



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-RAM<sup>TM</sup>、SRAM、Traveo<sup>TM</sup> マイクロコントローラー、業界唯一の PSoC<sup>®</sup> 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<sup>®</sup> 静電容量タッチセンシング コントローラー、Wireless BLE Bluetooth<sup>®</sup> 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。

**FR30**

32ビット・マイクロコントローラ

**MB91F127/128**

**ハードウェアマニュアル**

**FR30**

32ビット・マイクロコントローラ

**MB91F127/128**

**ハードウェアマニュアル**

**富士通株式会社**

# はじめに

## 本書の目的と対象読者

富士通半導体製品につきまして、平素より格別のご愛顧を賜り厚くお礼申し上げます。

MB91F127/128は、新しいRISCアーキテクチャのCPUをコアとした「32ビットシングルチップマイクロコントローラ FR30シリーズ」の一つとして開発された製品で、高性能なCPU処理パワーを要求される組み込み用途に最適な仕様となっています。

本書は、実際にこのMB91F127/128を使用して製品を開発される技術者を対象に、MB91F127/128の機能や動作について解説したものです。本書をご一読ください。

なお、各種命令の詳細については、『インストラクションマニュアル』をご参照ください。

## 商標

FRは、FUJITSU RISC controllerの略で、富士通株式会社の製品です。  
Embedded Algorithm TMはAdvanced Micro Devices社の商標です。

## 本書の全体構成

本書は、以下に示す17の章および付録から構成されています。

### 第1章 概要

この章では、MB91F127/128の特長、ブロックダイアグラム、機能概要など全体を知るための基本的なことからについて説明します。

### 第2章 デバイスの取扱いについて

MB91F127/128の取扱いに関して、端子処理のしかたや、電源の取扱いの注意などについて説明します。

### 第3章 CPUおよび制御部

この章では、FRシリーズのCPUコアの機能を知るために、アーキテクチャ、仕様、命令などの基本的なことについて説明します。

### 第4章 バスインタフェース

この章では、バスインタフェースの基本事項、レジスタ構成/機能、バス動作、バスタイミングおよびバス動作のプログラム例について説明します。

### 第5章 I/Oポート

この章では、I/Oポートの概要、レジスタ構成および外部端子をI/Oとして使用する条件について説明します。

### 第6章 16ビットリロードタイマ

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

### 第7章 PPGタイマ

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

### 第8章 多機能タイマ

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

### 第9章 U-TIMER

この章では、U-TIMERの概要、レジスタの構成/機能およびU-TIMERの動作について説明します。

### 第10章 外部割込み

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

### 第11章 遅延割込みモジュール

この章では、遅延割込みモジュールの概要、レジスタの構成/機能および遅延割込みモジュールの動作について説明します。

### 第12章 割込みコントローラ

この章では、割込みコントローラの概要、レジスタの構成/機能、割込みコントローラの動作およびホールドリクエスト取下げ要求機能の使用例について説明します。

### 第13章 A/Dコンバータ(逐次比較型)

この章では,A/Dコンバータの概要,レジスタの構成/機能およびA/Dコンバータの動作について説明します。

### 第14章 UART

この章では,UARTの概要,レジスタの構成/機能およびUARTの動作について説明します。

### 第15章 DMAコントローラ(DMAC)

この章では,DAMコントローラ(DMAC)の概要,レジスタの構成/機能およびDMAコントローラ(DMAC)の動作について説明します。

### 第16章 ビットサーチモジュール

この章では,ビットサーチモジュールの概要,レジスタ構成/機能およびビットサーチモジュールの動作,退避/復帰処理について説明します。

### 第17章 フラッシュメモリ

この章では,フラッシュメモリの機能や動作について説明します。

ここでは,FR-CPUからこのフラッシュメモリを使用する場合について解説します。ROMライタからこのフラッシュメモリを使用する場合の詳細につきましては,別途ROMライタの取扱説明書をご参照ください。

### 付 録

付録として,I/Oマップ,割込みベクタ,CPUステートでの端子状態を記載しています。

- ・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,ご用命の際は当社営業担当部門にご確認ください。
- ・本資料に記載された情報・回路図は,半導体デバイスの応用例として使用されており,実際に使用する機器への搭載を目的としたものではありません。また,これらの情報・回路図の使用に起因する第三者の特許権,その他の権利侵害について,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,必ず事前に当社営業担当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が,「外国為替および外国貿易法」に基づき規制されている貨物または技術に該当する場合には,本製品を輸出するに際して,同法に基づく許可が必要となります。

# 本書の読み方

## 本書の記述形式

以下に本書で使用する主要な用語に関する説明を示します。

用語	意 味
I-BUS	内部インストラクション用の16ビット幅のバスです。FRシリーズは、内部ハーバードアーキテクチャを採用しているため、命令とデータは独立したバスです。I-BUSには、バスコンバータが接続されます。
D-BUS	内部32ビット幅のデータバスです。D-BUSには、内部リソースが接続されます。
C-BUS	内部マルチプレクスバスです。スイッチを通してI-BUSとD-BUSに接続されます。C-BUSには外部インタフェースモジュールが接続されます。 外部データバスは、データと命令がマルチプレクスされます。
R-BUS	内部16ビット幅のデータバスです。R-BUSはアダプタを通してD-BUSに接続されます。R-BUSには、各種I/O、クロック発生部、割込みコントローラが接続されます。R-BUSは16ビット幅でかつアドレス、データがマルチプレクスされるため、CPUがこれらのリソースにアクセスする場合、複数サイクルの時間がかかります。
E-unit	演算実行ユニットです。
	システムクロック。R-BUSにつながる各内蔵リソースに対してクロック発生部から出力されるクロック。最速で原発振と同周期ですが、クロック発生部ギア制御レジスタ(GCR)のPCK1,0で1, 1/2, 1/4, 1/8(または 1/2, 1/4, 1/8, 1/16)に分周されたクロックとなります。
	システムクロック。R-BUS以外のバスに接続されるリソースとCPUの動作クロック。最速で原発振と同周期ですが、クロック発生部ギア制御レジスタ(GCR)のCCK1,0で1, 1/2, 1/4, 1/8(または 1/2, 1/4, 1/8, 1/16)に分周されたクロックとなります。

# 目次

第1章	概要	1
1.1	MB91F127/128の特長	2
1.2	ブロックダイヤグラム	5
1.3	外形寸法図	6
1.4	端子配列	7
1.5	端子機能一覧	8
1.6	入出力回路形式	12
第2章	デバイスの取扱いについて	15
2.1	デバイス取扱い上の注意	16
2.2	電源に関する取扱い上の注意	18
第3章	CPUおよび制御部	19
3.1	メモリ空間	20
3.2	内部アーキテクチャ	23
3.3	専用レジスタ	27
3.3.1	プログラムステータスレジスタ(PS)	30
3.4	汎用レジスタ	34
3.5	データ構造	35
3.6	ワードアライメント	36
3.7	メモリマップ	37
3.8	命令概要	39
3.8.1	遅延スロット付き分岐命令	41
3.8.2	遅延スロットなし分岐命令	43
3.9	EIT(例外, 割込み, トラップ)	44
3.9.1	EITの割込みレベル	45
3.9.2	割込み制御レジスタ(ICR)	47
3.9.3	システムスタックポインタ(SSP)	48
3.9.4	テーブルベースレジスタ(TBR)	49
3.9.5	多重EIT処理	51
3.9.6	EITの動作	53
3.10	リセットシーケンス	57



3.11	クロック発生部と制御部	58
3.11.1	リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)	60
3.11.2	スタンバイ制御レジスタ(STCR)	62
3.11.3	DMA要求抑止レジスタ(PDRR)	63
3.11.4	タイムベースタイマクリアレジスタ(CTBR)	64
3.11.5	ギア制御レジスタ(GCR)	65
3.11.6	ウォッチドッグリセット発生延期レジスタ(WPR)	68
3.11.7	PLL制御レジスタ(PCTR)	69
3.11.8	ウォッチドッグ機能	71
3.11.9	ギア機能	73
3.11.10	リセット要因保持	75
3.11.11	DMA抑止	77
3.11.12	クロックダブラ機能	79
3.11.13	PLLのクロック設定例	81
3.12	スタンバイモード(低消費電力メカニズム)	84
3.12.1	ストップ状態	86
3.12.2	スリープ状態	90
3.12.3	スタンバイモードの状態遷移	92
3.13	動作モード	93
第4章	バスインタフェース	95
4.1	バスインタフェースの概要	96
4.2	バスインタフェースのブロックダイアグラム	99
4.3	バスインタフェースのレジスタ	100
4.3.1	エリア選択レジスタ(ASR)とエリアマスクレジスタ(AMR)	101
4.3.2	エリアモードレジスタ0(AMD0)	104
4.3.3	エリアモードレジスタ1(AMD1)	106
4.3.4	エリアモードレジスタ32(AMD32)	107
4.3.5	エリアモードレジスタ4(AMD4)	108
4.3.6	エリアモードレジスタ5(AMD5)	109
4.3.7	外部端子制御レジスタ0(EPCR0)	110
4.3.8	外部端子制御レジスタ1(EPCR1)	113
4.3.9	リトルエンディアンレジスタ(LER)	114

4.4	バス動作 .....	115
4.4.1	データバス幅と制御信号との関係 .....	116
4.4.2	ビッグエンディアンのバスアクセス .....	118
4.4.3	リトルエンディアンのバスアクセス .....	124
4.4.4	ビッグエンディアンとリトルエンディアンの外部アクセスの比較 .....	128
4.5	バスタイミング .....	132
4.5.1	基本リードサイクル .....	133
4.5.2	基本ライトサイクル .....	135
4.5.3	各モードでのリードサイクル .....	137
4.5.4	各モードでのライトサイクル .....	139
4.5.5	リード/ライト混在サイクル .....	141
4.5.6	自動ウェイトサイクル .....	142
4.5.7	外部ウェイトサイクル .....	143
4.5.8	時分割入出力インタフェース .....	144
4.5.9	外部バスリクエスト .....	146
4.6	内部クロック逡倍動作(クロックダブラ).....	147
4.7	外部バスのプログラム例 .....	148
第5章	I/Oポート .....	151
5.1	I/Oポートの概要 .....	152
5.2	ポートデータレジスタ(PDR) .....	153
5.3	データ方向レジスタ(DDR) .....	154
5.4	外部端子をI/Oポートとして使用 .....	155
第6章	16ビットリロードタイマ .....	159
6.1	16ビットリロードタイマの概要 .....	160
6.2	16ビットリロードタイマのレジスタ.....	161
6.2.1	コントロールステータスレジスタ(TMCSR) .....	162
6.2.2	16ビットタイマレジスタ(TMR) .....	165
6.2.3	16ビットリロードレジスタ(TMRLR) .....	166
6.3	16ビットリロードタイマの動作 .....	167
6.4	カウンタの動作状態 .....	169
第7章	PPGタイマ .....	171
7.1	PPGタイマの概要 .....	172

7.2	PPGタイマのブロックダイアグラム.....	173
7.3	PPGタイマのレジスタ .....	174
7.3.1	コントロールステータスレジスタ(PCNH, PCNL) .....	175
7.3.2	PPG周期設定レジスタ(PCSR) .....	178
7.3.3	PPGデューティ設定レジスタ(PDUT) .....	179
7.3.4	PPGタイマレジスタ(PTMR) .....	180
7.3.5	ジェネラルコントロールレジスタ1(GCN1) .....	181
7.3.6	ジェネラルコントロールレジスタ2(GCN2) .....	183
7.4	PPG動作 .....	184
7.5	ワンショット動作 .....	185
7.6	割込み .....	186
7.7	PPG出力オール”L”とオール”H” .....	187
7.8	PPGタイマ複数チャンネルの起動 .....	188
<b>第8章</b>	<b>多機能タイマ .....</b>	<b>189</b>
8.1	多機能タイマの概要 .....	190
8.2	多機能タイマユニットのブロックダイアグラム.....	191
8.3	多機能タイマユニットのレジスタ .....	192
8.3.1	16ビットフリーランタイマのレジスタ .....	193
8.3.2	アウトプットコンペアのレジスタ .....	196
8.3.3	インプットキャプチャのレジスタ .....	199
8.4	多機能タイマユニットの動作 .....	201
8.4.1	16ビットフリーランタイマ部の動作 .....	202
8.4.2	16ビットアウトプットコンペア部の動作 .....	204
8.4.3	16ビットインプットキャプチャ部の動作 .....	206
<b>第9章</b>	<b>U-TIMER .....</b>	<b>207</b>
9.1	U-TIMERの概要 .....	208
9.2	U-TIMERのレジスタ .....	209
9.3	U-TIMERの動作 .....	212
<b>第10章</b>	<b>外部割込み .....</b>	<b>213</b>
10.1	外部割込みの概要 .....	214
10.2	外部割込みのレジスタ .....	215
10.2.1	割込み許可レジスタ(ENIR) .....	216

10.2.2	外部割込み要因レジスタ(EIRR)	217
10.2.3	外部割込み要求レベル設定レジスタ(ELVR,EHVR)	218
10.3	外部割込み動作	219
<b>第11章</b>	<b>遅延割込みモジュール</b>	<b>221</b>
11.1	遅延割込みモジュールの概要	222
11.2	遅延割込みモジュールのレジスタ	223
11.3	遅延割込みモジュールの動作	224
<b>第12章</b>	<b>割込みコントローラ</b>	<b>225</b>
12.1	割込みコントローラの概要	226
12.2	割込みコントローラのレジスタ	227
12.2.1	割込み制御レジスタ(ICR)	229
12.2.2	ホールドリクエスト取下げ要求レベル 設定レジスタ(HRCL)	230
12.3	割込みコントローラの動作	231
12.4	ホールドリクエスト取下げ要求機能(HRCR)の使用例	234
<b>第13章</b>	<b>A/Dコンバータ(逐次比較型)</b>	<b>237</b>
13.1	A/Dコンバータ(逐次比較型)の概要	238
13.2	A/Dコンバータのレジスタ	240
13.2.1	コントロールステータスレジスタ(ADCS)	241
13.2.2	データレジスタ(ADCR)	245
13.2.3	アナログ入力制御レジスタ(AIC)	247
13.3	A/Dコンバータの動作	248
13.4	変換データ保護機能	250
13.5	A/Dコンバータ使用上の注意	252
<b>第14章</b>	<b>UART</b>	<b>253</b>
14.1	UARTの概要	254
14.2	UARTのレジスタ	256
14.2.1	シリアルモードレジスタ(SMR)	257
14.2.2	シリアルコントロールレジスタ(SCR)	259
14.2.3	シリアルインプットデータレジスタ(SIDR)/ シリアルアウトプットデータレジスタ(SODR)	261
14.2.4	シリアルステータスレジスタ(SSR)	262

14.3	UARTの動作 .....	264
14.3.1	非同期(調歩同期)モード .....	265
14.3.2	CLK同期モード .....	266
14.3.3	UARTの割込み発生とフラグのセットタイミング .....	268
14.4	UARTの使用例 .....	271
14.5	ボーレートとU-TIMERのリロード値の設定例.....	273
<b>第15章</b>	<b>DMAコントローラ .....</b>	<b>275</b>
15.1	DMAコントローラの概要 .....	276
15.2	DMAコントローラのブロックダイアグラム.....	277
15.3	DMAコントローラのレジスタ .....	278
15.3.1	DMACパラメータディスクリプタポインタ(DPDP) .....	279
15.3.2	DMACコントロールステータスレジスタ(DACSR) .....	280
15.3.3	DMAC端子コントロールレジスタ(DATCR) .....	282
15.3.4	RAM上のディスクリプタ内レジスタ .....	284
15.4	DMAコントローラの転送モード .....	287
15.5	DMAコントローラのタイミング図 .....	290
15.5.1	ディスクリプタアクセス部のタイミング図 .....	291
15.5.2	データ転送部のタイミング図 .....	293
15.5.3	連続転送モードでの転送停止タイミング図 .....	294
15.5.4	転送の終了動作タイミング図 .....	296
15.6	DMAコントローラに関する注意事項.....	298
<b>第16章</b>	<b>ビットサーチモジュール .....</b>	<b>301</b>
16.1	ビットサーチモジュールの概要 .....	302
16.2	ビットサーチモジュールのレジスタ.....	303
16.3	ビットサーチモジュールの動作と退避/復帰処理.....	305
<b>第17章</b>	<b>フラッシュメモリ .....</b>	<b>307</b>
17.1	フラッシュメモリの概要 .....	308
17.2	フラッシュメモリのレジスタ .....	313
17.2.1	フラッシュメモリステータスレジスタ(FSTR) .....	314
17.2.2	フラッシュメモリウェイトレジスタ(FWTC) .....	316
17.3	フラッシュメモリのアクセスモード.....	317
17.4	自動アルゴリズム起動方法 .....	319

17.5	自動アルゴリズム実行状態 .....	323
17.6	セクタプロテクトオペレーション .....	329
付録	.....	333
A	I/Oマップ .....	334
B	割込みベクタ .....	340
C	各CPUステートにおける端子状態 .....	342
索引	.....	347



# 第1章 概要

---

この章では,MB91F127/128の特長,ブロックダイアグラム,機能概要など全体を知るための基本的なことがらについて説明します。

---

- 1.1 MB91F127/128の特長
- 1.2 ブロックダイアグラム
- 1.3 外形寸法図
- 1.4 端子配列
- 1.5 端子機能一覧
- 1.6 入出力回路形式



## 1.1 MB91F127/128の特長

MB91F127/128は、32ビットRISC CPU(FRファミリ)をコアとし、高性能/高速なCPU処理を要求される組込み制御用に各種I/Oリソースやバス制御機構を内蔵した標準シングルチップマイクロコントローラです。

MB91F127は、256Kバイトのフラッシュメモリと14KバイトのRAMを、またMB91F128は、510Kバイトのフラッシュメモリと14KバイトのRAMを内蔵しています。

ナビゲーションシステム、高性能FAX、プリンタ制御など、高性能なCPU処理パワーを要求される組込み用途に最適な仕様となっています。

### FR CPU

- 32ビットRISC, ロード/ストアアーキテクチャ, 5段パイプライン
- 動作周波数: 内部25MHz
- 汎用レジスタ: 32ビット×16本
- 16ビット固定長命令(基本命令), 1命令/1サイクル
- メモリ-メモリ間転送, ビット処理, バレルシフトなどの命令: 組込み用途に適した命令
- 関数入口/出口命令, レジスタ内容のマルチロードストア命令: 高級言語対応命令
- レジスタインタロック機能: アセンブラ記述の容易化
- 遅延スロット付き分岐命令: 分岐処理時のオーバーヘッドの低減
- 乗算器の内蔵/命令レベルでのサポート
  - 符号付き32ビット乗算: 5サイクル
  - 符号付き16ビット乗算: 3サイクル
- 割り込み(PC, PSの待避): 6サイクル, 16プライオリティレベル

### バスインタフェース

- 最大動作周波数: 内部25MHz
- 25ビットアドレスバス(32MB空間)
- 16ビットアドレス出力, 8/16ビットデータ入出力
- 基本バスサイクル: 2クロックサイクル
- 最小64Kバイト単位で設定可能なチップセレクト出力: 6本
- 自動ウェイトサイクル: 領域ごとに0~7サイクルのうちで任意に設定可能
- アドレス/データの時分割入出力インタフェースのサポート(領域1のみ)
- 未使用データ/アドレス端子は入出力ポートとして使用可能
- リトルエンディアンモードサポート(領域1~5のうち1領域選択)

### 内蔵RAM 14KB

- D-bus RAM 12KB, C-bus RAM 2KB

## リロードタイマ

- 16ビットタイマ:3チャンネル
- 内部クロック:2クロックサイクル分解能,2/8/32分周および外部クロックから選択

## その他のインタバルタイマ

- 16ビットタイマ:3チャンネル(U-TIMER)
- PPGタイマ:4チャンネル
- 16ビットOCU:4チャンネル,ICU:4チャンネル,フリーランタイマ:1チャンネル
- ウォッチドッグタイマ:1チャンネル

## 割込みコントローラ

- 外部割込み入力:通常割込み端子6本(INT0 ~ INT5)
- 内部割込み要因:UART,DMAC,A/D,リロードタイマ,UTIMER,遅延割込み,PPG,ICU,OCU
- 優先レベルをプログラマブルに設定可能(16レベル)

## A/Dコンバータ(逐次変換型)

- 8/10ビット分解能,8チャンネル
- 逐次比較変換:25MHz動作時,5.2 $\mu$ s
- サンプル&ホールド回路内蔵
- 変換モード:シングル変換/スキャン変換/リピート変換より選択
- 起動:ソフトウェア/外部トリガ/内蔵タイマより選択

## UART

- 3チャンネル
- 全二重ダブルバッファ
- データ長:7~9ビット(パリティ無し),6~8ビット(パリティ有り)
- 非同期(調歩同期),CLK同期通信の選択可能
- マルチプロセッサモード
- ボーレートジェネレータとして16ビットタイマ(U-TIMER)内蔵:任意のボーレートを発生
- 外部クロックを転送クロックとして使用可能
- エラー検出:パリティ,フレーム,オーバラン

#### DMAC(DMAコントローラ)

- 8チャンネル
- 転送要因:内蔵リソースの割込み要求
- 転送シーケンス - ステップ転送/ブロック転送/バースト転送/連続転送
- 転送データ長:8ビット/16ビット/32ビットから選択可
- 割込み要求によって一時停止が可能

#### ビットサーチモジュール

- 1ワード中のMSBからの最初の"1"/"0"の変化ビットの位置を1サイクルでサーチ

#### リセット要因

- パワーオンリセット/ウォッチドッグタイマ/ソフトウェアリセット/外部リセット

#### 低消費電力モード

- スリープ/ストップモード

#### クロック制御

- PLL回路内蔵,1逓倍,2逓倍の選択可能
- ギア機能:CPUとペリフェラルの動作クロック周波数を独立に任意に設定可能  
ギアクロックは1/1,1/2,1/4,1/8(または1/2,1/4,1/8,1/16)から選択可能  
ただし,ペリフェラルの動作は25MHzを上限とする。

#### フラッシュメモリ

- 256KBフラッシュROM(MB91F127)/510KBフラッシュROM(MB91F128):  
単一電源にてREAD/WRITE/ERASE可能

#### その他の特長

- パッケージ:LQFP-100
- CMOSテクノロジー:0.35  $\mu\text{m}$
- 電源電圧:3.3V  $\pm$  0.3V

#### シリーズ構成

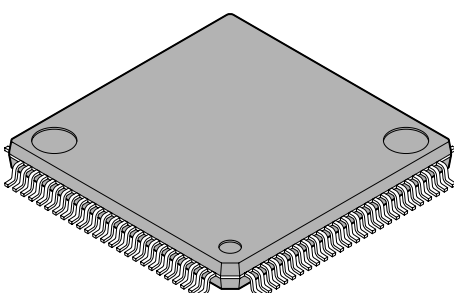
品種名	MB91FV129	MB91F127	MB91F128
概要	評価用エバチップ	量産版	量産版
フラッシュメモリ	510KB	256KB	510KB
D-bus RAM	16KB	12KB	12KB
C-bus RAM	2KB	2KB	2KB



1.3 外形寸法図

MB91F127/128には1種類のパッケージが用意されています。

LQFP-100P-M05の外形寸法図

<p>プラスチック・LQFP,100ピン</p>  <p>(FPT-100P-M05)</p>	リードピッチ	0.50 mm
	パッケージ幅 × パッケージ長さ	14.0×14.0 mm
	リード形状	ガルウィング
	封止方法	プラスチックモールド
	取付け高さ	1.70 mm MAX
	重 さ	0.65 g

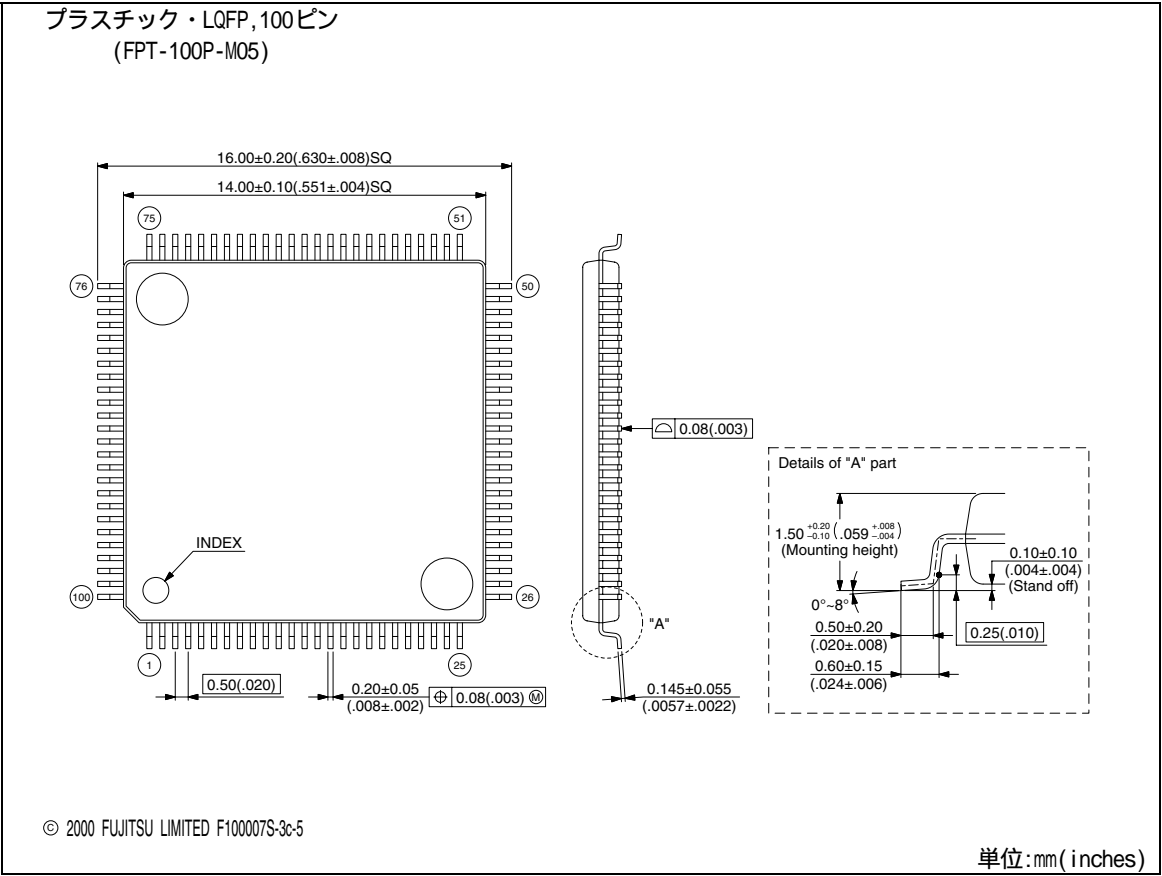


図 1.3-1 FPT-100P-M05の外形寸法図

## 1.4 端子配列

MB91F127/128の端子配列を示します。

端子配列図

図 1.4-1に, MB91F127/128の端子配列図を示します。

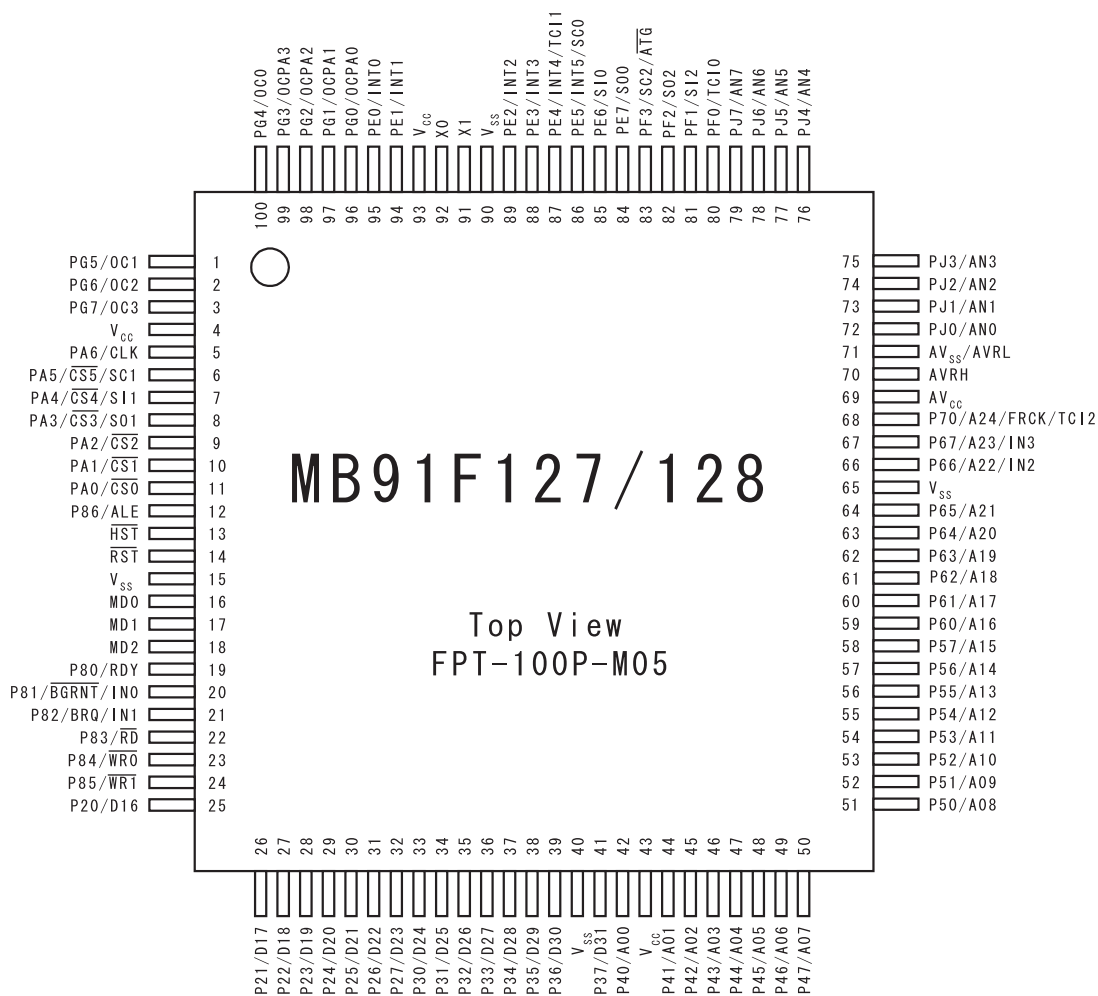


図 1.4-1 MB91F127/128の端子配列図

## 1.5 端子機能一覧

この節では,MB91F127/128の端子機能を説明します。

端子機能説明

表 1.5-1に,MB91F127/128の端子機能を示します。

表 1.5-1 端子機能表 (続く)

端子名	入出力回路形式	機 能
D16/P20 D17/P21 D18/P22 D19/P23 D20/P24 D21/P25 D22/P26 D23/P27	D	外部データバスのビット16～ビット23です。 外部バス幅が8ビットの設定のときまたはシングルチップモードのときは,汎用の入出力ポート(P20～P27)として使用できます。
D24/P30 D25/P31 D26/P32 D27/P33 D28/P34 D29/P35 D30/P36 D31/P37	D	外部データバスのビット24～ビット31です。 未使用時は汎用の入出力ポート(P30～P37)として使用できます。
A00/P40 A01/P41 A02/P42 A03/P43 A04/P44 A05/P45 A06/P46 A07/P47 A08/P50 A09/P51 A10/P52 A11/P53 A12/P54 A13/P55 A14/P56 A15/P57	D	外部アドレスバスのビット00～ビット15です。 アドレスバスとして使用しない場合は,汎用の入出力ポート(P40～P47, P50～P57)として使用できます。
A16/P60 A17/P61 A18/P62 A19/P63 A20/P64 A21/P65 A22/P66/IN2 A23/P67/IN3	D	外部アドレスバスのビット16～ビット23です。 アドレスバスとして使用しない場合は,汎用の入出力ポート(P60～P67)として使用できます。 [IN2, IN3] インプットキャプチャの入力端子です。この機能はインプットキャプチャが入力動作のときに有効になります。

表 1.5-1 端子機能表 (続く)

端子名	入出力回路形式	機 能			
A24/P70/FRCK/TC12	D	<p>外部アドレスバスのビット24です。</p> <p>[P70] A24,FRCK,TC12未使用時は汎用の入出力ポートとして使用できます。</p> <p>[FRCK] フリーランタイムの外部クロック入力です。この機能はフリーランタイムの外部クロック入力を使用するときに有効になります。</p> <p>[TC12] タイマ2の外部クロック入力です。この機能はタイマ2の外部クロック入力を使用するときに有効となります。</p>			
RDY/P80	D	外部レディ入力です。実行中のバスサイクルが完了しないときに "0" を入力します。未使用時は汎用の入出力ポートとして使用できます。			
BGRNT /P81/IN0	D	<p>外部バス開放受付出力です。外部バスを開放したときに "L" を出力します。未使用時は汎用の入出力ポートとして使用できます。</p> <p>[IN0] インプットキャプチャの入力です。この機能はインプットキャプチャが入力動作のときに有効になります。</p>			
BRQ/P82/IN1	D	<p>外部バス開放要求入力です。外部バスを開放してほしいときに "1" を入力します。未使用時は汎用の入出力ポートとして使用できます。</p> <p>[IN1] インプットキャプチャの入力です。この機能はインプットキャプチャが入力動作のときに有効になります。</p>			
RD /P83	D	外部バスリードストロープです。未使用時は汎用の入出力ポートとして使用できます。			
WR0 /P84	D	外部バスライトストロープです。各制御信号とデータバスのバイト位置は次のような関係になっています。			
WR1 /P85	D		16ビットバス幅	8ビットバス幅	シングルチップモード
		D31 ~ D24	WR0	WR0	(ポート可)
		D23 ~ D16	WR1	(ポート可)	(ポート可)
		<p>注意事項:</p> <p>WR1 はリセット中はHi-zになっています。16ビットバス幅で使用する場合は外部にプルアップ抵抗を付けてください。</p> <p>[P84またはP85] WR0, WR1 未使用時は汎用の入出力ポートとして使用できます。</p>			
CS0 /PA0 CS1 /PA1 CS2 /PA2	D	<p>チップセレクト0,1,2出力です。(Low active)</p> <p>[PA0,1,2] チップセレクトを使用しないときは汎用の入出力ポートとして使用できます。</p>			
CS3 /PA3/S01 CS4 /PA4/S11 CS5 /PA5/SC1	D	<p>チップセレクト3,4,5出力です。(Low active)</p> <p>[PA3,4,5] チップセレクト,UARTのch1を使用しないときは汎用の入出力ポートとして使用できます。</p> <p>[S01,S11,SC1] UART1のデータ出力,データ入力,クロックの端子です。UART1が動作を許可しているときに有効となります。</p>			
CLK/PA6	D	<p>システムクロック出力です。外部バス動作周波数と同じクロックを出力します。</p> <p>[PA6] 未使用時は汎用の入出力ポートとして使用できます。</p>			
OCPA0/PG0 OCPA1/PG1 OCPA2/PG2 OCPA3/PG3 OC0/PG4 OC1/PG5 OC2/PG6 OC3/PG7	D	<p>[OCPA0 ~ 3] PPGタイマの出力です。この機能はPPGタイマの出力指定が許可のときに有効となります。</p> <p>[OC0 ~ 3] アウトプットコンペアの出力です。この機能はアウトプットコンペアの出力指定が許可のときに有効となります。</p> <p>[PG0-7] 未使用時は汎用の入出力ポートとして使用できます。</p>			



表 1.5-1 端子機能表 (続く)

端子名	入出力回路形式	機 能
MD0 MD1 MD2	B	モード端子0~2です。これらの端子により,MCUの基本動作モードを設定します。 $V_{CC}$ または $V_{SS}$ に直接接続して使用してください。
X0 X1	A	クロック(発振)入力です。 クロック(発振)出力です。
$\overline{RST}$	C	外部リセット入力です。
HST	C	ハードウェアスタンバイ入力です。
P86/ALE	D	[ALE] アドレスラッチ信号出力です。この機能は,EPCRのALE出力指定が許可のときに有効となります。
INT0/PE0 INT1/PE1 INT2/PE2 INT3/PE3	D	[INT0,1,2,3] 外部割込み要求入力です。対応する外部割込みを許可している間はこの入力を随時使用していますので,意図的に行う以外は,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。 [PE0,1,2,3] 汎用の入出力ポートです。
INT4/PE4/TC11 INT5/PE5/SC0	D	[INT4,5] 外部割込み要求入力です。対応する外部割込みを許可している間はこの入力を随時使用していますので,意図的に行う以外は,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。 [TC11] タイマ1の外部クロック入力です。 [SC0] UART0のクロック入力です。 [PE4,5] 汎用の入出力ポートです。
SI0/PE6	D	[SI0] UART0のデータ入力です。この機能はUART0のデータ入力指定が許可のときに有効となります。 [PE6] 汎用の入出力ポートです。
SO0/PE7	D	[SO0] UART0のデータ出力です。この機能はUART0のデータ出力指定が許可のときに有効となります。 [PE7] 汎用の入出力ポートです。
PF0/TC10	D	[TC10] タイマ0の外部クロック入力です。 [PF0] 汎用の入出力ポートです。
SI2/PF1	D	[SI2] UART2のデータ入力です。この機能はUART2のデータ入力指定が許可のときに有効となります。 [PF1] 汎用の入出力ポートです。
SO2/PF2	D	[SO2] UART2のデータ出力です。この機能はUART2のデータ出力指定が許可のときに有効となります。 [PF2] 汎用の入出力ポートです。この機能はUART2のデータ出力指定が禁止のときに有効となります。
SC2/PF3/ $\overline{ATG}$	D	[SC2] UART2のクロック入力です。 [ $\overline{ATG}$ ] A/Dコンバータの外部トリガ入力です。それぞれの機能を選択している間は,この入力を随時使用していますので,意図的に行う以外は,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。 [PF3] 汎用の入出力ポートです。
AN0/PJ0 AN1/PJ1 AN2/PJ2 AN3/PJ3 AN4/PJ4 AN5/PJ5 AN6/PJ6 AN7/PJ7	E	[AN0~7] A/Dコンバータアナログ入力です。この機能はA1Cレジスタの指定がアナログ入力のとき有効となります。 [PJ0~7] 汎用入出力ポートです。
$AV_{CC}$	-	A/Dコンバータの $V_{CC}$ 電源です。
AVRH	-	A/Dコンバータの基準電圧です(高電位側)。この端子の投入/切断は必ず $V_{CC}$ にAVRH以上の電位が印加してある状態で行ってください。

表 1.5-1 端子機能表 (続き)

端子名	入出力回路形式	機 能
AV <sub>SS</sub> /AVRL	-	A/Dコンバータ V <sub>SS</sub> 電源および基準電圧(低電位側)です。
V <sub>CC</sub>	-	デジタル回路の電源です。必ず全端子とも電源を接続して使用してください。
V <sub>SS</sub>		デジタル回路の接地レベルです。

## &lt; 注意事項 &gt;

.....  
 上記の大部分の端子は,XXXX/PXXといったように,I/Oポートとリソースの入出力がマルチプレクスされます。これら端子でポートとリソースの出力同士が競合した場合,リソースが優先されます。  
 .....

# 1.6 入出力回路形式

この節では,入出力回路形式を示します。

入出力回路形式

表 1.6-1 入出力回路形式(続く)

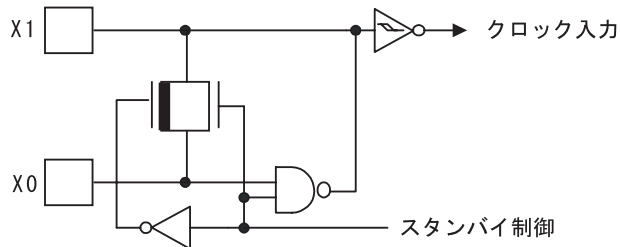
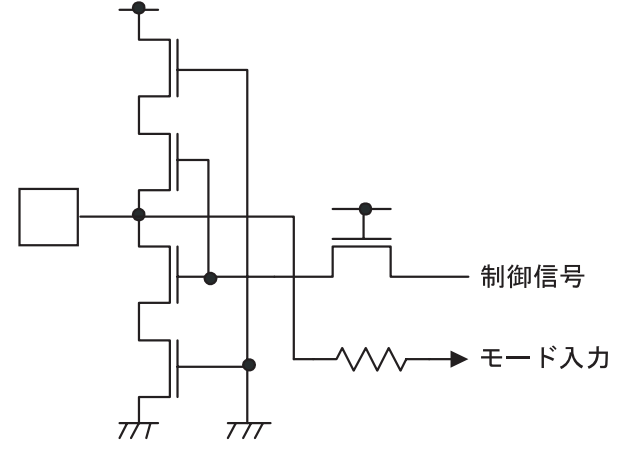
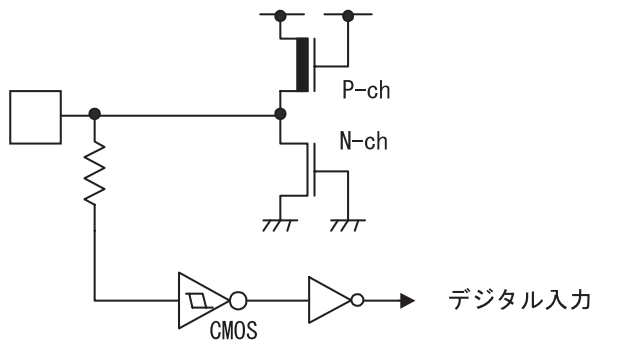
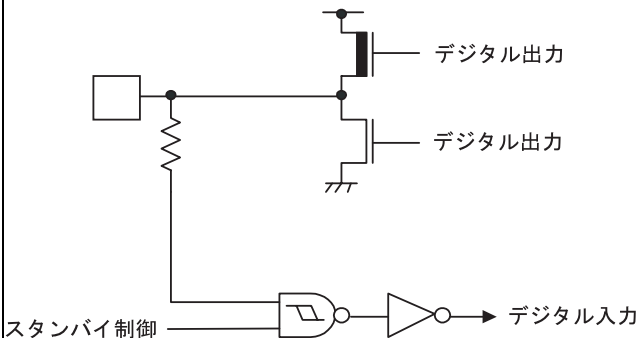
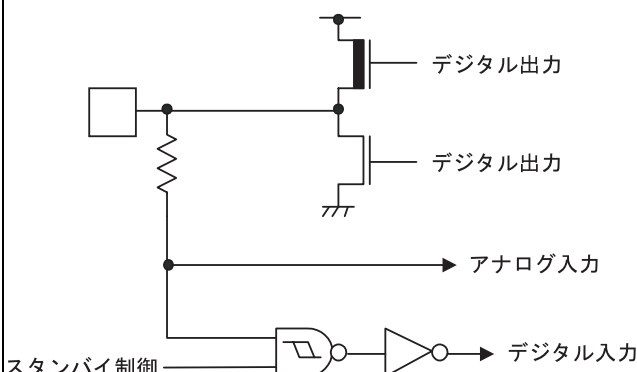
分類	回路形式	備考
A		<ul style="list-style-type: none"><li>・ 25MHz用</li><li>・ 発振帰還抵抗: 約1M</li><li>・ スタンバイ制御あり</li></ul>
B		<ul style="list-style-type: none"><li>・ CMOSレベル入力</li><li>・ フラッシュテスト用の高電圧制御あり</li></ul>
C		<ul style="list-style-type: none"><li>・ CMOSレベルヒステリシス入力</li><li>・ スタンバイ制御なし</li></ul>

表 1.6-1 入出力回路形式(続き)

分類	回路形式	備考
D		<ul style="list-style-type: none"><li>・CMOSレベル出力</li><li>・CMOSレベルヒステリシス入力</li><li>・スタンバイ制御あり</li></ul>
E		<ul style="list-style-type: none"><li>・スタンバイ制御あり</li><li>・CMOSレベル出力</li><li>・CMOSレベルヒステリシス入力</li><li>・アナログ入力</li></ul>



## 第2章 デバイスの取扱いについて

---

この章では,MB91F127/128の取扱い上の注意について説明します。

---

- 2.1 デバイス取扱い上の注意
- 2.2 電源に関する取扱い上の注意

## 2.1 デバイス取扱い上の注意

---

ここでは、ラッチアップ防止および端子処理について説明します。

---

### ラッチアップ防止のために

CMOS ICでは入力端子や出力端子に $V_{CC}$ より高い電圧や $V_{SS}$ より低い電圧を印加した場合または $V_{CC} \sim V_{SS}$ 間に定格を超える電圧を印加した場合に、ラッチアップ現象を生じることがあります。ラッチアップが生じると電源電流が激増し、素子の熱破壊に至ることがありますので、使用に際しては最大定格を超えることのないよう十分に注意してください。

また、アナログ端子はデジタル電源を超えないよう十分に注意してください。

### 未使用入力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、誤動作の原因となることがありますので、プルアップまたはプルダウンなどの処理をしてください。

### 電源端子の接続( $V_{CC}$ , $V_{SS}$ )について

$V_{CC}$ ,  $V_{SS}$ 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デバイス内部で同電位にすべきもの同士を接続してありますが、不要輻射の低減、グラウンドレベルの上昇によるストロブ信号の誤動作の防止/総電力電流規格を遵守などのために、必ずそれらすべてを外部で電源およびグラウンドに接続してください。また、電流供給源からできるかぎり低インピーダンスでこのデバイスの $V_{CC}$ ,  $V_{SS}$ に接続するような配慮をお願いします。

### 水晶発振回路について

X0, X1端子の近辺のノイズは、このデバイスを誤動作のもととなります。X0とX1および水晶発振子(またはセラミック発振子)さらにグラウンドへのバイパスコンデンサはできるかぎり近くに配置するように、また、その配線においてはほかの配線とできるかぎり交差しないようにプリント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板アートワークはより安定した動作ができますので、設計の際に考慮してください。

### NC端子の処理について

NC端子は、必ず開放にして使用してください。

### モード端子(MD0 ~ MD2)について

これらの端子は、 $V_{CC}$ または $V_{SS}$ に直接つないで使用してください。ノイズにより誤作動してしまうことを防ぐために、プリント板上の各モード端子と $V_{CC}$ または $V_{SS}$ 間のパターン長をできる限り短くし、これらを低インピーダンスで接続するようにしてください。

### 外部リセット入力

$\overline{RST}$  端子に“L”レベルを入力し、内部が確実にリセット状態となるためには、 $\overline{RST}$  端子の“L”レベル入力が最低5マシサイクル必要です。

## 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際には、原則としてX0端子、またX1端子にはX0と逆相のクロックを同時に供給してください。ただし、この場合にはSTOPモード(発振停止モード)は使用しないでください。(STOP時X1端子が“H”出力で停止するため)

また、12.5MHz以下ではX0端子のみの供給で使うことができます。

図 2.1-1, 図 2.1-2に、外部クロック使用方法例について示します。

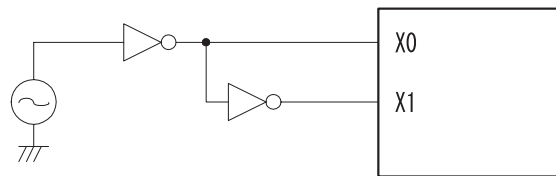


図 2.1-1 外部クロック使用例(通常)

## &lt; 注意事項 &gt;

STOPモード(発振停止モード)は使用できません。

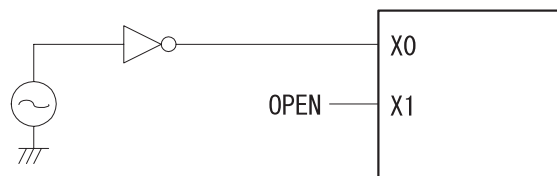


図 2.1-2 外部クロック使用例(12.5MHz以下の場合)は可能)

## スリープ/ストップ状態からの復帰

C-Bus RAM内のプログラムからスリープ/ストップ状態に入り、その状態から復帰する際には割込みによる復帰ではなく、リセットによる復帰をしてください。



## 2.2 電源に関する取扱い上の注意

---

電源投入時の端子の扱いや処理など、電源に関する注意事項について説明します。

---

### 電源投入時について

電源投入時には、必ず  $\overline{\text{RST}}$  端子を”L”レベル状態から開始し、電源が  $V_{CC}$  レベルになってから、最低でも内部動作クロックの5サイクル分の時間を確保してから”H”レベルにしてください。

### 電源投入時の端子状態について

電源投入時の端子の状態は不定です。電源を投入し発振が開始し、回路の初期化が行われます。

### 電源投入時の原振入力について

電源投入時は、必ず発振安定待ち解除されるまでの間クロックを入力してください。

### パワーオンリセットの初期化について

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には、電源の再投入によるパワーオンリセットを行ってください。

## 第3章 CPUおよび制御部

---

この章では,FRファミリの機能を知るために,アーキテクチャ,仕様,命令などの基本的なことがらについて説明します。

---

- 3.1 メモリ空間
- 3.2 内部アーキテクチャ
- 3.3 専用レジスタ
- 3.4 汎用レジスタ
- 3.5 データ構造
- 3.6 ワードアライメント
- 3.7 メモリマップ
- 3.8 命令概要
- 3.9 EIT(例外,割込み,トラップ)
- 3.10 リセットシーケンス
- 3.11 クロック発生部と制御部
- 3.12 スタンバイモード(低消費電力メカニズム)
- 3.13 動作モード

## 3.1 メモリ空間

---

FRファミリの論理アドレス空間は4Gバイト( $2^{32}$ 番地)あり,CPUはリニアにアクセスを行います。

---

### ダイレクトアドレッシング領域

アドレス空間の下記の領域は,I/O用に使用されます。

この領域をダイレクトアドレッシング領域と呼び,命令中で直接オペランドのアドレスを指定できます。

ダイレクトアドレッシング領域は,アクセスするデータのサイズにより,以下のように異なります。

- バイトデータアクセス :0 ~ 0FF<sub>H</sub>
- ハーフワードデータアクセス:0 ~ 1FF<sub>H</sub>
- ワードデータアクセス :0 ~ 3FF<sub>H</sub>

メモリマップ

図 3.1-1に,MB91F127のメモリ空間を示します。

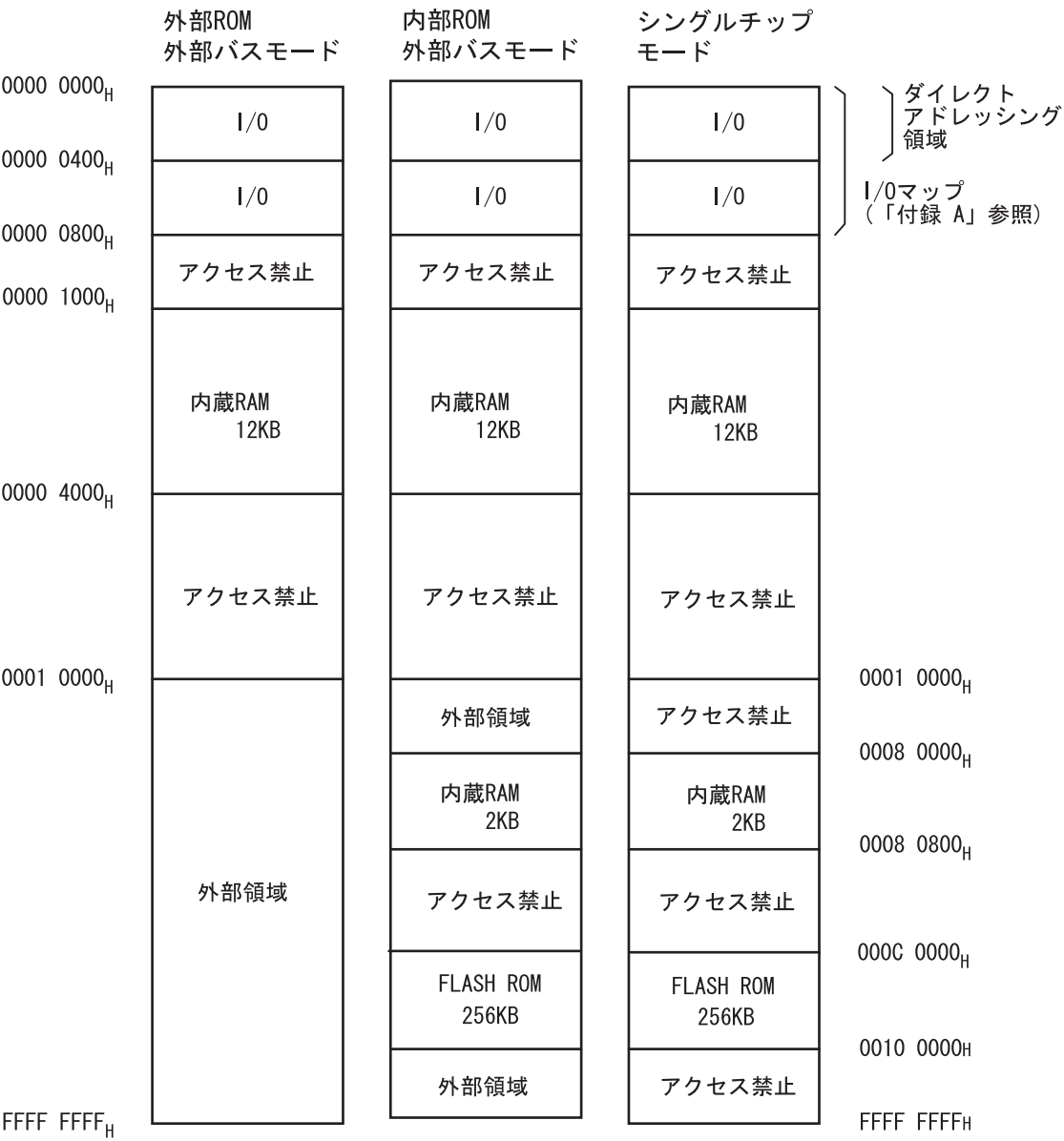


図 3.1-1 MB91F127のメモリマップ

< 注意事項 >

.....  
シングルチップモードでは外部領域へのアクセスはできません。外部領域へアクセスを行う場合は,モードレジスタで内部ROM外部バスモードを選択してください。  
.....

図 3.1-2に,MB91F128のメモリ空間を示します。

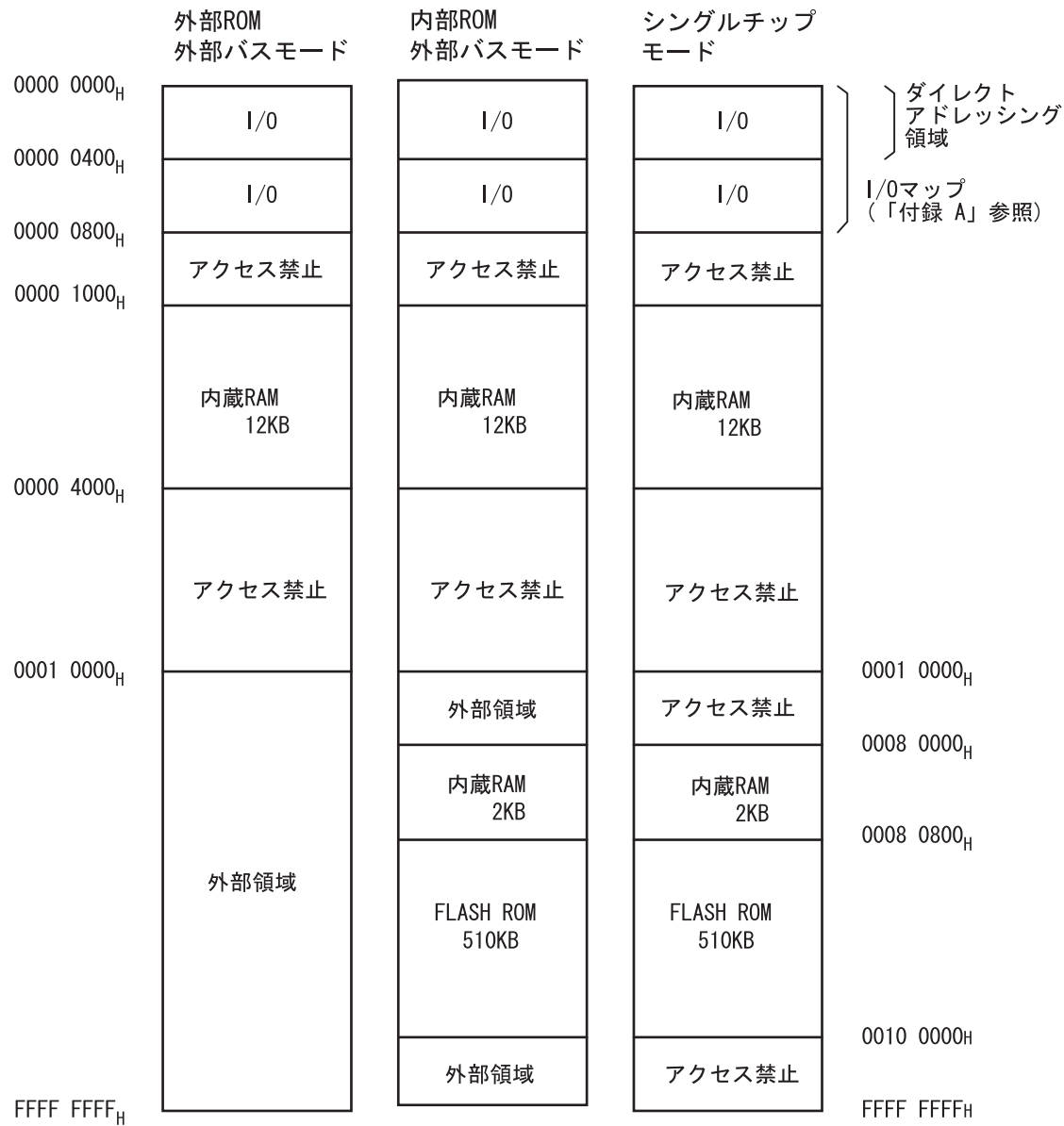


図 3.1-2 MB91F128のメモリマップ

< 注意事項 >

シングルチップモードでは外部領域へのアクセスはできません。外部領域へアクセスを行う場合は,モードレジスタで内部ROM外部バスモードを選択してください。

## 3.2 内部アーキテクチャ

FRファミリCPUは、RISCアーキテクチャを採用すると同時に、組み込み用途に向けた高機能命令を導入した高性能コアです。

### 特長

RISCアーキテクチャの採用

- 基本命令:1命令1サイクル

32ビットアーキテクチャ

- 汎用レジスタ:32ビット×16本

4GBのリニアなメモリ空間

乗算器の搭載

- 32ビット×32ビット乗算:5サイクル
- 16ビット×16ビット乗算:3サイクル

割込み処理機能の強化

- 高速応答速度:6サイクル
- 多重割込みのサポート
- レベルマスク機能:16レベル

I/O操作命令の強化

- メモリ-メモリ転送命令
- ビット処理命令

高いコード効率

- 基本命令語長:16ビット

低消費電力

- スリープモード,ストップモード

### 内部アーキテクチャ

FRファミリのCPUは、命令バスとデータバスが独立したハーバードアーキテクチャ構造を採用しています。32ビット 16ビットバスコンバータはデータバス(D-BUS)に接続され、CPUと周辺リソースとのインタフェースを実現します。ハーバード プリンストンバスコンバータはI-BUS、D-BUS双方に接続され、CPUとバスコントローラとのインタフェースを実現します。

図 3.2-1に、内部アーキテクチャを示します。

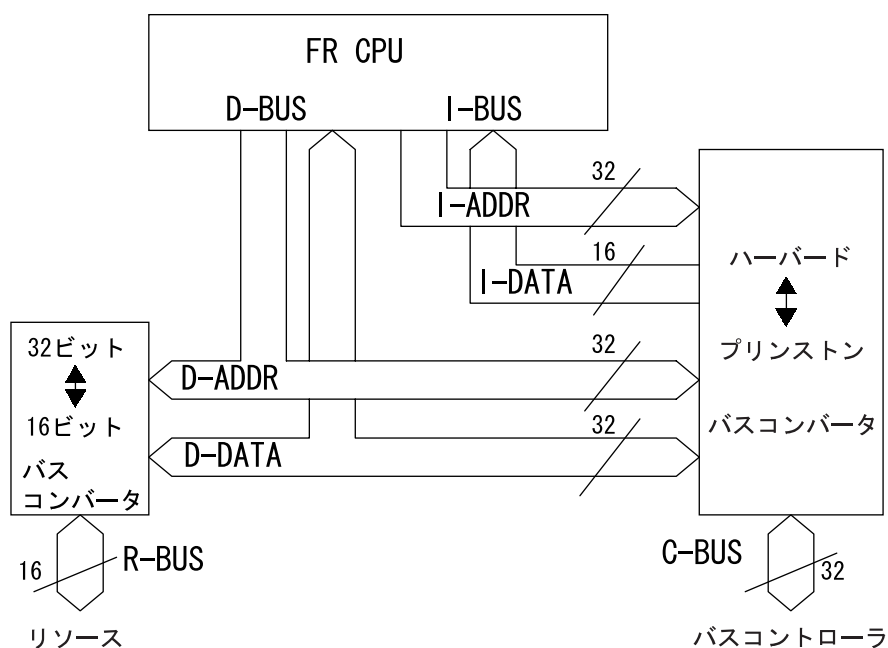


図 3.2-1 内部アーキテクチャ

## CPU

CPUは、32ビットRISCのFRアーキテクチャをコンパクトにインプリメントしたものです。

1サイクルあたり1命令を実行するため、5段階の命令パイプライン方式を採用しています。パイプラインは、以下のステージから構成されています。

- 命令フェッチ (IF) : 命令アドレスを出力し、命令をフェッチします。
- 命令デコード (ID) : フェッチした命令をデコードします。レジスタの読出しも行います。
- 実行 (EX) : 演算を実行します。
- メモリアクセス (MA) : メモリに対するロードまたはストアのアクセスを行います。
- ライトバック (WB) : 演算結果(またはロードされたメモリデータ)をレジスタに書き込みます。

図 3.2-2に、命令パイプラインの構成を示します。

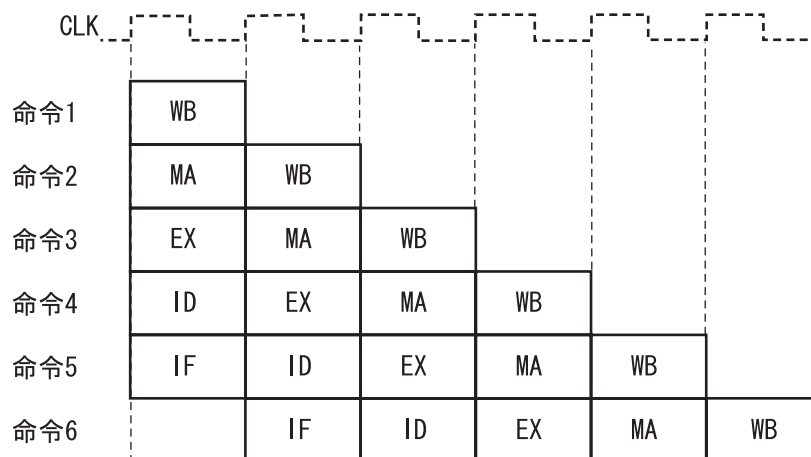


図 3.2-2 命令パイプライン

命令は、順不同で実行されることはありません。すなわち、命令Aが命令Bの前にパイプラインに入ると、命令Aは必ず命令Bの前に書込みバックステージに達します。

命令の実行は、原則として1サイクルあたり1命令の速度で行われます。ただし、メモリウェイトを伴ったロードストア命令、遅延スロットをもたない分岐命令および複数サイクル命令では、命令の実行に複数のサイクルが必要となります。また、命令の供給が遅い場合も命令の実行速度が低下します。

## 32ビット 16ビットバスコンバータ

32ビット幅で高速アクセスされるD-BUSと、16ビット幅でアクセスされるR-BUSとのインタフェースを行い、CPUから内蔵周辺回路へのデータアクセスを実現します。

CPUから32ビット幅のアクセスがあった場合、このバスコンバータがそれを2回の16ビット幅アクセスに変換してR-BUSへアクセスします。内蔵周辺回路の一部には、アクセス幅に関して制限のあるものがあります。



### ハーバード プリンストンバスコンバータ

CPUの命令アクセスとデータアクセスの整合をとり、外部バスとのスムーズなインタフェースを実現します。

CPUは、命令バスとデータバスが独立したハーバードアーキテクチャ構造です。一方、外部バスの制御を行うバスコントローラは、単一バスのプリンストンアーキテクチャ構造です。このバスコンバータは、CPUの命令アクセスとデータアクセスに優先順位をつけ、バスコントローラへのアクセスを制御します。この働きにより、外部へのバスアクセス順位が常に最適化されたものになります。また、CPUのバス待ち時間をなくすための2ワードの書込みバッファと、命令フェッチのための1ワードのプリフェッチバッファを持っています。

### 3.3 専用レジスタ

専用レジスタは、特定の目的のために使用します。プログラムカウンタ(PC)、プログラムステータス(PS)、テーブルベースレジスタ(TBR)、リターンポインタ(RP)、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)および乗除算結果レジスタ(MDH/MDL)が用意されています。

専用レジスタ一覧

図 3.3-1に、専用レジスタ一覧を示します。

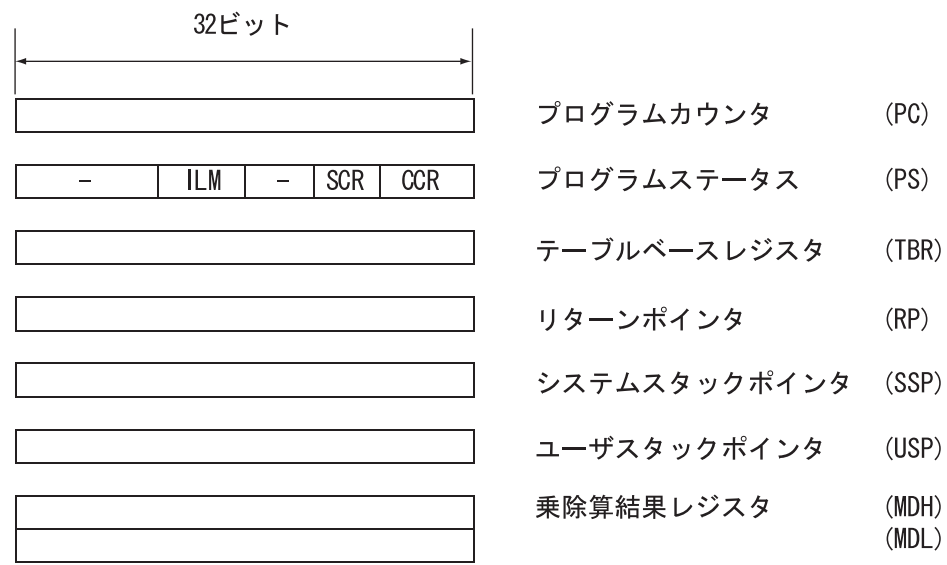


図 3.3-1 専用レジスタ一覧

プログラムカウンタ(PC)

プログラムカウンタ(PC:Program Counter)の機能について説明します。  
プログラムカウンタ(PC)は、32ビットから構成されています。  
図 3.3-2に、プログラムカウンタ(PC)のビット構成を示します。



図 3.3-2 プログラムカウンタ(PC)のビット構成

プログラムカウンタで、実行している命令のアドレスを示しています。  
命令の実行を伴うPCの更新時に、ビット0が”1”になる可能性があるのは、分岐先アドレスとして奇数番地を指定した場合だけです。ただし、その場合でもビット0は無効であり、命令は2の倍数のアドレスに置く必要があります。  
リセットによる初期値は不定です。

### プログラムステータス(PS)

プログラムステータスを保持するレジスタで,CCR,SCR,ILMの三つのパートに分かれています。詳細については「3.3.1 プログラムステータスレジスタ(PS)」を参照してください。

未定義のビットはすべて予約ビットです。読出し時,常に"0"が読み出されます。書込みは無効です。

### テーブルベースレジスタ(TBR)

テーブルベースレジスタ(TBR:Table Base Register)の機能について説明します。テーブルベースレジスタ(TBR)は,32ビットから構成されています。

図 3.3-3に,テーブルベースレジスタ(TBR)のビット構成を示します。

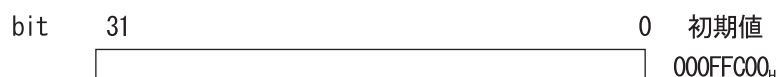


図 3.3-3 テーブルベースレジスタ(TBR)のビット構成

テーブルベースレジスタで,EIT処理の際に使用されるベクタテーブルの先頭アドレスを保持します。

リセットによる初期値は,000FFC00<sub>H</sub>です。

### リターンポインタ(RP)

リターンポインタ(RP:Return Pointer)の機能について説明します。

リターンポインタ(RP)は,32ビットから構成されています。

図 3.3-4に,リターンポインタ(RP)のビット構成を示します。



図 3.3-4 リターンポインタ(RP)のビット構成

リターンポインタで,サブルーチンから復帰するアドレスを保持します。

CALL命令実行時,PCの値がこのRPに転送されます。

RET命令実行時,RPの内容がPCに転送されます。

リセットによる初期値は不定です。

### システムスタックポインタ(SSP)

システムスタックポインタ(SSP:System Stack Pointer)の機能について説明します。

システムスタックポインタ(SSP)は,32ビットから構成されています。

図 3.3-5に,システムスタックポインタ(SSP)のビット構成を示します。



図 3.3-5 システムスタックポインタ(SSP)のビット構成

SSPは,システムスタックポインタです。

Sフラグが"0"のとき,R15として機能します。

SSPを明示的に指定することも可能です。

また,EIT発生時に,PSとPCを退避するスタックを指定するスタックポインタとしても使用されます。

リセットによる初期値は00000000<sub>H</sub>です。

## ユーザスタックポインタ(USP)

ユーザスタックポインタ(USP:User Stack Pointer)の機能について説明します。

ユーザスタックポインタ(USP)は、32ビットから構成されています。

図 3.3-6に、ユーザスタックポインタ(USP)のビット構成を示します。



図 3.3-6 ユーザスタックポインタ(USP)のビット構成

USPは、ユーザスタックポインタです。

Sフラグが"1"のとき、R15として機能します。

USPを明示的に指定することも可能です。

リセットによる初期値は不定です。

RETI命令で使用することはできません。

## 乗除算結果レジスタ(MDH/MDL)

乗除算結果レジスタ(MDH/MDL: Multiply & Divide register)の機能について説明します。

乗除算結果レジスタ(MDH/MDL)は、32ビットから構成されています。

図 3.3-7に、乗除算結果レジスタ(MDH/MDL)のビット構成を示します。

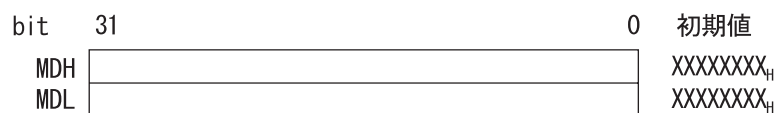


図 3.3-7 乗除算結果格納レジスタ(MDH/MDL)

乗除算用のレジスタで、MDHとMDLがあります。それぞれ32ビット長です。

リセットによる初期値は不定です。

## 乗算実行時の機能

32ビット×32ビットの乗算のとき、64ビット長の演算結果は、以下の配置で乗除算結果レジスタに格納されます。

- MDH: 上位32ビット
- MDL: 下位32ビット

16ビット×16ビットの乗算の場合は、以下のように結果が乗除算結果レジスタに格納されます。

- MDH: 不定
- MDL: 結果32ビット

## 除算実行時の機能

計算開始時、被除数をMDLに格納します。

DIVOS/DIVOU, DIV1, DIV2, DIV3, DIV4Sの各命令の実行により除算を計算すると、以下のように結果がMDLとMDHに格納されます。

- MDH: 剰余
- MDL: 商

### 3.3.1 プログラムステータスレジスタ(PS)

プログラムステータスレジスタ(PS:Program Status)は、プログラムステータスを保持するレジスタで、ILM,SCRおよびCCRの三つのパートに分かれています。未定義のビットはすべて予約ビットです。読出し時、常に”0”が読み出されます。書込みは無効です。

#### プログラムステータスレジスタ(PS)

##### プログラムステータスレジスタ(PS)

プログラムステータスレジスタ(PS)は、コンディションコードレジスタ(CCR)、システムコンディションコードレジスタ(SCR)および割込みレベルマスクレジスタ(ILM)から構成されています。

図 3.3-8に、プログラムステータスレジスタのレジスタ構成を示します。

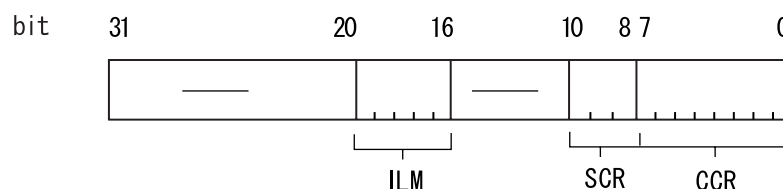


図 3.3-8 プログラムステータスレジスタ(PS)のレジスタ構成

##### コンディションコードレジスタ(CCR)

図 3.3-9に、コンディションコードレジスタ(CCR:Condition Code Register)の構成を示します。

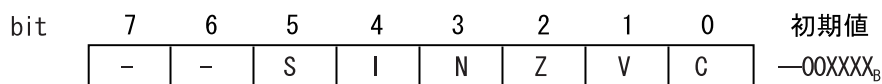


図 3.3-9 コンディションコードレジスタ(CCR)のレジスタ構成

以下に、各ビットの機能を説明します。

#### 【ビット5】S:スタックフラグ

このビットは、R15として使用させるスタックポインタを指定します。

S	スタックフラグ(S)の機能
0	システムスタックポインタ(SSP)がR15として使用されます。 EIT発生時、自動的に”0”となります。 (ただし、スタックに退避される値はクリアされる前の値です。)
1	ユーザスタックポインタ(USP)がR15として使用されます。

このビットは、リセットにより”0”にクリアされます。

RETI命令実行時は”0”にセットしてください。

## 【ビット4】I: 割込み許可フラグ

このビットは、ユーザ割込み要求の許可/禁止を制御します。

I	割込み許可フラグ(I)の機能
0	ユーザ割込禁止。 INT命令実行時,"0"にクリアされます。 (ただし,スタック退避させる値はクリアする前の値です。)
1	ユーザ割込許可。 ユーザ割込要求のマスク処理は,ILMの保持する値により制御されます。

このビットは、リセットにより"0"にクリアされます。

## 【ビット3】N: ネガティブフラグ

このビットは、演算結果を2の補数で表現された整数とみなしたときの符号を示します。

N	ネガティブフラグ(N)の機能
0	演算結果が正の値であったことを示します。
1	演算結果が負の値であったことを示します。

このビットは、リセットによる初期状態は不定です。

## 【ビット2】Z: ゼロフラグ

このビットは、演算結果が0であったかどうかを示します。

Z	ゼロフラグ(Z)の機能
0	演算結果が0以外の値であったことを示します。
1	演算結果が0であったことを示します。

このビットは、リセットによる初期状態は不定です。

## 【ビット1】V: オーバフローフラグ

このビットは、演算に用いたオペランドを2の補数で表現される整数であるとみなし、演算の結果、オーバフローが生じたかどうかを示します。

V	オーバフローフラグ(V)の機能
0	演算の結果、オーバフローは生じていないことを示します。
1	演算の結果、オーバフローが生じたことを示します。

このビットは、リセットによる初期状態は不定です。

## 【ビット0】C: キャリフラグ

このビットは、演算により最上位ビットからのキャリまたはボローが発生したかどうかを示します。

C	キャリフラグ(C)の機能
0	キャリもボローも発生していないことを示します。
1	キャリまたはボローが発生したことを示します。

このビットは、リセットによる初期状態は不定です。

## システムコンディションコードレジスタ(SCR)

図 3.3-10に、システムコンディションコードレジスタ(SCR: System Condition code Register)の構成を示します。。

bit	10	9	8	初期値
	D1	D0	T	XX0 <sub>B</sub>

図 3.3-10 システムコンディションコードレジスタ(SCR)のレジスタ構成

以下に、システムコンディションコードレジスタ(SCR)の各ビットの機能を説明します。

## 【ビット10,9】D1,D0: ステップ除算用フラグ

これらのビットは、ステップ除算実行時の中間データを保持します。

除算処理の実行途中は、変更してはいけません。

ステップ除算実行途中にほかの処理を行う場合は、プログラムステータスレジスタ(PS)の値を退避/復帰することによりステップ除算の再開が保証されます。

このビットは、リセットによる初期状態は不定です。

DIVOS命令の実行により、被除数と除数を参照して設定されます。

DIVOU命令の実行により、強制的にクリアされます。

## 【ビット8】T: ステップトレーストラップフラグ

このビットは、ステップトレーストラップを有効にするかどうかを指定するフラグです。

T	ステップトレーストラップ(T)の機能
0	ステップトレーストラップ無効
1	ステップトレーストラップ有効 このとき、ユーザ用NMIとユーザ割込みがすべて割込禁止となります。

このビットは、リセットにより"0"に初期化されます。

ステップトレーストラップの機能はエミュレータが使用します。エミュレータ使用時、ユーザプログラム中で使用することはできません。

## 割込みレベルマスクレジスタ(ILM)

図 3.3-11に、割込みレベルマスクレジスタ(ILM)の構成を示します。

bit	20	19	18	17	16	初期値
	ILM4	ILM3	ILM2	ILM1	ILM0	01111 <sub>B</sub>

図 3.3-11 割込みレベルレジスタ(ILM)のレジスタ構成

割込みレベルマスクレジスタ(ILM)は、割込みレベルマスク値を保持するレジスタで、このILMの保持する値がレベルマスクに使用されます。

CPUに入力される割込要求の中で、対応する割込みレベルが、このILMで示されるレベルよりも強い場合にのみ割込要求が受け付けられます。

レベル値は、0(00000<sub>B</sub>)が最強で、31(11111<sub>B</sub>)が最弱です。

プログラムから設定可能な値には制限があります。元の値が16～31の場合、新たな値として設定できるのは16～31です。0～15を設定する命令を実行すると、(指定した値+16)という値が転送されます。

元の値が0～15の場合は、0～31の任意の値が設定可能です。

このレジスタは、リセットにより15(01111<sub>B</sub>)に初期化されます。



### 3.4 汎用レジスタ

レジスタR0～R15は汎用レジスタです。各種演算におけるアキュムレータおよびメモリアクセスのポインタとして使用されます。

汎用レジスタ

図 3.4-1に、汎用レジスタの構成を示します。

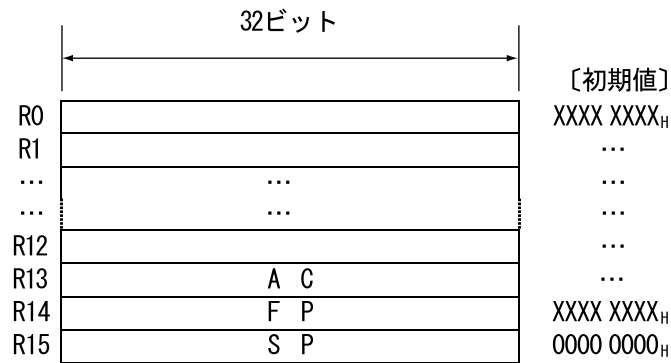


図 3.4-1 汎用レジスタの構成

16本のレジスタのうち、以下に示すレジスタは特殊な用途を想定しています。そのため、一部の命令が強化されています。

- R13: 仮想アキュムレータ (AC)
- R14: フレームポインタ (FP)
- R15: スタックポインタ (SP)

リセットによるR0～R14の初期値は不定です。R15の初期値は,00000000<sub>H</sub>(SSPの値)となります。

## 3.5 データ構造

FRファミリのデータ配置には、次の二つがあります。

- ・ビットオーダリングの場合:リトルエンディアン
- ・バイトオーダリングの場合:ビッグエンディアン

### ビットオーダリング

FRファミリでは、ビットオーダリングとして、リトルエンディアンを採用しています。

図 3.5-1に、ビットオーダリングのビット構成を示します。

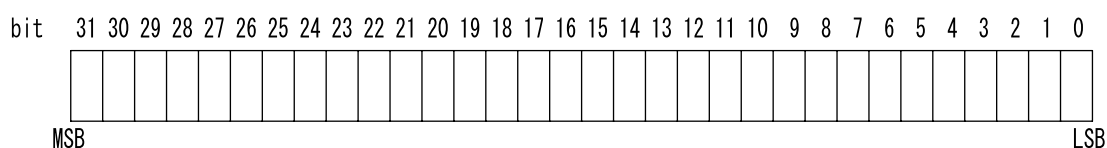


図 3.5-1 ビットオーダリングのビット構成

### バイトオーダリング

FRファミリでは、バイトオーダリングとして、ビッグエンディアンを採用しています。

図 3.5-2に、バイトオーダリングの構造を示します。

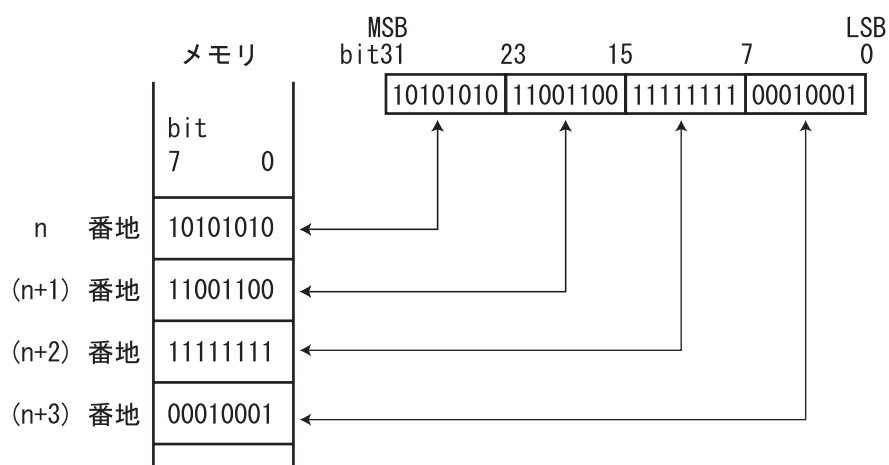


図 3.5-2 バイトオーダリングの構造

## 3.6 ワードアライメント

命令やデータはバイト単位でアクセスするため、配置するアドレスは命令長やデータ幅で異なります。

### プログラムアクセス

FRファミリのプログラムは、2の倍数のアドレスに配置する必要があります。

プログラムカウンタ(PC)のビット0は、命令の実行に伴うPCの更新時に、“0”に設定されます。

“1”になる可能性があるのは、分岐先アドレスとして奇数番地を指定した場合だけです。

ただし、その場合でもビット0は無効であり、命令は2の倍数のアドレスに置かなくてはなりません。

奇数アドレス例外はありません。

### データアクセス

FRファミリでは、データアクセスを行う際、その幅により以下のように強制的なアラインメントがアドレスに対して施されます。

- ワードアクセス : アドレスは、4の倍数  
(最下位2ビットは強制的に“00”)
- ハーフワードアクセス: アドレスは、2の倍数  
(最下位ビットは強制的に“0”)
- バイトアクセス :

ワードおよびハーフワードデータアクセス時に、一部のビットが強制的に“0”にされるのは、実効アドレスの計算結果に対してです。例えば、@ (R13, Ri) のアドレッシングモードの場合、加算前のレジスタは(たとえ最下位ビットが“1”であっても)そのまま計算に使用され、加算結果の下位ビットがマスクされます。計算前のレジスタがマスクされるわけではありません。

【例】 LD @ (R13, R2), R0

R13	00002222 <sub>H</sub>
R2	00000003 <sub>H</sub>
+)	
加算結果	00002225 <sub>H</sub>
	↓ 下位2ビット強制マスク
アドレス端子	00002224 <sub>H</sub>

## 3.7 メモリマップ

MB91F127/128のメモリマップと,FRファミリ共通のメモリマップを示します。

MB91F127/128のメモリマップ

メモリのアドレス空間は32ビットリニアです。

図 3.7-1に,MB91F127/128のメモリマップを示します。

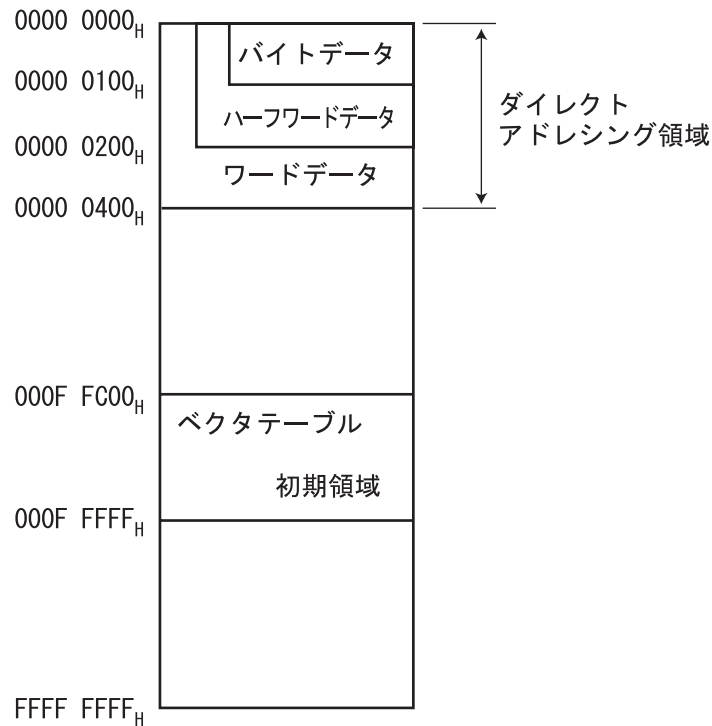


図 3.7-1 メモリマップ

### ダイレクトアドレッシング領域

アドレス空間の下記の領域は,I/O用の領域です。この領域は,ダイレクトアドレッシングにより,命令中で直接オペランドアドレスを指定することができます。

ダイレクトアドレス指定可能なアドレス領域の大きさは,データ長ごとに異なります。

- バイトデータ(8ビット) : 0 ~ 0FF<sub>H</sub>
- ハーフワードデータ(16ビット) : 0 ~ 1FF<sub>H</sub>
- ワードデータ(32ビット) : 0 ~ 3FF<sub>H</sub>

### ベクタテーブル初期領域

000FFC00<sub>H</sub> ~ 000FFFFF<sub>H</sub>の領域はEITベクタテーブル初期領域です。

EIT処理時に使用されるベクタテーブルは,TBRを書き換えることにより任意のアドレスに配置可能ですが,リセットによる初期化によってこのアドレスに配置されます。

FRファミリ共通のメモリマップ

FRファミリでは、メモリマップを以下のように定めています。これは、品種によらず共通です(シングルチップモードを除く)。  
図 3.7-2に、FRファミリ共通のメモリマップを示します。

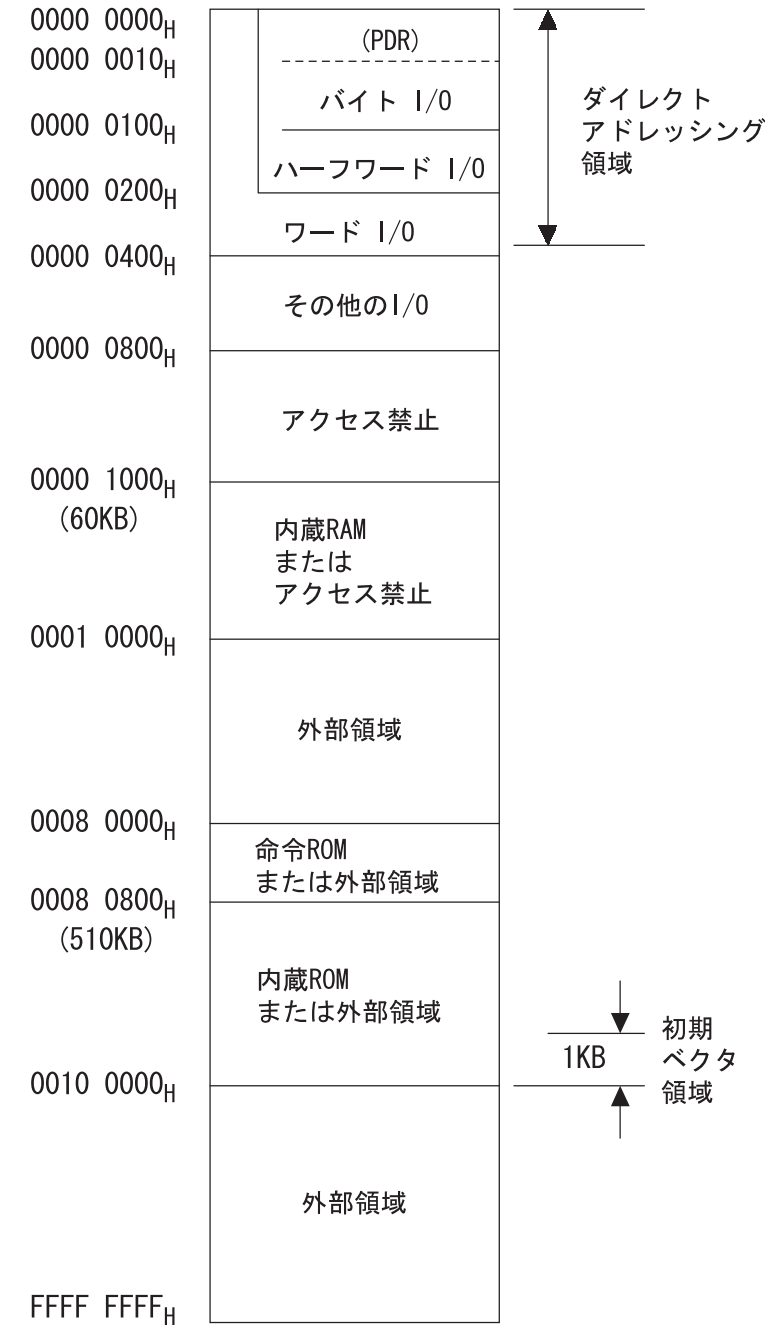


図 3.7-2 FRファミリ共通のメモリマップ

< 注意事項 >

シングルチップモードでは、外部領域はアクセス不可になります。  
MB91F127/128では、内蔵ROM領域の080000<sub>H</sub> ~ 0807FF<sub>H</sub>を内蔵RAM2KBにアサインしています。

## 3.8 命令概要

FRファミリは、一般的なRISCの命令体系に加え、組み込み用途に最適化された論理演算、ビット操作およびダイレクトアドレッシング命令をサポートしています。各命令は16ビット長(一部命令は32, 48ビット長)ですので、優れたメモリ使用効率を持ちます。

命令セットは以下の機能グループに分けることができます。

- ・算術演算
- ・ロードとストア
- ・分岐
- ・論理演算とビット操作
- ・ダイレクトアドレッシング
- ・その他

### 命令概要

#### 算術演算

標準の算術演算命令(加算, 減算, 比較)およびシフト命令(論理シフト, 算術演算シフト)があります。加算と減算については, 多ワード長演算で使用するキャリ付演算や, アドレス計算に便利なフラグ値を変化させない演算も可能です。

さらに, 32ビット×32ビット, 16ビット×16ビットの乗算命令と, 32ビット÷32ビットのステップ除算命令を持ちます。

また, レジスタに即値をセットする即値転送命令や, レジスタ間転送命令も備えています。

算術演算命令は, すべてCPU内の汎用レジスタおよび乗除算レジスタを用いて演算を行います。

#### ロードとストア

ロードとストアは外部メモリに対して読出しと書込みを行う命令です。また, チップ内の周辺回路(I/O)への読出しと書込みにも使用されます。

ロードとストアはバイト, ハーフワード, ワードの3種類のアクセス長を持ちます。また一般的なレジスタ間接のメモリアドレッシングに加え, 一部の命令についてはディスプレイースメント付レジスタ間接やレジスタインクリメントデクリメント付レジスタ間接のメモリアドレッシングも可能です。

#### 分岐

分岐, コール, 割込みおよび復帰の命令です。分岐命令は, 遅延スロットを持つものと持たないものがあり, 用途に応じて最適化を行うことができます。

分岐命令の詳細については「3.8.1 遅延スロット付き分岐命令」「3.8.2 遅延スロットなし分岐命令」を参照してください。

#### 論理演算とビット操作

論理演算命令は汎用レジスタ間または汎用レジスタとメモリ(およびI/O)間でAND, OR, EORの論理演算を行うことが可能です。また, ビット操作命令はメモリ(およびI/O)の内容を直接操作することができます。メモリアドレッシングは一般的なレジスタ間接です。

#### ダイレクトアドレッシング

ダイレクトアドレッシング命令はI/Oと汎用レジスタ間またはI/Oとメモリ間のアクセスに使用する命令です。I/Oのアドレスをレジスタ間接ではなく命令中で直接指定することにより、高速、高効率なアクセスを行うことができます。一部の命令についてはレジスタインクリメントデクリメント付レジスタ間接のメモリアドレッシングも可能です。

#### その他

プログラムステータスレジスタ(PS)内のフラグ設定、スタック操作、符号/ゼロ拡張等を行う命令です。また、高級言語対応の関数入口/出口、レジスタマルチロード/ストア命令も備えています。

### 3.8.1 遅延スロット付き分岐命令

遅延スロット付きの動作では、分岐先の命令を実行する前に、分岐命令の直後(“遅延スロット”と呼びます)に置かれた命令を実行した後に分岐します。

遅延スロット付き分岐命令

遅延スロット付きの分岐命令には、以下の命令があります。

JMP:D @Ri	CALL:D label12	CALL:D @Ri	RET:D
BRA:D label9	BNO:D label9	BEQ:D label9	BNE:D label9
BC:D label9	BNC:D label9	BN:D label9	BP:D label9
BV:D label9	BNV:D label9	BLT:D label9	BGE:D label9
BLE:D label9	BGT:D label9	BLS:D label9	BHI:D label9

遅延スロット付き分岐命令の動作説明

遅延スロット付きの動作では、分岐先の命令を実行する前に、分岐命令の直後(「遅延スロット」と呼びます)に置かれた命令を実行した後に、分岐します。

分岐動作の前に遅延スロットの命令を実行するため、見かけ上の実行速度が1サイクルとなります。その代わり、遅延スロットに有効な命令を入れることができないときは、NOP命令を置かなくてはなりません。

[ 例 ]

```

;      命令の並び
ADD      R1, R2          ;
BRA:D    LABEL           ; 分岐命令
MOV      R2, R3          ; 遅延スロット...分岐の前に実行される
...
LABEL : ST      R3, @R4    ; 分岐先

```

条件分岐命令の場合、分岐条件が成立する場合も成立しない場合も遅延スロットに置かれた命令は実行されます。

遅延分岐命令では、一部の命令の実行順序が反転するように見えますが、それはPCの更新動作だけについてであり、その他の動作(レジスタの更新/参照など)はあくまで記述された順番で実行されます。

以下に、具体例を示します。

JMP:D @Ri / CALL:D @Ri 命令で参照するRiは、遅延スロットの中の命令がRiを更新しても影響を受けません。

[ 例 ]

```

LDI:32 #Label, R0
JMP:D @R0          ; Label に分岐
LDI:8  #0, R0      ; 分岐先アドレスには影響を与えない。
...

```



RET:D 命令が参照するRPは、遅延スロットの中の命令がRPを更新しても影響を受けません。

[ 例 ]

```
RET:D                                ;これより前に設定されたRPの示すアドレスへの分岐
MOV    R8,        RP                ;リターン動作には影響を与えない
...
```

Bcc:D rel命令が参照するフラグも、遅延スロットの命令の影響を受けません。

[ 例 ]

```
ADD    #1,        R0                ;フラグ変化
BC:D    Overflow                    ;上記の命令の実行結果により分岐
ANDCCR #0                                ;このフラグ更新は上記分岐命令では参照しない
...
```

CALL:D命令の遅延スロット中の命令でRPを参照すると、CALL:D命令により更新された内容が読み出されます。

[ 例 ]

```
CALL:D Label                        ;RPを更新して分岐
MOV    RP,        R0                ;上記CALL:Dの実行結果のRPを転送
...
```

#### 遅延スロット付き分岐命令の制限事項

遅延スロットに置くことができる命令

遅延スロット内で実行できるのは、以下の条件を満たす命令のみです。

- 1サイクル命令
- 分岐命令ではないこと
- 順番が変化した場合でも動作に影響を与えない命令

「1サイクル命令」とは、命令一覧表中でサイクル数の欄が「1」、「a」、「b」、「c」または「d」と記載された命令です。

ステップトレーストラップ

遅延スロットを持つ分岐命令の実行と遅延スロットの間では、ステップトレーストラップは発生しません。

割込み/NMI

遅延スロットを持つ分岐命令の実行と遅延スロットの間では、割込み/NMIを受理しません。

未定義命令例外

遅延スロットに未定義命令があった場合、未定義命令例外は発生しません。このとき、未定義命令はNOP命令として動作します。

### 3.8.2 遅延スロットなし分岐命令

---

遅延スロットなしの動作では、あくまで命令の順に実行します。

---

遅延スロットなし分岐命令

遅延スロットなしの分岐命令には、以下の命令があります。

JMP @Ri	CALL label12	CALL @Ri	RET
BRA label9	BNO label9	BEQ label9	BNE label9
BC label9	BNC label9	BN label9	BP label9
BV label9	BNV label9	BLT label9	BGE label9
BLE label9	BGT label9	BLS label9	BHI label9

遅延スロットなし分岐命令の動作説明

遅延スロットなしの動作では、あくまで命令の並びの順に実行します。直後の命令が分岐前に実行されることはありません。

[ 例 ]

```

;      命令の並び
ADD      R1, R2          ;
BRA      LABEL           ; 分岐命令(遅延スロットなし)
MOV      2, R3           ; 実行されない
...
LABEL: ST      R3, @R4    ; 分岐先

```

遅延スロットなしの分岐命令の実行サイクル数は、分岐するとき2サイクル、分岐しないとき1サイクルとなります。これは、遅延スロットに適切な命令を入れることができないためにNOPを明記した遅延スロット付き分岐命令に比べ、命令コード効率を上げることができます。遅延スロットに有効な命令を設置できるときは遅延スロット付きの動作を選択し、そうでないときは遅延スロットなしの動作を選択することで、実行速度とコード効率を両立させることが可能となります。

## 3.9 EIT(例外, 割込み, トラップ)

EITとは, 現プログラム実行時にイベントの発生により, そのプログラムの実行を中断し, ほかのプログラムを実行することを指し, 例外(Exception), 割込み(Interrupt), トラップ(Trap)の総称です。

### EIT(例外, 割込み, トラップ)

例外とは, 実行中のコンテキストに関連して発生する事象です。例外を起こした命令から再実行します。

割込みとは, 実行中のコンテキストに無関係に発生する事象です。イベント要因は, ハードウェアです。

トラップとは, 実行中のコンテキストに関連して発生する事象です。システムコールのようにプログラムで指示するものがあります。トラップを起こした命令の次の命令から再実行します。

### 特長

- 割込みに多重割込みをサポート
- 割込みにレベルマスク機能(15レベルをユーザが使用可能)
- トラップ命令(INT)
- エミュレータ起動用EIT(ハードウェア/ソフトウェア)

### EIT要因

EIT要因として, 以下のものがあります。

- リセット
- ユーザ割込み(内部リソース, 外部割込み)
- 遅延割込み
- 未定義命令例外
- トラップ命令(INT)
- トラップ命令(INTE)
- ステップトレーストラップ
- コプロセッサ不在トラップ
- コプロセッサエラートラップ

### EITからの復帰

EITからの復帰には, RETI命令を使用します。

### EITの注意事項

- 遅延スロット  
分岐命令の遅延スロットには, EITに関して制約があります。詳しくは「3.8 命令概要」を参照してください。

### 3.9.1 EITの割込みレベル

割込みレベルは0～31で,5ビットで管理されます。

EITの割込みレベル

表 3.9-1に,各レベルの割当てを示します。

表 3.9-1 EITの割込みレベル

レベル		要因	備考
2進数	10進数		
00000	0	(システム予約)	ILMの元の値が16～31のとき,この範囲の値をプログラムによりILMに設定することはできません。
...	...	...	
...	...	...	
00011	3	(システム予約)	
00100	4	INTE命令 ステップトレーストラップ	
00101	5	(システム予約)	
...	...	...	ILM設定時,ユーザ割込禁止
...	...	...	
01110	14	(システム予約)	
01111	15	NMI(ユーザ用)	
10000	16	割込み	
10001	17	割込み	ICR設定時,割込禁止
...	...	...	
...	...	...	
11110	30	割込み	
11111	31		

操作が可能なのは,16～31のレベルです。

未定義命令例外,コプロセッサ不在トラップ,コプロセッサエラートラップおよびINT命令は,割込みレベルの影響を受けません。また,ILMを変化させることもありません。

#### Iフラグ

割込みの許可/禁止を指定するフラグです。プログラムステータスレジスタ(PS)のCCRのビット4として設けられています。

I	Iフラグの機能
0	割込み禁止。 INT命令実行時"0"にクリアされます。 (ただし,スタック退避させる値はクリアする前の値です。)
1	割込み許可。 割込要求のマスク処理は,ILMの保持する値により制御されます。

#### 割込みレベルマスクレジスタ (ILM)

割込みレベルマスク値を保持するプログラムステータスレジスタ (PS) (ビット20～16) です。

CPUに入力される割込み要求の中で、対応する割込みレベルが、この割込みレベルマスクレジスタ (ILM) で示されるレベルよりも強い場合にのみ割込要求が受け付けられます。

レベル値は、0 (00000<sub>B</sub>) が最強で、31 (11111<sub>B</sub>) が最弱です。

プログラムから設定可能な値には制限があります。元の値が16～31のとき、新たな値として設定できるのは16～31です。0～15の値を設定する命令を実行すると、“指定した値+16”という値が転送されます。

元の値が0～15のときは、0～31の任意の値が設定可能です。

#### < 注意事項 >

.....  
割込みレベルマスクレジスタ (ILM) を設定するには、STILM命令を使用してください。  
.....

#### 割込みに対するレベルマスク

割込み要求が発生したときは、割込み要因の持つ割込みレベル(「表 3.9-1」を参照してください)がILMの保持するレベルマスク値と比較されます。そして、次の条件が成立したときはマスクされ、要求は受理されません。

要因の持つ割込みレベル	レベルマスク値
-------------	---------

3.9.2 割り込み制御レジスタ(ICR)

割り込み制御レジスタ(ICR: Interrupt Control Register)は, 割り込みコントローラ内に設けられたレジスタで, 割り込みの各要求に対するレベルを設定します。ICRは, 割り込み要求入力のそれぞれに対応して用意されています。ICRはI/O空間にマッピングされており, CPUからはバスを通してアクセスされます。

割り込み制御レジスタ(ICR)のビット構成

図 3.9-1に, 割り込み制御レジスタ(ICR)のビット構成を示します。

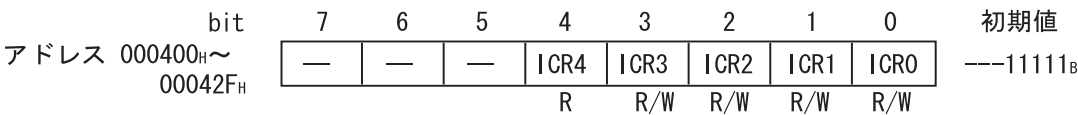


図 3.9-1 割り込み制御レジスタ(ICR)のビット構成

以下に, 割り込み制御レジスタ(ICR)の各ビットの機能を説明します。

【ビット4】ICR4

このビットは, 常に”1”です。

【ビット3~0】ICR3~0

これらのビットは, 対応する割り込み要因の割り込みレベルの下位4ビットです。読出し, 書込みが可能です。

ビット4と合わせて, ICRは16~31の範囲で値を設定することができます。

割り込み制御レジスタ(ICR)のマッピング

表 3.9-2に, 割り込要因と割り込みベクタの割当てを示します。

表 3.9-2 割り込要因と割り込みベクタの割当て

割り込要因	割り込み制御レジスタ		対応する割り込みベクタ		
	番号	アドレス	番 号		アドレス
			16進	10進	
IRQ00	ICR00	00000400 <sub>H</sub>	10 <sub>H</sub>	16	TBR + 3BC <sub>H</sub>
IRQ01	ICR01	00000401 <sub>H</sub>	11 <sub>H</sub>	17	TBR + 3B8 <sub>H</sub>
IRQ02	ICR02	00000402 <sub>H</sub>	12 <sub>H</sub>	18	TBR + 3B4 <sub>H</sub>
...	...	...	...	...	...
...	...	...	...	...	...
IRQ45	ICR45	0000042D <sub>H</sub>	3D <sub>H</sub>	61	TBR + 308 <sub>H</sub>
IRQ46	ICR46	0000042E <sub>H</sub>	3E <sub>H</sub>	62	TBR + 304 <sub>H</sub>
IRQ47	ICR47	0000042F <sub>H</sub>	3F <sub>H</sub>	63	TBR + 300 <sub>H</sub>

詳細は「第12章 遅延割り込みコントローラ」を参照してください。

### 3.9.3 システムスタックポインタ(SSP)

システムスタックポインタ(SSP)は,EITの受理および復帰動作時のデータ退避/復帰用スタックを示すポインタとして使用されます。

システムスタックポインタ(SSP)

システムスタックポインタ(SSP)は,32ビットから構成されています。

図 3.9-2に,システムスタックポインタ(SSP)のビット構成を示します。



図 3.9-2 システムスタックポインタ(SSP)のビット構成

EIT処理時に内容が8減算され,RETI命令の実行によるEITからの復帰動作時に8加算されます。

リセットによる初期値は00000000<sub>h</sub>です。

SSPは,CCR中のSフラグが“0”のとき,汎用レジスタR15としても機能します。

割込みスタック

システムスタックポインタ(SSP)により示される領域で,PCおよびPSの値が退避/復活されます。割込み後は,SSPの示すアドレスにPCが,“SSP+4”のアドレスにPSが格納されています。

図 3.9-3に,割込みスタックの例を示します。

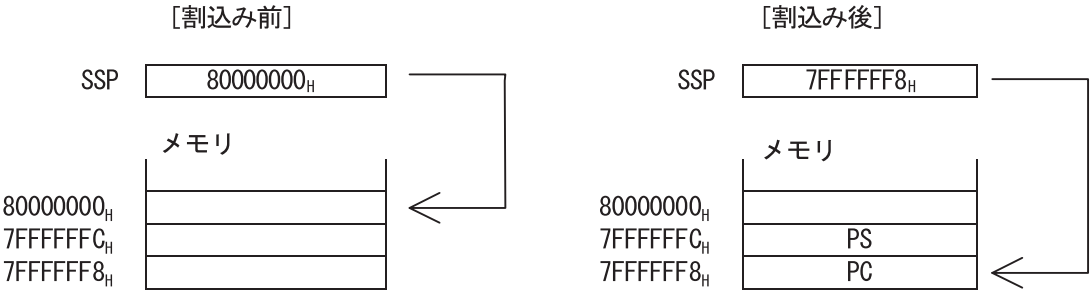


図 3.9-3 割込みスタックの例

### 3.9.4            テーブルベースレジスタ(TBR)

テーブルベースレジスタ(TBR:Table Base Register)は,EIT用ベクタテーブルの先頭アドレスを示すレジスタです。

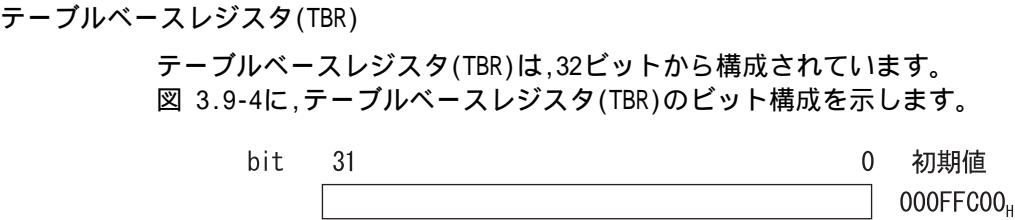


図 3.9-4    テーブルベースレジスタ(TBR)のビット構成

TBRとEIT要因ごとに決められたオフセット値を加算したアドレスが,ベクタアドレスとなります。  
リセットによる初期値は000FFC00<sub>H</sub>です。

#### EITベクタテーブル

テーブルベースレジスタ(TBR)の示すアドレスから1KBの領域が,EIT用ベクタ領域となっています。

1ベクタ当たりの大きさは4バイトで,ベクタ番号とベクタアドレスの関係は下記のように表されます。

$$\begin{aligned} \text{vctadr} &= \text{TBR} + \text{vctofs} \\ &= \text{TBR} + (3\text{FC}_{\text{H}} - 4 \times \text{vct}) \end{aligned}$$

vctadr:ベクタアドレス  
vctofs:ベクタオフセット  
vct    :ベクタ番号

加算結果の下位2ビットは,常に"00"として扱われます。  
000FFC00<sub>H</sub> ~ 000FFFFF<sub>H</sub>の領域がリセットによるベクタテーブルの初期領域です。  
ベクタの一部には特殊な機能が割り当てられています。  
表 3.9-3に,アーキテクチャ上のベクタテーブルを示します。



表 3.9-3 ベクタテーブル

ベクタオフセット (16進数)	ベクタ番号		説 明
	16進数	10進数	
3FC	00	0	リセット *1
3F8	01	1	システム予約
3F4	02	2	システム予約
3F0	03	3	システム予約
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮
3E0	07	7	システム予約
3DC	08	8	システム予約
3D8	09	9	INTE命令
3D4	0A	10	システム予約
3D0	0B	11	システム予約
3CC	0C	12	ステップトレーストラップ
3C8	0D	13	システム予約
3C4	0E	14	未定義命令例外
3C0	0F	15	NMI (ユーザ用)
3BC	10	16	マスク可能割り込み要因 #0
3B8	11	17	マスク可能割り込み要因 #1 *2
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮
300	3F	63	マスク可能割り込み要因/INT命令
2FC	40	64	システム予約 (REALOSにて使用)
2F8	41	65	システム予約 (REALOSにて使用)
2F4	42	66	マスク可能割り込み要因/INT命令
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮
000	FF	255	

\*1: TBRの値を変更してもリセットベクタは常に固定アドレス000FFFFC<sub>H</sub>が使用されます。

\*2: MB91F128におけるベクタテーブルは「付録 B 割り込みベクタ」を参照してください。

### 3.9.5 多重EIT処理

複数のEIT要因が同時に発生した場合,CPUは一つのEIT要因を選択して受理し,EITシーケンスを実行した後,再びEIT要因の検出を行う,という動作を繰り返します。

EIT要因検出の際に,受理可能なEIT要因がなくなったとき,最後に受理したEIT要因のハンドラの命令を実行します。そのため,複数のEIT要因が同時に発生した場合の各要因に対するハンドラの実行順序は,次の二つの要素によって決まります。

- ・EIT要因受理の優先順位
- ・受理した場合にほかの要因をどのようにマスクするか

#### EIT要因受理の優先順位

EIT要因受理の優先度とは,PSとPCを退避してPCを更新し(必要に応じて)ほかの要因のマスク処理を行うという,EITシーケンスを実行する要因を選ぶときの順番です。必ずしも,先に受理された要因のハンドラが先に実行されるわけではありません。

表 3.9-4に,EIT要因受理の優先度を示します。

表 3.9-4 EIT要因の受理の優先度とほかの要因へのマスク

受理の優先順位	要 因	ほかの要因に対するマスク
1	リセット	ほかの要因は破棄されます。
2	未定義命令例外	取消
3	INT命令	Iフラグ=0
	コプロセッサ不在トラップ	なし
	コプロセッサエラートラップ	
4	ユーザ割込み	ILM=受理した要因のレベル
5	NMI(ユーザ用)	ILM=15
6	ステップトレーストラップ	ILM=4
7	INTE命令	ILM=4

EIT要因を受理した後,ほかの要因に対するマスクの処理を加味すると同時に発生したEIT要因の各ハンドラの実行順序は以下ようになります。

表 3.9-5に,EIT要因の各ハンドラの実行順序を示します。

表 3.9-5 EITハンドラの実行順序

ハンドラの実行順序	要 因
1	リセット *1
2	未定義命令例外
3	ステップトレーストラップ *2
4	INTE命令 *2
5	NMI(ユーザ用)
6	INT命令
7	ユーザ割込み
8	コプロセッサ不在トラップ,コプロセッサエラートラップ

\*1:ほかの要因は破棄されます。

\*2:ステップトレーストラップによるEITが発生する状況下でのINTE命令は使用できません。

図 3.9-5に, 多重EIT処理の例を示します。

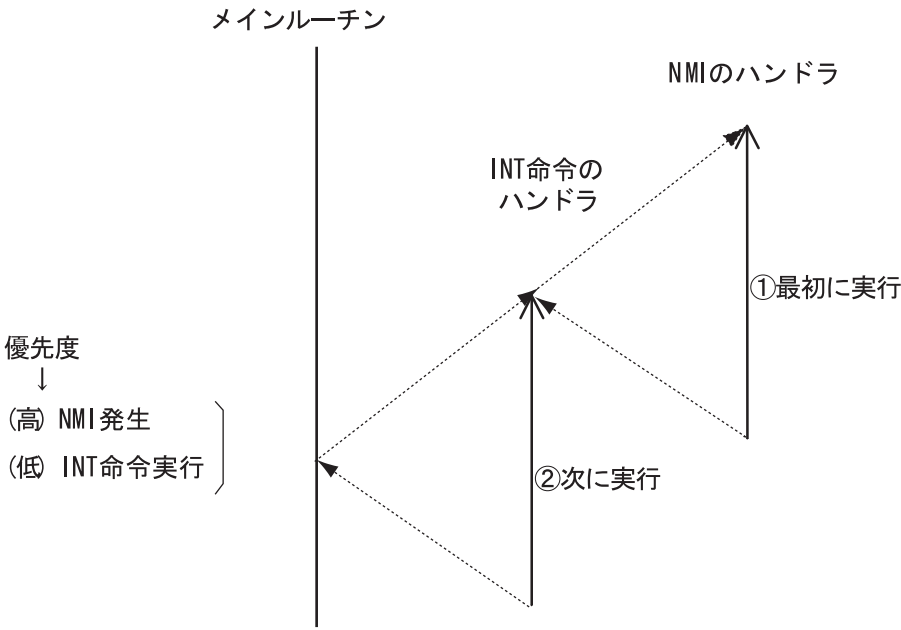


図 3.9-5 多重EIT処理の例

### 3.9.6 EITの動作

EITの動作について説明します。

#### EITの動作

以下の説明で、転送元の“PC”とは各EIT要因を検出した命令のアドレスを示すものとします。

また、動作説明の中で“次の命令のアドレス”とは、EITを検出した命令が下記のようにになっていることを意味します。

- LDIが32のとき:PC+6
- LDIが20で,COPOP,COPLD,COPST,COPSVのとき:PC+4
- その他の命令のとき:PC+2

#### ユーザ割込みの動作

ユーザ割込み要求が発生すると、以下の順序で要求受理の可否が判定されます。

##### 割込み要求受理の可否判定

- 1) 同時に発生した要求の割込みレベルを比較し、最も強いレベル(最も小さい数値)を保持するものが選択されます。比較に使用されるレベルは、マスク可能割込みについては対応するICRの保持する値が使用されます。
- 2) 同じレベルをもつ割込み要求が複数発生しているときは、最も若い割込み番号をもつ割込要求が選択されます。
- 3) 選択された割込み要求の割込みレベルをILMで決定されるレベルマスク値と比較します。  
割込みレベル    レベルマスク値のとき、割込要求はマスクされ受理されません。  
割込みレベル < レベルマスク値のときは、4)へ移ります。
- 4) 選択された割込み要求がマスク可能割込みであるとき、Iフラグが“0”ならば割込み要求はマスクされ、受理されません。Iフラグが“1”ならば 5)へ移ります。
- 5) 上記の条件が成立したとき、命令処理の切れ目で割込要求が受理されます。

EIT要求検出時にユーザ割込みの要求が受理されると、受理された割込み要求に対応した割込み番号を使用して、CPUは以下のように動作します。[動作]における( )は、レジスタの指すアドレスを表します。

##### 【動作】

- 1) SSP-4                  SSP
- 2) PS                    (SSP)
- 3) SSP-4                  SSP
- 4) 次の命令のアドレス      (SSP)
- 5) 受理した要求の割込みレベル      ILM
- 6) “0”                    Sフラグ
- 7) (TBR + 受理した割込み要求のベクタオフセット)      PC

割込みシーケンス終了後、ハンドラの先頭の命令を実行する前に、新たなEITの検出を行います。この時点で受理可能なEITが発生していると、CPUはEIT処理シーケンスに遷移します。

#### INT命令の動作

INT #u8命令は, 以下のように動作します。  
u8で示されるベクタの割込みハンドラへ分岐します。

##### 【動作】

- 1) SSP-4                      SSP
- 2) PS                        (SSP)
- 3) SSP-4                      SSP
- 4) PC+2                      (SSP)
- 5) "0"                        Iフラグ
- 6) "0"                        Sフラグ
- 7)  $(TBR + 3FC_H - 4 \times u8)$               PC

#### INTE命令の動作

INTE命令は, 以下のように動作します。  
ベクタ番号9のベクタの割込みハンドラへ分岐します。

##### 【動作】

- 1) SSP-4                      SSP
- 2) PS                        (SSP)
- 3) SSP-4                      SSP
- 4) PC+2                      (SSP)
- 5) "00100"                      ILM
- 6) "0"                        Sフラグ
- 7)  $(TBR + 3D8_H)$               PC

INTE命令およびステップトレーストラップの処理ルーチン中では, INTE命令は使  
用しないでください。

また, ステップ実行中はINTEによるEITの発生はありません。

### ステップトレーストラップの動作

PS中のSCRにおけるTフラグをセットしてステップトレースの機能をイネーブルにしておくと、一命令実行ごとにトラップが発生し、ブレイクします。ステップトレーストラップ検出の条件は、以下のとおりです。

ステップトレーストラップ検出の条件

- Tフラグ=1
- 遅延分岐命令ではないとき
- INTE命令,ステップトレーストラップの処理ルーチン以外を実行中であるとき

以上の条件が成立すると、命令動作の切れ目でブレイクします。

#### 【動作】

- 1) SSP-4                      SSP
- 2) PS                        (SSP)
- 3) SSP-4                      SSP
- 4) 次の命令のアドレス        (SSP)
- 5) "00100"                  ILM
- 6) "0"                        Sフラグ
- 7) (TBR + 3CC<sub>H</sub>)            PC

Tフラグをセットしてステップトレーストラップをイネーブルにしたとき、ユーザ割込みは禁止状態となります。また、INTE命令によるEITは発生しなくなります。

### 未定義命令例外の動作

命令のデコード時に未定義命令であることを検出すると、未定義命令例外が発生します。

未定義命令例外の検出条件は、以下のとおりです。

- 命令のデコード時に、未定義命令であることを検出。
- 遅延スロット外に置かれている。(遅延分岐命令の直後ではない。)

以上の条件が成立すると、未定義命令例外が発生しブレイクします。

#### 【動作】

- 1) SSP-4                      SSP
- 2) PS                        (SSP)
- 3) SSP-4                      SSP
- 4) PC                        (SSP)
- 5) "0"                        Sフラグ
- 6) (TBR + 3C4<sub>H</sub>)            PC

PCとして退避されるのは、未定義命令例外を検出した命令自身のアドレスです。

### コプロセッサ不在トラップ

実装していないコプロセッサを使用するコプロセッサ命令を実行すると、コプロセッサ不在トラップが発生します。

#### 【動作】

- 1) SSP-4                      SSP
- 2) PS                        (SSP)
- 3) SSP-4                      SSP
- 4) 次の命令のアドレス      (SSP)
- 5) "0"                        Sフラグ
- 6) (TBR + 3E0<sub>H</sub>)            PC

### コプロセッサエラートラップ

コプロセッサを使用しているときにエラーが発生した場合、次にそのコプロセッサを操作するコプロセッサ命令を実行したとき、コプロセッサエラートラップが発生します。(なお、MB91F128にはコプロセッサは搭載していません。)

#### 【動作】

- 1) SSP-4                      SSP
- 2) PS                        (SSP)
- 3) SSP-4                      SSP
- 4) 次の命令のアドレス      (SSP)
- 5) "0"                        Sフラグ
- 6) (TBR + 3DC<sub>H</sub>)            PC

### RETI命令の動作

RETI命令は、EIT処理ルーチンから復帰する命令です。

#### 【動作】

- 1) (R15)                    PC
- 2) R15+4                   R15
- 3) (R15)                    PS
- 4) R15+4                   R15

RETI命令は、Sフラグが"0"の状態で行う必要があります。

## 3.10 リセットシーケンス

---

この節では,CPUを初期状態にするときのリセットについて説明します。

---

### リセット要因

リセット要因は以下のとおりです。

- 外部リセット端子からの入力
- スタンバイ制御レジスタ(STCR)中のSRSTビット操作によるソフトウェアリセット
- ウォッチドッグタイマのカウントアップ
- パワーオンリセット

### リセットによる初期化

リセット要因が発生すると,CPUは初期化されます。

外部リセット端子/ソフトウェアリセットからの解除

- 端子を定められた状態に設定します。
- デバイス内部の各リソースを,リセット状態にします。制御レジスタは,あらかじめ決められた値に初期化されます。
- クロックとして,最も遅いギアが選択されます。

### リセットシーケンス

リセット要因が解除されると,CPUは以下のリセットシーケンスを実行します。

- (000FFFFC<sub>H</sub>) PC

### <注意事項>

---

リセット後は,モードレジスタの設定により,動作モードを細かく設定します。  
詳細は,「3.13 動作モード」を参照してください。

---



### 3.11 クロック発生部と制御部

---

クロック発生部と制御部は以下の機能を受け持つモジュールです。
・CPUクロック生成(ギア機能を含む)
・周辺クロック生成(ギア機能を含む)
・リセット発生および要因の保持
・スタンバイ機能
・DMA要求の抑止
・PLL(逡倍回路)内蔵

---

クロック発生部と制御部のレジスタ一覧

図 3.11-1に、クロック発生部と制御部のレジスタ一覧を示します。

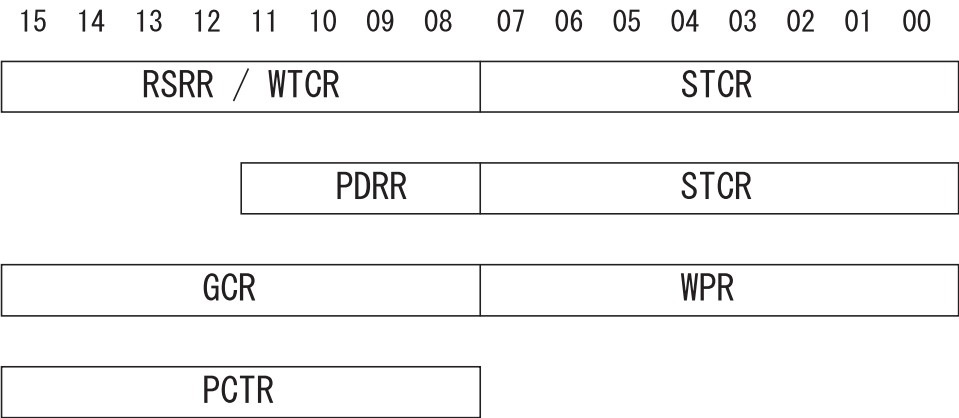


図 3.11-1 クロック発生部と制御部のレジスタ一覧

## クロック発生部と制御部のブロックダイアグラム

図 3.11-2に、クロック発生部と制御部のブロックダイアグラムを示します。

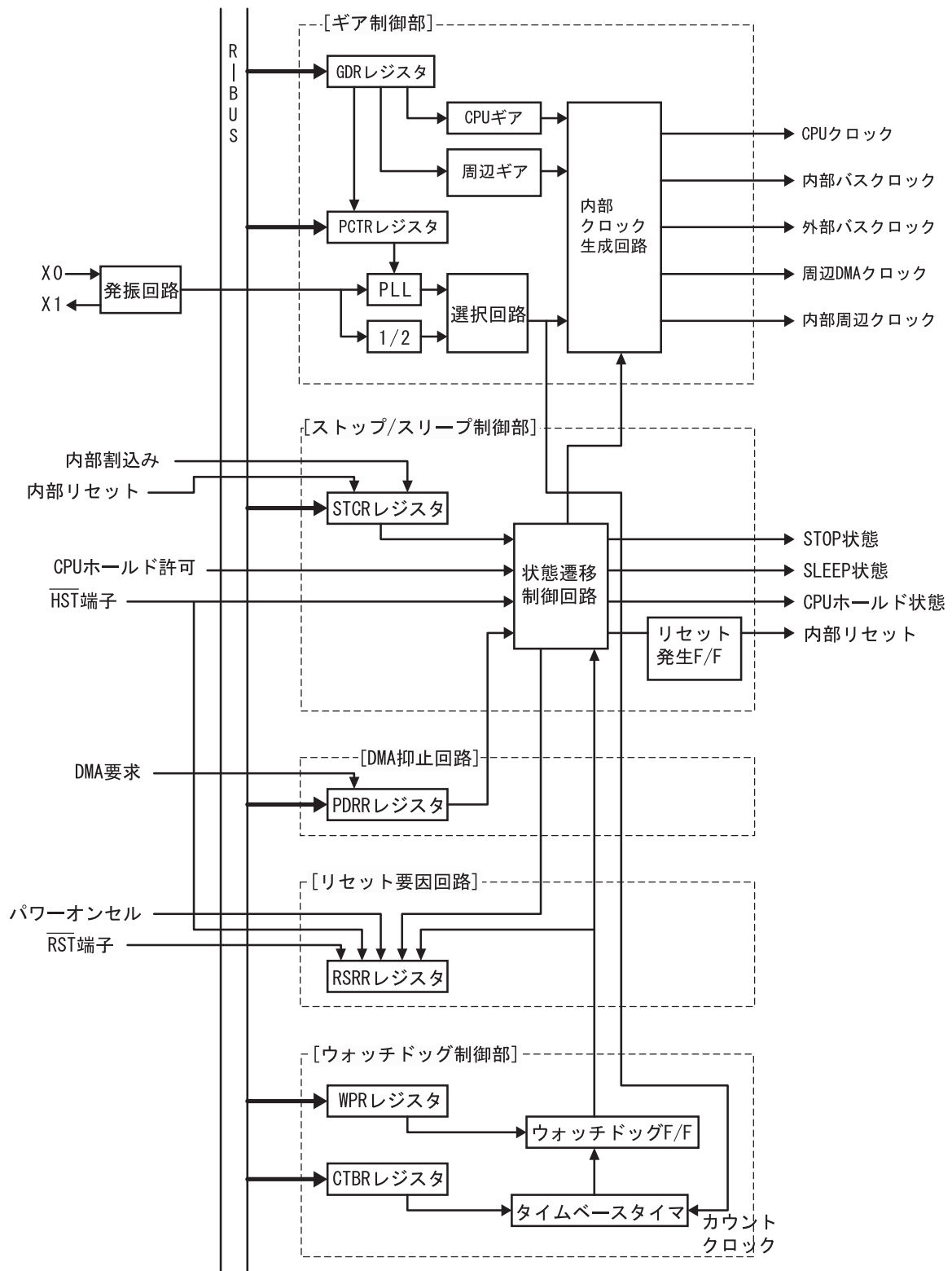


図 3.11-2 クロック発生部と制御部のブロックダイアグラム

### 3.11.1 リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)

リセット要因レジスタ(RSRR)は、発生したリセットの種別を保持するレジスタで、ウォッチドッグ周期制御レジスタ(WTCR)は、ウォッチドッグタイマの周期を指定するレジスタです。

リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)のレジスタ構成

図 3.11-3に、リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)のレジスタ構成を示します。

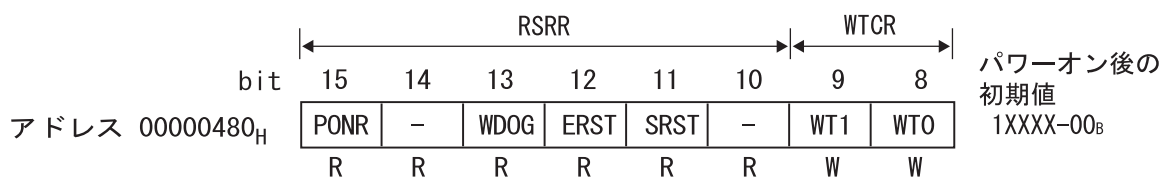


図 3.11-3 リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)のレジスタ構成

以下に、リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)の各ビットの機能を説明します。

【ビット15】:PONR

このビットが“1”であると、直前に発生したリセットはパワーオンリセットであったことを示します。このビットが“1”の場合は、このレジスタのこのビット以外の内容は無効となります。

【ビット14】:HSTB

このビットが“1”であると、直前に発生したリセットはハードウェアスタンバイであったことを示します。

【ビット13】:WDOG

このビットが“1”であると、直前に発生したリセットはウォッチドッグリセットであったことを示します。

【ビット12】:ERST

このビットが“1”であると、直前に発生したリセットは外部リセット端子に起因するリセットであったことを示します。

【ビット11】:SRST

このビットが“1”であると、直前に発生したリセットはソフトウェアリセット要求に起因するリセットであったことを示します。

【ビット10】:(Reserved)

このビットは予約ビットです。読出し値は不定です。

【ビット9,8】:WT1,0

このビットは,ウォッチドッグの周期を指定するものです。このビットと選択する周期は,表 3.11-1のような関係があります。このビットはすべてのリセットにより初期化されます。

表 3.11-1 WT1,WT0で指定されたウォッチドッグの周期

WT1	WT0	ウォッチドッグリセットの発生を抑止するために最低限必要なWPRへの書込み間隔	WPRへの最終5AH書込みからウォッチドッグリセットが発生するまでの時間
0	0	$\times 2^{15}$ [初期値]	$\times 2^{15} \sim \times 2^{16}$
0	1	$\times 2^{17}$	$\times 2^{17} \sim \times 2^{18}$
1	0	$\times 2^{19}$	$\times 2^{19} \sim \times 2^{20}$
1	1	$\times 2^{21}$	$\times 2^{21} \sim \times 2^{22}$

ただし, はGCRのCHC=1のときはX0の2倍,CHC=0のときはPLLの発振周波数の周期となります。

### 3.11.2 スタンバイ制御レジスタ(STCR)

スタンバイ動作の制御と発振安定待ち時間の指定を行うレジスタです。

スタンバイ制御レジスタ(STCR)のレジスタ構成

図 3.11-4に,スタンバイ制御レジスタ(STCR)のレジスタ構成を示します。

	bit	7	6	5	4	3	2	1	0	
アドレス	00000481 <sub>H</sub>	STOP	SLEP	HIZX	SRST	OSC1	OSC0	—	—	初期値 000111— <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 3.11-4 スタンバイ制御レジスタ(STCR)のレジスタ構成

以下に,スタンバイ制御レジスタ(STCR)の各ビットの機能を説明します。

【ビット7】:STOP

このビットに"1"を書き込むと内部周辺のクロック停止,内部CPUのクロック停止および発振の停止を行うストップ状態になります。

【ビット6】:SLEP

このビットに"1"を書き込むと内部CPUのクロック停止を行うスリープ状態になります。なお,ビット7とこのビット両方に"1"を書き込んだ場合は,ビット7の方が優先となりますので,ストップ状態になります。

【ビット5】:HIZX

このビットに"1"を書き込んだ状態でストップ状態になるとデバイスの端子をハイインピーダンスにします。

【ビット4】:SRST

このビットに"0"を書き込むとソフトウェアリセット要求を発生します。

【ビット3,2】:OSC1,0

このビットは,発振安定待ち時間を指定するものです。このビットと選択する周期は表 3.11-2のような関係があります。このビットはパワーオンリセットにより初期化され,それ以外のリセット要因では影響を受けません。

表 3.11-2 OSC1,OSC0で指定された発振安定待ち時間

OSC1	OSC0	発振安定待ち時間
0	0	$\times 2^{15}$
0	1	$\times 2^{17}$
1	0	$\times 2^{19}$
1	1	$\times 2^{21}$ [初期値]

ただし, はGCRのCHC=1の場合はX0の2倍,GCRのCHC=0の場合はX0の1倍の周期となります。

【ビット1,0】:(Reserved)

このビットは予約ビットです。読出し値は不定です。

### 3.11.3 DMA要求抑止レジスタ(PDERR)

DMA要求を一時的に抑止して,CPUを動作させるレジスタです。

DMA要求抑止レジスタ(PDERR)のレジスタ構成

図 3.11-5に,DMA要求抑止レジスタ(PDERR)のレジスタ構成を示します。

bit	15	14	13	12	11	10	9	8	初期値
アドレス 00000482 <sub>H</sub>	—	—	—	—	D3	D2	D1	D0	----0000 <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 3.11-5 DMA要求抑止レジスタ(PDERR)のレジスタ構成

以下に,スタンバイ制御レジスタ(STCR)の各ビットの機能を説明します。

【ビット11～8】:D3～D0

これらのビットに"0"以外の値を書き込むと,以降のDMAからCPUへのDMA転送要求を抑止します。以降,これらのビットの内容を"0"にしない限り,DMA転送は行えなくなります。

### 3.11.4 タイムベースタイマクリアレジスタ(CTBR)

タイムベースタイマの内容を0に初期化するためのレジスタです。

タイムベースタイマクリアレジスタ(CTBR)のレジスタ構成

図 3.11-6に、タイムベースタイマクリアレジスタ(CTBR)のレジスタ構成を示します。

	bit	7	6	5	4	3	2	1	0	初期値
アドレス 00000483 <sub>H</sub>		D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX <sub>B</sub>
		W	W	W	W	W	W	W	W	

図 3.11-6 タイムベースタイマクリアレジスタ(CTBR)のレジスタ構成

以下に、タイムベースタイマクリアレジスタ(CTBR)の各ビットの機能を説明します。

【ビット7～0】

これらのビットに連続してA5H, 5AHを書き込むと、5AHの直後にタイムベースタイマを0にクリアします。これらのビットの読出し値は不定です。A5H書込みと5AH書込みの間の時間は制限がありません。

< 注意事項 >

このレジスタを使用してタイムベースタイマをクリアすると、発振安定待ち間隔, ウォッチドッグ周期およびタイムベースを使用している周辺の周期が一時的に変動します。

### 3.11.5 ギア制御レジスタ(GCR)

CPU系,周辺系クロックのギア機能を制御するレジスタです。

ギア制御レジスタ(GCR)のレジスタ構成

図 3.11-7に,ギア制御レジスタ(GCR)のレジスタ構成を示します。

bit	15	14	13	12	11	10	9	8	初期値
アドレス 00000484 <sub>H</sub>	CCK1	CCK0	DBLAK	DBLON	PCK1	PCK0	—	CHC	110011-1 <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 3.11-7 ギア制御レジスタ(GCR)のレジスタ構成

以下に,ギア制御レジスタ(GCR)の各ビットの機能を説明します。

【ビット15,14】:CCK1,0

これらのビットは,CPU系のギア周期を指定するものです。これらのビットと選択する周期は,表 3.11-3のような関係があります。これらのビットはリセットにより初期化されます。

表 3.11-3 CPU系マシンのクロック

CCK1	CCK0	CHC	CPU系マシンのクロック
0	0	0	PLL × 1
0	1	0	PLL × 1/2
1	0	0	PLL × 1/4
1	1	0	PLL × 1/8
0	0	1	原振 × 1/2
0	1	1	原振 × 1/2 × 1/2
1	0	1	原振 × 1/2 × 1/4
1	1	1	原振 × 1/2 × 1/8 [初期値]

PLL:PLLの発振周波数

発振:X0からの入力周波数

【ビット13】:DBLAK

このビットは,クロックダブラの動作状態を示すビットです。読み出し専用であり,書き込みは無視されます。このビットはリセットにより初期化されます。

バス周波数の切替えにはタイムラグがありますが,実際に動作が切り替ったかどうかはこのビットで確認出来ます。

DBLAK	内部:外部の動作周波数
0	1:1で動作中 [初期値]
1	2:1で動作中

【ビット12】:DBLON

このビットは,クロックダブラの動作状態を指定するビットです。このビットはリセットにより初期化されます。

DBLON	内部:外部の動作周波数
0	1:1で動作 [初期値]
1	2:1で動作



【ビット11,10】:PCK1,0

これらのビットは、周辺系のギア周期を指定するものです。これらのビットと選択する周期は表 3.11-4のような関係があります。これらのビットはリセットにより初期化されます。

表 3.11-4 周辺系マシニングロック

PCK1	PCK0	CHC	周辺系マシニングロック (原振:X0からの入力周波数)
0	0	0	PLL × 1
0	1	0	PLL × 1/2
1	0	0	PLL × 1/4
1	1	0	PLL × 1/8
0	0	1	原振 × 1/2
0	1	1	原振 × 1/2 × 1/2
1	0	1	原振 × 1/2 × 1/4
1	1	1	原振 × 1/2 × 1/8 〔初期値〕

PLL:PLLの発振周波数

発振:X0からの入力周波数

なお、CPU系のクロックが25MHzよりも高い周波数で動作させるときは、周辺系クロックは、CPU系クロックの1/2以下の周波数になるように設定してください。周辺系クロックの最大周波数は25MHzです。

#### < 注意事項 >

.....  
CPU系ギアと周辺系ギアを同時に変更したい場合は、一度両方を同じギアに設定した後に、希望のギアに設定してください。変更前のCPU系を周辺系のギアが同じ設定である場合と、片方のギアだけを変更する場合、そして両方を同じギアに設定する場合は、目的の値に一度に設定するのでかまいません。クロックダブラがONになっているときは、GCRの値に関わらずCPU系ギアは固定値となるため、やはり目的の値に一度に設定するのでかまいません。  
.....

#### [プログラム記述例]

```
ldi    #0x484,    r1
ldi    #0x0d,     r0
stb    r0,        @r1                ; CPU:1/1,周辺:1/8
:
ldi    #0x484,    r1
ldi    #0xcd,     r0
stb    r0,        @r1                ; CPU:1/8,周辺:1/8   一度同じ比率に設定
ldi    #0xc5,     r0
stb    r0,        @r1                ; CPU:1/8,周辺:希望の比率に設定
```

## 【ビット9】:(Reserved)

このビットは予約ビットです。常に"1"を書いてください。

## 【ビット8】:CHC

このビットは、基準クロックの選択を設定するビットです。リセットにより初期化されます。

PLL制御レジスタ(PCTR)のVSTPビットが"1"のときには、このビットへの"0"書込みは無視されます。

CHC	クロック選択
0	発振回路の2分周を基準クロックとして使用      [初期値]
1	PLLからの発振出力を基準クロックとして使用

## &lt; 注意事項 &gt;

PCTRのVSTPが"0"のときにストップモードに移行すると、PLLは発振を停止しますが、VSTPは"0"のままとなります。外部割込みなどによりストップモードから復帰したときは、PLLの発振が安定するまでにSTCRのOSC1, OSC0で設定した発振安定待ち時間のほかに約100  $\mu$  秒必要となりますので、それより前にこのビットを"0"に設定しないでください。

ストップモードからの復帰時の手順、内部動作は「3.12.1 ストップ状態」を参照してください。

### 3.11.6 ウォッチドッグリセット発生延期レジスタ(WPR)

ウォッチドッグタイマ用フリップフロップをクリアするレジスタです。このレジスタを用いてウォッチドッグリセットの発生を延期させます。

ウォッチドッグリセット発生延期レジスタ(WPR)のレジスタ構成

図 3.11-8に、ウォッチドッグリセット発生延期レジスタ(WPR)のレジスタ構成を示します。

bit	7	6	5	4	3	2	1	0	初期値
アドレス 00000485 <sub>H</sub>	D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX <sub>B</sub>
	W	W	W	W	W	W	W	W	

図 3.11-8 ウォッチドッグリセット発生延期レジスタ(WPR)のレジスタ構成

以下に、ウォッチドッグリセット発生延期レジスタ(WPR)の各ビットの機能を説明します。

【ビット7～0】:D7～D0

これらのビットに連続してA5H,5AHを書き込むと,5AHの直後にウォッチドッグタイマ用フリップフロップを”0”にクリアし,ウォッチドッグリセットの発生を延期させます。

これらのビットの読出し値は不定です。A5Hと5AHの間の時間は制限がありませんが,表 3.11-5の期間内に両データの書き込みを終えないとウォッチドッグリセットが発生します。ただし,ストップ,スリープ,ホールド中は自動的にクリアを行いますので,これらの条件が発生すると,ウォッチドッグリセットは自動的に延期されます。

表 3.11-5 WT1,WT0で指定されたウォッチドッグの周期

WT1	WT0	ウォッチドッグリセットの発生を抑止するために最低限必要なWPRへの書き込み間隔	WPRへの最終5AH書き込みからウォッチドッグリセットが発生するまでの時間
0	0	$\times 2^{15}$	$\times 2^{15} \sim \times 2^{16}$
0	1	$\times 2^{17}$	$\times 2^{17} \sim \times 2^{18}$
1	0	$\times 2^{19}$	$\times 2^{19} \sim \times 2^{20}$
1	1	$\times 2^{21}$	$\times 2^{21} \sim \times 2^{22}$

ただし、はGCRのCHC=1のときはX0の2倍,CHC=0のときはPLLの発振周波数となります。

### 3.11.7 PLL制御レジスタ(PCTR)

PLLの発振を制御するレジスタです。このレジスタは,GCRのCHCが"1"のときのみ設定を変更可能です。

PLL制御レジスタ(PCTR)のレジスタ構成

図 3.11-9に,PLL制御レジスタ(PCTR)のレジスタ構成を示します。

bit	15	14	13	12	11	10	9	8	初期値
アドレス 00000488 <sub>H</sub>	SLCT1	SLCT0	—	—	VSTP	—	—	—	00—0—B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 3.11-9 PLL制御レジスタ(PCTR)のレジスタ構成

以下に,PLL制御レジスタ(PCTR)の各ビットの機能を説明します。

【ビット15,14】:SLCT1,0

このビットは,PLLの通倍率を制御するものです。電源投入時のみ初期化されます。

このビットの設定はGCRのCHCを"0"にしたときの内部動作周波数になります。

なお,2通倍の設定は,原振が12.5MHz以下の場合に限ります。

表 3.11-6に,内部動作周波数を示します。

表 3.11-6 内部動作周波数

SLCT1	SLCT0	内部動作周波数(原振:12.5MHzの場合)
0	0	6.25MHz 動作 [初期値]
0	1	12.5MHz 動作 (1通倍)
1	1	25.0 MHz 動作 (2通倍) *

\*:原振12.5MHz以下の場合のみ設定可能

< 注意事項 >

.....  
 原振が12.5MHzを越える場合には,2通倍の設定は行わないでください。  
 .....

【ビット13,12】:(Reserved)

このビットは予約ビットです。常に"0"を書いてください。

【ビット11】:VSTP

このビットは,PLLの発振を制御するものです。電源投入時および外部リセットにより初期化されます。なお,PLLを停止して使用する場合は,リセット解除後に毎回停止させる必要があります。

VSTP	PLL動作
0	発振 [初期値]
1	発振停止

< 注意事項 >

.....  
ストップモードに移行すると,このビットの設定に関わらず,PLLは停止します。  
.....

【ビット10,9,8】(Reserved)

このビットは予約ビットです。常に"0"を書いてください。

### 3.11.8 ウォッチドッグ機能

ウォッチドッグ機能は、プログラムの暴走状態を検出する機能です。プログラムの暴走などによりウォッチドッグリセット延期レジスタへのA5H,5AH書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。

ウォッチドッグ制御部のブロックダイアグラム

図 3.11-10に、ウォッチドッグ制御部のブロックダイアグラムを示します。

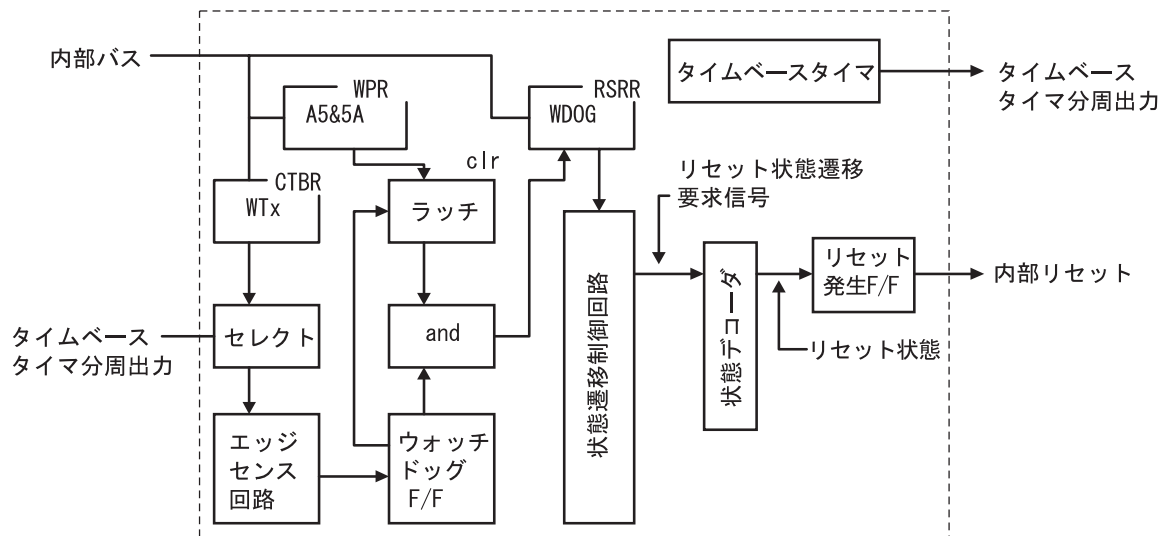


図 3.11-10 ウォッチドッグ制御部ブロックダイアグラム

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、ウォッチドッグ制御レジスタ(WTCR)への書込みにより動作を開始します。このとき、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。インターバル時間の設定は、最初の手書き込みで設定した時間だけが有効となり、それ以降の設定は無視されます。

〔例〕

```
LDI:8  #00000010b,R1      ; WT1,0=10
LDI:20  #WTCR,R2
STB    R1,@R2              ; ウォッチドッグ起動
```

リセット発生延期方法

一度、ウォッチドッグタイマを起動すると、プログラムで定期的にウォッチドッグリセット延期レジスタ(WPR)に対して定期的にA5H,5AHを書き込む必要があります。ウォッチドッグリセット用フリップフロップは、タイムベースタイマの選択したタップの立下りを記憶し、2回目の立下りでこのフリップフロップがクリアされていないと、リセットが生成されます。

図 3.11-11に、ウォッチドッグタイマの動作タイミングを示します。

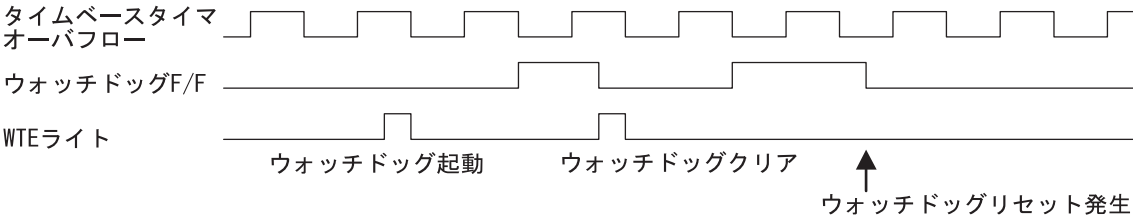


図 3.11-11 ウォッチドッグタイマの動作

< 注意事項 >

- 最初のA5Hと次の5AHの書き込み間隔の規定はありません。ウォッチドッグの延期は、2回の5AHの書き込み間隔がWTビットで指定した時間内であり、かつそれらの間に1回のA5Hの書き込みがあることでのみ行えます。
- 最初のA5Hの後の書き込みが5AH以外であったときは、最初のA5Hの書き込みがあったことを無効にします。このため、再度A5Hの書き込みを行う必要があります。

タイムベースタイマ

タイムベースタイマは、ウォッチドッグタイマへのクロック供給および発振安定時間待ち用のタイマとして使用されています。動作クロックはGCRのCHC=1のときはX0の2倍、GCRのCHC=0のときはPLLの発振周波数の周期となります。

また、DRAMのリフレッシュのときのカウント値には、RFCRでこのタイムベースタイマの値をカウントクロックとして使用します。

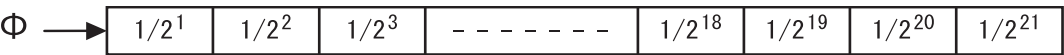


図 3.11-12 タイムベースタイマの構成

### 3.11.9 ギア機能

クロックを間引いて供給する機能です。CPU用と周辺用の2種の独立な回路があり、異なるギア比でもCPUと周辺の間でのデータの送受を行うことができます。さらに、原クロックの選択として、PLLからのクロックと同周期のクロックを使用するかまたは2分周回路を通したクロックを使用するかの指定が行えます。

ギア制御部のブロックダイアグラム

図 3.11-13に、ギア制御部のブロックダイアグラムを示します。

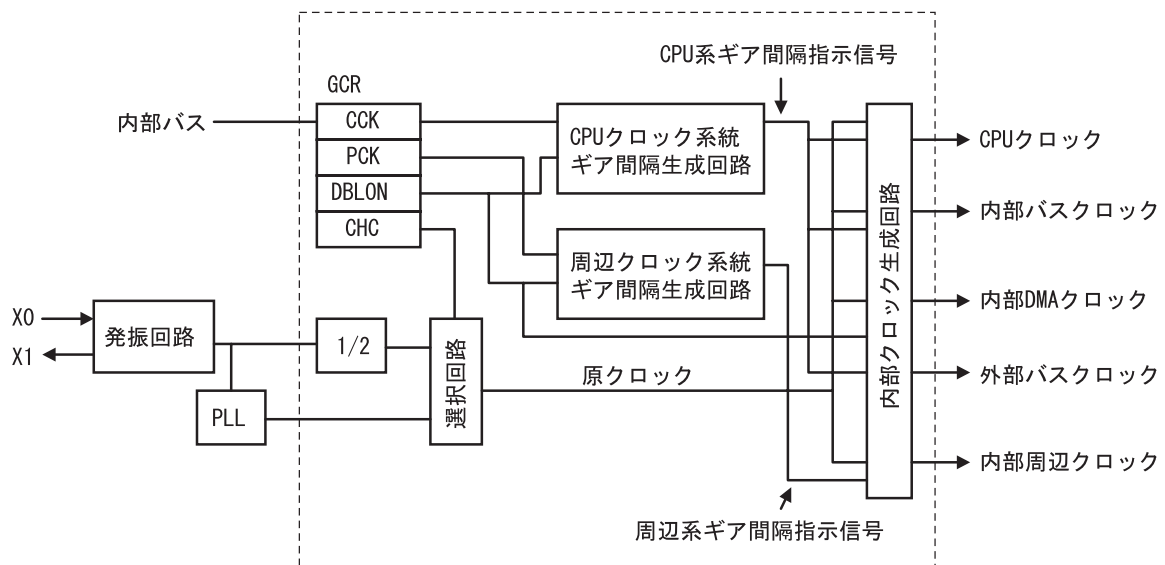


図 3.11-13 ギア制御部ブロックダイアグラム



## ギア機能の設定

CPUのクロックの制御であればギア制御レジスタ(GCR)のCCK1,CCK0ビットを,周期のクロックの制御であれば同レジスタのPCK1,PCK0ビットを,それぞれ望む値に設定することで,希望するギア比に設定することができます。

〔例〕

```
LDI:20 #GCR,R2
LDI:8  #11111110b,R1      ; CCK=11,PCK=11,CHC=0
STB   R1,@R2              ; CPU clock=1/8f, Peripheral clock=1/8f, f=direct
LDI:8  #01111010b,R1      ; CCK=01,PCK=10,CHC=0
STB   R1,@R2              ; CPU clock=1/2f, Peripheral clock=1/4f, f=direct
LDI:8  #00111010b,R1      ; CCK=00,PCK=10,CHC=0
STB   R1,@R2              ; CPU clock=f, Peripheral clock=1/4f, f=direct
LDI:8  #00110010b,R1      ; CCK=00,PCK=00,CHC=0
STB   R1,@R2              ; CPU clock=f, Peripheral clock=f, f=direct
LDI:8  #10110010b,R1      ; CCK=10,PCK=00,CHC=0
STB   R1,@R2              ; CPU clock=1/4f, Peripheral clock=f, f=direct
```

また,ギア制御レジスタ(GCR)のCHCビットを"1"にすると2分周回路の出力を原クロックとして選択し,"0"にすると発振回路からのクロックと同周期のクロックをそのまま使用します。原クロックを切り替えるために,CPU系と周辺系は同時に変化します。

〔例〕

```
LDI:8  #01110001b,R1      ; CCK=01,PCK=00,CHC=1
LDI:20 #GCR,R2
STB   R1,@R2              ; CPU clock=1/2f, Peripheral clock=f, f=1/2xtal
LDI:8  #00110011b,R1      ; CCK=00,PCK=00,CHC=1
STB   R1,@R2              ; CPU clock=f, Peripheral clock=f, f=1/2xtal
LDI:8  #00110010b,R1      ; CCK=00,PCK=00,CHC=0
STB   R1,@R2              ; CPU clock=f, Peripheral clock=f, f=direct
```

図 3.11-14に,クロック選択のタイミングチャートを示します。

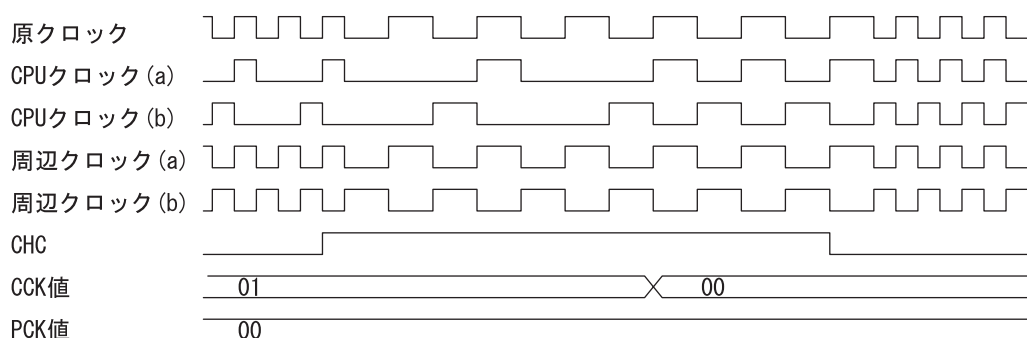


図 3.11-14 クロック選択のタイミングチャート

### 3.11.10 リセット要因保持

直前に発生したリセットの要因を保持します。読出しにより全フラグを”0”にします。

読み出さない限り、一度立った要因フラグは消えません。

リセット要因保持回路のブロックダイアグラム

図 3.11-15に、リセット要因保持回路のブロックダイアグラムを示します。

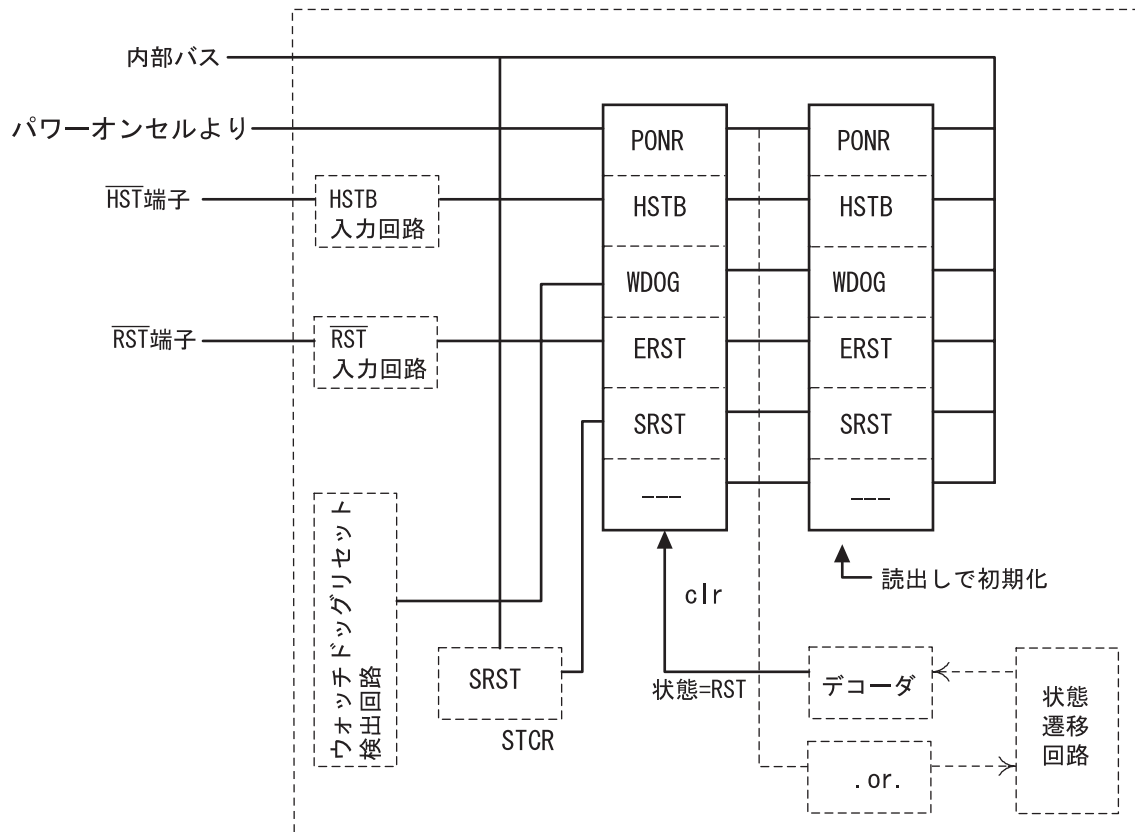


図 3.11-15 リセット要因保持回路のブロックダイアグラム

### リセット要因保持の設定

この機能を使用するにあたっての特別な設定は不要です。リセットエントリアドレスに置くプログラムの始めの方に、リセット要因レジスタの読出しと、適切なプログラムへ分岐する命令を置くようにします。

〔例〕

```
RESET-ENTRY
    LDI:20    #RSRR, R10
    LDI:8     #10000000B, R2
    LDUB     @R10, R1          ; GET RSRR VALUE INTO R1
    MOV      R1, R10           ; R10 USED AS A TEMPORARY REGISTER
    AND      R2, R10           ; WAS PONR RESET?
    BNE      PONR-RESET
    LSR      #1, R2            ; POINT NEXT BIT
    MOV      R1, R10           ; R10 USED AS A TEMPORARY REGISTER
    AND      R2, R10           ; WAS HARDWARE STANDBY RESET?
    BNE      HSTB-RESET
    LSR      #1, R2            ; POINT NEXT BIT
    MOV      R1, R10           ; R10 USED AS A TEMPORARY REGISTER
    AND      R2, R10           ; WAS WATCH DOG RESET?
    BNE      WDOG-RESET
    :
```

#### < 注意事項 >

- PONRビットが<sup>や</sup>“1”のときは、それ以外のビットの内容は不定として扱ってください。また、リセット要因をチェックする必要がある場合には、必ず先頭にパワーオンリセット確認の命令を置いてください。
- パワーオンリセットのチェック以外のリセット要因のチェックは、任意の位置にすることができます。置いた順に優先度が決まります。

### 3.11.11 DMA抑止

FRファミリはDMA転送中に、より優先度の高い割り込み要因が発生するとDMA転送を中断して該当割り込みルーチンへ分岐します。この機構は、割り込み要求がある限り有効ですが、割り込み要因をクリアすると抑止機構が動かなくなり、割り込み処理ルーチン内でDMA転送を再開します。

このため、DMA転送を中断するレベルの割り込み要因の処理ルーチン内で、割り込み要因クリア後のDMA再転送開始を抑止したいときは、DMA抑止機能を使用します。DMA抑止機能は、DMA抑止レジスタに”0”以外の値を書くことで起動し、”0”を書くことで停止します。

DMA抑止回路のブロックダイヤグラム

図 3.11-16に、DMA抑止回路のブロックダイヤグラムを示します。

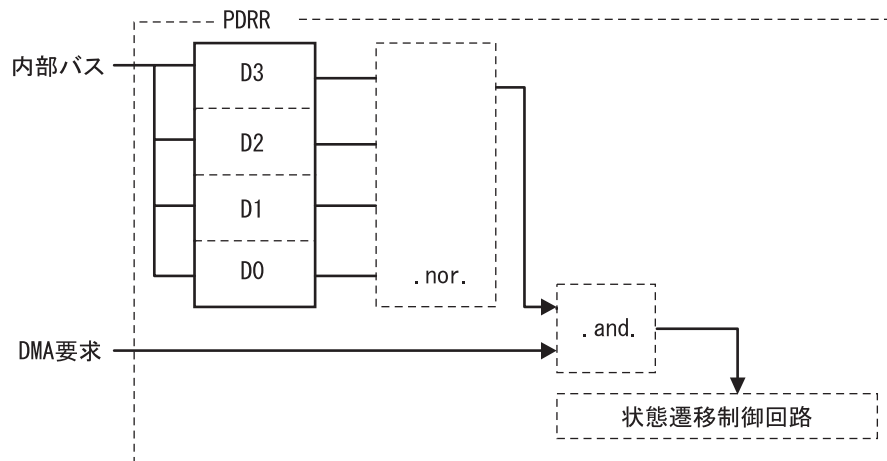


図 3.11-16DMA抑止回路のブロックダイヤグラム

## DMA抑止の設定

この機能は、主として割込み処理ルーチン内で使用します。

割込み処理ルーチンで、割込み要因をクリアする前にDMA抑止レジスタの内容を一つ増加させます。

このようにすると、以降、DMA転送は行いません。割込み処理への対応を行った後、復帰する前に、DMA抑止レジスタの内容を一つ減少させます。もし、多重割込みであれば、DMA抑止レジスタの内容はまだ"0"にならないため、引き続いてDMA転送は抑止されます。また、多重割込みでなければ、DMA抑止レジスタの内容は"0"になるため、その後すぐにDMA要求を有効にします。

## [例]

INT-ENTRY

```

LDI:20    #PDRR, R10
LD         @R10, R1          ; GET PDRR VALUE INTO R1
ADD        #1, R1
ST         R1, @R10          ; PDRR:=PDRR+1, DMA disabled
LDI:20    #int-REG, R10      ; int occurred with int-REG
LDI:8      #10H, R1          ; example, int-flag=#10h
ST         R1, @R10          ; CLEAR int-REQ, (but still DMA disabled)
:
; interrupt execute routine
:
LDI:20    #PDRR, R10
LD         @R10, R1          ; GET PDRR VALUE INTO R1
ADD2       #-1, R1
ST         R1, @R10          ; PDRR:=PDRR-1, DMA may be enabled
RETI

```

## &lt; 注意事項 &gt;

レジスタのビット数は4ビットですので、15レベルを超える多重割込みで、この機能を使用することができません。DMAタスクの優先順位は、必ずほかの割込みレベルより15レベルは上においてください。

### 3.11.12 クロックダブラ機能

内部動作周波数の上昇に伴い外部バスタイミングが厳しくなるのを避けるため、外部バスと内部動作の周波数を1:2にすることができます。

#### クロックダブラ機能の起動

クロックダブラ機能は、GCRのDBLONを"1"に設定することにより有効になります。DBLONが"1"に設定されると、C-BUSの全アクセスが終了するのを待って、外部バスクロックの切替えを行います。そのため、切り替るまでに若干のタイムラグがありますが、GCRのDBLAKの値により、切替りタイミングを知ることができます。また、クロックダブラ機能を起動させることにより、CPU系クロックのギアはGCRの設定に関わらず1/1になります。

このデバイスは外部バスの動作周波数として、原発振の2倍の周波数までが設定可能になっています。そのため、クロックダブラ機能をONにするときは、次のように設定してください。

[例]

```
DOUBLER-ON
    LDI :20      #GCR,R0
    BORL        #0001B,@R0      ;2分周系に切替え(CHC=1)
    BORH        #0001B,@R0      ;クロックダブラON (DBLON=1)
LOOP
    BTSTH       #0010B,@R0      ;DBLAKを確認
    BEQ         LOOP            ;DBLAKが1になるまでループ
    BANDL       #1110B,@R0      ;PLL系に切替え(CHC=0)
```

#### クロックダブラ機能の停止

クロックダブラ機能はGCRのDBLONを0に設定することにより無効になります。

同時にCPU系クロックのギアは1/1からギア制御レジスタ(GCR)のCCKビットの設定に戻ります。

[例]

```
DOUBLER-OFF
    LDI :20      #GCR,R0
    BORL        #0001B,@R0      ;2分周系に切替え(CHC=1)
    BANDH       #1110B,@R0      ;クロックダブラOFF(DBLON=0)
```

また、クロックダブラ機能をOFFにした後に、PLLのクロックを使用する場合は、次のように設定を行ってください。

[例]

```
DOUBLER-OFF
    LDI :20      #GCR,R0
    BORL        #0001B,@R0      ;2分周系に切替え(CHC=1)
    BANDH       #1110B,@R0      ;クロックダブラOFF(DBLON=0)
    LDI :20      #PCTR,R1
    LDI :8       #01000000B,R2
    STB         R2,@R1          ;PLL=25MHz
    BANDL       #1110B,@R0      ;PLL系に切替え(CHC=0)
```

## クロックダブラ機能のon/offに伴う注意事項

クロックダブラ機能をon/offさせると、内部クロックにデッドサイクルを生じることがあります。

デッドサイクルが生じた場合は、タイマでの時間測定やUARTの転送中には誤差として現れます。

## クロックダブラ機能のon/offによる動作周波数の組合せ

PLL制御レジスタ(PCTR)のSLCT1,SLCT0ビットと、ギア制御レジスタ(GCR)の設定の組合せにより、このデバイスの動作周波数は表 3.11-7のようになります。(原発振12.5MHzを使用した場合の例)

表 3.11-7 クロックダブラ機能のon/offによる動作周波数の組合せ

GCR		PLLの発振 周波数(MHz)	クロック ダブラ	内部動作 周波数(MHz)	外部バス 周波数(MHz)	備 考
CHC	ギア					
2分周	1/1		OFF	6.25	6.25	
	1/2		OFF	3.13	3.13	
	1/4		OFF	1.56	1.56	
	1/8		OFF	0.78	0.78	初期値
	*1		ON	6.25	3.13	
PLL *3	-	50.0	OFF	50.0	50.0	設定禁止
	1/1	25.0	OFF	25.0	25.0	
	1/2	25.0	OFF	12.5	12.5	
	1/4	25.0	OFF	6.25	6.25	
	1/8	25.0	OFF	3.13	3.13	
	1/1	12.5	OFF	12.5	12.5	
	1/2	12.5	OFF	6.25	6.25	
	1/4	12.5	OFF	3.13	3.13	
	1/8	12.5	OFF	1.56	1.56	
	*1	50.0	ON	50.0	25.0	設定禁止
	*1	25.0	ON	25.0	12.5	
	*1	12.5	ON	12.5	6.25	

\*1: どのように設定してあっても1/1固定となります。

\*2: クロックダブラをOFFにするときは、2分周系に切替えてからOFFにしてください。

\*3: PLLの発振周波数を変更するときは、2分周系に切替える必要があります。

### 3.11.13 PLLのクロック設定例

PLLのクロック設定例とアセンブラのソース例について説明します。

PLLのクロック設定例

図 3.11-17は,PLLを使用した25MHz動作への,切替え手順の例です。(原発振12.5 MHzの場合)

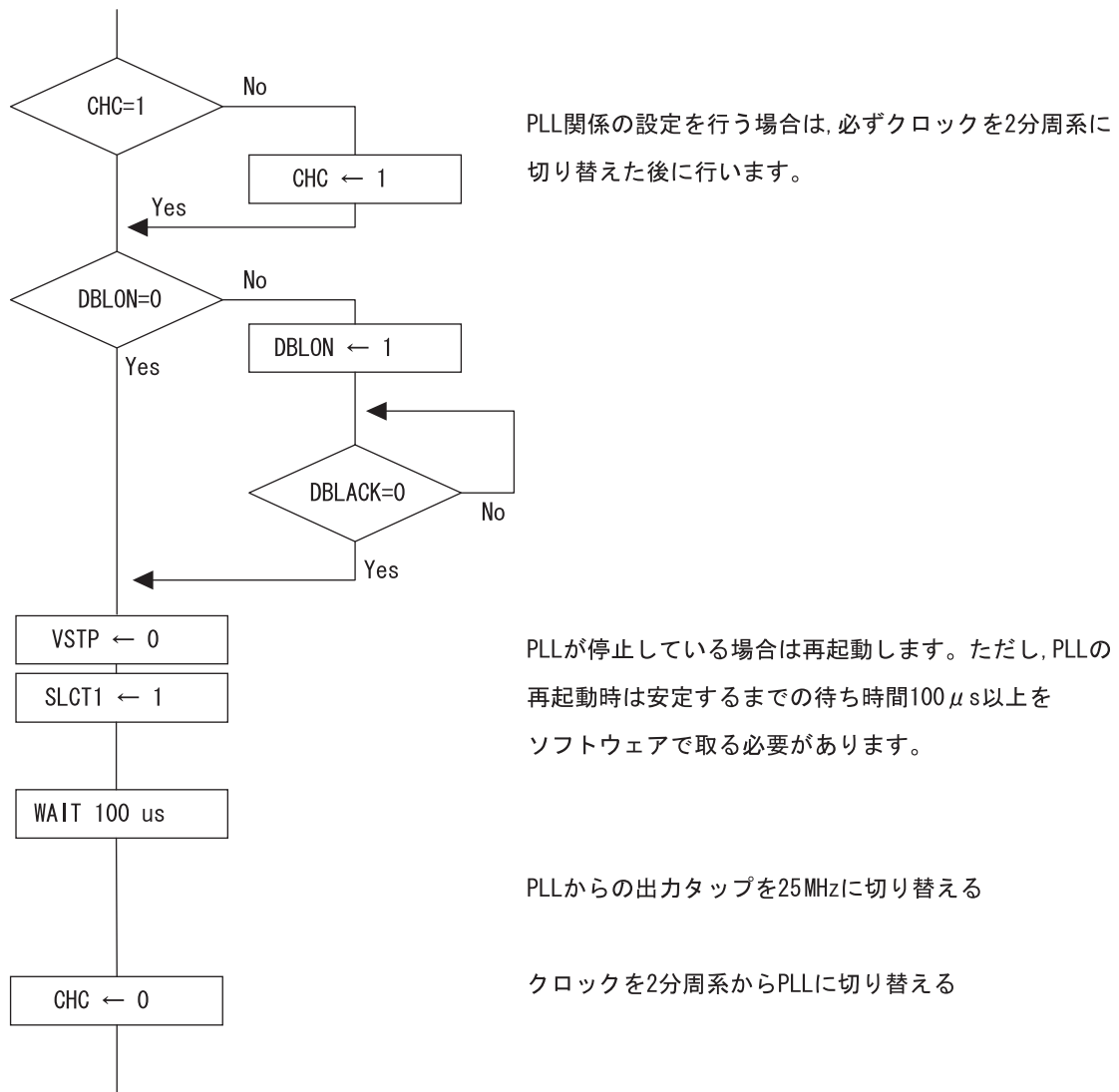


図 3.11-17 PLLのクロック設定例

#### < 注意事項 >

- ここに記載したDBLON,VSTP,SLCT0の各ビットの設定順序は特にありません。
- 周辺部の動作周波数が25MHzを超えないようにしてください。
- PLLのVCOの再起動時には,安定するまでの待ち時間100 $\mu$ s以上を,ソフトウェアで必ず取ってください。また,キャッシュON/OFFなどで待ち時間の不足を起こさないように注意してください。
- SLCTビットを書き換えて逡倍率を変更する場合も,PLLが安定するまでの待ち時間100 $\mu$ s以上を,ソフトウェアで必ず取ってください。



クロック系参考図

図 3.11-18に、クロック系参考図を示します。

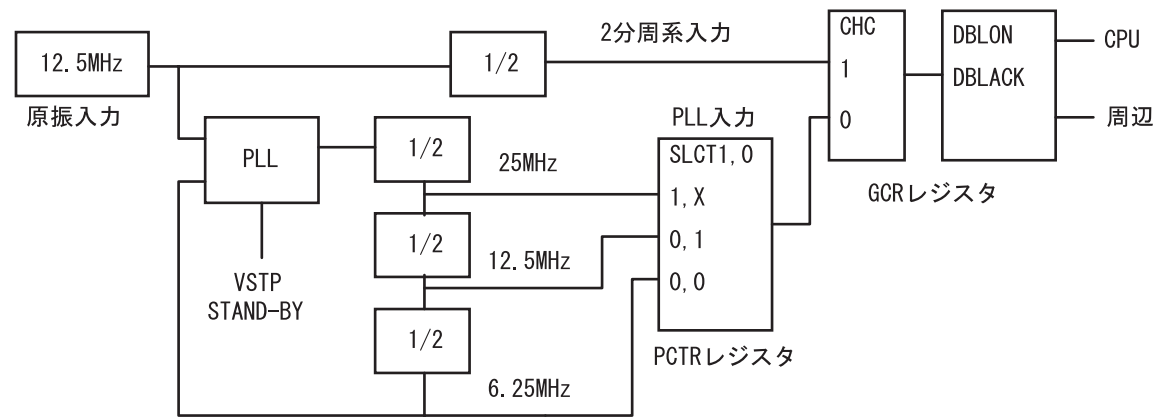


図 3.11-18 クロック系参考図

## アセンブラのソース例

```

; *****
;
; PLL Sample Program
; *****
; Load Setting Data
    ldi:20    #GCR, R0
    ldi:20    #PCTR, R1
    ldi:8     #GCR_MASK, R2        ; GCR_MASK = 0000 0001 b
    ldi:8     #PCTR_MASK, R3       ; PCTR_MASK = 0000 1000 b
    ldub      @R0, R4              ; read GCR register
    ldub      @R1, R5              ; read PCTR register
    st        PS, @-R15            ; push processor status
    stilm     #0x0                 ; disable interrupt
    ;
    and       R4, R2
    beq       CHC_0
    bra       CHC_1
CHC_0:
    borl      #0001B, @r0          ; to 1/2 clock @r0=GCR register
CHC_1:
    call      VCO_RUN
PLL_SET_END:
    ld        @R15+, PS            ; pop processor status
; *****
; VCO Setting
; *****
VCO_RUN:
    st        R3, @-R15            ; push R3
    ldi:8     #PCTR_MASK, R3       ; PCTR_MASK = 0000 1000 b
    and       R5, R3              ; PCTR->VSTP=1 ?
    beq       LOOP_100US_END      ; if VSTP = 0 return
    bandl     #0111B, @r1         ; set VSTP = 0
    st        R2, @-R15            ; push R2 for Loop counter
    ldi:20    #0x15E, R2          ; wait 100 mS
WAIT_100US:
    add2      #(-1), R2            ; 100us = 160ns(6.25MHz) * 7 * 100 (2BC)cycle
    bne       WAIT_100US ;        ; 2BCh/2 = 15Eh (if cache on)
LOOP_100US_END:
    ld        @R15+, R2            ; Pop R2
    ld        @R15+, R3            ; Pop R3
    ret

```

## 3.12 スタンバイモード(低消費電力メカニズム)

---

スタンバイモードには、ストップ状態とスリープ状態があります。

---

### ストップ状態の概要

ストップ状態とは、以下の状態をいい消費電力を最低限に抑えることができます。

- すべての内部クロックを停止
- 発振回路の動作停止

ストップ状態への移行は、以下の方法で行います。

- 命令による、スタンバイ制御レジスタ(STCR)への書込み
- $\overline{\text{HST}}$  端子へのLレベルの印加

また、復帰は、以下の方法で行います。

- 割込み要求(ただし、ストップ中でも割込み要求が発生可能な周辺に限ります。) 注: INT4とINT5は、ストップ状態からの復帰に使用できません。
- $\overline{\text{RST}}$  端子へのLレベルの印加
- $\overline{\text{HST}}$  端子へのLレベルからHレベルへの印加

ストップ状態では、すべての内部クロックが停止しますので、復帰用の割込みを生成できるもの以外の内蔵周辺は停止状態になります。

### スリープ状態の概要

#### 概要

スリープ状態とは、以下の状態をいい、CPU動作不要な状況での消費電力をある程度抑えることができます。

- CPUクロックと内部バスクロックが停止

スリープ状態への移行は、以下の方法で行います。

- 命令による、スタンバイ制御レジスタ(STCR)への書込み

また、復帰は、以下の方法で行います。

- 割込み要求  
注: INT4とINT5は、ストップ状態からの復帰に使用できません。
- リセット要因の発生

スリープ状態では、内部DMAクロックと周辺クロックは動作しますので、これら二つのクロックを使用しているすべての内蔵周辺の割込みで解除ができます。

## スタンバイモードの動作一覧

表 3.12-1に,スタンバイモード時の動作一覧を示します。

表 3.12-1 スタンバイモードの動作一覧

動作状態	遷移条件	発振器	内部クロック		周辺	端子	解除方法
			CPU/ 内部バス	DMA/ 周辺			
ラン						動作	
スリープ	STCRの SLEP=1		X			動作	
ストップ	STCRの STOP=1	X	X	X	X	*	
ハードウェア スタンバイ	$\overline{\text{HST}}=0$	X	X	X	X	Hi-Z	$\overline{\text{HST}}=1$

:動作 X:停止

\*:STCRのHIZX="0"で前の状態を保持します。HIZX="1"でHi-Zになります。

## &lt; 注意事項 &gt;

リセット:  $\overline{\text{RST}}="0"$

スタンバイ制御レジスタ(STCR)のSRSTビット="0"

ウォッチドッグリセット

パワーオンリセット

## ストップおよびスリープに入れるプログラムの配置アドレス

ストップおよびスリープに入れるプログラムは,C-bus上のROMまたは外部メモリのアドレス領域に置いてください。C-bus上のROMには置かないでください。

### 3.12.1 ストップ状態

図 3.12-1に、ストップ制御部のブロックダイアグラムを示し、ストップ状態への移行とストップ状態からの復帰について説明します。

ストップ制御部のブロックダイアグラム

図 3.12-1に、ストップ制御部のブロックダイアグラムを示します。

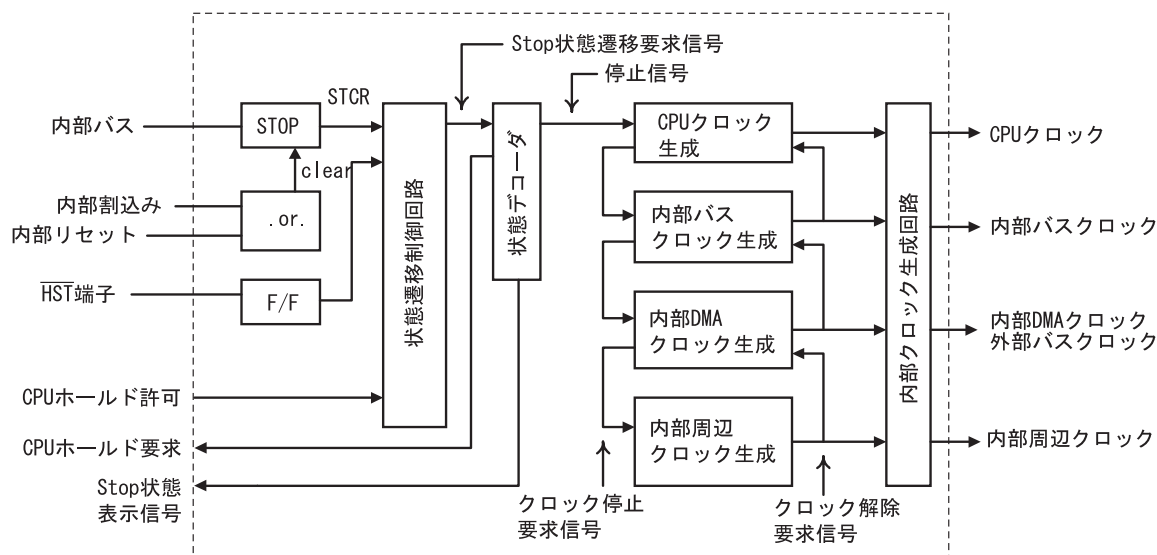


図 3.12-1 ストップ制御部ブロックダイアグラム

ストップ状態への移行

命令を用いてストップ状態にする方法

ストップ状態にイれるためには、スタンバイ制御レジスタ(STCR)のビット7へ"1"を書き込みます。

ストップ要求を発行したのち、CPUが内部バスを使用していない状態になったあと、以下の順にクロックを停止します。

CPUクロック    内部バスクロック    内部DMAクロック    内部周辺クロック

発振回路の停止は、内部周辺クロック停止と同時に行います。

< 注意事項 >

命令を用いてストップ状態にするためには、必ず下記ルーチンを使用してください。

- STCRへ書き込む前には、GCRのCCK1, CCK0とPCK1, PCK0の組に同じ値を設定し、CPU系クロックと周辺系クロックのギア比を同一にしてください。
- GCRのCHCビットが"0"(PLLで動作中)の状態ですトップ状態にしないでください。ストップ状態にイれるときは、GCRのCHCビットを"1"(2分周系を選択)にしてクロックを切替えてからストップ状態に遷移させてください。
- STCRへの書き込み直後は、最低6個の連続するNOP命令が必要です。

[設定方法: ギア最高速の場合の例]

```

LDI:20    #GCR,R0
LDI:8     #00000011b,R1      ; CHC=1, CPU=周辺ギア比
STB       R1,@R0             ; DBLON=0

loop
BTSTH     #0010b,@R0         ;
BNE        loop              ; DBLAK= 0 になるまで待つ

LDI:20    #STCR,R0
LDI:8     #10010000b,R1      ; STOP=1
STB       R1@R0
NOP
NOP
NOP
NOP
NOP
NOP
NOP

```

$\overline{\text{HST}}$  端子を用いてストップ状態にする方法

ストップ状態にいれるためには,  $\overline{\text{HST}}$  端子にLレベルを印加します。

$\overline{\text{HST}}$  端子にLレベルを印加したのち,CPUが内部バスを使用していない状態になったあとで, 次の順にクロックを停止します。

CPUクロック      内部バスクロック      内部DMAクロック      内部周辺クロック

発振回路の停止は内部周辺クロック停止と同時に行います。

#### < 注意事項 >

リセット中( $\overline{\text{RST}}=\text{L}$ )は,  $\overline{\text{HST}}$  端子をLレベルにしてもストップには入りません。また,  $\overline{\text{HST}}$  端子をLレベルのままリセットを解除した場合もストップには入りません。一度,  $\overline{\text{HST}}$  端子をHレベルに戻した以降,Lレベルにすることによりストップに入ることができます。同様に,  $\overline{\text{HST}}$  端子をLレベルのままパワーオンした場合もパワーオンリセット解除後にストップには入りません。

### ストップ状態からの復帰

ストップ状態からの復帰は割込みまたはリセットの発生で行うことができます。

#### 割込みによる復帰

周辺の機能に付随する割込み許可ビットが有効であれば、周辺の割込みの発生によりストップ状態から復帰します。ストップ状態から通常動作状態へは、以下の手順で行います。

割込み発生      発振回路動作再開      発振安定待ち      安定後、内部周辺クロック供給再開      内部DMAクロック供給再開      内部バスクロック供給再開      内部CPUクロック供給再開

発振安定待ち時間後のプログラム実行は、以下のようになります。

- 発生した割込みのレベルがCPUのILM, Iフラグで許可されている場合
  - レジスタ退避を行った後に、割込みベクタを取り込んで割込み処理ルーチンより実行
- 発生した割込みのレベルがCPUのILM, Iフラグで禁止されている場合
  - ストップ状態に入れた命令以降の命令より実行

#### RST 端子による復帰

ストップ状態からの通常動作状態へは、以下の手順で行います。

RST 端子へのLレベル印加      内部リセット発生      発振回路動作再開      発振安定待ち      安定後、内部周辺クロック供給再開      内部DMAバスクロック供給再開      内部バスクロック供給再開      内部CPUクロック供給再開      リセットベクタ取込み      リセットエントリアドレスより命令の実行再開

#### < 注意事項 >

- ・周辺からの割込み要求がすでに発生している場合は、ストップ状態にならず、書込みは無視されます。
- ・パワーオンリセット以外では発振安定待ち時間中のすべての内部クロック供給は行いません。パワーオンリセットでは内部状態の初期化を行う必要があるため、すべての内部クロックの供給を行います。
- ・C-bus RAM内プログラムからストップ状態に移行した場合、割込みによってではなく、リセットによって復帰するようにしてください。

$\overline{\text{HST}}$  端子による復帰

ストップ状態からの通常動作状態へは、以下の手順で行います。

$\overline{\text{HST}}$  端子へのHレベル印加    内部リセット発生    発振回路動作再開    発振安定待ち    安定後、内部周辺クロック供給再開    内部DMAクロック供給再開    内部バスクロック供給再開    内部CPUクロック供給再開    リセットベクタ取り込み    リセットエントリアドレスより命令の実行再開

< 注意事項 >

- ・周辺からの割込み要求がすでに発生している場合は、ストップ状態にならず、書込みは無視されます。
- ・パワーオンリセット以外では、発振安定待ち時間中はすべての内部クロック供給は行いません。パワーオンリセットでは内部状態の初期化を行う必要があるため、すべての内部クロックの供給を行います。
- ・ストップ状態から  $\overline{\text{HST}}$  端子を”L”レベルにすると、直ちにストップ状態が解除されて所定の発振安定待ち時間が取られます。発振安定待ち時間経過後、 $\overline{\text{HST}}$  端子が”L”レベルであると、 $\overline{\text{HST}}$  端子によるストップ状態へと遷移します。



## 3.12.2 スリープ状態

図 3.12-2に、スリープ制御部のブロックダイアグラムを示し、スリープ状態への移行とスリープ状態からの復帰について説明します。

スリープ制御部のブロックダイアグラム

図 3.12-2に、スリープ制御部のブロックダイアグラムを示します。

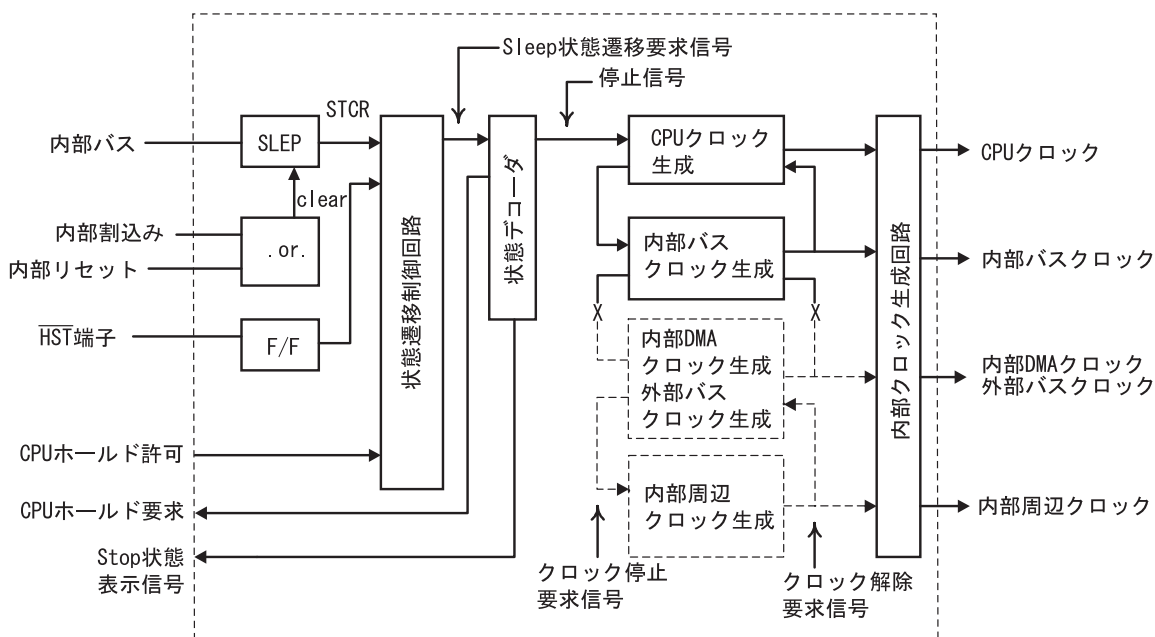


図 3.12-2 スリープ制御部ブロックダイアグラム

### スリープ状態への移行

スリープ状態にいれるためには、STCRのビット7へ”0”，ビット6へ”1”を書き込みます。

スリープ要求を発行したのち、CPUが内部バスを使用していない状態になったあと、以下の順にクロックを停止します。

CPUクロック    内部バスクロック

### < 注意事項 >

スリープ状態にするためには、必ず下記ルーチンを使用してください。

- ・ STCRへ書き込む前には、GCRのCCK1, CCK0とPCK1, PCK0ビットの組に同じ値を設定し、CPU系クロックと周辺系クロックのギア比を同一にしてください。
- ・ GCRのCHCビットに任意です。
- ・ STCRへの書き込み直後には、最低6個の連続するNOP命令が必要です。

[設定方法: ギア最高速の場合の例]

```
LDI :20    #GCR,R0
LDI :8     #00000011b,R1      ; CHC=1,CPU=周辺ギア比
STB       R1,@R0              ; DBLON=0の場合
LDI :20    #STCR,R0
LDI :8     #01010000b,R1      ; SLEP=1
STB       R1,@R0
NOP
NOP
NOP
NOP
NOP
NOP
```

#### スリープ状態からの復帰

スリープ状態からの復帰は割込みまたはリセットの発生で行うことができます。

##### 割込みによる復帰

周辺の機能に付随する割込み許可ビットが有効であれば、周辺の割込みの発生によりスリープ状態から復帰します。スリープ状態から通常動作状態へは、以下の手順で行います。

割込み発生      内部バスクロック供給再開      内部CPUクロック供給再開

クロック供給後のプログラム実行は、以下ようになります。

- 発生した割込みのレベルがCPUのILM,Iフラグで許可されている場合
  - レジスタ退避を行った後に、割込みベクタを取込んで割込み処理ルーチンより実行
- 発生した割込みのレベルがCPUのILM,Iフラグで禁止されている場合
  - スリープ状態に入れた命令以降の命令より実行

##### リセット要求による復帰

スリープ状態からの通常動作状態へは、以下の手順で行います。

内部リセット発生      内部バスクロック供給再開      内部CPUクロック供給再開  
 リセットベクタ取込み      リセットエントリアドレスより命令の実行再開

##### HST 端子による復帰

スリープ状態からの通常動作状態へは、以下の手順で行います。

HST 端子へのLレベル印加      ハードウェアスタンバイ状態へ遷移      HST 端子へのHレベル印加      内部リセット発生      発振回路動作再開      発振安定待ち      安定後、内部周辺クロック供給再開      内部DMAクロック供給再開  
 内部バスクロック供給再開      内部CPUクロック供給再開      リセットベクタ取り込み      リセットエントリアドレスより命令の実行再開

#### <注意事項>

- ・スリープモード中のDMA転送動作は使用できません。スリープモードにする前に、必ずDMA転送動作を動作禁止に設定してください。
- ・周辺からの割込み要求がすでに発生している場合は、スリープ状態になりません。

### 3.12.3 スタンバイモードの状態遷移

図 3.12-3にスタンバイモードの状態遷移図を示します。

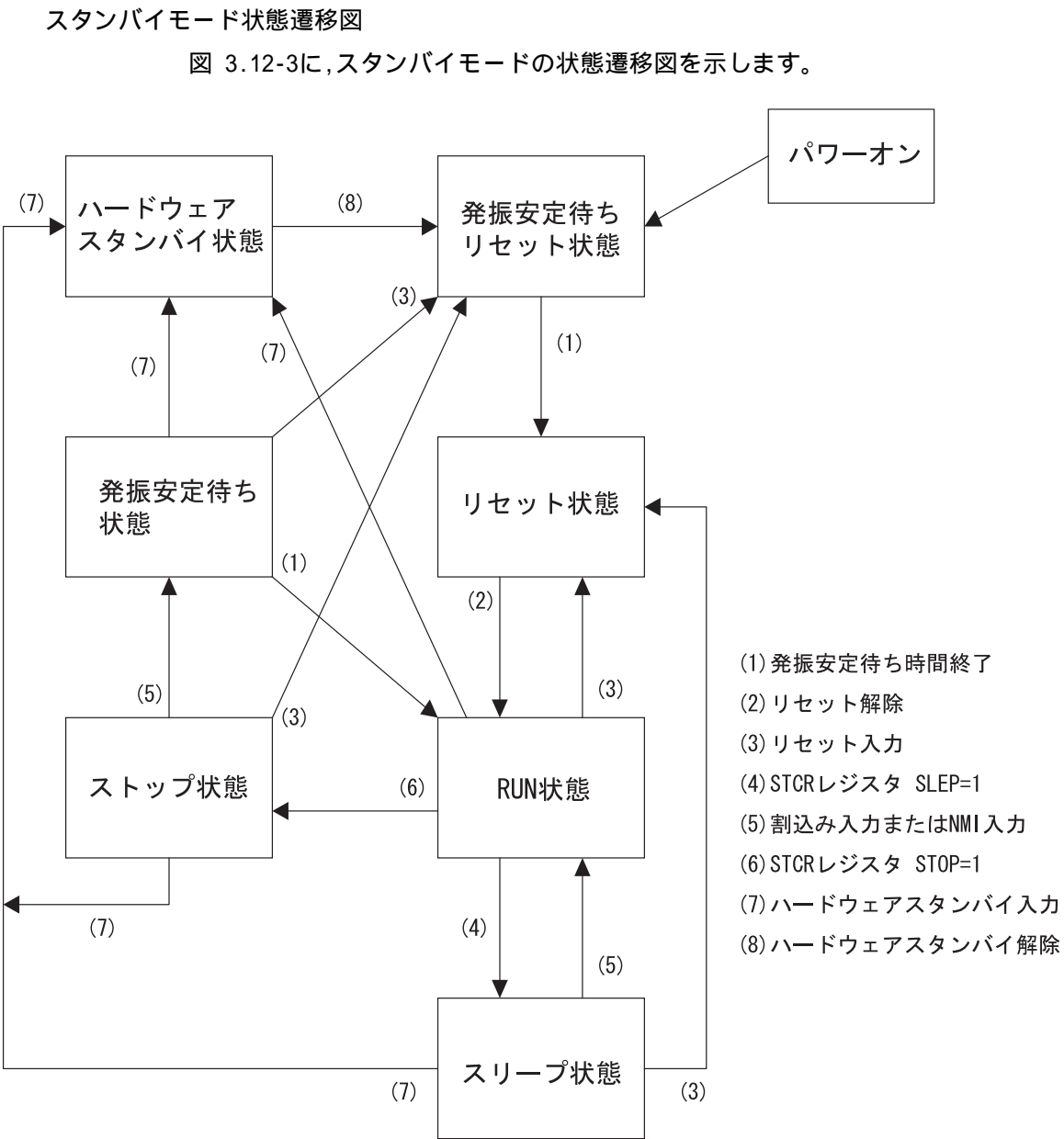


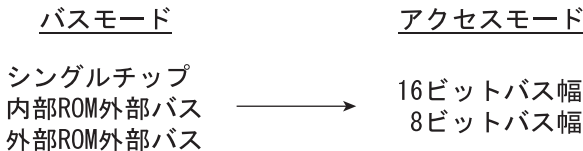
図 3.12-3 低消費電力モード状態遷移図

### 3.13 動作モード

動作モードには、バスモードとアクセスモードがあります。  
モード端子(MD2,MD1,MD0)と、モードレジスタ(MODR)で動作モードの制御を行います。

#### 動作モード

動作モードには、バスモードとアクセスモードがあります。



#### 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指し、モード設定端子(MD2,MD1,MD0)とモードレジスタ(MODR)内のM1,M0ビットの内容で指定します。

#### 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し、モード設定端子(MD2,MD1,MD0)とエリアモードレジスタ(AMD0,AMD1,AMD32,AMD4,AMD5)内のBW1,BW0ビットで指定します。

#### モード端子

MD2,MD1,MD0の3端子で、表 3.13-1に示すように動作の設定を行います。

表 3.13-1 モード設定

モード端子			モード名	リセットベクトル 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 bit	外部ROM 外部バスモード
0	0	1	外部ベクタ モード1	外部	16 bit	外部ROM 外部バスモード
0	1	0	-	-	-	設定禁止
0	1	1	内部ベクタ モード	内部	(モードレジスタ)	シングルチップモード
1	-	-	-	-	-	使用禁止

#### モードデータ

リセット後,CPUが "000007FF<sub>H</sub>"に書き込むデータをモードデータと呼びます。  
"000007FF<sub>H</sub>"に存在するのはモードレジスタ(MODR)で、このレジスタに設定した後は、このレジスタの設定モードで動作します。モードレジスタ(MODR)には、リセット後、1度だけしか書き込むことができません。  
このレジスタによる設定値は、書き込み直後に有効となります。

モードレジスタ(MODR)

図 3.13-1に、モードレジスタ(MODR)のビット構成を示します。

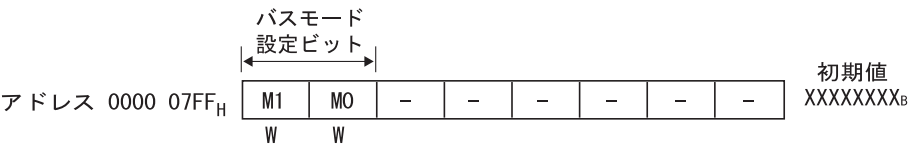


図 3.13-1 モードレジスタ(MODR)のビット構成

バスモード設定ビット(M1,M0)

モードレジスタ書き込み後のバスモードの指定を行います。

表 3.13-2に、各ビットの機能の関係を示します。

表 3.13-2バスモード設定ビットとその機能

M1	M0	機 能	備 考
0	0	シングルチップモード	
0	1	内部ROM外部バスモード	
1	0	外部ROM外部バスモード	
1	1	-	設定禁止

< 注意事項 >

内部ROMを持たない品種では、上記のうち“1, 0”のみ設定してください。

その他のビット(\*)

これらのビットには、常に“0”を書き込んでください。

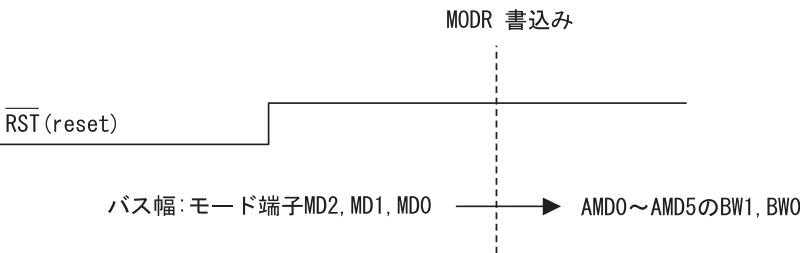
モードレジスタ(MODR)書き込み時の注意点

MODRに書き込む前には必ずAMD0～AMD5を設定し、各チップセレクト(CS)領域のバス幅を決定してください。MODRには、バス幅を設定するビットはありません。

バス幅に関しては、MODR書き込み時はモード端子MD2～MD0の値が、MODR書き込み後はAMD0～AMD5のBW1,BW0の設定値が有効になります。

例えば、外部リセットペクタは通常領域0( $\overline{CS0}$ がアクティブな領域)で行いますが、そのときのバス幅はMD2～MD0端子で決まります。MD2～MD0でそのときのバス幅を16ビットに設定し、AMD0には何も設定しないままMODRに書き込みを行ってしまうと、AMD0のバス幅の初期値は8ビットになっているため、領域0はMODRに書き込み後、8ビットバスモードに遷移してバス動作をしてしまい、誤動作します。

このような問題を防ぐため、MODRを書き込む前には必ずAMD0～AMD5を設定しておく必要があります。



## 第4章 バスインタフェース

---

この章では、外部バスインタフェースの基本事項、レジスタ構成/機能、バス動作、バスタイミングおよびバス動作のプログラム例について説明します。

なお、MB91F127/128にはDRAM制御信号用の端子を持っていませんので、DRAMインタフェースとして使用することはできません。

---

- 4.1 バスインタフェースの概要
- 4.2 バスインタフェースのブロックダイアグラム
- 4.3 バスインタフェースのレジスタ
- 4.4 バス動作
- 4.5 バスタイミング
- 4.6 内部クロック逡倍動作(クロックダブラ)
- 4.7 外部バスのプログラム例

## 4.1 バスインタフェースの概要

---

バスインタフェースは、外部メモリおよび外部I/Oとのインタフェースを制御します。

---

### バスインタフェースの特長

- 25ビット(32MB)のアドレス出力
- チップセレクト機能により6個の独立したバンク
  - 最小64KB単位で論理アドレス空間上の任意の位置に設定可能
  - アドレス端子とチップセレクト端子で合計32MB×6の領域設定可能
- チップセレクト領域ごとに16/8ビットのバス幅の設定が可能
- プログラマブルな自動メモリウェイト(最大で7サイクル分)の挿入
- アドレス/データの時分割入出力インタフェースのサポート
- 未使用のアドレス/データ端子はI/Oポートとして使用可能
- リトルエンディアンモードサポート

## チップセレクト領域

バスインタフェースには全部で、6種類のチップセレクト領域が用意されます。

各領域の位置は、エリア選択レジスタ (ASR1 ~ ASR5) とエリアマスクレジスタ (AMR1 ~ AMR5) で4GBの空間に、最小64KB単位で任意に配置することができます。また、これらのレジスタによって指定された領域に対して外部バスアクセスをしようとすると、対応するチップセレクト信号  $\overline{CS0}$  ~  $\overline{CS5}$  がアクティブ“L”になります。

リセット時は、 $\overline{CS0}$  を除くこれらの端子はインアクティブ“H”となります。

## &lt; 注意事項 &gt;

領域0は、ASR1 ~ ASR5によって指定された領域以外の空間に割り当てられます。

リセット時は、00010000<sub>H</sub> ~ 0005FFFF<sub>H</sub>以外の外部領域が領域0です。

図 4.1-1の(a)に領域1から領域5を00100000<sub>H</sub> ~ 0014FFFF<sub>H</sub>に64KB単位で配置した例を示します。また、図 4.1-1の(b)に領域1を00000000<sub>H</sub> ~ 0007FFFF<sub>H</sub>の512KB、領域2から領域5を00100000<sub>H</sub> ~ 004FFFFF<sub>H</sub>に1MB単位で配置した例を示します。

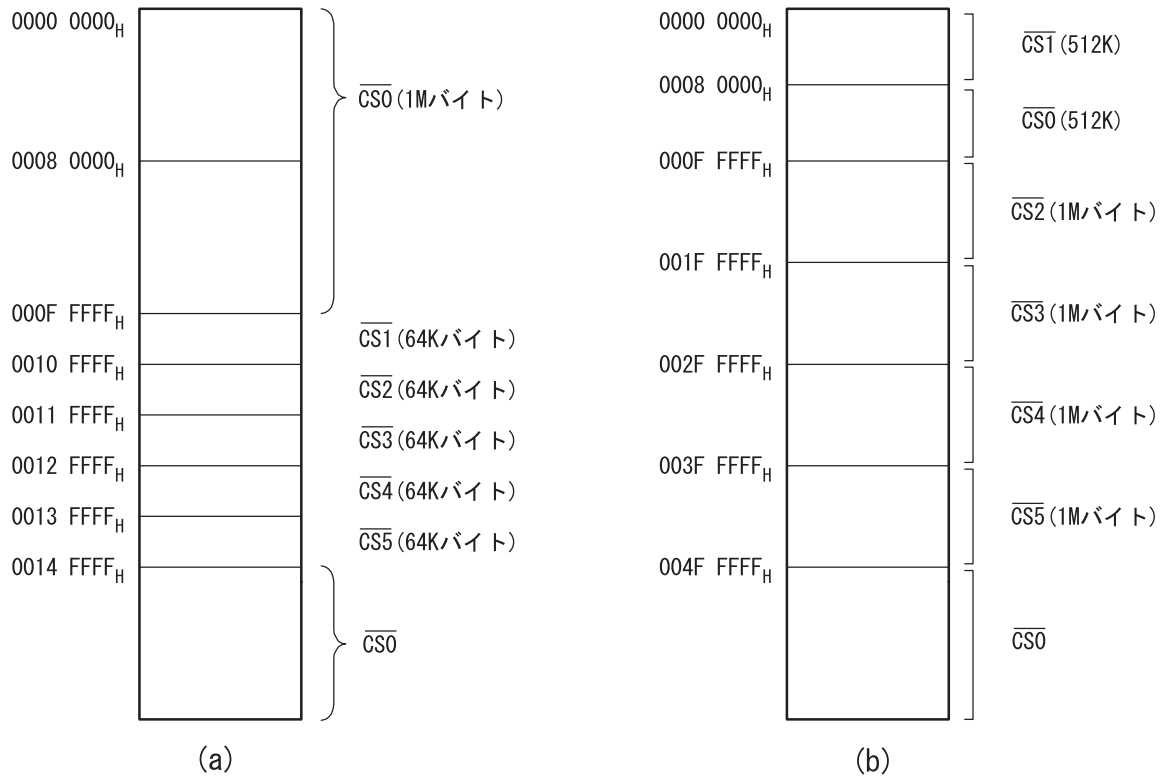


図 4.1-1 チップセレクト領域の設定例



## バスインタフェース

バスインタフェースには次のインタフェースがあります。

- 通常バスインタフェース
- アドレス/データ時分割入出力インタフェース

これらのインタフェースは、あらかじめ決められた領域でのみ使用できます。

表 4.1-1に、各チップセレクト領域と、使用可能なインタフェース機能との対応を示します。

エリアモードレジスタ(AMD)によって、これらのどのインタフェースを使用するかどうかを選択します。なお、選択しない場合は通常バスインタフェースになります。

表 4.1-1 チップセレクト領域と選択できるバスインタフェースとの対応

領域	選択できるバスインタフェース			備 考
	通常バス	時分割	DRAM	
0		-	-	リセット時
1			-	
2		-	-	
3		-	-	
4		-	-	
5		-	-	

## 時分割入出力指定

領域1はAMD1で設定された幅のバス上にアドレスとデータが時分割で入出力されます。ALE端子にアドレスのラッチパルスが出力されます。

## バスサイズ指定

各領域は、レジスタの設定によりバス幅の任意指定ができます。領域0は、リセット時MD2, MD1, MD0で設定されたバス幅になり、モードレジスタ(MODR)に書込み以降は、エリアモードレジスタ0(AMD0)の設定値によりバスサイズが指定されます。

## 4.2 バスインタフェースのブロックダイアグラム

図 4.2-1に,バスインタフェースのブロックダイアグラムを示します。

バスインタフェースのブロックダイアグラム

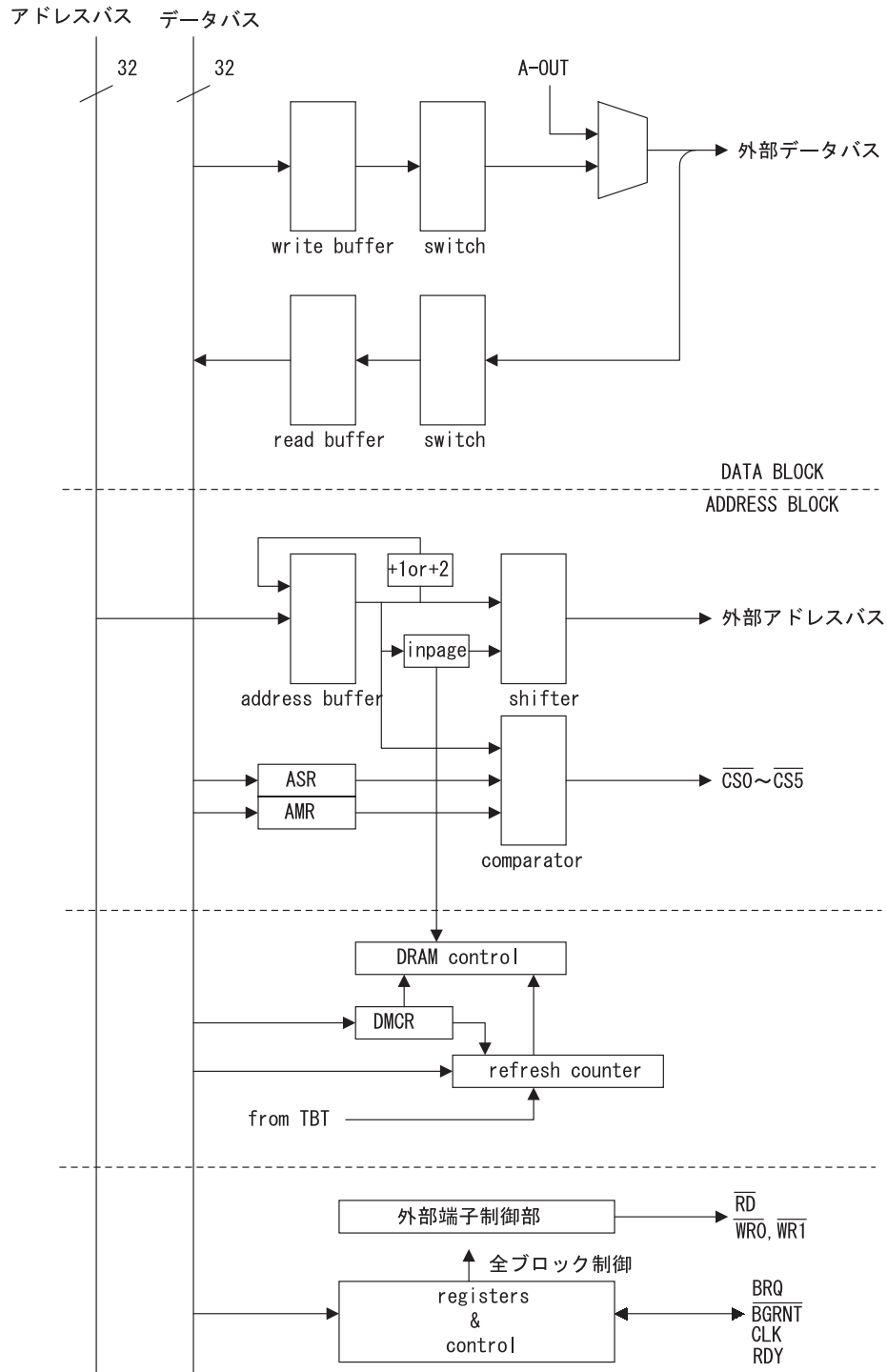


図 4.2-1 バスインタフェースのブロックダイアグラム

### 4.3 バスインタフェースのレジスタ

図 4.3-1に,バスインタフェースのレジスタ一覧を示します。

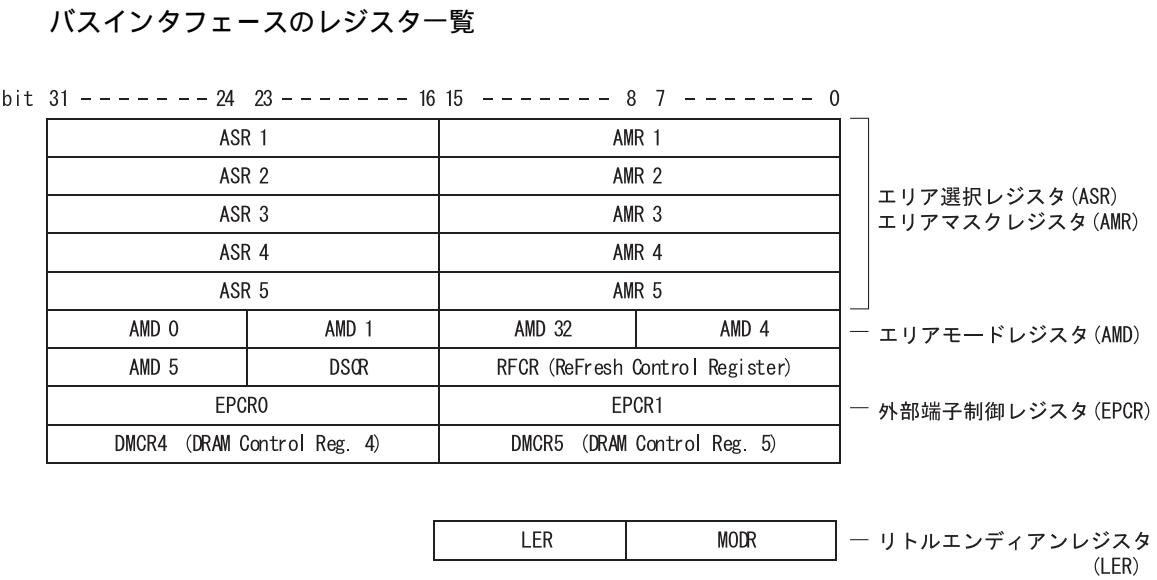


図 4.3-1 バスインタフェースのレジスタ一覧

モードレジスタ(MODR)については,「3.13 動作モード」を参照してください。

### 4.3.1 エリア選択レジスタ(ASR)とエリアマスクレジスタ(AMR)

エリア選択レジスタ(ASR1～ASR5)とエリアマスクレジスタ(AMR1～AMR5)は、チップセレクト領域1～5のアドレス空間の範囲を指定します。

エリア選択レジスタ(ASR)とエリアマスクレジスタ(AMR)の構成

エリア選択レジスタ(ASR)とエリアマスクレジスタ(AMR)のレジスタ構成は以下のとおりです。

エリア選択レジスタ(ASR1～ASR5)

ASR1 bit	15	14	13	12	...	2	1	0	初期値
アドレス 0000060C <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0001 <sub>H</sub>
	W	W	W			W	W	W	
ASR2 bit	15	14	13	12	...	2	1	0	
アドレス 00000610 <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0002 <sub>H</sub>
	W	W	W			W	W	W	
ASR3 bit	15	14	13	12	...	2	1	0	
アドレス 00000614 <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0003 <sub>H</sub>
	W	W	W			W	W	W	
ASR4 bit	15	14	13	12	...	2	1	0	
アドレス 00000618 <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0004 <sub>H</sub>
	W	W	W			W	W	W	
ASR5 bit	15	14	13	12	...	2	1	0	
アドレス 0000061C <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0005 <sub>H</sub>
	W	W	W			W	W	W	

図 4.3-2 エリア選択レジスタ(ASR1～ASR5)の構成

エリアマスクレジスタ(AMR1～AMR5)

AMR1 bit	15	14	13	12	...	2	1	0	初期値
アドレス 0000060E <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0000 <sub>H</sub>
	W	W	W			W	W	W	
AMR2 bit	15	14	13	12	...	2	1	0	
アドレス 00000612 <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0000 <sub>H</sub>
	W	W	W			W	W	W	
AMR3 bit	15	14	13	12	...	2	1	0	
アドレス 00000616 <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0000 <sub>H</sub>
	W	W	W			W	W	W	
AMR4 bit	15	14	13	12	...	2	1	0	
アドレス 0000061A <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0000 <sub>H</sub>
	W	W	W			W	W	W	
AMR5 bit	15	14	13	12	...	2	1	0	
アドレス 0000061E <sub>H</sub>	A31	A30	A29	...	...	A18	A17	A16	0000 <sub>H</sub>
	W	W	W			W	W	W	

図 4.3-3 エリアマスクレジスタ(AMR1～AMR5)

エリア選択レジスタ (ASR1 ~ ASR5) とエリアマスクレジスタ (AMR1 ~ AMR5) は、チップセレクト領域1 ~ 5のアドレス空間の範囲を指定します。

ASR1 ~ ASR5はアドレスの上位16ビット (A31 ~ A16) を指定し、AMR1 ~ AMR5で対応するアドレスビットをマスクします。AMR1 ~ AMR5の各ビットは"0"でcare, "1"でdon't careを示します。

"care"は、ASRの設定値が"0"の場合は"0", "1"の場合は"1"としてアドレス空間を示します。

"don't care"の場合は、ASRの設定値にかかわらず, "0", "1"両方の場合のアドレス空間を示します。

以下にASRとAMRの組合せによる、各チップセレクト領域指定の例を示します。

[例1]

```
ASR1 = 00000000 00000011B
AMR1 = 00000000 00000000B
```

上記を設定した場合、ASR1に"1"を設定したビットに対応するAMR1のビットは"0"のため、領域1のアドレス空間は、次のように64KBになります。

```
00000000 00000011 00000000 00000000B (00030000H)
      |
00000000 00000011 11111111 11111111B (0003FFFFH)
```

[例2]

```
ASR2 = 00001111 11111111B
AMR2 = 00000000 00000011B
```

上記を設定した場合、AMR2に"0"を設定したビットに対応するASR2の設定値は"1"と"0"をそのままcareし、AMR2に"1"を設定したビットに対応するASR2のビットは、"0"もしくは"1"のdon't careになることから、領域2のアドレス空間は、次のように256KBになります。

```
00001111 11111100 00000000 00000000B (0FFC0000H)
      |
00001111 11111111 11111111 11111111B (0FFFFFFFH)
```

領域1 ~ 5の各領域のアドレス空間は、ASR1 ~ 5とAMR1 ~ 5で、4GBの空間に最小64KB単位で任意に配置することができます。これらのレジスタによって指定された領域に対してバスアクセスを行うと、対応するチップセレクト端子 (CS0 ~ CS5) が"L"出力となります。

なお、領域0はASR1 ~ 5とAMR1 ~ 5で設定された領域以外の空間が割り当てられ、リセット時はASR1 ~ 5とAMR1 ~ 5の初期値により0001000<sub>H</sub> ~ 0005FFFF<sub>H</sub>以外の領域が割り当てられています。

< 注意事項 >

チップセレクト領域は、お互いにオーバーラップしないように設定してください。

図 4.3-4に、リセット時の初期値で64KBに設定されたマップと、[例1]と[例2]で設定した領域のマップを示します。

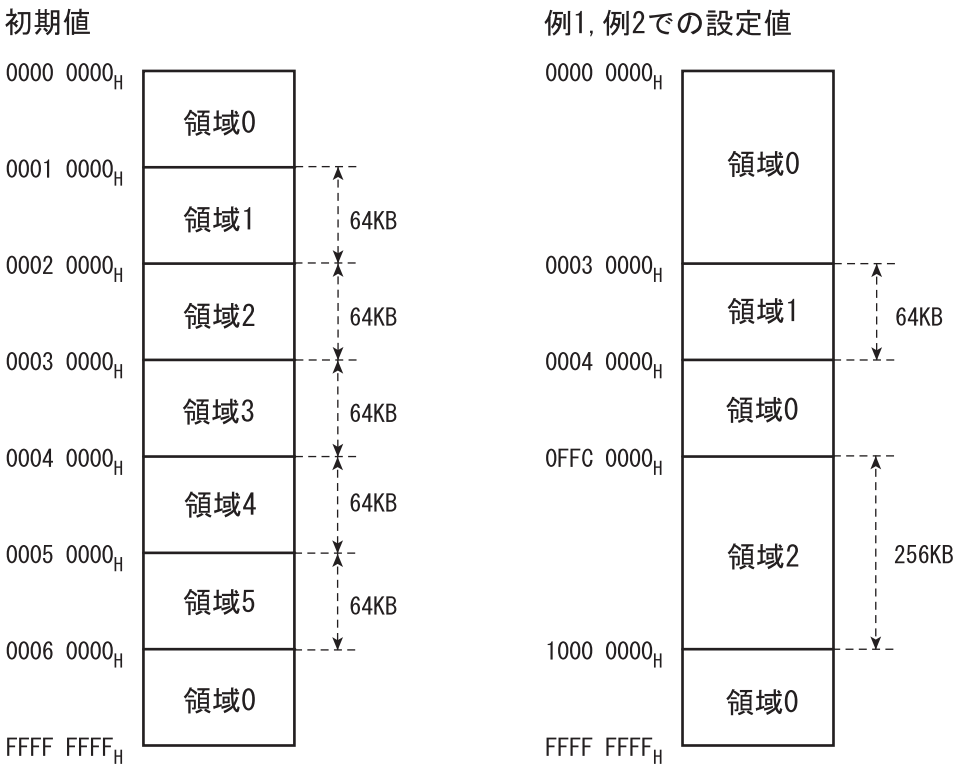


図 4.3-4 チップセレクト領域を設定したマップの例

### 4.3.2 エリアモードレジスタ0(AMD0)

エリアモードレジスタ0(AMD0)は,チップセレクト領域0(ASR1～5およびAMR1～5で指定された領域以外の領域)の動作モードを指定します。リセット時は,領域0が選択されます。

エリアモードレジスタ0(AMD0)の構成

図 4.3-5に,エリアモードレジスタ0(AMD0)のレジスタ構成を示します。

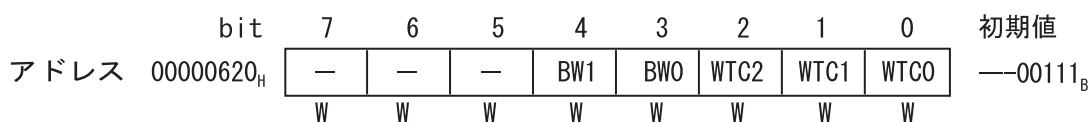


図 4.3-5 エリアモードレジスタ0(AMD0)の構成

エリアモードレジスタ0(AMD0)のビット機能

以下に,エリアモードレジスタ0(AMD0)の各ビットの機能説明します。

【ビット4,3】: BW1,0(Bus Width bit)

BW1,BW0は,領域0のバス幅を指定します。

BW1	BW0	バス幅
0	0	8ビット
0	1	16ビット
1	0	設定禁止
1	1	設定禁止

< 注意事項 >

BW1,BW0の初期値はともに”0”ですが,読出し時MODRを書き込むまでは,レジスタの値ではなくMD1,MD0の端子レベルが読まれます。

【ビット2～0】: WTC2～0(Wait Cycle bit)

WTC2～WTC0は,通常バスインタフェース時の自動ウェイトの挿入回数を指定します。

WTC2	WTC1	WTC0	挿入ウェイトサイクル数
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

AMD0のWTC2～WTC0はリセット時に”111”に設定され,リセット解除直後のバスアクセスでは自動的に7サイクルのウェイトが挿入されます。

< 注意事項 >

MODRを書き込む前に,必ずAMD0のBW1,BW0にMD2,MD1,MD0端子で設定したバス幅と同じバス幅を設定してください。

領域0のバス幅は、リセット時はMD2, MD1, MD0端子で設定され、モードレジスタ (MODR) 設定後は, AMD0で設定されているバス幅が有効になります。



MD2, MD1, MD0端子で領域0を16ビット幅に設定し、そのバス幅のままでAMD0を設定せずにMODRの書込みを行うと、AMD0のBW1, BW0の初期値は"00"のため、8ビットバス幅に遷移して誤動作してしまいます。



### 4.3.3 エリアモードレジスタ1(AMD1)

エリアモードレジスタ1(AMD1)は、チップセレクト領域1(ASR1とAMR1で指定された領域)の動作モードを指定します。

領域1ではアドレス/データ入出力の時分割入出力インタフェースを指定できます。時分割入出力インタフェースはデータバスにアドレス出力/データ入出力を行うインタフェースであり、以下のように8ビットバス幅/16ビットバス幅のみをサポートします。

- ・ 8ビットバス幅:A7～A0がD31～D24にマルチプレクス
- ・ 16ビットバス幅:A15～A0がD31～D16にマルチプレクス

エリアモードレジスタ1(AMD1)の構成

図 4.3-6に、エリアモードレジスタ1(AMD1)のレジスタ構成を示します。

	bit	7	6	5	4	3	2	1	0	初期値
アドレス	00000621 <sub>H</sub>	MPX	—	—	BW1	BW0	WTC2	WTC1	WTC0	0—00000 <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 4.3-6 エリアモードレジスタ1(AMD1)の構成

エリアモードレジスタ1(AMD1)のビット機能

以下に、エリアモードレジスタ1(AMD1)の各ビットの機能説明します。

【ビット7】：MPX (MultiPleX bit)

MPXは、アドレス/データ入出力の時分割入出力インタフェースを制御します。

MPX	機能
0	通常バスインタフェース
1	時分割入出力インタフェース

【ビット4,3】：BW1,BW0 (Bus Width bit)

BW1,BW0は、領域1のバス幅を指定します。

BW1	BW0	バス幅
0	0	8ビット
0	1	16ビット
1	0	設定禁止
1	1	Reserved

【ビット2～0】：WTC2～WTC0 (Wait Cycle bit)

WTCは、通常バスインタフェース動作時の自動挿入ウェイトサイクル数を指定します。動作はAMD0のWTC2～WTC0と同様ですが、リセットで”000”に初期化され、挿入ウェイトサイクル数は”0”になります。

### 4.3.4 エリアモードレジスタ32(AMD32)

エリアモードレジスタ(AMD32)は、チップセレクト領域2(ASR2とAMR2で指定された領域)とチップセレクト領域3(ASR3とAMR3で指定された領域)の動作モードを制御します。

これらの領域は通常バスアクセスのみであり、時分割入出力インタフェースは使用できません。

領域2,3のバス幅は、BW1, BW0ビットで一括して同じバス幅を制御し、自動ウェイトサイクルはそれぞれの領域で独立に設定できます。

エリアモードレジスタ32(AMD32)の構成

図 4.3-7に、エリアモードレジスタ32(AMD32)のレジスタ構成を示します。

	bit	7	6	5	4	3	2	1	0	初期値
アドレス	00000622 <sub>H</sub>	BW1	BW0	WT32	WT31	WT30	WT22	WT21	WT20	00000000 <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 4.3-7 エリアモードレジスタ32(AMD32)の構成

エリアモードレジスタ32(AMD32)のビット機能

以下に、エリアモードレジスタ32(AMD32)の各ビットの機能説明します。

【ビット7,6】：BW1, BW0 (Bus Width bit)

BW1, BW0は、領域2/領域3のバス幅を指定します。

BW1	BW0	バス幅
0	0	8ビット
0	1	16ビット
1	0	設定禁止
1	1	Reserved

【ビット5～3】：WT32～WT30 (Wait Cycle bit)

WT32～WT30は、領域3のメモリアクセス時の自動ウェイトサイクル数を指定します。

動作は、AMD0のWTC2～0と同様です。リセットで"000"に初期化され、挿入ウェイトサイクル数は"0"になります。

【ビット2～0】：WT22～WT20 (Wait Cycle bit)

WT22～WT20は、領域2のメモリアクセス時の自動挿入ウェイトサイクル数を指定します。

動作は、AMD0のWTC2～0と同様です。リセットで"000"に初期化され、挿入ウェイトサイクル数は"0"になります。

### 4.3.5 エリアモードレジスタ4(AMD4)

エリアモードレジスタ4(AMD4)は、チップセレクト領域4(ASR4とAMR4で指定された領域)の動作モードを指定します。

エリアモードレジスタ4(AMD4)の構成

図 4.3-8に、エリアモードレジスタ4(AMD4)のレジスタ構成を示します。

bit	7	6	5	4	3	2	1	0	初期値
アドレス 00000623 <sub>H</sub>	DRME	—	—	BW1	BW0	WTC2	WTC1	WTC0	0—00000 <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 4.3-8 エリアモードレジスタ4(AMD4)の構成

エリアモードレジスタ4(AMD4)のビット機能

以下に、エリアモードレジスタ4(AMD4)の各ビットの機能説明します。

【ビット7】：DRME (DRaM Enable bit)

DRMEは、領域4に対して通常バスインタフェースを選択します。

DRME	動作
0	通常バスインタフェース
1	設定禁止

【ビット4,3】：BW1,BW0 (Bus Width bit)

BW1,BW0は、領域4のバス幅を指定します。ほかのエリアモードレジスタ(AMD)のBWビットと同様の機能を持ちます。

BW1	BW0	バス幅
0	0	8ビット
0	1	16ビット
1	0	設定禁止
1	1	Reserved

【ビット2～0】：WTC2～WTC0 (Wait Cycle bit)

WTC2～WTC0は、領域4のメモリアクセス時の自動挿入ウェイトサイクル数を指定します。ほかのエリアモードレジスタ(AMD)のWTCビットと同様の機能を持ち、リセットで”000”に初期化された挿入ウェイトサイクル数は”0”になります。

4.3.6 エリアモードレジスタ5(AMD5)

エリアモードレジスタ5(AMD5)は,チップセレクト領域5(ASR5とAMR5で指定された領域)の動作モードを指定します。

エリアモードレジスタ5(AMD5)の構成

図 4.3-9に,エリアモードレジスタ5(AMD5)のレジスタ構成を示します。

bit		7	6	5	4	3	2	1	0	初期値
アドレス	00000624 <sub>H</sub>	DRME	—	—	BW1	BW0	WTC2	WTC1	WTC0	0—00000 <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 4.3-9 エリアモードレジスタ5(AMD5)の構成

エリアモードレジスタ5(AMD5)のビット機能

以下に,エリアモードレジスタ5(AMD5)の各ビットの機能説明します。

【ビット7】: DRME (DRaM Enable bit)

DRMEは,領域5に対して通常バスインタフェースかを選択します。

DRME	機能
0	通常バスインタフェース
1	設定禁止

【ビット4,3】: BW1,BW0 (Bus Width bit)

BW1,BW0は,領域5のバス幅を指定します。ほかのエリアモードレジスタ(AMD)のBWビットと同様の機能を持ちます。

BW1	BW0	バス幅
0	0	8ビット
0	1	16ビット
1	0	設定禁止
1	1	Reserved

【ビット2~0】: WTC2~WTC0 (Wait Cycle bit)

WTC2~WTC0は,領域5のメモリアクセス時の自動挿入ウェイトサイクル数を指定します。ほかのエリアモードレジスタ(AMD)のWTCビットと同様の機能を持ち,リセットで”000”に初期化され,挿入ウェイトサイクル数は”0”になります。

### 4.3.7 外部端子制御レジスタ0(EPCR0)

外部端子制御レジスタ0(EPCR0)は、各信号の出力を制御します。

出力許可の場合は、各バスモードで所望のタイミングを出力し、入力有効の場合は、外部からの入力信号を受付けます。

出力禁止/入力無効の場合は、I/Oポートとして使用できます。

外部端子制御レジスタ0(EPCR0)の構成

図 4.3-10に、外部端子制御レジスタ0(EPCR0)のレジスタ構成を示します。

bit	15	14	13	12	11	10	9	8	初期値
アドレス	00000628 <sub>H</sub>								—1-1100 <sub>B</sub>
	—	—	ALEE	—	WRE	RDXE	RDYE	BRE	
	W	W	W	W	W	W	W	W	
bit	7	6	5	4	3	2	1	0	
	—	CKE	COE5	COE4	COE3	COE2	COE1	COE0	-1111111 <sub>B</sub>
	W	W	W	W	W	W	W	W	

図 4.3-10 外部端子制御レジスタ0(EPCR0)の構成

外部端子制御レジスタ0(EPCR0)のビット機能

以下に、外部端子制御レジスタ0(EPCR0)の各ビットの機能説明します。

【ビット13】：ALEE (ALE output Enable bit)

ALEEは、ALE出力を許可するかどうかを選択します。

リセット時は出力許可となります。

ALEE	機能
0	出力禁止
1	出力許可 (初期値)

ALEは領域1を時分割モードで使用した際、外部周辺のアドレスストローブとして使用します。

【ビット11】：WRE (WRite pulse output Enable bit)

WREはライトパルス  $\overline{WR0} \sim \overline{WR1}$  を出力するかどうかを選択します。

リセット時は出力許可となります。

WRE	機能
0	出力禁止
1	出力許可 (初期値)

MB91F127/128では、WREビットによる  $\overline{WR0} \sim \overline{WR1}$  端子のI/Oポート制御を行わないため、このビットには、常に“1”を設定してください。ライトパルスはWREビットを“1”に設定していても、AMDで設定されたバス幅に応じてI/Oポートとして使用することができます(例えば、8ビットモードでは  $\overline{WR1}$  は出力されず、対応する端子はI/Oポートとして使用できます)。

## 【ビット10】: RDXE (ReaDX pulse output Enable bit)

RDXEは、リードパルス  $\overline{RD}$  を出力するかどうかを選択します。

リセット時は出力許可となります。

WRE	機能
0	出力禁止 (設定禁止)
1	出力許可 (初期値)

外部バスモード使用時は、RDXEビットによる  $\overline{RD}$  端子のI/Oポート制御は行わないため、このビットには、常に"1"を設定してください。

## 【ビット9】: RDYE (ReaDY input Enable bit)

RDYEは、RDY入力を以下のように制御します。

リセット時は入力無効となります。

RDYE	機能
0	RDY入力無効 (初期値)
1	RDY入力有効

## 【ビット8】: BRE (Bus Request Enable bit)

BREは、BRQと  $\overline{BGRNT}$  を以下のように制御します。

リセット時はBRQの入力は無効、 $\overline{BGRNT}$  の出力は禁止となります。

BRE	機能
0	BRQの入力無効、 $\overline{BGRNT}$ の出力禁止 (端子はI/Oポートとして機能) (初期値)
1	BRQの入力有効、 $\overline{BGRNT}$ の出力許可

## 【ビット6】: CKE (Clock output Enable bit)

CKEは、CLK(外部バスの動作クロック波形)の出力イネーブルビットです。

CKE	機能
0	出力禁止
1	出力許可 (初期値)

このビットはリセット時に"1"に初期化され、CLKは出力状態になります。

## 【ビット5】: COE5 (Chip select Output Enable 5)

COE5は、 $\overline{CS5}$  の出力を制御します。リセット時は出力許可となります。

COE5	機能
0	出力禁止
1	出力許可 (初期値)

## 【ビット4】: COE4 (Chip select Output Enable 4)

COE4は、 $\overline{CS4}$  の出力を制御します。リセット時は出力許可となります。

COE4	機能
0	出力禁止
1	出力許可 (初期値)

## 【ビット3】: COE3 (Chip select Output Enable 3)

COE3は、 $\overline{CS3}$  の出力を制御します。リセット時は出力許可となります。

COE3	機能
0	出力禁止
1	出力許可 (初期値)

【ビット2】：COE2 (Chip select Output Enable 2)

COE2は、 $\overline{CS2}$  の出力を制御します。リセット時は出力許可となります。

COE2	機能
0	出力禁止
1	出力許可 (初期値)

【ビット1】：COE1 (Chip select Output Enable 1)

COE1は、 $\overline{CS1}$  の出力を制御します。リセット時は出力許可となります。

COE1	機能
0	出力禁止
1	出力許可 (初期値)

【ビット0】：COE0 (Chip select Output Enable 0)

COE0は、 $\overline{CS0}$  の出力を制御します。リセット時は出力許可となります。

COE0	機能
0	出力禁止
1	出力許可 (初期値)

外部バスモード使用時には、COE0ビットによる  $\overline{CS0}$  端子のI/Oポート制御は行われないため、このビットには常に"1"を設定してください。

4.3.8 外部端子制御レジスタ1(EPCR1)

外部端子制御レジスタ1(EPCR1)は,アドレス信号の出力を制御します。

外部端子制御レジスタ1(EPCR1)の構成

図 4.3-11に,外部端子制御レジスタ1(EPCR1)のレジスタ構成を示します。

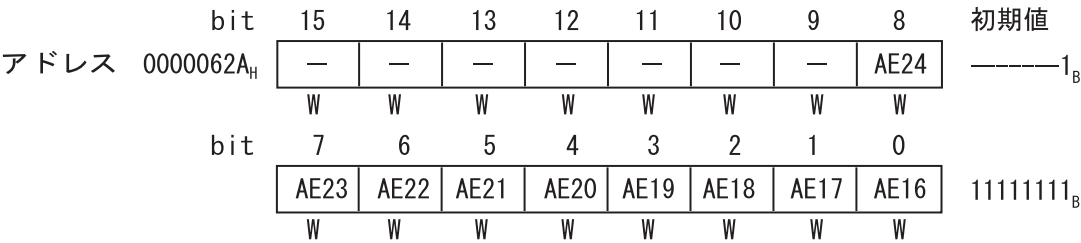


図 4.3-11 外部端子制御レジスタ1(EPCR1)の構成

外部端子制御レジスタ1(EPCR1)のビット機能

以下に,外部端子制御レジスタ1(EPCR1)の各ビットの機能説明します。

【ビット8～0】: AE24～AE16 (Address output Enable 24～16)

AE24～AE16は,対応するアドレスを出力するかどうかを指定します。  
出力禁止の場合は,I/Oポートとして使用できます。

AE24～AE16	機能
0	出力禁止
1	出力許可 (初期値)

AE24～AE16は,リセットによって"1FF<sub>H</sub>"に初期化されます。



### 4.3.9 リトルエンディアンレジスタ(LER)

MB91F127/128のバスアクセスは通常全領域ビッグエンディアン(big endian)で行いますが,リトルエンディアンレジスタ(LER)を設定することにより,領域1~5のどれか1領域をリトルエンディアン(little endian)領域として扱うことができます。

通常/時分割/DRAMインタフェースに依存せず,全バスモードに対してサポートします。ただし,領域0はリトルエンディアン対象外です。

リトルエンディアンレジスタ(LER)の構成

図 4.3-11に,リトルエンディアンレジスタ(LER)のレジスタ構成を示します。

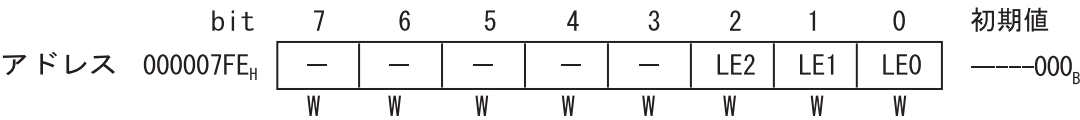


図 4.3-12 リトルエンディアンレジスタ(LER)の構成

リトルエンディアンレジスタ(LER)のビット機能

以下に, リトルエンディアンレジスタ(LER)の各ビットの機能説明します。

【ビット2~0】: LE2-0

表 4.3-1に示すように,LE2,LE1,LE0ビットの組合せでリトルエンディアン領域を指定します。

表 4.3-1 ビット(LE2,LE1,LE0)の組合せによるモードの設定

LE2	LE1	LE0	モード
0	0	0	リセット後の初期値 リトルエンディアン領域なし
0	0	1	領域1がリトルエンディアン 領域0,2~5はビッグエンディアン
0	1	0	領域2がリトルエンディアン 領域0~1,3~5はビッグエンディアン
0	1	1	領域3がリトルエンディアン 領域0~2,4~5はビッグエンディアン
1	0	0	領域4がリトルエンディアン 領域0~3,5はビッグエンディアン
1	0	1	領域5がリトルエンディアン 領域0~4はビッグエンディアン

< 注意事項 >

リトルエンディアンレジスタ(LER)には,リセット後一度だけしか書き込むことができません。

## 4.4 バス動作

---

バス動作の基本事項として、次の項目について説明します。

- ・ データバス幅と制御信号との関係
  - ・ ビッグエンディアンのバスアクセス
  - ・ リトルエンディアンのバスアクセス
  - ・ 外部アクセスの比較
- 

### データバス幅と制御信号との関係

次のバスインタフェースでのデータバス幅と制御信号の関係について説明します。

- 通常バスインタフェース
- 時分割入出力バスインタフェース
- DRAMインタフェース

### ビッグエンディアンのバスアクセス

外部アクセスについて、次の項目を説明します。

- データフォーマット
- データバス幅
- 外部バスアクセス
- 外部デバイスとの接続例

### リトルエンディアンのバスアクセス

外部アクセスについて、次の項目を説明します。

- リトルエンディアンの概要
- データフォーマット
- データバス幅
- 外部デバイスとの接続例

### ビッグエンディアンとリトルエンディアンの外部アクセスの比較

ビッグエンディアンとリトルエンディアンの外部アクセスの比較として、バス幅に対するワードアクセス、ハーフワードアクセスおよびバイトアクセスについて説明します。

### 4.4.1 データバス幅と制御信号との関係

データバスの制御信号 ( $\overline{WR0}$  ~  $\overline{WR1}$ ,  $CS0H$ ,  $CS0L$ ,  $CS1L$ ,  $CS1H$ ,  $DW0X$ ,  $DW1X$ ) は、ビッグエンディアン/リトルエンディアンやデータバス幅に関係なくデータバスのバイト位置に常に1対1に対応します。

#### データバス幅と制御信号との関係

ここでは、バスモード別に、設定されたデータバス幅で使用するMB91F127/128のデータバスのバイト位置とそれに対応する制御信号をまとめます。

#### 通常バスインタフェースでのデータバス幅と制御信号

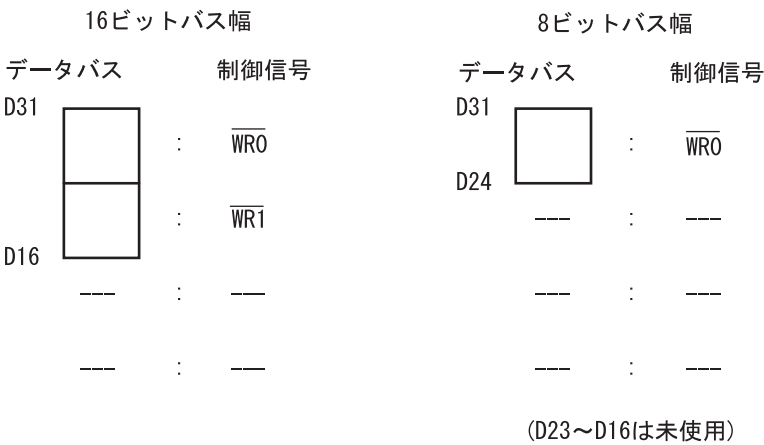


図 4.4-1 通常バスインタフェースでのデータバス幅と制御信号

#### 時分割入出力バスインタフェースでのデータバス幅と制御信号

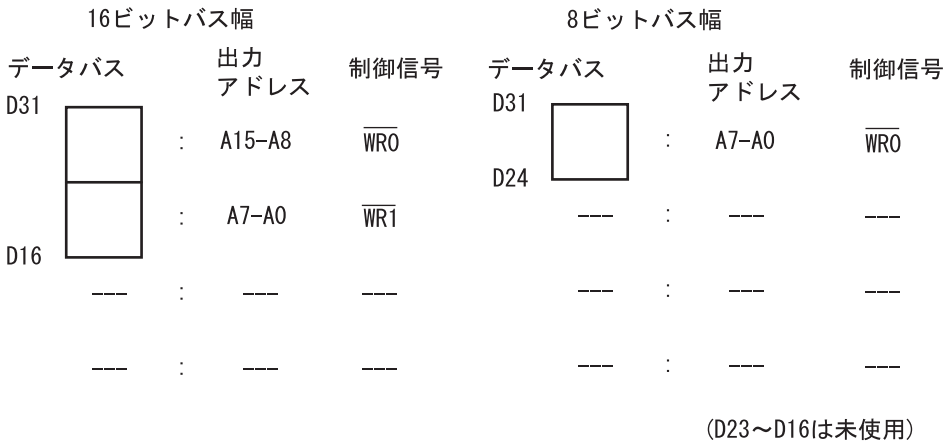


図 4.4-2 時分割入出力バスインタフェースでのデータバス幅と制御信号

DRAMインタフェースでのデータバス幅と制御信号

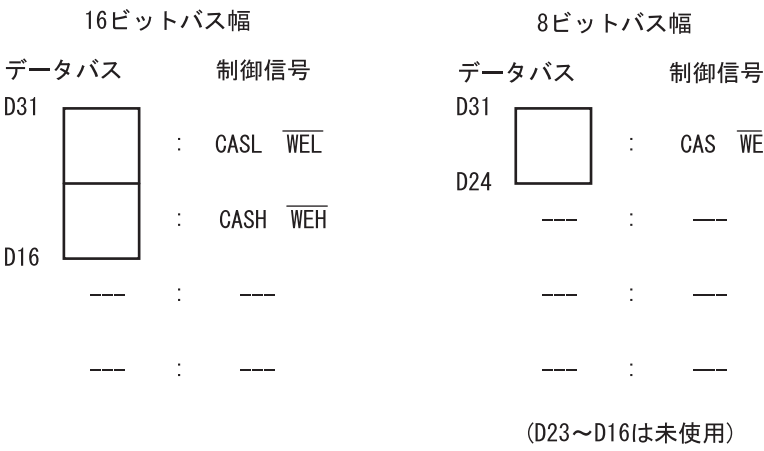


図 4.4-3 DRAMインタフェースでのデータバス幅と制御信号

表 4.4-1に,データバス幅と制御信号の関係を示します。

表 4.4-1 データバス幅と制御信号の関係

バス幅 データバス	16ビットバス幅			8ビットバス幅		
	WR	2CAS/1WE	1CAS/2WE	WR	2CAS/1WE	1CAS/2WE
D31-D24	$\overline{\text{WR0}}$	CASL	$\overline{\text{WEL}}$	$\overline{\text{WR0}}$	CAS	$\overline{\text{WE}}$
D23-D16	$\overline{\text{WR1}}$	CASH	$\overline{\text{WEH}}$			

### 4.4.2 ビッグエンディアンのバスアクセス

リトルエンディアンレジスタ(LER)を設定しない領域に対しては、ビッグエンディアンで外部バスアクセスを行います。  
FRファミリは通常ビッグエンディアンです。

データフォーマット

データフォーマットごとに、内部レジスタと外部データバスとの関係を示します。  
ワードアクセス(LD, ST命令実行時)

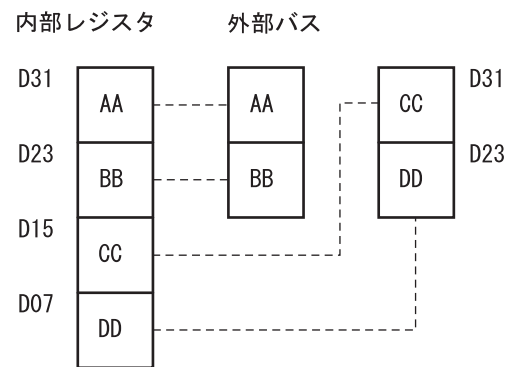


図 4.4-4 ワードアクセスの内部レジスタと外部データバスとの関係

ハーフワードアクセス(LDUH, STH命令実行時)

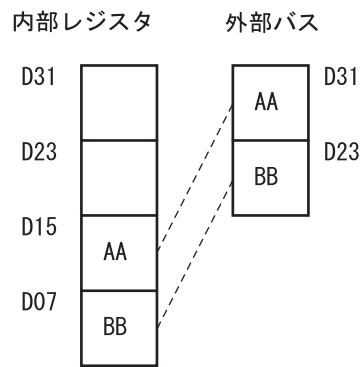
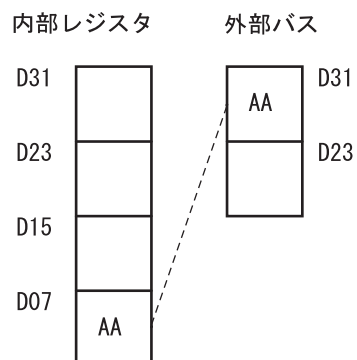


図 4.4-5 ハーフワードアクセスの内部レジスタと外部データバスとの関係

## バイトアクセス(LDUB,STB命令実行時)

(a) 出力アドレス下位“0”



(b) 出力アドレス下位“1”

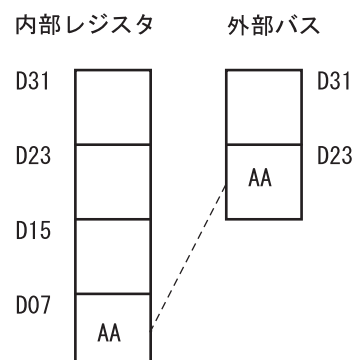


図 4.4-6 バイトアクセスの内部レジスタと外部データバスとの関係

## データバス幅

データバス幅ごとに、内部レジスタと外部データバスとの関係を示します。

## 16ビットバス幅

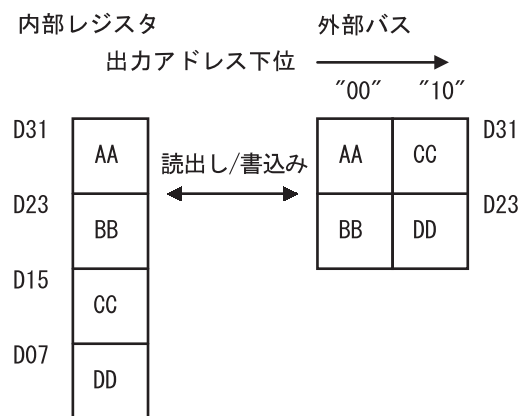


図 4.4-7 16ビットバス幅の内部レジスタと外部データバスとの関係

8ビットバス幅

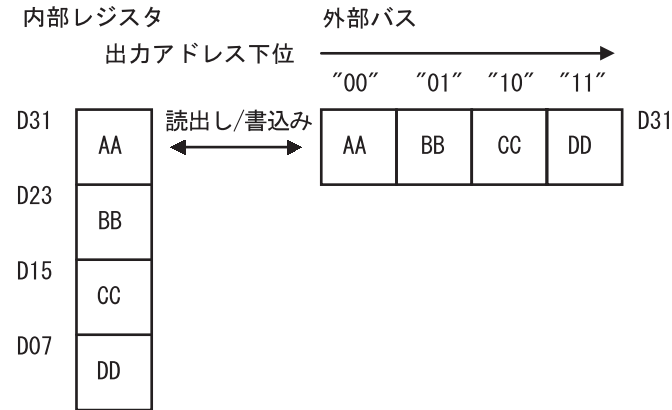


図 4.4-8 8ビットバス幅内部レジスタと外部データバスとの関係

外部バスアクセス

外部バスアクセス(16ビット/8ビットバス幅)をワード/ハーフワード/バイトアクセス別に図 4.4-9,図 4.4-10に示します。また,以下の項目も合わせて図 4.4-9,図 4.4-10に示します。

- アクセスバイト位置
- プログラムアドレスと出力アドレス
- バスアクセス回数

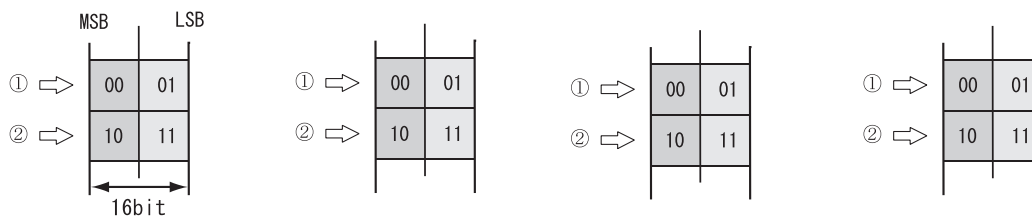
PA1/PA0 : プログラムで指定したアドレス下位2ビット  
出力 A1/A0 : 出力するアドレスの下位2ビット  
■ : 出力するアドレスの先頭バイト位置  
■ + ■ : アクセスするデータバイト位置  
①~④ : バスアクセス回数

< 注意事項 >

MB91F127/128は,ミスアラインエラーを検出しません。したがって,ワードアクセスの場合には,プログラムで指定したアドレス下位2ビットが"00", "01", "10", "11"であっても,出力するアドレスの下位2ビットはすべて"00"となり,ハーフワードアクセスの場合には"00", "01"のときは"00"に, "10", "11"のときには"10"になります。

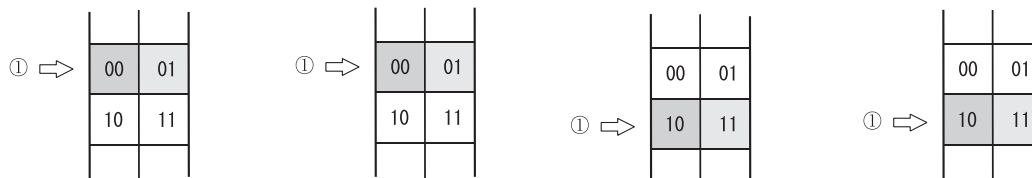
### 16ビットバス幅 ワードアクセス

- (a) PA1/PA0="00"  
→①出力A1/A0="00"  
②出力A1/A0="10"
- (b) PA1/PA0="01"  
→①出力A1/A0="00"  
②出力A1/A0="10"
- (c) PA1/PA0="10"  
→①出力A1/A0="00"  
②出力A1/A0="10"
- (d) PA1/PA0="11"  
→①出力A1/A0="00"  
②出力A1/A0="10"



### (B) ハーフワードアクセス

- (a) PA1/PA0="00"  
→①出力A1/A0="00"
- (b) PA1/PA0="01"  
→①出力A1/A0="00"
- (c) PA1/PA0="10"  
→①出力A1/A0="10"
- (d) PA1/PA0="11"  
→①出力A1/A0="10"



### (C) バイトアクセス

- (a) PA1/PA0="00"  
→①出力A1/A0="00"
- (b) PA1/PA0="01"  
→①出力A1/A0="01"
- (c) PA1/PA0="10"  
→①出力A1/A0="10"
- (d) PA1/PA0="11"  
→①出力A1/A0="11"

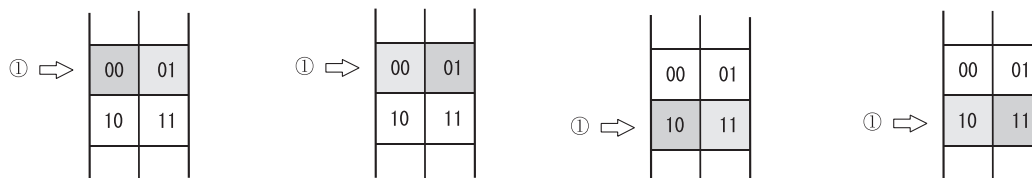
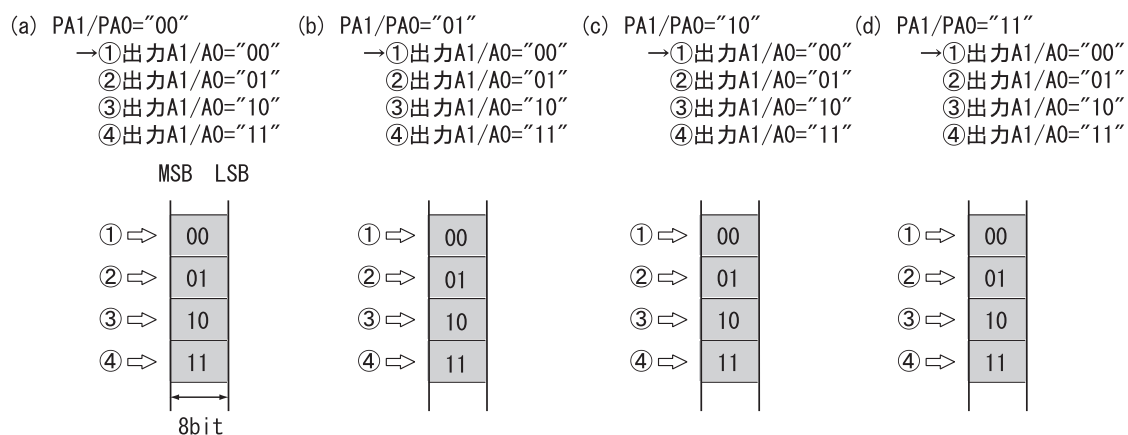


図 4.4-9 16ビット幅のときの外部バスアクセス

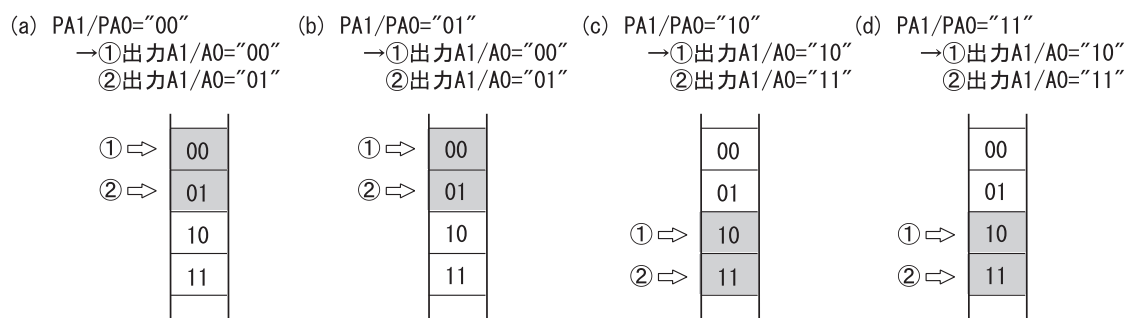


# 8ビットバス幅

## (A)ワードアクセス



## (B)ハーフワードアクセス



## (C)バイトアクセス

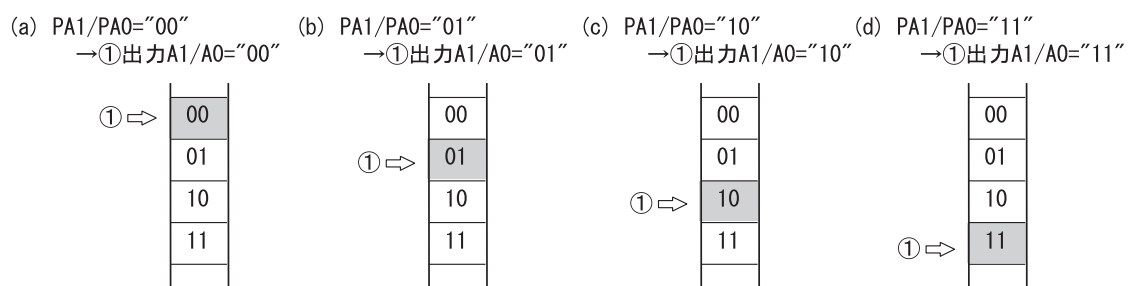
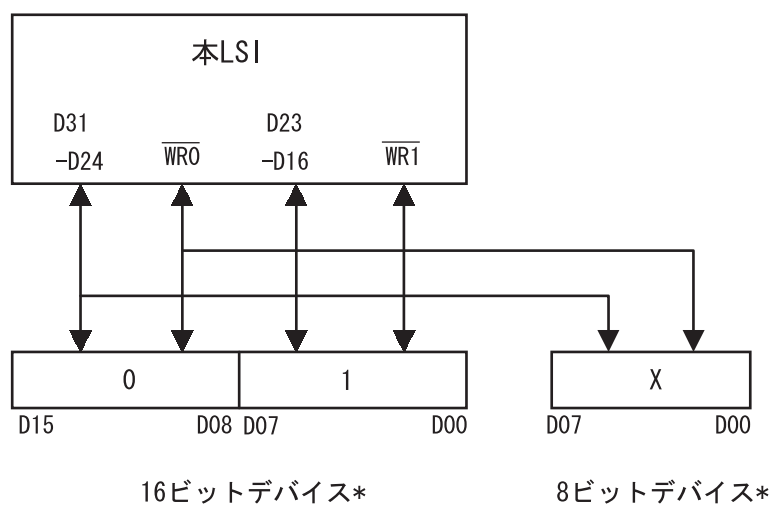


図 4.4-10 8ビットバス幅のときの外部バスアクセス

## 外部デバイスとの接続例

図 4.4-11に, MB91F127/128と外部デバイスとの接続例を示します。



(“0”, “1”はアドレス下位1ビット, “X”はアドレス下位1ビットが“0”, “1”可)  
 \*: 16ビット/8ビットデバイスの場合は, MB91F127/128のMSB側のデータバスを使用します。

図 4.4-11 MB91F127/128と外部デバイスの接続例

### 4.4.3 リトルエンディアンのバスアクセス

リトルエンディアン(LE)を設定した領域に対しては、リトルエンディアンで外部バスアクセスを行います。

#### リトルエンディアンの概要

MB91F127/128のリトルエンディアンバスアクセスは、ビッグエンディアン時のバスアクセス動作を利用し、基本的にはビッグエンディアン時の出力アドレスの順番と制御信号の出力は同じで、データバスのバイト位置をバス幅に応じてスワップすることにより実現しています。

接続時には、ビッグエンディアン領域とリトルエンディアン領域を物理的に分ける必要がありますので、十分注意が必要です。

- 出力するアドレスの順番はビッグエンディアン/リトルエンディアンで変わりありません。
- ワードアクセス:ビッグエンディアンのアドレス"00"に対応するMSB側のバイトデータが、リトルエンディアンでは、LSB側のバイトデータになります。  
ワードアクセスの場合は、ワード内の4バイトすべてのバイト位置がひっくりかえります。  
"00" "11", "01" "10", "10" "01", "11" "00"
- ハーフワードアクセス:ビッグエンディアンのアドレス"0"に対応するMSB側のバイトデータが、リトルエンディアンでは、LSB側のバイトデータになります。  
ハーフワードアクセスの場合は、ハーフワード内の2バイトのバイト位置がひっくりかえります。  
"0" "1", "1" "0"
- バイトアクセス:ビッグエンディアン/リトルエンディアンとも同じです。
- 16/8ビットバス幅で使用するデータバス制御信号はビッグエンディアン/リトルエンディアンで変わりありません。

## データフォーマット

データフォーマットごとに、内部レジスタと外部データバスとの関係を示します。

## ワードアクセス(LD,ST命令実行時)

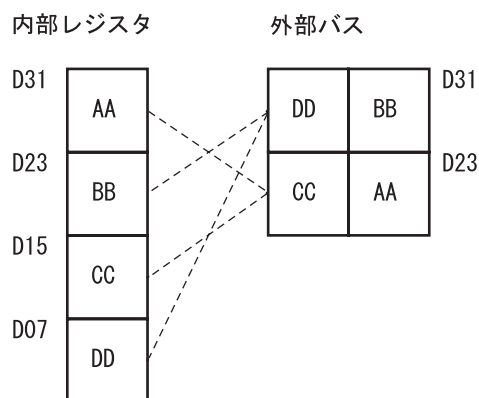


図 4.4-12 ワードアクセスの内部レジスタと外部データバスとの関係

## ハーフワードアクセス(LDUH,STH命令実行時)

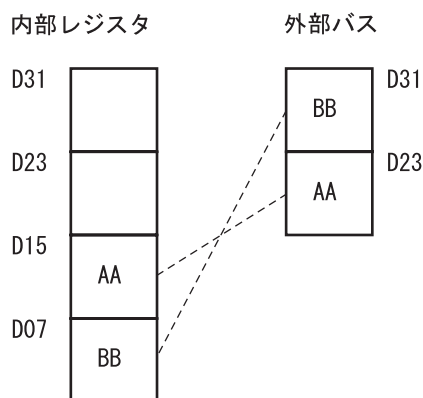


図 4.4-13 ハーフワードアクセスの内部レジスタと外部データバスとの関係

## バイトアクセス(LDUB,STB命令実行時)

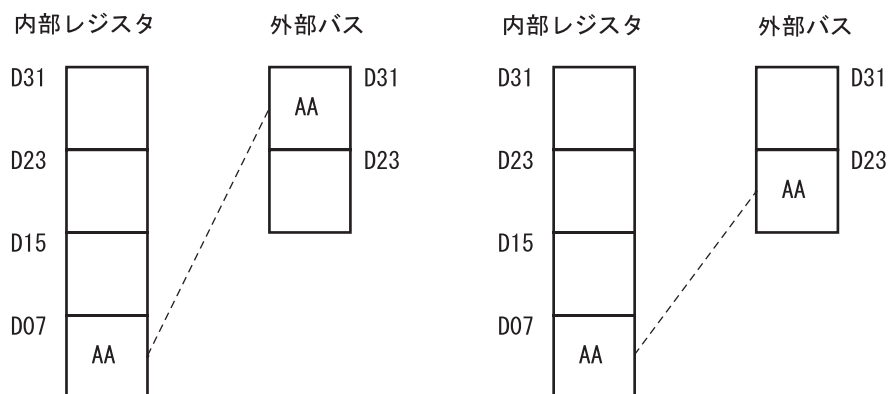


図 4.4-14 バイトアクセスの内部レジスタと外部データバスとの関係

データバス幅

データバス幅ごとに、内部レジスタと外部データバスとの関係を示します。

16ビットバス幅

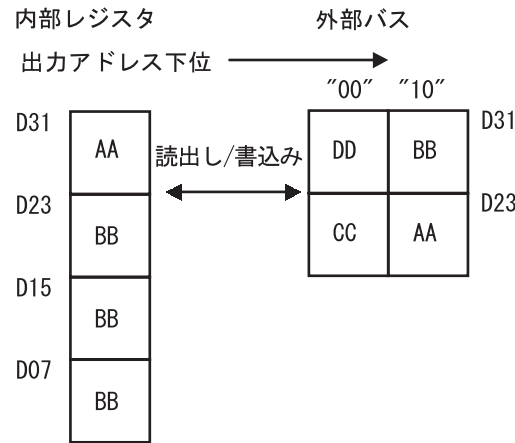


図 4.4-15 16ビットバス幅の内部レジスタと外部データバスとの関係

8ビットバス幅

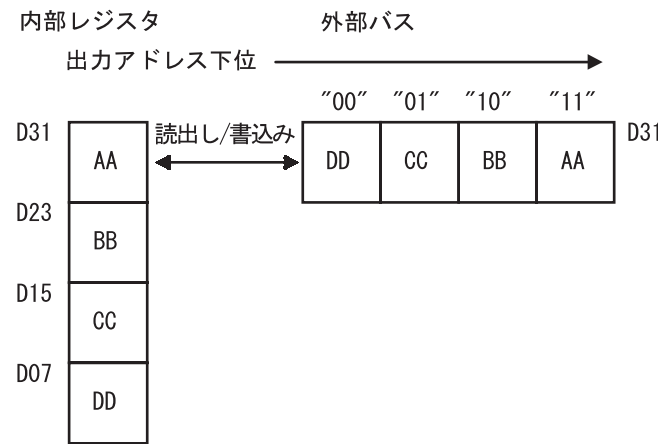


図 4.4-16 8ビットバス幅の内部レジスタと外部データバスとの関係

## 外部デバイスとの接続例

図 4.4-17と図 4.4-18に, MB91127/F128と外部デバイスとの接続例をバス幅ごとに示します。

## 16ビットバス幅

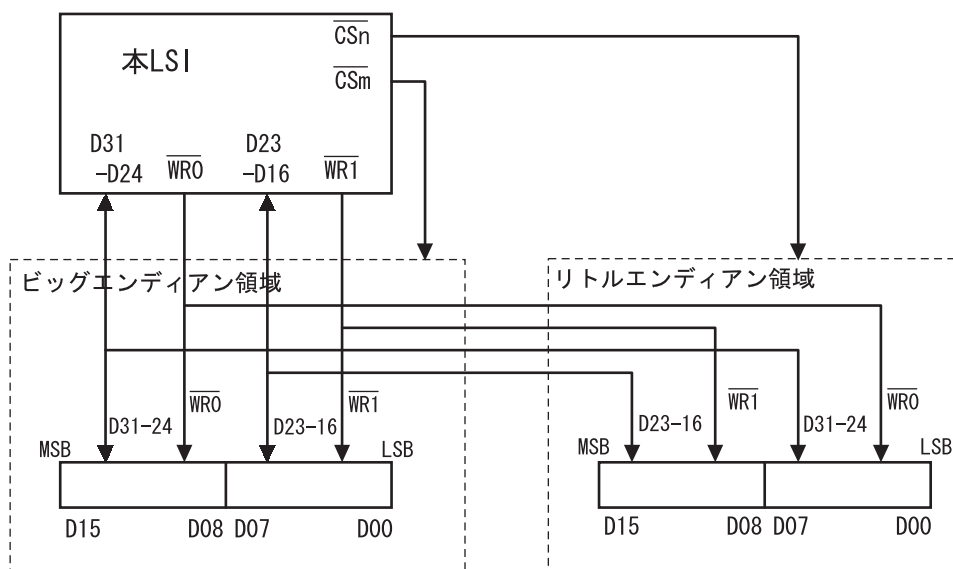


図 4.4-17 MB91F127/128と外部デバイスとの接続例(16ビットバス幅)

## 8ビットバス幅

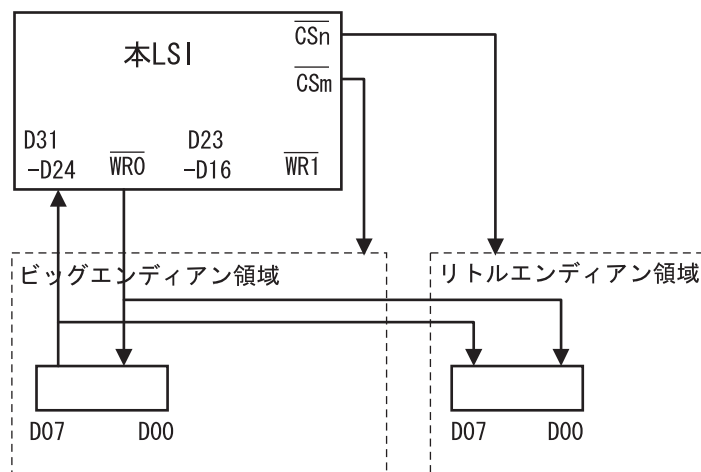


図 4.4-18 MB91F127/128外部デバイスとの接続例(8ビットバス幅)

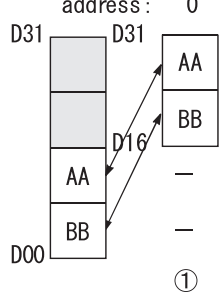
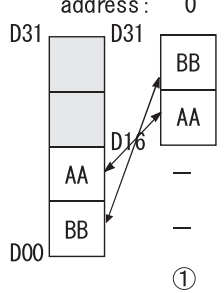
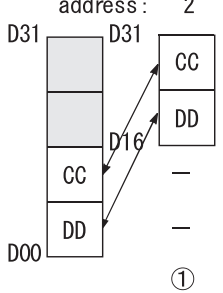
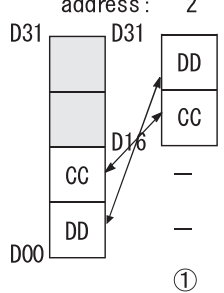
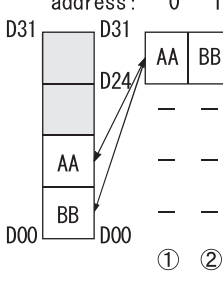
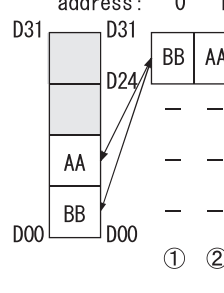
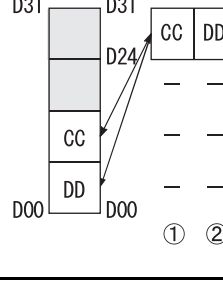
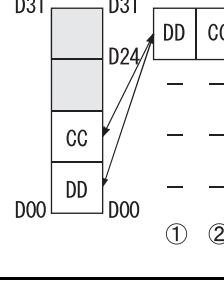
4.4.4 ビッグエンディアンとリトルエンディアンの外部アクセスの比較

バス幅に対するワードアクセス, ハーフワードアクセス, およびバイトアクセスについて, ビッグエンディアンとリトルエンディアンの外部アクセスの比較を示します。

ワードアクセス

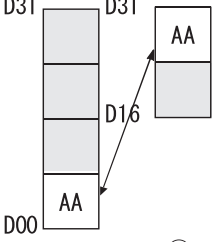
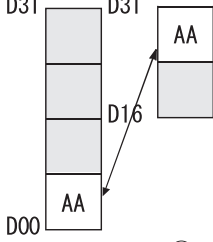
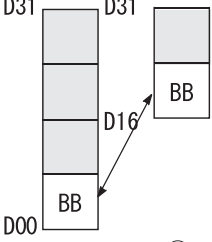
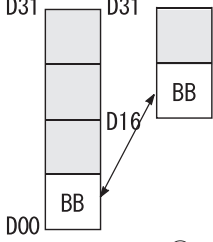
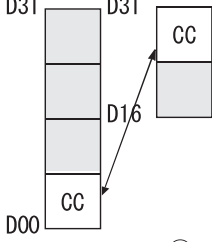
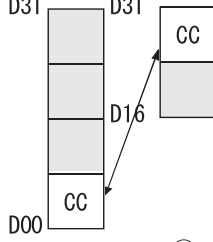
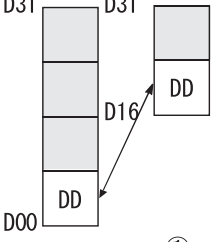
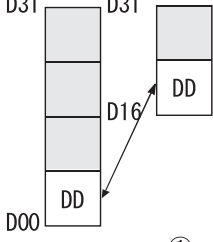
	ビッグエンディアンモード	リトルエンディアンモード
16ビット バス幅	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "0" "2"</div> <div><div>D31 AA BB CC DD D00</div><div>D31 AA   CC BB   DD D16</div><div>—   — —   — —   —</div><div>①   ②</div></div> <div><div>WRO   CASL   WEL</div><div>WR1   CASH   WEH</div><div>—   —   — —   —   —</div></div>	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "0" "2"</div> <div><div>D31 AA BB CC DD D00</div><div>D31 DD   BB CC   AA D16</div><div>—   — —   — —   —</div><div>①   ②</div></div> <div><div>WRO   CASL   WEL</div><div>WR1   CASH   WEH</div><div>—   —   — —   —   —</div></div>
8ビット バス幅	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "0" "1" "2" "3"</div> <div><div>D31 AA BB CC DD D00</div><div>D31 AA   BB   CC   DD D24</div><div>—   —   —   — —   —   —   — —   —   —   —</div><div>①   ②   ③   ④</div></div> <div><div>WRO   CASL   WEL</div><div>—   —   —</div><div>—   —   — —   —   —</div></div>	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "0" "1" "2" "3"</div> <div><div>D31 AA BB CC DD D00</div><div>D31 DD   CC   BB   AA D24</div><div>—   —   —   — —   —   —   — —   —   —   —</div><div>①   ②   ③   ④</div></div> <div><div>WRO   CASL   WEL</div><div>—   —   —</div><div>—   —   — —   —   —</div></div>

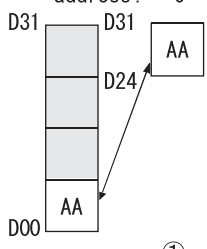
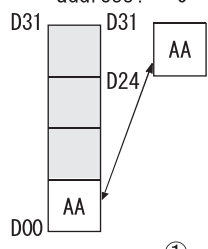
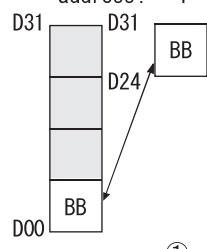
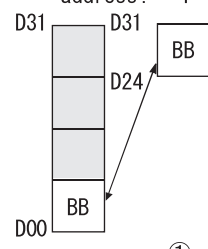
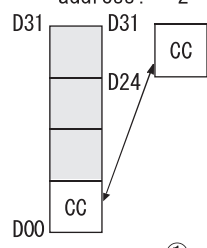
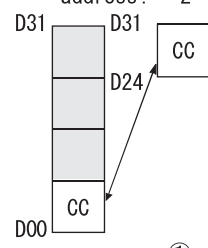
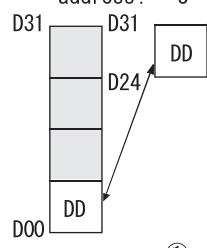
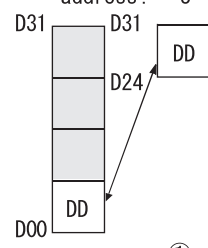
## ハーフワードアクセス

	ビッグエンディアンモード	リトルエンディアンモード
16ビット バス幅	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "0"</p>  <p>①</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "0"</p>  <p>①</p>
	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "2"</p>  <p>①</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "2"</p>  <p>①</p>
8ビット バス幅	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "0" "1"</p>  <p>① ②</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "0" "1"</p>  <p>① ②</p>
	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "2" "3"</p>  <p>① ②</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "2" "3"</p>  <p>① ②</p>



バイトアクセス

	ビッグエンディアンモード			リトルエンディアンモード																						
16ビット バス幅	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "0"</div> <div><div>①</div></div> <div><table><tr><td>WRO</td><td>CASL</td><td>WEL</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	WRO	CASL	WEL	—	—	—	—	—	—	—	—	—	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "0"</div> <div><div>①</div></div> <div><table><tr><td>WRO</td><td>CASL</td><td>WEL</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	WRO	CASL	WEL	—	—	—	—	—	—	—	—	—
	WRO	CASL	WEL																							
	—	—	—																							
	—	—	—																							
	—	—	—																							
WRO	CASL	WEL																								
—	—	—																								
—	—	—																								
—	—	—																								
<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "1"</div> <div><div>①</div></div> <div><table><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>WR1</td><td>CASH</td><td>WEH</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	—	—	—	WR1	CASH	WEH	—	—	—	—	—	—	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "1"</div> <div><div>①</div></div> <div><table><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>WR1</td><td>CASH</td><td>WEH</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	—	—	—	WR1	CASH	WEH	—	—	—	—	—	—	
—	—	—																								
WR1	CASH	WEH																								
—	—	—																								
—	—	—																								
—	—	—																								
WR1	CASH	WEH																								
—	—	—																								
—	—	—																								
<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "2"</div> <div><div>①</div></div> <div><table><tr><td>WRO</td><td>CASL</td><td>WEL</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	WRO	CASL	WEL	—	—	—	—	—	—	—	—	—	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "2"</div> <div><div>①</div></div> <div><table><tr><td>WRO</td><td>CASL</td><td>WEL</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	WRO	CASL	WEL	—	—	—	—	—	—	—	—	—	
WRO	CASL	WEL																								
—	—	—																								
—	—	—																								
—	—	—																								
WRO	CASL	WEL																								
—	—	—																								
—	—	—																								
—	—	—																								
<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "3"</div> <div><div>①</div></div> <div><table><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>WR1</td><td>CASH</td><td>WEH</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	—	—	—	WR1	CASH	WEH	—	—	—	—	—	—	<div>内部レジスタ    外部端子                      制御端子</div> <div>address: "3"</div> <div><div>①</div></div> <div><table><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>WR1</td><td>CASH</td><td>WEH</td></tr><tr><td>—</td><td>—</td><td>—</td></tr><tr><td>—</td><td>—</td><td>—</td></tr></table></div>	—	—	—	WR1	CASH	WEH	—	—	—	—	—	—	
—	—	—																								
WR1	CASH	WEH																								
—	—	—																								
—	—	—																								
—	—	—																								
WR1	CASH	WEH																								
—	—	—																								
—	—	—																								

	ビッグエンディアンモード	リトルエンディアンモード
8ビット バス幅	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "0"</p>  <p>①</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "0"</p>  <p>①</p>
	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "1"</p>  <p>①</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "1"</p>  <p>①</p>
	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "2"</p>  <p>①</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "2"</p>  <p>①</p>
	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "3"</p>  <p>①</p>	<p>内部レジスタ 外部端子 制御端子</p> <p>address: "3"</p>  <p>①</p>

## 4.5 バスタイミング

次の項目について、各モードでのバスアクセスのタイミング図と動作について説明します。

- ・通常バスアクセス
- ・ウェイトサイクル
- ・時分割入出力インタフェース
- ・外部バスリクエスト

### 通常バスアクセス

通常バスインタフェースでは、リードサイクル/ライトサイクルともに、“2クロックサイクル”が基本バスサイクルになります。このマニュアルでは、その2サイクルを“BA1”、“BA2”で表します

- 基本リードサイクル
- 基本ライトサイクル
- 各モードでのリードサイクル
- 各モードでのライトサイクル
- リード/ライト混合サイクル

### ウェイトサイクル

ウェイトサイクルには、エリアモードレジスタ(AMD)のWTCビットによる自動ウェイトサイクルと、RDY端子を使用する外部ウェイトサイクルがあります。

ウェイトサイクルは、前のサイクルを継続して行うモードであり、ウェイトが解除されるまで、“BA1サイクル”を繰り返します。

- 自動ウェイトサイクル
- 外部ウェイトサイクル

### 時分割入出力インタフェース

領域1ではアドレス/データの時分割入出力インタフェースがサポートされます。時分割入出力はBW1,0に指定されたバス幅において行われます。

時分割入出力インタフェースは、アドレス出力サイクル2クロック+データアクセスサイクル2クロックの計4クロックサイクルが基本バスサイクルになり、アドレス出力サイクルでは、出力アドレスラッチ信号としてALE端子をアサートします。

### 外部バスリクエスト

- バス権解放
- バス権獲得

### 4.5.1 基本リードサイクル

基本リードサイクルの動作タイミングを示します。

## 基本リードサイクルタイミング

バス幅:16ビット,アクセス:ワード,CS0領域のアクセス

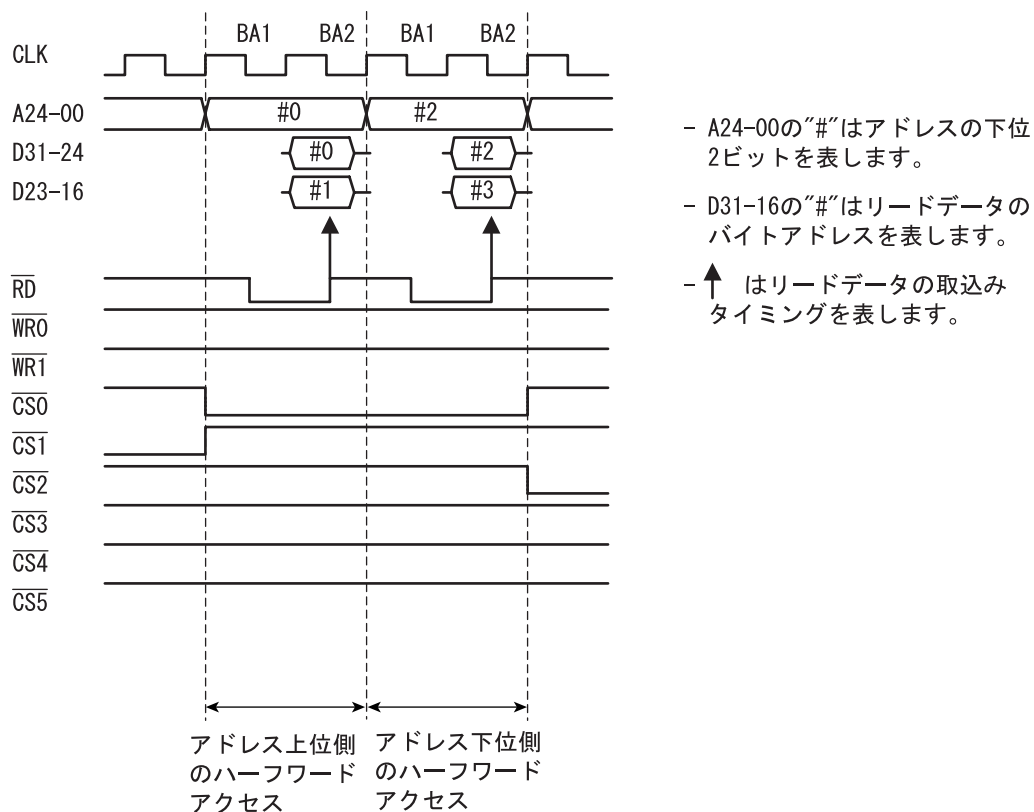


図 4.5-1 基本リードサイクルのタイミング図

【動作説明】

- CLKは、外部バスの動作クロックを出力します。  
ギアがかかったときには、ギア比に応じてCLKの周波数も下がります。
  - A24～A00(アドレス24～00)は、リードサイクルのワード/ハーフワード/バイトアクセスの先頭バイト位置のアドレスをバスサイクルの開始(BA1)から出力します。
- 図 4.5-1の場合、16ビットバス幅でワードアクセスを行うため、1回目のバスサイクルでワードアクセスの上位16ビットのアドレス(下位2ビット"0")を、2回目のバスサイクルで下位16ビットのアドレス(下位2ビット"2")を出力します。

- D31 ~ D16(データ31 ~ 16)は, 外部メモリ, I/Oからのリードデータを表します。リードサイクルの場合は,  $\overline{RD}$  の立ち上がりでD31 ~ D16を取り込みます。なお, リードサイクルの場合は, バス幅やワード/ハーフワード/バイトアクセスに関係なく,  $\overline{RD}$  の立ち上がりでD31 ~ D16をすべて取り込み, 取り込んだデータが有効かどうかの判断はチップ内部で行います。
- $\overline{RD}$  は, 外部データバスのリードストローク信号であり, BA1の立ち下がりでアサートし, BA2の立ち下がりでネゲートします。
- リードサイクルの場合は  $\overline{WR0}$ ,  $\overline{WR1}$  はネゲート状態です。
- $\overline{CS0}$  ~  $\overline{CS5}$  (領域チップセレクト)信号の出力は, A24 ~ A00と同タイミングでバスサイクルの開始(BA1)からアサートされます。 $\overline{CS0}$  ~  $\overline{CS5}$  はアドレス出力をデコードして作っており, アドレス出力が変化してASR, AMRで設定したチップセレクト領域が変わらない限り変化しません。  
また,  $\overline{CS0}$  ~  $\overline{CS5}$  は, いつも必ずどれかはアサートされています。

4.5.2 基本ライトサイクル

基本ライトサイクルの動作タイミングを示します。

基本ライトサイクルの動作タイミング

バス幅:8ビット,アクセス:ワード,CS0領域のアクセス

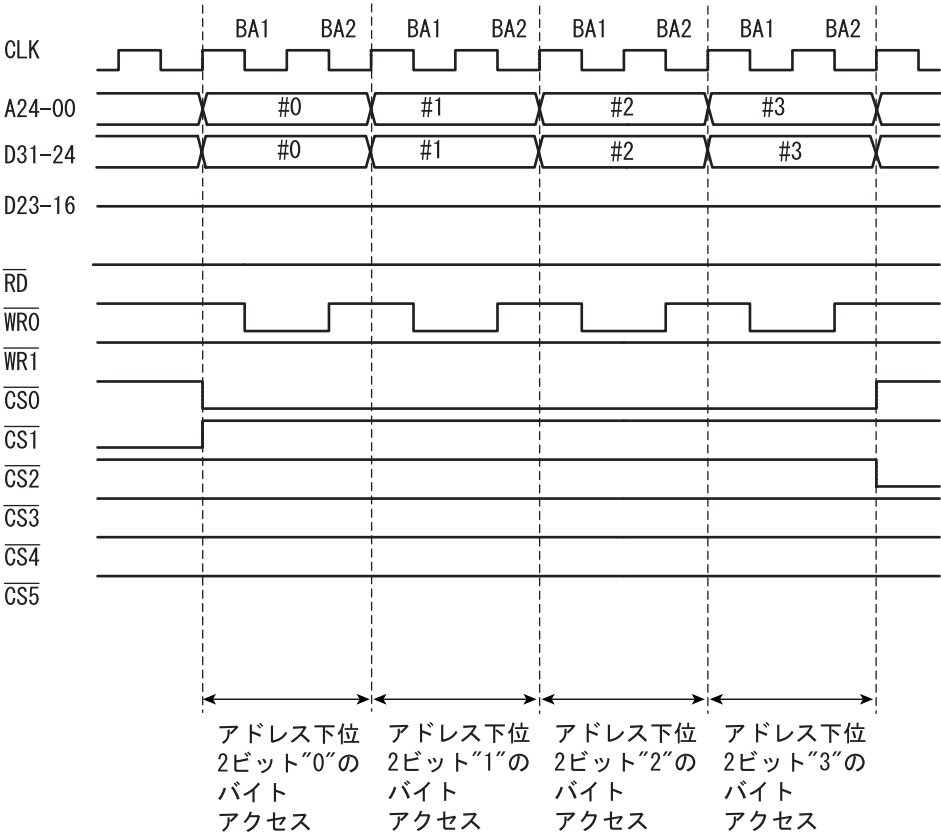


図 4.5-2 基本ライトサイクルの動作タイミング例

【動作説明】

- A24 ~ A00(アドレス24 ~ 00)は、ライトサイクルのワード/ハーフワード/バイトアクセスの先頭バイト位置のアドレスをバスサイクルの開始(BA1)から出力します。図 4.5-2の場合,8ビットバス幅でワードアクセスを行うため,まずワードアクセスの先頭バイト(アドレス下位"0")のアドレスを出力し,その後は順々に先頭バイトから+1のアドレス("1"),+2のアドレス("2"),+3のアドレス("3")を出力します。
- D31 ~ D16(データ31 ~ 16)は、外部メモリ,I/Oへのライトデータを表します。ライトサイクルの場合は,ライトデータをバスサイクルの開始(BA1)から出力し,バスサイクルの終了(BA2の終了)でHi-zにします。図 4.5-2の場合,8ビットデータバス幅のため,D31 ~ D24にライトデータが出力されます。
- ライトサイクルでは,  $\overline{RD}$  はネゲート状態です。

- $\overline{WR0}$  ,  $\overline{WR1}$  は、外部データバスのライトストローク信号であり、BA1の立ち下がりでアサートし、BA2の立ち下がりでネゲートします。D31～D24は $\overline{WR0}$  , D23～D16は $\overline{WR1}$  と、それぞれ対応するデータバスに応じてアサートされます。

図 4.5-2の場合は、8ビットデータバス幅のため、 $\overline{WR0}$  のみアサートされます。

- チップセレクト領域0～5の最大バス幅が8ビットの場合、すなわち設定されたすべての領域が8ビットの場合は、D23～D16と $\overline{WR1}$  は自動的にI/Oポートになり、Hi-zとなります。

図 4.5-2は、D23～D16と $\overline{WR1}$  がI/Oポートとして使用される場合を示しています。また、チップセレクト領域0～5のうち、バス幅がどれか1領域でも16ビットに設定されていると、D23～D16と $\overline{WR1}$  はI/Oポートとして使用できませんので、ご注意ください。

端子	D31-24	D23-16
最大バス幅	$\overline{WR0}$	$\overline{WR1}$
16ビット	D31-24 $\overline{WR0}$	D23-16 $\overline{WR1}$
8ビット	D31-24 $\overline{WR0}$	I/Oポート

- DACK0～DACK2, EOP0～EOP2はDMAの外部バスサイクルで出力されます。出力するかどうかは、DMACのレジスタ設定により決定され、出力タイミングは $\overline{WR0}$ ～ $\overline{WR1}$ と同じです。

### 4.5.3 各モードでのリードサイクル

各モードでのリードサイクルの動作タイミングを示します。

各モードでのリードサイクルの動作タイミング

バス幅:16ビット,アクセス:ハーフワード

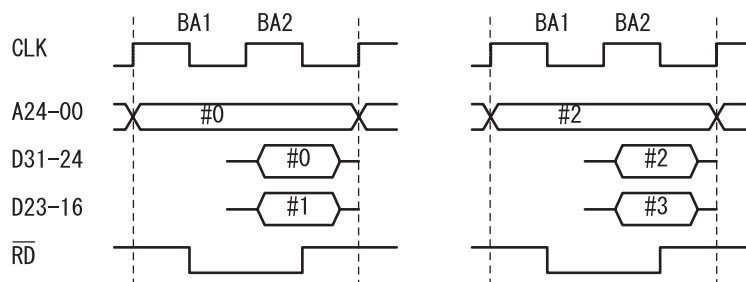
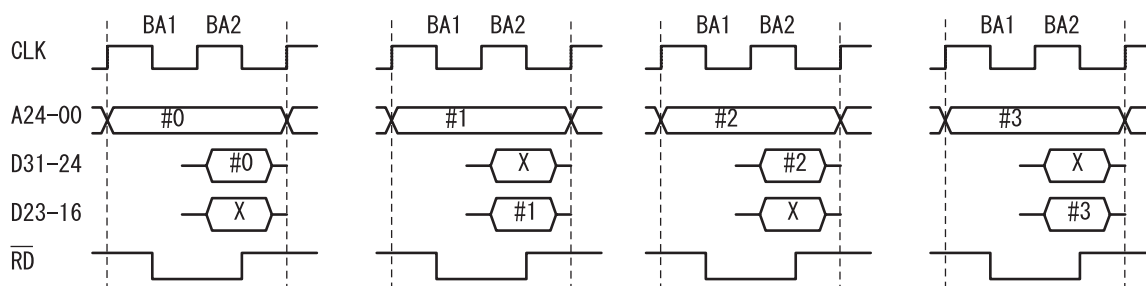


図 4.5-3 リードサイクルのタイミング例1

バス幅:16ビット,アクセス:バイト



X: 無効データ入力

図 4.5-4 リードサイクルのタイミング例2

バス幅:8ビット,アクセス:ワード

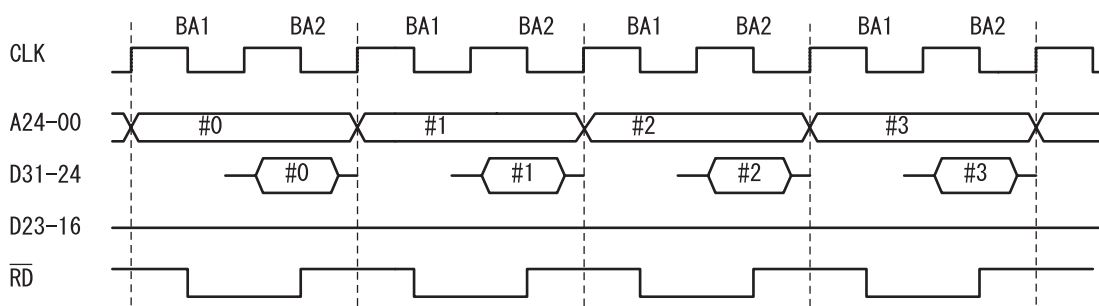


図 4.5-5 リードサイクルのタイミング例3



バス幅:8ビット,アクセス:ハーフワード

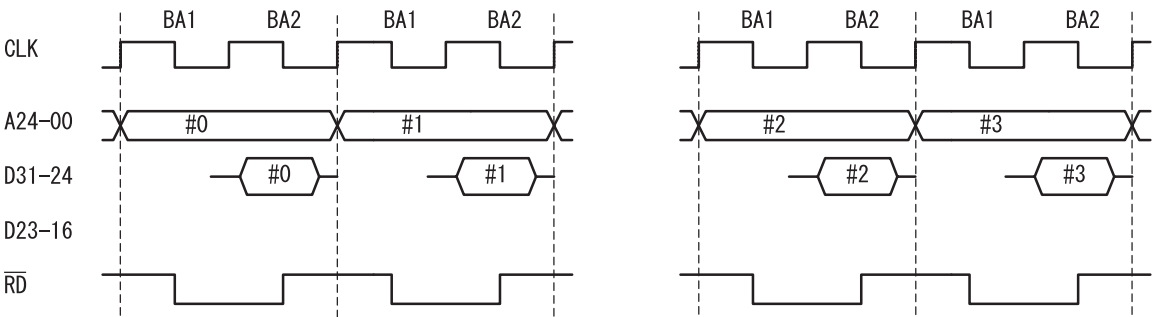


図 4.5-6 リードサイクルのタイミング例4

バス幅:8ビット,アクセス:バイト

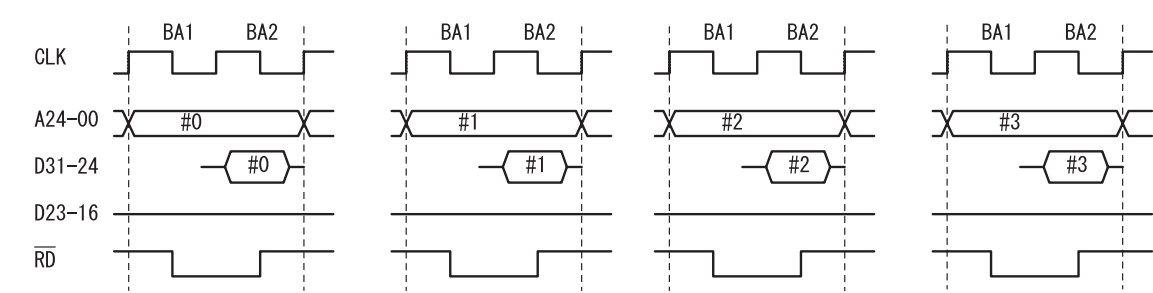


図 4.5-7 リードサイクルのタイミング例5

### 4.5.4 各モードでのライトサイクル

各モードでのライトサイクルの動作タイミングを示します。

ライトサイクルタイミング

バス幅:16ビット,アクセス:ワード

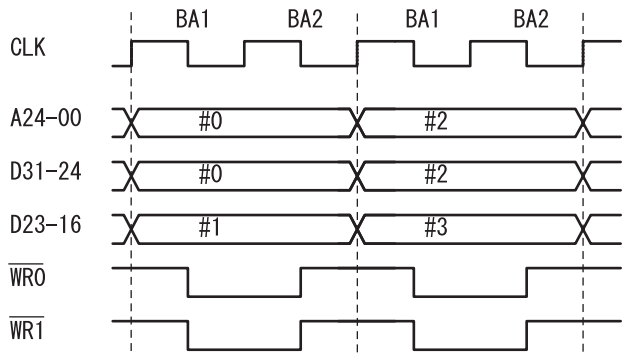


図 4.5-8 ライトサイクルのタイミング例1

バス幅:16ビット,アクセス:ハーフワード

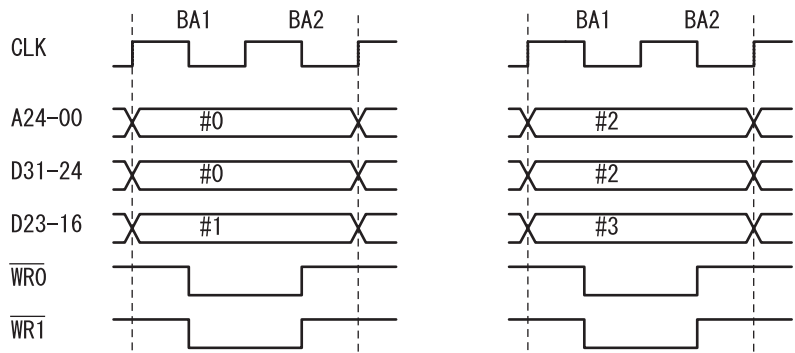


図 4.5-9 ライトサイクルのタイミング例2

バス幅:16ビット,アクセス:バイト

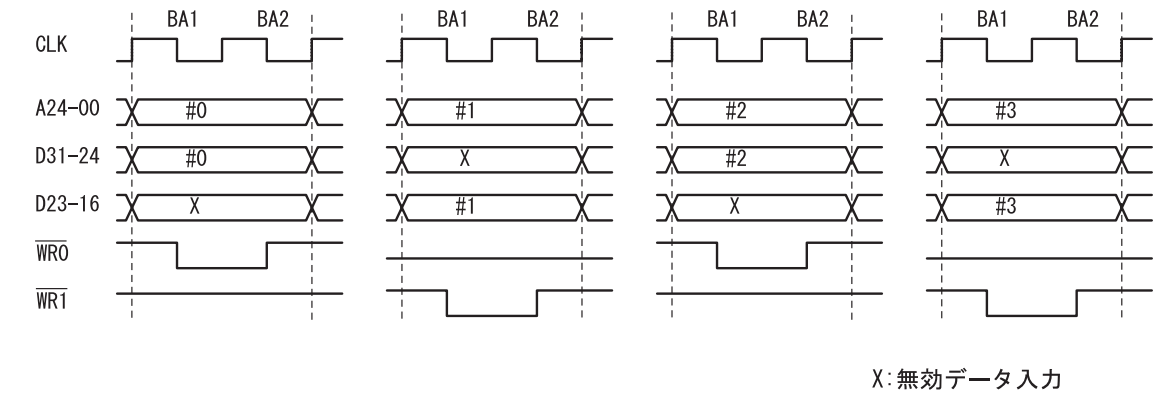


図 4.5-10 ライトサイクルのタイミング例3

バス幅:8ビット,アクセス:ハーフワード

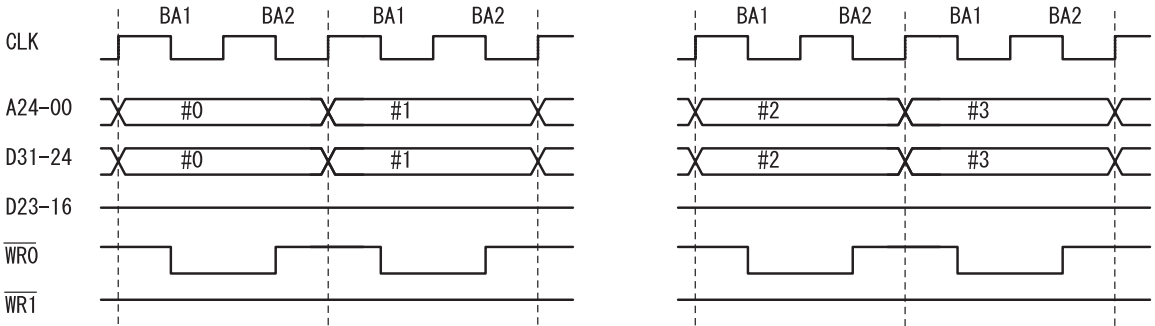


図 4.5-11 ライトサイクルのタイミング例4

バス幅:8ビット,アクセス:バイト

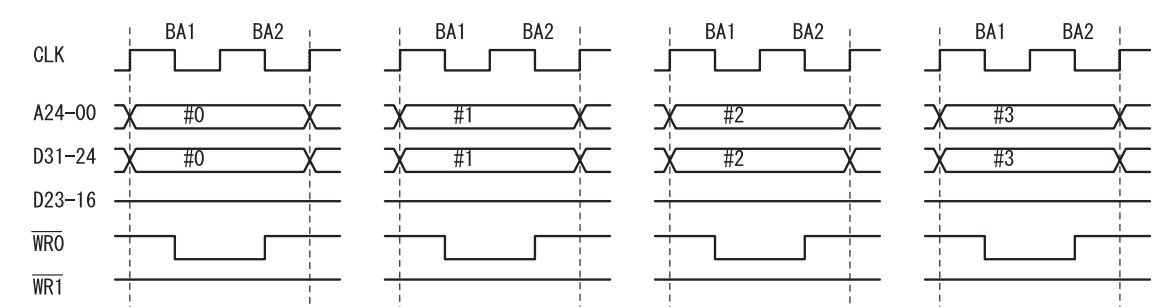


図 4.5-12 ライトサイクルのタイミング例5

### 4.5.5 リード/ライト混在サイクル

リード/ライト混在サイクルの動作タイミングを示します。

リード/ライト混在サイクルタイミング

CS0領域: バス幅16ビット, ワードリード  
CS1領域: バス幅 8ビット, ハーフワードリード

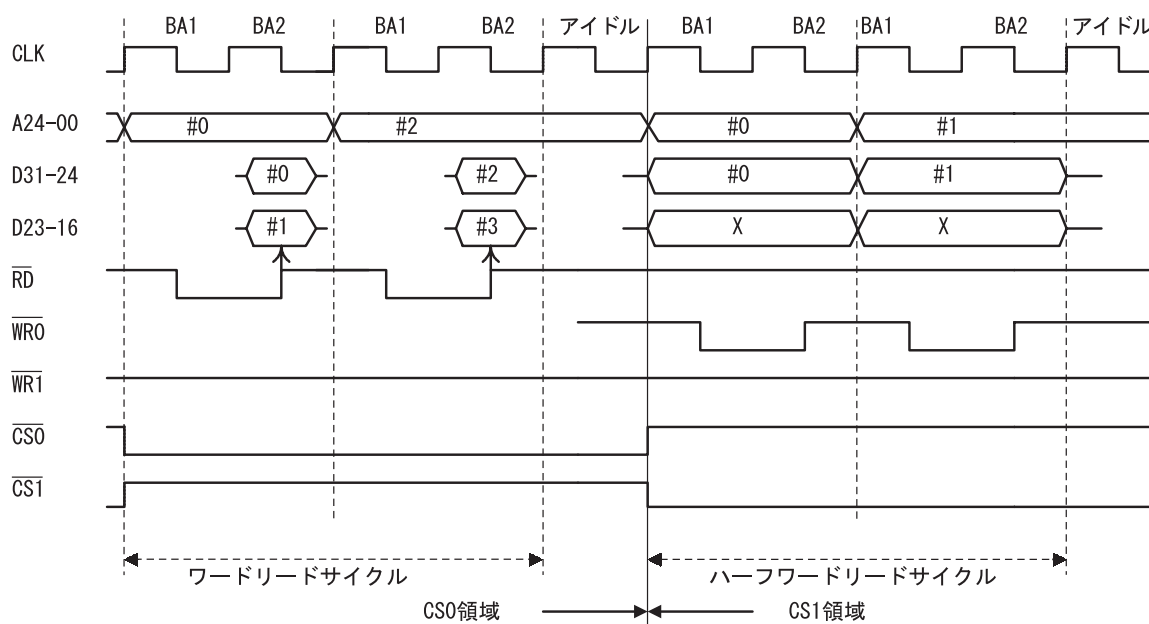


図 4.5-13 リード/ライト混在サイクルのタイミング例1

#### 【動作説明】

- 図 4.5-13は, チップセレクト領域の切り替わりに, アイドルサイクル(何もバスサイクルをおこなっていないサイクル)が挿入された場合を表しています。バスサイクルの間にアイドルサイクルが挿入された場合は, アドレスは次のバスサイクルが開始されるまで前のバスサイクルのアドレスをそのまま出力し続けます。  
それに伴い, 出力しているアドレスに対応する  $\overline{CS0} \sim \overline{CS5}$  もアサートし続けます。
- 図 4.5-13は, 16ビット/8ビットバスの混在です。  
最大バス幅が16ビットのため, 8ビットアクセス領域(CS1領域)でも, D23～D16と  $\overline{WR1}$  はI/Oポートになりません。D23～D16は不定データが出力され,  $\overline{WR1}$  はネゲート状態になります。

## 4.5.6 自動ウェイトサイクル

自動ウェイトサイクルの動作タイミングを示します。

自動ウェイトサイクルタイミング

バス幅:16ビット,アクセス:ハーフワードリード/ライト

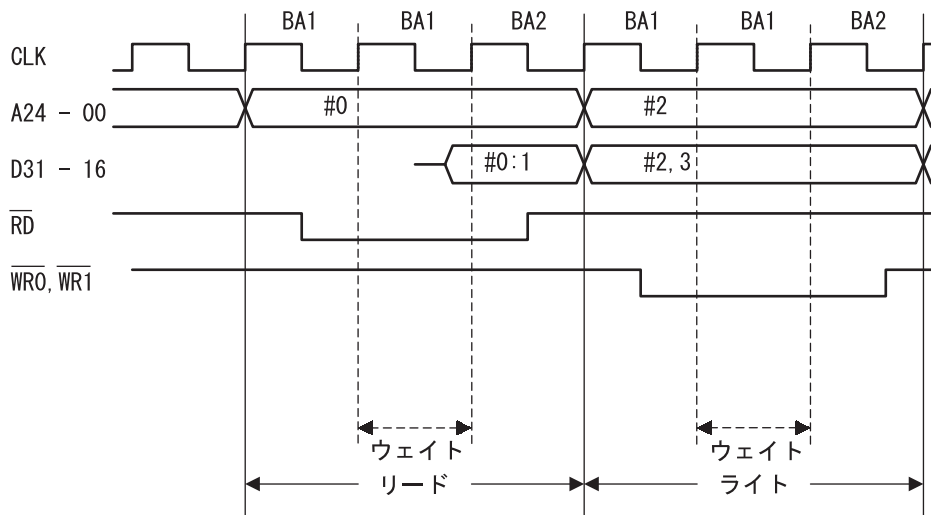


図 4.5-14 自動ウェイトサイクルのタイミング例

### 【動作説明】

- 自動ウェイトサイクルは,それぞれのチップセレクト領域のエリアモードレジスタ(AMD)のWTCビットを設定することにより実現できます。
- 図 4.5-14の場合は,WTCビットに"001"を設定して,通常バスサイクルに1ウェイトのバスサイクルを挿入した例を示しており,その場合は"通常バスサイクル2クロック"+ "ウェイトサイクル1クロック"の"計3クロックバスサイクル"になります。  
自動ウェイトは,最大7クロックサイクルまで(その場合,通常バスサイクルは9クロックサイクル)設定することができます。

#### 4.5.7 外部ウェイトサイクル

外部ウェイトサイクルの動作タイミングを示します。

外部ウェイトサイクルタイミング

バス幅:16ビット,アクセス:ハーフワード

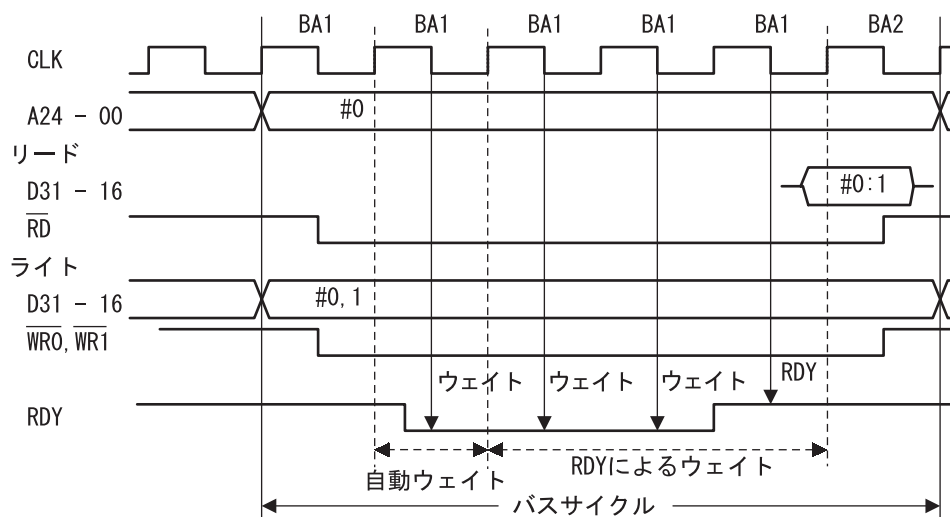


図 4.5-15 外部ウェイトサイクルのタイミング例

【動作説明】

- 外部ウェイトサイクルは、EPCROのRDYEビットを”1”に設定し、外部RDY端子の入力を有効にすることにより実現できます。
- 外部RDYを使用する場合は、必ず1クロック以上の自動ウェイトサイクル、すなわちAMDのWTCビットに”001”以上の値を設定してください。自動ウェイトサイクル中はRDYの検出はせず、自動ウェイトサイクル後検出します。
- また、外部RDYは、CLK端子出力の立ち下がりに同期して入力してください。CLKの立ち下がり時点で外部RDYが”L”レベルならばウェイトサイクルとなり、同じBA1サイクルを繰り返します。また、”H”レベルのときにはウェイトサイクル終了と判断し、BA2サイクルに遷移します。

## 4.5.8 時分割入出力インタフェース

時分割入出力インタフェースの動作タイミングを示します。

時分割入出力インタフェースの動作タイミング

8ビットバス幅リード

図 4.5-16に、下記条件の動作タイミング例を示します。

- AMD1:MPX=1,BW=00B,WTC=000B,EPCRO:ALEE=1
- アクセス:8ビットデータリード

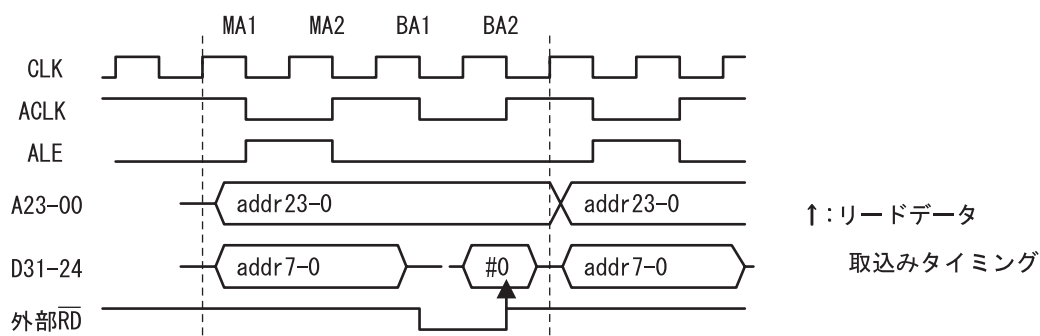


図 4.5-16 8ビットバス幅リード動作タイミング例

8ビットバス幅のときには、D31-24にアドレスA07-00が出力されます。

時分割モードでの自動ウェイト動作(16ビットバス幅ライト)

図 4.5-17に、下記条件の動作タイミング例を示します。

- AMD1:MPX=1,BW=01B,WTC=001B,EPCRO:ALEE=1
- アクセス:16ビットデータリード

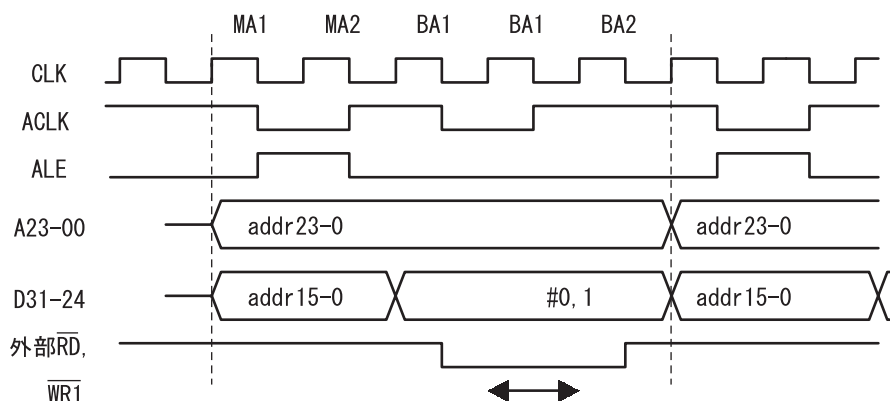


図 4.5-17 時分割モードでの自動ウェイト動作タイミング例

16ビットバス幅のときには、D31-16にアドレスA15-00が出力されます。

## 時分割モードでの外部ウェイト動作

図 4.5-18に, 下記条件の動作タイミング例を示します。

- AMD1:MPX=1, BW=00B, WTC=010B, EPCR0:ALEE=1
- アクセス:8ビットデータリード

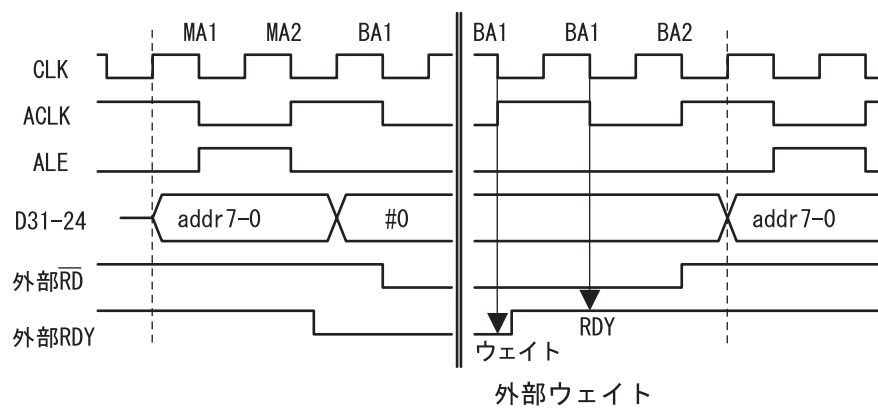


図 4.5-18 時分割モードでの外部ウェイト動作タイミング例



## 4.5.9 外部バスリクエスト

外部バスリクエストの動作タイミングを示します。

### バス権解放

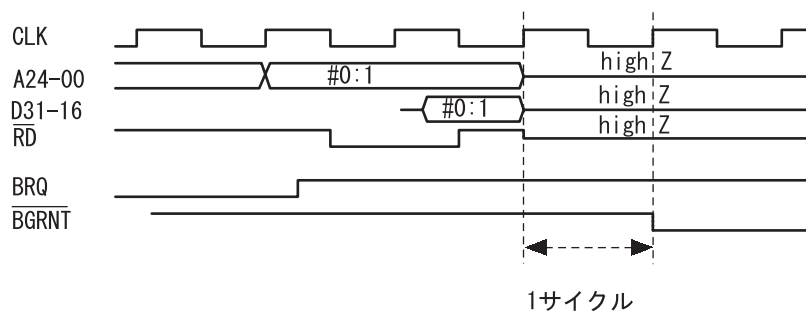


図 4.5-19 バス権解放のタイミング例

#### 【動作説明】

- EPCROのBREビットを”1”に設定することによりBRQ,  $\overline{\text{BGRNT}}$  によるバスアービトレーションを行うことができます。
- バス権解放時は、端子をHigh-Zにしてから1サイクル後に  $\overline{\text{BGRNT}}$  をアサートします。

### バス権獲得

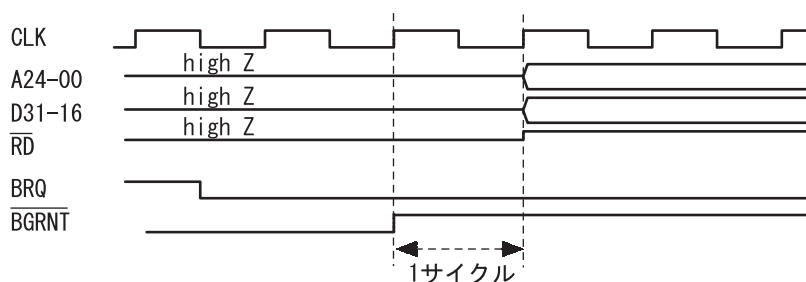


図 4.5-20 バス権獲得のタイミング例

#### 【動作説明】

- EPCROのBREビットを”1”に設定することにより, BRQ,  $\overline{\text{BGRNT}}$  によるバスアービトレーションを行うことができます。
- バス権解放時は,  $\overline{\text{BGRNT}}$  をネゲートしてから1クロック後に各端子をアクティブにします。

## 4.6 内部クロック逡倍動作(クロックダブラ)

MB91F127/128はクロック逡倍回路を持っており,CPU内部はバスインタフェースの2倍,1倍のいずれかの周波数で動作します。どちらのクロックを選択した場合でも,バスインタフェースはCLKからの外部アクセス要求が発生した場合,CLK出力の立ち上がりを待って外部へのアクセスを開始します。

### クロック選択方法

2倍,1倍のクロック選択の方法は「3.11.12 クロックダブラ機能」を参照してください。

クロック選択の変更は,チップ動作中でも任意に行うことができます。クロック選択の切替え中は,バス動作が一時抑止されます。また,リセット時には,クロック選択は自動的に1倍になります。

図 4.6-1に2倍クロックの,図 4.6-2に1倍クロックのタイミング例を示します。

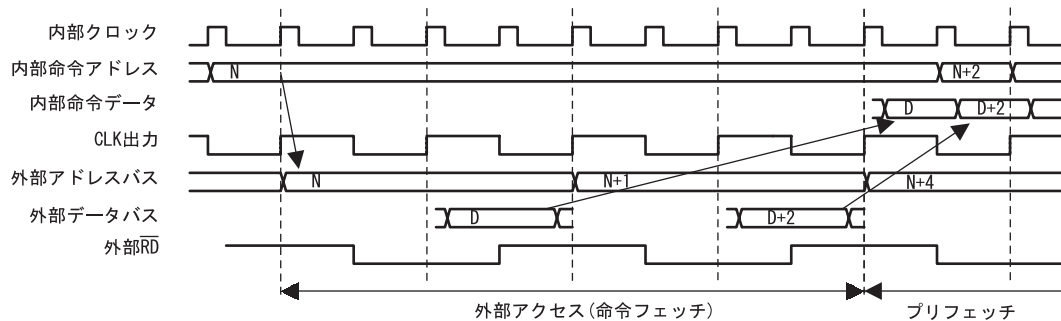


図 4.6-1 2倍クロックの場合のタイミング例(BW-16ビット,アクセス:ワードリード)

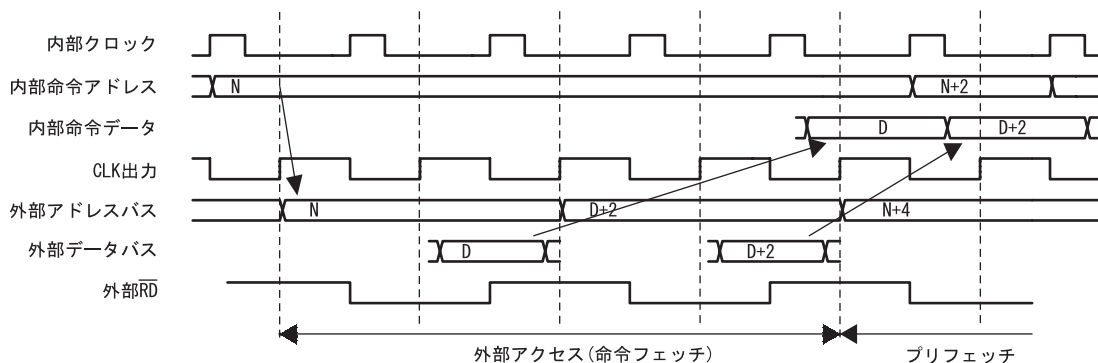


図 4.6-2 1倍クロックの場合のタイミング例(BW-16ビット,アクセス:ワードリード)

## 4.7 外部バスのプログラム例

---

外部バスを動作させるための簡単なプログラム例を示します。

---

外部バス動作のプログラム仕様例

レジスタの設定は以下のとおりです。

領域

- 領域0(AMD0): 16ビット, 通常バス, 自動ウェイト0
- 領域1(AMD1): 16ビット, 通常バス, 自動ウェイト2
- 領域2(AMD32): 16ビット, 通常バス, 自動ウェイト1
- 領域3(AMD32): 16ビット, 通常バス, 自動ウェイト1
- 領域4(AMD4): 16ビット, DRAM, ページサイズ256, 1CAS/2WE, ウェイトあり, CBRリフレッシュ
- 領域5(AMD5): 16ビット, DRAM, ページサイズ512, 2CAS/1WE, ウェイトなし, CBRリフレッシュ

ほかのバス

- リフレッシュ(RFCR): ウェイトなし, 1/8設定
- 外部端子(EPCR0): 外部RDY受付け, BRQ,  $\overline{\text{BGRNT}}$  のアービトレーション
- 外部端子(DSCR): DRAM端子の設定
- リトルエンディアン(LER): 領域2

その他, 以下の点に注意してください。

- MD2, MD1, MD0端子は"001"外部ベクタは16ビットモード
- 領域0を同じバス幅に設定してから, モードレジスタ(MODR)を設定
- 領域1~5はオーバーラップしないように設定

## 外部バス動作のプログラム例

このプログラムは説明のためにバイトレジスタはバイトで、ハーフワードレジスタはハーフワードで書き込みを行っています。

## \*\*\*\*\* プログラム例 \*\*\*\*\*

```
//各レジスタ設定
init_epcr      ldi:20  #0xffff,r0    // 外部端子設定
                                     // 外部RDYウェイト,
                                     // BRQ, BGRNT バスアービトラージュ
                                     //
                                     // epcr0 レジスタ アドレス設定
                                     // epcr0 レジスタライト
init_dscr      ldi:8   #0xff,r0      // DRAM端子設定
                                     // RAS, CAS, WE
                                     // dscr レジスタ アドレス設定
                                     // dscr レジスタライト
init_amd0      ldi:20  #0x628,r1     // 16ビットバス,0-wait
stb            r0,@r1
init_amd0      ldi:8   #0x08,r0      // amd0 レジスタ アドレス設定
stb            r0,@r1               // amd0 レジスタライト
init_amd1      ldi:8   #0x0a,r0      // 16ビットバス,2-wait
stb            r0,@r1               // amd1 レジスタ アドレス設定
                                     // amd1 レジスタライト
init_amd32     ldi:8   #0x49,r0      // 通常,16ビットバス,1-wait
stb            r0,@r1               // amd32 レジスタ アドレス設定
                                     // amd32 レジスタライト
init_amd4      ldi:8   #0x88,r0      // DRAM, 16ビットバス
stb            r0,@r1               // amd4 レジスタ アドレス設定
                                     // amd4 レジスタライト
init_amd5      ldi:8   #0x88,r0      // DRAM, 16ビットバス
stb            r0,@r1               // amd5 レジスタ アドレス設定
                                     // amd5 レジスタライト
init_dmcr4     ldi:20  #0x0c90,r0    // page size=256,Q1/Q4-wait,Page
                                     // 1CAS-2WE, CBR, パリティなし
                                     // dmcr4 レジスタ アドレス設定
                                     // dmcr4 レジスタライト
init_dmcr5     ldi:20  #0x10c0,r0    //page size=512, Q1/Q4-waitなし,
                                     // Page
                                     // 2CAS-1WE, CBR, パリティなし
                                     // dmcr5 レジスタ アドレス設定
                                     // dmcr5 レジスタライト
init_rfcr      ldi:20  #0x0205,r0    // REL=2, R1W/R3W-waitなし,
                                     // refresh, 1/8
                                     // rfcr レジスタ アドレス設定
                                     // rfcr レジスタライト
init_asr       ldi:32  #0x0013001,r0 // asr1,amr1 レジスタ設定値
stb            r0,@r1               // asr2,amr2 レジスタ設定値
lidi:32        #0x0015001,r1        // asr3,amr3 レジスタ設定値
lidi:32        #0x0017001,r2        // asr4,amr4 レジスタ設定値
lidi:32        #0x0019001,r3        // asr5,amr5 レジスタ設定値
lidi:32        #0x001b001,r4        // asr1,amr1 レジスタ アドレス設定
lidi:20        #0x60c,r5             // asr2,amr2 レジスタ アドレス設定
lidi:20        #0x610,r6             // asr3,amr3 レジスタ アドレス設定
lidi:20        #0x614,r7             // asr4,amr4 レジスタ アドレス設定
lidi:20        #0x618,r8             // asr5,amr5 レジスタ アドレス設定
lidi:20        #0x61c,r9             // asr1,amr1 レジスタライト
st             r0,@r5
```

```

        st      r1,@r6          // asr2,amr2 レジスタライト
        st      r2,@r7          // asr3,amr3 レジスタライト
        st      r3,@r8          // asr4,amr4 レジスタライト
        st      r4,@r9          // asr5,amr5 レジスタライト
init_ler      ldi:8  #0x02,r0     // CS2 little endian
               ldi:20 #0x7fe,r1   // ler レジスタ アドレス設定
               stb    r0,@r1      // ler レジスタライト
init_modr     ldi:8  #0x80,r0     // 外ROM 外バスexternal bus
               ldi:20 #0x7ff,r1   // modrレジスタ アドレス設定
               stb    r0,@r1      // modrレジスタライト
//外部バスアクセス

adr_set       ldi:32 #0x00136da0,r0 // CS1 アドレス
               ldi:32 #0x00151300,r1 // CS2 アドレス
               ldi:32 #0x00196434,r2 // CS4 アドレス (ページ内)
               ldi:32 #0x0019657c,r3 // CS4 アドレス (ページ内)
               ldi:32 #0x00196600,r4 // CS4 アドレス (ページ外)
               ldi:32 #0x001a6818,r5 // CS5 アドレス (ページ内)
               ldi:32 #0x001a6b8c,r6 // CS5 アドレス (ページ内)
               ldi:32 #0x001a6c00,r7 // CS5 アドレス (ページ外)
bus_acc       ld      @r0, r8     // CS1 data word load
               lduh   @r1,r9      // CS2 data half word load
               ld      @r2,r10     // CS4 data word load
               ldub   @r3,r11     // CS4 data byte load
               st      r8,@r4      // CS4 data word store
               sth     r9,@r5      // CS5 data half word store
               st      r10,@r6     // CS5 data word store
               stb     r11,@r7     // CS5 data byte store

```

## 第5章 I/Oポート

---

この章では、I/Oポートの概要、レジスタの構成および外部端子をI/Oとして使用する条件について説明します。

---

- 5.1 I/Oポートの概要
- 5.2 ポートデータレジスタ(PDR)
- 5.3 データ方向レジスタ(DDR)
- 5.4 外部端子をI/Oポートとして使用

## 5.1 I/Oポートの概要

MB91F127/128は、各端子に対応するリソースが入出力として端子を使用しない設定になっているとき、I/Oポートとして使用することができます。

### I/Oポートの基本ブロックダイアグラム

図 5.1-1に、I/Oポートの基本的な構成を示します。

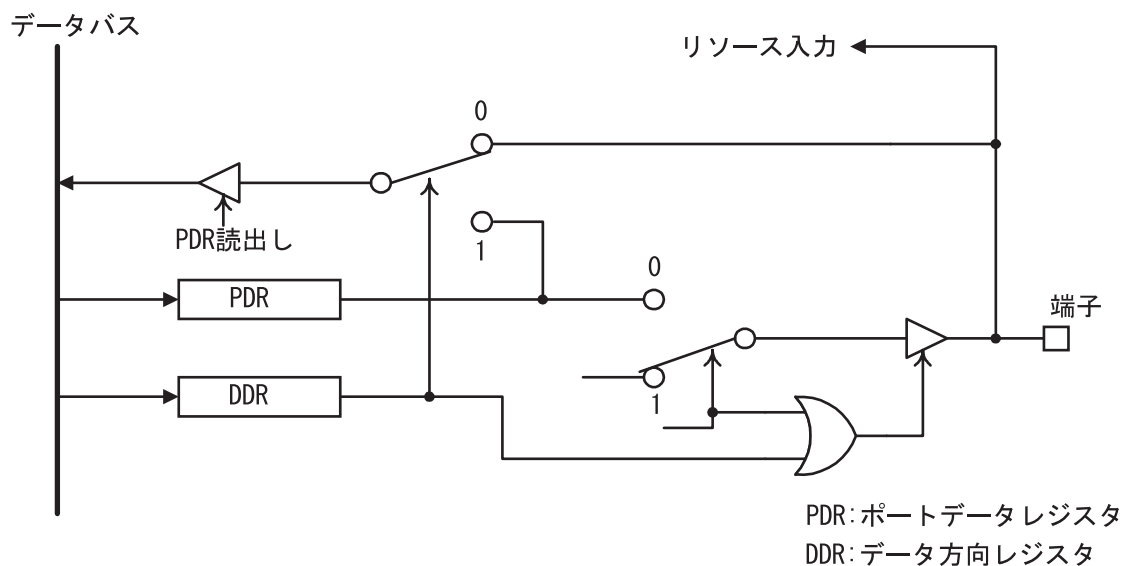


図 5.1-1 I/Oポートの基本ブロックダイアグラム

### I/Oポートのレジスタ

I/Oポートは、ポートデータレジスタ(PDR)とデータ方向レジスタ(DDR)で構成されています。

入力モード(DDR="0")時

- PDR読み出し時: 対応する外部端子のレベルが読み出されます。
- PDR書き込み時: PDRに設定値が書き込まれます。

出力モード(DDR="1")時

- PDR読み出し時: PDRの値が読み出されます。
- PDR書き込み時: PDRの値が対応する外部端子に出力されます。

## 5.2 ポートデータレジスタ(PDR)

ポートデータレジスタ(PDR2 ~ PDRJ)は、I/Oポートの入出力データレジスタです。  
対応するデータ方向レジスタ(DDR2 ~ DDRJ)で、入出力制御が行われます。

ポートデータレジスタ(PDR)の構成

図 5.2-1に、ポートデータレジスタ(PDR:Port Data Register)の構成を示します。

PDR2 bit	7	6	5	4	3	2	1	0	初期値
アドレス 000001 <sub>H</sub>	P27	P26	P25	P24	P23	P22	P21	P20	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDR3 bit	7	6	5	4	3	2	1	0	
アドレス 000000 <sub>H</sub>	P37	P36	P35	P34	P33	P32	P31	P30	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDR4 bit	7	6	5	4	3	2	1	0	
アドレス 000007 <sub>H</sub>	P47	P46	P45	P44	P43	P42	P41	P40	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDR5 bit	7	6	5	4	3	2	1	0	
アドレス 000006 <sub>H</sub>	P57	P56	P55	P54	P53	P52	P51	P50	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDR6 bit	7	6	5	4	3	2	1	0	
アドレス 000005 <sub>H</sub>	P67	P66	P65	P64	P63	P62	P61	P60	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDR7 bit	7	6	5	4	3	2	1	0	
アドレス 000004 <sub>H</sub>	—	—	—	—	—	—	—	P70	-----X <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDR8 bit	7	6	5	4	3	2	1	0	
アドレス 00000B <sub>H</sub>	—	P86	P85	P84	P83	P82	P81	P80	-XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDRA bit	7	6	5	4	3	2	1	0	
アドレス 000009 <sub>H</sub>	—	PA6	PA5	PA4	PA3	PA2	PA1	PA0	-XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDRE bit	7	6	5	4	3	2	1	0	
アドレス 000012 <sub>H</sub>	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDRF bit	7	6	5	4	3	2	1	0	
アドレス 000013 <sub>H</sub>	—	—	—	—	PF3	PF2	PF1	PF0	----XXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDRG bit	7	6	5	4	3	2	1	0	
アドレス 000014 <sub>H</sub>	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
PDRJ bit	7	6	5	4	3	2	1	0	
アドレス 000017 <sub>H</sub>	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0	XXXXXXXX <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 5.2-1 ポートデータレジスタ(PDR)の構成



## 5.3 データ方向レジスタ(DDR)

データ方向レジスタ(DDR2～DDRJ)は、対応するI/Oポートの入出力方向をビット単位で制御します。0で入力,1で出力制御が行われます。

データ方向レジスタ(DDR)の構成

図 5.3-1に、データ方向レジスタ(DDR:Data Direction Register)の構成を示します。

DDR2 bit	7	6	5	4	3	2	1	0	初期値
アドレス 000601 <sub>H</sub>	P27	P26	P25	P24	P23	P22	P21	P20	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDR3 bit	7	6	5	4	3	2	1	0	
アドレス 000600 <sub>H</sub>	P37	P36	P35	P34	P33	P32	P31	P30	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDR4 bit	7	6	5	4	3	2	1	0	
アドレス 000607 <sub>H</sub>	P47	P46	P45	P44	P43	P42	P41	P40	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDR5 bit	7	6	5	4	3	2	1	0	
アドレス 000606 <sub>H</sub>	P57	P56	P55	P54	P53	P52	P51	P50	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDR6 bit	7	6	5	4	3	2	1	0	
アドレス 000605 <sub>H</sub>	P67	P66	P65	P64	P63	P62	P61	P60	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDR7 bit	7	6	5	4	3	2	1	0	
アドレス 000604 <sub>H</sub>	—	—	—	—	—	—	—	P70	-----0 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDR8 bit	7	6	5	4	3	2	1	0	
アドレス 00060B <sub>H</sub>	—	P86	P85	P84	P83	P82	P81	P80	-0000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDRA bit	7	6	5	4	3	2	1	0	
アドレス 000609 <sub>H</sub>	—	PA6	PA5	PA4	PA3	PA2	PA1	PA0	-0000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDRE bit	7	6	5	4	3	2	1	0	
アドレス 0000D2 <sub>H</sub>	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDRF bit	7	6	5	4	3	2	1	0	
アドレス 0000D3 <sub>H</sub>	—	—	—	—	PF3	PF2	PF1	PF0	----0000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDRG bit	7	6	5	4	3	2	1	0	
アドレス 0000D8 <sub>H</sub>	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	
DDRJ bit	7	6	5	4	3	2	1	0	
アドレス 0000DB <sub>H</sub>	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0	00000000 <sub>B</sub>
	W	W	W	W	W	W	W	W	

図 5.3-1 データ方向レジスタ(DDR)の構成

## 5.4 外部端子をI/Oポートとして使用

各外部端子についての初期値と、I/Oポートとして使用するのか、制御端子として使用するのかを切り替えるレジスタとの関係を示します。

外部端子の機能(I/Oポートまたは制御端子)選択

表 5.4-1に、各外部端子についての初期値と、I/Oポートとして使用するのか、制御端子として使用するのかを切り替えるレジスタとの関係を示します。

表 5.4-1中のシングルチップ: ~, 外部バス: ~とあるのは、使用する動作モードによって、8ビット: ~, 16ビット: ~とあるのは、外部バス幅によって機能が変化することを示します。

表 5.4-1 外部端子の機能選択一覧(続く)

端子番号	端子記号	初期値(シングルチップモード)	切替えレジスタ
25 ~ 32	P20 ~ P27	P20 ~ P27	MD0 ~ 2, AMD0 ~ 5, M0, M1で設定したモードで自動的に切り替えます。 シングルチップ: P20 ~ P27 8ビット: P20 ~ P27 16ビット: D16 ~ D23
	D16 ~ D23		
33 ~ 39, 41	P30 ~ P37	P30 ~ P37	MD0 ~ 2, M0, M1で設定したモードで自動的に切り替えます。 シングルチップ: P30 ~ P37 外部バス: D24 ~ D31
	D24 ~ D31		
42 44 ~ 58	P40 ~ P47 P50 ~ P57 A00 ~ A15	P40 ~ P47 P50 ~ P57	MD0 ~ 2, M0, M1で設定したモードで自動的に切り替えます。 シングルチップ: P40 ~ P47, P50 ~ P57 外部バス: A00 ~ A15
59 ~ 64 66, 67	P60 ~ P67	P60 ~ P67	EPCR1 (AE16 ~ AE23ビット) 0: P60 ~ P67 1: A16 ~ A23
	A16 ~ A23		
68	P70	P70	EPCR1 (AE24ビット) 0: P70 1: A24 端子の値が常にFRCK, TC12に入力されます(ストップ時除く)。
	A24		
	FRCK		
	TC12		
19	P80	P80	EPCR0 (RDYEビット) 0: P80 1: RDY
	RDY		
20	P81	P81	EPCR0 (BREビット) 0: P81 1: BGRNT 端子の値が常にIN0に入力されます(ストップ時除く)。
	BGRNT		
	IN0		
21	P82	P82	EPCR0 (BREビット) 0: P82 1: BRQ 端子の値が常にIN1に入力されます(ストップ時除く)。
	BRQ		
	IN1		
22	P83	P83	EPCR0 (RDXEビット) 0: P83 1: RD
	RD		

表 5.4-1 外部端子の機能選択一覧(続く)

端子番号	端子記号	初期値 (シングルチップモード)	切替えレジスタ
23, 24	P84, P85 WR0, WR1	P84, P85	MD0 ~ 2, AMD0 ~ 5, M0, M1 で設定したモードで自動的に切り替ります。 シングルチップ: P84, P85 8ビット: WR0, P85 16ビット: WR0, WR1
11 ~ 9	PA0 ~ PA2 CS0 ~ CS2	PA0 ~ PA2	EPCRO(COE0 ~ COE2ビット) 0: PA0 ~ PA2 1: CS0 ~ CS2
8	PA3 CS3 S01	PA3	EPCRO(COE3ビット)とSMR(SOEビット) COE3, SOE 00: PA3 10: CS3 その他: S01
7, 6	PA4, PA5 CS4, CS5 SI1, SC1	PA4, PA5	EPCRO(COE4, COE5ビット) 0: PA4, PA5 1: CS4, CS5 端子の値が常にSI1, SC1に入力されます(ストップ時除く)。
5	PA6 CLK	PA6	EPCRO(COE6ビット) 0: PA6 1: CLK
96 ~ 99	PG0 ~ PG3 OCPA0 ~ OCPA3	PG0 ~ PG3	PCNL(POENビット) 0: PG0 ~ PG3 1: OCPA0 ~ OCPA3
100, 1 ~ 3	PG4 ~ PG7 OC0 ~ OC3	PG4 ~ PG7	OCS0 ~ 3(OTEビット) 0: PG4 ~ PG7 1: OC0 ~ OC3
16 ~ 18	MD0 ~ MD2	MD0 ~ MD2	
12	P86 ALE	P86	EPCRO(ALEEビット) 0: P86 1: ALE
72	PJ0 ~ PJ7 AN0 ~ AN7	PJ0 ~ PJ7	AIC(AIビット), DDRJ(PJビット) AI, PJ 00: AN その他: PJ
95 ~ 94	PE0, PE1 INT0, INT1	PE0/INT0, PE1/INT1	端子の値が常にINT0, INT1に入力されます。
89, 88	PE2, PE3 INT2, INT3	PE2/INT2, PE3/INT3	端子の値が常にINT2, INT3に入力されます。
87	PE4 INT4 TCI1	PE4/INT4/TCI1	端子の値が常にINT4, TCI1に入力されます(ストップ時除く)。
86	PE5 INT5 SC0	PE5/INT5/SC0	端子の値が常にINT5, SC0に入力されます(ストップ時除く)。
85	PE6 SI0	PE6/SI0	端子の値が常にSI0に入力されます(ストップ時除く)。
84	PE7 S00	PE7	SMR(SOEビット) 0: PE7 1: S00

表 5.4-1 外部端子の機能選択一覧(続き)

端子番号	端子記号	初期値 (シングルチップモード)	切替えレジスタ
80	PF0	PF0/TC10	端子の値が常にTC10に入力されます (ストップ時除く)。
	TC10		
81	PF1	PF1/SI2	端子の値が常にSI2に入力されます (ストップ時除く)。
	SI2		
82	PF2	PF2	SMR(SOEビット) 0:PF2 1:S02
	S02		
83	PF3	PF3/SC2/ $\overline{\text{ATG}}$	端子の値が常にSC2, $\overline{\text{ATG}}$ に入力されます(ストップ時除く)。
	SC2		
	$\overline{\text{ATG}}$		
69	$\text{AV}_{\text{CC}}$	$\text{AV}_{\text{CC}}$	-
70	AVRH	AVRH	-
71	$\text{AV}_{\text{SS}}$ (AVRL)	$\text{AV}_{\text{SS}}$ (AVRL)	-
14	$\overline{\text{RST}}$	$\overline{\text{RST}}$	-
13	$\overline{\text{HST}}$	$\overline{\text{HST}}$	-
92	X0	X0	-
91	X1	X1	-
4, 93, 43	$\text{V}_{\text{CC}}$	$\text{V}_{\text{CC}}$	-
15, 40, 65, 90	$\text{V}_{\text{SS}}$	$\text{V}_{\text{SS}}$	-



## 第6章 16ビットリロードタイマ

---

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

---

- 6.1 16ビットリロードタイマの概要
- 6.2 16ビットリロードタイマのレジスタ
- 6.3 16ビットリロードタイマの動作
- 6.4 カウンタの動作状態

## 6.1 16ビットリロードタイマの概要

16ビットリロードタイマは、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、内部カウンタクロック作成用プリスケアラ、コントロールレジスタで構成されています。

## 16ビットリロードタイマの概要

16ビットリロードタイマは、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、内部カウンタクロック作成用プリスケアラ、コントロール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(マシニングクロックの2/8/32分周)から選択  
できます。

割込みによるDMA転送の起動が可能です。MB91F127/128はこのタイマを3チャンネル内蔵しています。

リロードタイマのチャンネル2 TO出力は、LSI内部でA/Dコンバータに接続されています。したがって、リロードレジスタに設定された周期でA/D変換を起動することが可能です。

## ブロックダイアグラム

図 6.1-1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。

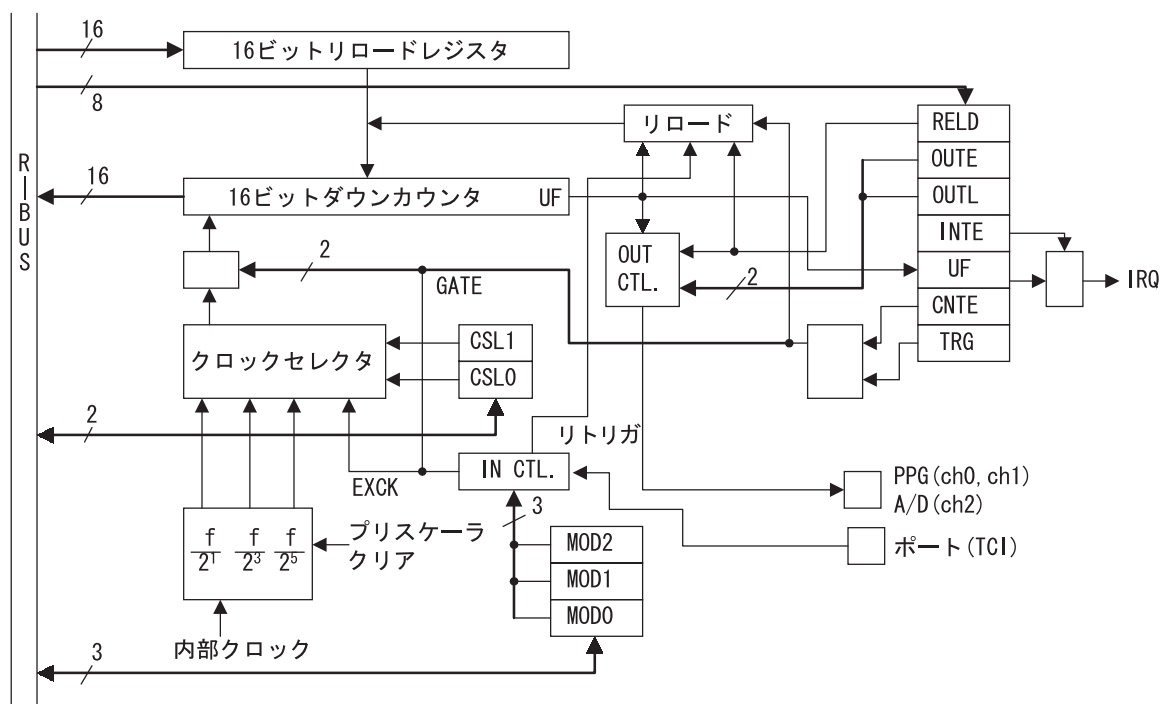


図 6.1-1 16ビットリロードタイマのブロックダイヤグラム

## 6.2 16ビットリロードタイマのレジスタ

この節では、16ビットリロードタイマで使用するレジスタの構成と機能について説明します。

16ビットリロードタイマのレジスタ一覧

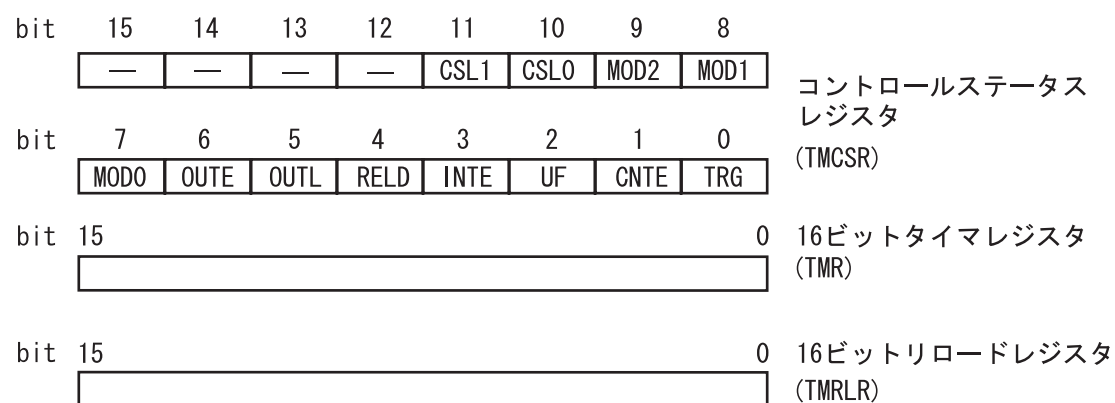


図 6.2-1 16ビットリロードタイマのレジスタ一覧



### 6.2.1 コントロールステータスレジスタ(TMCSR)

コントロールステータスレジスタ(TMCSR)は,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。

コントロールステータスレジスタ(TMCSR)のビット構成

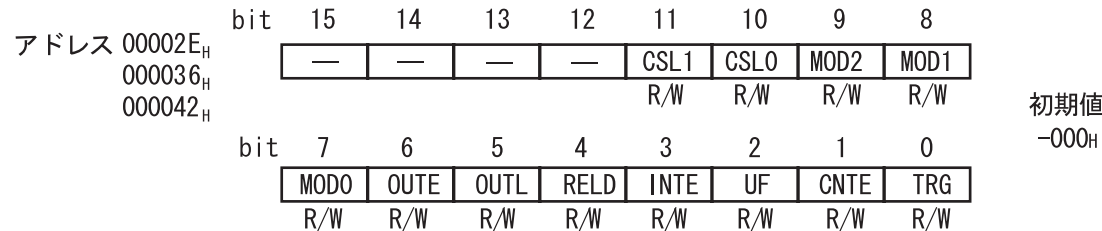


図 6.2-2 コントロールステータスレジスタ(TMCSR)のビット構成

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。

コントロールステータスレジスタ(TMCSR)は, 同時書込みが可能です。

コントロールステータスレジスタ(TMCSR)のビット機能

以下に, コントロールステータスレジスタ(TMCSR)のビット機能を説明します。

【ビット11, 10】 CSL1, CSL0 (Count clock SLect)

カウントクロックセレクトビットです。

表 6.2-1に, 選択されるクロックソースを示します。

表 6.2-1 CSLビット設定クロックソース

CSL1	CSL0	クロックソース(f: マシンクロック)
0	0	$f/2^1$
0	1	$f/2^3$
1	0	$f/2^5$
1	1	外部クロック

## 【ビット9,8,7】 MOD2,MOD1,MOD0 (MODE)

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、入力端子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、ゲートカウントモードになり、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機能の設定を行います。

表 6.2-2,表 6.2-3に,MOD2,1,0の設定を示します。

表 6.2-2 MOD2,1,0ビットの設定方法(1)

内部クロックモード時 (CSL0,1=00,01,10)

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	x	0	ゲート入力	“L”レベル
1	x	1		“H”レベル

表 6.2-3 MOD2,1,0ビットの設定方法(2)

イベントカウントモード時 (CSL0,1=11)

MOD2	MOD1	MOD0	入力端子機能	有効エッジ,レベル
x	0	0		
	0	1	イベント入力	立上りエッジ
	1	0		立下りエッジ
	1	1		両エッジ

## 【ビット6】 OUTE (OUTput Enable)

必ず“0”を設定してください。

## 【ビット5】 OUTL

必ず“0”を設定してください。

## 【ビット4】 RELD

リロード許可ビットです。

“1”のときリロードモードになり、カウンタの値が0000<sub>H</sub>からFFFF<sub>H</sub>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”のときカウンタの値が0000<sub>H</sub>からFFFF<sub>H</sub>へのアンダフローによりカウント動作を停止します。

## 【ビット3】 INTE

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”のときは割込み要求を発生しません。

【ビット2】 UF

タイマ割込み要求フラグです。カウンタの値が0000<sub>H</sub>からFFFF<sub>H</sub>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"が読み出されず。

【ビット1】 CNTE

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。

【ビット0】 TRG

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は,CNTE="1"のときのみ有効となります。CNTE="0"のときには何も起こりません。

## 6.2.2 16ビットタイマレジスタ(TMR)

16ビットタイマレジスタ(TMR)は,16ビット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。初期値は不定です。

このレジスタの読出しは,必ず16ビットデータ転送命令で行ってください。

16ビットタイマレジスタ(TMR)のビット構成

図 6.2-3に,16ビットタイマレジスタ(TMR)のビット構成を示します。



図 6.2-3 16ビットタイマレジスタ(TMR)のビット構成

### 6.2.3 16ビットリロードレジスタ(TMRLR)

16ビットリロードレジスタ(TMRLR)は, カウント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16ビットデータ転送命令で行ってください。

16ビットリロードレジスタ(TMRLR)のビット構成

図 6.2-4に, 16ビットリロードレジスタ(TMRLR)のビット構成を示します。

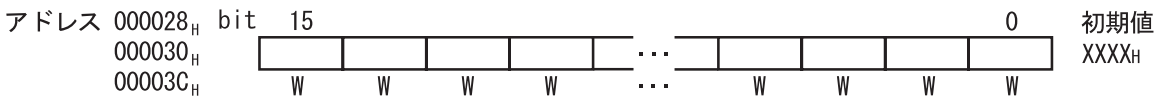


図 6.2-4 16ビットリロードレジスタ(TMRLR)のビット構成

## 6.3 16ビットリロードタイマの動作

この節では、16ビットリロードタイマの動作について説明します。

### 内部クロック動作

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、カウントソースとしてマシニングクロックの2, 8, 32分周のクロックから選択することができます。

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、コントロールステータスレジスタ(TMCSR)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")動作モードに係わらず常に有効です。

図 6.3-1に、カウンタの起動および動作を示します。

カウンタスタートのトリガが入力されてから、16ビットリロードレジスタ(TMRLR)のデータがカウンタへロードされるまでに、T(T:周辺系クロックマシニングサイクル)の時間がかかります。

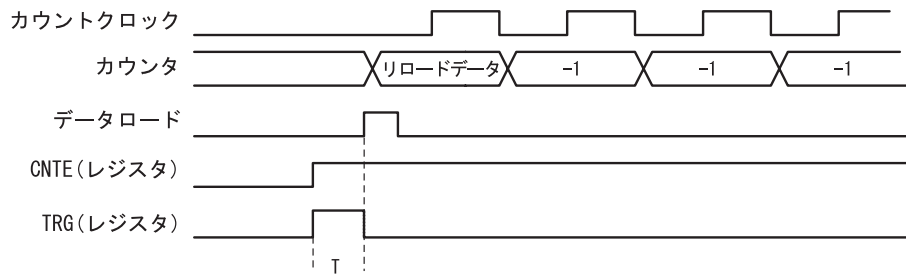


図 6.3-1 カウンタの起動および動作

### アンダフロー動作

カウンタの値が0000<sub>H</sub>からFFFF<sub>H</sub>になるときをアンダフローとしています。したがって、〔リロードレジスタの設定値+1〕カウントでアンダフローが発生することになります。

アンダフロー発生時にコントロールステータスレジスタ(TMCSR)のRELDビットが"1"のとき16ビットリロードレジスタ(TMRLR)の内容をカウンタへロードして、カウント動作を継続します。RELDビットが"0"のとき、カウンタはFFFF<sub>H</sub>で停止します。

アンダフローによりコントロールステータスレジスタ(TMCSR)のUFビットがセットされ、INTEビットが"1"のとき、割込要求が発生します。

図 6.3-2に、アンダフロー動作を示します。

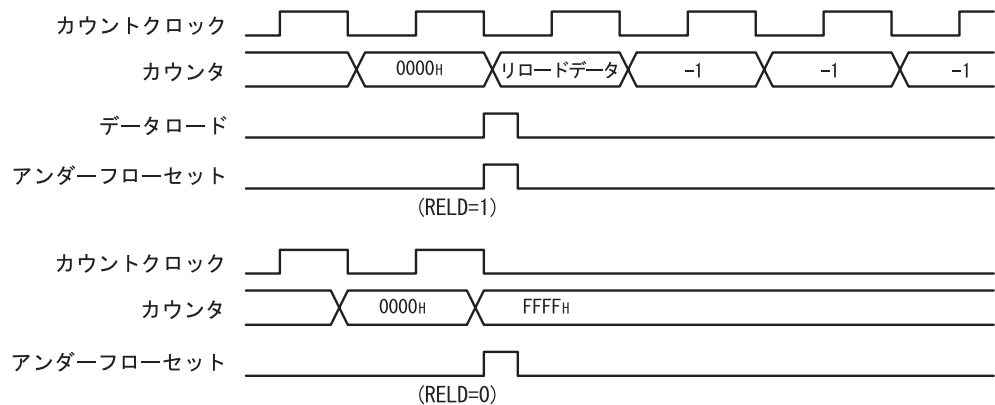


図 6.3-2 アンダフロー動作のタイミング

## 入力端子機能

クロックソースとして、内部クロックを選択した場合、TCI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16ビットリロードレジスタ(TMLR)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。

TCIは、 $2 \times T$  ( $T$ は周辺系クロックマシンサイクル)以上のパルスを入力してください。

図 6.3-3に、トリガ入力の動作を示します。

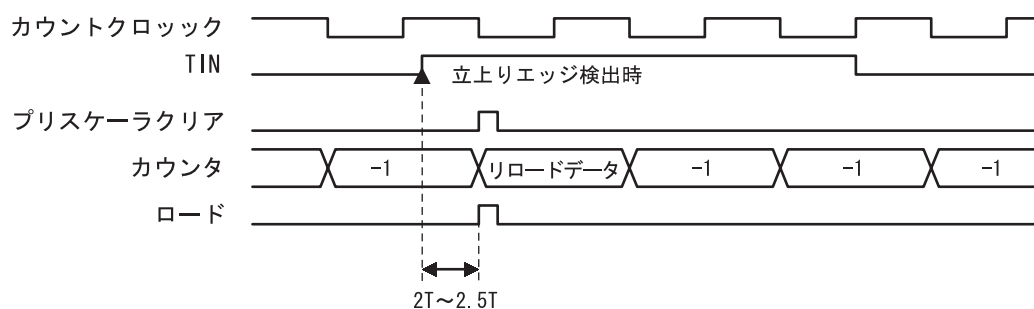


図 6.3-3 トリガ入力の動作

ゲート入力として使用する場合、コントロールステータスレジスタ(TMCSR)のMOD0ビットにより設定される有効レベルが、TCI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、ゲートレベルに関わらず可能です。TCI端子のパルス幅は、 $2 \times T$  ( $T$ は周辺系クロックマシンサイクル)以上にしてください。

図 6.3-4に、ゲート入力の動作を示します。



図 6.3-4 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、TCI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。TCI端子のパルス幅は、 $2 \times T$  ( $T$ は周辺系クロックマシンサイクル)以上にしてください。

## 6.4 カウンタの動作状態

カウンタの状態は、コントロールステータスレジスタ(TMCSR)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 6.4-1に、各状態の遷移を示します。

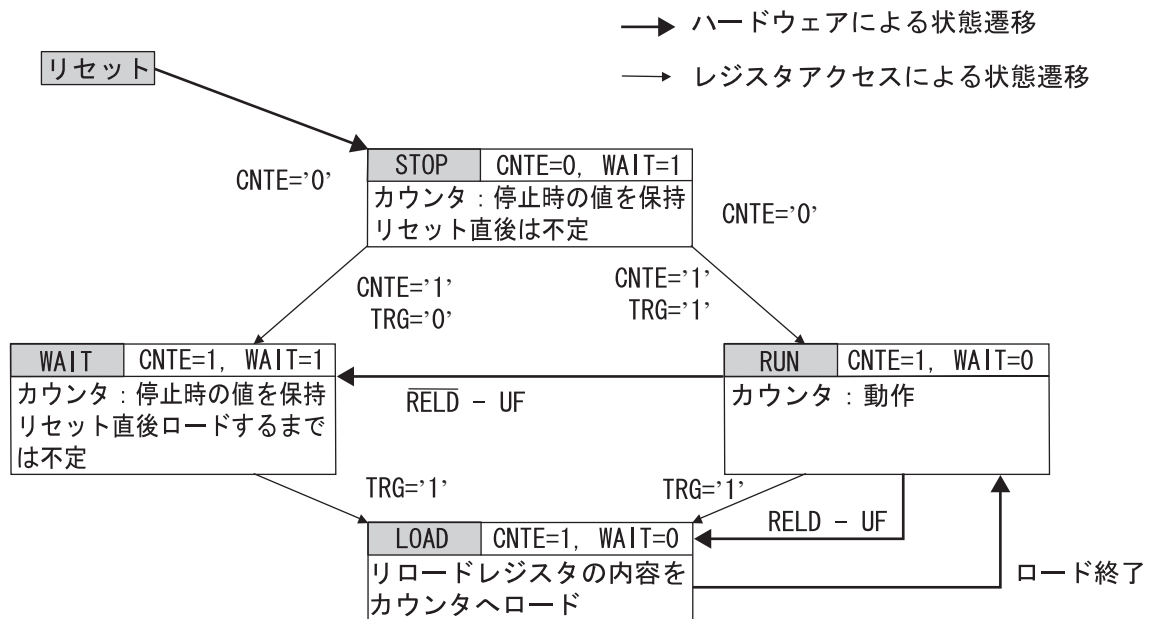


図 6.4-1 カウンタの状態遷移





## 第7章 PPGタイマ

---

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

---

- 7.1 PPGタイマの概要
- 7.2 PPGタイマのブロックダイアグラム
- 7.3 PPGタイマのレジスタ
- 7.4 PPG動作
- 7.5 ワンショット動作
- 7.6 割込み
- 7.7 PPG出力オール”L”とオール”H”
- 7.8 PPGタイマ複数チャネルの起動

## 7.1 PPGタイマの概要

---

PPGタイマは、精度の高いPPG波形を効率良く出力することができます。  
MB91F127/128は、PPGタイマを4チャンネル内蔵しています。

---

### PPGタイマの特長

各チャンネルは、16ビットダウンカウンタ、周期設定用バッファ付16ビットデータレジスタ、デューティ設定用バッファ付16ビットコンペアレジスタ、端子制御部から構成されます。

16ビットダウンカウンタのカウントクロックは、4種類から選択が可能です。

- 内部クロック   ,   /4,   /16,   /64

カウンタ値は、リセット、カウンタボローで"FFFF<sub>H</sub>"に初期化することができます。

各チャンネルごとにPPG出力があります。

### レジスタ概要

- 周期設定レジスタ: バッファ付、リロード用データレジスタ
- デューティ設定レジスタ: バッファ付、コンペアレジスタ
- バッファからの転送は、カウンタボローで行います。

### 端子制御

- デューティ一致で"1"にセット(優先)
- カウンタボローで"0"にリセット
- 出力値固定モードがあり、オール"L"(または"H")を簡単に出力できます。
- 極性指定も可能です。

割込み要求は、以下の組合せから選択して発生することができます。

- PPGタイマ起動
- カウンタボロー発生(周期一致)
- デューティ一致発生
- カウンタボロー発生(周期一致)またはデューティ一致発生

上記の割込み要求によって、DMA転送起動が可能です。

ソフトウェアまたはほかのインターバルタイマで複数チャンネルの同時起動が設定できます。

また、動作中の再起動も設定可能です。

## 7.2 PPGタイマのブロックダイアグラム

図 7.2-1にPPGタイマの全体ブロックダイアグラムを, 図 7.2-2にPPGタイマ1チャンネル分のブロックダイアグラムを示します。

PPGタイマの全体ブロックダイアグラム

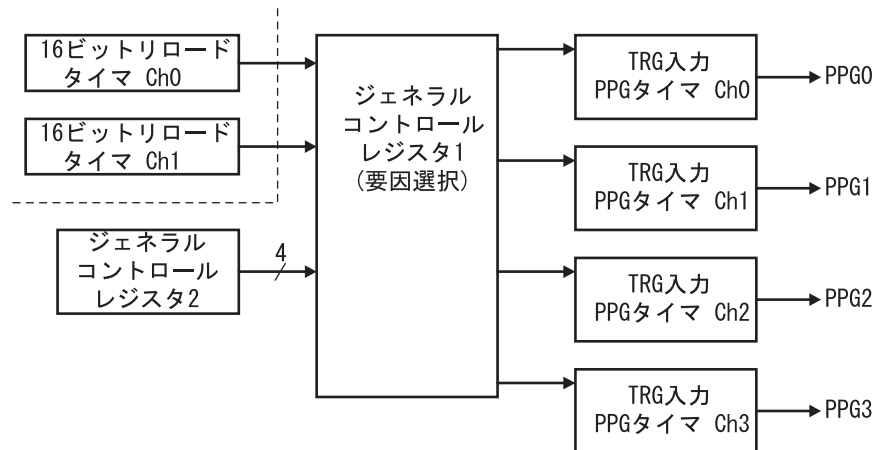


図 7.2-1 PPGタイマの全体ブロックダイアグラム

PPGタイマ1チャンネル分のブロックダイアグラム

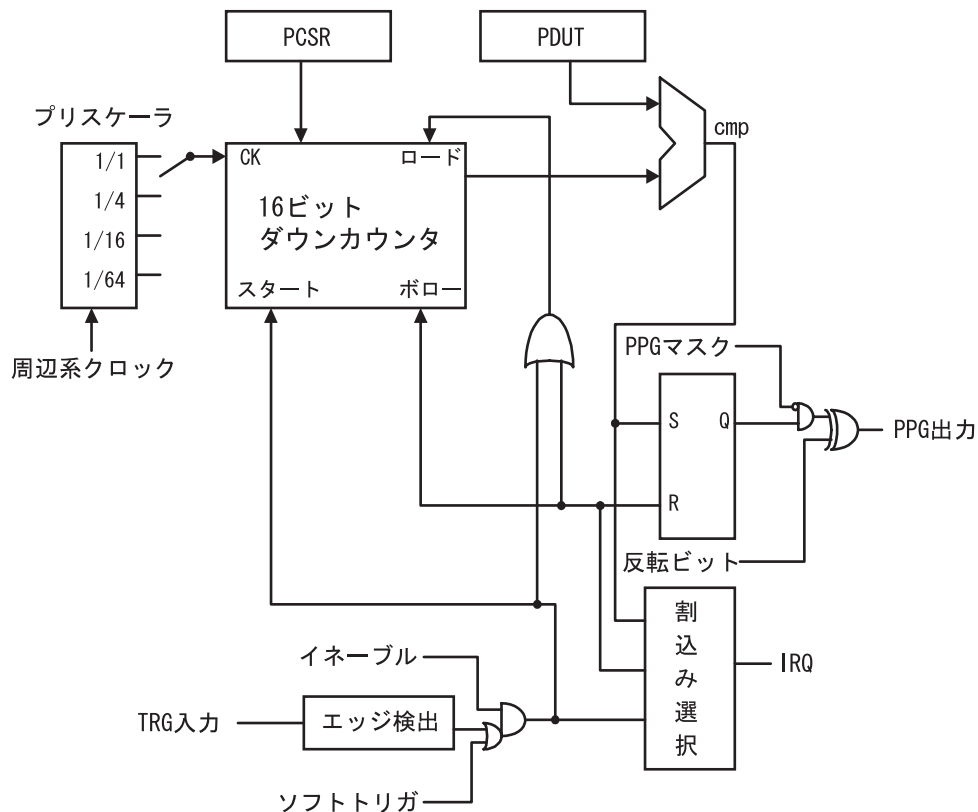


図 7.2-2 PPGタイマ1チャンネル分のブロックダイアグラム

### 7.3 PPGタイマのレジスタ

図 7.3-1にPPGタイマのレジスタ一覧を示します。

PPGタイマのレジスタ一覧

bit 15	0	
<div>GCN1</div>		ジェネラルコントロールレジスタ1
	<div>GCN2</div>	ジェネラルコントロールレジスタ2
<div>PTMR0</div>		ch0 タイマレジスタ
<div>PCSR0</div>		ch0 周期設定レジスタ
<div>PDUT0</div>		ch0 デューティ設定レジスタ
<div>PCNH0</div>	<div>PCNL0</div>	ch0 コントロールステータスレジスタ
<div>PTMR1</div>		ch1 タイマレジスタ
<div>PCSR1</div>		ch1 周期設定レジスタ
<div>PDUT1</div>		ch1 デューティ設定レジスタ
<div>PCNH1</div>	<div>PCNL1</div>	ch1 コントロールステータスレジスタ
<div>PTMR2</div>		ch2 タイマレジスタ
<div>PCSR2</div>		ch2 周期設定レジスタ
<div>PDUT2</div>		ch2 デューティ設定レジスタ
<div>PCNH2</div>	<div>PCNL2</div>	ch2 コントロールステータスレジスタ
<div>PTMR3</div>		ch3 タイマレジスタ
<div>PCSR3</div>		ch3 周期設定レジスタ
<div>PDUT3</div>		ch3 デューティ設定レジスタ
<div>PCNH3</div>	<div>PCNL3</div>	ch3 コントロールステータスレジスタ

図 7.3-1 PPGタイマのレジスタ一覧

### 7.3.1 コントロールステータスレジスタ(PCNH, PCNL)

コントロールステータスレジスタ(PCNH, PCNL)は, PPGタイマの制御およびステータス表示をします。PPGタイマ動作中に書き換え不可能なビットがありますので注意してください。

コントロールステータスレジスタ(PCNH, PCNL)のビット構成

図 7.3-2に, コントロールステータスレジスタ(PCNH, PCNL)のビット構成を示します。

PCNH アドレス	ch0	0000E6 <sub>H</sub>	bit 15	CNTE	14	STGR	13	MDSE	12	RTRG	11	CKS1	10	CKS0	9	PGMS	8	初期値 0000000-00000000 <sub>B</sub>
	ch1	0000EE <sub>H</sub>		R/W		R/W		R/W		R/W		R/W		R/W		R/W	—	
	ch2	0000F6 <sub>H</sub>		○		○		×		×		×		×		○	—	←動作中の書換え
	ch3	0000FE <sub>H</sub>																
PCNL アドレス	ch0	0000E7 <sub>H</sub>	bit 7	EGS1	6	EGS0	5	IREN	4	IRQF	3	IRS1	2	IRS0	1	POEN	0	
	ch1	0000EF <sub>H</sub>		R/W		R/W		R/W		R/W		R/W		R/W		R/W	R/W	
	ch2	0000F7 <sub>H</sub>		×		×		○		○		×		×		×	×	←動作中の書換え
	ch3	0000FF <sub>H</sub>																

図 7.3-2 コントロールステータスレジスタ(PCNH, PCNL)のビット構成

コントロールステータスレジスタ(PCNH, PCNL)のビット機能

以下に, コントロールステータスレジスタ(PCNH, PCNL)のビット機能を説明します。

【ビット15】CNTE: タイマ許可ビット

このビットは, 16ビットダウンカウンタの動作を許可するビットです。

CNTE	機能
0	停止 (初期値)
1	許可

【ビット14】STGR: ソフトウェアトリガビット

このビットに"1"を書き込むことによりソフトウェアトリガがかかります。  
STGRビットの読出し値は, 常に"0"です。

【ビット13】MDSE: モード選択ビット

このビットは, 連続してパルスを出すPPG動作か, 単一パルスを出すワンショット動作かを選択します。

MDSE	機能
0	PPG動作 (初期値)
1	ワンショット動作

【ビット12】RTRG: 再起動許可ビット

このビットは, ソフトウェアトリガによる再起動を許可するビットです。

RTRG	機能
0	再起動禁止 (初期値)
1	再起動許可

## 【ビット11,10】CKS1,CKS0:カウンタクロック選択ビット

これらのビットは、16ビットダウンカウンタのカウントクロックを選択します。

CKS1	CKS0	周期
0	0	(初期値)
0	1	/4
1	0	/16
1	1	/64

:周辺系マシクロック

## 【ビット9】PGMS:PPG出力マスク選択ビット

このビットに"1"を書き込むことによりモード設定、周期設定値、デューティ設定値に関わらずPPG出力を"0"または"1"にマスクできます。

極性	PPG出力
通常極性	L出力
反転極性	H出力

通常極性時にオール"H"または反転極性時にオール"L"を出力したい場合は、周期設定レジスタとデューティ設定レジスタに同値を書き込めば上記マスク値の反転を出力できます。

## 【ビット8】(reserved)

このビットは、未使用ビットです。

## 【ビット7,6】EGS1,EGS0:トリガ入力エッジ選択ビット

これらのビットは、ジェネラルコントロールレジスタ1で選んだ起動要因の有効エッジを選択します。

どのモードを選択していてもソフトトリガのビットに"1"を書き込むとソフトトリガは有効になります。

EGS1	EGS0	エッジ選択
0	0	無効 (初期値)
0	1	立上りエッジ
1	0	立下りエッジ
1	1	両エッジ

## 【ビット5】IREN:割込み要求許可ビット

このビットは、割込み要求を許可するビットです。

IREN	機能
0	禁止 (初期値)
1	許可

## 【ビット4】IRQF:割込み要求フラグ

このビットは、ビット5のIRENビットが許可されていてビット3,2のIRS1ビット、IRS0ビットにて選択した割込み要因が発生すると、このビットがセットされCPUに割込み要求が発生します。また、DMA転送の起動を選択している場合は、DMA転送が起動されます。

このビットのクリアは、"0"書込みとDMACからのクリア信号で行われます。

"1"を書き込んでもビット値は変化しません。

リードモディファイライト系命令における読出し値は、ビット値に関わらず"1"です。

## 【ビット3,2】IRS1, IRS0: 割込み要因選択ビット

これらのビットは、ビット4のIRQFビットをセットする要因を選択します。

IRS1	IRS0	割込み要因
0	0	ソフトウェアトリガ入力あり (初期値)
0	1	カウンタボロー発生(周期一致)
1	0	デューティ一致発生
1	1	カウンタボロー発生(周期一致)またはデューティ一致発生

## 【ビット1】POEN: PPG出力許可ビット

このビットは、"1"に設定することにより、PPG出力が端子から出力されます。

POEN	機能
0	汎用ポート (初期値)
1	PPG出力端子

## 【ビット0】OSEL: PPG出力極性指定ビット

このビットは、PPG出力の極性を設定します。

ビット9のPGMSビットとの組合せで、表 7.3-1のようになります。

表 7.3-1 PPG出力極性指定組合せ

PGMS	OSEL	PPG出力
0	0	通常極性 (初期値)
0	1	反転極性
1	0	出力"L"固定
1	1	出力"H"固定

表 7.3-2 PPG出力極性指定

極性	リセット後	デューティ一致	カウンタボロー
通常極性	"L"出力	立上りエッジ	立下りエッジ
反転極性	"H"出力	立下りエッジ	立上りエッジ



### 7.3.2 PPG周期設定レジスタ(PCSR)

PPG周期設定レジスタ(PCSR)は,周期を設定するためのバッファ付きレジスタです。バッファからの転送は,カウンタボローで行われます。

PPG周期設定レジスタ(PCSR)のビット構成

図 7.3-3に,PPG周期設定レジスタ(PCSR)のビット構成を示します。

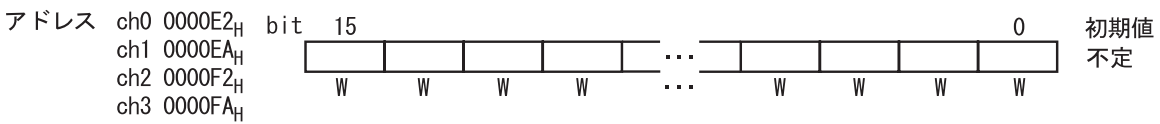


図 7.3-3 PPG周期設定レジスタ(PCSR)のビット構成

周期設定レジスタの初期設定時および書換え時は,周期設定レジスタの書込み後,必ずデューティ設定レジスタへの書込み動作を行ってください。  
このレジスタは,16ビットデータでアクセスしてください。

### 7.3.3 PPGデューティ設定レジスタ(PDUT)

PPGデューティ設定レジスタ(PDUT)は、デューティを設定するためのバッファ付レジスタです。バッファからの転送は、カウンタボローで行われます。

PPGデューティ設定レジスタ(PDUT)のビット構成

図 7.3-4は、PPGデューティ設定レジスタ(PDUT)のビット構成を示します。

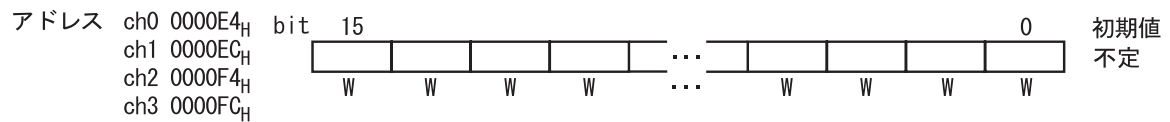


図 7.3-4 PPGデューティ設定レジスタ(PDUT)ビット構成

周期設定レジスタの値とデューティ設定レジスタの値を同じにすると、通常極性時にオール”H”を、反転極性時にオール”L”を出力します。

PCSR < PDUTとなるような値を設定しないでください。PPG出力は不定となります。このレジスタは、16ビットデータでアクセスしてください。

### 7.3.4 PPGタイマレジスタ(PTMR)

PPGタイマレジスタ(PTMR)は,16ビットダウンカウンタの値を読み出すことができます。

PPGタイマレジスタ(PTMR)のビット構成

図 7.3-5に,PPGタイマレジスタ(PTMR)のビット構成を示します。

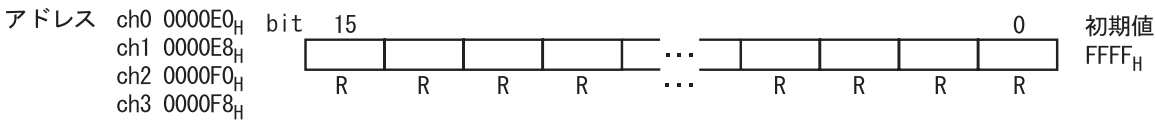


図 7.3-5 PPGタイマレジスタ(PTMR)のビット構成

このレジスタは,16ビットデータでアクセスしてください。

### 7.3.5 ジェネラルコントロールレジスタ1(GCN1)

ジェネラルコントロールレジスタ1(GCN1)は,PPGタイマのトリガ入力の変因を選択するレジスタです。

ジェネラルコントロールレジスタ1(GCN1)のビット構成

図 7.3-6に,ジェネラルコントロールレジスタ1(GCN1)のビット構成を示します。

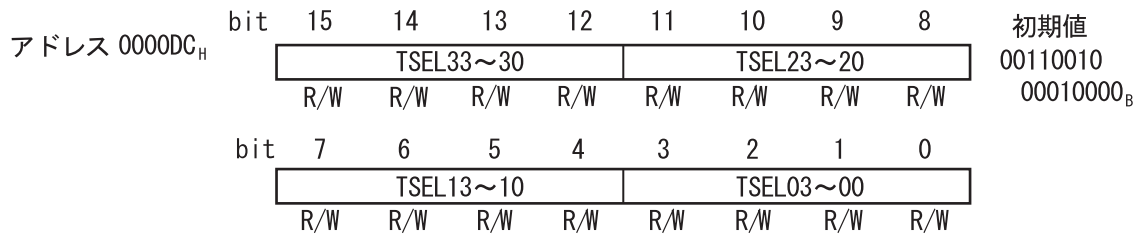


図 7.3-6 ジェネラルコントロールレジスタ1(GCN1)のビット構成

ジェネラルコントロールレジスタ1(GCN1)のビット詳細

【ビット15～12】TSEL33-30:ch3トリガ入力選択ビット

これらのビットは,ch3トリガ入力選択ビットです。

TSEL33-30				ch3トリガ入力
15	14	13	12	
0	0	0	0	GCN2のEN0ビット
0	0	0	1	GCN2のEN1ビット
0	0	1	0	GCN2のEN2ビット
0	0	1	1	GCN2のEN3ビット (初期値)
0	1	0	0	16ビットリロードタイマch0
0	1	0	1	16ビットリロードタイマch1
0	1	1	X	設定禁止
1	0	0	0	設定禁止
1	0	0	1	設定禁止
1	0	1	0	設定禁止
1	0	1	1	設定禁止
1	1	X	X	設定禁止

## 【ビット11～8】TSEL23-20:ch2トリガ入力選択ビット

これらのビットは, ch2トリガ入力選択ビットです。

TSEL23-20				ch2トリガ入力
11	10	9	8	
0	0	0	0	GCN2のEN0ビット
0	0	0	1	GCN2のEN1ビット
0	0	1	0	GCN2のEN2ビット (初期値)
0	0	1	1	GCN2のEN3ビット
0	1	0	0	16ビットリロードタイマch0
0	1	0	1	16ビットリロードタイマch1
0	1	1	X	設定禁止
1	0	0	0	設定禁止
1	0	0	1	設定禁止
1	0	1	0	設定禁止
1	0	1	1	設定禁止
1	1	X	X	設定禁止

## 【ビット7～4】TSEL13-10:ch1トリガ入力選択ビット

これらのビットは, ch1トリガ入力選択ビットです。

TSEL13-10				ch1トリガ入力
7	6	5	4	
0	0	0	0	GCN2のEN0ビット
0	0	0	1	GCN2のEN1ビット (初期値)
0	0	1	0	GCN2のEN2ビット
0	0	1	1	GCN2のEN3ビット
0	1	0	0	16ビットリロードタイマch0
0	1	0	1	16ビットリロードタイマch1
0	1	1	X	設定禁止
1	0	0	0	設定禁止
1	0	0	1	設定禁止
1	0	1	0	設定禁止
1	0	1	1	設定禁止
1	1	X	X	設定禁止

## 【ビット3～0】TSEL03-00:ch0トリガ入力選択ビット

これらのビットは, ch0トリガ入力選択ビットです。

TSEL00-03				ch0トリガ入力
3	2	1	0	
0	0	0	0	GCN2のEN0ビット (初期値)
0	0	0	1	GCN2のEN1ビット
0	0	1	0	GCN2のEN2ビット
0	0	1	1	GCN2のEN3ビット
0	1	0	0	16ビットリロードタイマch0
0	1	0	1	16ビットリロードタイマch1
0	1	1	X	設定禁止
1	0	0	0	設定禁止
1	0	0	1	設定禁止
1	0	1	0	設定禁止
1	0	1	1	設定禁止
1	1	X	X	設定禁止

### 7.3.6 ジェネラルコントロールレジスタ2(GCN2)

ジェネラルコントロールレジスタ2(GCN2)は、ソフトウェアによって、起動トリガを発生させるためのレジスタです。

ジェネラルコントロールレジスタ2(GCN2)のビット構成

図 7.3-7に、ジェネラルコントロールレジスタ2(GCN2)のビット構成を示します。

	bit	7	6	5	4	3	2	1	0	初期値
アドレス 0000DF <sub>H</sub>		—	—	—	—	EN3	EN2	EN1	EN0	00000000 <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 7.3-7 ジェネラルコントロールレジスタ2(GCN2)のビット構成

ジェネラルコントロールレジスタ1(GCN1)でこのレジスタのENビットを選択した場合、レジスタの値がそのままPPGタイマのトリガ入力に伝わります。

コントロールステータスレジスタ(PCN)のEGS1,0ビットで選択したエッジをソフトウェアで発生させることにより、複数チャネルのPPGタイマを同時に起動できます。

このレジスタのビット7～ビット4には、必ず0を書き込んでください。

## 7.4 PPG動作

PPG動作は、連続してパルスを出力する動作です。

### PPG動作

PPG動作では、起動トリガの検出時より連続してパルスを出力することができます。出力パルスの周期は、PCSR値を変えることにより制御することができ、またデューティ比は、PDUT値を変えることにより制御できます。

PCSRにデータを書き込んだ後は、必ずPDUTへの書込みを行ってください。

図 7.4-1にトリガの再起動を禁止した場合のPPG動作のタイミングチャートを、図 7.4-2にトリガの再起動を許可した場合のPPG動作のタイミングチャートを示します。

#### 再起動禁止の場合

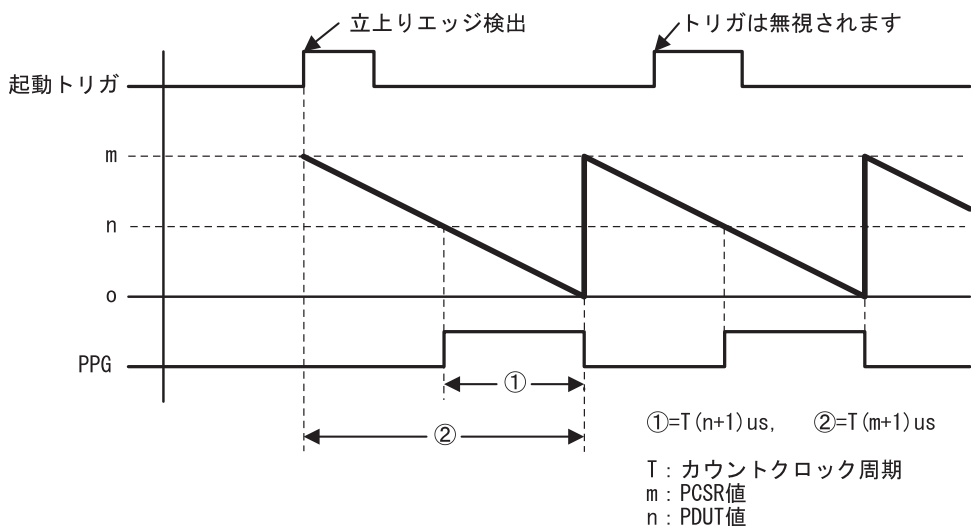


図 7.4-1 PPG動作のタイミングチャート(トリガ再起動禁止)

#### 再起動許可の場合

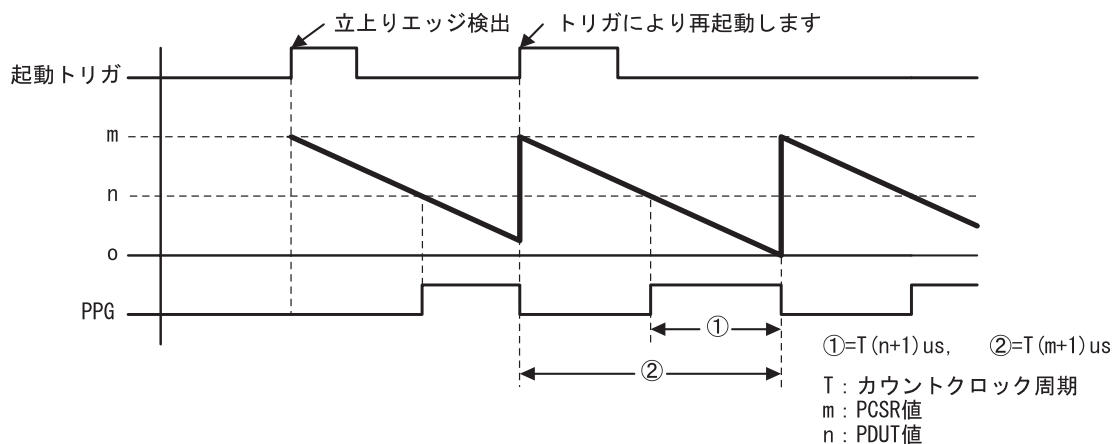


図 7.4-2 PPG動作のタイミングチャート(トリガ再起動許可)

## 7.5 ワンショット動作

ワンショット動作では,単一パルスを出力する動作です。

### ワンショット動作

ワンショット動作では,トリガにより任意の幅の単一パルスを出力することができます。再起動許可の場合は,動作中にエッジを検出するとカウンタをリロードします。

図 7.5-1にトリガの再起動を禁止した場合のワンショット動作のタイミングチャートを,図 7.5-2にトリガの再起動を許可した場合のワンショット動作のタイミングチャートを示します。

#### 再起動禁止の場合

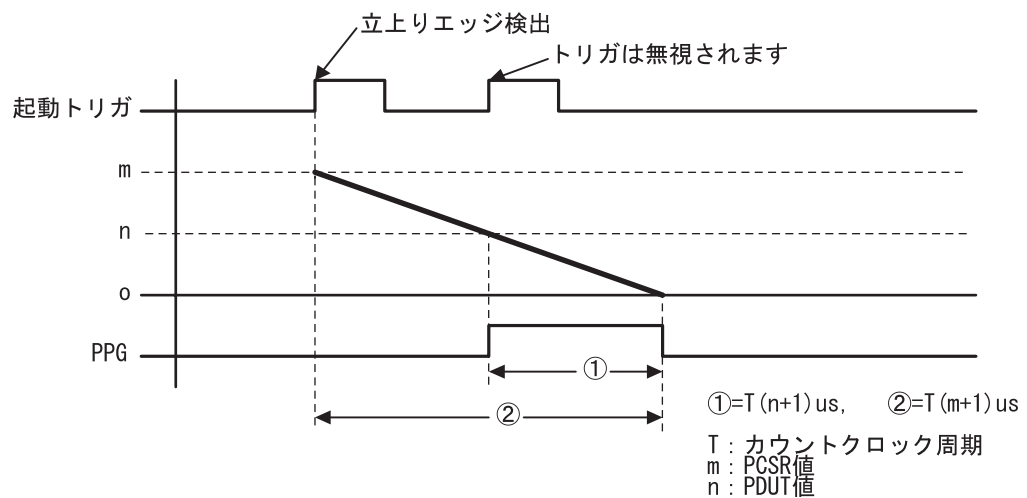


図 7.5-1 ワンショット動作のタイミングチャート(トリガ再起動禁止)

#### 再起動許可の場合

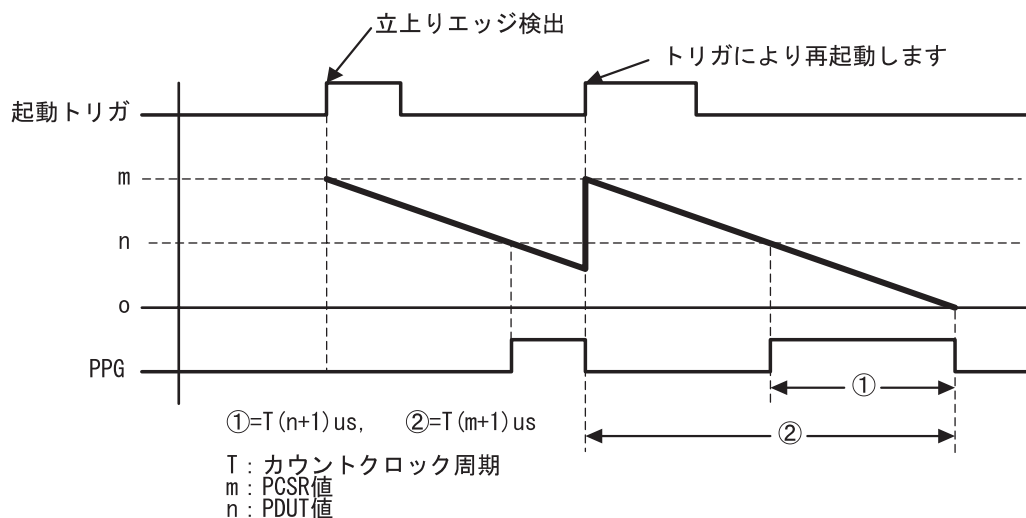


図 7.5-2 ワンショット動作のタイミングチャート(トリガ再起動許可)



# 7.6 割り込み

割り込み要因とタイミングチャートを示します。

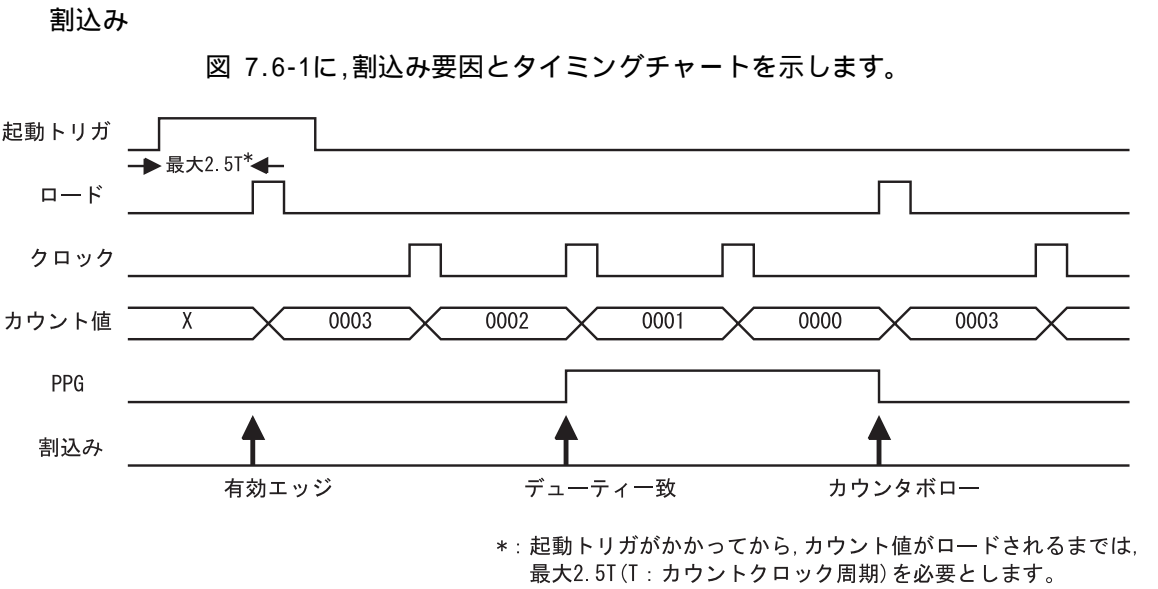


図 7.6-1 割り込み要因とタイミングチャート(PPG出力:通常極性)

## 7.7 PPG出力オール”L”とオール”H”

PPG出力オール”L”とオール”H”について示します。

PPG出力オール”L”とオール”H”

図 7.7-1に, PPG出力をオール”L”に, 図 7.7-2に, オール”H”にする出力方法を示します。

PPG出力をオール”L”レベルにする例

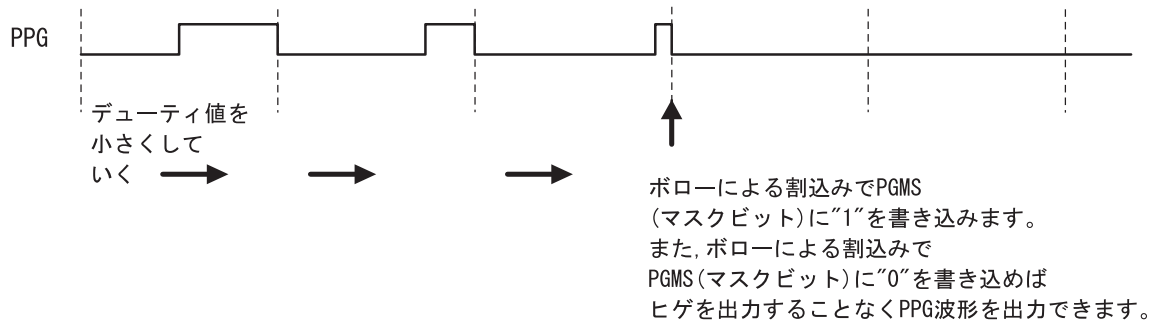


図 7.7-1 PPG出力をオール”L”レベルにする例

PPG出力をオール”H”レベルにする例

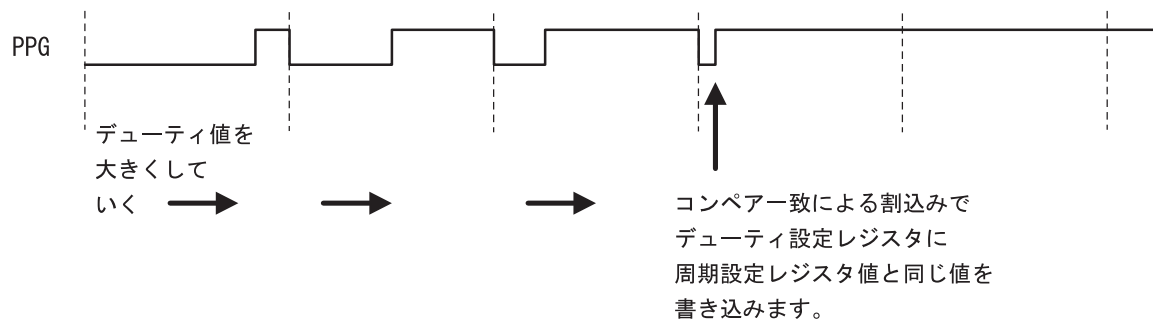


図 7.7-2 PPG出力をオール”H”レベルにする例

## 7.8 PPGタイマ複数チャネルの起動

ジェネラルコントロールレジスタ1,2(GCN1,GCN2)を使って,PPGタイマの複数チャネルを起動できます。ジェネラルコントロールレジスタ1(GCN1)で起動トリガを選択することにより,複数チャネルを同時に起動することができます。ここでは,ジェネラルコントロールレジスタ2(GCN2)を使ったソフトウェア起動例と,16ビットリロードタイマを使用して起動する場合の例を示します。

ソフトウェアによるPPGタイマ複数チャネル起動

### 【設定手順】

- 1) PCSRに周期を設定します。
- 2) PDUTにデューティを設定します。  
必ずPCSR PDUTの順で書込みを行ってください。
- 3) GCN1で,起動するチャネルのトリガ入力要因を決めます。  
ここでは,GCN2を使うので,初期設定のままとします。  
(ch0 EN0, ch1 EN1, ch2 EN2, ch3 EN3)
- 4) 起動するチャネルのコントロールステータスレジスタを設定します。
  - CNTE:1 タイマ動作を許可
  - STGR:0 GCN2で起動するのでここでは起動しない
  - MDSE:0 PPG動作
  - RTRG:0 再起動禁止とする
  - CSK1,0:00 カウントクロック=
  - PGMS:0 出力マスクしない
  - (ビット8:0 未使用ビット,何を設定しても構いません)
  - EGS1,0:01 立上りエッジ起動
  - IREN:1 割込み要求許可
  - IRQF:0 割込み要因をクリア
  - IRS1,0:01 カウンタボロー発生で割込み要求発生
  - POEN:1 PPG出力許可
  - OSEL:0 通常極性
- 5) GCN2にデータを書き込むことで,起動トリガを発生させます。

上記の設定でch0とch1を同時に起動させる場合,GCN2のEN0,EN1に"1"を書き込みます。立上りエッジが発生しPPG0,PPG1からパルスが出力されます。

### 16ビットリロードタイマを使用して起動する場合

16ビットリロードタイマを使用して起動する場合は,設定手順 3)のGCN1で要因として16ビットリロードタイマを選択し,設定手順 5)でGCN2の代わりに16ビットリロードタイマを起動します。また,コントロールステータスレジスタの設定で,

- RTRG:1 再起動許可とする
- EGS1,0:11 両エッジ起動

とし,16ビットリロードタイマ出力をトグル出力設定にすることにより,一定時間ごとにPPGタイマを再起動することも可能です。

## 第8章 多機能タイマ

---

この章では,多機能タイマの概要,レジスタの構成/機能および動作について説明します。

---

- 8.1 多機能タイマの概要
- 8.2 多機能タイマユニットのブロックダイアグラム
- 8.3 多機能タイマユニットのレジスタ
- 8.4 多機能タイマユニットの動作

## 8.1 多機能タイマの概要

多機能タイマユニットは、16ビットフリーランタイマ1本、16ビットアウトプットコンペア4本、16ビットインプットキャプチャ4本から構成されています。

### 多機能タイマの構成

#### 16ビットフリーランタイマ(×1)

16ビットフリーランタイマは16ビットのアップカウンタ、コントロールレジスタ、16ビットコンペアクリアレジスタ、プリスケラより構成されています。このカウンタの出力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(ベースタイマ)として使用されます。

- カウンタ動作クロックは6種類から選択可能です。  
内部クロック6種類( /2, /4, /8, /16, /32, /64)  
:マシクロック
- 割込みはカウンタ値のオーバフロー、コンペアクリアレジスタとのコンペアマッチにより発生することができます。  
(コンペアマッチはモード設定が必要です。)
- カウンタ値は、リセット、ソフトクリア、コンペアクリアレジスタとのコンペアマッチにより"0000<sub>H</sub>"に初期化することができます。

#### アウトプットコンペア(×4)

アウトプットコンペアは4本の16ビットコンペアレジスタ、コンペア出力用ラッチ、コントロール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マ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。

- 4本のコンペアレジスタを独立して動作が可能です。各コンペアレジスタに対応した出力端子と割込みフラグを持っています。
- 2本のコンペアレジスタをペアにして出力端子を制御することが可能です。コンペアレジスタ2本を使用して、出力レベルを反転することが可能です。
- 各出力端子の初期値を設定することが可能です。
- 割込みはコンペア一致により発生可能です。

#### インプットキャプチャ(×4)

インプットキャプチャは独立した4本の外部入力端子と対応したキャプチャレジスタ、コントロール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マ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。

- 外部入力信号の有効エッジ(立上りエッジ、立下りエッジ、両エッジ)を選択可能です。
- 4本のインプットキャプチャは独立して動作が可能です。
- 割込みは外部入力信号の有効エッジにより発生が可能です。

## 8.2 多機能タイマユニットのブロックダイアグラム

図 8.2-1に多機能タイマユニットのブロックダイアグラムを示します。

多機能タイマユニットのブロックダイアグラム

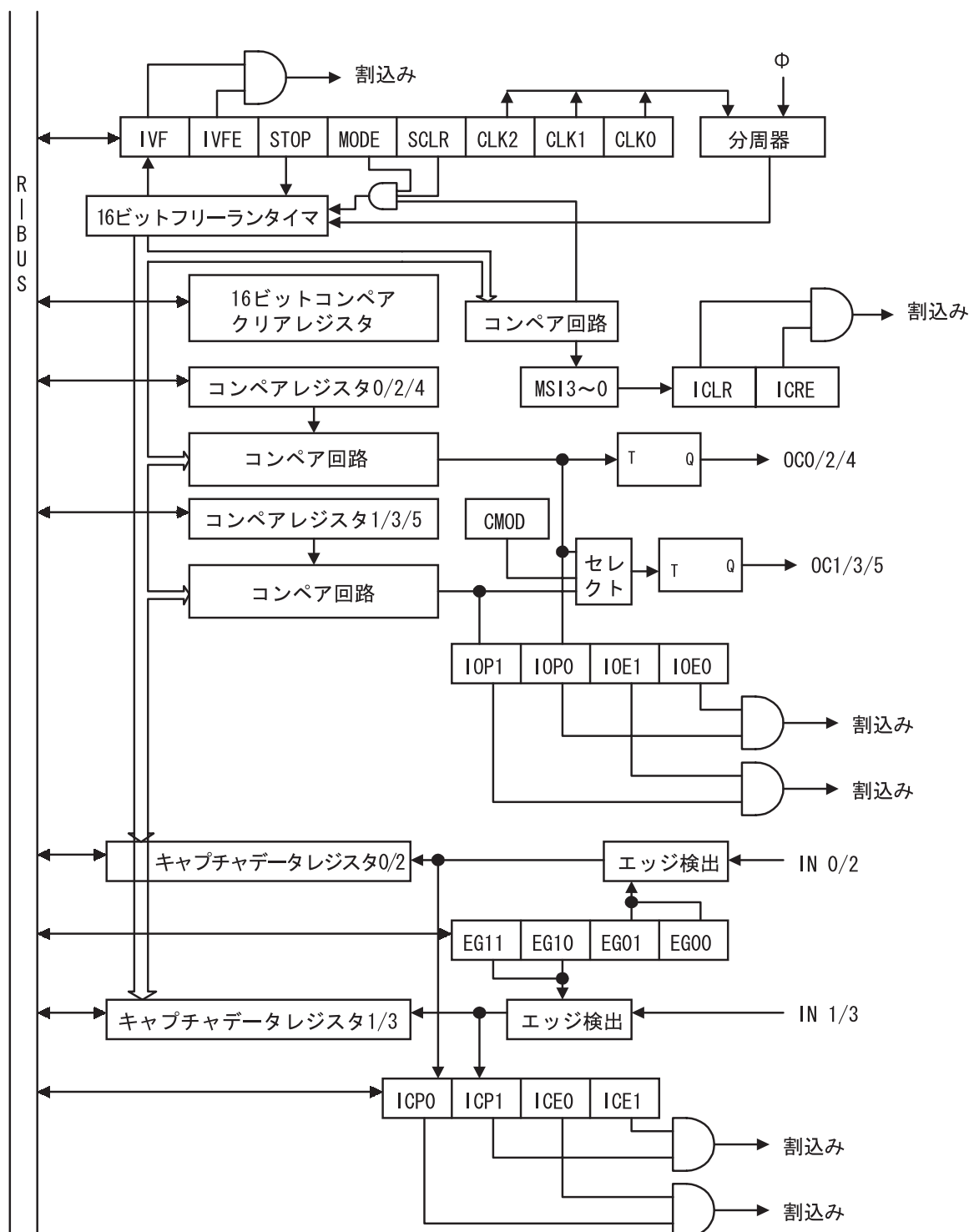


図 8.2-1 多機能タイマユニットのブロックダイアグラム

### 8.3 多機能タイマユニットのレジスタ

図 8.3-1に多機能タイマユニットのレジスタ一覧を示します。

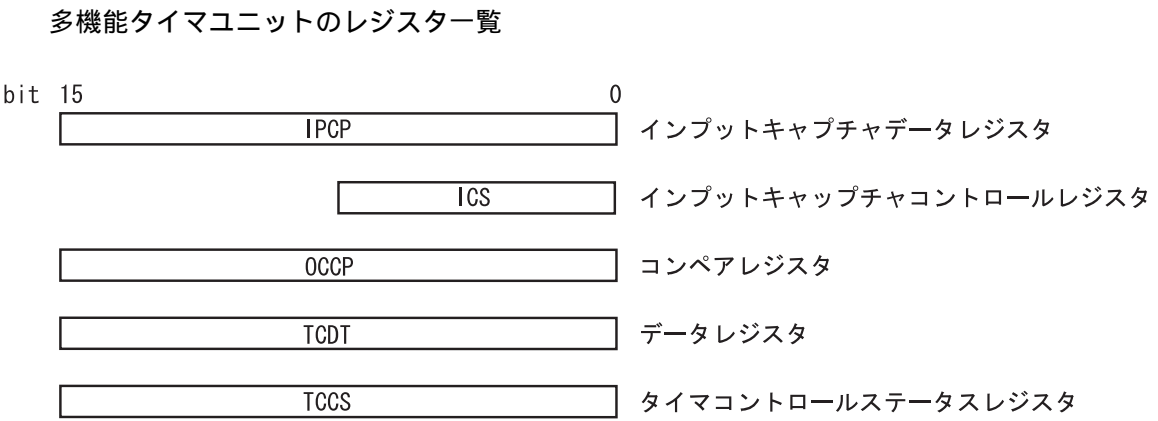


図 8.3-1 多機能タイマユニットのレジスタ一覧

### 8.3.1 16ビットフリーランタイマのレジスタ

16ビットフリーランタイマのレジスタには、次の三つのレジスタがあります。

- ・データレジスタ(TCDT)
- ・コンペアクリアレジスタ
- ・タイマコントロールステータスレジスタ(TCCS)

データレジスタ(TCDT)の構成

図 8.3-2に、データレジスタ(TCDT)のビット構成を示します。

データレジスタ上位	bit	15	14	13	12	11	10	9	8	初期値
アドレス 006C <sub>H</sub>		T15	T14	T13	T12	T11	T10	T09	T08	00000000
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	00000000 <sub>B</sub>
データレジスタ下位	bit	7	6	5	4	3	2	1	0	
アドレス 006D <sub>H</sub>		T07	T06	T05	T04	T03	T02	T01	T00	
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 8.3-2 データレジスタ(TCDT)のビット構成

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のできるレジスタです。カウンタ値は、リセット時に"0000<sub>H</sub>"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。

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
- タイマコントロールステータスレジスタ(TCCS)のクリア(SCLR)による初期化
- コンペアクリアレジスタ(Ch.6のコンペアレジスタ)値とタイマカウンタ値の一致による初期化(モード設定が必要です)

コンペアクリアレジスタ

16ビットフリーランタイマと比較する16ビット長のコンペアレジスタです。アウトプットコンペアのCh.6のコンペアレジスタが使用されます。このレジスタ値と16ビットフリーランタイマ値が一致したとき、16ビットフリーランタイマ値を"0000<sub>H</sub>"に初期化して、コンペアクリア割込みフラグをセットします。

また、割込み動作を許可している場合は、CPUに対して割込み要求を行います。



## タイマコントロールステータスレジスタ(TCCS)の構成

図 8.3-3に、タイマコントロールレジスタ(TCCS)のビット構成を示します。

タイマコントロール ステータスレジスタ上位	bit	15	14	13	12	11	10	9	8	初期値
アドレス 006E <sub>H</sub>		ECLK	—	—	—	—	—	—	—	0
	R/W	—	—	—	—	—	—	—	—	00000000 <sub>B</sub>
タイマコントロール ステータスレジスタ下位	bit	7	6	5	4	3	2	1	0	
アドレス 006F <sub>H</sub>		IVF	IVFE	STOP	MODE	SCLR	CLK2	CLK1	CLK0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 8.3-3 タイマコントロールレジスタ(TCCS)のビット構成

以下に、タイマコントロールレジスタ(TCCS)のビット機能を説明します。

【ビット15】:ECLK

16ビットフリーランタイマのカウンタクロックソースを内部か外部かを選択するビットです。この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。

ECLK	機能
0	内部クロックソースを選択 (初期値)
1	外部端子(FRCK)よりクロックを入力します

< 注意事項 >

内部クロックを選択した場合は、ビット2～ビット0(CLK2～CLK0)にカウンタクロックの設定を行ってください。このカウンタクロックがベースクロックとなります。また、FRCKよりクロックを入力する場合は、対応するDDRビットを”0”に設定してください。

【ビット14～8】: (reserved)

これらのビットは、未使用ビットです。

【ビット7】: IVF

16ビットフリーランタイマの割込み要求フラグです。16ビットフリーランタイマがオーバーフローを起こしたときにこのビットは”1”にセットされます。割込み要求許可ビット(ビット6: IVFE)がセットされていると割込みが発生します。このビットは”0”書き込みによりクリアされます。”1”書き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、常に”1”が読めます。

IVF	機能
0	割込み要求なし (初期値)
1	割込み要求あり

【ビット6】: IVFE

16ビットフリーランタイマの割込み許可ビットです。このビットが”1”のとき、割込みフラグ(ビット7: IVF)が”1”セットされると割込みが発生します。

IVFE	機能
0	割込み禁止 (初期値)
1	割込み許可

## 【ビット5】:STOP

16ビットフリーランタイマのカウントを停止するためのビットです。”1”書込み時にタイマのカウント停止。”0”書込み時にタイマのカウントを開始します。

STOP	機能
0	カウント許可 (動作) (初期値)
1	カウント禁止 (停止)

## &lt; 注意事項 &gt;

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

## 【ビット4】: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(ビット3:SCLR)でカウンタ値を初期化可能。”1”のときは、リセットとクリアビット(ビット3:SCLR)のほかにアウトプットコンペアのコンペアレジスタ6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。

MODE	機能
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6による初期化

## &lt; 注意事項 &gt;

カウンタ値の初期化はカウント値の変化点で行われます。

## 【ビット3】:SCLR

動作中の16ビットフリーランタイマ値を”0000<sub>H</sub>”に初期化するためのビットです。”1”書込み時にカウンタを”0000<sub>H</sub>”に初期化。”0”を書き込んでも意味を持ちません。読出し値は、常に”0”です。カウンタ値の初期化は、カウント値の変化点で行われます。

SCLR	フラグの意味
0	意味を持ちません (初期値)
1	カウンタ値を”0000 <sub>H</sub> ”に初期化します

## &lt; 注意事項 &gt;

タイマ停止中に初期化する場合は、データレジスタに”0000<sub>H</sub>”を書き込んでください。

## 【ビット2,1,0】:CLK2,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するビットです。このビットに書込み後すぐにクロックは変更されますのでアウトプットコンペア、インプットキャプチャが停止状態のときに変更してください。

CLK2	CLK1	CLK0	カウント クロック	=25MHz	=16MHz	=12.5MHz	=8MHz
0	0	0		40ns	62.5ns	80ns	0.125 μs
0	0	1	/2	80ns	0.125 μs	0.16 μs	0.25 μs
0	1	0	/4	0.16 μs	0.25 μs	0.32 μs	0.5 μs
0	1	1	/8	0.32 μs	0.5 μs	0.64 μs	1 μs
1	0	0	/16	0.64 μs	1 μs	1.28 μs	2 μs
1	0	1	/32	1.28 μs	2 μs	2.56 μs	4 μs
1	1	0	/64	2.56 μs	4 μs	5.12 μs	8 μs
1	1	1	/128	5.12 μs	8 μs	10.24 μs	16 μs

=マシクロック

### 8.3.2 アウトプットコンペアのレジスタ

アウトプットコンペアのレジスタには、次の二つのレジスタがあります。

- ・コンペアレジスタ(OCCP0～3)
- ・アウトプットコントロールレジスタ(OC0S0～3)

コンペアレジスタ(OCCP0～3)の構成

図 8.3-4に、コンペアレジスタ(OCCP0～3)のビット構成を示します。

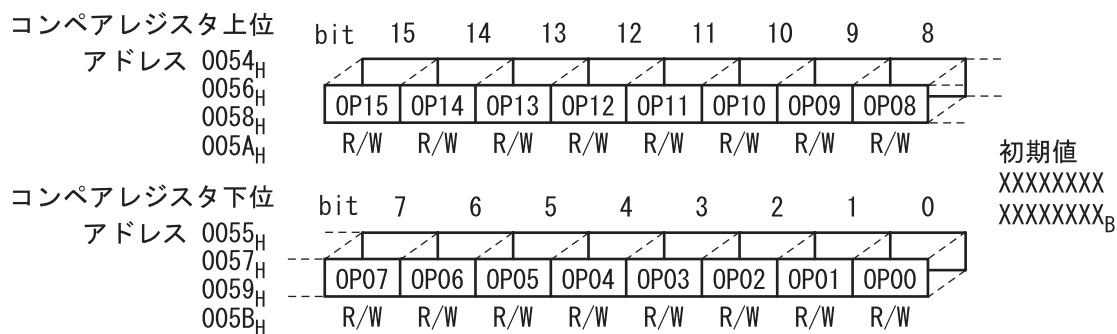


図 8.3-4 コンペアレジスタ(OCCP0～3)のビット構成

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。このレジスタ値は、初期値不定ですので設定してから起動を許可してください。このレジスタはワードアクセスしてください。このレジスタ値と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。

#### < 注意事項 >

コンペアレジスタを書き換える場合はコンペア割込みのルーチン内で行うかコンペアをディスエーブルの状態で行い、コンペア一致と書込みが同時に発生しないようにしてください。

アウトプットコントロールレジスタ(OCs0~3)の構成

図 8.3-5に、アウトプットコントロールレジスタ(OCs0~3)のビット構成を示します。

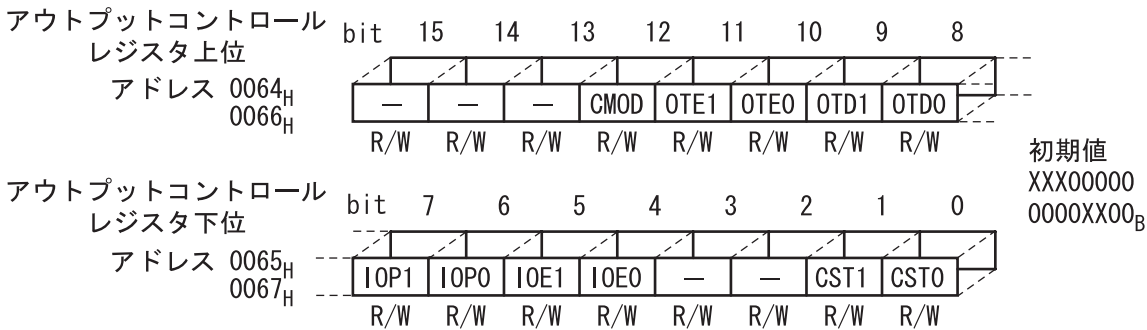


図 8.3-5 アウトプットコントロールレジスタ(OCs0~3)のビット構成

ch0,ch1について説明しますので,ch2,ch3はそれぞれch0 ch2,ch1 ch3と読み替えてください。

以下に、アウトプットコントロールレジスタ(OCs0~3)のビット機能を説明します。

【ビット12】:CMOD

端子出力を許可した場合(OTE1=0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替えます。

- 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。
  - OC0: コンペアレジスタ0の一致によりレベルを反転します。
  - OC1: 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のときはコンペアレジスタ0は,CMOD=0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(OC1)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が同じ値のときは,コンペアレジスタ1本のときと同じ動作をします。
  - OC0: コンペアレジスタ0の一致によりレベルを反転します。
  - OC1: コンペアレジスタ0と1の一致によりレベルを反転します。

【ビット11,10】: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。

OTE	機能
0	汎用ポートとして動作します (初期値)
1	アウトプットコンペア端子出力になります

OTE1: アウトプットコンペア1に対応  
 OTE0: アウトプットコンペア0に対応

## 【ビット9,8】:ODT1,ODT0

アウトプットコンペアレジスタ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。

ODT	機能
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

ODT1: アウトプットコンペア1に対応

ODT0: アウトプットコンペア0に対応

## 【ビット7,6】:IOP1,IOP0

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(IOE1,IOE0)が許可されているときにこのビットが"1"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。このビットは"0"書込みによりクリアされ"1"書込みでは意味を持ちません。リードモディファイライト系の命令では"1"が読めます。

IOP	機能
0	アウトプットコンペア一致なし (初期値)
1	アウトプットコンペア一致あり

IOP1: アウトプットコンペア1に対応,

IOP0: アウトプットコンペア0に対応

## 【ビット5,4】:IOE1,IOE0

アウトプットコンペアの割込みを許可するビットです。このビットが"1"のとき、割込みフラグ(IOP1,IOP0)が"1"に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。

IOE	機能
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

IOE1: アウトプットコンペア1に対応

IOE0: アウトプットコンペア0に対応

## 【ビット3,2】: (reserved)

これらのビットは、未使用ビットです。

## 【ビット1,0】:CST1,CST0

アウトプットコンペアの割込みを許可するビットです。このビットが"1"のとき、割込みフラグ(IOP1,IOP0)が"1"にセットされると、アウトプットコンペア割込みが発生します。

CST	機能
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

IOE1: アウトプットコンペア1に対応

IOE0: アウトプットコンペア0に対応

## &lt; 注意事項 &gt;

アウトプットコンペアは、16ビットフリーランタイム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。

### 8.3.3 インพุットキャプチャのレジスタ

- インพุットキャプチャデータレジスタには, 次の二つのレジスタがあります。
- ・インพุットキャプチャデータレジスタ(IPCP0~3)
  - ・インพุットキャプチャコントロールレジスタ(ICS01, ICS23)

インพุットキャプチャデータレジスタ(IPCP0~3)の構成

図 8.3-6に, インพุットキャプチャデータレジスタ(IPCP0~3)のビット構成を示します。

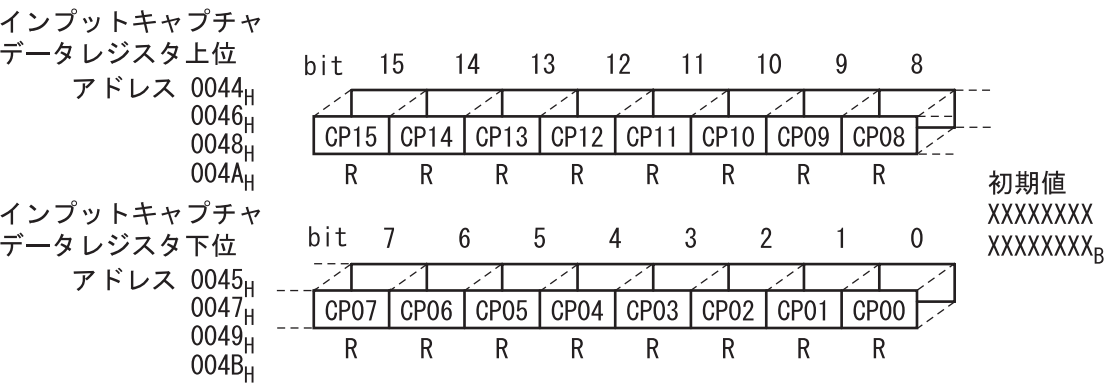


図 8.3-6 インพุットキャプチャデータレジスタ(IPCP0~3)のビット構成

このレジスタは, 対応した外部端子入力波形の有効エッジを検出したとき, 16ビットフリーランタイム値を保持するレジスタです。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コントロールレジスタ(ICS01, ICS23)の構成

図 8.3-7に、インプットキャプチャコントロールレジスタ(ICS01, ICS23)のビット構成を示します。

キャップチャコントロール レジスタ(ICS23)	bit	7	6	5	4	3	2	1	0	初期値
アドレス 004D <sub>H</sub>		ICP3	ICP2	ICE3	ICE2	EG31	EG30	EG21	EG20	00000000
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	00000000 <sub>B</sub>
キャップチャコントロール レジスタ(ICS01)	bit	7	6	5	4	3	2	1	0	
アドレス 004F <sub>H</sub>		ICP1	ICP0	ICE1	ICE0	EG11	EG10	EG01	EG00	
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 8.3-7 インプットキャプチャコントロールレジスタ(ICS01, ICS23)のビット構成

以下に、インプットキャプチャコントロールレジスタ(ICS01, ICS23)のビット機能を説明します。

【ビット7,6】: ICP3, ICP2, ICP1, ICP0

これらのビットは、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このビットを”1”にセットします。割込み許可ビット(ICE3, ICE2, ICE1, ICE0)がセットされていると、有効エッジを検出することにより割込みを発生することができます。このビットは、”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の命令では”1”が読めます。

ICP	機能
0	有効エッジ検出なし (初期値)
1	有効エッジ検出あり

ICPn:nの番号がインプットキャプチャのチャンネル番号に対応します。

【ビット5,4】: ICE3, ICE2, ICE1, ICE0

インプットキャプチャ割込み許可ビットです。このビットが”1”のとき割込みフラグ(ICP3, ICP2, ICP1, ICP0)が”1”にセットされるとインプットキャプチャ割込みが発生します。

ICE	機能
0	割込み禁止 (初期値)
1	割込み許可

ICE<sub>n</sub>:nの番号がインプットキャプチャのチャンネル番号に対応します

【ビット3～0】: EG31/30, EG21/20, EG11/10, EG01/00

外部入力の有効エッジ極性を選択するビットです。インプットキャプチャ動作許可も兼用しています。

EG31	EG30	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

EG<sub>n</sub>1/EG<sub>n</sub>0:

nの番号がインプットキャプチャのチャンネル番号に対応します。

## 8.4 多機能タイマユニットの動作

---

多機能タイマユニットの動作について説明します。

---

### 多機能タイマの動作説明

#### 16ビットフリーランタイマ

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<sub>h</sub>"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ビットアウトプットコンペア

16ビットアウトプットコンペアは、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マ値との値を比較して一致したら割込み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。

#### 16ビットインプットキャプチャ

16ビットインプットキャプチャは、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マ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。



### 8.4.1 16ビットフリーランタイマ部の動作

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<sub>H</sub>"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説明

カウンタ値は、次の条件でクリアされます。

- オーバフローが発生したとき。
- コンペアクリアレジスタ(アウトプットコンペアCh6のコンペアレジスタ)値とコンペアマッチしたとき。(モード設定が必要)
- 動作中にタイマコントロールステータスレジスタ(TCCS)のSCLRビットに"1"を書き込んだとき。
- タイマ停止中にTCDDTに"0000<sub>H</sub>"を書き込んだとき。

割込みは、オーバフローが発生したとき、コンペアクリアレジスタ値とコンペアマッチしてカウンタがクリアされたとき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。)

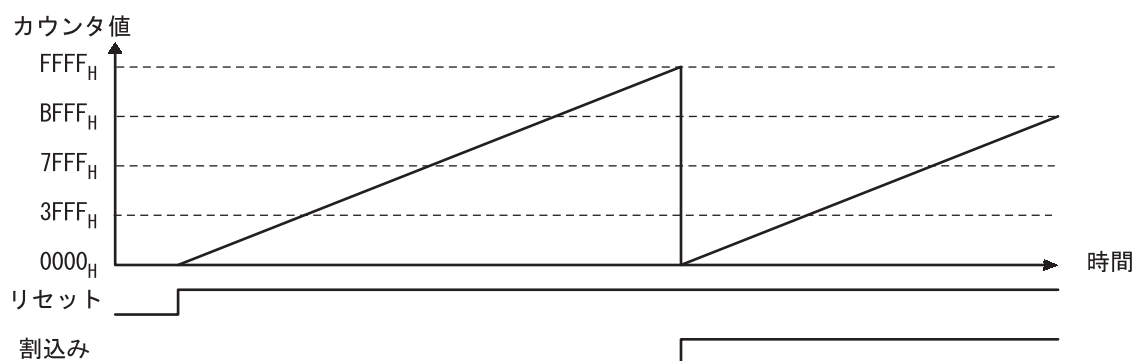


図 8.4-1 オーバフローによるカウンタの起動と動作のタイミング

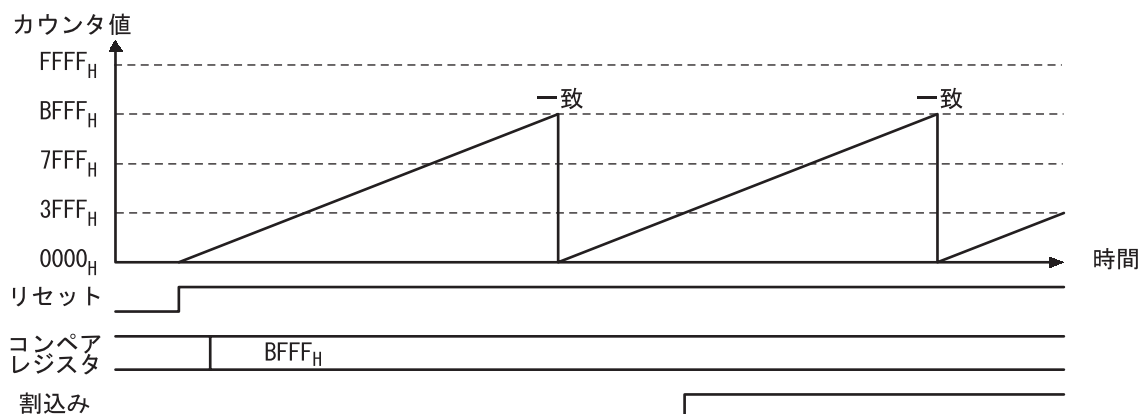


図 8.4-2 コンペアクリアレジスタ値とコンペアマッチしたときのカウンタクリア

16ビットフリーランタイマのクリアタイミング

カウンタのクリアは、リセット、ソフトウェア、コンペアクリアレジスタとの一致で行われます。リセットとソフトウェアでのカウンタクリアは、クリア発生とともに行われますが、コンペアクリアレジスタとの一致によるカウンタクリアはカウンタタイミングに同期して行われます。

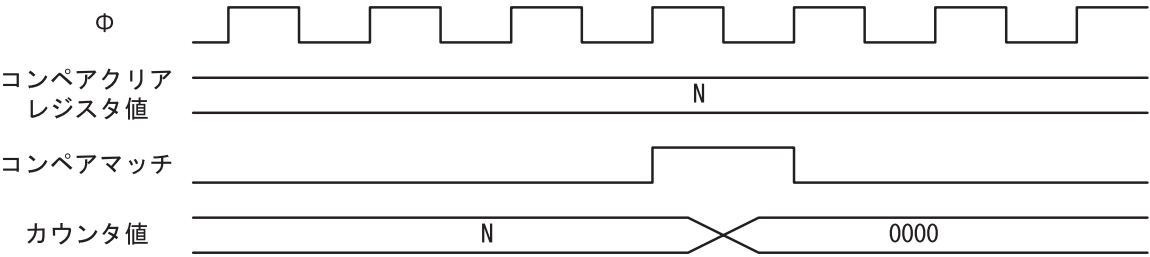


図 8.4-3 フリーランタイマのクリアタイミング

16ビットフリーランタイマのカウントタイミング

16ビットフリーランタイマは、入力されたクロック(内部または外部クロック)によりカウントアップされます。外部クロック選択時は、立上りエッジでカウントされます。

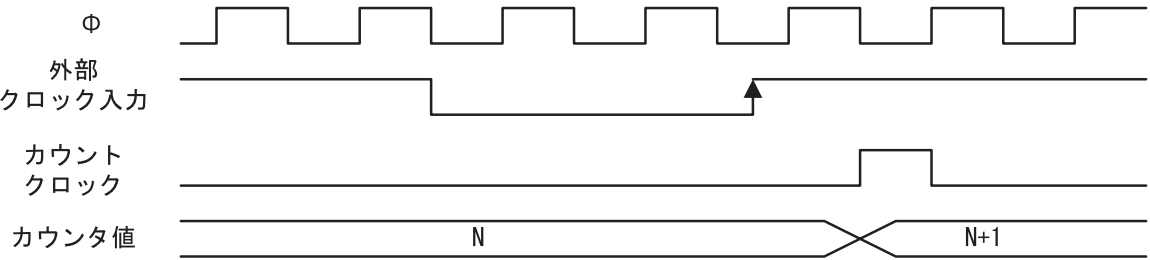


図 8.4-4 16ビットフリーランタイマのカウントタイミング

### 8.4.2 16ビットアウトプットコンペア部の動作

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値との値を比較して一致したら割込み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。

16ビットアウトプットコンペアの動作説明

1チャンネル独立でコンペア動作を行うことができます。(CMOD=0のとき)

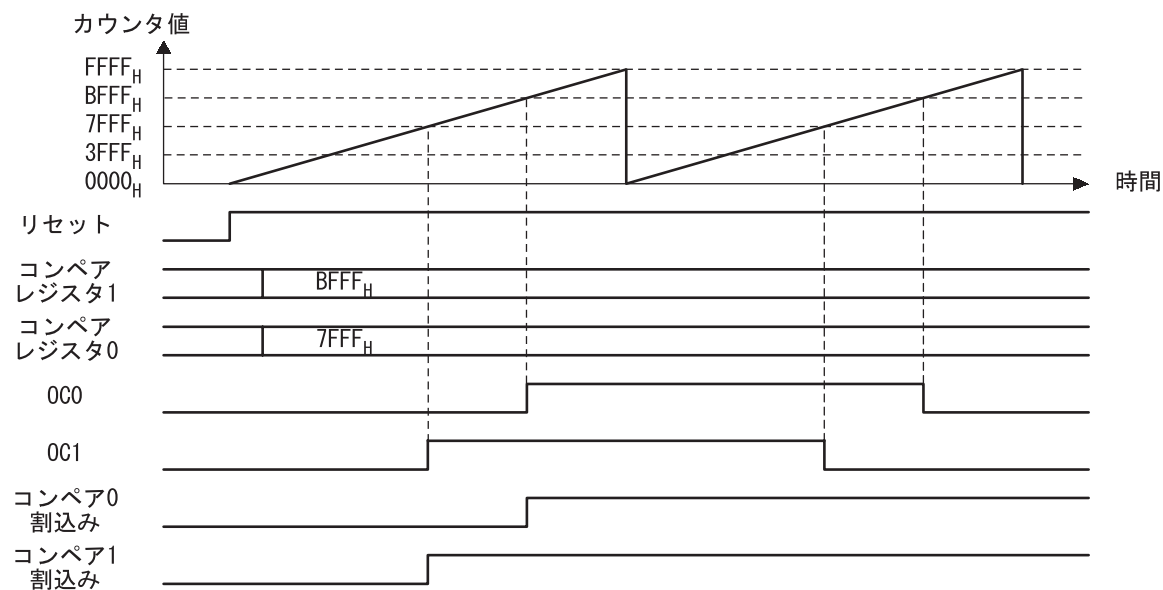


図 8.4-5 コンペアレジスタ0,1を使用したときの出力波形例(出力の初期は"0")

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)

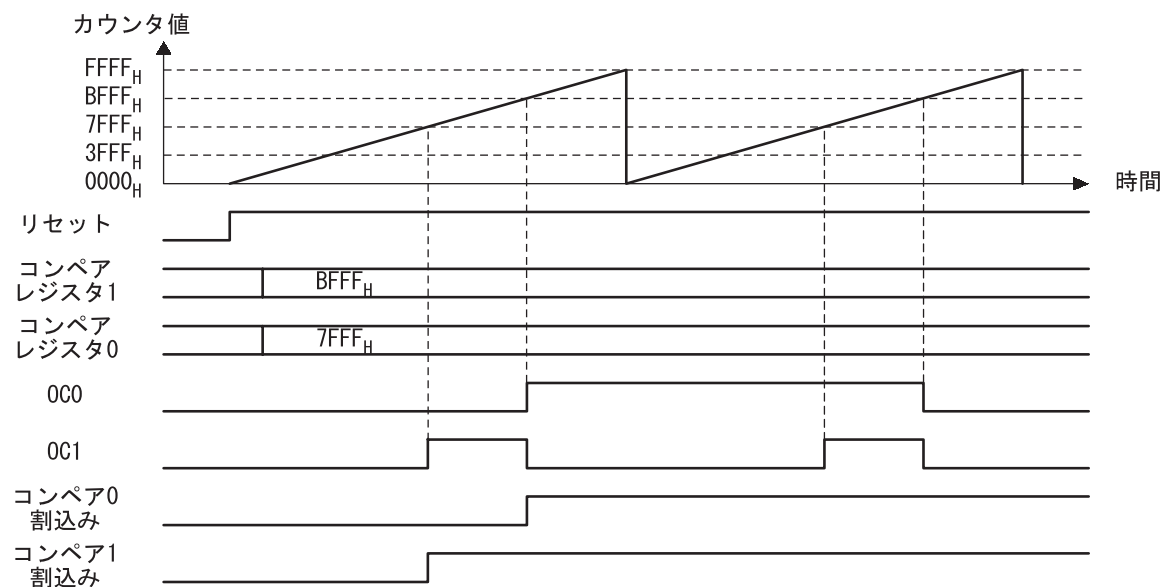


図 8.4-6 コンペアレジスタ0,1を使用したときの出力波形例(出力の初期値は"0")

16ビットアウトプットコンペアのタイミング

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)

アウトプットコンペアは、フリーランタイマと設定したコンペアレジスタの値が一致したときにコンペアマッチ信号が発生して出力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。コンペアレジスタ書き換え時のカウンタ値とはコンペアしません。

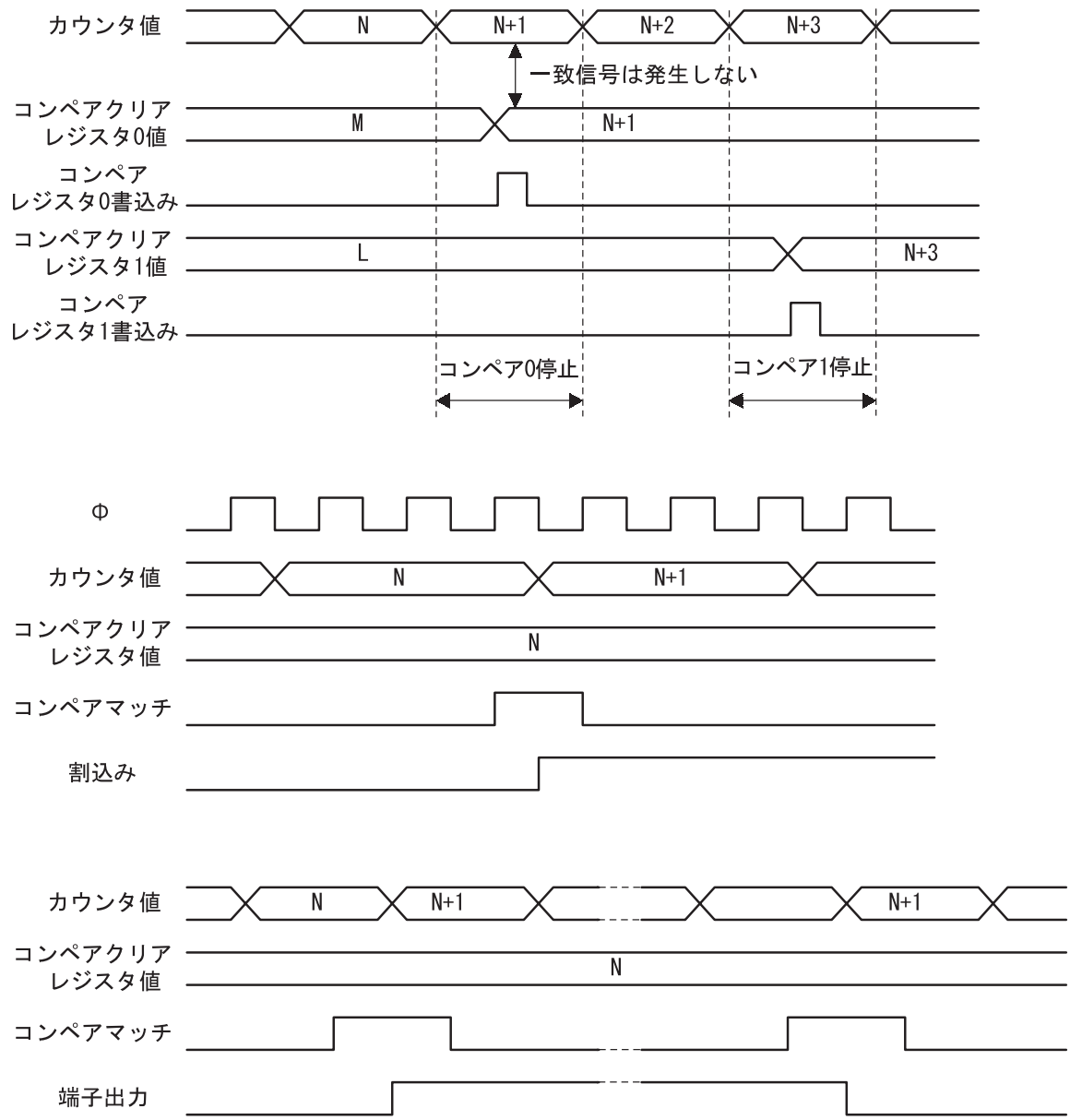


図 8.4-7 16ビットアウトプットコンペアのタイミング

### 8.4.3 16ビットインプットキャプチャ部の動作

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。

16ビットインプットキャプチャの動作

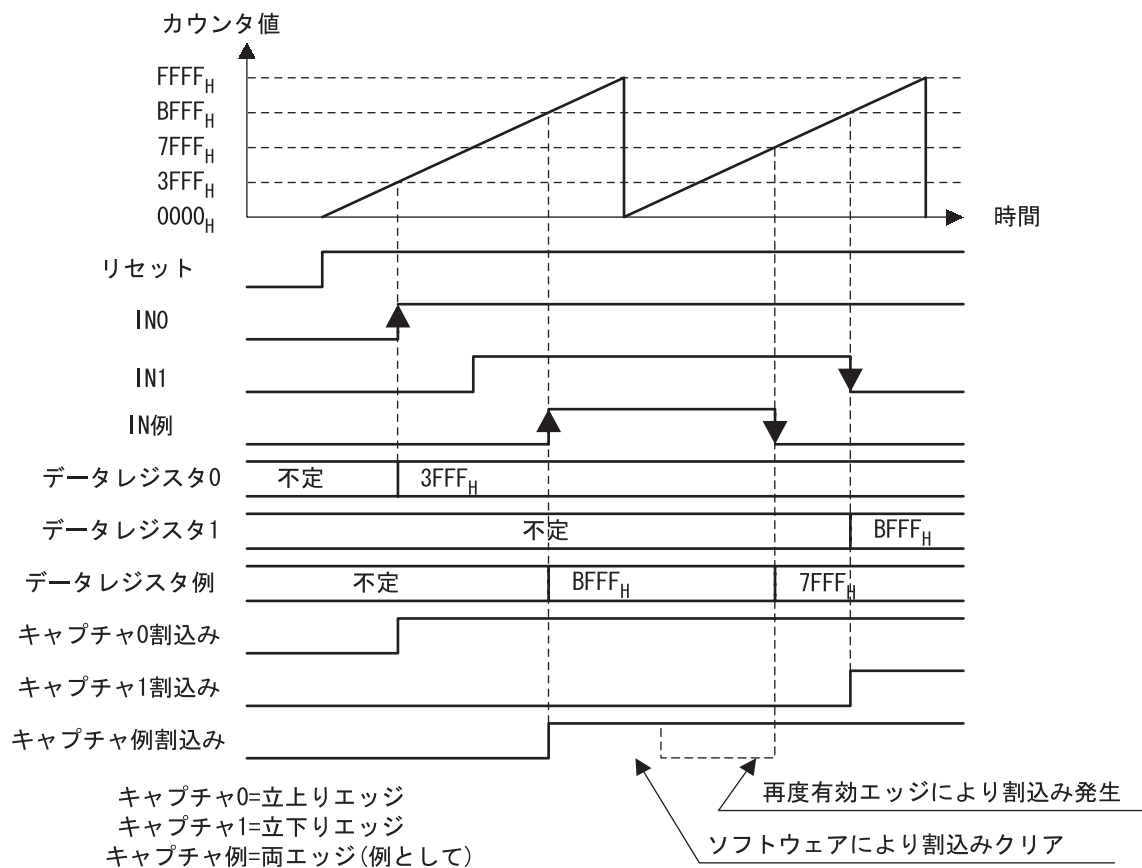


図 8.4-8 インプットキャプチャの取り込みタイミング例

16ビットインプットキャプチャのタイミング

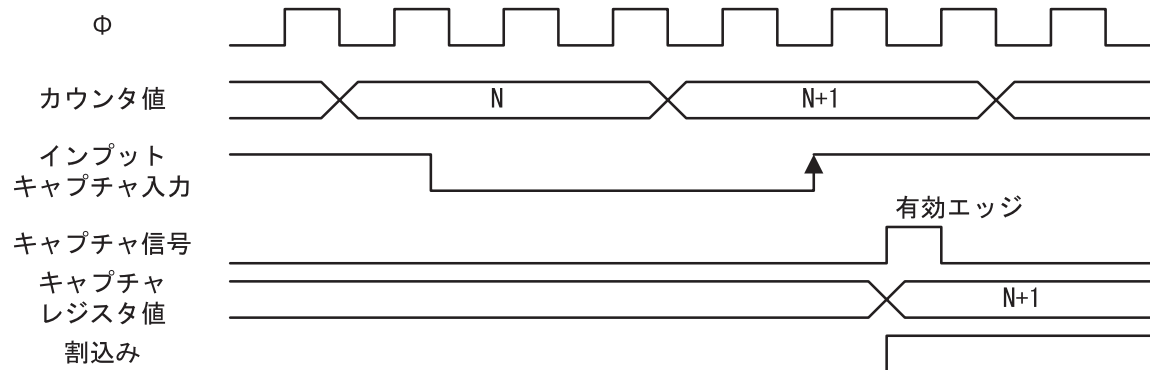


図 8.4-9 16ビットインプットキャプチャのタイミング

## 第9章 U-TIMER

---

この章では,U-TIMERの概要,レジスタの構成/機能および動作について説明します。

---

- 9.1 U-TIMERの概要
- 9.2 U-TIMERのレジスタ
- 9.3 U-TIMERの動作

## 9.1 U-TIMERの概要

U-TIMER(16 bit timer for UART baud rate generation)の概要とブロックダイアグラムを示します。

### U-TIMERの概要

U-TIMERは,UARTのボーレートを発生するための16ビットタイマです。チップの動作周波数と,U-TIMERのリロード値の組み合わせで任意のボーレートを設定できます。

また,カウントアンダフローで割込みを発生するので,インターバルタイマとしても使用できます。

MB91F127/128は,U-TIMERを3チャンネル内蔵しています。インターバルタイマとして使用する場合,2組のU-TIMER(ch0,1)をカスケードして使用でき,最大 $2^{32} \times$  のインタバルをカウントできます。

### ブロックダイアグラム

図 9.1-1に,U-TIMERのブロックダイアグラムを示します。

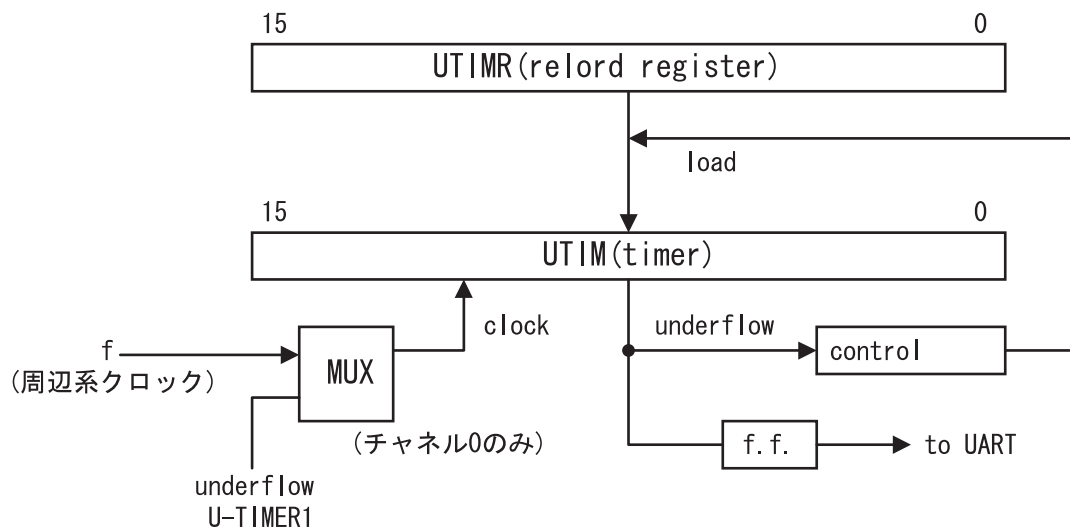


図 9.1-1 U-TIMERのブロックダイアグラム

## 9.2 U-TIMERのレジスタ

U-TIMERのレジスタには、以下の三つがあります。

- ・Uタイマ値レジスタ (UTIM)
- ・リロードレジスタ (UTIMR)
- ・Uタイマ制御レジスタ (UTIMC)

U-TIMERのレジスタ一覧

図 9.2-1に、U-TIMERのレジスタ一覧を示します。

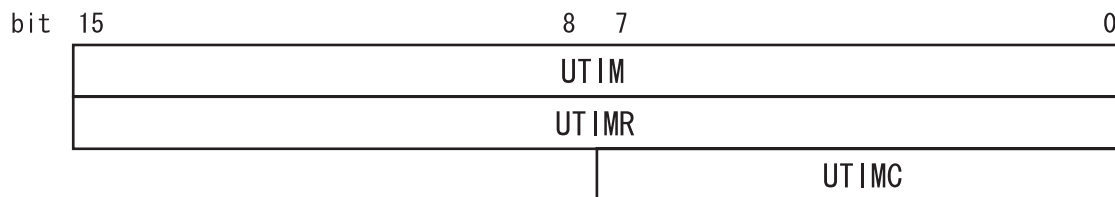


図 9.2-1 U-TIMERのレジスタ構成

Uタイマ値レジスタ(UTIM)

図 9.2-2に、Uタイマ値レジスタ(UTIM)のビット構成を示します。

アドレス	bit 15	14	.....	2	1	0	初期値
0000 0078 <sub>H</sub> (ch. 0)	b15	b14	.....	b2	b1	b0	00000000 00000000 <sub>B</sub>
0000 007C <sub>H</sub> (ch. 1)			.....				
0000 0080 <sub>H</sub> (ch. 2)	R	R	.....	R	R	R	

図 9.2-2 Uタイマ値レジスタ(UTIM)のビット構成

UTIMは、タイマの値を示します。16ビット転送命令でアクセスしてください。

リロードレジスタ(UTIMR)

図 9.2-3に、リロードレジスタ(UTIMR)のビット構成を示します。

アドレス	bit 15	14	.....	2	1	0	初期値
0000 0078 <sub>H</sub> (ch. 0)	b15	b14	.....	b2	b1	b0	00000000 00000000 <sub>B</sub>
0000 007C <sub>H</sub> (ch. 1)			.....				
0000 0080 <sub>H</sub> (ch. 2)	W	W	.....	W	W	W	

図 9.2-3 リロードレジスタ(UTIMR)のビット構成

UTIMRは、UTIMがアンダフローしたときにUTIMにリロードされる値を格納するレジスタです。

16ビット転送命令でアクセスしてください。



## U-TIMER制御レジスタ(UTIMC)

図 9.2-4に,U-TIMER制御レジスタ(UTIMC)のビット構成を示します。

アドレス	bit	7	6	5	4	3	2	1	0	初期値
0000 007B <sub>H</sub> (ch. 0)		UCC1	—	—	UTIE	UNDR	CLKS	UTST	UTCRC	0—00001 <sub>B</sub>
0000 007F <sub>H</sub> (ch. 1)		R/W	—	—	R/W	R/W	R/W	R/W	R/W	
0000 0083 <sub>H</sub> (ch. 2)										

図 9.2-4 U-TIMER制御レジスタ(UTIMC)のビット構成

UTIMCは,U-TIMERの動作を制御します。

## U-TIMER制御レジスタ(UTIMC)のビット詳細

以下に,U-TIMER制御レジスタ(UTIMC)の各ビットの機能を説明します。

## 【ビット7】 UCC1(U-timer Count Control 1)

このビットは,U-TIMERのカウントの仕方を制御します。

UCC1	機能
0	通常動作 $=2n+2$ (初期値)
1	+1モード $=2n+3$

n:UTIMRの設定値(10進数)

:UARTに対する出力クロックの周期

U-TIMERは,UARTに対して通常の $2(n+1)$ 周期のクロックのほかに奇数分周を設定できます。

UCC1を1に設定すると, $2n+3$ の周期を発生します。

設定例:1.UTIMR= 5, UCC1=0 発生周期= $2n+2=$  12サイクル

2.UTIMR=25, UCC1=1 発生周期= $2n+3=$  53サイクル

3.UTIMR=60, UCC1=0 発生周期= $2n+2=$ 122サイクル

U-TIMERをインターバルタイマとして使用する場合は,UCC1の値を"0"に設定してください。

## 【ビット6,5】 (reserved)

これらのビットは,予約ビットです。

## 【ビット4】 UTIE(U-TIMER Interrupt Enable)

このビットは,U-TIMERのアンダフローによる割込み許可ビットです。

UTIE	機能
0	割込み禁止 (初期値)
1	割込み許可

## 【ビット3】 UNDR(UNDeR flow flag)

このビットは,アンダフローが発生したことを示すフラグです。

ビット4のUTIEが"1"のときにUNDRがセットされると,アンダフロー割込みが発生します。UNDRは,リセットまたは"0"書込みによりクリアされます。

リードモディファイライト系命令の読出し時は,常に"1"が読み出されます。

また,UNDRビットに対する"1"書込みは無効です。

## 【ビット2】 CLKS(CLock Select)

このビットは,U-TIMERのチャンネル0とチャンネル1のカスケード指定ビットです。

CLKS	機能
0	クロックソースは周辺系クロック( ) (初期値)
1	U-TIMERチャンネル0のソースクロックタイミングとして,チャンネル1のアンダフロー信号を使用 *1

\*1:ブロックダイアグラムのf.f.

CLKSはチャンネル0でのみ有効です。チャンネル1では常に"0"にしてください。

## 【ビット1】 UTST(U-Timer SStart)

このビットは,U-TIMERの動作許可ビットです。

UTST	機能
0	停止。動作中でも"0"の書き込みで停止します。 (初期値)
1	動作。動作中に"1"を書き込んでも動作を続行します。

## 【ビット0】 UTCR(U-Timer Clear)

UTCRに"0"を書き込むと,U-TIMERは0000<sub>H</sub>にクリアされます(f.f.も"0"にクリア)。

このビットは,常に"1"が読み出されます。

## U-TIMER制御レジスタ(UTIMC)使用上の注意事項

- ストップ状態からスタートビットUTSTをアサート(スタート)すると自動的にリロードします。
- ストップ状態からクリアビットUTCRとスタートビットUTSTを同時にアサートすると,カウンタを"0"にクリアして,直後のカウントダウンでアンダフローが発生します。
- 動作中にクリアビットUTCRをアサートすると,カウンタも"0"にクリアされます。このため,出力波形にヒゲ状の短いパルスが出力される場合があります,UARTやカスケードモードの上位側のU-TIMERが誤動作する可能性があります。出力クロックを使用している場合には,動作中にクリアビットによってクリアを行わないでください。
- カスケードモードで,下位側のリロードレジスタ(UTIMR)に"0"または"1"をセットすると正しくカウントされません。

### 9.3 U-TIMERの動作

この節では,U-TIMERのボーレートの計算と,カスケードモードのタイミングを示します。

#### ボーレートの計算

UARTは,対応するU-TIMER(U-TIMER0 UART0,U-TIMER1 UART1,U-TIMER2 UART2)のアンダフロフリップフロップ(ブロックダイアグラム中のf.f.)をボーレート用クロックソースとして使用します。

#### 非同期(調歩同期)モード

UARTは,U-TIMERの出力を16分周して使用します。

- UCC1=0の場合

$$\text{bps} = \frac{f}{(2n+2) \times 16}$$

- UCC1=1の場合

$$\text{bps} = \frac{f}{(2n+3) \times 16}$$

n:UTIMR(リロード値)

f:周辺系マシンのクロック周波数(ギアにより変動)

#### CLK同期モード

- UCC1=0の場合

$$\text{bps} = \frac{f}{(2n+2)}$$

- UCC1=1の場合

$$\text{bps} = \frac{f}{(2n+3)}$$

n:UTIMR(リロード値)

f:周辺系マシンのクロック周波数(ギアにより変動)

#### カスケードモード

U-TIMERチャンネル0と1は,カスケードモードで使用できます。

図 9.3-1に,UTIMR ch0を"0100<sub>H</sub>"に,UTIMR ch1を"0002<sub>H</sub>"に設定した場合のタイミングチャート例を示します。

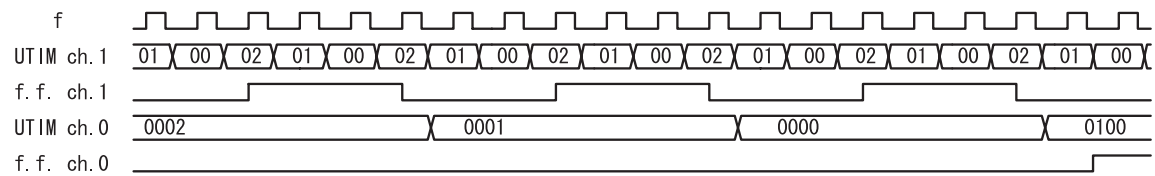


図 9.3-1 カスケードモードのタイミングチャート

## 第10章 外部割込み

---

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

---

- 10.1 外部割込みの概要
- 10.2 外部割込みのレジスタ
- 10.3 外部割込み動作

# 10.1 外部割込みの概要

外部割込みは、INT0 ~ INT5に入力される外部割込要求の制御を行うブロックです。  
検出する要求のレベルとして、”H”,”L”,”立上りエッジ”,”立下りエッジ”から選択できます。

外部割込みのブロックダイアグラム

図 10.1-1に、外部割込みのブロックダイアグラムを示します。

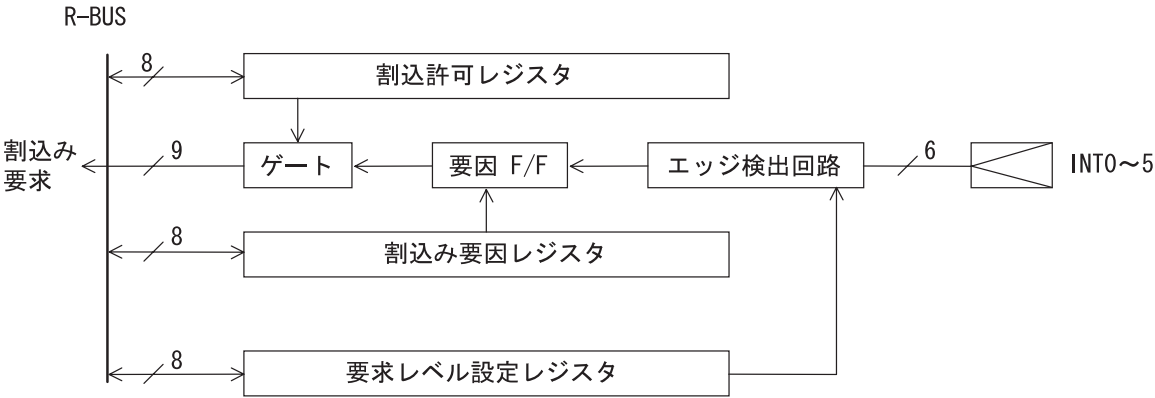


図 10.1-1 外部割込みのブロックダイアグラム

## 10.2 外部割込みのレジスタ

この節では、外部割込みで使用するレジスタの構成および機能について説明します。

外部割込みのレジスタ一覧

図 10.2-1に、外部割込みのレジスタ一覧を示します。

bit	7	6	5	4	3	2	1	0	
	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	外部割込み許可レジスタ (ENIR)
bit	15	14	13	12	11	10	9	8	
	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	外部割込み要因レジスタ (EIRR)
bit	7	6	5	4	3	2	1	0	
	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	要求レベル設定レジスタ (ELVR)
bit	15	14	13	12	11	10	9	8	
	LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	要求レベル設定レジスタ (EHVR)

図 10.2-1 外部割込みのレジスタ一覧

### 10.2.1 割込み許可レジスタ(ENIR)

割込み許可レジスタ(ENIR)は、外部割込み要求出力のマスク制御を行います。

割込み許可レジスタ(ENIR)の構成

図 10.2-2に、割込み許可レジスタ(ENIR:ENable Interrupt request Register)のビット構成を示します。

bit	7	6	5	4	3	2	1	0	初期値
アドレス 000095 <sub>H</sub>	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	00000000 <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 10.2-2 割込み許可レジスタ(ENIR)のビット構成

割込み許可レジスタ(ENIR)は、外部割込み要求出力のマスク制御を行います。このレジスタの”1”が書かれたビットに対応する割込み要求出力は許可され( INT0の許可をEN0が制御), 割込みコントローラに対して要求が出力されます。”0”が書かれたビットに対応する端子は割込み要因を保持しますが、割込みコントローラに対しては要求を発生しません。  
このデバイスでは、ビットEN6, EN7への書込みは意味を持ちません。  
ビットEN6, EN7には”0”を書き込んでください。

## 10.2.2 外部割込み要因レジスタ(EIRR)

外部割込み要因レジスタ(EIRR)のビット構成と機能を説明します。

外部割込み要因レジスタ(EIRR)の構成

図 10.2-3に、外部割込み要因レジスタ(EIRR:External Interrupt Request Register)のビット構成を示します。

bit	15	14	13	12	11	10	9	8	初期値
アドレス 000094 <sub>H</sub>	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	00000000 <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 10.2-3 外部割込み要因レジスタ(EIRR)のビット構成

外部割込み要因レジスタ(EIRR)は、読出し時には対応する外部割込み要求があることを示し、書込み時にはこの要求を示すフリップフロップ内容をクリアするレジスタです。このレジスタを読み出したときに”1”であった場合、このビットに対応する端子に外部割込み要求があることを示します。

また、このレジスタに”0”を書き込むと、対応するビットの要求フリップフロップがクリアされます。”1”の書込みは無効です。

リードモディファイライト系命令の読出し時には、”1”が読み出されます。



### 10.2.3 外部割込み要求レベル設定レジスタ(ELVR,EHVR)

外部割込み要求レベル設定レジスタ(ELVR,EHVR)のビット構成と機能を説明します。

外部割込み要求レベル設定レジスタ(ELVR,EHVR)の構成

図 10.2-4に,外部割込み要求レベル設定レジスタ(ELVR,EHVR)のビット構成を示します。

ELVR	bit	7	6	5	4	3	2	1	0	初期値
アドレス 000099 <sub>H</sub>		LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	00000000 <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
EHVR	bit	15	14	13	12	11	10	9	8	初期値
アドレス 000098 <sub>H</sub>		LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4	00000000 <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 10.2-4 外部割込み要求レベル設定レジスタ(ELVR,EHVR)のビット構成

外部割込み要求レベル設定レジスタ(ELVR,EHVR)は,要求検出の選択を行うレジスタです。INT0～INT5に2ビットずつが割り当てられています。

表 10.2-1に,設定内容を示します。

要求入力レベルの場合,外部割込み要因レジスタ(EIRR)の各ビットをクリアしても入力がアクティブレベルならば,該当するビットは再びセットされます。

表 10.2-1 外部割込み要求レベル割当て

LBx	LAx	動作
0	0	Lレベルで要求あり
0	1	Hレベルで要求あり
1	0	立上りエッジで要求あり
1	1	立下りエッジで要求あり

## 10.3 外部割込み動作

要求レベル,許可レジスタを設定した後,対応する端子に外部割込み要求レベル設定レジスタ(ELVR,EHVR)で設定された要求が入力されると,このモジュールは割込みコントローラに対して割込要求信号を発生します。

### 外部割込みの動作

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,この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,該当する割込みが発生します。

図 10.3-1に,外部割込みの動作を示します。

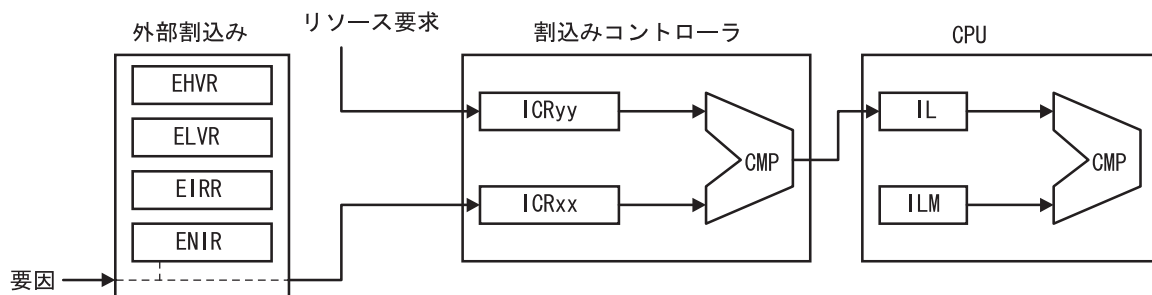


図 10.3-1 外部割込みの動作

### ストップからの復帰

クロック停止モードのストップ状態からの復帰に外部割込みを使う場合は,入力要求をHレベル要求としてください。Lレベル要求では誤動作を起こす可能性があります。

エッジ要求ではクロック停止モードのストップ状態からの復帰は行われません。

### 外部割込みの動作手順

外部割込み部に存在するレジスタの設定は,次の手順で行ってください。

- 1) 許可レジスタの対象となるビットをdisable状態にする。
- 2) 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。
- 3) 要因レジスタの対象となるビットをクリアする。
- 4) 許可レジスタの対象となるビットをenable状態にする。

ただし,3)と4)は,16ビットデータによる同時書込みができます。

このモジュール内のレジスタを設定するときには,必ず許可レジスタをdisable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enable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,レジスタ設定時に割込み許可状態時に誤って割込み要因が起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(周辺系クロックマシンサイクル)必要とします。

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、内部に要因保持回路が存在するので、割込みコントローラへの要求はアクティブのままです。

割り込みコントローラへの要求を取り下げるには、要因レジスタをクリアする必要があります。

図 10.3-2 にレベル設定時の要因保持回路のクリアを、図 10.3-3 に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を示します。

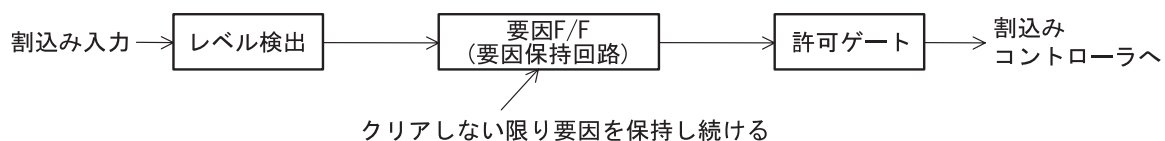


図 10.3-2 レベル設定時の要因保持回路のクリア

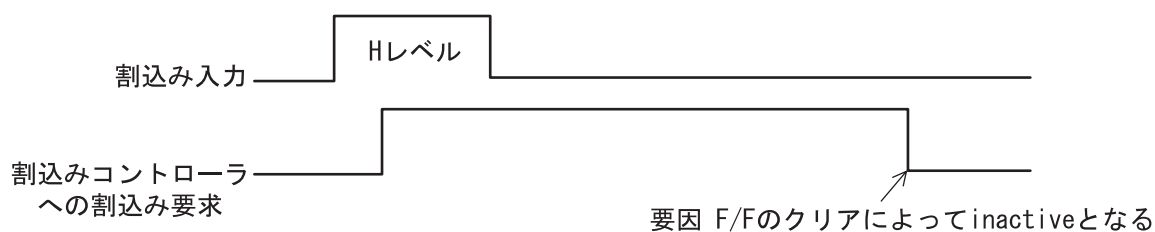


図 10.3-3 割込み許可時の割込み要因と割込みコントローラへの割込み要求

## 第11章 遅延割込みモジュール

---

この章では、遅延割込みモジュールの機能と動作について説明します。

---

- 11.1 遅延割込みモジュールの概要
- 11.2 遅延割込みモジュールのレジスタ
- 11.3 遅延割込みモジュールの動作

# 11.1 遅延割込みモジュールの概要

遅延割込みモジュールは、タスク切替え用の割込みを発生するためのモジュールです。このモジュールを使用することで、ソフトウェアで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。

遅延割込みモジュールのブロックダイアグラム

図 11.1-1に、遅延割込みモジュールのブロックダイアグラムを示します。

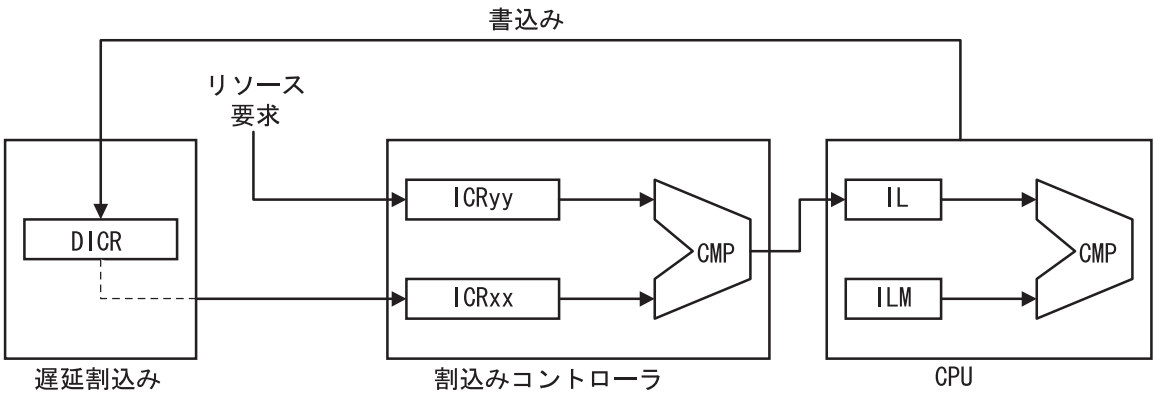


図 11.1-1 遅延割込みモジュールのブロックダイアグラム

## 11.2 遅延割込みモジュールのレジスタ

この節では、遅延割込みモジュールで使用するレジスタの構成および機能について説明します。

遅延割込みモジュールのレジスタ一覧

遅延割込みモジュールには、遅延割込み制御レジスタ(DICR)があります。

図 11.2-1に、遅延割込み制御レジスタ(DICR)の構成を示します。

bit	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	—	DLYI	遅延割込み制御レジスタ(DICR)

図 11.2-1 遅延割込み制御レジスタ(DICR)の構成

遅延割込み制御レジスタ(DICR)の構成

遅延割込み制御レジスタ(DICR:Delayed Interrupt Control Register)は、遅延割込みを制御するレジスタです。

図 11.2-2に、遅延割込み制御レジスタ(DICR)のビット構成を示します。

	bit	7	6	5	4	3	2	1	0	
アドレス 00000430 <sub>H</sub>		—	—	—	—	—	—	—	DLYI	初期値 -----0 <sub>B</sub>
										R/W

図 11.2-2 遅延割込み制御レジスタ(DICR)のビット構成

以下に、遅延割込み制御レジスタ(DICR)のビット機能を説明します。

【ビット0】 DLYI

このビットは、該当する割込み要因の発生/解除を制御します。

DLYI	機能
0	遅延割込要因の解除/要求なし (初期値)
1	遅延割込要因の発生

## 11.3 遅延割込みモジュールの動作

---

遅延割込みは、タスク切替え用の割込みを発生するものです。この機能を使用することにより、ソフトウェアで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。

---

### 割込み番号

遅延割込みは、最も大きな割込み番号に対応した割込み要因に割り当てられています。

MB91F127/128では、遅延割込みを割込み番号63(3F<sub>H</sub>)に割り当てています。

### DICRのDLYIビット

このビットに"1"を書き込むことによって、遅延割込み要因が発生します。また、"0"を書き込むことによって、遅延割込み要因を解除します。

このビットは、一般の割込みにおける割込み要因フラグと同じものであり、割込みルーチンの中でこのビットをクリアし、合わせてタスクの切替えを行うようにしてください。

## 第12章 割込みコントローラ

---

この章では、割込みコントローラの概要、レジスタの構成/機能、割込みコントローラの動作およびホールドリクエスト取下げ要求機能の使用例について説明します。

---

- 12.1 割込みコントローラの概要
- 12.2 割込みコントローラのレジスタ
- 12.3 割込みコントローラの動作
- 12.4 ホールドリクエスト取下げ要求機能(HRCR)の使用例



# 12.1 割込みコントローラの概要

割込みコントローラは, 割込み受付/調停処理を行います。

割込みコントローラのハードウェア構成

割込みコントローラは, 以下のものから構成されます。

- 割込み制御レジスタ (ICR)
- 割込み優先度判定回路
- 割込みレベル, 割込み番号 (ベクタ) 発生部
- HOLDリクエスト取下げ要求発生部

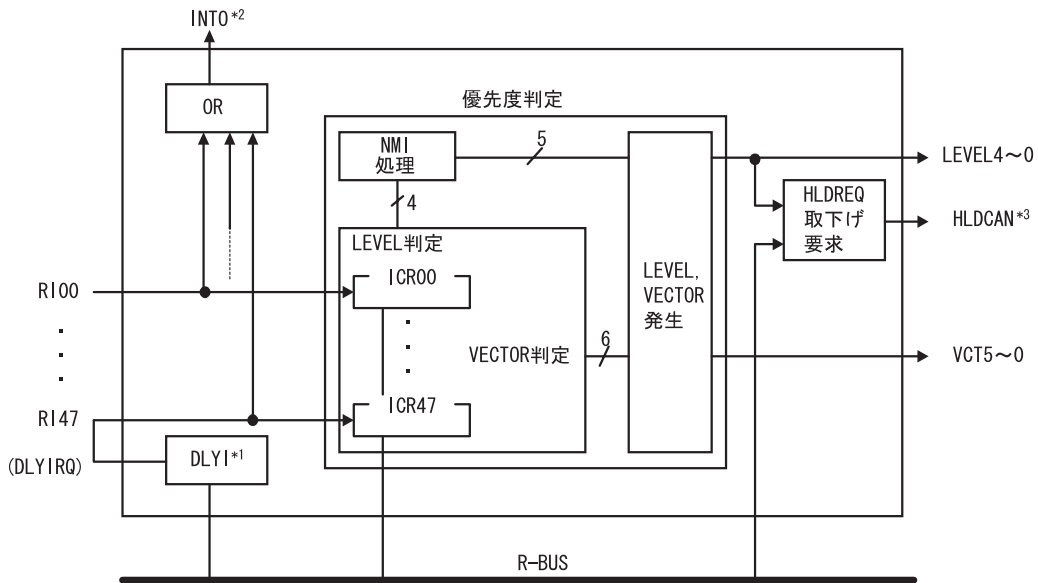
割込みコントローラの主要機能

割込みコントローラには, 主に以下のような機能があります。

- 割込み要求の検出
- 優先度判定 (レベルおよび番号による)
- 判定結果の要因の割込みレベル伝達 (CPUへ)
- 判定結果の要因の割込み番号伝達 (CPUへ)
- 割込み発生によるストップモードからの復帰指示
- バスマスタへのHOLDリクエスト取下げ要求発生

ブロックダイアグラム

図 12.1-1に, 割込みコントローラのブロックダイアグラムを示します。



\*1: DLY1は, 遅延割込みを意味します (詳細は「第11章 遅延割込みモジュール」を参照)。

\*2: INT0は, スリープ, ストップ時のクロック制御部に対するウェイクアップ信号です。

\*3: HLDGANは, CPU以外のバスマスタに対するバス明け渡し要求信号です。

図 12.1-1 割込みコントローラのブロックダイアグラム

## 12.2 割込みコントローラのレジスタ

この節では、割込みコントローラで使用するレジスタの構成および機能について説明します。

割込みコントローラのレジスタ一覧

図 12.2-1に、割込みコントローラのレジスタ一覧を示します。

bit		7	6	5	4	3	2	1	0	レジスタ名
アドレス	00000400 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR00
アドレス	00000401 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR01
アドレス	00000402 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR02
アドレス	00000403 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR03
アドレス	00000404 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR04
アドレス	00000405 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR05
アドレス	00000406 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR06
アドレス	00000407 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR07
アドレス	00000408 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR08
アドレス	00000409 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR09
アドレス	0000040A <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR10
アドレス	0000040B <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR11
アドレス	0000040C <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR12
アドレス	0000040D <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR13
アドレス	0000040E <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR14
アドレス	0000040F <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR15
アドレス	00000410 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR16
アドレス	00000411 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR17
アドレス	00000412 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR18
アドレス	00000413 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR19
アドレス	00000414 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR20
アドレス	00000415 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR21
アドレス	00000416 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR22
アドレス	00000417 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR23
アドレス	00000418 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR24
アドレス	00000419 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR25
アドレス	0000041A <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR26
アドレス	0000041B <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR27
アドレス	0000041C <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR28
アドレス	0000041D <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR29
アドレス	0000041E <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR30
アドレス	0000041F <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR31

図 12.2-1 割込みコントローラのレジスタ一覧(続く)

bit	7	6	5	4	3	2	1	0	レジスタ名
アドレス 00000420 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR32
アドレス 00000421 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR33
アドレス 00000422 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR34
アドレス 00000423 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR35
アドレス 00000424 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR36
アドレス 00000425 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR37
アドレス 00000426 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR38
アドレス 00000427 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR39
アドレス 00000428 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR40
アドレス 00000429 <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR41
アドレス 0000042A <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR42
アドレス 0000042B <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR43
アドレス 0000042C <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR44
アドレス 0000042D <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR45
アドレス 0000042E <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR46
アドレス 0000042F <sub>H</sub>	—	—	—	ICR4	ICR3	ICR2	ICR1	ICR0	ICR47
アドレス 00000431 <sub>H</sub>	—	—	—	LVL4	LVL3	LVL2	LVL1	LVL0	HRCL

図 12.2-1 割込みコントローラのレジスタ一覧(続き)

## 12.2.1 割り込み制御レジスタ(ICR)

割り込み制御レジスタです。各割り込み入力に対して一つずつ設けられており、対応する割り込み要求の割り込みレベルを設定します。

割り込み制御レジスタ(ICR)の構成

図 12.2-2に、割り込み制御レジスタ(ICR: Interrupt Control Register)のビット構成を示します。



図 12.2-2 割り込み制御レジスタ(ICR)のビット構成

以下に、割り込み制御レジスタ(ICR)の各ビットの機能を説明します。

【ビット4～0】 ICR4～0

これらのビットは割り込みレベル設定ビットで、対応する割り込み要求の割り込みレベルを指定します。

このレジスタに設定した割り込みレベルがCPUの割り込みレベルマスクレジスタ(ILM)に設定した(された)レベルマスク値以上の場合は、CPU側で割り込み要求はマスクされます。

リセットにより、11111<sub>B</sub>に初期化されます。

表 12.2-1に、設定可能な割り込みレベル設定ビットと割り込みレベルの対応を示します。

表 12.2-1 設定可能な割り込みレベル設定ビットと割り込みレベルの対応

ICR4	ICR3	ICR2	ICR1	ICR0	割り込みレベル
0	0	0	0	0	0
0	1	1	1	0	14
0	1	1	1	1	15
1	0	0	0	0	16
1	0	0	0	1	17
1	0	0	1	0	18
1	0	0	1	1	19
1	0	1	0	0	20
1	0	1	0	1	21
1	0	1	1	0	22
1	0	1	1	1	23
1	1	0	0	0	24
1	1	0	0	1	25
1	1	0	1	0	26
1	1	0	1	1	27
1	1	1	0	0	28
1	1	1	0	1	29
1	1	1	1	0	30
1	1	1	1	1	31

システム予約

NMI

設定可能な最強レベル  
(強)

(弱)

割り込み禁止

注1: ICR4は"1"固定で,"0"を書き込むことはできません。

# 12.2.2      ホールドリクエスト取下げ要求レベル 設定レジスタ(HRCL)

ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)は、ホールドリクエスト  
取下げ要求発生のためのレベル設定レジスタです。

ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の構成

図 12.2-3に、ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)のビット  
構成を示します。

bit	7	6	5	4	3	2	1	0	初期値
アドレス 00000431 <sub>H</sub>	—	—	—	LVL4	LVL3	LVL2	LVL1	LVL0	---11111 <sub>B</sub>
				R	R/W	R/W	R/W	R/W	

図 12.2-3    ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)のビット構成

以下に、ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の各ビットの  
機能を説明します。

【ビット4～0】 LVL4～0

これらのビットは、バスマスタへのホールドリクエスト取下げ要求を出すた  
めの割込みレベルを設定します。

このレジスタに設定した割込みレベルより高いレベルを持つ割込み要求が発  
生した場合は、バスマスタに対してホールドリクエスト取下げ要求を出します。

LVL4ビットは”1”固定で、”0”を書き込むことはできません。

## 12.3 割込みコントローラの動作

この節では、割込みコントローラの動作として、以下の項目について説明します。

- ・優先順位判定
- ・スタンバイモード(ストップ/スリープ)からの復帰
- ・ホールドリクエスト取下げ要求

### 優先順位判定

割込みコントローラでは、同時に発生している割込み要因の中で最も優先度の高い要因を選択し、その要因の割込みレベルと割込み番号をCPUへ出力します。

割込み要因の優先順位判定基準は、以下のとおりです。

- 以下の条件を満たす要因
  - 割込みレベルの数値が31以外(31は割込み禁止)。
  - 割込みレベルの数値が最も小さい要因。
  - その中で、最も小さい割込み番号を持つ要因。

表 12.3-1に、割込み要因と割込み番号、割込みレベルの関係を示します。

表 12.3-1 割込み要因と割込み番号、割込みレベルの関係(続く)

割込み要因	割込み番号		割込みレベル	オフセット	TBRデフォルトのアドレス
	10進	16進			
NMI要求	15	0F	15(F <sub>H</sub> )固定	3C <sub>H</sub>	000FFFC <sub>H</sub>
外部割込み 0	16	10	ICR00	3B <sub>C</sub> <sub>H</sub>	000FFFB <sub>C</sub> <sub>H</sub>
外部割込み 1	17	11	ICR01	3B <sub>8</sub> <sub>H</sub>	000FFFB <sub>8</sub> <sub>H</sub>
外部割込み 2	18	12	ICR02	3B <sub>4</sub> <sub>H</sub>	000FFFB <sub>4</sub> <sub>H</sub>
外部割込み 3	19	13	ICR03	3B <sub>0</sub> <sub>H</sub>	000FFFB <sub>0</sub> <sub>H</sub>
UART 0 受信完了	20	14	ICR04	3A <sub>C</sub> <sub>H</sub>	000FFFA <sub>C</sub> <sub>H</sub>
UART 1 受信完了	21	15	ICR05	3A <sub>8</sub> <sub>H</sub>	000FFFA <sub>8</sub> <sub>H</sub>
UART 2 受信完了	22	16	ICR06	3A <sub>4</sub> <sub>H</sub>	000FFFA <sub>4</sub> <sub>H</sub>
UART 0 送信完了	23	17	ICR07	3A <sub>0</sub> <sub>H</sub>	000FFFA <sub>0</sub> <sub>H</sub>
UART 1 送信完了	24	18	ICR08	39 <sub>C</sub> <sub>H</sub>	000FFF9 <sub>C</sub> <sub>H</sub>
UART 2 送信完了	25	19	ICR09	39 <sub>8</sub> <sub>H</sub>	000FFF9 <sub>8</sub> <sub>H</sub>
DMAC 0 (終了,エラー)	26	1A	ICR10	39 <sub>4</sub> <sub>H</sub>	000FFF9 <sub>4</sub> <sub>H</sub>
DMAC 1 (終了,エラー)	27	1B	ICR11	39 <sub>0</sub> <sub>H</sub>	000FFF9 <sub>0</sub> <sub>H</sub>
DMAC 2 (終了,エラー)	28	1C	ICR12	38 <sub>C</sub> <sub>H</sub>	000FFF8 <sub>C</sub> <sub>H</sub>
DMAC 3 (終了,エラー)	29	1D	ICR13	38 <sub>8</sub> <sub>H</sub>	000FFF8 <sub>8</sub> <sub>H</sub>
DMAC 4 (終了,エラー)	30	1E	ICR14	38 <sub>4</sub> <sub>H</sub>	000FFF8 <sub>4</sub> <sub>H</sub>
DMAC 5 (終了,エラー)	31	1F	ICR15	38 <sub>0</sub> <sub>H</sub>	000FFF8 <sub>0</sub> <sub>H</sub>
DMAC 6 (終了,エラー)	32	20	ICR16	37 <sub>C</sub> <sub>H</sub>	000FFF7 <sub>C</sub> <sub>H</sub>
DMAC 7 (終了,エラー)	33	21	ICR17	37 <sub>8</sub> <sub>H</sub>	000FFF7 <sub>8</sub> <sub>H</sub>
A/D	34	22	ICR18	37 <sub>4</sub> <sub>H</sub>	000FFF7 <sub>4</sub> <sub>H</sub>
リロードタイマ 0	35	23	ICR19	37 <sub>0</sub> <sub>H</sub>	000FFF7 <sub>0</sub> <sub>H</sub>
リロードタイマ 1	36	24	ICR20	36 <sub>C</sub> <sub>H</sub>	000FFF6 <sub>C</sub> <sub>H</sub>
リロードタイマ 2	37	25	ICR21	36 <sub>8</sub> <sub>H</sub>	000FFF6 <sub>8</sub> <sub>H</sub>
外部割込み 4	38	26	ICR22	36 <sub>4</sub> <sub>H</sub>	000FFF6 <sub>4</sub> <sub>H</sub>
外部割込み 5	39	27	ICR23	36 <sub>0</sub> <sub>H</sub>	000FFF6 <sub>0</sub> <sub>H</sub>
システム予約	40	28	ICR24	35 <sub>C</sub> <sub>H</sub>	000FFF5 <sub>C</sub> <sub>H</sub>
システム予約	41	29	ICR25	35 <sub>8</sub> <sub>H</sub>	000FFF5 <sub>8</sub> <sub>H</sub>

表 12.3-1 割り込み要因と割り込み番号, 割り込みレベルの関係(続き)

割り込み要因	割り込み番号		割り込みレベル	オフセット	TBRデフォルトのアドレス
	10進	16進			
U-TIMER 0	42	2A	ICR26	354 <sub>H</sub>	000FFF54 <sub>H</sub>
U-TIMER 1	43	2B	ICR27	350 <sub>H</sub>	000FFF50 <sub>H</sub>
U-TIMER 2	44	2C	ICR28	34C <sub>H</sub>	000FFF4C <sub>H</sub>
FLASHメモリ	45	2D	ICR29	348 <sub>H</sub>	000FFF48 <sub>H</sub>
システム予約	46	2E	ICR30	344 <sub>H</sub>	000FFF44 <sub>H</sub>
システム予約	47	2F	ICR31	340 <sub>H</sub>	000FFF40 <sub>H</sub>
PPG0	48	30	ICR32	33C <sub>H</sub>	000FFF3C <sub>H</sub>
PPG1	49	31	ICR33	338 <sub>H</sub>	000FFF38 <sub>H</sub>
PPG2	50	32	ICR34	334 <sub>H</sub>	000FFF34 <sub>H</sub>
PPG3	51	33	ICR35	330 <sub>H</sub>	000FFF30 <sub>H</sub>
ICU0(取込み)	52	34	ICR36	32C <sub>H</sub>	000FFF2C <sub>H</sub>
ICU1(取込み)	53	35	ICR37	328 <sub>H</sub>	000FFF28 <sub>H</sub>
ICU2(取込み)	54	36	ICR38	324 <sub>H</sub>	000FFF24 <sub>H</sub>
ICU3(取込み)	55	37	ICR39	320 <sub>H</sub>	000FFF20 <sub>H</sub>
OCU0(一致)	56	38	ICR40	31C <sub>H</sub>	000FFF1C <sub>H</sub>
OCU1(一致)	57	39	ICR41	318 <sub>H</sub>	000FFF18 <sub>H</sub>
OCU2(一致)	58	3A	ICR42	314 <sub>H</sub>	000FFF14 <sub>H</sub>
OCU3(一致)	59	3B	ICR43	310 <sub>H</sub>	000FFF10 <sub>H</sub>
システム予約	60	3C	ICR44	30C <sub>H</sub>	000FFF0C <sub>H</sub>
16ビットフリーランタイム	61	3D	ICR45	308 <sub>H</sub>	000FFF08 <sub>H</sub>
システム予約	62	3E	ICR46	304 <sub>H</sub>	000FFF04 <sub>H</sub>
遅延割り込み要因ビット	63	3F	ICR47	300 <sub>H</sub>	000FFF00 <sub>H</sub>

## 割り込み要因の解除

割り込みルーチンにおける, 割り込み要因解除のための命令とRETI命令の間には制限があります。

詳細は, 「3.9 EIT(例外, 割り込み, トラップ)」を参照してください。

## スタンバイモード(ストップ/スリープ)からの復帰

割り込み要求の発生により, ストップモードから復帰する機能をこのモジュールで実現します。周辺からの割り込み要求が一つでも発生すると, クロック制御部に対してストップモードからの復帰要求を発生します。

優先度判定部は, ストップ復帰後クロックが供給されてから動作を再開しますので, 優先度判定部の結果が出るまでの間, CPUは命令を実行することになります。

スリープ状態からの復帰においても, 同様に動作します。また, スリープ時は, このジュール内のレジスタはDMACなどでアクセス可能です。

## &lt; 注意事項 &gt;

ストップおよびスリープからの復帰の要因としない割り込み要因は, 対応する周辺の制御レジスタにて割り込み要求出力を禁止してください。スタンバイからの復帰要求信号は, 全割り込み要因の単なる論理和出力のため, ICRに設定した割り込みレベルの内容は加味されません。

スリープ時にDMA転送を行いたい場合は, 割り込み要求をこのモジュールへ伝えないようにDMA側で設定し, 誤ってスリープ状態から復帰しないようにしてください。

### ホールドリクエスト取下げ要求(HRCR:Hold Request Cancel Request)

優先度の高い割込処理を、CPUのホールド中に行いたい場合は、ホールドリクエスト発生元においてリクエストを取り下げてもら必要があります。この取下げ要求発生の基準となる割込みレベルをホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)に設定します。

#### 発生基準

ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)に設定した割込みレベルより高いレベルの割込み要因が発生した場合は、ホールドリクエスト取下げ要求を発生します。

- ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の割込みレベル > 優先度判定後の割込みレベル      取下げ要求発生
- ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の割込みレベル      優先度判定後の割込みレベル      取下げ要求なし

取下げ要求発生原因となった割込み要因をクリアしない限り、この取下げ要求は有効であり、結果的にいつまでもDMA転送が起こらないことになります。必ず対応する割込み要因をクリアしてください。

#### 設定可能なレベル

ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)に設定可能な値は、ICRと同様に"10000<sub>6</sub>"から"11111<sub>6</sub>"までです。

"11111<sub>6</sub>"に設定した場合は、すべての割込みレベルに対して取下げ要求を発生します。

表 12.3-2に、ホールドリクエスト取下げ要求発生となる割込みレベルの設定を示します。

表 12.3-2 ホールドリクエスト取下げ要求発生となる割込みレベルの設定

HRCLレジスタ	取下げ要求発生となる割込みレベル
16	
17	割込みレベル16
18	割込みレベル16～17
}	}
31	割込みレベル16～30      【初期値】

リセット後は、すべての割込みレベルに対してホールドリクエスト取下げ状態になります。すなわち、割込みが発生しているとDMA転送が行われませんので、ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の値を必要な値に設定してください。



## 12.4 ホールドリクエスト取下げ要求機能(HRCR)の使用例

DMA転送中に、CPUが優先度の高い処理を行いたいときは、DMAに対してホールドリクエストを取り下げてもらって、ホールド状態を解除する必要があります。ここでは、割込みを利用してDMAに対してホールドリクエストの取下げ、すなわちCPUの優先動作を実現します。

### 制御レジスタ

- ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)  
このレジスタに設定した割込みレベルより高いレベルの割込みが発生した場合に、DMAに対してホールドリクエスト取下げ要求を発生します。その基準となるレベルを設定します。
- 割込み制御レジスタ(ICR)  
使用する割込み要因に対応するICRに、ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)よりも高いレベルを設定します。
- DMA要求抑止レジスタ(PDRR): クロック制御部  
DMAからのホールドリクエストを一時的に抑止するためのレジスタです。割込み要因のクリアにより、再びホールド状態に戻るのを防ぎます。このレジスタの値が0000<sub>h</sub>のときのみ、DMAからのホールドリクエストをCPUに伝えます。使用方法としては、割込みルーチンの先頭でこのレジスタの内容をインクリメントし、ルーチンの出口でデクリメントしてください。

### ハードウェア構成

各信号の流れは、次のようになっています。

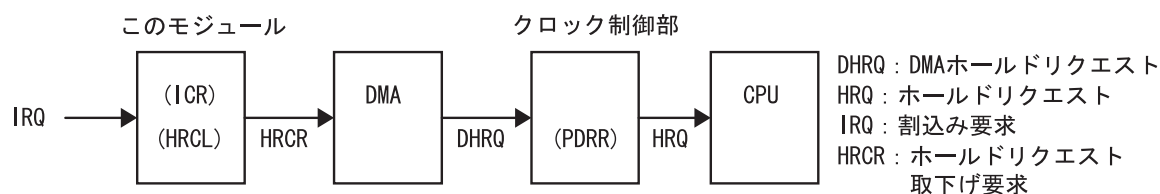


図 12.4-1 ホールドリクエスト取下げ要求機能を使用するためのハードウェア構成例

## ホールドリクエスト取下げ要求シーケンス

## 割込みルーチンの例

図 12.4-2に、ホールドリクエスト取下げ要求シーケンスのタイミング例(割込みレベル:HRCL > a)を示します。

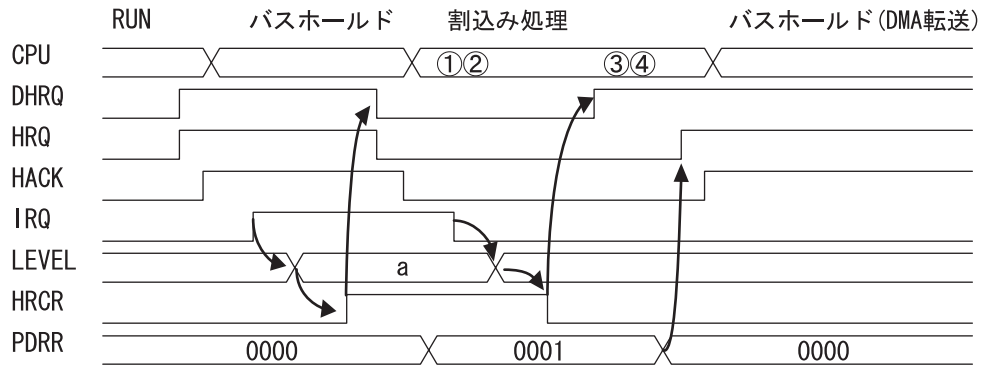


図 12.4-2 ホールドリクエスト取下げ要求シーケンスのタイミング例

PDDRインクリメント  
 割込要因クリア  
 ~  
 PDDRデクリメント  
 RETI

割込み要求が発生すると割込みレベルが変化して、これがホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)に設定したレベルよりも高いとDMAに対してHRCRをアクティブにします。これによってDMAはホールドリクエストを下げ、CPUはホールド状態から復帰して割込み処理を行います。割込みルーチンでは、PDDRをインクリメント( )し、割込み要因をクリア( )します。これにより割込みレベルは変化し、HRCRがインアクティブになりDMAは再びホールドリクエストを出すようになります。しかし、PDDRが0ではないので、このホールドリクエストは遮断されています。PDDRをデクリメント( )することで、初めてホールドリクエストがCPUに伝わり、再びDMA転送が行われるようになります。

## 割り込みルーチンの例

図 12.4-3に、多重割り込みの場合のホールドリクエスト取下げ要求シーケンスのタイミング例(割り込みレベル:HRCL > a > b)を示します。

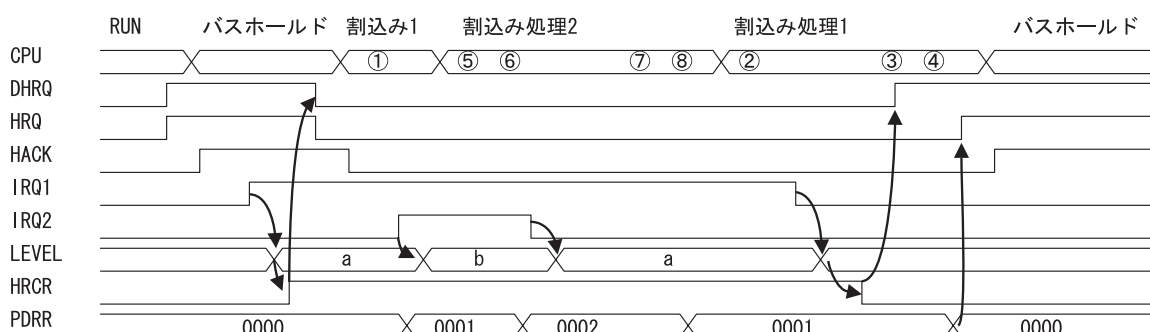


図 12.4-3 ホールドリクエスト取下げ要求シーケンスのタイミング例

- , PRRインクリメント
- , 割込要因クリア
- , PRRデクリメント
- , RETI

図 12.4-3では、割り込みルーチン1を実行中にそれよりも優先度の高い割り込みが発生した場合を示しています。

ここでも、各割り込みルーチンの先頭でPRRをインクリメントし、ルーチンの出口でデクリメントすることによって、不用意にホールドリクエストが上がるのを防ぐことができます。

## &lt; 注意事項 &gt;

- ・ PRRのインクリメント/デクリメントは、DMA転送中(CPUホールド中)に処理させない割り込みルーチンの先頭と出口で必ず行うようにしてください。これを行わないと、割り込みルーチンの途中で再びDMA転送が行われてしまいます。
- ・ 逆にPRRのインクリメント/デクリメントを通常の割り込みルーチンでは行わないでください。割り込みルーチン実行中にDMA転送が行えず、パフォーマンスを落とすことになります。
- ・ ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)とICRに設定する割り込みレベルの関係には、十分注意してください。

## 第13章 A/Dコンバータ(逐次比較型)

---

この章では,A/Dコンバータの概要,レジスタの構成/機能およびA/Dコンバータの動作について説明します。

---

- 13.1 A/Dコンバータ(逐次比較型)の概要
- 13.2 A/Dコンバータのレジスタ
- 13.3 A/Dコンバータの動作
- 13.4 変換データ保護機能
- 13.5 A/Dコンバータ使用上の注意

## 13.1 A/Dコンバータ(逐次比較型)の概要

---

A/Dコンバータは、逐次比較変換方式でアナログ入力電圧をデジタル値に変換するモジュールです。

---

A/Dコンバータの特長

- 最小変換時間: 5.2  $\mu$ s/チャンネル(マシクロック25MHz時)
- サンプル&ホールド回路内蔵
- 分解能: 10ビット
- アナログ入力は4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4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: 1チャンネルを変換したら一時停止して次の起動がかかるまで待機(変換開始の同期が可能)
- 割込みによるDMA転送の起動が可能
- 起動要因は、ソフトウェア、外部トリガ(立下りエッジ)、リロードタイマ(立上りエッジ)から選択

## ブロックダイアグラム

図 13.1-1に、A/Dコンバータのブロックダイアグラムを示します。

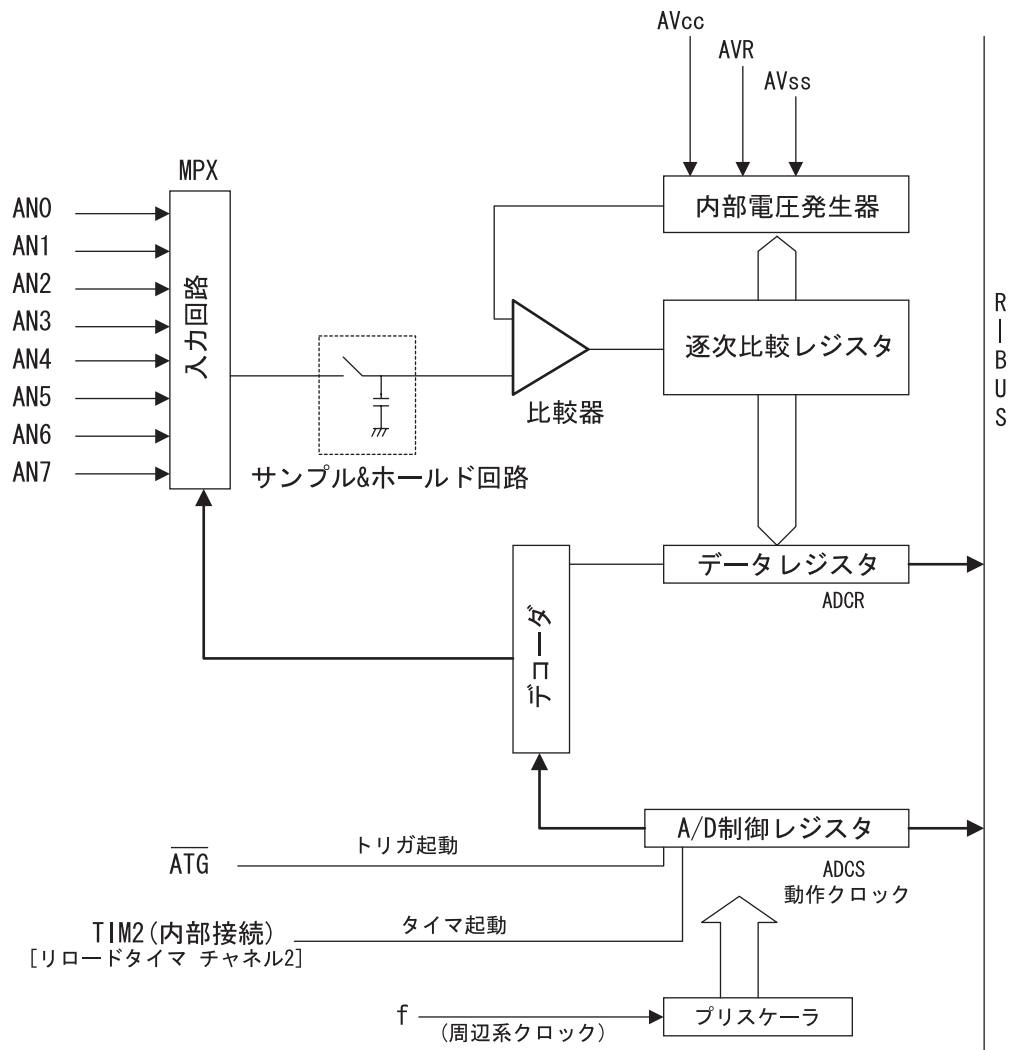


図 13.1-1 A/Dコンバータのブロックダイアグラム

## 13.2 A/Dコンバータのレジスタ

この節では,A/Dコンバータで使用するレジスタの構成および機能について説明します。

### A/Dコンバータのレジスタ

図 13.2-1にA/Dコンバータのレジスタ構成を,図 13.2-2にA/Dコンバータのレジスタ一覧を示します。

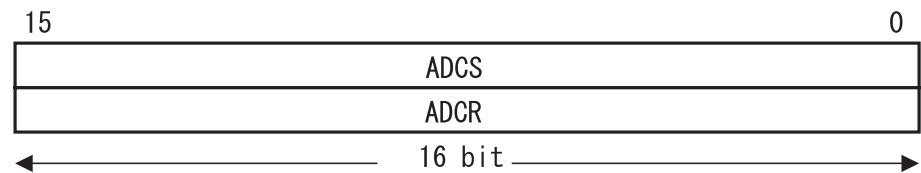


図 13.2-1 A/Dコンバータのレジスタ構成

bit	15	14	13	12	11	10	9	8	コントロールステータス レジスタ (ADCS)
	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	—	
bit	7	6	5	4	3	2	1	0	データレジスタ (ADCR)
	MD1	MD0	ANS2	ANS1	ANS0	ANE2	ANE1	ANE0	
bit	15	14	13	12	11	10	9	8	
	S10	ST1	ST0	CT1	CT0	—	9	8	
bit	7	6	5	4	3	2	1	0	
	7	6	5	4	3	2	1	0	

図 13.2-2 A/Dコンバータのレジスタ一覧

### 13.2.1 コントロールステータスレジスタ(ADCS)

コントロールステータスレジスタ(ADCS)は,A/Dコンバータの制御およびステータスの表示を行います。

コントロールステータスレジスタ(ADCS)のビット構成

図 13.2-3に,コントロールステータスレジスタ(ADCS)のビット構成を示します。

bit	15	14	13	12	11	10	9	8	初期値
アドレス 000 0052 <sub>H</sub>	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	—	00000000 <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
bit	7	6	5	4	3	2	1	0	初期値
	MD1	MD0	ANS2	ANS1	ANS0	ANE2	ANE1	ANE0	00000000 <sub>B</sub>
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 13.2-3 コントロールステータスレジスタ(ADCS)のビット構成

#### < 注意事項 >

コントロールステータスレジスタ(ADCS)は,A/D変換動作中に書き換えしないでください。また,リードモディファイライト系の命令でアクセスしないでください。

コントロールステータスレジスタ(ADCS)のビット詳細

以下に,コントロールステータスレジスタ(ADCS)の各ビットの機能を説明します。

#### 【ビット15】BUSY(BUSY flag and stop)

読出し時と書込み時で,以下のように機能が変わります。

- 読出し時:  
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。
- 書込み時:  
A/D動作中にこのビットに"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。RMW系命令では"1"が読まれます。単発モードではA/D変換終了でクリアされます。連続,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"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)



## 【ビット14】 INT(INTerrupt)

データ表示ビットです。変換データがADCRに書き込まれるとセットされます。

INTE(ビット13)が"1"のときに、このビットがセットされると割込み要求が発生します。また、DMA転送の起動を選択している場合は、DMAが起動されます。"1"書き込みは意味を持ちません。

クリアは"0"書き込みとDMACからのクリア信号で行われます。

## &lt; 注意事項 &gt;

このビットの"0"書き込みによるクリアは、A/D停止中に行ってください。リセット時"0"に初期化されます。リードモディファイライト系命令の読出しでは、"1"が読み出されます。

## 【ビット13】 INTE(INTerrupt Enable)

このビットは、変換終了による割込みの許可/不許可を指定します。

INTE	機能
0	割込みの禁止 (初期値)
1	割込みの許可

割込み要求発生でDMA転送を起動するときは、このビットをセットしてください。リセット時"0"に初期化されます。

## 【ビット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一つのため、連続で変換した場合に、変換結果をDMA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DMA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きます。この間A/D変換動作は停止します。

DMA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DMAを使用したときのみ有効です。

詳細については、「13.4 変換データ保護機能」を参照してください。リセット時"0"に初期化されます。

## 【ビット11,10】 STS1,STS0(STart Source select)

これらのビットは、リセット時"00"に初期化されます。これらのビットの設定により、A/D変換起動要因を選択します。

表 13.2-1に、A/Dコンバータ起動要因の選択を示します。

表 13.2-1 A/Dコンバータ起動要因の選択

STS1	STS0	機能
0	0	ソフトウェア起動
0	1	外部端子トリガでの起動またはソフトウェア起動
1	0	タイマでの起動またはソフトウェア起動
1	1	外部端子トリガ、タイマでの起動またはソフトウェア起動

起動が兼用になっているモードでは、最初に検出した要因で起動します。

起動要因は書換えと同時に変わりますので、A/D変換動作中に書き換えるときは注意してください。

## &lt; 注意事項 &gt;

- ・外部端子トリガは立下がりエッジを検出します。外部トリガ入力レベルが"L"のときにこのビットを書き換えて外部トリガ起動にすると、A/Dコンバータが起動する場合があります。
- ・タイマ選択時は、リロードタイマのチャンネル2が選択されます。リロードタイマの出力レベルが"H"のときにこのビットを書き換えてタイマ起動にするとA/Dコンバータが起動する場合があります。

## 【ビット9】 STRT(STaRT)

このビットに"1"を書き込むことにより,A/Dコンバータを起動します。再起動をかけるときは,再び書き込んでください。停止モード時は,動作機能上再起動はかかりません。

リセット時に"0"に初期化されます。

強制停止とソフトウェア起動を同時に行わないでください。(BUSY=0,STRT=1)

リードモディファイライト系の命令では"0"が読めます。

## 【ビット8】 (Reserved)

このビットは,試験用ビットです。書込み時は"0"を書き込んでください。

## 【ビット7,6】 MD1,MD0 (A/D converter MoDe set)

これらのビットは,動作モードを選択します。

表 13.2-2に,A/Dコンバータ動作モードの選択を示します。

表 13.2-2 A/Dコンバータ動作モードの選択

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,動作中の再起動は不可能
1	0	連続モード,動作中の再起動は不可能
1	1	停止モード,動作中の再起動は不可能

・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,1回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します。

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"00<sub>B</sub>"に初期化されます。

## &lt; 注意事項 &gt;

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,外部トリガおよびソフトウェアのすべての起動に適用されます。

【ビット5,4,3】 ANS2,ANS1,ANS0(ANalog Start channel set)

これらのビットにより,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,これらのビットで選択されたチャンネルからA/D変換を始めます。

表 13.2-3に,A/D変換開始チャンネルの設定を示します。

表 13.2-3 A/D変換開始チャンネルの設定

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

読出し時,これらのビットは,A/D変換中は変換チャンネルが読まれます。

停止モードで停止中は,前の変換チャンネルが読まれます。

リセット時"000<sub>B</sub>"に初期化されます。

【ビット2,1,0】 ANE2,ANE1,ANE0(ANalog End channel set)

これらのビットにより,A/D変換の終了チャンネルを設定します。

表 13.2-4に,A/D変換時の終了チャンネルの設定を示します。

表 13.2-4 A/D変換時の終了チャンネルの設定

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

ANS2～ANS0と同じチャンネルを設定すると,1チャンネル変換になります(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,これらのビット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。

設定チャンネルは,ANS ANEとしてください。

リセット時に"000<sub>B</sub>"に初期化されます。

設定例：

チャンネル設定 ANS=1チャンネル,ANE=3チャンネルで単発モードのときの動作  
変換チャンネル 1チャンネル 2チャンネル 3チャンネル に変わります。

## 13.2.2 データレジスタ(ADCR)

データレジスタ(ADCR)は、変換格納レジスタで、変換結果であるデジタル値が格納されます。A/D変換の分解能を選択する機能もあります。

データレジスタ(ADCR)の構成

図 13.2-4に、データレジスタ(ADCR)のビット構成を示します。

bit	15	14	13	12	11	10	9	8	初期値
アドレス 0000 0050 <sub>H</sub>	S10	ST1	ST0	CT1	CT0	—	9	8	000000XX <sub>B</sub>
	R	R	R	R	R	R	R	R	
bit	7	6	5	4	3	2	1	0	初期値
	7	6	5	4	3	2	1	0	XXXXXXXX <sub>B</sub>
	R	R	R	R	R	R	R	R	

図 13.2-4 データレジスタ(ADCR)のビット構成

データレジスタ(ADCR)の値は、1回の変換終了時ごとに更新されます。通常は、最終変換値が格納されています。

このレジスタは、リセット時は不定です。(上位の15～10ビットを除く)

以下に、データレジスタ(ADCR)の各ビットの機能を説明します。

【ビット15】S10 (Select 10bit or 8bit resolution)

A/D変換の分解能を選択するビットです。

S10	機能
0	10ビット分解能 (D9～D0) (初期値)
1	8ビット分解能 (D7～D0)

なお、分解能によって、使用されるデータビットが異なります。

【ビット14,13】ST1,ST0(Sampling Time)

A/D変換時のサンプリング時間を選択するビットです。

A/Dが起動されると、このビットに設定された時間、アナログ入力を取り込まれます。

表 13.2-5に、A/Dコンバータのサンプリング時間設定を示します。

表 13.2-5 A/Dコンバータのサンプリング時間設定

ST1	ST0	サンプリング時間	
0	0	11 *	1.4 μs @8MHz
0	1	23 *	1.4 μs @16MHz
1	0	33 *	1.3 μs @25MHz
1	1	45 *	

【ビット12,11】CT1,CT0(Conversion Time)

A/D変換時のコンペア時間を選択するビットです。

アナログ入力を取り込まれた(サンプリング時間経過)後,このビットに設定された時間後に,変換結果の

データが確定し,このレジスタのビット9～ビット0に格納されます。

表 13.2-6に,A/Dコンバータのコンペア時間設定を示します。

表 13.2-6 A/Dコンバータのコンペア時間設定

CT1	CT0	コンペア時間	
0	0	34 *	4.3 $\mu$ s @8MHz
0	1	67 *	4.2 $\mu$ s @16MHz
1	0	100 *	4.0 $\mu$ s @25MHz
1	1	122 *	

【ビット10】 空きビット

【ビット9～0】 D9～D0(Date register)

A/D変換の結果が格納され,レジスタは1回の変換終了ごとに書き換えられます。

通常は最終変換値が格納されます。これらのレジスタの初期値は不定です。

変換データ保護機能があります。

A/D変換中にこれらのビットにデータを書き込まないようにしてください。

13.2.3 アナログ入力制御レジスタ(AIC)

アナログ入力制御レジスタ(AIC)は,アナログ入力を選択するときに使用します。

アナログ入力制御レジスタ(AIC)の構成

図 13.2-5に,アナログ入力制御レジスタ(AIC)のビット構成を示します。

	bit	7	6	5	4	3	2	1	0	初期値
アドレス	0000D5 <sub>H</sub>	A17	A16	A15	A14	A13	A12	A11	A10	1111111 <sub>B</sub>
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 13.2-5 アナログ入力制御レジスタ(AIC)のビット構成

以下に,アナログ入力制御レジスタ(AIC)の各ビットの機能を説明します。

【ビット0～7】 A10～A17 (Analog Input control register)

対応するI/Oポートの各端子を以下のように制御します

AI	機能
0	アナログ入力モード
1	ポートモード (初期値)

< 注意事項 >

ポート入力モードに設定しているときに中間電圧レベルが入力されるとリーク電流が流れます。  
したがって,アナログ入力をするときには必ずアナログ入力設定をしてください。

## 13.3 A/Dコンバータの動作

A/Dコンバータは、逐次変換方式で動作し10ビットの分解能をもっています。

このA/Dコンバータは、変換結果格納用のレジスタが一つ(16ビット)しかないため、1回の変換終了とともに変換データレジスタ(ADCR)が更新されてしまいます。

連続で変換をする際は、DMA転送を使用することが可能です。

A/Dコンバータには、単発変換モード、連続変換モードおよび停止変換モードの3種類のモードがあります。各モードでの動作を説明します。

### 単発変換モード

単発変換モードは、ADCSの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コンバータ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、1チャンネル変換動作になります。

例：

- ANS=000<sub>B</sub>, ANE=011<sub>B</sub>の場合  
開始    AN0    AN1    AN2    AN3    終了
- ANS=010<sub>B</sub>, ANE=010<sub>B</sub>の場合  
開始    AN2    終了

### 連続変換モード

連続変換モードは、ADCSの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、A/D変換動作を続けます。

開始チャンネルと終了チャンネルが同じとき(ANS=ANE)は、1チャンネル変換を続けます。

例：

- ANS=000<sub>B</sub>, ANE=011<sub>B</sub>の場合  
開始    AN0    AN1    AN2    AN3    AN0    繰り返し
- ANS=010<sub>B</sub>, ANE=010<sub>B</sub>の場合  
開始    AN2    AN2    AN2    繰り返し

連続変換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"を書き込むと、強制動作停止します。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。

### 停止変換モード

停止変換モードは、ADCSの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ことに変換動作を一時停止します。一時停止を解除するには、もう一度起動す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、A/D変換動作を続けます。

開始チャンネルと終了チャンネルが同じ場合(ANS=ANE)は、1チャンネルを変換します。

例:

- ANS=000<sub>B</sub>, ANE=011<sub>B</sub>の場合  

開始	AN0	停止	起動	AN1	停止	起動	AN2	停止
	起動	AN3	停止	起動	AN0		繰り返し	
- ANS=010<sub>B</sub>, ANE=010<sub>B</sub>の場合  

開始	AN2	停止	起動	AN2	停止	起動	AN2
		繰り返し					

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、変換開始の同期をかけることが可能です。



## 13.4 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、DMAC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。

---

### 変換データ保護機能

変換データレジスタは一つですので、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、このA/Dコンバータは変換が終了しても前データがDMAC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。一時停止の解除は、DMA転送によってメモリへ転送された後に行われます。

前データが転送されていれば、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ADCSのINT, INTEビットに関係しています。

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、この機能は動作せず、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。

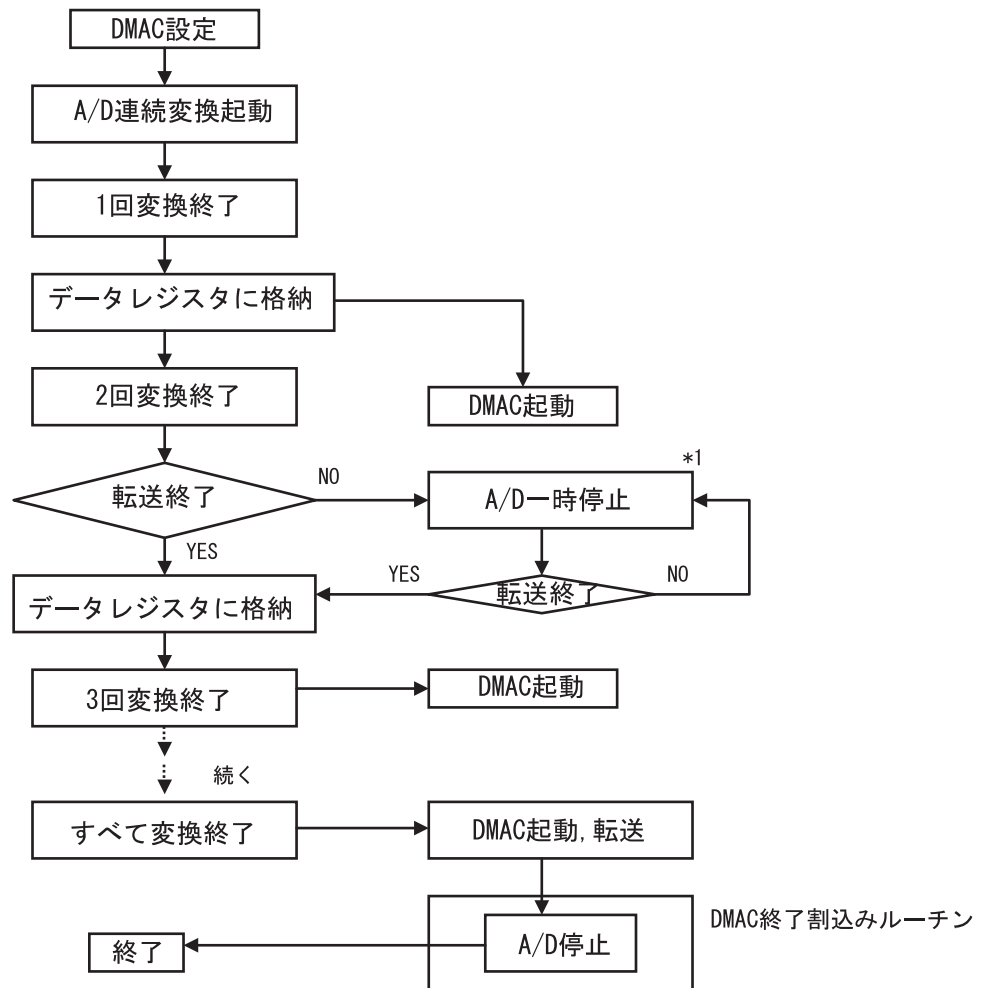
また、割込み許可 (INTE=1) 状態でDMA転送を使わない場合は、INTビットはクリアされないため、データ保護機能が働き、A/Dは変換を一時停止状態にします。この場合は、割込みシーケンスでINTビットをクリアすると停止状態が解除されます。

DMA動作中で、A/Dが一時停止をしているときに割込みを禁止すると、A/Dが動作し、転送する前に変換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。

---

図 13.4-1に、DMA転送使用時のデータ保護機能フローを示します。



\*1:一時停止中に再起動をかけると待機中の変換データは壊れてしまいます。

図 13.4-1 DMA転送使用時のデータ保護機能フロー

## 13.5 A/Dコンバータ使用上の注意

---

A/Dコンバータを使用するときの使用上の注意を示します。

---

### A/Dコンバータ使用上の注意

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

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, コントロールステータスレジスタ(ADCS)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にしてください。アクティブ側にしておくと誤動作します。

STS1, STS0設定時は,  $\overline{ATG}=1$ 入力, リロードタイマ(チャネル2)=0出力の状態にしてください。

### A/Dコンバータのその他の注意事項

外部インピーダンスが指定値以上に高くなると, 規定のサンプリング時間内にアナログ入力値をサンプリングできなくなり, 正しい変換結果が得られません。

## 第14章 UART

---

この章では, UARTの概要, レジスタの構成/機能およびUARTの動作について説明します。

---

- 14.1 UARTの概要
- 14.2 UARTのレジスタ
- 14.3 UARTの動作
- 14.4 UARTの使用例
- 14.5 ボーレートとU-TIMERのリロード値の設定例

## 14.1 UARTの概要

---

UART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。

MB91F127/128は、UARTを3チャンネル内蔵しています。

---

### UARTの特長

UARTは、以下の特長をもっています。

- 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
- 完全プログラマブルモード
  - 内蔵タイマにより任意のボーレートを設定可能(「9.3 U-TIMERの動作」参照)
- 外部クロックによる自由なボーレートの設定が可能
- 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
- 割込みによるDMA転送の起動が可能

ブロックダイアグラム

図 14.1-1に, UARTのブロックダイアグラムを示します。

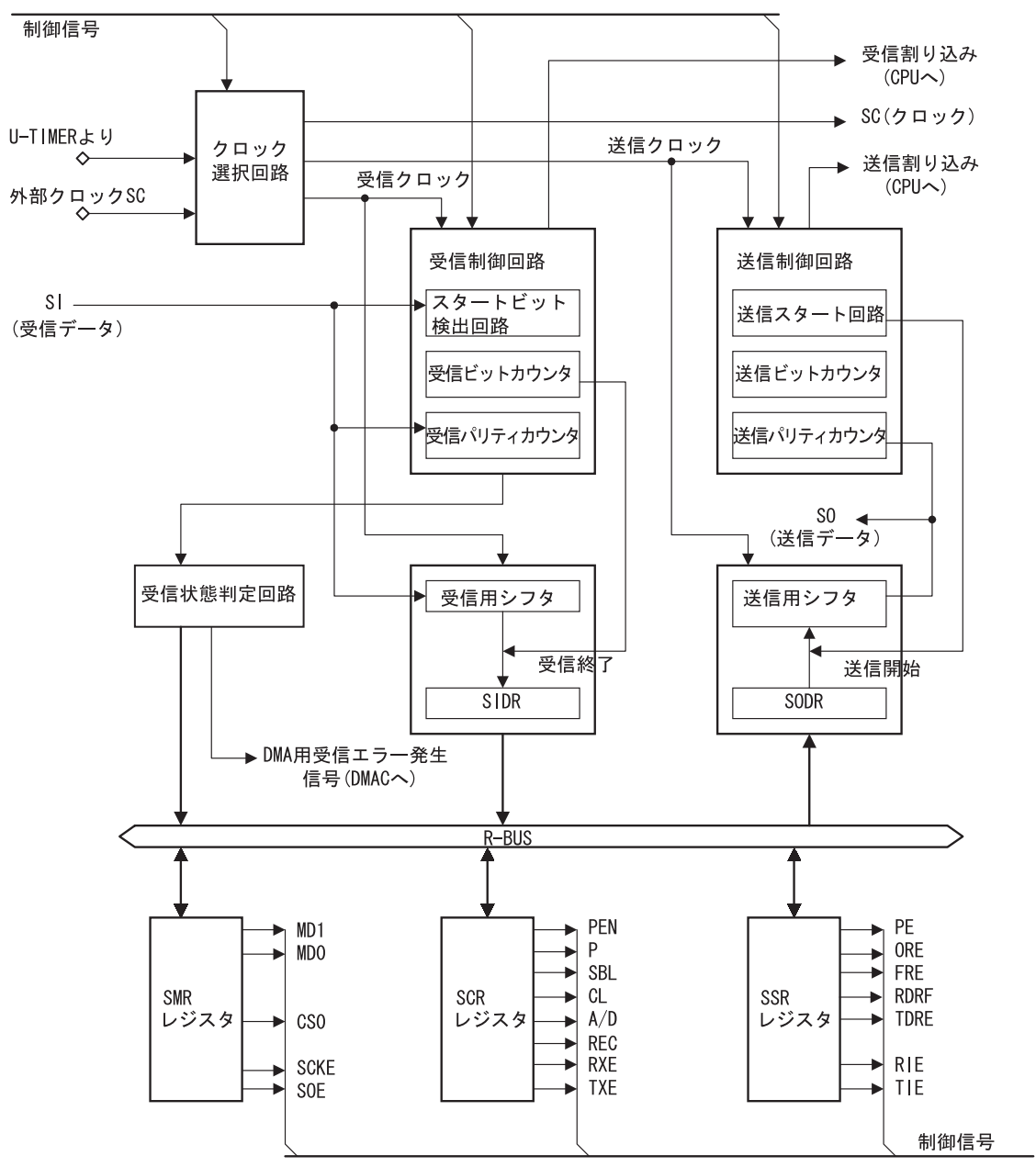


図 14.1-1 UARTのブロックダイアグラム

# 14.2 UARTのレジスタ

この節では, UARTで使用するレジスタの構成および機能について説明します。

UARTのレジスタ

図 14.2-1に, UARTのレジスタ構成を, 図 14.2-2に, UARTのレジスタ一覧を示します。

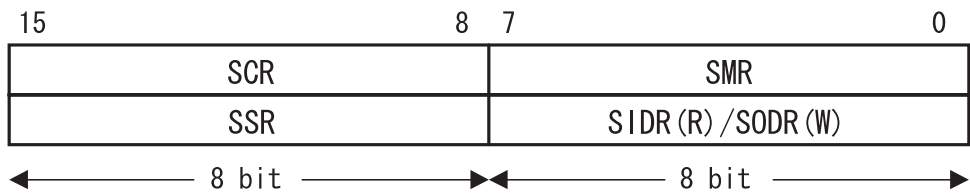


図 14.2-1 UARTのレジスタ構成

bit	7	6	5	4	3	2	1	0	
	D7	D6	D5	D4	D3	D2	D1	D0	シリアルインプットデータレジスタ シリアルアウトプットデータレジスタ (SIDR/SODR)

bit	15	14	13	12	11	10	9	8	
	PE	ORE	FRE	RDRF	TDRE	—	RIE	TIE	シリアルステータスレジスタ (SSR)

bit	7	6	5	4	3	2	1	0	
	MD1	MD0	—	—	CS0	—	SCKE	SOE	シリアルモードレジスタ (SMR)

bit	15	14	13	12	11	10	9	8	
	PEN	P	SBL	CL	A/D	REC	RXE	TXE	シリアルコントロールレジスタ (SCR)

図 14.2-2 UARTのレジスタ一覧

14.2.1 シリアルモードレジスタ(SMR)

シリアルモードレジスタ(SMR)は,UART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ます。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(SMR)のビット構成

図 14.2-3に,シリアルモードレジスタ(SMR)のビット構成を示します。

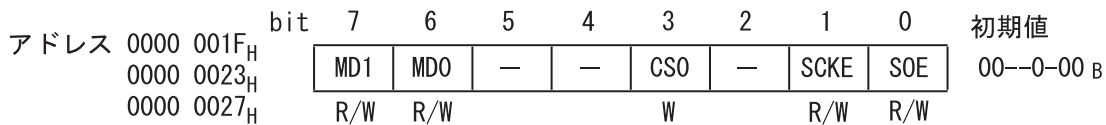


図 14.2-3 シリアルモードレジスタ(SMR)のビット構成

シリアルモードレジスタ(SMR)のビット詳細

以下に,シリアルモードレジスタ(SMR)の各ビットの機能を説明します。

【ビット7,6】MD1,MD0(MoDe select)

これらのビットは,UARTの動作モードを選択します。  
表 14.2-1に,UARTの動作モード選択を示します。

表 14.2-1 UARTの動作モード選択

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 (初期値)
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。このリソースでは,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,パリティチェック機能は使用できませんので,シリアルコントロールレジスタ(SCR)のPENIは”0”に設定してください。

【ビット5,4】(reserved)

これらのビットは,未使用ビットです。  
常に”1”を書き込んでください。

【ビット3】CS0(Clock Select)

このビットは,UARTの動作クロックを選択します。

CS0	機能
0	内蔵タイマ(U-TIMER) (初期値)
1	外部クロック

【ビット2】(reserved)

このビットは,未使用ビットです。  
常に”0”を書き込んでください。



【ビット1】SCKE(SCLK Enable)

このビットは,CLK同期モード(モード2)で通信をする場合,SC端子をクロック入力端子にするか,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。

SCKE	機能
0	クロック入力端子として機能します。(初期値)
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,CS0ビットを”1”にして外部クロックを選択しておく必要があります。

【ビット0】SOE (Serial Output Enable)

このビットは,汎用I/Oポート端子と兼用されている外部端子(S0)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。

SOE	機能
0	汎用I/Oポート端子として機能します。(初期値)
1	シリアルデータ出力端子(S0)として機能します。

## 14.2.2 シリアルコントロールレジスタ(SCR)

シリアルコントロールレジスタ(SCR)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。

シリアルコントロールレジスタ(SCR)のビット構成

図 14.2-4に、シリアルコントロールレジスタ(SCR)のビット構成を示します。

アドレス	bit	15	14	13	12	11	10	9	8	初期値
0000 001E <sub>H</sub>		PEN	P	SBL	CL	A/D	REC	RXE	TXE	00000100 <sub>B</sub>
0000 0022 <sub>H</sub>										
0000 0026 <sub>H</sub>		R/W	R/W	R/W	R/W	R/W	W	R/W	R/W	

図 14.2-4 シリアルコントロールレジスタ(SCR)のビット構成

シリアルコントロールレジスタ(SCR)のビット詳細

以下に、シリアルコントロールレジスタ(SCR)の各ビットの機能を説明します。

### 【ビット7】PEN(Parity Enable)

このビットは、シリアルデータに対して、パリティビットの付加(送信時)および検出(受信時)をするかどうかを選択します。

PEN	機能
0	パリティなし (初期値)
1	パリティあり

### < 注意事項 >

動作モード1,2を選択した場合は、パリティは使用できません。

このビットは常に”0”設定してください。

### 【ビット6】P(Parity)

このビットは、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。

P	機能
0	偶数パリティ (初期値)
1	奇数パリティ

### 【ビット5】SBL(Stop Bit Length)

このビットは、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。

SBL	機能
0	1ストップビット (初期値)
1	2ストップビット

## 【ビット4】CL(Character Length)

このビットは、送受信する1フレームのデータ長を指定します。

CL	機能
0	7ビットデータ (初期値)
1	8ビットデータ

## &lt; 注意事項 &gt;

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ビットデータとしてください。

## 【ビット3】A/D(Address/Data)

このビットは、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、送受信するフレームのデータ形式を指定します。

A/D	機能
0	データフレーム (初期値)
1	アドレスフレーム

## 【ビット2】REC(Receiver Error Clear)

このビットに、"0"を書き込むことによって、シリアルステータスレジスタ(SSR)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。

## 【ビット1】RXE(Receiver Enable)

このビットは、UARTの受信動作を制御します。

RXE	機能
0	受信動作を禁止します。 (初期値)
1	受信動作を許可します。

## &lt; 注意事項 &gt;

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/シリアルインプットデータレジスタ(SIDR)に受信データをストアしたところで受信動作を停止します。

## 【ビット0】TXE(Transmitter Enable)

このビットは、UARTの送信動作を制御します。

TXE	機能
0	送信動作を禁止します。 (初期値)
1	送信動作を許可します。

## &lt; 注意事項 &gt;

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(SODR)にデータがなくなった後に送信動作を停止します。"0"の書込みは、SODRにデータを書き込んだ後は、一定の期間を置いて行ってください。一定の期間は、クロック非同期転送モード時は、ボーレートの1/16時間です。クロック同期転送モード時は、ボーレートの時間です。

### 14.2.3 シリアルインプットデータレジスタ(SIDR)/ シリアルアウトプットデータレジスタ(SODR)

これらのレジスタは、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(SIDR)/シリアルアウトプットデータレジスタ(SODR)の構成

図 14.2-5に、シリアルインプットデータレジスタ(SIDR)およびシリアルアウトプットデータレジスタ(SODR)のビット構成を示します。

SIDR	bit	7	6	5	4	3	2	1	0	初期値
アドレス 0000 001D <sub>H</sub>										不定
0000 0021 <sub>H</sub>		D7	D6	D5	D4	D3	D2	D1	D0	
0000 0025 <sub>H</sub>		R	R	R	R	R	R	R	R	
SODR	bit	7	6	5	4	3	2	1	0	初期値
アドレス 同上										不定
		D7	D6	D5	D4	D3	D2	D1	D0	
		W	W	W	W	W	W	W	W	

図 14.2-5 シリアルインプットデータレジスタ(SIDR)/  
シリアルアウトプットデータレジスタ(SODR)のビット構成

データ長が7ビットの場合、SIDR、SODRのビット7(D7)は無効データとなります。シリアルアウトプットデータレジスタ(SODR)への書込みは、シリアルステータスレジスタ(SSR)のTDREビットが“1”のときに書き込んでください。

#### < 注意事項 >

このアドレスへの書込みはシリアルアウトプットデータレジスタ(SODR)への書込みを、読出しはシリアルインプットデータレジスタ(SIDR)からの読出しを意味します。

# 14.2.4 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(SSR)は,UART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(SSR)のビット構成

図 14.2-6に,シリアルステータスレジスタ(SSR)のビット構成を示します。

アドレス	0000 001C <sub>H</sub>	0000 0020 <sub>H</sub>	0000 0024 <sub>H</sub>	bit 15	14	13	12	11	10	9	8	初期値
				PE	ORE	FRE	RDRF	TDRE	—	RIE	TIE	00001-00 B
				R	R	R	R	R		R/W	R/W	

図 14.2-6 シリアルステータスレジスタ(SSR)のビット構成

シリアルステータスレジスタ(SSR)のビット詳細

以下に,シリアルステータスレジスタ(SSR)の各ビットの機能を説明します。

## 【ビット7】PE(Parity Error)

このビットは,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,シリアルコントロールレジスタ(SCR)のRECビット(ビット10)に"0"を書き込みます。

このビットがセットされた場合には,SIDRのデータは無効データとなります。

PE	機能
0	パリティエラーなし (初期値)
1	パリティエラーが発生

## 【ビット6】ORE(Over Run Error)

このビットは,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,シリアルコントロールレジスタ(SCR)のRECビットに"0"を書き込みます。

このビットがセットされた場合には,SIDRのデータは無効データとなります。

ORE	機能
0	オーバランエラーなし (初期値)
1	オーバランエラーが発生

## 【ビット5】FRE(FRaming Error):フレーミングエラーの有無

このビットは,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,シリアルコントロールレジスタ(SCR)のRECビットに"0"を書き込みます。

このビットがセットされた場合には,SIDRのデータは無効データとなります。

FRE	機能
0	フレーミングエラーなし (初期値)
1	フレーミングエラーが発生

## 【ビット4】RDRF(Receiver Data Register Full)

このビットは、シリアルインプットデータスレジスタ(SIDR)に受信データがあることを示す、割込み要求フラグです。

シリアルインプットデータスレジスタ(SIDR)に受信データがロードされるとセットされ、シリアルインプットデータスレジスタ(SIDR)を読み出すと自動的にクリアされます。

RDRF	機能
0	受信データなし (初期値)
1	受信データあり

## 【ビット3】TDRE(Transmitter Data Register Empty)

このビットは、SODR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。

シリアルアウトプットデータスレジスタ(SODR)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むことができることを表します。

TDRE	機能
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 (初期値)

## 【ビット2】(reserved)

このビットは、未使用ビットです。

## 【ビット1】RIE(Receiver Interrupt Enable)

このビットは、受信割込みを制御します。

RIE	機能
0	受信割込みを禁止します。 (初期値)
1	受信割込みを許可します。

## &lt; 注意事項 &gt;

受信割込み要因は、PE, ORE, FREによるエラー発生のほか、RDRFによる正常受信があります。

## 【ビット0】TIE(Transmitter Interrupt Enable)

このビットは、送信割込みを制御します。

TIE	機能
0	送信割込みを禁止します。 (初期値)
1	送信割込みを許可します。

## &lt; 注意事項 &gt;

送信割込み要因は、TDREによる送信要求があります。

# 14.3 UARTの動作

UARTは以下の三つの動作モードを持ち、シリアルモードレジスタ(SMR)、シリアルコントロールレジスタ(SCR)に値を設定することにより、モードを切り替えることができます。

- ・ 非同期(調歩同期) ノーマルモード
- ・ 非同期(調歩同期) マルチプロセッサモード
- ・ CLK同期モード

## UARTの動作モード

表 14.3-1に、UARTの動作モードを示します。

非同期(調歩同期)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。下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。

表 14.3-1 UART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	あり/なし	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
	あり/なし	8		
1	なし	8 + 1	非同期(調歩同期) マルチプロセッサモード	
2	なし	8	CLK同期モード	なし

## UARTのクロック選択

### 内部タイマ

CS0を”0”に設定してU-TIMERを選択した場合は、U-TIMERに設定したリロード値でボーレートが決まります。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) :  $f / (16 \times \text{リロード値})$
- CLK同期 :  $f / \text{リロード値}$   
: 周辺系マシンのクロック周波数(CLKP)  
: U-TIMERで設定した周期(2n+2または2n+3 nはリロード値)

非同期(調歩同期)モードのボーレートは、設定したボーレートの-1% ~ +1%までの範囲で転送が可能です。

### 外部クロック

CS0を”1”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 :  $f$   
ただし、fは最大3.125MHzまでです。

### 14.3.1 非同期(調歩同期)モード

UARTは, NRZ(Non Return to Zero)形式のデータのみを扱います。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の転送が行われ, 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。

#### 非同期(調歩同期)モードの転送データフォーマット

図 14.3-1に, 非同期(調歩同期)モード転送データフォーマットを示します。

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。

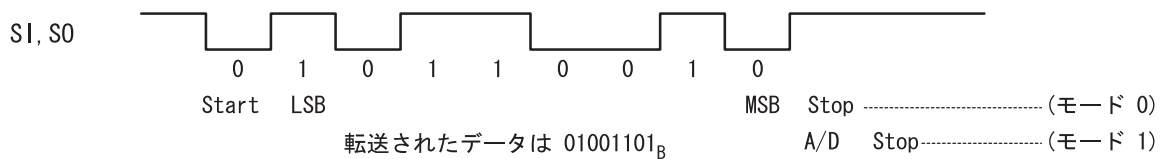


図 14.3-1 非同期(調歩同期)モードの転送データフォーマットモード0,1)

#### 受信動作

シリアルコントロールレジスタ(SCR)のRXEビット(ビット1)が"1"ならば, 常に受信動作が行われています。受信ラインにスタートビットが現れると, シリアルコントロールレジスタ(SCR)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ってから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, RDRFフラグ(シリアルステータスレジスタ(SSR)のビット4)がセットされます。このとき同じシリアルステータスレジスタ(SSR)のRIEビット(ビット1)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。シリアルステータスレジスタ(SSR)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(SIDR)を読み出して, エラーが発生していれば必要な処理を行うようにしてください。

RDRFフラグは, シリアルインプットデータレジスタ(SIDR)を読み出すとクリアされます。

#### 送信動作

シリアルステータスレジスタ(SSR)のTDREフラグ(ビット11)が"1"のとき, シリアルアウトプットデータレジスタ(SODR)に送信データを書き込みます。ここで, シリアルコントロールレジスタ(SCR)のTXEビット(ビット0)が"1"なら送信が行われます。

シリアルアウトプットデータレジスタ(SODR)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。このとき, 同じシリアルステータスレジスタ(SSR)のTIEビット(ビット0)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, シリアルアウトプットデータレジスタ(SODR)に送信データをセットするように要求します。

TDREフラグは, シリアルアウトプットデータレジスタ(SODR)にデータをセットすると一度クリアされます。



## 14.3.2 CLK同期モード

UART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 14.3-2に, CLK同期モードの送受信クロックとデータとの関係を示します。

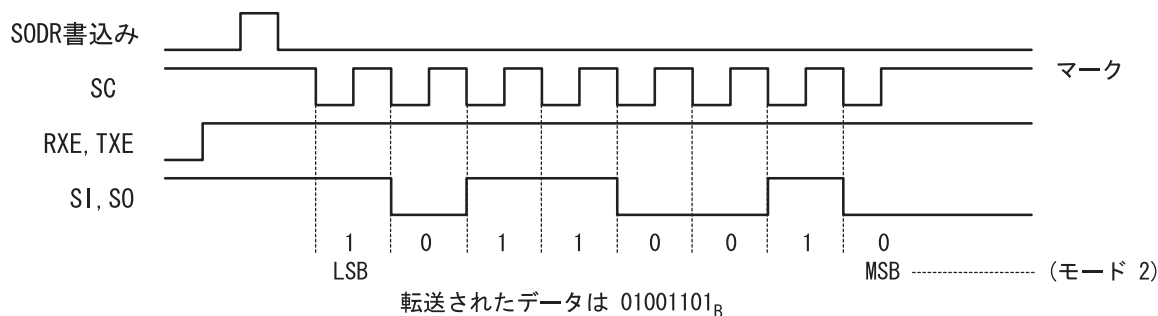


図 14.3-2 CLK同期モードの転送データフォーマット(モード2)

CS0を"0"に設定し, U-TIMERからの出力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/シリアルアウトデータレジスタ(SODR)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。

データ長は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。

### 初期化

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。

- シリアルモードレジスタ(SMR)
  - MD1, MD0: 10
  - CS : クロック入力を指定
  - SCKE : 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
- シリアルコントロールレジスタ(SCR)
  - 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
- シリアルステータスレジスタ(SSR)
  - 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

#### 通信開始

シリアルアウトプットデータレジスタ(SODR)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シリアルアウトプットデータレジスタ(SODR)に書き込む必要があります。

#### 通信終了

シリアルステータスレジスタ(SSR)のRDRFフラグが"1"に変化したことにより確認できます。シリアルステータスレジスタ(SSR)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。

### 14.3.3 UARTの割込み発生とフラグのセットタイミング

UARTには、五つのフラグと二つの割込み要因があります。

五つのフラグは、PE/ORE/FRE/RDRF/TDREです。

二つの割込み要因は、受信用のものと送信用のものです。

#### 割込み発生とフラグ

PEはパリティエラー、OREはオーバランエラー、FREはフレーミングエラーのことで、受信時エラーが発生したときにセットされ、シリアルコントロールレジスタ(SCR)のRECに"0"を書き込むとクリアされます。

RDRFは、受信データがシリアルインプットデータレジスタ(SIDR)にロードされるとセットされ、シリアルインプットデータレジスタ(SIDR)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、シリアルアウトプットデータレジスタ(SODR)が空になって書き込み可能な状態になるとセットされ、シリアルアウトプットデータレジスタ(SODR)へ書き込むとクリアされます。

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。

#### モード0の受信動作時割込みフラグのセットタイミング

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がアクティブ時は、SIDRのデータは無効データとなります。

図 14.3-3に、モード0におけるORE、FRE、RDRFのセットタイミングを示します。

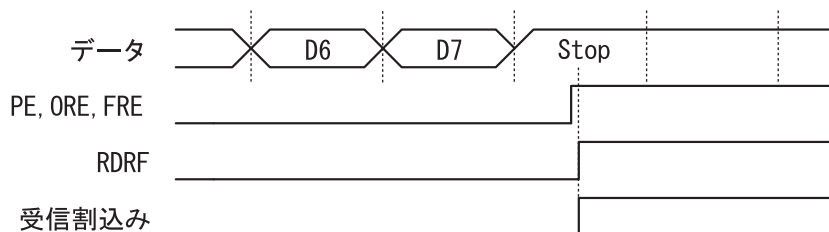


図 14.3-3 ORE、FRE、RDRFのセットタイミング(モード0)

#### モード1の受信動作時割込みフラグのセットタイミング

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、SIDRのデータは無効データとなります。

図 14.3-4に、モード1におけるORE, FRE, RDRFのセットタイミングを示します。

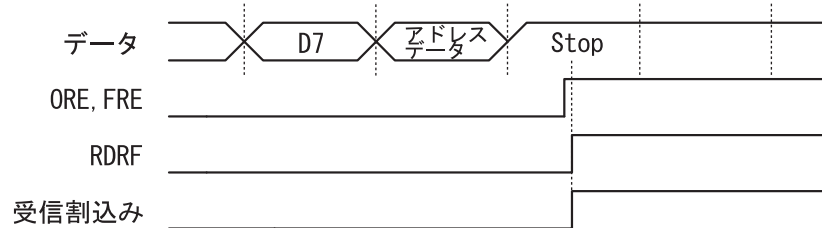


図 14.3-4 ORE, FRE, RDRFのセットタイミング(モード1)

#### モード2の受信動作時割込みフラグのセットタイミング

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、SIDRのデータは無効データとなります。

図 14.3-5に、モード2におけるORE, RDRFのセットタイミングを示します。

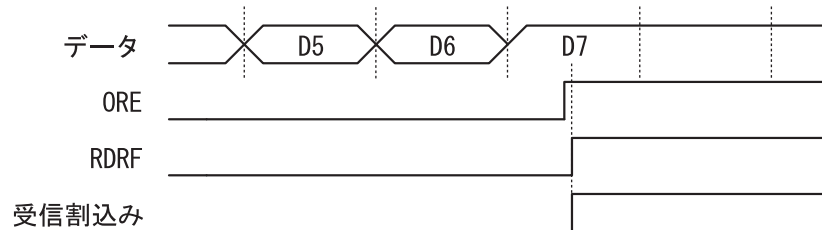


図 14.3-5 ORE, RDRFのセットタイミング(モード2)

## モード0,モード1,モード2の送信動作時割込みフラグのセットタイミング

TDREはシリアルアウトプットデータレジスタ(SODR)へ書き込まれるとクリアされ、内部のシフトレジスタに転送されます。その後、次のデータ書込みが可能な状態になるとセットされ、CPUへの割込み要求が発生します。送信動作中にシリアルコントロールレジスタ(SCR)のTXEに"0"(モード2のときはRXEも含む)を書き込むと、シリアルステータスレジスタ(SSR)のTDREが"1"となり、送信用のシフタが停止してからUARTの送信動作を禁止します。送信動作中にシリアルコントロールレジスタ(SCR)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シリアルアウトプットデータレジスタ(SODR)へ書き込まれたデータは送信されます。

図 14.3-6にモード0,1におけるTDREのセットタイミングを、図 14.3-7にモード2におけるTDREのセットタイミングを示します。

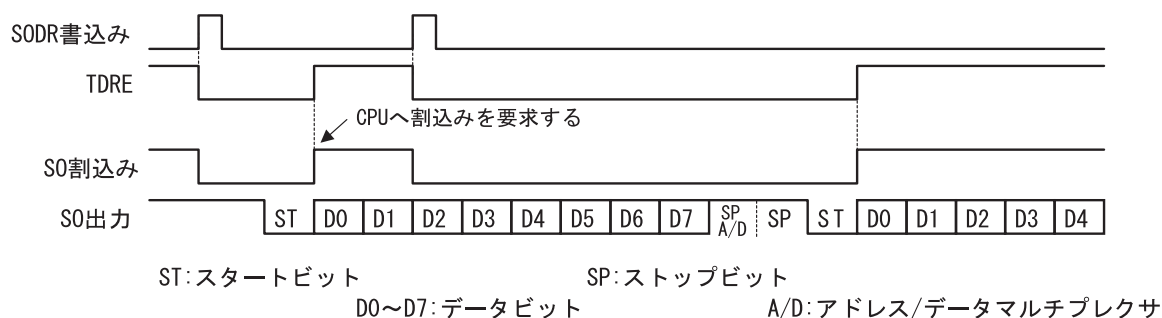


図 14.3-6 TDREのセットタイミング(モード0,1)

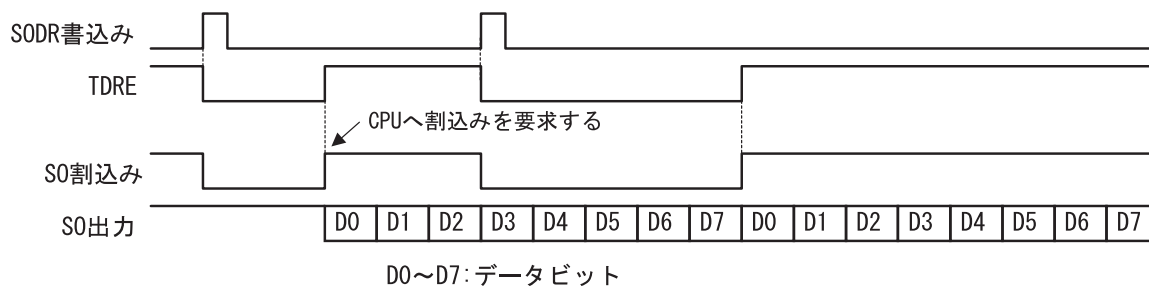


図 14.3-7 TDREのセットタイミング(モード2)

## 14.4 UARTの使用例

UARTの使用上の注意と使用例を示します。

### 使用上の注意

通信モードの設定は動作停止中に行ってください。モード設定時に送受信したデータは保証されません。同期転送モード(モード2)で使用しているときに、送信データレジスタ(SODR)への書込みが受信割込みの要求(RDRF=1)に相当するタイミングと一致した場合、通信制御回路が停止する場合がありますので、データ転送が完全に終わってからSODRに書き込むか、送信開始してからすぐにSODRへの書き込みをするようにしてください。

### UARTの使用例

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。

図 14.4-1に、モード1を使用時のシステム構築例を示します。このリソースでは、ホスト側の通信インタフェースのみサポートしています。

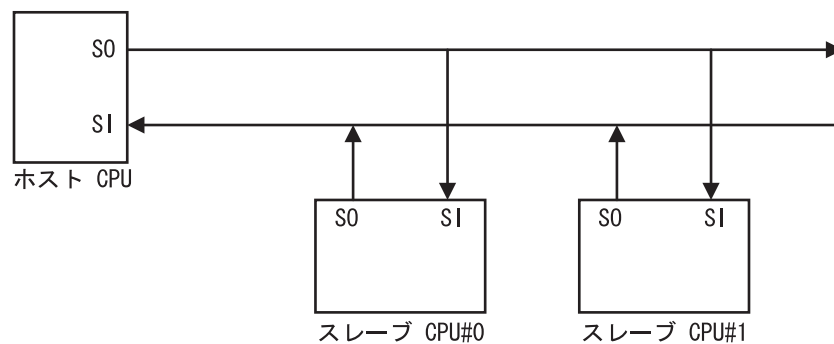


図 14.4-1 モード1を使用時のシステム構築例

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、シリアルコントロールレジスタ(SCR)のA/Dビットが”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、シリアルコントロールレジスタ(SCR)のA/Dビットが”0”のときのデータです。

図 14.4-2に、その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、シリアルコントロールレジスタ(SCR)のPENビットは”0”としてください。

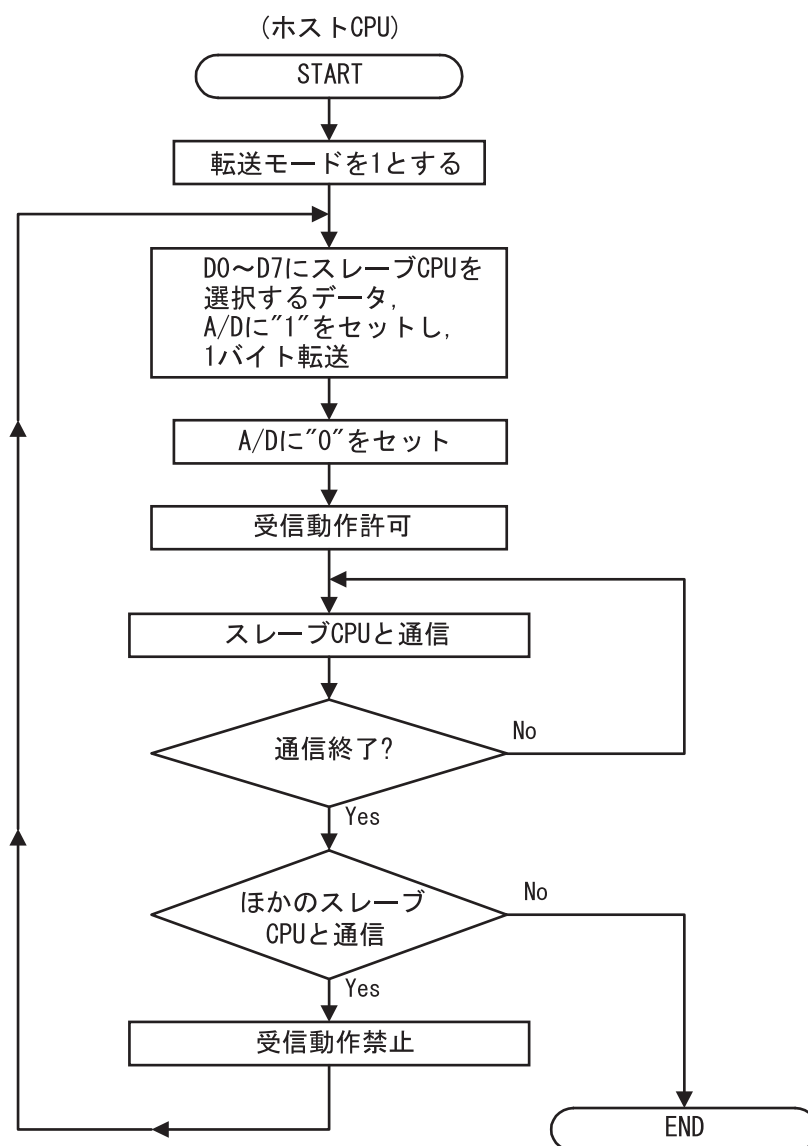


図 14.4-2 モード1を使用時の通信フローチャート

## 14.5 ボーレートとU-TIMERのリロード値の設定例

ボーレートとU-TIMERのリロード値の設定例を示します。

### ボーレートとU-TIMERのリロード値の設定例

表 14.5-1に、非同期(調歩同期)モードの設定値を、表 14.5-2にCLK同期モードの設定値を示します。

表中の周波数は、周辺系マシナクロック周波数を表します。また、UCC1は、U-TIMERのU-TIMERコントロールレジスタ(UTIMC)のUCC1ビットに設定する値です。表中の「-」部は、誤差が±1%を越えてしまうため使用できないことを示します。

表 14.5-1 非同期(調歩同期)モードの設定値

ボーレート	μs	25MHz	20MHz	12.5MHz	10MHz
1200	833.33	650 D (UCC1=0)	520 D (UCC1=0)	324 D (UCC1=1)	259 D (UCC1=1)
2400	416.67	324 D (UCC1=1)	259 D (UCC1=1)	162 D (UCC1=0)	129 D (UCC1=0)
4800	208.33	162 D (UCC1=0)	129 D (UCC1=0)	80 D (UCC1=1)	64 D (UCC1=0)
9600	104.17	80 D (UCC1=1)	64 D (UCC1=0)	39 D (UCC1=1)	31 D (UCC1=1)
19200	52.08	39 D (UCC1=1)	31 D (UCC1=1)	19 D (UCC1=1)	-
38400	26.04	19 D (UCC1=1)	-	12 D (UCC1=1)	-
57600	17.36	12 D (UCC1=1)	-	-	-
10400	96.15	74 D (UCC1=0)	59 D (UCC1=0)	36 D (UCC1=1)	29 D (UCC1=0)
31250	32.00	24 D (UCC1=0)	19 D (UCC1=0)	11 D (UCC1=1)	9 D (UCC1=0)
62500	16.00	11 D (UCC1=1)	9 D (UCC1=0)	-	4 D (UCC1=0)

表 14.5-2 CLK同期モードの設定値

ボーレート	μs	25MHz	20MHz	12.5MHz	10MHz
250K	4.00	49 D (UCC1=0)	39 D (UCC1=0)	24 D (UCC1=0)	19 D (UCC1=0)
500K	2.00	24 D (UCC1=0)	19 D (UCC1=0)	11 D (UCC1=1)	9 D (UCC1=0)
1M	1.00	11 D (UCC1=1)	9 D (UCC1=0)	5 D (UCC1=0)*1	4 D (UCC1=0)

\*1: ±1%以上の誤差あり





## 第15章 DMAコントローラ

---

この章では,DMAコントローラの概要,レジスタの構成/機能およびDMAコントローラの動作について説明します。

---

- 15.1 DMAコントローラの概要
- 15.2 DMAコントローラのブロックダイアグラム
- 15.3 DMAコントローラのレジスタ
- 15.4 DMAコントローラの転送モード
- 15.5 DMAコントローラのタイミング図
- 15.6 DMAコントローラに関する注意事項

# 15.1 DMAコントローラの概要

DMAコントローラは,DMA(Direct Memory Access)転送を行います。

DMAコントローラの特長

- 8チャンネル
- モード:シングル/ブロック転送,バースト転送,連続転送の3種
- アドレス全領域とアドレス全領域の間での転送
- 最大65536回の転送回数
- 転送終了時割込み機能
- 転送アドレス増加/減少をソフトウェアで選択可能

DMA転送要求要因一覧表

表 15.1-1に,DMA転送要求要因一覧を示します。

表 15.1-1 DMA転送要求要因一覧

チャンネル番号	転送要求要因
0	なし
1	なし
2	なし
3	PPG ch0
4	UART ch0 受信
5	UART ch0 送信
6	16ビットリロードタイマ ch0
7	A/Dコンバータ

## 15.2 DMAコントローラのブロックダイアグラム

図 15.2-1にDMAコントローラのブロックダイアグラムを示します。

DMAコントローラのブロックダイアグラム

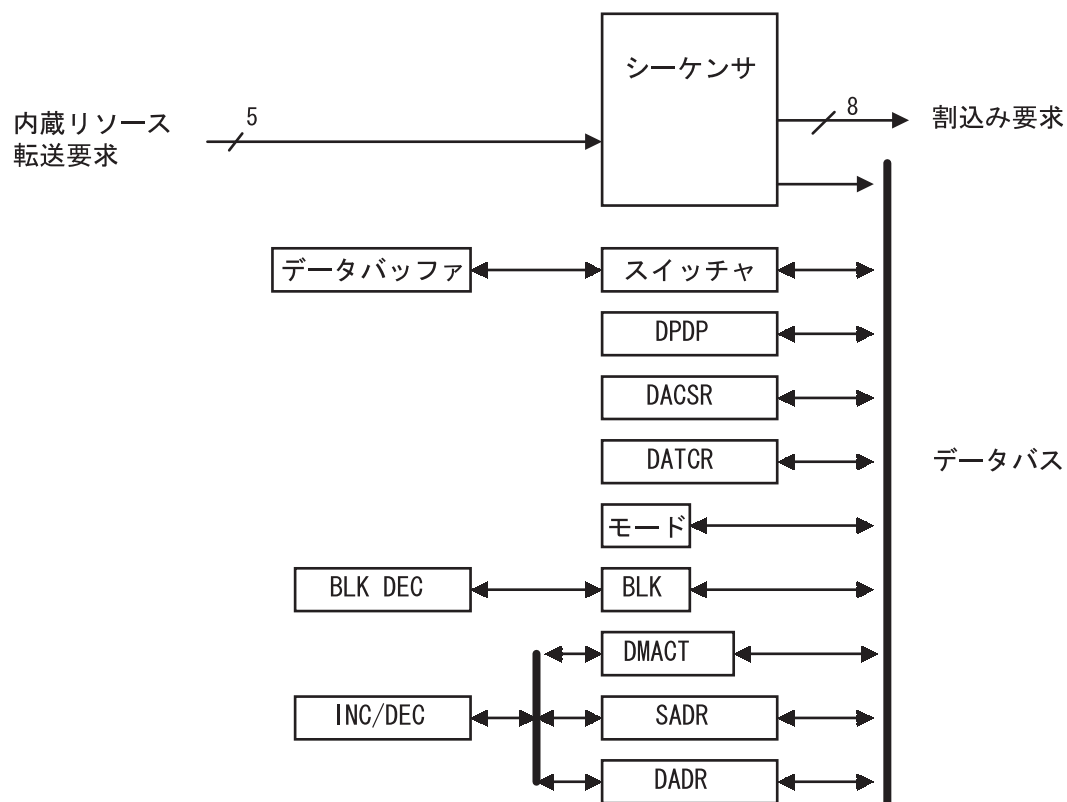


図 15.2-1 DMAコントローラのブロックダイアグラム

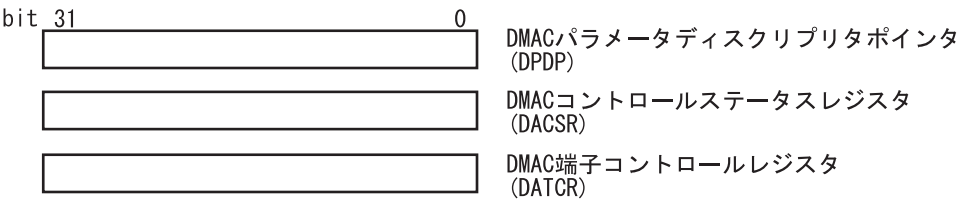
# 15.3 DMAコントローラのレジスタ

図 15.3-1にDMAコントローラのレジスタ一覧を示します。

DMAコントローラのレジスタ一覧

図 15.3-1に,DMAコントローラのレジスタ一覧を示します。

[DMAC内部レジスタ]



[RAM上のDMAディスクリプタ]

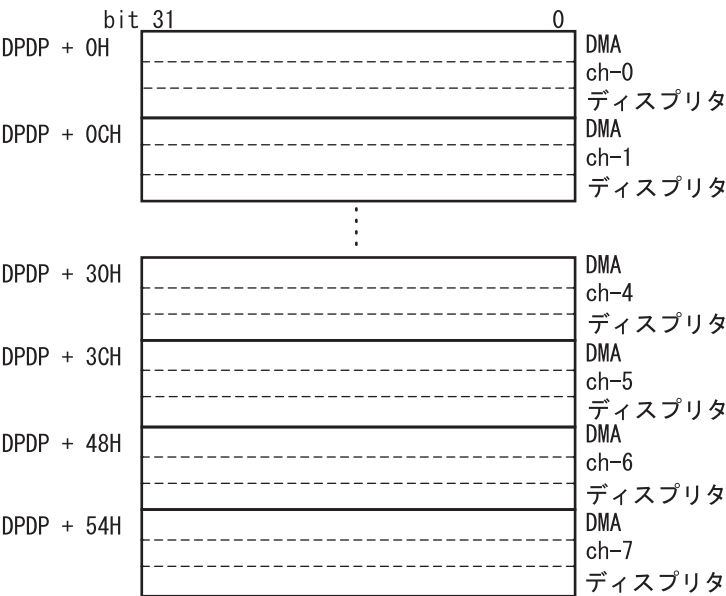


図 15.3-1 DMAコントローラのレジスタ一覧

### 15.3.1 DMACパラメータディスクリプタポインタ(DPDP)

DMACパラメータディスクリプタポインタ(DPDP)は,DMACの内部レジスタで,RAM上にあるDMAC用ディスクリプタテーブルの先頭アドレスを格納します。

DPDPのビット6～ビット0は常に0で,設定可能なディスクリプタの先頭アドレスは128バイト単位となります。

DMACパラメータディスクリプタポインタ(DPDP)の構成

図 15.3-2に,DMACパラメータディスクリプタポインタ(DPDP)のビット構成を示します。

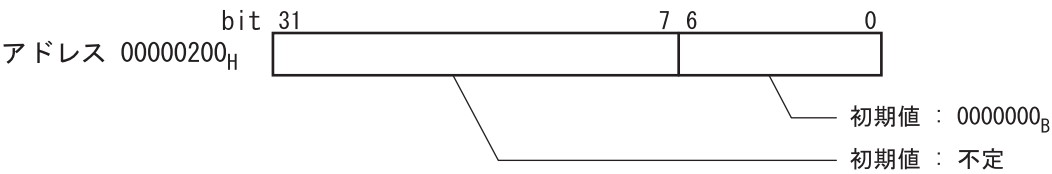


図 15.3-2 DMACパラメータディスクリプタポインタ(DPDP)のビット構成

リセット時は,初期化されません。読出し/書込み可能です。

このレジスタへのアクセスには,32ビット転送命令を使用してください。

各チャンネルの動作モードを指定するディスクリプタは,DPDPで指定される表15.3-1に示すアドレスに置かれます。

表 15.3-1 各チャンネルのディスクリプタアドレス

DMAチャンネル	ディスクリプタアドレス	DMAチャンネル	ディスクリプタアドレス
0	DPDP + 0 (00H)	4	DPDP + 48 (30H)
1	DPDP + 12 (0CH)	5	DPDP + 60 (3CH)
2	DPDP + 24 (18H)	6	DPDP + 72 (48H)
3	DPDP + 36 (24H)	7	DPDP + 84 (54H)

### 15.3.2 DMACコントロールステータスレジスタ(DACSR)

DMACコントロールステータスレジスタ(DACSR)は,DMACの内部レジスタで,DMAC全体に関する制御,ステータスを示します。

DMACコントロールステータスレジスタ(DACSR)のビット構成

図 15.3-3に,DMACコントロールステータスレジスタ(DACSR)のビット構成を示します。

bit	31	30	29	28	27	26	25	24	初期値
	DER7	DED7	DIE7	DOE7	DER6	DED6	DIE6	DOE6	
アドレス 00000204 <sub>H</sub>	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	00000000H
bit	23	22	21	20	19	18	17	16	
	DER5	DED5	DIE5	DOE5	DER4	DED4	DIE4	DOE4	
bit	15	14	13	12	11	10	9	8	
	DER3	DED3	DIE3	DOE3	DER2	DED2	DIE2	DOE2	
bit	7	6	5	4	3	2	1	0	
	DER1	DED1	DIE1	DOE1	DER0	DED0	DIE0	DOE0	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

図 15.3-3 DMACコントロールステータスレジスタ(DACSR)のビット構成

DMACコントロールステータスレジスタ(DACSR)のビット詳細

以下に,DMACコントロールステータスレジスタ(DACSR)の各ビットの機能を説明します。

【ビット31,27,23,19,15,11,7,3】 DERn(DMA ERror)

これらのビットは,チャンネルnのDMA要求発生元にエラーが発生して,DMA転送処理を中断したことを示します。

DER	機能
0	エラーはありません。
1	エラーが発生しました。

エラーが発生するか否かは,DMA要求発生元(リソース)により異なります。  
エラーが発生しないDMA要求発生元もあります。  
リセット時は,"0"に初期化されます。  
読出し/書込み可能ですが,これらのビットへの書込みは,"0"のみ有効です。  
リードモディファイライト系の命令による読出し値は,"1"が読み出されます。

## 【ビット30,26,22,18,14,10,6,2】 DEDn(DMA EnD)

これらのビットは、チャンネルnのDMA転送が終了したことを示します。

DED	機能
0	DMA転送動作は終了していません。
1	カウンタが0になったまたは転送要求発生元でエラーが発生したことを示します。

リセット時は,"0"に初期化されます。

読出し/書込み可能ですが、これらのビットへの書込みは,"0"のみ有効です。

リードモディファイライト系の命令による読出し時は,"1"が読み出されます。

## 【ビット29,25,21,17,13,9,5,1】 DIEn(DMA Interrupt Enable)

これらのビットは、チャンネルnのDMA転送終了時(DEDnが1になったとき)に割り込み要求を発生させるか否かを指定します。

DIE	機能
0	割り込み禁止
1	割り込み許可

リセット時は,"0"に初期化されます。読出し/書込み可能です。

## 【ビット28,24,20,16,12,8,4,0】 D0En(DMA Operation Enable)

これらのビットは、チャンネルnのDMA転送動作を許可します。

DOE	機能
0	動作禁止
1	動作許可

該当するチャンネルのDMA転送が完了すると(DEDnが1になると),D0Enは0にクリアされます。

転送の完了によるクリア動作と、バスからの書込みによるセット動作が同時に行われた場合は、セット動作が優先されます。

リセット時は,"0"に初期化されます。読出し/書込み可能です。



### 15.3.3 DMAC端子コントロールレジスタ(DATCR)

DMAC端子コントロールレジスタ(DATCR)は,DMACの内部レジスタで外部転送要求入力端子,外部転送要求受出力端子,外部転送終了出力端子に関する制御を行います。

DMAC端子コントロールレジスタ(DATCR)のビット構成

図 15.3-4に,DMAC端子コントロールレジスタ(DATCR)のビット構成を示します。

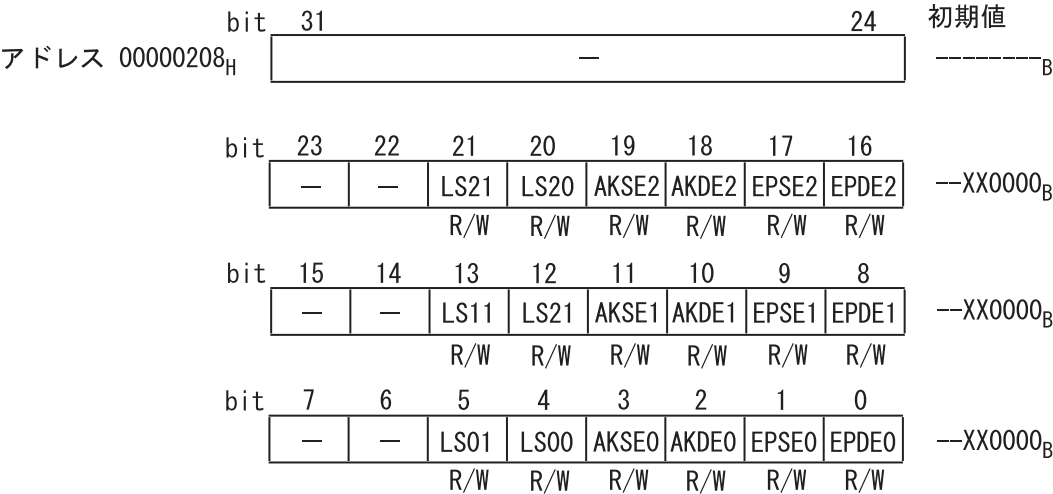


図 15.3-4 DMAC端子コントロールレジスタ(DATCR)のビット構成

DMAC端子コントロールレジスタ(DATCR)のビット詳細

以下に,DMAC端子コントロールレジスタ(DATCR)の各ビットの機能を説明します。

【ビット21,20,13,12,5,4】 LSn1,LSn0:転送要求入力検出レベル選択

これらのビットは,該当する外部転送要求入力端子DREQnの検出レベルを表15.3-2のように選択します。

表 15.3-2 転送要求入力検出レベル

LSn1	LSn0	動作制御機能
0	0	立上がりエッジ検出
0	1	立下がりエッジ検出
1	0	“H”レベル検出
1	1	“L”レベル検出

リセット時は,不定です。読み出し/書き込み可能です。  
連続転送モードを使用するときは,“H”レベル検出または“L”レベル検出を設定してください。

【ビット19,11,3】 AKSEn

【ビット18,10,2】 AKDn

これらのビットは、転送要求受出力信号を発生するタイミングを指定します。また、転送要求受出力信号の端子からの出力機能の許可/禁止も指定します。表 15.3-3に、転送要求受出力の指定を示します。

表 15.3-3 転送要求受出力の指定

AKSEn	AKDn	動作制御機能
0	0	転送受出力禁止
0	1	転送受出力許可, 転送先データアクセス時に出力
1	0	転送受出力許可, 転送元データアクセス時に出力
1	1	転送受出力許可 転送元および転送先データアクセス時に出力

リセット時は,"00"に初期化されます。読出し/書込み可能です。

【ビット17,9,1】 EPSEn

【ビット16,8,0】 EPDn

これらのビットは、転送終了出力信号を発生するタイミングを指定します。また、転送終了出力信号の端子からの出力機能の許可/禁止も指定します。表 15.3-4に、転送終了出力の指定を示します。

表 15.3-4 転送終了出力の指定

EPSEn	EPDn	動作制御機能
0	0	転送終了出力禁止
0	1	転送終了出力許可, 転送先データアクセス時に出力
1	0	転送終了出力許可, 転送元データアクセス時に出力
1	1	転送終了出力許可 転送元および転送先データアクセス時に出力

リセット時は,"00"に初期化されます。読出し/書込み可能です。

### 15.3.4 RAM上のディスクリプタ内レジスタ

DMA転送における、チャンネルごとの設定情報を格納します。チャンネル当たり12バイトの大きさを持ち、DPDPにより示されるアドレスのメモリが使用されます。

チャンネルごとのディスクリプタの先頭アドレスは、「表 15.3-1 各チャンネルのディスクリプタアドレス」を参照してください。

ディスクリプタ先頭ワードの構成

図 15.3-5に、ディスクリプタ先頭ワードのビット構成を示します。

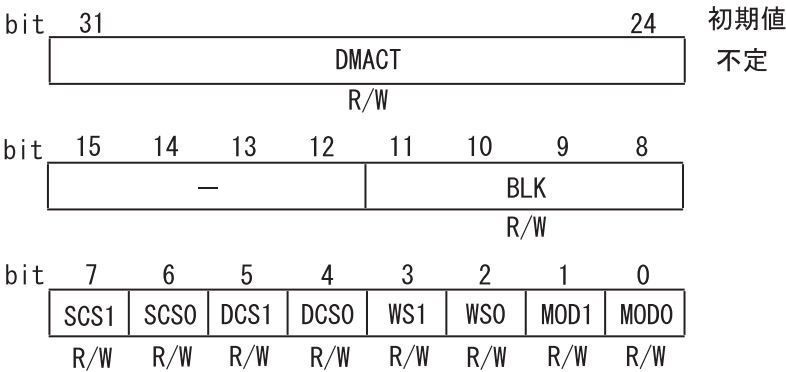


図 15.3-5 ディスクリプタ先頭ワードのビット構成

以下に、ディスクリプタ先頭ワードの各ビットの機能を説明します。

【ビット31～16】 DMACT: 転送回数指定

これらのビットは、DMA転送を行う回数を指定します。0000<sub>h</sub>を設定すると、65536回の転送を行います。1回転送を行うたびに、値が1ずつ減少します。

【ビット15～12】 空き

【ビット11～8】 BLK: ブロックサイズ指定

シングル/ブロック転送モードにおける、転送ブロックサイズを指定します。

“0”を設定すると、ブロックサイズとしては16が指定されたことになります。シングル転送を行うときは、“1”を指定してください。

【ビット7,6】 SCS1, SCS0: 転送元アドレス更新モード指定

## 【ビット5,4】 DCS1,DCS0:転送先アドレス更新モード指定

これらのビットは、転送元および転送先アドレスの1転送ごとの更新モードを指定します。表 15.3-5の組合せの指定が可能です。

表 15.3-5 転送元/転送先アドレス更新モードの指定

SCS1	SCS0	DCS1	DCS0	転送元アドレス	転送先アドレス
0	0	0	0	アドレス増加	アドレス増加
0	0	0	1	アドレス増加	アドレス減少
0	0	1	0	アドレス増加	アドレス固定
0	1	0	0	アドレス減少	アドレス増加
0	1	0	1	アドレス減少	アドレス減少
0	1	1	0	アドレス減少	アドレス固定
1	0	0	0	アドレス固定	アドレス増加
1	0	0	1	アドレス固定	アドレス減少
1	0	1	0	アドレス固定	アドレス固定
その他				設定禁止	

アドレスを更新するモードにおける増減の単位は、転送データサイズの指定に応じて、表 15.3-6のようになります。

表 15.3-6 アドレス増減の単位

転送データサイズ	アドレス増減の単位
バイト(8ビット)	±1バイト
ハーフワード(16ビット)	±2バイト
ワード(32ビット)	±4バイト

## 【ビット3,2】 WS1,WS0

これらのビットは、転送データサイズを指定します。  
表 15.3-7に、転送データサイズの指定を示します。

表 15.3-7 転送データサイズの指定

WS1	WS0	転送データサイズ
0	0	バイト
0	1	ハーフワード
1	0	ワード
1	1	設定禁止

## 【ビット1,0】 MOD1,MOD0:転送モード指定

これらのビットは、転送モードを指定します。  
表 15.3-8に、転送モードの指定を示します。

表 15.3-8 転送モードの指定

MOD1	MOD0	動作モード
0	0	シングル/ブロックモード
0	1	バーストモード
1	0	連続転送モード
1	1	設定禁止

連続転送モードが使用できるのは、ch0～ch2だけです。

### ディスクリプタ第2ワードの構成

図 15.3-6に,ディスクリプタ第2ワードのビット構成を示します。



図 15.3-6 ディスクリプタ第2ワードのビット構成

転送元アドレスを格納します。

アドレス更新モードの指定(SCS1,SCS0ビット)に基づき,転送動作に応じて値が更新されます。

転送データサイズがハーフワード長のときは2の倍数のアドレスを,ワード長のときは4の倍数のアドレスを指定してください。

### ディスクリプタ第3ワードの構成

図 15.3-7に,ディスクリプタ第3ワードのビット構成を示します。



図 15.3-7 ディスクリプタ第3ワードのビット構成

転送先アドレスを格納します。

アドレス更新モードの指定(DCS1,DCS0ビット)に基づき,転送動作に応じて値が更新されます。

転送データサイズがハーフワード長のときは2の倍数のアドレスを,ワード長のときは4の倍数のアドレスを指定してください。

## 15.4 DMAコントローラの転送モード

DMAコントローラには、次の三つの転送モードがあり、動作手順は以下のとおりです。

- ・シングル/ブロック転送モード
- ・連続転送モード
- ・バースト転送モード

### シングル/ブロック転送モード

- 1) 初期化ルーチンでディスクリプタの設定を行います。
- 2) プログラムにより、DMA転送要求発生元の初期化を行います。内部周辺回路を転送要求発生元とする場合は、割り込み要求を許可にします。同時に、割り込みコントローラのICRは割り込み禁止にします。
- 3) プログラムにより、所望するDACSRのDOEnビットに"1"を書き込みます。  
---この時点でDMAに関する設定は完了---
- 4) DMACがDMA転送要求入力を検出すると、CPUにバス権確保を要求します。
- 5) CPUがバス権を移譲すると、DMACはディスクリプタにある3ワードの情報をバスを経由してアクセスします。
- 6) DMACTの減算を行い、ディスクリプタ内の情報に従った転送をBLKで指定された回数またはDMACTが"0"になるまで行います。データの転送中、転送要求受出力信号を出力します(外部転送要求入力使用の場合)。減算したDMACTが"0"になったとき、データ転送中に転送終了出力信号を出力する。
- 7) 転送要求入力をクリアする。
- 8) SADRまたはDADRの増加減を行い、DMACTの値とともにディスクリプタに書き戻します。
- 9) CPUにバス権を返します。
- 10) DMACTの値が"0"であった場合、DACSRのDEDnを"1"にするとともに割り込み許可になっていればCPUに割り込みをかけます。

1回あたりの最低所要サイクル数は、ディスクリプタを内蔵RAMに格納し、データを外部バス間で転送、データ長がバイトの場合、以下ようになります。

- 転送元/転送先アドレスともに固定のとき:  $(6+5 \times \text{BLK})$  サイクル
- 転送元/転送先アドレスの一方のみ固定のとき:  $(7+5 \times \text{BLK})$  サイクル
- 転送元/転送先アドレスの両方が増加または減少のとき:  $(8+5 \times \text{BLK})$  サイクル

### 連続転送モード

- 1) 初期化ルーチンでディスクリプタの設定を行います。
- 2) プログラムにより,DMA転送要求発生元の初期化を行います。外部転送要求入力端子を,HまたはLレベル検出に設定します。
- 3) プログラムにより,所望するDACSRのD0Enビットに1を書き込みます。  
---この時点でDMAに関する設定は完了---
- 4) DMACがDMA転送要求入力を検出すると,CPUにバス権確保を要求します。
- 5) CPUがバス権を移譲すると,DMACはディスクリプタにある3ワードの情報をバスを経由してアクセスします。
- 6) DMACTの減算を行い,ディスクリプタ内の情報に従った転送を1回行います。  
データの転送中,転送要求受出力信号を出力します。減算したDMACTが"0"になったとき,データ転送中に転送終了出力信号を出力します。
- 7) DMACTの値が"0"ではなく,周辺からのDMA要求がまだ存在すれば,ふたたび6)から繰り返します(バスの状態によっては8)を経由します)。
- 8) DMACTの値が"0"であるか,周辺からのDMA要求が解除されたならば,SADRまたはDADRに増加減を行い,DMACTの値とともにディスクリプタに書き戻します。
- 9) CPUにバス権を返します。

カウンタの値が"0"であった場合,DACSRのDEDnを"1"にするとともに割込み許可になっていれば,CPUに割込みをかけます。

1回あたりの最低所要サイクル数は,ディスクリプタを内蔵RAMに格納し,データを外部バス間で転送,データ長がバイトの場合,以下のようになります。

- 転送元/転送先アドレスともに固定のとき:( $6+5 \times n$ )サイクル
- 転送元/転送先アドレスの一方のみ固定のとき:( $7+5 \times n$ )サイクル
- 転送元/転送先アドレスの両方が増加または減少のとき:( $8+5 \times n$ )サイクル

## バースト転送モード

- 1) 初期化ルーチンでディスクリプタの設定を行います。
- 2) プログラムにより,DMA転送要求発生もとの初期化を行います。内部周辺回路を転送要求元とする場合は,割込み要求を許可にします。同時に,割込みコントローラのICRは割込み禁止にします。
- 3) プログラムにより,所望するDACSRのDOEnビットに"1"を書き込みます。  
---この時点でDMAに関する設定は完了---
- 4) DMACがDMA転送要求入力を検出すると,CPUにバス権確保を要求する。
- 5) CPUがバス権を移譲すると,DMACはディスクリプタにある3ワードの情報をバスを経由してアクセスします。
- 6) DMACTの減算を行いながら,ディスクリプタ内の情報に従った転送をDMACTで設定した回数だけ行います。データの転送中,転送要求受付出力信号を出力します(外部転送要求入力使用の場合)。減算したDMACTが"0"になったとき,データ転送中に転送終了出力信号を出力します。
- 7) SADRまたはDADRの増加減を行い,DMACTの値とともにディスクリプタに書き戻します。
- 8) CPUにバス権を返します
- 9) DACSRのDEDnを"1"にするとともに割込み許可になっていれば,CPUに割込みをかけます。

1回あたりの最低所要サイクル数は,ディスクリプタを内蔵RAMに格納し,データを外部バス間で転送,データ長がバイトの場合は以下ようになります。

- 転送元/転送先アドレスともに固定のとき:  $(6+5 \times n)$  サイクル
- 転送元/転送先アドレスの一方のみ固定のとき:  $(7+5 \times n)$  サイクル
- 転送元/転送先アドレスの両方が増加または減少のとき:  $(8+5 \times n)$  サイクル



## 15.5 DMAコントローラのタイミング図

DMAコントローラの動作タイミング図を示します。

- ・ディスクリプタアクセス部のタイミング図
- ・データ転送部のタイミング図
- ・連続転送モードでの転送停止タイミング図
- ・転送の終了動作タイミング図

タイミング図で使用している記号の説明

表 15.5-1に、タイミング図で使用している記号の説明を示します。

表 15.5-1 タイミング図中の記号説明

記号	記号の意味
#0	ディスクリプタNo.0
#0H	ディスクリプタNo.0のビット31～ビット16
#0L	ディスクリプタNo.0のビット15～ビット0
#1	ディスクリプタNo.1
#1H	ディスクリプタNo.1のビット31～ビット16
#1L	ディスクリプタNo.1のビット15～ビット0
#2	ディスクリプタNo.2
#2H	ディスクリプタNo.2のビット31～ビット16
#2L	ディスクリプタNo.2のビット15～ビット0
#1/2	ディスクリプタNo.1またはNo.2 (SCS1,SCS0,DCS1,DCS0により決まります)
#1/2H	ディスクリプタNo.1またはNo.2のビット31～ビット16
#1/2L	ディスクリプタNo.1またはNo.2のビット15～ビット0
S	転送元
SH	転送元のビット31～ビット16
SL	転送元のビット15～ビット0
D	転送先
DH	転送先のビット31～ビット16
DL	転送先のビット15～ビット0

# 15.5.1 ディスクリプタアクセス部のタイミング図

ディスクリプタアクセス部のタイミング図を示します。

ディスクリプタアクセス部

要求端子入力モード: レベル, ディスクリプタアドレス: 外部

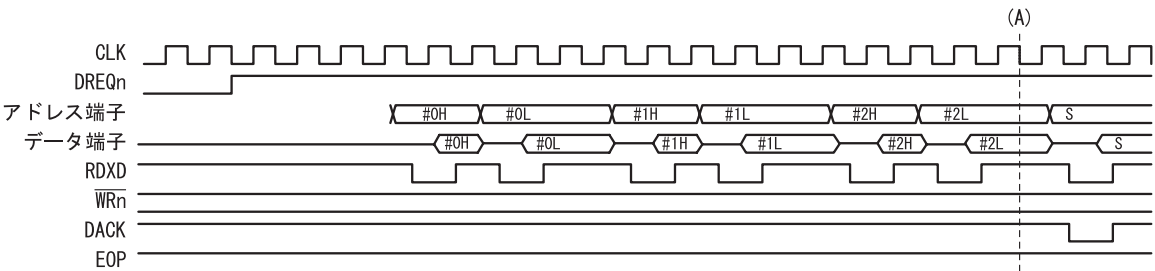


図 15.5-1 ディスクリプタアクセス部のタイミング図1

要求端子入力モード: レベル, ディスクリプタアドレス: 内部

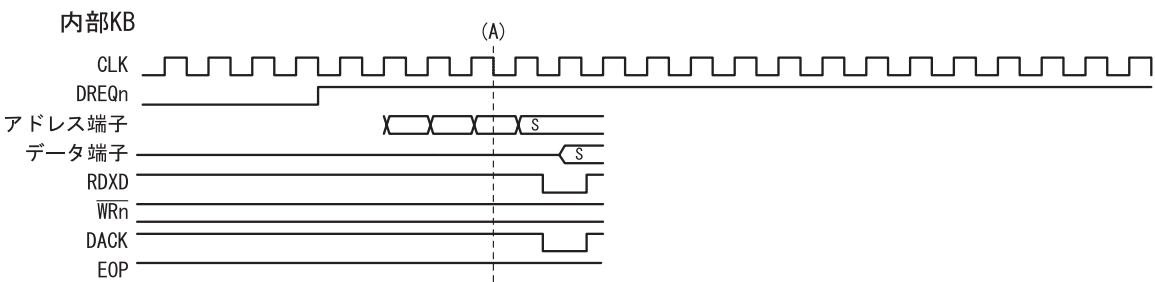


図 15.5-2 ディスクリプタアクセス部のタイミング図2

要求端子入力モード: エッジ, ディスクリプタアドレス: 外部

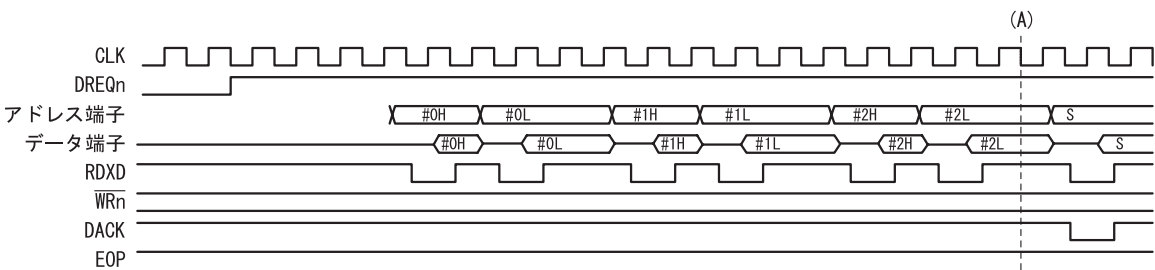


図 15.5-3 ディスクリプタアクセス部のタイミング図3

要求入力モード:エッジ,ディスクリプタアドレス:内部

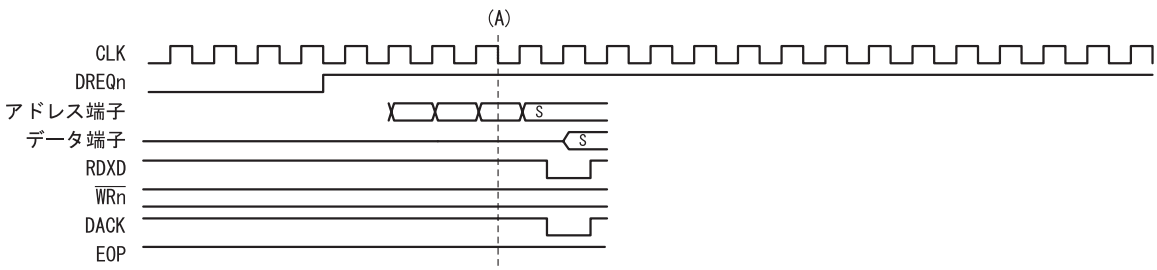


図 15.5-4 ディスクリプタアクセス部のタイミング図4

< 注意事項 >

DREQnの発生からDMAC動作開始までの部分については,最も早い条件の場合を示してあります。  
実際の動作では,CPUの命令フェッチやデータアクセスによるバスの競合があるため,DMACの動作  
開始が遅れる場合があります。

## 15.5.2 データ転送部のタイミング図

データ転送部のタイミング図を示します。

データ転送部, 16/8ビットデータ

転送元領域: 外部, 転送先領域: 外部

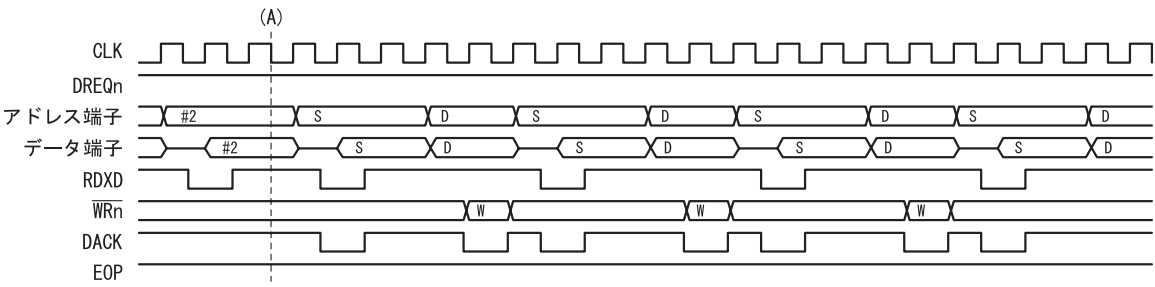


図 15.5-5 データ転送部のタイミング図1

転送元領域: 外部, 転送先領域: 内部RAM

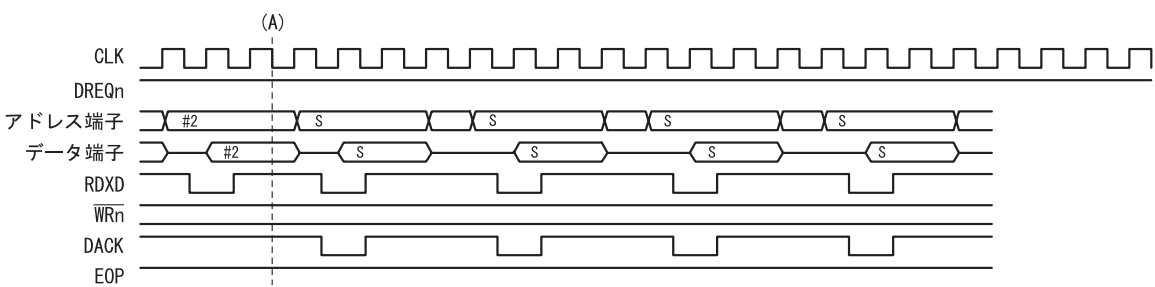


図 15.5-6 データ転送部のタイミング図2

転送元領域: 内部RAM, 転送先領域: 外部

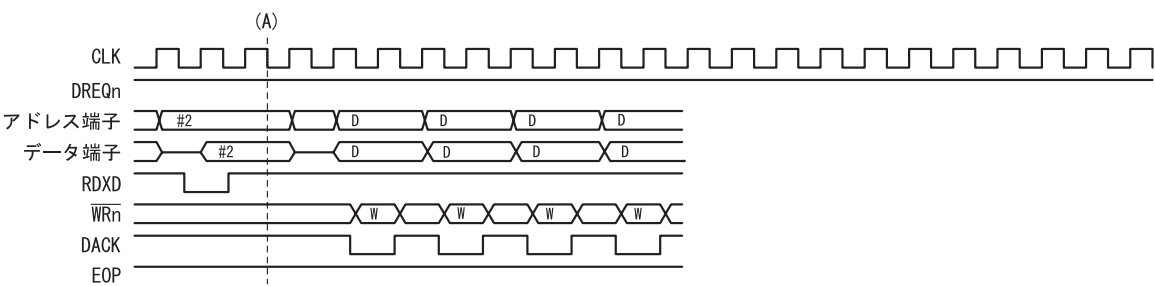


図 15.5-7 データ転送部のタイミング図3

### 15.5.3 連続転送モードでの転送停止タイミング図

連続転送モードでの転送停止タイミング図を示します。

連続転送モードでの転送停止(いずれか一方のアドレスは固定である場合)

16/8ビットデータ

転送元領域: 外部, 転送先領域: 外部

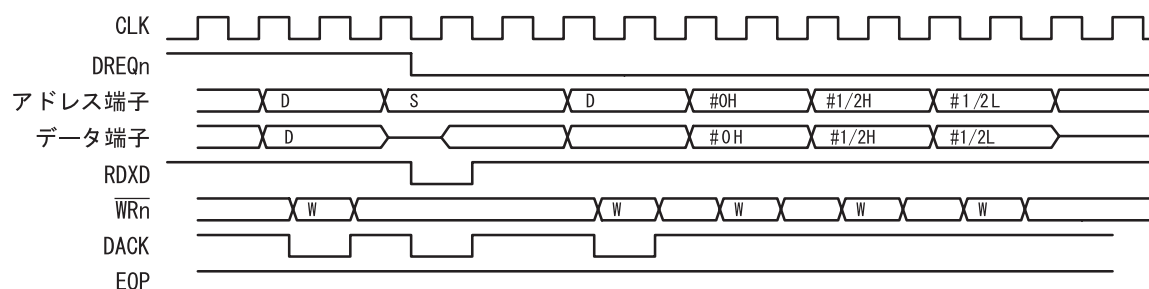


図 15.5-8 連続転送モードでの転送停止タイミング図1

転送元領域: 外部, 転送先領域: 内部RAM

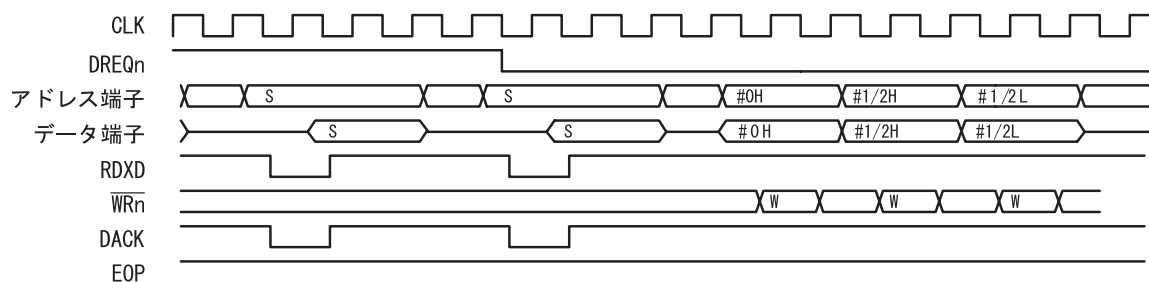


図 15.5-9 連続転送モードでの転送停止タイミング図2

転送元領域: 内部RAM, 転送先領域: 外部

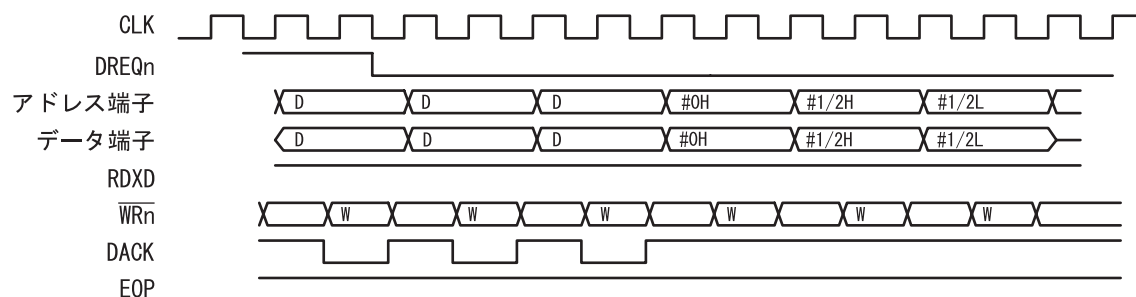


図 15.5-10 連続転送モードでの転送停止タイミング図3

## 連続転送モードでの転送停止(両方のアドレスが変化する場合)16/8ビットデータ

転送元領域:外部,転送先領域:外部

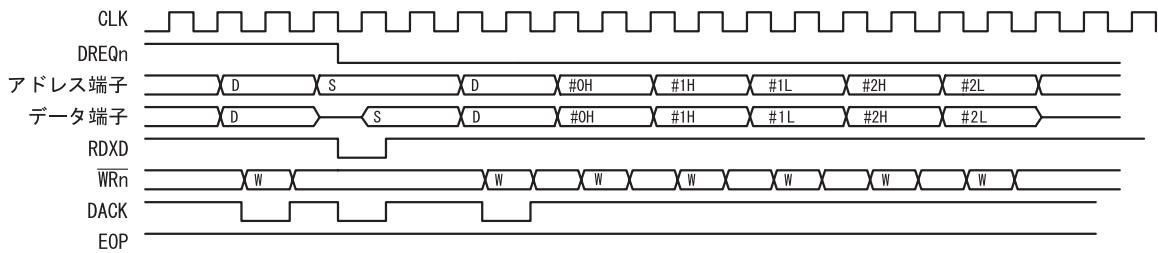


図 15.5-11 連続転送モードでの転送停止タイミング図4

転送元領域:外部,転送先領域:内部RAM

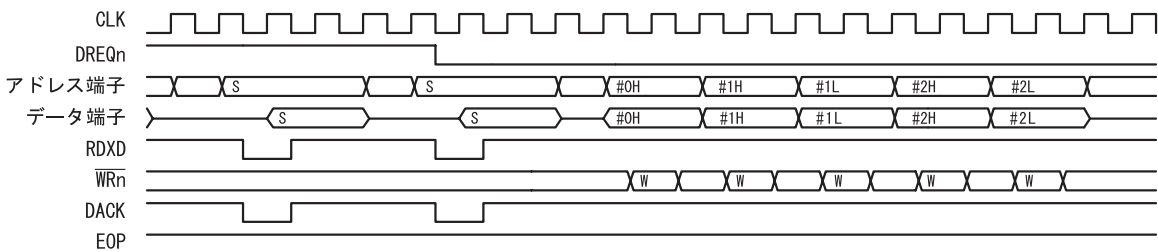


図 15.5-12 連続転送モードでの転送停止タイミング図5

転送元領域:内部RAM,転送先領域:外部

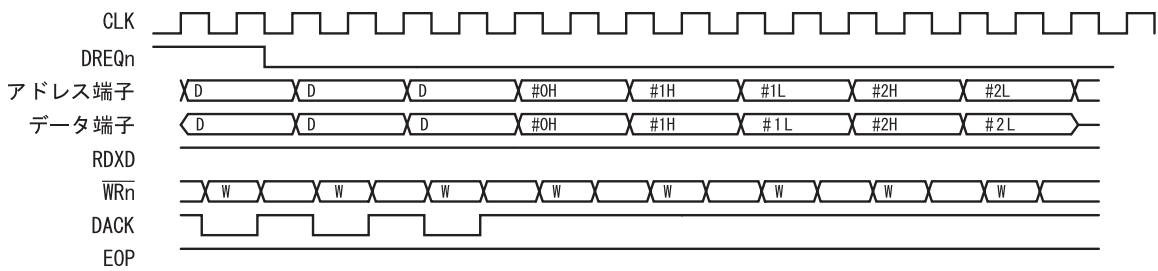


図 15.5-13 連続転送モードでの転送停止タイミング図6

### 15.5.4 転送の終了動作タイミング図

転送の終了動作タイミング図を示します。

転送の終了動作(いずれか一方のアドレスは固定である場合)

バス幅:16ビット,データ長:8/16ビット

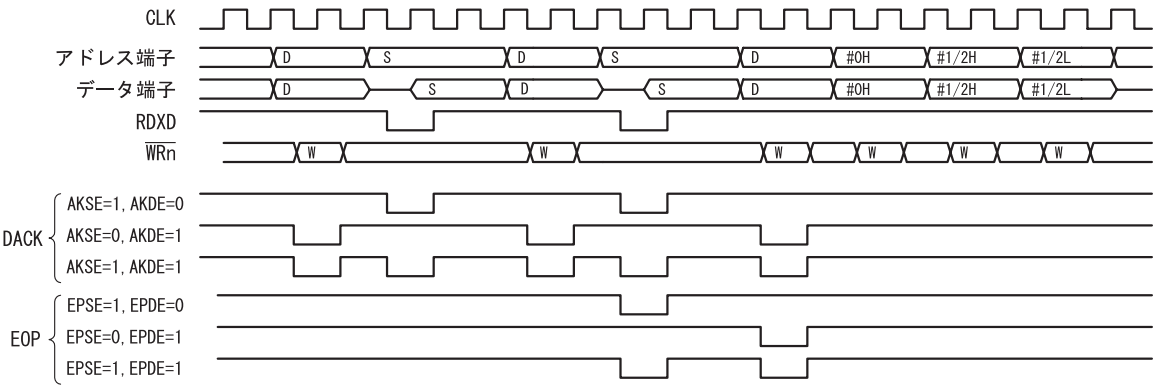


図 15.5-14 転送の終了動作タイミング図1

バス幅:16ビット,データ長:32ビット

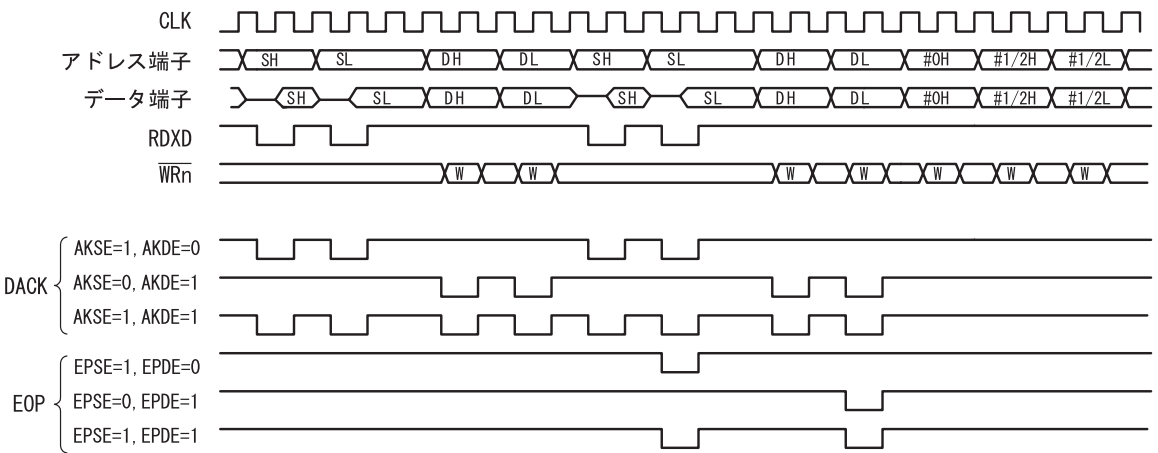


図 15.5-15 転送の終了動作タイミング図2

転送の終了動作(両方のアドレスが変化する場合)

バス幅:16ビット,データ長:8/16ビット

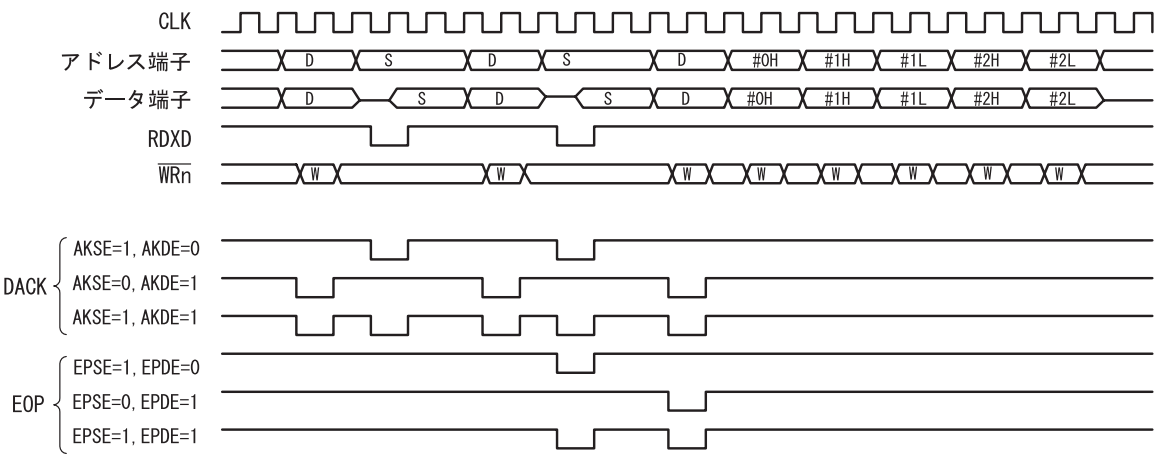


図 15.5-16 転送の終了動作タイミング図3

バス幅:16ビット,データ長:32ビット

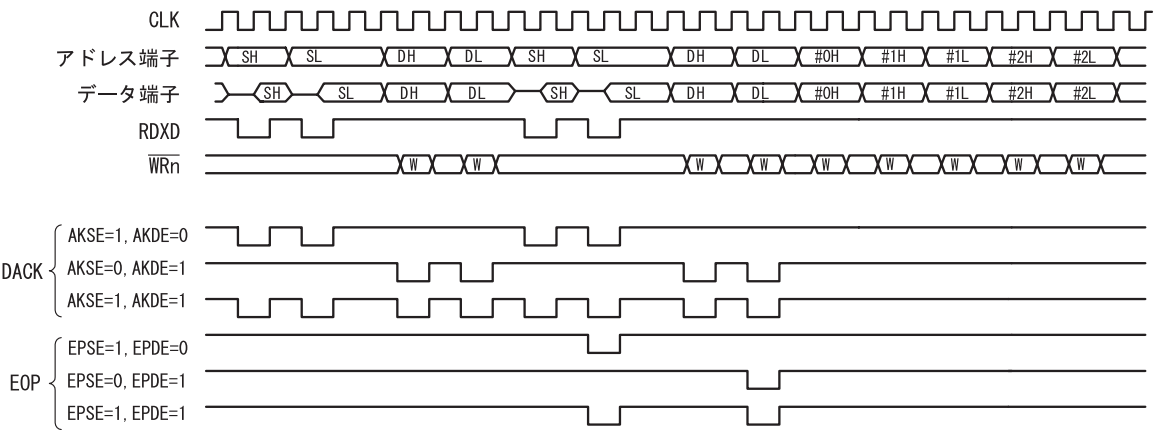


図 15.5-17 転送の終了動作タイミング図4



## 15.6 DMAコントローラに関する注意事項

---

DMAコントローラを使用する上での注意事項について説明します。

---

### チャンネル間優先順位

このDMAコントローラは、一つのチャンネルのDMA転送要求により一度起動されると、ほかのチャンネルのDMA転送要求が発生しても実行中の転送が終了するまでほかのチャンネルの転送要求は受理されず、保留されます。

DMAコントローラがDMA転送要求を検出するときに、複数のチャンネルの要求が同時にアクティブになっていた場合、受理するチャンネルは以下の優先度に従って決まります。

(強) ch0 > ch1 > ch2 > ch3 > ch4 > ch5 > ch6 > ch7 (弱)

複数のチャンネルの要求が同時に発生した場合、一つのチャンネルのDMA転送を実行した後、次のチャンネルのDMA転送を行う前に、一度CPUにバスの制御が戻ります。

### リソースの割込み要求をDMA転送要求として使用する場合

DMAコントローラ転送を望む場合、該当する割込みの割込みコントローラ内の割込みレベルは割込み禁止にする必要があります。

また、逆に割込み発生を望む場合、DMAコントローラ内のDMAコントローラ動作許可ビットを禁止状態にし、割込みレベルを適切な値にする必要があります。

### 優先度の高い割込み発生時のDMA転送禁止

DMA転送要求の発生によるDMA転送の実行中に、より優先度の高い割込み要求が発生した場合、DMA転送を停止させる機能があります。

#### ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)

割込みコントローラのホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)を操作することにより、割込み要求の発生時に、DMA転送動作を停止させることができます。

周辺回路からの割込み要求が発生したときに、その割込み要求に設定された割込みレベルが、HRCLに設定された割込みレベルよりも強い値であった場合、DMACのDMA転送動作が抑止されます。DMA転送動作の実行中であれば、転送動作の切れ目で動作を中止し、バス権をCPUに開放します。DMA転送要求の発生待ちの状態であれば、DMA転送要求が発生しても保留する状態になります。

リセット後、HRCLは最も弱いレベル(31)に設定されるので、すべての割込み要求に対してDMA転送動作は抑止されることになります。割込み要求が発生していてもDMA転送を動作させたいときは、ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)を必要な値に設定してください。

## DMA要求抑止レジスタ(PDRR)

ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の指定によるDMA転送動作の抑止機能は、より高い優先度の割込み要求がアクティブなときのみ有効です。そのため、たとえば割込みハンドラのプログラム中で割込み要求をクリアすると、その時点でホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)によるDMA転送の抑止は解除され、CPUがバス権を失ってしまう場合があります。

ほかの割込み要求を受理できるように割込み要求はクリアして、かつDMA転送動作は抑止するために、クロック制御部のDMA要求抑止レジスタ(PDRR)が用意されています。

割込みハンドラ中でPDRRに"0"以外の値を書き込むと、DMA転送動作は抑止されます。DMA転送動作の抑止を解除したいとき、PDRRに"0"を書き込んでください。

## スリープモード中のDMA転送動作

スリープモード中のDMA転送動作は使用できません。スリープモードにする前に必ずDMA転送動作を動作禁止に設定してください。

## DMAコントローラ内部レジスタへの転送動作

転送先アドレスとして、DMACの内部レジスタは指定しないでください。

## 連続転送について

連続転送モードでは、デバイスの内部バスバッファの状態によっては、転送中であつてもディスクリプタの書き戻しが発生する場合があります。その場合でも転送動作そのものは終了することはなく、再び継続して行われます。

## 内部メモリを外部転送する動作

ブロック転送モード時に、1回のDREQに対して、2回DMA転送を行ってしまいます。

また、連続転送モード時には、DREQを落としても1回余分にDMA転送をしてしまいます。以下のいずれかを選択して対応してください。

- DREQをエッジ検出モードで使用する(ブロックモード時のみ有効)。
- 転送先アドレスを外部領域に設定し、DACKを転送先にアクセス時に発生させるようにする。
- 転送元/転送先アドレスともに固定の場合を除き、ディスクリプタを外部メモリに設定する。



## 第16章 ビットサーチモジュール

---

この章では、ビットサーチモジュールの概要、レジスタ構成/機能およびビットサーチモジュールの動作、退避/復帰処理について説明します。

---

- 16.1 ビットサーチモジュールの概要
- 16.2 ビットサーチモジュールのレジスタ
- 16.3 ビットサーチモジュールの動作と退避/復帰処理

## 16.1 ビットサーチモジュールの概要

入力レジスタに書き込まれたデータに対して,0,1または変化点を検索し,検出したビット位置を返します。

ブロックダイヤグラム

図 16.1-1に,ビットサーチモジュールのブロックダイヤグラムを示します。

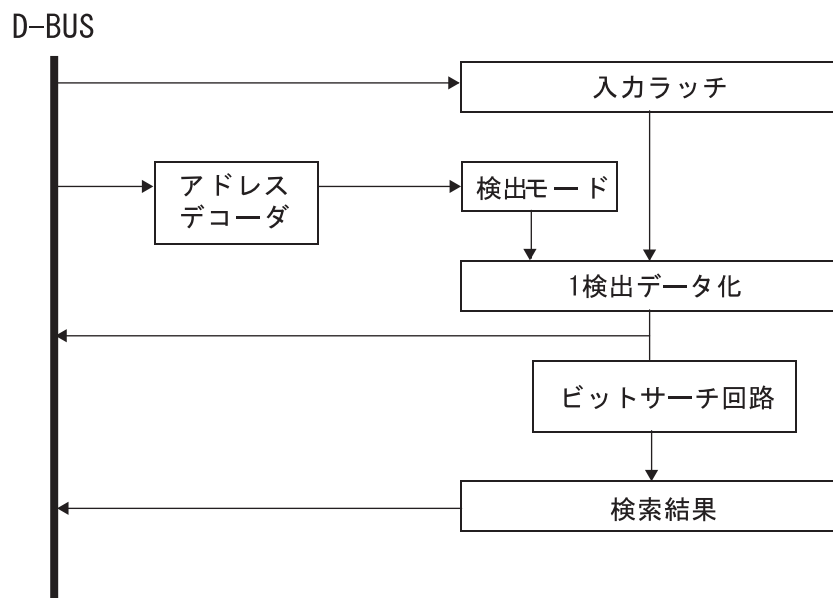


図 16.1-1 ビットサーチモジュールのブロックダイヤグラム

## 16.2 ビットサーチモジュールのレジスタ

ビットサーチモジュールのレジスタには以下の四つがあります。

- ・ 0検出用データレジスタ (BSD0)
- ・ 1検出用データレジスタ (BSD1)
- ・ 変化点検出用データレジスタ (BSDC)
- ・ 検出結果レジスタ (BSRR)

ビットサーチモジュールのレジスタ一覧

bit 31	0		
		0 検出用 データレジスタ	(BSD0)
		1 検出用 データレジスタ	(BSD1)
		変化点検出用 データレジスタ	(BSDC)
		検出結果レジスタ	(BSRR)

図 16.2-1 ビットサーチモジュールのレジスタ一覧

0検出用データレジスタ (BSD0)

図 16.2-2に, 0検出用データレジスタ (BSD0)のレジスタ構成を示します。

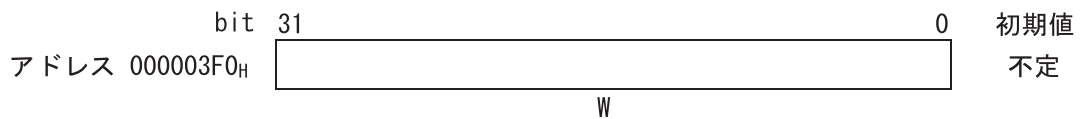


図 16.2-2 0検出用データレジスタ (BSD0)のレジスタ構成

書き込んだ値に対して, 0検出を行います。リセットによる初期値は不定です。  
読出し値は不定です。データ転送には, 32ビット長のデータ転送命令を使用して  
ください。8ビット長, 16ビット長のデータ転送命令は, 使わないでください。

### 1検出用データレジスタ(BSD1)

図 16.2-3に,1検出用データレジスタ(BSD1)のレジスタ構成を示します。

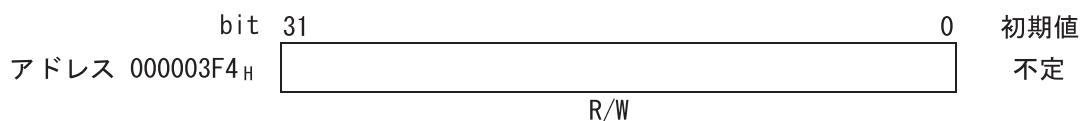


図 16.2-3 1検出用データレジスタ(BSD1)のレジスタ構成

データ転送には,32ビット長のデータ転送命令を使用してください。8ビット長,16ビット長のデータ転送命令は,使わないでください。

#### 書き込み時

書き込んだ値に対して,"1"を検出します。

#### 読出し時

ビットサーチモジュールの内部状態の退避用データが読み出されます。割込みハンドラ等がビットサーチモジュールを使用する場合に,元の状態を退避復帰するときに使用します。

0検出,変化点検出,データレジスタにデータを書き込んだ場合でも,1検出用データレジスタのみ操作することで退避復帰できます。

リセットによる初期値は不定です。

### 変化点検出用データレジスタ(BSDC)

図 16.2-4に,変化点検出用データレジスタ(BSDC)のレジスタ構成を示します。

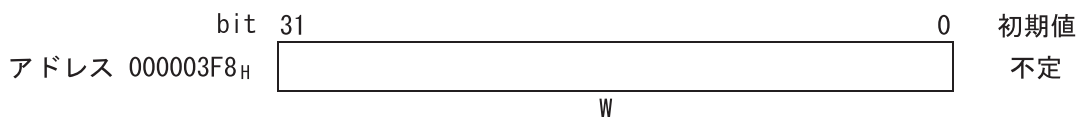


図 16.2-4 変化点検出用データレジスタ(BSDC)のレジスタ構成

書き込んだ値に対して,変化点の検出を行います。リセットによる初期値は不定です。

読出し値は不定です。データ転送には,32ビット長のデータ転送命令を使用してください。8ビット長,16ビット長のデータ転送命令は,使わないでください。

### 検出結果レジスタ(BSRR)

図 16.2-5に,検出結果レジスタ(BSRR)のレジスタ構成を示します。

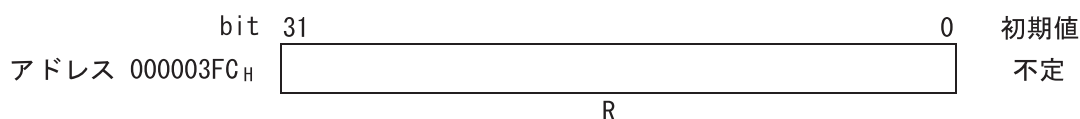


図 16.2-5 検出結果レジスタ(BSRR)のレジスタ構成

0検出,1検出または変化点検出の結果が読み出されます。どの検出結果が読み出されるかは,最後に書き込んだデータレジスタによって決定されます。

## 16.3 ビットサーチモジュールの動作と退避/復帰処理

ビットサーチモジュールの0検出,1検出および変化点検出の動作と退避/復帰処理について説明します。

### 0検出

0検出用データレジスタに書き込まれたデータをMSBからLSBへスキャンし,最初の"0"を検出した位置を返します。検出結果は,検出結果レジスタを読み出すことで得られます。

表 16.3-1に,検出した位置と返す数値の関係を示します。

"0"が存在しないとき,(すなわちFFFFFFFF<sub>H</sub>という数値のとき,)32という値をサーチ結果として返します。

#### 【実行例】

書込みデータ				読出し値(10進)	
11111111	11111111	11110000	00000000 <sub>B</sub>	(FFFFFF00 <sub>H</sub> )	20
11111000	01001001	11100000	10101010 <sub>B</sub>	(F849E0AA <sub>H</sub> )	5
10000000	00000010	10101010	10101010 <sub>B</sub>	(8002AAAA <sub>H</sub> )	1
11111111	11111111	11111111	11111111 <sub>B</sub>	(FFFFFFFF <sub>H</sub> )	32

### 1検出

1検出用データレジスタに書き込まれたデータをMSBからLSBへスキャンし,最初の"1"を検出した位置を返します。検出結果は,検出結果レジスタを読み出すことで得られます。

表 16.3-1に,検出した位置と返す値との関係を示します。

"1"が存在しないとき,(すなわち00000000<sub>H</sub>という数値のとき,)32という値をサーチ結果として返します。

#### 【実行例】

書込みデータ				読出し値(10進)	
00100000	00000000	00000000	00000000 <sub>B</sub>	(20000000 <sub>H</sub> )	2
00000001	00100011	01000101	01100111 <sub>B</sub>	(01234567 <sub>H</sub> )	7
00000000	00000011	11111111	11111111 <sub>B</sub>	(0003FFFF <sub>H</sub> )	14
00000000	00000000	00000000	00000001 <sub>B</sub>	(00000001 <sub>H</sub> )	31
00000000	00000000	00000000	00000000 <sub>B</sub>	(00000000 <sub>H</sub> )	32



## 変化点検出

変化点検出用データレジスタに書き込まれたデータを、ビット30からLSBへスキャンし、MSBの値と比較します。最初にMSBと異なる値を検出した位置を返します。検出結果は、検出結果レジスタを読み出すことで得られます。

表 16.3-1に、検出した位置と返す値を示します。

変化点が存在しないときは、32を返します。変化点検出では、結果として0を返すことはありません。

## 【実行例】

書き込みデータ				読出し値(10進)	
00100000	00000000	00000000	00000000 <sub>B</sub>	(20000000 <sub>H</sub> )	2
00000001	00100011	01000101	01100111 <sub>B</sub>	(01234567 <sub>H</sub> )	7
00000000	00000011	11111111	11111111 <sub>B</sub>	(0003FFFF <sub>H</sub> )	14
00000000	00000000	00000000	00000001 <sub>B</sub>	(00000001 <sub>H</sub> )	31
00000000	00000000	00000000	00000000 <sub>B</sub>	(00000000 <sub>H</sub> )	32
11111111	11111111	11110000	00000000 <sub>B</sub>	(FFFFFF00 <sub>H</sub> )	20
11111000	01001001	11100000	10101010 <sub>B</sub>	(F849E0AA <sub>H</sub> )	5
10000000	00000010	10101010	10101010 <sub>B</sub>	(8002AAAA <sub>H</sub> )	1
11111111	11111111	11111111	11111111 <sub>B</sub>	(FFFFFFFF <sub>H</sub> )	32

表 16.3-1 ビット位置と返す値(10進)

検出した ビット位置	返す値	検出した ビット位置	返す値	検出した ビット位置	返す値	検出した ビット位置	返す値
31	0	23	8	15	16	7	24
30	1	22	9	14	17	6	25
29	2	21	10	13	18	5	26
28	3	20	11	12	19	4	27
27	4	19	12	11	20	3	28
26	5	18	13	10	21	2	29
25	6	17	14	9	22	1	30
24	7	16	15	8	23	0	31
						存在しない	32

## 退避/復帰の処理

割込みハンドラ中で、ビットサーチモジュールを使う場合のように、ビットサーチモジュールの内部状態を退避/復帰させる必要がある場合は、以下の手順に従ってください。

- 1) 1検出用データレジスタを読出し、この内容を保存する(退避)
- 2) ビットサーチモジュールを使用
- 3) 1)で退避したデータを1検出用データレジスタに書き込む(復帰)

以上の操作により、次に検出結果レジスタを読み出した時に得られる値は、1)以前に、ビットサーチモジュールに書き込まれた内容に応じたものとなります。

最後に書き込まれたデータレジスタが0検出用または変化点検出用であっても、上記手順で正しく元に戻ります。

## 第17章 フラッシュメモリ

---

この章では、フラッシュメモリの概要、レジスタの構成/機能および動作について説明します。

---

- 17.1 フラッシュメモリの概要
- 17.2 フラッシュメモリのレジスタ
- 17.3 フラッシュメモリのアクセスモード
- 17.4 自動アルゴリズム起動方法
- 17.5 自動アルゴリズム実行状態
- 17.6 セクタプロテクトオペレーション

## 17.1 フラッシュメモリの概要

MB91F127およびMB91F128には、それぞれ256Kバイト(2Mビット)および510Kバイト(4Mビット)の容量で、+3V単一電源による全セクター一括消去およびセクタ単位での消去が可能で、FR-CPUによるハーフワード(16ビット)単位での書込みが可能なフラッシュメモリを内蔵しています。

### フラッシュメモリの概要

このフラッシュメモリは、内蔵の3V動作256Kバイト(MB91F127)/510Kバイト(MB91F128)フラッシュメモリです。

当社製4Mビット(510K×8/254K×16)フラッシュメモリMBM29LV400TCと(一部セクタ構成を除いて)同じものであり、ROMライターによるデバイス外部からの書込みも可能です。

また、MBM29LV400TC相当の機能に加え、FR-CPUの内蔵ROMとして使用する場合には、ワード(32ビット)単位での命令/データ読出しが可能であり、デバイスの高速動作を実現できます。

本書と併せてMBM29LV400TCデータシートもご参照ください。

フラッシュメモリマクロとFR-CPUインタフェース回路の組合せにより、以下の機能を実現します。

- CPUのプログラム/データ格納用メモリとして機能
  - ROMとして使用する際は32ビットバス幅でアクセス可能
  - CPUによる読出し/書込み/消去(自動プログラムアルゴリズム<sup>\*1</sup>)が可能
- 単体フラッシュメモリ製品のMBM29LV400TC相当の機能
  - ROMライターによる読出し/書込み/消去(自動プログラムアルゴリズム<sup>\*1</sup>)が可能

<sup>\*1</sup>:自動プログラムアルゴリズム= Embedded Algorithm TM

ここでは、FR-CPUからこのフラッシュメモリを使用する場合について解説しています。

ROMライターからこのフラッシュメモリを使用する場合の詳細につきましては、別途ROMライターの取り扱い説明書を参照ください。

### 自動アルゴリズム実行状態

CPUプログラミングモードにて自動アルゴリズムを起動した場合には、内部レディ/ビジー信号(RDY/BUSYX)により自動アルゴリズムの動作状態を知ることができます。このレディ/ビジー信号のレベルは、フラッシュメモリステータスレジスタの"RDY"ビットとして読むことができます。

"RDY"ビットが"0"の期間は、自動アルゴリズムによる書込みまたは消去が行われており、新たな書込みまたは消去コマンドを受け付けることはできません。また、フラッシュメモリアドレスからのデータ読み出しはできません。"RDY"ビットが"0"の期間に読み出したデータはフラッシュメモリのステータスを示すハードウェアシーケンスフラグとなっています(「17.4 自動アルゴリズム起動方法」を参照してください)。

### 割込み制御

自動アルゴリズムのシーケンス終了により、CPUに対し割込み要求を発生することができます。これにより、長時間にわたる自動アルゴリズムシーケンスの終了を即座に知ることが可能です。

自動アルゴリズム終了割込みは、フラッシュメモリステータスレジスタの"RDYINT"ビットと"INTE"ビットにより制御されます。

"RDYINT"ビットは自動アルゴリズム終了割込みフラグです。内部レディ/ビジー信号 (RDY/BUSYX) の"0"から"1"の立上りを検出すると,"1"にセットされます。"INTE"ビットが"1"のとき,"RDYINT"ビットがセットされるとCPUに対して割込み要求が出力されます。

割込み要求をキャンセルするには"RDYINT"ビットまたは"INTE"ビットに"0"を書き込んでください。

### ROMライターによる書込み

このフラッシュメモリは、ROMライターを用いてデバイス外部より書込みを行うことができます。この状態では、単体フラッシュメモリ製品のMBM29LV400TC相当の端子機能がデバイス外部端子に割り当てられ、FR-CPUは動作を停止します。CPUモードとはアドレス線の接続が変更され、メモリ領域内のマッピングが変化します。詳細については『対応ROMライターの仕様書』を参照してください。

### フラッシュメモリのブロックダイアグラム

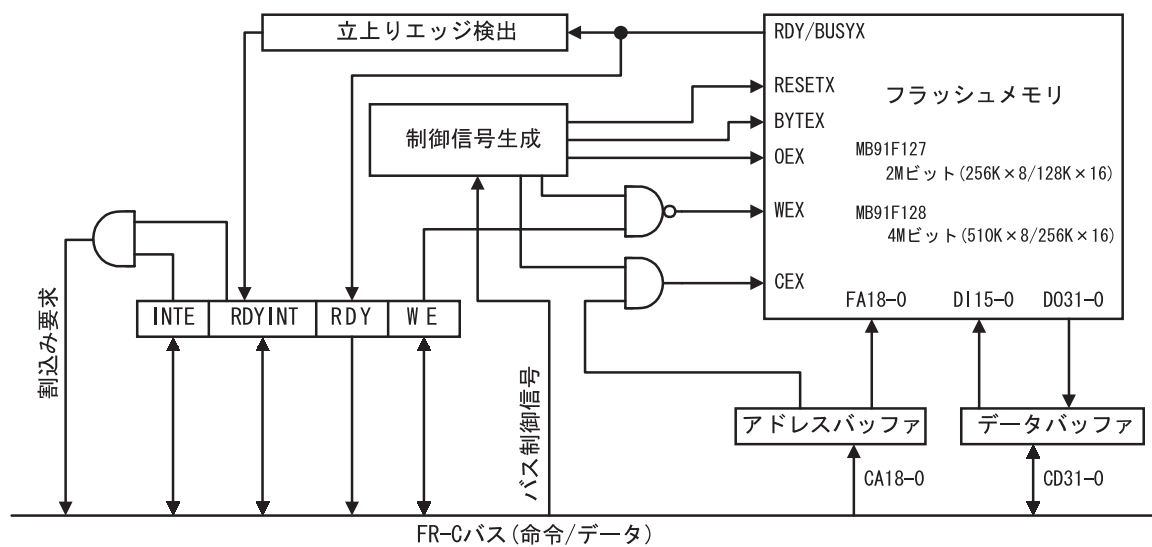


図 17.1-1 フラッシュメモリのブロックダイアグラム

## フラッシュメモリのセクタ構成 (MB91F127)

フラッシュメモリのアドレスマッピングは、FR-CPUからのアクセス時とROMライタでのアクセス時とは異なります。図 17.1-2と表 17.1-1にFR-CPUからのアクセス時におけるマッピングを、図 17.1-3にROMライタからのアクセス時におけるマッピングを示します。

FR-CPUからのアクセス時におけるマッピング

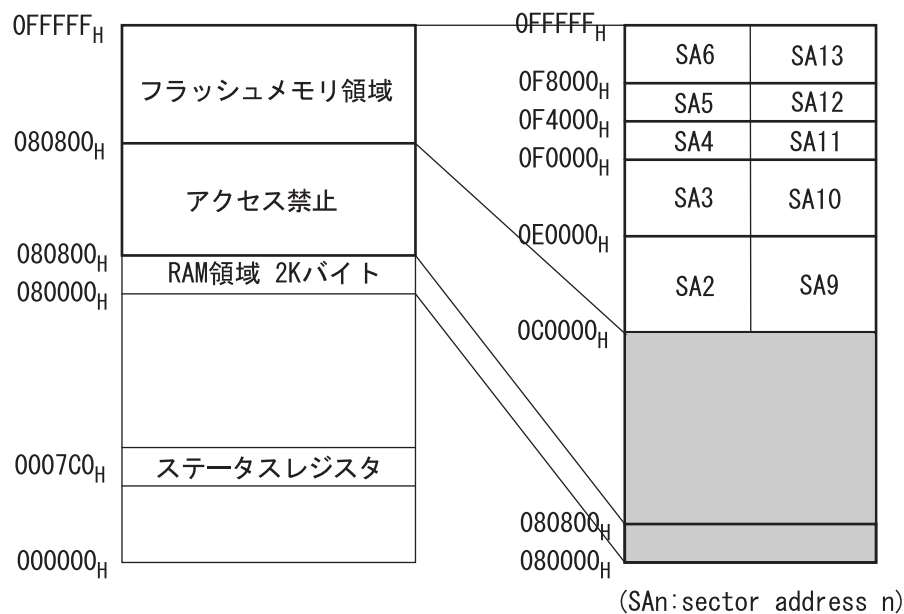


図 17.1-2 FR-CPUからのアクセス時におけるマッピング(MB91F127)

表 17.1-1 セクタアドレス一覧(MB91F127)

セクタ アドレス	アドレス範囲	対応ビット位置	セクタ容量
SA9	0C0002,3 <sub>H</sub> ~ 0DFFFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	64 Kバイト
SA10	0E0002,3 <sub>H</sub> ~ 0EFFFF, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	32 Kバイト
SA11	0F0002,3 <sub>H</sub> ~ 0F3FFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	8 Kバイト
SA12	0F4002,3 <sub>H</sub> ~ 0F7FFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	8 Kバイト
SA13	0F8002,3 <sub>H</sub> ~ 0FFFFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	16 Kバイト
SA2	0C0000,1 <sub>H</sub> ~ 0DFFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	64 Kバイト
SA3	0E0000,1 <sub>H</sub> ~ 0EFFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	32 Kバイト
SA4	0F0000,1 <sub>H</sub> ~ 0F3FFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	8 Kバイト
SA5	0F4000,1 <sub>H</sub> ~ 0F7FFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	8 Kバイト
SA6	0F8000,1 <sub>H</sub> ~ 0FFFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	16 Kバイト

ROMライターからのアクセス時におけるマッピングとセクタ構成

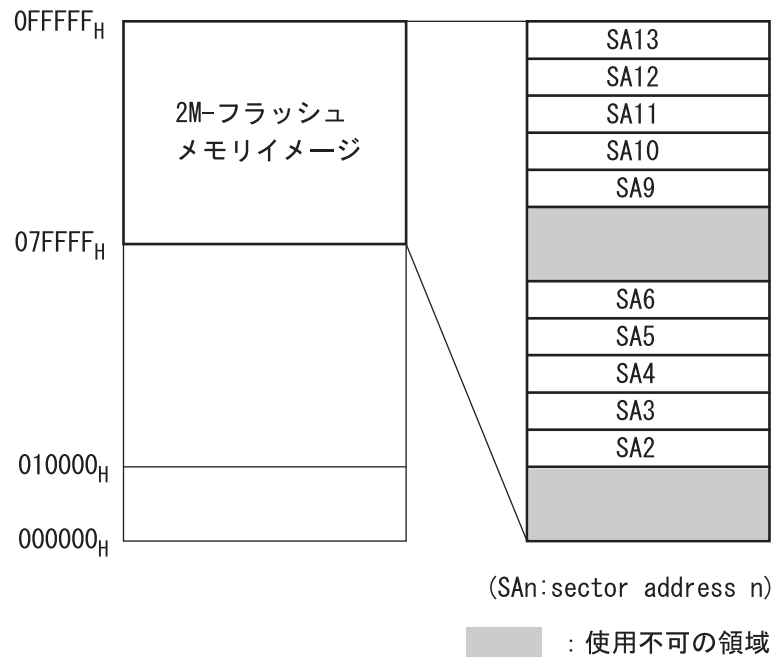


図 17.1-3 ROMライターからのアクセス時におけるマッピングとセクタ構成(MB91F127)

## フラッシュメモリのセクタ構成(MB91F128)

フラッシュメモリのアドレスマッピングは,FR-CPUからのアクセス時とROMライターでのアクセス時とは異なります。図 17.1-4と表 17.1-2にFR-CPUからのアクセス時におけるマッピングを, 図 17.1-5にROMライターからのアクセス時におけるマッピングを示します。

## FR-CPUからのアクセス時におけるマッピング

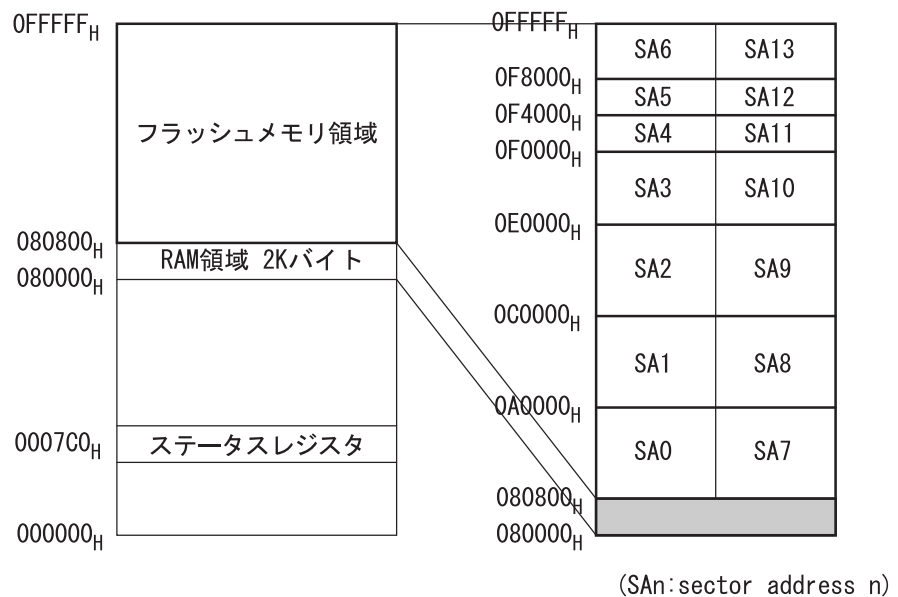


図 17.1-4 FR-CPUからのアクセス時におけるマッピング(MB91F128)

表 17.1-2 セクタアドレス一覧(MB91F128)

セクタ アドレス	アドレス範囲	対応ビット位置	セクタ容量
SA7	080802,3 <sub>H</sub> ~ 09FFFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	64 Kバイト
SA8	0A0002,3 <sub>H</sub> ~ 0BFFFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	64 Kバイト
SA9	0C0002,3 <sub>H</sub> ~ 0DFFFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	64 Kバイト
SA10	0E0002,3 <sub>H</sub> ~ 0EFFFF, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	32 Kバイト
SA11	0F0002,3 <sub>H</sub> ~ 0F3FFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	8 Kバイト
SA12	0F4002,3 <sub>H</sub> ~ 0F7FFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	8 Kバイト
SA13	0F8002,3 <sub>H</sub> ~ 0FFFFE, F <sub>H</sub> (LSB側 16ビット)	ビット15 ~ 0	16 Kバイト
SA0	080800,1 <sub>H</sub> ~ 09FFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	64 Kバイト
SA1	0A0000,1 <sub>H</sub> ~ 0BFFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	64 Kバイト
SA2	0C0000,1 <sub>H</sub> ~ 0DFFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	64 Kバイト
SA3	0E0000,1 <sub>H</sub> ~ 0EFFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	32 Kバイト
SA4	0F0000,1 <sub>H</sub> ~ 0F3FFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	8 Kバイト
SA5	0F4000,1 <sub>H</sub> ~ 0F7FFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	8 Kバイト
SA6	0F8000,1 <sub>H</sub> ~ 0FFFFC, D <sub>H</sub> (MSB側 16ビット)	ビット31 ~ 16	16 Kバイト

ROMライタからのアクセス時におけるマッピングとセクタ構成

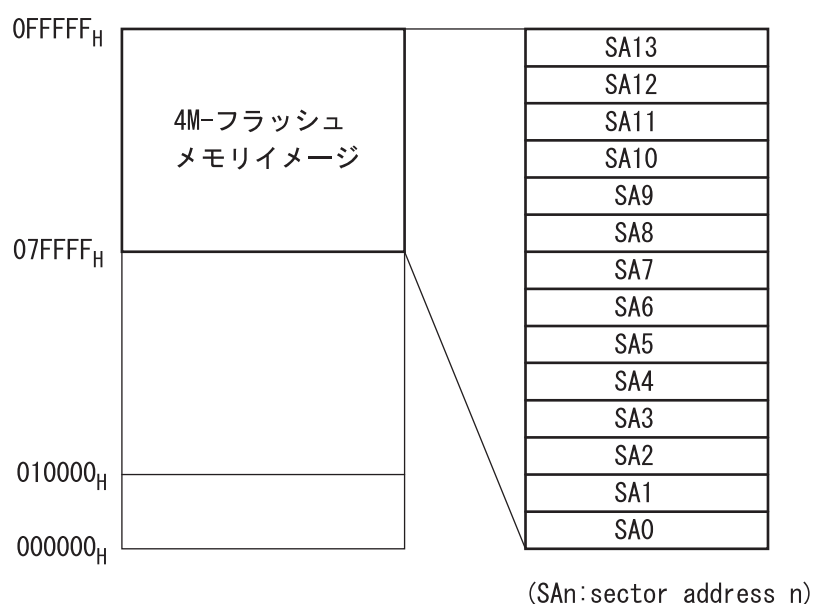


図 17.1-5 ROMライタからのアクセス時におけるマッピングとセクタ構成(MB91F128)

## 17.2 フラッシュメモリのレジスタ

---

フラッシュメモリにはフラッシュメモリステータスレジスタ (FSTR) とフラッシュメモリウェイトレジスタ (FWTC) の二つのレジスタがあります。

---

フラッシュメモリのレジスタ一覧

図 17.2-1に、フラッシュメモリのレジスタ一覧を示します。

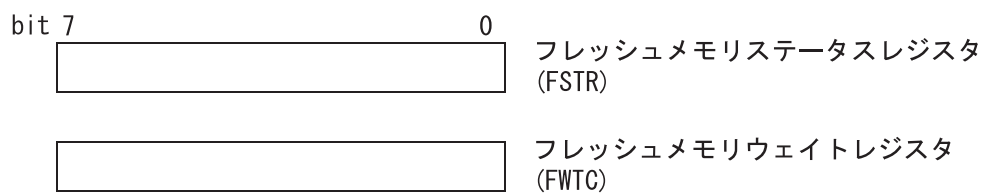


図 17.2-1 フラッシュメモリのレジスタ一覧



# 17.2.1 フラッシュメモリステータスレジスタ(FSTR)

フラッシュメモリステータスレジスタ(FSTR)は、フラッシュメモリの動作状態を示すレジスタです。

## フラッシュメモリステータスレジスタ(FSTR)の構成

このレジスタは、CPUに対する割込みの制御や、フラッシュメモリへの書込みの制御を行います。CPUでのみアクセスすることが可能です。ライター搭載時はアクセスできません。

このレジスタは、リードモディファイライト系命令ではアクセスしないでください。

図 17.2-2に、フラッシュメモリステータスレジスタ(FSTR)のビット構成を示します。

bit	7	6	5	4	3	2	1	0	初期値
アドレス	000007C0 <sub>H</sub>	INTE	RDYINT	WE	RDY	—	—	—	000XXXX0 <sub>B</sub>
		R/W	R/W	R/W	R	—	—	—	R/W

図 17.2-2 フラッシュメモリステータスレジスタ(FSTR)のビット構成

以下に、フラッシュメモリステータスレジスタ(FSTR)の各ビットの機能を説明します。

### 【ビット7】 INTE(INTerrupt Enable)

このビットは、フラッシュメモリの自動アルゴリズム(書込み/消去など)の終了による割込み発生を制御します。

リセット時は"0"に初期化されます。読出し/書込み可能です。

INTE	機能
0	自動アルゴリズム終了割込み出力禁止 (初期値)
1	自動アルゴリズム終了割込み出力許可

### 【ビット6】 RDYINT(ReaDY INTerrupt)

このビットは、フラッシュメモリの自動アルゴリズム(書込み/消去など)の終了により"1"になります。

ビット7のINTEを"1"に設定により割込み発生が許可されているときに、このビットが"1"になると、自動アルゴリズム終了割込み要求が発生します。

リセット時"0"に初期化されます。読出し/書込み可能です。ただし書込みは"0"のみ可能で、"1"を書き込んでもビット値は変化しません。

クリア要因: 命令による"0"の書込みによりクリア

セット要因: 自動アルゴリズムの終了によりセット(RDY/BUSYX信号の立上りエッジを検出)

## 【ビット5】 WE(Write Enable)

このビットは,CPUモードにおけるフラッシュメモリへのデータおよびコマンドの書込みを制御します。

このビットが"0"の間は,フラッシュメモリへのデータおよびコマンドの書込みはすべて無効になります。また,フラッシュメモリからのデータ読出しが32ビットアクセスとなります。

このビットが"1"の間は,フラッシュメモリへのデータおよびコマンドの書込みが有効となり,自動アルゴリズムの起動が可能となります。ただし,フラッシュメモリからのデータ読出しは16ビットアクセスとなり,32ビットアクセスはできないため,この間はプログラムメモリとして使用できません。

このビットの書換えは,必ずRDYビットにより自動アルゴリズム(書込み/消去)が停止していることを確認してから行ってください。RDYビットが"0"の間は,このビットの値を書き換えることができません。

リセット時"0"に初期化されます。読出し/書込み可能です。

WE	機能
0	フラッシュメモリへの書込み禁止および32ビット読出しの許可 (ROMモード) (初期値)
1	フラッシュメモリへの書込み許可および32ビット読出しの禁止 (プログラミングモード)

## 【ビット4】 RDY(ReaDY)

このビットは,自動アルゴリズム(書込み/消去)の動作状態を表示します。

このビットが"0"の間は,自動アルゴリズムによる書込みまたは消去が行われており,新たな書込みまたは消去コマンドを受け付けることはできません。また,フラッシュメモリアドレスからのデータ読出しはできません。読出したデータはフラッシュメモリのステータスを示します。

詳しくは,「17.5 自動アルゴリズム実行状態」を参照してください。

リセット時"0"に初期化されます。読出し/書込み可能です。

RDY	機能
0	書込み/消去動作中で,データの読出しや書込み/消去コマンドの受け付けはできません。
1	データの読出しおよび書込み/消去コマンドの受け付けが可能です。

## 【ビット3～1】 (reserved)

このビットは,予約ビットです。

読み出し値は不定で,書込みは動作に影響を与えません。

## 【ビット0】 (reserved)

このビットは,予約ビットです。

読み出し値は"0"です。このビットには常に"0"を書き込んでください。

"1"を書き込んだ場合の動作は保証できません。リセット時"0"に初期化されます。

## 17.2.2 フラッシュメモリウェイトレジスタ(FWTC)

フラッシュメモリウェイトレジスタ(FWTC)は,CPUモードにおいてフラッシュメモリのウェイト制御を行います。また,フラッシュメモリの高速読出しアクセスを行うための制御をします。

フラッシュメモリウェイトレジスタ(FWTC)の構成

図 17.2-3に,フラッシュメモリウェイトレジスタ(FWTC)のビット構成を示します。

bit	7	6	5	4	3	2	1	0	初期値
アドレス 000007C4 <sub>H</sub>	—	—	—	—	—	FACH	WTC1	WTC0	XXXXX000 <sub>B</sub>
	—	—	—	—	—	W	R/W	R/W	

図 17.2-3 フラッシュメモリウェイトレジスタ(FWTC)のビット構成

以下に,フラッシュメモリウェイトレジスタ(FWTC)の各ビットの機能を説明します。

【ビット1,0】 WTC1,WTC0

これらのビットは,フラッシュメモリのウェイト制御をします。

WTC1,WTC0	機能
00	+0ウェイト,2サイクル (初期値)
01	+1ウェイト,3サイクル
10	使用できません
11	使用できません

【ビット2】 FACH

このビットは,フラッシュメモリの読出し速度の制御をします。

FACH	機能
0	通常読出し (初期値)
1	高速読出し

## 17.3 フラッシュメモリのアクセスモード

FR-CPUによりアクセスする場合、以下の2種類のアクセスモードが存在します。

- ・FR-CPUモード: 内部バスを通してCPUにアクセス。  
ワード(32ビット)長データを一括で読み出すことができますが、  
書込みはできません。
- ・FLASHメモリモード: 外部端子から直接アクセス。  
ワード(32ビット)長アクセスは禁止されていますが、  
ハーフワード(16ビット)長での書込みが可能になります。

### FR-CPU ROMモード(32ビット, 読出しのみ)

FR-CPUの内蔵ROMとして機能するモードです。ワード(32ビット)長データの一括読出しが可能となりますが、フラッシュへの書込み、自動アルゴリズムの起動はできません。

#### モードの指定方法

フラッシュメモリステータスレジスタの"WE"ビットが"0"のとき、このモードとなります。

CPU動作時でのリセット解除後は常にこのモードになります。

CPU動作時以外ではこのモードにすることができません。

#### 動作内容

フラッシュメモリ領域の読出し時、メモリよりワード(32ビット)長データを一括で読み出します。

読出しにかかるサイクル数は2サイクル/1ワード(1ウェイト)です。これによりFR-CPUに対し、ウェイトなしで命令を供給することができます。

#### 制限事項

ROMライタ書込み時とはアドレスの与え方およびエンディアンが異なります。

このモードでは、フラッシュメモリへのコマンド/データともに書込みはできません。

### FR-CPU プログラミングモード(16ビット, 読出し/書込み)

データの消去/書込みが可能になるモードです。ワード(32ビット)長データの一括アクセスは不可能ですので、このモードで動作している間はフラッシュメモリ上のプログラムは実行が不可能となります。

#### モード指定の方法

フラッシュメモリステータスレジスタ(FSTR)の"WE"ビットが"1"のときこのモードとなります。

CPU動作時でのリセット解除後は"WE"ビットが"0"になっています。このモードにするためには"1"を書き込んでください。再度"0"を書き込むかまたはリセットの発生により"WE"ビットが"0"になるとROMモードに戻ります。

フラッシュメモリステータスレジスタ(FSTR)の"RDY"ビットが"0"の間は"WE"ビットを書き換えることができません。"RDY"ビットが"1"になったことを確認してから"WE"ビットを書き換えてください。

## 動作内容

フラッシュメモリ領域の読出し時、メモリよりハーフワード(16ビット)長データを一括で読み出します。読出しにかかるサイクル数は2サイクル/ハーフワード(1ウェイト)です。

フラッシュメモリへのコマンド書込みにより、自動アルゴリズムを起動することができます。自動アルゴリズムの起動により、フラッシュメモリの消去/書込みが可能となります。自動アルゴリズムの詳細については、「17.4 自動アルゴリズム起動方法」を参照してください。

## 制限事項

ROMライタ書込み時とはアドレスの与え方およびエンディアンが異なります。このモードではワード(32ビット)長でデータを読み出すことを禁止します。

## フラッシュメモリモード

MD2,1,0端子を”1,1,1”に設定してリセットをかけると、CPUは機能を停止します。このときフラッシュメモリインタフェース回路の機能により、ポート2,3,4,5,6からの信号がフラッシュメモリ本体の制御信号と直結状態になり、外部端子からフラッシュメモリ本体への直接制御が可能になります。イメージとしてフラッシュメモリが単体として外部端子に現れてくるモードであり、主にフラッシュメモリライタを用いた書込み/消去を行う際に設定します。

このモードでは、4Mビットフラッシュメモリの自動アルゴリズムの持つすべてのオペレーションが使用可能です。

## MBM29LV400TCとのフラッシュメモリ制御信号の対応

表 17.3-1に、MBM29LV400TCとのフラッシュメモリ制御信号の対応を示します。

表 17.3-1 MBM29LV400TCとのフラッシュメモリ制御信号の対応

MBM29LV400TC 外部端子	MB91F127/MB91F128 外部制御端子		
	FR-CPUモード	FLASHメモリモード	
		通常オペレーション	VID印加端子
RESET	RSTX	RSTX	MD1
RY/BY	なし (CPUに割込み要求)	RY/BYX	-
BYTE	内部 H 固定	BYTEX	-
WE	内部制御信号 + I/F回路による制御	WEX	-
OE		OEX	MD2
CE		CEX	-
A17 ~ A10	内部アドレスバス	AQ18 ~ AQ11	-
A9		AQ10	MD0
A8 ~ A0		AQ9 ~ AQ1	-
A-1		AQ0	-
DQ15 ~ DQ8	内部データバス	なし	-
DQ7 ~ DQ0		DQ7 ~ DQ0	-

## 17.4 自動アルゴリズム起動方法

フラッシュメモリに対する書込み/消去は、フラッシュメモリ自身を持つ自動アルゴリズムを起動することで行います。

### コマンドオペレーション

自動アルゴリズムを起動するには、フラッシュメモリに対し、1～6回のハーフワード(16ビット)データの連続書込みを実行します。これをコマンドと呼びます。

不正なアドレスとデータを書き込んだり、誤った順番でアドレスとデータを書き込んだりすると、フラッシュメモリは読出しモードにリセットされます。

表 17.4-1に、コマンドの一覧を示します。

表 17.4-1 コマンド一覧

コマンド シーケンス	アクセス 回数	第1書込み サイクル		第2書込み サイクル		第3書込み サイクル		第4書込み /読出し サイクル		第5書込み サイクル		第6書込み サイクル	
		アド レス	デー タ	アド レス	デー タ	アド レス	デー タ	アド レス	デー タ	アド レス	デー タ	アド レス	デー タ
読出し/ リセット	1	XXXXX	XXF0										
読出し/ リセット	4	D5555	XXAA	CAAAB	XX55	D5555	XXF0	(RA)	(RD)				
プログラム	4	D5555	XXAA	CAAAB	XX55	D5555	XXA0	(PA)	(PD)				
チップ消去	6	D5555	XXAA	CAAAB	XX55	D5555	XX80	D5555	XXAA	CAAAB	XX55	D5555	XX10
セクタ消去	6	D5555	XXAA	CAAAB	XX55	D5555	XX80	D5555	XXAA	CAAAB	XX55	(SA)	XX30
セクタ消去一時停止			XXXXX	XXB0									
セクタ消去再開			XXXXX	XX30									
Auto Select	3	D5555	XXAA	CAAAB	XX55	D5555	XX90						
連続モード	3	D5555	XXAA	CAAAB	XX55	D5555	XX20						
連続書込み	2	XXXXX	XXA0	(PA)	(PD)								
連続モード リセット	2	XXXXX	XX90	XXXXX	XXF0 または XX00								

すべてのアドレスとデータは16進数表記です。

(RA): 読出しアドレス

(PA): 書込みアドレス

(SA): セクタアドレス(セクタ内の任意の1アドレスを指定。「表 17.1-1 セクタアドレス一覧」参照。)

(RD): 読出しデータ

(PD): 書込みデータ

消去一時停止コマンド(B0h)と、消去再開コマンド(30h)は、セクタ消去動作中のみ有効です。

2種類の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。

## リード(読出し)/リセットコマンド

タイミングリミット超過から読み出しモードへ復帰するには、読出し/リセットコマンドシーケンスを発行することで行います。読出しサイクルでフラッシュメモリからデータを読み出します。

フラッシュメモリは、ほかのコマンドが入力されるまで、読出し状態を保ちます。

フラッシュメモリは、電源投入時自動的に読出し/リセットにセットされます。この場合は、データ読出しにコマンドは必要ありません。

## プログラム(書込み)

CPUプログラミングモードでは、ハーフワード単位を基本として書込みを行います。書込みは4回のバス動作で行われます。コマンドシーケンスには二つの“アンロック”サイクルがあり、書込みセットアップコマンドと書込みデータサイクルが続きます。そして、最後の書込みサイクルでメモリへの書込みが開始されます。

自動書込みアルゴリズムコマンドシーケンス実行後は、フラッシュメモリはそれ以上の外部よりの制御を必要としません。フラッシュメモリは、自動的に内部で作られた適切な書込みパルスを発生し、書き込まれたセルのマージンを検証します。自動書込み動作は、データポーリング機能により、ビット7のデータがこのビットに書き込んだデータに一致したとき終了し(「17.5 自動アルゴリズム実行状態」のハードウェアシーケンスフラグ参照)、このときをもって読出しモードに戻り、これ以上書込みアドレスは受け付けなくなります。この結果、フラッシュメモリはこの時点で次の有効アドレスを要求します。このようにデータポーリングは、メモリが書込み中であることを示します。

書込み中はフラッシュメモリに書き込みされたすべてのコマンドが無視されます。もし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。

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。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により、素子が不良と判定されるかまたは見かけ上データ“1”が書き込まれたように見えるかのどちらかです。しかし、リセット/読出しモード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”データにすることができます。

図 17.4-1に、書込みコマンドを使った書込み手順を示します。

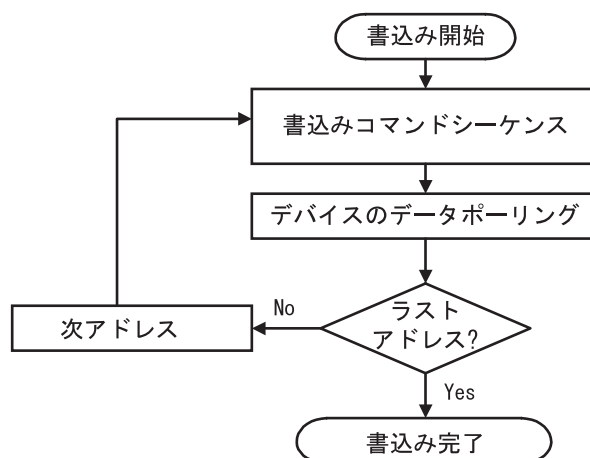


図 17.4-1 書込みコマンドを使った書込み手順

### チップ消去

チップ消去(全セクタの一括消去)は、6回のアクセスで行われます。まず最初に、二つの”アンロック”サイクルがあり、引き続き”セットアップ”コマンドが書き込まれます。チップ消去コマンドまでに、さらに二つの”アンロック”サイクルが続けられます。

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”のパターンに書き込んで検証します(プリプログラム)。この動作中には、フラッシュメモリは外部からの制御は必要としません。

自動消去はコマンドシーケンス中の書込みで開始され、ビット7が”1”になったときに終了し、このときフラッシュメモリは読出しモードに戻ります。チップ消去時間は「セクタ消去時間×全セクタ数+チップ書込み時間(プリプログラム)」となります。

図 17.4-2に、チップ消去コマンドを使ったチップ消去手順を示します。

### セクタ消去

セクタ消去は、6回のアクセスで行われます。二つの”アンロック”サイクルがあり、引き続き”セットアップ”コマンドを書き込みし、その後さらに二つの”アンロック”サイクルが続き、6サイクル目にセクタ消去コマンドを入力することによりセクタ消去が始まります。最後のセクタ消去コマンドの書込みから50 μsのタイムアウト期間中、次のセクタ消去コマンドの受け付けが可能です。

複数のセクタ消去は前述したような六つのバスサイクルを書き込むことで同時に受け付け可能となります。このシーケンスは同時に消去するセクタのアドレスにセクタ消去コマンド(30H)を引き続き書き込むことで行います。最後のセクタ消去コマンドの書込みから50 μsのタイムアウト期間終了により、セクタ消去が開始されます。すなわち、複数のセクタを同時に消去する場合には、次の消去セクタをそれぞれ50 μs以内に入力する必要があるため、それ以降ではコマンドは受け付けられないことがあります。引き続きセクタ消去コマンドが有効かどうかはビット3にてモニタ可能です(「17.5 自動アルゴリズム実行状態」のハードウェアシーケンスフラグ参照)。

タイムアウト中のセクタ消去コマンドまたは消去一時停止以外のいかなるコマンドも読出しにリセットし、その前のコマンドシーケンスは無視します。この場合は、そのセクタを再度消去することにより消去が完了します。セクタ消去バッファへのセクタアドレス入力、セクタのどのような組合せや数(0~6)からでも実行可能です。

セクタ消去では、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。フラッシュメモリは自動的に消去されるセクタ内のすべてのセルに書込みを行います(プリプログラム)。また、セクタ消去中はほかの消去されないセクタは何の影響も受けません。これらの動作中は、フラッシュメモリは外部からの制御は必要としません。

自動セクタ消去は、最後のセクタ消去コマンドの書込みから50 μsのタイムアウト期間の後に開始され、ビット7のデータが”1”になったとき(「17.5 自動アルゴリズム実行状態」のハードウェアシーケンスフラグ参照)終了し、フラッシュメモリは読出しモードに戻ります。ほかのコマンドは無視されます。データボーリングは消去されたセクタ内のどのアドレスでも働きます。複数セクタ消去時間は「(セクタ消去時間+セクタ書込み時間(プリプログラム))×消去セクタ数」となります。

図 17.4-2に、チップ消去コマンドを使ったチップ消去手順を示します。



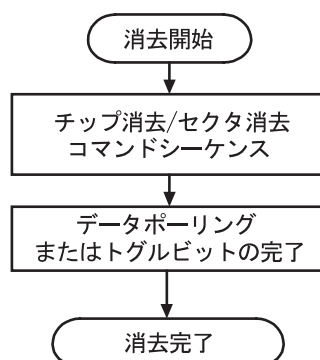


図 17.4-2 チップ消去コマンドを使ったチップ消去手順

## 消去一時停止

消去一時停止コマンドは、ユーザがセクタ消去中にフラッシュメモリの自動アルゴリズムを一時停止して、消去中でないセクタからのデータ読出しと書込みを可能にするものです。このコマンドはセクタ消去中のみ有効で、チップ消去や書込みの動作中は無視されます。消去一時停止コマンド(BOH)はセクタ消去コマンド(30H)後のセクタ消去タイムアウト期間を含むセクタ消去動作中にのみ有効です。このコマンドがタイムアウト期間中に入力されると直ちにタイムアウトを終了し、消去動作を中断します。消去再開コマンドが書き込まれると消去動作が再開されます。消去一時停止、消去再開コマンドの入力の際のアドレスは任意であってかまいません。

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、フラッシュメモリが消去動作を停止するのに最大20  $\mu$ sの時間がかかります。フラッシュメモリが消去一時停止モードに入ると、レディ/ビジー出力とビット7が"1"を出力し、ビット6がトグル動作をやめます。消去しているセクタのアドレスを入力し、ビット6とビット7の読出し値をモニタすることによって、消去動作を停止しているかどうかを確かめられます。さらに、消去一時停止コマンドの書込みは無視されます。

消去動作が停止したとき、フラッシュメモリは消去一時停止読出しモードになります。このモードでのデータの読出しはデータが消去一時停止していないセクタに有効となりますが、それ以外は標準的な読出しと同じです。消去一時停止読出し中、その消去一時停止したセクタからの連続的な読出しに対しては、ビット2はトグル動作をします(「17.5 自動アルゴリズム実行状態」のハードウェアシーケンスフラグ参照)。

消去一時停止読出しモードに入った後、ユーザは書込みのコマンドシーケンスを書き込むことによりフラッシュメモリに書込みができます。この書込みモードは、消去一時停止書込みモードとなります。このモードでの書込みは、データが消去一時停止していないセクタに有効となりますが、それ以外は通常のバイト書込みと同じです。消去一時停止書込みモード中、その消去一時停止したセクタからの連続的な読出しに対しては、ビット2はトグル動作をします。消去一時停止ビット(ビット6)によって検出できます。

使用上の注意として、ビット6はどんなアドレスに対しても読出し可能ですが、ビット7は書込みアドレスに対して読出しを行わなければなりません。

セクタ消去動作を再開するためには、再開コマンド(30H)を入力する必要があります。この時点でさらに再開コマンドを入力しても無視されます。他方、消去一時停止コマンドはフラッシュメモリが消去再開した後に入力することができます。

# 17.5 自動アルゴリズム実行状態

この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フラッシュメモリ外部に知らせるハードウェアを持ちます。一つはレディ/ビジー信号、もう一つはハードウェアシーケンスフラグです

## レディ/ビジー信号 (RDY/BUSYX)

フラッシュメモリは内部の自動アルゴリズムが実行中か終了したかを知らせる手段として、ハードウェアシーケンスフラグのほかに、レディ/ビジー信号を持っています。このレディ/ビジー信号はフラッシュメモリインタフェース回路に接続され、フラッシュメモリステータスレジスタの”RDY”ビットとして読み出すことが可能です。また、このレディ/ビジー信号の立ち上がりにより、CPUに対して割込み要求を発生することも可能です。(「17.1 フラッシュメモリの概要」を参照してください)。

“RDY”ビットの読出し値が”0”のとき、フラッシュメモリは書込みまたは消去動作中です。このときは書込みコマンドも消去コマンドも受け付けません。”RDY”ビットの読出し値が”1”のときフラッシュメモリは読出し/書込みまたは消去動作待ちの状態です。

## 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、自動アルゴリズム実行中にフラッシュメモリの任意のアドレス(バイトアクセス時は奇数アドレス)を読み出すことによりデータとして得られます。データのうち、有効ビットは5ビットで、それぞれが自動アルゴリズムの状態を示します。

図 17.5-1に、ハードウェアシーケンスフラグの構成を示します。

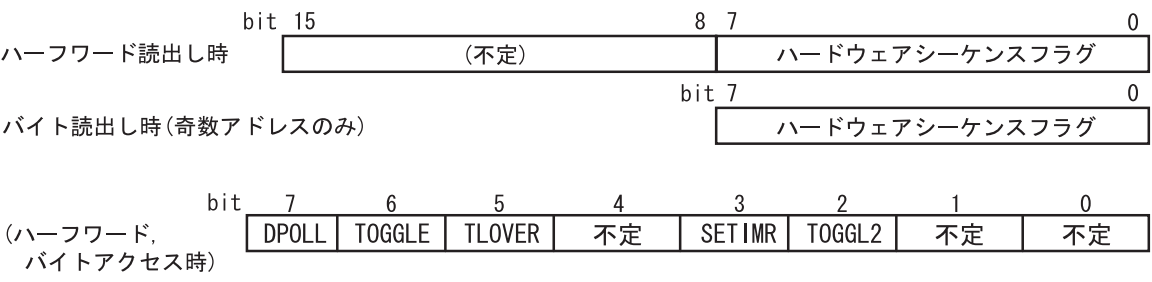


図 17.5-1 ハードウェアシーケンスフラグの構成

なお、これらのフラグについてはFR-CPU ROMモード時に意味を持ちません。必ずFR-CPUプログラミングモードのみで、ハーフワードまたはバイト読出しを行ってください。

表 17.5-1に、ハードウェアシーケンスフラグ状態一覧を示します。

表 17.5-1 ハードウェアシーケンスフラグ状態一覧

状態			DPOLL	TOGGLE	TLOVER	SETIMR	TOGGL2
実行中	自動書込み動作		反転データ	トグル	0	0	1
	自動消去動作		0	トグル	0	1	トグル
	消去一時停止モード	消去一時停止/読出し (消去一時停止しているセクタ)	1	1	0	0	トグル <sup>*1</sup>
		消去一時停止/読出し (消去一時停止していないセクタ)	データ	データ	データ	データ	データ
		消去一時停止/書込み (消去一時停止していないセクタ)	反転データ	トグル <sup>*2</sup>	0	0	1 <sup>*3</sup>
タイムリミット超過	自動書込み動作		反転データ	トグル	1	0	1
	自動消去動作		0	トグル	1	1	未定義
	消去一時停止時の書込み動作		0	トグル	1	1	未定義

\*1: 消去一時停止セクタからの連続的な読出し時にビット2はトグル動作をします。

\*2: どんなアドレスからの連続的な読出し時でも、ビット6はトグル動作をします。

\*3: 消去一時停止書込み中、その書込みされているアドレスの読出し時は、ビット2は“1”となります。  
しかし、消去一時停止しているセクタからの連続的な読出し時は、ビット2はトグル動作をします。

以下に、各ビットの説明します。

#### 【ビット7】 DPOLL: データポーリング

##### ● 自動書込み動作中

自動書込みアルゴリズム実行中に読出しを行うと、フラッシュメモリはビット7に最後に書き込まれたデータ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読出しアクセスを行うと、フラッシュメモリはアドレス信号の指し示す番地の読出しデータのビット7を出力します。

##### ● 自動消去動作中

自動消去アルゴリズム実行中に読出しを行うと、フラッシュメモリはアドレス信号の指し示す番地によらず“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中

セクタ消去一時停止中に読出しを行うと、フラッシュメモリはアドレス信号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、アドレス信号の指し示す番地の読み出し値のビット7を出力します。

後述のビット6のトグルビットとともに参照することで、現在セクタ消去一時停止状態にあるか否か、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 >

自動アルゴリズムの動作が終了に近づくとき、ビット7(データポーリング)は読み出し動作中、非同期的に変化します。これはフラッシュメモリが動作状態の情報をビット7に送り出し、そして確定したデータをその次に送り出すことを意味します。フラッシュメモリが自動アルゴリズムを終了したとき、また、ビット7が確定データを出力しているときでも、ほかのビットはまだ不確定です。ほかのビットの確定データは、連続した読出しの実行によって読み出されます。

## 【ビット6】 TOGGLE: トグルビット

## ● 自動書込み/消去動作中

自動書込みまたは消去アルゴリズム実行中に連続した読出しを行うと、フラッシュメモリは“1”と“0”をトグルする結果をビット6に出力します。自動書込みまたは消去アルゴリズムが終了すると、連続した読出しに対してビット6のトグル動作をやめ、有効なデータを出力します。

トグルビットは各コマンドシーケンスの差以後の書込みサイクルの後から有効になります。

なお、書込みの際、書き込もうとしたセクタが書き換え保証されているセクタの場合は、約2  $\mu$ sの間トグル動作した後、データに書き換えることなくトグル動作を終わります。消去の際、もし選択されたすべてのセクタが書込み保証されている場合には、トグルビットは約100  $\mu$ sトグル動作をし、その後データを書き換えないで読出しモードに戻ります。

## ● セクタ消去一時停止中

セクタ消去一時停止中に読出しを行うと、フラッシュメモリはアドレス信号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、アドレス信号の指し示す番地の読み出し値のビット6を出力します。

## 【ビット5】 TLOVER: タイミングリミット超過

## ● 自動書込み/消去動作中

ビット5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示します。この状態においてビット5は“1”を出力します。すなわち自動アルゴリズムが動作中で、このフラグが“1”を出力した場合は、書込みまたは消去が失敗したことを示します。

ビット5はまた、消去することなく非ブランクの部分に書き込もうとすると、フェイルとなります。この場合、ビット7(データポーリング)から確定データを読むことができず、またビット6(トグルビット)はトグルしたままとなります。この状態でタイムリミットを超えると、ビット5に“1”が出力されます。この場合は、フラッシュメモリが不良なのではなく、正しく使用されなかったということを表していることに注意してください。もし、この状態が発生したときには、リセットコマンドを実行してください。

## 【ビット3】 SETIMR:セクタ消去タイマ

## ● セクタ消去動作中

最初のセクタ消去コマンドシーケンス実行後,セクタ消去ウェイト期間中になります。ビット3は,この期間中"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。データポーリングとトグルビットは最初のセクタ消去コマンドシーケンスの実行後から有効となります。

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おり,続けてのコマンドライトはデータポーリングかトグルビットが消去の終了を示すまで無視されます(消去一時停止コードの入力のみ受け付けます)。このフラグが"0"の場合,フラッシュメモリは追加のセクタ消去コードの書き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き込みに先立ってソフトウェアで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,追加セクタ消去コードは受け付けられていない可能性があります。

セクタ消去一時停止中に読出しを行うと,フラッシュメモリはアドレス信号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,アドレス信号の指し示す番地の読み出し値のビット3を出力します。

## 【ビット2】 TOGGL2:トグルビット2

## ● セクタ消去動作中

このトグルビットは,ビット6のトグルビットに加えて,フラッシュメモリが自動消去動作中であるか,消去一時停止中であるかを検出することに使われます。自動消去中に消去しているセクタから連続して読み出しを行うとビット2がトグル動作をします。フラッシュメモリが消去一時停止読み出しモードならば,消去一時停止しているセクタから連続して読出しを行うとビット2はトグル動作をします。

フラッシュメモリが消去一時停止書き込みモードのときは,消去一時停止していないセクタからアドレスを連続して読み出すとビット2は"1"が読み出されます。ビット6はビット2と違い,通常書き込み,消去または消去一時停止書き込み動作中にのみトグル動作をします。

例えば,ビット2とビット6は,消去一時停止読み出しモードを検出するために,一緒に使われます(ビット2はトグル動作をするが,ビット6はトグル動作をしない)。さらにビット2は消去しているセクタの検出にも使われます。フラッシュメモリが消去動作のときは,ビット2は消去しているセクタからの読み出しならばトグル動作をします。

## ハードウェアシーケンスフラグの使用例

先に延べたハードウェアシーケンスフラグを用いることで、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。例として図 17.5-2、図 17.5-3に、書き込み/消去判定のフローチャートを、それぞれデータポーリング機能を用いた場合と、トグルビット機能を用いた場合について示します。

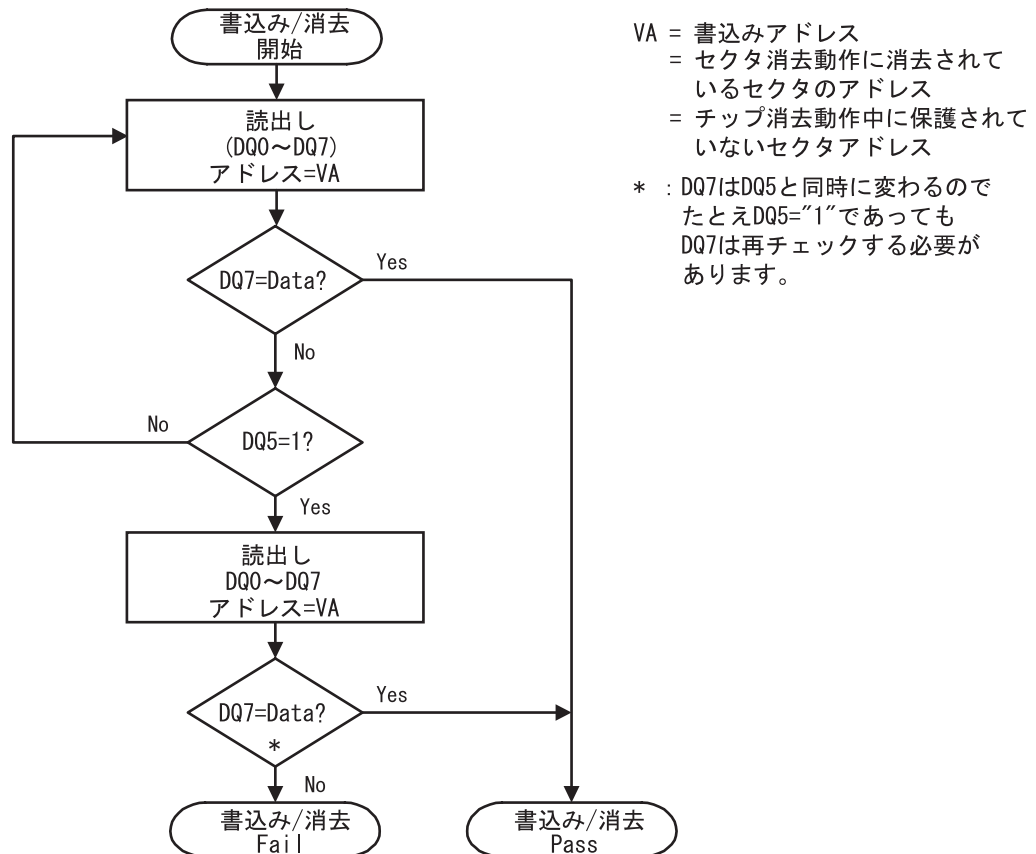
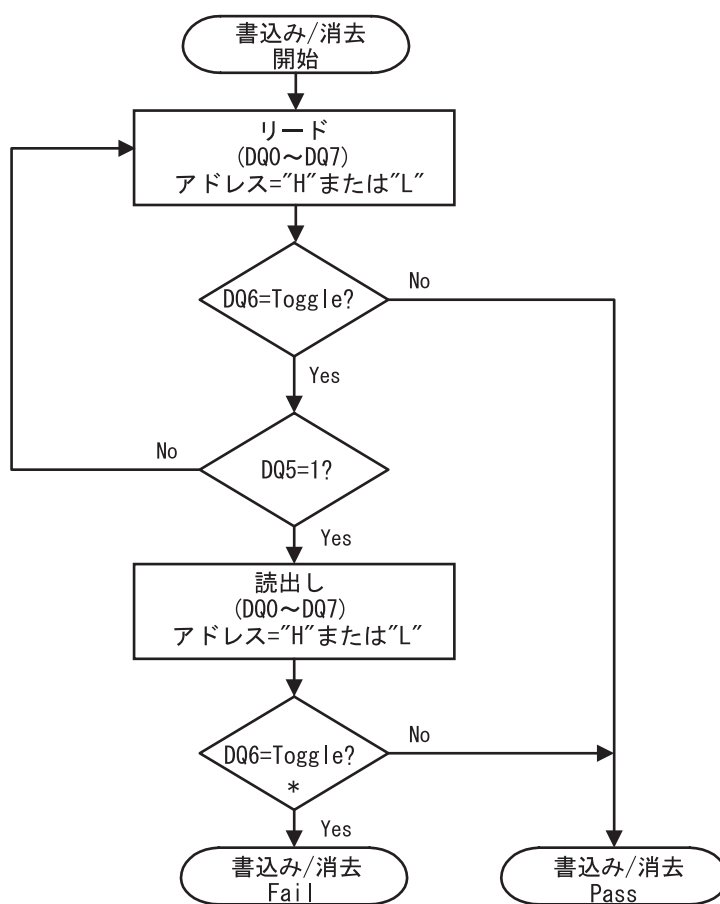


図 17.5-2 データポーリング機能を用いた書き込み/消去判定のフローチャート



\* : DQ5が"1"に変わるのと同時に, DQ6はトグリングをやめるので,  
たとえDQ5="1"であっても, DQ6は再チェックする必要があります。

図 17.5-3 トグルビット機能を用いた書込み/消去判定のフローチャート

## 17.6 セクタプロテクトオペレーション

このフラッシュメモリには不正な書込み/消去に対し、セクタ単位でこれを無効にするセクタ保護機能を持ちます。一度保護設定されたセクタはデバイスが破壊しない限り、その設定を持ち続けますが、一時的であれば保護を解除し、書込み/消去を行うことも可能です。これらの操作はセクタプロテクトオペレーションを通して行われます。

セクタプロテクトオペレーションには書込み/消去のような自動アルゴリズムは持ちません。また、通常モードには対応せず、フラッシュメモリモードでのみ実行可能です。このため主にフラッシュメモリライターを用いた外部端子制御で行われるべきものです。

### セクタプロテクトオペレーション一覧

セクタプロテクトオペレーションには、

- イネーブルセクタプロテクト
- ベリファイセクタプロテクト
- セクタプロテクト一時解除

の3種類があります。

表 17.6-1に、それぞれの端子設定を示します。

表 17.6-1 端子設定

オペレーション	CEX	OEX	WEX	AQ1	AQ2	AQ7	AQ13 ~ AQ18	DQ0 ~ DQ15	RSTX	MD2	MD1	MD0
イネーブル セクタプロテクト	L	H	L	L	H	L	セクタ アドレス	-	H	V <sub>ID</sub>	H	V <sub>ID</sub>
ベリファイ セクタプロテクト	L	L	H	L	H	L	セクタ アドレス	コード 出力	H	H	H	V <sub>ID</sub>
セクタプロテクト 一時解除	-	-	-	-	-	-	-	-	H	H	V <sub>ID</sub>	H



### イネーブルセクタプロテクト

イネーブルセクタプロテクトでは、フラッシュメモリ内部の保護回路への書込みを行います。

このオペレーションにより7個のセクタのどのセクタの組合せでも書込みも消去も無効にすることができます。なお、MB91F127/MB91F128ではすべてのセクタが保護解除された状態で出荷されます。

このオペレーションでは、まずアドレス信号に保護すべきセクタのセクタアドレス(AQ18,AQ17,AQ16,AQ15,AQ14,AQ13)の設定と,AQ8="0"の設定が必要となります。セクタとセクタアドレスの対応は、表 17.1-1を参照ください。

保護回路の書込みは、MD2とMD0にVID(=12V)を印加し、CEX="0"にした後、WEXパルスの立下がり開始され、立上りで終了します。なお、セクタアドレスはWEXパルスの間中一定に保たなければなりません。セクタ保護は一度設定されると、取り消すことができません。また、保護されたセクタへは、これ以降書込み/消去は不可能となります。

### ベリファイセクタプロテクト

ベリファイセクタプロテクトでは、フラッシュメモリ内部の保護回路の書込みに対する検証(ベリファイ)を行います。このオペレーションでは、まずCEXとOEXを"0"にし、WEXを"1"にしたままMD0にVIDを印加(マージンモード)します。(AQ7,AQ2,AQ1)=(0,1,0)の条件で、アドレス信号をあるセクタアドレスにして読み出すと、保護されたセクタでは出力DQ0に"1"が出力されます。保護されていないセクタでは、00Hが読み出されます。

図 17.6-1に、イネーブルセクタプロテクト、ベリファイセクタプロテクトを用いたセクタ保護のアルゴリズムを示します。

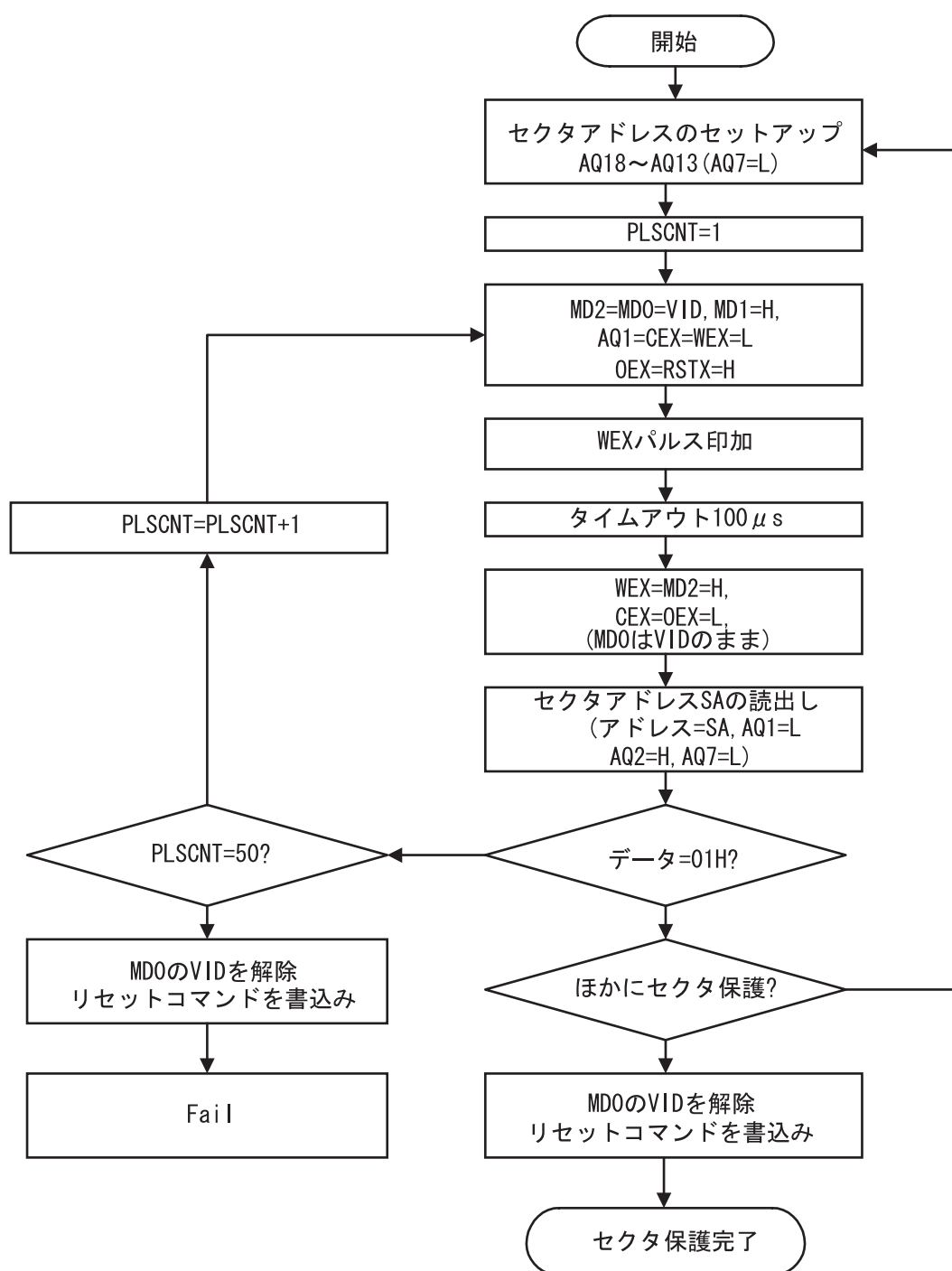


図 17.6-1 イネーブルセクタプロテクトおよびベリファイセクタプロテクトを用いたセクタ保護アルゴリズム

## セクタプロテクト一時解除

イネーブルセクタプロテクトにより保護されたセクタはデバイスが破壊しない限り、書き込み/消去ができませんが、セクタプロテクト一時解除のオペレーションでは、以前に設定されたセクタ保護情報の一時的解除を可能にします。このオペレーションは、MD1にVIDを印加し続けることで設定されます。この間、以前に設定されたセクタ保護情報は無視され、すべてのセクタに書き込み/消去が可能となります。

MD1を"1"(=5V)に戻すと、このオペレーションは解除され、以前に保護されたすべてのセクタが再び保護されます。

図 17.6-2に、セクタプロテクト一時解除のアルゴリズムを示します。

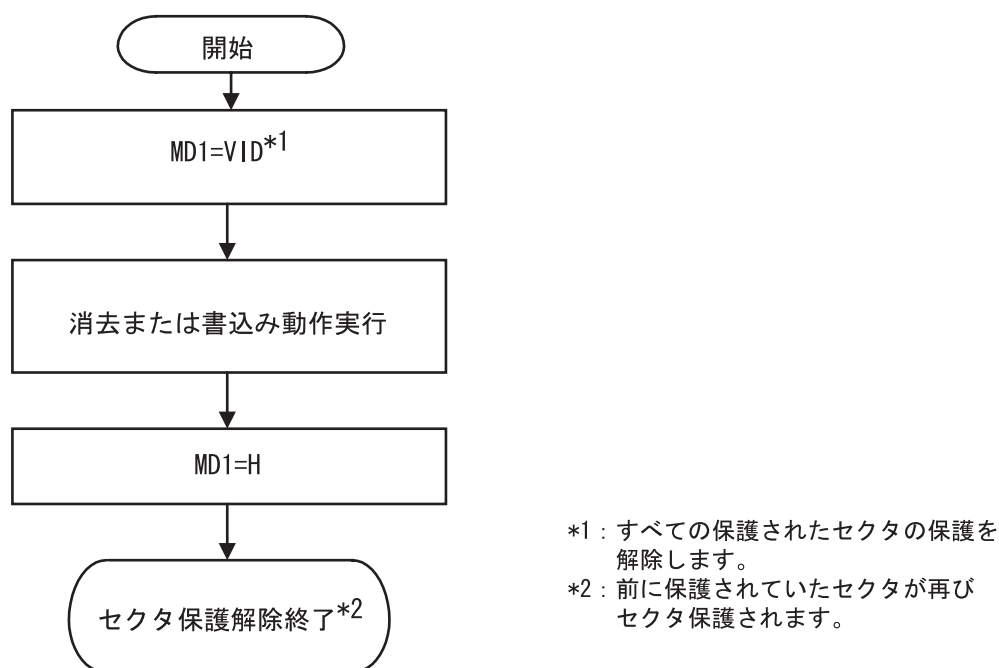


図 17.6-2 セクタプロテクト一時解除のアルゴリズム

## 付録

---

付録として、I/Oマップ、割込みベクタ、CPUステートでの端子状態について記載しています。

---

- A I/Oマップ
- B 割込みベクタ
- C 各CPUステートにおける端子状態

## A I/Oマップ

MB91F127/128に内蔵された周辺機器の各レジスタは、付表 A-1に示すアドレスが割当てられます。

## I/Oマップ

[ 表の見方 ]

アドレス	レジスタ				内部リソース
	+0	+1	+2	+3	
000000 <sub>H</sub>	PDR0[R/W] XXXXXXXX	PDR1[R/W] XXXXXXXX	PDR2[R/W] XXXXXXXX	PDR3[R/W] XXXXXXXX	ポートデータレジスタ

読出し/書込みアトリビュート  
 リセット後のレジスタ初期値  
 レジスタ名 (1コラムのレジスタが4n番地, 2コラムが4n+2番地…)  
 最左のレジスタ番地 (ワードでアクセスした際は, 1コラム目のレジスタがデータのMSB側となる。)

**< 注意事項 >**

レジスタのビット値は、以下のように初期値を表します。

“1”:初期値“1”

"0":初期値"0"

“X”:初期値“X”

"-": その位置に物理的にレジスタがない

付表 A-1 I/Oマップ (続く)

アドレス	レジスタ				内部リソース
	+0	+1	+2	+3	
000000 <sub>H</sub>	PDR3 [R/W] XXXXXXXX	PDR2 [R/W] XXXXXXXX	-	-	ポートデータ レジスタ
000004 <sub>H</sub>	PDR7 [R/W] -----X	PDR6 [R/W] XXXXXXXX	PDR5 [R/W] XXXXXXXX	PDR4 [R/W] XXXXXXXX	
000008 <sub>H</sub>	-	PDR4 [R/W] -XXXXXXX	-	PDR8[R/W] --XXXXXX	
00000C <sub>H</sub>	-				
000010 <sub>H</sub>	-	-	PDRE [R/W] XXXXXXXX	PDRF [R/W] ----XXXX	
000014 <sub>H</sub>	PDRG [R/W] XXXXXXXX	-	-	PDRJ [R/W] XXXXXXXX	
000018 <sub>H</sub>	-	-		--	未使用
00001C <sub>H</sub>	SSR [R/W] 00001-00	SIDR [R] SODR [W] XXXXXXXX	SCR [R/W] 00000100	SMR [R/W] 00--0-00	UART0
000020 <sub>H</sub>	SSR [R/W] 00001-00	SIDR [R] SODR [W] XXXXXXXX	SCR [R/W] 00000100	SMR [R/W] 00--0-00	UART1
000024 <sub>H</sub>	SSR [R/W] 00001-00	SIDR [R] SODR [W] XXXXXXXX	SCR [R/W] 00000100	SMR [R/W] 00--0-00	UART2
000028 <sub>H</sub>	TMRLR [W] XXXXXXXX XXXXXXXX		TMR [R] XXXXXXXX XXXXXXXX		リロードタイム0
00002C <sub>H</sub>	-		TMCSR [R/W] ----0000 00000000		
000030 <sub>H</sub>	TMRLR [W] XXXXXXXX XXXXXXXX		TMR [R] XXXXXXXX XXXXXXXX		リロードタイム1
000034 <sub>H</sub>	-		TMCSR [R/W] ----0000 00000000		
000038 <sub>H</sub>	-		-		未使用
00003C <sub>H</sub>	TMRLR [W] XXXXXXXX XXXXXXXX		TMR [R] XXXXXXXX XXXXXXXX		リロードタイム2
000040 <sub>H</sub>	-		TMCSR [R/W] ----0000 00000000		
000044 <sub>H</sub>	IPCP1[R] XXXXXXXX XXXXXXXX		IPCP0[R] XXXXXXXX XXXXXXXX		多機能タイマ
000048 <sub>H</sub>	IPCP3[R] XXXXXXXX XXXXXXXX		IPCP2[R] XXXXXXXX XXXXXXXX		
00004C <sub>H</sub>	-	ICS23[R/W] 00000000	-	ICS01[R/W] 00000000	
000050 <sub>H</sub>	ADCR [R] 000000XX XXXXXXXX		ADCS [R/W] 00000000 00000000		A/Dコンバータ (逐次比較)
000054 <sub>H</sub>	OCCP1[R/W] XXXXXXXX XXXXXXXX		OCCP0[R/W] XXXXXXXX XXXXXXXX		多機能タイマ
000058 <sub>H</sub>	OCCP3[R/W] XXXXXXXX XXXXXXXX		OCCP2[R/W] XXXXXXXX XXXXXXXX		
00005C <sub>H</sub>	-		-		未使用
000060 <sub>H</sub>	-		-		
000064 <sub>H</sub>	OCS2,3[R/W] XXX00000 0000XX00		OCS0,1[R/W] XXX00000 0000XX00		多機能タイマ

付表 A-1 I/Oマップ (続く)

アドレス	レジスタ				内部リソース
	+0	+1	+2	+3	
000068 <sub>H</sub>	-	-	-	-	未使用
00006C <sub>H</sub>	TCDT [R/W] 00000000	00000000	TCCS [R/W] 0-----	00000000	フリーランタイム 多機能タイマ
000070 <sub>H</sub>	-	-	-	-	未使用
000074 <sub>H</sub>	-	-	-	-	未使用
000078 <sub>H</sub>	UTIM [R] UTIMR [W] 00000000	00000000	-	UTIMC[R/W] 0--00001	U-Timer0
00007C <sub>H</sub>	UTIM [R] UTIMR [W] 00000000	00000000	-	UTIMC[R/W] 0--00001	U-Timer1
000080 <sub>H</sub>	UTIM [R] UTIMR [W] 00000000	00000000	-	UTIMC[R/W] 0--00001	U-Timer2
000084 <sub>H</sub>	-	-	-	-	未使用
000088 <sub>H</sub>	-	-	-	-	
00008C <sub>H</sub>	-	-	-	-	
000090 <sub>H</sub>	-	-	-	-	
000094 <sub>H</sub>	EIRR [R/W] 00000000	ENIR [R/W] 00000000	-	-	外部割込み
000098 <sub>H</sub>	EHVR [R/W] 00000000	ELVR [R/W] 00000000	-	-	
00009C <sub>H</sub>	-	-	-	-	未使用
0000A0 <sub>H</sub>	-	-	-	-	
0000A4 <sub>H</sub>	-	-	-	-	
0000A8 <sub>H</sub>	-	-	-	-	
0000AC <sub>H</sub>	-	-	-	-	
0000B0 <sub>H</sub>	-	-	-	-	
0000B4 <sub>H</sub>	-	-	-	-	
0000B8 <sub>H</sub>	-	-	-	-	
0000BC <sub>H</sub>	-	-	-	-	
0000C0 <sub>H</sub>	-	-	-	-	
0000C4 <sub>H</sub>	-	-	-	-	
0000C8 <sub>H</sub>	-	-	-	-	
0000CC <sub>H</sub>	-	-	-	-	
0000D0 <sub>H</sub>	-	-	DDRE [W] 00000000	DDRF [W] ----0000	ポート方向 レジスタ
0000D4 <sub>H</sub>	-	AIC3[R/W] 11111111	-	-	A/Dコンバ - タ
0000D8 <sub>H</sub>	DDRJ [W] 00000000	-	-	DDRJ [W] 00000000	ポート方向 レジスタ
0000DC <sub>H</sub>	GCN1 [R/W] 00110010	00010000	-	GCN2[R/W] 00000000	PPG
0000E0 <sub>H</sub>	PTMRO [R] 11111111	11111111	PCSR0 [W] XXXXXXX	XXXXXXX	PPG0
0000E4 <sub>H</sub>	PDUT0 [W] XXXXXXX	XXXXXXX	PCNH0[R/W] 0000000	PCNL0[R/W] 00000000	
0000E8 <sub>H</sub>	PTMR1 [R] 11111111	11111111	PCSR1 [W] XXXXXXX	XXXXXXX	PPG1
0000EC <sub>H</sub>	PDUT1 [W] XXXXXXX	XXXXXXX	PCNH1[R/W] 0000000-	PCNL1[R/W] 00000000	

付表 A-1 I/Oマップ (続く)

アドレス	レジスタ				内部リソース
	+0	+1	+2	+3	
0000F0 <sub>H</sub>	PTMR2 [R] 11111111 11111111		PCSR2 [W] XXXXXXXX XXXXXXXX		PPG2
0000F4 <sub>H</sub>	PDUT2 [W] XXXXXXXX XXXXXXXX		PCNH2[R/W] 0000000-	PCNL2[R/W] 00000000	
0000F8 <sub>H</sub>	PTMR3 [R] 11111111 11111111		PCSR3 [W] XXXXXXXX XXXXXXXX		PPG3
0000FC <sub>H</sub>	PDUT3 [W] XXXXXXXX XXXXXXXX		PCNH3[R/W] 0000000-	PCNL3[R/W] 00000000	
000100 <sub>H</sub> ~ 0001FC <sub>H</sub>	-				未使用
000200 <sub>H</sub>	DPDP [R/W] ----- ----- ----- 00000000				DMAコントローラ
000204 <sub>H</sub>	DACS [R/W] 00000000 00000000 00000000 00000000				
000208 <sub>H</sub>	DATCR [R/W] ----- --XX0000 --XX0000 --XX0000				
00020C <sub>H</sub>	-				
000210 <sub>H</sub> ~ 0002FC <sub>H</sub>	-				未使用
000300 <sub>H</sub> ~ 0003EC <sub>H</sub>	-				未使用
0003F0 <sub>H</sub>	BSD0 [W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				ビットサーチ モジュール
0003F4 <sub>H</sub>	BSD1 [R/W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0003F8 <sub>H</sub>	BSDC [W] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0003FC <sub>H</sub>	BSRR [R] XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000400 <sub>H</sub>	ICR00 [R/W] ---11111	ICR01 [R/W] ---11111	ICR02 [R/W] ---11111	ICR03 [R/W] ---11111	割込み コントローラ
000404 <sub>H</sub>	ICR04 [R/W] ---11111	ICR05 [R/W] ---11111	ICR06 [R/W] ---11111	ICR07 [R/W] ---11111	
000408 <sub>H</sub>	ICR08 [R/W] ---11111	ICR09 [R/W] ---11111	ICR10 [R/W] ---11111	ICR11 [R/W] ---11111	
00040C <sub>H</sub>	ICR12 [R/W] ---11111	ICR13 [R/W] ---11111	ICR14 [R/W] ---11111	ICR15 [R/W] ---11111	
000410 <sub>H</sub>	ICR16 [R/W] ---11111	ICR17 [R/W] ---11111	ICR18 [R/W] ---11111	ICR19 [R/W] ---11111	
000414 <sub>H</sub>	ICR20 [R/W] ---11111	ICR21 [R/W] ---11111	ICR22 [R/W] ---11111	ICR23 [R/W] ---11111	
000418 <sub>H</sub>	ICR24 [R/W] ---11111	ICR25 [R/W] ---11111	ICR26 [R/W] ---11111	ICR27 [R/W] ---11111	
00041C <sub>H</sub>	ICR28 [R/W] ---11111	ICR29 [R/W] ---11111	ICR30 [R/W] ---11111	ICR31 [R/W] ---11111	
000420 <sub>H</sub>	ICR32 [R/W] ---11111	ICR33 [R/W] ---11111	ICR34 [R/W] ---11111	ICR35 [R/W] ---11111	
000424 <sub>H</sub>	ICR36 [R/W] ---11111	ICR37 [R/W] ---11111	ICR38 [R/W] ---11111	ICR39 [R/W] ---11111	
000428 <sub>H</sub>	ICR40 [R/W] ---11111	ICR41 [R/W] ---11111	ICR42 [R/W] ---11111	ICR43 [R/W] ---11111	



付表 A-1 I/Oマップ (続き)

アドレス	レジスタ				内部リソース
	+0	+1	+2	+3	
00042C <sub>H</sub>	ICR44 [R/W] ---11111	ICR45 [R/W] ---11111	ICR46 [R/W] ---11111	ICR47 [R/W] ---11111	割込み コントローラ
000430 <sub>H</sub>	DICR [R/W] -----0	HRCL [R/W] ---11111	-	-	遅延割込み
000434 <sub>H</sub> ~ 00047C <sub>H</sub>	-				未使用
000480 <sub>H</sub>	RSRR [R] WTCR [W] 1XXXX-00	STCR [R/W] 000111--	PDDR [R/W] ----0000	CTBR [W] XXXXXXXX	クロック制御部
000484 <sub>H</sub>	GCR [R/W] 110011-1	WPR [W] XXXXXXXX	-	-	
000488 <sub>H</sub>	PTCR [R/W] 00--0---	-			PLL制御部
00048C <sub>H</sub> ~ 0005FC <sub>H</sub>	-				未使用
000600 <sub>H</sub>	DDR3 [W] 00000000	DDR2 [W] 00000000	-	-	データ方向 レジスタ
000604 <sub>H</sub>	DDR7 [W] -----0	DDR6 [W] 00000000	DDR5 [W] 00000000	DDR4 [W] 00000000	
000608 <sub>H</sub>	-	DDRA [W] -0000000	-	DDR8 [W] --000000	
00060C <sub>H</sub>	ASR1 [W] 00000000 00000001		AMR1 [W] 00000000 00000000		バス インタフェース
000610 <sub>H</sub>	ASR2 [W] 00000000 00000010		AMR2 [W] 00000000 00000000		
000614 <sub>H</sub>	ASR3 [W] 00000000 00000011		AMR3 [W] 00000000 00000000		
000618 <sub>H</sub>	ASR4 [W] 00000000 00000100		AMR4 [W] 00000000 00000000		
00061C <sub>H</sub>	ASR5 [W] 00000000 00000101		AMR5 [W] 00000000 00000000		
000620 <sub>H</sub>	AMD0 [W] ---00111	AMD1 [R/W] 0--00000	AMD32 [R/W] 00000000	AMD4 [R/W] 0--00000	
000624 <sub>H</sub>	AMD5 [R/W] 0--00000	DSCR [W] 00000000	RFCR [R/W] --XXXXXX 00---000		
000628 <sub>H</sub>	EPCRO [W] --1-1100 -1111111		EPCR1 [W] -----1 11111111		
00062C <sub>H</sub>	DMCR4 [R/W] 00000000 0000000-		DMCR5 [R/W] 00000000 0000000-		
000630 <sub>H</sub> ~ 0007BC <sub>H</sub>	-				未使用
0007C0 <sub>H</sub>	FSTR [R/W] 000XXXX0	-	-	-	フラッシュメモリ
0007C4 <sub>H</sub>	FWTC [R/W] XXXXX000	-	-	-	
0007C8 <sub>H</sub> ~ 0007F8 <sub>H</sub>	-				未使用
0007FC <sub>H</sub>	-		LER [W] -----000	MODR [W] XXXXXXXX	リトルエンデアン レジスタ モードレジスタ

## &lt; 注意事項 &gt;

ライトオンリーのビットのあるレジスタに対して, RMW系の命令を行わないでください。

RMW系の命令(RMW: リードモディファイライト)

AND Rj, @Ri	または Rj, @Ri	Eまたは Rj, @Ri
ANDH Rj, @Ri	またはH Rj, @Ri	EまたはH Rj, @Ri
ANDB Rj, @Ri	またはB Rj, @Ri	EまたはB Rj, @Ri
BANDL #u4, @Ri	BまたはL #u4, @Ri	BEまたはL #u4, @Ri
BANDH #u4, @Ri	BまたはH #u4, @Ri	BEまたはH #u4, @Ri

”未使用”または”-”の領域のデータは不定です。

## B 割込みベクタ

付表 B-1に割込みベクタテーブルを示します。

割込みベクタテーブルには,MB91F127/128の割込み要因と割込みベクタ/割込み制御レジスタの割当てが記載されています。

### 割込みベクタ

付表 B-1 割込みベクタ (続く)

割込要因	割込番号		割込レベル <sup>1</sup>	オフセット	TBRデフォルト のアドレス <sup>2</sup>
	10進	16進			
リセット	0	00	-	3FC <sub>H</sub>	000FFFFC <sub>H</sub>
システム予約	1	01	-	3F8 <sub>H</sub>	000FFFF8 <sub>H</sub>
システム予約	2	02	-	3F4 <sub>H</sub>	000FFFF4 <sub>H</sub>
システム予約	3	03	-	3F0 <sub>H</sub>	000FFFF0 <sub>H</sub>
システム予約	4	04	-	3EC <sub>H</sub>	000FFFE4 <sub>H</sub>
システム予約	5	05	-	3E8 <sub>H</sub>	000FFFE8 <sub>H</sub>
システム予約	6	06	-	3E4 <sub>H</sub>	000FFFE4 <sub>H</sub>
システム予約	7	07	-	3E0 <sub>H</sub>	000FFFE0 <sub>H</sub>
システム予約	8	08	-	3DC <sub>H</sub>	000FFFD4 <sub>H</sub>
システム予約	9	09	-	3D8 <sub>H</sub>	000FFFD8 <sub>H</sub>
システム予約	10	0A	-	3D4 <sub>H</sub>	000FFFD4 <sub>H</sub>
システム予約	11	0B	-	3D0 <sub>H</sub>	000FFFD0 <sub>H</sub>
システム予約	12	0C	-	3CC <sub>H</sub>	000FFFC4 <sub>H</sub>
システム予約	13	0D	-	3C8 <sub>H</sub>	000FFFC8 <sub>H</sub>
未定義命令例外	14	0E	-	3C4 <sub>H</sub>	000FFFC4 <sub>H</sub>
NMI 要求	15	0F	15(F <sub>H</sub> ) 固定	3C0 <sub>H</sub>	000FFFC0 <sub>H</sub>
外部割込み0	16	10	ICR00	3BC <sub>H</sub>	000FFB8 <sub>H</sub>
外部割込み1	17	11	ICR01	3B8 <sub>H</sub>	000FFB8 <sub>H</sub>
外部割込み2	18	12	ICR02	3B4 <sub>H</sub>	000FFB4 <sub>H</sub>
外部割込み3	19	13	ICR03	3B0 <sub>H</sub>	000FFB0 <sub>H</sub>
UART 0 受信完了	20	14	ICR04	3AC <sub>H</sub>	000FFAC <sub>H</sub>
UART 1 受信完了	21	15	ICR05	3A8 <sub>H</sub>	000FFA8 <sub>H</sub>
UART 2 受信完了	22	16	ICR06	3A4 <sub>H</sub>	000FFA4 <sub>H</sub>
UART 0 送信完了	23	17	ICR07	3A0 <sub>H</sub>	000FFA0 <sub>H</sub>
UART 1 送信完了	24	18	ICR08	39C <sub>H</sub>	000FF9C <sub>H</sub>
UART 2 送信完了	25	19	ICR09	398 <sub>H</sub>	000FF98 <sub>H</sub>
DMAC 0(終了,エラー)	26	1A	ICR10	394 <sub>H</sub>	000FF94 <sub>H</sub>
DMAC 1(終了,エラー)	27	1B	ICR11	390 <sub>H</sub>	000FF90 <sub>H</sub>
DMAC 2(終了,エラー)	28	1C	ICR12	38C <sub>H</sub>	000FF8C <sub>H</sub>
DMAC 3(終了,エラー)	29	1D	ICR13	388 <sub>H</sub>	000FF88 <sub>H</sub>
DMAC 4(終了,エラー)	30	1E	ICR14	384 <sub>H</sub>	000FF84 <sub>H</sub>
DMAC 5(終了,エラー)	31	1F	ICR15	380 <sub>H</sub>	000FF80 <sub>H</sub>
DMAC 6(終了,エラー)	32	20	ICR16	37C <sub>H</sub>	000FF7C <sub>H</sub>
DMAC 7(終了,エラー)	33	21	ICR17	378 <sub>H</sub>	000FF78 <sub>H</sub>
A/D(逐次型)	34	22	ICR18	374 <sub>H</sub>	000FF74 <sub>H</sub>
リロードタイマ 0	35	23	ICR19	370 <sub>H</sub>	000FF70 <sub>H</sub>
リロードタイマ 1	36	24	ICR20	36C <sub>H</sub>	000FF6C <sub>H</sub>
リロードタイマ 2	37	25	ICR21	368 <sub>H</sub>	000FF68 <sub>H</sub>

付表 B-1 割込みベクタ (続き)

割込要因	割込番号		割込レベル <sup>*1</sup>	オフセット	TBRデフォルト のアドレス <sup>*2</sup>
	10進	16進			
外部割込み 4	38	26	ICR22	364 <sub>H</sub>	000FFF64 <sub>H</sub>
外部割込み 5	39	27	ICR23	360 <sub>H</sub>	000FFF60 <sub>H</sub>
システム予約	40	28	ICR24	35C <sub>H</sub>	000FFF5C <sub>H</sub>
システム予約	41	29	ICR25	358 <sub>H</sub>	000FFF58 <sub>H</sub>
U-TIMER 0	42	2A	ICR26	354 <sub>H</sub>	000FFF54 <sub>H</sub>
U-TIMER 1	43	2B	ICR27	350 <sub>H</sub>	000FFF50 <sub>H</sub>
U-TIMER 2	44	2C	ICR28	34C <sub>H</sub>	000FFF4C <sub>H</sub>
フラッシュメモリ	45	2D	ICR29	348 <sub>H</sub>	000FFF48 <sub>H</sub>
システム予約	46	2E	ICR30	344 <sub>H</sub>	000FFF44 <sub>H</sub>
システム予約	47	2F	ICR31	340 <sub>H</sub>	000FFF40 <sub>H</sub>
PPG0	48	30	ICR32	33C <sub>H</sub>	000FFF3C <sub>H</sub>
PPG1	49	31	ICR33	338 <sub>H</sub>	000FFF38 <sub>H</sub>
PPG2	50	32	ICR34	334 <sub>H</sub>	000FFF34 <sub>H</sub>
PPG3	51	33	ICR35	330 <sub>H</sub>	000FFF30 <sub>H</sub>
ICU0(取込み)	52	34	ICR36	32C <sub>H</sub>	000FFF2C <sub>H</sub>
ICU1(取込み)	53	35	ICR37	328 <sub>H</sub>	000FFF28 <sub>H</sub>
ICU2(取込み)	54	36	ICR38	324 <sub>H</sub>	000FFF24 <sub>H</sub>
ICU3(取込み)	55	37	ICR39	320 <sub>H</sub>	000FFF20 <sub>H</sub>
OCU0(一致)	56	38	ICR40	31C <sub>H</sub>	000FFF1C <sub>H</sub>
OCU1(一致)	57	39	ICR41	318 <sub>H</sub>	000FFF18 <sub>H</sub>
OCU2(一致)	58	3A	ICR42	314 <sub>H</sub>	000FFF14 <sub>H</sub>
OCU3(一致)	59	3B	ICR43	310 <sub>H</sub>	000FFF10 <sub>H</sub>
システム予約	60	3C	ICR44	30C <sub>H</sub>	000FFF0C <sub>H</sub>
16ビットフリーランタイム	61	3D	ICR45	308 <sub>H</sub>	000FFF08 <sub>H</sub>
システム予約	62	3E	ICR46	304 <sub>H</sub>	000FFF04 <sub>H</sub>
遅延割込み要因ビット	63	3F	ICR47	300 <sub>H</sub>	000FFF00 <sub>H</sub>
システム予約 (REALOSにて使用) <sup>*3</sup>	64	40	-	2FC <sub>H</sub>	000FFEFC <sub>H</sub>
システム予約 (REALOSにて使用) <sup>*3</sup>	65	41	-	2F8 <sub>H</sub>	000FEF8 <sub>H</sub>
INT命令で使用	66 ~ 255	42 ~ FF	-	2F4 <sub>H</sub> ~ 000 <sub>H</sub>	000FEF4 <sub>H</sub> ~ 000FFC00 <sub>H</sub>

\*1 ICRは、割込みコントローラ内に設けられたレジスタで、割込みの各要求に対する割込みレベルを設定します。

ICRは、割込み要求の各々に対応して用意されています。

\*2 TBRは、EIT用ベクタテーブルの先頭アドレスを示すレジスタです。

TBRとEIT要因ごとに決められたオフセット値を加算したアドレスが、ベクタアドレスとなります。

\*3 REALOS/FRをご使用される場合は、システムコード用に0X40, 0X41の割込みを使用します。

#### 参考：

TBRの示すアドレスから1KBの領域が、EIT用ベクタ領域となっています。

1ベクタ当たりの大きさは4バイトで、ベクタ番号とベクタアドレスの関係は下記のように表されます。

$$\begin{aligned}
 \text{Vctadr} &= \text{TBR} + \text{vctofs} \\
 &= \text{TBR} + (3\text{FC}_H - 4 \times \text{vct})
 \end{aligned}$$

vctadr: ベクタアドレス    vctofs: ベクタオフセット    vct : ベクタ番号

## C 各CPUステートにおける端子状態

---

付表 C-1 ~ 付表 C-3に各CPUステートにおける端子状態を示します。

なお,外部バスモード時とシングルチップモード時では,リセット時の端子状態が異なりますので注意してください。

---

### 端子状態一覧表の用語説明

付表 C-1 ~ 付表 C-3中の端子の状態として使用している語句は,以下の意味を持ちます。

- 入力可  
入力機能が使用可能な状態であることを意味します。
- 入力0固定  
端子からすぐの入力ゲートで,外部入力を遮断して内部へ"0"を伝えている状態です。
- 出力Hi-Z  
端子駆動用トランジスタを駆動禁止状態にし,端子をハイインピーダンスにすることを意味します。
- 出力保持  
このモードになる直前に出力していた状態をそのまま出力することを意味します。すなわち,出力のある内蔵周辺が動作中であれば,その内蔵周辺に従い出力を行い,ポートなどとして出力している場合には,その出力を保持します。
- 直前の状態を保持  
このモードになる直前に出力していた状態をそのまま出力または入力であれば入力可能を意味します。

## 端子状態一覧表

付表 C-1 シングルチップモード時の端子状態

端子名	機能	スリープ時	ストップ時		リセット時
			HIZX=0	HIZX=1	
P20 ~ P27	ポート	直前の状態保持	直前の状態保持	出力Hi-z 入力0固定	出力Hi-x , 全端子入力可
P30 ~ P37					
P40 ~ P47					
P50 ~ P57					
P60 ~ P65					
P66	IN2				
P67	IN3				
P70	FRCK				
	TC12				
P80	ポート				
P81	IN0				
P82	IN1				
P83 ~ P86	ポート				
PA0 ~ PA2					
PA3	S01				
PA4	S11				
PA5	SC1				
PA6	ポート				
PG0	OCPA0				
PG1	OCPA1				
PG2	OCPA2				
PG3	OCPA3				
PG4	OC0				
PG5	OC1				
PG6	OC2				
PG7	OC3				
PE0	INT0		入力可	入力可	
PE1	INT1				
PE2	INT2				
PE3	INT3				
PE4	INT4				
	TC11				
PE5	INT5				
	SC0				
PE6	S10				
PE7	S00				
PF0	TC10				
PF1	S12				
PF2	S02				
PF3	SC2				
	ATG				
PJ0 ~ PJ7	AN0 ~ AN7				

付表 C-2 外部バス16ビットモード時の端子状態

端子名	機能	スリープ時	ストップ時		バス開放 (BGRNT)	リセット時		
			HIZX=0	HIZX=1				
P20 ~ P27	D16 ~ D23	出力保持	同左	出力Hi-z 入力0固定	出力Hi-z	出力Hi-x , 全端子入力可		
---	D24 ~ D31	または Hi-z						
---	A00 ~ A15	出力保持 (アドレス出力)						
P60 ~ P67	A16 ~ A23	P : 直前状態保持 F : アドレス出力	同左					
P70	A24	直前状態保持	同左					
P80	RDY	P : 直前状態保持 F : RDY入力	P,F : 直前状態保持				P : 直前状態保持 F : RDY入力 L出力	
P81	BGRNT	P : 直前状態保持 F : H出力	P,F : 直前状態保持					
P82	BRQ	P : 直前状態保持 F : BRQ出力	P,F : 直前状態保持					
---	RD	直前状態保持	同左				BRQ入力	H出力
---	WRO							
P85	WR1	P : 直前状態保持 F : H出力	P,F : 直前状態保持	H出力				
P86	ALE	P : 直前状態保持 F : L出力	P,F : 直前状態保持	L出力				
PA0	CS0	直前状態保持	H出力	L出力				
PA1	CS1	P : 直前状態保持 F : CS出力	P : 同左 F : H出力	H出力				
PA2	CS2							
PA3	CS3							
PA4	CS4							
PA5	CS5							
PA6	CLK	P : 直前状態保持 F : CLK出力	P,F : 直前状態保持	CLK出力	CLK出力			
PG0	OCPA0	直前状態保持	直前の状態保持	直前の状態保持				
PG1	OCPA1							
PG2	OCPA2							
PG3	OCPA3							
PG4	OC0							
PG5	OC1							
PG6	OC2							
PG7	OC3							
PE0	INT0					入力可	入力可	
PE1	INT1							
PE2	INT2							
PE3	INT3							
PE4	INT4							
	TCI1							
PE5	INT5							
	SC0							
	SIO							
PE6	S00							
PF0	TCI0							
PF1	S12							
PF2	S02							
PF3	SC2							
	ATG							
PJ0 ~ PJ7	ANO ~ AN7							

P: 汎用ポート指定時

F: 指定機能選択時

付表 C-3 外部バス8ビットモード時の端子状態

端子名	機能	スリープ時	ストップ時		バス開放 (BGRNT)	リセット時		
			HIZX=0	HIZX=1				
P20 ~ P27	ポート	直前状態保持	同左	出力Hi-z 入力0固定	直前の状態保持 出力Hi-z	出力Hi-x, 全端子入力可		
---	D24 ~ D31	出力保持 または Hi-z	同左					
---	A00 ~ A15	出力保持 (アドレス出力)	同左					
P60 ~ P67	A16 ~ A23	P: 直前状態保持 F: アドレス出力	同左					
P70	A24	直前状態保持	同左					
P80	RDY	P: 直前状態保持 F: RDY入力	P,F: 直前状態保持		P: 直前状態保持 F: RDY入力			
P81	BGRNT	P: 直前状態保持 F: H出力	P,F: 直前状態保持		L出力			
P82	BRQ	P: 直前状態保持 F: BRQ出力	P,F: 直前状態保持		BRQ入力			
---	RD	直前状態保持	同左			H出力		
---	WR0							
P85	ポート	直前状態保持	同左		直前の状態保持			
P86	ALE	P: 直前状態保持 F: L出力	P,F: 直前状態保持			L出力		
PA0	CS0	直前状態保持	H出力			L出力		
PA1	CS1	P: 直前状態保持 F: CS出力	P: 同左 F: H出力			H出力		
PA2	CS2							
PA3	CS3							
PA4	CS4							
PA5	CS5							
PA6	CLK	P: 直前状態保持 F: CLK出力	P,F: 直前状態保持		CLK出力	CLK出力		
PG0	OCPA0	直前状態保持	直前の状態保持		直前の状態保持			
PG1	OCPA1							
PG2	OCPA2							
PG3	OCPA3							
PG4	OC0							
PG5	OC1							
PG6	OC2							
PG7	OC3							
PE0	INT0		入力可	入力可				
PE1	INT1							
PE2	INT2							
PE3	INT3							
PE4	INT4							
	TCI1							
PE5	INT5							
	SC0							
PE6	SIO							
PE7	S00							
PF0	TCI0							
PF1	S12							
PF2	S02							
PF3	SC2							
	ATG							
PJ0 ~ PJ7	AN0 ~ AN7							

P: 汎用ポート指定時

F: 指定機能選択時





# 索引

<数字>

## 0

0検出	
0検出 .....	305
ビットサーチモジュール .....	305
0検出データレジスタ	
0検出用データレジスタ(BSD0) .....	303

## 1

16ビットアウトプットコンペア	
16ビットアウトプットコンペアのタイミング .....	205
16ビットアウトプットコンペアの動作説明 .....	204
16ビットインプットキャプチャ	
16ビットインプットキャプチャのタイミング .....	206
16ビットインプットキャプチャの動作 ...	206
16ビットタイマレジスタ	
16ビットタイマレジスタ(TMR)のビット構成 .....	165
16ビットフリーランタイム	
16ビットフリーランタイムのカウントタイミ .....	203
16ビットフリーランタイムの動作説明 ...	202
16ビットフリーランタイムのクリアタイミン .....	203
16ビットリロードタイマ	
16ビットリロードタイマの概要 .....	160
16ビットリロードタイマのレジスタ一覧 .....	161
16ビットリロードタイマを使用して起動する .....	188
場合 .....	167
アンダフロー動作 .....	168
外部イベントカウント .....	167
内部クロック動作 .....	167
入力端子機能 .....	168
16ビットリロードレジスタ	
16ビットリロードレジスタ(TMRLR)のビット構 .....	166
1検出	
1検出 .....	305
ビットサーチモジュール .....	305
1検出データレジスタ	
1検出用データレジスタ(BSD1) .....	304

<アルファベット>

## A

ADCR	
データレジスタ(ADCR)の構成 .....	245
ADCS	
コントロールステータスレジスタ(ADCS)の .....	241
ビット構成 .....	241
コントロールステータスレジスタ(ADCS)の .....	241
ビット詳細 .....	241
A/Dコンバータ	
A/Dコンバータ(逐次変換型) .....	3
A/Dコンバータ使用上の注意 .....	252
A/Dコンバータのその他の使用上の注意 .....	252
A/Dコンバータの特長 .....	238
A/Dコンバータのレジスタ .....	240
単発変換モード .....	248
停止変換モード .....	249
ブロックダイアグラム .....	239
変換データ保護機能 .....	250
連続変換モード .....	248
AIC	
アナログ入力制御レジスタ(AIC)の構成 .....	247
AMD0	
エリアモードレジスタ0(AMD0)の構成 ....	104
エリアモードレジスタ0(AMD0)のビット機能 .....	104
AMD1	
エリアモードレジスタ1(AMD1)の構成 ....	106
エリアモードレジスタ1(AMD1)のビット機能 .....	106
AMD32	
エリアモードレジスタ32(AMD32)の構成 .....	107
エリアモードレジスタ32(AMD32)のビット機能 .....	107
AMD4	
エリアモードレジスタ4(AMD4)の構成 ....	108
エリアモードレジスタ4(AMD4)のビット機能 .....	108
AMD5	
エリアモードレジスタ5(AMD5)の構成 ....	109
エリアモードレジスタ5(AMD5)のビット機能 .....	109
AMR	
エリア選択レジスタ(ASR)とエリアマスケレ .....	101
ジスタ(AMR)構成 .....	101

ASR	エリア選択レジスタ(ASR)とエリアマスクレジスタ(AMR)構成 .....	101
<b>B</b>		
BSD0	0検出用データレジスタ(BSD0) .....	303
BSD1	1検出用データレジスタ(BSD1) .....	304
BSDC	変化点検出用データレジスタ(BSDC) .....	304
BSRR	検出結果レジスタ(BSRR) .....	304
BUSYX	レディ/ビジー信号(RDY/BUSYX) .....	323
<b>C</b>		
CLK同期モード	CLK同期モードの転送データフォーマット .....	266
CTBR	タイムベースタイマクリアレジスタ(CTBR)のレジスタ構成 .....	64
<b>D</b>		
DACSR	DMAコントロールステータスレジスタ(DACSR)のビット構成 .....	280
	DMAコントロールステータスレジスタ(DACSR)のビット詳細 .....	280
DATCR	DMA端子コントロールレジスタ(DATCR)のビット構成 .....	282
	DMA端子コントロールレジスタ(DATCR)のビット詳細 .....	282
DDR	データ方向レジスタ(DDR)の構成 .....	154
DICR	DICRのDLYIビット .....	224
	遅延割込み制御レジスタ(DICR)の構成 .....	223
DLYIビット	DICRのDLYIビット .....	224
DMA	DMAコントローラの特長 .....	276
	DMA転送要求要因一覧表 .....	276
DMAC	DMAC(DMAコントローラ) .....	4
	DMAコントロールステータスレジスタ(DACSR)のビット構成 .....	280
	DMAコントロールステータスレジスタ(DACSR)のビット詳細 .....	280
	DMA端子コントロールレジスタ(DATCR)のビット構成 .....	282
	DMA端子コントロールレジスタ(DATCR)のビット詳細 .....	282
	DMACパラメータディスクリプタポイント(DPDP)の構成 .....	279
DMAコントローラ	DMAC(DMAコントローラ) .....	4
	DMAコントローラ内部レジスタへの転送動作 .....	299
	DMAコントローラのタイミング図 .....	290
	DMAコントローラの特長 .....	276
	DMAコントローラのブロックダイアグラム .....	277
	DMAコントローラのレジスタ一覧 .....	278
	リソースの割込要求をDMA転送要求として使用する場合 .....	298
	スリープモード中のDMA転送動作 .....	299
	チャンネル間優先順位 .....	298
	内部メモリを外部転送する動作 .....	299
	優先度の高い割込発生時のDMA転送禁止 .....	298
	連続転送について .....	299
DMA転送禁止	優先度の高い割込発生時のDMA転送禁止 .....	298
DMA転送動作	スリープモード中のDMA転送動作 .....	299
DMA転送要求	リソースの割込要求をDMA転送要求として使用する場合 .....	298
DMA要求抑止レジスタ	DMA要求抑止レジスタ(PDRR)のレジスタ構成 .....	63
DMA抑止	DMA抑止回路のブロックダイアグラム .....	77
	DMA抑止の設定 .....	78
DPDP	DMACパラメータディスクリプタポイント(DPDP)の構成 .....	279
<b>E</b>		
EHVR	外部割込み要求レベル設定レジスタ(ELVR,EHVR)の構成 .....	218
EIRR	外部割込み要因レジスタ(EIRR)の構成 .....	217
EIT	EITからの復帰 .....	44
	EITの注意事項 .....	44
	EITの動作 .....	53
	EITの割込みレベル .....	45
	EITベクタテーブル .....	49
	EIT要因 .....	44
	EIT要因受理の優先順位 .....	51
	EIT(例外,割込み,トラップ) .....	44
	特長 .....	44

ELVR	外部割込み要求レベル設定レジスタ (ELVR, EHVR)の構成	218
ENIR	割込み許可レジスタ(ENIR)の構成	216
EPCR0	外部端子制御レジスタ0(EPCR0)の構成 外部端子制御レジスタ0(EPCR0)のビット機能	110 110
EPCR1	外部端子制御レジスタ1(EPCR1)の構成 外部端子制御レジスタ1(EPCR1)のビット機能	113 113

## F

FR CPU	FR CPU	2
FR-CPU	FR-CPU ROMモード(32ビット, 読出しのみ) FR-CPU プログラミングモード(16ビット, 読出し/書込み)	317 317
FSTR	フラッシュメモリステータスレジスタ(FSTR)の構成	314
FWTC	フラッシュメモリウェイトレジスタ(FWTC)の構成	316

## G

GCN1	ジェネラルコントロールレジスタ1(GCN1)のビット構成 ジェネラルコントロールレジスタ1(GCN1)のビット詳細	181 181
GCN2	ジェネラルコントロールレジスタ2(GCN2)のビット構成	183
GCR	ギア制御レジスタ(GCR)のレジスタ構成	65

## H

HRCL	ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の構成	230
HRLC	ホールドリクエスト取下げ要求(HRLC)	233

## I

I/Oポート	外部端子の機能(I/Oポートまたは制御端子)選択	155
ICR	割込み制御レジスタ(ICR)の構成 割込み制御レジスタ(ICR)のビット構成 割込み制御レジスタ(ICR)のマッピング	229 47 47
ICS	インプットキャプチャコントロールレジスタ(ICS01, ICS23)の構成	200
ILM	割込みレベルマスクレジスタ(ILM)	46
INT	INT命令の動作	54
INTE	INTE命令の動作	54
I/O	I/Oポートの基本ブロックダイアグラム I/Oポートのレジスタ I/Oマップ	152 152 334
IPCP	インプットキャプチャデータレジスタ(IPCP0 ~ 3)の構成	199
Iフラグ	Iフラグ	45

## L

LER	リトルエンディアンレジスタ(LER)の構成 リトルエンディアンレジスタ(LER)のビット機能	114 114
LQFP-100P-M05	LQFP-100P-M05の外形寸法図	6

## M

MBM29LV400TC	MBM29LV400TCとのフラッシュメモリ制御信号の対応	318
MD	モード端子(MD0 ~ MD2)について	16
MDH	乗除算結果レジスタ(MDH/MDL)	29
MDL	乗除算結果レジスタ(MDH/MDL)	29
MODR	モードレジスタ(MODR) モードレジスタ(MODR)書込み時の注意点	94 94

## O

OCCP	コンペアレジスタ(OCCP0~3)の構成	196
OCS	アウトプットコントロールレジスタ(OCS0~3)の構成	197

## P

PC	プログラムカウンタ(PC)	27
PCNH	コントロールステータスレジスタ(PCNH, PCNL)のビット機能	175
	コントロールステータスレジスタ(PCNH, PCNL)のビット構成	175
PCNL	コントロールステータスレジスタ(PCNH, PCNL)のビット機能	175
	コントロールステータスレジスタ(PCNH, PCNL)のビット構成	175
PCSR	PPG周期設定レジスタのビット構成	178
PCTR	PLL制御レジスタ(PCTR)のレジスタ構成	69
PDR	ポートデータレジスタ(PDR)の構成	153
PDRR	DMA要求抑止レジスタ(PDRR)のレジスタ構成	63
PDUT	PPGデューティ設定レジスタ(PDUT)のビット構成	179
PLL	PLLのクロック設定例	81
PLL制御レジスタ	PLL制御レジスタ(PCTR)のレジスタ構成	69
PPG周期設定レジスタ	PPG周期設定レジスタ(PCSR)のビット構成	178
PPG出力	PPG出力オール	187
PPGタイマ	PPGタイマ1チャンネル分のブロックダイアグラム	173
	PPGタイマの全体ブロックダイアグラム	173
	PPGタイマの特長	172
	PPGタイマのレジスタ一覧	174
	ソフトウェアによるPPGタイマ複数チャンネル起動	188
PPGタイマレジスタ	PPGタイマレジスタ(PTMR)のビット構成	180
PPGデューティ設定レジスタ	PPGデューティ設定レジスタ(PDUT)のビット構成	179

## PPG動作

PPG動作	184
-------	-----

## PS

プログラムステータスレジスタ(PS)	30
--------------------	----

## PTMR

PPGタイマレジスタ(PTMR)のビット構成	180
------------------------	-----

## R

## RDY

レディ/ビジー信号(RDY/BUSYX)	323
----------------------	-----

## RETI

RETI命令の動作	56
-----------	----

## ROMライター

ROMライターによる書込み	309
---------------	-----

## RP

リターンポインタ(RP)	28
--------------	----

## RSRR

リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)のレジスタ構成	60
-----------------------------------------------	----

## S

## SCR

シリアルコントロールレジスタ(SCR)のビット構成	259
シリアルコントロールレジスタ(SCR)のビット詳細	259

## SIDR

シリアルインプットデータレジスタ(SIDR)の構成	261
---------------------------	-----

## SMR

シリアルモードレジスタ(SMR)のビット構成	257
シリアルモードレジスタ(SMR)のビット詳細	257

## SODR

シリアルアウトプットデータレジスタ(SODR)の構成	261
----------------------------	-----

## SSP

システムスタックポインタ(SSP)	28, 48
-------------------	--------

## SSR

シリアルステータスレジスタ(SSR)のビット構成	262
シリアルステータスレジスタ(SSR)のビット詳細	262

## STCR

スタンバイ制御レジスタ(STCR)のレジスタ構成	62
--------------------------	----

## T

## TBR

テーブルベースレジスタ(TBR)	28, 49
------------------	--------

TCCS	タイマコントロールステータスレジスタ (TCCS)の構成 .....	194
TCDT	データレジスタ(TCDT)の構成 .....	193
TMCSR	コントロールステータスレジスタ(TMCSR)のビット機能 .....	162
	コントロールステータスレジスタ(TMCSR)のビット構成 .....	162
TMR	16ビットタイマレジスタ(TMR)のビット構成 .....	165
TMRLR	16ビットリロードレジスタ(TMRLR)のビット構成 .....	166

## U

UART	UART .....	3
	UARTのクロック選択 .....	264
	UARTの使用例 .....	271
	UARTの動作モード .....	264
	UARTの特長 .....	254
	UARTのレジスタ .....	256
	使用上の注意 .....	271
	ブロックダイアグラム .....	255
	割込み発生とフラグ .....	268
USP	ユーザスタックポインタ(USP) .....	29
UTIM	Uタイマ値レジスタ(UTIM) .....	209
UTIMC	U-TIMER制御レジスタ(UTIMC) .....	210
	U-TIMER制御レジスタ(UTIMC)使用上の注意事項 .....	211
	U-TIMER制御レジスタ(UTIMC)のビット詳細 .....	210
U-TIMER	U-TIMERの概要 .....	208
	U-TIMERのレジスタ一覧 .....	209
	カスケードモード .....	212
	ポーレートとU-TIMERのリロード値の設定例 .....	273
	ポーレートの計算 .....	212
Uタイマ値レジスタ	Uタイマ値レジスタ(UTIM) .....	209
U-TIMER制御レジスタ	U-TIMER制御レジスタ(UTIMC) .....	210
	U-TIMER制御レジスタ(UTIMC)使用上の注意事項 .....	211
	U-TIMER制御レジスタ(UTIMC)のビット詳細 .....	210
UTIMR	リロードレジスタ(UTIMR) .....	209

## W

WPR	ウォッチドッグリセット発生延期レジスタ (WPR)のレジスタ構成 .....	68
WTCR	リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)のレジスタ構成 .....	60

## &lt;五十音&gt;

## あ

アウトプットコントロールレジスタ	
アウトプットコントロールレジスタ (OCS0 ~ 3)	
の構成	197
アセンブラ	
アセンブラのソース例	83
アドレッシング	
ダイレクトアドレッシング領域	20
アナログ入力制御レジスタ	
アナログ入力制御レジスタ (AIC) の構成	247
アンダフロー	
アンダフロー動作	167

## い

イネーブルセクタプロテクト	
イネーブルセクタプロテクト	330
インタバルタイマ	
その他のインタバルタイマ	3
インプットキャプチャコントロールレジスタ	
インプットキャプチャコントロールレジスタ	
(ICS01, ICS23) の構成	200
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	
~ 3) の構成	199

## う

ウェイトサイクル	
ウェイトサイクル	132
ウォッチドッグ	
ウォッチドッグ制御部のブロックダイアグラ	
ム	71
ウォッチドッグタイマの起動方法	71
ウォッチドッグ周期制御レジスタ	
リセット要因レジスタ (RSRR) とウォッチドッ	
グ周期制御レジスタ (WTCR) のレジスタ構	
成	60
ウォッチドッグタイマ	
ウォッチドッグタイマの起動方法	71
ウォッチドッグリセット発生延期レジスタ	
ウォッチドッグリセット発生延期レジスタ	
(WPR) のレジスタ構成	68

## え

エリア選択レジスタ	
エリア選択レジスタ (ASR) とエリアマスケジ	
スタ (AMR) の構成	101
エリアマスケジスタ	
エリア選択レジスタ (ASR) とエリアマスケジ	
スタ (AMR) 構成	101

エリアモードレジスタ0	
エリアモードレジスタ0 (AMD0) の構成	104
エリアモードレジスタ0 (AMD0) のビット機能	104
エリアモードレジスタ1	
エリアモードレジスタ1 (AMD1) の構成	106
エリアモードレジスタ1 (AMD1) のビット機能	106
エリアモードレジスタ32	
エリアモードレジスタ32 (AMD32) の構成	107
エリアモードレジスタ32 (AMD32) のビット機能	107
エリアモードレジスタ4	
エリアモードレジスタ4 (AMD4) の構成	108
エリアモードレジスタ4 (AMD4) のビット機能	108
エリアモードレジスタ5	
エリアモードレジスタ5 (AMD5) の構成	109
エリアモードレジスタ5 (AMD5) のビット機能	109

## お

オール	187
-----	-----

## か

外形寸法図	
LQFP-100P-M05の外形寸法図	6
外部アクセス	
ビッグエンディアンとリトルエンディアンの	
外部アクセスの比較	115
外部イベントカウンタ	
外部イベントカウンタ	168
外部ウェイトサイクル	
外部ウェイトサイクルタイミング	143
外部クロック	
外部クロック使用時の注意について	17
外部端子	
外部端子の機能 (I/Oポートまたは制御端子) 選	
択	155
外部端子制御レジスタ0	
外部端子制御レジスタ0 (EPCR0) の構成	110
外部端子制御レジスタ0 (EPCR0) のビット機能	110
外部端子制御レジスタ1	
外部端子制御レジスタ1 (EPCR1) の構成	113
外部端子制御レジスタ1 (EPCR1) のビット機能	113
外部デバイス	
外部デバイスとの接続例	123, 127
外部転送	
内部メモリを外部転送する動作	299

外部バス	
外部バス動作のプログラム仕様例	148
外部バスリクエスト	132
外部バスアクセス	
外部バスアクセス	120
外部バス動作	
外部バス動作のプログラム例	149
外部リセット	
外部リセット入力	16
外部割込み	
外部割込みの動作	219
外部割込みの動作手順	219
外部割込みのブロックダイヤグラム	214
外部割込みのレジスター一覧	215
外部割込み要求レベル	220
ストップからの復帰	219
外部割込み要因レジスタ	
外部割込み要因レジスタ(EIRR)の構成	217
外部割込み要求レベル設定レジスタ	
外部割込み要求レベル設定レジスタ(ELVR, EHVR)の構成	218
回路形式	
入出力回路形式	12
カウンタ	
カウンタの動作状態	169
カスケードモード	
カスケードモード	212

## き

ギア	
ギア機能の設定	74
ギア制御部のブロックダイヤグラム	73
ギア制御レジスタ	
ギア制御レジスタ(GCR)のレジスタ構成	65
基本ライトサイクル	
基本ライトサイクルの動作タイミング	135
基本リードサイクル	
基本リードサイクルタイミング	133

## く

クロック	
クロック選択方法	147
PLLのクロック設定例	81
UARTのクロック選択	264
クロック系参考図	82
クロック制御	4
クロック制御部	
クロック発生部と制御部のブロックダイヤグラム	59
クロック発生部と制御部のレジスター一覧	58

クロックダブラ機能	
クロックダブラ機能のon/offに伴う注意事項	80
クロックダブラ機能のon/offによる動作周波数の組合せ	80
クロックダブラ機能の起動	79
クロックダブラ機能の停止	79
クロック発生部	
クロック発生部と制御部のブロックダイヤグラム	59
クロック発生部と制御部のレジスター一覧	58

## け

検出結果レジスタ	
検出結果レジスタ(BSRR)	304
原振入力	
電源投入時の原振入力について	18

## こ

コプロセッサ	
コプロセッサエラートラップ	56
コプロセッサ不在トラップ	56
コマンドオペレーション	
コマンドオペレーション	319
コントロールステータスレジスタ	
コントロールステータスレジスタ(PCNH, PCNL)のビット機能	175
コントロールステータスレジスタ(PCNH, PCNL)のビット構成	175
コントロールステータスレジスタ(ADCS)のビット構成	241
コントロールステータスレジスタ(ADCS)のビット詳細	241
コントロールステータスレジスタ(TMCSR)のビット機能	162
コントロールステータスレジスタ(TMCSR)のビット構成	162
コンバータ	
A/Dコンバータ(逐次変換型)	3
コンペアクリアレジスタ	
コンペアクリアレジスタ	193
コンペアレジスタ	
コンペアレジスタ(OCCP0~3)の構成	196

## し

シーケンス	
ホールドリクエスト取下げ要求シーケンス	235



ジェネラルコントロールレジスタ1	
ジェネラルコントロールレジスタ1(GCN1)のビット構成	181
ジェネラルコントロールレジスタ1(GCN1)のビット詳細	181
ジェネラルコントロールレジスタ2	
ジェネラルコントロールレジスタ2(GCN2)のビット構成	183
システムスタックポインタ	
システムスタックポインタ(SSP)	28, 48
自動アルゴリズム	
自動アルゴリズム実行状態	308
自動ウェイトサイクル	
自動ウェイトサイクルタイミング	142
時分割入出力	
時分割入出力指定	98
時分割入出力インタフェース	
時分割入出力インタフェースの動作タイミング	144
時分割入出力インタフェース	132
乗除算結果レジスタ	
乗除算結果レジスタ(MDH/MDL)	29
使用例	
UARTの使用例	271
初期化	
リセットによる初期化	57
シリアルアウトデータレジスタ	
シリアルアウトデータレジスタ(SODR)の構成	261
シリアルインデータレジスタ	
シリアルインデータレジスタ(SIDR)の構成	261
シリアルコントロールレジスタ	
シリアルコントロールレジスタ(SCR)のビット構成	259
シリアルコントロールレジスタ(SCR)のビット詳細	259
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(SSR)のビット構成	262
シリアルステータスレジスタ(SSR)のビット詳細	262
シリアルモードレジスタ	
シリアルモードレジスタ(SMR)のビット構成	257
シリアルモードレジスタ(SMR)のビット詳細	257
シングル/ブロック転送	
シングル/ブロック転送モード	287

## す

水晶発振	
水晶発振回路について	16
スタック	
割込みスタック	48

スタンバイ制御レジスタ	
スタンバイ制御レジスタ(STCR)のレジスタ構成	62
スタンバイモード	
スタンバイモード状態遷移図	92
スタンバイモード(ストップ/スリープ)からの復帰	232
スタンバイモードの動作一覧	85
ステップトレーストラップ	
ステップトレーストラップの動作	55
ストップ	
スタンバイモード(ストップ/スリープ)からの復帰	232
ストップおよびスリープに入れるプログラムの配置アドレス	85
ストップからの復帰	219
ストップ制御部のブロックダイアグラム	86
ストップ状態	
ストップ状態の概要	84
ストップ状態からの復帰	88
ストップ状態への移行	86
スリープ/ストップ状態からの復帰	17
スリープ	
スタンバイモード(ストップ/スリープ)からの復帰	232
ストップおよびスリープに入れるプログラムの配置アドレス	85
スリープ状態からの復帰	91
スリープ状態への移行	90
スリープ制御部のブロックダイアグラム	90
スリープモード中のDMA転送動作	299
スリープ状態	
スリープ状態の概要	84
スリープ/ストップ状態からの復帰	17

## せ

制御信号	
データバス幅と制御信号との関係	115, 116
制御端子	
外部端子の機能(I/Oポートまたは制御端子)選択	155
制御レジスタ	
制御レジスタ	234
セクタ構成	
フラッシュメモリのセクタ構成(MB91F127)	310
フラッシュメモリのセクタ構成(MB91F128)	311

セットタイミング	
モード0の受信動作時割込みフラグのセットタイミング	268
モード0, モード1, モード2の送信動作時割込みフラグのセットタイミング	270
モード1の受信動作時割込みフラグのセットタイミング	269
モード2の受信動作時割込みフラグのセットタイミング	269
セクタプロテクト一時解除	
セクタプロテクト一時解除	332
セクタプロテクトオペレーション	
イネーブルセクタプロテクト	330
セクタプロテクト一時解除	332
セクタプロテクトオペレーション一覧	329
ペリファイセクタプロテクト	330
専用レジスタ	
専用レジスタ一覧	27

## そ

ソフトウェア	
ソフトウェアによるPPGタイマ複数チャネル起動	188

## た

退避	
退避/復帰の処理	306
タイマコントロールステータレジスタ	
タイマコントロールステータスレジスタ(TCCS)の構成	194
タイミング	
基本リードサイクルタイミング	133
タイミング図	
タイミング図で使用している記号の説明	290
タイムベースタイマ	
タイムベースタイマ	72
タイムベースタイマクリアレジスタ	
タイムベースタイマクリアレジスタ(CTBR)のレジスタ構成	64
ダイレクトアドレッシング	
ダイレクトアドレッシング領域	20
多機能タイマ	
多機能タイマの構成	190
多機能タイマの動作説明	201
多機能タイマユニットのレジスタ一覧	192
多機能タイマユニット	
多機能タイマユニットのブロックダイアグラム	191
端子	
NC端子の処理について	16
端子機能説明	8
端子状態一覧表の用語説明	342
端子配列図	7

電源端子の接続( $V_{CC}$ , $V_{SS}$ )について	16
電源投入時の端子状態について	18
未使用入力端子の処理について	16
モード端子(MD0 ~ MD2)について	16
端子状態	
端子状態一覧表	343
端子状態一覧表の用語説明	342
単発変換モード	
単発変換モード	248

## ち

遅延スロット付き	
遅延スロット付き分岐命令	41
遅延スロット付き分岐命令の制限事項	42
遅延スロット付き分岐命令の動作説明	41
遅延スロットなし	
遅延スロットなし分岐命令	43
遅延スロットなし分岐命令の動作説明	43
遅延割込み制御レジスタ	
遅延割込み制御レジスタ(DICR)の構成	223
遅延割込みモジュール	
DICRのDLYIビット	224
遅延割込みモジュールのブロックダイアグラム	222
遅延割込みモジュールのレジスタ一覧	223
割込み番号	224
チップセレクト	
チップセレクト領域	97
チャネル間	
チャネル間優先順位	298
調歩同期モード	
非同期(調歩同期)モードの転送データフォーマット	265

## つ

通常バスアクセス	
通常バスアクセス	132

## て

停止変換モード	
停止変換モード	249
低消費電力	
低消費電力モード	4
ディスクリプタ	
ディスクリプタアクセス部のタイミング図	291
ディスクリプタ先頭ワードの構成	284
ディスクリプタ第3ワードの構成	286
ディスクリプタ第2ワードの構成	286
データアクセス	
データアクセス	36
データ転送	
データ転送部, 16/8ビットデータ	293

データバス幅	
データバス幅	119
データバス幅	126
データバス幅と制御信号との関係	115, 116
データフォーマット	
データフォーマット	118, 125
データ方向レジスタ	
データ方向レジスタ(DDR)の構成	154
データレジスタ	
データレジスタ(ADCR)の構成	245
データレジスタ(TCDT)の構成	193
テーブルベースレジスタ	
テーブルベースレジスタ(TBR)	28, 49
電源投入	
電源投入時について	18
電源投入時の原振入力について	18
電源投入時の端子状態について	18
転送停止	
連続転送モードでの転送停止(いずれか一方のアドレスは固定である場合)16/8ビットデータ	294
連続転送モードでの転送停止(両方のアドレスが変化する場合)16/8ビットデータ	295
転送データ	
CLK同期モードの転送データフォーマット	266
非同期(調歩同期)モードの転送データフォーマット	265
転送動作	
DMAコントローラ内部レジスタへの転送動作	299
転送の終了動作	
転送の終了動作(いずれか一方のアドレスは固定である場合)	296
転送の終了動作タイミング図	296
転送の終了動作(両方のアドレスが変化する場合)	297
転送モード	
シングル/ブロック転送モード	287
バースト転送モード	289
連続転送モード	288
と	
動作周波数	
クロックダブラ機能のon/offによる動作周波数の組合せ	80
動作モード	
UARTの動作モード	264
動作モード	93
特長	
FR CPU	2
その他の特長	4
トラップ	
EIT(例外, 割込み, トラップ)	44

## な

内蔵RAM	
内蔵RAM 14KB	2
内部アーキテクチャ	
特長	23
内部アーキテクチャ	24
内部クロック	
内部クロック動作	167
内部メモリ	
内部メモリを外部転送する動作	299

## に

入出力回路形式	
入出力回路形式	12
入力端子	
入力端子機能	168
未使用入力端子の処理について	16

## は

バースト転送	
バースト転送モード	289
ハードウェア	
ハードウェアシーケンスフラグ	323
ハードウェアシーケンスフラグの使用例	327
ハードウェア構成	
ハードウェア構成	234
割込みコントローラのハードウェア構成	226
ハーフワードアクセス	
ハーフワードアクセス	129
バイトアクセス	
バイトアクセス	130
バイトオーダーリング	
バイトオーダーリング	35
バスアクセス	
通常バスアクセス	132
ビッグエンディアンのバスアクセス	115
リトルエンディアンのバスアクセス	115
バスインタフェース	
バスインタフェースのブロックダイアグラム	99
バスインタフェースのレジスター一覧	100
バスインタフェース	2, 98
バスインタフェースの特長	96
バス権	
バス権解放	146
バス権獲得	146
バスサイズ	
バスサイズ指定	98
パワーオンリセット	
パワーオンリセットの初期化について	18

汎用レジスタ	
汎用レジスタ .....	34

## ひ

ビジー信号	
レディ/ビジー信号 (RDY/BUSYX) .....	323
ビッグエンディアン	
ビッグエンディアンとリトルエンディアンの	
外部アクセスの比較 .....	115
ビッグエンディアンのバスアクセス .....	115
ビットオーダリング	
ビットオーダリング .....	35
ビットサーチモジュール	
退避/復帰の処理 .....	306
ビットサーチモジュール .....	4
ビットサーチモジュールのブロックダイアグラ	
ラム .....	302
ビットサーチモジュールのレジスター一覧	
.....	303
非同期モード	
非同期(調歩同期)モードの転送データフォー	
マット .....	265

## ふ

複数チャネル	
ソフトウェアによるPPGタイマ複数チャネル起	
動 .....	188
復帰	
退避/復帰の処理 .....	306
フラグ	
割込み発生とフラグ .....	268
フラッシュメモリ	
MBM29LV400TCとのフラッシュメモリ制御信号	
の対応 .....	318
ROMライタによる書込み .....	309
自動アルゴリズム実行状態 .....	308
フラッシュメモリ .....	4
フラッシュメモリの概要 .....	308
フラッシュメモリのセクタ構成(MB91F127)	
.....	310
フラッシュメモリのセクタ構成(MB91F128)	
.....	311
フラッシュメモリのブロックダイアグラム	
.....	309
フラッシュメモリのレジスター一覧 .....	313
フラッシュメモリモード .....	318
割込み制御 .....	309
フラッシュメモリウェイトレジスタ	
フラッシュメモリウェイトレジスタ(FWTC)の	
構成 .....	316

フラッシュメモリステータスレジスタ	
フラッシュメモリステータスレジスタ(FSTR)	
の構成 .....	314
プログラムアクセス	
プログラムアクセス .....	36
プログラムカウンタ	
プログラムカウンタ(PC) .....	27
プログラムステータスレジスタ	
プログラムステータスレジスタ(PS) .....	30
ブロックダイアグラム	
16ビットリロードタイマのブロックダイアグラ	
ラム .....	160
A/Dコンバータのブロックダイアグラム	
.....	239
DMAコントローラのブロックダイアグラム	
.....	277
DMA抑止回路のブロックダイアグラム .....	77
I/Oポートの基本ブロックダイアグラム	
.....	152
PPGタイマ1チャンネル分のブロックダイアグラ	
ム .....	173
PPGタイマの全体ブロックダイアグラム	
.....	173
UARTのブロックダイアグラム .....	255
U-TIMERのブロックダイアグラム .....	208
ウォッチドッグ制御部のブロックダイアグラ	
ム .....	71
外部割込みのブロックダイアグラム .....	214
ギア制御部のブロックダイアグラム .....	73
クロック発生部と制御部のブロックダイアグラ	
ラム .....	59
ストップ制御部のブロックダイアグラム	
.....	86
スリープ制御部のブロックダイアグラム	
.....	90
多機能タイマユニットのブロックダイアグラ	
ム .....	191
遅延割込みモジュールのブロックダイアグラ	
ム .....	222
バスインタフェースのブロックダイアグラム	
.....	99
ビットサーチモジュールのブロックダイアグラ	
ラム .....	302
フラッシュメモリのブロックダイアグラム	
.....	309
ブロックダイアグラム .....	5
リセット要因保持回路のブロックダイアグラ	
ム .....	75
割込みコントローラのブロックダイアグラム	
.....	226
分岐命令	
遅延スロット付き分岐命令 .....	41
遅延スロット付き分岐命令の制限事項 .....	42
遅延スロット付き分岐命令の動作説明 .....	41
遅延スロットなし分岐命令 .....	43
遅延スロットなし分岐命令の動作説明 .....	43

## へ

ペリファイセクタプロテクト	
ペリファイセクタプロテクト	330
変化点検出	
ビットサーチモジュール	306
変化点検出	306
変化点検出用データレジスタ	
変化点検出用データレジスタ(BSDC)	304
変換データ	
変換データ保護機能	250

## ほ

ポートデータレジスタ	
ポートデータレジスタ(PDR)の構成	153
ホールドリクエスト取下げ要求	
制御レジスタ	234
ハードウェア構成	234
ホールドリクエスト取下げ要求(HRLC)	233
ホールドリクエスト取下げ要求シーケンス	235
ホールドリクエスト取下げ要求レベル設定レジスタ	
ホールドリクエスト取下げ要求レベル設定レジスタ(HRCL)の構成	230
ポーレート	
ポーレートとU-TIMERのリロード値の設定例	273
ポーレートの計算	212

## ま

マッピング	
割り込み制御レジスタ(ICR)のマッピング	47

## み

未定義命令	
未定義命令例外の動作	55

## め

命令	
命令概要	39
メモリマップ	
FRファミリ共通のメモリマップ	38
MB91F127/128のメモリマップ	37
メモリマップ	21

## も

モード0	
モード0の受信動作時割り込みフラグのセットタイミング	268

モード0, モード1, モード2の送信動作時割り込みフラグのセットタイミング	270
----------------------------------------	-----

モード1	
モード0, モード1, モード2の送信動作時割り込みフラグのセットタイミング	270
モード1の受信動作時割り込みフラグのセットタイミング	269
モード2	
モード0, モード1, モード2の送信動作時割り込みフラグのセットタイミング	270
モード2の受信動作時割り込みフラグのセットタイミング	269
モード端子	
モード端子	93
モードデータ	
モードデータ	93
モードレジスタ	
モードレジスタ(MODR)	94
モードレジスタ(MODR)書込み時の注意点	94

## ゆ

ユーザスタックポインタ	
ユーザスタックポインタ(USP)	29
ユーザ割り込み	
ユーザ割り込みの動作	53
優先順位判定	
優先順位判定	231
優先度の高い割り込み	
優先度の高い割り込み発生時のDMA転送禁止	298

## ら

ライトサイクル	
ライトサイクルタイミング	139
ラッチアップ	
ラッチアップ防止のために	16

## り

リードサイクル	
各モードでのリードサイクルの動作タイミング	137
リード/ライト混在サイクル	
リード/ライト混在サイクルタイミング	141
リセット	
リセットシーケンス	57
リセットによる初期化	57
リセット発生延期方法	72
リセット要因	4, 57

リセット要因保持	
リセット要因保持回路のブロックダイアグラム	75
リセット要因保持の設定	76
リセット要因レジスタ	
リセット要因レジスタ(RSRR)とウォッチドッグ周期制御レジスタ(WTCR)のレジスタ構成	60
リソースの割込要求	
リソースの割込要求をDMA転送要求として使用する場合	298
リターンポイント	
リターンポイント(RP)	28
リトルエンディアン	
ビッグエンディアンとリトルエンディアンの外部アクセスの比較	115
リトルエンディアンの概要	124
リトルエンディアンのバスアクセス	115
リトルエンディアンレジスタ	
リトルエンディアンレジスタ(LER)の構成	114
リトルエンディアンレジスタ(LER)のビット機能	114
領域	
チップセレクト領域	97
リロードタイマ	
リロードタイマ	3
リロード値	
ポーレートとU-TIMERのリロード値の設定例	273
リロードレジスタ	
リロードレジスタ(UTIMR)	209

## れ

例外	
EIT(例外, 割込み, トラップ)	44
レジスタ一覧	
16ビットリロードタイマのレジスタ一覧	161
DMAコントローラのレジスタ一覧	278
クロック発生部と制御部のレジスタ一覧	58
専用レジスタ一覧	27
レディ信号	
レディ/ビジー信号(RDY/BUSYX)	323
レベルマスク	
割込みに対するレベルマスク	46
連続転送	
連続転送について	299
連続転送モード	288
連続転送モードでの転送停止(いずれか一方のアドレスは固定である場合)16/8ビットデータ	294

連続転送モードでの転送停止タイミング図	294
連続転送モードでの転送停止(両方のアドレスが変化する場合)16/8ビットデータ	295
連続変換モード	
連続変換モード	248

## わ

ワードアクセス	
ワードアクセス	128
割込み	
EITの割込みレベル	45
EIT(例外, 割込み, トラップ)	44
割込み	186
割込みスタック	48
割込み制御	309
割込みに対するレベルマスク	46
割込みベクタ	340
割込み許可レジスタ	
割込み許可レジスタ(ENIR)の構成	216
割込みコントローラ	
優先順位判定	231
割込みコントローラ	3
割込みコントローラの主要機能	226
割込みコントローラのハードウェア構成	226
割込みコントローラのレジスタ一覧	227
割込み制御レジスタ	
割込み制御レジスタ(ICR)のビット構成	47
割込み制御レジスタ(ICR)の構成	229
割込み制御レジスタ(ICR)のマッピング	47
割込み発生	
割込み発生とフラグ	268
割込み番号	
割込み番号	224
割込みフラグ	
モード0の受信動作時割込みフラグのセットタイミング	268
モード0, モード1, モード2の送信動作時割込みフラグのセットタイミング	270
モード1の受信動作時割込みフラグのセットタイミング	269
モード2の受信動作時割込みフラグのセットタイミング	269
割込み要因	
割込み要因の解除	232
割込みレベルマスクレジスタ	
割込みレベルマスクレジスタ(ILM)	46
ワンショット動作	
ワンショット動作	185



CM71-10115-1

---

**富士通半導体デバイス・CONTROLLER MANUAL**

FR30

32ビット マイクロコントローラ

MB91F127/128

ハードウェアマニュアル

---

2002年02月 初版発行

発 行 **富士通株式会社** 電子デバイス事業本部

編 集 技術標準部 技術情報開発部

---



FUJITSU



\* C M 7 1 - 1 0 1 1 5 - 1 \*

富士通半導体デバイス

FR30 32ビット・マイクロコントローラ MB91F127/128

ハードウェアマニュアル