

1: 調歩式 CLK 非同期転送

【bit 2】RFC (Receiver Flag Clear)

"0" を書き込むと、USR0 レジスタ中の RDRF, ORFE, PE フラグがクリアされます。"1" 書込みは無効であり、読出し値は常に "1" になります。

< 注意事項 >

受信割込みが UART0 操作中許可されている場合、RDRF, ORFE, PE のどれかが "1" になっていれば、RFC には "0" のみを書き込みます。

【bit 1】SCKE (SCLK Enable)

CLK 同期モードで本ビットに "1" を書き込むと、ポート端子が UART0 シリアルクロック出力端子に切り換わり、同期クロックが出力されます。CLK 同期モードまたは外部クロックモードで "0" を設定してください。

0: 汎用入出力ポートとして機能し、シリアルクロックを出力しません。ポートが入力モード (DDR=0) に設定され、RC3 ~ RC0 が "1111_B" に設定されている場合、外部クロック入力端子として機能します。

1: UART0 シリアルクロック出力端子として機能します。

【bit 0】SOE (Serial Output Enable)

"1" を書き込むと、ポート端子が UART0 シリアルデータ出力端子に切り換わり、シリアル出力が可能になります。

0: ポート端子として機能し、シリアルデータを出力しません。

1: UART0 シリアルデータ出力端子 (SOT) として機能します。

18.3.2 シリアルステータスレジスタ 0(USR0)

シリアルステータスレジスタ 0(USR0) は、UART0 ポートの現在の状態を示します。

■ シリアルステータスレジスタ 0(USR0)

シリアルステータスレジスタ 0(USR0) のビット構成を図 18.3-3 に示します。

図 18.3-3 シリアルステータスレジスタ 0(USR0)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000021 _H	RDRF	ORFE	PE	TDRE	RIE	TIE	RBF	TBF	USR0
リード / ライト	(R)	(R)	(R)	(R)	(R/W)	(R/W)	(R)	(R)	
初期値	(0)	(0)	(0)	(1)	(0)	(0)	(0)	(0)	

【bit 15】RDRF(Receiver Data Register Full)

UIDR0(シリアルインプットデータレジスタ) の状態を示すフラグです。UIDR0 レジスタに受信データがロードされるとセットされ、UIDR0 レジスタを読み出すか、UMC0 レジスタ中の RFC に "0" を書き込むと、クリアされます。RIE がアクティブな場合、RDRF がセットされていれば受信割込み要求が生成されます。

0:UIDR0 中にデータなし

1:UIDR0 中にデータあり

【bit14】ORFE (Over-run/framing error)

受信中にオーバランまたはフレーミングエラーが発生した場合にセットされるフラグです。UMC0 レジスタ中の RFC に "0" を書き込むと、クリアされます。本フラグがセットされると、UIDR0 中のデータは無効になり、受信用シフトから UIDR0 へのロードが実行されません。RIE がアクティブな場合、ORFE がセットされていれば受信割込み要求が生成されます。

0: エラーなし

1: エラーあり

表 18.3-2 に、RDRF または ORFE による受信完了後の UIDR0 の状態を示します。

表 18.3-2 受信完了後の UIDR0 の状態

RDRF	ORFE	UIDR0 データの状態
0	0	空
0	1	フレーミングエラー
1	0	有効データ
1	1	オーバランエラー

オーバランまたはフレーミングエラーが発生した場合、UIDR0 中のデータは無効です。次のデータはフラグクリア後に受信できます。

【bit 13】 PE(Parity Error)

受信パリティエラーが発生した場合にセットされるフラグです。UMC0 レジスタ中の RFC に "0" を書き込むと、クリアされます。本フラグがセットされると、UIDR0 中のデータは無効になり、受信用シフトから UIDR0 へのロードが実行されません。RIE がアクティブな場合、PE がセットされていれば受信割込み要求が生成されます。

0: パリティエラーなし

1: パリティエラー発生

【bit 12】 TDRE(Transmitter Data Register Empty)

UODR0(アウトプットデータレジスタ 0) の状態を示すフラグです。UODR0 レジスタに送信データを書き込むと、クリアされます。データが送信用シフトにロードされ、転送が開始されるとセットされます。TIE がアクティブな場合、TDRE がセットされていれば送信割込み要求が生成されます。

0:UODR0 中にデータあり

1:UODR0 中にデータなし

【bit 11】 RIE (Receiver Interrupt Enable)

受信割込み要求を許可します。

0: 割込みを禁止します。

1: 割込みを許可します。

【bit 10】 TIE (Transmitter Interrupt Enable)

送信割込み要求を許可します。TDRE が "1" のとき送信割込みが許可されると、送信割込みが直ちに生成されます。

0: 割込みを禁止します。

1: 割込みを許可します。

【bit 9】 RBF (Receiver Busy Flag)

UART0 がインプットデータを受信中であることを示すフラグです。スタートビットが検出されるとセットされ、ストップビットが検出されるとクリアされます。

0: 受信側がアイドル状態

1: 受信側がビジー状態

【bit 8】 TBF (Transmitter Busy Flag)

UART0 がインプットデータを送信中であることを示すフラグです。送信データが UODR0 レジスタに書き込まれるとセットされ、送信が完了するとクリアされます。

0: 送信側がアイドル状態

1: 送信側がビジー状態

18.3.3 シリアルインプットデータレジスタ 0(UIDR0) と シリアルアウトプットデータレジスタ 0(UODR0)

シリアルインプットデータレジスタ 0 (UIDR0) は, シリアルデータの入力 (受信) 用レジスタであり, シリアルアウトプットデータレジスタ 0 (UODR0) は, シリアルデータの出力 (送信) 用レジスタです。

■ シリアルインプットデータレジスタ 0 (UIDR0) とシリアルアウトプットデータレジスタ 0 (UODR0)

シリアルインプットデータレジスタ 0 (UIDR0) とシリアルアウトプットデータレジスタ 0 (UODR0) のビット構成を図 18.3-4 に示します。

図 18.3-4 シリアルインプットデータレジスタ 0 (UIDR0) と
シリアルアウトプットデータレジスタ 0 (UODR0)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000022 _H	D7	D6	D5	D4	D3	D2	D1	D0	UIDR0(リード) UODR0(ライト)
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

データ長が 6 ビットの場合最大有効ビットの D7 と D6 が無視され, データ長が 7 ビットの場合最大有効ビットの D7 が無視されます。USR0 レジスタ中の TDRE="1" のときのみ UODR0 に書き込み, USR0 レジスタ中の RDRF="1" のときのみ UIDR0 を読み出します。

18.3.4 レート・データレジスタ 0(URD0)

レート・データレジスタ 0(URD0) は , UART0 用のデータ転送速度 (ボーレート) を選択するレジスタです。送信データ長が 9 ビットするとき , データの最大有効ビット (bit8) を保持します。UART0 が停止中にボーレートとパリティを設定してください。

■ レート・データレジスタ 0 (URD0)

レート・データレジスタ 0 (URD0) のビット構成を図 18.3-5 に示します。

図 18.3-5 レート・データレジスタ 0 (URD0)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000023 _H	BCH	RC3	RC2	RC1	RC0	BCH0	P	D8	URD0
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)	(X)	

【bit 15, bit 10】BCH, BCH0 (Baud rate clock change)

ボーレートクロック用のマシンサイクルを指定します (詳細については , 「18.4 UART0 の動作」を参照)。

表 18.3-3 マシンサイクルの設定例

BCH	BCH0	分周比	各マシンサイクルの設定例
0	0	-	設定禁止
0	1	4 分周	16 MHz マシンサイクル 16/4 = 4 MHz
1	0	3 分周	12 MHz マシンサイクル 12/3 = 4 MHz
1	1	5 分周	10 MHz マシンサイクル 10/5 = 2 MHz

< 注意事項 >

BCH と BCH0 を "00_B" に設定しないでください。

【bit 14 ~ bit 11】RC3, RC2, RC1, RC0 (Rate control)

UART0 ポート用のクロック入力を選択します (詳細については「18.4 UART0 の動作」を参照)。

表 18.3-4 クロック入力選択

RC3 ~ RC0	クロック入力
0000 _B ~ 1011 _B	専用ボーレートジェネレータ
1101 _B	16 ビットリロードタイマ 0
1111 _B	外部クロック

< 注意事項 >

レート制御ビットを "1100_B", "1110_B" に設定しないでください。

【bit 9】P(Parity)

パリティがアクティブのとき, 偶数 / 奇数パリティを設定します (PEN = "1")。

0: 偶数パリティ

1: 奇数パリティ

【bit 8】D8

bit2 または 3 (9 ビットデータ長) およびパリティなしでの転送データの bit8 を保持します。読出し時には UIDR0 レジスタの bit8 として, 書込み時には UODR0 レジスタの bit8 として扱われます。そのほかのモードでは意味を持ちません。USR0 レジスタ中で TDRE = 1 のときのみ D8 に書き込みます。

18.4 UART0 の動作

表 18.4-1 に、UART0 の動作モードを示します。モードを切り換えるには、UMC0 レジスタに値を設定します。

■ UART0 の動作モード

表 18.4-1 UART0 の動作モード

モード	パリティ	データ長	クロックモード	ストップビット長*
0	ON	6	CLK 非同期または CLK 同期	1 ビットまたは 2 ビット
	OFF	7		
1	ON	7		
	OFF	8		
2	OFF	8+1		
3	ON	8		
	OFF	9		

※: ストップビット長については、送信動作のみ指定が可能です。受信動作については常に 1 ビット長となります。UART0 は上記モード以外では動作しませんので、設定しないでください。

< 注意事項 >

UART0 は、調歩クロック同期転送を使用するので、クロック同期転送でもスタートビットとストップビットをデータに追加します。

18.5 ボーレート

専用ボーレートジェネレータ使用時には、次の 2 種類があります。

- CLK 同期ボーレート
- CLK 非同期ボーレート

■ CLK 同期ボーレート

BCH, BCH0, RC3, RC2, RC1 の 5 つの URD0 レジスタビットが CLK 同期転送用のボーレートを選択します。

最初に、BCH, BCH0 を使用してマシンのクロック分周比を選択します。

BCH	BCH0	
0	1	4 で割る [例 , 16 MHz 時 $16/4 = 4$ MHz]
1	0	3 で割る [例 , 12 MHz 時 $12/3 = 4$ MHz]
1	1	5 で割る [例 , 10 MHz 時 $10/5 = 2$ MHz]

次に、上記で選択したクロックの分周比を RC3, RC2, RC1 に設定します。CLK 同期転送には、以下の 3 つの設定が可能です。それ以外の設定は禁止されます。

RC3	RC2	RC1	
0	1	0	2 で割る [例 , 4 MHz 時 $4/2 = 2.0$ M bps]
0	1	1	4 で割る [例 , 4 MHz 時 $4/4 = 1.0$ M bps]
1	0	0	8 で割る [例 , 4 MHz 時 $4/8 = 0.5$ M bps] (2 MHz では、上記の例の半分の速度になります。)

■ CLK 非同期ボーレート

BCH, BCH0, RC3, RC2, RC1, RC0 の 6 つの URD レジスタが CLK 非同期転送用のボーレートを選択します。

最初に、BCH, BCH0 を使用してマシンのクロック分周比を選択します。

BCH	BCH0	
0	1	4 で割る [例 , 16 MHz 時 $16/4 = 4$ MHz]
1	0	3 で割る [例 , 12 MHz 時 $12/3 = 4$ MHz]
1	1	5 で割る [例 , 10 MHz 時 $10/5 = 2$ MHz]

次に、上記で選択したクロックの非同期転送クロック分周比を RC3, RC2, RC1, RC0 に設定します。CLK 同期転送には、以下の設定が可能です。

RC3	RC2	RC1		RC0	
0	0	0	8 × 1 で割る	0	12 で割る
0	1	0	8 × 2 で割る	1	13 で割る
0	1	1	8 × 4 で割る		
1	0	0	8 × 8 で割る		
0	0	1	割らない	0	設定禁止
1	0	1	8 で割る	1	8 で割る

上記の 12 個のボーレートが選択可能であり, CLK 同期ボーレートの計算方法の公式を以下に示します。

$$\text{ボーレート} = \frac{\phi / 4}{2^{m-1}} \text{ bps (マシンサイクル = 16 MHz)}$$

$$\text{ボーレート} = \frac{\phi / 3}{2^{m-1}} \text{ bps (マシンサイクル = 12 MHz)}$$

$$\text{ボーレート} = \frac{\phi / 5}{2^{m-1}} \text{ bps (マシンサイクル = 10 MHz)}$$

ここで, ϕ はマシンサイクルであり, m は RC3 ~ RC1 の 10 進数表記です。

< 注意事項 >

$m = 0$ または $m = 1$ の場合, 上記公式は計算不可です。

CLK 非同期ボーレートが -1 % ~ +1 % の範囲にある場合, データ転送が可能です。ボーレートは, CLK 非同期ボーレートを 8 × 13, 8 × 12, 8 で割った値です。

表 18.5-1 に, マシンサイクルが 16MHz, 12MHz, 10MHz 時の例を示します。表中の "_" でマークした設定は使用しないでください。

表 18.5-1 ボーレート

RC 3	RC 2	RC 1	RC 0	CLK 非同期 (μs/bps)			CLK 非同期分周比	CLK 同期 (μs/bps)		
				16 MHz	12 MHz	10 MHz		16 MHz	12 MHz	10 MHz
				BCH/ BCH0=01 _B	BCH/ BCH0=10 _B	BCH/ BCH0=11 _B		BCH/ BCH0=01 _B	BCH/ BCH0=10 _B	BCH/ BCH0=11 _B
0	0	0	0	-	-	48/ 20833	8 × 12	-	-	-
0	0	0	1	26/ 38460	26/ 38460	52/ 19230	8 × 13	-	-	-
0	0	1	0	-	-	-	8	-	-	-
0	0	1	1	2/500000	2/500000	4/250000	8	-	-	-
0	1	0	0	48/ 20833	48/ 20833	96/ 10417	8 × 12	-	-	-
0	1	0	1	52/ 19230	52/ 19230	104/ 9615	8 × 13	0.5 / 2M	0.5 / 2M	1 / 1M
0	1	1	0	96/ 10417	96/ 10417	192/ 5208	8 × 12	-	-	-
0	1	1	1	104/ 9615	104/ 9615	208/ 4808	8 × 13	1 / 1M	1 / 1M	2 / 500K
1	0	0	0	192/ 5208	192/ 5208	-	8 × 12	-	-	-
1	0	0	1	208/ 4808	208/ 4808	416/ 2404	8 × 13	2 / 500K	2 / 500K	4 / 250K
1	0	1	0	-	-	-	8	-	-	-
1	0	1	1	16/ 62500	16/ 62500	32/ 31250	8	-	-	-

18.6 内部クロックおよび外部クロック

RC3 ~ RC0 を "1101_B" に設定すると, 16 ビットリロードタイマからのクロック信号を選択します。RC3 ~ RC0 を "1111_B" に設定すると, 外部クロックを選択します。

■ 内部クロックおよび外部クロック

CLK 非同期ボーレートは, CLK 同期ボーレートを 8 で割ったものです。また, CLK 非同期ボーレートが選択したボーレート -1 % ~ +1 % の範囲にある場合, データ転送が可能です。表 18.6-1 に, 内部タイマをクロックとして選択した場合のボーレートを示します。本表の値は, マシンサイクルを 7.3728 MHz として計算しています。ただし, 表中の "_" でマークした設定は使用しないでください。

$$\text{ボーレート} = \frac{\phi / X}{8 \times 2^{(n+1)}} \text{ bps}$$

ϕ : マシンサイクル
 X : 内部タイマのカウントクロックソースを割る比率
 n : リロード値 (10 進数)

表 18.6-1 ボーレートとリロード値

リロード値 ボーレート	X=2 ¹ (マシンクロックの 2 分周)	X=2 ³ (マシンクロックの 8 分周)
76800	2	
38400	5	
19200	11	2
9600	23	5
4800	47	11
2400	95	23
1200	191	47
600	383	95
300	767	191

表中の値は, 16 ビットリロードタイマのリロードカウンタ動作のリロード値 (10 進数) です。

18.7 転送データフォーマット

UART0 は、NRZ(non-return-to-zero) 形式のデータのみを扱います。図 18.7-1 に、送受信クロックと CLK 同期モードのデータとの関係を示します。

■ 転送データフォーマット

図 18.7-1 転送データフォーマット

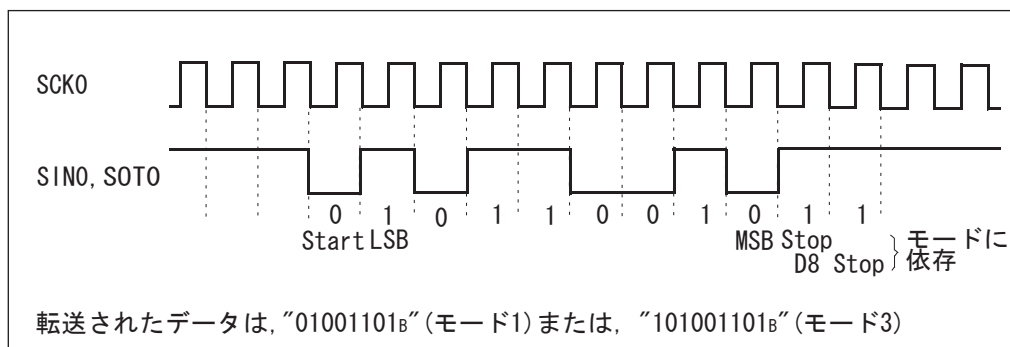


図 18.7-1 に示すように、転送データは必ずスタートビット ("L" レベルデータ) より始まり、LSB ファーストで指定されたデータビット長転送が行われ、ストップビット ("H" レベルデータ) で終了します。外部クロックを選択している場合は、常にクロックを入力してください。

内部クロック (専用ボーレートジェネレータまたは 16 ビットリロードタイマ) を選択している場合、クロックは連続的に出力されます。CLK 同期転送を使用する場合、選択したボーレートクロックが安定 (2 ボーレートクロックサイクル) するまでデータ転送を開始しないでください。

CLK 非同期転送を使用する場合、UMC0 レジスタ中の SCKE ビットを "0" に設定し、クロック出力を無効にしてください。SIN0 と SOT0 の転送データフォーマットは、図 18.7-1 に示すものと同じです。

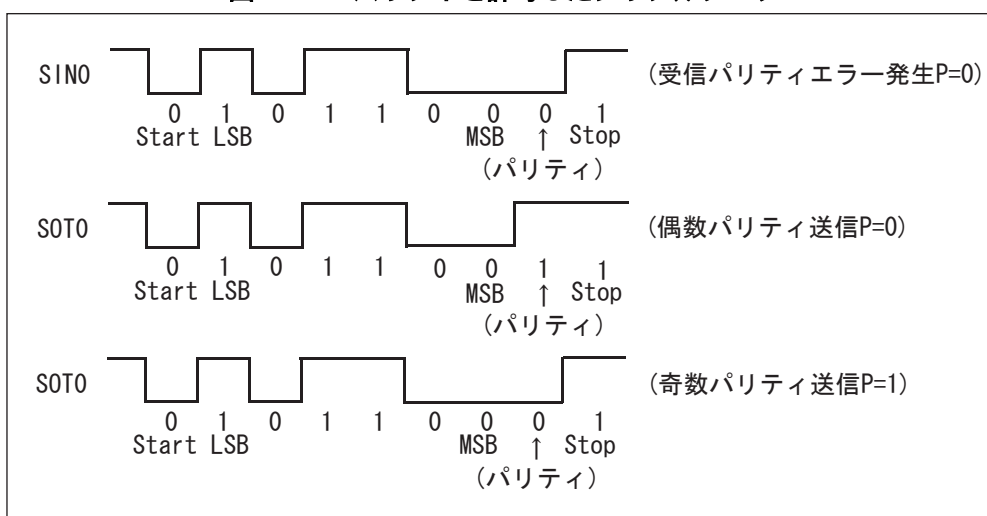
18.8 パリティビット

URD0 レジスタ中の P(パリティ) ビットは , パリティが許可されている場合に偶数 / 奇数パリティのどちらを使用するかを指定します。UMC0 レジスタ中の PEN(パリティイネーブル) ビットはパリティを許可します。

■ パリティビット

偶数パリティ設定時に図 18.8-1 に示すデータを SIN0 に入力すると , 受信パリティエラーが発生します。図 18.8-1 には , "001101_B" を偶数 / 奇数パリティで送信時に送信されるデータも示しています。

図 18.8-1 パリティを許可したシリアルデータ



18.9 割込み生成とフラグのセットタイミング

UART0 には、2 つの割込み要因と 6 個のフラグがあります。2 つの割込み要因には、受信割込みと送信割込みがあります。6 個のフラグとは、RDRF, ORFE, PE, TDRE, RBF, TBF です。受信については、RDRF, ORFE, PE フラグが割込みを要求します。送信については、TDRE フラグが割込みを要求します。

■ 6 つのフラグのセットタイミング

● RDRF フラグ

RDRF は UIDR0 レジスタに受信データがロードされるとセットされます。UMC0 レジスタ中の RFC に "0" を書き込むか、UIDR0 レジスタを読み出すと、クリアされます。

● ORFE フラグ

ORFE フラグは、オーバランまたはフレーミングエラーフラグです。受信エラーが発生した場合にセットされ、UMC0 レジスタ中の RFC に "0" を書き込むと、クリアされます。

● PE フラグ

PE フラグは、受信パリティエラーフラグです。受信パリティエラーが発生した場合にセットされ、UMC0 レジスタ中の RFC に "0" を書き込むと、クリアされます。パリティ検出機能はモード 2 では無効なので注意してください。

● TDRE フラグ

TDRE フラグは、UODR0 レジスタが空になり書込み可能になったときにセットされ、UODR0 レジスタに書込みを行うと、クリアされます。上記の 4 個のフラグ (RDRF, ORFE, PE, TDRE) は、送受信割込みを起動します。

● RBF と TBF フラグ

RBF と TBF フラグは、受信または送信が進行中であることを示します。RBF は受信中にアクティブになります。TBF は送信中にアクティブになります。

18.9.1 受信動作時 (モード 0, モード 1, モード 3) のフラグのセットタイミング

RDRF, ORFE, PE フラグは, 受信転送が終了し最後のストップビットを検出するときにセットされ, CPU への割込み要求が発生します。ORFE, PE のどちらかのビットがアクティブ時には, UIDR0 中のデータは無効となります。

■ 受信動作時 (モード 0, モード 1, モード 3) のフラグのセットタイミング

図 18.9-1, 図 18.9-2, 図 18.9-3 に, RDRF, ORFE, PE フラグのセットタイミングを示します。

図 18.9-1 RDRF のセットタイミング (モード 0, 1, 3)

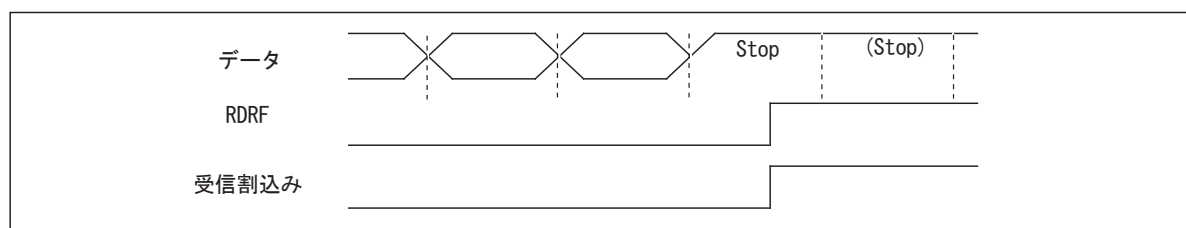


図 18.9-2 ORFE のセットタイミング (モード 0, 1, 3)

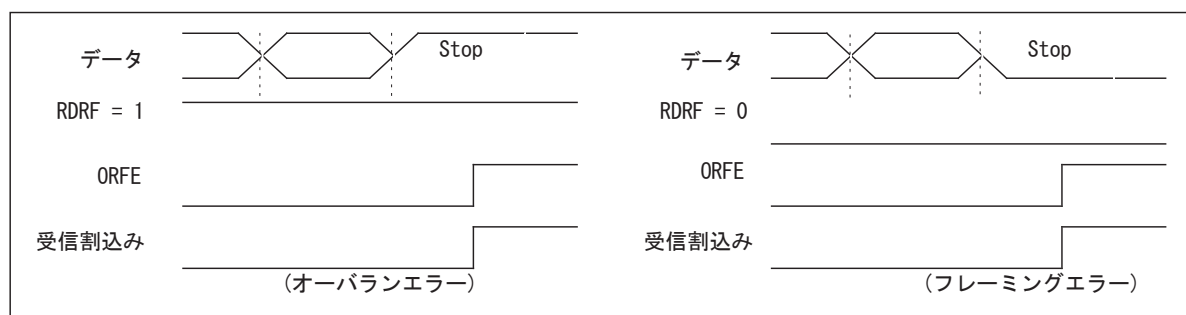
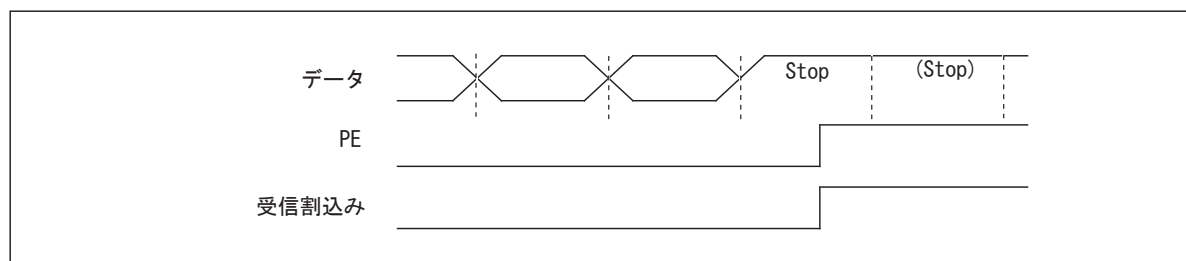


図 18.9-3 PE のセットタイミング (モード 0, 1, 3)



18.9.2 受信動作時 (モード 2) のフラグのセットタイミング

RDRF フラグは、最後のストップビットを検出し、最後のデータビット (D8) の値が "1" で受信転送が終了するときにセットされます。

ORFE フラグは、最後のデータビット (D8) の値に無関係に、最後のストップビットを検出するときにセットされます。ORFE ビットがアクティブ時には、UIDR0 中のデータは無効となります。

どれかのフラグがセットされると、CPU への割込み要求が発生します (モード 2 の使用法の詳細については、「18.10 UART0 の応用例」を参照)。

■ 受信動作時 (モード 2) のフラグのセットタイミング

図 18.9-4 RDRF のセットタイミング (モード 2)

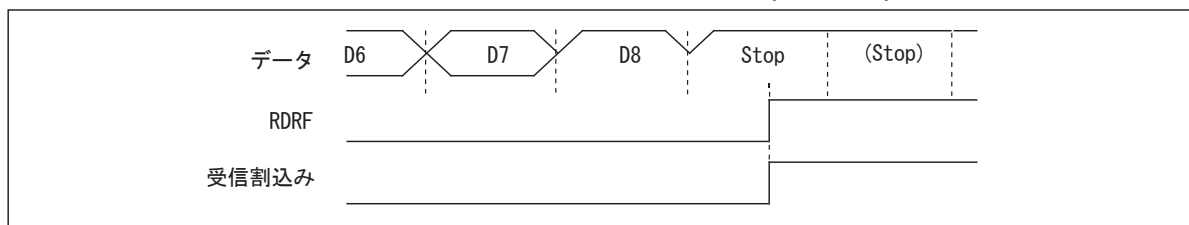
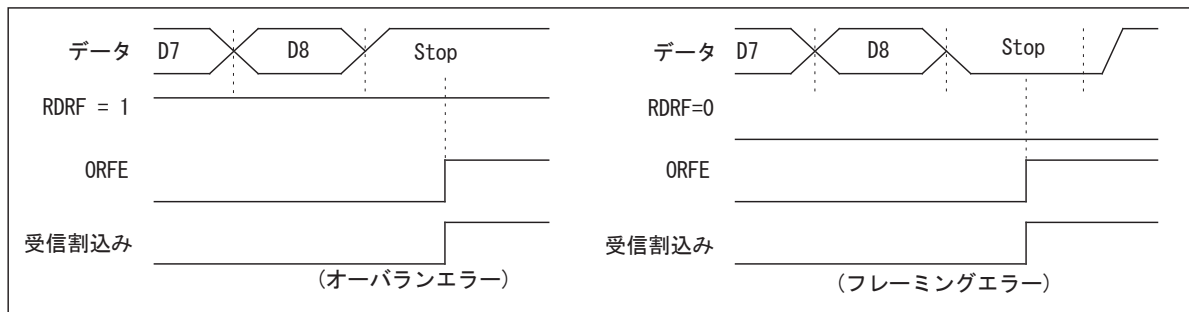


図 18.9-5 ORFE のセットタイミング (モード 2)

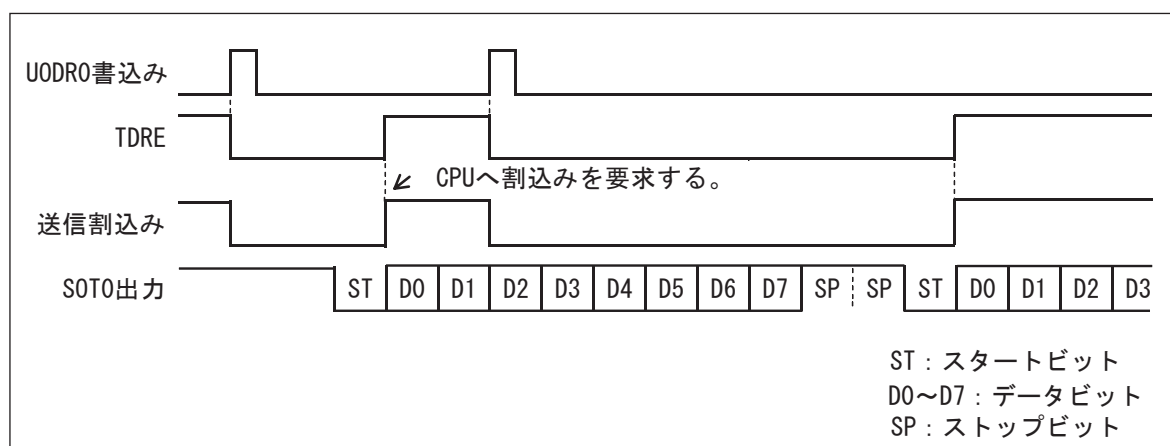


18.9.3 送信動作時のフラグのセットタイミング

送信動作時に UODR0 レジスタに書き込まれたデータは、内部シフトレジスタに転送されます。このとき、TDRE フラグがセットされ、CPU への割込み要求が発生して、次のデータを UODR0 フラグに書き込みができるようになります。

■ 送信動作時のフラグのセットタイミング

図 18.9-6 TDRE のセットタイミング (モード 0)



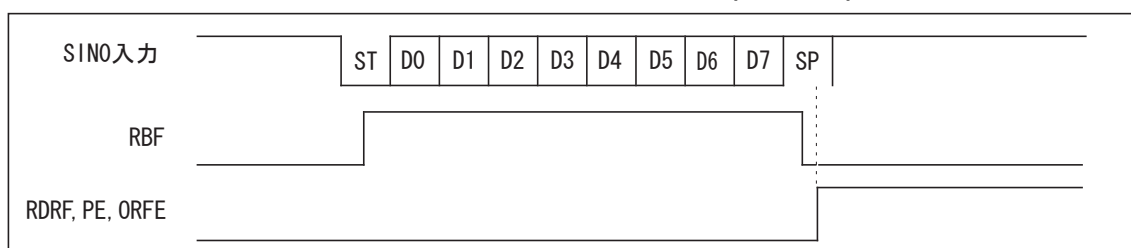
18.9.4 送受信動作時のステータスフラグ

送受信動作時に、RBF フラグは、スタートビットが検出されるとセットされ、ストップビットが検出されるとクリアされます。UIDR0 中の受信データは、RBF クリアタイミングではまだ無効状態です。UIDR0 中のデータは、RDRF セットタイミングで有効になります。

■ 送受信動作時のステータスフラグ

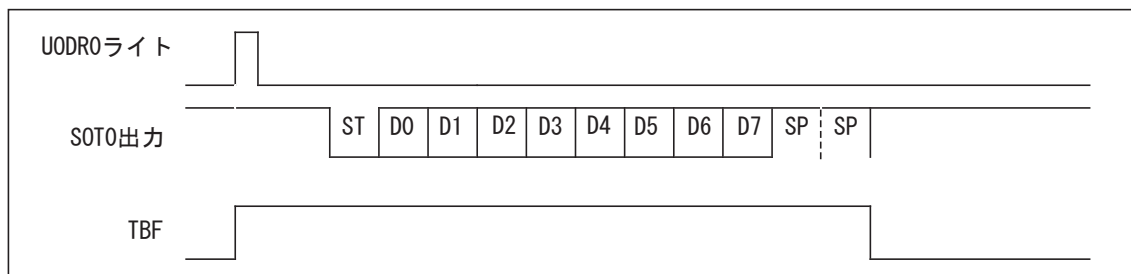
図 18.9-7 に、RBF と受信割込みフラグのタイミングとの関係を示します。

図 18.9-7 RBF のセットタイミング (モード 0)



UODR0 に送信データを書き込むと、TBF がセットされ、送信が完了するとクリアされます。

図 18.9-8 TBF のセットタイミング (モード 0)



< 注意事項 >

SIN0 入力端子が "1" に固定されていない限り、受信動作はリセットのリリース後に開始します。このため、モードを設定する前に、UMC0 レジスタ中の RFC に "0" を書き込み、セットされているすべての受信フラグをクリアしてください。

USR0 レジスタ中の RBF と TBF フラグが "0" のとき、通信モードをセットしてください。モード設定中の送受信データは保証されません。

■ EI²OS(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 (EI²OS) の詳細については、「3.6 拡張インテリジェント I/O サービス (EI²OS)」を参照してください。

18.10 UART0 の応用例

モード 2 は、1 台のホスト CPU に数台のスレーブ CPU が接続されるような場合に使用されます (図 18.10-1 を参照)。

■ UART0 の応用例

図 18.10-1 に示すように、通信はホスト CPU がアドレスデータを転送することによって始まります。アドレスデータは、9 番目のビット (D8) が "1" にセットされています。これにより通信先となるスレーブ CPU が選択され、ユーザが決定したプロトコルを使用してホスト CPU との通信を行います。通常データは D8 が "0" にセットされています。非選択のスレーブ CPU は、次の通信セッションが始まるまでスタンバイ状態で待機します。

図 18.10-2 に、本モードでの動作フローチャートを示します。

このモードにおいては、パリティチェック機能は使用できませんので、UMC0 レジスタの PEN ビットは "0" としてください。

図 18.10-1 モード 2 使用時のシステム構築例

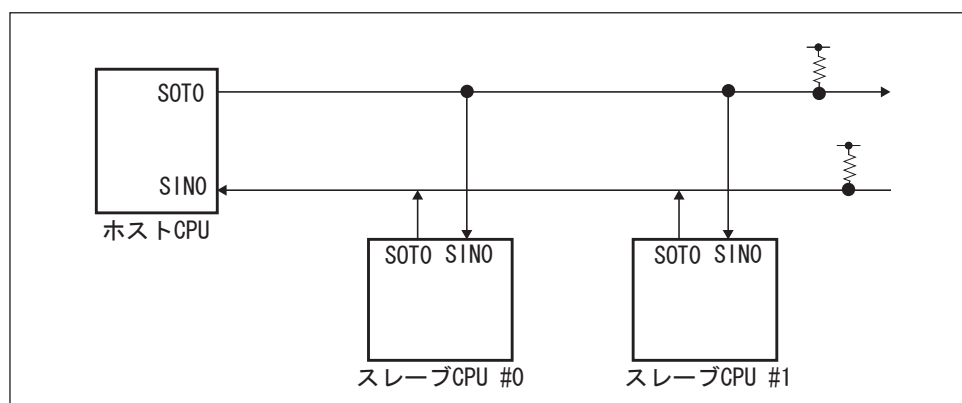
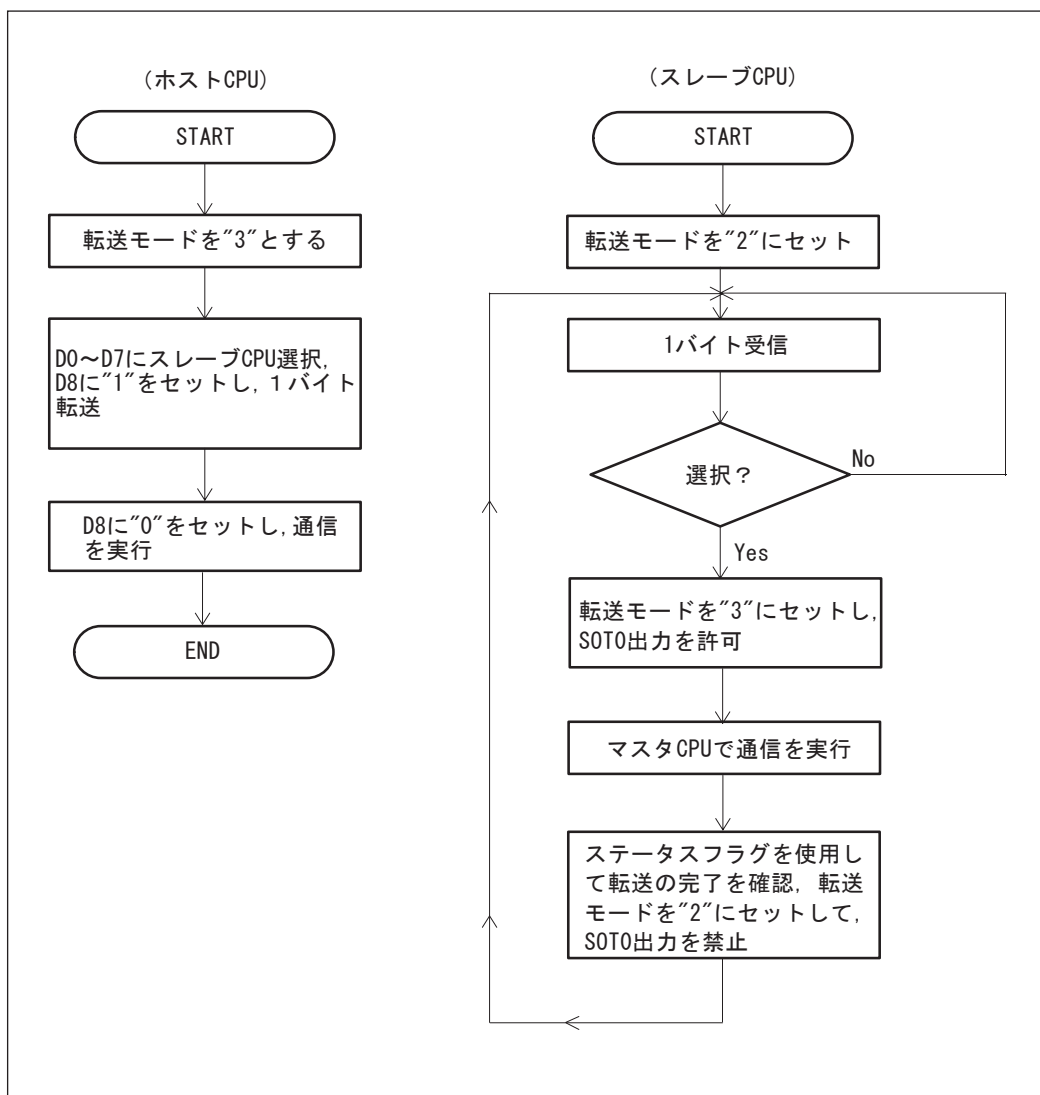


図 18.10-2 モード 2 使用時の通信フローチャート



第19章

UART1(SCI)

UART1(SCI) の機能と動作について示します。

- 19.1 UART1 の特長
- 19.2 UART1 のブロックダイアグラム
- 19.3 UART1 のレジスター一覧
- 19.4 UART1 の動作モードとクロック選択
- 19.5 UART1 のフラグと割込み発生要因
- 19.6 UART1 の割込みとフラグのセットタイミング
- 19.7 ネガティブクロック動作
- 19.8 UART1 の応用例と使用上の注意

19.1 UART1 の特長

UART1 は、非同期 (調歩同期) 通信または CLK 同期通信を行うためのシリアル I/O ポートです。

■ UART1 の特長

UART には以下の特長があります。

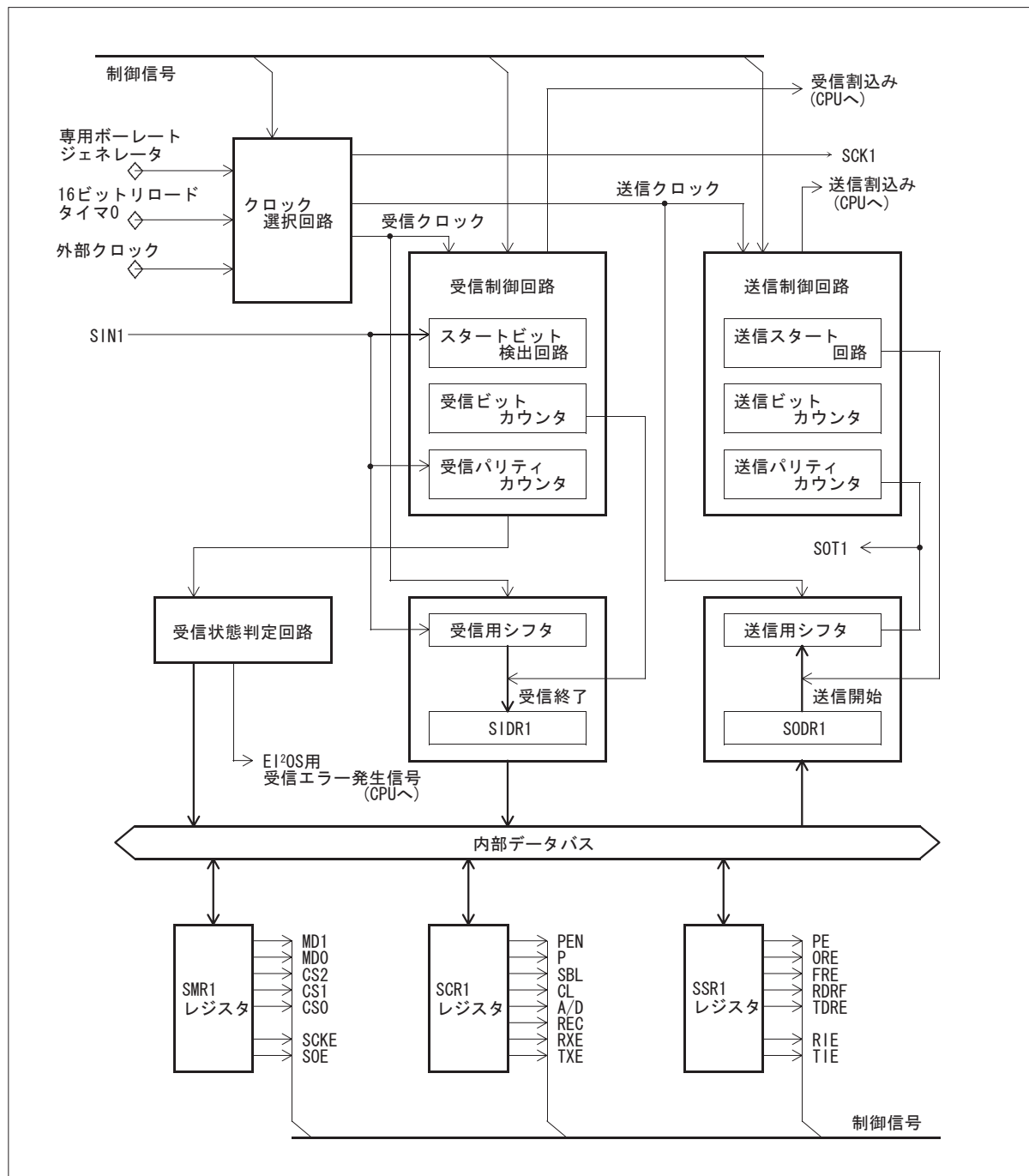
- 全二重ダブルバッファ
 - 非同期 (調歩同期), CLK 同期通信が可能
 - マルチプロセッサモードのサポート
 - 専用ボーレートジェネレータ内蔵
 - 非同期 : 62500 / 38460 / 31250 / 19230 / 9615 /
4808 / 2404 / 1202 bps
 - CLK 同期 : 1M / 500K / 250K / 125K / 62.5K bps
- } 内部マシンのクロック
6, 8, 10, 12, 16 MHz 時
- 外部クロック入力と内部タイマによる自動ボーレート設定
 - エラー検出機能 (パリティ , フレーミング , オーバラン)
 - 転送信号は NRZ 符号
 - インテリジェント I/O サービスのサポート

19.2 UART1 のブロックダイアグラム

図 19.2-1 に , UART1 のブロックダイアグラムを示します。

■ UART1 のブロックダイアグラム

図 19.2-1 UART1 のブロックダイアグラム



19.3.1 シリアルモードレジスタ 1(SMR1)

シリアルモードレジスタ 1(SMR1) は、UART1 の動作モードを指定します。動作モードの設定は動作停止中に行い、動作中にこのレジスタへの書込みは行わないでください。

■ シリアルモードレジスタ 1(SMR1)

シリアルモードレジスタ 1(SMR1) のビット構成を図 19.3-3 に示します。

図 19.3-3 シリアルモードレジスタ 1(SMR1)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000024 _H	MD1	MD0	CS2	CS1	CS0	予約	SCKE	SOE	SMR1
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)	

【bit 7, bit 6】MD1, MD0 (MoDe select):

このビットでは、UART1 の動作モードを選択します。表 19.3-1 に、動作モード選択設定を示します。

表 19.3-1 動作モード選択設定

モード	MD1	MD0	動作モード
0	0	0	非同期 (調歩同期) ノーマルモード
1	0	1	非同期 (調歩同期) マルチプロセッサモード
2	1	0	CLK 同期モード
-	1	1	設定禁止

< 注意事項 >

モード 1 の CLK 非同期モード (マルチプロセッサ) とは、1 台のホスト CPU に数台のスレーブ CPU が接続される使用法です。本リソースでは、受信データのデータ形式を判別できません。したがって、マルチプロセッサモードのマスタのみをサポートします。

また、この設定ではパリティチェック機能は使用できませんので SCR1 レジスタの PEN は "0" に設定してください。

【bit 5 ~ bit 3】CS2, CS1, CS0 (Clock Select):

このビットでは、ボーレートクロックソースを選択します。専用ボーレートジェネレータを選択した場合には、同時にボーレートも決定されます。表 19.3-2 にクロック入力選択設定を示します。

表 19.3-2 クロック入力選択設定

CS2 ~ CS0	クロック入力
000 _B ~ 100 _B	専用ボーレートジェネレータ
101 _B	予約
110 _B	内部タイマ*
111 _B	外部クロック

※: 内部タイマを選択した場合、MB90540/545 シリーズでは 16 ビットリロードタイマ 0 出力が選択されます。

【bit 2】予約ビット

必ず "0" を書き込んでください。

【bit 1】SCKE (SCI_K Enable):

CLK 同期モード (モード 2) で通信を行う場合、SCK1 端子をクロック入力端子として使うか、クロック出力端子として使うかを指定します。

CLK 非同期モード時または外部クロックモード時では "0" に設定してください。

0: クロック入力端子として機能します。

1: クロック出力端子として機能します。

< 注意事項 >

クロック入力端子として使うには、外部クロックソースが選択されている必要があります。

【bit 0】SOE (Serial Output Enable):

汎用 I/O ポート端子と兼用されている外部端子 (SOT1) を、シリアル出力端子として使うか、I/O ポート端子として使うかを指定します。

0: 汎用 I/O ポート端子として機能します。

1: シリアルデータ出力端子 (SOT1) として機能します。

19.3.2 シリアル制御レジスタ 1 (SCR1)

シリアル制御レジスタ 1 (SCR1) は、シリアル通信を行う場合の転送プロトコルを制御します。

■ シリアル制御レジスタ 1 (SCR1)

シリアル制御レジスタ 1(SCR1) のビット構成を図 19.3-4 に示します。

図 19.3-4 シリアル制御レジスタ 1 (SCR1)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000025 _H	PEN	P	SBL	CL	A/D	REC	RXE	TXE	SCR1
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(W)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(1)	(0)	(0)	

【bit 15】PEN (Parity ENable)

シリアル通信において、パリティを付加してデータ通信を行うかどうかを指定します。

0: パリティなし

1: パリティあり

< 注意事項 >

パリティを付加できるのは、非同期 (調歩同期) 通信モードのノーマルモード (モード 0) のみです。マルチプロセッサモード (モード 1) および CLK 同期通信 (モード 2) では、パリティを付加することはできません。

【bit 14】P (Parity)

パリティを付加してデータ通信を行うとき、偶数 / 奇数パリティを指定します。

0: 偶数パリティ

1: 奇数パリティ

【bit 13】SBL (Stop Bit Length)

非同期 (調歩同期) 通信を行うときのフレームエンドマークである、ストップビットのビット長を指定します。

0:1 ストップビット

1:2 ストップビット

【bit 12】CL (Character Length)

送受信する 1 フレームのデータ長を指定します。

0:7 ビットデータ

1:8 ビットデータ

< 注意事項 >

7 ビットデータを扱えるのは、非同期 (調歩同期) 通信のうちのノーマルモード (モード 0) のみです。マルチプロセッサモード (モード 1) および CLK 同期通信 (モード 2) では、8 ビットデータを使用してください。

【bit 11】A/D (Address/Data)

非同期 (調歩同期) 通信のマルチプロセッサモード (モード 1) において、送信するフレームのデータ形式を指定します。

0: データフレーム

1: アドレスフレーム

【bit 10】REC (Receiver Error Clear)

"0" 書込みで、SSR1 レジスタのエラーフラグ (PE, ORE, FRE) をクリアします。

"1" 書込みは無効であり、読出し値は常に "1" になります。

【bit 9】RXE (Receiver Enable)

UART1 の受信動作を制御します。

0: 受信動作を禁止します。

1: 受信動作を許可します。

< 注意事項 >

受信中 (受信シフトレジスタにデータが入力されているとき) に受信動作を禁止した場合には、そのフレームの受信を完了し受信データバッファ SDR1 レジスタに受信データを格納したところで受信動作を停止します。

【bit 8】TXE (Transmitter Enable)

UART1 の送信動作を制御します。

0: 送信動作を禁止します。

1: 送信動作を許可します。

< 注意事項 >

送信中 (送信レジスタからデータが出力されているとき) に送信動作を禁止した場合は、送信データバッファ SODR1 レジスタにデータがなくなった後に送信動作を停止します。

19.3.3 シリアルインプットデータレジスタ 1(SIDR1)/ シリアルアウトプットデータレジスタ 1(SODR1)

シリアルインプットデータレジスタ 1(SIDR1)/シリアルアウトプットデータレジスタ 1(SODR1) レジスタは、受信 / 送信用のデータバッファレジスタです。

■ シリアルインプットデータレジスタ 1(SIDR1)/シリアルアウトプットデータレジスタ 1 (SODR1)

シリアルインプットデータレジスタ 1(SIDR1) とシリアルアウトプットデータレジスタ 1 (SODR1) のビット構成を図 19.3-5 に示します。

図 19.3-5 シリアルインプットデータレジスタ 1(SIDR1)/
シリアルアウトプットデータレジスタ 1(SODR1)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000026 _H	D7	D6	D5	D4	D3	D2	D1	D0	SIDR1(リード) SODR1(ライト)
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

シリアルインプットデータレジスタ 1(SIDR1) 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。シリアルアウトプットデータレジスタ 1(SODR1) 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。データ長が 7 ビットの場合、上位 1 ビット (D7) は無効データとなります。SODR1 レジスタへの書込みは、SSR1 レジスタの TDRE が "1" のときに行ってください。

< 注意事項 >

このアドレスへの書込みは SODR1 レジスタへの書込みを、読出しは SIDR1 レジスタの読出しを意味します。

19.3.4 シリアルステータスレジスタ 1(SSR1)

シリアルステータスレジスタ 1(SSR1) は, UART1 の動作状態を表すフラグで構成されています。

■ シリアルステータスレジスタ 1(SSR1)

シリアルステータスレジスタ 1 (SSR1) のビット構成を図 19.3-6 に示します。

図 19.3-6 シリアルステータスレジスタ 1(SSR1)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000027 _H	PE	ORE	FRE	RDRF	TDRE		RIE	TIE	SSR1
リード / ライト	(R)	(R)	(R)	(R)	(R)	(-)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(1)	(-)	(0)	(0)	

【bit 15】PE (Parity Error)

受信時にパリティエラーが発生したときにセットされる, 割込み要求フラグです。一度セットされたフラグをクリアするには, SCR1 レジスタの REC ビット (bit 10) に "0" を書き込みます。

本ビットがセットされた場合には, SISR1 レジスタのデータは無効データとなります。

0: パリティエラーなし

1: パリティエラー発生

【bit 14】ORE (Over Run Error)

受信時にオーバランエラーが発生したときにセットされる, 割込み要求フラグです。一度セットされたフラグをクリアするには, SCR1 レジスタの REC ビット (bit 10) に "0" を書き込みます。

本ビットがセットされた場合には, SISR1 レジスタのデータは無効データとなります。

0: オーバランエラーなし

1: オーバランエラー発生

【bit 13】FRE (Framing Error)

受信時にフレーミングエラーが発生したときにセットされる, 割込み要求フラグです。一度セットされたフラグをクリアするには, SCR1 レジスタの REC ビット (bit 10) に "0" を書き込みます。

本ビットがセットされた場合には, SISR1 レジスタのデータは無効データとなります。

0: フレーミングエラーなし

1: フレーミングエラー発生

【bit 12】RDRF (Receiver Data Register Full)

SIDR1 レジスタに受信データがあることを示す、割込み要求フラグです。

SIDR1 レジスタに受信データがロードされるとセットされ、SIDR1 レジスタからデータが読み出されると自動的にクリアされます。

0: 受信データなし

1: 受信データあり

【bit 11】TDRE (Transmitter Data Register Empty)

SODR1 レジスタに送信データを書き込めることを示す、割込み要求フラグです。

SODR1 レジスタに送信データを書き込むとクリアされます。書き込んだデータが送信用シフトにロードされて転送が開始されると再びセットされ、次の送信データを書き込めることを表します。

0: 送信データの書き込み禁止

1: 送信データの書き込み許可

【bit 9】RIE (Receiver Interrupt Enable)

受信割込みを制御します。

0: 割込みを禁止します。

1: 割込みを許可します。

< 注意事項 >

受信割込み要因は、PE, ORE, FRE によるエラー発生のほか、RDRF フラグによって示される正常受信があります。

【bit 8】TIE (Transmitter Interrupt Enable)

送信割込みを制御します。

0: 割込みを禁止します。

1: 割込みを許可します。

< 注意事項 >

送信割込み要因は、TDRE フラグによる送信要求も含まれます。

19.3.5 UART1 通信プリスケアラ制御レジスタ (CDCR)

UART1 通信プリスケアラ制御レジスタ (CDCR) は、マシンのクロックの分周を制御するレジスタです。UART1 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。この通信プリスケアラによって、種々のマシンのクロックに対して一定のボーレートが得られるように設計されています。

■ UART1 通信プリスケアラ制御レジスタ (CDCR)

UART1 通信プリスケアラ制御レジスタ (CDCR) のビット構成を図 19.3-7 に示します。

図 19.3-7 UART1 通信プリスケアラ制御レジスタ (CDCR)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 000028 _H	MD				DIV3	DIV2	DIV1	DIV0	CDCR
リード / ライト	(R/W)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(0)	(-)	(-)	(-)	(1)	(1)	(1)	(1)	

【bit 7】 MD (Machine clock divide MoDe select)

通信プリスケアラの動作許可ビットです。

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。

【bit 3 ~ bit 0】 DIV3 ~ DIV0 (DIVide 3 ~ 0)

表 19.3-3 に従い、マシンのクロックの分周比を決定します。

表 19.3-3 マシンのクロックの分周比

DIV3	DIV2	DIV1	DIV0	分周比 *
1	1	1	0	2 分周
1	1	0	1	3 分周
1	1	0	0	4 分周
1	0	1	1	5 分周
1	0	1	0	6 分周
1	0	0	1	7 分周
1	0	0	0	8 分周

*: 分周比を変えた場合はクロックの安定時間として 2 周期分の時間を置いてから通信を行ってください。

19.4 UART1 の動作モードとクロック選択

UART1 は、非同期モードと CLK 同期モードの 2 種類の動作モードを持ち、SMR1 レジスタ、SCR1 レジスタに値を設定することによりモードを切り換えることができます。

■ UART1 の動作モード

表 19.4-1 に、UART1 の動作モードを示します。

表 19.4-1 UART1 の動作モード

モード	パリティ	データ長	動作モード	ストップビット長*
0	有 / 無	7	非同期 (調歩同期) ノーマルモード	1 ビットまたは 2 ビット
	有 / 無	8		
1	無	8+1	非同期 (調歩同期) マルチプロセッサモード	無
2	無	8	CLK 同期モード	

※: 非同期 (調歩同期) ノーマルモードでのストップビット長については、送信動作のみ指定が可能です。受信動作については常に 1 ビット長となります。上記モード以外では動作しませんので、設定しないでください。

■ UART1 のクロック選択

● 専用ボーレートジェネレータ

専用ボーレートジェネレータを選択したときのボーレートを表 19.4-2 および表 19.4-3 に示します。また、通信プリスケール設定値を表 19.4-4 に示します。

なお、表中の ϕ はマシンのクロックを示します。

表 19.4-2 ボーレート (非同期時)

CS2	CS1	CS0	$\phi/\text{div} = 2 \text{ MHz}$	$\phi/\text{div} = 4 \text{ MHz}$	$\phi/\text{div} = 8 \text{ MHz}$	算出式
0	0	0	9615	19230	38460	$(\phi \div \text{div}) / (8 \times 13 \times 2)$
0	0	1	4808	9615	19230	$(\phi \div \text{div}) / (8 \times 13 \times 2^2)$
0	1	0	2404	4808	9615	$(\phi \div \text{div}) / (8 \times 13 \times 2^3)$
0	1	1	1202	2404	4808	$(\phi \div \text{div}) / (8 \times 13 \times 2^4)$
1	0	0	31250	62500	-	$(\phi \div \text{div}) / 2^6$

表 19.4-3 ボーレート (CLK 同期時)

CS2	CS1	CS0	$\phi/\text{div} = 2 \text{ MHz}$	$\phi/\text{div} = 4 \text{ MHz}$	$\phi/\text{div} = 8 \text{ MHz}$	算出式
0	0	0	1 MHz	-	-	$(\phi \div \text{div}) / 2$
0	0	1	500 kHz	1 MHz	-	$(\phi \div \text{div}) / 2^2$
0	1	0	250 kHz	500 kHz	1 MHz	$(\phi \div \text{div}) / 2^3$
0	1	1	125 kHz	250 kHz	500 kHz	$(\phi \div \text{div}) / 2^4$
1	0	0	62.5 kHz	125 kHz	250 kHz	$(\phi \div \text{div}) / 2^5$

表 19.4-4 通信プリスケラ設定値

推奨マシクロック ϕ	div	DIV3	DIV2	DIV1	DIV0	ϕ/div
4 MHz	4	1	1	0	0	1 MHz
6 MHz	6	1	0	1	0	
8 MHz	8	1	0	0	0	
6 MHz	3	1	1	0	1	2 MHz
8 MHz	4	1	1	0	0	
10 MHz	5	1	0	1	1	
12 MHz	6	1	0	1	0	
14 MHz	7	1	0	0	1	
16 MHz	8	1	0	0	0	
8 MHz	2	1	1	1	0	4 MHz
12 MHz	3	1	1	0	1	
16 MHz	4	1	1	0	0	
16 MHz	2	1	1	1	0	8 MHz

● 内部タイマ

CS2 ~ CS0 を "110_B" に設定して内部タイマを選択した場合は、リロードタイマ 0 (16 ビット動作時) をリロードモードで動作させます。このときのボーレートの算出式は、次のとおりです。

非同期 (調歩同期) $(\phi \div N)/(16 \times 2 \times (n+1))$

CLK 同期 $(\phi \div N)/(2 \times (n+1))$

N: タイマのカウントクロックソース

n: タイマのリロード値

マシクロックを 7.3728 MHz としたときのボーレートとリロード値 (10 進数) の関係を、表 19.4-5 に示します。

表 19.4-5 ボーレートとリロード値

リロード値 ボーレート	N=2 ¹ (マシクロックの 2 分周)	N=2 ³ (マシクロックの 8 分周)
38400	2	-
19200	5	-
9600	11	2
4800	23	5
2400	47	11
1200	95	23
600	191	47
300	383	95

ボーレートクロックソースとして内部タイマ (16 ビットタイマ 0) を選択した場合には、16 ビットタイマ 0 の出力信号 TOT0 は、既に MB90540/545 コントローラに内部的に接続されていることに注意してください。したがって、16 ビットタイマ 0 の外部出力端子 TOT0 から UART1 の外部クロック入力端子 SCK1 へ外部で接続する必要はありません。また、タイマ端子をほかに使用していなければ、I/O ポート端子として使用することができます。

● 外部クロック

CS2 ~ CS0 を "111_B" に設定して外部クロックを選択した場合のボーレートは、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。

非同期 (調歩同期) $f/16$

CLK 同期 f

f の最大値は 2 MHz であることに、注意してください。

19.4.1 非同期 (調歩同期) モード

UART1 は , NRZ(Non Return to Zero) 形式のデータのみを扱います。

■ 非同期 (調歩同期) モードの転送データフォーマット

図 19.4-1 に , 転送データフォーマットを示します。

図 19.4-1 非同期 (調歩同期) モードの転送データフォーマット (モード 0, 1)

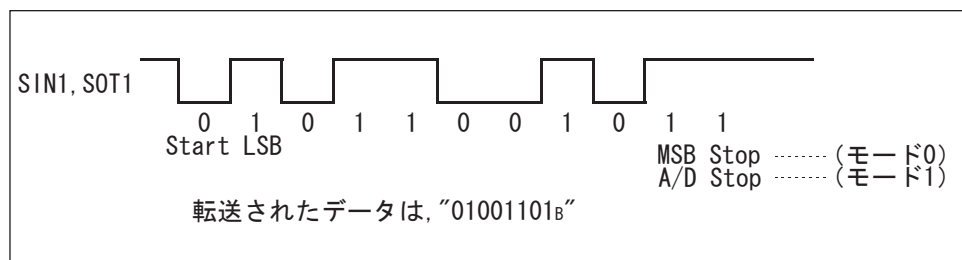


図 19.4-1 に示すように , 転送データは必ずスタートビット ("L" レベルデータ値) より始まり , LSB ファーストで指定されたデータビット長転送が行われ , ストップビット ("H" レベルデータ値) で終了します。外部クロックを選択している場合は , 常にクロックを入力してください。

ノーマルモード (モード 0) ではデータ長を 7 ビットまたは 8 ビットに設定することができますが , マルチプロセッサモード (モード 1) では 8 ビットでなければなりません。また , マルチプロセッサモードではパリティを付加することはできません。そのかわり , A/D ビットが必ず付加されます。

■ 非同期 (調歩同期) モードの受信動作

シリアル制御レジスタ 1 (SCR1) 内の RXE ビットが "1" ならば , 常に受信動作が行われています。

スタートビットを検出すると , SCR1 レジスタで決められたデータフォーマットに従い , 1 フレームデータの受信が行われます。1 フレームの受信が終わると , エラーが発生した場合にはエラーフラグのセットが行われた後 , シリアルステータスレジスタ (SSR1) 内の RDRF がセットされます。この場合 , 同じ SSR1 レジスタの RIE ビットが "1" にセットされていれば CPU に対して受信割込みが発生します。SSR1 レジスタの各フラグを調べ , 正常受信ならシリアルインプットデータレジスタ 1 (SIDR1) を読み出して , エラーが発生していれば , 必要なエラー処置を行うようにしてください。

RDRF フラグは , SIDR1 レジスタを読み出すとクリアされます。

■ 非同期 (調歩同期) モードの送信動作

シリアルステータスレジスタ 1(SSR1) 内の TDRE フラグが "1" の場合, シリアルアウト
プットデータレジスタ 1(SODR1) に送信データを書き込みます。ここで, シリアル制御
レジスタ 1(SCR1) 内の TXE ビットが "1" なら送信が行われます。

SODR1 レジスタにセットしたデータが送信用シフトレジスタにロードされて送信が開
始されると SSR1 レジスタ内の TDRE フラグが再びセットされ, 次の送信データをセッ
トできるようになります。この場合, 同じ SSR1 レジスタ内の TIE ビットが "1" にセッ
トされていれば CPU に対して送信割込みが発生して, SODR1 レジスタに送信データを
セットするように要求します。

SSR1 レジスタ内の TDRE フラグは, SODR1 レジスタにデータをセットするとクリア
されます。

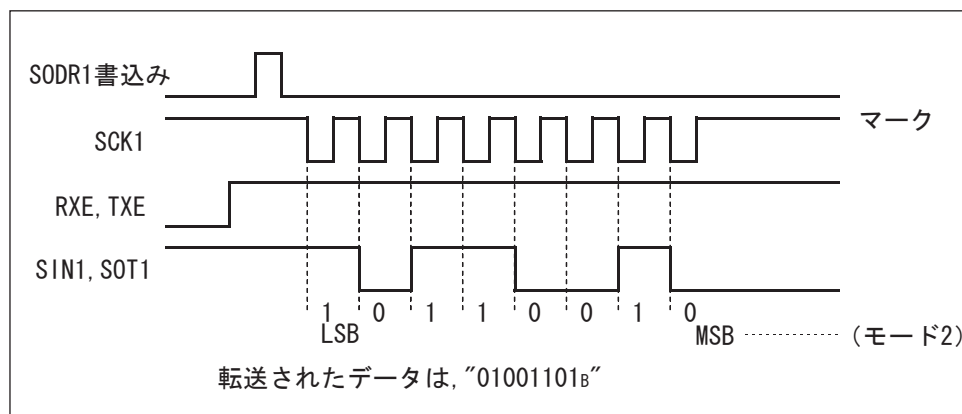
19.4.2 CLK 同期モード

UART1 は、NRZ(Non Return to Zero) 形式のデータのみを扱います。

■ CLK 同期モードの転送データフォーマット

図 19.4-2 に、CLK 同期モードにおける送受信クロックとデータとの関係を示します。

図 19.4-2 CLK 同期モードの転送データフォーマット (モード 2)



内部クロック (専用ボーレートジェネレータまたは内部タイマ) を選択している場合は、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、送信側 UART の送信用データバッファ SODR1 レジスタにデータがあること (TDRE フラグが "0") を確かめた後、正確に 1 バイト分のクロックを供給する必要があります。また、送信開始前と終了後は、必ずマークレベルにしてください。

データは 8 ビットのみとなり、パリティを付加することはできません。また、スタート / ストップビットがないのでオーバランエラー以外のエラー検出は行われません。

■ CLK 同期モードを使用する場合の各制御レジスタの設定値

CLK 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。

● SMR1 レジスタ

MD1, MD0 : "10"

CS2, CS1, CS0 : クロック入力を指定

SCKE : 専用ボーレートジェネレータまたは内部タイマの場合 "1"
外部クロックの場合 "0"

SOE : 送信を行う場合 "1", 受信のみの場合 "0"

● SCR1 レジスタ

PEN : "0"

P, SBL, A/D : これらのビットは意味を持ちません

CL : "1"

REC : "0" (初期化するため)

RXE, TXE : 少なくとも、どちらか一方を "1"

● SSR1 レジスタ

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 : "0"

■ CLK 同期モードの通信開始

SODR1 レジスタへの書込みによって通信を開始します。受信のみの場合でも、必ず仮の送信データを SODR1 レジスタに書き込む必要があります。

■ CLK 同期モードの通信終了

SSR1 レジスタの RDRF フラグが "1" に変化したことにより確認できます。SSR1 レジスタの ORE ビットによって、通信が正常に行われたかを判断してください。

19.5 UART1 のフラグと割込み発生要因

UART1 は PE/ORE/FRE/RDRF/TDRE の 5 つのフラグを持ちます。また、割込み要因には受信用のものと送信用のものがあります。

■ UART1 のフラグ

UART は、PE/ORE/FRE/RDRF/TDRE の 5 つのフラグを持ちます。はじめの 3 つは、送信エラーが生じたときにセットされ、PE フラグはパリティエラー、ORE フラグはオーバランエラー、FRE フラグはフレーミングエラーのことで、SCR1 レジスタの REC に "0" を書き込むとクリアされます。RDRF は受信データが SIDR1 レジスタにロードされるとセットされ、SIDR1 レジスタを読み出すことでクリアされます。ただし、モード 1 ではパリティ検出機能、モード 2 ではパリティ検出機能とフレーミングエラー検出機能はありません。TDRE は、SODR1 レジスタが空になり、書き込み可能な状態になるとセットされ、SODR1 レジスタへ書き込むとクリアされます。

■ UART1 の割込み発生要因

UART1 の割込み発生要因には、受信用のものと送信用のものがあります。受信時は、PE/ORE/FRE/RDRF により割込みを要求します。送信時は、TDRE により割込みを要求します。

各動作モード時の割込みとフラグのセットタイミングについては、「19.6 UART1 の割込みとフラグのセットタイミング」を参照してください。

19.6 UART1 の割込みとフラグのセットタイミング

各 UART1 動作モード時の割込みとフラグのセットタイミングを示します。

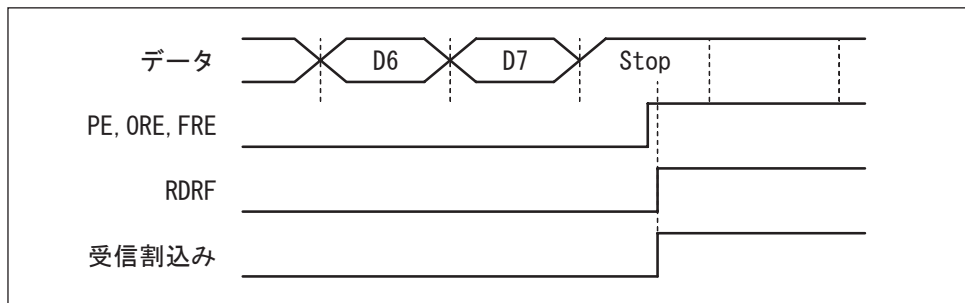
■ UART1 の割込みとフラグのセットタイミング

● モード 0 の受信動作時

PE, ORE, FRE, RDRF は受信転送が終了し、最後のストップビットを検出するときにフラグがセットされ、CPU への割込み要求が発生します。PE, ORE, FRE がアクティブ時には、SIDR1 レジスタのデータは無効データとなります。

図 19.6-1 に、PE, ORE, FRE, RDRF のセットタイミング (モード 0) を示します。

図 19.6-1 PE, ORE, FRE, RDRF のセットタイミング (モード 0)

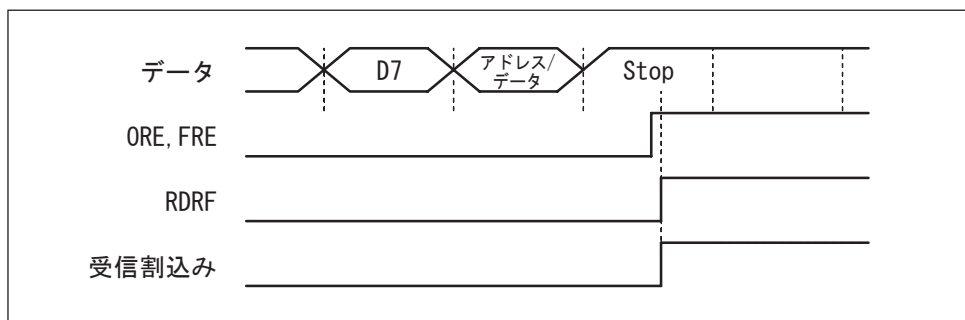


● モード 1 の受信動作時

ORE, FRE, RDRF は受信転送が終了し、最後のストップビットを検出するときにフラグがセットされ、CPU への割込み要求が発生します。また、受信可能なデータ長は 8 ビットのため、最後の 9 ビット目のアドレス / データを示すデータは無効データとなります。ORE, FRE がアクティブ時は、SIDR1 のデータは無効データとなります。

図 19.6-2 に、ORE, FRE, RDRF のセットタイミング (モード 1) を示します。

図 19.6-2 ORE, FRE, RDRF のセットタイミング (モード 1)

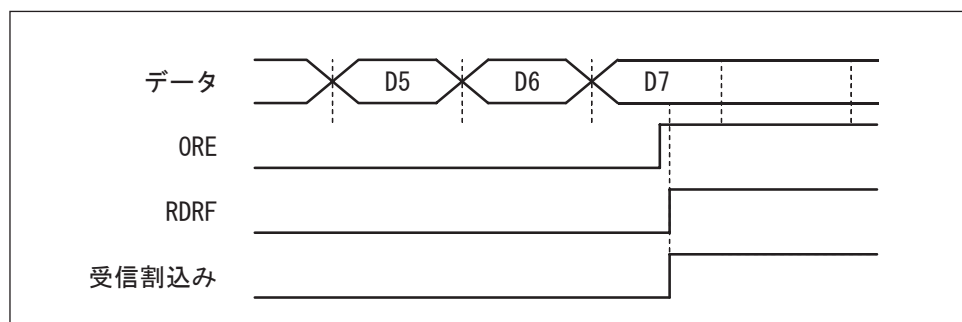


● モード 2 の受信動作時

ORE, RDRF は受信転送が終了し、最後のデータ (D7) を検出するときにフラグがセットされ、CPU への割込み要求が発生します。ORE がアクティブ時は、SIDR1 のデータは無効データとなります。

図 19.6-3 に、OR, RDRF のセットタイミング (モード 2) を示します。

図 19.6-3 ORE, RDRF のセットタイミング (モード 2)



● モード 0, モード 1, モード 2 の送信動作時

TDRE フラグは、SODR1 レジスタへ送信データが書き込まれるとクリアされます。また、SODR1 レジスタ値が内部シフトレジスタに転送されると SODR1 レジスタが書き込み可能な状態になるため、TDRE フラグがセットされます。このフラグがセットされると CPU への割込み要求が発生します。送信動作中に SCR1 レジスタの TXE に "0" (モード 2 のときは RXE も含む) を書き込むと、SSR1 レジスタの TDRE が "1" となり、送信用のシフトレジスタが停止して UART1 の送信動作を禁止します。送信動作中に SCR1 レジスタの TXE に "0" (モード 2 のときは RXE も含む) を書き込んだ後、送信が停止する前に SODR1 レジスタへ書き込まれたデータは送信されます。図 19.6-4 に、TDRE のセットタイミング (モード 0, 1) を示します。また、図 19.6-5 に、TDRE のセットタイミング (モード 2) を示します。

図 19.6-4 TDRE のセットタイミング (モード 0, 1)

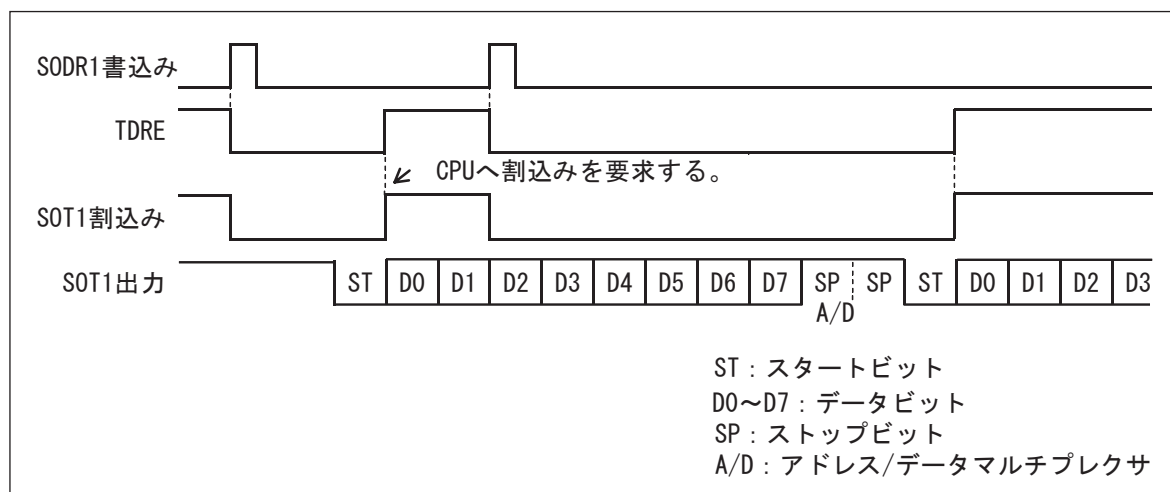
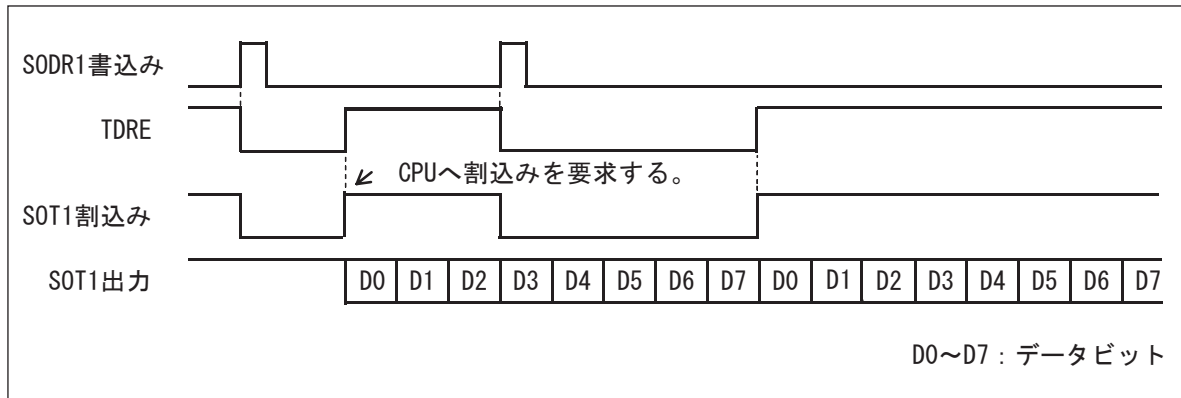


図 19.6-5 TDRE のセットタイミング (モード 2)



19.7 ネガティブクロック動作

MB90540/545 シリーズは、UART1 のネガティブクロック動作をサポートします。この動作では、シフトクロック信号はインバータにより単純に反転されます。このため、UART1 で進行中のセクションにおけるシフトクロック信号の定義は、ロジック "L" レベルからロジック "H" レベルへ、ネガティブエッジからポジティブエッジへ、あるいはこの逆に反転されます。このことは、シリアルクロックの入力および出力とも同じです。エッジセクタレジスタは、この目的のために用意されています。

■ ネガティブクロック動作

シリアルエッジセレクトレジスタ (SES1) のビット構成を図 19.7-1 に示します。

図 19.7-1 シリアルエッジセレクトレジスタ (SES1)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 000029 _H								NEG	SES1
リード / ライト	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(R/W)	
初期値	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(0)	

表 19.7-1 NEG ビットの設定

NEG	操作
0	通常動作 (デフォルト)
1	シフトクロック信号反転

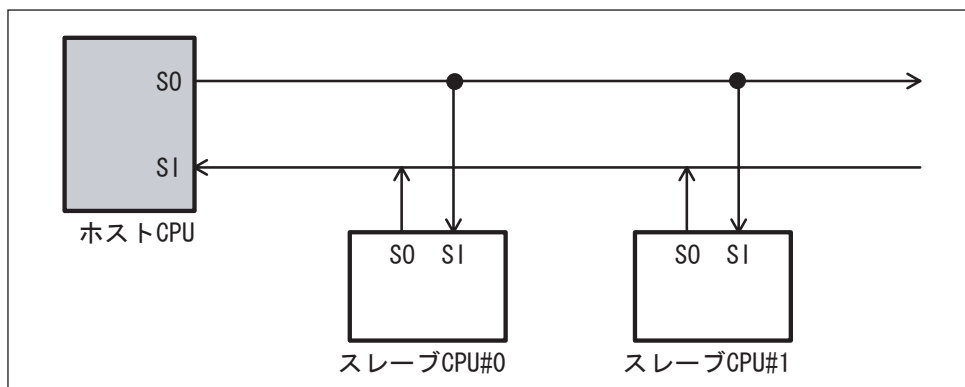
19.8 UART1 の応用例と使用上の注意

UART1 の応用例として、モード 1 使用時のシステム構築例と通信フローチャートを示します。

■ UART1 の応用例 (モード 1 使用時のシステム構築例)

図 19.8-1 に示すように、モード 1 は、1 台のホスト CPU に数台のスレーブ CPU が接続されるような場合に使用されます。この UART1 リソースでは、ホスト側の通信インターフェースのみサポートします。

図 19.8-1 モード 1 使用時のシステム構築例

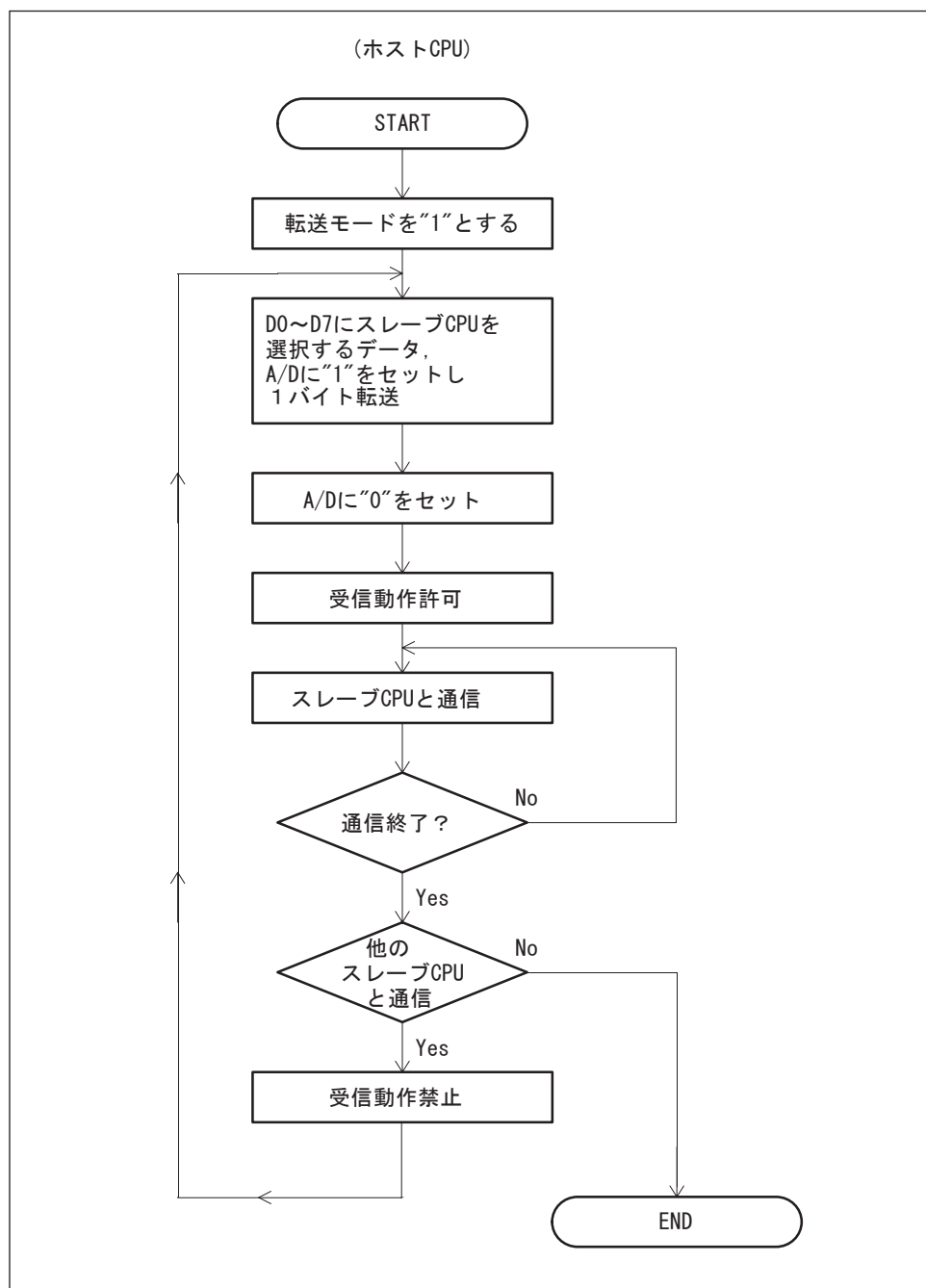


■ UART1 の通信フローチャート

通信はホスト CPU がアドレスデータを転送することによって始まります。アドレスデータとは、SCR1 レジスタの A/D が "1" のときのデータで、それにより通信先となるスレーブ CPU が選択され、ホスト CPU との通信が可能になります。通常データは、SCR1 レジスタの A/D が "0" のときのデータです。図 19.8-2 にそのフローチャートを示します。

モード 2 においては、パリティチェック機能は使用できませんので SCR1 レジスタの PEN ビットは "0" としてください。

図 19.8-2 モード 1 使用時の通信フローチャート



■ 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 (EI²OS) の詳細については、「3.6 拡張インテリジェント I/O サービス (EI²OS)」を参照してください。

■ UART1 の使用上の注意

通信モードの設定は、UART1 の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。

第20章

シリアル I/O

シリアル I/O の機能と動作について示します。

- 20.1 シリアル I/O の概要
- 20.2 シリアル I/O のレジスタ
- 20.3 シリアル I/O の動作
- 20.4 ネガティブクロック動作

20.1 シリアル I/O の概要

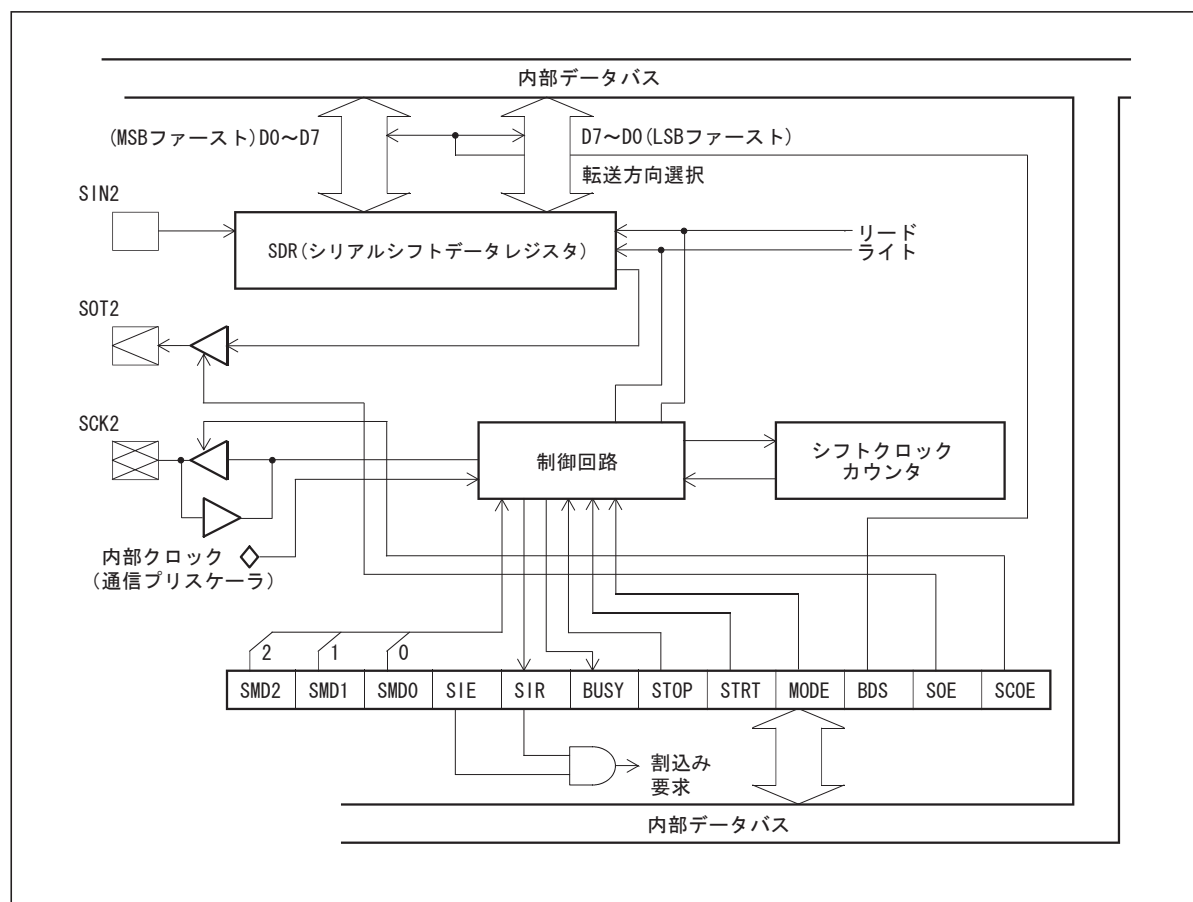
シリアル I/O の動作モードには、以下の 2 種類があります。

- 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 (SCK2) 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 (SCK2) 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。

■ シリアル I/O のブロックダイアグラム

クロック同期によるデータ転送を可能にするシリアル I/O インタフェースのブロックダイアグラムを示します。このインタフェースは 8 ビット×1 チャンネル構成で、データは LSB/MSB から転送できます。

図 20.1-1 シリアル I/O のブロックダイアグラム



20.2 シリアル I/O のレジスタ

シリアル I/O のレジスタには、次の 3 つがあります。

- シリアルモード制御ステータスレジスタ
- シリアルシフトデータレジスタ
- シリアル I/O プリスケアラ

■ シリアル I/O のレジスタ

シリアル I/O のレジスタのビット構成を図 20.2-1 に示します。

図 20.2-1 シリアル I/O のレジスタ

シリアルモード制御ステータスレジスタ									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 00002D _H	SMD2	SMD1	SMD0	SIE	SIR	BUSY	STOP	STRT	SMCS
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	
シリアルシフトデータレジスタ									
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00002C _H	-	-	-	-	MODE	BDS	SOE	SCOE	SMCS
リード / ライト	(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(-)	(-)	(-)	(-)	(0)	(0)	(0)	(0)	
シリアルシフトデータレジスタ									
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00002E _H	D7	D6	D5	D4	D3	D2	D1	D0	SDR
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
シリアル I/O プリスケアラ									
	15	14	13	12	11	10	9	8	ビット No.
アドレス : 00002B _H	MD	-	-	-	DIV3	DIV2	DIV1	DIV0	SCDCR
リード / ライト	(R/W)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(0)	(-)	(-)	(-)	(1)	(1)	(1)	(1)	

20.2.1 シリアルモード制御ステータスレジスタ (SMCS)

シリアルモード制御ステータスレジスタ (SMCS) は、シリアル I/O 転送モードを制御するレジスタです。

■ シリアルモード制御ステータスレジスタ (SMCS)

シリアルモード制御ステータスレジスタ (SMCS) のビット構成を図 20.2-2 に示します。

図 20.2-2 シリアルモード制御ステータスレジスタ (SMCS)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 00002D _H	SMD2	SMD1	SMD0	SIE	SIR	BUSY	STOP	STRT	SMCS
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(1)	(0)	
						*1		*2	
	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00002C _H	-	-	-	-	MODE	BDS	SOE	SCOE	SMCS
リード / ライト	(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(-)	(-)	(-)	(-)	(0)	(0)	(0)	(0)	
*1 : 書込み動作は、"0" のみ可能									
*2 : 書込み動作は、"1" のみ可能、読出し動作は常に "0"									

【bit 3】(MODE) シリアルモード選択ビット

このビットにより停止状態からの起動条件の選択を行います。ただし、動作中の書換えは禁止します。

表 20.2-1 シリアルモード選択ビットの設定

MODE	動作
0	STRT=1 にすることにより起動します [初期値]。
1	シリアルシフトデータレジスタのリード / ライトにより起動します。

リセットにより "0" に初期化されます。本ビットは、リード / ライトが可能です。インテリジェント I/O サービスを起動する際には "1" にしておいてください。

【bit 2】BDS:Bit Direction Select (転送方向選択ビット)

シリアルデータの入出力時に、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、表 20.2-2 に示すとおりを選択します。

表 20.2-2 転送方向選択ビットの設定

BDS	動作
0	LSB ファースト [初期値]
1	MSB ファースト

< 注意事項 >

転送方向選択ビットは、SDR にデータを書き込む前に、設定しておいてください。

【bit 1】SOE:Serial Out Enable (シリアル出力許可ビット)

シリアル I/O 用出力外部端子 (SOT2) の出力を表 20.2-3 に示すとおりに制御します。

表 20.2-3 シリアル出力許可ビットの設定

SOE	動作
0	汎用ポート端子 [初期値]
1	シリアルデータ出力

リセットにより "0" に初期化されます。本ビットは、リード / ライトが可能です。

【bit 0】SCOE:SCK2 Output Enable (シフトクロック出力許可ビット)

シフトクロック用入出力外部端子 (SCK2) の出力を表 20.2-4 に示すとおりに制御します。

表 20.2-4 シフトクロック出力許可ビットの設定

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

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。リセットにより "0" に初期化されます。本ビットは、リード / ライトが可能です。

【bit 15 ~ bit 13】SMD2, SMD1, SMD0:Serial Shift clock MoDe
(シフトクロック選択ビット)

シリアルシフトクロックモードを表 20.2-5 に示すとおりに選択します。

表 20.2-5 シリアルシフトクロックモードの設定

SM D2	SM D1	SM D0	φ/div = 4 MHz	φ/div = 2 MHz	φ/div = 1 MHz
0	0	0	2 MHz	1 MHz	500 kHz
0	0	1	1 MHz	500 kHz	250 kHz
0	1	0	250 kHz	125 kHz	62.5 kHz
0	1	1	125 kHz	62.5 kHz	31.25 kHz
1	0	0	62.5 kHz	31.25 kHz	15.625 kHz
1	0	1	外部シフトクロックモード		
1	1	0	予約		
1	1	1	予約		

div	M1	DIV3	DIV2	DIV1	DIV0	推奨マシン サイクル
3	1	1	1	0	1	6 MHz
4	1	1	1	0	0	8 MHz
5	1	1	0	1	1	10 MHz
6	1	1	0	1	0	12 MHz
7	1	1	0	0	1	14 MHz
8	1	1	0	0	0	16 MHz

各ビットの詳細については、「20.2.3 シリアル I/O ブリスケーラ (SCDCR)」を参照してください。

リセットにより "000_B" に初期化されます。本ビットの転送中の書換えは禁止します。
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。
SMD2, SMD1, SMD0 = "110_B", "111_B" はリザーブですので設定しないでください。

外部シフトクロックモードが選択されている場合には、シフトクロック入力を共有して
いる汎用 I/O のアウトプットレベルを変えることで、ビットシフト操作を行うこともで
きます。

【bit 12】SIE:Serial I/O Interrupt Enable (シリアル I/O 割込み許可ビット)

シリアル I/O の割込み要求を表 20.2-6 に示すとおりに制御します。

表 20.2-6 割込み要求許可ビットの設定

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

リセットにより "0" に初期化されます。本ビットは、リード / ライトが可能です。

【bit 11】SIR:Serial I/O Interrupt Request (シリアル I/O 割込み要求ビット)

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 へ割込み要求を発生します。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" 書込みによりク
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りク
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"
書込み動作によりクリアされます。

本ビットに "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系命令
のリード時は、常に "1" が読み出されます。

【bit 10】BUSY (転送状態ビット)

シリアル転送が実行中か否かを示すビットです。

表 20.2-7 転送状態ビットの設定

BUSY	動作
0	停止またはシリアルシフトデータレジスタ R/W 待機状態 [初期値]
1	シリアル転送状態

リセットにより "0" に初期化されます。本ビットは、リードのみ可能です。

【bit 9】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。本ビットを "1" にすると STOP = 1 による停止状態となります。

表 20.2-8 ストップビットの設定

STOP	動作
0	動作
1	STOP=1 による転送停止 [初期値]

リセットにより "1" に初期化されます。本ビットは、リード / ライトが可能です。

【bit 8】STRT:Start (スタートビット)

シリアル転送の起動を行うビットです。停止状態で "1" を書く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データレジスタ R/W 待機中の "1" の書込みは無視され、"0" の書込みは意味がありません。

リード時には常に "0" が読み出されます。

20.2.2 シリアルシフトデータレジスタ (SDR)

シリアルシフトデータレジスタ (SDR) は , シリアル I/O 転送データを格納するシリアルシフトデータレジスタで , 転送中は読み書き禁止です。

■ シリアルシフトデータレジスタ (SDR)

シリアルシフトデータレジスタ (SDR) のビット構成を図 20.2-3 に示します。

図 20.2-3 シリアルシフトデータレジスタ (SDR)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00002E _H	D7	D6	D5	D4	D3	D2	D1	D0	SDR
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

20.2.3 シリアル I/O プリスケーラ (SCDCR)

シリアル I/O プリスケーラ (SCDCR) は, シリアル I/O のシフトクロックを提供します。
シリアル I/O の動作クロックは, マシンクロックを分周することにより得られます。
シリアル I/O は, この通信プリスケーラによって種々のマシンクロックに対して一定のボーレートが得られるように設計されています。SCDCR レジスタは, マシンクロックの分周を制御するレジスタです。

■ シリアル I/O プリスケーラ (SCDCR)

シリアル I/O プリスケーラ (SCDCR) のビット構成を図 20.2-4 に示します。

図 20.2-4 シリアル I/O プリスケーラ (SCDCR)

	15	14	13	12	11	10	9	8	ビット No.
アドレス : 00002B _H	MD	-	-	-	DIV3	DIV2	DIV1	DIV0	SCDCR
リード / ライト	(R/W)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(0)	(-)	(-)	(-)	(1)	(1)	(1)	(1)	

【bit 15】 MD (Machine clock devide moDe select)

通信プリスケーラの動作許可ビットです。

0: シリアル I/O プリスケーラは停止します。

1: シリアル I/O プリスケーラは動作します。

【bit 11 ~ bit 8】 DIV3 ~ DIV0(DIVide 3 ~ DIVide 0)

マシンクロックの分周比を決定するビットです。

表 20.2-9 マシンクロックの分周比

DIV3 ~ DIV0	分周比
1101 _B	3
1100 _B	4
1011 _B	5
1010 _B	6
1001 _B	7
1000 _B	8

< 注意事項 >

分周比を変えた場合はクロックの安定時間として2周期分の時間を置いてから通信を行ってください。

20.3 シリアル I/O の動作

シリアル I/O は、シリアルモード制御ステータスレジスタ (SMCS)、シリアルシフトデータレジスタ (SDR) により構成されます。8 ビットのシリアルデータの入出力に使用します。

■ シリアル I/O の動作

シリアルデータの入出力は、シフト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 (SOT2 端子) に出力され、立上りに同期してシリアル入力端子 (SIN2 端子) からビット直列にシリアルシフトデータレジスタ (SDR) 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、シリアルモード制御ステータスレジスタ (SMCS) の方向指定ビット (BDS) により指定することができます。

転送が終了するとシリアルモード制御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。

- 停止状態からの復帰のときは 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)。
- シリアルシフトデータレジスタ R/W 待機状態からの復帰時はデータレジスタの読出しまたは書込みを行います。

20.3.1 シフトクロック

シフトクロックには内部シフトクロックモード、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で可能です。

■ 内部シフトクロックモード

内部シフトクロックモードは、内部クロックにより動作し、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 SCK2 端子より出力可能です。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(ボーレート)は以下ようになります。

$$\text{ボーレート} = \frac{\phi \div \text{div}}{A}$$

A は、SMCS の SMD ビットで示す分周比で、 $2^1, 2^2, 2^4, 2^5, 2^6$ です。

表 20.3-1 内部シフトクロックモードのボーレート算出式

SMD2	SMD1	SMD0	$\phi/\text{div} = 4 \text{ MHz}$	$\phi/\text{div} = 2 \text{ MHz}$	$\phi/\text{div} = 1 \text{ MHz}$	ボーレート算出式
0	0	0	2 MHz	1 MHz	500 kHz	$(\phi \div \text{div})/2^1$
0	0	1	1 MHz	500 kHz	250 kHz	$(\phi \div \text{div})/2^2$
0	1	0	250 kHz	125 kHz	62.5 kHz	$(\phi \div \text{div})/2^4$
0	1	1	125 kHz	62.5 kHz	31.25 kHz	$(\phi \div \text{div})/2^5$
1	0	0	62.5 kHz	31.2 kHz	15.625 kHz	$(\phi \div \text{div})/2^6$

Div の値は、表 19.4-4 を参照してください。

■ 外部シフトクロックモード

SCK2 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 $1/(8 \text{ マシンサイクル})$ 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、次のような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。さらに、SCK2 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、SCK2 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。シフトクロックの開始は "H" からスタートさせてください。

< 注意事項 >

シリアル I/O 動作中の SMCS, SDR への書き込みは禁止です。

20.3.2 シリアル I/O の動作状態

シリアル I/O の動作状態には、次の 4 種類があります。

- STOP 状態
 - 停止状態
 - SDR の R/W 待機状態
 - 転送状態
-

■ 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。

● シリアルシフトデータレジスタ 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 となりシリアルシフトデータレジスタ 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力します。

R/W 待機状態からの復帰はシリアルシフト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、動作遷移を示します。

図 20.3-1 シリアル I/O 動作遷移図

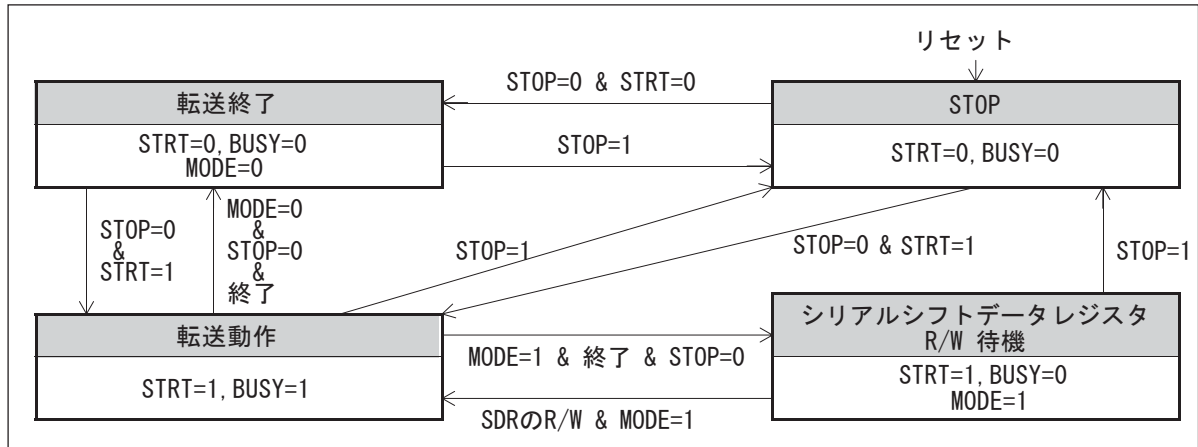
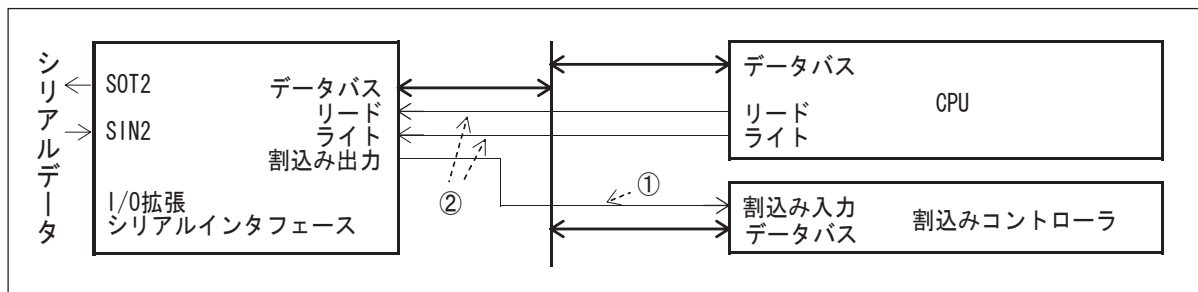


図 20.3-2 シリアルシフトデータレジスタへのリード、ライト概念図



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 となってリード/ライト待機状態に入ります。SIE ビットが "1" ならば割り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り込み信号は発生しません。

シリアルシフトデータレジスタが読み出し/書き込みされると、割り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

シフト動作をスタートする場合は、SMCS の STOP ビットを "0", STRT ビットを "1" に設定します。シフト動作のストップタイミングについては、STOP=1 によって停止する場合と、転送終了によって停止する場合があります。

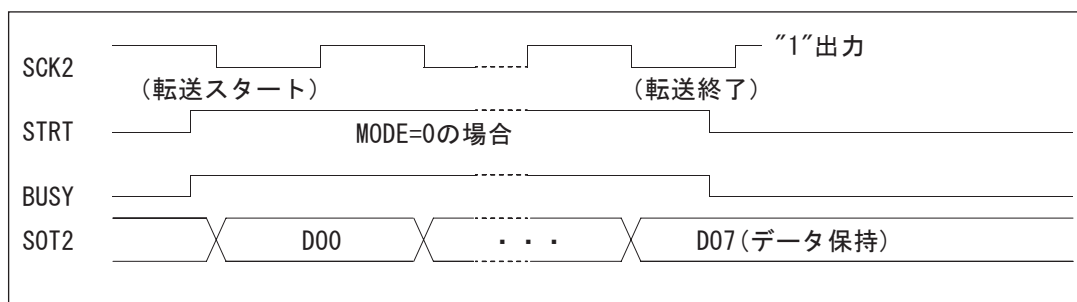
- STOP=1 によって停止 MODE ビットにかかわらず SIR=0 のまま停止
- 転送終了で停止 MODE ビットにかかわらず SIR=1 となり停止

BUSY ビットは MODE ビットにかかわらずシリアル転送状態のとき "1" となり、停止状態、または R/W 待機状態のとき "0" となります。転送状態を確認したいときはこのビットを読み出してください。

■ シフト動作のスタート/ストップタイミング

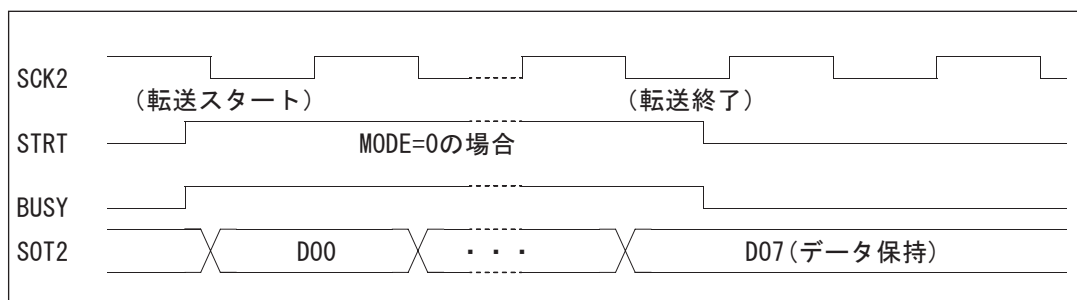
- 内部シフトクロックモード (LSB ファースト)

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)



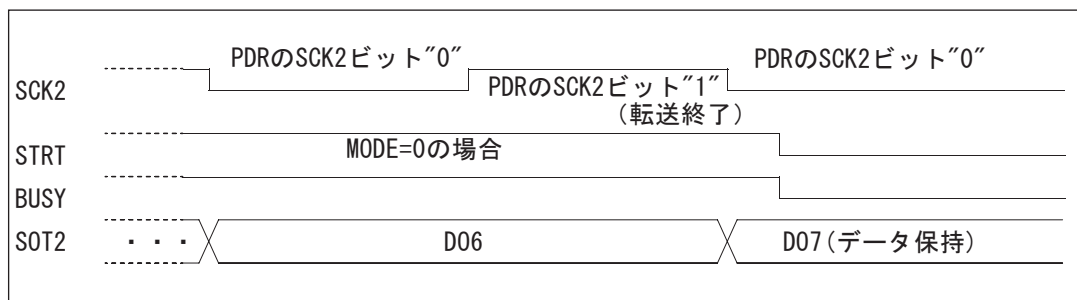
- 外部シフトクロックモード (LSB ファースト)

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)

図 20.3-5 シフト動作のスタート/ストップタイミング
(外部シフトクロックモードで命令シフト実行時)

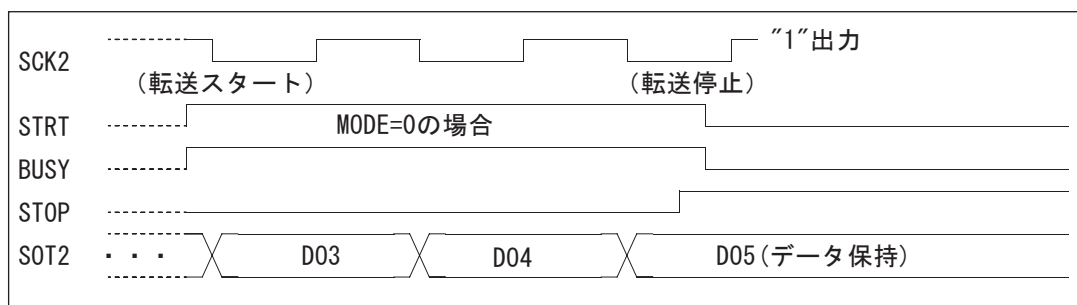


< 注意事項 >

命令シフトでは, PDR の SCK2 に対応するビットに "1" が書き込まれた場合 "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



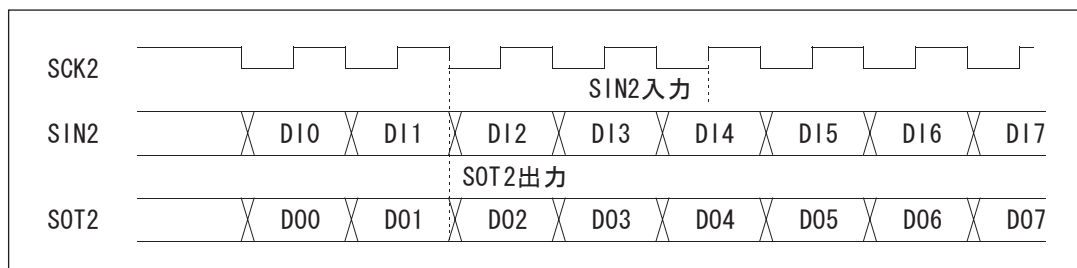
< 注意事項 >

DO7 ~ DO0 は出力データを示します。

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 (SOT2) からのデータが出力され, 立上りでシリアル入力端子 (SIN2) のデータが入力されます。

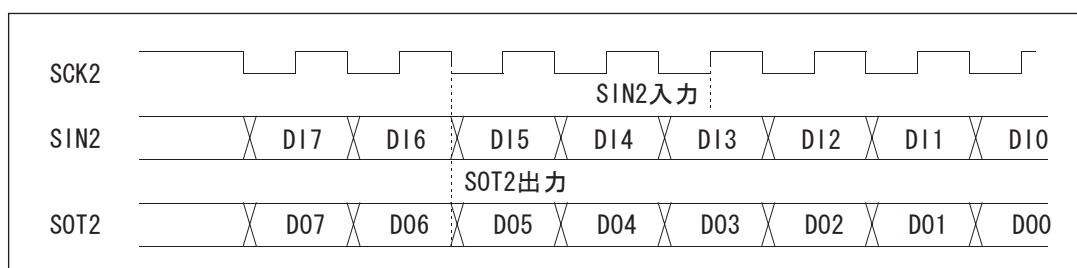
- LSB ファースト (BDS ビットが "0" の場合)

図 20.3-7 シリアルデータの入出力シフトタイミング 1



- MSB ファースト (BDS ビットが "1" の場合)

図 20.3-8 シリアルデータの入出力シフトタイミング 2

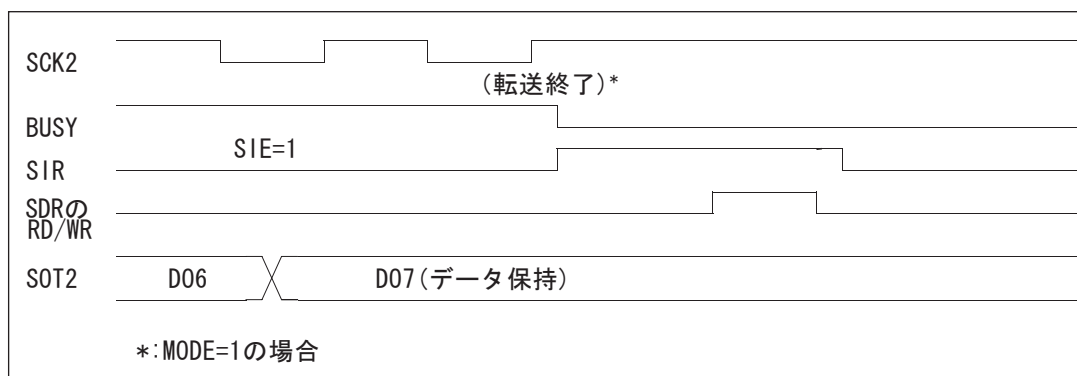


20.3.4 シリアル I/O の割込み機能

シリアル I/O は、CPU に対し割込み要求を発生することができます。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。

■ シリアル I/O の割込み機能

図 20.3-9 シリアル I/O の割込み信号出力タイミング



20.4 ネガティブクロック動作

MB90540/545 シリーズは、シリアル I/O のネガティブクロック動作をサポートします。この動作では、シフトクロック信号はインバータにより単純に反転されます。このため、前述のシリアル I/O のシフトクロック信号の定義は、ロジック "L" レベルからロジック "H" レベルへ、ネガティブエッジからポジティブエッジへ、あるいはこの逆に反転されます。このことは、シリアルクロックの入力および出力ともに同じです。エッジセクタレジスタは、この目的のために用意されています。

■ ネガティブクロック動作

シリアルエッジセレクトレジスタ (SES2) のビット構成を図 20.4-1 に示します。

図 20.4-1 シリアルエッジセレクトレジスタ (SES2)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 00002F _H								NEG	SES2
リード / ライト	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(R/W)	
初期値	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(0)	

表 20.4-1 NEG ビットの設定

NEG	動作
0	通常動作 (デフォルト)
1	シフトクロック信号反転

第21章

CAN コントローラ

CAN コントローラの機能と動作について説明します。

- 21.1 CAN コントローラの特長
- 21.2 CAN コントローラのブロックダイアグラム
- 21.3 全体制御レジスタの一覧
- 21.4 メッセージバッファ (ID レジスタ) の一覧
- 21.5 メッセージバッファ (DLC レジスタ / データレジスタ) の一覧
- 21.6 CAN コントローラのレジスタ分類
- 21.7 CAN コントローラの送信
- 21.8 CAN コントローラの受信
- 21.9 CAN コントローラの受信フローチャート
- 21.10 CAN コントローラの使用方法
- 21.11 メッセージバッファ (x) による送信方法
- 21.12 メッセージバッファ (x) による受信方法
- 21.13 マルチレベルメッセージバッファの構成の決定
- 21.14 CAN コントローラの使用上の注意

21.1 CAN コントローラの特長

MB90540 シリーズには、2 つの CAN コントローラ (CAN0, CAN1), MB90545 シリーズには 1 つの CAN コントローラ (CAN0) が装備されています。MB90V540 エバリュエーションチップにも、2 つの CAN コントローラが装備されています。CAN コントローラは、16 ビットマイクロコンピュータ (F²MC-16LX) 内に一体化されたモジュールです。CAN(コントローラエリアネットワーク) は自動車用コントローラ間のシリアル通信用の標準プロトコルであり、工業分野で広く使用されているものです。

■ CAN コントローラの特長

CAN コントローラの特長は次のとおりです。

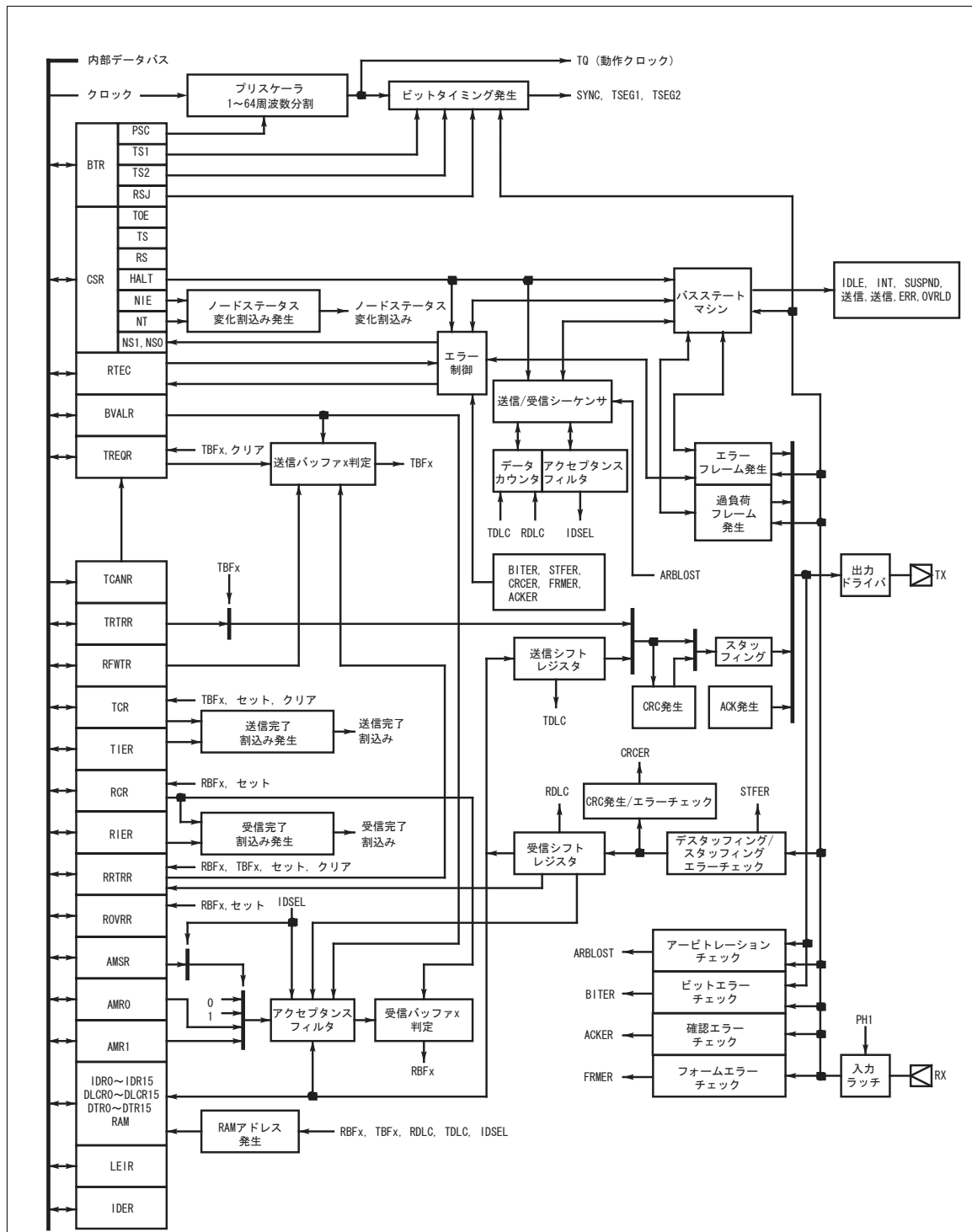
- CAN 仕様バージョン 2.0 パート A および B に適合しています。
標準フレームおよび拡張フレームフォーマットでの送信 / 受信をサポートしています。
- リモートフレームの受信によるデータフレームの送信をサポートしています。
- 16 個の送信 / 受信メッセージバッファ
 - 29 ビットの ID および 8 バイトデータ
 - マルチ・レベル・メッセージ・バッファ構造
- フルビット比較、フルビットマスク、ID アクセプタンスマスクとして、アクセプタンスレジスタ 0/ アクセプタンスレジスタ 1 を各メッセージバッファに提供しています。
標準フレームまたは拡張フレームフォーマットのいずれかで 2 個のアクセプタンスマスクレジスタを提供します。
- ビット速度は 10K ビット /s ~ 1M ビット /s でプログラマブルです (1M ビット /s でご使用の場合、マシクロックは 8 MHz 以上で動作する必要があります)。

21.2 CAN コントローラのブロックダイアグラム

図 21.2-1 に、CAN コントローラのブロックダイアグラムを示します。

■ CAN コントローラのブロックダイアグラム

図 21.2-1 CAN コントローラのブロックダイアグラム



21.3 全体制御レジスタの一覧

表 21.3-1 に、全体制御レジスタの一覧を示します。

■ 全体制御レジスタ

表 21.3-1 全体制御レジスタの一覧 (1 / 2)

アドレス		レジスタ	略号	アクセス	初期値
CAN0	CAN1				
000070 _H	000080 _H	メッセージバッファ有効レジスタ	BVALR	R/W	00000000 00000000
000071 _H	000081 _H				
000072 _H	000082 _H	送信要求レジスタ	TREQR	R/W	00000000 00000000
000073 _H	000083 _H				
000074 _H	000084 _H	送信解除レジスタ	TCANR	W	00000000 00000000
000075 _H	000085 _H				
000076 _H	000086 _H	送信完了レジスタ	TCR	R/W	00000000 00000000
000077 _H	000087 _H				
000078 _H	000088 _H	受信完了レジスタ	RCR	R/W	00000000 00000000
000079 _H	000089 _H				
00007A _H	00008A _H	リモート要求受信レジスタ	RRTRR	R/W	00000000 00000000
00007B _H	00008B _H				
00007C _H	00008C _H	受信オーバランレジスタ	ROVRR	R/W	00000000 00000000
00007D _H	00008D _H				
00007E _H	00008E _H	受信割込み許可レジスタ	RIER	R/W	00000000 00000000
00007F _H	00008F _H				
003B00 _H	003D00 _H	制御ステータスレジスタ	CSR	R/W, R	00---000 0---001
003B01 _H	003D01 _H				
003B02 _H	003D02 _H	ラストイベント表示レジスタ	LEIR	R/W	----- 000-0000
003B03 _H	003D03 _H				
003B04 _H	003D04 _H	受信および送信エラーカウンタ	RTEC	R	00000000 00000000
003B05 _H	003D05 _H				
003B06 _H	003D06 _H	ビットタイミングレジスタ	BTR	R/W	-1111111 11111111
003B07 _H	003D07 _H				
003B08 _H	003D08 _H	IDE レジスタ	IDER	R/W	XXXXXXXX XXXXXXXX
003B09 _H	003D09 _H				
003B0A _H	003D0A _H	送信 RTR レジスタ	TRTRR	R/W	00000000 00000000
003B0B _H	003D0B _H				
003B0C _H	003D0C _H	リモートフレーム受信待機レジスタ	RFWTR	R/W	XXXXXXXX XXXXXXXX
003B0D _H	003D0D _H				
003B0E _H	003D0E _H	送信割込み許可レジスタ	TIER	R/W	00000000 00000000
003B0F _H	003D0F _H				
003B10 _H	003D10 _H	アクセプタンスマスク選択レジスタ	AMSR	R/W	XXXXXXXX XXXXXXXX
003B11 _H	003D11 _H				XXXXXXXX XXXXXXXX
003B12 _H	003D12 _H				
003B13 _H	003D13 _H				

表 21.3-1 全体制御レジスタの一覧 (2 / 2)

アドレス		レジスタ	略号	アクセス	初期値
CAN0	CAN1				
003B14 _H	003D14 _H	アクセプタンスマスクレジスタ 0	AMR0	R/W	XXXXXXXX XXXXXXXX
003B15 _H	003D15 _H				XXXXXX--- XXXXXXXX
003B16 _H	003D16 _H				
003B17 _H	003D17 _H				
003B18 _H	003D18 _H	アクセプタンスマスクレジスタ 1	AMR1	R/W	XXXXXXXX XXXXXXXX
003B19 _H	003D19 _H				XXXXXX--- XXXXXXXX
003B1A _H	003D1A _H				
003B1B _H	003D1B _H				

21.4 メッセージバッファ (ID レジスタ) の一覧

表 21.4-1 に、メッセージバッファ (ID レジスタ) の一覧を示します。

■ メッセージバッファ (ID レジスタ) の一覧

表 21.4-1 メッセージバッファ (ID レジスタ) の一覧 (1 / 2)

アドレス		レジスタ	略号	アクセス	初期値
CAN0	CAN1				
003A00 _H ~ 003A1F _H	003C00 _H ~ 003C1F _H	汎用 RAM	--	R/W	XXXXXXXX ~ XXXXXXXX
003A20 _H	003C20 _H	ID レジスタ 0	IDR0	R/W	XXXXXXXX XXXXXXXX
003A21 _H	003C21 _H				XXXXXX--- XXXXXXXX
003A22 _H	003C22 _H				
003A23 _H	003C23 _H				
003A24 _H	003C24 _H	ID レジスタ 1	IDR1	R/W	XXXXXXXX XXXXXXXX
003A25 _H	003C25 _H				XXXXXX--- XXXXXXXX
003A26 _H	003C26 _H				
003A27 _H	003C27 _H				
003A28 _H	003C28 _H	ID レジスタ 2	IDR2	R/W	XXXXXXXX XXXXXXXX
003A29 _H	003C29 _H				XXXXXX--- XXXXXXXX
003A2A _H	003C2A _H				
003A2B _H	003C2B _H				
003A2C _H	003C2C _H	ID レジスタ 3	IDR3	R/W	XXXXXXXX XXXXXXXX
003A2D _H	003C2D _H				XXXXXX--- XXXXXXXX
003A2E _H	003C2E _H				
003A2F _H	003C2F _H				
003A30 _H	003C30 _H	ID レジスタ 4	IDR4	R/W	XXXXXXXX XXXXXXXX
003A31 _H	003C31 _H				XXXXXX--- XXXXXXXX
003A32 _H	003C32 _H				
003A33 _H	003C33 _H				
003A34 _H	003C34 _H	ID レジスタ 5	IDR5	R/W	XXXXXXXX XXXXXXXX
003A35 _H	003C35 _H				XXXXXX--- XXXXXXXX
003A36 _H	003C36 _H				
003A37 _H	003C37 _H				
003A38 _H	003C38 _H	ID レジスタ 6	IDR6	R/W	XXXXXXXX XXXXXXXX
003A39 _H	003C39 _H				XXXXXX--- XXXXXXXX
003A3A _H	003C3A _H				
003A3B _H	003C3B _H				
003A3C _H	003C3C _H	ID レジスタ 7	IDR7	R/W	XXXXXXXX XXXXXXXX
003A3D _H	003C3D _H				XXXXXX--- XXXXXXXX
003A3E _H	003C3E _H				
003A3F _H	003C3F _H				
003A40 _H	003C40 _H	ID レジスタ 8	IDR8	R/W	XXXXXXXX XXXXXXXX
003A41 _H	003C41 _H				XXXXXX--- XXXXXXXX
003A42 _H	003C42 _H				
003A43 _H	003C43 _H				

表 21.4-1 メッセージバッファ (ID レジスタ) の一覧 (2 / 2)

アドレス		レジスタ	略号	アクセス	初期値
CAN0	CAN1				
003A44 _H	003C44 _H	ID レジスタ 9	IDR9	R/W	XXXXXXXX XXXXXXXX
003A45 _H	003C45 _H				XXXXXX--- XXXXXXXX
003A46 _H	003C46 _H				
003A47 _H	003C47 _H				
003A48 _H	003C48 _H	ID レジスタ 10	IDR10	R/W	XXXXXXXX XXXXXXXX
003A49 _H	003C49 _H				XXXXXX--- XXXXXXXX
003A4A _H	003C4A _H				
003A4B _H	003C4B _H				
003A4C _H	003C4C _H	ID レジスタ 11	IDR11	R/W	XXXXXXXX XXXXXXXX
003A4D _H	003C4D _H				XXXXXX--- XXXXXXXX
003A4E _H	003C4E _H				
003A4F _H	003C4F _H				
003A50 _H	003C50 _H	ID レジスタ 12	IDR12	R/W	XXXXXXXX XXXXXXXX
003A51 _H	003C51 _H				XXXXXX--- XXXXXXXX
003A52 _H	003C52 _H				
003A53 _H	003C53 _H				
003A54 _H	003C54 _H	ID レジスタ 13	IDR13	R/W	XXXXXXXX XXXXXXXX
003A55 _H	003C55 _H				XXXXXX--- XXXXXXXX
003A56 _H	003C56 _H				
003A57 _H	003C57 _H				
003A58 _H	003C58 _H	ID レジスタ 14	IDR14	R/W	XXXXXXXX XXXXXXXX
003A59 _H	003C59 _H				XXXXXX--- XXXXXXXX
003A5A _H	003C5A _H				
003A5B _H	003C5B _H				
003A5C _H	003C5C _H	ID レジスタ 15	IDR15	R/W	XXXXXXXX XXXXXXXX
003A5D _H	003C5D _H				XXXXXX--- XXXXXXXX
003A5E _H	003C5E _H				
003A5F _H	003C5F _H				

21.5 メッセージバッファ (DLC レジスタ / データレジスタ) の一覧

表 21.5-1 に、メッセージバッファ (DLC レジスタ) の一覧を、表 21.5-2 に、メッセージバッファ (データレジスタ) の一覧を示します。

■ メッセージバッファ (DLC レジスタ) の一覧

表 21.5-1 メッセージバッファ (DLC レジスタ) の一覧

アドレス		レジスタ	略号	アクセス	初期値
CAN0	CAN1				
003A60 _H	003C60 _H	DLC レジスタ 0	DLCR0	R/W	-----XX
003A61 _H	003C61 _H				
003A62 _H	003C62 _H	DLC レジスタ 1	DLCR1	R/W	-----XX
003A63 _H	003C63 _H				
003A64 _H	003C64 _H	DLC レジスタ 2	DLCR2	R/W	-----XX
003A65 _H	003C65 _H				
003A66 _H	003C66 _H	DLC レジスタ 3	DLCR3	R/W	-----XX
003A67 _H	003C67 _H				
003A68 _H	003C68 _H	DLC レジスタ 4	DLCR4	R/W	-----XX
003A69 _H	003C69 _H				
003A6A _H	003C6A _H	DLC レジスタ 5	DLCR5	R/W	-----XX
003A6B _H	003C6B _H				
003A6C _H	003C6C _H	DLC レジスタ 6	DLCR6	R/W	-----XX
003A6D _H	003C6D _H				
003A6E _H	003C6E _H	DLC レジスタ 7	DLCR7	R/W	-----XX
003A6F _H	003C6F _H				
003A70 _H	003C70 _H	DLC レジスタ 8	DLCR8	R/W	-----XX
003A71 _H	003C71 _H				
003A72 _H	003C72 _H	DLC レジスタ 9	DLCR9	R/W	-----XX
003A73 _H	003C73 _H				
003A74 _H	003C74 _H	DLC レジスタ 10	DLCR10	R/W	-----XX
003A75 _H	003C75 _H				
003A76 _H	003C76 _H	DLC レジスタ 11	DLCR11	R/W	-----XX
003A77 _H	003C77 _H				
003A78 _H	003C78 _H	DLC レジスタ 12	DLCR12	R/W	-----XX
003A79 _H	003C79 _H				
003A7A _H	003C7A _H	DLC レジスタ 13	DLCR13	R/W	-----XX
003A7B _H	003C7B _H				
003A7C _H	003C7C _H	DLC レジスタ 14	DLCR14	R/W	-----XX
003A7D _H	003C7D _H				
003A7E _H	003C7E _H	DLC レジスタ 15	DLCR15	R/W	-----XX
003A7F _H	003C7F _H				

■ メッセージバッファ (データレジスタ) の一覧

表 21.5-2 メッセージバッファ (データレジスタ) の一覧

アドレス		レジスタ	略号	アクセス	初期値
CAN0	CAN1				
003A80 _H ~ 003A87 _H	003C80 _H ~ 003C87 _H	データレジスタ 0 (8 バイト)	DTR0	R/W	XXXXXXXX ~ XXXXXXXX
003A88 _H ~ 003A8F _H	003C88 _H ~ 003C8F _H	データレジスタ 1 (8 バイト)	DTR1	R/W	XXXXXXXX ~ XXXXXXXX
003A90 _H ~ 003A97 _H	003C90 _H ~ 003C97 _H	データレジスタ 2 (8 バイト)	DTR2	R/W	XXXXXXXX ~ XXXXXXXX
003A98 _H ~ 003A9F _H	003C98 _H ~ 003C9F _H	データレジスタ 3 (8 バイト)	DTR3	R/W	XXXXXXXX ~ XXXXXXXX
003AA0 _H ~ 003AA7 _H	003CA0 _H ~ 003CA7 _H	データレジスタ 4 (8 バイト)	DTR4	R/W	XXXXXXXX ~ XXXXXXXX
003AA8 _H ~ 003AAF _H	003CA8 _H ~ 003CAF _H	データレジスタ 5 (8 バイト)	DTR5	R/W	XXXXXXXX ~ XXXXXXXX
003AB0 _H ~ 003AB7 _H	003CB0 _H ~ 003CB7 _H	データレジスタ 6 (8 バイト)	DTR6	R/W	XXXXXXXX ~ XXXXXXXX
003AB8 _H ~ 003ABF _H	003CB8 _H ~ 003CBF _H	データレジスタ 7 (8 バイト)	DTR7	R/W	XXXXXXXX ~ XXXXXXXX
003AC0 _H ~ 003AC7 _H	003CC0 _H ~ 003CC7 _H	データレジスタ 8 (8 バイト)	DTR8	R/W	XXXXXXXX ~ XXXXXXXX
003AC8 _H ~ 003ACF _H	003CC8 _H ~ 003CCF _H	データレジスタ 9 (8 バイト)	DTR9	R/W	XXXXXXXX ~ XXXXXXXX
003AD0 _H ~ 003AD7 _H	003CD0 _H ~ 003CD7 _H	データレジスタ 10 (8 バイト)	DTR10	R/W	XXXXXXXX ~ XXXXXXXX
003AD8 _H ~ 003ADF _H	003CD8 _H ~ 003CDF _H	データレジスタ 11 (8 バイト)	DTR11	R/W	XXXXXXXX ~ XXXXXXXX
003AE0 _H ~ 003AE7 _H	003CE0 _H ~ 003CE7 _H	データレジスタ 12 (8 バイト)	DTR12	R/W	XXXXXXXX ~ XXXXXXXX
003AE8 _H ~ 003AEF _H	003CE8 _H ~ 003CEF _H	データレジスタ 13 (8 バイト)	DTR13	R/W	XXXXXXXX ~ XXXXXXXX
003AF0 _H ~ 003AF7 _H	003CF0 _H ~ 003CF7 _H	データレジスタ 14 (8 バイト)	DTR14	R/W	XXXXXXXX ~ XXXXXXXX
003AF8 _H ~ 003AFF _H	003CF8 _H ~ 003CFF _H	データレジスタ 15 (8 バイト)	DTR15	R/W	XXXXXXXX ~ XXXXXXXX

21.6 CAN コントローラのレジスタ分類

CAN コントローラのレジスタは、次の 3 つに分類されます。

- 全体制御レジスタ
 - メッセージバッファ制御レジスタ
 - メッセージバッファ
-

■ 全体制御レジスタ

全体制御レジスタには、次の 4 つがあります。

- 制御ステータスレジスタ (CSR)
- ラストイベント表示レジスタ (LEIR)
- 受信および送信エラーカウンタ (RTEC)
- ビットタイミングレジスタ (BTR)

■ メッセージバッファ制御レジスタ

メッセージバッファ制御レジスタには、次の 14 のレジスタがあります。

- メッセージバッファ有効レジスタ (BVALR)
- IDE レジスタ (IDER)
- 送信要求レジスタ (TREQR)
- 送信 RTR レジスタ (TRTRR)
- リモートフレーム受信待機レジスタ (RFWTR)
- 送信解除レジスタ (TCANR)
- 送信完了レジスタ (TCR)
- 送信割込み許可レジスタ (TIER)
- 受信完了レジスタ (RCR)
- リモート要求受信レジスタ (RRTRR)
- 受信オーバランレジスタ (ROVRR)
- 受信割込み許可レジスタ (RIER)
- アクセプタンスマスク選択レジスタ (AMSR)
- アクセプタンスマスクレジスタ 0 および 1 (AMR0/AMR1)

■ メッセージバッファ

メッセージバッファは、次の 3 つがあります。

- ID レジスタ $x(x=0 \sim 15)$ (IDRx)
- DLC レジスタ $x(x=0 \sim 15)$ (DLCRx)
- データレジスタ $x(x=0 \sim 15)$ (DTRx)

21.6.1 制御ステータスレジスタ (CSR)

制御ステータスレジスタ (CSR) の下位 8 ビットに対しては、ビット操作命令 (リードモディファイライト) を実行することは禁止されています。ただし、HALT ビットが変化しないタイミング (マクロの初期化時など) では、ビット操作命令を使用しても問題ありません。

■ 制御ステータスレジスタ (CSR)

CAN 制御ステータスレジスタ (CSR) のビット構成を図 21.6-1 に示します。

図 21.6-1 制御ステータスレジスタ (CSR)

アドレス : 003B01 _H (CAN0) 003D01 _H (CAN1)	15	14	13	12	11	10	9	8	ビット No.
	TS	RS				NT	NS1	NS0	CSR
リード / ライト	(R)	(R)	(-)	(-)	(-)	(R/W)	(R)	(R)	
初期値	(0)	(0)	(-)	(-)	(-)	(0)	(0)	(0)	
アドレス : 003B00 _H (CAN0) 003D00 _H (CAN1)	7	6	5	4	3	2	1	0	ビット No.
	TOE					NIE	予約	HALT	CSR
リード / ライト	(R/W)	(-)	(-)	(-)	(-)	(R/W)	(W)	(R/W)	
初期値	(0)	(-)	(-)	(-)	(-)	(0)	(0)	(1)	

【bit 15】TS: 送信ステータスビット

本ビットはメッセージが送信中であるかどうかを示します。

0: メッセージは送信されていません。

1: メッセージが送信されています。

本ビットはエラーフレームおよび過負荷フレームが送信されている間も "0" です。

【bit 14】RS: 受信ステータスビット

本ビットはメッセージが受信されているかどうかを示します。

0: メッセージは受信されていません。

1: メッセージが受信されています。

メッセージがバス上にある間、本ビットは "1" になります。このため本ビットはメッセージが送信されている間も "1" になります。本ビットは必ずしも、受信メッセージがアクセプタンスフィルタを通過したかどうかを示すものではありません。

結果として本ビットが "0" であるときは、バス動作は停止している (HALT = 0) か、バスはインターミッション / バスアイドルにある、またはエラー / オーバロードフレームがバス上にあることを意味します。

【bit 10】NT: ノードステータス遷移フラグ

ノードステータスがインクリメントに変化し,またはバスオフからエラーアクティブに変化しても本ビットは "1" になります。

すなわち, NT ビットは, ノードステータスがエラーアクティブ (00_B) からワーニング (01_B) に, ワーニング (01_B) からエラーパッシブ (10_B) に, エラーパッシブ (10_B) からバスオフ (11_B) に, バスオフ (11_B) からエラーアクティブ (00_B) に変化したときに "1" にセットされます。括弧内の数字は NS1 ビットおよび NS0 ビットの値を示します。

ノードステータス遷移割込み許可ビット (NIE) が "1" であるときに割込みが発生します。"0" を書き込むと NT ビットが "0" にセットされます。NT ビットに "1" を書き込んでも無視されます。リードモディファイライト系命令が読み出されたときに "1" が読み出されます。

【bit 9, bit 8】NS1, NS0: ノードステータスビット 1 および 0

本ビット群は現在のノードステータスを表します。

表 21.6-1 NS1, NS0 およびノードステータスの間の対応

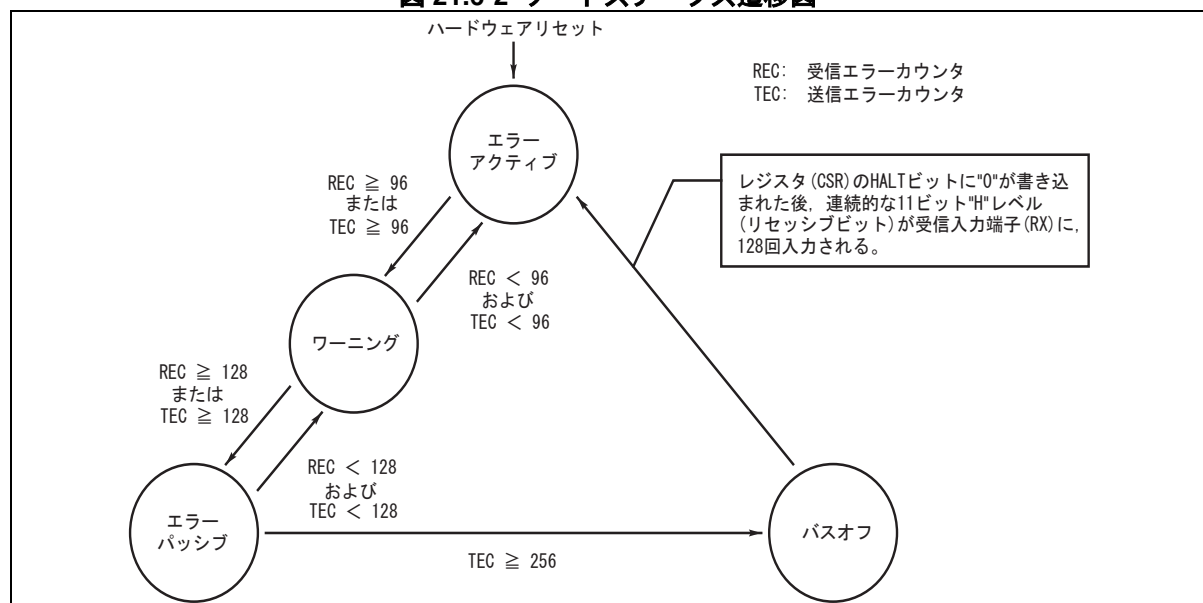
NS1	NS0	ノードステータス
0	0	エラーアクティブ
0	1	ワーニング (エラーアクティブ)
1	0	エラーパッシブ
1	1	バスオフ

< 注意事項 >

ワーニング (エラーアクティブ) はノードステータスに対する CAN 仕様 2.0B におけるエラーアクティブに含まれますが, 送信エラーカウンタまたは受信エラーカウンタが 96 を超えたことを表します。

ノードステータスの変化図を図 21.6-2 に示します。

図 21.6-2 ノードステータス遷移図



【bit 7】TOE: 送信出力許可ビット

本ビットに "1" を書き込むと汎用ポート端子を CAN コントローラの送信端子に切り換えます。

0: 汎用ポート端子

1: CAN コントローラの送信端子

【bit 2】NIE: ノードステータス遷移割込み許可ビット

本ビットはノードステータス遷移割込み (NT = 1 の場合) を許可または禁止します。

0: ノードステータス遷移割込みを禁止する

1: ノードステータス遷移割込みを許可する

【bit 1】Reserved: 予約ビット

このビットには常に "0" を書き込んでください。リード値は常に "0" です。

【bit 0】HALT: バス動作停止ビット

読出し時

0: バス動作中

1: バス動作停止

書込み時

0: バス動作停止を解除

1: バス動作停止をセット

< 注意事項 >

バスオフ時に HALT ビットに "0" を書き込む場合は、HALT ビットが "1" になっていることを確認後、"0" を書き込んでください。

参考プログラム例：

```
switch (IO_CANCTO.CSE.bit.NS)
{
    case 0 : /* error active */
        break;
    case 1 : /* warning */
        break;
    case 2 : /* error passive */
        break;
    default: /* bus off */
        for (1=0; (1<=500)&&(IO_CANCTO.CSR.bit.HALT=0);1++);
        IO_CANCTO.CSR.word = 0x0084; /* HALT */
        break;
}
```

* : 変数はフェイルセーフのために使用しています。

21.6.2 バス動作停止ビット (HALT=1)

バス動作停止ビットは、バス動作停止をセットもしくは解除し、またはその状態を表示します。

■ バス動作停止をセットする条件 (HALT=1)

バス動作停止をセットする条件 (HALT=1) として、次の 3 つがあります。

- ハードウェアリセットの後
- ノードステータスがバスオフに変化したとき
- HALT への "1" の書込み

< 注意事項 >

- バス動作は F²MC-16LX が低消費電力消費モード (停止モード、クロックモードおよびハードウェアスタンバイモード) に遷移する前に HALT に "1" を書き込むことにより停止しなければなりません。HALT に "1" が書き込まれるときに、送信が進行中であれば送信が終了した後にバス動作は停止します (HALT=1)。また、HALT に "1" が書き込まれるときに、受信が進行中であればバス動作は直ちに停止します (HALT = 1)。もし受信されたメッセージがメッセージバッファ(x) に格納中であればメッセージを格納した後に、バス動作は停止します (HALT = 1)。
- バス動作が停止したかどうかは、常に HALT ビットを読み出してチェックしてください。

■ バス動作停止を解除する条件 (HALT = 0)

HALT への "0" の書込み

< 注意事項 >

- 上記条件のハードウェアリセット後または HALT に "1" を書き込むことによるバス動作停止の解除は、HALT に "0" が書き込まれ受信入力端子 (RX) に継続的な 11 ビットの "H" レベル (リセシブビット) が入力された後に行われます。
 - 上記条件のノードステータスがバスオフに変化したときのバス動作停止の解除は、HALT に "0" が書き込まれ受信入力端子 (RX) に継続的な 11 ビットの "H" レベル (リセシブビット) が 128 回入力された後に行われます。次に送信および受信エラーカウンタの両者の値が "0" に達してノードステータスはエラーアクティブに変化します。
 - バスオフ時に HALT ビットに "0" を書き込む場合は、HALT ビットが "1" になっていることを確認後、"0" を書き込んでください。
-

■ バス動作停止間の状態 (HALT=1)

- バスは、送信および受信などの動作を一切行いません。
- 送信出力端子 (TX) は、"H" レベル (リセッシブビット) を出力します。
- ほかのレジスタおよびエラーカウンタの値は変化しません。

< 注意事項 >

ビットタイミングレジスタ (BTR) はバス動作停止 (HALT = 1) の間にセットしなければなりません。

21.6.3 ラストイベント表示レジスタ (LEIR)

本レジスタはラストイベントを示します。

NTE, TCE および RCE は排他的なものです。ラストイベントのビットが "1" にセットされたときほかのビットは "0" にセットされます。

■ ラストイベント表示レジスタ (LEIR)

ラストイベント表示レジスタ (LEIR) のビット構成を図 21.6-3 に示します。

図 21.6-3 ラストイベント表示レジスタ (LEIR)

アドレス : 003B02 _H (CAN0) 003D02 _H (CAN1)	7	6	5	4	3	2	1	0	ビット No.
	NTE	TCE	RCE		MBP3	MBP2	MBP1	MBP0	LEIR
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)	(0)	

【bit 7】 NTE: ノードステータス遷移イベントビット

- 本ビットが "1" のときはノードステータス遷移がラストイベントであることを表します。
- 本ビットは制御ステータスレジスタ (CSR) の NT ビットと同時に "1" にセットされます。
- 本ビットは、CSR のノードステータス遷移割込み許可ビット (NIE) のセッティングとは無関係に "1" にセットされます。
- 本ビットに "0" を書き込むと NT ビットは "0" にセットされます。本ビットに "1" を書き込んでも無視されます。
- リードモディファイライト系命令が実行されたときは "1" が読み出されます。

【bit 6】 TCE: 送信完了イベントビット

本ビットが "1" のときは送信完了がラストイベントであることを表します。

本ビットは送信完了レジスタ (TCR) のビットのいずれか 1 つと同時に "1" にセットされます。また本ビットは送信割込み許可レジスタ (TIER) のビットのセッティングとは無関係に "1" にセットされます。

"0" を書き込むと本ビットは "0" にセットされます。本ビットに "1" を書き込んでも無視されます。

リードモディファイライト系命令が実行されたときは "1" が読み出されます。

本ビットが "1" にセットされたとき MBP3 ~ MBP0 ビットは送信動作を完了したメッセージバッファの番号を示すために使用されます。

【bit 5】RCE: 受信完了イベントビット

本ビットが "1" であるときは受信完了がラストイベントであることを表します。

本ビットは受信完了レジスタ (RCR) のビットのいずれか 1 つと同時に "1" にセットされます。また本ビットは受信割込み許可レジスタ (RIER) のビットのセッティングとは無関係に "1" にセットされます。

"0" を書き込むと本ビットは "0" にセットされます。本ビットに "1" を書き込んでも無視されます。

リードモディファイライト系命令が実行されたときは "1" が読み出されます。

本ビットが "1" にセットされたとき MBP3 ~ MBP0 ビットは受信動作を完了したメッセージバッファの番号を示すために使用されます。

【bit 3 ~ bit 0】MBP3 ~ MBP0: メッセージバッファポインタビット

TCE または RCE ビットが "1" にセットされたときに本ビット群は対応するメッセージバッファ (0 ~ 15) の番号を示します。

NTE ビットが "1" にセットされると本ビット群は意味を持ちません。

"0" を書き込むと本ビット群は "0" にセットされます。本ビット群に "1" を書き込んでも無視されます。

リードモディファイライト系命令が実行されたときは "1" が読み出されます。

もし LEIR が CAN 割込みハンドラ内でアクセスされたとしても割込みを引き起こしたイベントは必ずしも LEIR により表されるものと同じではありません。割込みハンドラ内で LEIR アクセスに対する割込み要求の時点でほかの CAN イベントが生ずる可能性があります。

21.6.4 受信および送信エラーカウンタ (RTEC)

受信および送信エラーカウンタは、CAN 仕様で定められる送信エラーカウンタおよび受信エラーカウンタを示します。このレジスタは読出し専用です。

■ 受信および送信エラーカウンタ (RTEC)

受信および送信エラーカウンタ (RTEC) のビット構成を図 21.6-4 に示します。

図 21.6-4 受信および送信エラーカウンタ (RTEC)

アドレス : 003B05 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D05 _H (CAN1)	TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0	RTEC
リード / ライト	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

アドレス : 003B04 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D04 _H (CAN1)	REC7	REC6	REC5	REC4	REC3	REC2	REC1	REC0	RTEC
リード / ライト	(R)	(R)	(R)	(R)	(R)	(R)	(R)	(R)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

【bit 15 ~ bit 8】TEC7 ~ TEC0: 送信エラーカウンタ

これらは送信エラーカウンタです。

TEC7 ~ TEC0 は、カウンタ値が 256 より大きい場合 0 ~ 7 を示します。以降のインクリメントはカウンタ値にカウントされません。この場合バスオフがノードステータスで示されます (制御ステータスレジスタ CSR の NS1 および NS0 = 11_B)。

【bit 7 ~ bit 0】REC7 ~ REC0: 受信エラーカウンタ

これらは受信エラーカウンタです。

REC7 ~ REC0 は、カウンタ値が 256 より大きい場合に 0 ~ 7 を示します。以降のインクリメントはカウンタ値にカウントされません。この場合エラーパッシブがノードステータスで示されます (制御ステータスレジスタ CSR の NS1 および NS0 = 10_B)。

21.6.5 ビットタイミングレジスタ (BTR)

ビットタイミングレジスタ (BTR) は、プリスケアラおよびビットタイミングをセットするものです。

■ ビットタイミングレジスタ (BTR)

ビットタイミングレジスタ (BTR) のビット構成を図 21.6-5 に示します。

図 21.6-5 ビットタイミングレジスタ (BTR)

アドレス : 003B07 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D07 _H (CAN1)		TS2.2	TS2.1	TS2.0	TS1.3	TS1.2	TS1.1	TS1.0	BTR
リード / ライト	(-)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(-)	(1)	(1)	(1)	(1)	(1)	(1)	(1)	

アドレス : 003B06 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D06 _H (CAN1)	RSJ1	RSJ0	PSC5	PSC4	PSC3	PSC2	PSC1	PSC0	BTR
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	(1)	(1)	(1)	(1)	(1)	(1)	(1)	(1)	

< 注意事項 >

本レジスタはバス動作停止 (HALT = 1) の間にセットしなければなりません。

【bit 14 ~ bit 12】TS2.2 ~ TS2.0: タイムセグメント 2 セットビット 2 ~ 0

本ビット群は単位時間 (TQ) を [(TS2.2 ~ TS2.0)+1] で分周してタイムセグメント 2(TSEG2) を決定します。タイムセグメント 2 は CAN 仕様におけるフェーズバッファセグメント 2(PHASE_SEG2) に等しいものです。

【bit 11 ~ bit 8】TS1.3 ~ TS1.0: タイムセグメント 1 設定ビット 3 ~ 0

本ビット群は単位時間 (TQ) を [(TS1.3 ~ TS1.0)+1] で分周してタイムセグメント 1(TSEG1) を決定します。タイムセグメント 1 は CAN 仕様におけるプロパゲーションセグメント (PROP_SEG)+ フェーズバッファセグメント 1(PHASE_SEG1) に等しいものです。

【bit 7, bit 6】RSJ1, RSJ0: 再同期ジャンプ幅設定ビット 1 および 0

本ビット群は単位時間 (TQ) を [(RSJ1 ~ RSJ0)+1] で分周して再同期ジャンプ幅を決定させます。

【bit 5 ~ bit 0】PSC5 ~ PSC0: プリスケアラ設定ビット 5 ~ 0

本ビット群は入力クロックを [(PSC5 ~ PSC0)+1] の周波数で分割して CAN コントローラの単位時間 (TQ) を決定します。

CAN 仕様および CAN コントローラの各々におけるビット時間セグメントは図 21.6-6 および図 21.6-7 に示されています。

図 21.6-6 CAN 仕様におけるビット時間セグメント



図 21.6-7 CAN コントローラにおけるビット時間セグメント



入力クロック (CLK), 単位時間 (TQ), ビットタイム (BT), 同期セグメント (SYNC_SEG), タイムセグメント 1 (TSEG1), タイムセグメント 2 (TSEG2) および再同期ジャンプ幅 $[(RSJ1+RSJ0)+1]$ 周波数分割としたときの $PSC = PSC5 \sim PSC0$, $TS1 = TS1.3 \sim TS1.0$, $TS2 = TS2.2 \sim TS2.0$, $RSJ = RSJ1$ および $RSJ0$ の間の関係は次に示すとおりです。

入力クロックは, マシンクロックに組み込まれています。

$$TQ = (PSC + 1) \times CLK$$

$$\begin{aligned} BT &= SYNC_SEG + TSEG1 + TSEG2 \\ &= (1 + (TS1 + 1) + (TS2 + 1)) \times TQ \\ &= (3 + TS1 + TS2) TQ \end{aligned}$$

$$RSJW = (RSJ + 1) \times TQ$$

正しく動作を行うためには以下の条件を満足しなければなりません。

G サフィックス付きの製品

- 1 PSC 63 の場合
 - TSEG1 2TQ
 - TSEG1 RSJW
 - TSEG2 2TQ
 - TSEG2 RSJW
- PSC=0 の場合
 - TSEG1 5TQ
 - TSEG2 2TQ
 - TSEG2 RSJW

G サフィックスなしの製品

- 1 PSC 63 の場合
 - TSEG1 RSJW
 - TSEG2 RSJW+2TQ
- PSC=0 の場合
 - TSEG1 5TQ
 - TSEG2 RSJW+2TQ

CAN 仕様に従ったビットタイミングの設定をするためには, 遅延時間などほかの条件も考慮しなければなりません。

21.6.6 メッセージバッファ有効レジスタ (BVALR)

メッセージバッファ有効レジスタ (BVALR) は、メッセージバッファ (x) の有効性をセットし、またはその状態を表示するものです。

■ メッセージバッファ有効レジスタ (BVALR)

メッセージバッファ有効レジスタ (BVALR) のビット構成を図 21.6-8 に示します。

図 21.6-8 メッセージバッファ有効レジスタ (BVALR)

アドレス : 000071 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
000081 _H (CAN1)	BVAL15	BVAL14	BVAL13	BVAL12	BVAL11	BVAL10	BVAL9	BVAL8	BVALR
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)	
アドレス : 000070 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
000080 _H (CAN1)	BVAL7	BVAL6	BVAL5	BVAL4	BVAL3	BVAL2	BVAL1	BVAL0	BVALR
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)	

0: メッセージバッファ (x) 無効

1: メッセージバッファ (x) 有効

メッセージバッファ (x) が無効にセットされると、メッセージの送信または受信を行いません。

バッファが送信動作の間に無効にセットされると、送信が完了またはエラーにより終了した後で無効 (BVAL_x=0) になります。

バッファが受信動作の間に無効にセットされると、直ちに無効 (BVAL_x=0) になります。もし、受信メッセージがメッセージバッファ (x) に格納されていればメッセージバッファ (x) はメッセージを格納した後に無効となります。

< 注意事項 >

- x はメッセージバッファ番号を示します (x=0 ~ 15)。
- ビット (BVAL_x) に "0" を書き込むことによりメッセージバッファ (x) を無効としたときにビット操作命令の実行はそのビットが "0" にセットされるまで禁止されます。
- CAN コントローラが CAN 通信に参加している状態 (CSR: HALT ビットの読出し値が "0" で、CAN コントローラが CAN バスの通信に参加しており送受信が可能な状態) で、メッセージバッファを無効 (BVALR:BVAL = 0) にする場合は「21.14 CAN コントローラの使用上の注意」に従ってください。

21.6.7 IDE レジスタ (IDER)

本レジスタは送信 / 受信の間にメッセージバッファ (x) により使用されるフレームフォーマットを設定します。

■ IDE レジスタ (IDER)

IDE レジスタ (IDER) のビット構成を図 21.6-9 に示します。

図 21.6-9 IDE レジスタ (IDER)

アドレス : 003B09 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D09 _H (CAN1)	IDE15	IDE14	IDE13	IDE12	IDE11	IDE10	IDE9	IDE8	IDER
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003B08 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D08 _H (CAN1)	IDE7	IDE6	IDE5	IDE4	IDE3	IDE2	IDE1	IDE0	IDER
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

0: メッセージバッファ (x) に対して標準フレームフォーマット (ID11 ビット) が使用されます。

1: メッセージバッファ (x) に対して拡張フレームフォーマット (ID29 ビット) が使用されます。

< 注意事項 >

- 本レジスタはメッセージバッファ (x) が無効であるとき (メッセージバッファ有効レジスタの BVALx(BVALR) = 0) に設定しなければなりません。バッファが有効である (BVALx = 1) ときに設定を行うと受信メッセージを不必要に格納する可能性があります。
- CAN コントローラが CAN 通信に参加している状態 (CSR: HALT ビットの読出し値が "0" で、CAN コントローラが CAN バスの通信に参加しており送受信が可能な状態) で、メッセージバッファを無効 (BVALR:BVAL = 0) にする場合は、「21.14 CAN コントローラの使用上の注意」に従ってください。

21.6.8 送信要求レジスタ (TREQR)

送信要求レジスタ (TREQR) は、メッセージバッファ (x) に対する送信要求を設定し、またはその状態を表示します。

■ 送信要求レジスタ (TREQR)

送信要求レジスタ (TREQR) のビット構成を図 21.6-10 に示します。

図 21.6-10 送信要求レジスタ (TREQR)

アドレス : 000073 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
000083 _H (CAN1)	TREQ15	TREQ14	TREQ13	TREQ12	TREQ11	TREQ10	TREQ9	TREQ8	TREQR
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)	

アドレス : 000072 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
000082 _H (CAN1)	TREQ7	TREQ6	TREQ5	TREQ4	TREQ3	TREQ2	TREQ1	TREQ0	TREQR
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)	

TREQ_x に "1" が書き込まれたときメッセージバッファ (x) に対する送信を開始します。もしリモートフレーム受信待機レジスタ (RFTWR)^{*1} の RFTW_x が "0" であれば送信は直ちに開始します。しかし RFTW_x = 1 であればリモートフレームが受信される (リモート要求受信レジスタ (RRTRR)^{*1} が "1" になる) まで待機した後に送信が開始します。もし TREQ_x に "1" が書き込まれるときに RRTR_x が既に "1" であれば、RFTW_x = 1 であっても送信は直ちに開始します^{*2}。

*1: TRTRR および RFTWR に関しては、「21.6.9 送信 RTR レジスタ (TRTRR)」および「21.6.10 リモートフレーム受信待機レジスタ (RFTWR)」を参照してください。

*2: 送信の解除に関しては、「21.6.11 送信解除レジスタ (TCANR)」および「21.6.12 送信完了レジスタ (TCR)」を参照してください。

TREQ_x に "0" を書き込んでも無視されます。

リードモディファイライト系命令が読み出されたときは "0" が読み出されます。

もし送信動作の完了時における ("0" への) クリアおよび "1" の書き込みによる設定が同時に生じればクリア動作が優先されます。

もし 1 つ以上のビットに "1" が書き込まれれば下位番号のメッセージバッファ (x) から開始して送信が行われます。

TREQ_x は送信が待ち状態中は "1" であり、送信が完了または解除されたときに "0" になります。

21.6.9 送信 RTR レジスタ (TRTRR)

本レジスタはメッセージバッファ (x) による送信 RTR(リモート送信要求) ビットを設定するものです。

■ 送信 RTR レジスタ (TRTRR)

送信 RTR レジスタ (TRTRR) のビット構成を図 21.6-11 に示します。

図 21.6-11 送信 RTR レジスタ (TRTRR)

アドレス : 003B0B _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D0B _H (CAN1)	TRTR15	TRTR14	TRTR13	TRTR12	TRTR11	TRTR10	TRTR9	TRTR8	TRTRR
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)	
アドレス : 003B0A _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D0A _H (CAN1)	TRTR7	TRTR6	TRTR5	TRTR4	TRTR3	TRTR2	TRTR1	TRTR0	TRTRR
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)	

0: データフレームが送信されます。

1: リモートフレームが送信されます。

21.6.10 リモートフレーム受信待機レジスタ (RFWTR)

リモートフレーム受信待機レジスタ (RFWTR) は、データフレーム送信に対する要求が設定された (送信要求レジスタ (TREQR) の TREQx が "1" であり送信 RTR レジスタ (TRTRR) の TRTRx が "0" である) ときに送信を開始する条件を設定します。

0: 送信は直ちに開始します。

1: リモートフレーム受信 (リモート要求受信レジスタ (RRTRR) が "1" になる) まで待機した後に送信を開始します。

■ リモートフレーム受信待機レジスタ (RFWTR)

リモートフレーム受信待機レジスタ (RFWTR) のビット構成を図 21.6-12 に示します。

図 21.6-12 リモートフレーム受信待機レジスタ (RFWTR)

アドレス : 003B0D _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D0D _H (CAN1)	RFWT15	RFWT14	RFWT13	RFWT12	RFWT11	RFWT10	RFWT9	RFWT8	RFWTR
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003B0C _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D0C _H (CAN1)	RFWT7	RFWT6	RFWT5	RFWT4	RFWT3	RFWT2	RFWT1	RFWT0	RFWTR
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

< 注意事項 >

- 送信に対する要求が設定されたときに RRTRx が既に "1" であれば送信は直ちに開始します。
- リモートフレーム送信に対しては RFWTx を "1" にセットしないでください。

21.6.11 送信解除レジスタ (TCANR)

TCANx に "1" が書き込まれたときに本レジスタはメッセージバッファ (x) への送信に対する待ち状態中の要求を解除します。

解除の完了時に送信要求レジスタ (TREQR) の TREQx は "0" になります。TCANx に "0" を書き込んでも無視されます。

これは書き込み専用レジスタであり、そのリード値は常に "0" です。

■ 送信解除レジスタ (TCANR)

送信解除レジスタ (TCANR) のビット構成を図 21.6-13 に示します。

図 21.6-13 送信解除レジスタ (TCANR)

アドレス : 000075 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
000085 _H (CAN1)	TCAN15	TCAN14	TCAN13	TCAN12	TCAN11	TCAN10	TCAN9	TCAN8	TCANR
リード / ライト	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	
アドレス : 000074 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
000084 _H (CAN1)	TCAN7	TCAN6	TCAN5	TCAN4	TCAN3	TCAN2	TCAN1	TCAN0	TCANR
リード / ライト	(W)	(W)	(W)	(W)	(W)	(W)	(W)	(W)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

21.6.12 送信完了レジスタ (TCR)

メッセージバッファ (x) により送信の完了時に対応する TCx は "1" になります。もし送信完了割込み許可レジスタ (TIER) の TIE_x が "1" であれば、割込みが生じます。

■ 送信完了レジスタ (TCR)

送信完了レジスタ (TCR) のビット構成を図 21.6-14 に示します。

図 21.6-14 送信完了レジスタ (TCR)

アドレス : 000077 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
000087 _H (CAN1)	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8	TCR
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)	

アドレス : 000076 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
000086 _H (CAN1)	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0	TCR
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)	

● TC_x = 0 となる条件

- TC_x に "0" を書き込む。
- 送信要求レジスタ (TREQR) の TREQ_x に "1" を書き込む。

送信の完了の後で TC_x に "0" を書き込むとそれを "0" にセットします。TC_x に "1" を書き込んでも無視されます。リードモディファイライト系命令が実行されたときは "1" が読み出されます。

< 注意事項 >

送信動作の完了時の "1" への設定および "0" の書込みによるクリアが同時に生じたときは "1" への設定が優先します。

21.6.13 送信割込み許可レジスタ (TIER)

本レジスタはメッセージバッファ (x) による送信割込みを許可または禁止します。送信割込みは送信完了時 (送信完了レジスタ (TCR) の TCx が "1" になったとき) に発生します。

■ 送信割込み許可レジスタ (TIER)

送信割込み許可レジスタ (TIER) のビット構成を図 21.6-15 に示します。

図 21.6-15 送信割込み許可レジスタ (TIER)

アドレス : 003B0F _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D0F _H (CAN1)	TIE15	TIE14	TIE13	TIE12	TIE11	TIE10	TIE9	TIE8	TIER
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)	
アドレス : 003B0E _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D0E _H (CAN1)	TIE7	TIE6	TIE5	TIE4	TIE3	TIE2	TIE1	TIE0	TIER
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)	

0: 送信割込みは禁止されます。

1: 送信割込みは許可されます。

21.6.14 受信完了レジスタ (RCR)

メッセージバッファ (x) への受信メッセージの格納完了時に RCx は "1" になります。
もし受信完了割り込み許可レジスタの RIE_x が "1" であれば、割り込みが生じます。

■ 受信完了レジスタ (RCR)

受信完了レジスタ (RCR) のビット構成を図 21.6-16 に示します。

図 21.6-16 受信完了レジスタ (RCR)

アドレス : 000079 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
000089 _H (CAN1)	RC15	RC14	RC13	RC12	RC11	RC10	RC9	RC8	RCR
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)	

アドレス : 000078 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
000088 _H (CAN1)	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	RCR
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)	

● RC_x=0 となる条件

- RC_x に "0" を書き込む。

受信メッセージの処理が完了した後で RC_x に "0" を書き込んで "0" にセットしてください。RC_x に "1" を書き込んでも無視されます。

リードモディファイライト系命令が実行されたときは "1" が読み出されます。

< 注意事項 >

受信動作の完了時の "1" への設定および "0" の書込みによるクリアが同時に生じたときは、"1" への設定が優先します。

21.6.15 リモート要求受信レジスタ (RRTRR)

受信されたリモートフレームがメッセージバッファ (x) 内に格納されると、RRTRx は (RCx 設定が "1" になると同時に) "1" になります。

■ リモート要求受信レジスタ (RRTRR)

リモート要求受信レジスタ (RRTRR) のビット構成を図 21.6-17 に示します。

図 21.6-17 リモート要求受信レジスタ (RRTRR)

アドレス : 00007B _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
00008B _H (CAN1)	RRTR15	RRTR14	RRTR13	RRTR12	RRTR11	RRTR10	RRTR9	RRTR8	RRTRR
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)	

アドレス : 00007A _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
00008A _H (CAN1)	RRTR7	RRTR6	RRTR5	RRTR4	RRTR3	RRTR2	RRTR1	RRTR0	RRTRR
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)	

● RRTRx=0 となる条件

- RRTRx に "0" を書き込む。
- 受信されたデータフレームがメッセージバッファ (x) に格納された後 (RCx 設定が "1" になるのと同時)。
- メッセージバッファ (x) による送信が完了した後 (送信完了レジスタ (TCR) の TCx は "1")。

RRTRx に "1" を書き込んでも無視されます。

リードモディファイライト系命令が実行されたときは "1" が読み出されます。

< 注意事項 >

"1" への設定と "0" の書込みによるクリアが同時に生じたときは、"1" への設定が優先します。

21.6.16 受信オーバランレジスタ (ROVRR)

もし、メッセージバッファ (x) 内への受信メッセージの格納完了時に、受信完了レジスタ (RCR) が既に "1" であれば、ROVRx は "1" になり、受信がオーバランしたことを表します。

■ 受信オーバランレジスタ (ROVRR)

受信オーバランレジスタ (ROVRR) のビット構成を図 21.6-18 に示します。

図 21.6-18 受信オーバランレジスタ (ROVRR)

アドレス : 00007D _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
00008D _H (CAN1)	ROVR15	ROVR14	ROVR13	ROVR12	ROVR11	ROVR10	ROVR9	ROVR8	ROVRR
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)	

アドレス : 00007C _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
00008C _H (CAN1)	ROVR7	ROVR6	ROVR5	ROVR4	ROVR3	ROVR2	ROVR1	ROVR0	ROVRR
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)	

ROVRx に "0" を書き込むと ROVRx=0 になります。ROVRx に "1" を書き込んでも無視されます。受信がオーバランしたことをチェックした後で ROVRx に "0" を書き込むと "0" にクリアします。

リードモディファイライト系命令が実行されたときは "1" が読み出されます。

< 注意事項 >

"1" への設定と "0" の書込みによるクリアが同時に生じたときは、"1" への設定が優先します。

21.6.17 受信割込み許可レジスタ (RIER)

本レジスタはメッセージバッファ (x) による受信割込みを許可または禁止します。受信割込みは受信完了時 (受信完了レジスタ (RCR) の RCx が "1" であるとき) に発生します。

■ 受信割込み許可レジスタ (RIER)

受信割込み許可レジスタ (RIER) のビット構成を図 21.6-19 に示します。

図 21.6-19 受信割込み許可レジスタ (RIER)

アドレス : 00007F _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
00008F _H (CAN1)	RIE15	RIE14	RIE13	RIE12	RIE11	RIE10	RIE9	RIE8	RIER
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)	
アドレス : 00007E _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
00008E _H (CAN1)	RIE7	RIE6	RIE5	RIE4	RIE3	RIE2	RIE1	RIE0	RIER
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)	

0: 受信割込みは禁止されます。

1: 受信割込みは許可されます。

21.6.18 アクセプタンスマスク選択レジスタ (AMSR)

本レジスタは受信メッセージ ID とメッセージバッファ (x)ID の比較に対するマスク (アクセプタンスマスク) を選択します。

■ アクセプタンスマスク選択レジスタ (AMSR)

アクセプタンスマスク選択レジスタ (AMSR) のビット構成を図 21.6-20 に示します。

図 21.6-20 アクセプタンスマスク選択レジスタ (AMSR)

タイプ 1	7	6	5	4	3	2	1	0	ビット No.
アドレス : 003B10 _H (CAN0) 003D10 _H (CAN1)	AMS 3.1	AMS 3.0	AMS 2.1	AMS 2.0	AMS 1.1	AMS 1.0	AMS 0.1	AMS 0.0	AMSR バイト 0
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
タイプ 2	15	14	13	12	11	10	9	8	ビット No.
アドレス : 003B11 _H (CAN0) 003D11 _H (CAN1)	AMS 7.1	AMS 7.0	AMS 6.1	AMS 6.0	AMS 5.1	AMS 5.0	AMS 4.1	AMS 4.0	AMSR バイト 1
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
タイプ 3	7	6	5	4	3	2	1	0	ビット No.
アドレス : 003B12 _H (CAN0) 003D12 _H (CAN1)	AMS 11.1	AMS 11.0	AMS 10.1	AMS 10.0	AMS 9.1	AMS 9.0	AMS 8.1	AMS 8.0	AMSR バイト 2
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
タイプ 4	15	14	13	12	11	10	9	8	ビット No.
アドレス : 003B13 _H (CAN0) 003D13 _H (CAN1)	AMS 15.1	AMS 15.0	AMS 14.1	AMS 14.0	AMS 13.1	AMS 13.0	AMS 12.1	AMS 12.0	AMSR バイト 3
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

表 21.6-2 アクセプタンスマスクの選択

AMSx.1	AMSx.0	アクセプタンスマスク
0	0	フルビット比較
0	1	フルビットマスク
1	0	アクセプタンスマスクレジスタ 0(AMR0)
1	1	アクセプタンスマスクレジスタ 1(AMR1)

< 注意事項 >

- AMSx.1 および AMSx.0 はメッセージバッファ (x) が無効である (メッセージバッファ有効レジスタ (BVALR) の BVALx が "0" である) ときに設定しなければなりません。バッファが有効 (BVALx = 1) なときに設定を行うと不要な受信メッセージが格納される可能性があります。
 - CAN コントローラが CAN 通信に参加している状態 (CSR: HALT ビットの読出し値が "0" で、CAN コントローラが CAN バスの通信に参加しており送受信が可能な状態) で、メッセージバッファを無効 (BVALR: BVAL = 0) にする場合は、「21.14 CAN コントローラの使用上の注意」に従ってください。
-

21.6.19 アクセプタンスマスクレジスタ 0/1(AMR0/AMR1)

AMR0 および AMR1 の 2 つのアクセプタンスマスクレジスタがありますが、両者ともに標準フレームフォーマットまたは拡張フレームフォーマットのいずれかで利用できます。

AM28 ~ AM18(11 ビット) は標準フレームフォーマットにおけるアクセプタンスマスクに使用され AM28 ~ AM0(29 ビット) は拡張フォーマットでアクセプタンスマスクに使用されます。

■ アクセプタンスマスクレジスタ 0/1(AMR0/AMR1)

アクセプタンスマスクレジスタ 0/1(AMR0/AMR1) のビット構成を図 21.6-21 に示します。

図 21.6-21 アクセプタンスマスクレジスタ 0/1(AMR0/AMR1)

アドレス : 003B14 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D14 _H (CAN1)	AM28	AM27	AM26	AM25	AM24	AM23	AM22	AM21	AMR0 バイト 0
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003B15 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D15 _H (CAN1)	AM20	AM19	AM18	AM17	AM16	AM15	AM14	AM13	AMR0 バイト 1
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003B16 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D16 _H (CAN1)	AM12	AM11	AM10	AM9	AM8	AM7	AM6	AM5	AMR0 バイト 2
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003B17 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D17 _H (CAN1)	AM4	AM3	AM2	AM1	AM0				AMR0 バイト 3
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(-)	(-)	(-)	
初期値	(X)	(X)	(X)	(X)	(X)	(-)	(-)	(-)	
アドレス : 003B18 _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D18 _H (CAN1)	AM28	AM27	AM26	AM25	AM24	AM23	AM22	AM21	AMR1 バイト 0
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003B19 _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D19 _H (CAN1)	AM20	AM19	AM18	AM17	AM16	AM15	AM14	AM13	AMR1 バイト 1
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

(続く)

(続き)

アドレス : 003B1A _H (CAN0)	7	6	5	4	3	2	1	0	ビット No.
003D1A _H (CAN1)	AM12	AM11	AM10	AM9	AM8	AM7	AM6	AM5	AMR1 バイト 2
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003B1B _H (CAN0)	15	14	13	12	11	10	9	8	ビット No.
003D1B _H (CAN1)	AM4	AM3	AM2	AM1	AM0				AMR1 バイト 3
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(-)	(-)	(-)	
初期値	(X)	(X)	(X)	(X)	(X)	(-)	(-)	(-)	

● 0: コンペア

本ビットに対応するアクセプタンスコード (受信メッセージ ID と比較するための ID レジスタ IDR_x) を受信メッセージ ID のビットと比較します。マッチしなければメッセージは受信されません。

● 1: マスク

本ビットに対応するアクセプタンスコード ID レジスタ (IDR_x) をマスクします。受信メッセージ ID のビットとのコンペアは行われません。

< 注意事項 >

- AMR0 および AMR1 は AMR0 および AMR1 を選択するメッセージバッファ (x) のすべてが無効 (メッセージバッファ有効レジスタ (BVALR) の BVAL_x が "0") であるときに設定しなければなりません。バッファが有効 (BVAL_x = 1) であるときに設定を行うと不必要な受信メッセージが格納される可能性があります。
- CAN コントローラが CAN 通信に参加している状態 (CSR: HALT ビットの読出し値が "0" で、CAN コントローラが CAN バスの通信に参加しており、送受信が可能な状態) で、メッセージバッファを無効 (BVALR:BVAL = 0) にする場合は、「21.14 CAN コントローラの使用上の注意」に従ってください。

21.6.20 メッセージバッファ

メッセージバッファは 16 個あり, 1 つのメッセージバッファ $x(x=0 \sim 15)$ は ID レジスタ (IDRx), DLC レジスタ (DLCRx) およびデータレジスタ (DTRx) から構成されています。

■ メッセージバッファ

- メッセージバッファ (x) は送信および受信の両方に使用されます。
 - 下位番号のメッセージバッファの方が高い優先順位を割り当てられています。
 - 送信時に送信要求が 1 つ以上のメッセージバッファに対してためられたときには最下位番号のメッセージバッファから開始して送信が行われます (「21.7 CAN コントローラの送信」を参照)。
 - 受信時に 1 つ以上のメッセージバッファのアクセプタンスフィルタ (受信メッセージのアクセプタンスマスク ID とメッセージバッファとを比較する機構) を受信メッセージ ID が通過するときに受信メッセージは最下位番号のメッセージバッファ内に格納されます (「21.9 CAN コントローラの受信フローチャート」を参照)。
 - 同一のアクセプタンスフィルタが 1 つ以上のメッセージバッファ内に設定されたときにメッセージバッファはマルチレベルメッセージバッファとして使用することができます。これにより受信時間に対する余裕が与えられます (「21.13 マルチレベルメッセージバッファの構成の決定」を参照)。
-

< 注意事項 >

- メッセージバッファおよび汎用 RAM 領域への書込み動作は, ワード単位でイーブンアドレスに対して行わなければなりません。バイト単位で書込み動作を行うと下位バイトへの書込み時に上位バイトに不定データを書き込んでしまいます。上位バイトへの書込みは無視されます。
 - メッセージバッファ有効レジスタ (BVALR) の BVALx ビットが "0" (無効) であるときにメッセージバッファ x (IDRx, DLCRx および DTRx) は汎用 RAM として使用することができます。ただし, CAN コントローラが送受信動作中は, メッセージバッファを使用しますので, CPU アクセスが最大で 64 マシンサイクル待たされることがあります。これは汎用 RAM 領域 (アドレス 003A00_H ~ 003C1F_H およびアドレス 003D00_H ~ 003D1F_H) についても同様です。
-

21.6.21 ID レジスタ x(x=0 ~ 15)(IDRx)

ID レジスタ x(x=0 ~ 15)(IDRx) は、メッセージバッファ (x) に対する ID レジスタです。

■ ID レジスタ x(x=0 ~ 15)(IDRx)

ID レジスタ x(x=0 ~ 15)(IDRx) のビット構成を図 21.6-22 に示します。

図 21.6-22 ID レジスタ x(x=0 ~ 15)(IDRx)

アドレス : 003A20 _H +4x(CAN0) 003C20 _H +4x(CAN1)	7	6	5	4	3	2	1	0	ビット No. IDRx バイト 0
	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A21 _H +4x(CAN0) 003C21 _H +4x(CAN1)	15	14	13	12	11	10	9	8	ビット No. IDRx バイト 1
	ID20	ID19	ID18	ID17	ID16	ID15	ID14	ID13	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A22 _H +4x(CAN0) 003C22 _H +4x(CAN1)	7	6	5	4	3	2	1	0	ビット No. IDRx バイト 2
	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A23 _H +4x(CAN0) 003C23 _H +4x(CAN1)	15	14	13	12	11	10	9	8	ビット No. IDRx バイト 3
	ID4	ID3	ID2	ID1	ID0				
リード / ライト	(R/W)	(R/W)	(R/W)	(R/W)	(R/W)	(-)	(-)	(-)	
初期値	(X)	(X)	(X)	(X)	(X)	(-)	(-)	(-)	

標準フレームフォーマット (IDE レジスタ (IDER) の IDE_x = 0) でメッセージバッファ (x) を使用しているときは ID28 ~ ID18 の 11 ビットを使用してください。拡張フレームフォーマット (IDE_x = 1) でバッファを使用しているときは ID28 ~ ID0 の 29 ビットを使用してください。

ID28 ~ ID0 の機能は次のとおりです。

- アクセプタンスコード設定
受信メッセージ ID との比較用の ID
- 送信メッセージ ID の設定
- 受信メッセージ ID の格納

< 注意事項 >

- 標準フレームフォーマットでは ID28 ~ ID22 のビットのすべてに "1" をセットすることは禁止されています。
 - 受信メッセージ ID はアクセプタンスマスクでマスクされているビットにも格納されません。標準フレームフォーマットでは ID17 ~ ID0 は不定値 (受信シフトレジスタに残されている前回受信したメッセージの一部) が格納されます。
 - ID レジスタ x ($x = 0 \sim 15$) (IDRx) への書き込み動作はワード単位で行わなければなりません。バイト単位で書き込み動作を行うと下位バイトへの書き込み時に上位バイトに不定データを書き込んでしまいます。上位バイトへの書き込みは無視されます。
 - 本レジスタはメッセージバッファ (x) が無効 (メッセージバッファ有効レジスタ (BVALR) の BVALx が "0") であるときに設定されなければなりません。バッファが有効 ((BVALx = 1) であるときに設定を行うと不必要な受信メッセージが格納されてしまいます。
 - CAN コントローラが CAN 通信に参加している状態 (CSR: HALT ビットの読出し値が "0" で、CAN コントローラが CAN バスの通信に参加しており送受信が可能な状態) で、メッセージバッファを無効 (BVALR: BVAL = 0) にする場合は、「21.14 CAN コントローラの使用上の注意」に従ってください。
-

21.6.22 DLC レジスタ x(x=0 ~ 15)(DLCRx)

これはメッセージバッファ x に対する DLC を格納します。

■ DLC レジスタ x(x = 0 ~ 15)(DLCRx)

DLC レジスタ x(x = 0 ~ 15)(DLCRx) のビット構成を図 21.6-23 に示します。

図 21.6-23 DLC レジスタ x(x=0 ~ 15)(DLCRx)

アドレス :	7	6	5	4	3	2	1	0	ビット No.
003A60 _H +2x(CAN0)	-	-	-	-	DLC3	DLC2	DLC1	DLC0	DLCRx
003C60 _H +2x(CAN1)	-	-	-	-	(R/W)	(R/W)	(R/W)	(R/W)	バイト 0
リード / ライト	(-)	(-)	(-)	(-)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(-)	(-)	(-)	(-)	(X)	(X)	(X)	(X)	

● 送信

- データフレームが送信する (送信 RTR レジスタ (TRTRR) の TRTRx が "0") ときに送信メッセージのデータ長 (バイト単位) を設定します。
- リモートフレームが送信する (TRTRx = 1) ときに要求メッセージのデータ長 (バイト単位) を設定します。

< 注意事項 >

0000_B ~ 1000_B(0 ~ 8 バイト) 以外の設定は禁止されています。

● 受信

- データフレームが受信された (リモートフレーム要求受信レジスタ (RRTRR) の RRTRx が "0") ときに受信メッセージのデータ長 (バイト単位) を格納します。
- リモートフレームが受信された (RRTRx = 1) ときに要求メッセージのデータ長 (バイト単位) を格納します。

< 注意事項 >

本レジスタへの書込み動作はワード単位で行わなければなりません。バイト単位で書込み動作を行うと下位バイトへの書込み時に上位バイトに不定データを書き込んでしまいます。上位バイトへの書込みは無視されます。

21.6.23 データレジスタ x(x=0 ~ 15)(DTRx)

データレジスタ x(x = 0 ~ 15) (DTRx) は、メッセージバッファ (x) に対するデータレジスタです。

本レジスタはデータフレームの送信および受信のみ使用されるものであり、リモートフレームの送信および受信には使用されません。

■ データレジスタ x(x = 0 ~ 15) (DTRx)

データレジスタ x(x = 0 ~ 15) (DTRx) のビット構成を図 21.6-24 に示します。

図 21.6-24 データレジスタ x(x=0 ~ 15)(DTRx)

アドレス : 003A80 _H +8x(CAN0) 003C80 _H +8x(CAN1)	7	6	5	4	3	2	1	0	ビット No. DTRx バイト 0
	D7	D6	D5	D4	D3	D2	D1	D0	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A81 _H +8x(CAN0) 003C81 _H +8x(CAN1)	15	14	13	12	11	10	9	8	ビット No. DTRx バイト 1
	D7	D6	D5	D4	D3	D2	D1	D0	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A82 _H +8x(CAN0) 003C82 _H +8x(CAN1)	7	6	5	4	3	2	1	0	ビット No. DTRx バイト 2
	D7	D6	D5	D4	D3	D2	D1	D0	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A83 _H +8x(CAN0) 003C83 _H +8x(CAN1)	15	14	13	12	11	10	9	8	ビット No. DTRx バイト 3
	D7	D6	D5	D4	D3	D2	D1	D0	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A84 _H +8x(CAN0) 003C84 _H +8x(CAN1)	7	6	5	4	3	2	1	0	ビット No. DTRx バイト 4
	D7	D6	D5	D4	D3	D2	D1	D0	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A85 _H +8x(CAN0) 003C85 _H +8x(CAN1)	15	14	13	12	11	10	9	8	ビット No. DTRx バイト 5
	D7	D6	D5	D4	D3	D2	D1	D0	
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

(続く)

(続き)

アドレス : 003A86 _H +8x(CAN0)	7	6	5	4	3	2	1	0	ビット No.
003C86 _H +8x(CAN1)	D7	D6	D5	D4	D3	D2	D1	D0	DTRx バイト 6
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	
アドレス : 003A87 _H +8x(CAN0)	15	14	13	12	11	10	9	8	ビット No.
003C87 _H +8x(CAN1)	D7	D6	D5	D4	D3	D2	D1	D0	DTRx バイト 7
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	(X)	(X)	(X)	(X)	(X)	(X)	(X)	(X)	

● 送信メッセージデータ (0 ~ 8 バイトのいずれか) の設定

データは MSB で開始して BYTE0, BYTE1, ... BYTE7 の順で送信されます。

● 受信メッセージデータ

データは MSB で開始して BYTE0, BYTE1, ... BYTE7 の順で格納されます。

受信メッセージデータが 8 バイトより少なくてもデータが格納されるデータレジスタ (DTRx) の残りのバイトは不定となります。

< 注意事項 >

本レジスタへの書込み動作はワード単位で行わなければなりません。バイト単位で書込み動作を行うと下位バイトへの書込み時に上位バイトに不定データを書き込んでしまいます。上位バイトへの書込みは無視されます。

21.7 CAN コントローラの送信

CAN コントローラでは、送信要求レジスタ (TREQR) の TREQx に "1" が書き込まれたときにメッセージバッファ (x) による送信が開始します。このときに TREQx は "1" になり、送信完了レジスタ (TCR) の TCx は "0" になります。

■ CAN コントローラの送信の開始

リモートフレーム受信待機レジスタ (RFTWR) の RFTWx が "0" であれば送信は直ちに開始します。もし RFTWx が "1" であれば、リモートフレームが受信される (リモート要求受信レジスタ (RRTRR) の RRTRx が "1" になる) まで待機した後に送信が開始します。もし送信に対する要求が 1 つ以上のメッセージバッファに対してためされた (1 つ以上の TREQx が "1") ときには、最下位番号のメッセージバッファから開始して送信が行われます。

CAN バスに対するメッセージ送信 (送信出力端子 TX によるもの) はバスがアイドルのときに開始します。

送信 RTR レジスタ (TRTRR) の TRTRx が "0" であれば、データフレームが送信されます。もし TRTRx が "1" であれば、リモートフレームが送信されます。

送信に関してメッセージバッファが CAN バス上のほかの CAN コントローラと競合してアービトレーションが不調に終わった場合または送信中にエラーが生じた場合、メッセージバッファはバスがアイドルとなるまで待機しそれが成功するまで再送信を反復します。

■ CAN コントローラの送信要求の解除

● 送信解除レジスタ (TCANR) による解除

送信待ち状態中に送信を実行しなかったメッセージバッファ (x) に対する送信要求は送信解除レジスタ (TCANR) の TCANx に "1" を書き込むことにより解除することができます。解除の完了時に TREQx は "0" になります。

● 受信メッセージの格納による解除

送信要求にもかかわらず送信を実行しなかったメッセージバッファ (x) も受信を行います。

データフレームの送信に対する要求にもかかわらずメッセージバッファ (x) が送信を実行していなかった場合 (TRTRx = 0 または TREQx = 1) でも、アクセプタンスフィルタを通過した受信データフレームを格納した後は、送信要求が解除されます (TREQx = 0)。

< 注意事項 >

リモートフレームを格納することでは送信要求は解除されません (TREQx = 1 はそのままです)。

リモートフレームの送信に対する要求にもかかわらずメッセージバッファ (x) が送信を実行していなかった場合 (TRTRx = 1 または TREQx = 1) でも、アクセプタンスフィルタを通過した受信リモートフレームを格納した後は、送信要求が解除されます (TREQx = 0)。

< 注意事項 >

データフレームまたはリモートフレームのいずれを格納しても，送信要求は解除されません。

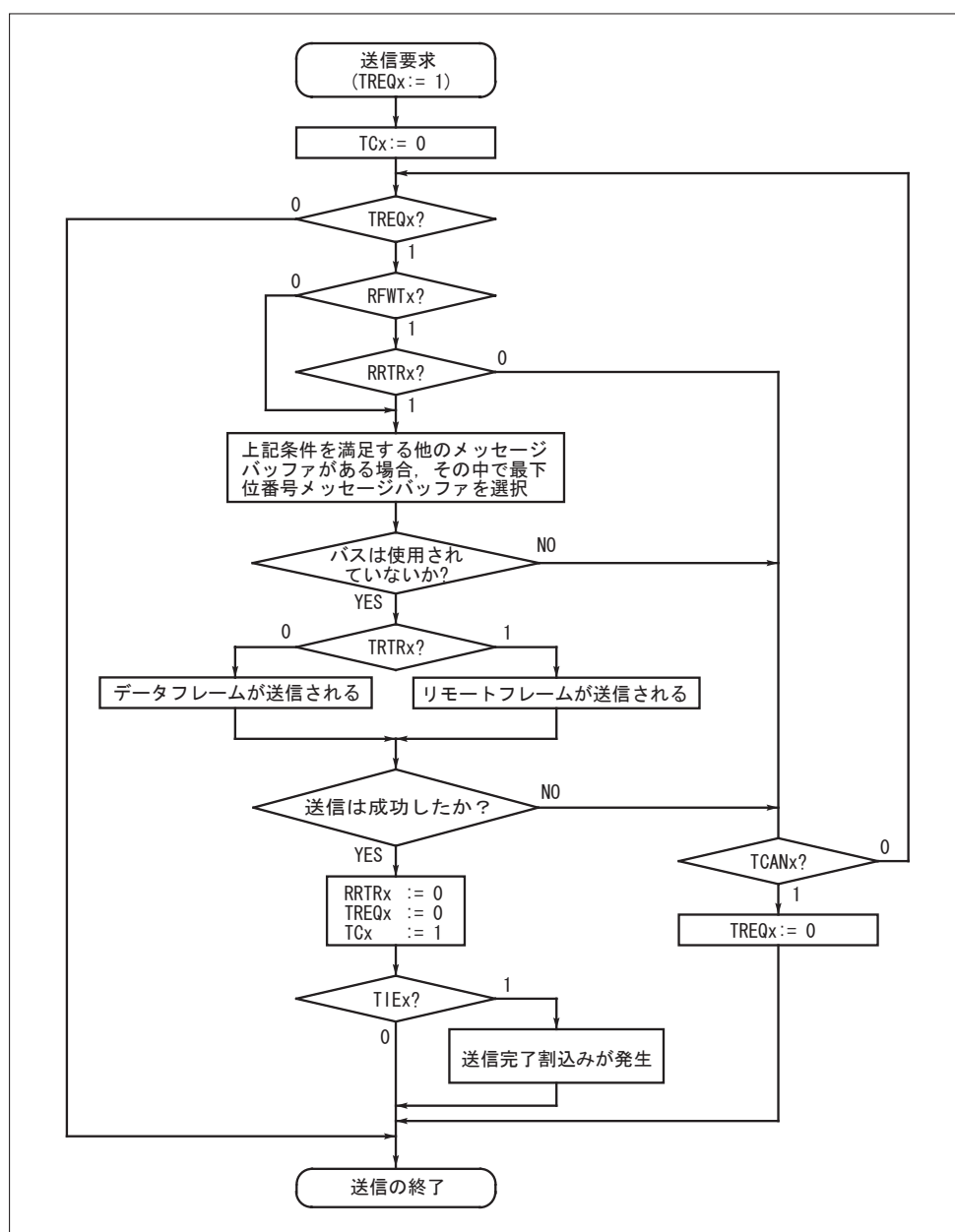
■ CAN コントローラの送信の完了

送信が成功したときに RRTRx は "0" になり TREQx は "0" になり送信完了レジスタ (TCR) の TCx は "1" になります。もし送信完了割込みが許可されていれば (送信完了割込み許可レジスタ (TIER) の TIEx は "1") 割込みが生じます。

■ CAN コントローラの送信フローチャート

図 21.7-1 に，CAN コントローラの送信フローチャートを示します。

図 21.7-1 CAN コントローラの送信フローチャート



21.8 CAN コントローラの受信

CAN バス上でデータフレームまたはリモートフレームの開始 (SOF) が検出されたときに受信は開始します。

■ アクセプタンスフィルタリング

標準フレームフォーマットの受信メッセージは、標準フレームフォーマットに設定されたメッセージバッファ (x) (IDE レジスタ (IDER) の IDE_x は "0") と比較されます。拡張フレームフォーマットの受信メッセージは、拡張フレームフォーマットに設定されたメッセージバッファ (x) (IDE_x は "1") と比較されます。

もし受信メッセージ ID とアクセプタンスコード (受信メッセージ ID との比較のための ID レジスタ (IDR_x)) との比較の後にアクセプタンスマスクにより比較すべきビットセットのすべてが一致すれば受信メッセージはメッセージバッファ (x) のアクセプタンスフィルタを通過します。

■ 受信メッセージの格納

受信動作が成功したときに受信メッセージはアクセプタンスフィルタを通過した ID を含めメッセージバッファ (x) に格納されます。

データフレームを受信したときに受信メッセージは ID レジスタ (IDR_x), DLC レジスタ (DLCR_x) およびデータレジスタ (DTR_x) に格納されます。

受信メッセージデータが 8 バイトより少なくてもデータは DTR_x の残りのバイトに格納されその値は不定となります。

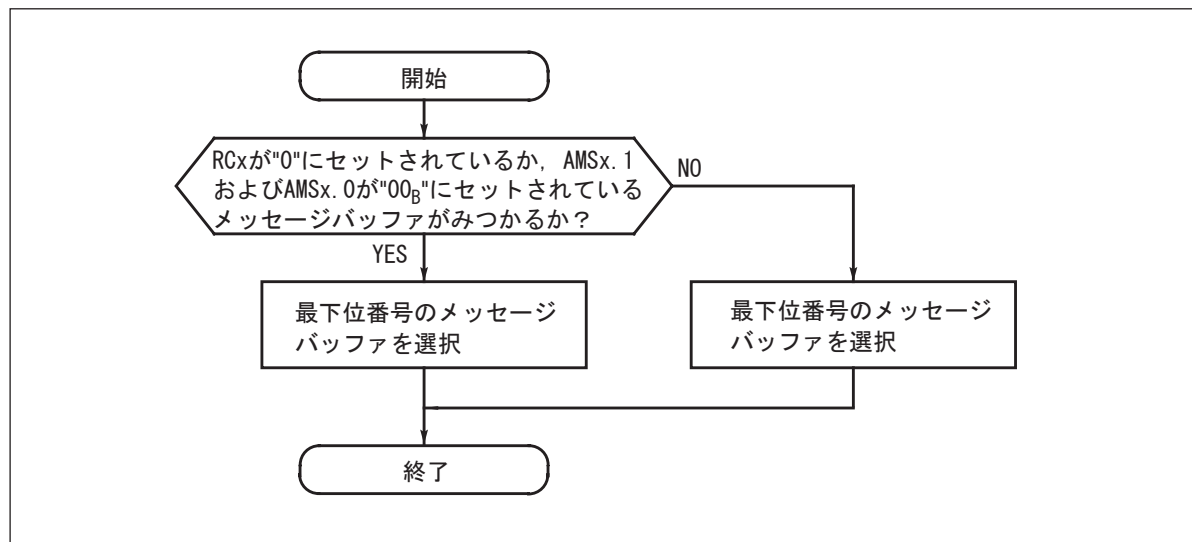
リモートフレームを受信したときに受信メッセージは IDR_x および DLCR_x のみに格納されて DTR_x はそのままです。

もしアクセプタンスフィルタを通過した ID を含む 1 つ以上のメッセージバッファがあれば受信メッセージが格納されるべきメッセージバッファ x は次の規則により決定されます。

- メッセージバッファ x (x=0 ~ 15) の優先順位はその番号が低いほど高くなります。すなわちメッセージバッファ 0 が最高の優先度を与えられメッセージバッファ 15 は最低の優先度を与えられます。
- 受信メッセージを格納する上では基本的に受信完了レジスタ (RCR) の RC_x ビットが "0" に設定されたメッセージバッファが優先されます。
- もしアクセプタンスマスク選択レジスタ (AMSR) の各ビットが全ビット比較 (AMS_{x.1} および AMS_{x.0} ビットが "00" に設定されたメッセージバッファ) に設定されていれば RCR の RC_x ビットの値にかかわらず受信メッセージは格納されます。
- もし RCR の RC_x ビットが "0" で、かつ AMSR のビットが全ビット比較に設定されているバッファが複数ある場合、受信メッセージは最下位番号 (最優先) のメッセージバッファ x に格納されます。
- もし上記条件に合致するメッセージバッファが無ければ、受信メッセージはアクセプタンスフィルタを通過した ID を持つ最下位番号のメッセージバッファに格納され、オーバランが発生 (ROVRR: ROVR_x = 1) します。

図 21.8-1 には、受信メッセージを格納すべきメッセージバッファ x を決定するためのフローチャートが示されています。メッセージバッファはバッファ番号順に、AMSR の各ビットが全ビット比較にセットされたメッセージバッファ、AMR0 または AMR1 を使用しているメッセージバッファおよび AMSR の各ビットが全ビットマスクにセットされたメッセージバッファの順番に設定されることを推奨します。

図 21.8-1 受信メッセージを格納するメッセージバッファ (x) を決定するフローチャート



■ 受信オーバーラン

受信メッセージが格納されるべきメッセージバッファ x に対応する受信完了レジスタ (RCR) の RC_x ビットが既に "1" にセットされていてメッセージバッファ x の受信メッセージの格納が完了したときに受信オーバーランレジスタ (ROVRR) の $ROVR_x$ ビットは "1" にセットされて受信オーバーランを示します。

■ データフレームおよびリモートフレームの受信に対する処理

● データフレームの受信に対する処理

リモート要求受信レジスタ (RRTRR) の $RRTR_x$ は "0" になります。

送信要求レジスタ (TREQR) の $TREQ_x$ は (受信メッセージを格納する直前に) "0" になります。送信を実行しなかったメッセージバッファ (x) に対する送信要求は解除されます。

< 注意事項 >

データフレームまたはリモートフレームのいずれの送信に対する要求も解除されます。

● リモートフレームの受信に対する処理

RRTR_x は "1" になります。

もし送信 RTR レジスタ (TRTRR) の TRTR_x が "1" であれば TREQ_x は "0" になります。
この結果、送信を実行しなかったメッセージバッファへのリモートフレームの送信に対する要求は解除されます。

< 注意事項 >

- データフレーム送信に対する要求は解除されません。
 - 送信要求の解除に関しては「21.7 CAN コントローラの送信」を参照してください。
-

■ 受信完了

受信完了レジスタ (RCR) の RC_x は受信メッセージを格納した後に "1" になります。

もし受信割込みが許可されていれば (受信割込み許可レジスタ (RIER) の RIE_x が "1")
割込みが生じます。

< 注意事項 >

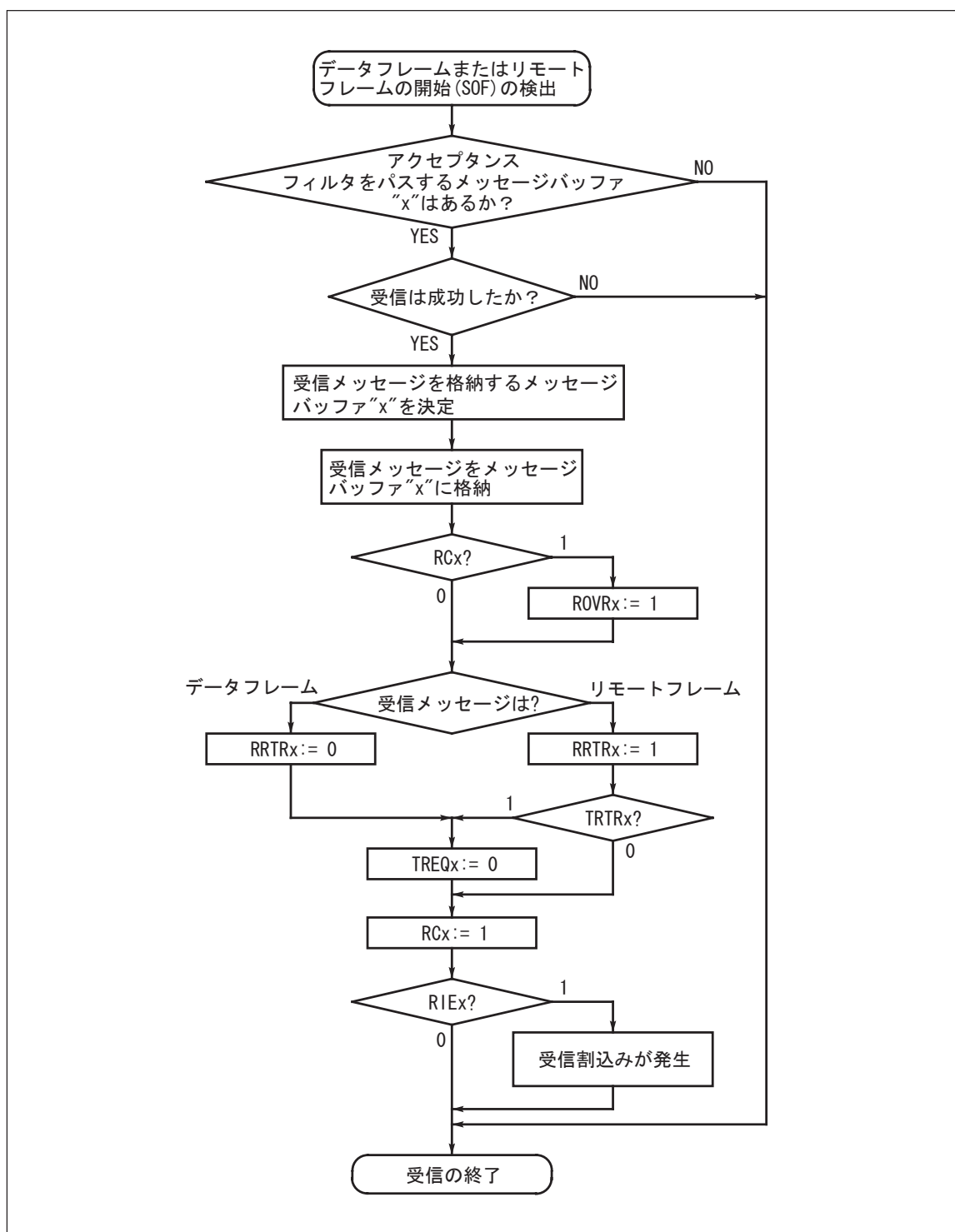
この CAN コントローラはそれ自身が送信したメッセージを受信することはありません。

21.9 CAN コントローラの受信フローチャート

図 21.9-1 に、CAN コントローラの受信フローチャートを示します。

■ CAN コントローラの受信フローチャート

図 21.9-1 CAN コントローラの受信フローチャート



21.10 CAN コントローラの使用法

CAN コントローラでは、次のような設定が必要です。

- ビットタイミングの設定
 - フレームフォーマットの設定
 - ID の設定
 - アクセプタンスフィルタの設定
 - 低消費電力モードの設定
-

■ ビットタイミングの設定

ビットタイミングレジスタ (BTR) はバス動作停止 (制御ステータスレジスタ (CSR) のバス動作停止ビット (HALT) が "1") の間に設定しなければなりません。

設定完了の後に HALT に "0" を書き込んでバス動作停止を解除してください。

■ フレームフォーマットの設定

メッセージバッファ (x) により使用されるフレームフォーマットを設定してください。標準フレームフォーマットを使用するときには IDE レジスタ (IDER) の IDE_x を "0" に設定してください。拡張フレームフォーマットを使用するときには IDE_x を "1" に設定してください。

この設定はメッセージバッファ(x)が無効 (メッセージバッファ有効レジスタ (BVALR) の BVAL_x が "0") であるときに行わなければなりません。バッファが有効 (BVAL_x = 1) のときに設定を行うと不必要な受信メッセージが格納される可能性があります。

■ ID の設定

メッセージバッファ (x) の ID を ID レジスタ (IDR_x) の ID28 ~ ID0 に設定してください。メッセージバッファ (x) の ID は標準フレームフォーマットでは ID17 ~ ID0 に設定する必要はありません。メッセージバッファ (x) の ID は送信では送信メッセージとして使用され受信ではアクセプタンスコードとして使用されます。

この設定はメッセージバッファ(x)が無効 (メッセージバッファ有効レジスタ (BVALR) の BVAL_x が "0") であるときに行わなければなりません。バッファが有効 (BVAL_x = 1) のときに設定を行うと不必要な受信メッセージが格納される可能性があります。

■ アクセプタンスフィルタの設定

メッセージバッファ(x)のアクセプタンスフィルタはアクセプタンスコードおよびアクセプタンスマスク設定により設定されます。それはアクセプタンスメッセージバッファ(x)が無効(メッセージバッファ許可レジスタ(BVALR)のBVALxが"0")であるときに設定しなければなりません。バッファが有効(BVALx = 1)であるときに設定を行うと不必要な受信メッセージが格納される可能性があります。

各メッセージバッファ(x)で使用されたアクセプタンスマスクをアクセプタンスマスク選択レジスタ(AMSR)により設定してください。アクセプタンスマスクレジスタ(AMR0およびAMR1)も使用するのであれば設定しなければなりません(設定の詳細に関しては「21.6.18 アクセプタンスマスク選択レジスタ(AMSR)」および「21.6.19 アクセプタンスマスクレジスタ 0/1(AMR0/AMR1)」を参照)。

アクセプタンスマスクは不必要な受信メッセージが格納されたときでも送信要求が解除されないように設定しなければなりません。例えば、送信 ID と同一の ID のメッセージのみを受信するのであればフルビット比較に設定しなければなりません。

■ 低電力モードの設定

F²MC-16LX を低電力モード(停止, 時計, ハードウェアスタンバイなど)に設定するには, 制御ステータスレジスタ(CSR)のバス動作停止ビット(HALT)に"1"を書き込んでからバス動作が停止したかどうか(HALT = 1)をチェックしてください。

21.11 メッセージバッファ (x) による送信方法

ビットタイミングの設定, フレームフォーマットの設定, ID の設定およびアクセプタンスフィルタの設定が完了した後で, BVALx を "1" に設定してメッセージバッファ (x) を有効にしてください。

■ メッセージバッファ (x) による送信方法

● 送信データ長コードの設定

送信データ長コード (バイト単位) を DLC レジスタ (DLCRx) の DLC3 ~ DLC0 に設定してください。

データフレーム送信 (送信 RTR レジスタ (TRTRR) の TRTRx が "0" のとき) に対しては送信メッセージのデータ長を設定してください。

リモートフレーム送信 (TRTRx = 1 のとき) に対しては要求メッセージのデータ長 (バイト単位) を設定してください。

< 注意事項 >

"0000_B" ~ "1000_B" (0 ~ 8 バイト) 以外の設定は禁止されています。

● 送信データの設定 (データフレームの送信に対してのみ)

データフレーム送信 (送信レジスタ (TRTRR) の TRTRx が "0" のとき) に対してはデータを送信バイト数分データレジスタ (DTRx) に設定してください。

< 注意事項 >

送信データは送信要求レジスタ (TREQR) の TREQx ビットを "0" に設定して再書き込みしなければなりません。メッセージバッファ有効レジスタ (BVALR) の BVALx ビットを "0" に設定する必要はありません。BVALx ビットを "0" に設定するとリモートフレーム受信を喪失する可能性があります。

● 送信 RTR レジスタの設定

データフレーム送信に対しては送信 RTR レジスタ (TRTRR) の TRTRx を "0" に設定してください。

リモートフレーム送信に対しては TRTRx を "1" に設定してください。

● 送信開始条件の設定 (データフレームの送信のみ)

データフレーム送信に対する要求が設定された直後に送信を開始するためには、リモートフレーム受信待機レジスタ (RFWTR) の RFWTx を "0" に設定してください (送信要求レジスタ (TREQR) の TREQx は "1" であり送信 RTR レジスタ (TRTRR) の TRTRx は "0" です)。

データフレーム送信に対する要求が設定された (TREQx = 1 および TRTRx = 0) 後にリモートフレームを受信する (リモート要求受信レジスタ (RRTRR) の RRTRx が "1" になる) まで待機した後に送信を開始するためには、RFWTx を "1" に設定してください。

< 注意事項 >

RFWTx が "1" に設定されるとリモートフレーム送信を行うことはできません。

● 送信完了割込みの設定

送信完了割込みを発生する場合には送信完了割込み許可レジスタ (TIER) の TIEx を "1" に設定してください。送信完了割込みを発生しない場合には TIEx を "0" に設定してください。

● 送信要求の設定

送信要求を行うには送信要求レジスタ (TREQR) の TREQx を "1" に設定してください。

● 送信要求の解除

メッセージバッファ(x)に対する送信の要求を解除するときには送信キャンセルレジスタ (TCANR) の TCANx に "1" を書き込んでください。

TREQx をチェックしてください。TREQx = 0 の場合には送信解除が終了したか送信が完了しています。送信完了レジスタ (TCR) の TCx をチェックしてください。TCx = 0 の場合には送信解除は終了しています。TCx = 1 ならば送信が完了しています。

● 送信完了の処理

送信が成功すれば送信完了レジスタ (TCR) の TCx は "1" になります。

もし送信完了割込みが許可されていれば (送信完了割込み許可レジスタ (TIER) の TIEx が 1) 割込みが生じます。

送信完了をチェックした後に TCx に "0" を書き込んでクリアしてください。これにより送信完了割込みが解除されます。

次の場合には待ち状態中の送信要求はメッセージの受信および格納により解除されません。

- データフレームの受信によるデータフレーム送信の要求
- データフレームの受信によるリモートフレーム送信の要求
- リモートフレームの受信によるリモートフレーム送信の要求

データフレーム送信に対する要求はリモートフレームの受信および格納によっては解除されません。ただし ID および DLC は受信したリモートフレームの ID および DLC により変更されます。送信されるべきデータフレームの ID および DLC が受信したリモートフレームの値になることに注意してください。

21.12 メッセージバッファ (x) による受信方法

ビットタイミングの設定, フレームフォーマットの設定, ID の設定およびアクセプタンスフィルタの設定の後で, 以下の設定をしてください。

■ メッセージバッファ (x) による受信方法

● 受信割込み設定

受信割込みを許可するには受信割込み許可レジスタ (RIER) の RIE_x を "1" に設定してください。

受信割込みを禁止するには RIE_x を "0" に設定してください。

● 受信開始

設定の後で受信を開始するときにはメッセージバッファ有効レジスタ (BVALR) の BVAL_x を "1" に設定してメッセージバッファ (x) を有効にしてください。

● 受信完了の処理

アクセプタンスフィルタへの通過後に受信が成功すれば, 受信メッセージはメッセージバッファ (x) に格納されて, 受信完了レジスタ (RCR) の RC_x は "1" になります。データフレーム受信に関してはリモート要求受信レジスタ (RRTRR) の RRTR_x は "0" になります。リモートフレーム受信に関しては, RRTR_x は "1" になります。

もし, 受信割込みが許可 (受信割込み許可レジスタ (RIER) の RIE_x が "1") されれば割込みが生じます。

受信完了をチェックした後 (RC_x=1) で受信メッセージを処理してください。

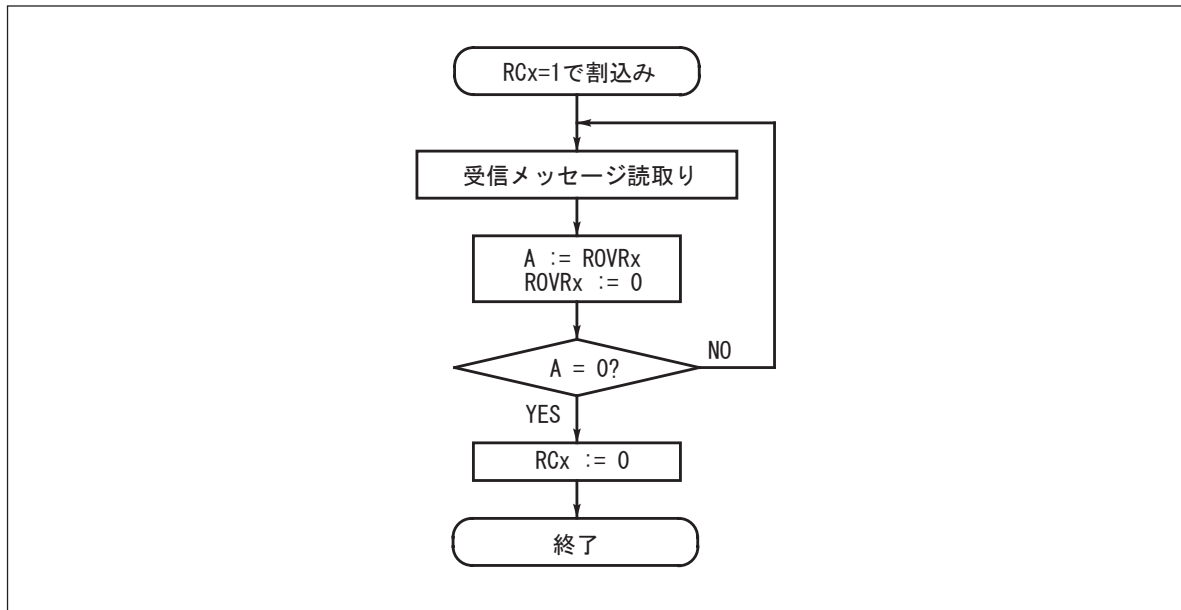
受信メッセージの処理を完了した後は受信オーバーランレジスタ (ROVRR) の ROVR_x をチェックしてください。

ROVR_x = 0 ならば処理された受信メッセージは有効です。CR_x に "0" を書き込んでそれを "0" に設定して (受信完了割込みも解除されます) 受信を終了します。

ROVR_x=1 ならば受信オーバーランが生じ, 処理された受信メッセージに新たな受信メッセージが上書きされている可能性があります。この場合には ROVR_x ビットに "0" を書き込んでそれを "0" に設定した後で受信メッセージを再処理しなければなりません。

図 21.12-1 に, 受信割込み処理の一例を示します。

図 21.12-1 受信割り込み処理の例



21.13 マルチレベルメッセージバッファの構成の決定

もし受信が頻繁に行われる場合または不特定多数のメッセージを受信するのであれば、すなわちメッセージを処理する時間が不十分であれば、1 つ以上のメッセージバッファを組み合わせるマルチレベルメッセージバッファとすることにより CPU による受信メッセージの処理に対する余裕を与えてください。

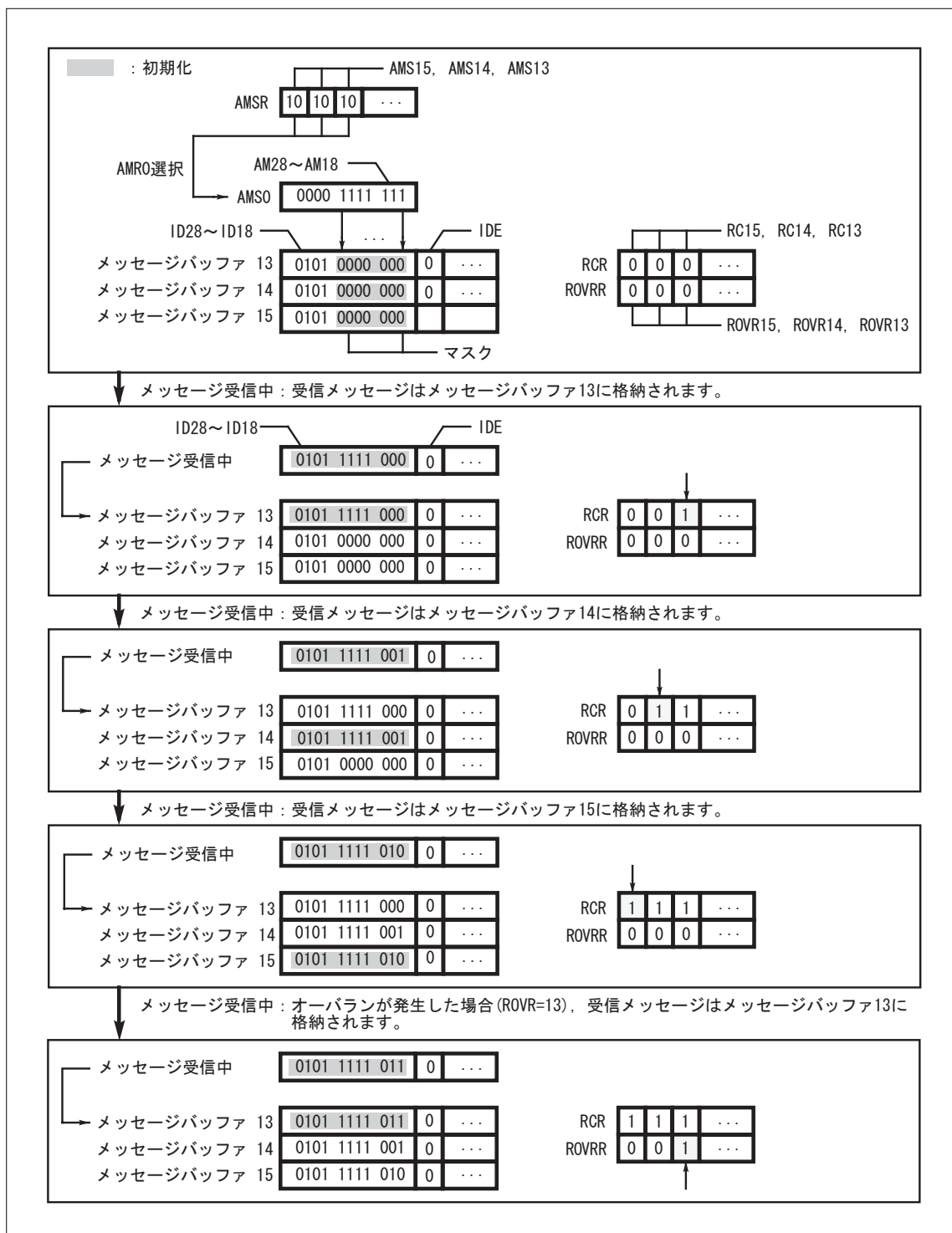
■ マルチレベルメッセージバッファの構成の決定

マルチレベルメッセージバッファを用意するには組み合わせられた各メッセージバッファに同じアクセプタンスフィルタを設定しなければなりません。

もし、アクセプタンスマスク選択レジスタ (AMSR) の各ビットが全ビット比較 ($(\text{AMSx.1}, \text{AMSx.0}) = (0, 0)$) に設定されていればメッセージバッファをマルチレベルメッセージ構成とすることはできません。これは、全ビット比較では受信完了レジスタ (RCR) の RCx ビットの値にかかわらず受信メッセージが格納されることから、1 つ以上のメッセージバッファに対して全ビット比較および同一アクセプタンスコード (ID レジスタ (IDRx)) が指定されたとしても、受信メッセージが常に下位番号の (優先度の低い) メッセージバッファに格納されるからです。このため、全ビット比較および同一アクセプタンスコードを 1 つ以上のメッセージバッファに指定してはなりません。

図 21.13-1 は、マルチレベルメッセージバッファの動作例を示しています。

図 21.13-1 マルチレベルメッセージバッファの動作例



< 注意事項 >

4 個のメッセージが同一のアクセプタンスフィルタ設定によりメッセージバッファ 13, 14 および 15 に受信されます。

21.14 CAN コントローラの使用上の注意

CAN コントローラを使用する場合は、以下の点に注意してください。

■ BVAL ビットによるメッセージバッファ禁止時の注意

メッセージバッファの内容の読出しや書込みの処理を行うために BVAL ビットを用いてメッセージバッファを禁止すると、CAN コントローラの送受信が正常に行われない可能性があります。本注意事項は、この現象を回避する方法を説明しています。

● 発生条件

以下の 2 つの条件を同時に満たす場合に、CAN コントローラの送受信が正常に行われない可能性があります。

- CAN コントローラが CAN 通信に参加している状態、つまり HALT ビットの読出し値が "0" で、CAN コントローラが CAN バスの通信に参加しており送受信が可能な状態。
- BVAL ビットの設定でメッセージバッファを禁止して、メッセージバッファの内容へ読出しや書込みの処理を行う。

● 処置

受信メッセージバッファの構成を変更する場合の操作

CAN 通信が確立されている間 (HALT ビットの読出し値が "0" で、CAN コントローラが CAN バスの通信に参加しており送受信が可能な状態) で、D レジスタや AMS レジスタ、AMR0/AMR1 レジスタの設定変更により、メッセージバッファの構成の変更を行う場合は、以下のいずれかの手段を使用してください。

- HALT ビットの使用

HALT ビットに "1" を書き込み、その後このビットが "1" になったことを確認した後 ID/AMS/AMR0/AMR1 レジスタの設定を変更してください。

- メッセージバッファ 0 を使用しない

メッセージバッファ 0 を送信にも受信にも使用しないでください。すなわち、メッセージバッファを禁止し (BVAL0 = 0)、受信割込みを禁止 (RIE0 = 0)、送信要求を行わない (TREQ0 = 0) してください。

受信メッセージの処理を行う場合の操作

次にくるメッセージによる上書き動作の防止のために、BVAL ビットによる受信禁止を行わないでください。メッセージの上書き有無の確認は、ROVR ビットを用いてください。詳細は、「21.6.16 受信オーバーランレジスタ (ROVRR)」および「21.12 メッセージバッファ (x) による受信方法」を参照してください。

送信要求の抑止をする場合の操作

送信要求を抑止ないし中止する場合は、BVAL ビットを使用せず、TCAN ビットを用いてください。

送信メッセージを構成する場合の操作

送信メッセージの構成を行うために、ID レジスタや IDE レジスタの設定を行う場合、BVAL ビットを使用してメッセージバッファを禁止してください。この際、送信要求ビットを読み出して "0" であること (TREQ=0) を確認するか、送信完了ビットにより送信が完了したこと (TC=1) を確認した後で、BVAL ビットによる禁止処理 (BVAL=0) を行ってください。

第22章

アドレス一致検出機能

アドレス一致検出の機能と動作について説明します。

22.1 アドレス一致検出機能の概要

22.2 アドレス一致検出機能のレジスタ

22.3 アドレス一致検出機能の動作

22.4 アドレス一致検出機能の使用例

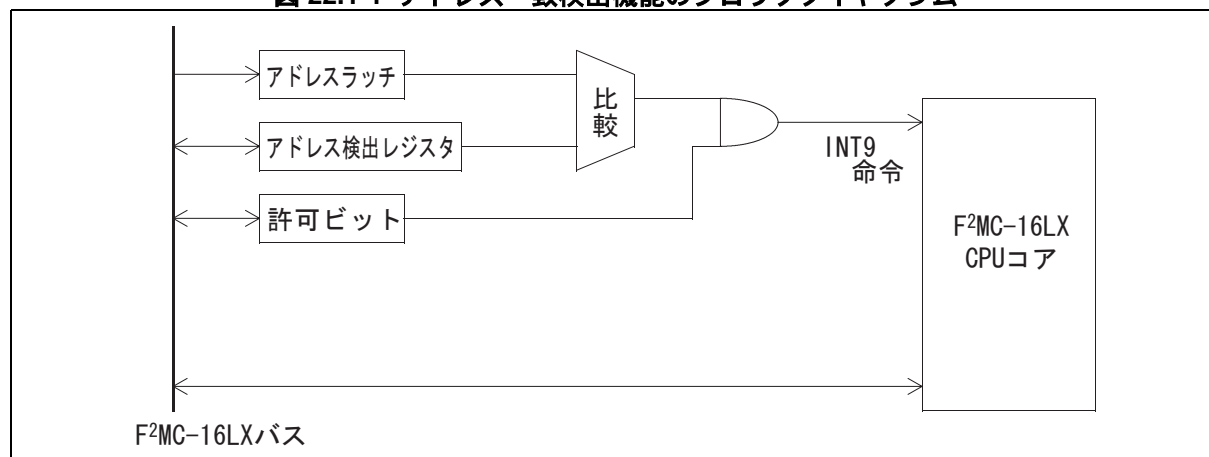
22.1 アドレス一致検出機能の概要

アドレスが、アドレス検出レジスタに設定された値と等しい場合に、CPU に読み込まれる命令コードを強制的に INT9 命令のコード (01_H) に置き換えます。その結果、CPU が設定された命令を実行するときに、INT9 命令を実行します。INT9 割込みルーチンで処理を行うことにより、アドレス一致検出機能を実現することができます。

アドレス検出レジスタは 2 本用意されており、各レジスタにコンペア許可ビットがあります。アドレス検出レジスタに設定された値とアドレスが一致して、かつ割込み許可ビットが "1" の場合、CPU に読み込まれる命令コードを強制的に INT9 命令のコードに置き換えます。

■ アドレス一致検出機能のブロックダイアグラム

図 22.1-1 アドレス一致検出機能のブロックダイアグラム



22.2 アドレス一致検出機能のレジスタ

アドレス一致検出機能には、次の 2 種類のレジスタがあります。

- プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1)
- 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 0/1 (PADR0/PADR1)

プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1:Program Address Detect Register 0/1) は、各レジスタに書き込まれた値と、アドレスを比較します。一致した場合は、PACSR の対応する割込み許可ビットが "1" の場合、CPU に対して INT9 命令の発生を要求します。

対応する割込み許可ビットが "0" の場合は、一致していても何も行いません。

プログラムアドレス検出レジスタ 0/1(PADR0/PADR1) のビット構成を図 22.2-1 に示します。

図 22.2-1 プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1)

アドレス：	バイト	バイト	バイト	アクセス	初期値
PADR0 001FF2 _H /001FF1 _H /001FF0 _H				R/W	不定
PADR1 001FF5 _H /001FF4 _H /001FF3 _H				R/W	不定

表 22.2-1 に、プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1) と PACSR との対応を示します。

表 22.2-1 PADR0/PADR1 レジスタと PACSR との対応

アドレス検出レジスタ	割込み許可ビット
PADR0	AD0E
PADR1	AD1E

■ プログラムアドレス検出制御ステータスレジスタ (PACSR)

プログラムアドレス検出制御ステータスレジスタ (PACSR: Program Address detect Control Status Register) は、アドレス検出機能の動作を制御します。

プログラムアドレス検出制御ステータスレジスタ (PACSR) のビット構成を図 22.2-2 に示します。

図 22.2-2 プログラムアドレス検出制御ステータスレジスタ (PACSR)

	7	6	5	4	3	2	1	0	ビット No.
アドレス：00009E _H	予約	予約	予約	予約	AD1E	予約	AD0E	予約	PACSR
リード/ライト	(-)	(-)	(-)	(-)	(R/W)	(-)	(R/W)	(-)	
初期値	(0)	(0)	(0)	(0)	(0)	(0)	(0)	(0)	

【bit 7 ~ bit 4】予約ビット

bit 7 ~ bit 4は、予約ビットです。PACSRを設定するときは、必ず"0"を設定してください。

【bit 3】AD1E(Address Detect register 1 Enable)

AD1E は、PADR1 の動作許可ビットです。

このビットが "1" のとき PADR1 レジスタとアドレスの比較を行い、一致した場合に CPU は INT9 命令を実行します。

【bit 2】予約ビット

bit2 は、予約ビットです。PACSR を設定するときは、必ず "0" を設定してください。

【bit 1】AD0E (Address Detect register 0 Enable)

AD0E ビットは、PADR0 の動作許可ビットです。

このビットが "1" のとき PADR0 レジスタとアドレスの比較を行い、一致した場合に CPU は INT9 命令を実行します。

【bit 0】予約ビット

bit0 は、予約ビットです。PACSR を設定するときは、必ず "0" を設定してください。

22.3 アドレス一致検出機能の動作

プログラムカウンタがアドレス一致検出レジスタと同じアドレスを指示している場合は、INT9 命令を実行します。INT9 命令ルーチンを処理することによってアドレス一致検出機能を実現することができます。

■ アドレス一致検出機能の動作

アドレス検出レジスタは2本用意されており、各レジスタにコンペア許可ビットがあります。アドレス検出レジスタに設定された値とプログラムカウンタの値が一致して、かつコンペア許可ビットが "1" の場合、CPU は INT9 命令を実行します。

< 注意事項 >

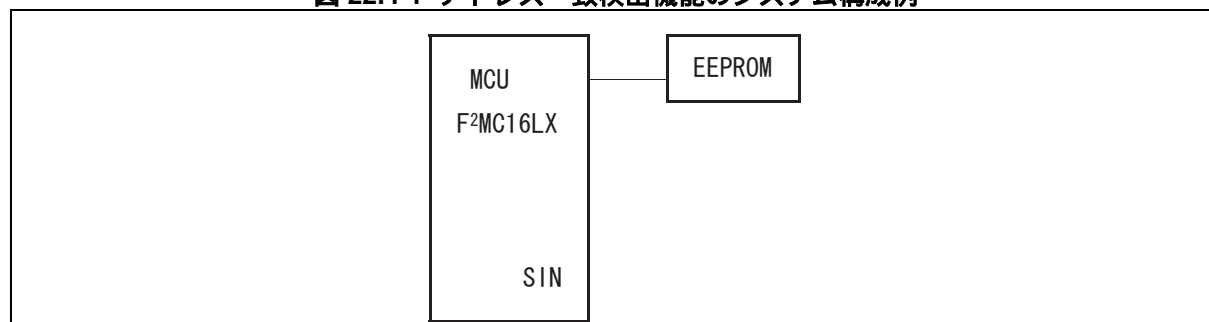
アドレス検出レジスタとプログラムカウンタの値が一致した場合は、内部データバスの内容が強制的に "01_H" に変更されるので、INT9 命令を実行します。アドレス検出レジスタの内容を変更する前には、コンペア許可ビットを必ず "0" にしてください。コンペア許可ビットが "1" になっている間に変更すると、誤動作を引き起こす可能性があります。

22.4 アドレス一致検出機能の使用例

図 22.4-1 にアドレス一致検出機能のシステム構成例を表 22.4-1 に EEPROM メモリマップを示します。

■ アドレス一致検出機能のシステム構成例

図 22.4-1 アドレス一致検出機能のシステム構成例



■ EEPROM メモリマップ

表 22.4-1 EEPROM メモリマップ

アドレス	意味
0000 _H	パッチプログラム No.0 バイト数 (0 のときプログラムミスなし)
0001 _H	プログラムアドレス No.0 bit7 ~ 0
0002 _H	プログラムアドレス No.0 bit15 ~ 8
0003 _H	プログラムアドレス No.0 bit24 ~ 16
0004 _H	パッチプログラム No.1 バイト数 (0 のときプログラムミスなし)
0005 _H	プログラムアドレス No.1 bit7 ~ 0
0006 _H	プログラムアドレス No.1 bit15 ~ 8
0007 _H	プログラムアドレス No.1 bit24 ~ 16
0010 _H ~	パッチプログラム No.0/1 本体

● 初期状態

EEPROM はすべて "0" とします。

● パッチが必要となる場合

コネクタ (UART) を通して、MCU にパッチプログラムの本体およびプログラムアドレスを転送します。MCU はその情報を EEPROM に書き込みます。

● リセットシーケンス

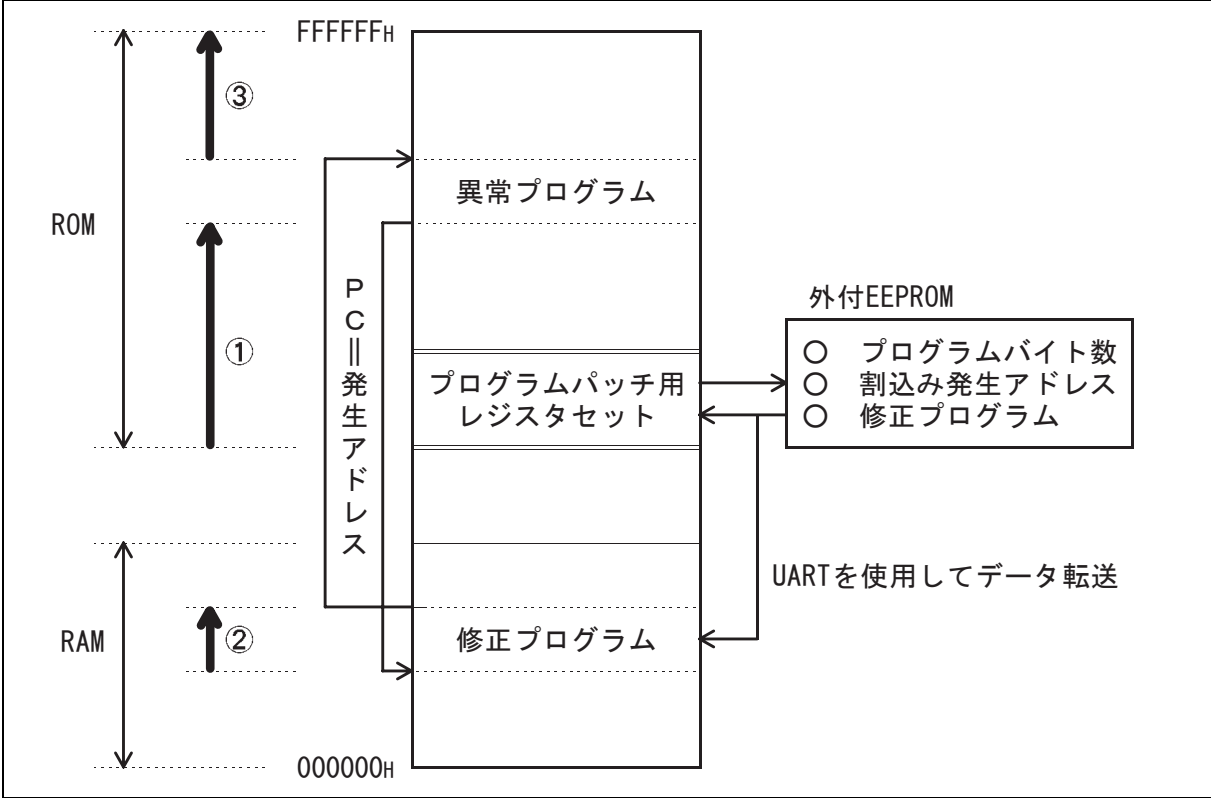
MCU はリセット後 EEPROM の値を読み出します。パッチプログラムのバイト数が "0" でなかった場合は、パッチプログラムの本体が EEPROM から読み出され RAM に書き込まれます。次に、MCU により PADR0 または PADR1 どちらか一方でパッチアドレスをセットし、コンペア許可ビットをセットします。再配置可能はパッチプログラムが望まれる場合は、RAM 領域に対してパッチ済みプログラムの最初のアドレスを書き込むことができます。この場合、INT9 ルーチンがこのユーザ定義の RAM 領域を参照し、パッチ済みプログラムへ向けてジャンプします。

● INT9 命令

割込みルーチンでは、スタックプログラムのカウンタの値を調べるにより、割込みが発生するアドレスを知ることができます。割込みの最中にスタックされた情報は破棄されます。

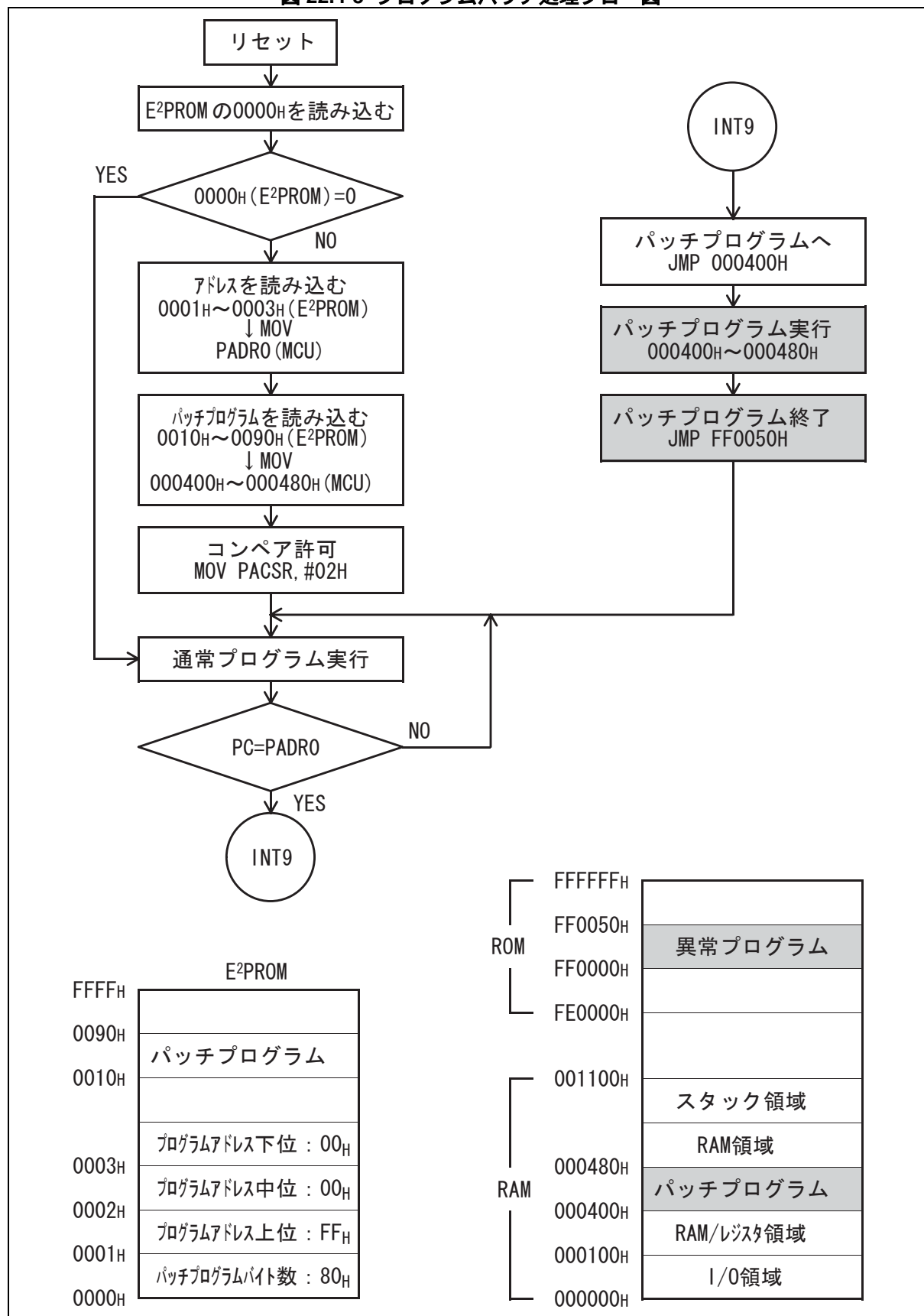
■ プログラムパッチ処理例

図 22.4-2 プログラムパッチ処理例



■ プログラムパッチ処理フロー図

図 22.4-3 プログラムパッチ処理フロー図



第23章

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機能と動作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

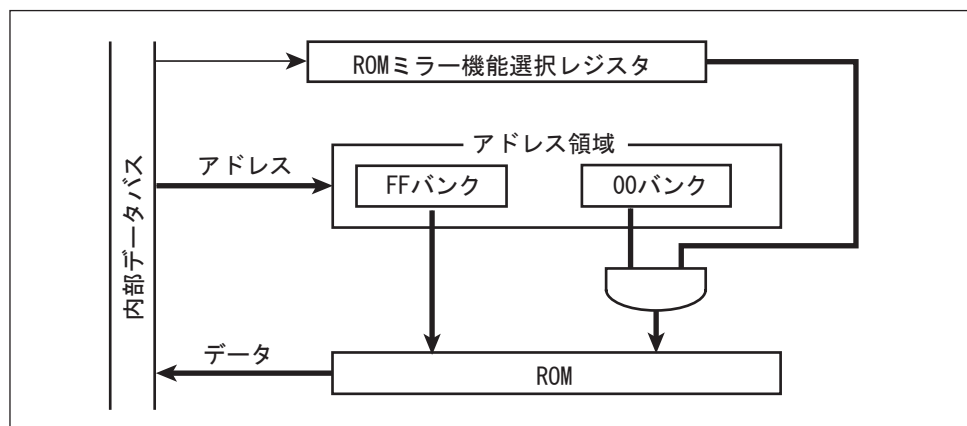
23.2 ROM ミラー機能選択レジスタ (ROMM)

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、ROM が配置されている FF バンクを 00 バンクでみることをレジスタの設定で選択することができます。

■ 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 ROM ミラー機能選択モジュールのブロックダイアグラム



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) は、アドレス 004000_H ~ 00FFFF_H 番地の使用中にアクセスしないでください。

■ 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) のビット構成を図 23.2-1 に示します。

図 23.2-1 ROM ミラー機能選択レジスタ (ROMM)

	15	14	13	12	11	10	9	8	ビット No. ROMM
アドレス : 00006F _H								MI	
リード / ライト	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(W)	
初期値	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(1)	

【bit 8】MI

"1" 書込み時、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" 書込み時、00 バンクはこの機能は働きません。

本ビットは、書込みのみ可能です。

< 注意事項 >

ROMミラー機能を起動している場合、FF4000_H ~ FFFFFFF_H番地のみ004000_H ~ 00FFFF_H番地にミラーされます。このため、FF0000_H ~ FF3FFF_H番地は 00 バンクにミラーされません。

第24章

1M/2M ビットフラッシュ メモリ

1M/2M ビットフラッシュメモリの機能と動作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 / 消去の方法には、下記の3とおりの方法があります。

- パラレルライター
- シリアル専用ライター
- プログラム実行による書込み / 消去

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。

24.1 1M/2M ビットフラッシュメモリの概要

24.2 フラッシュメモリのセクタ構成

24.3 書込み / 消去モード

24.4 フラッシュメモリ制御ステータスレジスタ (FMCS)

24.5 フラッシュメモリ自動アルゴリズム起動方法

24.6 自動アルゴリズム実行状態の確認

24.7 フラッシュメモリ書込み / 消去の詳細説明

24.8 1M/2M ビットフラッシュメモリ使用上の注意

24.9 フラッシュセキュリティの特長

24.10 1M/2M ビットフラッシュメモリのプログラム例

24.1 1M/2M ビットフラッシュメモリの概要

1M/2M ビットフラッシュメモリは、CPU メモリマップ上の "FE_H" / "FC_H" ~ "FF_H" バンクに配置され、フラッシュメモリインタフェース回路の機能により、マスク ROM と同様に CPU からのリードアクセスおよびプログラムアクセスが可能です。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 からの命令動作で行えます。このため、内蔵 CPU の制御による実装状態での書換えが可能となり、プログラムおよびデータの改善が効率よく行えます。

■ 1M/2M ビットフラッシュメモリの特長

- 自動プログラムアルゴリズム (Embedded AlogrithmTM*:MBM29LV200 と同様)
- 消去一時停止 / 消去再開機能の搭載
- データポーリング、トグルビットによる書込み / 消去完了検出
- CPU 割込みによる書込み / 消去の完了検出
- セクタごとの消去が可能 (セクタ組合せ自由)
- 書込み / 消去回数 (最小) 10,000 回

*: Embedded Alogrithm は Advanced Micro Devices 社の商標です。

< 注意事項 >

マニファクチャコードとデバイスコードの読出し機能はありません。また、これらのコードはコマンドによってもアクセスできません。

■ フラッシュメモリ書込み / 消去の方法

フラッシュメモリは書込み / 消去と読出しを同時に行うことはできません。すなわち、フラッシュメモリにデータ書込み / 消去動作を行う際には、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 実行することで、フラッシュメモリからプログラムアクセスをせず書込み動作だけを行うことが可能となります。

■ フラッシュメモリのレジスタ

フラッシュメモリ制御ステータスレジスタ (FMCS) のビット構成を図 24.1-1 に示します。

図 24.1-1 フラッシュメモリ制御ステータスレジスタ (FMCS)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000AE _H	INTE	RDYINT	WE	RDY	予約	LPM1	予約	LPM0	FMCS
リード / ライト	(R/W)	(R/W)	(R/W)	(R)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(X)	(0)	(0)	(0)	(0)	

24.2 フラッシュメモリのセクタ構成

図 24.2-1 に、フラッシュメモリのセクタ構成を示します。

■ 1M/2M ビットフラッシュメモリのセクタ構成

図 24.2-1 に、1M/2M ビットフラッシュメモリのセクタ構成を示します。図中アドレスには、各セクタの上位アドレスと下位アドレスを示します。

図 24.2-1 1M/2M ビットフラッシュメモリのセクタ構成

フラッシュメモリ	CPU アドレス	ライタアドレス *	フラッシュメモリ	CPU アドレス	ライタアドレス *
SA4(16K バイト)	FFFFFF _H	7FFFF _H	SA6(16K バイト)	FFFFFF _H	7FFFF _H
SA3(8K バイト)	FFBFFF _H	7BFFF _H	SA5(8K バイト)	FFBFFF _H	7BFFF _H
SA2(8K バイト)	FF9FFF _H	79FFF _H	SA4(8K バイト)	FF9FFF _H	79FFF _H
SA1(32K バイト)	FF7FFF _H	77FFF _H	SA3(32K バイト)	FF7FFF _H	77FFF _H
SA0(64K バイト)	FEFFFF _H	6FFFF _H	SA2(64K バイト)	FEFFFF _H	6FFFF _H
	FE0000 _H	60000 _H	SA1(64K バイト)	FDFFFF _H	5FFFF _H
			SA0(64K バイト)	FCFFFF _H	4FFFF _H
				FC0000 _H	40000 _H

*: ライタアドレスとは、フラッシュメモリにパラレルライタでデータ書込みを行う際、CPU アドレスに相当するアドレスです。汎用ライタを使用して書込み / 消去を行う際は、このアドレスで書込み / 消去を実行します。

24.3 書込み / 消去モード

フラッシュメモリのアクセスは、フラッシュメモリモードとその他モードの 2 つの異なる方法で行われます。フラッシュメモリモードでは、外部端子から直接書込み / 消去が行え、その他モードでは、内部バスを介して CPU から書込み / 消去が行えます。モードの選択はモード外部端子で行います。

■ フラッシュメモリモード

リセット信号が発生中にモード端子を "111_B" にセットすると、CPU が停止します。フラッシュメモリインタフェース回路は直接ポート 0, 2, 3, 4 に接続しているので、外部端子から直接制御することができます。このモードでは、MCU が外部端子中の標準フラッシュメモリと同様の動作を行い、フラッシュメモリプログラマを使用して書込み / 消去が行えます。

フラッシュメモリモードでは、フラッシュメモリ自動アルゴリズムでサポートされるすべての動作を使用できます。

■ その他モード

フラッシュメモリは、CPU メモリ空間の FE/FC ~ FF バンクに配置されており、通常のマスク ROM と同様にフラッシュメモリインタフェース回路を介して CPU から読出しアクセスおよびプログラムアクセスすることができます。

フラッシュメモリへの書込み / 消去は、フラッシュメモリインタフェース回路を介して CPU からの命令で実行されます。このため、このモードでは MCU が対象ボードにはんだ付けされていても再書込みが可能です。

これらのモードでは、セクタプロテクト動作を実行することはできません。

■ フラッシュメモリの制御信号

表 24.3-1 に、フラッシュメモリモードでのフラッシュメモリ制御信号を示します。

フラッシュメモリ制御信号と MBM29LV200 の外部端子にはほぼ 1 対 1 の対応関係が存在します。セクタプロテクト動作で必要となる V_{ID} (12V) 端子は、MBM29LV200 における A9, \overline{RESET} , \overline{OE} の代わりに、MD0 ~ MD2 となります。

フラッシュメモリモードでは、外部データバス幅は 8 ビットに制限されており、1 バイトのアクセスしか許可されません。DQ15 ~ DQ8 はサポートされていません。 \overline{BYTE} 端子は常に "0" にセットしてください。

表 24.3-1 フラッシュ制御信号

MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S)			MBM29LV200
端子番号	通常機能	フラッシュメモリモード	
1 ~ 8	P20 ~ P27	AQ0 ~ AQ7	A-1, A0 ~ A6
9	P30	AQ16	A15
10	P31	$\overline{\text{CE}}$	$\overline{\text{CE}}$
12	P32	$\overline{\text{CE}}$	$\overline{\text{CE}}$
13	P33	$\overline{\text{WE}}$	$\overline{\text{WE}}$
14	P34	AQ17	A16
16	P36	$\overline{\text{BYTE}}$	$\overline{\text{BYTE}}$
17	P37	RY/ $\overline{\text{BY}}$	RY/ $\overline{\text{BY}}$
18 ~ 22	P40 ~ P44	AQ8 ~ AQ12	A7 ~ A11
24 ~ 26	P45 ~ P47	AQ13 ~ AQ15	A12 ~ A14
49	MD0	MD0	A9(V _{ID})
50	MD1	MD1	$\overline{\text{RESET}}$ (V _{ID})
51	MD2	MD2	OE(V _{ID})
85 ~ 92	P00 ~ P07	DQ0 ~ DQ7	DQ0 ~ DQ7
77	$\overline{\text{RST}}$	$\overline{\text{RESET}}$	$\overline{\text{RESET}}$
使用不可			DQ8 ~ DQ15

24.4 フラッシュメモリ制御ステータスレジスタ (FMCS)

フラッシュメモリ制御ステータスレジスタ (FMCS) 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。

■ フラッシュメモリ制御ステータスレジスタ (FMCS)

フラッシュメモリ制御ステータスレジスタ (FMCS) のビット構成を図 24.4-1 に示します。

図 24.4-1 フラッシュメモリ制御ステータスレジスタ (FMCS)

	7	6	5	4	3	2	1	0	ビット No.
アドレス : 0000AE _H	INTE	RDYINT	WE	RDY	予約	LPM1	予約	LPM0	FMCS
リード / ライト	(R/W)	(R/W)	(R/W)	(R)	(R/W)	(R/W)	(R/W)	(R/W)	
初期値	(0)	(0)	(0)	(X)	(0)	(0)	(0)	(0)	

【bit 7】INTE(INTerrupt Enable)

フラッシュメモリの書込み / 消去の終了で CPU に割込みを発生させるビットです。

INTE ビットが "1" でかつ RDYINT ビットが "1" のときに CPU へ割込みが発生します。INTE ビットが "0" であれば割込みは発生しません。

0: 書込み / 消去終了での割込み禁止

1: 書込み / 消去終了での割込み許可

【bit 6】RDYINT(ReaDY INTerrupt)

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、このビットが "0" の間は、フラッシュメモリへの書込み / 消去はできません。書込み / 消去が終了して "1" になった後は、フラッシュメモリへの書込み / 消去が可能になります。

"0" 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 (「24.5 フラッシュメモリ自動アルゴリズム起動方法」を参照) 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) 系命令使用時は、必ず "1" が読めます。

0: 書込み / 消去動作実行中

1: 書込み / 消去動作終了 (割込み要求発生)

【bit 5】WE(Write Enable)

フラッシュメモリ領域へのライトイネーブルビットです。

このビットが "1" のとき、FE/FC ~ FF バンクへのコマンドシーケンス (「24.5 フラッシュメモリ自動アルゴリズム起動方法」を参照) 発行後の書込みは、フラッシュメモリ領域への書込みになります。本ビットが "0" のとき、書込み / 消去の信号は発生されません。このビットはフラッシュメモリの書込み / 消去のコマンドを起動する際に使用します。

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

【bit 4】RDY(ReadDY)

フラッシュメモリの書込み / 消去許可ビットです。

このビットが "0" の間は、フラッシュメモリへの書込み / 消去はできません。なお、この状態でも読出し / リセットコマンド、セクタ消去一時停止コマンドは受け付けられます。

0: 書込み / 消去動作実行中 (次データ書込み / 消去不可)

1: 書込み / 消去動作終了 (次データ書込み / 消去許可)

【bit 3】予約ビット

試験用予約ビットです。通常使用時は必ず "0" に設定してください。

【bit 1】空きビット

通常使用時は必ず "0" に設定してください。

【bit 2, bit 0】LPM1, LPM0(Low Power Mode)

LPM1, LPM0 ビットを使用すると、フラッシュメモリ本体の消費電力を制御できます。しかし、CPU からフラッシュメモリへのアクセスタイムが設定により大きく異なりますので、CPU の動作周波数により設定値を選択してください。

01: 低電力消費モード (内部動作周波数 4 MHz 以下で動作)

10: 低電力消費モード (内部動作周波数 8 MHz 以下で動作)

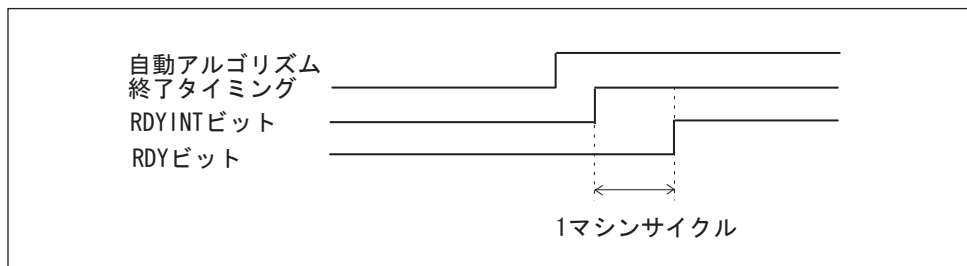
11: 低電力消費モード (内部動作周波数 12.58 MHz 以下で動作)

00: 通常消費電力モード (内部動作周波数 16 MHz 以下で動作)

< 注意事項 >

RDYINT ビットと RDY ビットは同時には変化しません。どちらか片方のビットで判定するようプログラムを作成してください (図 24.4-2 を参照)。

図 24.4-2 RDYINT ビットと RDY ビットの変化タイミング



24.5 フラッシュメモリ自動アルゴリズム起動方法

フラッシュメモリの自動アルゴリズムを起動するコマンドには、読出し/リセット、書込み、チップ消去、セクタ消去の 4 種類があり、セクタ消去については一時停止と再開の制御が可能です。

■ コマンドシーケンス表

表 24.5-1 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 24.5-1 コマンドシーケンス表

コマンド シー ケンス	バス ライト アクセス	1st バスライト サイクル		2nd バスライト サイクル		3rd バスライト サイクル		4th バスライト サイクル		5th バスライト サイクル		6th バスライト サイクル	
		アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ	アドレス	データ
読出し / リセット *	1	FxXXXX	XXF0	-	-	-	-	-	-	-	-	-	-
	4	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XXF0	RA	RD	-	-	-	-
書込み プロ グラム	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"Data(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"Data(xx30 _H) の入力で、セクタ消去中の消去一時停止後、消去開始											
Auto Select	3	FxAAAA	XXAA	Fx5554	XX55	FxAAAA	XX90	-	-	-	-	-	-

(注意事項)

- ・表中のアドレス Fx は、1M ビットフラッシュメモリの場合は FF, FE を、2M ビットフラッシュメモリの場合は FF, FE, FD, FC 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。
 - ・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
 - ・SA: セクタアドレス (「24.2 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
- *: 2 種類の読出し / リセットコマンドは、どちらもフラッシュメモリを読出しモードにリセットすることができます。

表 24.5-2 における Auto Select はセクタ保護の状態を知るためのコマンドです。実際には前期コマンドとともに下記のようにアドレスを設定する必要があります。

表 24.5-2 Auto Select 時のアドレス設定

	AQ13 ~ AQ16	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"

24.6 自動アルゴリズム実行状態の確認

フラッシュメモリでは、書込み / 消去のフローを自動アルゴリズムで行うため、フラッシュメモリ内部の動作状態や動作完了した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, DQ2 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), トグルビット 2 フラグ (DQ2) の機能を持ちます。これにより、書込み / チップ・セクタ消去終了、消去コードライトが有効かの確認をすることができます。

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 (「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 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リードアクセスすることで参照できます。表 24.6-1 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 24.6-1 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	DQ2	-	-

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、フラッシュメモリ制御レジスタ (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、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ライトが有効であるかも、ハードウェアシーケンスフラグによって確認することができます。

表 24.6-2 に、ハードウェアシーケンスフラグ機能一覧を示します。

表 24.6-2 ハードウェアシーケンスフラグ機能一覧

状態		DQ7	DQ6	DQ5	DQ3	DQ2
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	1 DATA:2
	チップセクタ消去動作 消去完了	0 1	Toggle Stop	0 1	1	Toggle Stop
	セクタ消去ウェイト 消去開始	0	Toggle	0	0 1	Toggle
	消去動作 セクタ消去一時停止 (消去中のセクタ)	0 1	Toggle 1	0	1 0	Toggle
	セクタ消去一時停止 消去再開 (消去中のセクタ)	1 0	1 Toggle	0	0 1	Toggle
	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	DATA:2
異常動作	書込み動作	$\overline{\text{DQ7}}$	Toggle	1	0	1
	チップセクタ消去動作	0	Toggle	1	1	*

*: DQ5 が "1" のとき (タイミングリミット超過), 書込み / 消去中セクタへの連続的な読出しに対しては, DQ2 はトグル動作をし, ほかのセクタへの読出しに対してはトグルしません。

24.6.1 データポーリングフラグ (DQ7)

データポーリングフラグ (DQ7) は、自動アルゴリズム実行が進行中もしくは終了状態であることをデータポーリング機能によって知らせるためのフラグです。

■ データポーリングフラグ (DQ7)

表 24.6-3 と表 24.6-4 に、データポーリングフラグの状態遷移を示します。

表 24.6-3 データポーリングフラグの状態遷移 (正常動作時の状態変化)

動作状態	書込み動作完了	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 24.6-4 データポーリングフラグの状態遷移 (異常動作時の状態変化)

動作状態	書込み動作	チップセクタ消去動作
DQ7	$\overline{\text{DQ7}}$	0

- 書込み動作時
自動書込みアルゴリズム実行中にリードアクセスすると、フラッシュメモリはアドレスの指し示す番地によらず、最後に書き込まれたデータの bit 7 の反転データを出力します。自動書込みアルゴリズム終了時にリードアクセスすると、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 bit 7 を出力します。
- チップ / セクタ消去動作時
チップ消去 / セクタ消去アルゴリズム実行中に、セクタ消去時は現在消去しているセクタから、チップ消去時はアドレスの指し示す番地に関係なく、リードアクセスするとフラッシュメモリは "0" を出力します。同様に終了時には "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リードアクセスすると、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 bit 7 (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、現在セクター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可能です。

< 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リード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リードアクセスの次に行うようにしてください。

24.6.2 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、もしくは終了状態であることをトグルビット機能によって知らせるためのフラグです。

■ トグルビットフラグ (DQ6)

表 24.6-5 と表 24.6-6 に、トグルビットフラグの状態遷移を示します。

表 24.6-5 トグルビットフラグの状態遷移 (正常動作時の状態変化)

動作状態	書込み動作完了	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 24.6-6 トグルビットフラグの状態遷移 (異常動作時の状態変化)

動作状態	書込み動作	チップセクタ消去動作
DQ6	Toggle	Toggle

● 書込み / チップ・セクタ消去時

自動書込みアルゴリズムおよびチップ・セクタ消去アルゴリズム実行中に、連続したリードアクセスを行うと、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。自動書込みアルゴリズムおよびチップ / セクタ消去アルゴリズム終了時に連続したリードアクセスを行うとフラッシュメモリは bit 6 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 bit 6 (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リードアクセスすると、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さないのであれば、アドレスの指し示す番地の読出し値の bit 6 (DATA:6) を出力します。

< 参考 >

書込みの際、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 μs 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 μs 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。

24.6.3 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フラグです。

■ タイミングリミット超過フラグ (DQ5)

表 24.6-7 と表 24.6-8 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 24.6-7 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 24.6-8 タイミングリミット超過フラグの状態遷移 (異常動作時の状態変化)

動作状態	書込み動作	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セクタ消去自動アルゴリズム起動後にリードアクセスすると、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とは無関係ですので、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もしくはトグルビット機能により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することができます。

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、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。

24.6.4 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フラグです。

■ セクタ消去タイマフラグ (DQ3)

表 24.6-9 と表 24.6-10 に、セクタ消去タイマフラグの状態遷移を示します。

表 24.6-9 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 24.6-10 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リードアクセスすると、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、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しまっている場合は "1" を出力します。

データポーリング機能やトグルビット機能により消去アルゴリズムが実行中を示している場合、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ち、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" であったなら追加セクタの消去コードは受け付けられてない可能性があります。

● セクタ消去動作時

セクタ消去一時停止中にリードアクセスすると、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 bit 3 (DATA:3) を出力します。

24.6.5 トグルビット 2 フラグ (DQ2)

トグルビット 2 フラグ (DQ2) は、セクタ消去一時停止中であることをトグルビット機能によって知らせるフラグです。

■ トグルビット 2 フラグ (DQ2)

表 24.6-11 と表 24.6-12 に、トグルビット 2 フラグの状態遷移を示します。

表 24.6-11 トグルビット 2 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
DQ2	1 DATA:2	Toggle Stop	Toggle	Toggle	Toggle	DATA:2

表 24.6-12 トグルビット 2 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
DQ2	1	*

* : DQ5 が "1" のとき (タイミングリミット超過), 書込み / 消去中セクタへの連続的な読出しに対しては, DQ2 はトグル動作をし, ほかのセクタへの読出しに対してはトグルしません。

● セクタ消去動作時

チップセクタ消去アルゴリズム実行中に連続したリードアクセスを行うと,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。チップセクタ消去アルゴリズム終了後に連続したリードアクセスを行うと,フラッシュメモリは bit 2 のトグル出力を止め,アドレスの指し示す番地の読出し値の bit 2 (DATA:2) 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中に連続したリードアクセスを行うと,フラッシュメモリはアドレスの指し示す番地が,消去中のセクタに属するならば "1" と "0" を交互に出力するトグル状態を出力します。消去中のセクタに属さないのであれば,アドレスの指し示す番地の読出し値の bit 2 (DATA:2) を出力します。

また,セクタ消去一時停止書込み時,消去一時停止していないセクタから連続したリードアクセスを行うと, "1" を出力します。

DQ2 は DQ6 とともに使用し,消去一時停止中であることを検出するために使用します (DQ2 はトグル動作するが, DQ6 はトグル動作しない)。

さらに, DQ2 は消去しているセクタの検出にも使用します。消去動作時には, DQ2 は消去しているセクタからのリードアクセスならばトグル動作をします。

< 参考 >

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 2 は約 100 μ s のトグル動作をし，その後データを書き換えしないで読出し / リセット状態に戻ります。

24.7 フラッシュメモリ書込み / 消去の詳細説明

自動アルゴリズムを起動するコマンドを発行し、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、セクタ消去再開のそれぞれの動作を行う手順を説明します。

■ フラッシュメモリ書込み / 消去の詳細説明

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、消去再開の動作がコマンドシーケンス（「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 を参照）のバスへのライトサイクルを行うことで自動アルゴリズムを実行することが可能です。それぞれのバスへのライトサイクルは必ず続けて行う必要があります。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。

各動作について、次項より下記の順に示します。

- 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

24.7.1 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。

■ フラッシュメモリの読出し / リセット状態

フラッシュメモリを読出し / リセット状態にするには、コマンドシーケンス表（「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 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可能です。

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リードアクセスでデータを読み出せます。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。

24.7.2 フラッシュメモリへのデータ書込み

書込みコマンドを発行し、フラッシュメモリへ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を書込み

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（「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 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可能です。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、自動書込みが開始します。

● 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。

すなわち、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も可能ですが、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、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 (DQ5) がエラーと判定するか、あるいは見かけ上データ "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。消去動作のみが "0" 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。

■ フラッシュメモリ書込み手順

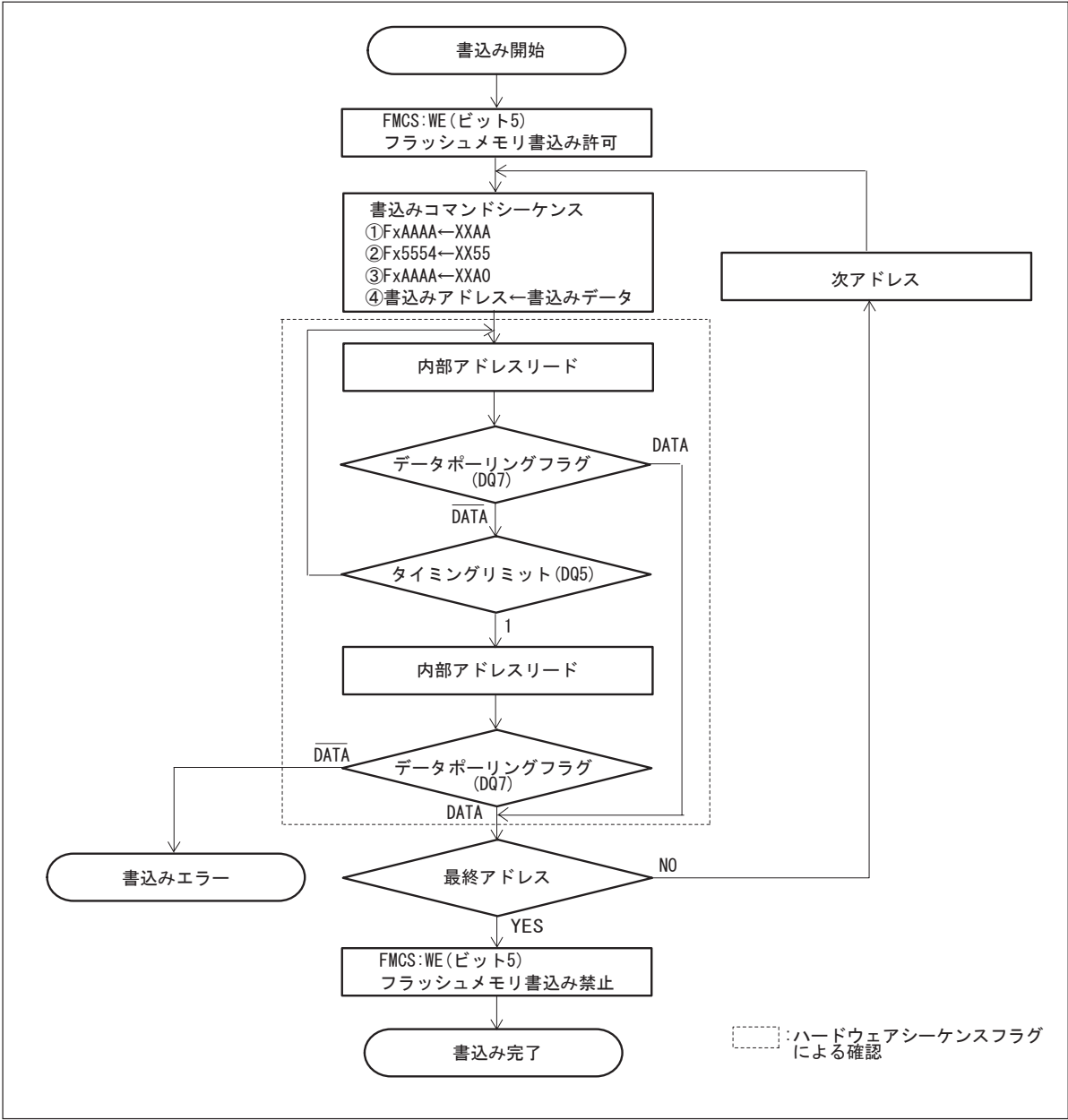
図 24.7-1 に、フラッシュメモリ書込みの手順の例を示します。ハードウェアシーケンスフラグ（「24.6 自動アルゴリズム実行状態の確認」を参照）を用いることでフラッシュメモリ内部の自動アルゴリズムの状態判定が可能です。ここでは、書込み終了の確認にデータポーリングフラグ (DQ7) を用いています。

フラグチェックのために読み出すデータは最後に書込みを行ったアドレスからの読出しとなります。

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 24.7-1 フラッシュメモリ書き込み手順の例



24.7.3 フラッシュメモリの全データの消去 (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の消去 (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 (「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 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続けて送ることで実行可能です。

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。

24.7.4 フラッシュメモリの任意データの消去 (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データを消去する (セクタ消去) 手順について説明します。セクタごとの消去が可能であり、複数のセクタを同時に指定することも可能です。

■ フラッシュメモリの任意データの消去 (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 (「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 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可能です。

● 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り 50 μ s のセクタ消去ウェイトが開始します。複数のセクタ消去を行う場合は、上記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ときの注意

最後のセクタ消去コードの書込みから 50 μ s のセクタ消去ウェイト期間終了により消去が開始します。すなわち、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り、それ以降では受け付けられないことがあります。引き続く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。なお、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。

■ フラッシュメモリのセクタ消去手順

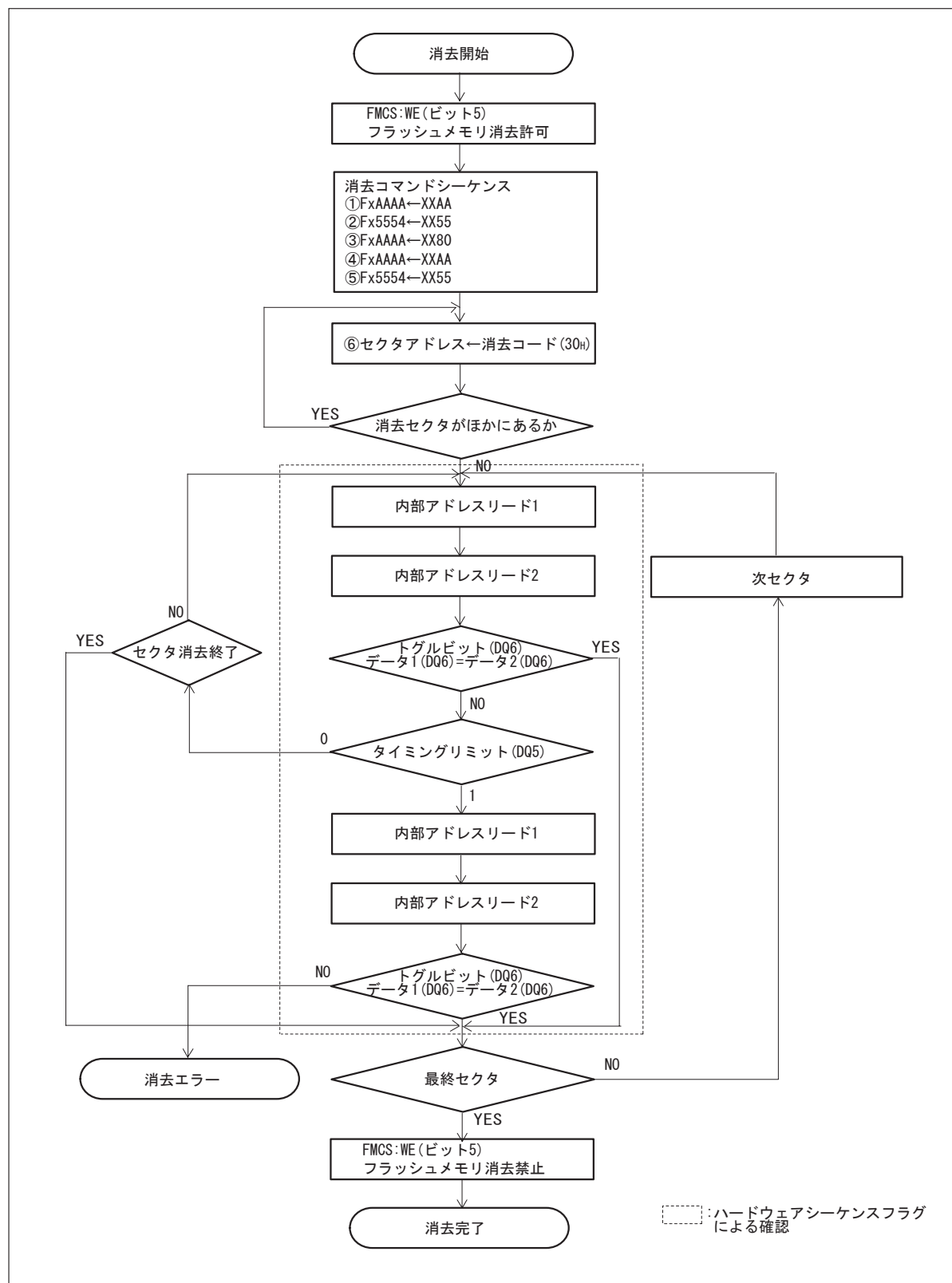
ハードウェアシーケンスフラグ (「24.6 自動アルゴリズム実行状態の確認」を参照) を用いることでフラッシュメモリ内部の自動アルゴリズムの状態判定が可能です。図 24.7-2 に、フラッシュメモリのセクタ消去手順の例を示します。ここでは、消去終了の確認にトグルビットフラグ (DQ6) を用いています。

フラグチェックのために読み出すデータは、消去しようとしているセクタからの読出しとなりますので、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を再チェックする必要があります。

図 24.7-2 フラッシュメモリのセクタ消去手順の例



24.7.5 フラッシュメモリのセクタ消去の一時停止

セクタ消去一時停止コマンドを発行し、フラッシュメモリセクタ消去の一時停止を行う手順について説明します。消去中でないセクタから、データを読み出すことが可能です。

■ フラッシュメモリのセクタ消去の一時停止

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（「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 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可能です。

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 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。

24.7.6 フラッシュメモリのセクタ消去の再開

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の再開

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 (「24.5 フラッシュメモリ自動アルゴリズム起動方法」の表 24.5-1 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送ることで実行可能です。

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) を書き込む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。

24.8 1M/2M ビットフラッシュメモリ使用上の注意

1M/2M ビットフラッシュメモリに関する注意を以下に示します。

■ フラッシュメモリ使用上の注意

● ハードウェアリセット ($\overline{\text{RST}}$) の入力

読出し中で自動アルゴリズムが起動していないときに、ハードウェアリセットを入力するには、"L" レベル幅として最低 500 ns をとる必要があります。この場合、ハードウェアリセット起動後フラッシュメモリからデータを読み出すまでに、最低 500 ns が必要となります。

同様に、書込み / 消去中で自動アルゴリズムが起動しているときに、ハードウェアリセットを入力するには、"L" レベル幅として最低 500ns をとる必要があります。この場合、フラッシュメモリを初期化するために実行中の動作を停止した後データを読み出すまでに、20 μ s が必要となります。

書込み中にハードウェアリセットすると、書き込まれているデータは不定となります。消去中のハードウェアリセットや電源切れにより、消去されているセクタは使用不可となる可能性があります。

● ソフトウェアリセット、ウォッチドッグタイマリセット、ハードウェアスタンバイの取消し

CPU アクセスでフラッシュメモリ書込み / 消去中に、自動アルゴリズムがアクティブ状態でリセット条件が発生した場合、CPU が暴走する可能性があります。それは、これらのリセット条件により、フラッシュメモリユニットが初期化されずに自動アルゴリズムが続行し、リセット離脱後にCPUがシーケンスを開始したときフラッシュメモリユニットが読出し状態になることが妨げられる可能性があるためです。これらのリセット条件は、フラッシュメモリの書込み / 消去中に禁止されている必要があります。

● フラッシュメモリへのプログラムアクセス

自動アルゴリズム動作中は、フラッシュメモリへの読出しアクセスは禁止されます。CPU のメモリアクセスモードが内蔵 ROM モードにセットされている場合、プログラム領域を RAM などの別の領域に切換え後、書込み / 消去を開始する必要があります。この場合、割込みベクタを含むセクタ (SA4/SA6) が消去されると、書込み / 消去割込み処理を実行できません。

● ホールド機能

CPU がホールド要求を受け付けたとき、フラッシュメモリユニットの書込み信号 $\overline{\text{WE}}$ は歪曲され、不正な書込みによる不正な書込み / 消去が可能になります。ホールド要求の受け付けが許可 (ECSR の HDE ビットが "1" をセット) されたとき、フラッシュメモリ制御ステータスレジスタ (FMCS) の WE ビットが必ず "0" になっているようにしてください。

● 拡張インテリジェント I/O サービス (EI²OS)

フラッシュメモリインタフェース回路から CPU に発行された書込み / 消去割込みは、EI²OS には受け付け不可能であり、使用できません。

● V_{ID} の適用

セクタプロテクト動作に必要な V_{ID} の印加は、電源が ON のとき開始し、終了する必要があります。

24.9 フラッシュセキュリティの特長

フラッシュセキュリティコントローラ機能により、フラッシュメモリの内容を外部端子から読み出されることを防止できます。

■ フラッシュセキュリティの特長

事前に定義されたフラッシュメモリのアドレスが、フラッシュセキュリティコントローラ (1M ビットフラッシュメモリ : FE0001_H, 2M ビットフラッシュメモリ : FC0001_H) に割り当てられます。このアドレスに保護コード "01_H" が書き込まれると、フラッシュメモリへのアクセスが制限されます。フラッシュメモリが一度保護されると、チップ消去を行うまで、機能のロックを解除することはできません。ロックを解除しない限り、いずれの外部端子からもフラッシュメモリへの読出し / 書込みはできません。

この機能は、フラッシュメモリに格納される自己完結型プログラムやデータのセキュリティを必要とするアプリケーションに適しています。マイクロコントローラの外部に位置するプログラム部分を対象アプリケーションが必要とする場合には、フラッシュセキュリティコントローラは、目的の機能を提供することができません。このため、保護コードを設定した場合は、外部ベクタフェッチモードは使用しません。

標準パラレルプログラマによるフラッシュマイクロコントローラのプログラミングには、独特のセットアップが必要になることがあります。例えば、ミナトエレクトロニクスのプログラマでは、デバイスチェックを切断する必要があります。保護コードは、フラッシュプログラミングの終わりにコーディングすることを推奨されます。これは、プログラミング中の不要な保護を回避するためです。

一度保護されたフラッシュメモリを再度プログラムするには、チップ消去操作を行う必要があります。

詳細については、弊社担当者にお問い合わせください。

24.10 1M/2M ビットフラッシュメモリのプログラム例

1M/2M ビットフラッシュメモリのプログラム例を掲載します。

■ 1M/2M ビットフラッシュメモリのプログラム例

```

NAME  FLASHWE
TITLE FLASHWE
;-----
;1M/2Mbit-FLASH サンプルプログラム
;
;1: FLASH にあるプログラム ( アドレス FFC000H セクタ SA4/SA6) を RAM( アドレス
; 000700H) に転送する。
;2: RAM 上でプログラムを実行する。
;3: PDR1 の値を FLASH( アドレス FE0000H セクタ SA0/SA2) に書き込む。
;4: 書き込んだ値 ( アドレス FE0000H セクタ SA0/SA2) を読み出し PDR2 に出力する。
;5: 書き込んだセクタ (SA0/SA2) を消去する。
;6: 消去データ確認の出力
;条件
;  ・ RAM 転送バイト数 : 100H(256B)
;  ・ 書込み , 消去の終了判定
;      DQ5( タイミングリミット超過フラグ ) での判定
;      DQ6( トグルビットフラグ ) での判定
;      RDY(FMCS) での判定
;  ・ エラー時の処理
;      P00 ~ P07 に Hi を出力する
;      リセットコマンド発行
;-----
;
RESOUS IOSEG  ABS=00      ;"RESOUS" I/O セグメントの定義
          ORG    0000H
PDR0     RB      1
PDR1     RB      1
PDR2     RB      1
PDR3     RB      1
          ORG    0010H
DDR0     RB      1
DDR1     RB      1
DDR2     RB      1
DDR3     RB      1
          ORG    00A1H
CKSCR    RB      1

```

第 24 章 1M/2M ビットフラッシュ メモリ

```

                ORG    00AEH
FMCS           RB     1
                ORG    006FH
ROMM           RB     1
RESOUS         ENDS
;
SSTA           SSEG
                RW     0127H
STA_T          RW     1
SSTA           ENDS
;
DATA           DSEG   ABS=0FFH      ;FLASH コマンドアドレス
                ORG    5554H
COMADR2        RW     1
                ORG    0AAAAH
COMADR1        RW     1
DATA           ENDS
;////////////////////////////////////
; メインプログラム (SA1)
;////////////////////////////////////
CODE           CSEG
START:
;////////////////////////////////////
; 初期化
;////////////////////////////////////
MOV    CKSCR,#0BAH    ;3 週倍に設定
MOV    RP,#0
MOV    A,#!STA_T
MOV    SSB,A
MOVW   A,#STA_T
MOVW   SP,A
MOV    ROMM,#00H      ; ミラー OFF
MOV    PDR0,#00H      ; エラー確認用
MOV    DDR0,#0FFH
MOV    PDR1,#00H      ; データ入力用ポート
MOV    DDR1,#00H
MOV    PDR2,#00H      ; データ出力用ポート
MOV    DDR2,#0FFH
;////////////////////////////////////
;RAM(700H番地)に"FLASH書き込み消去プログラム(FFC000H)"を転送する
;////////////////////////////////////
MOVW   A,#0700H      ; 転送先 RAM 領域

```

```

MOVW  A,#0C000H    ; 転送元アドレス ( プログラムのある位置 )
MOVW  RW0,#100H    ; 転送するバイト数
MOVS  ADB,PCB      ; FFBC00H から 000700H へ 100H 転送
CALLP 000700H      ; 転送したプログラムのあるアドレスへジャンプ
;////////////////////////////////////
; データ出力
;////////////////////////////////////
OUT    MOV    A,#0FEH
      MOV    ADB,A
      MOVW  RW2,#0000H
      MOVW  A,@RW2+00
      MOV    PDR2,A
END     JMP    *
CODE    ENDS
;////////////////////////////////////
; FLASH 書き込み消去プログラム (SA4/SA6)
;////////////////////////////////////
RAMPRG CSEG    ABS=0FFH
      ORG    0C000H
;
; //////////////////////////////////
; 初期化
; //////////////////////////////////
MOVW  RW0,#0500H    ; RW0: 入力データ確保用 RAM 空間    00:0500 ~
MOVW  RW2,#0000H    ; RW2: フラッシュメモリ書き込みアドレス  FD:0000 ~
MOV    A,#00H      ; DTB 変更
MOV    DTB,A       ; @RW0 用バンク指定
MOV    A,#0FEH     ; ADB 変更 1
MOV    ADB,A       ; 書き込みモード指定アドレス用バンク指定
MOV    PDR3,#00H   ; スイッチ初期化
MOV    DDR3,#00H
;
WAIT1  BBC    PDR3:0,WAIT1 ; PDR3:0 Hi で書き込みスタート
;
;////////////////////////////////////
; 書き込み (SA0/SA2)
;////////////////////////////////////
      MOV    A,PDR1
      MOVW  @RW0+00,A    ; RAM に PDR1 データを確保
      MOV    FMCS,#20H   ; 書き込みモード設定
      MOVW  ADB:COMADR1,#00AAH ; フラッシュ書き込みコマンド 1
      MOVW  ADB:COMADR2,#0055H ; フラッシュ書き込みコマンド 2
      MOVW  ADB:COMADR1,#00A0H ; フラッシュ書き込みコマンド 3

```



```

;
MOVW    A,@RW0+00          ; 入力データ (RW0) をフラッシュ
;                               メモリ (RW2) に書き込む
MOVW    @RW2+00,A
WRITE    ; 待ち時間チェック
; //////////////////////////////////////
; タイムリミット超過チェック - フラグが立ちトグル動作中であるとき ERROR
; //////////////////////////////////////
MOVW    A,@RW2+00
AND      A,#20H             ;DQ5 タイムリミットチェック
BZ       NTOW               ; タイムリミットオーバ
MOVW    A,@RW2+00           ;AH
MOVW    A,@RW2+00           ;AL
XORW    A                  ;AH AL の XOR( 値が違えば 1)
AND      A,#40H             ;DQ6 トグルビットは違っているか
BNZ      ERROR             ; 違えば ERROR へ
; //////////////////////////////////////
; 書込み終了チェック (FMCS-RDY)
; //////////////////////////////////////
NTOW     MOVW    A,FMCS
AND      A,#10H             ;FMCS RDY ビット (4 ビット) 抽出
BZ       WRITE             ; 書込み終了か ?
MOV      FMCS,#00H          ; 書込みモード解除
;////////////////////////////////////
; 書込みデータ出力
;////////////////////////////////////
MOVW    RW2,#0000H          ; 書込みデータ出力
MOVW    A,@RW2+00
MOV      PDR2,A
;
WAIT2    BBC      PDR3:1,WAIT2 ;PDR3:1 Hi でセクタ消去スタート
;
;////////////////////////////////////
; セクタ消去 (SA0/SA2)
;////////////////////////////////////
MOV      @RW2+00,#0000H     ; アドレス初期化
MOV      FMCS,#20H          ; 消去モード設定
MOVW     ADB:COMADR1,#00AAH ; フラッシュ消去コマンド 1
MOVW     ADB:COMADR2,#0055H ; フラッシュ消去コマンド 2
MOVW     ADB:COMADR1,#0080H ; フラッシュ消去コマンド 3
MOVW     ADB:COMADR1,#00AAH ; フラッシュ消去コマンド 4
MOVW     ADB:COMADR2,#0055H ; フラッシュ消去コマンド 5

```

```

MOV      @RW2+00,#0030H      ; 消すセクタに消去コマンド発行 6
ELS      ; 待ち時間チェック
;      ;////////////////////////
;      ; タイムリミット超過チェック - フラグが立ちトグル動作中であるとき ERROR
;      ;////////////////////////
MOVW     A,@RW2+00
AND      A,#20H              ;DQ5 タイムリミットチェック
BZ       NTOE                ; タイムリミットオーバ
MOVW     A,@RW2+00          ;AH 書込み動作中は , DQ6 から
MOVW     A,@RW2+00          ;AL 読出しごとHiLowが交互出力される
XORW     A                  ;AH と AL の XOR (DQ6 の値が違えば
;                               1 書込み動作中である )
AND      A,#40H              ;DQ6 トグルビットは Hi か
BNZ      ERROR              ;Hi なら ERROR へ
;      ;////////////////////////
;      ; 消去終了チェック (FMCS-RDY)
;      ;////////////////////////
NTOE     MOVW     A,FMCS      ;
AND      A,#10H              ;FMCS RDY ビット (4 ビット ) 抽出
BZ       ELS                 ; セクタ消去終了か ?
MOV      FMCS,#00H           ;FLASH 消去モード解除
RETP     ;メインプログラムに戻る
;////////////////////////
; エラー
;////////////////////////
ERROR    MOV      ADB:COMADR1,#0F0H ; リセットコマンド ( 読出しが可能になる )
MOV      FMCS,#00H           ;FLASH モード解除
MOV      PDR0,#0FFH          ; エラー処理の確認
RETP     ;メインプログラムに戻る
RAMPRG   ENDS
;////////////////////////
VECT     CSEG     ABS=0FFH
ORG      0FFDCH
DSL      START
DB       00H
VECT     ENDS
;
END      START

```


第25章

MB90F543/F549/F543G(S)/ F548G(S)/F549G(S)/ F546G(S)/F548GL(S) 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 AF220/
AF210/AF120/AF110 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の接続例につい
て説明します。

- 25.1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/
F546G(S)/F548GL(S) シリアル書込み接続の基本構成
- 25.2 シリアル書込み接続例 (ユーザ電源使用時)
- 25.3 シリアル書込み接続例 (ライターから電源供給時)
- 25.4 フラッシュマイコンプログラマとの最小限の接続例
(ユーザ電源使用時)
- 25.5 フラッシュマイコンプログラマとの最小限の接続例
(ライターから電源供給時)

25.1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) シリアル書込み接続の基本構成

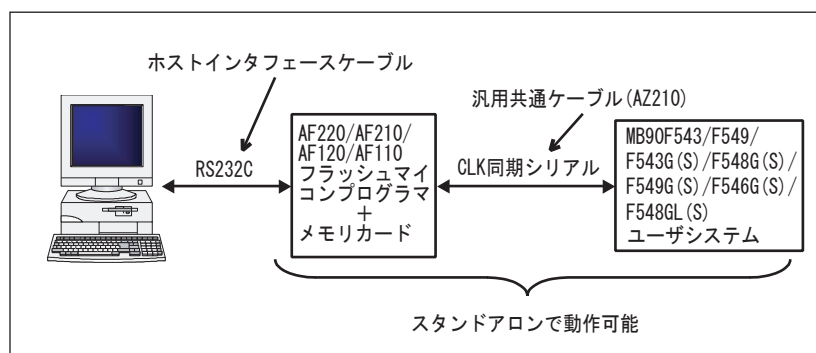
MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) では、フラッシュ ROM のシリアルオンボード書込み (富士通標準) をサポートしています。その仕様について以下に説明します。

■ B90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) シリアル書込み接続の基本構成

富士通標準シリアルオンボード書込みには、横河デジタルコンピュータ株式会社製 AF220/AF210/AF120/AF110 フラッシュマイコンプログラマを使用します。

図 25.1-1 に、MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) シリアル書込み接続の基本構成を示します。

図 25.1-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) シリアル書込み接続の基本構成



< 注意事項 >

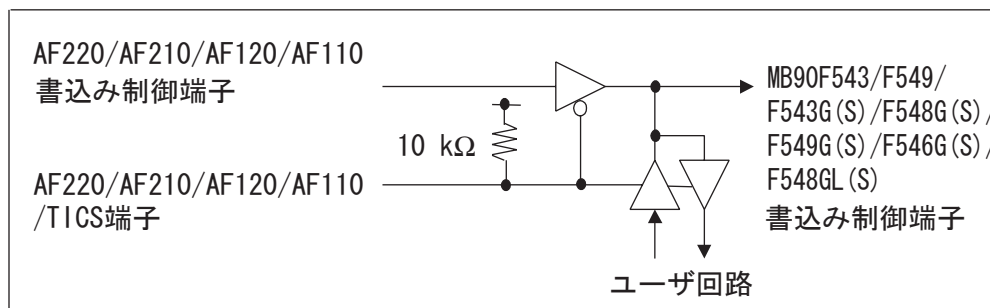
AF220/AF210/AF120/AF110 フラッシュマイコンプログラマの機能、操作方法、接続用汎用共通ケーブル (AZ210) およびコネクタについては、横河デジタルコンピュータ株式会社にお問合せください。

表 25.1-1 富士通標準シリアルオンボード書込みに使用する端子

端 子	機 能	補足説明
MD2, MD1, MD0	モード端子	フラッシュマイコンプログラマから、書込みモードに制御します。
X0, X1	発振子用端子	書込みモード時に、CPU 内部動作クロックは PLL クロック 1 通倍となっております。したがって、発振クロック周波数が、内部動作クロックとなります。
P00, P01	書込みプログラム起動端子	P00 に "L" レベルを、P01 に "H" レベルを入力してください。
$\overline{\text{RST}}$	リセット端子	-
SIN1	シリアルデータ入力端子	UART1 を CLK 同期モードとして使用します。
SOT1	シリアルデータ出力端子	
SCK1	シリアルクロック入力端子	
C	C 端子	電源安定化の容量端子です。外部に 0.1 μF 程度のセラミックコンデンサを接続してください。
V_{CC}	電源電圧供給端子	書込み電圧 (5V \pm 10%) をユーザシステムから供給する場合にはフラッシュマイコンプログラマとの接続は必要ありません。接続時にはユーザ側の電源と短絡しないようにしてください。
V_{SS}	GND 端子	フラッシュマイコンプログラマの GND と共通にします。
$\overline{\text{HST}}$	ハードウェアスタンバイ端子	シリアル書込みモード中は "H" レベルを入力してください。

なお、P00, SIN1, SOT1, SCK1 端子をユーザシステムでも使用する場合には、図 25.1-2 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マの /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)。

図 25.1-2 制御回路



「25.2 シリアル書込み接続例 (ユーザ電源使用時)」以降に、次の 4 つのシリアル書込み接続例を示していますので参照してください。

- MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) 内部ベクタモード時 シリアル書込み接続例 (ユーザ電源使用時)
- MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) 内部ベクタモード時 シリアル書込み接続例 (ライタから電源供給時)
- 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)
- フラッシュマイコンプログラマとの最小限の接続例 (ライタから電源供給時)

表 25.1-2 フラッシュマイコンプログラマシステム構成 (横河デジタルコンピュータ株式会社製)

型 格		機 能
本体	AF220/AC4P	イーサネットインタフェース内蔵モデル /100 V ~ 220 V 電源アダプタ
	AF210/AC4P	スタンダードモデル /100V ~ 220V 電源アダプタ
	AF120/AC4P	単キーイーサネットインタフェースモデル /100 V ~ 220 V 電源アダプタ
	AF110/AC4P	単キーモデル /100 V ~ 220 V 電源アダプタ
AZ221		ライター専用 PC/AT 用 RS232C ケーブル
AZ210		標準ターゲットプローブ (a) 長さ :1m
FF201		富士通製 F ² MC-16LX フラッシュマイコン用制御モジュール
AZ290		リモートコントローラ
/P2		2M バイト PC Card (Option) フラッシュメモリ容量 ~ 128K バイト対応
/P4		4M バイト PC Card (Option) フラッシュメモリ容量 ~ 512K バイト対応

問い合わせ先: 横河デジタルコンピュータ株式会社
電話 :042-333-6224

< 注意事項 >

AF200フラッシュマイコンプログラマについては、終息製品ですが、制御モジュールFF201を用いることで対応可能です。シリアル書込み接続例に関しては、「発振クロック周波数とシリアルクロック入力周波数」に示した接続例にて対応可能です。

■ 発振クロック周波数とシリアルクロック入力周波数

MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S) の入力可能なシリアルクロック周波数は以下の計算式より求められます。

$$\text{入力可能なシリアルクロック周波数} = 0.125 \times \text{発振クロック周波数}$$

したがって、ご使用の発振クロック周波数によって、シリアルクロック入力周波数をフラッシュマイコンプログラマの設定にて変更してください。

表 25.1-3 入力可能なシリアルクロック周波数の例

発振クロック周波数	マイコンの入力可能な最大シリアルクロック周波数	AF220/AF210/AF120/AF110 の設定可能な最大シリアルクロック周波数	AF200 の設定可能な最大シリアルクロック周波数
4 MHz 時	500 kHz	500 kHz	500 kHz
8 MHz 時	1 MHz	850 kHz	500 kHz
16 MHz 時	2 MHz	1.25 MHz	500 kHz

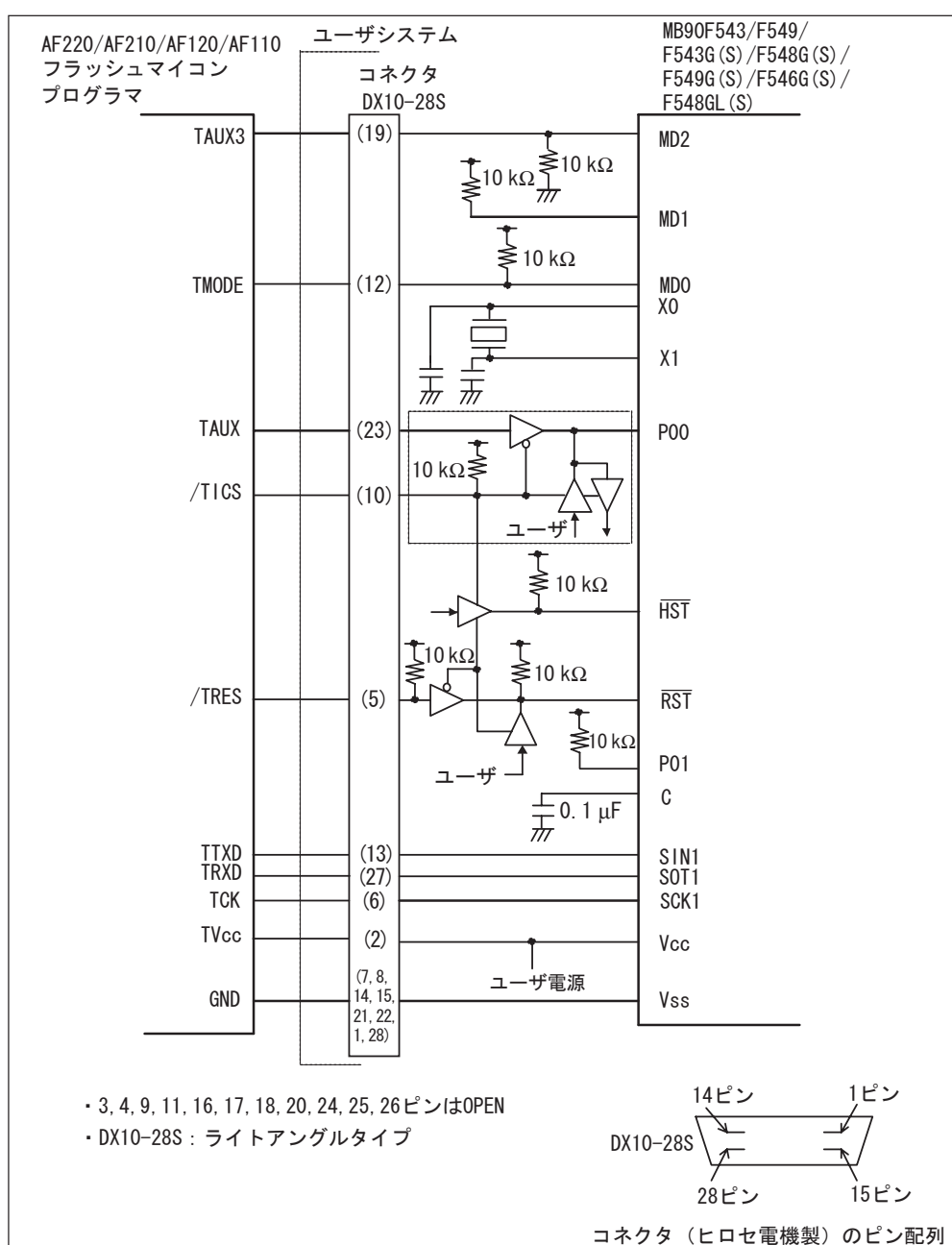
25.2 シリアル書き込み接続例 (ユーザ電源使用時)

図 25.2-1 に , マイコンの電源電圧をユーザ電源より供給する場合のシリアル書き込み接続例を示します。なお , モード端子 MD2, MD0 には , AF220/AF210/AF120/AF110 の TAUX3, TMODE より , MD2 = 1, MD0 = 0 が入力されます。

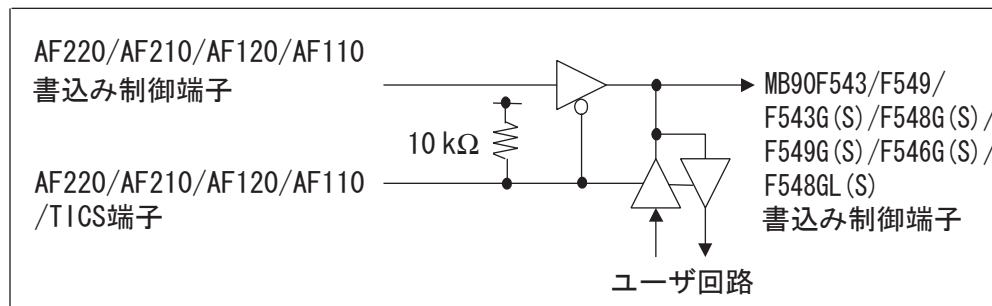
シリアル書換えモード : MD2, MD1, MD0 = 110_B

■ シリアル書き込み接続例 (ユーザ電源使用時)

図 25.2-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S)
シングルチップモード時シリアル書き込み接続例 (ユーザ電源使用時)



- SIN1, SOT1, SCK1 端子をユーザシステムでも使用する場合には, P00 と同様に, 下図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, シリアル書き込み中はユーザ回路を切り離すことができます)。



- AF220/AF210/AF120/AF110 との接続はユーザ電源が OFF の状態で行ってください。

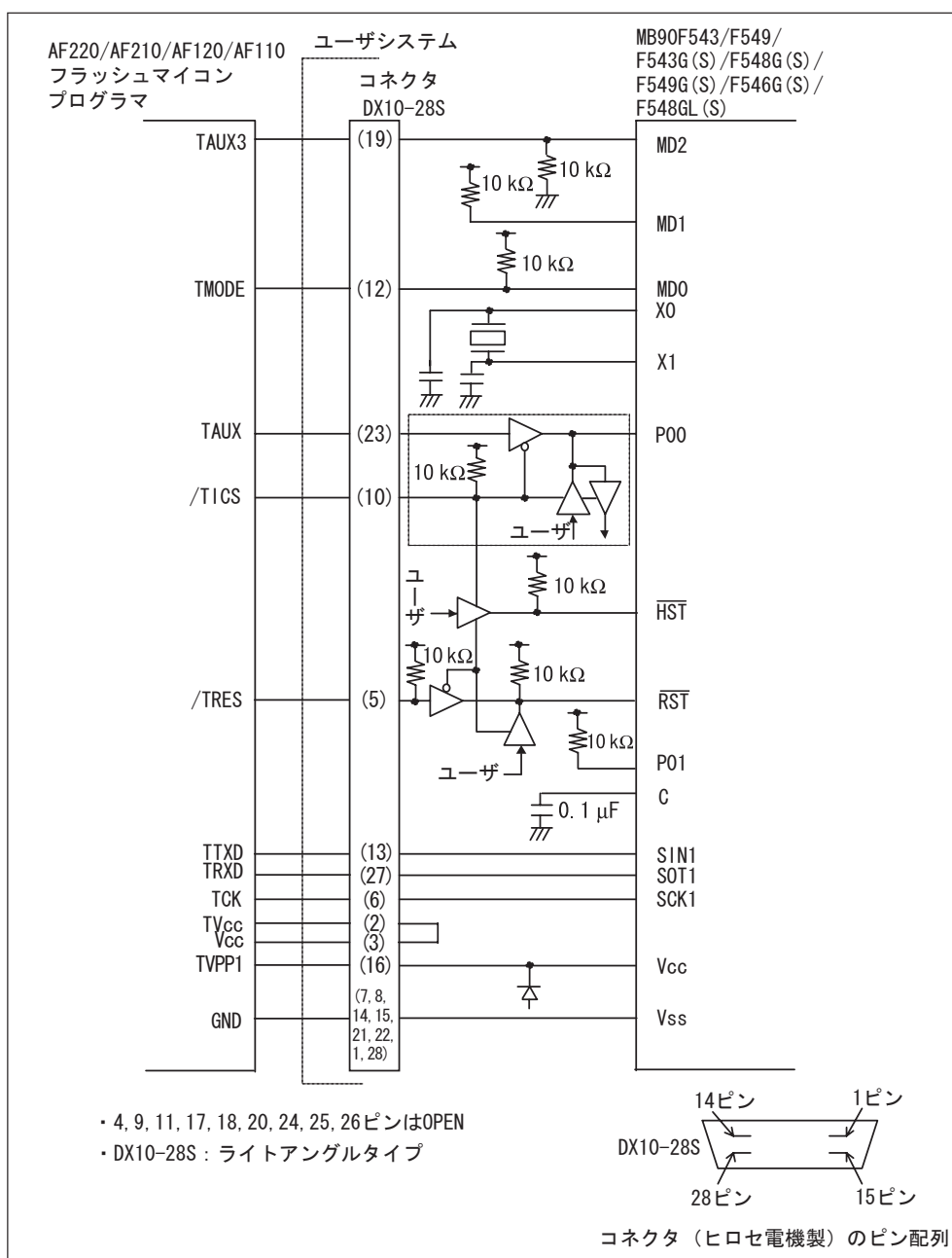
25.3 シリアル書込み接続例 (ライタから電源供給時)

図 25.3-1 に、マイコンの電源電圧をライタ電源より供給する場合のシリアル書込み接続例を示します。なお、モード端子 MD2, MD0 には、AF220/AF210/AF120/AF110 の TAUX3, TMODE より、MD2 = 1, MD0 = 0 が入力されます。

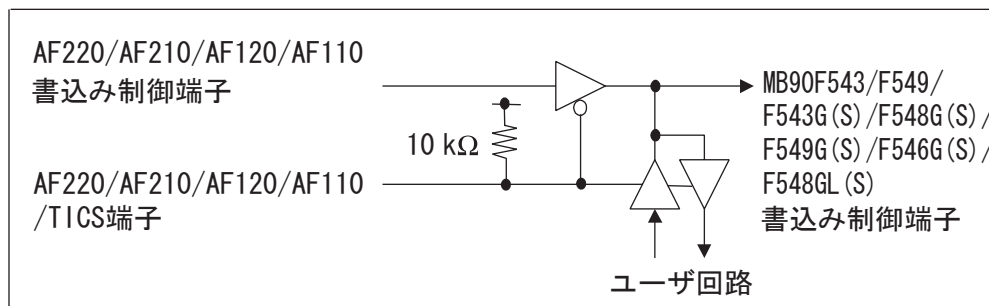
シリアル書換えモード：MD2, MD1, MD0 = 110_B

■ シリアル書込み接続例（ライタから電源供給時）

図 25.3-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S)
シングルチップモード時シリアル書き込み接続例 (ライターから電源供給時)



- SIN1, SOT1, SCK1 端子をユーザシステムでも使用する場合には、P00 と同様に、下図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、シリアル書き込み中はユーザ回路を切り離すことができます）。



- AF220/AF210/AF120/AF110 との接続はユーザ電源が OFF の状態で行ってください。
- 書き込み電源をAF220/AF210/AF120/AF110から供給するときはユーザ電源と短絡しないでください。

25.4 フラッシュマイコンプログラムの最小限の接続例 (ユーザ電源使用時)

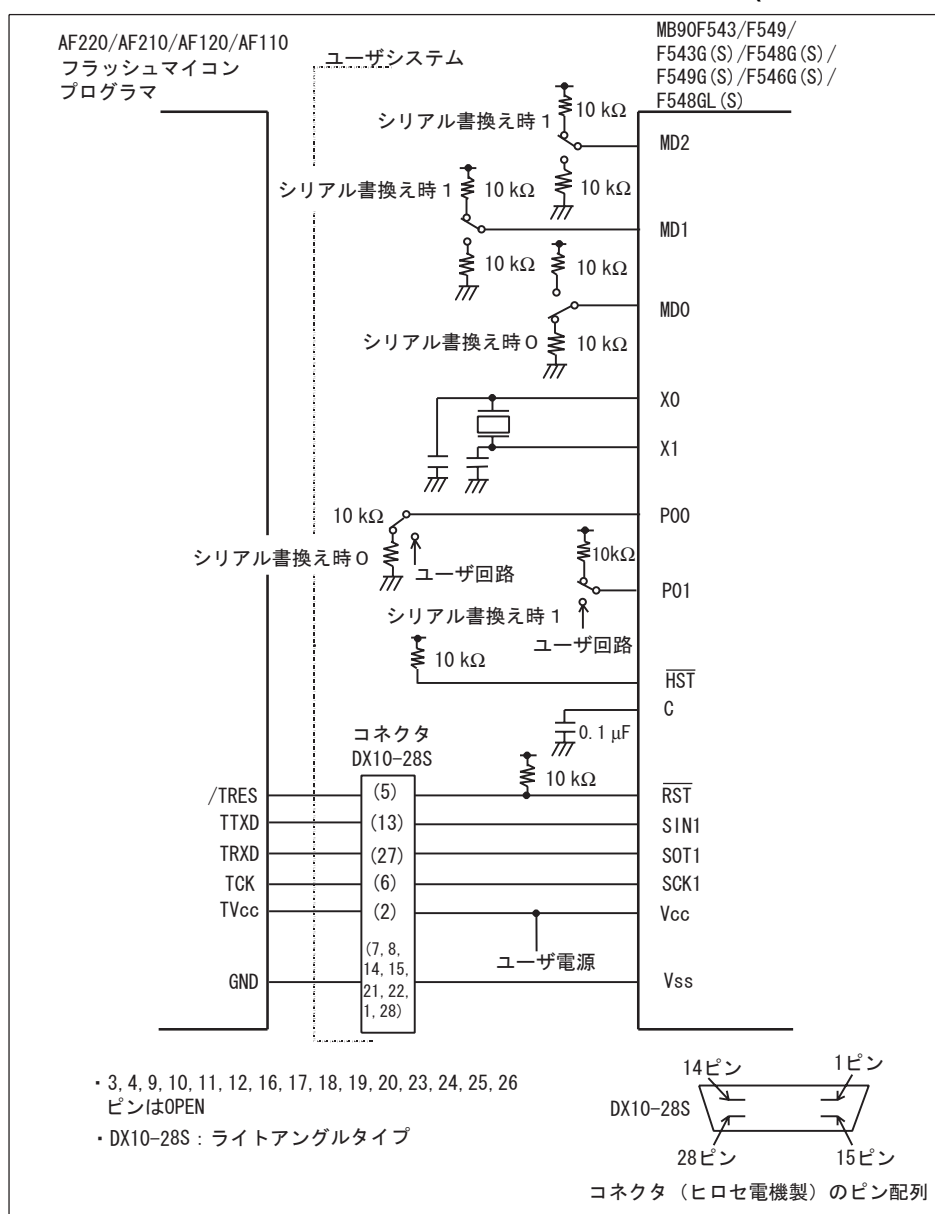
図 25.4-1 に、マイコンの電源電圧をユーザ電源より供給する場合のフラッシュマイコンプログラムの最小限の接続例を示します。

シリアル書換えモード : MD2, MD1, MD0 = 110_B

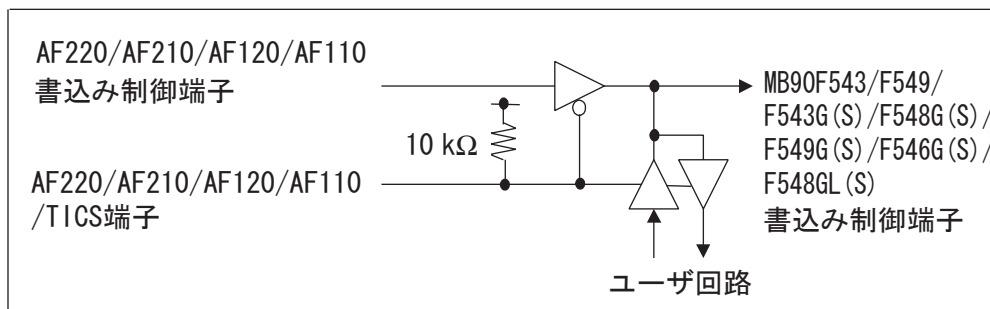
■ フラッシュマイコンプログラムの最小限の接続例 (ユーザ電源使用時)

フラッシュメモリ書込み時に、各端子を図 25.4-1 に示すように設定すると、MD2, MD1, MD0, P00 とフラッシュマイコンプログラムの接続は必要ありません。

図 25.4-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S)
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)



- SIN1, SOT1, SCK1 端子をユーザシステムでも使用する場合には、下図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、シリアル書き込み中はユーザ回路を切り離すことができます）。



- AF220/AF210/AF120/AF110 との接続はユーザ電源が OFF の状態で行ってください。

25.5 フラッシュマイコンプログラムの最小限の接続例 (ライターから電源供給時)

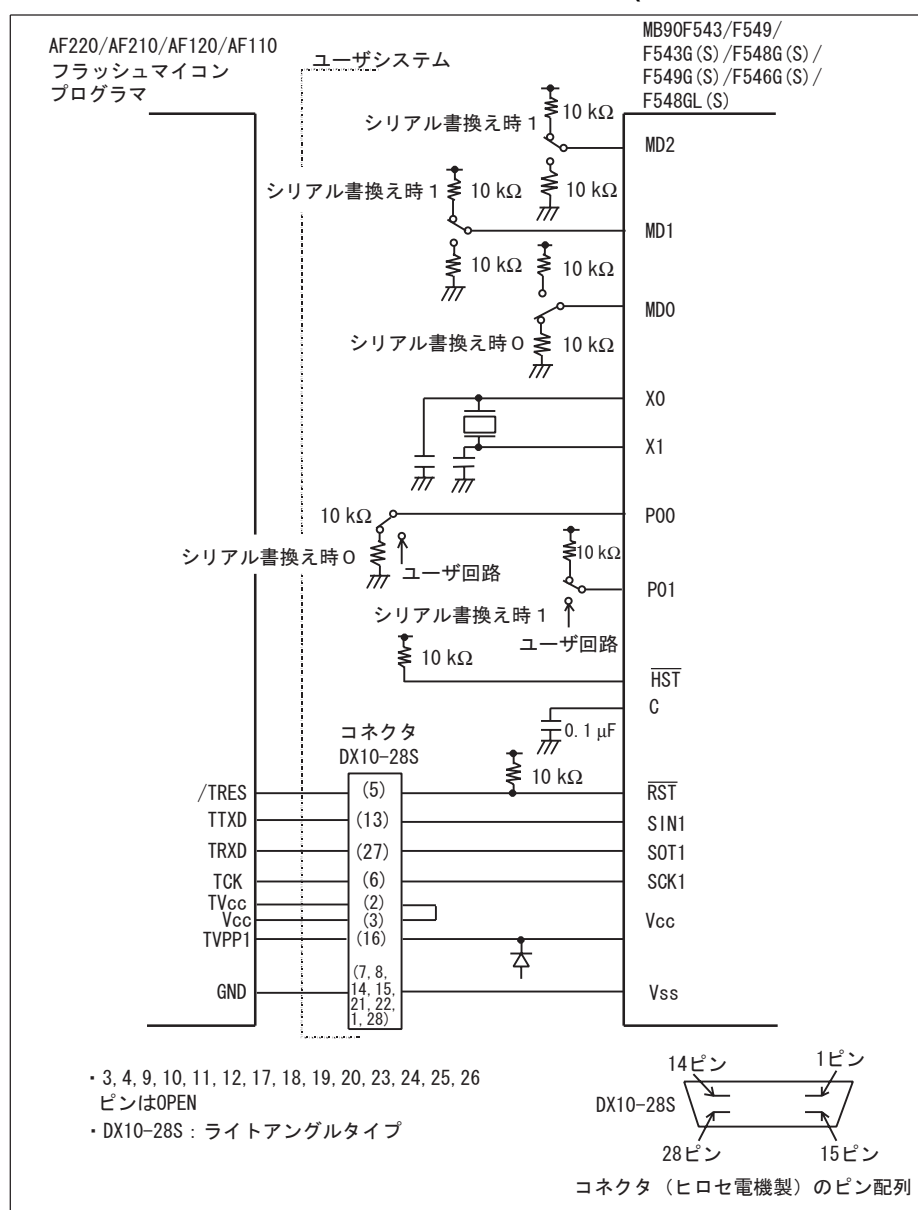
図 25.5-1 に、マイコンの電源電圧をライター電源より供給する場合のフラッシュマイコンプログラムの最小限の接続例を示します。

シリアル書換えモード : MD2, MD1, MD0 = 110_B

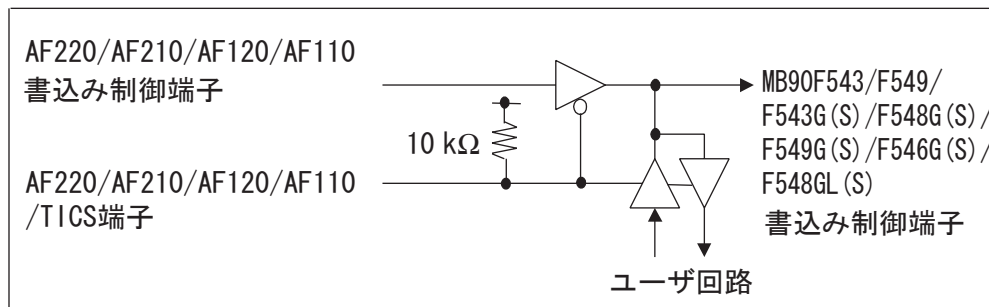
■ フラッシュマイコンプログラムの最小限の接続例 (ライターから電源供給時)

フラッシュメモリ書込み時に各端子を図 25.5-1 に示すように設定すれば、MD2 ~ MD0, P00 とフラッシュマイコンプログラムの接続は必要ありません。

図 25.5-1 MB90F543/F549/F543G(S)/F548G(S)/F549G(S)/F546G(S)/F548GL(S)
フラッシュマイコンプログラムの最小限の接続例 (ライターから電源供給時)



- SIN1, SOT1, SCK1 端子をユーザシステムでも使用する場合には、下図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、シリアル書き込み中はユーザ回路を切り離すことができます）。



- AF220/AF210/AF120/AF110 との接続はユーザ電源が OFF の状態で行ってください。
- 書き込み電源をAF220/AF210/AF120/AF110から供給するときはユーザ電源と短絡しないでください。

I/O マップ, 命令一覧表などを掲載します。

付録 A I/O マップ

付録 B 命令

付録 A I/O マップ

本マイクロコントローラの各リソースのレジスタは、以下のようなアドレスが割り当てられています。

■ I/O マップ

付表 A-1 I/O マップ (1 / 6)

アドレス	レジスタ	略称	アクセス	リソース	初期値
000000 _H	ポートデータレジスタ (ポート 0 用)	PDR0	R/W	ポート 0	XXXXXXXX _B
000001 _H	ポートデータレジスタ (ポート 1 用)	PDR1	R/W	ポート 1	XXXXXXXX _B
000002 _H	ポートデータレジスタ (ポート 2 用)	PDR2	R/W	ポート 2	XXXXXXXX _B
000003 _H	ポートデータレジスタ (ポート 3 用)	PDR3	R/W	ポート 3	XXXXXXXX _B
000004 _H	ポートデータレジスタ (ポート 4 用)	PDR4	R/W	ポート 4	XXXXXXXX _B
000005 _H	ポートデータレジスタ (ポート 5 用)	PDR5	R/W	ポート 5	XXXXXXXX _B
000006 _H	ポートデータレジスタ (ポート 6 用)	PDR6	R/W	ポート 6	XXXXXXXX _B
000007 _H	ポートデータレジスタ (ポート 7 用)	PDR7	R/W	ポート 7	XXXXXXXX _B
000008 _H	ポートデータレジスタ (ポート 8 用)	PDR8	R/W	ポート 8	XXXXXXXX _B
000009 _H	ポートデータレジスタ (ポート 9 用)	PDR9	R/W	ポート 9	XXXXXXXX _B
00000A _H	ポートデータレジスタ (ポート A 用)	PDRA	R/W	ポート A	- - - - - X _B
00000B _H ~ 00000F _H	予約領域				
000010 _H	ポート方向レジスタ (ポート 0 用)	DDR0	R/W	ポート 0	00000000 _B
000011 _H	ポート方向レジスタ (ポート 1 用)	DDR1	R/W	ポート 1	00000000 _B
000012 _H	ポート方向レジスタ (ポート 2 用)	DDR2	R/W	ポート 2	00000000 _B
000013 _H	ポート方向レジスタ (ポート 3 用)	DDR3	R/W	ポート 3	00000000 _B
000014 _H	ポート方向レジスタ (ポート 4 用)	DDR4	R/W	ポート 4	00000000 _B
000015 _H	ポート方向レジスタ (ポート 5 用)	DDR5	R/W	ポート 5	00000000 _B
000016 _H	ポート方向レジスタ (ポート 6 用)	DDR6	R/W	ポート 6	00000000 _B
000017 _H	ポート方向レジスタ (ポート 7 用)	DDR7	R/W	ポート 7	00000000 _B
000018 _H	ポート方向レジスタ (ポート 8 用)	DDR8	R/W	ポート 8	00000000 _B
000019 _H	ポート方向レジスタ (ポート 9 用)	DDR9	R/W	ポート 9	00000000 _B
00001A _H	ポート方向レジスタ (ポート A 用)	DDRA	R/W	ポート A	- - - - - 0 _B
00001B _H	アナログ入力許可レジスタ (ポート 6 用)	ADER	R/W	ポート 6, A/D	11111111 _B
00001C _H	プルアップ制御レジスタ (ポート 0 用)	PUCR0	R/W	ポート 0	00000000 _B
00001D _H	プルアップ制御レジスタ (ポート 1 用)	PUCR1	R/W	ポート 1	00000000 _B
00001E _H	プルアップ制御レジスタ (ポート 2 用)	PUCR2	R/W	ポート 2	00000000 _B
00001F _H	プルアップ制御レジスタ (ポート 3 用)	PUCR3	R/W	ポート 3	00000000 _B
000020 _H	シリアルモード制御レジスタ 0	UMC0	W, R/W	UART0	00000100 _B
000021 _H	シリアルステータスレジスタ 0	USR0	R, R/W		00010000 _B
000022 _H	シリアルインプットデータレジスタ 0/ シリアルアウトプットデータレジスタ 0	UIDR0/ UODR0	R/W		XXXXXXXX _B
000023 _H	レート・データレジスタ 0	URD0	R/W		0000000X _B

付表 A-1 I/O マップ (2/6)

アドレス	レジスタ	略称	アクセス	リソース	初期値
000024 _H	シリアルモードレジスタ 1	SMR1	R/W	UART1	00000000 _B
000025 _H	シリアル制御レジスタ 1	SCR1	W, R/W		00000100 _B
000026 _H	シリアルインプットデータレジスタ 1/ シリアルアウトプットデータレジスタ 1	SIDR1/ SODR1	R/W		XXXXXXXX _B
000027 _H	シリアルステータスレジスタ 1	SSR1	R, R/W		00001-00 _B
000028 _H	UART1 通信プリスケアラ制御レジスタ	CDCR	R/W		0---1111 _B
000029 _H	シリアルエッジセレクトレジスタ	SES1	R/W		-----0 _B
00002A _H	使用禁止				
00002B _H	シリアル I/O プリスケアラ	SCDCR	R/W	シリアル I/O	0---1111 _B
00002C _H	シリアルモード制御ステータスレジスタ	SMCS	R/W		----0000 _B
00002D _H					00000010 _B
00002E _H	シリアルシフトデータレジスタ	SDR	R/W		XXXXXXXX _B
00002F _H	シリアルエッジセレクトレジスタ	SES2	R/W		-----0 _B
000030 _H	DTP/ 外部割込み許可レジスタ	ENIR	R/W	DTP/ 外部 割込み	00000000 _B
000031 _H	DTP/ 外部割込み要因レジスタ	EIRR	R/W		XXXXXXXX _B
000032 _H	要求レベル設定レジスタ	ELVR	R/W		00000000 _B
000033 _H					00000000 _B
000034 _H	A/D 制御ステータスレジスタ 0	ADCS0	R/W	A/D コンバータ	00000000 _B
000035 _H	A/D 制御ステータスレジスタ 1	ADCS1	R/W		00000000 _B
000036 _H	A/D データレジスタ 0	ADCR0	R		XXXXXXXX _B
000037 _H	A/D データレジスタ 1	ADCR1	R, W		00001-XX _B
000038 _H	PPG0 動作モード制御レジスタ	PPGC0	W, R/W	PPG (ch0, ch1) ユニット 0	0-000--1 _B
000039 _H	PPG1 動作モード制御レジスタ	PPGC1	W, R/W		0-000001 _B
00003A _H	PPG ユニット 0 クロック選択レジスタ	PPG01	R/W		000000-- _B
00003B _H	予約領域				
00003C _H	PPG2 動作モード制御レジスタ	PPGC2	W, R/W	PPG (ch2, ch3) ユニット 1	0-000--1 _B
00003D _H	PPG3 動作モード制御レジスタ	PPGC3	W, R/W		0-000001 _B
00003E _H	PPG ユニット 1 クロック選択レジスタ	PPG23	R/W		000000-- _B
00003F _H	使用禁止				
000040 _H	PPG4 動作モード制御レジスタ	PPGC4	W, R/W	PPG (ch4, ch5) ユニット 2	0-000--1 _B
000041 _H	PPG5 動作モード制御レジスタ	PPGC5	W, R/W		0-000001 _B
000042 _H	PPG ユニット 2 クロック選択レジスタ	PPG45	R/W		000000-- _B
000043 _H	使用禁止				
000044 _H	PPG6 動作モード制御レジスタ	PPGC6	R/W	PPG (ch6, ch7) ユニット 3	0-000--1 _B
000045 _H	PPG7 動作モード制御レジスタ	PPGC7	R/W		0-000001 _B
000046 _H	PPG ユニット 3 クロック選択レジスタ	PPG67	R/W		000000-- _B
000047 _H ~ 00004B _H	使用禁止				
00004C _H	インプットキャプチャ制御 ステータスレジスタ 0/1	ICS01	R/W	インプット キャプチャ 0/1	00000000 _B
00004D _H	インプットキャプチャ制御 ステータスレジスタ 2/3	ICS23	R/W	インプット キャプチャ 2/3	00000000 _B
00004E _H	インプットキャプチャ制御 ステータスレジスタ 4/5	ICS45	R/W	インプット キャプチャ 4/5	00000000 _B
00004F _H	インプットキャプチャ制御 ステータスレジスタ 6/7	ICS67	R/W	インプット キャプチャ 6/7	00000000 _B

付表 A-1 I/O マップ (3 / 6)

アドレス	レジスタ	略称	アクセス	リソース	初期値
000050 _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 0	00000000 _B
000051 _H					----0000 _B
000052 _H					XXXXXXXX _B
000053 _H					XXXXXXXX _B
000054 _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 1	00000000 _B
000055 _H					----0000 _B
000056 _H					XXXXXXXX _B
000057 _H					XXXXXXXX _B
000058 _H	アウトプットコンペア制御レジスタ 0	OCS0	R/W	アウトプット コンペア 0/1	0000--00 _B
000059 _H	アウトプットコンペア制御レジスタ 1	OCS1			---00000 _B
00005A _H	アウトプットコンペア制御レジスタ 2	OCS2	R/W	アウトプット コンペア 2/3	0000--00 _B
00005B _H	アウトプットコンペア制御レジスタ 3	OCS3			---00000 _B
00005C _H ~ 00006B _H	使用禁止				
00006C _H	タイマカウンタデータレジスタ	TCDT	R/W	入出力タイマ	00000000 _B
00006D _H					00000000 _B
00006E _H					00000000 _B
00006F _H	ROM ミラー機能選択レジスタ	ROMM	W	ROM ミラー 機能選択 モジュール	-----1 _B
000070 _H ~ 00007F _H	予約領域 (CAN0 インタフェース用)				
000080 _H ~ 00008F _H	予約領域 (CAN1 インタフェース用)				
000090 _H ~ 00009D _H	使用禁止				
00009E _H	プログラムアドレス検出制御ステータス レジスタ	PACSR	R/W	アドレス一致 検出機能	00000000 _B
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 発生 モジュール	-----0 _B
0000A0 _H	低消費電力モード制御レジスタ	LPMCR	R/W	低消費電力 制御回路	00011000 _B
0000A1 _H	クロック選択レジスタ	CKSCR	R/W		11111100 _B
0000A2 _H ~ 0000A4 _H	使用禁止				
0000A5 _H	自動レディ機能選択レジスタ	ARSR	W	外部メモリ アクセス	0011--00 _B
0000A6 _H	外部アドレス出力制御レジスタ	HACR			00000000 _B
0000A7 _H	バス制御信号選択レジスタ	ECSR			00000000- _B
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	XXXXX111 _B
0000A9 _H	タイムベースタイマ制御レジスタ	TBTC	R/W	タイムベース タイマ	1--00100 _B
0000AA _H	時計タイマ制御レジスタ	WTC	R/W	時計タイマ	1X000000 _B
0000AB _H ~ 0000AD _H	使用禁止				
0000AE _H	フラッシュメモリ制御 ステータスレジスタ	FMCS	R/W	フラッシュ メモリ	000X0000 _B
0000AF _H	使用禁止				

付表 A-1 I/O マップ (4/6)

アドレス	レジスタ	略称	アクセス	リソース	初期値	
0000B0 _H	割込み制御レジスタ 00	ICR00	R/W	割込み コントローラ	書込み時	00000111 _B
					読み時	--000111 _B
0000B1 _H	割込み制御レジスタ 01	ICR01	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B2 _H	割込み制御レジスタ 02	ICR02	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B3 _H	割込み制御レジスタ 03	ICR03	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B4 _H	割込み制御レジスタ 04	ICR04	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B5 _H	割込み制御レジスタ 05	ICR05	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B6 _H	割込み制御レジスタ 06	ICR06	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B7 _H	割込み制御レジスタ 07	ICR07	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B8 _H	割込み制御レジスタ 08	ICR08	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000B9 _H	割込み制御レジスタ 09	ICR09	R/W		書込み時	00000111 _B
					読み時	--000111 _B
0000BA _H	割込み制御レジスタ 10	ICR10	R/W	書込み時	00000111 _B	
				読み時	--000111 _B	
0000BB _H	割込み制御レジスタ 11	ICR11	R/W	書込み時	00000111 _B	
				読み時	--000111 _B	
0000BC _H	割込み制御レジスタ 12	ICR12	R/W	書込み時	00000111 _B	
				読み時	--000111 _B	
0000BD _H	割込み制御レジスタ 13	ICR13	R/W	書込み時	00000111 _B	
				読み時	--000111 _B	
0000BE _H	割込み制御レジスタ 14	ICR14	R/W	書込み時	00000111 _B	
				読み時	--000111 _B	
0000BF _H	割込み制御レジスタ 15	ICR15	R/W	書込み時	00000111 _B	
				読み時	--000111 _B	
0000C0 _H ~ 0000FF _H	外部領域					
001FF0 _H	プログラムアドレス検出レジスタ 0	PADR0	R/W	アドレス一致 検出機能		XXXXXXXX _B
001FF1 _H			R/W			XXXXXXXX _B
001FF2 _H			R/W			XXXXXXXX _B
001FF3 _H	プログラムアドレス検出レジスタ 1	PADR1	R/W			XXXXXXXX _B
001FF4 _H			R/W			XXXXXXXX _B
001FF5 _H			R/W			XXXXXXXX _B

・初期値について 0: 初期値 "0" / 1: 初期値 "1" / X: 初期値不定 / "-": 初期値不定 (無し)

・00FF_H 以下のアドレスは予約領域です。外バスアクセス信号はできません。

・RAM 領域と予約領域の境界 #H は品種ごとに変わります。

(注意事項) 書込み可能なビットに関しては、リセットにより初期化される値が初期値として記述されています。
読み出し時の値ではありませんので注意してください。

また、LPMCR/CKSCR/WDTC では、リセットの種類により、初期化される場合と、初期化されない場合がありますが、初期化される場合の初期値が記述されています。

・コンペアレジスタ 0 ~ 3 はフリーランタイム 2 を、コンペアレジスタ 4 ~ 7 はフリーランタイム 1 を使用します。

なお、フリーランタイム 1 はインプットキャプチャでも使用します。

付表 A-1 I/O マップ (5/6)

アドレス	レジスタ	略称	アクセス	リソース	初期値
003900 _H	リロードレジスタ L	PRLLO	R/W	PPG(ch0, ch1) ユニット 0	XXXXXXXX _B
003901 _H	リロードレジスタ H	PRLH0	R/W		XXXXXXXX _B
003902 _H	リロードレジスタ L	PRLLO1	R/W		XXXXXXXX _B
003903 _H	リロードレジスタ H	PRLH1	R/W		XXXXXXXX _B
003904 _H	リロードレジスタ L	PRLLO2	R/W	PPG(ch2, ch3) ユニット 1	XXXXXXXX _B
003905 _H	リロードレジスタ H	PRLH2	R/W		XXXXXXXX _B
003906 _H	リロードレジスタ L	PRLLO3	R/W		XXXXXXXX _B
003907 _H	リロードレジスタ H	PRLH3	R/W		XXXXXXXX _B
003908 _H	リロードレジスタ L	PRLLO4	R/W	PPG(ch4, ch5) ユニット 2	XXXXXXXX _B
003909 _H	リロードレジスタ H	PRLH4	R/W		XXXXXXXX _B
00390A _H	リロードレジスタ L	PRLLO5	R/W		XXXXXXXX _B
00390B _H	リロードレジスタ H	PRLH5	R/W		XXXXXXXX _B
00390C _H	リロードレジスタ L	PRLLO6	R/W	PPG(ch6, ch7) ユニット 3	XXXXXXXX _B
00390D _H	リロードレジスタ H	PRLH6	R/W		XXXXXXXX _B
00390E _H	リロードレジスタ L	PRLLO7	R/W		XXXXXXXX _B
00390F _H	リロードレジスタ H	PRLH7	R/W		XXXXXXXX _B
003910 _H ~ 003917 _H	予約領域				
003918 _H	インプットキャプチャデータレジスタ 0	IPCP0	R	インプット キャプチャ 0/1	XXXXXXXX _B
003919 _H	インプットキャプチャデータレジスタ 0	IPCP0	R		XXXXXXXX _B
00391A _H	インプットキャプチャデータレジスタ 1	IPCP1	R		XXXXXXXX _B
00391B _H	インプットキャプチャデータレジスタ 1	IPCP1	R		XXXXXXXX _B
00391C _H	インプットキャプチャデータレジスタ 2	IPCP2	R	インプット キャプチャ 2/3	XXXXXXXX _B
00391D _H	インプットキャプチャデータレジスタ 2	IPCP2	R		XXXXXXXX _B
00391E _H	インプットキャプチャデータレジスタ 3	IPCP3	R		XXXXXXXX _B
00391F _H	インプットキャプチャデータレジスタ 3	IPCP3	R		XXXXXXXX _B
003920 _H	インプットキャプチャデータレジスタ 4	IPCP4	R	インプット キャプチャ 4/5	XXXXXXXX _B
003921 _H	インプットキャプチャデータレジスタ 4	IPCP4	R		XXXXXXXX _B
003922 _H	インプットキャプチャデータレジスタ 5	IPCP5	R		XXXXXXXX _B
003923 _H	インプットキャプチャデータレジスタ 5	IPCP5	R		XXXXXXXX _B
003924 _H	インプットキャプチャデータレジスタ 6	IPCP6	R	インプット キャプチャ 6/7	XXXXXXXX _B
003925 _H	インプットキャプチャデータレジスタ 6	IPCP6	R		XXXXXXXX _B
003926 _H	インプットキャプチャデータレジスタ 7	IPCP7	R		XXXXXXXX _B
003927 _H	インプットキャプチャデータレジスタ 7	IPCP7	R		XXXXXXXX _B
003928 _H	アウトプットコンペアレジスタ 0	OCCP0	R/W	アウト プット コンペア 0/1	XXXXXXXX _B
003929 _H	アウトプットコンペアレジスタ 0	OCCP0	R/W		XXXXXXXX _B
00392A _H	アウトプットコンペアレジスタ 1	OCCP1	R/W		XXXXXXXX _B
00392B _H	アウトプットコンペアレジスタ 1	OCCP1	R/W		XXXXXXXX _B
00392C _H	アウトプットコンペアレジスタ 2	OCCP2	R/W	アウト プット コンペア 2/3	XXXXXXXX _B
00392D _H	アウトプットコンペアレジスタ 2	OCCP2	R/W		XXXXXXXX _B
00392E _H	アウトプットコンペアレジスタ 3	OCCP3	R/W		XXXXXXXX _B
00392F _H	アウトプットコンペアレジスタ 3	OCCP3	R/W		XXXXXXXX _B
003930 _H ~ 0039FF _H	(予約領域)				
003A00 _H ~ 003AFF _H	(予約領域 :CAN0 インタフェース)				
003B00 _H ~ 003BFF _H	(予約領域 :CAN0 インタフェース)				

付表 A-1 I/O マップ (6 / 6)

アドレス	レジスタ	略称	アクセス	リソース	初期値
003C00 _H ~ 003CF _{FH}	(予約領域 :CAN1 インタフェース)				
003D00 _H ~ 003DFF _H	(予約領域 :CAN1 インタフェース)				
003E00 _H ~ 003FFF _H	(予約領域)				

- ・ 書込み / 読出しについての説明
 - R/W: リード・ライト可能
 - R: リードオンリ
 - W: ライトオンリ
- ・ 初期値についての説明
 - 0: このビットの初期値は "0" です。
 - 1: このビットの初期値は "1" です。
 - X: このビットの初期値は不定です。
 - : このビットは未使用です。初期値は不定です。

< 注意事項 >

I/O マップの予約領域への書込みは行わないでください。予約領域を読み出した場合は , 不定が読み出されます。

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 (rlst)
- 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。

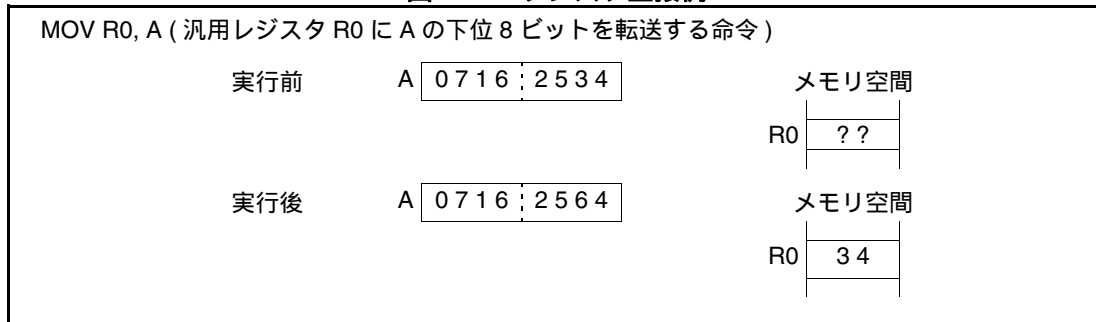
表 B.3-1 レジスタ直接

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。

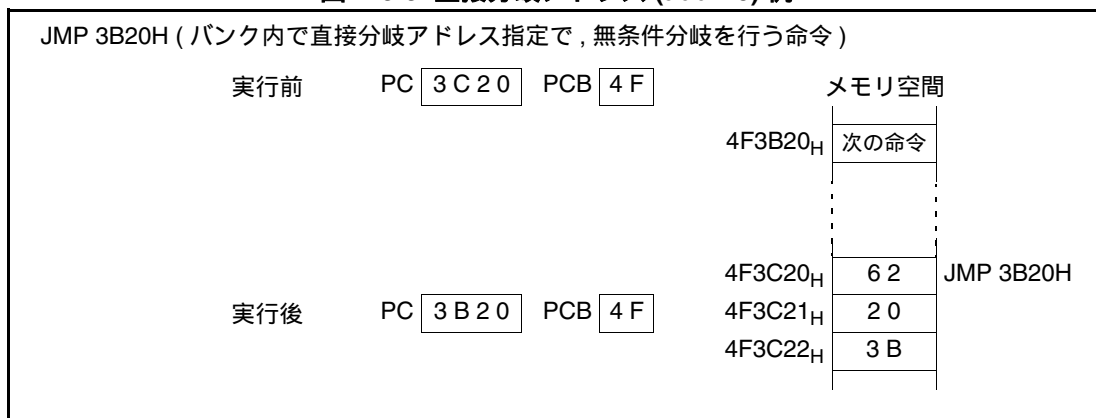
図 B.3-2 レジスタ直接例



● 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。

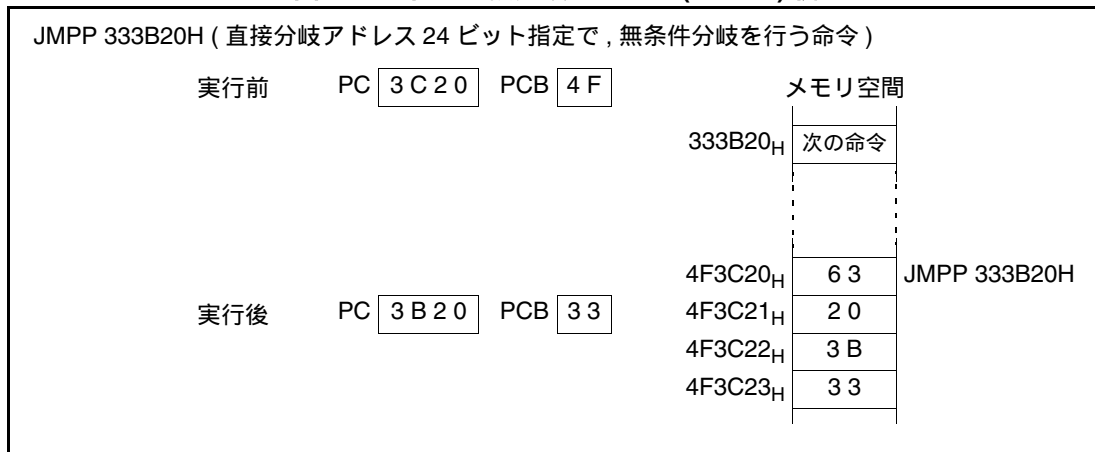
図 B.3-3 直接分岐アドレス (addr16) 例



● 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。

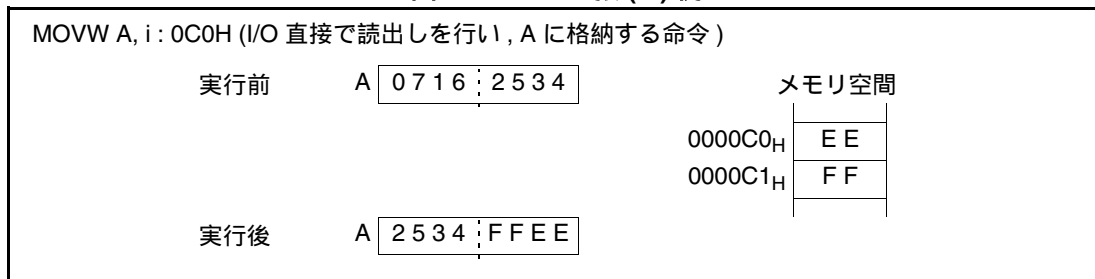
図 B.3-4 物理直接分岐アドレス (addr24) 例



● 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。

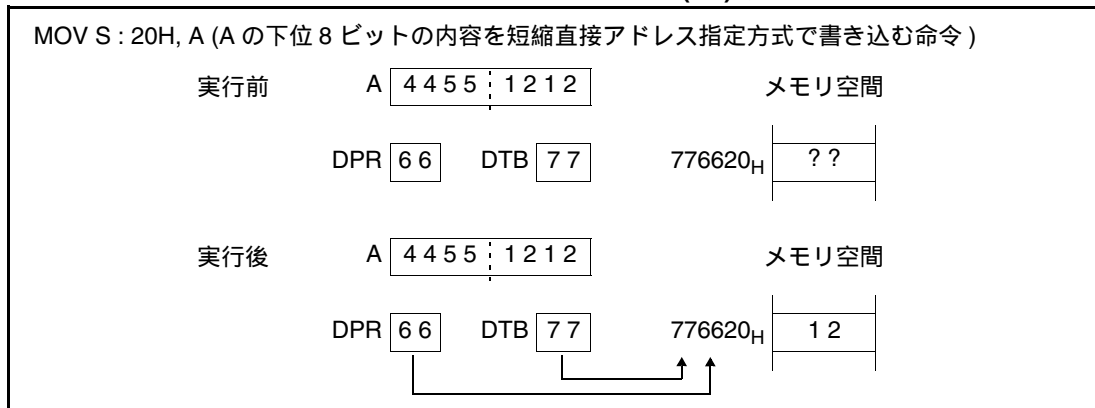
図 B.3-5 I/O 直接 (io) 例



● 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。

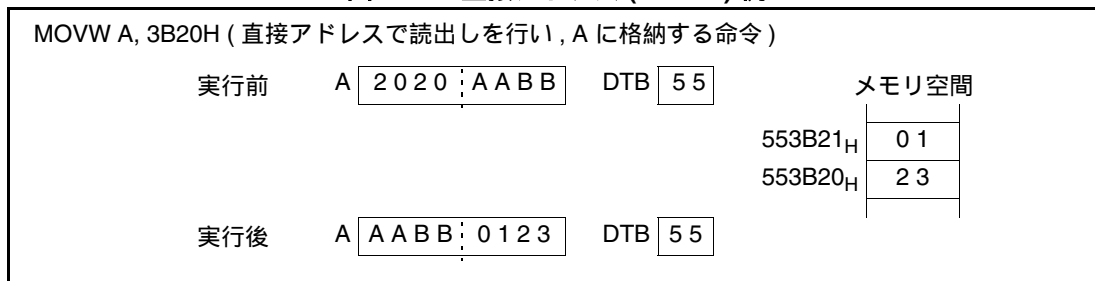
図 B.3-6 短縮直接アドレス (dir) 例



● 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。

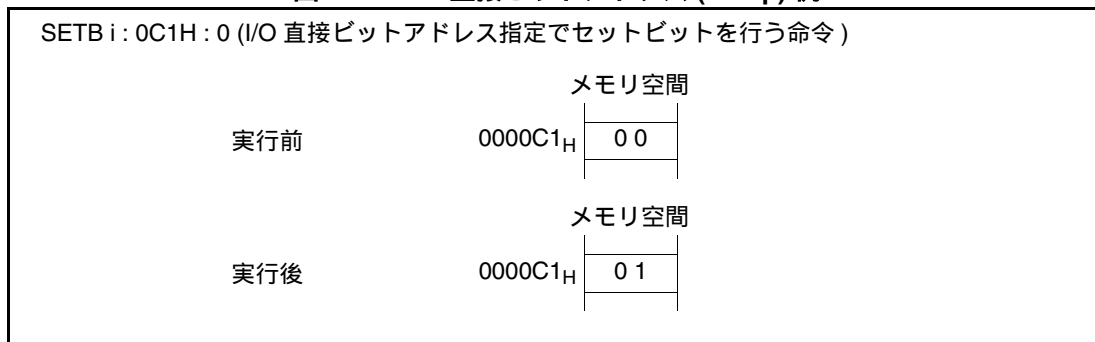
図 B.3-7 直接アドレス (addr16) 例



● 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。

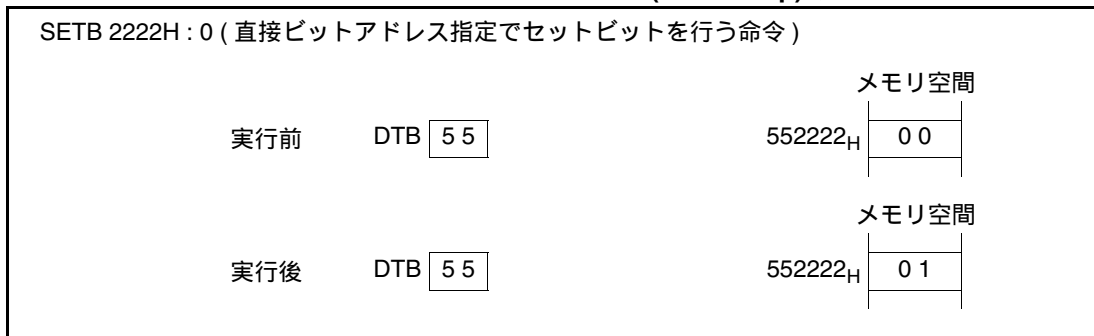
図 B.3-9 短縮直接ビットアドレス (dir: bp) 例



● 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

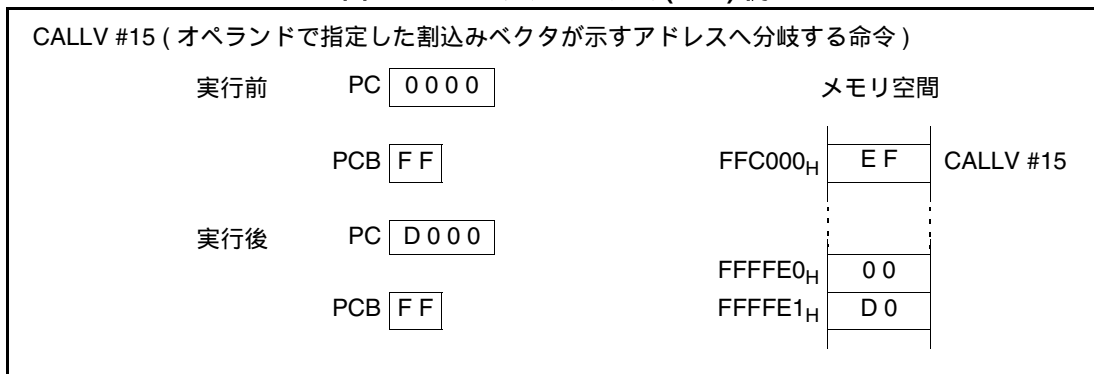


表 B.3-2 CALLV ベクター一覧表

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

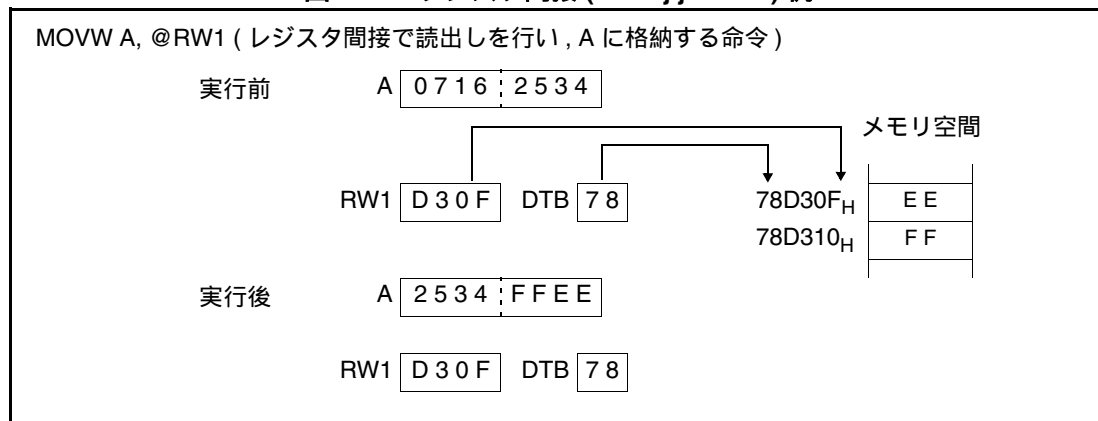
間接アドレッシングでは、記述したオペランドが示すアドレスのデータで、間接的にアドレスを指定します。

■ 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



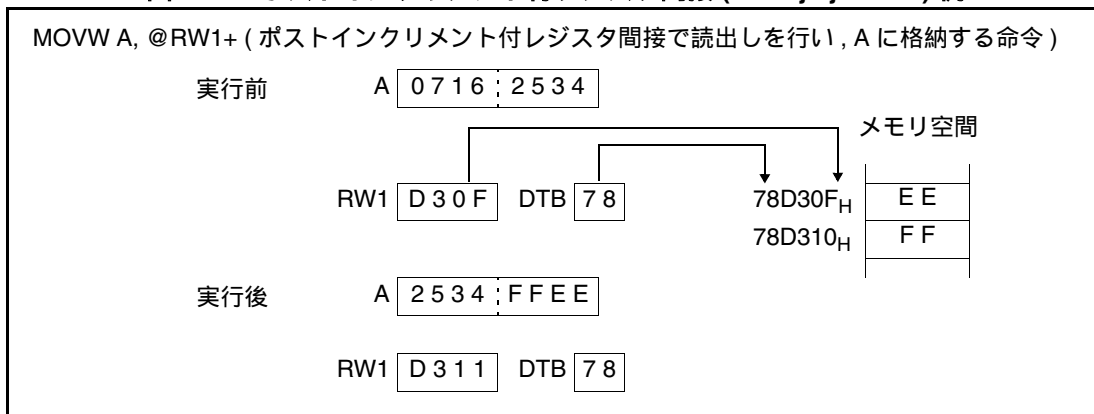
● ポストインクリメント付レジスタ間接 (@RWj+ j=0 ~ 3)

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、命令が書き込みだった場合は命令による書き込みが優先されるので、インクリメントするはずだったレジスタは書き込みデータとなります。

図 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

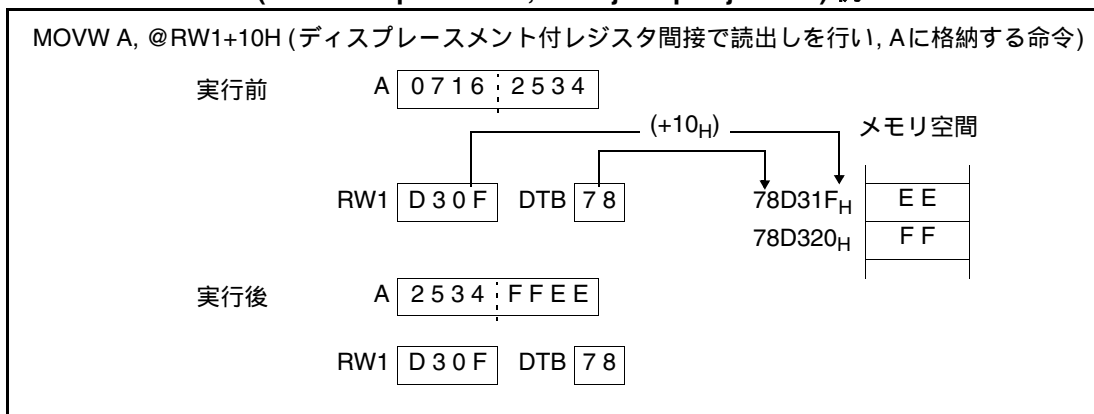


● ディスプレースメント付レジスタ間接

(@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。

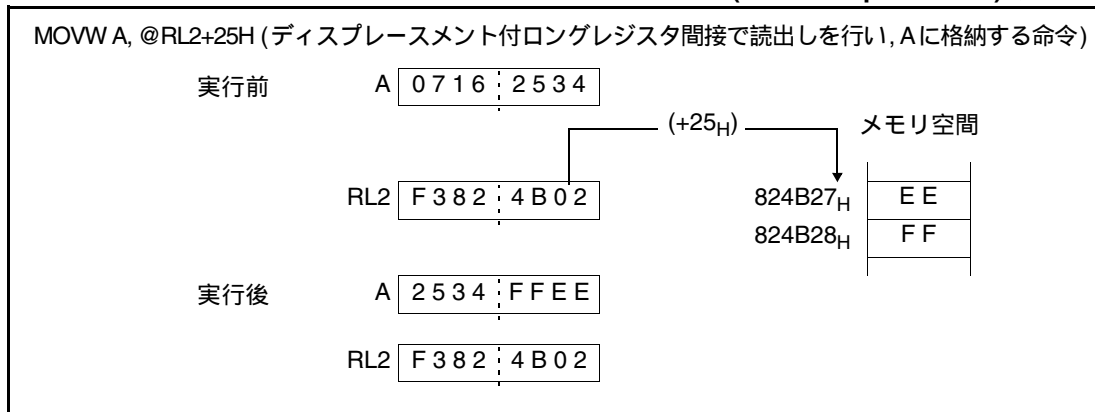
図 B.4-3 ディスプレースメント付レジスタ間接
(@RWi+disp8 i=0 ~ 7, @RWj+disp16 j=0 ~ 3) 例



● 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



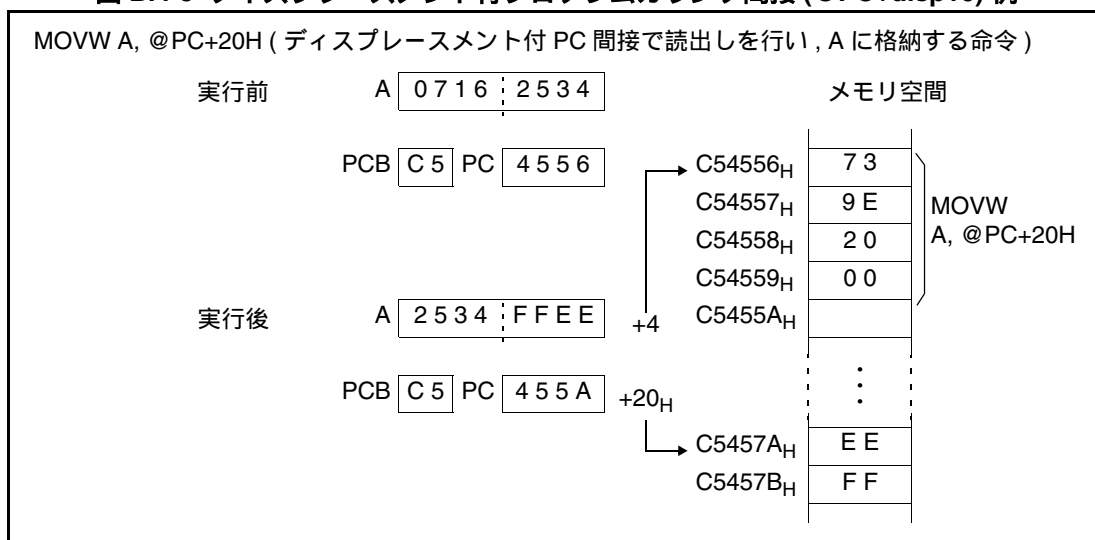
● ディスプレースメント付プログラムカウンタ間接 (@PC+disp16)

(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。

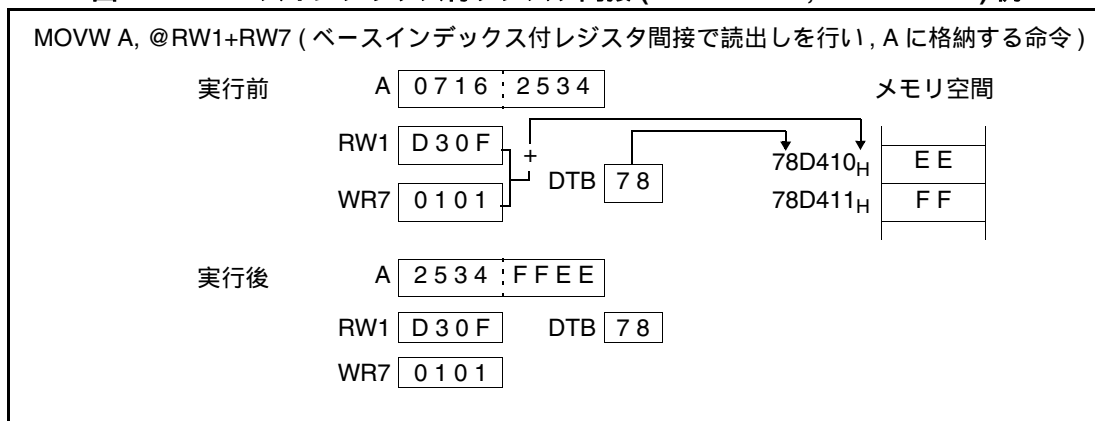
図 B.4-5 ディスプレースメント付プログラムカウンタ間接 (@PC+disp16) 例



● 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

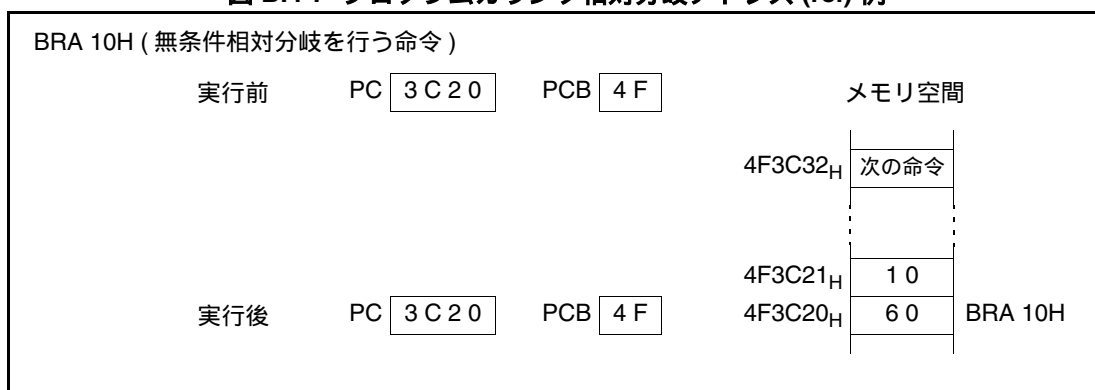


● プログラムカウンタ相対分岐アドレス (rel)

分岐先のアドレスはプログラムカウンタ (PC) の値と 8 ビットのディスプレースメントを加算した値となります。加算の結果が 16 ビットを超えた場合は, バンクレジスタのインクリメントまたはデクレメントをせずに, 超えた分は無視されますので, 64K バイトのバンク内で閉じたアドレスとなります。プログラムカウンタ相対分岐アドレスによるアドレッシングは, 無条件 / 条件分岐命令に使用します。アドレスの bit23 ~ bit16 はプログラムカウンタバンクレジスタ (PCB) で示されます。

図 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

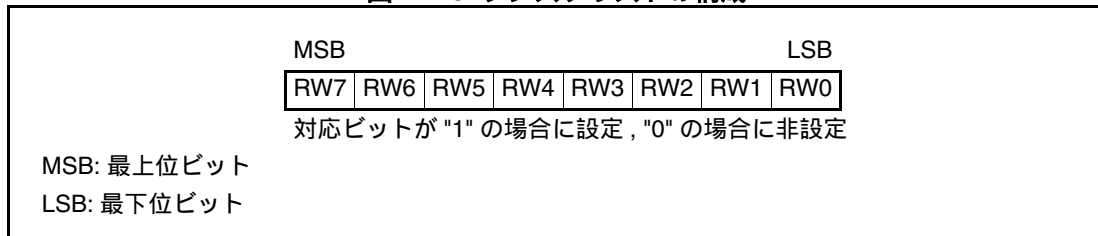
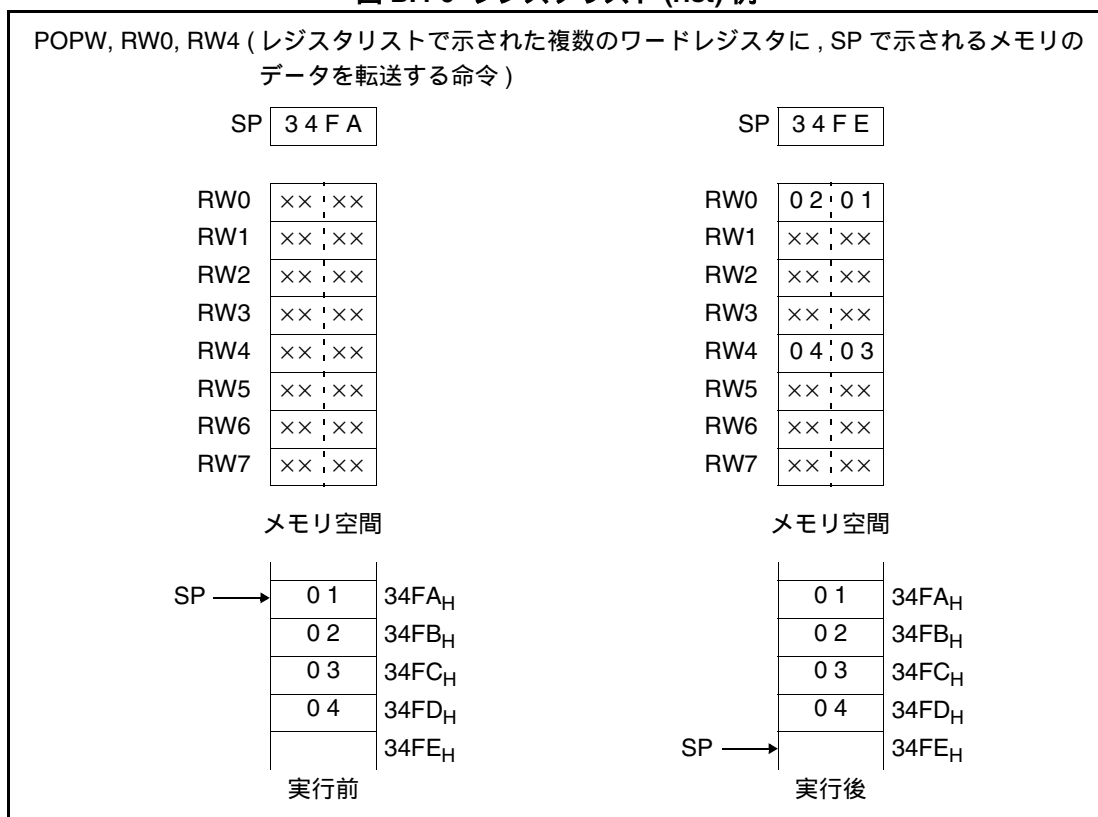


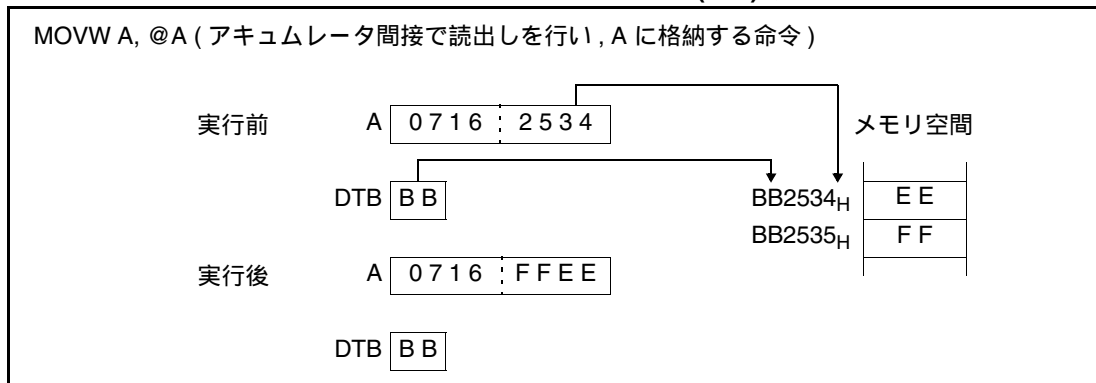
図 B.4-9 レジスタリスト (rlst) 例



● 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。

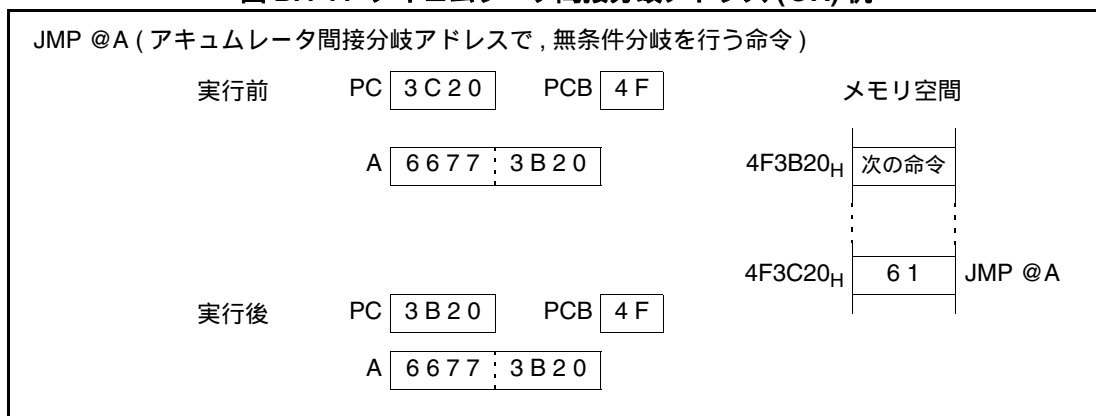
図 B.4-10 アキュムレータ間接 (@A) 例



● 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。

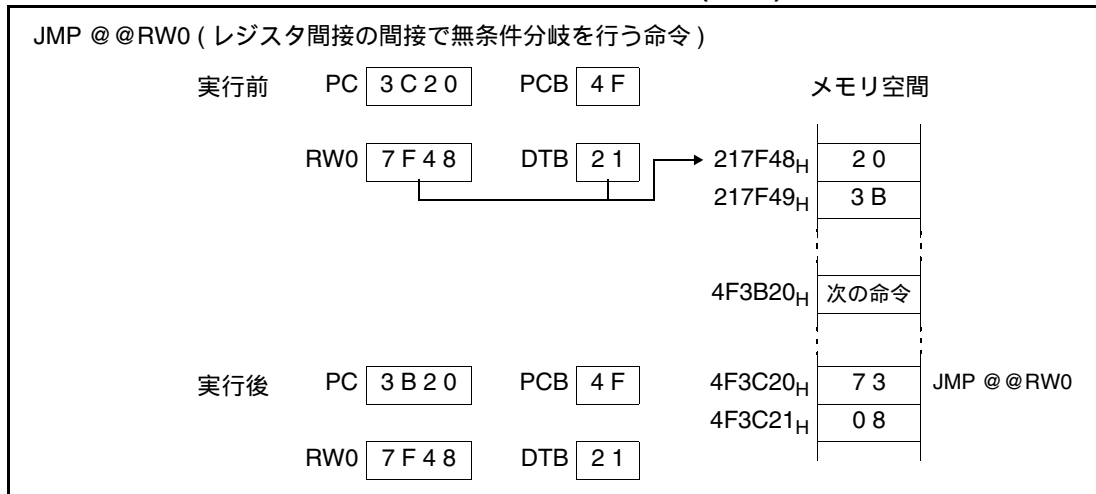
図 B.4-11 アキュムレータ間接分岐アドレス (@A) 例



● 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。

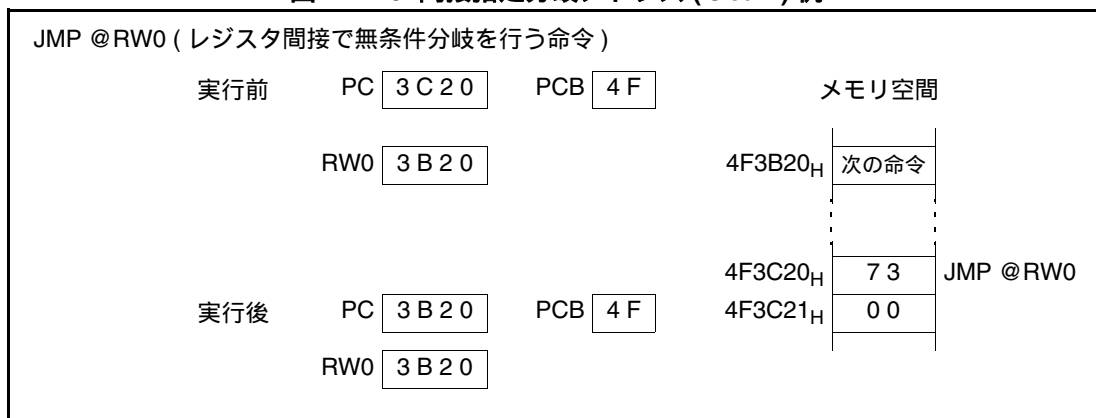
図 B.4-12 間接指定分岐アドレス (@ear) 例



● 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 ~ 07)
eam	実効アドレス指定 (コード 08 ~ 1F)
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。

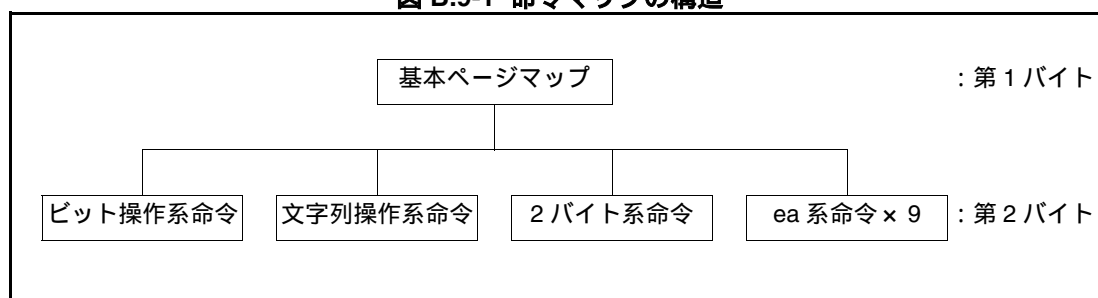
B.9 命令マップ

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

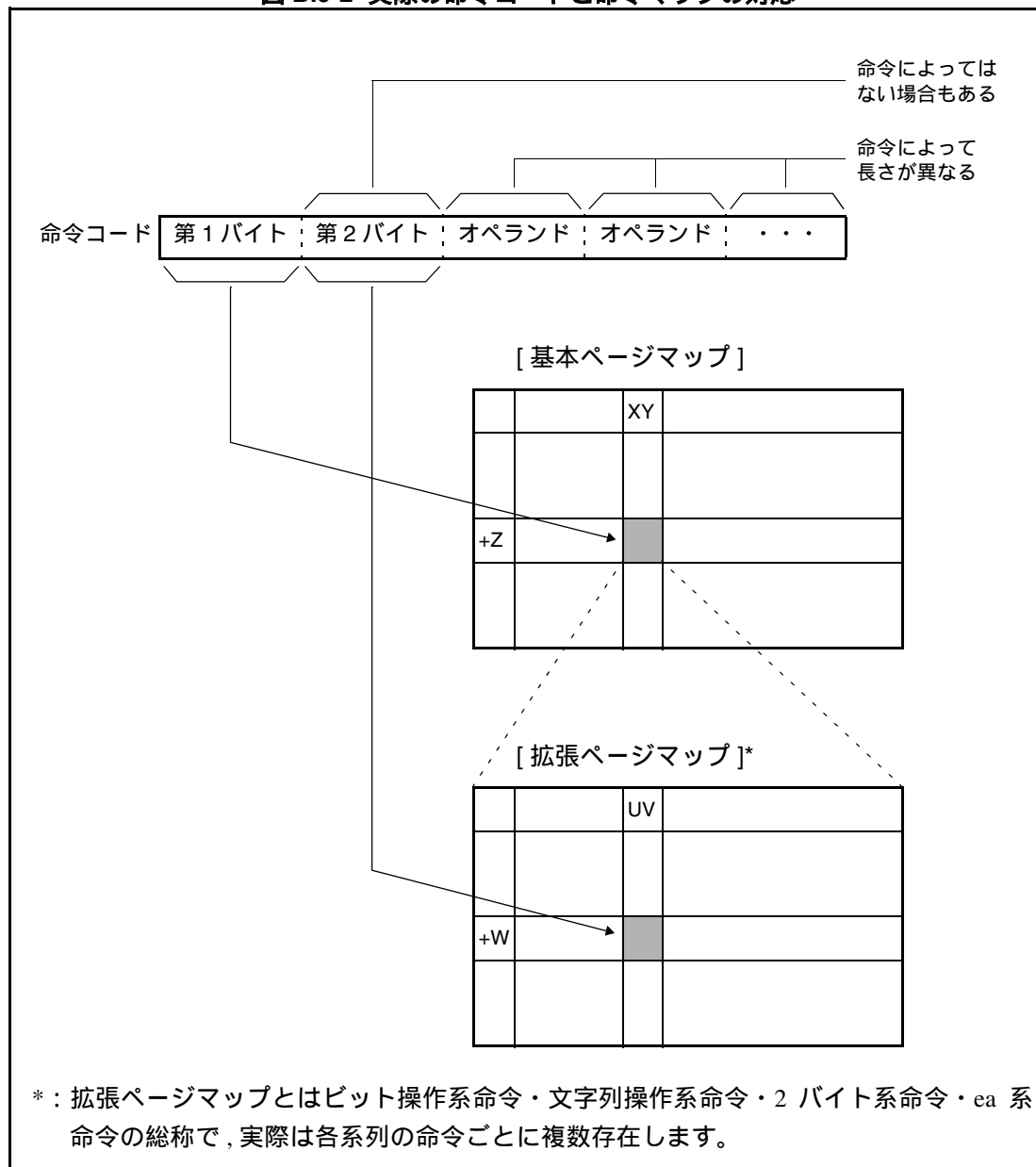
図 B.9-1 命令マップの構造



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	$60 + F = 6F$	$00 + 0 = 00$
@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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BZ/BEQ rel
+1	INT9	NCC	SUB A, dir	SUB A, #8	MOV dir, A	MOV io, A	JMP @A	ea 系命令 その 2	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BNZ/BNL rel
+2	ADDCA	SUBCA	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BNC/BHS rel
+4	PCB	EXT	AND CCR, #8	AND A, #8	MOV dir, #8	MOV io, #8	CALL addr16	ea 系命令 その 5	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BV rel
+7	SPB	ADDSP #8	MULU #8	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BNV rel
+8	LINK #imm8	ADDL A, #32	ADDW A	ADDW A, #16	MOVW A, dir	MOVW A, io	INT #vct8	ea 系命令 その 9	MOVW A, RWI	MOVW RWI, A	MOVW RWI, #16	MOVW A, RWI	MOVW @RWI+d8 @RWI+d8,A	MOV A, #4	CALL #4	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BNT rel
+A	MOV RP, #8	MOV ILM, #8	CBNE A, #8, rel	CBNE A, #16, rel	MOVW A, #16	MOVW A, addr16	INTP	MOV Ri, ea	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	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	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BGT rel
+E	ASRW A	SWAPW	XORW A	XORW A, #16	PUSHW PS	POPW PS	文字列操 作系命令	XCH Ri, ea	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	BLS rel
+F	LSRW A	ADDSP #16	MULW A	NOTW A	PUSHW r1st	POPW r1st	2 バイト 系命令	XCHW RWI, ea	MOV A, Ri	MOV Ri, A	MOV Ri, #8	MOV A, Ri	MOVX A, @RWI+d8	MOV A, #4	CALL #4	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	MOVB A, io:bp		MOVB io:bp, A		CLRB io:bp		SETB io:bp		BBC io:bp, rel		BBS io:bp, rel		WBTS io:bp		WBTC io:bp	
+1																
+2																
+3																
+4																
+5																
+6																
+7																
+8	MOVB A, dir:bp	MOVB A, addr16:bp	MOVB dir:bp, A	MOVB 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				SBBS 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								SCWEQI, SCEQD, SCWEQD, FILSI	PCB, DTB, ADB, SPB					FILSI, PCB, DTB, ADB, SPB	
+1	PCB, DTB															
+2	PCB, ADB															
+3	PCB, SPB															
+4	DTB, PCB															
+5	DTB, DTB															
+6	DTB, ADB															
+7	DTB, SPB															
+8	ADB, PCB															
+9	ADB, DTB															
+A	ADB, ADB															
+B	ADB, SPB															
+C	SPB, PCB															
+D	SPB, DTB															
+E	SPB, ADB															
+F	SPB, SPB															

表 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 A, @RL0+d8											
+1	MOV A, ADB	MOV ADB, A														
+2	MOV A, SSB	MOV SSB, A	MOVX A, @RL1+d8	MOV @RL1+d8, A	MOV A, @RL1+d8											
+3	MOV A, USB	MOV USB, A														
+4	MOV A, DPR	MOV DPR, A	MOVX A, @RL2+d8	MOV @RL2+d8, A	MOV A, @RL2+d8											
+5	MOV A, @A	MOV @AL, AH														
+6	MOV A, PCB	MOV A, @A	MOVX A, @RL3+d8	MOV @RL3+d8, A	MOV A, @RL3+d8											
+7	ROL A	ROL A														
+8				MOVW @RL0+d8, A	MOVW A, @RL0+d8			MUL A								
+9								MULW A								
+A				MOVW @RL1+d8, A	MOVW A, @RL1+d8			DIVU A								
+B																
+C	LSLW A, R0	LSLL A, R0	LSL A, R0	MOVW @RL2+d8, A	MOVW A, @RL2+d8											
+D	MOVW A, @A	MOVW @AL, AH	NRML A, R0													
+E	ASRW A, R0	ASRL A, R0	ASR A, R0	MOVW @RL3+d8, A	MOVW A, @RL3+d8											
+F	LSRW A, R0	LSRL A, R0	LSR A, R0													

表 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
					CBNE↓	CBNE↓									CBNE↓	
+0	ADDL A, A, RL0, @RW0+d8	SUBL A, A, RL0, @RW0+d8	SUBL A, A, RL0, @RW0+d8	SUBL A, A, RL0, @RW0+d8	RW0, @RW0+d8 #16, rel	CMPL A, A, RL0, @RW0+d8	CMPL A, A, RL0, @RW0+d8	CMPL A, A, RL0, @RW0+d8	ANDL A, A, RL0, @RW0+d8	ANDL A, A, RL0, @RW0+d8	ORL A, A, RL0, @RW0+d8	ORL A, A, RL0, @RW0+d8	XORL A, A, RL0, @RW0+d8	XORL A, A, RL0, @RW0+d8	R0, @RW0+d8 #8, rel	R0, @RW0+d8 #8, rel
+1	ADDL A, A, RL0, @RW1+d8	SUBL A, A, RL0, @RW1+d8	SUBL A, A, RL0, @RW1+d8	SUBL A, A, RL0, @RW1+d8	RW1, @RW1+d8 #16, rel	CMPL A, A, RL0, @RW1+d8	CMPL A, A, RL0, @RW1+d8	CMPL A, A, RL0, @RW1+d8	ANDL A, A, RL0, @RW1+d8	ANDL A, A, RL0, @RW1+d8	ORL A, A, RL0, @RW1+d8	ORL A, A, RL0, @RW1+d8	XORL A, A, RL0, @RW1+d8	XORL A, A, RL0, @RW1+d8	R1, @RW1+d8 #8, rel	R1, @RW1+d8 #8, rel
+2	ADDL A, A, RL1, @RW2+d8	SUBL A, A, RL1, @RW2+d8	SUBL A, A, RL1, @RW2+d8	SUBL A, A, RL1, @RW2+d8	RW2, @RW2+d8 #16, rel	CMPL A, A, RL1, @RW2+d8	CMPL A, A, RL1, @RW2+d8	CMPL A, A, RL1, @RW2+d8	ANDL A, A, RL1, @RW2+d8	ANDL A, A, RL1, @RW2+d8	ORL A, A, RL1, @RW2+d8	ORL A, A, RL1, @RW2+d8	XORL A, A, RL1, @RW2+d8	XORL A, A, RL1, @RW2+d8	R2, @RW2+d8 #8, rel	R2, @RW2+d8 #8, rel
+3	ADDL A, A, RL1, @RW3+d8	SUBL A, A, RL1, @RW3+d8	SUBL A, A, RL1, @RW3+d8	SUBL A, A, RL1, @RW3+d8	RW3, @RW3+d8 #16, rel	CMPL A, A, RL1, @RW3+d8	CMPL A, A, RL1, @RW3+d8	CMPL A, A, RL1, @RW3+d8	ANDL A, A, RL1, @RW3+d8	ANDL A, A, RL1, @RW3+d8	ORL A, A, RL1, @RW3+d8	ORL A, A, RL1, @RW3+d8	XORL A, A, RL1, @RW3+d8	XORL A, A, RL1, @RW3+d8	R3, @RW3+d8 #8, rel	R3, @RW3+d8 #8, rel
+4	ADDL A, A, RL2, @RW4+d8	SUBL A, A, RL2, @RW4+d8	SUBL A, A, RL2, @RW4+d8	SUBL A, A, RL2, @RW4+d8	RW4, @RW4+d8 #16, rel	CMPL A, A, RL2, @RW4+d8	CMPL A, A, RL2, @RW4+d8	CMPL A, A, RL2, @RW4+d8	ANDL A, A, RL2, @RW4+d8	ANDL A, A, RL2, @RW4+d8	ORL A, A, RL2, @RW4+d8	ORL A, A, RL2, @RW4+d8	XORL A, A, RL2, @RW4+d8	XORL A, A, RL2, @RW4+d8	R4, @RW4+d8 #8, rel	R4, @RW4+d8 #8, rel
+5	ADDL A, A, RL2, @RW5+d8	SUBL A, A, RL2, @RW5+d8	SUBL A, A, RL2, @RW5+d8	SUBL A, A, RL2, @RW5+d8	RW5, @RW5+d8 #16, rel	CMPL A, A, RL2, @RW5+d8	CMPL A, A, RL2, @RW5+d8	CMPL A, A, RL2, @RW5+d8	ANDL A, A, RL2, @RW5+d8	ANDL A, A, RL2, @RW5+d8	ORL A, A, RL2, @RW5+d8	ORL A, A, RL2, @RW5+d8	XORL A, A, RL2, @RW5+d8	XORL A, A, RL2, @RW5+d8	R5, @RW5+d8 #8, rel	R5, @RW5+d8 #8, rel
+6	ADDL A, A, RL3, @RW6+d8	SUBL A, A, RL3, @RW6+d8	SUBL A, A, RL3, @RW6+d8	SUBL A, A, RL3, @RW6+d8	RW6, @RW6+d8 #16, rel	CMPL A, A, RL3, @RW6+d8	CMPL A, A, RL3, @RW6+d8	CMPL A, A, RL3, @RW6+d8	ANDL A, A, RL3, @RW6+d8	ANDL A, A, RL3, @RW6+d8	ORL A, A, RL3, @RW6+d8	ORL A, A, RL3, @RW6+d8	XORL A, A, RL3, @RW6+d8	XORL A, A, RL3, @RW6+d8	R6, @RW6+d8 #8, rel	R6, @RW6+d8 #8, rel
+7	ADDL A, A, RL3, @RW7+d8	SUBL A, A, RL3, @RW7+d8	SUBL A, A, RL3, @RW7+d8	SUBL A, A, RL3, @RW7+d8	RW7, @RW7+d8 #16, rel	CMPL A, A, RL3, @RW7+d8	CMPL A, A, RL3, @RW7+d8	CMPL A, A, RL3, @RW7+d8	ANDL A, A, RL3, @RW7+d8	ANDL A, A, RL3, @RW7+d8	ORL A, A, RL3, @RW7+d8	ORL A, A, RL3, @RW7+d8	XORL A, A, RL3, @RW7+d8	XORL A, A, RL3, @RW7+d8	R7, @RW7+d8 #8, rel	R7, @RW7+d8 #8, rel
+8	ADDL A, A, @RW0, @RW0+d16	SUBL A, A, @RW0, @RW0+d16	SUBL A, A, @RW0, @RW0+d16	SUBL A, A, @RW0, @RW0+d16	@RW0, @RW0+d16 #16, rel	CMPL A, A, @RW0, @RW0+d16	CMPL A, A, @RW0, @RW0+d16	CMPL A, A, @RW0, @RW0+d16	ANDL A, A, @RW0, @RW0+d16	ANDL A, A, @RW0, @RW0+d16	ORL A, A, @RW0, @RW0+d16	ORL A, A, @RW0, @RW0+d16	XORL A, A, @RW0, @RW0+d16	XORL A, A, @RW0, @RW0+d16	@RW0, @RW0+d16 #8, rel	@RW0, @RW0+d16 #8, rel
+9	ADDL A, A, @RW1, @RW1+d16	SUBL A, A, @RW1, @RW1+d16	SUBL A, A, @RW1, @RW1+d16	SUBL A, A, @RW1, @RW1+d16	@RW1, @RW1+d16 #16, rel	CMPL A, A, @RW1, @RW1+d16	CMPL A, A, @RW1, @RW1+d16	CMPL A, A, @RW1, @RW1+d16	ANDL A, A, @RW1, @RW1+d16	ANDL A, A, @RW1, @RW1+d16	ORL A, A, @RW1, @RW1+d16	ORL A, A, @RW1, @RW1+d16	XORL A, A, @RW1, @RW1+d16	XORL A, A, @RW1, @RW1+d16	@RW1, @RW1+d16 #8, rel	@RW1, @RW1+d16 #8, rel
+A	ADDL A, A, @RW2, @RW2+d16	SUBL A, A, @RW2, @RW2+d16	SUBL A, A, @RW2, @RW2+d16	SUBL A, A, @RW2, @RW2+d16	@RW2, @RW2+d16 #16, rel	CMPL A, A, @RW2, @RW2+d16	CMPL A, A, @RW2, @RW2+d16	CMPL A, A, @RW2, @RW2+d16	ANDL A, A, @RW2, @RW2+d16	ANDL A, A, @RW2, @RW2+d16	ORL A, A, @RW2, @RW2+d16	ORL A, A, @RW2, @RW2+d16	XORL A, A, @RW2, @RW2+d16	XORL A, A, @RW2, @RW2+d16	@RW2, @RW2+d16 #8, rel	@RW2, @RW2+d16 #8, rel
+B	ADDL A, A, @RW3, @RW3+d16	SUBL A, A, @RW3, @RW3+d16	SUBL A, A, @RW3, @RW3+d16	SUBL A, A, @RW3, @RW3+d16	@RW3, @RW3+d16 #16, rel	CMPL A, A, @RW3, @RW3+d16	CMPL A, A, @RW3, @RW3+d16	CMPL A, A, @RW3, @RW3+d16	ANDL A, A, @RW3, @RW3+d16	ANDL A, A, @RW3, @RW3+d16	ORL A, A, @RW3, @RW3+d16	ORL A, A, @RW3, @RW3+d16	XORL A, A, @RW3, @RW3+d16	XORL A, A, @RW3, @RW3+d16	@RW3, @RW3+d16 #8, rel	@RW3, @RW3+d16 #8, rel
+C	ADDL A, A, @RW0+1, @RW0+1	SUBL A, A, @RW0+1, @RW0+1	SUBL A, A, @RW0+1, @RW0+1	SUBL A, A, @RW0+1, @RW0+1	@RW0+1, @RW0+1 #16, rel	CMPL A, A, @RW0+1, @RW0+1	CMPL A, A, @RW0+1, @RW0+1	CMPL A, A, @RW0+1, @RW0+1	ANDL A, A, @RW0+1, @RW0+1	ANDL A, A, @RW0+1, @RW0+1	ORL A, A, @RW0+1, @RW0+1	ORL A, A, @RW0+1, @RW0+1	XORL A, A, @RW0+1, @RW0+1	XORL A, A, @RW0+1, @RW0+1	@RW0+1, @RW0+1 #8, rel	@RW0+1, @RW0+1 #8, rel
+D	ADDL A, A, @RW1+1, @RW1+1	SUBL A, A, @RW1+1, @RW1+1	SUBL A, A, @RW1+1, @RW1+1	SUBL A, A, @RW1+1, @RW1+1	@RW1+1, @RW1+1 #16, rel	CMPL A, A, @RW1+1, @RW1+1	CMPL A, A, @RW1+1, @RW1+1	CMPL A, A, @RW1+1, @RW1+1	ANDL A, A, @RW1+1, @RW1+1	ANDL A, A, @RW1+1, @RW1+1	ORL A, A, @RW1+1, @RW1+1	ORL A, A, @RW1+1, @RW1+1	XORL A, A, @RW1+1, @RW1+1	XORL A, A, @RW1+1, @RW1+1	@RW1+1, @RW1+1 #8, rel	@RW1+1, @RW1+1 #8, rel
+E	ADDL A, A, @PC+d16	SUBL A, A, @PC+d16	SUBL A, A, @PC+d16	SUBL A, A, @PC+d16	@PC+d16, @PC+d16 #16, rel	CMPL A, A, @PC+d16, @PC+d16	CMPL A, A, @PC+d16, @PC+d16	CMPL A, A, @PC+d16, @PC+d16	ANDL A, A, @PC+d16, @PC+d16	ANDL A, A, @PC+d16, @PC+d16	ORL A, A, @PC+d16, @PC+d16	ORL A, A, @PC+d16, @PC+d16	XORL A, A, @PC+d16, @PC+d16	XORL A, A, @PC+d16, @PC+d16	@PC+d16, @PC+d16 #8, rel	@PC+d16, @PC+d16 #8, rel
+F	ADDL A, A, @RW3+, addr16	SUBL A, A, @RW3+, addr16	SUBL A, A, @RW3+, addr16	SUBL A, A, @RW3+, addr16	addr16, @RW3+, #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	@RW3+, addr16 #8, rel	@RW3+, 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 R1, #8, @RW1+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 R2, #8, @RW2+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 R3, #8, @RW3+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 R4, #8, @RW4+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 R5, #8, @RW5+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 R6, #8, @RW6+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 R7, #8, @RW7+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 R8, #8, @RW8+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 @RW1, #8, @RW1+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 @RW2, #8, @RW2+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 @RW3, #8, @RW3+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 @RW4, #8, @RW4+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 @RW1+, #8, @RW1+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 @RW2+, #8, @RW2+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 @RW3+, #8, @RW3+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 @RW4+, #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	ROR	R0' @RW0+d8	INC	R0' @RW0+d8	DEC	R0' @RW0+d8	MOV	A, R0' @RW0+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+1	ROL	R1' @RW1+d8	ROR	R1' @RW1+d8	INC	R1' @RW1+d8	DEC	R1' @RW1+d8	MOV	A, R1' @RW1+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+2	ROL	R2' @RW2+d8	ROR	R2' @RW2+d8	INC	R2' @RW2+d8	DEC	R2' @RW2+d8	MOV	A, R2' @RW2+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+3	ROL	R3' @RW3+d8	ROR	R3' @RW3+d8	INC	R3' @RW3+d8	DEC	R3' @RW3+d8	MOV	A, R3' @RW3+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+4	ROL	R4' @RW4+d8	ROR	R4' @RW4+d8	INC	R4' @RW4+d8	DEC	R4' @RW4+d8	MOV	A, R4' @RW4+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+5	ROL	R5' @RW5+d8	ROR	R5' @RW5+d8	INC	R5' @RW5+d8	DEC	R5' @RW5+d8	MOV	A, R5' @RW5+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+6	ROL	R6' @RW6+d8	ROR	R6' @RW6+d8	INC	R6' @RW6+d8	DEC	R6' @RW6+d8	MOV	A, R6' @RW6+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+7	ROL	R7' @RW7+d8	ROR	R7' @RW7+d8	INC	R7' @RW7+d8	DEC	R7' @RW7+d8	MOV	A, R7' @RW7+d8	MOV	MOV	MOVX	MOVX	XCH	XCH
+8	ROL	R0' @RW0+d16	ROR	R0' @RW0+d16	INC	R0' @RW0+d16	DEC	R0' @RW0+d16	MOV	A, R0' @RW0+d16	MOV	MOV	MOVX	MOVX	XCH	XCH
+9	ROL	R1' @RW1+d16	ROR	R1' @RW1+d16	INC	R1' @RW1+d16	DEC	R1' @RW1+d16	MOV	A, R1' @RW1+d16	MOV	MOV	MOVX	MOVX	XCH	XCH
+A	ROL	R2' @RW2+d16	ROR	R2' @RW2+d16	INC	R2' @RW2+d16	DEC	R2' @RW2+d16	MOV	A, R2' @RW2+d16	MOV	MOV	MOVX	MOVX	XCH	XCH
+B	ROL	R3' @RW3+d16	ROR	R3' @RW3+d16	INC	R3' @RW3+d16	DEC	R3' @RW3+d16	MOV	A, R3' @RW3+d16	MOV	MOV	MOVX	MOVX	XCH	XCH
+C	ROL	R0' @RW0+R7	ROR	R0' @RW0+R7	INC	R0' @RW0+R7	DEC	R0' @RW0+R7	MOV	A, R0' @RW0+R7	MOV	MOV	MOVX	MOVX	XCH	XCH
+D	ROL	R1' @RW1+R7	ROR	R1' @RW1+R7	INC	R1' @RW1+R7	DEC	R1' @RW1+R7	MOV	A, R1' @RW1+R7	MOV	MOV	MOVX	MOVX	XCH	XCH
+E	ROL	R2' @RW2+R7	ROR	R2' @RW2+R7	INC	R2' @RW2+R7	DEC	R2' @RW2+R7	MOV	A, R2' @RW2+R7	MOV	MOV	MOVX	MOVX	XCH	XCH
+F	ROL	R3' @RW3+R7	ROR	R3' @RW3+R7	INC	R3' @RW3+R7	DEC	R3' @RW3+R7	MOV	A, R3' @RW3+R7	MOV	MOV	MOVX	MOVX	XCH	XCH

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, A	MOVW A, RW1, @RW1+d8, A	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, A	MOVW A, RW2, @RW2+d8, A	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, A	MOVW A, RW3, @RW3+d8, A	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, A	MOVW A, RW4, @RW4+d8, A	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, A	MOVW A, RW5, @RW5+d8, A	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, A	MOVW A, RW6, @RW6+d8, A	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, A	MOVW A, RW7, @RW7+d8, A	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, A	MOVW A, @RW0, @RW0+d16, A	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, A	MOVW A, @RW1, @RW1+d16, A	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, A	MOVW A, @RW2, @RW2+d16, A	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, A	MOVW A, @RW3, @RW3+d16, A	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, A	MOVW A, @RW0+, @RW0+RW7, A	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, A	MOVW A, @RW1+, @RW1+RW7, A	MOVW @RW1+, #16, @RW1+RW7, #16	MOVW @RW1+, #16, @RW1+RW7, #16	MOVW @RW1+, #16, @RW1+RW7, #16	MOVW @RW1+, #16, @RW1+RW7, #16	XCHW A, @RW1+, @RW1+RW7	XCHW A, @RW1+, @RW1+RW7
+E	JMP @RW2+, @PC+d16	JMP @RW2+, @PC+d16	CALL @RW2+, @PC+d16	CALL @RW2+, @PC+d16	INCW @RW2+, @PC+d16	INCW @RW2+, @PC+d16	DECW @RW2+, @PC+d16	DECW @RW2+, @PC+d16	MOVW A, @RW2+, @PC+d16, A	MOVW A, @RW2+, @PC+d16, A	MOVW @RW2+, #16, @PC+d16, #16	MOVW @RW2+, #16, @PC+d16, #16	MOVW @RW2+, #16, @PC+d16, #16	MOVW @RW2+, #16, @PC+d16, #16	XCHW A, @RW2+, @PC+d16	XCHW A, @RW2+, @PC+d16
+F	JMP @RW3+, @addr16	JMP @RW3+, @addr16	CALL @RW3+, @addr16	CALL @RW3+, @addr16	INCW @RW3+, @addr16	INCW @RW3+, @addr16	DECW @RW3+, @addr16	DECW @RW3+, @addr16	MOVW A, @RW3+, @addr16, A	MOVW A, @RW3+, @addr16, A	MOVW @RW3+, #16, @addr16, #16	MOVW @RW3+, #16, @addr16, #16	MOVW @RW3+, #16, @addr16, #16	MOVW @RW3+, #16, @addr16, #16	XCHW A, @RW3+, @addr16	XCHW A, @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, r, RW0+d8, r	DBNZ R0, r, RW0+d8, r
+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, r, RW1+d8, r	DBNZ R1, r, RW1+d8, r
+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, r, RW2+d8, r	DBNZ R2, r, RW2+d8, r
+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, r, RW3+d8, r	DBNZ R3, r, RW3+d8, r
+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, r, RW4+d8, r	DBNZ R4, r, RW4+d8, r
+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, r, RW5+d8, r	DBNZ R5, r, RW5+d8, r
+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, r, RW6+d8, r	DBNZ R6, r, RW6+d8, r
+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, r, RW7+d8, r	DBNZ R7, r, RW7+d8, r
+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, r, W0+d16, r	DBNZ @RW0, r, W0+d16, r
+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, r, W1+d16, r	DBNZ @RW1, r, W1+d16, r
+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, r, W2+d16, r	DBNZ @RW2, r, W2+d16, r
+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, r, W3+d16, r	DBNZ @RW3, r, W3+d16, r
+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+, r, W0+RW7, r	DBNZ @RW0+, r, W0+RW7, r
+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+, r, W1+RW7, r	DBNZ @RW1+, r, W1+RW7, r
+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+, r, PC+d16, r	DBNZ @RW2+, r, PC+d16, r
+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+, r, addr16, r	DBNZ @RW3+, r, addr16, r

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 A, 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 A, 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 A, 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 A, 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 A, 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 A, 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 A, 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 A, 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 A, @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 A, @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 A, @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 A, @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 A, @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 A, @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 A, @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 A, @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, r' @RW0+d8, r	DWBZ RW0, r' @RW0+d8, r
+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, r' @RW1+d8, r	DWBZ RW1, r' @RW1+d8, r
+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, r' @RW2+d8, r	DWBZ RW2, r' @RW2+d8, r
+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, r' @RW3+d8, r	DWBZ RW3, r' @RW3+d8, r
+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, r' @RW4+d8, r	DWBZ RW4, r' @RW4+d8, r
+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, r' @RW5+d8, r	DWBZ RW5, r' @RW5+d8, r
+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, r' @RW6+d8, r	DWBZ RW6, r' @RW6+d8, r
+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, r' @RW7+d8, r	DWBZ RW7, r' @RW7+d8, r
+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, r' @RW0+d16, r	DWBZ @RW0, r' @RW0+d16, r
+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, r' @RW1+d16, r	DWBZ @RW1, r' @RW1+d16, r
+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, r' @RW2+d16, r	DWBZ @RW2, r' @RW2+d16, r
+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, r' @RW3+d16, r	DWBZ @RW3, r' @RW3+d16, r
+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+, r' @RW0+RW7, r	DWBZ @RW0+, r' @RW0+RW7, r
+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+, r' @RW1+RW7, r	DWBZ @RW1+, r' @RW1+RW7, r
+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+, r' @PC+d16, r	DWBZ @RW2+, r' @PC+d16, r
+F	ADDW A, @RW3+, addr 16	ADDW A, @RW3+, addr 16	SUBW A, @RW3+, addr 16	SUBW A, @RW3+, addr 16	ADDCW A, @RW3+, addr 16	ADDCW A, @RW3+, addr 16	CMPW A, @RW3+, addr 16	CMPW A, @RW3+, addr 16	ANDW A, @RW3+, addr 16	ANDW A, @RW3+, addr 16	ORW A, @RW3+, addr 16	ORW A, @RW3+, addr 16	XORW A, @RW3+, addr 16	XORW A, @RW3+, addr 16	DWBZ @RW3+, r' addr 16, r	DWBZ @RW3+, r' addr 16, r

表 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	NEGW RW0', @RW0+d8	NEGW 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	NEGW RW1', @RW1+d8	NEGW 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	NEGW RW2', @RW2+d8	NEGW 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	NEGW RW3', @RW3+d8	NEGW 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	NEGW RW4', @RW4+d8	NEGW 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	NEGW RW5', @RW5+d8	NEGW 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	NEGW RW6', @RW6+d8	NEGW 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	NEGW RW7', @RW7+d8	NEGW 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	NEGW @RW0', @RW0+d16	NEGW @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	NEGW @RW1', @RW1+d16	NEGW @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	NEGW @RW2', @RW2+d16	NEGW @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	NEGW @RW3', @RW3+d16	NEGW @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	NEGW @RW0+', @RW0+RW7	NEGW @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	NEGW @RW1+', @RW1+RW7	NEGW @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	NEGW @RW2+', @PC+d16	NEGW @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	NEGW @RW3+', addr16	NEGW @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, A, RW0', @RW0+d8	MULW A, RW0', @RW0+d8	MULW A, A, RW0', @RW0+d8	MUL A, R0', @RW0+d8	MUL A, A, R0', @RW0+d8	MULW A, RW0', @RW0+d8	MULW A, A, RW0', @RW0+d8	DIVU A, R0', @RW0+d8	DIVU A, A, R0', @RW0+d8	DIVUW A, RW0', @RW0+d8	DIVUW A, A, RW0', @RW0+d8	DIV A, R0', @RW0+d8	DIV A, A, R0', @RW0+d8	DIVW A, RW0', @RW0+d8	DIVW A, A, RW0', @RW0+d8
+1	MULU A, R1', @RW1+d8	MULU A, A, RW1', @RW1+d8	MULW A, RW1', @RW1+d8	MULW A, A, RW1', @RW1+d8	MUL A, R1', @RW1+d8	MUL A, A, R1', @RW1+d8	MULW A, RW1', @RW1+d8	MULW A, A, RW1', @RW1+d8	DIVU A, R1', @RW1+d8	DIVU A, A, R1', @RW1+d8	DIVUW A, RW1', @RW1+d8	DIVUW A, A, RW1', @RW1+d8	DIV A, R1', @RW1+d8	DIV A, A, R1', @RW1+d8	DIVW A, RW1', @RW1+d8	DIVW A, A, RW1', @RW1+d8
+2	MULU A, R2', @RW2+d8	MULU A, A, RW2', @RW2+d8	MULW A, RW2', @RW2+d8	MULW A, A, RW2', @RW2+d8	MUL A, R2', @RW2+d8	MUL A, A, R2', @RW2+d8	MULW A, RW2', @RW2+d8	MULW A, A, RW2', @RW2+d8	DIVU A, R2', @RW2+d8	DIVU A, A, R2', @RW2+d8	DIVUW A, RW2', @RW2+d8	DIVUW A, A, RW2', @RW2+d8	DIV A, R2', @RW2+d8	DIV A, A, R2', @RW2+d8	DIVW A, RW2', @RW2+d8	DIVW A, A, RW2', @RW2+d8
+3	MULU A, R3', @RW3+d8	MULU A, A, RW3', @RW3+d8	MULW A, RW3', @RW3+d8	MULW A, A, RW3', @RW3+d8	MUL A, R3', @RW3+d8	MUL A, A, R3', @RW3+d8	MULW A, RW3', @RW3+d8	MULW A, A, RW3', @RW3+d8	DIVU A, R3', @RW3+d8	DIVU A, A, R3', @RW3+d8	DIVUW A, RW3', @RW3+d8	DIVUW A, A, RW3', @RW3+d8	DIV A, R3', @RW3+d8	DIV A, A, R3', @RW3+d8	DIVW A, RW3', @RW3+d8	DIVW A, A, RW3', @RW3+d8
+4	MULU A, R4', @RW4+d8	MULU A, A, RW4', @RW4+d8	MULW A, RW4', @RW4+d8	MULW A, A, RW4', @RW4+d8	MUL A, R4', @RW4+d8	MUL A, A, R4', @RW4+d8	MULW A, RW4', @RW4+d8	MULW A, A, RW4', @RW4+d8	DIVU A, R4', @RW4+d8	DIVU A, A, R4', @RW4+d8	DIVUW A, RW4', @RW4+d8	DIVUW A, A, RW4', @RW4+d8	DIV A, R4', @RW4+d8	DIV A, A, R4', @RW4+d8	DIVW A, RW4', @RW4+d8	DIVW A, A, RW4', @RW4+d8
+5	MULU A, R5', @RW5+d8	MULU A, A, RW5', @RW5+d8	MULW A, RW5', @RW5+d8	MULW A, A, RW5', @RW5+d8	MUL A, R5', @RW5+d8	MUL A, A, R5', @RW5+d8	MULW A, RW5', @RW5+d8	MULW A, A, RW5', @RW5+d8	DIVU A, R5', @RW5+d8	DIVU A, A, R5', @RW5+d8	DIVUW A, RW5', @RW5+d8	DIVUW A, A, RW5', @RW5+d8	DIV A, R5', @RW5+d8	DIV A, A, R5', @RW5+d8	DIVW A, RW5', @RW5+d8	DIVW A, A, RW5', @RW5+d8
+6	MULU A, R6', @RW6+d8	MULU A, A, RW6', @RW6+d8	MULW A, RW6', @RW6+d8	MULW A, A, RW6', @RW6+d8	MUL A, R6', @RW6+d8	MUL A, A, R6', @RW6+d8	MULW A, RW6', @RW6+d8	MULW A, A, RW6', @RW6+d8	DIVU A, R6', @RW6+d8	DIVU A, A, R6', @RW6+d8	DIVUW A, RW6', @RW6+d8	DIVUW A, A, RW6', @RW6+d8	DIV A, R6', @RW6+d8	DIV A, A, R6', @RW6+d8	DIVW A, RW6', @RW6+d8	DIVW A, A, RW6', @RW6+d8
+7	MULU A, R7', @RW7+d8	MULU A, A, RW7', @RW7+d8	MULW A, RW7', @RW7+d8	MULW A, A, RW7', @RW7+d8	MUL A, R7', @RW7+d8	MUL A, A, R7', @RW7+d8	MULW A, RW7', @RW7+d8	MULW A, A, RW7', @RW7+d8	DIVU A, R7', @RW7+d8	DIVU A, A, R7', @RW7+d8	DIVUW A, RW7', @RW7+d8	DIVUW A, A, RW7', @RW7+d8	DIV A, R7', @RW7+d8	DIV A, A, R7', @RW7+d8	DIVW A, RW7', @RW7+d8	DIVW A, A, RW7', @RW7+d8
+8	MULU A, @RW0', @RW0+d16	MULU A, A, @RW0', @RW0+d16	MULW A, @RW0', @RW0+d16	MULW A, A, @RW0', @RW0+d16	MUL A, @RW0', @RW0+d16	MUL A, A, @RW0', @RW0+d16	MULW A, @RW0', @RW0+d16	MULW A, A, @RW0', @RW0+d16	DIVU A, @RW0', @RW0+d16	DIVU A, A, @RW0', @RW0+d16	DIVUW A, @RW0', @RW0+d16	DIVUW A, A, @RW0', @RW0+d16	DIV A, @RW0', @RW0+d16	DIV A, A, @RW0', @RW0+d16	DIVW A, @RW0', @RW0+d16	DIVW A, A, @RW0', @RW0+d16
+9	MULU A, @RW1', @RW1+d16	MULU A, A, @RW1', @RW1+d16	MULW A, @RW1', @RW1+d16	MULW A, A, @RW1', @RW1+d16	MUL A, @RW1', @RW1+d16	MUL A, A, @RW1', @RW1+d16	MULW A, @RW1', @RW1+d16	MULW A, A, @RW1', @RW1+d16	DIVU A, @RW1', @RW1+d16	DIVU A, A, @RW1', @RW1+d16	DIVUW A, @RW1', @RW1+d16	DIVUW A, A, @RW1', @RW1+d16	DIV A, @RW1', @RW1+d16	DIV A, A, @RW1', @RW1+d16	DIVW A, @RW1', @RW1+d16	DIVW A, A, @RW1', @RW1+d16
+A	MULU A, @RW2', @RW2+d16	MULU A, A, @RW2', @RW2+d16	MULW A, @RW2', @RW2+d16	MULW A, A, @RW2', @RW2+d16	MUL A, @RW2', @RW2+d16	MUL A, A, @RW2', @RW2+d16	MULW A, @RW2', @RW2+d16	MULW A, A, @RW2', @RW2+d16	DIVU A, @RW2', @RW2+d16	DIVU A, A, @RW2', @RW2+d16	DIVUW A, @RW2', @RW2+d16	DIVUW A, A, @RW2', @RW2+d16	DIV A, @RW2', @RW2+d16	DIV A, A, @RW2', @RW2+d16	DIVW A, @RW2', @RW2+d16	DIVW A, A, @RW2', @RW2+d16
+B	MULU A, @RW3', @RW3+d16	MULU A, A, @RW3', @RW3+d16	MULW A, @RW3', @RW3+d16	MULW A, A, @RW3', @RW3+d16	MUL A, @RW3', @RW3+d16	MUL A, A, @RW3', @RW3+d16	MULW A, @RW3', @RW3+d16	MULW A, A, @RW3', @RW3+d16	DIVU A, @RW3', @RW3+d16	DIVU A, A, @RW3', @RW3+d16	DIVUW A, @RW3', @RW3+d16	DIVUW A, A, @RW3', @RW3+d16	DIV A, @RW3', @RW3+d16	DIV A, A, @RW3', @RW3+d16	DIVW A, @RW3', @RW3+d16	DIVW A, A, @RW3', @RW3+d16
+C	MULU A, @RW0+', @RW0+RW7	MULU A, A, @RW0+', @RW0+RW7	MULW A, @RW0+', @RW0+RW7	MULW A, A, @RW0+', @RW0+RW7	MUL A, @RW0+', @RW0+RW7	MUL A, A, @RW0+', @RW0+RW7	MULW A, @RW0+', @RW0+RW7	MULW A, A, @RW0+', @RW0+RW7	DIVU A, @RW0+', @RW0+RW7	DIVU A, A, @RW0+', @RW0+RW7	DIVUW A, @RW0+', @RW0+RW7	DIVUW A, A, @RW0+', @RW0+RW7	DIV A, @RW0+', @RW0+RW7	DIV A, A, @RW0+', @RW0+RW7	DIVW A, @RW0+', @RW0+RW7	DIVW A, A, @RW0+', @RW0+RW7
+D	MULU A, @RW1+', @RW1+RW7	MULU A, A, @RW1+', @RW1+RW7	MULW A, @RW1+', @RW1+RW7	MULW A, A, @RW1+', @RW1+RW7	MUL A, @RW1+', @RW1+RW7	MUL A, A, @RW1+', @RW1+RW7	MULW A, @RW1+', @RW1+RW7	MULW A, A, @RW1+', @RW1+RW7	DIVU A, @RW1+', @RW1+RW7	DIVU A, A, @RW1+', @RW1+RW7	DIVUW A, @RW1+', @RW1+RW7	DIVUW A, A, @RW1+', @RW1+RW7	DIV A, @RW1+', @RW1+RW7	DIV A, A, @RW1+', @RW1+RW7	DIVW A, @RW1+', @RW1+RW7	DIVW A, A, @RW1+', @RW1+RW7
+E	MULU A, @RW2+', @PC+d16	MULU A, A, @RW2+', @PC+d16	MULW A, @RW2+', @PC+d16	MULW A, A, @RW2+', @PC+d16	MUL A, @RW2+', @PC+d16	MUL A, A, @RW2+', @PC+d16	MULW A, @RW2+', @PC+d16	MULW A, A, @RW2+', @PC+d16	DIVU A, @RW2+', @PC+d16	DIVU A, A, @RW2+', @PC+d16	DIVUW A, @RW2+', @PC+d16	DIVUW A, A, @RW2+', @PC+d16	DIV A, @RW2+', @PC+d16	DIV A, A, @RW2+', @PC+d16	DIVW A, @RW2+', @PC+d16	DIVW A, A, @RW2+', @PC+d16
+F	MULU A, @RW3+', addr16	MULU A, A, @RW3+', addr16	MULW A, @RW3+', addr16	MULW A, A, @RW3+', addr16	MUL A, @RW3+', addr16	MUL A, A, @RW3+', addr16	MULW A, @RW3+', addr16	MULW A, A, @RW3+', addr16	DIVU A, @RW3+', addr16	DIVU A, A, @RW3+', addr16	DIVUW A, @RW3+', addr16	DIVUW A, A, @RW3+', addr16	DIV A, @RW3+', addr16	DIV A, A, @RW3+', addr16	DIVW A, @RW3+', addr16	DIVW A, A, @RW3+', addr16

表 B.9-15 MOVEA RWi, ea 命令 (第1バイト = 79_H)[illegible]

表 B.9-16 MOV Ri, ea 命令 (第1バイト = 7A_H)

	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
+0	MOV R0, R0 @ RW0+d8	MOV R0, R0 @ RW0+d8	MOV R1, R1 @ RW0+d8	MOV R1, R0 @ RW0+d8	MOV R2, R0 @ RW0+d8	MOV R2, R0 @ RW0+d8	MOV R3, R0 @ RW0+d8	MOV R3, R0 @ RW0+d8	MOV R4, R0 @ RW0+d8	MOV R4, R0 @ RW0+d8	MOV R5, R0 @ RW0+d8	MOV R5, R0 @ RW0+d8	MOV R6, R0 @ RW0+d8	MOV R6, R0 @ RW0+d8	MOV R7, R0 @ RW0+d8	MOV R7, R0 @ RW0+d8
+1	MOV R0, R1 @ RW1+d8	MOV R0, R1 @ RW1+d8	MOV R1, R1 @ RW1+d8	MOV R2, R1 @ RW1+d8	MOV R2, R1 @ RW1+d8	MOV R2, R1 @ RW1+d8	MOV R3, R1 @ RW1+d8	MOV R3, R1 @ RW1+d8	MOV R4, R1 @ RW1+d8	MOV R4, R1 @ RW1+d8	MOV R5, R1 @ RW1+d8	MOV R5, R1 @ RW1+d8	MOV R6, R1 @ RW1+d8	MOV R6, R1 @ RW1+d8	MOV R7, R1 @ RW1+d8	MOV R7, R1 @ RW1+d8
+2	MOV R0, R2 @ RW2+d8	MOV R0, R2 @ RW2+d8	MOV R1, R2 @ RW2+d8	MOV R2, R2 @ RW2+d8	MOV R2, R2 @ RW2+d8	MOV R2, R2 @ RW2+d8	MOV R3, R2 @ RW2+d8	MOV R3, R2 @ RW2+d8	MOV R4, R2 @ RW2+d8	MOV R4, R2 @ RW2+d8	MOV R5, R2 @ RW2+d8	MOV R5, R2 @ RW2+d8	MOV R6, R2 @ RW2+d8	MOV R6, R2 @ RW2+d8	MOV R7, R2 @ RW2+d8	MOV R7, R2 @ RW2+d8
+3	MOV R0, R3 @ RW3+d8	MOV R0, R3 @ RW3+d8	MOV R1, R3 @ RW3+d8	MOV R2, R3 @ RW3+d8	MOV R2, R3 @ RW3+d8	MOV R2, R3 @ RW3+d8	MOV R3, R3 @ RW3+d8	MOV R3, R3 @ RW3+d8	MOV R4, R3 @ RW3+d8	MOV R4, R3 @ RW3+d8	MOV R5, R3 @ RW3+d8	MOV R5, R3 @ RW3+d8	MOV R6, R3 @ RW3+d8	MOV R6, R3 @ RW3+d8	MOV R7, R3 @ RW3+d8	MOV R7, R3 @ RW3+d8
+4	MOV R0, R4 @ RW4+d8	MOV R0, R4 @ RW4+d8	MOV R1, R4 @ RW4+d8	MOV R2, R4 @ RW4+d8	MOV R2, R4 @ RW4+d8	MOV R2, R4 @ RW4+d8	MOV R3, R4 @ RW4+d8	MOV R3, R4 @ RW4+d8	MOV R4, R4 @ RW4+d8	MOV R4, R4 @ RW4+d8	MOV R5, R4 @ RW4+d8	MOV R5, R4 @ RW4+d8	MOV R6, R4 @ RW4+d8	MOV R6, R4 @ RW4+d8	MOV R7, R4 @ RW4+d8	MOV R7, R4 @ RW4+d8
+5	MOV R0, R5 @ RW5+d8	MOV R0, R5 @ RW5+d8	MOV R1, R5 @ RW5+d8	MOV R2, R5 @ RW5+d8	MOV R2, R5 @ RW5+d8	MOV R2, R5 @ RW5+d8	MOV R3, R5 @ RW5+d8	MOV R3, R5 @ RW5+d8	MOV R4, R5 @ RW5+d8	MOV R4, R5 @ RW5+d8	MOV R5, R5 @ RW5+d8	MOV R5, R5 @ RW5+d8	MOV R6, R5 @ RW5+d8	MOV R6, R5 @ RW5+d8	MOV R7, R5 @ RW5+d8	MOV R7, R5 @ RW5+d8
+6	MOV R0, R6 @ RW6+d8	MOV R0, R6 @ RW6+d8	MOV R1, R6 @ RW6+d8	MOV R2, R6 @ RW6+d8	MOV R2, R6 @ RW6+d8	MOV R2, R6 @ RW6+d8	MOV R3, R6 @ RW6+d8	MOV R3, R6 @ RW6+d8	MOV R4, R6 @ RW6+d8	MOV R4, R6 @ RW6+d8	MOV R5, R6 @ RW6+d8	MOV R5, R6 @ RW6+d8	MOV R6, R6 @ RW6+d8	MOV R6, R6 @ RW6+d8	MOV R7, R6 @ RW6+d8	MOV R7, R6 @ RW6+d8
+7	MOV R0, R7 @ RW7+d8	MOV R0, R7 @ RW7+d8	MOV R1, R7 @ RW7+d8	MOV R2, R7 @ RW7+d8	MOV R2, R7 @ RW7+d8	MOV R2, R7 @ RW7+d8	MOV R3, R7 @ RW7+d8	MOV R3, R7 @ RW7+d8	MOV R4, R7 @ RW7+d8	MOV R4, R7 @ RW7+d8	MOV R5, R7 @ RW7+d8	MOV R5, R7 @ RW7+d8	MOV R6, R7 @ RW7+d8	MOV R6, R7 @ RW7+d8	MOV R7, R7 @ RW7+d8	MOV R7, R7 @ RW7+d8
+8	MOV R0, @ RW0	MOV R0, @ RW0+d16	MOV R1, @ RW0	MOV R2, @ RW0+d16	MOV R2, @ RW0	MOV R2, @ RW0+d16	MOV R3, @ RW0	MOV R3, @ RW0+d16	MOV R4, @ RW0	MOV R4, @ RW0+d16	MOV R5, @ RW0	MOV R5, @ RW0+d16	MOV R6, @ RW0	MOV R6, @ RW0+d16	MOV R7, @ RW0	MOV R7, @ RW0+d16
+9	MOV R0, @ RW1	MOV R0, @ RW1+d16	MOV R1, @ RW1	MOV R2, @ RW1+d16	MOV R2, @ RW1	MOV R2, @ RW1+d16	MOV R3, @ RW1	MOV R3, @ RW1+d16	MOV R4, @ RW1	MOV R4, @ RW1+d16	MOV R5, @ RW1	MOV R5, @ RW1+d16	MOV R6, @ RW1	MOV R6, @ RW1+d16	MOV R7, @ RW1	MOV R7, @ RW1+d16
+A	MOV R0, @ RW2	MOV R0, @ RW2+d16	MOV R1, @ RW2	MOV R2, @ RW2+d16	MOV R2, @ RW2	MOV R2, @ RW2+d16	MOV R3, @ RW2	MOV R3, @ RW2+d16	MOV R4, @ RW2	MOV R4, @ RW2+d16	MOV R5, @ RW2	MOV R5, @ RW2+d16	MOV R6, @ RW2	MOV R6, @ RW2+d16	MOV R7, @ RW2	MOV R7, @ RW2+d16
+B	MOV R0, @ RW3	MOV R0, @ RW3+d16	MOV R1, @ RW3	MOV R2, @ RW3+d16	MOV R2, @ RW3	MOV R2, @ RW3+d16	MOV R3, @ RW3	MOV R3, @ RW3+d16	MOV R4, @ RW3	MOV R4, @ RW3+d16	MOV R5, @ RW3	MOV R5, @ RW3+d16	MOV R6, @ RW3	MOV R6, @ RW3+d16	MOV R7, @ RW3	MOV R7, @ RW3+d16
+C	MOV R0, @ RW0+	MOV R0, @ RW0-RW7	MOV R1, @ RW0	MOV R2, @ RW0+	MOV R2, @ RW0+	MOV R2, @ RW0-RW7	MOV R3, @ RW0+	MOV R3, @ RW0-RW7	MOV R4, @ RW0+	MOV R4, @ RW0-RW7	MOV R5, @ RW0+	MOV R5, @ RW0-RW7	MOV R6, @ RW0+	MOV R6, @ RW0-RW7	MOV R7, @ RW0+	MOV R7, @ RW0-RW7
+D	MOV R0, @ RW1+	MOV R0, @ RW1+RW7	MOV R1, @ RW1	MOV R2, @ RW1+	MOV R2, @ RW1+	MOV R2, @ RW1+RW7	MOV R3, @ RW1+	MOV R3, @ RW1+RW7	MOV R4, @ RW1+	MOV R4, @ RW1+RW7	MOV R5, @ RW1+	MOV R5, @ RW1+RW7	MOV R6, @ RW1+	MOV R6, @ RW1+RW7	MOV R7, @ RW1+	MOV R7, @ RW1+RW7
+E	MOV R0, @ RW2+	MOV R0, @ PC+d16	MOV R1, @ RW2	MOV R2, @ RW2+	MOV R2, @ RW2+	MOV R2, @ PC+d16	MOV R3, @ RW2+	MOV R3, @ PC+d16	MOV R4, @ RW2+	MOV R4, @ PC+d16	MOV R5, @ RW2+	MOV R5, @ PC+d16	MOV R6, @ RW2+	MOV R6, @ PC+d16	MOV R7, @ RW2+	MOV R7, @ PC+d16
+F	MOV R0, @ RW3+	MOV R0, addr16	MOV R1, @ RW3	MOV R2, @ RW3+	MOV R2, @ RW3+	MOV R2, addr16	MOV R3, @ RW3+	MOV R3, addr16	MOV R4, @ RW3+	MOV R4, addr16	MOV R5, @ RW3	MOV R5, addr16	MOV R6, @ RW3	MOV R6, addr16	MOV R7, @ RW3	MOV R7, addr16

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, R1, @RW0-d8, R1	MOV R0, R2, @RW0+d8, R2	MOV R0, R3, @RW0+d8, R3	MOV R0, R4, @RW0+d8, R4	MOV R0, R5, @RW0+d8, R5	MOV R0, R6, @RW0+d8, R6	MOV R0, R7, @RW0+d8, R7	MOV R0, R8, @RW0+d8, R8	MOV R0, R9, @RW0+d8, R9	MOV R0, R10, @RW0+d8, R10	MOV R0, R11, @RW0+d8, R11	MOV R0, R12, @RW0+d8, R12	MOV R0, R13, @RW0+d8, R13	MOV R0, R14, @RW0+d8, R14
+1	MOV R1, R0, @RW1+d8, R0	MOV R1, R1, @RW1+d8, R1	MOV R1, R2, @RW1+d8, R2	MOV R1, R3, @RW1+d8, R3	MOV R1, R4, @RW1+d8, R4	MOV R1, R5, @RW1+d8, R5	MOV R1, R6, @RW1+d8, R6	MOV R1, R7, @RW1+d8, R7	MOV R1, R8, @RW1+d8, R8	MOV R1, R9, @RW1+d8, R9	MOV R1, R10, @RW1+d8, R10	MOV R1, R11, @RW1+d8, R11	MOV R1, R12, @RW1+d8, R12	MOV R1, R13, @RW1+d8, R13	MOV R1, R14, @RW1+d8, R14	MOV R1, R15, @RW1+d8, R15
+2	MOV R2, R0, @RW2+d8, R0	MOV R2, R1, @RW2+d8, R1	MOV R2, R2, @RW2+d8, R2	MOV R2, R3, @RW2+d8, R3	MOV R2, R4, @RW2+d8, R4	MOV R2, R5, @RW2+d8, R5	MOV R2, R6, @RW2+d8, R6	MOV R2, R7, @RW2+d8, R7	MOV R2, R8, @RW2+d8, R8	MOV R2, R9, @RW2+d8, R9	MOV R2, R10, @RW2+d8, R10	MOV R2, R11, @RW2+d8, R11	MOV R2, R12, @RW2+d8, R12	MOV R2, R13, @RW2+d8, R13	MOV R2, R14, @RW2+d8, R14	MOV R2, R15, @RW2+d8, R15
+3	MOV R3, R0, @RW3+d8, R0	MOV R3, R1, @RW3+d8, R1	MOV R3, R2, @RW3+d8, R2	MOV R3, R3, @RW3+d8, R3	MOV R3, R4, @RW3+d8, R4	MOV R3, R5, @RW3+d8, R5	MOV R3, R6, @RW3+d8, R6	MOV R3, R7, @RW3+d8, R7	MOV R3, R8, @RW3+d8, R8	MOV R3, R9, @RW3+d8, R9	MOV R3, R10, @RW3+d8, R10	MOV R3, R11, @RW3+d8, R11	MOV R3, R12, @RW3+d8, R12	MOV R3, R13, @RW3+d8, R13	MOV R3, R14, @RW3+d8, R14	MOV R3, R15, @RW3+d8, R15
+4	MOV R4, R0, @RW4+d8, R0	MOV R4, R1, @RW4+d8, R1	MOV R4, R2, @RW4+d8, R2	MOV R4, R3, @RW4+d8, R3	MOV R4, R4, @RW4+d8, R4	MOV R4, R5, @RW4+d8, R5	MOV R4, R6, @RW4+d8, R6	MOV R4, R7, @RW4+d8, R7	MOV R4, R8, @RW4+d8, R8	MOV R4, R9, @RW4+d8, R9	MOV R4, R10, @RW4+d8, R10	MOV R4, R11, @RW4+d8, R11	MOV R4, R12, @RW4+d8, R12	MOV R4, R13, @RW4+d8, R13	MOV R4, R14, @RW4+d8, R14	MOV R4, R15, @RW4+d8, R15
+5	MOV R5, R0, @RW5+d8, R0	MOV R5, R1, @RW5+d8, R1	MOV R5, R2, @RW5+d8, R2	MOV R5, R3, @RW5+d8, R3	MOV R5, R4, @RW5+d8, R4	MOV R5, R5, @RW5+d8, R5	MOV R5, R6, @RW5+d8, R6	MOV R5, R7, @RW5+d8, R7	MOV R5, R8, @RW5+d8, R8	MOV R5, R9, @RW5+d8, R9	MOV R5, R10, @RW5+d8, R10	MOV R5, R11, @RW5+d8, R11	MOV R5, R12, @RW5+d8, R12	MOV R5, R13, @RW5+d8, R13	MOV R5, R14, @RW5+d8, R14	MOV R5, R15, @RW5+d8, R15
+6	MOV R6, R0, @RW6+d8, R0	MOV R6, R1, @RW6+d8, R1	MOV R6, R2, @RW6+d8, R2	MOV R6, R3, @RW6+d8, R3	MOV R6, R4, @RW6+d8, R4	MOV R6, R5, @RW6+d8, R5	MOV R6, R6, @RW6+d8, R6	MOV R6, R7, @RW6+d8, R7	MOV R6, R8, @RW6+d8, R8	MOV R6, R9, @RW6+d8, R9	MOV R6, R10, @RW6+d8, R10	MOV R6, R11, @RW6+d8, R11	MOV R6, R12, @RW6+d8, R12	MOV R6, R13, @RW6+d8, R13	MOV R6, R14, @RW6+d8, R14	MOV R6, R15, @RW6+d8, R15
+7	MOV R7, R0, @RW7+d8, R0	MOV R7, R1, @RW7+d8, R1	MOV R7, R2, @RW7+d8, R2	MOV R7, R3, @RW7+d8, R3	MOV R7, R4, @RW7+d8, R4	MOV R7, R5, @RW7+d8, R5	MOV R7, R6, @RW7+d8, R6	MOV R7, R7, @RW7+d8, R7	MOV R7, R8, @RW7+d8, R8	MOV R7, R9, @RW7+d8, R9	MOV R7, R10, @RW7+d8, R10	MOV R7, R11, @RW7+d8, R11	MOV R7, R12, @RW7+d8, R12	MOV R7, R13, @RW7+d8, R13	MOV R7, R14, @RW7+d8, R14	MOV R7, R15, @RW7+d8, R15
+8	MOV @RW0, R0, @RW0+d16, R0	MOV @RW0, R1, @RW0+d16, R1	MOV @RW0, R2, @RW0+d16, R2	MOV @RW0, R3, @RW0+d16, R3	MOV @RW0, R4, @RW0+d16, R4	MOV @RW0, R5, @RW0+d16, R5	MOV @RW0, R6, @RW0+d16, R6	MOV @RW0, R7, @RW0+d16, R7	MOV @RW0, R8, @RW0+d16, R8	MOV @RW0, R9, @RW0+d16, R9	MOV @RW0, R10, @RW0+d16, R10	MOV @RW0, R11, @RW0+d16, R11	MOV @RW0, R12, @RW0+d16, R12	MOV @RW0, R13, @RW0+d16, R13	MOV @RW0, R14, @RW0+d16, R14	MOV @RW0, R15, @RW0+d16, R15
+9	MOV @RW1, R0, @RW1+d16, R0	MOV @RW1, R1, @RW1+d16, R1	MOV @RW1, R2, @RW1+d16, R2	MOV @RW1, R3, @RW1+d16, R3	MOV @RW1, R4, @RW1+d16, R4	MOV @RW1, R5, @RW1+d16, R5	MOV @RW1, R6, @RW1+d16, R6	MOV @RW1, R7, @RW1+d16, R7	MOV @RW1, R8, @RW1+d16, R8	MOV @RW1, R9, @RW1+d16, R9	MOV @RW1, R10, @RW1+d16, R10	MOV @RW1, R11, @RW1+d16, R11	MOV @RW1, R12, @RW1+d16, R12	MOV @RW1, R13, @RW1+d16, R13	MOV @RW1, R14, @RW1+d16, R14	MOV @RW1, R15, @RW1+d16, R15
+A	MOV @RW2, R0, @RW2+d16, R0	MOV @RW2, R1, @RW2+d16, R1	MOV @RW2, R2, @RW2+d16, R2	MOV @RW2, R3, @RW2+d16, R3	MOV @RW2, R4, @RW2+d16, R4	MOV @RW2, R5, @RW2+d16, R5	MOV @RW2, R6, @RW2+d16, R6	MOV @RW2, R7, @RW2+d16, R7	MOV @RW2, R8, @RW2+d16, R8	MOV @RW2, R9, @RW2+d16, R9	MOV @RW2, R10, @RW2+d16, R10	MOV @RW2, R11, @RW2+d16, R11	MOV @RW2, R12, @RW2+d16, R12	MOV @RW2, R13, @RW2+d16, R13	MOV @RW2, R14, @RW2+d16, R14	MOV @RW2, R15, @RW2+d16, R15
+B	MOV @RW3, R0, @RW3+d16, R0	MOV @RW3, R1, @RW3+d16, R1	MOV @RW3, R2, @RW3+d16, R2	MOV @RW3, R3, @RW3+d16, R3	MOV @RW3, R4, @RW3+d16, R4	MOV @RW3, R5, @RW3+d16, R5	MOV @RW3, R6, @RW3+d16, R6	MOV @RW3, R7, @RW3+d16, R7	MOV @RW3, R8, @RW3+d16, R8	MOV @RW3, R9, @RW3+d16, R9	MOV @RW3, R10, @RW3+d16, R10	MOV @RW3, R11, @RW3+d16, R11	MOV @RW3, R12, @RW3+d16, R12	MOV @RW3, R13, @RW3+d16, R13	MOV @RW3, R14, @RW3+d16, R14	MOV @RW3, R15, @RW3+d16, R15
+C	MOV @RW0+, R0, @RW0+RW7, R0	MOV @RW0+, R1, @RW0+RW7, R1	MOV @RW0+, R2, @RW0+RW7, R2	MOV @RW0+, R3, @RW0+RW7, R3	MOV @RW0+, R4, @RW0+RW7, R4	MOV @RW0+, R5, @RW0+RW7, R5	MOV @RW0+, R6, @RW0+RW7, R6	MOV @RW0+, R7, @RW0+RW7, R7	MOV @RW0+, R8, @RW0+RW7, R8	MOV @RW0+, R9, @RW0+RW7, R9	MOV @RW0+, R10, @RW0+RW7, R10	MOV @RW0+, R11, @RW0+RW7, R11	MOV @RW0+, R12, @RW0+RW7, R12	MOV @RW0+, R13, @RW0+RW7, R13	MOV @RW0+, R14, @RW0+RW7, R14	MOV @RW0+, R15, @RW0+RW7, R15
+D	MOV @RW1+, R0, @RW1+RW7, R0	MOV @RW1+, R1, @RW1+RW7, R1	MOV @RW1+, R2, @RW1+RW7, R2	MOV @RW1+, R3, @RW1+RW7, R3	MOV @RW1+, R4, @RW1+RW7, R4	MOV @RW1+, R5, @RW1+RW7, R5	MOV @RW1+, R6, @RW1+RW7, R6	MOV @RW1+, R7, @RW1+RW7, R7	MOV @RW1+, R8, @RW1+RW7, R8	MOV @RW1+, R9, @RW1+RW7, R9	MOV @RW1+, R10, @RW1+RW7, R10	MOV @RW1+, R11, @RW1+RW7, R11	MOV @RW1+, R12, @RW1+RW7, R12	MOV @RW1+, R13, @RW1+RW7, R13	MOV @RW1+, R14, @RW1+RW7, R14	MOV @RW1+, R15, @RW1+RW7, R15
+E	MOV @RW2+, R0, @PC+d16, R0	MOV @RW2+, R1, @PC+d16, R1	MOV @RW2+, R2, @PC+d16, R2	MOV @RW2+, R3, @PC+d16, R3	MOV @RW2+, R4, @PC+d16, R4	MOV @RW2+, R5, @PC+d16, R5	MOV @RW2+, R6, @PC+d16, R6	MOV @RW2+, R7, @PC+d16, R7	MOV @RW2+, R8, @PC+d16, R8	MOV @RW2+, R9, @PC+d16, R9	MOV @RW2+, R10, @PC+d16, R10	MOV @RW2+, R11, @PC+d16, R11	MOV @RW2+, R12, @PC+d16, R12	MOV @RW2+, R13, @PC+d16, R13	MOV @RW2+, R14, @PC+d16, R14	MOV @RW2+, R15, @PC+d16, R15
+F	MOV @RW3+, R0, addr16, R0	MOV @RW3+, R1, addr16, R1	MOV @RW3+, R2, addr16, R2	MOV @RW3+, R3, addr16, R3	MOV @RW3+, R4, addr16, R4	MOV @RW3+, R5, addr16, R5	MOV @RW3+, R6, addr16, R6	MOV @RW3+, R7, addr16, R7	MOV @RW3+, R8, addr16, R8	MOV @RW3+, R9, addr16, R9	MOV @RW3+, R10, addr16, R10	MOV @RW3+, R11, addr16, R11	MOV @RW3+, R12, addr16, R12	MOV @RW3+, R13, addr16, R13	MOV @RW3+, R14, addr16, R14	MOV @RW3+, R15, addr16, R15

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+0, RW0	MOVW RW0, RW0, @RW0+0, RW0	MOVW RW0, RW1, @RW0+0, RW1	MOVW RW0, RW1, @RW0+0, RW1	MOVW RW0, RW2, @RW0+0, RW2	MOVW RW0, RW2, @RW0+0, RW2	MOVW RW0, RW3, @RW0+0, RW3	MOVW RW0, RW3, @RW0+0, RW3	MOVW RW0, RW4, @RW0+0, RW4	MOVW RW0, RW4, @RW0+0, RW4	MOVW RW0, RW5, @RW0+0, RW5	MOVW RW0, RW5, @RW0+0, RW5	MOVW RW0, RW6, @RW0+0, RW6	MOVW RW0, RW6, @RW0+0, RW6	MOVW RW0, RW7, @RW0+0, RW7	MOVW RW0, RW7, @RW0+0, RW7
+1	MOVW RW1, RW0, @RW1+0, RW0	MOVW RW1, RW0, @RW1+0, RW0	MOVW RW1, RW1, @RW1+0, RW1	MOVW RW1, RW1, @RW1+0, RW1	MOVW RW1, RW2, @RW1+0, RW2	MOVW RW1, RW2, @RW1+0, RW2	MOVW RW1, RW3, @RW1+0, RW3	MOVW RW1, RW3, @RW1+0, RW3	MOVW RW1, RW4, @RW1+0, RW4	MOVW RW1, RW4, @RW1+0, RW4	MOVW RW1, RW5, @RW1+0, RW5	MOVW RW1, RW5, @RW1+0, RW5	MOVW RW1, RW6, @RW1+0, RW6	MOVW RW1, RW6, @RW1+0, RW6	MOVW RW1, RW7, @RW1+0, RW7	MOVW RW1, RW7, @RW1+0, RW7
+2	MOVW RW2, RW0, @RW2+0, RW0	MOVW RW2, RW0, @RW2+0, RW0	MOVW RW2, RW1, @RW2+0, RW1	MOVW RW2, RW1, @RW2+0, RW1	MOVW RW2, RW2, @RW2+0, RW2	MOVW RW2, RW2, @RW2+0, RW2	MOVW RW2, RW3, @RW2+0, RW3	MOVW RW2, RW3, @RW2+0, RW3	MOVW RW2, RW4, @RW2+0, RW4	MOVW RW2, RW4, @RW2+0, RW4	MOVW RW2, RW5, @RW2+0, RW5	MOVW RW2, RW5, @RW2+0, RW5	MOVW RW2, RW6, @RW2+0, RW6	MOVW RW2, RW6, @RW2+0, RW6	MOVW RW2, RW7, @RW2+0, RW7	MOVW RW2, RW7, @RW2+0, RW7
+3	MOVW RW3, RW0, @RW3+0, RW0	MOVW RW3, RW0, @RW3+0, RW0	MOVW RW3, RW1, @RW3+0, RW1	MOVW RW3, RW1, @RW3+0, RW1	MOVW RW3, RW2, @RW3+0, RW2	MOVW RW3, RW2, @RW3+0, RW2	MOVW RW3, RW3, @RW3+0, RW3	MOVW RW3, RW3, @RW3+0, RW3	MOVW RW3, RW4, @RW3+0, RW4	MOVW RW3, RW4, @RW3+0, RW4	MOVW RW3, RW5, @RW3+0, RW5	MOVW RW3, RW5, @RW3+0, RW5	MOVW RW3, RW6, @RW3+0, RW6	MOVW RW3, RW6, @RW3+0, RW6	MOVW RW3, RW7, @RW3+0, RW7	MOVW RW3, RW7, @RW3+0, RW7
+4	MOVW RW4, RW0, @RW4+0, RW0	MOVW RW4, RW0, @RW4+0, RW0	MOVW RW4, RW1, @RW4+0, RW1	MOVW RW4, RW1, @RW4+0, RW1	MOVW RW4, RW2, @RW4+0, RW2	MOVW RW4, RW2, @RW4+0, RW2	MOVW RW4, RW3, @RW4+0, RW3	MOVW RW4, RW3, @RW4+0, RW3	MOVW RW4, RW4, @RW4+0, RW4	MOVW RW4, RW4, @RW4+0, RW4	MOVW RW4, RW5, @RW4+0, RW5	MOVW RW4, RW5, @RW4+0, RW5	MOVW RW4, RW6, @RW4+0, RW6	MOVW RW4, RW6, @RW4+0, RW6	MOVW RW4, RW7, @RW4+0, RW7	MOVW RW4, RW7, @RW4+0, RW7
+5	MOVW RW5, RW0, @RW5+0, RW0	MOVW RW5, RW0, @RW5+0, RW0	MOVW RW5, RW1, @RW5+0, RW1	MOVW RW5, RW1, @RW5+0, RW1	MOVW RW5, RW2, @RW5+0, RW2	MOVW RW5, RW2, @RW5+0, RW2	MOVW RW5, RW3, @RW5+0, RW3	MOVW RW5, RW3, @RW5+0, RW3	MOVW RW5, RW4, @RW5+0, RW4	MOVW RW5, RW4, @RW5+0, RW4	MOVW RW5, RW5, @RW5+0, RW5	MOVW RW5, RW5, @RW5+0, RW5	MOVW RW5, RW6, @RW5+0, RW6	MOVW RW5, RW6, @RW5+0, RW6	MOVW RW5, RW7, @RW5+0, RW7	MOVW RW5, RW7, @RW5+0, RW7
+6	MOVW RW6, RW0, @RW6+0, RW0	MOVW RW6, RW0, @RW6+0, RW0	MOVW RW6, RW1, @RW6+0, RW1	MOVW RW6, RW1, @RW6+0, RW1	MOVW RW6, RW2, @RW6+0, RW2	MOVW RW6, RW2, @RW6+0, RW2	MOVW RW6, RW3, @RW6+0, RW3	MOVW RW6, RW3, @RW6+0, RW3	MOVW RW6, RW4, @RW6+0, RW4	MOVW RW6, RW4, @RW6+0, RW4	MOVW RW6, RW5, @RW6+0, RW5	MOVW RW6, RW5, @RW6+0, RW5	MOVW RW6, RW6, @RW6+0, RW6	MOVW RW6, RW6, @RW6+0, RW6	MOVW RW6, RW7, @RW6+0, RW7	MOVW RW6, RW7, @RW6+0, RW7
+7	MOVW RW7, RW0, @RW7+0, RW0	MOVW RW7, RW0, @RW7+0, RW0	MOVW RW7, RW1, @RW7+0, RW1	MOVW RW7, RW1, @RW7+0, RW1	MOVW RW7, RW2, @RW7+0, RW2	MOVW RW7, RW2, @RW7+0, RW2	MOVW RW7, RW3, @RW7+0, RW3	MOVW RW7, RW3, @RW7+0, RW3	MOVW RW7, RW4, @RW7+0, RW4	MOVW RW7, RW4, @RW7+0, RW4	MOVW RW7, RW5, @RW7+0, RW5	MOVW RW7, RW5, @RW7+0, RW5	MOVW RW7, RW6, @RW7+0, RW6	MOVW RW7, RW6, @RW7+0, RW6	MOVW RW7, RW7, @RW7+0, RW7	MOVW RW7, RW7, @RW7+0, RW7
+8	MOVW @RW0, RW0, @RW0+0, RW0	MOVW @RW0, RW0, @RW0+0, RW0	MOVW @RW0, RW1, +d16, RW1	MOVW @RW0, RW0, @RW0+0, RW0	MOVW @RW0, RW2, +d16, RW2	MOVW @RW0, RW2, +d16, RW2	MOVW @RW0, RW3, +d16, RW3	MOVW @RW0, RW3, +d16, RW3	MOVW @RW0, RW4, +d16, RW4	MOVW @RW0, RW4, +d16, RW4	MOVW @RW0, RW5, +d16, RW5	MOVW @RW0, RW5, +d16, RW5	MOVW @RW0, RW6, +d16, RW6	MOVW @RW0, RW6, +d16, RW6	MOVW @RW0, RW7, +d16, RW7	MOVW @RW0, RW7, +d16, RW7
+9	MOVW @RW1, RW0, @RW1+0, RW0	MOVW @RW1, RW0, @RW1+0, RW0	MOVW @RW1, RW1, +d16, RW1	MOVW @RW1, RW1, +d16, RW1	MOVW @RW1, RW2, +d16, RW2	MOVW @RW1, RW2, +d16, RW2	MOVW @RW1, RW3, +d16, RW3	MOVW @RW1, RW3, +d16, RW3	MOVW @RW1, RW4, +d16, RW4	MOVW @RW1, RW4, +d16, RW4	MOVW @RW1, RW5, +d16, RW5	MOVW @RW1, RW5, +d16, RW5	MOVW @RW1, RW6, +d16, RW6	MOVW @RW1, RW6, +d16, RW6	MOVW @RW1, RW7, +d16, RW7	MOVW @RW1, RW7, +d16, RW7
+A	MOVW @RW2, RW0, +d16, RW0	MOVW @RW2, RW0, +d16, RW0	MOVW @RW2, RW1, +d16, RW1	MOVW @RW2, RW1, +d16, RW1	MOVW @RW2, RW2, +d16, RW2	MOVW @RW2, RW2, +d16, RW2	MOVW @RW2, RW3, +d16, RW3	MOVW @RW2, RW3, +d16, RW3	MOVW @RW2, RW4, +d16, RW4	MOVW @RW2, RW4, +d16, RW4	MOVW @RW2, RW5, +d16, RW5	MOVW @RW2, RW5, +d16, RW5	MOVW @RW2, RW6, +d16, RW6	MOVW @RW2, RW6, +d16, RW6	MOVW @RW2, RW7, +d16, RW7	MOVW @RW2, RW7, +d16, RW7
+B	MOVW @RW3, RW0, +d16, RW0	MOVW @RW3, RW0, +d16, RW0	MOVW @RW3, RW1, +d16, RW1	MOVW @RW3, RW1, +d16, RW1	MOVW @RW3, RW2, +d16, RW2	MOVW @RW3, RW2, +d16, RW2	MOVW @RW3, RW3, +d16, RW3	MOVW @RW3, RW3, +d16, RW3	MOVW @RW3, RW4, +d16, RW4	MOVW @RW3, RW4, +d16, RW4	MOVW @RW3, RW5, +d16, RW5	MOVW @RW3, RW5, +d16, RW5	MOVW @RW3, RW6, +d16, RW6	MOVW @RW3, RW6, +d16, RW6	MOVW @RW3, RW7, +d16, RW7	MOVW @RW3, RW7, +d16, RW7
+C	MOVW @RW0+, RW0, +RW7, RW0	MOVW @RW0+, RW0, +RW7, RW0	MOVW @RW0+, RW1, +RW7, RW1	MOVW @RW0+, RW1, +RW7, RW1	MOVW @RW0+, RW2, +RW7, RW2	MOVW @RW0+, RW2, +RW7, RW2	MOVW @RW0+, RW3, +RW7, RW3	MOVW @RW0+, RW3, +RW7, RW3	MOVW @RW0+, RW4, +RW7, RW4	MOVW @RW0+, RW4, +RW7, RW4	MOVW @RW0+, RW5, +RW7, RW5	MOVW @RW0+, RW5, +RW7, RW5	MOVW @RW0+, RW6, +RW7, RW6	MOVW @RW0+, RW6, +RW7, RW6	MOVW @RW0+, RW7, +RW7, RW7	MOVW @RW0+, RW7, +RW7, RW7
+D	MOVW @RW1+, RW0, +RW7, RW0	MOVW @RW1+, RW0, +RW7, RW0	MOVW @RW1+, RW1, +RW7, RW1	MOVW @RW1+, RW1, +RW7, RW1	MOVW @RW1+, RW2, +RW7, RW2	MOVW @RW1+, RW2, +RW7, RW2	MOVW @RW1+, RW3, +RW7, RW3	MOVW @RW1+, RW3, +RW7, RW3	MOVW @RW1+, RW4, +RW7, RW4	MOVW @RW1+, RW4, +RW7, RW4	MOVW @RW1+, RW5, +RW7, RW5	MOVW @RW1+, RW5, +RW7, RW5	MOVW @RW1+, RW6, +RW7, RW6	MOVW @RW1+, RW6, +RW7, RW6	MOVW @RW1+, RW7, +RW7, RW7	MOVW @RW1+, RW7, +RW7, RW7
+E	MOVW @RW2+, RW0, +d16, RW0	MOVW @RW2+, RW0, +d16, RW0	MOVW @RW2+, RW1, +d16, RW1	MOVW @RW2+, RW1, +d16, RW1	MOVW @RW2+, RW2, +d16, RW2	MOVW @RW2+, RW2, +d16, RW2	MOVW @RW2+, RW3, +d16, RW3	MOVW @RW2+, RW3, +d16, RW3	MOVW @RW2+, RW4, +d16, RW4	MOVW @RW2+, RW4, +d16, RW4	MOVW @RW2+, RW5, +d16, RW5	MOVW @RW2+, RW5, +d16, RW5	MOVW @RW2+, RW6, +d16, RW6	MOVW @RW2+, RW6, +d16, RW6	MOVW @RW2+, RW7, +d16, RW7	MOVW @RW2+, RW7, +d16, RW7
+F	MOVW @RW3+, RW0, +d16, RW0	MOVW @RW3+, RW0, +d16, RW0	MOVW @RW3+, RW1, +d16, RW1	MOVW @RW3+, RW1, +d16, RW1	MOVW @RW3+, RW2, +d16, RW2	MOVW @RW3+, RW2, +d16, RW2	MOVW @RW3+, RW3, +d16, RW3	MOVW @RW3+, RW3, +d16, RW3	MOVW @RW3+, RW4, +d16, RW4	MOVW @RW3+, RW4, +d16, RW4	MOVW @RW3+, RW5, +d16, RW5	MOVW @RW3+, RW5, +d16, RW5	MOVW @RW3+, RW6, +d16, RW6	MOVW @RW3+, RW6, +d16, RW6	MOVW @RW3+, RW7, +d16, RW7	MOVW @RW3+, RW7, +d16, RW7

表 B.9-20 XCH Ri, ea 命令 (第1バイト = 7EH)

[illegible]

表 B.9-21 XCHW RWi, ea 命令 (第1バイト = 7FH)

[illegible]

索引

Numerics

1 系統品	
低消費電力モード状態遷移図 (1 系統品)	120
24 ビット	
24 ビットオペランド指定	29
2 系統品	
低消費電力モード状態遷移図 (2 系統品)	115
32 ビット	
32 ビットレジスタ間接指定	29

A

A/D	
A/D 制御ステータスレジスタ 0(ADCS0)	242
A/D 制御ステータスレジスタ 1(ADCS1)	245
A/D コンバータ	
A/D コンバータの特長	238
A/D コンバータのブロックダイアグラム	240
A/D コンバータのレジスタ一覧	241
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 0(ADCS0)	242
A/D 制御ステータスレジスタ 1(ADCS1)	245
A/D データレジスタ	
A/D データレジスタ 0/1(ADCR0, ADCR1)	248
A/D データレジスタ 0/1	
A/D データレジスタ 0/1(ADCR0, ADCR1)	248
ADCR	
A/D データレジスタ 0/1(ADCR0, ADCR1)	248
ADCR0	
A/D データレジスタ 0/1(ADCR0, ADCR1)	248
ADCR0, ADCR1	
A/D データレジスタ 0/1(ADCR0, ADCR1)	248
ADCR1	
A/D データレジスタ 0/1(ADCR0, ADCR1)	248
ADCS	
A/D 制御ステータスレジスタ 0(ADCS0)	242
A/D 制御ステータスレジスタ 1(ADCS1)	245
ADCS0	
A/D 制御ステータスレジスタ 0(ADCS0)	242
ADCS1	
A/D 制御ステータスレジスタ 1(ADCS1)	245
ADER	
アナログ入力許可レジスタ (ADER)	152
AMR0/AMR1	
アクセプタンスマスクレジスタ 0/1 (AMR0/AMR1)	363
AMSR	
アクセプタンスマスク選択レジスタ (AMSR)	361
ARSR	
自動レディ機能選択レジスタ (ARSR)	131

B

BAP	
バッファアドレスポインタ (BAP)	71
BTR	
ビットタイミングレジスタ (BTR)	347
BVAL	
BVAL ビットによるメッセージバッファ禁止時の 注意	385
BVALR	
メッセージバッファ有効レジスタ (BVALR)	349

C

CAN

CAN コントローラの受信フローチャート	376
CAN コントローラの送信の開始	371
CAN コントローラの送信の完了	372
CAN コントローラの送信フローチャート	372
CAN コントローラの送信要求の解除	371
CAN コントローラの特長	330
CAN コントローラのブロック ダイアグラム	331
制御ステータスレジスタ (CSR)	339
CAN コントローラ	
CAN コントローラの受信フローチャート	376
CAN コントローラの送信の開始	371
CAN コントローラの送信の完了	372
CAN コントローラの送信フローチャート	372
CAN コントローラの送信要求の解除	371
CAN コントローラの特長	330
CAN コントローラのブロック ダイアグラム	331
CAN 制御ステータスレジスタ	
CAN 制御ステータスレジスタ (CSR)	339
CCR	
コンディションコードレジスタ (CCR)	37
CDCR	
UART1 通信プリスケラ制御レジスタ (CDCR)	296
CKSCR	
クロック選択レジスタ (CKSCR)	93
CLK	
CLK 同期ポーレート	271
CLK 同期モードの通信開始	303
CLK 同期モードの通信終了	303
CLK 同期モードの転送データ フォーマット	302
CLK 同期モードを使用する場合の各制御 レジスタの設定値	303
CLK 非同期ポーレート	271
CLK 同期ポーレート	
CLK 同期ポーレート	271
CLK 同期モード	
CLK 同期モードの通信開始	303
CLK 同期モードの通信終了	303
CLK 同期モードの転送データ フォーマット	302
CLK 同期モードを使用する場合の各制御 レジスタの設定値	303
CLK 非同期ポーレート	
CLK 非同期ポーレート	271
CMR	
コモンレジスタバンクプリフィックス (CMR)	44
CPU	
CPU 間欠動作	110
CPU 間欠動作機能	86
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	110
CPU 間欠動作機能	86

CSR

CAN 制御ステータスレジスタ (CSR)	339
-----------------------	-----

D

DCT

データカウンタ (DCT)	69
---------------	----

DDR

ポート方向レジスタ (DDR0 ~ DDRA)	148
-------------------------	-----

DDR0

ポート方向レジスタ (DDR0 ~ DDRA)	148
-------------------------	-----

DDRA

ポート方向レジスタ (DDR0 ~ DDRA)	148
-------------------------	-----

DIRR

遅延割込み要因発生 / 解除レジスタ (DIRR)	225
---------------------------	-----

「DIV A,Ri」, 「DIVW A,RWi」

「DIV A,Ri」, 「DIVW A,RWi」 命令の 使用上の注意	47
--	----

DLC

DLC レジスタ x(x=0 ~ 15)(DLCRx)	368
-----------------------------	-----

DLCRx

DLC レジスタ x(x=0 ~ 15)(DLCRx)	368
-----------------------------	-----

DLC レジスタ x

DLC レジスタ x(x=0 ~ 15)(DLCRx)	368
-----------------------------	-----

DQ2

トグルビット 2 フラグ (DQ2)	414
--------------------	-----

DQ3

セクタ消去タイマフラグ (DQ3)	413
-------------------	-----

DQ5

タイミングリミット超過フラグ (DQ5)	412
----------------------	-----

DQ6

トグルビットフラグ (DQ6)	411
-----------------	-----

DQ7

データポーリングフラグ (DQ7)	409
-------------------	-----

DTP

DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)	230
--	-----

DTP/ 外部割込みの概要	228
---------------	-----

DTP/ 外部割込みの使用上の注意	235
-------------------	-----

DTP/ 外部割込みのブロックダイアグラム	228
-----------------------	-----

DTP/ 外部割込みのレジスタ	229
-----------------	-----

DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
--	-----

DTP の動作	232
---------	-----

DTP 要求と外部割込み要求の切換え	234
--------------------	-----

DTP/ 外部割込み

DTP/ 外部割込みの概要	228
---------------	-----

DTP/ 外部割込みの使用上の注意	235
-------------------	-----

DTP/ 外部割込みのブロックダイアグラム	228
-----------------------	-----

DTP/ 外部割込みのレジスタ	229
-----------------	-----

DTP/ 外部割込み許可レジスタ

DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)	230
--	-----

DTP/ 外部割込み要因レジスタ

DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
--	-----

DTP 要求

DTP 要求と外部割込み要求の切換え	234
--------------------	-----

DTRx	
データレジスタ x(x=0 ~ 15)(DTRx)	369

E

ECSR	
バス制御信号選択レジスタ (ECSR)	134
EEPROM	
EEPROM メモリマップ	392
EEPROM メモリマップ	392
EI ² OS	
EI ² OS(拡張インテリジェント I/O サービス)	281
EI ² OS ステータスレジスタ (ISCS)	70
EI ² OS を使った変換動作	252
インテリジェント I/O サービス (EI ² OS) 機能と 割込み	192
拡張インテリジェント I/O サービス (EI ² OS)	310
拡張インテリジェント I/O サービス (EI ² OS) の 概要	64
拡張インテリジェント I/O サービス (EI ² OS) の 構造	65
拡張インテリジェント I/O サービス (EI ² OS) の 実行時間	74
拡張インテリジェント I/O サービス (EI ² OS) の 動作フロー	72
単発モード時の EI ² OS の起動例	253
停止モード時の EI ² OS の起動例	255
変換データ保護機能のフロー例 (EI ² OS の場合)	257
連続モード時の EI ² OS の起動例	254
EI ² OS ステータスレジスタ	
EI ² OS ステータスレジスタ (ISCS)	70
EIRR	
DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
ELVR	
要求レベル設定レジスタ (ELVR:External level register)	231
ENIR	
DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)	230
External interrupt	
DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
External interrupt request register	
DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
External level register	
要求レベル設定レジスタ (ELVR:External level register)	231

F

F ² MC-16LX	
F ² MC-16LX 命令一覧表	477
FMCS	
フラッシュメモリ制御ステータスレジスタ (FMCS)	404

FPT-100P-M05	
FPT-100P-M05 端子配列図	9
FPT-100P-M05 パッケージ外形寸法図	7
FPT-100P-M06	
FPT-100P-M06 端子配列図	8
FPT-100P-M06 パッケージ外形寸法図	6

H

HACR	
外部アドレス出力制御レジスタ (HACR)	133
HALT	
バス動作停止間の状態 (HALT= 1)	343
バス動作停止を解除する条件 (HALT= 0)	342
バス動作停止をセットする条件 (HALT= 1)	342

I

I/O	
EI ² OS(拡張インテリジェント I/O サービス)	281
I/O ポートの概要	144
I/O ポートのレジスター一覧	145
I/O レジスタアドレスポインタ (IOA)	70
インテリジェント I/O サービス (EI ² OS) 機能と 割込み	192
拡張インテリジェント I/O サービス (EI ² OS)	310
拡張インテリジェント I/O サービス (EI ² OS) の 概要	64
拡張インテリジェント I/O サービス (EI ² OS) の 構造	65
拡張インテリジェント I/O サービス (EI ² OS) の 実行時間	74
拡張インテリジェント I/O サービス (EI ² OS) の 動作フロー	72
拡張インテリジェント I/O サービス ディスクリプタ (ISD)	69
シリアル I/O の動作	320
シリアル I/O の動作状態	322
シリアル I/O のブロックダイアグラム	312
シリアル I/O のレジスタ	313
シリアル I/O の割込み機能	327
シリアル I/O ブリスケラ (SCDCR)	319
I/O ポート	
I/O ポートの概要	144
I/O ポートのレジスター一覧	145
I/O マップ	
I/O マップ	446
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)	70
ICR	
割込み制御レジスタ (ICR)	66
ICS01	
インプットキャプチャ制御ステータスレジスタ (ICS01)	187
ID	
ID の設定	377
ID レジスタ x(x=0 ~ 15)(IDRx)	366
メッセージバッファ (ID レジスタ) の 一覧	334

IDE	
IDE レジスタ (IDER)	350
IDER	
IDE レジスタ (IDER)	350
IDE レジスタ	
IDE レジスタ (IDER)	350
IDRx	
ID レジスタ $x(x=0 \sim 15)(IDRx)$	366
ID レジスタ	
ID レジスタ $x(x=0 \sim 15)(IDRx)$	366
ILM	
割込みレベルマスクレジスタ (ILM)	39
Interrupt	
DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)	230
DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
Interrupt request enable register	
DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)	230
IOA	
I/O レジスタアドレスポインタ (IOA)	70
IPCP0/1	
インプットキャプチャデータレジスタ (IPCP0/1)	187
ISCS	
EI ² OS ステータスレジスタ (ISCS)	70
ISD	
拡張インテリジェント I/O サービス ディスクリプタ (ISD)	69
L	
LEIR	
ラストイベント表示レジスタ (LEIR)	344
LPMCR	
低消費電力モード制御レジスタ (LPMCR)	91
N	
NCC	
フラグ変化抑止プリフィックス (NCC)	44
O	
OCCP	
アウトプットコンペアのコンペアレジスタ (OCCP)	180
OCS0/1	
アウトプットコンペアの制御ステータスレジスタ (OCS0/1)	181
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)	389
PADR0/PADR1	
プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1)	389

PC	
プログラムカウンタ (PC)	40
PDR	
ポートデータレジスタ (PDR0 ~ PDRA)	146
PDRO	
ポートデータレジスタ (PDR0 ~ PDRA)	146
PDRA	
ポートデータレジスタ (PDR0 ~ PDRA)	146
PLL クロック	
PLL クロック逓倍機能	88
PLL クロック逓倍機能	
PLL クロック逓倍機能	88
PPG	
8/16 ビット PPG のカウントクロックの選択	218
8/16 ビット PPG の各ハードウェアの初期値	221
8/16 ビット PPG の機能	206
8/16 ビット PPG の出力動作	217
8/16 ビット PPG の動作	216
8/16 ビット PPG の動作モード	216
8/16 ビット PPG のパルスの端子出力の制御	219
8/16 ビット PPG のブロックダイヤグラム	207
8/16 ビット PPG のリロード値とパルス幅の関係	217
8/16 ビット PPG のレジスタ	209
8/16 ビット PPG の割込み	220
PPG0 動作モード制御レジスタ (PPGC0)	210
PPG1 動作モード制御レジスタ (PPGC1)	212
PPG ユニット 0 クロック選択レジスタ (PPG01)	214
PPG01	
PPG ユニット 0 クロック選択レジスタ (PPG01)	214
PPG0 動作モード制御レジスタ	
PPG0 動作モード制御レジスタ (PPGC0)	210
PPG1 動作モード制御レジスタ	
PPG1 動作モード制御レジスタ (PPGC1)	212
PPGC	
PPG0 動作モード制御レジスタ (PPGC0)	210
PPG1 動作モード制御レジスタ (PPGC1)	212
PPGC0	
PPG0 動作モード制御レジスタ (PPGC0)	210
PPGC1	
PPG1 動作モード制御レジスタ (PPGC1)	212
PPG ユニット 0	
PPG ユニット 0 クロック選択レジスタ (PPG01)	214
PPG ユニット 0 クロック選択レジスタ	
PPG ユニット 0 クロック選択レジスタ (PPG01)	214
PRL, PRLH	
リロードレジスタ (PRL, PRLH)	215
PS	
プロセッサステータス (PS)	37
PUCR	
ブルアップ制御レジスタ (PUCR0 ~ PUCR3)	150
ブルアップ制御レジスタ (PUCR0 ~ PUCR3) のブロックダイヤグラム	151

R

RCR	
受信完了レジスタ (RCR)	357
RFWTR	
リモートフレーム受信待機レジスタ (RFWTR)	353
RIER	
受信割込み許可レジスタ (RIER)	360
ROM	
ROM ミラー機能選択モジュールのブロック	
ダイアグラム	396
ROM ミラー機能選択レジスタ (ROMM)	397
ROMM	
ROM ミラー機能選択レジスタ (ROMM)	397
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	396
ROM ミラー機能選択レジスタ	
ROM ミラー機能選択レジスタ (ROMM)	397
ROVRR	
受信オーバーランレジスタ (ROVRR)	359
RP	
レジスタバンクポインタ (RP)	38
RRTRR	
リモート要求受信レジスタ (RRTRR)	358
RTEC	
受信および送信エラーカウンタ (RTEC)	346

S

SCDCR	
シリアル I/O プリスケアラ (SCDCR)	319
SCR	
シリアル制御レジスタ 1 (SCR1)	291
SCR1	
シリアル制御レジスタ 1 (SCR1)	291
SDR	
シリアルシフトデータレジスタ (SDR)	318
SIDR	
シリアルインプットデータレジスタ 1 (SIDR1)/	
シリアルアウトプットデータレジスタ 1	
(SODR1)	293
SMCS	
シリアルモード制御ステータスレジスタ (SMCS)	314
SMR	
シリアルモードレジスタ 1 (SMR1)	289
SODR	
シリアルインプットデータレジスタ 1 (SIDR1)/	
シリアルアウトプットデータレジスタ 1	
(SODR1)	293
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)	36
SSR	
シリアルステータスレジスタ 1 (SSR1)	294

T

TBTC	
タイムベースタイマ制御レジスタ (TBTC)	155
TCANR	
送信解除レジスタ (TCANR)	354
TCCS	
タイマカウンタ制御ステータスレジスタ (TCCS)	175
TCDT	
タイマカウンタデータレジスタ (TCDT)	174
TCR	
送信完了レジスタ (TCR)	355
TIER	
送信割込み許可レジスタ (TIER)	356
TMCSR	
タイマ制御ステータスレジスタ (TMCSR)	195
TMR	
16 ビットタイマレジスタ (TMR)/16 ビット	
リロードレジスタ (TMRLR)	198
TMRLR	
16 ビットタイマレジスタ (TMR)/16 ビット	
リロードレジスタ (TMRLR)	198
TREQR	
送信要求レジスタ (TREQR)	351
TRTRR	
送信 RTR レジスタ (TRTRR)	352

U

UART	
UART0 の応用例	282
UART0 の動作モード	270
UART0 の特長	260
UART0 のブロックダイアグラム	261
UART0 のレジスタ	262
UART1 通信プリスケアラ制御レジスタ (CDCR)	296
UART1 の応用例 (モード 1 使用時のシステム	
構築例)	309
UART1 のクロック選択	297
UART1 の使用上の注意	310
UART1 の通信フローチャート	309
UART1 の動作モード	297
UART1 の特長	286
UART1 のフラグ	304
UART1 のブロックダイアグラム	287
UART1 のレジスタ一覧	288
UART1 の割込みとフラグのセット	
タイミング	305
UART1 の割込み発生要因	304
UART0	
UART0 の応用例	282
UART0 の動作モード	270
UART0 の特長	260
UART0 のブロックダイアグラム	261
UART0 のレジスタ	262
UART1	
UART1 通信プリスケアラ制御レジスタ (CDCR)	296

UART1 の応用例 (モード 1 使用時のシステム構築例)	309
UART1 のクロック選択	297
UART1 の使用上の注意	310
UART1 の通信フローチャート	309
UART1 の動作モード	297
UART1 の特長	286
UART1 のフラグ	304
UART1 のブロックダイアグラム	287
UART1 のレジスタ一覧	288
UART1 の割込みとフラグのセット タイミング	305
UART1 の割込み発生要因	304
UART1 通信プリスケアラ制御レジスタ UART1 通信プリスケアラ制御レジスタ (CDCR)	296
UIDR シリアルインプットデータレジスタ 0(UIDR0) と シリアルアウトプットデータレジスタ	267
UMC シリアルモード制御レジスタ 0(UMC0)	263
UMC0 シリアルモード制御レジスタ 0(UMC0)	263
UODR シリアルインプットデータレジスタ 0(UIDR0) と シリアルアウトプットデータレジスタ	267
URD レート・データレジスタ 0(URD0)	268
URD0 レート・データレジスタ 0(URD0)	268
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)	36
USR シリアルステータスレジスタ 0(USR0)	265
W	
WDTC ウォッチドッグタイマ制御レジスタ (WDTC)	159
WTC 時計タイマ制御レジスタ (WTC)	165

あ

アウトプットコンペア アウトプットコンペア	179
アウトプットコンペア (モジュールごとに 2 チャンネル)	170
アウトプットコンペアのコンペアレジスタ (OCCP)	180
アウトプットコンペアの制御ステータスレジスタ (OCS0/1)	181
アウトプットコンペアのタイミング	185
アウトプットコンペアのブロック ダイアグラム	179
アキュムレータ アキュムレータ (A)	35
アクセス 多バイト長データのアクセス	32
低消費電力モード制御レジスタのアクセス	92
アクセプタンスフィルタ アクセプタンスフィルタの設定	378
アクセプタンスフィルタリング アクセプタンスフィルタリング	373
アクセプタンスマスク アクセプタンスマスク選択レジスタ (AMSR)	361
アクセプタンスマスクレジスタ 0/1 (AMR0/AMR1)	363
アクセプタンスマスク選択レジスタ アクセプタンスマスク選択レジスタ (AMSR)	361
アクセプタンスマスクレジスタ 0/1 アクセプタンスマスクレジスタ 0/1 (AMR0/AMR1)	363
アドレス一致検出機能 アドレス一致検出機能のシステム構成例	392
アドレス一致検出機能の動作	391
アドレス一致検出機能のブロック ダイアグラム	388
アドレス指定 アドレス指定の方式	27
バンク方式によるアドレス指定	30
アドレッシング アドレッシング	454
間接アドレッシング	463
直接アドレッシング	456
アナログ入力許可レジスタ アナログ入力許可レジスタ	238
アナログ入力許可レジスタ (ADER)	152
アンダフロー動作 16 ビットリロードタイマの アンダフロー動作	201

い

一時停止 フラッシュメモリのセクタ消去の 一時停止	423
イベントカウント 16 ビットリロードタイマ (イベントカウント 機能付き) の概要	192
外部イベントカウント	200

インターバル割込み	
タイムベースタイマのインターバル	
割込み機能	156
時計タイマのインターバル割込み機能	167
インテリジェント I/O サービス	
EI ² OS(拡張インテリジェント I/O	
サービス)	281
インテリジェント I/O サービス (EI ² OS) 機能と	
割込み	192
拡張インテリジェント I/O サービス	
(EI ² OS)	310
拡張インテリジェント I/O サービス (EI ² OS) の	
概要	64
拡張インテリジェント I/O サービス (EI ² OS) の	
構造	65
拡張インテリジェント I/O サービス (EI ² OS) の	
実行時間	74
拡張インテリジェント I/O サービス (EI ² OS) の	
動作フロー	72
インプットキャプチャ	
インプットキャプチャ	186
インプットキャプチャ (モジュールごとに	
2 チャンネル)	170
インプットキャプチャ制御ステータスレジスタ	
(ICS01)	187
インプットキャプチャ入力タイミング	189
インプットキャプチャの取込み	
タイミング例	189
インプットキャプチャのブロック	
ダイアグラム	186
インプットキャプチャ制御ステータスレジスタ	
インプットキャプチャ制御ステータスレジスタ	
(ICS01)	187
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/I)	187
う	
ウォッチドッグカウンタ	
ウォッチドッグカウンタのクリア	161
ウォッチドッグカウンタの停止	161
ウォッチドッグタイマ	
ウォッチドッグタイマ制御レジスタ	
(WDTC)	159
ウォッチドッグタイマの起動方法	161
ウォッチドッグタイマの動作クロックの	
設定	167
ウォッチドッグタイマのブロック	
ダイアグラム	158
ウォッチドッグタイマのリセット	161
ウォッチドッグタイマのレジスタ一覧	158
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)	159
お	
応用例	
UART0 の応用例	282
オペランド	
24 ビットオペランド指定	29

オペランド指定	
24 ビットオペランド指定	29

か

外形寸法図	
FPT-100P-M05 パッケージ外形寸法図	7
FPT-100P-M06 パッケージ外形寸法図	6
解除レジスタ	
遅延割込み要因発生 / 解除レジスタ (DIRR)	225
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)	133
外部イベントカウンタ	
外部イベントカウンタ	200
外部クロック	
内部クロックおよび外部クロック	274
外部シフトクロック	
外部シフトクロックモード	321
外部シフトクロックモード	
外部シフトクロックモード	321
外部バス端子制御回路	
外部メモリアクセス (外部バス端子	
制御回路)	129
外部メモリアクセス	
外部メモリアクセス (外部バス端子	
制御回路)	129
外部メモリアクセス制御信号	137
外部メモリアクセスのブロック	
ダイアグラム	129
外部メモリアクセスのレジスタ一覧	130
外部メモリアクセス制御信号	
外部メモリアクセス制御信号	137
外部割込み	
DTP/ 外部割込みの概要	228
DTP/ 外部割込みの使用上の注意	235
DTP/ 外部割込みのブロックダイアグラム	228
DTP/ 外部割込みのレジスタ	229
DTP 要求と外部割込み要求の切換え	234
外部割込みの動作	232
外部割込み要求	
DTP 要求と外部割込み要求の切換え	234
カウンタ	
カウンタの動作状態	203
受信および送信エラーカウンタ (RTEC)	346
カウントクロック	
8/16 ビット PPG のカウントクロックの	
選択	218
書込み	
シリアル書込み接続例 (ユーザ電源	
使用時)	437
シリアル書込み接続例 (ライタから	
電源供給時)	439
入出力領域への書込み中のハードウェア割込み	
要求	55
フラッシュメモリ書込み / 消去の詳細説明	416
フラッシュメモリ書込み / 消去の方法	400
フラッシュメモリ書込み手順	418
フラッシュメモリへのデータを書込み	418

拡張インテリジェント I/O サービス	
EI ² OS(拡張インテリジェント I/O	
サービス)	281
拡張インテリジェント I/O サービス	
(EI ² OS)	310
拡張インテリジェント I/O サービス (EI ² OS) の	
概要	64
拡張インテリジェント I/O サービス (EI ² OS) の	
構造	65
拡張インテリジェント I/O サービス (EI ² OS) の	
実行時間	74
拡張インテリジェント I/O サービス (EI ² OS) の	
動作フロー	72
拡張インテリジェント I/O サービスディスクリプタ	
拡張インテリジェント I/O サービス	
ディスクリプタ (ISD)	69
間欠動作	
CPU 間欠動作	110
CPU 間欠動作機能	86
間接アドレッシング	
間接アドレッシング	463

き

擬似時計モード	
擬似時計モードの解除	105
擬似時計モードへの遷移	105
起動方法	
ウォッチドッグタイマの起動方法	161
基本構成	
シリアル書込み接続の基本構成	434

く

クロック	
16 ビットリロードタイマの	
内部クロック動作	199
16 ビットリロードタイマの入力端子機能	
(内部クロックモード時)	199
PLL クロック逡倍機能	88
PPG ユニット 0 クロック選択レジスタ	
(PPG01)	214
UART1 のクロック選択	297
ウォッチドッグタイマの動作クロックの	
設定	167
クロック選択の状態遷移	96
クロック選択レジスタ (CKSCR)	93
クロック発生部の注意	78
内部クロックおよび外部クロック	274
発振クロック周波数とシリアルクロック	
入力周波数	436
メインクロック発振安定待ち時間設定	86
クロック選択	
PPG ユニット 0 クロック選択レジスタ	
(PPG01)	214
UART1 のクロック選択	297
クロック選択の状態遷移	96
クロック選択レジスタ (CKSCR)	93
クロック選択レジスタ	
PPG ユニット 0 クロック選択レジスタ	
(PPG01)	214
クロック選択レジスタ (CKSCR)	93

クロック発生部	
クロック発生部の注意	78
クロックモード	
16 ビットリロードタイマの入力端子機能	
(内部クロックモード時)	199

こ

コマンドシーケンス表	
コマンドシーケンス表	406
コモンレジスタ	
コモンレジスタバンクプリフィックス	
(CMR)	44
コモンレジスタバンクプリフィックス	
コモンレジスタバンクプリフィックス	
(CMR)	44
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)	37
コンディションコードレジスタ	
コンディションコードレジスタ (CCR)	37
コンペアレジスタ	
2 組のコンペアレジスタによる出力波形例	
(出力の初期値は 0 とする)	184
アウトプットコンペアのコンペアレジスタ	
(OCCP)	180
コンペアレジスタ 0, 1 を使用したときの出力	
波形例 (出力の初期値は 0 とする)	184
コンペアレジスタ 0, 1	
コンペアレジスタ 0, 1 を使用したときの出力	
波形例 (出力の初期値は 0 とする)	184

し

システム構成例	
アドレス一致検出機能のシステム構成例	392
システム構築例	
UART1 の応用例 (モード 1 使用時のシステム	
構築例)	309
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)	36
実効アドレス	
実効アドレスフィールド	472
実効アドレスフィールド	
実効アドレスフィールド	455
実行サイクル	
実行サイクル数	470
実行サイクル数計算方法	470
実行時間	
拡張インテリジェント I/O サービス (EI ² OS) の	
実行時間	74
自動	
自動レディ機能選択レジスタ (ARSR)	131
シフト動作	
シフト動作のスタート / ストップ	
タイミング	324
受信	
受信オーバーラン	374
受信オーバーランレジスタ (ROVRR)	359
受信および送信エラーカウンタ (RTEC)	346
受信メッセージの格納	373

受信オーバラン	
受信オーバラン	374
受信オーバランレジスタ	
受信オーバランレジスタ (ROVRR)	359
受信および送信エラーカウンタ	
受信および送信エラーカウンタ (RTEC)	346
受信完了	
受信完了	375
受信完了レジスタ (RCR)	357
受信完了レジスタ	
受信完了レジスタ (RCR)	357
受信動作	
非同期 (調歩同期) モードの受信動作	300
受信動作時	
受信動作時 (モード 0, モード 1, モード 3) の フラグのセットタイミング	278
受信動作時 (モード 2) のフラグのセット タイミング	279
受信フローチャート	
CAN コントローラの受信フローチャート	376
受信メッセージ	
受信メッセージの格納	373
受信割込み	
受信割込み許可レジスタ (RIER)	360
受信割込み許可レジスタ	
受信割込み許可レジスタ (RIER)	360
出力端子	
16 ビットリロードタイマの出力端子機能	202
出力波形例	
2 組のコンペアレジスタによる出力波形例 (出力の初期値は 0 とする)	184
コンペアレジスタ 0, 1 を使用したときの出力 波形例 (出力の初期値は 0 とする)	184
状態遷移図	
低消費電力モード状態遷移図 (1 系統品)	120
低消費電力モード状態遷移図 (2 系統品)	115
初期化	
リセット入力によって初期化されない レジスタ	80
初期値	
2 組のコンペアレジスタによる出力波形例 (出力の初期値は 0 とする)	184
8/16 ビット PPG の各ハードウェアの 初期値	221
コンペアレジスタ 0, 1 を使用したときの出力 波形例 (出力の初期値は 0 とする)	184
シリアル	
シリアル制御レジスタ 1 (SCR1)	291
シリアルモード制御ステータスレジスタ (SMCS)	314
シリアルモード制御レジスタ 0 (UMC0)	263
シリアル I/O	
シリアル I/O の動作	320
シリアル I/O の動作状態	322
シリアル I/O のブロックダイアグラム	312
シリアル I/O のレジスタ	313
シリアル I/O の割込み機能	327
シリアル I/O プリスケアラ (SCDCR)	319
シリアル I/O プリスケアラ	
シリアル I/O プリスケアラ (SCDCR)	319

シリアルアウトプットデータレジスタ	
シリアルインプットデータレジスタ 0 (UIDR0) と シリアルアウトプットデータレジスタ	267
シリアルインプットデータレジスタ 1 (SIDR1)/ シリアルアウトプットデータレジスタ 1 (SODR1)	293
シリアルインプットデータレジスタ	
シリアルインプットデータレジスタ 0 (UIDR0) と シリアルアウトプットデータレジスタ	267
シリアルインプットデータレジスタ 1 (SIDR1)/ シリアルアウトプットデータレジスタ 1 (SODR1)	293
シリアル書込み	
シリアル書込み接続の基本構成	434
シリアル書込み接続	
シリアル書込み接続の基本構成	434
シリアル書込み接続例	
シリアル書込み接続例 (ユーザ電源 使用時)	437
シリアル書込み接続例 (ライタから 電源供給時)	439
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	436
シリアルクロック入力周波数	
発振クロック周波数とシリアルクロック 入力周波数	436
シリアルシフトデータレジスタ	
シリアルシフトデータレジスタ (SDR)	318
シリアルステータス	
シリアルステータスレジスタ 0 (USR0)	265
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 (USR0)	265
シリアルステータスレジスタ 1	
シリアルステータスレジスタ 1 (SSR1)	294
シリアル制御レジスタ	
シリアル制御レジスタ 1 (SCR1)	291
シリアルモード制御ステータスレジスタ	
シリアルモード制御ステータスレジスタ (SMCS)	314
シリアルモード制御レジスタ	
シリアルモード制御レジスタ 0 (UMC0)	263
シリアルモードレジスタ	
シリアルモードレジスタ 1 (SMR1)	289

す

推奨設定	
推奨設定	128
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	324
スタック	
スタック内へのレジスタの退避	56
ステータスフラグ	
送受信動作時のステータスフラグ	281
ステータスレジスタ 0	
A/D 制御ステータスレジスタ 0 (ADCS0)	242

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	324
ストップモード	
ストップモードの解除	107
ストップモードへの遷移	107
スリープモード	
スリープモードの解除	104
スリープモードへの遷移	104

せ

制御	
8/16 ビット PPG のパルスの端子出力の	
制御	219
制御信号	
フラッシュメモリの制御信号	402
制御ステータスレジスタ	
アウトプットコンペアの制御ステータスレジスタ	
(OCS0/1)	181
製品概要	
MB90540/545 シリーズの製品概要	2
セクタ構成	
1M/2M ビットフラッシュメモリの	
セクタ構成	401
セクタ消去	
セクタ消去タイマフラグ (DQ3)	413
フラッシュメモリのセクタ消去の	
一時停止	423
フラッシュメモリのセクタ消去の再開	424
フラッシュメモリの任意データの消去	
(セクタ消去)	421
セクタ消去タイマフラグ	
セクタ消去タイマフラグ (DQ3)	413
セクタ消去手順	
フラッシュメモリのセクタ消去手順	421
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 (ライタから電源供給時)	443
セットタイミング	
6 つのフラグのセットタイミング	277
UART1 の割込みとフラグのセット	
タイミング	305
受信動作時 (モード 0, モード 1, モード 3) の	
フラグのセットタイミング	278
受信動作時 (モード 2) のフラグのセット	
タイミング	279
送信動作時のフラグのセットタイミング	280
遷移条件	
低消費電力モードの遷移条件	111
全体制御	
全体制御レジスタ	332, 338
全体制御レジスタ	
全体制御レジスタ	332, 338
専用レジスタ	
専用レジスタ	33

そ

送受信動作時	
送受信動作時のステータスフラグ	281

送信	
送信 RTR レジスタ (TRTRR)	352
送信 RTR レジスタ	
送信 RTR レジスタ (TRTRR)	352
送信解除	
送信解除レジスタ (TCANR)	354
送信解除レジスタ	
送信解除レジスタ (TCANR)	354
送信完了	
送信完了レジスタ (TCR)	355
送信完了レジスタ	
送信完了レジスタ (TCR)	355
送信動作	
非同期 (調歩同期) モードの送信動作	301
送信動作時	
送信動作時のフラグのセットタイミング	280
送信フローチャート	
CAN コントローラの送信フローチャート	372
送信要求	
CAN コントローラの送信要求の解除	371
送信要求レジスタ (TREQR)	351
送信要求レジスタ	
送信要求レジスタ (TREQR)	351
送信割込み	
送信割込み許可レジスタ (TIER)	356
送信割込み許可レジスタ	
送信割込み許可レジスタ (TIER)	356
その他モード	
その他モード	402
ソフトウェア割込み	
ソフトウェア割込みに関する注意事項	63
ソフトウェア割込みの概要	62
ソフトウェア割込みの構造	62
ソフトウェア割込みの動作	63

た

タイマ	
16 ビット入出力タイマの	
ブロックダイヤグラム	171
16 ビット入出力タイマのレジスタ	172
16 ビットリロードタイマ (イベントカウン	
機能付き) の概要	192
16 ビットリロードタイマの	
アンダフロー動作	201
16 ビットリロードタイマの出力端子機能	202
16 ビットリロードタイマの端子名	193
16 ビットリロードタイマの	
内部クロック動作	199
16 ビットリロードタイマの入力端子機能	
(内部クロックモード時)	199
16 ビットリロードタイマのブロック	
ダイヤグラム	192
16 ビットリロードタイマのレジスタ	194
タイマ制御ステータスレジスタ (TMCSR)	195
時計タイマ	167
時計タイマ制御レジスタ (WTC)	165
時計タイマのインターバル割込み機能	167
時計タイマのブロックダイヤグラム	164
時計タイマのレジスタ	164

タイマカウンタ	
タイマカウンタ制御ステータスレジスタ (TCCS)	175
タイマカウンタ制御ステータスレジスタ	
タイマカウンタ制御ステータスレジスタ (TCCS)	175
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)	174
タイマレジスタ	
16 ビットタイマレジスタ (TMR)/16 ビット	
リロードレジスタ (TMRLR)	198
タイミング	
16 ビットフリーランタイムのタイミング	178
アウトプットコンペアのタイミング	185
インプットキャプチャ入力タイミング	189
インプットキャプチャの取込み	
タイミング例	189
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)	412
タイムベースタイマ	
タイムベースタイマ	156
タイムベースタイマ制御レジスタ (TBTC)	155
タイムベースタイマのインターバル	
割込み機能	156
タイムベースタイマのブロック	
ダイヤグラム	154
タイムベースタイマのレジスター一覧	154
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)	155
多重割込み	
多重割込み	56
多バイト長データ	
多バイト長データのアクセス	32
メモリ空間における多バイト長データの配置	32
端子機能	
端子機能説明	10
端子機能説明	
端子機能説明	10
端子出力	
8/16 ビット PPG のパルスの端子出力の	
制御	219
端子配列図	
FPT-100P-M05 端子配列図	9
FPT-100P-M06 端子配列図	8
単発モード	
単発モード	250
単発モード時の EI ² OS の起動例	253

ち

遅延割込み	
遅延割込みのブロックダイヤグラム	224
遅延割込み発生	226
遅延割込み要因発生 / 解除レジスタ (DIRR)	225
遅延割込み要求ロックの使用上の注意	224
遅延割込み要因発生 / 解除	
遅延割込み要因発生 / 解除レジスタ (DIRR)	225

遅延割込み要求ロック	
遅延割込み要求ロックの使用上の注意	224
チップ消去	
フラッシュメモリの全データの消去	
(チップ消去)	420
注意事項	
ソフトウェア割込みに関する注意事項	63
注意事項の回避について	48
デバイスの取扱いに関する注意事項	20
調歩同期	
非同期 (調歩同期) モードの受信動作	300
非同期 (調歩同期) モードの送信動作	301
非同期 (調歩同期) モードの転送データ	
フォーマット	300
直接アドレッシング	
直接アドレッシング	456

つ

通信フローチャート	
UART1 の通信フローチャート	309

て

停止モード	
停止モード	251
停止モード時の EI ² OS の起動例	255
低消費電力制御回路	
低消費電力制御回路の動作モード	86
低消費電力制御回路のブロック	
ダイヤグラム	89
低消費電力制御回路のレジスタ	90
低消費電力モード	
低消費電力モード	100
低消費電力モード状態遷移図 (1 系統品)	120
低消費電力モード状態遷移図 (2 系統品)	115
低消費電力モード制御レジスタ (LPMCR)	91
低消費電力モード制御レジスタのアクセス	
	92
低消費電力モード動作状態	102
低消費電力モードの遷移条件	111
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)	91
低消費電力モード制御レジスタのアクセス	
	92
低電力	
低電力モードの設定	378
低電力モード	
低電力モードの設定	378
データカウンタ	
データカウンタ (DCT)	69
データフレーム	
データフレームおよびリモートフレームの受信に	
対する処理	374
データボーリングフラグ	
データボーリングフラグ (DQ7)	409
データレジスタ	
A/D データレジスタ 0/1 (ADCR0, ADCR1)	248
データレジスタ x (x=0 ~ 15) (DTRx)	369
ポートデータレジスタ (PDR0 ~ PDRA)	146
メッセージバッファ (データレジスタ) の	
一覧	337

レート・データレジスタ 0(URD0)	268
データレジスタ x	
データレジスタ x(x=0 ~ 15)(DTRx)	369
デバイス	
デバイスの取扱いに関する注意事項	20
デバイスの取扱	
デバイスの取扱いに関する注意事項	20
電源供給時	
フラッシュマイコンプログラマとの最小限の	
接続例 (ライタから電源供給時)	443
転送データ	
CLK 同期モードの転送データ	
フォーマット	302
転送データフォーマット	275
非同期 (調歩同期) モードの転送データ	
フォーマット	300
転送データフォーマット	
CLK 同期モードの転送データ	
フォーマット	302
転送データフォーマット	275
非同期 (調歩同期) モードの転送データ	
フォーマット	300
と	
動作クロック	
ウォッチドッグタイマの動作クロックの	
設定	167
動作状態	
低消費電力モード動作状態	102
動作フロー	
拡張インテリジェント I/O サービス (EI ² OS) の	
動作フロー	72
ハードウェア割込みの動作フロー	59
動作モード	
8/16 ビット PPG の動作モード	216
UART0 の動作モード	270
UART1 の動作モード	297
低消費電力制御回路の動作モード	86
特長	
A/D コンバータの特長	238
CAN コントローラの特長	330
UART0 の特長	260
UART1 の特長	286
特長	3
フラッシュセキュリティの特長	426
トグルビット 2 フラグ	
トグルビット 2 フラグ (DQ2)	414
トグルビットフラグ	
トグルビットフラグ (DQ6)	411
時計タイマ	
時計タイマ	167
時計タイマ制御レジスタ (WTC)	165
時計タイマのインターバル割込み機能	167
時計タイマのブロックダイヤグラム	164
時計タイマのレジスタ	164
時計タイマ制御レジスタ	
時計タイマ制御レジスタ (WTC)	165
時計モード	
時計モードの解除	106
時計モードへの遷移	106

な

内部クロック

16 ビットリロードタイマの	
内部クロック動作	199
16 ビットリロードタイマの入力端子機能	
(内部クロックモード時)	199
内部クロックおよび外部クロック	274
内部クロックモード	
16 ビットリロードタイマの入力端子機能	
(内部クロックモード時)	199
内部シフトクロック	
内部シフトクロックモード	321
内部シフトクロックモード	
内部シフトクロックモード	321

に

入出力回路形式

入出力回路形式	17
入出力タイマ	
16 ビット入出力タイマのブロック	
ダイヤグラム	171
16 ビット入出力タイマのレジスタ	172
入出力領域	
入出力領域への書き込み中のハードウェア割込み	
要求	55
入出力領域への書き込み	
入出力領域への書き込み中のハードウェア割込み	
要求	55
入カインピーダンス	
入カインピーダンス	239
入力タイミング	
インプットキャプチャ入力タイミング	189
入力端子	
16 ビットリロードタイマの入力端子機能	
(内部クロックモード時)	199

ね

ネガティブクロック動作

ネガティブクロック動作	308, 328
-------------------	----------

は

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	407
ハードウェアスタンバイモード	
ハードウェアスタンバイモードの解除	109
ハードウェアスタンバイモードへの遷移	109
ハードウェア割込み	
入出力領域への書き込み中のハードウェア割込み	
要求	55
ハードウェア割込みの概要	55
ハードウェア割込みの構造	55
ハードウェア割込みの動作	57
ハードウェア割込みの動作フロー	59
バス	
バス動作停止間の状態 (HALT= 1)	343
バス動作停止を解除する条件 (HALT= 0)	342

バス動作停止をセットする条件 (HALT= 1)	342
バス制御信号選択レジスタ	
バス制御信号選択レジスタ (ECSR)	134
バス動作停止	
バス動作停止を解除する条件 (HALT= 0)	342
バス動作停止をセットする条件 (HALT= 1)	342
バス動作停止間	
バス動作停止間の状態 (HALT= 1)	343
バスモード	
バスモード別メモリ空間	127
発振安定待ち	
メインクロック発振安定待ち時間設定	86
発振安定待ち時間	
メインクロック発振安定待ち時間設定	86
発振クロック周波数	
発振クロック周波数とシリアルクロック	
入力周波数	436
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)	71
パリティビット	
パリティビット	276
パルス	
8/16 ビット PPG のパルスの端子出力の	
制御	219
パルス幅	
8/16 ビット PPG のリロード値とパルス幅の	
関係	217
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	43
バンク方式	
バンク方式によるアドレス指定	30
汎用レジスタ	
汎用レジスタ	34

ひ

ビット	
ビットタイミングの設定	377
ビットタイミングレジスタ (BTR)	347
ビットタイミング	
ビットタイミングの設定	377
ビットタイミングレジスタ	
ビットタイミングレジスタ (BTR)	347
非同期	
CLK 非同期ボーレート	271
非同期 (調歩同期) モードの受信動作	300
非同期 (調歩同期) モードの送信動作	301
非同期 (調歩同期) モードの転送データ	
フォーマット	300

ふ

フラグ	
6 つのフラグのセットタイミング	277
UART1 のフラグ	304
UART1 の割込みとフラグのセット	
タイミング	305
受信動作時 (モード 0, モード 1, モード 3) の	
フラグのセットタイミング	278

受信動作時 (モード 2) のフラグのセット	
タイミング	279
送信動作時のフラグのセットタイミング	280
フラグ変化抑止	
フラグ変化抑止プリフィックス (NCC)	44
フラグ変化抑止プリフィックス	
フラグ変化抑止プリフィックス (NCC)	44
フラッシュセキュリティ	
フラッシュセキュリティの特長	426
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)	441
フラッシュマイコンプログラマとの最小限の	
接続例 (ライタから電源供給時)	443
フラッシュメモリ	
1M/2M ビットフラッシュメモリの	
セクタ構成	401
1M/2M ビットフラッシュメモリの特長	400
1M/2M ビットフラッシュメモリの	
プログラム例	427
フラッシュメモリ書込み / 消去の詳細説明	416
フラッシュメモリ書込み / 消去の方法	400
フラッシュメモリ書込み手順	418
フラッシュメモリ使用上の注意	425
フラッシュメモリ制御ステータスレジスタ	
(FMCS)	404
フラッシュメモリの制御信号	402
フラッシュメモリのセクタ消去手順	421
フラッシュメモリのセクタ消去の	
一時停止	423
フラッシュメモリのセクタ消去の再開	424
フラッシュメモリの全データの消去	
(チップ消去)	420
フラッシュメモリの任意データの消去	
(セクタ消去)	421
フラッシュメモリの読出し / リセット状態	417
フラッシュメモリのレジスタ	400
フラッシュメモリへのデータを書込み	418
フラッシュメモリモード	402
フラッシュメモリ制御ステータスレジスタ	
フラッシュメモリ制御ステータスレジスタ	404
(FMCS)	
フラッシュメモリのレジスタ	
フラッシュメモリのレジスタ	400
フラッシュメモリモード	
フラッシュメモリモード	402
フリーランタイマ	
16 ビットフリーランタイマ	170
16 ビットフリーランタイマのタイミング	178
16 ビットフリーランタイマの動作	177
16 ビットフリーランタイマのブロック	
ダイアグラム	173
プリフィックス	
フラグ変化抑止プリフィックス (NCC)	44
割込み抑止命令とプリフィックス命令に関する	
制約	45
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	46
プリフィックス命令	
割込み抑止命令とプリフィックス命令に関する	
制約	45

プルアップ制御レジスタ	
プルアップ制御レジスタ (PUCR0 ~ PUCR3)	150
プルアップ制御レジスタ (PUCR0 ~ PUCR3) の ブロックダイアグラム	151
フレーム	
フレームフォーマットの設定	377
フレームフォーマット	
フレームフォーマットの設定	377
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 (ライタから電源供給時)	443
プログラム	
プログラムアドレス検出制御ステータスレジスタ (PACSR)	389
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)	389
プログラムアドレス検出レジスタ 0/1	
プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1)	389
プログラムカウンタ	
プログラムカウンタ (PC)	40
プログラムパッチ	
プログラムパッチ処理フロー図	394
プログラムパッチ処理例	393
プログラムパッチ処理フロー	
プログラムパッチ処理フロー図	394
プログラムパッチ処理例	
プログラムパッチ処理例	393
プログラム例	
1M/2M ビットフラッシュメモリの プログラム例	427
プロセッサステータス	
プロセッサステータス (PS)	37
ブロックダイアグラム	
16 ビット入出力タイマの ブロックダイアグラム	171
16 ビットフリーランタイムのブロック ダイアグラム	173
16 ビットリロードタイマのブロック ダイアグラム	192
8/16 ビット PPG のブロックダイアグラム	207
A/D コンバータのブロックダイアグラム	240
CAN コントローラのブロック ダイアグラム	331
DTP/ 外部割込みのブロックダイアグラム	228
ROM ミラー機能選択モジュールのブロック ダイアグラム	396
UART0 のブロックダイアグラム	261
UART1 のブロックダイアグラム	287
アウトプットコンペアのブロック ダイアグラム	179
アドレス一致検出機能のブロック ダイアグラム	388
インプットキャプチャのブロック ダイアグラム	186
ウォッチドッグタイマのブロック ダイアグラム	158
外部メモリアクセスのブロック ダイアグラム	129
シリアル I/O のブロックダイアグラム	312

タイムベースタイマのブロック ダイアグラム	154
遅延割込みのブロックダイアグラム	224
低消費電力制御回路のブロック ダイアグラム	89
時計タイマのブロックダイアグラム	164
プルアップ制御レジスタ (PUCR0 ~ PUCR3) の ブロックダイアグラム	151
ブロックダイアグラム	5

へ

変換	
変換データ保護機能	256
変換データ保護機能の使用上の注意	257
変換データ保護機能のフロー例 (EI ² OS の場合)	257
変換データ保護機能	
変換データ保護機能	256
変換データ保護機能の使用上の注意	257
変換データ保護機能のフロー例 (EI ² OS の場合)	257

ほ

ポートデータレジスタ	
ポートデータレジスタ (PDR0 ~ PDRA)	146
ポート方向レジスタ	
ポート方向レジスタ (DDR0 ~ DDRA)	148
ホールド機能	
ホールド機能	141

ま

マシクロック	
マシクロックの切換え	87
マルチレベルメッセージバッファ	
マルチレベルメッセージバッファの構成の 決定	383

み

未定義命令	
未定義命令の実行による例外発生	75

め

命令	
未定義命令の実行による例外発生	75
命令の種類	453
命令マップの構造	491
割込み抑止命令	45
割込み抑止命令とプリフィックス命令に関する 制約	45
「DIV A,Ri」, 「DIVW A,RWi」命令の 使用上の注意	47
命令一覧表	
F ² MC-16LX 命令一覧表	477
メインクロック	
メインクロック発振安定待ち時間設定	86

メッセージバッファ	
BVAL ビットによるメッセージバッファ禁止時の注意	385
メッセージバッファ	338, 365
メッセージバッファ (DLC レジスタ) の一覧	336
メッセージバッファ (ID レジスタ) の一覧	334
メッセージバッファ (x) による受信方法	381
メッセージバッファ (x) による送信方法	379
メッセージバッファ (データレジスタ) の一覧	337
メッセージバッファ制御レジスタ	338
メッセージバッファ有効レジスタ (BVALR)	349
メッセージバッファ制御レジスタ	
メッセージバッファ制御レジスタ	338
メッセージバッファ有効レジスタ	
メッセージバッファ有効レジスタ (BVALR)	349
メモリアクセスモード	
メモリアクセスモードの概要	124
メモリ空間	
CPU メモリ空間の概要	27
バスモード別メモリ空間	127
メモリ空間における多バイト長データの配置	32
メモリ空間マップ	28
メモリ空間マップ	
メモリ空間マップ	28
メモリマップ	
EEPROM メモリマップ	392

も

モード 0	
受信動作時 (モード 0, モード 1, モード 3) のフラグのセットタイミング	278
モード 1	
UART1 の応用例 (モード 1 使用時のシステム構築例)	309
受信動作時 (モード 0, モード 1, モード 3) のフラグのセットタイミング	278
モード 2	
受信動作時 (モード 2) のフラグのセットタイミング	279
モード 3	
受信動作時 (モード 0, モード 1, モード 3) のフラグのセットタイミング	278
モード端子	
モード端子	125
モードデータ	
モードデータ	126
モジュール	
アウトプットコンペア (モジュールごとに 2 チャンネル)	170
インプットキャプチャ (モジュールごとに 2 チャンネル)	170

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)	36
ユーザ電源使用時	
シリアル書込み接続例 (ユーザ電源使用時)	437
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)	441

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)	231

ら

ライター	
フラッシュマイコンプログラマとの最小限の接続例 (ライターから電源供給時)	443
ライターから電源供給時	
シリアル書込み接続例 (ライターから電源供給時)	439
ラストイベント	
ラストイベント表示レジスタ (LEIR)	344
ラストイベント表示レジスタ	
ラストイベント表示レジスタ (LEIR)	344

り

リセット	
ウォッチドッグタイマのリセット	161
フラッシュメモリの読出し / リセット状態	417
リセット解除後の動作	79
リセット入力によって初期化されないレジスタ	80
リセット要因	83
リセット要因の発生	79
リセット解除後の動作	
リセット解除後の動作	79
リセット状態	
フラッシュメモリの読出し / リセット状態	417
リセット入力	
リセット入力によって初期化されないレジスタ	80
リセット要因	
リセット要因	83
リセット要因の発生	79
リモート	
リモート要求受信レジスタ (RRTRR)	358
リモートフレーム	
データフレームおよびリモートフレームの受信に対する処理	374
リモートフレーム受信待機レジスタ (RFWTR)	353

リモートフレーム受信待機レジスタ	
リモートフレーム受信待機レジスタ	
(RFWTR)	353
リモート要求受信レジスタ	
リモート要求受信レジスタ (RRTRR)	358
リロードタイマ	
16 ビットリロードタイマ (イベントカウンタ	
機能付き) の概要	192
16 ビットリロードタイマの	
アンダフロー動作	201
16 ビットリロードタイマの端子名	193
16 ビットリロードタイマの入力端子機能	
(内部クロックモード時)	199
16 ビットリロードタイマのブロック	
ダイアグラム	192
16 ビットリロードタイマのレジスタ	194
16 ビットリロードタイマの出力端子機能	202
16 ビットリロードタイマの	
内部クロック動作	199
リロード値	
8/16 ビット PPG のリロード値とパルス幅の	
関係	217
リロードレジスタ	
16 ビットタイマレジスタ (TMR)/16 ビット	
リロードレジスタ (TMRLR)	198
リロードレジスタ (PRL, PRLH)	215

れ

レート・データレジスタ 0	
レート・データレジスタ 0(URD0)	268
レジスタ	
16 ビットタイマレジスタ (TMR)/16 ビット	
リロードレジスタ (TMRLR)	198
16 ビット入出力タイマのレジスタ	172
16 ビットリロードタイマのレジスタ	194
2 組のコンペアレジスタによる出力波形例	
(出力の初期値は 0 とする)	184
32 ビットレジスタ間接指定	29
8/16 ビット PPG のレジスタ	209
A/D コンバータのレジスタ一覧	241
A/D 制御ステータスレジスタ 0(ADCS0)	242
A/D 制御ステータスレジスタ 1(ADCS1)	245
A/D データレジスタ 0/1(ADCR0, ADCR1)	248
CAN 制御ステータスレジスタ (CSR)	339
CLK 同期モードを使用する場合の各制御	
レジスタの設定値	303
DLC レジスタ $x(x=0 \sim 15)(DLCR_x)$	368
DTP/ 外部割込み許可レジスタ (ENIR:Interrupt	
request enable register)	230
DTP/ 外部割込みのレジスタ	229
DTP/ 外部割込み要因レジスタ (EIRR:External	
interrupt request register)	230
EI ² OS ステータスレジスタ (ISCS)	70
I/O ポートのレジスタ一覧	145
I/O レジスタアドレスポインタ (IOA)	70
IDE レジスタ (IDER)	350
ID レジスタ $x(x=0 \sim 15)(IDR_x)$	366
PPG0 動作モード制御レジスタ (PPGC0)	210
PPG1 動作モード制御レジスタ (PPGC1)	212
PPG ユニット 0 クロック選択レジスタ	
(PPG01)	214

ROM ミラー機能選択レジスタ (ROMM)	397
UART0 のレジスタ	262
UART1 通信プリスケアラ制御レジスタ	
(CDCR)	296
UART1 のレジスタ一覧	288
アウトプットコンペアのコンペアレジスタ	
(OCCP)	180
アウトプットコンペアの制御ステータスレジスタ	
(OCS0/1)	181
アクセプタンスマスク選択レジスタ	
(AMSR)	361
アクセプタンスマスクレジスタ 0/1	
(AMR0/AMR1)	363
アナログ入力許可レジスタ	238
アナログ入力許可レジスタ (ADER)	152
インプットキャプチャ制御ステータスレジスタ	
(ICS01)	187
インプットキャプチャデータレジスタ	
(IPCP0/1)	187
ウォッチドッグタイマ制御レジスタ	
(WDTC)	159
ウォッチドッグタイマのレジスタ一覧	158
外部アドレス出力制御レジスタ (HACR)	133
外部メモリアクセスのレジスタ一覧	130
クロック選択レジスタ (CKSCR)	93
コモンレジスタバンクプリフィックス	
(CMR)	44
コンディショニングコードレジスタ (CCR)	37
コンペアレジスタ 0, 1 を使用したときの出力	
波形例 (出力の初期値は 0 とする)	184
自動レディ機能選択レジスタ (ARSR)	131
受信オーバランレジスタ (ROVRR)	359
受信完了レジスタ (RCR)	357
受信割込み許可レジスタ (RIER)	360
シリアル I/O のレジスタ	313
シリアルインプットデータレジスタ 0(UIDR0) と	
シリアルアウトプットデータレジスタ	
.....	267
シリアルインプットデータレジスタ 1(SIDR1)/	
シリアルアウトプットデータレジスタ 1	
(SODR1)	293
シリアルシフトデータレジスタ (SDR)	318
シリアルステータスレジスタ 0(USR0)	265
シリアルステータスレジスタ 1(SSR1)	294
シリアル制御レジスタ 1(SCR1)	291
シリアルモード制御ステータスレジスタ	
(SMCS)	314
シリアルモード制御レジスタ 0(UMC0)	263
シリアルモードレジスタ 1(SMR1)	289
スタック内へのレジスタの退避	56
全体制御レジスタ	332, 338
専用レジスタ	33
送信 RTR レジスタ (TRTRR)	352
送信解除レジスタ (TCANR)	354
送信完了レジスタ (TCR)	355
送信要求レジスタ (TREQR)	351
送信割込み許可レジスタ (TIER)	356
タイマカウンタ制御ステータスレジスタ	
(TCCS)	175
タイマカウンタデータレジスタ (TCDT)	174
タイマ制御ステータスレジスタ (TMCSR)	195
タイムベースタイマ制御レジスタ (TBTC)	
.....	155

タイムベースタイマのレジスター一覧	154
遅延割込み要因発生 / 解除レジスタ (DIRR)	225
低消費電力制御回路のレジスタ	90
低消費電力モード制御レジスタ (LPMCR)	91
低消費電力モード制御レジスタのアクセス	92
データレジスタ $x(x=0 \sim 15)(DTRx)$	369
時計タイマ制御レジスタ (WTC)	165
時計タイマのレジスタ	164
バス制御信号選択レジスタ (ECSR)	134
汎用レジスタ	34
ビットタイミングレジスタ (BTR)	347
フラッシュメモリ制御ステータスレジスタ (FMCS)	404
フラッシュメモリのレジスタ	400
プルアップ制御レジスタ (PUCR0 ~ PUCR3)	150
プルアップ制御レジスタ (PUCR0 ~ PUCR3) のブロックダイアグラム	151
プログラムアドレス検出制御ステータスレジスタ (PACSR)	389
プログラムアドレス検出レジスタ 0/1 (PADR0/PADR1)	389
ポートデータレジスタ (PDR0 ~ PDRA)	146
ポート方向レジスタ (DDR0 ~ DDRA)	148
メッセージバッファ (DLC レジスタ) の一覧	336
メッセージバッファ (ID レジスタ) の一覧	334
メッセージバッファ (データレジスタ) の一覧	337
メッセージバッファ制御レジスタ	338
メッセージバッファ有効レジスタ (BVALR)	349
要求レベル設定レジスタ (ELVR:External level register)	231
ラストイベント表示レジスタ (LEIR)	344
リセット入力によって初期化されないレジスタ	80
リモートフレーム受信待機レジスタ (RFWTR)	353
リモート要求受信レジスタ (RRTRR)	358
リロードレジスタ (PRL, PRLH)	215
レート・データレジスタ 0(URD0)	268
レジスタバンク	41
レジスタバンクポインタ (RP)	38
割込み制御レジスタ (ICR)	66
割込みレベルマスクレジスタ (ILM)	39
レジスター一覧	
A/D コンバータのレジスター一覧	241
I/O ポートのレジスター一覧	145
UART1 のレジスター一覧	288
ウォッチドッグタイマのレジスター一覧	158
外部メモリアクセスのレジスター一覧	130
タイムベースタイマのレジスター一覧	154
レジスタバンク	
レジスタバンク	41
レジスタバンクポインタ	
レジスタバンクポインタ (RP)	38
レディ機能	
自動レディ機能選択レジスタ (ARSR)	131
レディ機能	139

連続モード	
連続モード	250
連続モード時の EI ² OS の起動例	254

わ

割込み	
8/16 ビット PPG の割込み	220
DTP/ 外部割込み許可レジスタ (ENIR:Interrupt request enable register)	230
DTP/ 外部割込みの概要	228
DTP/ 外部割込みの使用上の注意	235
DTP/ 外部割込みのブロックダイアグラム	228
DTP/ 外部割込みのレジスタ	229
DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
DTP 要求と外部割込み要求の切換え	234
UART1 の割込みとフラグのセット	
タイミング	305
UART1 の割込み発生要因	304
インテリジェント I/O サービス (EI ² OS) 機能と割込み	192
外部割込みの動作	232
受信割込み許可レジスタ (RIER)	360
シリアル I/O の割込み機能	327
送信割込み許可レジスタ (TIER)	356
ソフトウェア割込みに関する注意事項	63
ソフトウェア割込みの概要	62
ソフトウェア割込みの構造	62
ソフトウェア割込みの動作	63
タイムベースタイマのインターバル割込み機能	156
多重割込み	56
遅延割込みのブロックダイアグラム	224
遅延割込み発生	226
遅延割込み要因発生 / 解除レジスタ (DIRR)	225
遅延割込み要求ロックの使用上の注意	224
時計タイマのインターバル割込み機能	167
入出力領域への書込み中のハードウェア割込み要求	55
ハードウェア割込みの概要	55
ハードウェア割込みの構造	55
ハードウェア割込みの動作	57
ハードウェア割込みの動作フロー	59
割込み処理の開始に要する時間	60
割込み制御レジスタ (ICR)	66
割込みの概要	50
割込みベクタ	53
割込み要因	51
割込み抑止命令	45
割込み抑止命令とプリフィックス命令に関する制約	45
割込み処理	
割込み処理の開始に要する時間	60
割込み制御	
割込み制御レジスタ (ICR)	66
割込み発生要因	
UART1 の割込み発生要因	304
割込みベクタ	
割込みベクタ	53

割込み要因	
DTP/ 外部割込み要因レジスタ (EIRR:External interrupt request register)	230
割込み要因	51
割込み要求	
DTP 要求と外部割込み要求の切換え	234
遅延割込み要求ロックの使用上の注意	224
入出力領域への書き込み中のハードウェア割込み 要求	55
割込み抑止命令	
割込み抑止命令	45
割込み抑止命令とプリフィックス命令に関する 制約	45
割込みレベルマスク	
割込みレベルマスクレジスタ (ILM)	39

CM44-10108-6

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

MB90540/545 Series

ハードウェアマニュアル

2008 年 7 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 ビジネス推進部
