サイプレスはインフィニオン テクノロジーズになりました

この表紙に続く文書には「サイプレス」と表記されていますが、これは同社が最初にこの製品を開発したからです。新規および既存のお客様いずれに対しても、引き続きインフィニオンがラインアップの一部として当該製品をご提供いたします。

文書の内容の継続性

下記製品がインフィニオンの製品ラインアップの一部として提供されたとしても、それを理由としてこの文書に変更が加わることはありません。今後も適宜改訂は行いますが、変更があった場合は文書の履歴ページでお知らせします。

注文時の部品番号の継続性

インフィニオンは既存の部品番号を引き続きサポートします。ご注文の際は、データシート記載の注文部品番号をこれまで通りご利用下さい。

FR ファミリ FR81S, CY91570 シリーズ

32 ビット・マイクロコントローラ ハードウェアマニュアル

Document Number: 002-05575 Rev. *B

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

© Cypress Semiconductor Corporation, 2011-2019. 本書面は，Cypress Semiconductor Corporation 及び Spansion LLC を含むその子会社（以下「Cypress」という。）に帰属する財産である。本書面（本書面に含まれ又は言及されているあらゆるソフトウェア若しくはファームウェア（以下「本ソフトウェア」という。）を含む）は，アメリカ合衆国及び世界のその他の国における知的財産法令及び条約に基づき Cypress が所有する。Cypress はこれらの法令及び条約に基づく全ての権利を留保し，本段落で特に記載されているものを除き，その特許権，著作権，商標権又はその他の知的財産権のライセンスを一切許諾しない。本ソフトウェアにライセンス契約書が伴っておらず，かつ Cypress との間で別途本ソフトウェアの使用方法を定める書面による合意がない場合，Cypress は，(1) 本ソフトウェアの著作権に基づき，(a) ソースコード形式で提供されている本ソフトウェアについて，Cypress ハードウェア製品と共に用いるためにのみ，かつ組織内部でのみ，本ソフトウェアの修正及び複製を行うこと，並びに (b) Cypress のハードウェア製品ユニットに用いるためにのみ，(直接又は再販売者及び販売代理店を介して間接のいずれかで) 本ソフトウェアをバイナリーコード形式で外部エンドユーザーに配布すること，並びに (2) 本ソフトウェア (Cypress により提供され，修正がなされていないもの) が抵触する Cypress の特許権のクレームに基づき，Cypress ハードウェア製品と共に用いるためにのみ，本ソフトウェアの作成，利用，配布及び輸入を行うことについての非独占的で譲渡不能な一身専属的ライセンス（サブライセンスの権利を除く）を付与する。本ソフトウェアのその他の使用，複製，修正，変換又はコンパイルを禁止する。

適用される法律により許される範囲内で，Cypress は，本書面又はいかなる本ソフトウェア若しくはこれに伴うハードウェアに関しても，明示又は黙示をとわず，いかなる保証（商品性及び特定の目的への適合性の黙示の保証を含むがこれらに限られない）も行わない。いかなるコンピューティングデバイスも絶対に安全ということはない。従って，Cypress のハードウェアまたはソフトウェア製品に講じられたセキュリティ対策にもかかわらず，Cypress は，Cypress 製品への権限のないアクセスまたは使用といったセキュリティ違反から生じる一切の責任を負わない。加えて，本書面に記載された製品には，エラッタと呼ばれる設計上の欠陥またはエラーが含まれている可能性があり，公表された仕様とは異なる動作をする場合がある。適用される法律により許される範囲内で，Cypress は，別途通知することなく，本書面を変更する権利を留保する。Cypress は，本書面に記載のある，いかなる製品若しくは回路の適用又は使用から生じる一切の責任を負わない。本書面で提供されたあらゆる情報（あらゆるサンプルデザイン情報又はプログラムコードを含む）は，参照目的のためのみに提供されたものである。この情報で構成するあらゆるアプリケーション及びその結果としてのあらゆる製品の機能性及び安全性を適切に設計，プログラム，かつテストすることは，本書面のユーザーの責任において行われるものとする。Cypress 製品は，兵器，兵器システム，原子力施設，生命維持装置若しくは生命維持システム，蘇生用の設備及び外科的移植を含むその他の医療機器若しくは医療システム，汚染管理若しくは有害物質管理の運用のために設計され若しくは意図されたシステムの重要な構成部分としての使用，又は装置若しくはシステムの不具合が人身傷害，死亡若しくは物的損害を生じさせるようなその他の使用（以下「本目的外使用」という。）のためには設計，意図又は承認されていない。重要な構成部分とは，それの不具合が装置若しくはシステムの不具合を生じさせるか又はその安全性若しくは実効性に影響すると合理的に予想できるような装置若しくはシステムのあらゆる構成部分をいう。Cypress 製品のあらゆる本目的外使用から生じ，若しくは本目的外使用に関連するいかなる請求，損害又はその他の責任についても，Cypress はその全部又は一部をとわず一切の責任を負わず，かつ Cypress はそれら一切から本書により免除される。Cypress は Cypress 製品の本目的外使用から生じ又は本目的外使用に関連するあらゆる請求，費用，損害及びその他の責任（人身傷害又は死亡に基づく請求を含む）から免責補償される。

Cypress，Cypress のロゴ，Spansion，Spansion のロゴ及びこれらの組み合わせ，WICED，PSoC，CapSense，EZ-USB，F-RAM，及び Traveo は，米国及びその他の国における Cypress の商標又は登録商標である。Cypress のより完全な商標のリストは，cypress.com を参照すること。その他の名称及びブランドは，それぞれの権利者の財産として権利主張がなされている可能性がある。

はじめに

サイプレス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。

本シリーズをご利用になる前に、本書および『CY91570シリーズデータシート』をご一読ください。

本書の目的と対象読者

本シリーズは自動車用，産業用制御などの用途向けに設計した､サイプレス32ビットマイクロコントローラです。FRファミリと互換のFR81S CPUを搭載しています。FR81S CPUは命令パイプラインの強化，ロードストア処理の強化，内部バス転送の効率化により、FRファミリ最高性能を実現します。

自動車用アプリケーション制御として最適な仕様となっています。

本書は、実際に本シリーズを使用して製品を開発される技術者を対象に、機能や動作，使い方について解説しています。

商標

FRは、サイプレスの製品です。

REALOS, SOFTUNEは、サイプレスの登録商標です。

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

サンプルプログラムおよび開発環境

FR81Sファミリの周辺機能を動作させるためのサンプルプログラムを無償で提供しております。また、本シリーズで使用する開発環境も掲載しています｡当社マイコンの動作仕様や使用方法の確認などにお役立てください。

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

本書の使い方

機能の探し方

本書では次の方法で、使いたい機能の説明を探すことができます。

- 目次から探す
  本書の内容を記載順に示します。
- レジスタから探す
  本デバイスのレジスタが一覧で記載されています。調べたいレジスタの名称から、配置アドレスやレジスタの説明頁を探すことができます。
  本文中では各レジスタの配置アドレスを記載しておりません。各レジスタのアドレスを確認するときは『付録』の『I/O マップ』を参照してください。
- 索引から探す
  周辺機能の名称などのキーワードから機能の説明を探すことができます。

章について

本書では、基本的に1つの周辺機能を1つの章で説明しています。

用語について

本書で使用している用語について示します。

用語	説明
ワード	32 ビット単位でのアクセスを指します。
ハーフワード	16 ビット単位でのアクセスを指します。
バイト	8 ビット単位でのアクセスを指します。

本書の読み方

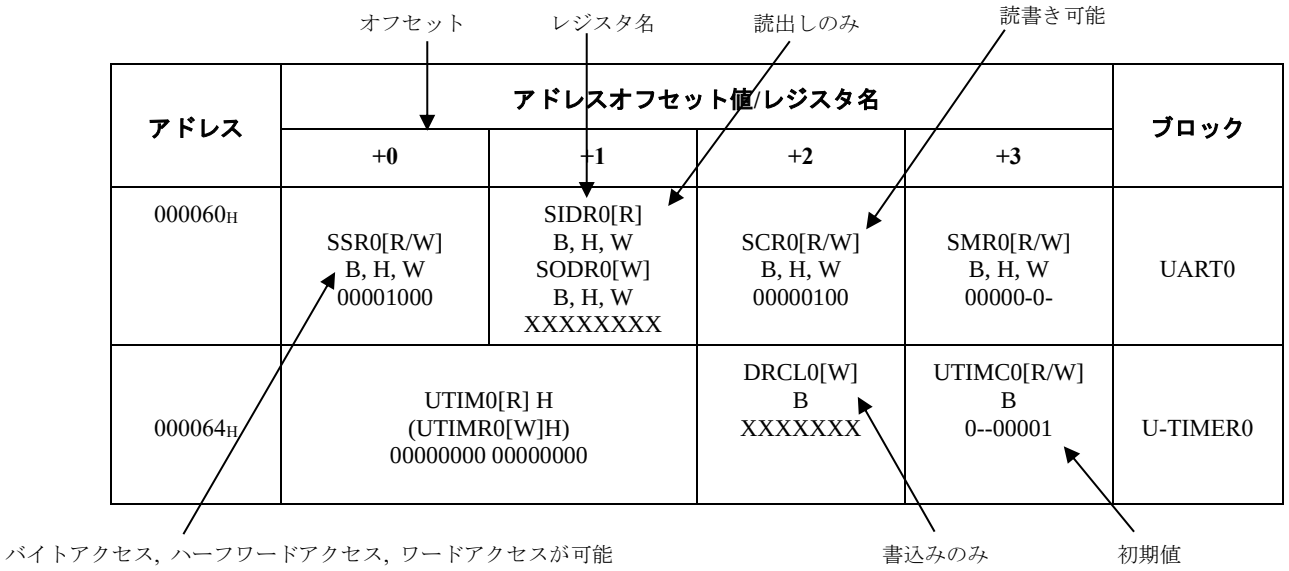
■ 主要な用語

本シリーズで使用する主要な用語に関して説明します。

用語	説明
XBS	32 ビット幅の高速内部バスです。 バスマスタには、CPU (インストラクションフェッチ) , CPU (データ読み・書き) , オンチップバスからのアクセスがあります。 バススレーブにはオンチップバスへのアクセス, RAM, (XBS 内蔵のワイルドレジスタを経由して) フラッシュメモリがあります。 バスはクロスバースイッチ構成となっており、各バスマスタから各バススレーブへ同時動作が可能です。
オンチップバス	32 ビット幅の高速内部バスです。XBS 用と DMA 用の 2 レイヤー構成になっており、それぞれが同時動作できます。 XBS 用レイヤーのバスマスタには XBS からのアクセスがあります。 DMA 用レイヤーのバスマスタには DMA からのアクセスがあります。 双方のレイヤーのバススレーブには外部バスインタフェース, CAN, 16/32 周辺バスブリッジなどがあります。 DMA 用レイヤーのみのバススレーブに XBS へのアクセスがあります。
32 ビット周辺バス	32 ビット幅の低速内部バスです。 各種ペリフェラルを接続しています。
16 ビット周辺バス (R-Bus)	16 ビット幅の低速内部バスです。 各種ペリフェラルを接続しています。本バスへの 32 ビット幅のアクセスは 16 ビット × 2 に分割されます。
外部バス (外バス)	8/16 ビット幅の低速外部バスです。メモリデバイスや ASIC などを接続します。本シリーズがバスマスタになり、外バスに接続されるデバイスがバススレーブになります。
メインクロック (MCLK)	高速側発振を起源とした LSI 動作の基準となるクロックです。 メイン発振安定待ち用タイマ, クロック発生部 (PLL) などに接続しています。
サブクロック (SBCLK)	低速側発振を起源とした LSI 動作の基準となるクロックです。 サブ発振安定待ち用タイマなどに接続しています。 クロック 2 系統品種のみ使用できます。
CR 発振	ウォッチドッグタイマ 1 (ハードウェアウォッチドッグ) 用のクロックです。
PLL クロック(PLLCLK)	メインクロックを PLL で逓倍したものです。
CPU クロック (CCLK)	XBS 配下のペリフェラル用のクロックです。
オンチップバスクロック (HCLK)	オンチップバス配下のペリフェラル用のクロックです。
周辺クロック (PCLK)	32 ビット周辺バスおよび 16 ビット周辺バス配下のペリフェラル用のクロックです。
外部バスクロック (TCLK)	X-BUS につながる外部バスインタフェースと外部クロック出力動作の基準となるクロックです。クロック発生部にてベースクロックから生成します。
メインクロックモード	メインクロックを基準に動作しているモードです。このメインクロックモードの中にはメイン RUN, メインスリープ, メインストップ, 発振安定待ち RUN, 発振安定待ちリセット, プログラムリセットの状態があります。
メイン RUN	メインクロックモードであり、かつすべての回路が動作可能な状態となります。
発振安定待ち時間	クロックを停止状態から発振状態に設定すると発振安定時間をとります。発振安定待ちの間は, そのクロックは供給されません。
OCD	本シリーズ用のオンチップデバッガです。
OCDU	本製品に内蔵されている、OCD のインタフェースです。
OCD ツール	OCD ツールを、本品種の DEBUG I/F 端子に接続します。
チップリセットシーケンス	チップリセットシーケンス中に OCD ツールの接続確認を行います。(1026+3) PCLK サイクルかかります。
電源遮断	対象となる回路への電源供給を止めて、消費電力を低減します。
常時電源 ON ブロック	電源遮断の対象とならない区画です。
PMU パワーマネジメントユニット	電源遮断の制御を行います。PMU は常時 ON ブロックにあります。

用語	説明
SSCG	「スペクトラム拡散クロックジェネレータ」 ＝Spread Spectrum Clock Generator 電子機器内のクロックが単一の周波数を発生すると、その周波数およびその高調波での輻射が大きくなります。 SSCG は、クロック周波数をわずかに変動させて発振させる(＝周波数変調)ことによって、EMI のピークを低く抑える働きをする技術です。

アクセス単位とアドレス位置



アクセスには、バイト, ハーフワード, ワードの3 種類がありますが、レジスタによってはアクセス制限がありますので注意ください。詳細につきましては、『I/O マップ』または、各章の『レジスタ詳細説明』を参照してください。

B, H, W ：バイトアクセス, ハーフワードアクセス, ワードアクセスが可能

B ：バイトアクセス (必ずバイトでアクセスしてください。)

H ：ハーフワードアクセス (必ずハーフワードでアクセスしてください。)

W ：ワードアクセス (必ずワードでアクセスしてください。)

B, H ：バイトアクセス, ハーフワードアクセスのみ (ワードではアクセスできません。)

H, W ：ハーフワードアクセス, ワードアクセスのみ (バイトではアクセスできません。)

(参考)

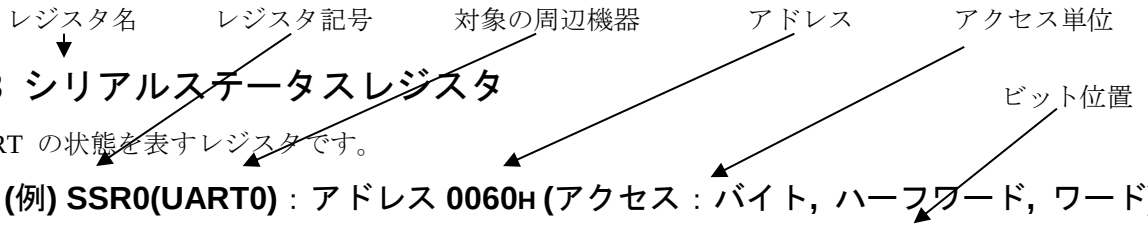
アクセス時のアドレス位置を説明します。

■ ワードアクセスの場合、アドレスは 4 の倍数(最下位 2 ビットは強制的に"00"です)
■ ハーフワードアクセスの場合、アドレスは 2 の倍数(最下位 1 ビットは強制的に"0"です)
■ バイトアクセスの場合、アドレスはそのままです。

このため、例えば、SSR0レジスタをハーフワードアクセスするには、アドレス060_H番地に対して、SSR0+SIDR0 (SODR0)レジスタにアクセスするかたちになります。

(アドレスオフセットが+1 と+2 (例: SIDR0+SCR0)でのハーフワードアクセスはできません。)

アクセス単位とビット位置について



4.3 シリアルステータスレジスタ

UART の状態を表すレジスタです。

(例) SSR0(UART0)：アドレス 0060H (アクセス：バイト，ハーフワード，ワード)

bit	7	6	5	4	3	2	1	0
	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE
初期値	0	0	0	0	1	0	0	0
属性	R/W	R/WX	R/WX	R/WX	R/WX	R/W	R/W	R/W

アクセス単位がかわるとビット位置が変わります。

アドレスオフセット値が+0の場合：(例 SSR0レジスタ)

アクセスサイズ	アドレス	ビット位置							
ワード	060H+0H	7	6	5	4	3	2	1	0
ハーフワード	060H+0H	15	14	13	12	11	10	9	8
バイト	060H+0H	31	30	29	28	27	26	25	24
ビット名		PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE

アドレスオフセット値が+1の場合: (例 SIDR0レジスタ)

アクセスサイズ	アドレス	ビット位置							
ワード	060H+1H	7	6	5	4	3	2	1	0
ハーフワード	060H+0H	7	6	5	4	3	2	1	0
バイト	060H+0H	23	22	21	20	19	18	17	16
ビット名		D7	D6	D5	D4	D3	D2	D1	D0

アドレスオフセット値が+2の場合 ：(例 SCR0レジスタ)

アクセスサイズ	アドレス	ビット位置							
ワード	060H+2H	7	6	5	4	3	2	1	0
ハーフワード	060H+2H	15	14	13	12	11	10	9	8
バイト	060H+0H	15	14	13	12	11	10	9	8
ビット名		PEN	P	SBL	CL	A/D	REC	RXE	TXE

アドレスオフセット値が+3の場合: (例 SMR0レジスタ)

アクセスサイズ	アドレス	ビット位置							
ワード	060H+3H	7	6	5	4	3	2	1	0
ハーフワード	060H+2H	7	6	5	4	3	2	1	0
バイト	060H+0H	7	6	5	4	3	2	1	0
ビット名		MD1	MD0	CS2	CS1	CS0	-	SCKE	-

ビット属性機能の意味について

R　　：読出し可能です。

W　　：書込み可能です。

RM　　：リードモディファイライト(RMW)動作時の読出し動作の意味です。

"/"(スラッシュ)　R/W　　：　読出しと書込みが可能です。(読出し値は，書き込んだ値です。)

","(カンマ)　　R, W　　：　読出しと書込みで意味が違います。(読出し値は，書き込んだ値と異なります。)

R0　　：読出し値は"0" です。

R1　　：読出し値は"1" です。

W0　　：必ず"0" を書き込んでください。

W1　　：必ず"1" を書き込んでください。

(RM0)　：リードモディファイライト(RMW)動作では"0" が読めます。

(RM1)　：リードモディファイライト(RMW)動作では"1" が読めます。

RX　　：読出し値は不定です。 (予約ビットまたは，未定義ビット)

WX　　：書込みを行っても動作に影響を与えません。 (未定義ビット)

■ R/W記載例

- R/W　　：リード/ライト可能 (読出し値は書込み値と同じです。)
- R,W　　：リード/ライト可能 (読出し値と書込み値が異なります。)
- R,RM/W　　：リード/ライト可能 (読出し値と書込み値が異なります。 リードモディファイライト(RMW)系命令時は書込み値が読み出されます。) 例: ポートデータレジスタ
- R(RM1), W　　：リード/ライト可能 (読出し値と書込み値が異なります。リードモディファイライト(RMW)系命令時は"1" を読み出します。) 例: 割込み要求フラグ
- R,WX　　：リードオンリ (読出しは可能です。書込みは動作に影響ありません。)
- R1,W　　：ライトオンリ (書込みは可能です。読出し値は"1" です。)
- R0,W　　：ライトオンリ (書込みは可能です。読出し値は"0" です。)
- RX,W　　：ライトオンリ (書込みは可能です。読出し値は不定です。)
- R/W0　　：予約ビット (書込み値は"0" です。 読出し値は書込み値です。)
- R0,W0　　：予約ビット (書込み値は"0" です。読出し値は"0" です。)
- R1,W0　　：予約ビット (書込み値は"0" です。読出し値は"1" です。)
- RX,W0　　：予約ビット (書込み値は"0" です。読出し値は不定です。)
- R/W1　　：予約ビット (書込み値は"1" です。読出し値は書込み値です。)
- R1,W1　　：予約ビット (書込み値は"1" です。読出し値は"1" です。)
- R0,W1　　：予約ビット (書込み値は"1" です。読出し値は"0" です。)
- RX,W1　　：予約ビット (書込み値は"1" です。読出し値は不定です。)

- RX,WX　　：未定義ビット (読出し値は不定です。書込みは動作に影響ありません。)
- R0,WX　　：未定義ビット (読出し値は"0" です。書込みは動作に影響ありません。)

■ 注釈記号

本ドキュメントでは、番号付き注釈記号を語句の先頭に表記する場合があります (例: *1)レジスタ)。これらの注釈に対する説明は、そのページの最下部の区切り線下に記載します。

Contents

はじめに **3**
本書の使い方 **4**
Contents **11**
1. 概要 **33**
1.1 概要 33
1.2 特長 33
1.2.1 FR81S CPU コア 34
1.2.2 周辺機能 35
1.3 品種構成 39
1.4 機能概要 43
1.5 ブロックダイヤグラム 48
1.6 CPU 49
1.6.1 汎用レジスタ 49
1.6.2 専用レジスタ 50
1.7 端子配列図 51
1.8 パッケージ外形図 53
1.9 端子機能説明 56
1.10 各機能の端子一覧 71
1.10.1 A/D コンバータの端子 72
1.10.2 ベースタイマの端子 (ch.0, ch.1) 73
1.10.3 CAN の端子 (ch.0～ch.2) 73
1.10.4 DAC の端子 (ch.0, ch.1) 73
1.10.5 外部バスインタフェースの端子 73
1.10.6 外部割込み要求入力の端子 (ch.0～ch.15) 75
1.10.7 フリーランタイマの端子 (ch.0～ch.5) 76
1.10.8 汎用入出力ポートの端子 76
1.10.9 HS_SPI の端子 79
1.10.10 インプットキャプチャの端子 (ch.0～ch.11) 79
1.10.11 LCDC の端子 80
1.10.12 LIN-UART の端子 (ch.2～ch.7) 81
1.10.13 マルチファンクションシリアルインタフェースの端子 (ch.0, ch.1, ch.8, ch.9) 82
1.10.14 アウトプットコンペアの端子 (ch.0～ch.11) 83
1.10.15 PPG の端子 (ch.0～ch.23) 84
1.10.16 リロードタイマの端子(ch.0～ch.3) 85
1.10.17 リアルタイムクロックの端子 86
1.10.18 ステッピングモータ制御の端子 (ch.0～ch.5) 86
1.10.19 サウンドジェネレータの端子 (ch.0～ch.4) 87
1.10.20 アップダウンカウンタの端子 (ch.0, ch.1) 87
1.10.21 クロック端子 87
1.10.22 モード端子 88
1.10.23 リセット端子 88
1.10.24 電源端子 88

2. デバイスの取扱い 95
2.1 取扱上のご注意 95
2.2 デバイス取り扱いについて 99
2.3 使用上の注意事項 102
2.3.1 兼用ポートの機能切り替え 102
2.3.2 低消費電力モード 102
2.3.3 ステータスフラグを含むレジスタに書込みを行う場合の注意 102
3. CPU 103
3.1 概要 103
3.2 特長 103
3.3 CPU 動作記述 105
3.4 パイプライン動作 106
3.5 浮動小数点演算処理 106
3.6 データ構造 106
3.7 アドレッシング 107
3.8 プログラミングモデル 108
3.8.1 汎用レジスタ，専用レジスタ，浮動小数点レジスタ 108
3.8.2 システムレジスタ 109
3.9 リセット・EIT 処理 110
3.9.1 リセット 110
3.9.2 EIT 処理 110
3.9.3 ベクタテーブル 111
3.10 メモリ保護機能（MPU） 113
3.10.1 概要 113
3.10.2 レジスター覧 114
3.10.3 レジスタ説明 115
3.10.4 メモリ保護機能（MPU）の動作説明 130
4. 動作モード 133
4.1 概要 133
4.2 特長 133
4.3 構成 134
4.4 レジスタ 135
4.5 動作説明 136
4.5.1 MD0, MD1, MD2, P127 端子の設定 136
4.5.2 動作モードの取込み 137
4.5.3 各モードの説明 138
5. クロック 139
5.1 概要 139
5.2 特長 141
5.3 構成 141
5.4 レジスタ 145
5.4.1 分周設定レジスタ 0 : DIVR0 (DIVide clock configuration Register 0) 146
5.4.2 分周設定レジスタ 1 : DIVR1 (DIVide clock configuration Register 1) 147
5.4.3 分周設定レジスタ 2 : DIVR2 (DIVide clock configuration Register 2) 148
5.4.4 クロックソース設定レジスタ: CSELR (Clock source SELect Register) 149
5.4.5 クロックソース監視レジスタ: CMONR (Clock source MONitor Register) 152
5.4.6 メインタイマ制御レジスタ: MTMCR (Main clock TiMer Control Register) 154
5.4.7 サブタイマ制御レジスタ: STMCR (Sub clock TiMer Control Register) 157
5.4.8 PLL 設定レジスタ : PLLCR (PLL Configuration Register) 159
5.4.9 発振安定待ち設定レジスタ : CSTBR (Clock STaBilization select Register) 161
5.4.10 PLL クロック発振安定待ちタイマ設定レジスタ : PTMCR: (PLL clock osc TiMer Control Register) 163
5.4.11 PLL/SSCG クロック選択レジスタ : CCPSSELR (CCtl Pll/Sscg clock SELect Register) 164

5.4.12 PLL/SSCG 出力クロック分周設定レジスタ : CCPSDIVR (CCtl Pll/Sscg clock DIVide Register) 165
5.4.13 PLL フィードバック分周設定レジスタ : CCPLLFBR (CCtl PLL FB clock divide Register)...167
5.4.14 SSCG フィードバック分周設定レジスタ 0 : CCSSFBR0 (CCtl SScg FB clock divide Register 0) 168
5.4.15 SSCG フィードバック分周設定レジスタ 1 : CCSSFBR1 (CCtl SScg FB clock divide Register 1) 169
5.4.16 SSCG コンフィグ設定レジスタ 0 : CCSSCCR0 (CCtl SSCg Config. Register 0)....170
5.4.17 SSCG コンフィグ設定レジスタ 1 : CCSSCCR1 (CCtl SSCg Config. Register 1)....172
5.4.18 クロックギアコンフィグ設定レジスタ 0 : CCCGRCR0 (CCtl Clock Gear Config. Register 0) 173
5.4.19 クロックギアコンフィグ設定レジスタ 1 : CCCGRCR1 (CCtl Clock Gear Config. Register 1) 175
5.4.20 クロックギアコンフィグ設定レジスタ 2 : CCCGRCR2 (CCtl Clock Gear Config. Register 2) 176
5.4.21 RTC/PMU クロック選択レジスタ : CCRTSELR (CCtl RTc pmu clock SELect Register) 177
5.4.22 PMU クロック分周設定レジスタ 0 : CCPMUCR0 (CCtl PMU Clock divide Register 0) 179
5.4.23 PMU クロック分周設定レジスタ 1 : CCPMUCR1 (CCtl PMU Clock divide Register 1) 180
5.4.24 同期/非同期コントロールレジスタ : SACR (Sync/Async Control Register)....181
5.4.25 周辺インタフェースクロック分周 : PICD (Peripheral Interface Clock Divider) 182
5.5 動作説明....183
5.5.1 発振制御 183
5.5.2 発振安定待ち 188
5.5.3 ソースクロック (SRCCLK) 選択 190
5.5.4 タイマ 195
5.5.5 クロック競合時の注意 199
5.5.6 クロックギア回路 199
5.5.7 MDI 通信中の動作....200
5.5.8 PMU クロック(PMUCLK) について 201
6. クロック・リセット状態遷移....203
6.1 概要....203
6.2 デバイス状態と各遷移 203
6.2.1 状態遷移図....204
6.2.2 各状態の説明 206
6.2.3 各状態遷移要求の優先順位....207
6.3 デバイス状態と対応するレギュレータモード....208
7. リセット....209
7.1 概要....209
7.2 特長....209
7.3 構成....210
7.4 レジスタ....212
7.4.1 リセット要因レジスタ : RSTRR (ReSeT Result Register) 212
7.4.2 リセット制御レジスタ : RSTCR (ReSeT Control Register)....214
7.4.3 CPU 異常動作レジスタ : CPUAR (CPU Abnormal operation Register) 215
7.4.4 PMU ステータスレジスタ : PMUSTR (Power Management Unit STatus register) 217
7.5 動作説明....218
7.5.1 リセットレベル....218
7.5.2 リセット要因 219
7.5.3 リセット受付 222
7.5.4 リセット発行 224
7.5.5 リセットシーケンス....227
7.5.6 注意事項 230
8. DMA コントローラ (DMAC)....231
8.1 概要....231

8.2 特長 231
8.3 構成 232
8.4 レジスタ 233
8.4.1 DMA コントロールレジスタ : DMACR (DMA Control Register) 235
8.4.2 DMA チャネルコントロールレジスタ 0-15 : DCCR0-15 (DMA Channel Control Register 0-15) 237
8.4.3 DMA チャネルステータスレジスタ 0-15 : DCSR0-15 (DMA Channel Status Register 0-15) 242
8.4.4 DMA 転送回数レジスタ 0-15 : DTCR0-15 : (DMA Transfer Count Register 0-15) 244
8.4.5 DMA 転送元レジスタ 0-15 : DSAR0-15 : (DMA Source Address Register 0-15) 245
8.4.6 DMA 転送先レジスタ 0-15 : DDAR0-15 (DMA Destination Address Register 0-15) 247
8.4.7 DMA 転送抑止 NMI フラグレジスタ : DNMIR (DMA-halt by NMI Register) 249
8.4.8 DMA 転送抑止レベルレジスタ : DILVR (DMA-halt by Interrupt Level Register) 250
8.5 動作説明 251
8.5.1 DMA 動作許可 251
8.5.2 各チャネルにおいて独立に設定する項目 252
8.5.3 動作 256
8.6 DMA 使用例 268
9. DMA 転送要求の発生・クリア 271
9.1 概要 271
9.2 特長 271
9.3 構成 272
9.4 レジスタ 273
9.4.1 DMA 要求クリアレジスタ 0 : ICSEL0 (Interrupt Clear SELect register 0) 274
9.4.2 DMA 要求クリアレジスタ 1 : ICSEL1 (Interrupt Clear SELect register 1) 275
9.4.3 DMA 要求クリアレジスタ 2 : ICSEL2 (Interrupt Clear SELect register 2) 276
9.4.4 DMA 要求クリアレジスタ 3 : ICSEL3 (Interrupt Clear SELect register 3) 277
9.4.5 DMA 要求クリアレジスタ 4 : ICSEL4 (Interrupt Clear SELect register 4) 277
9.4.6 DMA 要求クリアレジスタ 5 : ICSEL5 (Interrupt Clear SELect register 5) 278
9.4.7 DMA 要求クリアレジスタ 6 : ICSEL6 (Interrupt Clear SELect register 6) 278
9.4.8 DMA 要求クリアレジスタ 7 : ICSEL7 (Interrupt Clear SELect register 7) 279
9.4.9 DMA 要求クリアレジスタ 8 : ICSEL8 (Interrupt Clear SELect register 8) 279
9.4.10 DMA 要求クリアレジスタ 9 : ICSEL9 (Interrupt Clear SELect register 9) 280
9.4.11 DMA 要求クリアレジスタ 10 : ICSEL10 (Interrupt Clear SELect register 10) 280
9.4.12 DMA 要求クリアレジスタ 11 : ICSEL11 (Interrupt Clear SELect register 11) 281
9.4.13 DMA 要求クリアレジスタ 12 : ICSEL12 (Interrupt Clear SELect register 12) 281
9.4.14 DMA 要求クリアレジスタ 13 : ICSEL13 (Interrupt Clear SELect register 13) 282
9.4.15 DMA 要求クリアレジスタ 14 : ICSEL14 (Interrupt Clear SELect register 14) 282
9.4.16 DMA 要求クリアレジスタ 15 : ICSEL15 (Interrupt Clear SELect register 15) 283
9.4.17 DMA 要求クリアレジスタ 16 : ICSEL16 (Interrupt Clear SELect register 16) 283
9.4.18 DMA 要求クリアレジスタ 17 : ICSEL17 (Interrupt Clear SELect register 17) 284
9.4.19 DMA 要求クリアレジスタ 18 : ICSEL18 (Interrupt Clear SELect register 18) 284
9.4.20 DMA 要求クリアレジスタ 19 : ICSEL19 (Interrupt Clear SELect register 19) 285
9.4.21 DMA 要求クリアレジスタ 20 : ICSEL20 (Interrupt Clear SELect register 20) 286
9.4.22 DMA 要求クリアレジスタ 21 : ICSEL21 (Interrupt Clear SELect register 21) 287
9.4.23 DMA 要求クリアレジスタ 22 : ICSEL22 (Interrupt Clear SELect register 22) 287
9.4.24 IO 転送要求設定レジスタ 0-15 : IORR0-15 (IO triggered DMA Request Register for ch.0-15) 288
9.5 動作説明 289
9.5.1 設定 289
9.5.2 注意事項 289
10. FixedVector 機能 291
10.1 概要 291
10.2 動作説明 292
11. I/O ポート 293

11.1 概要 293
11.2 特長 293
11.3 構成 294
11.4 レジスタ 294
11.4.1 ポートデータレジスタ 00～19 : PDR00-19 (Port Data Register 00-19) 296
11.4.2 データ方向レジスタ 00～19 : DDR00-19 (Data Direction Register 00-19) 297
11.4.3 ポート機能レジスタ 00～19 : PFR00-19 (Port Function Register 00-19) 298
11.4.4 入力データダイレクトレジスタ 00～19 : PDDR00-19 (Port Data Direct Register 00-19) 299
11.4.5 プルアップダウン制御レジスタ 00～19 : PPCR00-19 (Port Pull-up/down Control Register 00-19) 300
11.4.6 プルアップダウン許可レジスタ 00～19 : PPER00-19 (Port Pull-up/down Enable Register 00-19) 301
11.4.7 ポート入力レベル選択レジスタ 00～19 : PILR00-19 (Port Input Level Register 00-19) 302
11.4.8 拡張ポート入力レベル選択レジスタ 00～19 : EPILR00-19 (Extended Port Input Level Register 00-19) 303
11.4.9 ポート出力駆動レジスタ 00～19 : PODR00-19 (Port Output Drive Register 00-19) 304
11.4.10 拡張ポート出力駆動レジスタ 01～03,06～08 : EPODR01-03,06-08 (Extended Port Output Drive Register 01-03,06-08) 306
11.4.11 拡張ポート機能レジスタ 00～54 : EPFR00-54 (Extended Port Function Register 00-54) 306
11.4.12 ポート入力許可レジスタ: PORTEN (PORT ENable register) 332
11.5 動作説明 333
11.5.1 端子の入出力の割り当て 334
11.5.2 EPFR 設定の優先度について 342
11.5.3 入力 I/O リロケーション設定時の注意 343
11.5.4 GPORTEN による入力遮断 343
11.5.5 A/D コンバータ機能のある端子の注意事項 343
11.5.6 ベースタイマ TIOA1 端子使用時の設定 343
11.5.7 電源遮断からのウェイクアップ時の動作 344
11.5.8 I/O ポート機能切り替え時の注意事項 344
11.5.9 特定リソース使用時の入力遮断 344
12. 割込み制御 (割込みコントローラ) 345
12.1 概要 345
12.2 特長 345
12.3 構成 346
12.4 レジスタ 347
12.4.1 割込みコントロールレジスタ 00～47 : ICR00-ICR47 (Interrupt Control Register 00-47) 348
12.5 動作説明 349
13. 外部割込み入力 351
13.1 概要 351
13.2 特長 351
13.3 構成 351
13.4 レジスタ 352
13.4.1 外部割込み要因レジスタ 0/1 : EIRR0/1: (External Interrupt Request Register 0/1) 353
13.4.2 外部割込み許可レジスタ 0/1 : ENIR0/1: (ENable Interrupt request Register 0/1) 354
13.4.3 外部割込み要求レベルレジスタ 0/1 : ELVR0/1: (External interrupt LeVel Register 0/1) 355
13.5 動作説明 356
13.6 設定 358
13.7 Q&A 359
13.8 注意事項 360
14. NMI 入力 361
14.1 概要 361
14.2 特長 361
14.3 構成 362
14.4 レジスタ 362

14.5 動作説明 363
14.6 使用例 364
15. 遅延割込み 365
15.1 概要 365
15.2 特長 365
15.3 構成 365
15.4 レジスタ 365
15.5 動作説明 366
15.6 制限事項 366
16. 割込み要求一括読出し 367
16.1 概要 367
16.2 特長 367
16.3 構成 367
16.4 レジスタ 368
16.4.1 割込み要求一括読出しレジスタ 0 上位 : IRPR0H (Interrupt Request Peripheral Read register 0H) 369
16.4.2 割込み要求一括読出しレジスタ 0 下位 : IRPR0L (Interrupt Request Peripheral Read register 0L) 370
16.4.3 割込み要求一括読出しレジスタ 1 上位 : IRPR1H (Interrupt Request Peripheral Read register 1H) 371
16.4.4 割込み要求一括読出しレジスタ 1 下位 : IRPR1L (Interrupt Request Peripheral Read register 1L) 372
16.4.5 割込み要求一括読出しレジスタ 2 上位 : IRPR2H (Interrupt Request Peripheral Read register 2H) 373
16.4.6 割込み要求一括読出しレジスタ 2 下位 : IRPR2L (Interrupt Request Peripheral Read register 2L) 374
16.4.7 割込み要求一括読出しレジスタ 3 上位 : IRPR3H (Interrupt Request Peripheral Read register 3H) 375
16.4.8 割込み要求一括読出しレジスタ 3 下位 : IRPR3L (Interrupt Request Peripheral Read register 3L) 376
16.4.9 割込み要求一括読出しレジスタ 4 上位 : IRPR4H (Interrupt Request Peripheral Read register 4H) 377
16.4.10 割込み要求一括読出しレジスタ 4 下位 : IRPR4L (Interrupt Request Peripheral Read register 4L) 378
16.4.11 割込み要求一括読出しレジスタ 5 上位 : IRPR5H (Interrupt Request Peripheral Read register 5H) 379
16.4.12 割込み要求一括読出しレジスタ 5 下位 : IRPR5L (Interrupt Request Peripheral Read register 5L) 380
16.4.13 割込み要求一括読出しレジスタ 6 上位 : IRPR6H (Interrupt Request Peripheral Read register 6H) 381
16.4.14 割込み要求一括読出しレジスタ 6 下位 : IRPR6L (Interrupt Request Peripheral Read register 6L) 382
16.4.15 割込み要求一括読出しレジスタ 7 上位 : IRPR7H (Interrupt Request Peripheral Read register 7H) 383
16.4.16 割込み要求一括読出しレジスタ 7 下位 : IRPR7L (Interrupt Request Peripheral Read register 7L) 384
16.4.17 割込み要求一括読出しレジスタ 8 上位 : IRPR8H (Interrupt Request Peripheral Read register 8H) 385
16.4.18 割込み要求一括読出しレジスタ 8 下位 : IRPR8L (Interrupt Request Peripheral Read register 8L) 386
16.4.19 割込み要求一括読出しレジスタ 9 上位 : IRPR9H (Interrupt Request Peripheral Read register 9H) 387
16.4.20 割込み要求一括読出しレジスタ 9 下位 : IRPR9L (Interrupt Request Peripheral Read register 9L) 388

16.4.21 割込み要求一括読出しレジスタ 10 上位 : IRPR10H (Interrupt Request Peripheral Read register 10H) 389
16.4.22 割込み要求一括読出しレジスタ 10 下位 : IRPR10L (Interrupt Request Peripheral Read register 10L) 390
16.4.23 割込み要求一括読出しレジスタ 11 上位 : IRPR11H (Interrupt Request Peripheral Read register 11H) 391
16.4.24 割込み要求一括読出しレジスタ 11 下位 : IRPR11L (Interrupt Request Peripheral Read register 11L) 392
16.4.25 割込み要求一括読出しレジスタ 12 上位 : IRPR12H (Interrupt Request Peripheral Read register 12H) 393
16.4.26 割込み要求一括読出しレジスタ 12 下位 : IRPR12L (Interrupt Request Peripheral Read register 12L) 394
16.4.27 割込み要求一括読出しレジスタ 13 上位 : IRPR13H (Interrupt Request Peripheral Read register 13H) 395
16.4.28 割込み要求一括読出しレジスタ 13 下位 : IRPR13L (Interrupt Request Peripheral Read register 13L) 396
16.4.29 割込み要求一括読出しレジスタ 14 上位 : IRPR14H (Interrupt Request Peripheral Read register 14H) 397
16.4.30 割込み要求一括読出しレジスタ 14 下位 : IRPR14L (Interrupt Request Peripheral Read register 14L) 398
16.4.31 割込み要求一括読出しレジスタ 15 上位 : IRPR15H (Interrupt Request Peripheral Read register 15H) 399
16.5 動作説明 400
17. PPG 401
17.1 概要 401
17.2 特長 402
17.3 構成 403
17.4 レジスタ 404
17.4.1 PPG 周期設定レジスタ : PCSR 408
17.4.2 PPG デューティ設定レジスタ : PDUT 409
17.4.3 PPG 制御ステータスレジスタ : PCN 410
17.4.4 ジェネラル制御レジスタ 10～13 : GCN10～GCN13 413
17.4.5 ジェネラル制御レジスタ 14, 15 : GCN14, GCN15 415
17.4.6 ジェネラル制御レジスタ 20～25 : GCN20～GCN25 416
17.4.7 PPG タイマレジスタ : PTMR 417
17.4.8 PPG0 出力分周設定レジスタ : PPGDIV 418
17.5 動作説明 419
17.5.1 PWM 動作 420
17.5.2 ワンショット動作 422
17.5.3 再起動動作 424
17.6 設定 425
17.7 Q&A 426
17.7.1 周期とデューティの設定 (書換え) 方法は? 427
17.7.2 PPG 動作を許可/停止するには? 427
17.7.3 PPG の動作モード (PWM 動作/ワンショット動作) を設定するには? 427
17.7.4 再起動させるには? 428
17.7.5 カウントクロックの種類と選択方法は? 428
17.7.6 PPG 端子出力のレベルを固定させるには? 429
17.7.7 起動トリガの種類と選択方法は? 430
17.7.8 出力極性を反転させるには? 431
17.7.9 端子を PPG 出力端子にするには? 432
17.7.10 起動トリガの発生方法は? 432
17.7.11 PPG 動作を停止するには? 432
17.7.12 割込み関連レジスタは? 433

17.7.13 割込みの種類と選択方法は?....434
17.7.14 割込みを許可/禁止/クリアするには?....434
17.8 サンプルプログラム....435
17.9 注意事項....439
18. ウォッチドッグタイマ....441
18.1 概要....441
18.2 特長....442
18.3 構成....443
18.4 レジスタ....444
18.4.1 ウォッチドッグタイマ 0 制御レジスタ : WDTCR0 (WatchDog Timer Control Register 0)....444
18.4.2 ウォッチドッグタイマ 0 クリアレジスタ : WDTCPR0 (WatchDog Timer Clear Pattern Register 0)....446
18.4.3 ウォッチドッグタイマ 1 周期インフォメーションレジスタ : WDTCR1 (WatchDog Timer Cycle information Register 1)....446
18.4.4 ウォッチドッグタイマ 1 クリアレジスタ : WDTCPR1 (WatchDog Timer Clear Pattern Register 1)....447
18.5 動作説明....448
18.6 使用例....450
19. ベースタイマ....451
19.1 概要....451
19.2 特長....451
19.2.1 16/32 ビットリロードタイマ....452
19.2.2 16 ビット PWM タイマ....453
19.2.3 16/32 ビット PWC タイマ....454
19.2.4 16 ビット PPG タイマ....455
19.3 構成....456
19.4 レジスタ....457
19.4.1 共通レジスタ....459
19.4.2 16/32 ビットリロードタイマ時のレジスタ....467
19.4.3 16 ビット PWM タイマ時のレジスタ....470
19.4.4 16 ビット PPG タイマ時のレジスタ....475
19.4.5 16/32 ビット PWC タイマ時のレジスタ....479
19.5 動作説明....483
19.5.1 タイマ機能の選択....483
19.5.2 入出力割当て....484
19.5.3 32 ビットモード動作....487
19.5.4 16/32 ビットリロードタイマの動作....488
19.5.5 16 ビット PWM タイマの動作....500
19.5.6 16 ビット PPG タイマの動作....510
19.5.7 16/32 ビット PWC タイマの動作....521
20. リロードタイマ....535
20.1 概要....535
20.2 特長....536
20.3 構成....537
20.4 レジスタ....538
20.4.1 コントロールステータスレジスタ : TMCSR (TiMer Control and Status Register)....539
20.4.2 16 ビットタイマレジスタ : TMR (16bit TiMer Register)....542
20.4.3 16 ビットタイマリロードレジスタ A, 16 ビットタイマリロードレジスタ B : TMRLRA, TMRLRB (16bit TiMer ReLoad Register A/B)....543
20.5 動作説明....545
20.5.1 設定....545
20.5.2 動作手順....549
20.5.3 各カウンタ動作の説明....554
20.5.4 カスケード入力....572

20.5.5 同時動作の優先順位 572
20.6 アプリケーションノート 573
20.6.1 シングルワンショットタイマ 575
20.6.2 リロードタイマ 577
20.6.3 PPG 579
20.6.4 PWM 582
20.6.5 PWC 585
21. フリーランタイマ 587
21.1 概要 587
21.2 特長 588
21.3 構成 589
21.4 レジスタ 590
21.4.1 タイマ制御レジスタ (上位ビット) : TCCSH 591
21.4.2 タイマ制御レジスタ (下位ビット) : TCCSL 592
21.4.3 コンペアクリアレジスタ : CPCLR 594
21.4.4 タイマデータレジスタ : TCDT 595
21.5 動作説明 596
21.5.1 フリーランタイマのカウント動作 597
21.5.2 カウントアップ 599
21.5.3 タイマクリア 600
21.5.4 フリーランタイマの各種クリア動作 601
21.5.5 タイマ割込み 602
21.6 設定 603
21.7 Q&A 604
21.7.1 内部クロック分周の選択方法は? 604
21.7.2 外部クロックの選択方法は? 605
21.7.3 フリーランタイマのカウント動作を許可/禁止するには? 605
21.7.4 フリーランタイマをクリアするには? 605
21.7.5 割込み関連レジスタは? 606
21.7.6 コンペアクリア割込みを許可するには? 606
21.7.7 フリーランタイマ動作を停止するには? 606
21.8 サンプルプログラム 607
21.9 注意事項 608
22. アウトプットコンペア 609
22.1 概要 609
22.2 特長 610
22.3 構成図 612
22.4 レジスタ 612
22.4.1 フリーランタイマ選択レジスタ: OCFS 614
22.4.2 アウトプット制御レジスタ (上位ビット) : OCSH 615
22.4.3 アウトプット制御レジスタ (下位ビット) : OCSL 617
22.4.4 コンペアレジスタ : OCCP 619
22.5 動作説明 620
22.5.1 アウトプットコンペア出力 (独立反転) CMOD = "0" 621
22.5.2 アウトプットコンペア出力 (連携反転) CMOD="1" 622
22.5.3 アウトプットコンペアの動作タイミング 623
22.6 設定 625
22.7 Q&A 626
22.7.1 コンペア値を設定するには? 626
22.7.2 コンペアモードを設定するには? (OCU1 での例) 626
22.7.3 コンペア動作を許可/禁止するには? (OCU0, OCU1 での例) 627
22.7.4 コンペア端子出力の初期レベルを設定するには? (OCU0, OCU1 での例) 627
22.7.5 コンペア端子 OCU0-OCU1 を出力に設定するには? 627
22.7.6 フリーランタイマをクリアするには? 627

22.7.7 コンペア動作許可するには?....627
22.7.8 割込み関連レジスタは?....628
22.7.9 割込みの種類は?....628
22.7.10 割込みを許可するには?....628
22.7.11 コンペア値の算出方法は?....629
22.8 サンプルプログラム....631
22.9 注意事項....633
23. アップダウンカウンタ....635
23.1 概要....635
23.2 特長....636
23.3 構成....638
23.4 レジスタ....640
23.4.1 リロードコンペアレジスタ (RCR0, RCR1)....641
23.4.2 アップダウンカウントレジスタ (UDCR0, UDCR1)....642
23.4.3 カウンタコントロールレジスタ (CCR0, CCR1)....643
23.4.4 カウンタステータスレジスタ (CSR0, CSR1)....648
23.5 割込み....650
23.6 動作説明と設定手順例....652
23.6.1 タイマモード時の動作....656
23.6.2 アップダウンカウントモード時の動作....658
23.6.3 位相差カウントモード (2 逓倍) 時の動作....661
23.6.4 位相差カウントモード (4 逓倍) 時の動作....663
23.6.5 BIN 端子....663
24. インプットキャプチャ....665
24.1 概要....665
24.2 特長....666
24.3 構成....666
24.4 レジスタ....667
24.4.1 インプットキャプチャデータレジスタ : IPCP....668
24.4.2 フリーランタイマ選択レジスタ : ICFS....669
24.4.3 インプットキャプチャ制御レジスタ : ICS....670
24.4.4 LIN SYNCH FIELD 切替えレジスタ : LSYNS....671
24.5 動作説明....673
24.5.1 取り込みタイミング, 割込みタイミング....674
24.5.2 インプットキャプチャのエッジ指定とその動作....675
24.6 設定....677
24.7 Q&A....678
24.7.1 外部入力の有効エッジ極性の種類と選択方法は?....678
24.7.2 外部入力端子 (ICU0, ICU1,ICU2,ICU3,ICU4,ICU5・・・) を有効にするには?....678
24.7.3 割込み関連レジスタは?....679
24.7.4 割込みの種類は?....679
24.7.5 割込みを許可するには?....679
24.7.6 入力信号のパルス幅を測定するには?....680
24.8 サンプルプログラム....681
24.9 注意事項....682
25. リアルタイムクロック (RTC)....683
25.1 概要....683
25.2 特長....684
25.3 構成....684
25.4 レジスタ....685
25.4.1 RTC 制御レジスタ : WTCR....686
25.4.2 サブセカンドレジスタ : WTBR....689
25.4.3 日時分秒 レジスタ : WTDR/WTHR/WTMR/WTSR....690
25.5 動作説明....692

25.6 設定 694
25.7 Q&A 695
25.7.1 0.5 秒のカウント周期を設定するには? 696
25.7.2 リアルタイムクロックを初期化するには? 696
25.7.3 日数 (日), 時刻 (時/分/秒) を設定/更新するには? 696
25.7.4 リアルタイムクロックのカウントを起動/停止するには? 696
25.7.5 リアルタイムクロックが動作しているか確認するには? 696
25.7.6 日数,時刻を知るには? 697
25.7.7 リアルタイムクロックを停止させるには? 697
25.7.8 リアルタイムクロックを補正するには？ 697
25.7.9 割込み関連レジスタは? 697
25.7.10 割込みの種類と選択方法は? 698
25.7.11 割込みを許可するには? 698
25.8 サンプルプログラム 699
25.9 注意事項 700
26. RTC／WDT1 補正 (キャリブレーション) 701
26.1 概要 701
26.2 特長 701
26.3 構成 702
26.4 レジスタ 703
26.4.1 補正ユニット制御レジスタ 0 : CUCR0: (Calibration Unit Control Register 0) 704
26.4.2 サブクロックタイマデータレジスタ : CUTD0 (Calibration Unit Timer Data register 0) 705
26.4.3 メイン発振タイマリザルトレジスタ 0 : CUTR0 (Calibration Unit Timer Result register 0) 706
26.4.4 補正ユニット制御レジスタ 1 : CUCR1 (Calibration Unit Control Register 1) 707
26.4.5 CR クロックタイマデータレジスタ : CUTD1 (Calibration Unit Timer Data register 1) 708
26.4.6 メイン発振タイマリザルトレジスタ 1 : CUTR1 (Calibration Unit Timer Result register 1) 709
26.4.7 CR 発振トリミング設定レジスタ : CRTR (CR oscillator calibration Trimming Register) 710
26.5 動作説明 711
26.5.1 リアルタイムクロック (RTC) 補正 711
26.5.2 WDT1 補正 (CR クロック補正) 711
26.5.3 注意事項 712
27. 消費電力制御 713
27.1 概要 713
27.2 特長 713
27.3 構成 714
27.4 レジスタ 716
27.4.1 スタンバイ制御レジスタ : STBCR (STandby mode Control Register) 716
27.4.2 PMU コントロールレジスタ : PMUCTLR (Power Management Unit ConTroL register) 718
27.4.3 Power on Timing コントロールレジスタ : PWRTMCTL (PoWeR on TiMing ConTroL register) 719
27.4.4 PMU 割込みフラグレジスタ 0 : PMUINTF0 (Power Management Unit INTerrupt Flag0 register) 720
27.4.5 PMU 割込みフラグレジスタ 1 : PMUINTF1 (Power Management Unit INTerrupt Flag1 register) 721
27.4.6 PMU 割込みフラグレジスタ 2 : PMUINTF2 (Power Management Unit INTerrupt Flag2 register) 722
27.5 動作説明 724
27.5.1 クロック制御 725
27.5.2 各低消費電力モードにおけるクロック供給一覧 725
27.5.3 スリープモード 726
27.5.4 スタンバイモード : 時計モード 728
27.5.5 スタンバイモード:時計モード (電源遮断) 730
27.5.6 スタンバイモード:ストップモード 733
27.5.7 スタンバイモード:ストップモード (電源遮断) 735

27.5.8 マイコン停止状態 738
27.5.9 不正スタンバイモード移行 738
27.5.10 電源遮断・通常スタンバイ制御の制限事項 739
27.6 使用例 742
28. 低電圧検出 (内部低電圧検出) 743
28.1 概要 743
28.2 特長 743
28.3 構成 744
28.4 レジスタ 744
28.4.1 内部低電圧検出レジスタ : LVD (Low Voltage Detect internal power fall register) 745
28.5 動作説明 747
28.5.1 内部低電圧検出 747
28.6 注意事項 748
29. 低電圧検出 (外部低電圧検出) 749
29.1 概要 749
29.2 特長 749
29.3 構成 750
29.4 レジスタ 751
29.4.1 外部低電圧検出立上り検出レジスタ : LVD5R (Low Voltage Detect external 5v Rise register) 751
29.4.2 外部低電圧検出立下り検出レジスタ : LVD5F (Low Voltage Detect external 5v Fall register) 752
29.5 動作説明 754
29.5.1 外部低電圧検出 754
29.6 注意事項 754
30. ワイルドレジスタ 755
30.1 概要 755
30.2 特長 755
30.3 構成 756
30.4 レジスタ 757
30.4.1 ワイルドレジスタデータイネーブルレジスタ : WREN (Wild Register data ENable register) 758
30.4.2 ワイルドレジスタアドレスレジスタ 00～15 : WRAR00-15 (Wild Register Address Register 00-15) 759
30.4.3 ワイルドレジスタデータレジスタ 00～15 : WRDR00-15 (Wild Register Data Register00-15) 760
30.5 動作説明 761
30.6 使用例 762
31. クロックスーパバイザ 763
31.1 概要 763
31.2 構成 763
31.3 レジスタ 765
31.4 動作説明 768
31.4.1 初期状態 769
31.4.2 CR 発振器およびクロックスーパバイザ機能の停止 770
31.4.3 クロックスーパバイザ再許可 771
31.4.4 サブクロックモード 772
31.4.5 ストップモード 773
31.4.6 時計モード 774
31.4.7 クロックスーパバイザによるリセット要因の確認 775
31.4.8 CR クロックからの切戻り 776
32. サウンドジェネレータ 777
32.1 概要 777
32.2 特長 778

32.3 構成 778
32.4 レジスタ 779
32.4.1 DMA 転送更新許可レジスタ ：SGDER (SG DMA Enable Register) 781
32.4.2 サウンドコントロールレジスタ ：SGCR (SG Control Register) 783
32.4.3 振幅データレジスタ ：SGAR (SG Amplitude Register) 786
32.4.4 周波数データレジスタ ：SGFR (SG Frequency Register) 787
32.4.5 トーン出力数レジスタ ：SGNR (SG tone Number Register) 788
32.4.6 周期レジスタ ：SGTCR (SG Tone Cycle Register) 789
32.4.7 増減量データレジスタ ：SGIDR (SG Increment Decrement Register) 790
32.4.8 PWM サイクル数データレジスタ ：SGPCR (SG PWM Cycle Register) 791
32.4.9 DMA 転送間接レジスタ ：SGDMAR (SG DMA Register) 792
32.5 動作説明 793
32.5.1 振幅データレジスタ (SGAR) と PWM パルスの関係 794
32.5.2 周波数データレジスタ (SGFR) とトーンパルス信号の関係 795
32.5.3 PWM サイクル数データレジスタ (SGPCR) と PWM サイクルの関係 796
32.5.4 DMA 転送更新許可レジスタ (SGDER) と DMA 転送回数/DMA 転送サイズ/転送バイト位置の関係について 797
32.5.5 サウンドジェネレータの動作 802
32.5.6 CPU によるサウンドジェネレータの連続動作 804
32.5.7 DMA と連携したサウンドジェネレータの動作 806
32.5.8 4 バイト×2 の DMA 転送を N 回行う場合 807
33. レギュレータ制御 819
33.1 概要 819
33.2 特長 819
33.3 構成 820
33.4 レジスタ 821
33.4.1 レギュレータ出力電圧選択レジスタ ：REGSEL (REGulator output voltage SELect register) 821
33.5 動作説明 822
34. ステッピングモータコントローラ 823
34.1 概要 823
34.2 特長 824
34.3 構成 824
34.4 レジスタ 824
34.4.1 PWM 制御レジスタ ：PWC 826
34.4.2 PWM1&2 コンペアレジスタ ：PWC1/PWC2 828
34.4.3 PWM1 選択レジスタ ：PWS1 829
34.4.4 PWM2 選択レジスタ ：PWS2 830
34.5 動作説明 831
34.5.1 PWM 動作 832
34.5.2 BS ビットによる PWM コンペアレジスタのロード動作 833
34.5.3 モータドライブ信号の選択 834
34.6 設定 835
34.7 Q&A 836
34.7.1 周期とデューティの設定方法は? 836
34.7.2 PWM 動作を許可/停止するには? 837
34.7.3 デューティの変更を反映させるには? 837
34.7.4 動作クロックの種類と選択方法は? 837
34.7.5 モータドライブ信号を変更したい場合は? 838
34.7.6 端子を PWM 出力端子にするには? 838
34.7.7 端子を A/D コンバータアナログ入力端子にするには? 838
34.8 サンプルプログラム 839
34.9 注意事項 840
35. LCD コントローラ 841

35.1 概要 841
35.2 特長 842
35.3 構成 843
35.4 レジスタ 844
35.4.1 LCD 制御レジスタ 0 : LCR0 844
35.4.2 表示用データメモリ : VRAM 846
35.4.3 LCDC 制御レジスタ 1 : LCR1 848
35.4.4 コモン端子切替えレジスタ : LCDCMR 848
35.4.5 LCDC スタティック制御レジスタ : LCRS 849
35.4.6 スタティック LCD ディスプレイデータレジスタ : LDR 851
35.5 動作説明 852
35.5.1 LCD コントローラ／ドライバ (LCDC) の動作 852
35.5.2 1/2 デューティ出力波形 854
35.5.3 1/3 デューティ出力波形 857
35.5.4 1/4 デューティ出力波形 859
35.5.5 スタティック駆動出力波形 862
35.6 設定 864
35.7 Q&A 866
35.7.1 端子を COM 出力端子, SEG 出力端子にするには? 867
35.7.2 VRAM の設定方法は? 869
35.7.3 フレーム周期を設定するには? 870
35.7.4 バイアスを設定するには? 871
35.7.5 デューティを設定するには? 871
35.7.6 LCD 動作開始／動作停止を制御するには? 871
35.7.7 表示を出す／消すには? 872
35.7.8 時計モード中に表示するには? 872
35.7.9 分割抵抗を内部か外部か選択するには? 873
35.7.10 V3 電圧の端子を選択するには? 873
35.7.11 分割抵抗を内部か外部か選択するには? 873
35.7.12 内部分割抵抗使用時に，輝度調整をするには? 875
35.7.13 外部分割抵抗で、LCD 停止時に電流を遮断させるには? 875
35.7.14 スタティック駆動(ST0～ST8)の LCD を表示／非表示にするには? 875
35.8 サンプルプログラム 876
35.9 注意事項 880
36. 外部バスインタフェース 881
36.1 概要 881
36.2 特長 882
36.3 構成 883
36.4 レジスタ 884
36.4.1 CS 領域設定レジスタ: ASR0～ASR3 (Area Setting Register 0-3) 885
36.4.2 CS バス設定レジスタ : ACR0～ACR3 (Area Configuration Register 0-3) 888
36.4.3 CS ウェイトレジスタ : AWR0～AWR3 (Area Wait Register 0-3) 890
36.4.4 外部 DMA 転送レジスタ: DMAR0-3 (DMA transfer Register 0-3) 895
36.5 動作説明 896
36.5.1 外部端子表 896
36.5.2 外バス信号プロトコル 897
36.5.3 アドレスアライメント 904
36.5.4 分割アクセス 904
36.5.5 データアライメント 905
36.5.6 アドレス情報 908
36.5.7 アイドルサイクル挿入機能 909
36.5.8 外バス出力信号タイミング設定 910
36.5.9 RDY 端子アクセスサイクル延長機能 915
36.5.10 CS 設定フロー 916

36.5.11 非同期メモリとの接続例 923
36.5.12 リトルエンディアンデバイスとの接続例 924
37. バス・パフォーマンス・カウンタ 925
37.1 概要 925
37.2 特長 925
37.3 構成 926
37.4 レジスタ 927
37.4.1 BPC-A 制御レジスタ ：BPCCRA (Bus Performance Counter Control Register A) 928
37.4.2 BPC-B 制御レジスタ ：BPCCRB (Bus Performance Counter Control Register B) 929
37.4.3 BPC-C 制御レジスタ ：BPCCRC (Bus Performance Counter Control Register C) 929
37.4.4 BPC-A カウントレジスタ ：BPCTRA (Bus Performance CounTer Register A) 930
37.4.5 BPC-B カウントレジスタ ：BPCTRB (Bus Performance CounTer Register B) 930
37.4.6 BPC-C カウントレジスタ ：BPCTRC (Bus Performance CounTer Register C) 931
37.5 動作説明 932
37.5.1 設定 933
37.5.2 起動と停止 934
37.5.3 動作 934
37.5.4 測定と結果処理 935
38. CRC 937
38.1 概要 937
38.2 特長 937
38.3 構成 938
38.4 レジスタ 938
38.4.1 CRC 制御レジスタ ：CRCCR (CRC Control Register) 939
38.4.2 CRC 初期値レジスタ ：CRCINIT (CRC Initial value register) 940
38.4.3 Input Data レジスタ ：CRCIN (CRC INput data register) 940
38.4.4 CRC レジスタ ：CRCR (CRC Register) 941
38.5 動作説明 942
38.5.1 CRC の定義 942
38.5.2 リセット動作 943
38.5.3 初期化 943
38.5.4 バイトオーダと、ビットオーダ 943
38.5.5 CRC 計算シーケンス 944
38.5.6 使用例 945
39. RAMECC 951
39.1 概要 951
39.2 特長 951
39.3 構成 952
39.4 レジスタ 954
39.4.1 ECC エラー制御レジスタ XBS RAM : EECSRX (Ecc Error Control and Status Register Xbs ram) 955
39.4.2 シングルビット ECC エラーアドレスレジスタ XBS RAM : SEEARX (Single bit Ecc Error Address Register Xbs ram) 956
39.4.3 ダブルビット ECC エラーアドレスレジスタ XBS RAM : DEEARX (Double bit Ecc Error Address Register Xbs ram) 957
39.4.4 ECC 疑似エラー発生アドレスレジスタ XBS RAM : EFEARX (Ecc False Error Address Register Xbs ram) 958
39.4.5 ECC 疑似エラー発生制御レジスタ XBS RAM : EFECRX (Ecc False Error Control Register Xbs ram) 959
39.4.6 ECC エラー制御レジスタ BACKUP-RAM : EECSRA (Ecc Error Control and Status Register bAckup-ram) 961
39.4.7 シングルビット ECC エラーアドレスレジスタ BACKUP-RAM : SEEARA (Single bit Ecc Error Address Register bAckup-ram) 962

39.4.8 ダブルビット ECC エラーアドレスレジスタ BACKUP-RAM : DEEARA (Double bit Ecc Error Address Register bAckup-ram) 963
39.4.9 ECC 疑似エラー発生アドレスレジスタ BACKUP-RAM : EFEARA (Ecc False Error Address Register bAckup-ram) 964
39.4.10 ECC 疑似エラー発生制御レジスタ BACKUP-RAM : EFECRA (Ecc False Error Control Register bAckup-ram) 965
39.5 動作説明 967
39.5.1 ECC 生成 967
39.5.2 ECC 検査 968
39.5.3 エラー検出による割込み 969
39.5.4 試験機能 970
40. マルチファンクションシリアルインタフェース 971
40.1 概要 971
40.2 特長 972
40.2.1 UART 972
40.2.2 CSIO 973
40.2.3 LIN-UART 974
40.2.4 I²C 975
40.2.5 注意事項 975
40.3 構成 976
40.4 レジスタ 980
40.4.1 共通レジスタ 984
40.4.2 UART 時のレジスタ 995
40.4.3 CSIO 時のレジスタ 1007
40.4.4 LIN-UART 時のレジスタ 1016
40.4.5 I²C 時のレジスタ 1025
40.5 UART の動作説明 1043
40.5.1 UART の割込み 1043
40.5.2 UART の動作 1051
40.5.3 設定手順とプログラムフロー 1063
40.6 CSIO の動作説明 1068
40.6.1 CSIO の割込み 1068
40.6.2 CSIO の動作 1075
40.6.3 設定手順とプログラムフロー 1096
40.7 LIN-UART の動作説明 1098
40.7.1 LIN-UART の割込み 1098
40.7.2 LIN-UART の動作 1106
40.7.3 設定手順とプログラムフロー 1116
40.8 I²C の動作説明 1121
40.8.1 I²C の割込み 1121
40.8.2 I²C インタフェース通信の動作 1123
40.8.3 I²C マスタモード 1126
40.8.4 I²C スレーブモード 1160
40.8.5 バスエラー 1169
40.8.6 I²C のフローチャート例 1170
41. LIN-UART 1177
41.1 概要 1177
41.2 特長 1178
41.2.1 機能 1178
41.2.2 動作モード 1179
41.3 構成 1180
41.3.1 LIN-UART のブロックダイヤグラム 1180
41.3.2 各ブロックの説明 1182
41.4 レジスタ 1187

41.4.1 シリアルコントロールレジスタ (SCR) 1189
41.4.2 シリアルモードレジスタ (SMR) 1191
41.4.3 シリアルステータスレジスタ (SSR) 1193
41.4.4 受信／送信データレジスタ (RDR/TDR) 1196
41.4.5 拡張ステータス制御レジスタ (ESCR) 1198
41.4.6 拡張通信コントロールレジスタ (ECCR) 1201
41.4.7 ボーレートジェネレータレジスタ (BGR) 1203
41.5 割込み 1204
41.5.1 概要 1204
41.5.2 受信割込み生成とフラグセットタイミング 1209
41.5.3 送信割込み生成とフラグタイミング 1211
41.6 ボーレート 1213
41.6.1 ボーレート選択 1213
41.6.2 ボーレートの設定 1215
41.6.3 リロードカウンタ 1218
41.7 動作説明 1220
41.7.1 概要 1220
41.7.2 非同期モード (動作モード 0, 1) 1223
41.7.3 同期モード (動作モード 2) 1227
41.7.4 LIN モード (動作モード 3) 1232
41.7.5 シリアル端子への直接アクセス 1237
41.7.6 双方向通信機能 (ノーマルモード) 1238
41.7.7 マスタ／スレーブ通信機能 (マルチプロセッサモード) 1240
41.7.8 LIN 通信機能 1243
41.7.9 LIN 通信モード (動作モード 3) LIN-UART サンプルフローチャート 1244
41.8 使用上の注意事項 1247
41.8.1 動作許可 1247
41.8.2 通信モードの設定 1247
41.8.3 送信割込み許可のタイミング 1247
41.8.4 動作設定の変更 1247
41.8.5 LIN synch break 検出 1248
41.8.6 LIN スレーブ設定 1248
41.8.7 プログラムの互換性 1248
41.8.8 アドレス/データ形式選択ビット (SCR:AD) 1248
41.8.9 LIN-UART ソフトウェアリセット 1248
41.8.10 LIN Synch Field のインプットキャプチャでの検出について 1249
41.8.11 バスアイドル検出機能 1249
41.9 DMAC 連携動作での注意 1250
41.9.1 送信動作 1250
41.9.2 受信動作 1250
42. CAN 1251
42.1 概要 1251
42.2 特長 1251
42.3 構成 1252
42.4 レジスタ 1253
42.4.1 概要 1253
42.4.2 全体コントロールレジスタ 1259
42.4.3 メッセージインタフェースレジスタ 1274
42.4.4 メッセージオブジェクト 1286
42.4.5 メッセージハンドラレジスタ 1292
42.4.6 CAN プリスケーラレジスタ (CANPRE) 1301
42.5 動作説明 1303
42.5.1 メッセージオブジェクト 1303
42.5.2 メッセージ送信動作 1305

42.5.3 メッセージ受信動作 1308
42.5.4 FIFO バッファ機能 1312
42.5.5 割込み機能 1315
42.5.6 ビットタイミング・CAN システムクロック (fsys) 生成 1316
42.5.7 テストモード 1318
42.5.8 ソフトウェア初期化 1323
42.6 制限事項 1325
42.6.1 INIT ビット 1325
43. D/A コンバータ 1329
43.1 概要 1329
43.2 特長 1329
43.3 構成 1330
43.4 レジスタ 1330
43.4.1 DA コントロールレジスタ : DACR (DA Control Register) 1331
43.4.2 DA データレジスタ : DADR (DA DATA Register) 1332
43.5 動作説明 1333
43.6 注意事項 1334
44. A/D コンバータ 1335
44.1 概要 1335
44.2 特長 1336
44.3 構成 1337
44.4 レジスタ 1337
44.4.1 アナログ入力許可レジスタ : ADER 1338
44.4.2 A/D 制御ステータスレジスタ (上位) : ADCS1 1339
44.4.3 A/D 制御ステータスレジスタ (下位) : ADCS0 1341
44.4.4 データレジスタ : ADCR0,ADCR1 1342
44.4.5 変換時間設定レジスタ : ADCT 1343
44.4.6 A/D 開始/終了チャネル設定レジスタ : ADSCH, ADECH 1345
44.4.7 拡張アナログ入力許可レジスタ : EADERLL 1347
44.4.8 拡張 A/D 制御ステータスレジスタ : EADCS 1347
44.5 動作説明 1349
44.5.1 シングル変換動作 1350
44.5.2 スキャン変換動作 1351
44.5.3 変換モード 1352
44.6 設定 1354
44.7 Q&A 1356
44.7.1 変換モードの種類と設定方法は? 1357
44.7.2 ビット長を指定するには? 1357
44.7.3 チャネルを選択するには? 1358
44.7.4 変換時間を設定するには? 1360
44.7.5 アナログ端子入力を有効にするには? 1361
44.7.6 A/D コンバータの起動方法を選択するには? 1362
44.7.7 A/D コンバータを起動するには? 1362
44.7.8 変換終了を確認するには? 1363
44.7.9 変換値を読み出すには? 1363
44.7.10 A/D 変換動作を強制的に停止させるには? 1363
44.7.11 割込み関連レジスタは?? 1364
44.7.12 割込みの種類は? 1364
44.7.13 割込みを許可/禁止/クリアするには? 1364
44.8 サンプルプログラム 1365
44.9 注意事項 1368
44.10 A/D コンバータの用語の定義 1368
45. フラッシュメモリ 1371
45.1 概要 1371

45.2 特長 1372
45.3 構成 1373
45.3.1 ブロックダイヤグラム 1373
45.3.2 セクタ構成図 1374
45.3.3 セクタ番号・フラッシュマクロ番号対応表 1378
45.4 レジスタ 1380
45.4.1 フラッシュ制御レジスタ ：FCTLR (Flash ConTroL Register) 1381
45.4.2 フラッシュステータスレジスタ ：FSTR (Flash STatus Register) 1383
45.4.3 フラッシュインタフェース制御レジスタ ：FLIFCTLR(Flash I/F Control Register) 1385
45.4.4 フラッシュインタフェースフィーチャ拡張レジスタ 1 : FLIFFER1(Flash I/F Feature Extension Register 1) 1387
45.4.5 フラッシュインタフェースフィーチャ拡張レジスタ 2 : FLIFFER2(Flash I/F Feature Extension Register 2) 1387
45.5 動作説明 1388
45.5.1 アクセスモード設定 1389
45.5.2 CPU によるフラッシュメモリ書込み 1390
45.5.3 自動アルゴリズム 1391
45.5.4 リセットコマンド 1398
45.5.5 書込みコマンド 1398
45.5.6 チップ消去コマンド 1401
45.5.7 セクタ消去コマンド 1401
45.5.8 セクタ消去一時停止コマンド 1403
45.5.9 セキュリティ機能 1404
45.5.10 フラッシュメモリの使用上の注意 1406
46. ワークフラッシュメモリ 1407
46.1 概要 1407
46.2 特長 1408
46.3 構成 1409
46.3.1 ブロックダイヤグラム 1409
46.3.2 セクタ構成図 1410
46.4 レジスタ 1411
46.4.1 ワークフラッシュ制御レジスタ ：DFCTLR (WorkFlash ConTroL Register) 1412
46.4.2 ワークフラッシュステータスレジスタ ：DFSTR (WorkFlash STatus Register) 1413
46.4.3 フラッシュインタフェース制御レジスタ ：FLIFCTLR (Flash I/F Control Register) 1415
46.5 動作説明 1417
46.5.1 アクセスモード設定 1418
46.5.2 CPU によるフラッシュメモリ書込み 1419
46.5.3 自動アルゴリズム 1420
46.5.4 リセットコマンド 1427
46.5.5 書込みコマンド 1427
46.5.6 チップ消去コマンド 1430
46.5.7 セクタ消去コマンド 1430
46.5.8 セクタ消去一時停止コマンド 1432
46.5.9 セキュリティ機能 1433
46.5.10 フラッシュメモリの使用上の注意 1434
47. HS_SPI 1435
47.1 概要 1435
47.2 特長 1435
47.3 構成 1435
47.3.1 ブロックダイヤグラム 1436
47.3.2 用語/略語 1437
47.4 レジスタ 1438
47.4.1 HS_SPI モジュール制御レジスタ ：HSSPIn_MCTRL (HS_SPI Module Control Register) .1441

47.4.2 HS_SPI ペリフェラル通信設定レジスタ 0~3 : HSSPIn_PCC0~3 (HS_SPI Peripheral Communication Configuration Register 0~3) 1443
47.4.3 HS_SPI TX 割込みフラグレジスタ (HSSPIn_TXF) 1446
47.4.4 HS_SPI TX 割込み許可レジスタ (HSSPIn_TXE) 1448
47.4.5 HS_SPI TX 割込みクリアレジスタ (HSSPIn_TXC) 1450
47.4.6 HS_SPI RX 割込みフラグレジスタ (HSSPIn_RXF) 1452
47.4.7 HS_SPI RX 割込み許可レジスタ (HSSPIn_RXE) 1454
47.4.8 HS_SPI RX 割込みクリアレジスタ (HSSPIn_RXC) 1456
47.4.9 HS_SPI フォルト割込みフラグレジスタ (HSSPIn_FAULTF) 1458
47.4.10 HS_SPI フォルト割込みクリアレジスタ (HSSPIn_FAULTC) 1459
47.4.11 HS_SPI ダイレクトモード設定レジスタ (HSSPIn_DMCFG) 1460
47.4.12 HS_SPI ダイレクトモード DMA 許可レジスタ (HSSPIn_DMDMAEN) 1461
47.4.13 HS_SPI ダイレクトモードスタートレジスタ (HSSPIn_DMSTART) 1462
47.4.14 HS_SPI ダイレクトモードストップレジスタ(HSSPIn_DMSTOP) 1463
47.4.15 HS_SPI ダイレクトモードペリフェラル選択レジスタ (HSSPIn_DMPSEL) 1464
47.4.16 HS_SPI ダイレクトモード転送プロトコルレジスタ (HSSPIn_DMTRP) 1465
47.4.17 HS_SPI ダイレクトモードバイトカウント制御レジスタ(HSSPIn_DMBCC) 1466
47.4.18 HS_SPI ダイレクトモードバイトカウントステータスレジスタ (HSSPIn_DMBCS) 1467
47.4.19 HS_SPI ダイレクトモードステータスレジスタ(HSSPIn_DMASTATUS) 1468
47.4.20 HS_SPI 送信ビットカウントレジスタ (HSSPIn_TXBITCNT) 1469
47.4.21 HS_SPI 受信ビットカウントレジスタ (HSSPIn_RXBITCNT) 1470
47.4.22 HS_SPI RX シフトレジスタ (HSSPIn_RXSHIFT) 1471
47.4.23 HS_SPI TX-FIFO レジスタ (HSSPIn_TXFIFO0~15) 1472
47.4.24 HS_SPI RX-FIFO レジスタ (HSSPIn_RXFIFO0~15) 1473
47.4.25 HS_SPI FIFO 設定レジスタ (HSSPIn_FIFOCFG) 1474
47.4.26 HS_SPI コマンドシーケンサ設定レジスタ (HSSPIn_CSCFG) 1476
47.4.27 HS_SPI コマンドシーケンサアイドルタイムレジスタ (HSSPIn_CSITIME) 1478
47.4.28 HS_SPI コマンドシーケンサアドレス拡張レジスタ (HSSPIn_CSAEXT) 1479
47.4.29 HS_SPI リードコマンドシーケンスデータ/制御レジスタ 0~7 (HSSPIn_RDCSDC0~7) 1480
47.4.30 HS_SPI ライトコマンドシーケンスデータ/制御レジスタ 0~7 (HSSPIn_WRCSDC0~7) 1482
47.4.31 HS_SPI アドレススワップ制御レジスタ (HSSSWPR) 1484
47.5 機能説明 1486
47.5.1 HS_SPI モードの動作と設定 1486
47.5.2 ダイレクトモード 1499
47.5.3 コマンドシーケンサモード 1505
47.6 HS_SPI の使用上の注意事項 1513
47.6.1 一般的な使用上の注意事項 1513
47.6.2 HS_SPI モジュールのプログラム手順 1514
47.6.3 ダイレクトモード動作での HS_SPI の注意事項 1515
47.6.4 メモリ領域に割り当てて使用する場合 1521
47.7 注意事項 1525
47.7.1 概要 1525
47.7.2 例 1 : 2 つの SPI 転送の開始 1527
47.7.3 例 2 : 転送におけるプロトコル切換え 1528
48. オンチップデバッガ (OCD) 1529
48.1 概要 1529
48.2 特長 1530
48.3 構成 1531
48.3.1 DEBUG I/F 用クロック 1533
48.4 レジスタ 1534
48.4.1 DBG レジスタ 1534
48.4.2 ユーザ IO レジスタ 1535
48.5 動作説明 1536
48.5.1 OCDU 動作モード 1536

48.5.2 DEBUG I/F 概要 1538
48.5.3 本シリーズの OCD ツール接続時での仕様制限 1542
48.5.4 本シリーズの OCD-DSU ID コードおよび実装タイプ情報 1549
49. 付録 1551
49.1 メモリマップ 1552
49.2 I/O マップ 1554
49.3 割込みベクター覧 1595
49.4 CPU 状態における端子状態 1598
49.5 主な変更内容 1599
改訂履歴 1600

1. 概要

概要について説明します。

1.1 概要

本シリーズの特長と基本的な仕様について説明します。

CY91F575/577/578/579は自動車用ダッシュボード制御向けのサイプレス32ビットマイクロコントローラです。CPUには、FRファミリと互換のFR81S CPUを使用しています。

1.2 特長

本シリーズの特長について説明します。

1.2.1. FR81S CPU コア

1.2.2. 周辺機能

1.2.1 FR81S CPU コア

FR81S CPUコアについて説明します。

- 32 ビット RISC、ロード/ストアアーキテクチャ、パイプライン 5 段
- 最大動作周波数: 80MHz (原振=4.0MHz, 20 逓倍(PLL クロック逓倍方式))
- 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
- 組込み用途に適した命令
  - メモリ→メモリ間転送命令
  - ビット処理命令
  - バレルシフト命令など
- 高級言語対応命令
  - 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
- ビットサーチ命令
  - 1 検出、0 検出、変化点検出
- 遅延スロット付き分岐命令
  - 分岐処理時のオーバヘッドの低減
- レジスタインターロック機能
  - アセンブラ記述の容易化
- 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
- 割込み (PC/PS 退避)
  - 6 サイクル (16 プライオリティレベル)
- ハーバードアーキテクチャにより、プログラムアクセスとデータアクセスを同時に実行可能
- FR ファミリとの命令互換
- メモリ保護機能 (MPU) 搭載
  - 命令・データ共用で 8 個の保護領域指定
  - 特権モード・ユーザモードそれぞれでアクセス権を制御
- FPU(浮動小数点演算)搭載
  - IEEE754 準拠
  - 浮動小数点レジスタ 32 ビット× 16 本

1.2.2 周辺機能

本シリーズの周辺機能について説明します。

- クロック生成 (SSCG 機能搭載)
  - メイン発振 (4 MHz)
  - サブ発振 (32kHz) またはサブ発振なし
  - PLL 逓倍率 : 1～20 逓倍
- 内蔵プログラム用フラッシュメモリ容量
  - CY91F575 : 512+64KB
  - CY91F577 : 1024+64KB
  - CY91F578 : 1536+64KB
  - CY91F579 : 2048+64KB
- 内蔵データ用フラッシュメモリ (ワークフラッシュ) 64KB
- 内蔵 RAM 容量
  - メイン RAM
    - CY91F575 : 40KB
    - CY91F577 : 64KB
    - CY91F578 : 96KB
    - CY91F579 : 128KB
  - BackupRAM
    - CY91F575 : 8KB
    - CY91F577 : 8KB
    - CY91F578 : 16KB
    - CY91F579 : 16KB
- 汎用ポート:
  - LQFP-144 パッケージ品 : 111 本(サブ発振なし)、109 本 (サブ発振あり)
  - LQFP-208 パッケージ品 : 159 本(サブ発振なし)、157 本 (サブ発振あり)
  - うち I^2C 用擬似オープンドレイン対応ポート 4 本
  - P057 は入力専用
- 外部バスインタフェース
  - 22 ビットアドレス、16 ビットデータ
  - 9 ビットアドレス、8 ビットデータ、ASX,CS0X,CS1X,RDX,WR0X,WR1X の 23 端子は、VCCE 電源により、5V/3.3V 選択可能
- DMA コントローラ
  - 同時に最大 16 チャネルの起動が可能
  - 2 つの転送要因(内部周辺要求/ソフトウェア)
- A/D コンバータ (逐次比較型)
  - 8/10 ビット分解能: 40 チャネル
  - 変換時間: 3µs
- D/A コンバータ (R-2R 型)
  - 8bit 分解能: 2 チャネル
- 外部割込み入力: 16 チャネル
  - レベル("H" / "L")、エッジ検出(立上り/立下り)可能
- LIN-UART:
  - 6 チャネル ch.2～ch.7
  - UART、同期モード、LIN-UART モードから選択
  - LIN プロトコル Revision 2.1 に対応 (LIN-UART)
  - SPI(Serial Peripheral Interface)対応 (同期モード)
  - 全二重ダブルバッファ方式
  - LIN synch break 検出(インプットキャプチャと連携)

- □ 専用ボーレートジェネレータ内蔵
- □ DMA 転送対応

■ マルチファンクションシリアル (送受信 FIFO 搭載) : 4 チャネル

< UART (非同期シリアルインタフェース) >

- □ 全二重ダブルバッファ方式、16 段の送信 FIFO、16 段の受信 FIFO
- □ パリティあり/なし選択可能
- □ 専用ボーレートジェネレータ内蔵
- □ 外部クロックを転送クロックとして使用可能
- □ パリティ、フレーム、オーバランエラー検出機能あり
- □ DMA 転送対応

<CSIO (同期シリアルインタフェース) >

- □ 全二重ダブルバッファ方式、16 段の送信 FIFO、16 段の受信 FIFO
- □ SPI に対応、マスタ/スレーブ両方に対応、データ長 5～9 ビットに設定可能
- □ 専用ボーレートジェネレータ内蔵(マスタ動作)
- □ 外部クロック入力可能(スレーブ動作)
- □ オーバランエラー検出機能あり
- □ DMA 転送対応

<LIN-UART (LIN対応非同期シリアルインタフェース) >

- □ 全二重ダブルバッファ方式、16 段の送信 FIFO、16 段の受信 FIFO
- □ LIN プロトコル Revision2.1 に対応
- □ マスタ／スレーブ両方に対応
- □ フレーミングエラー、オーバランエラー検出
- □ LIN Synch break 生成、検出、LIN Synch Delimiter 生成
- □ 専用ボーレートジェネレータ内蔵
- □ 外部クロックをリロードカウンタで調整可能
- □ DMA 転送対応

< I^2C >

- □ 全二重ダブルバッファ方式、16 段の送信 FIFO、16 段の受信 FIFO
- □ 標準モード(Max. 100kbps) / 高速モード(Max. 400kbps) に対応
- □ 送信のみ DMA 転送対応
- □ I^2C 対応 I/O は ch.0, ch.1 のみ

■ CAN コントローラ CAN : 3 チャネル

- □ 転送速度 最大 1Mbps
- □ 64 送受信メッセージバッファ 1 チャネル、32 送受信メッセージバッファ 2 チャネル

■ PPG: 16 ビット×24 チャネル

■ リロードタイマ: 16 ビット×7 チャネル (うち 3 チャネルは定期タイマ割込み発生用)

■ フリーランタイマ:
32 ビット×6 チャネル (インプットキャプチャ 用、アウトプットコンペア 用各 1 チャネル選択可能)

■ インプットキャプチャ:
32 ビット×12 チャネル (フリーランタイマと連動)

■ アウトプットコンペア: 32 ビット× 12 チャネル (フリーランタイマと連動)

■ サウンドジェネレータ: 5 チャネル

- □ 周波数・振幅シーケンサあり

■ ステッピングモータコントローラ: 6 チャネル

- □ 8/10 ビット PWM
- □ 大電流出力サポート(4 本×6 チャネル)

- □ 端子共有 A/D コンバータでモータからの逆起電力を参照可能
- ■ LCD コントローラ
  - □ コモン出力(4 本)、セグメント出力(32 本)
  - □ デューティ駆動(SEG0～SEG31) /スタティック駆動(ST0 ～ ST8) 切換え可能
  - □ デューティ駆動の COM0～COM3, SEG0～SEG31, V0, V1, V2, V3 端子は 1 端子ごとに汎用ポートへの切換えが可能(SEG23～SEG31 は、スタティック駆動への切換え可能)
  - □ V0, V1, V2, V3 端子すべて、ポート兼用、ただし V3 と兼用のポートは入力のみ
  - □ スタティック駆動の ST0～ST8 端子は、1 端子ごとに汎用ポートまたは、デューティ駆動のセグメント出力への切換えが可能
  - □ CY91F575/577: VCCE 電源が 3.3V の場合も、SEG0～SEG22 は、V3 端子に供給される電源または VCC5 電源の振幅で出力されます。
  - □ CY91F578/579: V3 入力端子には VCCE 以下の電圧のみ供給可能です。VCC5 を LCDC 基準電源としてソフトウェアで選択することは禁止です。
- ■ アップダウンカウンタ: 2 チャネル
  - □ 8/16bit アップダウンカウンタ
- ■ リアルタイムクロック(RTC) (日/時/分/秒)
  - □ 動作クロックとしてメイン発振/サブ発振を選択可能
- ■ キャリブレーション:CR 発振駆動のハードウェアウォッチドッグと、サブクロック駆動のリアルタイムクロック(RTC)
  - □ CR 発振周波数をトリミング可能
  - □ メインクロック対サブクロックの比で、リアルタイムクロックのプリスケーラを設定することにより補正
- ■ クロックスーパバイザ
  - □ 外部のサブ発振(32kHz)(クロック 2 系統品種)およびメイン発振(4 MHz)の異常(水晶の破損など)監視
  - □ 異常検出時には CR クロックに切り換える。
- ■ ベースタイマ: 2 チャネル
  - □ 16 ビットタイマ
  - □ PWM / PPG / PWC / リロードタイマの 4 機能を選択して使用可能
  - □ PWC 機能とリロードタイマ機能は、2 チャネルをカスケードモードで 32 ビットタイマとして使用可能
- ■ CRC 生成
- ■ HS-SPI

  (注意事項)　本シリーズでは HS_SPI は使用禁止です
  - □ Single/Dual/Quad-SPI プロトコルの E^2PROM および Flash デバイスを接続可能
  - □ VCCE 電源端子に供給される、5V/3.3V の電源を使用
  - □ 最大 16MHz　スレーブ時は最大 8MHz
- ■ ウォッチドッグタイマ
  - □ ハードウェアウォッチドッグ
  - □ ソフトウェアウォッチドッグ
- ■ NMI
- ■ 割込みコントローラ
- ■ 割込み要求一括読出し
  - □ 複数の周辺からの割込み有無を、一連のレジスタで読出し可能
- ■ I/O リロケーション
  - □ 周辺機能の端子位置変更
- ■ 低消費電力モード
  - □ スリープ/ストップ/時計/サブ RUN モード
  - □ ストップ(電源遮断)/時計(電源遮断)モード
- ■ パワーオンリセット/内部低電圧検出リセット
- ■ 低電圧検出リセット

- パッケージ:
  - LQFP-144: CY91F575/577/578/579
  - LQFP-208: CY91F578/579
- CMOS 90nm テクノロジ
- 電源
  - 5V 電源
  - 降圧回路により 5V から内部 1.2V 生成
  - 以下の I/O は VCCE 電源端子に供給される、5V/3.3V の 電源を使用
    LQFP-144: P010～P017、P020～P027、P030～P036
    LQFP-208: P140～P147、P150～P157、P160～P167、P170～P177、P180～P187、P190～P197

1.3 品種構成

本シリーズの品種構成を示します。

表 1-1 品種構成

CY 番号	CY91F575B/F575BS	CY91F575BH/F575BHS
システムクロック	オンチップ PLL クロック逓倍方式	
最小命令実行時間	12.5ns (80MHz)	
サブクロック	S なし型格:あり S あり型格:なし	
FLASH 容量(プログラム)	512+64KB	
FLASH 容量(データ)	64KB	
RAM	40KB + 8KB	
BI-ROM	4KB	
GDC	なし	
外部バス I/F	アドレス:22bit　データ:16bit (一部の端子は、電源として 5V/3.3V 選択可能)	
DMA 転送	16 チャネル	
16bit ベースタイマ	2 チャネル	
フリーランタイマ	6 チャネル	
インプットキャプチャ	12 チャネル	
アウトプットコンペア	12 チャネル	
16bit リロードタイマ	7 チャネル	
PPG	24 チャネル	
アップダウンカウンタ	2 チャネル	
クロックスーパバイザ	あり	
D/A	2 チャネル	
外部割込み	16 チャネル	
A/D	40 チャネル	
LIN-UART	6 チャネル	
マルチファンクション	4 チャネル*1	
HS-SPI	あり 最大 16MHz (注意事項) 本シリーズでは HS_SPI は使用禁止です	
LCD コントローラ	32seg × 4com スタティック駆動 8seg × 1com	
CAN	64msg ×1 チャネル / 32msg ×2 チャネル	
ステッピングモータコントローラ	6 チャネル	
サウンドジェネレータ	5 チャネル	
ソフトウェアウォッチドッグ	あり	
ハードウェアウォッチドッグ	あり	
クロックスーパバイザ	初期値 ON	初期値 OFF
CRC 生成	あり	
低電圧検出リセット(外部低電圧検出)	あり	
内部低電圧検出リセット	あり	
パッケージ	LQFP-144	
その他	FLASH 品	
On Chip Debug	OCD 搭載	

*1: I^2C対応はch.0, ch.1のみ

CY 番号	CY91F577B/F577BS	CY91F577BH/F577BHS
システムクロック	オンチップ PLL クロック逓倍方式	
最小命令実行時間	12.5ns (80MHz)	
サブクロック	S なし型格:あり S あり型格:なし	
FLASH 容量(プログラム)	1024+64KB	
FLASH 容量(データ)	64KB	
RAM	64KB + 8KB	
BI-ROM	4KB	
GDC	なし	
外部バス I/F	アドレス:22bit　データ:16bit (一部の端子は、電源として 5V/3.3V 選択可能)	
DMA 転送	16 チャネル	
16bit ベースタイマ	2 チャネル	
フリーランタイマ	6 チャネル	
インプットキャプチャ	12 チャネル	
アウトプットコンペア	12 チャネル	
16bit リロードタイマ	7 チャネル	
PPG	24 チャネル	
アップダウンカウンタ	2 チャネル	
クロックスーパバイザ	あり	
D/A	2 チャネル	
外部割込み	16 チャネル	
A/D	40 チャネル	
LIN-UART	6 チャネル	
マルチファンクション	4 チャネル*1	
HS-SPI	あり 最大 16MHz (注意事項) 本シリーズでは HS_SPI は使用禁止です	
LCD コントローラ	32seg × 4com スタティック駆動 8seg × 1com	
CAN	64msg ×1 チャネル / 32msg ×2 チャネル	
ステッピングモータコントローラ	6 チャネル	
サウンドジェネレータ	5 チャネル	
ソフトウェアウォッチドッグ	あり	
ハードウェアウォッチドッグ	あり	
クロックスーパバイザ	初期値 ON	初期値 OFF
CRC 生成	あり	
低電圧検出リセット (外部低電圧検出)	あり	
内部低電圧検出リセット	あり	
パッケージ	LQFP-144	
その他	FLASH 品	
On Chip Debug	OCD 搭載	

*1: I^2C対応はch.0, ch.1のみ

CY 番号	CY91F578C(M)/F578CS(M)	CY91F578CH(M)/F578CHS(M)
システムクロック	オンチップ PLL クロック逓倍方式	
最小命令実行時間	12.5ns (80MHz)	
サブクロック	S なし型格:あり S あり型格:なし	
FLASH 容量(プログラム)	1536+64KB	
FLASH 容量(データ)	64KB	
RAM	96KB + 16KB	
BI-ROM	4KB	
GDC	なし	
外部バス I/F	アドレス:22bit　データ:16bit (一部の端子は、電源として 5V/3.3V 選択可能)	
DMA 転送	16 チャネル	
16bit ベースタイマ	2 チャネル	
フリーランタイマ	6 チャネル	
インプットキャプチャ	12 チャネル	
アウトプットコンペア	12 チャネル	
16bit リロードタイマ	7 チャネル	
PPG	24 チャネル	
アップダウンカウンタ	2 チャネル	
クロックスーパバイザ	あり	
D/A	2 チャネル	
外部割込み	16 チャネル	
A/D	40 チャネル	
LIN-UART	6 チャネル	
マルチファンクション	4 チャネル*1	
HS-SPI	なし	
LCD コントローラ	32seg × 4com スタティック駆動 8seg × 1com	
CAN	64msg ×1 チャネル / 32msg ×2 チャネル	
ステッピングモータコントローラ	6 チャネル	
サウンドジェネレータ	5 チャネル	
ソフトウェアウォッチドッグ	あり	
ハードウェアウォッチドッグ	あり	
クロックスーパバイザ	初期値 ON	初期値 OFF
CRC 生成	あり	
低電圧検出リセット (外部低電圧検出)	あり	
内部低電圧検出リセット	あり	
パッケージ	LQFP-144, LQFP-208 (with Suffix "M")	
その他	FLASH 品	
On Chip Debug	OCD 搭載	

*1: I^2C対応はch.0, ch.1のみ

CY 番号	CY91F579C(M)/F579CS(M)	CY91F579CH(M)/F579CHS(M)
システムクロック	オンチップ PLL クロック逓倍方式	
最小命令実行時間	12.5ns (80MHz)	
サブクロック	S なし型格:あり S あり型格:なし	
FLASH 容量(プログラム)	2048+64KB	
FLASH 容量(データ)	64KB	
RAM	128KB + 16KB	
BI-ROM	4KB	
GDC	なし	
外部バス I/F	アドレス:22bit　データ:16bit (一部の端子は、電源として 5V/3.3V 選択可能)	
DMA 転送	16 チャネル	
16bit ベースタイマ	2 チャネル	
フリーランタイマ	6 チャネル	
インプットキャプチャ	12 チャネル	
アウトプットコンペア	12 チャネル	
16bit リロードタイマ	7 チャネル	
PPG	24 チャネル	
アップダウンカウンタ	2 チャネル	
クロックスーパバイザ	あり	
D/A	2 チャネル	
外部割込み	16 チャネル	
A/D	40 チャネル	
LIN-UART	6 チャネル	
マルチファンクション	4 チャネル*1	
HS-SPI	なし	
LCD コントローラ	32seg × 4com スタティック駆動 8seg × 1com	
CAN	64msg ×1 チャネル / 32msg ×2 チャネル	
ステッピングモータコントローラ	6 チャネル	
サウンドジェネレータ	5 チャネル	
ソフトウェアウォッチドッグ	あり	
ハードウェアウォッチドッグ	あり	
クロックスーパバイザ	初期値 ON	初期値 OFF
CRC 生成	あり	
低電圧検出リセット (外部低電圧検出)	あり	
内部低電圧検出リセット	あり	
パッケージ	LQFP-144, LQFP-208(with Suffix "M")	
その他	FLASH 品	
On Chip Debug	OCD 搭載	

*1: I^2C対応はch.0, ch.1のみ

1.4 機能概要

本シリーズの機能概要を示します。

表 1-2 機能概要

機能	特長
CPU	32 ビット RISC マイコン FR81S CPU コア メモリ保護機能(MPU) 8 チャネル搭載 浮動小数点演算(FPU)搭載
クロック	メイン発振 4MHz サブ発振 32kHz または なし PLL 逓倍率 最大 20 逓倍 ハードウェアウォッチドッグタイマのカウントクロックとして CR 発振器を搭載
入出力ポート	入力／出力または周辺信号として、ビット単位でプログラム可能 入力しきい値、駆動能力、pullup / pulldown を設定可能
外部バスインタフェース	22 ビットアドレス、8/16 ビットデータ出力
内部バスインタフェース	オンチップバス 32 ビット 、最大動作周波数 80MHz
周辺バスインタフェース	最大動作周波数 40MHz 32 ビット周辺バス、 もしくは 16 ビット周辺バス(R-bus) 両者とも同一周波数で動作します
フラッシュメモリ インタフェース	ワイルドレジスタ機能あり。小セクタ(64KB)対応
DMA コントローラ	同時に最大 16 チャネルの起動が可能 転送要因(内部周辺要求／ソフトウェア) 選択可能 バースト／ブロックの転送モードを選択可能 ■ 1 つの割込みベクタに複数の割込みがある場合、どの割込みから DMA 要求を発生させるかを選択可能 ■ 1 つの割込みベクタに複数の割込みがある場合、DMA 転送完了によりクリアする割込みを選択可能
ベースタイマ	16 ビットタイマ PWM / PPG / PWC /リロードタイマの 4 機能を選択して使用可能 リロードタイマ/PWC 機能に関して、2 チャネルカスケードモードで 32 ビットタイマとして使用可能
フリーランタイマ	32 ビットアップカウンタ

機能	特長
インプットキャプチャ	立上りエッジ、立下りエッジ、またはその両方を検出する 32 ビットキャプチャレジスタ 端子入力のエッジ検出で､32 ビットフリーランタイマのカウンタ値を取り込み、割込み要求を発生 LIN synch break/synch field の連携は以下のとおり インプットキャプチャ ch.0 → LIN-UART ch.2 インプットキャプチャ ch.1 → LIN-UART ch.3 インプットキャプチャ ch.2 → LIN-UART ch.4 インプットキャプチャ ch.3 → LIN-UART ch.5 インプットキャプチャ ch.4 → LIN-UART ch.6 インプットキャプチャ ch.5 → LIN-UART ch.7 インプットキャプチャ ch.6 → マルチファンクションシリアル ch.0 インプットキャプチャ ch.7 → マルチファンクションシリアル ch.1 インプットキャプチャ ch.8 → マルチファンクションシリアル ch.8 インプットキャプチャ ch.9 → マルチファンクションシリアル ch.9 フリーランタイマとの連携は以下のとおり インプットキャプチャ ch.0～ch.5 → フリーランタイマ ch.0 または ch.1 インプットキャプチャ ch.6～ch.11 → フリーランタイマ ch.2 または ch.3
アウトプットコンペア	32 ビットフリーランタイマとの照合時に割込み信号を出力 フリーランタイマとの連携は以下のとおり アウトプットコンペア ch.0～ch.3 → フリーランタイマ ch.0 または ch.1 アウトプットコンペア ch.4, ch.5 → フリーランタイマ ch.4 または ch.5 アウトプットコンペア ch.6～ch.9 → フリーランタイマ ch.2 または ch.3 アウトプットコンペア ch.10, ch.11 → フリーランタイマ ch.4 または ch.5
リロードタイマ	16 ビットリロードタイマ動作(トグル出力、ワンショット出力選択可能) イベントカウント機能選択可能 ch.4～ch.6 は外部端子なし
PPG	ワンショットの矩形波出力や､PWM 出力を行う周期とデューティをソフトウェアで変更可能 動作クロック周波数: PCLK×1, $1/2^2$, $1/2^4$, $1/2^6$ の 4 通りから選択可能
遅延割込み	タスク切替え用の割込みを発生。 ソフトウェアで CPU に対して割込み要求の発生/取消しが可能。
外部割込み	16 チャネル独立 割込み要因:立上りエッジ／立下りエッジ／"L"レベル／"H"レベル選択可能 スタンバイ復帰時のエッジ入力検出に対応
A/D コンバータ	10 ビットまたは 8 ビットの分解能の A/D コンバータ 1 チャネル内蔵 40 チャネルの入力ポートからアナログ値をサンプル可能 変換時間: 3µs 外部トリガ起動可能 内部タイマによる起動可能(16 ビットリロードタイマ使用)
D/A コンバータ	8 ビットの分解能×2 チャネル
LIN-UART(6 チャネル)	全二重方式 非同期/同期転送(スタート/ストップビット付き) 専用ボーレートジェネレータ内蔵 LIN プロトコル、スレーブノード対応、LIN synch break/synch field 検出可能 SPI(Serial Peripheral Interface)対応 LIN プロトコル Ver2.1 対応

機能	特長
マルチファンクション シリアル (4 チャネル)	UART / CSIO / LIN-UART / I²C-UART の 4 機能を選択して使用可能 送信 FIFO 16 段、受信 FIFO 16 段搭載 受信割込み要因(3 種類) ■ 受信エラー検出 (パリティ、オーバラン、フレームエラー) ■ FIFO に設定閾値分のデータが受信されたことを検出 ■ 受信データが FIFO 設定しきい値以下のとき、ボーレートクロック 8 クロック以上のアイドル期間を検出 送信割込み要因(2 種類) ■ 送信動作なし ■ 送信 FIFO エンプティ(送信中を含む) SPI(Serial Peripheral Interface)対応 LIN プロトコル Ver2.1 対応 (注意事項) 4 チャネル中、I²C 対応は ch.0, ch.1 のみです。
HS-SPI	Single/Dual/Quad-SPI プロトコルの Flash デバイス接続向け 3.3V 対応 マスタ時最大 16MHz スレーブ時は最大 8MHz (注意事項) 本シリーズでは HS_SPI は使用禁止です
LCD コントローラ/ ドライバ (1 チャネル)	＜デューティ駆動＞ LCD パネル(液晶表示)を直接駆動可能なセグメントドライバ、およびコモンドライバ 32SEG × 4COM 外部バスインタフェース使用時には、EPFR の設定により LCD の SEG0～SEG31,COM0～COM3,V0,V1,V2,V3 端子は使用可能 COM0～COM3,SEG0～SEG31,V0,V1,V2,V3 端子はソフトウェアで 1 端子ごとに汎用ポートへの切換えが可能(SEG23～SEG31 は、スタティック駆動への切換え可能) ＜スタティック駆動＞ LCD パネル(液晶表示)を直接駆動可能なセグメントドライバ、およびコモンドライバ 8SEG/1COM 外部バスインタフェース使用時には, EPFR の設定により LCD の ST0～ST8 端子は使用可能 ST0～ST8 端子はソフトウェアで 1 端子ごとに汎用ポートまたは、デューティ駆動のセグメント出力への切換えが可能
割込みコントローラ	割込み要求の検出 割込みレベルの設定
割込み要求一括読出し	複数の周辺の割込み有無を、一連のレジスタで読出し可能
CAN インタフェース	CAN 仕様バージョン 2.0 パート A およびパート B に準拠 64 個のメッセージバッファ× 1 チャネル と 32 個のメッセージバッファ× 2 チャネル 複数メッセージをサポート 受入れフィルタの柔軟な構成: 全ビットコンペア 全ビットマスク 2 個の部分ビットマスク 最高 1Mbps までサポート CAN 動作クロックのために CAN プリスケーラを実装
ステッピングモータコントローラ	大電流出力×4 本 PWM 周期 15.625kHz の設定が可能 (周辺クロック 16MHz 動作時) 端子共有 A/D コンバータでモータからの逆起電力を検出可能
アップダウン カウンタ (2 チャネル)	8/16bit アップダウンカウンタ×2 チャネル

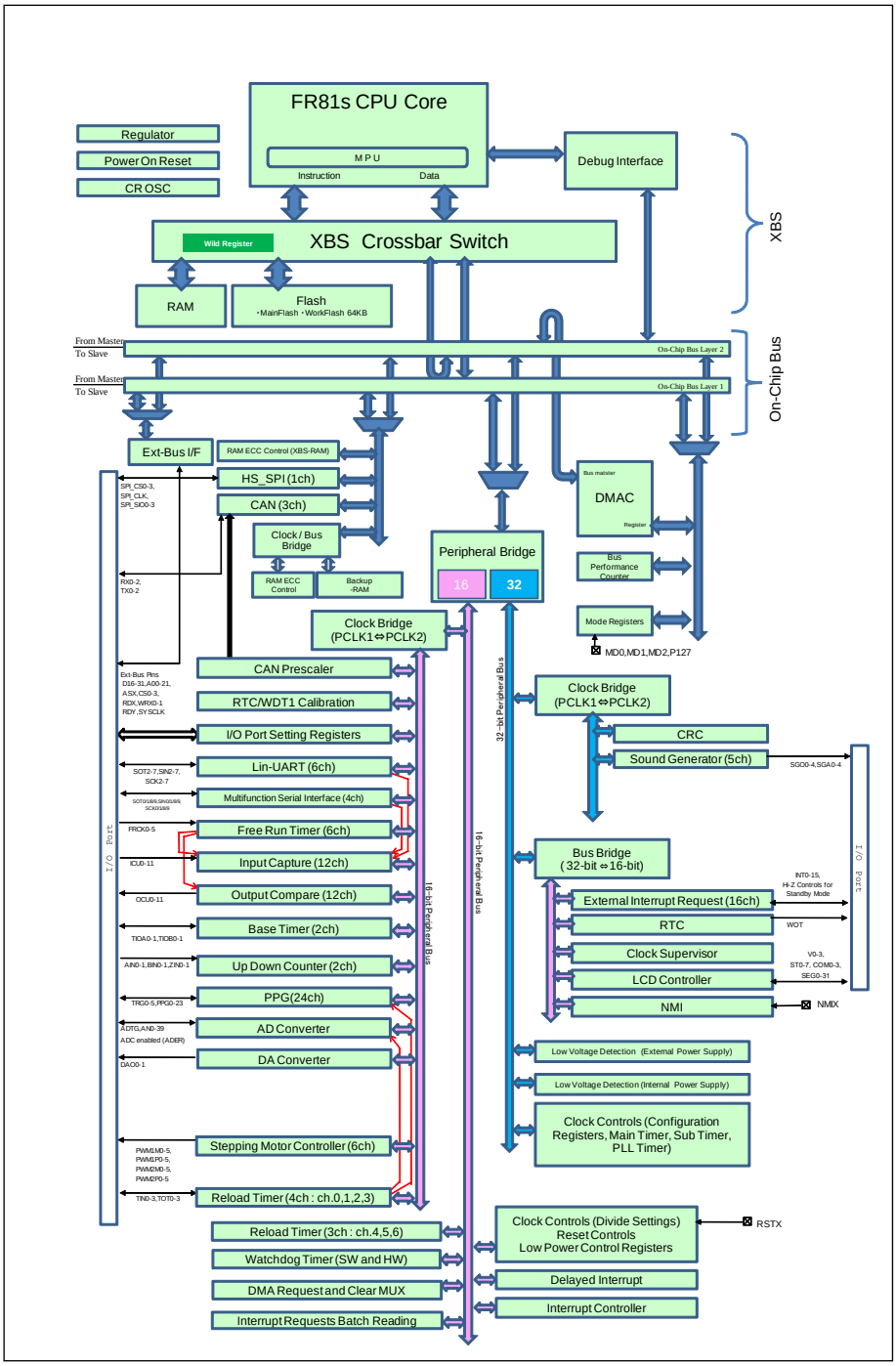
機能	特長
サウンドジェネレータ	周波数データ、振幅データの設定に加え、以下を設定可能 ■ デクリメント・インクリメントデータ、および実行周期 ■ トーン出力パルス数(出力間隔)
リアルタイムクロック	日時分秒レジスタ 動作クロックとしてメイン発振、サブ発振選択可能 サブクロック補正機能 ■ サブクロックの周期誤差をメインクロックにより監視する ■ 得られた誤差を秒カウンタ設定値に反映する 0.5 秒/秒/分/時/日で割込みを発生可能
キャリブレーション	メインクロックとの比較で、サブクロック駆動のリアルタイムクロックを補正 メインクロックとの比較で、CR 発振周波数を補正可能
ソフトウェアウォッチドッグ	CPU 動作中にカウント CPU 停止中はカウントを停止 周期は PCLK × (2^9～2^{24})サイクルの 16 通りから選択可能
ハードウェアウォッチドッグ	CR ベースの CPU の動作検出カウンタ プログラム暴走対策 周期: 260 ～ 416ms (通常 328ms, CR 発振の精度によってばらつきあり) 「RTC/WDT1 補正」回路によるキャリブレーション可能 周期に幅があるのは、製造のばらつきに起因するものです。任意に周期を設定できるわけではありませんので注意ください。
CRC 生成	入力レジスタへの逐次書込みにより、CRC コードを結果レジスタに表示
低電圧検出 (外部低電圧検出)	低電圧検出時リセット発生
内部低電圧検出	1.2V 系電圧を監視して、下回った場合はリセットを発生します。
低消費電力モード	スリープモード ストップモード 時計モード ストップモード(電源遮断) 時計モード(電源遮断) サブ RUN モード

機能	特長
I/O リロケーション	リロケーション対象周辺機能と分岐数 PPG 24 チャネル(ch.1 は 4 分岐, ch.0 および ch.2～ch.10 は 3 分岐, ch.11～ch.15 は 2 分岐, ch.16～ch.23 は分岐はなし) ■ インプットキャプチャ 12 チャネル × 3 分岐 ■ LIN-UART 6 チャネル × 2 分岐 ■ リロードタイマ 4 チャネル × 3 分岐 ■ フリーランタイマ 2 チャネル × 分岐なし、4 チャネル × 2 分岐 ■ アウトプットコンペア 12 チャネル × 2 分岐 ■ マルチファンクションシリアル 4 チャネル (ch.0,ch.1,ch.9 は 2 分岐、ch.8 は 3 分岐)。ただし、I²C リロケーション不可。 ■ 外部割込み 16 チャネル × 2 分岐 ■ アップダウンカウンタ 2 チャネル (ch.0 は 3 分岐、ch.1 は 2 分岐) ■ サウンドジェネレータ 5 チャネル(ch.0, ch,1, ch.2, ch.3 は分岐なし、ch.4 は 2 分岐) (注意事項) LIN-UART およびマルチファンクションシリアルの SCK/ SOT/ SIN は必ず同一グループ(SCKn/ SOTn/ SINn または SCKn_1/SOTn_1/ SINn_1)になるように設定してください。下記例のようにバラバラにリロケーションすることは禁止です。 禁止例： SCKn_0/ SOTn_1/ SINn_0 設定の詳細は I/O ポート章の拡張ポート機能レジスタの説明を参照してください。
NMI 要求	NMIX 端子から入力されるマスク不可能な割込み信号
デバッグインタフェース	OCD 搭載

1.5 ブロックダイヤグラム

本シリーズのブロックダイヤグラムを示します。

図 1-1 ブロックダイヤグラム



(注意事項)

本シリーズではHS_SPIは使用禁止です。

1.6 CPU

CPUの汎用レジスタおよび専用レジスタについて説明します。

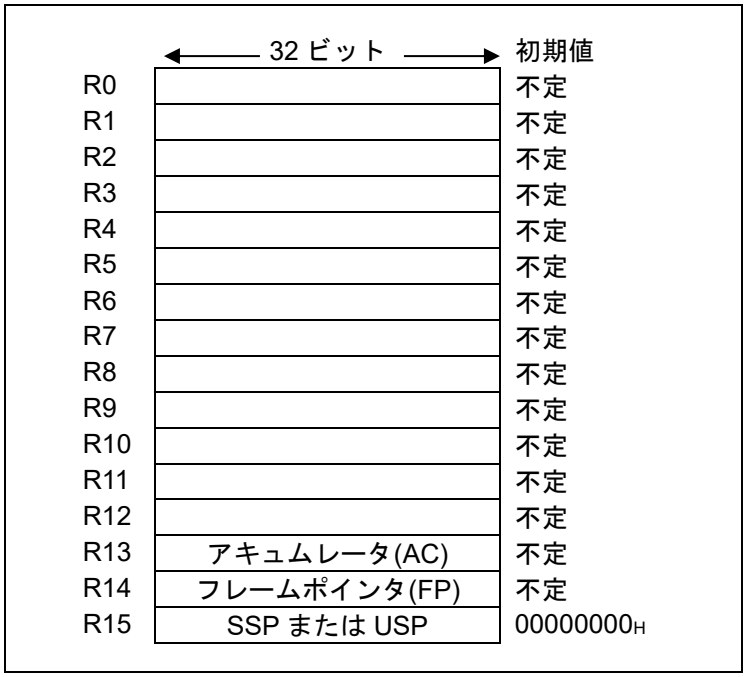
1.6.1汎用レジスタ

1.6.2専用レジスタ

1.6.1 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します。

図 1-2 汎用レジスタ



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おり、そのために一部の命令が強化されています。

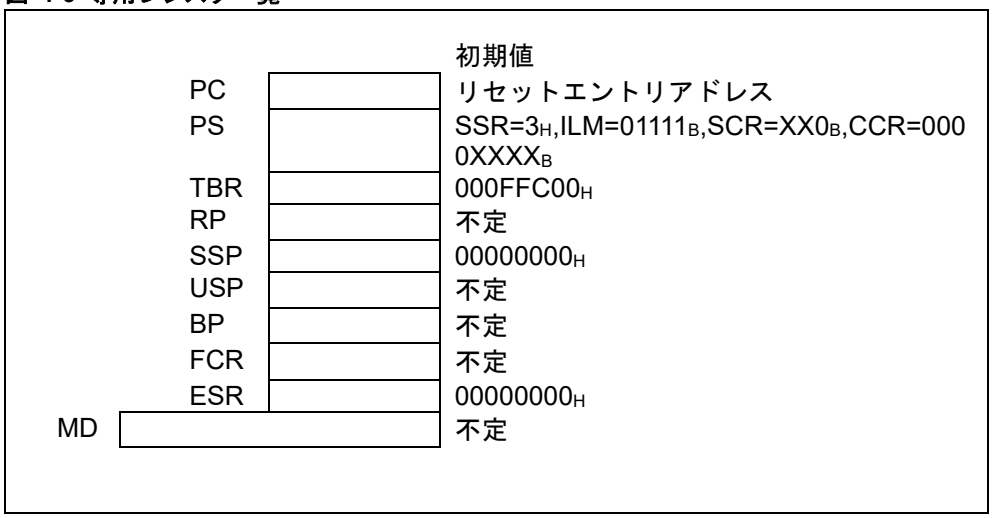
- R13: AC(アキュムレータ)
- R14: FP (フレームポインタ)
- R15: SP (スタックポインタ)

リセットによる初期値は、R0～R14は不定です。R15は、00000000_H (SSPの値) となります。

1.6.2 専用レジスタ

各種用途専用の32ビット長専用レジスタが10本と、乗除算用の64ビット長専用レジスタが1本あります。

図 1-3 専用レジスター覧



専用レジスタは、特定の目的のために使用します。

FRファミリでは、以下の専用レジスタが用意されています。

- 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)
- ベースポインタ (BP)
- FPU 制御レジスタ (FCR)
- 例外ステータスレジスタ (ESR)
- 乗除算レジスタ (MD)

1.7 端子配列図

本シリーズ端子配列図を示します。

図 1-4 端子配列図

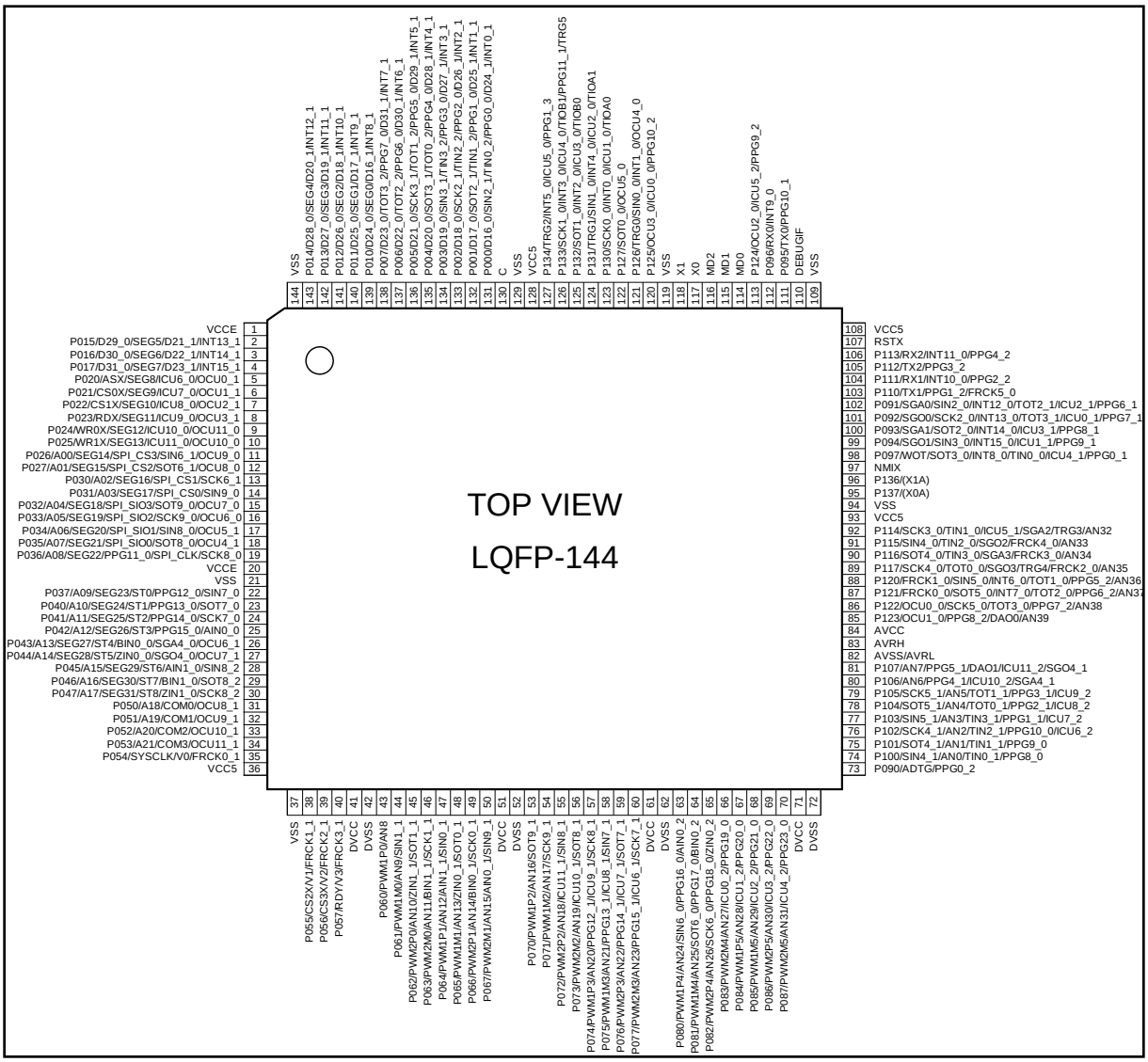
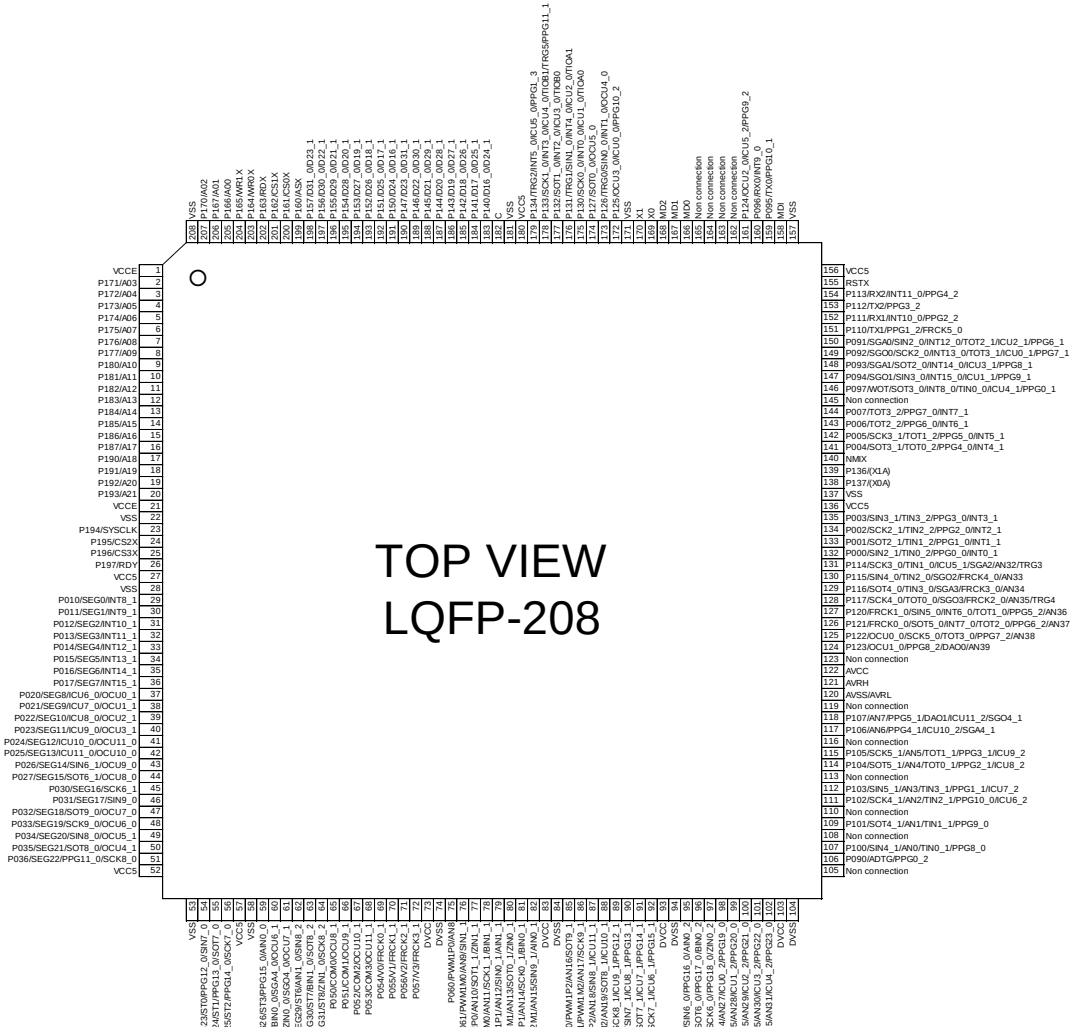


図 1-5 端子配列図

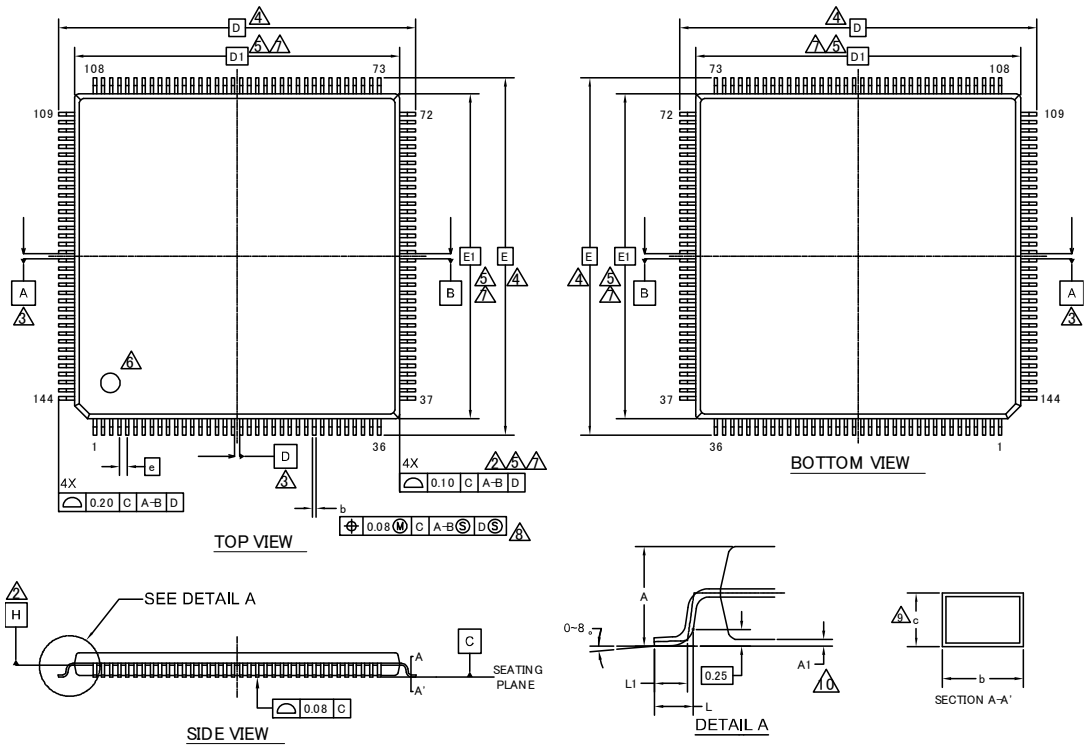


1.8 パッケージ外形図

本シリーズのパッケージ外形図を示します。

図 1-6 外形寸法図 LQFP-144 (LQS 144)

Package Type	Package Code
LQFP 144	LQS 144



SYMBOL	DIMENSIONS		
	MIN.	NOM.	MAX.
A	—	—	1.70
A1	0.05	—	0.15
b	0.17	0.22	0.27
c	0.09	—	0.20
D	22.00 BSC		
D1	20.00 BSC		
e	0.50 BSC		
E	22.00 BSC		
E1	20.00 BSC		
L	0.45	0.60	0.75
L1	0.30	0.50	0.70

NOTES

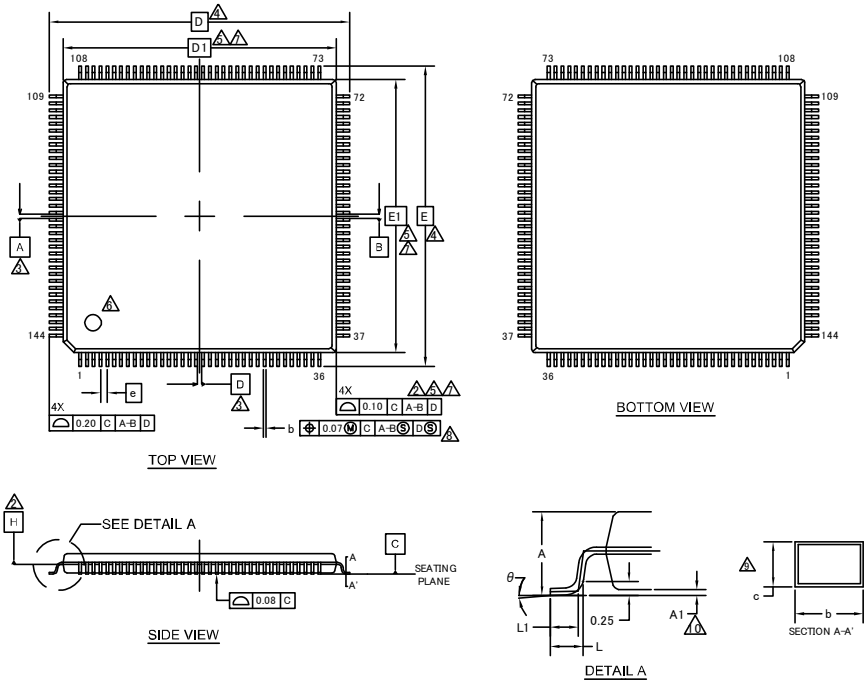
1. ALL DIMENSIONS ARE IN MILLIMETERS
2. DATUM PLANE H IS LOCATED AT THE BOTTOM OF THE MOLD PARTING LINE COINCIDENT WITH WHERE THE LEAD EXITS THE BODY.
3. DATUMS A-B AND D TO BE DETERMINED AT DATUM PLANE H.
4. TO BE DETERMINED AT SEATING PLANE C.
5. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25mm PRE SIDE. DIMENSIONS D1 AND E1 INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE H.
6. DETAILS OF PIN 1 IDENTIFIER ARE OPTIONAL BUT MUST BE LOCATED WITHIN THE ZONE INDICATED.
7. REGARDLESS OF THE RELATIVE SIZE OF THE UPPER AND LOWER BODY SECTIONS. DIMENSIONS D1 AND E1 ARE DETERMINED AT THE LARGEST FEATURE OF THE BODY EXCLUSIVE OF MOLD FLASH AND GATE BURRS. BUT INCLUDING ANY MISMATCH BETWEEN THE UPPER AND LOWER SECTIONS OF THE MOLDER BODY.
8. DIMENSION b DOES NOT INCLUDE DAMBAR PROTRUSION. THE DAMBAR PROTRUSION (S) SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED b MAXIMUM BY MORE THAN 0.08mm. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE LEAD FOOT.
9. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10mm AND 0.25mm FROM THE LEAD TIP.
10. A1 IS DEFINED AS THE DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT OF THE PACKAGE BODY.

PACKAGE OUTLINE, 144 LEAD LQFP
20.0X20.0X1.7 MM LQS144 REV*A

002-13015 *A

図 1-7 外形寸法図 LQFP-144 (LQN 144)

Package Type	Package Code
LQFP 144	LQN 144



SYMBOL	DIMENSIONS		
	MIN.	NOM.	MAX.
A	——	——	1.70
A1	0.05	——	0.15
b	0.145	0.18	0.215
c	0.115	——	0.195
D	18.00 BSC		
D1	16.00 BSC		
e	0.40 BSC		
E	18.00 BSC		
E1	16.00 BSC		
L	0.45	0.60	0.75
L1	0.30	0.50	0.70
θ	0°	——	8°

NOTES

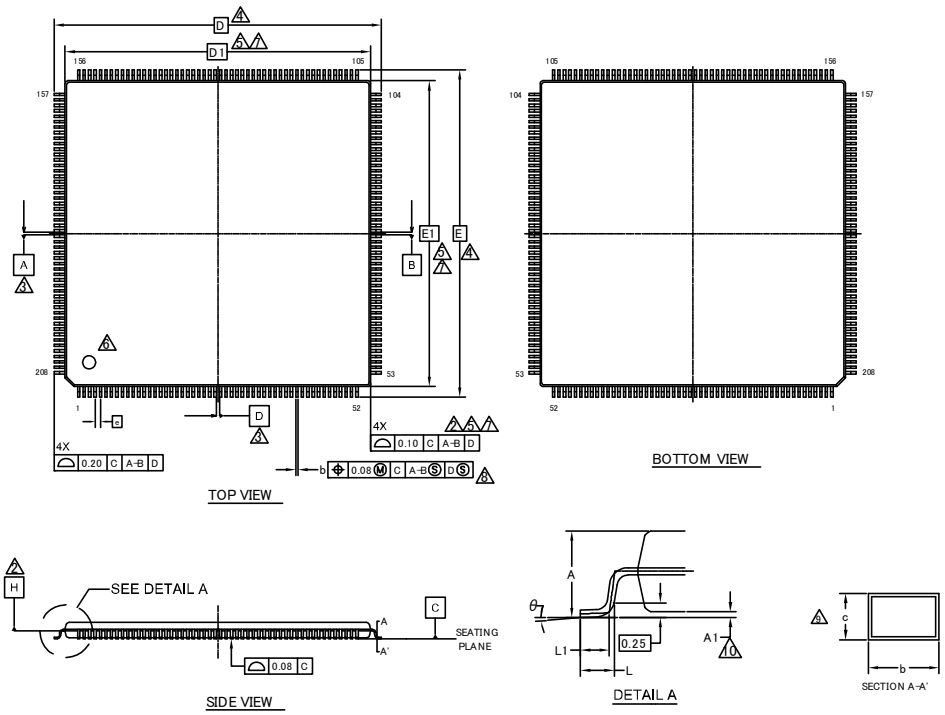
1. CONTROLLING DIMENSIONS ARE IN MILLIMETERS (mm)
2. DATUM PLANE H IS LOCATED AT THE BOTTOM OF THE MOLD PARTING LINE COINCIDENT WITH WHERE THE LEAD EXITS THE BODY.
3. DATUMS A-B AND D TO BE DETERMINED AT DATUM PLANE H.
4. TO BE DETERMINED AT SEATING PLANE C.
5. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25mm PRE SIDE. DIMENSIONS D1 AND E1 INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE H.
6. DETAILS OF PIN 1 IDENTIFIER ARE OPTIONAL BUT MUST BE LOCATED WITHIN THE ZONE INDICATED.
7. REGARDLESS OF THE RELATIVE SIZE OF THE UPPER AND LOWER BODY SECTIONS. DIMENSIONS D1 AND E1 ARE DETERMINED AT THE LARGEST FEATURE OF THE BODY EXCLUSIVE OF MOLD FLASH AND GATE BURRS. BUT INCLUDING ANY MISMATCH BETWEEN THE UPPER AND LOWER SECTIONS OF THE MOLDER BODY.
8. DIMENSION b DOES NOT INCLUDE DAMBER PROTRUSION. THE DAMBAR PROTRUSION (S) SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED b MAXIMUM BY MORE THAN 0.08mm. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE LEAD FOOT.
9. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10mm AND 0.25mm FROM THE LEAD TIP.
10. A1 IS DEFINED AS THE DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT OF THE PACKAGE BODY.

002-14045 **

PACKAGE OUTLINE, 144 LEAD LQFP
16.0X16.0X1.7 MM LQN144 REV**

図 1-8 外形寸法図 LQFP-208 (LQR 208)

Package Type	Package Code
LQFP 208	LQR 208



SYMBOL	DIMENSIONS		
	MIN.	NOM.	MAX.
A	—	—	1.70
A1	0.05	—	0.15
b	0.17	0.22	0.27
c	0.09	—	0.20
D	30.00 BSC		
D1	28.00 BSC		
e	0.50 BSC		
E	30.00 BSC		
E1	28.00 BSC		
L	0.45	0.60	0.75
L1	0.30	0.50	0.70
θ	0°	—	8°

NOTES

1. ALL DIMENSIONS ARE IN MILLIMETERS.
2. DATUM PLANE H IS LOCATED AT THE BOTTOM OF THE MOLD PARTING LINE COINCIDENT WITH WHERE THE LEAD EXITS THE BODY.
3. DATUMS A-B AND D TO BE DETERMINED AT DATUM PLANE H.
4. TO BE DETERMINED AT SEATING PLANE C.
5. DIMENSIONS D1 AND E1 DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25mm PRE SIDE. DIMENSIONS D1 AND E1 INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM PLANE H.
6. DETAILS OF PIN 1 IDENTIFIER ARE OPTIONAL BUT MUST BE LOCATED WITHIN THE ZONE INDICATED.
7. REGARDLESS OF THE RELATIVE SIZE OF THE UPPER AND LOWER BODY SECTIONS. DIMENSIONS D1 AND E1 ARE DETERMINED AT THE LARGEST FEATURE OF THE BODY EXCLUSIVE OF MOLD FLASH AND GATE BURRS. BUT INCLUDING ANY MISMATCH BETWEEN THE UPPER AND LOWER SECTIONS OF THE MOLDER BODY.
8. DIMENSION b DOES NOT INCLUDE DAMBER PROTRUSION. THE DAMBAR PROTRUSION (S) SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED b MAXIMUM BY MORE THAN 0.08mm. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE LEAD FOOT.
9. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10mm AND 0.25mm FROM THE LEAD TIP.
10. A1 IS DEFINED AS THE DISTANCE FROM THE SEATING PLANE TO THE LOWEST POINT OF THE PACKAGE BODY.

PACKAGE OUTLINE, 208 LEAD LQFP
28.0X28.0X1.7 MM LQR208 REV**

002-15151 **

1.9 端子機能説明

本シリーズの端子機能一覧を示します。

表 1-3 端子機能一覧表 (CY91F575/577)

端子番号	端子名	入出力回路形式	機　能
2	P015	H	汎用入出力ポート
	D29_0		外部バスデータ入出力
	SEG5		LCDC セグメント (Duty) 出力
	D21_1		外部バスデータ入出力
	INT13_1		外部割込み要求入力 ch.13 リロケーション 1
3	P016	H	汎用入出力ポート
	D30_0		外部バスデータ入出力
	SEG6		LCDC セグメント(Duty)出力
	D22_1		外部バスデータ入出力
	INT14_1		外部割込み要求入力 ch.14 リロケーション 1
4	P017	H	汎用入出力ポート
	D31_0		外部バスデータ入出力
	SEG7		LCDC セグメント (Duty) 出力
	D23_1		外部バスデータ入出力
	INT15_1		外部割込み要求入力 ch.15 リロケーション 1
5	P020	H	汎用入出力ポート
	ASX		外部バスアドレスストローブ出力
	SEG8		LCDC セグメント (Duty) 出力
	ICU6_0		インプットキャプチャ入力 ch.6 リロケーション 0
	OCU0_1		アウトプットコンペア出力 ch.0 リロケーション 1
6	P021	H	汎用入出力ポート
	CS0X		外部バスチップセレクト 0 出力
	SEG9		LCDC セグメント(Duty)出力
	ICU7_0		インプットキャプチャ入力 ch.7 リロケーション 0
	OCU1_1		アウトプットコンペア出力 ch.1 リロケーション 1
7	P022	H	汎用入出力ポート
	CS1X		外部バスチップセレクト 1 出力
	SEG10		LCDC セグメント(Duty)出力
	ICU8_0		インプットキャプチャ入力 ch.8 リロケーション 0
	OCU2_1		アウトプットコンペア出力 ch.2 リロケーション 1
8	P023	H	汎用入出力ポート
	RDX		外部バスリードストローブ出力
	SEG11		LCDC セグメント(Duty)出力
	ICU9_0		インプットキャプチャ入力 ch.9 リロケーション 0
	OCU3_1		アウトプットコンペア出力 ch.3 リロケーション 1
9	P024	H	汎用入出力ポート
	WR0X		外部バスライトストローブ 0 出力
	SEG12		LCDC セグメント(Duty)出力
	ICU10_0		インプットキャプチャ入力 ch.10 リロケーション 0
	OCU11_0		アウトプットコンペア出力 ch.11 リロケーション 0

端子番号	端子名	入出力回路形式	機　能
10	P025	H	汎用入出力ポート
	WR1X		外部バスライトストローブ 1 出力
	SEG13		LCDC セグメント(Duty)出力
	ICU11_0		インプットキャプチャ入力 ch.11 リロケーション 0
	OCU10_0		アウトプットコンペア出力 ch.10 リロケーション 0
11	P026	H	汎用入出力ポート
	A00		外部バスアドレス出力
	SEG14		LCDC セグメント(Duty)出力
	SPI_CS3		HS_SPI SSEL3 出力
	SIN6_1		LIN_UART シリアル入力 ch.6 リロケーション 1
	OCU9_0		アウトプットコンペア出力 ch.9 リロケーション 0
12	P027	H	汎用入出力ポート
	A01		外部バスアドレス出力
	SEG15		LCDC セグメント(Duty)出力
	SPI_CS2		HS_SPI SSEL2 出力
	SOT6_1		LIN_UART シリアル出力 ch.6 リロケーション 1
	OCU8_0		アウトプットコンペア出力 ch.8 リロケーション 0
13	P030	H	汎用入出力ポート
	A02		外部バスアドレス出力
	SEG16		LCDC セグメント(Duty)出力
	SPI_CS1		HS_SPI SSEL1 出力
	SCK6_1		LIN_UART シリアルクロック入出力 ch.6 リロケーション 1
14	P031	H	汎用入出力ポート
	A03		外部バスアドレス出力
	SEG17		LCDC セグメント(Duty)出力
	SPI_CS0		HS_SPI SSEL0 入出力
	SIN9_0		マルチファンクションシリアル入力 ch.9 リロケーション 0
15	P032	H	汎用入出力ポート
	A04		外部バスアドレス出力
	SEG18		LCDC セグメント(Duty)出力
	SPI_SIO3		HS_SPI SDATA3 入出力
	SOT9_0		マルチファンクションシリアル出力 ch.9 リロケーション 0
	OCU7_0		アウトプットコンペア出力 ch.7 リロケーション 0
16	P033	H	汎用入出力ポート
	A05		外部バスアドレス出力
	SEG19		LCDC セグメント(Duty)出力
	SPI_SIO2		HS_SPI SDATA2 入出力
	SCK9_0		マルチファンクションシリアルクロック入出力 ch.9 リロケーション 0
	OCU6_0		アウトプットコンペア出力 ch.6 リロケーション 0
17	P034	H	汎用入出力ポート
	A06		外部バスアドレス出力
	SEG20		LCDC セグメント(Duty)出力
	SPI_SIO1		HS_SPI SDATA1 入出力
	SIN8_0		マルチファンクションシリアル入力 ch.8 リロケーション 0
	OCU5_1		アウトプットコンペア出力 ch.5 リロケーション 1

端子番号	端子名	入出力回路形式	機　能
18	P035	H	汎用入出力ポート
	A07		外部バスアドレス出力
	SEG21		LCDC セグメント(Duty)出力
	SPI_SIO0		HS_SPI SDATA0 入出力
	SOT8_0		マルチファンクションシリアル出力 ch.8 リロケーション 0
	OCU4_1		アウトプットコンペア出力 ch.4 リロケーション 1
19	P036	H	汎用入出力ポート
	A08		外部バスアドレス出力
	SEG22		LCDC セグメント(Duty)出力
	PPG11_0		PPG 出力 ch.11 リロケーション 0
	SPI_CLK		HS_SPI SCLK 入出力
	SCK8_0		マルチファンクションシリアルクロック入出力 ch.8 リロケーション 0
22	P037	I	汎用入出力ポート
	A09		外部バスアドレス出力
	SEG23		LCDC セグメント(Duty)出力
	ST0		LCDC セグメント(Static)出力
	PPG12_0		PPG 出力 ch.12 リロケーション 0
	SIN7_0		LIN_UART シリアル入力 ch.7 リロケーション 0
23	P040	I	汎用入出力ポート
	A10		外部バスアドレス出力
	SEG24		LCDC セグメント(Duty)出力
	ST1		LCDC セグメント(Static)出力
	PPG13_0		PPG 出力 ch.13 リロケーション 0
	SOT7_0		LIN_UART シリアル出力 ch.7 リロケーション 0
24	P041	I	汎用入出力ポート
	A11		外部バスアドレス出力
	SEG25		LCDC セグメント(Duty) 出力
	ST2		LCDC セグメント(Static)出力
	PPG14_0		PPG 出力 ch.14 リロケーション 0
	SCK7_0		LIN_UART シリアルクロック入出力 ch.7 リロケーション 0
25	P042	I	汎用入出力ポート
	A12		外部バスアドレス出力
	SEG26		LCDC セグメント(Duty)出力
	ST3		LCDC セグメント(Static)出力
	PPG15_0		PPG 出力 ch.15 リロケーション 0
	AIN0_0		アップダウンカウンタ AIN 入力 ch.0 リロケーション 0
26	P043	I	汎用入出力ポート
	A13		外部バスアドレス出力
	SEG27		LCDC セグメント(Duty)出力
	ST4		LCDC セグメント(Static)出力
	BIN0_0		アップダウンカウンタ BIN 入力 ch.0 リロケーション 0
	SGA4_0		サウンドジェネレータ SGA 出力 ch.4 リロケーション 0
	OCU6_1		アウトプットコンペア出力 ch.6 リロケーション 1

端子番号	端子名	入出力回路形式	機　能
27	P044	I	汎用入出力ポート
	A14		外部バスアドレス出力
	SEG28		LCDC セグメント(Duty)出力
	ST5		LCDC セグメント(Static)出力
	ZIN0_0		アップダウンカウンタ ZIN 入力 ch.0 リロケーション 0
	SGO4_0		サウンドジェネレータ SGO 出力 ch.4 リロケーション 0
	OCU7_1		アウトプットコンペア出力 ch.7 リロケーション 1
28	P045	I	汎用入出力ポート
	A15		外部バスアドレス出力
	SEG29		LCDC セグメント(Duty)出力
	ST6		LCDC セグメント(Static)出力
	AIN1_0		アップダウンカウンタ AIN 入力 ch.1 リロケーション 0
	SIN8_2		マルチファンクションシリアル入力 ch.8 リロケーション 2
29	P046	I	汎用入出力ポート
	A16		外部バスアドレス出力
	SEG30		LCDC セグメント(Duty)出力
	ST7		LCDC セグメント(Static)出力
	BIN1_0		アップダウンカウンタ BIN 入力 ch.1 リロケーション 0
	SOT8_2		マルチファンクションシリアル出力 ch.8 リロケーション 2
30	P047	I	汎用入出力ポート
	A17		外部バスアドレス出力
	SEG31		LCDC セグメント(Duty)出力
	ST8		LCDC セグメント(Static)出力
	ZIN1_0		アップダウンカウンタ ZIN 入力 ch.1 リロケーション 0
	SCK8_2		マルチファンクションシリアルクロック入出力 ch.8 リロケーション 2
31	P050	I	汎用入出力ポート
	A18		外部バスアドレス出力
	COM0		LCDC セグメント(Duty)コモン出力
	OCU8_1		アウトプットコンペア出力 ch.8 リロケーション 1
32	P051	I	汎用入出力ポート
	A19		外部バスアドレス出力
	COM1		LCDC セグメント(Duty)コモン出力
	OCU9_1		アウトプットコンペア出力 ch.9 リロケーション 1
33	P052	I	汎用入出力ポート
	A20		外部バスアドレス出力
	COM2		LCDC セグメント(Duty)コモン出力
	OCU10_1		アウトプットコンペア出力 ch.10 リロケーション 1
34	P053	I	汎用入出力ポート
	A21		外部バスアドレス出力
	COM3		LCDC セグメント(Duty)コモン出力
	OCU11_1		アウトプットコンペア出力 ch.11 リロケーション 1
35	P054	I2	汎用入出力ポート
	SYSCLK		外部バスクロック出力
	V0		LCDC 基準電圧 V0 入力
	FRCK0_1		フリーランタイマクロック入力 ch.0 リロケーション 1

端子番号	端子名	入出力回路形式	機　能
38	P055	I2	汎用入出力ポート
	CS2X		外部バスチップセレクト 2 出力
	V1		LCDC 基準電圧 V1 入力
	FRCK1_1		フリーランタイマクロック入力 ch.1 リロケーション 1
39	P056	I2	汎用入出力ポート
	CS3X		外部バスチップセレクト 3 出力
	V2		LCDC 基準電圧 V2 入力
	FRCK2_1		フリーランタイマクロック入力 ch.2 リロケーション 1
40	P057	I3	汎用入出力ポート (入力のみ 出力なし)
	RDY		外部バス RDY 入力
	V3		LCDC 基準電圧 V3 入力
	FRCK3_1		フリーランタイマクロック入力 ch.3 リロケーション 1
43	P060	K	汎用入出力ポート
	PWM1P0		SMC 出力 ch.0
	AN8		ADC アナログ入力 ch.8
44	P061	K	汎用入出力ポート
	PWM1M0		SMC 出力 ch.0
	AN9		ADC アナログ入力 ch.9
	SIN1_1		マルチファンクションシリアル入力 ch.1 リロケーション 1
45	P062	K	汎用入出力ポート
	PWM2P0		SMC 出力 ch.0
	AN10		ADC アナログ入力 ch.10
	ZIN1_1		アップダウンカウンタ ZIN 入力 ch.1 リロケーション 1
	SOT1_1		マルチファンクションシリアル出力 ch.1 リロケーション 1
46	P063	K	汎用入出力ポート
	PWM2M0		SMC 出力 ch.0
	AN11		ADC アナログ入力 ch.11
	BIN1_1		アップダウンカウンタ BIN 入力 ch.1 リロケーション 1
	SCK1_1		マルチファンクションシリアルクロック入出力 ch.1 リロケーション 1
47	P064	K	汎用入出力ポート
	PWM1P1		SMC 出力 ch.1
	AN12		ADC アナログ入力 ch.12
	AIN1_1		アップダウンカウンタ AIN 入力 ch.1 リロケーション 1
	SIN0_1		マルチファンクションシリアル入力 ch.0 リロケーション 1
48	P065	K	汎用入出力ポート
	PWM1M1		SMC 出力 ch.1
	AN13		ADC アナログ入力 ch.13
	ZIN0_1		アップダウンカウンタ ZIN 入力 ch.0 リロケーション 1
	SOT0_1		マルチファンクションシリアル出力 ch.0 リロケーション 1
49	P066	K	汎用入出力ポート
	PWM2P1		SMC 出力 ch.1
	AN14		ADC アナログ入力 ch.14
	BIN0_1		アップダウンカウンタ BIN 入力 ch.0 リロケーション 1
	SCK0_1		マルチファンクションシリアルクロック入出力 ch.0 リロケーション 1

端子番号	端子名	入出力回路形式	機　能
50	P067	K	汎用入出力ポート
	PWM2M1		SMC 出力 ch.1
	AN15		ADC アナログ入力 ch.15
	AIN0_1		アップダウンカウンタ AIN 入力 ch.0 リロケーション 1
	SIN9_1		マルチファンクションシリアル入力 ch.9 リロケーション 1
53	P070	K	汎用入出力ポート
	PWM1P2		SMC 出力 ch.2
	AN16		ADC アナログ入力 ch.16
	SOT9_1		マルチファンクションシリアル出力 ch.9 リロケーション 1
54	P071	K	汎用入出力ポート
	PWM1M2		SMC 出力 ch.2
	AN17		ADC アナログ入力 ch.17
	SCK9_1		マルチファンクションシリアルクロック入出力 ch.9 リロケーション 1
55	P072	K	汎用入出力ポート
	PWM2P2		SMC 出力 ch.2
	AN18		ADC アナログ入力 ch.18
	ICU11_1		インプットキャプチャ入力 ch.11 リロケーション 1
	SIN8_1		マルチファンクションシリアル入力 ch.8 リロケーション 1
56	P073	K	汎用入出力ポート
	PWM2M2		SMC 出力 ch.2
	AN19		ADC アナログ入力 ch.19
	ICU10_1		インプットキャプチャ入力 ch.10 リロケーション 1
	SOT8_1		マルチファンクションシリアル出力 ch.8 リロケーション 1
57	P074	K	汎用入出力ポート
	PWM1P3		SMC 出力 ch.3
	AN20		ADC アナログ入力 ch.20
	PPG12_1		PPG 出力 ch.12 リロケーション 1
	ICU9_1		インプットキャプチャ入力 ch.9 リロケーション 1
	SCK8_1		マルチファンクションシリアルクロック入出力 ch.8 リロケーション 1
58	P075	K	汎用入出力ポート
	PWM1M3		SMC 出力 ch.3
	AN21		ADC アナログ入力 ch.21
	PPG13_1		PPG 出力 ch.13 リロケーション 1
	ICU8_1		インプットキャプチャ入力 ch.8 リロケーション 1
	SIN7_1		LIN_UART シリアル入力 ch.7 リロケーション 1
59	P076	K	汎用入出力ポート
	PWM2P3		SMC 出力 ch.3
	AN22		ADC アナログ入力 ch.22
	PPG14_1		PPG 出力 ch.14 リロケーション 1
	ICU7_1		インプットキャプチャ入力 ch.7 リロケーション 1
	SOT7_1		LIN_UART シリアル出力 ch.7 リロケーション 1

端子番号	端子名	入出力回路形式	機　能
60	P077	K	汎用入出力ポート
	PWM2M3		SMC 出力 ch.3
	AN23		ADC アナログ入力 ch.23
	PPG15_1		PPG 出力 ch.15 リロケーション 1
	ICU6_1		インプットキャプチャ入力 ch.6 リロケーション 1
	SCK7_1		LIN_UART シリアルクロック入出力 ch.7 リロケーション 1
63	P080	K	汎用入出力ポート
	PWM1P4		SMC 出力 ch.4
	AN24		ADC アナログ入力 ch.24
	SIN6_0		LIN_UART シリアル入力 ch.6 リロケーション 0
	PPG16_0		PPG 出力 ch.16 リロケーション 0
	AIN0_2		アップダウンカウンタ AIN 入力 ch.0 リロケーション 2
64	P081	K	汎用入出力ポート
	PWM1M4		SMC 出力 ch.4
	AN25		ADC アナログ入力 ch.25
	SOT6_0		LIN_UART シリアル出力 ch.6 リロケーション 0
	PPG17_0		PPG 出力 ch.17 リロケーション 0
	BIN0_2		アップダウンカウンタ BIN 入力 ch.0 リロケーション 2
65	P082	K	汎用入出力ポート
	PWM2P4		SMC 出力 ch.4
	AN26		ADC アナログ入力 ch.26
	SCK6_0		LIN_UART シリアルクロック入出力 ch.6 リロケーション 0
	PPG18_0		PPG 出力 ch.18 リロケーション 0
	ZIN0_2		アップダウンカウンタ ZIN 入力 ch.0 リロケーション 2
66	P083	K	汎用入出力ポート
	PWM2M4		SMC 出力 ch.4
	AN27		ADC アナログ入力 ch.27
	ICU0_2		インプットキャプチャ入力 ch.0 リロケーション 2
	PPG19_0		PPG 出力 ch.19 リロケーション 0
67	P084	K	汎用入出力ポート
	PWM1P5		SMC 出力 ch.5
	AN28		ADC アナログ入力 ch.28
	ICU1_2		インプットキャプチャ入力 ch.1 リロケーション 2
	PPG20_0		PPG 出力 ch.20 リロケーション 0
68	P085	K	汎用入出力ポート
	PWM1M5		SMC 出力 ch.5
	AN29		ADC アナログ入力 ch.29
	ICU2_2		インプットキャプチャ入力 ch.2 リロケーション 2
	PPG21_0		PPG 出力 ch.21 リロケーション 0
69	P086	K	汎用入出力ポート
	PWM2P5		SMC 出力 ch.5
	AN30		ADC アナログ入力 ch.30
	ICU3_2		インプットキャプチャ入力 ch.3 リロケーション 2
	PPG22_0		PPG 出力 ch.22 リロケーション 0

端子番号	端子名	入出力回路形式	機　能
70	P087	K	汎用入出力ポート
	PWM2M5		SMC 出力 ch.5
	AN31		ADC アナログ入力 ch.31
	ICU4_2		インプットキャプチャ入力 ch.4 リロケーション 2
	PPG23_0		PPG 出力 ch.23 リロケーション 0
73	P090	M	汎用入出力ポート
	ADTG		ADC 外部トリガ入力
	PPG0_2		PPG 出力 ch.0 リロケーション 2
74	P100	J	汎用入出力ポート
	SIN4_1		LIN_UART シリアル入力 ch.4 リロケーション 1
	AN0		ADC アナログ入力 ch.0
	TIN0_1		リロードタイマイベント入力 ch.0 リロケーション 1
	PPG8_0		PPG 出力 ch.8 リロケーション 0
75	P101	J	汎用入出力ポート
	SOT4_1		LIN_UART シリアル出力 ch.4 リロケーション 1
	AN1		ADC アナログ入力 ch.1
	TIN1_1		リロードタイマイベント入力 ch.1 リロケーション 1
	PPG9_0		PPG 出力 ch.9 リロケーション 0
76	P102	J	汎用入出力ポート
	SCK4_1		LIN_UART シリアルクロック入出力 ch.4 リロケーション 1
	AN2		ADC アナログ入力 ch.2
	TIN2_1		リロードタイマイベント入力 ch.2 リロケーション 1
	PPG10_0		PPG 出力 ch.10 リロケーション 0
	ICU6_2		インプットキャプチャ入力 ch.6 リロケーション 2
77	P103	J	汎用入出力ポート
	SIN5_1		LIN_UART シリアル入力 ch.5 リロケーション 1
	AN3		ADC アナログ入力 ch.3
	TIN3_1		リロードタイマイベント入力 ch.3 リロケーション 1
	PPG1_1		PPG 出力 ch.1 リロケーション 1
	ICU7_2		インプットキャプチャ入力 ch.7 リロケーション 2
78	P104	J	汎用入出力ポート
	SOT5_1		LIN_UART シリアル出力 ch.5 リロケーション 1
	AN4		ADC アナログ入力 ch.4
	TOT0_1		リロードタイマ出力 ch.0 リロケーション 1
	PPG2_1		PPG 出力 ch.2 リロケーション 1
	ICU8_2		インプットキャプチャ入力 ch.8 リロケーション 2
79	P105	J	汎用入出力ポート
	SCK5_1		LIN_UART シリアルクロック入出力 ch.5 リロケーション 1
	AN5		ADC アナログ入力 ch.5
	TOT1_1		リロードタイマ出力 ch.1 リロケーション 1
	PPG3_1		PPG 出力 ch.3 リロケーション 1
	ICU9_2		インプットキャプチャ入力 ch.9 リロケーション 2

端子番号	端子名	入出力回路形式	機　能
80	P106	J	汎用入出力ポート
	AN6		ADC アナログ入力 ch.6
	PPG4_1		PPG 出力 ch.4 リロケーション 1
	ICU10_2		インプットキャプチャ入力 ch.10 リロケーション 2
	SGA4_1		サウンドジェネレータ SGA 出力 ch.4 リロケーション 1
81	P107	L	汎用入出力ポート
	AN7		ADC アナログ入力 ch.7
	PPG5_1		PPG 出力 ch.5 リロケーション 1
	DAO1		DAC 出力 ch.1
	ICU11_2		インプットキャプチャ入力 ch.11 リロケーション 2
	SGO4_1		サウンドジェネレータ SGO 出力 ch.4 リロケーション 1
85	P123	L	汎用入出力ポート
	OCU1_0		アウトプットコンペア出力 ch.1 リロケーション 0
	PPG8_2		PPG 出力 ch.8 リロケーション 2
	DAO0		DAC 出力 ch.0
	AN39		ADC アナログ入力 ch.39
86	P122	J	汎用入出力ポート
	OCU0_0		アウトプットコンペア出力 ch.0 リロケーション 0
	SCK5_0		LIN_UART シリアルクロック入出力 ch.5 リロケーション 0
	TOT3_0		リロードタイマ出力 ch.3 リロケーション 0
	PPG7_2		PPG 出力 ch.7 リロケーション 2
	AN38		ADC アナログ入力 ch.38
87	P121	J	汎用入出力ポート
	FRCK0_0		フリーランタイマクロック入力 ch.0 リロケーション 0
	SOT5_0		LIN_UART シリアル出力 ch.5 リロケーション 0
	INT7_0		外部割込み要求入力 ch.7 リロケーション 0
	TOT2_0		リロードタイマ出力 ch.2 リロケーション 0
	PPG6_2		PPG 出力 ch.6 リロケーション 2
	AN37		ADC アナログ入力 ch.37
88	P120	J	汎用入出力ポート
	FRCK1_0		フリーランタイマクロック入力 ch.1 リロケーション 0
	SIN5_0		LIN_UART シリアル入力 ch.5 リロケーション 0
	INT6_0		外部割込み要求入力 ch.6 リロケーション 0
	TOT1_0		リロードタイマ出力 ch.1 リロケーション 0
	PPG5_2		PPG 出力 ch.5 リロケーション 2
	AN36		ADC アナログ入力 ch.36
89	P117	J	汎用入出力ポート
	SCK4_0		LIN_UART シリアルクロック入出力 ch.4 リロケーション 0
	TOT0_0		リロードタイマ出力 ch.0 リロケーション 0
	SGO3		サウンドジェネレータ SGO 出力 ch.3
	TRG4		PPG トリガ入力 4 (ch.16-ch.19)
	FRCK2_0		フリーランタイマクロック入力 ch.2 リロケーション 0
	AN35		ADC アナログ入力 ch.35

端子番号	端子名	入出力回路形式	機　能
90	P116	J	汎用入出力ポート
	SOT4_0		LIN_UART シリアル出力 ch.4 リロケーション 0
	TIN3_0		リロードタイマイベント入力 ch.3 リロケーション 0
	SGA3		サウンドジェネレータ SGA 出力 ch.3
	FRCK3_0		フリーランタイマクロック入力 ch.3 リロケーション 0
	AN34		ADC アナログ入力 ch.34
91	P115	J	汎用入出力ポート
	SIN4_0		LIN_UART シリアル入力 ch.4 リロケーション 0
	TIN2_0		リロードタイマイベント入力 ch.2 リロケーション 0
	SGO2		サウンドジェネレータ SGO 出力 ch.2
	FRCK4_0		フリーランタイマクロック入力 ch.4 リロケーション 0
	AN33		ADC アナログ入力 ch.33
92	P114	J	汎用入出力ポート
	SCK3_0		LIN_UART シリアルクロック入出力 ch.3 リロケーション 0
	TIN1_0		リロードタイマイベント入力 ch.1 リロケーション 0
	ICU5_1		インプットキャプチャ入力 ch.5 リロケーション 1
	SGA2		サウンドジェネレータ SGA 出力 ch.2
	TRG3		PPG トリガ入力 3 (ch.12-ch.15)
	AN32		ADC アナログ入力 ch.32
95	P137	M (Y)	汎用入出力ポート
	(X0A)		サブクロック OSC 入力 (サブクロック搭載品種のみ)
96	P136	M (Y)	汎用入出力ポート
	(X1A)		サブクロック OSC 出力(サブクロック搭載品種のみ)
97	NMIX	R	NMI 端子
98	P097	M	汎用入出力ポート
	WOT		RTC オーバフロー出力
	SOT3_0		LIN_UART シリアル出力 ch.3 リロケーション 0
	INT8_0		外部割込み要求入力 ch.8 リロケーション 0
	TIN0_0		リロードタイマイベント入力 ch.0 リロケーション 0
	ICU4_1		インプットキャプチャ入力 ch.4 リロケーション 1
	PPG0_1		PPG 出力 ch.0 リロケーション 1
99	P094	M	汎用入出力ポート
	SGO1		サウンドジェネレータ SGO 出力 ch.1
	SIN3_0		LIN_UART シリアル入力 ch.3 リロケーション 0
	INT15_0		外部割込み要求入力 ch.15 リロケーション 0
	ICU1_1		インプットキャプチャ入力 ch.1 リロケーション 1
	PPG9_1		PPG 出力 ch.9 リロケーション 1
100	P093	M	汎用入出力ポート
	SGA1		サウンドジェネレータ SGA 出力 ch.1
	SOT2_0		LIN_UART シリアル出力 ch.2 リロケーション 0
	INT14_0		外部割込み要求入力 ch.14 リロケーション 0
	ICU3_1		インプットキャプチャ入力 ch.3 リロケーション 1
	PPG8_1		PPG 出力 ch.8 リロケーション 1

端子番号	端子名	入出力回路形式	機　能
101	P092	M	汎用入出力ポート
	SGO0		サウンドジェネレータ SGO 出力 ch.0
	SCK2_0		LIN_UART シリアルクロック入出力 ch.2 リロケーション 0
	INT13_0		外部割込み要求入力 ch.13 リロケーション 0
	TOT3_1		リロードタイマ出力 ch.3 リロケーション 1
	ICU0_1		インプットキャプチャ入力 ch.0 リロケーション 1
	PPG7_1		PPG 出力 ch.7 リロケーション 1
102	P091	M	汎用入出力ポート
	SGA0		サウンドジェネレータ SGA 出力 ch.0
	SIN2_0		LIN_UART シリアル入力 ch.2 リロケーション 0
	INT12_0		外部割込み要求入力 ch.12 リロケーション 0
	TOT2_1		リロードタイマ出力 ch.2 リロケーション 1
	ICU2_1		インプットキャプチャ入力 ch.2 リロケーション 1
	PPG6_1		PPG 出力 ch.6 リロケーション 1
103	P110	M	汎用入出力ポート
	TX1		CAN TX データ出力 ch.1
	PPG1_2		PPG 出力 ch.1 リロケーション 2
	FRCK5_0		フリーランタイマクロック入力 ch.5 リロケーション 0
104	P111	M	汎用入出力ポート
	RX1		CAN RX データ入力 ch.1
	INT10_0		外部割込み要求入力 ch.10 リロケーション 0
	PPG2_2		PPG 出力 ch.2 リロケーション 2
105	P112	M	汎用入出力ポート
	TX2		CAN TX データ出力 ch.2
	PPG3_2		PPG 出力 ch.3 リロケーション 2
106	P113	M	汎用入出力ポート
	RX2		CAN RX データ入力 ch.2
	INT11_0		外部割込み要求入力 ch.11 リロケーション 0
	PPG4_2		PPG 出力 ch.4 リロケーション 2
107	RSTX	R	リセット端子
110	DEBUGIF	B	DEBUG I/F 端子
111	P095	M	汎用入出力ポート
	TX0		CAN TX データ出力 ch.0
	PPG10_1		PPG 出力 ch.10 リロケーション 1
112	P096	M	汎用入出力ポート
	RX0		CAN RX データ入力 ch.0
	INT9_0		外部割込み要求入力 ch.9 リロケーション 0
113	P124	M	汎用入出力ポート
	OCU2_0		アウトプットコンペア出力 ch.2 リロケーション 0
	ICU5_2		インプットキャプチャ入力 ch.5 リロケーション 2
	PPG9_2		PPG 出力 ch.9 リロケーション 2
114	MD0	A	モード端子
115	MD1	A	モード端子
116	MD2	R2	モード端子
117	X0	X	メインクロック OSC 入力

端子番号	端子名	入出力回路形式	機　能
118	X1	X	メインクロック OSC 出力
120	P125	M	汎用入出力ポート
	OCU3_0		アウトプットコンペア出力 ch.3 リロケーション 0
	ICU0_0		インプットキャプチャ入力 ch.0 リロケーション 0
	PPG10_2		PPG 出力 ch.10 リロケーション 2
121	P126	M	汎用入出力ポート
	TRG0		PPG トリガ入力 0 (ch.0-ch.3)
	SIN0_0		マルチファンクションシリアル入力 ch.0 リロケーション 0
	INT1_0		外部割込み要求入力 ch.1 リロケーション 0
	OCU4_0		アウトプットコンペア出力 ch.4 リロケーション 0
122	P127	N	汎用入出力ポート
	SOT0_0		マルチファンクションシリアル出力 ch.0 リロケーション 0
	OCU5_0		アウトプットコンペア出力 ch.5 リロケーション 0
123	P130	N	汎用入出力ポート
	SCK0_0		マルチファンクションシリアルクロック入出力 ch.0 リロケーション 0
	INT0_0		外部割込み要求入力 ch.0 リロケーション 0
	ICU1_0		インプットキャプチャ入力 ch.1 リロケーション 0
	TIOA0		ベースタイマ出力 ch.0
124	P131	M	汎用入出力ポート
	TRG1		PPG トリガ入力 1 (ch.4-ch.7)
	SIN1_0		マルチファンクションシリアル入力 ch.1 リロケーション 0
	INT4_0		外部割込み要求入力 ch.4 リロケーション 0
	ICU2_0		インプットキャプチャ入力 ch.2 リロケーション 0
	TIOA1		ベースタイマ入出力 ch.1
125	P132	N	汎用入出力ポート
	SOT1_0		マルチファンクションシリアル出力 ch.1 リロケーション 0
	INT2_0		外部割込み要求入力 ch.2 リロケーション 0
	ICU3_0		インプットキャプチャ入力 ch.3 リロケーション 0
	TIOB0		ベースタイマ入力 ch.0
126	P133	N	汎用入出力ポート
	SCK1_0		マルチファンクションシリアルクロック入出力 ch.1 リロケーション 0
	INT3_0		外部割込み要求入力 ch.3 リロケーション 0
	ICU4_0		インプットキャプチャ入力 ch.4 リロケーション 0
	TIOB1		ベースタイマ入力 ch.1
	PPG11_1		PPG 出力 ch.11 リロケーション 1
	TRG5		PPG トリガ入力 5 (ch.20-ch.23)
127	P134	M	汎用入出力ポート
	TRG2		PPG トリガ入力 2 (ch.8-ch.11)
	INT5_0		外部割込み要求入力 ch.5 リロケーション 0
	ICU5_0		インプットキャプチャ入力 ch.5 リロケーション 0
	PPG1_3		PPG 出力 ch.1 リロケーション 3

端子番号	端子名	入出力回路形式	機　能
131	P000	M	汎用入出力ポート
	D16_0		外部バスデータ入出力
	SIN2_1		LIN_UART シリアル入力 ch.2 リロケーション 1
	TIN0_2		リロードタイマイベント入力 ch.0 リロケーション 2
	PPG0_0		PPG 出力 ch.0 リロケーション 0
	D24_1		外部バスデータ入出力
	INT0_1		外部割込み要求入力 ch.0 リロケーション 1
132	P001	M	汎用入出力ポート
	D17_0		外部バスデータ入出力
	SOT2_1		LIN_UART シリアル出力 ch.2 リロケーション 1
	TIN1_2		リロードタイマイベント入力 ch.1 リロケーション 2
	PPG1_0		PPG 出力 ch.1 リロケーション 0
	D25_1		外部バスデータ入出力
	INT1_1		外部割込み要求入力 ch.1 リロケーション 1
133	P002	M	汎用入出力ポート
	D18_0		外部バスデータ入出力
	SCK2_1		LIN_UART シリアルクロック入出力 ch.2 リロケーション 1
	TIN2_2		リロードタイマイベント入力 ch.2 リロケーション 2
	PPG2_0		PPG 出力 ch.2 リロケーション 0
	D26_1		外部バスデータ入出力
	INT2_1		外部割込み要求入力 ch.2 リロケーション 1
134	P003	M	汎用入出力ポート
	D19_0		外部バスデータ入出力
	SIN3_1		LIN_UART シリアル入力 ch.3 リロケーション 1
	TIN3_2		リロードタイマイベント入力 ch.3 リロケーション 2
	PPG3_0		PPG 出力 ch.3 リロケーション 0
	D27_1		外部バスデータ入出力
	INT3_1		外部割込み要求入力 ch.3 リロケーション 1
135	P004	M	汎用入出力ポート
	D20_0		外部バスデータ入出力
	SOT3_1		LIN_UART シリアル出力 ch.3 リロケーション 1
	TOT0_2		リロードタイマ出力 ch.0 リロケーション 2
	PPG4_0		PPG 出力 ch.4 リロケーション 0
	D28_1		外部バスデータ入出力
	INT4_1		外部割込み要求入力 ch.4 リロケーション 1
136	P005	M	汎用入出力ポート
	D21_0		外部バスデータ入出力
	SCK3_1		LIN_UART シリアルクロック入出力 ch.3 リロケーション 1
	TOT1_2		リロードタイマ出力 ch.1 リロケーション 2
	PPG5_0		PPG 出力 ch.5 リロケーション 0
	D29_1		外部バスデータ入出力
	INT5_1		外部割込み要求入力 ch.5 リロケーション 1

端子番号	端子名	入出力回路形式	機　能
137	P006	M	汎用入出力ポート
	D22_0		外部バスデータ入出力
	TOT2_2		リロードタイマ出力 ch.2 リロケーション 2
	PPG6_0		PPG 出力 ch.6 リロケーション 0
	D30_1		外部バスデータ入出力
	INT6_1		外部割込み要求入力 ch.6 リロケーション 1
138	P007	M	汎用入出力ポート
	D23_0		外部バスデータ入出力
	TOT3_2		リロードタイマ出力 ch.3 リロケーション 2
	PPG7_0		PPG 出力 ch.7 リロケーション 0
	D31_1		外部バスデータ入出力
	INT7_1		外部割込み要求入力 ch.7 リロケーション 1
139	P010	H	汎用入出力ポート
	D24_0		外部バスデータ入出力
	SEG0		LCDC セグメント(Duty)出力
	D16_1		外部バスデータ入出力
	INT8_1		外部割込み要求入力 ch.8 リロケーション 1
140	P011	H	汎用入出力ポート
	D25_0		外部バスデータ入出力
	SEG1		LCDC セグメント(Duty)出力
	D17_1		外部バスデータ入出力
	INT9_1		外部割込み要求入力 ch.9 リロケーション 1
141	P012	H	汎用入出力ポート
	D26_0		外部バスデータ入出力
	SEG2		LCDC セグメント(Duty)出力
	D18_1		外部バスデータ入出力
	INT10_1		外部割込み要求入力 ch.10 リロケーション 1
142	P013	H	汎用入出力ポート
	D27_0		外部バスデータ入出力
	SEG3		LCDC セグメント(Duty)出力
	D19_1		外部バスデータ入出力
	INT11_1		外部割込み要求入力 ch.11 リロケーション 1
143	P014	H	汎用入出力ポート
	D28_0		外部バスデータ入出力
	SEG4		LCDC セグメント(Duty)出力
	D20_1		外部バスデータ入出力
	INT12_1		外部割込み要求入力 ch.12 リロケーション 1
1	VCCE	-	+3.3v/+5.0v 電源
20	VCCE	-	+3.3v/+5.0v 電源
21	VSS	-	GND
36	VCC5	-	+5.0v 電源
37	VSS	-	GND
41	DVCC	-	SMC 大電流用電源
42	DVSS	-	SMC 大電流用 GND
51	DVCC	-	SMC 大電流用電源

端子番号	端子名	入出力回路形式	機　能
52	DVSS	-	SMC 大電流用 GND
61	DVCC	-	SMC 大電流用電源
62	DVSS	-	SMC 大電流用 GND
71	DVCC	-	SMC 大電流用電源
72	DVSS	-	SMC 大電流用 GND
82	AVSS/AVRL	-	ADC,DAC GND / 下限基準電圧
83	AVRH	-	ADC 上限基準電圧
84	AVCC	-	ADC, DAC アナログ電源
93	VCC5	-	+5.0v 電源
94	VSS	-	GND
108	VCC5	-	+5.0v 電源
109	VSS	-	GND
119	VSS	-	GND
128	VCC5	-	+5.0v 電源
129	VSS	-	GND
130	C	-	外部容量接続端子
144	VSS	-	GND

(注意事項)　　CY91F578/9の端子機能説明はデータシートを参照してください。

1.10 各機能の端子一覧

各機能の端子一覧を示します。

1.10.1. A/Dコンバータの端子

1.10.2. ベースタイマの端子(ch.0, ch.1)

1.10.3. CANの端子(ch.0～ch.2)

1.10.4. DACの端子 (ch.0, ch.1)

1.10.5. 外部バスインタフェースの端子

1.10.6. 外部割込み要求入力の端子(ch.0～ch.15)

1.10.7. フリーランタイマの端子(ch.0～ch.5)

1.10.8. 汎用入出力ポートの端子

1.10.9. HS_SPIの端子

1.10.10. インプットキャプチャの端子(ch.0～ch.11)

1.10.11. LCDCの端子

1.10.12. LIN-UARTの端子(ch.2～ch.7)

1.10.13. マルチファンクションシリアルインタフェースの端子(ch.0, ch.1, ch.8, ch.9)

1.10.14. アウトプットコンペアの端子(ch.0～ch.11)

1.10.15. PPGの端子(ch.0～ch.23)

1.10.16. リロードタイマの端子(ch.0～ch.3)

1.10.17. リアルタイムクロックの端子

1.10.18. ステッピングモータ制御の端子(ch.0～ch.5)

1.10.19. サウンドジェネレータの端子(ch.0～ch.4)

1.10.20. アップダウンカウンタの端子(ch.0, ch.1)

1.10.21. クロック端子

1.10.22. モード端子

1.10.23. リセット端子

1.10.24. 電源端子

1.10.1 A/D コンバータの端子

A/Dコンバータの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- ADC 外部トリガ入力 (端子名) ADTG (端子番号) 73
- ADC アナログ入力 ch.0 (端子名) AN0 (端子番号) 74
- ADC アナログ入力 ch.1 (端子名) AN1 (端子番号) 75
- ADC アナログ入力 ch.2 (端子名) AN2 (端子番号) 76
- ADC アナログ入力 ch.3 (端子名) AN3 (端子番号) 77
- ADC アナログ入力 ch.4 (端子名) AN4 (端子番号) 78
- ADC アナログ入力 ch.5 (端子名) AN5 (端子番号) 79
- ADC アナログ入力 ch.6 (端子名) AN6 (端子番号) 80
- ADC アナログ入力 ch.7 (端子名) AN7 (端子番号) 81
- ADC アナログ入力 ch.8 (端子名) AN8 (端子番号) 43
- ADC アナログ入力 ch.9 (端子名) AN9 (端子番号) 44
- ADC アナログ入力 ch.10 (端子名) AN10 (端子番号) 45
- ADC アナログ入力 ch.11 (端子名) AN11 (端子番号) 46
- ADC アナログ入力 ch.12 (端子名) AN12 (端子番号) 47
- ADC アナログ入力 ch.13 (端子名) AN13 (端子番号) 48
- ADC アナログ入力 ch.14 (端子名) AN14 (端子番号) 49
- ADC アナログ入力 ch.15 (端子名) AN15 (端子番号) 50
- ADC アナログ入力 ch.16 (端子名) AN16 (端子番号) 53
- ADC アナログ入力 ch.17 (端子名) AN17 (端子番号) 54
- ADC アナログ入力 ch.18 (端子名) AN18 (端子番号) 55
- ADC アナログ入力 ch.19 (端子名) AN19 (端子番号) 56
- ADC アナログ入力 ch.20 (端子名) AN20 (端子番号) 57
- ADC アナログ入力 ch.21 (端子名) AN21 (端子番号) 58
- ADC アナログ入力 ch.22 (端子名) AN22 (端子番号) 59
- ADC アナログ入力 ch.23 (端子名) AN23 (端子番号) 60
- ADC アナログ入力 ch.24 (端子名) AN24 (端子番号) 63
- ADC アナログ入力 ch.25 (端子名) AN25 (端子番号) 64
- ADC アナログ入力 ch.26 (端子名) AN26 (端子番号) 65
- ADC アナログ入力 ch.27 (端子名) AN27 (端子番号) 66
- ADC アナログ入力 ch.28 (端子名) AN28 (端子番号) 67
- ADC アナログ入力 ch.29 (端子名) AN29 (端子番号) 68
- ADC アナログ入力 ch.30 (端子名) AN30 (端子番号) 69
- ADC アナログ入力 ch.31 (端子名) AN31 (端子番号) 70
- ADC アナログ入力 ch.32 (端子名) AN32 (端子番号) 92
- ADC アナログ入力 ch.33 (端子名) AN33 (端子番号) 91
- ADC アナログ入力 ch.34 (端子名) AN34 (端子番号) 90
- ADC アナログ入力 ch.35 (端子名) AN35 (端子番号) 89
- ADC アナログ入力 ch.36 (端子名) AN36 (端子番号) 88
- ADC アナログ入力 ch.37 (端子名) AN37 (端子番号) 87
- ADC アナログ入力 ch.38 (端子名) AN38 (端子番号) 86
- ADC アナログ入力 ch.39 (端子名) AN39 (端子番号) 85

1.10.2 ベースタイマの端子 (ch.0, ch.1)

ベースタイマの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- ベースタイマ出力 ch.0 (端子名) TIOA0 (端子番号) 123
- ベースタイマ入力 ch.0 (端子名) TIOB0 (端子番号) 125
- ベースタイマ入出力 ch.1 (端子名) TIOA1 (端子番号) 124
- ベースタイマ入力 ch.1 (端子名) TIOB1 (端子番号) 126

1.10.3 CAN の端子 (ch.0～ch.2)

CANの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- CAN RX データ入力 ch.0 (端子名) RX0 (端子番号) 112
- CAN RX データ入力 ch.1 (端子名) RX1 (端子番号) 104
- CAN RX データ入力 ch.2 (端子名) RX2 (端子番号) 106
- CAN TX データ出力 ch.0 (端子名) TX0 (端子番号) 111
- CAN TX データ出力 ch.1 (端子名) TX1 (端子番号) 103
- CAN TX データ出力 ch.2 (端子名) TX2 (端子番号) 105

1.10.4 DAC の端子 (ch.0, ch.1)

DACの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- DAC 出力 ch.0 (端子名) DAO0 (端子番号) 85
- DAC 出力 ch.1 (端子名) DAO1 (端子番号) 81

1.10.5 外部バスインタフェースの端子

外部バスインタフェースの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- 外部バスアドレス出力 (端子名) A00 (端子番号) 11
- 外部バスアドレス出力 (端子名) A01 (端子番号) 12
- 外部バスアドレス出力 (端子名) A02 (端子番号) 13
- 外部バスアドレス出力 (端子名) A03 (端子番号) 14
- 外部バスアドレス出力 (端子名) A04 (端子番号) 15
- 外部バスアドレス出力 (端子名) A05 (端子番号) 16
- 外部バスアドレス出力 (端子名) A06 (端子番号) 17
- 外部バスアドレス出力 (端子名) A07 (端子番号) 18
- 外部バスアドレス出力 (端子名) A08 (端子番号) 19
- 外部バスアドレス出力 (端子名) A09 (端子番号) 22

- 外部バスアドレス出力 (端子名) A10 (端子番号) 23
- 外部バスアドレス出力 (端子名) A11 (端子番号) 24
- 外部バスアドレス出力 (端子名) A12 (端子番号) 25
- 外部バスアドレス出力 (端子名) A13 (端子番号) 26
- 外部バスアドレス出力 (端子名) A14 (端子番号) 27
- 外部バスアドレス出力 (端子名) A15 (端子番号) 28
- 外部バスアドレス出力 (端子名) A16 (端子番号) 29
- 外部バスアドレス出力 (端子名) A17 (端子番号) 30
- 外部バスアドレス出力 (端子名) A18 (端子番号) 31
- 外部バスアドレス出力 (端子名) A19 (端子番号) 32
- 外部バスアドレス出力 (端子名) A20 (端子番号) 33
- 外部バスアドレス出力 (端子名) A21 (端子番号) 34
- 外部バスアドレスストローブ出力 (端子名) ASX (端子番号) 5
- 外部バスチップセレクト 0 出力 (端子名) CS0X (端子番号) 6
- 外部バスチップセレクト 1 出力 (端子名) CS1X (端子番号) 7
- 外部バスチップセレクト 2 出力 (端子名) CS2X (端子番号) 38
- 外部バスチップセレクト 3 出力 (端子名) CS3X (端子番号) 39
- 外部バスクロック出力 (端子名) SYSCLK (端子番号) 35
- 外部バスデータ入出力 (端子名) D16_0 (端子番号) 131
- 外部バスデータ入出力 (端子名) D16_1 (端子番号) 139
- 外部バスデータ入出力 (端子名) D17_0 (端子番号) 132
- 外部バスデータ入出力 (端子名) D17_1 (端子番号) 140
- 外部バスデータ入出力 (端子名) D18_0 (端子番号) 133
- 外部バスデータ入出力 (端子名) D18_1 (端子番号) 141
- 外部バスデータ入出力 (端子名) D19_0 (端子番号) 134
- 外部バスデータ入出力 (端子名) D19_1 (端子番号) 142
- 外部バスデータ入出力 (端子名) D20_0 (端子番号) 135
- 外部バスデータ入出力 (端子名) D20_1 (端子番号) 143
- 外部バスデータ入出力 (端子名) D21_0 (端子番号) 136
- 外部バスデータ入出力 (端子名) D21_1 (端子番号) 2
- 外部バスデータ入出力 (端子名) D22_0 (端子番号) 137
- 外部バスデータ入出力 (端子名) D22_1 (端子番号) 3
- 外部バスデータ入出力 (端子名) D23_0 (端子番号) 138
- 外部バスデータ入出力 (端子名) D23_1 (端子番号) 4
- 外部バスデータ入出力 (端子名) D24_0 (端子番号) 139
- 外部バスデータ入出力 (端子名) D24_1 (端子番号) 131
- 外部バスデータ入出力 (端子名) D25_0 (端子番号) 140
- 外部バスデータ入出力 (端子名) D25_1 (端子番号) 132
- 外部バスデータ入出力 (端子名) D26_0 (端子番号) 141
- 外部バスデータ入出力 (端子名) D26_1 (端子番号) 133
- 外部バスデータ入出力 (端子名) D27_0 (端子番号) 142
- 外部バスデータ入出力 (端子名) D27_1 (端子番号) 134
- 外部バスデータ入出力 (端子名) D28_0 (端子番号) 143
- 外部バスデータ入出力 (端子名) D28_1 (端子番号) 135
- 外部バスデータ入出力 (端子名) D29_0 (端子番号) 2
- 外部バスデータ入出力 (端子名) D29_1 (端子番号) 136
- 外部バスデータ入出力 (端子名) D30_0 (端子番号) 3
- 外部バスデータ入出力 (端子名) D30_1 (端子番号) 137
- 外部バスデータ入出力 (端子名) D31_0 (端子番号) 4

- 外部バスデータ入出力 (端子名) D31_1 (端子番号) 138
- 外部バス RDY 入力 (端子名) RDY (端子番号) 40
- 外部バスリードストローブ出力 (端子名) RDX (端子番号) 8
- 外部バスライトストローブ 0 出力 (端子名) WR0X (端子番号) 9
- 外部バスライトストローブ 1 出力 (端子名) WR1X (端子番号) 10

1.10.6 外部割込み要求入力の端子 (ch.0～ch.15)

外部割込み要求入力の端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- 外部割込み要求入力 ch.0 リロケーション 0 (端子名) INT0_0 (端子番号) 123
- 外部割込み要求入力 ch.0 リロケーション 1 (端子名) INT0_1 (端子番号) 131
- 外部割込み要求入力 ch.1 リロケーション 0 (端子名) INT1_0 (端子番号) 121
- 外部割込み要求入力 ch.1 リロケーション 1 (端子名) INT1_1 (端子番号) 132
- 外部割込み要求入力 ch.2 リロケーション 0 (端子名) INT2_0 (端子番号) 125
- 外部割込み要求入力 ch.2 リロケーション 1 (端子名) INT2_1 (端子番号) 133
- 外部割込み要求入力 ch.3 リロケーション 0 (端子名) INT3_0 (端子番号) 126
- 外部割込み要求入力 ch.3 リロケーション 1 (端子名) INT3_1 (端子番号) 134
- 外部割込み要求入力 ch.4 リロケーション 0 (端子名) INT4_0 (端子番号) 124
- 外部割込み要求入力 ch.4 リロケーション 1 (端子名) INT4_1 (端子番号) 135
- 外部割込み要求入力 ch.5 リロケーション 0 (端子名) INT5_0 (端子番号) 127
- 外部割込み要求入力 ch.5 リロケーション 1 (端子名) INT5_1 (端子番号) 136
- 外部割込み要求入力 ch.6 リロケーション 0 (端子名) INT6_0 (端子番号) 88
- 外部割込み要求入力 ch.6 リロケーション 1 (端子名) INT6_1 (端子番号) 137
- 外部割込み要求入力 ch.7 リロケーション 0 (端子名) INT7_0 (端子番号) 87
- 外部割込み要求入力 ch.7 リロケーション 1 (端子名) INT7_1 (端子番号) 138
- 外部割込み要求入力 ch.8 リロケーション 0 (端子名) INT8_0 (端子番号) 98
- 外部割込み要求入力 ch.8 リロケーション 1 (端子名) INT8_1 (端子番号) 139
- 外部割込み要求入力 ch.9 リロケーション 0 (端子名) INT9_0 (端子番号) 112
- 外部割込み要求入力 ch.9 リロケーション 1 (端子名) INT9_1 (端子番号) 140
- 外部割込み要求入力 ch.10 リロケーション 0 (端子名) INT10_0 (端子番号) 104
- 外部割込み要求入力 ch.10 リロケーション 1 (端子名) INT10_1 (端子番号) 141
- 外部割込み要求入力 ch.11 リロケーション 0 (端子名) INT11_0 (端子番号) 106
- 外部割込み要求入力 ch.11 リロケーション 1 (端子名) INT11_1 (端子番号) 142
- 外部割込み要求入力 ch.12 リロケーション 0 (端子名) INT12_0 (端子番号) 102
- 外部割込み要求入力 ch.12 リロケーション 1 (端子名) INT12_1 (端子番号) 143
- 外部割込み要求入力 ch.13 リロケーション 0 (端子名) INT13_0 (端子番号) 101
- 外部割込み要求入力 ch.13 リロケーション 1 (端子名) INT13_1 (端子番号) 2
- 外部割込み要求入力 ch.14 リロケーション 0 (端子名) INT14_0 (端子番号) 100
- 外部割込み要求入力 ch.14 リロケーション 1 (端子名) INT14_1 (端子番号) 3
- 外部割込み要求入力 ch.15 リロケーション 0 (端子名) INT15_0 (端子番号) 99
- 外部割込み要求入力 ch.15 リロケーション 1 (端子名) INT15_1 (端子番号) 4

1.10.7　フリーランタイマの端子 (ch.0～ch.5)

フリーランタイマの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- フリーランタイマクロック入力 ch.0 リロケーション 0　(端子名) FRCK0_0　(端子番号) 87
- フリーランタイマクロック入力 ch.0 リロケーション 1　(端子名) FRCK0_1　(端子番号) 35
- フリーランタイマクロック入力 ch.1 リロケーション 0　(端子名) FRCK1_0　(端子番号) 88
- フリーランタイマクロック入力 ch.1 リロケーション 1　(端子名) FRCK1_1　(端子番号) 38
- フリーランタイマクロック入力 ch.2 リロケーション 0　(端子名) FRCK2_0　(端子番号) 89
- フリーランタイマクロック入力 ch.2 リロケーション 1　(端子名) FRCK2_1　(端子番号) 39
- フリーランタイマクロック入力 ch.3 リロケーション 0　(端子名) FRCK3_0　(端子番号) 90
- フリーランタイマクロック入力 ch.3 リロケーション 1　(端子名) FRCK3_1　(端子番号) 40
- フリーランタイマクロック入力 ch.4 リロケーション 0　(端子名) FRCK4_0　(端子番号) 91
- フリーランタイマクロック入力 ch.5 リロケーション 0　(端子名) FRCK5_0　(端子番号) 103

1.10.8　汎用入出力ポートの端子

汎用入力ポートの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- 汎用入出力ポート　(端子名) P000　(端子番号) 131
- 汎用入出力ポート　(端子名) P001　(端子番号) 132
- 汎用入出力ポート　(端子名) P002　(端子番号) 133
- 汎用入出力ポート　(端子名) P003　(端子番号) 134
- 汎用入出力ポート　(端子名) P004　(端子番号) 135
- 汎用入出力ポート　(端子名) P005　(端子番号) 136
- 汎用入出力ポート　(端子名) P006　(端子番号) 137
- 汎用入出力ポート　(端子名) P007　(端子番号) 138
- 汎用入出力ポート　(端子名) P010　(端子番号) 139
- 汎用入出力ポート　(端子名) P011　(端子番号) 140
- 汎用入出力ポート　(端子名) P012　(端子番号) 141
- 汎用入出力ポート　(端子名) P013　(端子番号) 142
- 汎用入出力ポート　(端子名) P014　(端子番号) 143
- 汎用入出力ポート　(端子名) P015　(端子番号)　2
- 汎用入出力ポート　(端子名) P016　(端子番号)　3
- 汎用入出力ポート　(端子名) P017　(端子番号)　4
- 汎用入出力ポート　(端子名) P020　(端子番号)　5
- 汎用入出力ポート　(端子名) P021　(端子番号)　6
- 汎用入出力ポート　(端子名) P022　(端子番号)　7
- 汎用入出力ポート　(端子名) P023　(端子番号)　8
- 汎用入出力ポート　(端子名) P024　(端子番号)　9
- 汎用入出力ポート　(端子名) P025　(端子番号) 10
- 汎用入出力ポート　(端子名) P026　(端子番号) 11
- 汎用入出力ポート　(端子名) P027　(端子番号) 12
- 汎用入出力ポート　(端子名) P030　(端子番号) 13
- 汎用入出力ポート　(端子名) P031　(端子番号) 14

- 汎用入出力ポート (端子名) P032 (端子番号) 15
- 汎用入出力ポート (端子名) P033 (端子番号) 16
- 汎用入出力ポート (端子名) P034 (端子番号) 17
- 汎用入出力ポート (端子名) P035 (端子番号) 18
- 汎用入出力ポート (端子名) P036 (端子番号) 19
- 汎用入出力ポート (端子名) P037 (端子番号) 22
- 汎用入出力ポート (端子名) P040 (端子番号) 23
- 汎用入出力ポート (端子名) P041 (端子番号) 24
- 汎用入出力ポート (端子名) P042 (端子番号) 25
- 汎用入出力ポート (端子名) P043 (端子番号) 26
- 汎用入出力ポート (端子名) P044 (端子番号) 27
- 汎用入出力ポート (端子名) P045 (端子番号) 28
- 汎用入出力ポート (端子名) P046 (端子番号) 29
- 汎用入出力ポート (端子名) P047 (端子番号) 30
- 汎用入出力ポート (端子名) P050 (端子番号) 31
- 汎用入出力ポート (端子名) P051 (端子番号) 32
- 汎用入出力ポート (端子名) P052 (端子番号) 33
- 汎用入出力ポート (端子名) P053 (端子番号) 34
- 汎用入出力ポート (端子名) P054 (端子番号) 35
- 汎用入出力ポート (端子名) P055 (端子番号) 38
- 汎用入出力ポート (端子名) P056 (端子番号) 39
- 汎用入出力ポート(入力のみ出力なし) (端子名) P057 (端子番号) 40
- 汎用入出力ポート (端子名) P060 (端子番号) 43
- 汎用入出力ポート (端子名) P061 (端子番号) 44
- 汎用入出力ポート (端子名) P062 (端子番号) 45
- 汎用入出力ポート (端子名) P063 (端子番号) 46
- 汎用入出力ポート (端子名) P064 (端子番号) 47
- 汎用入出力ポート (端子名) P065 (端子番号) 48
- 汎用入出力ポート (端子名) P066 (端子番号) 49
- 汎用入出力ポート (端子名) P067 (端子番号) 50
- 汎用入出力ポート (端子名) P070 (端子番号) 53
- 汎用入出力ポート (端子名) P071 (端子番号) 54
- 汎用入出力ポート (端子名) P072 (端子番号) 55
- 汎用入出力ポート (端子名) P073 (端子番号) 56
- 汎用入出力ポート (端子名) P074 (端子番号) 57
- 汎用入出力ポート (端子名) P075 (端子番号) 58
- 汎用入出力ポート (端子名) P076 (端子番号) 59
- 汎用入出力ポート (端子名) P077 (端子番号) 60
- 汎用入出力ポート (端子名) P080 (端子番号) 63
- 汎用入出力ポート (端子名) P081 (端子番号) 64
- 汎用入出力ポート (端子名) P082 (端子番号) 65
- 汎用入出力ポート (端子名) P083 (端子番号) 66
- 汎用入出力ポート (端子名) P084 (端子番号) 67
- 汎用入出力ポート (端子名) P085 (端子番号) 68
- 汎用入出力ポート (端子名) P086 (端子番号) 69
- 汎用入出力ポート (端子名) P087 (端子番号) 70
- 汎用入出力ポート (端子名) P090 (端子番号) 73
- 汎用入出力ポート (端子名) P091 (端子番号) 102
- 汎用入出力ポート (端子名) P092 (端子番号) 101

- 汎用入出力ポート (端子名) P093 (端子番号) 100
- 汎用入出力ポート (端子名) P094 (端子番号) 99
- 汎用入出力ポート (端子名) P095 (端子番号) 111
- 汎用入出力ポート (端子名) P096 (端子番号) 112
- 汎用入出力ポート (端子名) P097 (端子番号) 98
- 汎用入出力ポート (端子名) P100 (端子番号) 74
- 汎用入出力ポート (端子名) P101 (端子番号) 75
- 汎用入出力ポート (端子名) P102 (端子番号) 76
- 汎用入出力ポート (端子名) P103 (端子番号) 77
- 汎用入出力ポート (端子名) P104 (端子番号) 78
- 汎用入出力ポート (端子名) P105 (端子番号) 79
- 汎用入出力ポート (端子名) P106 (端子番号) 80
- 汎用入出力ポート (端子名) P107 (端子番号) 81
- 汎用入出力ポート (端子名) P110 (端子番号) 103
- 汎用入出力ポート (端子名) P111 (端子番号) 104
- 汎用入出力ポート (端子名) P112 (端子番号) 105
- 汎用入出力ポート (端子名) P113 (端子番号) 106
- 汎用入出力ポート (端子名) P114 (端子番号) 92
- 汎用入出力ポート (端子名) P115 (端子番号) 91
- 汎用入出力ポート (端子名) P116 (端子番号) 90
- 汎用入出力ポート (端子名) P117 (端子番号) 89
- 汎用入出力ポート (端子名) P120 (端子番号) 88
- 汎用入出力ポート (端子名) P121 (端子番号) 87
- 汎用入出力ポート (端子名) P122 (端子番号) 86
- 汎用入出力ポート (端子名) P123 (端子番号) 85
- 汎用入出力ポート (端子名) P124 (端子番号) 113
- 汎用入出力ポート (端子名) P125 (端子番号) 120
- 汎用入出力ポート (端子名) P126 (端子番号) 121
- 汎用入出力ポート (端子名) P127 (端子番号) 122
- 汎用入出力ポート (端子名) P130 (端子番号) 123
- 汎用入出力ポート (端子名) P131 (端子番号) 124
- 汎用入出力ポート (端子名) P132 (端子番号) 125
- 汎用入出力ポート (端子名) P133 (端子番号) 126
- 汎用入出力ポート (端子名) P134 (端子番号) 127
- 汎用入出力ポート (端子名) P136 (端子番号) 96
- 汎用入出力ポート (端子名) P137 (端子番号) 95

1.10.9 HS_SPI の端子

HS_SPIの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- HS_SPI SCLK 入出力 (端子名) SPI_CLK (端子番号) 19
- HS_SPI SDATA0 入出力 (端子名) SPI_SIO0 (端子番号) 18
- HS_SPI SDATA1 入出力 (端子名) SPI_SIO1 (端子番号) 17
- HS_SPI SDATA2 入出力 (端子名) SPI_SIO2 (端子番号) 16
- HS_SPI SDATA3 入出力 (端子名) SPI_SIO3 (端子番号) 15
- HS_SPI SSEL0 入出力 (端子名) SPI_CS0 (端子番号) 14
- HS_SPI SSEL1 出力 (端子名) SPI_CS1 (端子番号) 13
- HS_SPI SSEL2 出力 (端子名) SPI_CS2 (端子番号) 12
- HS_SPI SSEL3 出力 (端子名) SPI_CS3 (端子番号) 11

(注意事項)

本シリーズではHS_SPIは使用禁止です

1.10.10 インプットキャプチャの端子 (ch.0～ch.11)

インプットキャプチャの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- インプットキャプチャ入力 ch.0 リロケーション 0 (端子名) ICU0_0 (端子番号) 120
- インプットキャプチャ入力 ch.0 リロケーション 1 (端子名) ICU0_1 (端子番号) 101
- インプットキャプチャ入力 ch.0 リロケーション 2 (端子名) ICU0_2 (端子番号) 66
- インプットキャプチャ入力 ch.1 リロケーション 0 (端子名) ICU1_0 (端子番号) 123
- インプットキャプチャ入力 ch.1 リロケーション 1 (端子名) ICU1_1 (端子番号) 99
- インプットキャプチャ入力 ch.1 リロケーション 2 (端子名) ICU1_2 (端子番号) 67
- インプットキャプチャ入力 ch.2 リロケーション 0 (端子名) ICU2_0 (端子番号) 124
- インプットキャプチャ入力 ch.2 リロケーション 1 (端子名) ICU2_1 (端子番号) 102
- インプットキャプチャ入力 ch.2 リロケーション 2 (端子名) ICU2_2 (端子番号) 68
- インプットキャプチャ入力 ch.3 リロケーション 0 (端子名) ICU3_0 (端子番号) 125
- インプットキャプチャ入力 ch.3 リロケーション 1 (端子名) ICU3_1 (端子番号) 100
- インプットキャプチャ入力 ch.3 リロケーション 2 (端子名) ICU3_2 (端子番号) 69
- インプットキャプチャ入力 ch.4 リロケーション 0 (端子名) ICU4_0 (端子番号) 126
- インプットキャプチャ入力 ch.4 リロケーション 1 (端子名) ICU4_1 (端子番号) 98
- インプットキャプチャ入力 ch.4 リロケーション 2 (端子名) ICU4_2 (端子番号) 70
- インプットキャプチャ入力 ch.5 リロケーション 0 (端子名) ICU5_0 (端子番号) 127
- インプットキャプチャ入力 ch.5 リロケーション 1 (端子名) ICU5_1 (端子番号) 92
- インプットキャプチャ入力 ch.5 リロケーション 2 (端子名) ICU5_2 (端子番号) 113
- インプットキャプチャ入力 ch.6 リロケーション 0 (端子名) ICU6_0 (端子番号) 5
- インプットキャプチャ入力 ch.6 リロケーション 1 (端子名) ICU6_1 (端子番号) 60
- インプットキャプチャ入力 ch.6 リロケーション 2 (端子名) ICU6_2 (端子番号) 76
- インプットキャプチャ入力 ch.7 リロケーション 0 (端子名) ICU7_0 (端子番号) 6
- インプットキャプチャ入力 ch.7 リロケーション 1 (端子名) ICU7_1 (端子番号) 59
- インプットキャプチャ入力 ch.7 リロケーション 2 (端子名) ICU7_2 (端子番号) 77

- インプットキャプチャ入力 ch.8 リロケーション 0 (端子名) ICU8_0 (端子番号) 7
- インプットキャプチャ入力 ch.8 リロケーション 1 (端子名) ICU8_1 (端子番号) 58
- インプットキャプチャ入力 ch.8 リロケーション 2 (端子名) ICU8_2 (端子番号) 78
- インプットキャプチャ入力 ch.9 リロケーション 0 (端子名) ICU9_0 (端子番号) 8
- インプットキャプチャ入力 ch.9 リロケーション 1 (端子名) ICU9_1 (端子番号) 57
- インプットキャプチャ入力 ch.9 リロケーション 2 (端子名) ICU9_2 (端子番号) 79
- インプットキャプチャ入力 ch.10 リロケーション 0 (端子名) ICU10_0 (端子番号) 9
- インプットキャプチャ入力 ch.10 リロケーション 1 (端子名) ICU10_1 (端子番号) 56
- インプットキャプチャ入力 ch.10 リロケーション 2 (端子名) ICU10_2 (端子番号) 80
- インプットキャプチャ入力 ch.11 リロケーション 0 (端子名) ICU11_0 (端子番号) 10
- インプットキャプチャ入力 ch.11 リロケーション 1 (端子名) ICU11_1 (端子番号) 55
- インプットキャプチャ入力 ch.11 リロケーション 2 (端子名) ICU11_2 (端子番号) 81

1.10.11 LCDC の端子

LCDCの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- LCDC 基準電圧 V0 入力 (端子名) V0 (端子番号) 35
- LCDC 基準電圧 V1 入力 (端子名) V1 (端子番号) 38
- LCDC 基準電圧 V2 入力 (端子名) V2 (端子番号) 39
- LCDC 基準電圧 V3 入力 (端子名) V3 (端子番号) 40
- LCDC セグメント(Duty)コモン出力 (端子名) COM0 (端子番号) 31
- LCDC セグメント(Duty)コモン出力 (端子名) COM1 (端子番号) 32
- LCDC セグメント(Duty)コモン出力 (端子名) COM2 (端子番号) 33
- LCDC セグメント(Duty)コモン出力 (端子名) COM3 (端子番号) 34
- LCDC セグメント(Duty)出力 (端子名) SEG0 (端子番号) 139
- LCDC セグメント(Duty)出力 (端子名) SEG1 (端子番号) 140
- LCDC セグメント(Duty)出力 (端子名) SEG2 (端子番号) 141
- LCDC セグメント(Duty)出力 (端子名) SEG3 (端子番号) 142
- LCDC セグメント(Duty)出力 (端子名) SEG4 (端子番号) 143
- LCDC セグメント(Duty)出力 (端子名) SEG5 (端子番号) 2
- LCDC セグメント(Duty)出力 (端子名) SEG6 (端子番号) 3
- LCDC セグメント(Duty)出力 (端子名) SEG7 (端子番号) 4
- LCDC セグメント(Duty)出力 (端子名) SEG8 (端子番号) 5
- LCDC セグメント(Duty)出力 (端子名) SEG9 (端子番号) 6
- LCDC セグメント(Duty)出力 (端子名) SEG10 (端子番号) 7
- LCDC セグメント(Duty)出力 (端子名) SEG11 (端子番号) 8
- LCDC セグメント(Duty)出力 (端子名) SEG12 (端子番号) 9
- LCDC セグメント(Duty)出力 (端子名) SEG13 (端子番号) 10
- LCDC セグメント(Duty)出力 (端子名) SEG14 (端子番号) 11
- LCDC セグメント(Duty)出力 (端子名) SEG15 (端子番号) 12
- LCDC セグメント(Duty)出力 (端子名) SEG16 (端子番号) 13
- LCDC セグメント(Duty)出力 (端子名) SEG17 (端子番号) 14
- LCDC セグメント(Duty)出力 (端子名) SEG18 (端子番号) 15
- LCDC セグメント(Duty)出力 (端子名) SEG19 (端子番号) 16
- LCDC セグメント(Duty)出力 (端子名) SEG20 (端子番号) 17
- LCDC セグメント(Duty)出力 (端子名) SEG21 (端子番号) 18
- LCDC セグメント(Duty)出力 (端子名) SEG22 (端子番号) 19

- LCDC セグメント(Duty)出力 (端子名) SEG23 (端子番号) 22
- LCDC セグメント(Duty)出力 (端子名) SEG24 (端子番号) 23
- LCDC セグメント(Duty)出力 (端子名) SEG25 (端子番号) 24
- LCDC セグメント(Duty)出力 (端子名) SEG26 (端子番号) 25
- LCDC セグメント(Duty)出力 (端子名) SEG27 (端子番号) 26
- LCDC セグメント(Duty)出力 (端子名) SEG28 (端子番号) 27
- LCDC セグメント(Duty)出力 (端子名) SEG29 (端子番号) 28
- LCDC セグメント(Duty)出力 (端子名) SEG30 (端子番号) 29
- LCDC セグメント(Duty)出力 (端子名) SEG31 (端子番号) 30
- LCDC セグメント(Static)出力 (端子名) ST0 (端子番号) 22
- LCDC セグメント(Static)出力 (端子名) ST1 (端子番号) 23
- LCDC セグメント(Static)出力 (端子名) ST2 (端子番号) 24
- LCDC セグメント(Static)出力 (端子名) ST3 (端子番号) 25
- LCDC セグメント(Static)出力 (端子名) ST4 (端子番号) 26
- LCDC セグメント(Static)出力 (端子名) ST5 (端子番号) 27
- LCDC セグメント(Static)出力 (端子名) ST6 (端子番号) 28
- LCDC セグメント(Static)出力 (端子名) ST7 (端子番号) 29
- LCDC セグメント(Static)出力 (端子名) ST8 (端子番号) 30

1.10.12 LIN-UART の端子 (ch.2～ch.7)

LIN-UARTの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- LIN_UART シリアルクロック入出力 ch.2 リロケーション 0 (端子名) SCK2_0 (端子番号) 101
- LIN_UART シリアルクロック入出力 ch.2 リロケーション 1 (端子名) SCK2_1 (端子番号) 133
- LIN_UART シリアルクロック入出力 ch.3 リロケーション 0 (端子名) SCK3_0 (端子番号) 92
- LIN_UART シリアルクロック入出力 ch.3 リロケーション 1 (端子名) SCK3_1 (端子番号) 136
- LIN_UART シリアルクロック入出力 ch.4 リロケーション 0 (端子名) SCK4_0 (端子番号) 89
- LIN_UART シリアルクロック入出力 ch.4 リロケーション 1 (端子名) SCK4_1 (端子番号) 76
- LIN_UART シリアルクロック入出力 ch.5 リロケーション 0 (端子名) SCK5_0 (端子番号) 86
- LIN_UART シリアルクロック入出力 ch.5 リロケーション 1 (端子名) SCK5_1 (端子番号) 79
- LIN_UART シリアルクロック入出力 ch.6 リロケーション 0 (端子名) SCK6_0 (端子番号) 65
- LIN_UART シリアルクロック入出力 ch.6 リロケーション 1 (端子名) SCK6_1 (端子番号) 13
- LIN_UART シリアルクロック入出力 ch.7 リロケーション 0 (端子名) SCK7_0 (端子番号) 24
- LIN_UART シリアルクロック入出力 ch.7 リロケーション 1 (端子名) SCK7_1 (端子番号) 60
- LIN_UART シリアル入力 ch.2 リロケーション 0 (端子名) SIN2_0 (端子番号) 102
- LIN_UART シリアル入力 ch.2 リロケーション 1 (端子名) SIN2_1 (端子番号) 131
- LIN_UART シリアル入力 ch.3 リロケーション 0 (端子名) SIN3_0 (端子番号) 99
- LIN_UART シリアル入力 ch.3 リロケーション 1 (端子名) SIN3_1 (端子番号) 134
- LIN_UART シリアル入力 ch.4 リロケーション 0 (端子名) SIN4_0 (端子番号) 91
- LIN_UART シリアル入力 ch.4 リロケーション 1 (端子名) SIN4_1 (端子番号) 74
- LIN_UART シリアル入力 ch.5 リロケーション 0 (端子名) SIN5_0 (端子番号) 88
- LIN_UART シリアル入力 ch.5 リロケーション 1 (端子名) SIN5_1 (端子番号) 77
- LIN_UART シリアル入力 ch.6 リロケーション 0 (端子名) SIN6_0 (端子番号) 63
- LIN_UART シリアル入力 ch.6 リロケーション 1 (端子名) SIN6_1 (端子番号) 11
- LIN_UART シリアル入力 ch.7 リロケーション 0 (端子名) SIN7_0 (端子番号) 22
- LIN_UART シリアル入力 ch.7 リロケーション 1 (端子名) SIN7_1 (端子番号) 58
- LIN_UART シリアル出力 ch.2 リロケーション 0 (端子名) SOT2_0 (端子番号) 100

- LIN_UART シリアル出力 ch.2 リロケーション 1 (端子名) SOT2_1 (端子番号) 132
- LIN_UART シリアル出力 ch.3 リロケーション 0 (端子名) SOT3_0 (端子番号) 98
- LIN_UART シリアル出力 ch.3 リロケーション 1 (端子名) SOT3_1 (端子番号) 135
- LIN_UART シリアル出力 ch.4 リロケーション 0 (端子名) SOT4_0 (端子番号) 90
- LIN_UART シリアル出力 ch.4 リロケーション 1 (端子名) SOT4_1 (端子番号) 75
- LIN_UART シリアル出力 ch.5 リロケーション 0 (端子名) SOT5_0 (端子番号) 87
- LIN_UART シリアル出力 ch.5 リロケーション 1 (端子名) SOT5_1 (端子番号) 78
- LIN_UART シリアル出力 ch.6 リロケーション 0 (端子名) SOT6_0 (端子番号) 64
- LIN_UART シリアル出力 ch.6 リロケーション 1 (端子名) SOT6_1 (端子番号) 12
- LIN_UART シリアル出力 ch.7 リロケーション 0 (端子名) SOT7_0 (端子番号) 23
- LIN_UART シリアル出力 ch.7 リロケーション 1 (端子名) SOT7_1 (端子番号) 59

1.10.13 マルチファンクションシリアルインタフェースの端子 (ch.0, ch.1, ch.8, ch.9)

マルチファンクションシリアルインタフェースTの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.0 リロケーション 0
  (端子名) SCK0_0 (端子番号) 123
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.0 リロケーション 1
  (端子名) SCK0_1 (端子番号) 49
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.1 リロケーション 0
  (端子名) SCK1_0 (端子番号) 126
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.1 リロケーション 1
  (端子名) SCK1_1 (端子番号) 46
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.8 リロケーション 0
  (端子名) SCK8_0 (端子番号) 19
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.8 リロケーション 1
  (端子名) SCK8_1 (端子番号) 57
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.8 リロケーション 2
  (端子名) SCK8_2 (端子番号) 30
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.9 リロケーション 0
  (端子名) SCK9_0 (端子番号) 16
- マルチファンクションシリアルインタフェース シリアルクロック入出力 ch.9 リロケーション 1
  (端子名) SCK9_1 (端子番号) 54
- マルチファンクションシリアルインタフェース シリアル入力 ch.0 リロケーション 0
  (端子名) SIN0_0 (端子番号) 121
- マルチファンクションシリアルインタフェース シリアル入力 ch.0 リロケーション 1
  (端子名) SIN0_1 (端子番号) 47
- マルチファンクションシリアルインタフェース シリアル入力 ch.1 リロケーション 0
  (端子名) SIN1_0 (端子番号) 124
- マルチファンクションシリアルインタフェース シリアル入力 ch.1 リロケーション 1
  (端子名) SIN1_1 (端子番号) 44
- マルチファンクションシリアルインタフェース シリアル入力 ch.8 リロケーション 0
  (端子名) SIN8_0 (端子番号) 17

- マルチファンクションシリアルインタフェース シリアル入力 ch.8 リロケーション 1
  (端子名) SIN8_1 (端子番号) 55
- マルチファンクションシリアルインタフェース シリアル入力 ch.8 リロケーション 2
  (端子名) SIN8_2 (端子番号) 28
- マルチファンクションシリアルインタフェース シリアル入力 ch.9 リロケーション 0
  (端子名) SIN9_0 (端子番号) 14
- マルチファンクションシリアルインタフェース シリアル入力 ch.9 リロケーション 1
  (端子名) SIN9_1 (端子番号) 50
- マルチファンクションシリアルインタフェース シリアル出力 ch.0 リロケーション 0
  (端子名) SOT0_0 (端子番号) 122
- マルチファンクションシリアルインタフェース シリアル出力 ch.0 リロケーション 1
  (端子名) SOT0_1 (端子番号) 48
- マルチファンクションシリアルインタフェース シリアル出力 ch.1 リロケーション 0
  (端子名) SOT1_0 (端子番号) 125
- マルチファンクションシリアルインタフェース シリアル出力 ch.1 リロケーション 1
  (端子名) SOT1_1 (端子番号) 45
- マルチファンクションシリアルインタフェース シリアル出力 ch.8 リロケーション 0
  (端子名) SOT8_0 (端子番号) 18
- マルチファンクションシリアルインタフェース シリアル出力 ch.8 リロケーション 1
  (端子名) SOT8_1 (端子番号) 56
- マルチファンクションシリアルインタフェース シリアル出力 ch.8 リロケーション 2
  (端子名) SOT8_2 (端子番号) 29
- マルチファンクションシリアルインタフェース シリアル出力 ch.9 リロケーション 0
  (端子名) SOT9_0 (端子番号) 15
- マルチファンクションシリアルインタフェース シリアル出力 ch.9 リロケーション 1
  (端子名) SOT9_1 (端子番号) 53

1.10.14 アウトプットコンペアの端子 (ch.0～ch.11)

アウトプットコンペアの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- アウトプットコンペア出力 ch.0 リロケーション 0 (端子名) OCU0_0 (端子番号) 86
- アウトプットコンペア出力 ch.0 リロケーション 1 (端子名) OCU0_1 (端子番号) 5
- アウトプットコンペア出力 ch.1 リロケーション 0 (端子名) OCU1_0 (端子番号) 85
- アウトプットコンペア出力 ch.1 リロケーション 1 (端子名) OCU1_1 (端子番号) 6
- アウトプットコンペア出力 ch.2 リロケーション 0 (端子名) OCU2_0 (端子番号) 113
- アウトプットコンペア出力 ch.2 リロケーション 1 (端子名) OCU2_1 (端子番号) 7
- アウトプットコンペア出力 ch.3 リロケーション 0 (端子名) OCU3_0 (端子番号) 120
- アウトプットコンペア出力 ch.3 リロケーション 1 (端子名) OCU3_1 (端子番号) 8
- アウトプットコンペア出力 ch.4 リロケーション 0 (端子名) OCU4_0 (端子番号) 121
- アウトプットコンペア出力 ch.4 リロケーション 1 (端子名) OCU4_1 (端子番号) 18
- アウトプットコンペア出力 ch.5 リロケーション 0 (端子名) OCU5_0 (端子番号) 122
- アウトプットコンペア出力 ch.5 リロケーション 1 (端子名) OCU5_1 (端子番号) 17
- アウトプットコンペア出力 ch.6 リロケーション 0 (端子名) OCU6_0 (端子番号) 16
- アウトプットコンペア出力 ch.6 リロケーション 1 (端子名) OCU6_1 (端子番号) 26
- アウトプットコンペア出力 ch.7 リロケーション 0 (端子名) OCU7_0 (端子番号) 15
- アウトプットコンペア出力 ch.7 リロケーション 1 (端子名) OCU7_1 (端子番号) 27
- アウトプットコンペア出力 ch.8 リロケーション 0 (端子名) OCU8_0 (端子番号) 12

- アウトプットコンペア出力 ch.8 リロケーション 1 (端子名) OCU8_1 (端子番号) 31
- アウトプットコンペア出力 ch.9 リロケーション 0 (端子名) OCU9_0 (端子番号) 11
- アウトプットコンペア出力 ch.9 リロケーション 1 (端子名) OCU9_1 (端子番号) 32
- アウトプットコンペア出力 ch.10 リロケーション 0 (端子名) OCU10_0 (端子番号) 10
- アウトプットコンペア出力 ch.10 リロケーション 1 (端子名) OCU10_1 (端子番号) 33
- アウトプットコンペア出力 ch.11 リロケーション 0 (端子名) OCU11_0 (端子番号) 9
- アウトプットコンペア出力 ch.11 リロケーション 1 (端子名) OCU11_1 (端子番号) 34

1.10.15 PPG の端子 (ch.0～ch.23)

PPGの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- PPG 出力 ch.0 リロケーション 0 (端子名) PPG0_0 (端子番号) 131
- PPG 出力 ch.0 リロケーション 1 (端子名) PPG0_1 (端子番号) 98
- PPG 出力 ch.0 リロケーション 2 (端子名) PPG0_2 (端子番号) 73
- PPG 出力 ch.1 リロケーション 0 (端子名) PPG1_0 (端子番号) 132
- PPG 出力 ch.1 リロケーション 1 (端子名) PPG1_1 (端子番号) 77
- PPG 出力 ch.1 リロケーション 2 (端子名) PPG1_2 (端子番号) 103
- PPG 出力 ch.1 リロケーション 3 (端子名) PPG1_3 (端子番号) 127
- PPG 出力 ch.2 リロケーション 0 (端子名) PPG2_0 (端子番号) 133
- PPG 出力 ch.2 リロケーション 1 (端子名) PPG2_1 (端子番号) 78
- PPG 出力 ch.2 リロケーション 2 (端子名) PPG2_2 (端子番号) 104
- PPG 出力 ch.3 リロケーション 0 (端子名) PPG3_0 (端子番号) 134
- PPG 出力 ch.3 リロケーション 1 (端子名) PPG3_1 (端子番号) 79
- PPG 出力 ch.3 リロケーション 2 (端子名) PPG3_2 (端子番号) 105
- PPG 出力 ch.4 リロケーション 0 (端子名) PPG4_0 (端子番号) 135
- PPG 出力 ch.4 リロケーション 1 (端子名) PPG4_1 (端子番号) 80
- PPG 出力 ch.4 リロケーション 2 (端子名) PPG4_2 (端子番号) 106
- PPG 出力 ch.5 リロケーション 0 (端子名) PPG5_0 (端子番号) 136
- PPG 出力 ch.5 リロケーション 1 (端子名) PPG5_1 (端子番号) 81
- PPG 出力 ch.5 リロケーション 2 (端子名) PPG5_2 (端子番号) 88
- PPG 出力 ch.6 リロケーション 0 (端子名) PPG6_0 (端子番号) 137
- PPG 出力 ch.6 リロケーション 1 (端子名) PPG6_1 (端子番号) 102
- PPG 出力 ch.6 リロケーション 2 (端子名) PPG6_2 (端子番号) 87
- PPG 出力 ch.7 リロケーション 0 (端子名) PPG7_0 (端子番号) 138
- PPG 出力 ch.7 リロケーション 1 (端子名) PPG7_1 (端子番号) 101
- PPG 出力 ch.7 リロケーション 2 (端子名) PPG7_2 (端子番号) 86
- PPG 出力 ch.8 リロケーション 0 (端子名) PPG8_0 (端子番号) 74
- PPG 出力 ch.8 リロケーション 1 (端子名) PPG8_1 (端子番号) 100
- PPG 出力 ch.8 リロケーション 2 (端子名) PPG8_2 (端子番号) 85
- PPG 出力 ch.9 リロケーション 0 (端子名) PPG9_0 (端子番号) 75
- PPG 出力 ch.9 リロケーション 1 (端子名) PPG9_1 (端子番号) 99
- PPG 出力 ch.9 リロケーション 2 (端子名) PPG9_2 (端子番号) 113
- PPG 出力 ch.10 リロケーション 0 (端子名) PPG10_0 (端子番号) 76
- PPG 出力 ch.10 リロケーション 1 (端子名) PPG10_1 (端子番号) 111
- PPG 出力 ch.10 リロケーション 2 (端子名) PPG10_2 (端子番号) 120
- PPG 出力 ch.11 リロケーション 0 (端子名) PPG11_0 (端子番号) 19
- PPG 出力 ch.11 リロケーション 1 (端子名) PPG11_1 (端子番号) 126

- PPG 出力 ch.12 リロケーション 0 (端子名) PPG12_0 (端子番号) 22
- PPG 出力 ch.12 リロケーション 1 (端子名) PPG12_1 (端子番号) 57
- PPG 出力 ch.13 リロケーション 0 (端子名) PPG13_0 (端子番号) 23
- PPG 出力 ch.13 リロケーション 1 (端子名) PPG13_1 (端子番号) 58
- PPG 出力 ch.14 リロケーション 0 (端子名) PPG14_0 (端子番号) 24
- PPG 出力 ch.14 リロケーション 1 (端子名) PPG14_1 (端子番号) 59
- PPG 出力 ch.15 リロケーション 0 (端子名) PPG15_0 (端子番号) 25
- PPG 出力 ch.15 リロケーション 1 (端子名) PPG15_1 (端子番号) 60
- PPG 出力 ch.16 リロケーション 0 (端子名) PPG16_0 (端子番号) 63
- PPG 出力 ch.17 リロケーション 0 (端子名) PPG17_0 (端子番号) 64
- PPG 出力 ch.18 リロケーション 0 (端子名) PPG18_0 (端子番号) 65
- PPG 出力 ch.19 リロケーション 0 (端子名) PPG19_0 (端子番号) 66
- PPG 出力 ch.20 リロケーション 0 (端子名) PPG20_0 (端子番号) 67
- PPG 出力 ch.21 リロケーション 0 (端子名) PPG21_0 (端子番号) 68
- PPG 出力 ch.22 リロケーション 0 (端子名) PPG22_0 (端子番号) 69
- PPG 出力 ch.23 リロケーション 0 (端子名) PPG23_0 (端子番号) 70
- PPG トリガ入力 0 (ch.0-ch.3) (端子名) TRG0 (端子番号) 121
- PPG トリガ入力 1 (ch.4-ch.7) (端子名) TRG1 (端子番号) 124
- PPG トリガ入力 2 (ch.8-ch.11) (端子名) TRG2 (端子番号) 127
- PPG トリガ入力 3 (ch.12-ch.15) (端子名) TRG3 (端子番号) 92
- PPG トリガ入力 4 (ch.16-ch.19) (端子名) TRG4 (端子番号) 89
- PPG トリガ入力 5 (ch.20-ch.23) (端子名) TRG5 (端子番号) 126

1.10.16 リロードタイマの端子(ch.0～ch.3)

リロードタイマの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- リロードタイマイベント入力 ch.0 リロケーション 0 (端子名) TIN0_0 (端子番号) 98
- リロードタイマイベント入力 ch.0 リロケーション 1 (端子名) TIN0_1 (端子番号) 74
- リロードタイマイベント入力 ch.0 リロケーション 2 (端子名) TIN0_2 (端子番号) 131
- リロードタイマイベント入力 ch.1 リロケーション 0 (端子名) TIN1_0 (端子番号) 92
- リロードタイマイベント入力 ch.1 リロケーション 1 (端子名) TIN1_1 (端子番号) 75
- リロードタイマイベント入力 ch.1 リロケーション 2 (端子名) TIN1_2 (端子番号) 132
- リロードタイマイベント入力 ch.2 リロケーション 0 (端子名) TIN2_0 (端子番号) 91
- リロードタイマイベント入力 ch.2 リロケーション 1 (端子名) TIN2_1 (端子番号) 76
- リロードタイマイベント入力 ch.2 リロケーション 2 (端子名) TIN2_2 (端子番号) 133
- リロードタイマイベント入力 ch.3 リロケーション 0 (端子名) TIN3_0 (端子番号) 90
- リロードタイマイベント入力 ch.3 リロケーション 1 (端子名) TIN3_1 (端子番号) 77
- リロードタイマイベント入力 ch.3 リロケーション 2 (端子名) TIN3_2 (端子番号) 134
- リロードタイマ出力 ch.0 リロケーション 0 (端子名) TOT0_0 (端子番号) 89
- リロードタイマ出力 ch.0 リロケーション 1 (端子名) TOT0_1 (端子番号) 78
- リロードタイマ出力 ch.0 リロケーション 2 (端子名) TOT0_2 (端子番号) 135
- リロードタイマ出力 ch.1 リロケーション 0 (端子名) TOT1_0 (端子番号) 88
- リロードタイマ出力 ch.1 リロケーション 1 (端子名) TOT1_1 (端子番号) 79
- リロードタイマ出力 ch.1 リロケーション 2 (端子名) TOT1_2 (端子番号) 136
- リロードタイマ出力 ch.2 リロケーション 0 (端子名) TOT2_0 (端子番号) 87
- リロードタイマ出力 ch.2 リロケーション 1 (端子名) TOT2_1 (端子番号) 102
- リロードタイマ出力 ch.2 リロケーション 2 (端子名) TOT2_2 (端子番号) 137

- リロードタイマ出力 ch.3 リロケーション 0 (端子名) TOT3_0 (端子番号) 86
- リロードタイマ出力 ch.3 リロケーション 1 (端子名) TOT3_1 (端子番号) 101
- リロードタイマ出力 ch.3 リロケーション 2 (端子名) TOT3_2 (端子番号) 138

1.10.17 リアルタイムクロックの端子

リアルタイムクロックの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- RTC オーバフロー出力 (端子名) WOT (端子番号) 98

1.10.18 ステッピングモータ制御の端子 (ch.0～ch.5)

ステッピングモータ制御の端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- SMC 出力 ch.0 (端子名) PWM1M0 (端子番号) 44
- SMC 出力 ch.0 (端子名) PWM1P0 (端子番号) 43
- SMC 出力 ch.0 (端子名) PWM2M0 (端子番号) 46
- SMC 出力 ch.0 (端子名) PWM2P0 (端子番号) 45
- SMC 出力 ch.1 (端子名) PWM1M1 (端子番号) 48
- SMC 出力 ch.1 (端子名) PWM1P1 (端子番号) 47
- SMC 出力 ch.1 (端子名) PWM2M1 (端子番号) 50
- SMC 出力 ch.1 (端子名) PWM2P1 (端子番号) 49
- SMC 出力 ch.2 (端子名) PWM1M2 (端子番号) 54
- SMC 出力 ch.2 (端子名) PWM1P2 (端子番号) 53
- SMC 出力 ch.2 (端子名) PWM2M2 (端子番号) 56
- SMC 出力 ch.2 (端子名) PWM2P2 (端子番号) 55
- SMC 出力 ch.3 (端子名) PWM1M3 (端子番号) 58
- SMC 出力 ch.3 (端子名) PWM1P3 (端子番号) 57
- SMC 出力 ch.3 (端子名) PWM2M3 (端子番号) 60
- SMC 出力 ch.3 (端子名) PWM2P3 (端子番号) 59
- SMC 出力 ch.4 (端子名) PWM1M4 (端子番号) 64
- SMC 出力 ch.4 (端子名) PWM1P4 (端子番号) 63
- SMC 出力 ch.4 (端子名) PWM2M4 (端子番号) 66
- SMC 出力 ch.4 (端子名) PWM2P4 (端子番号) 65
- SMC 出力 ch.5 (端子名) PWM1M5 (端子番号) 68
- SMC 出力 ch.5 (端子名) PWM1P5 (端子番号) 67
- SMC 出力 ch.5 (端子名) PWM2M5 (端子番号) 70
- SMC 出力 ch.5 (端子名) PWM2P5 (端子番号) 69

1.10.19 サウンドジェネレータの端子 (ch.0～ch.4)

サウンドジェネレータの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- サウンドジェネレータ SGA 出力 ch.0 (端子名) SGA0 (端子番号) 102
- サウンドジェネレータ SGA 出力 ch.1 (端子名) SGA1 (端子番号) 100
- サウンドジェネレータ SGA 出力 ch.2 (端子名) SGA2 (端子番号) 92
- サウンドジェネレータ SGA 出力 ch.3 (端子名) SGA3 (端子番号) 90
- サウンドジェネレータ SGA 出力 ch.4 リロケーション 0 (端子名) SGA4_0 (端子番号) 26
- サウンドジェネレータ SGA 出力 ch.4 リロケーション 1 (端子名) SGA4_1 (端子番号) 80
- サウンドジェネレータ SGO 出力 ch.0 (端子名) SGO0 (端子番号) 101
- サウンドジェネレータ SGO 出力 ch.1 (端子名) SGO1 (端子番号) 99
- サウンドジェネレータ SGO 出力 ch.2 (端子名) SGO2 (端子番号) 91
- サウンドジェネレータ SGO 出力 ch.3 (端子名) SGO3 (端子番号) 89
- サウンドジェネレータ SGO 出力 ch.4 リロケーション 0 (端子名) SGO4_0 (端子番号) 27
- サウンドジェネレータ SGO 出力 ch.4 リロケーション 1 (端子名) SGO4_1 (端子番号) 81

1.10.20 アップダウンカウンタの端子 (ch.0, ch.1)

アップダウンカウンタの端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- アップダウンカウンタ AIN 入力 ch.0 リロケーション 0 (端子名) AIN0_0 (端子番号) 25
- アップダウンカウンタ AIN 入力 ch.0 リロケーション 1 (端子名) AIN0_1 (端子番号) 50
- アップダウンカウンタ AIN 入力 ch.0 リロケーション 2 (端子名) AIN0_2 (端子番号) 63
- アップダウンカウンタ AIN 入力 ch.1 リロケーション 0 (端子名) AIN1_0 (端子番号) 28
- アップダウンカウンタ AIN 入力 ch.1 リロケーション 1 (端子名) AIN1_1 (端子番号) 47
- アップダウンカウンタ BIN 入力 ch.0 リロケーション 0 (端子名) BIN0_0 (端子番号) 26
- アップダウンカウンタ BIN 入力 ch.0 リロケーション 1 (端子名) BIN0_1 (端子番号) 49
- アップダウンカウンタ BIN 入力 ch.0 リロケーション 2 (端子名) BIN0_2 (端子番号) 64
- アップダウンカウンタ BIN 入力 t ch.1 リロケーション 0 (端子名) BIN1_0 (端子番号) 29
- アップダウンカウンタ BIN 入力 ch.1 リロケーション 1 (端子名) BIN1_1 (端子番号) 46
- アップダウンカウンタ ZIN 入力 ch.0 リロケーション 0 (端子名) ZIN0_0 (端子番号) 27
- アップダウンカウンタ ZIN 入力 ch.0 リロケーション 1 (端子名) ZIN0_1 (端子番号) 48
- アップダウンカウンタ ZIN 入力 ch.0 リロケーション 2 (端子名) ZIN0_2 (端子番号) 65
- アップダウンカウンタ ZIN 入力 ch.1 リロケーション 0 (端子名) ZIN1_0 (端子番号) 30
- アップダウンカウンタ ZIN 入力 ch.1 リロケーション 1 (端子名) ZIN1_1 (端子番号) 45

1.10.21 クロック端子

クロック端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- メインクロック OSC 入力 (端子名) X0 (端子番号) 117
- メインクロック OSC 出力 (端子名) X1 (端子番号) 118

- サブクロック OSC 入力 (クロック 2 系統品) (端子名) (X0A) (端子番号) 95
- サブクロック OSC 出力 (クロック 2 系統品) (端子名) (X1A) (端子番号) 96

1.10.22 モード端子

モード端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- モード端子 (端子名) MD0 (端子番号) 114
- モード端子 (端子名) MD1 (端子番号) 115
- モード端子 (端子名) MD2 (端子番号) 116

1.10.23 リセット端子

リセット端子について示します。

端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

- リセット端子 (端子名) RSTX (端子番号) 107
- DEBUG I/F (端子名) DEBUGIF (端子番号) 110
- NMI 端子 (端子名)NMIX (端子番号) 97

1.10.24 電源端子

電源端子について示します。

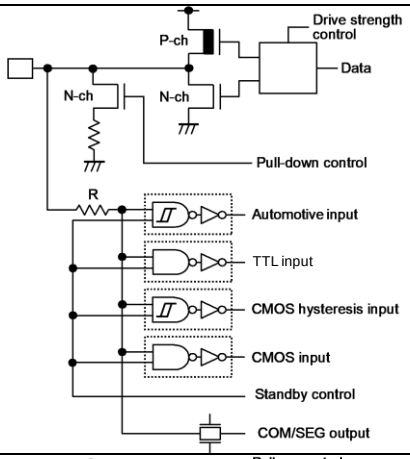
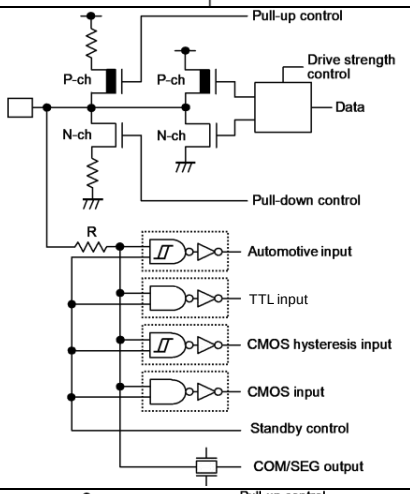
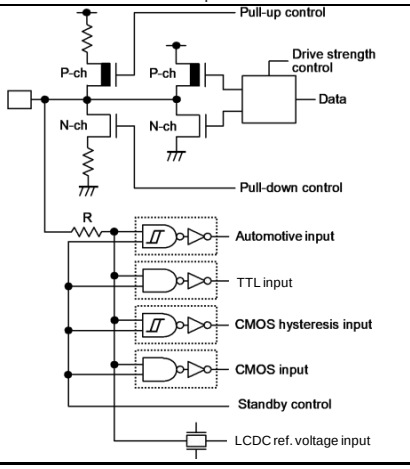
端子番号はLQFP-144のものを示します。LQFP-208の端子番号は端子配列図またはデータシートの端子機能説明を参照してください。

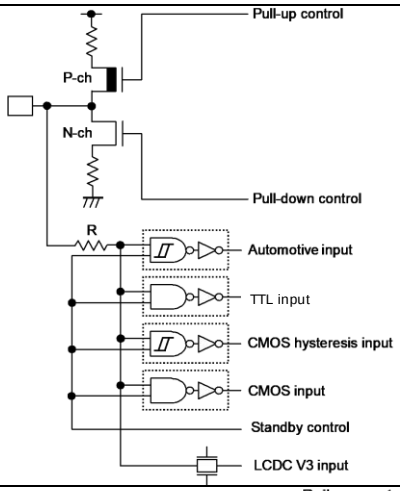
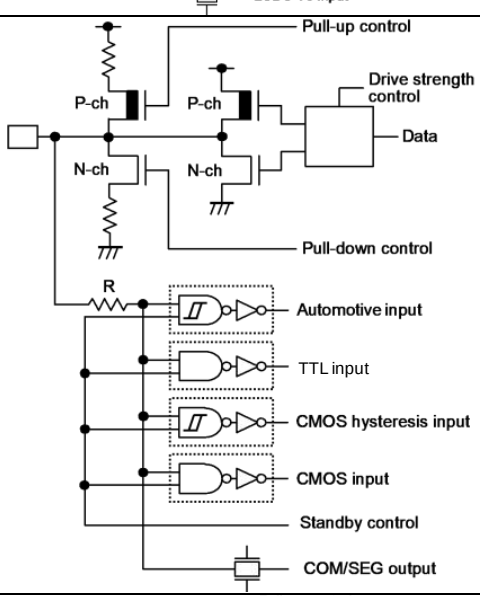
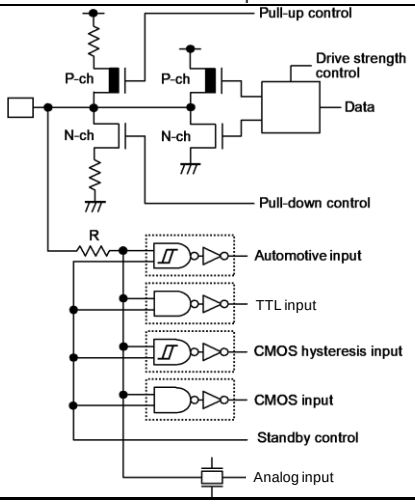
- +3.3v/+5.0v 電源 (端子名) VCCE (端子番号) 1
- +3.3v/+5.0v 電源 (端子名) VCCE (端子番号) 20
- +5.0v 電源 (端子名) VCC5 (端子番号) 108
- +5.0v 電源 (端子名) VCC5 (端子番号) 128
- +5.0v 電源 (端子名) VCC5 (端子番号) 36
- +5.0v 電源 (端子名) VCC5 (端子番号) 93
- ADC, DAC アナログ電源 (端子名) AVCC (端子番号) 84
- ADC, DAC GND / 下限基準電圧 (端子名) AVSS/AVRL (端子番号) 82
- ADC 上限基準電圧 (端子名) AVRH (端子番号) 83
- 外部容量接続端子 (端子名) C (端子番号) 130
- GND (端子名) VSS (端子番号) 109, 119, 129, 144, 21, 37, 94
- SMC 大電流用 GND (端子名) DVSS (端子番号) 42, 52, 62, 72
- SMC 大電流用電源 (端子名) DVCC (端子番号) 41, 51, 61, 71

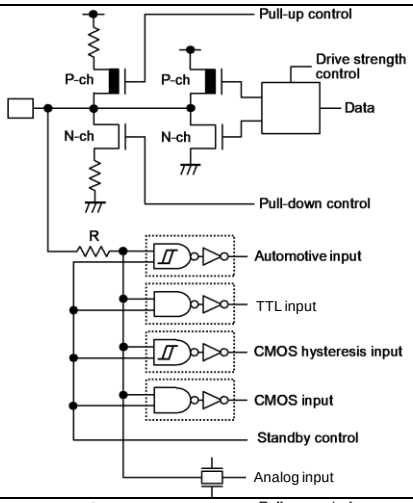
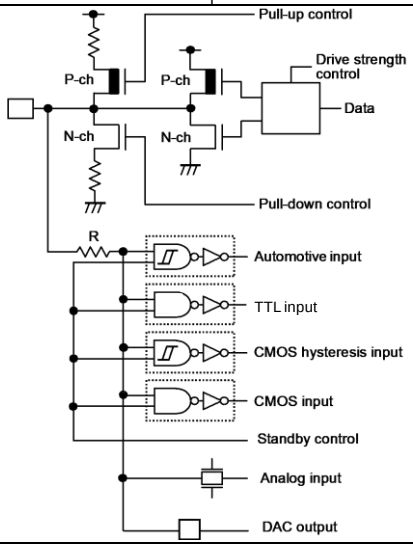
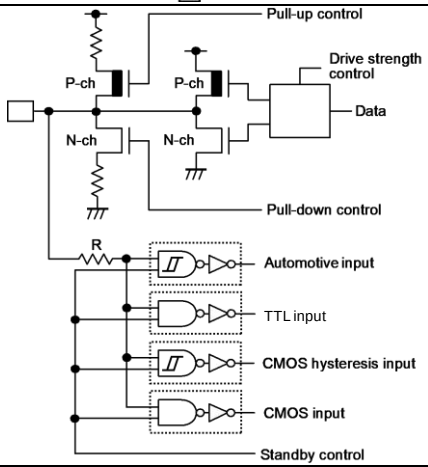
1.11 入出力回路形式

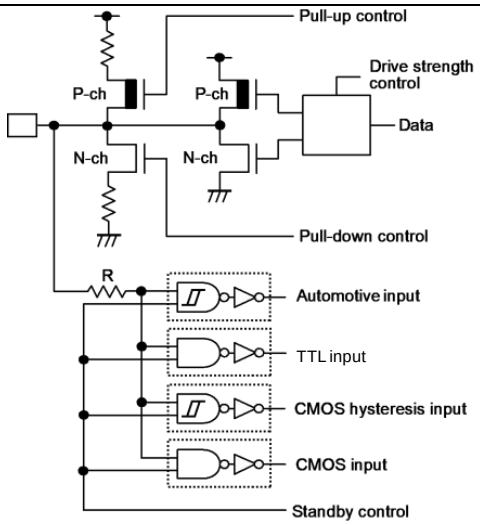
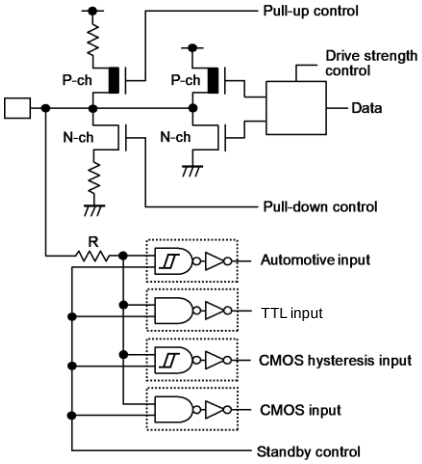
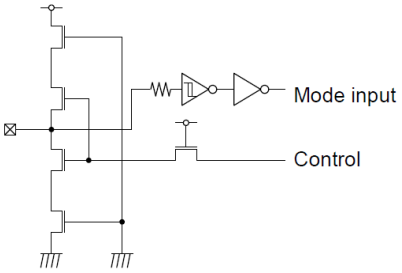
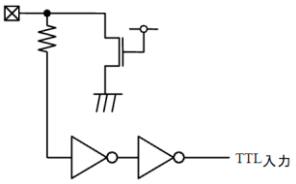
入出力回路形式について示します。

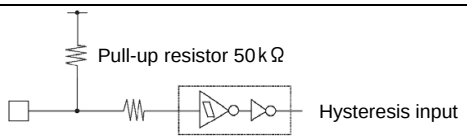
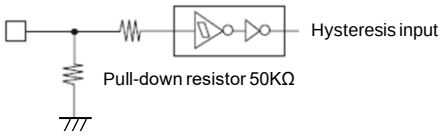
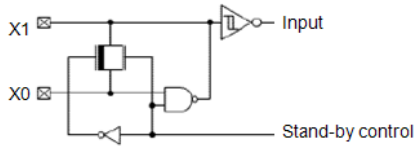
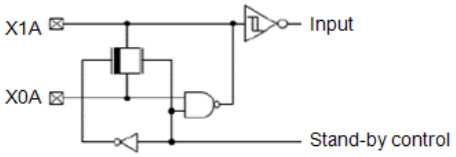
表 1-4 入出力回路形式

分類	回路	備考
H		COM/SEG 出力付き、3V 電源対応(5V トレラント)汎用入出力ポート I_{OH} = -1/-2mA(@VCCE=5V), I_{OH} = -0.5/-1/-2mA(@VCCE=3.3V), I_{OL} = 1/2mA(@VCCE=5V), I_{OL} = 0.5/1/2mA(@VCCE=3.3V) プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
I		COM/SEG 出力付き汎用入出力ポート I_{OH} = -1/-2mA, I_{OL} = 1/2mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
I2		LCDC 基準電圧入力付き汎用入出力ポート I_{OH} = -1/-2mA, I_{OL} = 1/2mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力

分類	回路	備考
I3		LCDC V3 入力付き汎用入力ポート プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
I4		COM/SEG 出力機能付き汎用入出力ポート IOH = -1mA,　IOL = 1mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
J		アナログ入力付き汎用入出力ポート I_{OH} = -1/-2mA,　I_{OL} = 1/2mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力

分類	回路	備考
K		アナログ入力, SMC の大電流付き汎用入出力ポート I_{OH} = -1/-2/-30mA,　I_{OL} = 1/2/30mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
L		アナログ入力,DAC 出力付き汎用入出力ポート I_{OH} = -1/-2mA,　I_{OL} = 1/2mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
M		汎用入出力ポート I_{OH} = -1/-2mA,　I_{OL} = 1/2mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力

分類	回路	備考
M2		汎用入出力ポート IOH = -1mA,　IOL = 1mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
N		I²C 出力付き汎用入出力ポート I_{OH} = -1/-2/-3mA,　I_{OL} = 1/2/3mA プルアップ抵抗制御付き プルダウン抵抗制御付き Automotive レベル入力 TTL レベル入力 CMOS レベルヒステリシス入力 CMOS レベル入力
A		モード端子
B		DEBUG I/F 端子

分類	回路	備考
R		CMOS レベルヒステリシス入力
R2		CMOS レベルヒステリシス入力
X		メインクロック
Y		サブクロック

2. デバイスの取扱い

本シリーズの使用上の注意について説明します。

2.1 取扱上のご注意

半導体デバイスは、ある確率で故障します。また、半導体デバイスの故障は、使用される条件 (回路条件, 環境条件など) によっても大きく左右されます。

以下に、半導体デバイスをより信頼性の高い状態で使用していただくために、注意・配慮しなければならない事項について説明します。

■ 設計上の注意事項

ここでは、半導体デバイスを使用して電子機器の設計を行う際に注意すべき事項について述べます。

● 絶対最大定格の遵守

半導体デバイスは、過剰なストレス (電圧, 電流, 温度など) が加わると破壊する可能性があります。この限界値を定めたものが絶対最大定格です。従って、定格を一項目でも超えることのないようご注意ください。

● 推奨動作条件の遵守

推奨動作条件は、半導体デバイスの正常な動作を保証する条件です。電気的特性の規格値は、全てこの条件の範囲内で保証されます。常に推奨動作条件下で使用してください。この条件を越えて使用すると、信頼性に悪影響を及ぼすことがあります。
本資料に記載されていない項目, 使用条件, 論理組み合わせでの使用は、保証していません。記載されている以外の条件での使用をお考えの場合は、必ず事前に営業部門までご相談ください。

● 端子の処理と保護

半導体デバイスには、電源および各種入出力端子があります。これらに対して以下の注意が必要です。

1. 過電圧・過電流の防止
   各端子に最大定格を超える電圧・電流が印加されると、デバイスの内部に劣化が生じ、著しい場合には破壊に至ります。機器の設計の際には、このような過電圧・過電流の発生を防止してください。
2. 出力端子の保護
   出力端子を電源端子または他の出力端子とショートしたり、大きな容量負荷を接続すると大電流が流れる場合があります。この状態が長時間続くとデバイスが劣化しますので、このような接続はしないようにしてください。
3. 未使用入力端子の処理
   インピーダンスの非常に高い入力端子は、オープン状態で使用すると動作が不安定になる場合があります。適切な抵抗を介して電源端子やグランド端子に接続してください。

● 供給電圧の安定化

半導体デバイスは、基板上にP 型とN 型の領域を形成することにより構成されます。外部から異常な電圧が加えられた場合、内部の寄生PNPN 接合 (サイリスタ構造) が導通して、数百mA を越える大電流が電源端子に流れ続けることがあります。これをラッチアップとよびます。この現象が起きるとデバイスの信頼性を損ねるだけでなく、破壊に至り発熱・発煙・発火の恐れもあります。これを防止するために、以下の点にご注意ください。

1. 最大定格以上の電圧が端子に加わることが無いようにしてください。異常なノイズ, サージ等にも注意してください。
2. 電源投入シーケンスを考慮し、異常な電流が流れないようにしてください。

● 安全等の規制と規格の遵守

世界各国では､安全や, 電磁妨害等の各種規制と規格が設けられています｡お客様が機器を設計するに際しては、これらの規制と規格に適合するようお願いします。

● フェイル・セーフ設計

半導体デバイスは、ある確率で故障が発生します。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, 過電流防止設計, 誤動作防止設計などの安全設計をお願いします。

● 用途に関する注意

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。当社は、これらの用途に当該製品が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。

■ パッケージ実装上の注意事項

パッケージには、リード挿入形と表面実装形があります。いずれの場合も、はんだ付け時の耐熱性に関する品質保証は、当社の推奨する条件での実装に対してのみ適用されます。実装条件の詳細については営業部門までお問い合わせください。

● リード挿入形

リード挿入形パッケージのプリント板への実装方法は、プリント板へ直接はんだ付けする方法とソケットを使用してプリント板に実装する方法とがあります。
プリント板へ直接はんだ付けする場合は､プリント板のスルーホールにリード挿入後､噴流はんだによるフローはんだ方法 (ウェーブソルダリング法) が一般的に使用されます。この場合、はんだ付け実装時には、通常最大定格の保存温度を上回る熱ストレスがリード部分に加わります。当社の実装推奨条件で実装してください。
ソケット実装方法でご使用になる場合、ソケットの接点の表面処理とIC のリードの表面処理が異なるとき、長時間経過後、接触不良を起こすことがあります。このため、ソケットの接点の表面処理とIC のリードの表面処理の状態を確認してから実装することをお勧めします。

● 表面実装形

表面実装形パッケージは、リード挿入形と比較して、リードが細く薄いため、リードが変形し易い性質をもっています。
また、パッケージの多ピン化に伴い、リードピッチも狭く、リード変形によるオープン不良や, はんだブリッジによるショート不良が発生しやすいため、適切な実装技術が必要となります。
当社ははんだリフロー方法を推奨し、製品ごとに実装条件のランク分類を実施しています。当社推奨のランク分類に従って実装してください。

● 鉛フリーパッケージ

BGA パッケージのSn-Ag-Cu 系ボール品をSn-Pb 共晶はんだにて実装した場合、使用状況により接合強度が低下することがありますのでご注意願います。

● 半導体デバイスの保管について

プラスチックパッケージは樹脂でできているため、自然の環境に放置することにより吸湿します。吸湿したパッケージに実装時の熱が加わった場合、界面剥離発生による耐湿性の低下やパッケージクラックが発生することがあります。以下の点にご注意ください。

1. 急激な温度変化のある所では製品に水分の結露が起こります。このような環境を避けて、温度変化の少ない場所に保管してください。
2. 製品の保管場所はドライボックスの使用を推奨します。相対湿度 70 %RH 以下, 温度 5℃ ～ 30℃ で保管をお願いします。
   ドライパッケージを開封した場合には湿度 40% ～ 70%RH を推奨いたします。
3. 当社では必要に応じて半導体デバイスの梱包材として防湿性の高いアルミラミネート袋を用い、乾燥剤としてシリカゲルを使用しております。半導体デバイスはアルミラミネート袋に入れて密封して保管してください。
4. 腐食性ガスの発生する場所や塵埃の多い所は避けてください。

● ベーキングについて

吸湿したパッケージはベーキング (加熱乾燥) を実施することにより除湿することが可能です。ベーキングは、当社の推奨する条件で実施してください。
条件：125℃/24 時間

● 静電気

半導体デバイスは静電気による破壊を起こしやすいため、以下の点についてご注意ください。

1. 作業環境の相対湿度は 40 % ～ 70 %RH にしてください。
   除電装置 (イオン発生装置) の使用なども必要に応じて検討してください。
2. 使用するコンベア, 半田槽, 半田ゴテ, および周辺付帯設備は大地に接地してください。
3. 人体の帯電防止のため、指輪または腕輪などから高抵抗 (1 MΩ 程度) で大地に接地したり、 導電性の衣服・靴を着用し、床に導電マットを敷くなど帯電電荷を最小限に保つようにしてください。
4. 治具, 計器類は、接地または帯電防止化を実施してください。
5. 組立完了基板の収納時、発泡スチロールなどの帯電し易い材料の使用は避けてください。

■ 使用環境に関する注意事項

半導体デバイスの信頼性は、先に述べました周囲温度とそれ以外の環境条件にも依存します。ご使用にあたっては、以下の点にご注意ください。

1. 湿度環境
   高湿度環境下での長期の使用は、デバイス自身だけでなくプリント基板等にもリーク性の不具合が発生する場合があります。高湿度が想定される場合は、防湿処理を施す等の配慮をお願いします。
2. 静電気放電
   半導体デバイスの直近に高電圧に帯電したものが存在すると、放電が発生し誤動作の原因となることがあります。
   このような場合、帯電の防止または放電の防止の処置をお願いします。
3. 腐食性ガス，塵埃，油
   腐食性ガス雰囲気中や，塵埃，油等がデバイスに付着した状態で使用すると、化学反応によりデバイスに悪影響を及ぼす場合があります。このような環境下でご使用の場合は、防止策についてご検討ください。
4. 放射線・宇宙線
   一般のデバイスは、設計上、放射線，宇宙線にさらされる環境を想定しておりません。したがって、これらを遮蔽してご使用ください。
5. 発煙・発火
   樹脂モールド型のデバイスは、不燃性ではありません。発火物の近くでは、ご使用にならないでください。発煙・発火しますと、その際に毒性を持ったガスが発生する恐れがあります。

その他，特殊な環境下でのご使用をお考えの場合は、営業部門にご相談ください。

2.2 デバイス取り扱いについて

デバイス取り扱いについて説明します。

■ デバイス取扱い上の注意

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VCCより高い電圧やVSSより低い電圧を印加した場合またはVCC～VSS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、アナログ系の電源, 高電流出力バッファ端子用電源, 5/3V電源投入時、および切断時においてもアナログ電源(AVcc, AVRH)とアナログ入力および高電流出力バッファ端子用電源 (DVcc), 5/3V電源 (VccE) は、デジタル電源 (Vcc5)を超えないように注意してください。

電源投入の順序は､デジタル電源 (Vcc5), アナログ電源(AVcc, AVRH), 高電流出力バッファ端子用電源 (DVcc), 5/3V電源 (VccE)を同時に投入するか、デジタル電源(Vcc5)を投入後、アナログ電源(AVcc, AVRH)および高電流出力バッファ端子用電源(DVcc)、5/3V電源(VccE)を投入してください。

● 未使用端子の処理について

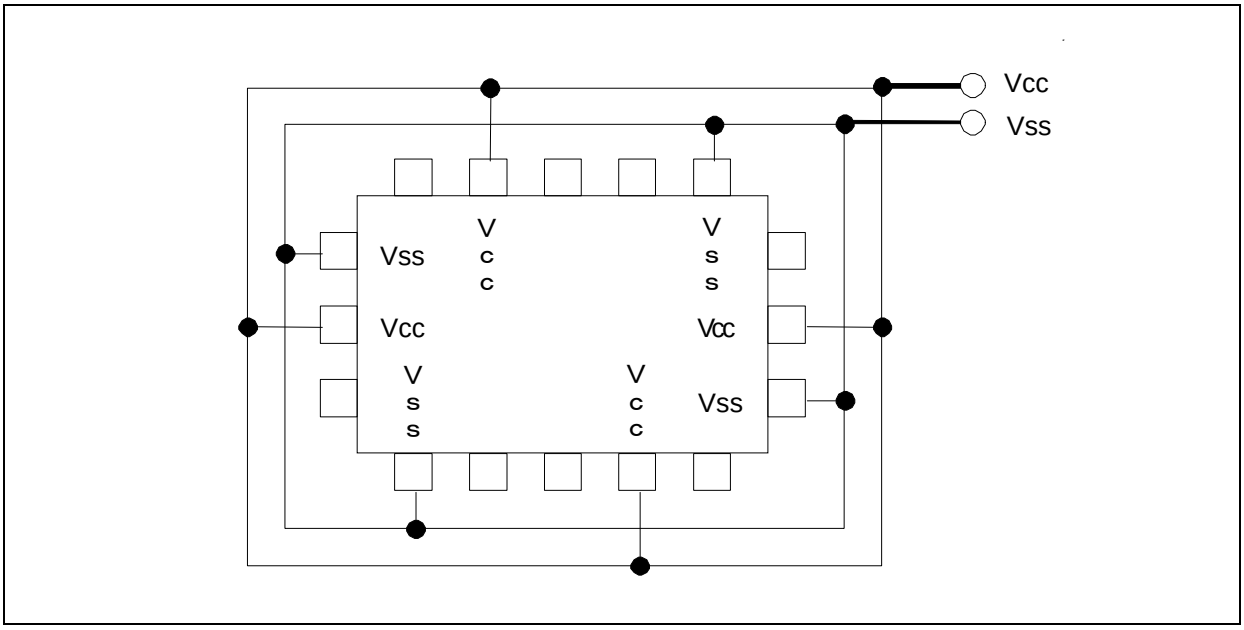
使用していない入力端子を開放のままにしておくと、誤動作およびラッチアップによる永久破壊の原因になることがありますので、2kΩ以上の抵抗を介して、プルアップまたはプルダウンなどの処置をしてください。

また、使用していない入出力端子がある場合は、出力状態に設定して解放とするか、入力状態に設定して入力端子と同じ処理をしてください。

● 電源端子について

VCC・VSS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るべきものどうしを接続してありますが、不要輻射の低減・グランドレベルの上昇によるストローブ信号の誤動作の防止・総出力電流規格の遵守などのため、必ずそれらすべてを外部で電源およびグランドに接続してください。図 2-1 に示すように、すべてのVss電源端子も同様に扱ってください。複数のVccまたはVssシステムがある場合、デバイスは保証された動作範囲内でも正しく動作しません。

図 2-1 電源入力端子



また、電源供給源から低インピーダンスで本デバイスのVCC, VSSに接続するような配慮をお願いいたします。

本デバイスの近くで､VCCとVSSの間にC端子のコンデンサより値の大きなセラミックコンデンサをバイパスコンデンサとして接続することをお勧めいたします。

● 水晶発振回路について

X0, X1端子へのノイズは誤動作のもととなります。X0とX1および水晶発振子 (あるいはセラミック発振子) さらにグランドへのバイパスコンデンサはデバイスの直近に配置するようにプリント板を設計してください。

X0, X1端子の周りをグランドで囲むようなプリント板アートワークを推奨いたします。

● モード端子 (MD2, MD1, MD0) について

モード端子MD2, MD1, MD0は、VCCまたはVSSに直接つないで使用してください。ノイズによって誤ってテストモードに入っ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VCCまたはVSS間のパターン長を短くし、低インピーダンスで接続するようにしてください。

● 電源投入時について

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

● PLL クロック動作中の注意について

PLLクロックを選択しているときに発振子が外れたり、あるいは入力が停止したりした場合、PLL内部の自励発振回路の自走周波数で動作を継続し続ける場合があります。この動作は保証範囲外の動作です。

● A/D コンバータの電源端子処理

A/Dコンバータを使用しない場合においてもAVcc=AVRH=Vcc5, AVss/AVRL=Vssとなるよう接続してください。

● 外部クロック使用時の注意について

外部クロックは非対応です。

外部ダイレクトクロック入力はメインクロック，サブクロックとも使用できません。

● A/D コンバータの電源アナログ入力の投入順序

A/Dコンバータの電源 (AVcc, AVRH, AVRL) およびアナログ入力 (AN0～AN39) の印加は、必ずデジタル電源 (Vcc) の投入後に行ってください。また、電源切断時はA/Dコンバータの電源およびアナログ入力切断の後で、デジタル電源 (Vcc5) の切断を行ってください。その際、AVRHはAVccを超えないように投入、切断を行ってください。アナログ入力と兼用している端子を入力ポートとして使用する場合においても、入力電圧はAVccを超えないようにしてください (アナログ電源とデジタル電源を同時に投入･切断をすることは問題ありません)。

● 高電流出力バッファ端子用電源 (DVcc, DVss) の取扱い

高電流出力バッファ端子用電源 (DVcc, DVss) の印加は､必ずデジタル電源 (Vcc) の投入後に行ってください。また､電源切断時は高電流出力バッファ端子用電源切断の後で､デジタル電源 (Vcc) の切断を行ってください。

高電流出力バッファ端子を汎用ポートとして使用する場合も､高電流出力バッファ端子用電源 (DVcc, DVss) の印加を行ってください (高電流出力バッファ端子用電源とデジタル電源を同時に投入・切断をすることは問題ありません)。

● C 端子の処理について

本デバイスは降圧回路を内蔵します。C端子にはデバイス内部安定化のため、必ずコンデンサを接続してください。規格値については、最新データシートの「推奨動作条件」を参照してください。

＜注意事項＞

動作電圧の詳細仕様については、最新のデータシートを参照してください。

2.3 使用上の注意事項

使用上の注意事項について説明します。

2.3.1. 兼用ポートの機能切り替え

2.3.2. 低消費電力モード

2.3.3. ステータスフラグを含むレジスタに書込みを行う場合の注意

2.3.1 兼用ポートの機能切り替え

兼用ポートの機能切り替えについて示します。

ポートと兼用端子の切替えは、PFR (ポートファンクションレジスタ) で行います。ただし、外バス兼用端子は外バスの設定により切り替わります。詳細は『I/Oポート』の章を参照してください。

2.3.2 低消費電力モード

低消費電力モードについて説明します。

スリープモード・時計モード・ストップモードまたは時計モード (電源遮断)・ストップモード (電源遮断) に入れる場合は、『CHAPTER 27: 消費電力制御』の章の『スリープモード・時計モード・ストップモードの起動』または『時計モード (電源遮断)・ストップモード (電源遮断) の起動』で説明している手順を実行してください。

モニタデバッガを使用する場合は、以下のことを行わないでください。

- 低消費電力移行プログラムに対する、ブレークポイントの設定
- 低消費電力移行プログラムに対する、ステップ実行

2.3.3 ステータスフラグを含むレジスタに書込みを行う場合の注意

ステータスフラグを含むレジスタに書込みを行う場合の注意について説明します。

ステータスフラグ (特に割込み要求フラグなど) を含むレジスタに機能の制御のために書込みを行う場合には、ステータスフラグを誤ってクリアしないように配慮することが大切です。

つまり、書込み時には、ステータスビットに対してフラグをクリアしないよう設定した上で、制御ビットを希望した値に設定してください。

特に複数ビットで構成している制御ビットの場合、ビット命令が使えないため (ビット命令は1ビットアクセスのみ)、バイト/ハーフワード/ワードアクセスによって、同時に制御ビットとステータスフラグへの書込みを行うことになります。しかし、このときに目的以外のビット (この場合ステータスフラグのビット) を誤ってクリアしないよう注意してください。

＜注意事項＞

RMWをサポートしているレジスタに対して、ビット命令は、この点を配慮していますので、注意の必要はありません。RMWをサポートしていないレジスタに対してビット命令を使う場合には注意が必要です。

3. CPU

CPUについて説明します。

3.1 概要

CPUの概要について説明します。

FR81 アーキテクチャは、FRファミリ命令セットを採用し、浮動小数点機能，メモリ保護機能，オンチップデバッグシステムを搭載しています。

整数系命令セットは、FR80と互換です。

詳細は「FR Family FR81 32ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。

3.2 特長

CPUの特長について説明します。

FRファミリは、32ビットRISCベースのコントローラ向けCPUコアです。特に高速制御を行う必要がある組込み制御向けマイクロコントローラのCPUコアとして最適なアーキテクチャとなっています。

■ 全般

- 汎用レジスタアーキテクチャ(32 ビット×16 本)
- 32 ビットアドレス空間(4GB)
- 16 ビット固定命令長(即値データ転送命令除く)
- 5 段パイプライン構造による基本命令 1 命令 1 サイクルの高速処理
- 32 ビット×32 ビットの演算を 5 サイクルで完了する乗算命令
- ステップ除算命令による 32 ビット÷32 ビットの除算実行
- 周辺アクセスのためのダイレクトアドレッシング命令
- 6 サイクルで完了する高速割込み処理
- 単精度浮動小数点演算命令
- 浮動小数点レジスタ 32bit ×16 本
- 特権モード・ユーザモード
  - 一部のアドレスマップされたレジスタをシステムレジスタとしてユーザモード時のアクセスから保護
  - 一部の命令を特権命令としてユーザモードから保護
- FPU，命令アクセス、データアクセス系例外機能
  - FPU 例外
  - 命令アクセス保護違反例外
  - データアクセス保護違反例外
  - 不正命令例外 (未定義命令例外からの変更)
  - データアクセスエラー例外
  - FPU 不在例外

■ メモリ保護機能 (MPU)

- 命令・データ共用で 8 個の保護領域指定
- 保護領域は固定優先順位で判定。(領域の重複可能)
- ページアドレスとページサイズで領域指定
  - ☐ ページサイズ: 16 バイトから 2^n で指定可能
  - ☐ ページアドレス: ミスアラインもサポート
- 特権モード・ユーザモードそれぞれで以下のアクセス権を制御
  - ☐ 命令フェッチ(実行)許可/禁止
  - ☐ リード許可/禁止
  - ☐ ライト許可/禁止
- 領域ごとに以下の属性を指定可能
  - ☐ Bufferable/Non-Bufferable
- 未設定領域に対してもアクセス権, 属性指定可能
- 保護違反時は、命令アクセス保護違反例外/データアクセス保護違反例外を発生

■ 浮動小数点演算

- IEEE754 準拠
- 単精度をサポート
- 6 個の例外要因をサポート
  - ☐ アンダフロー
  - ☐ オーバフロー
  - ☐ ゼロ除算
  - ☐ 無効演算
  - ☐ 不正確
  - ☐ 非正規化数の入力
- 丸めモードは、最近値のみをサポート
- 非正規化数は 0 への切捨てもしくは例外発生
- 浮動小数点レジスタ 32 ビット×16 本
- Multiply and Add, Multiply and Sub 命令サポート
- 除算, 開平演算サポート

3.3 CPU 動作記述

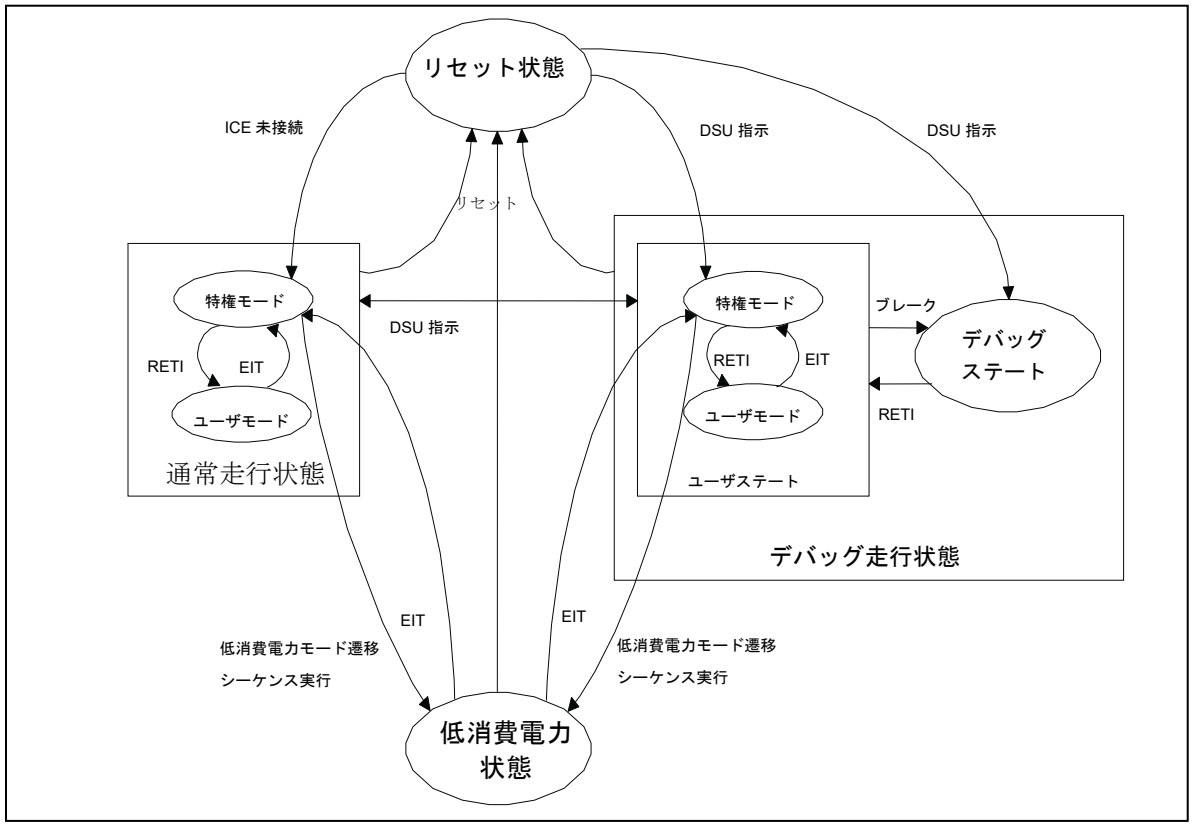
CPUの動作について説明します。

■ CPU 動作状態

CPUの動作状態には、リセット状態/通常走行状態/低消費電力状態/デバッグ走行状態の4つの状態があります。

以下に動作状態の遷移を示します。

図 3-1 CPU 動作状態遷移図



● リセット状態

CPUがリセットされている状態です。リセットにはイニシャライズレベルとリセットレベルの2レベルがあります。イニシャライズレベルのリセットが発行されるとチップ内のすべてが初期化されます。リセットレベルの場合は、デバッグ制御機能やクロック，リセット制御機能の一部を除き初期化されます。

● 通常走行状態

順次命令実行およびEIT処理が実行されている状態です。通常走行状態には特権モードとユーザモードがあります。

ユーザモードでは、命令やアクセス先に制限があり、特権モードでのみ実行可能な命令やアクセス先があります。リセット解除後通常走行状態になった場合は特権モードになりRETIの実行により、ユーザモードへ遷移します。通常走行状態でのユーザモードから特権モードへはリセット, EITの実行で遷移し、特権モードからユーザモードへの遷移はRETIの実行で行います。

● 低消費電力状態

CPUが動作を停止し、消費電力を抑えている状態です。低消費電力状態への移行はクロック制御部のスタンバイ制御で行います。低消費電力状態にはスリープ/ストップ/時計モードの3つのモードがあります。低消費電力状態からの復帰は割込みで行います。

● デバッグ走行状態

ICEを接続し、デバッグ関連機能が有効な状態です。デバッグ走行状態にはユーザステートとデバッグステートがあります。デバッグ走行状態とほかの状態への遷移は基本的にリセット状態を介して行います。ただし、通常走行状態から強制的にデバッグ走行状態に遷移することも可能です。

ユーザステートには通常走行状態と同様に特権モードとユーザモードがあります。ただし、デバッグ用のブレークを行った場合はデバッグステートへ遷移します。デバッグステートでは、特権モードで実行され、メモリ保護機能などすべて無効な状態ですべてのレジスタ/メモリにアクセス可能です。デバッグステートからユーザステートへの遷移はRETI命令実行により行います。

3.4 パイプライン動作

CPUのパイプライン動作について説明します。

FR81ではデコードステージまでは共通に処理しますが、実行ステージからは整数パイプライン, 浮動小数点パイプラインの2種のパイプラインを持ちます。各パイプライン間での完了は命令発行の順序とは異なりますが、プログラム列の順序に従った処理結果は保証します。

詳細は「FR Family FR81 32ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。

3.5 浮動小数点演算処理

CPUの浮動小数点演算処理について説明します。

本シリーズは、FPUを搭載しています。

浮動小数点演算処理の詳細は「FR Family FR81 32ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。

3.6 データ構造

データ構造について説明します。

FR81 ファミリCPU で扱うことのできるデータ種類として、FR80 ファミリまでの整数型に加えて、単精度浮動小数点型が存在します。

整数型ではビットオーダリングとしてリトルエンディアン、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。

詳細は「FR Family FR81 32ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。

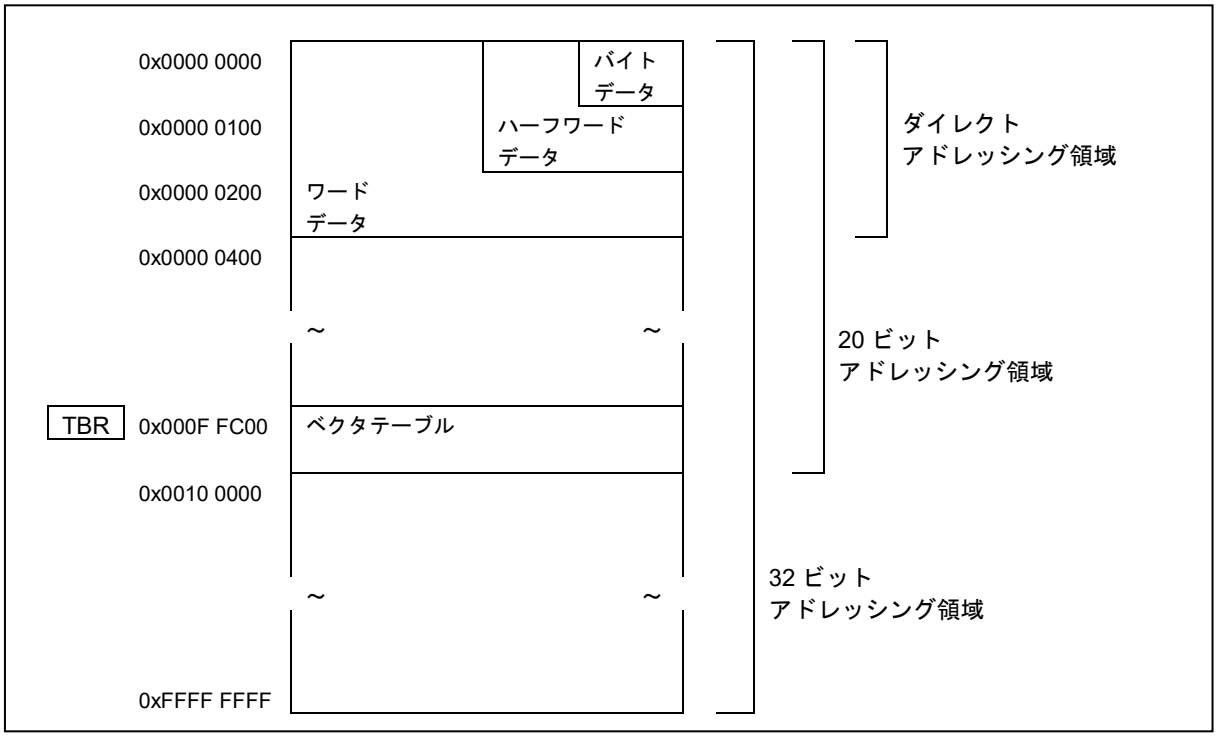
3.7 アドレッシング

アドレッシングについて説明します。

メモリ空間は32ビットリニアです。

CPU はアドレス空間をバイト単位で管理します。CPU からは、アドレス空間上のアドレスを32ビットの値で指定してアクセスします。アドレス空間を 図7-1 に示します。

図 3-2 メモリ空間



アドレス空間をメモリ空間とも呼びます。アドレス空間はCPU から見た論理アドレス空間です。アドレス変換は行いません。CPU からみた論理アドレスと、実際にメモリやI/O の配置されている物理アドレスは同一になります。

詳細は「FR Family FR81 32ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。

3.8 プログラミングモデル

プログラミングモデルについて説明します。

FR81 CPUには、汎用レジスタ，専用レジスタ，浮動小数点レジスタがあります。また、これらとは別にFR81コアとしてアドレスマップされたシステムレジスタがあります。

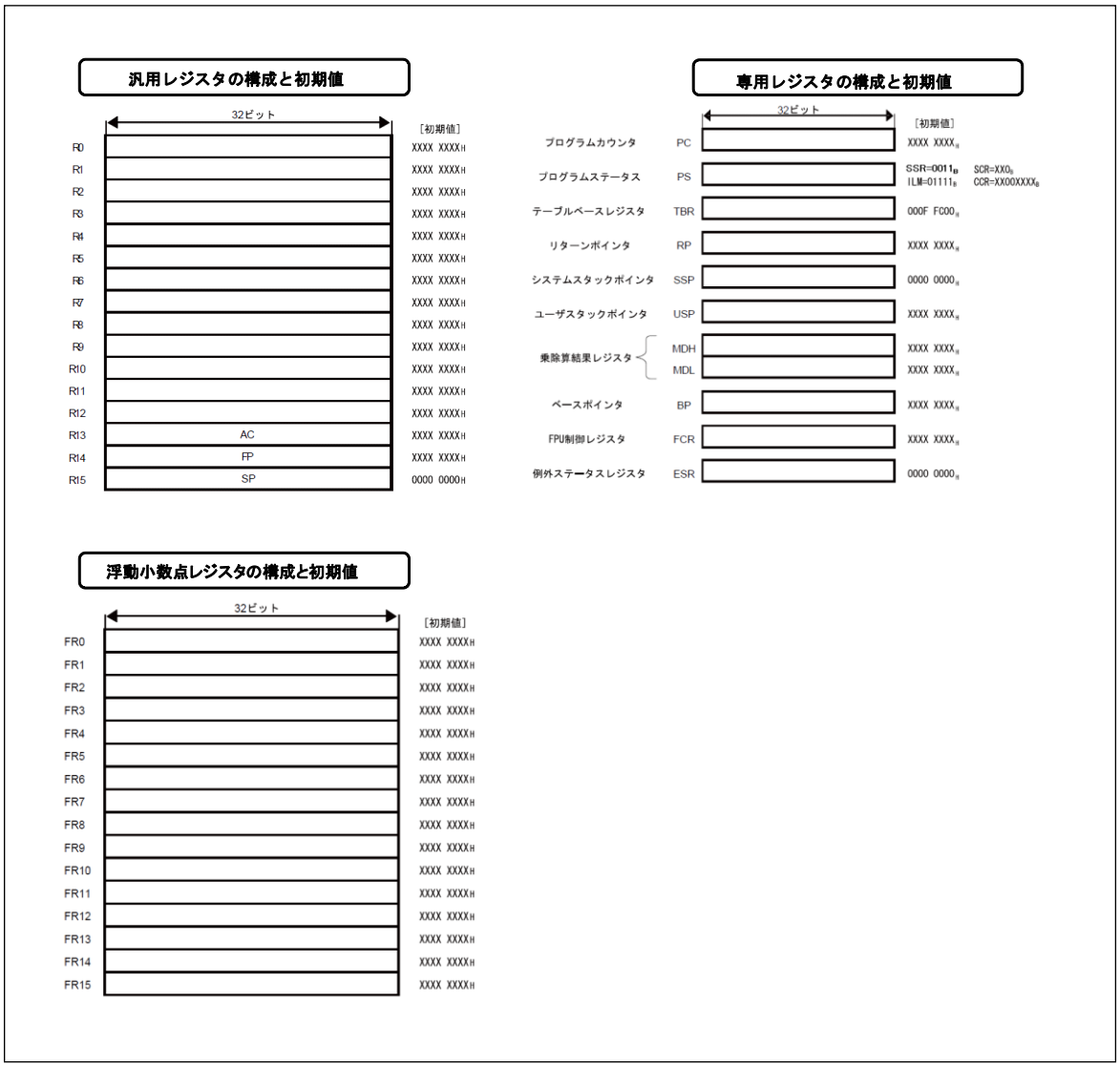
3.8.1 汎用レジスタ，専用レジスタ，浮動小数点レジスタ

汎用レジスタ、専用レジスタ、浮動小数点レジスタについて示します。

図 3-3に、本シリーズでの初期値を示します。

各レジスタの詳細は、「FR Family FR81 32ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。

図 3-3 汎用レジスタ，専用レジスタ，浮動小数点レジスタの初期値



3.8.2 システムレジスタ

システムレジスタについて説明します。

システムレジスタはアドレスマップされたシステム制御用のレジスタです。これらのレジスタは特権モードでのみアクセス可能です。システムレジスタには以下のようなものがあります。

- クロック制御関連レジスタ
- リセット制御関連レジスタ
- デバッグ制御関連レジスタ
- メモリ保護関連レジスタ
- DMA 関連レジスタ
- ウォッチドッグタイマレジスタ
- Wild register 制御レジスタ
- FLASH 制御レジスタ

ユーザモードでこれらのレジスタに読出し/書込みを行った場合は、不正命令例外(データアクセスエラー)を発生させます。

システムレジスタへのアクセス保護は、メモリ保護機能に優先して判定します。したがって、メモリ保護機能でシステムレジスタ領域へユーザアクセス可とした場合や特権モードでのアクセス不可とした場合もその設定は無効とし、特権モードでのみ読出し/書込み可能でユーザモードでは、読出し/書込み不可となります。

3.9 リセット・EIT 処理

リセット・EIT処理について説明します。

リセット・EIT処理とは、リセット，例外(Exception)，割込み(Interrupt)，トラップ(Trap)が検出されたときに通常とは異なるプログラムで実行される処理のことです。

3.9.1 リセット

リセットについて説明します。

リセットは、現在実行中の処理を強制的に中断し、デバイスを初期化して、リセットベクタエントリアドレスからプログラムを再開します。

＜注意事項＞

本シリーズではFiexedVector機能により、リセットベクタはフラッシュメモリ0xF_FFFC番地に書き込まれた値でなく、フラッシュメモリの先頭アドレス ＋0x0024番地が返されます。詳細は『FixedVector機能』の章を参照してください。

3.9.2 EIT 処理

EIT処理について説明します。

EIT処理は、現在実行中の処理を中断し、再開できる情報をメモリへ退避した後に、決められた処理プログラムへ制御を移します。

3.9.3 ベクタテーブル

ベクタテーブルについて示します。

表 3-1 ベクタテーブル

割込み要因	割込みベクタ番号		割込みレベル	オフセット	TBR 初期値時でのアドレス
	10 進	16 進			
リセット	0	00	―	0x3FC	0x000FFFFC
システム予約	1	01	―	0x3F8	0x000FFFF8
システム予約	2	02	―	0x3F4	0x000FFFF4
システム予約	3	03	―	0x3F0	0x000FFFF0
システム予約	4	04	―	0x3EC	0x000FFFEC
FPU 例外	5	05	―	0x3E8	0x000FFFE8
命令アクセス保護違反例外	6	06	―	0x3E4	0x000FFFE4
データアクセス保護違反例外	7	07	―	0x3E0	0x000FFFE0
データアクセスエラー割込み	8	08	―	0x3DC	0x000FFFDC
INTE 命令	9	09	―	0x3D8	0x000FFFD8
命令ブレーク	10	0A	―	0x3D4	0x000FFFD4
システム予約	11	0B	―	0x3D0	0x000FFFD0
システム予約	12	0C	―	0x3CC	0x000FFFCC
システム予約	13	0D	―	0x3C8	0x000FFFC8
不正命令例外	14	0E	―	0x3C4	0x000FFFC4
NMI 要求	15	0F	15(0xF) 固定	0x3C0	0x000FFFC0
ペリフェラル割込み #0	16	10	ICR00	0x3BC	0x000FFFBC
ペリフェラル割込み #1	17	11	ICR01	0x3B8	0x000FFFB8
ペリフェラル割込み #2	18	12	ICR02	0x3B4	0x000FFFB4
ペリフェラル割込み #3	19	13	ICR03	0x3B0	0x000FFFB0
ペリフェラル割込み #4	20	14	ICR04	0x3AC	0x000FFFAC
ペリフェラル割込み #5	21	15	ICR05	0x3A8	0x000FFFA8
ペリフェラル割込み #6	22	16	ICR06	0x3A4	0x000FFFA4
ペリフェラル割込み #7	23	17	ICR07	0x3A0	0x000FFFA0
ペリフェラル割込み #8	24	18	ICR08	0x39C	0x000FFF9C
ペリフェラル割込み #9	25	19	ICR09	0x398	0x000FFF98
ペリフェラル割込み #10	26	1A	ICR10	0x394	0x000FFF94
ペリフェラル割込み #11	27	1B	ICR11	0x390	0x000FFF90
ペリフェラル割込み #12	28	1C	ICR12	0x38C	0x000FFF8C
ペリフェラル割込み #13	29	1D	ICR13	0x388	0x000FFF88
ペリフェラル割込み #14	30	1E	ICR14	0x384	0x000FFF84
ペリフェラル割込み #15	31	1F	ICR15	0x380	0x000FFF80
ペリフェラル割込み #16	32	20	ICR16	0x37C	0x000FFF7C
ペリフェラル割込み #17	33	21	ICR17	0x378	0x000FFF78
ペリフェラル割込み #18	34	22	ICR18	0x374	0x000FFF74
ペリフェラル割込み #19	35	23	ICR19	0x370	0x000FFF70
ペリフェラル割込み #20	36	24	ICR20	0x36C	0x000FFF6C
ペリフェラル割込み #21	37	25	ICR21	0x368	0x000FFF68
ペリフェラル割込み #22	38	26	ICR22	0x364	0x000FFF64
ペリフェラル割込み #23	39	27	ICR23	0x360	0x000FFF60
ペリフェラル割込み #24	40	28	ICR24	0x35C	0x000FFF5C
ペリフェラル割込み #25	41	29	ICR25	0x358	0x000FFF58
ペリフェラル割込み #26	42	2A	ICR26	0x354	0x000FFF54
ペリフェラル割込み #27	43	2B	ICR27	0x350	0x000FFF50

割込み要因	割込みベクタ番号		割込みレベル	オフセット	TBR 初期値時でのアドレス
	10 進	16 進			
ペリフェラル割込み #28	44	2C	ICR28	0x34C	0x000FFF4C
ペリフェラル割込み #29	45	2D	ICR29	0x348	0x000FFF48
ペリフェラル割込み #30	46	2E	ICR30	0x344	0x000FFF44
ペリフェラル割込み #31	47	2F	ICR31	0x340	0x000FFF40
ペリフェラル割込み #32	48	30	ICR32	0x33C	0x000FFF3C
ペリフェラル割込み #33	49	31	ICR33	0x338	0x000FFF38
ペリフェラル割込み #34	50	32	ICR34	0x334	0x000FFF34
ペリフェラル割込み #35	51	33	ICR35	0x330	0x000FFF30
ペリフェラル割込み #36	52	34	ICR36	0x32C	0x000FFF2C
ペリフェラル割込み #37	53	35	ICR37	0x328	0x000FFF28
ペリフェラル割込み #38	54	36	ICR38	0x324	0x000FFF24
ペリフェラル割込み #39	55	37	ICR39	0x320	0x000FFF20
ペリフェラル割込み #40	56	38	ICR40	0x31C	0x000FFF1C
ペリフェラル割込み #41	57	39	ICR41	0x318	0x000FFF18
ペリフェラル割込み #42	58	3A	ICR42	0x314	0x000FFF14
ペリフェラル割込み #43	59	3B	ICR43	0x310	0x000FFF10
ペリフェラル割込み #44	60	3C	ICR44	0x30C	0x000FFF0C
ペリフェラル割込み #45	61	3D	ICR45	0x308	0x000FFF08
ペリフェラル割込み #46	62	3E	ICR46	0x304	0x000FFF04
遅延割込み	63	3F	ICR47	0x300	0x000FFF00
システム予約 (REALOS にて使用)	64	40	―	0x2FC	0x000FFEFC
システム予約 (REALOS にて使用)	65	41	―	0x2F8	0x000FFEF8
INT 命令で使用	66 | 255	42 | FF	―	0x2F4 | 0x000	0x000FFEF4 | 0x000FFC00

3.10 メモリ保護機能 (MPU)

メモリ保護機能 (MPU) について説明します。

3.10.1. 概要

3.10.2. レジスタ一覧

3.10.3. レジスタ説明

3.10.4. メモリ保護機能 (MPU) の動作説明

3.10.1 概要

メモリ保護機能 (MPU) の概要について示します。

本アーキテクチャではメモリ保護機能をサポートします。メモリ保護機能とは、指定した領域に対するアクセスを監視し、許可されていないアクセスであれば例外を発生させる機能です。ただし、システムレジスタへの保護指定は無効です。

- 命令・データ共通で 8 個の保護領域を指定可能
- 保護領域は、領域 0 の優先度が高く、1, 2, 3,…優先度が低くなります。(領域の重複可能)
- ページアドレスとページサイズで領域指定
  - ページサイズ: 16 バイトから 2^n 単位で指定
  - ページアドレス: ミスアラインもサポート
- 特権モード/ユーザモードそれぞれに以下のアクセス権を制御
  - 命令フェッチ: 許可/禁止
  - データリード: 許可/禁止
  - データライト: 許可/禁止
- 領域ごとにアクセス属性を指定
  - バッファ: 許可/禁止
- 未定義領域はデフォルト領域として、アクセス権・属性を制御
- 保護違反発生時は保護違反例外を発生
- メモリ保護機能用レジスタはシステムレジスタとして特権モードでのみアクセス可能
- データアクセスエラー通知機能
- I/O 領域(00000000_H ～$0000FFFF_H$)は、バッファ禁止固定

3.10.2 レジスター一覧

レジスター一覧について示します。

表 3-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0310	予約		MPUCR		MPU 制御レジスタ
0x0314	予約				
0x0318	予約				
0x031C	予約				
0x0320	DPVAR				データアクセス保護違反アドレスレジスタ
0x0324	予約		DPVSR		データアクセス保護違反ステータスレジスタ
0x0328	DEAR				データアクセスエラーアドレスレジスタ
0x032C	予約		DESR		データアクセスエラーステータスレジスタ
0x0330	PABR0				保護領域ベースアドレスレジスタ 0
0x0334	予約		PACR0		保護領域制御レジスタ 0
0x0338	PABR1				保護領域ベースアドレスレジスタ 1
0x033C	予約		PACR1		保護領域制御レジスタ 1
0x0340	PABR2				保護領域ベースアドレスレジスタ 2
0x0344	予約		PACR2		保護領域制御レジスタ 2
0x0348	PABR3				保護領域ベースアドレスレジスタ 3
0x034C	予約		PACR3		保護領域制御レジスタ 3
0x0350	PABR4				保護領域ベースアドレスレジスタ 4
0x0354	予約		PACR4		保護領域制御レジスタ 4
0x0358	PABR5				保護領域ベースアドレスレジスタ 5
0x035C	予約		PACR5		保護領域制御レジスタ 5
0x0360	PABR6				保護領域ベースアドレスレジスタ 6
0x0364	予約		PACR6		保護領域制御レジスタ 6
0x0368	PABR7				保護領域ベースアドレスレジスタ 7
0x036C	予約		PACR7		保護領域制御レジスタ 7

3.10.3　レジスタ説明

レジスタについて示します。

3.10.3.1. MPU制御レジスタ (MPUCR)
3.10.3.2. 命令アクセス保護違反アドレスレジスタ (IPVAR)
3.10.3.3. 命令アクセス保護違反ステータスレジスタ (IPVSR)
3.10.3.4. データアクセス保護違反アドレスレジスタ(DPVAR)
3.10.3.5. データアクセス保護違反ステータスレジスタ(DPVSR)
3.10.3.6. データアクセスエラーアドレスレジスタ(DEAR)
3.10.3.7. データアクセスエラーステータスレジスタ(DESR)
3.10.3.8. 保護領域ベースアドレスレジスタ0～7(PABR0～PABR7)
3.10.3.9. 保護領域制御レジスタ0～7(PACR0～PACR7)

3.10.3.1 MPU 制御レジスタ (MPUCR)

MPU制御レジスタのビット構成について説明します。

MPUの有効/無効ならびにデフォルト領域(保護領域指定されていない領域)での特権モード, ユーザモードでのアクセス許可を設定します。

■ MPUCR : アドレス 0312$_H$ (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	PIE	PRE	PWE	UIE	URE	UWE	予約	BE
初期値	0	0	0	0	0	0	−	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R0,W0	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				PAN1	PAN0	DEE	MPE
初期値	−	−	−	−	0	1	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,WX	R1,WX	R/W	R/W

[bit15] PIE (Privilege Mode Instruction Fetch Enable)

デフォルト領域 (保護領域指定されていない領域) に対する特権モードでの命令フェッチを許可するためのビットです。

PIE	デフォルト領域へのアクセス
0	特権モードで命令フェッチ禁止(初期値)
1	特権モードで命令フェッチ可能

[bit14] PRE (Privilege Mode Read Access Enable)

デフォルト領域 (保護領域指定されていない領域) に対する特権モードでのデータリードアクセスを許可するためのビットです。

PRE	デフォルト領域へのアクセス
0	特権モードでリードアクセス禁止 (初期値)
1	特権モードでリードアクセス可能

[bit13] PWE (Privilege Mode Write Access Enable)

デフォルト領域 (保護領域指定されていない領域) に対する特権モードでのデータライトアクセスを許可するためのビットです。

PWE	デフォルト領域へのアクセス
0	特権モードでライトアクセス禁止(初期値)
1	特権モードでライトアクセス可能

[bit12] UIE (User Mode Instruction Fetch Enable)

デフォルト領域(保護領域指定されていない領域)に対するユーザモードでの命令フェッチを許可するためのビットです。

UIE	デフォルト領域へのアクセス
0	ユーザモードで命令フェッチ禁止 (初期値)
1	ユーザモードで命令フェッチ可能

[bit11] URE (User Mode Read Access Enable)

デフォルト領域 (保護領域指定されていない領域) に対するユーザモードでのデータリードアクセスを許可するためのビットです。

URE	デフォルト領域へのアクセス
0	ユーザモードでリードアクセス禁止 (初期値)
1	ユーザモードでリードアクセス可能

[bit10] UWE (User Mode Write Access Enable)

デフォルト領域 (保護領域指定されていない領域) に対するユーザモードでのデータライトアクセスを許可するためのビットです。

UWE	デフォルト領域へのアクセス
0	ユーザモードでライトアクセス禁止 (初期値)
1	ユーザモードでライトアクセス可能

[bit9] 予約

"0"を書き込んでください。"0"が読み出されます。

[bit8] BE (Buffer Enable)

デフォルト領域(保護領域指定されていない領域)に対して、データアクセス時にバッファを使用することを許可します。バッファ使用が禁止されている場合、CPUはパイプライン動作を停止しデータアクセスの完了を待って次の動作を開始します。したがって、データアクセス効率は下がりますが、命令に同期したデータアクセスが可能となります。バッファが禁止されている場合のみデータアクセスでエラーが発生した場合に不正命令例外を発生します。バッファが許可されている場合にはデータアクセスエラーは割込みとして通知可能です。

BE	デフォルト領域の Bufferable 指定
0	バッファ禁止 (初期値)
1	バッファ許可

[bit7～bit4] 予約

予約ビットです。書込み時は必ず"0"を書き込んでください。

[bit3, bit2] PAN[1:0] (Protection Area Number)

保護領域に指定可能な領域の実装個数を示します。本ビットはリードオンリでハードウェアで実装した個数を示します。

PAN[1:0]	メモリ保護領域実装数
00	予約
01	8 領域
10	12 領域
11	16 領域

[bit1] DEE (Data Access Error Interrupt Enable)

バッファ動作が有効になっている領域でデータアクセスエラーが発生したときの割込み発生を許可するビットです。このビットが有効なときにバッファ動作許可領域でデータアクセスエラーが発生するとデータアクセスエラー割込みを発生します。このときエラーを発生させたアドレスはデータアクセスエラーアドレスレジスタ (DEAR) にアクセス内容はデータアクセスエラーステータスレジスタ (DESR) に保持されます。割込みが禁止されている場合は上記レジスタの更新のみを行います。

DEE	データアクセスエラー割込みの許可
0	データアクセスエラー割込み禁止 (初期値)
1	データアクセスエラー割込み許可

[bit0] MPE (Memory Protection Unit Enable)

メモリ保護機能を有効にするためのビットです。メモリ保護機能が無効な場合、すべての領域に対するアクセスがバッファ禁止の設定になります。

MPE	メモリ保護機能
0	メモリ保護機能無効 (初期値)
1	メモリ保護機能有効

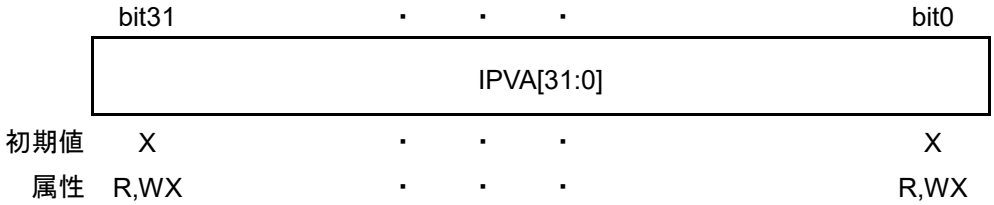
3.10.3.2 命令アクセス保護違反アドレスレジスタ (IPVAR)

命令アクセス保護違反アドレスレジスタのビット構成について説明します。

命令アクセス保護違反を発生したアドレスを保持します。

「3.10.4.2 命令アクセス保護違反 および 3.10.4.7 注意事項」も参照してください。

■ IPVAR： アドレス 0318$_H$ (アクセス： ワード)



[bit31～bit0] IPVA[31:0] (Instruction fetch Protection Violation Address)

命令アクセス保護違反ステータスレジスタで違反が発生していないとき (IPVSR:IPV=0) に、命令アクセス保護違反を発生したアドレスを保持します。アラインはされません。

＜注意事項＞

本レジスタは、使用禁止です。

3.10.3.3 命令アクセス保護違反ステータスレジスタ (IPVSR)

命令アクセス保護違反ステータスレジスタのビット構成について説明します。

命令アクセス保護違反時のステータスを表示します。

本レジスタの内容はIPV=0のときのみ、ハードウェアで更新されます。IPVビットへの"0"書込みのみ有効です。そのほかのビットへの書込みおよびIPVビットへの"1"書込みは無効です。

「3.10.4.2 命令アクセス保護違反 および 3.10.4.7 注意事項」 も参照してください。

■ IPVSR : アドレス 031E$_H$ (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	−	−	−	−	−	−	−	−
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		SZ[1:0]		MD	予約		IPV
初期値	−	−	0	0	0	−	−	0
属性	R0,W0	R0,W0	R,WX	R,WX	R,WX	R0,W0	R0,W0	R,W

[bit15～bit6, bit2, bit1] 予約

予約ビットです。必ず"0"を書き込んでください。

[bit5, bit4] SZ[1:0]

違反を発生したときのアクセスサイズです。

SZ[1:0]	アクセスサイズ
00	バイト
01	ハーフワード
10	ワード
11	予約

[bit3] MD

アクセス時のモードを示します。

MD	動作モード
0	ユーザモードでのアクセス
1	特権モードでのアクセス

[bit0] IPV (Instruction fetch Protection Violation)

命令アクセス保護違反が発生したことを示します。新たな保護違反の内容を保持するにはこのビットをクリアしてください。

IPV	命令アクセス保護違反
0	命令アクセス保護違反未検出 (初期値)
1	命令アクセス保護違反検出

＜注意事項＞

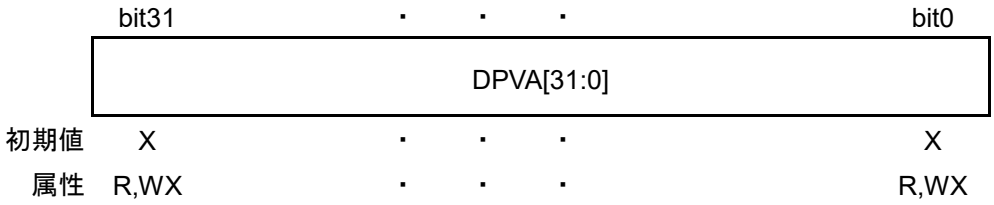
本レジスタは、使用禁止です。

3.10.3.4 データアクセス保護違反アドレスレジスタ(DPVAR)

データアクセス保護違反アドレスレジスタのビット構成について説明します。

データアクセス保護違反を発生したアドレスを保持します。

■ DPVAR： アドレス 0320H (アクセス： ワード)



[bit31～bit0] DPVA[31:0] (Data Access Protection Violation Address)

データアクセス保護違反ステータスレジスタで違反が発生していないとき (DPVSR:DPV=0) に、データアクセス保護違反を発生したアドレスを保持します。CPUからの要求アドレスを示し、アドレスはアラインされません。

3.10.3.5 データアクセス保護違反ステータスレジスタ(DPVSR)

データアクセス保護違反ステータスレジスタのビット構成について説明します。

データアクセス保護違反時のステータスを表示します。

本レジスタはDPV=0のときのみハードウェアで更新されます。DPVビットへの"0"書込みのみ有効です。そのほかのビットおよびDPVビットへの"1"書込みは無効です。

■ DPVSR： アドレス 0326H (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RW[1:0]		SZ[1:0]		MD	予約		DPV
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R0,W0	R0,W0	R,W

[bit15～bit8, bit2, bit1] 予約

予約ビットです。必ず"0"を書き込んでください。

[bit7, bit6] RW[1:0] (Read/Write)

違反を発生したときのアクセス種類です。リードモディファイライトの実行にはリードとライトのアクセス権が必要になり、判定は最初のリードサイクルで行いますので、リードモディファイライトのライトで違反となる場合にもRW=01B リード (リードモディファイライト) となります。

RW[1:0]	アクセス種類
00	リード
01	リード (リードモディファイライト)
10	ライト
11	予約

[bit5, bit4] SZ[1:0]

違反を発生したときのアクセスサイズです。

SZ[1:0]	アクセスサイズ
00	バイト
01	ハーフワード
10	ワード
11	予約

[bit3] MD

アクセス時のモードを示します。

MD	動作モード
0	ユーザモードでのアクセス
1	特権モードでのアクセス

[bit0] DPV (Data Access Protection Violation)

データアクセス保護違反が発生したことを示します。新たな保護違反の内容を保持するにはこのビットをクリアしてください。

本ビットへは"0"書込みのみ有効です。"1"書込みは無効です。

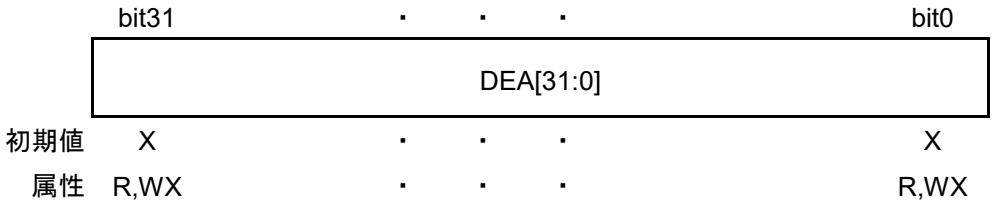
DPV	データアクセス保護違反
0	データアクセス保護違反未検出 (初期値)
1	データアクセス保護違反検出

3.10.3.6 データアクセスエラーアドレスレジスタ(DEAR)

データアクセスエラーアドレスレジスタのビット構成について説明します。

データアクセスエラーを発生したアドレスを保持します。

■ DEAR : アドレス 0328H (アクセス: ワード)



[bit31～bit0] DEA[31:0] (Data Access Error Address)

データアクセスエラーステータスレジスタで違反が発生していないとき (DESR:DAE=0) に、データアクセスエラーを発生したアドレスを保持します。システムレジスタアクセス保護違反時は、CPUからのアクセスアドレスそのままでアラインされません。バスアクセスを行った結果エラーとなる場合にはアドレスはアラインされます。

3.10.3.7 データアクセスエラーステータスレジスタ(DESR)

データアクセスエラーステータスレジスタのビット構成について説明します。

データアクセスエラー時のステータスを表示します。本レジスタはDAE=0のときのみハードウェアで更新されます。DAEビットへの"0"書込みのみ有効です。そのほかのビットへの書込みおよびDAEビットへの"1"書込みは無効です。

■ DESR : アドレス 032E$_H$ (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RW[1:0]		SZ[1:0]		MD	予約		DAE
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R0,W0	R0,W0	R,W

[bit15～bit8, bit2, bit1] 予約

予約ビットです。必ず"0"を書き込んでください。"0"が読み出されます。

[bit7, bit6] RW[1:0] (Read/Write)

エラーを発生したときのアクセス種類です。

RW[1:0]	アクセス種類
00	リード
01	リード (リードモディファイライト)
10	ライト
11	予約

[bit5, bit4] SZ[1:0]

エラーを発生したときのアクセスサイズです。

SZ[1:0]	アクセスサイズ
00	バイト
01	ハーフワード
10	ワード
11	予約

[bit3] MD

アクセス時のモードを示します。

MD	動作モード
0	ユーザモードでのアクセス
1	特権モードでのアクセス

[bit0] DAE (Data Access Error)

データアクセスエラーが発生したことを示します。新たなデータエラーの内容を保持するにはこのビットをクリアしてください。

データアクセスエラー割込みを有効にしている場合､本ビットをクリアすることで割込み要求を取り下げます。本ビットへは"0"書込みのみ有効です。"1"書込みは無効です。

DAE	データアクセスエラー
0	データアクセスエラー未検出 (初期値)
1	データアクセスエラー検出

3.10.3.8 保護領域ベースアドレスレジスタ 0～7(PABR0～PABR7)

保護領域ベースアドレスレジスタ0～7のビット構成について説明します。

各MPUチャネルの保護領域のベースアドレスを設定します。

■ PABR0～PABR7： アドレス 0330_H , 0338_H , 0340_H・・・(アクセス: ワード)

	bit31		・	・	・		bit8
	PABR[31:8]						
初期値	X	X	・	・	・	X	X	X
属性	R/W	R/W	・	・	・	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PABR[7:0]							
初期値	X	X	X	X	0	0	0	0
属性	R/W	R/W	R/W	R/W	R0,WX	R0,WX	R0,WX	R0,WX

[bit31～bit0] PABR[31:0] (Protection Area Base Address Register)

保護領域のベースアドレスを示します。ここで指定したアドレスから､保護領域制御レジスタ(PACR0～PACR7)で指定されたサイズが対象となる保護領域となります。アドレスは、保護領域サイズでアラインする必要はありません。

PABRレジスタの下位4ビットは0000_B 固定です。

3.10.3.9 保護領域制御レジスタ 0～7(PACR0～PACR7)

保護領域制御レジスタ0～7のビット構成について説明します。

各MPUチャネルによるアクセス許可制限を設定します。

■ PACR0～PACR7： アドレス 0336H, 033EH, 0346H・・・(アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	PIE	PRE	PWE	UIE	URE	UWE	予約	BE
初期値	0	0	0	0	0	0	－	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R0,W0	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ASZ[4:0]					予約		PAE
初期値	0	0	0	0	0	－	－	0
属性	R/W	R/W	R/W	R/W	R/W	R0,W0	R0,W0	R/W

[bit15] PIE (Privilege Mode Instruction Fetch Enable)

指定保護領域に対する特権モードでの命令フェッチを許可するためのビットです。

PIE	指定保護領域へのアクセス
0	特権モードで命令フェッチ禁止 (初期値)
1	特権モードで命令フェッチ可能

[bit14] PRE (Privilege Mode Read Access Enable)

指定保護領域に対する特権モードでのデータリードアクセスを許可するためのビットです。

PRE	指定保護領域へのアクセス
0	特権モードでリードアクセス禁止 (初期値)
1	特権モードでリードアクセス可能

[bit13] PWE (Privilege Mode Write Access Enable)

指定保護領域に対する特権モードでのデータライトアクセスを許可するためのビットです。

PWE	指定保護領域へのアクセス
0	特権モードでライトアクセス禁止 (初期値)
1	特権モードでライトアクセス可能

[bit12] UIE (User Mode Instruction Fetch Enable)

指定保護領域に対するユーザモードでの命令フェッチを許可するためのビットです。

UIE	指定保護領域へのアクセス
0	ユーザモードで命令フェッチ禁止 (初期値)
1	ユーザモードで命令フェッチ可能

[bit11] URE (User Mode Read Access Enable)

指定保護領域に対するユーザモードでのデータリードアクセスを許可するためのビットです。

URE	指定保護領域へのアクセス
0	ユーザモードでリードアクセス禁止 (初期値)
1	ユーザモードでリードアクセス可能

[bit10] UWE (User Mode Write Access Enable)

指定保護領域に対するユーザモードでのデータライトアクセスを許可するためのビットです。

UWE	指定保護領域へのアクセス
0	ユーザモードでライトアクセス禁止(初期値)
1	ユーザモードでライトアクセス可能

[bit9] 予約

"0"を書き込んでください。"0"が読み出されます。

[bit8] BE (Buffer Enable)

指定保護領域に対して、データアクセス時にバッファを使用することを許可します。バッファ使用が禁止されている場合、CPUはパイプライン動作を停止しデータアクセスの完了を待って次の動作を開始します。したがって、データアクセス効率は下がりますが、命令に同期したデータアクセスが可能となります。バッファが禁止されている場合のみデータアクセスでエラーが発生した場合に不正命令例外を発生します。バッファが許可されている場合にはデータアクセスエラーは割込みとして通知可能です。

BE	指定保護領域の Buffer enable 指定
0	バッファ禁止 (初期値)
1	バッファ許可

[bit7～bit3] ASZ[4:0] (Area Size)

指定保護領域のサイズを指定します。アドレス指定は下記サイズにアラインする必要はありません。また、アドレス・サイズ指定で領域の下限が$FFFFFFFF_H$を超える場合は$FFFFFFFF_H$を領域の下限とします。

ASZ[4:0]	指定保護領域のサイズ
00000	予約
00001	予約
00010	予約
00011	16B
00100	32B
00101	64B
00110	128B
00111	256B
01000	512B
01001	1KB
01010	2KB
01011	4KB
01100	8KB
01101	16KB
01110	32KB
01111	64KB
10000	128KB
10001	256KB
10010	512KB
10011	1MB
10100	2MB
10101	4MB
10110	8MB
10111	16MB
11000	32MB
11001	64MB
11010	128MB
11011	256MB
11100	512MB
11101	1GB
11110	2GB
11111	4GB

[bit2, bit1] 予約

予約ビットです。書込み時は必ず"0"を書き込んでください。

[bit0] PAE (Protection Area Enable)

メモリ保護機能を有効にするためのビットです。

PAE	メモリ保護領域
0	メモリ保護領域指定無効 (初期値)
1	メモリ保護領域指定有効

3.10.4 メモリ保護機能 (MPU) の動作説明

メモリ保護機能 (MPU) の動作について説明します。

3.10.4.1. メモリ保護領域の設定

3.10.4.2. 命令アクセス保護違反

3.10.4.3. データアクセス保護違反

3.10.4.4. データアクセスエラー

3.10.4.5. 遅延スロットでのメモリ保護動作

3.10.4.6. DEAR, DESRの更新

3.10.4.7. 注意事項

3.10.4.1 メモリ保護領域の設定

メモリ保護領域の設定について説明します。

メモリ保護機能はアドレスとサイズで指定される最大8個の保護領域とそれらの領域に含まれないデフォルト領域に対して、特権モード/ユーザモードで命令/データリード/データライトそれぞれ許可・禁止の設定をすることにより行います。同時に各領域に対して、バッファ許可・禁止の設定も可能です。

指定した保護領域に重なりがある場合は、領域番号の小さい方を優先させます。

メモリ保護機能が無効な場合 (MPUCR:MPE=0) は、すべての領域に対してアクセス可能でバッファ禁止としてアクセスを行います。

3.10.4.2 命令アクセス保護違反

命令アクセス保護違反について説明します。

メモリ保護ユニット (MPU) は、CPUの命令フェッチを監視しアクセスのあった領域への命令フェッチが許可されているかの判断を行います。命令アクセス保護違反例外が発生したときの命令アドレスはシステムスタックに退避されているPC値から判断してください。

3.10.4.3 データアクセス保護違反

データアクセス保護違反について説明します。

メモリ保護ユニット (MPU) は、CPUのデータアクセスを監視し、該当領域へのアクセス (リード・ライト) が許可されているかの判断を行います。アクセスが許可されていなかった場合、MPUはそのアドレスとアクセス情報をそれぞれデータアクセス保護違反アドレスレジスタ (DPVAR), データアクセス保護違反ステータスレジスタ (DPVSR) に格納します。ただし、既にデータアクセス保護違反情報が上記レジスタに存在する場合 (DPVSR:DPV=1) 上書きは行いません。このとき違反を起こしたデータアクセスは行われません。

複数回のデータアクセスを行う命令実行中にデータアクセス保護違反が発生した場合、違反が発生するまでに実行したデータアクセスは取り消しされません。LDM0, LDM1, STM0, STM1, FLDM, FSTM命令の途中でデータアクセス保護違反例外が発生した場合例外ステータスレジスタESR:RLに残りのレジスタリストが保持されます。

EIT処理シーケンスおよびRETI命令中にデータアクセス保護違反が発生した場合CPUは停止状態となり、ブレーク割込みおよびリセットでのみ復帰可能となります。

3.10.4.4 データアクセスエラー

データアクセスエラーについて示します。

データアクセス中に以下の条件が成立した場合にデータアクセスエラーとし、データアクセスエラーアドレスレジスタ (DEAR), データアクセスエラーステータスレジスタ (DESR) にそのときのアクセス情報を格納します。ただし、既にデータアクセスエラー情報が上記レジスタに存在する場合 (DESR:DAE=1) 上書きは行いません。

- ユーザモードでのシステムレジスタアクセス
- データアクセス中のバスエラー

データアクセス中のバスエラーはバッファ許可アクセスとバッファ禁止アクセスとでエラー発生後の動作が異なります。ユーザモードでのシステムレジスタアクセスは常に不正命令例外(データアクセス)として処理されます。

バッファ禁止領域へのアクセス中にデータアクセスエラーが発生した場合、CPUは不正命令例外 (データアクセスエラー) として処理します。

バッファ許可領域へのアクセス中にデータアクセスエラーが発生した場合、MPU制御レジスタMPUCR:DEE=1でデータアクセスエラー割込みが有効であればデータアクセスエラー割込みを通知し、CPUはデータアクセスエラー割込み処理を行います。バッファ許可領域へのアクセス中にデータアクセスエラーが発生した場合、CPUは後続の命令を実行していますので、データアクセスエラー割込み時に退避される PC 値はデータアクセス命令を行った命令のPC 値とはなりません。

複数回のデータアクセスを行う命令実行中に不正命令例外 (データアクセスエラー) が発生した場合、エラーが発生するまでに実行したデータアクセスは取り消しされません。LDM0, LDM1, STM0, STM1, FLDM, FSTM命令の途中で不正命令例外 (データアクセスエラー) が発生した場合例外ステータスレジスタESR:RLに残りのレジスタリストが保持され、ESR:INV6データアクセスエラーを示すビットがセットされます。

EIT処理シーケンスおよびRETI命令中に不正命令例外 (データアクセスエラー) が発生した場合CPUは停止状態となり、ブレーク割込みおよびリセットでのみ復帰可能となります。

3.10.4.5 遅延スロットでのメモリ保護動作

遅延スロットでのメモリ保護動作について示します。

遅延スロットに配置された命令は16ビットとして処理します。したがって、遅延スロットに32ビット命令を配置しその下位16ビットに命令アクセス保護違反要因や命令アクセスエラー要因があったとしても不正命令例外 (遅延スロットへ配置できない命令) として例外が発生します。

3.10.4.6 DEAR, DESR の更新

DEAR, DESRの更新について示します。

データアクセスエラーアドレスレジスタ (DEAR), データアクセスエラーステータスレジスタ (DESR) は次の場合に更新されます。

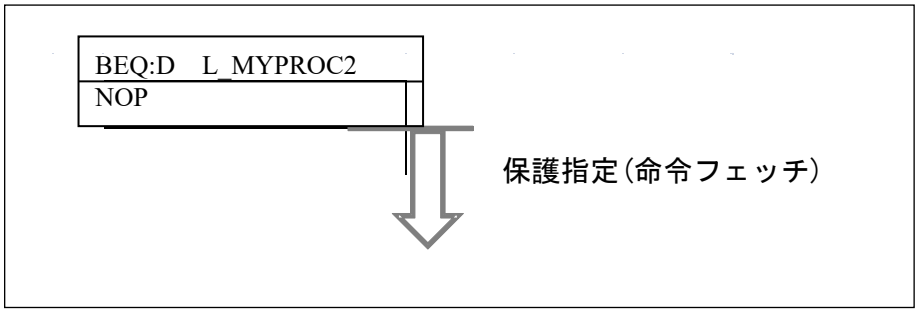
- ユーザモードでのシステムレジスタアクセス (不正命令例外)
- バッファ禁止領域アクセスでのバスエラー (不正命令例外)
- バッファ許可領域アクセスでのバスエラー (データアクセスエラー割込み)

不正命令例外の発生するケースでは、該当するアクセスを行った命令で DEAR, DESRが更新されデータアクセスエラー割込みが発生するケースでは命令動作とは非同期に更新されます。なお、要因が同時に発生した場合は不正命令例外要因を優先します。

3.10.4.7 注意事項

注意事項について示します。

- アクセス保護違反例外はアクセス保護違反となる命令の実行時に発生します。詳細は「FR Family FR81 32ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。命令アクセス保護違反と命令アクセス保護違反例外については「3.10.4.2 命令アクセス保護違反」も参照してください。
- 下図のように遅延スロットと命令アクセス保護領域の境界が重なる場合、分岐の成立/不成立に関係なく命令アクセス保護違反が発生します。例外が発生する PC は遅延分岐命令の PC です。



4. 動作モード

動作モードについて説明します。

4.1 概要

動作モードの概要について説明します。

リセット解除後に決定される、本シリーズの動作モードについて説明します。各消費電力制御のモードや各クロック選択のモードについては『消費電力制御』の章を参照してください。

4.2 特長

動作モードの特長について説明します。

本シリーズは以下のモードを持ちます。

● ユーザモード

外バスが使用可能です。

プログラムは内蔵FLASHから起動します。

● シリアルライタモード

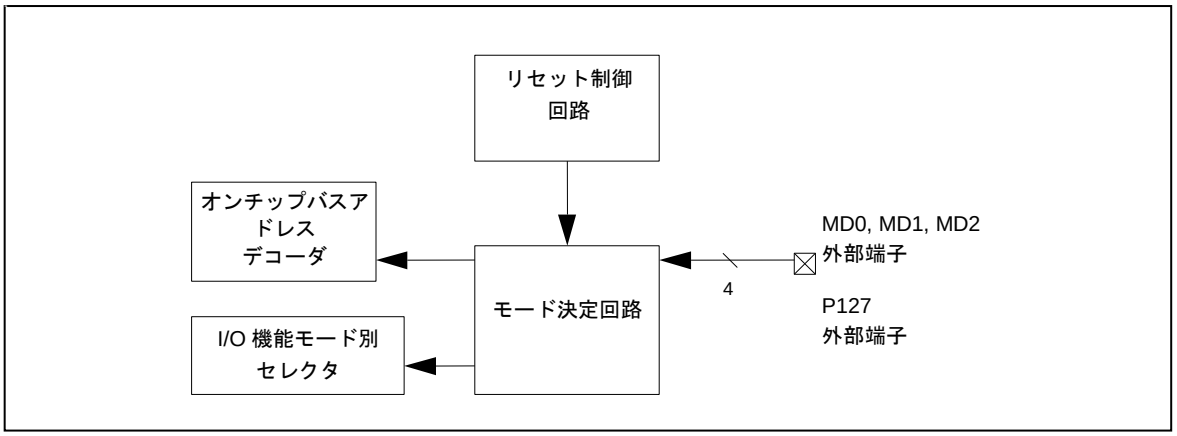
シリアルライタを使用して、内蔵FLASHのプログラムを行うモードです。

プログラムは、内蔵Boot-ROMから起動します。

4.3 構成

動作モードの構成について説明します。

図 4-1 ブロックダイヤグラム



4.4 レジスタ

動作モードのレジスタについて説明します。

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x07FC	BMODR	予約	予約	予約	バスモードデータレジスタ

■ バスモードレジスタ ：BMODR (Bus MODe Register)

起動時に設定されたモードを示すレジスタです。読出しのみ可能で、書込みは本レジスタの値に影響しません。

● BMODR： アドレス 07FC$_H$ (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BMOD[7:0]							
初期値	*	*	*	*	*	*	*	*
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

* 動作モードにより、初期値が異なります。

[bit7～bit0] BMOD[7:0] : 動作モード

動作モードを示します。書込みは無効です。

BMOD[7:0]	動作モード
0101xxxx	ユーザモード
0111xx1x	シリアルライタモード

4.5 動作説明

動作モードの動作について示します。

4.5.1. MD0, MD1, MD2, P127端子の設定

4.5.2. 動作モードの取込み

4.5.3. 各モードの説明

4.5.1 MD0, MD1, MD2, P127 端子の設定

MD0, MD1, MD2, P127端子の設定について示します。

表 4-1 端子設定

動作モード	MD2	MD1	MD0	P127
ユーザモード	0	0	0	−
シリアルライタモード	0	0	1	1

上記組合せ以外の設定は禁止です。

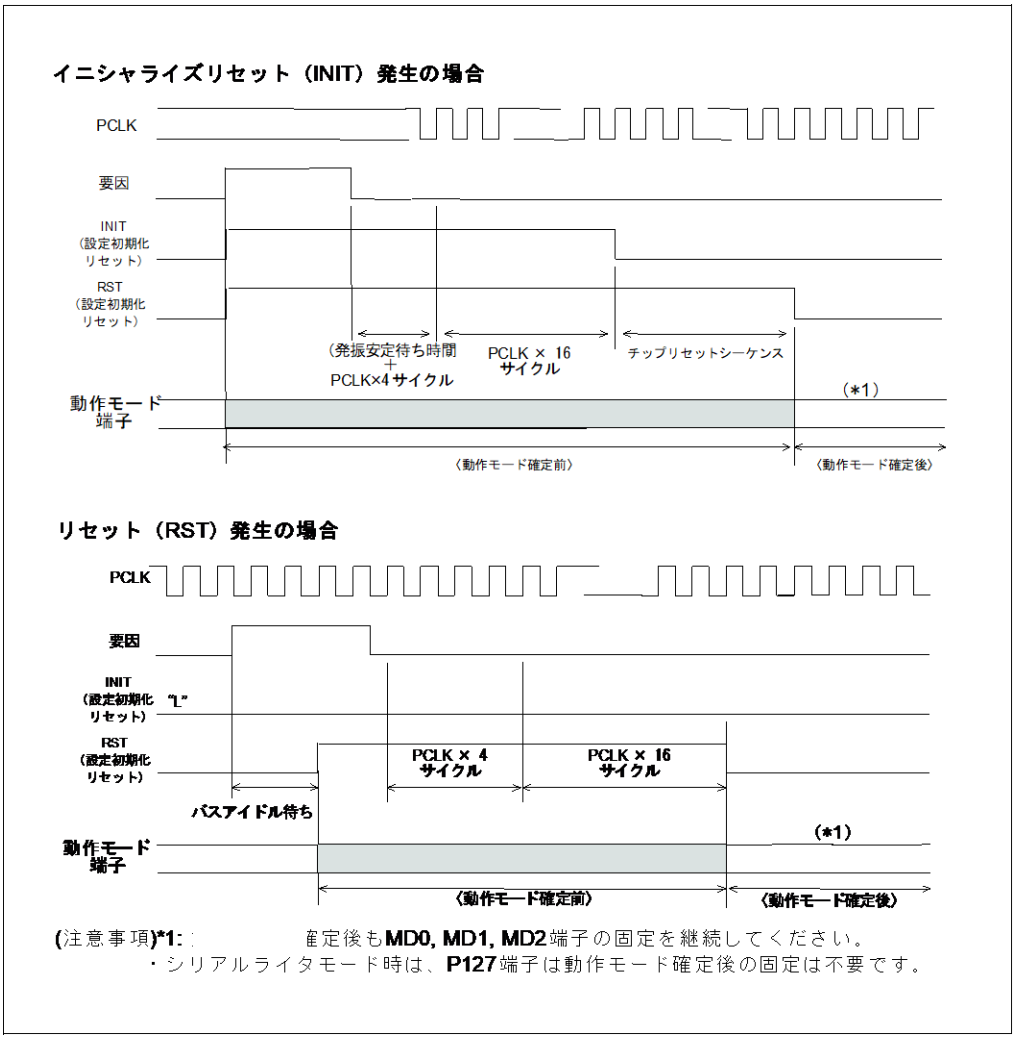
4.5.2 動作モードの取込み

動作モードの取込みについて説明します。

動作モードの取り込みはRST (リセット) によりサンプリングされます。RSTが発行されてから解除されるまでの区間はMD0, MD1, MD2, P127端子入力が確定している必要があります。(ユーザモードではP127端子確定は不要です。)

以下にリセット要因発生～動作モード確定までのシーケンスを示します。

図 4-2 動作モードの取込みタイミング図



4.5.3 各モードの説明

各モードについて説明します。

以下に各動作モードの動作の詳細を示します。

■ ユーザモード

外部リセット端子によるリセット入力時に、直ちに外部バス端子をリセット状態にします。詳細は『付録』の『付録D.CPU状態における端子状態』を参照してください。

■ シリアルライタモード

弊社営業担当にお問い合わせください。

5. クロック

クロックについて説明します。

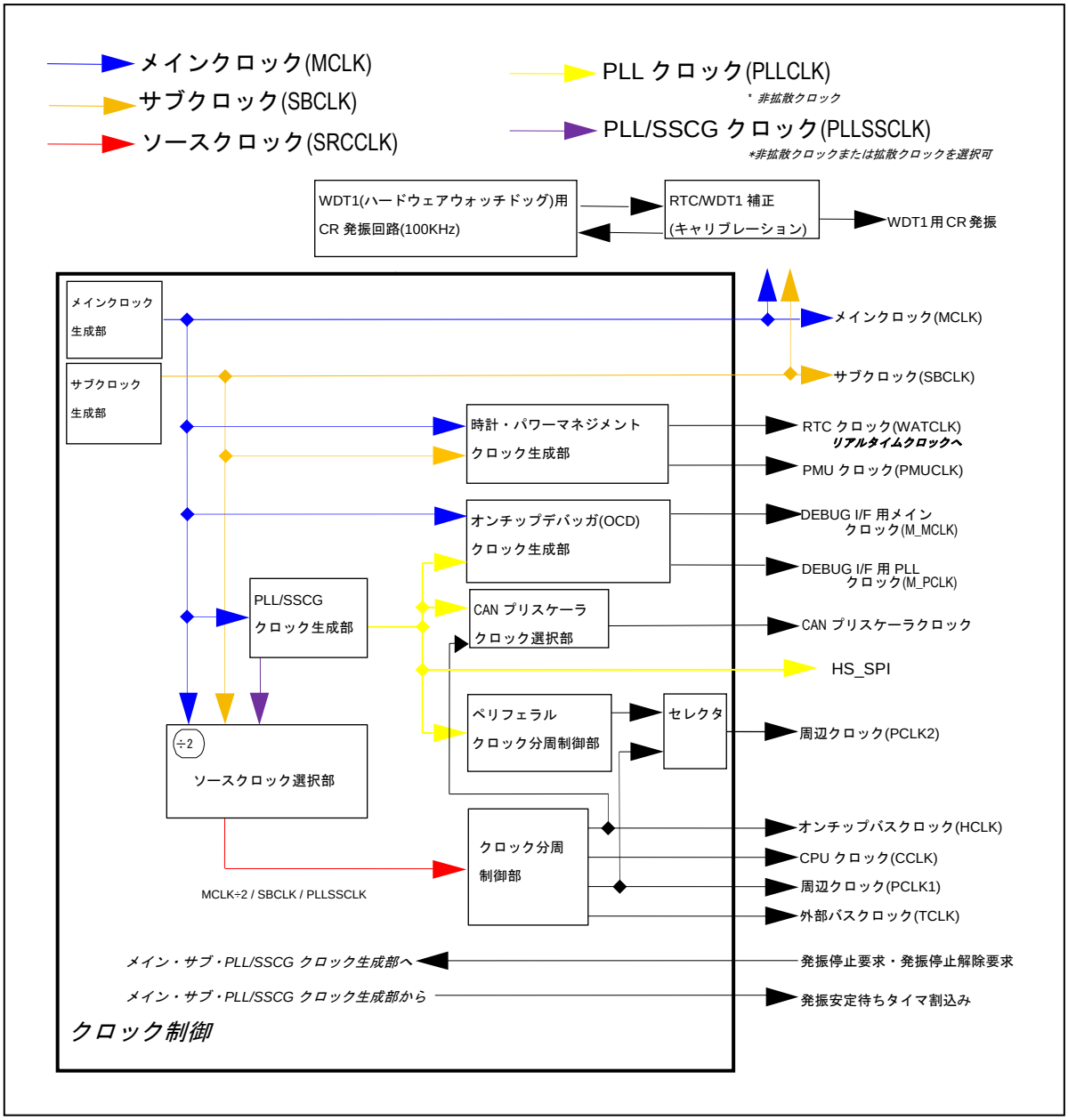
5.1 概要

クロックの概要について説明します。

搭載発振回路で2系統のクロックを生成し、それらよりチップ内の各クロック系統を生成します。また、本シリーズはウォッチドッグタイマ1用のCR発振回路を搭載しています。

- 搭載発振回路用外部端子:
  メインクロック　　: 水晶発振子を接続します。
  サブクロック　　: 水晶発振子を接続します。
- ソースクロックの生成: メインクロック(MCLK)を PLL/SSCG 逓倍したもの、2 分周したものまたはサブクロック (SBCLK)を選択します。
- ソースクロックの分周: ソースクロックを分周し、各部に供給する動作クロックを生成します。

図 5-1 クロック生成系統図



5.2 特長

クロックの特長について説明します。

- 2 系統のオンチップオシレータを搭載。
- メインクロック(MCLK)は、オンチップ PLL/SSCG により逓倍されます。
- 逓倍クロックは独立した PLL/SSCG から供給されます。
- 各クロックが安定するまで、各クロックはタイマにより供給停止されます。(発振安定待ちタイマ)
- 発振安定待ち終了の割込みを発生させることができます。
- メインクロック発振安定待ちタイマ(メインタイマ)とサブクロック発振安定待ちタイマ(サブタイマ)はメイン・サブの各クロック発振安定後、汎用割込みインターバルタイマとして使用できます。
- リアルタイムクロック用クロックはメインクロック(MCLK)、サブクロック(SBCLK)から選択可能です。
- 100kHz の WDT1 クロック用 CR 発振回路を搭載。同発振回路の設定(補正)については『RTC/WDT1 補正(キャリブレーション)』の章を参照してください。
- CAN プリスケーラ用クロックを生成します。PLL 使用時は PLL クロック(PLLCLK)[非拡散クロック]、それ以外の場合はオンチップバスクロック(HCLK)を使用します。
- ノイズ低減のため、CPU およびリソースのクロックとして SSCG クロック[拡散クロック]を選択できます。

5.3 構成

クロックの構成について説明します。

図 5-2 クロックの接続図 (1)-1 メインクロック生成部

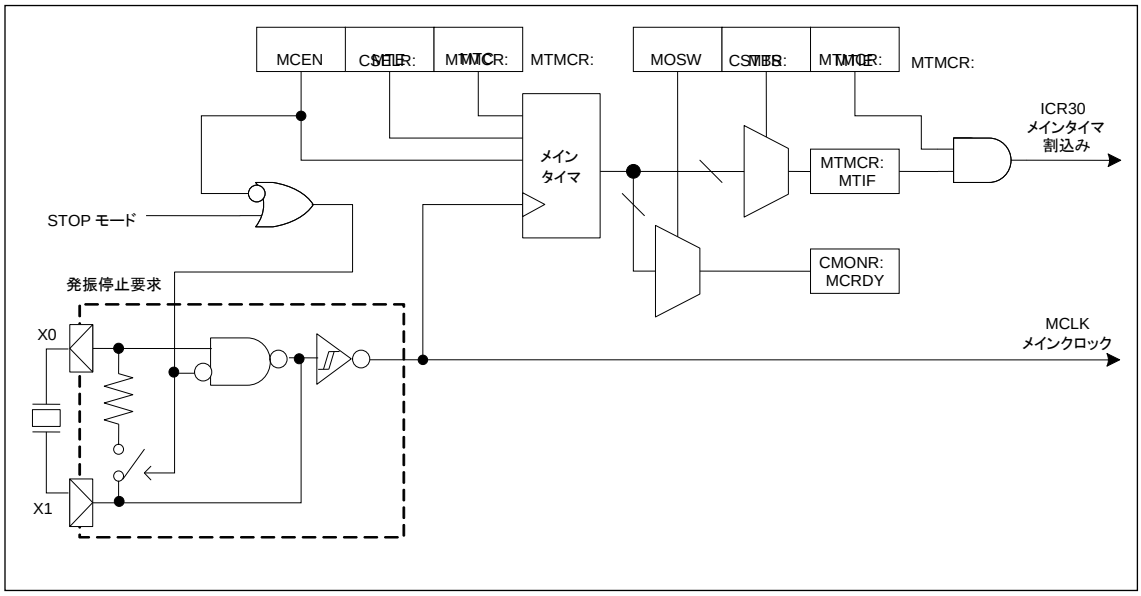


図 5-3 クロックの接続図 (1)-2 サブクロック生成部

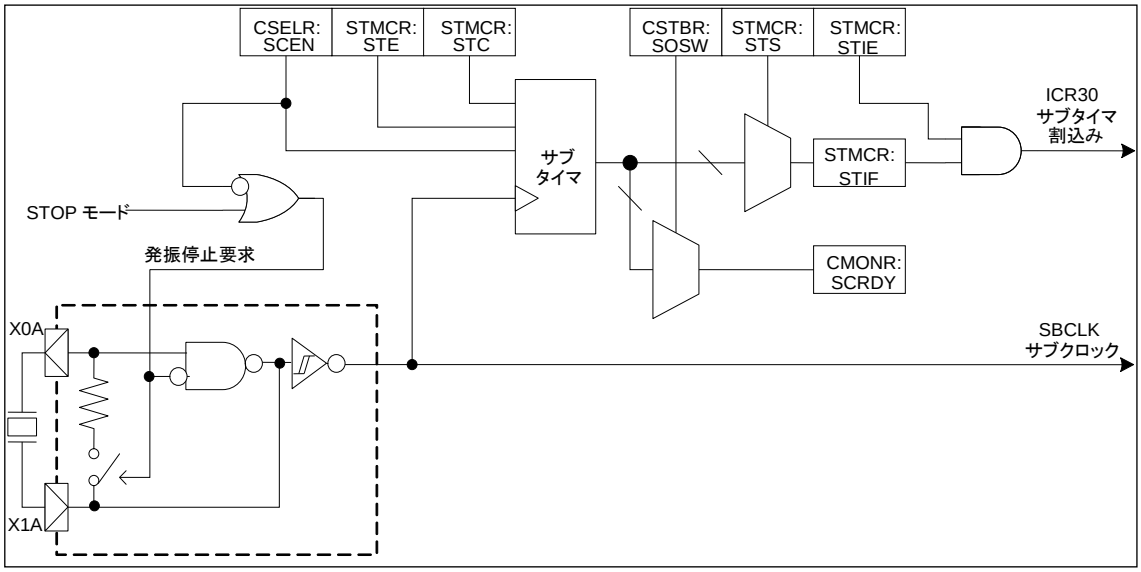


図 5-4 クロックの接続図 (1)-3 PLL クロック生成部

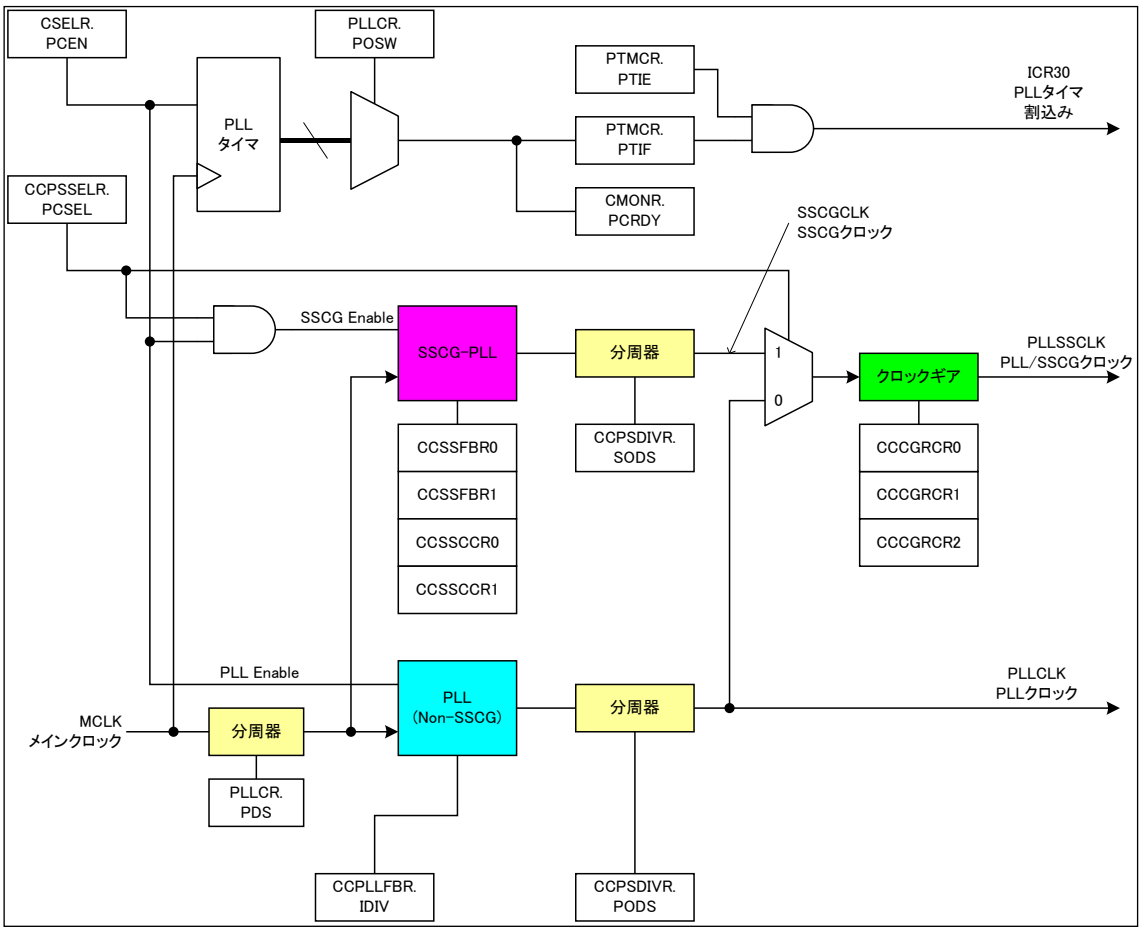


図 5-5 クロックの接続図(2)ソースクロック選択部

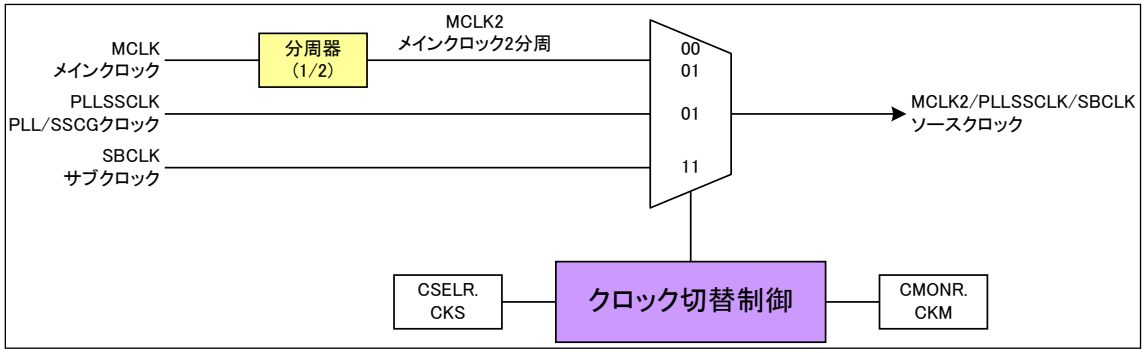


図 5-6 クロックの接続図(3)分周制御

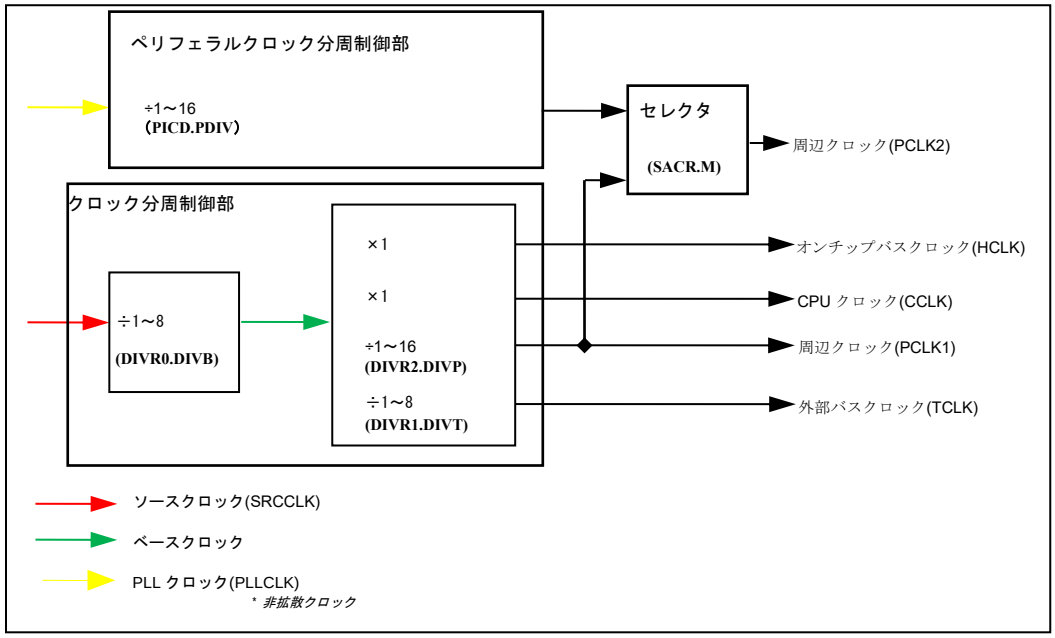


図 5-7 クロックの接続図(4) CAN プリスケーラクロック生成

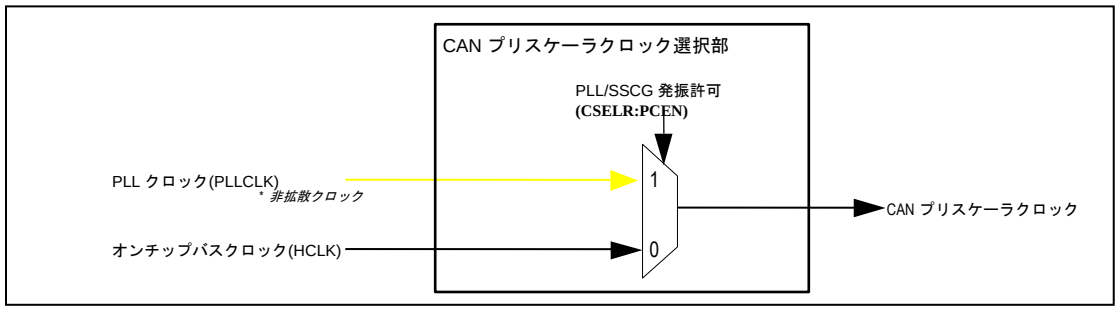


図 5-8 クロックの接続図(6) 時計・パワーマネジメントクロック生成部

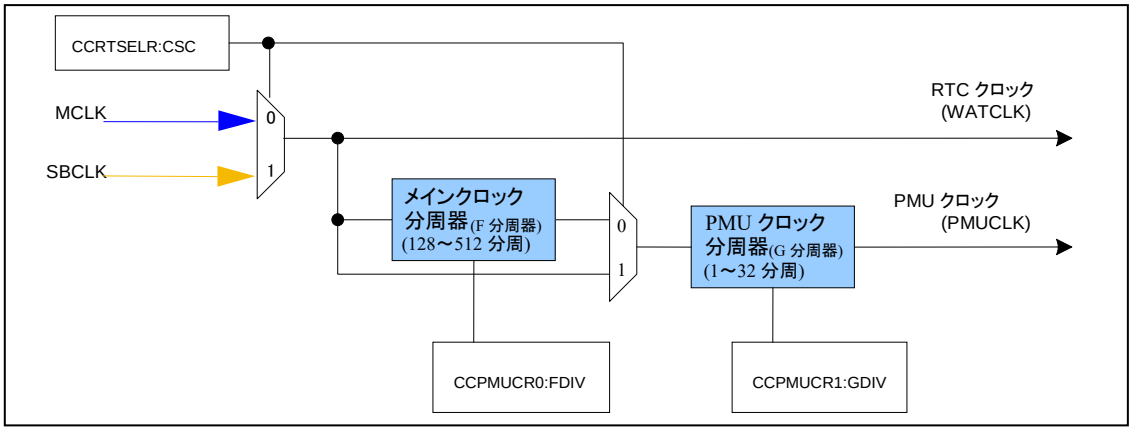
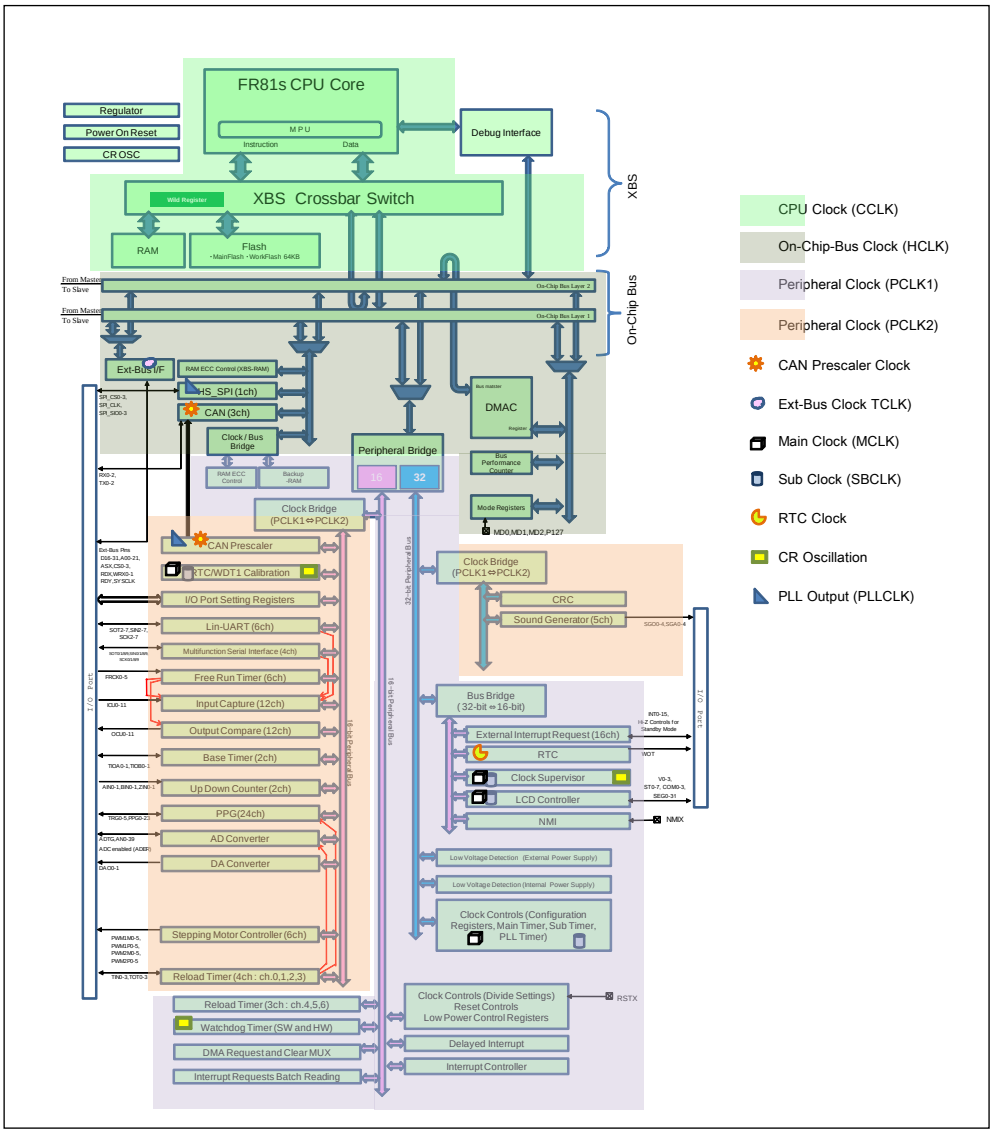


図 5-9 クロック系統図



5.4 レジスタ

クロックのレジスタについて説明します。

表 5-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0488	DIVR0	DIVR1	DIVR2	予約	分周設定レジスタ 0 分周設定レジスタ 1 分周設定レジスタ 2
0x0510	CSELR	CMONR	MTMCR	STMCR	クロックソース設定レジスタ クロックソース監視レジスタ メインタイマ制御レジスタ サブタイマ制御レジスタ
0x0514	PLLCR		CSTBR	PTMCR	PLL 設定レジスタ 発振安定待ち設定レジスタ PLL クロック発振安定待ちタイマ制御レジスタ
0x0520	CCPSSELR	予約	予約	CCPSDIVR	PLL/SSCG クロック選択レジスタ PLL/SSCG 出力クロック分周設定レジスタ
0x0524	予約	CCPLLFBR	CCSSFBR0	CCSSFBR1	PLL フィードバック分周設定レジスタ SSCG フィードバック分周設定レジスタ 0 SSCG フィードバック分周設定レジスタ 1
0x0528	予約	CCSSCCR0	CCSSCCR1		SSCG コンフィグ設定レジスタ 0 SSCG コンフィグ設定レジスタ 1
0x052C	予約	CCCGRCR0	CCCGRCR1	CCCGRCR2	クロックギアコンフィグ設定レジスタ 0 クロックギアコンフィグ設定レジスタ 1 クロックギアコンフィグ設定レジスタ 2
0x0530	CCRTSELR	予約	CCPMUCR0	CCPMUCR1	RTC/PMU クロック選択レジスタ PMU クロック分周設定レジスタ 0 PMU クロック分周設定レジスタ 1
0x0534	予約	予約	予約	予約	予約
0x0538	予約	予約	予約	予約	予約
0x053C	予約	予約	予約	予約	予約
0x1000	SACR	PICD	予約	予約	Sync/Async Control Register Peripheral Interface Clock Divider

5.4.1 分周設定レジスタ 0 : DIVR0 (DIVide clock configuration Register 0)

分周設定レジスタ0のビット構成について示します。

クロックの分周を制御します。

■ DIVR0 : アドレス 0488H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DIVB[2:0]			予約				
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit5] DIVB[2:0] (DIVide ratio of Baseclock) : ベースクロック分周設定

ソースクロック(SRCCLK)からベースクロックを生成する部分での分周を以下のように設定します。
CPUクロック (CCLK), オンチップバスクロック (HCLK) はベースクロックと同一周波数です。

DIVB[2:0]	分周比
000	分周しない(初期値)
001	2 分周
010	3 分周
011	4 分周
100	5 分周
101	6 分周
110	7 分周
111	8 分周

[bit4 ～ bit0] (予約)

5.4.2 分周設定レジスタ 1 : DIVR1 (DIVide clock configuration Register 1)

分周設定レジスタ1のビット構成について示します。

クロックの分周を制御します。

■ DIVR1: アドレス 0489H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TSTP	DIVT[2:0]			予約			
初期値	0	0	0	1	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7] TSTP (TCLK SToP) : **外部バスクロック停止許可**

スリープモード時に、外部バスクロック(TCLK)を停止するかどうかを設定します。

TSTP	スリープモード時の TCLK
0	停止しない (初期値)
1	停止する

[bit6～bit4] DIVT[2:0] (DIVide ratio of TCLK) : **外部バスクロック分周設定**

ベースクロックから外部バスクロック(TCLK)を生成する際の分周比を設定します。

DIVT[2:0]	ベースクロック→TCLK 分周比
000	分周しない
001	2 分周 (初期値)
010	3 分周
011	4 分周
100	5 分周
101	6 分周
110	7 分周
111	8 分周

＜注意事項＞

外部バスクロック (TCLK) は必ず40MHz以下になるように本レジスタを設定してください。

[bit3 ～ bit0] (予約)

5.4.3 分周設定レジスタ 2 : DIVR2 (DIVide clock configuration Register 2)

分周設定レジスタ2のビット構成について示します。

クロックの分周を制御します。

■ DIVR2: アドレス 048AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DIVP[3:0]				予約			
初期値	0	0	1	1	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit4] DIVP[3:0] (DIVide ratio of PCLK) : 周辺クロック分周設定

ベースクロックから周辺クロック (PCLK1) を生成する際の分周比を設定します。

DIVP[3:0]	ベースクロック→PCLK1 分周比
0000	分周しない
0001	2 分周
0010	3 分周
0011	4 分周 (初期値)
0100	5 分周
0101	6 分周
0110	7 分周
0111	8 分周
1000	9 分周
1001	10 分周
1010	11 分周
1011	12 分周
1100	13 分周
1101	14 分周
1110	15 分周
1111	16 分周

＜注意事項＞

周辺クロック (PCLK1) は必ず40MHz以下になるように本レジスタを設定してください。

[bit3 ～ bit0] (予約)

5.4.4 クロックソース設定レジスタ: CSELR (Clock source SELect Register)

クロックソース設定レジスタのビット構成について示します。

各クロックソースの制御，ソースクロック (SRCCLK) の選択を行います。

＜注意事項＞

本レジスタに設定した値、また本レジスタを読み出した値は実際に制御/選択されている状態ではありません。本レジスタに設定した値が実際に反映されたかどうかは、CMONRを読み出すことで確認できます。本レジスタ値とCMONRが同じ値であることを確認した後に、本レジスタを書換えてください。クロック切換え中 (CKS[1:0] ≠CKM[1:0])、本レジスタの書込みは無視されます。

■ CSELR: アドレス 0510H (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SCEN	PCEN	MCEN	予約			CKS[1:0]	
初期値	*	0	1	0	0	0	0	0
属性	R,W	R,W	R,W	R0,WX	R0,WX	R0,WX	R,W	R,W

* 時計モード(電源遮断)からの復帰では初期化しません。それ以外は0に初期化します。

[bit7] SCEN (Sub Clock ENable) : **サブクロック発振許可**

サブクロック(SBCLK)用発振回路を以下のように制御します。

SCEN	サブクロック用発振制御
0	発振を停止する(初期値)
1	発振する

サブクロック (SBCLK) をソースクロック (SRCCLK) として選択している場合、本ビットは書換えることができません。
ストップモード時は、本ビットの値にかかわらずサブクロック用発振回路は停止します。
本ビットが"0"のとき、サブタイマはクリアされます。
クロック1系統品種の場合、本ビットは常に"0"が読み出され、書込みは動作に影響ありません。

＜注意事項＞

スタンバイ時計モード (電源遮断) からの復帰時には、本ビットは初期化されません。また、スタンバイ時計モード (電源遮断) からの復帰中は内部リセットが発行されており、パワーオンリセット/内部低電圧リセット/RSTX・NMIX同時アサートによるリセット以外のリセット要因を受け付けません。

このとき、本ビットは初期化されませんので、立上げ後にRSTX端子入力からのリセット入力または、外部低電圧検出フラグがセットされている場合には、必要があれば本ビットを初期化してください。

[bit6] PCEN (PLL Clock ENable) : PLL発振許可

PLL/SSCGクロック発振回路を以下のように制御します。

PCEN	PLL/SSCG クロック(PLLSSCLK)用発振制御
0	発振を停止する (初期値)
1	発振する

PLL/SSCGクロック (PLLSSCLK) をソースクロック (SRCCLK) として選択している場合、本ビットは書き換えることができません。メインクロック発振停止またはメインクロック発振安定待ち中 (CMONR.MCRDY=0) の場合、本ビットは書き換えることができません。
ストップモードにする前に、本ビットを"0"にしてください。
MCENビットを"0"に書き換えると、本ビットも"0"になります。

＜注意事項＞

MDI高速通信中、本ビットの値にかかわらず、PLLは発振許可状態となります。

[bit5] MCEN (Main Clock ENable) : メインクロック発振許可

メインクロック用発振回路を以下のように制御します。

MCEN	メインクロック用発振制御
0	発振を停止する
1	発振する(初期値)

メインクロック (MCLK) またはPLL/SSCGクロック (PLLSSCLK) をソースクロック (SRCCLK) として選択している場合、本ビットは書き換えることができません。
ストップモード時は、本ビットの値にかかわらずメインクロック用発振回路は停止します。
本ビットが"0"のとき、メインタイマがクリアされます。

＜注意事項＞

MDI低速通信中、本ビットの値にかかわらず、メインクロックは発振許可状態となります。

[bit4～bit2] (予約)

[bit1, bit0] CKS[1:0] (ClocK Select) : ソースクロック選択

ソースクロック (SRCCLK) を以下に示すように選択します。

CKS[1:0]	ソースクロック選択
00	メインクロック (MCLK) の 2 分周 (初期値)
01	メインクロック (MCLK) の 2 分周
10	PLL/SSCG クロック (PLLSSCLK)
11	サブクロック (SBCLK)

ただしCKS[1:0]≠CKM[1:0]の場合、本ビットを書き換えることはできません。また本ビットで切り換えようとするクロックの発振が停止または安定待ち中 (CMONR:xCRDY=0) の場合、本ビットは書き換えることができません。
PLL/SSCGクロック (PLLSSCLK) からサブクロック (SBCLK)、サブクロック (SBCLK) からPLL/SSCGクロック (PLLSSCLK) へは直接切り換えることができません。
本ビットを変更できる組み合わせは以下のとおりです。

変更前の CKS の値	書換え可能な値	書換え条件	書換え不可な値
00	00, 01	MCRDY=1	11
	10	PCRDY=1	
01	00, 01	MCRDY=1	10
	11	SCRDY=1	
10	00	MCRDY=1	01,11
	10	PCRDY=1	
11	01	MCRDY=1	00,10
	11	SCRDY=1	

書換え不可能な値を書込みしないでください。

5.4.5 クロックソース監視レジスタ: CMONR (Clock source MONitor Register)

クロックソース監視レジスタのビット構成について示します。

各クロックソースの状態、ソースクロック (SRCCLK) を表示します。

本レジスタを読み出すことにより、CSELRに設定された値が実際の状態に反映されているかどうかを確認することができます。

＜注意事項＞

CSELR設定値を変更した場合は、必ず本レジスタを読出し、CSELRに設定した値と同じ値を示すまで、次のCSELR設定値変更は行わないでください。

■ CMONR: アドレス 0511H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SCRDY	PCRDY	MCRDY	予約			CKM[1:0]	
初期値	*	0	1	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX

*時計モード (電源遮断) からの復帰では初期化しません。それ以外は0に初期化します。

[bit7] SCRDY (Sub Clock ReaDY) : サブクロックレディ

サブクロック (SBCLK) の状態を以下のように示します。

SCRDY	サブクロック (SBCLK) の状態
0	発振が停止しているか、発振安定待ち状態である
1	安定発振中であり、ソースクロックとして使用可能である

本ビットが"0"のときは、サブクロック (SBCLK) をソースクロック (SRCCLK) として選択できません。

＜注意事項＞

SCEN=1→0に変更直後、SCRDY=1が読み出される場合があります。

[bit6] PCRDY (PLL Clock ReaDY) : PLLクロックレディ

PLL/SSCGクロック (PLLSSCLK) の状態を以下のように示します。

PCRDY	PLL/SSCG クロック (PLLSSCLK) の状態
0	発振が停止しているか、発振安定待ち状態である
1	安定発振中であり、ソースクロックとして使用可能である

本ビットが"0"のときは、PLL/SSCGクロック (PLLSSCLK) をソースクロック (SRCCLK) として選択できません。

＜注意事項＞

PCEN=1→0に変更直後、PCRDY=1が読み出される場合があります。
MDI高速通信中、本ビットの値にかかわらず、PLLは発振許可状態となります。

[bit5] MCRDY (Main Clock ReaDY) : **メインクロックレディ**

メインクロック(MCLK)の状態を以下のように示します。

MCRDY	メインクロック (MCLK) の状態
0	発振が停止しているか、発振安定待ち状態である
1	安定発振中であり、ソースクロックとして使用可能である

本ビットが"0"のときは、メインクロック(MCLK)またはPLL/SSCGクロック(PLLSSCLK)をソースクロック(SRCCLK)として選択できません。
本ビットの初期値が"1"であるのは、パワーオンリセット後の初のリセットベクタフェッチ時点で発振安定中であるという意味であり、パワーオンリセット直後に既に発振安定中であるという意味ではありません。

＜注意事項＞

MCEN=1→0に変更直後、MCRDY=1が読み出される場合があります。
MDI高速通信中、本ビットの値にかかわらず、メインクロックは発振許可状態となります。

[bit4～bit2] (予約)

[bit1, bit0] CKM[1:0] (ClocK Monitor) : **ソースクロック表示**

現在選択中のソースクロック (SRCCLK)を示します。

CKM[1:0]	ソース選択
00	メインクロック (MCLK) の 2 分周
01	メインクロック (MCLK) の 2 分周
10	PLL/SSCG クロック (PLLSSCLK)
11	サブクロック (SBCLK)

5.4.6 メインタイマ制御レジスタ: MTMCR (Main clock TiMer Control Register)

メインタイマ制御レジスタのビット構成について示します。

メインクロック(MCLK)で動作するメインタイマの制御を行います。

■ MTMCR: アドレス 0512H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MTIF	MTIE	MTC	MTE	MTS[3:0]			
初期値	0	0	0	0	1	1	1	1
属性	R(RM1),W	R/W	R(RM0),W	R/W	R1,WX	R/W	R/W	R/W

メインタイマはメインクロック (MCLK) の発振安定待ち時間生成に使用されるため、メインクロックの発振が安定した後のみ使用できます。

メインクロック発振停止中 (MCEN=0) またはストップモード中は、メインタイマはクリアされます。

メインタイマの動作が許可されていない時 (MTE=0) は、メインクロック発振安定待ち時以外、メインタイマは停止します。MTIEビットを除き、MCRDY=1のときのみ本レジスタの書込みが有効になります。そのため、メインクロック発振安定待ち時 (MCEN=1かつMCRDY=0) のMTC=1によるメインタイマのクリアは無効です。

メインタイマ停止時(MTE=0)、メインタイマはクリアされ、クリア中はMTC=1が読み出されます。

そのとき、メインタイマ割込みフラグ (MTIF) はセットされません。メインタイマオーバフロー周期 (MTS[3:0]) の変更はメインタイマ停止 (MTE=0) 期間中に行ってください。

MTE=1→0書換え時、MTC=0となるまでメインタイマは動作します。その間に、メインタイマ割込みフラグが"1"になる可能性があります。MTC=1書込み時、MTC=0となるまでメインタイマは動作します。その間に、メインタイマ割込みフラグが"1"になる可能性があります。MTE=0→1書換えと同時にMTC=1書込みをすると、クリアしてから動作開始となり、スタートが遅れます。

[bit7] MTIF (Main clock Timer Interrupt Flag) : **メインタイマ割込みフラグ**

メインタイマの選択した周期でのオーバフローが発生したことを示すフラグです。
MTIEビットが"1"のとき、本ビットがセットさせるとメインタイマ割込み要求が発生します。

クリア要因	・"0"書込み ・メインタイマ割込みによる DMA 転送の発生
セット要因	・MTS[3:0]で設定された周期でのオーバフロー発生 ・MCEN=0→1 後のメインクロックの発振安定待ち時間の終了 ・ストップモード解除時のメインクロック(MCLK)の発振安定待ち時間の終了 (SINIT によるリセット後の発振安定待ち時間の終了時はセットされません)

本ビットへの"1"書込みは無効です。

MTIEビットが"0"のとき、DMA転送による本ビットのクリアは行われません。

リードモディファイライト命令では、"1"がリードされます。

セット要因とクリア要因が同時に発生した場合、セット要因が優先されます。

スタンバイモード(電源遮断)からの復帰中は、内部リセットが発行されており、メインタイマ割込みフラグはセットされません。

[bit6] MTIE (Main clock Timer Interrupt Enable) : **メインタイマ割込み許可**

メインタイマのオーバフローによる割込みを以下に示すように制御します。

MTIE	メインタイマ割込み
0	割込み禁止(初期値)
1	割込み許可(MTIF ビットが"1"のとき、割込み要求を出力)

[bit5] MTC (Main clock Timer Clear) : **メインタイマクリア**

メインタイマをクリアします。

MTC	書込み時
0	何もしない
1	メインタイマをクリアする

MTC	読出し時
0	通常動作中
1	メインタイマクリア中

"1"書込み後、自動的に"0"に戻ります。

リードモディファイライト命令では、"0"がリードされます。

MTC=1のとき、MTC=1を書き込むと、2回目の書込みは無視されます。

[bit4] MTE (Main clock Timer Enable) : **メインタイマ動作許可**

メインタイマの動作を以下に示すように制御します。

MTE	メインタイマ動作
0	動作禁止 (初期値)
1	動作許可

MTC=1のとき、MTE=1書込みは禁止です。

PLL/SSCGクロック発振安定待ちを行う場合、必ず本ビットを"0"にして、メインタイマを停止してください。

[bit3～bit0] MTS[3:0] (Main clock Timer interval Select) : **メインタイマ周期選択**

メインタイマのオーバフロー周期を以下に示すように選択します。

MTS[3:0]	メインタイマオーバフロー周期	4MHz 時
1000	2^9 × メインクロック周期	128.0[µs]
1001	2^{10} × メインクロック周期	256.0[µs]
1010	2^{11} × メインクロック周期	512.0[µs]
1011	2^{12}× メインクロック周期	1024.0[µs]
1100	2^{13}× メインクロック周期	2048.0[µs]
1101	2^{14}× メインクロック周期	4096.0[µs]
1110	2^{15}× メインクロック周期	8192.0[µs]
1111	2^{16} × メインクロック周期(初期値)	16384.0[µs]

MTS[3]ビットは常に"1"が読み出されます。

MTS[3:0]の変更はメインタイマ停止 (MTE=0) 期間中に行ってください。

5.4.7 サブタイマ制御レジスタ: STMCR (Sub clock TiMer Control Register)

サブタイマ制御レジスタのビット構成について示します。

サブクロックで動作するサブタイマの制御をします。

■ STMCR: アドレス 0513H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	STIF	STIE	STC	STE	予約	STS[2:0]		
初期値	0	0	0	0	0	1	1	1
属性	R(RM1),W	R/W	R(RM0),W	R/W	R0,WX	R/W	R/W	R/W

サブタイマはサブクロック (SBCLK) の発振安定待ち時間生成に使用されるため、サブクロックの発振が安定した後のみ使用できます。

サブクロック発振停止中 (SCEN=0) またはストップモード中は、サブタイマはクリアされます。

サブタイマの動作が許可されていない時 (STE=0) は、サブクロック発振安定待ち時間以外、サブタイマは停止します。STIEを除き、SCRDY=1のときのみ、本レジスタの書込みが有効になります。そのため、サブクロック発振安定待ち時(SCEN=1かつSCRDY=0)のSTC=1によるサブタイマクリアは無効です。

サブタイマ停止時 (STE=0)、サブタイマはクリアされ、クリア中はSTC=1が読み出されます。そのとき、サブタイマ割込みフラグはセットされません。サブタイマオーバフロー周期 (STS[2:0]) の変更はサブタイマ停止 (STE=0) 期間中に行なってください。

STE=1→0書換え時、STC=0となるまでサブタイマは動作します。その間に、サブタイマ割込みフラグが"1"になる可能性があります。STC=1書込み時、STC=0となるまでサブタイマは動作します。その間に、サブタイマ割込みフラグが"1"になる可能性があります。STE=0→1書換えと同時にSTC=1書込みをすると、クリアしてから動作開始となり、スタートが遅れます。

[bit7] STIF (Sub clock Timer Interrupt Flag) : **サブタイマ割込みフラグ**

サブタイマの選択した周期でのオーバフローが発生したことを示すフラグです。
STIEビットが"1"のとき、本ビットがセットされるとサブタイマ割込み要求が発生します。

クリア要因	・"0"書込み ・サブタイマ割込みによる DMA 転送の発生
セット要因	・STS[2:0]で設定された周期でのオーバフロー発生 ・SCEN=0→1 後のサブクロックの発振安定待ち時間の終了 ・ストップモード解除時のサブクロックの発振安定待ち時間の終了

本ビットへの"1"書込みは無効です。

STIEビットが"0"のとき、DMA転送による本ビットのクリアは行われません。

リードモディファイライト命令では、"1"がリードされます。

セット要因とクリア要因が同時に発生した場合、セット要因が優先されます。

スタンバイモード (電源遮断) からの復帰中は、内部リセットが発行されており、サブタイマ割込みフラグはセットされません。

[bit6] STIE (Sub clock Timer Interrupt Enable) : **サブタイマ割込み許可**

サブタイマのオーバフローによる割込みを以下に示すように制御します。

STIE	サブタイマ割込み
0	割込み禁止 (初期値)
1	割込み許可(STIF ビットが"1"のとき、割込み要求を出力)

[bit5] STC (Sub clock Timer Clear) : **サブタイマクリア**

サブタイマをクリアします。

STC	書込み時
0	何もしない
1	サブタイマをクリアする

STC	読出し時
0	通常動作中
1	サブタイマクリア中

"1"書込み後、自動的に"0"に戻ります。

リードモディファイライト命令では、"0"がリードされます。

STC=1のとき、STC=1を書き込むと、2回目の書込みは無視されます。

[bit4] STE (Sub clock Timer Enable) : **サブタイマ動作許可**

サブタイマの動作を以下に示すように制御します。

STE	サブタイマ動作
0	動作禁止 (初期値)
1	動作許可

STC=1のとき、STE=1書込みは禁止です。

[bit3] (予約)

[bit2～bit0] STS[2:0] (Sub clock Timer interval Select) : **サブタイマ周期選択**

サブタイマのオーバフロー周期を選択します。

STS[2:0]	サブタイマオーバフロー周期	32kHz 時
000	2^8× サブクロック周期	8[ms]
001	2^9× サブクロック周期	16[ms]
010	2^{10}× サブクロック周期	32[ms]
011	2^{11}× サブクロック周期	64[ms]
100	2^{12}× サブクロック周期	128[ms]
101	2^{13}× サブクロック周期	0.256[s]
110	2^{14}× サブクロック周期	0.512[s]
111	2^{15}× サブクロック周期(初期値)	1.024[s]

5.4.8 PLL 設定レジスタ ：PLLCR (PLL Configuration Register)

PLL設定レジスタのビット構成について示します。

PLL/SSCGクロック発振回路内の逓倍率や分周比，発振安定待ち時間を設定します。

■ PLLCR: アドレス 0514H (アクセス: バイト，ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約		予約	予約				
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,WX	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	POSW[3:0]				PDS[3:0]			
初期値	1	1	1	1	0	0	0	0
属性	R1,WX	R,W	R,W	R,W	R,W	R,W	R,W	R,W

メインクロック (MCLK) からPLL/SSCGクロック (PLLSSCLK) を生成するPLL/SSCGクロック発振回路内のパラメータを設定します。

PLL/SSCGクロック発振許可中 (CSELR:PCEN=1) のとき、本レジスタへの書込みは無効です。

[bit15, bit14] 予約

必ず"0"を書き込んでください。

[bit13] (予約)

[bit12～bit8] 予約

必ず"0"を書き込んでください。

[bit7～bit4] POSW[3:0] (Pll clock clock OSc Wait) : PLLクロック発振安定待ち選択

PLL/SSCGクロック (PLLSSCLK の発振安定待ち時間を以下に示すように選択します。

POSW[3:0]	PLL/SSCG クロック発振安定待ち時間	4MHz 時	8MHz 時
1000	2^9× メインクロック周期	128.0[µs]	64.0[µs]
1001	2^{10}× メインクロック周期	256.0[µs]	128.0[µs]
1010	2^{11}× メインクロック周期	512.0[µs]	256.0[µs]
1011	2^{12}× メインクロック周期	1024.0[µs]	512.0[µs]
1100	2^{13}× メインクロック周期	2048.0[µs]	1024.0[µs]
1101	2^{14}× メインクロック周期	4096.0[µs]	2048.0[µs]
1110	2^{15}× メインクロック周期	8192.0[µs]	4096.0[µs]
1111	2^{16}× メインクロック周期(初期値)	16384.0[µs]	8192.0[µs]

POSW3は常に"1"が読み出されます。

＜注意事項＞

本シリーズのPLL/SSCGクロックの発振安定待ち時間仕様は200[µs]となっています。以下どちらかの方法で200[µs]以上の待ち時間を確保してください。

- 256[µs] 以上の POSW[3:0]を選択
- POSW[3:0]の設定値にかかわらず、ソフトウェア処理で 200[µs]以上の待ち時間を確保

[bit3～bit0] PDS[3:0] (Pll input clock Divider Select) : PLL入力クロック分周選択

PLLおよびSSCG入力クロックのメインクロック(MCLK)分周を以下に示すように選択します。

PDS[3:0]	PLL および SSCG 入力クロック分周選択
0000	PLL および SSCG 入力クロック = メインクロック/ 1
0001	PLL および SSCG 入力クロック = メインクロック/ 2
0010	PLL および SSCG 入力クロック = メインクロック/ 3
0011	PLL および SSCG 入力クロック = メインクロック/ 4
0100	PLL および SSCG 入力クロック = メインクロック/ 5
0101	PLL および SSCG 入力クロック = メインクロック/ 6
0110	PLL および SSCG 入力クロック = メインクロック/ 7
0111	PLL および SSCG 入力クロック = メインクロック/ 8
1000	PLL および SSCG 入力クロック = メインクロック/ 9
1001	PLL および SSCG 入力クロック = メインクロック/ 10
1010	PLL および SSCG 入力クロック = メインクロック/ 11
1011	PLL および SSCG 入力クロック = メインクロック/ 12
1100	PLL および SSCG 入力クロック = メインクロック/ 13
1101	PLL および SSCG 入力クロック = メインクロック/ 14
1110	PLL および SSCG 入力クロック = メインクロック/ 15
1111	PLL および SSCG 入力クロック = メインクロック/ 16

PLL/SSCGおよびシステムのスペックに合わせて、設定してください。
設定例に関しては「5.5.1.3 PLL/SSCGクロック (PLLSSCLK)」を参照してください。

設定値の制限があります。設定時には、「5.5.1.4 PLL/SSCGクロック使用時の制限事項」を参照してください。

5.4.9 発振安定待ち設定レジスタ ：CSTBR (Clock STaBilization select Register)

発振安定待ち設定レジスタのビット構成について示します。

各クロックソースの発振安定待ちを設定します。

本レジスタで設定した発振安定待ち時間は、ストップ/時計モードからの復帰時や、ソースクロック(SRCCLK)として選択していないクロックを発振許可してからそのクロックのレディ(CMONR: *CRDY)が"1"となるまでに使用されます。リセット時に発振安定待ちが必要な場合、必ず本レジスタの初期値で選択される安定待ち時間となります。メインクロック発振安定待ち時間中は(MCEN=1かつMCRDY=0)、MOSW[3:0]への書込みは無効です。サブクロック発振安定待ち時間中は(SCEN=1かつSCRDY=0)、SOSW[2:0]への書込みは無効です。

■ CSTBR: アドレス 0516H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	SOSW[2:0]			MOSW[3:0]			
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R,W	R,W	R,W	R,W	R,W	R,W	R,W

[bit7] (予約)

[bit6～bit4] : SOSW[2:0] (Sub clock OSc Wait) : サブクロック発振安定待ち選択

サブクロック(SBCLK)の発振安定待ち時間を以下に示すように選択します。

SOSW[2:0]	サブクロック発振安定待ち時間	32kHz 時
000	2^8× サブクロック周期(初期値)	8[ms]
001	2^9 × サブクロック周期	16[ms]
010	2^{10}× サブクロック周期	32[ms]
011	2^{11}× サブクロック周期	64[ms]
100	2^{12}× サブクロック周期	128[ms]
101	2^{13}× サブクロック周期	0.256[s]
110	2^{14}× サブクロック周期	0.512[s]
111	2^{15}× サブクロック周期	1.024[s]

[bit3～bit0] MOSW[3:0] (Main clock OSc Wait) : **メインクロック発振安定待ち選択**

MOSW[3:0]の設定値により、メインタイマの周期が設定されます。
メインクロック(MCLK)の発振安定待ち時間を以下に示すように選択します。

MOSW[3:0]	メインクロック発振安定待ち時間	4MHz 時
0000	2^{15}×メインクロック周期(初期値)	8[ms]
0001	2^{1}×メインクロック周期	500[ns]
0010	2^{5}×メインクロック周期	8[us]
0011	2^{6}×メインクロック周期	16[us]
0100	2^{7}×メインクロック周期	32[us]
0101	2^{8}×メインクロック周期	64[us]
0110	2^{9}×メインクロック周期	128[us]
0111	2^{10}×メインクロック周期	256[us]
1000	2^{11}×メインクロック周期	512[us]
1001	2^{12}×メインクロック周期	1[ms]
1010	2^{13}×メインクロック周期	2[ms]
1011	2^{14}×メインクロック周期	4[ms]
1100	2^{17}×メインクロック周期	33[ms]
1101	2^{19}×メインクロック周期	131[ms]
1110	2^{21}×メインクロック周期	524[ms]
1111	2^{23}×メインクロック周期	2[s]

＜注意事項＞

クロックスーパバイザ機能が有効な場合、本レジスタの設定周期よりも断検出の周期が短いと、発振安定待ち中に断検出されてしまうので、注意してください。

5.4.10 PLL クロック発振安定待ちタイマ設定レジスタ ：PTMCR: (PLL clock osc TiMer Control Register)

PLLクロック発振安定待ちタイマ設定レジスタのビット構成について示します。

PLL/SSCGクロック発振安定待ちを行うメインクロックで動作するタイマの制御を行います。

PLL/SSCGクロック発振安定待ちタイマはPLL/SSCGクロック(PLLSSCLK)の発振安定待ち時間にのみ使用されます。

PLL/SSCGクロック発振安定待ち時間はPLLCR:POSW[3:0]で設定した時間になります。

PLL/SSCGクロック発振安定待ちタイマはPLL/SSCGクロック発振許可(CSELR:PCEN = 1)されてから動作を開始し、発振安定待ち時間動作した後、停止します。また、PLL/SSCGクロック発振停止(CSELR:PCEN = 0)されるとクリアされます。

■ PTMCR: アドレス 0517H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PTIF	PTIE	予約					
初期値	0	0	0	0	0	0	0	0
属性	R(RM1),W	R/W	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] PTIF (Pll clock osc wait Timer Interrupt Flag) : PLLクロック発振安定待ちタイマ割込みフラグ

PLLクロック発振安定待ち選択(PLLCR:POSW[3:0])で設定した時間でのオーバフローが発生したことを示すフラグです。PTIEビットが"1"のとき、本ビットがセットされるとPLL/SSCGクロック発振安定待ちタイマ割込み要求が発生します。

クリア要因	・"0"書込み ・PLL/SSCG クロック発振安定待ちタイマによる DMA 転送の発生
セット要因	・PCEN=0→1 後の PLL/SSCG クロック発振安定待ちクロックの発振安定待ち時間の終了

本ビットへの"1"書込みは無効です。

PTIEビットが"0"のとき、DMA転送による本ビットのクリアは行われません。

リードモディファイライト命令では、"1"がリードされます。

セット要因とクリア要因が同時に発生した場合、セット要因が優先されます。

[bit6] PTIE (Pll clock osc wait Timer Interrupt Enable) : PLLクロック発振安定待ちタイマ割込み許可

PLL/SSCGクロック発振安定待ちタイマのオーバフローによる割込みを以下に示すように制御します。

PTIE	動作
0	割込み禁止(初期値)
1	割込み許可(PTIF ビットが"1"のとき、割込み要求を出力)

[bit5 ～ bit0] (予約)

5.4.11 PLL/SSCG クロック選択レジスタ ：CCPSSELR (CCtl Pll/Sscg clock SELect Register)

PLL/SSCGクロック選択レジスタのビット構成について示します。

PLLもしくはSSCGのどちらを使用するか選択するレジスタです。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN =0)のときのみ書込み可能です。

■ CCPSSELR: アドレス 0520H (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							PCSEL
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit7～bit1] (予約)

[bit0] PCSEL (Pll Clock source SELect) : PLL/SSCGクロックソース選択

PLL/SSCGクロックのソースを選択します。

PCSEL	PLL または SSCG
0	PLL を選択
1	SSCG を選択

＜注意事項＞

PCSEL=0の場合、SSCGは(未使用のため)常にリセット状態となります。
PCSEL=1の場合でも、CANおよびOCDUにはPLLクロックが供給されます。

5.4.12 PLL/SSCG 出力クロック分周設定レジスタ ：CCPSDIVR (CCtl Pll/Sscg clock DIVide Register)

PLL/SSCG出力クロック分周設定レジスタのビット構成について示します。

PLL/SSCGクロックの分周比を設定するレジスタです。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN = 0)のときのみ書込み可能です。

■ CCPSDIVR: アドレス 0523H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	PODS[2:0]			予約	SODS[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R/W	R/W	R/W	R0,WX	R/W	R/W	R/W

[bit7] (予約)

[bit6～bit4] PODS[2:0] (Pll Oscillator Divider Select) : PLLマクロ発振クロック分周比選択

PLLクロックの分周比を設定します。

PODS[2:0]	分周比設定
000	PLL クロック＝PLL マクロ発振クロック/2
001	PLL クロック＝PLL マクロ発振クロック/4
010	PLL クロック＝PLL マクロ発振クロック/6
011	PLL クロック＝PLL マクロ発振クロック/8
100	PLL クロック＝PLL マクロ発振クロック/10
101	PLL クロック＝PLL マクロ発振クロック/12
110	PLL クロック＝PLL マクロ発振クロック/14
111	PLL クロック＝PLL マクロ発振クロック/16

＜注意事項＞

本ビットでの設定では偶数分周のみです。奇数分周は設定できません。
出力するクロックのDutyは50%になります。
必ずPLLクロックが80MHz以下になるように設定してください。(80MHzを超える周波数での動作保証は行っておりません。)

[bit3] (予約)

[bit2～bit0] SODS[2:0] (Sscg Oscillator Divider Select) : SSCGマクロ発振クロック分周比選択

SSCGクロックの分周比を設定します。

SODS[2:0]	分周比設定
000	SSCG クロック＝SSCG マクロ発振クロック/2
001	SSCG クロック＝SSCG マクロ発振クロック/4
010	SSCG クロック＝SSCG マクロ発振クロック/6
011	SSCG クロック＝SSCG マクロ発振クロック/8
100	SSCG クロック＝SSCG マクロ発振クロック/10
101	SSCG クロック＝SSCG マクロ発振クロック/12
110	SSCG クロック＝SSCG マクロ発振クロック/14
111	SSCG クロック＝SSCG マクロ発振クロック/16

＜注意事項＞

本ビットでの設定では偶数分周のみです。奇数分周は設定できません。
出力するクロックのDutyは50%になります。
必ずSSCGクロックが80MHz以下になるように設定してください。(80MHzを超える周波数での動作保証は行っておりません。)

設定値の制限があります。設定時には、「5.5.1.4 PLL/SSCGクロック使用時の制限事項」を参照してください。

5.4.13 PLL フィードバック分周設定レジスタ ：CCPLLFBR (CCtl PLL FB clock divide Register)

PLLフィールド分周設定レジスタのビット構成について示します。

PLLの逓倍率を設定するレジスタです。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN = 0)のときのみ書込み可能です。

■ CCPLLFBR: アドレス 0525H (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	IDIV[6:0]						
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7] (予約)

[bit6～bit0] IDIV[6:0] (pll feedback Input DIVider ratio settings) : PLLマクロFB入力分周比設定

PLLの逓倍率を設定します。

IDIV[6:0]	分周比設定
0000000～ 0001011	設定禁止
0001100	13
0001101	14
0001110	15
…	……
1100010	99
1100011	100
1100100～ 1111111	設定禁止

設定値の制限があります。設定時には、「5.5.1.4 PLL/SSCGクロック使用時の制限事項」を参照してください。

5.4.14 SSCG フィードバック分周設定レジスタ 0 : CCSSFBR0 (CCtl SScg FB clock divide Register 0)

SSCGフィードバック分周設定レジスタ0のビット構成について示します。

SSCGの逓倍率Nを設定するレジスタです。SSCGの逓倍率はCCSSFBR1の設定と合わせてP×Nとなります。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN = 0)のときのみ書込み可能です。

■ CCSSFBR0: アドレス 0526H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		NDIV[5:0]					
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W

[bit7, bit6] (予約)

[bit5～bit0] NDIV[5:0] (sscg feedback input N-DIVider ratio settings) : SSCGマクロFB入力N分周比設定

SSCGの逓倍率Nを設定します。

NDIV[5:0]	分周比設定
000000	設定禁止
000001	2
000010	3
000011	4
…	……
111101	62
111110	63
111111	設定禁止

設定値の制限があります。設定時には、「5.5.1.4 PLL/SSCGクロック使用時の制限事項」を参照してください。

5.4.15 SSCG フィードバック分周設定レジスタ 1 : CCSSFBR1 (CCtl SScg FB clock divide Register 1)

SSCGフィールドバック分周設定レジスタ1のビット構成について示します。

SSCGの逓倍率Pを設定するレジスタです。SSCGの逓倍率はCCSSFBR0の設定と合わせてP×Nとなります。

本レジスタはPLL/SSCGクロック発振停止(CSELR.PCEN = 0)のときのみ書込み可能です。

■ CCSSFBR1: アドレス 0527H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			PDIV[4:0]				
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W

[bit7～bit5] (予約)

[bit4～bit0] PDIV[4:0] (sscg feedback input P-DIVider ratio settings) : SSCGマクロFB入力P分周比設定

SSCGの逓倍率Pを設定します。

PDIV[4:0]	分周比設定
00000	1
00001	2
00010	3
00011	4
…	……
11101	30
11110	31
11111	設定禁止

設定値の制限があります。設定時には、「5.5.1.4 PLL/SSCGクロック使用時の制限事項」を参照してください。

5.4.16 SSCG コンフィグ設定レジスタ 0 : CCSSCCR0 (CCtl SSCg Config. Register 0)

SSCGコンフィグ設定レジスタ0のビット構成について示します。

SSCGの各種設定を行います。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN = 0)のときのみ書込み可能です。

■ CCSSCCR0: アドレス 0529H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				SFREQ[1:0]		SMODE	SSEN
初期値	0	0	0	1	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W

[bit7～bit4] (予約)

[bit3, bit2] SFREQ[1:0] (Spread spectrum modulation FREQuency settings) : 拡散モジュレーション周波数設定

SSCGの拡散変調周波数を設定します。

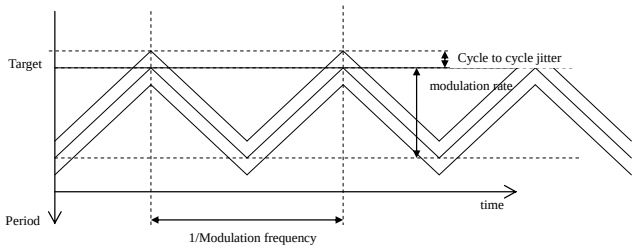
SFREQ[1:0]	モジュレーション周波数
00	1/1024
01	1/2048
1x	1/4096

[bit1] SMODE (Spread spectrum modulation MODE settings) : 拡散モジュレーションモード選択

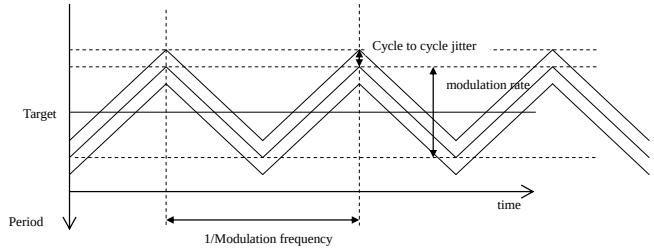
SSCGの拡散変調モードを設定します。

SMODE	モジュレーションモード
0	Down Spread
1	Center Spread

● Down Spread



● Center Spread



[bit0] SSEN (Spread Spectrum ENable) : **拡散スペクトル許可**

SSCGの拡散スペクトルを有効にします。

SSEN	拡散スペクトル許可
0	拡散スペクトラム無効
1	拡散スペクトラム有効

＜注意事項＞

SSENを無効に設定すると、CCSSCCR1:RATESELの設定にかかわらず拡散率は0% になります。

5.4.17 SSCG コンフィグ設定レジスタ 1 : CCSSCCR1 (CCtl SSCg Config. Register 1)

SSCGコンフィグ設定レジスタ1のビット構成について示します。

SSCGの各種設定を行います。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN = 0)のときのみ書込み可能です。

■ CCSSCCR1: アドレス 052A$_H$ (アクセス: ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	RATESEL[2:0]			予約				
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R0,WX	R0,WX	R0,WX	R/W0	R/W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0	R/W0

[bit15～bit13] RATESEL[2:0] (spread spectrum modulation RATE SELection) : 拡散モジュレーションレート選択

SSCGの拡散変調率を設定します。

RATESEL[2:0]	モジュレーションレート
00x	0.5%
010	1%
011	2%
100	3%
101	4%
110	5%
111	設定禁止

[bit12～bit10] (予約)

書込みは効果ありません。

[bit9～bit0] (予約)

必ず"0"を書き込んでください。

5.4.18 クロックギアコンフィグ設定レジスタ 0 : CCCGRCR0 (CCtl Clock Gear Config. Register 0)

クロックギアコンフィグ設定レジスタ0のビット構成について示します。

クロックギアの各種設定を行います。

■ CCCGRCR0: アドレス 052DH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	GRSTS[1:0]		予約				GRSTR	GREN
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R(RM0),W1	R/W

[bit7, bit6] GRSTS[1:0] (clock GeaR STatuS flags) : **クロックギアステータスフラグ**

クロックギアの状態を表示します。

GRSTS[1:0]	ステータス
00	クロックギア低速発振状態で停止または クロックギアを使用しない(CCCGRCR0:GREN=0)または PLL/SSCG リセット状態(CSELR:PCEN=0)
01	GEAR UP 動作中
10	クロックギア高速発振状態で停止
11	GEAR DOWN 動作中

[bit5～bit2] (予約)

[bit1] GRSTR (clock GeaR STaRt) : **クロックギアスタート**

本ビットへの"1"書込みによりクロックギアが動作を開始します。
クロックギアの動作はGRSTSビットの値により動作が異なります。(ギアアップまたはギアダウン)

GRSTS=00の場合

GRSTR	動作
"0"書込み	動作には影響しない
"1"書込み	ギアアップ動作開始

GRSTS=01/11の場合

GRSTR	動作
"0"書込み	動作には影響しない
"1"書込み	動作には影響しない

GRSTS=10の場合

GRSTR	動作
"0"書込み	動作には影響しない
"1"書込み	ギアダウン動作開始

＜注意事項＞

CSELR:CKS[1:0]=10(PLL/SSCGクロック(PLLSSCLK)を選択)かつCCCGRCR0:GREN=1 (クロックギアイネーブル)のときのみ本ビットへの書込みが可能です。
本ビットはクロックギアアップ (ダウン) 動作が終了すると、自動的に"0"にクリアされます。また、CSELR:PCEN=0 (PLL/SSCGクロック発振停止) のとき、本ビットは"0"にクリアされます。
リードモディファイライト命令では、本ビットは常に"0"が読み出されます。本ビットが"1"の期間に書込みを行った場合、2回目以降の書込みは無視されます。

[bit0] GREN (clock GeaR ENable) : **クロックギアイネーブル**

クロックギア動作を許可します。

GREN	動作
0	クロックギアを使用しない
1	クロックギアを使用する

＜注意事項＞

本ビットはPLL/SSCGクロック発振停止 (CSELR:PCEN = 0) のときのみ書込みできます。

5.4.19 クロックギアコンフィグ設定レジスタ 1 : CCCGRCR1 (CCtl Clock Gear Config. Register 1)

クロックギアコンフィグ設定レジスタ1のビット構成について示します。

クロックギアの各種設定を行います。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN = 0)のときのみ書込み可能です。

■ CCCGRCR1: アドレス 052E$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	GRSTP[1:0]		GRSTN[5:0]					
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7, bit6] GRSTP[1:0] (clock GeaR STeP select) : クロックギアステップ選択

クロックギアアップ/ダウン時のステップ数(インクリメント・デクリメント数)を選択します。

GRSTP[1:0]	ステップ数
00	1
01	2
10	3
11	4

[bit5～bit0] GRSTN[5:0] (clock GeaR STart step Number select) : クロックギア開始ステップ選択

クロックギア動作開始時のステップを選択します。0～63ステップの間で選択できます。

GRSTN[5:0]	ステップ数
000000	0
000001	1
000010	2
…	……
111101	61
111110	62
111111	63

＜注意事項＞

GRSTN = 111111(ステップ数63)設定の場合にはギアは動作しません。

5.4.20 クロックギアコンフィグ設定レジスタ 2 : CCCGRCR2 (CCtl Clock Gear Config. Register 2)

クロックギアコンフィグ設定レジスタ2のビット構成について示します。

クロックギアの各種設定を行います。

本レジスタはPLL/SSCGクロック発振停止(CSELR:PCEN = 0)のときのみ書込み可能です。

■ CCCGRCR2: アドレス 052F$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	GRLP[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] GRLP[7:0] (clock GeaR LooP number select) : クロックギア繰返し回数選択

1ステップの繰り返し回数を選択します。設定可能な繰返し回数は1-256です。本ビットで設定した回数が終了すると、ステップがインクリメント/デクリメントします。

GRLP[7:0]	ループ数
0000_0000	1
0000_0001	2
0000_0010	3
…	……
1111_1101	254
1111_1110	255
1111_1111	256

5.4.21 RTC/PMU クロック選択レジスタ ：CCRTSELR (CCtl RTc pmu clock SELect Register)

RTC/PMUクロック選択レジスタのビット構成について示します。

RTC/PMUクロックのクロック・ソースを選択します。

■ CCRTSELR: アドレス 0530H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	CST	予約						CSC
初期値	*	0	0	0	0	0	0	*
属性	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

* 時計モード(電源遮断)からの復帰では初期化しません。それ以外は0に初期化します。

[bit7] CST (Clock source select STatus monitor): クロックソース選択ステータスモニタ

CSCビットは書込み後、クロック切換えが完了するまで、クロック乗換によるタイムラグが発生します。本ビットにより、切換えが完了したかどうか確認ができます。

CST	内容
0	クロック切換え完了
1	クロック切換え中

＜注意事項＞

クロック1系統品の場合、本ビットは常に"0"固定となります。通常、メインクロック×約3サイクル＋サブクロック×約3サイクルで切換えが完了します。

[bit6～bit1] (予約)

[bit0] CSC (Clock SourCe select) : クロックソース選択

RTC/PMUクロックのクロック・ソースを選択します。

CSC	クロックソース
0	メイン発振クロック
1	サブ発振クロック

＜注意事項＞

SCRDY=1かつMCRDY=1の場合のみ、CSCビットは書換え可能です。クロック1系統品の場合、本ビットは書込み値にかかわらず"0"固定となります。

＜注意事項＞

- CSC ビット書換え後、RTC および PMU クロックの切換え動作が完了するまでにメインクロック×約 3 サイクル＋サブクロック×約 3 サイクルかかります。切換え動作中にメインクロックやサブクロックの発振が停止した場合、切換え動作は正しく完了されません。必ず CST ビットが"0"の状態(切換え完了状態)で発振を停止するようにしてください。
- スタンバイ時計モード（電源遮断）からの復帰時には、本ビットは初期化されません。
  また、スタンバイ時計モード（電源遮断）からの復帰中は内部リセットが発行されており、パワーオンリセット/内部低電圧リセット/RSTX・NMIX 同時アサートによるリセット以外のリセット要因を受付けません。
  このとき、本ビットは初期化されませんので、立上げ後に RSTX 端子入力からのリセット入力または、外部低電圧検出フラグがセットされている場合には、必要があれば本ビットを初期化してください。

5.4.22 PMU クロック分周設定レジスタ 0 : CCPMUCR0 (CCtl PMU Clock divide Register 0)

PMUクロック分周設定レジスタ0のビット構成について示します。

PMUクロックの分周設定を行います。

■ CCPMUCR0: アドレス 0532H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FST	予約					FDIV[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit7] FST (F-divider STatus monitor): F分周器ステータスモニタ

FDIV[1:0]ビットは書込み後、書込み値が反映されるまで、クロック乗換によるタイムラグが発生します。本ビットにより、設定値が反映されたかどうか確認ができます。

通常、RTCクロック×約4サイクル＋PCLK1×約4サイクルで設定値が反映されます。

FST	内容
0	書込み値を反映済み
1	書込み値を反映中

[bit6～bit2] (予約)

[bit1, bit0] FDIV[1:0] (F-DIVide ratio setting): F分周比設定

F分周器の分周比を設定します。PMUには32kHz以下のクロックを供給しなければなりません。CCRTSELR:CSC=0 (メイン発振クロックを選択)の場合、F分周器を用いて、32kHz以下になるように本ビットを設定します。

FDIV[1:0]	分周比	対象メイン発振周波数
00	128 分周(初期値)	4MHz 時
01	256 分周	8MHz 時
10	384 分周	12MHz 時
11	512 分周	16MHz 時

＜注意事項＞

CCPMUCR0:FSTビットが"1"の期間は、本ビットへの書込みは無視されます。
CCRTSELR:CSC=1のとき (サブ発振クロック選択時)、本ビットの値にかかわらず、F分周比は非分周となります。

5.4.23 PMU クロック分周設定レジスタ 1 : CCPMUCR1 (CCtl PMU Clock divide Register 1)

PMUクロック分周設定レジスタ1のビット構成について示します。

PMUクロックの分周設定を行います。

■ CCPMUCR1: アドレス 0533H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	GST	予約		GDIV[4:0]				
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W

[bit7] GST (G-divider STatus monitor): G分周器ステータスモニタ

GDIV[4:0]ビットは書込み後、書込み値が反映されるまで、クロック乗換によるタイムラグが発生します。本ビットにより、設定値が反映されたかどうか確認ができます。

通常、RTCクロック×約4サイクル＋PCLK1×約4サイクルで設定値が反映されます。

GST	内容
0	書込み値を反映済み
1	書込み値を反映中

＜注意事項＞

本ビットが"1"の期間は、CCPMUCR1:GDIV[4:0]への書込みは無視されます。

[bit6, bit5] (予約)

[bit4～bit0] GDIV[4:0] (G-DIVide ratio setting) : G分周比設定

G分周器の分周比を設定します。PMUクロックの周期は、PMUに供給される (APB) バスクロックの周期の4倍以上でなければなりません。本分周器を用いて、上記の関係を満たすように、PMUクロックの分周比を設定します。

GDIV[4:0]	分周比
00000	分周しない(初期値)
00001	2
00010	3
…	……
11101	30
11110	31
11111	32

＜注意事項＞

CCPMUCR1:GSTビットが"1"の期間は、本ビットへの書込みは無視されます。

5.4.24 同期/非同期コントロールレジスタ ：SACR (Sync/Async Control Register)

同期/非同期コントロールレジスタのビット構成について示します。

周辺クロック(PCLK2)の選択を行います。

■ SACR: アドレス 1000H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							M
初期値	1	1	1	1	1	1	1	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W

[bit7～bit1] (予約)

[bit0] M : 周辺クロック (PCLK2) の同期/非同期設定レジスタ

CPUがSSCGクロックを選択しているときに周辺クロック (PCLK2) の切換えを行います。

M	同期/非同期設定
0	同期 (CPU/周辺共に PLL/SSCG クロック)
1	非同期 (CPU は PLL/SSCG クロック, 周辺は PLL クロック)

5.4.25 周辺インタフェースクロック分周 ：PICD (Peripheral Interface Clock Divider)

周辺インタフェースクロック分周のビットの構成について示します。

PLLクロック(PLLCLK)から作成する周辺クロックの分周設定を行います。

■ PICD: アドレス 1001H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				PDIV[3:0]			
初期値	1	1	1	1	0	0	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W

[bit7～bit4] (予約)

[bit3～bit0] PDIV[3:0] : 周辺クロック分周比設定

SACR.M=1時のPLLクロック (PLLCLK) [非拡散クロック] から周辺クロック (PCLK2) の分周比を設定します。

PDIV[3:0]	PLL クロック (PLLCLK) [非拡散クロック] → PCLK2 分周比
0000	分周しない
0001	2 分周
0010	3 分周
0011	4 分周 (初期値)
0100	5 分周
0101	6 分周
0110	7 分周
0111	8 分周
1000	9 分周
1001	10 分周
1010	11 分周
1011	12 分周
1100	13 分周
1101	14 分周
1110	15 分周
1111	16 分周

＜注意事項＞

周辺クロック (PCLK2) は必ず40MHz以下になるように本レジスタを設定してください。

5.5 動作説明

動作について説明します。

5.5.1. 発振制御

5.5.2. 発振安定待ち

5.5.3. ソースクロック (SRCCLK) 選択

5.5.4. タイマ

5.5.5. クロック競合時の注意

5.5.6. クロックギア回路

5.5.7. MDI通信中の動作

5.5.8. PMUクロック(PMUCLK) について

5.5.1 発振制御

発振制御について示します。

5.5.1.1. メインクロック (MCLK)

5.5.1.2. サブクロック (SBCLK)

5.5.1.3. PLL/SSCGクロック (PLLSSCLK)

5.5.1.4. PLL/SSCGクロック使用時の制限事項

5.5.1.1 メインクロック (MCLK)

メインクロック (MCLK) について示します。

メインクロックが発振を停止するのは、以下のいずれかの条件となります。

- SINIT リセット(『リセット』の章を参照してください。)
- ストップモード期間中
- ソースクロック (SRCCLK) としてサブクロック (SBCLK) を選択し、CSELR:MCEN に"0"を設定している期間中

上記発振停止条件のすべてが取り下げられた後、CSTBR:MOSW[3:0] に設定された発振安定待ち時間を経過してから、クロック供給を開始します。リセット入力からの復帰では、CSTBR:MOSW[3:0] は初期化されるため、必ず初期値で規定される発振安定待ち時間となります。

＜注意事項＞

クロック1系統品の場合、メインクロック発振許可は常に有効 (MCEN=1) となります。

5.5.1.2 サブクロック (SBCLK)

サブクロック(SBCLK)について示します。

サブクロックが発振を停止するのは、以下のいずれかの条件となります。

- リセット発生後 (停止前のバスアイドル待ち時間はとります。『リセット』の章を参照してください。)
- ストップモード期間中
- ソースクロック (SRCCLK) としてサブクロック (SBCLK) 以外を選択し、CSELR:SCEN ビットに"0"を設定している期間中
- サブ発振とポートが兼用 (メタルオプション) になっているため、ポートとして使用する場合

上記発振停止条件のすべてが取り下げられた後、CSTBR:SOSW[2:0]に設定された発振安定待ち時間を経過してから、クロック供給を開始します。リセット入力またはINIT状態からの復帰では、CSELR:SCENビットは"0"に初期化されるため、"1"を設定するまではサブクロックの発振は停止しています。

＜注意事項＞

- クロック 1 系統品の場合、サブクロック発振許可は常に無効 (SCEN=0) となります。
- クロック 1 系統品の場合、サブタイマは使用できません。

5.5.1.3 PLL/SSCG クロック (PLLSSCLK)

PLL/SSCGクロック (PLLSSCLK) について示します。

本LSIは、PLLおよびSSCG (拡散クロックを発生するPLL) を持ち、ノイズ低減の目的でSSCGを選択することができます。CPUおよび周辺機能が選択できるクロックの組合せは以下のとおりです。

表 5-2 クロックモード

	クロックモード		
	RUN1	RUN2	RUN3
CPU	PLL	SSCG	SSCG
CAN	PLL	PLL	PLL
周辺	PLL	SSCG	PLL
OCDU	PLL	PLL	PLL

CPU/周辺 (タイマ/通信系) クロックの選択はCCPSSELR:PCSELビットで行います。また、CPUがSSCGクロックで動作している時､周辺 (タイマ/通信系) をPLLクロックで動作させることができます｡このときの周辺のクロック選択はSACR:Mビットで行い、分周をPICD:PDIV[3:0]で行います。

＜注意事項＞

CPUをSSCGクロック、周辺をPLLクロックで動作させた場合には、CPU/周辺間で非同期乗換えが入るため、アクセスサイクルに5～8PCLK2のペナルティが加算されます。このとき、PCLK1とPCLK2の周波数は同一になるように設定してください。また、CPU/周辺ともにPLLクロックで動作させたい場合には、必ずSACR:Mで同期を選択してください。

PLL/SSCGクロック(PLLSSCLK)が発振を停止するのは、以下のいずれかの条件となります。

1. リセット発生後(停止前のバスアイドル待ち時間は取ります。『リセット』の章を参照してください。)
2. メインクロックの発振が停止している期間中 (PCEN=0 となる)
3. メインクロックの発振安定待ち時間中 (PCEN=0 となる)
4. 時計モード期間中
5. ソースクロック(SRCCLK)として PLL/SSCG クロック(PLLSSCLK)以外を選択し、CSELR:PCEN ビットに"0"を設定している期間中

上記発振停止条件のすべてが取り下げられた後、PLLCR:POSW[3:0]に設定されたPLL/SSCGクロック ロック待ち時間を経過してから、クロック供給を開始します。リセット入力またはINIT状態からの復帰では、CSELR:PCENビットは"0"に初期化されるため、"1"を設定するまではPLL/SSCGクロックの発振は停止しています。

PLL/SSCGに関係するクロックの周波数と逓倍率の計算方法は、以下のとおりです。
(PLL/SSCG設定)

6. PLL/SSCG 入力クロック周波数 =(メイン発振周波数) / (PLLCR:PDS[3:0] 分周比)
7. PLL 逓倍率 =(CCPLLFBR:IDIV[6:0] FB 入力分周比)
   SSCG 逓倍率 =(CCSSFBR0:NDIV[5:0] FB 入力分周比)×(CCSSFBR1:PDIV[4:0] FB 入力分周比)
8. PLL マクロ発振クロック周波数 =(PLL/SSCG 入力クロック周波数)×PLL 逓倍率
   SSCG マクロ発振クロック周波数 =(PLL/SSCG 入力周波数)×SSCG 逓倍率
9. PLL クロック周波数 =(PLL マクロ発振クロック周波数)÷(CCPSDIVR:PODS[2:0]分周比)
   SSCG クロック周波数 =(SSCG マクロ発振クロック周波数)÷(CCPSDIVR:SODS[2:0]分周比)

図 5-10 PLL 周辺ブロック図

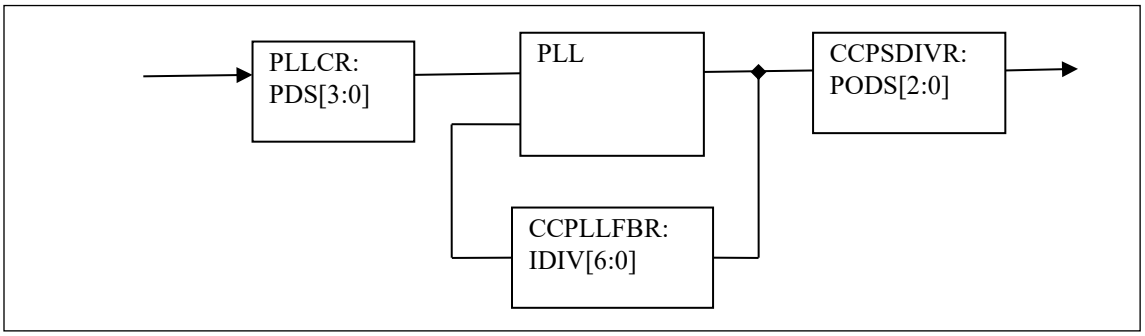
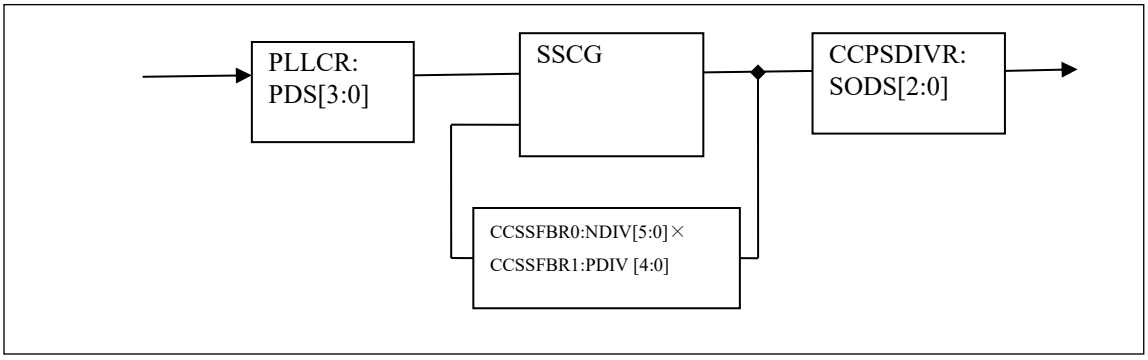


図 5-11 SSCG 周辺ブロック図



PLL/SSCG入力クロック, PLL/SSCG逓倍率, PLL/SSCGマクロ発振クロックは、本シリーズ搭載PLL/SSCGの以下の使用条件内になるよう設定してください。PLL/SSCG使用条件範囲は以下の注意事項を参照してください。

＜注意事項＞

- デバッグ動作時は、MDI 通信に常時 PLL クロック供給が必要なため、PLL は停止できません。
- PLL-SSCG 切換え時は、割込みを正常に受け渡しできません。
  そのため、PLL-SSCG/同期非同期の切換えを行う場合は、リソースからの割込みを禁止してください。
- PLL/SSCGマクロ発振クロック周波数は、上限および下限があります。下記の範囲を超えないようにPLL/SSCGの逓倍率を設定してください。
  PLL/SSCG：
  - ☐ 200MHz≦PLL マクロ発振クロック周波数≦320MHz
  - ☐ 200MHz≦SSCG マクロ発振クロック周波数≦320MHz(Down Spread)

5.5.1.4 PLL/SSCG クロック使用時の制限事項

PLL/SSCGクロックの制限事項について示します。

PLLクロックおよびSSCGクロックを使用する場合には、以下の制限事項にしたがって使用してください。

■ クロック制御 PLL クロック周波数

周波数(max)	FCTLR:FAW	CCPSSELR: PCSEL	備考
80MHz	00	0	

＜注意事項＞

周波数(max)を超えないようにPLLCR、またはCCPSDIVRとCCPLLFBRを設定してください。

■ クロック制御 SSCG クロック周波数

周波数 (max)	FCTLR: FAW	CCPSSELR: PCSEL	CCSSCCR0: SSEN	CCSSCCR0: SMODE	CCSSCCR1: RATESEL	備考
80MHz	01	1	1	0 / 1	000 ～ 110	
72MHz	00	1	1	0	000 ～ 110	DownSpread
72MHz	00	1	1	1	000	CenterSpread(0.5%)
72MHz	00	1	1	1	010	CenterSpread(1%)
72MHz	00	1	1	1	011	CenterSpread(2%)
71MHz	00	1	1	1	100	CenterSpread(3%)
71MHz	00	1	1	1	101	CenterSpread(4%)
70MHz	00	1	1	1	110	CenterSpread(5%)
80MHz	01	1	0	0/1	000 ～ 110	拡散率 0%
72MHz	00	1	0	0/1	000 ～ 110	拡散率 0%

＜注意事項＞

周波数(max)を超えないようにCCPSDIVR, CCSSFBR0とCCSSFBR1を設定してください。
周辺クロックは40MHzを超えないようにDIVR2.PICDを設定してください。
周辺クロックは40MHzを超えないようにDIVR2を設定してください。

■ SSCG 使用時のモジュレーション・レートと分周比の関係

CCSSCCR1.RATESEL[2:0]		CCSSFBR0.NDIV[5:0]		
モジュレーションレート	設定値	分周比範囲	設定値下限	設定値上限
0.50%	00x	8 - 60	7h	3Bh
1.00%	010	8 - 60	7h	3Bh
2.00%	011	8 - 48	7h	2Fh
3.00%	100	8 - 31	7h	1Eh
4.00%	101	8 - 23	7h	16h
5.00%	110	8 - 18	7h	11h

5.5.2 発振安定待ち

発振安定待ちについて示します。

各クロック入力の発振安定待ちについて説明します。

5.5.2.1 安定待ち時間生成条件

安定待ち時間生成条件について示します。

各クロックの発振停止制御が解除されると、発振安定待ち状態となります。クロックごとに設定された発振安定待ち時間を経由した後、発振安定待ち状態は解除され、クロック供給を再開します。

メインクロック(MCLK)は、リセットにより設定レジスタが初期化されますので、リセット解除前に発振停止していた場合は発振安定待ち状態に入ります。INITレベルおよびRSTレベルではリセットによるメインクロック発振停止はしませんので、INITレベルおよびRSTレベルのリセットでメインクロックが発振していた場合は、発振安定待ちはとりません。

5.5.2.2 安定待ち時間選択

安定待ち時間選択について示します。

各クロックの発振安定待ち時間は、CSTBR, PLLCRの設定により変更することができます。

クロック発振安定待ち時間の設定リセット後の初期値

- メインクロック　CSTBR:MOSW[3:0]ビット　2^{15}×メインクロック周期
- PLL/SSCG クロック　PLLCR:POSW[3:0]ビット　2^{16}×メインクロック周期
- サブクロック　CSTBR:SOSW[2:0]ビット　2^{8}×サブクロック周期

リセット(INITまたはRST)により、CSTBR:MOSW[3:0]は初期化されるため、かならず初期値で規定されるメインクロック発振安定待ち時間となります。それ以外の場合、CSTBR:MOSW[3:0]に設定することによりメインクロック発振安定待ち時間を変更することが可能です。

リセット(INITまたはRST)により、PLLCR:POSW[3:0]は初期化されるため、かならず初期値で規定されるPLL/SSCG クロック　ロック待ち時間となります。それ以外の場合、PLLCR:POSW[3:0]に設定することによりPLL/SSCGクロック　ロック待ち時間を変更することが可能です。PLLCR:POSW[3:0]を設定後、CSELR:PCENを"1"に設定してください。詳細は「5.4.8 PLL設定レジスタ : PLLCR (PLL Configuration Register)」の、POSWの説明を参照してください。

リセット(INITまたはRST)により、CSTBR:SOSW[2:0]は初期化されるため、かならず初期値で規定されるサブクロック発振安定待ち時間となります。それ以外の場合、CSTBR:SOSW[2:0]に設定することによりサブクロック発振安定待ち時間を変更することが可能です。

5.5.2.3 安定待ち時間の終了

安定待ち時間の終了について示します。

ソースクロック(SRCCLK)として選択しているクロックが発振安定待ち状態にある場合、動作を停止しています。発振安定待ち時間の終了により、動作を再開します。ソースクロックとして選択していないクロックが発振安定待ち時間中であることは、各クロックがイネーブルのときCMONRレジスタの各クロックに対応したレディビットの値で確認することができます。

クロック発振安定待ち状態の表示発振安定状態の表示

- メインクロック　CMONR:MCRDY="0", CMONR:MCRDY="1"
- PLL/SSCG クロック(PLLSSCLK)　CMONR:PCRDY="0", CMONR:PCRDY="1"
- サブクロック(SBCLK)　CMONR:SCRDY="0", CMONR:SCRDY="1"

5.5.3 ソースクロック (SRCCLK) 選択

ソースクロック(SRCCLK)選択について示します。

動作クロックとなるソースクロック(SRCCLK)の選択制御について説明します。

5.5.3.1 初期化時のソースクロック選択

初期化時のソースクロック選択について示します。

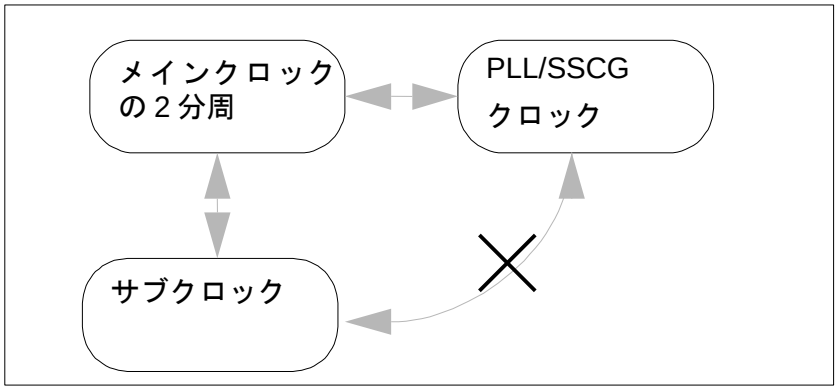
リセット(RST)の発生により、ソースクロック(SRCCLK)はメインクロック(MCLK)の2分周を選択します。以降、プログラム動作開始後は、CSELR:CKS[1:0]の設定によってソースクロックを変更することが可能となります。

5.5.3.2 ソースクロック切換え手順

ソースクロック切換え手順について示します。

ソースクロック(SRCCLK)をPLL/SSCGクロック(PLLSSCLK)からサブクロック(SBCLK)、サブクロックからPLL/SSCGクロックへ直接切換えはできません。一度メインクロックの2分周に切換えてください。ソースクロックを切換えても発振許可ビット(CSELR:xCEN)の値は保持されていますので、必要に応じて発振停止設定を行ってください。

図 5-12 ソースクロックの切換え手順



1. メインクロックの 2 分周→PLL/SSCG クロック

   ソースクロックにメインクロックの2分周を選択中(CMONR:CKM[1:0]=00)

   ↓

   PLL/SSCG逓倍率, SSCG変調, PLL/SSCG選択, PLL/SSCGロック待ち時間を設定(PLLCR/ CCPSSELR/ CCPSDIVR/ CCPLLFBR/ CCSSFBR0/ CCSSFBR1/ CCSSCCR0/ CCSSCCR1を設定)　-- PLL/SSCG発振有効にしていない場合--

   ↓

   クロック・ギアを設定(CCCGRCR0:GREN/CCCGRCR1/CCCGRCR2)

   ↓

   PLL/SSCGクロック発振安定待ちタイマ割込み要因クリア(PTIF=0)

   ↓

(必要に応じ)PLL/SSCGクロック発振安定待ちタイマ割込みイネーブル設定(PTIE=1)

↓

PLL/SSCG発振開始(PCEN=0→1)

↓

PLL/SSCGロック待ちループ(PCRDY=1になるまでループ)、または割込み待ち。

↓

PLL/SSCGクロック発振安定待ちタイマ割込みクリア(PTIF=0, PTIE=0)

↓

ソースクロックをPLL/SSCGクロックに切換え(CSELR:CKS[1:0]=00→10)

↓

クロックギアを開始(CCCGRCR0:GRSTR=1)

↓

クロックギアが高速発振停止状態であることを確認(CCCGRCR0:GRSTS[1:0]=10)

↓

ソースクロックにPLL/SSCGクロックを選択中(CMONR:CKM[1:0]=10)

2. PLL/SSCG クロック→メインクロックの 2 分周

ソースクロックにPLL/SSCGクロックを選択中(CMONR:CKM[1:0]=10)

↓

クロックギアを開始(CCCGRCR0:GRSTR=1)

↓

クロックギアが低速発振停止状態であることを確認(CCCGRCR0:GRSTS[1:0]=00)

↓

ソースクロックをメインクロックの2分周に切換え(CSELR:CKS[1:0]=10→00)

↓

ソースクロックにメインクロックを選択中(CMONR:CKM[1:0]=00)

3. メインクロックの 2 分周→サブクロック

ソースクロックにメインクロックの2分周を選択中(CMONR:CKM[1:0]=01)

↓

サブクロック発振安定待ち時間を設定(CSTBR:SOSW[2:0]を設定)
--サブ発振有効にしていない場合--

↓

サブタイマ割込み要因クリア(STIF=0)

↓

(必要に応じ)サブタイマ割込みイネーブル設定(STIE=1)

↓

サブ発振開始(SCEN=0→1)

↓

サブクロック発振安定待ちループ(SCRDY=1になるまでループ)、または割込み待ち。

↓

サブタイマ割込みクリア(STIF=0)

↓

ソースクロックをサブクロックに切換え(CSELR:CKS[1:0]=01→11)

↓

ソースクロックにサブクロックを選択中(CMONR:CKM[1:0]=11)

4. サブクロック→メインクロックの 2 分周

ソースクロックにサブクロックを選択中(CMONR:CKM[1:0]=11)

↓

メインクロック発振安定待ち時間を設定(CSTBR:MOSW[3:0]を設定)
-- メイン発振有効にしていない場合 --

↓

メインタイマ割込み要因クリア(MTIF=0)

↓

(必要に応じ)メインタイマ割込みイネーブル設定(MTIE=1)

↓

メイン発振開始(MCEN=0→1)

↓

メインクロック発振安定待ちループ(MCRDY=1になるまでループ)、または割込み待ち。

↓

メインタイマ割込みクリア(MTIF=0)

↓

ソースクロックをメインクロックの2分周に切換え(CSELR:CKS[1:0]=11→01)

↓

ソースクロックにメインクロックを選択中(CMONR:CKM[1:0]=01)

図 5-13 PLL/SSCG モード設定例 メイン→PLL/SSCG

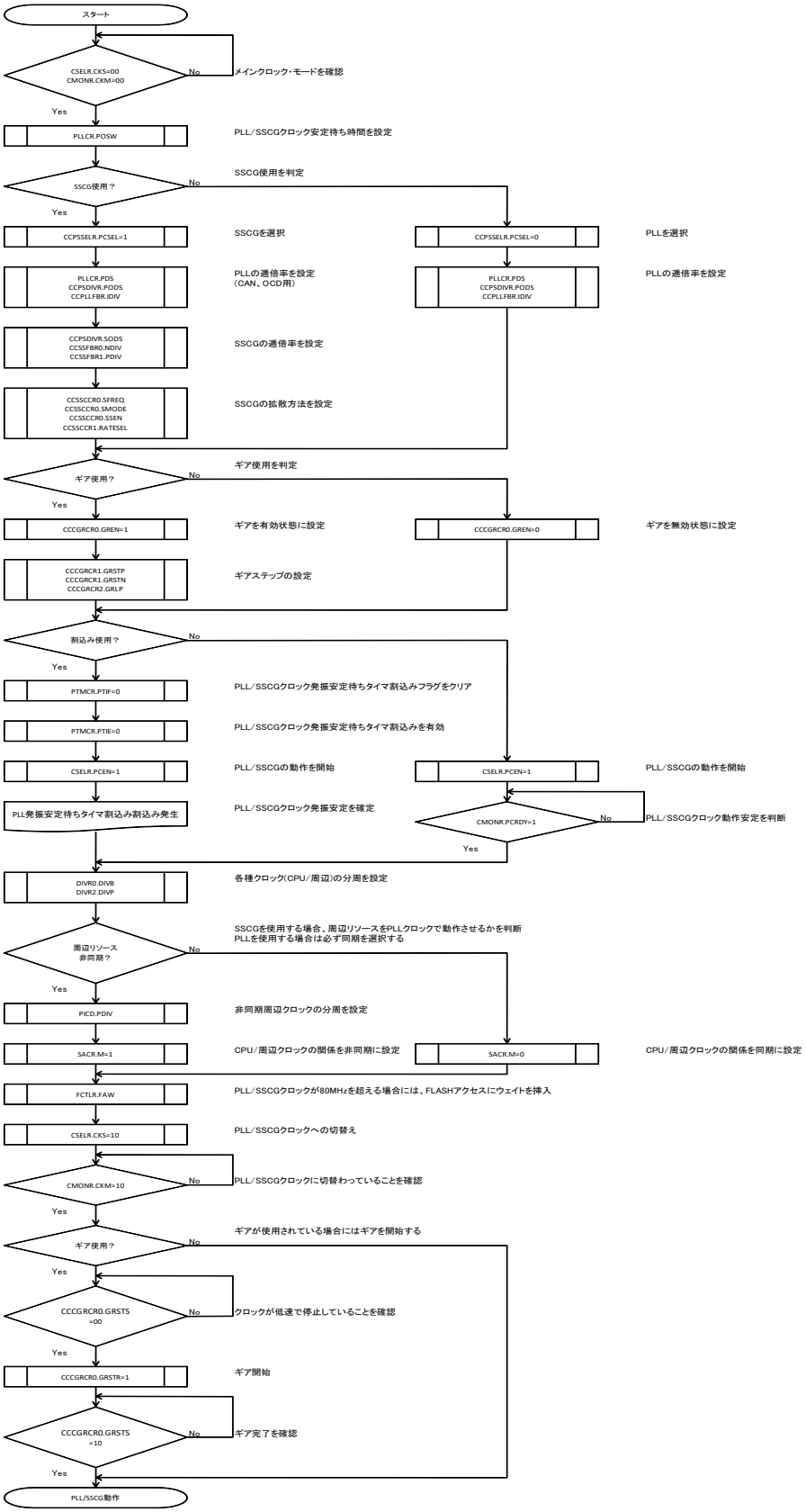
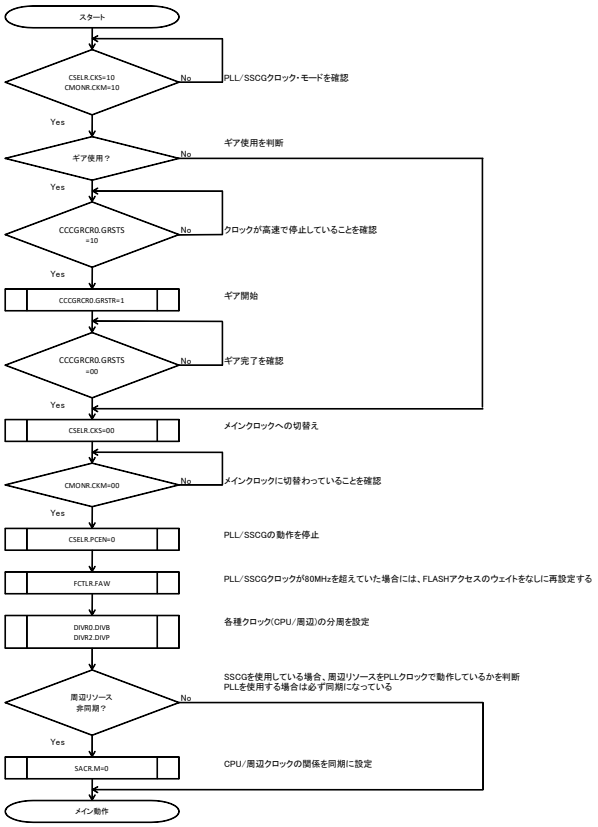


図 5-14 PLL/SSCG モード設定例 PLL/SSCG→メイン



5.5.4 タイマ

タイマについて示します。

5.5.4.1. メインクロック発振安定待ちタイマ (メインタイマ)

5.5.4.2. サブクロック発振安定待ちタイマ (サブタイマ)

5.5.4.3. PLL/SSCGクロック発振安定待ちタイマ (PLLタイマ)

5.5.4.4. 設定

5.5.4.5. タイマ割込み設定手順

5.5.4.6. タイマ動作

5.5.4.7. 時計モードとタイマ割込み

5.5.4.1 メインクロック発振安定待ちタイマ (メインタイマ)

メインクロック発振安定待ちタイマ (メインタイマ) について示します。

メインタイマは、メインクロック (MCLK) で動作するタイマです。メインクロック発振安定待ち時間生成に使用され、それら以外のメインクロック安定状態では一定期間で割込みを発生するタイマとして使用することができます。

5.5.4.2 サブクロック発振安定待ちタイマ (サブタイマ)

サブクロック発振安定待ちタイマ(サブタイマ)について示します。

サブタイマは、サブクロック (SBCLK) で動作するタイマです。サブクロック発振安定待ち時間生成に使用され､それ以外のサブクロック安定状態では一定期間で割込みを発生するタイマとして使用することができます。

5.5.4.3 PLL/SSCG クロック発振安定待ちタイマ (PLL タイマ)

PLL/SSCGクロック発振安定待ちタイマ (PLLタイマ) について示します。

PLLタイマは、メインクロックで動作するタイマで、PLL/SSCGクロック発振安定待ち時間生成専用です。このタイマは汎用タイマとしては使用できません。

5.5.4.4 設定

設定について示します。

メインタイマ動作許可(MTMCR:MTE=1)にするとメインタイマのカウントが開始します。メインタイマ動作禁止(MTMCR:MTE=0)にするとメインタイマのカウントが停止しメインタイマはクリアされます。メインタイマクリア(MTMCR:MTC=1)にするとメインタイマはクリアされます。

クリアされるまでの間、MTMCR:MTC=1が読み出されます。MTMCR:MTS[3:0]で割込みの周期を設定できます。MTMCR:MTIE=1のとき、MTMCR:MTIF=1となるとメインタイマ割込みが発生します。MTMCR:MTIFは"0"書込みによりクリアされます。

サブタイマ動作許可(STMCR:STE=1)にするとサブタイマのカウントが開始します。サブタイマ動作禁止(STMCR:STE=0)にするとサブタイマのカウントが停止し、サブタイマはクリアされます。

サブタイマクリア(STMCR:STC=1)にするとサブタイマはクリアされます。クリアされるまでの間、STMCR:STC=1が読み出されます。STMCR:STS[2:0]で割込みの周期を設定できます。STMCR:STIE=1のとき、STMCR:STIF=1となるとサブタイマ割込みが発生します。STMCR:STIFは"0"書込みによりクリアされます。

＜注意事項＞

タイマ割込みの周期設定(MTSおよびSTS)は、PCLK1 × 5クロック以上の周期を設定してください。タイマ割込みの周期を極端に短く設定した場合、割込み要因がセットされない可能性があります。

5.5.4.5 タイマ割込み設定手順

タイマ割込み設定手順について示します。

割込み設定手順について説明します。割込みの設定例を以下に示します。

タイマ割込み禁止設定(MTMCR:MTIE=0)/(STMCR:STIE=0)、割込みフラグクリア設定(MTMCR:MTIF=0)/(STMCR:STIF=0)

↓

タイマ動作禁止設定(MTMCR:MTE=0)/(STMCR:STE=0)

↓

MTC=0/STC=0を確認

↓

タイマ周期設定(MTMCR:MTS=1000～1111)/(STMCR:STS=000～111)

↓

タイマ割込み許可設定(MTMCR:MTIE=1)/(STMCR:STIE=1)

↓

タイマ動作許可設定(MTMCR:MTE=1)/(STMCR:STE=1)

↓

設定時間後割込み発生

↓

割込みルーチンへ

↓

割込みフラグクリア設定(MTMCR:MTIF=0)/(STMCR:STIF=0)*

↓

割込みフラグ確認(MTMCR:MTIF=0)/(MTMCR:STIF=0)

↓

プログラム動作

↓

RETI

*割込みフラグクリア設定は"0"書込みしてもすぐに反映されないので"0"になるまで読込みを繰り返してください。

5.5.4.6 タイマ動作

タイマ動作について示します。

メインタイマはMTMCR:MTE=1の間、メインクロック(MCLK)でカウントアップします。MTMCR:MTS[3:0]で選択された周期でオーバフローしたら、MTMCR:MTIF=1となります。

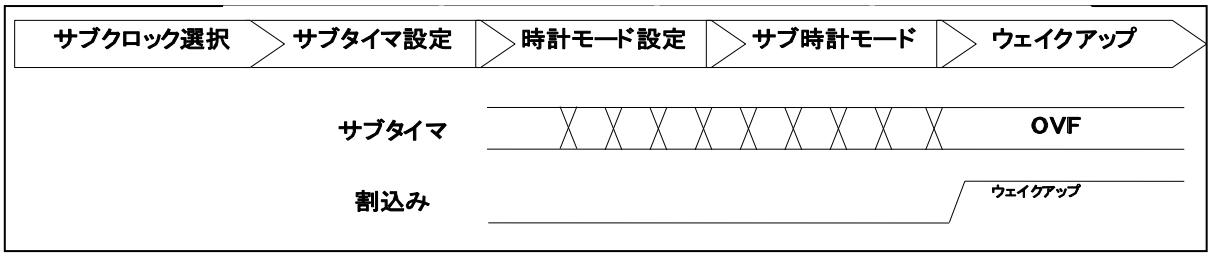
サブタイマはSTMCR:STE=1の間、サブクロック(SBCLK)でカウントアップします。STMCR:STS[2:0]で選択された周期でオーバフローしたら、STMCR:STIF=1となります。

5.5.4.7 時計モードとタイマ割込み

時計モードとタイマ割込みについて示します。

時計モードは、特定の機能とタイマ以外のすべての動作を停止させます。(『消費電力制御』の章を参照してください。) メイン/サブタイマ割込みまたはRTC割込みを利用して時計モードからウェイクアップすることができます。下記にサブタイマよりウェイクアップする設定での時計モードの遷移例を示します。

図 5-15 時計モードからのウェイクアップ



5.5.5 クロック競合時の注意

クロック競合時の注意について示します。

割込みハンドラ内で、CPUクロック(CCLK)より非常に低い周波数で駆動しているペリフェラルの割込みクリアをして直ちに割込みハンドラを終了すると、割込みハンドラ期間内にそのペリフェラルは内部処理を完了しきれず、二重に割込みハンドラが呼び出されることがあるので注意してください。

5.5.6 クロックギア回路

クロックギア回路について示します。

メインクロックからPLL/SSCGクロックへの切換え時またはPLL/SSCGクロックからメインクロックへの切換え時に周波数が急激に変動するため、電源電流も大きく変動します。クロック切換え部にクロックギア回路を持っており、これを使用することにより、動作周波数を低周波から高周波または高周波から低周波に徐々に変動させることができ、電源電流の変動を低減できます。

5.5.6.1 ギアアップの手順

ギアアップの手順について示します。

1. 発振安定待ちタイマ完了後、クロックギア開始ステップ選択に設定した開始ステップのクロックを出力します。
2. クロックギアスタート (CCCGRCR0.GRSTR) を"1"にセットし、立上りを検出すると、クロックギアステータスフラグ(CCCGRCR0.GRSTS[1:0])が"00"->"01"に遷移します。(ギアアップスタート)
3. クロックギア・ステップ選択, 繰返し回数選択にしたがって、ギアアップを行います。
   ステップ数が小さく、繰返し回数が大きいほど緩やかに変化します。
4. クロックが最大ステップに到達すると、クロックギアステータスフラグ (CCCGRCR0.GRSTS[1:0]) が"01"->"10"に遷移します。(ギアアップ終了, ギア停止)
   これ以降は最大ステップ (64 ステップ) でクロックを出力します。
5. ギア停止後、クロックギアスタート (CCCGRCR0.GRSTR) はハードウェアで"0"にクリアされます。

5.5.6.2 ギアダウンの手順

ギアダウンの手順について示します。

1. クロックギアスタート(CCCGRCR0.GRSTR)を"1"にセットし、立上りを検出すると、クロックギアステータスフラグ(CCCGRCR0.GRSTS[1:0])が"10"->"11"に遷移します。(ギアダウンスタート)
2. クロックギア・ステップ選択, 繰返し回数選択にしたがって、ギアダウンを行います。
   ステップ数が小さく、繰返し回数が大きいほど緩やかに変化します。
3. クロックが最小ステップに到達すると、クロックギアステータスフラグ(CCCGRCR0.GRSTS[1:0])が"11"->"00"に遷移します。(ギアダウン終了, ギア停止)
   これ以降はクロックギア開始ステップ選択に設定した開始ステップでのクロックを出力します。
4. ギア停止後、クロックギアスタート(CCCGRCR0.GRSTR)はハードウェアで"0"にクリアされます。

5.5.7 MDI 通信中の動作

MDI通信中の動作について示します。

MDI通信中は、STOPモードに移行した場合でもメイン発振が停止しないように制御しています。

また、MDI高速通信中は、CSELR:PCENがクリアされてもPLLリファレンスクロックが供給されるように制御します。PLL関連レジスタの値が保持されて更新されません。しかし、ソフトウェアがCSELR:PCEN=0を設定した場合、PLL関連レジスタの値は自由に更新(書込み)可能です。

もし、PLL関連レジスタに前回設定した値と、違う値を書き込んでPLL/SSCGクロック発振許可を有効(CSELR:PCEN=1)とした場合、PLLクロックの周波数は更新されません。(PLLはロックした状態を保持しているため。)

通常は、PLL関連レジスタには常に同じ値を書き込んで下さい。
PLL関連レジスタは以下のとおりです。

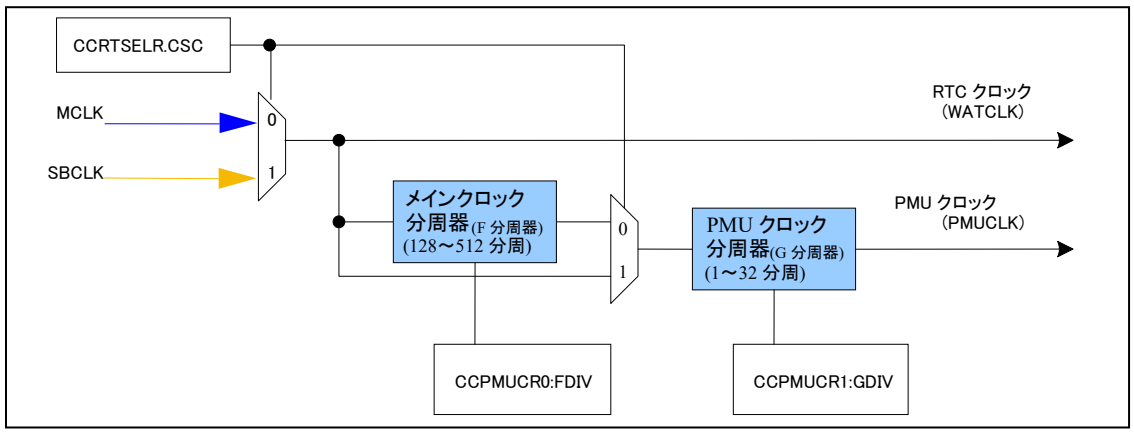
- CCPSDIVR:PODS
- CCPLLFBR:IDIV
- PLLCR:PDS

5.5.8 PMU クロック(PMUCLK) について

PMUクロック(PMUCLK) について示します。

PMUクロックは、パワーマネジメントユニット(PMU) の動作クロックです。スタンバイモードの制御を行う前に、本クロックの設定を完了させてください。

図 5-16 時計・パワーマネジメントクロック生成部



PMUクロックの周波数は以下の式で算出されます。

■ CCRTSELR:CSC＝0 (メインクロックを選択) の場合

PMUクロック周波数＝
メインクロック周波数÷CCPMUCR0:FDIV[1:0]分周比÷CCPMUCR1:GDIV[4:0]分周比

■ CCRTSELR:CSC＝1 (サブクロックを選択) の場合

PMUクロック周波数＝
サブクロック周波数÷CCPMUCR1:GDIV[4:0]分周比

またPMUクロックは以下の仕様制限を遵守してください。 (本制限が守られない場合、正常にシャットダウン処理が行われない可能性があります。)

(1) CCRTSELR:CSCは発振中のクロックを選択してください*。
(2) PMUクロックは32kHz以下になるようにF分周器を使用してください。
(3) 周辺クロック (PCLK1) 周波数の1/4になるようにG分周器を使用してください。
*クロック1系統品の場合、CCRTSELR:CSCは常に"0"が読み出されます。

それぞれの仕様制限について以下のとおり説明します。

(1) CCRTSELR:CSCは発振中のクロックを選択してください。
メインクロックおよびサブクロックの発振状態は、CMONR:MCRDYビットおよびCMONR: SCRDYビットを確認してください。またCCRTSELR:CSCビットを書換える場合、メインクロックとサブクロックのハンドシェイク処理 (クロック乗換) が発生します。この期間、双方のクロックが発振状態 (CMONR:MCRDY＝CMONR:SCRDY＝1) でなければ、切換え動作は正常に完了しません。クロック乗換の状態はCCRTSELR:CSTビットで確認してください。

(2) PMUクロックは32kHz以下になるようにF分周器を使用してください。
PMUクロックはパワースイッチの制御に使用され、電源入力時の昇圧時間の安定化などの理由から、32kHz以下の周波数が推奨されます。

CCRTSELR:CSC＝0の場合、PMUクロックはソースクロックとしてメインクロックが選択されます。PMUクロックの周波数が32kHz以下になるように、CCPMUCR0:FDIVレジスタを設定してください。CCRTSELR:CSC＝1の場合、F分周器は動作に影響しません。

FDIV[1:0]	分周比	対象メイン発振周波数
00	128 分周(初期値)	4MHz 時
01	設定禁止	8MHz 時
10	設定禁止	12MHz 時
11	設定禁止	16MHz 時

(3) 周辺クロック (PCLK1) 周波数の1/4になるようにG分周器を使用してください。
周辺クロック (PCLK1) とPMUクロック (PMUCLK) のクロック乗換には、PMUクロック × 4サイクルが必要です。

周辺クロック (PCLK1) のクロックソースがサブクロックの場合 (CMONR:CKM=10)、PMUクロックの周波数が周辺クロック (PCLK1) 周波数の1/4になるように、CCPMUCR1:GDIVレジスタを設定してください。

また、周辺クロック (PCLK1) のクロックソースがメインクロックの2分周の場合 (CMONR:CKM=00またはCMONR:CKM=01) においても、DIVR0:DIVBやDIVR2:DIVPの設定で周辺クロック (PCLK1) が128kHz (32kHz × 4) 以下になる場合には、同様にCCPMUCR1:GDIVレジスタを設定してください。

GDIV[4:0]	分周比
00000	分周しない(初期値)
00001	2 分周
…	…
11110	31 分周
11111	32 分周

【参考】
周辺クロック (PCLK1) の周波数は以下の式で算出できます。

クロック (PCLK1) 周波数＝
CMONR:CKMで選択中のクロック周波数 ÷ DIVR0:DIVB[2:0]分周比 ÷ DIVR2:DIVP[3:0]分周比

6. クロック・リセット状態遷移

クロック・リセット状態遷移について説明します。

6.1 概要

クロック・リセット状態遷移の概要について説明します。

クロックおよびリセットに関わる状態の遷移について説明します。消費電力制御の状態の特長、設定については『消費電力制御』の章を参照してください。リセットの動作については『リセット』の章を参照してください。レギュレータモードについては『レギュレータ制御』の章を参照してください。

6.2 デバイス状態と各遷移

クロック・リセット状態遷移のデバイス状態と各遷移について説明します。

6.2.1 状態遷移図

6.2.2 各状態の説明

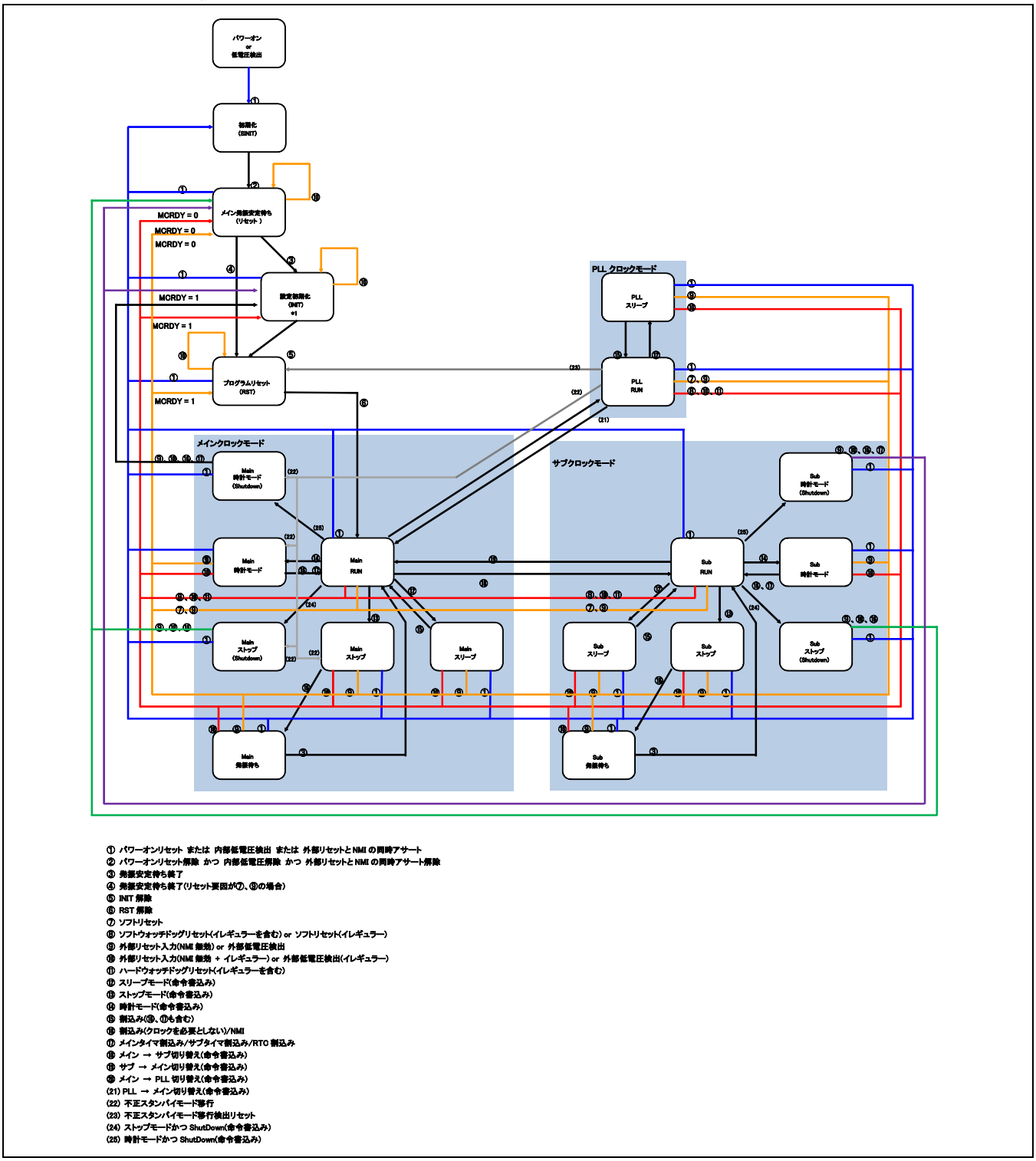
6.2.3 各状態遷移要求の優先順位

6.2.1 状態遷移図

状態遷移図について示します。

本シリーズのデバイス状態遷移を示します。

図 6-1 デバイス状態遷移図



① パワーオンリセット または 内部低電圧検出 または 外部リセットと NMI の同時アサート
② パワーオンリセット解除 かつ 内部低電圧解除 かつ 外部リセットと NMI の同時アサート解除
③ 発振安定待ち終了
④ 発振安定待ち終了(リセット要因が⑦、⑨の場合)
⑤ INIT 解除
⑥ RST 解除
⑦ ソフトリセット
⑧ ソフトウォッチドッグリセット(イレギュラーを含む) or ソフトリセット(イレギュラー)
⑨ 外部リセット入力(NMI 無効) or 外部低電圧検出
⑩ 外部リセット入力(NMI 無効 + イレギュラー) or 外部低電圧検出(イレギュラー)
⑪ ハードウォッチドッグリセット(イレギュラーを含む)
⑫ スリープモード(命令書込み)
⑬ ストップモード(命令書込み)
⑭ 時計モード(命令書込み)
⑮ 割込み(⑱、⑰も含む)
⑯ 割込み(クロックを必要としない)/NMI
⑰ メインタイマ割込み/サブタイマ割込み/RTC 割込み
⑱ メイン → サブ切り替え(命令書込み)
⑲ サブ → メイン切り替え(命令書込み)
⑳ メイン → PLL 切り替え(命令書込み)
(21) PLL → メイン切り替え(命令書込み)
(22) 不正スタンバイモード移行
(23) 不正スタンバイモード移行検出リセット
(24) ストップモードかつ ShutDown(命令書込み)
(25) 時計モードかつ ShutDown(命令書込み)

時計モード(Shutdown)からの復帰時とストップモード(Shutdown)からの復帰時には、リセットされないレジスタがあります。詳細は、『消費電力制御』の章の『電源遮断・通常スタンバイ制御の制限事項』を参照してください。

＜注意事項＞

- OCD ツール接続時は、上図と異なる遷移をする場合があります。『オンチップデバッガ(OCD)』の章を参照してください。
- 1 系統品種ではサブクロック入力がないため、サブクロックモードには遷移しません。

6.2.2 各状態の説明

各状態について説明します。

本シリーズのデバイスの動作状態には以下のものがあります。

■ RUN 状態 (通常動作)

プログラム実行状態です。すべての内部クロックが供給され、すべての回路が動作可能な状態です。ストップ状態と時計モード状態の外部端子のハイインピーダンス制御は解除されます。

■ スリープ状態

プログラム停止状態です。プログラム動作により遷移します。CPUのプログラム実行のみ停止する設定(CPUスリープモード)と、CPUおよびオンチップバス(オンチップバス)およびオンチップバスクロック(HCLK)駆動のペリフェラルを停止させる設定(バススリープモード)があります。詳細は『消費電力制御』の章を参照してください。

■ 時計モード状態

デバイス停止状態です。プログラム動作により遷移します。発振回路(メインクロック生成部、サブクロック生成部)以外の内部回路が停止します。時計モード状態に遷移する前にPLLの発振を停止してください。また、設定により、外部端子を一律ハイインピーダンスにすることが可能です(一部端子を除く)。特定の(クロックを必要としない)有効な割込み、メインタイマ割込み、サブタイマ割込みや時計カウンタ割込みにより、RUN状態へ遷移します。詳細は『消費電力制御』の章を参照してください。

■ 時計モード (電源遮断) 状態

時計モードに不必要な部分の電源を遮断したデバイス停止状態です。プログラム動作により遷移します。内部回路の電源を遮断し、発振回路(メインクロック生成部、サブクロック生成部)以外の内部回路が停止します。時計モード(電源遮断)状態に遷移する前にPLLの発振を停止してください。また、設定により、外部端子を一律ハイインピーダンスにすることが可能です(一部端子を除く)。特定の(クロックを必要としない)有効な割込み、メインタイマ割込み、サブタイマ割込みや時計カウンタ割込みにより、設定初期化(INIT)状態へ遷移します。詳細は『消費電力制御』の章を参照してください。

■ ストップ状態

デバイス停止状態です。プログラム動作により遷移します。すべての内部回路が停止します。ストップ状態に遷移する前にPLLの発振を停止してください。また、設定により、外部端子を一律ハイインピーダンスにすることが可能です(一部端子を除く)。NMI割込みにより、発振安定待ちRUN状態へ遷移します。詳細は『消費電力制御』の章を参照してください。

■ ストップ (電源遮断) 状態

ストップ状態に不必要な部分の電源を遮断したデバイス停止状態です。プログラム動作により遷移します。内部回路の電源を遮断し、すべての内部回路が停止します。ストップ(電源遮断)状態に遷移する前にPLLの発振を停止してください。また、設定により、外部端子を一律ハイインピーダンスにすることが可能です(一部端子を除く)。NMI割込みにより、メイン発振安定待ち(リセット)状態へ遷移します。詳細は『消費電力制御』の章を参照してください。

■ メイン発振安定待ち、サブ発振安定待ち (RUN) 状態

デバイス停止状態です。ストップ状態からの復帰後に遷移します。発振安定待ちのためのタイマ動作を除くすべての内部回路が停止します。内部クロックはすべて停止しますが、動作許可されていた発振回路は動作しています。設定された発振安定待ち時間の経過により、RUN状態(通常動作)へ遷移します。

■ メイン発振安定待ち (リセット) 状態

デバイス停止状態です。初期化(SINIT)状態からの復帰後に遷移します。発振安定待ちのためのタイマ動作を除くすべての内部回路が停止します。内部クロックはすべて停止しますが、メイン発振回路は動作しています。内部回路に対し、プログラムリセット(RST)を出力します。受け付けたリセットレベルがイニシャライズリセットの場合、設定初期化リセット(INIT)も出力します。メインクロック発振安定待ち時間の経過(2^{15}×メインクロック周期)により、設定初期化(INIT)状態へ遷移します。

■ プログラムリセット (RST) 状態

プログラム初期化状態です。動作初期化リセット(RST)要求の受付、または設定初期化(INIT)状態の終了により遷移します。内部回路に対し、プログラムリセット(RST)を出力します。INITから遷移してきた場合、OCDチップリセットシーケンス(1026+3 PCLKサイクル)をとります。

動作初期化リセット(RST)要求の消失により、RUN状態(通常動作)へ遷移します。詳細は『リセット』の章を参照してください。

■ 設定初期化 (INIT) 状態

全設定初期化状態です。設定初期化(INIT)要求の受付により遷移します。メイン発振回路は動作しますが、サブ発振回路とPLLは動作を停止します。内部回路に対し、設定初期化(INIT)およびプログラムリセット(RST)を出力します。設定初期化(INIT)要求の消失により、本状態は解除され、プログラムリセット(RST)状態へ遷移します。詳細は『リセット』の章を参照してください。

6.2.3 各状態遷移要求の優先順位

各状態遷移要求の優先順位について示します。

どの状態においても、各状態遷移要求は以下の優先順位に従います。ただし、一部要求は特定の状態でしか発生しませんので、その状態でしか有効になりません。

【最強】 初期化 (SINIT) 要求
↓ 設定初期化(INIT)要求
↓ 発振安定待ち時間の終了
(発振安定待ちリセット状態および発振安定待ちRUN状態のみ発生)
↓ プログラムリセット(RST)要求
↓ 有効な割込み要求
(RUN、スリープ、ストップ、時計モード状態のみ発生)
↓ ストップモード要求 (レジスタ書込み) (RUN状態のみ発生)
↓ 時計モード要求 (レジスタ書込み) (RUN状態のみ発生)
【最弱】 スリープモード要求 (レジスタ書込み) (RUN状態のみ発生)

6.3 デバイス状態と対応するレギュレータモード

デバイス状態と対応するレギュレータモードについて示します。

下表に各デバイス状態に対応するレギュレータモードを示します。レギュレータモードについては『レギュレータ制御』の章を参照してください。

表 6-1 デバイス状態とレギュレータモードの関係 (クロック 1 系統)

デバイス状態	メインクロック	レギュレータモード
Main RUN	発振	メインモード
Main スリープ	発振	メインモード
Main 時計モード	発振	スタンバイモード
Main 時計モード(Shutdown)	発振	スタンバイモード
Main ストップ	停止	スタンバイモード
Main ストップ(Shutdown)	停止	スタンバイモード
Main 発振待ち	発振	メインモード
PLL RUN	発振	メインモード
PLL スリープ	発振	メインモード

表 6-2 デバイス状態とレギュレータモードの関係 (クロック 2 系統)

デバイス状態	メインクロック	サブクロック	レギュレータモード
Main RUN	発振	発振 または停止	メインモード
Main スリープ	発振	発振 または停止	メインモード
Main 時計モード	発振	発振 または停止	スタンバイモード
Main 時計モード(Shutdown)	発振	発振 または停止	スタンバイモード
Main ストップ	停止	停止	スタンバイモード
Main ストップ(Shutdown)	停止	停止	スタンバイモード
Main 発振待ち	発振	発振または停止	メインモード
Sub RUN 1	発振	発振	メインモード
Sub RUN 2	停止	発振	サブモード
Sub スリープ 1	発振	発振	メインモード
Sub スリープ 2	停止	発振	サブモード
Sub 時計モード	発振 または停止	発振	スタンバイモード
Sub 時計モード(Shutdown)	発振 または停止	発振	スタンバイモード
Sub ストップ	停止	停止	スタンバイモード
Sub ストップ(Shutdown)	停止	停止	スタンバイモード
Sub 発振待ち 1	発振	発振	メインモード
Sub 発振待ち 2	停止	発振	サブモード
PLL RUN	発振	発振または停止	メインモード
PLL スリープ	発振	発振 または停止	メインモード

＜注意事項＞

上記いずれの表でも、OCDツール接続時、レギュレータモードはメインモードとなります。

7. リセット

リセットについて説明します。

7.1 概要

リセットの概要について説明します。

リセット要因が発生すると、デバイスはすべてのプログラムおよび大部分のハードウェア動作を停止し、状態を初期化します。この状態をリセットとよびます。

7.2 特長

リセットの特長について説明します。

本シリーズは、以下のリセット要因を持ち、各要因の受付によりデバイス内部の初期化のためのリセットを発行します。

- パワーオンリセット
- RSTX 端子入力
- ウォッチドッグリセット 0 (ソフトウェアウォッチドッグ)
- ウォッチドッグリセット 1 (ハードウェアウォッチドッグ)
- ソフトウェアリセット
- 不正スタンバイモード移行検出リセット
- フラッシュセキュリティ違反
- 内部低電圧検出
- 外部低電圧検出
- クロックスーパバイザリセット
- スタンバイ(電源遮断)からの復帰リセット

イレギュラーリセットになる場合を除き(「7.4.1. リセット要因レジスタ : RSTRR (ReSeT Result Register)」参考)、リセット発行はすべてのバスアクセスの完了を確認後に行うため、リセットによるアクセス中のメモリ内容(RAM、Flash)は破壊されません。

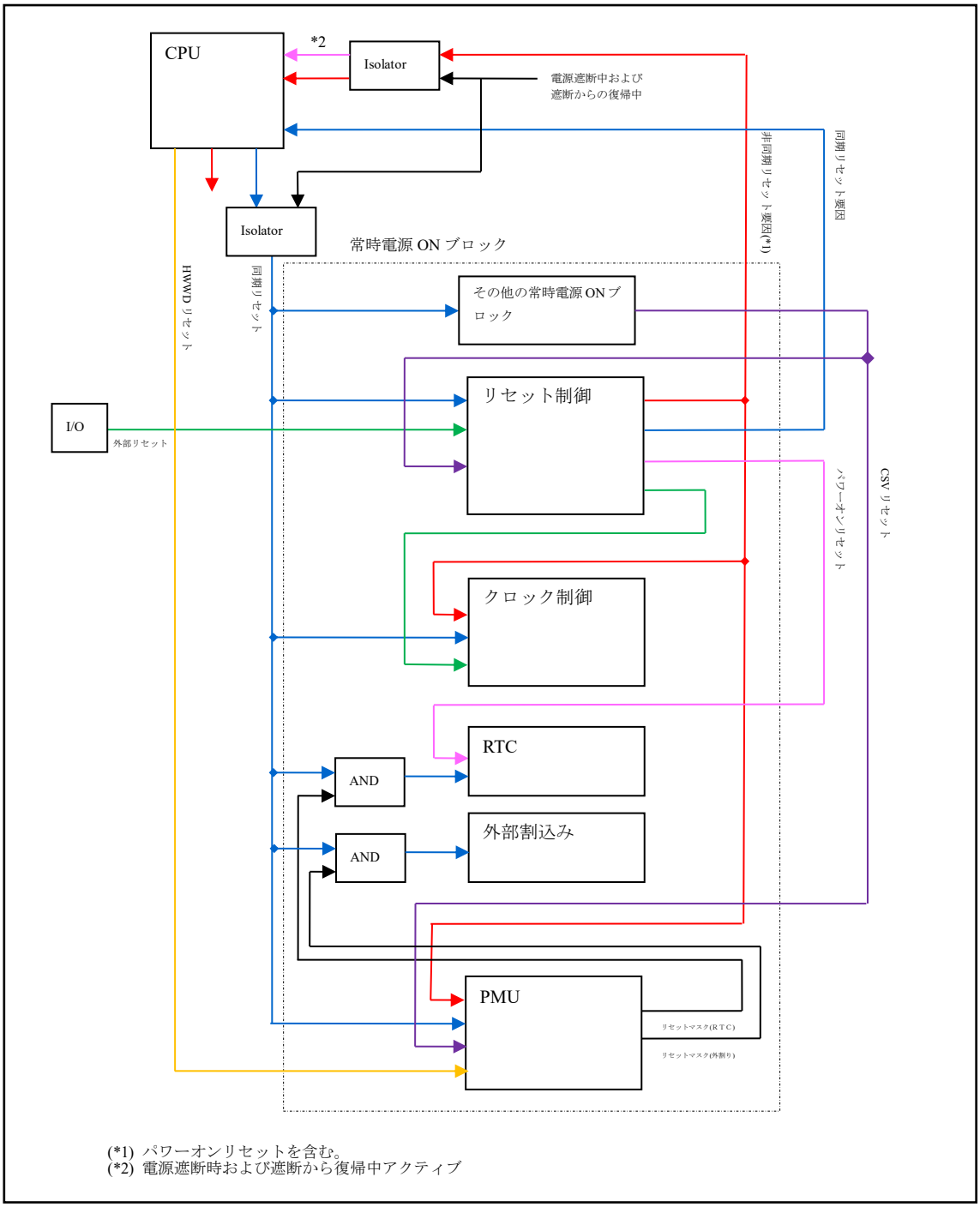
バスの応答が一定時間内に返されない時の強制リセット発行のために、リセット発行遅延カウンタを持ち、設定された時間内に応答がない時はバスの応答の有無にかかわらずリセットを発行します。(リセットタイムアウト)

クロックスーパバイザリセットについては、『CHAPTER 31: クロックスーパバイザ』の章を参照してください。

7.3 構成

リセットの構成について説明します。

図 7-1 リセット構成図



(*1) パワーオンリセットを含む。
(*2) 電源遮断時および遮断から復帰中アクティブ

図 7-2 リセット構成図 (リセット制御)

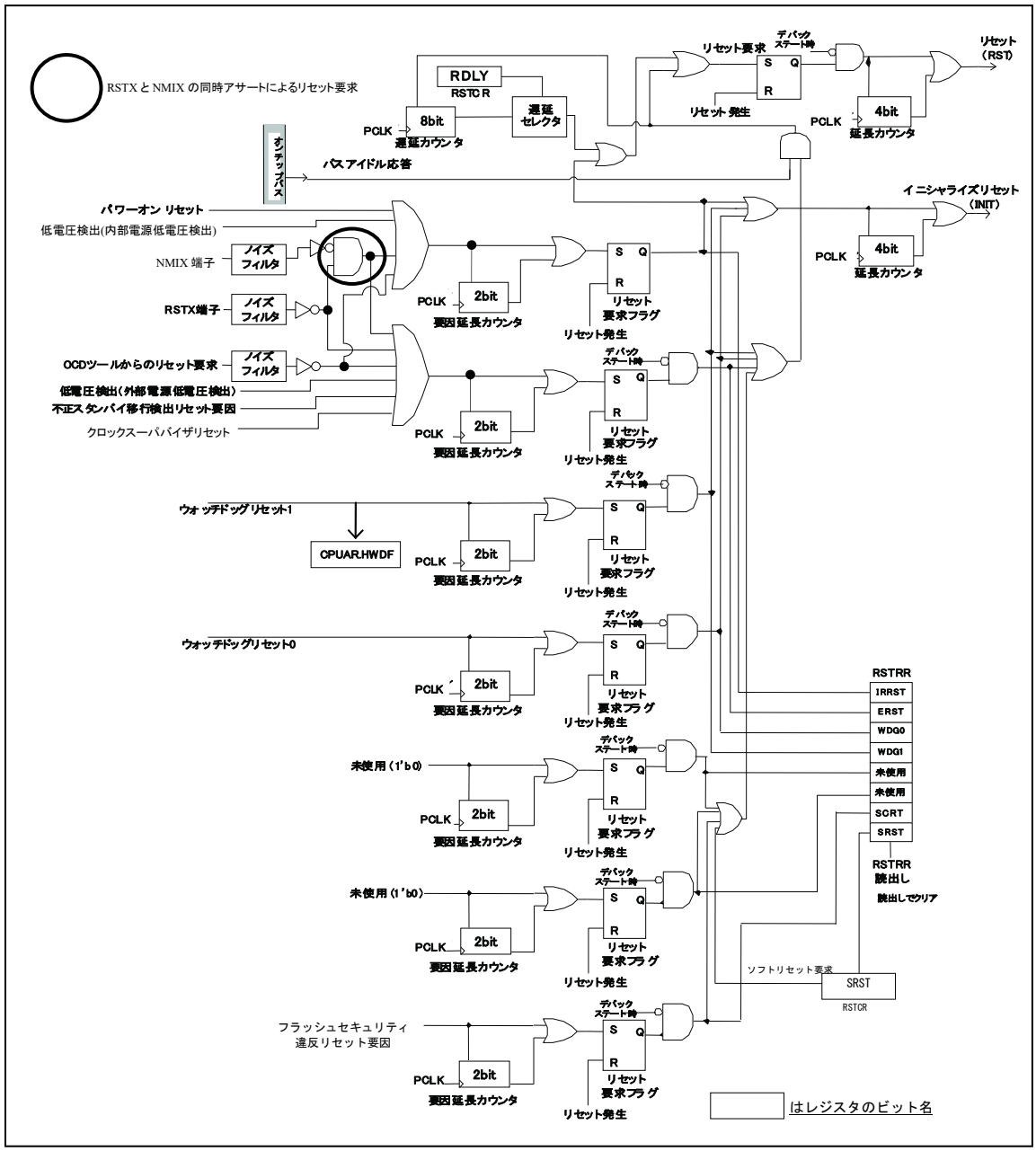
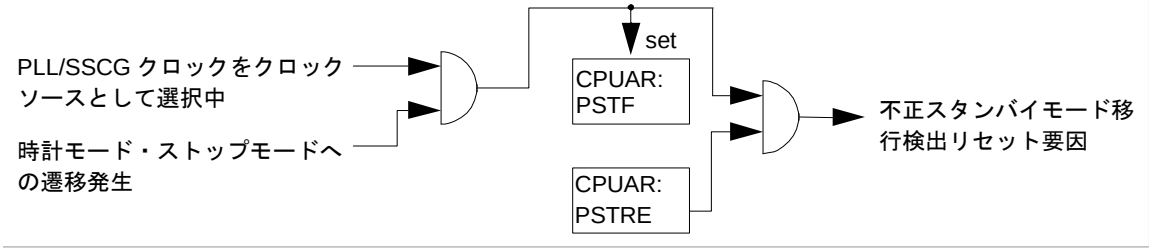


図 7-3 不正スタンバイモード移行検出リセット要因の生成図



7.4 レジスタ

リセットのレジスタについて説明します。

表 7-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0480	RSTRR	RSTCR	予約	予約	リセット要因レジスタ リセット制御レジスタ
0x0518	予約	予約	CPUAR	予約	CPU 異常動作レジスタ
0x0590	PMUSTR	予約	予約	予約	PMU ステータスレジスタ

＜注意事項＞

0x0482, 0x0591, 0x0592番地には、『CHAPTER 27: 消費電力制御』の章のレジスタが割り当てられているので注意してください。

7.4.1 リセット要因レジスタ ：RSTRR (ReSeT Result Register)

リセット要因レジスタのビット構成について説明します。

直前までに発生した各種リセット要因を表示します。

＜注意事項＞

このレジスタを読み出すと、すべてのビットがクリアされます。
デバッグステート中の読出しではクリアされません。
デバッグステート中は、各リセット要因がマスクされるため、本レジスタもリセット要因を検出しません。

■ RSTRR ：アドレス 0480H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IRRST	ERST	WDG1	WDG0	予約		SCRT	SRST
初期値	*	*	*	*	—	—	*	*
属性	R,WX	R,WX	R,WX	R,WX	RX,WX	RX,WX	R,WX	R,WX

*リセット要因による

[bit7] IRRST (IRregular ReSeT) : イレギュラーリセット

パワーオンリセット、内部低電圧検出、またはリセットタイムアウト、またはRSTX外部端子とNMIX外部端子の同時アサートのいずれかが発生し、リセット発行時のバスアクセス状態を保証できないことを示します。リセット後本ビットが"0"のとき、直前のリセット時にはバスアクセスが行われず、メモリ内容がリセットにより破壊されていないことを保証します。リセット後の本ビットが"1"のとき、直前のリセット時にはバスアクセスが行われていた可能性があり、メモリ内容がリセットにより破壊されていないことが保証されません。

IRRST	イレギュラーリセット検出
0	イレギュラーリセット未検出
1	イレギュラーリセット検出

本ビットは読出しでクリアされます。

[bit6] ERST (External ReSeT) : **リセット端子入力・不正スタンバイモード移行検出・外部低電圧検出・クロックスーパバイザリセット・RSTX外部端子とNMIX外部端子の同時アサート**

RSTX端子入力からのリセット入力、不正スタンバイモード移行検出リセット、外部低電圧検出、クロックスーパバイザリセット、またはRSTX外部端子とNMIX外部端子の同時アサートの発生を示します。

本リセット要因でリセットタイムアウトが発生していたとき、本ビットと共にIRRST=1となります。

ERST	RSTX 端子リセット検出，不正スタンバイモード移行検出， 外部低電圧検出，クロックスーパバイザリセット検出，または RSTX 外部端子と NMIX 外部端子の同時アサート検出
0	未検出
1	検出

本ビットは読出しでクリアされます。

[bit5] WDG1 (WatchDoG reset 1) : **ウォッチドッグリセット1**

ウォッチドッグタイマ1からのリセットを示します。

本リセット要因でリセットタイムアウトが発生していたとき、本ビットと共にIRRST="1"となります。

WDG1	ウォッチドッグタイマ 1 リセット
0	未検出
1	検出

本ビットは読出しでクリアされます。

CPUARレジスタにも、ウォッチドッグリセット1によるリセット要因発生を示すフラグがあります。そちらでは読み出してもクリアされません。

[bit4] WDG0 (WatchDoG reset 0) : **ウォッチドッグリセット0**

ウォッチドッグタイマ0からのリセットを示します。

本リセット要因でリセットタイムアウトが発生していたとき、本ビットと共にIRRST=1となります。

WDG0	ウォッチドッグタイマ 0 リセット
0	未検出
1	検出

本ビットは読出しでクリアされます。

[bit1] SCRT (Flash SeCuRiTy violation) : **フラッシュセキュリティ違反リセット**

フラッシュメモリのセキュリティ違反リセットが発生したことを示します。

本リセット要因でリセットタイムアウトが発生していたとき、本ビットと共にIRRST=1となります。

SCRT	フラッシュセキュリティ違反リセット
0	未検出
1	検出

本ビットは読出しでクリアされます。

[bit0] SRST (Software ReSeT) : ソフトウェアリセット

RSTCR:SRSTビットへの"1"書込みによるリセットを示します。

本リセット要因でリセットタイムアウトが発生していたとき、本ビットと共にIRRST=1となります。

SRST	ソフトウェアリセット
0	未検出
1	検出

本ビットは読出しでクリアされます。

7.4.2 リセット制御レジスタ ：RSTCR (ReSeT Control Register)

リセット制御レジスタのビット構成について説明します。

各種リセット発行制御を行うレジスタです。

■ RSTCR ：アドレス 0481H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RDLY[2:0]			予約				SRST
初期値	1	1	1	0	0	0	0	0
属性	R,W	R,W	R,W	R/W	R/W	R/W	R/W	R,W

[bit7~bit5] RDLY[2:0] (Reset DeLaY) : リセット発行遅延

リセットタイムアウト値を設定します。リセット要因が検出されてから、すべてのバスがアイドルになるか、本ビットによるリセットタイムアウトまでカウントされるとリセットが発行されます(後者の場合､イレギュラーリセットになります)。本ビットはリセット後1回のみ書込みが可能です。

RDLY[2:0]	リセットタイムアウト値
000	PCLK × 2 サイクル
001	PCLK × 4 サイクル
010	PCLK × 8 サイクル
011	PCLK × 16 サイクル
100	PCLK × 32 サイクル
101	PCLK × 64 サイクル
110	PCLK × 128 サイクル
111	PCLK × 256 サイクル (初期値)

[bit0] SRST (Software ReSeT) : ソフトウェアリセット

本ビットに"1"を書き込んだ後、RSTCRを読み出すことによりソフトウェアリセット要求を発生します。

本ビットに"1"を書き込んだ後は、リセットが発生するまでRSTCRへの書込みは無視され、レジスタ値を書き換えることはできません。

デバッグステート中のRSTCR読出しでは、リセットは発生しません。

SRST	ソフトウェアリセット
0	出力しない(初期値)
1	RSTCR 読出しによりセット要求を出力する

7.4.3 CPU 異常動作レジスタ ：CPUAR (CPU Abnormal operation Register)

CPU異常動作レジスタのビット構成について説明します。

CPU異常動作時に関する設定、状態を示すレジスタです。

■ CPUAR ：アドレス 051AH (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PSTRE	予約				PMDF	PSTF	HWDF
初期値	0	0	0	0	*	*	*	*
属性	R/W	R0,WX	R0,WX	R0,WX	RX,WX	R(RM1), W	R(RM1), W	R(RM1), W

* RSTX端子のアサートで"0"に初期化されます。(NMIXとの同時アサートも含みます。)それ以外のリセット要因では初期化されません。

[bit7] PSTRE (illegal PLL-run to STandby Reset Enable) : 不正スタンバイモード移行検出リセット許可

PLLクロックをクロックソースとして選択時、時計モードまたはストップモード遷移を検出したときに(不正スタンバイモード移行)、リセットを発行するかどうかを設定します。

許可している場合、PLLラン状態から時計モードまたはストップモードへ遷移すると不正スタンバイモード移行検出要因によるリセットが発生します。

PSTRE	説明
0	リセットを発生しない (初期値)
1	リセット発生許可

＜注意事項＞

本ビットをセットする場合は、本ビットをセットする前にPSTFビットに"0"書き込みしてPSTFビットをクリアしてください。PSTFビットのパワーオンリセット後の値は不定のため、PSTFビットをクリアする前に本ビットをセットするとリセットが発生する場合があります。

[bit2] PMDF (PLL mode Main clock Down detection Flag) : PLLモードメイン発振断検出フラグ

PLL 出力をクロックソースとして選択時、クロックスーパーバイザがメイン発振断検出をした場合に、本ビットがセットされます。またその場合、ソースクロックは自動的にメインモード(CKS=CKM=00)に書き換わり、すぐにリセット(RSTレベル)が発生します。
リードモディファイライト系命令では"1"が読み出されます。

PMDF	読出し	書込み
0	PLL モード中にメイン発振断検出なし (初期値)	本ビットをクリア
1	PLL モード中にメイン発振断検出あり	効果ありません

セット要因とクリア要因が同時に発生した場合、セット要因が優先されます。

[bit1] PSTF (illegal PLL-run to STandby Flag) : **不正スタンバイモード移行検出フラグ**

PLLクロックをクロックソースとして選択時、時計モードまたはストップモード遷移を検出したときに(不正スタンバイモード移行)、本ビットがセットされます。またその場合、ソースクロックは自動的にメインモード(CKS=CKM=00)に書き換わります。PSTREビットが"1"の場合にはリセット(RSTレベル)が発生します。
"0"書込みで本ビットをクリアします。
リードモディファイライト系命令では"1"が読み出されます。

PSTF	読出し	書込み
0	不正スタンバイモード移行は検出されていません	本ビットをクリア
1	不正スタンバイモード移行を検出しました	効果ありません

[bit0] HWDF (Hardware WatchDog Flag) : **ハードウェアウォッチドッグ検出フラグ**

ウォッチドッグタイマ1 (ハードウェアウォッチドッグ)リセット要因が検出されると、本ビットがセットされます。
"0"書込みで本ビットをクリアします。
リードモディファイライト系命令では"1"が読み出されます。

HWDF	読出し	書込み
0	ウォッチドッグタイマ 1 (ハードウェアウォッチドッグ) リセット要因は発生していません	本ビットをクリア
1	ウォッチドッグタイマ 1 (ハードウェアウォッチドッグ) リセット要因が発生しました	効果ありません

セット要因とクリア要因が同時に発生した場合、セット要因が優先されます。

＜注意事項＞

RSTRR:WDG1にも検出フラグはありますが、こちらはリードクリアであるため、一度リードを行うと要因が消えてしまいます。CPUAR:HWDFは保持されているのでクリアするまで要因が保持されています。

7.4.4 PMU ステータスレジスタ ：PMUSTR (Power Management Unit STatus register)

PMUステータスレジスタのビット構成について説明します。

PMUのステータスを示すレジスタです。

■ PMUSTR ：アドレス 0590H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PMUST	予約					PONR_F	RSTX_F
初期値	0	0	0	0	0	0	1	*
属性	R,W	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R,W	R,W

*RSTX端子のアサートで"1"に初期化されます。(NMIXとの同時アサートも含みます。)それ以外のリセット要因では初期化されません。

[bit7] PMUST (Power Management Unit STatus)

直前の状態が、シャットダウンモードであったかどうかの情報を示します。

PMUST	PMU ステータス
0	初期状態、初期化リセットからの動作復帰
1	ShutDown モードからの動作復帰

ビットは、"0"書込みでクリアします。"1"書込みは、無効です。

本ビットは､パワーオンリセット､低電圧検出リセットおよびRSTXとNMIXの同時アサートでのみ初期化され､他のリセット要因では初期化されません。そのため、本ビットでシャットダウンからの復帰を確認する前に､他のリセット要因を確認してください。

[bit6~bit2] 予約

常に"0" が読み出されます。必ず"0"を書き込んでください。

[bit1] PONR_F(Power ON Reset Flag)

パワーオンリセット検出フラグです。

PONR_F	パワーオンリセット
0	検出していない
1	検出した

ビットは、"0"書込みでクリアします。"1"書込みは、無効です。
本ビットは、パワーオンリセット以外のリセット要因では初期化されません。

[bit0] RSTX_F(ReSeTX input Flag)

外部リセット検出フラグです。

RSTX_F	RSTX 入力リセット
0	検出していない
1	検出した

ビットは、"0"書込みでクリアします。"1"書込みは、無効です。
本ビットはパワーオンリセットでは初期化されません。必ずクリアしてから使用してください。

7.5 動作説明

以下、本シリーズのリセットの各動作について説明します。

7.5.1 リセットレベル

7.5.2 リセット要因

7.5.3 リセット受付

7.5.4 リセット発行

7.5.5 リセットシーケンス

7.5.6 注意事項

7.5.1 リセットレベル

リセットレベルについて説明します。

リセットには以下の2つのレベルがあります。

＜注意事項＞

本シリーズでは、デバッグインタフェース部(OCDU)用レジスタを除き、双方のレベルのリセットで初期化されるレジスタは同一です。

7.5.1.1 イニシャライズリセット (INIT)

イニシャライズリセット(INIT)について説明します。

すべてのレジスタ設定、およびすべてのハードウェアを初期化します。CPUのプログラム実行は停止し、プログラムカウンタは初期化されます。周辺回路はすべて初期化されます。メイン発振回路は動作継続、停止していた場合は動作再開しますが、サブ発振回路とPLLは動作を停止します。

以下のリセット要因によるリセット時のみ、このリセットレベルとなります。

- イレギュラーリセット
- ウォッチドッグリセット 0, 1

このリセットレベルでのみ初期化されるのは以下のレジスタです。

- デバッグインタフェース部 (OCDU)のレジスタ

7.5.1.2 リセット (RST)

リセット(RST)について説明します。

イニシャライズリセット(INIT)のみで初期化されるレジスタを除くすべてのレジスタ､およびすべてのハードウェアを初期化します。CPUのプログラムは停止し、プログラムカウンタは初期化されます。周辺回路はすべて初期化されます。

イニシャライズリセット(INIT)が発行されると、同時にリセット(RST)も発行されます。

すべてのドキュメント内にて、特に指定がない場合のリセットは、本リセットレベルを示します。

7.5.2 リセット要因

本品種の各リセット要因について説明します。

7.5.2.1. パワーオンリセット

7.5.2.2. RSTX端子入力

7.5.2.3. ウォッチドッグリセット0

7.5.2.4. ウォッチドッグリセット1

7.5.2.5. 外部低電圧検出リセット

7.5.2.6. 不正スタンバイモード移行検出リセット

7.5.2.7. 内部低電圧検出リセット

7.5.2.8. フラッシュセキュリティ違反リセット

7.5.2.9. ソフトウェアリセット (RSTCR:SRST)

7.5.2.10. スタンバイ（電源遮断）からの復帰

7.5.2.1 パワーオンリセット

パワーオンリセットについて示します。

電源の立上りを検出することにより発生するリセット要因です。

本リセット要因によるリセットは常にイレギュラーリセットとして検出され、イニシャライズリセット (INIT) を発行します。

7.5.2.2 RSTX 端子入力

RSTX端子入力について示します。

デバイス外部から入力されるハードウェアリセットです。

本リセット要因によるリセットは、リセットタイムアウト時、またはNMIX端子が同時にアサートされていた場合にはイレギュラーリセットとして検出されます。
イレギュラーリセット検出時以外は、リセット (RST) のみを発行します。

7.5.2.3 ウォッチドッグリセット 0

ウォッチドッグリセット0について示します。

FR81Sコア内蔵のウォッチドッグタイマ0(ソフトウェアウォッチドッグ)から入力されるハードウェアリセットです。

本リセット要因によるリセットは、リセットタイムアウト時のみイレギュラーリセットとして検出されます。
イレギュラーリセット検出の有無にかかわらず、イニシャライズリセット (INIT) を発行します。

7.5.2.4 ウォッチドッグリセット 1

ウォッチドッグリセット1について示します。

FR81Sコア内蔵のウォッチドッグタイマ1 (ハードウェアウォッチドッグ) から入力されるハードウェアリセットです。

本リセット要因によるリセットは、リセットタイムアウト時のみイレギュラーリセットとして検出されます。イレギュラーリセット検出の有無にかかわらず、イニシャライズリセット(INIT)を発行します。

7.5.2.5 外部低電圧検出リセット

外部低電圧検出リセットについて示します。

低電圧検出(外部電圧)はデバイス内部の低電圧検出回路から入力されるハードウェアリセットです。本リセット要因によるリセットは、リセットタイムアウト時のみイレギュラーリセットとして検出されます。イレギュラーリセット検出時以外は、リセット (RST) のみを発行します。

検出電圧については『低電圧検出 (外部低電圧検出)』の章を参照してください。

7.5.2.6 不正スタンバイモード移行検出リセット

不正スタンバイモード移行検出リセットについて示します。

PLLクロックをクロックソースとして選択時、時計モードまたはストップモード遷移を検出したときに(不正スタンバイモード移行)発生されるハードウェアリセットです。本リセット要因によるリセットは、リセットタイムアウト時のみイレギュラーリセットとして検出されます。イレギュラーリセット検出時以外は、リセット(RST)のみを発行します。

7.5.2.7 内部低電圧検出リセット

内部低電圧検出リセットについて示します。

低電圧検出(内部電圧)はデバイス内部の低電圧検出回路から入力されるハードウェアリセットです。本リセット要因によるリセットは、イレギュラーリセットとして検出されイニシャライズリセット(INIT)を発行します。

検出電圧については『低電圧検出(内部低電圧検出)』の章を参照してください。

7.5.2.8 フラッシュセキュリティ違反リセット

フラッシュセキュリティ違反リセットについて示します。

フラッシュメモリのセキュリティ保護違反が発生した場合に発行されるリセットです。

本リセット要因によるリセットは、リセットタイムアウト時のみイレギュラーリセットとして検出されます。

イレギュラーリセット検出時以外は、リセット(RST)のみを発行します。

7.5.2.9 ソフトウェアリセット (RSTCR:SRST)

ソフトウェアリセット(RSTCR:SRST)について示します。

デバイス内部で発生するソフトウェアリセットです。

RSTCRのbit0:SRSTビットに"1"を書き込んだ後、RSTCRを読み出すことにより発生します。

本リセット要因によるリセットは、リセットタイムアウト時のみイレギュラーリセットとして検出されます。イレギュラーリセット検出時以外は、リセット(RST)のみを発行します。

[例] ソフトウェアリセット発行のサンプルプログラム

```
LDI     #value_of_reset, R0     ; SRSTビット=1
LDI     #_RSTCR, R12            ;
STB     R0, @R12                ; ライト
LDUB    @R12, R0                ; リード(ソフトウェアリセット要求発生)
MOV     R0, R0                  ; パイプライン調整のためのダミー処理
NOP                             ; パイプライン調整のためのダミー処理
```

7.5.2.10 スタンバイ (電源遮断) からの復帰

スタンバイ(電源遮断)からの復帰について示します。

大部分のブロックが、スタンバイからの起動により、電源投入リセット相当の動作をします。ただし、パワーオンリセット要因は常時電源ONのブロックにあるため、リセット要因レジスタ(RSTRR)レジスタに検出が表示されません。要因は、PMUステータスレジスタ(PMUSTR)に表示されますので、マイコンが再起動した場合には、本レジスタを確認してください。

本リセット要因によるリセットはイニシャライズリセット(INIT)を発行します。

7.5.3 リセット受付

各リセット要因の受付処理について説明します。

7.5.3.1. リセット要求の生成

7.5.3.2. リセット要求の受理

7.5.3.3. リセット発行遅延カウンタ

7.5.3.4. イレギュラーリセット

7.5.3.1 リセット要求の生成

リセット要求の生成について示します。

少なくとも1つのリセット要因が取り込まれると、リセット要求を生成します。リセット要求は内部バス制御部へ通知され、以下の処理を行います。

- CPU のプログラム動作の停止 (スリープモードと同一処理)
- オンチップバスのバス制御権の取得
- すべてのバスへアイドル要求が通知されたことの確認

7.5.3.2 リセット要求の受理

リセット要求の受理について示します。

リセット要求に対するすべての処理が完了すると、リセット発行部にてリセット要求が受理され、リセット要因に応じたレベルのリセットを発行します。また、リセット発行遅延カウンタのオーバフロー＝リセットタイムアウトが発生すると、リセット要求に対する処理の終了を待たずにリセット要求が受理され、イレギュラーリセットが発行されます。

7.5.3.3 リセット発行遅延カウンタ

リセット発行遅延カウンタについて示します。

リセット要求が生成されると同時に、8ビット長のリセット発行遅延カウンタがカウントを開始します。リセットが発行されないまま、RSTCRレジスタのbit7~bit5:RDLY[2:0]ビットで指定された遅延サイクルが経過し、カウンタがオーバフローする＝リセットタイムアウトが発生すると、イレギュラーリセットが発行されます。

RSTCRのRDLY[2:0]ビットはリセットにより初期化され、リセット解除後は1回のみ書換えが可能です。遅延サイクルの設定が短い場合、イレギュラーリセットが発行される可能性が高くなります。遅延サイクルの設定が長い場合、リセット要因が発生してからリセットが発行される迄に長時間かかる場合があります。

7.5.3.4 イレギュラーリセット

イレギュラーリセットについて示します。

リセット要求に対する処理完了を確認せずにリセットが発行されたとき、イレギュラーリセットとなります。

イレギュラーリセットが発生すると、以下の処理が行われます。

- リセット要因の種類にかかわらず、イニシャライズリセット(INIT)を発行します。
- RSTRR レジスタの bit7:IRRST ビットを"1"にセットします。

イレギュラーリセットが発生したときは、リセットが入力される時点でバスアクセスが行われていた可能性があり、メモリ内容がリセットにより破壊されていないことが保証できません。イレギュラーリセットの発生は必ずしもメモリの内容が破壊されたことを示す訳ではありませんが、その際のバスアクセスの内容を特定できないためです。

7.5.4 リセット発行

リセット要求受理後、リセットが発行されます。以下、各種リセット発行について説明します。

7.5.4.1. 電源投入リセット (SINIT)

7.5.4.2. イニシャライズリセット (INIT)

7.5.4.3. リセット (RST)

7.5.4.1 電源投入リセット (SINIT)

電源投入リセット(SINIT)について示します。

パワーオンリセット、または内部低電圧検出、またはRSTX・NMIX同時アサート時に最初に電源投入リセット(SINIT)が発行されます。本リセットは分周回路などの不定状態を初期化するためのみに限定的に使用されます。

本リセット発行中はすべてのクロックが停止します。

本リセットが発行される時、必ず同時にイニシャライズリセット(INIT)およびリセット(RST)を発行します。

本リセットにより、クロック制御レジスタが初期化されます

本リセットはメインクロック発振安定待ちを伴います。制御レジスタ初期化に伴い、発振安定待ち時間は2^{15}×メインクロック周期になります。

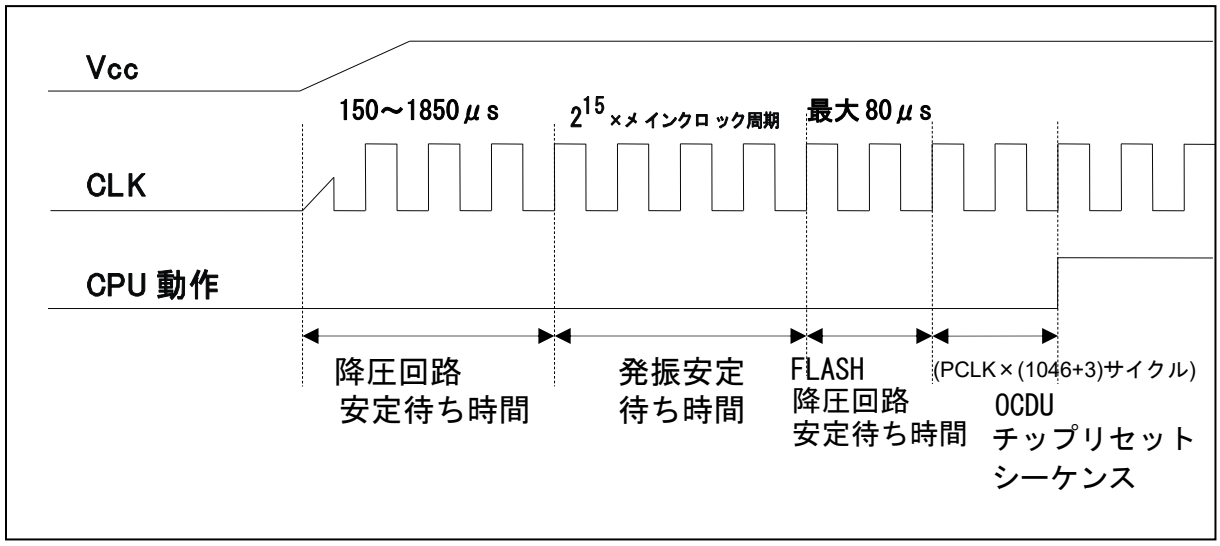
表 7-2 発振安定待ち時間 (SINIT)

種類	メインクロック発振安定待ち時間
パワーオンリセット	2^{15}×メインクロック周期
内部低電圧検出	2^{15}×メインクロック周期
RSTX・NMIX 同時アサート	2^{15}×メインクロック周期

＜注意事項＞

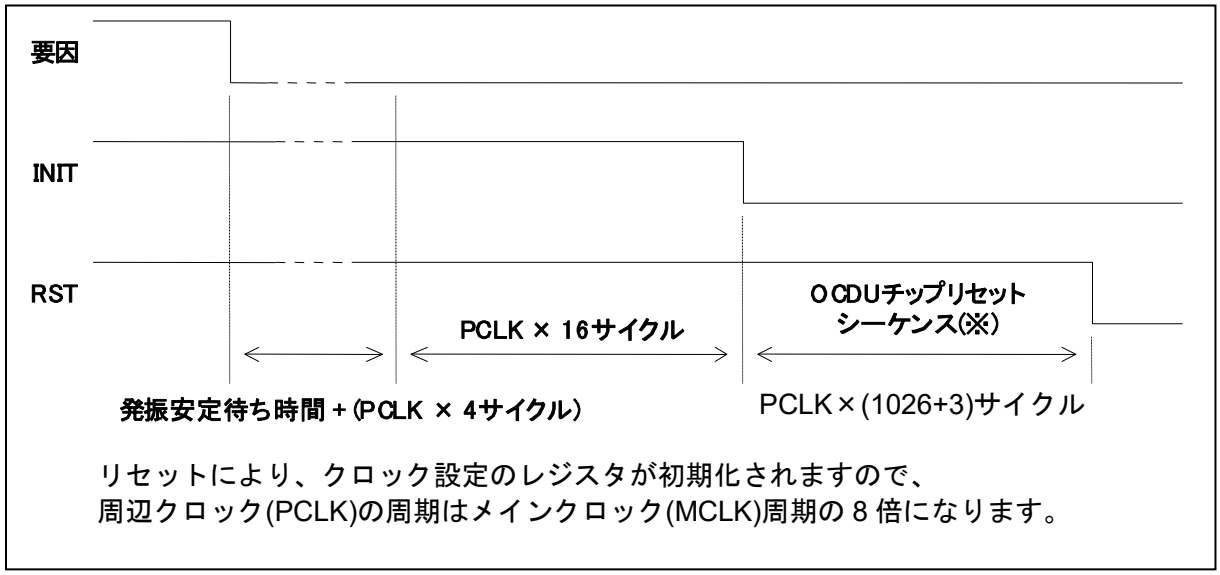
電源投入・電圧復帰に関わる、レギュレータ安定待ち時間およびFLASH安定待ち時間は上表の発振安定待ち時間には含まれていません。パワーオンリセット時には、これらの安定待ち時間(150～1850µsおよび最大80µs)が必要になります。

図 7-4 パワーオンリセット時の発振安定待ち時間



本リセットのリセット要因が解除された後の各リセット発行シーケンスを以下に示します。

図 7-5 電源投入リセット(SINIT)シーケンス



7.5.4.2 イニシャライズリセット (INIT)

イニシャライズリセット(INIT)について示します。

イニシャライズリセット(INIT)レベルのリセット要因が発生したとき、イニシャライズリセット(INIT)およびリセット(RST)を最初に同時に発行します。本リセットはリセット(RST)で初期化されない一部のレジスタの初期化のみに使用します。

本リセット発行中はすべてのクロックが動作します。本リセットが発行される場合、必ず同時にリセット(RST)を発行します。本リセットにより、クロック制御レジスタが初期化されますが、メインクロック(MCLK)が発振中の場合、発振しているという動作自体は変化しません。

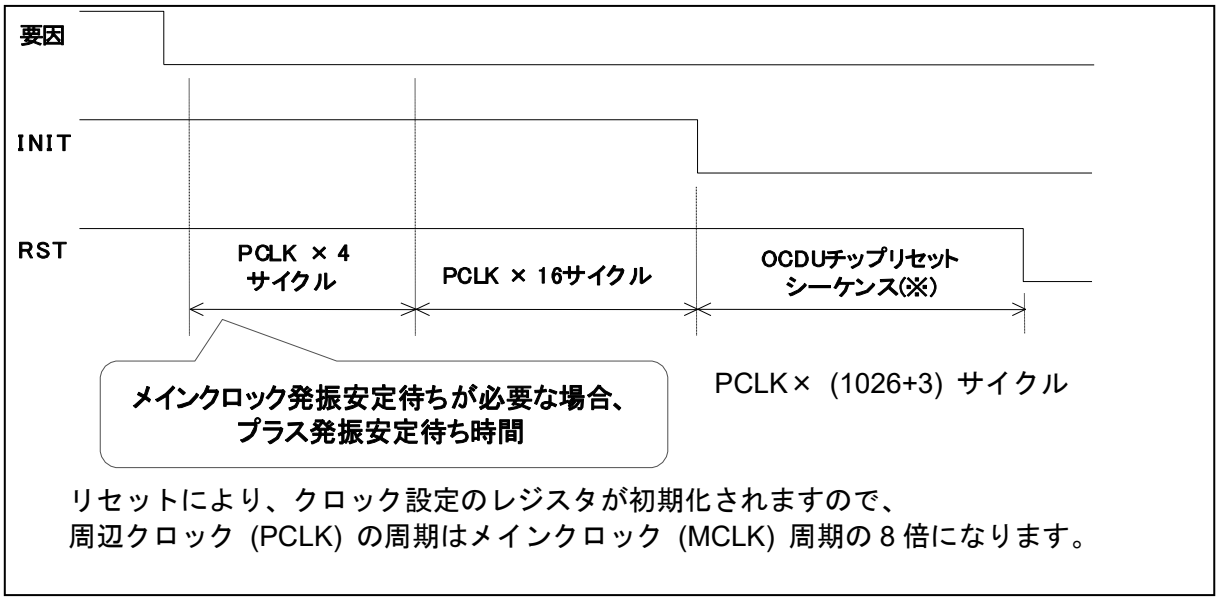
ストップモード中など、メインクロック停止中だった場合、メインクロック発振安定待ち時間を取ります。クロック制御部のレジスタがリセットにより初期化されるため、発振安定待ち時間は本シリーズのデフォルト値(2^{15}×メインクロック周期)となります。

表 7-3 発振安定待ち時間 (INIT)

リセット投入前メインクロック発振停止中？	メインクロック発振安定待ち時間
No	なし
Yes	2^{15}×メインクロック周期

本リセットのリセット要因が解除された後の各リセット発行シーケンスを以下に示します。

図 7-6 イニシャライズリセット (INIT) シーケンス



7.5.4.3 リセット (RST)

リセット (RST) について示します。

イニシャライズリセット (INIT) レベルでないリセット要因が発生した場合、リセット(RST)のみを発行します。

本リセットは一部のレジスタ(7.5.1.1参照)を除くすべてのハードウェアの初期化に使用します。

本リセット発行中はすべてのクロックが動作します。

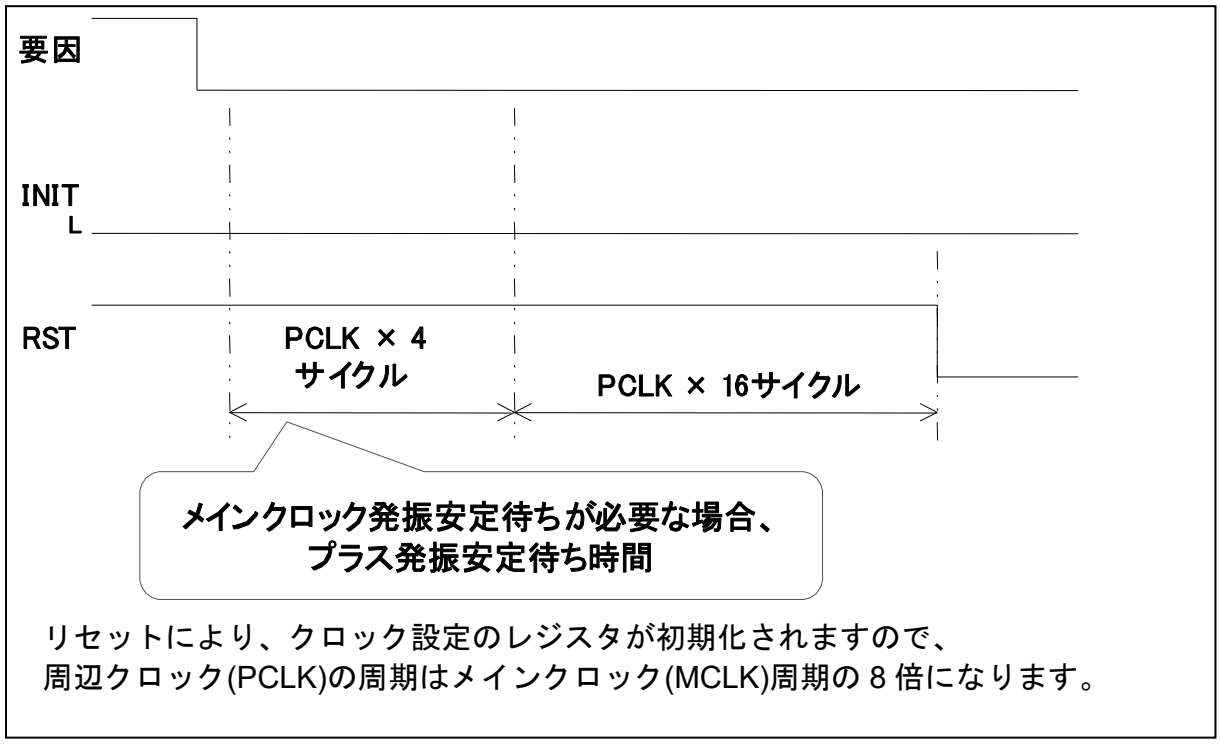
リセット前にストップモード中などでメインクロック停止中だった場合、メインクロック発振安定待ち時間を取ります。クロック制御部のレジスタがリセットにより初期化されるため、発振安定待ち時間は本品種のデフォルト値(2^{15}×メインクロック周期)となります。

表 7-4 発振安定待ち時間 (RST)

リセット投入前メインクロック発振停止中？	メインクロック発振安定待ち時間
No	なし
Yes	2^{15}×メインクロック周期

本リセットのリセット要因が解除された後の各リセット発行シーケンスを以下に示します。

図 7-7 リセット (RST) シーケンス

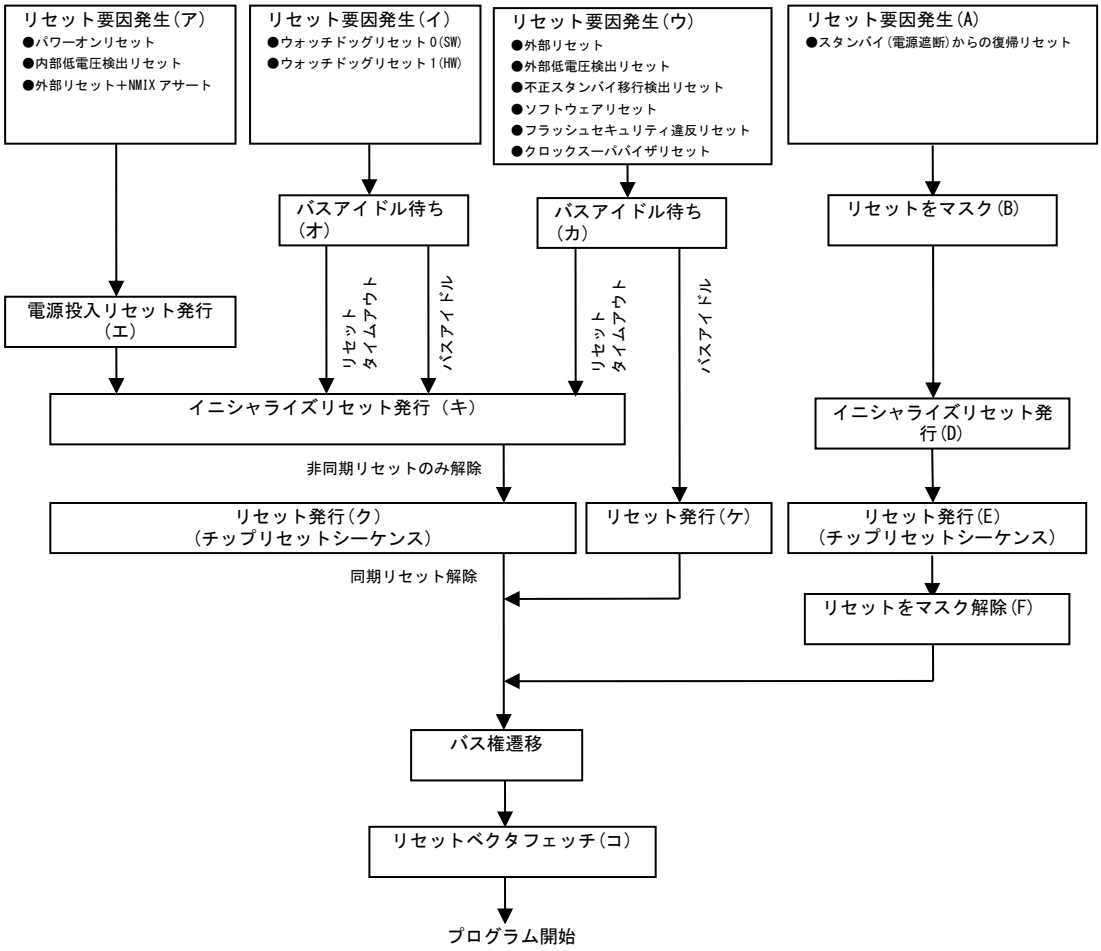


7.5.5 リセットシーケンス

リセットシーケンスについて示します。

リセット要因の消失により、本シリーズは初期状態からプログラムおよびハードウェア動作を開始します。このリセットから動作開始にいたる一連の動作をリセットシーケンスとよびます。以下、リセットシーケンスについて説明します。

図 7-8 リセットシーケンス



(注意事項)

・(キ)以降、または(オ)(カ)中に、(ア)が発生した場合、(ア)から始め直します。

・(B)以降、(ア)が発生した場合、(ア)から始め直します。

詳細は『クロック・リセット状態遷移』の章の「図 2-1 デバイス状態遷移図」を参照してください。

・(エ)中に、メインクロック発振安定待ち時間を取ります。

・(キ)(ク)(ケ)中に、必要な際(CMONR:MCRDY=0)はメインクロック発振安定待ちを取ります。

・(コ)『FixedVector 機能』の章を参照してください。

・不正スタンバイ移行検出リセットの場合、リセット要因発生後にすでにバスアイドル状態なので、即(ケ)へ遷移します。

・スタンバイ(電源遮断)からの復帰(A)では、電源投入リセットが発行されますが、下記ブロックへのリセットを抑止するため、リセット期間中は本ブロックへの電源投入リセットを除くリセットをマスク(B)します。

(1) RTC(時計モードのみ)
(2) 外部割り込みブロック
(3) 電源マネジメントユニット
(4) クロック生成ブロック(サブクロック選択レジスタのみ)　(5)LCD コントローラ

7.5.5.1 リセットサイクル

リセットサイクルについて示します。

リセット要因の解除後、4×周辺クロック(PCLK)周期の間リセット要求が延長され、その後リセットレベルごとに周辺クロック(PCLK)×16サイクルの期間ずつリセットサイクルを維持します。これにより、各リセットの最小発行サイクル数は20サイクルとなります。メインクロック発振安定待ちが必要な際は、当該分だけ延長されます。

7.5.5.2 リセット解除

リセット解除について示します。

リセットサイクルが終了すると、各リセットが解除され、各ハードウェアが動作を開始します。リセット解除直後はモード制御回路がオンチップバスのバスマスタとなります。

7.5.5.3 動作モード確定

動作モード確定について示します。

バスマスタとなったモード制御回路は、取得したモード設定値に基づき決定した動作モードを各ハードウェアに通知し、その後にオンチップバスのバス権を解放します。

7.5.5.4 バス権の遷移

バス権の遷移について示します。

モード制御回路がオンチップバスのバス権を解放した後は、CPUがバス権を獲得し、CPUによるバス動作を開始します。

7.5.5.5 リセットベクタフェッチ

リセットベクタフェッチについて示します。

リセット解除後、CPUはリセットベクタ(0x000FFFFC番地)のフェッチを開始します。

CPUがバス権を獲得後、リセットベクタへのアクセスがオンチップバスを介して行われ、取得したリセットベクタをPCに取り込み、プログラム動作を開始します。

7.5.5.6 リセットと強制ブレーク

リセットと強制ブレークについて示します。

リセット解除時に強制ブレークが発生していた場合、リセットベクタフェッチ終了後に強制ブレークを受付けます。このため、取得したリセットベクタによるPC値がエミュレータ空間側へ退避されます(E_BPCHR,E_BPCLRレジスタにストアされます)。

7.5.6 注意事項

注意事項について示します。

スタンバイ時計モード(電源遮断)およびスタンバイストップモード (電源遮断) からの復帰中は、内部リセットが発行されており、パワーオンリセット、内部電源低電圧検出リセット、RSTX・NMIX同時アサートによるリセット以外のリセット要因を受付けません。

8. DMA コントローラ (DMAC)

DMAコントローラ (DMAC) について説明します。

8.1 概要

DMAコントローラ (DMAC) の概要について説明します。

DMACは、DMA (Direct Memory Access) 転送を行うモジュールです。本モジュール制御によるDMA転送により、CPUを介さずに各種データ転送を高速に行うことが可能となり、システムのパフォーマンスを増加させます。

8.2 特長

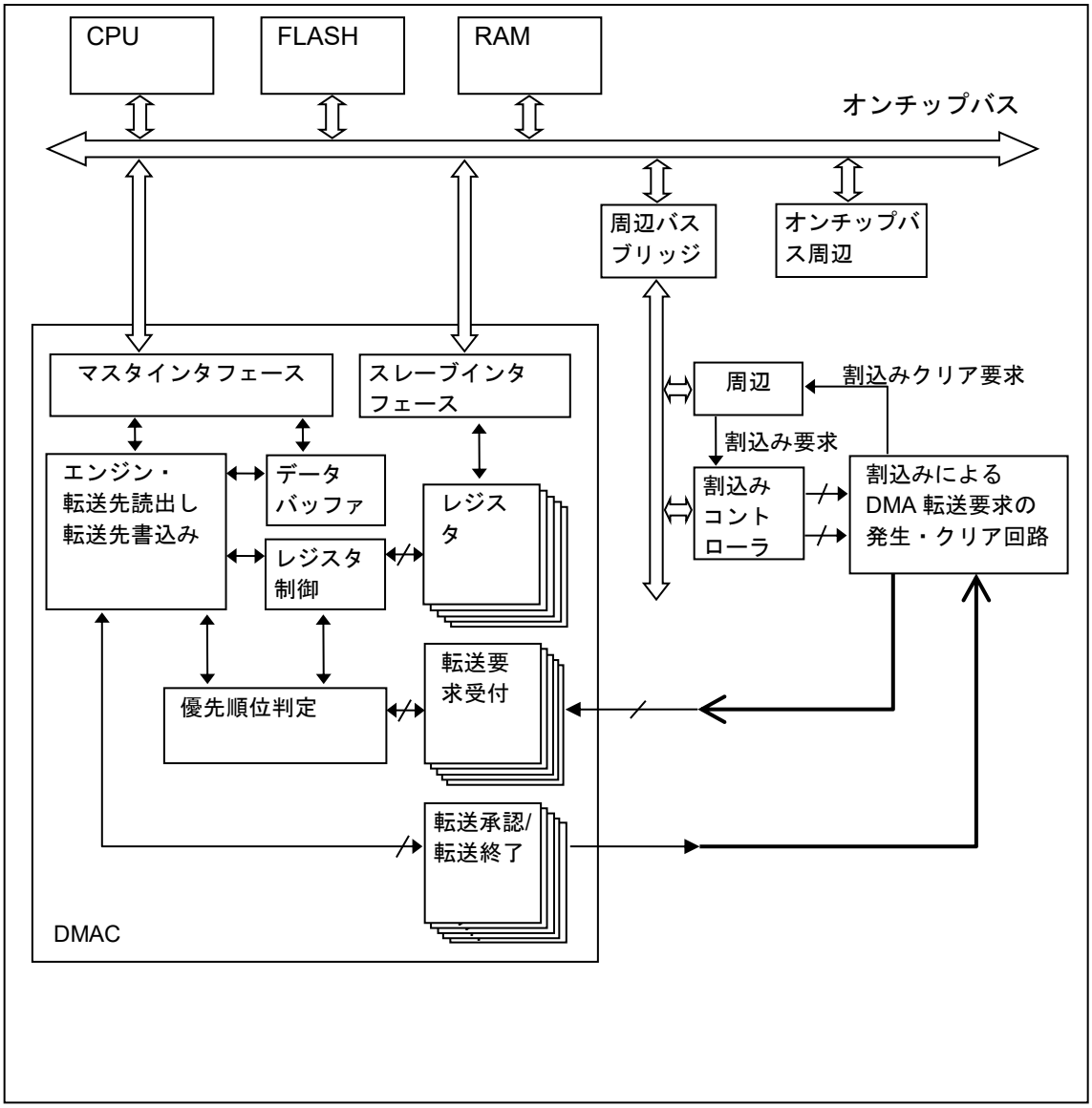
DMAコントローラ(DMAC)の特長について説明します。

- チャネル数 : 16 チャネル
- アドレス空間 : 32 ビットのアドレス空間 (4GB)
- 転送モード : ブロック転送/バースト転送
- アドレス更新 : 増加/減少/固定 (増減値は 1, 2, 4 固定)
- 転送サイズ : 8 ビット, 16 ビット, 32 ビット
- ブロックサイズ: 1 ～ 16
- 転送回数 : 1 ～ 65535 回
- 転送要求 :
  - ソフトウェア転送要求
  - ペリフェラルの割込みによる転送要求(ペリフェラルの割込みによる転送要求の場合、チャネルごとに割込みの選択が必要になります。『DMA 転送要求の発生・クリア』の章を参照してください。)
- 転送停止要求 : 割込みによる転送停止要求
- リロード機能 : 全チャネルリロード指定可
  - 転送元アドレスリロード
  - 転送先アドレスリロード
  - 転送回数リロード
- 優先順位 :
  - 固定 (ch.0 > ch.1 > ch.2 > ch.3 > ch.4 > ch.5 > ch.6 > ch.7 > ch.8 > ch.9 > ch.10 > ch.11 > ch.12 > ch.13 > ch.14 > ch.15)
  - もしくはラウンドロビン
- 割込み要求 : 正常終了割込み要求, 異常終了割込み要求, 転送停止要求による転送中断割込み要求を発生可能

8.3 構成

DMAコントローラ(DMAC)のブロック構成について示します。

図 8-1 ブロックダイヤグラム



8.4 レジスタ

DMAコントローラ(DMAC)のレジスタについて説明します。

表 8-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0C00	DCCR0				DMA チャネルコントロールレジスタ 0
0x0C04	DCSR0		DTCR0		DMA チャネルステータスレジスタ 0 DMA 転送回数レジスタ 0
0x0C08	DSAR0				DMA 転送元アドレスレジスタ 0
0x0C0C	DDAR0				DMA 転送先アドレスレジスタ 0
0x0C10	DCCR1				DMA チャネルコントロールレジスタ 1
0x0C14	DCSR1		DTCR1		DMA チャネルステータスレジスタ 1 DMA 転送回数レジスタ 1
0x0C18	DSAR1				DMA 転送元アドレスレジスタ 1
0x0C1C	DDAR1				DMA 転送先アドレスレジスタ 1
0x0C20	DCCR2				DMA チャネルコントロールレジスタ 2
0x0C24	DCSR2		DTCR2		DMA チャネルステータスレジスタ 2 DMA 転送回数レジスタ 2
0x0C28	DSAR2				DMA 転送元アドレスレジスタ 2
0x0C2C	DDAR2				DMA 転送先アドレスレジスタ 2
0x0C30	DCCR3				DMA チャネルコントロールレジスタ 3
0x0C34	DCSR3		DTCR3		DMA チャネルステータスレジスタ 3 DMA 転送回数レジスタ 3
0x0C38	DSAR3				DMA 転送元アドレスレジスタ 3
0x0C3C	DDAR3				DMA 転送先アドレスレジスタ 3
0x0C40	DCCR4				DMA チャネルコントロールレジスタ 4
0x0C44	DCSR4		DTCR4		DMA チャネルステータスレジスタ 4 DMA 転送回数レジスタ 4
0x0C48	DSAR4				DMA 転送元アドレスレジスタ 4
0x0C4C	DDAR4				DMA 転送先アドレスレジスタ 4
0x0C50	DCCR5				DMA チャネルコントロールレジスタ 5
0x0C54	DCSR5		DTCR5		DMA チャネルステータスレジスタ 5 DMA 転送回数レジスタ 5
0x0C58	DSAR5				DMA 転送元アドレスレジスタ 5
0x0C5C	DDAR5				DMA 転送先アドレスレジスタ 5
0x0C60	DCCR6				DMA チャネルコントロールレジスタ 6
0x0C64	DCSR6		DTCR6		DMA チャネルステータスレジスタ 6 DMA 転送回数レジスタ 6
0x0C68	DSAR6				DMA 転送元アドレスレジスタ 6
0x0C6C	DDAR6				DMA 転送先アドレスレジスタ 6
0x0C70	DCCR7				DMA チャネルコントロールレジスタ 7
0x0C74	DCSR7		DTCR7		DMA チャネルステータスレジスタ 7 DMA 転送回数レジスタ 7
0x0C78	DSAR7				DMA 転送元アドレスレジスタ 7
0x0C7C	DDAR7				DMA 転送先アドレスレジスタ 7
0x0C80	DCCR8				DMA チャネルコントロールレジスタ 8
0x0C84	DCSR8		DTCR8		DMA チャネルステータスレジスタ 8 DMA 転送回数レジスタ 8
0x0C88	DSAR8				DMA 転送元アドレスレジスタ 8

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0C8C	DDAR8				DMA 転送先アドレスレジスタ 8
0x0C90	DCCR9				DMA チャネルコントロールレジスタ 9
0x0C94	DCSR9		DTCR9		DMA チャネルステータスレジスタ 9 DMA 転送回数レジスタ 9
0x0C98	DSAR9				DMA 転送元アドレスレジスタ 9
0x0C9C	DDAR9				DMA 転送先アドレスレジスタ 9
0x0CA0	DCCR10				DMA チャネルコントロールレジスタ 10
0x0CA4	DCSR10		DTCR10		DMA チャネルステータスレジスタ 10 DMA 転送回数レジスタ 10
0x0CA8	DSAR10				DMA 転送元アドレスレジスタ 10
0x0CAC	DDAR10				DMA 転送先アドレスレジスタ 10
0x0CB0	DCCR11				DMA チャネルコントロールレジスタ 11
0x0CB4	DCSR11		DTCR11		DMA チャネルステータスレジスタ 11 DMA 転送回数レジスタ 11
0x0CB8	DSAR11				DMA 転送元アドレスレジスタ 11
0x0CBC	DDAR11				DMA 転送先アドレスレジスタ 11
0x0CC0	DCCR12				DMA チャネルコントロールレジスタ 12
0x0CC4	DCSR12		DTCR12		DMA チャネルステータスレジスタ 12 DMA 転送回数レジスタ 12
0x0CC8	DSAR12				DMA 転送元アドレスレジスタ 12
0x0CCC	DDAR12				DMA 転送先アドレスレジスタ 12
0x0CD0	DCCR13				DMA チャネルコントロールレジスタ 13
0x0CD4	DCSR13		DTCR13		DMA チャネルステータスレジスタ 13 DMA 転送回数レジスタ 13
0x0CD8	DSAR13				DMA 転送元アドレスレジスタ 13
0x0CDC	DDAR13				DMA 転送先アドレスレジスタ 13
0x0CE0	DCCR14				DMA チャネルコントロールレジスタ 14
0x0CE4	DCSR14		DTCR14		DMA チャネルステータスレジスタ 14 DMA 転送回数レジスタ 14
0x0CE8	DSAR14				DMA 転送元アドレスレジスタ 14
0x0CEC	DDAR14				DMA 転送先アドレスレジスタ 14
0x0CF0	DCCR15				DMA チャネルコントロールレジスタ 15
0x0CF4	DCSR15		DTCR15		DMA チャネルステータスレジスタ 15 DMA 転送回数レジスタ 15
0x0CF8	DSAR15				DMA 転送元アドレスレジスタ 15
0x0CFC	DDAR15				DMA 転送先アドレスレジスタ 15
0x0DF4	予約	予約	DNMIR	DILVR	DMA 転送抑止 NMI フラグレジスタ DMA 転送抑止割込みレベルレジスタ
0x0DF8	DMACR				DMA コントロールレジスタ
0x0DFC	予約				予約

8.4.1 DMA コントロールレジスタ ：DMACR (DMA Control Register)

DMAコントロールレジスタ (DMACR) のビット構成について説明します。

DMAC全体 (全チャネル) を制御するための32ビットレジスタです。本レジスタには32ビットでアクセスしてください。

■ DMACR： アドレス 0DF8H (アクセス: ワード)

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	DME	予約						
初期値	0	0	0	0	0	0	0	0
属性	R/W	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	AT	予約						
初期値	0	0	0	0	0	0	0	0
属性	R/W	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

[bit31] DME (DMA Enable) : DMA動作許可

DMAC全体の動作を制御します。本ビットが"0"の場合は各チャネルが動作許可状態であってもDMA転送は行いません。本ビットが"1"の場合に各チャネルの設定にしたがった動作を行います。

DMA転送中に"0"が書き込まれた場合はDCCRn:BLKで指定したブロック単位で転送を停止します。

DME	DMA 動作許可
0	DMA 動作禁止(初期値)
1	DMA 動作許可

[bit30～bit16] 予約

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit15] AT (Arbitration Type) 優先順位設定

各チャネルの優先順位を決める方法を設定します。優先順位を固定 (AT = 0) に設定した場合は、ch.0 > ch.1 > ch.2 > ch.3となります。優先順位をラウンドロビン(AT =1)に設定した場合は、転送を開始したチャネルの優先順位が一番低くなり、それより下位にあったチャネルの優先順位が一つずつ上がります。優先順位の判定は、優先順位設定にかかわらずDCCRn:BLKで指定したブロック単位の転送ごとに行います。

AT	優先順位設定
0	固定 (初期値)
1	ラウンドロビン

[bit14～bit0] 予約

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

8.4.2 DMAチャネルコントロールレジスタ 0-15 : DCCR0-15 (DMA Channel Control Register 0-15)

DMAチャネルコントロールレジスタ0-15 (DCCR0-15)のビット構成について説明します。

DMAC各チャネルの動作制御を行う32ビットレジスタで、各チャネル独立して存在します。本レジスタには32ビットでアクセスしてください。

■ DCCR0-15 : アドレス BASE + 0000_H (アクセス: ワード)

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	CE	予約				AIE	SIE	NIE
初期値	0	0	0	0	0	0	0	0
属性	R,W	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	予約		RS[1:0]		予約		TM[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R/W	R/W	R0,W0	R0,W0	R/W	R/W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	ST	SAR	SAC[1:0]		DT	DAR	DAC[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TCR	予約	TS[1:0]		BLK[3:0]			
初期値	0	0	0	0	0	0	0	0
属性	R/W	R0,W0	R/W	R/W	R/W	R/W	R/W	R/W

[bit31] CE (Channel Enable) : チャネル動作許可

チャネルの動作を制御します。要求元をソフトウェアに設定している場合はこのビットに"1"を書き込むと設定した内容にしたがいDMA転送が開始します。この場合、転送要求に応じた転送を終了するとCEビットは自動でクリアされます。要求元がソフトウェア以外の場合は"1"を書き込むとチャネル動作許可になります。チャネル動作許可後､対応する転送要求を検出するとDMA転送が開始します｡ソフトウェア以外の要求で転送回数リロード (DCCRn:TCR) が指定されている場合、CEビットは自動でクリアされません。転送回数リロードを禁止している場合は、すべての転送が終了するとCEビットがクリアされます。要求元によらず動作中に"0"が書き込まれた場合はDCCRn:BLKで指定したブロック単位で転送を停止します。再び"1"を書き込み、新たに転送要求を検出すると動作を再開します。

CE	チャネル動作許可
0	禁止 (初期値)
1	許可

[bit30～bit27] 予約

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit26] AIE (Abnormal completion Interrupt Enable) : 異常終了割込み許可

DMAチャネルコントロールレジスタ (DCCR) に設定禁止の値を設定した場合の割込み発生を制御します。レジスタ設定禁止項目は以下のとおりです。

- 転送モード : DCCR n :TM = 10_B
- 転送元アドレスカウント : DCCR n :SAC = 10_B
- 転送先アドレスカウント : DCCR n :DAC = 10_B
- 転送サイズ : DCCR n :TS = 11_B
- ソフトウェア要求によるデマンド転送モード : DCCR n :RS = 00_B かつ DCCR n :TM = 11_B

割込み要因はステータスレジスタ (DCSR n) を確認してください。

AIE	異常終了割込み許可
0	禁止 (初期値)
1	許可

[bit25] SIE (Stop Interrupt Enable) : 転送停止要求による転送中断割込み許可

転送要求元からの転送停止要求によりDMA転送を中断した場合の割込み発生を制御します｡割込み要因はステータスレジスタ(DCSR n)を確認してください。

SIE	転送中断割込み許可
0	禁止 (初期値)
1	許可

[bit24] NIE (Normal completion Interrupt Enable) : **正常終了割込み許可**

DMA転送を正常に終了した場合の割込み発生を制御します。設定した転送回数 (DTCRn:DTC) の転送が終了したときまたは、転送回数が"0"のときに対応するチャネルのDCCRn:CEビットに"1"書込みしたときに正常終了となります。割込み要因はステータスレジスタ(DCSRn)を確認してください。

NIE	正常終了割込み許可
0	禁止 (初期値)
1	許可

[bit23,bit22] **予約**

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit21, bit20] RS[1:0] (Request Source) : **DMA転送要求元**

チャネルに対する転送要求元を選択します。

RS[1:0]	DMA 転送要求元
00	ソフトウェア (初期値)
01	割込み
10	予約(設定禁止)
11	予約(設定禁止)

[bit19, bit18] **予約**

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit17, bit16] TM[1:0] (Transfer Mode) : **転送モード**

DMA転送モードを指定します。

TM[1:0]	転送モード
00	ブロック転送 (初期値)
01	バースト転送
10	予約(設定禁止)
11	予約(設定禁止)

[bit15] ST (Source Type) : **転送元タイプ**

DMA転送要求元(DCCR:RS[1:0]), 転送元アドレス (DSAR), 転送先アドレス (DDAR) の組み合わせにより設定値が異なります。設定は「■ STビット (転送元タイプ), DTビット (転送先タイプ) の設定」を参照してください。

ST	転送元タイプ
0	「ST ビット (転送元タイプ), DT ビット (転送先タイプ) の設定」を参照してください
1	

[bit14] SAR (Source Address Reload) : **転送元アドレスリロード**

転送元アドレスレジスタのリロードを指定します。リロードを指定している場合、転送終了後に転送元アドレスレジスタ値を初期設定値に戻します。リロードを禁止している場合、転送終了後の転送元アドレスレジスタは最終アドレスの次のアクセスアドレスを示します。

SAR	転送元アドレスリロード指定
0	リロード禁止 (初期値)
1	リロード

[bit13, bit12] SAC[1:0] (Source Address Count) : **転送元アドレスカウント**

転送元アドレスの1転送ごとのアドレス更新を指定します。増加/減少を指定した場合の更新値は、転送サイズ(DCCRn:TS)により1, 2, 4のいずれかになります。

SAC[1:0]	転送元アドレスカウント
00	アドレス増加 (初期値)
01	アドレス減少
10	予約 (設定禁止)
11	アドレス固定

[bit11] DT (Destination Type) : **転送先タイプ**

DMA転送要求元(DCCR:RS[1:0]), 転送元アドレス(DSAR), 転送先アドレス (DDAR) の組み合わせにより設定値が異なります。設定は「■ STビット (転送元タイプ), DTビット (転送先タイプ) の設定」を参照してください。

DT	転送先タイプ
0	「■ ST ビット (転送元タイプ), DT ビット (転送先タイプ) の設定」を参照してください
1	

[bit10] DAR (Destination Address Reload) : **転送先アドレスリロード**

転送先アドレスレジスタのリロードを指定します。リロードを指定している場合、転送終了後に転送先アドレスレジスタ値を初期設定値に戻します。リロードを禁止している場合、転送終了後の転送先アドレスレジスタは最終アドレスの次のアクセスアドレスを示します。

DAR	転送先アドレスリロード指定
0	リロード禁止 (初期値)
1	リロード

[bit9, bit8] DAC[1:0] (Destination Address Count) : **転送先アドレスカウント**

転送先アドレスの一転送ごとのアドレス更新を指定します。増加/減少を指定した場合の更新値は、転送サイズ(DCCRn:TS)により1, 2, 4のいずれかになります。

DAC[1:0]	転送先アドレスカウント
00	アドレス増加 (初期値)
01	アドレス減少
10	予約 (設定禁止)
11	アドレス固定

[bit7] TCR (Transfer Count Reload) : **転送回数リロード**

転送回数レジスタのリロードを指定します。

リロードを指定している場合、転送終了後に転送回数レジスタ値を初期設定値に戻します。ここで転送要求元をソフトウェア以外に設定している場合は、転送終了時にDCCRn:CEビットはクリアされず転送要求待ち状態となります。リロードを禁止している場合、転送終了後の転送回数レジスタは"0"を示します。この場合は、転送要求元によらず転送終了時にDCCRn:CEビットはクリアされます。

TCR	転送回数リロード
0	リロード禁止 (初期値)
1	リロード

[bit6] **予約**

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit5, bit4] TS[1:0] (Transfer Size) : **転送サイズ**

転送サイズを指定します。ここで指定されたビット幅で1回のDMA転送を行います。

TS[1:0]	転送サイズ
00	8 ビット : バイト (初期値)
01	16 ビット : ハーフワード
10	32 ビット : ワード
11	予約 (設定禁止)

DSARn, DDARnレジスタは本ビットで指定する転送サイズに対してミスアラインドとならない値を設定してください。

[bit3～bit0] BLK[3:0] (BLock Size) : **ブロックサイズ**

ブロックサイズを指定します。1ブロックは、DCCRn:TSビットで指定した転送サイズでブロック数分転送を繰り返します。

BLK[3:0]	ブロックサイズ
0000	1byte (初期値)
0001	2byte
0010	3byte
0011	4byte
0100	5byte
0101	6byte
0110	7byte
0111	8byte
1000	9byte
1001	10byte
1010	11byte
1011	12byte
1100	13byte
1101	14byte
1110	15byte
1111	16byte

8.4.3 DMA チャネルステータスレジスタ 0-15 : DCSR0-15 (DMA Channel Status Register 0-15)

DMAチャネルステータスレジスタ0-15 (DCSR0-15) のビット構成について説明します。

DMAC 各チャネルの状態を示す16ビットレジスタで、各チャネル独立して存在します。本レジスタには16ビットでアクセスしてください。

■ DCSR0-15 : アドレス BASE + 0004$_H$ (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	CA	予約						
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					AC	SP	NC
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R,W	R,W	R,W

[bit15] CA (Channel Active) : チャネルアクティブ

チャネルの動作状態を示します。対応するチャネルのDCCRn:CEビットに"1"を書き込むと起動状態となります。設定した転送回数が終了するかDCCRn:CEビットに"0"を書き込むと停止状態となります。

本ビットへの書込みは無効です。

CA	チャネル動作状態
0	停止状態 (初期値)
1	チャネル起動中

[bit14～bit3] 予約

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit2] AC (Abnormal Completion): 異常終了状態

DMAチャネルコントロールレジスタ(DCCR)に設定禁止の値を設定したことを示します。レジスタ設定禁止項目は以下のとおりです。

- 転送モード : DCCRn:TM = 10_B
- 転送元アドレスカウント : DCCRn:SAC = 10_B
- 転送先アドレスカウント : DCCRn:DAC = 10_B
- 転送サイズ : DCCRn:TS = 11_B
- ソフトウェア要求によるデマンド転送モード : DCCRn:RS = 00_B かつ DCCRn:TM = 11_B

異常終了割込み (DCCRn:AIE) を許可していた場合、本ビットへの"0"書込みにより割込みをクリアします。本ビットへの"1"書込みは無効です。

本ビットは自動でクリアしませんので、DMA動作を許可する前に必ずクリアしてください。

AC	異常終了状態
0	異常終了未検出 (初期値)
1	異常終了

[bit1] SP (Stop) : 転送停止要求による転送中断状態

DMA転送が転送要求元からの転送停止要求により中断したことを示します。転送中断割込み (DCCRn:SIE) を許可していた場合、本ビットへの"0"書込みにより割込みをクリアします。本ビットへの"1"書込みは無効です。

本ビットは自動でクリアしませんので、DMA動作を許可する前に必ずクリアしてください。

SP	転送中断状態
0	転送中断未検出 (初期値)
1	転送中断

[bit0] NC (Normal Completion) : 正常終了状態

DMA転送が正常終了したことを示します。設定した転送回数の転送を終了したときまたは、転送回数が"0"のときに対応するチャネルのDCCRn:CEビットに"1"書込みしたときに正常終了となります。正常終了割込み (DCCRn:NIE) を許可していた場合、本ビットへの"0"書込みにより割込みをクリアします。本ビットへの"1"書込みは無効です。

本ビットは自動でクリアしませんので、DMA動作を許可する前に必ずクリアしてください。

NC	正常終了状態
0	正常終了未検出 (初期値)
1	正常終了

8.4.4 DMA 転送回数レジスタ 0-15 : DTCR0-15 : (DMA Transfer Count Register 0-15)

DMA転送回数レジスタ0-15 (DTCR0-15) のビット構成について説明します。

DMAC各チャネルの転送回数を示す16ビットレジスタで、各チャネル独立して存在します。本レジスタには16ビットでアクセスしてください。

■ DTCR0-15 : アドレス BASE + 0006H (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	DTC[15:8]							
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DTC[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

[bit15～bit0] DTC (DMA Transfer Count) : DMA転送回数

転送回数を示すレジスタです。ブロック単位の転送終了ごとにデクリメントし、"0"になると転送終了します。転送回数に"0"を設定した場合は転送を行いません。また、専用のリロードレジスタを持ち、DCCRn:TCRが"1"の場合、転送終了後に初期設定値に戻ります。

8.4.5 DMA 転送元レジスタ 0-15 : DSAR0-15 : (DMA Source Address Register 0-15)

DMA転送元レジスタ0-15 (DSAR0-15)のビット構成について説明します。

DMAC各チャネルの転送元アドレスを示す32ビットレジスタで、各チャネル独立して存在します。本レジスタには32ビットでアクセスしてください。

■ DSAR0-15 : アドレス BASE + 0008_H (アクセス: ワード)

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	DSA[31:24]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	DSA[23:16]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	DSA[15:8]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DSA[7:0]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

[bit31～bit0] DSA[31:0] (DMA Source Address) : DMA転送元アドレス

転送元アドレスを示すレジスタです。DCCRn:SACで増加/減少が指定されている場合は、転送サイズ (DCCRn:TS) にしたがってアドレスが更新されます。また、専用のリロードレジスタを持ち、DCCRn:SARが"1"の場合、転送終了後に初期設定値に戻ります。

本レジスタへは、DCCRn:TSで指定する転送サイズに対してミスアラインドにならない値を設定してください。

DMA転送要求元がペリフェラル割込みの場合(DCCRn:RS[1:0]=01)、転送元アドレス(DSAR)または転送先アドレス(DDAR)かのどちらか最低一方は16ビット周辺バスまたは32ビット周辺バス配下のペリフェラルのアドレス範囲である必要があります。詳細は「■ STビット (転送元タイプ), DTビット (転送先タイプ) の設定」を参照してください。

8.4.6 DMA 転送先レジスタ 0-15 : DDAR0-15 (DMA Destination Address Register 0-15)

DMA転送先レジスタ0-15 (DDAR0-15)のビット構成について説明します。

DMAC各チャネルの転送先アドレスを示す32ビットレジスタで、各チャネル独立して存在します。本レジスタには32ビットでアクセスしてください。

■ DDAR0-15 : アドレス BASE + 000C$_H$ (アクセス: ワード)

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	DDA[31:24]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	DDA[23:16]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	DDA[15:8]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DDA[7:0]							
初期値	X	X	X	X	X	X	X	X
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

[bit31～bit0] DDA [31:0] (DMA Destination Address) : DMA転送先アドレス

転送先アドレスを示すレジスタです。DCCRn:DACで増加/減少が指定されている場合は、転送サイズ(DCCRn:TS)にしたがってアドレスが更新されます。また、専用のリロードレジスタを持ち、DCCRn:DAR が"1"の場合、転送終了後に初期設定値に戻ります。

本レジスタへは、DCCRn:TSで指定する転送サイズに対してミスアラインドにならない値を設定してください。

DMA転送要求元がペリフェラル割込みの場合(DCCRn:RS[1:0]=01)、転送元アドレス(DSAR)または転送先アドレス(DDAR)かのどちらか最低一方は16ビット周辺バスまたは32ビット周辺バス配下のペリフェラルのアドレス範囲である必要があります。詳細は「■ STビット (転送元タイプ), DTビット (転送先タイプ) の設定」を参照してください。

8.4.7 DMA 転送抑止 NMI フラグレジスタ : DNMIR (DMA-halt by NMI Register)

DMA転送抑止NMIフラグレジスタ (DNMIR)のビット構成について説明します。

ユーザNMIによるDMA転送の抑止を制御するための8ビットレジスタです。本レジスタには8ビットでアクセスしてください。

■ DNMIR : アドレス 0DF6H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	NMIH	予約						NMIHD
初期値	0	0	0	0	0	0	0	0
属性	R,W	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R/W

[bit7] NMIH (NMI Halt) : DMA抑止フラグ (NMI要因)

NMIHDビットが"0"のときにユーザ用NMI要求が発生したことを示すフラグです。NMIの"H"レベルを検出して本ビットに"1"をセットします。DMA転送を再開する場合には、本ビットに"0"を書き込んでください。

本ビットへの"1"書込みは無効です。

NMIH	DMA 抑止フラグ
0	DMA 転送を抑止していません。 (初期値)
1	ユーザ用 NMI により DMA 転送を停止しています。

[bit6～bit1] 予約

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit0] NMIHD (NMI Halt Disable) : DMA抑止制御 (NMI要因)

ユーザ用NMI要求が発生した場合にDMA転送を停止することを指示する制御ビットです。

本ビットが"0"のときにNMIが発生すると、DMACは新たなDMA転送を開始しません。DMA転送中である場合は、ブロック単位の転送が終了した時点で転送を停止します。

NMIHD	DMA 抑止制御
0	ユーザ用 NMI により DMA を停止します。 (初期値)
1	ユーザ用 NMI により DMA を停止しません。

8.4.8 DMA 転送抑止レベルレジスタ : DILVR (DMA-halt by Interrupt Level Register)

DMA転送抑止レベルレジスタ (DILVR) のビット構成について説明します。

周辺割込みによるDMA転送の抑止を制御するための8ビットレジスタです。本レジスタには8ビットでアクセスしてください。

■ DILVR : アドレス 0DF7H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			LVL4	LVL[3:0]			
初期値	0	0	0	1	1	1	1	1
属性	R0,W0	R0,W0	R0,W0	R1,WX	R/W	R/W	R/W	R/W

[bit7～bit5] 予約

このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit4～bit0] LVL[4:0] (Level) : DMA抑止割込みレベル

DMA転送抑止を行う割込みレベルを設定します。本レジスタに設定したレベルより強いレベルの周辺割込みが発生しているとDMA転送を抑止します。LVL4は"1"固定で、LVL[3:0]が設定可能です。

LVL[4:0]	DMA 抑止制御
11111	すべての周辺割込み要求で DMA 転送を抑止します。(初期値)
11110	レベルが $1E_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11101	レベルが $1D_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11100	レベルが $1C_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11011	レベルが $1B_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11010	レベルが $1A_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11001	レベルが 19_H より強い周辺割込み要求で DMA 転送を抑止します。
11000	レベルが 18_H より強い周辺割込み要求で DMA 転送を抑止します。
10111	レベルが 17_H より強い周辺割込み要求で DMA 転送を抑止します。
10110	レベルが 16_H より強い周辺割込み要求で DMA 転送を抑止します。
10101	レベルが 15_H より強い周辺割込み要求で DMA 転送を抑止します。
10100	レベルが 14_H より強い周辺割込み要求で DMA 転送を抑止します。
10011	レベルが 13_H より強い周辺割込み要求で DMA 転送を抑止します。
10010	レベルが 12_H より強い周辺割込み要求で DMA 転送を抑止します。
10001	レベルが 11_H より強い周辺割込み要求で DMA 転送を抑止します。
10000	周辺割込み要求では DMA 転送を抑止しません。

8.5 動作説明

DMAコントローラ(DMAC)の動作について説明します。

■ 設定

全チャネルにおいて共通に設定する項目と、各チャネルにおいて独立に設定する項目について説明します。

● 全チャネルにおいて共通に設定する項目

DMAC全体を制御するレジスタの設定項目を「8.5.1. DMA動作許可」に示します。

8.5.1 DMA 動作許可

DMA動作許可について説明します。

DMACR:DMEにより、DMAC全体の動作制御を設定します。

- DMA 動作禁止 (DMACR:DME=0)
- DMA 動作許可 (DMACR:DME=1)

■ チャネル優先順位

DMACR:ATにより、チャネル間の優先順位の決定方法を設定します。

- 固定 (DMACR:AT=0)
- ラウンドロビン (DMACR:AT=1)

■ 割込み発生時の DMA 転送抑止設定

DNMIR:NMIHDにより、ユーザNMI発生時のDMA転送抑止制御を設定します。

- ユーザ NMI により DMA を停止します。(DNMIR:NMIHD=0)
- ユーザ NMI により DMA を停止しません。(DNMIR:NMIHD=1)

また、DILVR:LVLにより割込み発生時にDMA転送より優先して処理したい割込みレベルを設定します。設定可能なレベルは、0x1Fから0x10までです。

8.5.2 各チャネルにおいて独立に設定する項目

各チャネル独立に設定する項目について説明します。

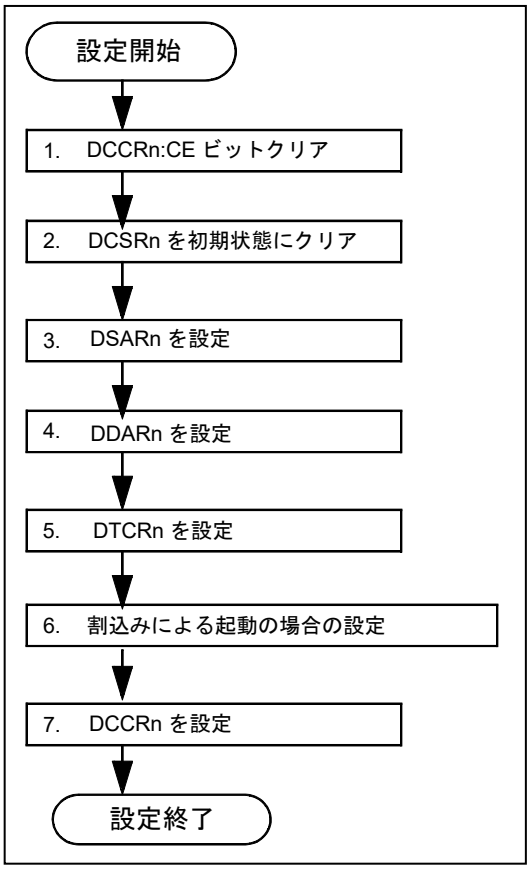
各チャネル独立に設定する項目とレジスタの設定手順を以下に示します。

■ レジスタ設定手順

チャネルレジスタの設定手順は以下のとおりです。DCCRn:CEビットに"1"書き込みする際は、必ずDTCRnに"1"以上を設定してから行ってください。

1. DCCRn:CE ビットをクリアし、チャネルの動作を禁止する。
2. DCSRn の各ビットをクリアし、チャネル状態を示すフラグを初期状態にする。
3. DSARn に転送開始時の転送元アドレス値を設定する。
4. DDARn に転送開始時の転送先アドレス値を設定する。
5. DTCRn に転送回数を設定する。このとき、必ず"1"以上を設定してください。
6. ペリフェラルの割込みによる起動の場合は、各ペリフェラルの割込み発生を許可し、ICSEL, IORR レジスタを設定する。(ICSEL, IORR レジスタについては『DMA 転送要求の発生・クリア』の章を参照してください。)
7. DCCRn を設定する。このとき、DCCRn:CE ビットのセットによりチャネルは動作許可となります。

図 8-2 チャネルレジスタ設定手順



■ 転送元アドレス，転送先アドレスの設定

DSARn:DSAにより、転送開始時の転送元アドレス値を設定します。

DDARn:DDAにより、転送開始時の転送先アドレス値を設定します。

転送元および転送先アドレスは、転送サイズ(DDCRn:TS)にしたがってアドレスアラインを行い、転送サイズが16ビットの場合は下位1ビットを、転送サイズが32ビットの場合は下位2ビットをそれぞれ無視します。

■ 転送回数の設定

DTCRn:DTCにより、転送終了までに転送するブロックの転送回数を設定します。転送回数は、 1～65535回から設定できます。DMACは転送サイズ，およびブロックサイズ(「■ 転送サイズ，ブロックサイズの設定」参照)で設定したバイト数のデータ(1ブロックのデータ)を転送回数で設定した回数分転送します。

■ チャネル動作許可

DCCRn:CEにより、チャネルの動作制御を設定します。

- チャネル動作禁止 (DCCRn:CE=0)
- チャネル動作許可 (DCCRn:CE=1)

転送要求元にソフトウェアが選択された場合、DCCRn:CEをセットするとチャネル動作許可となるとともに転送を開始します。

■ 割込み許可設定

DCCRn:AIEにより、異常終了した場合の割込み許可を設定します。

- 異常終了割込み禁止 (DCCRn:AIE=0)
- 異常終了割込み許可 (DCCRn:AIE=1)

DCCRn:SIEにより、転送停止要求によって転送中断した場合の割込み許可を設定します。

- 転送停止要求検出による転送中断割込み禁止 (DCCRn:SIE=0)
- 転送停止要求検出による転送中断割込み許可 (DCCRn:SIE=1)

DCCRn:NIEにより、正常終了した場合の割込み許可を設定します。

- 正常終了割込み禁止 (DCCRn:NIE=0)
- 正常終了割込み許可 (DCCRn:NIE=1)

■ 転送要求元の設定

DCCRn:RSにより、転送要求を受け付ける転送要求元を設定します。

- ソフトウェアによる要求 (DCCRn:RS=00)
- 割込みによる要求 (DCCRn:RS=01)

■ 転送モードの設定

DCCRn:TMにより、DMA転送の転送モードを設定します。

- ブロック転送 (DCCRn:TM=00)
- バースト転送 (DCCRn:TM=01)

■ ST ビット (転送元タイプ), DT ビット (転送先タイプ) の設定

以下の表にしたがい、設定します。⑤の組合せでのDMA転送は対応していません。

表 8-2 ST ビット (転送元タイプ), DT ビット (転送先タイプ) の設定

	転送要求元・転送元・転送先の組合せ			DMA 転送対応	ST・DT ビット 設定
	転送要求元 (DCCRn:RS[1:0])	転送元 (DSAR)	転送先 (DDAR)		
①	ソフトウェアによる要求 (DCCRn:RS[1:0]=00)	すべての組合せ		対応	ST=0, DT=0
②	ペリフェラル割込み (DCCRn:RS[1:0]=01)	●	□	対応	ST=1, DT=0
③		□	●	対応	ST=0, DT=1
④		●	●	対応	ST=0, DT=1
⑤		□	□	非対応	―

●：16ビット周辺バスまたは32ビット周辺バス配下のペリフェラルのアドレス範囲

□：上記以外のアドレス範囲

ST・DTビットを上表の組合せ以外に設定すると、割込みによるDMA転送要求発生後の自動割込みクリアがされない場合があります。

■ 転送アドレスリロード設定

DCCRn:SARにより、転送終了後の転送元アドレスのリロード制御を設定します。

- 転送終了後、転送元アドレスのリロードはしません。(最終アドレスの次アクセスアドレスを示す。) (DCCRn:SAR=0)
- 転送終了後、転送元アドレスを初期設定値に戻します。 (DCCRn:SAR=1)

DCCRn:DARにより、転送終了後の転送先アドレスのリロード制御を設定します。

- 転送終了後、転送先アドレスのリロードはしません。(最終アドレスの次アクセスアドレスを示す。) (DCCRn:DAR=0)
- 転送終了後、転送先アドレスを初期設定値に戻します。 (DCCRn:DAR=1)

■ 転送アドレス更新設定

DCCRn:SACにより、DMA転送における転送元アドレスのアドレス更新を設定します。

- アドレス増加 (DCCRn:SAC=00)
- アドレス減少 (DCCRn:SAC=01)
- アドレス固定 (DCCRn:SAC=11)

DCCRn:DACにより、DMA転送における転送先アドレスのアドレス更新を設定します。

- アドレス増加 (DCCRn:DAC=00)
- アドレス減少 (DCCRn:DAC=01)
- アドレス固定 (DCCRn:DAC=11)

■ 転送回数リロード設定

DCCRn:TCRにより、転送終了後の転送回数のリロード制御を設定します。

- 転送終了後、転送回数のリロードはしません。(転送正常終了後、転送回数は"0"を示す。) (DCCRn:TCR=0)
- 転送終了後、転送回数を初期設定値に戻します。 (DCCRn:TCR=1)

■ 転送サイズ，ブロックサイズの設定

DMA転送の転送単位(1ブロックとして転送するバイト数)を指定するために、転送サイズ，ブロックサイズを設定します。

DCCRn:TSにより、1回のDMA転送で転送するデータサイズ(8ビット/16ビット/32ビット)を設定します。

- 8 ビット (DCCRn:TS=00)
- 16 ビット (DCCRn:TS=01)
- 32 ビット (DCCRn:TS=10)

DCCRn:BLKにより、1ブロックとして転送するDMA転送の回数を設定します。ブロックサイズは1～16回から設定できます。1ブロックの転送は、転送サイズ(DCCRn:TS)で設定したビット幅のデータを、ブロックサイズで設定した回数転送することになります。

8.5.3 動作

DMACの動作について説明します。

DMACの動作について、以下の項目を説明します。

(1) チャネル状態確認
(2) 転送動作

■ (1) チャネル状態確認

DMACのチャネルごとの状態はDCSRnにより確認することができます。

- チャネルが動作許可のとき(チャネル起動中)は、DCSRn:CA ビットは"1"を示します。チャネルが停止状態になると"0"を示します。
- データ転送が異常終了したときは、DCSRn:AC ビットは"1"を示します。
- データ転送が転送停止要求により中断したときは、DCSRn:SP ビットは"1"を示します。
- データ転送が正常終了したときは、DCSRn:NC ビットは"1"を示します。

DCSRn:CAビットへの書込みは無効です。

DCSRn:AC, DCSRn:SP, DCSRn:NCビットは自動でクリアしませんので、DMA転送を許可する前に必ずクリアしてください。

■ (2) 転送動作

DMACは､転送元アドレスと転送先アドレスを設定することによりDMA転送を行います｡転送元リードにより、DCCRn:TSで設定したビット幅(8ビット/16ビット/32ビット)のデータを転送元アドレスより読込みDMAC内部のデータバッファに一時格納します。引き続き転送先ライトにより、DMAC内部に一時格納したデータを転送先アドレスに書き込みます。

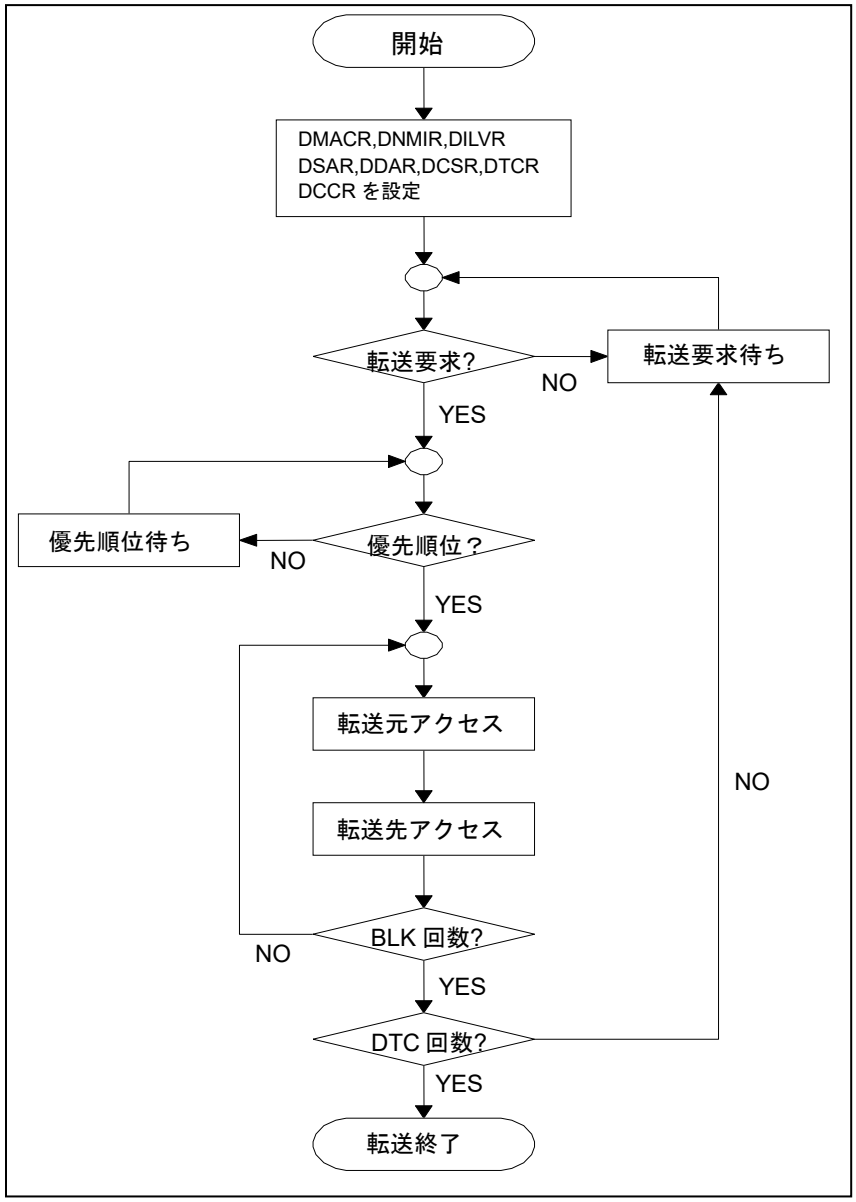
● 転送モード

転送モードには、ブロック転送モード, バースト転送モードがあります。

● ブロック転送モード

1回の転送要求によって、1ブロックの転送を行います。1ブロックの転送終了後、再び転送要求を検出するとさらに1ブロックの転送を行います。これを転送終了まで繰り返し行います。1ブロックの転送は、DCCRn:TSで設定したサイズのデータをブロックサイズで設定した回数分転送することになります。

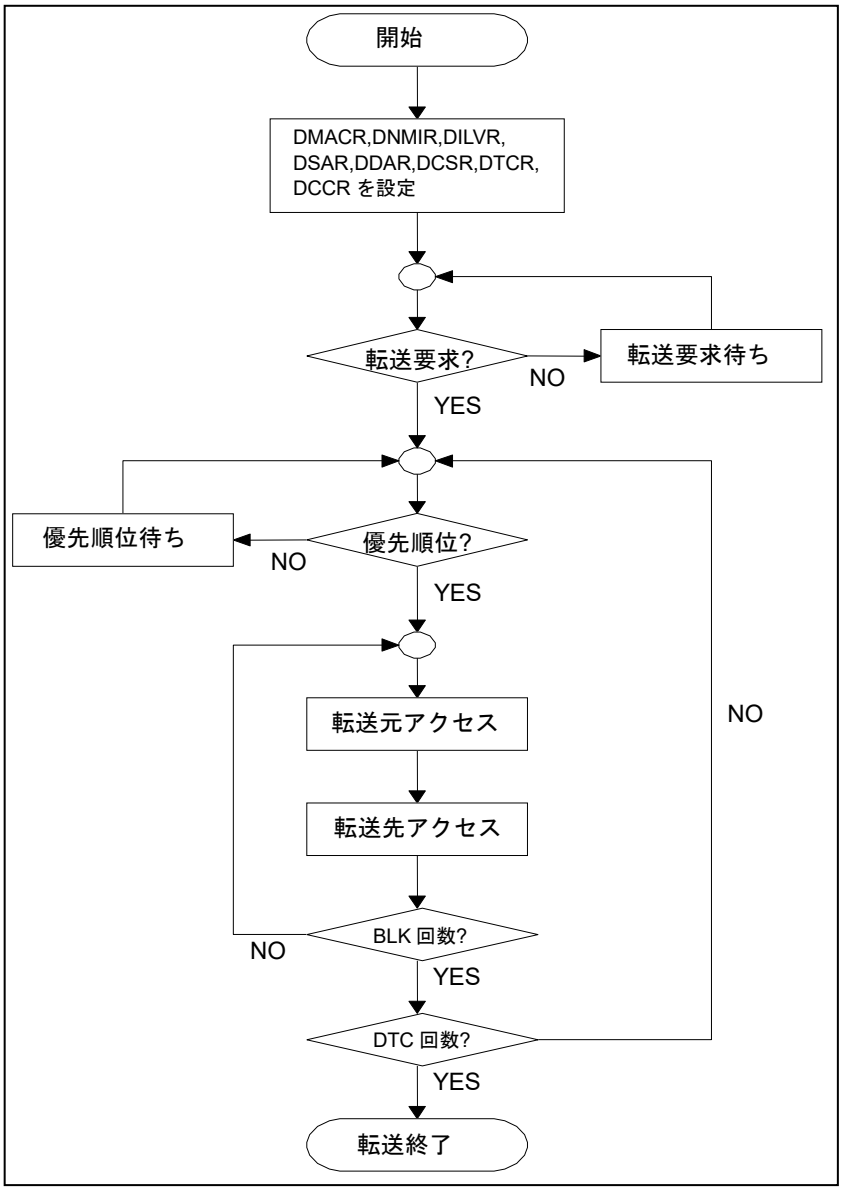
図 8-3 各転送モード(ブロック転送)



● バースト転送モード

1回の転送要求によって、転送終了まで連続して転送を行います。(DCCRn:TSビットで設定したサイズのデータをブロックサイズ×転送回数分連続して転送します。)

図 8-4 各転送モード(バースト転送)



● 転送要求

転送要求には、ソフトウェアによる要求, 割込みによる要求があります。転送要求検出の条件と転送モードとの関係について以下に示します。

● ソフトウェアによる要求

DCCRn:CEビットに"1"を書き込むと転送要求を検出します。DMA動作許可状態(DMACR:DME=1)のとき、優先順位判定後直ちに転送を開始します。転送要求に応じた転送が終了するとDCCRn:CEビットは自動でクリアされます。

● 割込みによる要求

チャネルが動作許可(DCCRn:CE=1)のとき転送要求待ち状態となります。割込みコントローラで設定した周辺の割込みが発生すると転送要求を検出します。DMA動作許可状態(DMACR:DME=1)のとき、優先順位判定後直ちに転送を開始します。

周辺から転送停止要求がアサートされている期間は、転送要求の検出は行いません。

また、チャネルごとに転送要求に使用する割込みベクタの設定が必要になります。

『CHAPTER 9: DMA転送要求の発生・クリア』の章を参照してください。

周辺からの割込み要求はエッジ検出のため、割込み要求発生中にCE=0→1にしても転送要求を検出しません。CE=1後に周辺機能の割込み許可などを行ってください。

表 8-3 転送要求検出の条件と転送モードの関係

	ブロック転送	バースト転送	備考
ソフトウェアによる要求	DCCRn:CE に"1"書込み	DCCRn:CE に"1"書込み	−
割込みによる要求	エッジ検出	エッジ検出	−

また、検出した転送要求とDMACR:DMEビット, DCCRn:CEビットとの関係について表 8-4に示します。転送中にDMEビットまたはCEビットをクリアすると、ブロック単位で転送を停止します。

表 8-4 転送要求と DME ビット, CE ビットの関係

		DME ビット	CE ビット
DME/CE クリア		既に検出した転送要求はクリアされない。	既に検出した転送要求はクリアされる
転送中断後, DME/CE セット	ブロック転送	新たな転送要求を検出すると、優先順位にしたがって転送再開。	新たな転送要求を検出すると、優先順位にしたがって転送再開。
	バースト転送	DME ビットをセットすると優先順位にしたがって直ちに転送再開。	

● DMA 転送要求によるスタンバイ復帰要求

DMACは、MCUがスタンバイモード中に転送要求されると、MCUへスタンバイモードからの復帰を要求します。転送動作が許可された状態で転送要求元からの転送要求がアサートされている期間、スタンバイ復帰を要求します。

● チャネル優先順位

DMACは複数の転送要求が発生した場合、チャネル優先順位にしたがって優先度の高いチャネルからデータ転送を行います。チャネル優先順位には、固定とラウンドロビンがあります。優先順位判定は、ブロック単位の転送ごと、もしくは転送終了のタイミングで行います。

● 固定 (DMACR:AT = 0)

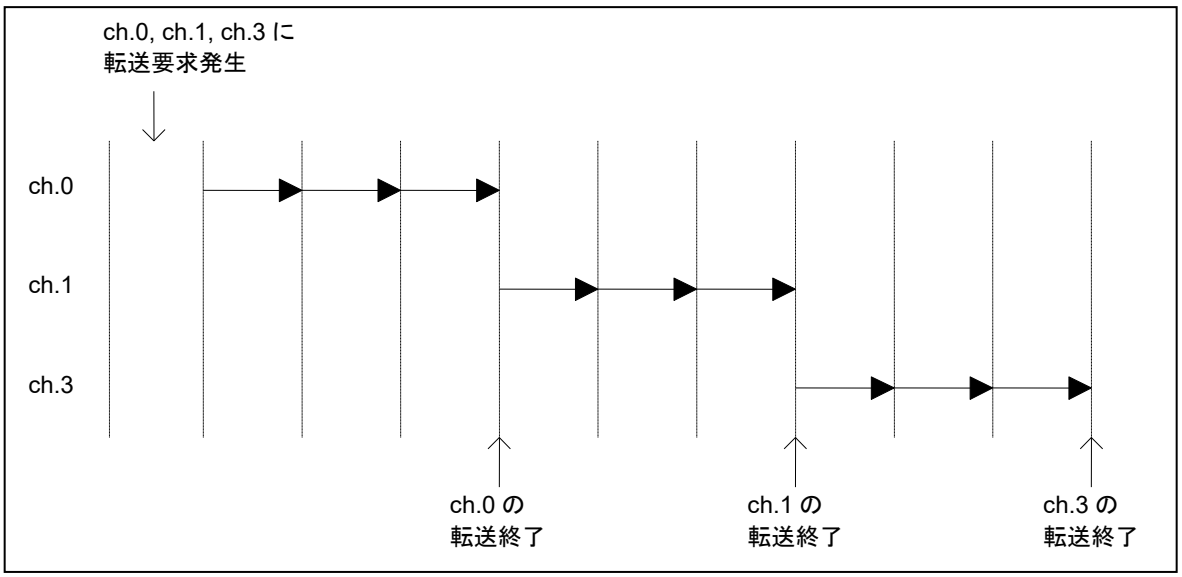
ch.0 > ch.1 > ch.2 > ch.3の順でチャネルの優先順位を固定します。具体例を以下に示します。

＜例1＞ ch.0, ch.1, ch.3に転送要求が同時に発生した場合、ch.0から転送が開始します。ch.0の転送が終わると、続いてch.1の転送が開始します。ch.1の転送が終わると、ch.3の転送が開始します。以下に、転送例を示します。図中の点線はブロック単位の区切りです。

転送要求 : ch.0, ch.1, ch.3に同時発生

設定 : ch.0, ch.1, ch.3ともにバースト転送モード, 転送回数は3

図 8-5 チャネル優先順位固定における転送例 1

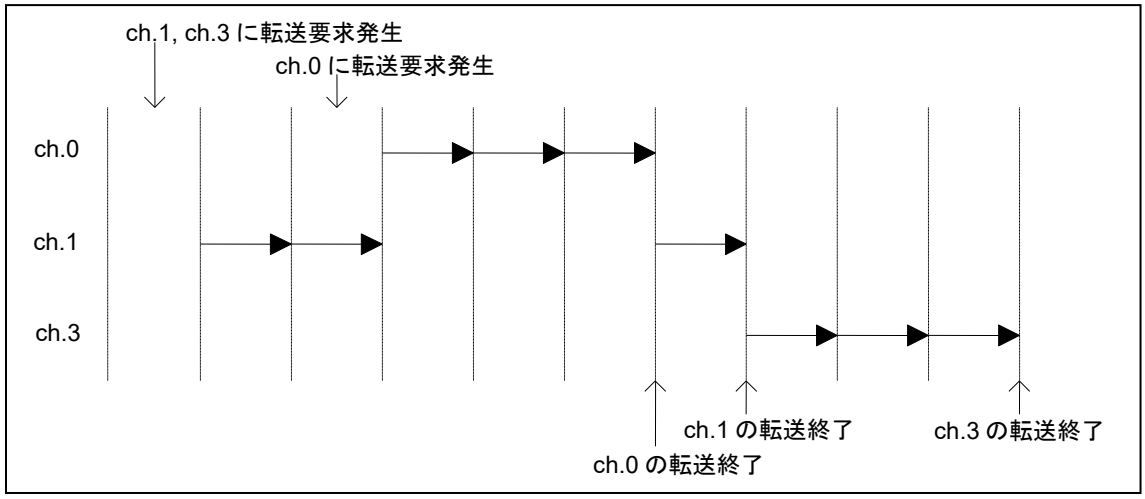


＜例2＞ ch.1, ch.3 に転送要求が同時にあり、ch.1の転送中にch.0に転送要求が発生した場合、ch.1の転送を一時中断し、ch.0の転送を開始します。このとき、チャネルの遷移はブロック単位で行います。ch.0の転送要求に応じた転送が終わると、ch.1の転送を再開します。図中の点線はブロック単位の区切りです。

転送要求 : ch.1, ch.3に同時発生。ch.1転送中にch.0に発生。

設定 : ch.0, ch.1, ch.3ともにバースト転送モード, 転送回数は3

図 8-6 チャネル優先順位固定における転送例 2

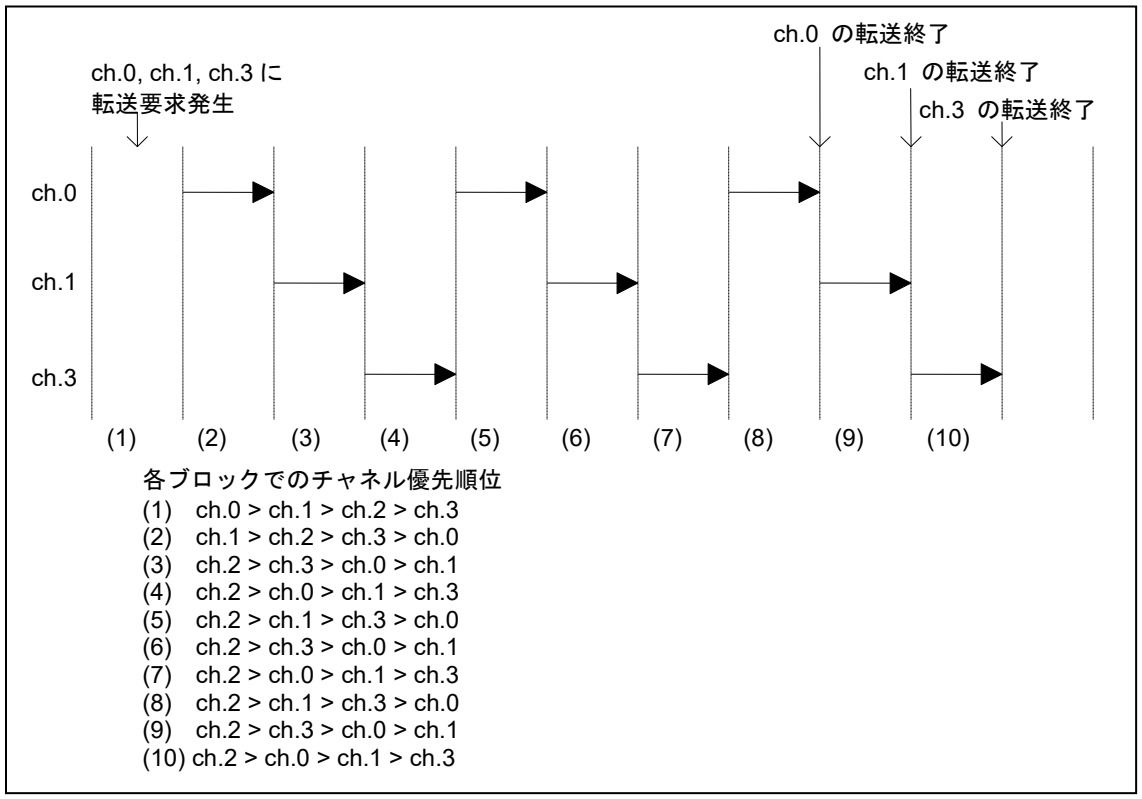


● ラウンドロビン (DMACR:AT = 1)

転送を開始したチャネルの優先順位が一番低くなり、それより下位にあったチャネルの優先順位が1つずつ上がります。ラウンドロビンでは、転送要求が発生した時点での優先順位にしたがって、優先度の高いチャネルから転送を行います。転送を開始したチャネルの優先順位は一番低くなります。ブロック単位の転送ごとに優先順位判定を行い、その時点で優先度の高いチャネルの転送を行います。以下に、転送例を示します。図中の点線はブロック単位の区切りです。

＜例＞ 転送要求 : ch.0, ch.1, ch.3に同時発生
設定 : ch.0, ch.1, ch.3ともにバースト転送モード, 転送回数は3

図 8-7 チャネル優先順位ラウンドロビンにおける転送例



● 転送アドレス更新

DCCRn:TSで設定したサイズのデータを転送するごとに、転送元アドレスと転送先アドレスを更新します。アドレスの更新は増加/減少/固定のいずれかです。増加/減少の場合、アドレス値の増減幅は転送サイズ(DCCRn:TS)により決まります。固定の場合は、アドレス値は変化しません。表 8-5 にアドレス更新におけるアドレス値の増減幅を示します。アドレス更新によりオーバフローが起こったときは、そのビットを切り捨てます。

表 8-5 転送元アドレス，転送先アドレスの更新

アドレス指定		転送サイズ	1転送ごとのアドレス更新	
転送元(SAC)	転送先(DAC)	(TS)	転送元(DSA)	転送先(DDA)
増加 ("00")	増加 ("00")	8 ビット("00")	1 増加	1 増加
		16 ビット ("01")	2 増加	2 増加
		32 ビット ("10")	4 増加	4 増加
	減少 ("01")	8 ビット ("00")	1 増加	1 減少
		16 ビット ("01")	2 増加	2 減少
		32 ビット ("10")	4 増加	4 減少
	固定 ("11")	8 ビット ("00")	1 増加	更新しません。
		16 ビット ("01")	2 増加	
		32 ビット ("10")	4 増加	
減少 ("01")	増加 ("00")	8 ビット ("00")	1 減少	1 増加
		16 ビット ("01")	2 減少	2 増加
		32 ビット ("10")	4 減少	4 増加
	減少 ("01")	8 ビット ("00")	1 減少	1 減少
		16 ビット ("01")	2 減少	2 減少
		32 ビット ("10")	4 減少	4 減少
	固定 ("11")	8 ビット ("00")	1 減少	更新しません。
		16 ビット ("01")	2 減少	
		32 ビット ("10")	4 減少	
固定 ("11")	増加 ("00")	8 ビット ("00")	更新しません。	1 増加
		16 ビット ("01")		2 増加
		32 ビット ("10")		4 増加
	減少 ("01")	8 ビット ("00")		1 減少
		16 ビット ("01")		2 減少
		32 ビット ("10")		4 減少
	固定 ("11")	8 ビット ("00")		更新しません。
		16 ビット ("01")		
		32 ビット ("10")		

● 転送アドレスのリロード

DMACは、転送回数で設定した回数の転送終了後に転送アドレスのリロードを行うことができます。

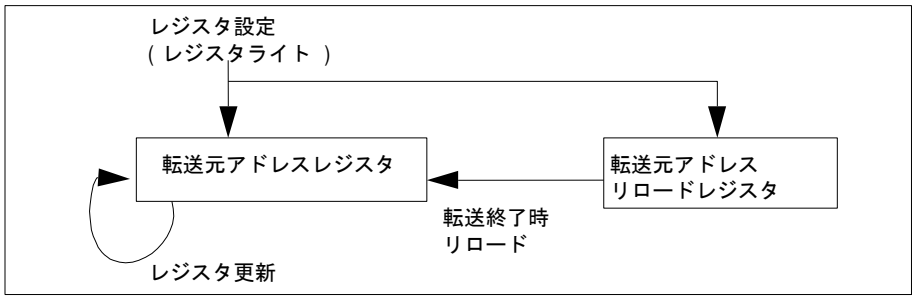
● 転送元アドレスのリロード

転送元アドレスのリロードを指定している場合は、転送終了後にDSARn:DSAビットを初期設定値に戻します。

転送元アドレスのリロードを禁止している場合は、転送終了後のDSARn:DSAビットは最終アドレスの次のアクセスアドレスを示します。

転送元アドレスのリロード設定にかかわらず、転送回数分の転送が終了する前に中断、異常終了した場合は、DSARn:DSAビットは終了したアドレスの次のアクセスアドレスを示します。

図 8-8 転送元アドレスレジスタのリロード



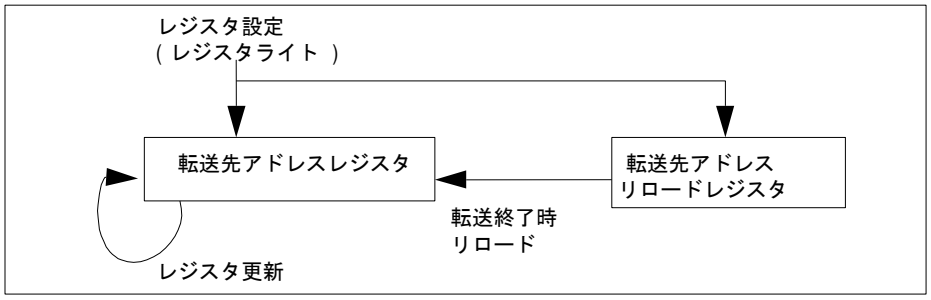
● 転送先アドレスレジスタのリロード

転送先アドレスのリロードを指定している場合は、転送終了後にDDARn:DDAビットを初期設定値に戻します。

転送先アドレスのリロードを禁止している場合は、転送終了後のDDARn:DDAビットは最終アドレスの次のアクセスアドレスを示します。

転送先アドレスのリロード設定にかかわらず、転送回数分の転送が終了する前に中断，異常終了した場合は、DDARn:DDAビットは終了したアドレスの次のアクセスアドレスを示します。

図 8-9 転送先アドレスレジスタのリロード



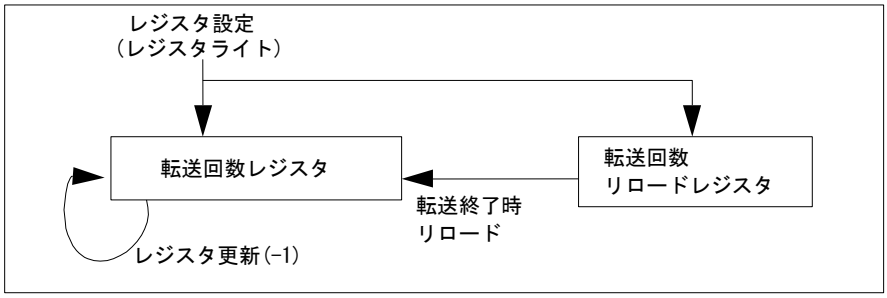
● 転送回数のリロード

転送回数のリロードを指定している場合は、転送終了後にDTCRn:DTCビットを初期設定値に戻します。

転送回数のリロードを禁止している場合は、転送終了後のDTCRn:DTCビットは"0"を示します。

転送回数のリロード設定にかかわらず、転送回数分の転送が終了する前に中断，異常終了した場合は、DTCRn:DTCビットは残りの転送回数を示します。

図 8-10 転送回数アドレスレジスタのリロード



転送回数のリロード設定により、転送終了時のDCCRn:CEビットの状態が異なります。転送回数のリロード設定と転送要求元の関係を次に示します。

表 8-6 転送終了時における DCCRn:CE ビット

	ソフトウェア要求	ソフトウェア以外の要求
転送回数リロードを指定	DCCRn:CE はクリアされる。	DCCRn:CE はクリアされない。
転送回数リロードを禁止	DCCRn:CE はクリアされる。	DCCRn:CE はクリアされる。

● 転送中断

DMACは、以下の要因によりDMA転送を中断します。

- DMACR:DME ビットのクリアによる中断
- DCCRn:CE ビットのクリアによる中断
- 転送要求元ペリフェラルからの転送停止要求による中断

転送の中断は、ブロック単位で行います。転送を中断すると新たな転送は行わず停止状態となります。転送を再開する場合は、中断の要因によって設定が異なります。

- DMACR:DME ビットのクリアによる中断

DMACR:DMEビットをクリアするとすべてのチャネルが停止状態となり、転送中のチャネルはブロック単位の転送が終了した時点で転送を中断します。転送を再開する場合は、DMACR:DMEビットをセットしてください。

- DCCRn:CE ビットのクリアによる中断

DCCRn:CEビットをクリアするとそのチャネルは停止状態となり、ブロック単位の転送が終了した時点で転送を中断します。また、DCCRn:CEビットのクリアにより、既に検出した転送要求もクリアされます。転送を再開する場合は、停止状態にあるチャネルのDCCRn:CEビットをセットし、新たに転送要求を行ってください。

- 転送要求元ペリフェラルからの転送停止要求

次のペリフェラルはそれぞれ示す条件で転送停止要求を発行します。

(A) マルチファンクションシリアルインタフェース

PE, FREまたはOREフラグ発生時

(B) LIN-UART

PE, FREまたはOREフラグ発生時

転送停止要求が発行されると、転送中のデータを1ブロック転送し終わった時点で転送が中断されます。転送が中断されると次の状態が発生します。

- DMA チャネルステータスレジスタ (DCSRn) の SP ビットが"1" に変わる。
- DMA チャネルコントロールレジスタ (DCCRn) の CE ビットが"0" に変わる。
- 既に検出した転送要求がクリアされる。

転送停止要求が発行されている期間は、新たな転送要求を受け付けません。次の手順でDMA 転送を再開してください。

(A), (B) のフラグをクリアして転送停止要求を無効にする。

対応するチャネルのDMAチャネルステータスレジスタ (DCSRn) のSPビットに"0" を書き込む。

DMA チャネルコントロールレジスタ (DCCRn) のCE ビットに"1" を書き込む。

新たに転送要求を行う。

表 8-7 転送中断後の再開のための設定

	DME をクリア	CE をクリア	転送要求元ペリフェラルからの 転送停止要求を検出
転送再開の設定	(1) DME をセット	(1) CE をセット (2) 転送要求	(1) 転送要求をネゲート (2) SP をクリア (3) CE をセット (4) 転送要求

● 転送終了

転送終了には、正常終了と異常終了があります。

● 正常終了

転送回数(DTCRn:DTC)で設定した回数の転送を終了した時点で正常終了となります。正常終了すると、対応するチャネルのDCSRn:NCビットをセットするとともに、DCCRn:CEビットをクリアし停止状態となります。ただし、転送要求元がソフトウェア以外で転送回数リロードを指定している場合、そのチャネルのDCCRn:CEビットはクリアされません。

また、転送回数(DTCRn:DTC)が"0"のときに対応するチャネルのDCCRn:CEに"1"書込みすると、正常終了と同様に、DCSRn:NCビットをセットします。DCCRn:CEに"1"を設定する際は、必ずDTCRn:DTCに"1"以上を設定してから行なってください。

● 異常終了

レジスタに設定禁止の値を設定すると異常終了となります。異常終了すると、対応するチャネルのDCSRn:.ACビットをセットするとともに、DCCRn:CEビットをクリアし停止状態となります。

レジスタ設定禁止項目は以下のとおりです。

- 転送モード : DCCRn:TM = 10
- 転送元アドレスカウント : DCCRn:SAC = 10
- 転送先アドレスカウント : DCCRn:DAC = 10
- 転送サイズ : DCCRn:TS = 11
- ソフトウェア要求によるデマンド転送モード : DCCRn:RS = 00 かつ DCCRn:TM = 11

● 割込み要求

DMACは、転送の正常終了，異常終了，転送停止要求による転送中断によって割込み要求を発生することができます。割込み要求を発生する場合、割込みコントローラの設定も行なってください。

割込み要求の要因確認および割込み要求のクリアはDMAチャネルステータスレジスタ(DCSRn)によって行なってください。

● 正常終了による割込み要求

チャネルの正常終了割込みを許可(DCCRn:NIE=1)している場合、正常終了すると割込み要求を出力します。

ただし、対応するチャネルのDCSRn:NCビットは、正常終了割込み(DCCRn:NIE)の設定によらずセットします。

割込み要求のクリアは、対応するチャネルのDCSRn:NCビットをクリアすることにより行なってください。

● 異常終了による割込み要求

チャネルの異常終了割込みを許可(DCCRn:AIE=1)している場合、異常終了すると割込み要求を出力します。ただし、対応するチャネルのDCSRn:ACビットは、異常終了割込み(DCCRn:AIE)の設定によらずセットします。

割込み要求のクリアは、対応するチャネルのDCSRn:ACビットをクリアすることにより行なってください。

● 転送停止要求による転送中断の割込み要求

チャネルの転送中断割込みを許可(DCCRn:SIE=1)している場合、転送停止要求によって転送を中断すると割込み要求を出力します。ただし、対応するチャネルのDCSRn:SPビットは転送中断割込み(DCCRn:SIE)の設定によらずセットします。

割込み要求のクリアは、対応するチャネルのDCSRn:SPビットをクリアすることにより行なってください。

● DMA 転送の抑止

以下の要因により、DMA転送が抑止されます。

- DSU/OCD からの DMA 転送抑止要求 (デバッグ用)
- NMI
- 周辺割込み

DMA転送の抑止は、ブロック単位で行なわれます。転送が抑止されると新たな転送は行わず、停止状態となります。転送の再開は、DMA転送抑止の要因によって異なります。

● DSU/OCD からの DMA 抑止要求 (デバッグ用)

DSU/OCDからのDMA転送抑止要求がアサートされた場合、新たな転送は行わず転送中の場合はブロック単位で転送を停止します。DSU/OCDからのDMA転送抑止に対しては、アクノリッジを返しません。

● NMI による DMA 転送抑止

NMIHDビットが"0"に設定されている場合、ユーザNMIが発生するとNMIHフラグをセットし、ブロック単位の転送が終了した時点でDMA転送が抑止されます。

転送を再開する場合は、NMIHフラグに"0"を書き込んでください。

● 周辺割込みによる DMA 転送抑止

DILVRに設定した割込みレベルより強い割込みが発生した場合、ブロック単位の転送が終了した時点でDMA転送が抑止されます。

割込み要求をクリアし、割込みレベルがLVL[4:0]と同じか弱いレベルになるとDMA転送を再開します。

表 8-8 DMA 転送抑止のための LVL[4:0]設定

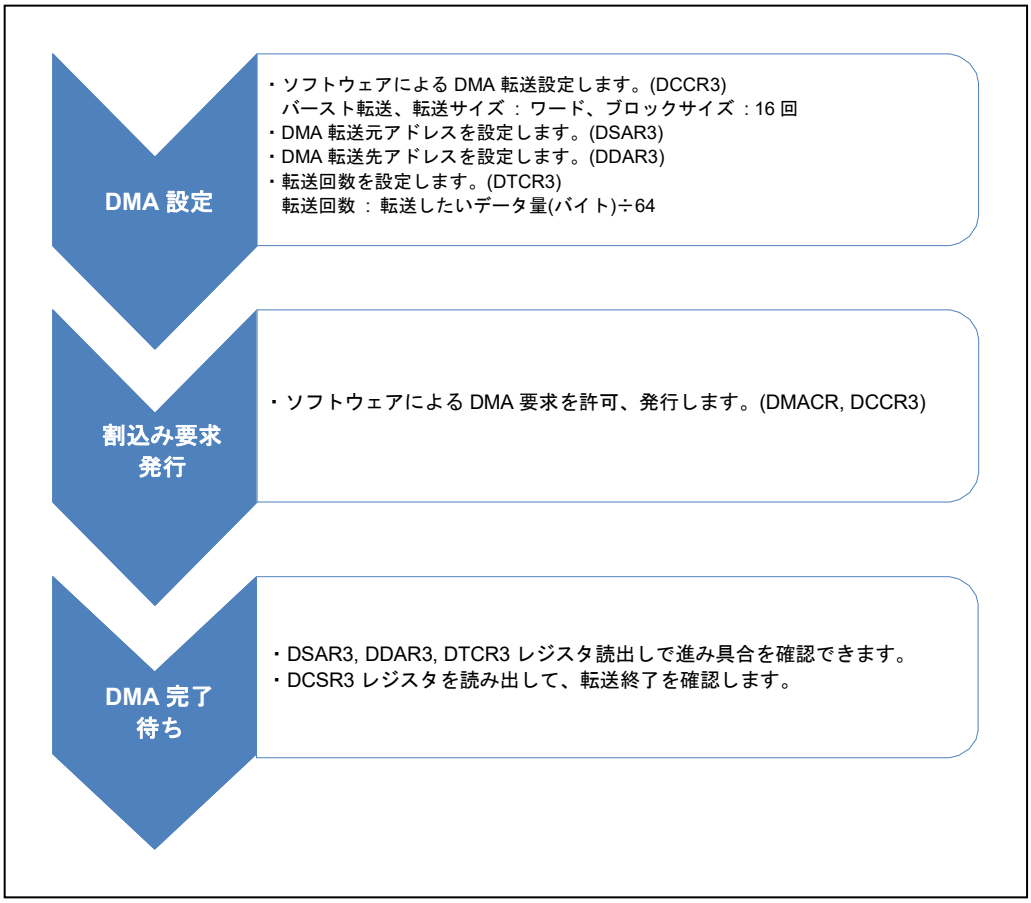
LVL[4:0]	DMA 抑止制御
11111	すべての周辺割込み要求で DMA 転送を抑止します。(初期値)
11110	レベルが $1E_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11101	レベルが $1D_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11100	レベルが $1C_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11011	レベルが $1B_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11010	レベルが $1A_H$ より強い周辺割込み要求で DMA 転送を抑止します。
11001	レベルが 19_H より強い周辺割込み要求で DMA 転送を抑止します。
11000	レベルが 18_H より強い周辺割込み要求で DMA 転送を抑止します。
10111	レベルが 17_H より強い周辺割込み要求で DMA 転送を抑止します。
10110	レベルが 16_H より強い周辺割込み要求で DMA 転送を抑止します。
10101	レベルが 15_H より強い周辺割込み要求で DMA 転送を抑止します。
10100	レベルが 14_H より強い周辺割込み要求で DMA 転送を抑止します。
10011	レベルが 13_H より強い周辺割込み要求で DMA 転送を抑止します。
10010	レベルが 12_H より強い周辺割込み要求で DMA 転送を抑止します。
10001	レベルが 11_H より強い周辺割込み要求で DMA 転送を抑止します。
10000	周辺割込み要求では DMA 転送を抑止しません。

8.6 DMA 使用例

DMA使用例について示します。

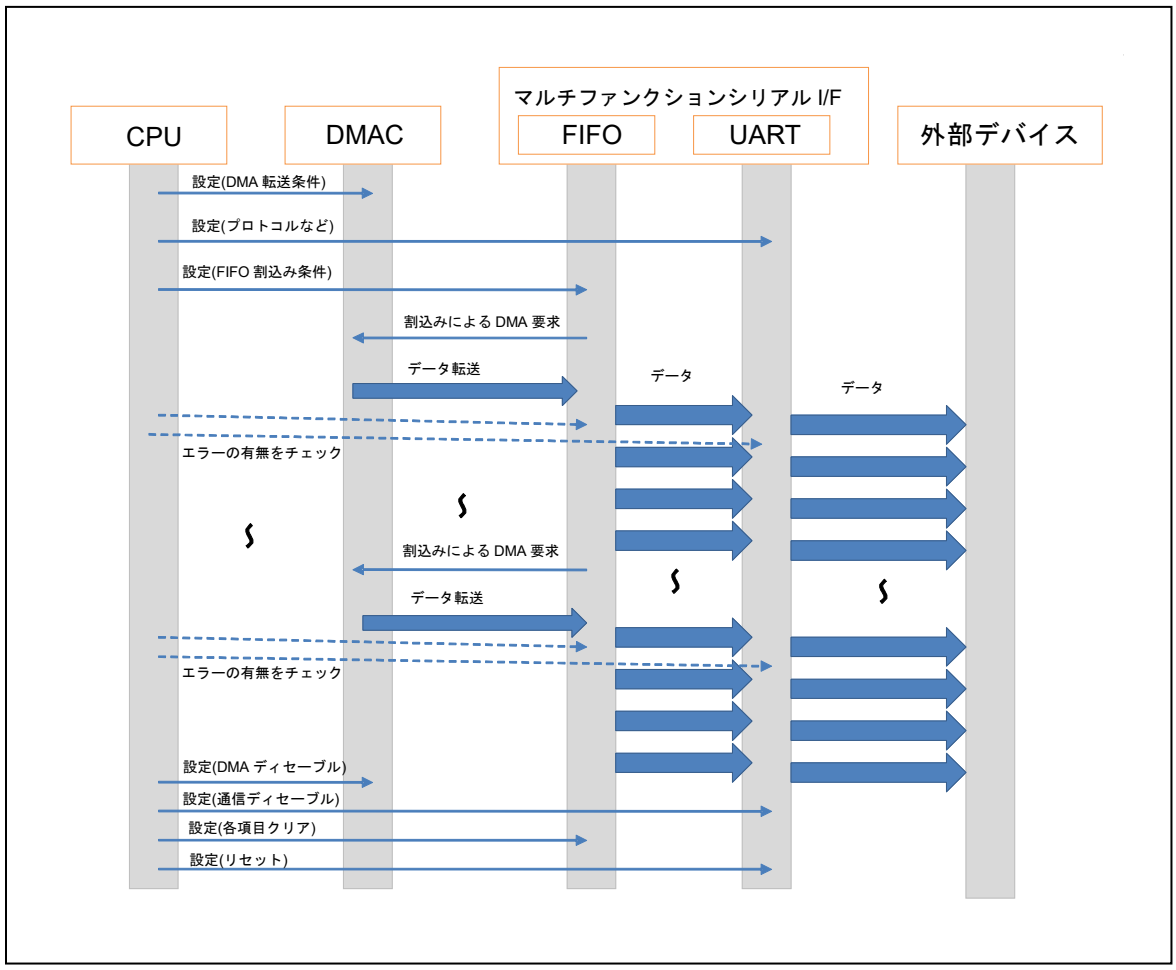
DMAを使用した64バイト単位memcpyの例です。最もシンプルなDMA転送の例です。

図 8-11 DMA を使用した memcpy 例 (ch.3 使用)



DMAを使用したマルチファンクションシリアルインタフェースでの通信例です。
この例ではマルチファンクションシリアルインタフェースの割込みはDMA転送要求に占有されるので、エラー有無のチェックにCPUでのステータスレジスタのポーリングを行っています。

図 8-12 DMA を使用したマルチファンクションシリアルインタフェース通信例



9. DMA 転送要求の発生・クリア

DMA転送要求の発生・クリアについて説明します。

9.1 概要

DMA転送要求の発生・クリアの概要について説明します。

本シリーズは、周辺機能の割込み要求を利用してDMA転送を起動することができます。DMA転送を起動する割込み要求を選択するレジスタがDMAコントローラ(DMAC)のチャネルごとに用意されています。1つの割込みベクタ番号に複数の割込み要求が割り当てられていた場合に、どの割込み要求フラグをDMAコントローラ(DMAC)でクリアするのかも設定する必要があります。

DMAコントローラ(DMAC)のレジスタでDMA転送要求の発生要因(転送要求元)を周辺機能の割込み要求に設定できます。割込みベクタ番号に対応する値を指定して、使用する割込み要求を選択します。

9.2 特長

DMA転送要求の発生・クリアの特長について説明します。

■ 転送要求の発生設定

16チャネルのDMAに対し、それぞれに割込みベクタ番号0x10(10進で16)から0x3F(10進で63)の、どの割込みによりDMA転送要求を発生させるかを指定します。

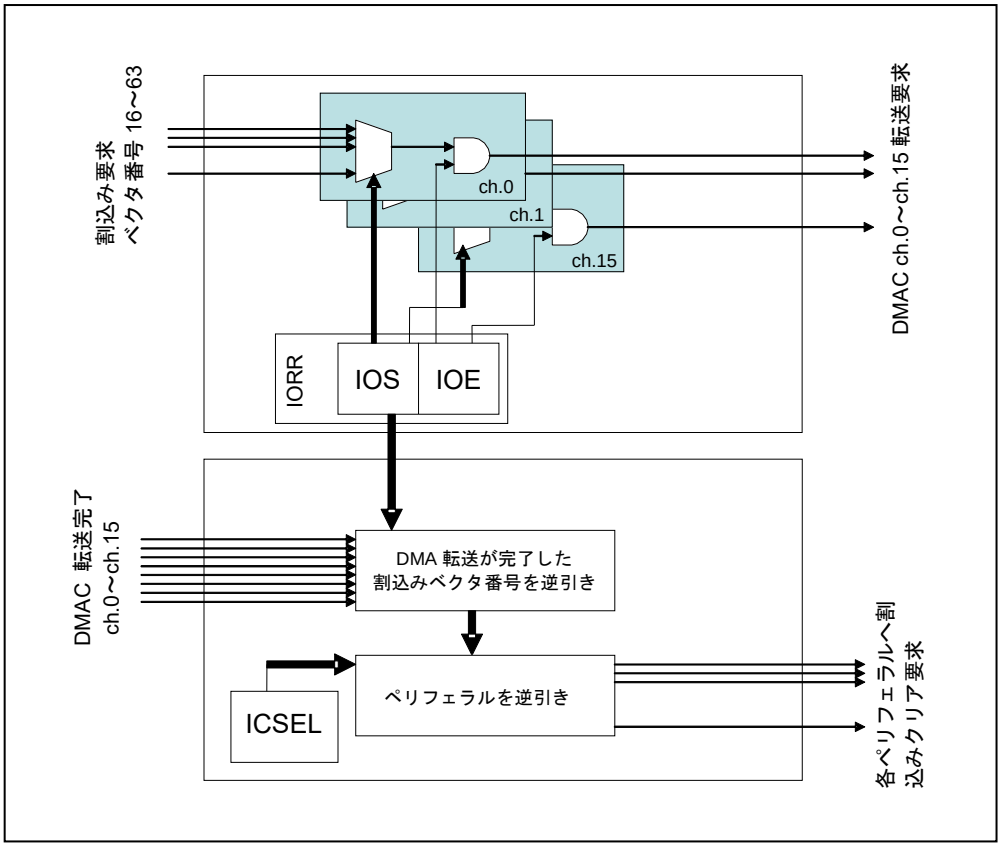
■ 割込みクリアの設定

DMAの転送終了後に、転送要求元が複数の割込み元ペリフェラルを持つベクタ番号だった場合、どの割込み元ペリフェラルの割込み要求をクリアするかを設定します。

9.3 構成

DMA転送要求の発生・クリアの構成について説明します。

図 9-1 ブロックダイヤグラム



9.4 レジスタ

DMA転送要求の発生・クリアのレジスタについて説明します。

表 9-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0400	ICSEL0	ICSEL1	ICSEL2	ICSEL3	DMA クリア要求レジスタ 0(ベクタ番号#16 用) DMA クリア要求レジスタ 1(ベクタ番号#17 用) DMA クリア要求レジスタ 2(ベクタ番号#18 用) DMA クリア要求レジスタ 3(ベクタ番号#19 用)
0x0404	ICSEL4	ICSEL5	ICSEL6	ICSEL7	DMA クリア要求レジスタ 4(ベクタ番号#38 用) DMA クリア要求レジスタ 5(ベクタ番号#39 用) DMA クリア要求レジスタ 6(ベクタ番号#40 用) DMA クリア要求レジスタ 7(ベクタ番号#41 用)
0x0408	ICSEL8	ICSEL9	ICSEL10	ICSEL11	DMA クリア要求レジスタ 8(ベクタ番号#42 用) DMA クリア要求レジスタ 9(ベクタ番号#43 用) DMA クリア要求レジスタ 10(ベクタ番号#44 用) DMA クリア要求レジスタ 11(ベクタ番号#46 用)
0x040C	ICSEL12	ICSEL13	ICSEL14	ICSEL15	DMA クリア要求レジスタ 12(ベクタ番号#47 用) DMA クリア要求レジスタ 13(ベクタ番号#52 用) DMA クリア要求レジスタ 14(ベクタ番号#53 用) DMA クリア要求レジスタ 15(ベクタ番号#54 用)
0x0410	ICSEL16	ICSEL17	ICSEL18	ICSEL19	DMA クリア要求レジスタ 16(ベクタ番号#55 用) DMA クリア要求レジスタ 17(ベクタ番号#56 用) DMA クリア要求レジスタ 18(ベクタ番号#57 用) DMA クリア要求レジスタ 19(ベクタ番号#58 用)
0x0414	ICSEL20	ICSEL21	ICSEL22	予約	DMA クリア要求レジスタ 20(ベクタ番号#59 用) DMA クリア要求レジスタ 21(ベクタ番号#60 用) DMA クリア要求レジスタ 22(ベクタ番号#61 用)
0x0490	IORR0	IORR1	IORR2	IORR3	IO 転送要求レジスタ 0 IO 転送要求レジスタ 1 IO 転送要求レジスタ 2 IO 転送要求レジスタ 3
0x0494	IORR4	IORR5	IORR6	IORR7	IO 転送要求レジスタ 4 IO 転送要求レジスタ 5 IO 転送要求レジスタ 6 IO 転送要求レジスタ 7
0x0498	IORR8	IORR9	IORR10	IORR11	IO 転送要求レジスタ 8 IO 転送要求レジスタ 9 IO 転送要求レジスタ 10 IO 転送要求レジスタ 11
0x049C	IORR12	IORR13	IORR14	IORR15	IO 転送要求レジスタ 12 IO 転送要求レジスタ 13 IO 転送要求レジスタ 14 IO 転送要求レジスタ 15

9.4.1 DMA 要求クリアレジスタ 0 : ICSEL0 (Interrupt Clear SELect register 0)

DMA要求クリアレジスタ0のビット構成について示します。

■ ICSEL0： アドレス 0400H (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					EISEL[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W

[bit2～bit0] EISEL[2:0] (External Interrupt request SELection)： 外部割込み0-7割込みクリア選択ビット

割込みベクタ番号#16に割り当てられた、割込みをクリアするペリフェラルを選択します。

EISEL[2:0]	クリア対象
000	外部割込み 0
001	外部割込み 1
010	外部割込み 2
011	外部割込み 3
100	外部割込み 4
101	外部割込み 5
110	外部割込み 6
111	外部割込み 7

9.4.2 DMA 要求クリアレジスタ 1 : ICSEL1 (Interrupt Clear SELect register 1)

DMA要求クリアレジスタ1のビット構成について示します。

■ ICSEL1 : アドレス 0401H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					EISEL[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W

[bit2～bit0] EISEL[2:0] (External Interrupt request SELection) : 外部割込み8～15割込みクリア選択ビット

割込みベクタ番号#17に割り当てられた、割込みをクリアするペリフェラルを選択します。

EISEL[2:0]	クリア対象
000	外部割込み 8
001	外部割込み 9
010	外部割込み 10
011	外部割込み 11
100	外部割込み 12
101	外部割込み 13
110	外部割込み 14
111	外部割込み 15

9.4.3 DMA 要求クリアレジスタ 2 : ICSEL2 (Interrupt Clear SELect register 2)

DMA要求クリアレジスタ2のビット構成について示します。

■ ICSEL2 : アドレス 0402H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							RTSEL0
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] RTSEL0 (Reload Timer SELection) : リロードタイマ0/1割込みクリア選択ビット

割込みベクタ番号#18に割り当てられた、割込みをクリアするペリフェラルを選択します。

＜注意事項＞

リロードタイマのch.0とch.4およびch.1とch.5は割込みを共用しています。そのため、ch.0でDMA転送を行う場合はch.4の割込みが上がらないように設定してください。同様に、ch.1でDMA転送を行う場合は、ch.5の割込みが上がらないように設定してください。

RTSEL0	クリア対象
0	リロードタイマ 0
1	リロードタイマ 1

9.4.4 DMA 要求クリアレジスタ 3 : ICSEL3 (Interrupt Clear SELect register 3)

DMA要求クリアレジスタ3のビット構成について示します。

■ ICSEL3 : アドレス 0403H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							RTSEL1
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] RTSEL1 (Reload Timer SELection) : リロードタイマ2/3割込みクリア選択ビット

割込みベクタ番号#19に割り当てられた、割込みをクリアするペリフェラルを選択します。

＜注意事項＞

リロードタイマのch.2とch.6は割込みを共用しています。そのため、ch.2でDMA転送を行う場合はch.6の割込みが上がらないように設定してください。

RTSEL1	クリア対象
0	リロードタイマ 2
1	リロードタイマ 3

9.4.5 DMA 要求クリアレジスタ 4 : ICSEL4 (Interrupt Clear SELect register 4)

DMA要求クリアレジスタ4のビット構成について示します。

■ ICSEL4 : アドレス 0404H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							SG_RX_SEL0
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] SG_RX_SEL0 (SG_RX SELection0) : サウンドジェネレータch.0 / LIN-UART ch.7 受信完了割込みクリア選択ビット

割込みベクタ番号#38に割り当てられた、割込みをクリアするペリフェラルを選択します。

SG_RX_SEL0	クリア対象
0	サウンドジェネレータ ch.0
1	LIN-UART ch.7 受信完了

9.4.6 DMA 要求クリアレジスタ 5 : ICSEL5 (Interrupt Clear SELect register 5)

DMA要求クリアレジスタ5のビット構成について示します。

■ ICSEL5 : アドレス 0405H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							SG_RX_SEL1
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] SG_RX_SEL1 (SG_RX SELection1) : サウンドジェネレータch.1 / LIN-UART ch.7 送信完了割込みクリア選択ビット

割込みベクタ番号#39に割り当てられた、割込みをクリアするペリフェラルを選択します。

SG_RX_SEL1	クリア対象
0	サウンドジェネレータ ch.1
1	LIN-UART ch.7 送信完了

9.4.7 DMA 要求クリアレジスタ 6 : ICSEL6 (Interrupt Clear SELect register 6)

DMA要求クリアレジスタ6のビット構成について示します。

■ ICSEL6 : アドレス 0406H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					PPGSEL0[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W

[bit2-bit0] PPGSEL0[2:0] (PPG SELection0) : PPG0,1,10,11,20,21 割込みクリア選択ビット

割込みベクタ番号#40に割り当てられた、割込みをクリアするペリフェラルを選択します。

PPGSEL0[2:0]	クリア対象
000	PPG0
001	PPG1
010	PPG10
011	PPG11
100	PPG20
101	PPG21
110	予約 (どの割込みもクリアされません)
111	予約 (どの割込みもクリアされません)

＜注意事項＞

PPGSEL0[2:0]="3'b110","3'b111"は設定禁止とし、この設定時にはどの割込みクリアも選択されません。

9.4.8 DMA 要求クリアレジスタ 7 : ICSEL7 (Interrupt Clear SELect register 7)

DMA要求クリアレジスタ7のビット構成について示します。

■ ICSEL7 : アドレス 0407H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					PPGSEL1[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W

[bit2～bit0] PPGSEL 1[2:0](PPG SELection1) : PPG2,3,12,13,22,23割込みクリア選択ビット

割込みベクタ番号#41に割り当てられた、割込みをクリアするペリフェラルを選択します。

PPGSEL1[2:0]	クリア対象
000	PPG2
001	PPG3
010	PPG12
011	PPG13
100	PPG22
101	PPG23
110	予約 (どの割込みもクリアされません)
111	予約 (どの割込みもクリアされません)

＜注意事項＞

PPGSEL1[2:0]="3'b110","3'b111"は設定禁止とし、この設定時にはどの割込みクリアも選択されません。

9.4.9 DMA 要求クリアレジスタ 8 : ICSEL8 (Interrupt Clear SELect register 8)

DMA要求クリアレジスタ8のビット構成について示します。

■ ICSEL8 : アドレス 0408H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						PPGSEL2[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit1, bit0] PPGSEL2[1:0] (PPG SELection2) : PPG4,5,14,15 割込みクリア選択ビット

割込みベクタ番号#42に割り当てられた、割込みをクリアするペリフェラルを選択します。

PPGSEL2[1:0]	クリア対象
00	PPG4
01	PPG5
10	PPG14
11	PPG15

9.4.10 DMA 要求クリアレジスタ 9 : ICSEL9 (Interrupt Clear SELect register 9)

DMA要求クリアレジスタ9のビット構成について示します。

■ ICSEL9 : アドレス 0409H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						PPGSEL3[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit1, bit0] PPGSEL3[1:0] (PPG SELection3) : PPG6,7,16,17割込みクリア選択ビット

割込みベクタ番号#43に割り当てられた、割込みをクリアするペリフェラルを選択します。

PPGSEL3[1:0]	クリア対象
00	PPG6
01	PPG7
10	PPG16
11	PPG17

9.4.11 DMA 要求クリアレジスタ 10 : ICSEL10 (Interrupt Clear SELect register 10)

DMA要求クリアレジスタ10のビット構成について示します。

■ ICSEL10 : アドレス 040AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						PPGSEL4[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit1, bit0] PPGSEL4[1:0] (PPG SELection4) : PPG8,9,18,19割込みクリア選択ビット

割込みベクタ番号#44に割り当てられた、割込みをクリアするペリフェラルを選択します。

PPGSEL4[1:0]	クリア対象
00	PPG8
01	PPG9
10	PPG18
11	PPG19

9.4.12 DMA 要求クリアレジスタ 11 : ICSEL11 (Interrupt Clear SELect register 11)

DMA要求クリアレジスタ11のビット構成について示します。

■ ICSEL11 : アドレス 040BH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						PMSTSEL[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit1, bit0] PMSTSEL[1:0] (PLL, Main, Sub Timer SELection) : メインタイマ/サブタイマ/PLLタイマ/ マルチファンクションシリアルch.8送信完了割込みクリア選択ビット

割込みベクタ番号#46に割り当てられた、割込みをクリアするペリフェラルを選択します。

PMSTSEL[1:0]	クリア対象
00	メインタイマ
01	サブタイマ
10	PLL タイマ
11	マルチファンクションシリアル ch.8(送信完了)

9.4.13 DMA 要求クリアレジスタ 12 : ICSEL12 (Interrupt Clear SELect register 12)

DMA要求クリアレジスタ12のビット構成について示します。

■ ICSEL12 : アドレス 040CH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						SG_RX_SEL[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit1, bit0] SG_RX_SEL[1:0] (SG_RX Selection) : SG4/マルチファンクションシリアルch.9受信完了割込みクリア選択ビット

割込みベクタ番号#47に割り当てられた、割込みをクリアするペリフェラルを選択します。

SG_RX_SEL[1:0]	クリア対象
00	予約 (どの割込みもクリアされません)
01	サウンドジェネレータ ch.4
10	マルチファンクションシリアル ch.9(受信完了)
11	予約 (どの割込みもクリアされません)

9.4.14 DMA 要求クリアレジスタ 13 : ICSEL13 (Interrupt Clear SELect register 13)

DMA要求クリアレジスタ13のビット構成について示します。

■ ICSEL13 : アドレス 040DH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							ICUSEL0
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] ICUSEL0 : ICU ch.0, ch.6割込みクリア選択ビット

割込みベクタ番号#52に割り当てられた、割込みをクリアするペリフェラルを選択します。

ICUSEL0	クリア対象
0	ICU ch.0
1	ICU ch.6

9.4.15 DMA 要求クリアレジスタ 14 : ICSEL14 (Interrupt Clear SELect register 14)

DMA要求クリアレジスタ14のビット構成について示します。

■ ICSEL14 : アドレス 040EH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							ICUSEL1
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] ICUSEL1 : ICU ch.1, ch.7割込みクリア選択ビット

割込みベクタ番号#53に割り当てられた、割込みをクリアするペリフェラルを選択します。

ICUSEL1	クリア対象
0	ICU ch.1
1	ICU ch.7

9.4.16 DMA 要求クリアレジスタ 15 : ICSEL15 (Interrupt Clear SELect register 15)

DMA要求クリアレジスタ15のビット構成について示します。

■ ICSEL15 : アドレス 040F$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							ICUSEL2
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] ICUSEL2 : ICU ch.2, ch.8割込みクリア選択ビット

割込みベクタ番号#54に割り当てられた、割込みをクリアするペリフェラルを選択します。

ICUSEL2	クリア対象
0	ICU ch.2
1	ICU ch.8

9.4.17 DMA 要求クリアレジスタ 16 : ICSEL16 (Interrupt Clear SELect register 16)

DMA要求クリアレジスタ16のビット構成について示します。

■ ICSEL16 : アドレス 0410$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							ICUSEL3
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] ICUSEL3 : ICU ch.3, ch.9割込みクリア選択ビット

割込みベクタ番号#55に割り当てられた、割込みをクリアするペリフェラルを選択します。

ICUSEL3	クリア対象
0	ICU ch.3
1	ICU ch.9

9.4.18 DMA 要求クリアレジスタ 17 : ICSEL17 (Interrupt Clear SELect register 17)

DMA要求クリアレジスタ17のビット構成について示します。

■ ICSEL17 : アドレス 0411H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							ICUSEL4
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] ICUSEL4 : ICU ch.4, ch.10割込みクリア選択ビット

割込みベクタ番号#56に割り当てられた、割込みをクリアするペリフェラルを選択します。

ICUSEL4	クリア対象
0	ICU ch.4
1	ICU ch.10

9.4.19 DMA 要求クリアレジスタ 18 : ICSEL18 (Interrupt Clear SELect register 18)

DMA要求クリアレジスタ18のビット構成について示します。

■ ICSEL18 : アドレス 0412H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							ICUSEL5
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit0] ICUSEL5 : ICU ch.5, ch.11割込みクリア選択ビット

割込みベクタ番号#57に割り当てられた、割込みをクリアするペリフェラルを選択します。

ICUSEL5	クリア対象
0	ICU ch.5
1	ICU ch.11

9.4.20 DMA 要求クリアレジスタ 19 : ICSEL19 (Interrupt Clear SELect register 19)

DMA要求クリアレジスタ19のビット構成について示します。

■ ICSEL19 : アドレス 0413H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					OCUSEL0[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W

[bit2～bit0] OCUSEL0[2:0] (OCU Selection0) OCU0, 1, 6, 7,10, 11 : 割込みクリア選択ビット

割込みベクタ番号#58に割り当てられた、割込みをクリアするペリフェラルを選択します。

OCUSEL0[2:0]	クリア対象
000	OCU0
001	OCU1
010	OCU6
011	OCU7
100	OUC10
101	OCU11
110	予約 (どの割込みもクリアされません)
111	予約 (どの割込みもクリアされません)

＜注意事項＞

OCUSEL0[2:0]="3'b110","3'b111"は設定禁止とし、この設定時にはどの割込みクリアも選択されません。

9.4.21 DMA 要求クリアレジスタ 20 : ICSEL20 (Interrupt Clear SELect register 20)

DMA要求クリアレジスタ20のビット構成について示します。

■ ICSEL20 : アドレス 0414H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					OCUSEL1[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W

[bit2～bit0] OCUSEL1[2:0] (OCU Selection1) OCU2, 3, 4, 5, 8, 9 : 割込みクリア選択ビット

割込みベクタ番号#59に割り当てられた、割込みをクリアするペリフェラルを選択します。

OCUSEL1[2:0]	クリア対象
000	OCU2
001	OCU3
010	OCU4
011	OCU5
100	OCU8
101	OCU9
110	予約 (どの割込みもクリアされません)
111	予約 (どの割込みもクリアされません)

＜注意事項＞

OCUSEL1[2:0]="3'b110","3'b111"は設定禁止とし、この設定時にはどの割込みクリアも選択されません。

9.4.22 DMA 要求クリアレジスタ 21 : ICSEL21 (Interrupt Clear SELect register 21)

DMA要求クリアレジスタ21のビット構成について示します。

■ ICSEL21 : アドレス 0415H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						BT_SG_SEL0[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit1, bit0] BT_SG_SEL0[1:0] (BT_SG Selection0) : Base Timer0 IRQ0, IRQ1/ SG2割込みクリア選択ビット

割込みベクタ番号#60に割り当てられた、割込みをクリアするペリフェラルを選択します。

BT_SG_SEL0[1:0]	クリア対象
00	Base Timer0 IRQ0
01	Base Timer0 IRQ1
10	サウンドジェネレータ ch.2
11	予約 (どの割込みもクリアされません)

＜注意事項＞

BT_SG_SEL0[1:0]="2'b11"は設定禁止とし、この設定時にはどの割込みクリアも選択されません。

9.4.23 DMA 要求クリアレジスタ 22 : ICSEL22 (Interrupt Clear SELect register 22)

DMA要求クリアレジスタ22のビット構成について示します。

■ ICSEL22 : アドレス 0416H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						BT_SG_SEL1[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W

[bit1, bit0] BT_SG_SEL1[1:0] (BT_SG_Selection1) : Base Timer1 IRQ0, IRQ1/ SG3割込みクリア選択ビット

割込みベクタ番号#61に割り当てられた、割込みをクリアするペリフェラルを選択します。

BT_SG_SEL1[1:0]	クリア対象
00	Base Timer1 IRQ0
01	Base Timer1 IRQ1
10	サウンドジェネレータ ch.3
11	予約 (どの割込みもクリアされません)

＜注意事項＞

BT_SG_SEL1[1:0]="2'b11"は設定禁止とし、この設定時にはどの割込みクリアも選択されません。

9.4.24 IO 転送要求設定レジスタ 0-15 : IORR0-15 (IO triggered DMA Request Register for ch.0-15)

IO転送要求設定レジスタ0-15のビット構成について示します。

DMA転送要求の発生要因をペリフェラルの割込み要求に設定したときに、どのベクタ番号の割込み要求をDMA転送要求の発生要因にするかを設定するレジスタです。

DMAコントローラ(DMAC)のチャネルごとに、このレジスタが用意されています。

■ IORR0-15 : アドレス 0490H ～ 049FH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	IOE	IOS[5:0]					
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7] 予約

本ビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit6] IOE (IO Enabled) : 転送要求許可ビット

IOS5～IOS0ビットで指定した割込み要求が発生したときに、対応するチャネルのDMAコントローラ(DMAC)に、DMA転送要求を出力するかどうかを設定します。

IOE	機能
0	DMA転送要求を出力しません -- ペリフェラルで発生した割込み要求をDMA転送要求として使用しません (初期値)
1	DMA 転送要求を出力します

[bit5～bit0] IOS (IO triggered DMA transfer request Select) : 転送要求選択ビット

このレジスタに対応するチャネルのDMAコントローラ(DMAC)が、どのベクタ番号の割込み要求を転送要求元として使用するか設定します。

IOS[5:0]	割込みベクタ番号(16 進)
000000	0x10 (初期値)
000001	0x11
000010	0x12
000011	0x13
000100	0x14
000101	0x15
:	:
101100	0x3C
101101	0x3D
101110	0x3E
101111	0x3F
11xxxx	予約

＜注意事項＞

同じ割込みベクタ番号の割込み要求が、複数のDMAチャネルの転送要求元になる設定(例: IORR0=0x42とIORR1=0x42の同時設定) は禁止です。

9.5 動作説明

DMA転送要求の発生・クリアの動作について説明します。

9.5.1. 設定

9.5.2. 注意事項

9.5.1 設定

DMA転送要求の発生・クリアの設定について説明します。

設定する順番は次のようになります。
(1) IORRに、転送要求元ペリフェラルの割込みベクタ番号とIOEビットをセット。
(2) (1)で選択したベクタ番号に複数のペリフェラルがある場合は、ICSELをセット。
(3) ペリフェラル側の割込み設定関連のレジスタを設定。
(4) DMACを設定。

9.5.2 注意事項

DMA転送要求の発生・クリアの注意事項について説明します。

DMACがペリフェラルの要求によるDMA転送を許可している状態で、IORRおよびICSELを変更しないでください。

リソースナンバ割り当て(『付録』を参照してください)のないペリフェラルには、DMA転送完了後の割込みクリア機構はありません。したがって、これらのペリフェラルにDMA転送要求発生を割り当てても、DMA転送完了後に割込みがクリアされませんので注意してください。

転送要求に使用した割込み要求は，CPUへの割込要求としてもみえますので割込みコントローラの設定を割込み禁止に設定してください。(ICRレジスタ)

10. FixedVector 機能

FixedVector 機能について説明します。

10.1 概要

FixedVector 機能の概要について説明します。

FixedVector 機能は、リセット時の割込みベクタに当該番地(0xF_FFFC)のフラッシュメモリの内容でなく、フラッシュメモリの先頭アドレス＋0x0024 番地を返す機能です。

■ 特長

- FixedVector 機能が返す、リセット時の割込みベクタ
- 0x0007_0024

■ 構成

構成図は、『フラッシュメモリ』の章の『図 3-2』を参照してください。

■ レジスタ

ありません。

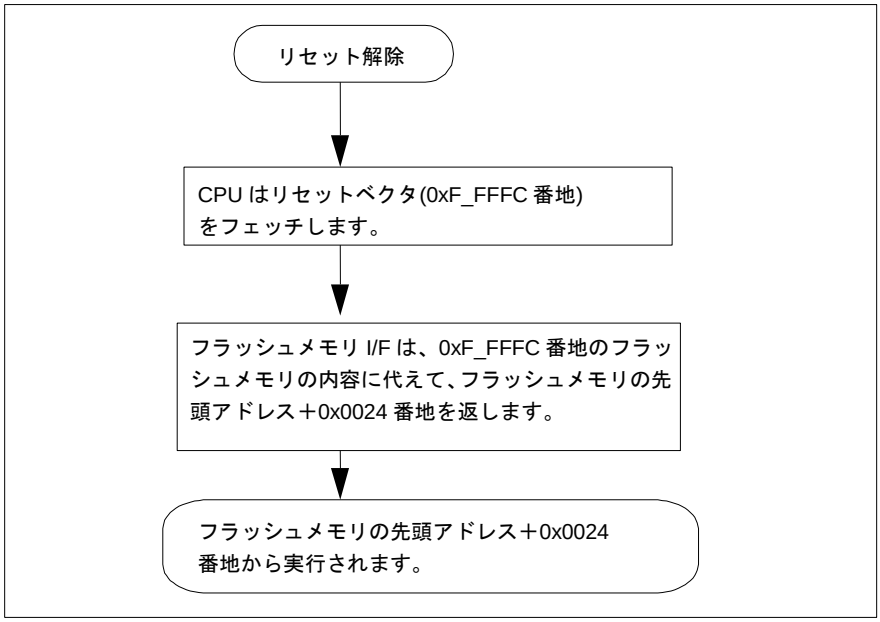
10.2 動作説明

FixedVector 機能の動作について説明します。

■ リセット解除後の動作

次に示すフローで、リセット解除時にフラッシュメモリ内の 0xF_FFFC 番地の内容に代えて、フラッシュメモリの先頭アドレス＋0x0024 番地を返します。

図 10-1 リセット後動作フロー



■ 使用方法

本シリーズはリセット解除後、0x000F_FFFC 番地に書き込まれた値でなく、フラッシュメモリの先頭アドレス＋0x0024 番地から実行されます。

■ 注意事項

リセットベクタフェッチ以外での 0x000F_FFFC ～ 0x000F_FFFF 番地の読出し(例:TBR が初期値(=0x000F_FC00) のときに INT #00H 実行した際の呼び出し先) では、0x000F_FFFC ～ 0x000F_FFFF 番地のフラッシュメモリの内容が返されます。

11. I/O ポート

I/O ポートについて説明します。

11.1 概要

I/O ポートの概要について説明します。

本シリーズにおいて、外部端子の割り当て設定(ペリフェラルならびに外バス)、ならびに外部端子を I/O ポートとして利用する際の設定を説明します。

11.2 特長

I/O ポートの特長について説明します。

● I/O マルチプレクシング

1 つの外部端子に複数のペリフェラルから入出力が割り当てられている場合、どのペリフェラルを使用するかを選択します。

● I/O リロケーション

1 つのペリフェラルの 1 つの端子が、複数の外部端子入出力に対応している場合、どの外部端子を使用するかを選択します。

● ポート機能

外部端子を General Purpose I/O として出力にしてその値を設定したり、入力にしたりして入力値を読み取ります。

図 11-1 I/O マルチプレクシング、I/O リロケーション説明図



11.3 構成

I/O ポートの構成について説明します。

構成図はありません。

11.4 レジスタ

I/O ポートのレジスタについて説明します。

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0000	PDR00	PDR01	PDR02	PDR03	ポートデータレジスタ 00～19*
0x0004	PDR04	PDR05	PDR06	PDR07	
0x0008	PDR08	PDR09	PDR10	PDR11	
0x000C	PDR12	PDR13	PDR14	PDR15	
0x0010	PDR16	PDR17	PDR18	PDR19	
0x0E00	DDR00	DDR01	DDR02	DDR03	データ方向レジスタ 00～19*
0x0E04	DDR04	DDR05	DDR06	DDR07	
0x0E08	DDR08	DDR09	DDR10	DDR11	
0x0E0C	DDR12	DDR13	DDR14	DDR15	
0x0E10	DDR16	DDR17	DDR18	DDR19	
0x0E20	PFR00	PFR01	PFR02	PFR03	ポート機能レジスタ 00～19*
0x0E24	PFR04	PFR05	PFR06	PFR07	
0x0E28	PFR08	PFR09	PFR10	PFR11	
0x0E2C	PFR12	PFR13	PFR14	PFR15	
0x0E30	PFR16	PFR17	PFR18	PFR19	
0x0E40	PDDR00	PDDR01	PDDR02	PDDR03	入力データダイレクトリードレジスタ 00～19*
0x0E44	PDDR04	PDDR05	PDDR06	PDDR07	
0x0E48	PDDR08	PDDR09	PDDR10	PDDR11	
0x0E4C	PDDR12	PDDR13	PDDR14	PDDR15	
0x0E50	PDDR16	PDDR17	PDDR18	PDDR19	
0x0E60	EPFR00	EPFR01	EPFR02	EPFR03	拡張ポート機能レジスタ 00～35
0x0E64	EPFR04	EPFR05	EPFR06	EPFR07	
0x0E68	EPFR08	EPFR09	EPFR10	EPFR11	
0x0E6C	EPFR12	EPFR13	EPFR14	EPFR15	
0x0E70	EPFR16	EPFR17	EPFR18	EPFR19	
0x0E74	EPFR20	EPFR21	EPFR22	EPFR23	
0x0E78	EPFR24	EPFR25	EPFR26	EPFR27	
0x0E7C	EPFR28	EPFR29	EPFR30	EPFR31	
0x0E80	EPFR32	EPFR33	EPFR34	EPFR35	
0x0E84	EPFR36	EPFR37	EPFR38	EPFR39	拡張ポート機能レジスタ 36～54
0x0E88	EPFR40	EPFR41	EPFR42	EPFR43	
0x0E8C	EPFR44	EPFR45	EPFR46	EPFR47	
0x0E90	予約	予約	予約	予約	
0x0E94	EPFR52	EPFR53	EPFR54	予約	

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0EA0	PPCR00	PPCR01	PPCR02	PPCR03	ポートプルアップダウン制御レジスタ 00～19*
0x0EA4	PPCR04	PPCR05	PPCR06	PPCR07	
0x0EA8	PPCR08	PPCR09	PPCR10	PPCR11	
0x0EAC	PPCR12	PPCR13	PPCR14	PPCR15	
0x0EB0	PPCR16	PPCR17	PPCR18	PPCR19	
0x0EC0	PPER00	PPER01	PPER02	PPER03	ポートプルアップダウン許可レジスタ 00～19*
0x0EC4	PPER04	PPER05	PPER06	PPER07	
0x0EC8	PPER08	PPER09	PPER10	PPER11	
0x0ECC	PPER12	PPER13	PPER14	PPER15	
0x0ED0	PPER16	PPER17	PPER18	PPER19	
0x0EE0	PILR00	PILR01	PILR02	PILR03	ポート入力レベル選択レジスタ 00～19*
0x0EE4	PILR04	PILR05	PILR06	PILR07	
0x0EE8	PILR08	PILR09	PILR10	PILR11	
0x0EEC	PILR12	PILR13	PILR14	PILR15	
0x0EF0	PILR16	PILR17	PILR18	PILR19	
0x0F00	EPILR00	EPILR01	EPILR02	EPILR03	拡張ポート入力レベル選択レジスタ 00～19*
0x0F04	EPILR04	EPILR05	EPILR06	EPILR07	
0x0F08	EPILR08	EPILR09	EPILR10	EPILR11	
0x0F0C	EPILR12	EPILR13	EPILR14	EPILR15	
0x0F10	EPILR16	EPILR17	EPILR18	EPILR19	
0x0F20	PODR00	PODR01	PODR02	PODR03	ポート出力駆動レジスタ 00～19*
0x0F24	PODR04	PODR05	PODR06	PODR07	
0x0F28	PODR08	PODR09	PODR10	PODR11	
0x0F2C	PODR12	PODR13	PODR14	PODR15	
0x0F30	PODR16	PODR17	PODR18	PODR19	
0x0F34	予約	EPODR01	EPODR02	EPODR03	拡張ポート出力駆動レジスタ 01,02,03,06,07,08
0x0F38	EPODR06	EPODR07	EPODR08	予約	
0x0F40	PORTEN	予約	予約	予約	ポート入力許可レジスタ

*: LQFP-208 でのみ使用可能なレジスタを含みます。LQFP-144 で書込み、読出した場合、動作に影響を与えません。

11.4.1 ポートデータレジスタ 00～19 : PDR00-19 (Port Data Register 00-19)

ポートデータレジスタ 00～19 のビット構成について示します。

ポート出力モードにおけるポートごとの端子の出力レベルを格納するレジスタです。

■ PDR00-19 : アドレス 0000_H、0001_H、・・・(アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	X	X	X	X	X	X	X	X
属性	R,RM/W	R,RM/W	R,RM/W	R,RM/W	R,RM/W	R,RM/W	R,RM/W	R,RM/W

[bit7～bit0] P[7:0] (Port) : ポートデータ設定ビット

ポート出力モード時の、外部端子 P000、P001、・・・ の出力レベルを設定します。

PDR00:P[7:0]は外部端子 P007、P006、・・・P000
PDR01:P[7:0]は外部端子 P017、P016、・・・P010
PDR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

P[n]	動作説明
0	"0"を出力
1	"1"を出力

リードモディファイ系命令使用時の読出し値は、データ方向レジスタ (DDR) との組み合わせで決まります。

DDR	リードモディファイ系命令による読出し	PDR の読出し値
1	No	PDR の値が読み出せます
1	Yes	PDR の値が読み出せます
0	No	端子の値が読み出せます
0	Yes	PDR の値が読み出せます

PDR13:P5 は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では PDR13:P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.2 データ方向レジスタ 00～19 : DDR00-19 (Data Direction Register 00-19)

データ方向レジスタ 00～19 のビット構成について示します。

端子がポート機能時の際の入出力の方向を設定するレジスタです。また、ペリフェラル入力端子として使用する場合は本ビットを入力に設定します。

■ DDR00-19 : アドレス $0E00_H$、$0E01_H$、・・・ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] P[7:0] (Port) : データ方向選択ビット

ポート出力モード時の、外部端子 P000、P001、・・・の入出力方向を設定します。

DDR00:P[7:0]は外部端子 P007、P006、・・・P000
DDR01:P[7:0]は外部端子 P017、P016、・・・P010
DDR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

P[n]	動作説明
0	入力(初期値)
1	出力

DDR05:P7 は予約ビットです。

書込み、読出しともに動作に影響ありません。

DDR13:P[5]は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では DDR13:P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.3 ポート機能レジスタ 00～19 : PFR00-19 (Port Function Register 00-19)

ポート機能レジスタ 00～19 のビット構成について示します。

端子をポート機能か、それ以外として使用するかを設定します。ペリフェラルの入力端子として使用する場合は、ポート機能側に設定します。

■ PFR00-19 : アドレス 0E20H、0E21H、 ・・・(アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	*	*	*	*	*	*	*	*
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

*各レジスタの初期値については、『付録』の『B. I/O マップ』を参照してください。

[bit7～bit0] P[7:0] (Port) : ポート機能選択ビット

ポート機能を設定します。

PFR00:P[7:0]は外部端子 P007、P006、・・・P000
PFR01:P[7:0]は外部端子 P017、P016、・・・P010
PFR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

P[n]	動作説明
0	ポート機能、またはペリフェラル入力端子(初期値)
1	ペリフェラル入出力(双方向)端子、ペリフェラル出力端子、または外バス端子(EPFR にて設定)

PFR09:P6 には対応するペリフェラル入出力、ペリフェラル出力、外バス端子がありません。
PFR13:P5 は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では PFR13:P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.4 入力データダイレクトレジスタ 00～19 : PDDR00-19 (Port Data Direct Register 00-19)

入力データダイレクトレジスタ 00～19 のビット構成について示します。

外部端子ごとの電圧レベルが常に読み出せるレジスタです。本レジスタでは条件なしで読み出せます。

■ PDDR00-19 : アドレス 0E40H、0E41H、・・・ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	X	X	X	X	X	X	X	X
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit7～bit0] P[7:0] (Port) : 読出しビット

外部端子の値を読み出せます。

PDDR00:P[7:0]は外部端子 P007、P006、・・・P000
PDDR01:P[7:0]は外部端子 P017、P016、・・・P010
PDDR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

P[n]	動作説明
0	Low レベル
1	High レベル

PDDR13:P5 は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では PDDR13:P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.5 プルアップダウン制御レジスタ 00～19 : PPCR00-19 (Port Pull-up/down Control Register 00-19)

プルアップダウン制御レジスタ 00～19 のビット構成について示します。

入力状態時に、ポートごとにプルアップまたはプルダウンを選択するレジスタです。プルアップダウン許可レジスタ(PPER)と組み合わせて設定します。

■ PPCR00-19 : アドレス $0EA0_H$、$0EA1_H$、・・・ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] P[7:0] (Port) : プルアップダウン制御選択ビット

PPCR00:P[7:0]は外部端子 P007、P006、・・・P000
PPCR01:P[7:0]は外部端子 P017、P016、・・・P010
PPCR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

P[n]	動作説明
0	プルダウン
1	プルアップ(初期値)

プルアップ/プルダウンの有無は『概要』の章の『9. 端子機能説明』および『11. 入出力回路形式』を参照してください。

PPCR13:bit5 は予約ビットです。

書込み、読出しともに動作に影響ありません。

クロック 2 系統品種では PPCR13:P[7:6]は、予約ビットです。

書込み、読出しともに動作に影響ありません。

11.4.6 プルアップダウン許可レジスタ 00～19 : PPER00-19 (Port Pull-up/down Enable Register 00-19)

プルアップダウン許可レジスタ 00～19 のビット構成について示します。

入力状態時に、ポートごとにプルアップまたはプルダウンの有無を設定するレジスタです。プルアップダウン制御レジスタ(PPCR)と組み合わせて設定します。

■ PPER00-19 : アドレス $0EC0_H$、$0EC1_H$、・・・ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] P[7:0] (Port) : プルアップダウン許可選択ビット

PPER00:P[7:0]は外部端子 P007、P006、・・・P000
PPER01:P[7:0]は外部端子 P017、P016、・・・P010
PPER02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

P[n]	動作説明
0	プルアップダウンなし(初期値)
1	プルアップダウンあり

プルアップ・プルダウンの有無は『概要』の章の『9. 端子機能説明』および『11. 入出力回路形式』を参照してください。
PPER13:P5 は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では PPER13:P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.7 ポート入力レベル選択レジスタ 00～19 : PILR00-19 (Port Input Level Register 00-19)

ポート入力レベル選択レジスタ 00～19 のビット構成について示します。

ポートごとに入力レベルを設定するレジスタです。 グリッチ入力が起きる場合がありますので、当該端子をペリフェラルの外部入力クロックやトリガとして使用する場合などは、そのペリフェラルをディセーブルにしておいてください。
拡張ポート入力レベル選択レジスタ(EPILR)と組で使用します。

■ PILR00-19 : アドレス $0EE0_H$、$0EE1_H$、・・・(アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] P[7:0] (Port) : ポート入力レベル選択ビット

PILR00:P[7:0]は外部端子 P007、P006、・・・P000
PILR01:P[7:0]は外部端子 P017、P016、・・・P010
PILR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

PILR:P[n]	EPILR:P[n]	入力レベル	備考
0	0	CMOS Schmitt VIL=0.3Vcc VIH=0.7Vcc	
0	1	TTL VIL=0.8[V] VIH=2.0[V]	
1	0	Automotive VIL=0.5Vcc VIH=0.8Vcc	初期値
1	1	CMOS VIL=0.3Vcc VIH=0.7Vcc	

PILR13:P5 は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では PILR13:P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.8 拡張ポート入力レベル選択レジスタ 00～19 : EPILR00-19 (Extended Port Input Level Register 00-19)

拡張ポート入力レベル選択レジスタ 00～19 のビット構成について示します。

ポート入力レベル選択レジスタ(PILR)と組で使用します。「11.4.7. ポート入力レベル選択レジスタ 00～19 : PILR00-19 (Port Input Level Register 00-19)」を参照してください。

■ EPILR00-19 : アドレス $0F00_H$、$0F01_H$、・・・ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] P[7:0] (Port) : 拡張ポート入力レベル選択ビット

EPILR00:P[7:0]は外部端子 P007、P006、・・・P000
EPILR01:P[7:0]は外部端子 P017、P016、・・・P010
EPILR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。設定は PILR の項を参照してください。

EPILR13:P5 は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では EPILR13:P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.9 ポート出力駆動レジスタ 00～19 : PODR00-19 (Port Output Drive Register 00-19)

ポート出力駆動レジスタ 00～19 のビット構成について示します。

ポートごとの駆動力を設定するレジスタです。

■ PODR00-19 : アドレス 0F20H 、0F21H 、...0F2CH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] P[7:0] (Port) : ポート出力駆動選択ビット

PODR00:P[7:0]は外部端子 P007、P006、・・・P000
PODR01:P[7:0]は外部端子 P017、P016、・・・P010
PODR02:P[7:0]は外部端子 P027、P026、・・・P020
(以下同様)
という割り当てです。

11.4.9.1 P010・・・P036 の端子

拡張ポート出力駆動レジスタ(EPODR)との組で設定します。
P010・・・P036

VCCE=5V 時

PODR:P[n]	EPODR:P[n]	動作説明
0	0	1mA
0	1	予約*1
1	0	2mA
1-	1	2mA

*1: サポートしません。設定した場合、駆動能力は変更されません。

(注意事項) VCCE=3.3V 時の駆動能力についてはデータシートを参照してください。

11.4.9.2 P060・・・P087 の端子

ステッピングモータコントローラの出力と兼用されている端子の駆動能力は拡張ポート出力駆動レジスタ(EPODR)設定との組合せにより 1mA/2mA/30mA の設定が可能です。
端子がステッピングモータコントローラの出力端子に設定されている場合、設定に関係なく駆動電流は 30mA となります。

P060・・・P087

PODR:P[n]	EPODR:P[n]	動作説明	備考
0	0	1mA	初期値
0	1	30mA	
1	0	2mA	
1	1	2mA	

11.4.9.3 P010・・・P036, P060・・・P087 以外の端子

(注意事項) CY91F578/9 には適用されない説明です。

P010・・・P036、P060・・・P087 以外

PODR:P[n]	動作説明
0	1mA
1	2mA

【P127, P130, P132, P133 端子のみ】
マルチファンクションシリアルインタフェースを選択しマルチファンクションシリアルインタフェースの動作モードで I2C を選択している場合は、3mA になります。それ以外の場合では上表の設定に従います。

PODR13:P5 は予約ビットです。書込み、読出しともに効果はありません。
クロック 2 系統品種では PODR13.P[7:6]は予約ビットです。書込み、読出しともに効果はありません。

11.4.10 拡張ポート出力駆動レジスタ 01～03,06～08 : EPODR01-03,06-08 (Extended Port Output Drive Register 01-03,06-08)

拡張ポート出力駆動レジスタ 01～03,06～08 のビット構成について示します。

ポートごとの駆動力を設定するレジスタです。PODR との組で使用します。

- **EPODR01 : アドレス 0F35H (アクセス: バイト, ハーフワード, ワード)**
- **EPODR02 : アドレス 0F36H (アクセス: バイト, ハーフワード, ワード)**
- **EPODR03 : アドレス 0F37H (アクセス: バイト, ハーフワード, ワード)**
- **EPODR06 : アドレス 0F38H (アクセス: バイト, ハーフワード, ワード)**
- **EPODR07 : アドレス 0F39H (アクセス: バイト, ハーフワード, ワード)**
- **EPODR08 : アドレス 0F3AH (アクセス: バイト, ハーフワード, ワード)**

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	P[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

設定については「11.4.9. ポート出力駆動レジスタ 00～19 : PODR00-19 (Port Output Drive Register 00-19) 」を参照してください。

EPODR03:P7 は予約ビットです。書込み、読出しともに効果はありません。

11.4.11 拡張ポート機能レジスタ 00～54 : EPFR00-54 (Extended Port Function Register 00-54)

拡張ポート機能レジスタ 00～54 のビット構成について示します。

ペリフェラル／外バス機能の切換え、I/O リロケーション、I/O マルチプレクシングを制御するレジスタです。このレジスタはほかのポートレジスタとは異なり、端子ごとではなく、ペリフェラルごとにイネーブルビットを持ちます。

I/O リロケーションを行うと切換えによるグリッジが発生し信号変化と認識して動作する可能性がありますので、入力の I/O リロケーションは、周辺リソースの入力を無視する状態で行ってください。また、外部割込みフラグは割込みを許可する前にクリアしてください。

端子の周辺リソースへの割当ては PFR、EPFR レジスタで行いますが、すべてのレジスタを同時に変更することができないため、出力の I/O リロケーションはポート設定状態(PFRxx:P[n]=0)で行ってください。

11.4.11.1 拡張ポート機能レジスタ 00, 01,39 : EPFR00, 01, 39 (Extended Port Function Register 00, 01, 39)

拡張ポート機能レジスタ 00, 01, 39 のビット構成について示します。

インプットキャプチャの入力端子の選択を行います。(I/O リロケーション)

■ EPFR00 : アドレス 0E60H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICU3E[1:0]		ICU2E[1:0]		ICU1E[1:0]		ICU0E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR01 : アドレス 0E61H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICU7E[1:0]		ICU6E[1:0]		ICU5E[1:0]		ICU4E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR39 : アドレス 0E87H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICU11E[1:0]		ICU10E[1:0]		ICU9E[1:0]		ICU8E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ICU0E[1:0] : インプットキャプチャ ch.0 入力端子選択
ICU1E[1:0] : インプットキャプチャ ch.1 入力端子選択
ICU2E[1:0] : インプットキャプチャ ch.2 入力端子選択
ICU3E[1:0] : インプットキャプチャ ch.3 入力端子選択
ICU4E[1:0] : インプットキャプチャ ch.4 入力端子選択
ICU5E[1:0] : インプットキャプチャ ch.5 入力端子選択
ICU6E[1:0] : インプットキャプチャ ch.6 入力端子選択
ICU7E[1:0] : インプットキャプチャ ch.7 入力端子選択
ICU8E[1:0] : インプットキャプチャ ch.8 入力端子選択
ICU9E[1:0] : インプットキャプチャ ch.9 入力端子選択
ICU10E[1:0] : インプットキャプチャ ch.10 入力端子選択
ICU11E[1:0] : インプットキャプチャ ch.11 入力端子選択

ICUnE[1:0] (n=0〜11)	動作説明
00	ICUn_0 端子からの入力
01	ICUn_1 端子からの入力
10	ICUn_2 端子からの入力
11	予約 (ICUn_2 端子からの入力)

11.4.11.2 拡張ポート機能レジスタ 02～05 : EPFR02-05 (Extended Port Function Register 02-05)

拡張ポート機能レジスタ 02～05 のビット構成について示します。

リロードタイマの出力の許可および出力、入力端子の選択を行います。(I/O リロケーション・I/O マルチプレクシング)

■ EPFR02 : アドレス 0E62H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			TOT0E[2:0]			TIN0E[1:0]	
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR03 : アドレス 0E63H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			TOT1E[2:0]			TIN1E[1:0]	
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR04 : アドレス 0E64H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			TOT2E[2:0]			TIN2E[1:0]	
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR05 : アドレス 0E65H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			TOT3E[2:0]			TIN3E[1:0]	
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

TOT0E[2:0] : リロードタイマ ch.0 TOT 出力端子選択
TIN0E[1:0] : リロードタイマ ch.0 TIN 入力端子選択
TOT1E[2:0] : リロードタイマ ch.1 TOT 出力端子選択
TIN1E[1:0] : リロードタイマ ch.1 TIN 入力端子選択
TOT2E[2:0] : リロードタイマ ch.2 TOT 出力端子選択
TIN2E[1:0] : リロードタイマ ch.2 TIN 入力端子選択
TOT3E[2:0] : リロードタイマ ch.3 TOT 出力端子選択
TIN3E[1:0] : リロードタイマ ch.3 TIN 入力端子選択

TOTnE[2:0]　(n=0～3)	動作説明
000	出力しない
xx1	TOTn_0 端子からの出力
x1x	TOTn_1 端子からの出力
1xx	TOTn_2 端子からの出力

TINnE[1:0]　(n=0～3)	動作説明
00	TINn_0 端子からの入力
01	TINn_1 端子からの入力
10	TINn_2 端子からの入力
11	予約 (TINn_2 端子からの入力)

11.4.11.3 拡張ポート機能レジスタ 06～09,33,34 : EPFR06-09,33,34 (Extended Port Function Register 06-09,33,34)

拡張ポート機能レジスタ 06～09,33,34 のビット構成について示します。

LIN-UART の出力の許可および出力、入力端子の選択を行います。(I/O リロケーション・I/O マルチプレクシング)

＜注意事項＞

LIN-UART の SCK/ SOT/ SIN は必ず同一グループ(SCKn/ SOTn/ SINn または SCKn_1/ SOTn_1/ SINn_1)になるように設定してください。下記例のようにバラバラにリロケーションすることは禁止です。
禁止例: SCKn_0/ SOTn_1/ SINn_0

■ EPFR06 ：アドレス 0E66H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT2E[1:0]		SCK2E[1:0]		SIN2E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR07 ：アドレス 0E67H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT3E[1:0]		SCK3E[1:0]		SIN3E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR08 ：アドレス 0E68H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT4E[1:0]		SCK4E[1:0]		SIN4E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR09 ：アドレス 0E69H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT5E[1:0]		SCK5E[1:0]		SIN5E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR33 ：アドレス 0E81H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT6E[1:0]		SCK6E[1:0]		SIN6E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR34 ：アドレス 0E82H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT7E[1:0]		SCK7E[1:0]		SIN7E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

SOT2E[1:0] : LIN-UART ch.2 SOT 出力端子選択
SCK2E[1:0] : LIN-UART ch.2 SCK 出力・入力端子選択
SIN2E : LIN-UART ch.2 SIN 入力端子選択
SOT3E[1:0] : LIN-UART ch.3 SOT 出力端子選択
SCK3E[1:0] : LIN-UART ch.3 SCK 出力・入力端子選択
SIN3E : LIN-UART ch.3 SIN 入力端子選択
SOT4E[1:0] : LIN-UART ch.4 SOT 出力端子選択
SCK4E[1:0] : LIN-UART ch.4 SCK 出力・入力端子選択
SIN4E : LIN-UART ch.4 SIN 入力端子選択
SOT5E[1:0] : LIN-UART ch.5 SOT 出力端子選択
SCK5E[1:0] : LIN-UART ch.5 SCK 出力・入力端子選択
SIN5E : LIN-UART ch.5 SIN 入力端子選択
SOT6E[1:0] : LIN-UART ch.6 SOT 出力端子選択
SCK6E[1:0] : LIN-UART ch.6 SCK 出力・入力端子選択
SIN6E : LIN-UART ch.6 SIN 入力端子選択
SOT7E[1:0] : LIN-UART ch.7 SOT 出力端子選択
SCK7E[1:0] : LIN-UART ch.7 SCK 出力・入力端子選択
SIN7E : LIN-UART ch.7 SIN 入力端子選択

SOTnE[1:0]　(n=2～7)	動作説明
00	出力しない
x1	SOTn_0 端子からの出力
1x	SOTn_1 端子からの出力

SCKnE[1:0]　(n=2～7)	動作説明
00	SCKn_0 からの入力・出力しない
01	SCKn_0 からの入力・SCKn_0 からの出力
10	SCKn_1 からの入力・SCKn_1 からの出力
11	予約 (SCKn_1 からの入力・SCKn_1 からの出力)

SINnE　(n=2～7)	動作説明
0	SINn_0 端子からの入力
1	SINn_1 端子からの入力

11.4.11.4 拡張ポート機能レジスタ 10～15,45,46：EPFR10-15,45,46 (Extended Port Function Register 10-15,45,46)

拡張ポート機能レジスタ 10～15,45,46 のビット構成について示します。

PPG の出力の許可および出力端子の選択を行います。(I/O リロケーション・I/O マルチプレクシング)

■ EPFR10： アドレス 0E6AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		PPG1E[3:0]			PPG0E[2:0]		
初期値	1	0	0	0	0	0	0	0
属性	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR11： アドレス 0E6BH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		PPG3E[2:0]			PPG2E[2:0]		
初期値	1	1	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR12： アドレス 0E6CH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		PPG5E[2:0]			PPG4E[2:0]		
初期値	1	1	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR13： アドレス 0E6DH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		PPG7E[2:0]			PPG6E[2:0]		
初期値	1	1	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR14： アドレス 0E6EH (アクセス: バイト, ハーフワード, ワード)

bit	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		PPG9E[2:0]			PPG8E[2:0]		
初期値	1	1	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR15： アドレス 0E6F$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	PPG12E[1:0]		PPG11E[1:0]		PPG10E[2:0]		
初期値	1	0	0	0	0	0	0	0
属性	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR45： アドレス 0E8D$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PPG17E	PPG16E	PPG15E[1:0]		PPG14E[1:0]		PPG13E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR46： アドレス 0E8E$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		PPG23E	PPG22E	PPG21E	PPG20E	PPG19E	PPG18E
初期値	1	1	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

PPG0E[2:0] : PPG ch.0 出力端子選択
PPG2E[2:0] : PPG ch.2 出力端子選択
PPG3E[2:0] : PPG ch.3 出力端子選択
PPG4E[2:0] : PPG ch.4 出力端子選択
PPG5E[2:0] : PPG ch.5 出力端子選択
PPG6E[2:0] : PPG ch.6 出力端子選択
PPG7E[2:0] : PPG ch.7 出力端子選択
PPG8E[2:0] : PPG ch.8 出力端子選択
PPG9E[2:0] : PPG ch.9 出力端子選択
PPG10E[2:0] : PPG ch.10 出力端子選択

PPGnE[2:0] (n=0,2～10)	動作説明
000	出力しない
xx1	PPGn_0 端子からの出力
x1x	PPGn_1 端子からの出力
1xx	PPGn_2 端子からの出力

PPG1E[3:0]　PPG ch.1 出力端子選択

PPGnE[3:0]　(n=1)	動作説明
0000	出力しない
xxx1	PPGn_0 端子からの出力
xx1x	PPGn_1 端子からの出力
x1xx	PPGn_2 端子からの出力
1xxx	PPGn_3 端子からの出力

PPG11E[1:0] : PPG ch.11 出力端子選択

PPG12E[1:0] : PPG ch.12 出力端子選択

PPG13E[1:0] : PPG ch.13 出力端子選択

PPG14E[1:0] : PPG ch.14 出力端子選択

PPG15E[1:0] : PPG ch.15 出力端子選択

PPGnE[1:0]　(n=11～15)	動作説明
00	出力しない
01	PPGn_0 端子からの出力
1x	PPGn_1 端子からの出力

PPG16E : PPG ch.16 出力端子選択

PPG17E : PPG ch.17 出力端子選択

PPG18E : PPG ch.18 出力端子選択

PPG19E : PPG ch.19 出力端子選択

PPG20E : PPG ch.20 出力端子選択

PPG21E : PPG ch.21 出力端子選択

PPG22E : PPG ch.22 出力端子選択

PPG23E : PPG ch.23 出力端子選択

PPGnE　(n=16～23)	動作説明
0	出力しない
1	PPGn_0 端子からの出力

11.4.11.5 拡張ポート機能レジスタ 16～20 : EPFR16-20 (Extended Port Function Register 16-20)

拡張ポート機能レジスタ 16～20 のビット構成について示します。

LCD コントローラのセグメント、コモン出力の許可、基準電源電圧入力の許可を行います。(I/O マルチプレクシング)

■ EPFR16 : アドレス 0E70H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SEG7E	SEG6E	SEG5E	SEG4E	SEG3E	SEG2E	SEG1E	SEG0E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR17 : アドレス 0E71H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SEG15E	SEG14E	SEG13E	SEG12E	SEG11E	SEG10E	SEG9E	SEG8E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR18 : アドレス 0E72H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SEG23E	SEG22E	SEG21E	SEG20E	SEG19E	SEG18E	SEG17E	SEG16E
初期値	1	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR19 : アドレス 0E73H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SEG31E	SEG30E	SEG29E	SEG28E	SEG27E	SEG26E	SEG25E	SEG24E
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR20 : アドレス 0E74H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	V3E	V2E	V1E	V0E	COM3E	COM2E	COM1E	COM0E
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SEGnE (n=0～31) : LCD コントローラデューティ駆動ドライバセグメント n 出力イネーブル

COMnE (n=0～3) : LCD コントローラドライバコモン n 出力イネーブル

VnE (n=0～2) : LCD コントローラ基準電源入力 n イネーブル
V3E : LCD コントローラ基準電源入力選択

SEGnE (n=0～31)	動作説明
0	LCDC セグメント出力禁止
1	LCDC セグメント出力許可

COMnE (n=0~2）も、SEGnE と同様です。

VnE (n=0～2)	動作説明
0	LCDC 基準電源入力禁止
1	LCDC 基準電源入力許可

V3 選択の場合、V3E と PFR05.bit7 を組で設定してください。

{V3E, PFR05.bit7}	動作説明
{1,1}	V3 を LCDC 基準電源 V3 入力として使用
上記以外	設定禁止 ただし CY91F575/577 の場合に限り、LCDC 基準電源 V3 は VCC を使用 その場合、P057/RDY/FRCK3_1 入力として使用可能

LCDC 基準電源 V3 入力として外部 V3 と VCC の切り替えを行う際にはメイン発振状態で切り替えを行ってください。

11.4.11.6 拡張ポート機能レジスタ 21～23 : EPFR21-23 (Extended Port Function Register 21-23)

拡張ポート機能レジスタ 21～23 のビット構成について示します。

ステッピングモータコントローラの出力の許可を行います。(I/O マルチプレクシング)

■ EPFR21 : アドレス 0E75H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PWM2M1E	PWM2P1E	PWM1M1E	PWM1P1E	PWM2M0E	PWM2P0E	PWM1M0E	PWM1P0E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR22 : アドレス 0E76H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PWM2M3E	PWM2P3E	PWM1M3E	PWM1P3E	PWM2M2E	PWM2P2E	PWM1M2E	PWM1P2E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR23 : アドレス 0E77H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PWM2M5E	PWM2P5E	PWM1M5E	PWM1P5E	PWM2M4E	PWM2P4E	PWM1M4E	PWM1P4E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

PWM2MnE (n=0～5) : SMC チャネル n PWM 制御 (M2) 出力イネーブル

PWM2PnE (n=0～5) : SMC チャネル n PWM 制御 (P2) 出力イネーブル

PWM1MnE (n=0～5) : SMC チャネル n PWM 制御 (M1) 出力イネーブル

PWM1PnE (n=0～5) : SMC チャネル n PWM 制御 (P1) 出力イネーブル

PWM2MnE (n=0～5)	動作説明
0	SMC チャネル n PWM M2 出力禁止(初期値)
1	SMC チャネル n PWM M2 出力許可

PWM2PnE、PWM1MnE、PWM1PnE (n=0～5) も、PWM2MnE と同様です。

11.4.11.7 拡張ポート機能レジスタ 24 : EPFR24 (Extended Port Function Register 24)

拡張ポート機能レジスタ 24 のビット構成について示します。

CAN の出力の許可を行います。(I/O マルチプレクシング)

■ EPFR24 : アドレス 0E78H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					TX2E	TX1E	TX0E
初期値	1	1	1	1	1	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W

TXnE (n=0～2) : CAN チャネル n 送信データ出力イネーブル

TXnE (n=0～2)	動作説明
0	CAN チャネル n 出力禁止(初期値)
1	CAN チャネル n 出力許可

11.4.11.8 拡張ポート機能レジスタ 25 : EPFR25 (Extended Port Function Register 25)

拡張ポート機能レジスタ 25 のビット構成について示します。

予備レジスタです。

■ EPFR25 : アドレス 0E79H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					EPFR25D[2:0]		
初期値	1	1	1	1	1	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W0	R/W0	R/W0

EPFR25D[2:0] : 予備ビット

本ビットには"0"を書き込んでください。

11.4.11.9 拡張ポート機能レジスタ 26 : EPFR26 (Extended Port Function Register 26)

拡張ポート機能レジスタ 26 のビット構成について示します。

ベースタイマの出力の許可を行います。(I/O マルチプレクシング)

■ EPFR26 : アドレス 0E7AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				TIB1E	TIB0E	TIA1E	TIA0E
初期値	1	1	1	1	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W

TIBnE (n=0, 1)　予備ビット

本ビットを設定しても効果はありません。

TIAnE (n=0, 1)　ベースタイマ TIOAn 出力イネーブル

TIAnE (n=0, 1)	動作説明
0	ベースタイマ TIOAn 出力禁止(初期値)
1	ベースタイマ TIOAn 出力許可

11.4.11.10 拡張ポート機能レジスタ 27,30 : EPFR27,30 (Extended Port Function Register 27,30)

拡張ポート機能レジスタ 27,30 のビット構成について示します。

リアルタイムクロック、サウンドジェネレータの出力の許可を行います。(I/O マルチプレクシング、I/O リロケーション)

■ EPFR27 : アドレス 0E7B$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			WOTE	SGO1E	SGA1E	SGO0E	SGA0E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR30 : アドレス 0E7E$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SGO4E[1:0]		SGA4E[1:0]		SGO3E	SGA3E	SGO2E	SGA2E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SGAnE (n=0～3) : サウンドジェネレータ チャネル n SGA 出力イネーブル

SGOnE (n=0～3) : サウンドジェネレータ チャネル n SGO 出力イネーブル

WOTE :リアルタイムクロック オーバフロー出力イネーブル

SGAnE (n=0～3)	動作説明
0	サウンドジェネレータ チャネル n SGA 出力禁止 (初期値)
1	サウンドジェネレータ チャネル n SGA 出力許可

SGOnE (n=0～3)、WOTE も同様です。

SGA4E [1:0] : サウンドジェネレータ チャネル 4 SGA 出力イネーブル

SGO4E [1:0] : サウンドジェネレータ チャネル 4 SGO 出力イネーブル

SGA4E[1:0]	動作説明
00	出力しない
01	SGA4_0 端子からの出力
10	SGA4_1 端子からの出力
11	設定禁止

SGO4E[1:0]も同様です。

11.4.11.11 拡張ポート機能レジスタ 28 : EPFR28 (Extended Port Function Register 28)

拡張ポート機能レジスタ 28 のビット構成について示します。

フリーランタイマのクロック入力の許可を行います。(I/O マルチプレクシング)

■ EPFR28 : アドレス 0E7CH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				FRCK3E	FRCK2E	FRCK1E	FRCK0E
初期値	1	1	1	1	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W

[bit0] FRCK0E : フリーランタイマ ch.0 クロック入力選択

[bit1] FRCK1E : フリーランタイマ ch.1 クロック入力選択

[bit2] FRCK2E : フリーランタイマ ch.2 クロック入力選択

[bit3] FRCK3E : フリーランタイマ ch.3 クロック入力選択

FRCKnE (n=0～3)	動作説明
0	FRCKn_0 からの入力 (初期値)
1	FRCKn_1 からの入力

11.4.11.12 拡張ポート機能レジスタ 29,31,32 : EPFR29,31,32 (Extended Port Function Register 29,31,32)

拡張ポート機能レジスタ 29,31,32 のビット構成について示します。

アウトプットコンペアの出力の許可を行います。(I/O マルチプレクシング・I/O リロケーション)

■ EPFR29 : アドレス 0E7DH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	OCU3E[1:0]		OCU2E[1:0]		OCU1E[1:0]		OCU0E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR31 : アドレス 0E7FH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	OCU7E[1:0]		OCU6E[1:0]		OCU5E[1:0]		OCU4E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR32 : アドレス 0E80H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	OCU11E[1:0]		OCU10E[1:0]		OCU9E[1:0]		OCU8E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

OCUnE[1:0] (n=0～11) : アウトプットコンペア チャネル n 出力イネーブル

OCUnE[1:0] (n=0～11)	動作説明
00	出力しない
01	OCUn_0 から出力
10	OCUn_1 から出力
11	設定禁止

11.4.11.13 拡張ポート機能レジスタ 35～38 : EPFR35-38 (Extended Port Function Register 35-38)

拡張ポート機能レジスタ 35～38 のビット構成について示します。

マルチファンクションシリアルインタフェースの出力の許可を行います。(I/O マルチプレクシング・I/O リロケーション)

＜注意事項＞

マルチファンクションシリアルの SCK/ SOT/ SIN は必ず同一グループ(SCKn/ SOTn/ SINn または SCKn_1/SOTn_1/ SINn_1)になるように設定してください。下記例のようにバラバラにリロケーションすることは禁止です。

禁止例: SCKn_0/ SOTn_1/ SINn_0

■ EPFR35 : アドレス 0E83H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT0E[1:0]		SCK0E[1:0]		SIN0E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR36 : アドレス 0E84H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT1E[1:0]		SCK1E[1:0]		SIN1E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR37 : アドレス 0E85H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SOT8E[2:0]			SCK8E[2:0]			SIN8E[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR38 : アドレス 0E86H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			SOT9E[1:0]		SCK9E[1:0]		SIN9E
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

SOT0E[1:0] : マルチファンクションシリアルインタフェース ch.0 SOT 出力・入力端子選択
SCK0E[1:0] : マルチファンクションシリアルインタフェース ch.0 SCK 出力・入力端子選択
SIN0E :マルチファンクションシリアルインタフェース ch.0 SIN 入力端子選択
SOT1E[1:0] : マルチファンクションシリアルインタフェース ch.1 SOT 出力・入力端子選択
SCK1E[1:0] : マルチファンクションシリアルインタフェース ch.1 SCK 出力・入力端子選択
SIN1E : マルチファンクションシリアルインタフェース ch.1 SIN 入力端子選択
SOT8E[2:0] : マルチファンクションシリアルインタフェース ch.8 SOT 出力・入力端子選択
SCK8E[2:0] : マルチファンクションシリアルインタフェース ch.8 SCK 出力・入力端子選択
SIN8E[1:0] : マルチファンクションシリアルインタフェース ch.8 SIN 入力端子選択
SOT9E[1:0] : マルチファンクションシリアルインタフェース ch.9 SOT 出力・入力端子選択
SCK9E[1:0] : マルチファンクションシリアルインタフェース ch.9 SCK 出力・入力端子選択
SIN9E : マルチファンクションシリアルインタフェース ch.9 SIN 入力端子選択

ch.0, ch.1 の I²C のリロケーションは対応していません。ch.0, ch.1 で I²C 選択時は _0 のリロケーションが選択されるようにレジスタを設定してください。ch.8,ch.9 は I²C 入出力に対応していません。

SCK0,1,9 端子選択(n はチャネル番号)

SCK0E1 SCK1E1 SCK9E1	SCK0E0 SCK1E0 SCK9E0	動作説明
0	0	SCKn_0 からの入力 ／ 出力しない
0	1	SCKn_0 からの入力 ／ SCKn_0 からの出力
1	0	SCKn_1 からの入力 ／ SCKn_1 からの出力
1	1	設定禁止

ch.0, ch.1 の I²C のリロケーションは対応していません。 ch.0, ch.1 で I²C 選択時は 01 を設定してください。

SOT0,1,9 端子選択(n はチャネル番号)

SOT0E1 SOT1E1 SOT9E1	SOT0E0 SOT1E0 SOT9E0	動作説明
0	0	SOTn_0 からの入力 ／ 出力しない
0	1	SOTn_0 からの入力 ／ SOTn_0 からの出力
1	0	SOTn_1 からの入力 ／ SOTn_1 からの出力
1	1	設定禁止

ch.0, ch.1 の I²C のリロケーションは対応していません。 ch.0, ch.1 で I²C 選択時は 01 を設定してください。

SIN0,1,9 端子選択 (n はチャネル番号)

SIN0E SIN1E SIN9E	動作説明
0	SINn_0 からの入力
1	SINn_1 からの入力

ch.0, ch.1 の I²C のリロケーションは対応していません。ch.0, ch.1 で I²C 選択時は 0 を設定してください。

SCK8 端子選択

SCK8E2	SCK8E1	SCK8E0	動作説明
0	0	0	SCK8_0 からの入力　／　出力しない
0	0	1	SCK8_0 からの入力　／　SCK8_0 からの出力
0	1	0	SCK8_1 からの入力　／　SCK8_1 からの出力
1	0	0	SCK8_2 からの入力　／　SCK8_2 からの出力
上記以外			設定禁止

SOT8 端子選択

SOT8E2	SOT8E1	SOT8E0	動作説明
0	0	0	SOT8_0 からの入力　／　出力しない
0	0	1	SOT8_0 からの入力　／　SOT8_0 からの出力
0	1	0	SOT8_1 からの入力　／　SOT8_1 からの出力
1	0	0	SOT8_2 からの入力　／　SOT8_2 からの出力
上記以外			設定禁止

SIN8 端子選択

SIN8E1	SIN8E0	動作説明
0	0	SIN8_0 からの入力
0	1	SIN8_1 からの入力
1	0	SIN8_2 からの入力
1	1	設定禁止

11.4.11.14 拡張ポート機能レジスタ 40, 41 : EPFR40, 41 (Extended Port Function Register 40, 41)

拡張ポート機能レジスタ 40, 41 のビット構成について示します。

アップンダウンカウンタの端子を選択します。(I/O リロケーション)

■ EPFR40： アドレス 0E88H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		ZIN0E[1:0]		BIN0E[1:0]		AIN0E[1:0]	
初期値	1	1	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR41： アドレス 0E89H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					ZIN1E	BIN1E	AIN1E
初期値	1	1	1	1	1	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W

AIN0E[1:0]	動作説明
00	AIN0_0 からの入力 (初期値)
01	AIN0_1 からの入力
10	AIN0_2 からの入力
11	設定禁止

BIN0E, ZIN0E も同様です。

AIN1E	動作説明
0	AIN1_0 からの入力 (初期値)
1	AIN1_1 からの入力

BIN1E, ZIN1E も同様です。

11.4.11.15 拡張ポート機能レジスタ 42 : EPFR42 (Extended Port Function Register 42)

拡張ポート機能レジスタ 42 のビット構成について示します。

DA コンバータの出力信号の選択を行います。(I/O マルチプレクシング)

■ EPFR42 : アドレス 0E8AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						DAS1	DAS0
初期値	1	1	1	1	1	1	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W

[bit0] DAS0 : DA コンバータ 0 の出力データ選択イネーブル
[bit1] DAS1 : DA コンバータ 1 の出力データ選択イネーブル

DASn (n=0,1)	動作説明
0	DAOn 出力禁止(初期値)
1	DAOn 出力許可

11.4.11.16 拡張ポート機能レジスタ 43,44 : EPFR43,44 (Extended Port Function Register 43,44)

拡張ポート機能レジスタ 43～44 のビット構成について示します。

外部割込みの端子を選択します。(I/O リロケーション)

■ EPFR43 : アドレス 0E8BH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	INT7E	INT6E	INT5E	INT4E	INT3E	INT2E	INT1E	INT0E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ EPFR44 : アドレス 0E8CH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	INT15E	INT14E	INT13E	INT12E	INT11E	INT10E	INT9E	INT8E
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

INTnE (n=0～15)	動作説明
0	INTn_0 からの入力 (初期値)
1	INTn_1 からの入力

11.4.11.17 拡張ポート機能レジスタ 47 : EPFR47 (Extended Port Function Register47)

拡張ポート機能レジスタ 47 のビット構成について示します。

外部バスインタフェースのデータ端子の上位バイトと下位バイトをスワップ(決まった組合せの端子間でデータ端子の機能が入れ替わり)します。

■ EPFR47 : アドレス 0E8F$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							EBDSWPE
初期値	1	1	1	1	1	1	1	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W

[bit0] EBDSWPE : スワップ許可

EBDSWPE	動作説明*	
0	ノーマルコネクション 端子番号 131/183 D16 端子番号 132/184 D17 ~ 端子番号 137/189 D22 端子番号 138/190 D23	 端子番号 139/191 D24 端子番号 140/192 D25 ~ 端子番号 3/197 D30 端子番号 4/198 D31
1	スワップコネクション 端子番号 131/183 D24 端子番号 132/184 D25 ~ 端子番号 137/189 D30 端子番号 138/190 D31	 端子番号 139/191 D16 端子番号 140/192 D17 ~ 端子番号 3/197 D22 端子番号 4/198 D23

*: 端子番号は LQFP-144/LQFP-208 をそれぞれスラッシュ"/"で区切って記載しています。

11.4.11.18 拡張ポート機能レジスタ 48～51 : EPFR48-51 (Extended Port Function Register 48-51)

拡張ポート機能レジスタ 48～51 のビット構成について示します。

予備レジスタです。これらのレジスタは使用しないでください。

■ **EPFR48 : アドレス 0E90H (アクセス: バイト, ハーフワード, ワード)**

■ **EPFR49 : アドレス 0E91H (アクセス: バイト, ハーフワード, ワード)**

■ **EPFR50 : アドレス 0E92H (アクセス: バイト, ハーフワード, ワード)**

■ **EPFR51 : アドレス 0E93H (アクセス: バイト, ハーフワード, ワード)**

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							
初期値	1	1	1	1	1	1	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX

11.4.11.19 拡張ポート機能レジスタ 52 : EPFR52 (Extended Port Function Register52)

拡張ポート機能レジスタ 52 のビット構成について示します。

HS_SPI　SDATA[1:0]と SPI_SIO0,1 の接続を設定します。

■ **EPFR52 : アドレス 0E94H (アクセス: バイト, ハーフワード, ワード)**

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							HSSWAP
初期値	1	1	1	1	1	1	1	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W

[bit0] HSSWAP : HS_SPI D0,D1 スワップ許可

HSSWAP	動作説明
0	ノーマルコネクション SPI_SIO0 入力 → SDATA[0]入力 SPI_SIO1 入力 → SDATA[1]入力 SDATA[0]出力 → SPI_SIO0 出力 SDATA[1]出力 → SPI_SIO1 出力
1	スワップコネクション SPI_SIO0 入力 → SDATA[1]入力 SPI_SIO1 入力 → SDATA[0]入力 SDATA[0]出力 → SPI_SIO1 出力 SDATA[1]出力 → SPI_SIO0 出力

11.4.11.20 拡張ポート機能レジスタ 53,54 : EPFR53,54 (Extended Port Function Register53,54)

拡張ポート機能レジスタ 53,54 のビット構成について示します。

HS_SPI の出力の許可を行います。(I/O マルチプレクシング)。

■ EPFR53 : アドレス 0E95H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			HSCKE	HSDE[3:0]			
初期値	1	1	1	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W

■ EPFR54 : アドレス 0E96H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				HSSELE[3:0]			
初期値	1	1	1	1	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W

[bit4] HSCKE : HS_SPI SCK 出力許可

HSCKE	動作説明
0	出力しない
1	出力許可

[bit3～bit0] HSDE[3:0] : HS_SPI SDATAn (n=0～3) 出力許可

HSDE[n] (n=0～3)	動作説明
0	出力しない
1	出力許可

[bit3～bit0] HSSELE[3:0] : HS_SPI SSELn (n=0～3) 出力許可

HSSELE[n] (n=0～3)	動作説明
0	出力しない
1	出力許可

HS_SPI をスレーブモードで使用する場合は、HSCKE および HSSELE を 0（出力しない）に設定して、11.5.1.2 ペリフェラル入力の割り当て に従い端子をペリフェラル入力に設定してください。

11.4.12 ポート入力許可レジスタ: PORTEN (PORT ENable register)

ポート入力許可レジスタのビット構成について示します。

ポートの入力遮断の解除を設定するレジスタです。ポートがソフトウェアにより設定される前に貫通電流の変動を避けるために、パワーオンリセット時に大部分の端子は入力遮断になります。入力遮断になる端子については、『付録』の『D. CPU 状態における端子状態』を参照してください。 各ポート端子をその機能に応じて設定した後、グローバルポート許可(PORTEN:GPORTEN) でポート入力を許可する必要があります。

■ PORTEN : アドレス 0F40H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							GPORTEN
初期値	1	1	1	1	1	1	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W

[bit0] GPORTEN (Global PORT ENable) : グローバル入力遮断解除

GPORTEN	動作説明
0	大半の端子を入力遮断にします。入力遮断になる端子については、『付録』の『D. CPU 状態における端子状態』を参照してください。(初期値)
1	本ビットによる入力遮断を解除します。

11.5 動作説明

I/O ポートの動作について説明します。

11.5.1. 端子の入出力の割り当て

11.5.2. EPFR 設定の優先度について

11.5.3. 入力 I/O リロケーション設定時の注意

11.5.4. GPORTEN による入力遮断

11.5.5. A/D コンバータ機能のある端子の注意事項

11.5.6. ベースタイマ TIOA1 端子使用時の設定

11.5.7. 電源遮断からのウェイクアップ時の動作

11.5.8. I/O ポート機能切り替え時の注意事項

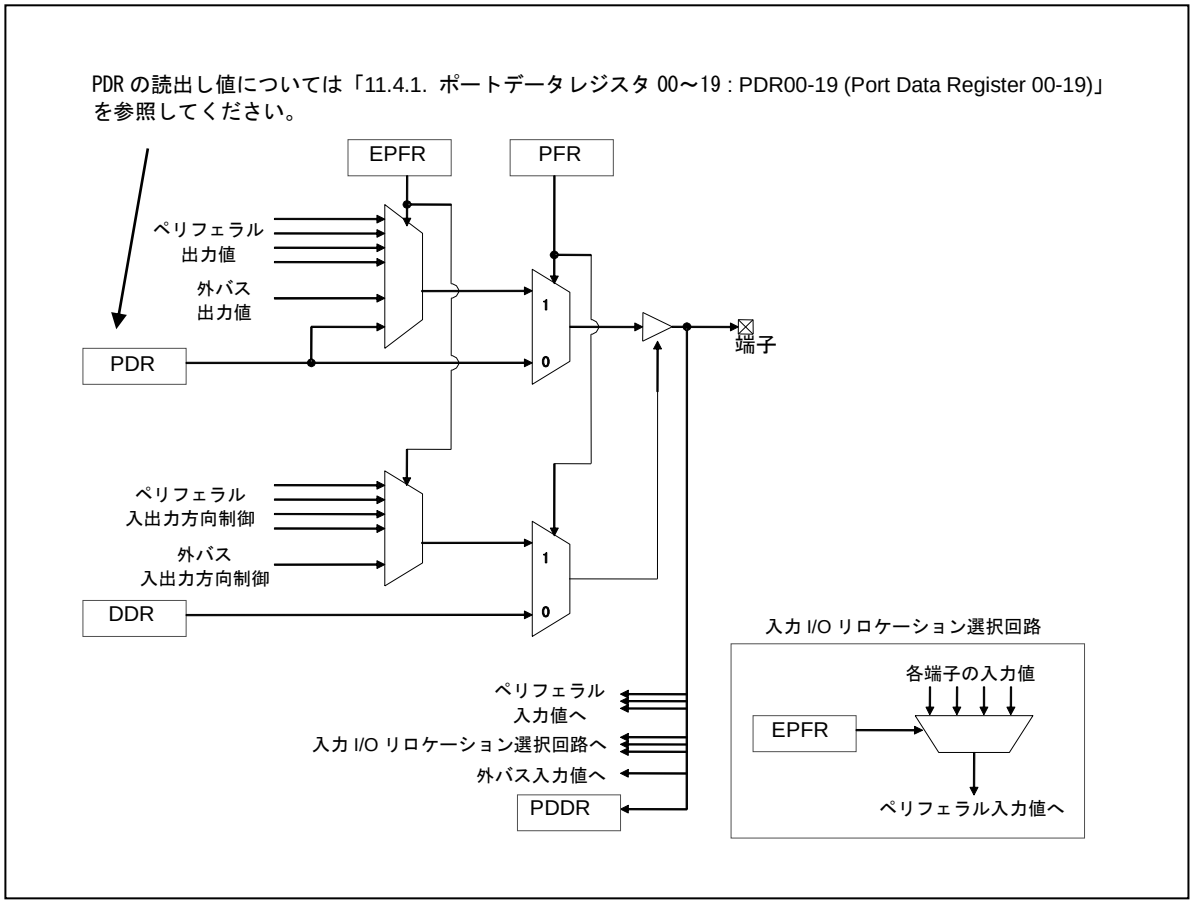
11.5.9. 特定リソース使用時の入力遮断

11.5.1 端子の入出力の割り当て

端子の入出力の割り当てについて説明します。

端子の入出力の割り当てについて説明します。各端子の入出力方向は下図に示す構成で制御されています。

図 11-2 端子入出力方向，出力値選択，入力値取り込み構成図



端子割り当ての説明各項で説明するように､最初に PFR をポート機能側に変更してください｡その際端子はポート機能になりますので、必要な場合は前もって DDR および PDR 値を設定しておいてください。その際、いったん、端子の入出力方向が DDR で指定される方向になりますので注意してください。また、A/D コンバータ機能のある端子では、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『A/D コンバータ』の章を参照してください。

11.5.1.1 ペリフェラル入出力 (双方向) 端子の割り当て

ペリフェラル入出力(双方向)端子の割り当てについて説明します。

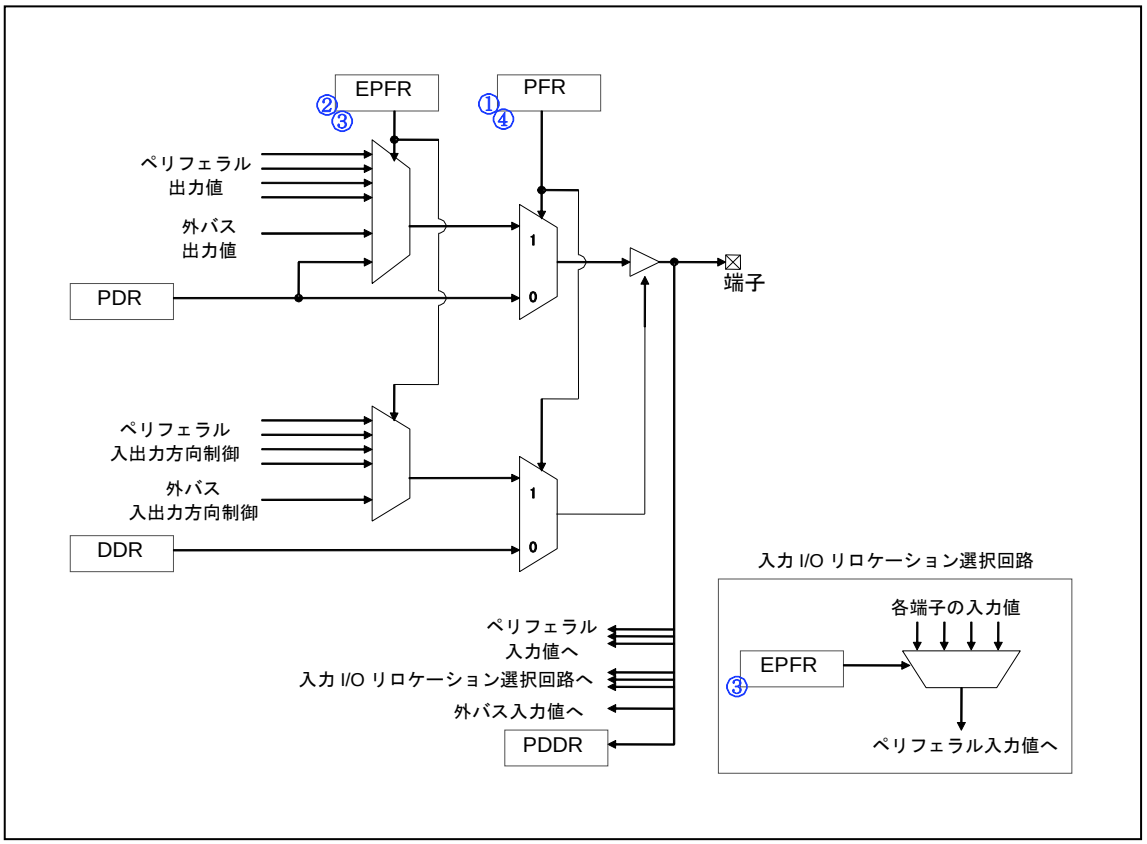
(準備)

①でいったん端子はポート機能になりますので、必要な場合は前もって DDR および PDR 値を設定しておいてください。

A/D コンバータ機能のある端子では、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『A/D コンバータ』の章を参照してください。

①当該端子の PFR をポート機能側に設定します。
②当該端子を使用するほかのすべてのペリフェラルの EPFR をディセーブルにします。
③当該端子が外バス兼用の場合や、当該ペリフェラルが I/O マルチプレクシング対象の場合、当該ペリフェラルの EPFR を設定します。また、当該ペリフェラルに I/O リロケーションがある場合、当該ペリフェラルの EPFR で使用する端子を設定します。
④PFR をペリフェラル側に設定します。

図 11-3 ペリフェラル入出力の割り当て手順図



11.5.1.2 ペリフェラル入力の割り当て

ペリフェラル入力の割り当てについて説明します。

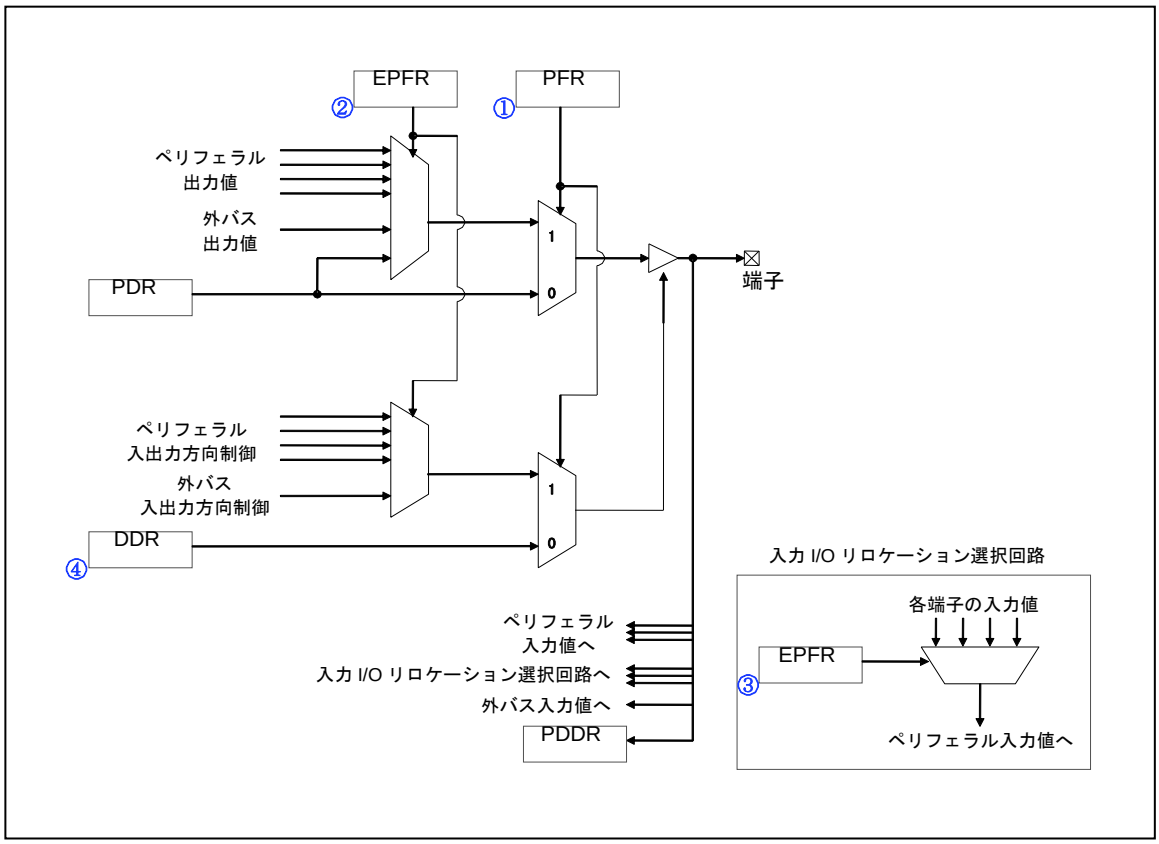
(準備)

①でいったん端子はポート機能になりますので、必要な場合は前もって DDR および PDR 値を設定しておいてください。

A/D コンバータ機能のある端子では、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『CHAPTER 44: A/D コンバータ』の章を参照してください。

①当該端子の PFR をポート機能側に設定します。
②当該端子を使用するほかのすべてのペリフェラルの EPFR をディセーブルにします。
③当該ペリフェラルに I/O リロケーションがある場合、当該ペリフェラルの EPFR で使用する端子を設定します。
④DDR を入力側に設定します。

図 11-4 ペリフェラル入力の割り当て手順図



＜注意事項＞

上図で示しているように、端子がペリフェラル出力などに設定されている場合は、端子を共有するほかのペリフェラル入力へはその出力値が取り込まれます。
(例) INT10_0 と PPG2_2 は同じ端子に配置されていますので(端子番号 104・P111 端子)端子を PPG2_2 のペリフェラル出力、外部割込み 10 のリロケーションを"0"に設定すれば、PPG2 の出力で外部割込み 10 を発生させることができます。

11.5.1.3 ペリフェラル出力の割り当て

ペリフェラル出力の割り当てについて説明します。

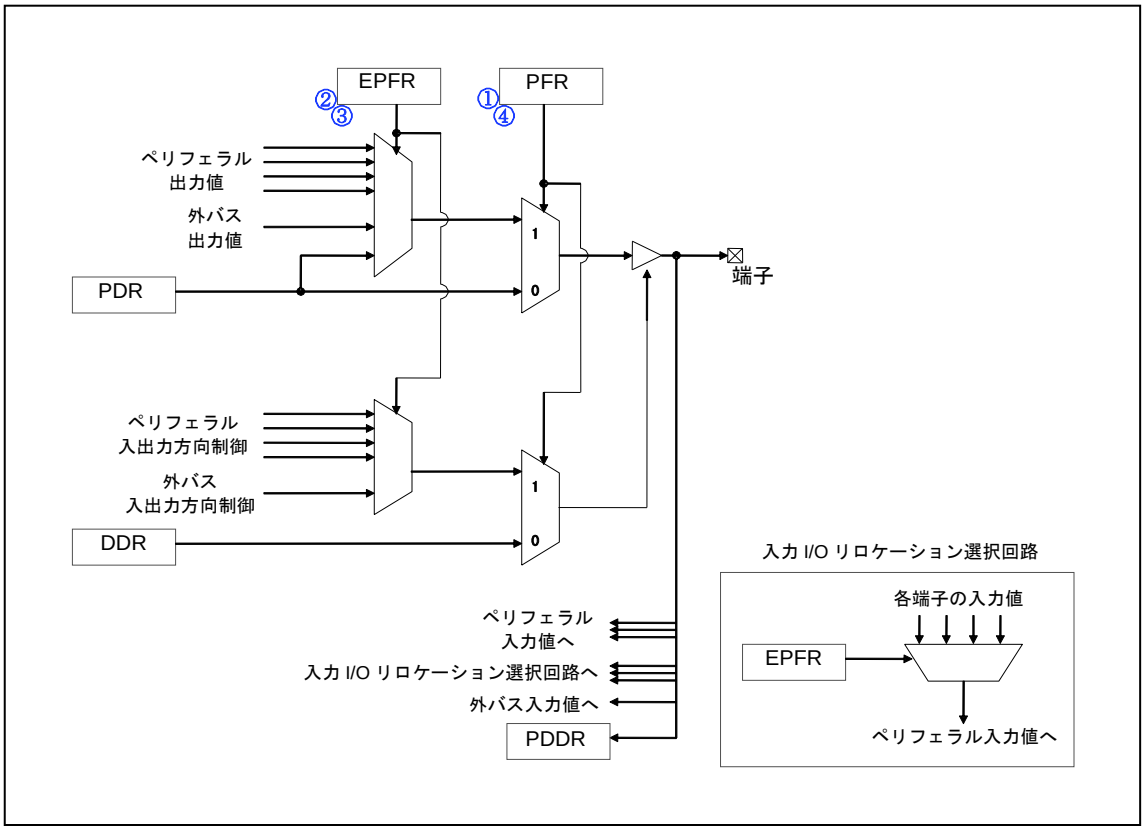
設定方法は「11.5.1.1. ペリフェラル入出力（双方向）端子の割り当て」と同一です。

(準備)

①でいったん端子はポート機能になりますので、必要な場合は前もって DDR および PDR 値を設定しておいてください。

A/D コンバータ機能のある端子では、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『A/D コンバータ』の章を参照してください。
①当該端子の PFR をポート機能側に設定します。
②当該端子を使用するほかのすべてのペリフェラルの EPFR をディセーブルにします
③当該端子が外バス兼用の場合や、当該ペリフェラルが I/O マルチプレクシング対象の場合、当該ペリフェラルの EPFR を設定します。また、当該ペリフェラルに I/O リロケーションがある場合、当該ペリフェラルの EPFR で使用する端子を設定します。
④PFR をペリフェラル側に設定します。

図 11-5 ペリフェラル出力の割り当て手順図



11.5.1.4 外バスの割り当て

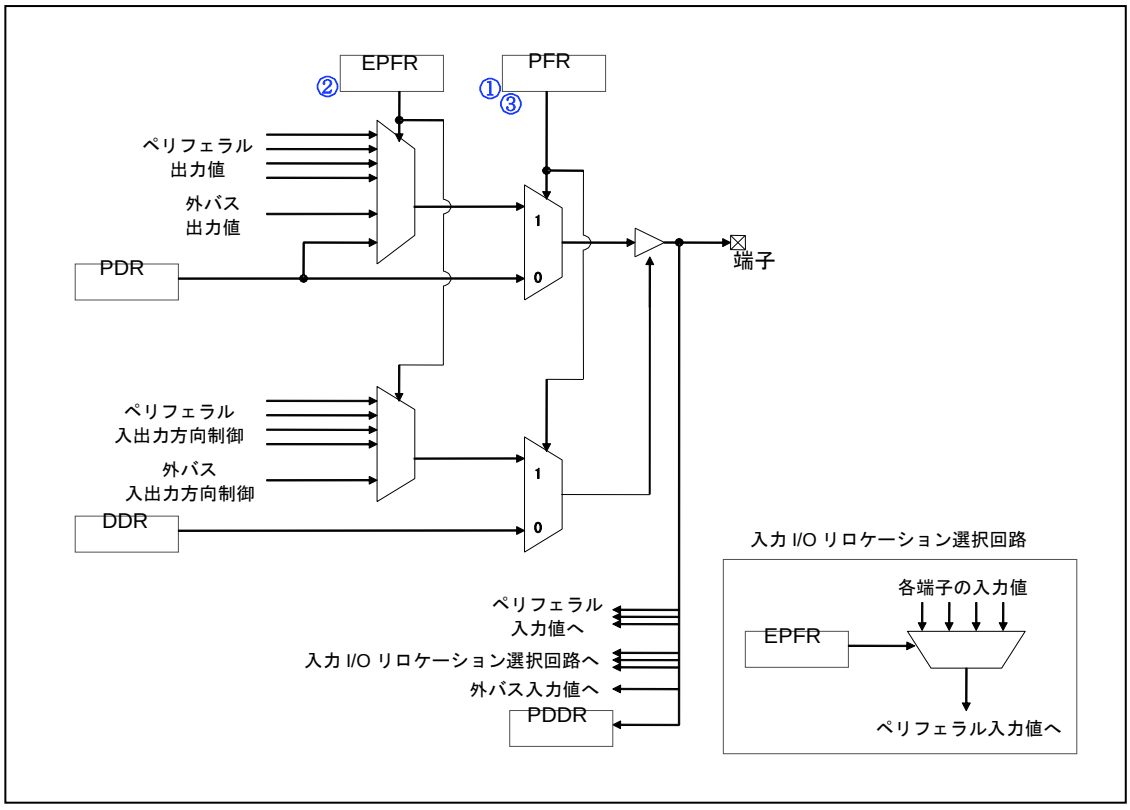
外バスの割り当てについて説明します。

(準備)

①でいったん端子はポート機能になりますので、必要な場合は前もって DDR および PDR 値を設定しておいてください。

A/D コンバータ機能のある端子では、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『A/D コンバータ』の章を参照してください。
①当該端子の PFR をポート機能側に設定します。
②外バスと同じ端子を使用するほかのすべてのペリフェラルの EPFR をディセーブルにします。
③PFR をペリフェラル側に設定します。

図 11-6 外バスの割り当て手順図



11.5.1.5 ポート機能 (入力) の割り当て

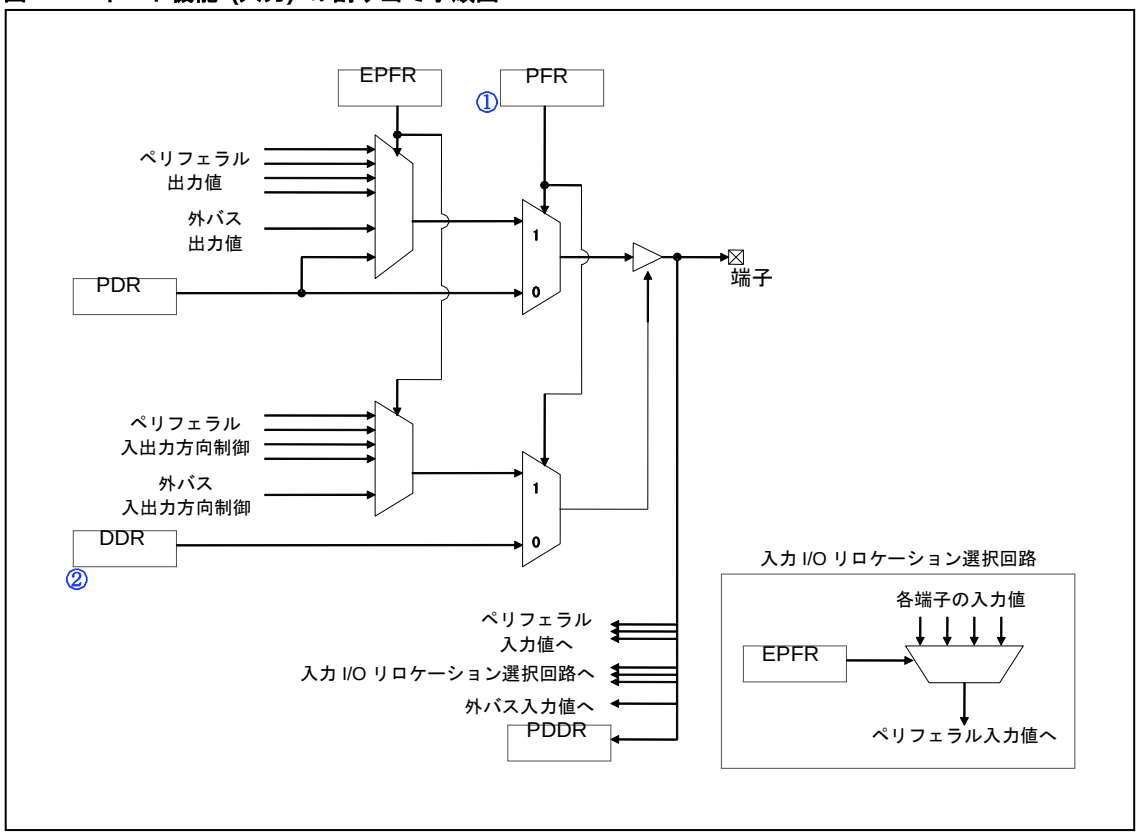
ポート機能 (入力) の割り当てについて説明します。

(準備)

A/D コンバータ機能のある端子では、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『A/D コンバータ』の章を参照してください。

①PFR をポート機能側に設定します。
②DDR を入力側に設定します。

図 11-7 ポート機能 (入力) の割り当て手順図



11.5.1.6 ポート機能 (出力) の割り当て

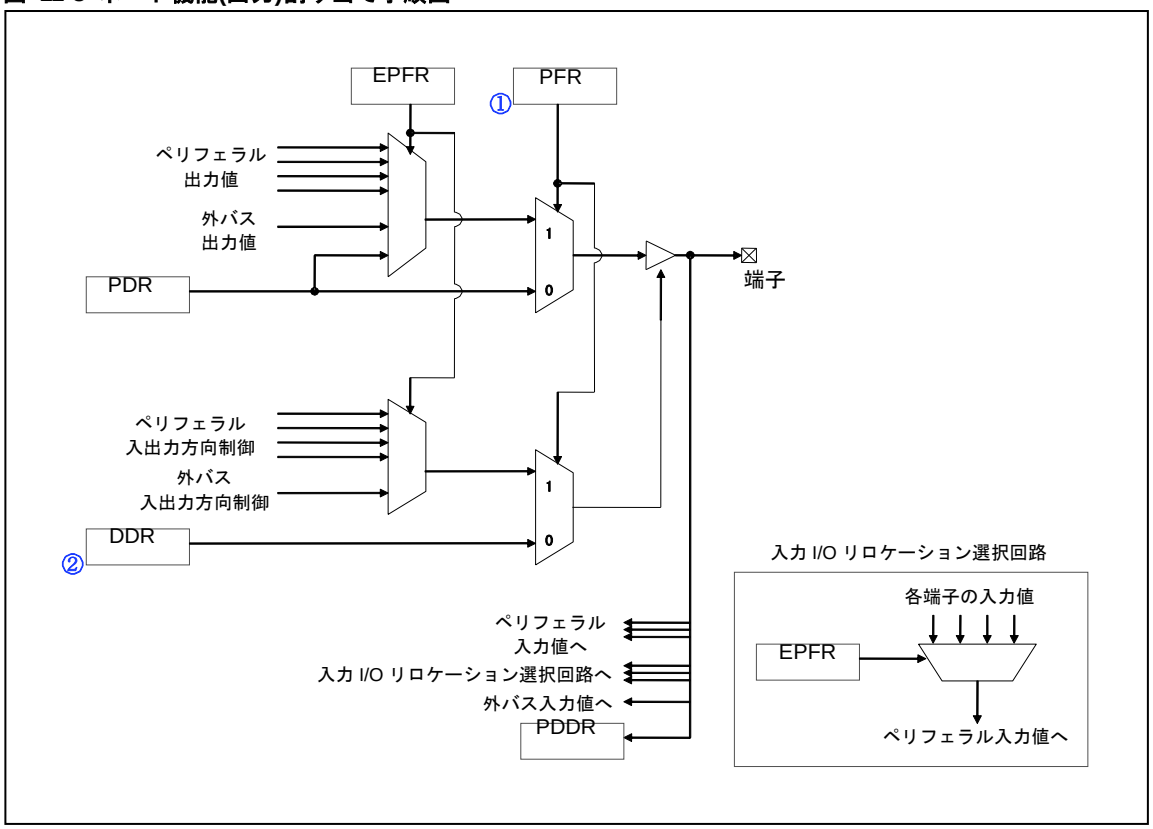
ポート機能(出力)の割り当てについて説明します。

(準備)

A/D コンバータ機能のある端子では、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『A/D コンバータ』の章を参照してください。

①PFR をポート機能側に設定します。
②DDR を出力側に設定します。

図 11-8 ポート機能(出力)割り当て手順図



11.5.1.7 A/D コンバータ入力の割り当て

A/D コンバータ入力の割り当てについて説明します。

A/D コンバータのアナログ入力許可レジスタ(ADER)をアナログ入力モードに設定します。『A/D コンバータ』の章を参照してください。
A/D コンバータの割り当ては最上位の優先度なので、設定は以上のみです。

11.5.1.8 D/A コンバータ出力の割り当て

DA コンバータ出力の割り当てについて示します。

「11.5.1.3. ペリフェラル出力の割り当て」と同じです。

11.5.1.9 LCD コントローラ V0・V1・V2・V3 (LCDC 基準電源入力) の割当て

LCD コントローラ V0～V3(LCDC 基準電源入力) の割当てについて示します。

(準備)

①でいったん端子はポート機能になりますので、必要な場合は前もって DDR および PDR 値を設定しておいてください。

①当該 PFR(PFR05.bit4/5/6/7) をポート機能側にします。
②端子共有するペリフェラルの EPFR をディセーブル側に設定します。
③当該 EPFR(EPFR20.bit4/5/6/7) を LCDC 基準電源入力許可側にします。
④当該 PFR(PFR05.bit4/5/6/7) をペリフェラル側に設定します。

11.5.2 EPFR 設定の優先度について

EPFR 設定の優先度について示します。

PFR がペリフェラル側に設定されていて、1 つの端子に対し EPFR 設定が重複する場合、次の優先度で有効なペリフェラルが決定します。

1. LCD 出力、DA コンバータ出力
2. CAN
3. マルチファンクションシリアルインタフェース
4. LIN-UART
5. HS_SPI
6. PPG
7. サウンドジェネレータ
8. リアルタイムクロック
9. ベースタイマ
10. リロードタイマ
11. アウトプットコンペア

11.5.3 入力 I/O リロケーション設定時の注意

入力 I/O リロケーション設定時の注意について示します。

入力端子を別端子に切り換える際に、切換え前と後との端子のレベルが違うと、I/O リロケーション切換えが当該端子をトリガとするペリフェラルへのトリガ入力になることがありますので注意してください。

11.5.4 GPORTEN による入力遮断

GPORTEN による入力遮断について示します。

ポートがソフトウェアにより設定される前に貫通電流の変動を避けるために、パワーオンリセット時に大部分の端子は入力遮断になります。入力遮断になる端子については、『付録』の『D. CPU 状態による端子状態』を参照してください。入力遮断解除方法については「11.4.12. ポート入力許可レジスタ: PORTEN (PORT ENable register)」を参照してください。

GPORTEN による入力遮断中に、入力遮断となる端子の状態を読み出した場合、必ず"0"が読み出されます。

11.5.5 A/D コンバータ機能のある端子の注意事項

A/D コンバータ機能のある端子の注意事項について示します。

A/D コンバータ機能のある端子で A/D コンバータ以外の機能を利用する場合は、A/D コンバータのアナログ入力許可レジスタ(ADER)の該当ビットを「ポート入力/出力モード」に設定しておいてください。設定方法は『A/D コンバータ』の章を参照してください。アナログ入力許可の場合、ポート機能レジスタ (PFR00～PFR19) や拡張ポート機能レジスタ (EPFR00～EPFR54)の設定にかかわらず、 ポートおよび周辺機能からの入力は"0"に、出力は Hi-Z に固定されます。

11.5.6 ベースタイマ TIOA1 端子使用時の設定

ベースタイマ TIOA1 端子使用時の設定について示します。

ベースタイマ TIOA1 端子を使用する場合､ベースタイマ入出力モード 1 の場合は入力､ベースタイマ入出力モード 1 以外の場合は出力に設定する必要があります。ベースタイマ TIOA1 端子を使用する場合で、ベースタイマ入出力モード 1 の場合は端子をペリフェラル入力(「11.5.1.2. ペリフェラル入力の割り当て」参照)、ベースタイマ入出力モード 1 以外の場合は端子をペリフェラル出力(「11.5.1.3. ペリフェラル出力の割り当て」参照)に設定してください。

11.5.7　電源遮断からのウェイクアップ時の動作

電源遮断からのウェイクアップ時の動作について示します。

PMUCTLR:IOCTMD をセットしている場合には、電源遮断からのウェイクアップ時に I/O は保持されます。I/O の保持は、PMUCTLR:IOCT をセットするまで継続します。

PMUCTLR:IOCTMD をクリアしている場合には、電源遮断からのウェイクアップ時に I/O は保持されますが、ウェイクアップ完了後に保持状態は解除され、I/O ポートのレジスタが有効になります。

電源遮断からのリセット復帰後には、I/O のラッチが解除されていない場合があります。

電源遮断からのリセット復帰後には、必ず PMUCTLR.IOCT ビットに"1"を書き込んで IO 保持の解除をおこなってください。

11.5.8　I/O ポート機能切り替え時の注意事項

I/O ポート機能切り替え時の注意事について示します。

I/O ポートをポート機能からリソースに切り替えた場合もしくはリソースからポート機能に切り替えた場合、一瞬 PDR の値が出力されることがあります。

切り替え時に、ポート機能が「入力から出力」もしくは「出力から入力」に変わる場合に起こります。

本出力が問題になる場合、あらかじめ PDR に問題の起こらないレベルの値を書き込んでください。

11.5.9　特定リソース使用時の入力遮断

特定リソース使用時の入力遮断について示します。

端子を A/D コンバータ, D/A コンバータ, LCD, SMC 機能として使用する場合、該当端子の状態を読み出した場合、必ず"0"が読み出されます。

12. 割込み制御 (割込みコントローラ)

割込み制御 (割込みコントローラ) について説明します。

12.1 概要

割込み制御 (割込みコントローラ) の概要について説明します。

割込みコントローラは、割込み要求の調停を行います。

12.2 特長

割込み制御 (割込みコントローラ) の特長について説明します。

本モジュールは、以下により構成されています。

- ICR レジスタ
- 割込み優先度判定回路
- 割込みレベル, 割込みベクタ発生回路

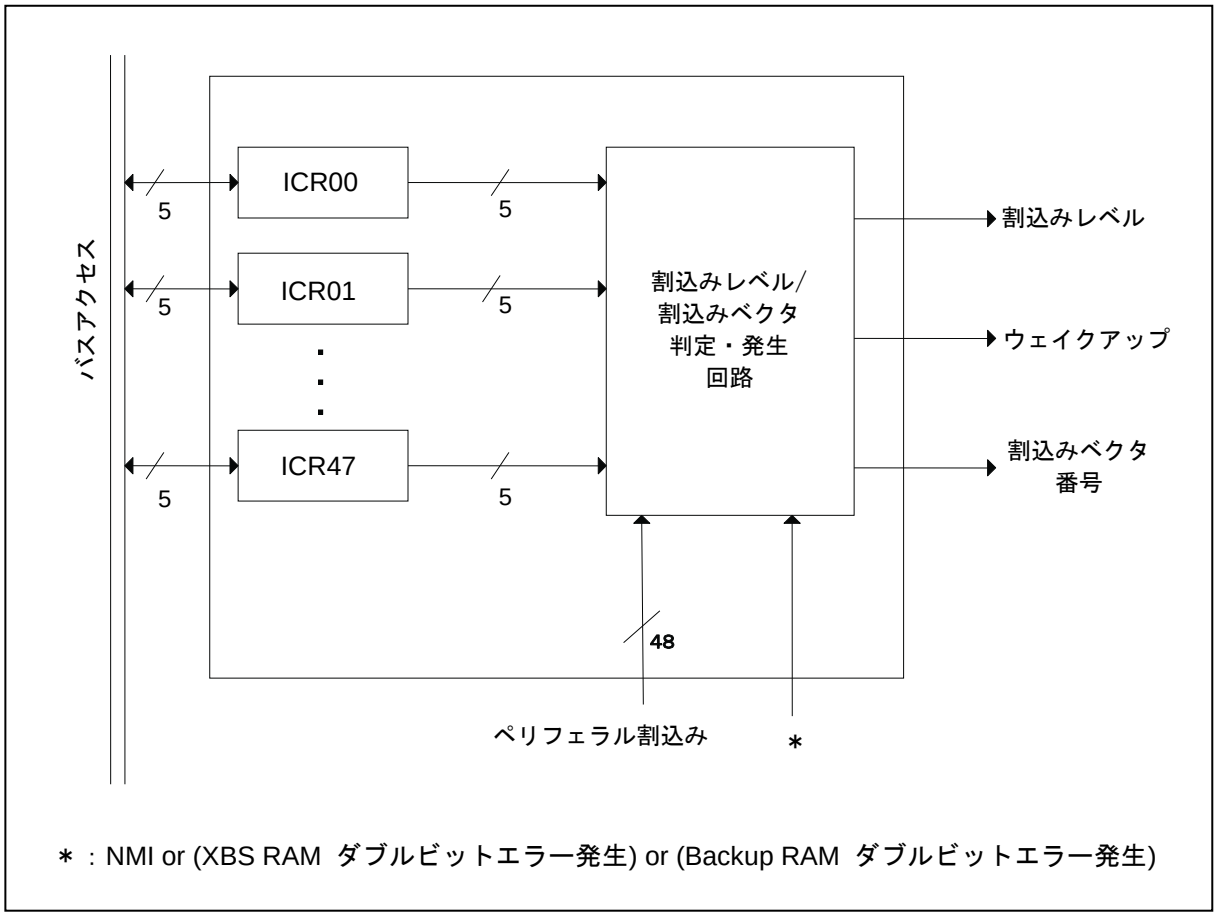
本モジュールには以下の機能があります。

- NMI 要求 / ペリフェラル割込み要求の検出
- 優先度判定 (レベルおよび割込みベクタによる)
- 最も優先度の高い要因の割込みレベルを CPU へ伝達
- 最も優先度の高い要因の割込みベクタ番号を CPU へ伝達
- NMI/ 割込みレベルが"11111"以外の割込み発生によるウェイクアップ要求の生成

12.3 構成

割込み制御 (割込みコントローラ) の構成について説明します。

図 12-1 ブロックダイヤグラム



12.4 レジスタ

割込み制御 (割込みコントローラ) のレジスタについて説明します。

表 12-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0440	ICR00	ICR01	ICR02	ICR03	割込みコントロールレジスタ 00～47
0x0444	ICR04	ICR05	ICR06	ICR07	
0x0448	ICR08	ICR09	ICR10	ICR11	
0x044C	ICR12	ICR13	ICR14	ICR15	
0x0450	ICR16	ICR17	ICR18	ICR19	
0x0454	ICR20	ICR21	ICR22	ICR23	
0x0458	ICR24	ICR25	ICR26	ICR27	
0x045C	ICR28	ICR29	ICR30	ICR31	
0x0460	ICR32	ICR33	ICR34	ICR35	
0x0464	ICR36	ICR37	ICR38	ICR39	
0x0468	ICR40	ICR41	ICR42	ICR43	
0x046C	ICR44	ICR45	ICR46	ICR47	

12.4.1 割込みコントロールレジスタ 00～47 : ICR00-ICR47 (Interrupt Control Register 00-47)

割込みコントロールレジスタ 00～47 (ICR00-ICR47)のビット構成について説明します。

各割込み入力に対して 1 ずつ設けられており、対応する割込み要求のレベルを設定します。

● ICR00～ICR47 : アドレス 0440H～046FH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			IL[4:0]				
初期値	1	1	1	1	1	1	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W

[bit4～bit0] IL[4:0] (Interrupt Level control) : **割込みレベルコントロール**
割込みレベル設定ビットで、対応する割込み要求の割込みレベルを指定します。本レジスタに設定した割込みレベルが CPU の ILM レジスタのレベルマスク値以上の場合は、CPU 側にて割込み要求はマスクされます。リセットにより、"5'b11111"に初期化されます。

設定可能な割込みレベル設定ビットと割込みレベルの対応を示します。

IL[4:0]	割込みレベル	
10000	16	設定可能な最強レベル
10001	17	↑ (強)
10010	18	|
10011	19	|
10100	20	|
10101	21	|
10110	22	|
10111	23	|
11000	24	|
11001	25	|
11010	26	|
11011	27	|
11100	28	|
11101	29	|
11110	30	↓ (弱)
11111	31	割込み禁止

IL4 は"1"固定です。書込みは効果ありません。

12.5 動作説明

割込み制御 (割込みコントローラ) の動作について説明します。

■ 設定

1. 割込みを発生させたいペリフェラルに対応する割込みベクタ番号の ICR レジスタを設定します。
2. 割込みを発生させたいペリフェラルの設定をします。(ペリフェラル側で割込み出力を許可に設定します。)

■ 起動

設定したペリフェラルを起動します。

■ 優先順位判定

本モジュール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割込み要因を選択し、その割込み要因の割込みレベルと割込みベクタ番号を CPU へ出力します。

割込み要因の優先順位判定基準は、次のとおりです。

3. NMI
4. 以下の条件を満たす要因

- 割込みレベルの数値が 31(5'b11111)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。
- 割込みレベルが同じとき (31 以外) は、その中で最も小さい割込みベクタ番号を持つ要因。

上記の判定基準により、割込み要因が 1 つも選択されなかった場合は、割込みレベルとして 31(5'b11111) を出力します。そのときの割込みベクタ番号は不定です。

■ ストップモードからの復帰

割込み要求でストップモードから復帰する機能を本モジュールで実現します。(NMI を含む) ペリフェラルからの割込み要求 (割込みレベルが"5'b11111"以外) が発生すると、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。

割込み優先度判定部はストップから復帰後、クロックが供給されてから動作を再開しますので、割込み優先度判定部の結果が出るまでの間 CPU は命令を実行します。

ストップモードからの復帰要因として使用しない割込みは､対応する割込みコントロールレジスタ (ICR00～ICR47) の割込みレベルを"5'b11111" (割込み禁止) に設定してください。

■ スタンバイモード (電源遮断) からの復帰

割込みレベルが ICR=0x1F (割込み禁止) より高く、かつ割込み要因が発生している状態では、スタンバイ復帰要因が有効であるため、電源遮断状態へ遷移することはできません。そのまま命令実行を続けます。

割込みレベルが ICR=0x1F(割込み禁止)で、かつ割込み要因が発生している状態では、スタンバイ復帰要因にはなりませんので、電源遮断状態へいちどは遷移しますが、電源遮断復帰の要因ありの状態なので、直ぐに電源遮断復帰シーケンスを経て復帰します。(リセットベクタから実行されます)

13. 外部割込み入力

外部割込み入力について説明します。

13.1 概要

外部割込み入力の概要について説明します。

外部割込み入力端子(INT0～INT15)から入力される割込み要求です。

13.2 特長

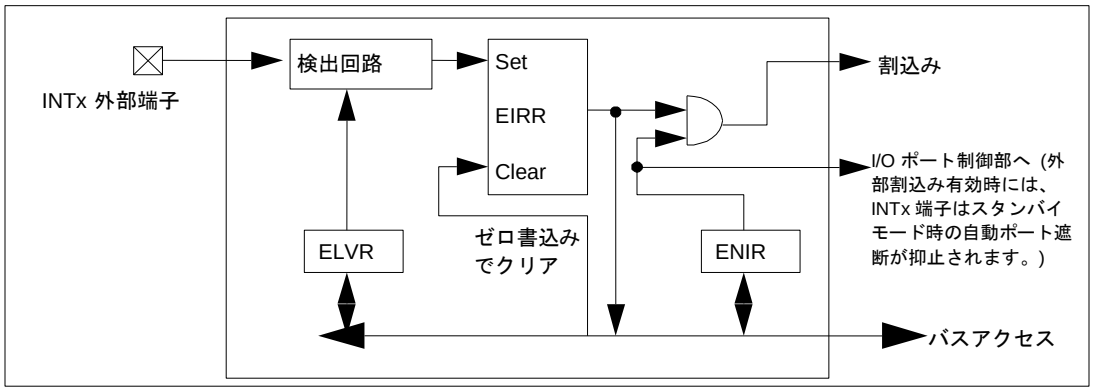
外部割込み入力の特長について説明します。

- 16 系統の外部割込み入力端子 (INT0～INT15)
- 割込み検出要因 : 4 種類 ("L" レベル,"H" レベル, 立上りエッジ, 立下りエッジ)

13.3 構成

外部割込み入力の構成について示します。

図 13-1 ブロックダイヤグラム



13.4 レジスタ

外部割込み入力のレジスタについて説明します。

チャネル	Base_addr	外部端子
		INT
0	0x0550	INT0_0 / INT0_1
1	0x0550	INT1_0 / INT1_1
2	0x0550	INT2_0 / INT2_1
3	0x0550	INT3_0 / INT3_1
4	0x0550	INT4_0 / INT4_1
5	0x0550	INT5_0 / INT5_1
6	0x0550	INT6_0 / INT6_1
7	0x0550	INT7_0 / INT7_1
8	0x0554	INT8_0 / INT8_1
9	0x0554	INT9_0 / INT9_1
10	0x0554	INT10_0 / INT10_1
11	0x0554	INT11_0 / INT11_1
12	0x0554	INT12_0 / INT12_1
13	0x0554	INT13_0 / INT13_1
14	0x0554	INT14_0 / INT14_1
15	0x0554	INT15_0 / INT15_1

表 13-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0550	EIRR0	ENIR0	ELVR0		外部割込み要因レジスタ 0 外部割込み許可レジスタ 0 外部割込み要求レベルレジスタ 0
0x0554	EIRR1	ENIR1	ELVR1		外部割込み要因レジスタ 1 外部割込み許可レジスタ 1 外部割込み要求レベルレジスタ 1

13.4.1 外部割込み要因レジスタ 0/1 : EIRR0/1: (External Interrupt Request Register 0/1)

外部割込み要因レジスタ 0/1 (EIRR0/1)のビット構成について説明します。

外部割込み要因が発生したことを保持するレジスタです。

■ **EIRR0 ：アドレス 0550H (アクセス: バイト, ハーフワード, ワード)**

■ **EIRR1 ：アドレス 0554H (アクセス: バイト, ハーフワード, ワード)**

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0
初期値	X	X	X	X	X	X	X	X
属性	R(RM1),W	R(RM1),W	R(RM1),W	R(RM1),W	R(RM1),W	R(RM1),W	R(RM1),W	R(RM1),W

[bit7～bit0] ER7～ER0 (External interrupt Request7-0) : **外部割込みリクエストビット**
INT 外部端子入力による割込み要求がある事を示すフラグです。"0"書込みでクリアします。

ERn	意味	
	読出し時	書込み時
0	外部割込み要求なし	クリア
1	外部割込み要求あり	動作に影響なし

1. EIRR0:ER0 が INT0 端子、EIRR0:ER1 が INT1 端子、・・、EIRR0:ER7 が INT7 端子、EIRR1:ER0 が INT8 端子、・・EIRR1:ER7 が INT15 端子に対応します。
2. 本ビットへの"1"の書込みは動作に影響を与えません。
3. リード・モディファイ・ライト(RMW)系命令における読出し値は常に"1"となります。
4. 外部割込み検出条件が L レベルまたは"H"レベル設定のとき、EIRR レジスタの各ビットをクリアしても外部割込み端子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。
5. 割込み要求レベルレジスタの変更により，割込み要因レジスタ内の要因ビットがセットされる場合があります。割込み要求レベルレジスタの変更後には，割込み要因レジスタを初期化してください。
6. 本レジスタのリセット後の値は、リセット後の端子状態によります。
7. 本レジスタはPMUCTLR:IOCTMD=1時のスタンバイ (電源遮断) からの復帰リセットを除いたすべてのリセット要因で初期化されます。

13.4.2 外部割込み許可レジスタ 0/1 : ENIR0/1: (ENable Interrupt request Register 0/1)

外部割込み許可レジスタ 0/1 (ENIR0/1)のビット構成について説明します。

外部割込み入力を許可するレジスタです。

■ **ENIR0 ：アドレス 0551H (アクセス: バイト, ハーフワード, ワード)**

■ **ENIR1 ：アドレス 0555H (アクセス: バイト, ハーフワード, ワード)**

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] EN7～EN0 (interrupt ENable) : **外部割込み許可ビット**
外部端子 INT 入力による割込み要求のマスク制御を行います。

ENn	外部端子検出時の動作
0	割込み要求マスク。割込み要求を保持するが出力しない。(初期値)
1	割込み要求許可。割込み要求を許可する。

1. ENIR0:EN0 が INT0 端子、ENIR0:EN1 が INT1 端子、・・、ENIR0:EN7 が INT7 端子、ENIR1:EN0 が INT8 端子、・・ENIR1:EN7 が INT15 端子に対応します。
2. 本レジスタは PMUCTLR:IOCTMD=1 時のスタンバイ (電源遮断) からの復帰リセットを除いたすべてのリセット要因で初期化されます。

13.4.3 外部割込み要求レベルレジスタ 0/1 : ELVR0/1: (External interrupt LeVel Register 0/1)

外部割込み要求レベルレジスタ 0/1 (ELVR0/1)のビット構成について説明します。

外部割込み要求の検出条件を選択するレジスタです。

■ **ELVR0 ：アドレス 0552H (アクセス: バイト, ハーフワード, ワード)**

■ **ELVR1 ：アドレス 0556H (アクセス: バイト, ハーフワード, ワード)**

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	LB7	LA7	LB6	LA6	LB5	LA5	LB4	LA4
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit1] LB7～LB0 (Level select B) : レベル選択 B

[bit14～bit0] LA7～LA0 (Level select A) : レベル選択 A

外部割込み要求の検出条件を選択します。LA, LB ビットの 2 ビットを組み合わせて使用します。

LBn	LAn	検出条件
0	0	"L"レベル検出 (初期値)
0	1	"H"レベル検出
1	0	立上りエッジ検出
1	1	立下りエッジ検出

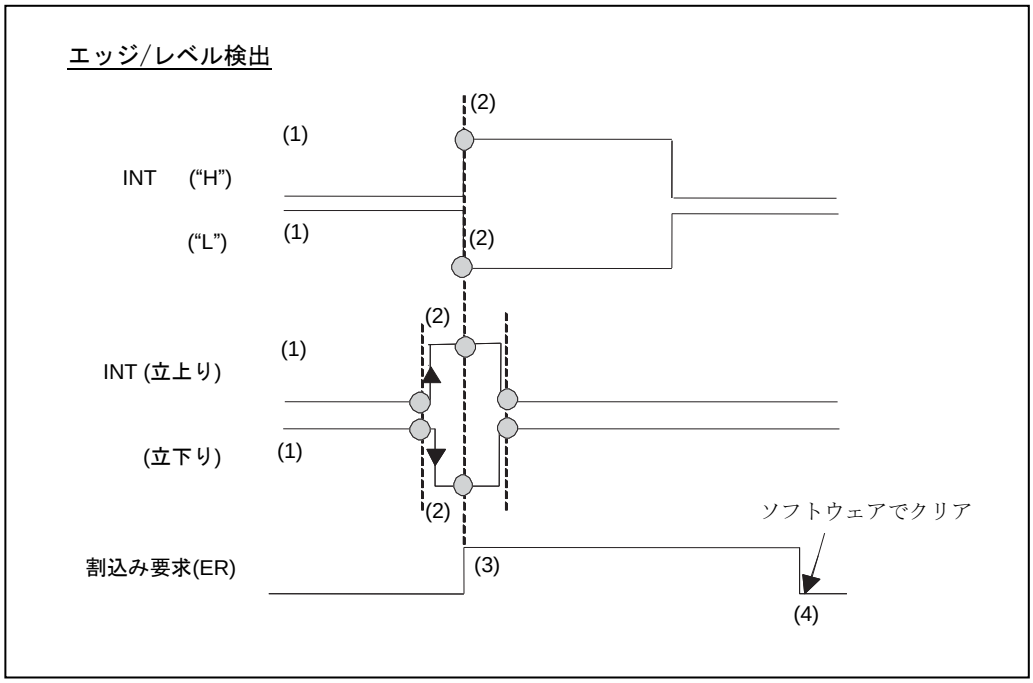
要求入力がレベルの場合 (LAn, LBn =00 または 01), 外部割込み要求ビット (ERn) を"0"にしても INTn 端子入力が有効レベルならば該当するビット (ERn) は再び"1" になります。

1. ELVR0:LA/LB0 が INT0 端子、ELVR0:LA/LB1 が INT1 端子、・・、ELVR0:LA/LB7 が INT7 端子、ELVR1:LA/LB0 が INT8 端子、・・ELVR1:LA/LB7 が INT15 端子に対応します。
2. 割込み要求レベルレジスタの変更により, 割込み要因レジスタ内の要因ビットがセットされる場合があります。割込み要求レベルレジスタの変更後には, 割込み要因レジスタを初期化してください。
3. 本レジスタは PMUCTLR:IOCTMD=1 時のスタンバイ (電源遮断) からの復帰リセットを除いたすべてのリセット要因で初期化されます。

13.5 動作説明

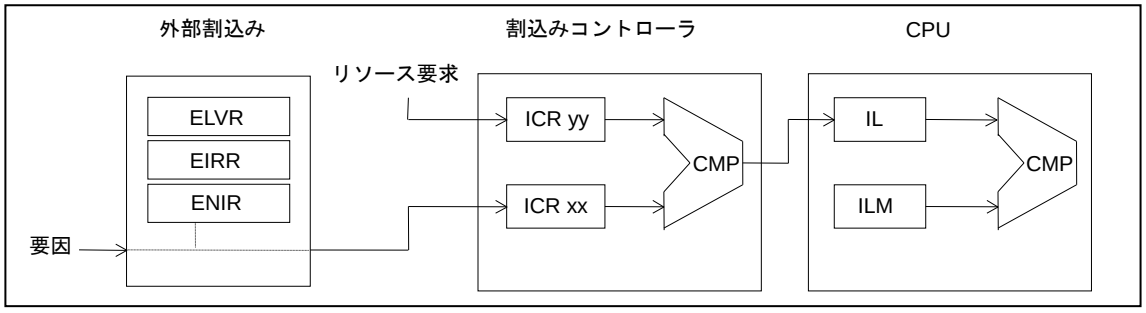
外部割込み入力の動作について説明します。

図 13-2 動作説明図



(1) 外部割込み信号 (INT) 入力
(2) 割込み信号を検出 (レベル/エッジ) する
(3) 割込み要求が発生する
(4) 割込み要求をソフトウェアでクリアする

図 13-3 外部割込みの動作



1. 外部割込みの動作について
   要求レベル、許可レジスタの設定の後、対応する端子に ELVR レジスタで設定された要求が入力されると本モジュールは、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。割込みコントローラ内で同時発生した割込みの優先順位を識別した結果、本リソースからの割込みが最も優先順位が高かったときに、該当する割込みが発生します。

2. スタンバイモードへの移行について
使用しないチャネルは、スタンバイに入る前に、必ず禁止状態にしてください。また、有効にしてあるチャネルの外部端子へのスタンバイモード自動入出力遮断は抑止されます。自動入出力遮断については『消費電力制御』の章を参照してください。

3. 外部割込みの設定手順について
外部割込み部内に存在するレジスタの設定を行う際、次の手順で設定してください。

① 許可レジスタの対象となるビットを禁止状態にする。
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 要求レベルレジスタを読み出す。
④ 要因レジスタの対象となるビットをクリアする。
⑤ 許可レジスタの対象となるビットを許可状態にする。
（ただし、④と⑤は 16 ビットデータによる同時書込み可。）

本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禁止状態に設定しておかなくてはなりません。また、許可レジスタを許可状態にする前に必ず要因レジスタをクリアしておく必要があります。

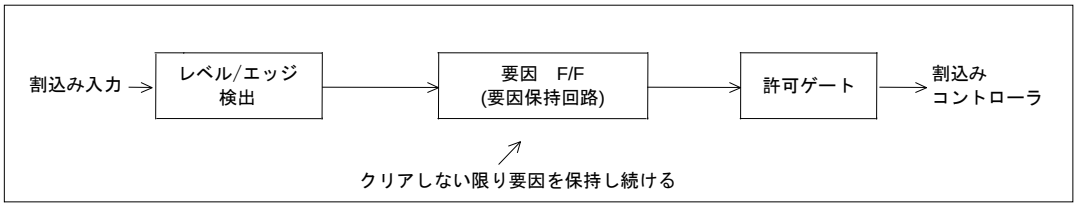
これは、レジスタ設定時や割込み許可状態時に誤って割込み要因が起こってしまうことを避けるためです。

4. 外部割込み要因について
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。

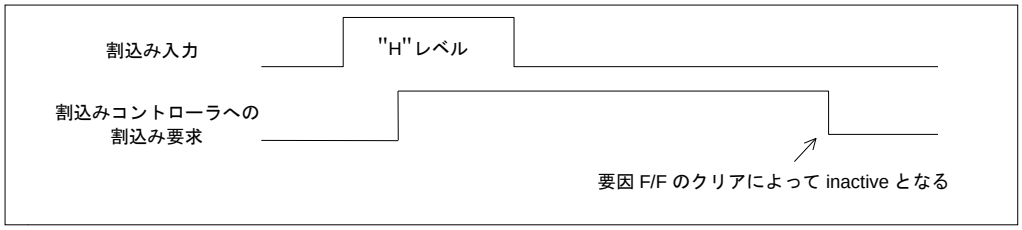
割込みコントローラへの要求を取り下げるには要因レジスタをクリアする必要があります。

図 13-4 要因保持回路のクリア、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

要因保持回路のクリア



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



13.6 設定

外部割込み入力の設定について説明します。

表 13-2 外部割込みを使うために必要な設定

設定	設定レジスタ	設定方法
検出レベルの設定	外部割込み要求レベル設定レジスタ (ELVR0、ELVR1)	「13.7 Q&A」の「■検出レベルの種類と設定方法は?」の項目を参照してください。
外部端子を入力にする	『I/O ポート』の章を参照してください。	『CHAPTER 11: I/O ポート』の章を参照してください。
外部割込み	外部端子より入力 → INT0～INT15 端子に信号を入力	—

13.7 Q&A

外部割込み入力の Q&A について説明します。

■ 検出レベルの種類と設定方法は?

検出レベルは 4 種類("L" レベル, "H" レベル, 立上り, 立下り)あります。

設定は, 検出レベルビット (ELVRy:LBn, LAn) (n=0～7, y=0,1) にて行ってください。

動作モード	検出レベルビット(LBn, LAn) n=0～7
"L" レベル検出にするには	"00"にする
"H" レベル検出にするには	"01"にする
立上り検出にするには	"10"にする
立下り検出にするには	"11"にする

■ 外部端子を入力にするには?

『I/O ポート』の章を参照してください。

■ 割込み関連レジスタは?

『割込み制御(割込みコントローラ)』の章を参照してください。

■ 割込みの種類は?

割込み要因は外部割込みのみです。選択のビットはありません。

■ 割込みを許可/禁止/クリアするには?

割込み要求の許可フラグ, 割込み要求フラグ

割込み許可の設定は, 割込み許可ビット(ENIR0/1:EN0～EN7)にて行います。

動作	割込み許可ビット(ENn)
割込み要求を禁止するには	"0"にする
割込み要求を許可するには	"1"にする

割込み要求のクリアは, 割込み要求ビット(EIRR0/1:ER0～ER7)にて行います。

動作	割込み要求ビット(ERn)
割込み要求をクリアするには	"0"を書き込む

13.8 注意事項

外部割込み入力の注意事項について説明します。

PMUCTLR:IOCTMD=1時のスタンバイ時計モード(電源遮断)およびスタンバイストップモード(電源遮断)からの復帰時では、外部割込み入力のレジスタは初期化されません。復帰前および復帰中の状態を保持する場合には、スタンバイ設定する前にPMUCTLR.IOCTMDをセットして、I/O保持状態とし、IOポート設定後にPMUCTLR.IOCTをセットしてI/O保持状態を解除してください。PMUCTLRレジスタについては『消費電力制御』の章を参照してください。

また、PMUCTLR:IOCTMD=1時のスタンバイ時計モード(電源遮断)およびスタンバイストップモード(電源遮断)からの復帰中は、内部リセットが発行されており、パワーオンリセット、内部電源低電圧リセットおよびRSTX・NMIX同時アサート以外のリセット要因を受付けません。このとき、外部割込み入力のレジスタは初期化されませんので、立上げ後にRSTX端子入力からのリセット入力または外部低電圧検出フラグがセットされている場合には、外部割込み入力のレジスタを初期化してから使用してください。

14. NMI 入力

NMI 入力について説明します。

14.1 概要

NMI 入力の概要について説明します。

NMI (Non Maskable Interrupt) は、NMIX 端子から入力されるマスクすることができない割込み入力です。NMI はストップモードからの復帰要因として使用できます。

14.2 特長

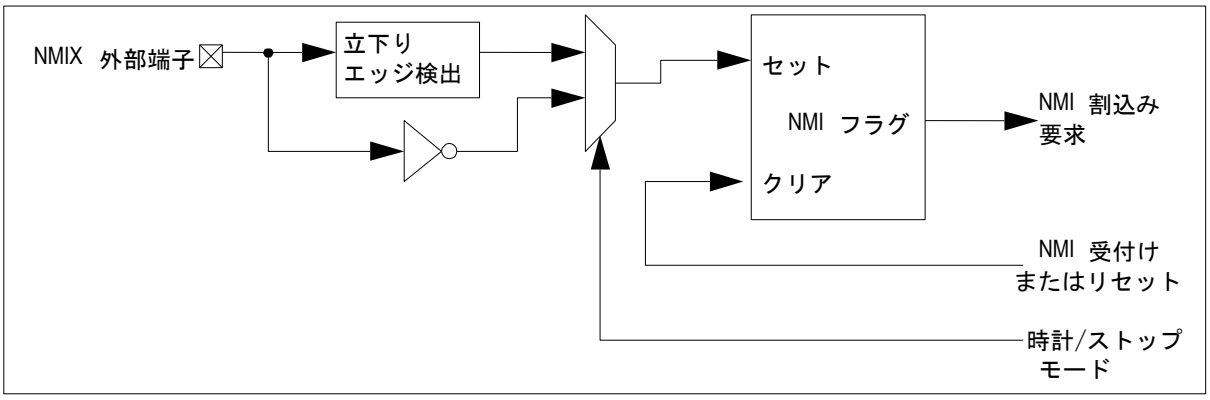
NMI 入力の特長について説明します。

ストップモード(電源遮断含む)、時計モード(電源遮断含む)でも使用可能です。

14.3 構成

NMI 入力の構成について示します。

図 14-1 ブロックダイヤグラム



14.4 レジスタ

NMI 入力のレジスタについて説明します。

本機能は、レジスタを持ちません。

14.5 動作説明

NMI 入力の動作について説明します。

■ NMI 割込みレベル

NMI は、ユーザ割込みの中で最強の割込みで、マスクできません。例外として、リセット後 CPU が ILM を設定するまで NMI はマスクされます。

■ NMI 外部端子

ストップモード中は L レベル検出、それ以外のときは立下りエッジ検出となります。

■ 割込み要求出力

NMI 要求検出部は NMI フラグがあり、NMI 要求によりセットされ、NMI 自身の割込み受付け、またはリセットでのみクリアされます。NMI フラグはリード/ライトできません。

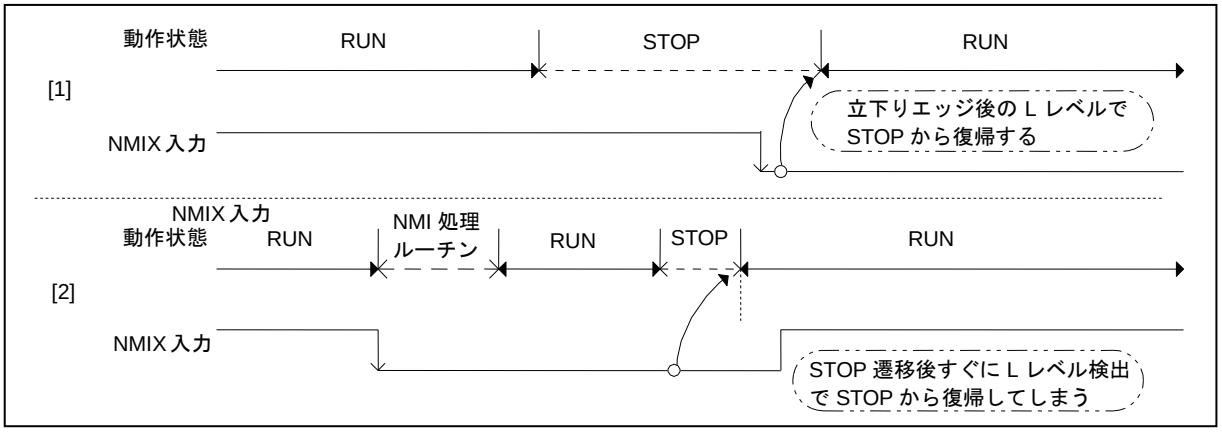
NMIX 外部端子による NMI かほかの要因による NMI かを判別するには IRPR15H レジスタをリードしてください。同レジスタの詳細は『割込み要求一括読出し』の章を参照してください。

■ ストップモードからの復帰

ストップモードに遷移している時、NMIX 入力が"L"レベルで NMI 要求が割込みコントローラに出力され、ストップモードから復帰します。通常状態(ストップモードではない時)で NMI 処理ルーチン終了後、NMIX 端子を"H"レベルに戻さずにストップモードに遷移すると、ストップモードへの遷移後すぐにストップから復帰してしまいます（図 14-2[2]）。また、同様に NMIX 端子を H レベルに戻さずにストップモード(電源遮断)に遷移すると、電源遮断制御はしません。ストップモード遷移前に NMIX 端子を"H"レベルに戻して、ストップモード中に NMIX 端子を"L"レベルにするようにしてください。

スタンバイモード(電源遮断)からの復帰中は、内部リセットが発行されており、NMI 要求を受け付けません。

図 14-2 ストップモードからの復帰



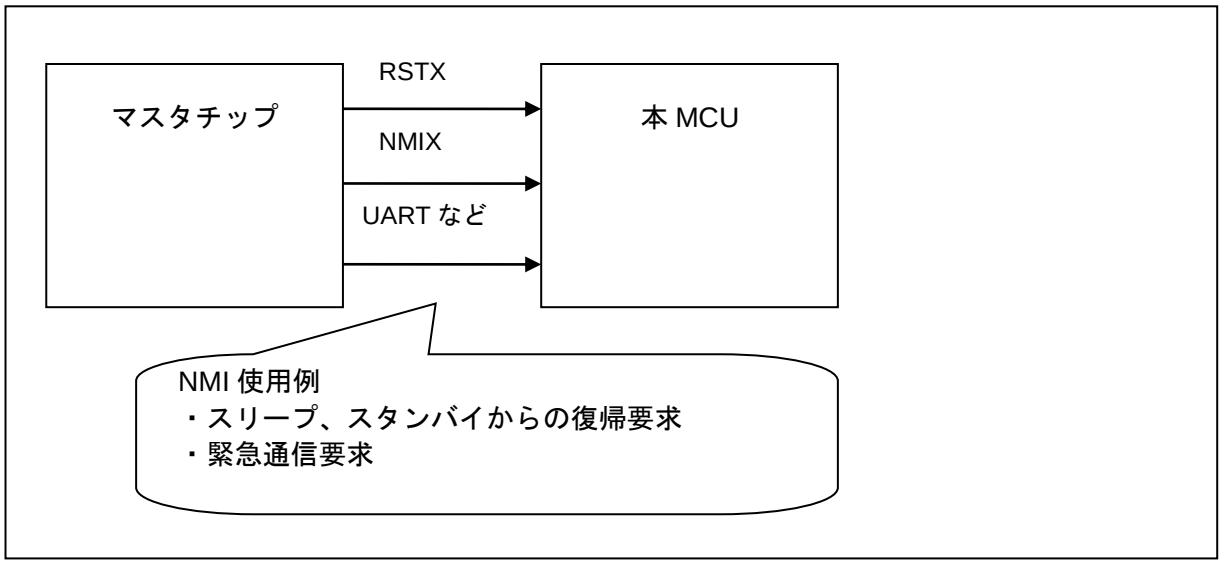
時計モード、時計モード (電源遮断) も同様な制御になります。

14.6 使用例

NMI 入力の使用例について示します。

NMI 機能の使用例です。

図 14-3 使用例



15. 遅延割込み

遅延割込みについて説明します。

15.1 概要

遅延割込みの概要について説明します。

遅延割込みとは、OS (オペレーションシステム)でのタスク切換え用の割込みを発生するための機能です。

本機能によって、ソフトウェアで CPU に対して割込み要求の発生 / 取消しができます。

15.2 特長

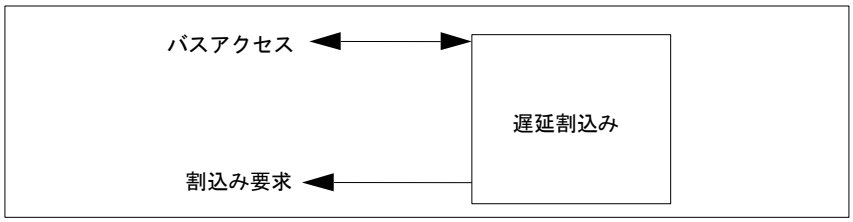
遅延割込みの特長について説明します。

レジスタ書込みにより、割込みを発生させることができます。

15.3 構成

遅延割込みの構成について示します。

図 15-1 ブロックダイヤグラム



15.4 レジスタ

遅延割込みのレジスタについて説明します。

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0044	DICR	予約	予約	予約	遅延割込み制御レジスタ

■ 遅延割込み制御レジスタ: DICR (Delayed Interrupt Control Register)

遅延割込みを制御します。

● DICR： アドレス 0044H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							DLYI
初期値	1	1	1	1	1	1	1	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W

[bit0] DLYI (DeLaYed Interrupt enable) : **遅延割込み許可ビット**

遅延割込み要因の発生・解除をします。

DLYI	説明
0 書込み	遅延割込み要因の解除
1 書込み	遅延割込み要因の発生

15.5 動作説明

遅延割込みの動作について説明します。

遅延割込みは、タスク切換え用の割込みを発生するものです。本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, 取消しができます。

■ 割込みベクタ番号

遅延割込みは、最も大きな割込みベクタ番号に対応した割込み要因に割り当てられています。

本コアでは、遅延割込みを割込みベクタ番号 63 (0x3F) に割り当てています。

■ DICR レジスタの DLYI ビット

このビットに"1"を書き込むことで、遅延割込み要因が発生します。また、"0"を書き込むことで、遅延割込み要因を解除します。

本ビットは、ほか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本ビットをクリアし、タスクの切換えをしてください。

15.6 制限事項

遅延割込みの制限事項について説明します。

遅延割込みは DMA 転送要求に使用しないでください。

16. 割込み要求一括読出し

割込み要求一括読出しの概要、特長、構成などについて説明します。

16.1 概要

割込み要求一括読出しの概要について説明します。

本モジュールにおいて、1 つの割込みベクタ番号に割り当てられた複数の割込み要求を一括で読み出すことができます。FR80 ファミリ CPU のビットサーチ命令を使用することで、どの割込み要求が発生しているかを確認することができます。

16.2 特長

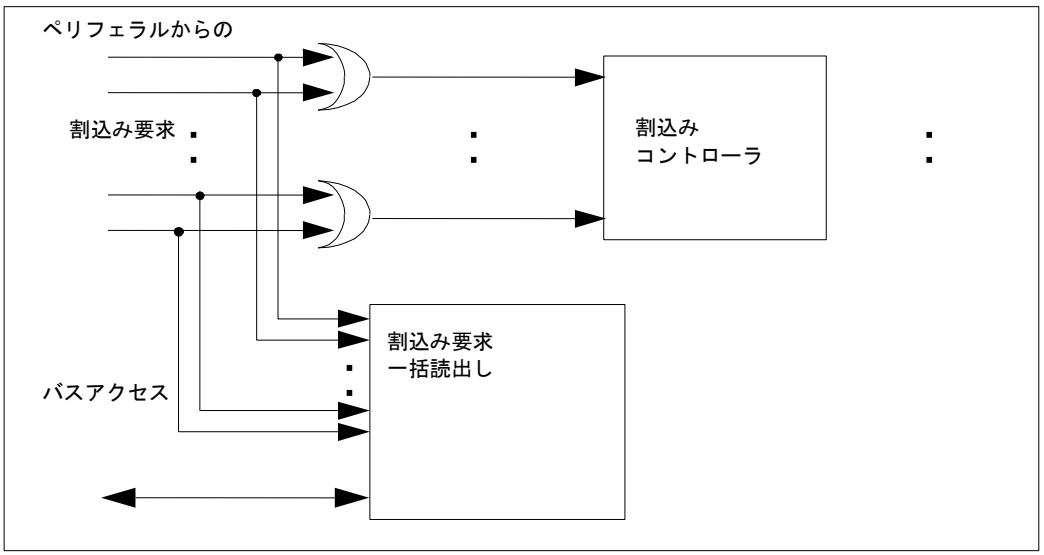
割込み要求一括読出しの特長について示します。

本モジュールを使用することにより割込み発生の有無を簡単に確認することができます。

16.3 構成

割込み要求一括読出しの構成について説明します。

図 16-1 ブロックダイヤグラム



16.4 レジスタ

割込み要求一括読出しのレジスタについて説明します。

表 16-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0418	IRPR0H	IRPR0L	IRPR1H	IRPR1L	割込み要求一括読出しレジスタ 0 上位(#18) 割込み要求一括読出しレジスタ 0 下位(#19) 割込み要求一括読出しレジスタ 1 上位(#20) 割込み要求一括読出しレジスタ 1 下位(#22)
0x041C	IRPR2H	IRPR2L	IRPR3H	IRPR3L	割込み要求一括読出しレジスタ 2 上位(#38) 割込み要求一括読出しレジスタ 2 下位(#39) 割込み要求一括読出しレジスタ 3 上位(#40) 割込み要求一括読出しレジスタ 3 下位(#41)
0x0420	IRPR4H	IRPR4L	IRPR5H	IRPR5L	割込み要求一括読出しレジスタ 4 上位(#42) 割込み要求一括読出しレジスタ 4 下位(#43) 割込み要求一括読出しレジスタ 5 上位(#44) 割込み要求一括読出しレジスタ 5 下位(#36)
0x0424	IRPR6H	IRPR6L	IRPR7H	IRPR7L	割込み要求一括読出しレジスタ 6 上位(#45) 割込み要求一括読出しレジスタ 6 下位(#46) 割込み要求一括読出しレジスタ 7 上位(#47) 割込み要求一括読出しレジスタ 7 下位(#49)
0x0428	IRPR8H	IRPR8L	IRPR9H	IRPR9L	割込み要求一括読出しレジスタ 8 上位(#50) 割込み要求一括読出しレジスタ 8 下位(#51) 割込み要求一括読出しレジスタ 9 上位(#52) 割込み要求一括読出しレジスタ 9 下位(#53)
0x042C	IRPR10H	IRPR10L	IRPR11H	IRPR11L	割込み要求一括読出しレジスタ 10 上位(#54) 割込み要求一括読出しレジスタ 10 下位(#55) 割込み要求一括読出しレジスタ 11 上位(#56) 割込み要求一括読出しレジスタ 11 下位(#57)
0x0430	IRPR12H	IRPR12L	IRPR13H	IRPR13L	割込み要求一括読出しレジスタ 12 上位(#58) 割込み要求一括読出しレジスタ 12 下位(#59) 割込み要求一括読出しレジスタ 13 上位(#60) 割込み要求一括読出しレジスタ 13 下位(#61)
0x0434	IRPR14H	IRPR14L	IRPR15H	予約	割込み要求一括読出しレジスタ 14 上位(#62) 割込み要求一括読出しレジスタ 14 下位(#62) 割込み要求一括読出しレジスタ 15 上位(#15)

#nn ： 割込みベクタ番号(10 進)

16.4.1 割込み要求一括読出しレジスタ 0 上位 ：IRPR0H (Interrupt Request Peripheral Read register 0H)

割込み要求一括読出しレジスタ 0 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#18)

■ IRPR0H： アドレス 0418H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RTIR0	RTIR1	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] RTIR0 (Reload Timer Interrupt Request 0) : リロードタイマ 0, 4 割込み要求

[bit6] RTIR1 (Reload Timer Interrupt Request 1) : リロードタイマ 1, 5 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

＜注意事項＞

RTIR0 はリロードタイマ 0 割込み要求とリロードタイマ 4 割込み要求の OR 論理、RTIR1 はリロードタイマ 1 割込み要求とリロードタイマ 5 割込み要求の OR 論理になっています。

16.4.2 割込み要求一括読出しレジスタ 0 下位 ：IRPR0L (Interrupt Request Peripheral Read register 0L)

割込み要求一括読出しレジスタ 0 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#19)

■ IRPR0L： アドレス 0419H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RTIR2	RTIR3	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] RTIR2 (Reload Timer Interrupt Request 2) : リロードタイマ 2, 6 割込み要求

[bit6] RTIR3 (Reload Timer Interrupt Request 3) : リロードタイマ 3 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

＜注意事項＞

RTIR2 はリロードタイマ 2 割込み要求とリロードタイマ 6 割込み要求の OR 論理になっています。

16.4.3 割込み要求一括読出しレジスタ 1 上位 ：IRPR1H (Interrupt Request Peripheral Read register 1H)

割込み要求一括読出しレジスタ 1 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#20)

■ IRPR1H ：アドレス 041AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RXIR0	ISIR0	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] RXIR0 (Multi-Function-Serial-Interface RX Interrupt Request 0) : マルチファンクションシリアルインタフェース ch.0 受信完了割込み要求

[bit6] ISIR0 (Multi-Function-Serial-Interface Status Interrupt Request 0) : マルチファンクションシリアルインタフェース ch.0 ステータス割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.4 割込み要求一括読出しレジスタ 1 下位 ：IRPR1L (Interrupt Request Peripheral Read register 1L)

割込み要求一括読出しレジスタ 1 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#22)

■ IRPR1L： アドレス 041BH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RXIR1	ISIR1	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] RXIR1 (Multi-Function-Serial-Interface RX Interrupt Request 1) : マルチファンクションシリアルインタフェース ch.1 受信完了割込み要求

[bit6] ISIR1 (Multi-Function-Serial-Interface Status Interrupt Request 1) : マルチファンクションシリアルインタフェース ch.1 ステータス割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.5 割込み要求一括読出しレジスタ 2 上位 ：IRPR2H (Interrupt Request Peripheral Read register 2H)

割込み要求一括読出しレジスタ 2 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#38)

■ IRPR2H： アドレス 041CH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SGIR0	RXIR7	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] SGIR0 (SG Interrupt Request 0) : サウンドジェネレータ 0 割込み要求

[bit6] RXIR7 (RX Interrupt Request 7) : LIN-UART7 受信完了割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.6 割込み要求一括読出しレジスタ 2 下位 ：IRPR2L (Interrupt Request Peripheral Read register 2L)

割込み要求一括読出しレジスタ 2 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#39)

■ IRPR2L： アドレス 041DH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SGIR1	TXIR7	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] SGIR1 (SG Interrupt Request 1) : サウンドジェネレータ 1 割込み要求

[bit6] TXIR7 (TX Interrupt Request 7) : LIN-UART7 送信完了割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.7 割込み要求一括読出しレジスタ 3 上位 ：IRPR3H (Interrupt Request Peripheral Read register 3H)

割込み要求一括読出しレジスタ 3 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#40)

■ IRPR3H：アドレス 041EH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PPGIR0	PPGIR1	PPGIR10	PPGIR11	PPGIR20	PPGIR21	予約	
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX

[bit7] PPGIR0 (PPG Interrupt Request 0) : PPG0 割込み要求
[bit6] PPGIR1 (PPG Interrupt Request 1) : PPG1 割込み要求
[bit5] PPGIR10 (PPG Interrupt Request 10) : PPG10 割込み要求
[bit4] PPGIR11 (PPG Interrupt Request 11) : PPG11 割込み要求
[bit3] PPGIR20 (PPG Interrupt Request 20) : PPG20 割込み要求
[bit2] PPGIR21 (PPG Interrupt Request 21) : PPG21 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.8 割込み要求一括読出しレジスタ 3 下位 ：IRPR3L (Interrupt Request Peripheral Read register 3L)

割込み要求一括読出しレジスタ 3 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#41)

■ IRPR3L： アドレス 041F$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PPGIR2	PPGIR3	PPGIR12	PPGIR13	PPGIR22	PPGIR23	予約	
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX

[bit7] PPGIR2 (PPG Interrupt Request 2) : PPG2 割込み要求
[bit6] PPGIR3 (PPG Interrupt Request 3) : PPG3 割込み要求
[bit5] PPGIR12 (PPG Interrupt Request 12) : PPG12 割込み要求
[bit4] PPGIR13 (PPG Interrupt Request 13) : PPG13 割込み要求
[bit3] PPGIR22 (PPG Interrupt Request 22) : PPG22 割込み要求
[bit2] PPGIR23 (PPG Interrupt Request 23) : PPG23 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.9 割込み要求一括読出しレジスタ 4 上位 ：IRPR4H (Interrupt Request Peripheral Read register 4H)

割込み要求一括読出しレジスタ 4 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#42)

■ IRPR4H： アドレス 0420H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PPGIR4	PPGIR5	PPGIR14	PPGIR15	予約			
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] PPGIR4 (PPG Interrupt Request 4) : PPG4 割込み要求

[bit6] PPGIR5 (PPG Interrupt Request 5) : PPG5 割込み要求

[bit5] PPGIR14 (PPG Interrupt Request 14) : PPG14 割込み要求

[bit4] PPGIR15 (PPG Interrupt Request 15) : PPG15 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.10 割込み要求一括読出しレジスタ 4 下位 ：IRPR4L (Interrupt Request Peripheral Read register 4L)

割込み要求一括読出しレジスタ 4 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#43)

■ IRPR4L： アドレス 0421H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PPGIR6	PPGIR7	PPGIR16	PPGIR17	予約			
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] PPGIR6 (PPG Interrupt Request 6) : PPG6 割込み要求
[bit6] PPGIR7 (PPG Interrupt Request 7) : PPG7 割込み要求
[bit5] PPGIR16 (PPG Interrupt Request 16) : PPG16 割込み要求
[bit4] PPGIR17 (PPG Interrupt Request 17) : PPG17 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.11 割込み要求一括読出しレジスタ 5 上位 ：IRPR5H (Interrupt Request Peripheral Read register 5H)

割込み要求一括読出しレジスタ 5 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#44)

■ IRPR5H： アドレス 0422H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PPGIR8	PPGIR9	PPGIR18	PPGIR19	予約			
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] PPGIR8 (PPG Interrupt Request 8) : PPG8 割込み要求

[bit6] PPGIR9 (PPG Interrupt Request 9) : PPG9 割込み要求

[bit5] PPGIR18 (PPG Interrupt Request 18) : PPG18 割込み要求

[bit4] PPGIR19 (PPG Interrupt Request 19) : PPG19 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.12 割込み要求一括読出しレジスタ 5 下位 ：IRPR5L (Interrupt Request Peripheral Read register 5L)

割込み要求一括読出しレジスタ 5 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#36)

■ IRPR5L： アドレス 0423H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	CANIR2	UDCIR0	UDCIR1	予約				
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] CANIR2 (CAN Interrupt Request 2) : CAN ch.2 割込み要求

[bit6] UDCIR0 (UpDown Counter Interrupt Request 0) : アップダウンカウンタ ch.0 割込み要求

[bit5] UDCIR1 (UpDown Counter Interrupt Request 1) : アップダウンカウンタ ch.1 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.13 割込み要求一括読出しレジスタ 6 上位 ：IRPR6H (Interrupt Request Peripheral Read register 6H)

割込み要求一括読出しレジスタ 6 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#45)

■ IRPR6H： アドレス 0424H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		RXIR8	ISIR8	HS_RIR	予約		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX

[bit5] RXIR8 (multifunction serial RX Interruut Request 8) : マルチファンクションシリアルインタフェース ch.8 受信完了割込み要求

[bit4] ISIR8 (multifunction serial Inform Status Interruut Request 8) : マルチファンクションシリアルインタフェース ch.8 ステータス割込み要求

[bit3] HS_RIR (HS_SPI RX Interruut Request) : HS_SPI 受信割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.14 割込み要求一括読出しレジスタ 6 下位 ：IRPR6L (Interrupt Request Peripheral Read register 6L)

割込み要求一括読出しレジスタ 6 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#46)

■ IRPR6L： アドレス 0425H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MTIR	STIR	PTIR	TXIR8	HS_TIR	予約		
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX

[bit7] MTIR (Main Timer Interrupt Request) : メインタイマ割込み要求

[bit6] STIR (Sub Timer Interrupt Request) : サブタイマ割込み要求

[bit5] PTIR (PLL Timer Interrupt Request) : PLL タイマ割込み要求

[bit4] TXIR8 (multifunction serial TX Interruut Request 8) : マルチファンクションシリアルインタフェース ch.8 送信完了割込み要求

[bit3] HS_TIR (HS_SPI TX Interruut Request) : HS_SPI 送信割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.15 割込み要求一括読出しレジスタ 7 上位 ：IRPR7H (Interrupt Request Peripheral Read register 7H)

割込み要求一括読出しレジスタ 7 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#47)

■ IRPR7H： アドレス 0426H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	SUBIR	SGIR4	RXIR9	ISIR9	予約		
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX

[bit6] SUBIR (SUB Interrupt Request) : クロックキャリブレーション(Sub)割込み要求

[bit5] SGIR4 (SG Interrupt Request 4) : サウンドジェネレータ 4 割込み要求

[bit4] RXIR9 (multifunction serial RX Interrupt Request 9) マルチファンクションシリアルインタフェース ch.9 受信完了割込み要求

[bit3] ISIR9 (multifunction serial Inform Status Interrupt Request 9) マルチファンクションシリアルインタフェース ch.9 ステータス割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.16 割込み要求一括読出しレジスタ 7 下位 ：IRPR7L (Interrupt Request Peripheral Read register 7L)

割込み要求一括読出しレジスタ 7 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#49)

■ IRPR7L： アドレス 0427H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						CRIR	TXIR9
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX

[bit1] CRIR (CR clock calibration Interrupt Request) : クロックキャリブレーション(CR)割込み要求

[bit0] TXIR9 (multifunction serial TX Interrupt Request 9) : マルチファンクションシリアルインタフェース ch.9 送信完了割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.17 割込み要求一括読出しレジスタ 8 上位 ：IRPR8H (Interrupt Request Peripheral Read register 8H)

割込み要求一括読出しレジスタ 8 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#50)

■ IRPR8H： アドレス 0428H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FRTIR0	FRTIR2	FRTIR4	予約				
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] FRTIR0 (FRT Interrupt Request 0) : フリーランタイマ ch.0 割込み要求
[bit6] FRTIR2 (FRT Interrupt Request 2) : フリーランタイマ ch.2 割込み要求
[bit5] FRTIR4 (FRT Interrupt Request 4) : フリーランタイマ ch.4 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.18 割込み要求一括読出しレジスタ 8 下位 ：IRPR8L (Interrupt Request Peripheral Read register 8L)

割込み要求一括読出しレジスタ 8 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#51)

■ IRPR8L： アドレス 0429H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FRTIR1	FRTIR3	FRTIR5	予約				
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] FRTIR1 (FRT Interrupt Request 1) : フリーランタイマ ch.1 割込み要求
[bit6] FRTIR3 (FRT Interrupt Request 3) : フリーランタイマ ch.3 割込み要求
[bit5] FRTIR5 (FRT Interrupt Request 5) : フリーランタイマ ch.5 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.19 割込み要求一括読出しレジスタ 9 上位 ：IRPR9H (Interrupt Request Peripheral Read register 9H)

割込み要求一括読出しレジスタ 9 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#52)

■ IRPR9H： アドレス 042AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICUIR0	ICUIR6	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] ICUIR0 (ICU Interrupt Request 0) : インプットキャプチャ ch.0 割込み要求

[bit6] ICUIR6 (ICU Interrupt Request 6) : インプットキャプチャ ch.6 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.20 割込み要求一括読出しレジスタ 9 下位 ：IRPR9L (Interrupt Request Peripheral Read register 9L)

割込み要求一括読出しレジスタ 9 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#53)

■ IRPR9L： アドレス 042B$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICUIR1	ICUIR7	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] ICUIR1 (ICU Interrupt Request 1) : インプットキャプチャ ch.1 割込み要求

[bit6] ICUIR7 (ICU Interrupt Request 7) : インプットキャプチャ ch.7 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.21 割込み要求一括読出しレジスタ 10 上位 ：IRPR10H (Interrupt Request Peripheral Read register 10H)

割込み要求一括読出しレジスタ 10 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#54)

■ IRPR10H： アドレス 042CH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICUIR2	ICUIR8	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] ICUIR2 (ICU Interrupt Request 2) : インプットキャプチャ ch.2 割込み要求

[bit6] ICUIR8 (ICU Interrupt Request 8) : インプットキャプチャ ch.8 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.22 割込み要求一括読出しレジスタ 10 下位 ：IRPR10L (Interrupt Request Peripheral Read register 10L)

割込み要求一括読出しレジスタ 10 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#55)

■ IRPR10L： アドレス 042DH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICUIR3	ICUIR9	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] ICUIR3 (ICU Interrupt Request 3) : インプットキャプチャ ch.3 割込み要求

[bit6] ICUIR9 (ICU Interrupt Request 9) : インプットキャプチャ ch.9 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.23 割込み要求一括読出しレジスタ 11 上位 ：IRPR11H (Interrupt Request Peripheral Read register 11H)

割込み要求一括読出しレジスタ 11 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#56)

■ IRPR11H： アドレス 042E$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICUIR4	ICUIR10	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] ICUIR4 (ICU Interrupt Request 4) : インプットキャプチャ ch.4 割込み要求

[bit6] ICUIR10 (ICU Interrupt Request 10) : インプットキャプチャ ch.10 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.24 割込み要求一括読出しレジスタ 11 下位 ：IRPR11L (Interrupt Request Peripheral Read register 11L)

割込み要求一括読出しレジスタ 11 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#57)

■ IRPR11L： アドレス 042F$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICUIR5	ICUIR11	予約					
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] ICUIR5 (ICU Interrupt Request 5) : インプットキャプチャ ch.5 割込み要求

[bit6] ICUIR11 (ICU Interrupt Request 11) : インプットキャプチャ ch.11 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.25 割込み要求一括読出しレジスタ 12 上位 ：IRPR12H (Interrupt Request Peripheral Read register 12H)

割込み要求一括読出しレジスタ 12 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#58)

■ IRPR12H： アドレス 0430H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	OCUIR0	OCUIR1	OCUIR6	OCUIR7	OCUIR10	OCUIR11	予約	
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX

[bit7] OCUIR0 (OCU Interrupt Request 0) : アウトプットコンペア ch.0 割込み要求
[bit6] OCUIR1 (OCU Interrupt Request 1) : アウトプットコンペア ch.1 割込み要求
[bit5] OCUIR6 (OCU Interrupt Request 6) : アウトプットコンペア ch.6 割込み要求
[bit4] OCUIR7 (OCU Interrupt Request 7) : アウトプットコンペア ch.7 割込み要求
[bit3] OCUIR10 (OCU Interrupt Request 10) : アウトプットコンペア ch.10 割込み要求
[bit2] OCUIR11 (OCU Interrupt Request 11) : アウトプットコンペア ch.11 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.26 割込み要求一括読出しレジスタ 12 下位 ：IRPR12L (Interrupt Request Peripheral Read register 12L)

割込み要求一括読出しレジスタ 12 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#59)

■ IRPR12L： アドレス 0431H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	OCUIR2	OCUIR3	OCUIR4	OCUIR5	OCUIR8	OCUIR9	予約	
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX

[bit7] OCUIR2 (OCU Interrupt Request 2) : アウトプットコンペア ch.2 割込み要求
[bit6] OCUIR3 (OCU Interrupt Request 3) : アウトプットコンペア ch.3 割込み要求
[bit5] OCUIR4 (OCU Interrupt Request 4) : アウトプットコンペア ch.4 割込み要求
[bit4] OCUIR5 (OCU Interrupt Request 5) : アウトプットコンペア ch.5 割込み要求
[bit3] OCUIR8 (OCU Interrupt Request 8) : アウトプットコンペア ch.8 割込み要求
[bit2] OCUIR9 (OCU Interrupt Request 9) : アウトプットコンペア ch.9 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.27 割込み要求一括読出しレジスタ 13 上位 ：IRPR13H (Interrupt Request Peripheral Read register 13H)

割込み要求一括読出しレジスタ 13 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#60)

■ IRPR13H： アドレス 0432H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BT0IR0	BT0IR1	SGIR2	予約				
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] BT0IR0 (BT0 Interrupt Request 0) : ベースタイマ ch.0 割込み要求 0

[bit6] BT0IR1 (BT0 Interrupt Request 1) : ベースタイマ ch.0 割込み要求 1

[bit5] SGIR2 (SG Interrupt Request 2) : サウンドジェネレータ 2 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.28 割込み要求一括読出しレジスタ 13 下位 ：IRPR13L (Interrupt Request Peripheral Read register 13L)

割込み要求一括読出しレジスタ 13 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#61)

■ IRPR13L： アドレス 0433H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BT1IR0	BT1IR1	SGIR3	XB_ECC_SE	BR_ECC_SE	予約		
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX

[bit7] BT1IR0 (BT1 Interrupt Request 0) : ベースタイマ ch.1 割込み要求 0
[bit6] BT1IR1 (BT1 Interrupt Request 1) : ベースタイマ ch.1 割込み要求 1
[bit5] SGIR3 (SG Interrupt Request 3) : サウンドジェネレータ 3 割込み要求
[bit4] XB_ECC_SE : XBS RAM シングルビットエラー発生割込み要求
[bit3] BR_ECC_SE : Backup RAM シングルビットエラー発生割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.29 割込み要求一括読出しレジスタ 14 上位 ：IRPR14H (Interrupt Request Peripheral Read register 14H)

割込み要求一括読出しレジスタ 14 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#62)

■ IRPR14H： アドレス 0434H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DMAC0IR	DMAC1IR	DMAC2IR	DMAC3IR	DMAC4IR	DMAC5IR	DMAC6IR	DMAC7IR
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit7] DMAC0IR (DMAC 0 Interrupt Request) : DMAC ch.0 割込み要求
[bit6] DMAC1IR (DMAC 1 Interrupt Request) : DMAC ch.1 割込み要求
[bit5] DMAC2IR (DMAC 2 Interrupt Request) : DMAC ch.2 割込み要求
[bit4] DMAC3IR (DMAC 3 Interrupt Request) : DMAC ch.3 割込み要求
[bit3] DMAC4IR (DMAC 4 Interrupt Request) : DMAC ch.4 割込み要求
[bit2] DMAC5IR (DMAC 5 Interrupt Request) : DMAC ch.5 割込み要求
[bit1] DMAC6IR (DMAC 6 Interrupt Request) : DMAC ch.6 割込み要求
[bit0] DMAC7IR (DMAC 7 Interrupt Request) : DMAC ch.7 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.30 割込み要求一括読出しレジスタ 14 下位 ：IRPR14L (Interrupt Request Peripheral Read register 14L)

割込み要求一括読出しレジスタ 14 下位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#62)

■ IRPR14L： アドレス 0435H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DMAC8IR	DMAC9IR	DMAC10IR	DMAC11IR	DMAC12IR	DMAC13IR	DMAC14IR	DMAC15IR
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit7] DMAC8IR (DMAC 8 Interrupt Request) : DMAC ch.8 割込み要求
[bit6] DMAC9IR (DMAC 9 Interrupt Request) : DMAC ch.9 割込み要求
[bit5] DMAC10IR (DMAC 10 Interrupt Request) : DMAC ch.10 割込み要求
[bit4] DMAC11IR (DMAC 11 Interrupt Request) : DMAC ch.11 割込み要求
[bit3] DMAC12IR (DMAC 12 Interrupt Request) : DMAC ch.12 割込み要求
[bit2] DMAC13IR (DMAC 13 Interrupt Request) : DMAC ch.13 割込み要求
[bit1] DMAC14IR (DMAC 14 Interrupt Request) : DMAC ch.14 割込み要求
[bit0] DMAC15IR (DMAC 15 Interrupt Request) : DMAC ch.15 割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.4.31 割込み要求一括読出しレジスタ 15 上位 ：IRPR15H (Interrupt Request Peripheral Read register 15H)

割込み要求一括読出しレジスタ 15 上位のビット構成は、以下のとおりです。

割込み要求を行っているペリフェラルを表示します。(割込みベクタ番号#15)

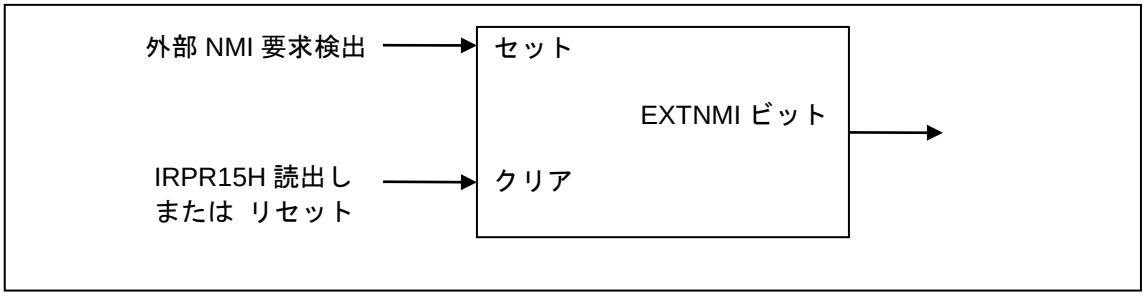
■ IRPR15H： アドレス 0436H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EXTNMI	XB_ECC_DE	BR_ECC_DE	予約				
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

[bit7] EXTNMI： 外部 NMI 要求

EXTNMI ビットは、外部 NMI 要求の検出によりセットされ、本レジスタのリードによりクリアされます。

スタンバイモード(電源遮断)からの復帰中は、内部リセットが発行されており、NMI 要求は保持できません。



[bit6] XB_ECC_DE : XBS RAM ダブルビットエラー発生割込み要求

[bit5] BR_ECC_DE : Backup RAM ダブルビットエラー発生割込み要求

各ビットの読出し値	意味
0	割込み要求は出ていません
1	割込み要求が出ています

16.5 動作説明

割込み要求一括読出しの動作について説明します。

各割込みハンドラ内において、各レジスタを読み出し、どのビットがセットされているかで、どの割込み要求が発生しているのかを判断します。

＜注意事項＞
外部割込み入力用には本レジスタ機能は用意されていません。外部割込み入力の EIRR0, EIRR1 レジスタを読み出してください。

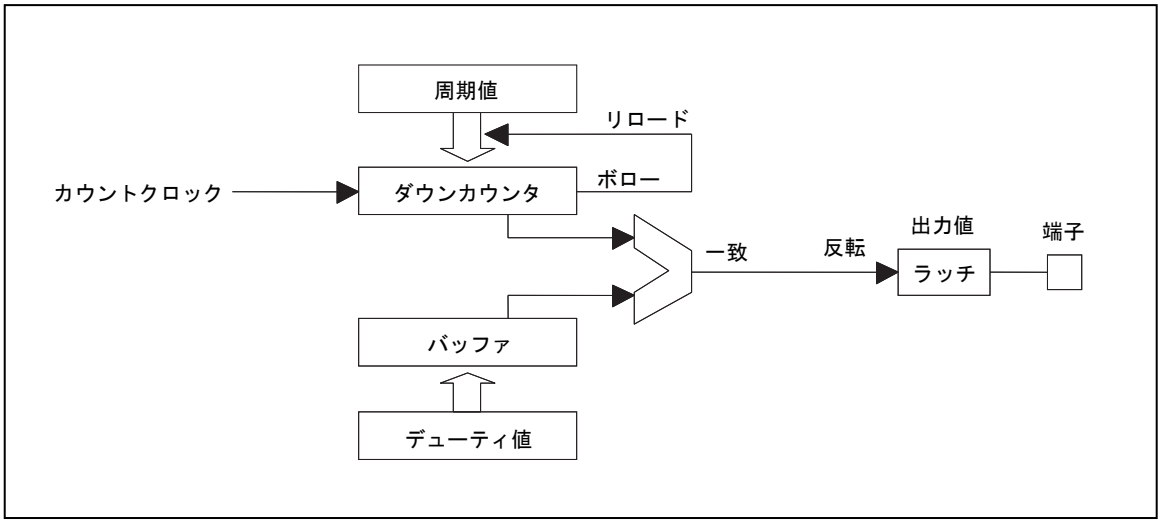
17. PPG

PPG について説明します。

17.1 概要

PPG の概要について説明します。

PPG (Programmable Pulse Generator) は、ワンショット (の矩形波) 出力や PWM (Pulse width modulation) 出力を行うときに使います。また、この PPG では周期とデューティをソフトウェアで自由に変更できますので、幅広い用途に使用できます。

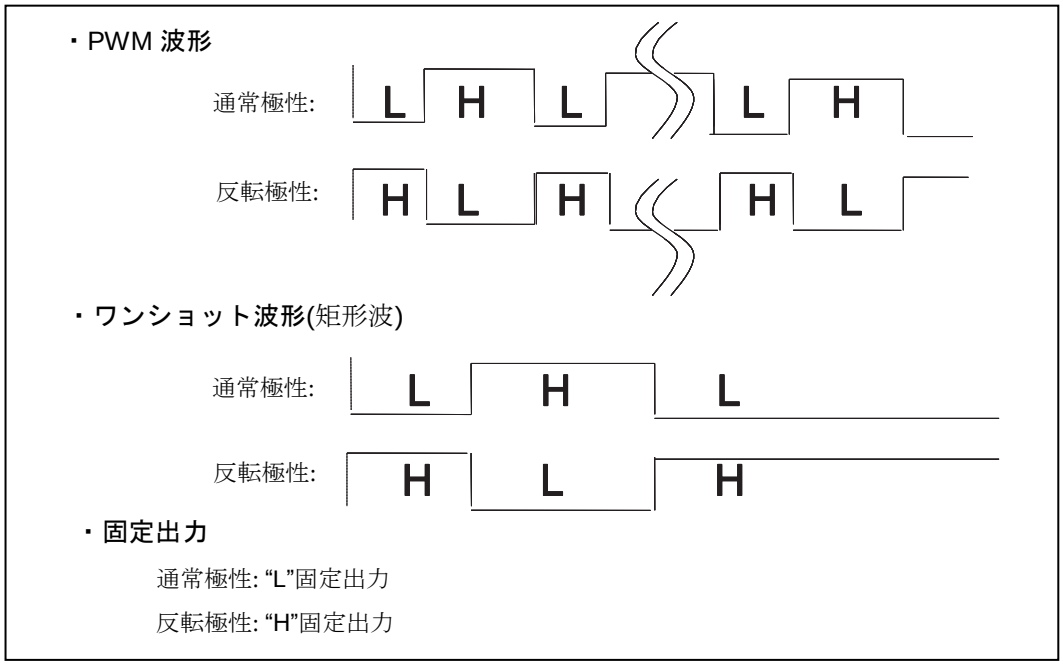


17.2 特長

PPG の特長について説明します。

- 個数 : 24 (出力 : 24 チャネル PPG0, PPG1, PPG2, PPG3, PPG4, PPG5, PPG6, PPG7, PPG8, PPG9, PPG10, PPG11, PPG12, PPG13, PPG14, PPG15, PPG16, PPG17, PPG18, PPG19, PPG20, PPG21, PPG22, PPG23)
- カウントクロック : 4 種類の中から選択(周辺クロック(PCLK)÷1, ÷4, ÷16, ÷64)
- 周期 :
  周期 = カウントクロック× (PCSR レジスタ値＋1)
  (例)カウントクロック= 16MHz(62.5ns), PCSR 値= 63999 の場合
  周期 = 62.5ns × (63999+1) = 4ms
- デューティ :
  デューティ = カウントクロック× (PDUT レジスタ値＋1)
- 出力波形 : 図 2-1 の 6 種類

図 17-1 出力波形



● 割込み要因 : 4 種類の中から選択

- ソフトウェアトリガおよびトリガ入力
- 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

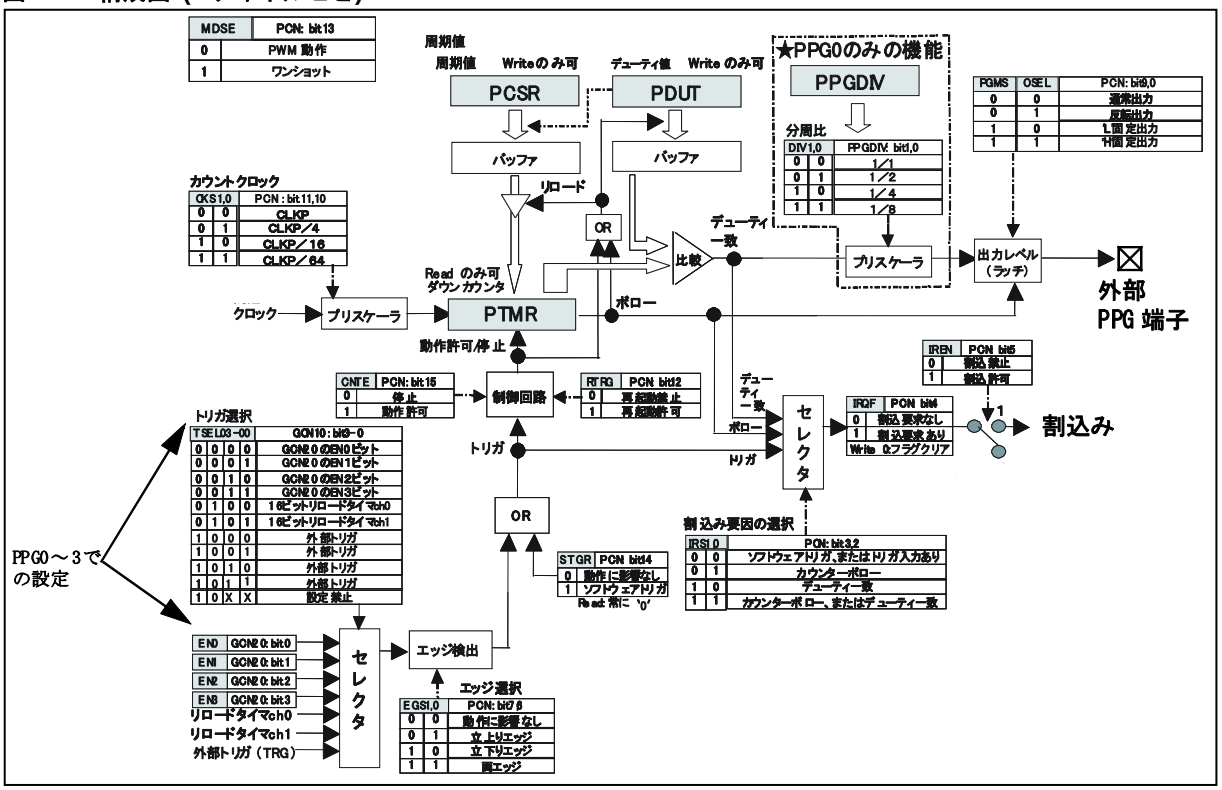
● 起動トリガ

- ソフトウェアトリガ (ソフトウェアトリガビットの設定)
- 内部トリガ :
  レジスタ書込みによるトリガ
  リロードタイマによるトリガ
- 外部トリガ

17.3 構成

PPG の構成について説明します。

図 17-2 構成図 (1 チャネルごと)



17.4 レジスタ

PPG のレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子表

表 17-1 ベースアドレス・外部端子表

チャネル	Base_addr	外部端子	
		PPG 出力	トリガ入力
0	0x026C	PPG0_0/PPG0_1/PPG0_2	TRG0
1	0x0274	PPG1_0/PPG1_1/PPG1_2/PPG1_3	
2	0x027C	PPG2_0/PPG2_1/PPG2_2	
3	0x0284	PPG3_0/PPG3_1/PPG3_2	
4	0x028C	PPG4_0/PPG4_1/PPG4_2	TRG1
5	0x0294	PPG5_0/PPG5_1/PPG5_2	
6	0x029C	PPG6_0/PPG6_1/PPG6_2	
7	0x02A4	PPG7_0/PPG7_1/PPG7_2	
8	0x02AC	PPG8_0/PPG8_1/PPG8_2	TRG2
9	0x02B4	PPG9_0/PPG9_1/PPG9_2	
10	0x02BC	PPG10_0/PPG10_1/PPG10_2	
11	0x0150	PPG11_0/PPG11_1	
12	0x0158	PPG12_0/PPG12_1	TRG3
13	0x0160	PPG13_0/PPG13_1	
14	0x0168	PPG14_0/PPG14_1	
15	0x0170	PPG15_0/PPG15_1	
16	0x0178	PPG16_0	TRG4
17	0x0180	PPG17_0	
18	0x0188	PPG18_0	
19	0x0190	PPG19_0	
20	0x0198	PPG20_0	TRG5
21	0x01A0	PPG21_0	
22	0x01A8	PPG22_0	
23	0x01B0	PPG23_0	

■ レジスタマップ

表 17-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0144	GCN13		予約	GCN23	ジェネラル制御レジスタ 13 ジェネラル制御レジスタ 23
0x0148	GCN14		予約	GCN24	ジェネラル制御レジスタ 14 ジェネラル制御レジスタ 24
0x014C	GCN15		予約	GCN25	ジェネラル制御レジスタ 15 ジェネラル制御レジスタ 25
0x0150	PTMR11		PCSR11		PPG タイマレジスタ 11 PPG 周期設定レジスタ 11
0x0154	PDUT11		PCN11		PPG デューティ設定レジスタ 11 PPG 制御ステータスレジスタ 11
0x0158	PTMR12		PCSR12		PPG タイマレジスタ 12 PPG 周期設定レジスタ 12
0x015C	PDUT12		PCN12		PPG デューティ設定レジスタ 12 PPG 制御ステータスレジスタ 12
0x0160	PTMR13		PCSR13		PPG タイマレジスタ 13 PPG 周期設定レジスタ 13
0x0164	PDUT13		PCN13		PPG デューティ設定レジスタ 13 PPG 制御ステータスレジスタ 13
0x0168	PTMR14		PCSR14		PPG タイマレジスタ 14 PPG 周期設定レジスタ 14
0x016C	PDUT14		PCN14		PPG デューティ設定レジスタ 14 PPG 制御ステータスレジスタ 14
0x0170	PTMR15		PCSR15		PPG タイマレジスタ 15 PPG 周期設定レジスタ 15
0x0174	PDUT15		PCN15		PPG デューティ設定レジスタ 15 PPG 制御ステータスレジスタ 15
0x0178	PTMR16		PCSR16		PPG タイマレジスタ 16 PPG 周期設定レジスタ 16
0x017C	PDUT16		PCN16		PPG デューティ設定レジスタ 16 PPG 制御ステータスレジスタ 16
0x0180	PTMR17		PCSR17		PPG タイマレジスタ 17 PPG 周期設定レジスタ 17
0x0184	PDUT17		PCN17		PPG デューティ設定レジスタ 17 PPG 制御ステータスレジスタ 17
0x0188	PTMR18		PCSR18		PPG タイマレジスタ 18 PPG 周期設定レジスタ 18
0x018C	PDUT18		PCN18		PPG デューティ設定レジスタ 18 PPG 制御ステータスレジスタ 18
0x0190	PTMR19		PCSR19		PPG タイマレジスタ 19 PPG 周期設定レジスタ 19
0x0194	PDUT19		PCN19		PPG デューティ設定レジスタ 19 PPG 制御ステータスレジスタ 19
0x0198	PTMR20		PCSR20		PPG タイマレジスタ 20 PPG 周期設定レジスタ 20
0x019C	PDUT20		PCN20		PPG デューティ設定レジスタ 20 PPG 制御ステータスレジスタ 20
0x01A0	PTMR21		PCSR21		PPG タイマレジスタ 21 PPG 周期設定レジスタ 21

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x01A4	PDUT21		PCN21		PPG デューティ設定レジスタ 21 PPG 制御ステータスレジスタ 21
0x01A8	PTMR22		PCSR22		PPG タイマレジスタ 22 PPG 周期設定レジスタ 22
0x01AC	PDUT22		PCN22		PPG デューティ設定レジスタ 22 PPG 制御ステータスレジスタ 22
0x01B0	PTMR23		PCSR23		PPG タイマレジスタ 23 PPG 周期設定レジスタ 23
0x01B4	PDUT23		PCN23		PPG デューティ設定レジスタ 23 PPG 制御ステータスレジスタ 23
0x025C	GCN10		予約	GCN20	ジェネラル制御レジスタ 10 ジェネラル制御レジスタ 20
0x0260	GCN11		予約	GCN21	ジェネラル制御レジスタ 11 ジェネラル制御レジスタ 21
0x0264	GCN12		予約	GCN22	ジェネラル制御レジスタ 12 ジェネラル制御レジスタ 22
0x0268	予約			PPGDIV	PPG0 出力分周設定レジスタ
0x026C	PTMR0		PCSR0		PPG タイマレジスタ 0 PPG 周期設定レジスタ 0
0x0270	PDUT0		PCN0		PPG デューティ設定レジスタ 0 PPG 制御ステータスレジスタ 0
0x0274	PTMR1		PCSR1		PPG タイマレジスタ 1 PPG 周期設定レジスタ 1
0x0278	PDUT1		PCN1		PPG デューティ設定レジスタ 1 PPG 制御ステータスレジスタ 1
0x027C	PTMR2		PCSR2		PPG タイマレジスタ 2 PPG 周期設定レジスタ 2
0x0280	PDUT2		PCN2		PPG デューティ設定レジスタ 2 PPG 制御ステータスレジスタ 2
0x0284	PTMR3		PCSR3		PPG タイマレジスタ 3 PPG 周期設定レジスタ 3
0x0288	PDUT3		PCN3		PPG デューティ設定レジスタ 3 PPG 制御ステータスレジスタ 3
0x028C	PTMR4		PCSR4		PPG タイマレジスタ 4 PPG 周期設定レジスタ 4
0x0290	PDUT4		PCN4		PPG デューティ設定レジスタ 4 PPG 制御ステータスレジスタ 4
0x0294	PTMR5		PCSR5		PPG タイマレジスタ 5 PPG 周期設定レジスタ 5
0x0298	PDUT5		PCN5		PPG デューティ設定レジスタ 5 PPG 制御ステータスレジスタ 5
0x029C	PTMR6		PCSR6		PPG タイマレジスタ 6 PPG 周期設定レジスタ 6
0x02A0	PDUT6		PCN6		PPG デューティ設定レジスタ 6 PPG 制御ステータスレジスタ 6
0x02A4	PTMR7		PCSR7		PPG タイマレジスタ 7 PPG 周期設定レジスタ 7
0x02A8	PDUT7		PCN7		PPG デューティ設定レジスタ 7 PPG 制御ステータスレジスタ 7

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x02AC	PTMR8		PCSR8		PPG タイマレジスタ 8 PPG 周期設定レジスタ 8
0x02B0	PDUT8		PCN8		PPG デューティ設定レジスタ 8 PPG 制御ステータスレジス 8
0x02B4	PTMR9		PCSR9		PPG タイマレジスタ 9 PPG 周期設定レジスタ 9
0x02B8	PDUT9		PCN9		PPG デューティ設定レジスタ 9 PPG 制御ステータスレジスタ 9
0x02BC	PTMR10		PCSR10		PPG タイマレジスタ 10 PPG 周期設定レジスタ 10
0x02C0	PDUT10		PCN10		PPG デューティ設定レジスタ 10 PPG 制御ステータスレジスタ 10

17.4.1 PPG 周期設定レジスタ ：PCSR

PPG 周期設定レジスタ (PCSR)のビット構成について示します。

PPG の周期を設定するためのレジスタです。

■ PCSR： アドレス Base_addr + 02H (アクセス: ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	S15	S14	S13	S12	S11	S10	S9	S8
初期値	X	X	X	X	X	X	X	X
属性	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	S7	S6	S5	S4	S3	S2	S1	S0
初期値	X	X	X	X	X	X	X	X
属性	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W

- PPG 周期設定レジスタはバッファ付きです。バッファからカウンタへ値の転送は、カウンタのボロー発生時に自動で行われます。
- PPG 周期設定レジスタの書換え後は、必ず PPG デューティ設定レジスタ(PDUT)の設定を行ってください。
- PPG 周期設定レジスタは、必ずハーフワード(16 ビット)またはワード(32 ビット)でアクセスしてください。(「17.9. 注意事項」参照)

17.4.2 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)のビット構成について示します。

PPG の出力波形のデューティを設定するためのレジスタです。

■ PDUT： アドレス Base_addr + 04H (アクセス: ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	D15	D14	D13	D12	D11	D10	D9	D8
初期値	X	X	X	X	X	X	X	X
属性	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	X	X	X	X	X	X	X	X
属性	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W

- PPG デューティ設定レジスタはバッファ付きです。バッファからカウンタへ値の転送は、カウンタのボロー発生時に自動で行われます。
- PPG デューティ設定レジスタの設定値には、PPG 周期設定レジスタ(PCSR)の設定より小さな値を設定してください。
  (「17.9 注意事項」参照)
- PPG デューティ設定レジスタの設定値を PPG 周期設定レジスタ(PCSR)の設定と同じ値に設定した場合、下記のようになります。
  - ☐ 通常極性時(OSEL="0")には常に"H"を出力します。
  - ☐ 反転極性時(OSEL="1")には常に"L"を出力します。
    (OSEL ビットは、PPG 制御ステータスレジスタ(PCN)内の出力極性指定ビットです。)
- PPG デューティ設定レジスタは、必ずハーフワード(16 ビット)またはワード(32 ビット) でアクセスしてください。
  (「17.9 注意事項」参照)

17.4.3 PPG 制御ステータスレジスタ ：PCN

PPG 制御ステータスレジスタ (PCN)のビット構成について示します

PPG の動作制御および状態を表示するレジスタです。

■ PCN： アドレス Base_addr + 06H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	CNTE	STRG	MDSE	RTRG	CKS1	CKS0	PGMS	—
初期値	0	0	0	0	0	0	0	—
属性	R/W	R0,W	R/W	R/W	R/W	R/W	R/W	R1,WX
動作中の書換え	○	○	×	×	×	×	○	×

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EGS1	EGS0	IREN	IRQF	IRS1	IRS0	予約	OSEL
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R(RM1),W	R/W	R/W	R/W0	R/W
動作中の書換え	×	×	○	○	×	×	×	×

○：書換え可能, ×：書換え不可 (「17.9. 注意事項」参照)

[bit15] CNTE： タイマ動作許可

CNTE	動作
0	停止
1	動作

PPG の動作を許可するビットです。

[bit14] STRG： ソフトウェアトリガ

STRG	動作
0	書込みによる動作への影響なし (読出し値は常に"0")
1	ソフトウェアトリガ起動

ソフトウェアトリガビットを"1"にすると、ソフトウェアトリガが発生し、PPG が起動します。ソフトウェアトリガは、EN ビットによるトリガの発生とは別に PPG を起動します。

[bit13] MDSE： モード選択

MDSE	モード
0	PWM 動作
1	ワンショット動作

- モード選択ビットを"0"にすると PWM 動作となり、連続してパルスを出力します。
- モード選択ビットを"1"にすると、1 回だけパルスを出力します。

[bit12] RTRG : 再起動許可

RTRG	動作
0	再起動禁止
1	再起動許可

再起動許可ビットを"1"にすると、トリガ (ソフト/内部/外部) により再起動を許可します。

[bit11, bit10] CKS1, CKS0 : カウンタクロック選択

CKS1	CKS0	ダウンカウンタのカウントクロックの選択
0	0	周辺クロック(PCLK)
0	1	周辺クロックの 4 分周
1	0	周辺クロックの 16 分周
1	1	周辺クロックの 64 分周

[bit9] PGMS : PPG 出力マスク選択

PGMS	動作
0	出力マスクなし
1	出力マスクあり (出力"L"レベル固定：OSEL="0"のとき)

- PPG 出力マスク選択ビットを"1"にすると、モード設定, 周期設定, デューティ設定にかかわらず PPG の出力を"L"または"H"に固定できます。
- 出力レベルは出力極性指定ビット(PCNn:OSEL)で指定できます。(n = 0～23)

＜注意事項＞

本ビットを"1"から"0"に設定し、PPG 出力マスクを解除する場合には、周期のはじめから、デューティ一致となる前までの間に設定を行ってください。

[bit8] - : 未定義

常に"1"が読み出されます。書込みは動作に影響ありません。

[bit7, bit6] EGS1, EGS0 : トリガ入力エッジ選択

EGS1	EGS0	選択エッジ
0	0	書込みによる動作への影響はありません
0	1	立上りエッジ
1	0	立下りエッジ
1	1	両エッジ (立上りエッジ, もしくは, 立下りエッジ)

各 PPG のトリガ指定ビット(GCN10/11/12/13/14/15:TSEL3/2/1/0) にて選択したトリガ入力に対して、 起動要因となるエッジをトリガ入力エッジ選択ビット(EGS[1:0])で選択します。

[bit5] IREN : 割込み要求許可

IREN	動作
0	割込み要求禁止
1	割込み要求許可

[bit4] IRQF : 割込み要求フラグ

IRQF	読出し動作	書込み動作
0	割込み要求なし	割込み要求フラグをクリア
1	割込み要求あり	書込みによる動作への影響はありません

割込み要求フラグ(IRQF)が"1"になるのと、フラグへの"0"の書込みが同時に行われた場合、割込み要求フラグのセット(IRQF="1") が優先します。

[bit3, bit2] IRS1, IRS0 : 割込み要因選択

IRS1	IRS0	選択
0	0	ソフトウェアトリガ, または, トリガ入力
0	1	カウンタのボロー発生 (周期一致)
1	0	カウンタがデューティ値と一致発生
1	1	カウンタのボロー発生 (周期一致), または, カウンタがデューティ値と一致発生

割込み要求を発生する動作を選択します。

[bit1] 予約

必ず"0"を設定してください。

[bit0] OSEL : PPG 出力極性指定

OSEL	動作
0	通常極性
1	反転極性

PPG 出力マスク選択ビット(PCNn:PGMS)に"1"を設定している場合、出力極性指定ビット(OSEL)を"0"にすると出力が"L"固定となり、出力極性指定ビットを"1"にすると出力が"H"固定となります。(n=0～23)

17.4.4　ジェネラル制御レジスタ 10～13 : GCN10～GCN13

ジェネラル制御レジスタ 10～13 (GCN10～GCN13)のビット構成について示します。

PPG0～PPG15 のトリガ入力を選択するレジスタです。
GCN10 : PPG0～PPG3
GCN11 : PPG4～PPG7
GCN12 : PPG8～PPG11
GCN13 : PPG12～PPG15

■ GCN10： アドレス 025C$_H$ (アクセス: ハーフワード)

■ GCN11： アドレス 0260$_H$ (アクセス: ハーフワード)

■ GCN12： アドレス 0264$_H$ (アクセス: ハーフワード)

■ GCN13： アドレス 0144$_H$ (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TSEL3[3:0]				TSEL2[3:0]			
初期値	0	0	1	1	0	0	1	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TSEL1[3:0]				TSEL0[3:0]			
初期値	0	0	0	1	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit12] TSEL3[3:0] : PPG3・PPG7・PPG11・PPG15 のトリガ指定

[bit11～bit8] TSEL2[3:0] : PPG2・PPG6・PPG10・PPG14 のトリガ指定

[bit7～bit4] TSEL1[3:0] : PPG1・PPG5・PPG9・PPG13 のトリガ指定

[bit3～bit0] TSEL0[3:0] : PPG0・PPG4・PPG8・PPG12 のトリガ指定

TSEL0[3:0] (PPG0/4/8/12) TSEL1[3:0] (PPG1/5/9/13) TSEL2[3:0] (PPG2/6/10/14) TSEL3[3:0] (PPG3/7/11/15)				**起動トリガ指定**
0	0	0	0	EN0 ビット(GCN20/21/22/23 レジスタ)
0	0	0	1	EN1 ビット(GCN20/21/22/23 レジスタ)
0	0	1	0	EN2 ビット(GCN20/21/22/23 レジスタ)
0	0	1	1	EN3 ビット(GCN20/21/22/23 レジスタ)
0	1	0	0	16 ビットリロードタイマ 0
0	1	0	1	16 ビットリロードタイマ 1
1	0	0	0	外部トリガ
1	0	0	1	外部トリガ
1	0	1	0	外部トリガ
1	0	1	1	外部トリガ
1	1	X	X	設定禁止
上記以外				設定禁止 (「17.9. 注意事項」参照)

指定した起動トリガにおいて，トリガ入力エッジ選択ビット(PCNn:EGS[1:0]) (n=0～15)で指定したエッジを検出すると選択した PPG0～PPG15 が起動します。

17.4.5　ジェネラル制御レジスタ 14, 15 : GCN14, GCN15

ジェネラル制御レジスタ 14, 15 (GCN14, GCN15)のビット構成について示します。

PPG16～PPG23 のトリガ入力を選択するレジスタです。
GCN14 : PPG16～PPG19
GCN15 : PPG20～PPG23

■ GCN14 : アドレス 0148H (アクセス: ハーフワード)

■ GCN15 : アドレス 014CH (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TSEL3[3:0]				TSEL2[3:0]			
初期値	0	0	1	1	0	0	1	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TSEL1[3:0]				TSEL0[3:0]			
初期値	0	0	0	1	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit12] TSEL3[3:0] : PPG19・PPG23 のトリガ指定
[bit11～bit8] TSEL2[2:0] : PPG18・PPG22 のトリガ指定
[bit7～bit4] TSEL1[3:0] : PPG17・PPG21 のトリガ指定
[bit3～bit0] TSEL0[3:0] : PPG16・PPG20 のトリガ指定

TSEL0[3:0] (PPG16/20) TSEL1[3:0] (PPG17/21) TSEL2[3:0] (PPG18/22) TSEL3[3:0] (PPG19/23)				**起動トリガ指定**
0	0	0	0	EN0 ビット(GCN24/25 レジスタ)
0	0	0	1	EN1 ビット(GCN24/25 レジスタ)
0	0	1	0	EN2 ビット(GCN24/25 レジスタ)
0	0	1	1	EN3 ビット(GCN24/25 レジスタ)
0	1	0	0	16 ビットリロードタイマ 2
0	1	0	1	16 ビットリロードタイマ 3
1	0	0	0	外部トリガ
1	0	0	1	外部トリガ
1	0	1	0	外部トリガ
1	0	1	1	外部トリガ
1	1	X	X	設定禁止
上記以外				設定禁止 (「17.9 注意事項」参照)

指定した起動トリガにおいて、トリガ入力エッジ選択ビット(PCNn:EGS[1:0]) (n=16～23)で指定したエッジを検出すると選択した PPG16～PPG23 が起動します。

17.4.6　ジェネラル制御レジスタ 20～25 : GCN20～GCN25

ジェネラル制御レジスタ 20～25 (GCN20～GCN25)のビット構成について示します。

ソフトウェアによって PPG0～PPG23 の内部トリガ用のレベルを発生させるためのレジスタです。
GCN20 : PPG0～PPG3
GCN21 : PPG4～PPG7
GCN22 : PPG8～PPG11
GCN23 : PPG12～PPG15
GCN24 : PPG16～PPG19
GCN25 : PPG20～PPG23

■ GCN20： アドレス 025F$_H$ (アクセス: バイト)

■ GCN21： アドレス 0263$_H$ (アクセス: バイト)

■ GCN22： アドレス 0267$_H$ (アクセス: バイト)

■ GCN23： アドレス 0147$_H$ (アクセス: バイト)

■ GCN24： アドレス 014B$_H$ (アクセス: バイト)

■ GCN25： アドレス 014F$_H$ (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	予約	予約	予約	EN3	EN2	EN1	EN0
初期値	0	0	0	0	0	0	0	0
属性	R/W0	R/W0	R/W0	R/W0	R/W	R/W	R/W	R/W

[bit7～bit4] 予約
必ず"0"を書き込んでください。(「17.9 注意事項」参照)

[bit3] EN3： トリガ入力
[bit2] EN2： トリガ入力
[bit1] EN1： トリガ入力
[bit0] EN0： トリガ入力

EN0, EN1, EN2, EN3	内部トリガ EN0, EN1, EN2, EN3
0	レベルを"L"にする
1	レベルを"H"にする

- 内部トリガのレベルを設定します。
- ジェネラル制御レジスタ(GCN10～GCN15)のトリガ指定ビット(TSEL3, TSEL2, TSEL1, TSEL0) で EN トリガ入力(EN0, EN1, EN2, EN3)のどれかを選択した場合、その選択した EN が PPG のトリガ入力ビットになります。
- PPG 制御ステータスレジスタのトリガ入力エッジ選択ビット(EGS[1:0]) で選択した状態を、トリガ入力ビット(選択した EN0, EN1, EN2, EN3)でソフトウェアによって発生させると、それが起動トリガとなり PPG が起動します。

17.4.7 PPG タイマレジスタ ：PTMR

PPG タイマレジスタ (PTMR)のビット構成について示します。

PPG0～PPG23 のカウント値を読み出すためのレジスタです。

■ PTMR： アドレス Base_addr + 00H (アクセス: ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	T15	T14	T13	T12	T11	T10	T9	T8
初期値	1	1	1	1	1	1	1	1
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	T7	T6	T5	T4	T3	T2	T1	T0
初期値	1	1	1	1	1	1	1	1
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

- 16 ビットダウンカウンタのカウント値を読み出すことができます。
- PPG タイマレジスタ(PTMRn) は、バイトアクセスした場合正しく読み出せません。(n = 0～23)

17.4.8　PPG0 出力分周設定レジスタ ：PPGDIV

PPG0 出力分周設定レジスタ (PPGDIV)のビット構成について示します。

PPG0 の出力分周比を設定します。

■ PPGDIV： アドレス 026BH (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	－	－	－	－	－	－	DIV1	DIV0
初期値	－	－	－	－	－	－	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W

[bit7～bit2] - : 未定義
常に"1"が読み出されます。書込みは動作に影響ありません。

[bit1, bit0] DIV1,DIV0 : 分周設定

DIV1	DIV0	分周比
0	0	1/1
0	1	1/2
1	0	1/4
1	1	1/8

PPG0 出力の分周比を設定します。

＜注意事項＞
1/2, 1/4, 1/8 分周設定を使用する場合、以下の制限があります。

- 出力波形はデューティ 50%固定になります。
- ワンショット動作(PCN:MDSE=1)は設定禁止です。
- PPG 出力反転機能(PCN:OSEL=1)は設定禁止です。
- PPG 出力固定状態(PCN:PGMS,OSEL=01,10,11)は設定禁止です。
- PCSR=PDUT のときは、設定禁止です。

17.5 動作説明

PPG の動作について説明します。

PPG (Programmable Pulse Generator)が 24 個あり、独立/連動してプログラマブルなパルス出力をします。

各動作モードについて以下に説明をします。

17.5.1. PWM 動作

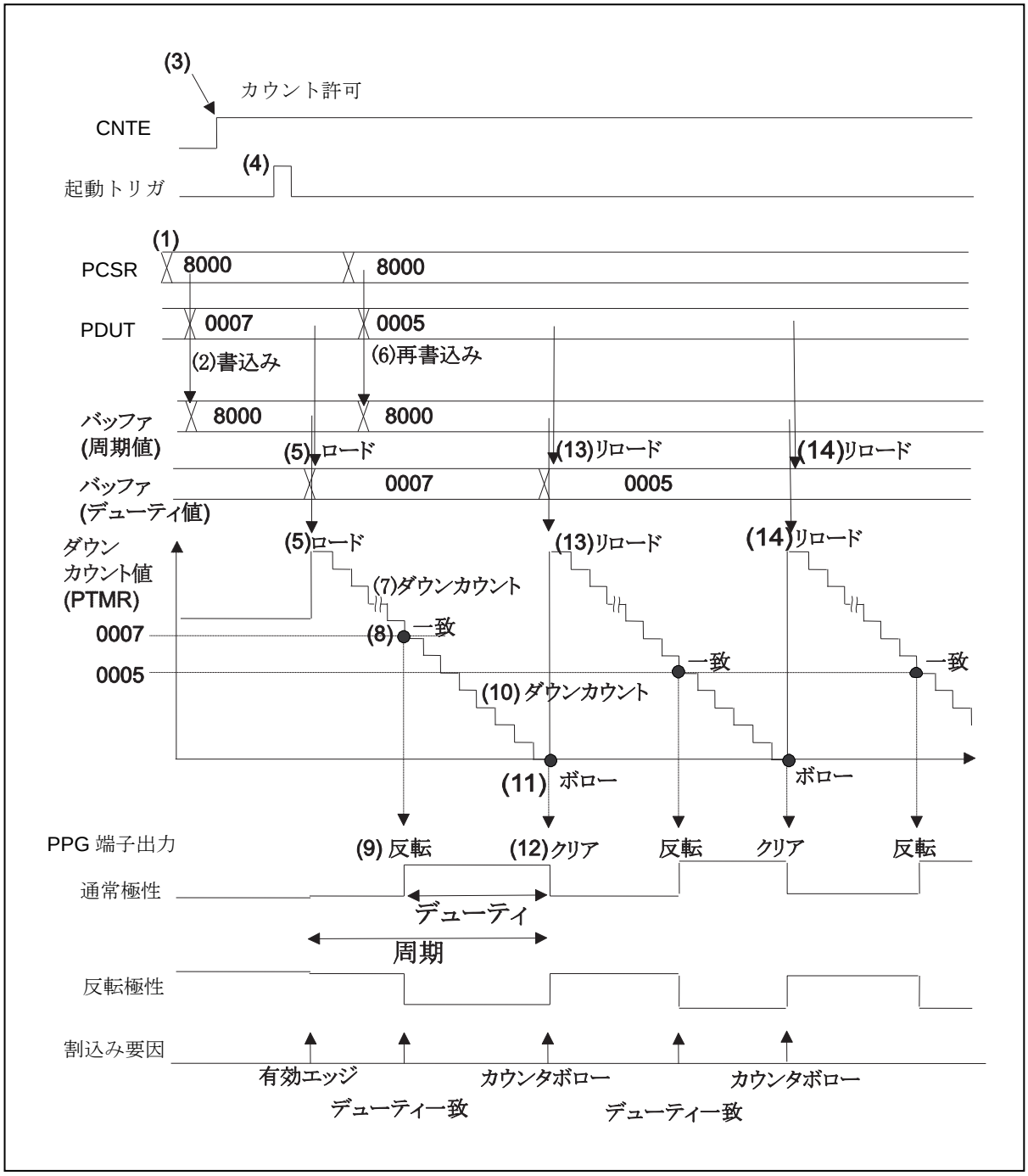
17.5.2. ワンショット動作

17.5.3. 再起動動作

17.5.1 PWM 動作

PPG の PWM 動作について説明します。

PWM 動作では、PPG 端子からプログラマブルにデューティ可変のパルスを出力します。



(1) 周期値の書込み
(2) デューティ値の書込み，周期値のバッファへの転送
(3) PPG 動作許可
(4) 起動トリガ発生

(5) 周期値のロード, デューティ値のロード
(6) デューティ値の再書込み, 周期値のバッファへの転送
(7) カウンタのダウンカウント
(8) ダウンカウンタ値とデューティ値の一致
(9) PPG 端子出力レベルの反転
(10) カウンタのダウンカウント
(11) カウンタのボロー発生
(12) PPG 端子出力レベルのクリア (通常レベルに戻る)
(13) 周期値のリロード
(14) デューティ値のリロード
(15) (6)から(14)までの動作を繰り返します (「17.9. 注意事項」参照)

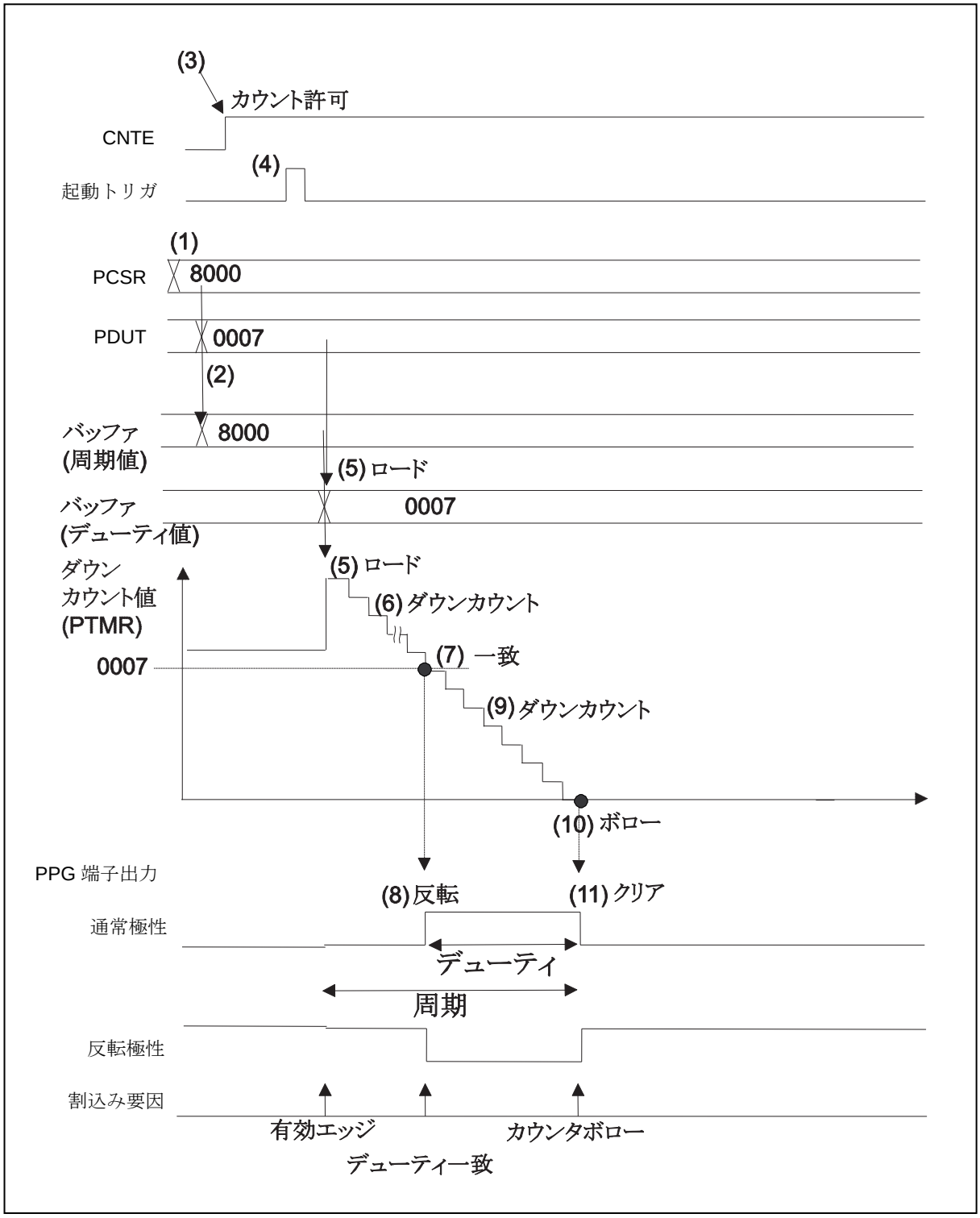
式

周期 = {周期値(PCSR)＋1}×カウントクロック
デューティ= {デューティ値(PDUT)＋1}×カウントクロック
パルスが出るまでの幅 = {周期値(PCSR) － デューティ値(PDUT)}×カウントクロック

17.5.2 ワンショット動作

PPG のワンショット動作について説明します。

ワンショット動作では、PPG 端子からワンショットのパルスを出力します。



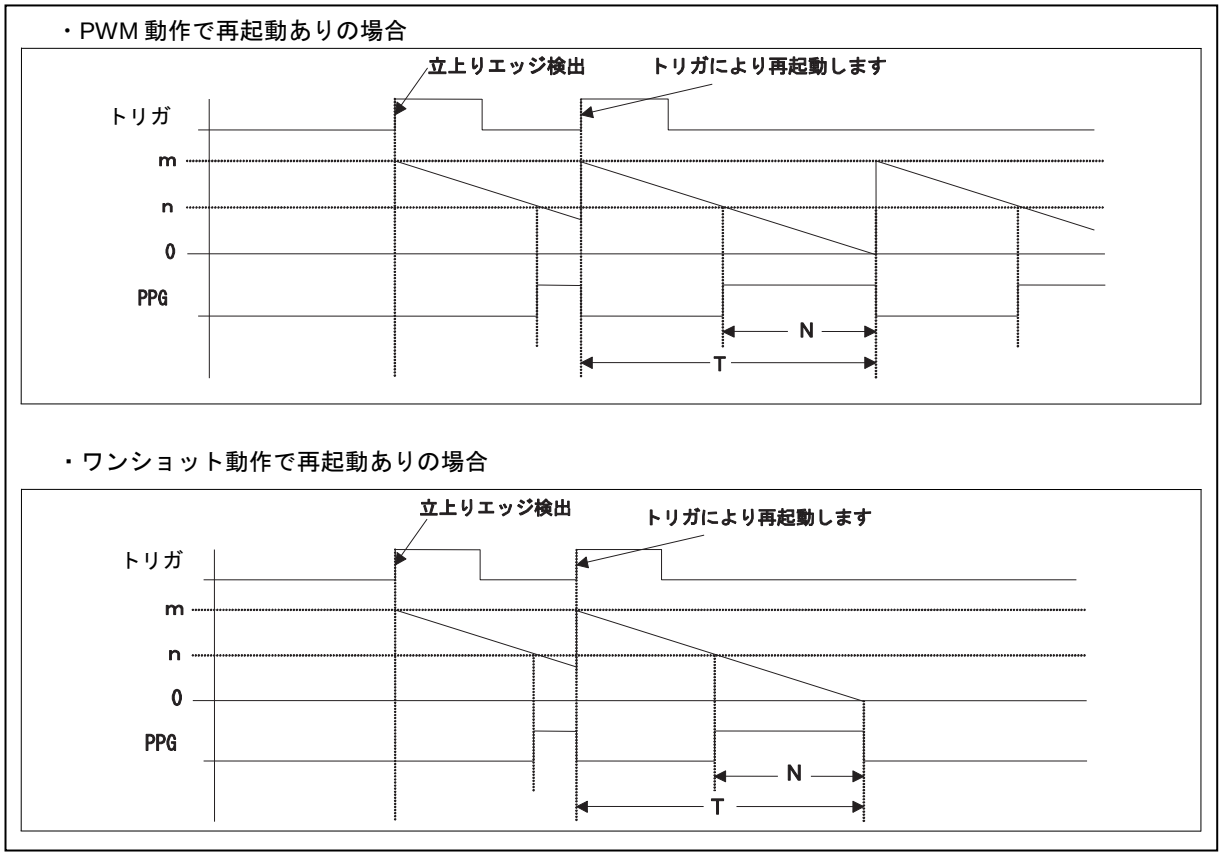
(1) 周期値の書込み
(2) デューティ値の書込み, 周期値のバッファへの転送
(3) PPG 動作許可
(4) 起動トリガ発生
(5) 周期値のロード, デューティ値のロード
(6) カウンタのダウンカウント
(7) ダウンカウンタ値とデューティ値の一致
(8) PPG 端子出力レベルの反転
(9) カウンタのダウンカウント
(10) カウンタのボロー発生
(11) PPG 端子出力レベルのクリア (通常レベルに戻る)
(12) 上記で動作を終了します。
(「17.9. 注意事項」参照)

17.5.3 再起動動作

PPG の再起動動作について説明します。

再起動の動作を下記に示します。

N = デューティ, T = 周期, m = 周期設定レジスタ(PCSR)値,
n = デューティ設定レジスタ(PDUT)値



再起動なしの場合は、PWM 動作, ワンショット動作とも 2 回目以降のトリガが無効となります。
(2 回目以降でも、ダウンカウンタが停止した後のトリガは有効です)

17.6 設定

PPG の設定について説明します。

表 17-3 PPG を動作させるために必要な設定

設定	設定レジスタ	設定方法
周期値, デューティ値の設定	PPG 周期設定 (PCSR0～PCSR23) PPG デューティ設定 (PDUT0～PDUT23)	17.7.1 参照
PPG 動作許可	PPG 制御ステータス (PCN0～PCN23)	17.7.2 参照
動作モードの選択 (PWM/ワンショット)		17.7.3 参照
再起動許可		17.7.4 参照
カウントクロックの選択		17.7.5 参照
PPG 出力マスクの選択		17.7.6 参照
トリガ選択 ・ソフトトリガ ・外部トリガ ・内部トリガ (リロードタイマ, GCN20/21/22/23/24/25:EN ビット)	PPG 制御ステータス (PCN0～PCN23)	17.7.7 参照
	TRG 端子からのトリガ入力	
	ジェネラル制御 10/11/12/13/14/15 (GCN10/11/12/13/14/15)	
出力極性指定	PPG 制御ステータス (PCN0～PCN23)	17.7.8 参照
PPG 端子出力設定	端子をペリフェラル出力に設定してください。 設定方法は『I/O ポート』の章を参照してください。	
トリガの発生 ・外部トリガ ・ソフトウェアトリガ	TRG 端子からのトリガ入力	17.4.3 参照
	PPG 制御ステータス (PCN0～PCN23)	
・リロードタイマ	『リロードタイマ』の章を参照してください。	
・GCN20/21/22/23/24/25:EN ビット	ジェネラル制御 20/21/22/23/24/25 (GCN20/21/22/23/24/25)	17.4.6 参照

表 17-4 PPG を停止させるために必要な設定

設定	設定レジスタ	設定方法
PPG 停止ビットの設定	PPG 制御ステータス (PCN0～PCN23)	17.7.11 参照

表 17-5 出力レベルを固定させるために必要な設定

設定	設定レジスタ	設定方法
出力極性指定	PPG 制御ステータス (PCN0～PCN23)	17.7.8 参照
PPG 出力マスクの選択		17.7.6 参照
周期値 = デューティ値の設定	PPG デューティ設定 (PDUT0～PDUT23)	17.7.6 参照

表 17-6 PPG 割込みを行うために必要な設定

設定	設定レジスタ	設定方法
PPG 割込みベクタ, PPG 割込みレベルの設定	『割込み制御 (割込みコントローラ)』の章を参照してください。	17.7.12 参照
PPG 割込み要因選択 (起動トリガ発生, ボロー発生, デューティ一致)	PPG 制御ステータス (PCN0～PCN23)	17.7.13 参照
PPG 割込み設定 割込み要求のクリア 割込み要求の許可		17.7.14 参照

17.7 Q&A

PPG の Q&A について説明します。

17.7.1 周期とデューティの設定 (書換え) 方法は?

17.7.2 PPG 動作を許可/停止するには?

17.7.3 PPG の動作モード (PWM 動作/ワンショット動作) を設定するには?

17.7.4 再起動させるには?

17.7.5 カウントクロックの種類と選択方法は?

17.7.6 PPG 端子出力のレベルを固定させるには?

17.7.7 起動トリガの種類と選択方法は?

17.7.8 出力極性を反転させるには?

17.7.9 端子を PPG 出力端子にするには?

17.7.10 起動トリガの発生方法は?

17.7.11 PPG 動作を停止するには?

17.7.12 割込み関連レジスタは?

17.7.13 割込みの種類と選択方法は?

17.7.14 割込みを許可/禁止/クリアするには?

17.7.1 周期とデューティの設定 (書換え) 方法は?

周期とデューティの設定(書換え)方法について説明します。

周期値の設定, デューティ値の設定

- 周期値は PPG 周期設定レジスタ PCSRn に設定します。(n=0～23)
- デューティ値は PPG デューティ設定レジスタ PDUTn に設定します。(n=0～23)
- PPG 周期設定レジスタ, PPG デューティ設定レジスタのそれぞれが、バッファを持っているので書込みタイミングを意識する必要がありません。

式

PCSR レジスタ値 = {周期/カウントクロック} －1
PDUT レジスタ値 = {"H"幅(デューティ)*/カウントクロック} －1

*: 通常極性の場合 (OSEL=0)

設定可能な範囲

PCSR レジスタ値 = PDUT レジスタ値 ～ $FFFF_H$(65535)
PDUT レジスタ値 = 0 ～ PCSR レジスタ値

＜注意事項＞

周期を設定した後、必ずデューティを設定してください。(「17.9. 注意事項」参照)

17.7.2 PPG 動作を許可/停止するには?

PPG 動作の許可/停止について説明します。

PPG の動作許可

PPG 動作許可ビット(PCNn:CNTE)で行ってください。(n=0～23)

制御	PPG 動作許可ビット (CNTE)
PPG 動作を停止させるには	"0"にする
PPG 動作を許可するには	"1"にする

PPG 動作許可は PPG を起動する前にしてください。
(「17.9. 注意事項」参照)

17.7.3 PPG の動作モード (PWM 動作/ワンショット動作) を設定するには?

PPG の動作モード (PWM 動作/ワンショット動作) の設定について説明します。

動作モードの選択は、モード選択ビット(PCNn:MDSE)で行ってください。(n=0～23)

動作モード	モード選択ビット (MDSE)
PWM 動作にするには	"0"にする
ワンショット動作にするには	"1"にする

(「17.9. 注意事項」参照)

17.7.4　再起動させるには?

再起動について説明します。

再起動許可

PPG 動作中に PPG の再起動を許可できます。

設定は、再起動許可ビット(PCNn:RTRG)で行います。(n=0～23)
(「17.9. 注意事項」参照)

17.7.5　カウントクロックの種類と選択方法は?

カウントクロックの種類と選択方法について説明します。

カウントクロックの選択

カウントクロックは下表の 4 種類の中から選択できます。

カウントクロック選択ビット(PCNn:CKS[1:0])で行います。(n=0～23)

カウントクロック	カウントクロック選択ビット		(例) 周辺クロック(PCLK)=16MHz 時	
	CKS1	CKS0	カウントクロック	周期(1～FFFF$_H$)
PCLK	0	0	16MHz	125.0ns～4.096ms
PCLK/4	0	1	4MHz	500ns～16.384ms
PCLK/16	1	0	1MHz	2.0µs～65.536ms
PCLK/64	1	1	250kHz	8.0µs～262.144ms

(「17.9. 注意事項」参照)

17.7.6 PPG 端子出力のレベルを固定させるには?

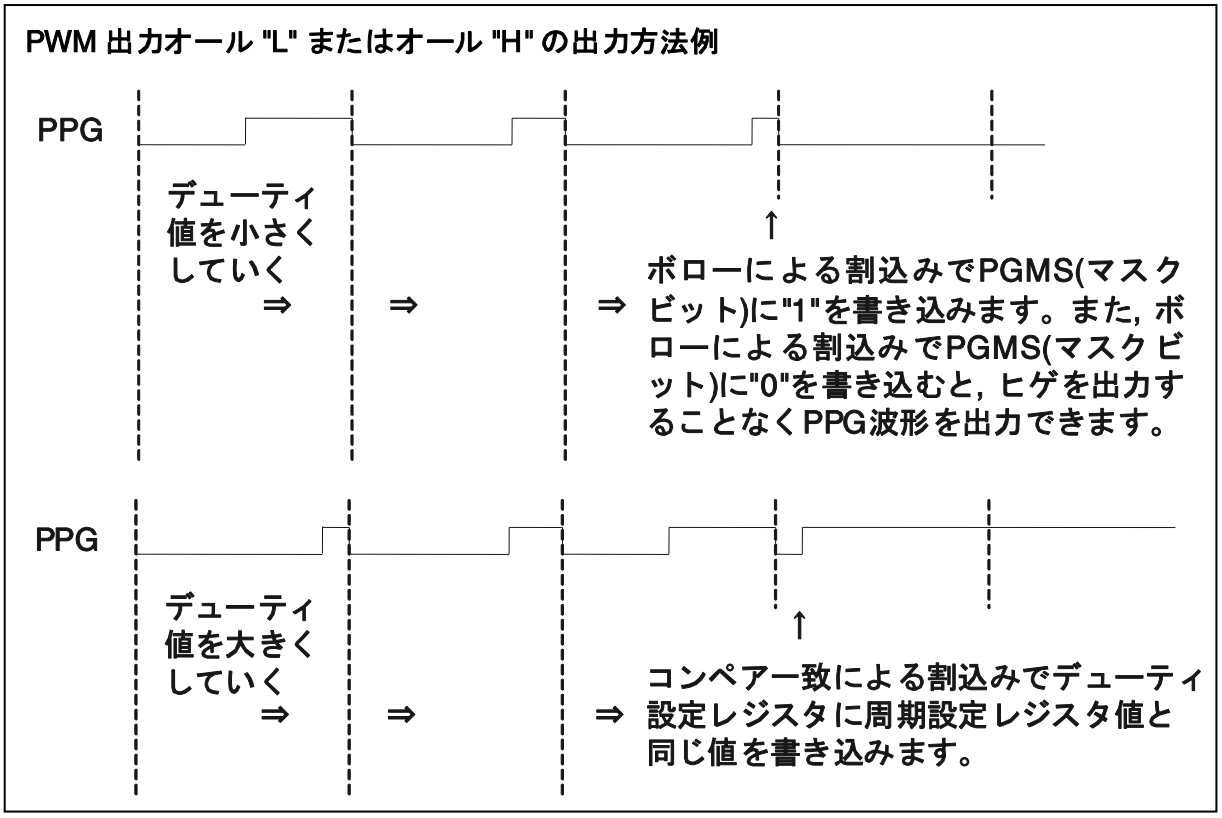
PPG 端子出力のレベルの固定について説明します。

PPG 出力マスクの選択

PPG の端子出力のレベルを固定できます。

設定は、PPG 出力マスク選択ビット (PCNn:PGMS) と、デューティ値 (PDUT) で行います。(n=0～23)

PPG 端子出力	PPG 出力極性指定ビット (OSEL)	設定方法
通常極性時に"L"レベル固定するには	"0"のとき	PPG 出力マスク選択ビット(PGMS)を"1"にする
通常極性時に"H"レベル固定するには	"0"のとき	周期値 (PCSR) = デューティ値 (PDUT)を設定
反転極性時に"H"レベル固定するには	"1"のとき	PPG 出力マスク選択ビット(PGMS)を"1"にする
反転極性時に"L"レベル固定するには	"1"のとき	周期値 (PCSR) = デューティ値 (PDUT) を設定



17.7.7 起動トリガの種類と選択方法は?

起動トリガの種類と選択方法について説明します。

■ 内部トリガの選択

- ソフトトリガは常に有効です。
- PPG0～PPG3 の内部トリガは GCN20 レジスタ,
  PPG4～PPG7 の内部トリガは GCN21 レジスタ,
  PPG8～PPG11 の内部トリガは GCN22 レジスタ,
  PPG12～PPG15 の内部トリガは GCN23 レジスタ,
  PPG16～PPG19 の内部トリガは GCN24 レジスタ,
  PPG20～PPG23 の内部トリガは GCN25 レジスタです。
- 内部トリガの設定は、
  GCN10 レジスタ(PPG0~PPG3)・
  GCN11 レジスタ(PPG4~PPG7)・
  GCN12 レジスタ(PPG8~PPG11)・
  GCN13 レジスタ(PPG12~PPG15)・
  GCN14 レジスタ(PPG16~PPG19)・
  GCN15 レジスタ(PPG20~PPG23)
  の TSEL0/TSEL1/TSEL2/TSEL3 で行います。

PPG0～PPG3 での設定を以下に示します。

内部トリガ	PPG0 での例 (GCN10: TSEL0[3:0]の設定値)	PPG1 での例 (GCN10: TSEL1[3:0]の設定値)	PPG2 での例 (GCN10: TSEL2[3:0]の設定値)	PPG3 での例 (GCN10: TSEL3[3:0]の設定値)
GCN20 レジスタの EN0 ビットを選択するには	"0000"にする			
GCN20 レジスタの EN1 ビットを選択するには	"0001"にする			
GCN20 レジスタの EN2 ビットを選択するには	"0010"にする			
GCN20 レジスタの EN3 ビットを選択するには	"0011"にする			
リロードタイマ 0 を選択するには	"0100"にする			
リロードタイマ 1 を選択するには	"0101"にする			

■ 外部トリガの選択

外部トリガの設定は、
GCN10 レジスタ(PPG0~PPG3)・
GCN11 レジスタ(PPG4~PPG7)・
GCN12 レジスタ(PPG8~PPG11)・
GCN13 レジスタ(PPG12~PPG15)・
GCN14 レジスタ(PPG16~PPG19)・
GCN15 レジスタ(PPG20~PPG23)
の TSEL0/TSEL1/TSEL2/TSEL3 で行います。

PPG0～PPG3 の場合での設定を以下に示します。

外部トリガ	PPG0 での例 (GCN10: TSEL0[3:0]の設定値)	PPG1 での例 (GCN10: TSEL1[3:0]の設定値)	PPG2 での例 (GCN10: TSEL2[3:0]の設定値)	PPG3 での例 (GCN10: TSEL3[3:0]の設定値)
外部トリガ(TRGn)を選択するには	次のいずれかに設定します。 "1000","1001","1010","1011"			

複数の PPG で同じトリガを指定することで、同時に複数の PPG を起動できます。
(「17.9. 注意事項」参照)

■ 内部トリガ, 外部トリガエッジの選択

内部トリガと外部トリガのエッジの設定は、トリガ入力エッジ選択ビット(PCN0:EGS[1:0])～(PCN23:EGS[1:0])で行います。

内部トリガのエッジ選択	トリガ入力エッジ選択ビット (EGS[1:0])
検出しないとき (ソフトウェアトリガのみ)	"00"にする
"L"→"H" (立上り)でトリガ発生のとき	"01"にする
"H"→"L" (立下り)でトリガ発生のとき	"10"にする
両エッジでトリガ発生のとき	"11"にする

(「17.9. 注意事項」参照)

17.7.8 出力極性を反転させるには?

出力極性の反転について説明します。

出力極性指定

通常状態の極性を下表のように指定できます。

設定は PPG 出力極性指定ビット(PCNn:OSEL)で行います。(n=0～23)

("通常状態"とはパルスを出力していないときの状態です。)

通常状態での出力レベル	PPG 出力極性指定ビット (OSEL)
"L" レベル出力 (通常極性) にするには 	"0"にする
"H" レベル出力 (反転極性) にするには 	"1" にする

17.7.9　端子を PPG 出力端子にするには?

端子を PPG 出力端子にすることについて説明します。

端子をペリフェラル出力に設定します。設定方法は『I/O ポート』の章を参照してください。

17.7.10 起動トリガの発生方法は?

起動トリガの発生方法について説明します。

トリガの発生

以下に起動トリガの発生方法を示します。

ソフトトリガの起動方法
設定はソフトウェアトリガビット (PCNn:STRG)で行います。(n=0～23)

ソフトウェアトリガビット(STRG)に"1"を書き込むと起動トリガが発生します。

GCN10～GCN15 レジスタに関係なく常に有効です。

外部トリガでの起動方法
「17.7.7 起動トリガの種類と選択方法は?」を参照してください。また、TRG0,TRG1,TRG2 端子をペリフェラル入力に設定してください。設定方法は『I/O ポート』の章を参照してください。その後、TRG0,TRG1,TRG2 端子への入力レベルを変化させて起動トリガを発生させることができます。

リロードタイマ 0,リロードタイマ 1 での起動方法
リロードタイマの設定と起動が必要です。詳細は『リロードタイマ』の章を参照してください。

リロードタイマのアンダフローによって、リロードタイマの出力信号に指定したエッジが発生すると、起動トリガが発生します。

EN トリガ入力ビット (GCN20/21/22/23/24/25:EN[0:3])での起動方法
ソフトウェアで EN トリガ入力ビット(GCN20/21/22/23/24/25:EN[0:3])のレベルを書き変えることで起動トリガを発生させることができます。

エッジ	ソフトによる設定方法 (EN0, EN1, EN2, EN3)
立上りエッジ	最初に EN ビットを"0"にして，次に EN ビットを"1"にする。
立下りエッジ	最初に EN ビットを"1"にして，次に EN ビットを"0"にする。

複数の PPG を同時に起動する方法
PPG トリガ指定ビットで同じトリガ (トリガ入力ビット) を指定することでトリガ発生時に同時に起動できます。

＜注意事項＞
PPG 動作を許可する前に起動トリガを発生させても PPG は起動しません。起動トリガを発生させる前に必ず動作を許可してください。(「17.7.2. PPG 動作を許可/停止するには?」参照)

17.7.11 PPG 動作を停止するには?

PPG 動作の停止について説明します。

PPG 停止ビットを設定してください。(「17.7.2. PPG 動作を許可/停止するには?」参照)

17.7.12 割込み関連レジスタは?

割込み関連レジスタについて説明します。

PPG 割込みベクタ, PPG 割込みレベルの設定

PPG 番号,割込みレベル, 割込みベクタの関係は下表のとおりです。

割込みレベル, 割込みベクタの詳細については『割込み制御(割込みコントローラ)』の章を参照してください。

	割込みベクタ (デフォルト)	割込みレベル設定レジスタ(ICR[4:0])
PPG0	#40 Address: 0FFF5C$_H$	割込みレベルレジスタ(ICR24) Address: 00458$_H$
PPG1		
PPG10		
PPG11		
PPG20		
PPG21		
PPG2	#41 Address: 0FFF58$_H$	割込みレベルレジスタ(ICR25) Address: 00459$_H$
PPG3		
PPG12		
PPG13		
PPG22		
PPG23		
PPG4	#42 Address: 0FFF54$_H$	割込みレベルレジスタ(ICR26) Address: 0045A$_H$
PPG5		
PPG14		
PPG15		
PPG6	# 43 Address: 0FFF50$_H$	割込みレベルレジスタ(ICR27) Address: 0045B$_H$
PPG7		
PPG16		
PPG17		
PPG8	# 44 Address: 0FFF4C$_H$	割込みレベルレジスタ(ICR28) Address: 0045C$_H$
PPG9		
PPG18		
PPG19		

割込み要求フラグ(PCNn:IRQF)は自動的にクリアしませんので、割込み処理から復帰する前にソフトウェアにてクリアしてください。(IRQF ビットに"0"を書き込む)(n=0～23)

17.7.13 割込みの種類と選択方法は?

割込みの種類と選択方法について説明します。

割込み要因選択

割込み要因には、以下の 4 つがあり選択が可能です。

設定は、割込み要因設定ビット (PCNn:IRS[1:0]) にて行います。(n=0～23)

割込み要因	割込み要因設定ビット (IRS[1:0])
ソフトウェアトリガ または 内部トリガ発生	"00"にする
ダウンカウンタのボロー発生 (周期一致)	"01"にする
デューティ 一致発生	"10"にする
ダウンカウンタのボロー発生 (周期一致) または デューティ 一致発生	"11"にする

17.7.14 割込みを許可/禁止/クリアするには?

割込みの許可/禁止/クリアについて説明します。

割込み要求の許可フラグ、割込み要求フラグ

割込み許可の設定は、割込み要求許可ビット (PCNn:IREN)にて行います。(n=0～23)

動作	割込み要求許可ビット (IREN)
割込み要求を禁止するには	"0"にする
割込み要求を許可するには	"1"にする

割込み要求のクリアは, 割込み要求ビット (PCNn:IRQF) にて行います。(n=0～23)

動作	割込み要求ビット (IRQF)
割込み要求をクリアするには	"0"を書き込む

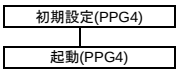
(「17.9. 注意事項」参照)

17.8 サンプルプログラム

サンプルプログラムについて説明します。

設定手順例 1

PPG4 から PWM 出力ソフトトリガ(duty1/4)通常極性



<初期設定>

・ポート	レジスタ名．ビット名
ポートの PPG 出力設定	『I/O ポート』の章を参照

・PPG4 制御	レジスタ名．ビット名
制御レジスタの設定	PCN4
タイマ動作許可≫	.CNTE
ソフトトリガ(未処理) ≫	.STRG
動作モード選択≫	.MDSE
再起動禁止≫	.RTRG
クロックソース選択≫	.CKS1-0
出力マスク選択≫	.PGMS
エッジ選択≫	.EGS1-0
割込み禁止≫	.IREN
割込みフラグクリア≫	.IRQF
	.IRS1-0
出力極性選択≫	.OSEL

・周期設定	レジスタ名．ビット名
PPG4 の周期設定	PCSR4

・デューティ設定	レジスタ名．ビット名
PPG4 のデューティ設定	PDUT4

<起動>

・PPG4 起動	レジスタ名．ビット名
PPG4 起動	PCN4.STRG

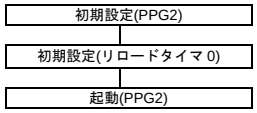
<その他>
(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 1

```
void PPG_sample_1(void)
{
        PPG4_initial();
        PPG4_start();
}


void PPG4_initial(void)
{
        PORT_SETTING_PPG4_OUT();    /* PPG4 端子をペリフェラル入力に設定してください。 */



        IO_PCN4.hword = 0x8000;     /*  設定値=1000_0000_0000_0000 */
                                    /* bit15 = 1       CNTE タイマ許可 */
                                    /* bit14 = 0       STRG ソフトウェアトリガ */
                                    /* bit13 = 0       MDSE PWM 動作 */
                                    /* bit12 = 0       RTRG 再起動禁止 */
                                    /* bit11-10 = 00 CKS1,0 */
                                    /* bit9 = 0        PGMS PPG 出力マスク */
                                    /* bit8 = 0        未定義ビット */
                                    /* bit7-6 = 00     EGS1,0 エッジ選択：無効 */
                                    /* bit5 = 0        IREN 割込み要求許可 */
                                    /* bit4 = 0        IRQF 割込み要求フラグ */
                                    * bit3-2 = 00     IRS1,0 割込み要因:ソフトウェアトリガ */
                                    /* bit1 = 0        未定義ビット */
                                    /* bit0 = 0        OSEL 通常極性 */


        IO_PCSR4 = 0x0909;              /* PPG 周期設定 */

        IO_PDUT4 = 0x0242;              /* PPG duty 比(1/4)設定 */
}

void PPG4_start(void)
{
        IO_PCN4.bit.STRG = 1;       /* bit14 = 1 STRG ソフトウェアトリガ */
}
```

設定手順例 2

PPG2 から PPG ワンショット出力リロードタイマ ch.0(duty1/2) 通常極性



<初期設定>

・ポート

	レジスタ名．ビット名
ポートの PPG 出力設定	『I/O ポート』の章を参照

・PPG2 制御

	レジスタ名．ビット名
制御レジスタの設定	PCN2
タイマ動作許可≫	.CNTE
ソフトトリガ(未処理) ≫	.STRG
動作モード選択≫	.MDSE
再起動禁止≫	.RTRG
クロックソース選択≫	.CKS1-0
出力マスク選択≫	.PGMS
エッジ選択≫	.EGS1-0
割込み禁止≫	.IREN
割込みフラグクリア≫	.IRQF
	.IRS1-0
出力極性選択≫	.OSEL

・周期設定

	レジスタ名．ビット名
PPG2 の周期設定	PCSR2

・デューティ設定

	レジスタ名．ビット名
PPG2 のデューティ設定	PDUT2

・トリガ選択

	レジスタ名．ビット名
PPG2 のトリガ選択	GCN10.TSEL2

＜初期設定(リロードタイマ 0) ＞

・リロードタイマ 0 の制御

	レジスタ名．ビット名
制御レジスタの設定	TMCSR0
モード選択≫	.MOD
内部クロック選択≫	.TRGM,CSL
トリガ選択≫	.TRGM
出力レベル選択≫	.OUTL
リロード許可≫	.RELD
割込み禁止≫	.INTE
割込みフラグクリア≫	.UF
カウント許可≫	.CNTE
ソフトトリガ(未処理)≫	.TRG

・カウント値

カウント値の設定	TMRLRA0

<起動>

・リロードタイマ 0 の起動により, PPG2 にトリガを入力

	レジスタ名．ビット名
ソフトトリガの発生	TMCSR0.TRG

<その他>

(注意事項)

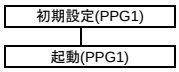
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 2

```
void PPG_sample_2(void)
{
        PPG2_initial();
        RTIM0_initial();
        RTIM0_start ();
}



void PPG2_initial(void)
{
        PORT_SETTING_PPG2_OUT();     /* PPG2 端子をペリフェラル入力に設定してください。 */



        IO_PCN2.hword = 0x8040;      /*  設定値=1000_0000_0100_0000 */
                                     /* bit15 = 1        CNTE タイマ許可 */
                                     /* bit14 = 0        STRG ソフトウェアトリガ */
                                     /* bit13 = 0        MDSE PWM 動作 */
                                     /* bit12 = 0        RTRG 再起動禁止 */
                                     /* bit11-10 = 00 CKS1,0 */
                                     /* bit9 = 0         PGMS PPG 出力マスク */
                                     /* bit8 = 0         未定義ビット */
                                     /* bit7-6 = 01      EGS1,0 エッジ選択：立上りエッジ */
                                     /* bit5 = 0         IREN 割込み要求許可 */
                                     /* bit4 = 0         IRQF 割込み要求フラグ */
                                     /* bit3-2 = 00      IRS1,0 割込み要因:ソフトウェアトリガ */
                                     /* bit1 = 0         未定義ビット */
                                     /* bit0 = 0         OSEL 通常極性 */


            IO_PCSR2 = 0x0909;               /* PPG 周期設定 */

            IO_PDUT2 = 0x0484;               /* PPG duty 比(1/2)設定 */

        IO_GCN10.bit.TSEL2 = 4;      /* bit11-8 = 0100 TSEL23-20  リロードタイマ ch.0 */
}

void RTIM0_initial(void)
{
        IO_TMCSR0.hword = 0x0012; /*  設定値  = 0000_0000_0001_0010 */
                                  /* bit15-14 = 00     MOD=00  シングルモード */
                                  /* bit13-12 = 00     TRGM=00  外部トリガ検出なし/  ソフトウェアトリガ */
                                  /* bit11-9 = 000     CSL=000  カウントソース選択(  周辺クロック÷  2) */
                                  /* bit8-6 = 000      GATE=0, EF=0 */
                                  /* bit5 = 0 OUTL=0  外部出力レベル*/
                                  /* bit4 = 1 RELD=1  リロード許可 */
                                  /* bit3 = 0 INTE=0  割込み要求禁止 */
                                  /* bit2 = 0 UF=0     フラグクリア */
                                  /* bit1 = 1 CNTE=1  タイマ動作許可/  起動トリガ待ち */
                                  /* bit0 = 0 TRG=0    トリガはまだ入れない */


        IO_TMRLRA0 = 0xffff;       /*  カウントの初期値 */
}

void rtim0_start(void)
{
        IO_TMCSR0 = IO_TMCSR0 | 0x0001;   /* bit0 = 1 TRG  ソフトウェアトリガ */
}
```

設定手順例 3

PPG1 から PPG ワンショット出力 High 出力起動トリガ
(GCN20:EN1)



<初期設定>

・ポート	レジスタ名. ビット名
ポートの PPG 出力設定	『I/O ポート』の章を参照

・PPG1 制御	レジスタ名. ビット名
制御レジスタの設定	PCN1
タイマ動作許可≫	.CNTE
ソフトトリガ(未処理) ≫	.STRG
動作モード選択≫	.MDSE
再起動禁止≫	.RTRG
クロックソース選択≫	.CKS1-0
出力マスク選択≫	.PGMS
エッジ選択≫	.EGS1-0
割込み禁止≫	.IREN
割込みフラグクリア≫	.IRQF
	.IRS1-0
出力極性選択≫	.OSEL

・周期設定	レジスタ名. ビット名
PPG1 の周期設定	PCSR1

・デューティ設定	レジスタ名. ビット名
PPG1 のデューティ設定	PDUT1

・トリガ選択	レジスタ名. ビット名
PPG1 のトリガ選択	GCN10.TSEL1

・トリガ信号レベル	レジスタ名. ビット名
トリガレベル="L"	GCN20.EN1

<起動>

・PPG1 起動	レジスタ名. ビット名
PPG1 起動	PCN4.STRG

・トリガ信号レベル	レジスタ名. ビット名
トリガレベル="H"	GCN20.EN1

<その他>
(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 3

```
void PPG_sample_3(void)
{
        PPG1_initial();
        PPG1_start ();
}


void PPG1_initial(void)
{
        PORT_SETTING_PPG1_OUT();    /* PPG1 端子をペリフェラル入力に設定してください。 */

        IO_PCN1.hword = 0xA040;     /*  設定値=1010_0000_0100_0000 */
                                    /* bit15 = 1       CNTE タイマ許可 */
                                    /* bit14 = 0       STRG ソフトウェアトリガ */
                                    /* bit13 = 1       MDSE ワンショット動作 */
                                    /* bit12 = 0       RTRG 再起動禁止 */
                                    /* bit11-10 = 00 CKS1,0 */
                                    * bit9 = 0         PGMS PPG 出力マスク */
                                    /* bit8 = 0        未定義ビット */
                                    /* bit7-6 = 01     EGS1,0 エッジ選択：立上りエッジ */
                                    /* bit5 = 0        IREN 割込み要求許可 */
                                    /* bit4 = 0        IRQF 割込み要求フラグ */
                                    /* bit3-2 = 00     IRS1,0 割込み要因:ソフトウェアトリガ */
                                    /* bit1 = 0        未定義ビット */
                                    /* bit0 = 0        OSEL 通常極性 */


        IO_PCSR1 = 0x0909;          /* PPG 周期設定 */

        IO_PDUT1 = 0x0484;          /* PPG duty 比(1/2)設定 */


        IO_GCN10.bit.TSEL1 = 1;     /* bit3-0 = 0001 TSEL03-00 GCN20 の EN1 ビット */

        IO_GCN20 = 0x00;            /* bit1 = 0 GNC20 の EN1 ビット */
}

void PPG1_start(void)
{
        IO_PCN4.bit.STRG = 1;       /* bit14 = 1 STRG ソフトウェアトリガ */

        IO_GCN20 = 0x02;            /* bit1 = 1 GNC20 の EN1 ビット */
}
```

設定手順例 4

インターバル割込み
PPG4 から PPG 出力ソフトウェアトリガ (duty1/4) 通常極性



<初期設定>

・ポート	レジスタ名．ビット名
ポートの PPG 出力設定	『I/O ポート』の章を参照

・PPG1 制御	レジスタ名．ビット名
制御レジスタの設定	PCN4
タイマ動作許可≫	.CNTE
ソフトトリガ(未処理) ≫	.STRG
動作モード選択≫	.MDSE
再起動禁止≫	.RTRG
クロックソース選択≫	.CKS1-0
出力マスク選択≫	.PGMS
エッジ選択≫	.EGS1-0
割込み禁止≫	.IREN
割込みフラグクリア≫	.IRQF
	.IRS1-0
出力極性選択≫	.OSEL

・周期設定	レジスタ名．ビット名
PPG4 の周期設定	PCSR4

・デューティ設定	レジスタ名．ビット名
PPG4 のデューティ設定	PDUT4

・割込み処理	レジスタ名．ビット名
PPG4 割込みレベルの設定	ICR26
I フラグの設定	(CCR)

<起動>

・PPG4 起動	レジスタ名．ビット名
割込み許可	PCN4.IREN
PPG4 起動	PCN4.STRG

<割込み>

・割込み処理	レジスタ名．ビット名
(任意の処理)	
割込み要求フラグのクリア	PCN4.IRQF

<割込みベクタ>
ベクタテーブルの設定

<その他>
(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 4

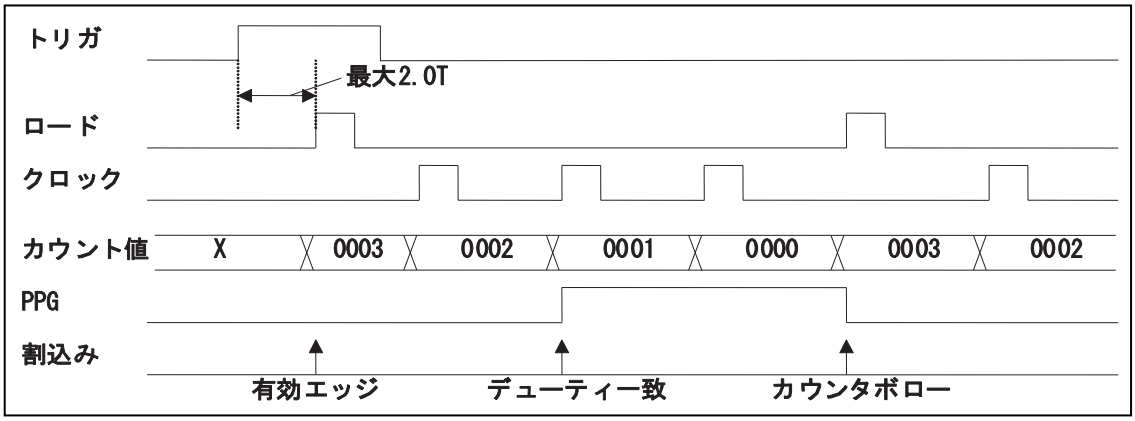
```
void PPG_sample_4(void)
{
        PPG4_initial();
        PPG4_start ();
}



void PPG4_initial(void)
{
        PORT_SETTING_PPG4_OUT();    /* PPG4 端子をペリフェラル入力に設定してください。 */

        IO_PCN4.hword = 0x8004;     /*  設定値=1000_0000_0000_0100 */
                                    /* bit15 = 1       CNTE タイマ許可 */
                                    /* bit14 = 0       STRG ソフトウェアトリガ */
                                    /* bit13 = 1       MDSE ワンショット動作 */
                                    /* bit12 = 0       RTRG 再起動禁止 */
                                    /* bit11-10 = 00 CKS1,0 */
                                    /* bit9 = 0        PGMS PPG 出力マスク */
                                    /* bit8 = 0        未定義ビット */
                                    /* bit7-6 = 01     EGS1,0 エッジ選択：立上りエッジ */
                                    /* bit5 = 0        IREN 割込み要求許可 */
                                    /* bit4 = 0        IRQF 割込み要求フラグ */
                                    /* bit3-2 = 01     IRS1,0 割込み要因:周期一致 */
                                    /* bit1 = 0        未定義ビット */
                                    /* bit0 = 0        OSEL 通常極性 */


        IO_PCSR4 = 0x0909;           /* PPG 周期設定 */

        IO_PDUT4 = 0x0242;           /* PPG duty 比(1/4)設定 */


        IO_ICR[26].byte = 0x10;     /*  割込みレベル( 値は任意)*/
        __EI();                      /*  割込み許可*/
}

void PPG4_start(void)
{
        IO_PCN4.bit.IREN = 1; /* bit5 = 1 IREN  割込み要求許可 */
        IO_PCN4.bit.STRG = 1; /* bit14 = 1 STRG  ソフトウェアトリガ */
}

__interrupt void PPG4_int(void)
{
                                    /*  任意の処理 */
        IO_PCN4.bit.IRQF = 0; /* bit14 = 0 IRQF  割込み要求フラグ */
}
```

ベクタテーブルにて割込みルーチンの指定が必要
#pragma intvect PPG4_int 42

17.9 注意事項

PPG の注意事項について説明します。

- 割込み要求フラグ (PCNn:IRQF) が"1"になるタイミングと"0"になるタイミングが重複した場合には、割込み要求フラグを"1"にセットする動作が優先し、フラグをクリアする要求は無効となります。(n=0～23)
- ダウンカウンタは、ロードとカウントのタイミングが重複した場合にはロード動作を優先します。



- 起動トリガがかかってから、カウンタ値がロードされるまで最大 2.0T (T:周辺クロック) を必要とします。
- 周期値 PCSRn の初期設定および書換えを行うときは、周期値書込み後に必ずデューティ値 PDUTn の書込みをおこなってください。
  (必ず、(1)PCSRn (2)PDUTn の順で書込みをしてください。)
  なお、デューティのみの書換えの場合は、PDUT のみ書き換えることができます。(n=0～23)
- デューティ値 PDUTn を設定する場合、周期値 PCSRn より小さな値を設定してください。もし大きな値を設定した場合には、PPG 動作禁止後にデューティ値を小さな値に書き直してください。(n=0～23)
- PPG 周期設定レジスタ PCSRn と PPG デューティ設定レジスタ PDUTn は、必ずハーフワード(16 ビット)でアクセスしてください。バイトアクセスした場合、上位/下位とも値が書き込まれません。(n=0～23)
- PPG を起動するには、起動よりも前もしくは同時にタイマ動作許可ビット(PCNn:CNTE)を"1"にして PPG 動作を許可する必要があります。(n=0～23)
- PPG 動作中に、モード(MDSE), 再起動許可(RTRG), カウントクロック(CKS[1:0]), トリガ入力エッジ(EGS[1:0]), 割込み要因(IRS[1:0]), 内部トリガ(TSEL), 出力極性指定(OSEL)の選択を変えてはいけません。PPG 動作中に値を変更した場合には、いったん PPG を動作禁止させてからレジスタ設定をやり直してください。
- GCN20/21/22/23/24/25 に値を書き込むとき、上位 4 ビットの未定義部分には必ず"0"を書き込んでください。もし"1"を書き込んだ場合には、PPG を動作停止にしてから、書き直してください。
- GCN10/11/12/13/14/15 の起動トリガ指定ビット(TSEL3, TSEL2, TSEL1, TSEL0)に、指定以外の値 (1100～1111) を設定した場合には、PPG 動作を禁止にしてから指定した値を書き込むと、正常にもどります。
- PPGn 動作中に、タイマ動作許可ビット(PCNn:CNTE)を"0"にして PPG 動作を禁止にすると、PPG はカウント値を保持して停止し、出力レベルは"L"(PCNn:OSEL=0 の場合)にクリアされます。その状態でタイマ動作許可ビット(PCNn:CNTE)を"1"にして PPG 動作を許可し、起動トリガが発生すると、周期(PCSR)とデューティ(PDUT)がそれぞれリロードされ、PPG の動作が再開します。(n=0～23)
- PPG コントロールレジスタの bit11, bit10(カウントクロック選択ビット CKS1 および CKS0)は書込み後直ちに反映されるので、設定変更はカウント停止状態で行ってください。
- ワンショット波形モードで動作させたときに「周期＝デューティ」に設定すると、出力される H パルス幅は、「(周期レジスタ設定値+2)×PCLK」となります。
  - タイマ動作許可ビット(PCN:CNTE)に"0"を設定して、PPG を禁止した場合には、PPG 出力が停止するまで、内部クロックで 3 クロックかかります。

- デューティ一致よりも後に PGMS を"1"から"0"にした場合は、次の周期から設定されたデューティでパルス出力されます。
- 周期値(PCSR)、デューティ値(PDUT)の書き込みは、必ず、(1)PCSR、(2)PDUT の順で書き込んでください。
  周期値(PCSR)、デューティ値(PDUT)の書き換え時の注意事項を次に示します。
  ① 周期値(PCSR)、デューティ値(PDUT)は、デューティ値(PDUT)書き込みを契機にバッファに取り込まれ、起動トリガ、または、ボロー発生時にバッファからカウンタへ転送されます。
  ② PPG 動作中に、周期値(PCSR)、または、デューティ値(PDUT)を書き換えたときの出力波形への反映は、デューティ値(PDUT)書き換え後の次の周期からとなります。
  ③ 周期値(PCSR)のみ書き換えたい場合でも、(1)PCSR、(2)PDUT の順で、デューティ値(PDUT)を同一値として、再設定し直す必要があります。
  ④ デューティ値(PDUT)は、任意に書き換えることができます。
- PPG 出力マスク中(PCN.PGMS='1')は、以下となります。デューティ一致による割込み要因により、割込みフラグが”1”にセットされることはありません。カウンタのボロー発生による割込み要因により、割込みフラグは”1”にセットされます。トリガ(ソフトウェアトリガ、外部トリガまたは GATE 信号によるトリガ)による割込み要因により割込みフラグは”1”にセットされます。

18. ウォッチドッグタイマ

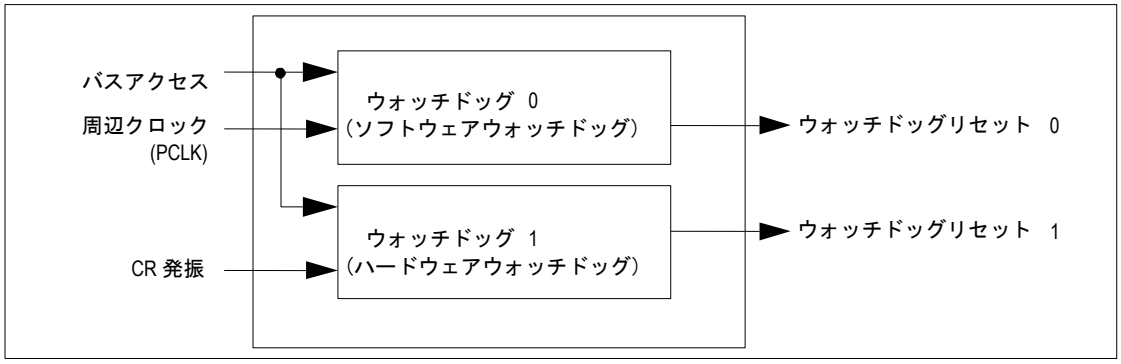
ウォッチドッグタイマについて説明します。

18.1 概要

ウォッチドッグタイマの概要について説明します。

本シリーズは、2 つのウォッチドッグタイマを持ち、ソフトウェアおよびハードウェアの暴走状態を検出してリセット要求を発生することができます。

図 18-1 ブロックダイヤグラム (概要)



18.2 特長

ウォッチドッグタイマの特長について説明します。

■ ウォッチドッグタイマ 0 (ソフトウェアウォッチドッグ)

● ストップモード検出機能
時計モード，ストップモードへの遷移を検出してリセット要求を発生することができます。

● ウォッチドッグタイマのクリア
動作初期化リセット，または前回クリアレジスタに書き込んだ値の反転値を書き込むことでタイマをクリアします。

● 不正書込み検出機能
クリアレジスタに不正な値が書き込まれるとリセット要求を発生します。

● ウォッチドッグタイマ周期
周辺クロック (PCLK) × (2^9 ～ 2^{24}) サイクルの 16 とおりから選択できます。

● カウント停止条件
CPU 停止中はカウントを停止します。

■ ウォッチドッグタイマ 1 (ハードウェアウォッチドッグ)
リセット解除後、すぐに搭載 CR 発振回路で生成されたクロックで駆動します。(CR 発振の設定(キャリブレーション) については『RTC/WDT1 補正 (キャリブレーション) 』の章を参照してください。)

● ウォッチドッグタイマのクリア
動作初期化リセット，またはクリアレジスタに"0xA5"を書き込むことでタイマをクリアします。

● 不正書込み検出機能
クリアレジスタに"0xA5"以外の値が書き込まれるとリセット要求を発生します。

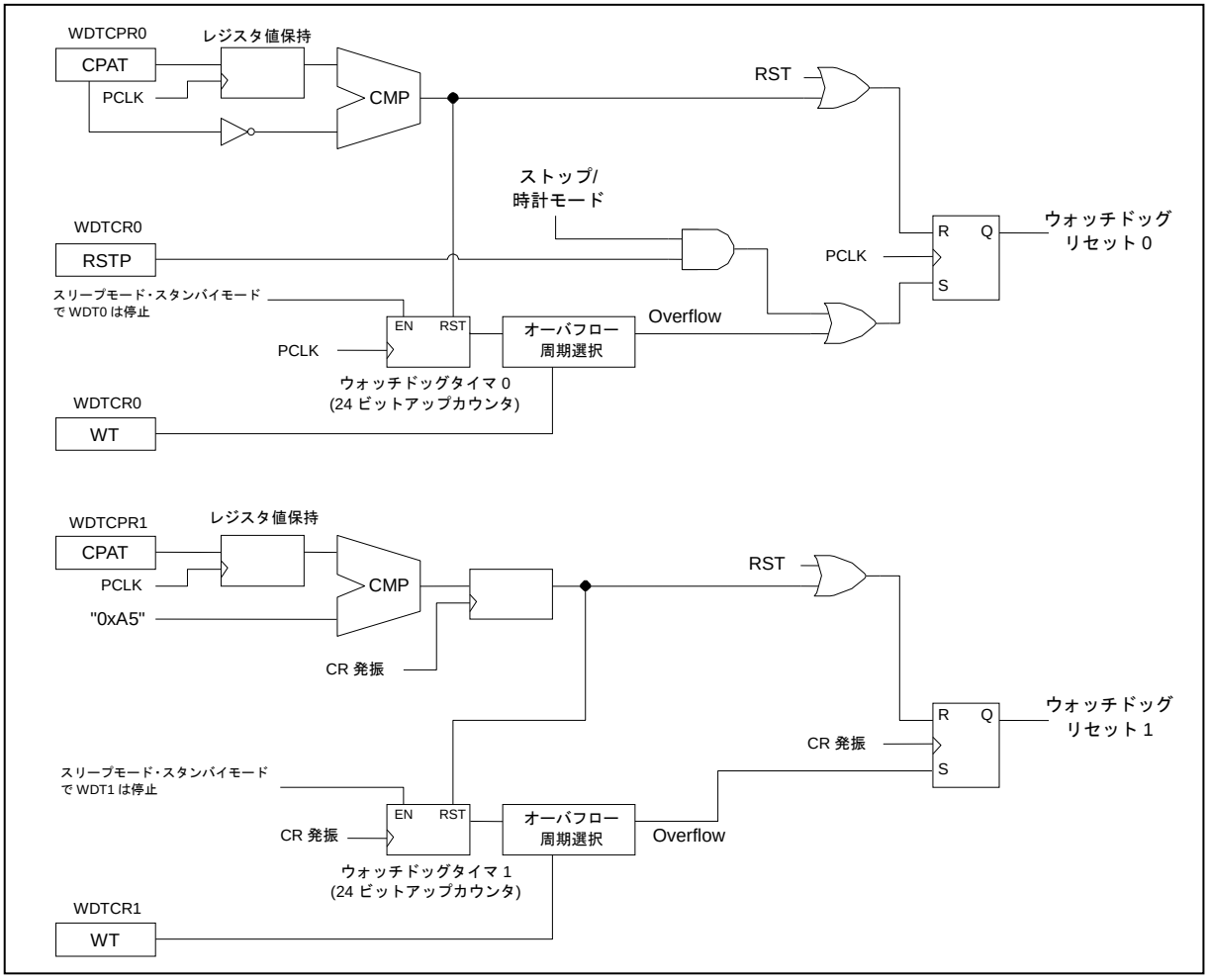
● ウォッチドッグタイマ周期
ハードウェア固定で、CR 発振×2^{15} サイクルです。

● カウント停止条件
ICE 使用中、スリープモード時，時計モード時，ストップモード時，スタンバイモードからの復帰時の発振安定待ち中は、カウントを停止します。

18.3 構成

ウォッチドッグタイマの構成について示します。

図 18-2 ブロックダイヤグラム(詳細)



18.4 レジスタ

ウォッチドッグタイマのレジスタについて説明します。

表 18-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x003C	WDTCR0	WDTCPR 0	WDTCR1	WDTCPR 1	ウォッチドッグタイマ 0 制御レジスタ ウォッチドッグタイマ 0 クリアレジスタ ウォッチドッグタイマ 1 周期インフォメーションレジスタ ウォッチドッグタイマ 1 クリアレジスタ

18.4.1 ウォッチドッグタイマ 0 制御レジスタ : WDTCR0 (WatchDog Timer Control Register 0)

ウォッチドッグ制御レジスタ 0 (WDTCR0) のビット構成について説明します。

ウォッチドッグタイマ 0 の各種設定を行います。

ウォッチドッグタイマ 0 起動後の本レジスタへの書込みは無効です。

■ WDTCR0 : アドレス 003C$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	RSTP	予約		WT[3:0]			
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R/W	R0,W0	R0,W0	R/W	R/W	R/W	R/W

[bit7] 予約
このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit6] RSTP (Reset by SToP) : ストップモード検出リセット許可
ウォッチドッグタイマ 0 動作時、時計モードまたはストップモード遷移を検出したときに、リセットを発生するかどうかを設定します。許可した場合、時計モードまたはストップモードへ遷移するとウォッチドッグリセット 0 が発生します。許可していない場合、時計モードまたはストップモードへ遷移するとウォッチドッグタイマ 0 は一時停止し、時計モードまたはストップモードから復帰する迄カウントを行いません。

RSTP	ストップモード検出
0	検出しない(初期値)
1	検出してリセットを発生する

ウォッチドッグタイマ 0 起動後の本ビットへの書込みは無効です。

[bit5,bit4] 予約
このビットは、常に"0"を書き込んでください。読出し値は"0"です。

[bit 3〜bit0] WT[3:0] (Watchdog Timer interval) : **ウォッチドッグタイマ周期選択**

ウォッチドッグタイマ 0 が最後にクリアされてからウォッチドッグリセット 0 が発行されるまでのサイクル数を以下のように設定します。

WT[3:0]	ウォッチドッグタイマ 0 周期
0000	PCLK (周辺クロック) × 2^9 サイクル
0001	PCLK × 2^{10} サイクル
0010	PCLK × 2^{11} サイクル
0011	PCLK × 2^{12} サイクル
0100	PCLK × 2^{13} サイクル
0101	PCLK × 2^{14} サイクル
0110	PCLK × 2^{15} サイクル
0111	PCLK × 2^{16} サイクル
1000	PCLK × 2^{17} サイクル
1001	PCLK × 2^{18} サイクル
1010	PCLK × 2^{19} サイクル
1011	PCLK × 2^{20} サイクル
1100	PCLK × 2^{21} サイクル
1101	PCLK × 2^{22} サイクル
1110	PCLK × 2^{23} サイクル
1111	PCLK × 2^{24} サイクル

ウォッチドッグタイマ 0 起動後の本ビットへの書込みは無効です。

ウォッチドッグタイマ 0 は、CPU が動作停止している期間はカウントを行いません。

DMA 転送が行われていても、CPU が動作している期間はカウントを行います。

18.4.2 ウォッチドッグタイマ 0 クリアレジスタ ：WDTCPR0 (WatchDog Timer Clear Pattern Register 0)

ウォッチドッグタイマ 0 クリアレジスタ(WDTCPR0)のビット構成について説明します。

ウォッチドッグタイマ 0 の起動およびクリア (リセット発行延期) を行うレジスタです。

■ WDTCPR0： アドレス 003DH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	CPAT[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W

[bit7～bit0] CPAT[7:0] (Clear PATtern) : **ウォッチドッグタイマ 0 クリア**
リセット解除後、本レジスタへの 1 回目の書込みを行うことによりウォッチドッグタイマ 0 が起動します。起動後のウォッチドッグタイマのクリアは、前回書き込んだ値の全ビット反転した値を書き込むことにより行います。前回書き込んだ値の反転値以外を書き込んだ場合、その時点でウォッチドッグリセット 0 が発行されます。

本レジスタの読出し値は、書込み値にかかわらず常に"0x00"となります。

18.4.3 ウォッチドッグタイマ 1 周期インフォメーションレジスタ ： WDTCR1 (WatchDog Timer Cycle information Register 1)

ウォッチドッグタイマ 1 周期インフォメーションレジスタ (WDTCR1)のビット構成について説明します。

ウォッチドッグタイマ 1 の各種設定を行います。

■ WDTCR1： アドレス 003EH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				WT[3:0]			
初期値	0	0	0	0	0	1	1	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R1,WX	R1,WX	R0,WX

本レジスタは書き換えできません。

[bit7～bit4] **予約**
常に"0"が読み出されます。書込みは動作に影響ありません。

[bit3～bit0] WT[3:0] (Watchdog Timer interval) : **ウォッチドッグタイマ周期選択**
ウォッチドッグタイマ 1 が最後にクリアされてからウォッチドッグリセット 1 が発行されるまでのサイクル数です。2^{15} サイクルに固定されています。本ビットへの書込みは無効です。

WT[3:0]	ウォッチドッグタイマ 1 周期
0110	CR 発振 × 2^{15} サイクル(初期値・固定)

18.4.4 ウォッチドッグタイマ 1 クリアレジスタ ：WDTCPR1 (WatchDog Timer Clear Pattern Register 1)

ウォッチドッグタイマ 1 クリアレジスタ (WDTCPR1)のビット構成について説明します。

ウォッチドッグタイマ 1 のクリア (リセット発行延期) を行うレジスタです。

■ WDTCPR1： アドレス 003F$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	CPAT[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W

[bit7～bit0] CPAT[7:0] (Clear PATtern) : ウォッチドッグタイマ 1 クリア

リセット解除後、ウォッチドッグタイマ 1 が起動します。起動後のウォッチドッグタイマのクリアは、"0xA5" を書き込むことにより行います。"0xA5"以外を書き込んだ場合、その時点でウォッチドッグリセット 1 が発行されます。本レジスタの読出し値は書込み値にかかわらず常に"0x00"となります。

18.5 動作説明

ウォッチドッグタイマの動作について説明します。

以下、ウォッチドッグタイマ機能について説明します。

■ ソフトウェアウォッチドッグ機能

● 設定

ウォッチドッグタイマ 0 を起動する前に、WDTCR0 レジスタの bit3-bit0:WT[3:0]の設定を行い、ウォッチドッグタイマのクリアからリセット発生までの周期を選択します。

ウォッチドッグタイマ 0 は CPU 動作のみのカウントを行うため、プログラムステップ数とクロック分周の設定を基準に周期を設定してください。

ウォッチドッグタイマ 0 を起動する前に WDTCR0 レジスタの bit6:RSTP の設定を行い、時計モード / ストップモード遷移を検出してリセットを発生するかどうかを選択します。

- RSTP=0 時、時計モード / ストップモード中はタイマを停止します。
- RSTP=1 時、時計モード / ストップモード遷移と同時にリセットを発生します。

時計モード, ストップモードを使用する場合は RSTP="0"を設定してください。ウォッチドッグタイマ 0 起動後の RSTP ビットへの書込みは無効です。

● 起動

ウォッチドッグタイマ 0 は、リセット後 1 回目の WDTCPR0 レジスタへの任意データの書込みにより起動します。

書込みデータに制限はありません。

WDTCPR0 レジスタは書込みデータにかかわらず常に"0x00"が読み出されます。

● 動作

起動後のウォッチドッグタイマ 0 の動作について説明します。

カウント条件

ウォッチドッグタイマ 0 は、CPU が動作中、周辺クロック(PCLK)の立上りエッジでカウントを行います。

DMA 転送は、カウント動作に影響を与えません。

スリープモードなど、CPU が停止している期間のみカウントを停止します。CPU 動作状態のサンプリングは周辺クロック(PCLK)で行うため、周辺クロック周期未満の動作状態の変化は無視されます。

ICE 接続時のエミュレータモード中は、カウントを停止します。ICE 接続時、デバッグインタフェースの機能でウォッチドッグリセットの抑止機能が有効の場合は、カウントを停止します。

上記いずれの条件においても、カウントを停止する際にはカウンタのクリアは行わずに一時停止を行うので、カウントを再開すると停止前のカウンタ値からの継続カウントとなります。

ソースクロックの発振安定待ち時間中は、周辺クロックが停止しますので、ウォッチドッグタイマのカウントも停止しています。

タイマのクリア
ウォッチドッグタイマの起動後、タイマ周期が経過する前に、タイマをクリアしなければなりません。

ウォッチドッグタイマのクリアは、WDTCPR0 へのデータの書込みにより行います。書き込むデータは、前回 WDTCPR0 に書き込んだデータの全ビット反転値でなければなりません。

ウォッチドッグタイマ 0 起動時に、WDTCPR0 に例えば"0x55"を書き込んで起動した場合は、以降は"0xAA"→"0x55"→"0xAA"→"0x55"と交互に書き込んで行くことによりタイマのクリアを行います。

WDTCPR0 は読出し値が常に"0x00"であるため、WDTCPR0 を読み出して前回の書込み値を確認することはできません。前回書き込んだ値を他所に保持しておけない場合、1 回のクリア時に 2 回連続で書込みを行うことで対処してください。

リセット要求生成
ウォッチドッグタイマ 0 は、以下の条件でウォッチドッグリセット要求を発生します。

- 設定したウォッチドッグタイマ周期のオーバフローの発生
- ストップモード検出リセット許可中の時計モードまたはストップモードへの遷移
- クリアレジスタへの前回書込んだ値の反転値以外の値の書込み

■ ハードウェアウォッチドッグ機能

● 設定
ウォッチドッグタイマ 1 の WDTCR1 レジスタの bit3-bit0:WT[3:0]は、ハードウェアで固定です。

● 起動
ウォッチドッグタイマ 1 は、リセット解除後すぐに起動します。

● 動作
起動後のウォッチドッグタイマ 1 の動作について説明します。

カウント条件
ウォッチドッグタイマ 1 は CR 発振の立上りエッジでカウントを行います。

ICE 接続時のエミュレータモード中は、カウントを停止します。ICE 接続時、デバッグインタフェースの機能でウォッチドッグリセットの抑止機能が有効の場合は、カウントを停止します。

スリープモード時, 時計モード時, ストップモード時, スタンバイモードからの復帰時の発振安定待ち中は、カウントを停止します。

タイマのクリア
ウォッチドッグタイマの起動後、タイマ周期が経過する前に、タイマをクリアしなければなりません。

ウォッチドッグタイマ 1 のクリアは、WDTCPR1 へ"0xA5"を書き込むことにより行います。

リセット要求生成
ウォッチドッグタイマ 1 は、以下の条件でウォッチドッグリセット要求を発生します。

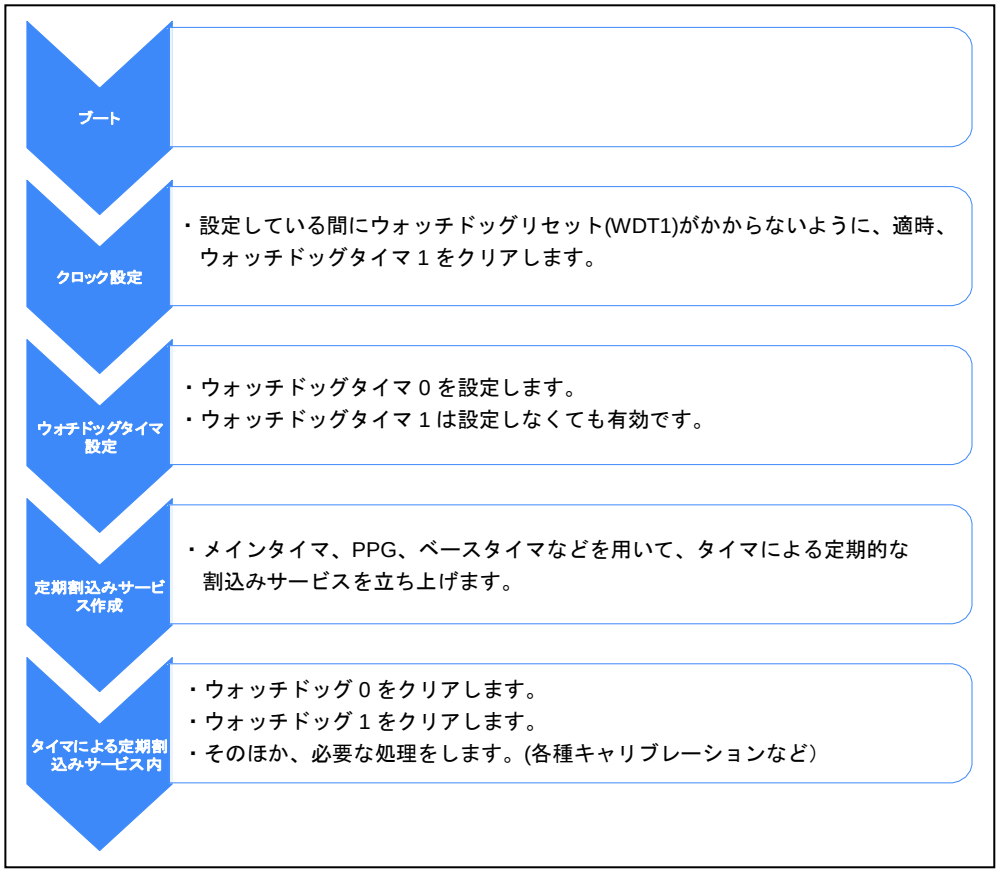
- ウォッチドッグタイマ周期のオーバフローの発生
- WDTCPR1 へ"0xA5"以外の値の書込み

18.6 使用例

ウォッチドッグタイマの使用例について示します。

ウォッチドッグタイマのクリアのための使用例です。

図 18-3 ウォッチドッグタイマクリアの例



19. ベースタイマ

ベースタイマについて説明します。

19.1 概要

ベースタイマの概要について説明します。

本シリーズはベースタイマを 2 チャネル搭載しています。ベースタイマは、次の機能を提供します。

- 16/32 ビットリロードタイマ
- 16 ビット PWM タイマ
- 16 ビット PPG タイマ
- 16/32 ビット PWC タイマ

19.2 特長

ベースタイマの特長について説明します。

本シリーズはベースタイマを 2 チャネル搭載しています。それぞれのチャネルで次の機能を選択して使用します。

19.2.1. 16/32 ビットリロードタイマ

19.2.2. 16 ビット PWM タイマ

19.2.3. 16/32 ビット PWC タイマ

19.2.4. 16 ビット PPG タイマ

19.2.1 16/32 ビットリロードタイマ

ベースタイマの 16/32 ビットリロードタイマについて説明します。

ベースタイマを 16/32 ビットリロードタイマとして使用できます。16/32 ビットリロードタイマはあらかじめ設定した値からカウントダウンするタイマです。

● 入出力モード

ベースタイマ入出力選択機能を利用して、信号(外部クロック/ 外部起動トリガ/ 波形)の入出力動作を選択できます。

● タイマモード

チャネルごとに個別に動作させることも、2チャネルの16ビットリロードタイマを組み合わせて32ビットリロードタイマとしても利用できます。

● 動作モード

次の 2 種類から選択できます。

- リロードモード : ダウンカウンタがアンダフローすると, 設定している値(周期)をリロードしてカウントを繰り返すモードです。
- ワンショットモード : ダウンカウンタがアンダフローすると、カウントを停止するモードです。

● カウント用クロック

内部クロック(周辺クロック) 5 種類, 外部クロック(ECK 信号) 3 種類の中から選択できます。

- 内部クロック(周辺クロック) : 周辺クロック(PCLK)の 1 分周・4 分周・16 分周・128 分周・256 分周
- 外部クロック(ECK 信号) : 立上りエッジ, 立下りエッジ, 両エッジ検出

● 起動トリガ

次の中から選択できます。

- ソフトウェアトリガ
- 外部イベント : 立上りエッジ, 立下りエッジ, 両エッジ
- 16/32 ビットリロードタイマの再起動 : カウント動作中に起動トリガを検出したときに 16/32 ビットリロードタイマを再起動できます。

● 割込み要求

次の場合に割込み要求を発生できます。

- IRQ0 : アンダフローが発生したとき
- IRQ1 : 16/32 ビットリロードタイマの起動トリガを検出したとき

19.2.2 16 ビット PWM タイマ

ベースタイマの 16 ビット PWM タイマについて説明します。

16 ビット PWM タイマは、パルス幅変調タイマ(Pulse Width Modulator Timer)の略で、パルス幅のデューティ比を設定することで外部端子から任意の波形を出力するタイマです。

● 入出力モード

ベースタイマ入出力選択機能を利用して、信号(外部クロック/ 外部起動トリガ/ 波形)の入出力動作を選択できます。

● 動作モード

次の 2 種類から選択できます。

- リロードモード : 16 ビットダウンカウンタがアンダフローすると設定してある周期をリロードしてカウントを繰り返すモードです。
- ワンショットモード : 16 ビットダウンカウンタがアンダフローするとカウントを停止するモードです。

● カウント用クロック

内部クロック(周辺クロック) 5 種類, 外部クロック(ECK 信号) 3 種類の中から選択できます。

- 内部クロック(周辺クロック) : 周辺クロック(PCLK)の 1 分周, 4 分周, 16 分周, 128 分周, 256 分周
- 外部クロック(ECK 信号) : 立上りエッジ, 立下りエッジ, 両エッジ検出

● 起動トリガ

次の中から選択できます。

- ソフトウェアトリガ
- 外部イベント 3 種類 : (立上りエッジ, 立下りエッジ, 両エッジ検出)

● 16 ビット PWM タイマの再起動

カウント動作中に起動トリガを検出したときに 16 ビット PWM タイマを再起動できます。

● 出力波形

外部端子からの出力信号を"L"レベルまたは"H"レベルに固定できます。

● 割込み要求

次の場合に割込み要求を発生できます。

- IRQ0 : アンダフローが発生したとき、あらかじめ決めておいた値(デューティ)までカウントしたとき
- IRQ1 : 16 ビット PWM タイマの起動トリガを検出したとき

19.2.3 16/32 ビット PWC タイマ

ベースタイマの 16/32 ビット PWC タイマ概要について説明します。

16/32 ビット PWC タイマは, パルス幅カウンタタイマ(Pulse Width Counter)の略で、パルス幅や周期を測定するタイマです。

● 入出力モード

ベースタイマ入出力選択機能を利用して、信号(波形)の入力動作を選択できます。

● タイマモード

チャネルごとに個別に動作させることも、2 チャネルの 16 ビット PWC タイマを組み合わせて 32 ビット PWC タイマとしても利用できます。

● 動作モード

次の 2 種類から選択できます。

- 単発測定モード : 測定を 1 回のみ行うモードです。
- 連続測定モード :1 回測定が終わると、次の測定開始エッジが入力されるまで待機し、再度測定開始エッジが入力されると測定を行うモードです。

● カウント用クロック

周辺クロック(PCLK)を分周して生成した内部クロック(周辺クロック) 5 種類の中から選択できます。

- 周辺クロック(PCLK)の 1 分周, 4 分周, 16 分周, 128 分周, 256 分周

● 測定モード

測定するパルス幅や周期を次の 5 種類から選択できます。

- "H"パルス幅 :"H"レベルの信号が入力されている期間
- "L"パルス幅 :"L"レベルの信号が入力されている期間
- 立上りエッジ間周期 : 立上りエッジを検出してから、次の立上りエッジを検出するまでの期間
- 立下りエッジ間周期 : 立下りエッジを検出してから、次の立下りエッジを検出するまでの期間
- 全エッジ間パルス幅 : 連続して入力されるエッジ間の幅は次のいずれかになります。
  - 立上りエッジを検出してから立下りエッジを検出するまでの期間
  - 立下りエッジを検出してから立上りエッジを検出するまでの期間

● 16/32 ビット PWC タイマの再起動

カウント動作中に起動トリガを検出したときに 16/32 ビット PWC タイマを再起動できます。

● 割込み要求

次の場合に割込み要求を発生できます。

- IRQ0 : オーバフローが発生したとき
- IRQ1 : 測定が終了したとき

19.2.4 16 ビット PPG タイマ

ベースタイマの 16 ビット PPG タイマについて説明します。

16 ビット PPG タイマは、プログラマブルパルス発生タイマ(Programmable Pulse Generator Timer)の略で、任意のパルス幅を持つ波形を出力するタイマです。

● 入出力モード

ベースタイマ入出力選択機能を利用して、信号(外部クロック/ 外部起動トリガ/ 波形)の入出力動作を選択できます。

● 動作モード

次の 2 種類から選択できます。

- リロードモード : "L"レベルと"H"レベルの信号を連続して出力(連続パルス)するモードです。
- ワンショットモード : "L"レベルと"H"レベルの信号を 1 回ずつ出力(単一パルス)するモードです。

● カウント用クロック

内部クロック(周辺クロック) 5 種類、外部クロック(ECK 信号) 3 種類の中から選択できます。

- 内部クロック(周辺クロック) : 周辺クロック(PCLK)の 1 分周, 4 分周, 16 分周, 128 分周, 256 分周
- 外部クロック(ECK 信号) : 立上りエッジ, 立下りエッジ, 両エッジ検出

● 起動トリガ

次の中から選択できます。

- ソフトウェアトリガ
- 外部イベント 3 種類 : (立上りエッジ, 立下りエッジ, 両エッジ検出)

● 16 ビット PPG タイマの再起動

カウント動作中に起動トリガを検出したときに 16 ビット PPG タイマを再起動できます。

● 割込み要求

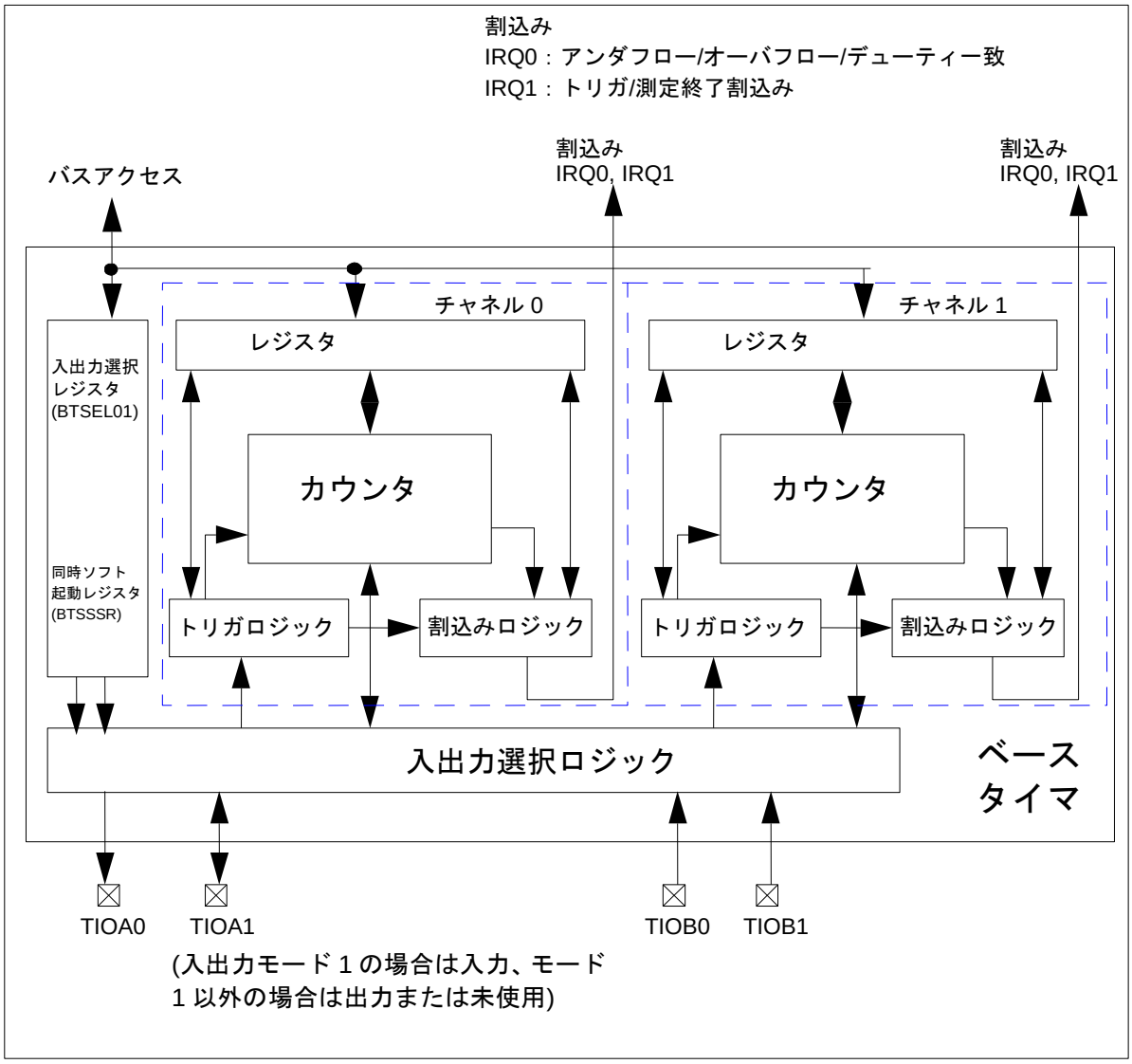
次の場合に割込み要求を発生できます。

- IRQ0 : ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値でアンダフローが発生したとき
- IRQ1 : 16 ビット PPG タイマの起動トリガを検出したとき

19.3 構成

ベースタイマの構成について示します。

図 19-1 ブロックダイヤグラム (概要)



19.4 レジスタ

ベースタイマのレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子一覧

表 19-1 ベースアドレス (Base_addr)・外部端子表

チャネル番号	ベースアドレス	外部端子
0	0x0080	TIOA0, TIOA1, TIOB0, TIOB1 を BTSEL01 レジスタの設定に応じて割当て
1	0x0090	

■ レジスタマップ

表 19-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0080	【共通】BT0TMR		【共通】BT0TMCR		【共通】タイマレジスタ 0 【共通】制御レジスタ 0
0x0084	予約	【リロードタイマ】 BT0STC 【PWM】 BT0STC 【PPG】 BT0STC 【PWC】 BT0STC	予約		【リロードタイマ】ステータス制御レジスタ 0 【PWM】ステータス制御レジスタ 0 【PPG】ステータス制御レジスタ 0 【PWC】ステータス制御レジスタ 0
0x0088	【リロードタイマ】 BT0PCSR 【PWM】 BT0PCSR 【PPG】 BT0PRLL 【PWC】 予約		【リロードタイマ】 予約 【PWM】 BT0PDUT 【PPG】 BT0PRLH 【PWC】 BT0DTBF		【リロードタイマ】周期設定レジスタ 0 【PWM】周期設定レジスタ 0 【PPG】L 幅設定リロードレジスタ 0 【PWM】デューティ設定レジスタ 0 【PPG】H 幅設定リロードレジスタ 0 【PWC】データバッファレジスタ 0
0x008C	予約				
0x0090	【共通】BT1TMR		【共通】BT1TMCR		【共通】タイマレジスタ 1 【共通】制御レジスタ 1
0x0094	予約	【リロードタイマ】 BT1STC 【PWM】 BT1STC 【PPG】 BT1STC 【PWC】 BT1STC	予約		【リロードタイマ】ステータス制御レジスタ 1 【PWM】ステータス制御レジスタ 1 【PPG】ステータス制御レジスタ 1 【PWC】ステータス制御レジスタ 1
0x0098	【リロードタイマ】 BT1PCSR 【PWM】 BT1PCSR 【PPG】 BT1PRLL 【PWC】 予約		【リロードタイマ】 予約 【PWM】 BT1PDUT 【PPG】 BT1PRLH 【PWC】 BT1DTBF		【リロードタイマ】周期設定レジスタ 1 【PWM】周期設定レジスタ 1 【PPG】L 幅設定リロードレジスタ 1 【PWM】デューティ設定レジスタ 1 【PPG】H 幅設定リロードレジスタ 1 【PWC】データバッファレジスタ 1
0x009C	BTSEL01	予約	BTSSSR		入出力選択レジスタ 同時ソフト起動レジスタ

19.4.1 共通レジスタ

ベースタイマの共通レジスタについて説明します。

各動作で共通のレジスタです。

19.4.1.1 タイマレジスタ 0, 1 : BTxTMR (Base Timer 0/1 TiMer Register)

タイマレジスタ 0, 1 (BTxTMR)のビット構成について説明します。

タイマのカウンタ値を読み出すレジスタです。リロードタイマ, PWM, PPG 時のみ有効です。PWC 時の読出し値は不定です。読出し値については「5.動作説明」を参照してください。

＜注意事項＞
このレジスタは 16 ビットでアクセスしてください。

■ BTxTMR： アドレス Base_addr + 00H (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	D[15:0]					
初期値	0	0	・・・	0	0	0
属性	R,WX	R,WX	・・・	R,WX	R,WX	R,WX

19.4.1.2 タイマ制御レジスタ 0, 1 : BTxTMCR (Base Timer 0/1 TiMer Control Register)

タイマ制御レジスタ 0, 1 (BTxTMCR) のビット構成について説明します。

ベースタイマの各種設定 および動作停止，ソフトウェアトリガ発行を行うレジスタです。

＜注意事項＞

- FMD[2:0]の設定を変更する場合、いったん FMD[2:0]=000 を設定してから、FMD[2:0]を設定してください。
- 予約となっているビットには"0"を設定してください。
- ソフトウェアトリガ(STRG)以外の本レジスタビット設定する場合は、次の順序で行ってください。
  1. いったん FMD[2:0]=000, または CTEN=0 を書き込みして動作を停止させる。
  2. 各ビット、並びにタイマ機能選択(FMD[2:0])の設定したい値を書き込む。
- ソフトウェアトリガ(STRG)に書き込む際は、ほかのビットをクリアしてしまわないように注意してください。
- FMD[2:0]=000 はリセットモードのため、FMD[2:0]=000 と同時にほかのビットを設定することはできません。
- このレジスタは 16 ビットでアクセスしてください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxTMCR： アドレス Base_addr + 02_H (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	CKS[2:0]			【PWM・PPG】RTGEN【他】予約	【PWM・PPG】PMSK【PWC】EGS[2]【他】予約	EGS[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W0 R0,W0(*3)	R/W	R/W	R/W	R/W R0,WX(*1)	R/W R0,WX(*1)	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	【リロードタイマ・PWC】T32【他】予約	FMD[2:0]			【リロードタイマ・PWM・PPG】OSEL【他】予約	MDSE	CTEN	STRG
初期値	0	0	0	0	0	0	0	0
属性	R/W R0,W0(*1) R0,W0(*2)	R/W	R/W	R/W	R/W R/W0(*1)	R/W	R,W	R0,W R0,W0(*1)

(*1) 予約の場合の属性
(*2) 32 ビットタイマ奇数チャネル側時の属性
(*3) 32 ビットタイマ奇数チャネル側時または 16/32 ビット PWC タイマ時の属性

[bit15] 予約
0 を書き込んでください。

[bit14～bit12] CKS[2:0] (ClocK Select) : カウントクロック選択ビット
カウント用クロックを選択します。

CKS[2:0]	説明	
	クロックソース	説明
000	内部クロック (周辺クロック(PCLK))	1 分周
001		4 分周
010		16 分周
011		128 分周
100		256 分周
101	【リロードタイマ・PWM・PPG】外部クロック(ECK 信号) 【PWC】設定禁止	立上りエッジ
110		立下りエッジ
111		両エッジ

PWC モードのときは、101, 110, 111 は設定禁止です。

【PWM・PPG】[bit11] RTGEN (Restart by TriGger ENable) : 再起動許可ビット
STRG ビットに"1"が書き込まれた場合や、外部起動トリガ(TGIN 信号)が検出された場合に、周期設定レジスタ(BTxPCSR)/L 幅設定リロードレジスタ(BTxPRLL)の値を 16 ビットダウンカウンタにリロードして、 カウントし直すかどうかを設定します。

RTGEN	動作内容
0	再起動しない
1	再起動する

【PWM・PPG】[bit10] PMSK (Pulse MaSK) : パルス出力マスクビット
出力する波形(TOUT 信号)のレベルを次の中から選択します。

- 通常出力 : 16 ビット PWM/PPG タイマからの出力波形をそのまま出力します。
- 固定出力 : 周期やデューティの設定にかかわらず"L"レベルまたは"H"レベルを出力し続けます。

PMSK	説明
0	通常出力
1	固定出力

このビットに"1"を書き込んで固定出力を選択した場合、出力されるレベルは OSEL ビットの設定によって異なります。

- OSEL=0 の場合 : "L"レベルが出力されます。
- OSEL=1 の場合 : "H"レベルが出力されます。

【リロードタイマ・PWM・PPG】[bit9, bit8] EGS[1:0] (EdGe Select) : トリガ入力選択ビット
外部起動トリガ(TGIN)信号の有効エッジを選択します。

EGS[1:0]	説明
00	トリガ入力は動作に影響しません
01	立上りエッジ
10	立下がりエッジ
11	両エッジ

【PWC】[bit10～bit8] EGS[2:0] (EdGe Select) : 測定モード選択ビット
測定する種類を選択します。

EGS[2:0]	説明
000	"H"パルス幅測定 :"H"レベルの信号が入力されている期間
001	立上りエッジ間周期測定 : 立上りエッジを検出してから、次の立上りエッジを検出するまでの期間
010	立下りエッジ間周期測定 : 立下りエッジを検出してから、次の立下りエッジを検出するまでの期間
011	全エッジ間パルス幅測定 : 連続して入力されるエッジ間の幅①②のいずれかになります。 ①立上りエッジを検出してから立下りエッジを検出するまでの期間 ②立下りエッジを検出してから立上りエッジを検出するまでの期間
100	"L"パルス幅測定 :"L" レベルの信号が入力されている期間 (立下りエッジを検出してから, 立上りエッジを検出するまでの期間)
101 110 111	設定禁止

【リロードタイマ・PWC】[bit7] T32 (Timer 32bit) : 32 ビットタイマ選択ビット
16/32 ビットタイマを 1 チャネルずつ個別に動作させるか、2 チャネルをカスケード接続して 32 ビットのタイマとして使うかを選択します。ch.0 と ch.1 の本ビットの両方を設定してください。

T32(ch.0)	T32(ch.1)	説明
0	0	それぞれ 16 ビットタイマで 独立動作
0	1	設定禁止
1	0	32 ビットタイマ
1	1	設定禁止

＜注意事項＞
FMD[2:0]を 000 にしてから、本ビットを変更してください。(いったん FMD[2:0]を 000 に設定した後、T32 ビットと FMD[2:0]を同時に所要の値に設定します。)

[bit6～bit4] FMD[2:0] (Function MoDe) : **タイマ機能選択ビット**
ベースタイマの機能を選択するビットです。本ビットを変更する際は、いったん 000(リセットモード)を経由してから、ほかのモードへと設定してください。

FMD[2:0]	説明
000	リセットモード (FMD=000 書込みにより、ベースタイマをリセット後の状態に戻します。各レジスタは初期値に戻ります。)
001	16 ビット PWM タイマ
010	16 ビット PPG タイマ
011	16/32 ビットリロードタイマ
100	16/32 ビット PWC タイマ
101 110 111	設定禁止

[bit3] OSEL (Output SELect) : **出力極性指定ビット**
本ビットがセットされていると、TOUT から出力される信号レベル(H/L)が反転されます。

OSEL	説明
0	通常出力
1	反転出力

[bit2] MDSE (MoDe Select) : **モード選択ビット**
【リロードタイマ・PWM】

MDSE	説明
0	リロードモード : ダウンカウンタがアンダフローすると、ベースタイマ x 周期設定レジスタ(BTxPCSR)の値をリロードしてカウント動作を続けます。
1	ワンショットモード : ダウンカウンタがアンダフローすると、カウント動作を停止します。

【PPG】

MDSE	説明
0	リロードモード : "L"レベルと"H"レベルの信号を連続して出力(連続パルス)するモードです。
1	ワンショットモード : "L"レベルと"H"レベルの信号を 1 回ずつ出力(単一パルス)するモードです。

【PWC】

MDSE	説明
0	連続測定モード : 1 回測定が終わると、測定開始エッジが入力されるまで待機し、再度測定開始エッジが入力されると測定を行うモードです。
1	単発測定モード : 測定を 1 回のみ行うモードです。

[bit1] CTEN (CounT ENable) : **カウンタ動作許可ビット**
カウンタ動作を許可・禁止します。

＜注意事項＞
入出力モード 4, 入出力モード 6 のタイマ動作中に偶数チャネルから立下りエッジが出力されると奇数チャネルの本ビットは 0 にクリアされます。

CTEN	説明	
	読出し	書込み
0	停止中	本ビットを 0 にします。
1	動作許可中	本ビットを 1 にします。

[bit0] STRG (Software TRiGger) : **ソフトウェアトリガビット**
タイマ起動などのトリガとして機能します。

PWC のとき読出し値は"0"です。PWC のとき、このビットへは"0"を書き込んでください。

＜注意事項＞
- 本ビットを書き込む際に、ほかのビットをクリアしてしまわないように注意してください。
- CTEN, FMD[2:0]との同時書込みは、動作許可と同時にトリガを発行します。

STRG	説明
0	無視されます。
1	トリガを発行します。

19.4.1.3 入出力選択レジスタ ：BTSEL01 (Base Timer SElect register ch.0 and ch.1)

入出力選択レジスタ (BTSEL01) のビット構成について説明します。

ベースタイマの ch.0 および ch.1 の入出力モードを設定するビットです。

＜注意事項＞

- このレジスタは 8 ビットでアクセスしてください。
- このレジスタは、ベースタイマ x タイマ制御レジスタ(BTxTMCR)の FMD2~FMD0 ビットで、ベースタイマリセットモードに設定(FMD2~FMD0=000)してから書き換えてください。

■ BTSEL01： アドレス 009CH (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				SEL01[3:0]			
初期値	1	1	1	1	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W

[bit3～bit0] SEL01[3:0] (SELect) : ch.0/ch.1 用入出力選択ビット
ベースタイマの ch.0 および ch.1 の入出力モードを設定するビットです。

SEL01[3:0]	説明
0000	入出力モード 0 (16 ビットタイマ標準モード)
0001	入出力モード 1 (32 ビットタイマフルモード)
0010	入出力モード 2 (外部トリガ共有モード)
0011	設定禁止
0100	入出力モード 4 (タイマ起動/ 停止モード)
0101	入出力モード 5 (同時ソフト起動モード)
0110	入出力モード 6 (ソフト起動タイマ起動/ 停止モード)
0111	入出力モード 7 (タイマ起動モード)
1xxx	設定禁止

19.4.1.4 同時ソフト起動レジスタ ：BTSSSR (Base Timer Software Synchronous Start Register)

同時ソフト起動レジスタ (BTSSSR) のビット構成について説明します。

入出力モード 5, 6 における入力信号になります。本レジスタを使用すれば、すべてのチャネルに同時にトリガを発生させることができます。

■ BTSSSR： アドレス 009E$_H$ (アクセス: バイト, ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	1	1	1	1	1	1	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約						SSSR1	SSSR0
初期値	1	1	1	1	1	1	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,W	R1,W

[bit1] SSSR1 (Software Synchronous Start Register ch.1) : 同時ソフト起動ビット ch.1

[bit0] SSSR0 (Software Synchronous Start Register ch.0) : 同時ソフト起動ビット ch.0

入出力モード 5, 6 における入力信号になります。接続は図 19-3 各入出力モード配線図(2)を参照してください。

SSSR0/1	説明
0	動作に影響ありません。
1	"1"パルスを入力に当て、対応したチャネルを起動します。

19.4.2　16/32 ビットリロードタイマ時のレジスタ

16/32 ビットリロードタイマ時のレジスタについて説明します。

19.4.2.1. ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

19.4.2.2. 周期設定レジスタ 0, 1 : BTxPCSR (Base Timer 0/1 Pulse Counter Start Register)

19.4.2.1 ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

ステータス制御レジスタ 0, 1 (BTxSTC) のビット構成について説明します。

割込み要求を制御するレジスタです。

＜注意事項＞

- 予約ビットには"0"を書き込んでください。
- TGIR, UDIR ビットへのリードモディファイライト系命令では"1"が読み出されます。
- このレジスタは 8 ビットでアクセスしてください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxSTC : アドレス Base_addr + 05_H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	TGIE	予約	UDIE	予約	TGIR	予約	UDIR
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R/W	R0,W0	R/W	R0,W0	R(RM1),W	R0,W0	R(RM1),W

[bit6] TGIE (TriGger Interrupt Enable) : **トリガ割込み要求許可ビット**
16/32 ビットリロードタイマの起動トリガが検出されたとき(TGIR=1)に、トリガ割込み要求を発生させるかどうかを設定します。

[bit4] UDIE (UnDerflow Interrupt Enable) : **アンダフロー割込み要求許可ビット**
ダウンカウンタがアンダフローしたとき(UDIR=1)にアンダフロー割込み要求を発生させるかどうかを設定します。

TGIE/UDIE	説明
0	禁止します
1	許可します

[bit2] TGIR (TriGger Interrupt Register) : **トリガ割込み要求フラグビット**
16/32 ビットリロードタイマの起動トリガが検出されたことを示します。このビットが"1" のときに、TGIE ビットに"1" が設定されていると，トリガ割込み要求が発生します。

[bit0] UDIR (UnDerflow Interrupt Register) : **アンダフロー割込み要求フラグビット**
ダウンカウンタの値が"0000_H" から"$FFFF_H$" に変わり，アンダフローしたことを示します。このビットが"1" のときに、UDIE ビットに"1" が設定されていると，アンダフロー割込み要求が発生します。

TGIR/UDIR	読出し時	書込み時
0	トリガ検出/アンダフローはありません。	このビットをクリアします。
1	トリガ検出/アンダフローがありました。	動作に影響しません

19.4.2.2 周期設定レジスタ 0, 1 : BTxPCSR (Base Timer 0/1 Pulse Counter Start Register)

周期設定レジスタ 0, 1 (BTxPCSR) のビット構成について説明します。

16/32 ビットリロードタイマの周期を設定するバッファ付きレジスタです。ダウンカウンタが、このレジスタに設定した値からカウントダウンします。

＜注意事項＞

- このレジスタは 16 ビットでアクセスしてください。
- タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットで、ベースタイマの機能を 16/32 ビットリロードタイマに選択(FMD2～FMD0=011)してから、このレジスタを設定してください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxPCSR ：アドレス Base_addr + 08H (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	D[15:0]					
初期値	X	X	・・・	X	X	X
属性	R/W	R/W	・・・	R/W	R/W	R/W

[bit15～bit0] D[15:0] (Data) : **データビット**
16/32 ビットリロードタイマの周期を設定するバッファ付きレジスタです。ダウンカウンタが、このレジスタに設定した値からカウントダウンします。

このレジスタに設定した値は、次の場合に 16 ビットダウンカウンタにロードされます。

- 16/32 ビットリロードタイマの起動時
- ダウンカウンタのアンダフロー時

2チャネルの16ビットリロードタイマをカスケード接続して、32ビットリロードタイマとして使用する場合は、このレジスタに設定した値は次のようになります。

- 偶数チャネルの周期設定レジスタ(BTxPCSR)の値 ： 下位 16 ビットの値
- 奇数チャネルの周期設定レジスタ(BTxPCSR)の値 ： 上位 16 ビットの値

そのため、32 ビットタイマモード時にこのレジスタに値を書き込む場合は、次の順番で値を書き込んでください。

1. 奇数チャネルのベースタイマ x 周期設定レジスタ(BTxPCSR)
2. 偶数チャネルのベースタイマ x 周期設定レジスタ(BTxPCSR)

19.4.3　16 ビット PWM タイマ時のレジスタ

16 ビット PWM タイマ時のレジスタについて説明します。

19.4.3.1. ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

19.4.3.2. 周期設定レジスタ 0, 1 : BTxPCSR (Base Timer 0/1 Pulse Counter Start Register)

19.4.3.3. デューティ設定レジスタ 0, 1 : BTxPDUT (Base Timer 0/1 Pulse DuTy register)

19.4.3.1 ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

ステータス制御レジスタ 0, 1 (BTxSTC) のビット構成について説明します。

割込み要求を制御するレジスタです。

＜注意事項＞

- 予約ビットには"0"を書き込んでください。
- TGIR, DTIR, UDIR へのリードモディファイライト系命令では"1"が読み出されます。
- このレジスタは 8 ビットでアクセスしてください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxSTC : アドレス Base_addr + 05H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	TGIE	DTIE	UDIE	予約	TGIR	DTIR	UDIR
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R/W	R/W	R/W	R0,W0	R(RM1),W	R(RM1),W	R(RM1),W

[bit6] TGIE (TriGger Interrupt Enable) : **トリガ割込み要求許可ビット**
16 ビット PWM タイマの起動トリガが検出されたとき(TGIR=1)にトリガ割込み要求を発生させるかどうかを設定します。

[bit5] DTIE (DuTy Interrupt Enable) : **デューティー致割込み要求許可ビット**
16 ビットダウンカウンタの値が, ベースタイマ x デューティ設定レジスタ(BTxPDUT)の値と一致したとき(DTIR=1)にデューティ一致割込み要求を発生させるかどうかを設定します。

[bit4] UDIE (UnDerflow Interrupt Enable) : **アンダフロー割込み要求許可ビット**
ダウンカウンタがアンダフローしたとき(UDIR=1)にアンダフロー割込み要求を発生させるかどうかを設定します。

TGIE/DTIE/UDIE	説明
0	禁止します
1	許可します

[bit2] TGIR (TriGger Interrupt Register) : **トリガ割込み要求フラグビット**
16 ビット PWM タイマの起動トリガが検出されたことを示します。このビットが"1"のときに TGIE ビットに"1"が設定されていると、トリガ割込み要求発生します。

[bit1] DTIR (DuTy Interrupt Register) : **デューティー致割込み要求フラグビット**
16 ビットダウンカウンタの値がデューティ設定レジスタ(BTxPDUT)と一致したこと(デューティが一致したこと)を示します。このビットが"1" のときに DTIE ビットに"1"が設定されているとデューティ一致割込み要求が発生します。

[bit0] UDIR (UnDerflow Interrupt Register) : **アンダフロー割込み要求フラグビット**
16 ビットダウンカウンタの値が"0000_H"から"$FFFF_H$"に変わりアンダフローが発生したことを示します。このビットが"1"のときに UDIE ビットに"1"が設定されているとアンダフロー割込み要求が発生します。

TGIR/DTIR/UDIR	読出し時	書込み時
0	トリガ検出/デューティ一致/ アンダフローはありません。	このビットをクリアします。
1	トリガ検出/デューティ一致/ アンダフローがありました。	動作に影響ありません。

19.4.3.2 周期設定レジスタ 0, 1 : BTxPCSR (Base Timer 0/1 Pulse Counter Start Register)

周期設定レジスタ 0, 1 (BTxPCSR) のビット構成について説明します。

16 ビット PWM タイマの周期を設定するバッファ付きレジスタです。16 ビットダウンカウンタがこのレジスタに設定した値からカウントダウンします。カウンタの値がこのレジスタに設定した値と一致すると、出力信号(TOUT)のレベルが反転します。

＜注意事項＞

- このレジスタは 16 ビットでアクセスしてください。
- タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットで、ベースタイマの機能を 16 ビット PWM タイマに選択してから、このレジスタを設定してください。
- このレジスタを書き換えた場合、必ずデューティ設定レジスタ(BTxPDUT)も書き換えてください。
- このレジスタには デューティ設定レジスタ(BTxPDUT)より小さな値は設定しないでください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxPCSR : アドレス Base_addr + 08_H (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	D[15:0]					
初期値	X	X	・・・	X	X	X
属性	R/W	R/W	・・・	R/W	R/W	R/W

[bit15～bit0] D[15:0] (Data) : **データビット**
16 ビット PWM タイマの周期を設定するバッファ付きレジスタです。16 ビットダウンカウンタがこのレジスタに設定した値からカウントダウンします。カウンタの値がこのレジスタに設定した値と一致すると、出力信号(TOUT)のレベルが反転します。

このレジスタはバッファ付きのため、カウント動作中に書き換えることができます。このレジスタに設定した値は、次の場合に 16 ビットダウンカウンタにロードされます。

- 16 ビット PWM タイマの起動時
- 16 ビットダウンカウンタのアンダフロー時

このレジスタとベースタイマ x デューティ設定レジスタ(BTxPDUT)に同じ値を設定すると、出力信号(TOUT)のレベルを固定できます。出力される信号レベルはベースタイマ x タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって次のようになります。

- OSEL=0 : "H" レベル
- OSEL=1 : "L" レベル

19.4.3.3 デューティ設定レジスタ 0, 1 : BTxPDUT (Base Timer 0/1 Pulse DuTy register)

デューティ設定レジスタ 0, 1 (BTxPDUT)のビット構成について説明します。

16 ビット PWM タイマのデューティを設定するバッファ付きレジスタです。16 ビットダウンカウンタがこのレジスタに設定した値と一致すると出力信号(TOUT)のレベルが反転します。

＜注意事項＞

- このレジスタは 16 ビットでアクセスしてください。
- タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットで、ベースタイマの機能を 16 ビット PWM タイマに選択してから、このレジスタを設定してください。
- このレジスタを書き換えた場合、必ず周期設定レジスタ(BTxPCSR)より大きい値は設定しないでください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxPDUT : アドレス Base_addr + 0A$_H$ (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	D[15:0]					
初期値	0	0	・・・	0	0	0
属性	R/W	R/W	・・・	R/W	R/W	R/W

[bit15～bit0] D[15:0] (Data) : **データビット**
16 ビット PWM タイマのデューティを設定するバッファ付きレジスタです。16 ビットダウンカウンタがこのレジスタに設定した値と一致すると出力信号(TOUT)のレベルが反転します。

このレジスタはバッファ付きのため、カウント動作中に書き換えることができます。

16 ビットダウンカウンタがアンダフローすると、バッファの値が転送されます。

このレジスタとベースタイマ x 周期設定レジスタ(BTxPCSR)に同じ値を設定すると、出力信号(TOUT)のレベルを固定できます。出力される信号レベルはベースタイマ x タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって次のようになります。

- OSEL=0 : オール"H"レベル
- OSEL=1 : オール"L"レベル

19.4.4　16 ビット PPG タイマ時のレジスタ

16 ビット PPG タイマ時のレジスタについて説明します。

19.4.4.1. ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

19.4.4.2. L 幅設定レジスタ 0, 1 : BTxPRLL (Base Timer 0/1 Pulse Length of "L" register)

19.4.4.3. H 幅設定レジスタ 0, 1 : BTxPRLH (Base Timer 0/1 Pulse Length of "H" register)

19.4.4.1 ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

ステータス制御レジスタ 0, 1 (BTxSTC) のビット構成について説明します。

割込み要求を制御するレジスタです。

＜注意事項＞

- 予約ビットには"0"を書き込んでください。
- TGIR, UDIR へのリードモディファイライト系命令では"1"が読み出されます。
- このレジスタは 8 ビットでアクセスしてください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxSTC : アドレス Base_addr + 05H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	TGIE	予約	UDIE	予約	TGIR	予約	UDIR
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R/W	R0,W0	R/W	R0,W0	R(RM1),W	R0,W0	R(RM1),W

[bit6] TGIE (TriGger Interrupt Enable) : **トリガ割込み要求許可ビット**
16 ビット PPG タイマの起動トリガが検出されたとき(TGIR=1)にトリガ割込み要求を発生させるかどうかを設定します。

[bit4] UDIE (UnDerflow Interrupt Enable) : **アンダフロー割込み要求許可ビット**
ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)のカウントダウンが終了し、アンダフローが発生したとき(UDIR=1)にアンダフロー割込み要求を発生させるかどうかを設定します。

TGIE/UDIE	説明
0	禁止します
1	許可します

[bit2] TGIR (TriGger Interrupt Register) : **トリガ割込み要求フラグビット**
16 ビット PPG タイマの起動トリガが検出されたことを示します。このビットが"1"のときに TGIE ビットに"1"が設定されていると、トリガ割込み要求が発生します。

[bit0] UDIR (UnDerflow Interrupt Register) **アンダフロー割込み要求フラグビット**
ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)のカウントダウンが終了しアンダフローが発生したことを示します。16 ビットダウンカウンタの値が"0000_H"の状態でさらにカウントダウンしようとするとアンダフローが発生します。このビットが"1" のときに、UDIE ビットに"1"が設定されていると、アンダフロー割込み要求が発生します。

TGIR/UDIR	読出し時	書込み時
0	トリガ検出/アンダフローはありません。	このビットをクリアします。
1	トリガ検出/アンダフローがありました。	動作に影響ありません。

19.4.4.2 L 幅設定レジスタ 0, 1 : BTxPRLL (Base Timer 0/1 Pulse Length of "L" register)

L 幅設定レジスタ 0, 1 (BTxPRLL) のビット構成について説明します。

16 ビット PPG タイマから初めに出力する信号レベルの幅を設定するレジスタです。

＜注意事項＞

- このレジスタは 16 ビットでアクセスしてください。
- タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットで、ベースタイマの機能を PPG タイマに選択してから、このレジスタを設定してください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxPRLL： アドレス Base_addr + 08H (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	D[15:0]					
初期値	X	X	・・・	X	X	X
属性	R/W	R/W	・・・	R/W	R/W	R/W

[bit15～bit0] D[15:0] (Data)： **データビット**

16 ビット PPG タイマから初めに出力する信号レベルの幅を設定するレジスタです。16 ビットダウンカウンタがこのレジスタに設定した値をカウントダウンし終わると、出力波形(TOUT)のレベルが反転します。このレジスタとベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)を設定することで、出力信号の"L"レベルの幅と"H"レベルの幅が決まります。このレジスタでどちらの信号レベルの幅を設定するかは、タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって次のようになります。

- OSEL=0 の場合："L"レベルの幅
- OSEL=1 の場合："H"レベルの幅

16 ビット PPG タイマの起動トリガ検出時や、ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値のカウントが終了し、アンダフローが発生したときに、このレジスタに設定した値が 16 ビットダウンカウンタにロードされます。

19.4.4.3 H 幅設定レジスタ 0, 1 : BTxPRLH (Base Timer 0/1 Pulse Length of "H" register)

H 幅設定レジスタ 0, 1 (BTxPRLH)のビット構成について説明します。

ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)の値をカウント終了後に、出力される信号レベルの幅を設定するバッファ付きレジスタです。

＜注意事項＞

- このレジスタは 16 ビットでアクセスしてください。
- タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットで、ベースタイマの機能を PPG タイマに選択してから、このレジスタを設定してください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxPRLH : アドレス Base_addr + 0A$_H$ (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	D[15:0]					
初期値	X	X	・・・	X	X	X
属性	R/W	R/W	・・・	R/W	R/W	R/W

[bit15～bit0] D[15:0] (Data) : **データビット**
L 幅設定リロードレジスタ(BTxPRLL)の値をカウント終了後に、出力される信号レベルの幅を設定するバッファ付レジスタです。16 ビットダウンカウンタがこのレジスタに設定した値をカウントダウンし終わると、出力波形(TOUT)の信号レベルが反転します。

このレジスタとベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)を設定することで、出力信号の"L"レベルの幅と"H"レベルの幅が決まります。このレジスタでどちらの信号レベルの幅を設定するかは、ベースタイマ x タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって次のようになります。

- OSEL=0 の場合："H"レベルの幅
- OSEL=1 の場合："L"レベルの幅

このレジスタはバッファ付きのため、カウント動作中に書き換えることができます。このレジスタの値の転送タイミングは次のとおりです。

- バッファへの転送
  - 16 ビット PPG タイマの起動トリガ検出時
  - ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値のカウントが終了し、アンダフローが発生したとき
- 16 ビットダウンカウンタへの転送
  - ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)の値のカウントが終了したとき

書き換えるタイミングについては、「19.5.6.3. リロードモード時の動作」の「■ 書込みタイミング」を参照してください。

19.4.5　16/32 ビット PWC タイマ時のレジスタ

16/32 ビット PWC タイマ時のレジスタについて説明します。

19.4.5.1. ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

19.4.5.2. データバッファレジスタ 0, 1 : BTxDTBF (Base Timer 0/1 DaTa BuFfer register)

19.4.5.1 ステータス制御レジスタ 0, 1 : BTxSTC (Base Timer 0/1 STatus Control)

ステータス制御レジスタ 0, 1 (BTxSTC) のビット構成について説明します。

割込み要求を制御するレジスタです。

＜注意事項＞

- 予約ビットには"0"を書き込んでください。
- OVIR へのリードモディファイライト系命令では"1"が読み出されます。
- このレジスタは 8 ビットでアクセスしてください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxSTC : アドレス Base_addr + 05H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ERR	EDIE	予約	OVIE	予約	EDIR	予約	OVIR
初期値	0	0	0	0	0	0	0	0
属性	R,W0	R/W	R0,W0	R/W	R0,W0	R,WX	R0,W0	R(RM1),W

[bit7] ERR (ERRor) : **エラーフラグビット**
連続測定モードで動作中に データバッファレジスタ(BTxDTBF)から測定結果を読み出す前に次の測定が終了し、測定結果が新しい値に上書きされたことを示します。このとき上書きされる前の値は破棄されます。データバッファレジスタ(BTxDTBF)を読み出すとこのビットが"0"にクリアされます。

ERR	説明
0	測定結果は上書きされていません。
1	測定結果は上書きされました。

[bit6] EDIE (EnD Interrupt Enable) : **測定終了割込み要求許可ビット**
16/32 ビット PWC タイマの測定が終了したとき(EDIR ビット=1)に、測定終了割込み要求を発生させるかどうかを設定します。

[bit4] OVIE (OVerflow Interrupt Enable) : **オーバフロー割込み要求許可ビット**
アップカウンタがオーバフローしたとき(OVIR ビット=1)にオーバフロー割込み要求を発生させるかどうかを設定します。

EDIE/OVIE	説明
0	禁止します
1	許可します

[bit2] EDIR (EnD Interrupt Register) : **測定終了割込み要求フラグビット**
16/32 ビット PWC タイマの測定が終了したことを示します。このビットが"1" のときに EDIE ビットに"1" が設定されていると測定終了割込み要求が発生します。本ビットは測定結果 (BTxDTBF)を読出しすることによりクリアされます。

[bit0] OVIR (OVerflow Interrupt Register) : **オーバフロー割込み要求フラグビット**

アップカウンタの値が"$FFFF_H$"から"0000_H"に変わり、オーバフローが発生したことを示します。このビットが"1"のときに、OVIE ビットに"1" が設定されていると、オーバフロー割込み要求が発生します。本ビットは"0"書込みによりクリアされます。

EDIR/OVIR	読出し時	書込み時
0	測定終了/オーバフローはありません。	(EDIR) 動作に影響ありません。 (OVIR)このビットをクリアします。
1	測定終了/オーバフローがありました。	動作に影響ありません。

19.4.5.2 データバッファレジスタ 0, 1 : BTxDTBF (Base Timer 0/1 DaTa BuFfer register)

データバッファレジスタ 0, 1 (BTxDTBF) のビット構成について説明します。

16/32 ビット PWC タイマの測定値やアップカウンタの値を読み出すために利用するレジスタです。

＜注意事項＞

- このレジスタは 16 ビットでアクセスしてください。
- 本レジスタはリセットモードへの設定(BTxTMCR:FMD=000 書込み)によっても初期化されます。

■ BTxDTBF : アドレス Base_addr + 0A$_H$ (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	D[15:0]					
初期値	0	0	・・・	0	0	0
属性	R,WX	R,WX	・・・	R,WX	R,WX	R,WX

[bit15～bit0]: D[15:0] (Data) : **データビット**
16/32 ビット PWC タイマの測定値やアップカウンタの値を読み出すために利用するレジスタです。このレジスタは、単発測定モード時と連続測定モード時で読み出す値が異なります。

- 単発測定モード時 : アップカウンタの動作中はアップカウンタの値を、測定終了後は測定結果を読み出します。
- 連続測定モード時 : アップカウンタの動作中/測定終了後とも前回測定した結果を読み出します。アップカウンタの値を読み出すことはできません。

2 チャネルの 16 ビット PWC タイマをカスケード接続して、32 ビット PWC タイマとして使用する場合は、このレジスタの値は次のようになります。

- 偶数チャネルのデータバッファレジスタ(BTxDTBF)の値 : 下位 16 ビットの値
- 奇数チャネルのデータバッファレジスタ(BTxDTBF)の値 : 上位 16 ビットの値

そのため、32 ビットタイマモード時は、次の順番でこのレジスタを読み出してください。

1. 偶数チャネルのデータバッファレジスタ(BTxDTBF)
2. 奇数チャネルのデータバッファレジスタ(BTxDTBF)

19.5 動作説明

ベースタイマの動作について説明します。

19.5.1. タイマ機能の選択

19.5.2. 入出力割当て

19.5.3. 32 ビットモード動作

19.5.4. 16/32 ビットリロードタイマの動作

19.5.5. 16 ビット PWM タイマの動作

19.5.6. 16 ビット PPG タイマの動作

19.5.7. 16/32 ビット PWC タイマの動作

19.5.1 タイマ機能の選択

タイマ機能の選択について説明します。

BTxTMCR:FMD[2:0]で、使用するタイマ機能を選択してください。

19.5.2 入出力割当て

入出力割当てについて説明します。

タイマを使用する前に、BTSEL01 レジスタでベースタイマの入出力設定を行ってください。次の 7 とおりから選択できます。

- **入出力モード 0**

16 ビットタイマ標準モード

ベースタイマを 1 チャネルごとに個別に動作させるモードです。

- **入出力モード 1**

32 ビットタイマフルモード

ベースタイマの偶数チャネルの信号を個別に外部端子に割り当てて動作させるモードです。

- **入出力モード 2**

外部トリガ共有モード

2 チャネルのベースタイマに対して同時に外部起動トリガを入力できるモードです。このモードを利用すると、2 チャネルのベースタイマを同時に起動できます。

- **入出力モード 4**

タイマ起動/ 停止モード

偶数チャネルで奇数チャネルの起動/ 停止を制御するモードです。奇数チャネルは、偶数チャネルからの出力信号の立上りエッジ(*)で起動し、立下りエッジ(*)で停止します。

- **入出力モード 5**

同時ソフト起動モード

ソフトウェアで複数のチャネルを同時に起動するモードです。

- **入出力モード 6**

ソフト起動タイマ起動/ 停止モード

偶数チャネルで奇数チャネルの起動/ 停止を制御するモードです。偶数チャネルはソフトウェアで起動します。奇数チャネルは、偶数チャネルからの出力信号の立上りエッジ(*)で起動し、立下りエッジ(*)で停止します。

- **入出力モード 7**

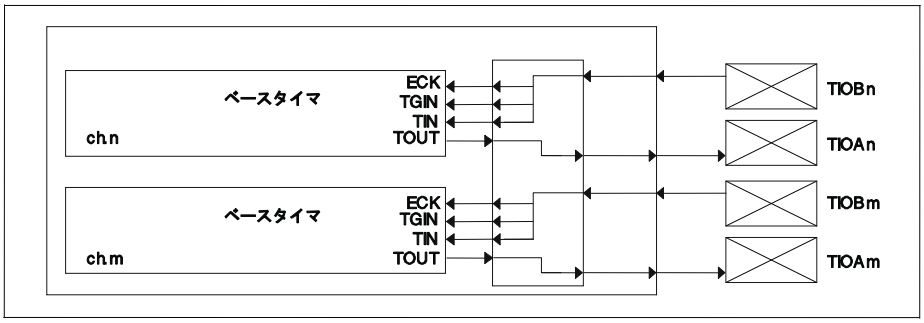
タイマ起動モード

偶数チャネルで奇数チャネルの起動を制御するモードです。奇数チャネルは、偶数チャネルからの出力信号の立上りエッジ(*)で起動します。

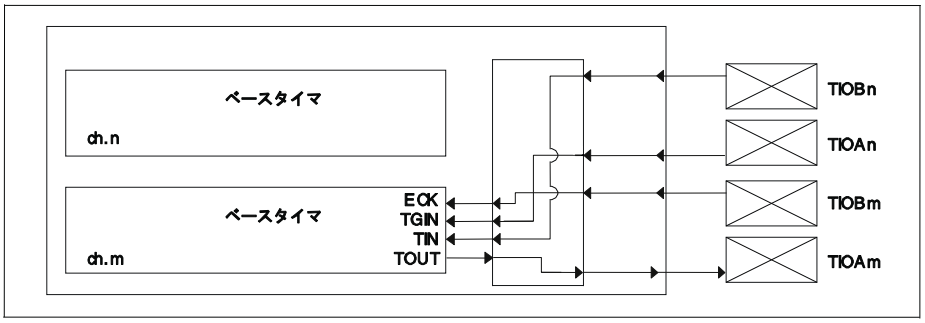
(*)：トリガ入力選択ビット(BTxTMCR:EGS)で設定してください。

図 19-2 各入出力モード配線図(1)

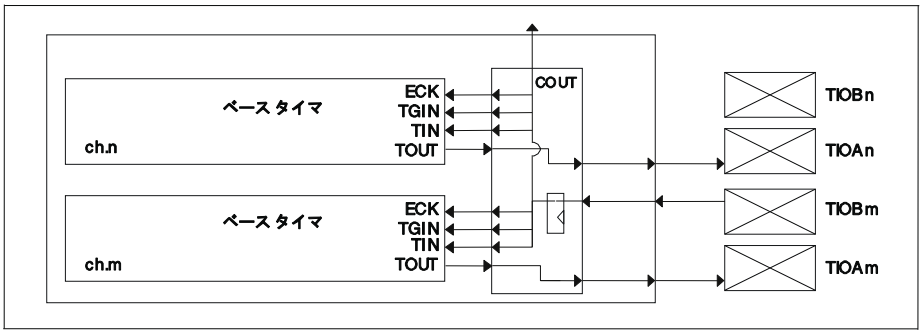
入出力モード 0（16 ビットタイマ標準モード）のブロックダイヤグラム



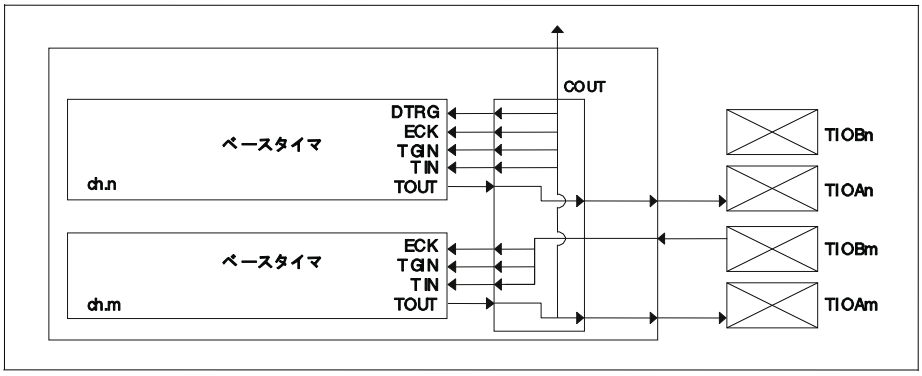
入出力モード 1（32 ビットタイマフルモード）のブロックダイヤグラム



入出力モード 2（外部トリガ共有モード）のブロックダイヤグラム

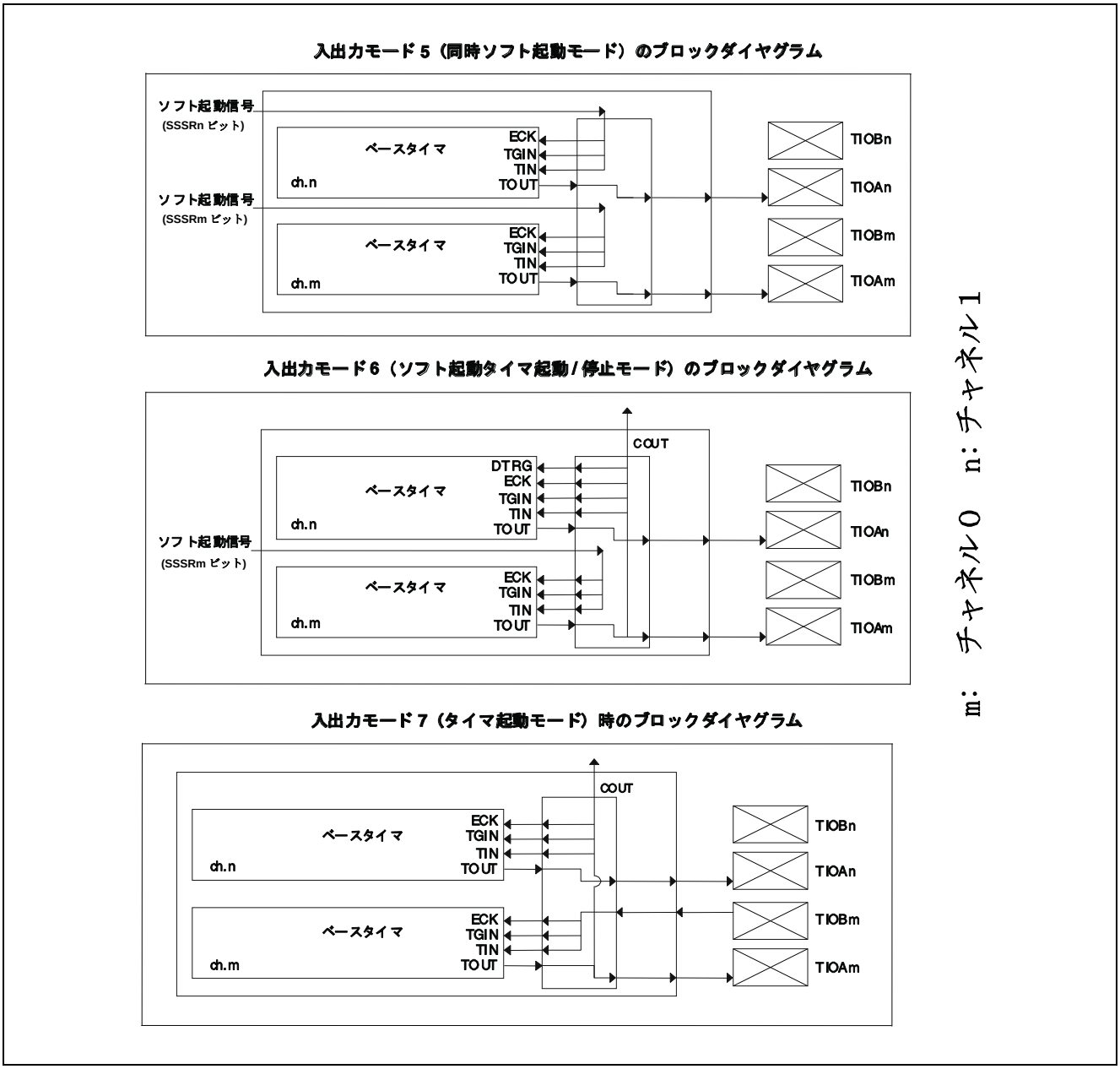


入出力モード 4（タイマ起動 / 停止モード）のブロックダイヤグラム



m: チャネル 0　n: チャネル 1

図 19-3 各入出力モード配線図(2)



19.5.3 32 ビットモード動作

32 ビットモード動作について説明します。

リロードタイマ, PWC タイマは、2 チャネルを使用して 32 ビットモード動作が可能です。以下に、32 ビットモード機能における基本機能/動作について示します。

19.5.3.1 32 ビットモード機能

32 ビットモード機能について説明します。

ベースタイマを 2 チャネル組み合せて 32 ビットデータのリロードタイマまたは 32 ビットデータの PWC タイマ動作を実現する機能です。偶数チャネルの下位 16 ビットタイマ・カウンタ値を読み出す際に、奇数チャネルの上位 16 ビットタイマ・カウンタ値も取り込むので、動作中のタイマ・カウンタ値も読み出すことが可能です。

19.5.3.2 32 ビットモード設定

32 ビットモード設定について説明します。

まず、偶数チャネルの BTxTMCR レジスタの FMD ビットを"000"でリセットモードにして状態をリセットしてから、 16 ビットモード時と同様にリロードタイマ、または PWC タイマ選択と動作の設定を行います。このとき、BTxTMCR レジスタの T32 ビットにも"1"を書き込むことで 32 ビット動作モードに設定します。奇数チャネルの T32 ビットは"0"のままにしてください。リセットモードの設定も必要ありません。次に、リロードタイマの場合は、奇数チャネルの周期設定レジスタに 32 ビットのうち、上位 16 ビットのリロード値を設定し、その後に偶数チャネルの周期設定レジスタに下位 16 ビットのリロード値を設定します。

32 ビット動作モードへの移行は T32 ビット書込み後、直ちに反映されるので、設定変更は両チャネルともカウント停止状態で行ってください。

32 ビットモードから 16 ビットモードへの移行は、偶数チャネルの BTxTMCR レジスタの FMD ビットを"000"でリセットモードにして偶数、奇数の両チャネルの状態をリセットし、チャネルごとに 16 ビットモードでの設定を行います。

19.5.3.3 32 ビットモード動作

32 ビットモード動作について説明します。

32 ビットモード設定の後、偶数チャネルの制御によりリロードタイマ、または PWC タイマを起動すると、偶数チャネルのタイマ/ カウンタは下位 16 ビット動作となり、奇数チャネルのタイマ/ カウンタは上位 16 ビット動作となります。

32 ビットモードでの動作は偶数チャネルの設定に従うので、奇数チャネルの設定は (リロードタイマ時の周期設定レジスタを除き) 無視します。タイマ起動, 波形出力, 割込み信号も偶数チャネルのものが有効となります (奇数チャネルは L 固定にマスクされます)。

構成は図 19-12 32 ビットタイマモード時の構成および図 19-29 32 ビットタイマモード時の構成を参照してください。

19.5.4 16/32 ビットリロードタイマの動作

16/32 ビットリロードタイマの動作について説明します。

本シリーズに内蔵されているベースタイマを16/32 ビットリロードタイマとして使用する場合の動作について説明します。また、各動作状態を設定するための手順例も示します。

図 19-4 ブロックダイヤグラム(16 ビットリロードタイマ動作時)

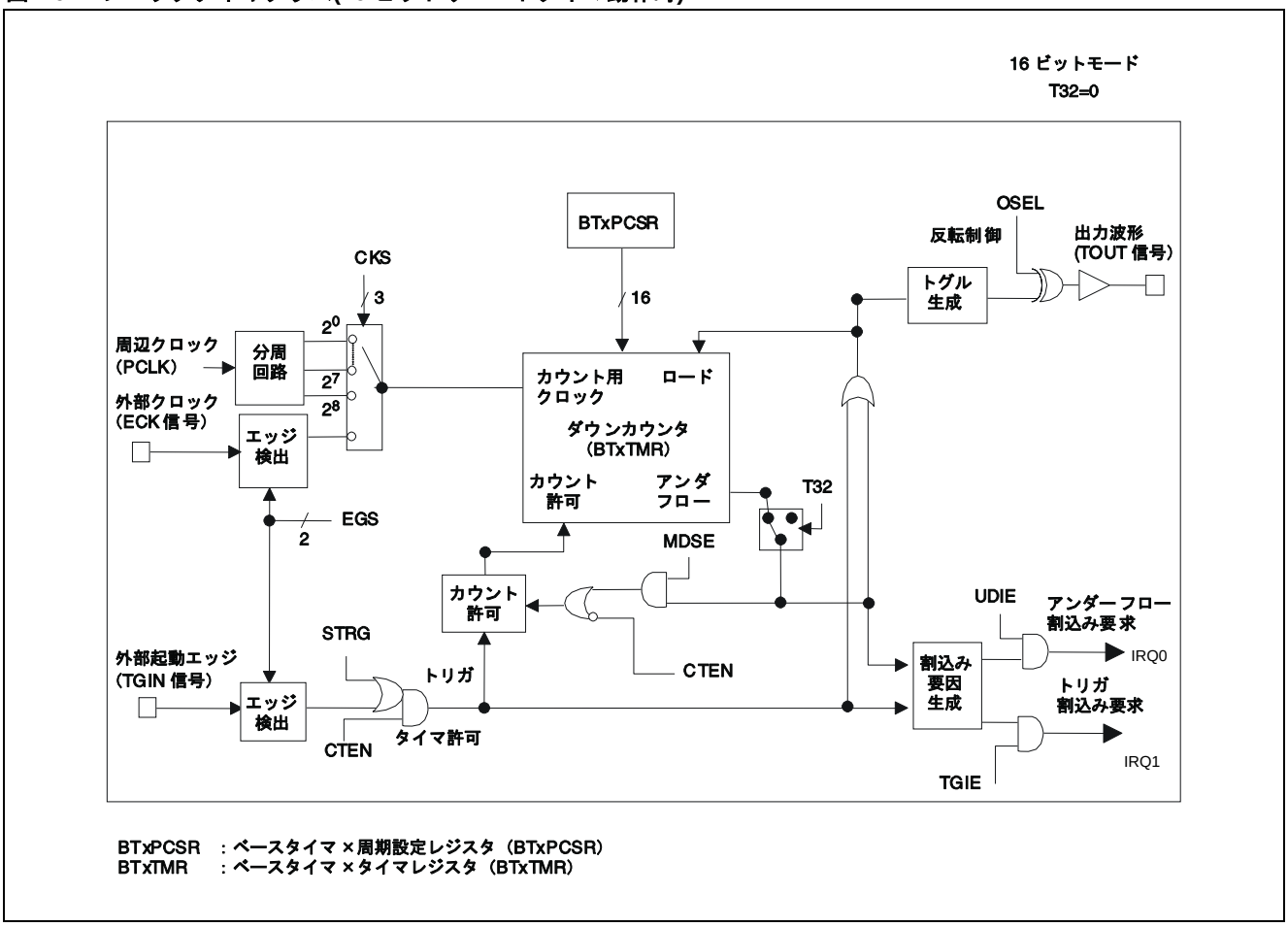
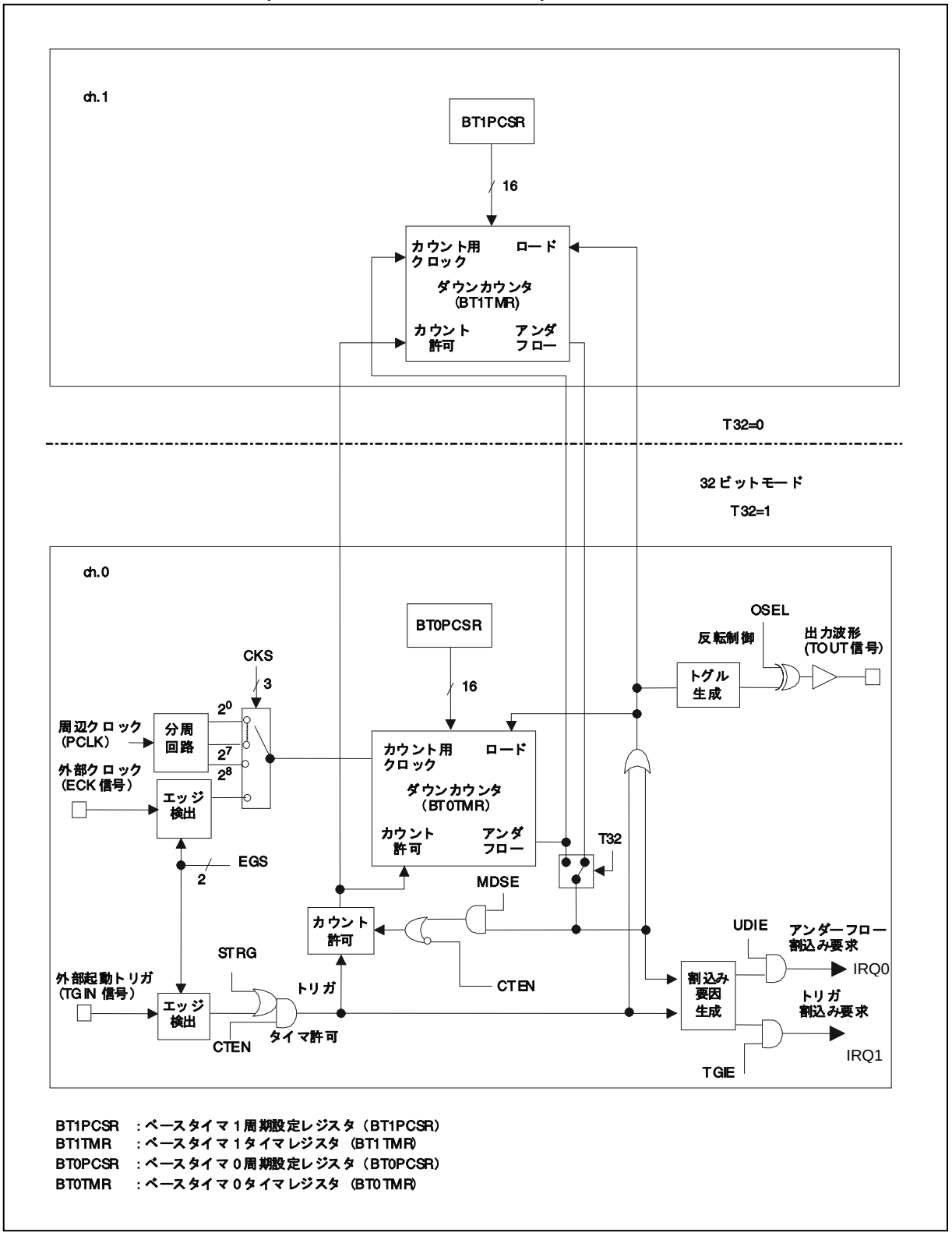


図 19-5 ブロックダイヤグラム(32 ビットリロードタイマ動作時)



19.5.4.1 概要

16/32 ビットリロードタイマの概要について説明します

16/32 ビットリロードタイマは、ベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値からカウントダウンするタイマです。ダウンカウンタがアンダフローしたときに、アンダフロー割込み要求を発生させる機能があります。

16/32 ビットリロードタイマには、タイマモードと動作モードの 2 つのモードがあり、それぞれのモードの組み合わせによって、動作が異なります。

- タイマモード : ベースタイマ x タイマ制御レジスタ(BTxTMCR)の T32 ビットで次の 2 種類のモードから選択します。
  - 16 ビットタイマモード(T32=0) : 16 ビットリロードタイマを 1 チャネルずつ個別に動作させます。
  - 32 ビットタイマモード(T32=1) : 2 チャネルをカスケード接続して 32 ビットリロードタイマとして使用します。
- 動作モード : ベースタイマ x タイマ制御レジスタ(BTxTMCR)の MDSE ビットで次の 2 種類のモードから選択します。
  - リロードモード(MDSE=0) : ダウンカウンタがアンダフローすると、設定している値(周期)をリロードしてカウントを繰り返すモードです。
  - ワンショットモード(MDSE=1) : ダウンカウンタがアンダフローすると、カウントを停止するモードです。

19.5.4.2 リロードモード時の動作

リロードモード時の動作について説明します。

リロードモード時の動作について説明します。

■ 概要

アンダフローが発生するたびに、ベースタイマ x 周期設定レジスタ(BTxPCSR)の値をリロードして、カウントダウンを継続するモードです。このモードを利用するには、ベースタイマ x タイマ制御レジスタ(BTxTMCR)の MDSE ビットでリロードモードを設定(MDSE=0)してください。

■ 動作

● 起動

次の手順で 16/32 ビットリロードタイマを起動してください。

1. ベースタイマ x タイマ制御レジスタ(BTxTMCR)の CTEN ビットで, 16/32 ビットリロードタイマの動作を許可(CTEN=1)する
   16/32 ビットリロードタイマが起動トリガ待ち状態になります。

2. 次のいずれかの方法で起動トリガを入力する
   - □ ベースタイマ x タイマ制御レジスタ(BTxTMCR)の STRG ビットに"1"を書き込む(ソフトウェアトリガ)
   - □ 外部起動トリガ(TGIN 信号)の有効エッジ(EGS1, EGS0 ビットで設定したエッジ)を入力する

＜注意事項＞

- 外部起動トリガ(TGIN 信号)の入力方法は、入出力選択レジスタ(BTSEL01)で設定した入出力モードによって異なります。「19.5.2 入出力割当て」を参照してください。
- 動作を許可すると同時にカウントを開始したい場合はベースタイマ x タイマ制御レジスタ(BTxTMCR)の CTEN ビットと STRG ビットの両方に"1"を書き込んでください。

カウント動作

起動トリガが入力されると、次の時間の経過後にベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値(周期)がダウンカウンタにロードされ、カウントダウンが開始されます。

- ソフトウェアトリガ入力時 : 1T(T: カウント用クロックの周期)
- 外部起動トリガ(TGIN 信号)入力時 : 2T～3T(T: カウント用クロックの周期)

カウントの開始タイミングを次に示します。

図 19-6 カウントの開始タイミング(ソフトウェアトリガ)

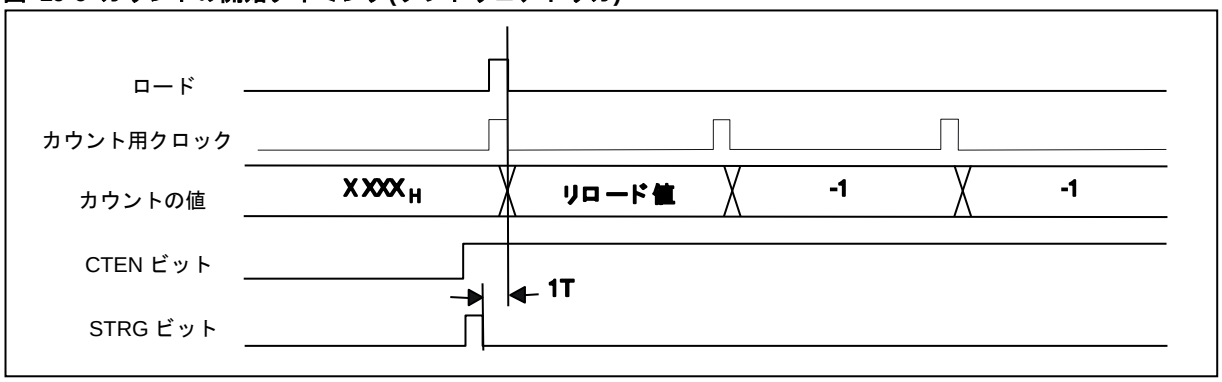
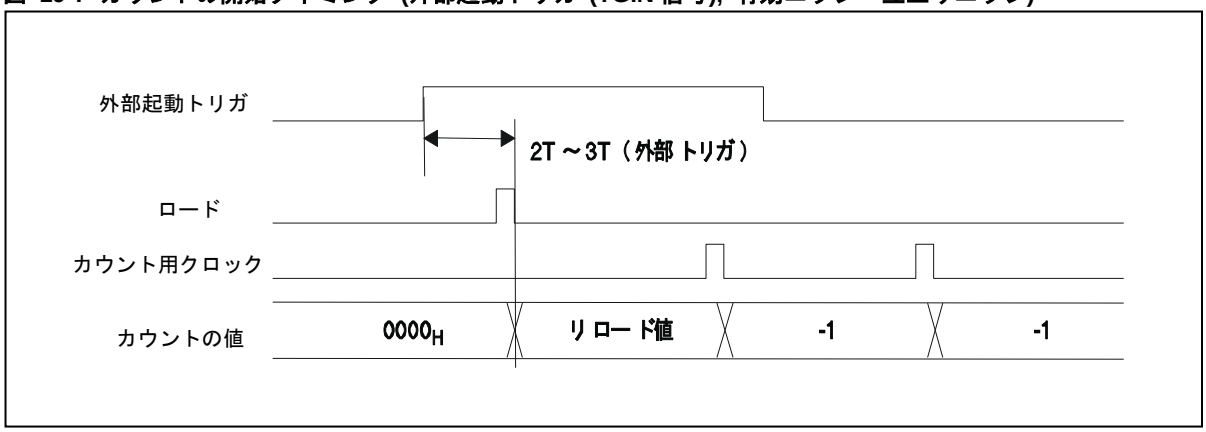


図 19-7 カウントの開始タイミング (外部起動トリガ (TGIN 信号), 有効エッジ= 立上りエッジ)

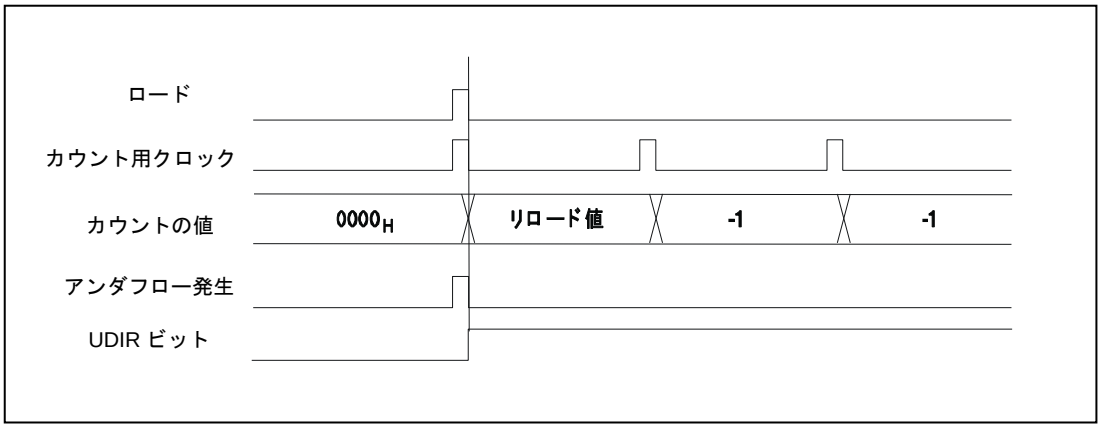


＜注意事項＞

外部起動トリガ(TGIN 信号)の入力方法は、入出力選択レジスタ(BTSEL01)で設定した入出力モードによって異なります。「19.5.2 入出力割当て」を参照してください。

ダウンカウンタの値が"0000_H"からさらにカウントダウンしようとして、アンダフローすると、ベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値(周期)が再びダウンカウンタにロードされ、カウント動作を続けます。また、アンダフローが発生すると、ベースタイマ x ステータス制御レジスタ(BTxSTC)の UDIR ビットが"1" に変わります。このとき、UDIE ビットが"1" に設定されているとアンダフロー割込み要求が発生します。アンダフロー発生時の動作を次に示します。

図 19-8 アンダフロー発生時の動作



■ 出力波形

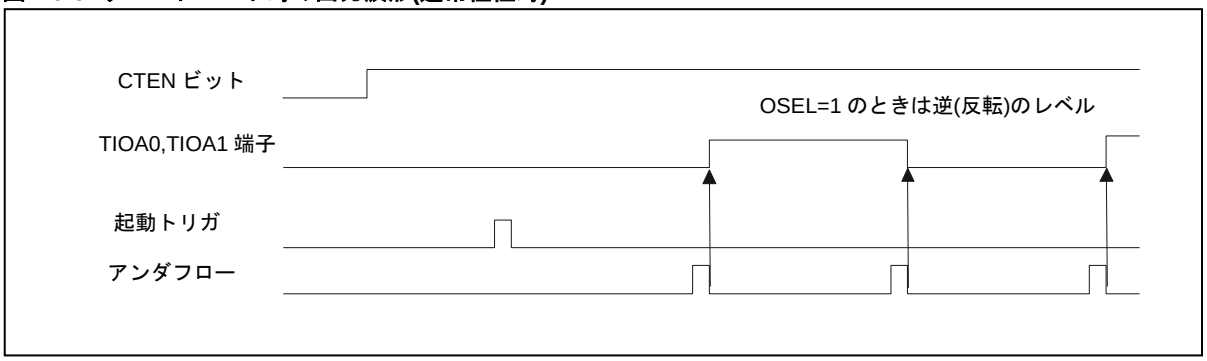
16/32 ビットリロードタイマの波形(TOUT 信号)を出力できます。出力する波形(TOUT 信号)はベースタイマ x タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって異なります。

表 19-3 出力極性と出力波形の対応

出力極性	出力波形
通常極性(OSEL=0)	カウント開始時に"L"レベルを出力 以降、アンダフローが発生するたびに出力レベルが反転
反転極性(OSEL=1)	カウント開始時に"H"レベルを出力 以降、アンダフローが発生するたびに出力レベルが反転

リロードモード時の出力波形を次に示します。

図 19-9 リロードモード時の出力波形(通常極性時)



19.5.4.3 ワンショットモード時の動作

ワンショットモード時の動作について説明します。

ワンショットモード時の動作について説明します。

■ 概要

アンダフローが発生するとカウントダウンを停止するモードです。

このモードを利用するには，ベースタイマ x タイマ制御レジスタ(BTxTMCR)の MDSE ビットでワンショットモードを設定(MDSE=1)してください。

■ 動作

● 起動

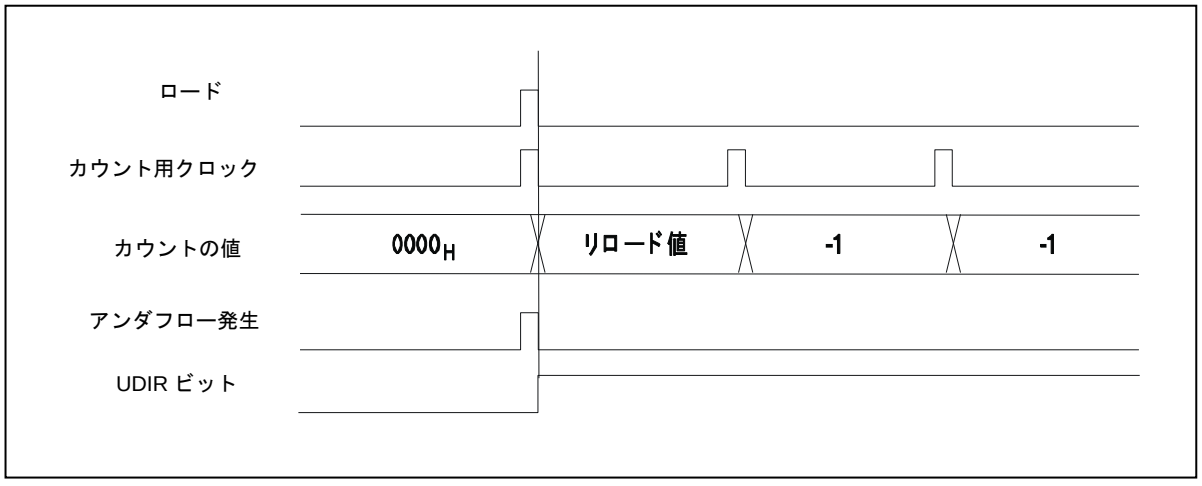
リロードモード時の動作と同様です。「19.5.4.2. リロードモード時の動作」の「■ 動作」を参照してください。

● カウント動作

アンダフローが発生するまでの動作は、リロードモード時の動作と同様です。「■ 動作」を参照してください。アンダフローすると、ベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値(周期)が再びダウンカウンタにロードされます。ただし、ダウンカウンタはカウント動作を停止します。また、アンダフローが発生すると、ベースタイマ x ステータス制御レジスタ(BTxSTC)の UDIR ビットが"1" に変わります。このとき、ベースタイマ x ステータス制御レジスタ(BTxSTC)の UDIE ビットが"1"に設定されているとアンダフロー割込み要求が発生します。

アンダフロー発生時の動作を次に示します。

図 19-10 アンダフロー発生時の動作



■ 出力波形

16/32 ビットリロードタイマの波形(TOUT 信号)を出力できます。出力する波形(TOUT 信号)はベースタイマ x タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって異なります。

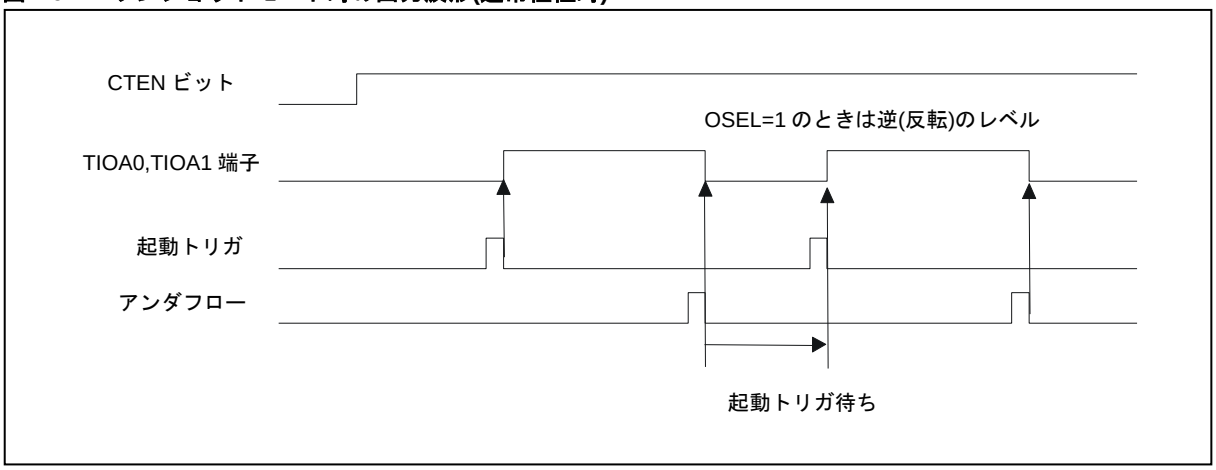
出力極性と出力波形の対応を次に示します。

表 19-4 出力極性と出力波形の対応

出力極性	出力波形
通常極性(OSEL=0)	起動トリガが入力されると(カウント中)"H"レベルを出力 起動トリガ待ち状態時は"L"レベルを出力
反転極性(OSEL=1)	起動トリガが入力されると(カウント中)"L"レベルを出力 起動トリガ待ち状態時は"H"レベルを出力

ワンショットモード時の出力波形を次図に示します。

図 19-11 ワンショットモード時の出力波形(通常極性時)



19.5.4.4 32 ビットタイマモード時の動作

32 ビットタイマモード時の動作について説明します。

16 ビットリロードタイマ 2 チャネルをカスケード接続して、32 ビットのリロードタイマとして使用する場合の設定と動作について説明します。

■ 概要

ベースタイマ x タイマ制御レジスタ(BTxTMCR)の T32 ビットで 16 ビットリロードタイマ 2 チャネルをカスケード接続して、32 ビットのリロードタイマとして使用できます。このモードでは、偶数チャネルが下位 16 ビットの動作に、奇数チャネルが上位 16 ビットの動作に対応します。そのため、リロード値の設定は、上位 16 ビット(奇数チャネル) → 下位 16 ビット(偶数チャネル)の順に、ダウンカウンタの値の読出しは、下位 16 ビット(偶数チャネル) → 上位 16 ビット(奇数チャネル)の順に行ってください。

■ 設定手順例

32 ビットタイマモードを設定する場合は、偶数チャネルのベースタイマ x タイマ制御レジスタ(BTxTMCR)の T32 ビットを"1"に、奇数チャネルのベースタイマ x タイマ制御レジスタ(BTxTMCR)の T32 ビットを"0"に設定してください。また、32 ビットタイマモード設定時は次の手順で各レジスタを設定してください。偶数チャネルと奇数チャネルでレジスタの設定が異なります。カスケード接続する場合を例に説明します。

3. ベースタイマ 0 タイマ制御レジスタ(BT0TMCR)の FMD2～FMD0 ビットで ch.0 をリセットモードに設定(FMD2～FMD0=000)
4. ch.0 と ch.1 のベースタイマ x タイマ制御レジスタ(BT0TMCR, BT1TMCR)の FMD2～FMD0 ビットで、ch.0 と ch.1 を 16/32 ビットリロードタイマを設定(FMD2～FMD0=011)
   同時にベースタイマ 0 タイマ制御レジスタ(BT0TMCR)の T32 ビットで 32 ビットタイマモードを設定(T32=1)
5. ベースタイマ 1 周期設定レジスタ(BT1PCSR)に上位 16 ビットのリロード値を設定
6. ベースタイマ 0 周期設定レジスタ(BT0PCSR)に下位 16 ビットのリロード値を設定

＜注意事項＞

- T32 ビットは、偶数チャネル/ 奇数チャネル両方の動作が停止している状態で書き換えてください。カウント動作が停止しているかどうかは、ベースタイマ x タイマ制御レジスタ(BTxTMCR)の CTEN ビットで確認できます(CTEN=0)。
- ベースタイマ x 周期設定レジスタ(BTxPCSR)にリロード値を設定する際は、必ず奇数チャネル→ 偶数チャネルの順番で設定してください。

■ 動作

32 ビットタイマモード設定時も、カウント動作は基本的に 16 ビットタイマモード時と同様です。

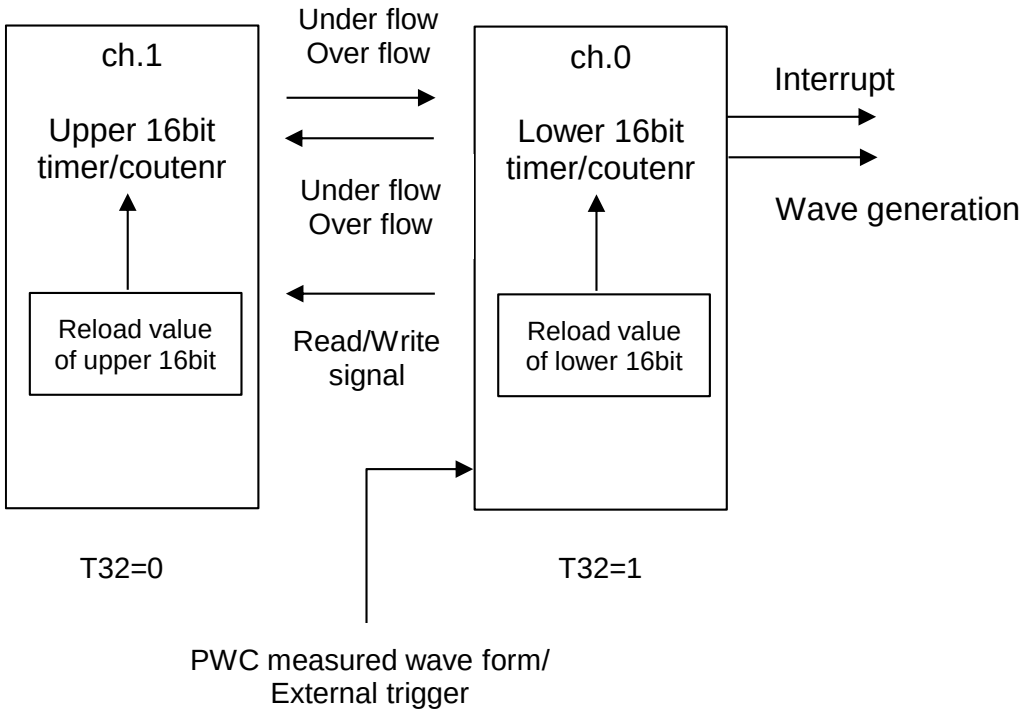
ただし、カウント動作は偶数チャネルの設定に従うため、奇数チャネルの次のレジスタの設定は無視されます。

- ベースタイマ x タイマ制御レジスタ(BTxTMCR)
- ベースタイマ x ステータス制御レジスタ(BTxSTC)

32 ビットタイマモード時のカウント動作について説明します。

1. 32 ビットリロードタイマが起動すると、奇数チャネルのベースタイマ x 周期設定レジスタ(BTxPCSR)と偶数チャネルのベースタイマ x 周期設定レジスタ(BTxPCSR)の値(下位 16 ビット)がダウンカウンタにロードされます。
2. 偶数チャネルを下位 16 ビット、奇数チャネルを上位 16 ビットの 32 ビットカウンタとして、ダウンカウンタがカウント動作を開始します。
3. ダウンカウンタがアンダフローすると、偶数チャネルのベースタイマ x タイマ制御レジスタ(BTxTMCR)の UDIR ビットが"1"に変わります。32 ビットタイマモード時のチャネル構成を次に示します。

図 19-12 32 ビットタイマモード時の構成



＜注意事項＞

- ダウンカウンタの値は、ベースタイマ x タイマレジスタ(BTxTMR)を読み出すことで確認できます。32 ビットタイマモード時は必ず下位 16 ビット(偶数チャネル) → 上位 16 ビット(奇数チャネル)の順で読み出してください。
- 32 ビットタイマモード時は、32 ビットリロードタイマの動作は偶数チャネルの設定に従います。そのため、起動トリガや割込み要求は偶数チャネルのものが有効になります。また、奇数チャネルの端子からの出力信号(TOUT)は"L"レベルに固定されます。

19.5.4.5 割込み

ベースタイマの割込みについて説明します。

次のいずれかの場合に割込み要求が発生します。

- 起動トリガ検出時(トリガ割込み要求)
- アンダフロー発生時(アンダフロー割込み要求)

表 19-5 割込み発生条件

割込み要求	割込み要求フラグ	割込み要求許可	割込み要求のクリア
トリガ割込み要求	BTxSTC の TGIR=1	BTxSTC の TGIE=1	BTxSTC の TGIR ビットに"0"を書き込む。
アンダフロー割込み要求	BTxSTC の UDIR=1	BTxSTC の UDIE=1	BTxSTC の UDIR ビットに"0"を書き込む。

＜注意事項＞

- 割込み要求フラグが"1"のときに割込み要求の発生を許可すると割込みを許可した時点で、割込み要求が発生します。割込み要求の発生を許可する場合は、次のいずれかの処理を行ってください。
  - □ 割込み要求の発生を許可する前に割込み要求をクリアする
  - □ 割込み許可と同時に割込み要求をクリアする。
- 割込み要求は割込み要求の発生を禁止してからクリアするか、割込み処理ルーチン内でクリアしてください。
- 各割込み要求の割込みベクタ番号については、『付録 C 割込みベクタ一覧』を参照してください。
- 割込みベクタ番号に対応する割込みレベルは、割込みコントロールレジスタ(ICR00 ～ICR47)で設定します。割込みレベルの設定については、『割込み制御(割込みコントローラ)』の章を参照してください。

19.5.4.6 使用上の注意

使用上の注意について説明します。

16/32 ビットリロードタイマを使用する際は、次の点に注意してください。

■ プログラムで設定する場合の注意

- ベースタイマ x タイマ制御レジスタ(BTxTMCR)の次のビットは、CTEN ビットで 16 ビットダウンカウンタの動作を停止(CTEN=0)してから書き換えてください。
  - CKS2～CKS0 ビット
  - EGS1, EGS0 ビット
  - T32 ビット
  - FMD2～FMD0 ビット
  - MDSE ビット
- タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットでリセットモードを設定(FMD2～ FMD0=000)すると、すべてのレジスタが初期化されます。
- ベースタイマの機能を変更する場合や T32 ビットを変更する場合は、一度ベースタイマをリセットする必要があります。リセット後に、タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットや T32 ビットを書き換える場合以外は、必ず FMD2～FMD0 ビットでリセットモードを選択(FMD2～FMD0=000)してから、これらのビットを書き換えてください。

■ 動作に関する注意

- ダウンカウンタのカウントのタイミングとロードのタイミングが重なった場合は、ロード動作が優先されます。
- ワンショットモードで、カウント終了時に 16/32 ビットリロードタイマの起動トリガが検出されると、ベースタイマ x 周期設定レジスタ(BTxPCSR)の値(周期)が 16 ビットダウンカウンタにロードされ、カウント動作を開始します。
- ベースタイマ入出力選択機能によって、信号(外部クロック/ 外部起動トリガ/ 波形)の入出力動作が異なります。

■ 割込みに関する注意

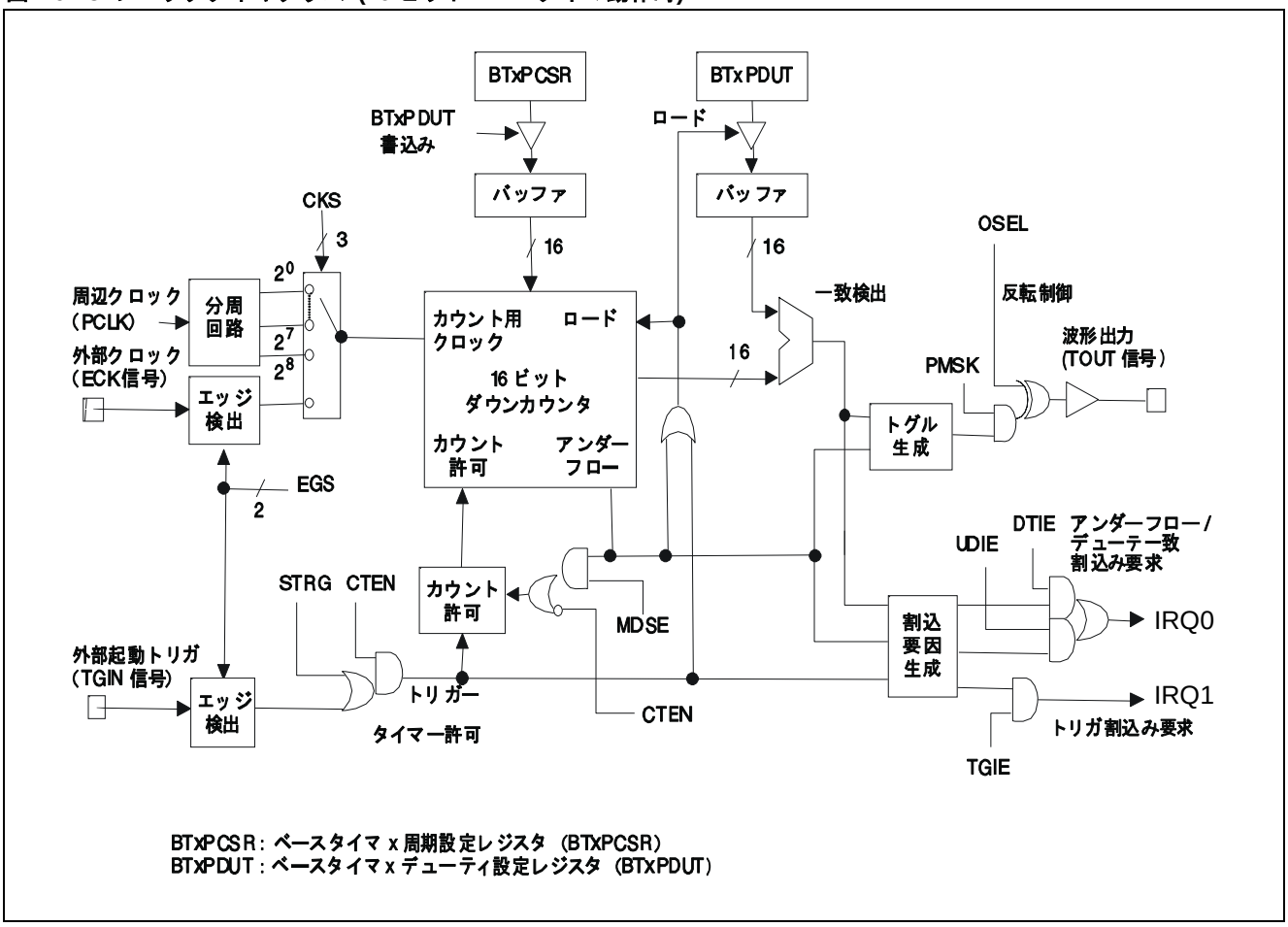
- 割込み要求フラグのクリアと、割込み要求フラグが"1"に変わるタイミングが重なった場合は、割込み要求フラグのクリア動作は無視され、割込み要求フラグは"1"のままになります。

19.5.5　16 ビット PWM タイマの動作

16 ビット PWM タイマの動作について説明します。

本シリーズに内蔵されているベースタイマを 16 ビット PWM タイマとして使用する場合の動作について説明します。また、各動作状態を設定するための手順例も示します。

図 19-13 ブロックダイヤグラム (16 ビット PWM タイマ動作時)



19.5.5.1 概要

16 ビット PWM タイマの概要について説明します。

16 ビット PWM タイマは、周期設定レジスタ(BTxPCSR)に周期を、デューティ設定レジスタ(BTxPDUT)にデューティを設定します。これらのレジスタの値を設定することで任意の波形(TOUT 信号)を出力します。16 ビット PWM タイマは、ベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値からカウントダウンを開始します。ダウンカウンタの値がデューティ設定レジスタ(BTxPDUT)の値と一致すると、出力信号(TOUT)のレベルを反転させます。ダウンカウンタがアンダフローすると再度出力レベルを反転させます。これにより、周期とデューティが任意の波形(TOUT 信号)を出力できます。

16 ビット PWM の動作モードは、タイマ制御レジスタ(BTxTMCR)の MDSE ビットで次の 2 種類から選択できます。

- リロードモード(MDSE=0) : 16 ビットダウンカウンタがアンダフローすると設定してある周期をリロードしてカウントを繰り返すモードです。
- ワンショットモード(MDSE=1) : 16 ビットダウンカウンタがアンダフローするとカウントを停止するモードです。

19.5.5.2 リロードモード時の動作

リロードモード時の動作について説明します。

リロードモード時の動作について説明します。

■ 概要

アンダフローが発生するたびに、ベースタイマ x 周期設定レジスタ(BTxPCSR)の値をリロードして、カウントダウンを継続するモードです。このモードを利用するには、ベースタイマ x タイマ制御レジスタ(BTxTMCR)の MDSE ビットでリロードモードを設定(MDSE=0)してください。

■ 動作

● 起動

次の手順で 16 ビット PWM タイマを起動してください。

1. ベースタイマ x タイマ制御レジスタ(BTxTMCR)の CTEN ビットで、16 ビット PWM タイマの動作を許可(CTEN=1)する
   - □ 16 ビット PWM タイマが起動トリガ待ち状態になります。
2. 次のいずれかの方法で起動トリガを入力する
   - □ ベースタイマ x タイマ制御レジスタ(BTxTMCR)の STRG ビットに"1"を書き込む(ソフトウェアトリガ)
   - □ 外部起動トリガ(TGIN 信号)の有効エッジ(EGS1, EGS0 ビットで設定したエッジ)を入力する

16 ビットダウンカウンタがベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値からカウントダウンを開始します。

＜注意事項＞

- 外部起動トリガ(TGIN 信号)の入力方法は、入出力選択レジスタ(BTSEL01)で設定した入出力モードによって異なります。
- 16 ビット PWM タイマの起動トリガを検出してからベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値が 16 ビットダウンカウンタにロードされるまで、次の時間が必要です。
  - □ ソフトウェアトリガ時 : 1T (T: カウント用クロックの周期)
  - □ 外部イベントトリガ時 : 2T～3T (T: カウント用クロックの周期)

● カウント動作

起動トリガが入力されると、16 ビットダウンカウンタがカウント用クロックに同期して、周期設定レジスタ(BTxPCSR)の値からカウントダウンを開始します。

16 ビットダウンカウンタの値がデューティ設定レジスタ(BTxPDUT)の値と一致すると、次の動作が行われます。

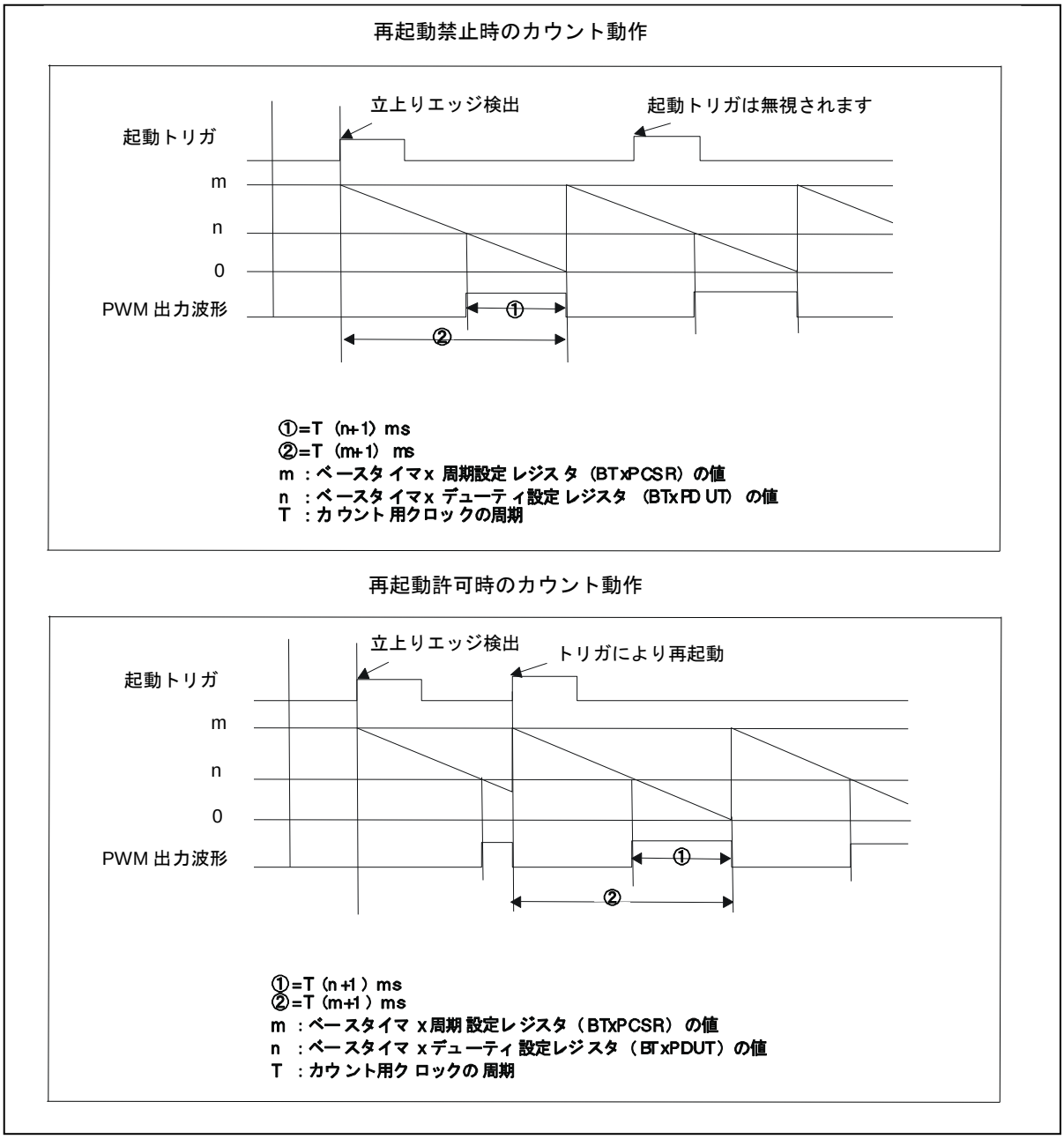
- ステータス制御レジスタ(BTxSTC)の DTIR ビットが"1"に変わる
- 出力信号(TOUT)のレベルが反転する
- カウントダウンを継続する。その後、16 ビットダウンカウンタがアンダフローすると、次の動作が行われます。
- ステータス制御レジスタ(BTxSTC)の UDIR ビットが"1"に変わる出力信号(TOUT)のレベルが反転する
- 周期設定レジスタ(BTxPCSR)の値をリロードし、カウントダウンを継続する

このように、アンダフローが発生するたびに周期設定レジスタ(BTxPCSR)の値をリロードし、カウント動作を続けます。また、カウント中に、起動トリガが入力された場合の動作は、タイマ制御レジスタ(BTxTMCR)の RTGEN ビットで再起動を許可しているかどうかで異なります。

- 再起動禁止(RTGEN=0)の場合 ： カウント中に入力された起動トリガは無視されます。
- 再起動許可(RTGEN=1)の場合 ： ベースタイマ x ステータス制御レジスタ(BTxSTC)の TGIR ビットが"1" に変わります。また、ベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値が 16 ビットダウンカウンタにリロードされ、カウントが開始されます。

それぞれの動作を次に示します。

図 19-14 カウント動作



＜注意事項＞

16 ビットダウンカウンタのカウントのタイミングとロードのタイミングが一致した場合は、ロード動作が優先されます。

■ 出力波形

16 ビット PWM タイマの波形(TOUT 信号)を出力できます。出力する波形(TOUT 信号)はベースタイマ x タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって異なります。

● 通常極性(OSEL=0)時

- 16 ビット PWM タイマ起動時 : "L"レベル
- デューティ一致発生時 : "H"レベル
- アンダフロー発生時 : "L"レベル

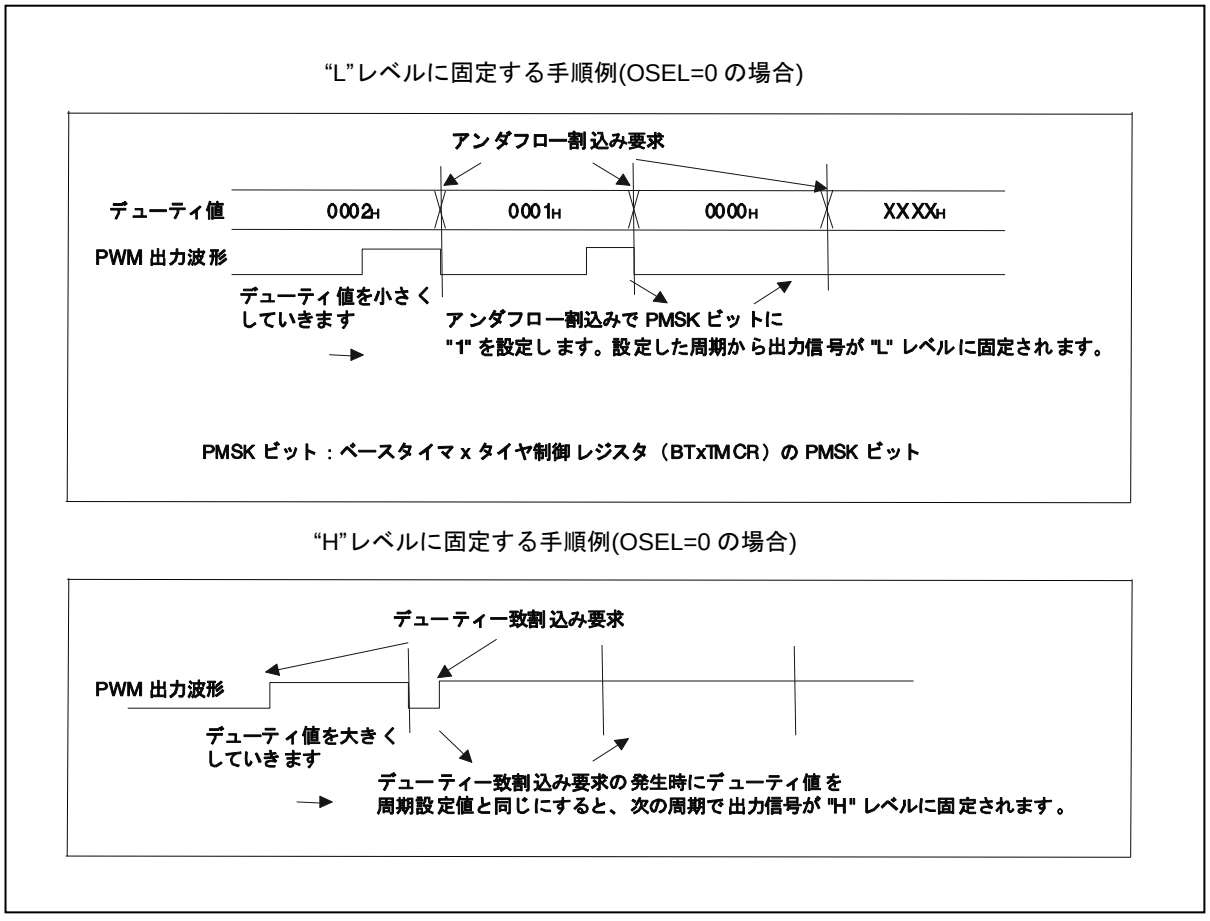
● 反転極性(OSEL=1)時

- 16 ビット PWM タイマ起動時 : "H"レベル
- デューティ一致発生時 : "L"レベル
- アンダフロー発生時 : "H"レベル

また、出力(TOUT 信号)を"L"レベルまたは"H"レベルに固定することもできます。

ベースタイマ x タイマ制御レジスタ(BTxTMCR)の OSEL ビットの設定によって、出力レベルが変わります。手順例を次に示します。

図 19-15 "L・H"レベルに固定する手順例



＜注意事項＞

16 ビット PWM タイマの波形(TOUT 信号)の出力方法/ 出力先は、次の設定によって異なります。

- ベースタイマの入出力モード
- TIOA0, TIOA1 端子機能

■ 割込み発生タイミング

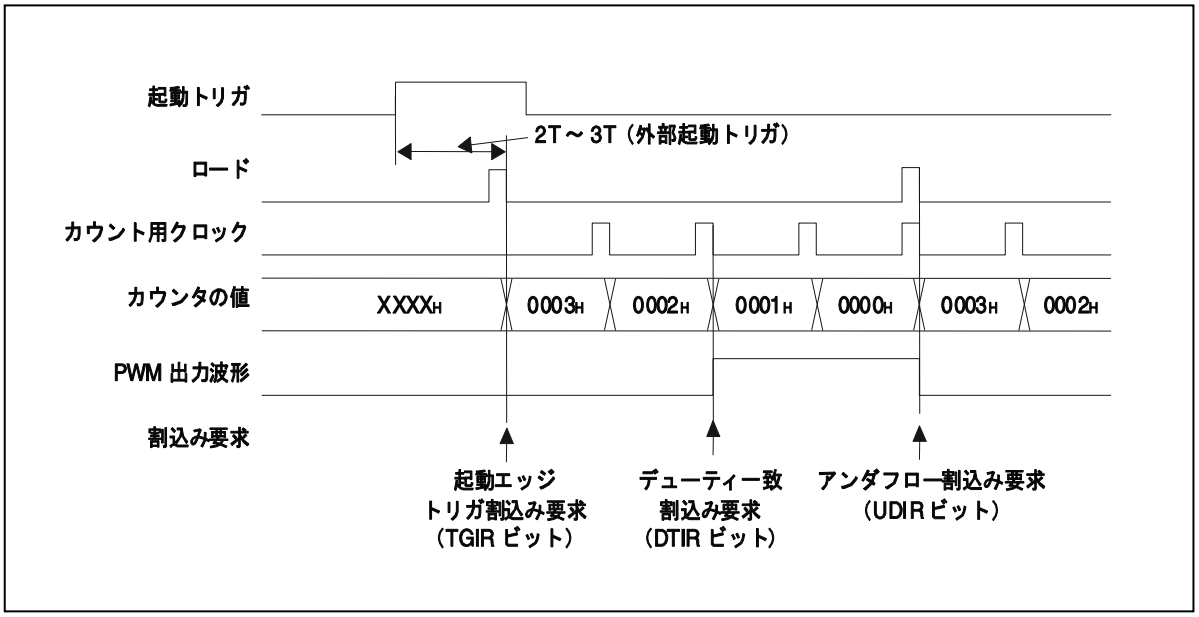
16 ビット PPG タイマは次の場合に割込み要求を発生できます。

- 起動トリガ検出時
- 16 ビットダウンカウンタの値が、ベースタイマ x デューティ設定レジスタ(BTxPDUT)の値と一致したとき
- アンダフロー発生時

割込み要求発生のタイミングを次の設定がされているときを例にとって、次に示します。

- 周期設定レジスタ(BTxPCSR)の値=0003_H
- デューティ設定レジスタ(BTxPDUT)の値=0001_H

図 19-16 割込み要求発生のタイミングチャート



19.5.5.3 ワンショットモード時の動作

ワンショットモード時の動作について説明します。

ワンショットモード時の動作について説明します。

■ カウント動作

16 ビットダウンカウンタの値が周期設定レジスタの設定値(BTxPCSR)から"$FFFF_H$"に変わり、アンダフローが発生するとカウント動作を停止するモードです。

このモードを利用するには、タイマ制御レジスタ(BTxTMCR)の MDSE ビットでワンショットモードを設定(MDSE=1)してください。

● 起動

リロードモード時と同様です。「5.5.2.リロードモード時の動作」の「■ 動作」を参照してください。

● カウント動作

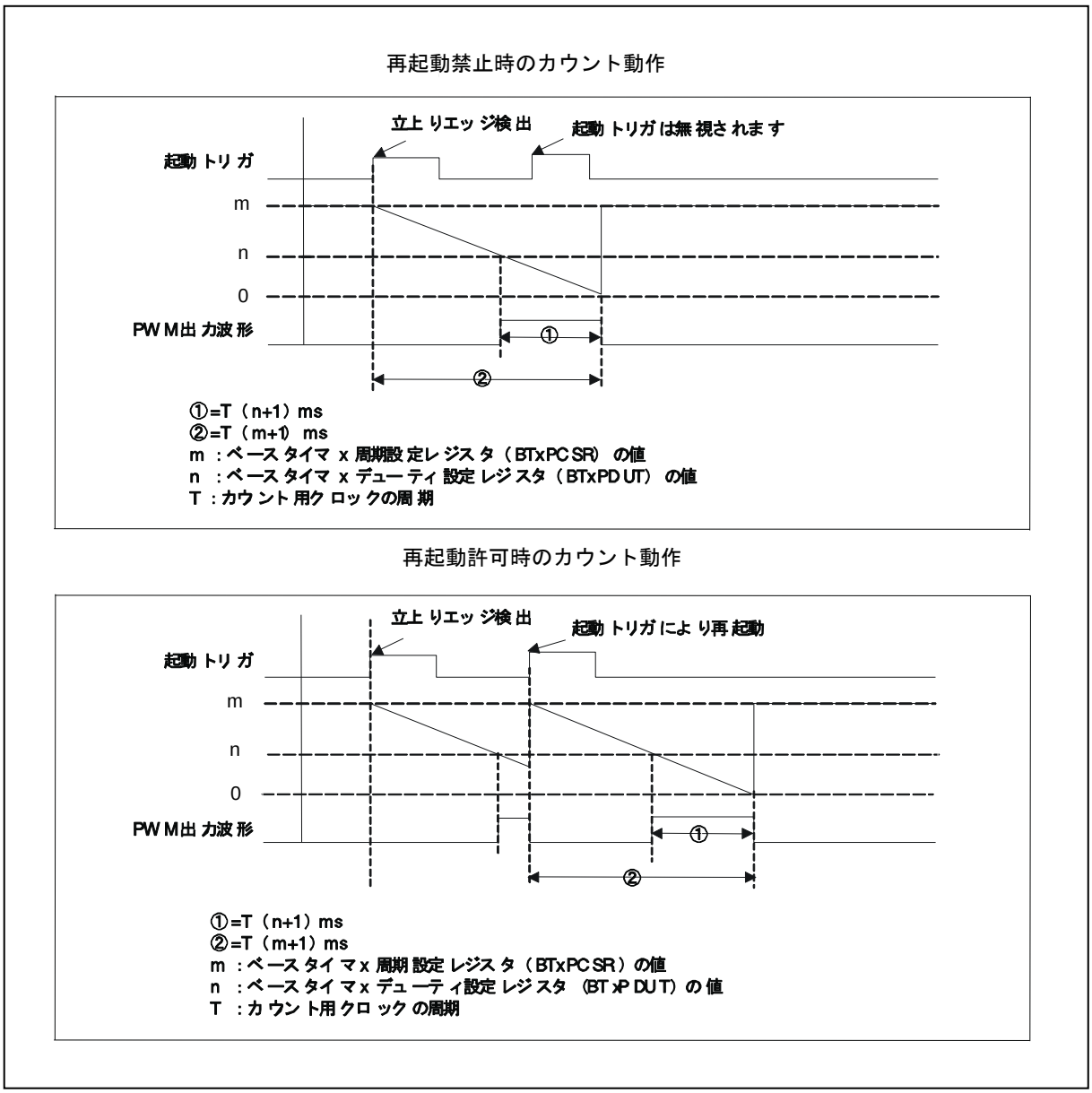
起動トリガが入力されると、16 ビットダウンカウンタがカウント用クロックに同期して、周期設定レジスタ(BTxPCSR)の値からカウントダウンを開始します。16 ビットダウンカウンタの値がデューティ設定レジスタ(BTxPDUT)の値と一致すると、次の動作が行われます。

- ベースタイマ x ステータス制御レジスタ(BTxSTC)の DTIR ビットが"1"に変わる
- 出力信号(TOUT 信号)のレベルが反転する
- カウントダウンを継続する。その後、16 ビットダウンカウンタがアンダフローすると、次の動作が行われます。
- ベースタイマ x ステータス制御レジスタ(BTxSTC)の UDIR ビットが"1"に変わる
- 出力信号(TOUT 信号)のレベルが反転する
- カウント動作を停止する(16 ビットダウンカウンタの値は"$FFFF_H$"で止まります。)

また、カウント中に、起動トリガが入力された場合の動作は、タイマ制御レジスタ(BTxTMCR)の RTGEN ビットで再起動を許可しているかどうかで異なります。

- 再起動禁止(RTGEN=0)の場合：カウント中に入力された起動トリガは無視されます。
- 再起動許可(RTGEN=1)の場合 ： ベースタイマ x ステータス制御レジスタ(BTxSTC)の TGIR ビットが"1"に変わります。また、ベースタイマ x 周期設定レジスタ(BTxPCSR)に設定した値が 16 ビットダウンカウンタにリロードされ、カウントが開始されます。

図 19-17 カウント動作



＜注意事項＞

カウント終了時に、16 ビット PWM タイマの起動トリガが検出されると周期設定レジスタ(BTxPCSR)に設定した値が 16 ビットダウンカウンタにロードされ、カウントを開始します。

■ 出力波形

リロードモード時と同様です。「19.5.5.2. リロードモード時の動作」の「■ 出力波形」を参照してください。

■ 割込み発生タイミング

リロードモード時と同様です。「19.5.5.2. リロードモード時の動作」の「■ 割込み発生タイミング」を参照してください。

19.5.5.4 割込み

割込みについて説明します。

次のいずれかの場合に割込み要求が発生します。

- 起動トリガ検出時 (トリガ割込み要求)
- 16 ビットダウンカウンタの値が (ベースタイマ x デューティ設定レジスタ(BTxPDUT))の値と一致したとき (デューティ一致割込み要求)
- アンダフロー発生時 (アンダフロー割込み要求)

表 19-6 割込み発生条件

割込み要求	割込み要求フラグ	割込み要求許可	割込み要求のクリア
トリガ割込み要求	BTxSTC の TGIR=1	BTxSTC の TGIE=1	BTxSTC の TGIR ビットに"0"を書き込む。
デューティ一致割込み要求	BTxSTC の DTIR=1	BTxSTC の DTIE=1	BTxSTC の DTIR ビットに"0"を書き込む。
アンダフロー割込み要求	BTxSTC の UDIR=1	BTxSTC の UDIE=1	BTxSTC の UDIR ビットに"0"を書き込む。

＜注意事項＞

- 割込み要求フラグが"1"のときに割込み要求の発生を許可すると割込みを許可した時点で、割込み要求が発生します。割込み要求の発生を許可する場合は、次のいずれかの処理を行ってください。
  - □ 割込み要求の発生を許可する前に割込み要求をクリアする
  - □ 割込み許可と同時に割込み要求をクリアする。
- 割込み要求は割込み要求の発生を禁止してからクリアするか、割込み処理ルーチン内でクリアしてください。
- 各割込み要求の割込みベクタ番号については、『付録 C 割込みベクタ一覧』を参照してください。
- 割込みベクタ番号に対応する割込みレベルは、割込みコントロールレジスタ(ICR00 ～ICR47)で設定します。割込みレベルの設定については、『割込み制御(割込みコントローラ)』の章を参照してください。

19.5.5.5 使用上の注意

使用上の注意について説明します。

16 ビット PWM タイマを使用する際は、次の点に注意してください。

■ プログラムで設定する場合の注意

- タイマ制御レジスタ(BTxTMCR)の次のビットは、CTEN ビットで 16 ビットダウンカウンタの動作を停止(CTEN=0)してから、書き換えてください。
  - ☐ CKS2～CKS0 ビット
  - ☐ EGS1, EGS0 ビット
  - ☐ FMD2～FMD0 ビット
  - ☐ MDSE ビット
- ベースタイマ x タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットでリセットモードを設定(FMD2～FMD0=000)すると、すべてのレジスタが初期化されます。
- ベースタイマの機能を変更する場合は、一度ベースタイマをリセットする必要があります。リセット後に、ベースタイマ x タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットを書き換える場合以外は、必ず FMD2～FMD0 ビットでリセットモードを選択(FMD2～FMD0=000)してから、再度 FMD2～FMD0 ビットでベースタイマの機能を選択してください。
- 16 ビット PWM タイマの周期やデューティは次の手順で設定してください。
  1. ベースタイマ x タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットでベースタイマの機能に 16 ビット PWM タイマを設定(FMD2～FMD0=001)
  2. ベースタイマ x 周期設定レジスタ(BTxPCSR)に周期を設定
  3. ベースタイマ x デューティ設定レジスタ(BTxPDUT)にデューティを設定

■ 動作に関する注意

- 16 ビットダウンカウンタのカウントのタイミングとロードのタイミングが重なった場合は、ロード動作が優先されます。
- ワンショットモードでカウント終了時に、16 ビット PWM タイマの再起動トリガが検出されると、ベースタイマ x 周期設定レジスタ(BTxPCSR)の値が 16 ビットダウンカウンタにロードされ、カウントが開始されます。
- ベースタイマ入出力選択機能によって、信号(外部クロック/ 外部起動トリガ/ 波形)の入出力動作が異なります。

■ 割込みに関する注意

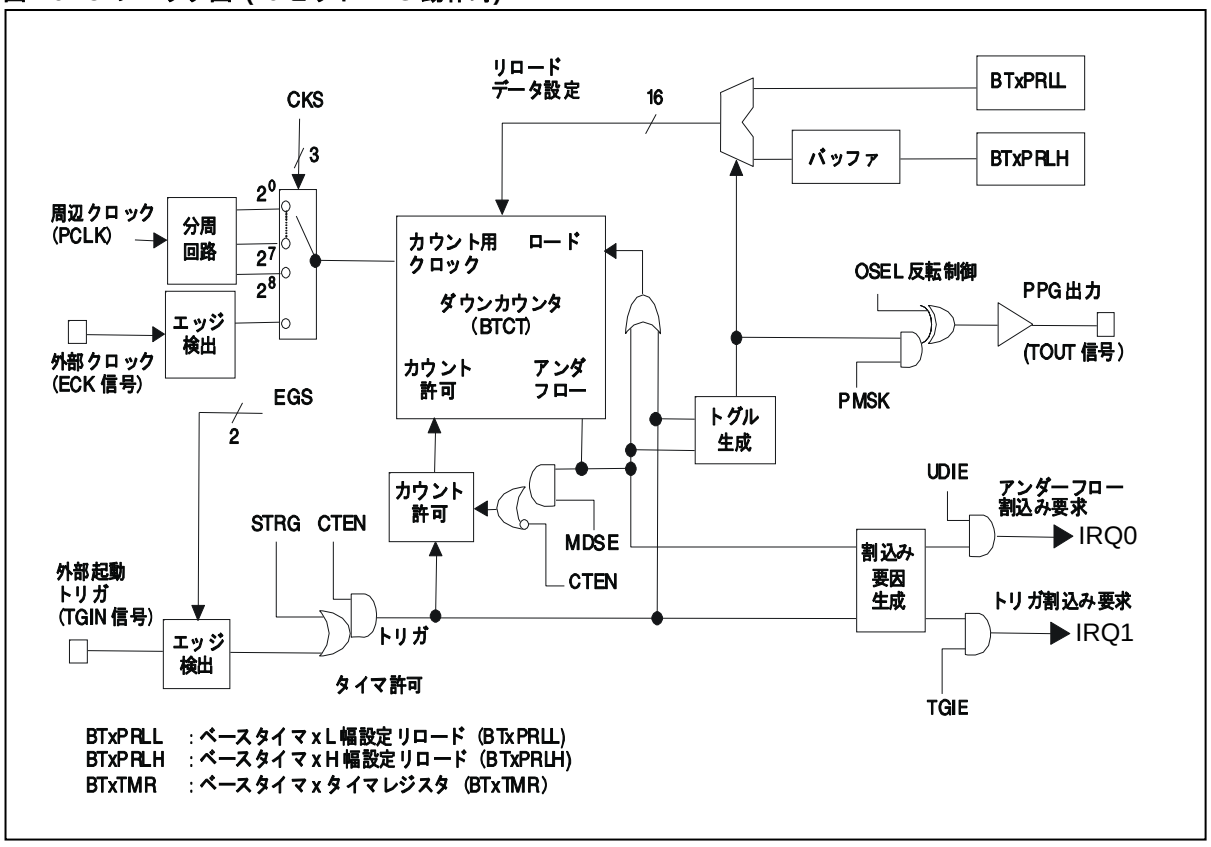
- 割込み要求フラグのクリアと、割込み要求フラグが"1"に変わるタイミングが重なった場合は、 割込み要求フラグのクリア動作は無視され、割込み要求フラグは"1"のままになります。

19.5.6　16 ビット PPG タイマの動作

16 ビット PPG タイマの動作について説明します。

本製品に内蔵されているベースタイマを 16 ビット PPG タイマとして使用する場合の動作について説明します。また、各動作状態を設定するための手順例も示します。

図 19-18 ブロック図 (16 ビット PPG 動作時)



19.5.6.1 概要

16 ビット PPG タイマの概要について説明します。

16 ビット PPG タイマが起動すると、初めにベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)の値をカウントダウンします。L 幅設定リロードレジスタ(BTxPRLL)の値をカウントダウンし終わると、次に H 幅設定リロードレジスタ(BTxPRLH)に設定された値をカウントダウンします。

各レジスタの値をカウント終了後、出力信号(TOUT)のレベルが反転するので、L 幅設定リロードレジスタ(BTxPRLL)H 幅設定リロードレジスタ(BTxPRLH)を設定することで出力する信号の"L"レベルの幅と"H"レベルの幅を任意に設定できます。

16 ビット PPG タイマの動作モードはタイマ制御レジスタ(BTxTMCR)の MDSE ビットで次の 2 種類から選択できます。

- リロードモード(MDSE=0) : "L"レベルと"H"レベルの信号を連続して出力(連続パルス)するモードです。
- ワンショットモード(MDSE=1) : "L"レベルと"H"レベルの信号を 1 回ずつ出力(単一パルス)するモードです。

19.5.6.2 パルス幅の計算方法

パルス幅の計算方法について説明します。

16 ビット PPG タイマは、L 幅設定リロードレジスタ(BTxPRLL)/ ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)に設定した値+1 カウントすると、出力信号(TOUT)のレベルが反転します。そのため、出力される信号のパルス幅は、次の計算式で求められます。

例：出力極性が通常極性の場合

"L"レベルのパルス幅= T × (L+1)
"H"レベルのパルス幅= T × (H+1)
T: カウント用クロックの周期
L: ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)の値
H: ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値

つまり、L 幅設定リロードレジスタ(BTxPRLL)・H 幅設定リロードレジスタ(BTxPRLH)に"0000_H"を設定すると、カウント用クロック 1 周期のパルス幅になります。また、"$FFFF_H$"を設定すると、カウント用クロック 65536 周期のパルス幅が設定されます。

19.5.6.3 リロードモード時の動作

リロードモード時の動作について説明します。

リロードモード時の動作について説明します。

■ 概要

ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)とベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値を交互にリロードして、カウントダウンを継続するモードです。アンダフロー割込み要求の発生タイミングで､ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)/ ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)を書き換えると、任意のパルス幅を連続で出力できます。

このモードを利用するには、ベースタイマ x タイマ制御レジスタ(BTxTMCR)の MDSE ビットでリロードモードを設定(MDSE=0)してください。

■ 動作

● 起動

次の手順で 16 ビット PPG タイマを起動してください。

1. タイマ制御レジスタ(BTxTMCR)の CTEN ビットで、16 ビット PPG タイマの動作を許可(CTEN=1)する。16 ビット PPG タイマが起動トリガ待ち状態になります。
2. 次のいずれかの方法で起動トリガを入力する
   - □ ベースタイマ x タイマ制御レジスタ(BTxTMCR)の STRG ビットに"1"を書き込む(ソフトウェアトリガ)
   - □ 外部起動トリガ(TGIN 信号)の有効エッジ(EGS1, EGS0 ビットで設定したエッジ)を入力する

＜注意事項＞

- ■ 外部起動トリガ(TGIN 信号)の入力方法は、入出力選択レジスタ(BTSEL01)で設定した入出力モードによって異なります。
- ■ 16 ビット PPG タイマの起動トリガを検出してから L 幅設定リロードレジスタ(BTxPRLL)に設定した値(周期)が 16 ビットダウンカウンタにロードされるまで、次の時間が必要です。
  - □ ソフトウェアトリガ時 : 1T (T: カウント用クロックの周期)
  - □ 外部イベントトリガ時 : 2T～3T (T: カウント用クロックの周期)

● カウント動作

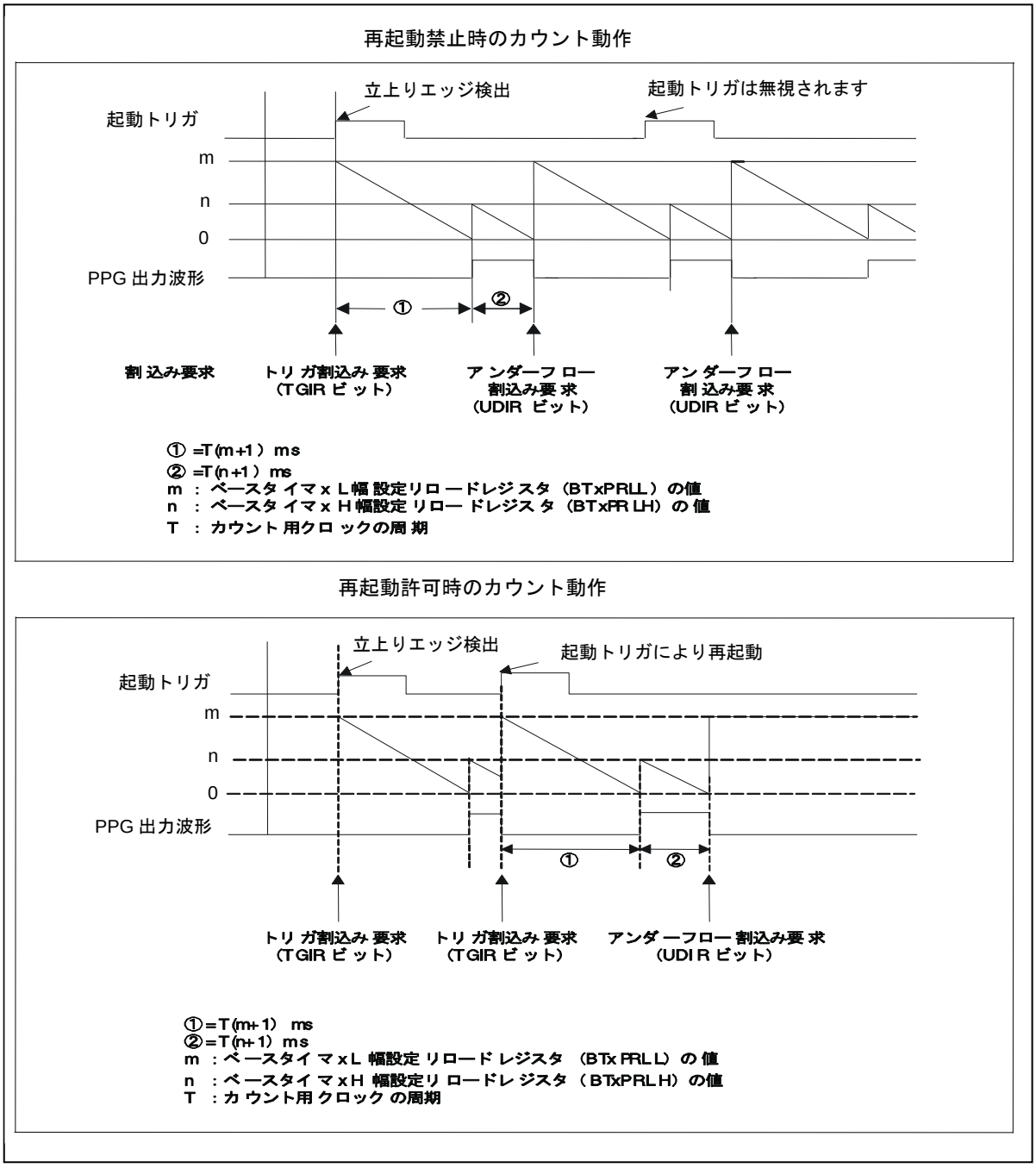
起動トリガが入力されてからのカウント動作を タイマ制御レジスタ(BTxTMCR)の OSEL ビットで通常極性を設定した(OSEL=0)場合を例にとって説明します。

1. L 幅設定リロードレジスタ(BTxPRLL)の値が 16 ビットダウンカウンタに、ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値がバッファに転送され、L 幅設定リロードレジスタ(BTxPRLL)の値のカウントダウンが開始される。このとき、出力信号(TOUT)は"L"レベルになります。
2. 16 ビットダウンカウンタが L 幅設定リロードレジスタ(BTxPRLL)の値をカウントダウンし終わる。
3. バッファに転送されていた H 幅設定リロードレジスタ(BTxPRLH)の値が 16 ビットダウンカウンタにリロードされ、カウント動作を続ける。このとき、出力信号(TOUT)は"H"レベルになります。
4. 16 ビットダウンカウンタが、H 幅設定リロードレジスタ(BTxPRLH)に設定した値をカウントダウンし終わり、アンダフローが発生する。
5. L 幅設定リロードレジスタ(BTxPRLL)の値が 16 ビットダウンカウンタにリロードされ、カウント動作を続ける。このとき、出力信号(TOUT)は"L"レベルになります。また、H 幅設定リロードレジスタ(BTxPRLH)の値がバッファに転送されます。
6. 手順 2～5 が繰り返され、カウント動作を継続します。

また、カウント中の再起動が許可/ 禁止されている場合の動作は、タイマ制御レジスタ(BTxTMCR)の RTGEN ビットで再起動を許可しているかどうかで異なります。

- 再起動禁止(RTGEN=0)の場合 ： カウント中に入力された起動トリガは無視されます。
- 再起動許可(RTGEN=1)の場合 ： ベースタイマ x ステータス制御レジスタ(BTxSTC)の TGIR ビットが"1" に変わります。また、L 幅設定リロードレジスタ(BTxPRLL)に設定した値が 16 ビットダウンカウンタにリロードされ、カウントが開始されます。

図 19-19 リロードモード時カウント動作例



＜注意事項＞

- 16 ビット PPG タイマの出力信号(TOUT)の出力方法/ 出力先は、次の設定によって異なります。
  - ベースタイマの入出力モード
  - TIOA0, TIOA1 端子機能
- 16 ビットダウンカウンタのカウントのタイミングとロードのタイミングが一致した場合は、ロード動作が優先されます。

■ 書込みタイミング

ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)/ ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値は、次のタイミングでリロードされます。

ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)の値

次のいずれかの場合に 16 ビットダウンカウンタへロードされます。

- 起動トリガ検出時
- ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値のカウント後、アンダフローが発生したとき

ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値

次のいずれかの場合にバッファに転送されます。

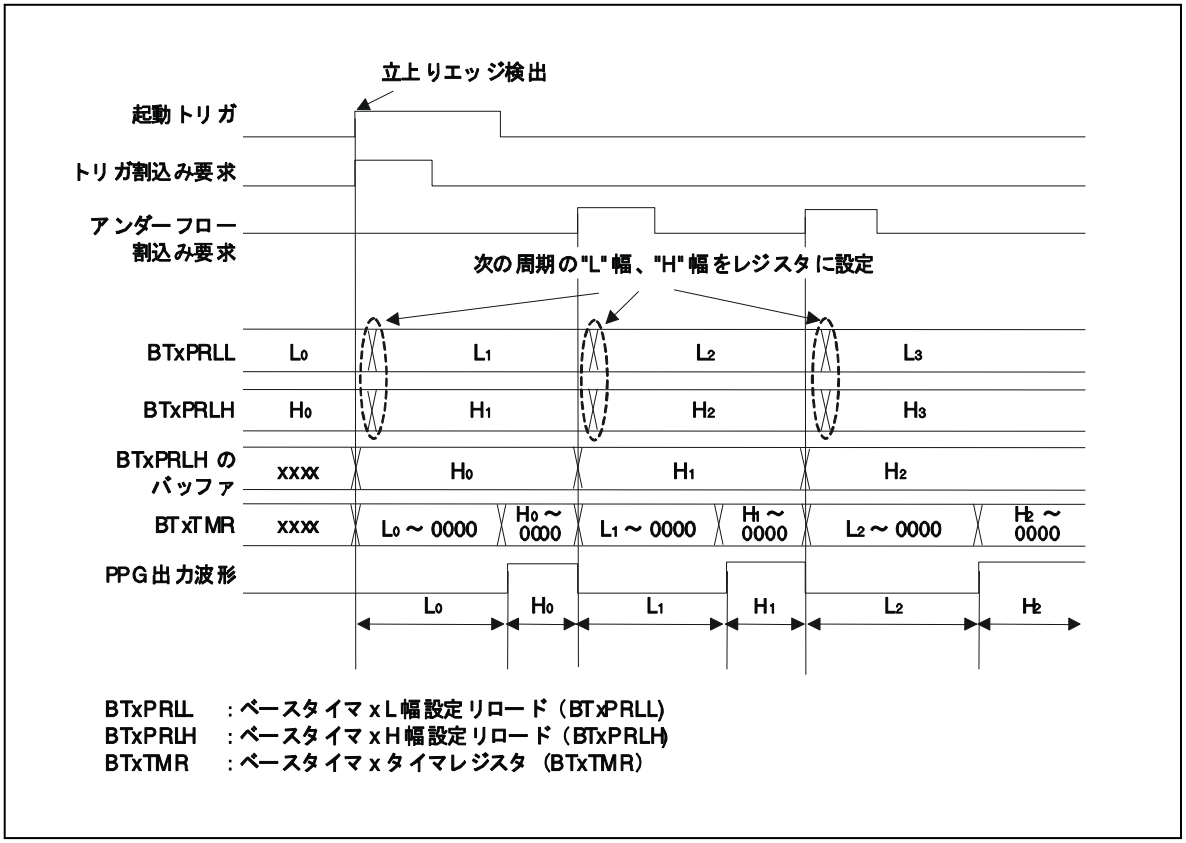
- 起動トリガ検出時
- ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値のカウント後、アンダフローが発生したとき

次の場合に、バッファから 16 ビットダウンカウンタにロードされます。

- ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)の値のカウントが終了したとき

そのため、ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)・ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)は、アンダフローが発生してから(ステータス制御レジスタ(BTxSTC)の UDIR ビットが"1"に変わってから)、次の周期のカウントが開始されるまでの間に書き換えてください。書き換えたデータは次の周期として反映されます。

図 19-20 書込みタイミング



■ 割込み発生タイミング

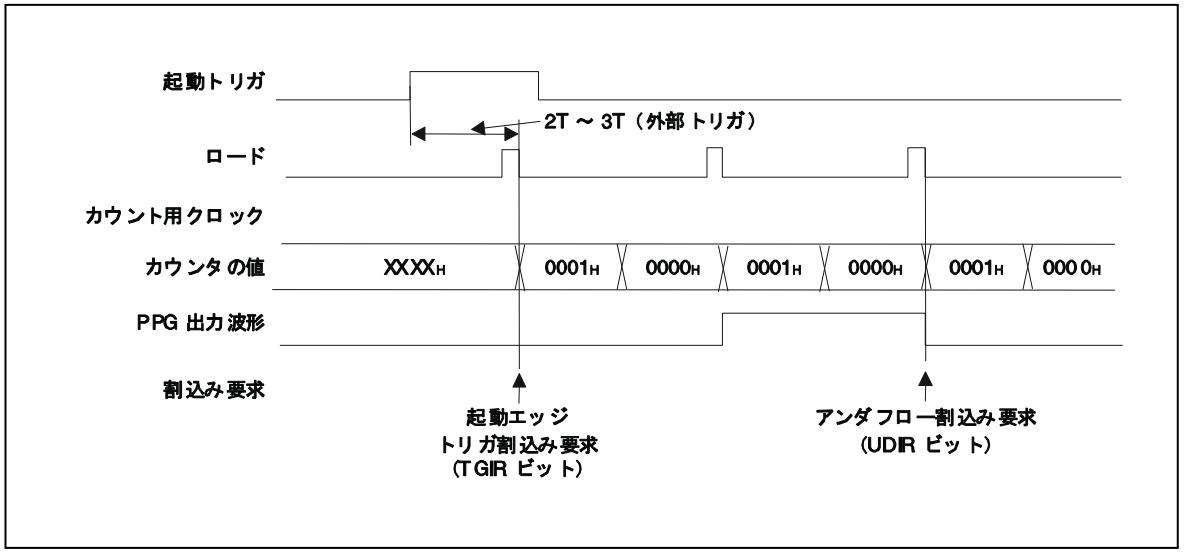
16 ビット PPG タイマは次の場合に割込み要求を発生できます。

- 起動トリガ検出時
- H 幅設定リロードレジスタ(BTxPRLH)の値で、アンダフローが発生したとき

割込み要求発生のタイミングを次の設定がされている時を例にとって、次図に示します。

- L 幅設定リロードレジスタ(BTxPRLL)の値=0001_H
- H 幅設定リロードレジスタ(BTxPRLH)の値=0001_H

図 19-21 割込み要求発生のタイミングチャート



19.5.6.4 ワンショットモード時の動作

ワンショットモード時の動作について説明します。

ワンショットモード時の動作について説明します。

■ カウント動作

● 起動

リロードモード時と同様です。「19.5.6.3. リロードモード時の動作」の「■ 動作」を参照してください。

● カウント動作

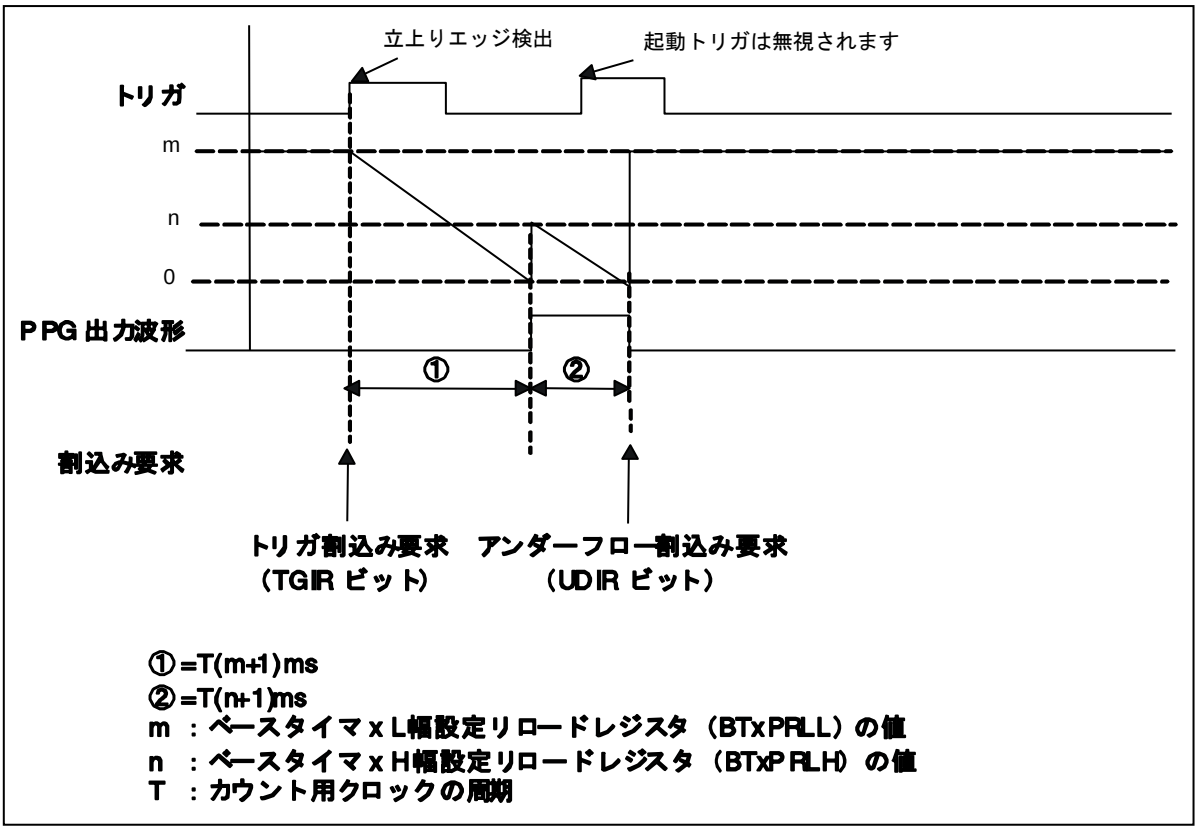
起動トリガが入力されてからのカウント動作をタイマ制御レジスタ(BTxTMCR)の OSEL ビットで通常極性を設定した(OSEL=0)場合を例にとって説明します。

1. ベースタイマ xL 幅設定リロードレジスタ(BTxPRLL)の値が 16 ビットダウンカウンタに、 ベースタイマ xH 幅設定リロードレジスタ(BTxPRLH)の値がバッファに転送され、L 幅設定リロードレジスタ(BTxPRLL)の値のカウントダウンが開始される。このとき、出力信号(TOUT)は"L"レベルになります。
2. 16 ビットダウンカウンタが L 幅設定リロードレジスタ(BTxPRLL)の値をカウントダウンし終わる
3. バッファに転送されていた H 幅設定リロードレジスタ(BTxPRLH)の値が 16 ビットダウンカウンタにリロードされ、カウント動作を続ける。このとき、出力信号(TOUT)は"H"レベルになります。
4. 16 ビットダウンカウンタが、H 幅設定リロードレジスタ(BTxPRLH)に設定した値をカウントダウンし終わり、アンダフローが発生する。
5. カウント動作が停止する。

また、カウント中の再起動が許可/ 禁止されている場合の動作は、タイマ制御レジスタ(BTxTMCR)の RTGEN ビットで再起動を許可しているかどうかで異なります。

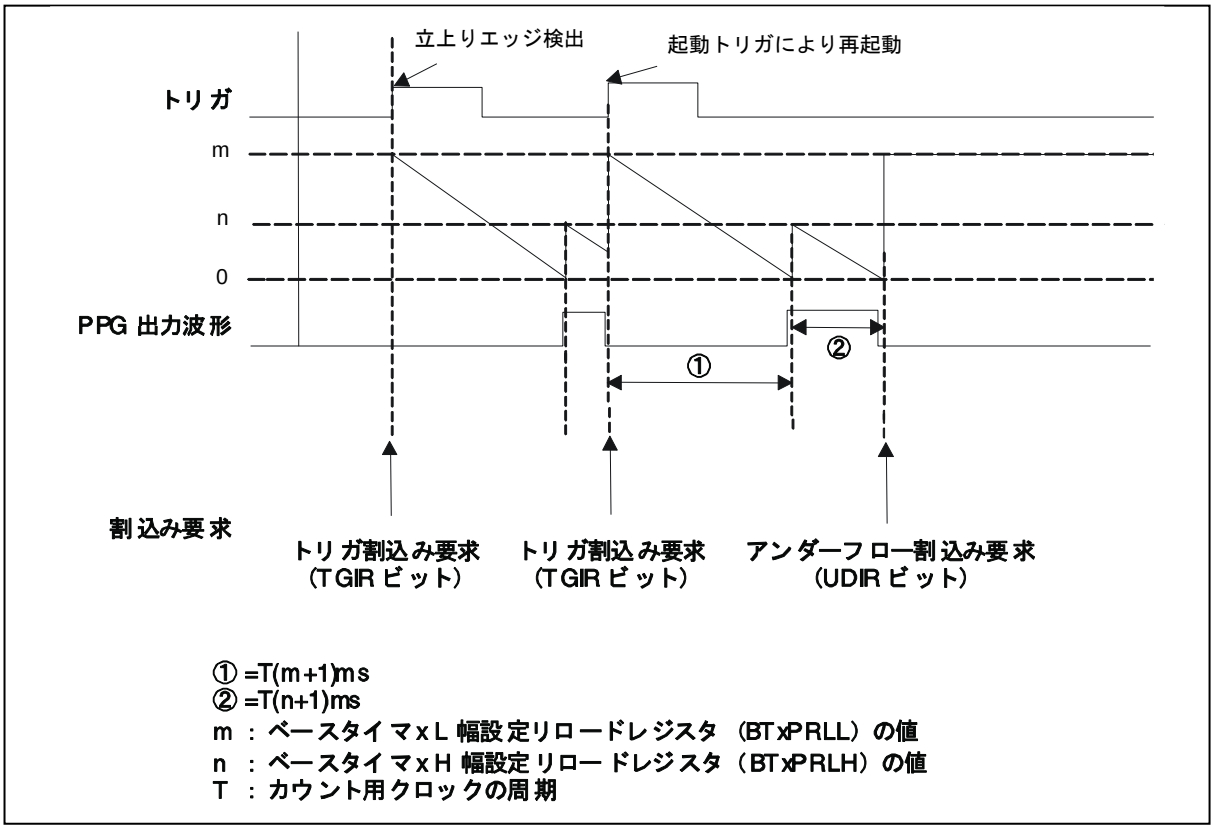
- 再起動禁止(RTGEN=0)の場合 ： カウント中に入力された起動トリガは無視されます。
- 再起動許可(RTGEN=1)の場合 ： ステータス制御レジスタ(BTxSTC)の TGIR ビットが"1"に変わります。また、L 幅設定リロードレジスタ(BTxPRLL)に設定した値が 16 ビットダウンカウンタにリロードされ、カウントが開始されます。

図 19-22 再起動禁止時のカウント動作例



①=T(m+1)ms
②=T(n+1)ms
m ：ベースタイマ x L幅設定リロードレジスタ （BTxPRLL）の値
n ：ベースタイマ x H幅設定リロードレジスタ （BTxPRLH）の値
T ：カウント用クロックの周期

図 19-23 再起動許可時のカウント動作例



① =T(m+1)ms
② =T(n+1)ms
m ：ベースタイマ x L 幅設定リロードレジスタ （BTxPRLL）の値
n ：ベースタイマ x H 幅設定リロードレジスタ （BTxPRLH）の値
T ：カウント用クロックの周期

＜注意事項＞
16 ビット PPG タイマの出力信号(TOUT)の出力方法/ 出力先は、次の設定によって異なります。

- ベースタイマの入出力モード
- TIOA0, TIOA1 端子機能
- カウント終了時に、16 ビット PPG タイマの起動トリガが検出されると、L 幅設定リロードレジスタ(BTxPRLL)の値(周期)が 16 ビットダウンカウンタにロードされ、カウントが開始されます。

■ 割込み発生タイミング

リロードモード時と同様です。「19.5.6.3. リロードモード時の動作」の「■ 割込み発生タイミング」を参照してください。

19.5.6.5 割込み

16 ビット PPG タイマの割込みについて説明します。

次のいずれかの場合に割込み要求が発生します。

- 起動トリガ検出時(トリガ割込み要求)
- H 幅設定リロードレジスタ(BTxPRLH)の値でアンダフローが発生したとき(アンダフロー割込み要求)

表 19-7 割込み発生条件

割込み要求	割込み要求フラグ	割込み要求許可	割込み要求のクリア
トリガ割込み要求	BTxSTC の TGIR=1	BTxSTC の TGIE=1	BTxSTC の TGIR ビットに"0"を書き込む。
アンダフロー割込み要求	BTxSTC の UDIR=1	BTxSTC の UDIE=1	BTxSTC の UDIR ビットに"0"を書き込む。

＜注意事項＞
- 割込み要求フラグが"1"のときに割込み要求の発生を許可すると割込みを許可した時点で、割込み要求が発生します。
- 割込み要求の発生を許可する場合は、次のいずれかの処理を行ってください。
  - ☐ 割込み要求の発生を許可する前に割込み要求をクリアする。
  - ☐ 割込み許可と同時に割込み要求をクリアする。
- 割込み要求は割込み要求の発生を禁止してからクリアするか、割込み処理ルーチン内でクリアしてください。
- 割込みベクタ番号に対応する割込みレベルは、割込みコントロールレジスタ(ICR00～ICR47)で設定します。割込みレベルの設定については、『割込み制御(割込みコントローラ)』の章を参照してください。

19.5.6.6 使用上の注意

16 ビット PPG タイマの使用上の注意について説明します。

16 ビット PPG タイマを使用する際は、次の点に注意してください。

■ プログラムで設定する場合の注意

- タイマ制御レジスタ(BTxTMCR)の次のビットは、CTEN ビットで 16 ビットダウンカウンタの動作を停止(CTEN=0)してから、書き換えてください。
  - ☐ CKS2～CKS0 ビット
  - ☐ EGS1, EGS0 ビット
  - ☐ FMD2～FMD0 ビット
  - ☐ MDSE ビット
- タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットでリセットモードを設定(FMD2～FMD0=000)すると、すべてのレジスタが初期化されます。
- ベースタイマの機能を変更する場合は、一度ベースタイマをリセットする必要があります。リセット後に、タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットを書き換える場合以外は、必ず FMD2～FMD0 ビットでリセットモードを選択(FMD2～FMD0=000)してから、再度 FMD2～FMD0 ビットでベースタイマの機能を選択してください。
- 16 ビット PPG タイマは次の手順で設定してください。
  1. タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットでベースタイマの機能に 16 ビット PPG タイマを設定(FMD2～FMD0=010)
  2. L 幅設定リロードレジスタ(BTxPRLL)を設定
  3. H 幅設定リロードレジスタ(BTxPRLH)を設定

■ 動作に関する注意

- 16 ビットダウンカウンタのカウントのタイミングとロードのタイミングが重なった場合は、ロード動作が優先されます。
- ワンショットモードでカウント終了時に、16 ビット PPG タイマの再起動トリガが検出されると、 L 幅設定リロードレジスタ(BTxPRLL)の値(周期)が 16 ビットダウンカウンタにロードされ、カウントが開始されます。
- ベースタイマ入出力選択機能によって、信号(外部クロック/外部起動トリガ/ 波形)の入出力動作が異なります。

■ 割込みに関する注意

- 割込み要求フラグのクリアと、割込み要求フラグが"1"に変わるタイミングが重なった場合は、 割込み要求フラグのクリア動作は無視され、割込み要求フラグは"1"のままになります。

19.5.7　16/32 ビット PWC タイマの動作

16/32 ビット PWC タイマの動作について示します。

本シリーズに内蔵されているベースタイマを 16/32 ビット PWC タイマとして使用する場合の動作について説明します。また、各動作状態を設定するための手順例も示します。

図 19-24 ブロックダイヤグラム (16 ビット PWC 動作時)

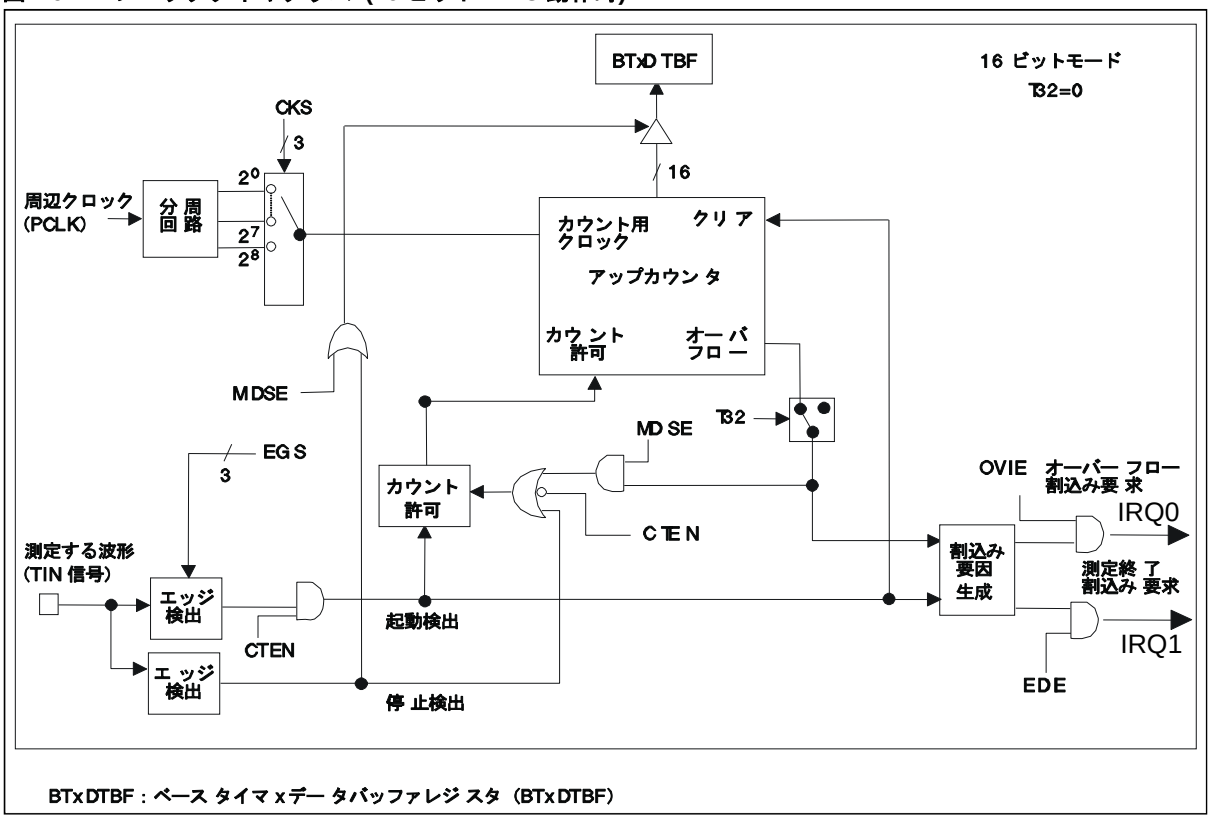
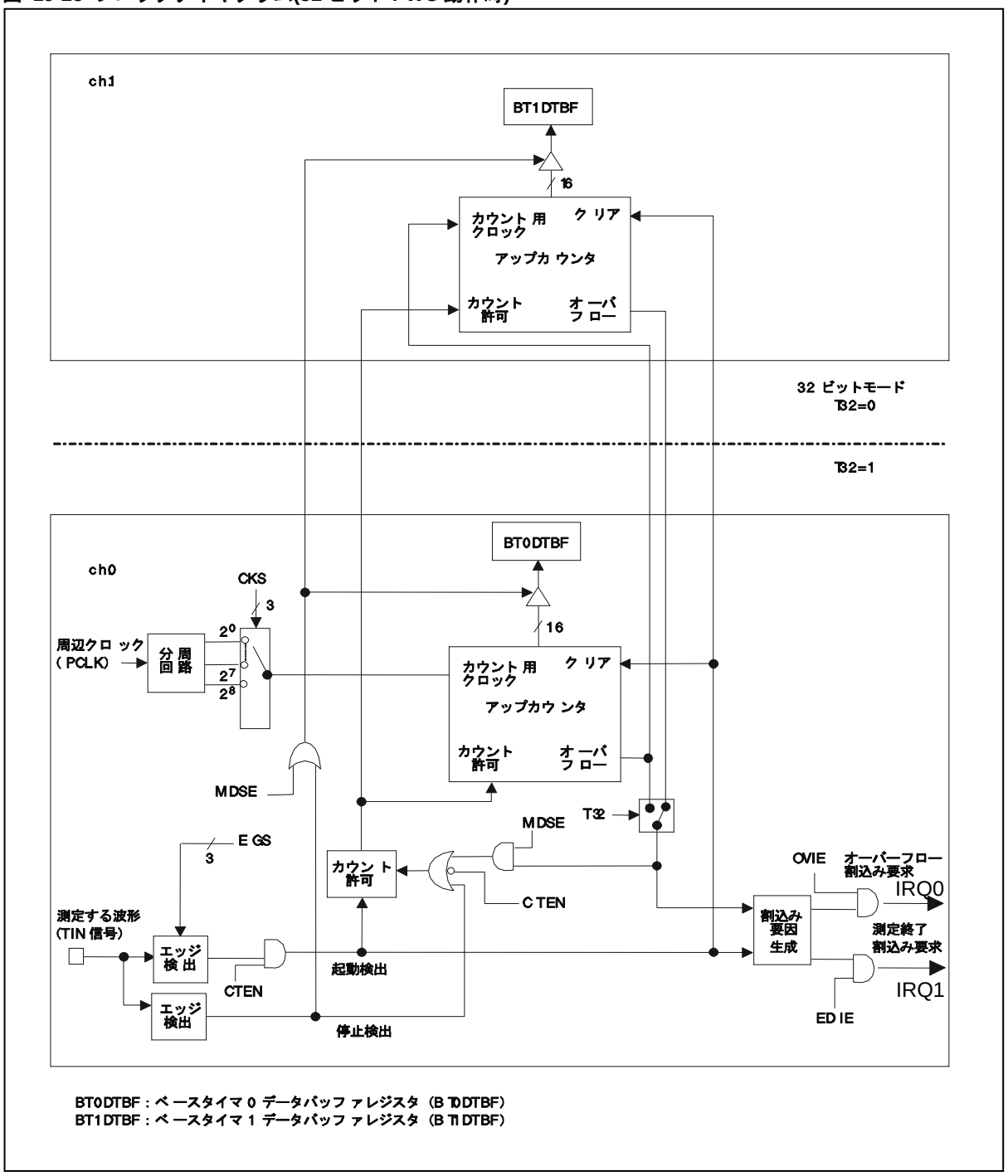


図 19-25 ブロックダイヤグラム(32 ビット PWC 動作時)



19.5.7.1 概要

16/32 ビット PWC タイマの概要について説明します。

16/32 ビット PWC タイマは、入力される信号のパルス幅や周期を測定するタイマです。入力信号(TIN)で測定開始エッジを検出するとカウントアップを開始し、測定終了エッジを検出するとカウント動作を停止します。カウントされた値 (測定結果)がパルス幅や周期としてデータバッファレジスタ (BTxDTBF)に格納されます。

16/32 ビット PWC タイマには、タイマモード, 動作モード, 測定モードの 3 種類のモードがあり、 それぞれのモードの組み合わせによって、動作が異なります。

＜注意事項＞
TIN 信号の入力方法は、入出力選択レジスタ(BTSEL01)で設定した入出力モードによって異なります。「19.5.2 入出力割当て」を参照してください。

■ タイマモード

タイマ制御レジスタ(BTxTMCR)の T32 ビットで次の 2 種類のモードから選択します。

- 16 ビットタイマモード(T32=0) : 16 ビット PWC タイマを 1 チャネルずつ個別に動作させます。
- 32 ビットタイマモード(T32=1) : 2 チャネルをカスケード接続して 32 ビット PWC タイマとして使用します。

32 ビットタイマモード時の動作については、「19.5.7.3 32 ビットタイマモード時の動作」を参照してください。

＜注意事項＞
32 ビットタイマモードを設定する場合は、偶数チャネルと奇数チャネルの T32 ビットの設定が異なります。詳しくは、「19.5.7.3 32 ビットタイマモード時の動作」を参照してください。

■ 動作モード

タイマ制御レジスタ(BTxTMCR)の MDSE ビットで次の 2 種類のモードから選択します。

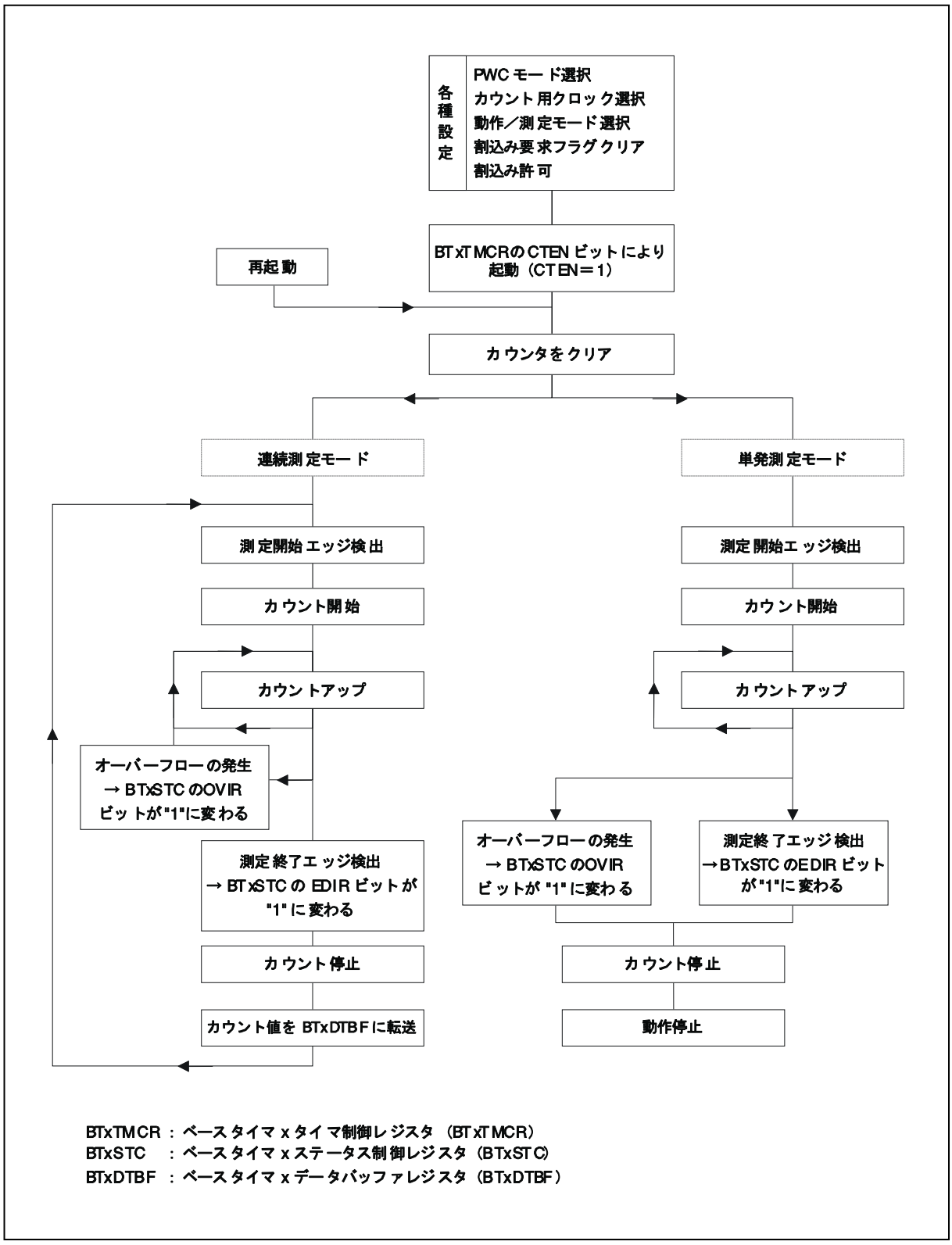
- 連続測定モード(MDSE=0) : 1 回測定が終わると、測定開始エッジが入力されるまで待機し、再度測定開始エッジが入力されると測定を行うモードです。
- 単発測定モード(MDSE=1) : 測定を 1 回のみ行うモードです。単発測定モードと連続測定モードの相違点を次表 に示します。

表 19-8 単発測定モードと連続測定モードの相違点

	単発測定モード	連続測定モード
測定動作	測定終了エッジを検出すると測定動作を停止。	測定終了エッジを検出すると、測定動作を停止し、測定開始エッジが検出されるまで待機。 測定開始エッジが検出されると、再度測定を開始。
BTxDTBF の機能	測定動作中 : 測定中の値を保持 測定終了後 : 測定結果を保持	測定動作中 : 前回の測定結果を保持 測定終了後 : 測定結果を保持
オーバフロー時の動作	測定を停止。	0x0000 から再度測定を開始

動作フローを図 19-26 に示します。

図 19-26 動作フロー

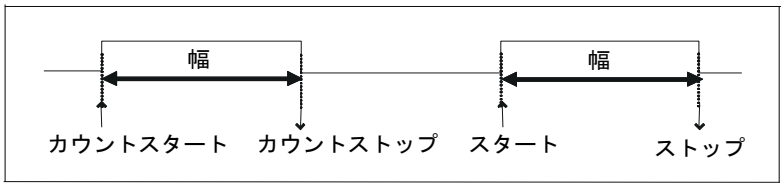
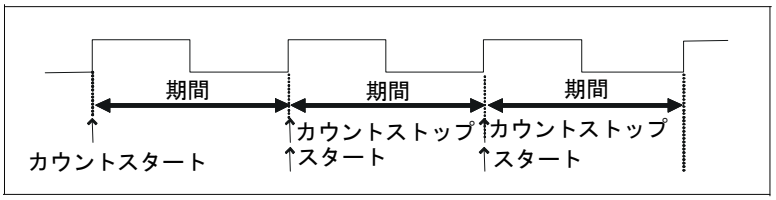
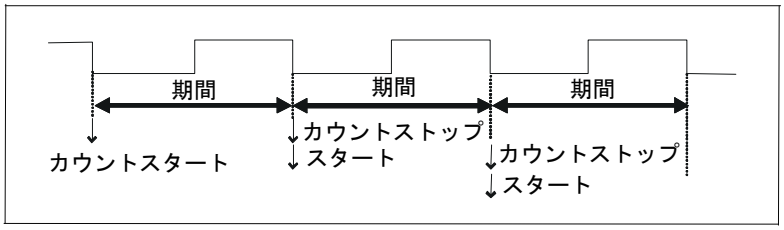

各種設定
PWC モード選択
カウント用クロック選択
動作／測定モード選択
割込み要求フラグクリア
割込み許可
再起動
BTxTMCRのCTENビットにより起動（CTEN＝1）
カウンタをクリア
連続測定モード
単発測定モード
測定開始エッジ検出
カウント開始
カウントアップ
オーバーフローの発生→BTxSTCのOVIRビットが"1"に変わる
測定終了エッジ検出→BTxSTCのEDIRビットが"1"に変わる
カウント停止
カウント値をBTxDTBFに転送
測定開始エッジ検出
カウント開始
カウントアップ
オーバーフローの発生→BTxSTCのOVIRビットが"1"に変わる
測定終了エッジ検出→BTxSTCのEDIRビットが"1"に変わる
カウント停止
動作停止
BTxTMCR ： ベースタイマxタイマ制御レジスタ（BTxTMCR）
BTxSTC ： ベースタイマxステータス制御レジスタ（BTxSTC）
BTxDTBF ： ベースタイマxデータバッファレジスタ（BTxDTBF）

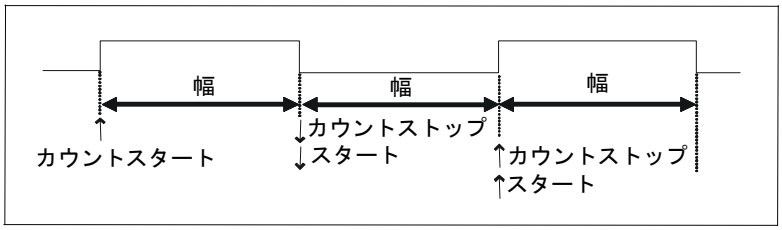
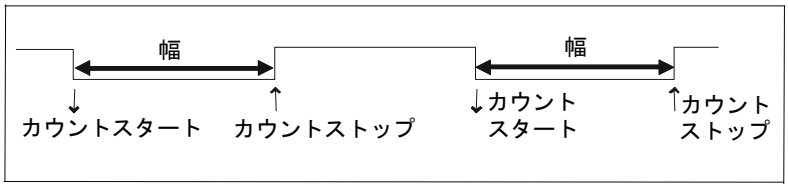
＜注意事項＞
連続測定モードの場合、データバッファレジスタ(BTxDTBF)から測定結果を読み出す前に次の測定が終了すると、データバッファレジスタ(BTxDTBF)に保持されている値が新しい値に上書きされ、古い値は破棄されます。このとき、ステータス制御レジスタ(BTxSTC)の ERR ビットが"1"に変わります。ベースタイマ x データバッファレジスタ(BTxDTBF)を読み出すと ERR ビットを"0"にクリアできます。

■ 測定モード

タイマ制御レジスタ (BTxTMCR) の EGS2～EGS0 ビットで次の 5 種類から選択します。

図 19-27 測定モードと測定内容

測定モード (EGS2～EGS0)	測定内容	参照先
H パルス幅測定 (EGS[2:0]=000)	"H"レベルの信号が入力されている期間の幅を測定します。  カウント(測定)開始: 立上りエッジ検出時 カウント(測定)終了: 立下りエッジ検出時	
立上りエッジ間周期測定 (EGS[2:0]=001)	立上りエッジを検出してから、次の立上りエッジを検出するまでの周期を測定します。  カウント(測定)開始: 立上りエッジ検出時 カウント(測定)終了: 立上りエッジ検出時	
立下りエッジ間周期測定 (EGS[2:0]=010)	立下りエッジを検出してから、次の立下りエッジを検出するまでの周期を測定します。  カウント(測定)開始: 立下りエッジ検出時 カウント(測定)終了: 立下りエッジ検出時	

測定モード (EGS2～EGS0)	測定内容	参照先
全エッジ間パルス幅測定 (EGS[2:0]=011)	連続して入力されるエッジ間の幅を測定します。 ・立上りエッジ検出から、立下りエッジ検出まで ・立下りエッジ検出から、立上りエッジ検出まで  カウント(測定)開始: エッジ検出時 カウント(測定)終了: エッジ検出時	
L パルス幅周期測定 (EGS[2:0]=100)	"L"レベルの信号が入力されている期間の幅を測定します。  カウント(測定)開始: 立下りエッジ検出時 カウント(測定)終了: 立上りエッジ検出時	

19.5.7.2 PWC 測定時の動作

PWC 測定時の動作について説明します。

測定時の動作について説明します。以下文中のセンシティブエッジ①およびセンシティブエッジ②の内容は、図 19-27 測定モードと測定内容を参照してください。

■ 起動

次の手順で 16/32 ビット PWC タイマを起動してください。

タイマ制御レジスタ(BTxTMCR)の CTEN ビットで、16/32 ビット PWC タイマの動作を許可(CTEN=1)する。カウンタの値が"0000_H" にクリアされ、16/32 ビット PWC タイマが測定開始エッジ入力待機状態になります。(測定開始エッジが入力されるまではカウントは行われません。)

■ カウント動作

● 単発測定モード時の動作

測定開始エッジ待機中に、入力信号(TIN)でセンシティブエッジ①を検出すると、アップカウンタがカウント用クロックに同期して"0001_H"からカウントアップを開始します。入力信号(TIN)でセンシティブエッジ②が検出されると、アップカウンタの動作が停止します。また、アップカウンタの値がデータバッファレジスタ(BTxDTBF)に格納されます。測定が終了したときや、 オーバフローが発生したときに割込み要求を発生させることができます。

＜注意事項＞

- 単発測定モードでは、オーバフローが発生するとカウント動作を停止します。
- 測定する波形(TIN 信号)の入力方法は、入出力選択レジスタ(BTSEL01)で設定した入出力モードによって異なります。

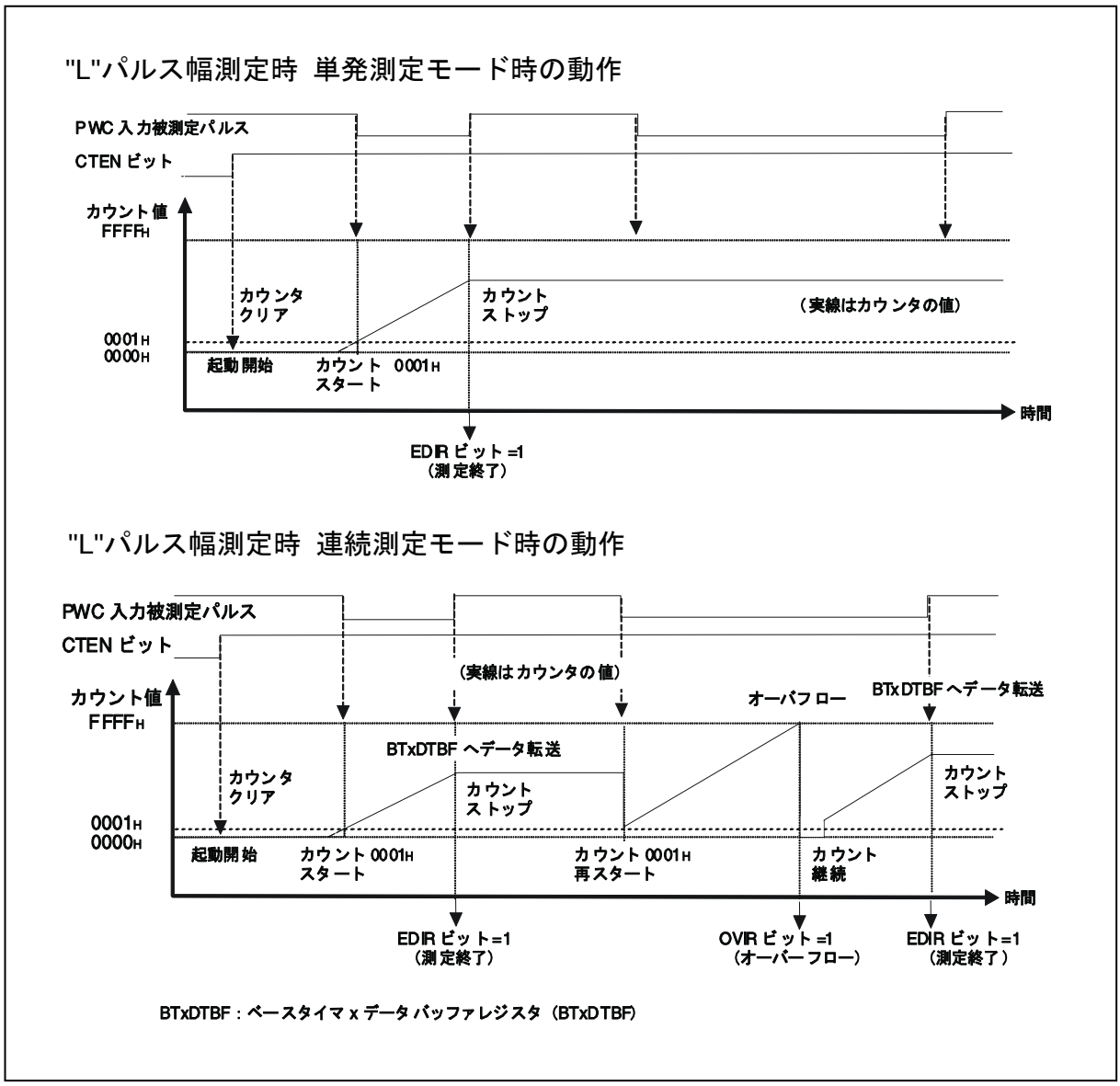
● 連続測定モード時の動作

測定開始エッジ待機中に、入力信号(TIN)でセンシティブエッジ①を検出すると、アップカウンタがカウント用クロックに同期して"0001_H"からカウントアップを開始します。入力信号(TIN)でセンシティブエッジ②が検出されると、アップカウンタの動作が停止し、測定開始エッジ入力待機状態になります。また、アップカウンタの値がデータバッファレジスタ (BTxDTBF)に格納されます。測定開始エッジ待機中に、入力信号(TIN)で立上りエッジを検出すると、アップカウンタが再度"0001_H"からカウントアップを開始します。測定が終了したときや、オーバフローが発生したときに割込み要求を発生させることができます。

＜注意事項＞

測定する波形(TIN 信号)の入力方法は、入出力選択レジスタ(BTSEL01)で設定した入出力モードによって異なります。

図 19-28 動作例



■ 再起動

カウント動作中にベースタイマ x タイマ制御レジスタ(BTxTMCR)の CTEN ビットに"1"が書き込まれると、アップカウンタに再起動がかかり次のように動作します。

● 測定開始エッジ待機中に再起動がかけられた場合

そのまま測定開始エッジ待機状態を継続します。

● 測定中に再起動がかけられた場合

アップカウンタの値が"0000_H"にクリアされ、測定開始エッジ待機状態になります。

＜注意事項＞

測定終了エッジの検出と再起動が同時に発生すると、次のようなことが起こる場合があります。割込み要求フラグの動作に注意して割込み制御を行なってください。

- 単発測定モード時:再起動を行い測定開始エッジ待機状態となりますが、ステータス制御レジスタ(BTxSTC)の EDIR ビット(測定終了割込み要求フラグ)が"1"に変わります。
- 連続測定モード時:再起動を行い測定開始エッジ待機状態となりますが、ステータス制御レジスタ(BTxSTC)の EDIR ビット(測定終了割込み要求フラグ)が"1"に変わります。また、その時点での測定結果がデータバッファレジスタ(BTxDTBF)に転送されます。
- 連続測定モードで 16/32 ビット PWC タイマを再起動したと同時に、入力信号(TIN)で測定開始エッジを検出した場合は直ちに"0001_H"からカウントを開始します。

■ パルス幅算出方法

測定終了後、ベースタイマ x データバッファレジスタ(BTxDTBF)に格納された測定結果から測定したパルス幅を次の計算式で算出できます。

パルス幅 = n × T

n : データバッファレジスタ(BTxDTBF)の値

T : カウント用クロックの周期

19.5.7.3 32 ビットタイマモード時の動作

32 ビットタイマモード時の動作について説明します。

16 ビット PWC タイマ 2 チャネルをカスケード接続して、32 ビットの PWC タイマとして使用する場合の設定と動作について説明します。

■ 概要

タイマ制御レジスタ (BTxTMCR)の T32 ビットで 16 ビット PWC タイマ 2 チャネルをカスケード接続して、32 ビットの PWC タイマとして使用できます。

このモードでは、偶数チャネルが下位 16 ビットの動作に、奇数チャネルが上位 16 ビットの動作に対応します。そのためアップカウンタの値の読出しは、下位 16 ビット(偶数チャネル) → 上位 16 ビット(奇数チャネル)の順に行ってください。

■ 設定手順例

32 ビットタイマモードを設定する場合は、偶数チャネルのベースタイマ x タイマ制御レジスタ(BTxTMCR)の T32 ビットを"1"に、奇数チャネルの T32 ビットを"0"に設定してください。また、32 ビットタイマモード設定時は次の手順で各レジスタを設定してください。

偶数チャネルと奇数チャネルでレジスタの設定が異なります。ch.0 と ch.1 をカスケード接続する場合を例に説明します。

1. ベースタイマ 0 タイマ制御レジスタ(BT0TMCR)の FMD2～FMD0 ビットで ch.0 をリセットモードに設定(FMD2～FMD0=000)
2. ch.0 と ch.1 のベースタイマ x タイマ制御レジスタ (BT0TMCR, BT1TMCR)の FMD2～FMD0 ビットで、ch.0 と ch.1 を 16/32 ビット PWC タイマを設定(FMD2～FMD0=100)。同時にベースタイマ 0 タイマ制御レジスタ (BT0TMCR)の T32 ビットで 32 ビットタイマモードを設定。(T32=1)

＜注意事項＞

T32 ビットは、偶数チャネル/奇数チャネル両方の動作が停止している状態で書き換えてください。カウント動作が停止しているかどうかは、タイマ制御レジスタ(BTxTMCR)の CTEN ビットで確認できます(CTEN=0)。

■ 動作

32 ビットタイマモード設定時も、カウント動作は基本的に 16 ビットタイマモード時と同様です。ただし、カウント動作は偶数チャネルの設定に従うため、奇数チャネルの次のレジスタの設定は無視されます。

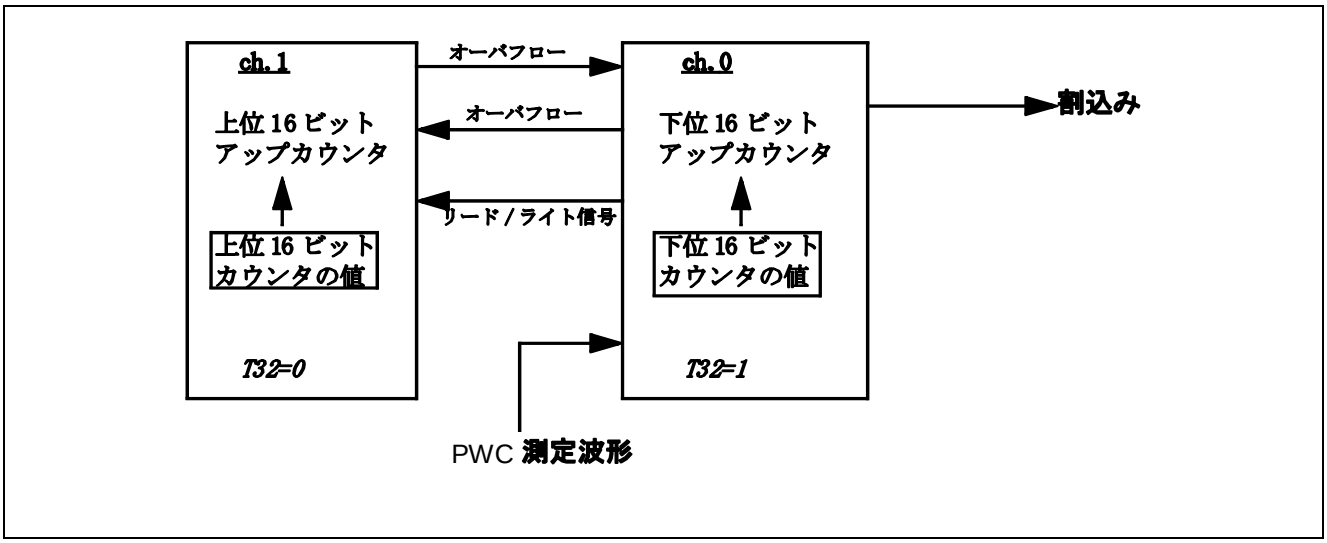
- ベースタイマ x タイマ制御レジスタ (BTxTMCR)
- ベースタイマ x ステータス制御レジスタ (BTxSTC)

32 ビットタイマモード時のカウント動作について説明します。

1. 偶数チャネルのタイマ制御レジスタ (BTxTMCR)の CTEN ビットで、16/32 ビット PWC タイマの動作を許可 (CTEN=1)すると、32 ビット PWC タイマが起動します。
2. 入力信号 (TIN)で測定開始エッジを検出するとカウント動作が開始されます。
3. 偶数チャネルが下位 16 ビット、奇数チャネルが上位 16 ビットの 32 ビットカウンタとしてアップカウンタがカウントを開始します。
4. 入力信号 (TIN)で測定終了エッジを検出するとアップカウンタの値の下位 16 ビットが偶数チャネルの上位 16 ビットが奇数チャネルのデータバッファレジスタ(BTxDTBF)に格納されます。

32 ビットタイマモード時のチャネル構成を次に示します。

図 19-29 32 ビットタイマモード時の構成



＜注意事項＞

- ダウンカウンタの値はデータバッファレジスタ(BTxDTBF)を読み出すことで確認できます。32 ビットタイマモード時は必ず下位 16 ビット (偶数チャネル) → 上位 16 ビット (奇数チャネル)の順で読み出してください。
- 32 ビットタイマモード時は、32 ビット PWC タイマの動作は偶数チャネルの設定に従います。そのため、割込み要求は偶数チャネルのものが有効になります。

19.5.7.4 割込み

ベースタイマの割込みについて説明します。

次のいずれかの場合に割込み要求が発生します。

- オーバフロー発生時 (オーバフロー割込み要求)
- 測定が終了したとき (測定終了割込み要求)

表 19-9 割込み発生条件

割込み要求	割込み要求フラグ	割込み要求許可	割込み要求のクリア
オーバフロー割込み要求	BTxSTC の OVIR=1	BTxSTC の OVIE=1	BTxSTC の OVIR ビットに"0"を書き込む。
測定終了割込み要求	BTxSTC の EDIR=1	BTxSTC の EDIE=1	BTxDTBF を読み出す

＜注意事項＞

- 割込み要求フラグが"1"のときに割込み要求の発生を許可すると割込みを許可した時点で、割込み要求が発生します。
- 割込み要求の発生を許可する場合は、次のいずれかの処理を行ってください。
  - 割込み要求の発生を許可する前に割込み要求をクリアする。
  - 割込み許可と同時に割込み要求をクリアする。
- 割込み要求は割込み要求の発生を禁止してからクリアするか、割込み処理ルーチン内でクリアしてください。
- 各割込み要求の割込みベクタ番号については、『付録 C. 割込みベクター覧』を参照してください。
- 割込みベクタ番号に対応する割込みレベルは、割込みコントロールレジスタ(ICR00 ～ICR47)で設定します。割込みレベルの設定については、『割込み制御(割込みコントローラ)』の章を参照してください。

19.5.7.5 使用上の注意

ベースタイマの使用上の注意について説明します。

16/32 ビット PWC タイマを使用する際は、次の点に注意してください。

■ プログラムで設定する場合の注意

- ベースタイマ x タイマ制御レジスタ(BTxTMCR)の以下のビットは、CTEN ビットでアップカウンタの動作を停止(CTEN=0)してから、書き換えてください。
  - CKS2～CKS0 ビット
  - EGS2～EGS0 ビット
  - T32 ビット
  - FMD2～FMD0 ビット
  - MDSE ビット
  - タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットでリセットモードを設定(FMD2～FMD0=000)すると、すべてのレジスタが初期化されます。
  - ベースタイマの機能を変更する場合や T32 ビットを変更する場合は、 一度ベースタイマをリセットする必要があります。リセット後に、タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットや T32 ビットを書き換える場合以外は、必ず FMD2～FMD0 ビットでリセットモードを選択(FMD2 ～FMD0=000)してから、これらのビットを書き換えてください。
- システムリセット/ リセットモード時に、次の設定を同時に行うと、その直前の測定信号の状態によって動作する場合があります。
  - ベースタイマ x タイマ制御レジスタ(BTxTMCR)の FMD2～FMD0 ビットで、 ベースタイマの機能を 16/32 ビット PWC タイマに設定(FMD2～FMD0=100)
  - ベースタイマ x タイマ制御レジスタ(BTxTMCR)の CTEN ビットで 16/32 ビット PWC タイマの動作を許可(CTEN=1)

■ 動作に関する注意

- アップカウンタのカウントのタイミングとロードのタイミングが重なった場合は、ロード動作が優先されます。
- ベースタイマ x タイマ制御レジスタ(BTxTMCR)の CTEN ビットで 16/32 ビット PWC タイマの動作を許可(CTEN=1)すると、アップカウンタの値がクリアされ、起動許可前のアップカウンタの値は無効になります。
- 連続測定モードで 16/32 ビット PWC タイマを再起動したと同時に入力信号(TIN)で測定開始エッジを検出した場合は直ちに"0001_H"からカウントを開始します。
- 32 ビット PWC タイマとして使用する場合は、偶数チャネルの 16 ビット PWC タイマへの設定が有効になり、奇数チャネルの設定は無視されます。
- ベースタイマ入出力選択機能によって、測定する波形の入力動作が異なります。

■ 割込みに関する注意

- 割込み要求フラグのクリアと、割込み要求フラグが"1"に変わるタイミングが重なった場合は、 割込み要求フラグのクリア動作は無視され、割込み要求フラグは"1" のままになります。
- 測定終了エッジの検出と同時に 16/32 ビット PWC タイマを再起動すると次のようなことが起こる場合があります。割込み要求フラグの動作に注意して割込み制御を行なってください。
  - パルス幅単発測定モード時：再起動を行い測定開始エッジ待機状態となりますが、測定終了割込み要求フラグ(EDIR) は"1"に変わります。
  - パルス幅連続測定モード時：再起動を行い測定開始エッジ待機状態となりますが、測定終了割込み要求フラグ(EDIR) が"1"に変わり、その時点での測定結果がデータバッファレジスタ(BTxDTBF)に転送されます。

20. リロードタイマ

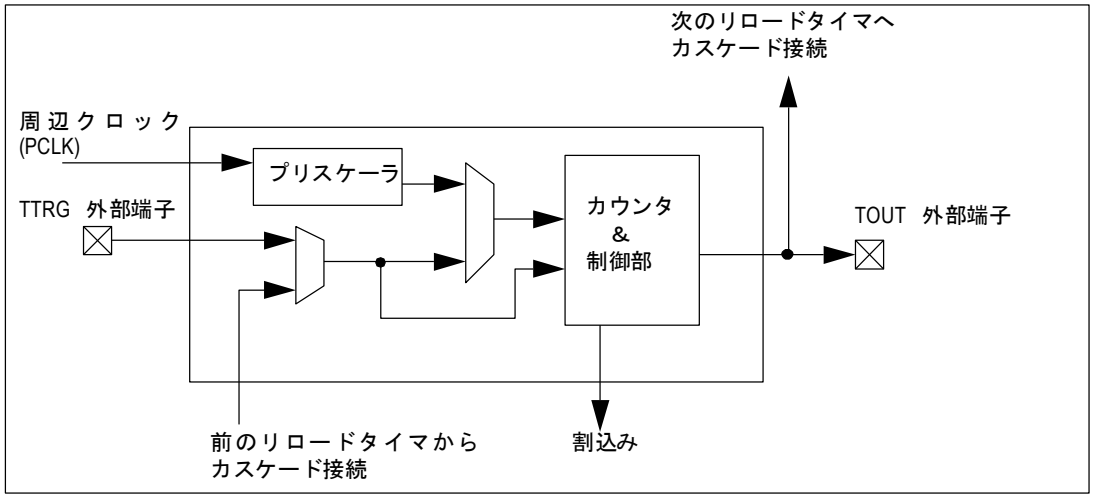
リロードタイマについて説明します。

20.1 概要

リロードタイマの概要について説明します。

本モジュールは、内部クロックをカウントするインターバルタイマモードと外部イベントをカウントするイベントカウンタモードを持つ 16 ビットのリロードダウンカウントタイマです。

図 20-1 リロードタイマのブロックダイヤグラム (1 チャネル分, 概要)



＜注意事項＞
チャネルごとに動作クロックが異なります。

ch.0-3: PCLK2

ch.4-6: PCLK1

20.2 特長

リロードタイマの特長について説明します。

本シリーズは 7 チャネルのリロードタイマを搭載しています。各チャネルは以下により構成されています。

- 16 ビットダウンカウンタ ×1
- 16 ビットリロードレジスタ ×1
- 16 ビットリロード / コンペア / キャプチャレジスタ ×1
- 上記 バッファ ×1
- 内部カウントクロック作成用 6 ビットプリスケーラ ×1
- 外部トリガ/イベント入力(TTRG) ×1
- 外部トグル出力(TOUT) ×1
- コントロールレジスタ ×1
- カウント比較器 ×1

本タイマは以下のインターバルタイマモード/イベントカウンタモードをもち、レジスタを設定することにより、以下の用途・機能で使用することができます。

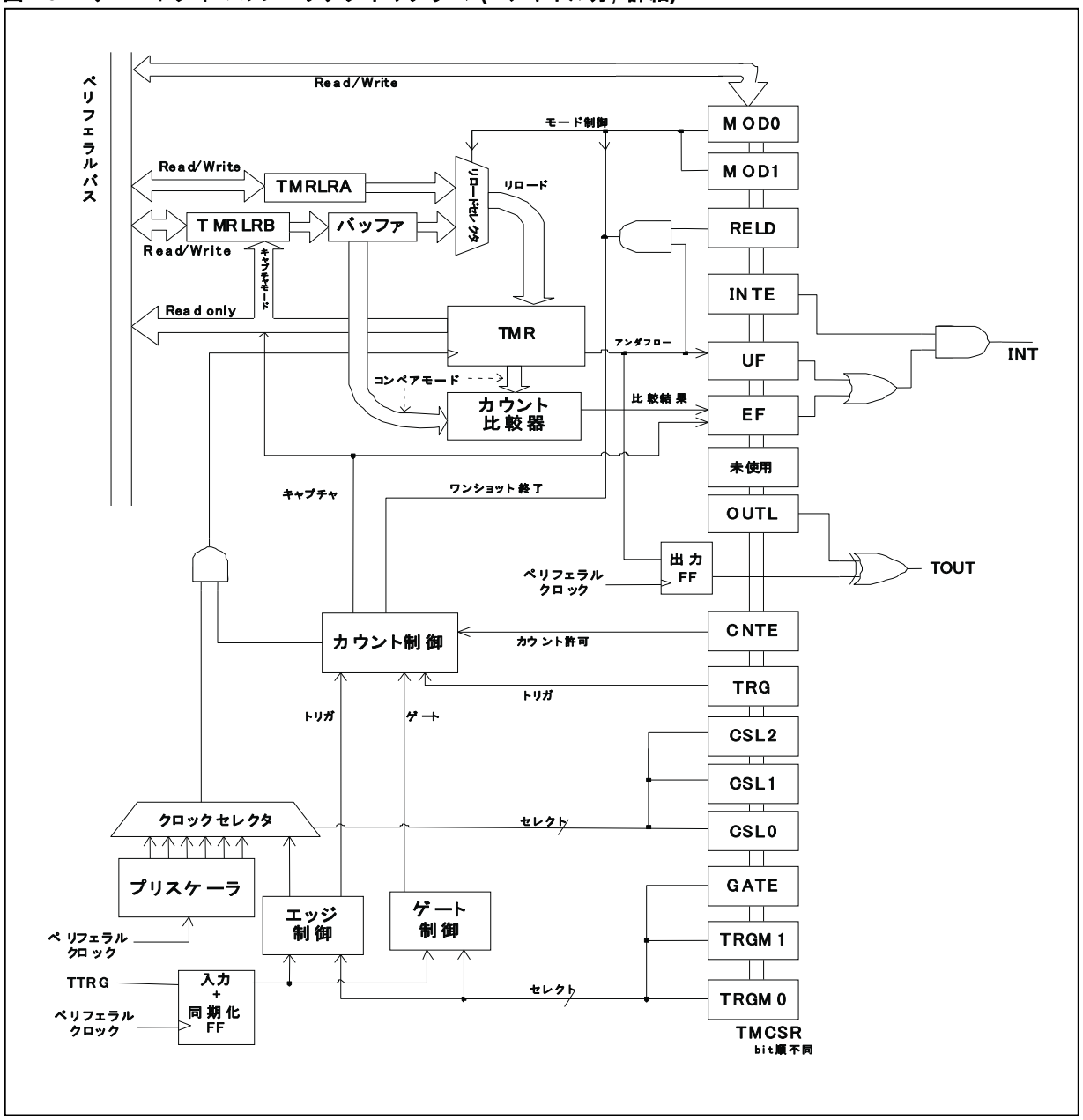
- インターバルタイマモード
  - ① シングルワンショット動作 => シングルショットタイマ
  - ② デュアルワンショット動作
  - ③ シングルリロード動作 => リロードタイマ
  - ④ デュアルリロード動作 => PPG(Programmable Pulse Generator)
  - ⑤ コンペアモード => アウトプットコンペア、PWM(Pulse Width Modulator)
  - ⑥ キャプチャモード(外部トリガ入力/ソフトウェアトリガ使用)
    => PWC(Pulse Width Counter)
  - ⑦ アンダフロー割込み / キャプチャ割込み
  - ⑧ 内部クロック 6 種類(周辺クロック(PCLK)の 2/4/8/16/32/64 分周)
  - ⑨ 外部トリガ入力(立上りエッジ / 立下りエッジ / 両エッジ)
  - ⑩ 外部ゲート入力

- イベントカウンタモード
  - ⑪ シングルワンショット動作
  - ⑫ デュアルワンショット動作
  - ⑬ シングルリロード動作
  - ⑭ デュアルリロード動作
  - ⑮ コンペアモード
  - ⑯ キャプチャモード(ソフトウェアトリガのみ)
  - ⑰ アンダフロー割込み / キャプチャ割込み / コンペア割込み
  - ⑱ 外部イベント入力エッジ検出(立上りエッジ検出 / 立下りエッジ検出 / 両エッジ検出)
  - ⑲ カスケードモード
    ch.0 出力を ch.1 入力, ch.1 出力を ch.2 入力, ch.2 出力を ch.3 入力に使用
    ch.4 出力を ch.5 入力, ch.5 出力を ch.6 入力に使用

20.3 構成

リロードタイマの構成について示します。

図 20-2 リロードタイマのブロックダイヤグラム (1 チャネル分, 詳細)



20.4 レジスタ

リロードタイマのレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子表

表 20-1 ベースアドレス (Base_addr)・外部端子表

チャネル	Base_addr	外部端子	
		TOUT	TTRG
0	0x0060	TOT0_0/TOT0_1/TOT0_2	TIN0_0/TIN0_1/TIN0_2
1	0x0100	TOT1_0/TOT1_1/TOT1_2	TIN1_0/TIN1_1/TIN1_2
2	0x0108	TOT2_0/TOT2_1/TOT2_2	TIN2_0/TIN2_1/TIN2_2
3	0x0110	TOT3_0/TOT3_1/TOT3_2	TIN3_0/TIN3_1/TIN3_2
4	0x0048	なし	なし
5	0x0050	なし	なし
6	0x0058	なし	なし

■ レジスタマップ

表 20-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0048	TMRLRA4		TMR4		16 ビットタイマリロードレジスタ A4 16 ビットタイマレジスタ 4
0x004C	TMRLRB4		TMCSR4		16 ビットタイマリロードレジスタ B4 コントロールステータスレジスタ 4
0x0050	TMRLRA5		TMR5		16 ビットタイマリロードレジスタ A5 16 ビットタイマレジスタ 5
0x0054	TMRLRB5		TMCSR5		16 ビットタイマリロードレジスタ B5 コントロールステータスレジスタ 5
0x0058	TMRLRA6		TMR6		16 ビットタイマリロードレジスタ A6 16 ビットタイマレジスタ 6
0x005C	TMRLRB6		TMCSR6		16 ビットタイマリロードレジスタ B6 コントロールステータスレジスタ 6
0x0060	TMRLRA0		TMR0		16 ビットタイマリロードレジスタ A0 16 ビットタイマレジスタ 0
0x0064	TMRLRB0		TMCSR0		16 ビットタイマリロードレジスタ B0 コントロールステータスレジスタ 0
0x0100	TMRLRA1		TMR1		16 ビットタイマリロードレジスタ A1 16 ビットタイマレジスタ 1
0x0104	TMRLRB1		TMCSR1		16 ビットタイマリロードレジスタ B1 コントロールステータスレジスタ 1
0x0108	TMRLRA2		TMR2		16 ビットタイマリロードレジスタ A2 16 ビットタイマレジスタ 2
0x010C	TMRLRB2		TMCSR2		16 ビットタイマリロードレジスタ B2 コントロールステータスレジスタ 2
0x0110	TMRLRA3		TMR3		16 ビットタイマリロードレジスタ A3 16 ビットタイマレジスタ 3
0x0114	TMRLRB3		TMCSR3		16 ビットタイマリロードレジスタ B3 コントロールステータスレジスタ 3

20.4.1 コントロールステータスレジスタ ：TMCSR (TiMer Control and Status Register)

コントロールステータスレジスタのビット構成について説明します。

動作モード、および割込みを制御します。
bit7, bit3～bit0 以外は bit1:CNTE=1 のときは書き換えできません。
bit15～bit8, bit6～bit4 の書換えと CNTE=1 書込みによるカウンタ動作許可の同時書込み、または bit15～bit8, bit6～bit4 の書換えと CNTE=0 書込みによる動作停止の同時書込みは可能です。

■ TMCSR： アドレス Base_addr + 06H (アクセス: バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	MOD[1:0]		TRGM[1:0]		CSL[2:0]			GATE
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EF	予約	OUTL	RELD	INTE	UF	CNTE	TRG
初期値	0	0	0	0	0	0	0	0
属性	R(RM1),W	R,W	R,W	R,W	R/W	R(RM1),W	R/W	R0,W

[bit15, bit14] MOD[1:0] (MODe) : モード選択ビット

MOD[1:0]	動作モード
00	シングルモード(初期値)
01	デュアルモード
10	コンペアモード
11	キャプチャモード

[bit13, bit12] TRGM[1:0] (TRiGger input Mode select) : TTRG 入力モード選択ビット
入力端子機能を制御します。インターバルタイマモードとイベントカウンタモードで機能が変わります。

【インターバルタイマモード時・トリガ入力(bit8:GATE=0)のとき】
TTRG 入力による、リロードトリガとなる有効外部エッジを以下のように選択します。

TRGM[1:0]	TTRG の有効外部エッジ
00	外部トリガ検出なし(初期値)
01	立上りエッジ
10	立下りエッジ
11	両エッジ

【インターバルタイマモード時・ゲート入力 (bit8:GATE=1)のとき】
TTRG 入力中にカウンタ許可となる端子レベルを以下のように選択します。

TRGM[1:0]	TTRG の有効レベル
x0	TTRG 端子"L"入力期間のみカウント(初期値)
x1	TTRG 端子"H"入力期間のみカウント

【イベントカウンタモード時の有効エッジ設定】
イベントカウンタモード時、外部イベント検出用のエッジを以下のように選択します。外部イベントを検出すると、カウンタはその都度ダウンカウントします。外部イベント選択時、bit8:GATE ビットの設定は無効となります。

TRGM[1:0]	カウント対象エッジ
00	予約
01	立上りエッジ
10	立下りエッジ
11	両エッジ

[bit11～bit9] CSL[2:0] (Conut source SeLect) : **カウントソース選択ビット**
カウントソース選択ビットです。内部クロック(周辺クロック*)と外部イベント(TTRG 入力)より、カウントソースを以下に示すように選択します。イベントカウンタモードを設定した場合のカウント有効エッジは bit13, bit12:TRGM[1:0]ビットで設定します。

CSL[2:0]	カウントソース	動作モード
000	周辺クロックの 2 分周 (初期値)	インターバル タイマモード
001	周辺クロックの 4 分周	
010	周辺クロックの 8 分周	
011	周辺クロックの 16 分周	
100	周辺クロックの 32 分周	
101	周辺クロックの 64 分周	
110	カスケードモード (ch.0:TTRG0,ch.1:TOUT0,ch.2:TOUT1,ch.3:TOUT2, ch.4:TTRG4,ch.5:TOUT4,ch.6:TOUT5)	イベントカウンタモード
111	外部イベント(TTRG 入力)	

*: ch.0～ch.3 は PLCK2, ch.4～ch.6 は PLCK1

[bit8] GATE (GATE input enable) : **ゲート入力許可ビット**
インターバルタイマモード時(bit11～bit9:CSL[2:0]=000～101)の入力端子(TTRG)の機能を以下に示すように制御します。

GATE	TTRG 入力端子機能
0	トリガ入力として使用(初期値)
1	ゲート入力として使用

イベントカウンタモード時は、本ビットによる動作への影響はありません。

[bit7] EF (Extended Flag) : **拡張割込みフラグ**
コンペアモード時にコンペア一致割込み、またはキャプチャモード時にキャプチャ入力割込みが発生した事を示すフラグです。

セット要因	【イベントカウンタモードのコンペアモード時】 コンペア一致(TMR＝TMRLRB)からのカウントダウン発生 【キャプチャモード時】 キャプチャ入力(リトリガ)
クリア要因	本ビットへの"0"書込み、またはリセット

このビットへの"1"書込みは動作に影響しません。コンペアモードでは、カウントクロックに同期して、セット/クリアを行います。リードモディファイライト系命令における読出し値は、常に"1"となります。

[bit6] **予約**
予約ビットです。書込みをしても動作に影響しません。

[bit5] OUTL (OUTput Level) : **出力極性設定ビット**
タイマ出力端子(TOUT)の出力極性を制御します。

OUTL	TOUT の初期値	TOUT の初期出力レベル
0	正極性(初期値)	L レベル
1	負極性	H レベル

[bit4] RELD (RELoaD enable) : **リロード動作許可ビット**
アンダフロー発生時のリロード動作を以下に示すように設定します。

RELD	動作モード	動作内容
0	ワンショット モード	カウンタのアンダフローが発生すると同時にカウント動作を停止します。次にトリガを入力するまでリロードを行いません。*(初期値)
1	リロードモード	カウンタのアンダフローが発生する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。

*:ただし、デュアルワンショット機能では TMRLRA のアンダフローと同時に TMRLRB のリロードを行い、カウントを継続します。その後、TMRLRB のアンダフローと同時にカウント動作を停止します。

[bit3] INTE (INTerrupt Enable) : **割込み要求許可ビット**
アンダフロー発生/コンペア一致(イベントカウンタモード時)/キャプチャ時の割込み要求を以下に示すように制御をします。

INTE	動作内容
0	割込み禁止(UF/EF ビットがセットされても割込みは発生しません。) (初期値)
1	割込み許可(UF/EF ビットがセットされると割込み要求を発生します。)

[bit2] UF (Under flow Flag) : **アンダフローフラグ**
カウンタの値が、0x0000 からダウンカウント行ったときに、アンダフローが発生した事を示すフラグです。

セット要因	カウンタのアンダフローの発生
クリア要因	本ビットへの"0"書込み、またはリセット

[bit1] CNTE (timer CouNTer Enable) : **タイマカウント許可ビット**
タイマの動作を以下のように制御します。

CNTE	動作内容
0	動作禁止(初期値)
1	動作許可(起動トリガ待ち)

[bit0] TRG (software TRiGger) : **ソフトウェアトリガビット**
タイマのソフトウェアトリガを発生します。ソフトウェアトリガが発生すると、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。

TRG	動作内容
"0"書込み	動作に影響しません
"1"書込み	ソフトウェアトリガ発生

このビットへの"0"書込みは、動作に影響しません。読出し値は常に"0"となります。
このレジスタによるトリガ入力は、bit1:CNTE =1 のときのみ有効となります。
TRG ビットの"1"書込みは、タイマが起動状態(bit1:CNTE=1)のとき、動作モードに関わらす常に有効トリガを発生します。

20.4.2 16 ビットタイマレジスタ ：TMR (16bit TiMer Register)

16 ビットタイマレジスタのビット構成について示します。

タイマのカウント値を読み出すことができます。

このレジスタには必ず 16 ビットアクセスをしてください。

■ TMR： アドレス Base_addr + 02_H (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	TMR[15:0]					
初期値	X	X	・・・	X	X	X
属性	R,WX	R,WX	・・・	R,WX	R,WX	R,WX

[bit15～bit0] TMR[15:0] (TiMeR) : **16 ビットタイマ**
16 ビットタイマのカウンタ値を読み出す事ができるレジスタです。初期値は不定です。

20.4.3 16 ビットタイマリロードレジスタ A, 16 ビットタイマリロードレジスタ B : TMRLRA, TMRLRB (16bit TiMer ReLoad Register A/B)

16 ビットタイマリロードレジスタ A, 16 ビットタイマリロードレジスタ B のビット構成について示します。

TMRLRA はカウント初期値を設定します。TMRLRB は動作モードにより機能が異なります。

このレジスタには必ず 16 ビットアクセスをしてください。

■ TMRLRA : アドレス Base_addr + 00H (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	TMRLRA[15:0]					
初期値	X	X	・・・	X	X	X
属性	R/W	R/W	・・・	R/W	R/W	R/W

■ TMRLRB : アドレス Base_addr + 04H (アクセス: ハーフワード)

	bit15	bit14	・・・	bit2	bit1	bit0
	TMRLRB[15:0]					
初期値	X	X	・・・	X	X	X
属性	R,W	R,W	・・・	R,W	R,W	R,W

[bit15～bit0] TMRLRA[15:0] (TiMer ReLoad Register A) : 16 ビットリロード設定レジスタ A

[bit15～bit0] TMRLRB[15:0] (TiMer ReLoad Register B) : 16 ビットリロード設定レジスタ B

TMRLRA レジスタは、カウント初期値を保持しておくレジスタです。 TMRLRA は TMCSR レジスタの bit15, bit14:MOD[1:0]の設定に関係なく全モードで使用されます。

TMRLRB は TMCSR レジスタの bit15, bit14:MOD[1:0]の設定により以下のように使用されます。

モード	MOD[1:0]	TMRLRB の機能
シングルモード	00	使用しません
デュアルモード	01	H 幅(OUTL=0 のとき)カウンタ値
コンペアモード	10	コンペアレジスタ(H 幅設定 OUTL=0 のとき)
キャプチャモード	11	キャプチャレジスタ(リトリガ入力時の TMR 値)

カウンタ値として使用の際は、0x0000 を書き込んだ時は 1 カウント、0xFFFF を書き込んだ時は 65,536 カウントするとアンダフローを発生します。

タイマ出力波形(TOUT)の H 幅・L 幅は MOD[1:0](TMCSR レジスタの bit15, bit14)、RELD(TMCSR レジスタの bit4)、OUTL(TMCSR レジスタの bit5)ビット設定と TMRLRA/B レジスタ値により決まります。出力される波形(TOUT)の H 幅・L 幅の設定を下表に示します。

MOD[1:0]	モード	RELD	OUTL	TOUT 出力	
				H 幅	L 幅
00	シングル	0	0	TMRLRA+1	―
			1	―	TMRLRA+1
		1	0	TMRLRA+1	
			1		
01	デュアル	0	0	TMRLRB+1	TMRLRA+1
			1	TMRLRA+1	TMRLRB+1
		1	0	TMRLRB+1	TMRLRA+1
			1	TMRLRA+1	TMRLRB+1
10	コンペア	0	0	以下を参照*	
			1		
		1	0		
			1		
11	キャプチャ	0	0	TMRLRA+1	―
			1	―	TMRLRA+1
		1	0	TMRLRA+1	
			1		

*: コンペアモード時の H 幅・L 幅は以下となります。

■ TMRLRB < TMRLRA のとき
(OUTL=0)　TMRLRA-TMRLRB + 1 の"L"幅、TMRLRB の"H"幅
(OUTL=1)　TMRLRA-TMRLRB + 1 の"H"幅、TMRLRB の"L"幅

■ TMRLRB = 0 のとき
(OUTL=0)　"L"出力固定
(OUTL=1)　"H"出力固定

■ TMRLRB > TMRLRA のとき
(OUTL=0)　"H"出力固定
(OUTL=1)　"L"出力固定

■ TMRLRB = TMRLRA のとき
(OUTL=0)　1 サイクルの"L"出力、TMRLRB の"H"幅
(OUTL=1)　1 サイクルの"H"出力、TMRLRB の"L"幅

インターバルタイマモード時でシングルモード・デュアルモードとして使用している時の TOUT 出力時間(TOUT)を表す式を以下に示します。

TOUT = (本レジスタの設定値 + 1) × カウントソースの周期
*: 上式はインターバルタイマモードのときのみ有効

20.5 動作説明

リロードタイマの動作について説明します。

20.5.1. 設定

20.5.2. 動作手順

20.5.3. 各カウンタ動作の説明

20.5.4. カスケード入力

20.5.5. 同時動作の優先順位

20.5.1 設定

リロードタイマの設定について説明します。

本タイマは「カウントソース」(TMCSR:CSL[2:0]で選択)と、カウンタ動作({TMCSR:MOD[1:0], TMCSR:RELD})で動作を設定します。

20.5.1.1 カウントソース

リロードタイマのカウントソースについて説明します。

TMCSR:CSL[2:0]により、ダウンカウンタのダウンカウント条件を選択します。

表 20-3 カウントソース一覧

CSL[2:0]	カウントソース	動作モード
000	周辺クロックの 2 分周 (初期値)	インターバル タイマモード
001	周辺クロックの 4 分周	
010	周辺クロックの 8 分周	
011	周辺クロックの 16 分周	
100	周辺クロックの 32 分周	
101	周辺クロックの 64 分周	
110	カスケードモード (ch.0:TTRG0, ch.1:TOUT0, ch.2:TOUT1, ch.3:TOUT2 ch.4:TTRG4,ch.5:TOUT4,ch.6:TOUT5)	イベントカウンタ モード
111	外部イベント(TTRG 入力)	

20.5.1.2 タイマのアンダフロー周期

タイマのアンダフロー周期について説明します。

カウンタが 0x0000 からダウンカウントする時をアンダフローとしています。タイマがカウント動作を初めてからアンダフローが発生するまでの時間(周期)をリロードレジスタ(TMRLRA/ TMRLRB)に設定します。リロードレジスタのロード後は"リロードレジスタの設定値＋ 1"カウントでアンダフローを発生します。インターバルタイマモード時のタイマのアンダフロー周期 TUF は以下で表せます。

TUF = 周辺クロック(PCLK)周期 × プリスケーラの分周値(2～64)
× (リロードレジスタ値(TMRLRA/B) ＋ 1)

20.5.1.3 トリガ

リロードタイマのトリガについて説明します。

トリガには以下の 2 種類があります。

■ ソフトフェアトリガ・・・TMCSR:TRG に"1"書込みで発生。
■ 外部端子トリガ・・・TTRG 端子から入力。

イベントカウンタモードでは TTRG 端子はカウントソースとして使用されるため、常にソフトフェアトリガを使用します。インターバルタイマモード時は TMCSR レジスタで設定します。

20.5.1.4 ゲート

リロードタイマのゲートについて説明します。

インターバルタイマモード時・ゲート入力(TMCSR:GATE =1)設定において、TTRG 外部端子を使用して、カウンタのダウンカウントを止めることができます。

表 20-4 TTRG 有効レベル

TRGM[0]	TTRG の有効レベル
0	TTRG 端子"L"入力期間のみカウント(初期値)
1	TTRG 端子"H"入力期間のみカウント

20.5.1.5 カウンタ動作選択

カウンタ動作選択について説明します。

カウンタのアンダフローが発生した時の動作をモード選択ビット(TMCSR レジスタの bit15, bit14: MOD[1:0])と、リロード動作許可ビット(TMCSR レジスタの bit4:RELD)で選択します。各モードでの動作詳細は各カウンタ動作の項を参照してください。

表 20-5 カウンタ動作一覧

MOD[1:0]	RELD	アンダフロー発生時の動作	カウンタ動作名
00	0	カウントを 0xFFFF で停止	シングルワンショット
	1	TMRLRA をリロード	シングルリロード
01	0	①TMRLRB をリロード ②カウントを 0xFFFF で停止 (「20.5.3.3. デュアルワンショット動作」を参照)	デュアルワンショット
	1	TMRLRA、TMRLRB を交互にリロード	デュアルリロード
10	0	カウントを 0xFFFF で停止	コンペアワンショット
	1	TMRLRA をリロード	コンペアリロード
11	0	カウントを 0xFFFF で停止	キャプチャワンショット
	1	TMRLRA をリロード	キャプチャリロード

20.5.1.6 TOUT 端子レベル設定

TOUT 端子レベル設定について説明します。

TMCSR レジスタの bit5:OUTL ビットによって端子出力極性の設定を行います。

以下に各機能でのイベントと TOUT の関係を示します。

下図中の UF(アンダフロー)の項の A/B は、TMRLRA/TMRLRB どちらのデータをロードした値でのダウンカウントでアンダフローが発生したかを表します。CMP(比較一致)は、TMRLRB = TMR からダウンカウントが発生したタイミングを表します。

図 20-3 各イベントでの TOUT 出力変化 (1 / 3)



図 20-4 各イベントでの TOUT 出力変化 (2 / 3)

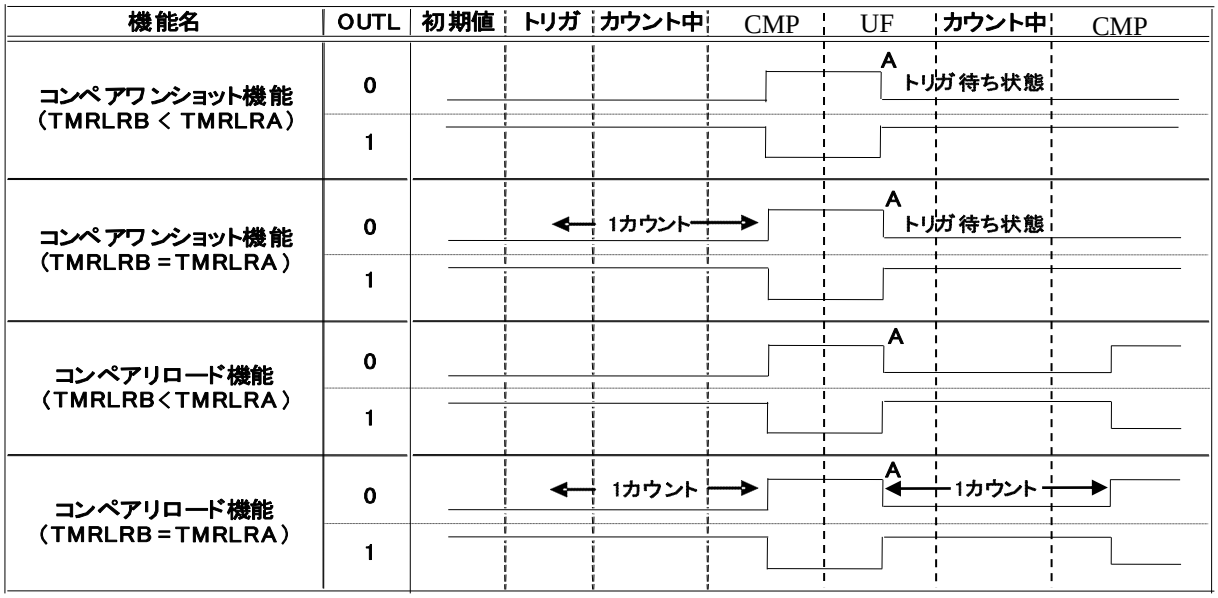
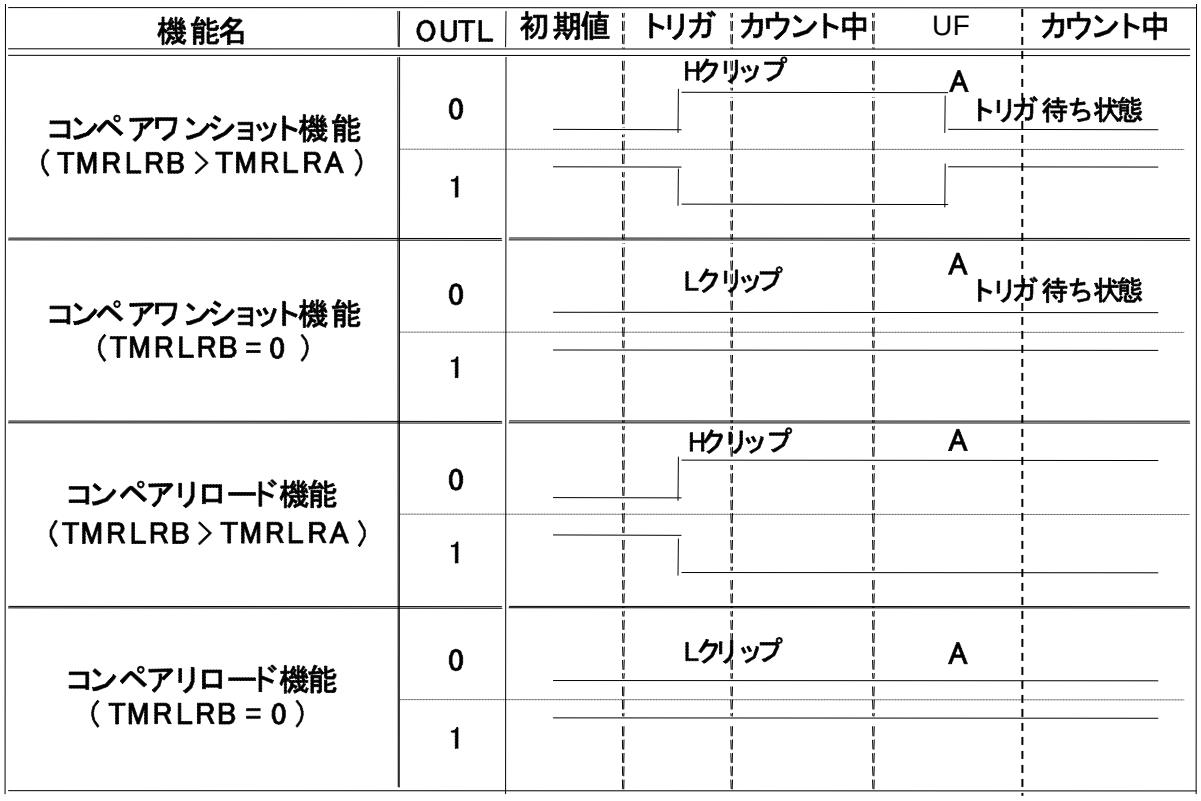


図 20-5 各イベントでの TOUT 出力変化 (3 / 3)



20.5.2 動作手順

動作手順について示します。

20.5.2.1. 起動

20.5.2.2. リトリガ

20.5.2.3. アンダフロー/リロード

20.5.2.4. 割込み要求発生

20.5.2.5. レジスタ書込みとタイマ動作の同時動作

20.5.2.1 起動

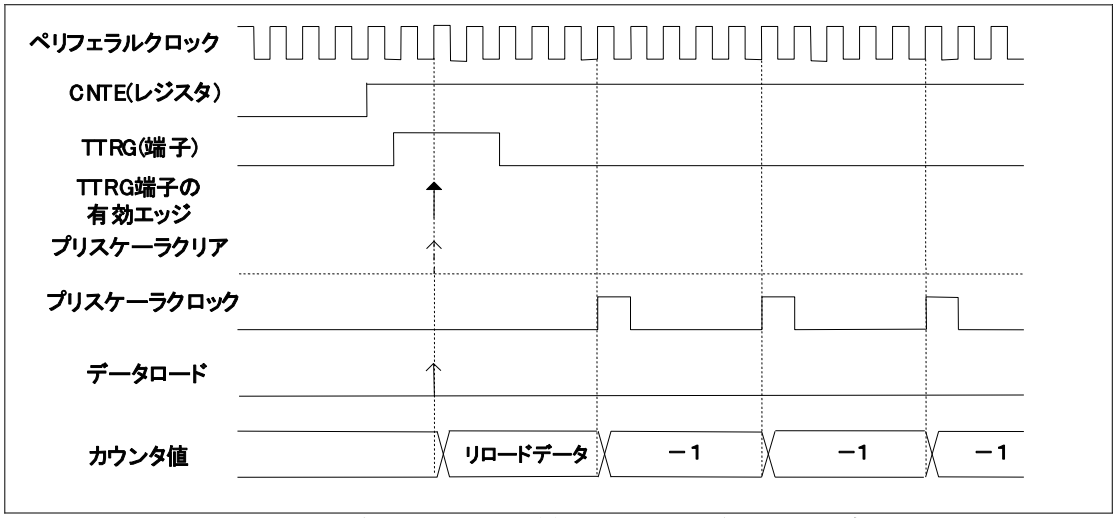
起動について説明します。

TMCSR レジスタの bit1:CNTE ビットに"1"を書き込むとカウンタは起動トリガ待ち状態になります。

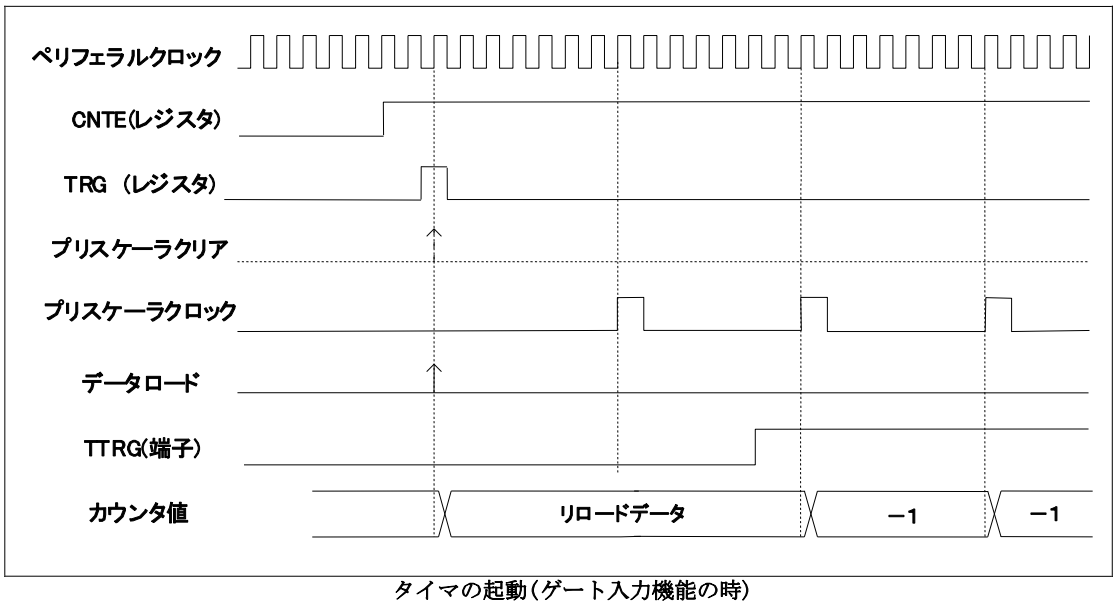
- TTRG 入力がトリガ入力機能時
  起動トリガ待ち状態中に TMCSR レジスタの bit0:TRG ビットへの"1"書込み、または TTRG 入力による外部トリガが入力されると、プリスケーラのクリアが発生し、タイマはリロードレジスタから値をロードしてダウンカウント動作を行います。TTRG 入力は、2×T (T は周辺クロック(PCLK)周期)以上のパルスを入力してください。

- TTRG 入力がゲート入力機能時
  起動トリガ待ち状態中に TMCSR レジスタの bit0:TRG ビットへ"1"書込みを行うと、プリスケーラのクリアが発生し、タイマはリロードレジスタから値をロードし有効入力極性待ち状態になります。有効入力極性待ち状態で TTRG 入力から有効な極性のゲート入力があると、タイマはダウンカウント動作を行います。TTRG 入力は、2×T (T は周辺クロック(PCLK)周期)以上のパルスを入力してください。

図 20-6 タイマの起動



タイマの起動(トリガ入力機能・立上りエッジトリガ選択時)



タイマの起動(ゲート入力機能の時)

20.5.2.2 リトリガ

リトリガについて説明します。

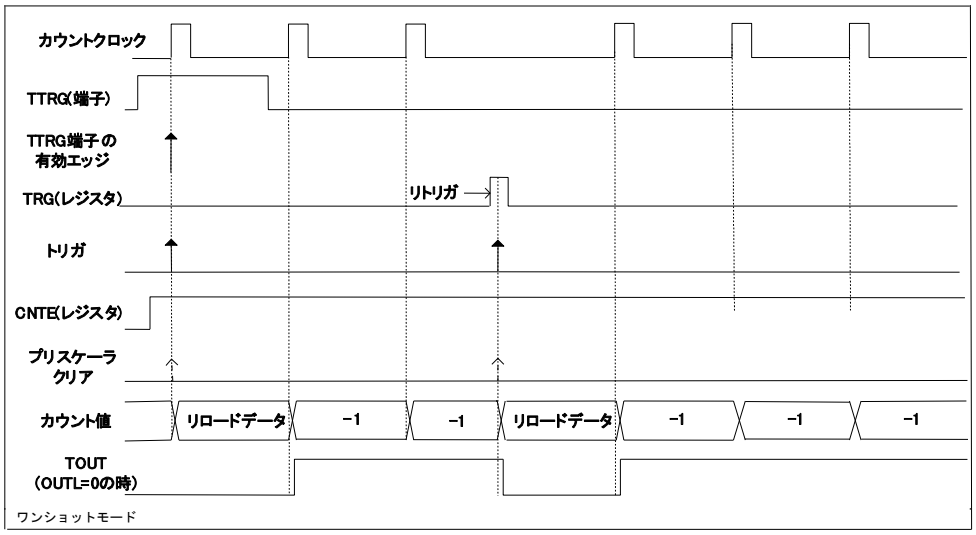
タイマのカウント中にトリガが発生した場合の事をリトリガとよびます。その際、

7. TOUT を初期化
8. リロードレジスタの値をカウンタにロード
9. 6 ビットプリスケーラのクリア
10. カウント継続

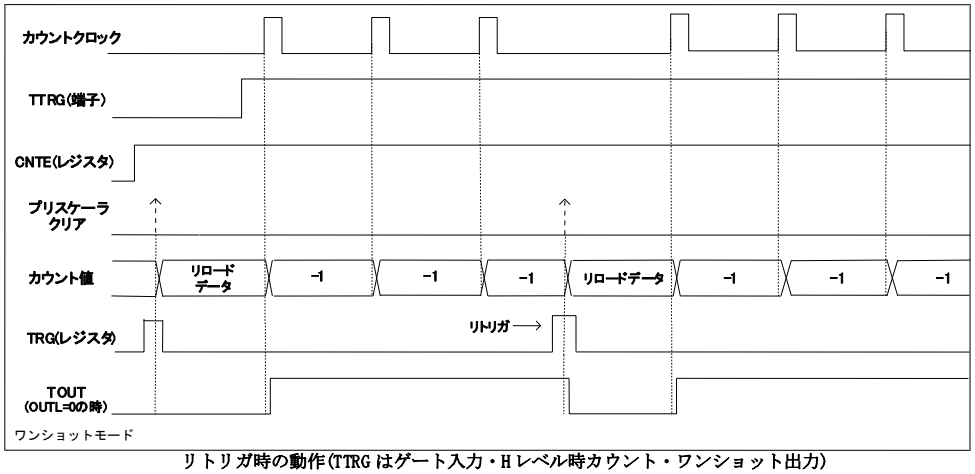
を行います。キャプチャモード時のみ、リトリガ発生によりカウント中の値を TMRLRB に転送し、TMCSR レジスタの EF ビットをセットします。

＜注意事項＞
ワンショットモードではリトリガ時、TOUT は初期化されません。

図 20-7 リトリガ時の動作



リトリガ時の動作(TTRG はトリガ入力・立上りエッジトリガ時・ワンショット出力)



リトリガ時の動作(TTRG はゲート入力・H レベル時カウント・ワンショット出力)

20.5.2.3 アンダフロー/リロード

アンダフロー/リロードについて説明します。

タイマが 0x0000 からダウンカウントする時をアンダフローとしています。アンダフローが発生すると、TMCSR レジスタの bit2:UF ビットがセットされます。タイマは"リロードレジスタの設定値＋1"カウントでアンダフローを発生します。

20.5.2.4 割込み要求発生

割込み要求発生について説明します。

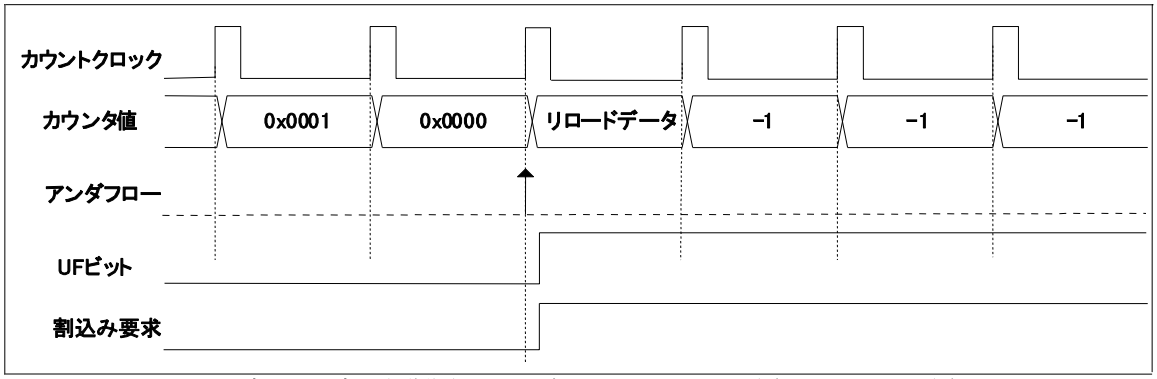
TMCSR レジスタの bit3:INTE ビットが"1"のときに、bit2:UF ビット/bit7:EF ビットがセットされると割込み要求が発生します。インターバルタイマモードでは、以下の条件のときに UF ビット/EF ビットがセットされます。

- UF ビットがセット ： カウンタのアンダフローが発生した時
- EF ビットがセット ： キャプチャモード時にキャプチャ入力が発生した時

TMCSR レジスタの bit2:UF ビットのセットと、UF ビットへの"0 書込み"によるクリアが同時に発生した時は、UF ビットへの"0 書込み"は無効となり UF ビットがセットされます。また、bit7:EF ビットのセットと EF ビットへの"0 書込み"によるクリアが同時に発生した時は、EF ビットへの"0 書込み"は無効となり EF ビットがセットされます。

割込み要求発生例を以下に示します。

図 20-8 UF 割込み要求出力動作例



UF 割込み要求出力動作(TMCSR レジスタの bit4:RELD= ‘1’・bit3:INTE= ‘1’

20.5.2.5 レジスタ書込みとタイマ動作の同時動作

レジスタ書込みとタイマ動作の同時動作について説明します。

ユーザの操作によるレジスタ書込みとタイマ動作が同時に発生した時の実行動作について下表に示します。

表 20-6 同時動作

レジスタ書込み	タイマ動作	実行する動作
UF ビットへの"0"書込みによるクリア	UF ビットのセット	UF ビットのセット ("0"書込みは無効になる)
EF ビットの"0"書込みによるクリア	EF ビットのセット	EF ビットのセット ("0"書込みは無効になる)
リロードレジスタへの書込み	リトリガによるタイマのロード	旧データのリロード (新しく書き換えた値は次回)

20.5.3 各カウンタ動作の説明

各カウンタ動作について示します。

20.5.3.1. シングルワンショット動作

20.5.3.2. シングルリロード動作

20.5.3.3. デュアルワンショット動作

20.5.3.4. デュアルリロード動作

20.5.3.5. コンペアワンショット動作

20.5.3.6. コンペアリロード動作

20.5.3.7. キャプチャモード

20.5.3.1 シングルワンショット動作

シングルワンショット動作について説明します。

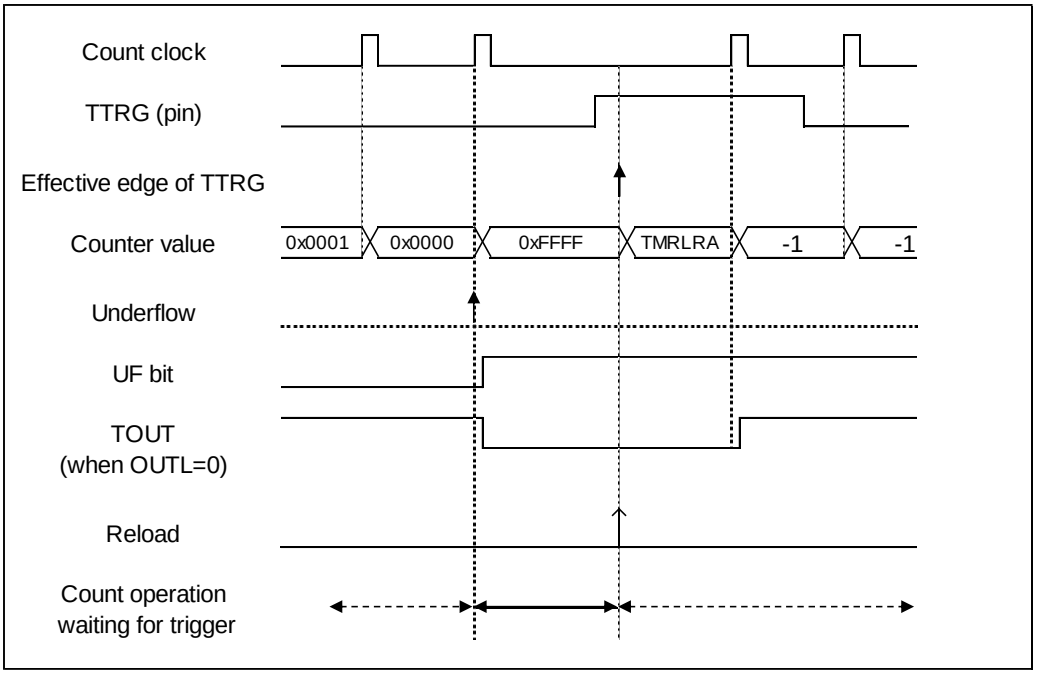
TMCSR レジスタの bit15, bit14:MOD[1:0] =00, bit4:RELD=0 のとき、タイマはアンダフローの発生により 0xFFFF で停止するシングルワンショット動作を行います。

シングルワンショット設定時にアンダフローが発生した時は以下の動作を行います。

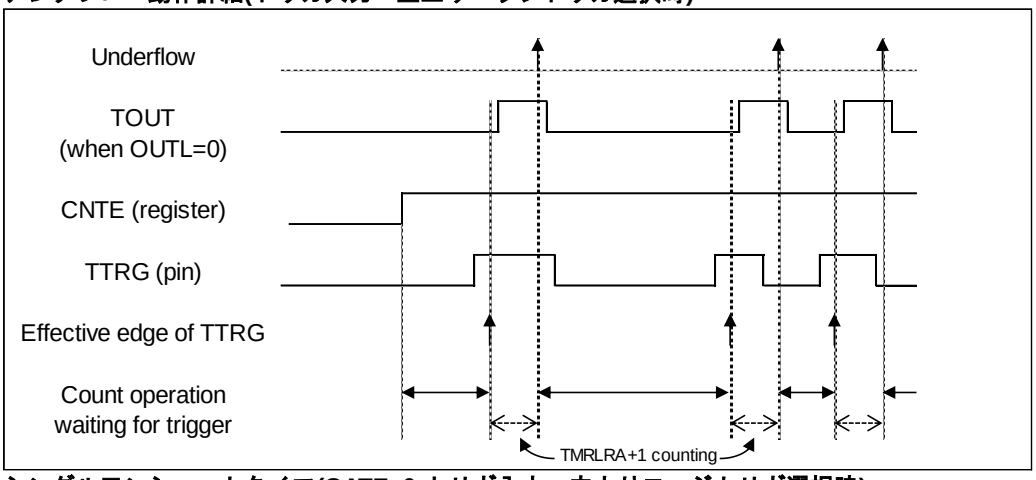
- TMCSR レジスタの UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- 0xFFFF でカウント停止
- TOUT 出力を初期化
- タイマはトリガ待ち状態

シングルワンショットタイマでは TMRLRA はリロード時のカウンタの初期値になります。TMRLRB は使用しません。

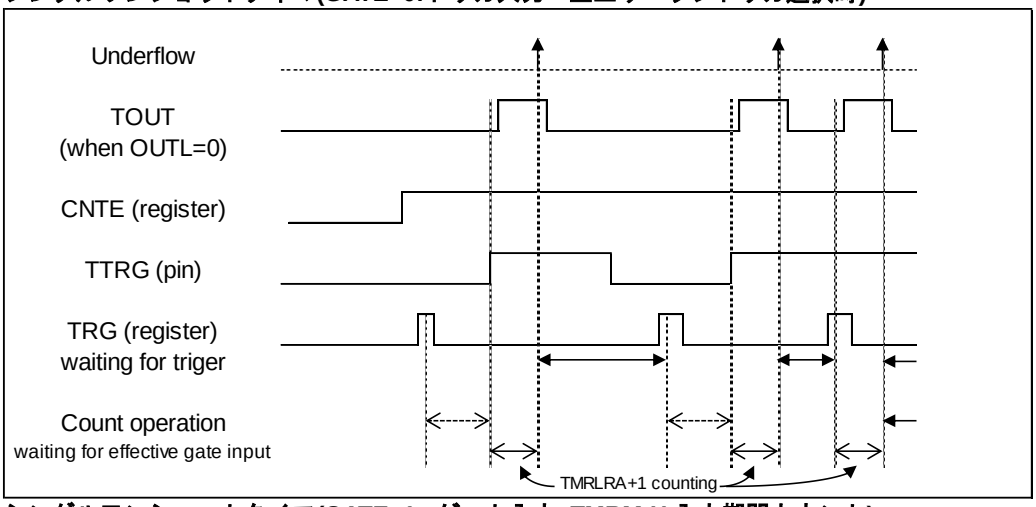
図 20-9 シングルワンショット動作



アンダフロー動作詳細(トリガ入力・立上りエッジトリガ選択時)



シングルワンショットタイマ(GATE=0:トリガ入力・立上りエッジトリガ選択時)



シングルワンショットタイマ(GATE=1: ゲート入力, TMRM:H 入力期間カウント)

20.5.3.2 シングルリロード動作

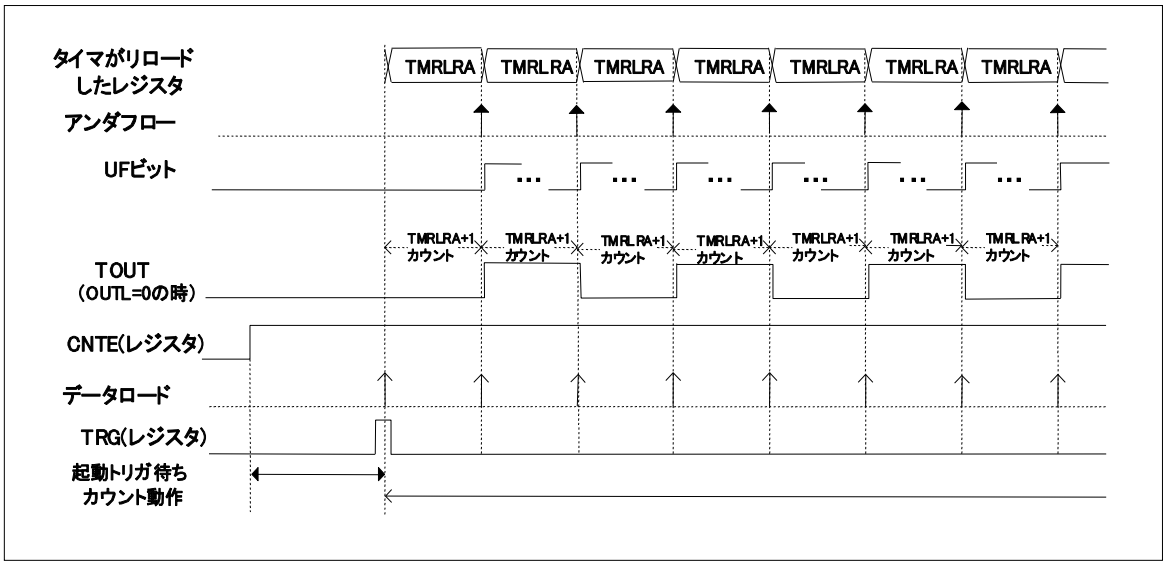
シングルリロード動作について説明します。

TMCSR レジスタの bit15, bit14:MOD[1:0] =00, bit4:RELD =1 のとき、シングルリロード動作を行います。

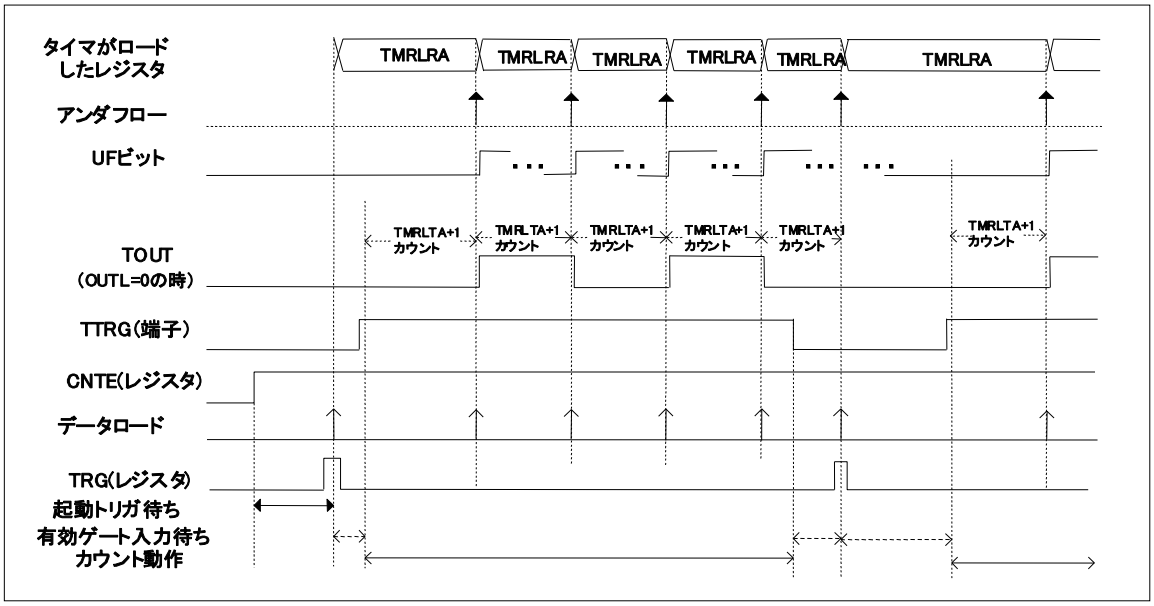
シングルリロード動作は、トリガ入力により TMRLRA から値をタイマへロードして、ダウンカウント動作を始めます。アンダフローが発生すると、再び TMRLRA から値をリロードしてダウンカウント動作を継続します。TMRLRA の値はタイマがリロードする時間を表します。TMRLRB レジスタは使用しません。シングルリロード設定時に、アンダフローが発生した時は以下の動作を行います。

- TMCSR レジスタの bit2:UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- TMRLRA レジスタをカウンタにロード
- TOUT 出力を反転
- ダウンカウント継続

図 20-10 シングルリロード動作



シングルリロード機能(GATE='0':トリガ入力)



シングルリロード機能(GATE='1':ゲート入力、TRGM:H入力期間カウント)

20.5.3.3 デュアルワンショット動作

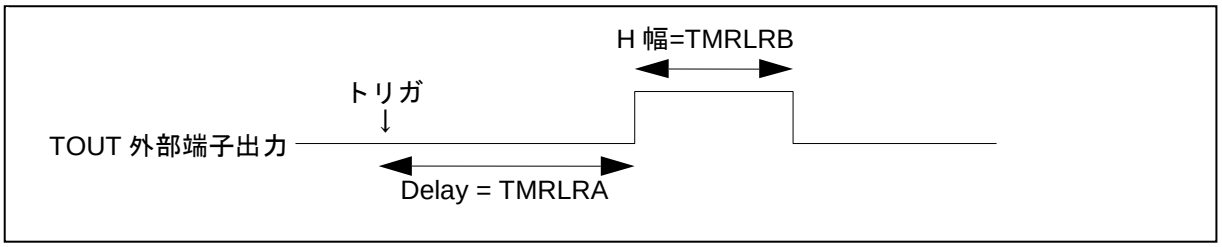
デュアルワンショット動作について説明します。

TMCSR レジスタの bit15, bit14:MOD[1:0] =01, bit4: RELD =0 のとき、タイマはデュアルワンショット動作を行います。ワンショット PPG として使用できます。

デュアルワンショット動作では、TMRLRA→TMRLRB の順に 1 回ずつ値をカウンタへロードして、それぞれダウンカウントを行い、2 回目のアンダフローによりカウントを停止します。

TMCSR レジスタの bit5:OUTL=0 のとき、TMRLRA の値はタイマの起動(TOUT 出力は L レベル)から TOUT 出力が H へトグルするまでの時間、TMRLRB の値は TOUT 出力の H 幅の時間を示します。

図 20-11 TOUT パルス幅



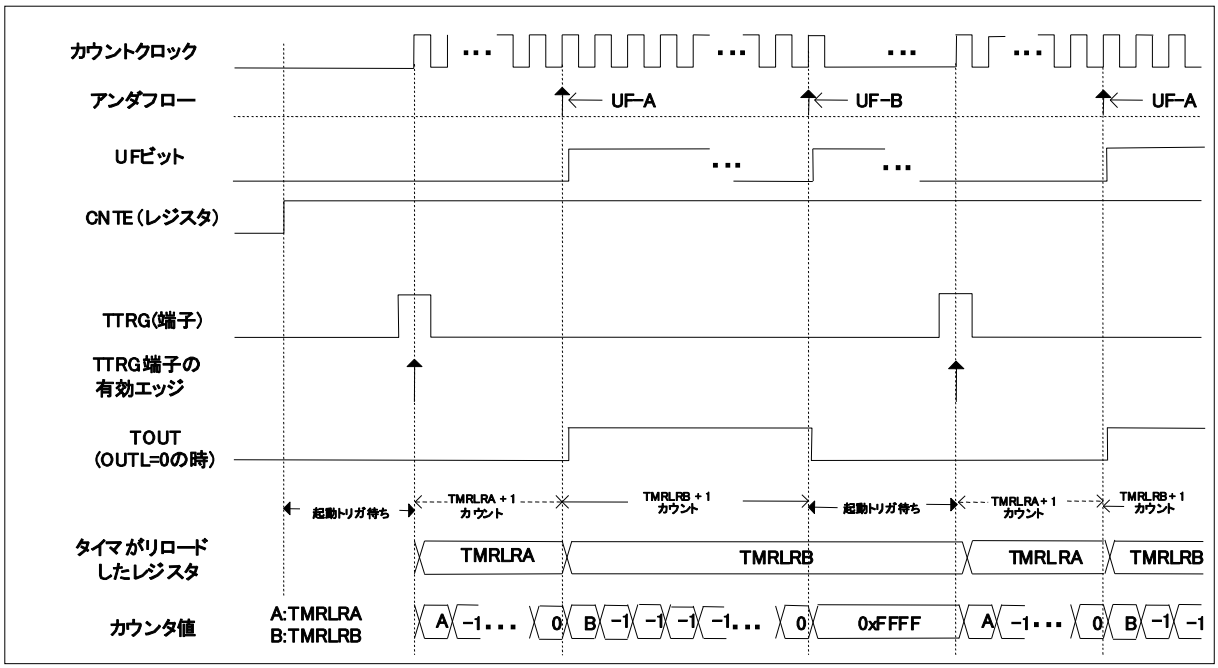
1 回目のアンダフロー(UF-A)が発生した時は以下の動作を行います。

- TMCSR レジスタの bit2:UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- TMRLRB をカウンタにロード
- TOUT 出力を反転
- TMRLRB からダウンカウント開始

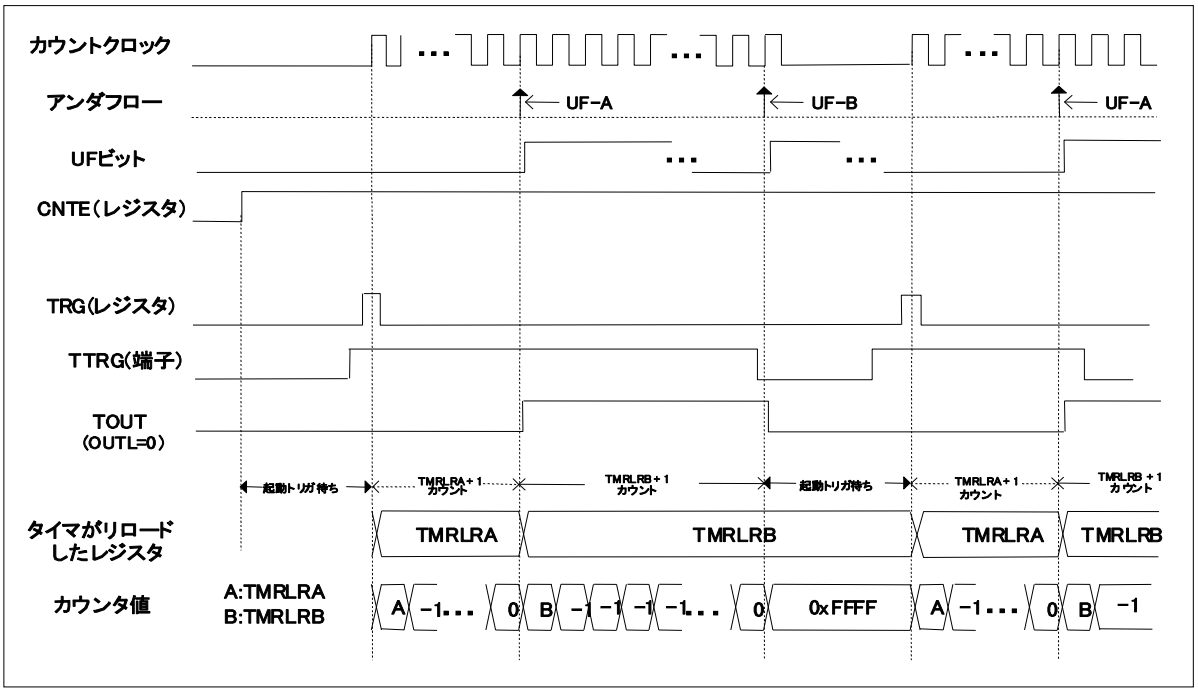
2 回目のアンダフロー(UF-B)が発生した時は以下の動作を行います。

- TMCSR レジスタの bit2:UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- 0xFFFF でカウント停止
- TOUT 出力を初期化
- タイマは起動トリガ待ち状態へ

図 20-12 デュアルワンショット動作



デュアルワンショット動作(トリガ入力・立上りエッジトリガ選択時)



デュアルワンショット動作(ゲート入力)

20.5.3.4 デュアルリロード動作

デュアルリロード動作について説明します。

TMCSR レジスタの bit15, bit14:MOD[1:0] =01, bit4: RELD =1 のとき、タイマはデュアルリロード動作を行います。

デュアルリロード動作では、TMRLRA をカウンタへロードして、ダウンカウントを行い、アンダフローが発生すると、TMRLRB をカウンタへロードして、ダウンカウントを行いアンダフローが発生すると、TMRLRA をカウンタへロードしてダウンカウント・・・のように TMRLRA と TMRLRB を交互にロードしカウントを行う機能です。

TMCSR レジスタの bit5:OUTL=0 のとき、TMRLRA の値はタイマの起動(TOUT 出力は L レベル)から TOUT 出力が H へトグルするまでの時間、TMRLRB の値は TOUT 出力の H 幅の時間を示します。

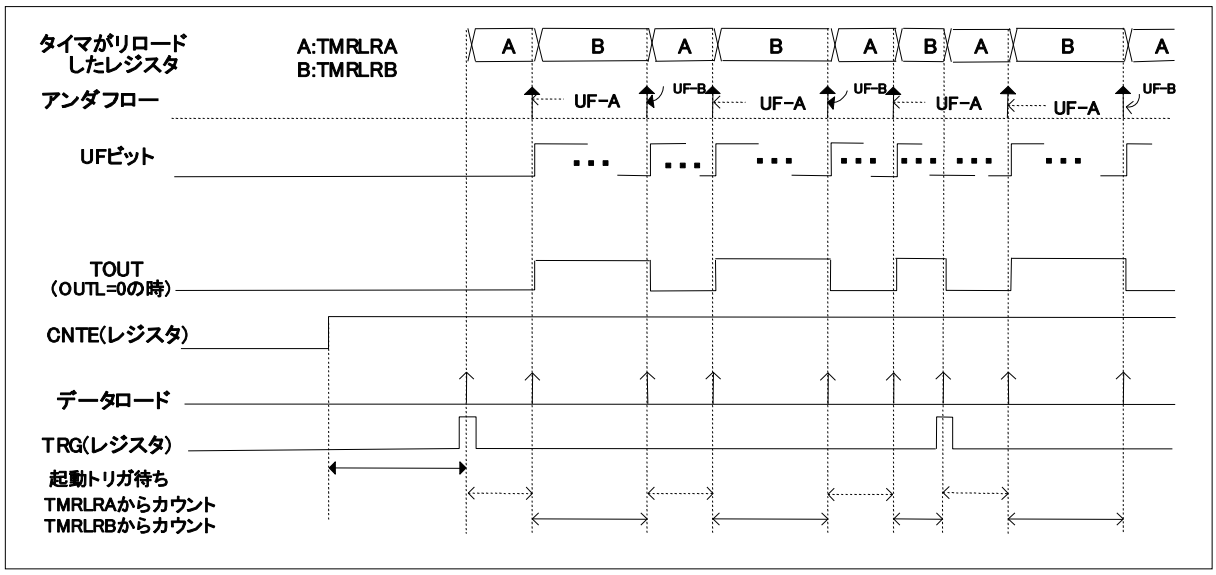
TMRLRA から値をロードした後のダウンカウントでアンダフロー(UF-A)が発生した時は以下の動作を行います。

- TMCSR レジスタの bit2:UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- TMRLRB をカウンタにロード
- TOUT 出力を反転
- TMRLRB からダウンカウント開始

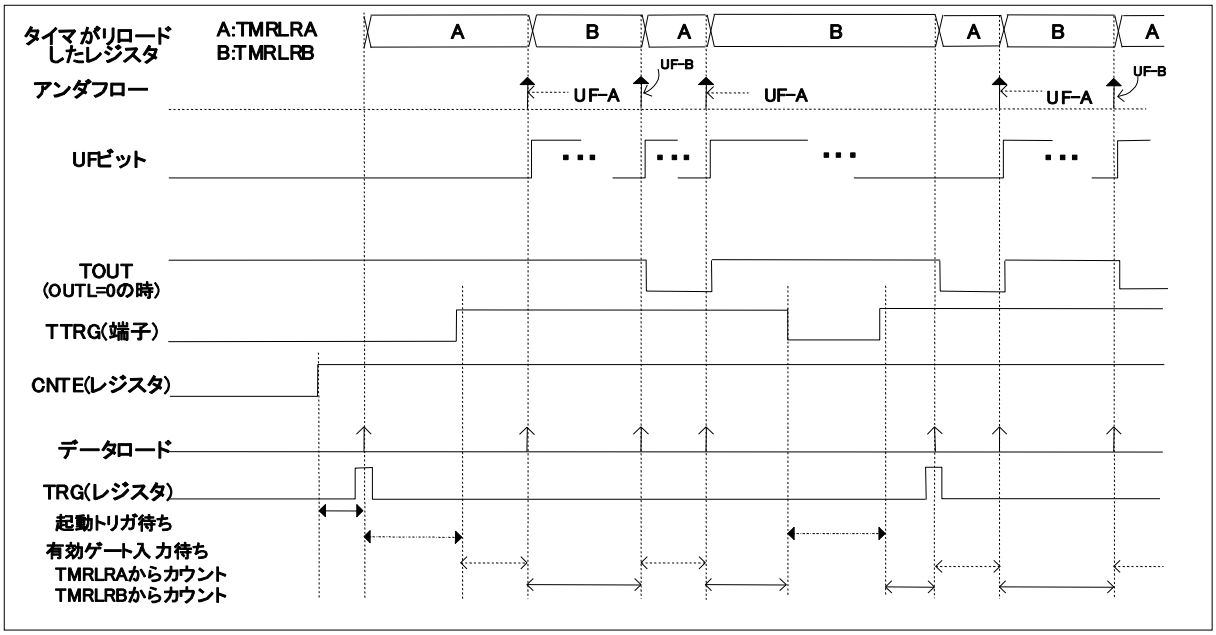
TMRLRB から値をロードした後のダウンカウントでアンダフロー(UF-B)が発生した時は以下の動作を行います。

- TMCSR レジスタの bit2:UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- TMRLRA をカウンタにロード
- TOUT 出力を反転
- TMRLRA からダウンカウント開始

図 20-13 デュアルリロード動作



デュアルリロード機能（GATE= ‘0’ :トリガ入力）



デュアルリロード機能（GATE= ‘1’ :ゲート入力、H入力期間カウント）

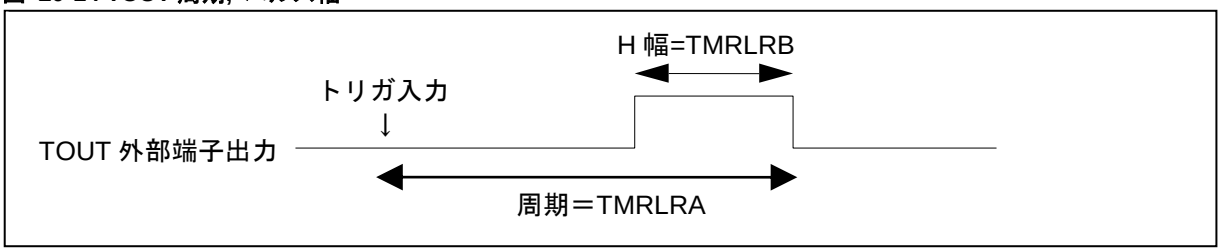
20.5.3.5 コンペアワンショット動作

コンペアワンショット動作について説明します。

TMCSR レジスタの bit15, bit14:MOD[1:0] =10, bit4:RELD =0 のとき、ダウンカウントごとにカウンタ値(TMR)とTMRLRB レジスタの値を比較するコンペアワンショット動作を行います。トリガを受け付け後、TMRLRA レジスタの値のロードを行い、ダウンカウントを始めます。比較一致(TMR = TMRLRB) からダウンカウントすると、TOUT 出力を反転させます。アンダフローが発生すると、カウント動作を停止・TOUT 出力を初期化し、起動トリガ待ち状態になります。

TMRLRA の値はタイマの起動から停止するまでの時間、TMRLRB の値は、TOUT 出力の H 幅が出力され始めるカウンタ値を示します。OUTL="0"のとき、TMR < TMRLRB のときに TOUT 出力は"H レベル"となります。

図 20-14 TOUT 周期, パルス幅



ダウンカウント開始時から TMR = TMRLRB となるまで(TMR ≧ TMRLRB のとき)は以下の動作を行います。

- TOUT 出力は初期値継続
- タイマはカウント継続

  TMR = TMRLRB からダウンカウントが発生した時は以下の動作を行います。
- TOUT 出力を反転
- タイマはカウント継続

  (インターバルタイマモードでのコンペア動作時は、TMCSR レジスタの bit7:EF ビットはセットされません。)

アンダフローが発生した時は以下の動作を行います。

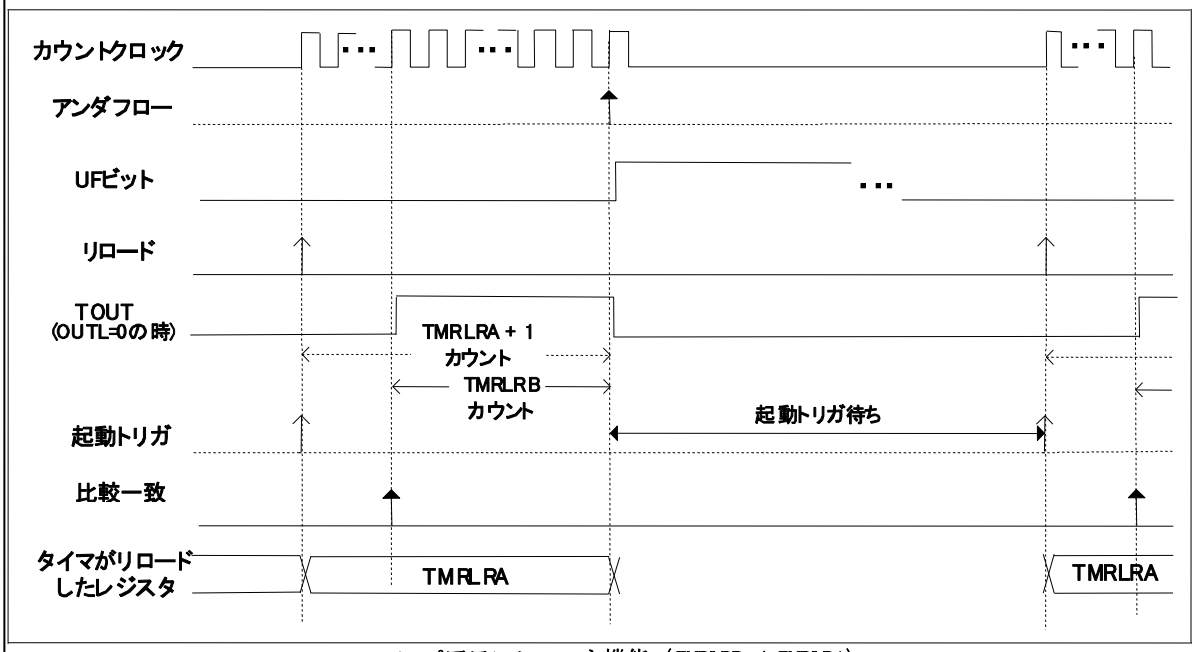
- TMCSR レジスタの bit2:UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- TOUT 出力を初期化
- タイマは 0xFFFF で停止
- タイマは起動トリガ待ち状態

コンペア機能の動作は TMRLRA と TMRLRB の設定関係により変わります。

図 20-15 コンペアワンショット動作 (1 / 2)

・TMRLRB < TMRLRA　を設定

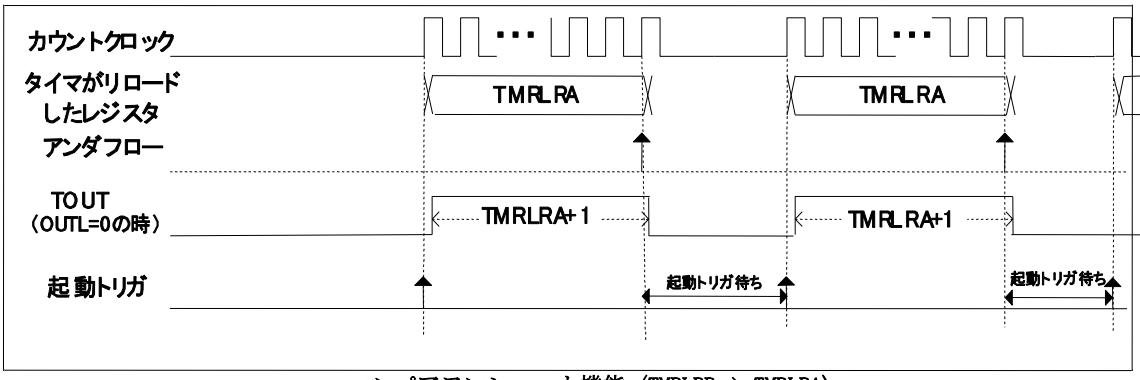
上記レジスタ関係の時、タイマへのロードを行ってから TMR = TMRLRB になるまで、TOUT 出力は、‘L レベル’となります。比較一致(TMR = TMRLRB) からダウンカウントすると、TOUT 出力を反転させて、アンダフローが発生するまで‘H レベル’となります。アンダフローが発生すると、TOUT 出力は初期化されます。アンダフローによりタイマはカウント動作を停止して起動トリガ待ち状態になります。(OUTL=‘0’の時)



コンペアワンショット機能（TMRLRB < TMRLRA)

・TMRLRB > TMRLRA を設定

上記レジスタの関係の時、タイマへのロードを行うと、すでに TMR < TMRLRB となっているので、TOUT 出力は、起動トリガが発生してからアンダフローが発生するまで‘H レベル’となります。アンダフローの発生によりタイマは起動トリガ待ち状態となり、TOUT 出力は‘L レベル’となります。(OUTL=‘0’の時)

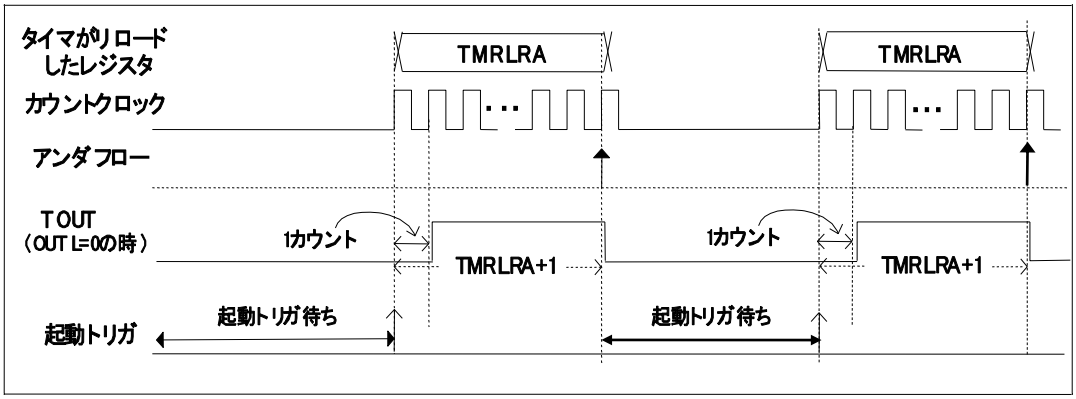


コンペアワンショット機能（TMRLRB > TMRLRA)

図 20-16 コンペアワンショット動作 (2 / 2)

・TMRLRB = TMRLRA を設定

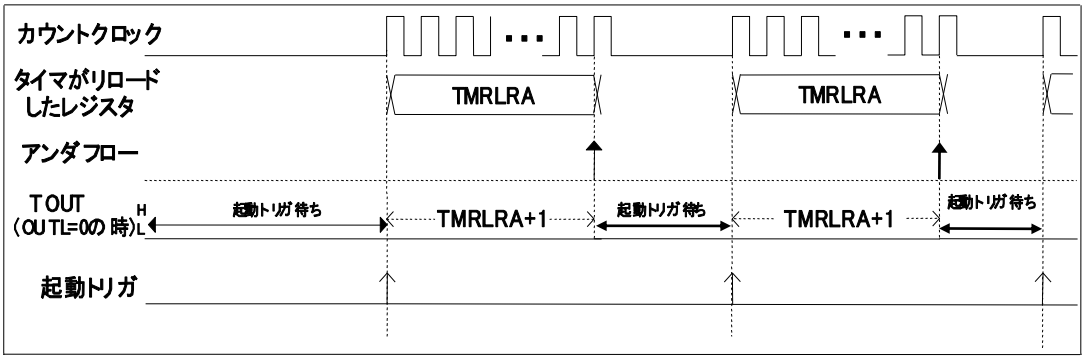
上記レジスタの関係の時、タイマへのロード後、1 カウントするとTMR < TMRLRB となるので、TOUT 出力は、ダウンカウント 1 カウント分"L レベル"を出力し、アンダフローが発生するまで"H レベル"を出力します。アンダフローの発生によりタイマは起動トリガ待ち状態となり、TOUT 出力は "L レベル"となります。(OUTL = "0"の時)



コンペアワンショット機能 (TMRLRB = TMRLRA)

・TMRLRB = 0 を設定

上記レジスタの関係の時、TMR < TMRLRB となることは無いので、ダウンカウント開始時からアンダフローが発生するまで TOUT 出力は "L レベル"となります。アンダフロー発生後も"L レベル" のままです。(OUTL = "0"の時)



コンペアワンショット機能 (TMRLRB = 0)

20.5.3.6 コンペアリロード動作

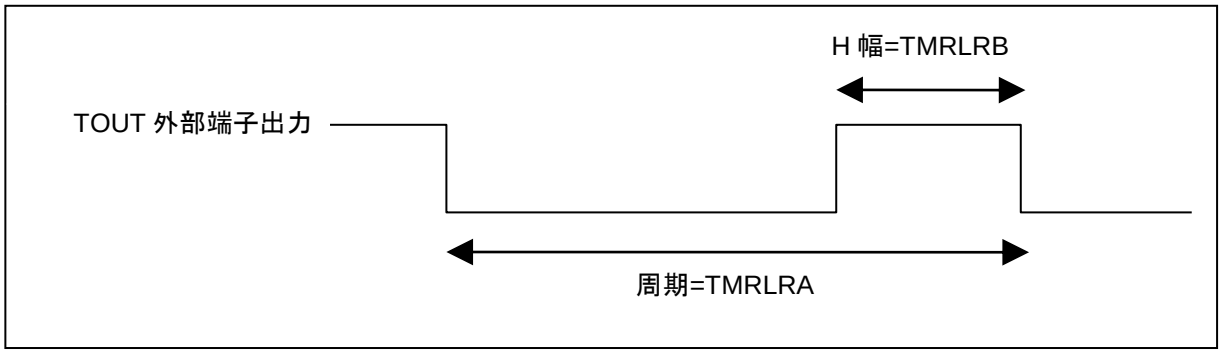
コンペアリロード動作について説明します。

TMCSR レジスタの bit15, bit14:MOD[1:0] =10, bit4: RELD =1 のとき、タイマはダウンカウントごとにカウンタ値(TMR)と TMRLRB の値を比較し、比較一致(TMR = TMRLRB)からダウンカウントすると TOUT 出力を反転します。アンダフローが発生すると、再び TMRLRA から値をロードしダウンカウント動作を行うコンペアリロード動作を行います。カウンタへのロードは TMRLRA から行います。

TMRLRA の値はタイマが起動してからリロードするまでのカウンタ周期、TMRLRB の値は TOUT 出力が"L レベル出力"から"H レベル出力"へ反転した後の"H レベル幅"を示します。

TMR + 1 = TMRLRB のとき、TOUT 出力は反転し、"H レベル"となります(OUTL=0 のとき)。

図 20-17 TOUT 周期、パルス幅



ダウンカウント開始時から TMR = TMRLRB となるまで(TMR ≧ TMRLRB のとき)は以下の動作を行います。

- TOUT 出力は初期値継続
- カウント継続

TMR = TMRLRB からダウンカウントとなった時は以下の動作を行います。

- TOUT 出力を反転
- カウント継続
  (インターバルタイマモードでのコンペア動作時は、TMCSR レジスタの bit7:EF ビットはセットされません。)

アンダフローが発生した時は以下の動作を行います。

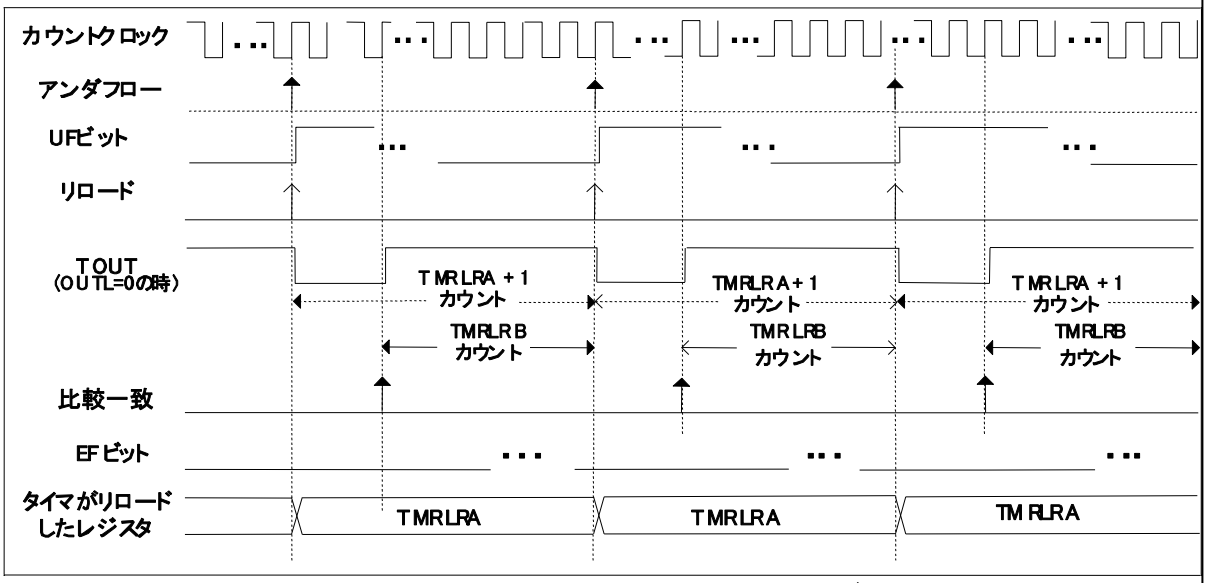
- TMCSR レジスタの bit2:UF ビットをセット
- 割込み許可(TMCSR レジスタの bit3:INTE=1)のとき、割込み発生
- TOUT 出力を初期化
- TMRLRA から値をリロード
- タイマはカウント継続

コンペア機能の動作は TMRLRA と TMRLRB の設定関係により変わります。

図 20-18 コンペアリロード動作 (1 / 2)

・TMRLRB < TMRLRA を設定

上記レジスタ関係の時、タイマへのロードを行ってから TMR = TMRLRB の関係になるまで、TOUT 出力は、‘L レベル’ となります。比較一致 (TMR = TMRLRB) からダウンカウントした時、TOUT 出力を反転させ、アンダフローが発生するまで‘H レベル’となります。アンダフローが発生すると、TOUT 出力は初期化されます。アンダフローにより、タイマは TMRLRA からリロードを行い、カウント動作を継続します。(OUTL=‘0’ の時)

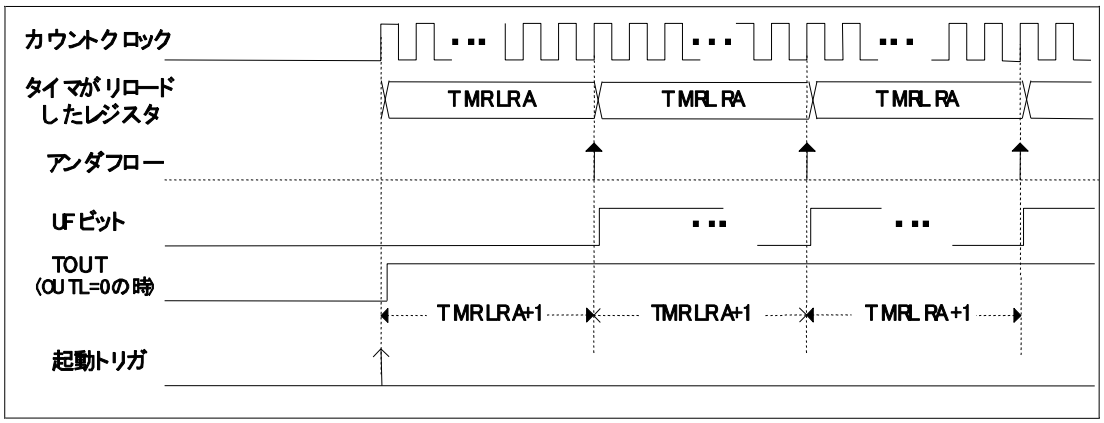


コンペアリロード機能 (TMRLRB < TMRLRA) トリガ入力

・TMRLRB > TMRLRA を設定

上記レジスタ関係の時、常に TMR < TMRLRB なので、TOUT 出力は、起動トリガが発生してからアンダフローが発生するまで‘H レベル’となります。アンダフローが発生しても TOUT 出力は‘H レベル’を継続します。また、アンダフローによりタイマは TMRLRA からロードを行い、カウント動作を継続します。

(OUTL=‘0’ の時)



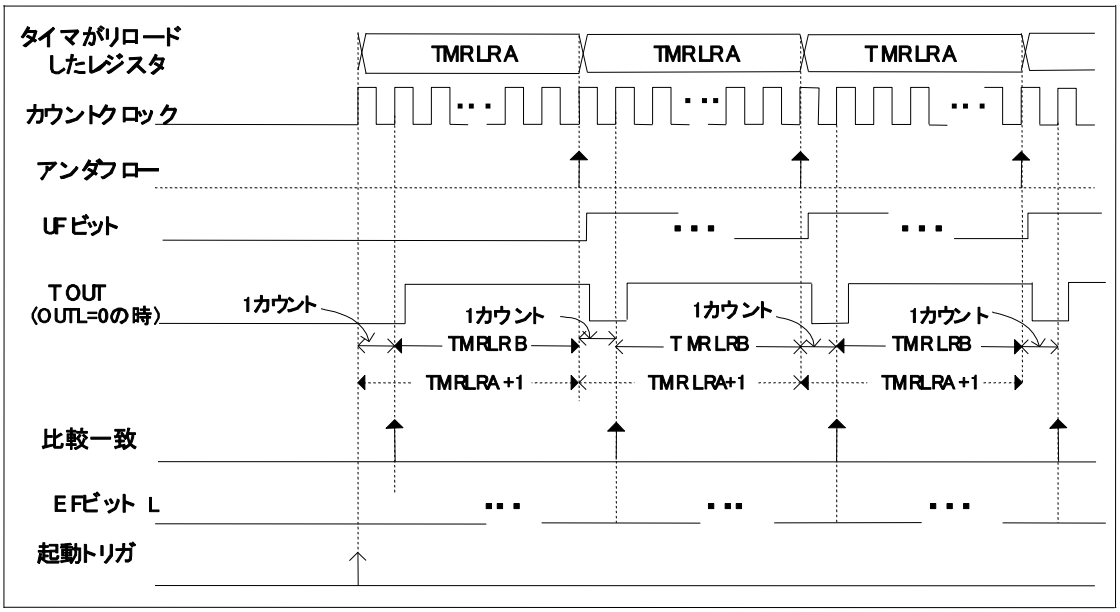
コンペアリロード機能 (TMRLRB > TMRLRA) トリガ入力

図 20-19 コンペアリロード動作 (2 / 2)

・TMRLRB = TMRLRA を設定

上記レジスタ関係の時、タイマへのロード後、1カウントするとTMR＜TMRLRBとなるので、TOUT出力は、ダウンカウント1カウント分‘Lレベル’を出力し、アンダフローが発生するまで‘Hレベル’を出力します。

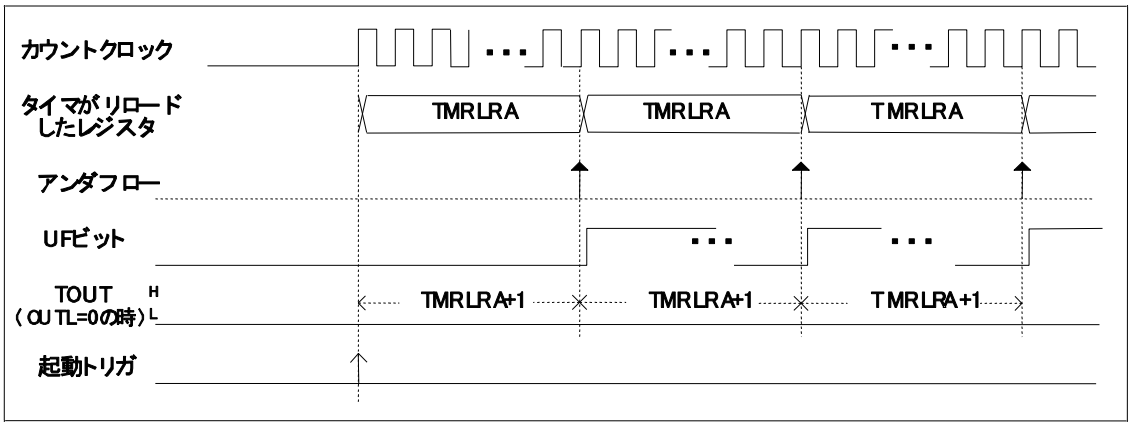
アンダフローの発生によりタイマはTMRLRAからロードを行い、カウント動作を継続します。TOUT出力は‘Lレベル’となります。(OUTL=‘0’の時)



コンペアリロード機能 (TMRLRB = TMRLRA) トリガ入力

・TMRLRB = 0 を設定

上記レジスタ関係の時、タイマへのロード後、TMR＜TMRLRBとなることは無いので、ダウンカウント開始時からアンダフローが発生するまでTOUT出力は‘Lレベル’となります。アンダフローが発生後も‘Lレベル’のままです。



コンペアリロード機能 (TMRLRB = ‘0’) トリガ入力

20.5.3.7 キャプチャモード

キャプチャモードについて説明します。

TMCSR レジスタの bit15, bit14:MOD[1:0] =11 のとき、タイマはキャプチャ動作を行います。リトリガが発生した時、その時点の TMR の値を TMRLRB レジスタにキャプチャし、TMCSR レジスタの bit7:EF をセットします。

TTRG 入力をゲート入力として使用(TMCSR レジスタの bit8:GATE=1 のとき)しているときは、TMCSR レジスタの bit0:TRG によってリトリガを発生させてください。

キャプチャモード以外では、リトリガ発生時にキャプチャは行いません。EF 割込みも発生しません。

タイマの動作/TOUT 出力はシングルワンショット機能・シングルリロード機能どちらも同じです。

＜注意事項＞
ワンショットモードではリトリガ時、TOUT は初期化されません。

図 20-20 キャプチャ動作

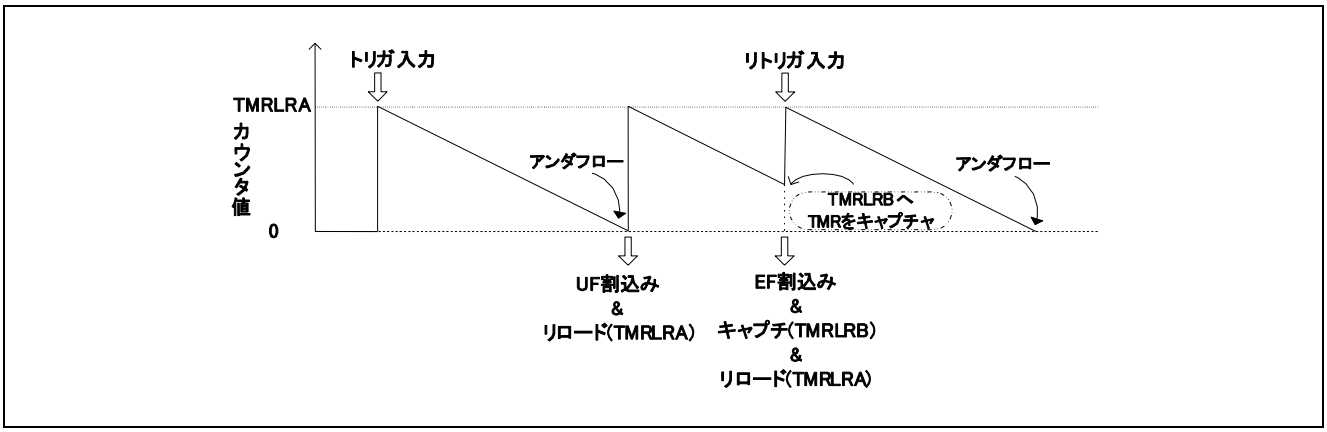


図 20-21 インターバルタイマモード時におけるトリガ入力機能のフローチャート

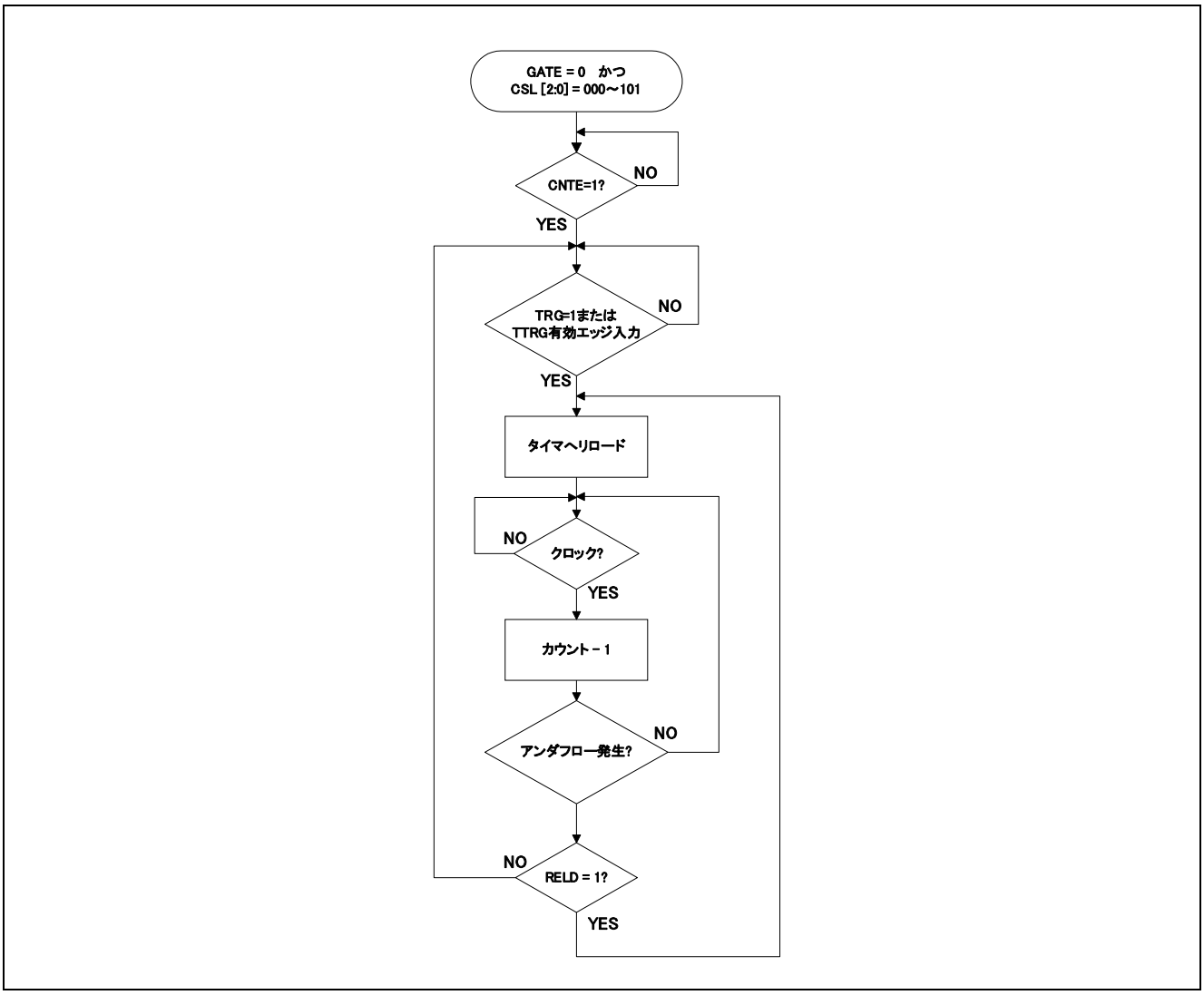
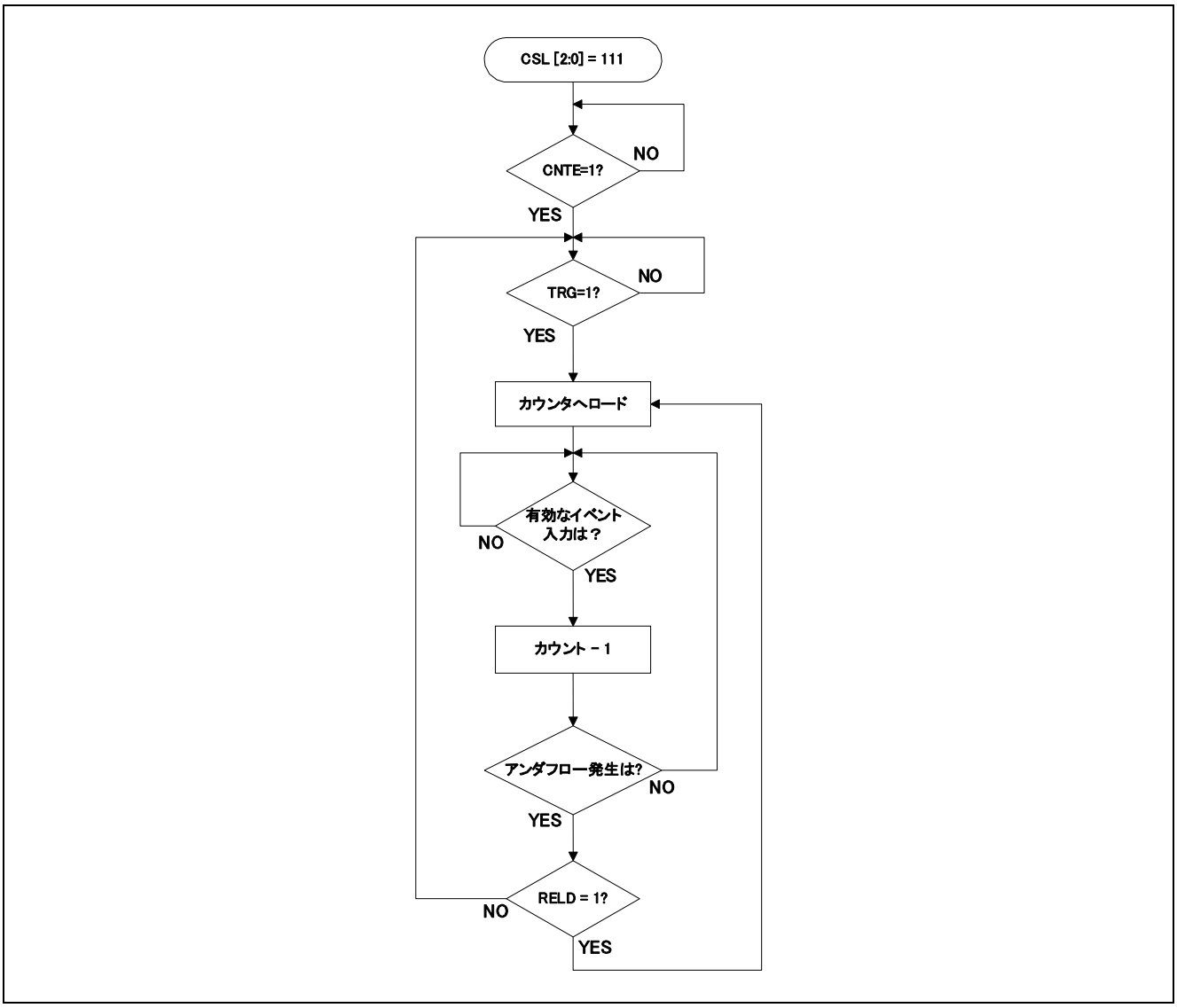


図 20-22 イベントカウンタモード時におけるフローチャート



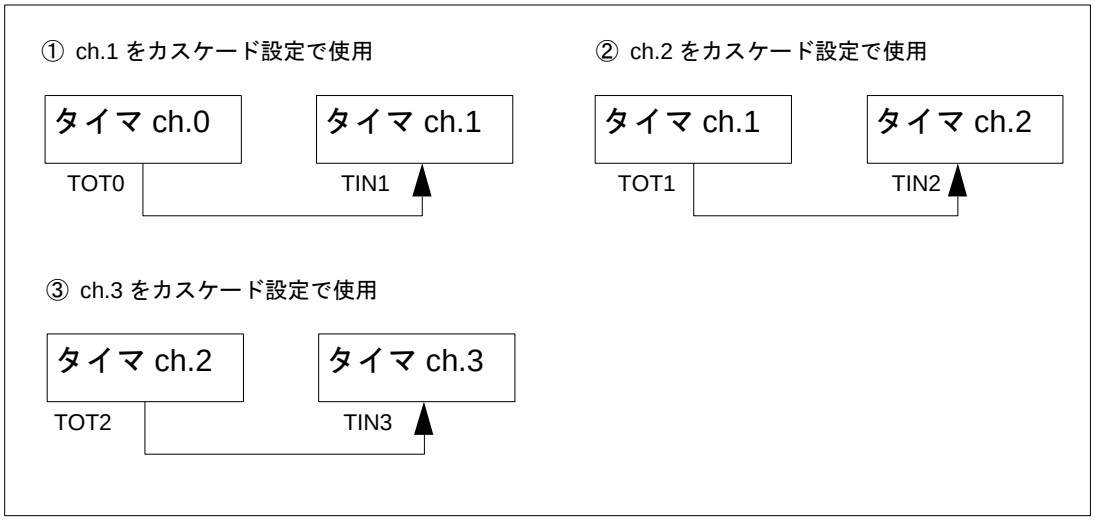
20.5.4　カスケード入力

カスケード入力について説明します。

カスケード入力(TMCSR レジスタの bit11～bit9:CSL[2:0]= 110)を選択すると、タイマの ch.0 の出力(TOUT0)を ch.1 の入力(TTRG1)として、ch.1 の出力(TOUT1)を ch.2 の入力(TTRG2)として、ch.2 の出力(TOUT2)を ch.3 の入力(TTRG3)として使用することができます。

ch.4～ch.6 も同様です。

図 20-23 カスケード入力設定時のタイマの入出力



20.5.5　同時動作の優先順位

同時動作の優先順位について説明します。

タイマの動作を決定するイベントが同時に発生したときに、動作状態を決定する優先順位を示します。

12. レジスタ書込み
11. トリガ入力
12. アンダフロー
13. クロック入力

タイマの動作による各フラグのセットと、レジスタ書込みによるフラグのクリアが同時に発生した時の動作を決定する優先順位を示します。

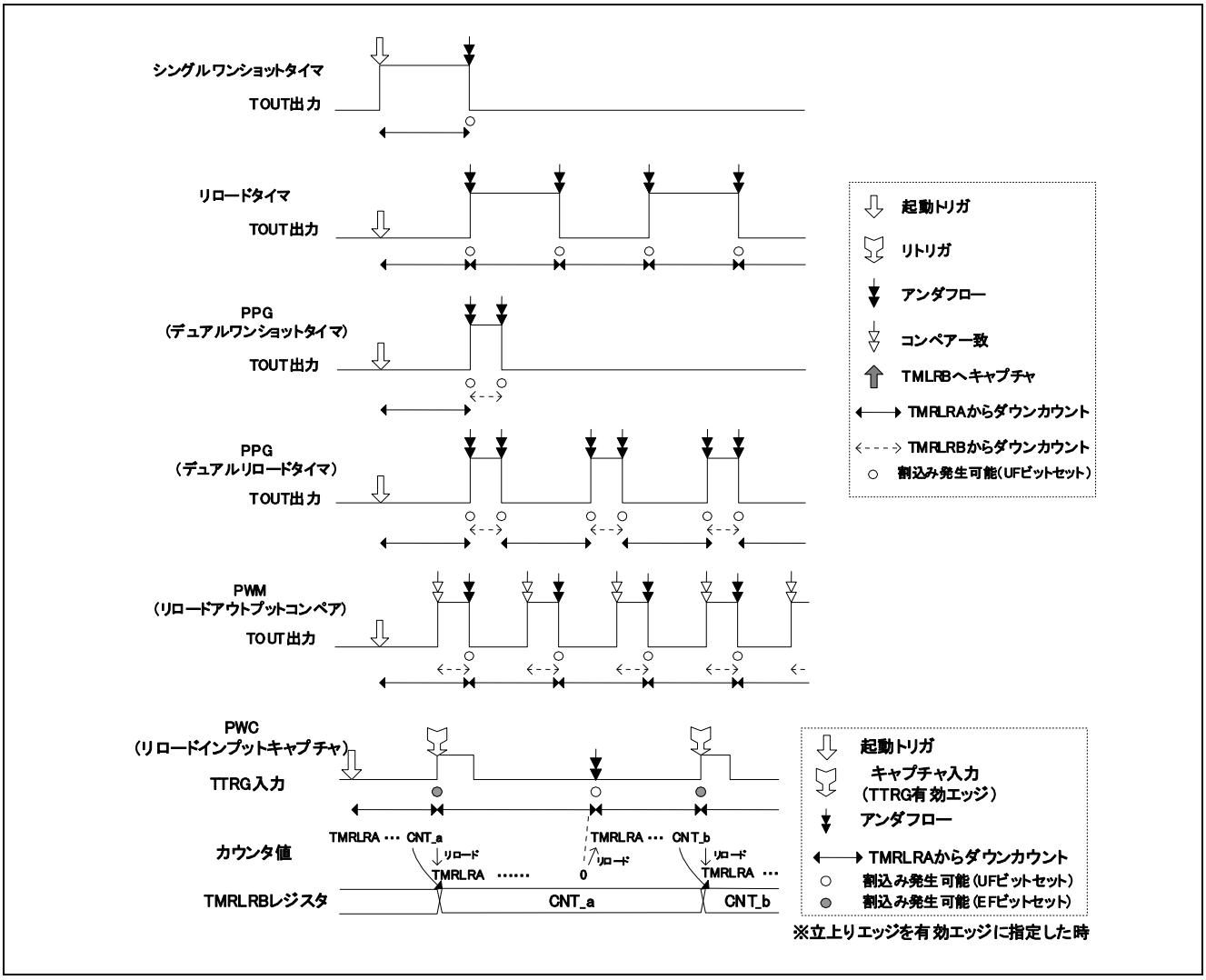
13. タイマ動作によるフラグのセット
14. UF ビット/EF ビットへのフラグクリアのためのレジスタ書込み

20.6 アプリケーションノート

アプリケーションノートについて示します。

本タイマで実現できる代表的な機能を示します。

図 20-24 使用例



上図の例を使用するための設定を示します。

表 20-7 設定例

機能	MOD[1:0]	RELD	TMRLRA	TMRLRB
シングルワンショットタイマ	00 (シングルモード)	0	必須	−
リロードタイマ	00 (シングルモード)	1	必須	−
PPG (Programmable Pulse Generator)	01 (デュアルモード)	0 または 1	必須	必須
PWM (Pulse Width Modulator)	10 (コンペアモード)	1	必須	必須
PWC (Pulse Width Counter)	11 (キャプチャモード)	1	必須	−

20.6.1 シングルワンショットタイマ

シングルワンショットタイマについて説明します。

シングルワンショットタイマは、トリガが入力されると、TMRLRA レジスタからカウンタへロードを行い、ダウンカウント動作を行います。アンダフローが発生するとカウント動作を停止します。

TOUT 端子は、カウント中は"H レベル"を出力し、アンダフローが発生すると"L レベル"を出力します。(OUTL=0 のとき)

【設定】シングルワンショットタイマとして使用するためには、以下の設定を行ってください。

1. TTRG 入力を使用しない時

TMCSR											TMRLRA
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	カウント初期値設定
00	00	*1	0	－	*2	0	*3	－	1	S	

S ：タイマ起動時に使用
－ ：動作に影響なし
*1 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*2 : TOUT 出力極性設定
OUTL=0・・初期値 L=>カウント開始 H=>アンダフロー発生 L
OUTL=1・・初期値 H=>カウント開始 L=>アンダフロー発生 H
*3 : 割込み要求許可設定
INTE=0・・割込み禁止
INTE=1・・割込み許可

2. TTRG 入力をゲート入力として使用する時

TMCSR											TMRLRA
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	カウント初期値設定
00	*1	*2	1	－	*3	0	*4	－	1	S	

S ：タイマ起動時に使用
－ ：動作に影響なし
*1 : TTRG の有効レベル設定
TRGM[1:0]=x0・・L 入力期間のみカウント
TRGM[1:0]=x1・・H 入力期間のみカウント
*2 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周

*3 : TOUT 出力極性設定
　　OUTL= 0・・初期値 L=>カウント開始 H=>アンダフロー発生 L
　　OUTL= 1・・初期値 H=>カウント開始 L=>アンダフロー発生 H
*4 : 割込み要求許可設定
　　INTE= 0・・割込み禁止
　　INTE= 1・・割込み許可

3.　TTRG 入力をトリガ入力として使用する時

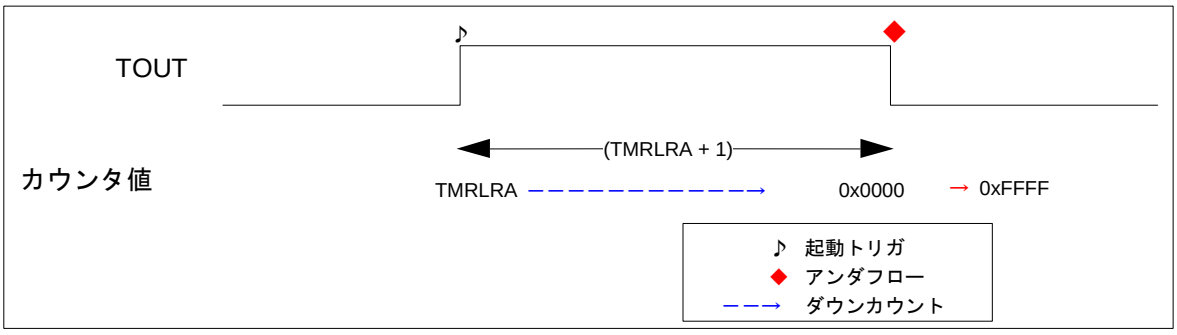
TMCSR											TMRLRA
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	カウント初期値設定
00	*1	*2	0	－	*3	0	*4	－	1	S	

S　：タイマ起動時に使用
－ ：動作に影響なし
*1 : TTRG の有効レベル設定
　　TRGM[1:0]= 00・・外部トリガエッジ検出しない
　　TRGM[1:0]= 01・・立上りエッジ
　　TRGM[1:0]= 10・・立下りエッジ
　　TRGM[1:0]= 11・・両エッジ
*2 : カウントクロック分周設定
　　CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
　　CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
　　CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
　　CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
　　CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*3 : OUT 出力極性設定
　　OUTL= 0・・初期値 L=>カウント開始 H=>アンダフロー発生 L
　　OUTL= 1・・初期値 H=>カウント開始 L=>アンダフロー発生 H
*4 : 割込み要求許可設定
　　INTE= 0・・割込み禁止
　　INTE= 1・・割込み許可

【タイマ起動】 以下の手順でタイマの起動を行ってください。

- 起動トリガを入力(TRG ビットへの"1"書込みまたは TTRG 端子からの有効外部エッジ入力)
- TTRG 端子入力をゲート入力として使用する際は、有効レベルを入力

図 20-25 動作例 (OUTL = 0)



20.6.2　リロードタイマ

リロードタイマについて説明します。

リロードタイマは、アンダフローが発生するたびに、TMRLRA レジスタからカウンタへロードを行い、ダウンカウント動作を繰り返します。TOUT 出力は、起動トリガから最初のアンダフローまでのカウント中は"L レベル"、アンダフローが発生するたびに出力は反転し、最初のアンダフローの発生によって"H レベル"を出力します。また、リトリガが発生すると、TOUT 出力は初期値になります。(OUTL=0 のとき)

【設定】リロードタイマとして使用するためには、以下の設定を行ってください。

1.　TTRG 入力を使用しない時

TMCSR											TMRLRA
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	カウント初期値設定
00	00	*1	0	－	*2	1	*3	－	1	S	

S　：タイマ起動時に使用
－ ：動作に影響なし
*1 : カウントクロック分周設定
　　CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
　　CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
　　CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
　　CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
　　CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*2: TOUT 出力極性設定
　　OUTL=0・・初期値 L=>カウント開始 L=>アンダフロー発生のたびに反転
　　OUTL=1・・初期値 H=>カウント開始 H=>アンダフロー発生のたびに反転
*3 : 割込み要求許可設定
　　INTE=0・・割込み禁止
　　INTE=1・・割込み許可

2.　TTRG 入力をゲート入力として使用する時

TMCSR											TMRLRA
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	カウント初期値設定
00	*1	*2	1	－	*3	1	*4	－	1	S	

S　：タイマ起動時に使用
－ ：動作に影響なし
*1 : TTRG の有効レベル設定
　　TRGM[1:0]=x0・・TTRG=L 入力期間のみカウント
　　TRGM[1:0]=x1・・TTRG=H 入力期間のみカウント
*2 : カウントクロック分周設定
　　CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
　　CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
　　CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
　　CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
　　CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*3 : OUT 出力極性設定
　　OUTL=0・・初期値 L=>カウント開始 L=>アンダフロー発生のたびに反転
　　OUTL=1・・初期値 H=>カウント開始 H=>アンダフロー発生のたびに反転

*4 : 割込み要求許可設定
INTE=0・・割込み禁止
INTE=1・・割込み許可

3. TTRG 入力をトリガ入力として使用する時

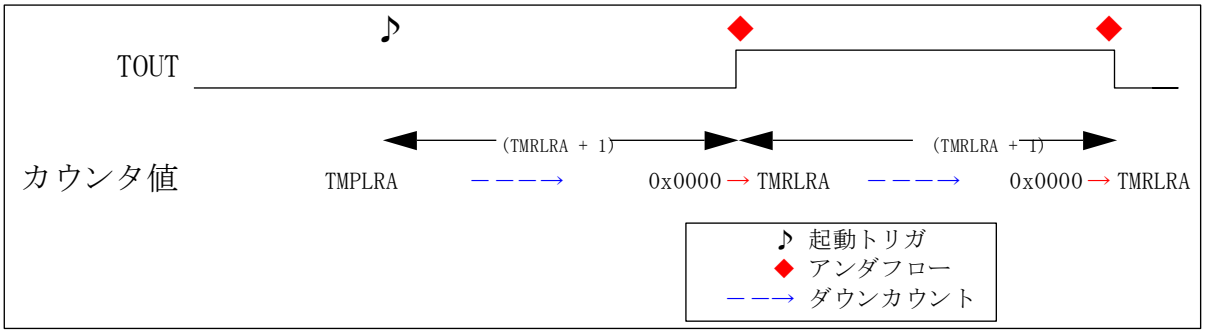
TMCSR											TMRLRA
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	カウント初期値設定
00	*1	*2	0	－	*3	1	*4	－	1	S	

S ： タイマ起動時に使用
－ ： 動作に影響なし
*1 : TTRG の有効エッジ設定
TRGM[1:0]= 00・・外部トリガエッジ検出しない
TRGM[1:0]= 01・・立上りエッジ
TRGM[1:0]= 10・・立下りエッジ
TRGM[1:0]= 11・・両エッジ
*2 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*3 : OUT 出力極性設定
OUTL= 0・・初期値 L=>カウント開始 L=>アンダフロー発生のたびに反転
OUTL= 1・・初期値 H=>カウント開始 H=>アンダフロー発生のたびに反転
*4 : 割込み要求許可設定
INTE= 0・・割込み禁止
INTE= 1・・割込み許可

【タイマ起動】以下の手順でタイマ起動を行ってください。

- 起動トリガを入力(TRG ビットへの"1"書込みまたは TTRG 端子からの有効外部エッジ入力)
- TTRG 端子入力をゲート入力として使用する際は、有効レベルを入力

図 20-26 動作例 (OUTL=0)



20.6.3 PPG

PPG について説明します。

PPG は、パルスの L 幅/H 幅を設定して出力パルスを生成する機能です。 起動トリガによって TMRLRA からカウンタへロードしダウンカウントを行い、アンダフローが発生すると TMRLRB から値をロードしダウンカウントを実行します。

RELD=0 のときは起動トリガ=>TMRLRA ロード=>ダウンカウント=>アンダフロー=>TMRLRB
ロード=>ダウンカウント=>アンダフローでダウンカウントを停止します。

RELD=1 のときは起動トリガ=>TMRLRA ロード=>ダウンカウント=>アンダフロー=>TMRLRB
ロード=>ダウンカウント=>アンダフロー=> TMRLRA ロード=>ダウンカウント=>アンダフロー=>TMRLRB ロード というように、アンダフローが発生するたびに TMRLRA/TMRLRB を交互に
ロードを行い、ダウンカウントを実行します。

TOUT 出力は､TMRLRA からのダウンカウントによるアンダフローが発生するまでのカウント中は"L レベル"、TMRLRB からのダウンカウントによるアンダフローが発生するまでは、"H レベル"を出力します。また、リトリガが発生すると、TOUT 出力は初期値になります。

＜注意事項＞
ワンショットモードではリトリガ時、TOUT は初期化されません。

【設定】PPG として使用するためには、以下の設定を行ってください。

1. TTRG 入力を使用しない時

TMCSR											TMRLRA	TMRLRB
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	(A)	(B)
01	00	*1	0	－	*2	*3	*4	－	1	S		

(A):起動トリガ発生時カウント初期値/TMRLRB 値からのカウントによるアンダフロー時のリロード値(RELD=1 のとき)
(B):TMRLRA 値からのカウントによるアンダフロー時のリロード値
S ：タイマ起動時に使用
－ ：動作に影響なし
* 1 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
* 2 : TOUT 出力極性設定
OUTL= 0・・
初期値 L=>TMRLRA からカウント L=> アンダフロー発生で H=>
TMRLRB からカウント H=> アンダフロー発生で L
OUTL= 1・・
初期値 H=>TMRLRA からカウント H=> アンダフロー発生で L=>
TMRLRB からカウント L=> アンダフロー発生で H

*3 : アンダフロー発生時リロード設定
RELD= 0・・ワンショットモード
RELD= 1・・リロードモード
*4 : 割込み要求許可設定
INTE= 0・・割込み禁止
INTE= 1・・割込み許可

2. TTRG 入力をゲート入力として使用する時

TMCSR											TMRLRA	TMRLRB
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	(A)	(B)
01	*1	*2	1	ー	*3	*4	*5	ー	1	S		

(A):起動トリガ発生時カウント初期値/TMRLRB 値からのカウントによるアンダフロー時のリロード値(RELD=1 のとき)
(B):TMRLRA 値からのカウントによるアンダフロー時のリロード値
S : タイマ起動時に使用
ー : 動作に影響なし
* 1 : TTRG の有効レベル設定
TRGM[1:0]= x0・・TTRG=L 入力期間のみカウント
TRGM[1:0]= x1・・TTRG=H 入力期間のみカウント
* 2 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
* 3 : TOUT 出力極性設定
OUTL= 0・・
初期値 L=>TMRLRA からカウント L=> アンダフロー発生で H=>
TMRLRB からカウント H=> アンダフロー発生で L
OUTL= 1・・
初期値 H=>TMRLRA からカウント H=> アンダフロー発生で L=>
TMRLRB からカウント L=> アンダフロー発生で H
*4 : アンダフロー発生時リロード設定
RELD=0・・ワンショットモード
RELD=1・・リロードモード
*5 : 割込み要求許可設定
INTE=0・・割込み禁止
INTE=1・・割込み許可

3. TTRG 入力をトリガ入力として使用する時

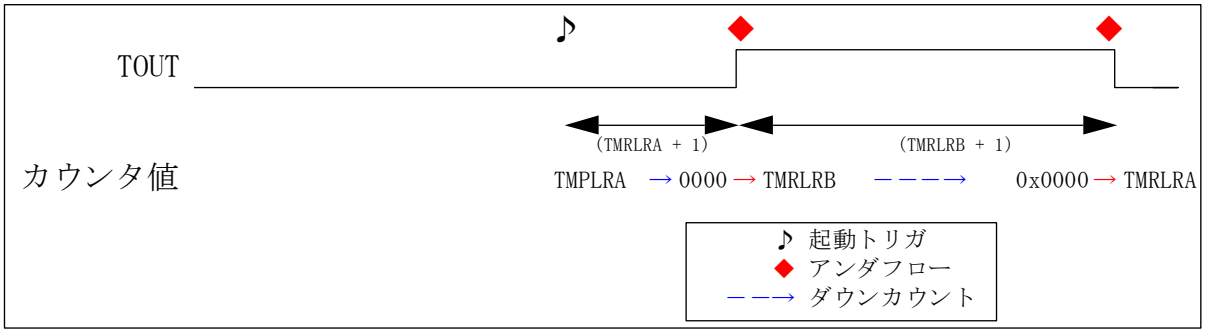
TMCSR											TMRLRA	TMRLRB
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	(A)	(B)
01	*1	*2	0	－	*3	*4	*5	－	1	S		

(A):起動トリガ発生時カウント初期値/TMRLRB 値からのカウントによるアンダフロー時のリロード値(RELD=1 のとき)
(B):TMRLRA 値からのカウントによるアンダフロー時のリロード値
S ：タイマ起動時に使用
－ ：動作に影響なし
*1 : TTRG の有効エッジ設定
TRGM[1:0]= 00・・外部トリガエッジ検出しない
TRGM[1:0]= 01・・立上りエッジ
TRGM[1:0]= 10・・立下りエッジ
TRGM[1:0]= 11・・両エッジ
*2 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*3 : TOUT 出力極性設定
OUTL= 0・・初期値 L=>TMRLRA からカウント L=>アンダフロー発生のたびに反転
OUTL= 1・・初期値 H=>TMRLRA からカウント H=>アンダフロー発生のたびに反転
*4 : アンダフロー発生時リロード設定
RELD=0・・ワンショットモード
RELD=1・・リロードモード
*5 : 割込み要求許可設定
INTE=0・・割込み禁止
INTE=1・・割込み許可

【タイマ起動】以下の手順でタイマ起動を行ってください。

- 起動トリガを入力(TRG ビットへの"1"書込みまたは TTRG 端子からの有効外部エッジ入力)
- TTRG 端子入力をゲート入力として使用する際は、有効レベルを入力

図 20-27 動作例 (OUTL=0)



20.6.4 PWM

PWM について説明します。

PWM は、パルスの周期と H 幅を設定して出力パルスを生成する機能です。

起動トリガによって TMRLRA からカウンタへロードしダウンカウント動作を行います。

TOUT 出力は、起動トリガ後は"L レベル"を出力し、TMRLRB の設定値よりもカウンタ値が小さくなると"H レベル"を出力します。アンダフローが発生すると、TOUT 出力は初期値に戻ります。(OUTL=0 のとき)

RELD=0 のときは、起動トリガ=>TMRLRA ロード=>ダウンカウント=>アンダフローでダウンカウントを停止します。

RELD=1 のときは、起動トリガ=>TMRLRA ロード=>ダウンカウント=>アンダフロー=>TMRLRA ロード=>ダウンカウント・・というように、アンダフローが発生するたびに TMRLRA のロードを行い、ダウンカウントを実行します。

【設定】PWM として使用するためには、以下の設定を行ってください。

1. TTRG 入力を使用しない時

TMCSR											TMRLRA	TMRLRB
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	(A)	(B)
10	0	*1	0	－	*2	*3	*4	－	1	S		

(A):起動トリガ発生時のカウント初期値/アンダフロー発生時のリロード値(RELD=1 のとき)
(B):カウンタ値と比較する値を設定(TMRLRB < TMRLRA) *5
S ：タイマ起動時に使用
－ ：動作に影響なし
*1 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*2 : TOUT 出力極性設定
OUTL= 0・・初期値 L=>TMRLRA からカウント L=>TMRLRB よりカウンタ値が小さい H
OUTL= 1・・初期値 H=>TMRLRA からカウント H=>TMRLRB よりカウンタ値が小さい L
*3 : アンダフロー発生時リロード設定
RELD= 0・・ワンショットモード
RELD= 1・・リロードモード
*4 : 割込み要求許可設定
INTE= 0・・割込み禁止
INTE= 1・・割込み許可
*5 : L クリップ出力で TOUT 出力を使用するには、TMRLRB =0 に設定
H クリップ出力で TOUT 出力を使用するには、TMRLRB ="TMRLRA + 1"に設定

2. TTRG 入力をゲート入力として使用する時

TMCSR											TMRLRA	TMRLRB
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	(A)	(B)
10	*1	*2	1	－	*3	*4	*5	－	1	S		

(A):起動トリガ発生時のカウント初期値/アンダフロー発生時のリロード値(RELD=1 のとき)
(B):カウンタ値と比較する値を設定(TMRLRB < TMRLRA) *6
S ： タイマ起動時に使用
－ ：動作に影響なし
*1 : TTRG の有効レベル設定
TRGM[1:0]= x0・・TRGM=L 入力期間のみカウント
TRGM[1:0]= x1・・TRGM=H 入力期間のみカウント
*2 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*3 : TOUT 出力極性設定
OUTL= 0・・初期値 L=>TMRLRA からカウント L=>TMRLRB よりカウンタ値が小さい H
OUTL= 1・・初期値 H=>TMRLRA からカウント H=>TMRLRB よりカウンタ値が小さい L
*4 : アンダフロー発生時リロード設定
RELD=0・・ワンショットモード
RELD=1・・リロードモード
*5 : 割込み要求許可設定
INTE=0・・割込み禁止
INTE=1・・割込み許可
*6 : L クリップ出力で TOUT 出力を使用するには、TMRLRB =0 に設定
H クリップ出力で TOUT 出力を使用するには、TMRLRB = "TMRLRA + 1"に設定

3. TTRG 入力をトリガ入力として使用する時

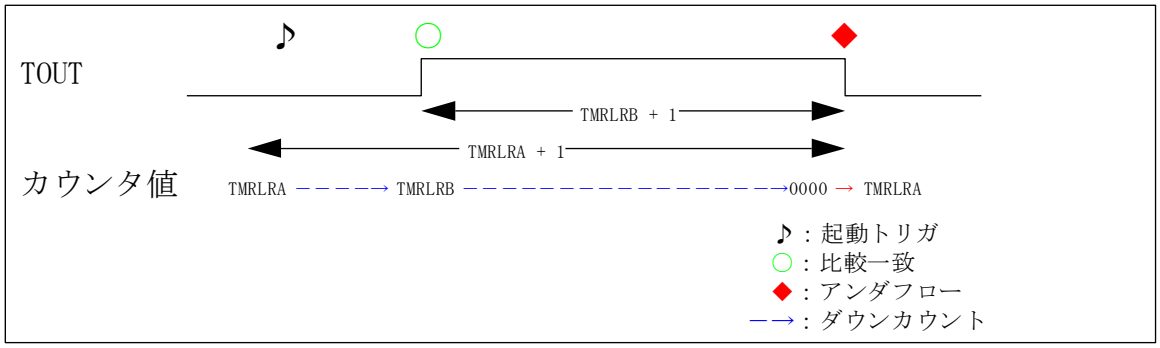
TMCSR											TMRLRA	TMRLRB
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	(A)	(B)
10	*1	*2	0	－	*3	*4	*5	－	1	S		

(A):起動トリガ発生時のカウント初期値/アンダフロー発生時のリロード値(RELD=1 のとき)
(B):カウンタ値と比較する値を設定(TMRLRB < TMRLRA) *6
S ：タイマ起動時に使用
－ ：動作に影響なし
*1 : TTRG の有効エッジ設定
TRGM[1:0]= 00・・外部トリガエッジ検出しない
TRGM[1:0]= 01・・立上りエッジ
TRGM[1:0]= 10・・立下りエッジ
TRGM[1:0]= 11・・両エッジ
*2 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*3 : TOUT 出力極性設定
OUTL= 0・・初期値 L=>TMRLRA からカウント L=>TMRLRB よりカウンタ値が小さい H
OUTL= 1・・初期値 H=>TMRLRA からカウント H=>TMRLRB よりカウンタ値が小さい L
*4 : アンダフロー発生時リロード設定
RELD= 0・・ワンショットモード
RELD= 1・・リロードモード
*5 : 割込み要求許可設定
INTE= 0・・割込み禁止
INTE= 1・・割込み許可
*6 : L クリップ出力で TOUT 出力を使用するには、TMRLRB =0 に設定
H クリップ出力で TOUT 出力を使用するには、TMRLRB ="TMRLRA + 1"に設定

【タイマ起動】以下の手順でタイマ起動を行ってください。

- 起動トリガを入力(TRG ビットへの"1"書込みまたは TTRG 端子からの有効外部エッジ入力)
- TTRG 端子入力をゲート入力として使用する際は、有効レベルを入力

図 20-28 動作例 (OUTL=0)



20.6.5 PWC

PWC について説明します。

PWC は、入力するトリガのトリガ間の時間を測定する機能です。

起動トリガの入力によって TMRLRA からカウンタへ値をロードしダウンカウント動作を開始します。カウント中にトリガ入力をすることにより、TMRLRB へそのときのカウンタ値をキャプチャするので、入力トリガ間の時間の測定ができます。

【設定】PWC として使用するためには、以下の設定を行ってください。

TMCSR											TMRLRA	TMRLRB
MOD [1:0]	TRGM [1:0]	CSL [2:0]	GATE	EF	OUTL	RELD	INTE	UF	CNTE	TRG	(A)	(B)
11	*1	*2	0	－	*3	*4	*5	－	1	S		

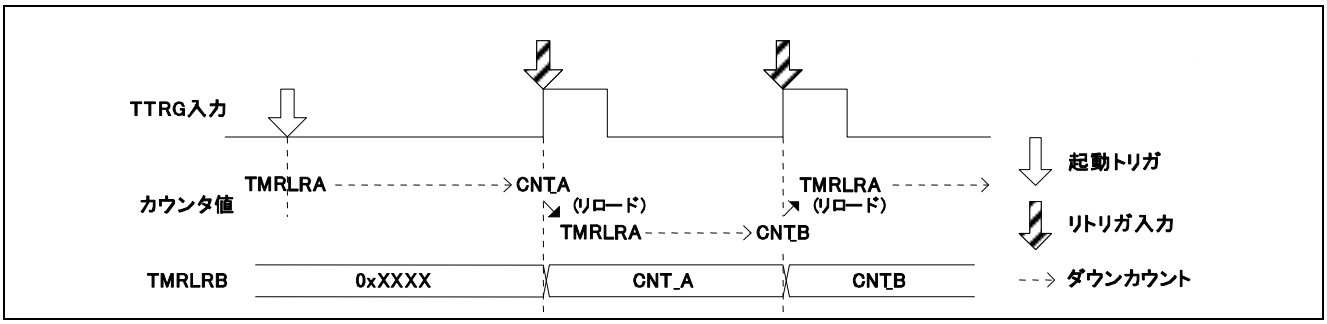
(A):起動トリガ発生時のカウント初期値/アンダフロー発生時のリロード値(RELD=1 のとき)
S ： タイマ起動時に使用
－ ： 動作に影響なし
*1 : TTRG の有効エッジ設定
TRGM[1:0]= 00・・外部トリガエッジ検出しない
TRGM[1:0]= 01・・立上りエッジ
TRGM[1:0]= 10・・立下りエッジ
TRGM[1:0]= 11・・両エッジ
*2 : カウントクロック分周設定
CSL[2:0]= 000・・周辺クロック(PCLK)の 2 分周
CSL[2:0]= 010・・周辺クロック(PCLK)の 8 分周
CSL[2:0]= 011・・周辺クロック(PCLK)の 16 分周
CSL[2:0]= 100・・周辺クロック(PCLK)の 32 分周
CSL[2:0]= 101・・周辺クロック(PCLK)の 64 分周
*3 : TOUT 出力極性設定
OUTL= 0・・初期値 L=>TMRLRA からカウント L=>アンダフロー発生のたびに反転
OUTL= 1・・初期値 H=>TMRLRA からカウント H=>アンダフロー発生のたびに反転
*4 : アンダフロー発生時リロード設定
RELD= 0・・ワンショットモード
RELD= 1・・リロードモード
*5 : 割込み要求許可設定
INTE= 0・・割込み禁止
INTE= 1・・割込み許可

【タイマ起動】以下の手順でタイマ起動を行ってください。

■ 起動トリガを入力(TRG ビットへの"1"書込みまたは TTRG 端子からの有効外部エッジ入力)
ダウンカウント中、トリガ入力があるたびにカウンタ値が TMRLRB へキャプチャされます。入力するトリガのエッジ間の時間は以下となります。

T = (TMRLRA の設定値 － TMRLRB のキャプチャ値) × 周辺クロック(PCLK)の周期 × CSL で設定した分周比

図 20-29 動作例(TRGM=01)



21. フリーランタイマ

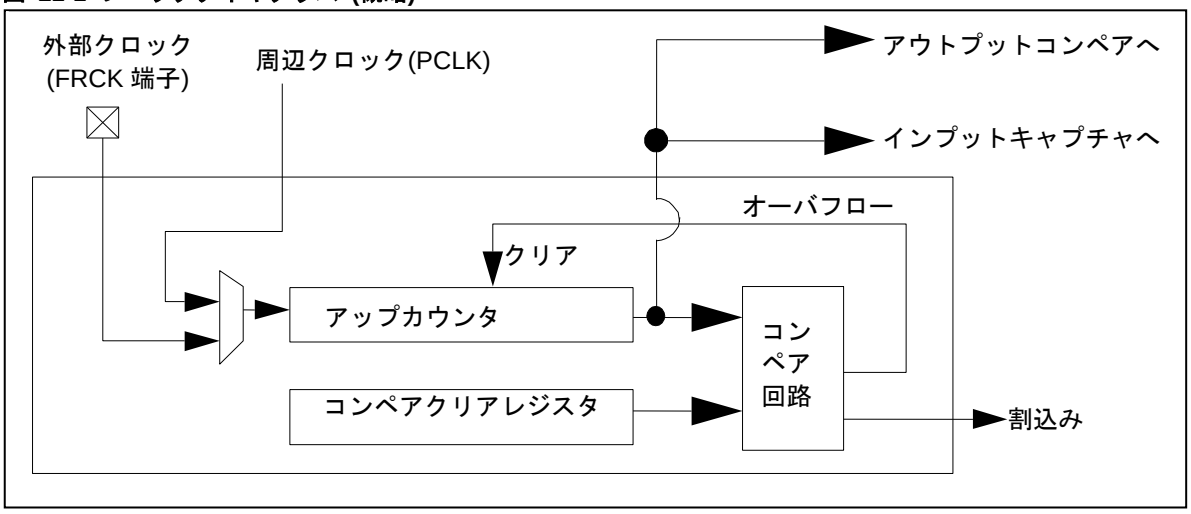
フリーランタイマについて説明します。

21.1 概要

フリーランタイマの概要について説明します。

フリーランタイマは、32 ビットアップカウンタ，制御回路で構成しています。フリーランタイマはインプットキャプチャ，アウトプットコンペアと組み合わせて使用できます。

図 21-1 ブロックダイヤグラム (概略)



21.2 特長

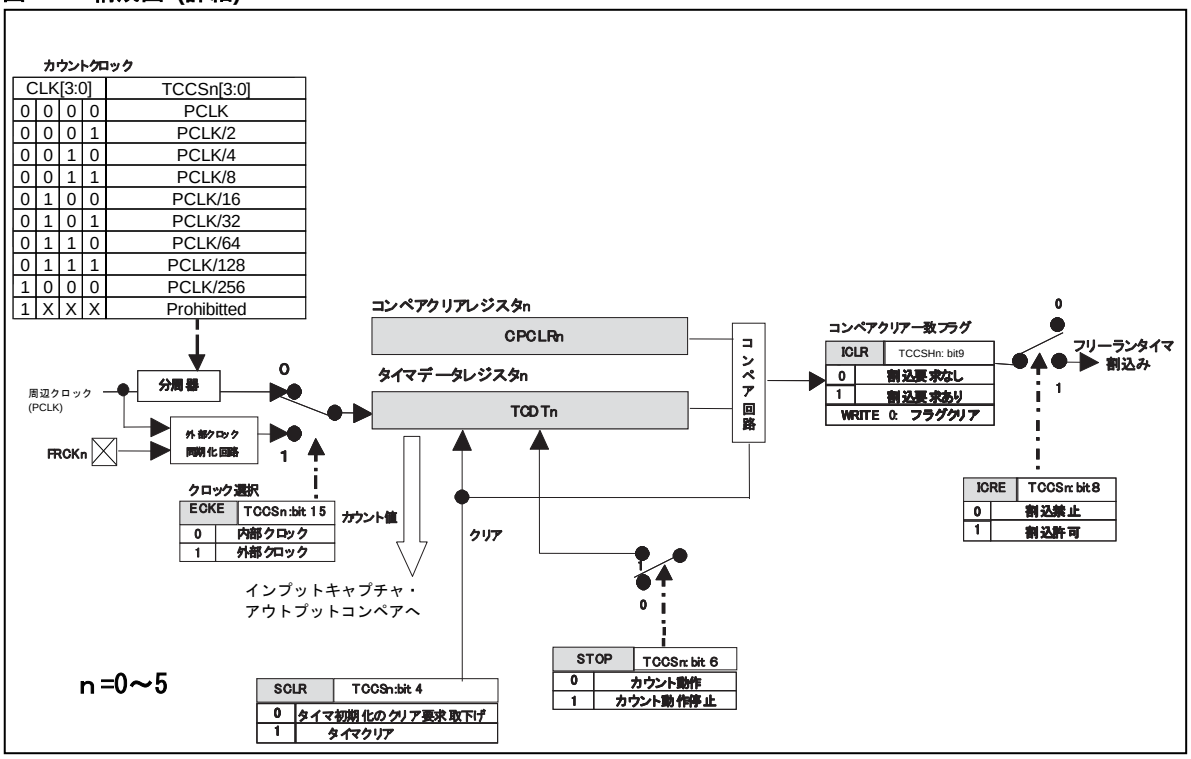
フリーランタイマの特長について説明します。

- 形式 : 32 ビットアップカウンタ
- 個数 : 6
- クロックソース : 内部クロック(周辺クロック(PCLK))の 9 種類の分周 (÷1、÷2、÷4、÷8、÷16、÷32、÷64、÷128、÷256 、または外部クロック(FRCK)
- カウントのクリア要因 :
  - ソフトウェア
  - リセット
  - コンペア一致 (コンペアクリアレジスタとフリーランタイマのカウント値の一致)
- 動作開始/停止 : ソフトウェアで動作開始と停止の制御が可能
- 割込み : コンペアクリア割込み
- カウント値 : 読書き可能 (書込みはカウント停止時のみ)
- 32 ビットフリーランタイマは 32 ビットアップカウンタ, 制御レジスタ, 32 ビットコンペアクリアレジスタ, およびプリスケーラから構成されています。
- コンペアクリア割込みは、コンペアクリアレジスタと 32 ビットフリーランタイマが比較され、一致した場合に生成されます。
- リセットやソフトウェアクリア、あるいはコンペアクリアレジスタとのコンペア一致が発生すると、カウンタ値は"00000000H"にリセットされます。
- アウトプットコンペア,インプットキャプチャのリファレンスカウントとして使用されます。

21.3 構成

フリーランタイマの構成について示します。

図 21-2 構成図 (詳細)



21.4 レジスタ

フリーランタイマのレジスタについて説明します。

ベースアドレス (Base_addr)・外部端子表

チャネル	Base_addr	外部端子
		FRCK
0	0x0240	FRCK0_0/FRCK0_1
1	0x024C	FRCK1_0/FRCK1_1
2	0x0FA0	FRCK2_0/FRCK2_1
3	0x0FAC	FRCK3_0/FRCK3_1
4	0x0FB8	FRCK4_0
5	0x0FC4	FRCK5_0

表 21-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0240	CPCLR0				コンペアクリアレジスタ 0
0x0244	TCDT0				タイマデータレジスタ 0
0x0248	TCCSH0	TCCSL0	予約		タイマ制御レジスタ上位 0 タイマ制御レジスタ下位 0
0x024C	CPCLR1				コンペアクリアレジスタ 1
0x0250	TCDT1				タイマデータレジスタ 1
0x0254	TCCSH1	TCCSL1	予約		タイマ制御レジスタ上位 1 タイマ制御レジスタ下位 1
0x0FA0	CPCLR2				コンペアクリアレジスタ 2
0x0FA4	TCDT2				タイマデータレジスタ 2
0x0FA8	TCCSH2	TCCSL2	予約		タイマ制御レジスタ上位 2 タイマ制御レジスタ下位 2
0x0FAC	CPCLR3				コンペアクリアレジスタ 3
0x0FB0	TCDT3				タイマデータレジスタ 3
0x0FB4	TCCSH3	TCCSL3	予約		タイマ制御レジスタ上位 3 タイマ制御レジスタ下位 3
0x0FB8	CPCLR4				コンペアクリアレジスタ 4
0x0FBC	TCDT4				タイマデータレジスタ 4
0x0FC0	TCCSH4	TCCSL4	予約		タイマ制御レジスタ上位 4 タイマ制御レジスタ下位 4
0x0FC4	CPCLR5				コンペアクリアレジスタ 5
0x0FC8	TCDT5				タイマデータレジスタ 5
0x0FCC	TCCSH5	TCCSL5	予約		タイマ制御レジスタ上位 5 タイマ制御レジスタ下位 5

21.4.1 タイマ制御レジスタ (上位ビット) : TCCSH

タイマ制御レジスタ(上位ビット)のビット構成について説明します。

フリーランタイマの動作を制御するためのレジスタです。

■ TCCSH0-5 (フリーランタイマ 0-5) : アドレス Base_addr+08H (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	ECKE	−	−	−	−	−	ICLR	ICRE
初期値	0	0	0	0	0	0	0	0
属性	R/W	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R(RM1),W	R/W

[bit15] ECKE : クロック選択

ECKE	カウントクロック選択
0	内部クロック
1	外部クロック (FRCK 端子)

■ "0"を設定した場合 : 内部クロックが選択されます。カウントクロック周波数を選択するためには、TCCSL レジスタのクロック周波数選択ビット(CLK3～CLK0:bit3～bit0)も選択してください。

■ "1"を設定した場合 : 外部クロックが選択されます。外部クロックは、FRCK 端子から入力されます。したがって、FRCK 入力端子に対応するポート方向レジスタ(DDR)のビットへ"0"を書込み、対応するポート機能レジスタ(PFR)のビットへ"0"を書き込んでポート入力状態として、外部クロック入力を有効にしてください。ECKE ビットが外部クロックを選択している場合、クロックカウントは両エッジにて動作いたします。外部クロックのパルス幅は 4/F_{PCLK} 以上にしてください。

＜注意事項＞

カウントクロック選択ビットの設定変更は、フリーランタイマの出力を利用しているほかの周辺モジュール (アウトプットコンペア, インプットキャプチャ) が停止している状態で行ってください。

[bit14～bit10] - : 未定義

■ 常に"0"が読み出されます。書込みは動作に影響ありません。

[bit9] ICLR : コンペアクリア割込みフラグ

ICLR	状態	
	読出し時	書込み時
0	コンペアクリア一致なし	フラグ(ICLR)をクリア
1	コンペアクリア一致あり	動作に影響なし

■ コンペアクリア値と 32 ビットフリーランタイマ値が一致すると、このビットには"1"が設定されます。

[bit8] ICRE : コンペアクリア割込み要求許可

ICRE	動作
0	割込み禁止
1	割込み許可

■ ICRE ビットとコンペアクリア割込みフラグビット(ICLR)に"1"が設定されると、CPU に対する割込み要求が生成されます。

21.4.2 タイマ制御レジスタ (下位ビット) : TCCSL

タイマ制御レジスタ (下位ビット) のビット構成について説明します。

フリーランタイマの動作を制御するためのレジスタです。

■ TCCSL0-5 (フリーランタイマ 0-5) : アドレス Base_addr+09H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	−	STOP	−	SCLR	CLK3	CLK2	CLK1	CLK0
初期値	0	1	0	0	0	0	0	0
属性	R0,WX	R/W	R0,WX	R0,W	R/W	R/W	R/W	R/W

[bit7] - : 未定義
常に"0"が読み出されます。書込みは動作に影響ありません。

[bit6] STOP : タイマ許可

STOP	動作
0	カウント許可 (動作)
1	カウント禁止 (停止)

■ STOP ビットは、32 ビットフリーランタイマのカウントを停止/開始するために使用します。
■ STOP ビットが"0"のとき :32 ビットフリーランタイマのカウンタは開始します。
■ STOP ビットが"1"のとき :32 ビットフリーランタイマのカウンタは停止します。

＜注意事項＞
アウトプットコンペアを使用している場合、フリーランタイマが停止するとアウトプットコンペアの動作も停止します。

[bit5] - : 未定義
■ 常に"0"が読み出されます。書込みは動作に影響ありません。

[bit4] SCLR : タイマクリア

SCLR	状態	
	読出し時	書込み時
0	常に"0"が読み出されます。	"0"ライトは動作に影響しません。
1		フリーランタイマをクリア

■ クリアビットを"1"にすると、フリーランタイマのカウント値を"00000000H"にクリアします。この際、マクロ内部のプリスケーラもクリアされます。
■ 読出し値は、必ず"0"です。

＜注意事項＞
"1"を設定した後、次の内部クロックのタイミングでタイマクリアが行われます。

[bit3～bit0] CLK3～CLK0 : クロック周波数選択 (内部クロック選択時)

CLK3	CLK2	CLK1	CLK0	クロック周波数選択 (F_{PCLK} : 周辺クロック(PCLK))				
				カウントクロック	F_{PCLK} =16MHz	F_{PCLK} =8MHz	F_{PCLK} =4MHz	F_{PCLK} =1MHz
0	0	0	0	$1/F_{PCLK}$	62.5ns	125ns	0.25µs	1µs
0	0	0	1	$2/F_{PCLK}$	125ns	0.25µs	0.5µs	2µs
0	0	1	0	$4/F_{PCLK}$	0.25µs	0.5µs	1µs	4µs
0	0	1	1	$8/F_{PCLK}$	0.5µs	1µs	2µs	8µs
0	1	0	0	$16/F_{PCLK}$	1µs	2µs	4µs	16µs
0	1	0	1	$32/F_{PCLK}$	2µs	4µs	8µs	32µs
0	1	1	0	$64/F_{PCLK}$	4µs	8µs	16µs	64µs
0	1	1	1	$128/F_{PCLK}$	8µs	16µs	32µs	128µs
1	0	0	0	$256/F_{PCLK}$	16µs	32µs	64µs	256µs
その他の設定				—	禁止	禁止	禁止	禁止

■ クロック周波数選択ビットへの設定変更と同時に周波数を変更します。フリーランタイマのカウントクロックとして内部クロックを選択している場合 (クロック選択ビット(ECKE= 0))、 フリーランタイマの出力を利用しているほかの周辺モジュール (アウトプットコンペア, インプットキャプチャ) が停止している状態で変更してください。

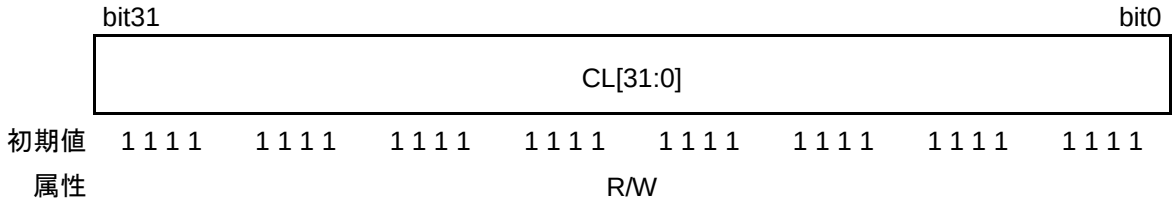
■ アウトプットコンペアのコンペアデータとして使用する場合は、CLK[3:0]= 0000_B の設定は禁止となります。

21.4.3 コンペアクリアレジスタ ：CPCLR

コンペアクリアレジスタ (CPCLR) のビット構成について説明します。

コンペアクリアレジスタは、フリーランタイマと比較するための 32 ビットレジスタです。

■ CPCLR0-5 (フリーランタイマ 0-5) : アドレス Base_addr+00_H (アクセス : ワード)



[bit31～bit0] CL[31:0] : **コンペアクリア**

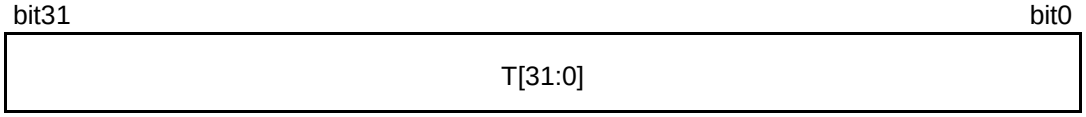
- コンペアクリアレジスタは、32 ビットフリーランタイマのカウント値と比較するために使用します。本レジスタのカウント値がフリーランタイマのカウント値と一致すると、32 ビットフリーランタイマは、"00000000_H" にリセットされ、本レジスタに設定した値とカウンタ値が一致したときに割込みが発生します。ただし、値の書込みはタイマの停止中 (タイマ状態制御レジスタ下位 (TCCSL) の STOP=1)でなければなりません。
- このレジスタは動作中に書き込んでも意味を持ちません。
- このレジスタへアクセスする場合は、ワードアクセス命令を使用してください。

21.4.4 タイマデータレジスタ ：TCDT

タイマデータレジスタ (TCDT) のビット構成について説明します。

タイマデータレジスタは、32 ビットフリーランタイマのカウント値を読み出すために使用します。

■ TCDT0-5 (フリーランタイマ 0-5) : アドレス Base_addr+04H (アクセス ： ワード)



初期値	0000	0000	0000	0000	0000	0000	0000	0000
属性	R,W							

- タイマデータレジスタを読むと、32 ビット フリーランタイマのカウント値が読み出せます。
- タイマデータレジスタに書くことで、フリーランタイマにタイマ値が書き込めます。書込みは、必ずフリーランタイマが停止状態 (タイマ制御レジスタ下位(TCCSL の STOP= 1)) で行ってください。
- このレジスタへアクセスする場合は、ワードアクセス命令を使用してください。
- 32 ビットフリーランタイマは、以下の要因が発生すると直ちに初期化されます。
  - ☐ ⑳リセット
  - ☐ タイマ状態制御レジスタ (TCCSL)のクリアビット (SCLR = 1)
  - ☐ コンペアクリアレジスタとタイムカウント値の一致
- このレジスタは動作中に書き込んでも意味を持ちません。

21.5 動作説明

フリーランタイマの動作について説明します。

21.5.1. フリーランタイマのカウント動作

21.5.2. カウントアップ

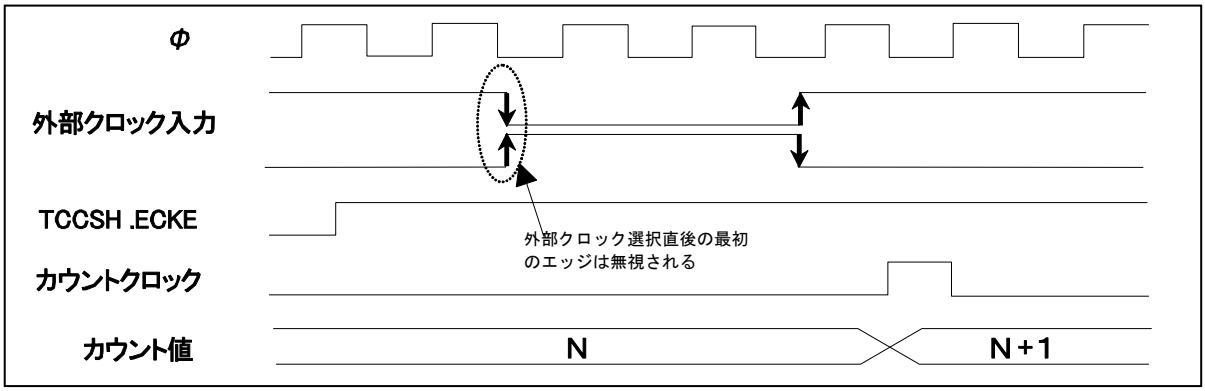
21.5.3. タイマクリア

21.5.4. フリーランタイマの各種クリア動作

21.5.5. タイマ割込み

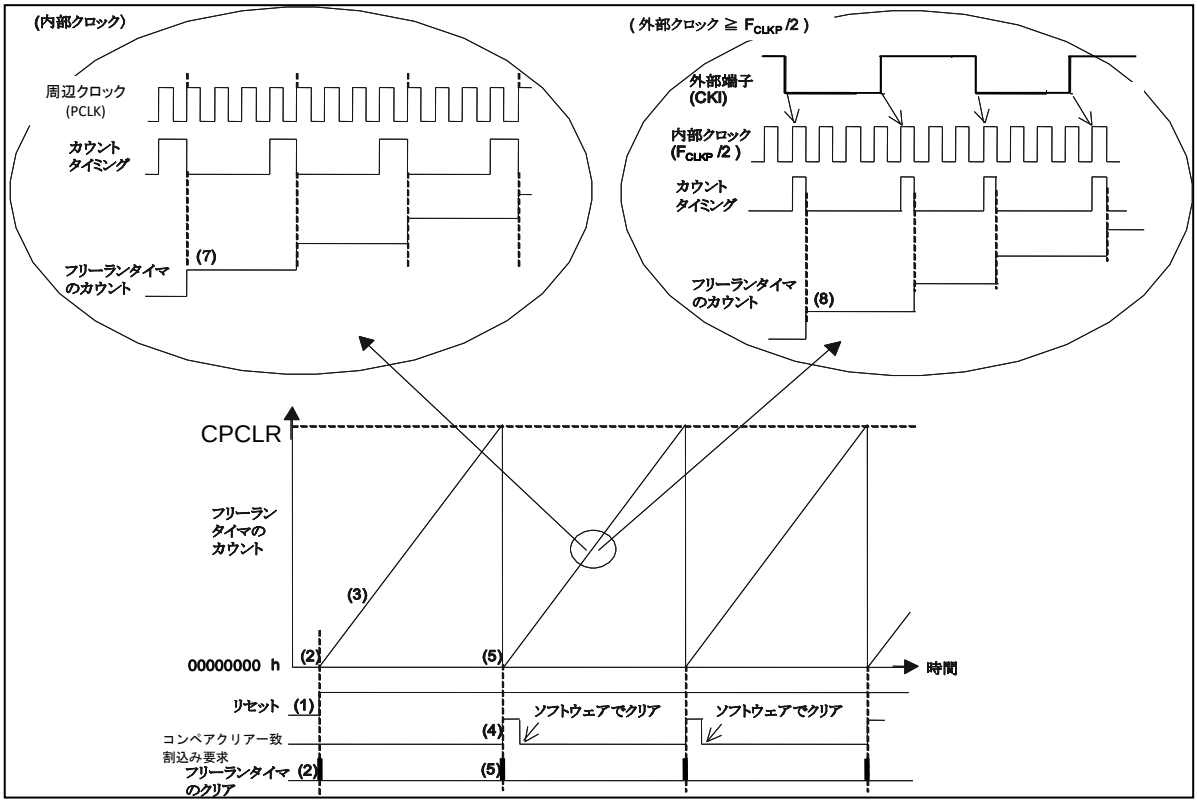
21.5.1 フリーランタイマのカウント動作

フリーランタイマのカウント動作について説明します。



フリーランタイマは、入力クロック (内部クロックまたは外部クロック)に基づいてインクリメントされます。外部クロックモード (TCCSH:ECKE = 1)に選択されると、フリーランタイマは、外部入力クロックの立上り，立下りの両エッジでカウントアップします。

なお、外部クロックモード選択直後の一番目の外部クロック立上り，立下りのエッジは無視されます。つまり外部クロック入力の初期値が"1"のときは最初の立下りエッジが無視され、外部クロック入力の初期値が"0"のときは最初の立上りエッジが無視されます。



(1) リセット
(2) リセットによるフリーランタイマのクリア (カウント値"00000000_H")
(3) フリーランタイマのカウントアップ
(4) フリーランタイマのコンペアクリアー致とその割込み発生

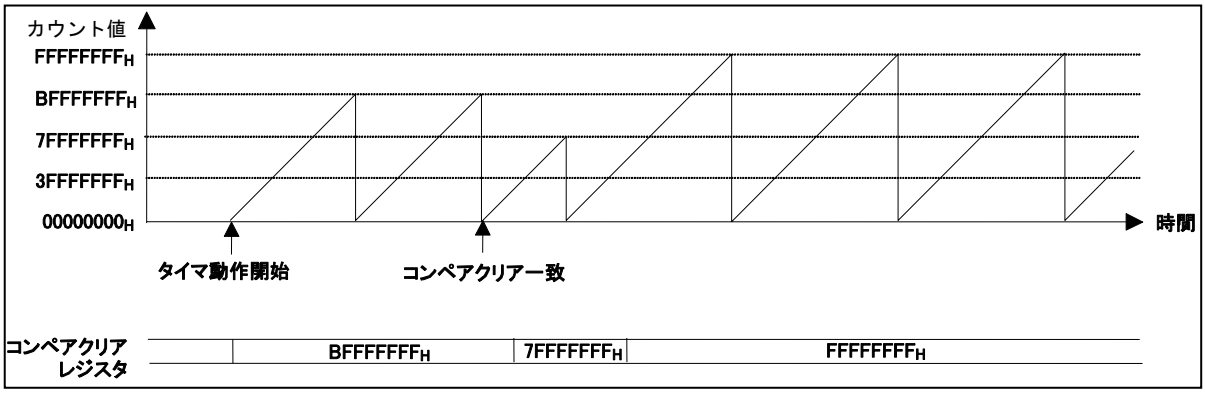
(5) コンペアクリア一致によるフリーランタイマのクリア (カウント値"00000000_H")
(6) (3)から(5)の繰返し
(7) 内部クロックを分周したクロック (カウントクロック) でフリーランタイマがカウントアップします。
(8) 外部クロックを内部クロックで同期化したカウントクロックでフリーランタイマがカウントアップします。

21.5.2 カウントアップ

フリーランタイマのカウントアップについて説明します。

32 ビットフリーランタイマは、アップカウンタです。カウンタは事前に設定されているタイマデータレジスタ(TCDT)からカウントアップを開始し、カウント値がコンペアクリアレジスタ (CPCLR)の値と一致するまでカウントアップし、次にカウンタは、"00000000_H"にクリアされ、次に再びカウントアップします。

図 21-3 アップカウンタ動作



21.5.3 タイマクリア

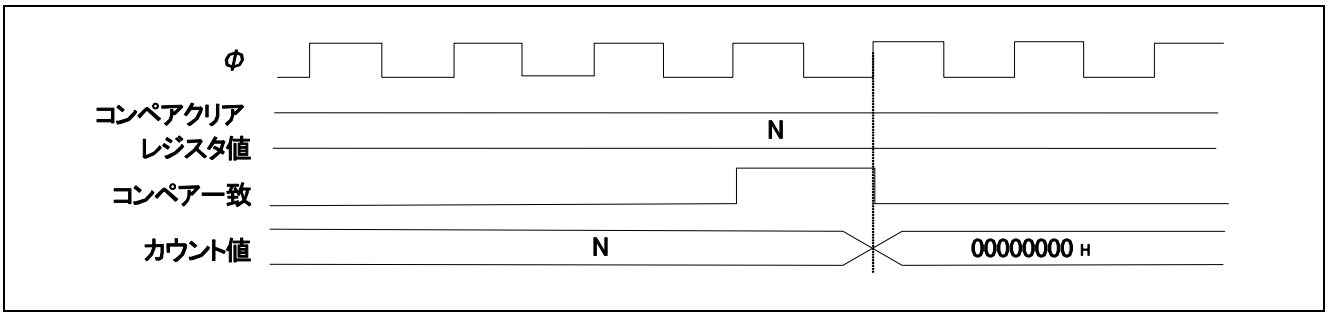
フリーランタイマのタイマクリアについて説明します。

フリーランタイマのカウント値は、下記のいずれかの場合にクリアされます。

- コンペアクリアレジスタとの一致が検出された場合
- 動作中に TCCSL レジスタの SCLR に"1"が書き込まれた場合
- 停止中に TCDT レジスタに"00000000_H"が書き込まれた場合
- リセットされた場合

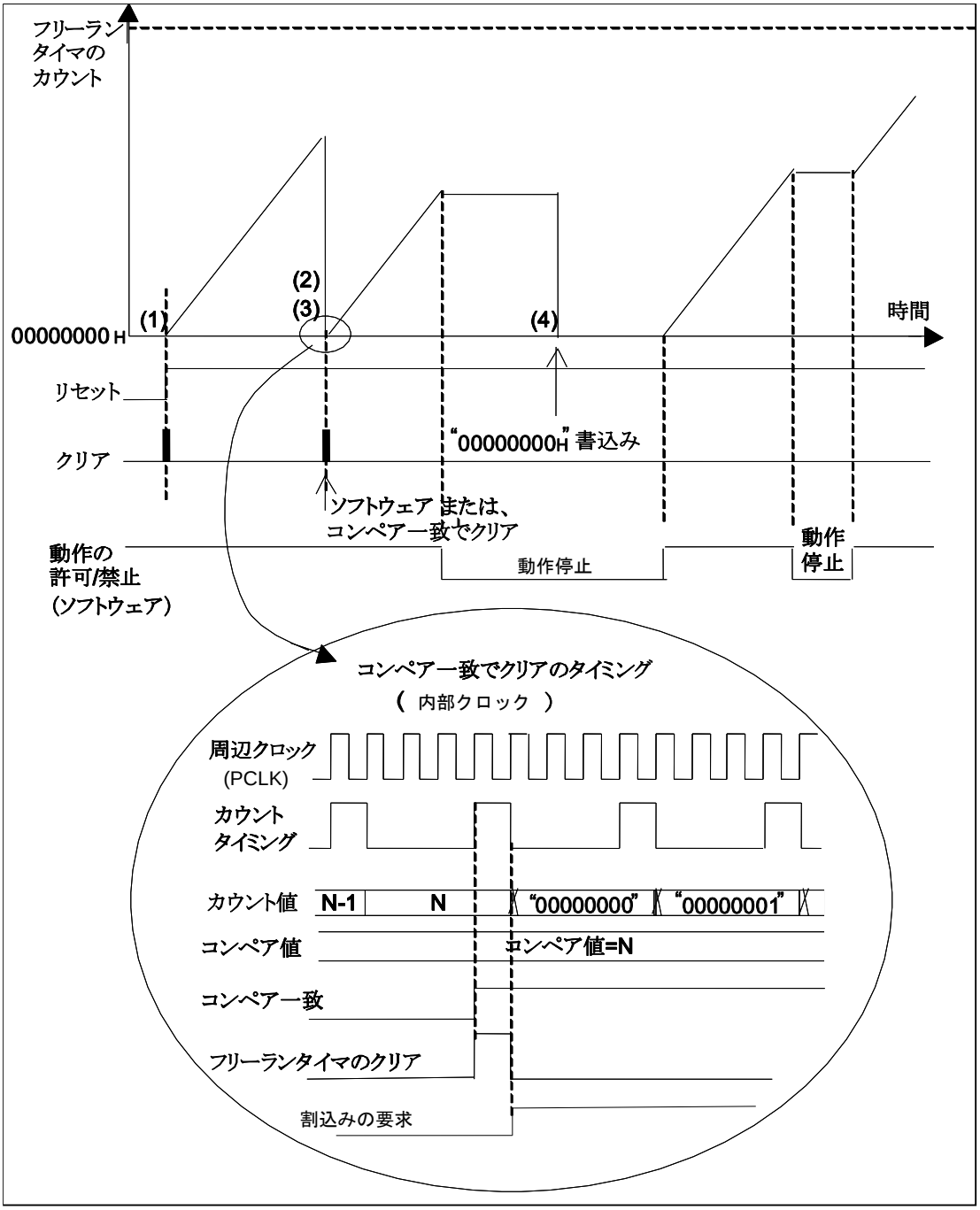
リセットされると、カウンタは直ちにクリアされます。コンペアクリアレジスタとの一致が発生した場合は、カウンタはカウントタイミングと同期してクリアされます。

図 21-4 フリーランタイマのクリアタイミング



21.5.4 フリーランタイマの各種クリア動作

フリーランタイマの各種クリア動作について説明します。



フリーランタイマのクリア (4 種類)

(1) リセットされた場合
(2) 動作中に TCCSL レジスタの SCLR:bit4 に"1"が書き込まれた場合
(3) コンペアクリアレジスタとの一致が検出された場合
(4) 停止中に TCDT レジスタに"00000000_H"が書き込まれた場合

21.5.5 タイマ割込み

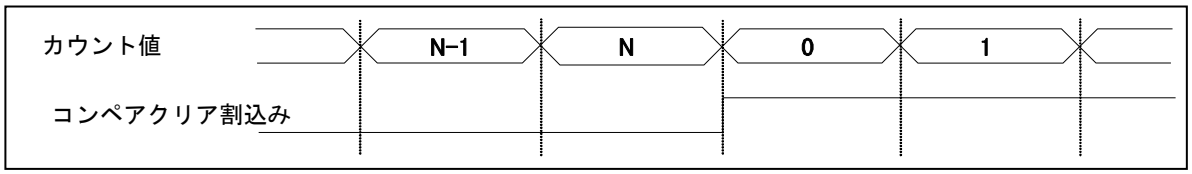
フリーランタイマのタイマ割込みについて説明します。

フリーランタイマでは、以下の割込みを生成できます。

■ コンペアクリア割込み

コンペアクリア割込みは、タイマ値がコンペアクリアレジスタ CPCLR の値と一致すると生成されます。

図 21-5 割込み



21.6 設定

フリーランタイマの設定について説明します。

表 21-2 フリーランタイマを使うために必要な設定

設定	設定レジスタ	設定方法
タイマの初期化条件の設定	タイマ制御レジスタ (TCCSHn, TCCSLn)	21.7.4 参照
カウントクロックの設定 内部クロックの選択		21.7.1 参照
外部クロックの選択		21.7.2 参照
カウント動作開始		21.7.3 参照
外部クロックの場合 クロック入力端子(FRCK)を入力に設定	端子をペリフェラル入力に設定します。 『I/O ポート』の章を参照してください。	

表 21-3 フリーランタイマ割込みを行うために必要な設定

設定	設定レジスタ	設定方法
フリーランタイマ割込みベクタ, フリーランタイマ割込みレベルの設定	『割込み制御(割込みコントローラ)』の章を参照してください。	21.7.5 参照
フリーランタイマ割込み設定 割込み要求のクリア 割込み要求の許可	タイマ制御レジスタ (TCCSHn)	21.7.6 参照

表 21-4 フリーランタイマを停止させるために必要な設定

設定	設定レジスタ	設定方法
フリーランタイマ停止ビットの設定	タイマ制御レジスタ (TCCSLn)	21.7.7 参照

21.7 Q&A

フリーランタイマの Q&A について説明します。

21.7.1. 内部クロック分周の選択方法は?

21.7.2. 外部クロックの選択方法は?

21.7.3. フリーランタイマのカウント動作を許可/禁止するには?

21.7.4. フリーランタイマをクリアするには?

21.7.5. 割込み関連レジスタは?

21.7.6. コンペアクリア割込みを許可するには?

21.7.7. フリーランタイマ動作を停止するには?

21.7.1 内部クロック分周の選択方法は?

内部クロック分周の選択方法について説明します。

9 通りの内部クロック分周があります。クロック選択ビット (TCCSHn:ECKE)と、カウントクロックビット (TCCSLn:CLK[3:0])で設定してください(n はチャネル番号)。

内部クロック	設定	
	クロック選択ビット (ECKE)	カウントクロックビット(CLK[3:0])
F_{PCLK} を選択するには	"0"にする	"0000"にする
2/F_{PCLK} を選択するには	"0"にする	"0001"にする
4/F_{PCLK} を選択するには	"0"にする	"0010"にする
8/F_{PCLK} を選択するには	"0"にする	"0011"にする
16/F_{PCLK} を選択するには	"0"にする	"0100"にする
32/F_{PCLK} を選択するには	"0"にする	"0101"にする
64/F_{PCLK} を選択するには	"0"にする	"0110"にする
128/F_{PCLK} を選択するには	"0"にする	"0111"にする
256/F_{PCLK} を選択するには	"0"にする	"1000"にする

21.7.2 外部クロックの選択方法は?

外部クロックの選択方法について説明します。

クロック選択ビット (TCCSHn:ECKE)と、データ方向ビット，ポート機能ビットで設定してください(n はチャネル番号)。

外部クロック入力にするには	設定		端子	パルス幅 (H 幅, L 幅)
フリーランタイマ 0	クロック選択ビット (ECKE)を"1"にする	FRCK0 端子をペリフェラル入力に設定する。(『I/O ポート』の章を参照してください。)	FRCK0	4/F_{PCLK} 以上
フリーランタイマ 1		FRCK1 端子をペリフェラル入力に設定する。(『I/O ポート』の章を参照してください。)	FRCK1	

ch.2～も同様

21.7.3 フリーランタイマのカウント動作を許可/禁止するには?

フリーランタイマのカウント動作の許可/禁止について説明します。

カウント動作ビット(TCCSLn:STOP)で設定してください(n はチャネル番号)。

動作	カウント動作ビット (STOP)
フリーランタイマを動作させるには	"0"にする
フリーランタイマを停止させるには	"1"にする

21.7.4 フリーランタイマをクリアするには?

フリーランタイマのクリアについて説明します。

フリーランタイマをクリアするには以下の方法があります。

■ クリアビット(TCCSLn:SCLR)で設定する(n はチャネル番号)。

動作	クリアビット (SCLR)
フリーランタイマをクリアするには	"1"を書き込む

■ リセットをかける。
リセット (RSTX 端子入力，ウォッチドッグリセット，ソフトウェアリセットなど) をかけるとフリーランタイマがクリアされ、フリーランタイマは初期状態となります。

■ フリーランタイマが動作停止中に"00000000_H"を書き込む。
フリーランタイマの動作停止中に"00000000_H"を書き込むと、カウント値が"00000000_H"になります。

■ フリーランタイマのオーバフローでカウント値は、"00000000_H"にもどります。
■ コンペアクリアレジスタとの一致が検出された場合、クリアされます。

21.7.5 割込み関連レジスタは?

割込み関連レジスタについて説明します。

フリーランタイマ割込みベクタ，フリーランタイマ割込みレベルの設定

フリーランタイマ番号，割込みレベル，ベクタの関係は『付録』の『付録 C. 割込みベクター覧』を参照してください。

割込みレベル，割込みベクタの詳細については『割込み制御(割込みコントローラ)』の章を参照してください。

番号	割込みベクタ(デフォルト)	割込みレベル設定ビット (ICR[4:0])
フリーランタイマ 0	#50 Address: 0FFF34$_H$	割込みレベルレジスタ(ICR34) Address: 00462$_H$
フリーランタイマ 1	#51 Address: 0FFF30$_H$	割込みレベルレジスタ(ICR35) Address: 00463$_H$
フリーランタイマ 2	#50 Address: 0FFF34$_H$	割込みレベルレジスタ(ICR34) Address: 00462$_H$
フリーランタイマ 3	#51 Address: 0FFF30$_H$	割込みレベルレジスタ(ICR35) Address: 00463$_H$

ch.4～ は『付録』の『付録 C. 割込みベクター覧』を参照してください。

割込み要求フラグ (TCCSHn:ICLR)(n はチャネル番号)は自動的にクリアされませんので、割込み処理から復帰する前にソフトウェアにてクリアしてください。(ICLR ビットに"0"を書き込む)

21.7.6 コンペアクリア割込みを許可するには?

コンペアクリア割込みの許可について説明します。

割込み要求の許可，割込み要求フラグ

割込み許可の設定は、割込み要求許可ビット(TCCSHn:ICRE)にて行ってください(n はチャネル番号)。

動作	コンペアクリア割込み要求許可ビット(ICRE)
割込み禁止	"0"にする
割込み許可	"1"にする

割込み要求のクリアは、割込みフラグビット(TCCSHn:ICLR)にて行ってください(n はチャネル番号)。

動作	コンペアクリア割込みフラグビット(ICLR)
割込み要求クリア	"0"を書き込む

21.7.7 フリーランタイマ動作を停止するには?

フリーランタイマの動作停止について説明します。

カウント動作ビット(TCCS0:STOP), (TCCS1:STOP), (TCCS2:STOP), (TCCS3:STOP)・・・で設定してください。

「21.7.3 フリーランタイマのカウント動作を許可/禁止するには?」を参照してください。

21.8 サンプルプログラム

フリーランタイマのサンプルプログラムについて説明します。

設定手順例 1

フリーランタイマ 0, クロック＝ PCLK/2^6、
割込み処理でコンペアー致回数をカウント。

<初期設定>

・フリーランタイマ ch.0 の制御　　レジスタ名．ビット名

制御レジスタの設定 クロック選択≫	TCCSH0/TCCSL0 .ECKE
コンペア割込み要求フラグ≫ コンペア割込み要求許可≫	.ICLR .ICRE
カウント動作≫	.STOP
TCDT クリア≫ カウントクロック≫	.CLR .CLK3-0
タイマデータ値の設定	TCDT0

・割込み関連　　レジスタ名．ビット名

割込みレベル設定	ICR34
I フラグの設定	(CCR)

・変数の設定

<起動>

・フリーランタイマ ch.0 起動　　レジスタ名．ビット名

カウント動作起動	TCCS0 .STOP

<割込み>

・割込み処理　　レジスタ名．ビット名

割込み要求フラグのクリア	TCCS0.ICLR
(任意の処理)	
変数のカウント	

<割込みベクタ>
　ベクタテーブルの設定

<その他>
(注意事項)
　事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 1

```
void FREE_RUN_TIMER0_sample(void)
{
        FREERUN0_initial();
        FREERUN0_start();
}

void FREERUN0_initial(void)
{
        IO_TCCS1.word = 0x0041; /*  設定値=0000_0000_0100_0001 */
                                /* bit15 = 0          ECKE  内部クロックソース  */
                                /* bit14 -10= 00000  予約ビット  */
                                /* bit9 = 0           ICLR  コンペア割込み要求フラグ  */
                                /* bit8 = 0           ICRE  コンペア割込み禁止  */
                                /* bit7 = 0           予約ビット*/
                                /* bit6 = 1           STOP  カウント禁止  */
                                /* bit5 = 0           予約ビット  */
                                /* bit4 = 0           SCLR  フリーランタイマ値の初期化( なし)*/
                                /* bit3-0 = 0001      CLK3-0  カウントクロック  PCLK/2=32MHz/2 */

        IO_TCDT0 = 0x0000;      /*  タイマデータ値の初期化  */

        IO_ICR[34].byte = 0x10; /*  フリーランタイマ 0  割込みレベル設定(  値は任意) */
        __EI();                 /*  割込み許可  */
        count = 0;
}

void FREERUN0_start(void)
{
        IO_TCCS0.bit.STOP = 0;  /* bit6 = 0 STOP  カウント許可  */
}

__interrupt void
FREE_RUN_TIMER0_int(void)
{
        IO_TCCS0.bit.ICLR = 0;  /* bit9 = 0 ICLR  コンペアー致フラグのクリア  */

        count++;
}

ベクタテーブルにて割込みルーチンの指定が必要
#pragma intvect FREE_RUN_TIMER0_int 50
```

21.9 注意事項

フリーランタイマの注意事項について説明します。

● フリーランタイマのクリア

- リセット (RSTX 端子入力, ウォッチドッグリセット, ソフトウェアリセットなど)すると、カウンタは初期化"00000000_H"してカウントを停止します。
- ソフトウェアクリア(TCCSL:SCLR=1)は、クリア要求が発生すると次のサイクルでカウンタがクリアしますが、コンペア一致でのカウンタのクリアは、カウントアップするタイミングと同じタイミングで行われます。
- フリーランタイマのカウンタクリア動作(ソフトウェア, コンペア一致)は、フリーランタイマが動作中のとき有効となります。フリーランタイマ停止中にクリアするには、タイマカウントデータレジスタに"00000000_H"を書き込む必要があります。

● タイマデータレジスタへの書込み

- フリーランタイマに値を書き込む場合には、書込みは必ずフリーランタイマが停止している状態(STOP = "1")でかつ、ワードアクセスで行ってください。

● 外部クロック動作

- 外部クロック時のコンペア一致出力と割込み発生のタイミングは、コンペア一致後の次のカウントクロックタイミングとなります。そのため、コンペア一致出力と割込みを発生させるためには、コンペア一致後に最低 1 クロック分 (外部クロック) を入力する必要があります。

● リードモディファイライト

- タイマ制御レジスタのコンペアクリア割込みフラグビットをリードモディファイライト系命令で読み出すと"1"が読み出せます。

22. アウトプットコンペア

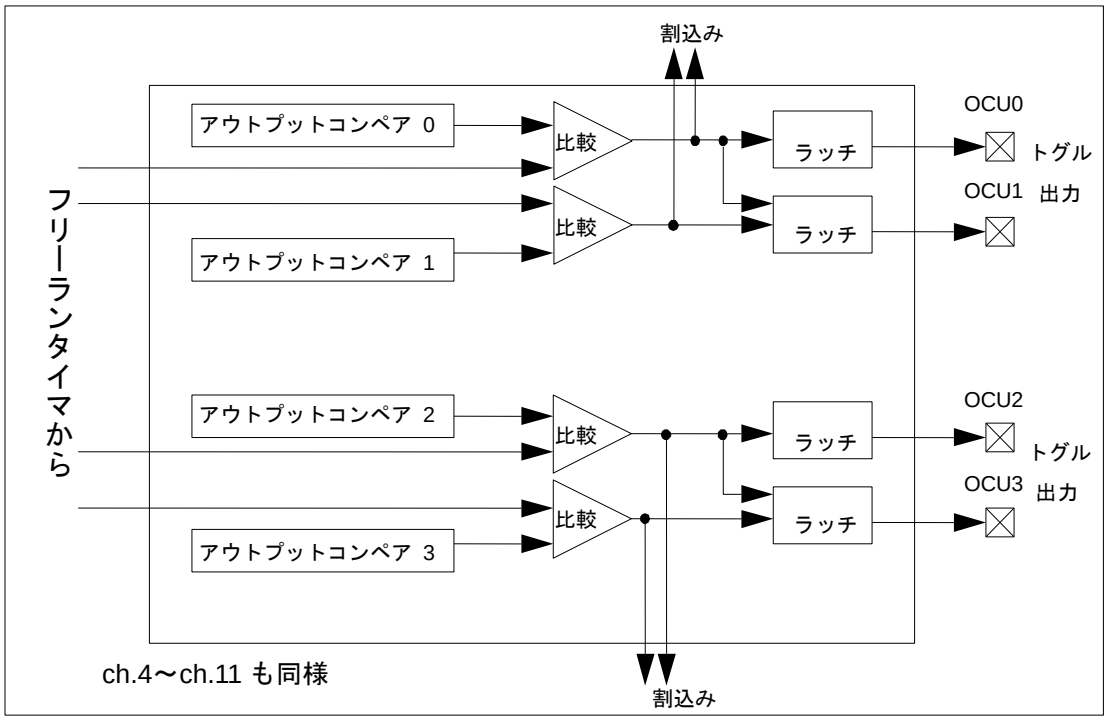
アウトプットコンペアについて説明します。

22.1 概要

アウトプットコンペアの概要について説明します。

アウトプットコンペアは、32 ビットコンペアレジスタ，コンペア出力ラッチ，およびコンペア制御レジスタで構成されています。32 ビットフリーランタイマ値がコンペアレジスタ値と一致すると、出力レベルが反転し、割込みを発生させることもできます。

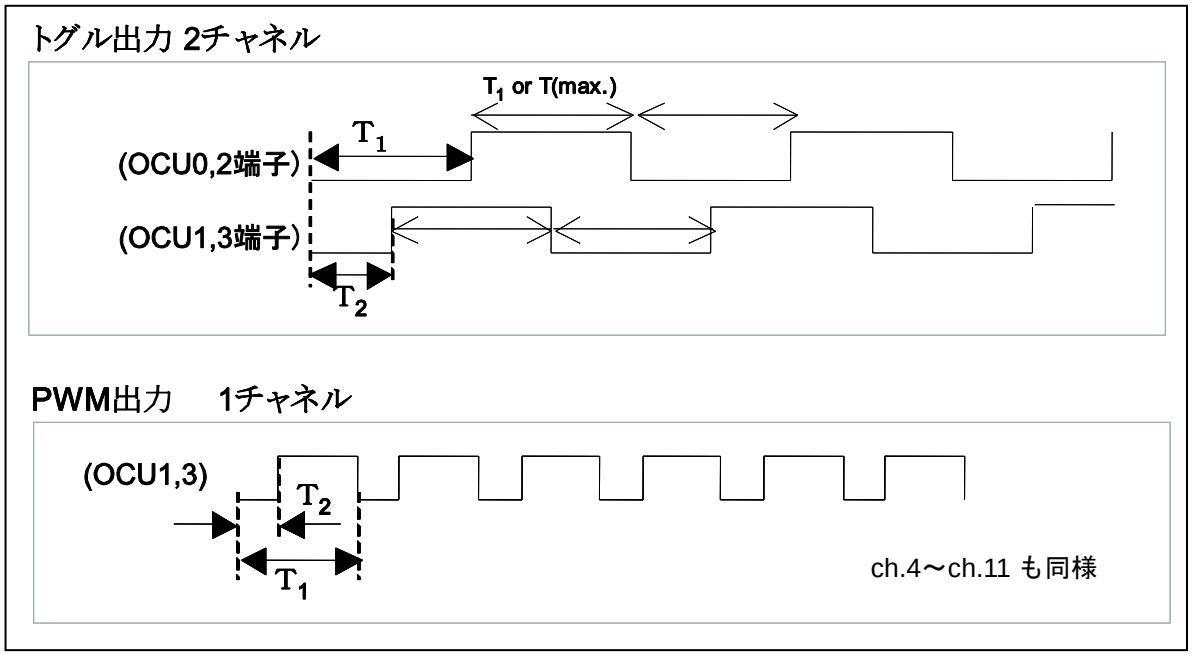
図 22-1 ブロックダイヤグラム (概要)



22.2 特長

アウトプットコンペアの特長について説明します。

図 22-2 出力波形



- 形式 : 32 ビットコンペアレジスタ×4＋比較回路
- 対応するタイマ : フリーランタイマを使用
  - □ アウトプットコンペア 0-3 : フリーランタイマ 0 または 1 を使用
  - □ アウトプットコンペア 6-9 : フリーランタイマ 2 または 3 を使用
  - □ アウトプットコンペア 4-5,10-11 : フリーランタイマ 4 または 5 を使用
- 個数 : 12 チャネル
- コンペア一致による動作 :
  - □ 端子出力値の反転（トグル出力）
  - □ 割込み発生
- カウント精度 : 周辺クロック(PCLK) ÷2、÷4、÷8、÷16、÷32、÷64、÷128、÷256 (フリーランタイマに依存)

＜注意事項＞

周辺クロック(PCLK) ÷1 は設定禁止です。

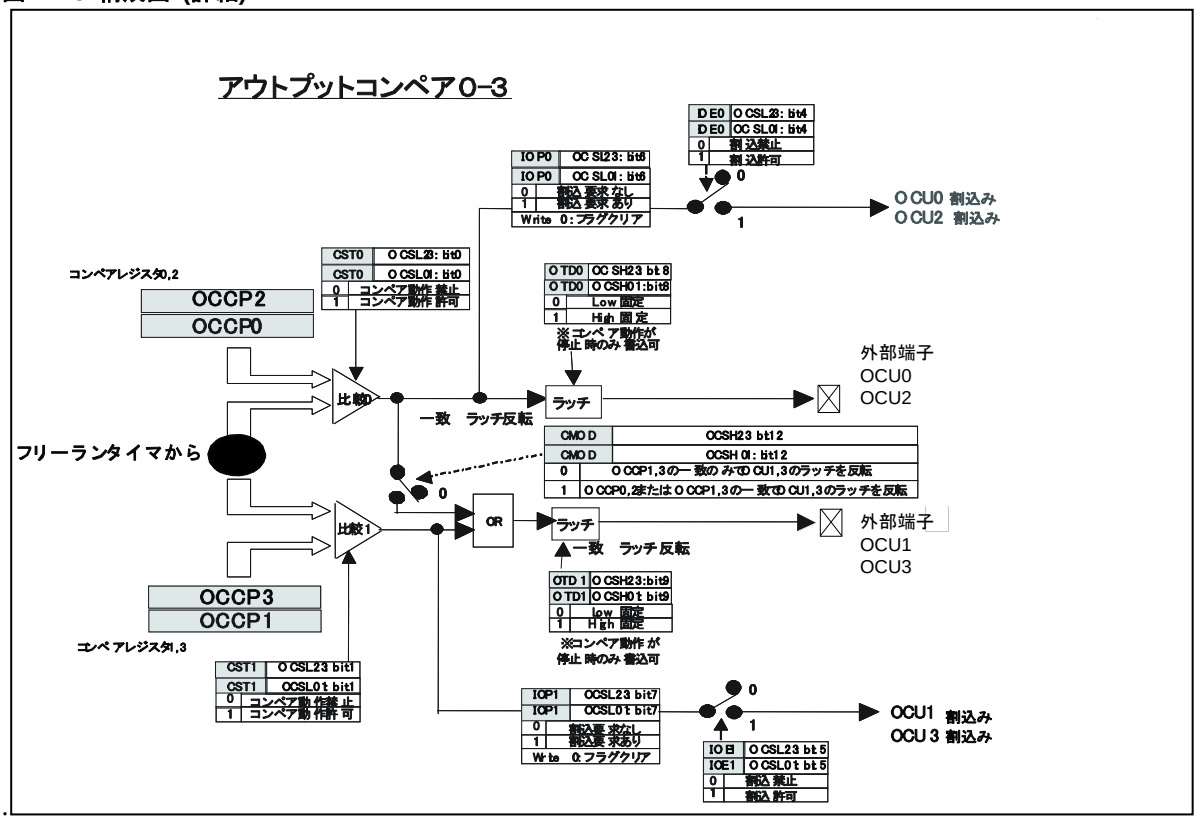
- トグル変化の幅 (T) : 1×カウント精度～100000000H×カウント精度
- 割込み : コンペア一致割込み
- その他 :
  - □ 出力レベルの初期値設定が可能（"H"/"L"）
  - □ OCU 出力として未使用の端子は, 汎用ポートとして使用が可能
  - □ 12 個のコンペアレジスタは独立に使用できます。
  - □ 出力端子と割込みフラグはコンペアレジスタに対応しています。
  - □ 出力端子は 2 つのコンペアレジスタを使用して反転させることができます。 (OCU1, 3,5,7,9,11 のみの機能)

- 各出力端子の初期値を設定できます。
- アウトプットコンペアレジスタが 32 ビットフリーランタイマと一致したとき、割込みを発生させることができます。

22.3 構成図

アウトプットコンペアの構成図について示します。

図 22-3 構成図 (詳細)



ch.4～ch.11 も同様

22.4 レジスタ

アウトプットコンペアのレジスタについて説明します。

チャネル	Base_addr	外部端子
		OCU
0	0x02E8	OCU0_0, OCU0_1
1	0x02E8	OCU1_0, OCU1_1
2	0x02F4	OCU2_0, OCU2_1
3	0x02F4	OCU3_0, OCU3_1
4	0x0F90	OCU4_0, OCU4_1
5	0x0F90	OCU5_0, OCU5_1
6	0x0120	OCU6_0, OCU6_1
7	0x0120	OCU7_0, OCU7_1
8	0x012C	OCU8_0, OCU8_1
9	0x012C	OCU9_0, OCU9_1
10	0x0138	OCU10_0, OCU10_1
11	0x0138	OCU11_0, OCU11_1

表 22-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0120	OCCP6				コンペアレジスタ 6
0x0124	OCCP7				コンペアレジスタ 7
0x0128	OCFS67	予約	OCSH67	OCSL67	フリーランタイマ選択レジスタ 67 アウトプット制御レジスタ 67 上位 アウトプット制御レジスタ 67 下位
0x012C	OCCP8				コンペアレジスタ 8
0x0130	OCCP9				コンペアレジスタ 9
0x0134	OCFS89	予約	OCSH89	OCSL89	フリーランタイマ選択レジスタ 89 アウトプット制御レジスタ 89 上位 アウトプット制御レジスタ 89 下位
0x0138	OCCP10				コンペアレジスタ 10
0x013C	OCCP11				コンペアレジスタ 11
0x0140	OCFS1011	予約	OCSH1011	OCSL1011	フリーランタイマ選択レジスタ 1011 アウトプット制御レジスタ 1011 上位 アウトプット制御レジスタ 1011 下位
0x02E8	OCCP0				コンペアレジスタ 0
0x02EC	OCCP1				コンペアレジスタ 1
0x02F0	OCFS01	予約	OCSH01	OCSL01	フリーランタイマ選択レジスタ 01 アウトプット制御レジスタ 01 上位 アウトプット制御レジスタ 01 下位
0x02F4	OCCP2				コンペアレジスタ 2
0x02F8	OCCP3				コンペアレジスタ 3
0x02FC	OCFS23	予約	OCSH23	OCSL23	フリーランタイマ選択レジスタ 23 アウトプット制御レジスタ 23 上位 アウトプット制御レジスタ 23 下位
0x0F90	OCCP4				コンペアレジスタ 4
0x0F94	OCCP5				コンペアレジスタ 5
0x0F98	OCFS45	予約	OCSH45	OCSL45	フリーランタイマ選択レジスタ 45 アウトプット制御レジスタ 45 上位 アウトプット制御レジスタ 45 下位

22.4.1 フリーランタイマ選択レジスタ: OCFS

フリーランタイマ選択レジスタのビット構成について説明します。

コンペア元のフリーランタイマを選択します。

x : チャネル番号 0, 2, 4, 6, 8, 10
y : チャネル番号 1, 3, 5, 7, 9, 11

■ OCFSxy (フリーランタイマ選択 xy) : アドレス Base_addr+08H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	—	—	—	—	—	—	SELy	SELx
初期値	—	—	—	—	—	—	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W

[bit7～bit2] - : 未定義
このビットへの書込みは動作に影響しません。

[bit1] SELy : フリーランタイマ選択(アウトプットコンペア y)
[bit0] SELx : フリーランタイマ選択(アウトプットコンペア x)

SEL{0,1,2,3}	動作
0	フリーランタイマ 0
1	フリーランタイマ 1

SEL{4,5}	動作
0	フリーランタイマ 4
1	フリーランタイマ 5

SEL{6,7,8,9}	動作
0	フリーランタイマ 2
1	フリーランタイマ 3

SEL{10,11}	動作
0	フリーランタイマ 4
1	フリーランタイマ 5

22.4.2 アウトプット制御レジスタ (上位ビット) : OCSH

アウトプット制御レジスタ上位のビット構成について説明します。

アウトプットコンペアの動作を制御するためのレジスタです。

x : チャネル番号 0, 2, 4, 6, 8, 10
y : チャネル番号 1, 3, 5, 7, 9, 11

■ OCSHxy (アウトプットコンペア xy) : アドレス Base_addr+0AH (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	—	—	—	CMOD	予約	予約	OTDy	OTDx
初期値	—	—	—	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R/W	R/W0	R/W0	R,W	R,W

[bit15～bit13] - : 未定義
これらのビットへの書込みは動作に影響しません。

[bit12] CMOD : コンペアモード

CMOD	動作モード
0	独立動作(OCU0～OCU11 端子の出力のレベル反転動作は独立) ・OCU0, OCU2, 4, 6, 8, 10 端子:フリーランタイマとコンペアレジスタ 0, 2, 4, 6, 8, 10 (OCCP0, OCCP2, 4, 6, 8, 10)が一致した場合に出力反転 ・OCU1, OCU3, 5, 7, 9, 11 端子:フリーランタイマとコンペアレジスタ 1, 3, 5, 7, 9, 11 (OCCP1, OCCP3, 5, 7, 9, 11)と一致した場合に出力反転 比較対象のフリーランタイマは OCFS レジスタで選択します。
1	・連携動作 OCU0, OCU2, 4, 6, 8, 10 端子:フリーランタイマとコンペアレジスタ 0, 2, 4, 6, 8, 10 (OCCP0, OCCP2, 4, 6, 8, 10)が一致した場合に出力反転 ・OCU1, OCU3, 5, 7, 9, 11 端子 : フリーランタイマとコンペアレジスタ (0 または 1) (2 または 3)・・・(8 または 9) (10 または 11) のどちらかと一致した場合に出力反転 比較対象のフリーランタイマは OCFS レジスタで選択します。

■ コンペアレジスタ 0,1 と 2,3 が同じ値の場合は、ただ 1 つのコンペアレジスタが使用される場合と同じ動作になります。(コンペアレジスタ 4-7, コンペアレジスタ 8-11 についても同様です。)

[bit11, bit10] 予約
これらのビットには必ず"0"を設定してください。

[bit9] OTDy: 端子レベル設定

OCUn 端子の出力を許可した場合の端子出力レベル(初期値)を指定します(n は奇数チャネル番号)。

OTDy	動作
0	OCUn 端子の出力レベル(初期値)を"L"に設定
1	OCUn 端子の出力レベル(初期値)を"H"に設定

OCUn 端子の出力を行う場合には、汎用ポートの設定が必要です。設定はコンペア動作を停止してから行ってください。読出し動作では、アウトプットコンペアの端子出力が読み出されます。

[bit8] OTDx: 端子レベル設定

OCUm 端子の出力を許可した場合の端子出力レベル(初期値)を指定します(m は偶数チャネル番号)。

OTDx	動作
0	OCUm 端子の出力レベル(初期値)を"L"に設定
1	OCUm 端子の出力レベル(初期値)を"H"に設定

OCUm 端子の出力を行う場合には、汎用ポートの設定が必要です。設定はコンペア動作を停止してから行ってください。読出し動作では、アウトプットコンペアの端子出力が読み出されます。

22.4.3 アウトプット制御レジスタ (下位ビット) : OCSL

アウトプット制御レジスタ(下位ビット)のビット構成について説明します。

アウトプットコンペアの動作を制御するためのレジスタです。

x : チャネル番号 0, 2, 4, 6, 8, 10
y : チャネル番号 1, 3, 5, 7, 9, 11

■ OCSLxy (アウトプットコンペア xy) : アドレス Base_addr+0B$_H$ (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IOPy	IOPx	IOEy	IOEx	—	—	CSTy	CSTx
初期値	0	0	0	0	1	1	0	0
属性	R(RM1),W	R(RM1),W	R/W	R/W	R1,WX	R1,WX	R/W	R/W

[bit7] IOPy : 割込み要求フラグ(アウトプットコンペア y)
[bit6] IOPx : 割込み要求フラグ(アウトプットコンペア x)

IOP	状態	
	読出し	書込み
0	割込み要求なし	フラグ(IOP)をクリア
1	割込み要求あり	動作に影響なし

- フリーランタイマのカウント値TCDTがアウトプットコンペアのコンペアレジスタ OCCPと一致すると"1"になります。
- 割込み要求は、割込み許可ビット(IOE)が"1"の場合に有効となります。

[bit5] IOEy : 割込み要求許可(アウトプットコンペア y)
[bit4] IOEx : 割込み要求許可(アウトプットコンペア x)

IOE	状態
0	アウトプットコンペア割込み要求の禁止
1	アウトプットコンペア割込み要求の許可

- このビットは、コンペアレジスタのアウトプットコンペア割込みを"許可"にするために使用します。
- このビットに"1"が書き込まれている間にコンペア一致割込みフラグビット(IOP)が設定されると、アウトプットコンペア割込みが発生します。

[bit3, bit2] - : 未定義
書込みは動作に影響ありません。

[bit1] CSTy : 動作許可(アウトプットコンペア y)

[bit0] CSTx : 動作許可(アウトプットコンペア x)

CST	動作
0	アウトプットコンペアの動作停止
1	アウトプットコンペアの動作許可

- フリーランタイマのカウント値 TCDT とアウトプットコンペアのコンペアレジスタの比較動作を許可するビットです。
- コンペア動作を許可する前に必ずコンペアレジスタ OCCP に値を設定してください。
- アウトプットコンペアはフリーランタイマと同期させているため、フリーランタイマを停止するとアウトプットコンペア動作も停止します。

22.4.4 コンペアレジスタ ：OCCP

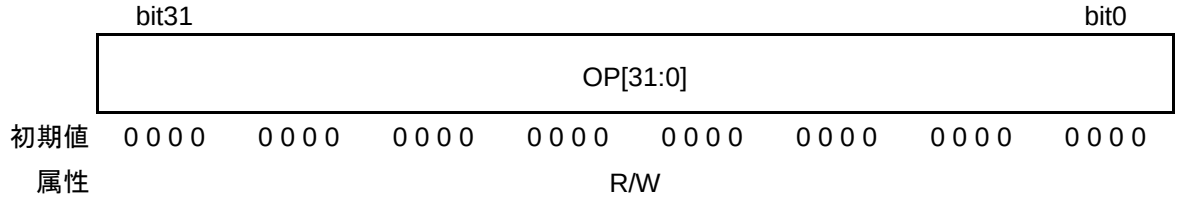
コンペアレジスタのビット構成について説明します。

32 ビットフリーランタイマのカウント値と比較を行う値を設定するレジスタです。

x : チャネル番号 0, 2, 4, 6, 8, 10
y : チャネル番号 1, 3, 5, 7, 9, 11

■ OCCPx (アウトプットコンペア x) : アドレス Base_addr+00H (アクセス： ワード)

■ OCCPy (アウトプットコンペア y) : アドレス Base_addr+04H (アクセス： ワード)



- コンペアレジスタ OCCP と、フリーランタイマのカウント値 TCDT を比較します。
- OCCP レジスタ値と 32 ビットフリーランタイマ値が一致した場合、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。コンペア値は、書込み命令が終了後反映されます。そのため、動作中にコンペア値が変更されると、以前のコンペア値より新規書込みしたコンペア値が大きい数の場合、1 回のフリーランカウント中に 2 回の割込みが生じる事があります。
- また、ポート機能レジスタ(PFR)の対応する OCU をセットし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。
- このレジスタへアクセスする場合、ワードアクセス命令を使用してください。

22.5 動作説明

アウトプットコンペアの動作について説明します。

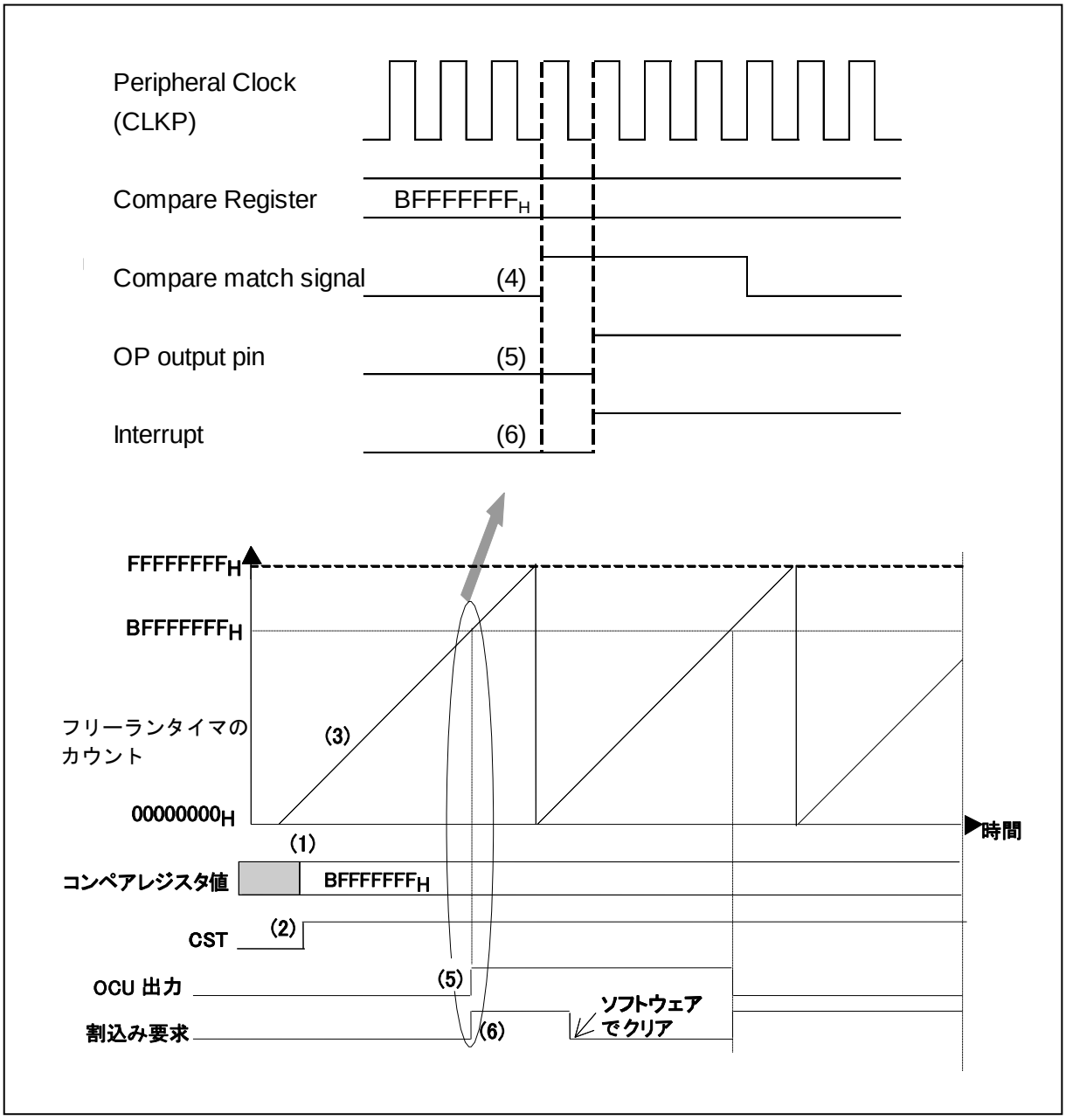
22.5.1 アウトプットコンペア出力 (独立反転) CMOD = "0"

22.5.2 アウトプットコンペア出力 (連携反転) CMOD="1"

22.5.3 アウトプットコンペアの動作タイミング

22.5.1 アウトプットコンペア出力 (独立反転) CMOD = "0"

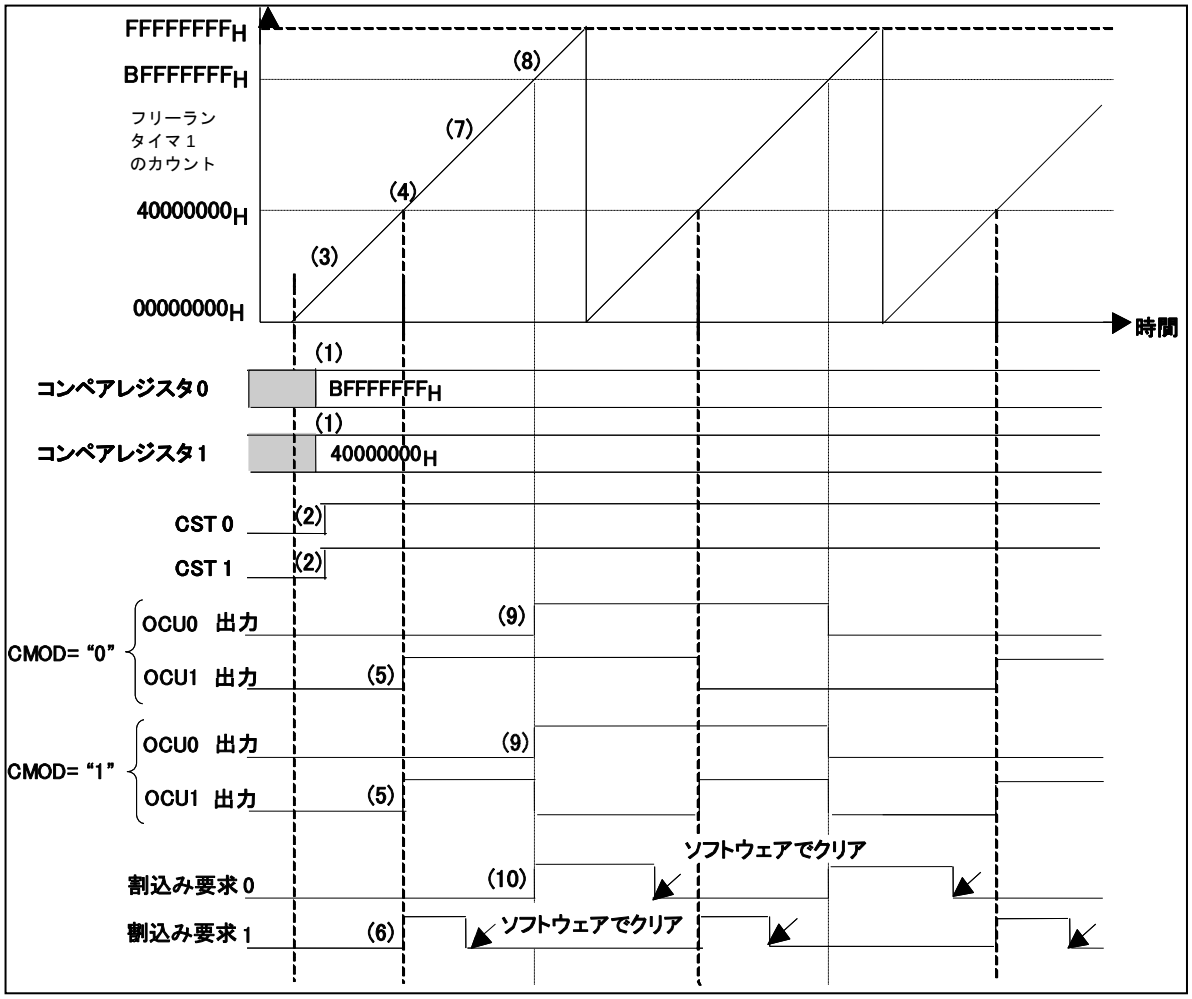
アウトプットコンペア出力 (独立反転) について説明します。



(1) コンペア値の設定
(2) コンペア動作の許可(CST=1)
(3) フリーランタイマのカウントアップ (4 クロックで 1 カウントの例)
(4) フリーランタイマ値とコンペア値を比較して一致 (コンペア一致)
(5) OCU 出力レベルの反転
(6) コンペア一致割込み要求の発生

22.5.2 アウトプットコンペア出力 (連携反転) CMOD="1"

アウトプットコンペア出力 (連携反転) について説明します。



(1) コンペア 0 とコンペア 1 の値の設定
(2) コンペア動作の許可
(3) フリーランタイマのカウントアップ
(4) コンペア 1 一致
(5) OCU1 出力レベルの反転
(6) コンペア 1 一致割込み
(7) フリーランタイマのカウントアップ
(8) コンペア 0 一致
(9) OCU0 出力レベルの反転
CMOD=1 のときは、OCU1 出力レベルも反転
(10) コンペア 0 一致割込み

22.5.3 アウトプットコンペアの動作タイミング

アウトプットコンペアの動作タイミングについて説明します。

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)

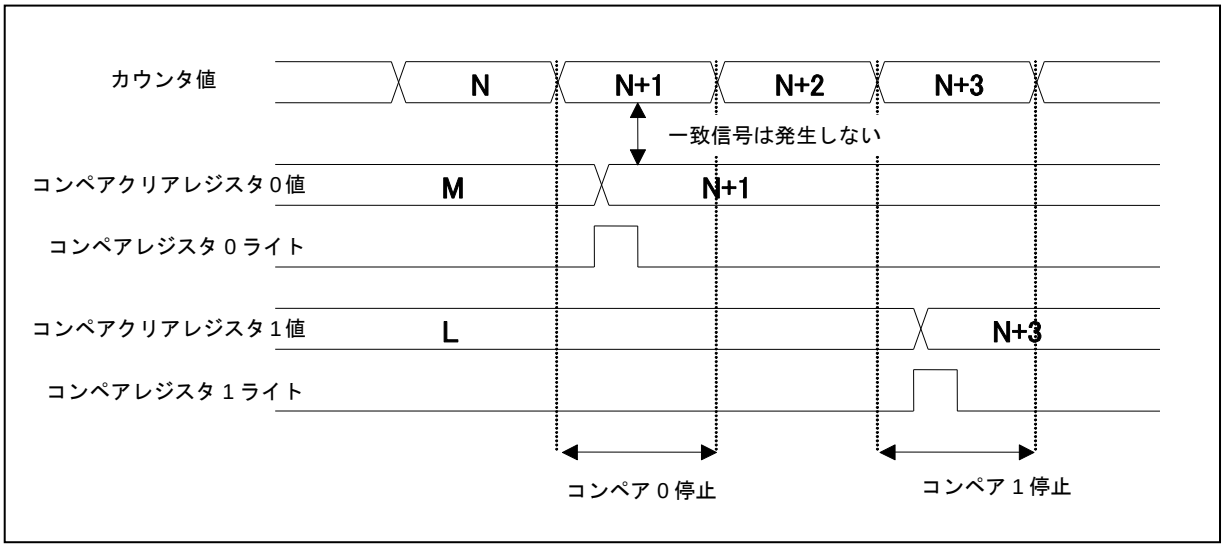
アウトプットコンペアは、フリーランタイマと、設定したコンペアレジスタの値が一致したときにコンペアマッチ信号が発生して出力を反転するとともに割込みを発生することができます。コンペアマッチ時の出力反転タイミングは、カウンタのカウントタイミングに同期して行われます。

22.5.3.1 コンペアレジスタ書込み

コンペアレジスタ書込みについて示します。

コンペアレジスタ書換え時には、カウンタ値とのコンペア動作は行いません。

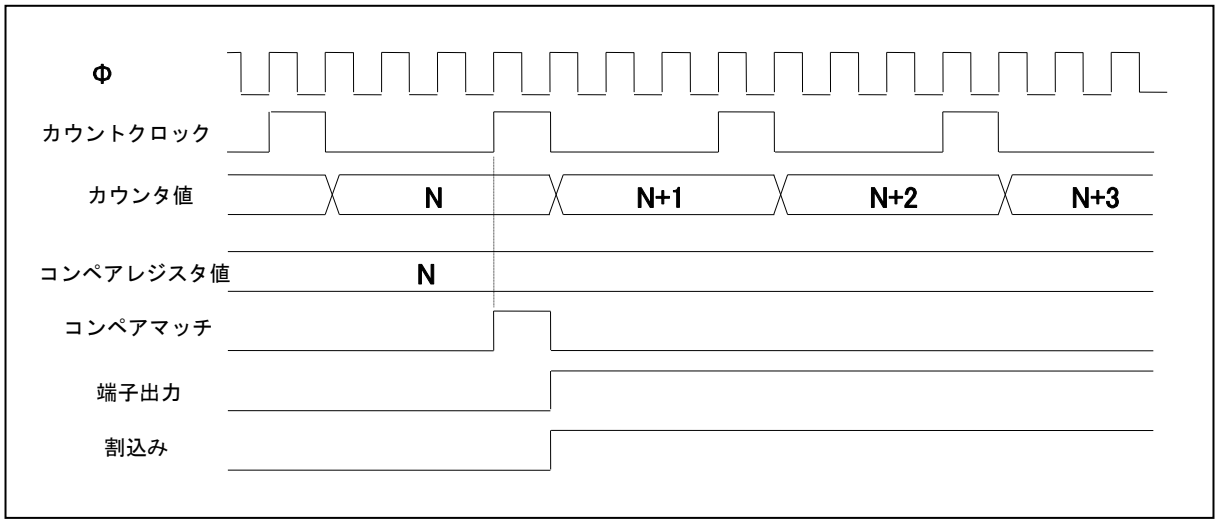
図 22-4 コンペアレジスタ Write タイミング



22.5.3.2 コンペアマッチ、割込み

コンペアマッチおよび割込みについて示します。

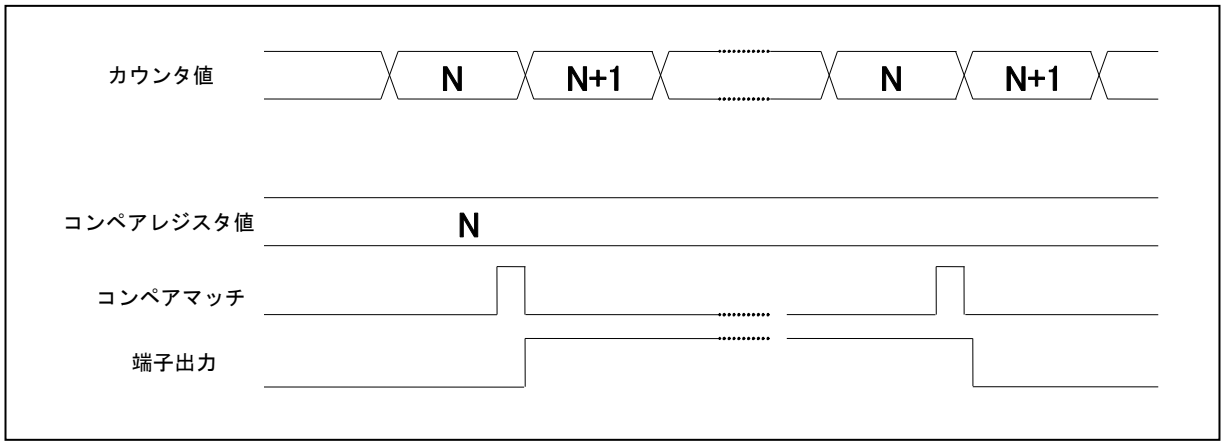
図 22-5 コンペアマッチ, 割込みタイミング



22.5.3.3 端子出力

端子出力について示します。

図 22-6 端子出力タイミング



22.6 設定

アウトプットコンペアの設定について説明します。

図 22-7 アウトプットコンペアを使うために必要な設定

設定	設定レジスタ	設定方法
フリーランタイマの設定	『フリーランタイマ』の章を参照してください。	-
コンペア値の設定	コンペアレジスタ (OCCPx)	22.7.1 参照
コンペアモードの設定	アウトプット制御レジスタ (OCSHxx, OCSLxx)	22.7.2 参照
コンペア動作停止		22.7.3 参照
コンペア端子出力の初期レベルを設定		22.7.4 参照
OCUn 端子を出力に設定*1	各端子をペリフェラル出力に設定してください。 設定方法は『I/O ポート』の章を参照してください。	
フリーランタイマのクリア	タイマ制御レジスタ (TCCSL) 『フリーランタイマ』の章を参照してください。	22.7.6 参照
コンペア動作許可 (起動)	アウトプット制御レジスタ(OCSLxx)	22.7.7 参照

*1: n はチャネル番号

表 22-2 割込みを行うために必要な項目

設定	設定レジスタ	設定方法
アウトプットコンペア割込みベクタ, アウトプットコンペア割込みレベルの設定	『割込み制御(割込みコントローラ)』の章を参照してください。	22.7.8 参照
アウトプットコンペア割込み設定 ・割込み要求のクリア ・割込み要求の許可	アウトプット制御レジスタ(OCSHxx , OCSLxx)	22.7.10 参照

22.7 Q&A

アウトプットコンペアの Q&A について説明します。

22.7.1 コンペア値を設定するには?

22.7.2 コンペアモードを設定するには? (OCU1 での例)

22.7.3 コンペア動作を許可/禁止するには? (OCU0, OCU1 での例)

22.7.4 コンペア端子出力の初期レベルを設定するには? (OCU0, OCU1 での例)

22.7.5 コンペア端子 OCU0-OCU1 を出力に設定するには?

22.7.6 フリーランタイマをクリアするには?

22.7.7 コンペア動作許可するには?

22.7.8 割込み関連レジスタは?

22.7.9 割込みの種類は?

22.7.10 割込みを許可するには?

22.7.11 コンペア値の算出方法は?

22.7.1 コンペア値を設定するには?

コンペア値の設定について説明します。

コンペアレジスタ OCCPx にコンペア値を書き込んでください。

22.7.2 コンペアモードを設定するには? (OCU1 での例)

コンペアモードの設定について説明します。

コンペアモードビット(OCSH01:CMOD)で行ってください。

動作	コンペアモードビット
フリーランタイマ値とコンペアレジスタ 1(OCCP1)が一致した場合に OCU1, OCU3 端子を出力反転させるには	(OCSH01:CMOD)ビットを "0"にする
フリーランタイマ値とコンペアレジスタ 0(OCCP0)またはフリーランタイマ値とコンペアレジスタ 1(OCCP1)のどちらかと一致した場合に OCU1 端子を出力反転させるには	(OCSH01:CMOD)ビットを "1"にする

CMOD ビットとは関係なく、下記のようになります。

- OCU0 出力は、コンペアモードビット (OCSH01:CMOD)の設定とは関係なく、フリーランタイマ値とコンペアレジスタ (OCCP0)が一致した場合に出力反転します。

22.7.3 コンペア動作を許可/禁止するには? (OCU0, OCU1 での例)

コンペア動作の許可/禁止について説明します。

コンペア動作許可ビット(OCSL01:CST0) , (OCSL01:CST1)にて行ってください。

動作	コンペア	コンペア動作許可ビット
コンペア動作を停止 (禁止) にするには	コンペア 0	(OCSL01:CST0) を"0"にする
	コンペア 1	(OCSL01:CST1) を"0"にする
コンペア動作を許可するには	コンペア 0	(OCSL01:CST0) を"1"にする
	コンペア 1	(OCSL01:CST1) を"1"にする

22.7.4 コンペア端子出力の初期レベルを設定するには? (OCU0, OCU1 での例)

コンペア端子出力の初期レベルの設定について説明します。

コンペア端子出力指定ビット(OCSH01:OTD0), (OCSH01:OTD1)で設定してください。

動作	コンペア端子出力指定ビット
コンペア 0 端子を"L"にするには	(OCSH01:OTD0) を"0"にする
コンペア 0 端子を"H"にするには	(OCSH01:OTD0) を"1"にする
コンペア 1 端子を"L"にするには	(OCSH01:OTD1) を"0"にする
コンペア 1 端子を"H"にするには	(OCSH01:OTD1) を"1"にする

22.7.5 コンペア端子 OCU0-OCU1 を出力に設定するには?

コンペア端子 OCU0-OCU1 の出力設定について説明します。

端子をペリフェラル出力に設定してください。設定方法は『I/O ポート』の章を参照してください。

22.7.6 フリーランタイマをクリアするには?

フリーランタイマのクリアについて説明します。

利用するフリーランタイマのクリアビット (TCCSL:SCLR) で設定してください。

動作	クリアビット(SCLR)
フリーランタイマをクリアするには	"1"を書き込む

ほかの方法については『フリーランタイマ』の章を参照してください。

22.7.7 コンペア動作許可するには?

コンペア動作許可について説明します。

コンペア動作許可ビット(OCSL01:CST0, OCSL01:CST1, OCSL23:CST2, OCSL23:CST3)にて行ってください。

「22.7.3 コンペア動作を許可/禁止するには? (OCU0, OCU1 での例)」を参照してください。

22.7.8 割込み関連レジスタは?

割込み関連レジスタについて説明します。

アウトプットコンペア割込みベクタ, アウトプットコンペア割込みレベルも設定します。

アウトプットコンペア番号, 割込みレベル, ベクタの関係は下表のとおりです。

割込みレベル, 割込みベクタの詳細については『割込み制御(割込みコントローラ)』の章を参照してください。

番号	割込みベクタ (デフォルト)	割込みレベル設定ビット (ICR[4:0])
アウトプットコンペア 0/1/6/7/10/11	#58 Address: 0FFF14H	割込みレベルレジスタ(ICR42) Address: 0046AH
アウトプットコンペア 2/3/4/5/8/9	#59 Address: 0FFF10H	割込みレベルレジスタ(ICR43) Address: 0046BH

割込み要求フラグ(OCSLxy:IOPx, OCSLxy:IOPy)は自動的にクリアされません。割込み処理から復帰する前にソフトウェアにてそれぞれのビットに"0"を書き込みクリアしてください。(x, y はチャネル番号)

22.7.9 割込みの種類は?

割込みの種類について説明します。

割込みは 1 種類のみです。コンペア一致で発生します。

22.7.10 割込みを許可するには?

割込みの許可について説明します。

割込み許可の設定は、割込み要求許可ビット(OCSLxy:IOEx, OCSLxy:IOEy) にて行ってください。(x, y はチャネル番号)

動作	割込み要求許可ビット (OCSLxy:IOEx, OCSLxy:IOEy)
割込み禁止	"0"にする
割込み許可	"1"にする

割込み要求のクリアは, 割込み要求フラグビット(OCSLxy:IOPx, OCSLxy:IOPy)にて行ってください。(x, y はチャネル番号)

動作	割込み要求フラグビット (OCSLxy:IOPx, OCSLxy:IOPy)
割込み要求クリア	"0"を書き込む

22.7.11 コンペア値の算出方法は?

コンペア値の算出方法について説明します。

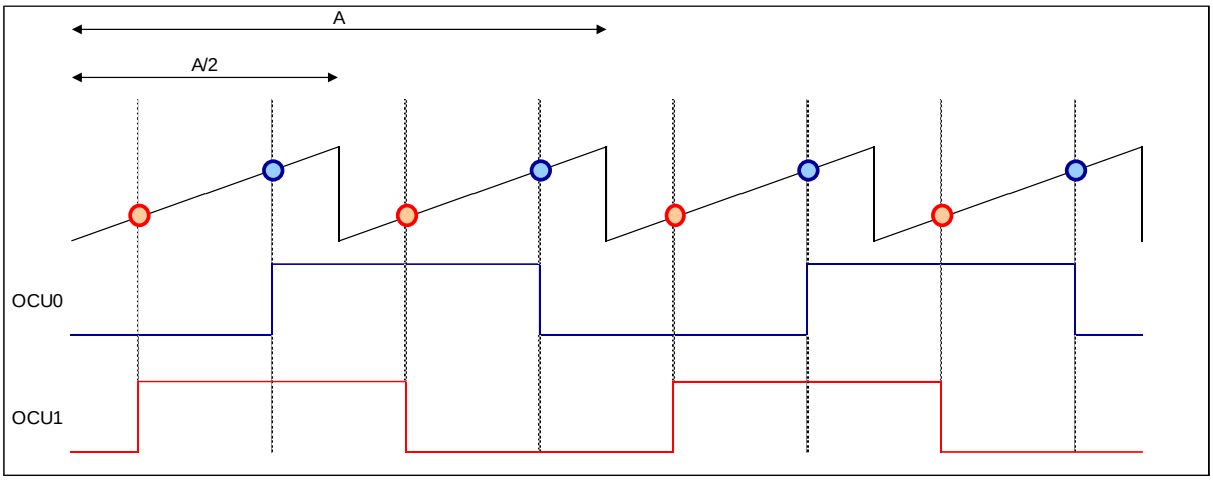
22.7.11.1. トグル出力パルス

22.7.11.2. PWM 出力

22.7.11.1 トグル出力パルス

トグル出力パルスについて説明します。

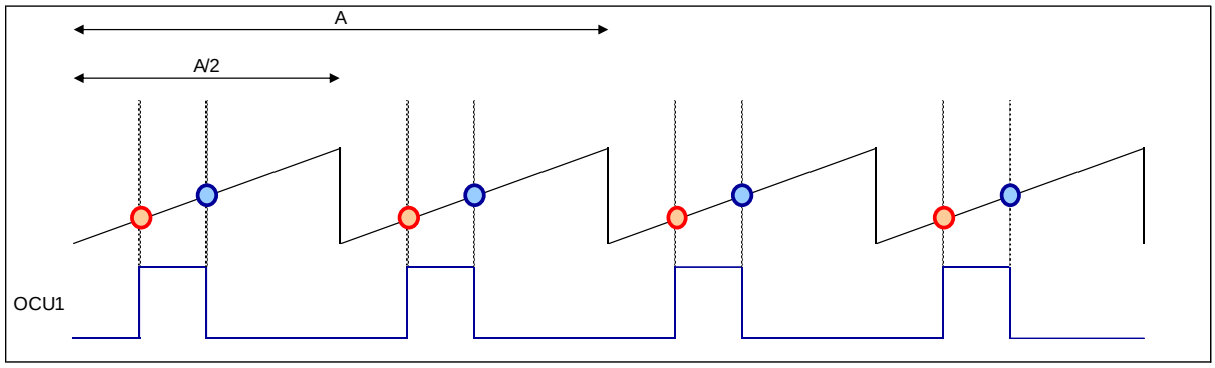
(例)OCU0, OCU1 使用、周期 : A, 位相差 1/4 の 2 相パルスを出すには
FreeRunTimer.CPCLR = (A/2) - 1
Output Compare.OCCP0 = (A/2) × (3/4) - 1
Output Compare.OCCP1 = (A/2) × (1/4) - 1
Output Compare.OCSH01.CMOD = 0
に設定します。



22.7.11.2 PWM 出力

PWM 出力について説明します。

(例)OCU0, OCU1 使用、周期：A, デューティ 1/4 の PWM を出すには
FreeRunTimer.CPCLR = (A/2) - 1
Output Compare.OCCP0 = (A/2) × (1/2) - 1
Output Compare.OCCP1 = (A/2) × (1/4) - 1
Output Compare.OCSH01.CMOD = 1
に設定します。



22.8 サンプルプログラム

サンプルプログラムについて説明します。

設定手順例 1

.2 チャネル独立出力
コンペア動作(7FFF, BFFF)
割込み発生コンペアクリアなし

1. 初期設定

・フリーランタイマ ch.1 の制御　　レジスタ名. ビット名

制御レジスタの設定 クロック選択≫	TCCSH1,TCCSL11 .ECKE
コンペア割込み要求フラグ≫ コンペア割込み要求許可≫	.ICLR .ICRE
カウント動作≫	.STOP
TCDT クリア≫ カウントクロック≫	.SCLR .CLK3-0
タイマデータ値の設定	TCDT1

・ポート　　レジスタ名. ビット名

ポート OCU0 出力設定 ポート OCU1 出力設定	『I/O ポート』の章参照

・アウトプットコンペアの制御　　レジスタ名. ビット名

フリーランタイマ選択 制御レジスタの設定	OCFS01 OCSH01,OCSL01
端子出力レベル反転動作≫	.CMOD
端子出力レベル指定≫ 割込み要求フラグ≫ 割込み要求許可≫	.OTD1,OTD0 .IOP1,IOP0 .IOE1,IOE0
動作許可設定≫	.CST1,CST0
コンペア値 ch0 の設定 コンペア値 ch1 の設定	OCCP0 OCCP1

・割込み関連　　レジスタ名. ビット名

割込みレベルの設定	ICR42 ICR43
I フラグの設定	(CCR)

2. 起動

・アウトプットコンペア起動　　レジスタ名. ビット名

割込み制御	OCSL01.IOE1
コンペア動作起動	OCSL01.CST1 OCSL01.CST0

・フリーランタイマ ch1 起動　　レジスタ名. ビット名

カウント動作起動	TCCSL1.STOP

3. 割込み

・割込み処理　　レジスタ名. ビット名

割込み要求フラグのクリア	OCSL01.IOP0
(任意の処理)	
.....	
割込み要求フラグのクリア	OCSL01.IOP1
(任意の処理)	
.....	

4. 割込みベクタ

・ベクタテーブルの設定

(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み(割込みコントローラ)』の章を参照してください。

プログラム例 1

```
void OUTPUT01_sample(void)
{
        freerun1_initial();
        OUTPUT01_initial();
        OUTPUT01_start();
        freerun1_start();
}

void freerun1_initial(void)
{
        IO_TCCS1.word = 0x0041;         /*  設定値=0000_0000_0100_0001 */
                                        /* bit15 = 0        ECKE  内部クロックソース  */
                                        /* bit14 -10 =0     予約ビット  */
                                        /* bit9 = 0         ICLR 割込みフラグクリア  */
                                        /* bit8 = 0         ICLR 割込み禁止  */
                                        /* bit7 = 0         予約ビット  */
                                        /* bit6 = 1         STOP カウント禁止*/
                                        /* bit5 = 0         予約ビット  */
                                        /* bit4 = 0         SCLR フリーランタイマ値の初期化(なし) */
                                        /* bit3-0 = 0001    CLK3-0 カウントクロック PCLK/2=32MHz/2 */
        IO_TCDT1 = 0x0000;              /*  タイマデータ値の初期化  */
}

void OUTPUT01_initial(void)
{
        PORT_SETTING_OCU0_OUT(); /* OCU0  端子をペリフェラル入力に設定してください。  */
        PORT_SETTING_OCU1_OUT(); /* OCU1  端子をペリフェラル入力に設定してください。  */

        IO_OCFS01.hword = 0x0003;       /*  フリーランタイマ 1 を選択  */
        IO_OCS01.hword = 0xEC0C;        /*  設定値=1110_1100_0000_1100 */
                                        /* bit15-13 = 111     未定義ビット  */
                                        /* bit12 = 0          CMOD ch.0,ch.1 レベル反転  */
                                        /* bit11-10 = 11      未定義ビット  */
                                        /* bit9-8 = 00        OTD1,OTD0 コンペア端子出力 0   */
                                        /* bit7-6 = 00        IOP1,IOP0 アウトプットコンペアー致なし  */
                                        /* bit5-4 = 00        IOE1,IOE0  アウトプットコンペア割込み禁止  */
                                        /* bit3-2 = 11        未定義ビット*/
                                        /* bit1-0 = 00        CST1,CST0 コンペア動作禁止  */
        IO_OCCP0 = BFFF                 /*  コンペアレジスタ  ch.0 の設定  */
        IO_OCCP1 = 7FFF                 /*  コンペアレジスタ  ch.1 の設定  */

        IO_ICR[42].byte = 0x10;         /*  アウトプットコンペア ch.0  割込みレベル設定(  値は任意) */
        IO_ICR[43].byte = 0x10;         /*  アウトプットコンペア ch.1  割込みレベル設定(  値は任意) */
        __EI();                         /*  割込み許可  */
}

void OUTPUT01_start(void)
{
        IO_OCS01.hword = 0xEC3C;        /* bit5-4 = 11 IOE1,IOE0 アウトプットコンペア割込み許可  */
        IO_OCS01.hword = 0xEC3F;        /* bit1-0 = 11 CST1,CST0 コンペア動作許可  */
}

void freerun1_start(void)
{
        IO_TCCSLL1.bit.STOP = 0;        /* bit4 = 0 STOP  カウント許可  */
}

__interrupt void INPUT0_int(void)
{
        IO_OCSL01.byte & = 0xBF;        /* bit6 = 0 IOP0 割込みフラグのクリア  */
        ……
}
__interrupt void INPUT0_int(void)
{
        IO_OCSL01.byte & = 0x7F;        /* bit7 = 0 IOP1 割込みフラグのクリア  */
        ……
}

ベクタテーブルにて割込みルーチンの指定が必要
#pragma intvect OUTPUT0_int 58
#pragma intvect OUTPUT1_int 59
```

設定手順例 2

.2 組のコンペア
ch1 の出力
コンペア動作(7FFF, BFFF)
大きい方のコンペア値を周期としてコンペアクリアする。
割込み発生

1. 初期設定

・フリーランタイマ ch.1 の制御

	レジスタ名. ビット名
制御レジスタの設定	TCCSH1,TCCSL1
クロック選択≫	.ECKE
コンペア割込み要求フラグ≫	.ICLR
コンペア割込み要求許可≫	.ICRE
カウント動作≫	.STOP
TCDT クリア≫	.SCLR
カウントクロック≫	.CLK3-0
タイマデータ値の設定	TCDT1

・ポート

	レジスタ名. ビット名
ポートの OCU1 出力設定	『 I/O ポート』の章参照

・アウトプットコンペアの制御

	レジスタ名. ビット名
フリーランタイマ選択	OCFS01
制御レジスタの設定	OCSH01,OCSL01
端子出力レベル反転動作≫	.CMOD
端子出力レベル指定≫	.OTD1,OTD0
割込み要求フラグ≫	.IOP1,IOP0
割込み要求許可≫	.IOE1,IOE0
動作許可設定≫	.CST1,CST0
コンペア値 ch0 の設定	OCCP0
コンペア値 ch1 の設定	OCCP1

・割込み関連

	レジスタ名. ビット名
割込みレベルの設定	ICR42 ICR43
I フラグの設定	(CCR)

2. 起動

・アウトプットコンペア起動

	レジスタ名. ビット名
割込み制御	OCSL01.IOE1
コンペア動作起動	OCSL01.CST1 OCSL01.CST0

・フリーランタイマ ch1 起動

	レジスタ名. ビット名
カウント動作起動	TCCS1.STOP

3. 割込み

・割込み処理

	レジスタ名. ビット名
割込み要求フラグのクリア	OCSL01.IOP0
(任意の処理)	
.....	

4. 割込みベクタ
・ベクタテーブルの設定

(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み(割込みコントローラ)』の章を参照してください。

プログラム例 2

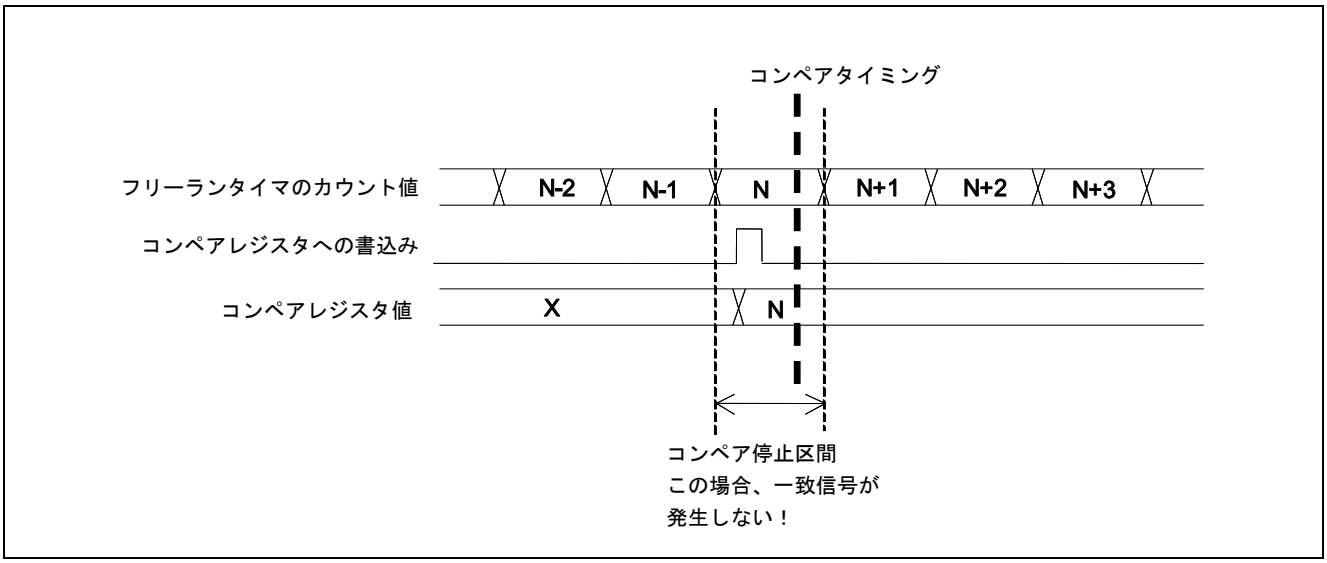
```
void OUTPUT23_sample(void)
{
        freerun1_initial();
        OUTPUT01_initial();
        OUTPUT01_start();
        freerun1_start();
}

void freerun1_initial(void)
{
        IO_TCCS1.word = 0x0041;         /* 設定値=0000_0000_0100_0001 */
                                        /* bit15 = 0        ECKE 内部クロックソース */
                                        /* bit14 -10 =0     予約ビット */
                                        /* bit9 = 0         ICLR 割込みフラグクリア */
                                        /* bit8 = 0         ICLR 割込み禁止 */
                                        /* bit7 = 0         予約ビット */
                                        /* bit6 = 1         STOP カウント禁止*/
                                        /* bit5 = 0         予約ビット */
                                        /* bit4 = 0         SCLR フリーランタイマ値の初期化(なし) */
                                        /* bit3-0 = 0001    CLK3-0 カウントクロック PCLK/2=32MHz/2 */
        IO_TCDT1 = 0x0000;              /* タイマデータ値の初期化 */
}

void OUTPUT01_initial(void)
{
        PORT_SETTING_OCU0_OUT(); /* OCU1 端子をペリフェラル入力に設定してください。 */

        IO_OCFS01.hword = 0x0003;       /* フリーランタイマ 1 を選択 */
        IO_OCS01.hword = 0xEC0C;        /* 設定値=1110_1100_0000_1100 */
                                        /* bit15-13 = 111    未定義ビット */
                                        /* bit12 = 0         CMOD ch.0,ch.1 レベル反転 */
                                        /* bit11-10 = 11     未定義ビット */
                                        /* bit9-8 = 00       OTD1,OTD0 コンペア端子出力 0 */
                                        /* bit7-6 = 00       IOP1,IOP0 アウトプットコンペアー致なし */
                                        /* bit5-4 = 00       IOE1,IOE0 アウトプットコンペア割込み禁止 */
                                        /* bit3-2 = 11       未定義ビット*/
                                        /* bit1-0 = 00       CST1,CST0 コンペア動作禁止 */
        IO_OCCP0 = BFFF                 /* コンペアレジスタ ch.0 の設定 */
        IO_OCCP1 = 7FFF                 /* コンペアレジスタ ch.1 の設定 */

        IO_ICR[42].byte = 0x10;         /* アウトプットコンペア ch.0 割込みレベル設定( 値は任意) */
        IO_ICR[43].byte = 0x10;         /* アウトプットコンペア ch.1 割込みレベル設定( 値は任意) */
        __EI();                         /* 割込み許可 */
}

void OUTPUT01_start(void)
{
        IO_OCS01.hword = 0xEC3C;        /* bit5-4 = 11 IOE1,IOE0 アウトプットコンペア割込み許可 */
        IO_OCS01.hword = 0xEC3F;        /* bit1-0 = 11 CST1,CST0 コンペア動作許可 */
}

void freerun1_start(void)
{
        IO_TCCS1.bit.STOP = 0;          /* bit4 = 0 STOP カウント許可 */
}


__interrupt void INPUT0_int(void)
{
        IO_OCSL01.byte & = 0xBF;    /* bit6 = 0 IOP0 割込みフラグのクリア */
        ……
        IO_OCSL01.byte & = 0x7F;    /* bit7 = 0 IOP1 割込みフラグのクリア */
        ……
}


ベクタテーブルにて割込みルーチンの指定が必要
#pragma intvect OUTPUT1_int 59
```

22.9 注意事項

アウトプットコンペアの注意事項について説明します。

● コンペア動作中のコンペア停止区間について

下記に示すようにコンペアレジスタにコンペア値を書き込んだ直後の 1 カウント分は、コンペア動作が行われません。



- CMOD="1"でかつ OCCP0＝OCCP1, OCCP2 = OCCP3 の設定の場合、コンペア一致が発生するとポートは 1 回のみ反転します。(ch4～11 でも同様)
- コンペアの端子 (OCU0, OCU1, OCU2, OCU3,・・・・)の出力レベルの指定をするときには、コンペア動作を停止させてから行ってください。
- アウトプットコンペアは、フリーランタイマと同期しているため、フリーランタイマを停止させるとコンペア動作も停止します。
- コンペアモードビットを CMOD = "1"にした場合でも、割込み動作は OCU0, OCU1, OCU2, OCU3,・・・・それぞれ独立で発生します。
- フリーランタイマをアウトプットコンペアのコンペアデータとして使用する場合、フリーランタイマのクロック周波数(TCCSL:CLK[3:0]には、"0000_B"($1/F_{PCLK}$)の設定は禁止となります。

● リードモディファイライト

割込み要求フラグビット(IOP0),(IOP1),(IOP2), (IOP3),・・・・をリードモディファイライトで読み出すと"1"が読み出せます。

23. アップダウンカウンタ

アップダウンカウンタについて説明します。

23.1 概要

アップダウンカウンタの概要について説明します。

アップダウンカウンタは, 設定によってカウントアップ/ダウンするカウンタです。

16 ビットアップダウンカウンタの下位バイトのみを使用して、8 ビットアップダウンカウンタとして使用することもできます。

8 ビットアップダウンカウンタ時は 00_H～FF_H の範囲で、16 ビットアップダウンカウンタ時は 0000_H～$FFFF_H$ の範囲でカウントできます。

本製品は、16 ビットアップダウンカウンタを 2 チャネル内蔵しています。ただし、8 ビットアップダウンカウンタとして使用できるのは下位バイトのみになるため、8 ビット時も、16 ビット時も使用できるチャネルは合計で 2 チャネルになります。

23.2 特長

アップダウンカウンタの特長について説明します。

- カウンタモード：次の2モードから選択できます。
  - 8ビットアップダウンカウンタ (8ビットモード)
  - 16ビットアップダウンカウンタ (16ビットモード)
- 動作モード：次の3モード (4種類) から選択できます。
  - **タイマモード**

    カウント用クロックに同期してカウントダウンします。

    カウント用クロックは周辺クロック (PCLK) をプリスケーラで2分周/8分周して生成された内部クロックを使用します。
  - **アップダウンカウントモード**

    2本の外部信号入力端子から入力される信号をカウントアップ/カウントダウンします。カウントするエッジを、立上りエッジ, 立下りエッジ, 両エッジの中から選択できます。
  - **位相差カウントモード**

    2本の外部信号入力端子から入力される信号の位相差をカウントアップ/カウントダウンします。

    位相差カウントモードは、モータなどのエンコーダのカウントに適しています。エンコーダのA相, B相, Z相の出力をそれぞれ入力することにより、回転角度や回転数などを高い精度で容易にカウントできます。

    位相差カウントモードには2逓倍モードと4逓倍モードがあり、それぞれカウント方法が異なります。

アップダウンカウンタの動作モードを表 23-1 に示します。

表 23-1 アップダウンカウンタの動作モード

動作モード	カウントタイミング	カウント方向
タイマモード	内部クロック	カウントダウン
アップダウンカウントモード	外部クロック	カウントアップ/カウントダウン
位相差カウントモード (2 逓倍/4 逓倍)	外部信号入力端子からの入力信号の位相	カウントアップ/カウントダウン

- リロードコンペア機能：次の3種類から選択できます。
  - **コンペア機能**

    あらかじめ設定した値までカウントすると、カウンタの値をクリアして、カウントを続けます。
  - **リロー機能**

    アンダフローが発生すると、リロード値をロードしてカウントを続けます。
  - **リロードコンペア機能**

    コンペア機能とリロード機能を組み合わせて使用できます。
- カウント方向：直前のカウント方向 (カウントアップ/カウントダウン) を確認できます。
- 割込み要求：次の場合に割込み要求を発生できます。
  - カウント方向が反転したとき
  - カウンタの値があらかじめ設定した値と一致したとき

- ☐ オーバフローが発生したとき
- ☐ アンダフロー (リロード) が発生したとき

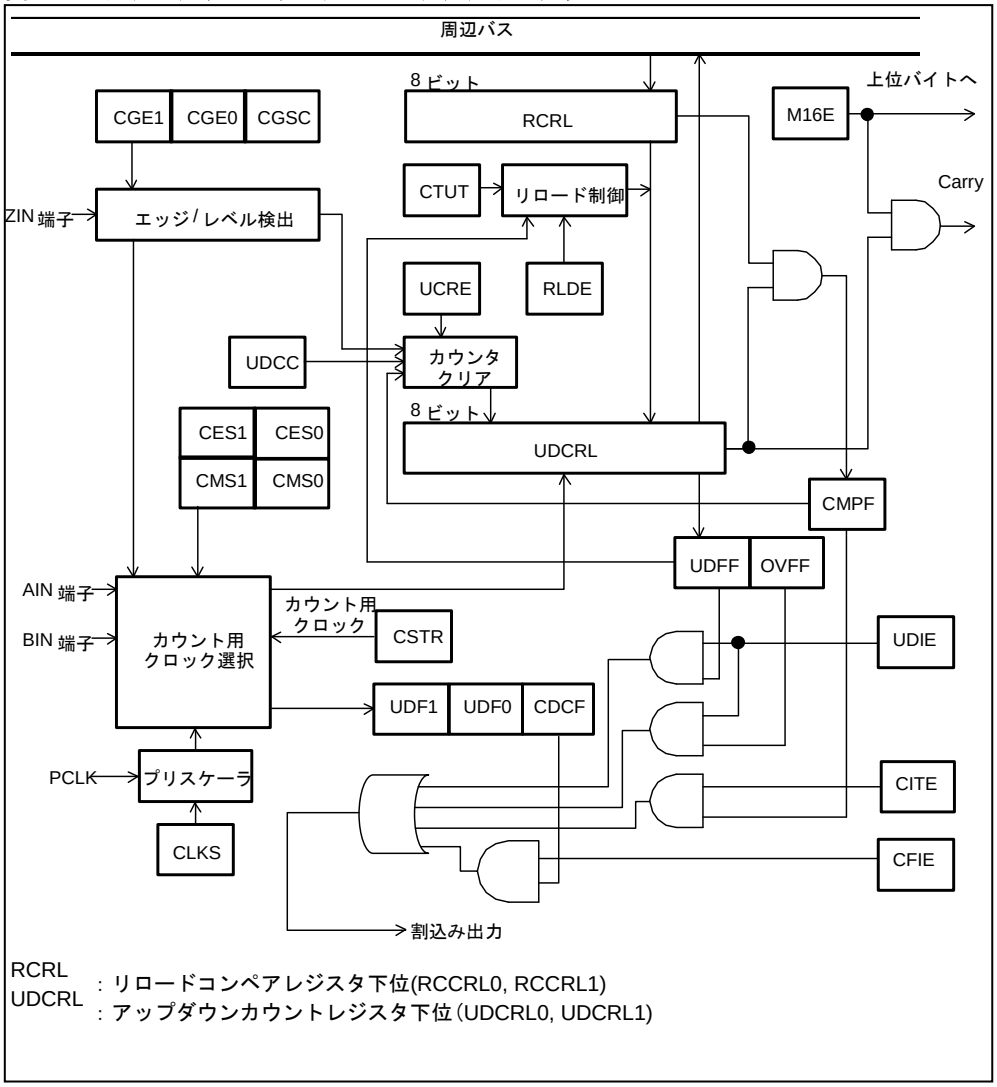
23.3 構成

アップダウンカウンタの構成を示します。

■ アップダウンカウンタのブロックダイヤグラム

アップダウンカウンタのブロックダイヤグラムを ch.0 を例にとって図 23-1 に示します。

図 23-1 アップダウンカウンタのブロックダイヤグラム



■ リロードコンペアレジスタ (RCR)

アップダウンカウンタのリロード値およびコンペア値を設定するレジスタです。

次のように上位8ビットと下位8ビットに分かれています。

8ビットモードで使用する場合は，下位側を使用します。

- □ リロードコンペアレジスタ上位 (RCRH)
- □ リロードコンペアレジスタ下位 (RCRL)

- アップダウンカウントレジスタ (UDCR)
  アップダウンカウンタのカウンタとして動作するレジスタです。
  次のように上位8ビットと下位8ビットに分かれています。
  8ビットモードで使用する場合は、下位側を使用します。
  - □ アップダウンカウントレジスタ上位 (UDCRH)
  - □ アップダウンカウントレジスタ下位 (UDCRL)
- カウンタコントロールレジスタ (CCR)
  アップダウンカウンタを制御するレジスタです。
- カウンタステータスレジスタ (CSR)
  アップダウンカウンタの状態を確認したり、割込み要求の制御をしたりするレジスタです。
- カウント用クロック選択回路
  アップダウンカウンタのカウント用クロックを選択する回路です。
- プリスケーラ
  アップダウンカウンタをタイマモードで使用するときに、周辺クロック (PCLK) の分周比を選択します。

■ クロック

アップダウンカウンタで使用するクロックを表 23-2 に示します。

表 23-2 アップダウンカウンタで使用するクロック

クロック名	内容	備考
動作クロック	周辺クロック (PCLK)	-
カウント用クロック	内部クロック (周辺クロック)	周辺クロック(PCLK)を分周して生成
	外部端子からの入力をカウント	AIN 端子, BIN 端子から入力

23.4 レジスタ

アップダウンカウンタで使用するレジスタの構成と機能について説明します。

■ 端子とチャネルの対応

チャネルと端子の対応を表 23-3 に示します。

表 23-3 チャネルと端子の対応

チャネル番号	外部信号入力端子		
0	AIN0_0, AIN0_1, AIN0_2	BIN0_0, BIN0_1, BIN0_2	ZIN0_0, ZIN0_1, ZIN0_2
1	AIN1_0, AIN1_1	BIN1_0, BIN1_1	ZIN1_0, ZIN1_1

ch.0, ch.1 は、IO リロケーション機能により、使用する外部端子を選択します。

■ レジスタマップ

アップダウンカウンタのレジスタマップを表 23-4 に示します。

表 23-4 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0F70	RCRH0	RCRL0	UDCRH0	UDCRL0	リロードコンペアレジスタ上位 0 リロードコンペアレジスタ下位 0 アップダウンカウントレジスタ上位 0 アップダウンカウントレジスタ下位 0
0x0F74	CCR0		予約	CSR0	カウンタコントロールレジスタ 0 カウンタステータスレジスタ 0
0x0F80	RCRH1	RCRL1	UDCRH1	UDCRL1	リロードコンペアレジスタ上位 1 リロードコンペアレジスタ下位 1 アップダウンカウントレジスタ上位 1 アップダウンカウントレジスタ下位 1
0x0F84	CCR1		予約	CSR1	カウンタコントロールレジスタ 1 カウンタステータスレジスタ 1

23.4.1 リロードコンペアレジスタ (RCR0, RCR1)

リロードコンペアレジスタのビット構成について示します。

アップダウンカウンタのリロード値およびコンペア値を設定するレジスタです。
リロード値はカウントダウン時にカウントを開始する値、コンペア値はカウントアップ時にカウントされた値と比較する値 (ここまで数えるという値) です。リロード値とコンペア値は同一です。

■ RCRH0： アドレス 0F70H (アクセス：ハーフワード，ワード)

■ RCRH1： アドレス 0F80H (アクセス：ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	D15	D14	D13	D12	D11	D10	D9	D8
初期値	X	X	X	X	X	X	X	X
属性	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W

■ RCRL0： アドレス 0F71H (アクセス：バイト，ハーフワード，ワード)

■ RCRL1： アドレス 0F81H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	X	X	X	X	X	X	X	X
属性	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W

このレジスタは、次のように上位バイトと下位バイトに分かれています。

- リロードコンペアレジスタ上位 (RCRH0, RCRH1)
- リロードコンペアレジスタ下位 (RCRL0, RCRL1)

16 ビットモード時は、両方の値が使用され、8 ビットモード時は、下位バイトの値が使用されます。
このレジスタに書き込んだ値をアップダウンカウントレジスタ (UDCR) に転送することで、アップダウンカウンタは、"0000H" (8 ビット時は"00H") ～このレジスタに設定した値の範囲でカウントを行います。

＜注意事項＞

- カウンタコントロールレジスタ (CCR) の CTUT ビットに"1"を書き込むと、このレジスタに設定した値をアップダウンカウントレジスタ (UDCR) に転送できます。ただし、カウンタコントロールレジスタ (CCR) の CTUT ビットは、アップダウンカウンタの停止中に書き込んでください。
- カウンタコントロールレジスタ(CCR)の M16E ビットで 16 ビットモードを設定した場合 (M16E=1)、このレジスタは必ずハーフワードで書き込んでください。
- カウンタコントロールレジスタ (CCR) の M16E ビットで 8 ビットモードを設定した場合 (M16E=0)、必ずリロードコンペアレジスタ下位 (RCRL) にバイトで書き込んでください。

23.4.2 アップダウンカウントレジスタ (UDCR0, UDCR1)

アップダウンカウントレジスタのビット構成について示します。

アップダウンカウンタのカウンタとして動作するレジスタです。このレジスタを読み出すとカウンタの値を確認できます。

■ UDCRH0：アドレス 0F72H (アクセス：ハーフワード，ワード)

■ UDCRH1：アドレス 0F82H (アクセス：ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	D15	D14	D13	D12	D11	D10	D9	D8
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ UDCRL0：アドレス 0F73H (アクセス：バイト，ハーフワード，ワード)

■ UDCRL1：アドレス 0F83H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

このレジスタは，次のように上位バイトと下位バイトに分かれています。

- アップダウンカウントレジスタ上位 (UDCRH0, UDCRH1)
- アップダウンカウントレジスタ下位 (UDCRL0, UDCRL1)

8 ビットモード時は，上位バイトの値は無効です。
アップダウンカウントレジスタ下位 (UDCRL) の値を読み出してください。

＜注意事項＞

- このレジスタは読出し専用です。このレジスタに値を設定する場合は、次の手順でリロードコンペアレジスタ (RCR) の値をこのレジスタに転送してください。
  1. リロードコンペアレジスタ (RCR) に値を書き込む
  2. カウンタステータスレジスタ (CSR) の CSTR ビットに"0"を書き込む
  3. カウンタコントロールレジスタ (CCR) の CTUT ビットに"1"を書き込む
- カウンタコントロールレジスタ(CCR)の M16E ビットで 16 ビットモードを設定した場合 (M16E=1)、このレジスタは必ずハーフワードで読み出してください。
- カウンタコントロールレジスタ(CCR)の M16E ビットで 8 ビットモードを設定した場合 (M16E=0)、アップダウンカウントレジスタ下位 (UDCRL) の値を読み出してください。

23.4.3 カウンタコントロールレジスタ (CCR0, CCR1)

カウンタコントロールレジスタのビット構成について示します。

アップダウンカウンタの動作を制御するレジスタです。

■ CCR0： アドレス 0F74H (アクセス：バイト，ハーフワード)

■ CCR1： アドレス 0F84H (アクセス：バイト，ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	M16E	CDCF	CFIE	CLKS	CMS1	CMS0	CES1	CES0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R(RM1),W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	CTUT	UCRE	RLDE	UDCC	CGSC	CGE1	CGE0
初期値	0	0	0	0	1	0	0	0
属性	R0,W0	R0,W	R/W	R/W	R1,W	R/W	R/W	R/W

[bit15] M16E : 16 ビットモード選択ビット

アップダウンカウンタを 8 ビットで使用するか、16 ビットで使用するかを選択します。

書込み値	説明
0	8 ビットモード (1 チャネル) で使用します。
1	16 ビットモード (1 チャネル) で使用します。

[bit14] CDCF : カウント方向転換フラグビット

カウント方向が、カウントダウンからカウントアップ、またはカウントアップからカウントダウンに 1 回以上反転したことを示します。

このビットが"1"のときに CFIE ビットが"1"に設定されていると、カウント方向転換割込み要求が発生します。

CDCF	読出し時	書込み時
0	カウント方向は反転していません。	このビットを"0"にクリアします。
1	カウント方向が 1 回以上反転しました。	無視されます。

＜注意事項＞

- デバイスのリセットが発生した場合、カウント方向はカウントダウンに設定されます。そのため，リセット直後にカウントアップが行われると、このビットが"1"に変わります。
- カウント方向が短期間で連続して変化した場合は、カウント方向が元に戻り、カウンタステータスレジスタ (CSR) の UDF1, UDF0 ビットが変化しない場合があります。

[bit13] CFIE : カウント方向転換割込み許可ビット

カウント方向が反転したとき (CDCF=1) に、カウント方向転換割込み要求を発生させるかどうかを設定します。

書込み値	説明
0	カウント方向転換割込み要求の発生を禁止します。
1	カウント方向転換割込み要求の発生を許可します。

[bit12] CLKS : 内部クロック分周選択ビット

タイマモード選択時に、このビットで設定した分周比で分周された周辺クロック (PCLK) をカウント用クロックとして使用します。

書込み値	説明
0	周辺クロック (PCLK) の 2 分周
1	周辺クロック (PCLK) の 8 分周

＜注意事項＞

このビットは、CMS1, CMS0 ビットで動作モードをタイマモードに設定 (CMS1, CMS0=00) した場合のみ有効です。そのほかの動作モードを選択しているときは、このビットの設定は無視されます。

[bit11, bit10] CMS1, CMS0 : 動作モード選択ビット

アップダウンカウンタの動作モードを次の中から選択します。

- タイマモード
  カウント用クロックに同期してカウントダウンします。
- アップダウンカウントモード
  2 本の外部信号入力端子からの入力信号をカウントアップ/カウントダウンします。
- 位相差カウントモード
  2 本の外部信号入力端子の位相差をカウントアップ/カウントダウンします。位相差カウントモードには 2 逓倍モードと 4 逓倍モードがあり、それぞれカウント方法が異なります。

CMS1	CMS0	動作モード
0	0	タイマモード
0	1	アップダウンカウントモード
1	0	位相差カウントモード (2 逓倍)
1	1	位相差カウントモード (4 逓倍)

[bit9, bit8] CES1, CES0 : カウント用クロックエッジ選択ビット

AIN 端子および BIN 端子の検出エッジを選択します。

アップダウンカウントモード選択時に、このビットで選択したエッジが検出されるたびに、カウント動作が行われます。

CES1	CES0	検出エッジ
0	0	エッジ検出禁止
0	1	立下りエッジ
1	0	立上りエッジ
1	1	両エッジ

＜注意事項＞

このビットは、CMS1, CMS0 ビットで動作モードをアップダウンカウントモードに設定 (CMS1, CMS0=01) した場合のみ有効です。そのほかの動作モードを選択しているときは、このビットの設定は無視されます。

[bit7] 予約ビット

書込み時	必ず"0"を書き込んでください。
読出し時	"0"が読み出されます。

[bit6] CTUT : カウンタライトビット

リロードコンペアレジスタ (RCR) に設定した値をアップダウンカウントレジスタ (UDCR) に転送します。

CTUT	読出し時	書込み時
0	"0"が読み出されます。	無視されます。
1		値を転送します。

＜注意事項＞

このビットに"1"を書き込んだ時点で、リロードコンペアレジスタ (RCR) の値が転送されるため、カウンタステータスレジスタ (CSR) の CSTR ビットが"1"のとき (カウンタの動作中) は、このビットを"1"に書き換えないでください。

[bit5] UCRE : カウンタクリア許可ビット
コンペア機能の使用を許可/禁止します。

コンペア機能とは、カウンタの値が、リロードコンペアレジスタ (RCR) に設定した値と一致したときに、カウンタの値を"0000_H"にクリアし、カウントを続ける機能です。

書込み値	説明
0	コンペア機能の使用を禁止します。
1	コンペア機能の使用を許可します。

＜注意事項＞
このビットで制御できるのは、コンペア機能を利用してのクリア動作のみです。
次のクリア動作は、このビットでは制御できません。

- 本デバイスがリセットされたことによるクリア
- ZIN 端子からの有効エッジ入力によるクリア (CGSC ビット=0 のとき)
- UDCC ビットに"0"を書き込むことによるクリア (ソフトウェアによるクリア)

[bit4] RLDE : リロード許可ビット
リロード機能の使用を許可/禁止します。

リロード機能とは、カウントダウン時にカウンタがアンダフローすると、リロードコンペアレジスタ (RCR) に設定した値をカウンタにリロードして、カウントを続ける機能です。

書込み値	説明
0	リロード機能の使用を禁止します。
1	リロード機能の使用を許可します。

[bit3] UDCC : カウンタクリアビット
カウンタの値を"0000_H"にクリアします。

UDCC	読出し時	書込み時
0	"1"が読み出されます。	クリアします。
1		無視されます。

[bit2] CGSC : カウンタクリア/ゲート選択ビット
ZIN 端子の機能を次の中から選択します。

- カウンタクリア機能
  ZIN 端子から有効エッジが入力されたときに，カウンタの値を"0000_H"にクリアします。
- ゲート機能
  ZIN 端子から有効レベルが入力されている間だけ，カウンタが動作します。

書込み値	説明
0	カウンタクリア機能
1	ゲート機能

＜注意事項＞
ZIN 端子は、このビットと CGE1, CGE0 ビットの設定を組み合わせることで機能します。必ず、CGE1, CGE0 ビットも設定してください。

[bit1, bit0] CGE1, CGE0 : エッジ/レベル選択ビット
ZIN 端子の有効エッジ/有効レベルを選択します。CGSC ビット設定によって、このビットの意味は異なります。

- CGSC ビットでカウンタクリア機能を選択した場合 (CGSC=0)
  有効エッジを選択します。
  このビットで選択したエッジが ZIN 端子で検出されると、カウンタの値が"0000_H"にクリアされます。
- CGSC ビットでゲート機能を選択した場合 (CGSC=1)
  有効レベルを選択します。
  このビットで選択したレベルが ZIN 端子から入力されている間だけ、カウンタが動作します。

CGE1	CGE0	カウンタクリア機能選択時(CGSC=0)	ゲート機能選択時(CGSC=1)
0	0	エッジ検出禁止	レベル検出禁止 (カウント禁止)
0	1	立下りエッジ	"L"レベル
1	0	立上りエッジ	"H"レベル
1	1	設定禁止	設定禁止

23.4.4 カウンタステータスレジスタ (CSR0, CSR1)

カウンタステータスレジスタのビット構成について示します。

アップダウンカウンタの状態の確認と、割込み要求を制御するレジスタです。

■ CSR0： アドレス 0F77H (アクセス：バイト)

■ CSR1： アドレス 0F87H (アクセス：バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	CSTR	CITE	UDIE	CMPF	OVFF	UDFF	UDF1	UDF0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R(RM1),W	R(RM1),W	R(RM1),W	R,WX	R,WX

[bit7] CSTR： カウント起動ビット

アップダウンカウンタを起動/停止します。

書込み値	説明
0	カウント動作を停止します。
1	アップダウンカウンタを起動します。

[bit6] CITE： 比較結果一致割込み許可ビット

カウンタの値が、リロードコンペアレジスタ (RCR) に設定した値と一致したとき (CMPF=1) に、比較結果一致割込み要求を発生させるかどうかを設定します。

書込み値	説明
0	比較結果一致割込み要求の発生を禁止します。
1	比較結果一致割込み要求の発生を許可します。

[bit5] UDIE： オーバフロー/アンダフロー割込み許可ビット

アップダウンカウンタがオーバフロー/アンダフローしたとき(OVFF/UDFF=1) に、オーバフロー/アンダフロー割込み要求を発生させるかどうかを設定します。

書込み値	説明
0	オーバフロー/アンダフロー割込み要求の発生を禁止します。
1	オーバフロー/アンダフロー割込み要求の発生を許可します。

[bit4] CMPF : 比較結果一致検出フラグビット
カウンタの値がリロードコンペアレジスタ (RCR) に設定した値と一致したことを示します。
このビットが"1"のときに CITE ビットが"1"に設定されていると、比較結果一致割込み要求が発生します。

CMPF	読出し時	書込み時
0	値は一致していません。	このビットを"0"にクリアします。
1	値が一致しました。	無視されます。

＜注意事項＞
このビットは, 次の場合に"1"に変わります。

- カウントアップで値が一致したとき
- リロードコンペアレジスタ (RCR) の値をカウンタにリロードしたとき
- アップダウンカウンタを起動したときに, 既に値が一致していたとき

[bit3] OVFF : オーバフロー検出フラグビット
アップダウンカウンタがオーバフローしたことを示します。
このビットが"1"のときに UDIE ビットが"1"に設定されていると、オーバフロー割込み要求が発生します。

OVFF	読出し時	書込み時
0	オーバフローは発生していません。	このビットを"0"にクリアします。
1	オーバフローが発生しました。	無視されます。

オーバフローは、カウンタの値が"$FFFF_H$"のときにカウントアップしようとすると発生します。

[bit2] UDFF : アンダフロー検出フラグビット
アップダウンカウンタがアンダフローしたことを示します。
このビットが"1"のときに UDIE ビットが"1"に設定されていると、アンダフロー割込み要求が発生します。

UDFF	読出し時	書込み時
0	アンダフローは発生していません。	このビットを"0"にクリアします。
1	アンダフローが発生しました。	無視されます。

アンダフローは、カウンタの値が"0000_H"のときに、カウントダウンしようとすると発生します。

[bit1, bit0] UDF1, UDF0 : アップダウンフラグビット
直前のカウント方向を示します。
このビットは、アップダウンカウンタがカウントするたびに更新されます。

UDF1	UDF0	説明
0	0	入力なし
0	1	カウントダウン
1	0	カウントアップ
1	1	カウントアップ/カウントダウン同時発生

23.5 割込み

アップダウンカウンタの割込みについて示します。

次のいずれかの場合に割込み要求が発生します。

- カウント方向が反転したとき (カウント方向転換割込み要求)
- カウンタの値がリロードコンペアレジスタ (RCR) に設定した値と一致したとき (比較結果一致割込み要求)
- オーバフローが発生したとき (オーバフロー割込み要求)
- アンダフローが発生したとき (アンダフロー割込み要求)

アップダウンカウンタの動作モードによって、発生する割込み要求は異なります。

動作モードと割込み要求の対応を表 23-5 に示します。

表 23-5 動作モードと割込み要求の対応

割込み要求	タイマモード	アップダウンカウントモード	位相差カウントモード (2 逓倍/4 逓倍)
カウント方向転換割込み要求	×	○	○
比較結果一致割込み要求	○	○	○
オーバフロー割込み要求	×	○	○
アンダフロー割込み要求	○	○	○

アップダウンカウンタで使用できる割込みについて表 23-6 に示します。

表 23-6 アップダウンカウンタの割込み

割込み要求	割込み要求フラグ	割込み要求許可	割込み要求のクリア
カウント方向転換割込み要求	CCR の CDCF=1	CCR の CFIE=1	CCR の CDCF ビットに"0"を書き込む
比較結果一致割込み要求	CSR の CMPF=1	CSR の CITE=1	CSR の CMPF ビットに"0"を書き込む
オーバフロー割込み要求	CSR の OVFF=1	CSR の UDIE=1	CSR の OVFF ビットに"0"を書き込む
アンダフロー割込み要求	CSR の UDFF=1	CSR の UDIE=1	CSR の UDFF ビットに"0"を書き込む

CCR：カウンタコントロールレジスタ
CSR：カウンタステータスレジスタ

＜注意事項＞

- 割込み要求が発生すると、割込み要求フラグがクリアされるまで、アップダウンカウンタは動作を停止します。
- カウンタコントロールレジスタ (CCR) の CMPF ビットは、カウントアップで値が一致した場合に加え、リロードコンペアレジスタ (RCR) の値がリロードされた場合や、アップダウンカウンタを起動時、既に値が一致していた場合も"1"に変わります。
- カウンタのクリアおよびリロードのタイミングについては、「23.6. 動作説明と設定手順例」の「■クリアイベント」および「■リロードイベント」を参照してください。
- 割込み要求フラグが"1"のときに割込み要求の発生を許可すると割込みを許可した時点で、割込み要求が発生します。
- 割込み要求の発生を許可する場合は、次のいずれかの処理を行ってください。
- 割込み要求の発生を許可する前に割込み要求をクリアする
- 割込み許可と同時に割込み要求をクリアする
- 各割込み要求の割込みベクタ番号については、『付録 C. 割込みベクター覧』を参照してください。
- 割込みベクタ番号に対応する割込みレベルは、割込みコントロールレジスタ (ICR00～ICR47) で設定します。割込みレベルの設定については、『割込み制御(割込みコントローラ)』の章を参照してください。

23.6 動作説明と設定手順例

アップダウンカウンタの動作について説明します。また、動作状態を設定するための手順例も示します。

■ 概要

● カウンタモード

アップダウンカウンタは、設定によって 16 ビットアップダウンカウンタとして使用することも、8 ビットアップダウンカウンタとして使用することもできます。

カウンタコントロールレジスタ (CCR) の M16E ビットで設定してください。

- 8 ビットモード (M16E=0)
  アップダウンカウントレジスタ下位 (UDCRL) のみを利用します。リロード値およびコンペア値は、リロードコンペアレジスタ下位 (RCRL) のみにバイトで書き込んでください。
- 16 ビットモード (M16E=1)
  アップダウンカウントレジスタ (UDCR) の上位バイトと下位バイトの両方を利用します。リロード値およびコンペア値は、リロードコンペアレジスタ (RCR) にハーフワードで書き込んでください。

● 動作モード

アップダウンカウンタの動作モードは、カウンタコントロールレジスタ (CCR) の CMS1, CMS0 ビットで次の 3 モード (4 種類) から選択できます。

- タイマモード (CMS1, CMS0=00)

あらかじめ設定した値からカウント用クロックに同期してカウントダウンするモードです。
カウント用クロックは、周辺クロック (PCLK) をプリスケーラで 2 分周/8 分周して生成されます。

- アップダウンカウントモード (CMS1, CMS0=01)
  外部信号入力端子から入力される信号をカウントアップ/カウントダウンするモードです。
- 位相差カウントモード (2 逓倍) (CMS1, CMS0=10) /位相差カウントモード (4 逓倍) (CMS1, CMS0=11)
  外部信号入力端子から入力される信号の位相差をカウントアップ/カウントダウンするモードです。エンコーダの A 相を AIN 端子, B 相を BIN 端子, Z 相を ZIN 端子から入力することで、回転角度や回転数のカウント, 回転方向の検出などを高精度で行えるため、モータなどのエンコーダのカウントに適しています。

■ 利用できる機能

● リロード/コンペア機能

8/16 ビットアップダウンカウンタは、カウンタコントロールレジスタ (CCR) の RLDE ビットと UCRE ビットでリロード機能およびコンペア機能の利用を許可/禁止できます。

- リロード機能
  カウントダウン時にアンダフローが発生すると、リロードコンペアレジスタ (RCR) に設定した値をリロードして、再度カウントダウンする機能です。動作については、「23.6.1. タイマモード時の動作」の「■カウント動作」を参照してください。
- コンペア機能
  アップダウンカウンタの値がリロードコンペアレジスタ (RCR) に設定した値と一致 (比較結果一致) した状態で、さらにカウントアップが行われようとすると、アップダウンカウンタの値を"0000_H"にクリアして、再度カウントアップする機能です。動作については, 「23.6.2. アップダウンカウントモード時の動作」の「■ カウント動作」を参照してください。
  この機能はタイマモードでは利用できません。

■ リロードコンペア機能
リロード機能とコンペア機能を組み合わせて使用する機能です。"0000_H"とリロードコンペアレジスタ (RCR) に設定した値の間でカウントダウン/アップを行うため、任意幅でのカウントが可能です。「23.6.2. アップダウンカウントモード時の動作」の「■ カウント動作」を参照してください。
この機能はタイマモードでは利用できません。

リロード機能/コンペア機能の設定方法を表 23-7 に示します。

表 23-7 リロード機能/コンペア機能の設定方法

RLDE	UCRE	説明
0	0	リロード機能/コンペア機能の利用禁止
0	1	リロード機能の利用禁止 コンペア機能の利用許可
1	0	リロード機能の利用許可 コンペア機能の利用禁止
1	1	リロード機能/コンペア機能の利用許可

● ZIN 端子の機能

カウンタコントロールレジスタ (CCR) の CGSC ビットで ZIN 端子の機能を次の中から選択できます。

■ カウンタクリア機能 (CGSC=0)
カウント動作中に ZIN 端子から有効エッジが入力されると、カウンタの値を"0000_H"にクリアします。
■ ゲート機能 (CGSC=1)
ZIN 端子から有効レベルが入力されている間だけ、カウンタが動作します。

カウンタクリア機能を選択した場合は有効エッジ、ゲート機能を選択した場合は有効レベルをカウンタコントロールレジスタ (CCR) の CGE1, CGE0 ビットで選択してください。

CGE1	CGE0	カウンタクリア機能選択時 (CGSC=0)	ゲート機能選択時 (CGSC=1)
0	0	エッジ検出禁止	レベル検出禁止 (カウント禁止)
0	1	立下りエッジ	"L" レベル
1	0	立上りエッジ	"H" レベル
1	1	設定禁止	設定禁止

■ クリアイベント

カウンタの値は、次のいずれかの場合に"0000_H"にクリアされます。

■ 本デバイスがリセットされた
■ ZIN 端子から有効エッジが入力された
(カウンタコントロールレジスタ (CCR) の CGSC ビットで ZIN 端子の機能をカウンタクリア機能 (CGSC=0) に設定している場合)
■ ソフトウェアによるクリア
カウンタコントロールレジスタ (CCR) の UDCC ビットに"0"が書き込まれた

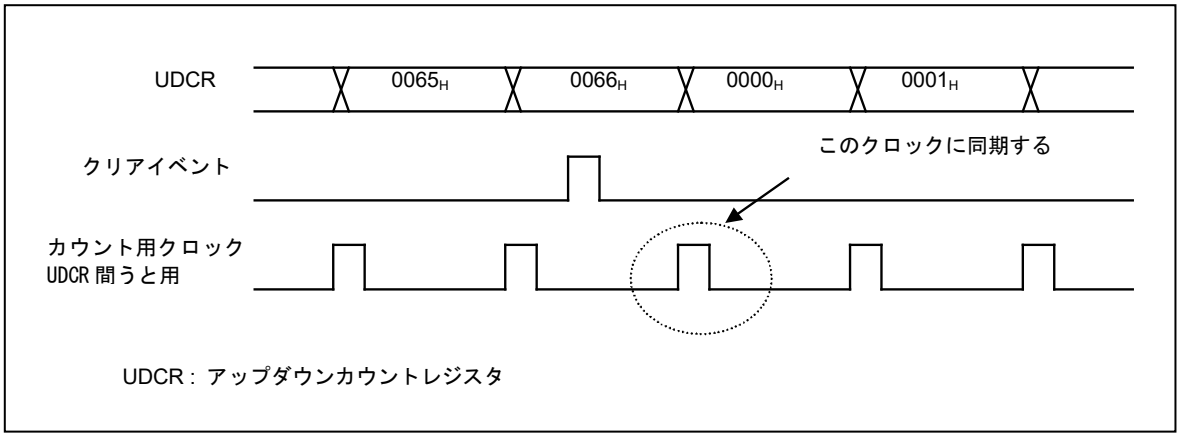
- コンペア機能によるクリア
  カウンタの値が、リロードコンペアレジスタ (RCR) に設定した値と一致し，さらにカウントアップが行われようとした
  (カウントダウンが行われた場合や、カウンタが停止した場合はクリアされません。)
- オーバフロー発生によるクリア
  カウンタの値が"$FFFF_H$" (8 ビットモード時は"FF_H") になった後のカウントアップ/カウントダウンのタイミング

カウンタの値が"0000_H"にクリアされるタイミングは、アップダウンカウンタの動作状態によって次のようになります。

- カウント動作中にクリアイベントが発生した場合カウント用クロックに同期して、値がクリアされます。

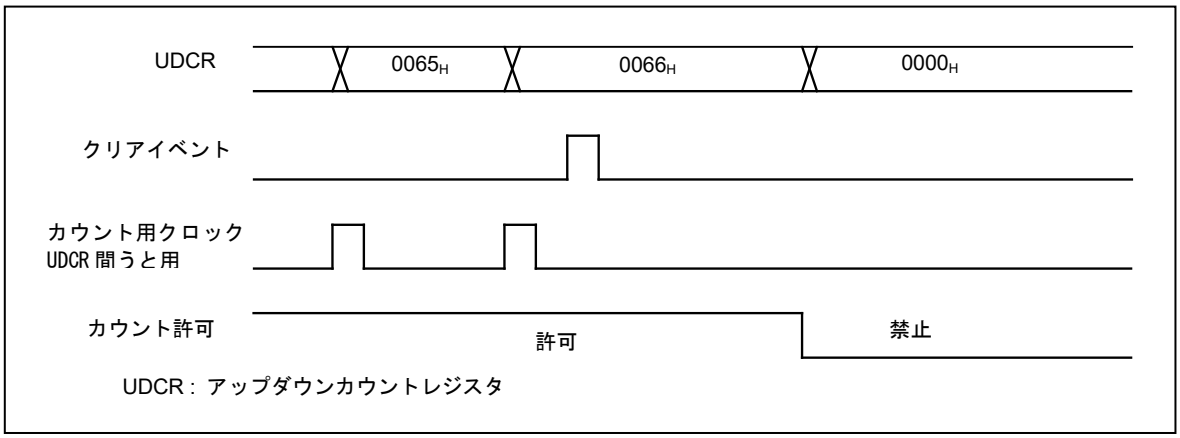
クリアイベント発生タイミングを図 23-2 クリアイベント発生タイミングに示します。

図 23-2 クリアイベント発生タイミング



- カウント動作中にクリアイベントが発生し、次のカウント用クロックが入力される前にカウント動作を停止した場合 (カウンタステータスレジスタ (CSR) の CSTR ビット=0) アップダウンカウンタが停止した時点で、値がクリアされます。
  クリアイベント発生タイミングを図 23-3 クリアイベント発生タイミングに示します。

図 23-3 クリアイベント発生タイミング



■ リロードイベント

アップダウンカウンタの値は、次のいずれかの場合にリロードされます。

- カウンタコントロールレジスタ (CCR) の CTUT ビットに"1"を書き込んだ場合
- リロード機能によって値がリロードされた場合

アップダウンカウンタの値がリロードされるタイミングは、アップダウンカウンタの動作状態によって次のようになります。

- カウント動作中にリロードイベントが発生した場合
  カウント用クロックに同期して、値がリロードされます。
- カウント停止中にリロードイベントが発生した場合
  リロードイベントが発生した時点で、値がリロードされます。

＜注意事項＞

- カウント動作中は、カウンタコントロールレジスタ (CCR) の CTUT ビットに"1"を書き込まないでください。
- リロードイベントとクリアイベントが同時に発生した場合は、クリアイベントが優先されます。

23.6.1　タイマモード時の動作

タイマモード時の動作について説明します。

■ 概要

リロードコンペアレジスタ (RCR) に設定した値から、カウントダウンするモードです。周辺クロック (PCLK) をプリスケーラで分周して、カウント用クロックとして使用します。

カウンタがアンダフローしたときに、リロードコンペアレジスタ (RCR) の値をリロードして、再度カウントダウンするリロード機能を使用することもできます。

■ カウント動作

● 通常動作

1. リロードコンペアレジスタ (RCR) にリロード値/コンペア値を設定
2. カウンタコントロールレジスタ (CCR) のCTUTビットに"1"を書き込むと設定した値がアップダウンカウントレジスタ (UDCR) に転送されます。
3. カウンタステータスレジスタ (CSR)のCSTRビットに"1"を書き込んでアップダウンカウンタの動作を許可するとリロードコンペアレジスタ (RCR) の設定した値からカウントダウンを開始します。

カウンタがアンダフローすると、カウンタステータスレジスタ (CSR) の UDFF ビットが"1"に変わります。このとき、カウンタステータスレジスタの UDIE ビットに"1"が設定されていると、アンダフロー割込み要求が発生します。

なお、カウンタコントロールレジスタ (CCR) の CGSC ビットで ZIN 端子をゲート機能 (CGSC=1) に設定した場合は、CGE1, CGE0 ビットで設定した有効レベルが ZIN 端子から入力されている間のみカウントします。

有効レベルの設定については、「23.4.3. カウンタコントロールレジスタ (CCR0, CCR1)」を参照してください。

＜注意事項＞

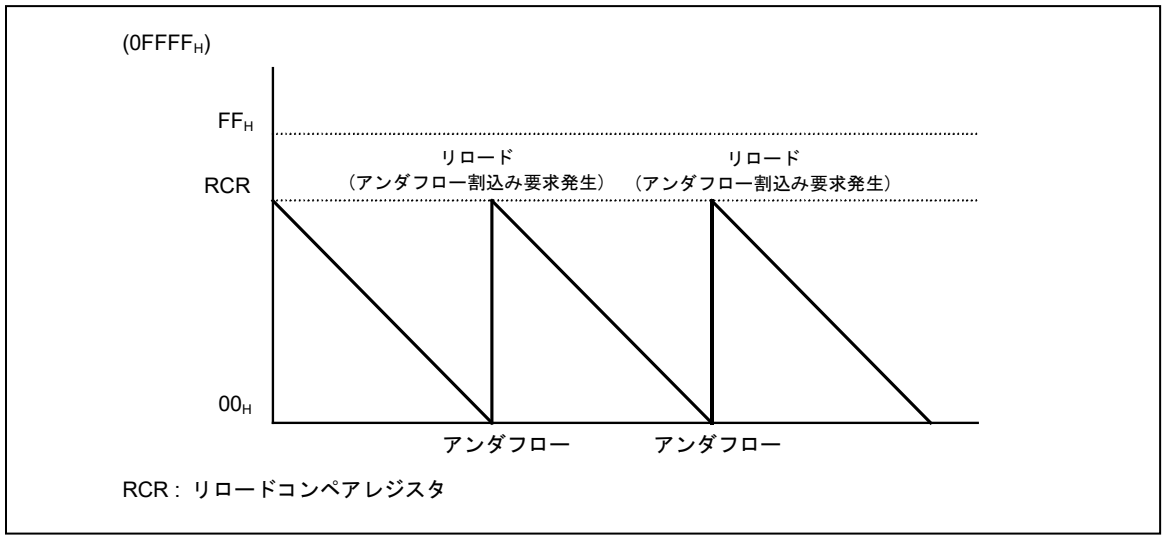
ZIN 端子に必要な最低パルス幅は、 2T (T：周辺クロック (PCLK) の周期) です。

● リロード機能使用時の動作

カウントダウン時に,カウンタがアンダフローすると､カウンタステータスレジスタ (CSR) のUDFFビットが"1"に変わります｡アンダフローが発生した次のカウントダウンタイミングで､リロードコンペアレジスタ (RCR) の値がリロードされ､再度カウントダウンを開始します｡このとき､カウンタステータスレジスタ (CSR) の UDIE ビットに"1"が設定されていると、アンダフロー割込み要求が発生します。

リロード機能使用時の動作を図 23-4 リロード機能使用時の動作に示します。

図 23-4 リロード機能使用時の動作



＜注意事項＞

リロードコンペアレジスタ (RCR) の値は、リロード値とコンペア値を兼ねています。そのため、リロードコンペアレジスタ (RCR) の値がリロードされると、カウンタステータスレジスタ (CSR) の CMPF ビットも"1"に変わります。

23.6.2　アップダウンカウントモード時の動作

アップダウンカウントモード時の動作について説明します。

■ 概要

AIN端子およびBIN端子から入力される外部信号をカウント用クロックとして、カウントアップ/カウントダウンするモードです。

AIN 端子から外部信号が入力されたときはカウントアップし、BIN 端子から外部信号が入力されたときはカウントダウンします。

外部信号のどのエッジでカウントするかは、カウンタコントロールレジスタ (CCR) の CES1, CES0 ビットで次の中から選択します。

- 立下りエッジ (CES1, CES0=01)
- 立上りエッジ (CES1, CES0=10)
- 両エッジ (CES1, CES0=11)

また、アップダウンカウントモード時には、次の 3 種類の機能を使用できます。

- リロード機能
- コンペア機能
- リロードコンペア機能

■ カウント動作

● 通常動作

カウンタが動作可能な状態で、AIN 端子から有効エッジが入力されたときはカウントアップし、BIN 端子から有効エッジが入力されたときはカウントダウンします。

カウントアップからカウントダウン、またはカウントダウンからカウントアップのようにカウント方向が反転すると、カウンタコントロールレジスタ (CCR) の CDCF ビットが"1"に変わります。このとき、カウンタコントロールレジスタ (CCR) のCFIE ビットに"1"が設定されていると、カウント方向転換割込み要求が発生します。

なお、カウンタコントロールレジスタ (CCR) の CGSC ビットで ZIN 端子をゲート機能 (CGSC=1) に設定した場合は、CGE1, CGE0 ビットで設定した有効レベルが ZIN 端子から入力されている間のみカウントします。

有効レベルの設定については、「23.4.3. カウンタコントロールレジスタ (CCR0, CCR1)」を参照してください。

＜注意事項＞

AIN 端子，BIN 端子，ZIN 端子に必要な最低パルス幅は、 2T (T：周辺クロック (PCLK) の周期) です。

● リロード機能使用時の動作

タイマモード時の動作と同様です。「23.6.1. タイマモード時の動作」の「■ カウント動作」を参照してください。

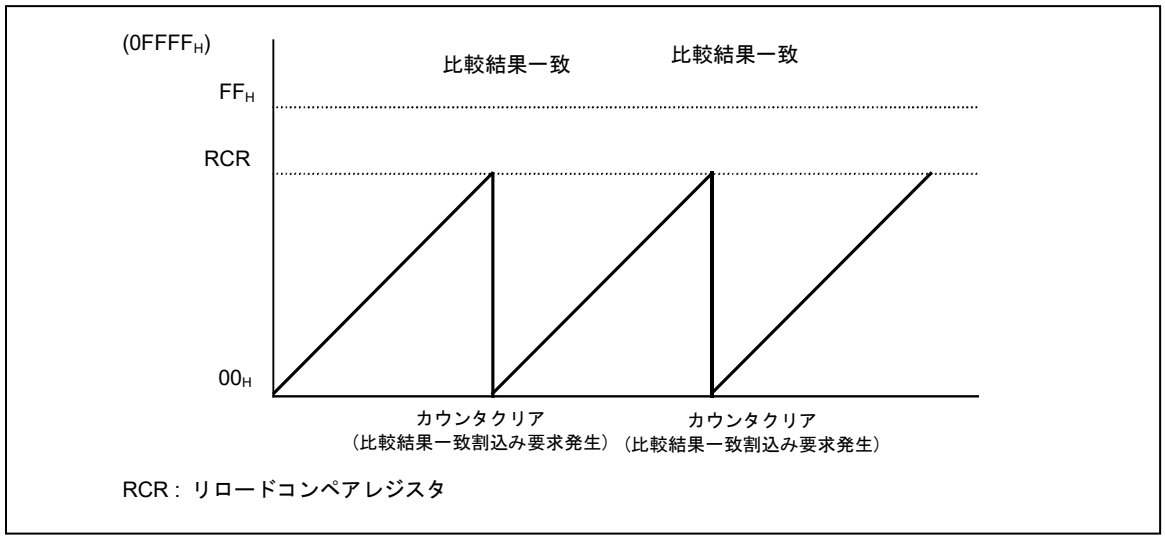
● コンペア機能使用時の動作

アップダウンカウンタの値がリロードコンペアレジスタ (RCR) に設定した値と一致するとカウンタステータスレジスタ (CSR) の CMPF ビットが"1"に変わります。このとき、カウンタステータスレジスタ (CSR) の CITE ビットに"1"が設定されていると、比較結果一致割込み要求が発生します。

この状態でさらにカウントアップが行われようとすると、アップダウンカウンタの値を"0000_H"にクリアして、再度カウントアップを開始します。

コンペア機能使用時の動作を図 23-5 コンペア機能使用時の動作に示します。

図 23-5 コンペア機能使用時の動作



＜注意事項＞

コンペア機能を使用した場合は，次の条件を満たしたときに，アップダウンカウンタの値が"0000_H"にクリアされます。

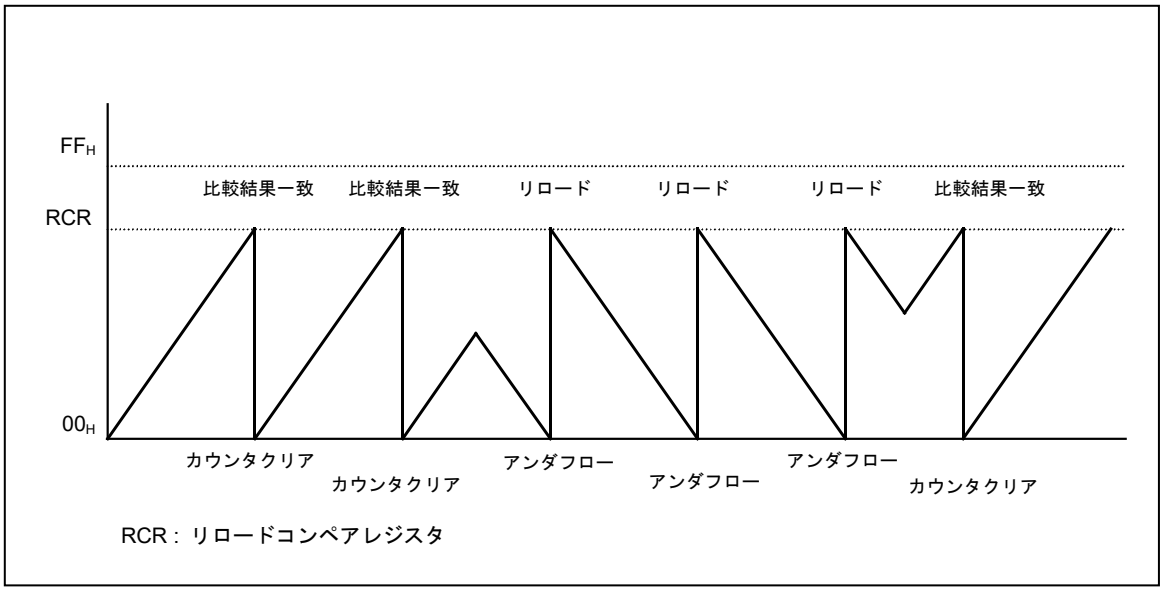
- アップダウンカウンタの値とリロードコンペアレジスタ (RCR) に設定した値が一致 (比較結果一致) し，さらに次のカウントアップが行われた
  ただし，比較結果が一致しても，次の場合はアップダウンカウンタの値はクリアされません。
- 次の動作がカウントダウン
- アップダウンカウンタが停止

● リロードコンペア機能使用時の動作

カウントダウン時はリロード機能を、カウントアップ時はコンペア機能を使用します。

リロードコンペア機能使用時の動作を図 23-6 リロードコンペア機能使用時の動作に示します。

図 23-6 リロードコンペア機能使用時の動作



● カウント方向の確認

このモードでは、カウントアップとカウントダウンの両方が行われます。そのため、カウント方向をカウンタステータスレジスタ (CSR) の UDF1, UDF0 ビットで確認できます。カウントが行われるたびに、このビットが書き換えられるため、現在のカウント方向を確認することができます。モータの制御などで回転方向を知りたい場合などに利用すると便利です。

UDF1, UDF0 ビットの示すカウント方向を表 23-8 UDF1, UDF0 ビットとカウント方向の対応に示します。

表 23-8 UDF1, UDF0 ビットとカウント方向の対応

UDF1	UDF0	カウント方向
0	0	入力なし
0	1	カウントダウン
1	0	カウントアップ
1	1	カウントアップ/カウントダウン同時発生

また、カウント方向が、カウントダウンからカウントアップ、またはカウントアップからカウントダウンに 1 回以上反転すると、カウンタコントロールレジスタ (CCR) の CDCF ビットが"1"に変わります。このとき、方向転換割込み要求も発生させることができるため、CDCF ビットと方向転換割込み要求の発生を利用して、カウント方向が反転したかどうかを確認できます。

＜注意事項＞

カウント方向の転換が短期間に連続発生した場合は、カウント方向が元に戻り, カウンタステータスレジスタ (CSR) の UDF1, UDF0 ビットで示す方向が、CDCF ビットが"1"に変わる前と同じ方向になる場合があります。

23.6.3 位相差カウントモード (2 逓倍) 時の動作

位相差カウントモード (2 逓倍) 時の動作について説明します。

■ 概要

2 本の外部信号入力端子から入力される信号の位相差をカウントするモードです。エンコーダ出力の A 相と B 相の位相差をカウントするのに適しています。

BIN 端子から立上りエッジ, 立下りエッジが検出されたときに、AIN 端子の入力レベルを確認し、BIN 端子と AIN 端子の位相差をカウントアップ/カウントダウンします。A 相が B 相より進んでいる場合はカウントアップし、遅れている場合はカウントダウンします。

カウントアップするかカウントダウンするかは、BIN 端子の検出エッジと AIN 端子の入力レベルによって異なります。

カウント方法を表 23-9 カウント方法に示します。

表 23-9 カウント方法

BIN端子	AIN端子	カウント方向
立上りエッジ	"H"レベル	カウントアップ
	"L"レベル	カウントダウン
立下りエッジ	"H"レベル	カウントダウン
	"L"レベル	カウントアップ

また、位相差カウントモード (2 逓倍) 時は、次の 3 種類の機能を使用できます。

- リロード機能
- コンペア機能
- リロードコンペア機能

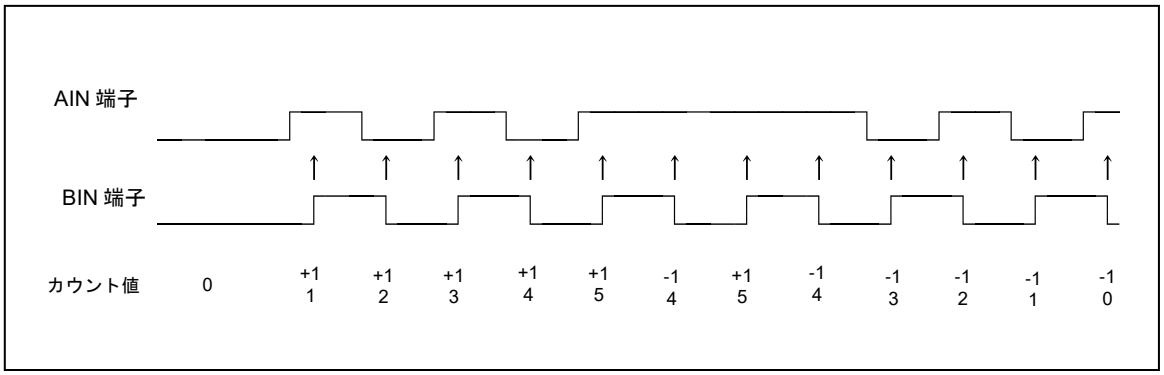
■ カウント動作

● 通常動作

カウンタが動作可能な状態で、BIN 端子から立上りエッジ/立下りエッジが入力されると、AIN 端子の入力レベルを検出し、カウントアップ/カウントダウンします。

位相差カウントモード (2 逓倍) 時の動作を図 23-7 位相差カウントモード (2 逓倍) 時の動作に示します。

図 23-7 位相差カウントモード (2 逓倍) 時の動作



なお、カウンタコントロールレジスタ (CCR) の CGSC ビットで ZIN 端子をゲート機能 (CGSC=1) に設定した場合は、CGE1, CGE0 ビットで設定した有効レベルが ZIN 端子から入力されている間のみカウントします。
有効レベルの設定については、「23.4.3. カウンタコントロールレジスタ (CCR0, CCR1)」を参照してください。

＜注意事項＞
AIN 端子，BIN 端子，ZIN 端子に必要な最低パルス幅は、 2T (T：周辺クロック (PCLK) の周期) です。

● リロード機能使用時の動作

タイマモード時の動作と同様です。「23.6.1. タイマモード時の動作」の「■ カウント動作」を参照してください。

● コンペア機能使用時の動作

アップダウンカウントモード時の動作と同様です。「23.6.2. アップダウンカウントモード時の動作」の「■ カウント動作」を参照してください。

● リロードコンペア機能使用時の動作

アップダウンカウントモード時の動作と同様です。「23.6.2. アップダウンカウントモード時の動作」の「■ カウント動作」を参照してください。

● カウント方向の確認

アップダウンカウントモード時と同様です。「23.6.2. アップダウンカウントモード時の動作」の「● カウント方向の確認」を参照してください。

23.6.4 位相差カウントモード (4 逓倍) 時の動作

位相差カウントモード (4 逓倍) 時の動作について説明します。

■ 概要

2 本の外部信号入力端子から入力される信号の位相差をカウントするモードです。エンコーダ出力の A 相と B 相の位相差をカウントするのに適しています。

AIN 端子または BIN 端子から立上りエッジ、立下りエッジが検出されたときに、もう一方の端子からの入力レベルを確認し、AIN 端子と BIN 端子の位相差をカウントアップ/カウントダウンします。

カウントアップするかカウントダウンするかは、検出するエッジと入力レベルの組合せによって異なります。

カウント方法を表 23-10 カウント方法に示します。

表 23-10 カウント方法

エッジ検出端子	検出エッジ	レベル確認端子	入力レベル	カウント方向
23.6.5 BIN 端子	立上りエッジ	AIN 端子	"H" レベル	カウントアップ
			"L" レベル	カウントダウン
	立下りエッジ		"H" レベル	カウントダウン
			"L" レベル	カウントアップ
AIN 端子	立上りエッジ	BIN 端子	"H" レベル	カウントダウン
			"L" レベル	カウントアップ
	立下りエッジ		"H" レベル	カウントアップ
			"L" レベル	カウントダウン

また、位相差カウントモード (4 逓倍) 時は、次の 3 種類の機能を使用できます。

- リロード機能
- コンペア機能
- リロードコンペア機能

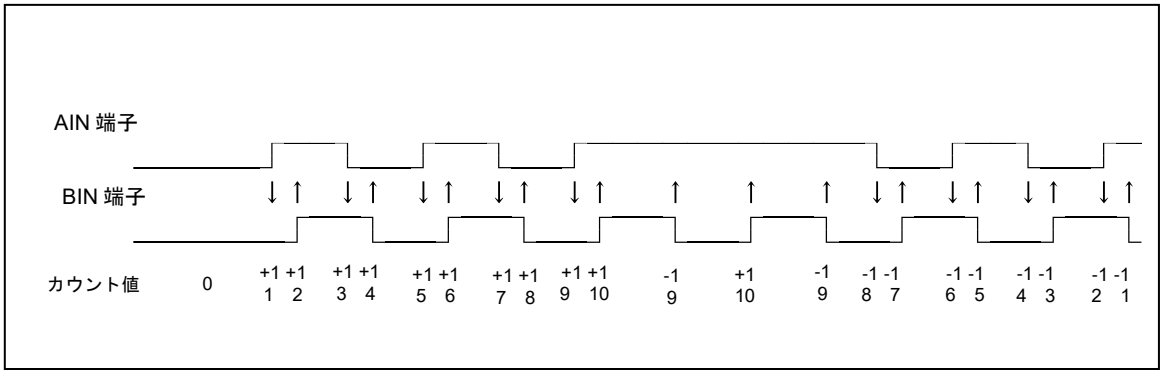
■ カウント動作

● 通常動作

カウンタが動作可能な状態で、AIN 端子または BIN 端子から立上りエッジ/立下りエッジが入力されると、もう一方の端子の入力レベルを検出し、カウントアップ/カウントダウンします。

位相差カウントモード (4 逓倍) 時の動作を図 23-8 位相差カウントモード (4 逓倍) 時の動作に示します。

図 23-8 位相差カウントモード (4 逓倍) 時の動作



なお、カウンタコントロールレジスタ (CCR) の CGSC ビットで ZIN 端子をゲート機能 (CGSC=1) に設定した場合は、CGE1, CGE0 ビットで設定した有効レベルが ZIN 端子から入力されている間のみカウントします。
有効レベルの設定については、「23.4.3. カウンタコントロールレジスタ (CCR0, CCR1)」を参照してください。

＜注意事項＞
AIN 端子, BIN 端子, ZIN 端子に必要な最低パルス幅は、2T (T：周辺クロック (PCLK) の周期) です。

● リロード機能使用時の動作

タイマモード時の動作と同様です。「23.6.1. タイマモード時の動作」の「■ カウント動作」を参照してください。

● コンペア機能使用時の動作

アップダウンカウントモード時の動作と同様です。「23.6.2. アップダウンカウントモード時の動作」の「■ カウント動作」を参照してください。

● リロードコンペア機能使用時の動作

アップダウンカウントモード時の動作と同様です。「23.6.2. アップダウンカウントモード時の動作」の「■ カウント動作」を参照してください。

● カウント方向の確認

アップダウンカウントモード時と同様です。「23.6.2. アップダウンカウントモード時の動作」の「● カウント方向の確認」を参照してください。

24. インプットキャプチャ

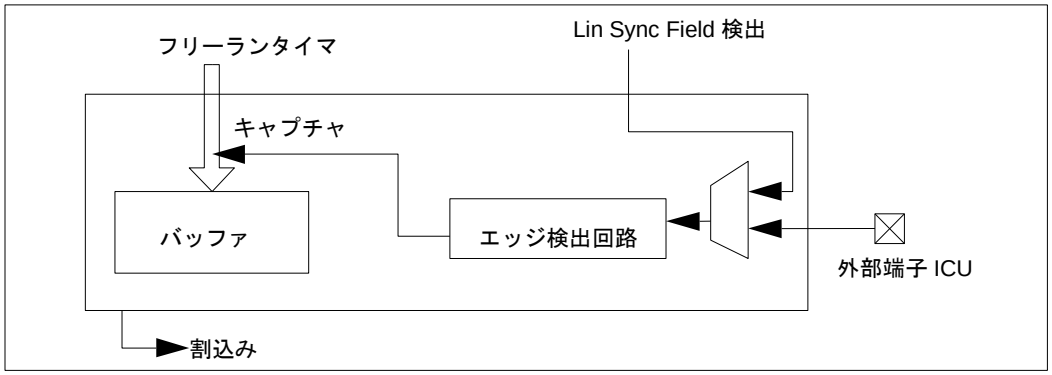
インプットキャプチャについて説明します。

24.1 概要

インプットキャプチャの概要について説明します。

インプットキャプチャは、外部からの信号を検出したタイミングで 32 ビットフリーランタイマのカウント値を記録します。そして、繰返し記録したカウント値から、信号間の時間を算出できます。外部入力端子からの有効エッジ検出時に割込みを発生することができます。

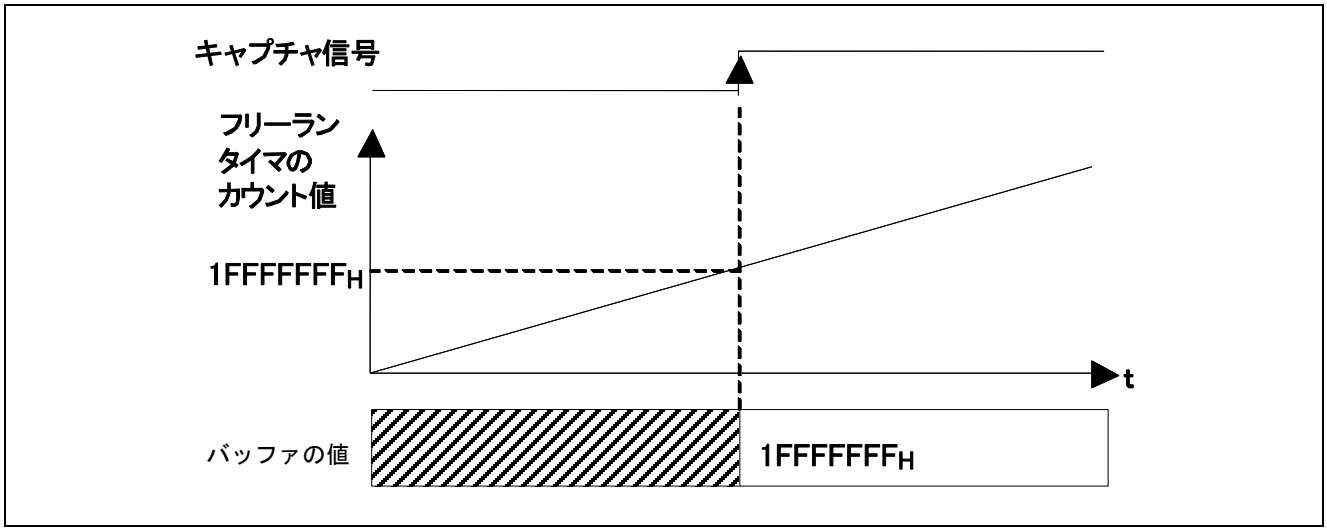
図 24-1 ブロックダイヤグラム



24.2 特長

インプットキャプチャの特長について説明します。

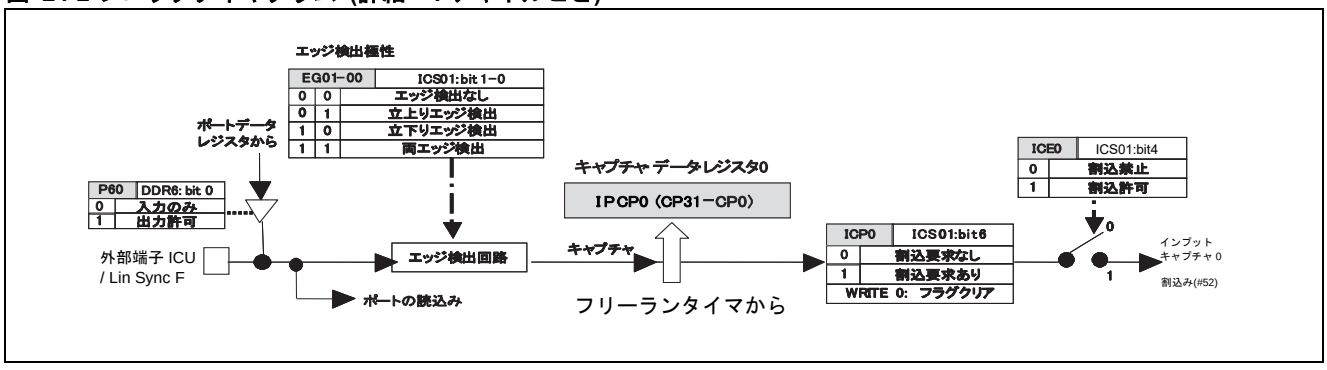
- 形式：エッジ検出回路＋32 ビットバッファ（キャプチャレジスタ）
- 個数：12
- エッジ検出：立上り/立下り/両エッジ
- 割込み：エッジ検出割込み
- キャプチャ値：タイマのカウント値（00000000H～FFFFFFFFH）
- タイマ：
  インプットキャプチャ 0～5:　フリーランタイマ 0 または 1 を使用
  インプットキャプチャ 6～11:　フリーランタイマ 2 または 3 を使用
- 精度：周辺クロック(PCLK) ÷1、÷2、÷4、÷8、÷16、÷32、÷64、÷128、÷256 (フリーランタイマのカウントクロック)



24.3 構成

インプットキャプチャの構成について説明します。

図 24-2 ブロックダイヤグラム (詳細・1 チャネルごと)



24.4 レジスタ

インプットキャプチャのレジスタについて説明します

■ ベースアドレス (Base_addr)・外部端子表

表 24-1 ベースアドレス (Base_addr)・外部端子表

チャネル	Base_addr	外部端子
		ICU 入力
0	0x02C4	ICU0_0／ICU0_1／ICU0_2
1	0x02C4	ICU1_0／ICU1_1／ICU1_2
2	0x02D0	ICU2_0／ICU2_1／ICU2_2
3	0x02D0	ICU3_0／ICU3_1／ICU3_2
4	0x02DC	ICU4_0／ICU4_1／ICU4_2
5	0x02DC	ICU5_0／ICU5_1／ICU5_2
6	0x0FD0	ICU6_0／ICU6_1／ICU6_2
7	0x0FD0	ICU7_0／ICU7_1／ICU7_2
8	0x0FDC	ICU8_0／ICU8_1／ICU8_2
9	0x0FDC	ICU9_0／ICU9_1／ICU9_2
10	0x0FE8	ICU10_0／ICU10_1／ICU10_2
11	0x0FE8	ICU11_0／ICU11_1／ICU11_2

表 24-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x02C4	IPCP0				インプットキャプチャデータレジスタ 0
0x02C8	IPCP1				インプットキャプチャデータレジスタ 1
0x02CC	ICFS01	予約	LSYNS0	ICS01	フリーランタイマ選択レジスタ 01 LIN SYNCH FIELD 切替えレジスタ 0 インプットキャプチャ制御レジスタ 01
0x02D0	IPCP2				インプットキャプチャデータレジスタ 2
0x02D4	IPCP3				インプットキャプチャデータレジスタ 3
0x02D8	ICFS23	予約		ICS23	フリーランタイマ選択レジスタ 23 インプットキャプチャ制御レジスタ 23
0x02DC	IPCP4				インプットキャプチャデータレジスタ 4
0x02E0	IPCP5				インプットキャプチャデータレジスタ 5
0x02E4	ICFS45	予約		ICS45	フリーランタイマ選択レジスタ 45 インプットキャプチャ制御レジスタ 45
0x0FD0	IPCP6				インプットキャプチャデータレジスタ 6
0x0FD4	IPCP7				インプットキャプチャデータレジスタ 7
0x0FD8	ICFS67	予約	LSYNS1	ICS67	フリーランタイマ選択レジスタ 67 LIN SYNCH FIELD 切替えレジスタ 1 インプットキャプチャ制御レジスタ 67
0x0FDC	IPCP8				インプットキャプチャデータレジスタ 8
0x0FE0	IPCP9				インプットキャプチャデータレジスタ 9
0x0FE4	ICFS89	予約		ICS89	フリーランタイマ選択レジスタ 89 インプットキャプチャ制御レジスタ 89
0x0FE8	IPCP10				インプットキャプチャデータレジスタ 10
0x0FEC	IPCP11				インプットキャプチャデータレジスタ 11
0x0FF0	ICFS1011	予約		ICS1011	フリーランタイマ選択レジスタ 1011 インプットキャプチャ制御レジスタ 1011

24.4.1　インプットキャプチャデータレジスタ ：IPCP

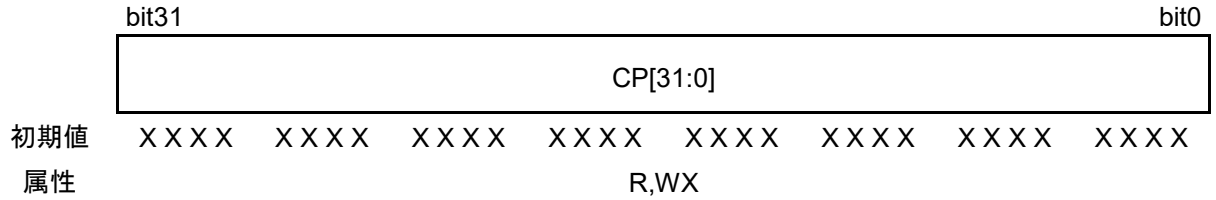
インプットキャプチャデータレジスタ（IPCP）のビット構成について説明します

外部からの入力信号の変化をトリガとして、フリーランタイマのカウント値を保持し、読み出すことができるレジスタです。

x：チャネル番号 0,2,4,6,8,10
y：チャネル番号 1,3,5,7,9,11

■ IPCPx (インプットキャプチャ x)：アドレス Base_addr+00H (アクセス：ワード)

■ IPCPy (インプットキャプチャ y)：アドレス Base_addr+04H (アクセス：ワード)



＜注意事項＞
このレジスタには、ワードアクセス命令を使用してください。また、このレジスタにデータを書き込むことはできません。

24.4.2　フリーランタイマ選択レジスタ ：ICFS

フリーランタイマ選択レジスタ（ICFS)のビット構成について説明します。

キャプチャ元フリーランタイマを選択します。

x：チャネル番号 0,2,4,6,8,10
y：チャネル番号 1,3,5,7,9,11

■ ICFSxy (フリーランタイマ選択 xy)：アドレス Base_addr+08H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	—	—	—	—	—	—	SELy	SELx
初期値	—	—	—	—	—	—	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W

[bit7～bit2] - :未定義
これらのビットへの書込みは動作に影響しません。

[bit1, bit0] SELy, SELx： フリーランタイマ選択

SEL{0,1,2,3,4,5}	動作
0	フリーランタイマ 0
1	フリーランタイマ 1

SEL{6,7,8,9,10,11}	動作
0	フリーランタイマ 2
1	フリーランタイマ 3

24.4.3 インプットキャプチャ制御レジスタ ：ICS

インプットキャプチャ制御レジスタ（ICS)のビット構成について説明します。

インプットキャプチャを制御するためのレジスタです。

x： チャネル番号 0,2,4,6,8,10
y： チャネル番号 1,3,5,7,9,11

■ ICSxy (インプットキャプチャ x-y)：アドレス base_addr+0B$_H$ (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ICPy	ICPx	ICEy	ICEx	EGy1	EGy0	EGx1	EGx0
初期値	0	0	0	0	0	0	0	0
属性	R(RM1),W	R(RM1),W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7, bit6] ICPy, ICPx： インプットキャプチャ割込み要求フラグ

ICPn	状態	
	読出し時	書込み時
0	割込み要求なし	フラグをクリア
1	割込み要求あり (エッジ検出あり)	動作に影響なし

■ キャプチャ有効エッジ選択ビット(EG[n1:n0])で選択した信号変化(エッジ)を、外部端子からの入力信号で検出すると、フラグが"1"になります。
■ CPU の割込み要求を有効にするには、割込み要求許可設定(ICEn=1)が必要です。
(注意事項) ICPn：n の番号がインプットキャプチャのチャネル番号に対応します。

[bit5, bit4] ICEy, ICEx： インプットキャプチャ割込み要求許可

ICEn	動作
0	割込み禁止
1	割込み許可

インプットキャプチャ割込み要求許可ビットが"1"の時、インプットキャプチャ割込み要求フラグが"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。

(注意事項)ICEn：n の番号がインプットキャプチャのチャネル番号に対応します。

[bit3～bit0] EGn1, EGn0： インプットキャプチャ n 有効エッジ選択

EGn1	EGn0	エッジ選択
0	0	インプットキャプチャ停止
0	1	立上りエッジ
1	0	立下りエッジ
1	1	両エッジ (立上りエッジおよび立下りエッジ)

■ 外部端子からのインプットキャプチャ信号に対するキャプチャ有効エッジを選択します。
■ 有効エッジ選択ビットが"00$_B$"の場合、インプットキャプチャは停止状態になります。
(注意事項)EGn1, EGn0：n の番号がインプットキャプチャのチャネル番号に対応します。

24.4.4 LIN SYNCH FIELD 切替えレジスタ ：LSYNS

LIN SYNCH FIELD 切替えレジスタ（LSYNS)のビット構成について説明します。

■ LSYNS0 (インプットキャプチャ 0-5)：アドレス 02CEH (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	—	—	LSYN5	LSYN4	LSYN3	LSYN2	LSYN1	LSYN0
初期値	—	—	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

[bit7, bit6] - : 未定義
常に"1"が読み出されます。書込みは動作に影響ありません。

[bit5～bit0] LSYN5～LSYN0： インプットキャプチャ 5-0 入力選択

LSYNn (n=0～5)	入力選択
0	外部端子入力(ICUn)
1	LIN-UART ch.(n+2)からの LIN synch field 検出信号入力

＜注意事項＞
インプットキャプチャの入力切替えはキャプチャ停止(ICS:EG[n1:n0]=00)の状態で行ってください。

キャプチャ動作が許可(ICS:EG[n1:n0]が"00"以外)で、外部端子入力の信号レベルと、LIN synch field 検出信号の状態(信号レベル)が異なる状態で入力切替えを行うと、エッジが検出され、キャプチャ有効エッジとして動作します。

■ LSYNS1 (インプットキャプチャ 6-9)： アドレス 0FDAH (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	—	—	—	—	LSYN9	LSYN8	LSYN7	LSYN6
初期値	—	—	—	—	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W	R/W	R/W

[bit7～bit4] - : 未定義
常に"1"が読み出されます。書込みは動作に影響ありません。

[bit3, bit2] LSYN9, LSYN8 :インプットキャプチャ 9-8 入力選択

LSYNn (n=8, 9)	入力選択
0	外部端子入力(ICUn)
1	マルチファンクションシリアルインタフェース ch.(n)からの LIN synch field 検出信号入力

[bit1, bit0] LSYN7, LSYN6 : インプットキャプチャ 7-6 入力選択

LSYNn (n=6, 7)	入力選択
0	外部端子入力(ICUn)
1	マルチファンクションシリアルインタフェース ch.(n - 6)からの LIN synch field 検出信号入力

＜注意事項＞

インプットキャプチャの入力切替えはキャプチャ停止(ICS:EG[n1:n0]=00)の状態で行ってください。

24.5 動作説明

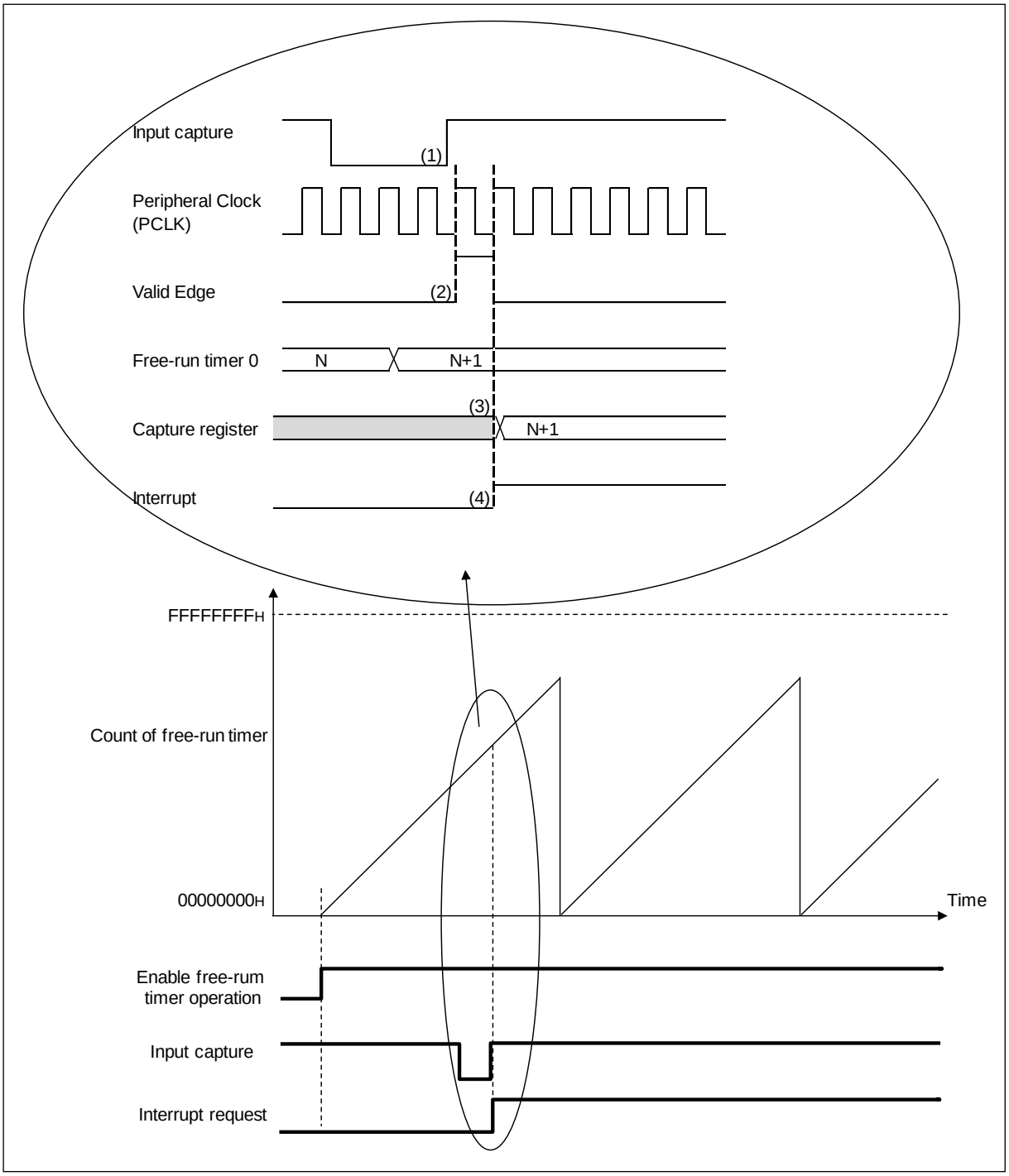
インプットキャプチャの動作について説明します。

32 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、32 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させることができます。

インプットキャプチャの動作について説明します。

24.5.1 取り込みタイミング，割込みタイミング

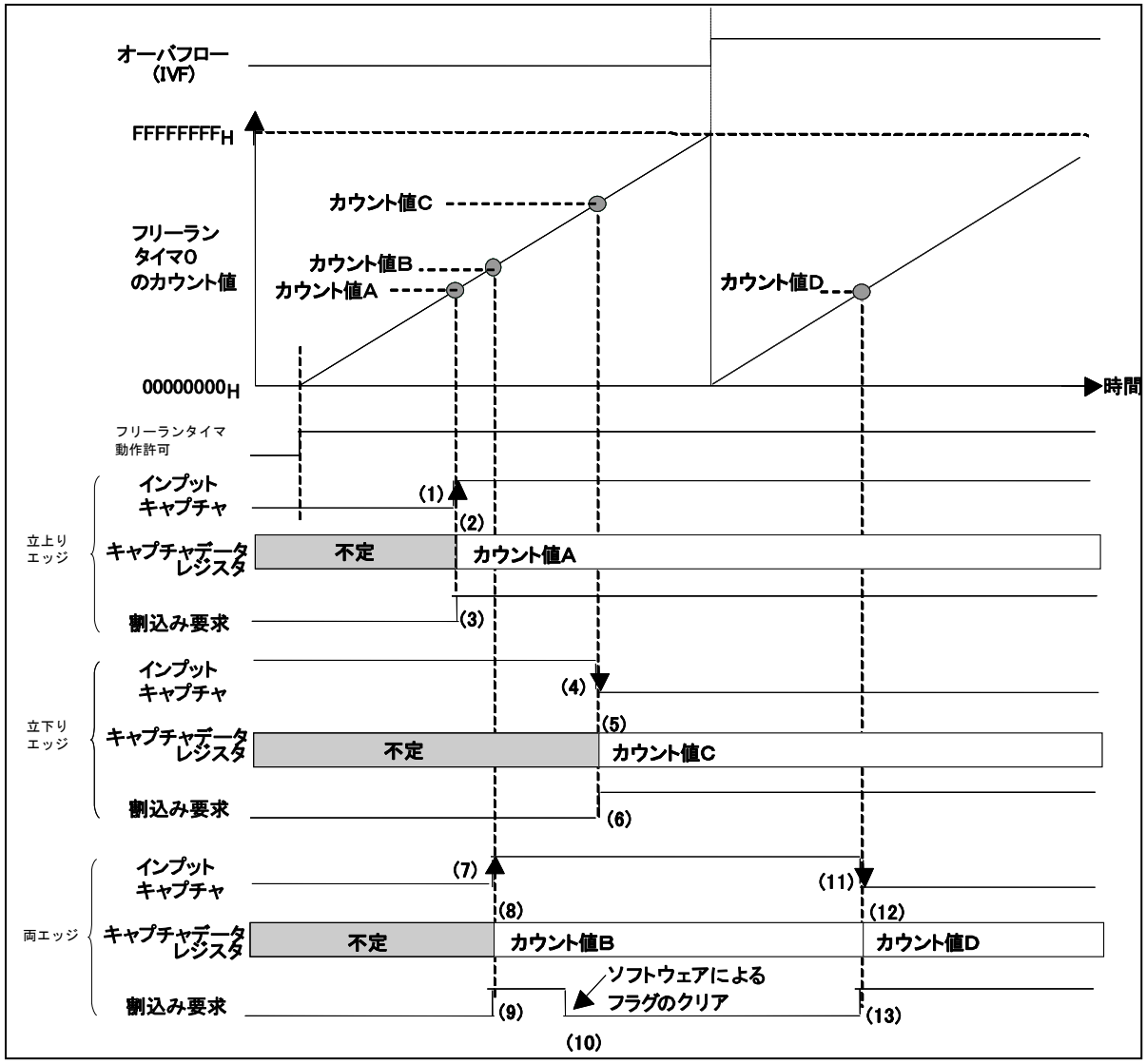
インプットキャプチャの取り込みタイミング，割込みタイミングについて説明します。



(1) 入力信号の立上りエッジ
(2) エッジ検出により発生した内部信号 (周辺クロックに同期)
(3) フリーランタイマの値をキャプチャレジスタに記録 (キャプチャ)
(4) インプットキャプチャ割込み発生(ICPn)=1)(n はチャネル番号)

24.5.2 インプットキャプチャのエッジ指定とその動作

インプットキャプチャのエッジ指定と動作について説明します。



● 立上りエッジ指定時

(1) 入力信号の立上りエッジ検出
(2) フリーランカウンタ値をキャプチャレジスタに記録 (キャプチャ)
(3) インプットキャプチャ割込み発生

● 立下りエッジ指定時

(4) 入力信号の立下りエッジ検出
(5) フリーランカウンタ値をキャプチャレジスタに記録 (キャプチャ)
(6) インプットキャプチャ割込み発生

● 両エッジ

(7)　入力信号の立上りエッジを検出
(8)　フリーランカウンタ値をキャプチャレジスタに記録 (キャプチャ)
(9)　インプットキャプチャ割込み発生
(10)　ソフトウェアで割込み要求フラグ(ICS01:ICP0),(ICS01:ICP1),(ICS23:ICP2),(ICS23:ICP3),・・・をクリア
(11)　入力信号の立下りエッジを検出
(12)　フリーランカウンタ値をキャプチャレジスタに記録 (キャプチャ)
(13)　インプットキャプチャ割込み発生

24.6 設定

インプットキャプチャの設定について説明します。

表 24-3 インプットキャプチャを使うために必要な設定

設定	設定レジスタ	設定方法
フリーランタイマの設定 フリーランタイマの起動	『フリーランタイマ』の章を参照してください。	—
入力端子 ICUn[*1] と インプットキャプチャ 入力切替えの設定	LIN_UART またはマルチファンクションシリアルインタフェースとの連携機能を使用する場合： LIN SYNCH FIELD 切替えレジスタ(LSYNS0),(LSYNS1) 外部入力の場合： LIN SYNCH FIELD 切替えレジスタ(LSYNS0), (LSYNS1) ICU 端子の設定 (『I/O ポート』の章を参照してください。)	24.7.2 参照
外部入力の有効エッジ極性選択	インプットキャプチャ制御レジスタ(ICS01),(ICS23),・・・	24.7.1 参照

*1: n はチャネル番号

表 24-4 インプットキャプチャ割込みを行うために必要な設定

設定	設定レジスタ	設定方法
インプットキャプチャ割込みベクタ, インプットキャプチャ割込みレベルの設定	『割込み制御(割込みコントローラ)』の章を参照してください。	24.7.3 参照
インプットキャプチャ割込み設定 割込み要求のクリア 割込み要求の許可	インプットキャプチャ制御レジスタ (ICSxy)[*2]	24.7.5 参照

*2: x, y はチャネル番号

24.7 Q&A

インプットキャプチャの Q&A について説明します。

24.7.1. 外部入力の有効エッジ極性の種類と選択方法は?

24.7.2. 外部入力端子 (ICU0, ICU1,ICU2,ICU3,ICU4,ICU5・・・) を有効にするには?

24.7.3. 割込み関連レジスタは?

24.7.4. 割込みの種類は?

24.7.5. 割込みを許可するには?

24.7.6. 入力信号のパルス幅を測定するには?

24.7.1 外部入力の有効エッジ極性の種類と選択方法は?

外部入力の有効エッジ極性の種類と選択方法について説明します。

有効エッジ極性は、立上り，立下り，両エッジの 3 種類です。

外部入力の有効エッジ極性ビット(ICS01:EG[01:00]), (ICS01:EG[11:10]),
(ICS23:EG[21:20]), (ICS23:EG[31:30]) ,(ICS45:EG[41:40]), (ICS45:EG[51:50]) ,
(ICS67:EG[61:60]), (ICS67:EG[71:70])・・・ で設定してください。

動作	外部入力の有効エッジ極性ビット (EG[01:00]),(EG[11:10]),(EG[21:20]), (EG[31:30]),(EG[41:40]),(EG[51:50]), (EG[61:60]),(EG[71:70])・・・
立上りエッジを選択するには	"01"を選択する
立下りエッジを選択するには	"10"を選択する
両エッジを選択するには	"11"を選択する

24.7.2 外部入力端子 (ICU0, ICU1,ICU2,ICU3,ICU4,ICU5・・・) を有効にするには?

外部入力端子 (ICU0～ICU11)を有効設定について説明します。

LSYNS0/1 レジスタを外部端子入力に設定してください。また、ICU0～ICU11 端子をペリフェラル入力に設定してください。設定方法は『I/O ポート』の章を参照してください。

24.7.3 割込み関連レジスタは?

割込み関連レジスタについて説明します。

インプットキャプチャ割込みベクタ, インプットキャプチャ割込みレベルの設定

割込み番号は『付録』の『C. 割込みベクター覧』を参照してください。

割込みレベル, 割込みベクタの詳細については『割込み制御(割込みコントローラ)』の章を参照してください。

割込み要求フラグ (ICS01:ICP0), (ICS01:ICP1), (ICS23:ICP2), (ICS23:ICP3), (ICS45:ICP4), (ICS45:ICP5), (ICS67:ICP6), (ICS67:ICP7)・・・は自動的にクリアしませんので、割込み処理から復帰する前にソフトウェアにてインプットキャプチャ割込み要求フラグ(ICP0, ICP1, ICP2, ICP3, ICP4, ICP5, ICP6, ICP7・・・)に"0"を書込みクリアしてください。

24.7.4 割込みの種類は?

割込みの種類について説明します。

割込みは 1 種類のみです。入力信号のエッジ検出で発生します。

24.7.5 割込みを許可するには?

割込みの許可について説明します。

割込み要求の許可, 割込み要求フラグ
割込み許可の設定は、割込み要求許可ビット
(ICS01:ICE0), (ICS01:ICE1), (ICS23:ICE2),(ICS23:ICE3), (ICS45:ICE4), (ICS45:ICE5), (ICS67:ICE6), (ICS67:ICE7)・・・にて行ってください。

動作	割込み要求許可ビット (ICE0), (ICE1), (ICE2), (ICE3), (ICE4), (ICE5), (ICE6), (ICE7)・・・
割込み禁止	"0"にする
割込み許可	"1"にする

割込み要求のクリアは、割込み要求フラグ
(ICS01:ICP0), (ICS01:ICP1), (ICS23:ICP2),(ICS23:ICP3), (ICS45:ICP4), (ICS45:ICP5), (ICS67:ICP6), (ICS67:ICP7)・・・にて行ってください。

動作	割込要求フラグビット (ICP0), (ICP1), (ICP2), (ICP3), (ICP4), (ICP5), (ICP6), (ICP7)・・・
割込み要求クリア	"0"を書き込む

24.7.6 入力信号のパルス幅を測定するには?

入力信号のパルス幅測定について説明します。

● "H"幅測定

エッジ検出に両エッジを指定してください。

最初に立上りエッジを検出し、次に立下りエッジを検出するように使用してください。

パルス幅 = {立下り時に記録した値 (インプットキャプチャレジスタ値)
＋"100000000_H"× オーバフロー回数
－立上り時に記録した値 (インプットキャプチャレジスタ値)}
×フリーランタイマのカウントクロック幅

例 : 立下り時に記録した値 = 23200000_H, 立上り時に記録した値 = $A6350000_H$,
オーバフロー回数 = 1, カウントクロック = 125ns
==> パルス幅 = (23200000_H+100000000_H-$A6350000_H$)×125ns = 261.972s

● 周期測定

エッジ検出に立上り (または立下り) を指定してください。

エッジを 2 回検出します。

周期 = {2 回目に記録した値 (インプットキャプチャレジスタ値)
＋"100000000_H" × オーバフロー回数
－1 回目に記録した値 (インプットキャプチャレジスタ値)}
×フリーランタイマのカウントクロック幅

＜注意事項＞
この計算式は、コンペア一致クリア機能を使わない場合の例です。

24.8 サンプルプログラム

インプットキャプチャのサンプルプログラムについて示します。

設定手順例 1

ICU0 に入力するパルスの立上りを検出してフリーランタイマの値を記録する。
これを2 回繰り返してトリガからトリガまでの時間を計測する。ただし，キャプチャ値の読出しと計算処理は割込み処理とする。

1. 初期設定

・フリーランタイマ ch.0 の制御　　レジスタ名．ビット名

制御レジスタの設定	TCCS0
クロック選択≫	.ECKE
コンペア割込み要求フラグ≫	.ICLR
コンペア割込み要求許可≫	.ICRE
カウント動作≫	.STOP
TCDT クリア	.SCLR
カウントクロック≫	.CLK3-0
タイマデータ値の設定	TCDT0

・ポート　　レジスタ名．ビット名

ポートの ICU0 入力設定	『I/O ポート』の章を参照

・インプットキャプチャの制御　　レジスタ名．ビット名

制御レジスタの設定	ICS0
割込み要求フラグ≫	.ICP1,ICP0
割込み要求許可≫	.ICE1,ICE0
ch1 有効エッジ極性選択≫	.EG11,EG10
ch0 有効エッジ極性選択≫	.EG01,EG00

・割込み関連　　レジスタ名．ビット名

割込みレベルの設定	ICR36
I フラグの設定	(CCR)

・変数の設定

2. 起動

・インプットキャプチャ ch.0 起動　　レジスタ名．ビット名

割込み制御	ICS01.ICE0

・フリーランタイマ ch.0 起動　　レジスタ名．ビット名

カウント動作起動	TCCS0.STOP

3. 割込み

・割込み処理　　レジスタ名．ビット名

割込み要求フラグのクリア	ICS01.ICP0
(任意の処理)	
.....	

4. 割込みベクタ

・ベクタテーブルの設定

(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 1

```
void INPUT0_sample_1(void)
{
        freerun0_initial();
        INPUT0_initial();
        INPUT0_start();
        freerun0_start();
}

void freerun0_initial(void)
{
        IO_TCCS0.word = 0x0041;     /* 設定値=0000_0000_0100_0001 */
                                    /* bit15 = 0       ECKE 内部クロックソース */
                                    /* bit14 -10 =0    予約ビット */
                                    /* bit9 = 0        割込みフラグクリア */
                                    /* bit8 = 0        割込み禁止 */
                                    /* bit7 = 0        予約ビット */
                                    /* bit6 = 1        */
                                    /* bit5 = 0        予約ビット */
                                    /* bit4 = 0        */
                                    /* bit3-0 = 0001 */
        IO_TCDT0 = 0x0000;          /* タイマデータ値の初期化 */
}

void INPUT0_initial(void)
{
        PORT_SETTING_ICU0_IN(); /* ICU0 端子をペリフェラル入力に設定してください。 */

        IO_ICS01.byte = 0x01;       /* 設定値=0000_0001 */
                                    /* bit7-6 = 00 ICP1, 0 有効エッジ検出なし */
                                    /* bit5-4 = 00 ICE1, 0 割込み禁止 */
                                    /* bit3-2 = 00 EG11, EG10 ch.1 エッジ検出なし */
                                    /* bit1-0 = 01 EG01, EG00 ch.0 立上りエッジ検出 */

        IO_ICR[36].byte = 0x10;     /* インプットキャプチャ ch.0 割込みレベル設定( 値は任意) */
        __EI();                     /* 割込み許可 */
        count = 0;
}

void INPUT0_start(void)
{
        IO_ICS01.bit.ICE0 = 1;      /* bit4 = 1 ICE0 ch.0 割込み許可 */
}

void freerun0_start(void)
{
        IO_TCCS0.bit.STOP = 0;      /* bit6 = 0 STOP カウント許可 */
}

__interrupt void INPUT0_int(void)
{
        IO_ICS01.bit.ICP0 = 0;      /* bit6 = 0 ICP0 有効エッジ検出フラグのクリア */
        if(count==0)
        data1 = IO_IPCP0;           /* フリーランタイマ値を記録する。(1 回目) */
        else if(count==1) {
        data2 = IO_IPCP0;           /* フリーランタイマ値を記録する。(2 回目) */
        cycle = (data2-data1)*125;  /* 時間を計測する。*/
        count = 0;
        }
} count++;

ベクタテーブルにて割込みルーチンの指定が必要
#pragma intvect INPUT0_int 52
```

24.9 注意事項

インプットキャプチャの注意事項について説明します。

● インプットキャプチャレジスタ

リセット時のインプットキャプチャレジスタの値は不定です。

インプットキャプチャレジスタの読出しは、必ず 32 ビットアクセスで行ってください。

● リードモディファイライト

インプットキャプチャ割込み要求ビット(ICP0), (ICP1), (ICP2), (ICP3), (ICP4), (ICP5), (ICP6), (ICP7)・・・をリードモディファイライト系命令で読み出すと"1"が読み出せます。

25. リアルタイムクロック (RTC)

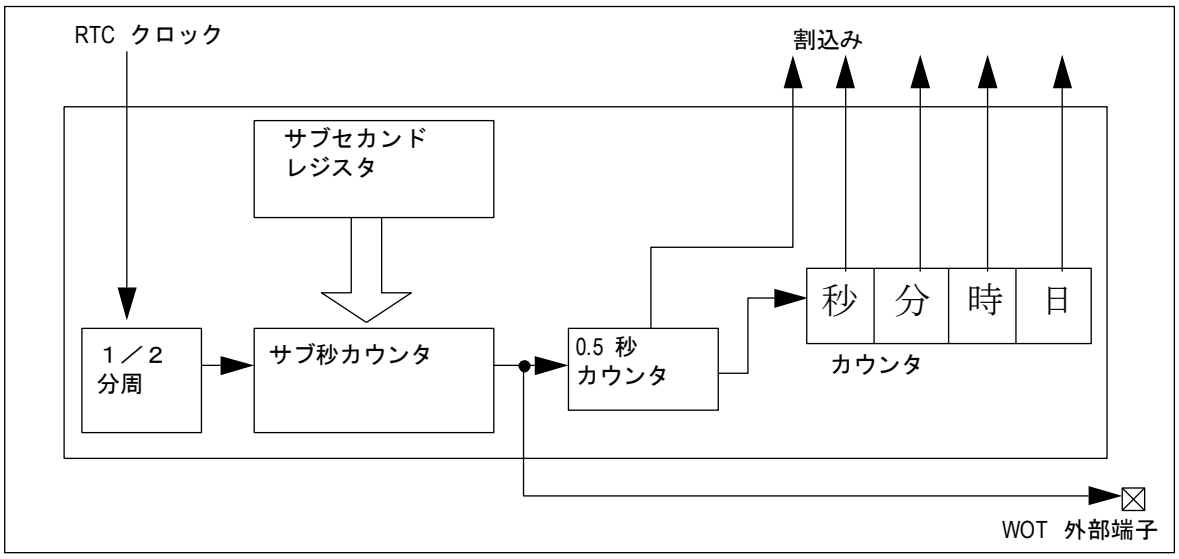
リアルタイムクロック (RTC)について説明します。

25.1 概要

リアルタイムクロック (RTC) の概要について説明します。

リアルタイムクロック(時計タイマ)は、タイマ制御レジスタ, サブセカンドレジスタ, 秒/分/時間/日数レジスタ, 1/2 クロック分周器, サブ秒カウンタ(22 ビットダウンカウンタ)および秒/分/時間/日数カウンタで構成されています。リアルタイムクロックの動作はリアルワールドタイマの動作と同じであり、リアルワールドタイマ情報を提供します。

図 25-1 ブロックダイヤグラム (概要)



25.2 特長

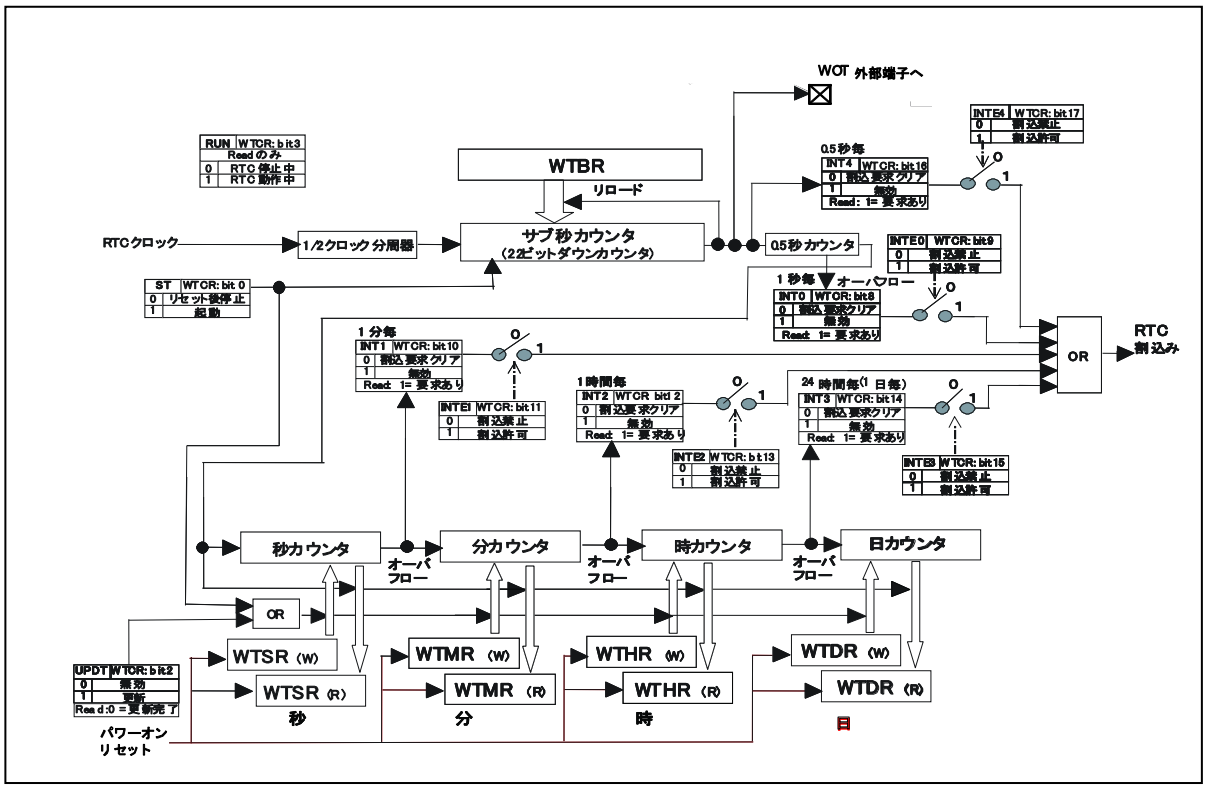
リアルタイムクロック (RTC)の特長について説明します。

- 機能：日数時刻 (日/時/分/秒) のカウント (時計モード中も動作を継続します。)
  日数時刻の初期設定, 修正が可能です。
- 動作クロック：RTC クロック (RTC クロックのクロックソース選択については『クロック』の章を、ソースとしてサブクロック(2 系統品のみ)を選択中の場合の補正については『RTC／WDT1 補正』の章を参照してください。)
- 割込み：0.5 秒, 1 秒, 1 分, 1 時間, 1 日の 5 つの周期の割込みができます。また、サブセカンド値を変えることで任意の周期 (短周期から長周期まで) の割込みを発生させることも可能です。

25.3 構成

リアルタイムクロック (RTC) の構成について示します。

図 25-2 構成図



25.4 レジスタ

リアルタイムクロック (RTC)のレジスタについて説明します。

表 25-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x055C	予約	予約	WTDR		日時分秒レジスタ(日)
0x0560	予約	WTCR			RTC 制御レジスタ
0x0564	予約	WTBR			サブセカンドレジスタ
0x0568	WTHR	WTMR	WTSR	予約	日時分秒レジスタ(時) 日時分秒レジスタ(分) 日時分秒レジスタ(秒)

25.4.1 RTC 制御レジスタ ：WTCR

RTC 制御レジスタ (WTCR)のビット構成について説明します。

リアルタイムクロックモジュールの動作を制御するためのレジスタです。

■ WTCRH：アドレス 0561H (アクセス：バイト)

■ WTCRM：アドレス 0562H (アクセス：バイト, ハーフワード)

■ WTCRL：アドレス 0563H (アクセス：バイト, ハーフワード)

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	—	—	—	—	—	—	INTE4	INT4
初期値	—	—	—	—	—	—	0	0
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R(RM1), W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	INTE3	INT3	INTE2	INT2	INTE1	INT1	INTE0	INT0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R(RM1), W	R/W	R(RM1), W	R/W	R(RM1), W	R/W	R(RM1), W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	予約	予約	予約	RUN	UPDT	予約	ST
初期値	0	0	0	0	0	0	0	0
属性	R/W0	R/W0	R/W0	R/W0	R,WX	R(RM0),W	R/W0	R/W

本レジスタは時計モード（電源遮断）からの復帰リセットを除いたすべてのリセット要因で初期化されます。

[bit23～bit18] - : 未定義

常に"1"が読み出されます。書込みは動作に影響ありません。

[bit17] INTE4 : 0.5 秒間割込み要求許可

INTE4	動作
0	0.5 秒間割込み要求禁止
1	0.5 秒間割込み要求許可

[bit16] INT4 : 0.5 秒間割込み要求フラグ

INT4	状態	
	読出し時	書込み時
0	0.5 秒間割込み要求なし	フラグクリア
1	0.5 秒間割込み要求あり	動作への影響はありません

サブ秒カウンタ (22 ビットダウンカウンタ)のボロー信号の分周出力が有効になったときにフラグが"1"になります。

[bit15] INTE3 : 1 日割込み要求許可

INTE3	動作
0	1 日(24 時間)割込み要求禁止
1	1 日(24 時間)割込み要求許可

[bit14] INT3 : 1 日割込み要求フラグ

INT3	状態	
	読出し時	書込み時
0	1 日(24 時間)割込み要求なし	フラグクリア
1	1 日(24 時間)割込み要求あり	動作への影響はありません

時間カウンタのオーバフロー時にフラグが"1"になります。

[bit13] INTE2 : 1 時間割込み要求許可

INTE2	動作
0	1 時間割込み要求禁止
1	1 時間割込み要求許可

[bit12] INT2 : 1 時間割込み要求フラグ

INT2	状態	
	読出し時	書込み時
0	1 時間割込み要求なし	フラグクリア
1	1 時間割込み要求あり	動作への影響はありません

分カウンタのオーバフロー時にフラグが"1"になります。

[bit11] INTE1 : 1 分間割込み要求許可

INTE1	動作
0	1 分間割込み要求禁止
1	1 分間割込み要求許可

[bit10] INT1 : 1 分間割込み要求フラグ

INT1	動作	
	読出し時	書込み時
0	1 分間割込み要求なし	フラグクリア
1	1 分間割込み要求あり	動作への影響はありません

秒カウンタのオーバフロー時にフラグが"1"になります。

[bit9] INTE0 : 1 秒間割込み要求許可

INTE0	動作
0	1 秒間割込み要求禁止
1	1 秒間割込み要求許可

[bit8] INT0 : 1 秒間割込み要求フラグ

INT0	状態	
	読出し時	書込み時
0	1 秒間割込み要求なし	フラグクリア
1	1 秒間割込み要求あり	動作への影響はありません

0.5 秒カウンタのオーバフロー時にフラグが"1"になります。

[bit7～bit4] 予約
必ず"0"を書き込んでください。

[bit3] RUN : 動作状態

RUN	状態
0	リアルタイムクロックモジュールは停止
1	リアルタイムクロックモジュールは動作中

[bit2] UPDT : 更新

UPDT	状態/動作	
	読出し時	書込み時
0	更新完了	動作への影響はありません
1	更新中	それぞれの時分秒カウンタのカウンタ値を 日時分秒レジスタ値に更新

更新ビット(UPDT)に"1"を書き込む前に、日時分秒レジスタに更新する値を設定してください。

日時分秒レジスタの更新は、サブ秒カウンタ(22 ビットダウンカウンタ)のリロード発生時に実行します。

UPDT ビットは、カウンタ値が更新されると、ハードウェアによりクリアされます。ただし、"1"書込みと更新完了が同時に発生した場合、UPDT ビットは"0"にクリアされません。

[bit1] 予約
必ず"0"を書き込んでください。

[bit0] ST : スタート

ST	動作
0	リアルタイムクロックモジュールは停止します。 すべてのカウンタがクリアされます。
1	日時分秒レジスタに設定した値を日時分秒カウンタ取り込み、 リアルタイムクロックが動作を開始します。

＜注意事項＞
RTC 停止の状態(ST=0)からスタートビット(ST)への"1"書込み(RTC 動作開始)の際には、スタートビットと同時に更新ビット(UPDT)への"1"書込みを行わないでください。
(ST=0 状態での、バイト即値による ST ビットと UPDT ビットの同時"1"書込みは禁止です。)

＜注意事項＞
更新ビット(UPDT)に"1"を書き込む場合は、RTC 動作中(ST=1)の状態で行ってください。

＜注意事項＞
更新ビット(UPDT)が"1"の状態で、スタートビット(ST)に"0"を書き込む(RTC 停止)ことは禁止です。

25.4.2 サブセカンドレジスタ ：WTBR

サブセコンドレジスタ (WTBR) のビット構成について説明します。

サブ秒カウンタ(22 ビットダウンカウンタ)のリロード値を格納するためのレジスタです。

■ WTBRH：アドレス 0565H (アクセス：バイト)

■ WTBRM：アドレス 0566H (アクセス：バイト)

■ WTBRL：アドレス 0567H (アクセス：バイト)

WTBRH

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ー	ー	D21	D20	D19	D18	D17	D16
初期値	ー	ー	X	X	X	X	X	X
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

WTBRM

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D15	D14	D13	D12	D11	D10	D9	D8
初期値	X	X	X	X	X	X	X	X
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

WTBRL

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	X	X	X	X	X	X	X	X
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

サブセカンドレジスタは、サブ秒カウンタ(22 ビットダウンカウンタ)で使用するリロード値を格納します。

この値は、サブ秒カウンタ(22 ビットダウンカウンタ)が"0"になるとリロードされます。サブセカンドレジスタを修正する場合は、書込み命令中にリロード動作が行われていないことを確認してください。そうしないと、サブ秒カウンタ(22 ビットダウンカウンタ)は、新旧のデータバイトを結合した正しくない値をロードしてしまいます。一般的には、サブセカンドレジスタは、ST ビットが"0"の間に更新することが推奨されています。サブセカンドレジスタを"0"に設定した場合、サブ秒カウンタ(22 ビットダウンカウンタ)は一切動作しません。

0.5 秒をカウントさせるためのサブ秒レジスタの設定は、以下のとおりです。

表 25-2 WTBR 設定例

RTC クロックの周波数	WTBR 設定値
32kHz	0x001F3F
4MHz	0x0F423F

25.4.3 日時分秒 レジスタ：WTDR/WTHR/WTMR/WTSR

日時分秒レジスタ (WTDR/WTHR/WTMR/WTSR) のビット構成について説明します。

リアルタイムクロックの時間情報 (日/時/分/秒) を表しているレジスタです。

- **WTDR (日レジスタ) ：アドレス 055E$_H$ (アクセス：ハーフワード)**
- **WTHR (時レジスタ) ：アドレス 0568$_H$ (アクセス：バイト, ハーフワード)**
- **WTMR (分レジスタ) ：アドレス 0569$_H$ (アクセス：バイト, ハーフワード)**
- **WTSR (秒レジスタ) ：アドレス 056A$_H$ (アクセス：バイト)**

WTDR

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	N15	N14	N13	N12	N11	N10	N9	N8
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	N7	N6	N5	N4	N3	N2	N1	N0
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

WTHR

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	―	―	―	H4	H3	H2	H1	H0
初期値	―	―	―	0	0	0	0	0
属性	R1,WX	R1,WX	R1,WX	R,W	R,W	R,W	R,W	R,W

WTMR

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	―	―	M5	M4	M3	M2	M1	M0
初期値	―	―	0	0	0	0	0	0
属性	R1,WX	R1,WX	R,W	R,W	R,W	R,W	R,W	R,W

WTSR

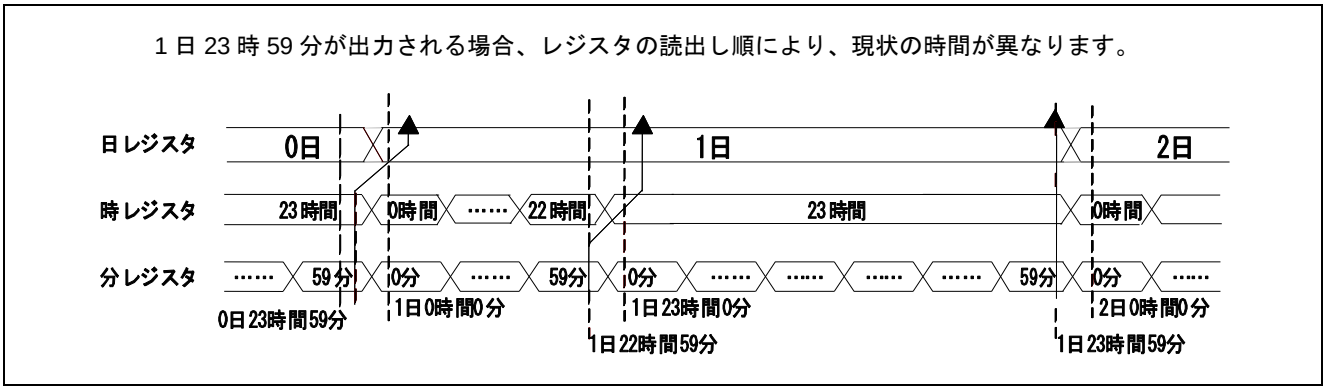
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ー	ー	S5	S4	S3	S2	S1	S0
初期値	ー	ー	0	0	0	0	0	0
属性	R1,WX	R1,WX	R,W	R,W	R,W	R,W	R,W	R,W

本レジスタはパワーオンリセット要因でのみ初期化されます。

- 秒/分/時/日レジスタは、日時間情報を格納します。秒，分，時および日の 2 進表記です。
- レジスタを読み出すと、カウンタ値が読み出せます。書込みデータは、UPDT ビットを"1"に設定した後に、カウンタにロードされます。
- ワードアクセスはできませんので、それぞれのレジスタごとにアクセスしてください。
- 日数レジスタも同様に、ワードアクセスはできません。なお、日数レジスタは、16 ビットカウンタで日数をカウントしていますので、必ずハーフワードでアクセスしてください。バイトでアクセスした場合、読出し中に桁上りが生じて、読出し値が不適切な値となる可能性がありますので、バイトアクセスおよびワードアクセスは禁止です。

- 時分秒レジスタへの設定は、下記範囲で設定してください。
  時間(WTHR)： $0～17_H$ (0 時間～23 時間)
  分(WTMR)： $0～3B_H$ (0 分～59 分)
  秒(WTSR)： $0～3B_H$ (0 秒～59 秒)
- 日時分秒レジスタの 4 つのレジスタから出力された値に、矛盾がないことを確認してください。下記例の可能性があります。
  [例] 出力値 「1 日， 23 時間,59 分， 59 秒」，「0 日， 23 時間，59 分， 59 秒」，
  「1 日， 0 時間， 0 分， 0 秒」， 「1 日， 22 時間，59 分， 59 秒」，
  「1 日， 23 時間，0 分， 0 秒」， 「2 日， 0 時間， 0 分， 0 秒」

図 25-3 日時分秒レジスタ遷移図



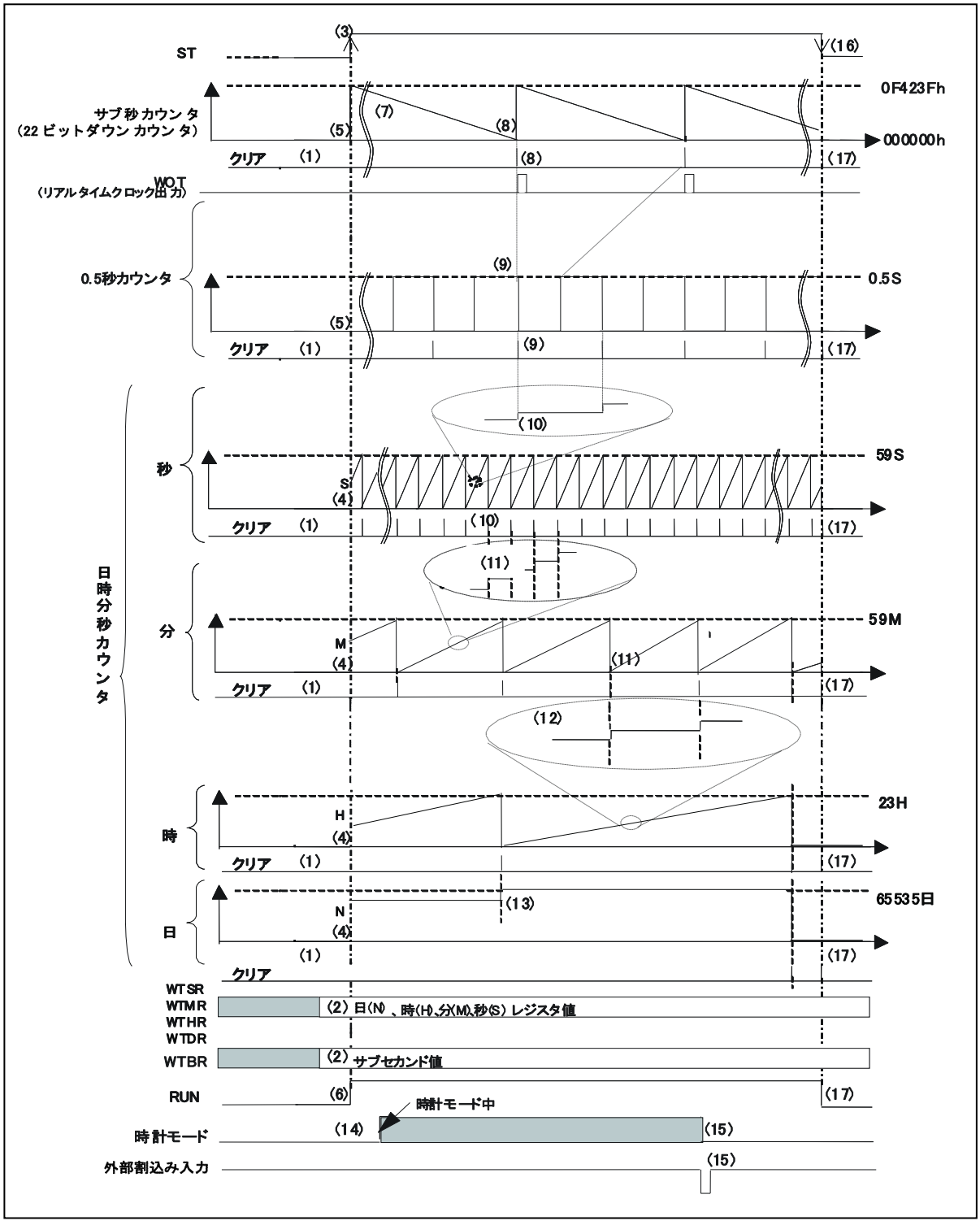
- 動作クロックがメインクロックの 2 分周の場合（PLL が停止している場合)、時分秒レジスタからの読出し値は誤っている場合があります。これは、読出し動作とカウント動作の同期調整のために起こります。したがって、読出し命令のトリガには、秒割込みを使用することを推奨します。
- カウンタが停止した時間を初期値として、再度動作開始する場合には、再起動する前に日時分秒レジスタを読み出し、その値を再度、日時分秒レジスタへ書き込んだ後開始してください。
- 本シリーズでは、RTC 検出リセットの機能は、搭載しておりませんので、パワーオンリセット発生でのみ、日時分秒レジスタはクリアされます。したがって、マイコン部内部低電圧検出フラグがセットされた場合、日時分秒レジスタをクリアすることを推奨します。

25.5 動作説明

リアルタイムクロック (RTC) の動作について説明します。

リアルタイムクロックの動作について説明をします。

図 25-4 リアルタイムクロック動作説明



(1) スタートビット(ST="0")により、サブ秒カウンタ(22 ビットダウンカウンタ), 日時分秒タイマがリセット(0)して停止。

(2) ・ソフトウェアで日時分秒の値を日時分秒レジスタ WTDR, WTHR, WTMR, WTSR に書き込む。
・ソフトウェアでサブセカンドレジスタ WTBRH, WTBRM, WTBRL に値"0F$_H$", "42$_H$", "3F$_H$"を書き込む。
・割込み要求ビット(INT0, INT1, INT2, INT3,INT4)を初期化, 割込み要求許可ビット(INTE0, INTE1, INTE2, INTE3,INTE4)を設定(使用する割込みを許可)する。

(3) スタートビット(ST)を"1"にする。

(4) スタートビット(ST="1")により、日時分秒レジスタ WTDR, WTHR, WTMR, WTSR の値を、日時分秒タイマにロードする。

(5) さらに、サブ秒カウンタ(22 ビットダウンカウンタ)のカウント値が"000000$_H$"のため、セカンドレジスタ WTBRH, WTBRM, WTBRL の値を、サブ秒カウンタ(22 ビットダウンカウンタ)にロードする。

(6) 動作フラグ(RUN)が"1"になる。

(7) サブ秒カウンタ(22 ビットダウンカウンタ)がメインクロックを 2 分周したクロック (4/2MHz) でカウントを始める。

(8) サブ秒カウンタ(22 ビットダウンカウンタ)が"000000$_H$"になると、サブセカンドレジスタ値"0F423$_H$"をサブ秒カウンタ(22 ビットダウンカウンタ)にロードする。
また、0.5 秒カウンタの割込み要求が発生する。
さらに、リアルタイムクロック出力許可(WOT 端子出力許可)に設定した場合、WOT 端子にメインクロックの 2 分周幅分の"H"レベルが出力される。
(例:メインクロック 4MHz の場合,500ns 幅の"H"出力)

(9) 0.5 秒カウンタがカウントアップした後、次のカウントアップでクリアされ、日時分秒カウンタの秒カウンタがカウントアップするとともに、秒の割込み要求が発生する。

(10) 日時分秒カウンタの秒カウンタがカウントアップしていき、値が"59"になると次のカウントアップでクリアし、分のカウンタがカウントアップするとともに、このとき分の割込み要求が発生する。

(11) 日時分秒カウンタの分カウンタがカウントアップしていき、値が"59"になると次のカウントアップでクリアし、時のカウンタがカウントアップするとともに、このとき時の割込み要求が発生する。

(12) 日時分秒カウンタの時カウンタがカウントアップしていき、値が"23"になると次のカウントアップでクリアし、日のカウンタがカウントアップするとともに、このとき日割込み要求が発生する。

(13) 日時分秒カウンタの日カウンタがカウントアップしていき、値が"65535"になると次のカウントアップでクリアする。

(14) ソフトウェアで時計モードに遷移させる。
時計モード中でもリアルタイムクロックは動作を継続。

(15) 割込み端子(INTxx)端子より信号を入力し、時計モードから復帰させ、CPU を起動させる。

(16) スタートビット(ST)を"0"にする。

(17) スタートビット ST="0"により、サブ秒カウンタ(22 ビットダウンカウンタ)、日時分秒カウンタがクリア(リセット) して停止。

25.6 設定

リアルタイムクロック (RTC) の設定について説明します。

表 25-3 リアルタイムクロックを動かすために必要な設定

設定	設定レジスタ	設定方法
リロード値 (サブセカンドレジスタ) の設定	サブセカンドレジスタ (WTBRH, WTBRM, WTBRL)	25.7.1 参照
リアルタイムクロックの初期化	RTC 制御レジスタ(WTCR)	25.7.2 参照
日数, 時刻 (日時分秒) の設定	日時分秒レジスタ (WTDR,WTHR, WTMR, WTSR)	25.7.3 参照
リアルタイムクロックの起動	RTC 制御レジスタ(WTCR)	25.7.4 参照

表 25-4 時刻を知るための必要設定項目

設定	設定レジスタ	設定方法
日数, 時刻の読出し	日時分秒レジスタ (WTDR,WTHR, WTMR, WTSR)	25.7.6 参照

表 25-5 リアルタイムクロックを停止させるための必要な設定

設定	設定レジスタ	設定方法
リアルタイムクロックの停止	RTC 制御レジスタ(WTCR)	25.7.7 参照

表 25-6 リアルタイムクロック割込みを行うために必要な設定

設定	設定レジスタ	設定方法
RTC 割込みベクタ, RTC 割込みレベルの設定	『割込み制御(割込みコントローラ)』の章を参照	25.7.10 参照
RTC 割込み設定 割込み要求のクリア 割込み要求の許可	RTC 制御レジスタ(WTCR)	25.7.11 参照

25.7 Q&A

リアルタイムクロック (RTC) の Q&A について説明します。

25.7.1. 0.5 秒のカウント周期を設定するには?

25.7.2. リアルタイムクロックを初期化するには?

25.7.3. 日数 (日), 時刻 (時/分/秒) を設定/更新するには?

25.7.4. リアルタイムクロックのカウントを起動/停止するには?

25.7.5. リアルタイムクロックが動作しているか確認するには?

25.7.6. 日数,時刻を知るには?

25.7.7. リアルタイムクロックを停止させるには?

25.7.8. リアルタイムクロックを補正するには？

25.7.9. 割込み関連レジスタは?

25.7.10. 割込みの種類と選択方法は?

25.7.11. 割込みを許可するには?

25.7.1 0.5 秒のカウント周期を設定するには?

0.5 秒のカウント周期設定について説明します。

リアルタイムクロックを停止させてから、RTC クロック周波数に応じてサブセカンドレジスタ (WTBR) を表 25-2 WTBR 設定例に示す値に設定することで行います。

25.7.2 リアルタイムクロックを初期化するには?

リアルタイムクロックの初期化について説明します。

スタートビット(WTCR:ST)にて設定してください。

スタートビットを"1"から"0"に書き換えると、時分秒カウンタとサブ秒カウンタ(22 ビットダウンカウンタ)のすべてが、"0"にリセット (初期化) し、カウント動作を停止します。

25.7.3 日数 (日), 時刻 (時/分/秒) を設定/更新するには?

日数(日), 時刻 (時/分/秒)の設定/更新について説明します。

日時分秒レジスタ (WTDR, WTHR, WTMR, WTSR) に値を書込み、更新ビット (UPDT) で設定してください。

動作	更新ビット(UPDT)
日時分秒カウンタを更新するには	"1"にする

25.7.4 リアルタイムクロックのカウントを起動/停止するには?

リアルタイムクロックのカウントの起動/停止について説明します。

スタートビット(WTCR:ST)にて設定してください。

動作	スタートビット(ST)
リアルタイムクロックのカウントを停止させるには	"0"にする
リアルタイムクロックのカウントを動作させるには	"1"にする

25.7.5 リアルタイムクロックが動作しているか確認するには?

リアルタイムクロックの動作確認について説明します。

動作フラグ(WTCR:RUN)にて確認してください。

動作	動作フラグ(RUN)
リアルタイムクロックは停止している	読出し値が"0"
リアルタイムクロックは動作している	読出し値が"1"

25.7.6 日数,時刻を知るには?

日数, 時刻の確認について説明します。

日時分秒レジスタ WTDR,WTHR, WTMR, WTSR を読み出すことで確認できます。

ただし、ワードアクセスはできませんのでそれぞれのレジスタごとにアクセスしてください。時/分のカウントの境目で読み込んだ場合、時刻を見誤る危険性がありますので、複数回読み出して論理的に正しい時間を使用してください。

例：

秒から読み込んだ場合：
1 日 2 時 59 分 59 秒 => 1 日 3 時 59 分 59 秒 => 1 日 3 時 0 分 0 秒

時から読み込んだ場合：
1 日 2 時 59 分 59 秒 => 1 日 2 時 0 分 0 秒 => 1 日 3 時 0 分 0 秒

25.7.7 リアルタイムクロックを停止させるには?

リアルタイムクロックの停止について説明します。

「25.7.4」を参照してください。

25.7.8 リアルタイムクロックを補正するには？

リアルタイムクロックの補正について説明します。

RTC クロックとしてサブクロック(2 系統品のみ)が選択されている場合、メインクロック：サブクロックの比率を用いて補正することができます。『RTC／WDT1 補正』の章を参照してください。

25.7.9 割込み関連レジスタは?

割込み関連レジスタについて説明します。

RTC 割込みベクタ, RTC 割込みレベルの設定

割込みレベル, 割込みベクタの関係は下表のとおりです。

割込みレベル, 割込みベクタの詳細については『割込み制御(割込みコントローラ)』の章を参照してください。

割込みベクタ (デフォルト)	割込みレベル設定ビット(ICR[4:0])
#37 (0FFF68H)	割込みレベルレジスタ ICR21 (00455H)

割込み要求フラグ(INT0, INT1, INT2, INT3, INT4)は自動的にクリアしません。このため、割込み処理から復帰する前にソフトウェアにてクリアしてください。 (INT0, INT1, INT2, INT3, INT4 ビットに"0"を書き込む)

25.7.10 割込みの種類と選択方法は?

割込みの種類と選択方法について説明します。

割込み要因には, 以下の 5 つがあります。

割込み要因	割込み要求ビット	割込み要求許可ビット
時刻 (1 秒) カウントのタイミング	INT0	INTE0
時刻 (分) カウントのタイミング	INT1	INTE1
時刻 (時) カウントのタイミング	INT2	INTE2
1 日のカウントのタイミング	INT3	INTE3
時刻(0.5 秒)のカウントのタイミング	INT4	INTE4

5 つの要因の OR で割り込みますので、割込み要求許可ビットで選択してください。

25.7.11 割込みを許可するには?

割込みの許可について説明します。

割込み要求許可ビット (WTCR:INTE0, WTCR:INTE1, WTCR:INTE2, WTCR:INTE3, WTCR:INTE4)にて行ってください。

動作	設定方法 割込み要求許可ビット (INTE0, INTE1, INTE2, INTE3,INTE4)
割込みを禁止するには	"0"にする
割込みを許可するには	"1"にする

割込み要求のクリアは, 割込み要求ビット (WTCR:INT0, WTCR:INT1, WTCR:INT2, WTCR:INT3, WTCR:INT4) にて行ってください。

動作	設定方法 割込み要求ビット (INT0, INT1, INT2, INT3, INT4)
割込み要求をクリアするには	"0"を書き込む

25.8 サンプルプログラム

リアルタイムクロックのサンプルプログラムについて説明します。

設定手順例 1

リアルタイムクロックを10 日10 時10 分00 秒からカウント開始し, "H" レベル検出の外部割込み(INT0) を許可して, 時計モードに遷移。
外部割込み検出で時計モードから復帰して, リアルタイムクロックの時間を読み出す。

RTC 初期化
RTC 起動、割込みレベル設定
外部割込み設定
時計モードへ遷移
時計モード復帰後、 RTC の読出し

<RTC 初期設定>

・RTC

レジスタの初期化	WTCR.ST
インターバル時間(1 秒)の設定	WTBR
時刻の初期値の設定	WTSR WTMR WTHR WTDR
RTC 割込みの初期設	WTCRM,WTCRL WTCRH

<RTC 起動、割込みレベル設定>

RTC 起動	WTCR.ST
割込みレベルの設定(RTC)	ICR21
割込みレベル設定 (INT0)	ICR00
I フラグの設定	(CCR)

<RTC 時刻読出し準備(割込み設定)>

RTC 割込み設定	WTCR .INT0 .INTE0

<RTC 割込み>

時刻の読出し	WTHR WTMR WTSR WTDR
割込み禁止	WTCR.INTE0

<外部割込み>

割込み要求フラグのクリア	EIRR.ER0

<割込みベクタ>
ベクタテーブルの設定

<その他>
(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。「クロック」の章および「割込み制御(割込みコントローラ)」の章を参照してください。

プログラム例 1

```
void RTC_sample1(void)
{
        RTC_initial();
        RTC_start();
        EX_INT0_initial();              /* 外部割込み設定のサブルーチン*/
        STOP_Hiz_hold_with_clock(); /* 時計モード遷移のサブルーチン*/
        RTC_read();
}



void RTC_initial(void)
{
        IO_WTCR.bit.ST = 1;                             /* 初期化準備 */
        IO_WTCR.bit.ST = 0;                             /* 停止( レジスタ初期化) */
        IO_WTBR.word = 0x0F423F;                        /* カウント値設定 4MHz/2 × 0x0F423F=0.5 秒 */
        IO_WTSR.byte = 0x00;                            /* 秒設定 */
        IO_WTMR.byte = 0x0A;                            /* 分設定 */
        IO_WTHR.byte = 0x0A;                            /* 時設定 */
        IO_WTDR.hword = 0x000A;                         /* 日設定 */
        IO_WTCRL.hword = IO_WTCRL.hword & 0x0000; /* 割込みフラグクリア, 割込み禁止 */
        IO_WTCRH.byte = 0x00                            /* 割込みフラグクリア, 割込み禁止 */
}

void RTC_start(void)
{
        IO_WTCR.bit.ST = 1;         /* RTC 起動 */
        IO_ICR[21].bit.ICR = 18;    /* 値は任意 */
        IO_ICR[00].bit.ICR = 20;    /* 値は任意 */
        __EI();                     /* 割込み許可 */
}

RTC_read(void)
{
        IO_WTCR.bit.INT0 = 0;       /* RTC 秒割込み要求フラグクリア */
        IO_WTCR.bit.INTE0 = 1;      /* RTC 秒割込み要求許可 */
}

__interrupt void RTC_read_int(void) /* RTC 割込み */
{
        JIKAN(char) = IO_WTHR.byte & 0x1F;    /* 時 */
        FUNN(char) = IO_WTMR.byte & 0x3F;     /* 分 */
        BYOU(char) = IO_WTSR.byte & 0x3F;     /* 秒 */
        HI(char) = IO_WTDR.hword ;            /* 日 */
                                              /* 複数回読出し */
        IO_WTCR.bit.INTE0 = 0;                /* RTC 割込み禁止 */
}

__interrupt void INT0_int() /* 外部割込み */
{
        IO_EIRR0.bit.ER0= 0; /* ER0 秒割込み要求フラグクリア */
}
```

25.9 注意事項

リアルタイムクロックの注意事項について説明します。

- 割込み要求フラグ(WTCR:INT0, WTCR:INT1, WTCR:INT2, WTCR:INT3, WTCR:INT4) が、オーバフローの発生で"1"にセットされるタイミングと、"0"を書き込むタイミングが同時の場合、フラグは"1"になります。(フラグのセットが優先)
- サブセカンドレジスタ (WTBRH,WTBRM,WTBRL) の更新途中でリロードが発生した場合、サブ秒カウンタ(22 ビットダウンカウンタ)に意図しない値をリロードする可能性があります。したがって、サブセカンドレジスタ(WTBR) は、スタートビット(WTCR:ST)が"0"の状態で更新することを推奨します。
- サブセカンドレジスタ (WTBRH,WTBRM,WTBRL) をすべて"0"に設定すると、サブ秒カウンタ(22 ビットダウンカウンタ)は動作しないため、リアルタイムクロックは動作しません。
- 日時分秒レジスタ (WTDR,WTHR, WTMR, WTSR) の読出し中に桁上りが生じて、読出し値が不適切な値となる可能性があります。このため、日数時刻 (日/時/分/秒) の読出しは、 割込み (INT0) を用いることを推奨します。
- 日時分秒レジスタ(WTDR,WTHR, WTMR, WTSR)は、ワードアクセスはできませんので、それぞれのレジスタごとにアクセスする必要があります。そのため、読込みを時/分のカウントの境目で読み込んだ場合、時刻を見誤る危険性がありますので、複数回読み出して論理的に正しい時間を使用してください。

  例：
  秒から読み込んだ場合：
  1 日 23 時 59 分 59 秒 => 2 日 0 時 59 分 59 秒 => 2 日 0 時 0 分 0 秒
  時から読み込んだ場合：
  1 日 23 時 59 分 59 秒 => 2 日 23 時 0 分 0 秒 => 2 日 0 時 0 分 0 秒
  日から読み込んだ場合：
  1 日 23 時 59 分 59 秒 => 1 日 0 時 0 分 0 秒 => 2 日 0 時 0 分 0 秒

  このような場合には、2 日 0 時と判断してください。

- 日時分秒レジスタは内部リセットでクリアされませんが、日時分秒カウンタは、内部リセットでクリアされます。内部リセット発生後、ST フラグはクリアされ、RTC マクロは停止状態です。なお、日時分秒レジスタには、内部リセット発生前のカウンタ値がセットされています。内部リセット発生時の日時分秒を使用する場合、日時分秒カウンタを読み出した値を、日時分秒レジスタにセットしてください。
- 日数レジスタは、日数を"0 日"~"65535 日"までカウントする機能を搭載しています。
- RTC 制御レジスタ設定時の注意
  - ☐ RTC 停止の状態(ST=0)からスタートビット(ST)への"1"書込み(RTC 動作開始)の際には、スタートビットと同時に更新ビット(UPDT)への"1"書込みを行わないでください。
    (ST=0 状態での、バイト即値による ST,UPDT ビットの同時"1"書込みは禁止です。)
  - ☐ 更新ビット(UPDT)に"1"を書き込む場合は、RTC 動作中(ST=1)の状態で行ってください。
  - ☐ 更新ビット(UPDT)が"1"の状態で、スタートビット(ST)に"0"を書き込む(RTC 停止)ことは禁止です。
- スタンバイ時計モード(電源遮断)からの復帰時には RTC のレジスタは初期化されません。
- また、スタンバイ時計モード(電源遮断)からの復帰中は、内部リセットが発行されており、パワーオンリセット、内部電源低電圧リセットおよび RSTX・NMIX 同時アサートによるリセット以外のリセット要因を受付けません。このとき、RTC のレジスタは初期化されませんので、復帰後に RSTX 端子入力からのリセット入力、または外部低電圧検出フラグがセットされている場合には、RTC のレジスタを初期化してから使用してください。

26. RTC／WDT1 補正（キャリブレーション）

RTC/WDT1 補正 (キャリブレーション) について説明します。

26.1 概要

RTC/WDT1 補正 (キャリブレーション) の概要について説明します。

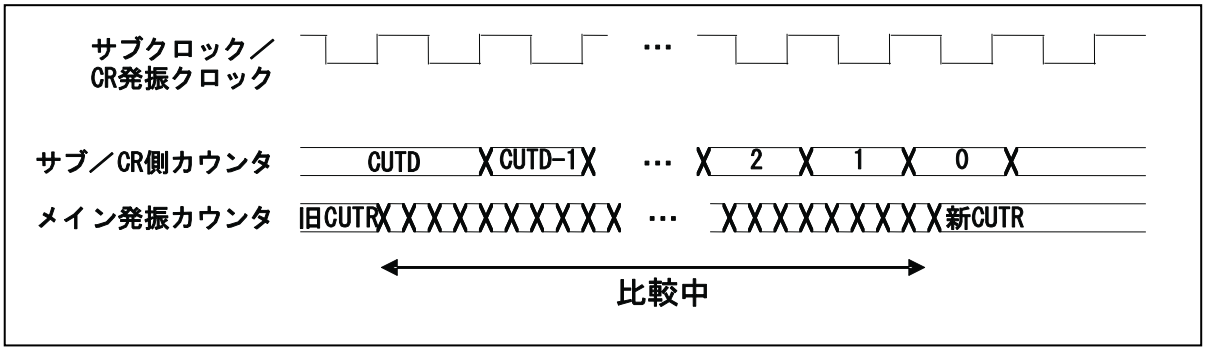
本モジュールを使用して、リアルタイムクロック、WDT1 および CSV 用搭載 CR 発振回路の周波数の補正のための値を計算します。

26.2 特長

RTC/WDT1 補正 (キャリブレーション)の特長について説明します。

- RTC クロックソース選択レジスタ
  - 選択方法は『クロック』の章を参照してください。メインクロック、またはサブクロック(2 系統品のみ)を選択できます。
- リアルタイムクロック (RTC) 補正 (2 系統品のみ。サブクロック使用時のみ効果のある機能です。)
  - メインクロック駆動カウンタとサブクロック駆動カウンタを同時動作させ (図 26-1)、それによってサブクロックの周波数をメインクロックの周波数から計算して、RTC のプリスケーラ値を設定します。
- WDT1 (CR クロック) 補正
  - メインクロック駆動カウンタと CR クロック駆動カウンタを同時動作させ (図 26-1)、それによって CR クロック周波数をメインクロックの周波数から計算して、CR クロックトリミング値を設定します。

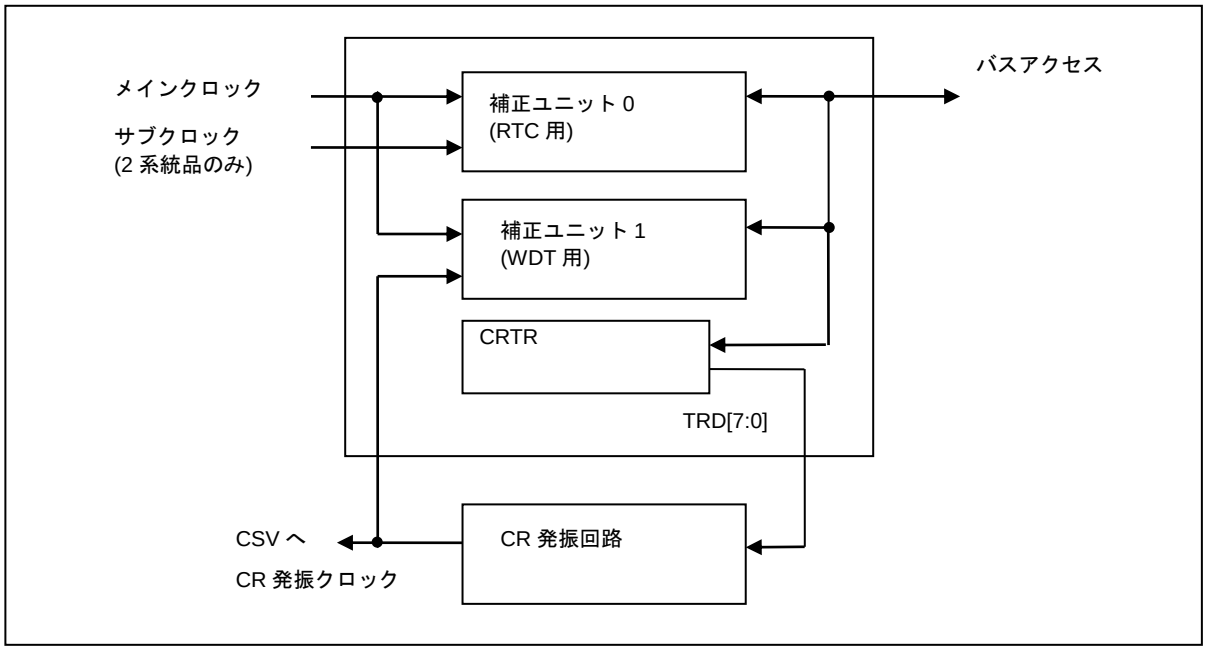
図 26-1 各クロック駆動カウンタ比較



26.3 構成

RTC/WDT1 補正 (キャリブレーション) の構成について示します。

図 26-2 ブロックダイヤグラム



26.4 レジスタ

RTC/WDT1 補正 (キャリブレーション) のレジスタについて説明します。

表 26-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x04B8	CUCR0		CUTD0		補正ユニット制御レジスタ 0 サブクロックタイマデータレジスタ
0x04BC	CUTR0				メイン発振タイマデータレジスタ 0
0x04C0	予約	予約	予約	予約	予約
0x04C4	CUCR1		CUTD1		補正ユニット制御レジスタ 1 CR 発振タイマデータレジスタ
0x04C8	CUTR1				メイン発振タイマデータレジスタ 1
0x04CC	CRTR	予約	予約	予約	CR 発振トリミング設定レジスタ

26.4.1 補正ユニット制御レジスタ 0 : CUCR0: (Calibration Unit Control Register 0)

補正ユニット制御レジスタ 0 (CUCR0)のビット構成について説明します。

RTC 用補正ユニットの補正開始、割込みの設定をします。

■ CUCR0 : アドレス 04B8H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	1	1	1	1	1	1	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	予約		STRT	予約		INT	INTEN
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,WX	R0,WX	R,W	R0,WX	R0,WX	R,W	R/W

[bit7] 予約
必ず"0"を書き込んでください。

[bit4] STRT (calibration STaRT) : 補正開始
メインクロック、サブクロック駆動カウンタを開始させます。比較が完了すると、INT ビットがセットされます。

STRT	機能
"0"書込み	比較を中止します
"1"書込み	比較を開始します

"0"をセットすると、比較を中止します。比較中に"1"書込みを行っても影響はありません。比較が完了すると、本ビットは"0"にクリアされます。

[bit1] INT (calibration INTerrupt) : 割込み
比較が完了すると、INT ビットが"1"にセットされます。INTEN ビットがセットされている場合は、割込みが発生します。本ビットは"0"書込みでクリアされます。

[bit0] INTEN (calibration INTerrupt ENable) : 割込み許可
INT ビットがセットされた際の、割込みの有無を設定します。

INTEN	割込み
0	無効
1	有効

26.4.2 サブクロックタイマデータレジスタ ：CUTD0 (Calibration Unit Timer Data register 0)

サブクロックタイマデータレジスタ (CUTD0) のビット構成について説明します。

サブクロック駆動カウンタ駆動期間を設定します。

■ CUTD0： アドレス 04BA$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TDD[15:8]							
初期値	1	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TDD[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit0] TDD[15:0] (Timer Data Data field) : **タイマデータ**
比較期間をサブクロック数で設定します。

26.4.3 メイン発振タイマリザルトレジスタ 0 : CUTR0 (Calibration Unit Timer Result register 0)

メイン発振タイマリザルトレジスタ 0 (CUTR0) のビット構成について説明します。

CUTD0 で設定された期間内の、メインクロック駆動カウンタ数を表示します。

■ CUTR0 : アドレス 04BCH (アクセス: バイト, ハーフワード, ワード)

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	TDR[23:16]							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TDR[15:8]							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TDR[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit31～bit24] 予約
必ず"0"が読み出されます。書込みは他の機能に影響しません。

[bit23～bit0] TDR[23:0] (Timer Data Register) : タイマデータ
比較期間内でカウントされたカウント数を表示します。比較完了後に読み出してください。比較中の読出し値は不定です。書込みは動作に影響ありません。

26.4.4 補正ユニット制御レジスタ 1 : CUCR1 (Calibration Unit Control Register 1)

補正ユニット制御レジスタ 1 (CUCR1) のビット構成について説明します。

WDT 用補正ユニットの補正開始、割込みの設定をします。

■ CUCR1 : アドレス 04C4H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	1	1	1	1	1	1	1	1
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	予約		STRT	予約		INT	INTEN
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,WX	R0,WX	R,W	R0,WX	R0,WX	R,W0	R/W

[bit7] 予約
必ず"0"を書き込んでください。

[bit4] STRT (calibration STaRT) : 補正開始
メインクロック、CR クロック駆動カウンタを開始させます。比較が完了すると、INT ビットがセットされます。

STRT	機能
"0"書込み	比較を中止します
"1"書込み	比較を開始します

"0"をセットすると、比較を中止します。比較中に"1"書込みを行っても動作に影響はありません。比較が完了すると、本ビットは"0"にクリアされます。

[bit1] INT (calibration INTerrupt) : 割込み
比較が完了すると、INT ビットが"1"にセットされます。INTEN ビットがセットされている場合は、割込みが発生します。本ビットは"0"書込みでクリアされます。

[bit0] INTEN (calibration INTerrupt Enable) : 割込み許可
INT ビットがセットされた際の、割込みの有無を設定します。

INTEN	割込み
0	無効
1	有効

26.4.5 CR クロックタイマデータレジスタ ：CUTD1 (Calibration Unit Timer Data register 1)

CR クロックタイマデータレジスタ (CUTD1) のビット構成について説明します。

CR クロック駆動カウンタ駆動期間を設定します。

■ CUTD1： アドレス 04C6H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TDD[15:8]							
初期値	1	1	0	0	0	0	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TDD[7:0]							
初期値	0	1	0	1	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit0] TDD[15:0] (Timer Data Data field) : **タイマデータ**
比較期間を CR クロック数で設定します。

26.4.6 メイン発振タイマリザルトレジスタ 1 : CUTR1 (Calibration Unit Timer Result register 1)

メイン発振タイマリザルトレジスタ 1 (CUTR1) のビット構成について説明します。

CUTD1 で設定された期間内の、メインクロック駆動カウンタ数を表示します。

■ CUTR1 : アドレス 04C8H (アクセス: バイト, ハーフワード, ワード)

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	TDR[23:16]							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TDR[15:8]							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TDR[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit31～bit24] 予約
必ず"0"が読み出されます。書込みは他の機能に影響しません。

[bit23～bit0] TDR[23:0] (Timer Data Register) : タイマデータ
比較期間内でカウントされたカウント数を表示します。比較完了後に読み出してください。比較中の読出し値は不定です。書込みは無効です。

26.4.7 CR 発振トリミング設定レジスタ ：CRTR (CR oscillator calibration Trimming Register)

CR 発振トリミング設定レジスタ (CRTR) のビット構成について説明します。

CR 発振回路のトリミング値を設定します。

■ CRTR： アドレス 04CC$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TRD[7:0]							
初期値	0	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] TRD[7:0] (Trimming Data) : トリミングデータ

トリミング値 約 0.4%ステップ

TRD7	TRD6	TRD5	TRD4	TRD3	TRD2	TRD1	TRD0	トリミング値*	n 値
0	0	0	0	0	0	0	0	-48.01%	0
0	0	0	0	0	0	0	1	-47.61%	1
0	0	0	0	0	0	1	0	-47.23%	2
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
0	1	1	1	1	1	1	1	0%[初期値]	127
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
1	1	1	1	1	1	0	1	+45.62%	253
1	1	1	1	1	1	1	0	+45.98%	254
1	1	1	1	1	1	1	1	+46.37%	255
*: 温度などの条件で変化するので、「26.5.2」 で説明する手順を踏む必要があります。									

26.5 動作説明

RTC/WDT1 補正 (キャリブレーション)の動作について説明します。

26.5.1. リアルタイムクロック (RTC) 補正

26.5.2. WDT1 補正 (CR クロック補正)

26.5.3. 注意事項

26.5.1 リアルタイムクロック (RTC) 補正

リアルタイムクロック (RTC) 補正について説明します。

補正手続きは以下のようになります。

1. CUTD0 設定
2. CUCR0:INTEN セット
3. CUCR0:STRT セット
4. 割込み待ちループ
5. 割込み発生
6. CUTR0 読出し
7. CUTR0:CUTD0 で、メインクロック周波数 : サブクロック周波数の比が計算できます。
8. RTC 内プリスケーラ値を(7)で計算した値を利用して設定します。

26.5.2 WDT1 補正 (CR クロック補正)

WDT1 補正 (CR クロック補正)について説明します。

トリミング値の算出は以下のように行ってください。

1. TRD[7:0]を 00000000 に設定し、補正ユニットを動作させて、CUTR の値を得ます。この CUTR 値から計算した CR 発振周波数を Fmin とします。
2. TRD[7:0]を 11111111 に設定し、補正ユニットを動作させて、CUTR の値を得ます。この CUTR 値から計算した CR 発振周波数を Fmax とします。
3. 次式の n に 0 から 255 を代入し、Fer が最小になるポイントの n がトリミング値です。
   Fstep = (Fmax－Fmin)÷255
   Fer = | (100kHz)－(Fmin＋Fstep×n) |

26.5.3 注意事項

RTC/WDT1 補正 (キャリブレーション) の注意事項について説明します。

スタンバイモード移行などの要因が入った場合のカウンタ値は無効になります。STRT ビットに"0"書込みを行い停止させてから、もう一度"1"を書込み、やりなおしてください。

TOSC32/OSC100 > 2 x TOSC4 ＋ 3 x TCLKP を満たす必要があります。

TOSC4 : メインクロック周期

TOSC32 : サブクロック周期

TOSC100 : CR 発振回路発振周期

TCLKP : 周辺クロック発振周期

27. 消費電力制御

消費電力制御について説明します。

27.1 概要

消費電力制御の概要について説明します。

本シリーズは、多彩な低消費電力モードを持ち、状況に応じた消費電力制御を行うことができます。

27.2 特長

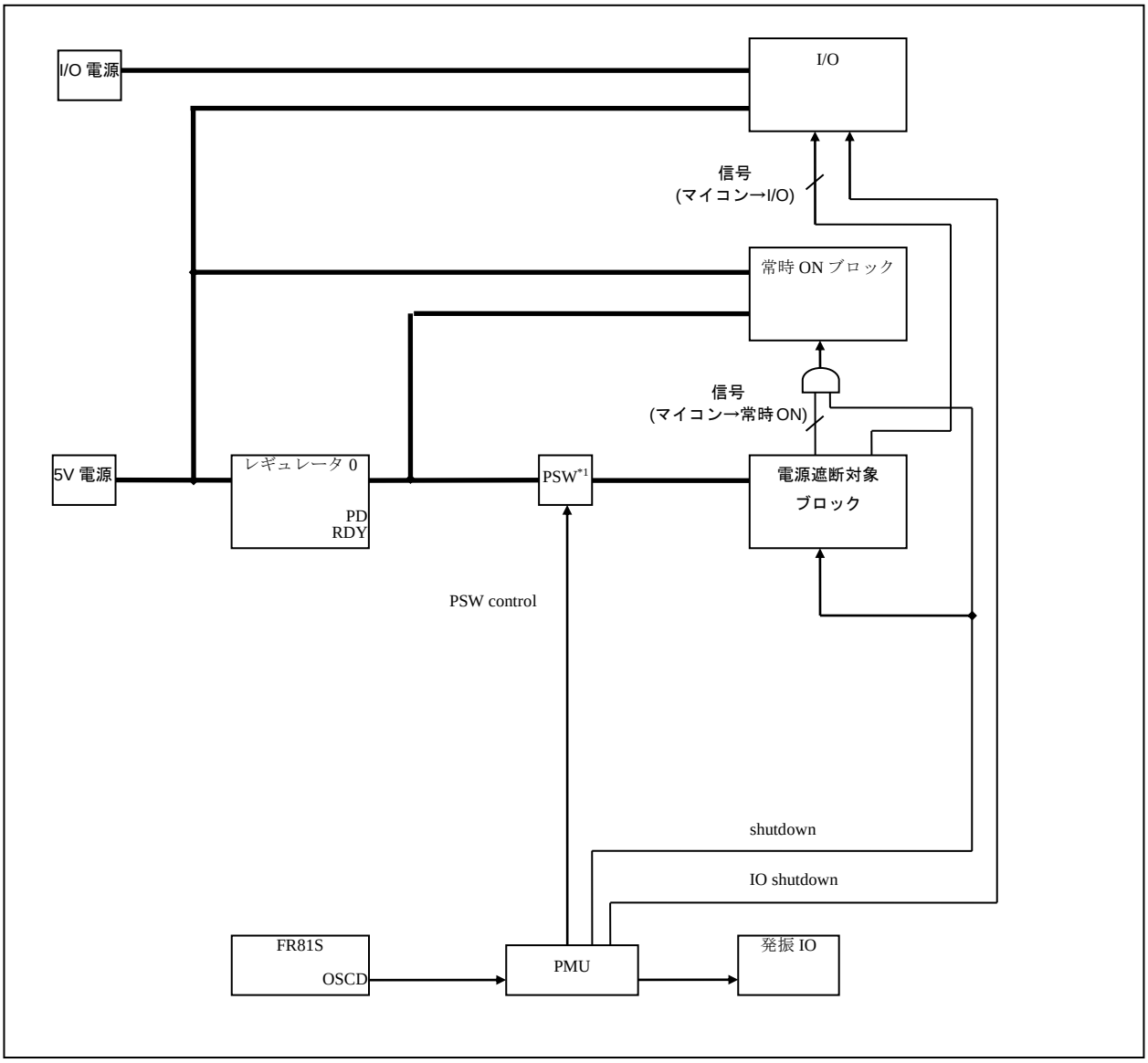
消費電力制御の特長について説明します。

■ クロック制御
クロック分周
各動作クロックの分周比を変更し、動作周波数を落とすことができます。『クロック』の章を参照してください。
■ スリープモード
CPU スリープモード
CPU 動作のみを停止させることができます。
バススリープモード
CPU およびオンチップバスの動作を停止させることができます。
■ スタンバイモード
時計モード
特定のクロック発振とタイマ以外のすべての動作を停止させることができます。
ストップモード
すべてのクロック発振と動作を停止させることができます。
■ スタンバイモード(電源遮断)
時計モード(電源遮断)
電源を遮断させるとともに、特定のクロック発振とタイマ以外のすべての動作を停止させることができます。
ストップモード(電源遮断)
電源を遮断させるとともに、すべてのクロック発振と動作を停止させることができます。

27.3 構成

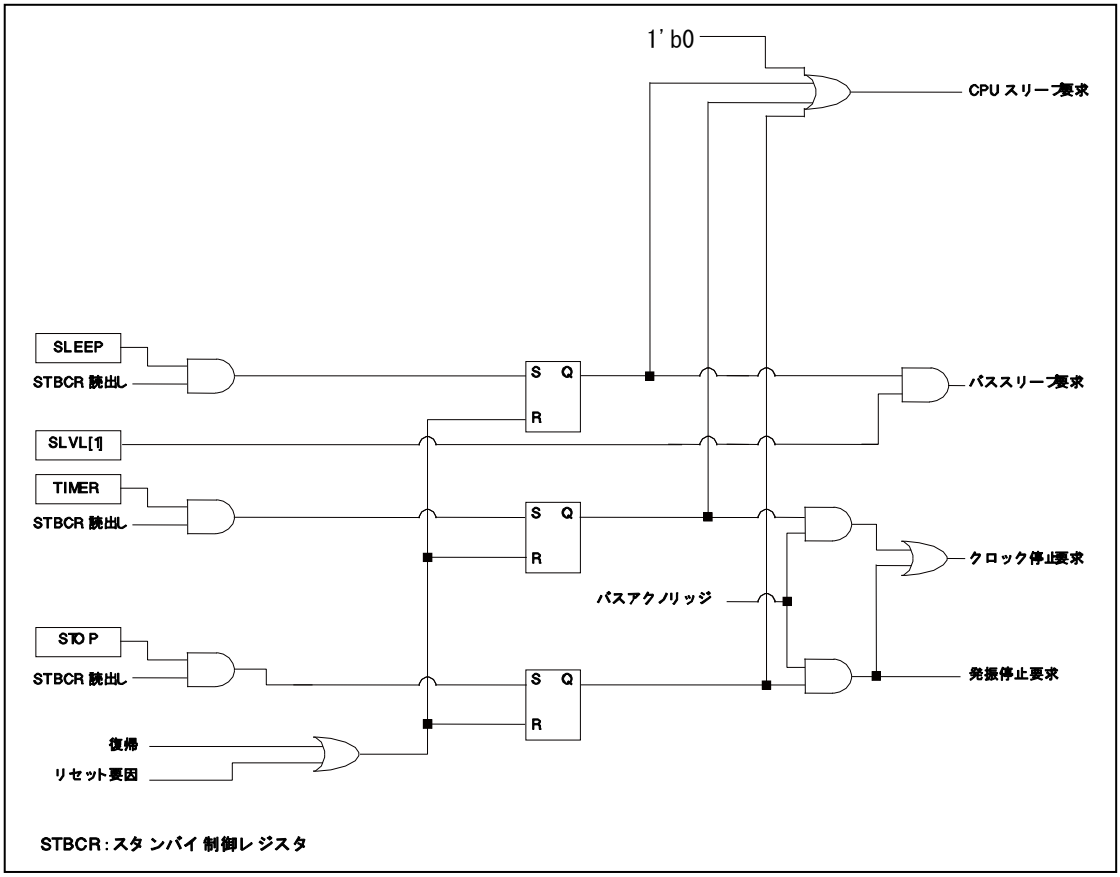
消費電力制御の構成について示します。

図 27-1 全体制御ブロックダイヤグラム



*1: PSW (Power Switch)

図 27-2 全体制御ブロックダイヤグラム



27.4 レジスタ

消費電力制御のレジスタについて説明します。

表 27-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0480	予約	予約	STBCR	予約	スタンバイ制御レジスタ
0x0590	予約	PMUCTLR	PWRTMCTL	予約	PMU コントロールレジスタ PoWeR on TiMing コントロールレジスタ
0x0594	PMUINTF0	PMUINTF1	PMUINTF2	予約	PMU 割込みフラグレジスタ 0～2
0x0598	予約	予約	予約	予約	予約
0x059C	予約	予約	予約	予約	予約

＜注意事項＞

0x0480～0x0481、0x0590 番地には、「リセット」のレジスタが割り当てられているので注意してください。(『リセット』の章を参照してください。)

＜注意事項＞

本レジスタ群(STBCR は除きます)は、以下のいずれかの要因が発生したときのみ初期化されます。

1. パワーオンリセット
2. 内部低電圧検出
3. RSTX 外部端子と NMIX 外部端子の同時アサート
4. ハードウェアウォッチドッグ・リセット

INIT レベルおよび RST レベルのリセットでは初期化されません(STBCR は除きます)。

27.4.1 スタンバイ制御レジスタ ：STBCR (STandby mode Control Register)

スタンバイ制御レジスタのビット構成について説明します。
各低消費電力モードの設定を行います。

＜注意事項＞

本レジスタへの DMA による書込みは禁止です。

■ STBCR： アドレス 0482H (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	STOP	TIMER	SLEEP	予約	予約		SLVL[1:0]	
初期値	0	0	0	0	0	0	1	1
属性	R,W	R,W	R,W	R0,W0	R0,W0	R0,W0	R/W	R/W

[bit7] STOP (STOP mode) : ストップモード許可

[bit6] TIMER (TIMER mode) : 時計モード許可

[bit5] SLEEP (SLEEP mode) : **スリープモード許可**
ストップ/時計/スリープの各スタンバイモードへの遷移を許可します。本ビットに下記に示す値を書き込んだ後、STBCR を読み出すことにより各モードへ遷移します。

STOP	TIMER	SLEEP	各スタンバイモードの遷移許可
0	0	0	遷移しない(初期値)
0	0	1	STBCR 読出しによりスリープモードへ遷移する
0	1	X	STBCR 読出しにより時計モードへ遷移する
1	X	X	STBCR 読出しによりストップモードへ遷移する

なお、各ビットの読出し値は、書き込んだ値にかかわらず以下のとおりとなります。

STOP	TIMER	SLEEP	各スタンバイモードの遷移許可
0	0	0	遷移しない
0	0	1	スリープモードへ遷移する
0	1	0	時計モードへ遷移する
1	0	0	ストップモードへ遷移する

各低消費電力モードからのウェイクアップ要因の発生により、初期値へクリアされます。

[bit4] **予約**
常に"0" が読み出されます。必ず"0" を書き込んでください。

[bit3, bit2] **予約**
常に"0" が読み出されます。必ず"0" を書き込んでください。

[bit1, bit0] SLVL[1:0] (Standby LeVeL) : **スタンバイレベル設定**
各スタンバイモード、スリープモードにおける詳細動作を以下のように制御します。

モード	SLVL[1:0]	各動作制御
ストップモード	0x	端子をハイインピーダンスにしない
	1x	端子をハイインピーダンスにする
時計モード	0x	端子をハイインピーダンスにしない
	1x	端子をハイインピーダンスにする
スリープモード	0x	CPU スリープモード(CPU のみ停止)
	1x	バススリープモード(CPU、オンチップバス停止) *

*: DMA 転送中のみはオンチップバスが動作します。
ハイインピーダンスされる端子については『付録』を参照してください。

27.4.2 PMU コントロールレジスタ ：PMUCTLR (Power Management Unit ConTroL register)

PMU コントロールレジスタのビット構成について説明します。

PMU を制御するレジスタです。

■ PMUCTLR： アドレス 0591H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SHDE	予約	IOCTMD	IOCT	予約			
初期値	0	0	0	0	0	0	0	0
属性	R/W	R0,W0	R/W	R/W	R0,W0	R0,W0	R0,W0	R0,W0

本レジスタは、パワーオンリセット、内部低電圧リセット、RSTX/NMIX 同時アサートによるリセットおよびハードウェアウォッチドッグタイマリセットで初期化されます。

[bit7] SHDE (SHut Down Enable)
CPU のモードがスタンバイ(時計/STOP)に遷移した際にシャットダウンモードにするかどうかの設定です。

SHDE	SHDE モード許可
0	スタンバイ遷移時に ShutDown 処理を行わない。
1	スタンバイ遷移時に ShutDown 処理を行う。

[bit6] 予約
常に"0" が読み出されます。必ず"0" を書き込んでください。

[bit5] IOCTMD (I/O Clear Timing MoDe)
スタンバイ(ShutDown)モードからの復帰時に I/O の状態保持するタイミングを選択します。(ハード処理)

IOCTMD	I/O 保持解除要求モード
0	スタンバイ(時計、STOP)モードから復帰するまで、I/O 状態を保持します。
1	IOCT のレジスタをクリアするまで、I/O 状態を保持します。

[bit4] IOCT (I/O Clear Timing)
IOCTMD=1 時に ShutDown からの復帰時に I/O 状態保持を解除します。(ソフト処理)

IOCT	I/O 保持解除要求
0	要求なし
1	要求中

本ビットは、"1"を書込みし、I/O 保持解除要求で I/O 保持解除後、自動的に"0"にクリアされます。
また、I/O 保持中以外の書込みは、無効となります。
本ビットは、"0"書込みは無効です。

[bit3～bit0] 予約
常に"0" が読み出されます。必ず"0" を書き込んでください。

27.4.3 Power on Timing コントロールレジスタ ：PWRTMCTL (PoWeR on TiMing ConTroL register)

Power on Timing コントロールレジスタのビット構成について説明します。

パワーオンリセットなどのタイミングを制御するレジスタです。

■ PWRTMCTL ：アドレス 0592H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					PTC[2:0]		
初期値	0	0	0	0	0	0	1	1
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W

本レジスタは、パワーオンリセット、内部低電圧リセット、RSTX/NMIX 同時アサートによるリセットおよびハードウェアウォッチドッグタイマリセットで初期化されます。

[bit7～bit3] 予約
常に"0" が読み出されます。必ず"0" を書き込んでください。

[bit2～bit0] PTC[2:0] (Power on Timing Cycle setting)
PSW の立上げ時間を設定します。

PTC[2:0]	立上げ時間	備考(PMUCLK=32kHz の場合)
000	1×(1/PMUCLK)	30µs
001	3×(1/PMUCLK)	90µs
010	5×(1/PMUCLK)	150µs
011	9×(1/PMUCLK)	270µs
100	設定禁止	-
101	2×(1/PMUCLK)	60µs
110	4×(1/PMUCLK)	120µs
111	7×(1/PMUCLK)	210µs

27.4.4 PMU 割込みフラグレジスタ 0 : PMUINTF0 (Power Management Unit INTerrupt Flag0 register)

PMU 割込みフラグレジスタ 0 のビット構成について説明します。

Shutdown 時に外部入力により割込み要求があったことを示すレジスタです。

■ PMUINTF0 : アドレス 0594H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EIF15	EIF14	EIF13	EIF12	EIF11	EIF10	EIF9	EIF8
初期値	0	0	0	0	0	0	0	0
属性	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W

本レジスタは、パワーオンリセット、内部低電圧リセット、RSTX/NMIX 同時アサートによるリセットおよびハードウェアウォッチドッグタイマリセットで初期化されます。

[bit7～bit0] EIF15～EIF8 (External Interrupt Flag15-8)
ShutDown 時に外部入力により割込み要求があった事を示すフラグです。

EIFn	外部割込み要求
0	要求なし
1	要求あり

n : 15～8 の数字が入ります。

本ビットは、ShutDown 時のみ有効です。
本ビットは、"0"書込みでクリアします。"1"書込みは、無効です。

27.4.5 PMU 割込みフラグレジスタ 1 : PMUINTF1 (Power Management Unit INTerrupt Flag1 register)

PMU 割込みフラグレジスタ 1 のビット構成について説明します。

Shutdown 時に外部入力により割込み要求があったことを示すレジスタです。

■ PMUINTF1 : アドレス 0595H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EIF7	EIF6	EIF5	EIF4	EIF3	EIF2	EIF1	EIF0
初期値	0	0	0	0	0	0	0	0
属性	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W

本レジスタは、パワーオンリセット、内部低電圧リセット、RSTX/NMIX 同時アサートによるリセットおよびハードウェアウォッチドッグタイマリセットで初期化されます。

[bit7～bit0] EIF7～EIF0 (External Interrupt Flag7-0)
ShutDown 時に外部入力により割込み要求があった事を示すフラグです。

EIFn	外部割込み要求
0	要求なし
1	要求あり

n :7～0 の数字が入ります。

本ビットは、ShutDown 時のみ有効です。
本ビットは、"0"書込みでクリアします。"1"書込みは、無効です。

27.4.6 PMU 割込みフラグレジスタ 2 : PMUINTF2 (Power Management Unit INTerrupt Flag2 register)

PMU 割込みフラグレジスタ 2 のビット構成について説明します。

Shutdown 時に割込み要求があったことを示すレジスタです。

■ PMUINTF2 : アドレス 0596H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	RIF	NIF	MTIF	STIF	予約			
初期値	0	0	0	0	0	0	0	0
属性	R(RM1), W	R(RM1), W	R(RM1), W	R(RM1), W	R0,W0	R0,W0	R0,W0	R0,W0

本レジスタは、パワーオンリセット、内部低電圧リセット、RSTX/NMIX 同時アサートによるリセットおよびハードウェアウォッチドッグタイマリセットで初期化されます。

[bit7] RIF (Rtc Interrupt Flag)
ShutDown 時に RTC より割込み要求があった事を示すフラグです。

RIF	RTC 割込み要求
0	要求なし
1	要求あり

本ビットは、ShutDown 時のみ有効です。
本ビットは、"0"書込みでクリアします。"1"書込みは、無効です。

[bit6] NIF(Nmi Flag)
ShutDown 時に NMI より割り込み要求があった事を示すフラグです。

NIF	NMI 割り込み要求
0	要求なし
1	要求あり

本ビットは、ShutDown 時のみ有効です。
本ビットは、"0"書込みでクリアします。"1"書込みは、無効です。

[bit5] MTIF (Main Timer Interrupt Flag)
ShutDown 時に Main Timer より割込み要求があった事を示すフラグです。

MTIF	メインタイマ割込み要求
0	要求なし
1	要求あり

本ビットは、ShutDown 時のみ有効です。
本ビットは、"0"書込みでクリアします。"1"書込みは、無効です。

スタンバイモード(電源遮断)からの復帰中は、内部リセットが発行されており、メインタイマ割込みフラグはセットされません。

[bit4] STIF (Sub Timer Interrupt Flag)
ShutDown 時に Sub Timer より割込み要求があった事を示すフラグです。

STIF	サブタイマ割込み要求
0	要求なし
1	要求あり

本ビットは、ShutDown 時のみ有効です。
本ビットは、"0"書込みでクリアします。"1"書込みは、無効です。

スタンバイモード(電源遮断)からの復帰中は、内部リセットが発行されており、サブタイマ割込みフラグはセットされません。

[bit3～bit0] 予約
常に"0" が読み出されます。必ず"0" を書き込んでください。

27.5 動作説明

本シリーズの消費電力制御機能について説明します。

27.5.1. クロック制御

27.5.2. 各低消費電力モードにおけるクロック供給一覧

27.5.3. スリープモード

27.5.4. スタンバイモード : 時計モード

27.5.5. スタンバイモード:時計モード (電源遮断)

27.5.6. スタンバイモード:ストップモード

27.5.7. スタンバイモード:ストップモード (電源遮断)

27.5.8. マイコン停止状態

27.5.9. 不正スタンバイモード移行

27.5.10. 電源遮断・通常スタンバイ制御の制限事項

27.5.1　クロック制御

消費電力制御のクロック制御について説明します。

本シリーズの各動作クロックを調整することで、消費電力と処理能力の最適化を行うことができます。

■ 分周設定

『クロック』の章を参照してください。

■ 未使用クロックの停止

以下にあげるクロックは、個別に停止させる設定があります。

□　外部バスクロック(TCLK)　：バススリープ中の供給/停止が選択可能

設定方法は『クロック』の章を参照してください。

27.5.2　各低消費電力モードにおけるクロック供給一覧

各低消費電力モードにおけるクロック供給一覧について示します。

表 27-2 各低消費電力モードにおけるクロック供給一覧

クロック	スタンバイ		スリープ	
	ストップ	時計	バス	CPU
CPU クロック(CCLK)	○	○	○	×
CAN プリスケーラクロック	○	○	*1	×
オンチップバスクロック(HCLK)	○	○	○	×
周辺クロック(PCLK)	○	○	×	×
外部バス I/F クロック(TCLK)	○	○	*2	×
PLL クロック(PLLCLK)	○	○	×	×
メインクロック(MCLK)	○	×	×	×
サブクロック(SBCLK)	○	×	×	×
CR 発振	○*4	○*4	×*3	×*3

○:　　停止します。
×:　　停止しません。
(メイン・サブ・PLL が各クロック設定レジスタで停止させているときはそれに従います。)

*1: CAN プリスケーラクロックにオンチップバスクロック(HCLK)が選択されている場合に停止します。PLL 出力が選択されている場合は、PLL 出力に従います。それ以外の場合は停止しません。
*2: DIVR1:TSTP ビットで設定します。『クロック』の章を参照してください。
*3: スリープモード中は、CR 発振は停止しませんが、ウォッチドッグタイマ 1 (HWWDT) は停止します。
*4: CR 発振をスタンバイ時に停止させるには、あらかじめ設定が必要です。『クロックスーパバイザ』の章の CSVCR:RCE ビットの記述を参照してください。

27.5.3 スリープモード

スリープモードについて説明します。

スリープモードは、CPU やオンチップバスを停止させ、ペリフェラルのみ動作させるモードです。スリープモードには、停止させる範囲の違いにより、以下のモードがあります。

- CPU スリープモード　　：CPU のみを停止します。
- バススリープモード　　：CPU とオンチップバスを停止します。

ウェイクアップ要求が発生するまで、停止状態を継続します。ウェイクアップ要求の発生により、数クロックでプログラム動作へ復帰できます。

以下、各モードの動作を説明します。

■ CPU スリープモード

CPU スリープモードは、CPU 動作を停止させるモードです。

本モードでは、DMA コントローラやオンチップバスは動作を継続できますが、その分バススリープモードより多くの電力を消費します。

■ バススリープモード

バススリープモードは、CPU およびオンチップバスを停止させるモードです。本モードでは、CPU クロック(CCLK)およびオンチップバスクロック(HCLK)は停止します。

バススリープモード中に DMA 転送要求を受け付けると、一時的にオンチップバスクロック(HCLK)の供給を再開し、DMA 転送を行います。DMA 転送が終了後、再度オンチップバスクロック(HCLK)を停止します。

本モードでは、CPU スリープモードより消費電力を低減することが可能ですが、DMA 転送要求に対するレスポンスが若干低下します。

■ スリープモードの設定

スリープモードを起動する前に、DIVR1 レジスタの bit7:TSTP へ設定する値により、スリープモード中の外部バスクロックを停止するか供給するかを選択します。

- DIVR1 レジスタの bit7:TSTP="0"時、外部バスクロックは停止しません。
- DIVR1 レジスタの bit7:TSTP="1"時、外部バスクロックは停止します。

スリープモードを起動する際、STBCR レジスタの bit1:SLVL1 へ設定する値により、スリープモードのレベルを選択します。

- STBCR レジスタの bit1:SLVL1= "0"時、CPU スリープモードとなります。
- STBCR レジスタの bit1:SLVL1= "1"時、バススリープモードとなります。

■ スリープモードの起動

スリープモードは、以下の手順で起動します。

- STBCR レジスタの bit7:STOP, bit6:TIMER, bit5:SLEEP に"001"をライトする
- STBCR をリードする

FR81S コアは、リード値を次の命令で使用しない場合、リードの終了を待たずに次の命令を実行しますので、スリープモードに入る前に命令が先に進まないように、次の命令ではリード値を使用するダミー処理を行ってください。

[例] スリープモード起動のサンプルプログラム

```
LDI     #value_of_sleep, R0   ; SLEEP ビット="1", SLVL 設定
LDI     #_STBCR, R12          ;
STB     R0, @R12              ; ライト
LDUB    @R12, R0              ; リード (スリープモードの起動)
MOV     R0, R0                ; パイプライン調整のためのダミー処理
NOP                           ; パイプライン調整のためのダミー処理
```

■ スリープモードからのウェイクアップ

スリープモードは、以下の条件により終了します。

- リセット
- 対応する ICR レジスタ値が"0x1F"以外の割込み要求の発生
- NMI 要求の発生
- ICE 接続中のツールブレークの発生

割込み要求によるウェイクアップの場合、CPU がこの割込み要求を受け付ける設定である必要はありません。割込み要求が受け付けられなかった場合、スリープモードを起動した次の命令より、プログラムを続行します。

バススリープモードでは、DMA 転送要求の発生によりオンチップバスクロック(HCLK)を一時的に復帰させ、DMA 転送を行います。DMA 転送の終了後、再度オンチップバスクロック(HCLK)を停止します。

■ スリープモードの効果

スリープモードにより、ペリフェラルや外部入力のイベント待ち状態での消費電力を大幅に低減することができます。周辺クロック(PCLK)は動作するため、時計モードやストップモードほどの消費電力低減はできません。反面、ウェイクアップ要求の発生により、数クロックでプログラム動作へ復帰することが可能です。

27.5.4　スタンバイモード：時計モード

スタンバイモード：時計モードについて説明します。

時計モードは、特定のクロック発振のみを継続し、そのクロックに対応したクロックタイマのみをカウントさせるモードです。クロックソースとしてサブクロック(SBCLK)を選択している場合、サブクロック発振のみ動作し、サブタイマのみカウントを行います。

＜注意事項＞
スタンバイモードに入るのは、メイン RUN またはサブ RUN 時のみからにしてください。PLL ランからスタンバイモードへの遷移を行った際の動作については、「27.5.9 不正スタンバイモード移行」を参照してください。

＜注意事項＞
FLASH プログラム/イレーズ中にスタンバイ状態に遷移することは禁止です。

■ 時計モードの設定

時計モードを起動する前に、STBCR レジスタの bit1:SLVL1 により、時計モード中の外部端子の状態を設定します。

- STBCR レジスタの bit1:SLVL1= "0"時、外部端子は直前状態の保持となります。
- STBCR レジスタの bit1:SLVL1= "1"時、外部端子はハイインピーダンスとなります。

状態制御される端子は、品種により異なります。『付録』を参照してください。

■ 時計モードの起動

時計モードは、以下の手順で起動します。

- PMUCTLR レジスタの bit7:SHDE に"0"をライトします。
- PLL RUN 時の場合、メイン RUN に遷移します。 (サブ RUN 時の場合はサブ RUN のままで時計モードに遷移できます。)
- STBCR レジスタの bit7:STOP, bit6:TIMER, bit5:SLEEP に"010"をライトします。
- STBCR レジスタをリードします。

FR81S コアは、リード値を次の命令で使用しない場合、リードの終了を待たずに次の命令を実行しますので、時計モードに入る前に命令が先に進まないように、次の命令ではリード値を使用するダミー処理を行ってください。

[例] 時計モード起動のサンプルプログラム

```
LDI     #value_of_timer, R0  ; TIMER ビット="1", SLVL 設定
LDI     #_STBCR, R12         ;
STB     R0, @R12             ; ライト
LDUB    @R12, R0             ; リード (時計モードの起動)
MOV     R0, R0               ; パイプライン調整のためのダミー処理
NOP                          ; パイプライン調整のためのダミー処理
```

■ 時計モードからのウェイクアップ

時計モードは、以下の条件により終了します。

- リセット
- 対応する ICR レジスタ値(『割込み制御(割込みコントローラ)』の章を参照)が"0x1F"以外の割込み要求の発生
- NMI 要求の発生
- ICE 接続中のツールブレークの発生

割込み要求によるウェイクアップの場合、CPU がこの割込み要求を受け付ける設定である必要はありません。割込み要求が受付けられなかった場合、時計モードを起動した次の命令より、プログラムを続行します。

■ 時計モードの効果

時計モードにより、外部入力のイベント待ち状態での消費電力を大幅に低減することができます。許可されたクロックの発振は動作するため、ストップモードほどの消費電力低減はできません。反面、クロックタイマを動作させることもでき、またウェイクアップ要求の発生により、ストップモードからの復帰に比べて短時間でプログラム動作へ復帰することが可能です。*

*: 止めていないクロックでプログラム動作を続行する場合

27.5.5 スタンバイモード:時計モード (電源遮断)

スタンバイモード:時計モード(電源遮断)について説明します。

時計モード(電源遮断)は、電源を遮断した上で特定のクロック発振のみを継続し、そのクロックに対応したクロックタイマのみをカウントさせるモードです。クロックソースとしてサブクロック(SBCLK)を選択している場合、サブクロック発振のみ動作し、サブタイマのみカウントを行います。

＜注意事項＞
スタンバイモードに入るのは、メイン RUN またはサブ RUN 時のみからにしてください。PLL ランからスタンバイモードへの遷移を行った際の動作については、「27.5.9 不正スタンバイモード移行」を参照してください。

＜注意事項＞
FLASH プログラム/イレーズ中にスタンバイ状態に遷移することは禁止です。

■ 時計モード (電源遮断) の設定

時計モード (電源遮断) を起動する前に、以下の設定/制御を行ってください。

(1) STBCR レジスタの bit1:SLVL1 により、時計モード(電源遮断)中の外部端子の状態を設定します。
- STBCR レジスタの bit1:SLVL1= "0"時、外部端子は直前状態の保持となります。
- STBCR レジスタの bit1:SLVL1= "1"時、外部端子はハイインピーダンスとなります。

状態制御される端子は、品種により異なります。『付録』を参照してください。

■ 時計モード (電源遮断) の起動

時計モード (電源遮断) は、以下の手順で起動します。

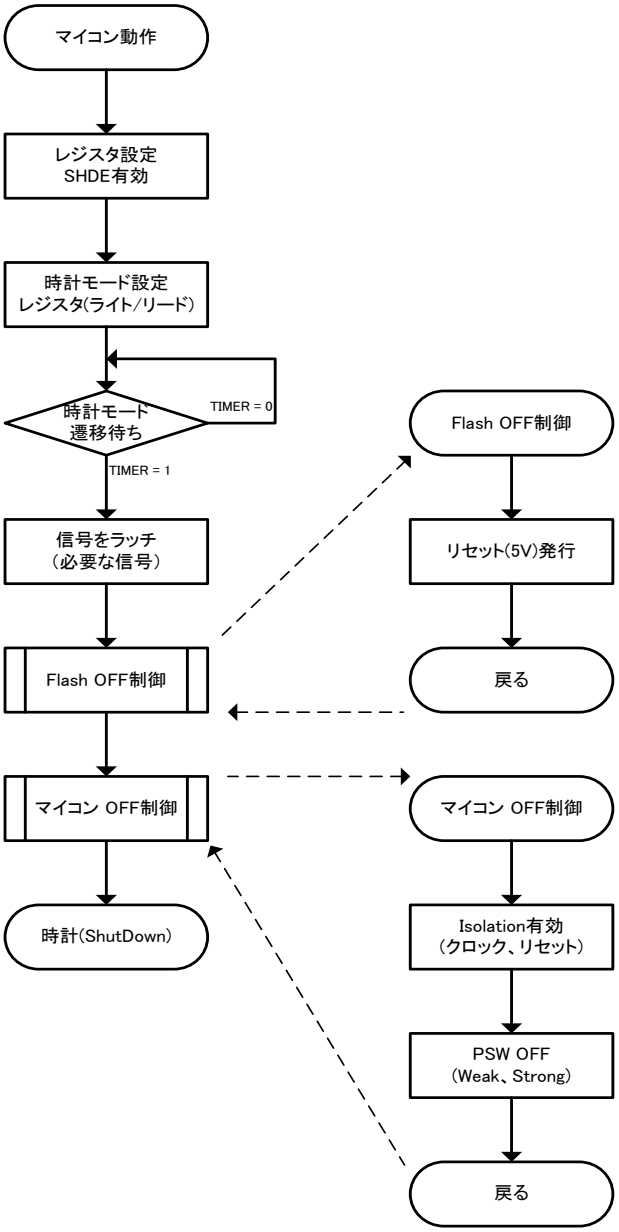
- PMUCTLR レジスタの bit7:SHDE に"1"をライトします。
- PLL RUN 時の場合、メイン RUN に遷移します。 (サブ RUN 時の場合はサブ RUN のままで時計モード(電源遮断)に遷移できます。
- STBCR レジスタの bit7:STOP, bit6:TIMER, bit5:SLEEP に"010"をライトします。
- STBCR レジスタをリードします。

FR81S コアは、リード値を次の命令で使用しない場合、リードの終了を待たずに次の命令を実行しますので、時計モード(電源遮断)に入る前に命令が先に進まないように、次の命令ではリードを使用するダミー処理を行ってください。

[例] 時計モード(電源遮断)起動のサンプルプログラム

```
LDI     #value_of_PMU, R0     ; SHDE ビット="1", IOCTMD/IOCT ビット設定
LDI     #_PMUCTLR, R12        ;
STB     R0, @R12              ; ライト
LDI     #value_of_timer, R0   ; TIMER ビット="1", SLVL 設定
LDI     #_STBCR, R12          ;
STB     R0, @R12              ; ライト
LDUB    @R12, R0              ; リード (時計モード(電源遮断)の起動)
MOV     R0, R0                ; パイプライン調整のためのダミー処理
NOP                           ; パイプライン調整のためのダミー処理
```

図 27-3 時計 (電源遮断) モードへの遷移シーケンス



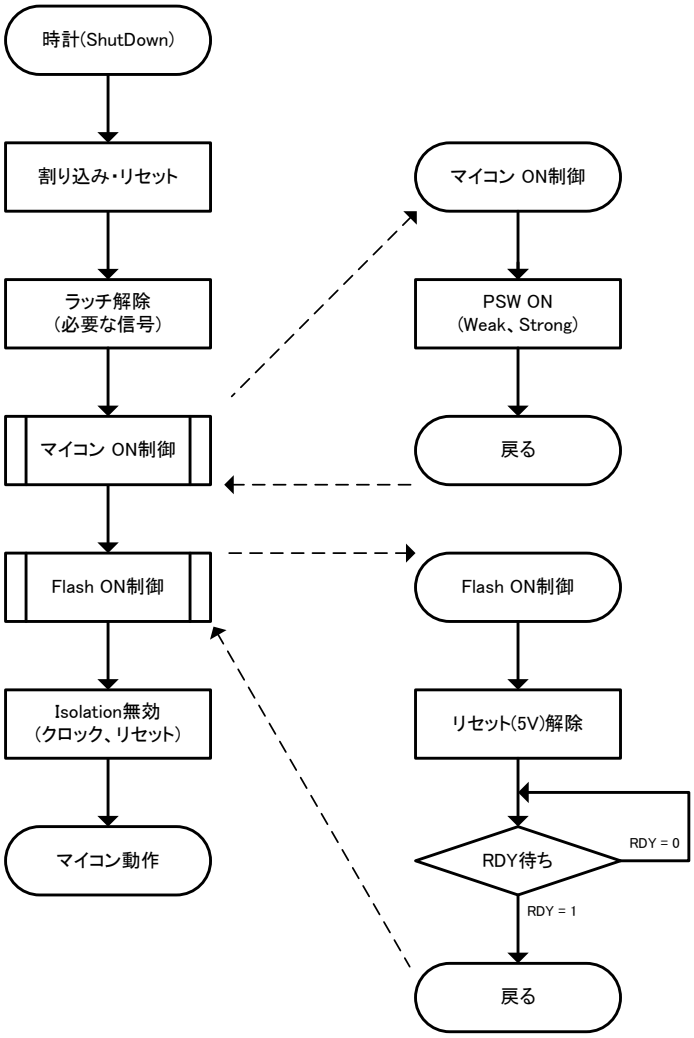
■ 時計モード (電源遮断) からのウェイクアップ

時計モード (電源遮断) は、以下の条件により終了します。

- リセット
- 外部割込み要求の発生
- NMI 要求の発生
- RTC 割込み要求の発生
- メイン/サブ・タイマ割込み要求の発生

割込み要求によるウェイクアップの場合、CPU および割込みコントローラがこの割込み要求を受け付ける設定である必要はありません。CPU は必ずリセット状態から動作を開始します。
RTC、LCD コントローラ、外部割込み入力(IOCTMD=1 時)のレジスタは初期化されません。
また、ウェイクアップ中はパワーオンリセット、内部低電圧リセット、RSTX/NMIX 同時アサートによるリセット以外のリセット要因を受付けません。このとき、RTC、LCD コントローラ、外部割込み入力(IOCTMD=1 時)のレジスタは初期化されませんので、立上げ後に RSTX 端子入力からのリセット入力または外部低電圧検出フラグがセットされている場合には、本レジスタを初期化してから使用してください。

図 27-4 時計 (電源遮断) モードからの復帰シーケンス



■ 時計モード (電源遮断) の効果

時計モード (電源遮断) により、不要な回路の待機電流を大幅に低減することができます。許可されたクロックの発振は動作するため、ストップモードほどの消費電力低減はできません。反面、クロックタイマを動作させることもでき、またウェイクアップ要求の発生により、クロック発振安定待ちをせずにプログラム動作へ復帰することが可能です。

27.5.6　スタンバイモード:ストップモード

スタンバイモード:ストップモードについて説明します。

ストップモードは、すべてのクロック発振を停止し、本シリーズの消費電力を最少にするモードです。

＜注意事項＞
スタンバイモードに入るのは、メイン RUN またはサブ RUN 時のみからにしてください。PLL ランからスタンバイモードへの遷移を行った際の動作については、「27.5.9 不正スタンバイモード移行」を参照してください。

＜注意事項＞
FLASH プログラム/イレーズ中にスタンバイ状態に遷移することは禁止です。

■ ストップモードの設定

ストップモードを起動する前に、STBCR レジスタの bit1:SLVL1 により、ストップモード中の外部端子の状態を設定します。

- STBCR レジスタの bit1:SLVL1= "0"時、外部端子は直前状態の保持となります。
- STBCR レジスタの bit1:SLVL1= "1"時、外部端子はハイインピーダンスとなります。

状態制御される端子は、品種により異なります。『付録』を参照してください。

■ ストップモードの起動

ストップモードは、以下の手順で起動します。

- PMUCTLR レジスタの bit7:SHDE に"0"をライトします。
- PLL RUN 時の場合、メイン RUN に遷移します。 (サブ RUN 時の場合はサブ RUN のままでストップモードに遷移できます。
- STBCR レジスタの bit7:STOP, bit6:TIMER, bit5:SLEEP に"100"をライトします。
- STBCR レジスタをリードします。

FR81S コアは、リード値を次の命令で使用しない場合、リードの終了を待たずに次の命令を実行しますので、ストップモードに入る前に、命令が先に進まないように、次の命令ではリード値を使用するダミー処理を行ってください。

[例]ストップモード起動のサンプルプログラム

```
LDI     #value_of_stop, R0    ; STOP ビット="1", SLVL 設定
LDI     #_STBCR, R12          ;
STB     R0, @R12              ; ライト
LDUB    @R12, R0              ; リード (ストップモードの起動)
MOV     R0, R0                ; パイプライン調整のためのダミー処理
NOP                           ; パイプライン調整のためのダミー処理
```

■ ストップモードからのウェイクアップ

ストップモードは、以下の条件により終了します。

- リセット
- 対応する ICR レジスタ値(『割込み制御(割込みコントローラ)』の章を参照)が"0x1F"以外の割込み要求の発生
- NMI 要求の発生
- ICE 接続中のツールブレークの発生

割込み要求によるウェイクアップの場合、CPU がこの割込み要求を受け付ける設定である必要はありません。割込み要求が受け付けられなかった場合、ストップモードを起動した次の命令より、プログラムを続行します。

■ ストップモードの効果

ストップモードにより、外部入力のイベント待ち状態での消費電力を最小化することができます。反面、ウェイクアップ要求の発生後、プログラム動作へ復帰する迄には、発振安定待ち時間を必要とします。

27.5.7　スタンバイモード:ストップモード (電源遮断)

スタンバイモード:ストップモード(電源遮断)について説明します。

ストップモード(電源遮断)は、すべてのクロック発振を停止し、本シリーズの消費電力を最少にするモードです。

＜注意事項＞
スタンバイモードに入るのは、メイン RUN またはサブ RUN 時のみからにしてください。PLL ランからスタンバイモードへの遷移を行った際の動作については、「27.5.9 不正スタンバイモード移行」を参照してください。

＜注意事項＞
FLASH プログラム/イレーズ中にスタンバイ状態に遷移することは禁止です。

■ ストップモード (電源遮断) の設定

ストップモード(電源遮断)を起動する前に、以下の設定/制御を行ってください。

(1) STBCR レジスタの bit1:SLVL1 により、ストップモード(電源遮断)中の外部端子の状態を設定します。
- □　STBCR レジスタの bit1:SLVL1= "0"時、外部端子は直前状態の保持となります。
- □　STBCR レジスタの bit1:SLVL1= "1"時、外部端子はハイインピーダンスとなります。

状態制御される端子は、品種により異なります。『付録』を参照してください。

■ ストップモード (電源遮断) の起動

ストップモード(電源遮断)は、以下の手順で起動します。

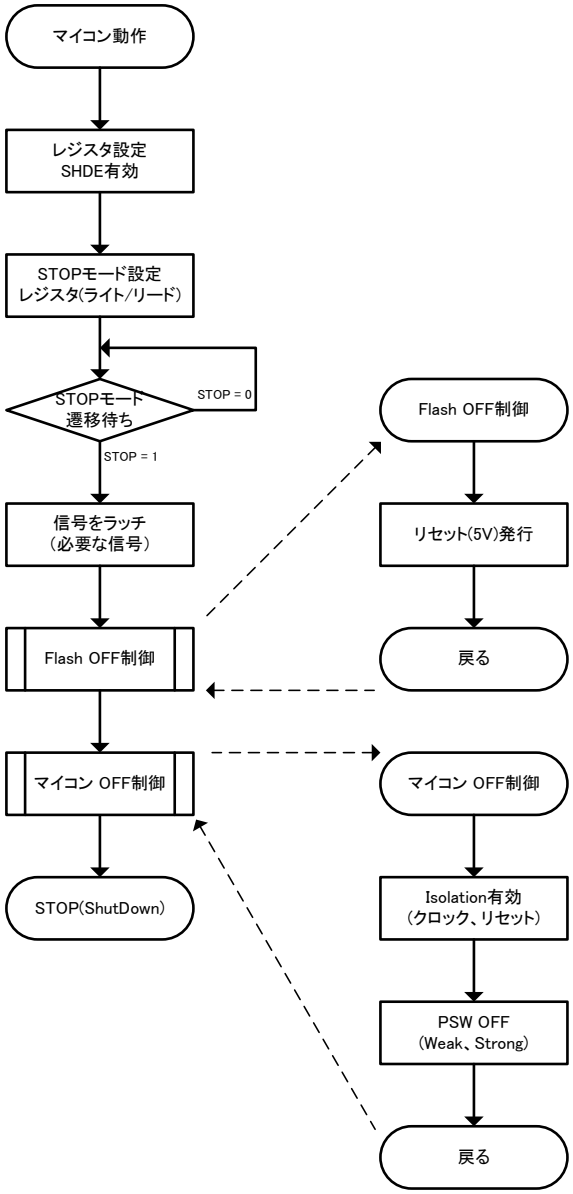
- PMUCTLR レジスタの bit7:SHDE に"1"をライトします。
- PLL RUN 時の場合、メイン RUN に遷移します。(サブ RUN 時の場合はサブ RUN のままでストップモード(電源遮断)に遷移できます。)
- STBCR レジスタの bit7:STOP, bit6:TIMER, bit5:SLEEP に"100"をライトします。
- STBCR レジスタをリードします。

FR81S コアは、リード値を次の命令で使用しない場合、リードの終了を待たずに次の命令を実行しますので、ストップモード(電源遮断)に入る前に、命令が先に進まないように、次の命令ではリード値を使用するダミー処理を行ってください。

[例]ストップモード(電源遮断)起動のサンプルプログラム

```
LDI     #value_of_PMU, R0     ; SHDE ビット="1", IOCTMD/IOCT ビット設定
LDI     #_PMUCTLR, R12        ;
STB     R0, @R12              ; ライト
LDI     #value_of_stop, R0    ; STOP ビット="1", SLVL 設定
LDI     #_STBCR, R12          ;
STB     R0, @R12              ; ライト
LDUB    @R12, R0              ; リード (ストップモード(電源遮断)の起動)
MOV     R0, R0                ; パイプライン調整のためのダミー処理
NOP                           ; パイプライン調整のためのダミー処理
```

図 27-5 ストップ (電源遮断) モードへの遷移シーケンス



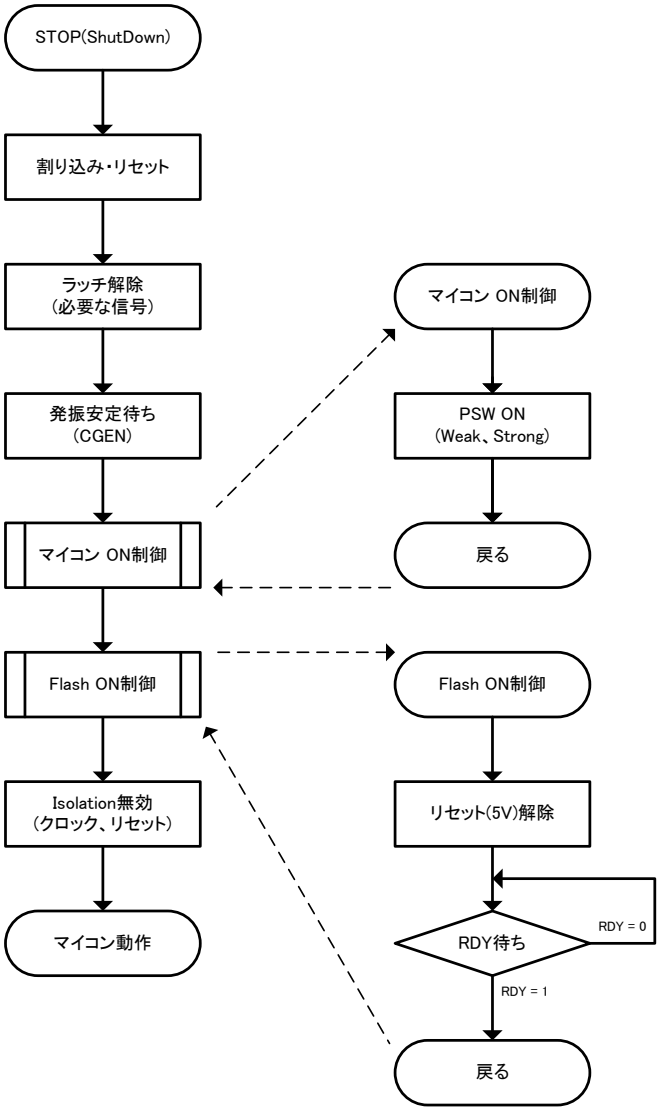
■ ストップモード (電源遮断) からのウェイクアップ

ストップモード (電源遮断) は、以下の条件により終了します。

- リセット
- 外部割込み要求の発生
- NMI 要求の発生

割込み要求によるウェイクアップの場合、CPU および割込みコントローラがこの割込み要求を受け付ける設定である必要はありません。CPU は必ずリセット状態から動作を開始します。
外部割込み入力(IOCTMD=1 時)のレジスタは初期化されません。
また、ウェイクアップ中はパワーオンリセット、内部低電圧リセット、RSTX/NMIX 同時アサートによるリセット以外のリセット要因を受付けません。このとき、外部割込み入力(IOCTMD=1 時)のレジスタは初期化されませんので、立上げ後に RSTX 端子入力からのリセット入力または外部低電圧検出フラグがセットされている場合には、本レジスタを初期化してから使用してください。

図 27-6 ストップ(電源遮断)モードからの復帰シーケンス



■ ストップモード(電源遮断)の効果

ストップモード(電源遮断)により、不要な回路の待機電流を最小化することができます。　　反面、ウェイクアップ要求の発生後、プログラム動作へ復帰する迄には、発振安定待ち時間を必要とします。

27.5.8　マイコン停止状態

マイコン停止状態について示します。

スタンバイモード(時計モード・時計モード(電源遮断)・ストップモード・ストップモード(電源遮断))移行禁止状態からスタンバイへの移行を制御しようとしたとき、スタンバイ移行が完結しません。

<スタンバイ移行禁止状態>
① OCD 接続中
② PLL 動作中

<マイコン停止状態で行われないスタンバイ制御>
① フラッシュメモリ省電力制御
② 発振停止(ストップモード・ストップモード(電源遮断)のとき)

ただし、PLL 動作中にスタンバイモード移行制御を行ったときには、不正スタンバイモード移行を検出して、発振停止動作を行います。不正スタンバイモード移行については、「27.5.9 不正スタンバイモード移行」を参照してください。

27.5.9　不正スタンバイモード移行

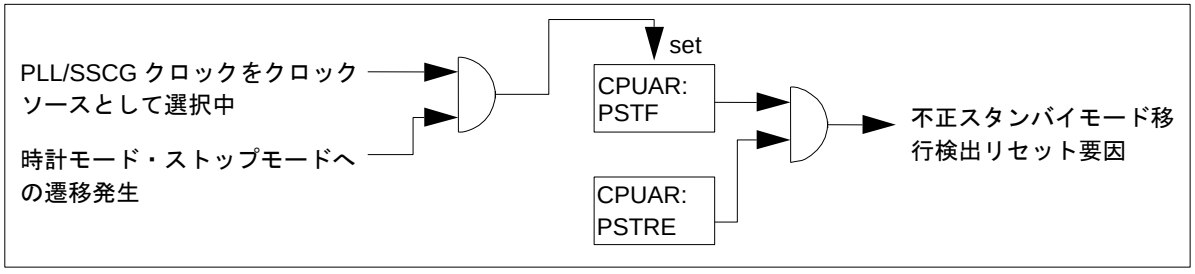
不正スタンバイモード移行について示します。

PLL ラン状態からスタンバイモード(時計モード・時計モード(電源遮断)・ストップモード・ストップモード(電源遮断))への遷移を行うと、スタンバイモードに設定され、PLL 発振安定は解除されます。(不正スタンバイモード移行)

スタンバイモードからの復帰後は CSELR:CKS[1:0]=00 および CMONR:CKM[1:0]=00(メインクロックの 2 分周)になります。

また、スタンバイモード移行と同時に CPUAR レジスタの PSTF フラグがセットされます。CPUAR レジスタの PSTRE ビットがセットされている場合は、不正スタンバイモード移行検出リセット要因によるリセットが発生します。CPUAR レジスタについては『リセット』の章の『4.3 CPU 異常動作レジスタ: CPUAR (CPU Abnormal operation Register)』を参照してください。

図 27-7 不正スタンバイモード移行検出リセット要因の生成図



27.5.10 電源遮断・通常スタンバイ制御の制限事項

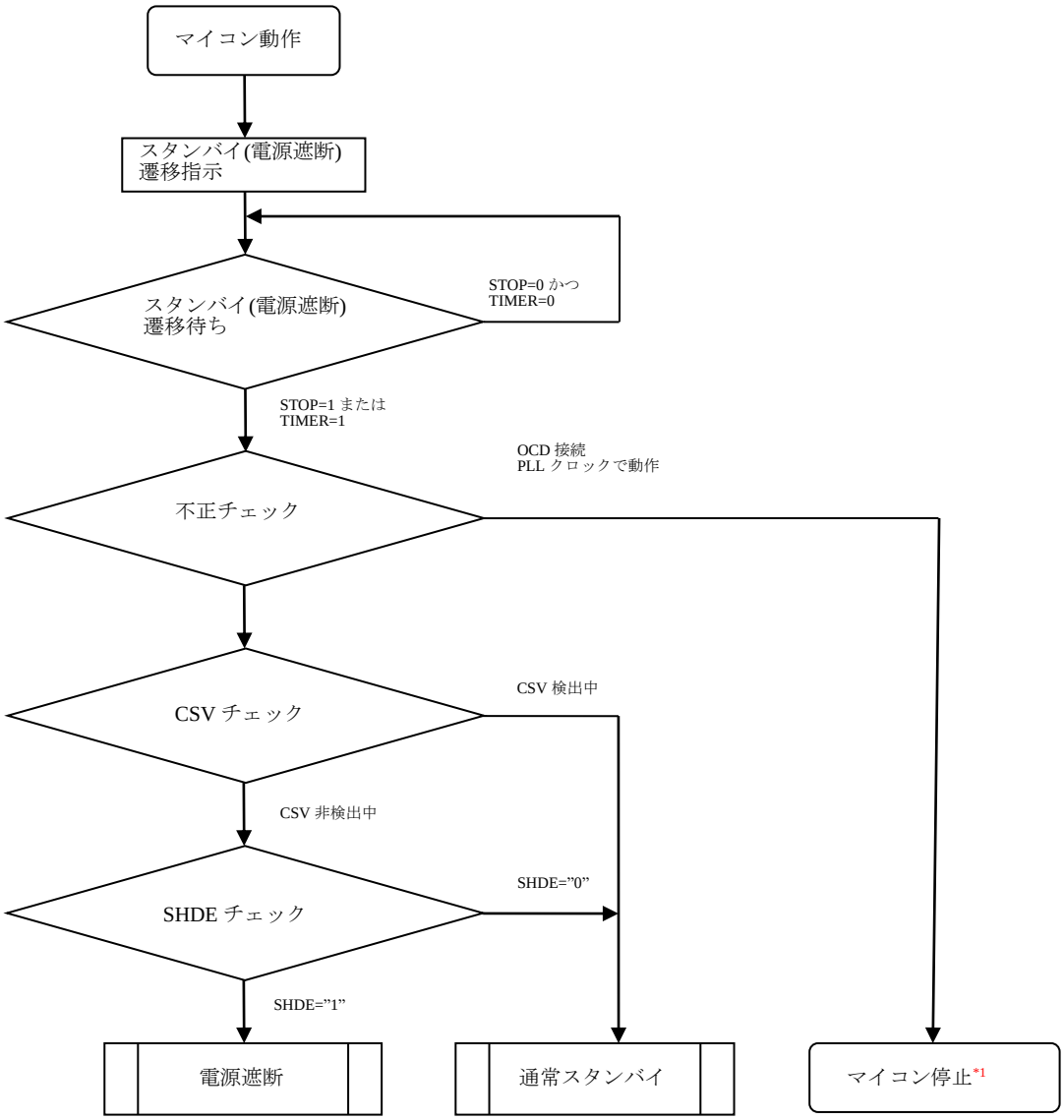
電源遮断・通常スタンバイ制御の制限事項について説明します。

本マイコンでは、以下の条件ではスタンバイ制御を行いません。

- CPU が PLL で動作している場合
- OCD 動作許可中
- CSV 機能でクロック断検出している場合*2, *3

以上の状態のときのスタンバイ制御は動作しませんが、CPU はスタンバイ状態となります。

図 27-8 電源遮断・通常スタンバイ制御の制限



*1: 電源遮断と認識せず、CPU はスタンバイモードに遷移する状態

*2: 動作しているクロックソースが断検出している場合です。例えばメインクロック動作中の断検出し、RC クロックで動作中などがあります。メインクロック動作中にサブクロックの断検出している場合は、制限事項に

はあたりません。

*3: クロック断検出した後にスタンバイ（電源遮断）遷移指示をした場合は、通常スタンバイ処理となります。また断検出していない状態で、電源遮断許可を有効にすると CSV の機能が停止しますので、注意が必要です。

スタンバイ(電源遮断)モードでは、チップ内部の大部分のブロックに電源が供給されないため、復帰時に一部のレジスタ以外は保持しません。表 27-3 にスタンバイ(電源遮断)からの復帰時に保持されるレジスタの一覧を示します。

表 27-3 スタンバイモード(電源遮断)からの復帰時に保持されるレジスタ

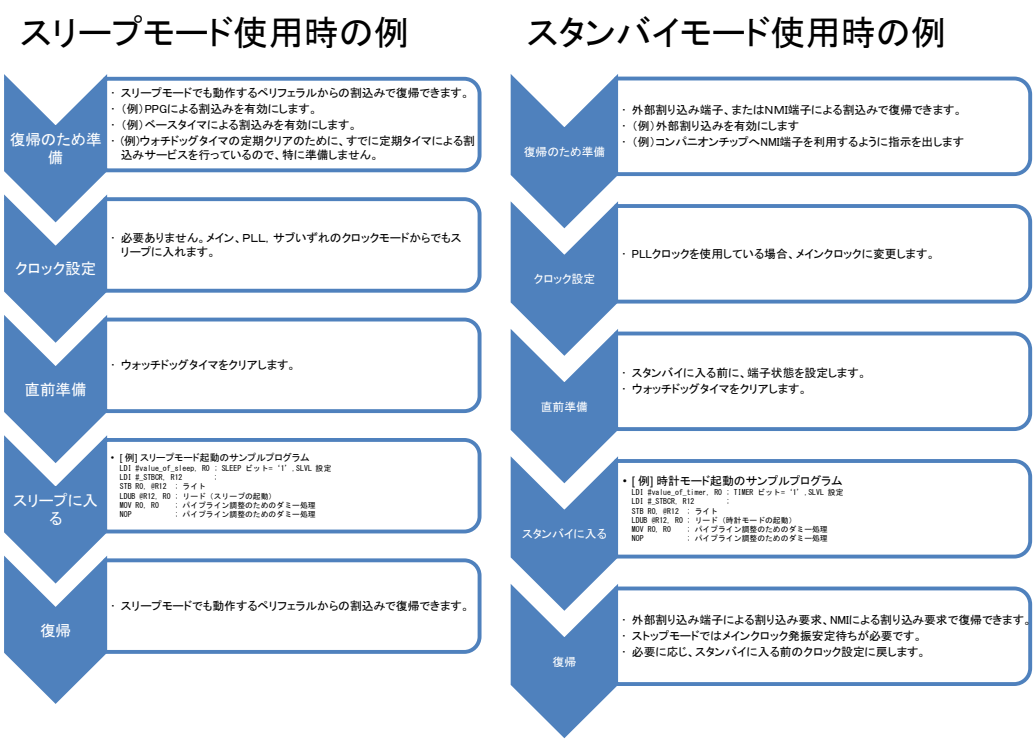
レジスタグループ	レジスタ・フラグ名称	種別	アドレス	備考
PMU レジスタ	PMUSTR.PMUST	フラグ	0590H bit7	
	PMUSTR.PONR_F	フラグ	0590H bit1	
	PMUSTR.RSTX_F	フラグ	0590 H bit0	
	PMUCTLR	レジスタ	0591 H	
	PWRTMCTL	レジスタ	0592 H	
	PMUINTF0	フラグ	0594 H	
	PMUINTF1	フラグ	0595 H	
	PMUINTF2	フラグ	0596 H	
リセット要因レジスタ	CPUAR.PMDF	フラグ	051A H bit2	
	CPUAR.PSTF	フラグ	051A H bit1	
	CPUAR.HWDF	フラグ	051A H bit0	
	LVD5R.LVD5R_F	フラグ	0584 H bit0	
	LVD5F.LVD5F_F	フラグ	0585 H bit0	
	LVD.LVD_F	フラグ	0586 H bit0	
低電圧検出レジスタ	LVD5F.LVD5F_PD	レジスタ	0585 H bit7	
	LVD5F.LVD5F_OE	レジスタ	0585 H bit3	
	LVD.LVD_PD	レジスタ	0586 H bit7	
	LVD.LVD_OE	レジスタ	0586 H bit3	
CSV レジスタ	CSVCR	レジスタ	056D H	
LCD コントローラレジスタ	すべて		05A8H-05BFH	*1
外部割込みレジスタ	EIRR0/1	レジスタ	0550 H /0554 H	*3
	ENIR0/1	レジスタ	0551 H /0555 H	*3
	ELVR0/1	レジスタ	0552 H /0556 H	*3
RTC レジスタ	WTDR	レジスタ	055E H -055FH	
	WTCR	レジスタ	0561 H -0563 H	*1
	WTBR	レジスタ	0565 H -0567 H	
	WTHR	レジスタ	0568 H	
	WTMR	レジスタ	0569 H	
	WTSR	レジスタ	056A H	
クロック選択レジスタ	CSELR.SCEN	レジスタ	0510 H bit7	*1,*2
	CMONR.SCRDY	フラグ	0511 H bit7	*1,*2
	CCRTSELR.CST	フラグ	0530 H bit7	*1,*2
	CCRTSELR.CSC	レジスタ	0530 H bit0	*1,*2

*1 ストップモード(電源遮断)からの復帰時は初期化されます。
*2 クロック 2 系統品種のみ対象です。
*3 PMUCTLR:IOCTMD = 0 のときは初期化されます。

27.6 使用例

スリープモード、スタンバイモード起動例を示します。

図 27-9 スリープモード、スタンバイモード起動例



28. 低電圧検出 (内部低電圧検出)

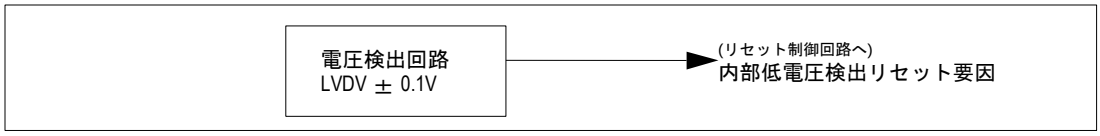
低電圧検出 (内部低電圧検出) について説明します。

28.1 概要

低電圧検出(内部低電圧検出)の概要について説明します。

内部低電圧検出は、内部電圧を監視し、検出電圧値より内部電圧が下がったことを検出する機能です。低電圧を検出した場合に、検出フラグをセットします。内部低電圧検出が検出フラグをセットすると、低電圧検出リセットによりリセット状態となります。

図 28-1 低電圧検出 (内部低電圧検出) ブロックダイヤグラム (概要)



28.2 特長

低電圧検出 (内部低電圧検出) の特長について説明します。

内部低電圧検出回路

形式　　：電圧 LVDV±0.1V 以下の電圧検出による設定初期化リセット発生

LVDV：0.9V

個数　　：1 個

動作　　：スリープ，ストップ，時計モード時も動作を続けます。

電圧比較回路　：検出電圧と内部電圧を比較して低電圧を検出すると出力を"H"から"L"にします。

電源投入後は、常に動作しています。

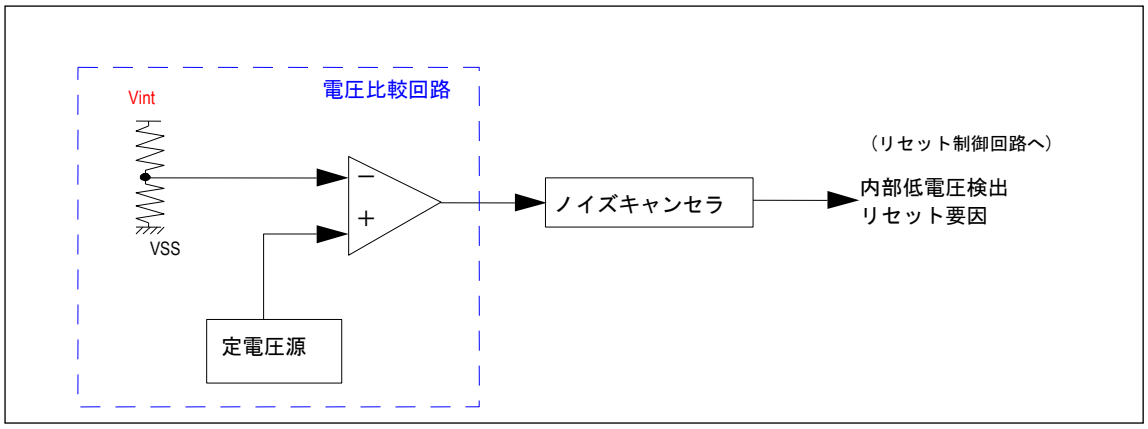
28.3 構成

低電圧検出 (内部低電圧検出) の構成を示します。

■ 低電圧検出 (内部低電圧検出) の構成図

構成図を、図 28-2 に示します。

図 28-2 構成図



28.4 レジスタ

低電圧検出 (内部低電圧検出) のレジスタについて説明します。

表 28-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0584	LVD5R	LVD5F	LVD	予約	内部低電圧検出 レジスタ

28.4.1 内部低電圧検出レジスタ ：LVD (Low Voltage Detect internal power fall register)

内部低電圧検出レジスタのビット構成について説明します。

内部低電圧検出フラグ (LVD_F) および制御ビットをもつレジスタです。

■ LVD： アドレス 0586H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	LVD_PD	LVD_SEL[2:0]			LVD_OE	予約		LVD_F
初期値	0	1	0	0	0	0	0	0
属性	R/W	R/W1	R/W0	R/W0	R/W	R0,WX	R0,WX	R(RM1), W

[bit7] LVD_PD (Low Voltage Detect fall Power Down)
内部電圧立下り検出をするかしないかの設定です。

LVD_PD	内部電圧立下りパワーダウン設定
0	無効 (検出を実行)
1	有効 (検出を停止)

本ビットはパワーオンリセットでのみ初期化されます。

＜注意事項＞
本ビットの設定によるパワーダウン有効→無効(動作開始)時には、100µs 後に検出許可(OE=0)にしてください。
100µs より前に検出許可にした場合、検出フラグがセットされる場合があります。

[bit6～bit4] LVD_SEL[2:0] (Low Voltage Detect internal power fall SELect)
内部電圧立下り検出の検出レベルの選択信号です。

LVD_SEL[2:0]	内部電圧立下り検出電圧設定
100	0.9V±0.1V
上記以外	設定禁止

LVD_OE="1"時のみ書換えが可能です。

[bit3] LVD_OE (Low Voltage Detect internal power fall Output Enable)
内部電圧立下り検出の出力許可信号です。

LVD_OE	内部電圧立下り検出出力許可設定
0	許可
1	停止

本ビットはパワーオンリセットでのみ初期化されます。

[bit2, bit1] 予約

[bit0] LVD_F (Low Voltage Detect internal power fall Flag) :内部低電圧検出フラグ
内部電圧立下り検出フラグです。

LVD_F	内部電圧立下り検出フラグ	
	読出し	書込み
0	検出してない	フラグをクリア
1	検出した	動作に影響なし

マイコン部の内部電圧の低下が検出されると、LVD_F ビットが"1"にセットされます。
外部リセット入力時のみ初期化されます。

28.5 動作説明

低電圧検出(内部低電圧検出)の動作について説明します。

28.5.1. 内部低電圧検出

28.5.1 内部低電圧検出

内部低電圧検出について説明します。

内部低電圧検出は、内部電圧を監視し、検出電圧値よりも内部電圧が低下したことを検出し、検出フラグをセットします。低電圧を検出してフラグがセットされた場合、設定初期化リセットが発生します。内部電圧が、検出電圧より低下した場合、内部低電圧検出電圧が復帰した後に発振安定待ち時間をとります。詳細は『リセット』の章を参照してください。

発振安定待ち時間	2^{15}×メインクロック周期

28.6 注意事項

低電圧検出(内部低電圧検出)の注意事項について説明します。

● 内部低電圧検出の動作

内部電圧が低下して内部低電圧検出フラグがセットされたとき(LVD:LVD_F="1")は、低電圧検出リセット機能により内部リセットが発生しています。したがって、内部低電圧検出レジスタ(LVD)の書込み/読出しはできません。

内部低電圧検出回路はスリープ、ストップモード、時計モード動作時においても、動作可能であり、動作している場合には電流を消費します。

内部低電圧検出回路はユーザの設定によって動作/停止が可能です。

● 内部低電圧検出フラグ (LVD:LVD_F) の初期値

内部低電圧検出フラグは、パワーオン直後に"1"にセットされます。内部低電圧検出フラグは、外部リセット、または内部低電圧検出レジスタ(LVD)の LVD_F への"0"書込みでクリアされます。

● 発振安定待ち時間

内部電圧が、検出電圧より低下した場合、内部電圧が復帰した後に発振安定待ち時間をとります。詳細は『リセット』の章を参照してください。

● 検出/解除のヒステリシス

検出/解除は 0.05V のヒステリシスを持つため、解除電圧は設定値+0.05V となります。例えば LVD : 1.0V±0.1V 設定のとき、解除電圧は 1.05V±0.1V となります。

29. 低電圧検出 (外部低電圧検出)

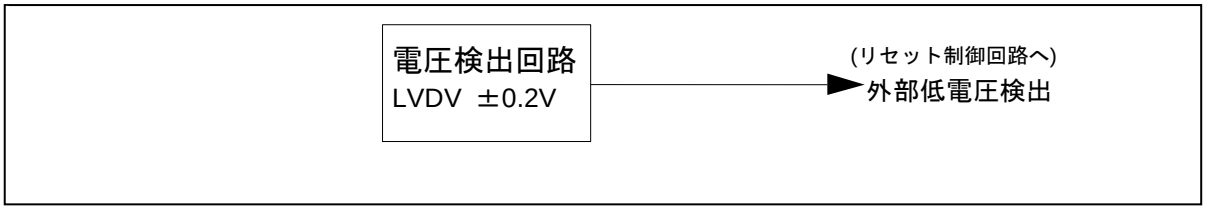
低電圧検出 (外部低電圧検出) について説明します。

29.1 概要

低電圧検出 (外部低電圧検出) の概要について説明します。

外部低電圧検出は、外部電圧を監視し、検出電圧値より電圧が下がったことを検出する機能があります。

図 29-1 ブロックダイヤグラム



(注意事項) 立上り LVDV：2.3V
立下り LVDV：3.7~4.3V (0.2V 単位)可変

29.2 特長

低電圧検出 (外部低電圧検出) の特長について説明します。

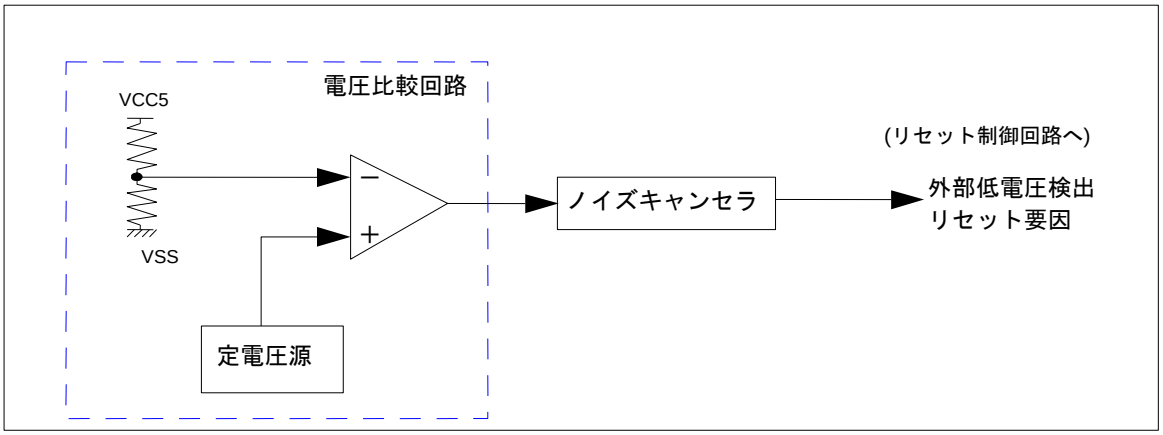
● 外部低電圧検出回路

- 形式 : 電圧 LVDV±0.2V 以下の電圧検出による設定初期化リセット発生
  (立上り LVDV : 2.3V 固定、立下り LVDV : 3.7～4.3V、0.2V 単位可変)
- 個数 : 1 個
- 動作 : ユーザ設定により、動作/停止を切り替えます。
  内部 RAM 書込み期間は、書込み終了後に低電圧リセットが発生します。
- 電圧比較回路 : 検出電圧とマイコン部外部電圧を比較して低電圧を検出すると出力を"L"にします。

29.3 構成

低電圧検出 (外部低電圧検出) の構成について説明します。

図 29-2 構成図



29.4 レジスタ

低電圧検出 (外部低電圧検出) のレジスタについて説明します。

表 29-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0584	LVD5R	LVD5F	LVD	予約	外部低電圧検出 立上り検出レジスタ 外部低電圧検出 立下り検出レジスタ

29.4.1 外部低電圧検出立上り検出レジスタ ：LVD5R (Low Voltage Detect external 5v Rise register)

外部低電圧検出立上り検出レジスタ (LVD5R)について説明します。

マイコン側の外部電源電圧立上り検出フラグです。

■ LVD5R：アドレス 0584H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							LVD5R_F
初期値	0	0	0	0	0	0	0	1
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R(RM1), W

[bit7～bit1] 予約

[bit0] LVD5R_F (Low Voltage Detect external 5v Rise Flag) :外部電圧立上り検出フラグ
外部電圧立上り検出フラグです。

LVD5R_F	外部電圧立上り検出フラグ	
	読出し	書込み
0	検出してない	フラグをクリア
1	検出した	動作に影響なし

外部電圧の立上りが検出されると、LVD5R_F ビットが"1"にセットされます。

外部リセット入力時にクリアされます。

29.4.2 外部低電圧検出立下り検出レジスタ ：LVD5F (Low Voltage Detect external 5v Fall register)

外部低電圧立下り検出レジスタ (LVD5F) について説明します。

低電圧検出リセットフラグクリアと低電圧検出回路の設定を行うレジスタです。

■ LVD5F：アドレス 0585H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	LVD5F_PD	予約	LVD5F_SEL[1:0]		LVD5F_OE	予約		LVD5F_F
初期値	0	0	1	0	0	0	0	1
属性	R/W	R0,WX	R/W	R/W	R/W	R0,WX	R0,WX	R(RM1), W

[bit7] LVD5F_PD (Low Voltage Detect external 5v Fall Power Down) :外部電圧立下りパワーダウン設定
外部電圧立下り検出をするかしないかの設定です。

LVD5F_PD	外部電圧立下りパワーダウン設定
0	無効(検出を実行)
1	有効(検出を停止)

本ビットはパワーオンリセットでのみ初期化されます。

＜注意事項＞
本ビットの設定によるパワーダウン有効⇒無効(動作開始)時には、100µs 後に検出許可(OE=0)にしてください。
100µs より前に検出許可にした場合、検出フラグがセットされる場合があります。

[bit6] 予約

[bit5, bit4] LVD5F_SEL[1:0] (Low Voltage Detect external 5v Fall SELect) :外部電圧立下り検出電圧設定
外部電圧立下り検出の検出レベルの選択信号です。

LVD5F_SEL[1:0]	外部立下り検出電圧設定
00	3.7V±0.2V
01	3.9V±0.2V
10	4.1V±0.2V
11	4.3V±0.2V

LVD5F_OE="1"時のみ書換えが可能です。

[bit3] LVD5F_OE (Low Voltage Detect external 5v Fall Output Enable) :外部電圧立下り検出出力許可設定
外部電圧立下り検出の出力許可信号です。

LVD5F_OE	外部電圧立下り検出出力許可設定
0	許可
1	停止

本ビットはパワーオンリセットでのみ初期化されます。

[bit2, bit1] 予約

[bit0] LVD5F_F(Low Voltage Detect external 5v Fall Flag) :外部電圧立下り検出フラグ
外部電圧立下り検出フラグです。

LVD5F_F	外部電圧立下り検出フラグ	
	読出し	書込み
0	検出してない	フラグをクリア
1	検出した	動作に影響なし

外部電圧の低下が検出されると、LVD5F_F ビットが"1"にセットされます。

外部リセット入力時にクリアされます。

29.5 動作説明

低電圧検出 (外部低電圧検出)について説明します。

29.5.1. 外部低電圧検出

29.5.1 外部低電圧検出

外部低電圧検出について説明します。

外部低電圧検出は、外部電圧を監視し、設定値よりも外部電圧が低下したときに初期化リセットを発生します。低電圧を検出して設定初期化リセットが発生した場合、レジスタの内容は保証できません。低電圧リセット解除後は発振安定待ち時間を取らずに、リセットシーケンスを実行した後にリセットベクタで指定されたアドレスからプログラムが再スタートします。

29.6 注意事項

低電圧検出 (外部低電圧検出) の注意事項について説明します。

■ 低電圧検出リセット回路使用上の注意

● プログラムでの動作

- 低電圧検出リセット回路は、外部低電圧検出立上り検出を除き、設定にしたがって、動作します。外部低電圧検出立上り検出はパワーオン・リセットとして使用されます。
- 外部低電圧検出立上り検出は常に動作していますので、スリープモード、ストップモード、時計モード動作時においても、電流を消費します。

● ストップモード時の動作

- 低電圧検出リセットは、設定によってストップモードでも動作を続けることができます。このとき、ストップモード中に低電圧を検出すると、設定初期化リセットが発生し、ストップモードは解除されます。

● 検出/解除のヒステリシス

- 検出/解除は 0.125V のヒステリシスを持つため、検出/解除電圧は設定値±0.125V となります。立下り検出電圧の場合、設定値は検出電圧を示します。例えば 4.1V±0.2V 設定のとき、解除電圧は 4.225V±0.2V となります。立上り検出電圧の場合、設定値は解除電圧を示します。例えば 2.5V±0.2V 設定のとき、検出電圧は 2.375V±0.2V となります。

30. ワイルドレジスタ

ワイルドレジスタについて説明します。

30.1 概要

ワイルドレジスタの概要について説明します。

ワイルドレジスタ機能は、アドレスレジスタに設定したパッチ対象アドレスのデータをデータレジスタに設定したデータとの置換えを行います。

30.2 特長

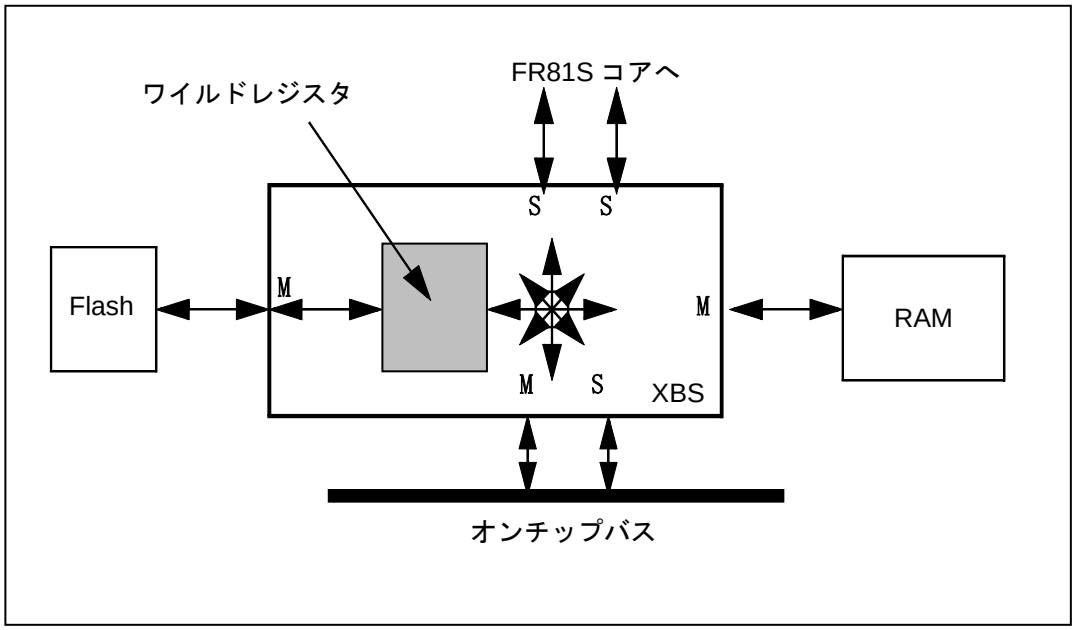
ワイルドレジスタの特長について説明します。

- 1 ワードごとに 16 箇所にパッチを当てることができます。
- 対象は Flash 領域のみです。
- 16 ビット制御レジスタ 1 本
- 32 ビットアドレス設定レジスタ 16 本
- 32 ビットデータ設定レジスタ 16 本

30.3 構成

ワイルドレジスタの構成について示します。

図 30-1 構成図



＜注意事項＞

本機能は、FLASH メモリへのアクセスウェイトを 1 サイクルに設定すると、使用できません。

30.4 レジスタ

ワイルドレジスタのレジスタについて説明します。

表 30-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0858	予約		WREN		ワイルドレジスタデータイネーブルレジスタ
0x0880	WRAR00				ワイルドレジスタアドレスレジスタ 00
0x0884	WRDR00				ワイルドレジスタデータレジスタ 00
0x0888	WRAR01				ワイルドレジスタアドレスレジスタ 01
0x088C	WRDR01				ワイルドレジスタデータレジスタ 01
0x0890	WRAR02				ワイルドレジスタアドレスレジスタ 02
0x0894	WRDR02				ワイルドレジスタデータレジスタ 02
0x0898	WRAR03				ワイルドレジスタアドレスレジスタ 03
0x089C	WRDR03				ワイルドレジスタデータレジスタ 03
0x08A0	WRAR04				ワイルドレジスタアドレスレジスタ 04
0x08A4	WRDR04				ワイルドレジスタデータレジスタ 04
0x08A8	WRAR05				ワイルドレジスタアドレスレジスタ 05
0x08AC	WRDR05				ワイルドレジスタデータレジスタ 05
0x08B0	WRAR06				ワイルドレジスタアドレスレジスタ 06
0x08B4	WRDR06				ワイルドレジスタデータレジスタ 06
0x08B8	WRAR07				ワイルドレジスタアドレスレジスタ 07
0x08BC	WRDR07				ワイルドレジスタデータレジスタ 07
0x08C0	WRAR08				ワイルドレジスタアドレスレジスタ 08
0x08C4	WRDR08				ワイルドレジスタデータレジスタ 08
0x08C8	WRAR09				ワイルドレジスタアドレスレジスタ 09
0x08CC	WRDR09				ワイルドレジスタデータレジスタ 09
0x08D0	WRAR10				ワイルドレジスタアドレスレジスタ 10
0x08D4	WRDR10				ワイルドレジスタデータレジスタ 10
0x08D8	WRAR11				ワイルドレジスタアドレスレジスタ 11
0x08DC	WRDR11				ワイルドレジスタデータレジスタ 11
0x08E0	WRAR12				ワイルドレジスタアドレスレジスタ 12
0x08E4	WRDR12				ワイルドレジスタデータレジスタ 12
0x08E8	WRAR13				ワイルドレジスタアドレスレジスタ 13
0x08EC	WRDR13				ワイルドレジスタデータレジスタ 13
0x08F0	WRAR14				ワイルドレジスタアドレスレジスタ 14
0x08F4	WRDR14				ワイルドレジスタデータレジスタ 14
0x08F8	WRAR15				ワイルドレジスタアドレスレジスタ 15
0x08FC	WRDR15				ワイルドレジスタデータレジスタ 15

30.4.1 ワイルドレジスタデータイネーブルレジスタ : WREN (Wild Register data ENable register)

ワイルドレジスタデータイネーブルレジスタのビット構成について説明します。

チャネルごとに、ワイルドレジスタの機能の有効・無効を設定します。

■ WREN : アドレス 085A$_H$ (アクセス : ハーフワード)

	bit15	bit14	・	・	・	bit2	bit1	bit0
	WREN[15:0]							
初期値	0	0	・	・	・	0	0	0
属性	R/W	R/W	・	・	・	R/W	R/W	R/W

[bit15～bit0] WREN[15:0] (Wild Register ENable) : イネーブルビット
チャネルごとに、ワイルドレジスタ機能の有効・無効を設定します。

WRENn (n=0～15)	機能
0	ch.n のワイルドレジスタ機能は無効です
1	ch.n のワイルドレジスタ機能は有効です

30.4.2 ワイルドレジスタアドレスレジスタ 00～15 : WRAR00-15 (Wild Register Address Register 00-15)

ワイルドレジスタアドレスレジスタ 00～15 のビット構成について説明します。

ワイルドレジスタ機能により修正するアドレスを設定するレジスタです。ワイルドレジスタ動作許可中の読出し値は不定です。

レジスタの設定は必ず 32 ビット単位で行ってください。

■ WRAR : アドレス 0880H ～08F8H (アクセス: ワード)

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	予約		WRAR[21:16]					
初期値	0	0	X	X	X	X	X	X
属性	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	WRAR[15:8]							
初期値	X	X	X	X	X	X	X	X
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	WRAR[7:2]						予約	
初期値	X	X	X	X	X	X	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R0,WX	R0,WX

[bit21～bit2] WRAR[21:2] (Wild Register Address Register) : アドレスレジスタ
パッチ対象のアドレスを設定します。対象アドレスは(WRAR & 0x003FFFFC)となります。

ワイルドレジスタ動作許可中の読出し値は不定です。

30.4.3 ワイルドレジスタデータレジスタ 00～15 : WRDR00-15 (Wild Register Data Register00-15)

ワイルドレジスタデータレジスタ 00～15 について説明します

置き換えるデータを設定するレジスタです。ワイルドレジスタアドレスレジスタ(WRAR00～WRAR15)で指定したアドレスのメモリ内容を読み出すと、実際のメモリ内容ではなく、このレジスタに設定した値が読み出されます。
ワイルドレジスタ機能動作中の本レジスタの読出し値は不定になります。
レジスタの設定は必ず 32 ビット単位で行ってください。

■ WRDR： アドレス 0884H～08FCH (アクセス ： ワード)

	bit31	bit30	・	・	・	bit2	bit1	bit0
	WRDR[31:0]							
初期値	X	X	・	・	・	X	X	X
属性	R/W	R/W				R/W	R/W	R/W

[bit31～bit0] WRDR[31:0] (Wild Register Data Register) : **データレジスタ**
置換え値を設定します。

ワイルドレジスタ機能動作中の本レジスタの読出し値は不定になります。

30.5 動作説明

ワイルドレジスタの動作について説明します。

本機能を使用して Flash 領域にパッチを当てます。イネーブルレジスタがリセットで初期化されるため、使用する際はリセットごとに本レジスタを設定してください。

設定するアドレスは、重ならないように設定してください。アドレスが重なる場合の読出し値は不定です。
データのバイト並びは、ビッグエンディアンです。
置換え対象となる領域は、Flash 領域のみです。

30.6 使用例

ワイルドレジスタの使用例について示します。

本機能の使用例です。この例ではリセット解除後に外付けデバイスから本機能の設定を呼び出します。

図 30-2 使用例



31. クロックスーパバイザ

クロックスーパバイザの概要、特長、レジスタなどについて説明します。

31.1 概要

クロックスーパバイザの概要について説明します。

クロックになんらかの問題が生じて、意図せず停止した場合、内蔵 CR 発振器にクロックの代行をさせる事ができます。

メイン用のクロックスーパーバイザとサブ用のクロックスーパーバイザは独立しており、別々に許可、禁止の設定ができます。

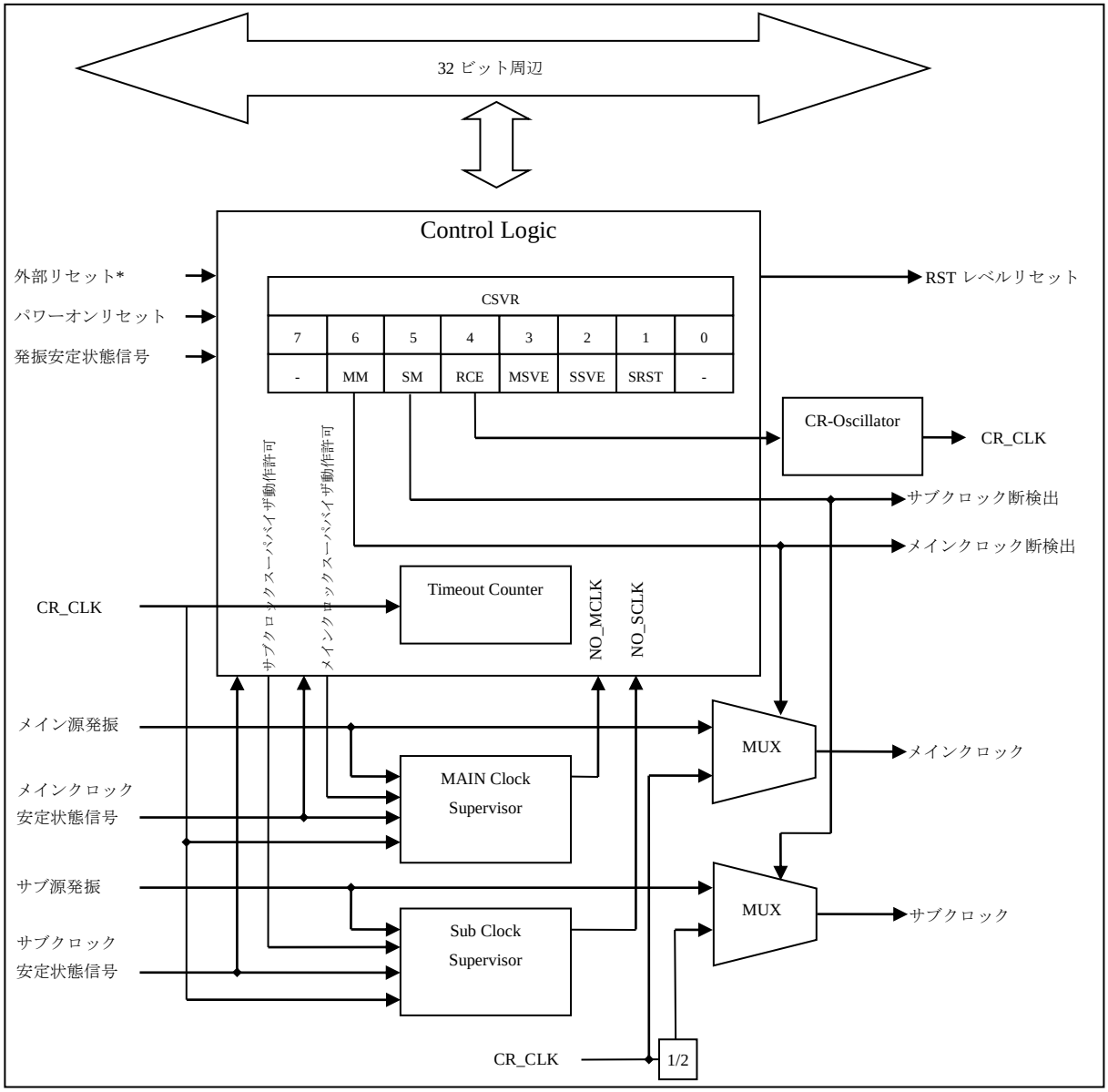
31.2 構成

クロックスーパバイザの構成について示します。

クロックスーパバイザを構成するブロックは、以下のとおりです。

- クロックスーパバイザ
- タイムアウトカウンタ
- 制御ロジック
- CR 発振器

図 31-1 クロックスーパバイザのブロックダイヤグラム (詳細)



*: 外部リセット: RSTX 端子のアサート(NMIX との同時アサートも含みます。)

＜注意事項＞

クロック 2 系統品の場合には、サブクロックスーパバイザが使用できます。

31.3 レジスタ

クロックスーパバイザのレジスタについて説明します。

表 31-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x056C	予約	CSVCR	予約	予約	クロックスーパバイザ制御レジスタ

■ クロックスーパバイザ制御レジスタ ：CSVCR(Clock SuperVisor Control Register)

本レジスタは、クロックスーパバイザの動作モードを設定します。
クロックの故障を示すビットが存在します。

● CSVCR：アドレス 056DH (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	MM	SM	RCE	MSVE	SSVE	SRST	予約
初期値	0	0	0	1	1/0	1	0	0
属性	R0/W0	R,W	R,W	R/W	R/W	R/W	R0/W0	R0/W0

（注意）bit3 の初期値は、型格によって異なります。

[bit7] 予約
必ず "0"を書き込んでください。

[bit6] MM (Main clock Missing) : メインクロック停止
このビットが"1" の場合、メイン発振クロックに問題が生じている事を示します。
このビットが"0" の場合、メインクロックには問題がありません。
メインクロックが復旧していない時、"0"書込みは無視されます。
このビットは、電源投入時もしくは外部リセットにより"0" にクリアされます。ほかの種類のリセットは、このビットに影響を与えません。

MM	読出し	書込み
0	メイン発振クロック停止未検出	メインクロックが復旧している時、 本ビットをクリア
1	メイン発振クロック停止検出	効果ありません

＜注意事項＞

本ビットが"1"の場合、PLL/SSCG 発振動作は許可しないでください。

[bit5] SM (Sub clock Missing) : **サブクロック停止**
このビットが"1" の場合、サブ発振クロックに問題が生じている事を示します。
このビットが"0" の場合、サブクロックに問題はありません。
サブクロックが復旧していない時、"0"書込みは無視されます。
このビットは、電源投入時もしくは外部リセットにより"0" にクリアされます。ほかの種類のリセットは、このビットに影響を与えません。

SM	読出し	書込み
0	サブ発振クロック停止未検出	サブクロックが復旧している時、本ビットをクリア
1	サブ発振クロック停止検出	効果ありません

[bit4] RCE (CR-oscillator Enable) : CR **発振許可**
このビットを"1" に設定すると、スタンバイモード時にも CR 発振器の発振が許可されます。
メインクロックスーパバイザまたはサブクロックスーパバイザがまだ許可されている間は、このビットを"0" に設定することは禁止です。
まずスーパバイザを禁止してから、MM ビットおよび SM ビットが"0" であることを確認する必要があります。その後、RCE を"0" に設定してください。
MM ビットまたは SM ビットのいずれかが"1" の場合は、RCE を"0" に設定しないでください。
このビットは、電源投入時もしくは外部リセットにより"1" にクリアされます。ほかの種類のリセットは、このビットに影響を与えません。

RCE	説明
0	STBY モード時の CR 発振禁止
1	STBY モード時の CR 発振許可(初期値)

[bit3] MSVE (Main clock SuperVisor Enable) : **メインクロックスーパバイザ許可**
このビットを"1" に設定すると、メインクロックスーパバイザが許可されます。
このビットは、電源投入時にのみ"1" に初期化されます。
ほかの種類のリセットは、このビットに影響を与えません。

MSVE	説明
0	メインクロックスーパバイザ禁止（初期値 OFF 型格の初期値）
1	メインクロックスーパバイザ許可(初期値 ON 型格の初期値)

＜注意事項＞
型格によりメインクロックスーパバイザの初期状態が異なります。そのため、本ビットの初期値は型格によって異なります。型格の詳細は、『1.3 品種構成』を参照してください。

[bit2] SSVE (Sub clock SuperVisor Enable) : **サブクロックスーパバイザ許可**
このビットを"1" に設定すると、サブクロックスーパバイザが許可されます。
このビットは、電源投入時にのみ"1" に初期化されます。
ほかの種類のリセットは、このビットに影響を与えません。

SSVE	説明
0	サブクロックスーパバイザ禁止
1	サブクロックスーパバイザ許可(初期値)

[bit1] SRST(Sub clock mode reset) : **サブクロックモード・リセット**
必ず "0"を書き込んでください。

SRST	説明
0	サブクロック断状態でメインクロックからサブクロックに切り換えたときにリセットを発行しない(初期値)
1	サブクロック断状態でメインクロックからサブクロックに切り換えたときにリセットを発行する

[bit0] **予約**
必ず "0"を書き込んでください。

31.4 動作説明

クロックスーパバイザの動作について説明します。

CPU がメインクロックで動作中にメインクロックが停止した場合は、クロックが CR 発振器に置き換わった後、直ちにリセットされます。30µs～40µs の期間、クロックが入力されない場合、停止と判断します。メインクロックが停止した事を示すビットはレジスタに残るので、ソフトウェアで問題が起きた事を判断可能です。

CPU がサブクロックで動作中にサブクロックが停止した場合は、クロックが CR 発振器に置き換わった後, 直ちにリセットされます。310µs～320µs の期間、クロックが入力されない場合, 停止と判断します。サブクロックが停止した事を示すビットはレジスタに残るので、ソフトウェアで問題が起きた事を判断可能です。

CPU がメインクロックで動作中にサブクロックが停止した場合は、即時リセットは発生しません。サブクロックモードに移行した場合は, CR クロックで動作します。

意図してメインクロックを停止させた場合、メインクロックスーパバイザは自動的に停止します。意図してサブクロックを停止させた場合, サブクロックスーパバイザは自動的に停止します。

スタンバイモード時の CR 発振を禁止しておくと、スタンバイモード遷移時、CR 発振器は自動的に停止します。スタンバイモードから復帰するときに、CR 発振器は自動的に再起動します。

＜注意事項＞

メインクロック停止検出後、メインクロックが CR 発振器に置き換わって動作しているとき、PLL/SSCG 発振動作は許可しないでください。

以下にクロックスーパバイザの動作モードについて説明します。

31.4.1 初期状態

クロックスーパバイザの初期状態について説明します。

初期設定時は、CR 発振器の発振、メインクロックスーパバイザ機能およびサブクロックスーパバイザ機能が許可されています。

■ CR 発振

電源投入時に発振が許可されます。

スタンバイモード時の発振許可ビット(CSVCR:RCE)に"0" を書き込んだ状態で、スタンバイモードに遷移したときのみ停止します。スタンバイモードが解除されると自動的に発振を再開します。

■ メインクロックスーパバイザ

初期値 ON 型格では、メイン発振安定待ち時間経過後に許可されます。

初期値 OFF 型格では、初期状態は不許可です。クロックスーパバイザの再許可を行うことで、許可することが可能です。

メインクロックスーパバイザが許可されているときに、メインクロックが停止した場合、メインクロックは CR 発振クロックで置き換えられます。

このとき、CSVCR レジスタの MM ビットが"1"になり、RST レベルのリセットが発生します。

【補足】

初期値 ON 型格について、メイン発振安定待ち時間は、メインクロック自身で計られますので、発振安定待ち時間経過前にメインクロックが停止してしまうと、メイン発振安定待ち時間が終了せず、メインクロックスーパバイザが許可されません。

その場合は、内部 CR 発振器で計られるタイムアウト時間が経過した後に、発振安定待ち時間とは無関係に、メインスーパバイザ機能が許可され、メインクロック停止が検知されます。

■ サブクロックスーパバイザ

内部 CR 発振器で計られるタイムアウト時間が経過した後に許可されます。

サブクロックスーパバイザが許可されているときに、サブクロックが停止した場合の振舞いは MCU がメインクロックで動作しているかサブクロックで動作しているかによって変わります。

● メインクロックモードの場合

メインクロックモードで動作中にサブクロックが停止した場合、サブクロックは CR 発振クロックの 2 分周に置き換わります。その後、CSVCR レジスタの SM ビットが"1" に設定されますが、リセットは発生されずメインクロックモードで動作し続けます。

この状態で、サブクロックモードに遷移した場合、CR 発振クロックで動作するサブクロックモードに遷移します。

● サブクロックモードの場合

サブクロックモードで動作中にサブクロックが停止した場合、CR 発振クロックの 2 分周がサブクロックに置き換わります。その後、CSVCR レジスタの SM ビットが"1" に設定されて、RST レベルのリセットが発生します。

31.4.2　CR 発振器およびクロックスーパバイザ機能の停止

CR 発振器およびクロックスーパバイザ機能の停止について説明します。

■ CR 発振器

CR 発振器は、スタンバイモード時のみ停止させる事ができます。スタンバイモード時の発振許可ビット(CSVCR:RCE)を"0" に設定してから、スタンバイモードに遷移してください。

メインクロックもしくはサブクロックに問題があるときに CR 発振器を停止することは禁止です。クロックに問題があるかないかは、CSVCR レジスタの MM ビットおよび SM ビットで確認できます。

【補足】

クロックに問題がある場合は、既に動作クロックが CR 発振クロックに置き換わっているので、CR 発振を停止すると動作クロックも停止してしまいます。

■ メインクロックスーパバイザ

CSVCR レジスタの MSVE ビットを"0"に設定します。

■ サブクロックスーパバイザ

CSVCR レジスタの SSVE ビットを"0"に設定します。

31.4.3　クロックスーパバイザ再許可

クロックスーパバイザの再許可について説明します。

■ メインクロックスーパバイザ

メインクロックスーパバイザ機能を再許可するには、CSVCR レジスタの MSVE ビットを"1"に設定します。

CR 発振器が停止している状態で、メインクロックスーパバイザ機能を許可することは禁止です。

【補足】

メイン発振安定待ち時間は、メインクロック自身で計られますので、発振安定待ち時間経過前にメインクロックが停止してしまうと、メイン発振安定待ち時間が終了せず、メインクロックスーパバイザが許可されません。

その場合は、内部 CR 発振器で計られるタイムアウト時間が経過した後に、発振安定待ち時間とは無関係に、メインスーパバイザ機能が許可され、メインクロック停止が検知されます。

■ サブクロックスーパバイザ

サブクロックスーパバイザ機能を再許可するには、CSVCR レジスタの SSVE ビットを"1"に設定します。CR 発振器が停止している状態で、サブクロックスーパバイザ機能を許可することは禁止です。

31.4.4　サブクロックモード

サブクロックモードについて説明します。

メインクロックスーパーバイザ機能が許可されている状態で、デバイスがサブクロックモードに遷移すると、メインクロックスーパバイザ機能は自動的に停止します。

許可ビット(CSVCR:MSVE)は"0" になりません。

デバイスが, サブクロックモードからメインクロックモードに遷移すると、メインクロックの発振安定待ち時間経過後に、メインクロックスーパバイザ機能は再び許可されます。

【補足】
メイン発振安定待ち時間は、メインクロック自身で計られますので、発振安定待ち時間経過前にメインクロックが停止してしまうと、メイン発振安定待ち時間が終了せず、メインクロックスーパバイザが許可されません。

その場合は、内部 CR 発振器で計られるタイムアウト時間が経過した後に、発信安定待ち時間とは無関係に、メインスーパバイザ機能が許可され、メインクロック停止が検知されます。

31.4.5　ストップモード

ストップモードについて説明します。

■ CR 発振器

ストップモード時の発振許可ビット(CSVCR:RCE)が"0" に設定されている場合、ストップモードに遷移することで発振が停止します。

ストップモード解除後、自動的に再許可されます。

■ メインクロックスーパバイザ

メインクロックスーパバイザ機能が許可されている状態で、ストップモードに遷移すると、自動的に停止します。

メインクロックスーパバイザ許可ビット(CSVCR:MSVE)は"0"になりません。

ストップモード解除後、メイン発振安定待ち時間を待って、自動的に再許可されます。

【補足】

メイン発振安定待ち時間は、メインクロック自身で計られますので、発振安定待ち時間経過前にメインクロックが停止してしまうと、メイン発振安定待ち時間が終了せず、メインクロックスーパバイザが許可されません。

その場合は、内部 CR 発振器で計られるタイムアウト時間が経過した後に、発振安定待ち時間とは無関係に、メインスーパバイザ機能が許可され、メインクロック停止が検知されます。

＜注意事項＞

メインクロックスーパバイザ機能が禁止されている状態で、ストップモードに遷移した場合は、ストップモードが解除後も禁止のままです。

■ サブクロックスーパバイザ

サブクロックスーパバイザ機能が許可されている状態で、ストップモードに遷移すると、自動的に停止します。

サブクロックスーパバイザ許可ビット(CSVCR:SSVE)は"0"になりません。

ストップモード解除後、メイン発振安定待ち時間を待って、自動的に再許可されます。

＜注意事項＞

サブクロックスーパバイザ機能が禁止されている状態で、ストップモードに遷移した場合は、ストップモードが解除後も禁止のままです。

31.4.6 時計モード

時計モードについて説明します。

■ メインクロックスーパバイザ

メインクロックスーパバイザ機能は時計モードへの遷移に影響を受けません。

メインクロックスーパバイザ機能が許可されていてかつRTCにメインクロックが接続されている場合は、メインクロックが停止するとCR発振クロックに切換わり、リセットを発行します。時計モードは解除され、RTCは初期化されます。

メインクロックスーパバイザ機能が禁止されていてかつRTCにメインクロックが接続されている場合は、メインクロックが停止しても検出をしないため、単にRTCクロックが停止します。

■ サブクロックスーパバイザ

サブクロックスーパバイザ機能は時計モードへの遷移に影響を受けません。

サブクロックスーパバイザ機能が許可されていてかつRTCにサブクロックが接続されている場合は、サブクロックが停止するとCR発振クロックに切り換わりますが、リセットは発行されません。

サブクロックスーパバイザ機能が禁止されていてかつRTCにサブクロックが接続されている場合は、サブクロックが停止しても検出をしないため、単にRTCクロックが停止します。

31.4.7　クロックスーパバイザによるリセット要因の確認

クロックスーパバイザによるリセット要因の確認について説明します。

クロックスーパバイザがクロックの問題を検知してリセットが発生したかどうかを調べる方法は以下のとおりです。

まず、RSTRR レジスタ(『リセット』の章の『4.1. リセット要因レジスタ: RSTRR (ReSeT Result Register)』参照)を読み出してリセット要因を確認します。

RSTRR レジスタの ERST ビットが"1"になっている場合、RSTX 外部端子からのリセット入力、不正スタンバイモード移行検出リセット、外部電源低電圧検出、クロックスーパバイザリセット、もしくは RSTX 外部端子と NMIX 外部端子の同時アサートのいずれかが発生したことを示します。

この場合に、CSVCR レジスタを読み出して、MM ビットを確認してください。合わせて、RSTRR レジスタ(『リセット』の章の『4.1. リセット要因レジスタ: RSTRR (ReSeT Result Register)』参照)を読み出して、リセット要因を確認してください。

リセット要因は、以下のように確認できます。

MM	SM	リセット要因
1	0	メインクロックスーパバイザリセット
0	1	サブクロックスーパバイザリセット
1	1	メインクロックスーパバイザリセット もしくは サブクロックスーパバイザリセット

【補足】
MM/SM ビットは、電源投入および外部リセット以外ではクリアされませんので、RSTRR レジスタ（『リセット』の章の『4.1. リセット要因レジスタ: RSTRR (ReSeT Result Register)』を参照）を読み出してほかのリセット要因も合わせて確認する必要があります。

31.4.8 CR クロックからの切戻り

CR クロックからの切戻りについて説明します。

■ メインクロックスーパバイザ

MPU はリセットからの回復後に MM ビットがセットされていたことを検出すると、メインクロックが停止して CR 発振クロックに切り換っていると判断できます。このとき、メインクロックが復旧していることを確認できれば、MM ビットに"0"を書き込むことによって、メインクロックに戻すことが可能です。

メインクロックが復旧していない場合、MM ビットへの"0"書込みは何の影響もあたえません。MM ビットは"1"を保持し続けます。

MM ビットに"0"を書き込んだときにメインクロックが動作していた場合、MM ビットはクリアされ、クロックは同期ステージを経由してメインクロックに戻ります。

メインクロックが復旧するまで MM ビットをポーリングすることができます。

```
ldi #_csvcr,r1
clear_CSV_loop:
bandh #0b1001,@r1 ;; Clear MM+SM
btsth #0b0110,@r1 ;; Check: Is one of them 1?
bne clear_CSV_loop
```

＜注意事項＞

メインクロックの切戻しを行う場合、PMUCTLR.SHDE に"0"を設定してください。

■ サブクロックスーパバイザ

MPU は SM ビットがセットされていたことを検出すると、サブクロックが停止して CR 発振クロックに切り換っていると判断できます。このとき、サブクロックが復旧していることを確認できれば、SM ビットに"0"を書き込むことによって、サブクロックに戻すことが可能です。

サブクロックが復旧していない場合、SM ビットへの"0"書込みは何の影響もあたえません。SM ビットは"1"を保持し続けます。

SM ビットに"0"を書き込んだときにサブクロックが動作していた場合、SM ビットはクリアされ、クロックは同期ステージを経由してサブクロックに戻ります。

サブクロックが復旧するまで SM ビットをポーリングすることができます。(メインクロックスーパバイザと同じ方法が使用できます。)

＜注意事項＞

サブクロックの切戻しを行う場合、PMUCTLR.SHDE に"0"を設定してください。

32. サウンドジェネレータ

サウンドジェネレータについて説明します。

32.1 概要

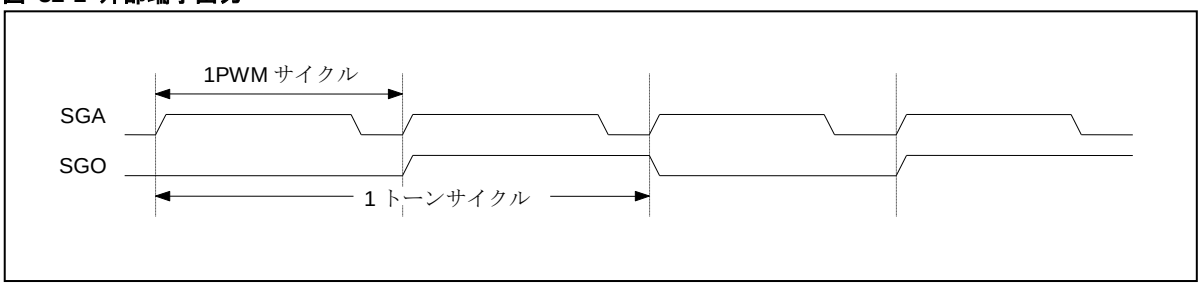
サウンドジェネレータの概要について説明します。

本シリーズには 5 チャネルのサウンドジェネレータが搭載されています。サウンドジェネレータは、CPU からの設定にしたがって、トーンパルス信号(またはトーンパルス信号と PWM パルス信号の混合信号)、および PWM パルス信号を生成し出力します。出力するトーンパルス信号の周波数、音量(PWM パルスの振幅)、音の長さは設定可能です。

サウンドジェネレータは、以下のレジスタおよびカウンタで構成されています。

- DMA 転送更新許可レジスタ
- サウンドコントロールレジスタ
- 振幅データレジスタ
- 周波数データレジスタ
- 周期レジスタ
- トーン出力数レジスタ
- 増減量データレジスタ
- PWM サイクル数データレジスタ
- DMA 転送間接レジスタ
- PWM パルス生成器
- 周波数カウンタ
- デクリメントカウンタ
- トーンパルスカウンタ

図 32-1 外部端子出力



32.2 特長

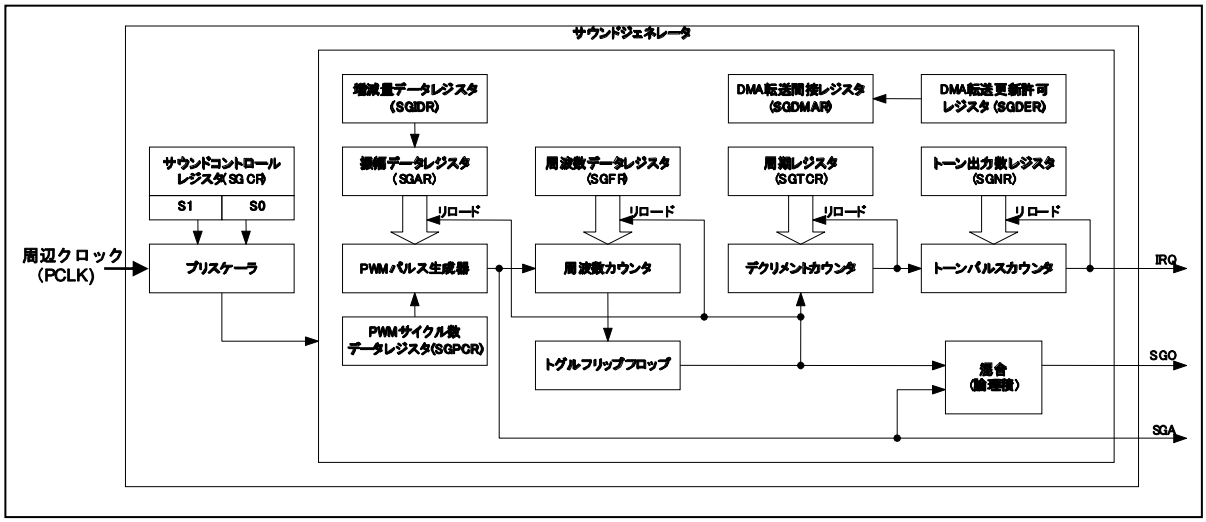
サウンドジェネレータの特長について説明します。

No	項目	機能
1	動作クロック	・周辺クロック(PCLK) 16MHz～40MHz
2	クロック入力	サウンドジェネレータのクロック入力は、周辺クロック(PCLK)を分周して使用します。 ・周辺クロック(PCLK) ・1/2 × 周辺クロック(PCLK) ・1/4 × 周辺クロック(PCLK) ・1/8 × 周辺クロック(PCLK)
3	波形	・サウンド用矩形波 (SGO 端子にてサウンド出力)
4	音量	・任意の音量に設定可能 (SGA 端子にて振幅出力)
5	周波数	・サウンド信号周波数を任意の値に設定可能 (周波数設定および PWM サイクル数設定)
6	音の長さ	・任意に設定可能
7	割込み	・設定された音の長さを出力し終えると割込み要求を発生させることが可能 (トーンパルスカウンタのオーバフロー) ・DMA モード(SGCR:DMA="1")時、スタートビット(SGCR:ST)に"1"を書き込んだ場合に割込みを発生させることが可能

32.3 構成

サウンドジェネレータの構成について示します。

図 32-2 ブロックダイヤグラム



32.4 レジスタ

サウンドジェネレータのレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子表

表 32-1 ベースアドレス・外部端子表

チャネル	Base_addr	外部端子	
		SGO	SGA
0	0x1040	SGO0	SGA0
1	0x1060	SGO1	SGA1
2	0x1080	SGO2	SGA2
3	0x10A0	SGO3	SGA3
4	0x10C0	SGO4_0/SGO4_1	SGA4_0/SGA4_1

■ レジスタマップ

表 32-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x1040	予約	SGDER0	SGCR0		DMA 転送更新レジスタ 0 サウンドコントロールレジスタ 0
0x1044	SGAR0		SGFR0	SGNR0	振幅データレジスタ 0 周波数データレジスタ 0 トーン出力数レジスタ 0
0x1048	SGTCR0	SGIDR0	SGPCR0		周期レジスタ 0 増減量データレジスタ 0 PWM サイクル数レジスタ 0
0x104C	SGDMAR0				DMA 転送間接レジスタ 0
0x1060	予約	SGDER1	SGCR1		DMA 転送更新レジスタ 1 サウンドコントロールレジスタ 1
0x1064	SGAR1		SGFR1	SGNR1	振幅データレジスタ 1 周波数データレジスタ 1 トーン出力数レジスタ 1
0x1068	SGTCR1	SGIDR1	SGPCR1		周期レジスタ 1 増減量データレジスタ 1 PWM サイクル数レジスタ 1
0x106C	SGDMAR1				DMA 転送間接レジスタ 1
0x1080	予約	SGDER2	SGCR2		DMA 転送更新レジスタ 2 サウンドコントロールレジスタ 2

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x1084	SGAR2		SGFR2	SGNR2	振幅データレジスタ 2 周波数データレジスタ 2 トーン出力数レジスタ 2
0x1088	SGTCR2	SGIDR2	SGPCR2		周期レジスタ 2 増減量データレジスタ 2 PWM サイクル数レジスタ 2
0x108C	SGDMAR2				DMA 転送間接レジスタ 2
0x10A0	予約	SGDER3	SGCR3		DMA 転送更新レジスタ 3 サウンドコントロールレジスタ 3
0x10A4	SGAR3		SGFR3	SGNR3	振幅データレジスタ 3 周波数データレジスタ 3 トーン出力数レジスタ 3
0x10A8	SGTCR3	SGIDR3	SGPCR3		周期レジスタ 3 増減量データレジスタ 3 PWM サイクル数レジスタ 3
0x10AC	SGDMAR3				DMA 転送間接レジスタ 3
0x10C0	予約	SGDER4	SGCR4		DMA 転送更新レジスタ 4 サウンドコントロールレジスタ 4
0x10C4	SGAR4		SGFR4	SGNR4	振幅データレジスタ 4 周波数データレジスタ 4 トーン出力数レジスタ 4
0x10C8	SGTCR4	SGIDR4	SGPCR4		周期レジスタ 4 増減量データレジスタ 4 PWM サイクル数レジスタ 4
0x10CC	SGDMAR4				DMA 転送間接レジスタ 4

32.4.1 DMA 転送更新許可レジスタ ：SGDER (SG DMA Enable Register)

DMA 転送更新許可レジスタ (SGDER)のビット構成について説明します。

DMA 転送更新許可レジスタ(SGDER)は、DMA 転送時に更新するレジスタ(SGAR, SGFR, SGNR, SGTCR, SGIDR, SGPCR)をバイト単位に設定するレジスタです。サウンドジェネレータは、本レジスタの設定にしたがって、DMA 転送時に更新するレジスタを判断します。また、本レジスタの設定により、DMA 転送時の転送回数、転送バイト数、および DMA 転送間接レジスタ(SGDMAR)の有効バイト位置を判断します。

■ SGDER： アドレス Base_addr + 01H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ARE1	ARE0	FRE	NRE	TCRE	IDRE	PCRE1	PCRE0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ビット名		機能
bit7	ARE1: 振幅データ(上位バイト)更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介して振幅データレジスタ(SGAR)の振幅データ(上位バイト)の更新を許可します。 "0"に設定した場合: 振幅データ(上位バイト)を更新しません。 "1"に設定した場合: 振幅データ(上位バイト)を更新します。
bit6	ARE0: 振幅データ(下位バイト)更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介して振幅データレジスタ(SGAR)の振幅データ(下位バイト)の更新を許可します。 "0"に設定した場合: 振幅データ(下位バイト)を更新しません。 "1"に設定した場合: 振幅データ(下位バイト)を更新します。
bit5	FRE: 周波数データ更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介して周波数データレジスタ(SGFR)の周波数データの更新を許可します。 "0"に設定した場合: 周波数データを更新しません。 "1"に設定した場合: 周波数データを更新します。
bit4	NRE: トーン出力数更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介してトーン出力数レジスタ(SGNR)のトーン出力数の更新を許可します。 "0"に設定した場合: トーン出力数を更新しません。 "1"に設定した場合: トーン出力数を更新します。
bit3	TCRE: 周期更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介して周期レジスタ(SGTCR)の周期の更新を許可します。 "0"に設定した場合: 周期を更新しません。 "1"に設定した場合: 周期を更新します。
bit2	IDRE: 増減量データ更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介して増減量データレジスタ(SGIDR)の増減量データの更新を許可します。 "0"に設定した場合: 増減量データを更新しません。 "1"に設定した場合: 増減量データを更新します。
bit1	PCRE1: PWM サイクル数データ(上位バイト)更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介して PWM サイクル数データレジスタ(SGPCR)の PWM サイクル数データ(上位バイト)の更新を許可します。 "0"に設定した場合: PWM サイクル数データ(上位バイト)を更新しません。 "1"に設定した場合: PWM サイクル数データ(上位バイト)を更新します。

ビット名		機能
bit0	PCRE0: PWMサイクル数データ(下位バイト)更新許可ビット	DMA 転送時に、DMA 転送間接レジスタを介して PWM サイクル数データレジスタ(SGPCR)の PWM サイクル数データ(下位バイト)の更新を許可します。 "0"に設定した場合: PWM サイクル数データ(下位バイト)を更新しません。 "1"に設定した場合: PWM サイクル数データ(下位バイト)を更新します。

32.4.2 サウンドコントロールレジスタ ：SGCR (SG Control Register)

サウンドコントロールレジスタ (SGCR) のビット構成について説明します。

サウンドコントロールレジスタ(SGCR)は、サウンドジェネレータの割込みの制御や動作状態の制御を行います。

■ SGCR： アドレス Base_addr + 02H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	SRST	DMA	GID	GEN	予約	BUSY	予約
初期値	0	0	0	0	0	0	0	0
属性	R/W0	R0,W	R/W	R/W	R/W	R0,W0	R/W	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	S1	S0	TONE	予約		INTE	INT	ST
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R(RM1),W	R,W

ビット名		機能
bit15	予約:	予約ビットです。 常に"0"を書き込んでください。
bit14	SRST: ソフトウェアリセットビット	ソフトウェアリセットビットです。 本ビットに"1"を設定するとハードウェアはソフトウェアリセットを発行します。 リード時、常に"0"が読み出され、本ビットに"0"を設定しても無効です。
bit13	DMA: DMA 転送開始割込みセット許可ビット	DMA 転送開始割込みセット許可ビットです。 "0"に設定した場合: スタートビット(SGCR:ST)に"1"を書き込んだときに、割込みビット(SGCR:INT)がセットされません。 "1"に設定した場合: スタートビット(SGCR:ST)に"1"を書き込んだときに、割込みビット(SGCR:INT)がセットされます。 (注意事項) サウンドジェネレータ動作中(SGCR:ST=1)に、本ビットの設定を変更しないでください。
bit12	GID: 増減設定ビット	周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、および自動増減許可ビット(SGCR.GEN)とともに、サウンドの自動増減用に設計されています。周期レジスタが指定したトグルフリップフロップからのトーンパルス数をデクリメントカウンタがカウントする(増減設定反映タイミング)ごとに、振幅データレジスタに記憶されている値を増減量データレジスタの値だけ、増減設定ビットの設定にしたがって増減します。 "0"に設定した場合: 振幅データレジスタに記憶されている値を減少させます。"1"に設定した場合: 振幅データレジスタに記憶されている値を増加させます。 (注意事項) 自動増減許可ビットが許可設定(SGCR:GEN=1)時のみ有効です。 動作中に設定を変更した場合、増減設定反映タイミングのレジスタ値が有効になります。

ビット名		機能
bit11	GEN: 自動増減許可ビット	周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、および増減設定ビット(SGCR:GID)とともに、サウンドの自動増減用に設計されています。 "0"に設定した場合: サウンドの自動増減を禁止します。 "1"に設定した場合: サウンドの自動増減を許可します。周期レジスタが指定したトグルフリップフロップからのトーンパルス数をデクリメントカウンタがカウントする(増減設定反映タイミング)ごとに、振幅データレジスタに記憶されている値を増減量データレジスタの値だけ、増減設定ビットの設定にしたがって増減します。 (注意事項) 自動増減許可設定から禁止設定にした場合、振幅データレジスタの値は、現在の値を保持します。自動増減禁止設定から許可設定にした場合、振幅データレジスタの値は、増減設定反映タイミングから、現在の値を増減します。
bit10	予約:	予約ビットです。リード時、"0"が読み出され、ライト時、"0"を書き込んでください。
bit9	BUSY: ビジービット	サウンドジェネレータが動作中であるかどうかを示します。 "1"になる条件: SGCR:ST ビットを"1"に設定すると、"1"にセットされます。 "0"になる条件: SGCR:ST ビットを"0"に設定し、1 トーンサイクル終了時に動作が完了すると、"0"にクリアされます。 (注意事項) ソフトウェアリセット(SGCR:SRST=1 ライト)により、本ビットは"0"になります。
bit8	予約:	予約ビットです。リード時、"0"が読み出され、ライト時、"0"を書き込んでください。
bit7, bit6	S1, S0: 動作クロック選択ビット	サウンドジェネレータ用のクロック入力信号を指定します。 {S1,S0}　クロック入力 00　1/1 入力クロック 01　1/2 入力クロック 10　1/4 入力クロック 11　1/8 入力クロック
bit5	TONE: トーン出力ビット	SGO 信号を設定します。 "0"に設定した場合: SGO 信号はトーンパルスと PWM パルスの混合(論理積)信号になります。 "1"に設定した場合: SGO 信号はトグルフリップフロップからの単純方形波形(トーンパルス)信号になります。
bit4	予約	予約ビットです。書込み値は無視されます。
bit3	予約	予約ビットです。書込み値は無視されます。
bit2	INTE: 割込み許可ビット	サウンドジェネレータの割込み信号を許可します。 "0"に設定した場合: SGCR:INT ビットによる割込みを禁止します。 "1"に設定した場合: SGCR:INT ビットによる割込みを許可します。SGCR:INT=1 で割込み信号を出力します。
bit1	INT: 割込みビット	トーンパルスカウンタが、トーン出力数レジスタおよび周期レジスタに指定したトーンパルス数をカウントすると、このビットは"1"に設定されます。 また、DMA モード(SGCR:DMA=1)時、振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)、周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)のいずれのレジスタにも書込みを行わずにスタートビット(SGCR:ST)に"1"を書き込んだ場合にもこのビットは"1"となり、これは DMA 転送の開始要求として使用できます。 このビットは、"0"を書き込むと"0"にクリアされますが、"1"を書き込んでも無効となり、前の値を保持します。 "1"になる条件: トーンパルスカウント数 ≧ (周期レジスタ値+1) × (トーン出力数レジスタ値+1) "0"になる条件: "0"を書き込むと、"0"にクリアされます。 (注意事項) リードモディファイライト命令のリードアクセス時は、常に"1"が読み出されます。 ソフトウェアリセット(SGCR:SRST=1 ライト)により、本ビットは"0"になります。

ビット名		機能
bit0	ST: スタートビット	サウンドジェネレータの動作開始用です。 "1"に設定した場合: サウンドジェネレータの動作を開始します。 "0"に設定した場合: サウンドジェネレータは現行のトーンサイクル終了時に動作を停止します。 (注意事項) このビットが"1"である間は、サウンドジェネレータは動作しています。"0"に設定した場合は、このビットは"0"にクリアされますが、サウンドジェネレータは現行のトーンサイクル終了時に動作を停止します。サウンドジェネレータが完全に停止したかどうかは、SGCR:BUSY ビットで示されます。 ソフトウェアリセット(SGCR:SRST=1 ライト)により、本ビットは"0"になります。

32.4.3 振幅データレジスタ ：SGAR (SG Amplitude Register)

振幅データレジスタ (SGAR) のビット構成について説明します。

振幅データレジスタ(SGAR)は、PWM パルス生成器に対するリロード値を記憶します。レジスタ値はサウンドの振幅をあらわします。レジスタ値はトーンサイクルが終わるごとに PWM パルス生成器にリロードされます。

■ SGAR： アドレス Base_addr + 04H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	・	・	・	bit2	bit1	bit0
	D[15:0]							
初期値	0	0	・	・	・	0	0	0
属性	R/W	R/W	・	・	・	R/W	R/W	R/W

[bit15～bit0] D[15:0] (Data)： **振幅データビット**
PWM パルス生成器に対するリロード値を記憶します。ソフトウェアは、PWM パルス生成器に対するリロード値を設定します。ハードウェアは、増減許可ビットが許可(SGCR:GEN=1)のとき、周期レジスタ(SGTCR)が指定したトグルフリップフロップからのトーンパルス数をデクリメントカウンタがカウントするごとに、振幅データレジスタに記憶されている値を増減量データレジスタ(SGIDR)の値だけ、増減設定ビット(SGCR:GID)の設定にしたがって増減します。増減設定ビットが減少設定(SGCR:GID=0)のとき、振幅データレジスタ値が"0x0000"になると、それ以上のデクリメントはしません。増減設定ビットが増減設定(SGCR:GID=1)のとき、振幅データレジスタ値が"0xFFFF"になると、それ以上のインクリメントはしません。ただし、サウンドジェネレータは、スタートビット(SGCR:ST)がクリアされるまで動作し続けます。

32.4.4 周波数データレジスタ ：SGFR (SG Frequency Register)

周波数データレジスタ (SGFR) のビット構成について説明します。

周波数データレジスタ(SGFR)は、周波数カウンタに対するリロード値を記憶します。記憶された値は、サウンド(またはトグルフリップフロップからのトーン信号)の周波数を表します。レジスタ値は、トグル信号が遷移するたびにカウンタにリロードされます。

■ SGFR： アドレス Base_addr + 06H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] D[7:0] (Data) : **周波数データビット**
周波数カウンタに対するリロード値を記憶します。ソフトウェアは、周波数カウンタに対するリロード値を設定します。ハードウェアは、サウンド(またはトグルフリップフロップからのトーン信号)の周波数に使用します。レジスタ値は、トグル信号が遷移するたびにカウンタにリロードされます。

＜注意事項＞
動作中にレジスタ値を変更すると、変更タイミングによって、サウンド(またはトグルフリップフロップからのトーン信号)のデューティサイクルの 50%が変更されることもありますので、注意してください。

32.4.5 トーン出力数レジスタ ：SGNR (SG tone Number Register)

トーン出力数レジスタ (SGNR) のビット構成について説明します。

トーン出力数レジスタ(SGNR)は、トーンパルスカウンタに対するリロード値を記憶します。トーンパルスカウンタはトーンパルスの数(またはサウンドの増幅動作の回数)を蓄積して、それがリロード値に達したとき、割込みビット(SGCR:INT)をセットします。これらは割込みの頻度を減少させることを目的としています。

■ SGNR： アドレス Base_addr + 07H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] D[7:0] (Data) : トーン出力数ビット
トーンパルスカウンタに対するリロード値を記憶します。ソフトウェアは、トーンパルスカウンタに対するリロード値を設定します。ハードウェアでは、トーンパルスカウンタでトーンパルスの数(またはサウンドの増幅動作の回数)を蓄積して、それがリロード値に達したとき、割込みビット(SGCR:INT)をセットします。

トーンカウントレジスタに"0x00"が設定されると、トーンパルスカウンタは、デクリメントカウンタからのキャリーアウト信号ごとに、SGCR:INT ビットを設定します。累積トーンパルス数は次式で表されます。なお、周期レジスタはデクリメントカウンタのリロード値を記憶するものです。

(周期レジスタ値+1) ×(トーン出力数レジスタ値+1)

トーン出力レジスタおよび周期レジスタの両レジスタが"0x00"に設定されると、トーンサイクルごとに割込みビット (SGCR:INT) がセットされます。

32.4.6　周期レジスタ ：SGTCR (SG Tone Cycle Register)

周期レジスタ (SGTCR) のビット構成について説明します。

周期レジスタ(SGTCR)は、デクリメントカウンタに対するリロード値を記憶します。これは振幅データレジスタに記憶した値を自動的に増減できるように設計されています。

自動増減許可ビット(SGCR:GEN)が許可設定で、周期レジスタが指定したトグルフリップフロップからのトーンパルス数をデクリメントカウンタがカウントするごとに、振幅データレジスタ(SGAR)に記憶されている値を増減量データレジスタ(SGIDR)の値だけ、増減設定ビット(SGCR:GID)の設定にしたがって増減します。

この動作によって、サウンドの自動増減動作が、より少ない CPU 介入でできるようになります。

このレジスタが指定したパルス数は、「レジスタ値+1」であることに注意してください。"0x00"に設定されると、自動増減動作はトーンサイクルごとに行われます。

■ SGTCR : アドレス Base_addr + 08H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] D[7:0] (Data) : 周期ビット
デクリメントカウンタに対するリロード値を記憶します。ソフトウェアは、デクリメントカウンタに対するリロード値を設定します。ハードウェアは、周期レジスタが指定したトグルフリップフロップからのトーンパルス数をデクリメントカウンタがカウントするごとに、振幅データレジスタ(SGAR)に記憶されている値を増減量データレジスタ(SGIDR)の値だけ、増減設定ビット(SGCR:GID)の設定にしたがって増減します。

＜注意事項＞
このレジスタが指定したパルス数は、「レジスタ値+1」であることに注意してください。

32.4.7 増減量データレジスタ ：SGIDR (SG Increment Decrement Register)

増減量データレジスタ (SGIDR) のビット構成について説明します。

増減量データレジスタ(SGIDR)は、振幅データレジスタ(SGAR)に対する増減量を記憶します。周期レジスタが指定したトグルフリップフロップからのトーンパルス数をデクリメントカウンタがカウントするごとに、振幅データレジスタ(SGAR)に記憶されている値を、増減設定ビット(SGCR:GID)の設定にしたがって増減します。

■ SGIDR： アドレス Base_addr + 09H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] D[7:0] (Data)： **増減量データビット**
振幅データレジスタ(SGAR)に対する増減量を記憶します。ソフトウェアは、振幅データレジスタ(SGAR)に対する増減量を設定します。ハードウェアは、周期レジスタが指定したトグルフリップフロップからのトーンパルス数をデクリメントカウンタがカウントするごとに、振幅データレジスタ(SGAR)に記憶されている値を、増減設定ビット(SGCR:GID)の設定にしたがって増減します。 増減設定ビットが減少設定(SGCR:GID="0")のとき、振幅データレジスタ(SGAR)値が"0x0000"になると、それ以上のデクリメントはしません。増減設定が増加設定(SGCR:GID=1)のとき､振幅データレジスタ(SGAR)値が"0xFFFF"になると､それ以上のインクリメントはしません。また、増減量データレジスタ値が"0x00"のときは、増減を行いません。

32.4.8 PWM サイクル数データレジスタ ：SGPCR (SG PWM Cycle Register)

PWM サイクル数データレジスタ (SGPCR) のビット構成について説明します。

PWM サイクル数データレジスタ(SGPCR)は、1PWM サイクルのサイクル数を記憶します。

■ SGPCR： アドレス Base_addr + 0A$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	D[15:8]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D[7:0]							
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit0] D[15:0] (Data) : PWM サイクル数データビット
1PWM サイクルのサイクル数を記憶します。基準となるクロックサイクルは入力クロック(プリスケーラ)です。

＜注意事項＞
このレジスタが指定したクロックサイクル数は、「レジスタ値+1」であることに注意してください。

32.4.9 DMA 転送間接レジスタ ：SGDMAR (SG DMA Register)

DMA 転送間接レジスタ (SGDMAR) のビット構成について説明します。

DMA 転送間接レジスタ(SGDMAR)は、振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)、周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)に対する DMA 転送に使用する間接レジスタです。リード値は常に"0"です。本レジスタにライトした値をリードしたい場合は、振幅データレジスタ(SGAR)から PWM サイクル数データレジスタ(SGPCR)の該当するレジスタをリードしてください。

本レジスタには、DMA 転送更新許可レジスタ(SGDER)の設定にしたがって、1/2/4 バイトでアクセスしてください。アクセス位置は、1 バイトアクセスのときは bit31-bit24、2 バイトアクセスのときは bit31-bit16 固定となります。

サウンドジェネレータは、DMA 転送開始割込みセット許可(SGCR:DMA=1)時に動作許可(SGCR:ST="1")に設定された場合、周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)で指定される回数だけサウンドを出力した場合に、割込みビット(SGCR:INT)をセットして割込み信号(PIRQ)をアサートします。

DMA コントローラは、サウンドジェネレータからの割込みを受信したときに、本レジスタに対して DMA 転送を行います。なお、DMA コントローラでは、転送先アドレスを本レジスタ固定としてください。

サウンドジェネレータは、DMA 転送更新許可レジスタ(SGDER)の設定にしたがって、本レジスタにライトされるデータを、振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ、周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、および PWM サイクル数データレジスタ(SGPCR)にライトします。

■ SGDMAR： アドレス Base_addr + 0C$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit31	bit30	・	・	・	bit2	bit1	bit0
	D[31:0]							
初期値	0	0	・	・	・	0	0	0
属性	R0/W	R0/W	・	・	・	R0/W	R0/W	R0/W

[bit31～bit0] D[31:0] (Data) : DMA 転送データビット
振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)、周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)に対する DMA 転送に使用するレジスタです。

DMA 転送サイズ、転送回数は、DMA 転送更新許可レジスタ(SGDER)の設定にしたがって行ってください。 1 バイトアクセスのときは bit31-bit24、2 バイトアクセスのときは bit31-bit16 固定となります。

なお、転送回数は 1 回または 2 回のいずれかで、1 回の転送で本レジスタに転送するデータは、「振幅データ、周波数データ、トーン出力数」または「周期、増減量データ、PWM サイクル数データ」のすべて、または一部としてください。振幅データレジスタ(SGAR)から PWM サイクル数データレジスタ(SGPCR)でアドレスが 4 バイト境界を超えるような組み合わせで 1 回の転送を行うことはできません。(例: 周波数データと増減量データを 1 回転送できません。2 回転送となります。)

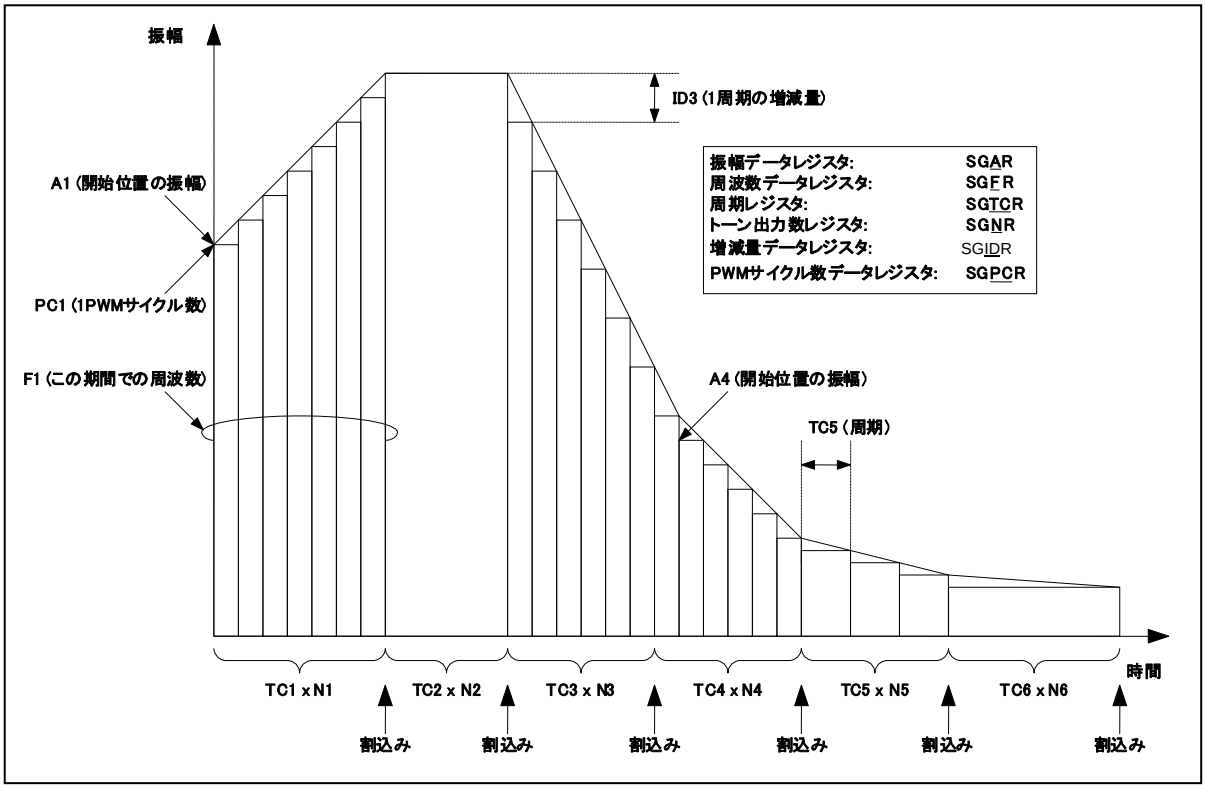
＜注意事項＞
DMA 転送更新許可レジスタ (SGDER) の設定による転送回数以上の DMA 転送を行った場合、値が更新されます。

32.5 動作説明

サウンドジェネレータの動作について説明します。

サウンドジェネレータの動作について説明します。以下に、サウンドジェネレータの動作概念図を示します。

図 32-3 サウンドジェネレータの動作概念図



出力開始位置の振幅を振幅データレジスタ(SGAR)に、トーンパルス信号の周波数を周波数データレジスタ(SGFR)に、1 周期あたりのトーンパルス信号の出力数を周期レジスタ(SGTCR)に、割込みを発生させる周期数をトーン出力数レジスタ(SGNR)に、1 周期あたりの増減量を増減量データレジスタ(SGIDR)に、1PWM サイクル数を PWM サイクル数データレジスタ(SGPCR)に、その他サウンドジェネレータ制御情報をサウンドコントロールレジスタ(SGCR)に設定します。

サウンドジェネレータは、これらの設定により、トーンパルス信号と振幅データを出力します。

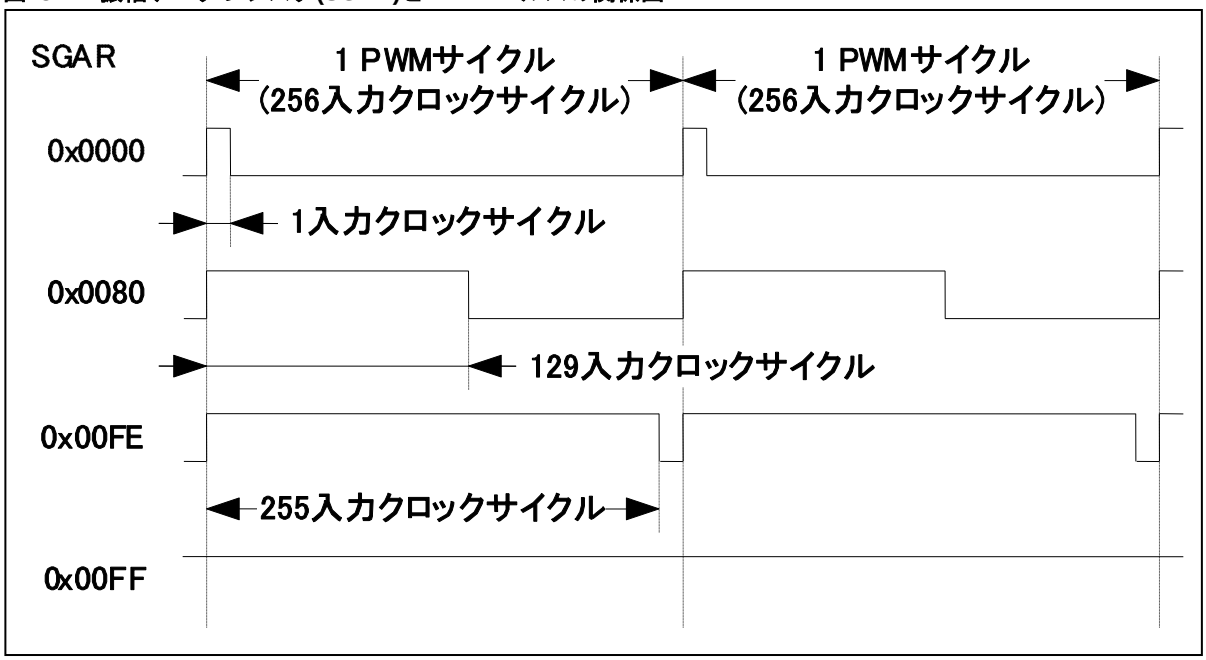
上記、動作概念図では、6 種類の信号を出力しています。振幅データレジスタをはじめ各種レジスタ値は、出力開始時、および割込み発生ごとに設定されています。

32.5.1 振幅データレジスタ (SGAR) と PWM パルスの関係

振幅データレジスタ (SGAR)と PWM パルスの関係について説明します。

振幅データレジスタ(SGAR)と PWM パルスの関係について説明します。以下に、振幅データレジスタ(SGAR)と PWM パルスの関係図を示します。

図 32-4 振幅データレジスタ(SGAR)と PWM パルスの関係図



振幅データは、SGA 端子により PWM (Pulse Width Modulation)パルスとして出力されます。

1PWM サイクルは、PWM サイクル数データレジスタ(SGPCR)で設定可能で、256 入力クロックサイクル(SGPCR=0x00FF)としています。

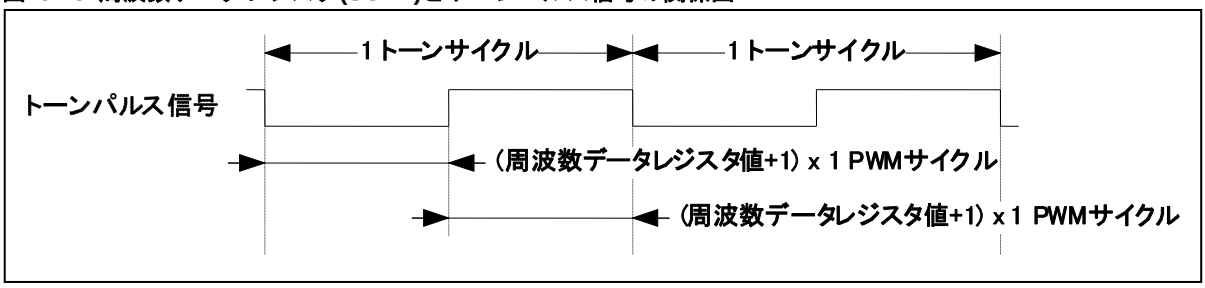
振幅データレジスタ(SGAR)の「レジスタ値+1」の値が、1PWM サイクル中に SGA 端子="H"となる入力クロックサイクル数です。なお、振幅データレジスタ(SGAR)≧PWM サイクル数データレジスタ(SGPCR)の場合、SGA 端子は常に"H"になります。

32.5.2 周波数データレジスタ (SGFR) とトーンパルス信号の関係

周波数データレジスタ(SGFR)とトーンパルス信号の関係について説明します。

周波数データレジスタ(SGFR)とトーンパルス信号の関係について説明します。以下に、周波数データレジスタ(SGFR)とトーンパルス信号の関係図を示します。

図 32-5 周波数データレジスタ(SGFR)とトーンパルス信号の関係図



トーンパルス信号は、「(周波数データレジスタ値+1) × 1PWM サイクル」のサイクルで、"L"と"H"を繰り返します。これはトグルフリップフロップにより生成されます。

トーンパルス信号は、サウンドコントロールレジスタ(SGCR)のトーン出力ビット(TONE)が"0"のとき、PWM パルスと混合(論理積)され、SGO 端子から出力されます。また、サウンドコントロールレジスタ(SGCR)のトーン出力ビット(TONE)が"1"のとき、トーンパルス信号がそのまま SGO 端子から出力されます。

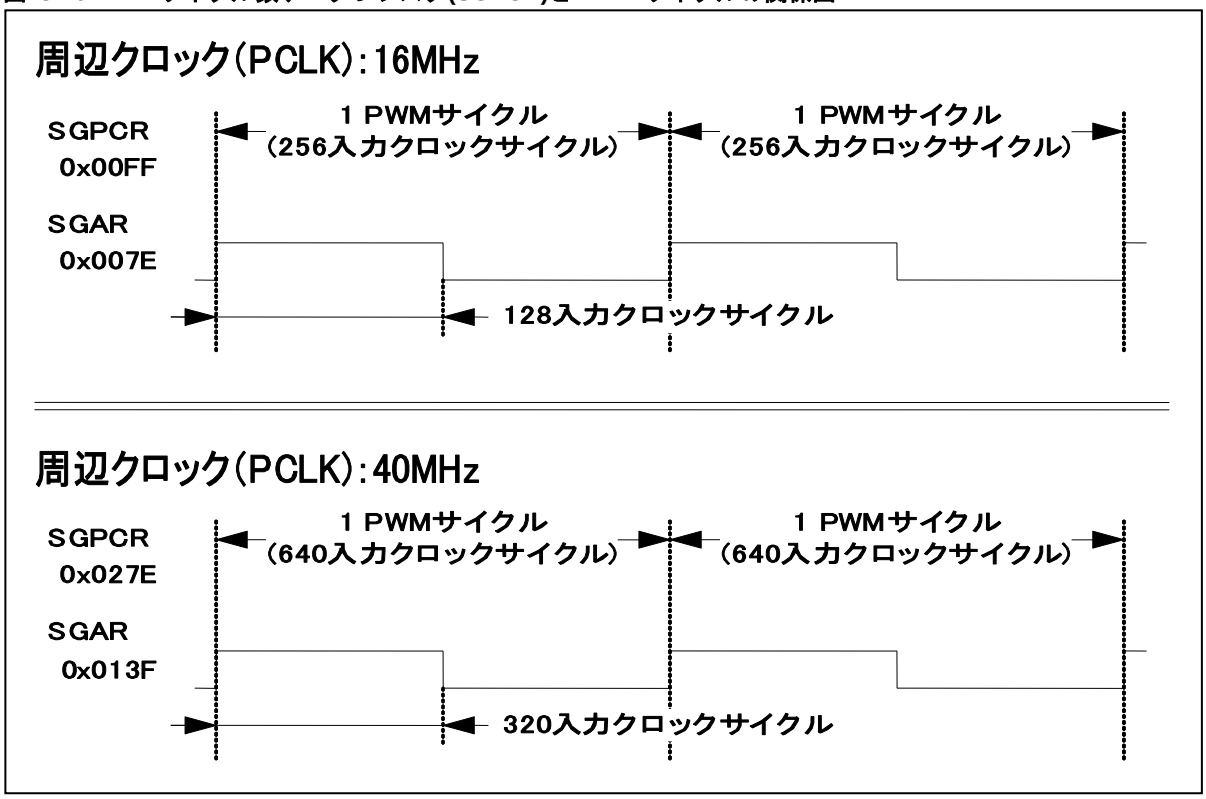
＜注意事項＞
サウンドジェネレータ動作中にレジスタ値を変更すると、変更のタイミングによって、デューティサイクルの50%が変更されることがありますので、注意してください。

32.5.3 PWM サイクル数データレジスタ (SGPCR) と PWM サイクルの関係

PWM サイクル数データレジスタ (SGPCR)と PWM サイクルの関係について説明します。

PWM サイクル数データレジスタ(SGPCR)と PWM サイクルの関係について説明します。以下に、PWM サイクル数データレジスタ(SGPCR)と PWM サイクルの関係図を示します。

図 32-6 PWM サイクル数データレジスタ(SGPCR)と PWM サイクルの関係図



PWM サイクルは、PWM サイクル数データレジスタ(SGPCR)で設定可能です。

PWM サイクル数データレジスタ(SGPCR)の「レジスタ値+1」の値が、1PWM サイクルの入力クロックサイクル数です。入力クロックは、周辺クロック(PCLK)を分周したものです。

PWM サイクルは、トーンパルス信号(またはトーンパルス信号と PWM パルス信号の混合信号)、および PWM パルス信号の基準となるサイクルです。

周辺クロック(PCLK)が 16MHz のときのサウンド出力と、周辺クロック(PCLK)が 40MHz のときのサウンド出力を同じにしたい場合は、PWM サイクル数データレジスタ(SGPCR)と振幅データレジスタ(SGAR)の値を変更することにより可能です。周辺クロック(PCLK)が 16MHz と 40MHz の比は 1：2.5 のため、上記レジスタ値の比を 1:2.5 とすることにより実現可能です。

32.5.4 DMA 転送更新許可レジスタ (SGDER) と DMA 転送回数 /DMA 転送サイズ/転送バイト位置の関係について

DMA 転送更新許可レジスタ(SGDER)と DMA 転送回数/DMA 転送サイズ/転送バイト位置の関係について説明します。

DMA 転送更新許可レジスタ(SGDER)と DMA 転送回数/DMA 転送サイズ/転送バイト位置の関係について説明します。

32.5.4.1 DMA 転送回数について

DMA 転送回数について説明します。

nバイト×1回のDMA転送をするか、nバイト×2回のDMA転送をするかは、DMA転送更新許可レジスタ(SGDER)の設定により決まります。

SGDER:ARE1、SGER:ARE0、SGDER:FRE、SGDER:NRE のすべてが"0"の場合、または、SGDER:TCRE、SGDER:IDRE、SGDER:PCRE1、SGDER:PCRE0 のすべてが"0"の場合に、DMA 転送回数は 1 回となります。それ以外の場合は、DMA 転送回数は 2 回となります。

32.5.4.2 DMA 転送サイズについて

DMA 転送サイズについて説明します。

DMA 転送サイズが、1 バイトか、2 バイトか、4 バイトかは、DMA 転送更新許可レジスタ(SGDER)の設定により決まります。また、SGDER:ARE1、SGER:ARE0、SGDER:FRE、SGDER:NRE の設定値と SGDER:TCRE、SGDER:IDRE、SGDER:PCRE1、SGDER:PCRE0 設定値の転送サイズが大きいほうが、DMA 転送サイズとなります。なお、3 バイト以上の転送は、4 バイトとみなします。

32.5.4.3 DMA 転送間接レジスタへの転送バイト位置について

DMA 転送間接レジスタへの転送バイト位置について説明します。

DMA 転送間接レジスタ(SGDMAR)への転送バイト位置は、DMA 転送更新許可レジスタ(SGDER)の設定、および DMA 転送サイズにより決定します。

DMA 転送間接レジスタ(SGDMAR)への 1 回の DMA 転送サイズが 4 バイト未満の場合、転送バイト位置は左詰めとなります。(1 回の DMA 転送サイズが 3 バイトとなるときも左詰めとなります。)

DMA 転送更新許可レジスタ(SGDER)と DMA 転送間接レジスタ(SGDMAR)への振幅データ(SGAR[15:0])、周波数データ(SGFR[7:0])、トーン出力数(SGNR[7:0])の転送バイト位置の関係を以下に示します。DMA 転送更新許可レジスタ(SGDER)の SGDER:ARE1、SGDER:ARE0、SGDER:FRE、SGDER:NRE から算出される DMA 転送間接レジスタ(SGDMAR)への転送サイズ#1 が 2 バイト以下のとき、転送バイト位置は左詰めとなります。

表 32-3 DMA 転送更新許可レジスタ (SGDER) と SGDMAR への転送バイト位置について #1

	SGDER 設定				転送サイズ#1 *1	SGDMAR への転送バイト位置			
	ARE1	ARE0	FRE	NRE		SGAR [15:8]	SGAR [7:0]	SGFR [7:0]	SGNR [7:0]
1	0	0	0	0	0	-	-	-	-
2	1	0	0	0	1	SGDMAR [31:24]	-	-	-
3	0	1	0	0	1	-	SGDMAR [31:24]	-	-
4	1	1	0	0	2	SGDMAR [31:24]	SGDMAR [23:16]	-	-
5	0	0	1	0	1	-	-	SGDMAR [31:24]	-
6	1	0	1	0	2	SGDMAR [31:24]	-	SGDMAR [23:16]	-
7	0	1	1	0	2	-	SGDMAR [31:24]	SGDMAR [23:16]	-
8	1	1	1	0	4	SGDMAR [31:24]	SGDMAR [23:16]	SGDMAR [15:8]	-
9	0	0	0	1	1	-	-	-	SGDMAR [31:24]
10	1	0	0	1	2	SGDMAR [31:24]	-	-	SGDMAR [23:16]
11	0	1	0	1	2	-	SGDMAR [31:24]	-	SGDMAR [23:16]

	SGDER 設定				転送サイズ#1 *1	SGDMAR への転送バイト位置			
	ARE1	ARE0	FRE	NRE		SGAR [15:8]	SGAR [7:0]	SGFR [7:0]	SGNR [7:0]
12	1	1	0	1	4	SGDMAR [31:24]	SGDMAR [23:16]	-	SGDMAR [15:8]
13	0	0	1	1	2	-	-	SGDMAR [31:24]	SGDMAR [23:16]
14	1	0	1	1	4	SGDMAR [31:24]	-	SGDMAR [23:16]	SGDMAR [15:8]
15	0	1	1	1	4	-	SGDMAR [31:24]	SGDMAR [23:16]	SGDMAR [15:8]
16	1	1	1	1	4	SGDMAR [31:24]	SGDMAR [23:16]	SGDMAR [15:8]	SGDMAR [7:0]

*1: {SGDER:ARE1, SGER:ARE0, SGDER:FRE, SGDER:NRE}から算出される転送サイズ
X, - : Don't Care

DMA 転送更新許可レジスタ(SGDER)と DMA 転送間接レジスタ(SGDMAR)への周期(SGTCR[7:0])、増減量データ(SGFR[7:0])、PWM サイクル数データ(SGPCR[15:0])の転送バイト位置の関係を以下に示します。DMA 転送更新許可レジスタ(SGDER)と SGDER:TCRE、SGDER:IDRE、SGDER:PCRE1、SGDER:PCRE0 から算出される DMA 転送間接レジスタ(SGDMAR)への転送サイズ#2 が 2 バイト以下のとき、転送バイト位置は左詰めとなります。

表 32-4 DMA 転送更新許可レジスタ(SGDER)と SGDMAR への転送バイト位置について #2

No.	SGDER 設定				転送サイズ#2 *1	SGDMAR への転送バイト位置			
	TCRE	IDRE	PCRE1	PCRE0		SGTCR [7:08]	SGIDR [7:0]	SGPCR [15:8]	SGPCR [7:0]
1	0	0	0	0	0	-	-	-	-
2	1	0	0	0	1	SGDMAR [31:24]	-	-	-
3	0	1	0	0	1	-	SGDMAR [31:24]	-	-
4	1	1	0	0	2	SGDMAR [31:24]	SGDMAR [23:16]	-	-
5	0	0	1	0	1	-	-	SGDMAR [31:24]	-
6	1	0	1	0	2	SGDMAR [31:24]	-	SGDMAR [23:16]	-
7	0	1	1	0	2	-	SGDMAR [31:24]	SGDMAR [23:16]	-
8	1	1	1	0	4	SGDMAR [31:24]	SGDMAR [23:16]	SGDMAR [15:8]	-
9	0	0	0	1	1	-	-	-	SGDMAR [31:24]
10	1	0	0	1	2	SGDMAR [31:24]	-	-	SGDMAR [23:16]
11	0	1	0	1	2	-	SGDMAR [31:24]	-	SGDMAR [23:16]
12	1	1	0	1	4	SGDMAR [31:24]	SGDMAR [23:16]	-	SGDMAR [15:8]
13	0	0	1	1	2	-	-	SGDMAR [31:24]	SGDMAR [23:16]
14	1	0	1	1	4	SGDMAR [31:24]	-	SGDMAR [23:16]	SGDMAR [15:8]
15	0	1	1	1	4	-	SGDMAR [31:24]	SGDMAR [23:16]	SGDMAR [15:8]
16	1	1	1	1	4	SGDMAR [31:24]	SGDMAR [23:16]	SGDMAR [15:8]	SGDMAR [7:0]

*1: {SGDER:TCRE, SGDER:IDRE, SGDER:PCRE1, SGDER:PCRE0}から算出される転送サイズ
X: Don't Care

32.5.4.4 DMA 転送イメージ

DMA 転送イメージについて説明します。

DMA 転送更新許可レジスタ(SGDER)の設定が、

{SGDER:ARE1, SGER:ARE0, SGDER:FRE, SGDER:NRE} = 1001
{SGDER:TCRE, SGDER:IDRE, SGDER:PCRE1, SGDER:PCRE0} = 0100

の場合の例を以下に示します。

DMA 転送回数 : 2 回

DMA 転送サイズ : 2 バイト

DMA 転送間接レジスタへの転送バイト位置 :

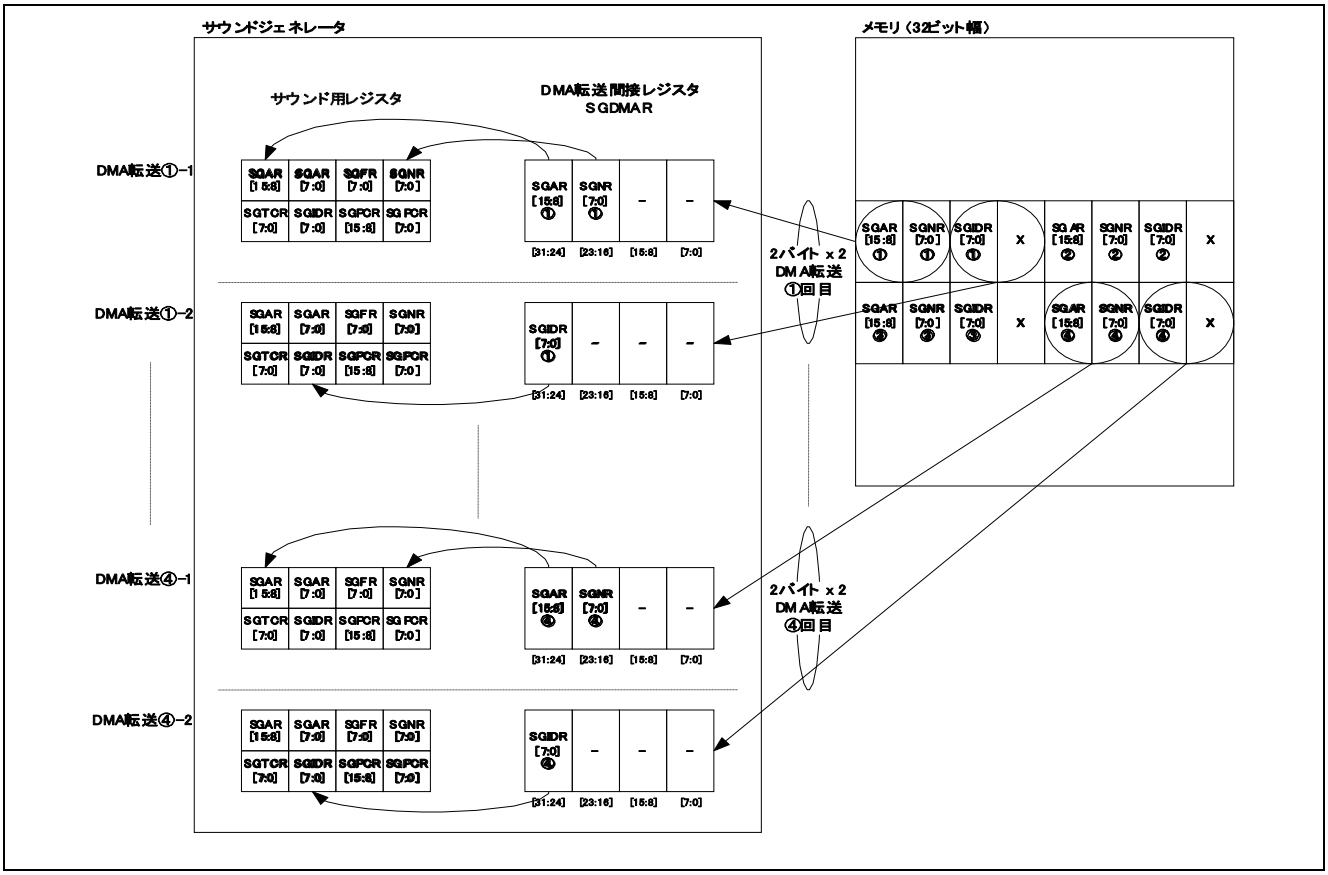
1 回目

SGDMAR[31:24] ← 振幅データ(上位バイト) / SGAR[15:8]
SGDMAR[23:16] ← トーン出力数 / SGNR[7:0]
SGDMAR[15:0] ← Don't care

2 回目

SGDMAR[31:24] ← 増減量データ / SGIDR[7:0]
SGDMAR[23:0] ← Don't care

図 32-7 DMA 転送イメージ

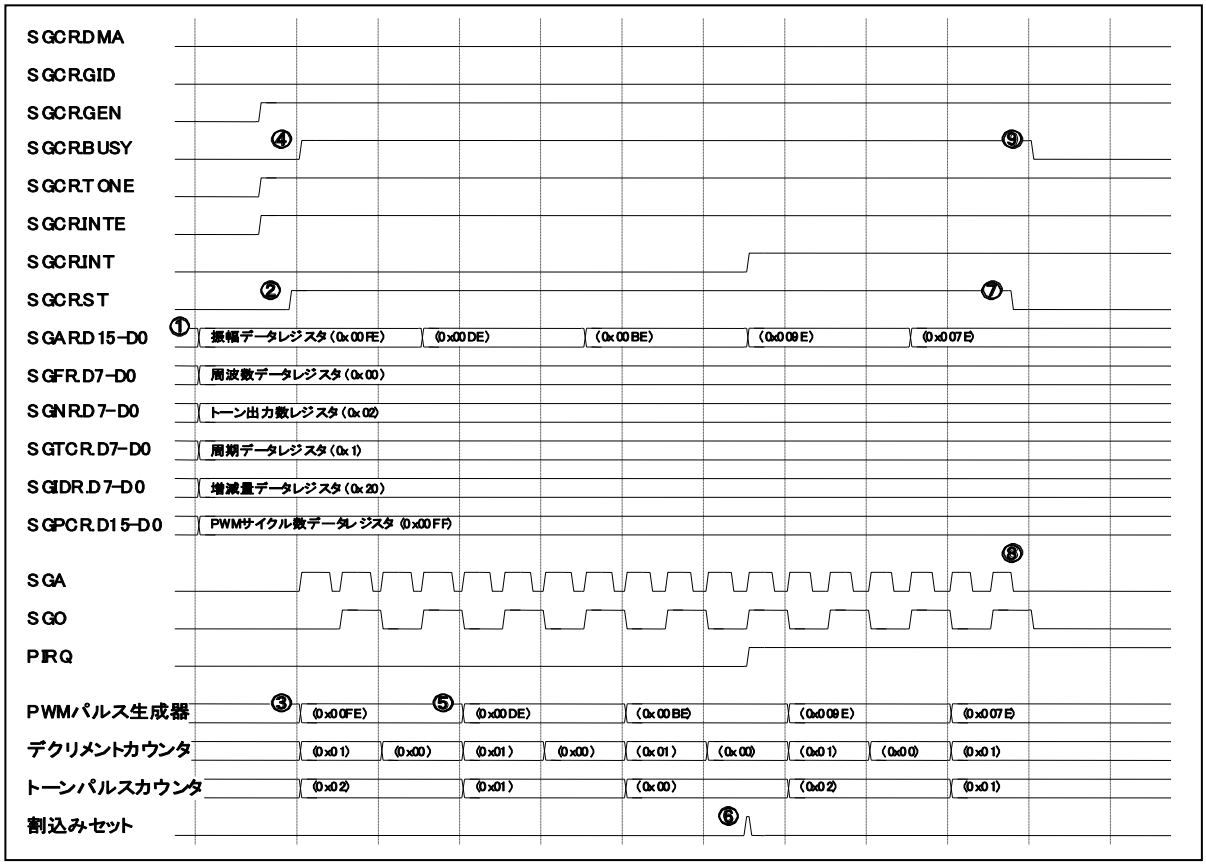


32.5.5 サウンドジェネレータの動作

サウンドジェネレータの動作について説明します。

サウンドジェネレータの動作について説明します。以下に、サウンドジェネレータの動作を示します。

図 32-8 サウンドジェネレータの動作



① ソフトウェアでリロード値を振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)、周期レジスタ(SGTCR)に、振幅の増減量を増減量データレジスタ(SGIDR)に、1PWM サイクル数を PWM サイクル数データレジスタ(SGPCR)に書込みます。また、その他サウンドジェネレータ制御情報をサウンドコントロールレジスタ(SGCR)に設定します。割込みビット(SGCR:INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。

② スタートビット(SGCR:ST)を"1"に設定します。

③ スタートビット(SGCR:ST)が"1"に設定されたことにより、振幅データレジスタ(SGAR)の値を PWM 生成器に、周波数データレジスタ(SGFR)の値を周波数カウンタに、トーン出力数レジスタ(SGNR)の値をトーンパルスカウンタに、周期レジスタ(SGTCR)の値をデクリメントカウンタにロードします。

④ 動作フラグ(SGCR:BUSY)が"1"になります。

⑤ デクリメントカウンタがトーンパルス数をリロード値までカウントすることにより、振幅データレジスタ(SGAR)の値が、自動増減許可ビット(SGCR:GEN)および増減設定ビット(SGCR:GID)にしたがって増減します。

⑥ トーン出力数レジスタ(SGNR)および周期レジスタ(SGTCR)により指定されたトーンパルスの個数がトーンパルスカウンタによりカウントされたとき(トーンパルスカウンタ="0x00"かつデクリメントカウンタ="0x00"かつ SGO="L"→"H"となるタイミングのとき)、割込みセット要求が発生して、割込みビット(SGCR:INT)がセットされ、割込み(PIRQ)が発生します。

⑦ スタートビット(SGCR:ST)を"0"に設定します。ビジービット(SGCR:BUSY)が"0"になるまでは動作を継続します。

⑧ サウンドジェネレータは、現行のトーンサイクル終了時に動作を停止します。
⑨ 動作フラグ(SGCR:BUSY)が"0"になります。

■ DMA 転送開始割込みセット許可ビットについて

DMA 転送開始割込みセット許可ビットの設定により、CPU によるスタート指示後の初回の割込みのアサート条件が異なります。

- 通常モード：サウンドジェネレータが周期レジスタ(SGTCR)に設定されたトーンパルス数を出力したとき
- DMA モード：スタート指示直後 (DMA 転送要求)

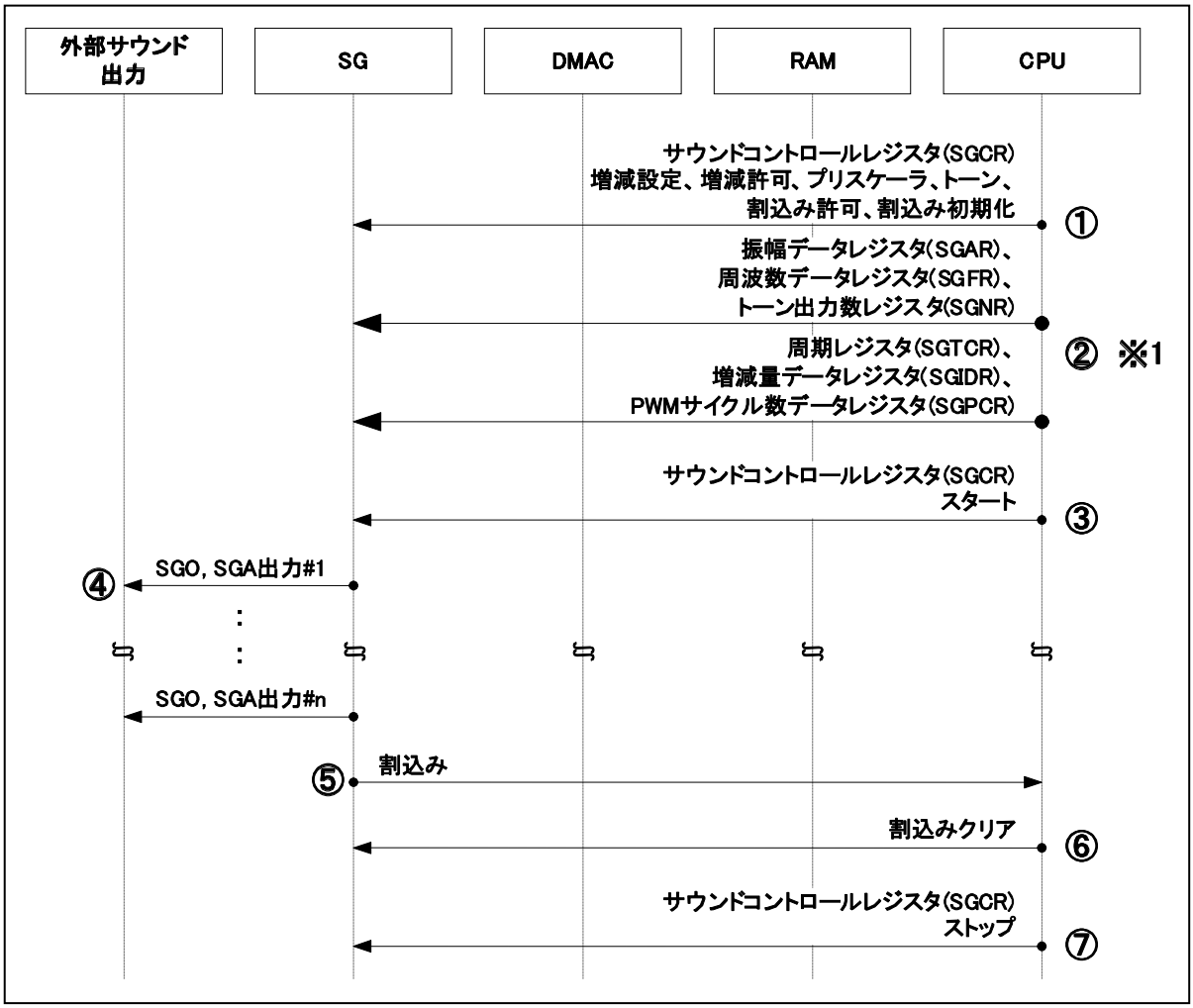
■ DMA 転送

SGAR、SGFR、SGNR、SGTCR、SGIDR、SGPCR への設定は、DMA 転送間接レジスタ(SGDMAR)を介して行います。

■ CPU によるサウンドジェネレータの単一動作

以下に、CPU によるサウンドジェネレータの単一動作フローを示します。

図 32-9 CPU によるサウンドジェネレータの単一動作



① ソフトウェアで、サウンドジェネレータのサウンドコントロールレジスタ(SGCR)に、サウンドジェネレータ制御情報を設定します。割込みビット(SGCR:INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。

② ソフトウェアで、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」に設定を行います。(*1: なお、すべてのレジスタへの設定は必須ではありません。)
③ スタートビット(SGCR:ST)を"1"に設定します。
④ サウンドジェネレータの SGO, SGA 出力が開始されます。
⑤ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑥ CPU が割込みをクリアします。
⑦ CPU からサウンドジェネレータにストップ指示(SGCR:ST=0)をします。

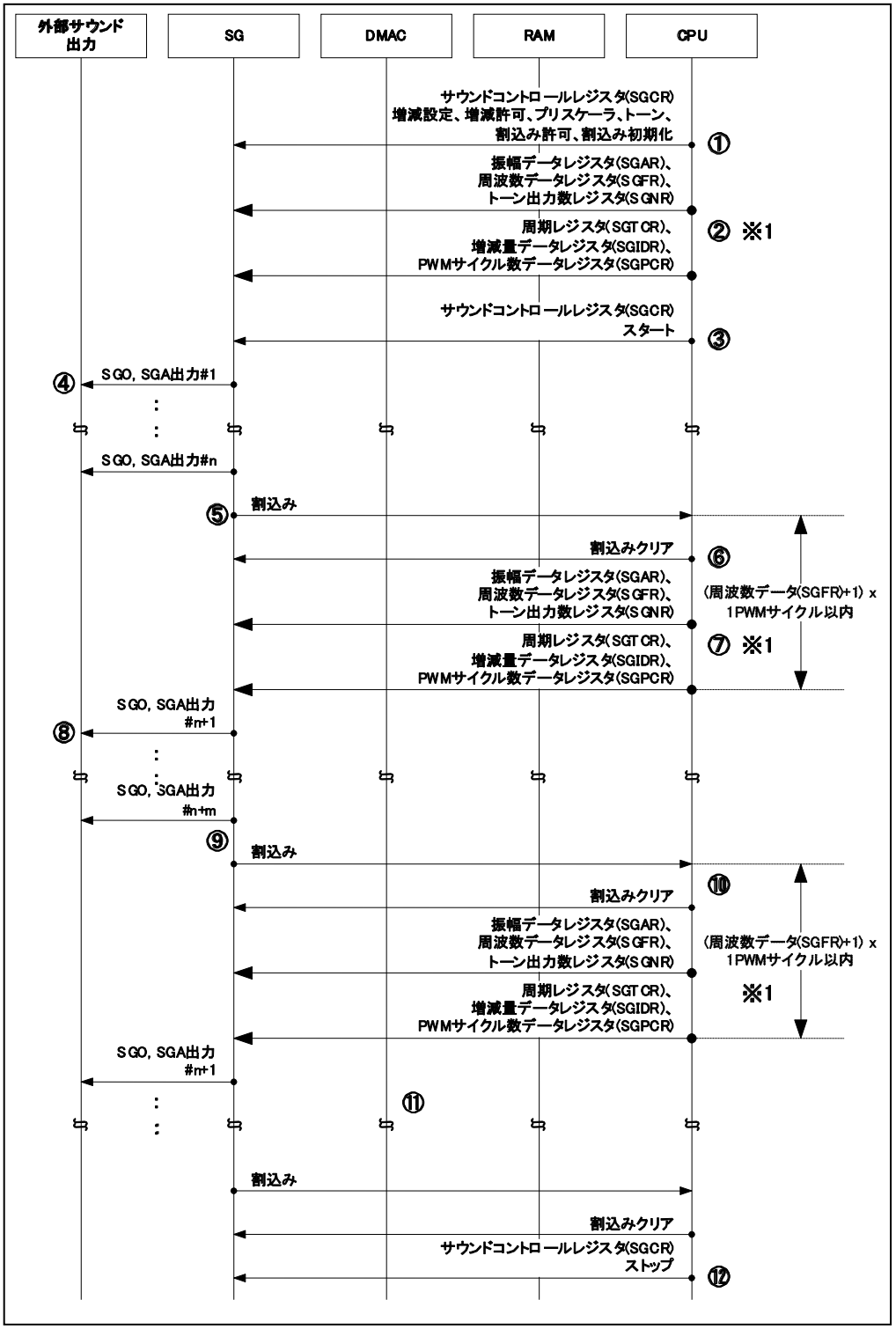
32.5.6 CPU によるサウンドジェネレータの連続動作

CPU によるサウンドジェネレータの連続動作について説明します。

以下に、CPU によるサウンドジェネレータの連続動作フローを示します。

⑦以降が、CPU によるサウンドジェネレータの単一動作フローと異なります。

図 32-10 CPU によるサウンドジェネレータの連続動作



① ソフトウェアで、サウンドジェネレータのサウンドコントロールレジスタ(SGCR)に、サウンドジェネレータ制御情報を設定します。割込みビット(SGCR:INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。

② ソフトウェアで、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」に設定を行います。(*1: なお、すべてのレジスタへの設定は必須ではありません。)

③ スタートビット(SGCR:ST)を"1"に設定します。
④ サウンドジェネレータの SGO, SGA 出力が開始されます。
⑤ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑥ CPU が割込みをクリアします。
⑦ ソフトウェアで、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」に設定を行います。(*1: なお、すべてのレジスタへの設定は必須ではありません。)
⑧ 設定値で、サウンドジェネレータの SGO, SGA 出力を行います。
⑨ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑩ CPU が割込みをクリアします。
⑪ 以降、⑦から⑪が繰り返されます。
⑫ CPU からサウンドジェネレータにストップ指示(SGCR:ST=0)をします。

＜注意事項＞
⑤から⑦までのようにサウンドジェネレータから割込みを通知し必要な設定を行うまでは、(周波数データ(SGFR)+1) × 1PWM サイクル以内で完了する必要があります。

＜注意事項＞
増減設定を変更する際は、上記サイクル内でサウンドコントロールレジスタ(SGCR)の増減設定ビット(SGCR:GID)および自動増減許可ビット(SGCR:GEN)を変更する必要があります。

32.5.7 DMA と連携したサウンドジェネレータの動作

DMA と連携したサウンドジェネレータの動作について説明します。

DMA と連携したサウンドジェネレータの動作フローを示します。

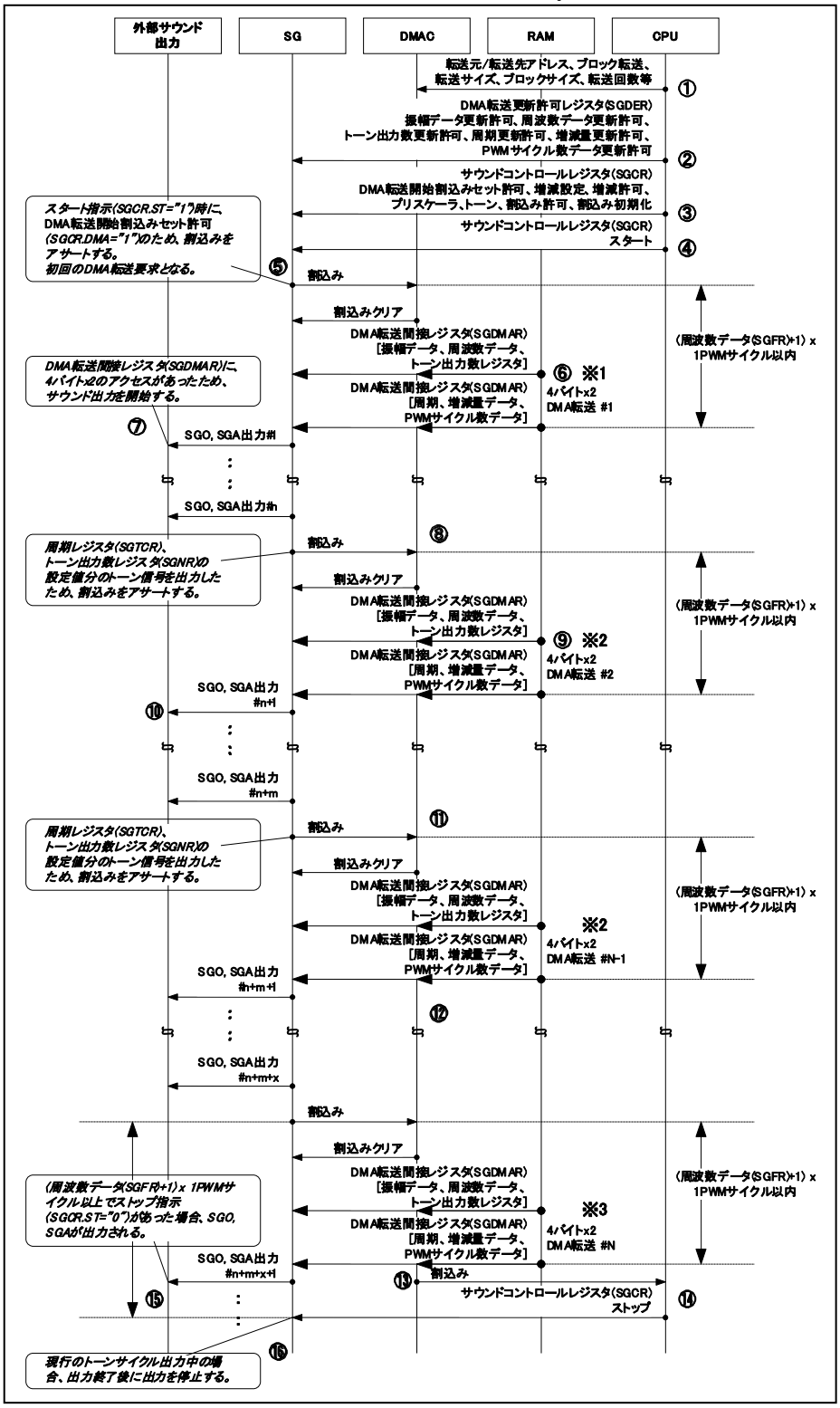
サウンド用のデータレジスタの設定は DMAC が行います。初回の割込みのアサートが、CPU によるサウンドジェネレータの動作フローと異なります。また、サウンド用のデータレジスタの転送は、DMA 転送間接レジスタ(SGDMAR)を介して行います。

＜注意事項＞
割込み信号を DMA 転送の要求とするため、ソフトウェアで割込み許可設定(SGCR:INTE=1)にする必要があります。

32.5.8　4 バイト×2 の DMA 転送を N 回行う場合

4 バイト×2 の DMA 転送を N 回行う場合について説明します。

図 32-11 DMA と連携したサウンドジェネレータの動作 (4 バイト×2 の DMA 転送を N 回行う場合)



① ソフトウェアで、DMAC に DMA 転送に必要な設定を行います。なお、DMA 転送では、4 バイト×2 のブロック転送を N 回行います。DMA 転送間接レジスタ(SGDMAR)を介して、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」のデータを転送します。なお、DMAC の転送先アドレスの設定は、アドレス固定(DMA 転送間接レジスタのアドレス)とします。
② ソフトウェアで、サウンドジェネレータの DMA 転送更新許可レジスタ(SGDER)に、DMA 転送を行うレジスタの更新許可設定を行います。ここでは、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」のすべてのレジスタを更新するように設定します。
③ ソフトウェアで、サウンドジェネレータのサウンドコントロールレジスタ(SGCR)に、サウンドジェネレータ制御情報を設定します。また、DMA 転送開始割込みセット許可ビットを許可設定にします。割込みビット(SGCR.INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。
④ スタートビット(SGCR:ST)を"1"に設定します。
⑤ スタート指示(SGCR:ST=1)時に、DMA 転送開始割込みセット許可設定(SGCR:DMA=1)のため、割込みビット(SGCR:INT)がセットされて、割込み(PIRQ)が発生します。これは DMA 転送要求として使用されます。
⑥ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して、1 回目の転送で振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)の値を、2 回目の転送で周期レジスタ(SGTCR)、増減量データレジスタ(SGDR)、PWM サイクル数データレジスタ(SGPCR)の値を転送します。(*1:DMA 転送間接レジスタへの 4 バイト×2 のブロック転送が必須です。)
⑦ DMA 転送間接レジスタ(SGDMAR)に 4 バイト×2 のブロック転送が行われたため、サウンドジェネレータの SGO, SGA 出力が開始されます。
⑧ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑨ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して DMA 転送を行います。 (*2: 2 回目以降の DMA 転送も、DMA 転送間接レジスタへの 4 バイト×2 のブロック転送が必須です。)
⑩ DMA 転送されたデータで、サウンド出力されます。
⑪ DMAC で設定回数分の DMA 転送が完了したとき、DMAC から CPU に割込みが通知されます。
⑫ 以降、同様の動作が続きます。
⑬ DMAC が設定回数分の DMA 転送(4 バイト×2 を N 回転送)を終了したので、CPU 対して割込みが発生します。
⑭ CPU からサウンドジェネレータにストップ指示(SGCR:ST=0)を行います。
⑮ (周波数データ(SGFR)+1) × 1PWM サイクル以内でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は出力されません。(*3: N 回目の転送データは、サウンド用レジスタにライトされますが、サウンド出力はされません。N 回目の転送は、DMAC から CPU に対して割込みを発生させるための転送です)
⑯ (周波数データ(SGFR)+1) × 1PWM サイクルより後でストップ指示(SGCR.ST=0)があった場合、N 回目の転送データによる SGO, SGA は現行のトーンサイクル終了後に出力が停止します。(*3: N 回目の転送データが、サウンド出力されます。)

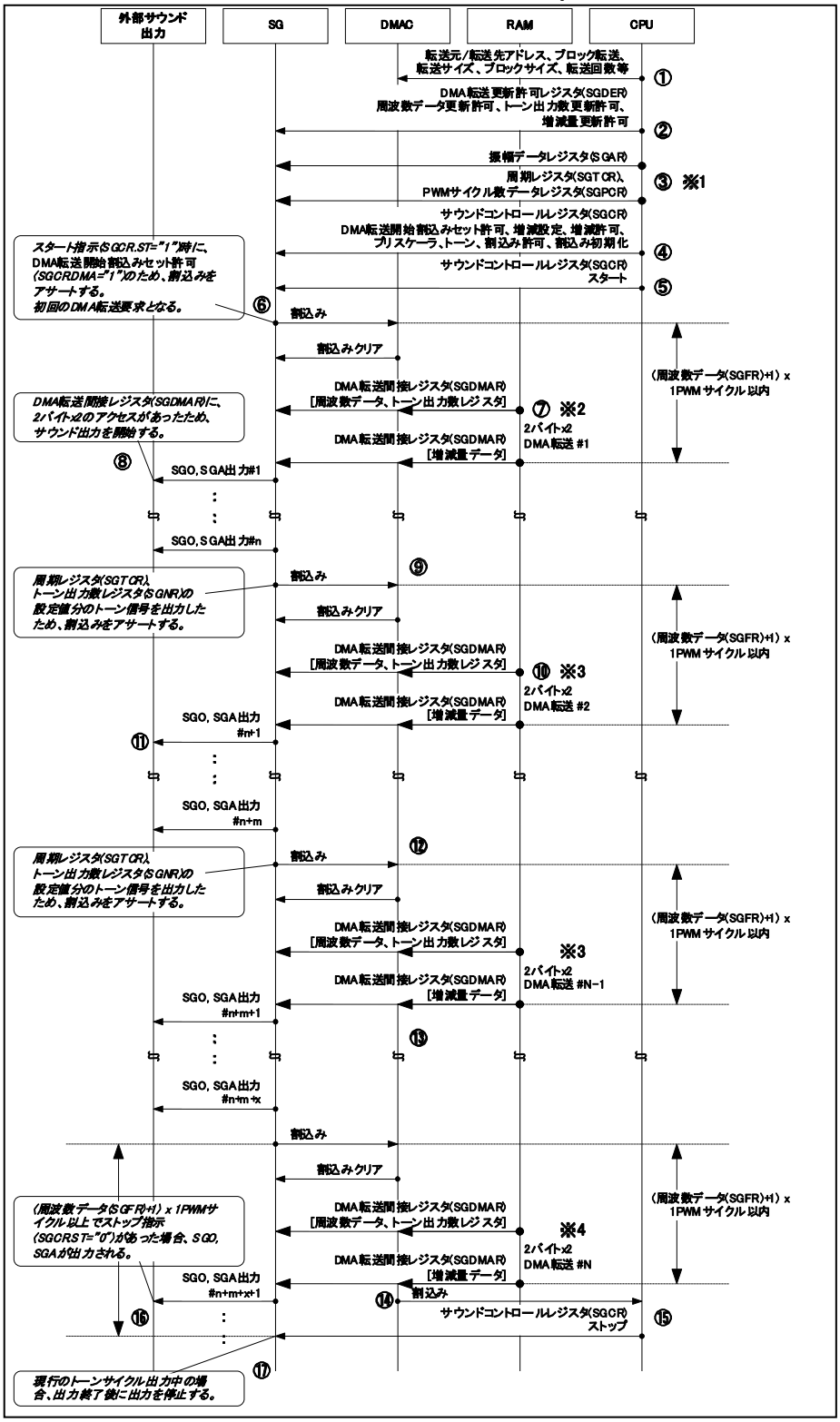
＜注意事項＞

⑤から⑥までのようにサウンドジェネレータから割込みを通知し DMA 転送が終了するまでには、(周波数データ(SGFR)+1) × 1PWM サイクル以内で完了する必要があります。

32.5.8.1 2 バイト×2 の DMA 転送を N 回行う場合

2 バイト×2 の DMA 転送を N 回行う場合について説明します。

図 32-12 DMA と連携したサウンドジェネレータの動作 (2 バイト×2 の DMA 転送を N 回行う場合)



① ソフトウェアで、DMAC に DMA 転送に必要な設定を行います。なお、DMA 転送では、2 バイト×2 のブロック転送を N 回行います。DMA 転送間接レジスタ(SGDMAR)を介して、「周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「増減量データレジスタ(SGIDR)」のデータを転送します。なお、DMAC の転送先アドレスの設定は、アドレス固定(DMA 転送間接レジスタのアドレス)とします。
② ソフトウェアで、サウンドジェネレータの DMA 転送更新許可レジスタ(SGDER)に、DMA 転送を行うレジスタの更新許可設定を行います。ここでは、「周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「増減量データレジスタ(SGIDR)」を更新するように設定します。
③ ソフトウェアで、サウンドジェネレータの「振幅データレジスタ(SGAR)」および「周期レジスタ(SGTCR)、PWM サイクル数データレジスタ(SGPCR)」に設定します。(*1: DMA 転送で更新しないレジスタを設定します。)
④ ソフトウェアで、サウンドジェネレータのサウンドコントロールレジスタ(SGCR)に、サウンドジェネレータ制御情報を設定します。また、DMA 転送開始割込みセット許可ビットを許可設定にします。割込みビット(SGCR:INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。
⑤ スタートビット(SGCR:ST)を"1"に設定します。
⑥ スタート指示(SGCR:ST=1)時に、DMA 転送開始割込みセット許可設定(SGCR:DMA=1)のため、割込みビット(SGCR:INT)がセットされて、割込み(PIRQ)が発生します。これは DMA 転送要求として使用されます。
⑦ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して、1 回目の転送で周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)の値を、2 回目の転送で増減量データレジスタ(SGDR)の値を転送します。(*2:DMA 転送間接レジスタへの 2 バイト×2 のブロック転送が必須です。)
⑧ DMA 転送間接レジスタ(SGDMAR)に 2 バイト×2 のブロック転送が行われたため、サウンドジェネレータの SGO, SGA 出力が開始されます。
⑨ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑩ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して DMA 転送を行います。(*3: 2 回目以降の DMA 転送も、DMA 転送間接レジスタへの 2 バイト×2 のブロック転送が必須です。)
⑪ DMA 転送されたデータで、サウンド出力されます。
⑫ DMAC で設定回数分の DMA 転送が完了したとき、DMAC から CPU に割込みが通知されます。
⑬ 以降、同様の動作が続きます。
⑭ DMAC が設定回数分の DMA 転送(2 バイト×2 を N 回転送)を終了したので、CPU 対して割込みが発生します。
⑮ CPU からサウンドジェネレータにストップ指示(SGCR:ST=0)を行います。
⑯ (周波数データ(SGFR)+1) × 1PWM サイクル以内でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は出力されません。(*4: N 回目の転送データは、サウンド用レジスタにライトされますが、サウンド出力はされません。N 回目の転送は、DMAC から CPU に対して割込みを発生させるための転送です)
⑰ (周波数データ(SGFR)+1)×1PWM サイクルより後でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は現行のトーンサイクル終了後に出力が停止します。(*4: N 回目の転送データが、サウンド出力されます。)

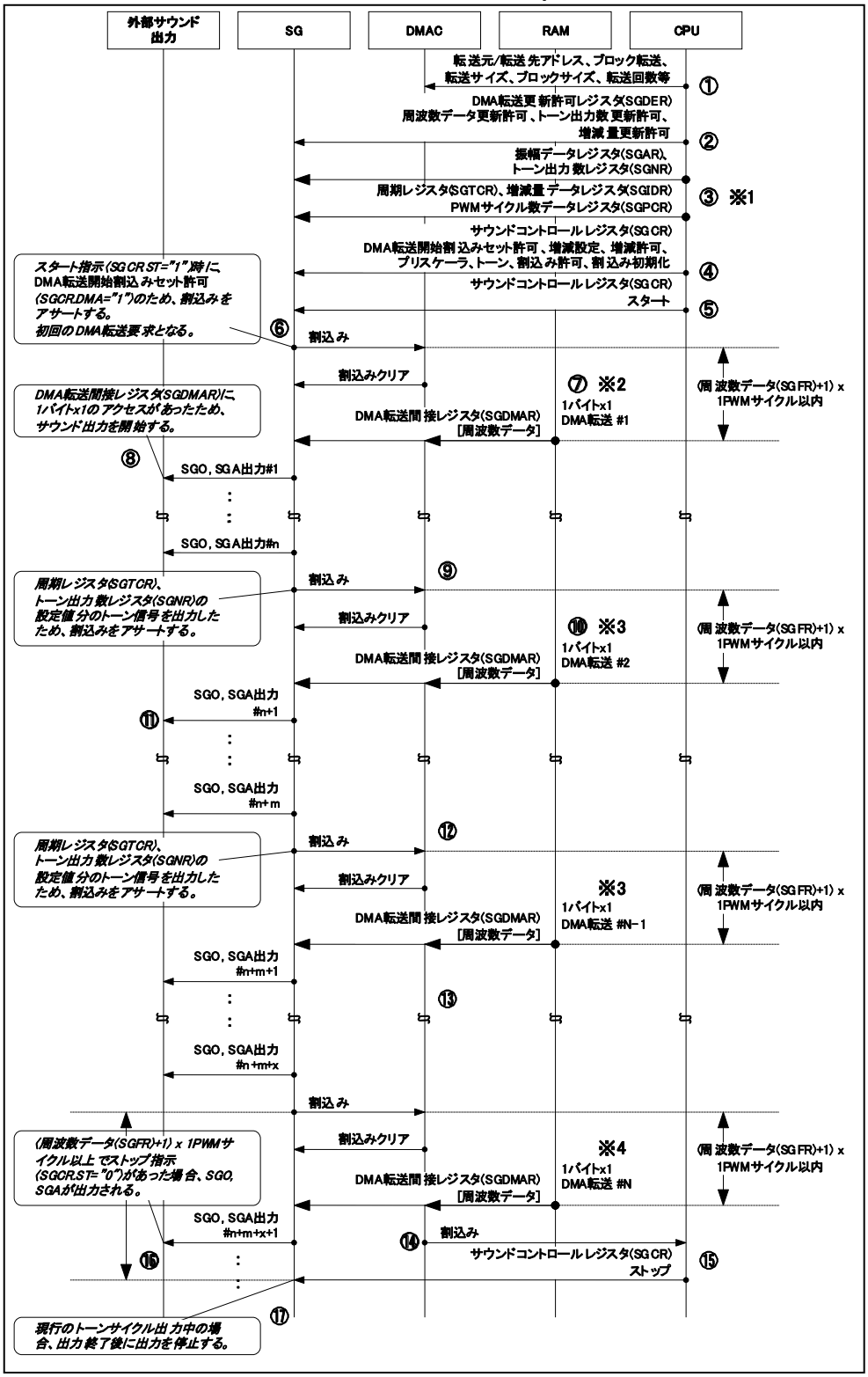
＜注意事項＞

⑥から⑦までのようにサウンドジェネレータから割込みを通知し DMA 転送が終了するまでには、(周波数データ(SGFR)+1) × 1PWM サイクル以内で完了する必要があります。

32.5.8.2 1 バイト×1 の DMA 転送を N 回行う場合

1 バイト×1 の DMA 転送を N 回行う場合について説明します。

図 32-13 DMA と連携したサウンドジェネレータの動作 (1 バイト×1 の DMA 転送を N 回行う場合)



① ソフトウェアで、DMAC に DMA 転送に必要な設定を行います。なお、DMA 転送では、1 バイト×1 のブロック転送を N 回行います。DMA 転送間接レジスタ(SGDMAR)を介して、「周波数データレジスタ(SGFR)」のデータを転送します。なお、DMAC の転送先アドレスの設定は、アドレス固定(DMA 転送間接レジスタのアドレス)とします。
② ソフトウェアで、サウンドジェネレータの DMA 転送更新許可レジスタ(SGDER)に、DMA 転送を行うレジスタの更新許可設定を行います。ここでは、「周波数データレジスタ(SGFR)」を更新するように設定します。
③ ソフトウェアで、サウンドジェネレータの「振幅データレジスタ(SGAR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」に設定します。(*1: DMA 転送で更新しないレジスタを設定します。)
④ ソフトウェアで、サウンドジェネレータのサウンドコントロールレジスタ(SGCR)に、サウンドジェネレータ制御情報を設定します。また、DMA 転送開始割込みセット許可ビットを許可設定にします。割込みビット(SGCR:INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。
⑤ スタートビット(SGCR:ST)を"1"に設定します。
⑥ スタート指示(SGCR:ST=1)時に、DMA 転送開始割込みセット許可設定(SGCR:DMA=1)のため、割込みビット(SGCR:INT)がセットされて、割込み(PIRQ)が発生します。これは DMA 転送要求として使用されます。
⑦ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して、周波数データレジスタ(SGFR)の値を転送します。(*2:DMA 転送間接レジスタへの 1 バイト×1 のブロック転送が必須です。)
⑧ DMA 転送間接レジスタ(SGDMAR)に 1 バイト×1 のブロック転送が行われたため、サウンドジェネレータの SGO, SGA 出力が開始されます。
⑨ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑩ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して DMA 転送を行います。 (*3: 2 回目以降の DMA 転送も、DMA 転送間接レジスタへの 1 バイト×1 のブロック転送が必須です。)
⑪ DMA 転送されたデータで、サウンド出力されます。
⑫ DMAC で設定回数分の DMA 転送が完了したとき、DMAC から CPU に割込みが通知されます。
⑬ 以降、同様の動作が続きます。
⑭ DMAC が設定回数分の DMA 転送(1 バイト×1 を N 回転送)を終了したので、CPU 対して割込みが発生します。
⑮ CPU からサウンドジェネレータにストップ指示(SGCR:ST=0)を行います。
⑯ (周波数データ(SGFR)+1) × 1PWM サイクル以内でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は出力されません。(*4: N 回目の転送データは、サウンド用レジスタにライトされますが、サウンド出力はされません。N 回目の転送は、DMAC から CPU に対して割込みを発生させるための転送です)
⑰ (周波数データ(SGFR)+1) × 1PWM サイクルより後でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は現行のトーンサイクル終了後に出力が停止します。(*4: N 回目の転送データが、サウンド出力されます。)

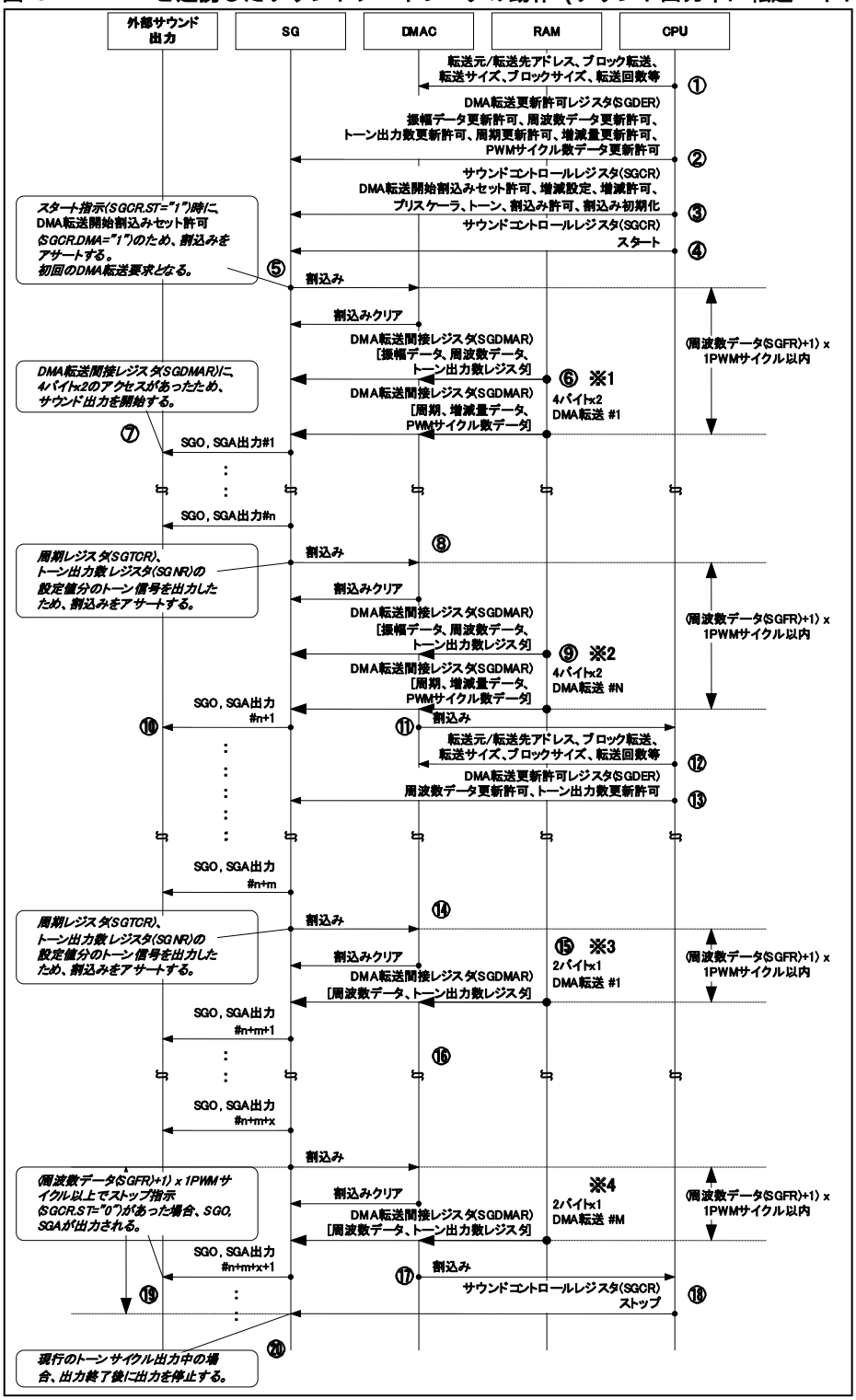
＜注意事項＞

⑥から⑦までのようにサウンドジェネレータから割込みを通知し DMA 転送が終了するまでには、(周波数データ(SGFR)+1) × 1PWM サイクル以内で完了する必要があります。

32.5.8.3 4 バイト×2 の DMA 転送を N 回と 2 バイト×1 の DMA 転送を M 回行う場合 (サウンド出力中に転送バイト数変更)

4 バイト×2 の DMA 転送を N 回と 2 バイト×1 の DMA 転送を M 回行う場合について示します。

図 32-14 DMA と連携したサウンドジェネレータの動作 (サウンド出力中に転送バイト数変更)



① ソフトウェアで、DMAC に DMA 転送に必要な設定を行います。なお、DMA 転送では、4 バイト×2 のブロック転送を N 回行います。DMA 転送間接レジスタ(SGDMAR)を介して、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」のデータを転送します。なお、DMAC の転送先アドレスの設定は、アドレス固定(DMA 転送間接レジスタのアドレス)とします。
② ソフトウェアで、サウンドジェネレータの DMA 転送更新許可レジスタ(SGDER)に、DMA 転送を行うレジスタの更新許可設定を行います。ここでは、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」のすべてのレジスタを更新するように設定します。
③ ソフトウェアで、サウンドジェネレータのサウンドコントロールレジスタ(SGCR)に、サウンドジェネレータ制御情報を設定します。また、DMA 転送開始割込みセット許可ビットを許可設定にします。割込みビット(SGCR:INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。
④ スタートビット(SGCR:ST)を"1"に設定します。
⑤ スタート指示(SGCR:ST=1)時に、DMA 転送開始割込みセット許可設定(SGCR:DMA=1)のため、割込みビット(SGCR:INT)がセットされて、割込み(PIRQ)が発生します。これは DMA 転送要求として使用されます。
⑥ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して、1 回目の転送で振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)の値を、2 回目の転送で周期レジスタ(SGTCR)、増減量データレジスタ(SGDR)、PWM サイクル数データレジスタ(SGPCR)の値を転送します。(*1:DMA 転送間接レジスタへの 4 バイト×2 のブロック転送が必須です。)
⑦ DMA 転送間接レジスタ(SGDMAR)に 4 バイト×2 のブロック転送が行われたため、サウンドジェネレータの SGO, SGA 出力が開始されます。
⑧ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑨ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して DMA 転送を行います。 (*2: 2 回目以降の DMA 転送も、DMA 転送間接レジスタへの 4 バイト×2 のブロック転送が必須です。)
⑩ DMA 転送されたデータで、サウンド出力されます。
⑪ DMAC で設定回数分の DMA 転送(4 バイト×2 を N 回転送)が完了したとき、DMAC から CPU に割込みが通知されます。
⑫ ソフトウェアで、DMAC に DMA 転送に必要な設定を行います。なお、DMA 転送では、2 バイト×1 のブロック転送を M 回行います。DMA 転送間接レジスタ(SGDMAR)を介して、「周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」のデータを転送します。
⑬ ソフトウェアで、サウンドジェネレータの DMA 転送更新許可レジスタ(SGDER)に、DMA 転送を行うレジスタの更新許可設定を行います。ここでは、「周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」を更新するように設定します。
⑭ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑮ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して DMA 転送を行います。(*3: DMA 転送間接レジスタへの 2 バイト×1 のブロック転送が必須です。)
⑯ 以降、同様の動作が続きます。
⑰ DMAC が設定回数分の DMA 転送(2 バイト×1 を M 回転送)を終了したので、CPU 対して割込みが発生します。
⑱ CPU からサウンドジェネレータにストップ指示(SGCR:ST=0)を行います。
⑲ (周波数データ(SGFR)+1) × 1PWM サイクル以内でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は出力されません。(*4: N 回目の転送データは、サウンド用レジスタにライトされますが、サウンド出力はされません。N 回目の転送は、DMAC から CPU に対して割込みを発生させるための転送です)
⑳ (周波数データ(SGFR)+1) × 1PWM サイクルより後でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は現行のトーンサイクル終了後に出力が停止します。(*4: N 回目の転送データが、サウンド出力されます。)

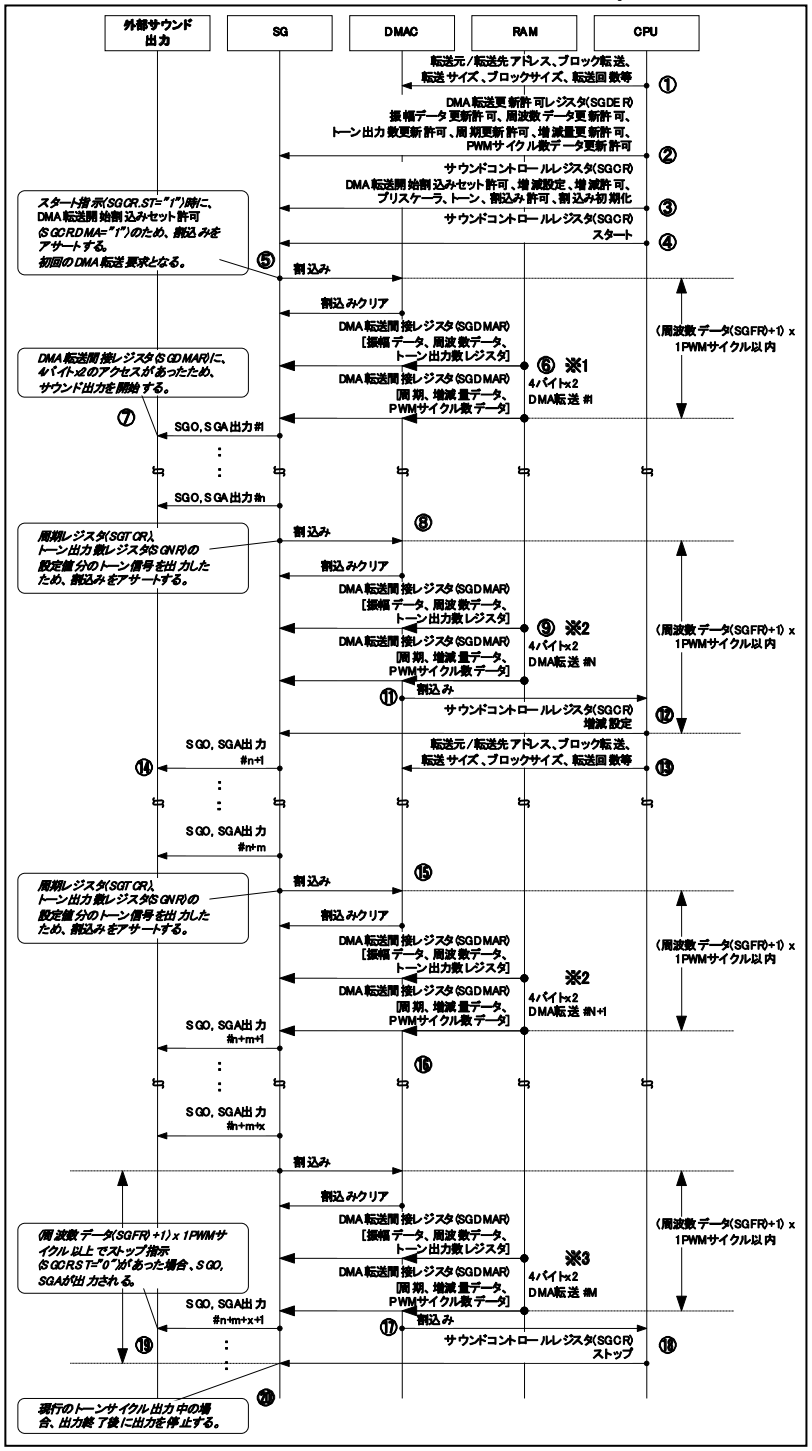
＜注意事項＞

⑤から⑥までのようにサウンドジェネレータから割込みを通知し DMA 転送が終了するまでには、(周波数データ(SGFR)+1) × 1PWM サイクル以内で完了する必要があります。

32.5.8.4 4 バイト×2 の DMA 転送を N 回と 4 バイト×2 の DMA 転送を M 回行う場合 (サウンド出力中に転送バイト数変更、および増減設定を変更)

4 バイト×2 の DMA 転送を N 回と 4 バイト×2 の DMA 転送を M 回行う場合について示します。

図 32-15 DMA と連携したサウンドジェネレータの動作 (サウンド出力中に転送バイト数変更、および増減設定を変更)



① ソフトウェアで、DMAC に DMA 転送に必要な設定を行います。なお、DMA 転送では、4 バイト×2 のブロック転送を N 回行います。DMA 転送間接レジスタ(SGDMAR)を介して、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」のデータを転送します。なお、DMAC の転送先アドレスの設定は、アドレス固定(DMA 転送間接レジスタのアドレス)とします。
② ソフトウェアで、サウンドジェネレータの DMA 転送更新許可レジスタ(SGDER)に、DMA 転送を行うレジスタの更新許可設定を行います。ここでは、「振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)」および「周期レジスタ(SGTCR)、増減量データレジスタ(SGIDR)、PWM サイクル数データレジスタ(SGPCR)」のすべてのレジスタを更新するように設定します。
③ ソフトウェアで、サウンドジェネレータのサウンドコントロールレジスタ(SGCR)に、サウンドジェネレータ制御情報を設定します。また、DMA 転送開始割込みセット許可ビットを許可設定にします。割込みビット(SGCR:INT)を初期化し、割込み許可ビット(SGCR:INTE)を設定します。
④ スタートビット(SGCR:ST)を"1"に設定します。
⑤ スタート指示(SGCR:ST=1)時に、DMA 転送開始割込みセット許可設定(SGCR:DMA=1)のため、割込みビット(SGCR:INT)がセットされて、割込み(PIRQ)が発生します。これは DMA 転送要求として使用されます。
⑥ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して、1 回目の転送で振幅データレジスタ(SGAR)、周波数データレジスタ(SGFR)、トーン出力数レジスタ(SGNR)の値を、2 回目の転送で周期レジスタ(SGTCR)、増減量データレジスタ(SGDR)、PWM サイクル数データレジスタ(SGPCR)の値を転送します。(*1:DMA 転送間接レジスタへの 4 バイト×2 のブロック転送が必須です。)
⑦ DMA 転送間接レジスタ(SGDMAR)に 4 バイト×2 のブロック転送が行われたため、サウンドジェネレータの SGO, SGA 出力が開始されます。
⑧ サウンドジェネレータが周期レジスタ(SGTCR)およびトーン出力数レジスタ(SGNR)に設定されたトーンパルス数を出力すると割込みが発生します。
⑨ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して DMA 転送を行います。 (*2: 2 回目以降の DMA 転送も、DMA 転送間接レジスタへの 4 バイト×2 のブロック転送が必須です。)
⑩ DMA 転送されたデータで、サウンド出力されます。
⑪ DMAC で設定回数分の DMA 転送が完了(4 バイト×2 を N 回転送)したとき、DMAC から CPU に割込みが通知されます。
⑫ ソフトウェアで、サウンドコントロールレジスタ(SGCR)の増減設定を変更します。
⑬ ソフトウェアで、DMAC に DMA 転送に必要な設定を行います。ここでは、4 バイト×2 を M 回転送するように設定します。
⑭ DMA 転送されたデータで、サウンド出力されます。
⑮ DMAC が割込みをクリアし、DMA 転送間接レジスタ(SGDMAR)を介して DMA 転送を行います。(*2: DMA 転送間接レジスタへの 4 バイト×2 のブロック転送が必須です。)
⑯ 以降、同様の動作が続きます。
⑰ DMAC が設定回数分の DMA 転送(4 バイト×2 を M 回転送)を終了したので、CPU 対して割込みが発生します。
⑱ CPU からサウンドジェネレータにストップ指示(SGCR:ST=0)を行います。
⑲ (周波数データ(SGFR)+1) × 1PWM サイクル以内でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は出力されません。(*3: N 回目の転送データは、サウンド用レジスタにライトされますが、サウンド出力はされません。N 回目の転送は、DMAC から CPU に対して割込みを発生させるための転送です)
⑳ (周波数データ(SGFR)+1) × 1PWM サイクルより後でストップ指示(SGCR:ST=0)があった場合、N 回目の転送データによる SGO, SGA は現行のトーンサイクル終了後に出力が停止します。(*3: N 回目の転送データが、サウンド出力されます。)

＜注意事項＞

⑤から⑥までのようにサウンドジェネレータから割込みを通知し DMA 転送が終了するまでには、(周波数データ(SGFR)+1) × 1PWM サイクル以内で完了する必要があります。

＜注意事項＞
⑧から⑫までのようにサウンドジェネレータから割込みを通知しソフトウェアが増減設定を変更するまでは、(周波数データ(SGFR)+1) × 1PWM サイクル以内で完了する必要があります。

＜注意事項＞
増減設定は、⑭のサウンド出力から有効になります。4 バイト×2 の N 回目の転送データを使用して、自動増減を行います。

33. レギュレータ制御

レギュレータ制御の概要，特長，構成など説明します。

33.1 概要

レギュレータ制御の概要について説明します。

内部電圧を生成しているレギュレータの動作は、デバイス状態遷移に応じて自動的に変更されます。以下の 3 つのレギュレータモードに自動的に変更されます。

- メインモード (通常動作時)
- サブモード (サブラン時)
- スタンバイモード (ストップ・時計モード時)

33.2 特長

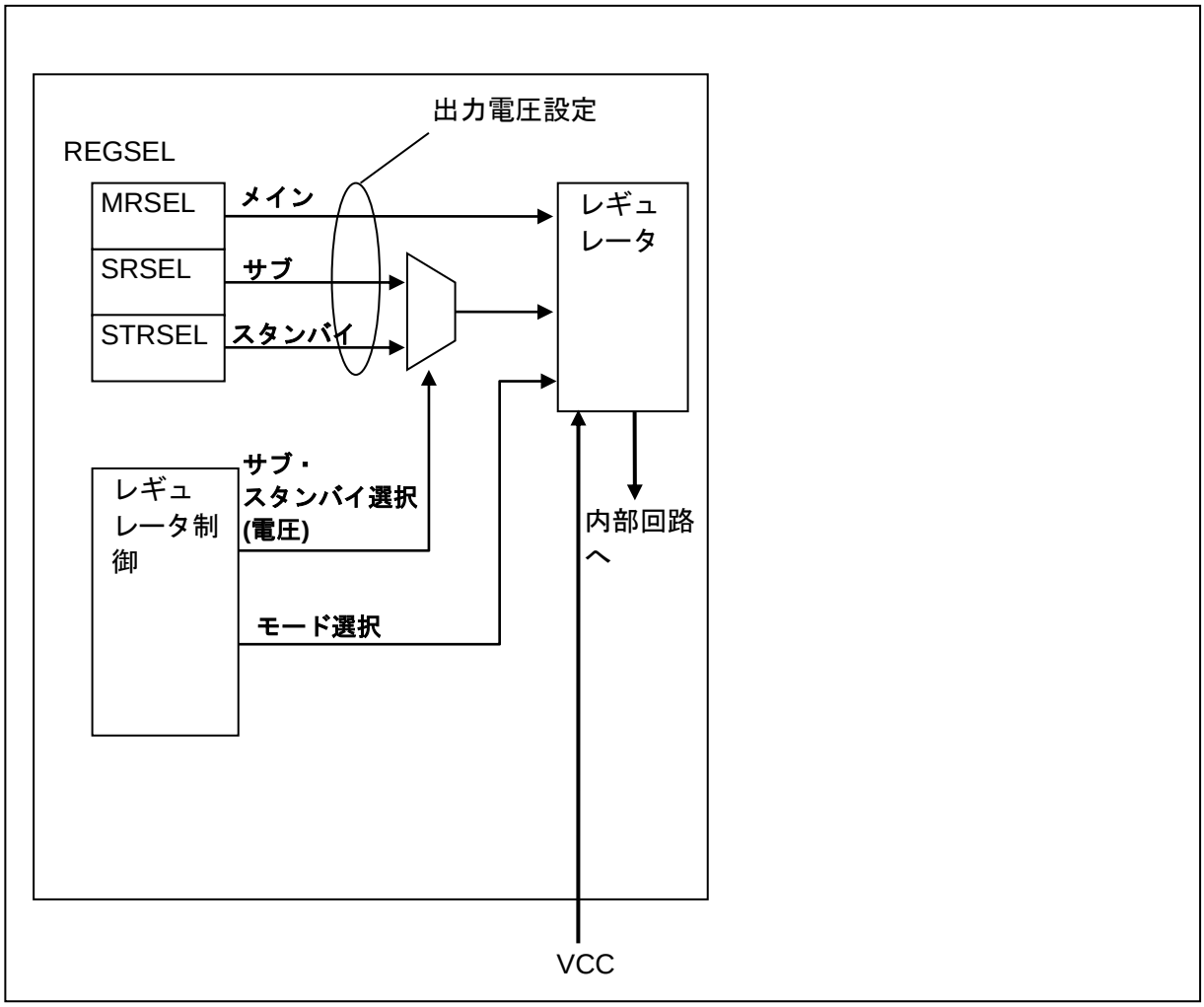
レギュレータ制御の特長について説明します。

- デバイス状態遷移に応じて、自動的にレギュレータモードを変更します。

33.3 構成

レギュレータ制御の構成について示します。

図 33-1 レギュレータ制御のブロックダイヤグラム



＜注意事項＞

サブモードとスタンバイモードの違いは、出力電圧設定のみです。

33.4 レジスタ

レギュレータ制御のレジスタについて説明します。

表 33-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0580	REGSEL	予約	予約	予約	レギュレータ出力電圧選択レジスタ

33.4.1 レギュレータ出力電圧選択レジスタ ：REGSEL (REGulator output voltage SELect register)

レギュレータ出力電圧選択レジスタのビット構成について説明します。

メイン・サブ・スタンバイの各レギュレータモードの出力電圧レベルを選択するレジスタです。

■ REGSEL： アドレス 0580H (アクセス ： バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MRSEL[1:0]		SRSEL[1:0]		STRSEL[2:0]			予約
初期値	0	1	1	0	0	1	1	0
属性	R/W0	R/W1	R/W1	R/W0	R/W1	R/W1	R/W0	R0,WX

[bit7, bit6] MRSEL[1:0] (Main Regulator voltage SELect)
メインレギュレータ出力電圧レベルを設定します。

MRSEL[1:0]	メインレギュレータ出力電圧
00	予約
01	1.2±0.1V
10	予約
11	予約

[bit5, bit4] SRSEL[1:0] (Sub Regulator voltage SELect)
サブレギュレータ出力電圧レベルを設定します。

SRSEL[1:0]	サブレギュレータ出力電圧
00	予約
01	予約
10	1.2±0.1V
11	予約

[bit3～bit1] STRSEL[2:0] (STandby Regulator voltage SELect)
スタンバイレギュレータ出力電圧レベルを設定します。

STRSEL[2:0]	スタンバイレギュレータ出力電圧
000	予約
001	予約
010	予約
011	0.9±0.1V
100	予約
101	予約
110	1.2±0.1V
111	予約

＜注意事項＞
本設定値は必ず 1.2V(STRSEL[2:0]=110)にして使用してください。

[bit0] 予約

33.5 動作説明

レギュレータ制御の動作説明について説明します。

スタンバイモードに入る前にSTRSEL[2:0]を110に設定してください。リセット直後はこの値 (STRSEL[2:0]=110)に設定されていないので注意してください。

34. ステッピングモータコントローラ

ステッピングモータコントローラについて説明します。

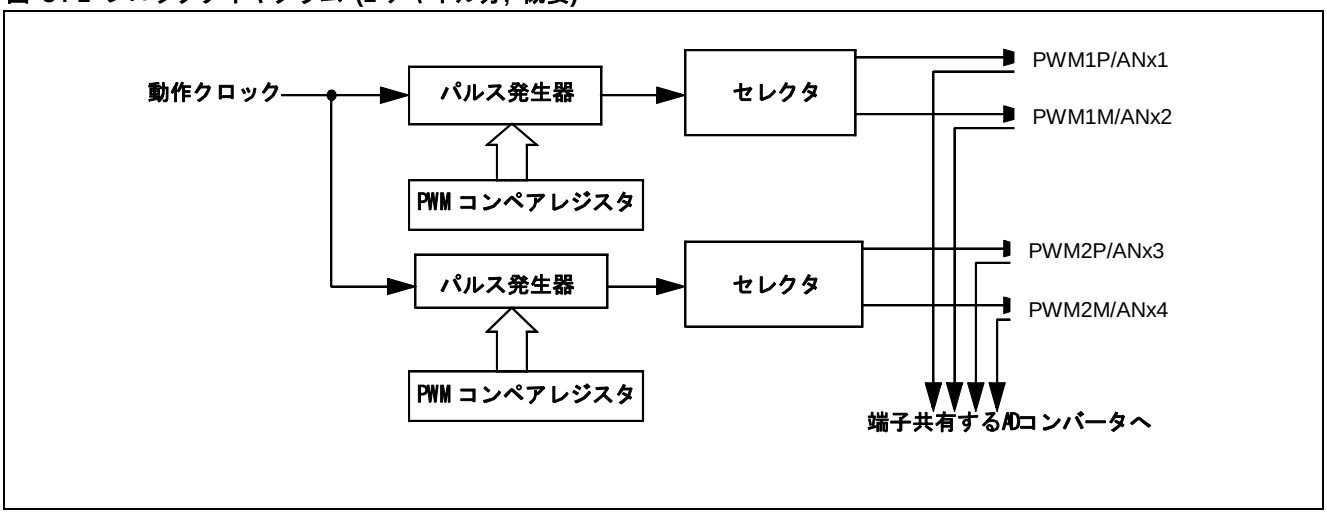
34.1 概要

ステッピングモータコントローラの概要について説明します。

本ステッピングモータコントローラは2個のPWMパルス発生器, 4個のモータドライバから構成されています。

4個のモータドライバは高出力駆動能力があり 2 個のモータコイルは 4 個の端部に直接的に接続することができます。PWM パルス発生器とセレクタロジックを組合せてモータの回転を制御するように設計されています。同期機構により 2 個の PWM 同期作動が確かなものとなります。

図 34-1 ブロックダイヤグラム (1 チャネル分, 概要)



34.2 特長

ステッピングモータコントローラの特長について説明します。

PWM 動作モード：

- 8/10 ビット動作から選択が可能

個数：6

PWM 動作クロック：

- 周辺クロック÷1, ÷2, ÷4, ÷5, ÷6, ÷8

34.3 構成

ステッピングモータコントローラの構成について示します。

図 34-2 ブロック図 (1 チャネル分, 詳細)



34.4 レジスタ

ステッピングモータコントローラのレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子一覧

チャネル番号	Base_addr	外部端子名			
		PWM1P	PWM1M	PWM2P	PWM2M
0	0x0200	PWM1P0	PWM1M0	PWM2P0	PWM2M0
1	0x0208	PWM1P1	PWM1M1	PWM2P1	PWM2M1
2	0x0210	PWM1P2	PWM1M2	PWM2P2	PWM2M2
3	0x0218	PWM1P3	PWM1M3	PWM2P3	PWM2M3
4	0x0220	PWM1P4	PWM1M4	PWM2P4	PWM2M4
5	0x0228	PWM1P5	PWM1M5	PWM2P5	PWM2M5

■ レジスタマップ

表 34-1　レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0200	PWC20		PWC10		PWM2 コンペアレジスタ 0 PWM1 コンペアレジスタ 0
0x204	予約	PWC0	PWS20	PWS10	PWM 制御レジスタ 0 PWM2 選択レジスタ 0 PWM1 選択レジスタ 0
0x0208	PWC21		PWC11		PWM2 コンペアレジスタ 1 PWM1 コンペアレジスタ 1
0x020C	予約	PWC1	PWS21	PWS11	PWM 制御レジスタ 1 PWM2 選択レジスタ 1 PWM1 選択レジスタ 1
0x0210	PWC22		PWC12		PWM2 コンペアレジスタ 2 PWM1 コンペアレジスタ 2
0x0214	予約	PWC2	PWS22	PWS12	PWM 制御レジスタ 2 PWM2 選択レジスタ 2 PWM1 選択レジスタ 2
0x0218	PWC23		PWC13		PWM2 コンペアレジスタ 3 PWM1 コンペアレジスタ 3
0x021C	予約	PWC3	PWS23	PWS13	PWM 制御レジスタ 3 PWM2 選択レジスタ 3 PWM1 選択レジスタ 3
0x0220	PWC24		PWC14		PWM2 コンペアレジスタ 4 PWM1 コンペアレジスタ 4
0x0224	予約	PWC4	PWS24	PWS14	PWM 制御レジスタ 4 PWM2 選択レジスタ 4 PWM1 選択レジスタ 4
0x0228	PWC25		PWC15		PWM2 コンペアレジスタ 5 PWM1 コンペアレジスタ 5
0x022C	予約	PWC5	PWS25	PWS15	PWM 制御レジスタ 5 PWM2 選択レジスタ 5 PWM1 選択レジスタ 5

34.4.1 PWM 制御レジスタ ：PWC

PWM 制御レジスタ (PWC)のビット構成について説明します。

ステッピングモータコントローラの起動・停止などの設定を行います。

■ PWC： アドレス Base_addr + 05H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	biit1	bit0
	予約	P2	P1	P0	CE	SC	ー	予約
初期値	0	0	0	0	0	0	ー	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R1,WX	R/W0

[bit7] 予約

[bit6～bit4] P2～P0： 動作クロック選択ビット

P2～P0 ビットは PWM パルス発生器に対するクロック入力信号を指定します。

P2	P1	P0	クロック入力	PWM 周期(FPCLK=16MHz 時)	
				SC=0	SC=1
0	0	0	FPCLK	16.0µs	64.0µs
0	0	1	1/2×FPCLK	32.0µs	128.0µs
0	1	0	1/4×FPCLK	64.0µs	256.0µs
0	1	1	1/8×FPCLK	128.0µs	512.0µs
1	0	0	予約	—	—
1	0	1	1/5×FPCLK	80.0µs	320.0µs
1	1	0	1/6×FPCLK	96.0µs	384.0µs
1	1	1	予約	—	—

FPCLK：周辺クロック(PCLK)

[bit3] CE： カウント許可ビット

CE ビットは PWM パルス発生器の動作を許可します。CE ビットが"1"に設定された場合、PWM パルス発生器は動作を開始します。出力ドライバからのスイッチングノイズを減少させるため、PWM2 パルス発生器は PWM1 パルス発生器から 1 マシンサイクル遅れてスタートします。

PWM パルス発生器が動作中に CE ビットを"0"クリアした場合は、PWM パルス発生器は初期化され動作を停止します。

＜注意事項＞

CE に"1"をセットする場合は、動作クロックの選択が完了した状態で行なってください。

[bit2] SC : 8/10 ビット切替えビット

SC ビットが"1"に設定されたとき PWM は 10 ビットで動作します。SC ビットが"0"に設定されたとき PWM は 8 ビットで動作します。

[bit1] - : **未定義**
常に"1"が読み出されます。書込みは動作に影響ありません。

[bit0] **予約**
必ず"0"を設定してください。

34.4.2 PWM1&2 コンペアレジスタ ：PWC1/PWC2

PWM1&2 コンペアレジスタ (PWC1/PWC2) のビット構成について説明します。

PWM1&2 の 2 つの 8(10)ビットコンペアレジスタの内容は、PWM パルスの幅を決定するものです。記憶された"00_H" ("000_H")の値は PWM のデューティが 0%であることを表し、"FF_H" ("$3FF_H$")は 99.6%(99.9%) のデューティを表します。

PWM1&2 コンペアレジスタは、任意の時点でアクセスすることができますが、変更された値は PWM2 選択レジスタの BS ビットが"1"に設定された後における現在の PWM サイクルの終わりにパルス幅に反映されます。

PWM 制御レジスタの SC ビットを"0"に設定し、PWM が 8 ビット動作をしている場合は、D9,D8 は、不定値となります。

PWM1&2 コンペアレジスタは、必ずハーフワードアクセスまたはワードアクセスしてください。

■ PWC1： アドレス Base_addr + 02_H (アクセス: ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	—	—	—	—	—	—	D9	D8
初期値	—	—	—	—	—	—	X	X
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	X	X	X	X	X	X	X	X
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ PWC2： アドレス Base_addr + 00_H (アクセス: ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	—	—	—	—	—	—	D9	D8
初期値	—	—	—	—	—	—	X	X
属性	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R1,WX	R/W	R/W
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	X	X	X	X	X	X	X	X
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit10] - : **未定義**
常に"1"が読み出されます。書込みは動作に影響ありません。

[bit9～bit0] D9～D0 : **コンペアデータ**
PWM のパルス幅を設定します。

34.4.3 PWM1 選択レジスタ ：PWS1

PWM1 選択レジスタ (PWS1) のビット構成について説明します。

PWM1 選択レジスタは、ステッピングモータコントローラの外部端子の出力が"0", "1", PWM パルス, ハイインピーダンスかを選択します。

■ PWS1： アドレス Base_addr + 07H(アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ー	ー	P2	P1	P0	M2	M1	M0
初期値	ー	ー	0	0	0	0	0	0
属性	R1,WX	R1,WX	R/W	R/W	R/W	R/W	R/W	R/W

[bit7, bit6] - : 未定義ビット
常に"1"が読み出されます。書込みは動作に影響ありません。

[bit5～bit3] P2～P0 : 出力選択ビット
P2, P1, P0 ビットは PWM1P0～PWM1P5 における出力信号を選択するものです。

[bit2～bit0] M2～M0 : 出力選択ビット
M2, M1, M0 ビットは PWM1M0～PWM1M5 における出力信号を選択するものです。

次の表は出力レベルと選択ビットの間の関係を示すものです。

P2	P1	P0	PWM1Pn	M2	M1	M0	PWM1Mn
0	0	0	L	0	0	0	L
0	0	1	H	0	0	1	H
0	1	X	PWM パルス	0	1	X	PWM パルス
1	X	X	ハイインピーダンス	1	X	X	ハイインピーダンス

n=0～5

34.4.4 PWM2 選択レジスタ ：PWS2

PWM2 選択レジスタ (PWS2) のビット構成について説明します。

PWM2 選択レジスタは、ステッピングモータコントローラの外部端子の出力が"0", "1", PWM パルス，ハイインピーダンスかを選択します。

■ PWS2： アドレス Base_addr + 06H (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ー	BS	P2	P1	P0	M2	M1	M0
初期値	ー	0	0	0	0	0	0	0
属性	R1,WX	R,W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7] - : 未定義
常に"1"が読み出されます。書込みは動作に影響ありません。

[bit6] BS : 書換えビット
BS ビットは PWM 出力に対する設定を同期させるために用意されています。BS ビットがセットされるまで、2 個のコンペアレジスタおよび 2 個の選択レジスタに対して行われた変更は出力信号には反映されません。

BS ビットが"1"にセットされたとき、PWM パルス発生器およびセレクタはレジスタ内容を現在の PWM サイクルの終わりでロードします。BS ビットは次の PWM サイクルの始めに自動的に"0"にクリアされます。もし BS ビットがこの自動クリアと同時にソフトウェアにより"1"に設定されると、BS ビットは"1"にセットされ、(すなわち変更されないままであり) 自動クリアは解除されます。

もし BS ビットがこの自動クリアと同時にソフトウェアにより"0"に設定されると、BS ビットは"0"にクリアされ、PWM パルス発生器およびセレクタはレジスタ内容を現在の PWM サイクルの終わりでロードされます。

＜注意事項＞
BS="1"の状態で、BS 以外のビットにリードモディファイライト系命令を実行すると BS ビットは"1"がリードされ、再度 BS ビットに"1"がライトされます。リードとライトの間に、PWM サイクルの始まりで BS ビットが自動クリアされた場合、BS ビットクリア後に再度"1"がセットされます。したがって、次の PWM サイクルの終わりまでに BS ビットが"1"にセットされない場合でも、PWM パルス発生器およびセレクタにレジスタの内容がロードされます。

[bit5～bit3] P2～P0 : 出力選択ビット
P2, P1, P0 ビットは PWM2P0～PWM2P5 における出力信号を選択するものです。

[bit2～bit0] M2～M0 : 出力選択ビット
M2, M1, M0 ビットは PWM2M0～PWM2M5 における出力信号を選択するものです。

次の表は出力レベルと選択ビットの間の関係を示すものです。

P2	P1	P0	PWM2Pn	M2	M1	M0	PWM2Mn
0	0	0	L	0	0	0	L
0	0	1	H	0	0	1	H
0	1	X	PWM パルス	0	1	X	PWM パルス
1	X	X	ハイインピーダンス	1	X	X	ハイインピーダンス

n=0～5

34.5 動作説明

ステッピングモータコントローラの動作について説明します。

34.5.1. PWM 動作

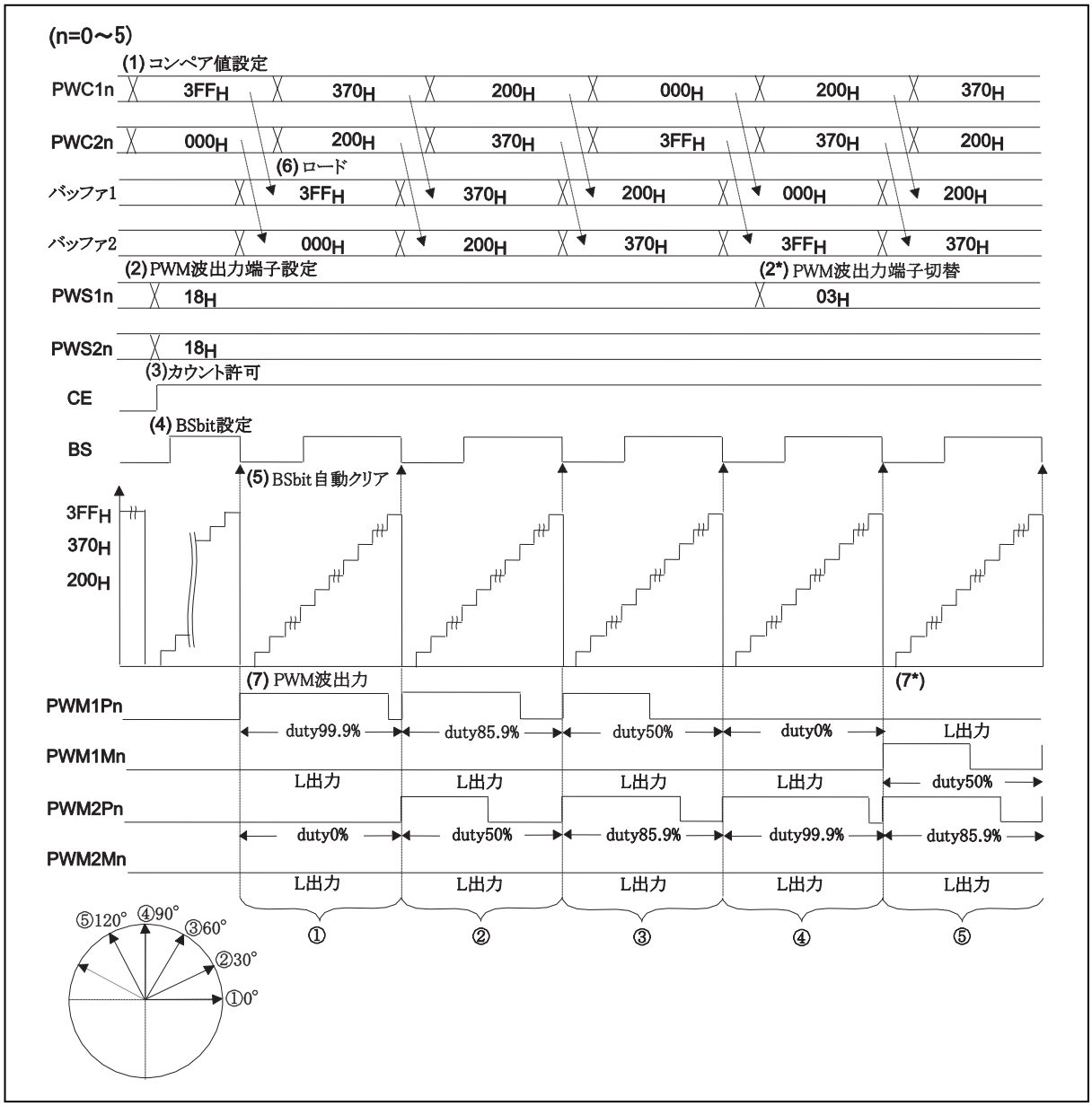
34.5.2. BS ビットによる PWM コンペアレジスタのロード動作

34.5.3. モータドライブ信号の選択

34.5.1 PWM 動作

PWM 動作について示します。

図 34-3 PWM 動作



(1) コンペア値設定
(2) PWM 波出力端子設定 / (2*)PWM 波出力端子切替え
(3) カウント許可
(4) BS ビット設定
(5) BS ビット自動クリア
(6) コンペア値ロード
(7) PWM 波出力 / (7*)(2*)による PWM 波出力切替え
(8) (1)から(7)までの動作を繰り返します

34.5.2 BS ビットによる PWM コンペアレジスタのロード動作

BS ビットによる PWM コンペアレジスタのロード動作について示します。

図 34-4 PWM コンペアレジスタ値のロード動作



(1) BS ビットの自動クリア ： ロードされ出力信号に反映される。
(2) 自動クリアと同時に BS ビットを"1"に設定した場合 ： ロードされ出力信号に反映される。
(3) 自動クリアと同時に BS ビットを"0"に設定した場合 ： ロードされ出力信号に反映される。
(4) PWM サイクル終了前に BS ビットを"0"に設定した場合 ： ロードされず出力信号は反映されない。
(「[bit6] BS： 書換えビットの＜注意事項＞」参照)

34.5.3 モータドライブ信号の選択

モータドライブ信号の選択について説明します。

ステッピングモータコントローラ関連の各端子に出力するモータドライブ信号は PWM 選択レジスタの設定により、端子ごとに 4 種類の信号から選択することができます。

図 34-2 にモータドライブ信号の選択と PWM 選択レジスタ 1, 2 の設定について示します。

これらの設定を行ったのち、PWM 選択レジスタ 2 の BS ビットに"1"を書き込むと、現在の PWM サイクルの終了時に設定値が有効となります。この BS ビットは次の PWM サイクルの先頭で自動的にクリアされます。BS ビットへの書込みと PWM サイクルの先頭でも BS ビットクリアが同時のときは、BS ビットへの書込みが優先され、BS ビットクリアは取り消されます。

表 34-2 モータドライブ信号の選択と PWM 選択レジスタ 1, 2 の設定

P2, P1, P0 ビット	PWM1P0～5 出力 PWM2P0～5 出力	M2, M1, M0 ビット	PWM1M0～5 出力 PWM2M0～5 出力
000_B	L	000_B	L
001_B	H	001_B	H
$01X_B$	PWM パルス	$01X_B$	PWM パルス
$1XX_B$	ハイインピーダンス	$1XX_B$	ハイインピーダンス

34.6 設定

ステッピングモータコントローラの設定について説明します。

表 34-3 PWM を動かすために必要な設定

設定	設定レジスタ	設定方法
PWM 動作の開始	PWM 制御 (PWC0～PWC5)	34.7.2 参照
PWM 動作クロックの設定		34.7.4 参照
8/10bit モード切替え		34.7.1 参照
コンペア値(デューティ値) の設定	PWM1&2 コンペア (PWC10～PWC15/PWC20～PWC25)	
モータドライブ信号の選択	PWS1&2 選択 (PWS10～PWS15/PWS20～PWS25)	34.7.5 参照
PWM 端子出力設定	端子をペリフェラル出力に設定 『I/O ポート』の章を参照してください。	

表 34-4 PWM を停止させるために必要な設定

設定	設定レジスタ	設定方法
PWM 動作の停止	PWM 制御 (PWC0～PWC5)	34.7.2 参照

表 34-5 PWM 出力を変更するために必要な設定

設定	設定レジスタ	設定方法
コンペア値(デューティ値)の設定	PWM1&2 コンペア (PWC10～PWC15/PWC20～PWC25)	34.7.1 参照
モータドライブ信号の選択	PWS1&2 選択 (PWS10～PWS15/PWS20～PWS25)	34.7.5 参照
書換えビット(BS ビット)の設定	PWM 制御 (PWC0～PWC5)	34.7.3 参照

34.7 Q&A

ステッピングモータコントローラの Q&A について説明します。

34.7.1. 周期とデューティの設定方法は?

34.7.2. PWM 動作を許可/停止するには?

34.7.3. デューティの変更を反映させるには?

34.7.4. 動作クロックの種類と選択方法は?

34.7.5. モータドライブ信号を変更したい場合は?

34.7.6. 端子を PWM 出力端子にするには?

34.7.7. 端子を A/D コンバータアナログ入力端子にするには?

34.7.1 周期とデューティの設定方法は?

周期とデューティの設定方法について説明します。

周期の設定, デューティ値の設定

- 周期 (動作クロック選択, 8/10bit 動作選択)は PWM 制御レジスタ PWC0～PWC5 に設定します。
- デューティ値は PWM1&2 コンペアレジスタ(PWC10～PWC15, PWC20～PWC25)に設定します。

式

周期 :

8bit 動作時 (PWC0～PWC5:SC=0) : (1/動作クロック)×256

10bit 動作時 (PWC0～PWC5:SC=1) : (1/動作クロック)×1024

＜注意事項＞

動作クロックは PWC0～PWC5:P[2:0]で指定します。(F_{PCLK}, 1/2×F_{PCLK}, 1/4×F_{PCLK}, 1/8×F_{PCLK}, 1/5×F_{PCLK}, 1/6×F_{PCLK} (F_{PCLK} : 周辺クロック))

デューティ :

8bit 動作時(PWC0～PWC5:SC=0) :

PWC1&2 コンペアレジスタ値 ＝デューティ×(256/100)

10bit 動作時(PWC0～PWC5:SC=1) :

PWC1&2 コンペアレジスタ値 ＝デューティ×(1024/100)

設定可能な範囲

周期 :

16µs, 32µs, 64µs, 80µs, 96µs, 128µs, 256µs, 320µs, 384µs, 512µs (F_{PCLK}=16MHz 時)

PWC1&2 コンペアレジスタ値 :

8bit 動作時 (PWC0～PWC5:SC=0) : 0～99.6%(0～FF_H)

10bit 動作時 (PWC0～PWC5:SC=1) : 0～99.9%(0～$3FF_H$)

34.7.2　PWM 動作を許可/停止するには?

PWM 動作を許可/停止について説明します。

PWM の動作許可

カウント許可ビット (PWC0～PWC5:CE)で行ってください。

制御	カウント許可ビット(CE)
PWM 動作を停止させるには	"0"にする
PWM 動作を許可するには	"1"にする

カウント許可は動作クロックの選択が完了した状態で行ってください。

＜注意事項＞
「34.4.1 PWM 制御レジスタ : PWC」の「bit3:カウント許可ビット」を参照してください。

34.7.3　デューティの変更を反映させるには?

デューティの変更の反映について説明します。

デューティの変更

PWM1&2 選択レジスタの BS ビットに"1"を設定することで、現在の PWM サイクルの終わりにパルス幅が更新されます。

制御	書換えビット(BS)
デューティを変更するには	"1"にする

＜注意事項＞
PWM1&2 コンペアレジスタのロードタイミングについては図 34-4 PWM コンペアレジスタ値のロード動作を参照してください。

34.7.4　動作クロックの種類と選択方法は?

動作クロックの種類と選択方法について説明します。

動作クロックの選択

動作クロック選択ビット(PWC0～PWC5:P[2:0])で行います。

動作クロック選択ビット(PWC0～PWC5:P[2:0])の設定については「34.4.1 PWM 制御レジスタ : PWC」を参照してください。

34.7.5　モータドライブ信号を変更したい場合は?

モータドライブ信号に設定することについて説明します。

モータドライブ信号の変更

出力選択ビット(PWS10～PWS15/PWS20～PWS25)でL, H, PWMパルス，ハイインピーダンスの選択が可能です。

出力選択ビットの設定については「34.4.3 PWM1 選択レジスタ ：PWS1」の「Bit2-0:出力選択ビット」および「34.4.4 PWM2 選択レジスタ ：PWS2」の「Bit2-0:出力選択ビット」を参照してください。

34.7.6　端子を PWM 出力端子にするには?

端子を PWM 出力端子に設定することについて説明します。

端子をペリフェラル出力に設定します。設定方法は『I/O ポート』の章を参照してください。

34.7.7　端子を A/D コンバータアナログ入力端子にするには?

端子を A/D コンバータアナログ入力端子に設定することについて説明します。

端子を A/D コンバータ入力に設定します。設定方法は『I/O ポート』の章を参照してください。

34.8 サンプルプログラム

サンプルプログラムについて示します。

設定手順例 1

PWM1P0,PWM2P0 から PWM パルス出力



<初期設定(SMC0)>

・ポート

ポートの SMC0 出力設定	『I/O ポート』の章参照

・SMC0 の制御

	レジスタ名．ビット名
PWM 制御レジスタの設定 サンプルクロック選択≫ 動作クロック選択≫ カウント設定≫ 8/ 10bit 切替え≫	PWC0 .S2 .P[2:0] .CE .SC

・デューティ設定

	レジスタ名．ビット名
PWC1 コンペアレジスタの設定	PWC10
PWC2 コンペアレジスタの設定	PWC20

・出力端子設定

	レジスタ名．ビット名
PWS1 選択レジスタの設定 PWM1P0 端子出力選択≫ PWM1M0 端子出力選択≫	PWS10 .P[2:0] .M[2:0]
PWS2 選択レジスタの設定 PWM2P0 端子出力選択≫ PWM2M0 端子出力選択≫	PWS20 .P[2:0] .M[2:0]

<起動(SMC0)>

・SMC0 起動

	レジスタ名．ビット名
カウント許可	PWC0.CE

・デューティ変更

	レジスタ名．ビット名
PWC1 コンペアレジスタの設定	PWC10
PWC2 コンペアレジスタの設定	PWC20

・BS ビットセット

	レジスタ名．ビット名
PWS2 選択レジスタの設定	PWS20.BS

プログラム例 1

```
void SMC0_sample_1(void)
{
        SMC0_initial();
        SMC0_start();
}
void SMC0_initial(void)
{
        IPORT_SETTING_SMC0_OUT()  /* SMC0 端子をペリフェラル出力に設定してください。 */

        IO_PWC0.byte = 0x36;      /* 設定値= 0011_0110 */
                                  /* bit7 = 0 S2 サンプルクロック設定 */
                                  /* bit6-4 = 011 P[2:0] 動作クロック設定 */
                                  /* bit3= 0 CE カウント禁止 */
                                  /* bit2 = 1 SC 10bit 動作 */
                                  /* bit1 = 1 未定義ビット */
                                  /* bit0 = 0 予約ビット */

        IO_PWC10.hword = 0x03ff;  /* PWM10 デューティ設定 */
        IO_PWC20.hword = 0x0000;  /* PWM20 デューティ設定 */

        IO_PWS10.byte = 0x1f;     /* 設定値= 0001_1111 */
                                  /* bit7-6 = 00 未定義ビット */
                                  /* bit5-3 = 011 P[2:0] PWM1P0=PWM 出力 */
                                  /* bit2-0 = 111 M[2:0] PWM1M0=Hi-Z 出力 */
        IO_PWS20.byte = 0x58;     /* 設定値= 0101_1000 */
                                  /* bit7 = 0 未定義ビット */
                                  /* bit6 = 1 BS 書換え設定 */
                                  /* bit5-3 = 011 P[2:0] PWM2P0=PWM 出力 */
                                  /* bit2-0 = 000 M[2:0] PWM2M0=L 出力 */
}

Void SMC0_start(void)
{
        IO_PWC0.bit.CE = 1;       /* bit3= 1 CE カウント許可 */
        ……
        ……                        /* BS ビットの自動クリア待ち */

        IO_PWC10.hword = 0x0370;  /* PWM10 デューティ変更 */
        IO_PWC20.hword = 0x0200;  /* PWM20 デューティ変更 */

        IO_PWS20.byte = 0x58;     /* 設定値= 0101_1000 */
        ……
}
```

34.9 注意事項

ステッピングモータコントローラの注意事項について説明します。

■ PWM 設定値変更時の注意

- PWM コンペアレジスタ 1, 2 (PWC10～PWC15, PWC20～PWC25), PWM 選択レジスタ 1, 2 (PWS10～PWS15, PWS20～PWS25) は常時アクセス可能ですが、PWM の"H"幅設定を変更する、あるいは PWM 出力を変更するためには、これらのレジスタに設定値を書き込んだあと(もしくは同時に)、PWM2 選択レジスタの BS ビットに"1"を書き込む必要があります。
- BS ビットが"1"にセットされると、現在の PWM サイクルの終了時に新しい設定値が有効になり BS ビットは自動的にクリアされます。
- また、BS ビットへの"1"書込みと、PWM サイクル終了時の BS ビットリセットが同時に行われた場合は、"1"書込みが優先され、BS ビットリセットはキャンセルされます。

35. LCD コントローラ

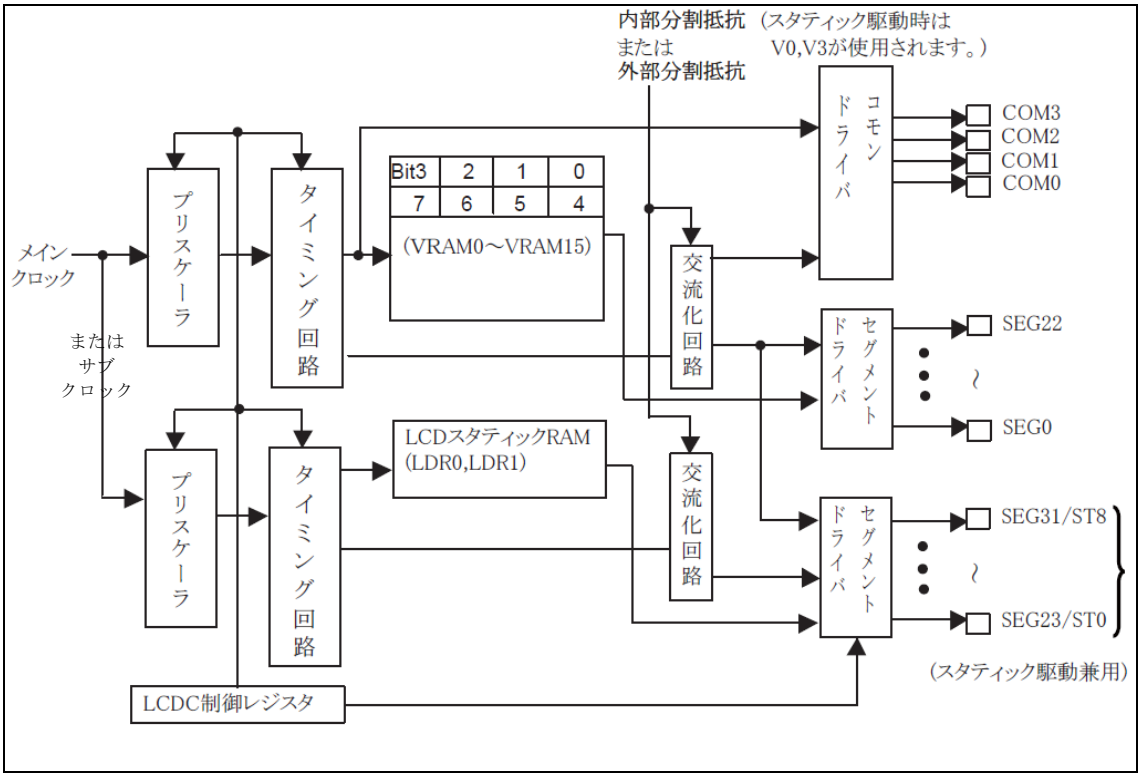
LCD コントローラについて説明します。

35.1 概要

LCD コントローラの概要について説明します。

デューティが 1/2, 1/3, 1/4 から選択できて、最大 128 素子の表示が可能な LCD です。

さらに、スタティック LCD 出力として、最大 8 素子の表示が可能です。



35.2 特長

LCD コントローラの特長について説明します。

＜デューティ駆動＞

- 個数 : 1 (4 コモン×32 セグメント)
- 表示 : 最大 128 素子(1/4 デューティ時)
- デューティ : 3 種類から選択 (1/2, 1/3, 1/4)
- バイアス : 1/2, 1/3 から選択

バイアス，デューティ，コモン出力の組合せ

バイアス	1/2 デューティ 出力モード	1/3 デューティ 出力モード	1/4 デューティ 出力モード
1/2 バイアス	○	×	×
1/3 バイアス	×	○	○

○：推奨モード
×：使用禁止

- フレーム周期 : 4 種類から選択可能 (クロックは、メインクロックまたはサブクロックを使用)
- ドライバ : 内蔵 (内部分割抵抗), または V0～V3 端子に外部分割抵抗を接続可能
- データメモリ : 16 バイトの表示用データメモリ内蔵
- 非表示選択 : 可能
  - 端子 : COM0～COM3, SEG0～SEG31, V0, V1, V2, V3 端子は、汎用ポートと兼用で切替えが可能
    SEG23～SEG31 端子は、スタティック駆動端子(ST0～ST8)と兼用で切替えが可能
- デューティ駆動時は、最大で 4 本のコモン出力(COM0～COM3)と 32 本のセグメント出力(SEG0～SEG31)が使用できます。
  - その他 : 外部分割抵抗でも LCD 停止時に電流を遮断可能

＜スタティック駆動＞

- 個数 : 1 (1 コモン×8 セグメント)
- 表示 : 最大 8 素子
- フレーム周期 : 4 種類から選択可能 (クロックは, メインクロックまたはサブクロックを使用)
- 端子 : ST0～ST8 端子は、汎用ポートとデューティ駆動端子 (SEG23～SEG31) と兼用で切替えが可能
- スタティック駆動時は、最大で 1 本のスタティックコモン出力と 8 本のスタティックセグメント出力 (ST0～ST8) が使用できます。

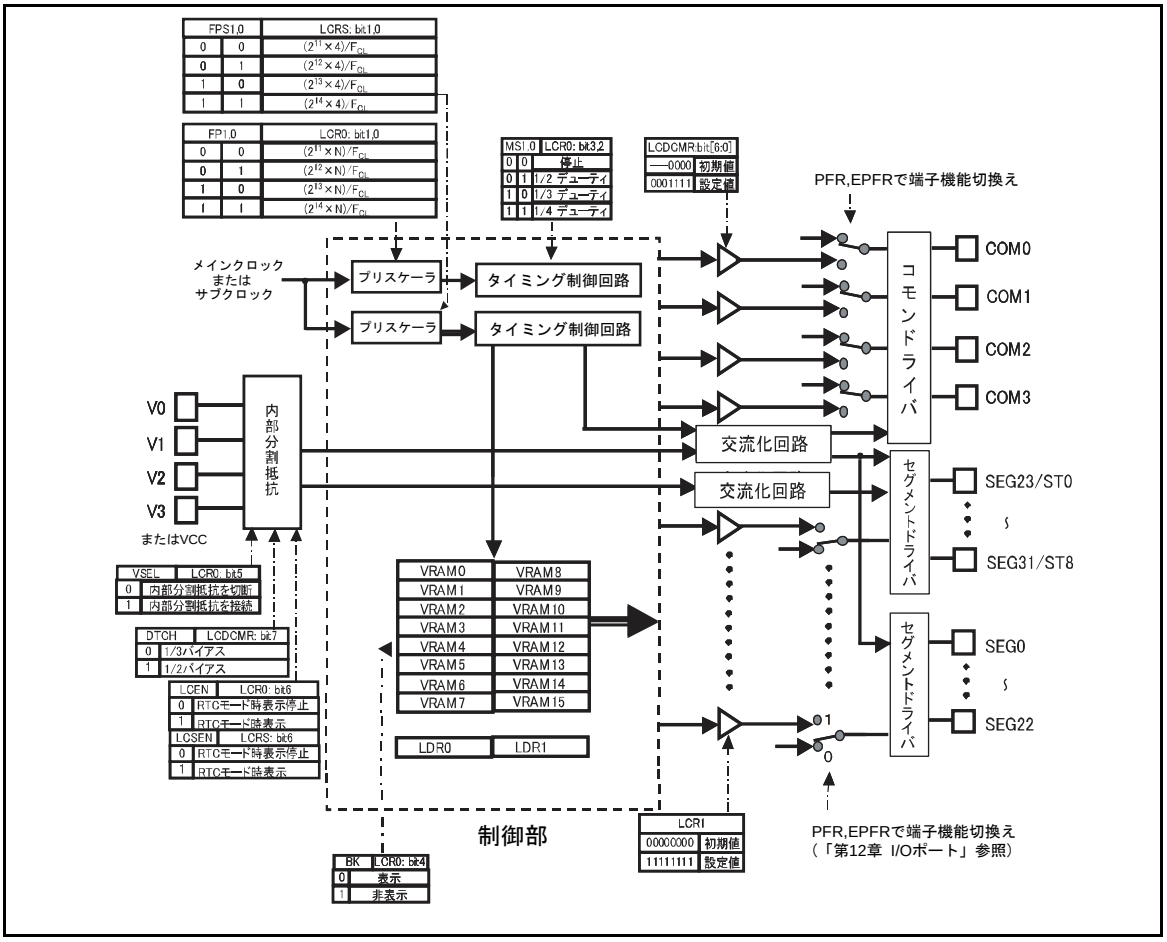
35.3 構成

LCD コントローラの構成を示します。

■ LCD コントローラの構成図

LCD コントローラの構成図を、図 35-1 に示します。

図 35-1 構成図



35.4 レジスタ

LCD コントローラのレジスタについて説明します。

■ レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x05A8	LCDCMR	LCRS	LCR0	LCR1	コモン端子切替えレジスタ LCDC スタティック制御レジスタ LCD 制御レジスタ 0 LCD 制御レジスタ 1
0x05AC	VRAM0	VRAM1	VRAM2	VRAM3	表示用データメモリ
0x05B0	VRAM4	VRAM5	VRAM6	VRAM7	
0x05B4	VRAM8	VRAM9	VRAM10	VRAM11	
0x05B8	VRAM12	VRAM13	VRAM14	VRAM15	
0x05BC	LDR		予約		スタティック LCD ディスプレイデータレジスタ

35.4.1 LCD 制御レジスタ 0 : LCR0

LCD 制御レジスタ 0 のビット構成について示します。

フレーム周期とそのクロック選択, 表示モード, 表示/非表示選択モードでの動作許可, 駆動電源制御などを行うためのレジスタです。

■ LCR0 : アドレス 05AA$_H$ (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	CSS	LCEN	VSEL	BK	MS1	MS0	FP1	FP0
初期値	0	0	0	1	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7] CSS : クロック選択ビット
本モジュールで使用するクロックを選択します。

CSS	動作
0	メインクロック
1	サブクロック

クロック 1 系統品種の場合は、"0"を設定してください。

[bit6] LCEN : 時計モード動作許可

LCEN	動作
0	時計モードで LCD 表示停止
1	時計モードで LCD 表示

[bit5] VSEL : LCD 駆動電源制御

VSEL	動作
0	内部分割抵抗切断
1	内部分割抵抗接続

外部に分割抵抗を接続する場合には、LCD 駆動電源制御ビット(VSEL)を"0"にする必要があります。

[bit4] BK : ブランキング選択

BK	動作
0	LCD 表示
1	LCD 非表示

[bit3, bit2] MS1, MS0 : 表示モード選択

MS1	MS0	表示モード
0	0	LCD 動作停止
0	1	1/2 デューティ出力モード (時分割数：N=2, COM0, COM1)
1	0	1/3 デューティ出力モード (時分割数：N=3, COM0～COM2)
1	1	1/4 デューティ出力モード (時分割数：N=4, COM0～COM3)

表示モード選択ビット(MS[1:0])を"00"に設定した場合、LCD コントローラは動作を停止し、コモン端子/セグメント端子は、"L"レベルを出力します。

[bit1, bit0] FP1, FP0 : フレーム周期

FP1	FP0	フレーム周期
0	0	CSS=0 の場合：$(2^{11}\times N)/F_{CL}$　CSS=1 の場合：$(2^{3}\times N)/F_{CL}$
0	1	CSS=0 の場合：$(2^{12}\times N)/F_{CL}$　CSS=1 の場合：$(2^{4}\times N)/F_{CL}$
1	0	CSS=0 の場合：$(2^{13}\times N)/F_{CL}$　CSS=1 の場合：$(2^{5}\times N)/F_{CL}$
1	1	CSS=0 の場合：$(2^{14}\times N)/F_{CL}$　CSS=1 の場合：$(2^{6}\times N)/F_{CL}$

F_{CL} : メインクロック(LCR0:CSS=0), またはサブクロック(LCR0:CSS=1)
N : 時分割数 (表示モード選択ビット MS1, MS0 の設定による)

使用する LCD パネルのフレーム周波数に対応した最適な設定を選択してください。

35.4.2 表示用データメモリ ：VRAM

表示用データメモリについて示します。

表示用データを設定するためのメモリ(VRAM)領域です。

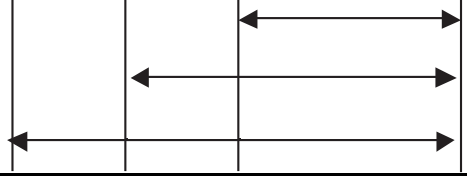
- **VRAM0 (SEG0, SEG1) ： アドレス 05AC$_H$ (アクセス ： バイト, ハーフワード,ワード)**
- **VRAM1 (SEG2, SEG3) ： アドレス 05AD$_H$ (アクセス ： バイト, ハーフワード,ワード)**
- **VRAM2 (SEG4, SEG5) ： アドレス 05AE$_H$ (アクセス ： バイト, ハーフワード,ワード)**
- **VRAM3 (SEG6, SEG7) ： アドレス 05AF$_H$ (アクセス ： バイト, ハーフワード,ワード)**
- **VRAM4 (SEG8, SEG9) ： アドレス 05B0$_H$ (アクセス ： バイト, ハーフワード,ワード)**
- **VRAM5 (SEG10, SEG11) ： アドレス 05B1$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM6 (SEG12, SEG13) ： アドレス 05B2$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM7 (SEG14, SEG15) ： アドレス 05B3$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM8 (SEG16, SEG17) ： アドレス 05B4$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM9 (SEG18, SEG19) ： アドレス 05B5$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM10 (SEG20, SEG21) ： アドレス 05B6$_H$ (アクセス ： バイト, ハーフワード,ワード)**
- **VRAM11 (SEG22, SEG23) ： アドレス 05B7$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM12 (SEG24, SEG25) ： アドレス 05B8$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM13 (SEG26, SEG27) ： アドレス 05B9$_H$ (アクセス ： バイト, ハーフワード, ワード)**
- **VRAM14 (SEG28, SEG29) ： アドレス 05BA$_H$ (アクセス ： バイト,ハーフワード,ワード)**
- **VRAM15 (SEG30, SEG31) ： アドレス 05BB$_H$ (アクセス ： バイト,ハーフワード,ワード)**

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D07	D06	D05	D04	D03	D02	D01	D00
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

表示用 RAM は、LCD コントローラ/ドライバの動作に関係なく任意のタイミングでリード/ライトできます。

VRAM とコモン端子／セグメント端子の対応

VRAM0	bit3	bit2	bit1	bit0	SEG0
	bit7	bit6	bit5	bit4	SEG1
VRAM1	bit3	bit2	bit1	bit0	SEG2
	bit7	bit6	bit5	bit4	SEG3
VRAM2	bit3	bit2	bit1	bit0	SEG4
	bit7	bit6	bit5	bit4	SEG5
VRAM3	bit3	bit2	bit1	bit0	SEG6
	bit7	bit6	bit5	bit4	SEG7
VRAM4	bit3	bit2	bit1	bit0	SEG8
	bit7	bit6	bit5	bit4	SEG9
VRAM5	bit3	bit2	bit1	bit0	SEG10
	bit7	bit6	bit5	bit4	SEG11
VRAM6	bit3	bit2	bit1	bit0	SEG12
	bit7	bit6	bit5	bit4	SEG13
VRAM7	bit3	bit2	bit1	bit0	SEG14
	bit7	bit6	bit5	bit4	SEG15
VRAM8	bit3	bit2	bit1	bit0	SEG16
	bit7	bit6	bit5	bit4	SEG17
VRAM9	bit3	bit2	bit1	bit0	SEG18
	bit7	bit6	bit5	bit4	SEG19
VRAM10	bit3	bit2	bit1	bit0	SEG20
	bit7	bit6	bit5	bit4	SEG21
VRAM11	bit3	bit2	bit1	bit0	SEG22
	bit7	bit6	bit5	bit4	SEG23
VRAM12	bit3	bit2	bit1	bit0	SEG24
	bit7	bit6	bit5	bit4	SEG25
VRAM13	bit3	bit2	bit1	bit0	SEG26
	bit7	bit6	bit5	bit4	SEG27
VRAM14	bit3	bit2	bit1	bit0	SEG28
	bit7	bit6	bit5	bit4	SEG29
VRAM15	bit3	bit2	bit1	bit0	SEG30
	bit7	bit6	bit5	bit4	SEG31
	COM3	COM2	COM1	COM0	



1/2デューティ出力モードの場合に使用されるRAM領域とコモン端子

1/3デューティ出力モードの場合に使用されるRAM領域とコモン端子

1/4デューティ出力モードの場合に使用されるRAM領域とコモン端子

35.4.3 LCDC 制御レジスタ 1 : LCR1

LCDC 制御レジスタ 1 のビット構成について示します。

■ LCR1 : アドレス 05AB$_H$ (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1

[bit7～bit0] 予約
LCD 使用時には常に"11111111"を設定してください。

35.4.4 コモン端子切替えレジスタ : LCDCMR

コモン端子切替えレジスタのビット構成について示します。

■ LCDCMR : アドレス 05A8$_H$ (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DTCH	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W0	R/W0	R/W0	R/W1	R/W1	R/W1	R/W1

[bit7] DTCH : バイアス選択

DTCH	動作
0	1/3 バイアス
1	1/2 バイアス

[bit6～bit4] 予約
LCD 使用時には常に"000"を設定してください。

[bit3～bit0] 予約
LCD 使用時には常に"1111"を設定してください。

35.4.5 LCDC スタティック制御レジスタ ：LCRS

LCDC スタティック制御レジスタのビット構成について示します。

■ LCRS： アドレス 05A9H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SCSS	LCSEN	LCS3	LCS2	LCS1	LCS0	FPS1	FPS0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7] SCSS： スタティック駆動用フレーム周期発生クロック選択ビット

SCSS	動作
0	メインクロック
1	サブクロック

クロック 1 系統品種の場合は, "0"を設定してください。

[bit6] LCSEN： 時計モード動作許可

LCSEN	動作
0	時計モードで LCD 表示停止
1	時計モードで LCD 表示

[bit5～bit2] LCS3～LCS0： スタティック駆動選択

LCS3	LCS2	LCS1	LCS0	スタティック駆動選択ポート
				コモン／セグメント出力
0	0	0	0	OFF
0	0	0	1	ST0, ST1
0	0	1	0	ST0～ST2
0	0	1	1	ST0～ST3
0	1	0	0	ST0～ST4
0	1	0	1	ST0～ST5
0	1	1	0	ST0～ST6
0	1	1	1	ST0～ST7
1	X	X	X	ST0～ST8

＜注意事項＞

スタティック駆動選択の使用方法については、「35.5.5 スタティック駆動出力波形」を参照してください。

[bit1, bit0] FPS1, FPS0 : フレーム周期

FPS1	FPS0	フレーム周期
0	0	SCSS=0 の場合：$(2^{11}\times4)/F_{CL}$　SCSS=1 の場合：$(2^{3}\times4)/F_{CL}$
0	1	SCSS=0 の場合：$(2^{12}\times4)/F_{CL}$　SCSS=1 の場合：$(2^{4}\times4)/F_{CL}$
1	0	SCSS=0 の場合：$(2^{13}\times4)/F_{CL}$　SCSS=1 の場合：$(2^{5}\times4)/F_{CL}$
1	1	SCSS=0 の場合：$(2^{14}\times4)/F_{CL}$　SCSS=1 の場合：$(2^{6}\times4)/F_{CL}$

F_{CL} :　メインクロック(LCRS:SCSS=0), またはサブクロック(LCRS:SCSS=1)

使用する LCD パネルのフレーム周波数に対応した最適な設定を選択してください。
スタティック駆動時のフレーム周期は、1/4 デューティ出力モードの周期と同じです。
PFR 設定と LCS[3:0]="0000"以外の設定をすると、スタティック駆動が有効になります。
PFR 設定と LCS[3:0]="0000"を設定すると、デューティ駆動が有効になります。

35.4.6 スタティック LCD ディスプレイデータレジスタ ：LDR

スタティック LCD ディスプレイデータレジスタのビット構成について示します。

■ LDR0： アドレス 05BC$_H$ (アクセス：バイト，ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	–	–	–	–	–	–	–	ST8
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W

[bit7～bit1] 未定義ビット
常に"0"が読み出されます。書込みは動作に影響ありません。

[bit0] ST8： スタティック出力データ
ST8 に対するスタティック出力データになります。

■ LDR1： アドレス 05BD$_H$ (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ST7	ST6	ST5	ST4	ST3	ST2	ST1	ST0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] ST7～ST0： スタティック出力データ
ST0～ST7 に対するスタティック出力データになります。

スタティック出力データ対応表

LDR	動作
ST8	ST8 スタティック出力データ
ST7	ST7 スタティック出力データ
ST6	ST6 スタティック出力データ
ST5	ST5 スタティック出力データ
ST4	ST4 スタティック出力データ
ST3	ST3 スタティック出力データ
ST2	ST2 スタティック出力データ
ST1	ST1 スタティック出力データ
ST0	ST0 スタティック出力データ

本レジスタは、(LCRS レジスタで設定された) LCD のスタティック駆動ポートへ出力するデータを設定するレジスタです。設定されたデータが周期的に出力反転することにより、LCD のスタティック駆動を行います。

特にコモン出力端子は設けておりません。セグメントの 1 つをコモン出力に割り当ててください。例えば、ST0～ST8 までをスタティック駆動端子とする場合、ST8 をコモン端子とし LDR0:ST8="0"にセットします。さらに、LDR1:ST[7:0]="11111111"とすることにより、ST0～ST7 で選択された LCD が表示されます。

35.5 動作説明

LCD コントローラの動作について説明します。

35.5.1. LCD コントローラ／ドライバ (LCDC) の動作

35.5.2. 1/2 デューティ出力波形

35.5.3. 1/3 デューティ出力波形

35.5.4. 1/4 デューティ出力波形

35.5.5. スタティック駆動出力波形

35.5.1 LCD コントローラ／ドライバ (LCDC) の動作

LCD コントローラ／ドライバ (LCDC) の動作について示します。

(1) 表示用データメモリについて

＜デューティ駆動＞

表示用データメモリ(VRAM) にあらかじめ値を設定します。

＜スタティック駆動＞

表示用データメモリ(LDR0, LDR1) にあらかじめ値を設定します。

(2) 各レジスタに必要な設定を書き込みます。

(3) 出力端子について

＜デューティ駆動＞

フレーム周期発生用クロックが発振しているとLCDを駆動する波形を、コモン／セグメント出力端子 (COM0～COM3, SEG0～SEG31) に出力します。

VRAMの内容を、コモン信号のタイミングに同期して自動的に読み出されセグメント出力端子より出力します。

(ビットを"1"に設定した場合は、セグメント出力端子より選択波形を出力します。
ビットを"0"に設定した場合は、セグメント出力端子より非選択波形を出力します。)

表示モードが 1/2 デューティの場合の COM2, COM3 端子と、1/3 デューティの場合の COM3 端子からは、非選択レベルの波形を出力します。

＜スタティック駆動＞

フレーム周期発生用クロックが発振していると LCD を駆動する波形を、コモン／セグメント出力端子 (ST0～ST8) に出力します。

(4)　出力波形について

＜デューティ駆動＞

出力波形は、デューティの設定にしたがって 2 フレーム交流化波形で駆動しています。

＜スタティック駆動＞

出力波形は、1 フレーム交流化波形で駆動しています。

(5)　時計モード時の動作について

＜デューティ駆動＞

動作許可(LCEN="1") の場合、LCD は表示されます。

＜スタティック駆動＞

動作許可(LCSEN="1") の場合、LCD は表示されます。

(6)　ブランキング機能について

＜デューティ駆動＞

ブランキングで非表示 (BK="1") を選んで、LCD を消灯させることができます。

ただし、非選択波形が出ています。

(7)　LCD の停止状態について

＜デューティ駆動＞

LCD の表示動作を停止した場合 (MS[1:0] = "00")、コモン／セグメント出力端子は，共に "L" レベルになります。

＜スタティック駆動＞

LCD の表示動作を停止した場合 (LCS[3:0] = "0000")、コモン／セグメント出力端子は，共に"L" レベルになります。

スタティック駆動(ST0～ST8)の LCD を表示から非表示させる場合は、スタティック駆動選択ポート(LCS[3:0])は変更せずに、LDR0:ST8="0" かつ LDR1:ST[7:0]="00000000" に設定してください。

スタティック駆動(ST0～ST8)の端子からは、同電位のパルスが出力されます。

35.5.2 1/2 デューティ出力波形

1/2 デューティ出力波形について示します。

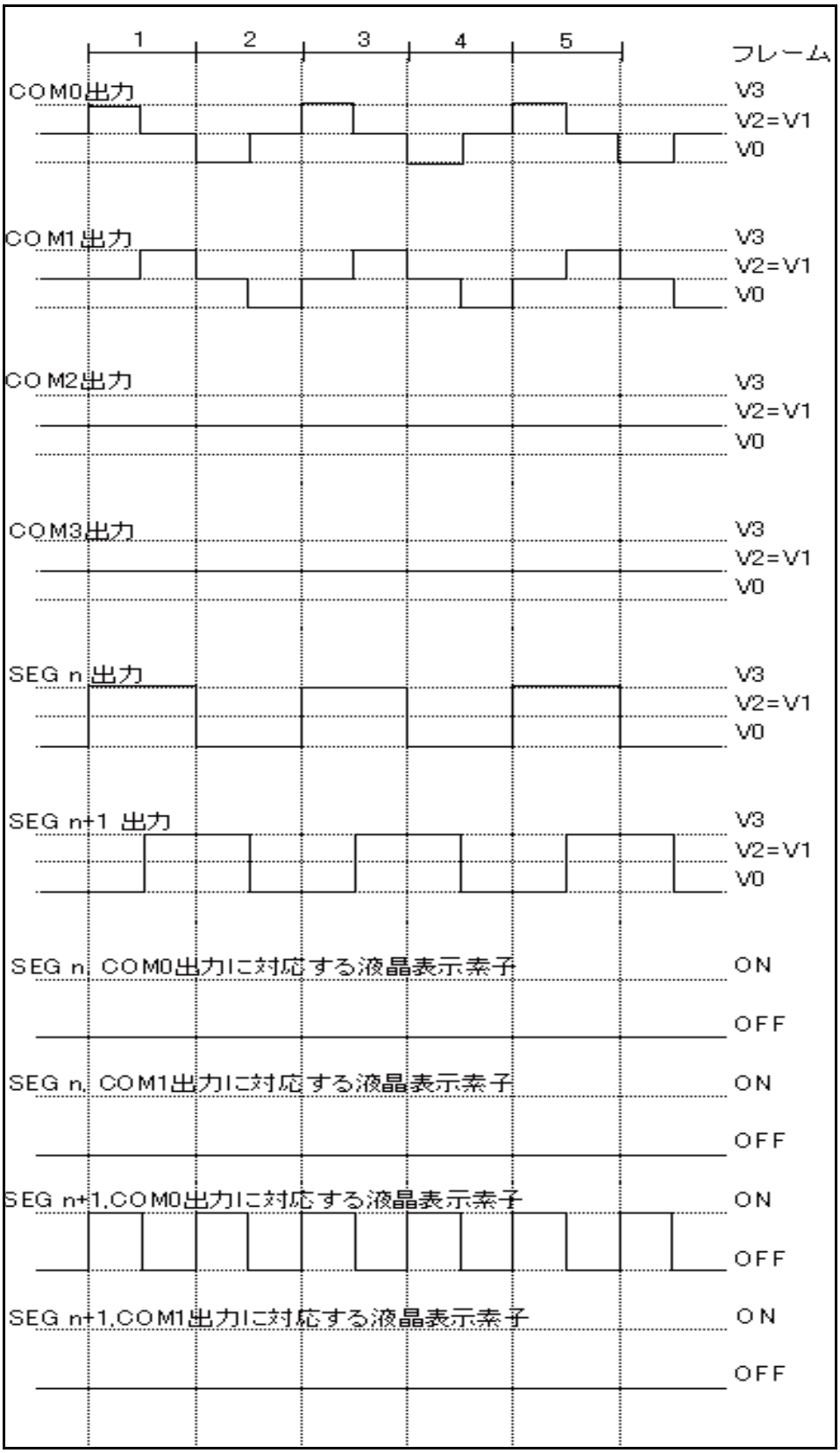
LCD の表示に使用するのは、COM0 出力と COM1 出力のみです。COM2 出力, COM3 出力は使用しません。

1/2 バイアス出力波形例

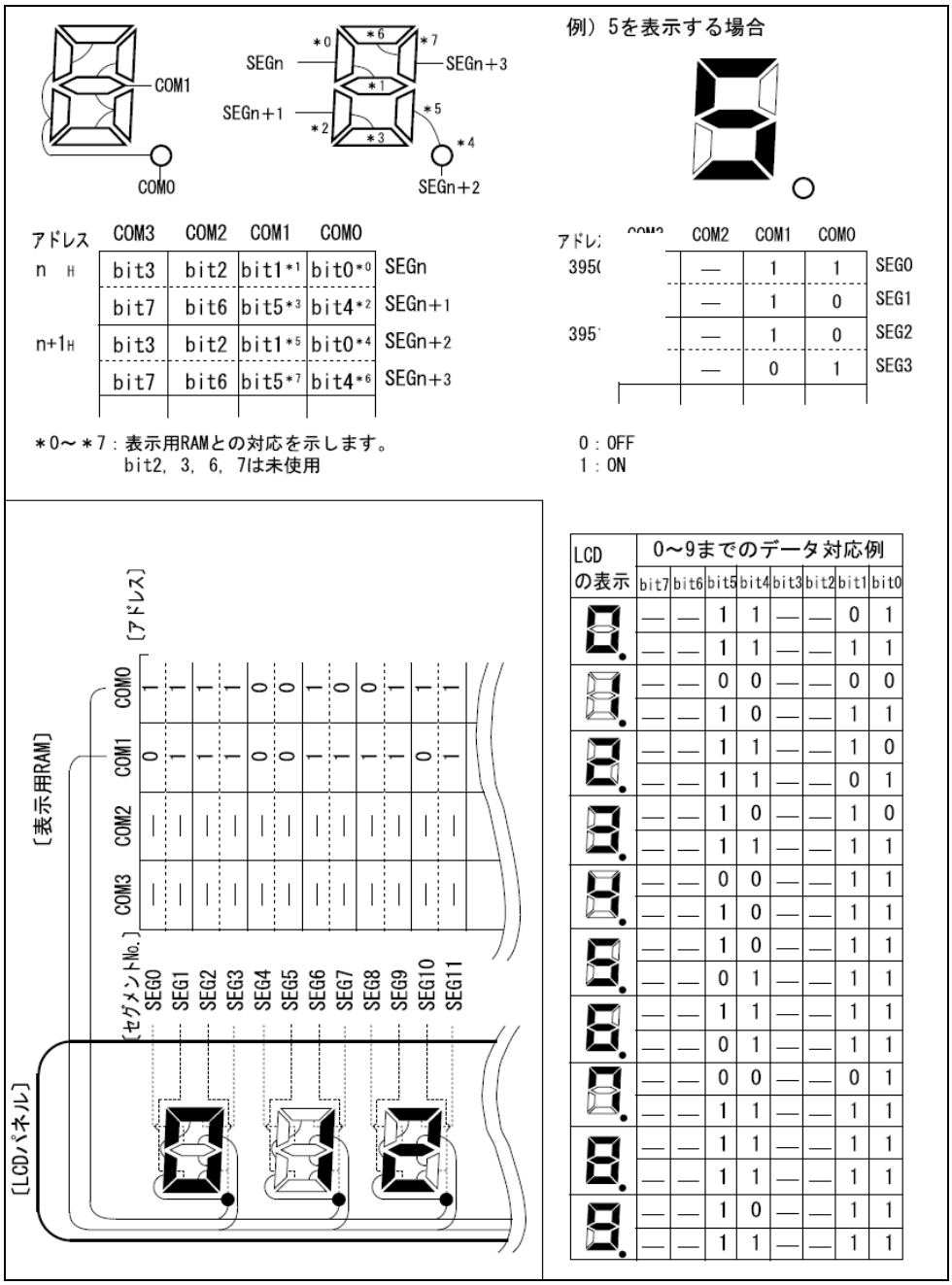
コモン出力とセグメント出力の電位差が最大となった液晶素子が点灯します。

表示用データメモリの内容例

セグメント	表示用データメモリの内容			
	COM3 出力	COM2 出力	COM1 出力	COM0 出力
SEG n 出力	−	−	0	0
SEG n+1 出力	−	−	0	1


1
2
3
4
5
フレーム
COM0出力
COM1出力
COM2出力
COM3出力
SEG n 出力
SEG n+1 出力
V3
V2=V1
V0
SEG n, COM0出力に対応する液晶表示素子
SEG n, COM1出力に対応する液晶表示素子
SEG n+1,COM0出力に対応する液晶表示素子
SEG n+1,COM1出力に対応する液晶表示素子
ON
OFF

■ LCDパネルの結線例と表示データ例 (1/2デューティ駆動方式)



アドレス	COM3	COM2	COM1	COM0	
n H	bit3	bit2	bit1*1	bit0*0	SEGn
	bit7	bit6	bit5*3	bit4*2	SEGn+1
n+1H	bit3	bit2	bit1*5	bit0*4	SEGn+2
	bit7	bit6	bit5*7	bit4*6	SEGn+3

*0～*7：表示用RAMとの対応を示します。
bit2, 3, 6, 7は未使用

アドレス	COM3	COM2	COM1	COM0	
395(		—	1	1	SEG0
		—	1	0	SEG1
395'		—	1	0	SEG2
		—	0	1	SEG3

0：OFF
1：ON

LCDの表示	0～9までのデータ対応例							
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	—	—	1	1	—	—	0	1
	—	—	1	1	—	—	1	1
1	—	—	0	0	—	—	0	0
	—	—	1	0	—	—	1	1
2	—	—	1	1	—	—	1	0
	—	—	1	1	—	—	0	1
3	—	—	1	0	—	—	1	0
	—	—	1	1	—	—	1	1
4	—	—	0	0	—	—	1	1
	—	—	1	0	—	—	1	1
5	—	—	1	0	—	—	1	1
	—	—	0	1	—	—	1	1
6	—	—	1	1	—	—	1	1
	—	—	0	1	—	—	1	1
7	—	—	0	0	—	—	0	1
	—	—	1	1	—	—	1	1
8	—	—	1	1	—	—	1	1
	—	—	1	1	—	—	1	1
9	—	—	1	0	—	—	1	1
	—	—	1	1	—	—	1	1

35.5.3 1/3 デューティ出力波形

1/3 デューティ出力波形について示します。

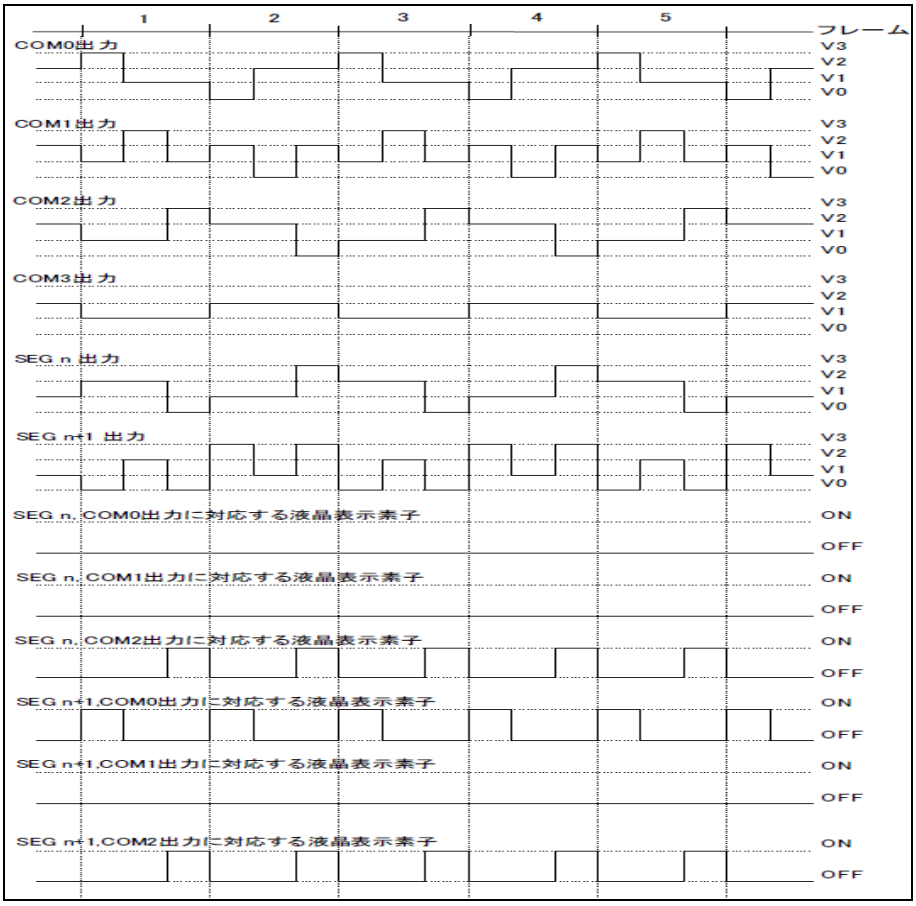
1/3 デューティ出力モードの場合、LCD の表示に使用するのは、COM0 出力, COM1 出力, COM2 出力です。COM3 出力は使用しません。

■ 1/3バイアス出力波形例

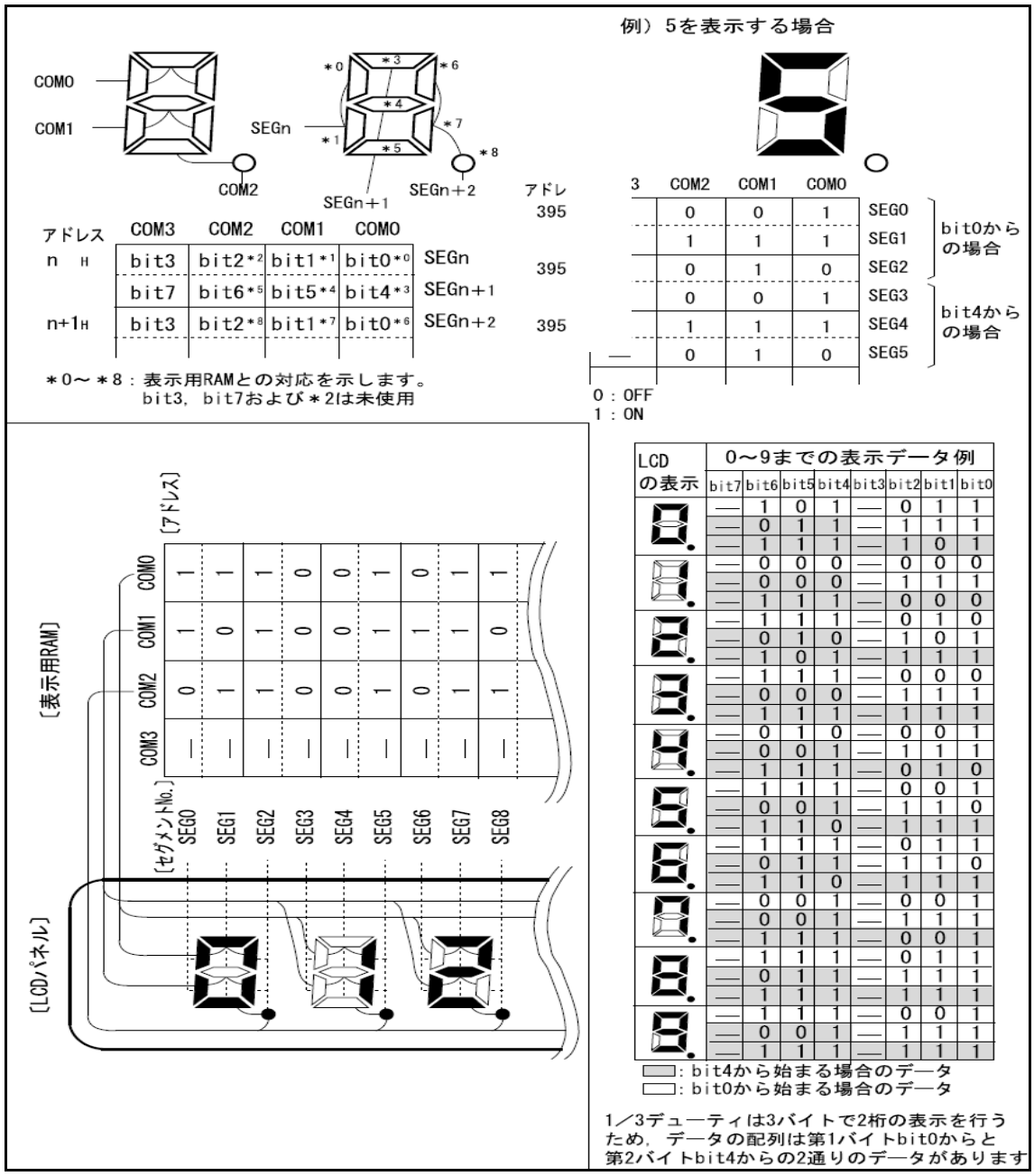
コモン出力とセグメント出力の電位差が最大となった液晶素子が点灯します。

表示用データメモリの内容例

セグメント	表示用データメモリの内容			
	COM3 出力	COM2 出力	COM1 出力	COM0 出力
SEG n 出力	−	1	0	0
SEG n+1 出力	−	1	0	1



■ LCDパネルの結線例と表示データ例 (1/3デューティ駆動方式)



35.5.4 1/4 デューティ出力波形

1/4 デューティ出力波形について示します。

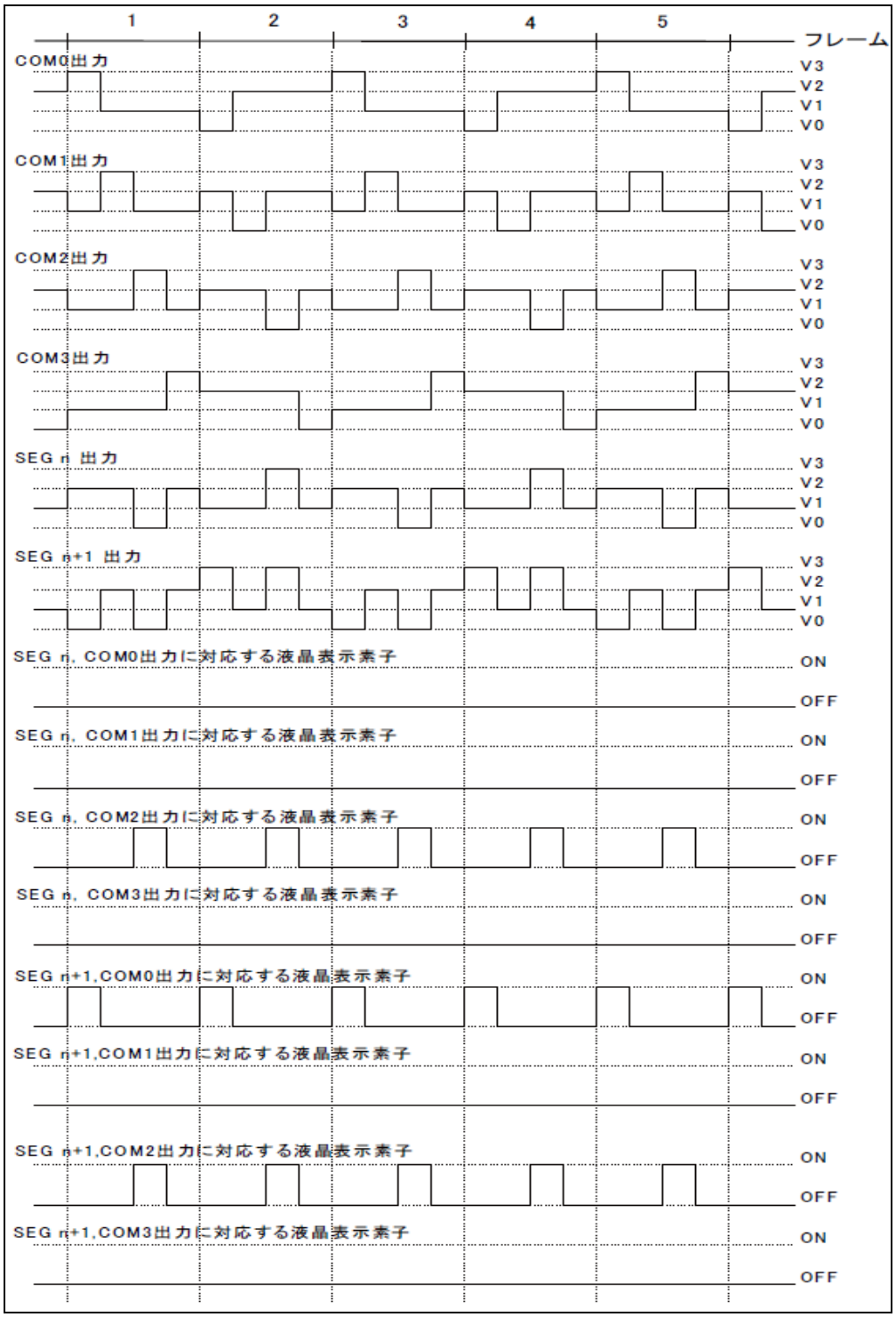
1/4 デューティ出力モードの場合は LCD の表示用に、COM0 出力, COM1 出力, COM2 出力, COM3 出力のすべてを使用します。

■ 1/4バイアス出力波形例

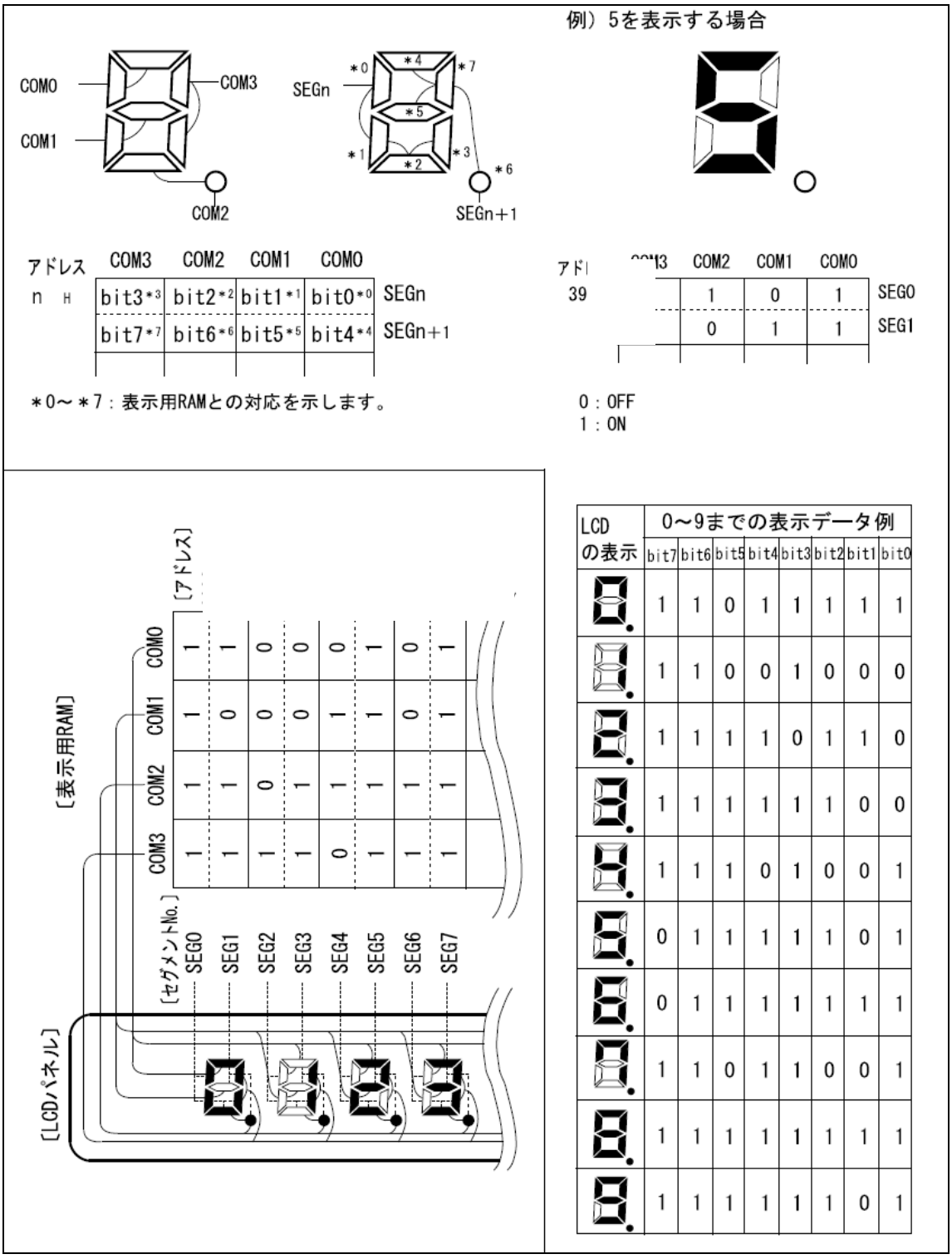
LCD では、コモン出力とセグメント出力の電位差が最大となった液晶素子が点灯します。

表示用データメモリの内容例

セグメント	表示用データメモリの内容			
	COM3 出力	COM2 出力	COM1 出力	COM0 出力
SEG n 出力	0	1	0	0
SEG n+1 出力	0	1	0	1


1
2
3
4
5
フレーム
COM0出力
COM1出力
COM2出力
COM3出力
SEG n 出力
SEG n+1 出力
V3
V2
V1
V0
SEG n, COM0出力に対応する液晶表示素子
SEG n, COM1出力に対応する液晶表示素子
SEG n, COM2出力に対応する液晶表示素子
SEG n, COM3出力に対応する液晶表示素子
SEG n+1,COM0出力に対応する液晶表示素子
SEG n+1,COM1出力に対応する液晶表示素子
SEG n+1,COM2出力に対応する液晶表示素子
SEG n+1,COM3出力に対応する液晶表示素子
ON
OFF

■ LCDパネルの結線例と表示データ例 (1/4デューティ駆動方式)



LCDの表示	0～9までの表示データ例							
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	1	1	0	1	1	1	1	1
1	1	1	0	0	1	0	0	0
2	1	1	1	1	0	1	1	0
3	1	1	1	1	1	1	0	0
4	1	1	1	0	1	0	0	1
5	0	1	1	1	1	1	0	1
6	0	1	1	1	1	1	1	1
7	1	1	0	1	1	0	0	1
8	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	0	1

35.5.5 スタティック駆動出力波形

スタティック駆動出力波形について示します。

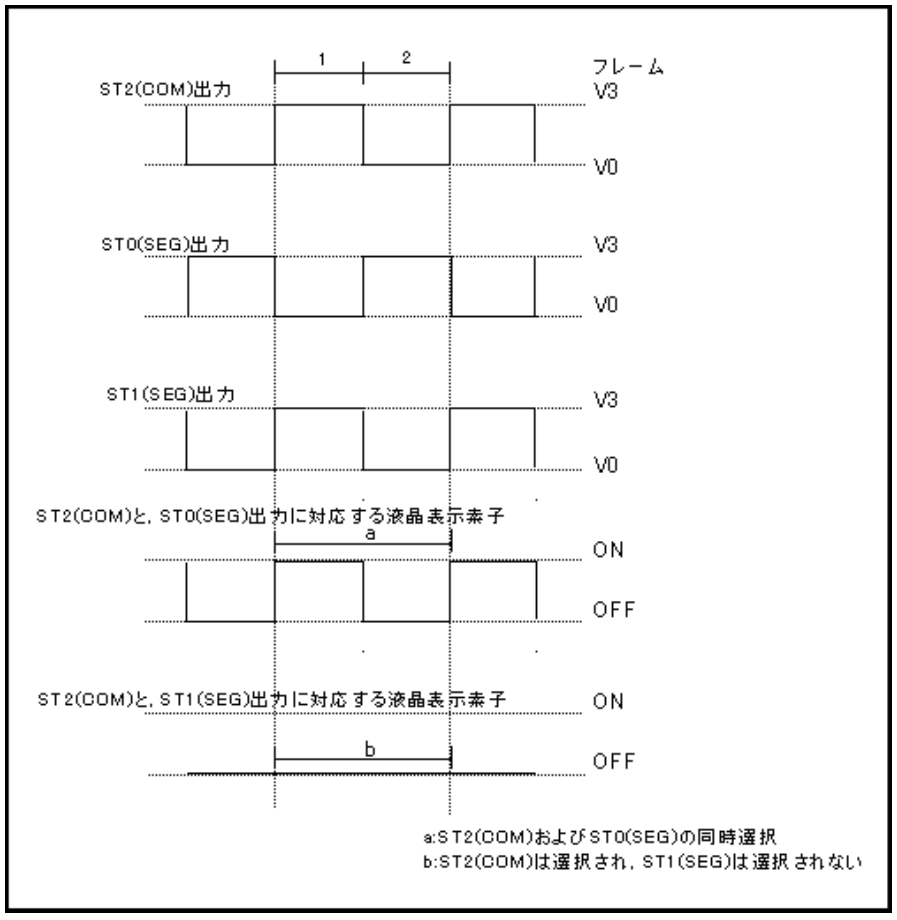
スタティック駆動出力モードの場合は LCD の表示用に、コモン／セグメント出力端子 (ST0 ～ ST8) を使用します。

■ スタティック駆動出力波形例

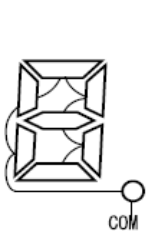
LCD では、コモン出力とセグメント出力の電位差が最大となった液晶素子が点灯します。

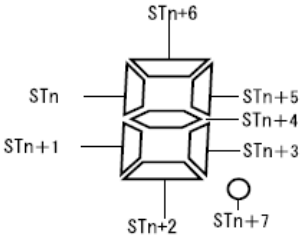
表示用データメモリの内容例 (出力端子 ST0, ST1 にスタティック SEG 出力, ST2 にスタティック COM 出力する例)

表示用データメモリの内容 (LDR0, LDR1)		
ST2 出力 LDR0[2]	**ST1 出力 LDR0[1]**	**ST0 出力 LDR0[0]**
0	0	1



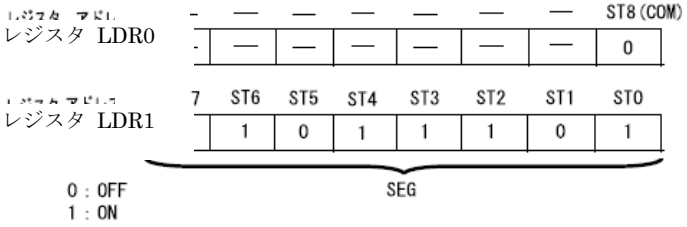
■ LCDパネルの結線例と表示データ例 (スタティック駆動方式)





例）5を表示する場合





LCD の表示	0～9までのデータ対応例(LCR1)							
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0.	1	1	1	0	1	1	1	1
1.	0	0	1	0	1	0	0	0
2.	1	1	1	1	0	1	1	0
3.	1	1	1	1	1	1	0	0
4.	1	0	1	1	1	0	0	1
5.	1	1	0	1	1	1	0	1
6.	1	1	0	1	1	1	1	1
7.	1	1	1	0	1	0	0	1
8.	1	1	1	1	1	1	1	1
9.	1	1	1	1	1	1	0	1

35.6 設定

LCD コントローラの設定について示します。

＜デューティ駆動＞

LCD を使うために必要な設定

設定	設定レジスタ	設定方法
事前の設定	LCD 制御レジスタ 1 (LCR1) コモン端子切り替えレジスタ (LCDCMR)	35.4.3, 35.4.4 参照
分割抵抗の設定	LCD 制御レジスタ 0 (LCR0)	35.7.9, 35.7.11 参照
ポート設定	ポート機能レジスタ (PFR)	35.7.1 参照
表示データの設定	表示用データメモリ (VRAM)	35.7.2 参照
フレーム周期の設定	LCD 制御レジスタ 0 (LCR0)	35.7.3 参照
デューティ選択 (起動)		35.7.5 参照
表示選択		35.7.7 参照

LCD の表示を消すのに必要な設定

設定	設定レジスタ	設定方法
非表示選択	LCD 制御レジスタ 0 (LCR0)	35.7.7 参照

LCD の動作を停止させるのに必要な設定

設定	設定レジスタ	設定方法
LCD の動作停止	LCD 制御レジスタ 0 (LCR0)	35.7.6 参照

時計モード中に LCD を表示させるのに必要な設定

設定	設定レジスタ	設定方法
時計モード中の表示選択	LCD 制御レジスタ 0 (LCR0)	35.7.8 参照
時計モードへの遷移	『消費電力制御』の章を参照してください。	−

＜スタティック駆動＞

LCD を使うために必要な設定

設定	設定レジスタ	設定方法
事前の設定	LCD 制御レジスタ 1 (LCR1) コモン端子切り替えレジスタ (LCDCMR)	35.4.3 参照 35.4.4 参照
ポート設定	端子をペリフェラル出力に設定 『I/O ポート』の章を参照してください。	
	LCD 制御レジスタ (LCRS)	35.4.5 参照
表示データの設定	表示用データメモリ (LDR0,LDR1)	35.4.6 参照
フレーム周期の設定	LCD 制御レジスタ(LCRS)	35.7.3 参照

LCD の動作を停止させるのに必要な設定

設定	設定レジスタ	設定方法
LCD の動作停止	LCD 制御レジスタ (LCRS)	35.7.6 参照

時計モード中に LCD を表示させるのに必要な設定

設定	設定レジスタ	設定方法
時計モード中の表示選択	LCD 制御レジスタ (LCRS)	35.7.8 参照
時計モードへの遷移	『消費電力制御』の章を参照してください。	―

35.7 Q&A

LCD コントローラの Q&A について示します。

35.7.1. 端子を COM 出力端子, SEG 出力端子にするには?
35.7.2. VRAM の設定方法は?
35.7.3. フレーム周期を設定するには?
35.7.4. バイアスを設定するには?
35.7.5. デューティを設定するには?
35.7.6. LCD 動作開始／動作停止を制御するには?
35.7.7. 表示を出す／消すには?
35.7.8. 時計モード中に表示するには?
35.7.9. 分割抵抗を内部か外部か選択するには?
35.7.10. V3 電圧の端子を選択するには?
35.7.11. 分割抵抗を内部か外部か選択するには?
35.7.12. 内部分割抵抗使用時に, 輝度調整をするには?
35.7.13. 外部分割抵抗で、LCD 停止時に電流を遮断させるには?
35.7.14. スタティック駆動(ST0～ST8)の LCD を表示／非表示にするには?

35.7.1 端子を COM 出力端子, SEG 出力端子にするには?

端子を COM 出力端子、SEG 出力端子に設定することについて示します。

COM 出力, SEG 出力設定

ソフトウェア設定でポートを COM, SEG 出力に切替え可能です。

COM 出力, SEG 出力する端子をペリフェラル出力に設定してください。端子設定については『I/O ポート』の章を参照してください。

＜デューティ駆動＞

<table>
<tr><th>端子</th><th>端子設定方法</th><th>PFR レジスタ・ビット番号</th></tr>
<tr><td>V0</td><td rowspan="4">端子を LCDC V0/V1/V2/V3(LCDC 基準電源入力)に設定してください。
『I/O ポート』の章を参照してください。</td><td rowspan="17">『I/O ポート』の章を参照してください。</td></tr>
<tr><td>V1</td></tr>
<tr><td>V2</td></tr>
<tr><td>V3</td></tr>
<tr><td>COM0</td><td rowspan="13">端子をペリフェラル出力に設定してください。
『I/O ポート』の章を参照してください。</td></tr>
<tr><td>COM1</td></tr>
<tr><td>COM2</td></tr>
<tr><td>COM3</td></tr>
<tr><td>SEG0</td></tr>
<tr><td>SEG1</td></tr>
<tr><td>SEG2</td></tr>
<tr><td>SEG3</td></tr>
<tr><td>SEG4</td></tr>
<tr><td>SEG5</td></tr>
<tr><td>SEG6</td></tr>
<tr><td>SEG7</td></tr>
<tr><td>SEG8</td></tr>
<tr><td>SEG9</td></tr>
<tr><td>SEG10</td></tr>
<tr><td>SEG11</td></tr>
</table>

端子	端子設定方法	PFR レジスタ・ビット番号
SEG12	端子をペリフェラル出力に設定してください。 『I/O ポート』の章を参照してください。	『I/O ポート』の章を参照してください。
SEG13		
SEG14		
SEG15		
SEG16		
SEG17		
SEG18		
SEG19		
SEG20		
SEG21		
SEG22		
SEG23/ST0		
SEG24/ST1		
SEG25/ST2		
SEG26/ST3		
SEG27/ST4		
SEG28/ST5		
SEG29/ST6		
SEG30/ST7		
SEG31/ST8		

＜スタティック駆動＞

COM 出力, SEG 出力設定

ソフトウェア設定でポートを COM, SEG 出力に切替え可能です。

スタティック駆動させる端子をペリフェラル出力に設定してください。端子設定については『I/O ポート』の章を参照してください。

端子	端子設定方法	PFR レジスタ・ビット番号
V0	端子を LCDC V0/V3(LCDC 基準電源入力)に設定してください。 『I/O ポート』の章を参照してください。	『I/O ポート』の章を参照してください。
V3		
SEG23/ST0	端子をペリフェラル出力に設定してください。 『I/O ポート』の章を参照してください。	
SEG24/ST1		
SEG25/ST2		
SEG26/ST3		
SEG27/ST4		
SEG28/ST5		
SEG29/ST6		
SEG30/ST7		
SEG31/ST8		

35.7.2 VRAM の設定方法は?

VRAM の設定方法について示します。

＜デューティ駆動＞

端子と VRAM(n)のビット位置のマトリックスを以下に示します。(n=0～15)

1/2 デューティ

端子	COM1	COM0
SEG 2n	bit1	bit0
SEG 2n+1	bit5	bit4

1/3 デューティ

端子	COM2	COM1	COM0
SEG 2n	bit2	bit1	bit0
SEG 2n+1	bit6	bit5	bit4

1/4 デューティ

端子	COM3	COM2	COM1	COM0
SEG 2n	bit3	bit2	bit1	bit0
SEG 2n+1	bit7	bit6	bit5	bit4

(該当しない端子からは、非選択の波形を出力します。)

例：1/4 デューティ

VRAMn の bit6 を"1"にすると、COM2 の SEGn+1 に選択波形を出力します。

"0"を設定した場所は非選択波形を端子に出力します。

35.7.3　フレーム周期を設定するには?

フレーム周期の設定について示します。

＜デューティ駆動＞

フレーム周期は、フレーム周期ビット(LCR0:FP[1:0])で行います。以下の設定ができます。

フレーム周期(メインクロック選択時)	選択値
	フレーム周期ビット(FP[1:0])
(2^{11} × N)/メインクロック周波数	"00"を設定する
(2^{12} × N)/メインクロック周波数	"01"を設定する
(2^{13} × N)/メインクロック周波数	"10"を設定する
(2^{14} × N)/メインクロック周波数	"11"を設定する

N (時分割数) ＝MS[1:0]の値＋"1"

フレーム周期(サブクロック選択時)	選択値
	フレーム周期ビット(FP[1:0])
(2^{3} × N)/サブクロック周波数	"00"を設定する
(2^{4} × N)/サブクロック周波数	"01"を設定する
(2^{5} × N)/サブクロック周波数	"10"を設定する
(2^{6} × N)/サブクロック周波数	"11"を設定する

N (時分割数) ＝ MS[1:0]の値＋"1"

＜スタティック駆動＞

フレーム周期は, フレーム周期ビット(LCRS:FPS[1:0])で行います。以下の設定ができます。

フレーム周期	選択値
	フレーム周期ビット(FPS[1:0])
(2^{11} × 4)/メインクロック(F_{CL})	"00"を設定する
(2^{12} × 4)/メインクロック(F_{CL})	"01"を設定する
(2^{13} × 4)/メインクロック(F_{CL})	"10"を設定する
(2^{14} × 4) /メインクロック(F_{CL})	"11"を設定する

35.7.4　バイアスを設定するには?

バイアスの設定について示します。

＜デューティ駆動＞

バイアス選択ビット (LCDCMR:DTCH) で行ってください。

バイアス	バイアス選択ビット (DTCH)
1/3 バイアスにするには	"0"にする
1/2 バイアスにするには	"1"にする

35.7.5　デューティを設定するには?

デューティの設定について示します。

＜デューティ駆動＞

表示モード選択ビット (LCR0:MS[1:0]) で行ってください。

制御内容	表示モード選択ビット (MS[1:0])	N (時分割数)
LCD 動作停止(端子出力"L")	"00"にする	―
1/2 デューティ出力モードにするには	"01"にする	2
1/3 デューティ出力モードにするには	"10"にする	3
1/4 デューティ出力モードにするには	"11"にする	4

表示モード選択ビットは動作開始／停止の制御ビットでもあります。

35.7.6　LCD 動作開始／動作停止を制御するには?

LCD 動作開始／動作停止の制御について示します。

＜デューティ駆動＞

表示モード選択ビット(LCR0:MS[1:0])で動作の開始／停止が制御できます。

『35.7.5 デューティを設定するには?』を参照してください。

＜スタティック駆動＞

表示モード選択ビット(LCRS:LCS[3:0])で動作の開始／停止が制御できます。

35.7.7　表示を出す／消すには?

LCD の表示・非表示について示します。

＜デューティ駆動＞

以下の 2 つの方法があります。

■　ブランキング選択ビット (LCR0:BK) で行う場合

制御内容	ブランキング選択ビット(BK)
LCD 表示を出す場合には?	"0"にする
LCD 表示を消す場合には? (セグメント端子に非選択波形を出力)	"1"にする

■　表示モード選択ビット (LCR0:MS[1:0]) で動作を停止させて表示を消す場合

制御内容	表示モード選択ビット (MS[1:0])
LCD 動作停止 (コモン端子, セグメント端子から"L"出力)	"00"を設定する

35.7.8　時計モード中に表示するには?

時計モード中の表示について示します。

＜デューティ駆動＞

時計モード動作許可ビット(LCR0:LCEN)で行ってください。

制御内容	時計モード動作許可ビット (LCEN)
時計モードで LCD 表示をしない場合は	"0"にする
時計モードで LCD 表示を行う場合は	"1"にする

＜スタティック駆動＞

時計モード動作許可ビット(LCRS:LCSEN)で行ってください。

制御内容	時計モード動作許可ビット (LCSEN)
時計モードで LCD 表示をしない場合は	"0"にする
時計モードで LCD 表示を行う場合は	"1"にする

35.7.9　分割抵抗を内部か外部か選択するには?

分割抵抗を内部か外部かの選択について示します。

＜デューティ駆動＞

LCD 駆動電源制御ビット(LCR0:VSEL)で行ってください。

制御内容	LCD 駆動電源制御ビット(VSEL)
外部分割抵抗を使用する場合には (内部分割抵抗を切断)	"0"にする
内部分割抵抗を使用する場合には (内部分割抵抗を接続)	"1"にする

35.7.10 V3 電圧の端子を選択するには?

V_3 電圧の端子の選択について示します。

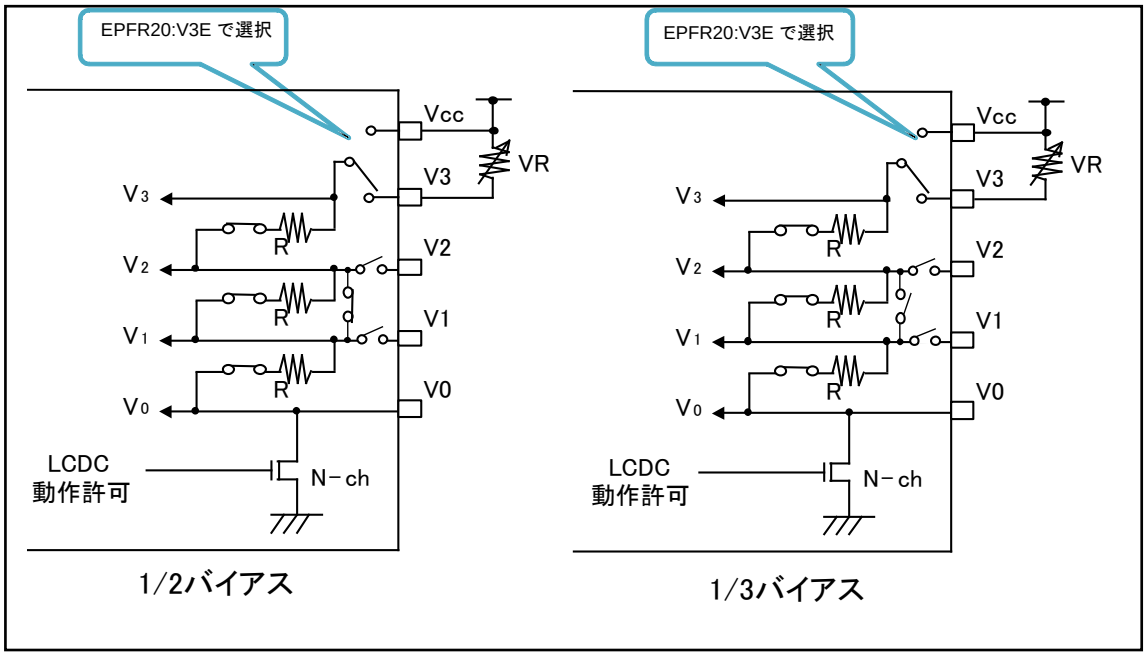
EPFR20:V3E ビットを設定します。詳細は『I/O ポート』の章を参照してください。

35.7.11 分割抵抗を内部か外部か選択するには?

分割抵抗の内部および外部の選択について示します。

● 内部分割抵抗のとき

V3 端子を V_3 電圧として使用する場合は、内部分割抵抗を使用した場合でも、Vcc-V3 間には、外部抵抗を接続してください。



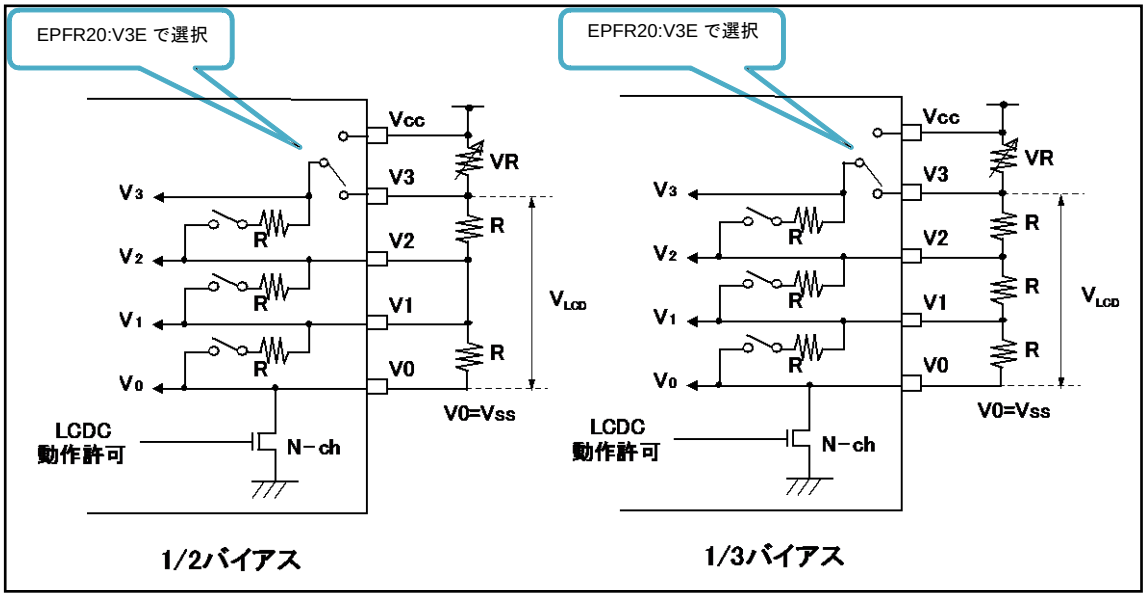
● 外部分割抵抗のとき

LCD 駆動用電圧は、LCD 駆動用電源端子(V0～V3)に外部分割抵抗を接続して設定します。

LCD 駆動電圧の設定

	V3	V2	V1	V0
1/2 バイアス	VLCD	1/2VLCD	1/2VLCD	VSS
1/3 バイアス	VLCD	2/3VLCD	1/3VLCD	VSS

V_0～V_3: V0～V3 端子の電圧
V_{LCD}: LCD の動作電圧



内部分割抵抗の影響を避けるには、LCD 駆動電源制御ビット (LCR0:VSEL)を"0"に設定して、内部分割抵抗を切断する必要があります。

35.7.12 内部分割抵抗使用時に，輝度調整をするには?

内部分割抵抗使用時の輝度調整について示します。

内部分割抵抗を使用して希望する輝度が得られない場合には、外部(Vcc～V3 端子間)に VR(可変抵抗)を接続して V3 の電圧を調整してください。

35.7.13 外部分割抵抗で、LCD 停止時に電流を遮断させるには?

外部分割抵抗で、LCD 停止時に電流の遮断について示します。

V0 端子は内部でトランジスタを介して Vss(GND)へ接続しています。このため、外部分割抵抗を使用する場合、外部分割抵抗の Vss 側を V0 端子に接続することで、LCD コントローラ停止時に流れる電流を遮断できます。表示モード選択ビット (MS[1:0]＝"00")で電流を遮断してください。

35.7.14 スタティック駆動(ST0～ST8)の LCD を表示／非表示にするには?

スタティック駆動(ST0～ST8)の LCD の表示／非表示について示します。

リセット後、スタティック駆動(ST0～ST8)の LCD を非表示させる場合は LDR0:ST8="0" かつ LDR1:ST[7:0]="00000000" に設定し、スタティック駆動選択ポート (LCS[3:0])の設定をしてください。

スタティック駆動(ST0～ST8)の端子からは、同電位のパルスが出力されます。

スタティック駆動(ST0～ST8)の LCD を非表示から表示させる場合は、(LCS[3:0])の設定は変更せずに、LDR0:ST8 設定 かつ LDR1:ST[7:0]設定をしてください。

スタティック駆動(ST0～ST8)の LCD を表示から非表示させる場合は、スタティック駆動選択ポート(LCS[3:0])は変更せずに、LDR0:ST8="0" かつ LDR1:ST[7:0]="00000000"に設定してください。

スタティック駆動(ST0～ST8)の端子からは、同電位のパルスが出力されます。

35.8 サンプルプログラム

サンプルプログラムについて示します。

設定手順 1

1/2デューティ駆動方式でLCDに "0 1 2 3 "と4桁の数字を表示させる

初期設定 (LCDC)

＜初期設定＞

1. ・ポート　　レジスタ名

ポートのCOM, SEG出力設定	『I/Oポート』の章を参照してください

2. ・VRAMの設定　　レジスタ名

VRAMの設定	VRAM00-VRAM07

3. ・制御レジスタの設定　　レジスタ名

固定値	LCR1
バイアス設定	LCDCMR
制御レジスタの設定	LCR0
	. LCEN
	. VSEL
	. BK
	. MS[1:0]
	. FP[1:0]

＜その他＞

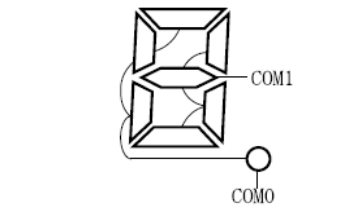
(注意事項)

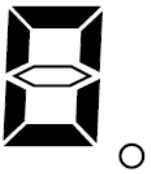
事前にクロック関連の設定および、__set_il(数値)の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

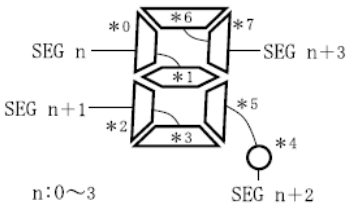
プログラム 1

```
void LCD_sample_1(void)
{
    LCDC_initial();
}

void LCDC_initial(void)
{
    PORT_SETTING_LCDC_OUT();        /* LCDコントローラの端子をペリフェラル */
                                    /* 出力に設定してください。 */

    IO_VRAM00 = 0x31;
    IO_VRAM01 = 0x32;
    IO_VRAM02 = 0x00;
    IO_VRAM03 = 0x22;
    IO_VRAM04 = 0x32;
    IO_VRAM05 = 0x30;
    IO_VRAM06 = 0x22;
    IO_VRAM07 = 0x32;

    IO_LCR1.byte = 0xFF;        /* FFに設定してください*/
    IO_LCDCMR. byte = 0x8F;     /* 1/2バイアス*/
    IO_LCR0.byte = 0x04;        /* 設定値=0000_0100 */
                                /* bit7 = 0 クロック選択　メインクロック */
                                /* bit6 = 0 LCEN 時計モードで表示を停止 */
                                /* bit5 = 0 VSEL 内部分割抵抗を切断 */
                                /* bit4 = 0 BK ブランキング選択ビット */
                                /* bit3-2 = 01 MS[1:0]　　1/2 dutyモード */
                                /* bit1-0 = 00 FP[1:0] */
}
```

例）0を表示する例と値（かっこ内）







COM3	COM2	COM1	COM0	
—	—	*1(0)	*0(1)	SEG0
—	—	*3(1)	*2(1)	SEG1
—	—	*5(1)	*4(0)	SEG2
—	—	*7(1)	*6(1)	SEG3

設定手順 2

1/3デューティ駆動方式でLCDに "0 1 2 3 "と4桁の数字を表示させる

初期設定 (LCDC)

<初期設定>

1. ・ポート　　レジスタ名

ポートのCOM, SEG出力設定	『I/Oポート』の章を参照してください

2. ・VRAMの設定　　レジスタ名

VRAMの設定	VRAM00-VRAM05

3. ・制御レジスタの設定　　レジスタ名

固定値	LCR1
バイアス設定	LCDCMR
制御レジスタの設定	LCR0
	. LCEN
	. VSEL
	. BK
	. MS[1:0]
	. FP[1:0]

<その他>

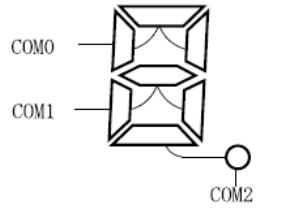
(注意事項)

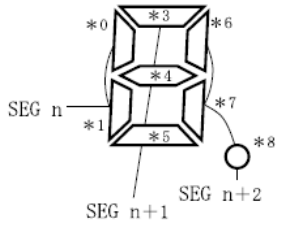
事前にクロック関連の設定および、__set_il(数値)の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

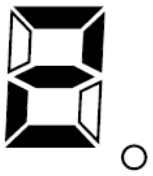
プログラム 2

```
void LCD_sample_2(void)
{
        LCDC_initial();
}

void LCDC_initial(void)
{
        PORT_SETTING_LCDC_OUT();            /* LCDコントローラの端子をペリフェラル */
                                            /* 出力に設定してください。 */

        IO_VRAM00 = 0x53;
        IO_VRAM01 = 0x03;
        IO_VRAM02 = 0x30;
        IO_VRAM03 = 0x72;
        IO_VRAM04 = 0x01;
        IO_VRAM05 = 0x37;

        IO_LCR1.byte = 0xFF;        /* FFに設定してください*/
        IO_LCDCMR. byte = 0x0F;     /* 1/3バイアス*/
        IO_LCR0.byte = 0x08;        /* 設定値=0000_1000 */
                                    /* bit7 = 0 クロック選択　メインクロック */
                                    /* bit6 = 0 LCEN 時計モードで表示を停止 */
                                    /* bit5 = 0 VSEL 内部分割抵抗を切断 */
                                    /* bit4 = 0 BK ブランキング選択ビット */
                                    /* bit3-2 = 10 MS[1:0] 1/3 dutyモード */
                                    /* bit1-0 = 00 FP[1:0] */
}
```

例）2を表示する例と値（かっこ内）







COM3	COM2	COM1	COM0		
—	(-)	*1(1)	*0(0)	SEG0	1桁目
—	*5(1)	*4(1)	*3(1)	SEG1	
—	*8(0)	*7(0)	*6(1)	SEG2	
—	-	1	0	SEG3	2桁目
—	1	1	1	SEG4	
—	0	0	1	SEG5	

設定手順 3

1/4デューティ駆動方式でLCDに "0 1 2 3 "と4桁の数字を表示させる

初期設定 (LCD)

<初期設定>

1. ・ポート　　レジスタ名

ポートのCOM, SEG出力設定	『I/Oポート』の章を参照してください

2. ・VRAMの設定　　レジスタ名

VRAMの設定	VRAM00-VRAM03

3. ・制御レジスタの設定　　レジスタ名．ビット名

固定値	LCR1
バイアス設定	LCDCMR
制御レジスタの設定	LCR0
	. LCEN
	. VSEL
	. BK
	. MS[1:0]
	. FP[1:0]

<その他>

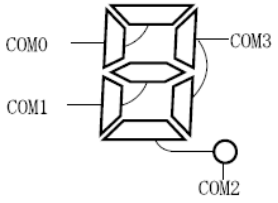
（注意事項）

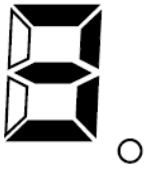
事前にクロック関連の設定および、__set_il(数値)の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

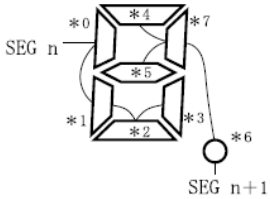
プログラム 3

```
void LCD_sample_3(void)
{
      LCD_initial();
}

void lcdc_initial(void)
{
      PORT_SETTING_LCDC_OUT();          /* LCDコントローラの端子をペリフェラル */
                                        /* 出力に設定してください。 */

      IO_VRAM00 = 0x9F;
      IO_VRAM01 = 0x88;
      IO_VRAM02 = 0xB6;
      IO_VRAM03 = 0xBC;

      IO_LCR1.byte = 0xFF;        /* FFに設定してください*/
      IO_LCDCMR. byte = 0x0F;     /* 1/3バイアス*/
      IO_LCR0.byte = 0x0C;        /* 設定値=0000_1100 */
                                  /* bit7 = 0 クロック選択　メインクロック */
                                  /* bit6 = 0 LCEN　時計モードで表示を停止 */
                                  /* bit5 = 0 VSEL 内部分割抵抗を切断 */
                                  /* bit4 = 0 BK ブランキング選択ビット */
                                  /* bit3-2 = 11 MS[1:0] 1/4 dutyモード */
                                  /* bit1-0 = 00 FP[1:0] */
}
```

例）3を表示する場合の値（かっこ内）







COM3	COM2	COM1	COM0	
*3(1)	*2(1)	*1(0)	*0(0)	SEG0
*7(1)	*6(0)	*5(1)	*4(1)	SEG1

設定手順 4

スタティック駆動方式でLCDに1桁の数字を表示させる

初期設定 (LCD)

＜初期設定＞

1. ・ポート　　レジスタ名

ポートのCOM, SEG出力設定	『I/Oポート』の章を参照してください

2. ・データの設定　　レジスタ名

データの設定	LDR0 LDR1

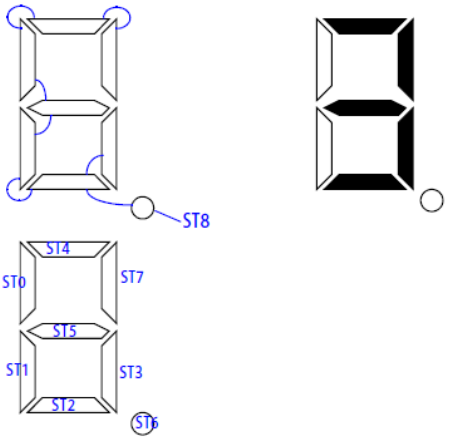
3. ・制御レジスタの設定　　レジスタ名．ビット名

固定値	LCR1
バイアス設定	LCDCMR
制御レジスタの設定	LCRS
	．LCSEN
	．LCS[3:0]
	．FPS[1:0]

プログラム 4

```
void LCD_sample_4(void)
{
      LCD_initial();
}

void lcdc_initial(void)
{
      PORT_SETTING_LCDC_OUT();          /* LCDコントローラの端子をペリフェラル */
                                        /* 出力に設定してください。 */

      IO_LDR0.byte = 0x00;
      IO_LDR1.byte = 0xBC;



      IO_LCR1.byte = 0xFF;        /* FFに設定してください*/
      IO_LCDCMR. byte = 0x0F;     /* 1/3バイアス*/
      IO_LCRS.byte = 0x20;        /* 設定値=0010_0000 */
                                  /* bit7 = 0 クロック選択　メインクロック */
                                  /* bit6 = 0 LCSEN　時計モードで非表示 */
                                  /* bit5-2 = 1000　LCS[3:0]　ST0～ST8 */
                                  /* bit1-0 = 00 FPS[1:0]　フレーム周期設定 */


}
```

例）3 を表示する場合の値



LDR0・LDR1 レジスタ

ST 8	ST 7	ST 6	ST 5		
0	1	0	1		
ST 4	**ST 3**	**ST 2**	**ST 1**	**ST 0**	
1	1	1	0	0	

35.9 注意事項

注意事項について示します。

- 外部に接続する外部分割抵抗は、使用する LCD によって異なりますので適切な値の抵抗を接続ください。
- 表示モードが 1/2 デューティの場合の COM2, COM3 端子と, 1/3 デューティの場合の COM3 端子からは、非選択レベルの波形を出力します。
- LCD 駆動電源制御(VSEL), デューティ選択(MS[1:0]), フレーム周期選択(FP[1:0]) などの設定が適切でない場合、LCD は正しく表示しません。
- LCD や汎用ポートを使用しない場合、V3～V0 端子にプルアップまたはプルダウン抵抗を接続してください。
- LCS[3:0]="0000"以外の設定をすると、スタティック駆動が有効になります。LCS[3:0]="0000" に設定すると, デューティ駆動が有効になります。
- スタティック駆動(ST0～ST8)の LCD を表示／非表示にする設定について
  リセット後、スタティック駆動(ST0～ST8)の LCD を非表示させる場合は LDR0:ST8="0" かつ LDR1:ST[7:0]="00000000" に設定し、スタティック駆動選択ポート (LCS[3:0])の設定をしてください。スタティック駆動(ST0～ST8)の端子からは、同電位のパルスが出力されます。

スタティック駆動(ST0～ST8)の LCD を非表示から表示させる場合は、スタティック駆動選択ポート(LCS[3:0]) は変更せずに、LDR0:ST8 と LDR1:ST[7:0]の設定をしてください。

スタティック駆動(ST0～ST8)の LCD を表示から非表示させる場合は、スタティック駆動選択ポート(LCS[3:0]) は変更せずに、LDR0:ST8="0" かつ LDR1:ST[7:0]="00000000"に設定してください。

スタティック駆動(ST0～ST8)の端子からは, 同電位のパルスが出力されます。

- LCD 使用時(スタティック駆動, デューティ駆動)には下記の設定をしてください。
  LDR1:ST[7:0]="11111111"
  LCDCMR:bit[3:0]="1111"

- 時計モード (電源遮断) 時も時計モード (電源遮断なし) 時同様に動作継続可能です。時計モード時の動作設定は「35.7.8. 時計モード中に表示するには?」を参照してください。また、時計モード (電源遮断) からのウェイクアップによるリセットでは LCD コントローラは初期化されません。動作継続します。

36. 外部バスインタフェース

外部バスインタフェースについて説明します。

36.1 概要

外部バスインタフェースの概要について説明します。

外部バスインタフェースの各機能について解説します。

36.2 特長

外部バスインタフェースの特長について説明します。

- 最大 22 ビット長 (4M バイト空間) のアドレスを出力。(ACR0～ACR3:ADTY ビットの設定によっては最下位 1 ビットを固定扱いにしてその分上位ビットを 1 ビット拡張して、8M バイト空間に拡張できます。)
- アドレス・データスプリットバス対応
  - 非同期メモリ接続可能。
- アドレス・データマルチプレックスバス対応。
- 独立した 4 個チップセレクト領域 (以下 CS 領域とする) を設定でき、それぞれに対応したチップセレクト出力が可能。
- 各 CS 領域のサイズは 64K バイト～4M バイトの範囲で 16 とおり設定可能。
- 各 CS 領域は外バス領域内の任意の位置に設定可能。
- CS 領域ごとに以下の機能が独立に設定可能。
  - 有効・無効。
  - データバス幅 (8 ビット/16 ビット)
  - 書込み禁止 (読み出し専用) 設定
  - バイトオーダ
    CS0 領域　　：ビックエンディアン
    CS0 領域以外：ビック/リトルエンディアンサポート
  - アドレスシフト出力モード
- CS 領域ごとにバスタイプが選択可能。
- アドレス・データスプリットバス
- アドレス・データマルチプレックスバス
- タイプ 0(バイトライトストローブ信号出力)
- CS 領域ごとに以下のタイミング設定が可能
  - リード/ライトアクセス共通
    アドレス -> CS 信号セットアップサイクル数
  - アドレスストローブ信号出力サイクル数
  - 外部レディ入力によるリード/ライトバスサイクルの延長
  - リードアクセス
  - リードアクセス自動ウェイト。
  - CS 信号 -> リードストローブ信号セットアップサイクル数
  - リードストローブ信号 -> CS 信号ホールドサイクル数
  - リードアクセス -> ライトアクセス間へのアイドルサイクル挿入
  - ライトアクセス
  - ライトアクセス自動ウェイト
  - CS 信号 -> ライトストローブ信号セットアップサイクル数
  - ライトストローブ信号 -> CS 信号ホールドサイクル数
- ライトリカバリーサイクル挿入
  - アドレス・データマルチプレックスバス
  - アドレス出力サイクル数

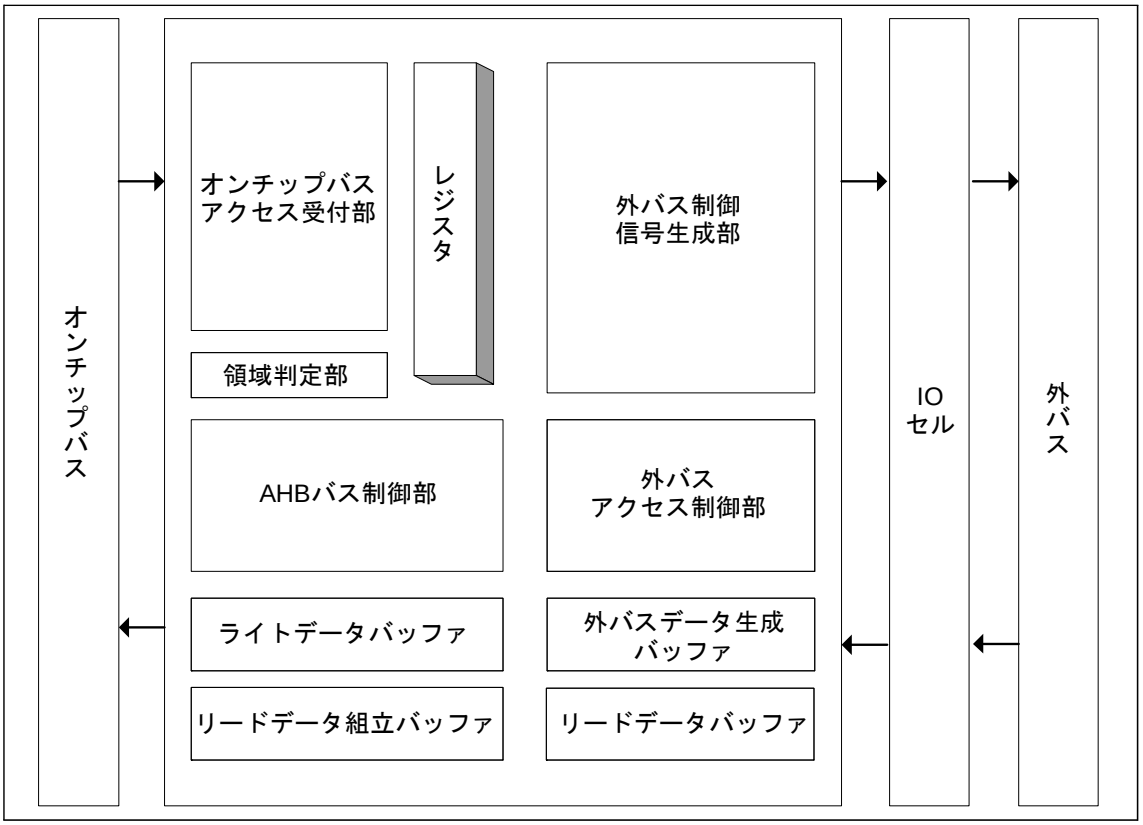
36.3 構成

外部バスインタフェースの構成を示します。

■ 外部バスインタフェースのブロックダイヤグラム

外部バスインタフェースのブロックダイヤグラムを、図 36-1 に示します。

図 36-1 外部バスインタフェースのブロックダイヤグラム



36.4 レジスタ

外部バスインタフェースのレジスタについて説明します。

■ レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0600	ASR0				CS0 領域レジスタ
0x0604	ASR1				CS1 領域レジスタ
0x0608	ASR2				CS2 領域レジスタ
0x060C	ASR3				CS3 領域レジスタ
0x0640	ACR0				CS0 バス設定レジスタ
0x0644	ACR1				CS1 バス設定レジスタ
0x0648	ACR2				CS2 バス設定レジスタ
0x064C	ACR3				CS3 バス設定レジスタ
0x0680	AWR0				CS0 ウェイトレジスタ
0x0684	AWR1				CS1 ウェイトレジスタ
0x0688	AWR2				CS2 ウェイトレジスタ
0x068C	AWR3				CS3 ウェイトレジスタ
0x06C0	予約 (DMAR0)				ch.0 外部 DMA 転送レジスタ (本シリーズではサポートしていません)
0x06C4	予約 (DMAR1)				ch.1 外部 DMA 転送レジスタ (本シリーズではサポートしていません)
0x06C8	予約 (DMAR2)				ch.2 外部 DMA 転送レジスタ (本シリーズではサポートしていません)
0x06CC	予約 (DMAR3)				ch.3 外部 DMA 転送レジスタ (本シリーズではサポートしていません)

36.4.1 CS 領域設定レジスタ: ASR0～ASR3 (Area Setting Register 0-3)

CS 領域設定レジスタのビット構成について示します。

CS 領域 CS0-CS3 の、領域の設定を行うレジスタです。1 つの CS 領域に対して 1 つの ASR レジスタを持ちます。CS 領域は重なり合わないように設定してください。このレジスタの設定手順は「36.5.10. CS 設定フロー」を参照してください。

■ **ASR0 ：アドレス 0600H (アクセス:ワード)**

■ **ASR1 ：アドレス 0604H (アクセス: ワード)**

■ **ASR2 ：アドレス 0608H (アクセス: ワード)**

■ **ASR3 ：アドレス 060CH (アクセス: ワード)**

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	SADR[31:24]							
初期値	*1	*1	*1	*1	*1	*1	*1	*1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	SADR[23:16]							
初期値	*1	*1	*1	*1	*1	*1	*1	*1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	*1	*1	*1	*1	*1	*1	*1	*1
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ASZ[3:0]				予約	WREN	LEDN	CSEN
初期値	*1	*1	*1	*1	*1	*1	*1	*1
属性	R/W	R/W	R/W	R/W	R0,W0	R/W	R/W*2	R/W

*1: [初期値] ASR0 0000_0000_0000_0000_0000_0000_1111_0001B
ASR0 以外 XXXX_XXXX_XXXX_XXXX_0000_0000_XXXX_0XX0B

*2: ASR0 では R0,W0

[bit31～bit16] SADR[31:16] (Start ADdRess)：CS 領域開始アドレス
SADR は、CS 領域の開始アドレスを指定します。ASR0 の初期値は"0000_0000_0000_0000", ASR0 以外の初期値は不定です。開始アドレスは、32 ビットアドレスの上位 16 ビットを設定します。CS 領域はこのレジスタに設定されたアドレスから始まる ASZ[3:0]で設定された範囲の領域となります。CS 領域のバウンダリは，本レジスタの bit7-4:ASZ[3:0]の設定にしたがって決まります。例えば、ASZ[3:0]=0100 とし 1M バイトの CS 領域を設定した場合は、SADR の bit[19:16]は無視され、SADR[31:20]のみが意味を持ちます。

＜注意事項＞
CS 領域に割当て可能なアドレス範囲は品種により異なります。『付録』を参照してください。

[bit15～bit8] 予約
必ず"0"を書き込んでください。

[bit7～bit4] ASZ[3:0] (Area SiZe) :CS 領域サイズ
CS 領域のサイズを以下のとおり設定します。また、その際に実際にアドレスと比較される SADR のビット位置を示します。

ASZ[3:0]	CS 領域のサイズ	実際にアドレスと比較される SADR のビット
0000	64K バイト	SADR[31:16]
0001	128K バイト	SADR[31:17]
0010	256K バイト	SADR[31:18]
0011	512K バイト	SADR[31:19]
0100	1M バイト	SADR[31:20]
0101	2M バイト	SADR[31:21]
0110	4M バイト	SADR[31:22]
0111	8M バイト	SADR[31:23]
1000	16M バイト	SADR[31:24]
1001	32M バイト	SADR[31:25]
1010	64M バイト	SADR[31:26]
1011	128M バイト	SADR[31:27]
1100	256M バイト	SADR[31:28]
1101	512M バイト	SADR[31:29]
1110	1G バイト	SADR[31:30]
1111	2G バイト (初期値)	SADR[31]

[bit3] 予約
必ず"0"を書き込んでください。

[bit2] WREN (WRite ENable) : ライトイネーブル
CS 領域への書込みの有効・無効を設定します。

WREN	書込み有効・無効
0	書込み無効
1	書込み有効

ASR0 の初期値は"0"、それ以外の ASR の初期値は不定です。
書込み無効の領域に対する書込みが内部バスから発生した場合、そのアクセスは無視され外部アクセスを一切行いません。データ領域など書込みが必要な領域は、WREN を"1"に設定してください。

[bit1] LEDN (Little EnDiaN) : リトルエンディアン
LEDN は CS 領域のバイトオーダを設定します。
ASR0 にはこのビットはありません。常に"0"が読み出されます。

LEDN	Endian
0	ビッグエンディアン
1	リトルエンディアン

ASR0 以外の初期値は不定です。

[bit0] CSEN (Chip Select ENable) : CS 領域イネーブル
CS 領域の有効・無効を設定します。CSEN に"1"を設定することにより、ASR レジスタ, ACR レジスタ, AWR レジスタの設定にしたがい、動作を開始します。

CSEN	CS 領域の有効・無効
0	無効
1	有効

ASR0 の初期値は"1"、それ以外の ASR の初期値は"0"です。

36.4.2 CS バス設定レジスタ ：ACR0～ACR3 (Area Configuration Register 0-3)

CS バス設定レジスタのビット構成について示します。

CS 領域の、バス設定を行うレジスタです。1 つの CS 領域に対して 1 つの ACR レジスタを持ちます。このレジスタの設定手順は「36.5.10. CS 設定フロー」を参照してください。

■ **ACR0 ：アドレス 0640H (アクセス: ワード)**

■ **ACR1 ：アドレス 0644H (アクセス: ワード)**

■ **ACR2 ：アドレス 0648H (アクセス: ワード)**

■ **ACR3 ：アドレス 064CH (アクセス: ワード)**

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	予約							
初期値	*	*	*	*	*	*	*	*
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	予約							
初期値	*	*	*	*	*	*	*	*
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	*	*	*	*	*	*	*	*
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DBW[1:0]		予約		ADTY	BSTY	予約	予約
初期値	*	*	*	*	*	*	*	*
属性	R/W	R/W	R0,W0	R0,W0	R/W	R/W	RX,W0	RX,W0

* [初期値] ACR0 0000_0000_0000_0000_0000_0000_0100_0000B
ACR0 以外 0000_0000_0000_0000_0000_0000_XX00_XX0XB

[bit31～bit8] 予約
必ず"0"を書き込んでください。

[bit7, bit6] DBW[1:0] (Data Bus Width) : **データバス幅**
データバス幅を設定します。

DBW[1:0]	データバス幅	使用されるビット位置 (D31～D16)
00	8bit	D[31:24]
01	16bit	D[31:16]
10	Reserved (32bit)	―
11	Reserved (32bit)	―

本シリーズでは 32 ビットデータバス幅はサポートしていません。
ACR0 の初期値は"01"です。ACR0 以外の初期値は不定です。

[bit5, bit4] **予約**
必ず"0"を書き込んでください。

[bit3] ADTY (ADdress output TYpe) : **アドレスタイプ**
アドレス出力タイプを設定します。
ACR0 の初期値は"00", それ以外の ASR の初期値は不定です。

ADTY	説明
0	通常出力
1	16 ビット時においてアドレスを 1 ビットシフトして出力します。 詳細は「36.5.6. アドレス情報」を参照してください。

[bit2] BSTY (BuS TYpe) : **バスタイプ**
バスタイプを設定します。
ACR0 の初期値は"00", それ以外の ASR の初期値は不定です。

BSTY	説明
0	アドレス・データスプリットバス
1	アドレス・データマルチプレックスバス

[bit1, bit0] **予約**
必ず"0"を書き込んでください。

36.4.3 CS ウェイトレジスタ ：AWR0～AWR3 (Area Wait Register 0-3)

CS ウェイトレジスタのビット構成について示します。

CS 領域 CS0～CS3 の、各種ウェイトの設定を行うレジスタです。1 つの CS 領域に対して 1 つの AWR レジスタを持ちます。このレジスタの設定手順は「36.5.10. CS 設定フロー」を参照してください。

- **AWR0 ：アドレス 0680H (アクセス: ワード)**
- **AWR1 ：アドレス 0684H (アクセス: ワード)**
- **AWR2 ：アドレス 0688H (アクセス: ワード)**
- **AWR3 ：アドレス 068CH (アクセス: ワード)**

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	予約				RWT[3:0]			
初期値	*	*	*	*	*	*	*	*
属性	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W	R/W

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	WWT[3:0]				RIDL[1:0]		WRCV[1:0]	
初期値	*	*	*	*	*	*	*	*
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	CSRD[1:0]		RDCS[1:0]		CSWR[1:0]		WRCS[1:0]	
初期値	*	*	*	*	*	*	*	*
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ADCY[1:0]		ACS[1:0]		ASCY	予約	RDYE	予約
初期値	*	*	*	*	*	*	*	*
属性	R/W	R/W	R/W	R/W	R/W	R0,W0	R/W	R0,W0

* [初期値] AWR0 0000_1111_0000_0000_1111_0000_0000_0000B
AWR0 以外 0000_XXXX_XXXX_XXXX_XXXX_XXXX_XXXX_X0X0B

[bit31～bit28] 予約
必ず"0"を書き込んでください。

[bit27～bit24] RWT[3:0] (Read access auto WaiT) : **リードアクセスオートウェイト**
RWT[3:0]は、リードアクセスサイクルにおけるデータ取込み時のオートウェイトサイクル数を設定します。

RWT[3:0]	リードアクセスウェイト
0000	0 サイクル
0001	1 サイクル
0010	2 サイクル
0011	3 サイクル
～	～
1110	14 サイクル
1111	15 サイクル(AWR0 初期値)

[bit23～bit20] WWT[3:0] (Write access auto WaiT) : **ライトアクセスオートウェイト**
WWT[3:0]は、ライトアクセスサイクルにおけるオートウェイトサイクル数を設定します。

WWT[3:0]	ライトアクセスウェイト
0000	0 サイクル(AWR0 初期値)
0001	1 サイクル
0010	2 サイクル
0011	3 サイクル
～	～
1110	14 サイクル
1111	15 サイクル

[bit19, bit18] RIDL[1:0] (Read access IDLe cycle) : **リードアクセスアイドルサイクル**
RIDL[1:0]は出力オフタイムが長いデバイスからのリードデータとそれに続くアクセスに関係するデータのデータバス上での衝突を防ぐ目的で設定します。リードアクセスに連続して下記のいづれかに当てはまるアクセスが行なわれる場合、リードアクセス終了後 RIDL に設定されたアイドルサイクルが挿入されます。

- ライトアクセス
- 別の CS 領域へのアクセス
- アドレス・データマルチプレックスバスタイプに設定された CS 領域に対するアクセス

スプリットバスタイプ (ACR:BSTY=0) に設定された同一CS領域に対するリードアクセスが連続する場合は, RIDL によるアイドルサイクルの挿入は行われません。アイドルサイクル中は、すべての CS 信号をネゲートし、データ端子はハイインピーダンス状態になります。

RIDL[1:0]	リードアクセスアイドルサイクル
00	0 サイクル(AWR0 初期値)
01	1 サイクル
10	2 サイクル
11	3 サイクル

[bit17, bit16] WRCV[1:0] (Write ReCoVery cycle) : ライトリカバリサイクル
WRCV[1:0]は、ライトリカバリサイクルの設定であり、ライトアクセス後に続けてアクセスする際にその間隔に制限があるデバイスへのアクセスを制御するために設定します。ライトリカバリサイクル中はすべてのチップセレクト信号をネゲートし、ライトストローブ信号 WRnX (n=0, 1)もネゲート状態を保ちます。またこの期間内は新たなアクセスは開始されません。ライトリカバリサイクルを 1 サイクル以上に設定した場合、ライトアクセスの後には必ずライトリカバリサイクルが挿入されます。

WRCV[1:0]	ライトリカバリサイクル
00	0 サイクル(AWR0 初期値)
01	1 サイクル
10	2 サイクル
11	3 サイクル

[bit15, bit14] CSRD[1:0] (CSnX to RDX setup cycle) : CSnX to RDX セットアップサイクル
CSRD[1:0]は、リードアクセスの CSnX → RDX セットアップサイクルの設定であり、CSnX アサート後に RDX をアサートするまでの期間を設定します。

アドレス・データマルチプレックスバス (ACR:BSTY=1) 設定時は、プロトコルが正しく成立するように下記の条件を満たすよう AWR のパラメータを設定してください。

ACS + CSRD ≧ 1 かつ ACS + CSWR ≧ 1

CSRD[1:0]	CSnX → RDX セットアップ延長サイクル
00	0 サイクル
01	1 サイクル
10	2 サイクル
11	3 サイクル(AWR0 初期値)

[bit13, bit12] RDCS[1:0] (RDX to CSnX hold cycle) : RDX to CSnX ホールドサイクル
RDCS[1:0]は、リードアクセスの RDX → CSnX ホールドサイクルの設定であり、RDX ネゲート後に CSnX をネゲートするまでの期間を設定します。

RDCS[1:0]	RDX → CSnX ホールド延長サイクル
00	0 サイクル
01	1 サイクル
10	2 サイクル
11	3 サイクル(AWR0 初期値)

[bit11, bit10] CSWR[1:0] (CSnX to WRnX setup cycle) : CSnX to WRnX セットアップサイクル
CSWR[1:0]は、ライトアクセスの CSnX → WRnX セットアップサイクルの設定であり、CSnX アサート後、WRnX をアサートするまでの期間を設定します。

アドレス・データマルチプレックスバス (ACR:BSTY=1) 設定時は、プロトコルが正しく成立するように下記の条件を満たすよう AWR のパラメータを設定してください。

ACS + CSRD ≧ 1 かつ ACS + CSWR ≧ 1

CSWR[1:0]	CSnX → WRnX セットアップ延長サイクル
00	0 サイクル(AWR0 初期値)
01	1 サイクル
10	2 サイクル
11	3 サイクル

[bit9, bit8] WRCS[1:0] (WRnX to CSnX hold cycle) : WRnX to CSnX ホールドサイクル
WRCS[1:0]は、ライトアクセスの WRnX→CSnX ホールドサイクルの設定であり、WRnX ネゲート後に CSnX をネゲートするまでの期間を設定します。

WRCS[1:0]	WRnX → CSnX ホールド延長サイクル
00	0 サイクル(AWR0 初期値)
01	1 サイクル
10	2 サイクル
11	3 サイクル

[bit7, bit6] ADCY[1:0] (ADdress CYcle) : アドレス出力延長サイクル数
ADCY[1:0]は、アドレス・データマルチプレックスバスタイプ設定 CS 領域へのアクセスにおける、データバスへのアドレス出力の延長サイクル数を設定します。このビットの設定は、アドレス・データマルチプレックスバスタイプ設定時のみにおいて有効です。

ADCY を 1 以上に設定する場合には、プロトコルが正しく成立するように下記の条件を満たすよう AWR のパラメータを設定してください。

ADCY + 1 ≦ ACS + CSRD かつ ADCY + 1 ≦ ACS + CSWR

ADCY[1:0]	アドレス・データマルチプレックス時のアドレス出力延長サイクル数
00	0 サイクル(AWR0 初期値)
01	1 サイクル
10	2 サイクル
11	3 サイクル

[bit5, bit4] ACS[1:0] (A00～A21 to CSnX delay cycle) : A00～A21 to CSnX ディレイサイクル数
ACS[1:0]は、A00～A21 および ASX 出力から CSnX 出力の遅延サイクル数を設定します。CSnX のアサートに対してアドレスが一定のセットアップを必要とする場合や、連続して同一チップセレクト領域にアクセスした場合にも CSnX のエッジを必要とする場合に使用します。

ACS[1:0]	A00～A21 → CSnX 遅延サイクル数
00	0 サイクル(AWR0 初期値)
01	1 サイクル
10	2 サイクル
11	3 サイクル

[bit3] ASCY (ASX CYcle) : ASX 出力延長サイクル数
ASCY は、ASX 出力を何サイクル延長するかを設定します。ASX は最小で 1 サイクル出力されます。

ASCY	ASX 出力延長遅延サイクル数
0	0 サイクル(AWR0 初期値)
1	1 サイクル

[bit2] 予約
必ず"0"を書き込んでください。

[bit1] RDYE (RDY Enable) : RDY イネーブル
RDYE は、RDY 端子による外部からのウェイト挿入機能の有効・無効を設定します。

RDYE	RDY 端子有効
0	RDY 端子によるウェイト挿入無効(AWR0 初期値)
1	RDY 端子によるウェイト挿入有効

[bit0] 予約
必ず"0"を書き込んでください。

36.4.4 外部 DMA 転送レジスタ: DMAR0-3 (DMA transfer Register 0-3)

外部 DMA 転送レジスタのビット構成について示します。

DMA 転送用外部端子の設定を行うレジスタです。本シリーズではサポートしていません。

■ **DMAR0 ：アドレス 06C0H (アクセス: ワード)**

■ **DMAR1 ：アドレス 06C4H (アクセス: ワード)**

■ **DMAR2 ：アドレス 06C8H (アクセス: ワード)**

■ **DMAR3 ：アドレス 06CCH (アクセス: ワード)**

	bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
	予約							
初期値	*	*	*	*	*	*	*	*
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0
	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	予約							
初期値	*	*	*	*	*	*	*	*
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	*	*	*	*	*	*	*	*
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				REQL	ACKMD	ACKL	EOPL
初期値	*	*	*	*	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R/W0	R/W0	R/W0	R/W0

* [初期値] 0000_0000_0000_0000_0000_0000_0000_0000B

[bit31～bit4] 予約
必ず"0"を書き込んでください。

[bit3] REQL
書き込む場合は"0"を書き込んでください。

[bit2] ACKMD
書き込む場合は"0"を書き込んでください。

[bit1] ACKL
書き込む場合は"0"を書き込んでください。

[bit0] EOPL
書き込む場合は"0"を書き込んでください。

36.5 動作説明

外部バスインタフェースの動作について説明します。

36.5.1 外部端子表

外部端子表について示します。

本シリーズでの外部バスインタフェース用外部端子を示します。

表 36-1 外部端子表 (LQFP-144)

本シリーズでの 外部端子	本シリーズでの 端子番号	説明
SYSCLK	35	システムクロック出力
ASX	5	アドレスストローブ出力
CS0X, CS1X, CS2X, CS3X	6, 7, 38, 39	チップセレクト出力
RDX	8	リードストローブ出力
WR0X, WR1X	9, 10	ライトストローブ出力
RDY	40	バスレディ入力
D16_0～D31_0/ D16_1～D31_1	131～138, 139～143, 2～ 4/ 139～143, 2～4, 131～ 138	データ入出力、 アドレス出力 (アドレスマルチプレックス時)
A00～A21	11～19, 22～34	アドレス出力

表 36-2 外部端子表 (LQFP-208)

本シリーズでの 外部端子	本シリーズでの 端子番号*1	説明
SYSCLK	23	システムクロック出力
ASX	199	アドレスストローブ出力
CS0X, CS1X, CS2X, CS3X	200, 201, 24, 25	チップセレクト出力
RDX	202	リードストローブ出力
WR0X, WR1X	203, 204	ライトストローブ出力
RDY	26	バスレディ入力
D16_0～D31_0/ D16_1～D31_1	183 ~ 198/ 191 ~ 198, 183 ~ 190	データ入出力、 アドレス出力 (アドレスマルチプレックス時)
A00～A21	205 ~ 207, 2 ~ 20	アドレス出力

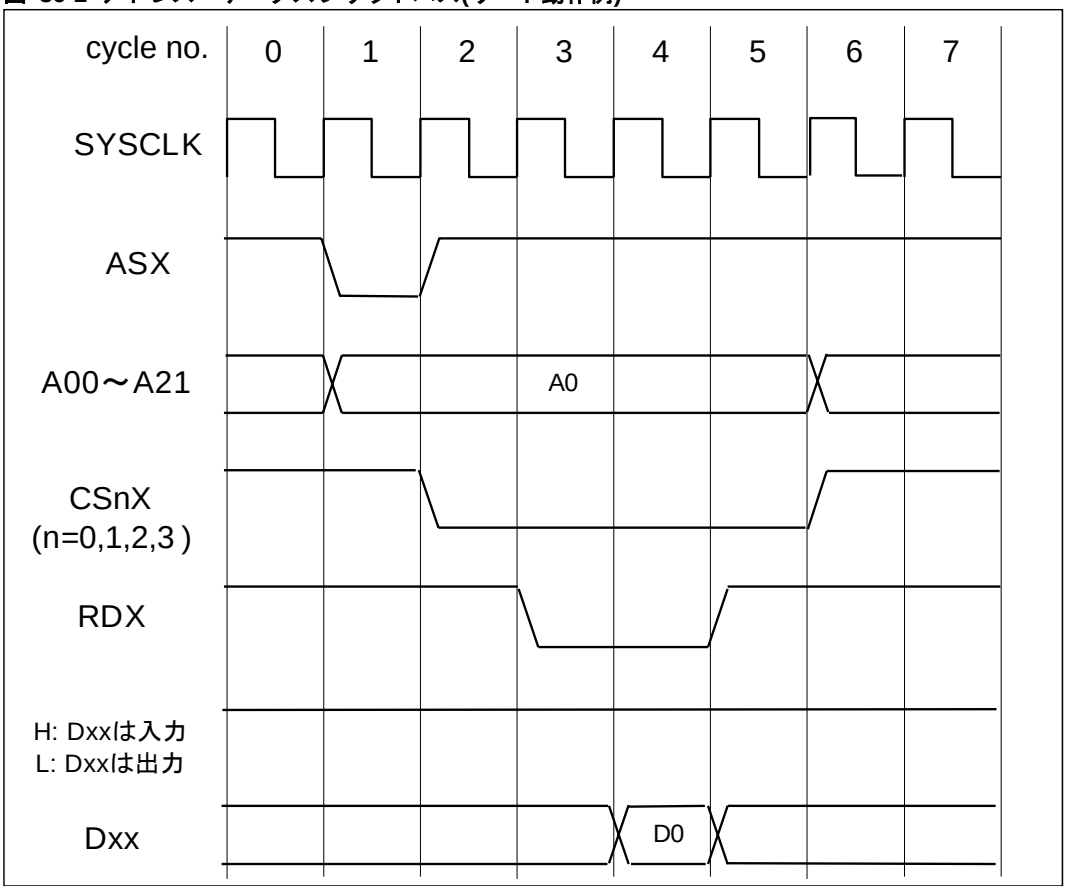
36.5.2　外バス信号プロトコル

外バス信号プロトコルについて示します。

● アドレス・データスプリットバス　リードプロトコル

アドレス・データスプリットバスにおけるリードアクセスのプロトコルを示します。

図 36-2 アドレス・データスプリットバス(リード動作例)



● 動作例説明

cycle1 : ASX に"L"が 1 サイクル出力され、このサイクルからアクセスが開始されたことを示します。A00～A21 はこのサイクルよりアクセス先のアドレス情報を示します。

cycle2 : アクセス開始時より設定カウント終了後、CSnX (n=0～3)に"L" の出力を開始し、アクセス完了まで継続します。外バス上のデバイスは CSnX="L"の期間内でのみ、アクセスに対して処理を行う必要があります。

cycle3 : CSnX="L"の出力開始から設定カウント終了後 RDX に"L"が出力されます。外バスデバイスは RDX="L" で示されるストローブ期間内に D16～D31 にリードデータを返す必要があります。

cycle4 : RDX="L"の出力開始から設定カウント終了後 RDX の出力を"H"に戻します。本モジュールは RDX= L 期間内の、最後の SYSCLK の立上りエッジで D16～D31 のデータを内部のバッファに取り込みます。

cycle5 : RDX="H"に戻してから設定カウント終了後 CSnX の出力を"H"に戻し、リードアクセス完了となります。この例ではこのサイクルの終わりで CSnX が"H"に戻りリードアクセス完了となります。

● 信号説明

外バスの出力信号は SYSCLK の立上りエッジに同期します。

ASX
アクセスの開始を示します。また，アドレスストローブとしても機能します。

アクセス開始時から 1 または 2 サイクル期間の"L"パルスを出力します。

A00～A21
アクセス先のアドレス情報を出力します。

アクセス開始時より出力し、アクセス完了まで継続します。

CSnX (n=0～3)
アクセス先のアドレスが対応した CS 領域内である事を示します。外バスデバイスはこの信号が"L"の期間でのみバスからの要求を処理する必要があります。アクセス開始時から設定カウント終了後、"L"の出力を開始し，アクセス完了まで継続します。

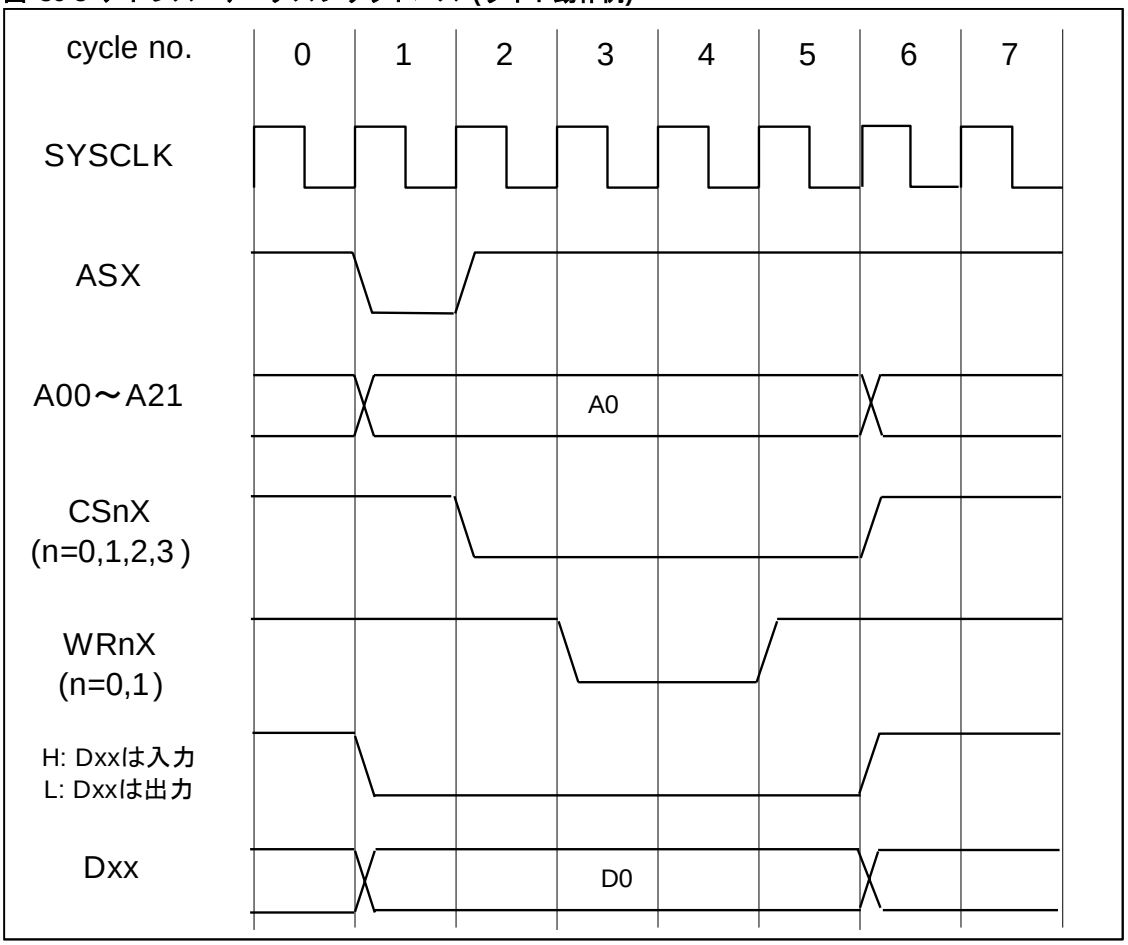
RDX
リードストローブ期間を示します。CSnX (n=0～3)駆動時より設定カウント終了後，リードアクセス時 "L" を出力します。リードオートウェイトカウント終了後"H"出力に戻します。外バスデバイスは RDX="L"の期間内に D16～D31 に有効なデータを返す必要があります。本モジュールは RDX="L"期間内の最後の SYSCLK の立上りエッジで, D16～D31 データを内部のバッファに取り込みます。

D16～D31
外バスデバイスは RDX="L"の期間内に D16～D31 に有効なデータを返す必要があります。本モジュールは RDX="L"期間内の最後の SYSCLK の立上りエッジで, D16～D31 データを内部のバッファに取り込みます。

● アドレス・データスプリットバス ライトプロトコル

アドレス・データスプリットバスにおけるライトアクセスのプロトコルを示します。

図 36-3 アドレス・データスプリットバス (ライト動作例)



● 動作例説明

cycle1 : ASX に"L"が 1 サイクル出力され、このサイクルからアクセスが開始されたことを示します。A00～A21 はこのサイクルより、アクセス先のアドレス情報を示します。

cycle2 : アクセス開始から設定カウント終了後, CSnX (n=0～3)に"L" が出力されます。CSnX はアクセス完了まで"L"の出力を継続します。外バス上のデバイスは CSnX="L"の期間内でのみ, アクセスに対する処理を実行する必要があります。

cycle3 : CSnX に"L"の出力を開始してから設定カウント終了後、WRnX (n=0, 1)に"L"が出力されます。外バスデバイスは WRnX に"L"が出力されているライトストローブ期間内に D16～D31 の値を取り込む必要があります。

cycle4 : WRnX="L"出力開始から設定カウント終了後、WRnX を"H"出力に戻し、ライトストローブ期間を終了します。この例ではライトストローブ期間が 1 サイクル延長されています。このサイクルの終わりで WRnX を"H"出力に戻し、ストローブ期間を終了しています。

cycle5 : WRnX="H"に戻してから設定カウント終了後 CSnX の出力を"H"に戻し、ライトアクセス完了となります。この例ではこのサイクルの終わりで CSnX が"H"に戻りライトアクセス完了となります。

● 信号説明

外バスの出力信号は SYSCLK の立上りエッジに同期します。

ASX
アクセスの開始を示します。また、アドレスストローブとしても機能します。

アクセス開始時から 1 または 2 サイクル期間の"L"パルスを出力します。

A00～A21
アクセス先のアドレス情報を出力します。

アクセス開始時より出力し、アクセス完了まで継続します。

CSnX (n=0～3)
アクセス先のアドレスが対応した CS 領域内である事を示します。外バスデバイスはこの信号が"L"の期間でのみバスからの要求を処理する必要があります。アクセス開始時から設定カウント終了後、"L"の出力を開始し、アクセス完了まで継続します。

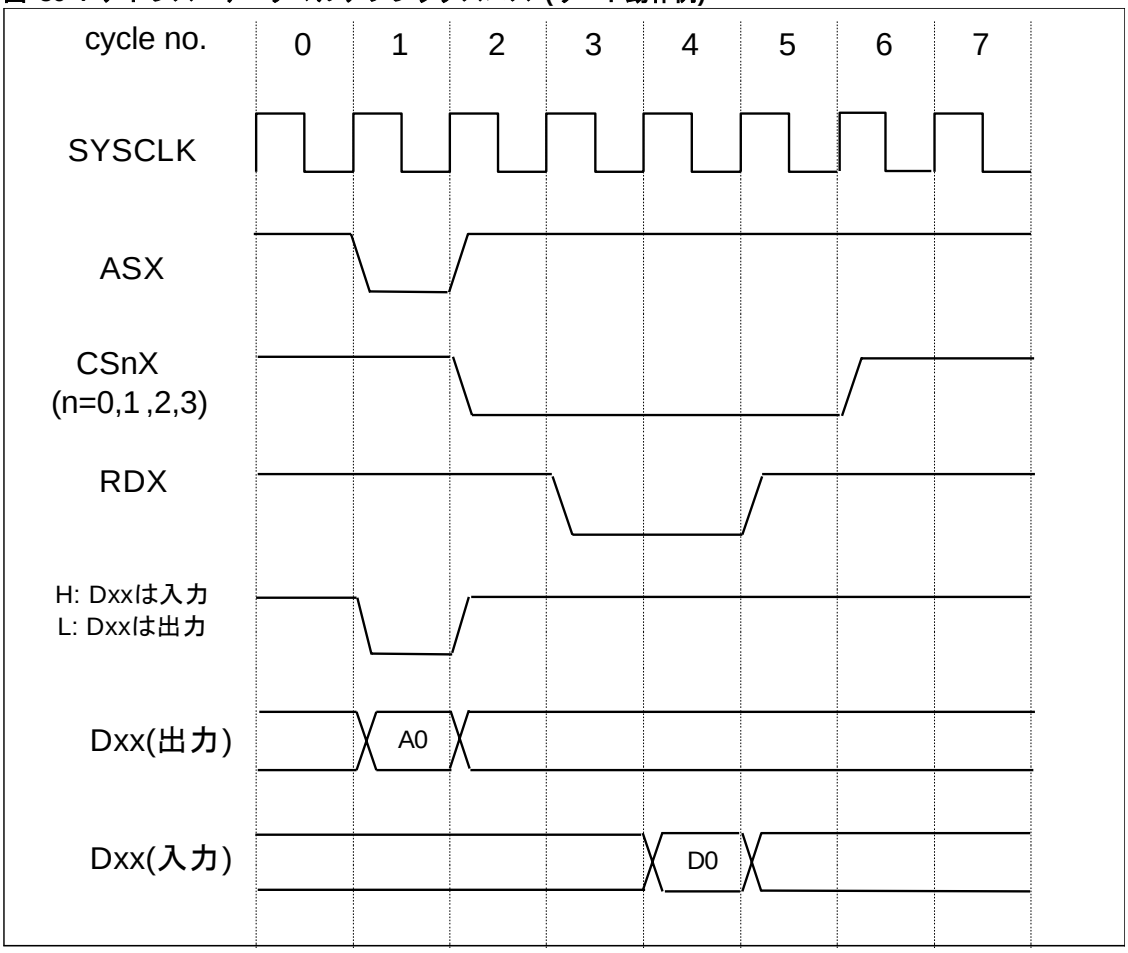
WRnX (n=0, 1)
ライトサイクルストローブ期間を示します。CSnX (n=0～3)駆動時よりカウント終了後、ライトアクセス時には"L "を出力します。ライトオートウェイトカウント終了後"H"出力に戻します。外バスデバイスはWRnX (n=0, 1)="L"の期間に、D16～D31 のデータを取り込む必要があります。

D16～D31
アクセス開始時からライトデータを出力します。アクセス完了まで、ライトデータの出力が継続されます。外バスデバイスは WRnX="L"の期間内に D16～D31 のデータを取り込む必要があります。

● アドレス・データマルチプレックスバス リードプロトコル

アドレス・データマルチプレックスバスにおけるリードアクセスのプロトコルを示します。

図 36-4 アドレス・データマルチプレックスバス (リード動作例)



● 動作例説明

cycle1 : ASX に"L"が出力され、このサイクルからアクセスが開始されたことを示しています。データバス D16～D31 にはアドレス情報 A0 が出力されます。ASX はこのアドレス情報のストローブ信号として機能します。このアドレス情報は設定カウントサイクル出力します。設定カウント終了後、D16～D31 は入力状態とします。

cycle2 : アクセス開始時より設定カウント終了後、CSnX (n=0～3)に"L" の出力を開始し、アクセス完了まで継続します。外バス上のデバイスは CSnX="L"の期間内でのみ、アクセスに対して処理を行う必要があります。

cycle3 : CSnX="L"の出力開始から設定カウント終了後 RDX に"L"が出力されます。外バスデバイスは RDX="L"で示されるストローブ期間内に D16～D31 にリードデータを返す必要があります。

cycle4 : RDX="L"の出力開始から設定カウント終了後 RDX の出力を"H"に戻します。RDX="L"期間内の、最後の SYSCLK の立上りエッジで D16～D31 のデータを内部のバッファに取り込みます。

cycle5 : RDX="H"に戻してから設定カウント終了後 CSnX (n=0～3)の出力を"H"に戻し、リードアクセス完了となります。この例ではこのサイクルの終わりで CSnX が"H"に戻りリードアクセス完了となります。

● 信号説明

外バスの出力信号は SYSCLK の立上りエッジに同期します。

ASX

アクセスの開始を示します。また, アドレスストローブとしても機能します。

アクセス開始時から 1 または 2 サイクル期間の"L"パルスを出力します。

CSnX (n=0～3)

アクセス先のアドレスが対応した CS 領域内である事を示します。外バスデバイスはこの信号が"L"の期間でのみバスからの要求を処理する必要があります。アクセス開始時から設定カウント終了後、"L"の出力を開始し, アクセス完了まで継続します。

RDX

リードストローブ期間を示します。CSnX (n=0～3)駆動時より設定カウント終了後、リードアクセス時 "L" を出力します。リードオートウェイトカウント終了後"H"出力に戻します。外バスデバイスは RDX="L"の期間内に D16～D31 に有効なデータを返す必要があります。本モジュールは RDX="L"期間内の最後の SYSCLK の立上りエッジで, D16～D31 のデータを内部のバッファに取り込みます。

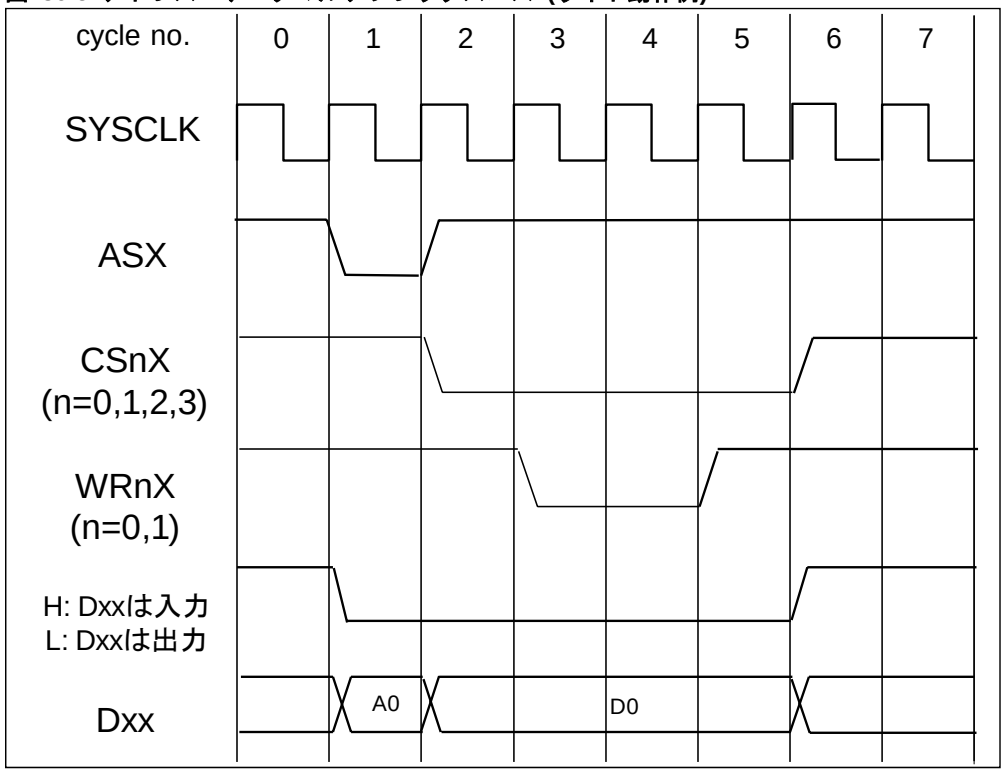
D16～D31

アクセス開始時からアドレス情報を出力します。設定カウント終了後入力状態となり、外バスデバイスからのリードデータを受け付けます。本モジュールは RDX="L"期間内の最後の SYSCLK の立上りエッジで、D16～D31 のデータを内部のバッファに取り込みます。

● アドレス・データマルチプレックスバス ライトプロトコル

アドレス・データマルチプレックスバスにおけるライトアクセスのプロトコルを示します。

図 36-5 アドレス・データマルチプレックスバス (ライト動作例)



● 動作例説明

cycle1 : アクセス開始サイクルです。アクセス開始を示す ASX に"L"が出力されます。D16～D31 にはアドレス情報が出力されます。ASX はこのアドレス情報のストローブ信号として機能します。このアドレス情報は設定カウントサイクル出力します。

cycle2 : アクセス開始時より設定カウント終了後、CSnX (n=0～3)に"L" の出力を開始し、アクセス完了まで継続します。外バス上のデバイスは CSnX="L"の期間内でのみ、アクセスに対して処理を行う必要があります。

cycle3 : CSnX に"L"の出力を開始してから設定カウント終了後、WRnX (n=0, 1)に"L"が出力されます。外バスデバイスは WRnX="L"で示されるライトストローブ期間内に D16～D31 の値を取り込む必要があります。

cycle4 : WRnX="L"出力開始から設定カウント終了後, WRnX を"H"出力に戻し、ライトストローブ期間を終了します。この例ではライトストローブ期間が 1 サイクル延長されています。このサイクルの終わりで WRnX を"H"出力に戻し、ライトストローブ期間を終了しています。

cycle5 : WRnX="H"に戻してから設定カウント終了後 CSnX の出力を"H"に戻し、ライトアクセス完了となります。この例ではこのサイクルの終わりで CSnX が"H"に戻りライトアクセス完了となります。

● 信号説明

外バスの出力信号は SYSCLK の立上りエッジに同期します。

ASX

アクセスの開始を示します。また, アドレスストローブとしても機能します。

アクセス開始時から 1 または 2 サイクル期間の"L"パルスを出力します。

CSnX (n=0～3)

アクセス先のアドレスが対応した CS 領域内である事を示します。外バスデバイスはこの信号が"L"の期間でのみバスからの要求を処理する必要があります。アクセス開始時から設定カウント終了後、"L"の出力を開始し、アクセス完了まで継続します。

WRnX (n=0, 1)

ライトストローブ期間を示します。CSnX 駆動時よりカウント終了後、ライトアクセス時には"L "を出力します。ライトオートウェイトカウント終了後"H"出力に戻します。外バスデバイスは WRnX="L"の期間に、D16～D31 のデータを取り込む必要があります。

D16～D31

アクセス開始時から、アクセス先のアドレス情報を出力します。設定カウント終了後ライトデータの出力を開始し、アクセス完了まで継続します。外バスデバイスはライトストローブ期間内に D16～D31 の値を取り込む必要があります。

36.5.3　アドレスアライメント

アドレスアライメントについて示します。

外部バスインタフェースは、アクセス先アドレスの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・ハーフワードアクセスの場合には、以下のようになります。

● ワードアクセス

プログラムで指定したアドレスの最下位 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"00"となります。

● ハーフワードアクセス

プログラムで指定したアドレスの最下位 2 ビットが"00""01"の場合、出力するアドレスの下位 2 ビットは"00"となり、"10", "11"の場合、出力するアドレスの最下位 2 ビットは"10"となります。

36.5.4　分割アクセス

分割アクセスについて示します。

アクセスサイズがバス幅よりも大きい場合には、1 つのアクセスが分割されて実行されます。

● 分割アクセス回数

バス幅	アクセスサイズ		
	バイト	ハーフワード	ワード
8 ビット	1 回	2 回	4 回
16 ビット	1 回	1 回	2 回

36.5.5 データアライメント

データアライメントについて示します。

各 CS 領域はビッグエンディアンとリトルエンディアンの両方エンディアンをサポートしています。ただし、CS0 はビッグエンディアンのみのサポートです。データバス幅は、CS 領域ごとに 8 ビット/16 ビットを選択可能です。

以下に、各エンディアン設定・データバス幅設定時の、外部アクセスサイズに対するデータアライメントと対応する制御信号を示します。

● ビッグエンディアン・16 ビット

アクセス		分割アクセス	出力端子				
サイズ	アドレス下位 2 ビット		A01, A00	D31～D24	D23～D16	WR0X	WR1X
バイト	00	―	00	bit7～bit0		○	
	01	―	01		bit7～bit0		○
	10	―	10	bit7～bit0		○	
	11	―	11		bit7～bit0		○
ハーフワード	0n	―	00	bit15～bit8	bit7～bit0	○	○
	1n	―	10	bit15～bit8	bit7～bit0	○	○
ワード	nn	分割アクセス 1 回目	00	bit31～bit24	bit23～bit16	○	○
		分割アクセス 2 回目	10	bit15～bit8	bit7～bit0	○	○

● ビッグエンディアン・8 ビット

アクセス		分割アクセス	出力端子				
サイズ	アドレス下位 2 ビット		A01, A00	D31～D24	D23～D16	WR0X	WR1X
バイト	00	―	00	bit7～bit0	―	○	―
	01	―	01	bit7～bit0	―	○	―
	10	―	10	bit7～bit0	―	○	―
	11	―	11	bit7～bit0	―	○	―
ハーフワード	0n	分割アクセス 1 回目	00	bit15～bit8	―	○	―
		分割アクセス 2 回目	01	bit7～bit0	―	○	―
	1n	分割アクセス 1 回目	10	bit15～bit8	―	○	―
		分割アクセス 2 回目	11	bit7～bit0	―	○	―
ワード	nn	分割アクセス 1 回目	00	bit31～bit24	―	○	―
		分割アクセス 2 回目	01	bit23～bit15	―	○	―
		分割アクセス 3 回目	10	bit15～bit8	―	○	―
		分割アクセス 4 回目	11	bit7～bit0	―	○	―

● リトルエンディアン・16 ビット

アクセス		分割アクセス	出力端子				
サイズ	アドレス下位 2 ビット		A01, A00	D31～D24	D23～D16	WR0X	WR1X
バイト	00	―	00	bit7～bit0	―	○	―
	01	―	01	―	bit7～bit0	―	○
	10	―	10	bit7～bit0	―	○	―
	11	―	11	―	bit7～bit0	―	○
ハーフワード	0n	―	00	bit7～bit0	bit15～bit8	○	○
	1n	―	10	bit7～bit0	bit15～bit8	○	○
ワード	nn	分割アクセス 1 回目	00	bit7～bit0	bit15～bit8	○	○
		分割アクセス 2 回目	10	bit23～bit16	bit31～bit24	○	○

● リトルエンディアン・8 ビット

アクセス		分割アクセス	出力端子				
サイズ	アドレス下位 2 ビット		A01, A00	D31～D24	D23～D16	WR0X	WR1X
バイト	00	―	00	bit7～bit0	―	○	―
	01	―	01	bit7～bit0	―	○	―
	10	―	10	bit7～bit0	―	○	―
	11	―	11	bit7～bit0	―	○	―
ハーフワード	0n	分割アクセス 1 回目	00	bit7～bit0	―	○	―
		分割アクセス 2 回目	01	bit15～bit8	―	○	―
	1n	分割アクセス 1 回目	10	bit7～bit0	―	○	―
		分割アクセス 2 回目	11	bit15～bit8	―	○	―
ワード	nn	分割アクセス 1 回目	00	bit7～bit0	―	○	―
		分割アクセス 2 回目	01	bit15～bit8	―	○	―
		分割アクセス 3 回目	10	bit23～bit16	―	○	―
		分割アクセス 4 回目	11	bit31～bit24	―	○	―

36.5.6 アドレス情報

アドレス情報について示します。

● アドレス情報と出力端子

■ アドレス・データスプリットバス

22 ビットのアドレス情報を A00～A21 に出力します。

■ アドレス・データマルチプレックスバス

アドレス・データマルチプレックスバスでは，アドレス出力サイクル中データバス端子 D16～D31 にアドレス情報を出力します。データバスの幅の設定によって出力可能なアドレスのビット幅が決まります。アドレス・データマルチプレックスバスを選択時においても、アドレス端子 A00～A21 はアドレスを出力します。D16～D31 端子に出力されるアドレス情報の欠けている部分は、アドレス端子 A00～A21 を使用して補完することができます。

● アドレスタイプ

アドレス情報の出力を通常どおり出力する通常タイプとビットシフトして出力するシフトタイプから選択できます。ACR:ADTY で設定します。

- ADTY=0
  通常出力モードです。アドレス情報をビットシフトせずにそのまま端子に出力します。
- ADTY=1
  アドレスシフト出力モードです。アドレスバス情報をビットシフトして端子に出力します。

アドレスタイプ(ACR:ADTY)，バスタイプ(ACR:BSTY)，バス幅と出力アドレス情報，アドレス出力端子の関係は下表のとおりです。

● 出力アドレスと出力端子

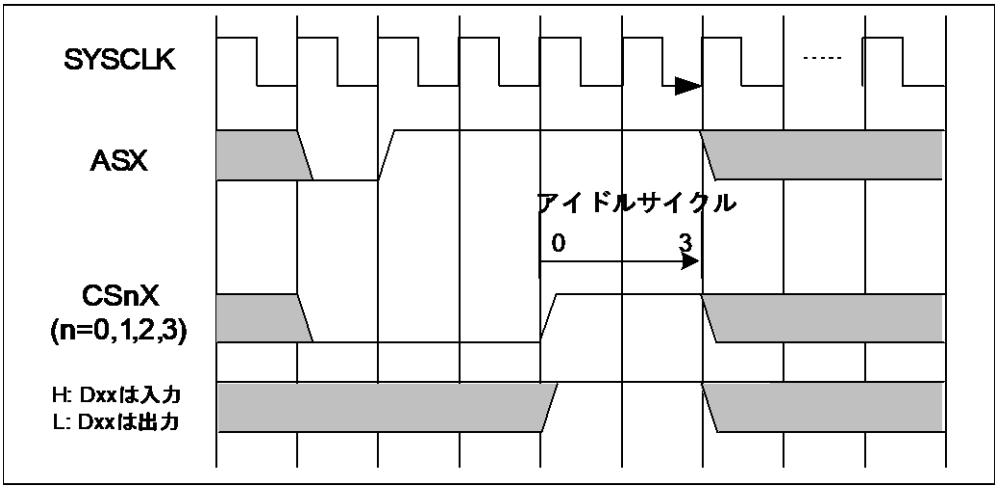
ACR レジスタ		バス幅 [bit]	A21～A00	アドレス出力サイクル時の出力端子 D31～D16	
ADTY	BSTY			D31～D24	D23～D16
0	0	8	アドレス[21:0]	－	－
		16			
0	1	8	アドレス[21:0]	アドレス[7:0]	－
		16	アドレス[21:0]	アドレス[15:8]	アドレス[7:0]
1	0	8	アドレス[21:0]	－	－
		16	アドレス[22:1]		
1	1	8	アドレス[21:0]	アドレス[7:0]	－
		16	アドレス[22:1]	アドレス[16:9]	アドレス[8:1]

36.5.7　アイドルサイクル挿入機能

アイドルサイクル挿入機能について示します。

アクセスとアクセスの間にアイドルサイクルを挿入することができます。アイドルサイクル中は要求があっても次のアクセスを開始せず, アイドルサイクルカウント終了後開始します。

図 36-6 アイドルサイクル挿入



● リードアクセスアイドルサイクル

リードアクセスに連続して下記の何れかに当てはまるアクセスが行なわれる場合、リードアクセス終了後アイドルサイクルが挿入されます。AWR:RIDL[1:0]で設定します。

- ライトアクセス
- 別の CS 領域へのアクセス
- アドレス・データマルチプレックスバスタイプに設定された CS 領域に対するアクセス

＜注意事項＞

RIDL によるアイドルサイクル挿入が行なわれないのは、スプリットバスタイプに設定された同一 CS 領域に対するリードアクセスが連続する場合のみです。

● ライトリカバリサイクル

ライトアクセス終了後アイドルサイクルを挿入します。AWR:WRCV[1:0]で設定します。

36.5.8　外バス出力信号タイミング設定

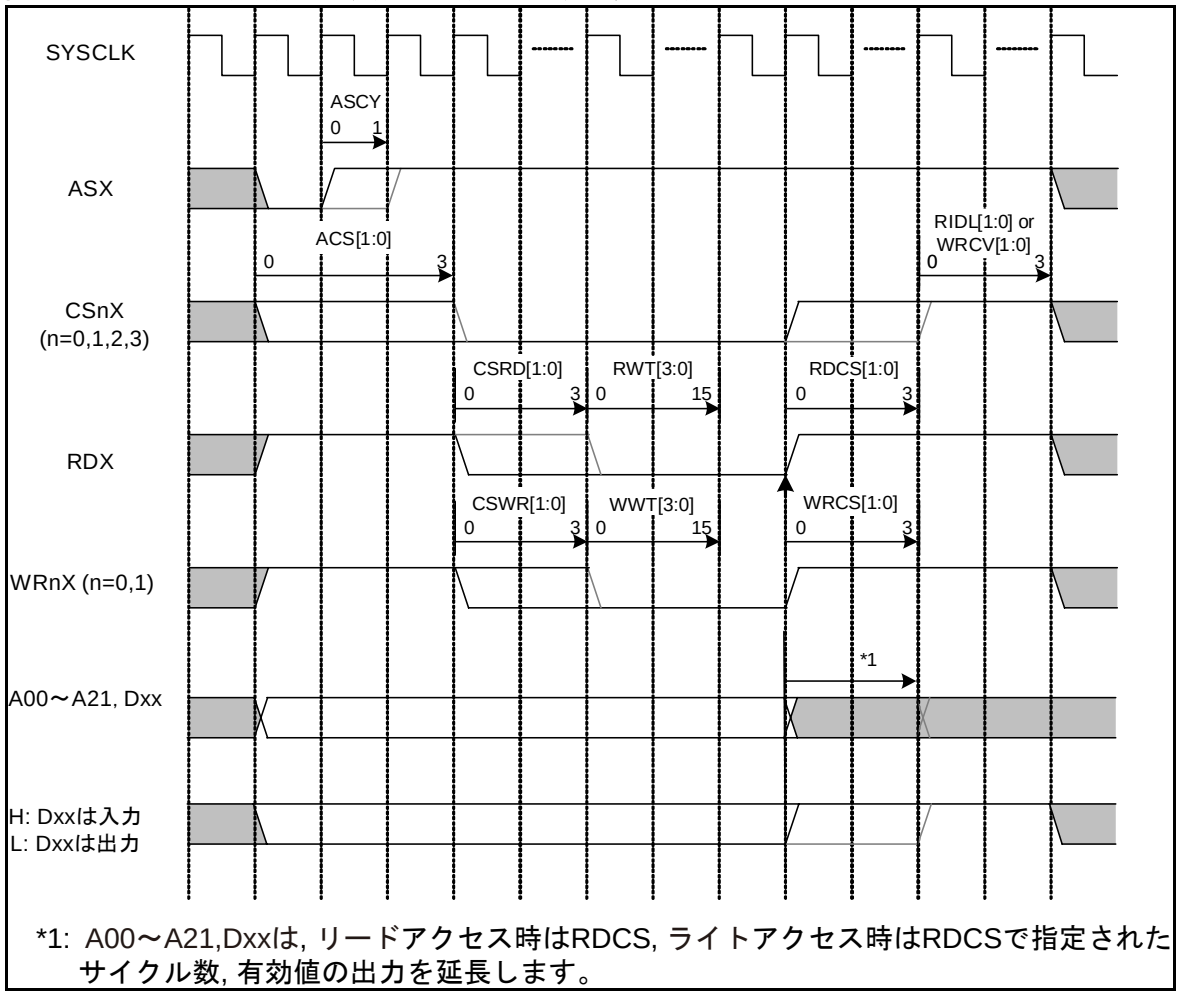
外バス出力信号タイミング設定について示します。

外バス信号の出力タイミングは以下のパラメータによって決定されます。タイミングパラメータはレジスタへの設定値で決まります。

● アドレス・データスプリットバスタイミングパラメータ

アドレス・データスプリットバスで設定可能なタイミングパラメータを示します。

図 36-7 アドレス・データスプリットバスタイミングパラメータ



*1: A00～A21,Dxxは, リードアクセス時はRDCS, ライトアクセス時はRDCSで指定されたサイクル数, 有効値の出力を延長します。

● アドレス・データスプリットバスタイミングパラメータ

パラメータ名	機能名	説明
ASCY(ASX CYcle)	ASX 出力延長サイクル数	アクセス開始時から、ASX に(ASCY+1)サイクル"L"を出力します。
ACS[1:0] (A00～A21 to CSnX delay cycle)	A00～A21 → CSnX ディレイサイクル数	ASX 出力から ACS カウント終了後, CSnX (n=0～3)に"L"の出力を開始します。
CSRD[1:0] (CSnX to RDX setup cycle)	CSnX → RDX セットアップサイクル	リードアクセス時、CSnX の"L"出力開始から CSRD カウント終了後 RDX に"L"の出力を開始します。
RWT[3:0] (Read access auto WaiT)	リードアクセスオートウェイト	リードアクセス時、RDX の"L"出力開始から (RWT+1)カウント終了後, RDX の出力を"H"に戻します。
RDCS[1:0] (RDX to CSnX hold cycle)	RDX → CSnX ホールドサイクル	リードアクセス時、RDX の出力を"H"出力に戻したサイクルから RDCS カウント後、CSnX の出力を"H"に戻します。
CSWR[1:0] (CSnX to WRnX setup cycle)	CSnX → WRnX セットアップサイクル	ライトアクセス時、CSnX の"L"出力開始から CSWR カウント終了後 WRnX に"L"の出力を開始します。
WWT[3:0] (Write access auto WaiT)	ライトアクセスオートウェイト	ライトアクセス時、(WWT+1)カウント終了後, WRnX(n=0, 1)の出力を"H"に戻します。
WRCS[1:0] (WRnX to CSnX hold cycle)	WRnX → CSnX ホールドサイクル	ライトアクセス時、 WRnX の出力を"H"出力に戻したサイクルから WRCS カウント後 CSnX の出力を"H"に戻します。
RIDL[1:0] (Read access IDLe cycle)	リードアクセスアイドルサイクル	リードアクセス終了後は、RIDL カウント終了後、次のアクセスが開始可能となります。
WRCV[1:0] (Write ReCoVery cycle)	ライトリカバリサイクル	ライトアクセス終了後は、WRCV カウント終了後、次のアクセスが開始可能となります。

アクセスサイクル数は次の式で決まります。

リードアクセスサイクル数 = アドレス&データ出力(1) + ACS(0～3) + CSRD(0～3) + RWT(0～15) +RDCS(0～3)

最小:1 サイクル, 最大:25 サイクル

ライトアクセスサイクル数 = アドレス&データ出力(1) + ACS(0～3) + CSWR(0～3) + WWT(0～15) + WRCS(0～3)

最小:1 サイクル, 最大 25 サイクル

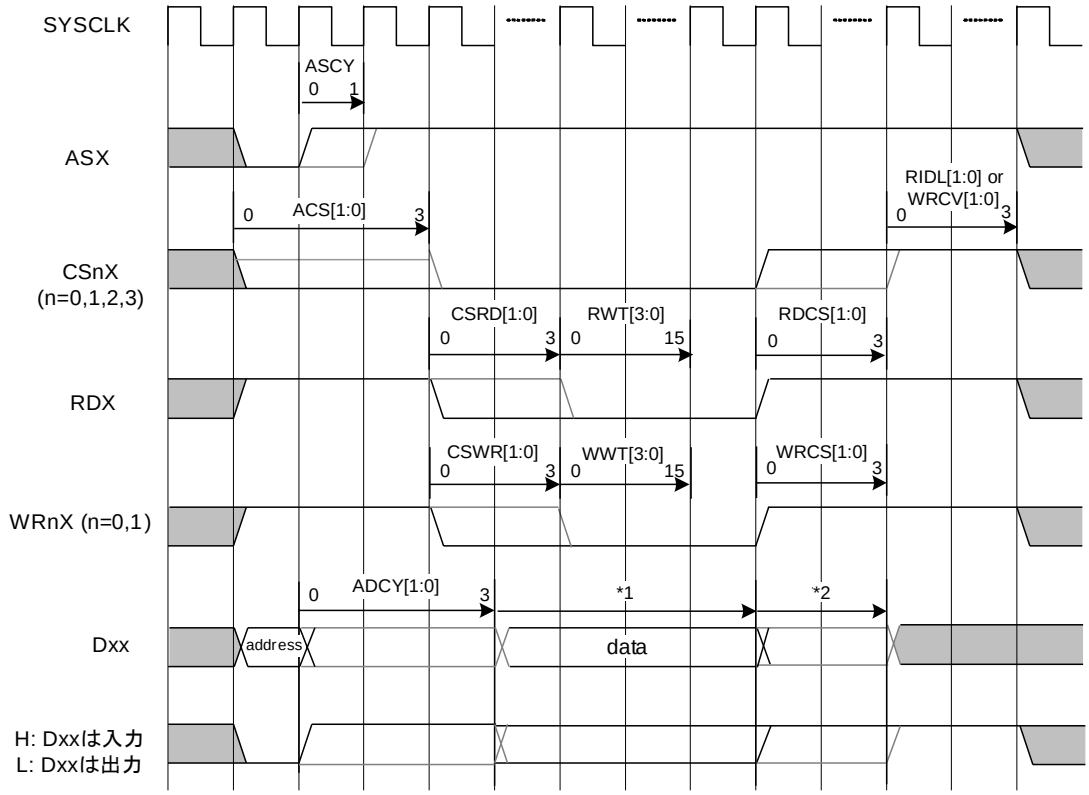
プロトコルが正しく成立するように, 以下の条件が成立している必要があります。

ASCY ≦ ACS + CSRD + RWT + RDCS かつ ASCY ≦ ACS + CSWR + WWT + WRCS

● アドレス・データマルチプレックスバスタイミングパラメータ

アドレス・データマルチプレックスバスで設定可能なタイミングパラメータを示します。

図 36-8 アドレス・データマルチプレックスバスタイミングパラメータ



*1: Dxxはライトアクセスではアドレス出力サイクルが終了次第ライトデータを出力します。
*2: Dxxはライトアクセス時WRCSで設定されたサイクル数, 有効値の出力を延長します。

● アドレス・データマルチプレックスバスタイミングパラメータ

パラメータ名	機能名	説明
ASCY(ASX CYcle)	ASX 出力延長 サイクル数	アクセス開始時から、ASX に(ASCY+1)サイクル"L"を出力します。
ACS[1:0] (A00～A21 to CSnX delay cycle)	A00～A21 → CSnX ディレイサイクル数	ASX 出力から ACS カウント終了後、CSnX(n=0～3)に"L"の出力を開始します。
ADCY[1:0](ADdress CYcle)	アドレス出力 延長サイクル数	ADCY≧ASCY のとき D16～D31 はアクセス開始時から、(ADCY+1)サイクルアドレス情報を出力します。ライト時はカウント終了後ライトデータをアクセス終了まで出力します。 ADCY＜ASCY のとき カウント値が(ADCY+1)から(ASCY+1)に変更になります。 それ以外の点に違いはありません。 ADCY のカウントは他のカウンタとは独立に動作します。また他のカウンタの開始条件にも影響をおよぼしません。したがいまして、全体のプロトコルとして正しく動作するためにはカウンタ値の設定方法に制限があります。表外の設定禁止条件を参照してください。
CSRD[1:0] (CSnX to RDX setup cycle)	CSnX → RDX セットアップサイクル	リードアクセス時、CSnX の"L"出力開始から CSRD カウント終了後 RDX に"L"の出力を開始します。
RWT[3:0] (Read access auto WaiT)	リードアクセス オートウェイト	リードアクセス時、RDX の"L"出力開始から(RWT+1)カウント終了後、RDX の出力を"H"に戻します。
RDCS[1:0] (RDX to CSnX hold cycle)	RDX → CSnX ホールドサイクル	リードアクセス時、RDX の出力を"H"出力に戻したサイクルから RDCS カウント後、CSnX の出力を"H"に戻します。
CSWR[1:0] (CSnX to WRnX setup cycle)	CSnX → WRnX セットアップサイクル	ライトアクセス時、CSnX の"L"出力開始から CSWR カウント終了後 WRnX(n=0, 1)に"L"の出力を開始します。
WWT[3:0] (Write access auto WaiT)	ライトアクセス オートウェイト	ライトアクセス時、(WWT+1)カウント終了後、WRnX の出力を"H"に戻します。
WRCS[1:0] (WRnX to CSnX hold cycle)	WRnX → CSnX ホールドサイクル	ライトアクセス時、WRnX の出力を"H"出力に戻したサイクルから WRCS カウント後 CSnX の出力を"H"に戻します。
RIDL[1:0] (Read access IDLe cycle)	リードアクセス アイドルサイクル	リードアクセス終了後は、RIDL カウント終了後、次のアクセスが開始可能となります。
WRCV[1:0] (Write ReCoVery cycle)	ライトリカバリ サイクル	ライトアクセス終了後は、WRCV カウント終了後，次のアクセスが開始可能となります。

アクセスサイクル数は次の式で決まります。

リードアクセスサイクル数= アドレス出力(1) + ACS(0～3) + CSRD(0～3) + データ出力(1) + RWT(0～15) + RDCS(0～3)

最小:2 サイクル，最大:26 サイクル

ライトアクセスサイクル数= アドレス出力(1) + ACS(0～3) + CSWR(0～3) + データ出力(1) + WWT(0～15) + WRCS(0～3)

最小:2 サイクル, 最大 26 サイクル

プロトコルが正しく成立するように, 以下の 4 条件が成立している必要があります。

ADCY + 1 ≦ ACS + CSRD
ADCY + 1 ≦ ACS + CSWR
ASCY + 1 ≦ ACS + CSRD
ASCY + 1 ≦ ACS + CSWR

36.5.9 RDY 端子アクセスサイクル延長機能

RDY 端子アクセスサイクル延長機能について示します。

RDY 端子に"0"を入力することにより、オートウェイトサイクル終了後もリード, ライトストローブサイクルを延長することができます。

AWR:RDYE に"1"を設定することにより、対応する領域へのアクセスに対して RDY 端子によるこの機能が有効になります。

この機能は対応する領域のオートウェイトサイクルを 2 以上に設定して使用してください。

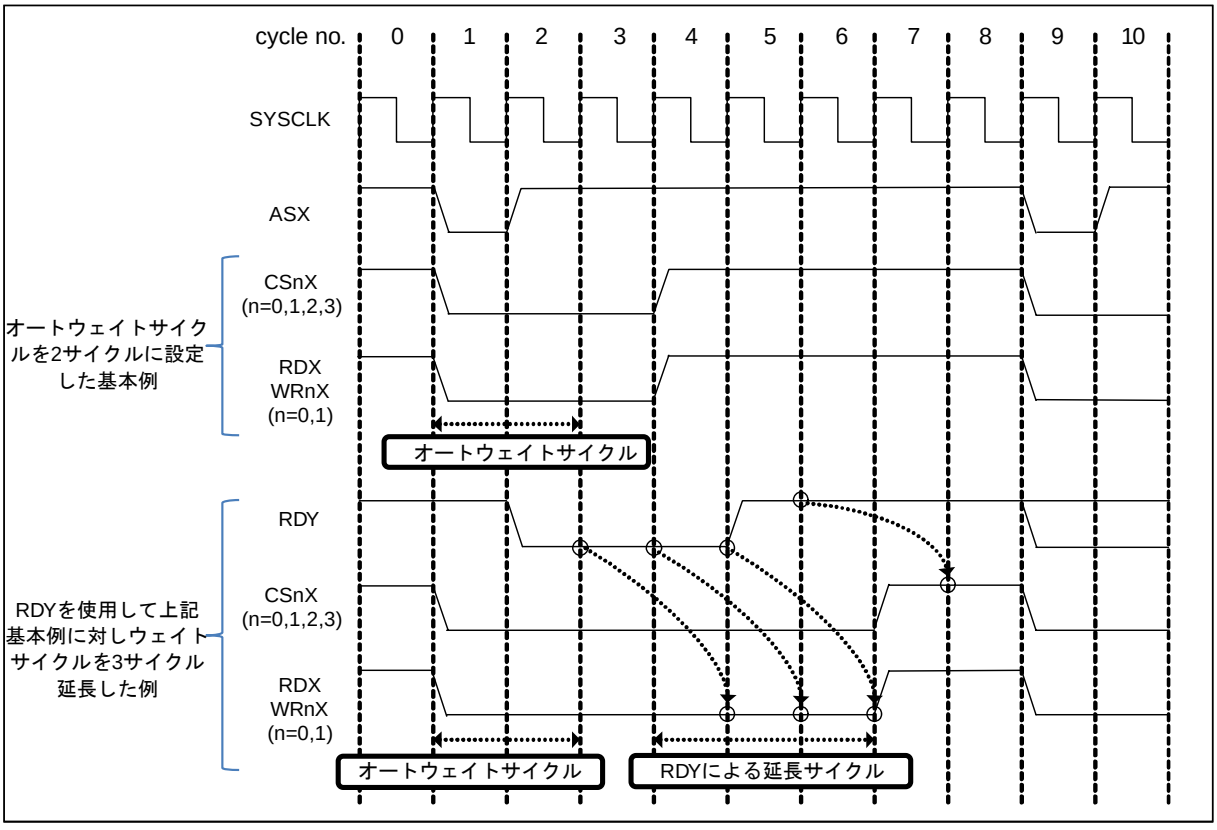
オートウェイトサイクル終了後、RDY に"0"が入力されている期間はリード, ライトストローブサイクルを延長します。その後 RDY に"1"が入力されると、次のサイクルでリード, ライトストローブサイクルを終了します。

RDY 信号の入力仕様

入力する RDY 信号は以下の仕様を守ってください。

- オートウェイトサイクルを延長する時以外は RDY=1 を入力してください。
- ASX="L"と CSnX="L"でオートウェイトサイクル延長対象領域へのアクセスが開始されたことを確認してから RDY=0 の入力を開始してください。
- RDY=0 の入力はオートウェイトサイクルの終了迄に開始してください。オートウェイトサイクル終了後の RDY=0 の入力は禁止です。
- 必要な延長サイクルが終了したら RDY=1 を入力してください。

図 36-9 RDY タイミング例



36.5.10 CS 設定フロー

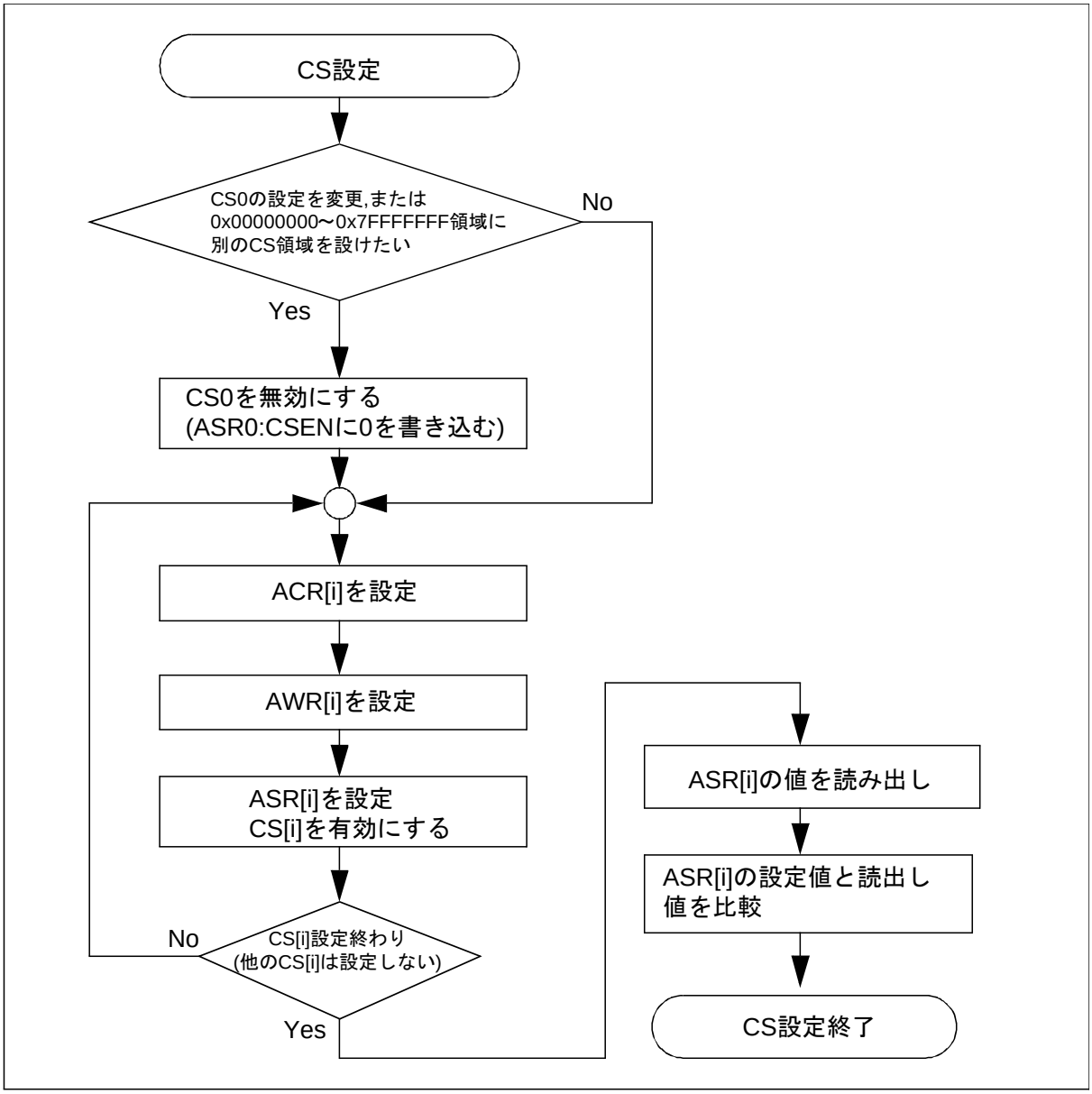
CS の設定方法について説明します。

＜注意事項＞

- CS の設定はリセット後の初期設定の中で行い、その後変更しないでください。
- ROM を搭載する品種では ROM 内に配置する初期設定プログラムで CS 領域の変更・設定を行ってください。
- ROM を搭載しない品種では、リセット後の命令フェッチは CS0 領域から行うため、CS0 領域の変更を伴う場合には、CS 設定プログラムを内蔵 RAM へいったん転送後、内蔵 RAM 内のプログラム領域に分岐を行い CS 領域の設定を行ってください。
- CS 領域へのアクセス中に、関係する CS 領域の設定を変更した場合の動作は保証できません。

以下に CS 設定のフローを示します。

図 36-10 CS 設定フロー



● CS0 無効化

CS0 の変更を行う場合にはまず CS0 を無効にする必要があります。ASR0 に 0x0 をワードで書き込んでください。

● ACR 設定

CS 領域のバス幅, バスタイプなどを設定します。

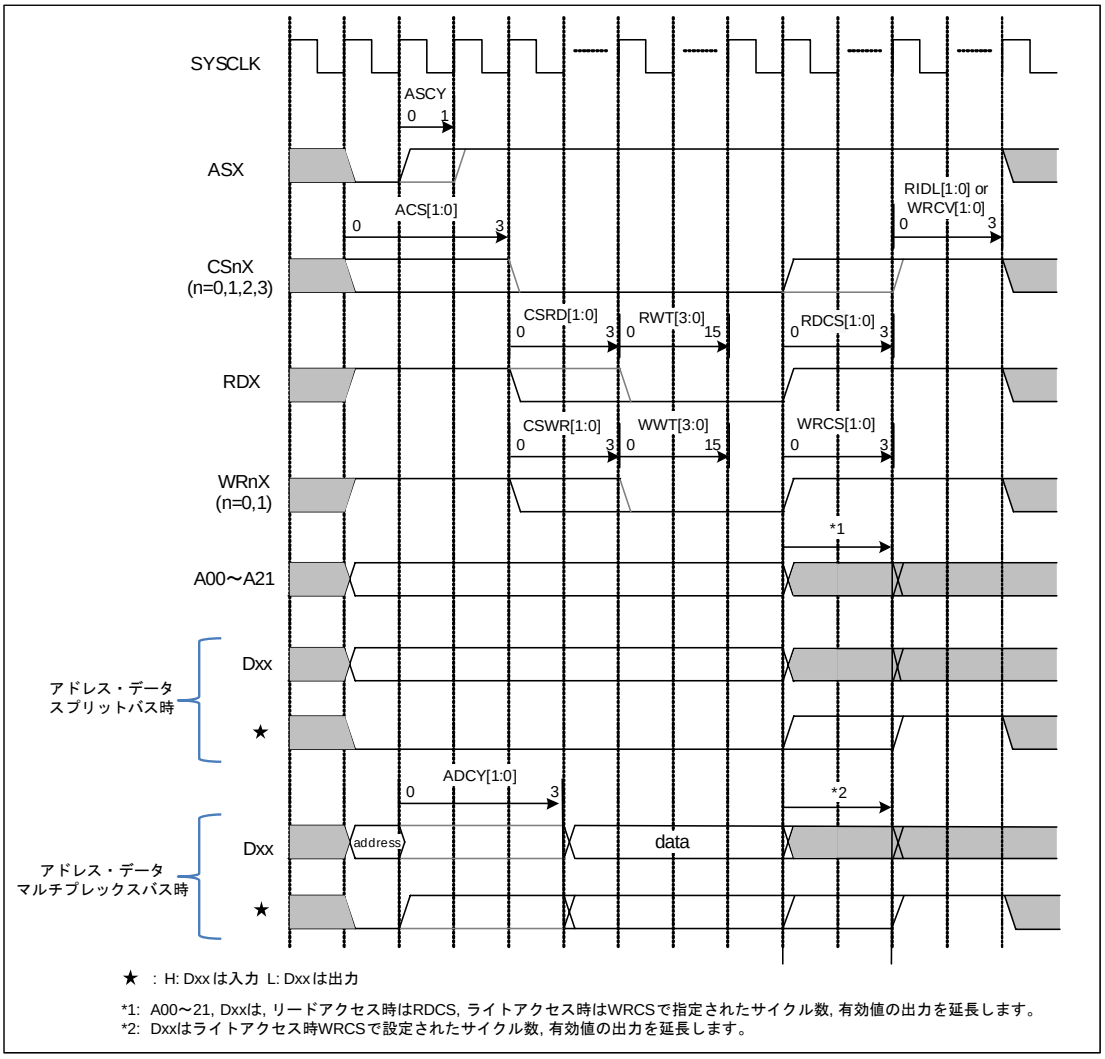
1. 設定する CS 領域のデータバス幅を 8 ビット, 16 ビットから選択します。
2. アドレス出力のタイプを通常出力とシフト出力から選択します。
3. バスタイプをアドレス・データスプリットバス, アドレス・データマルチプレックスバスより選択します。

以上の設定値をワード単位で ACR に書き込みます。

● AWR の設定

外バス信号の出力タイミングを決めるパラメータと RDY 端子機能の有効・無効を設定します。設定値をワードで AWR に書き込みます。

図 36-11 AWR 設定可能なパラメータ



● パラメーター一覧

パラメータ名	説明
RWT[3:0]	リードアクセスサイクルにおけるオートウェイトサイクル数を設定します。 リードアクセスサイクルを延長したい場合に設定します。
WWT[3:0]	ライトアクセスサイクルにおけるオートウェイトサイクル数を設定します。 ライトアクセスサイクルを延長したい場合に設定します。
RIDL[1:0]	リードアクセス後のアイドルサイクル数を設定します。 RIDL は出力オフタイムが長いデバイスからのリードデータと，それに続くアクセスに関係するデータのデータバス上での衝突を防ぐ目的で設定します。
WRCV[1:0]	ライトリカバリサイクル数を設定します。 ライトアクセス後に続けてアクセスする際にその間隔に制限があるデバイスへのアクセスを制御するために設定します。
CSRD[1:0]	CSnX (n=0～3)アサート後に RDX をアサートするまでのサイクル数を設定します。 リードアクセス時に RDX のアサートに対する CSnX のアサートのセットアップタイムが必要な場合に設定します。
RDCS[1:0]	RDX ネゲート後に CSnX (n=0～3)をネゲートするまでのサイクル数を設定します。 リードアクセス時に RDX のネゲートから CSnX のネゲートに関してホールドタイムを必要とする場合に設定します。
CSWR[1:0]	CSnX アサート後に WRnX(n=0, 1)をアサートするまでのサイクル数を設定します。 ライトアクセス時に WRnX のアサートに対する CSnX のアサートのセットアップタイムが必要な場合に設定します。
WRCS[1:0]	WRnX ネゲート後に CSnX をネゲートするまでのサイクル数を設定します。 ライトアクセス時に WRnX のネゲートから CSnX のネゲートに関してホールドタイムを必要とする場合に設定します。
ADCY[1:0]	アドレス・データマルチプレックスバス選択時にデータバスへのアドレス出力の延長サイクル数を設定します。ADCY=00 と設定した場合でも、ASCY=1 に設定した場合にはアドレス出力サイクルは 1 サイクル延長されます。アドレス・データスプリットバス選択時には 00 を設定してください。
ACS[1:0]	A00～A21 および ASX 出力から CSnX 出力の遅延サイクル数を設定します。CSnX のアサートに対してアドレスがセットアップタイムを必要とする場合や、連続して同一チップセレクト領域にアクセスした場合にも CSnX のエッジを必要とする場合に使用します。
ASCY	ASX のアサートの延長サイクル数を設定します。
RDYE	RDY 端子による外部からのウェイト挿入機能の有効・無効を設定します。

● ASR の設定

ASR では以下の設定を行います。

1. CSの領域設定します。
2. 書込み許可・禁止選択します。
3. バイトオーダの選択を行います。
4. CSを有効にします。

以上の設定値をワードで ASR に書き込みます。

CS の領域設定について次に説明します。

1. CS 領域のサイズを決め、「36.4.1. CS 領域設定レジスタ: ASR0～ASR3 (Area Setting Register 0-3)」から ASZ[3:0] の値を選択します。

2. CS 領域の開始アドレスを設定します。開始アドレスはアドレスの上位ビットを SADR に設定します。ただし開始アドレスは、次頁表のように領域のサイズ指定によってあらかじめバウンダリが決められています。「36.4.1. CS 領域設定レジスタ: ASR0～ASR3 (Area Setting Register 0-3)」 にしたがって SADR の有効なビットを設定してください。無効な SADR のビットには"0"を設定してください。

● CS 領域のサイズと ASZ, SADR の設定

CS 領域のサイズ	ASZ[3:0]	有効な SADR のビット
64KB	0000	SADR[31:16]
128KB	0001	SADR[31:17]
256KB	0010	SADR[31:18]
512KB	0011	SADR[31:19]
1MB	0100	SADR[31:20]
2MB	0101	SADR[31:21]
4MB	0110	SADR[31:22]
8MB	0111	SADR[31:23]
16MB	1000	SADR[31:24]
32MB	1001	SADR[31:25]
64MB	1010	SADR[31:26]
128MB	1011	SADR[31:27]
256MB	1100	SADR[31:28]
512MB	1101	SADR[31:29]
1GB	1110	SADR[31:30]
2GB(ASR0 初期値)	1111	SADR[31]

＜注意事項＞
各CS領域は重なり合わないように配置してください。CS領域が重なっている場合の動作は保証いたしません。

SADR と ASZ への設定値と実際に割り当てられる CS の領域との例を以下に示します。

設定例

- **CS0設定**

  ASR0:ASZ[3:0]=0010
  ASR0:SADR[31:16]=0x000C

  → 0x000C0000～0x000FFFFF は CS0 領域になります。

- **CS1設定**

  ASR1:ASZ[3:0]=0000
  ASR1:SADR[31:16]=0x0006

  → 0x00060000～0x0006FFFF は CS1 領域になります。

- **CS2設定**

  0x00110000～ 1M バイトの空間を割り当てたい。

  1M バイトの空間を設けたいので ASZ[3:0]=0100 を設定します。このときの SADR の有効ビットは[31:20]になります。SADR[19:16]はアドレスとの比較対象になりません。したがって CS2 領域の開始アドレスは 0x00110000 とすることはできず 0x00100000 になります。

  ASR2:ASZ[3:0]=0100
  ASR2:SADR[31:16]=0x0010 を設定

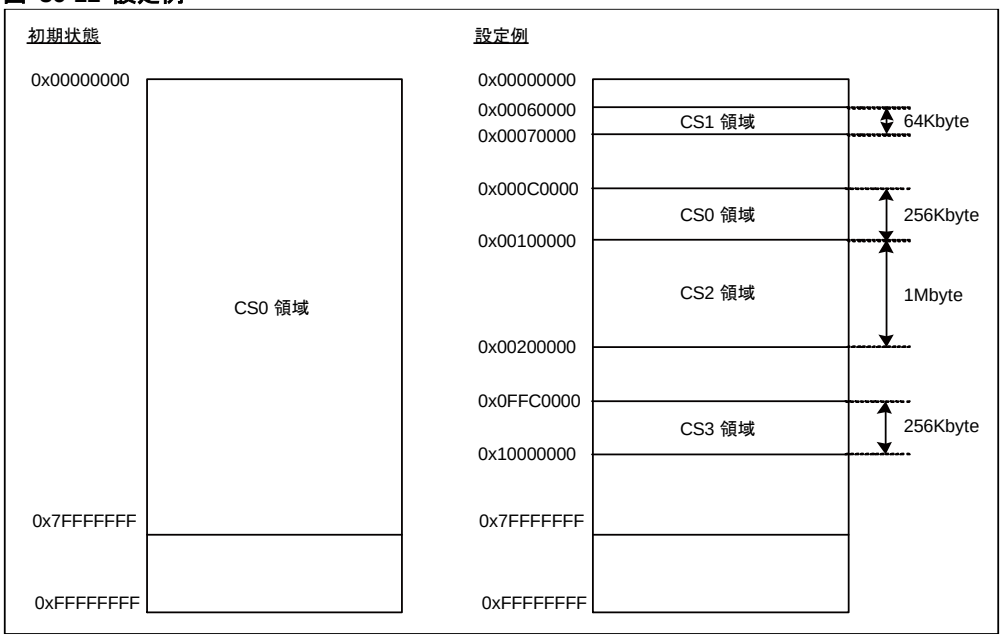
  → 0x00100000～0x001FFFFF は CS2 領域になります。

- **CS3設定**

  ASR3:ASZ[3:0]=0010
  ASR3:SADR[31:16]=0x0FFC

  → 0x0FFC0000～0x0FFFFFFF は CS3 領域になります。

図 36-12 設定例



● ASR の読出し，比較

必要な CS に対して ACR, AWR, ASR の設定を行った後、それ以降のアクセスに対して CS の設定が反映される事を保証する目的で、最後に設定した ASR を読み出して設定値との比較を行ってください。

● CS 設定・変更サンプルプログラム

CS 設定のサンプルプログラムとして CS1 を設定する場合を示します。

図 36-13 CS1 設定サンプルプログラム

■ACR1設定例
下表の場合の設定値を示します。

データバス幅	16bit
アドレス出力タイプ	通常
バスタイプ	アドレス・データスプリットバス
ライトシグナルタイプ	ライトタイプ0

上記設定ビット以外のビットは予約になっているので0を設定します。

ACR1設定値 : 0x40

■AWR1設定例
下表の設定値を示します。

RWT	3サイクル
WWT	4サイクル
RIDL	2サイクル
WRCV	3サイクル
CSRD	1サイクル
RDCS	1サイクル
CSWR	2サイクル
WRCS	2サイクル
ADCY	アドレス・データスプリットバス設定
ACS	0サイクル
ASCY	0サイクル
RDYE	無効

上記設定ビット以外のビットは予約になっているので0を設定します。

AWR1設定値 : 0x034b5a00

■ASR1設定例
・CS1領域サイズ : 64Kbyte
・CS1領域アドレス : 0x0040_0000～0x0040_FFFF
・書き込み許可
・ビックエンディアン
・CS1有効

ASR1設定値 : 0x00400005

■プログラム例

```
_disable_CS0
ldi     #_ASR0, r0          //#_ASR0はASR0のアドレス値
ldi     0x0, r1
st      r1, @r0
_set_ACR1
ldi     #_ACR0, r0          //#_ACR1はACR1のアドレス値
ldi     #0x40, r1           //0x40をACR1に設定
st      r1, @r0
_set_AWR1
ldi     #_AWR1, r0          //#_AWR1はAWR1のアドレス値
ldi     #0x034b5a00 r1      //0x034b5a00をAWR1に設定
st      r1, @r0
_set_ASR1
ldi     #_ASR1, r0          //#_ASR1はASR1のアドレス値
ldi     #0x00400005 r1      //0x00400005をASR0に設定
st      r1, @r0
ld      @r0, r2
cmp     r1, r2              //ASR1の設定値を確認
```

36.5.11 非同期メモリとの接続例

非同期メモリとの接続例について示します。

外バス端子と非同期メモリとの接続例を示します。

図 36-14 SRAM との接続例 1 (8 ビット SRAM×2)

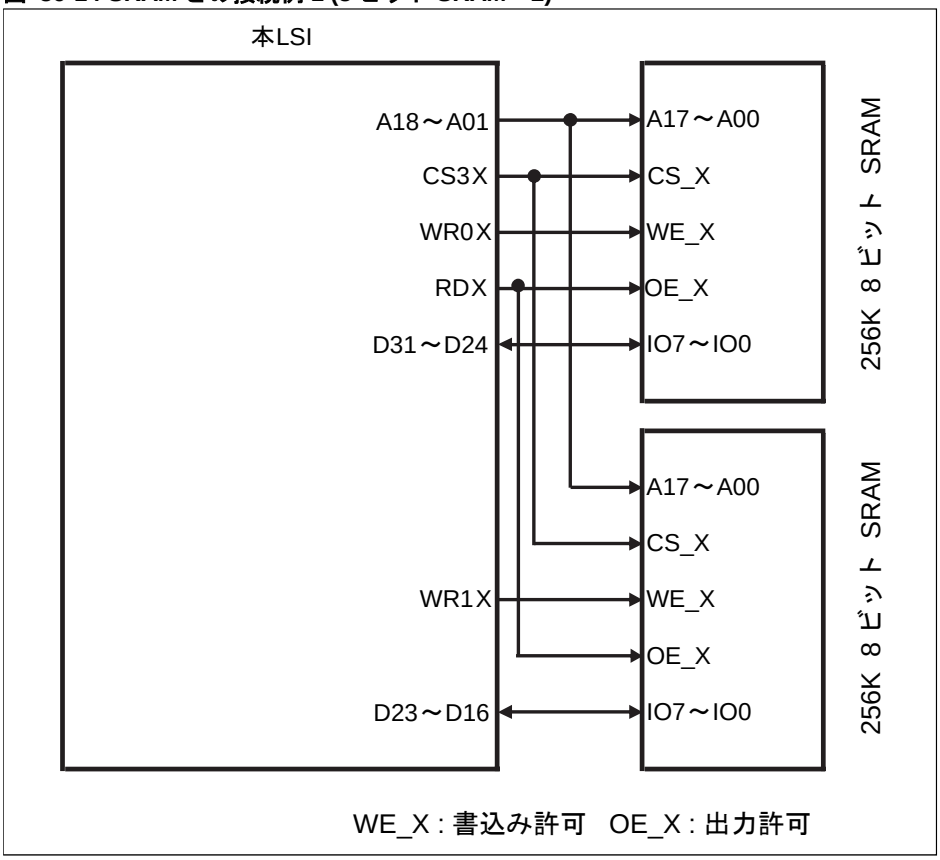
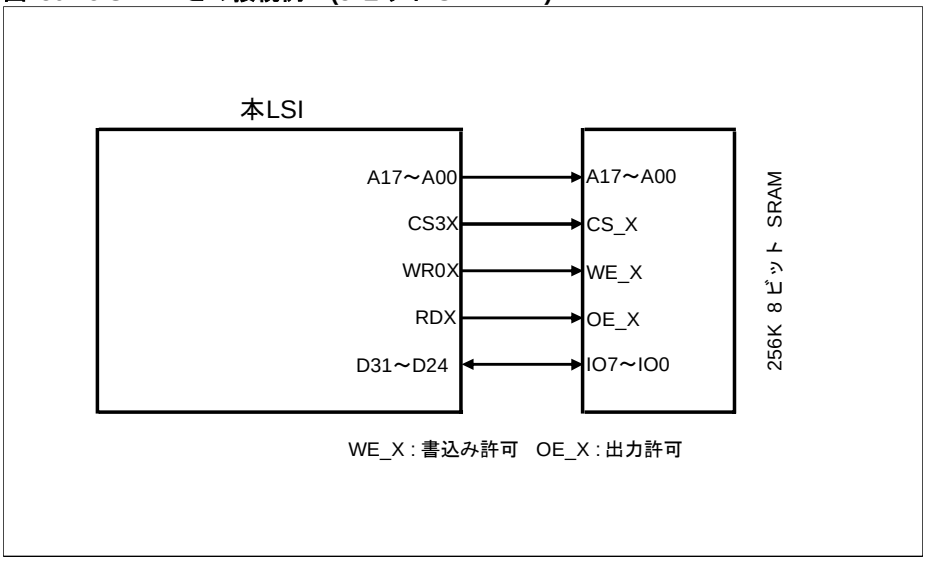


図 36-15 SRAM との接続例 2 (8 ビット SRAM×1)



36.5.12 リトルエンディアンデバイスとの接続例

リトルエンディアンデバイスとの接続例について示します。

リトルエンディアンデバイスとのデータバスおよびバイトイネーブル信号の接続方法を示します。

図 36-16 16 ビットバス幅

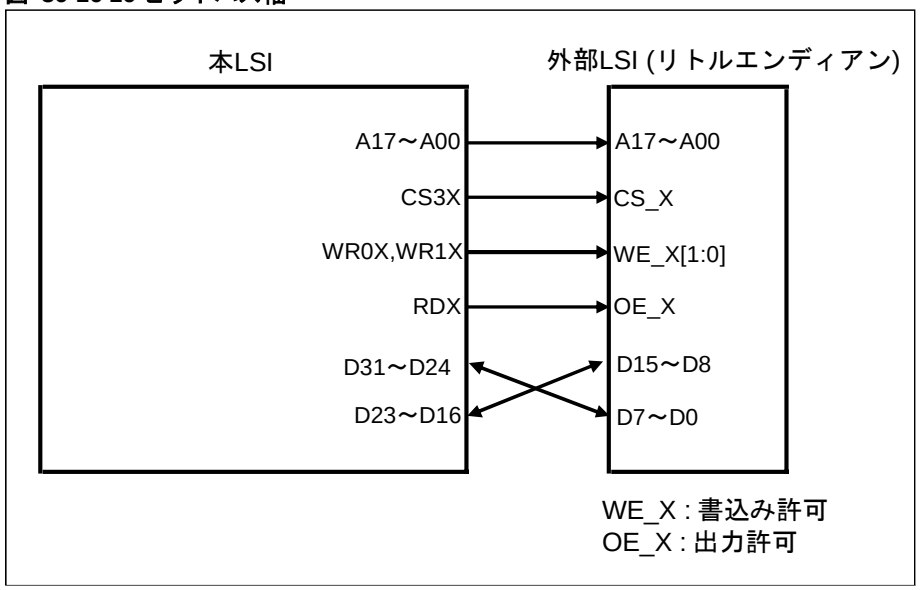
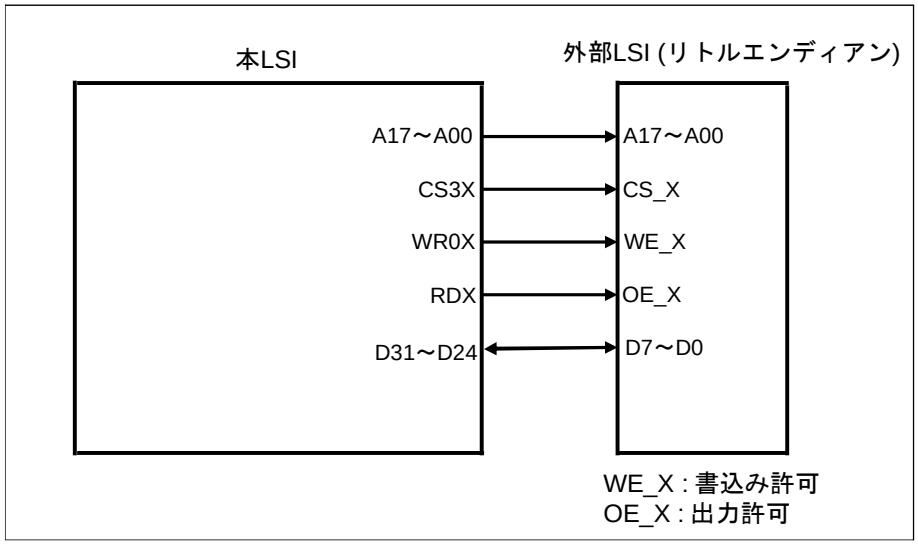


図 36-17 8 ビットバス幅



37. バス・パフォーマンス・カウンタ

バス・パフォーマンス・カウンタについて説明します。

37.1 概要

バス・パフォーマンス・カウンタの概要について説明します。

本シリーズは、オンチップバスのパフォーマンスを測定するバス・パフォーマンス・カウンタ(BPC)を搭載しています。BPC は、オンチップバス上のトラフィックの内訳を計測し、バスのパフォーマンス改善の指針となる情報を提供します。BPC では、オンチップバスがアイドル状態でのカウントは行いませんので、時間計測のためにはシステム内のタイマを同時に使用してください。

37.2 特長

バス・パフォーマンス・カウンタの特長について説明します。

■ カウンタ構成

カウントクロック : オンチップバス用クロック

カウンタビット長 : 32 ビット × 3 チャネル (BPC-A, BPC-B, BPC-C)

オーバフロー検出 : なし

カウンタ値書換え : 可能

■ 主要機能

各チャネルにおいて、以下の動作を選択してカウント

- オンチップバス上のリードアクセス回数
- オンチップバス上のライトアクセス回数
- オンチップバス上のウェイトサイクル数

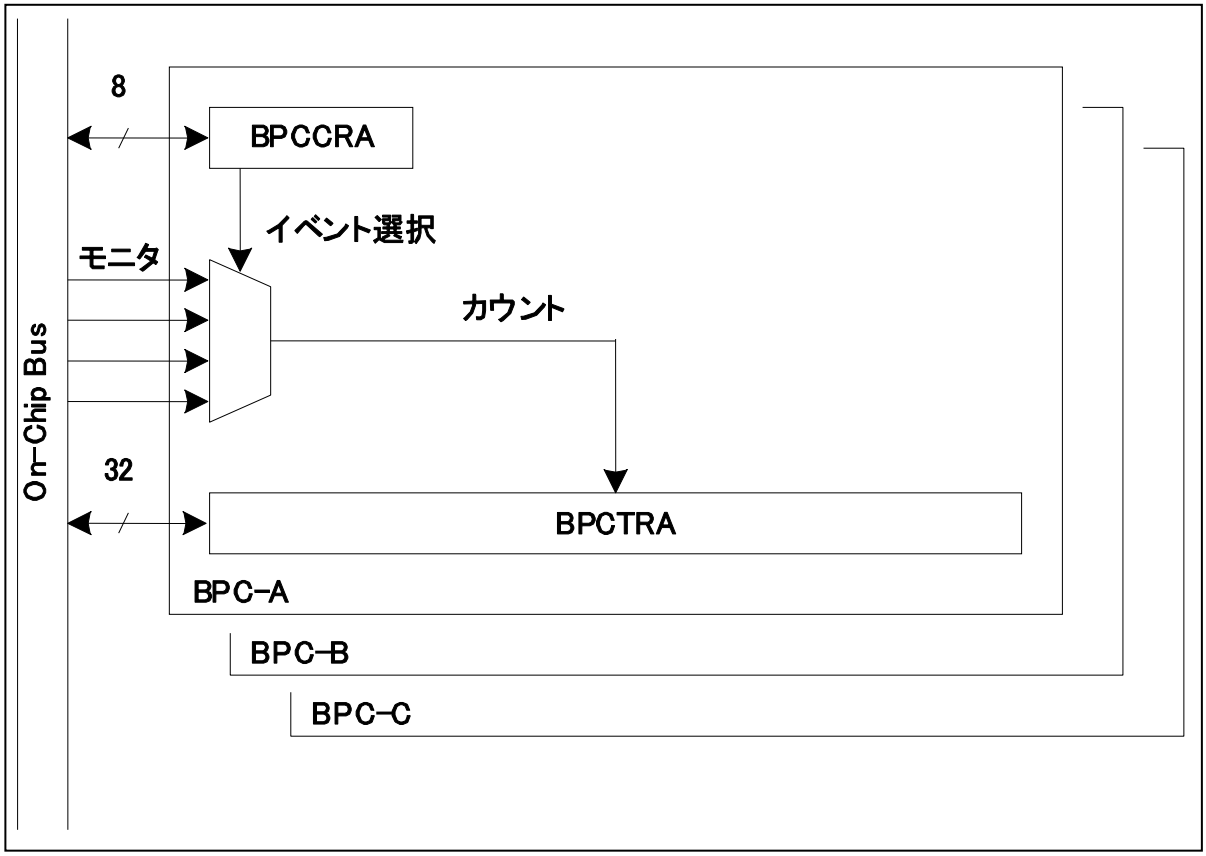
各チャネルにおいて、以下のうちいずれかを選択してカウント

- 特定のバスマスタ(CPU, DMAC, その他,またはすべて)
- 特定のターゲット(ICH, MCH, その他,またはすべて)

37.3 構成

バス・パフォーマンス・カウンタの構成について示します。

図 37-1 ブロックダイヤグラム



37.4 レジスタ

バス・パフォーマンス・カウンタのレジスタについて説明します。

表 37-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0710	BPCCRA	BPCCRB	BPCCRC	予約	BPC-A 制御レジスタ BPC-B 制御レジスタ BPC-C 制御レジスタ
0x0714	BPCTRA				BPC-A カウントレジスタ
0x0718	BPCTRB				BPC-B カウントレジスタ
0x071C	BPCTRC				BPC-C カウントレジスタ

37.4.1 BPC-A 制御レジスタ ：BPCCRA (Bus Performance Counter Control Register A)

BPC-A 制御レジスタのビット構成について説明します。

バス・パフォーマンス・カウンタ A (BPC-A)の測定対象の設定を行います。

バス・パフォーマンス・カウンタには A, B, C の 3 チャネルがありそれぞれのカウンタに対して制御レジスタが存在します。制御レジスタの各フィールドは、各チャネル共通です。

■ BPCCRA : アドレス 0710H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FUNC[1:0]		MST[3:0]				SLV[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7, bit6] FUNC[1:0] (Function Selection) : **測定イベント選択**
BPC で測定するイベントを選択します。

FUNC[1:0]	イベント
00	BPC-A 動作停止(初期値)
01	リードアクセス回数
10	ライトアクセス回数
11	ウェイトサイクル数

[bit5～bit2] MST[3:0] (bus MaSTer select) : **バスマスタ選択**
BPC で測定するイベント対象となるバスマスタを選択します。

MST[3:0]	バスマスタ
0000	すべてのバスマスタ(初期値)
0001	CPU(XBS)
0010	DMAC
0011	予約
0100	予約
0101 ～ 1111	予約

[bit1, bit0] SLV[1:0] (SLaVe select) : **スレーブ選択**
BPC で測定するイベント対象となるスレーブを選択します。

SLV[1:0]	スレーブ
00	すべてのスレーブ(初期値)
01	MCH(レジスタ, 外バス)
10	ICH(ペリフェラル)
11	MCH/ICH 以外

37.4.2 BPC-B 制御レジスタ ：BPCCRB (Bus Performance Counter Control Register B)

BPC-B 制御レジスタのビット構成について説明します。

バス・パフォーマンス・カウンタ B (BPC-B)の測定対象の設定を行います。

各ビットの機能は BPCCRA と等価です。

■ BPCCRB： アドレス 0711H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FUNC[1:0]		MST[3:0]				SLV[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

37.4.3 BPC-C 制御レジスタ ：BPCCRC (Bus Performance Counter Control Register C)

BPC-C 制御レジスタのビット構成について説明します。

バス・パフォーマンス・カウンタ C (BPC-C)の測定対象の設定を行います。

各ビットの機能は BPCCRA と等価です。

■ BPCCRC： アドレス 0712H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FUNC[1:0]		MST[3:0]				SLV[1:0]	
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

37.4.4 BPC-A カウントレジスタ ：BPCTRA (Bus Performance CounTer Register A)

BPC-A カウントレジスタのビット構成について説明します。

BPCCRA レジスタで設定したイベントをカウントする 32 ビット長のカウントレジスタです。

■ BPCTRA： アドレス 0714H (アクセス: ワード)

	bit31	bit30	・	・	・	bit3	bit2	bit1	bit0
	BPCTRA[31:0]								
初期値	0	0	・	・	・	0	0	0	0
属性	R/W	R/W	・	・	・	R/W	R/W	R/W	R/W

[bit31～bit0] BPCTRA[31:0] (Bus Performance CounTer Register A) : BPC-A カウント
BPCCRA レジスタの bit7,bit6:FUNC に"00"以外を設定すると、対象イベントのカウントを開始します。本レジスタはリードライト可能で、32 ビットアクセスのみ可能です。カウント開始時にカウンタの初期は行いませんので、新たにカウントを始める場合は初期値の設定を行ってください。またオーバフロー制御は行いませんので、カウンタがオーバフローした場合は"0"に戻ってカウントを継続します。

37.4.5 BPC-B カウントレジスタ ：BPCTRB (Bus Performance CounTer Register B)

BPC-B カウントレジスタのビット構成について説明します。

BPCCRB レジスタで設定したイベントをカウントする 32 ビット長のカウントレジスタです。使用法は BPCTRA と同じです。

■ BPCTRB： アドレス 0718H (アクセス: ワード)

	bit31	bit30	・	・	・	bit3	bit2	bit1	bit0
	BPCTRB[31:0]								
初期値	0	0	・	・	・	0	0	0	0
属性	R/W	R/W	・	・	・	R/W	R/W	R/W	R/W

37.4.6 BPC-C カウントレジスタ ：BPCTRC (Bus Performance CounTer Register C)

BPC-C カウントレジスタのビット構成について説明します。

BPCCRC レジスタで設定したイベントをカウントする 32 ビット長のカウントレジスタです。使用法は BPCTRA と同じです。

■ BPCTRC： アドレス 071C$_H$ (アクセス: ワード)

	bit31	bit30	・	・	・	bit3	bit2	bit1	bit0
	BPCTRC[31:0]								
初期値	0	0	・	・	・	0	0	0	0
属性	R/W	R/W	・	・	・	R/W	R/W	R/W	R/W

37.5 動作説明

バス・パフォーマンス・カウンタの動作について説明します。

37.5.1. 設定

37.5.2. 起動と停止

37.5.3. 動作

37.5.4. 測定と結果処理

37.5.1 設定

バス・パフォーマンス・カウンタの設定について説明します。

BPC の各チャネルを起動する前に、BPCTRA/BPCTRB/BPCTRC へ"0x00000000"を書込み、各カウンタの初期化をしてください。測定対象を変更する際も、同様に各カウンタの初期化をしてください。リセット後はカウンタ値が不定のため、動作許可する前に必ずカウンタ値を書き込んでください。

BPC の各チャネルを起動する際に、BPCCRA/BPCCRB/BPCCRC により、各カウンタの測定対象を設定します。

バス・パフォーマンス・カウンタ A(B, C)制御レジスタ (BPCCRA(B, C))の設定により監視するイベントは以下のようになります。下記の表に存在しない組み合わせの場合は、動作を保証しません。 また、エミュレータモード時はカウントを行いません。

表 37-2 BPC 設定一覧

FUNC[1:0]	MST[3:0]	SLV[1:0]	対象イベント
01	0000	00	XBS,DMAC からリードアクセス
		01	XBS,DMAC から MCH リード
		10	XBS,DMAC から ICH リード
		11	XBS,DMAC から MCH/ICH 以外のリード
	0001	00	XBS からリードアクセス
		01	XBS から MCH リード
		10	XBS から ICH リード
		11	XBS から MCH/ICH 以外へのリード
	0100	00	DMAC からリードアクセス
		01	DMAC から MCH リード
		10	DMAC から ICH リード
		11	DMAC から MCH/ICH 以外へのリード
10	0000	00	XBS,DMAC からライトアクセス
		01	XBS,DMAC から MCH ライト
		10	XBS,DMAC から ICH ライト
		11	XBS,DMAC から MCH/ICH 以外へのライト
	0001	00	XBS からライトアクセス
		01	XBS から MCH ライト
		10	XBS から ICH ライト
		11	XBS から MCH/ICH 以外へのライト
	0100	00	DMAC からライトアクセス
		01	DMAC から MCH ライト
		10	DMAC から ICH ライト
		11	DMAC から MCH/ICH 以外へのライト
11	0000	00	XBS,DMAC のウェイトアクセス
		01	XBS,DMAC から MCH ウェイト
		10	XBS,DMAC から ICH ウェイト
		11	XBS,DMAC から MCH/ICH 以外のウェイト
	0001	00	XBS からウェイトアクセス
		01	XBS から MCH ウェイト
		10	XBS から ICH ウェイト
		11	XBS から MCH/ICH 以外へのウェイト
	0100	00	DMAC からウェイトアクセス
		01	DMAC から MCH ウェイト
		10	DMAC から ICH ウェイト
		11	DMAC から MCH/ICH 以外へのウェイト

37.5.2 起動と停止

バス・パフォーマンス・カウンタの起動と停止について説明します。

バス・パフォーマンス・カウンタ A 制御レジスタ (BPCCRA)の FUNC[1:0]ビットを"00"以外の値に設定することにより、対象イベントのカウントを開始します。ただし、このときバス・パフォーマンス・カウンタ A レジスタ (BPCTRA)は初期化せず、その時点の値からカウント開始します。BPCCRA:FUNC[1:0]を"00"に設定するとバス・パフォーマンス・カウンタ動作を停止します

37.5.3 動作

バス・パフォーマンス・カウンタの動作について説明します。

制御レジスタの設定により動作許可されると、以降オンチップバスが動作中は各測定対象動作のカウントを続けます。ただし、以下に示す状態ではカウントを一時停止します。

■ エミュレータモード中

各種低消費電力制御を設定している場合のカウント動作は次のようになります。

- CPU スリープモード
  各種測定対象動作をカウントします。
- バススリープモード
  オンチップバスが動作する DMA 転送時にのみカウントします。それ以外の期間は、測定対象動作が発生しませんので、カウントを行いません。
- スタンバイモード(時計モード/ストップモード)
  測定対象動作が発生しませんので、カウントを行いません。

リセットの発生により、制御レジスタが初期化されます。リセット発生直後は、カウントを行いません。

37.5.4 測定と結果処理

バス・パフォーマンス・カウンタの測定と結果処理について説明します。

BPC の使用は、ICE 接続時あるいはモニタデバッガ利用時を想定しています。測定の設定および結果の読出しは、ユーザプログラムの実行が中断しているデバッグモードにて行います。

測定例としては、以下のようなものがあります。

- ユーザプログラム上の 2 点間の測定
- 基準時間ベースの測定

以下、これらについて説明します。

- ユーザプログラム上の 2 点間の測定
  この測定では、ユーザプログラム上の測定開始ポイントおよび測定終了ポイントを以下のとおりに設定します。
  - 測定開始ポイント：ユーザプログラムの実行開始ポイント
  - 測定終了ポイント：ユーザプログラムのブレークポイント

測定フローを以下に示します。

1. デバッグモードにて測定の設定およびカウンタの初期化を行う
2. 測定開始ポイントからユーザプログラムを実行開始する
3. 測定終了ポイントにてブレークし、ユーザプログラムを実行終了する
4. デバッグモードへ移行し、測定結果を読み出す

- 基準時間ベースの測定
  この測定では、基準時間ごとにデバッグモードに移行して、測定結果の読出しとカウンタの初期化を行います。

  基準時間ごとのデバッグモード移行方法としては、次の 2 つが挙げられます。

  - ICE 側から基準時間ごとにツールブレークをアサートし、デバッグモードに移行する(ICE 接続時)
  - 内蔵タイマのインターバル時間に基準時間を設定し、タイマ割込みルーチンにおける INTE 命令実行でデバッグモードへ移行する

測定フローを以下に示します。

1. デバッグモードにて測定の設定およびカウンタの初期化を行う
2. 測定対象のユーザプログラムを実行開始する
3. 基準時間によるツールブレーク、あるいは、内蔵タイマの割込みルーチンにおける INTE 命令実行
4. デバッグモードへ移行し、測定結果を読み出す
5. 測定カウンタの初期化
6. 以下、2.～5.を繰り返す

測定結果は、Softune Workbench などのデバッガのホストプログラムにて解析します。解析結果は、直感的に理解できるようにグラフ (円グラフ, 棒グラフ, 折れ線グラフなど)にて視覚的に表示され､ユーザのプログラムチューニングに有益な情報を提供します(バス・パフォーマンス・アナライザ機能)。以下は、解析例です。

解析例 :

1. バスマスタのアクセス比率
   例. CPU アクセス対 DMAC アクセスの比率, 全アクセスに占める特定バスマスタアクセスなど
2. 発生イベント比率
   例. リードアクセス対ライトアクセスの比率, 全サイクルに占めるウェイトサイクル比率など
3. ターゲットの被アクセス比率
   例. MCH 対 ICH の比率, 全アクセスに占める特定ターゲットへのアクセス比率など
4. 特定バスマスタから特定ターゲットへの特定アクセス比率
   例. 全アクセスに占める CPU から MCH へのリードアクセスの比率など
5. 特定ターゲットでのウェイトサイクル発生比率
   例. 全ウェイトサイクルに占める MCH アクセス時のウェイトサイクル比率など
6. プログラムの特定 2 点間の各種バス動作解析
   例. プログラムの特定 2 点間の全サイクルに占めるリード/ライト/ウェイトサイクル比率など
7. 一定時間ごとの時間推移における各種バス動作解析
   例. 全アクセスに占める特定バスマスタの特定ターゲットへのアクセス比率の時間推移、など

38. CRC

CRC (Cyclic Redundancy Check) について説明します。

38.1 概要

CRC (Cyclic Redundancy Check)の概要について説明します。

本モジュールは、CRC 値を計算します。

CRC (Cyclic Redundancy Check)とは、誤り検出方式の一種です。入力データ列を高次の多項式とみなして、あらかじめ定められた生成多項式 (Generator Polynomial)で割ったときの余りが CRC コードです。通常は、データ列の後ろに CRC コードを付けて送信し、受信データに対して同様に生成多項式で割り算を行い、余りがなければ受信データは正しいと判断します。

38.2 特長

CRC (Cyclic Redundancy Check)の特長について説明します。

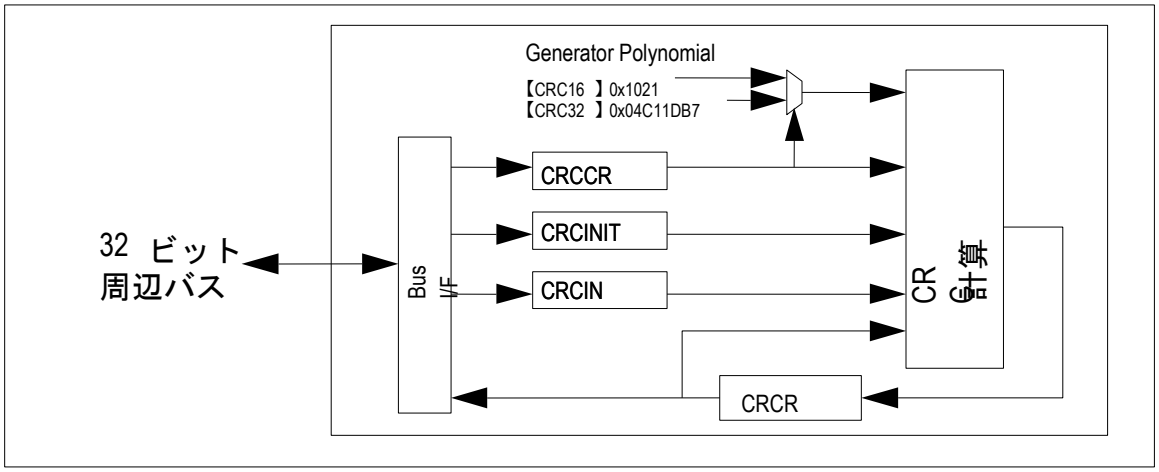
本モジュールでは、CCITT CRC16 と、IEEE-802.3 CRC32 を計算できます。本モジュールでは生成多項式はこれら 2 つ用の数値に固定されていますので、ほかの生成多項式に基づく CRC 値の計算はできません。

- CCITT CRC16 生成多項式 : 0x1021
- IEEE-802.3 CRC32 生成多項式 : 0x04C11DB7

38.3 構成

CRC (Cyclic Redundancy Check) の構成について示します。

図 38-1 ブロックダイヤグラム



38.4 レジスタ

CRC (Cyclic Redundancy Check) のレジスタについて説明します。

表 38-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x1130	予約			CRCCR	CRC 制御レジスタ
0x1134	CRCINIT				CRC 初期値レジスタ
0x1138	CRCIN				Input Data レジスタ
0x113C	CRCR				CRC レジスタ

38.4.1 CRC 制御レジスタ ：CRCCR (CRC Control Register)

CRC 制御レジスタのビット構成について説明します。

CRC 計算の制御を行います。

■ CRCCR： アドレス 1133H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	FXOR	CRCLSF	CRCLTE	LSBFST	LTLEND	CRC32	INIT
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R/W	R/W	R/W	R/W	R/W	R/W	R0,W

[bit7] 予約
必ず"0"を書き込んでください。

[bit6] FXOR (Final XOR) : Final XOR 制御ビット
CRC 結果を XOR 値と XOR して出力します。XOR 値は ALL "H"で、FXOR=1 時はビット反転となります。CRC レジスタの後段で処理を行いますので、本ビット設定後すぐに CRC 結果は読出し値に反映されます。

[bit5] CRCLSF (CRC result LSb First) : CRC 結果ビットオーダ設定ビット
CRC 結果のビットオーダ設定ビットです。 バイト内のビット並び替えを行います。"0"のとき MSB First、"1"のとき LSB First になります。CRC レジスタの後段で処理を行いますので、本ビット設定後すぐに CRC 結果を読出し値に反映されます。

[bit4] CRCLTE (CRC result LiTtle Endian) : CRC 結果バイトオーダ設定ビット
CRC 結果のバイトオーダ設定ビットです。ワード内のバイトオーダ並び替えを行います。"0"のときビッグエンディアン、"1"のときリトルエンディアンになります。CRC レジスタの後段で処理を行いますので、本ビット設定後すぐに CRC 結果読出し値に反映されます。CRC16 の場合に本ビットを"1"にすると、31～16 ビット目に出力となります。

[bit3] LSBFST (LSB FirST) : ビットオーダ設定ビット
ビットオーダ設定ビットです。 バイト(8 ビット)の先頭ビットを指定します。"0"のとき MSB First、"1"のとき LSB First になります。LTLEND ビットの設定と組み合わせて、4 とおりの処理順を指定可能です。

[bit2] LTLEND (LitTtLe ENDian) : バイトオーダ設定ビット
バイトオーダ設定ビットです。書込み幅でのバイト配置順を指定します。"0"のときビッグエンディアン、"1"のときリトルエンディアンになります。

[bit1] CRC32 (CRC32) : CRC モード選択ビット
CRC16 と CRC32 のモード選択ビットです。CRC32=1 のとき、CRC32 の演算モードになります。

[bit0] INIT (INITialize) : 初期化ビット
初期化ビットです。ソフトウェアにより、本ビットに"1"を書き込むと、初期化が行われます。このビットは値を持たず、読込み時は常に"0"を返します。初期化では、初期値レジスタの値が、ハードウェアにより CRC レジスタにロードされます。初期化は、CRC 計算の最初に 1 度実行してください。

38.4.2 CRC 初期値レジスタ : CRCINIT (CRC Initial value register)

CRC 初期値レジスタのビット構成について説明します。

CRC 計算の初期値を設定します。

■ CRCINIT : アドレス 1134H (アクセス: バイト, ハーフワード, ワード)

	bit31	bit30	・	・	・	bit2	bit1	bit0
	D[31:0]							
初期値	1	1	・	・	・	1	1	1
属性	R/W	R/W	・	・	・	R/W	R/W	R/W

[bit31～bit0] D[31:0] (Data) : 初期値ビット
CRC 計算の初期値を記憶します。 ソフトウェアは、CRC 計算の初期値を書き込みします。(リセット後は 0xFFFF_FFFF) CRC16 のときは、D15～D0 を使用し、D31～D16 は無視します。

38.4.3 Input Data レジスタ : CRCIN (CRC INput data register)

Input Data レジスタのビット構成について説明します。

CRC 計算の入力データを設定します。

■ CRCIN : アドレス 1138H (アクセス: バイト, ハーフワード, ワード)

	bit31	bit30	・	・	・	bit2	bit1	bit0
	D[31:0]							
初期値	0	0	・	・	・	0	0	0
属性	R/W	R/W	・	・	・	R/W	R/W	R/W

[bit31～bit0] D[31:0] (Data) : Input Data ビット
CRC 計算の入力データを設定します。ソフトウェアは、CRC 計算の入力データを書き込みします。ビット幅は 8, 16, 32 に対応します。混在も可能です。バイト書込み、ハーフワード書込み時の、書込み位置は任意です。取り得るアドレス位置は、バイト書込み : +0, +1, +2, +3、ハーフワード書込み : +0, +2。

38.4.4 CRC レジスタ ：CRCR (CRC Register)

CRC レジスタのビット構成について説明します。

CRC 計算の結果を出力します。

■ CRCR： アドレス 113CH (アクセス: バイト, ハーフワード, ワード)

	bit31	bit30	・	・	・	bit2	bit1	bit0
	D[31:0]							
初期値	1	1	・	・	・	1	1	1
属性	R,WX	R,WX	・	・	・	R,WX	R,WX	R,WX

[bit31～bit0] D[31:0] (Data) : CRC ビット
CRC 計算の結果を出力します。 ソフトウェアが、初期化ビット(CRCCR:INIT)に"1"を書き込むと、初期値レジスタ(CRCINIT)の値が本レジスタにロードされます。ソフトウェアが、CRC 計算の入力データを Input Data レジスタ(CRCIN)に書込みすると、ハードウェアによって、直ちに CRC 計算結果が本レジスタに設定されます。すべての入力データ書込みが完了したとき、本レジスタは最終的な CRC コードを保持していることになります。CRC16 の場合は、バイトオーダがビッグエンディアン(CRCLTE=0)のときは D15～D0、リトルエンディアン(CRCLTE=1)のときは D31～D16 の位置に結果が出力されます。

38.5 動作説明

CRC (Cyclic Redundancy Check)の動作について説明します。

38.5.1. CRC の定義

38.5.2. リセット動作

38.5.3. 初期化

38.5.4. バイトオーダと、ビットオーダ

38.5.5. CRC 計算シーケンス

38.5.6. 使用例

38.5.1 CRC の定義

CRC (Cyclic Redundancy Check) の定義について説明します。

● CCITT CRC16 Standard

生成多項式	0x1021	(CRCCR:CRC32=0)
初期値	0xFFFF	
Final XOR 値	0x0000	(CRCCR:FXOR=0)
ビットオーダ	MSB First	(CRCCR:LSBFST=0)
出力ビットオーダ	MSB First	(CRCCR:CRCLSF=0)

(入出力のバイトオーダは任意に設定可能)

● IEEE-802.3 CRC32 Ethernet Standard

生成多項式	0x04C11DB7	(CRCCR:CRC32=1)
初期値	0xFFFF_FFFF	
Final XOR 値	0xFFFF_FFFF	(CRCCR:FXOR=1)
ビットオーダ	LSB First	(CRCCR:LSBFST=1)
出力ビットオーダ	LSB First	(CRCCR:CRCLSF=1)

(入出力のバイトオーダは任意に設定可能)

38.5.2　リセット動作

CRC (Cyclic Redundancy Check)のリセット動作について説明します。

リセット時は、CRC 初期値レジスタ(CRCINIT)と、CRC レジスタ(CRCR)を 0xFFFF_FFFF に設定します。その他のレジスタは 0 クリアです。

38.5.3　初期化

CRC (Cyclic Redundancy Check)の初期化について説明します。

CRCCR:INIT による初期化では、初期値レジスタの値を CRC レジスタ(CRCR)にロードします。

38.5.4　バイトオーダと、ビットオーダ

CRC (Cyclic Redundancy Check)のバイトオーダ，ビットオーダについて説明します。

例を用いて説明します。次の 1 ワードを CRC 演算器に入力します。

133.82.171.1 = 10000101　　01010010 10101011 00000001

バイトオーダをビッグエンディアン(CRCCR:LTLEND=0)とすると、バイト単位の送信順序は、

10000101	01010010	10101011	00000001
(1 番目)	(2 番目)	(3 番目)	(4 番目)

ビットオーダを LSB First(CRCCR:LSBFST=1)とするとビット単位の送信順序は、

10100001	01001010	11010101	10000000
(先頭)			(最後)

＜注意事項＞

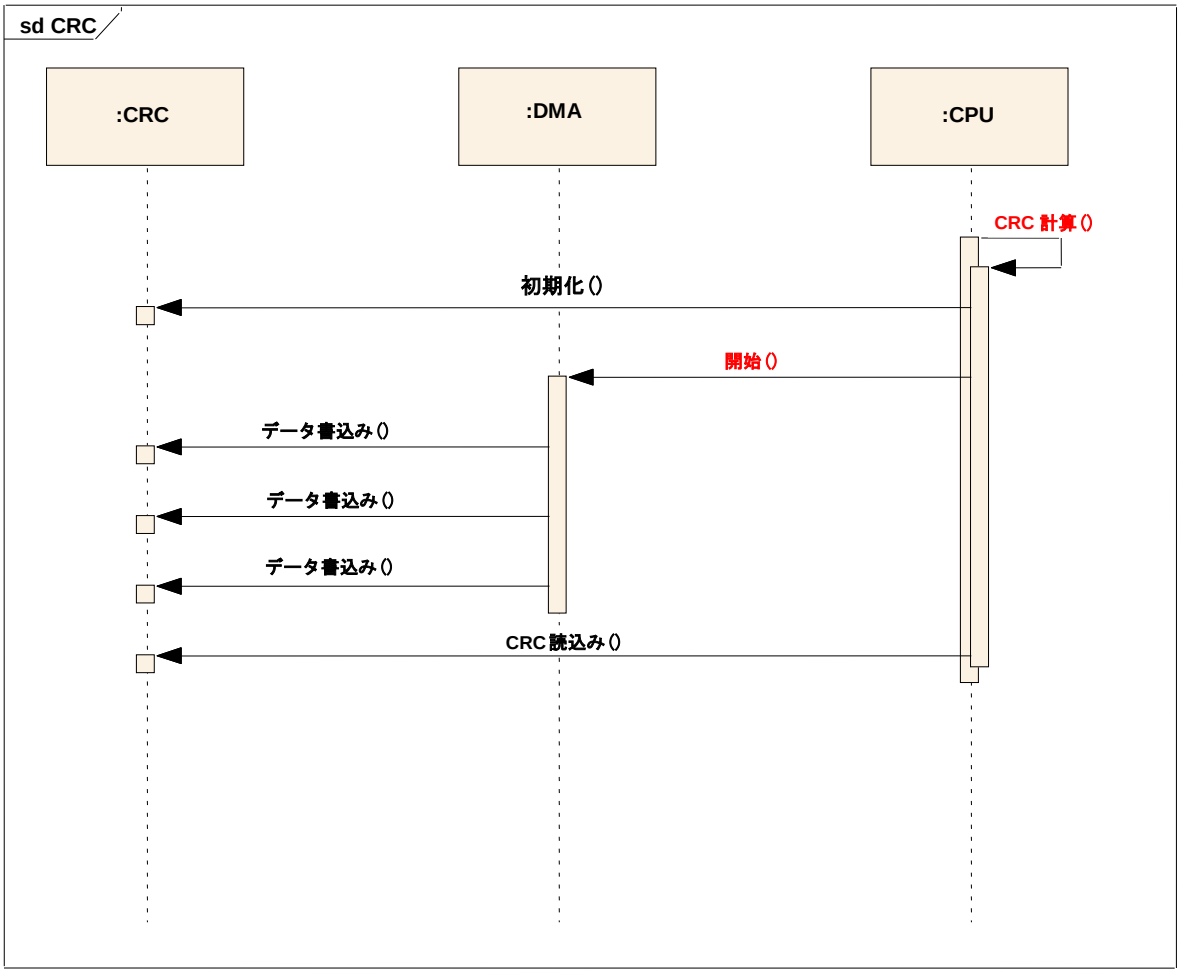
- CRCCR:CRCLTE=1 のとき、CRC 結果は、CRC16, CRC32 ともに 32 ビット幅でのバイト並び替えとなります。
- 特に CRC16 のときは bit31～bit16 の位置が出力となるため、注意してください。

38.5.5 CRC 計算シーケンス

CRC (Cyclic Redundancy Check)の CRC 計算シーケンスについて説明します。

CRC 計算のシーケンスを次に示します。CRC 初期値レジスタ(CRCINIT)の設定, CRC16/32 の選択(CRCCR:CRC32), バイトオーダ・ビットオーダの設定(CRCCR:LTLEND, CRCCR:LSBFST)は既にされているとします。(初期値が ALL "H"でよい場合は、初期値レジスタ(CRCINIT)の設定操作は省略可能です)

図 38-2 CRC 計算シーケンス



- 初期化は、初期化ビット(CRCCR:INIT)への"1"書込みで行います。CRC レジスタ(CRCR)に初期値レジスタの値がロードされます。
- 入力データ書込みは、CRC Input Data レジスタ(CRCIN)への書込みで行います。書込み操作により、CRC 計算が開始されます。連続書込みに対応します。また、異なるビット幅書込みをシーケンス中に混在させることが可能です。
- CRC コード取得は、CRC レジスタ(CRCR)の読込みで行います。

38.5.6　使用例

CRC (Cyclic Redundancy Check)の使用例について説明します。

38.5.6.1. 使用例 1　CRC16, バイト入力固定

38.5.6.2. 使用例 2　CRC16, 入力ビット幅異種混在

38.5.6.3. 使用例 3　CRC32, バイトオーダ, ビッグエンディアン

38.5.6.4. 使用例 4　CRC32, バイトオーダ, リトルエンディアン

38.5.6.1　使用例 1　CRC16, バイト入力固定

使用例 1　CRC16, バイト入力固定について示します。

図 38-3 使用例 1

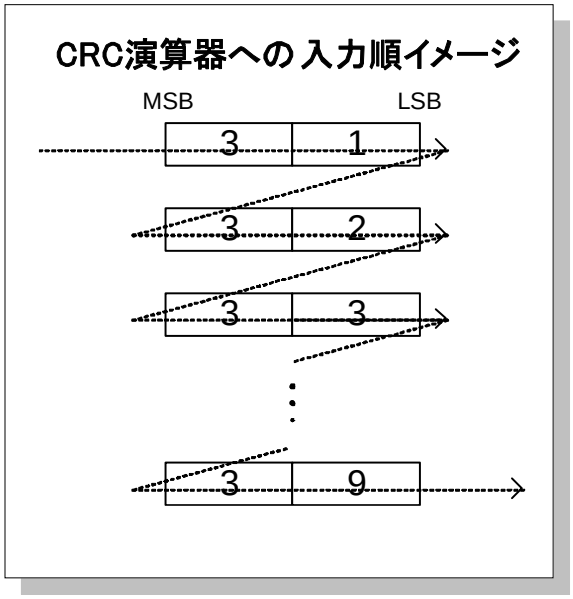
```
//****************************************
// CRC16 (CRC ITU-T)
// polynomial: 0x1021
// initial value: 0xFFFF
// CRCCR.CRC32:  0 // CRC16
// CRCCR.LTLEND: 0 // big endian
// CRCCR.LSBFST: 0 // MSB First
// CRCCR.CRCLTE: 0 // CRC big endian
// CRCCR.CRCLSF: 0 // CRC MSB First
// CRCCR.FXOR:   0 // CRC Final XOR off
//****************************************

//
// 例1－1 (byte単位書込み)
//

// 初期化
B_WRITE (CRCCR, 0x01);

// data write "123456789"
B_WRITE (CRCIN, 0x31);
B_WRITE (CRCIN, 0x32);
B_WRITE (CRCIN, 0x33);
B_WRITE (CRCIN, 0x34);
B_WRITE (CRCIN, 0x35);
B_WRITE (CRCIN, 0x36);
B_WRITE (CRCIN, 0x37);
B_WRITE (CRCIN, 0x38);
B_WRITE (CRCIN, 0x39);

// read result
H_READ (CRCR+2, data);

// check result
assert (data == 0x29B1);

//
// 例1－2 (CRCチェック)
//

// 初期化
B_WRITE (CRCCR, 0x01);

// data write "123456789" + CRC
B_WRITE (CRCIN, 0x31);
B_WRITE (CRCIN, 0x32);
B_WRITE (CRCIN, 0x33);
B_WRITE (CRCIN, 0x34);
B_WRITE (CRCIN, 0x35);
B_WRITE (CRCIN, 0x36);
B_WRITE (CRCIN, 0x37);
B_WRITE (CRCIN, 0x38);
B_WRITE (CRCIN, 0x39);
B_WRITE (CRCIN, 0x29); // <-- CRC
B_WRITE (CRCIN, 0xB1); // <-- CRC

// read result
H_READ (CRCR+2, data);

// check result
assert (data == 0x0000);
```

（以下を想定）

B_WRITE　— バイト書込み
H_WRITE　— ハーフワード書込み
W_WRITE　— ワード書込み

B_READ　— バイト読込み
H_READ　— ハーフワード読込み
W_READ　— ワード読込み

CRCCR　— 制御レジスタアドレス
CRCINIT　— 初期値レジスタアドレス
CRCIN　— 入力データレジスタアドレス
CRCR　— カレントCRCレジスタアドレス



- バイト・ハーフワードの書込み位置は任意です。本使用例では+0 の位置に連続して書込みをしています。
- CRC16 で、CRC 結果のバイトオーダがビッグエンディアンの場合は bit15～bit0 の位置に結果が出力となりますので、H_READ (ハーフワード読込み)のアドレスは+2 としています。

38.5.6.2 使用例 2 CRC16, 入力ビット幅異種混在

使用例 2 CRC16, 入力ビット幅異種混在について示します。

図 38-4 使用例 2

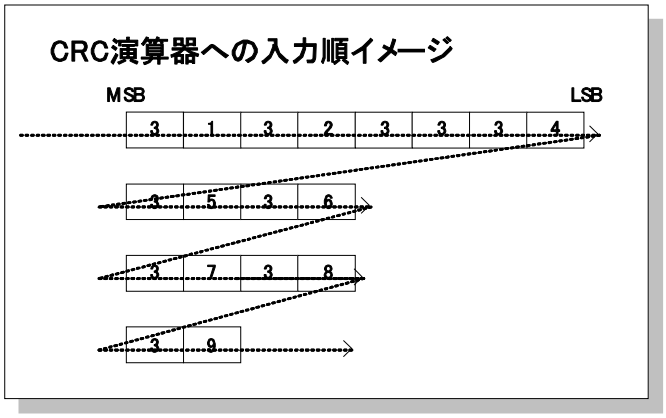
```
//*************************************
// CRC16 (CRC ITU-T)
// polynomial:    0x1021
// initial value: 0xFFFF
// CRCCR.CRC32    0 // CRC16
// CRCCR.LTLEND:  0 // big endian
// CRCCR.LSBFST:  0 // MSB First
// CRCCR.CRCLTE:  0 // CRC big endian
// CRCCR.CRCLSF:  0 // CRC MSB First
// CRCCR.FXOR:    0 // CRC Final XOR off
//*************************************

//
//  例 2－1(書込みサイズ混在)
//

// 初期化
B_WRITE (CRCCR, 0x01);

// data write "123456789"
W_WRITE (CRCIN, 0x31323334);
H_WRITE (CRCIN, 0x3556);
H_WRITE (CRCIN+2, 0x3738);
B_WRITE (CRCIN+3, 0x39);

// read result
H_READ (CRCR+2, data);

// check result
assert (data == 0x29B1);

//
// 例 2－2(CRC チェック)
//

// 初期化
B_WRITE (CRCCR, 0x01);

// data write "123456789" + CRC
W_WRITE (CRCIN, 0x31313334);
W_WRITE (CRCIN, 0x35363738);
H_WRITE (CRCIN, 0x3929); // <-- CRC(0x29)
B_WRITE (CRCIN, 0xB1);   // <-- CRC(0xB1)

// read result
H_READ (CRCR+2, data);

// check result
assert (data == 0x0000);
```

（以下を想定）

B_WRITE — バイト書込み
H_WRITE — ハーフワード書込み
W_WRITE — ワード書込み

B_READ — バイト読込み
H_READ — ハーフワード読込み
W_READ — ワード読込み

CRCCR — 制御レジスタアドレス
CRCINIT — 初期値レジスタアドレス
CRCIN — 入力データレジスタアドレス
CRCR — カレントCRCレジスタアドレス



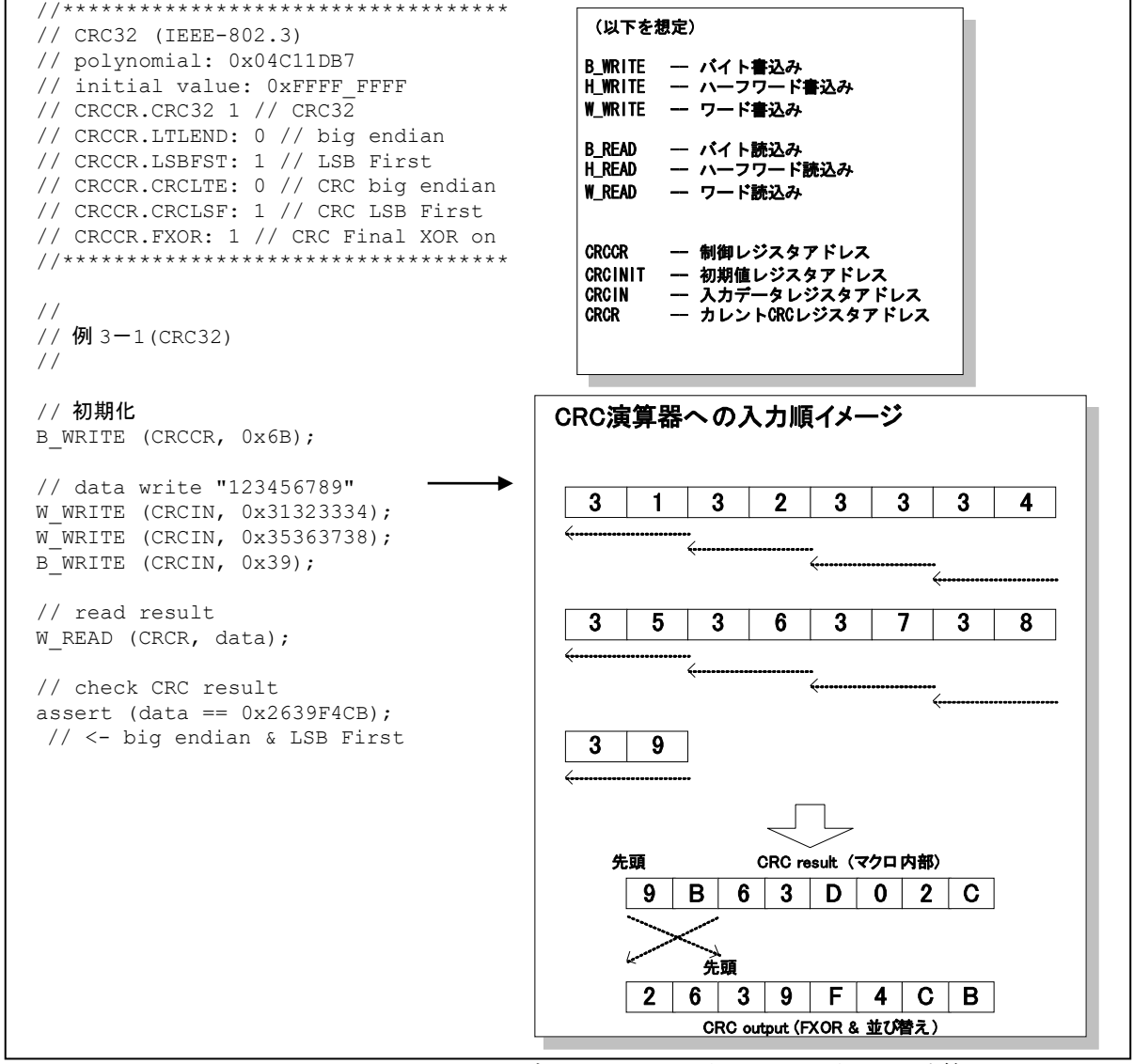
- バイトオーダ, ビットオーダを正しく設定し、CRC 演算器へのビット入力順が同じであれば、書込み幅は任意にできます。
- 例えば、基本をワード書込みとし、最後に 1, 2, 3 バイトの端数が出た場合に、バイト/ハーフワード書込みが混在するケースが考えられます。

38.5.6.3 使用例 3 CRC32, バイトオーダ, ビッグエンディアン

使用例 3 CRC32, バイトオーダ, ビッグエンディアンについて示します。

図 38-5 使用例 3

```
//**********************************
// CRC32 (IEEE-802.3)
// polynomial: 0x04C11DB7
// initial value: 0xFFFF_FFFF
// CRCCR.CRC32 1 // CRC32
// CRCCR.LTLEND: 0 // big endian
// CRCCR.LSBFST: 1 // LSB First
// CRCCR.CRCLTE: 0 // CRC big endian
// CRCCR.CRCLSF: 1 // CRC LSB First
// CRCCR.FXOR: 1 // CRC Final XOR on
//**********************************

//
// 例 3－1(CRC32)
//

// 初期化
B_WRITE (CRCCR, 0x6B);

// data write "123456789"
W_WRITE (CRCIN, 0x31323334);
W_WRITE (CRCIN, 0x35363738);
B_WRITE (CRCIN, 0x39);

// read result
W_READ (CRCR, data);

// check CRC result
assert (data == 0x2639F4CB);
 // <- big endian & LSB First
```



- CRC32(IEEE-802.3)のときは、ビットオーダは LSB First となります。本 CRC 演算器では、バイトオーダはどちらでも対応可能であり、上図はビッグエンディアンの場合を示しています。

38.5.6.4 使用例 4 CRC32, バイトオーダ, リトルエンディアン

使用例 4 CRC32, バイトオーダ, リトルエンディアンについて示します。

図 38-6 使用例 4

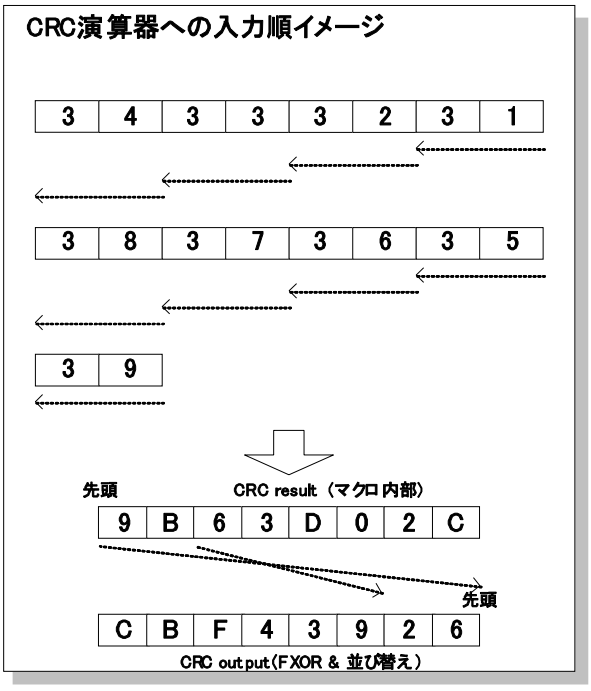
```
//*************************************
// CRC32 (IEEE-802.3)
// polynomial: 0x04C11DB7
// initial value: 0xFFFF_FFFF
// CRCCR.CRC32 1 // CRC32
// CRCCR.LTLEND: 1 // little endian
// CRCCR.LSBFST: 1 // LSB First
// CRCCR.CRCLTE: 1 // CRC little endian
// CRCCR.CRCLSF: 1 // CRC LSB First
// CRCCR.FXOR: 1 // CRC Final XOR on
//*************************************

//
// 例 4－1 (CRC32)
//

// 初期化
B_WRITE (CRCCR, 0x7F);

// data write "123456789"
W_WRITE (CRCIN, 0x34333231);
W_WRITE (CRCIN, 0x38373635);
B_WRITE (CRCIN, 0x39);

// read result
W_READ (CRCR, data);

// check result
assert (data == 0xCBF43926);
 // <- little endian & LSB First
```

（以下を想定）

B_WRITE — バイト書込み
H_WRITE — ハーフワード書込み
W_WRITE — ワード書込み

B_READ — バイト読込み
H_READ — ハーフワード読込み
W_READ — ワード読込み

CRCCR — 制御レジスタアドレス
CRCINIT — 初期値レジスタアドレス
CRCIN — 入力データレジスタアドレス
CRCR — カレントCRCレジスタアドレス



- CRC32(IEEE-802.3)のときは、ビットオーダは LSB First となります。本 CRC 演算器では、バイトオーダはどちらでも対応可能であり、上図はリトルエンディアンの場合を示しています。
- CRC結果のビット反転が不要な場合は、初期化を 0x3F で行って演算を行うか、データ入力後に、CRCCR:FXOR ビットを"0" (例えば CRCCR=0x3E)にすることで、現在の結果に対してビット反転を解除できます。

39. RAMECC

RAMECC について説明します。

39.1 概要

RAMECC の概要について説明します。

RAM のソフトエラー耐力を上げるため、書込み/読出し時にバイト単位での 1 ビット誤り訂正 2 ビット誤り検出符合の生成・検査を実施します。

39.2 特長

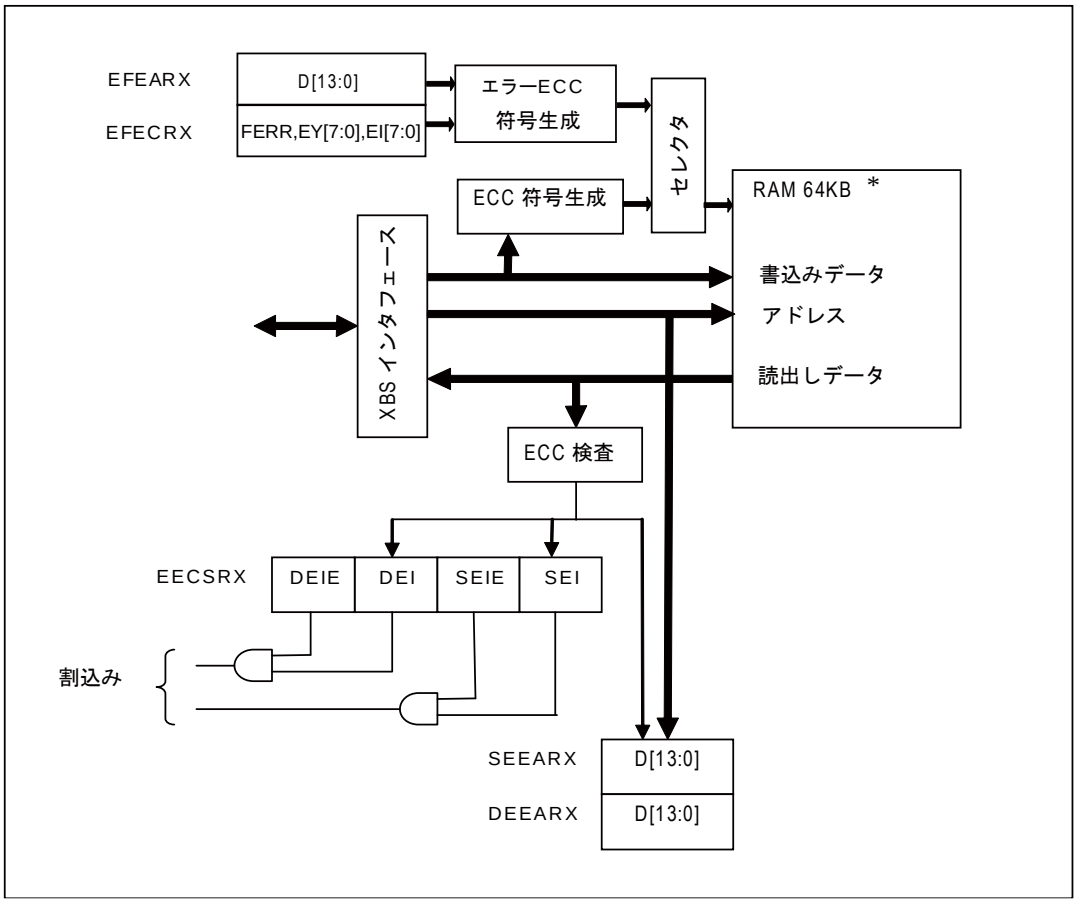
RAMECC の特長について説明します。

- 対象 RAM :
  XBS RAM : 40K バイト(CY91F575)
  64K バイト(CY91F577)
  Backup RAM : 8K バイト
- ECC : 1 バイト単位に 5 ビットの ECC を付加。1 ビット誤り訂正、2 ビット誤り検出が可能。
- 割込み機能 : シングルビットエラーを感知し RAM シングルビットエラー割込み信号、ダブルビットエラーを感知し RAM ダブルビットエラー割込み信号を発生します。
- 試験機能 : ソフトデバッグ用として、擬似エラーを発生します。

39.3 構成

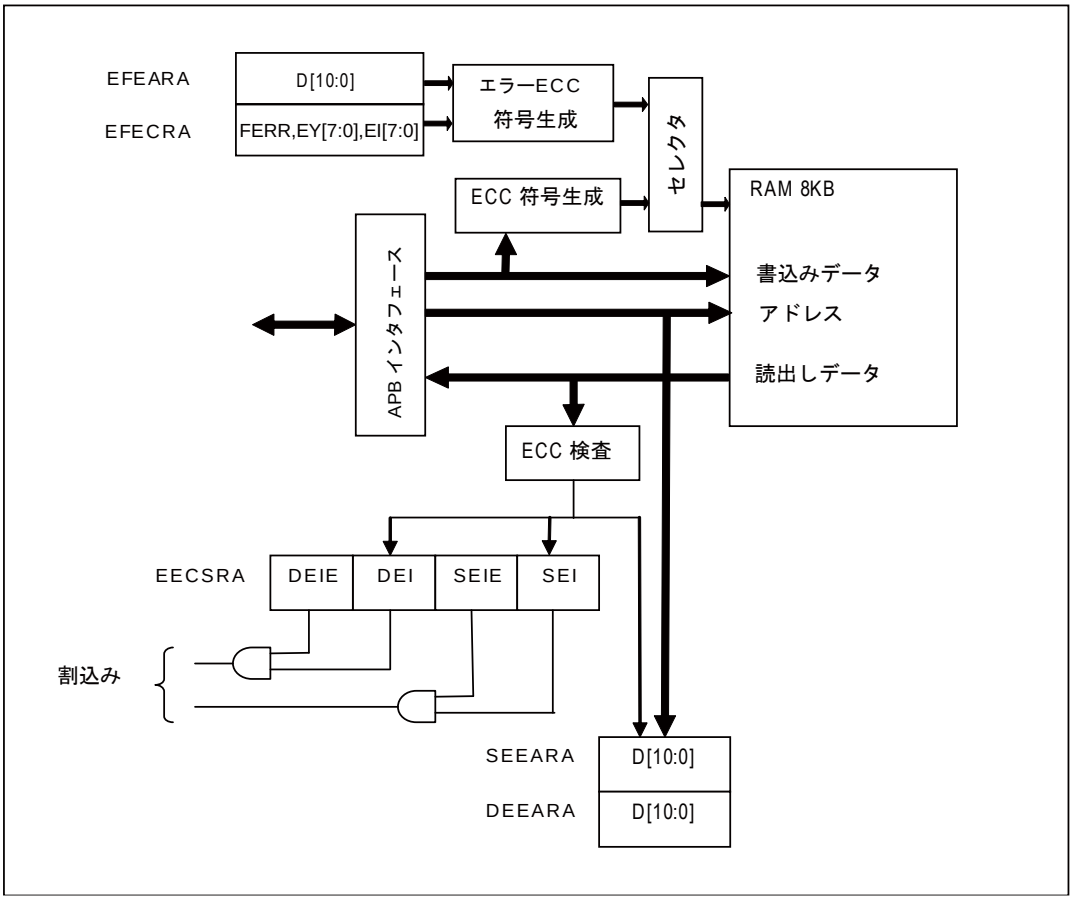
RAMECC の構成について示します。

図 39-1 ブロックダイヤグラム (XBS RAM)



* CY91F575 では 40KB

図 39-2 ブロックダイヤグラム (Backup RAM)



39.4 レジスタ

RAMECC のレジスタについて説明します。

表 39-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x2400	SEEARX		DEEARX		シングルビット ECC エラーアドレスレジスタ XBS RAM ダブルビット ECC エラーアドレスレジスタ XBS RAM
0x2404	EECSRX	予約	EFEARX		ECC エラー制御レジスタ XBS RAM ECC 疑似エラー発生アドレスレジスタ XBS RAM
0x2408	予約	EFECRX			ECC 疑似エラー発生制御レジスタ XBS RAM
0x3000	SEEARA		DEEARA		シングルビット ECC エラーアドレスレジスタ BACKUP-RAM ダブルビット ECC エラーアドレスレジスタ BACKUP-RAM
0x3004	EECSRA	予約	EFEARA		ECC エラー制御レジスタ BACKUP-RAM ECC 疑似エラー発生アドレスレジスタ BACKUP-RAM
0x3008	予約	EFECRA			ECC 疑似エラー発生制御レジスタ BACKUP-RAM

39.4.1 ECC エラー制御レジスタ XBS RAM : EECSRX (Ecc Error Control and Status Register Xbs ram)

ECC エラー制御レジスタ XBS RAM のビット構成について説明します。

XBS RAM の ECC 検査時に、1 ビット誤り訂正あるいは 2 ビット誤り検出が実施されたかどうかという状態を保持し、またそれら事象による割込みを許可するかどうかの設定を行います。

■ EECSRX : アドレス 2404H (アクセス: バイト,ハーフワード,ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				DEIE	DEI	SEIE	SEI
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R(RM1),W	R/W	R(RM1),W

[bit7～bit4] 予約
必ず "0"を書き込んでください。読出し時は"0"が読み出されます。

[bit3] DEIE : ダブルビットエラー要因の割込み許可ビット

DEIE	設定内容
0	割込み禁止
1	割込み許可

[bit2] DEI : ダブルビットエラー発生ビット

DEI	読出し	書込み
0	ダブルビットエラーは発生していない	本ビットをクリア
1	ダブルビットエラーが発生した	効果ありません

[bit1] SEIE : シングルビットエラー要因の割込み許可ビット

SEIE	設定内容
0	割込み禁止
1	割込み許可

[bit0] SEI : シングルビットエラー発生ビット

SEI	読出し	書込み
0	シングルビットエラーは発生していない	本ビットをクリア
1	シングルビットエラーが発生した	効果ありません

39.4.2 シングルビット ECC エラーアドレスレジスタ XBS RAM : SEEARX (Single bit Ecc Error Address Register Xbs ram)

シングルビット ECC エラーアドレスレジスタ XBS RAM のビット構成について説明します。

XBS RAM の ECC 検査時に、1 ビット誤り訂正が実施されたとき、その発生アドレスを保持します。

■ SEEARX : アドレス 2400H (アクセス: バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	D14	D13	D12	D11	D10	D9	D8
初期値	0	0	0	0	0	0	0	0
属性*1	R0,W0	R0,W0	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
属性*2	R0,W0	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit15, bit14] 予約*1

[bit15] 予約*2
読出し時は"0"が読み出されます。

[bit13～bit0] D13～D0*1 : シングルビットエラー発生アドレスビット

[bit14～bit0] D14～D0*2 : シングルビットエラー発生アドレスビット
ECC 検査時に 1 ビット誤り訂正が実施されたとき、その発生アドレスを保持します。

本ビットにセットされた状態でさらに上記事象が検出された場合、本ビットが上書きされることはなく、最初の値が保持されます。

*1: CY91F575/7
*2: CY91F578/9

＜注意事項＞
上記アドレスはワード単位のオフセットになります。絶対アドレスは上記オフセット・アドレスに下位 2 ビット加えて XBS RAM のベース・アドレスを加えて算出してください。
(絶対アドレス)=(0x00010000)+(SEEARX で示されるオフセット+2b'00)

39.4.3 ダブルビット ECC エラーアドレスレジスタ XBS RAM : DEEARX (Double bit Ecc Error Address Register Xbs ram)

ダブルビット ECC エラーアドレスレジスタ XBS RAM のビット構成について説明します。

XBS RAM の ECC 検査時に、2 ビット誤り検出が実施されたとき、その発生アドレスを保持します。

■ DEEARX : アドレス 2402H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	D14	D13	D12	D11	D10	D9	D8
初期値	0	0	0	0	0	0	0	0
属性*1	R0,W0	R0,W0	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
属性*2	R0,W0	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit15, bit14] 予約*1

[bit15] 予約*2
読出し時は"0"が読み出されます。

[bit13～bit0] D13～D0*1 : ダブルビットエラー発生アドレスビット

[bit14～bit0] D14～D0*2 : ダブルビットエラー発生アドレスビット
ECC 検査時に 2 ビット誤り検出が実施されたとき、その発生アドレスを保持します。

本ビットにセットされた状態でさらに上記事象が検出された場合、本ビットが上書きされることはなく、最初の値が保持されます。

*1: CY91F575/7
*2: CY91F578/9

＜注意事項＞
上記アドレスはワード単位のオフセットになります。絶対アドレスは上記オフセット・アドレスに下位 2 ビット加えて XBS RAM のベース・アドレスを加えて算出してください。
(絶対アドレス)=(0x00010000)+(DEEARX で示されるオフセット+2b'00)

39.4.4 ECC 疑似エラー発生アドレスレジスタ XBS RAM : EFEARX (Ecc False Error Address Register Xbs ram)

ECC 疑似エラー発生アドレスレジスタ(EFEARX) のビット構成について説明します。

ECC 疑似エラー発生アドレスレジスタ(EFEARX) は、疑似エラーを発生させるアドレスを指定します。

■ EFEARX : アドレス 2406H (アクセス: バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	D14	D13	D12	D11	D10	D9	D8
初期値	0	0	0	0	0	0	0	0
属性*1	R0,W0	R0,W0	R/W	R/W	R/W	R/W	R/W	R/W
属性*2	R0,W0	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15, bit14] 予約*1

[bit15] 予約*2

必ず "0"を書き込んでください。読出し時は"0"が読み出されます。

[bit13～bit0] D13～D0*1 : 疑似エラー発生アドレス設定ビット

[bit14～bit0] D14～D0*2 : 疑似エラー発生アドレス設定ビット

疑似 ECC エラーを発生させるアドレスを設定します。

EFECRX:FERR="1"のとき、本アドレスへのライトアクセスを発生させ、EFECRX の設定にしたがって、書き込むデータに故意にエラーを含ませることで ECC エラーを起こします。

*1: CY91F575/7
*2: CY91F578/9

39.4.5 ECC 疑似エラー発生制御レジスタ XBS RAM : EFECRX (Ecc False Error Control Register Xbs ram)

ECC 疑似エラー発生制御レジスタ(EFECRX) のビット構成について説明します。

ECC 疑似エラー発生制御レジスタ(EFECRX) は、発生させる疑似エラーの内容を、発生バイト、発生ビットという形で指定します。

■ EFECRX : アドレス 2409H (アクセス: バイト,ハーフワード,ワード)

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	予約							FERR
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R/W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	EY7	EY6	EY5	EY4	EY3	EY2	EY1	EY0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EI7	EI6	EI5	EI4	EI3	EI2	EI1	EI0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit23～bit17] 予約
必ず "0"を書き込んでください。読出し時は"0"が読み出されます。

[bit16] FERR : 疑似エラー発生許可ビット

FERR	設定内容
0	疑似エラー発生禁止
1	疑似エラー許可

[bit15～bit8] EY7～EY0 : **疑似エラー発生バイト設定ビット**
疑似 ECC エラーを発生させる対象のバイト位置を指定します。

EYn	RAM 上の対象バイト
EY0	RAM データ[7:0]
EY1	RAM データ[15:8]
EY2	RAM データ[23:16]
EY3	RAM データ[31:24]
EY4	RAM データ[36:32]
EY5	RAM データ[41:37]
EY6	RAM データ[46:42]
EY7	RAM データ[51:47]

例えば、EY2="1"でほかは"0"の場合、擬似エラーを発生させる対象バイトは、RAM データ[23:16] となり、ほかのデータではエラーは発生しません。

また、EY2=EY3="1"でほかは"0"の場合、擬似エラーを発生させる対象バイトは、RAM データ[31:16]となります。

[bit7～bit0] EI7～EI0 : **疑似エラー発生ビット設定ビット**
疑似 ECC エラーを発生させる対象のビット位置を指定します。

EIn	バイト上の対象ビット
EI0	bit0
EI1	bit1
EI2	bit2
EI3	bit3
EI4	bit4
EI5	bit5
EI6	bit6
EI7	bit7

例えば、EY2="1",EI4="1"で、ほかは"0"の場合、擬似エラーを発生させる対象ビットは、RAM データ[20]となり、シングルビットエラーを訂正することになります。

また、EY2="1",EI4=EI7="1"で、ほかは"0"の場合、擬似エラーを発生させる対象ビットは、RAM データ[23]、RAM データ[20]となり、ダブルビットエラーを検出することになります。

また、EY2=EY3="1",EI4="1"でほかは"0"の場合、擬似エラーを発生させる対象ビットは、RAM データ[28]、RAM データ[20]となり、それぞれのバイトでシングルビットエラーを訂正することになります。

39.4.6 ECC エラー制御レジスタ BACKUP-RAM : EECSRA (Ecc Error Control and Status Register bAckup-ram)

ECC エラー制御レジスタ BACKUP-RAM のビット構成について説明します。

Backup RAM の ECC 検査時に、1 ビット誤り訂正あるいは 2 ビット誤り検出が実施されたかどうかという状態を保持し、またそれら事象による割込みを許可するかどうかの設定を行います。

■ EECSRA : アドレス 3004H (アクセス: バイト,ハーフワード,ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約				DEIE	DEI	SEIE	SEI
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R(RM1),W	R/W	R(RM1),W

[bit7～bit4] 予約
必ず "0"を書き込んでください。読出し時は"0"が読み出されます。

[bit3] DEIE : ダブルビットエラー要因の割込み許可ビット

DEIE	設定内容
0	割込み禁止
1	割込み許可

[bit2] DEI : ダブルビットエラー発生ビット

DEI	読出し	書込み
0	ダブルビットエラーは発生していない	本ビットをクリア
1	ダブルビットエラーが発生した	効果ありません

[bit1] SEIE : シングルビットエラー要因の割込み許可ビット

SEIE	設定内容
0	割込み禁止
1	割込み許可

[bit0] SEI : シングルビットエラー発生ビット

SEI	読出し	書込み
0	シングルビットエラーは発生していない	本ビットをクリア
1	シングルビットエラーが発生した	効果ありません

39.4.7 シングルビット ECC エラーアドレスレジスタ BACKUP-RAM : SEEARA (Single bit Ecc Error Address Register bAckup-ram)

シングルビット ECC エラーアドレスレジスタ BACKUP-RAM のビット構成について説明します。

Backup RAM の ECC 検査時に、1 ビット誤り訂正が実施されたとき、その発生アドレスを保持します。

■ SEEARA : アドレス 3000H (アクセス: バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約				D11	D10	D9	D8
初期値	0	0	0	0	0	0	0	0
属性*1	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R,WX	R,WX	R,WX
属性*2	R0,W0	R0,W0	R0,W0	R0,W0	R,WX	R,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit15～bit11] 予約*1

[bit15～bit12] 予約*2
読出し時は"0"が読み出されます。

[bit10～bit0] D10～D0*1 : シングルビットエラー発生アドレスビット

[bit11～bit0] D11～D0*2 : シングルビットエラー発生アドレスビット
ECC 検査時に 1 ビット誤り訂正あるいは実施されたとき、その発生アドレスを保持します。

本ビットにセットされた状態でさらに上記事象が検出された場合、本ビットが上書きされることはなく、最初の値が保持されます。

*1: CY91F575/7
*2: CY91F578/9

＜注意事項＞
上記アドレスはワード単位のオフセットになります。絶対アドレスは上記オフセット・アドレスに下位 2 ビット加えて Backup RAM のベース・アドレスを加えて算出してください。
(絶対アドレス)=(0x00004000)+(SEEARA で示されるオフセット+2b'00)

39.4.8 ダブルビットECCエラーアドレスレジスタ BACKUP-RAM : DEEARA (Double bit Ecc Error Address Register bAckup-ram)

ダブルビット ECC エラーアドレスレジスタ BACKUP-RAM のビット構成について説明します。

Backup RAM の ECC 検査時に、2 ビット誤り検出が実施されたとき、その発生アドレスを保持します。

■ DEEARA : アドレス 3002H (アクセス: バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約				D11	D10	D9	D8
初期値	0	0	0	0	0	0	0	0
属性*1	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R,WX	R,WX	R,WX
属性*2	R0,W0	R0,W0	R0,W0	R0,W0	R,WX	R,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit15～bit11] 予約*1

[bit15～bit12] 予約*2
読出し時は"0"が読み出されます。

[bit10～bit0] D10～D0*1 : ダブルビットエラー発生アドレスビット

[bit11～bit0] D11～D0*2 : ダブルビットエラー発生アドレスビット
ECC 検査時に 2 ビット誤り検出が実施されたとき、その発生アドレスを保持します。

本ビットにセットされた状態でさらに上記事象が検出された場合、本ビットが上書きされることはなく、最初の値が保持されます。

*1: CY91F575/7
*2: CY91F578/9

＜注意事項＞
上記アドレスはワード単位のオフセットになります。絶対アドレスは上記オフセット・アドレスに下位 2 ビット加えて Backup RAM のベース・アドレスを加えて算出してください。
(絶対アドレス)=(0x00004000)+(DEEARA で示されるオフセット+2b'00)

39.4.9 ECC 疑似エラー発生アドレスレジスタ BACKUP-RAM : EFEARA (Ecc False Error Address Register bAckup-ram)

ECC 疑似エラー発生アドレスレジスタ BACKUP-RAM のビット構成について説明します。

ECC 疑似エラー発生アドレスレジスタ(EFEARA) は、疑似エラーを発生させるアドレスを指定します。

■ EFEARA : アドレス 3006H (アクセス: バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約				D11	D10	D9	D8
初期値	0	0	0	0	0	0	0	0
属性*1	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W
属性*2	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit11] 予約*1

[bit15～bit12] 予約*2
必ず "0"を書き込んでください。読出し時は"0"が読み出されます。

[bit10～bit0] D10～D0*1 : 疑似エラー発生アドレス設定ビット

[bit11～bit0] D11～D0*2 : 疑似エラー発生アドレス設定ビット
疑似 ECC エラーを発生させるアドレスを設定します。

EFECRA:FERR="1"のとき、本アドレスへのライトアクセスを発生させ、EFECRA の設定にしたがって、書き込むデータに故意にエラーを含ませることで ECC エラーを起こします。

*1: CY91F575/7
*2: CY91F578/9

39.4.10 ECC疑似エラー発生制御レジスタ BACKUP-RAM : EFECRA (Ecc False Error Control Register bAckup-ram)

ECC 疑似エラー発生制御レジスタ BACKUP-RAM のビット構成について説明します。

ECC 疑似エラー発生制御レジスタ(EFECRA) は、発生させる疑似エラーの内容を、発生バイト、発生ビットという形で指定します。

■ EFECRA : アドレス 3009H (アクセス: バイト,ハーフワード,ワード)

	bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
	予約							FERR
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R/W

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	EY7	EY6	EY5	EY4	EY3	EY2	EY1	EY0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EI7	EI6	EI5	EI4	EI3	EI2	EI1	EI0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit23～bit17] 予約
必ず "0"を書き込んでください。読出し時は"0"が読み出されます。

[bit16] FERR : 疑似エラー発生許可ビット

FERR	設定内容
0	疑似エラー発生禁止
1	疑似エラー許可

[bit15～bit8] EY7～EY0 : 疑似エラー発生バイト設定ビット
疑似 ECC エラーを発生させる対象のバイト位置を指定します。

EYn	RAM 上の対象バイト
EY0	RAM データ[7:0]
EY1	RAM データ[15:8]
EY2	RAM データ[23:16]
EY3	RAM データ[31:24]
EY4	RAM データ[36:32]
EY5	RAM データ[41:37]
EY6	RAM データ[46:42]
EY7	RAM データ[51:47]

例えば、EY2="1"でほかは"0"の場合、擬似エラーを発生させる対象バイトは、RAM データ[23:16] となり、ほかのデータではエラーは発生しません。

また、EY2=EY3="1"でほかは"0"の場合、擬似エラーを発生させる対象バイトは、RAM データ[31:16]となります。

[bit7～bit0] EI7～EI0 : 疑似エラー発生ビット設定ビット
疑似 ECC エラーを発生させる対象のビット位置を指定します。

EIn	バイト上の対象ビット
EI0	bit0
EI1	bit1
EI2	bit2
EI3	bit3
EI4	bit4
EI5	bit5
EI6	bit6
EI7	bit7

例えば、EY2="1",EI4="1"で、ほかは"0"の場合、擬似エラーを発生させる対象ビットは、RAM データ[20]となり、シングルビットエラーを訂正することになります。

また、EY2="1",EI4=EI7="1"で、ほかは"0"の場合、擬似エラーを発生させる対象ビットは、RAM データ[23]、RAM データ[20]となり、ダブルビットエラーを検出することになります。

また、EY2=EY3="1",EI4="1"でほかは"0"の場合、擬似エラーを発生させる対象ビットは、RAM データ[28]、RAM データ[20]となり、それぞれのバイトでシングルビットエラーを訂正することになります。

39.5 動作説明

RAMECC の動作について説明します。

38.5.1. ECC 生成

39.5.2. ECC 検査

39.5.3. エラー検出による割込み

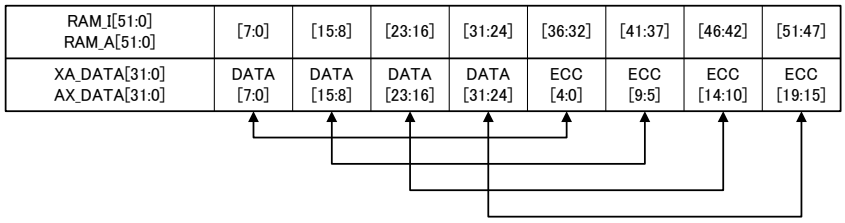
39.5.4. 試験機能

39.5.1 ECC 生成

ECC の生成について説明します。

XBS インタフェース上 32 ビットのデータに対して、8 ビット(1 バイト)単位での ECC 生成を行います。

図 39-3 XBS データと RAM データの関係



上図のように、XBS データ(XA_DATA[31:0] / AX_DATA[31:0])と RAM データ(RAM_I[51:0] / RAM_A[51:0])の関係を定義します。図中の矢印で接続されているものは DATA と ECC の対になっています。(例 ：{XA_DATA[15:8] / AX_DATA[15:8]}の 8 ビットは、RAM データとしては、{RAM_I[15:8] / RAM_A[15:8]} として格納され、それに対応する ECC[9:5]は、RAM データとしては、{RAM_I[41:37] / RAM_A[41:37]}として格納されています。)

また、ECC 符号行列としては、下表に示す(13,8)奇数重み符号を採用しています。

表 39-2 ECC 符号行列

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	0	0	1	1	1	0	1	1	0	0	0	0
2	1	1	0	0	1	0	1	0	0	1	0	0	0
3	1	1	1	0	0	1	0	1	0	0	1	0	0
4	0	1	1	1	0	1	1	0	0	0	0	1	0
5	0	0	1	1	1	0	1	1	0	0	0	0	1

これにより、8 ビットのデータに対して、5 または 6 個の排他的論理和を求めることで、ECC の各ビットが算出可能となっています。

なお、これは検査行列そのものとなっており、生成行列は 8 列までの 5×8 行列の転置行列と単位行列との組み合わせによって求められます。

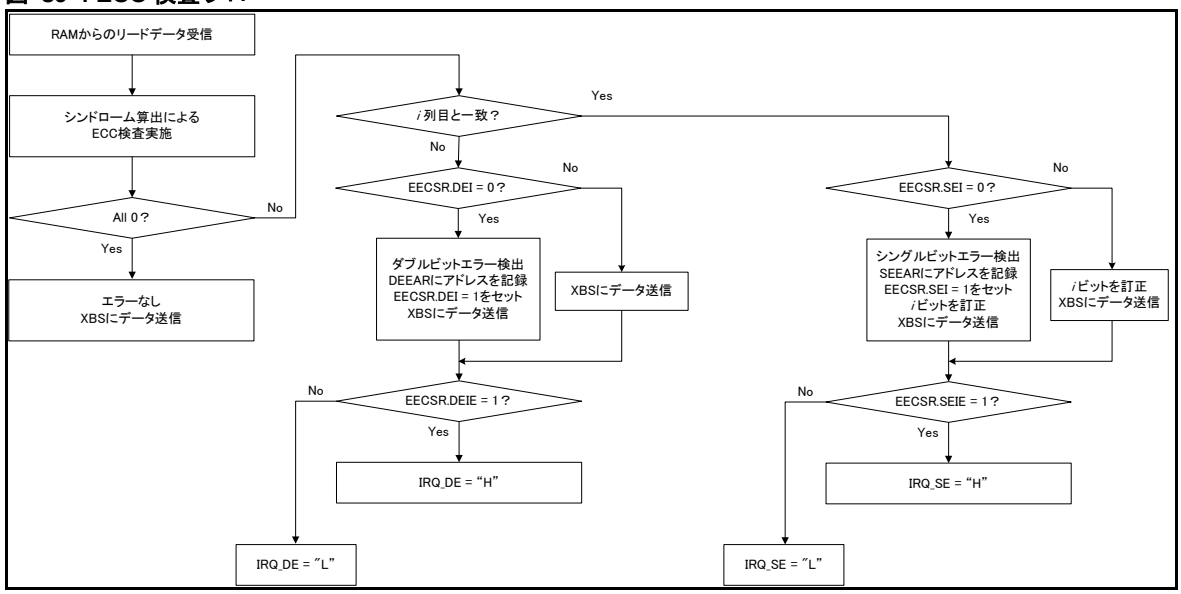
XBS データを受信してから、1 クロックサイクルのうちに ECC を生成し、RAM への書込みを完了させてください。

39.5.2 ECC 検査

ECC の検査について説明します。

図 39-3 XBS データと RAM データの関係のように RAM 上にあるデータから、ECC を含む 13 ビットの符号ベクトルを構築し、これに対して表 39-2 ECC 符号行列の検査行列を用いて下図の手順で検査を実施します。

図 39-4 ECC 検査フロー



39.5.3 エラー検出による割込み

エラー検出時の割込みについて説明します。

ECC エラー検出時に割込みを発生することができます。割込みを発生させるには、用途に応じて DEIE, SEIE ビットに"1"を書き込み、RAMECC 割込みベクタ、割込みレベルの設定を行ってください。

割込み要因	割込みベクタ	割込みレベル
SEI (RAM シングルビットエラー割込み)	#61($000FFF08_H$)	ICR45
DEI (RAM ダブルビットエラー割込み)	#15($000FFFC0_H$)	15(F_H)固定

割込みレベル, 割込みベクタの詳細については、『割込み制御(割込みコントローラ)』の章を参照してください。

割込み要求フラグ(DEI, SEI) は自動的にはクリアしません。このため、割込み処理から復帰する前にソフトウェアにてクリアしてください(DEI, SEI ビットに"0"を書き込む)。

ダブルビットエラーを検出した場合には、訂正が行われず、エラーデータがそのまま読み出されますので、NMI レベルの割込みを発生します。

39.5.4 試験機能

試験機能について説明します。

ソフトデバッグ用として、擬似エラーを発生します。

ECC 擬似エラー発生動作は以下の手順で発生します。

1. ECC疑似エラー発生アドレスレジスタ(EFEARX/EFEARA)に疑似エラーを発生させるアドレスを指定します。
2. ECC 疑似エラー発生制御レジスタ(EFECRX/EFECRA) で、バイト、ビットを設定します。
   (a) EFECRX:EY[7:0]/EFECRA:EY[7:0]に擬似エラーを発生させるバイト位置を指定します。
   (b) EFECRX:EI[7:0]/EFECRA:EI[7:0]に擬似エラーを発生させるビット位置を指定します。
3. ECC 疑似エラー発生制御レジスタ(EFECRX/EFECRA)の FERR ビットに"1"を書き込みます。

EFEARX/EFEARA で指定されたアドレスに、EY[7:0],EI[7:0]の内容にしたがって、故意にエラーを含むデータの書込みを開始します。書込みに引き続き読出しを行い、擬似エラーを検出します。

40. マルチファンクションシリアルインタフェース

マルチファンクションシリアルインタフェースについて説明します。

40.1 概要

マルチファンクションシリアルインタフェースの概要について説明します。

本モジュールは、UART (非同期シリアルインタフェース), CSIO (SPI 対応, クロック同期シリアルインタフェース), LIN-UART (LIN 処理用ハードウェア付きシリアルインタフェース)と f のシリアル通信機能を提供します。

40.2 特長

マルチファンクションシリアルインタフェースの特長について説明します。

本シリーズには4チャネルのマルチファンクションシリアルインタフェース通信モジュールが搭載されており、それぞれシリアルモードレジスタ(SMR)で、UART/CSIO/LIN-UART/I²C のいずれかを選択して使用します。

＜注意事項＞
I²C 機能は、ch.0, ch.1 のみ対応しています。

40.2.1 UART

マルチファンクションシリアルインタフェースの UART について説明します。

UART (非同期シリアルインタフェース)は、外部装置と非同期通信(調歩同期)をするための、汎用のシリアルデータ通信インタフェースです。双方向通信機能(ノーマルモード), マスタ/スレーブ型通信機能(マルチプロセッサモード：マスタ/スレーブ両方サポート)をサポートしています。また、送信/受信用の FIFO を搭載しています。

名称	機　　能
データ	・全二重ダブルバッファ(FIFO 未使用時) ・送信/受信 FIFO (各 16 バイト) (FIFO 使用時)
シリアル入力	3 回オーバサンプリングを行い、サンプリング値の多数決により受信値を決定します。
転送形式	非同期
ボーレート	・専用ボーレートジェネレータ(15 ビットリロードカウンタ構成) ・外部クロック入力をリロードカウンタで調節可能
データ長	・5～9 ビット(ノーマルモード時), 7, 8 ビット(マルチプロセッサモード時)
信号方式	NRZ (Non Return to Zero), 反転 NRZ
スタートビット検出	・スタートビット立下りエッジに同期(NRZ 方式の場合) ・スタートビット立上りエッジに同期(反転 NRZ 方式の場合)
受信エラー検出	・フレーミングエラー ・オーバランエラー ・パリティエラー*
割込み要求	・受信割込み (受信完了, フレーミングエラー, オーバランエラー, パリティエラー*) ・送信割込み(送信データエンプティ, 送信バスアイドル) ・送信 FIFO 割込み (送信 FIFO がエンプティのとき) ・送受信とも DMA 機能あり
マスタ/スレーブ型通信機能(マルチプロセッサモード)	1 (マスタ)対 n (スレーブ)間の通信が可能 (マスタとスレーブシステムの両方をサポート)
FIFO オプション	・送受信 FIFO 搭載(送信 FIFO 16 バイト, 受信 FIFO 16 バイト) ・送信 FIFO と受信 FIFO を選択可能 ・送信データ再送可能 ・受信 FIFO 割込みタイミングをソフトウェアで可変可能 ・独立して FIFO リセットサポート
DMA 転送対応	送信 ： 対応 受信 ： 対応

*：パリティエラーはノーマルモード時のみ。

40.2.2 CSIO

マルチファンクションシリアルインタフェースの CSIO について説明します。

CSIO (クロック同期シリアルインタフェース)は、外部装置と同期通信をするための、汎用のシリアルデータ通信インタフェースです。(SPI に対応します) また、送信/受信 (各 16 バイト) の FIFO を搭載しています。

名称	機　　能
データバッファ	・全二重ダブルバッファ(FIFO 未使用時) ・送信/受信 FIFO (各 16 バイト)(FIFO 使用時)
転送形式	・クロック同期(スタートビット/ストップビットなし) ・マスタ/スレーブ機能 ・SPI に対応 (マスタ/スレーブ両方サポート)
ボーレート	・専用ボーレートジェネレータあり(15 ビットリロードカウンタから構成, マスタ動作時) ・外部クロック入力可能(スレーブ動作時)
データ長	・5～9 ビットに可変可能
受信エラー検出	・オーバランエラー
割込み要求	・受信割込み(受信完了, オーバランエラー) ・送信割込み(送信データエンプティ, 送信バスアイドル) ・送信 FIFO 割込み (送信 FIFO がエンプティのとき) ・送受信とも DMA 転送サポート機能あり
同期モード	マスタまたはスレーブ機能
端子アクセス	シリアルデータ出力端子を"1"設定可能
FIFO オプション	・送受信 FIFO 搭載(送信 FIFO 16 バイト, 受信 FIFO 16 バイト) ・送信 FIFO と受信 FIFO を選択可能 ・送信データ再送可能 ・受信 FIFO 割込みタイミングをソフトウェアで可変可能 ・独立して FIFO リセットサポート
DMA 転送対応	送信 ： 対応 受信 ： 対応

40.2.3 LIN-UART

マルチファンクションシリアルインタフェースの LIN-UART について説明します。

LIN-UART (LIN 通信制御 UART)は、LIN バスに対応するための特別な機能をサポートしています。また、送信/受信(各 16 バイト)の FIFO を搭載しています。

名称	機　　能
データバッファ	・全二重ダブルバッファ(FIFO 未使用時) ・送信/受信 FIFO (各 16 バイト)(FIFO 使用時)
シリアル入力	周辺クロック(PCLK)で 3 回オーバサンプリングを行い、サンプリング値の多数決により受信値を決定します。
転送モード	・非同期
ボーレート	・専用ボーレートジェネレータあり(15 ビットリロードカウンタから構成) ・外部クロックをリロードカウンタで調節可能
データ長	8 ビット
信号方式	NRZ (Non Return to Zero)
スタートビット検出	スタートビット立下りエッジに同期
受信エラー検出	・フレーミングエラー ・オーバランエラー
割込み要求	・受信割込み (受信完了，フレーミングエラー，オーバランエラー) ・送信割込み(送信データエンプティ，送信バスアイドル) ・ステータス割込み(LIN synch break 検出) ・ICU への割込み要求 (LIN synch field 検出 : LSYN) ・送信 FIFO 割込み (送信 FIFO がエンプティのとき) ・送受信とも DMA 機能あり
LIN バスオプション	・LIN プロトコル Revision 2.1 に対応 ・マスタデバイス動作 ・スレーブデバイス動作 ・LIN Synch break 生成(13～16 ビット長に可変可能) ・Synch Delimiter 生成(1～4 ビット長に可変可能) ・LIN Synch break 検出 ・インプットキャプチャに接続している LIN synch field のスタート/ストップエッジをインプットキャプチャで検出(『インプットキャプチャ』の章を参照してください。)
FIFO オプション	・送受信 FIFO 搭載(送信 FIFO 16 バイト，受信 FIFO 16 バイト) ・送信 FIFO と受信 FIFO を選択可能 ・送信データ再送可能 ・受信 FIFO 割込みタイミングをソフトウェアで可変可能 ・独立して FIFO リセットサポート
DMA 転送対応	送信 ： 対応 受信 ： 対応 ステータス ： 対応していません

40.2.4 I²C

マルチファンクションシリアルインタフェースの I²C について説明します。

I²C インタフェースは IC 間バスをサポートし、I²C バス上のマスタ/スレーブデバイスとして動作します。また、送信/受信(各 16 バイト)の FIFO を搭載しています。

名称	機　　能
データバッファ	・全二重ダブルバッファ(FIFO 未使用時) ・送信/受信 FIFO (各 16 バイト) (FIFO 使用時)
シリアル入力	シリアルクロック・シリアルデータ入力に対し、周辺クロック(PCLK)で 2 クロックまでのノイズを除去します。
転送モード	同期
ボーレート	・専用ボーレートジェネレータあり(15 ビットリロードカウンタから構成) ・外部クロックをリロードカウンタで調節可能
データ長	8 ビット
信号方式	NRZ (Non Return to Zero)
スタートビット検出	スタートビット立下りエッジに同期
割込み要求	・受信割込み ・送信割込み ・ステータス割込み・ICU への割込み要求 ・送信 FIFO 割込み (送信 FIFO がエンプティのとき)
I²C	・マスタ/スレーブ送受信機能 ・調停機能 ・クロック同期機能 ・伝送方向検出機能 ・反復スタート条件の発生と検出機能 ・バスエラー検出機能 ・ゼネラルコールアドレッシング機能 ・マスタおよびスレーブとしての 7 ビットアドレッシング ・伝送, およびバスエラー時に割込み発生可能 ・10 ビットアドレッシング機能は、プログラムで対応可能
FIFO	・送受信 FIFO 搭載(送信 FIFO 16 バイト, 受信 FIFO 16 バイト) ・送信 FIFO と受信 FIFO を選択可能 ・送信データ再送可能 ・受信 FIFO 割込みタイミングをソフトウェアで可変可能 ・独立して FIFO リセットサポート
DMA 転送対応	送信 ： 対応 受信 ： 対応していません ステータス ： 対応していません

40.2.5 注意事項

＜注意事項＞

- シリアル通信においては、ノイズなどにより間違ったデータを受信する可能性があります。
  そのため、ノイズを抑えるボードの設計をしてください。
  また、万が一ノイズなどの影響により、誤ったデータを受信した場合を考慮して最後にデータのチェックサムなどを付加してエラーが発生した場合には再送を行うなどの処理をしてください。
- I²C 機能は、ch.0, ch.1 のみ対応しています。

40.3 構成

マルチファンクションシリアルインタフェースの構成について示します。

図 40-1 ブロックダイヤグラム (UART： 動作モード 0, 1)

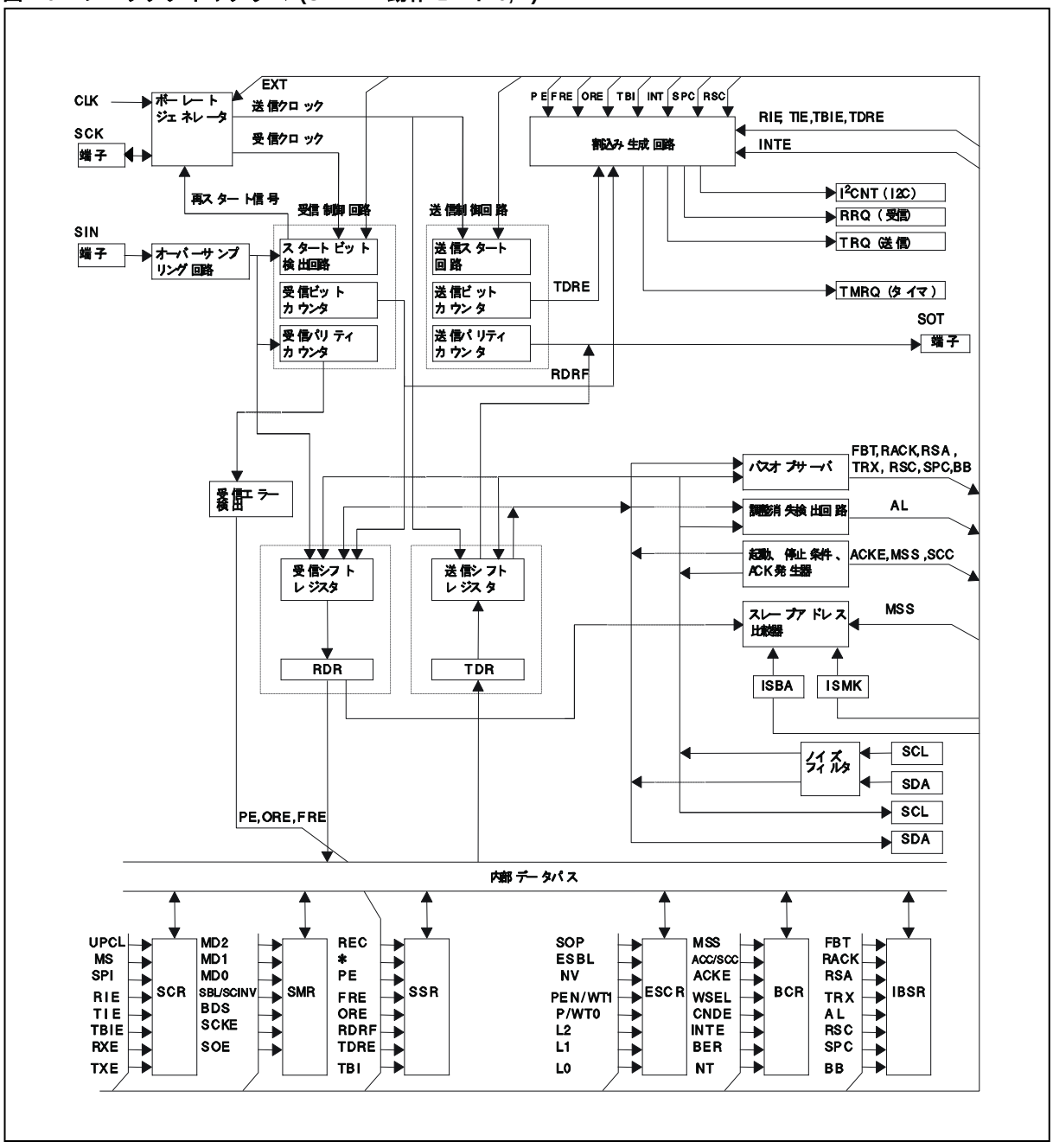


図 40-2 ブロックダイヤグラム (CSIO : 動作モード 2)

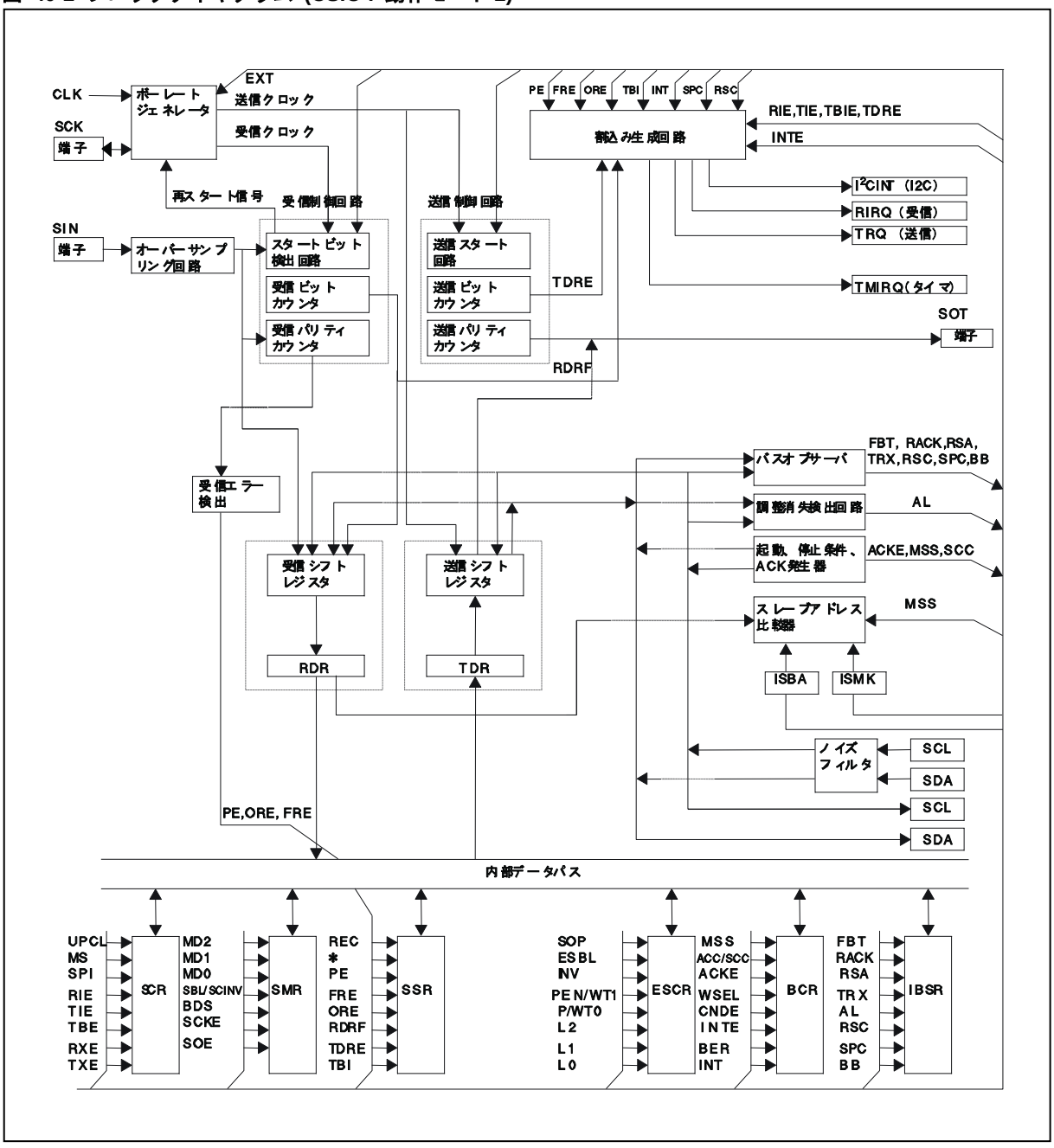


図 40-3 ブロックダイヤグラム (LIN-UART : 動作モード 3)

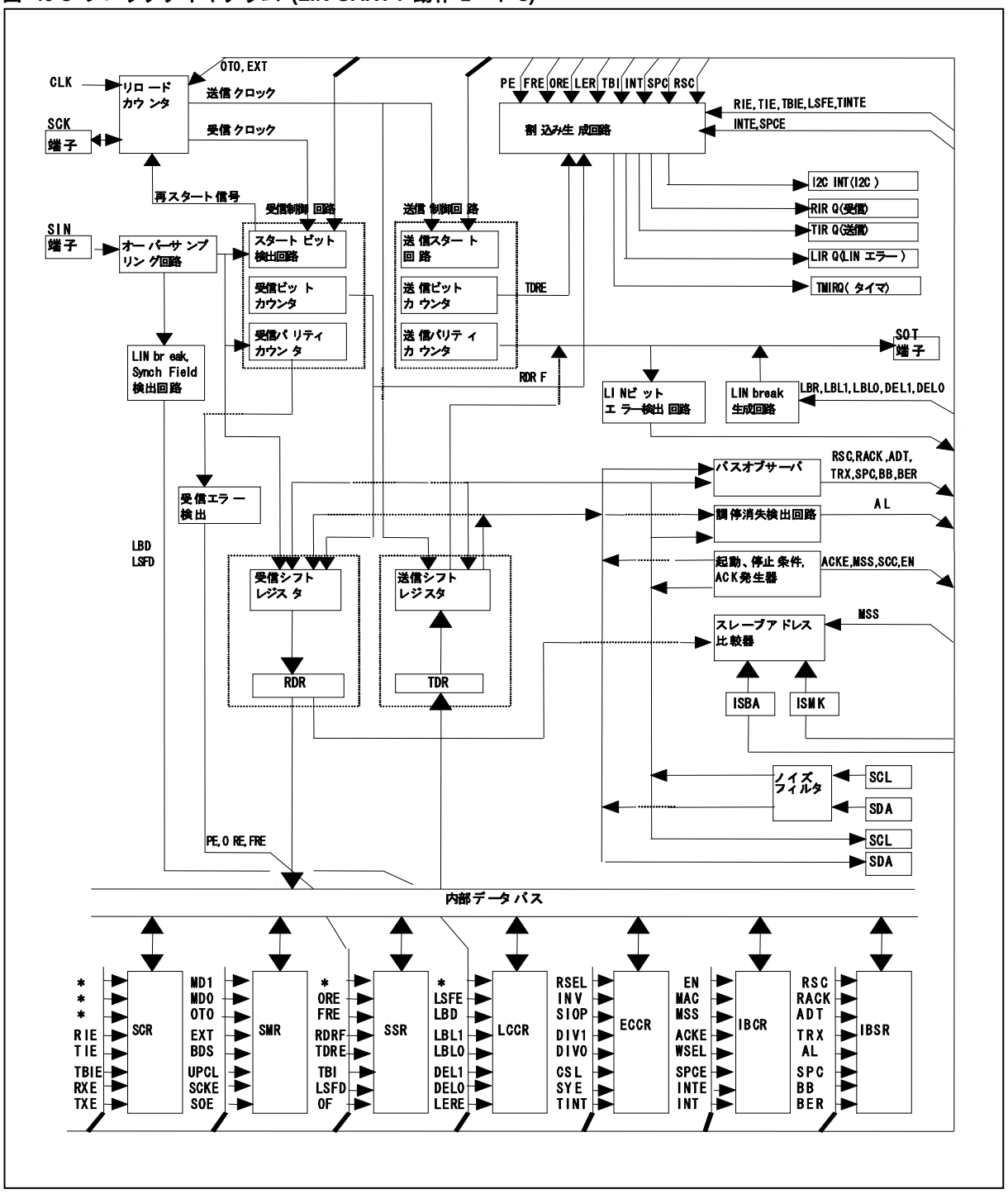
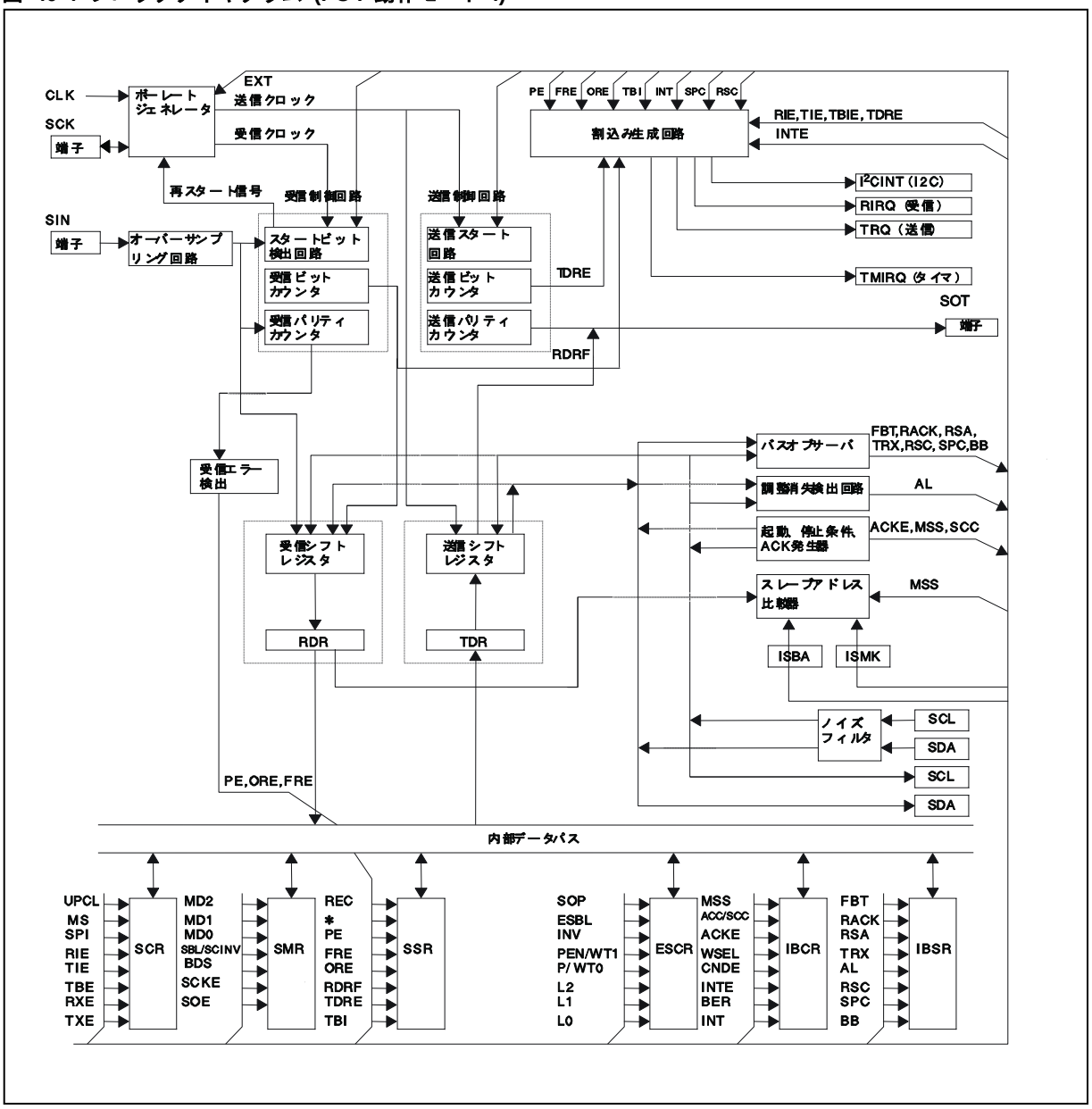


図 40-4 ブロックダイヤグラム (I²C : 動作モード 4)



40.4 レジスタ

マルチファンクションシリアルインタフェースのレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子表

表 40-1 ベースアドレス(Base_addr)・外部端子表

チャネル	Base_addr	外部端子		
		SCK SCL(*)	SOT SDA(*)	SIN
0	0x00B0	SCK0_0/SCK0_1	SOT0_0/SOT0_1	SIN0_0/SIN0_1
1	0x00C0	SCK1_0/SCK1_1	SOT1_0/SOT1_1	SIN1_0/SIN1_1
8	0x04E0	SCK8_0/SCK8_1/ SCK8_2	SOT8_0/SOT8_1/ SOT8_2	SIN8_0/SIN8_1/ SIN8_2
9	0x04F0	SCK9_0/SCK9_1	SOT9_0/SOT9_1	SIN9_0/SIN9_1

*： I²C 設定時の端子名(I²C として使用できるのは、SCK0_0, SCK1_0, SOT0_0, SOT1_0 のみです)

■ レジスタマップ

表 40-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x00B0	【UART】 SCR0 【CSIO】 SCR0 【LIN-UART】SCR0 【I²C】 IBCR0	【共通】 SMR0	【UART】 SSR0 【CSIO】 SSR0 【LIN-UART】 SSR0 【I²C】 SSR0	【UART】 ESCR0 【CSIO】 ESCR0 【LIN-UART】 ESCR0 【I²C】 IBSR0	―――ch.0――― 【UART】シリアル制御レジスタ 【CSIO】シリアル制御レジスタ 【LIN-UART】シリアル制御レジスタ 【I²C】I²C バス制御レジスタ 【共通】シリアルモードレジスタ 【UART】シリアルステータスレジスタ 【CSIO】シリアルステータスレジスタ 【LIN-UART】シリアルステータスレジスタ 【I²C】シリアルステータスレジスタ 【UART】拡張通信制御レジスタ 【CSIO】拡張通信制御レジスタ 【LIN-UART】拡張通信制御レジスタ 【I²C】I²C バスステータスレジスタ
0x00B4	【UART】 RDR0/TDR0 【CSIO】 RDR0/TDR0 【LIN-UART】 RDR0/TDR0 【I²C】 RDR0/TDR0		【UART】 BGR0 【CSIO】 BGR0 【LIN-UART】 BGR0 【I²C】 BGR0		―――ch.0――― 【UART】送受信データレジスタ 【CSIO】送受信データレジスタ 【LIN-UART】送受信データレジスタ 【I²C】送受信データレジスタ 【UART】ボーレートジェネレータレジスタ 【CSIO】ボーレートジェネレータレジスタ 【LIN-UART】ボーレートジェネレータレジスタ 【I²C】ボーレートジェネレータレジスタ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x00B8	【UART】 予約 【CSIO】 予約 【LIN-UART】 予約 【I²C】 ISMK0	【UART】 予約 【CSIO】 予約 【LIN-UART】 予約 【I²C】 ISBA0	予約		―――ch.0――― 【I²C】7 ビットスレーブアドレスマスクレジスタ 【I²C】7 ビットスレーブアドレスレジスタ
0x00BC	【共通】 FCR10	【共通】 FCR00	【共通】 FBYTE0		―――ch.0――― 【共通】FIFO 制御レジスタ 1 【共通】FIFO 制御レジスタ 0 【共通】FIFO バイトレジスタ
0x00C0	【UART】 SCR1 【CSIO】 SCR1 【LIN-UART】SCR1 【I²C】 IBCR1	【共通】 SMR1	【UART】 SSR1 【CSIO】 SSR1 【LIN-UART】 SSR1 【I²C】 SSR1	【UART】 ESCR1 【CSIO】 ESCR1 【LIN-UART】 ESCR1 【I²C】 IBSR1	―――ch.1――― 【UART】シリアル制御レジスタ 【CSIO】シリアル制御レジスタ 【LIN-UART】シリアル制御レジスタ 【I²C】I²C バス制御レジスタ 【共通】シリアルモードレジスタ 【UART】シリアルステータスレジスタ 【CSIO】シリアルステータスレジスタ 【LIN-UART】シリアルステータスレジスタ 【I²C】シリアルステータスレジスタ 【UART】拡張通信制御レジスタ 【CSIO】拡張通信制御レジスタ 【LIN-UART】拡張通信制御レジスタ 【I²C】I²C バスステータスレジスタ
0x00C4	【UART】 RDR1/TDR1 【CSIO】 RDR1/TDR1 【LIN-UART】 RDR1/TDR1 【I²C】 RDR1/TDR1		【UART】 BGR1 【CSIO】 BGR1 【LIN-UART】 BGR1 【I²C】 BGR1		―――ch.1――― 【UART】送受信データレジスタ 【CSIO】送受信データレジスタ 【LIN-UART】送受信データレジスタ 【I²C】送受信データレジスタ 【UART】ボーレートジェネレータレジスタ 【CSIO】ボーレートジェネレータレジスタ 【LIN-UART】ボーレートジェネレータレジスタ 【I²C】ボーレートジェネレータレジスタ
0x00C8	【UART】 予約 【CSIO】 予約 【LIN-UART】 予約 【I²C】 ISMK1	【UART】 予約 【CSIO】 予約 【LIN-UART】 予約 【I²C】 ISBA1	予約		―――ch.1――― 【I²C】7 ビットスレーブアドレスマスクレジスタ 【I²C】7 ビットスレーブアドレスレジスタ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x00CC	【共通】 FCR11	【共通】 FCR01	【共通】 FBYTE1		―――ch.1――― 【共通】FIFO 制御レジスタ 1 【共通】FIFO 制御レジスタ 0 【共通】FIFO バイトレジスタ
0x04E0	【UART】 SCR8 【CSIO】 SCR8 【LIN-UART】SCR8	【共通】 SMR8	【UART】 SSR8 【CSIO】 SSR8 【LIN-UART】 SSR8	【UART】 ESCR8 【CSIO】 ESCR8 【LIN-UART】 ESCR8	―――ch.8――― 【UART】シリアル制御レジスタ 【CSIO】シリアル制御レジスタ 【LIN-UART】シリアル制御レジスタ 【共通】シリアルモードレジスタ 【UART】シリアルステータスレジスタ 【CSIO】シリアルステータスレジスタ 【LIN-UART】シリアルステータスレジスタ 【UART】拡張通信制御レジスタ 【CSIO】拡張通信制御レジスタ 【LIN-UART】拡張通信制御レジスタ
0x04E4	【UART】 RDR8/TDR8 【CSIO】 RDR8/TDR8 【LIN-UART】 RDR8/TDR8 【I²C】 RDR8/TDR8		【UART】 BGR8 【CSIO】 BGR8 【LIN-UART】 BGR8 【I²C】 BGR8		―――ch.8――― 【UART】送受信データレジスタ 【CSIO】送受信データレジスタ 【LIN-UART】送受信データレジスタ 【UART】ボーレートジェネレータレジスタ 【CSIO】ボーレートジェネレータレジスタ 【LIN-UART】ボーレートジェネレータレジスタ
0x04E8	予約				
0x04EC	【共通】 FCR18	【共通】 FCR08	【共通】 FBYTE8		―――ch.8――― 【共通】FIFO 制御レジスタ 1 【共通】FIFO 制御レジスタ 0 【共通】FIFO バイトレジスタ
0x04F0	【UART】 SCR9 【CSIO】 SCR9 【LIN-UART】SCR9	【共通】 SMR9	【UART】 SSR9 【CSIO】 SSR9 【LIN-UART】 SSR9	【UART】 ESCR9 【CSIO】 ESCR9 【LIN-UART】 ESCR9	―――ch.9――― 【UART】シリアル制御レジスタ 【CSIO】シリアル制御レジスタ 【LIN-UART】シリアル制御レジスタ 【共通】シリアルモードレジスタ 【UART】シリアルステータスレジスタ 【CSIO】シリアルステータスレジスタ 【LIN-UART】シリアルステータスレジスタ 【UART】拡張通信制御レジスタ 【CSIO】拡張通信制御レジスタ 【LIN-UART】拡張通信制御レジスタ
0x04F4	【UART】 RDR9/TDR9 【CSIO】 RDR9/TDR9 【LIN-UART】 RDR9/TDR9		【UART】 BGR9 【CSIO】 BGR9 【LIN-UART】 BGR9		―――ch.9――― 【UART】送受信データレジスタ 【CSIO】送受信データレジスタ 【LIN-UART】送受信データレジスタ 【UART】ボーレートジェネレータレジスタ 【CSIO】ボーレートジェネレータレジスタ 【LIN-UART】ボーレートジェネレータレジスタ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x04F8	予約				
0x04FC	【共通】 FCR19	【共通】 FCR09	【共通】 FBYTE9		―――ch.9――― 【共通】FIFO 制御レジスタ 1 【共通】FIFO 制御レジスタ 0 【共通】FIFO バイトレジスタ

40.4.1　共通レジスタ

マルチファンクションシリアルインタフェースの共通レジスタについて説明します。

40.4.1.1. シリアルモードレジスタ : SMR (Serial Mode Register)

40.4.1.2. FIFO 制御レジスタ 1 : FCR1 (FIFO Control Register 1)

40.4.1.3. FIFO 制御レジスタ 0 : FCR0 (FIFO Control Register 0)

40.4.1.4. FIFO バイトレジスタ : FBYTE (FIFO BYTE Register)

40.4.1.1　シリアルモードレジスタ : SMR (Serial Mode Register)

シリアルモードレジスタのビット構成について説明します。

シリアル通信方式 (UART または I²C) を選択します。bit3～bit0 は UART か CSIO か I²C かにより機能が変わります。

■ SMR : アドレス Base_addr + 01H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MD[2:0]			予約	SBL/ SCINV/ RIE	BDS/TIE	SCKE/ (予約)	SOE/ (予約)
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W0	R/W	R/W (R/W0)	R/W (R/W0)	R/W (R/W0)

[bit7～bit5] MD[2:0] (MoDe) : 動作モード
通信方式を設定します。

000_B : 動作モード 0 (非同期ノーマルモード)に設定されます。
001_B : 動作モード 1 (非同期マルチプロセッサモード)に設定されます。
010_B : 動作モード 2 (CSIO モード)に設定されます。
011_B : 動作モード 3 (LIN 通信モード)に設定されます。
100_B : 動作モード 4 (I²C モード)に設定されます。

＜注意事項＞
- 上記設定以外は禁止です。
- 動作モード設定後、各レジスタを設定してください。
- 【UART】【CSIO】【LIN-UART】動作モードを切り換える場合は、プログラマブルクリア実行(SCR: UPCL=1)後、動作モードを切り換えてください。
- 【I²C】動作モードを切り換える場合は、I²C 禁止(ISMK:EN=0)後、動作モードを切り換えてください。
- ch.8,ch.9 を"100"に設定することは禁止です。

[bit4] 予約
0 を書いてください。

[bit3] SBL/SCINV/RIE (Stop Bit Length / Serial Clock Inversion / Receive Interrupt Enable) : ストップビット長選択ビット・シリアルクロック反転ビット・受信割込み許可ビット

【UART】【LIN-UART】
ストップビット(送信データのフレームエンドマーク)のビット長を設定します。

SBL=0, ESCR:ESBL=0 に設定した場合 : ストップビットは 1 ビットに設定されます。
SBL=1, ESCR:ESBL=0 に設定した場合 : ストップビットは 2 ビットに設定されます。
SBL=0, ESCR:ESBL=1 に設定した場合 : ストップビットは 3 ビットに設定されます。
SBL=1, ESCR:ESBL=1 に設定した場合 : ストップビットは 4 ビットに設定されます。

＜注意事項＞
- 受信時は、常にストップビットの 1 ビット目だけを検出します。
- 本ビットは送信が禁止(TXE=0)のときに設定してください。

【CSIO】
シリアルクロックフォーマットを反転するビットです。

"0"に設定した場合 :

シリアルクロック出力のマークレベルを"H"にします。送信データは、ノーマル転送では、シリアルクロックの立下りエッジ、SPI 転送では、シリアルクロックの立上りエッジに同期して出力します。受信データは、ノーマル転送では、シリアルクロックの立上りエッジ、SPI 転送では、シリアルクロックの立下りエッジでサンプリングします。

"1"に設定した場合 :

シリアルクロック出力のマークレベルを"L"にします。送信データは、ノーマル転送では、シリアルクロックの立上りエッジ、SPI 転送では、シリアルクロックの立下りエッジに同期して出力します。受信データは、ノーマル転送では、シリアルクロックの立下りエッジ、SPI 転送では、シリアルクロックの立上りエッジでサンプリングします。

＜注意事項＞
- 本ビットは、送受信が禁止(TXE=RXE=0)のときに設定してください。
- SCINV ビット設定後に、受信許可(SCR:RXE=1)に設定してください。

【I²C】
CPU への受信割込み要求出力を許可/禁止するビットです。

RIE ビットと受信データフラグビット(SSR:RDRF)が"1"の場合、またはエラーフラグビット(SSR:ORE)のいずれかが"1"の場合、受信割込み要求を出力します。

＜注意事項＞
I²C バス制御レジスタ (IBCR) の INT ビットを使用してデータを受信する場合､本ビットは"0"にしてください。

[bit2] BDS/TIE (Bit Direction Select / Transmit Interrupt Enable) : **転送方向選択ビット・送信割込み許可ビット**

【LIN-UART】
LIN-UART では本ビットに常に 0 を書いてください。

【UART】【CSIO】
転送シリアルデータを最下位ビット側から先に転送するか(LSB ファースト, BDS=0)最上位ビット側から先に転送するか(MSB ファースト, BDS=1)を選択するビットです。

＜注意事項＞
本ビットは、送受信が禁止(TXE=RXE=0)のときに設定してください。

【I²C】
CPU への送信割込み要求出力を許可/禁止するビットです。
TIE ビットと SSR: TDRE ビットが"1"の場合、送信割込み要求を出力します。

＜注意事項＞
I²C バス制御レジスタ(IBCR)の INT ビットを使用してデータを送信する場合、本ビットは"0"にしてください。

[bit1] SCKE (Serial ClocK Enable) : **シリアルクロック出力許可ビット**

【UART】【LIN-UART】
UART/LIN-UART では本ビットを使用しません。読出し値は"0"です。常に"0"を書き込んでください。

【CSIO】
シリアルクロックの入出力ポートを制御するビットです。

"0"に設定: シリアルクロック出力を禁止

"1"に設定: シリアルクロック出力を許可
SCK 端子として使用する場合は GPIO 設定も行ってください。

＜注意事項＞
SCK 端子をシリアルクロック入力(SCKE=0)として使用する場合は、汎用入出力ポートを入力ポートに設定してください。また、外部クロック選択ビットによって外部クロックを選択(BGR:EXT=1)してください。

【I²C】
予約ビットです。本ビットには必ず"0"を設定してください。

[bit0] SOE (Serial Output Enable) : **シリアルデータ出力許可ビット**

【UART】【CSIO】【LIN-UART】
シリアルデータの出力を許可/禁止するビットです。

シリアルデータ出力(SOE=1)の場合、SOT 端子は汎用入出力ポート(DDR)の設定にかかわらず、SOT 端子として機能します。SOT 端子として使用する場合は GPIO 設定も行ってください。

【I²C】
予約ビットです。本ビットには必ず"0"を設定してください。

40.4.1.2 FIFO 制御レジスタ 1 : FCR1 (FIFO Control Register 1)

FIFO 制御レジスタ 1 のビット構成について説明します。

FIFO 制御レジスタ(FCR1)は、FIFO のテスト設定，送受信 FIFO の選択，送信 FIFO 割込み許可の設定および割込みフラグの制御を行います。

■ FCR1 : アドレス Base_addr + 0C$_H$ (アクセス : バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		予約	FLSTE	FRIIE	FDRQ	FTIE	FSEL
初期値	0	0	ー	0	0	1	0	0
属性	R/W0	R/W0	R0,W0	R/W	R/W	R(RM1), W	R/W	R/W

[bit7, bit6] 予約
必ず"0"を書き込んでください。

[bit5] 予約
読出し値は"0"です。常に"0"を書き込んでください。

[bit4] FLSTE (Flag for data LoST detection Enable) : 再送データロスト検出許可ビット
FIFO 再送データロストフラグ(FLST)検出を許可するビットです。

"0"に設定した場合 : FLST ビット検出禁止
"1"に設定した場合 : FLST ビット検出許可

＜注意事項＞
本ビットに"1"を設定する場合、FSET ビットに"1"を設定してから本ビットに"1"を設定してください。

[bit3] FRIIE (Flag for Receive FIFO IdlE detection Enable) : 受信 FIFO アイドル検出許可ビット
受信 FIFO に有効なデータが存在した状態で 8 ビット時間以上の受信アイドル状態を検出するかどうかを設定するビットです。受信割込み許可(SCR:RIE=1)されていると、受信アイドル状態が検出されると受信割込みが発生します。

"0"に設定した場合 : 受信アイドル状態検出禁止
"1"に設定した場合 : 受信アイドル状態検出許可

＜注意事項＞
受信 FIFO を使用する場合、本ビットを"1"に設定してください。

[bit2] FDRQ (transmit FIFO Data ReQuest) : **送信 FIFO データ要求ビット**
送信 FIFO のデータ要求ビットです。本ビットが"1"のとき、送信データを要求していることを示します。このとき、送信 FIFO 割込み許可(FTIE=1)されていると、FIFO 送信割込み要求を出力します。

FDRQ セット条件

- FBYTE(送信用)=0 (送信 FIFO がエンプティ)

FDRQ リセット条件

- 本ビットへの"0"書込み
- 送信 FIFO がフルになった場合

＜注意事項＞
- 送信 FIFO 許可のときに"0"書込みは有効です。
- FBYTE (送信用)=0 のときに本ビットへの"0"書込みは禁止です。
- 本ビットに"1"を設定した場合動作に影響を与えません。
- リードモディファイライト系命令時、"1"が読み出されます。
- 本ビットが"0"のときに FSEL ビットの変更は禁止です。

[bit1] FTIE (Flag for Transmit Interrupt Enable) : **送信 FIFO 割込み許可ビット**
送信 FIFO の割込み許可ビットです。本ビットに"1"を設定すると FDRQ ビットが"1"のときに割込みが発生します。

[bit0] FSEL (FIFO SELect) : **FIFO 選択ビット**
送受信 FIFO を選択するビットです。

"0"に設定した場合 ： 送信 FIFO : FIFO1, 受信 FIFO : FIFO2 に割当てられます。
"1"に設定した場合 ： 送信 FIFO : FIFO2, 受信 FIFO : FIFO1 に割当てられます。

＜注意事項＞
- 本ビットは、FIFO リセット(FCR0:FCL2, FCL1=1)ではクリアされません。
- 本ビットを変更する場合は、FIFO 動作禁止(FCR0:FE2, FE1=0)にしてから行ってください。

40.4.1.3 FIFO 制御レジスタ 0 : FCR0 (FIFO Control Register 0)

FIFO 制御レジスタ 0 のビット構成について説明します。

FIFO 制御レジスタ 0 (FCR0)は、FIFO 動作の許可/禁止, FIFO リセット，リードポインタの保存，再送信設定を行います。

■ FCR0 : アドレス Base_addr + 0D$_H$ (アクセス : バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	FLST	FLD	FSET	FCL2	FCL1	FE2	FE1
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R,WX	R/W	R/W	R/W	R/W	R/W	R/W

[bit7] 予約
必ず"0"を書き込んでください。

[bit6] FLST (FIFO data LoST) : FIFO 再送データロストフラグビット
送信 FIFO の再送データが失われたことを示すビットです。

FLST セット条件

FIFO 制御レジスタ 1 (FCR1)の FLSTE ビットが"1"で、送信 FIFO のライトポインタと FSET ビットによって保存 したリードポインタが一致しているときに FIFO へ書き込んだ(上書きした)場合

FLST リセット条件

- ☐ FIFO リセット(FCL への"1"書込み)
- ☐ FSET ビットへ"1"書込み

本ビットに"1"が設定されると FSET ビットで保存したリードポインタが示すデータを上書きしてしまい、エラーが発生しても FLD ビットによって再送の設定ができません。本ビットに"1"が設定された状態で再送を行う場合には FIFO リセットを実施し、再度 FIFO にデータを書き込んでください。

[bit5] FLD (FIFO pointer reLoaD bit) : FIFO ポインタリロードビット
送信 FIFO に FSET ビットによって保存したデータをリードポインタにリロードするビットです。本ビットは通信エラーなどが発生し再送するときに使用します。再送設定が完了した場合、本ビットは"0"になります。

＜注意事項＞
- ■ 本ビットが"1"にセットされている間、リードポインタへのリロード中のため FIFO リセット以外の書込みは行わないでください。
- ■ FIFO 許可状態または送信中に本ビットに"1"を設定することは禁止です。
- ■ TIE ビットと TBIE ビットは"0"にしてから本ビットに"1"を書込み、送信 FIFO 許可後 TIE ビットと TBIE ビットを"1"にしてください。

[bit4] FSET (FIFO pointer SET) : FIFO ポインタ保存ビット
送信 FIFO のリードポインタを保存するビットです。通信前にリードポインタを保存すれば、通信エラーなどが発生した場合、FLST ビットが"0"であれば、再送可能となります。

"1"に設定した場合 : 現在のリードポインタの値を保存します。
"0"に設定した場合 : 動作に影響しません。

＜注意事項＞
送信バイト数 (FBYTE) が 0 を示しているときに本ビットを"1"に設定してください。

[bit3] FCL2 (FIFO Clear 2) : FIFO2 リセットビット
FIFO2 をリセットするビットです。
本ビットを"1"に設定すると、FIFO2 の内部状態を初期化します。
FCR1 : FLST ビットのみ初期化され、FCR1/FCR0 レジスタのほかのビットは保持されます。

＜注意事項＞
- 送受信を禁止してから、FIFO2 リセットを実行してください。
- 送信 FIFO 割込み許可ビットを"0"にしてから実行してください。
- FBYTE2 レジスタの有効データ数は 0 になります。

[bit2] FCL1 (FIFO Clear 1) : FIFO1 リセットビット
FIFO1 をリセットするビットです。
本ビットを"1"に設定すると、FIFO1 の内部状態を初期化します。
FCR1:FLST ビットのみ初期化され、FCR1/FCR0 レジスタのほかのビットは保持されます。

＜注意事項＞
- 送受信を禁止してから、FIFO1 リセットを実行してください。
- 送信 FIFO 割込み許可ビットを"0"にしてから実行してください。
- FBYTE1 レジスタの有効データ数は 0 になります。

[bit1] FE2 (FIFO Enable 2) : FIFO2 動作許可ビット
FIFO2 の動作を許可/禁止するビットです。

① モード 1, 2, 3 の場合
- FIFO2 を使用する場合、本ビットに"1"を設定してください。
- FIFO2 を送信 FIFO に設定し(FCR1:FSEL=1)、本ビットに"1"を書き込んだときに FIFO2 にデータが存在し、送信許可(SCR:TXE=1)のとき、直ちに送信を開始します。このとき、SCR:TIE ビットと SCR:TBIE ビットは"0"にしてから本ビットに"1"を書込み、SCR:TIE ビットと SCR:TBIE ビットを"1"にしてください。
- FSEL ビットによって受信 FIFO として選択された場合、受信エラーが発生すると本ビットは"0"にクリアされ受信エラーがクリアされない限り本ビットに"1"を設定することはできません。
- 送信 FIFO で使用する場合には送信バッファがエンプティ(SSR:TDRE=1)、受信 FIFO で使用する場合には受信バッファがエンプティ(SSR:RDRF=0)のときに本ビットに"1"または"0"を設定してください。
- FIFO2 を禁止にしても FIFO2 の状態は保持されます。

② モード 4 の場合
- FIFO2 を使用する場合、本ビットに"1"を設定してください。
- FCR1:FSEL ビットによって受信 FIFO として選択された場合、受信エラーが発生後に本ビットは"0"にクリアされ、受信エラーがクリアされない限り、本ビットに"1"を設定することはできません。
- FIFO2 を送信 FIFO で使用する場合には送信データがエンプティ(SSR:TDRE=“1”)のときに本ビットに“1”または“0”を設定してください。
- FIFO2 を受信 FIFO で使用する場合には、I^2C インタフェースを禁止(ISMK:EN=0)、動作フラグ(IBCR:ACT)が”0”または割込みフラグ(IBCR:INT)が”1”で受信バッファがエンプティ（SSR:RDRF=“0”）および受信 FIFO に有効なデータがない(FBYTE2=0)ときに本ビットにまたは“0”を設定してください。
- FIFO2 を受信 FIFO で使用する場合には、I^2C インタフェースを禁止(ISMK:EN=0)または動作フラグ(IBCR:ACT)が”0”または割込みフラグ(IBCR：INT)が”1”で受信バッファがエンプティ（SSR:RDRF=“0”）のときに本ビットにまたは“1”を設定してください。
- FIFO2 を禁止にしても FIFO2 の状態は保持されます。

＜注意事項＞
- IBSR:BB ビットが"0"または IBCR:INT ビットが"1"のときに許可/禁止の変更を行ってください。
- 受信 FIFO として選択されていて予約アドレスを検出し、スレーブ送信として動作する場合、予約アドレス検出による割込みで本ビットを"0"にし、IBCR:ACKE="0"にしてください。
- 受信 FIFO として使用していて本ビットを"1"から"0"に変更したときに SSR の SSR:RDRF ビットが"1"になっていると"0"になるまで受信 FIFO は禁止になりません。
- 送信 FIFO として使用していて FIFO2 にデータが存在し、本ビットを"0"から"1"に変更する場合、SMR:TIE ビットを"0"にしてから本ビットに"1"を書込み、SMR:TIE ビットを"1"にしてください。

[bit0] FE1 (FIFO Enable 1) : FIFO1 動作許可ビット
FIFO1 の動作を許可/禁止するビットです。

① モード 1, 2, 3 の場合
- FIFO1 を使用する場合、本ビットに"1"を設定してください。
- FIFO1 を送信 FIFO に設定し(FCR1:FSEL=0)、本ビットに"1"を書き込んだときに FIFO1 にデータが存在し、送信許可(SCR:TXE=1)のとき、直ちに送信を開始します。このとき、SCR:TIE ビットと SCR:TBIE ビットは"0"にしてから本ビットに"1"を書込み、TIE ビットと SCR:TBIE ビットを"1"にしてください。
- FSEL ビットによって受信 FIFO として選択された場合、受信エラーが発生すると本ビットは"0"にクリアされ受信エラーがクリアされない限り本ビットに"1"を設定することはできません。
- 送信 FIFO で使用する場合には送信バッファがエンプティ(SSR:TDRE=1)、受信 FIFO で使用する場合には受信バッファがエンプティ(SSR:RDRF=0)のときに本ビットに"1"または"0"を設定してください。
- FIFO1 を禁止にしても FIFO1 の状態は保持されます。

② モード 4 の場合
- FIFO1 を使用する場合、本ビットに"1"を設定してください。
- FCR1:FSEL ビットによって受信 FIFO として選択された場合、受信エラーが発生後に本ビットは"0"にクリアされ、受信エラーがクリアされない限り、本ビットに"1"を設定することはできません。
- FIFO1 を送信 FIFO で使用する場合には送信データがエンプティ(SSR:TDRE=“1”)のときに本ビットに“1”または“0”を設定してください。
- FIFO1 を受信 FIFO で使用する場合には、I²C インタフェースを禁止(ISMK:EN=0)、動作フラグ(IBCR:ACT)が”0”または割込みフラグ(IBCR:INT)が”1”で受信バッファがエンプティ（SSR:RDRF=“0”）および受信 FIFO に有効なデータがない(FBYTE2=0)ときに本ビットにまたは“0”を設定してください。
- FIFO1 を受信 FIFO で使用する場合には、I²C インタフェースを禁止(ISMK:EN=0)または動作フラグ(IBCR:ACT)が”0”または割込みフラグ(IBCR：INT)が”1”で受信バッファがエンプティ（SSR:RDRF=“0”）のときに本ビットにまたは“1”を設定してください。
- FIFO1 を禁止にしても FIFO1 の状態は保持されます。

＜注意事項＞
- IBSR:BB ビットが"0"または IBCR:INT ビットが"1"のときに許可/禁止の変更を行ってください。
- 受信 FIFO として選択されていて予約アドレスを検出し、スレーブ送信として動作する場合、予約アドレス検出による割込みで本ビットを"0"にし、IBCR:ACKE="0"にしてください。
- 受信 FIFO として使用していて本ビットを"1"から"0"に変更したときに SSR の RDRF ビットが"1"になっていると"0"になるまで受信 FIFO は禁止になりません。
- 送信 FIFO として使用していて FIFO1 にデータが存在し、本ビットを"0"から"1"に変更する場合、SMR:TIE ビットを"0"にしてから本ビットに"1"を書込み、SMR:TIE ビットを"1"にしてください。

40.4.1.4 FIFO バイトレジスタ ：FBYTE (FIFO BYTE Register)

FIFO バイトレジスタのビット構成について説明します。

■ FBYTE： アドレス Base_addr + 0E$_H$ (アクセス ： バイト, ハーフワード, ワード)

本レジスタは読出し時と書込み時で機能が異なります。
読出し時は、FIFO バイトレジスタ(FBYTE)は、FIFO の有効なデータ数を示します。
書込み時は、受信 FIFO で所定のデータ数を受信したときに受信割込みを発生させるかを設定できます。

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	FBYTE2[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FBYTE1[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

[bit15～bit8] FBYTE2[7:0] (FIFO Byte 2) : FIFO2 データ数表示ビット

[bit7～bit0] FBYTE1[7:0] (FIFO Byte 1) : FIFO1 データ数表示ビット
FBYTE レジスタは、FIFO に書込みまたは受信した有効なデータ数を示し、FCR1:FSEL ビットの設定によって以下のようになります。

FSEL	FIFO 選択	データ数表示
0	FIFO2 : 受信 FIFO, FIFO1 : 送信 FIFO	FIFO2 : FBYTE2, FIFO1 : FBYTE1
1	FIFO2 : 送信 FIFO, FIFO1 : 受信 FIFO	FIFO2 : FBYTE2, FIFO1 : FBYTE1

- FBYTE の転送数の初期値は 08$_H$ です。
- 受信 FIFO の FBYTE に受信割込みフラグを発生させるデータ数を設定します。その設定された転送数と FBYTE のデータ数表示が一致すると割込みフラグ(SSR:RDRF)が"1"にセットされます。
- 受信 FIFO アイドル検出許可ビット(FRIIE)が"1"で受信 FIFO に存在するデータ数が転送数に達しない場合、受信アイドル状態がボーレートクロックで 8 クロック以上続くと割込みフラグ(RDRF)が"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは"0 にリセットされ、再度 8 クロックをカウントします。受信 FIFO が禁止されるとそのカウンタは 0"にリセットされます。受信 FIFO にデータが残っている状態で受信 FIFO を許可すると再度、カウントを開始します。

- 【CSIO】マスタ動作で、データを受信する場合(マスタ受信)、TIE ビットと TBIE ビットを"0"にし送信 FIFO の FBYTE レジスタに受信データ数を設定し、FDRQ ビットに"0"を書きます。その後、TXE ビットが"1"のとき設定データ分のシリアルクロックが出力され、設定値分データを受信することができます。TIE ビット, TBIE ビットに"1"を設定したい場合には FDRQ が"1"になった後に"1"に設定してください。
- 【I²C】マスタ動作で、データを受信する場合(マスタ受信)、TIE ビットを"0"にし送信 FIFO の FBYTE に受信データ数を設定し、FDRQ ビットに"0"を書きます。設定データ分の SCL のクロックが出力され、その後、INT ビットが"1"になります。TIE ビットに"1"を設定したい場合には FDRQ ビットが"1"になった後に"1"に設定してください。

＜注意事項＞

- 【UART】【LIN-UART】送信 FIFO の FBYTE には 00_H を設定してください。
- 【CSIO】【I²C】マスタ動作で、データを受信するとき以外、送信 FIFO の FBYTE は"8'h00"を設定してください。
- 【CSIO】マスタ動作でデータを受信するときの送信データ数の設定は送信 FIFO がエンプティで SCR:TIE ビット, SSR:TBIE ビットが"0"のときに行ってください。
- 【I²C】マスタ動作でデータを受信するときの送信データ数の設定は送信 FIFO がエンプティで SCR:TIE ビットが"0"のときに行ってください。
- 【CSIO】マスタ動作でデータを受信中に受信禁止(SCR:RXE=0)にする場合には、送信 FIFO を禁止にしてから送受信を禁止にしてください。
- 【I²C】マスタ動作でデータを受信中に I²C インタフェースを禁止(ISMK:EN=0)にする場合には、送受信 FIFO を禁止にしてから禁止してください。
- 【共通】受信 FIFO の FBYTE は 1 以上のデータを設定してください。
- 【共通】送受信を禁止してから変更してください。
- 【共通】本レジスタはリードモディファイライト系命令を使用することはできません。
- 【共通】FIFO 容量を超える設定は禁止です。

40.4.2 UART 時のレジスタ

UART 時のレジスタについて示します。

40.4.2.1 シリアル制御レジスタ : SCR (Serial Control Register)

40.4.2.2 シリアルステータスレジスタ : SSR (Serial Status Register)

40.4.2.3 拡張通信制御レジスタ : ESCR (Extended Serial Control Register)

40.4.2.4 送信データレジスタ/受信データレジスタ : RDR/TDR (Receive Data Register / Transmit Data Register)

40.4.2.5 ボーレートジェネレータレジスタ : BGR (Baud rate Generator Register)

40.4.2.1 シリアル制御レジスタ ：SCR (Serial Control Register)

シリアル制御レジスタのビット構成について説明します。

シリアル制御レジスタ(SCR)は、送受信の許可/禁止，送受信割込みの許可/禁止，送信バスアイドル割込みの許可/禁止, UART リセットをすることができます。

■ SCR： アドレス Base_addr + 00_H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	UPCL	予約	予約	RIE	TIE	TBIE	RXE	TXE
初期値	0	－	－	0	0	0	0	0
属性	R0,W	RX,WX	RX,WX	R/W	R/W	R/W	R/W	R/W

ビット名		機　　能
bit7	UPCL : プログラマブルクリアビット	UART の内部状態を初期化するビットです。 "1"を設定した場合 ： ・UART を直接リセット(ソフトウェアリセット)します。ただし、レジスタの設定は維持されます。その際、送受信状態のものは直ちに切断されます。 ・ボーレートジェネレータは、BGR レジスタの設定値をリロードし、再スタートします。 ・すべての送受信割込み要因(SSR:PE, FRE, ORE, RDRF, TDRE, TBI)は初期化(000011_B)されます。 "0"を設定した場合 ：動作に影響ありません。 読出し時は、常に"0"が読み出されます。 (注意事項) ・割込み禁止に設定した後に、プログラマブルクリアを実行してください。 ・FIFO 使用時は、FIFO 禁止(FCR0:FE2, FE1=0)にしてからプログラマブルクリアを実行してください。
bit6, bit5	予約	読み出した場合 ： 値は不定です。 書き込みした場合 ： 動作に影響しません。
bit4	RIE : 受信割込み許可ビット	・CPU への受信割込み要求出力を許可/禁止するビットです。 ・RIE ビットと受信データフラグビット(SSR:RDRF)が"1"の場合、または、エラーフラグビット(SSR:PE, ORE, FRE)のいずれかが"1"の場合、受信割込み要求を出力します。
bit3	TIE : 送信割込み許可ビット	・CPU への送信割込み要求出力を許可/禁止するビットです。 ・TIE ビットと SSR:TDRE ビットが"1"の場合、送信割込み要求を出力します。

ビット名		機　　能
bit2	TBIE : 送信バスアイドル割込み 許可ビット	・CPUへの送信バスアイドル割込み要求出力を許可/禁止するビットです。 ・TBIE ビットと SSR:TBI ビットが"1"のとき、送信バスアイドル割込み要求を出力します。
bit1	RXE : 受信動作許可ビット	UART の受信動作を許可/禁止します。 ・"0"に設定した場合 ： 受信動作が禁止されます。 ・"1"に設定した場合 ： 受信動作が許可されます。 (注意事項) ・受信動作許可(RXE=1)にしても、スタートビットの立下りエッジ(NRZ フォーマット(ESCR:INV=0)の場合)が入力されないと受信動作を開始しません。(反転 NRZ フォーマット(ESCR:INV=1)の場合は、立上りエッジが入力されるまで受信動作を開始しません) ・受信中に受信動作を禁止(RXE=0)した場合には、直ちに受信動作を停止します。
bit0	TXE : 送信動作許可ビット	UART の送信動作を許可/禁止します。 ・"0"に設定した場合 ： 送信動作が禁止されます。 ・"1"に設定した場合 ： 送信動作が許可されます。 (注意事項) ・送信中に送信動作を禁止(TXE=0)した場合には、直ちに送信動作を停止します。

40.4.2.2 シリアルステータスレジスタ ：SSR (Serial Status Register)

シリアルステータスレジスタのビット構成について説明します。

シリアルステータスレジスタ(SSR)は、送受信の状態の確認，受信エラーフラグの確認，また,受信エラーフラグをクリアします。

■ SSR： アドレス Base_addr + 02H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	REC	予約	PE	FRE	ORE	RDRF	TDRE	TBI
初期値	0	－	0	0	0	0	1	1
属性	R0,W	RX,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

ビット名		機　　能
bit7	REC : 受信エラーフラグ クリアビット	シリアルステータスレジスタ(SSR)の PE, FRE, ORE フラグをクリアするビットです。 ・"1"書込みで、エラーフラグがクリアされます。 ・"0"書込みは、動作に影響しません。 読み出した場合、常に"0"が読み出されます。
bit6	予約	読み出した場合 ： 値は不定です。 書き込みした場合 ： 動作に影響しません。
bit5	PE : パリティエラーフラグビット (動作モード 0 のみ機能)	"0"読出し ： パリティエラーなし "1"読出し ： パリティエラーあり ・ESCR:PEN=1 で受信時にパリティエラーが発生すると"1"にセットされ、シリアルステータスレジスタ(SSR)の REC ビットに"1"を書き込むとクリアされます。 ・PE ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・本フラグがセットされた場合は､受信データレジスタ(RDR)のデータは無効です。 ・受信 FIFO 使用時に本フラグがセットされた場合は、受信 FIFO の許可ビットがクリアされ、受信データは受信 FIFO に格納されません。

ビット名		機　　能
bit4	FRE : フレーミングエラーフラグビット	"0"読出し：フレーミングエラーなし "1"読出し：フレーミングエラーあり ・受信時にフレーミングエラーが発生すると"1"にセットされ、シリアルステータスレジスタ(SSR)の REC ビットに"1"を書き込むとクリアされます。 ・FRE ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・本フラグがセットされた場合は、受信データレジスタ(RDR)のデータは無効です。 ・受信 FIFO 使用時に本フラグがセットされた場合は、受信 FIFO の許可ビットがクリアされ、受信データは受信 FIFO に格納されません。
bit3	ORE : オーバラン エラーフラグビット	"0"読出し：オーバランエラーなし "1"読出し：オーバランエラーあり ・受信時にオーバランが発生すると"1"にセットされ、シリアルステータスレジスタ(SSR)の REC ビットに"1"を書き込むとクリアされます。 ・ORE ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・本フラグがセットされた場合は、受信データレジスタ(RDR)のデータは無効です。 ・受信 FIFO 使用時に本フラグがセットされた場合は、受信 FIFO の許可ビットがクリアされ、受信データは受信 FIFO に格納されません。
bit2	RDRF : 受信データ フルフラグビット	"0"読出し：受信データレジスタ RDR がエンプティ "1"読出し：受信データレジスタ RDR にデータが存在する ・受信データレジスタ(RDR)の状態を示すフラグです。 ・RDR に受信データがロードされると、"1"にセットされ、受信データレジスタ(RDR)を読み出すと"0"にクリアされます。 ・RDRF ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら RDRF が"1"にセットされます。 ・受信 FIFO 使用時に、受信 FIFO アイドル検出許可ビット(FCR1:FRIIE)が"1"で、受信 FIFO に所定のデータ数を受信せずに受信 FIFO にデータが残っていて受信アイドル状態がボーレートクロックで 8 クロック以上続いた場合、RDRF が"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは 0 にリセットされ、再度 8 クロックをカウントします。 ・受信 FIFO 使用時は、受信 FIFO がエンプティになると"0"にクリアされます。

ビット名		機　　能
bit1	TDRE : 送信データエンプティフラグビット	"0"読出し：送信データレジスタ TDR にデータが存在する "1"読出し：送信データレジスタ TDR にデータがエンプティ ・送信データレジスタ(TDR)の状態を示すフラグです。 ・TDR に送信データを書き込むと、"0"となり TDR に有効なデータが存在していることを示します。データが送信シフトレジスタにロードされて送信が開始されると"1"になり TDR に有効なデータが存在していないことを示します。 ・TDRE ビットと SCR:TIE ビットが"1"の場合、送信割込み要求を出力します。 ・シリアル制御レジスタ(SCR)の UPCL ビットに"1"をセットすると TDRE ビットは"1"になります。 ・送信 FIFO 使用時の TDRE ビットのセット/リセットタイミングは、「40.5.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング」を参照してください。
bit0	TBI : 送信バスアイドルフラグビット	"0"読出し：送信中 "1"読出し：送信動作なし ・UART が送信動作をしていないことを示すビットです。 ・送信データレジスタ(TDR)へ送信データを書き込んだ場合に本ビットは"0"になります。 ・送信データレジスタがエンプティ(TDRE=1)で、送信動作をしていない場合に本ビットが"1"になります。 ・シリアル制御レジスタ(SCR)の UPCL ビットに"1"をセットすると TBI ビットは"1"になります。 ・本ビットが"1"で、送信バスアイドル割込許可(SCR:TBIE=1)されていると送信割込み要求を出力します。

40.4.2.3 拡張通信制御レジスタ ：ESCR (Extended Serial Control Register)

拡張通信制御レジスタのビット構成について説明します。

拡張通信制御レジスタ(ESCR)は、送受信データ長の設定，パリティビットの許可/禁止，パリティビットの選択，シリアルデータフォーマットの反転，ストップビット長の選択の設定ができます。

■ ESCR： アドレス Base_addr + 03H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	ESBL	INV	PEN	P	L[2:0]		
初期値	ー	0	0	0	0	0	0	0
属性	RX,W0	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ビット名		機　　能
bit7	予約	本ビットには必ず"0"を設定してください。
bit6	ESBL : 拡張ストップビット長選択ビット	ストップビット(送信データのフレームエンドマーク)のビット長を設定します。 SMR:SBL=0, ESBL=0 に設定した場合 : ストップビットは 1 ビットに設定されます。 SMR:SBL=1, ESBL=0 に設定した場合 : ストップビットは 2 ビットに設定されます。 SMR:SBL=0, ESBL=1 に設定した場合 : ストップビットは 3 ビットに設定されます。 SMR:SBL=1, ESBL=1 に設定した場合 : ストップビットは 4 ビットに設定されます。 (注意事項) ・受信時は、常にストップビットの 1 ビット目だけを検出します。 ・本ビットは送信が禁止(SCR:TXE=0)のときに設定してください。
bit5	INV : 反転シリアルデータフォーマットビット	シリアルデータフォーマットを NRZ フォーマット、または反転 NRZ フォーマットを選択します。 ・"0"に設定した場合 : NRZ フォーマットに設定されます。 ・"1"に設定した場合 ： 反転 NRZ フォーマットに設定されます。
bit4	PEN : パリティ許可ビット(動作モード 0 のみ機能)	パリティビットの付加(送信時)と検出(受信時)を行うかどうかを設定します。 ・"0"に設定した場合 ： パリティビットは付加されません。 ・"1"に設定した場合 ： パリティビットは付加されます。 (注意事項) 動作モード 1 のときは、本ビットは内部で"0"に固定されます。

ビット名		機　　能
bit3	P : パリティ選択ビット (動作モード0のみ機能)	パリティあり(ESCR:PEN=1)に設定した場合に、奇数パリティ"1"か偶数パリティ"0"のいずれかに設定します。 ・"0"に設定した場合 ： 偶数パリティに設定されます。 ・"1"に設定した場合 ： 奇数パリティに設定されます。
bit2 ～ bit0	L2, L1, L0 : データ長選択ビット	送受信データのデータ長を指定します。 ・000_Bに設定した場合 ： データ長は、8 ビットに設定されます。 ・001_Bに設定した場合 ： データ長は、5 ビットに設定されます。 ・010_Bに設定した場合 ： データ長は、6 ビットに設定されます。 ・011_Bに設定した場合 ： データ長は、7 ビットに設定されます。 ・100_Bに設定した場合 ： データ長は、9 ビットに設定されます。 (注意事項) ・ 上記以外の設定は禁止です。 ・ 動作モード 1 では、データ長は、7, 8 ビットに設定してください。その他の設定は禁止です。

40.4.2.4 送信データレジスタ/受信データレジスタ ：RDR/TDR (Receive Data Register / Transmit Data Register)

送信データレジスタ/受信データレジスタのビット構成について説明します。

受信データと送信データレジスタは同一アドレスに配置されています。読み出した場合は、受信データレジスタとして機能し、書き込みした場合は送信データレジスタとして機能します。FIFO 動作許可の場合、RDR/TDR アドレスは FIFO 読出し、書込みアドレスとなります。

■ RDR/TDR： アドレス Base_addr + 04H (アクセス ： バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							D8
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

リード

受信データレジスタ(RDR)は、シリアルデータ受信用の 9 ビットのデータバッファレジスタです。

- シリアル入力端子(SIN)に送られてきたシリアルデータ信号がシフトレジスタで変換されて、受信データレジスタ(RDR)に格納されます。
- データ長に応じ、以下のように上位ビットに"0"が入ります。

データ長	D8	D7	D6	D5	D4	D3	D2	D1	D0
9 ビット	X	X	X	X	X	X	X	X	X
8 ビット	0	X	X	X	X	X	X	X	X
7 ビット	0	0	X	X	X	X	X	X	X
6 ビット	0	0	0	X	X	X	X	X	X
5 ビット	0	0	0	0	X	X	X	X	X

(X は受信データビット)

- 受信データが、受信データレジスタ(RDR)に格納されると、受信データフルフラグビット(SSR: RDRF)が"1"にセットされます。受信割込みが許可されている場合(SSR:RIE=1)、受信割込み要求が発生します。

- 受信データレジスタ(RDR)は、受信データフルフラグビット(SSR:RDRF)が"1"の状態で読み出してください。受信データフルフラグビット(SSR:RDRF)は、受信データレジスタ(RDR)を読み出すと自動的に"0"にクリアされます。
- 受信エラーが発生(SSR:PE, ORE, FRE のいずれかが"1")した場合、受信データレジスタ(RDR)のデータは無効となります。
- 動作モード 1 (マルチプロセッサモード)では、7 ビット, 8 ビット長の動作となり、受信した AD ビットは、D8 ビットに格納されます。
- 9 ビット長転送、および動作モード 1 の場合 RDR の読出しは 16 ビットアクセスで行います。

＜注意事項＞

- 受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら SSR:RDRF が"1"にセットされます。
- 受信 FIFO 使用時は、受信 FIFO がエンプティになると SSR:RDRF が"0"にクリアされます。
- 受信 FIFO 使用時に、受信エラーが発生(SSR:PE, ORE, FRE のいずれかが"1")した場合、受信 FIFO の許可ビットはクリアされ、受信データは受信 FIFO に格納しません。

ライト

送信データレジスタ(TDR)は、シリアルデータ送信用の 9 ビットデータバッファレジスタです。

- 送信動作が許可されている場合に(SCR:TXE=1)、送信するデータを送信データレジスタ(TDR)に書き込むと、送信データが送信用シフトレジスタに転送されシリアルデータに変換されて、シリアルデータ出力端子(SOUT)から送出されます。
- データ長に応じ、以下のように上位ビットから順に無効データとなります。

データ長	D8	D7	D6	D5	D4	D3	D2	D1	D0
9 ビット	X	X	X	X	X	X	X	X	X
8 ビット	無効	X	X	X	X	X	X	X	X
7 ビット	無効	無効	X	X	X	X	X	X	X
6 ビット	無効	無効	無効	X	X	X	X	X	X
5 ビット	無効	無効	無効	無効	X	X	X	X	X

(X は送信データビット)

- 送信データエンプティフラグ(SSR:TDRE)は、送信データが送信データレジスタ(TDR)に書き込まれると、"0"クリアされます。
- 送信データエンプティフラグ(SSR:TDRE)は、送信データが送信用シフトレジスタへ転送され、送信が開始されると、送信 FIFO が禁止または送信 FIFO がエンプティの場合、"1"にセットされます。
- 送信データエンプティフラグ(SSR:TDRE)が"1"のとき、送信データを書き込むことができます。送信割込みが許可されている場合には送信割込みが発生します。送信データの書込みは、送信割込みの発生によるか、送信データエンプティフラグ(SSR:TDRE)が"1"の状態で行ってください。
- 送信データエンプティフラグ(SSR:TDRE)が"0"で送信 FIFO が禁止または送信 FIFO がフルのときは、送信データを書き込むことはできません。
- 動作モード 1 (マルチプロセッサモード)では、7 ビット, 8 ビット長の動作となり、AD ビットの送信は、D8 ビットへの書込みにより行います。
- 9 ビット長転送、および動作モード 1 の場合 TDR への書込みは 16 ビットアクセスで行います。

＜注意事項＞

- 送信データレジスタは書込み専用のレジスタで、受信データレジスタは読出し専用のレジスタです。送受信レジスタは同一アドレスに配置されているため書込み値と読出し値が異なります。したがって、INC/DEC命令などリードモディファイライト(RMW)命令は使用できません。
- 送信 FIFO 使用時の送信データエンプティフラグ(SSR:TDRE)のセットタイミングは、「40.5.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング」を参照してください。

40.4.2.5 ボーレートジェネレータレジスタ ：BGR (Baud rate Generator Register)

ボーレートジェネレータレジスタのビット構成について説明します。

ボーレートジェネレータレジスタ(BGR)は、シリアルクロックの分周比を設定します。　　また、リロードカウンタのクロックソースとして外部クロックを選択できます。

■ BGR ： アドレス Base_addr + 06H (アクセス ： ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	EXT	BGR[14:8]						
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BGR[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15] EXT (EXTernal clock) : **外部クロック選択ビット**
ボーレート生成用内部リロードカウンタに、内部クロックソースを使用するか、外部クロックソースを使用するかを選択します。EXT=1 のとき、外部クロックソースを使用します。品種によっては実装されていません。

[bit14～bit0] BGR[14:0] (Baud rate GeneratoR) : **ボーレートジェネレータビット**
ボーレート生成用内部リロードカウンタのリロード値を設定します。
本レジスタにリロード値を書き込むとリロードカウンタはカウントを開始します。

＜注意事項＞

- ボーレートジェネレータレジスタ(BGR)への書込みは、16 ビットアクセスで行ってください。
- BGR の値が偶数の場合、シリアルクロックの"H"幅は"L"幅より 1 サイクル分短くなります。奇数の場合、デューティ比 1:1 になります。
- ボーレートジェネレータ動作中に外部クロックの設定(EXT=1)に変更する場合、ボーレートジェネレータ(BGR)に"0"を書込み、プログラムクリア(SCR:UPCL)実行後、外部クロック(EXT=1)に設定してください。
- ボーレートジェネレータレジスタ(BGR)の設定値を変更した場合、カウンタ値が"15h00"になってから、新しい設定値がリロードされます。したがって、新しい設定値を即有効にしたい場合は、BGR1/ BGR 0 の設定値を変更した後、プログラマブルクリア(SCR:UPCL)を実行してください。
- BGR へは 4 以上の値を設定してください。ただし、ボーレートの誤差とリロード値の設定に　　よって正常にデータを受信できないことがあります。

40.4.3 CSIO 時のレジスタ

CSIO 時のレジスタについて示します。

40.4.3.1. **シリアル制御レジスタ** : SCR (Serial Control Register)

40.4.3.2. **シリアルステータスレジスタ** : SSR (Serial Status Register)

40.4.3.3. **拡張通信制御レジスタ** : ESCR (Extended Serial Control Register)

40.4.3.4. **受信データレジスタ/送信データレジスタ** : RDR/TDR (Receive Data Register / Transmit Data Register)

40.4.3.5. **ボーレートジェネレータレジスタ** : BGR (Baud rate Generator Register)

40.4.3.1 シリアル制御レジスタ ：SCR (Serial Control Register)

シリアル制御レジスタのビット構成について説明します。

シリアル制御レジスタ(SCR)は、送受信の許可/禁止，送受信割込みの許可/禁止，送信バスアイドル割込みの許可/禁止, UART リセットをすることができます。

■ SCR： アドレス Base_addr + 00H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	UPCL	MS	SPI	RIE	TIE	TBIE	RXE	TXE
初期値	0	0	0	0	0	0	0	0
属性	R0,W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ビット名		機　　能
bit7	UPCL : プログラマブルクリアビット	CSIO の内部状態を初期化するビットです。 "1"を設定した場合 ： ・CSIO を直接リセット(ソフトウェアリセット)します。ただし、レジスタの設定は保持されます。その際、送受信状態のものは直ちに切断されます。 ・ボーレートジェネレータは、BGR レジスタの設定値をリロードし、再スタートします。 ・すべての送受信割込み要因(SSR:TDRE, TBI, RDRF, ORE)は初期化されます。 "0"を設定した場合 ：動作に影響しません。 読出し時は、常に"0"が読み出されます。 (注意事項) ・割込み禁止に設定した後に、プログラマブルクリアを実行してください。 ・FIFO 使用時は、FIFO 禁止(FCR0:FE2, FE1=0)にしてからプログラマブルクリアを実行してください。
bit6	MS : マスタ/スレーブ機能選択ビット	マスタまたはスレーブモードを選択します。 "0"に設定した場合 ：マスタモードに設定されます。 "1"に設定した場合 ：スレーブモードに設定されます。 (注意事項) ・スレーブモードを選択した場合、SMR:SCKE=0 であれば、外部クロックが直接入力されます。 ・MS ビット設定後に、受信許可(RXE=1)に設定してください。
bit5	SPI : SPI 対応ビット	本ビットは、SPI に対応した通信をさせるためのビットです。 "0"に設定した場合 ：ノーマル同期通信を行います。 "1"に設定した場合 ：SPI に対応します。

ビット名		機　　能
bit4	RIE : 受信割込み許可ビット	・CPU への受信割込み要求出力を許可/禁止するビットです。 ・RIE ビットと受信データフラグビット(SSR:RDRF)が"1"の場合、または、エラーフラグビット(ORE)のいずれかが"1"の場合、受信割込み要求を出力します。
bit3	TIE : 送信割込み許可ビット	・CPU への送信割込み要求出力を許可/禁止するビットです。 ・TIE ビットと SSR:TDRE ビットが"1"の場合、送信割込み要求を出力します。
bit2	TBIE : 送信バスアイドル割込み許可ビット	・CPUへの送信バスアイドル割込み要求出力を許可/禁止するビットです。 ・TBIE ビットと SSR:TBI ビットが"1"のとき、送信バスアイドル割込み要求を出力します。
bit1	RXE : 受信動作許可ビット	CSIO の受信動作を許可/禁止します。 ・"0"に設定した場合 ： データフレーム受信動作が禁止されます。 ・"1"に設定した場合 ： データフレーム受信動作が許可されます。 (注意事項) ・受信中に受信動作を禁止(RXE=0)した場合には、直ちに受信動作を停止します。 ・MS ビットおよび SMR:SCINV ビット設定後に、受信許可(RXE=1)に設定してください。
bit0	TXE : 送信動作許可ビット	CSIO の送信動作を許可/禁止します。 ・"0"に設定した場合 ： データフレーム送信動作が禁止されます。 ・"1"に設定した場合 ： データフレーム送信動作が許可されます。 (注意事項) ・送信中に送信動作を禁止(TXE=0)した場合には、直ちに送信動作を停止します。

40.4.3.2 シリアルステータスレジスタ ：SSR (Serial Status Register)

シリアルステータスレジスタのビット構成について説明します。

シリアルステータスレジスタ(SSR)は、送受信の状態の確認，受信エラーフラグの確認，また、受信エラーフラグをクリアします。

■ SSR： アドレス Base_addr + 02H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	REC	予約			ORE	RDRF	TDRE	TBI
初期値	0	－	－	－	0	0	1	1
属性	R0,W	RX,WX	RX,WX	RX,WX	R,WX	R,WX	R,WX	R,WX

ビット名		機　　能
bit7	REC : 受信エラーフラグクリアビット	シリアルステータスレジスタ(SSR)のOREフラグをクリアするビットです。 ・"1"書込みで、エラーフラグがクリアされます。 ・"0"書込みは、動作に影響しません。 読み出した場合、常に"0"が読み出されます。
bit6 ～ bit4	予約	読み出した場合 ： 値は不定です。 書き込みした場合 ： 動作に影響しません。
bit3	ORE : オーバランエラーフラグビット	"0"読出し ： オーバランエラーなし "1"読出し ： オーバランエラーあり ・受信時にオーバランが発生すると"1"にセットされ、シリアルステータスレジスタ(SSR)の REC ビットに"1"を書き込むとクリアされます。 ・ORE ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・本フラグがセットされた場合は、受信データレジスタ(RDR)のデータは無効です。 ・受信FIFO使用時に本フラグがセットされた場合は、受信FIFOの許可ビットがクリアされ、受信データは受信 FIFO に格納されません。

ビット名		機　　能
bit2	RDRF : 受信データ フルフラグビット	"0"読出し ： 受信データレジスタ RDR がエンプティ "1"読出し ： 受信データレジスタ RDR にデータが存在する ・受信データレジスタ(RDR)の状態を示すフラグです。 ・RDR に受信データがロードされると、"1"にセットされ、受信データレジスタ(RDR)を読み出すと"0"にクリアされます。 ・RDRF ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら RDRF ビットが"1"にセットされます。 ・受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信せずに受信 FIFO にデータが残っていて受信アイドル状態がボーレートクロックで 8 クロック以上続いた場合、RDRF ビットが"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは 0 にリセットされ、再度 8 クロックをカウントします。 ・受信 FIFO 使用時は、受信 FIFO がエンプティになると"0"にクリアされます。
bit1	TDRE : 送信データエンプティフラグビット	"0"読出し ： 送信データレジスタ TDR にデータが存在する "1"読出し ： 送信データレジスタがエンプティ ・送信データレジスタ(TDR)の状態を示すフラグです。 ・TDR に送信データを書き込むと、"0"となり TDR に有効なデータが存在していることを示します。データが送信シフトレジスタにロードされて送信が開始されると"1"になり TDR に有効なデータが存在していないことを示します。 ・TDRE ビットと SCR:TIE ビットが"1"の場合、送信割込み要求を出力します。 ・シリアル制御レジスタ(SCR)の UPCL ビットに"1"をセットすると TDRE ビットは"1"になります。 ・送信 FIFO 使用時の TDRE ビットのセット/リセットタイミングは、「40.6.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング」を参照してください。
bit0	TBI : 送信バスアイドルフラグビット	"0"読出し ：　送信中 "1"読出し ：　送信動作なし ・CSIO が送信動作をしていないことを示すビットです。 ・送信データレジスタ(TDR)へデータを書き込んだ場合に本ビットは"0"になります。 ・送信データレジスタ(TDR)がエンプティ(TDRE=1)で、送信動作をしていない場合に本ビットが"1"になります。 ・シリアル制御レジスタ(SCR)の UPCL ビットに"1"をセットすると TDRE ビットは"1"になります。 ・本ビットが"1"で、送信バスアイドル割込み許可(SCR:TBIE=1)されていると送信割込み要求を出力します。

40.4.3.3 拡張通信制御レジスタ ：ESCR (Extended Serial Control Register)

拡張通信制御レジスタのビット構成について説明します。

拡張通信制御レジスタ(ESCR)は、送受信データ長の設定、シリアル出力を"H"固定の設定ができます。

■ ESCR： アドレス Base_addr + 03H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SOP	予約		WT[1:0]		L[2:0]		
初期値	0	－	－	0	0	0	0	0
属性	R0,W	RX,WX	RX,WX	R/W	R/W	R/W	R/W	R/W

ビット名		機　　能
bit7	SOP : シリアル出力端子 セットビット	・シリアル出力端子を"H"にセットするビットです。本ビットに"1"を書いたときに SOUT 端子を"H"にしますが、その後、本ビットに"0"を書く必要はありません。 ・読み出した場合、常に"0"が読み出されます。 (注意事項) シリアルデータ送信中に、本ビットの設定はしないでください。
bit6, bit5	予約	読み出した場合 ： 値は不定です。 書き込みした場合 ： 動作に影響しません。
bit4, bit3	WT1, WT0 : データ送受信 ウェイト選択ビット	マスタ時、連続データの送信または受信に対し、ウェイト数を指定します。スレーブ時は"00"の動作になります。 ・"00"に設定した場合 ： 連続的に SCK が出力されます。 ・"01"に設定した場合 ：1 ビット時間ウェイト後、SCK が出力されます。 ・"10"に設定した場合 ：2 ビット時間ウェイト後、SCK が出力されます。 ・"11"に設定した場合 ：3 ビット時間ウェイト後、SCK が出力されます。
bit2～ bit0	L2, L1, L0 : データ長選択ビット	送受信データのデータ長を指定します。 ・"000_B"に設定した場合 ： データ長は、8 ビットに設定されます。 ・"001_B"に設定した場合 ： データ長は、5 ビットに設定されます。 ・"010_B"に設定した場合 ： データ長は、6 ビットに設定されます。 ・"011_B"に設定した場合 ： データ長は、7 ビットに設定されます。 ・"100_B"に設定した場合 ： データ長は、9 ビットに設定されます。 (注意事項) 上記設定以外は禁止です。

40.4.3.4 受信データレジスタ/送信データレジスタ ：RDR/TDR (Receive Data Register / Transmit Data Register)

受信データレジスタ/送信データレジスタのビット構成について説明します。

受信データと送信データレジスタは同一アドレスに配置されています。読み出した場合は、受信データレジスタとして機能し、書き込みした場合は送信データレジスタとして機能します。FIFO 動作許可の場合、RDR/TDR アドレスは FIFO 読出し、書込みアドレスとなります。

■ RDR/TDR： アドレス Base_addr + 04H (アクセス ： バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							D8
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

リード

受信データレジスタ(RDR)は、シリアルデータ受信用の 9 ビットのデータバッファレジスタです。

- シリアル入力端子(SIN)に送られてきたシリアルデータ信号がシフトレジスタで変換されて、受信データレジスタ(RDR)に格納されます。
- データ長に応じ、以下のように上位ビットに"0"が入ります。

データ長	D8	D7	D6	D5	D4	D3	D2	D1	D0
9 ビット	X	X	X	X	X	X	X	X	X
8 ビット	0	X	X	X	X	X	X	X	X
7 ビット	0	0	X	X	X	X	X	X	X
6 ビット	0	0	0	X	X	X	X	X	X
5 ビット	0	0	0	0	X	X	X	X	X

(X は受信データビット)

- 受信データが、受信データレジスタ(RDR)に格納されると、受信データフルフラグビット(SSR: RDRF)が"1"にセットされます。受信割込みが許可されている場合(SSR:RIE=1)、受信割込み要求が発生します。

- 受信データレジスタ(RDR)は、受信データフルフラグビット(SSR:RDRF)が"1"の状態で読み出してください。受信データフルフラグビット(SSR:RDRF)は、受信データレジスタ(RDR)を読み出すと自動的に"0"にクリアされます。
- 受信エラーが発生(SSR:ORE が"1")した場合、受信データレジスタ(RDR)のデータは無効となります。
- 9 ビット長転送の場合 RDR の読出しは 16 ビットアクセスで行います。

＜注意事項＞

- 受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら SSR:RDRF ビットが"1"にセットされます。
- 受信 FIFO 使用時は、受信 FIFO がエンプティになると SSR:RDRF ビットが"0"にクリアされます。
- 受信 FIFO 使用時に、受信エラーが発生(SSR:ORE が"1")した場合、受信 FIFO の許可ビットはクリアされ、受信データは受信 FIFO に格納しません。

ライト

送信データレジスタ(TDR)は、シリアルデータ送信用の 9 ビットデータバッファレジスタです。

- 送信動作が許可されている場合に(SCR:TXE=1)、送信するデータを送信データレジスタ(TDR)に書き込むと、送信データが送信用シフトレジスタに転送されシリアルデータに変換されて、シリアルデータ出力端子(SOUT)から送出されます。
- データ長に応じ、以下のように上位ビットから順に無効データとなります。

データ長	D8	D7	D6	D5	D4	D3	D2	D1	D0
9 ビット	X	X	X	X	X	X	X	X	X
8 ビット	無効	X	X	X	X	X	X	X	X
7 ビット	無効	無効	X	X	X	X	X	X	X
6 ビット	無効	無効	無効	X	X	X	X	X	X
5 ビット	無効	無効	無効	無効	X	X	X	X	X

(X は送信データビット)

- 送信データエンプティフラグ(SSR:TDRE)は、送信データが送信データレジスタ(TDR)に書き込まれると、"0"クリアされます。
- 送信データエンプティフラグ(SSR:TDRE)は、送信データが送信用シフトレジスタへ転送され、送信が開始されると、送信 FIFO が禁止または送信 FIFO がエンプティの場合、"1"にセットされます。
- 送信データエンプティフラグ(SSR:TDRE)が"1"のとき、送信データを書き込むことができます。送信割込みが許可されている場合には送信割込みが発生します。送信データの書込みは、送信割込みの発生によるか、送信データエンプティフラグ(SSR:TDRE)が"1"の状態で行ってください。
- 送信データエンプティフラグ(SSR:TDRE)が"0"で送信 FIFO が禁止または送信 FIFO がフルのときは、送信データを書き込むことはできません。
- 9 ビット長転送の場合 TDR への書込みは 16 ビットアクセスで行います。

＜注意事項＞

- 送信データレジスタは書込み専用のレジスタで、受信データレジスタは読出し専用のレジスタです。送受信レジスタは同一アドレスに配置されているため書込み値と読出し値が異なります。したがって、INC/DEC 命令などリードモディファイライト(RMW)命令は使用できません。
- 送信 FIFO 使用時の送信データエンプティフラグ(SSR:TDRE)のセットタイミングは、「40.6.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング」を参照してください。

40.4.3.5 ボーレートジェネレータレジスタ ：BGR (Baud rate Generator Register)

ボーレートジェネレータレジスタのビット構成について説明します。

ボーレートジェネレータレジスタ(BGR)は、シリアルクロックの分周比を設定します。

■ BGR ： アドレス Base_addr + 06H (アクセス ： ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	BGR[14:8]						
初期値	0	0	0	0	0	0	0	0
属性	RX,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BGR[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit14～bit0] BGR[14:0] (Baud rate GeneratoR) : **ボーレートジェネレータビット**
ボーレート生成用内部リロードカウンタのリロード値を設定します。
本レジスタにリロード値を書き込むとリロードカウンタはカウントを開始します。

＜注意事項＞

- ボーレートジェネレータレジスタ(BGR)への書込みは、16 ビットアクセスで行ってください。
- リロード値が偶数の場合、シリアルクロックの"H"幅と"L"幅は SCINV ビットの設定によって以下のようになります。奇数の場合、シリアルクロックの"H"幅と"L"幅は同じになります。
  - ☐ SMR:SCINV=0 のとき、シリアルクロックの"H"幅が周辺クロック(PCLK)1 サイクル分長くなります。
  - ☐ SMR:SCINV=1 のとき、シリアルクロックの"L"幅が周辺クロック(PCLK)1 サイクル分長くなります。
- リロード値は 3 以上を設定してください。
- ボーレートジェネレータレジスタ(BGR)の設定値を変更した場合、カウンタ値が"15h00"になってから、新しい設定値がリロードされます。したがって、新しい設定値を即有効にしたい場合は、BGR の設定値を変更した後、CSIO リセット(SCR:UPCL)を実行してください。
- 受信 FIFO 使用時、受信 FIFO アイドル検出許可ビット(FCR1:FRIIE)を"1"に設定しスレーブモードで動作させる場合、BGR にボーレートを設定してください。

40.4.4 LIN-UART 時のレジスタ

LIN-UART 時のレジスタについて示します。

40.4.4.1 シリアル制御レジスタ : SCR (Serial Control Register)

40.4.4.2 シリアルステータスレジスタ : SSR (Serial Status Register)

40.4.4.3 拡張通信制御レジスタ : ESCR (Extended Serial Control Register)

40.4.4.4 受信データレジスタ・送信データレジスタ : RDR/TDR (Receive Data Register / Transmit Data Register)

40.4.4.5 ボーレートジェネレータレジスタ : BGR (Baud rate Generator Register)

40.4.4.1 シリアル制御レジスタ : SCR (Serial Control Register)

シリアル制御レジスタのビット構成について説明します。

シリアル制御レジスタ(SCR)は、送受信割込みの許可/禁止，送信アイドル割込みの許可/禁止，送受信動作の許可/禁止の設定を行います。また、LIN Synch Break 生成, LIN-UART リセットの設定があります。

■ SCR : アドレス Base_addr + 00H (アクセス : バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	UPCL	MS	LBR	RIE	TIE	TBIE	RXE	TXE
初期値	0	0	0	0	0	0	0	0
属性	R0,W	R/W	R0,W	R/W	R/W	R/W	R/W	R/W

ビット名		機能
bit7	UPCL : プログラマブル クリアビット	LIN-UART の内部状態を初期化するビットです。 "1"を設定した場合 : ・LIN-UART を直接リセット(ソフトウェアリセット)します。ただし、レジスタの設定は維持されます。その際、送受信状態のものは直ちに切断されます。 ・ボーレートジェネレータは、BGR レジスタの設定値をリロードし、再スタートします。 ・すべての送受信割込み要因(SSR:TDRE, TBI, RDRF, FRE, ORE, LBD)は初期化されます。 "0"を設定した場合 : 動作に影響ありません。 読出し時は、常に"0"が読み出されます。 (注意事項) ・割込み禁止に設定した後に、プログラマブルクリアを実行してください。 ・FIFO 使用時は、FIFO 禁止(FCR:FE2, FE1=0)にしてからプログラマブルクリアを実行してください。
bit6	MS : マスタ/スレーブ 選択ビット	マスタ/スレーブを選択します。 0 : マスタ 1 : スレーブ

ビット名		機　　能
bit5	LBR : Lin Synch Break 設定ビット (マスタ時のみ機能します)	本ビットに"1"を設定すると ESCR : LBL1/LBL0 ビットおよび、DEL1/DEL0 で設定された長さの LIN Synch Break と LIN Synch デリミタを生成します。 書き込みした場合 ： "0"書込み ： 動作に影響しません。 "1"書込み : LIN Synch Break を生成します。 読み出した場合 ： 常に"0"が読み出されます。 (注意事項) ・マスタ動作のみ機能します。 ・LIN Break field 生成中に本ビットを"1"に設定しないでください。
bit4	RIE : 受信割込み許可 ビット	・CPU への受信割込み要求出力を許可/禁止するビットです。 ・RIE ビットと受信データフラグビット(SSR:RDRF)が"1"の場合、または、エラーフラグビット(SSR:LER, FRE, ORE)のいずれかが"1"の場合、受信割込み要求を出力します。
bit3	TIE : 送信割込み許可 ビット	・CPU への送信割込み要求出力を許可/禁止するビットです。 ・TIE ビットと SSR:TDRE ビットが"1"の場合、送信割込み要求を出力します。
bit2	TBIE : 送信バスアイドル割込み許可ビット	・CPU への送信バスアイドル割込み要求出力を許可/禁止するビットです。 ・TBIE ビットと SSR:TBI ビットが"1"のとき、送信バスアイドル割込み要求を出力します。
bit1	RXE : 受信動作許可 ビット	LIN-UART の受信動作を許可/禁止します。 ・"0"に設定した場合 ： データフレーム受信動作が禁止されます。 ・"1"に設定した場合 ： データフレーム受信動作が許可されます。 (注意事項) ・受信動作許可(RXE=1)にしても、スタートビットの立下りエッジが入力されないと受信動作を開始しません。 ・マスタ動作時、LIN Synch Break 送信中、受信動作が許可(RXE=1)状態でもデータは受信しません。 ・受信中に受信動作を禁止(RXE=0)した場合には、直ちに受信動作を停止します。 ・LIN synch break 検出を行う際には、LIN synch break 検出割込みを許可(ESCR:LBIE=1)に設定した後、受信禁止(SCR:RXE=0)に設定してください。
bit0	TXE : 送信動作許可 ビット	LIN-UART の送信動作を許可/禁止します。 ・"0"に設定した場合 ： データフレーム送信動作が禁止されます。 ・"1"に設定した場合 ： データフレーム送信動作が許可されます。 (注意事項) 送信中に送信動作を禁止(TXE=0)した場合には、直ちに送信動作を停止します。

40.4.4.2 シリアルステータスレジスタ ：SSR (Serial Status Register)

シリアルステータスレジスタのビット構成について説明します。

シリアルステータスレジスタ(SSR)は、送受信状態の確認, 受信エラーフラグの確認, LIN Synch break の検出, また、受信エラーフラグのクリアを行います。

■ SSR ： アドレス Base_addr + 02_H (アクセス ： バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	REC	予約	LBD	FRE	ORE	RDRF	TDRE	TBI
初期値	0	ー	0	0	0	0	1	1
属性	R0,W	RX,WX	R(RM1), W	R,WX	R,WX	R,WX	R,WX	R,WX

ビット名		機　　能
bit7	REC : 受信エラーフラグ クリアビット	シリアルステータスレジスタ(SSR)の FRE, ORE フラグをクリアするビットです。 ・"1"書込みで、エラーフラグがクリアされます。 ・"0"書込みは、動作に影響しません。 読み出した場合、常に"0"が読み出されます。
bit6	予約	読み出した場合 ： 値は不定です。 書き込みした場合 ： 動作に影響しません。
bit5	LBD : LIN Synch Break 検出 フラグビット (スレーブ動作のみ 機能)	"0"読出し : Synch Break なし "1"読出し : Synch Break あり "0"書込み : LBD フラグクリア "1"書込み ： 動作に影響なし LIN Synch Break 検出を示すビットです。 シリアル入力(SIN)が 11 ビット幅以上"0"入力されると、LBD ビットは"1"にセットされます。このとき、LIN Synch Break 割込み許可ビット(LBIE)が"1"にセットされていると、ステータス割込みを発生します。 (読み出した場合) "1"の場合 : LIN Synch Break が検出されています。 "0"の場合 : LIN Synch Break が検出されていません。 (書き込みした場合) "0"を書き込んだ場合 : LBD ビットをクリアします。 "1"を書き込んだ場合 ： 動作に影響しません。 (注意事項) ・本機能は、スレーブ動作のみ機能します。 ・リードモディファイライト系命令時、"1"が読み出されます。

ビット名		機　　能
bit4	FRE : フレーミング エラーフラグ ビット	"0"読出し：フレーミングエラーなし "1"読出し：フレーミングエラーあり ・受信時にフレーミングエラーが発生すると"1"にセットされ、シリアルステータスレジスタ(SSR)の REC ビットに"1"を書き込むとクリアされます。 ・FRE ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・本フラグがセットされた場合は、受信データレジスタ(RDR)のデータは無効です。 ・受信 FIFO 使用時に本フラグがセットされた場合は、受信 FIFO の許可ビットがクリアされ、受信データは受信 FIFO に格納されません。
bit3	ORE : オーバラン エラーフラグ ビット	"0"読出し：オーバランエラーなし "1"読出し：オーバランエラーあり ・受信時にオーバランが発生すると"1"にセットされ、シリアルステータスレジスタ(SSR)の REC ビットに"1"を書き込むとクリアされます。 ・ORE ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・本フラグがセットされた場合は、受信データレジスタ(RDR)のデータは無効です。 ・受信 FIFO 使用時に本フラグがセットされた場合は、受信 FIFO の許可ビットがクリアされ、受信データは受信 FIFO に格納されません。
bit2	RDRF : 受信データ フルフラグビット	"0"読出し：受信データレジスタ(RDR)がエンプティ "1"読出し：受信データレジスタ(RDR)にデータが存在する ・受信データレジスタ(RDR)の状態を示すフラグです。 ・RDR に受信データがロードされると、"1"にセットされ、受信データレジスタ(RDR)を読み出すと"0"にクリアされます。 ・RDRF ビットと SCR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら RDRF が"1"にセットされます。 ・受信 FIFO 使用時は、受信 FIFO がエンプティになると"0"にクリアされます。
bit1	TDRE : 送信データエンプティフラグビット	"0"読出し：送信データレジスタ(TDR)にデータが存在する "1"読出し：送信データレジスタ(TDR)がエンプティ ・送信データレジスタ(TDR)の状態を示すフラグです。 ・TDR に送信データを書き込むと、"0"となり TDR に有効なデータが存在していることを示します。データが送信シフトレジスタにロードされて送信が開始されると"1"になり TDR に有効なデータが存在していないことを示します。 ・TDRE ビットと SCR:TIE ビットが"1"の場合、送信割込み要求を出力します。 ・シリアル制御レジスタ(SCR)の UPCL ビットに"1"をセットすると TDRE ビットは"1"になります。 ・送信 FIFO 使用時の TDRE ビットのセット/リセットタイミングは、「40.7.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング」を参照してください。
bit0	TBI : 送信バスアイドル フラグビット	"0"読出し：　送信中 "1"読出し：　送信動作なし ・LIN-UART が送信動作をしていないことを示すビットです。 ・送信データレジスタ(TDR)へ送信データを書き込んだ場合に本ビットは"0"になります。 ・LIN Break field が設定(SCR:LBR=1)された場合に本ビットは"0"になります。 ・送信データレジスタ(TDR)がエンプティ(TDRE=1)で、送信動作をしていない場合に本ビット が"1"になります。 ・LIN Break field 送信が終了し、送信データレジスタがエンプティの場合に本ビットは"1"になります。 ・本ビットが"1"で、送信バスアイドル割込み許可(SCR:TBIE=1)されていると送信割込み要求を出力します。

40.4.4.3 拡張通信制御レジスタ ：ESCR (Extended Serial Control Register)

拡張通信制御レジスタのビット構成について説明します。

拡張通信制御レジスタ(ESCR)は、LIN Synch Break 割込みの許可/禁止, LIN Synch Break の検出, LIN Synch Break 長, Synch デリミタ長の設定，ストップビット長の選択を行います。

■ ESCR ： アドレス Base_addr + 03H (アクセス ： バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	ESBL	予約	LBIE	LBL[1:0]		DEL[1:0]	
初期値	－	0	－	0	0	0	0	0
属性	R0,W0	R/W	RX,WX	R/W	R/W	R/W	R/W	R/W

ビット名		機　　能
bit7	予約	未定義ビットです。読出し値は"0"です。常に"0"を書き込んでください。
bit6	ESBL : 拡張ストップビット長選択ビット	ストップビット(送信データのフレームエンドマーク)のビット長を設定します。 SMR:SBL=0, ESBL=0 に設定した場合 ： ストップビットは 1 ビットに設定されます。 SMR:SBL=1, ESBL=0 に設定した場合 ： ストップビットは 2 ビットに設定されます。 SMR:SBL=0, ESBL=1 に設定した場合 ： ストップビットは 3 ビットに設定されます。 SMR:SBL=1, ESBL=1 に設定した場合 ： ストップビットは 4 ビットに設定されます。 (注意事項) ・受信時は、常にストップビットの 1 ビット目だけを検出します。 ・本ビットは送信が禁止(TXE=0)のときに設定してください。
bit5	予約	読み出した場合 ： 値は不定です。 書き込みした場合 ： 動作に影響しません。
bit4	LBIE : LIN Synch Break 検出割込み許可ビット	LIN Synch Break 検出割込みを許可/禁止するビットです。 LIN Synch Break 検出フラグ(SSR:LBD)が"1"のとき、割込みが許可(LBIE=1)されると受信割込みを発生します。 (注意事項) ・LIN synch break 検出を行う際には、LIN synch break 検出割込みを 許可(LBIE=1)に設定した後、受信禁止(SCR:RXE=0)に設定してください。
bit3, bit2	LBL[1:0] : LIN synch break 長選択ビット (マスタ動作のみ機能)	00 : 13 ビット長 01 : 14 ビット長 10 : 15 ビット長 11 : 16 ビット長 ・これらのビットは、LIN Synch Break の生成時間を何ビット分とするかを設定します。 ・シリアル制御レジスタ(SCR)の LBR ビットに"1"を設定(LIN Synch Break 送信)する前に、本ビットを設定してください。 ・スレーブ動作時、LIN Synch Break 検出タイミングは、本ビットの設定値によらず、常に 11 ビット目で検出します。 (注意事項) 本機能は、マスタ動作のみ機能します。

ビット名		機　　能
bit1, bit0	DEL[1:0] : LIN synch デリミタ長 選択ビット (マスタ動作のみ機能)	00 : 1 ビット長 01 : 2 ビット長 10 : 3 ビット長 11 : 4 ビット長 ・これらのビットは、LIN Synch デリミタ長を何ビット分とするかを設定します。 ・シリアル制御レジスタ(SCR)の LBR ビットを"1"に設定(LIN Synch Break 送信)する前に、本ビットを設定してください。 (注意事項) 本機能は、マスタ動作のみ機能します。

40.4.4.4 受信データレジスタ・送信データレジスタ ：RDR/TDR (Receive Data Register / Transmit Data Register)

受信データレジスタ・送信データレジスタのビット構成について説明します。

受信データと送信データレジスタは同一アドレスに配置されています。読み出した場合は、受信データレジスタとして機能し、書き込みした場合は送信データレジスタとして機能します。

■ RDR/TDR ： アドレス Base_addr + 04_H (アクセス ： バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	0	0	0	0	0	0	0	0
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

リード

受信データレジスタ(RDR)は、シリアルデータ受信用のデータバッファレジスタです。

- シリアル入力端子(SIN)に送られてきたシリアルデータ信号がシフトレジスタで変換されて、受信データレジスタ(RDR)に格納されます。
- 受信データが、受信データレジスタ(RDR)に格納されると、受信データフルフラグビット(SSR:RDRF)が"1"にセットされます。受信割込みが許可されている場合は(SSR:RIE=1)、受信割込み要求を発生します。
- 受信データレジスタ(RDR)は､受信データフルフラグビット(SSR:RDRF)が"1"の状態で読み出してください。受信データフルフラグビット(SSR:RDRF)は、シリアル受信データレジスタ(RDR)を読み出すと自動的に"0"にクリアされます。
- 受信エラーが発生(SSR:ORE, FRE のいずれかが"1")した場合、受信データレジスタ(RDR)のデータは無効となります。

＜注意事項＞

- 受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら SSR:RDRF が"1"にセットされます。
- 受信 FIFO 使用時は、受信 FIFO がエンプティになると SSR:RDRF が"0"にクリアされます。
- 受信 FIFO 使用時に、受信エラーが発生(SSR:ORE, FRE のいずれかが"1")した場合、受信 FIFO の許可ビットはクリアされ、受信データは受信 FIFO に格納しません。

ライト

送信データレジスタ(TDR)は、シリアルデータ送信用のデータバッファレジスタです。

- 送信動作が許可されている場合に(SCR:TXE=1)、送信するデータを送信データレジスタ(TDR)に書き込むと、送信データが送信用シフトレジスタに転送されシリアルデータに変換されて、シリアルデータ出力端子(SOUT)から送出されます。
- 送信データエンプティフラグ(SSR:TDRE)は、送信データがシリアル送信データレジスタ(TDR)に書き込まれると、"0"にクリアされます。
- 送信データエンプティフラグ(SSR:TDRE)は、送信データが送信用シフトレジスタへ転送され、送信が開始されると、送信 FIFO が禁止または送信 FIFO がエンプティの場合、"1"にセットされます。
- 送信データエンプティフラグ(SSR:TDRE)が"1"の場合は、次の送信用データを書き込むことができます。送信割込みが許可されている場合には送信割込みが発生します。次の送信データの書込みは、送信割込みの発生によるか、送信データエンプティフラグ(SSR:TDRE)が"1"の状態で行ってください。
- 送信データエンプティフラグ(SSR:TDRE)が"0"で送信 FIFO が禁止または送信 FIFO がフルのときは、送信データレジスタ(TDR)に送信データを書き込むことはできません。

＜注意事項＞

- 送信データレジスタは書込み専用のレジスタで、受信データレジスタは読出し専用のレジスタです。2 つのレジスタは同一アドレスに配置されているため書込み値と読出し値が異なります。したがって、INC/DEC 命令などリードモディファイライト(RMW)命令は使用できません。
- 送信 FIFO 使用時の送信データエンプティフラグ(SSR:TDRE)のセットタイミングは、「40.7.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング」を参照してください。

40.4.4.5 ボーレートジェネレータレジスタ ：BGR (Baud rate Generator Register)

ボーレートジェネレータレジスタのビット構成について説明します。

ボーレートジェネレータレジスタ(BGR)は、シリアルクロックの分周比を設定します。また、リロードカウンタのクロックソースとして外部クロックを選択できます。

■ BGR ： アドレス Base_addr + 06H (アクセス ： ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	EXT	BGR[14:8]						
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BGR[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15] EXT (EXTernal clock) : **外部クロック選択ビット**
ボーレート生成用内部リロードカウンタに、内部クロックソースを使用するか、外部クロックソースを使用するかを選択します。EXT=1 のとき、外部クロックソースを使用します。

[bit14-0] BGR[14:0] (Baud rate GeneratoR) : **ボーレートジェネレータビット**
ボーレート生成用内部リロードカウンタのリロード値を設定します。

本レジスタにリロード値を書き込むとリロードカウンタはカウントを開始します。

＜注意事項＞

- ボーレートジェネレータレジスタ(BGR)への書込みは、16 ビットアクセスで行ってください。
- BGR の値が偶数の場合、シリアルクロックの"H"幅は"L"幅より 1 サイクル分短くなります。奇数の場合、デューティ比 1:1 になります。
- ボーレートジェネレータ動作中に外部クロックの設定(EXT=1)に変更する場合、ボーレートジェネレータ(BGR)に"0"を書込み、プログラムクリア(SCR:UPCL)実行後、外部クロック(EXT=1)に設定してください。
- ボーレートジェネレータレジスタ(BGR)の設定値を変更した場合、カウンタ値が"15h00"になってから、新しい設定値がリロードされます。したがって、新しい設定値を即有効にしたい場合は、BGR1/ BGR 0 の設定値を変更した後、プログラマブルクリア(SCR:UPCL)を実行してください。
- BGR へは､3 以上の値を設定してください。ただし、ボーレートの誤差とリロード値の設定によって正常にデータを受信できないことがあります。

40.4.5 I²C 時のレジスタ

I²C 時のレジスタについて示します。

40.4.5.1. I2C バス制御レジスタ : IBCR (I2C Bus Control Register)

40.4.5.2. シリアルステータスレジスタ : SSR (Serial Status Register)

40.4.5.3. I2C バスステータスレジスタ : IBSR (I2C Bus Status Register)

40.4.5.4. 受信データレジスタ/送信データレジスタ : RDR/TDR (Receive Data Register / Transmit Data Register)

40.4.5.5. ボーレートジェネレータレジスタ : BGR (Baud rate Generator Register)

40.4.5.6. 7 ビットスレーブアドレスマスクレジスタ : ISMK (I2C 7-bit Slave address Mask register)

40.4.5.7. 7 ビットスレーブアドレスレジスタ : ISBA (I2C 7-bit Slave Bus address register)

40.4.5.1 I²C バス制御レジスタ ：IBCR (I²C Bus Control Register)

I²C バス制御レジスタのビット構成について説明します。

I²Cバス制御レジスタ(IBCR)は､マスタ/スレーブモード選択，反復スタート条件の発生，アクノ　リッジ許可，割込み許可を設定し、割込みフラグを表示します。

■ IBCR ： アドレス Base_addr + 00H (アクセス ： バイト,ハーフワード,ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MSS	ACT/SCC	ACKE	WSEL	CNDE	INTE	BER	INT
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R,WX	R(RM1),W

ビット名		機　　能
bit7	MSS : マスタ/スレーブ選択ビット	・本ビットに"1"を設定するとI²Cバスがアイドル状態(ISMK: EN=1, IBSR: BB=0)のとき、マスタモードとなります。 ・IBSR レジスタの BB ビットが"1"のとき、このビットに"1"を設定すると BB ビットが"0"になるまでスタート条件の発生をウェイトします。そのウェイト中にスレーブアドレスが一致してスレーブとして動作する場合には本ビットは"0"になり、IBSR レジスタの AL ビットが"1"になります。 ・マスタ動作中(MSS=1, ACT=1)で割込みフラグ(INT)が"1"のとき、本ビットに"0"を書き込むとストップ条件が発生します。 MSS ビットは以下の条件でクリアされます。 (1) I²C インタフェースの動作禁止(ISMK: EN ビット=0) (2) アービトレーションロスト発生時 (3) バスエラー検出(BER ビット=1) (4) INT=1 のとき、MSS ビットへの"0"書込み (5) DMA モードが許可(SSR:DMA=1)で SSR:TBI=1 のとき、MSS ビットへの"0"書込み MSS ビットと ACT ビットの関係を以下に示します。 MSS=0, ACT=0 アイドル MSS=0, ACT=1 スレーブアドレス一致または予約アドレスに対し ACK 応答*し、スレーブ動作中(スレーブモード) MSS=1, ACT=0 マスタ動作待機中 MSS=1, ACT=1 マスタ動作中(マスタモード) * : ACK 応答 ： アクノリッジ区間に I²C バスの SDA が"L"であることを示します。 (注意事項) ・DMA モードが禁止(SSR:DMA=0)で MSS ビットが"1"に設定されていて MSS ビットを"0"に変更する場合、MSS ビット=1, INT ビット=1 のときに行ってください。ACT ビットが"1"のときに MSS ビットに"0"を書込むと INT ビットも"0"にクリアされます。 ・DMA モードが許可(SSR：DMA=1)で MSS ビットが"1"に設定されていて MSS ビットを"0"に変更する場合、MSS ビット=1, INT ビット=1 または、SSR:TBI ビットが"1"のときに行ってください。ACT ビットが"1"のときに MSS ビットに"0"を書き込むと INT ビットも"0"にクリアされます。 ・マスタ動作中、MSS ビットに"0"を書いても ACT ビットが"1"の間、"1"が読み出されます。

ビット名		機　　能
bit6	ACT/SCC : 動作フラグ /反復スタート条件発生ビット	このビットは、読出しと書込みで意味が異なります。 読出し : ACT ビット 書込み : SOC ビット ACT ビットはマスタモードまたは、スレーブモードとして動作していることを示します。 ACT ビットのセット条件 : (1) スタート条件を I²C バスに出力したとき(マスタモード) (2) スレーブアドレスとマスタから送信されたアドレスが一致したとき(スレーブモード) (3) 予約アドレスを検出し、それに対しアクノリッジ応答したとき(MSS=0 のときスレーブモードとなる) ACT ビットのリセット条件 : <マスタモード> (1) ストップ条件検出 (2) アービトレーションロスト検出 (3) バスエラー検出 (4) I²C インタフェースの動作禁止(ISMK:EN ビット=0) <スレーブモード> (1) (反復)スタート条件検出 (2) ストップ条件検出 (3) 予約アドレス検出状態(IBSR:RSA=1)でアクノリッジ応答しなかったとき (4) I²C インタフェースの動作禁止(ISMK:EN ビット=0) (5) バスエラーの発生(BER ビット=1) マスタモード時、このビットに"1"を書き込むと反復スタートを実行します。"0"書込みは無効です。 ＜注意事項＞ ・SCC ビットへの"1"書込みは、マスタモードの割込み中(MSS=1, ACT=1, INT=1)に行ってください。ACT ビットが"1"のときに SCC ビットに"1"を書き込むと INT ビットは"0"にクリアされます。 ・スレーブモード(MSS=0, ACT＝1)時、本ビットに"1"を書くことは禁止です。 ・SCC ビットに"1", MSS ビットに"0"を書いた場合には、MSS ビットが優先されます。 ・リードモディファイライト系命令の読出しでは SCC ビットが読み出されます。 ・8 ビット目のマスタモード割込み時(MSS=1, ACT=1, INT=1, WSEL=1)に SCC ビットへの"1"書込みを行い、9 ビット目に NACK を受信した場合、INT ビットに"1"がセットされ、I²C バスがウェイト(SCL="L")されます。反復スタート条件を発生させるためには、再度 SCC ビットに"1"を書き込み、INT ビットをクリアする必要があります。 ・DMA モードが許可(SSR:DMA=1)時で SSR:TBI ビットが"1"で INT ビットが"0"のときに反復スタート条件を発行する場合は、INT ビットに"1"を書き込んだ後、INT ビットが"1"にセットされていることを確認してから TDR にスレーブアドレスを書き込み、本ビットに"1"を設定してください。

ビット名		機　　能
bit5	ACKE : データバイトアクノリッジ許可ビット	・本ビットに"1"を設定するとアクノリッジタイミングで"L"を出力します。 ・ACT=1 のときに本ビットを変更する場合、INT ビットが"1"のときに行ってください。 ・DMA モードが禁止(SSR:DMA=0)で ACT=1 のときに本ビットを変更する場合、INT ビットが"1"、DMA 割込み許可時(SSM:DMA=1)に SSR:TBI ビットが"1"、または DMA 割込み許可(SSM:DMA=1)でスレーブ受信時に SSM:RDRF が"1"のときに行ってください。 ・DMA モードが許可(SSR:DMA=1)で ACT=1 のときに本ビットを変更する場合、INT ビットが"1"、SSR:TBI ビットが"1"、またはスレーブ受信時に SSM:RDRF が"1"のときに行ってください。 本ビットは以下の条件では無効となります。 (1) 予約アドレス以外のアドレスフィールドに対するアクノリッジ(自動生成) (2) データ送信時(IBSR:RSA=0, IBSR:TRX=1, IBSR:FBT=0) (3) 受信 FIFO 許可でスレーブ受信時(FCR0:FE=1, MSS=0, ACT=1)、常に ACK 応答します。 (4) 受信 FIFO 許可, WSEL が"0", マスタ受信時(FCR0:FE=1, MSS=1, ACT=1, WSEL=0)、SSR:TDRE ビットが"0"のとき ACK 応答し、SSR:TDRE ビットが"1"のとき NACK 応答します。 (5) 受信 FIFO 許可、WSEL=0,予約アドレス検出してスレーブ送信時(IBSR:RSA=1、IBSR:TRX=1, IBSR:FBT=1)、常に ACK 応答します。NACK 応答させる場合、予約アドレス検出後の割込み時、受信 FIFO を禁止にし、ACKE=0 にしてください。 (6) 受信 FIFO 許可, WSEL が"1", マスタ受信で送信データレジスタにデータがあるとき(FCR0:FE=1, MSS=1, ACT=1, WSEL=1, SSR:TDRE=0)
bit4	WSEL : ウェイト選択ビット	・DMA モードが禁止(SSR:DMA=0)時、本ビットはアクノリッジ前か後のどちらに割込み(INT=1)を発生させ、I^2C バスをウェイトさせるか選択します。 ・DMA モードが許可(SSR:DMA=1)時、本ビットはアクノリッジ前か後のどちらに割込み(INT=1、送信時は SSR:TBI=1、受信時は SSR:RDRF=1)を発生させ、I^2C バスをウェイトさせるか選択します。 ・WSEL ビットは以下の条件では無効になります。 (1) 第一バイト*1 に対する割込み発生時(INT=1) (2) 予約アドレス検出時(IBSR:FBT=1, IBSR:RSA=1) (3) FIFO 使用時のデータ転送途中での NACK 応答*2 検出時(FCR0:FE=1, IBSR:RACK=1, ACT=1) (4) 受信 FIFO 使用時、受信 FIFO が FULL になったとき *1 : 第一バイト : (反復)スタート条件後のデータを指します。 *2 : NACK 応答 : アクノリッジ期間 I^2C バスの SDA が"H"であることを指します。
bit3	CNDE : 条件検出割込み許可ビット	マスタモードまたはスレーブモード時(ACT=1)、ストップ条件または反復スタート条件が検出された場合、割込みの発生を許可するビットです。IBSR レジスタの RSC または SPC ビットが"1"で本ビットが"1"のときに割込みが発生します。
bit2	INTE : 割込み許可ビット	マスタモードまたはスレーブモード時、データ送受信およびバスエラーに対する割込み(INT=1)を許可するビットです。

ビット名		機　　能
bit1	BER : バスエラーフラグビット	本ビットは I²C バス上でエラーを検出したことを示します。 BER ビットのセット条件 : (1) 第一バイト*転送中にスタート条件またはストップ条件を検出 (2) 第二バイト以降、データの 2～9 (アクノリッジ)ビット目で(反復)スタート条件またはストップ条件を検出 BER ビットのリセット条件 : (1) BER=1 のときに INT ビットへの"0"書込みした場合 (2) I²C インタフェースの禁止(EN=0)の場合 * : 第一バイト :(反復)スタート条件後のデータを指します。 (注意事項) 割込みフラグ(INT ビット)が"1"になったときにこのビットを確認し、"1"になっていると正常に送受信ができていませんので再送などの処理を行ってください。
bit0	INT : 割込みフラグビット	本ビットはマスタモードまたはスレーブモード時､データ送受信の 8 ビットまたは 9 ビット(ACK)後、またはバスエラー時にこのフラグを"1"にセットします。バスエラー時以外は、INT ビットが"1"になると SCL を"L"にし、INT ビットが"0"になると SCL の"L"の状態を解除します。 INT ビットのセット条件 : < 8 ビット目> <DMA モードに関係ない場合> (1) 第一バイトで予約アドレス検出した場合 (2) WSEL が"1", 第二バイト以降でアービトレーションロストを検出した場合 <DMA モードが禁止の場合(SSR:DMA=0)> (3) DMA モードが禁止時(SSR:DMA=0)、WSEL が"1", マスタ動作中, 第二バイト以降で SSR:TDRE ビットが"1"の場合 (4) DMA モードが禁止時(SSR:DMA=0)、WSEL が"1", スレーブ動作中, 受信 FIFO 禁止, 第二バイト以降で SSR:TDRE ビットが"1"の場合 (5) DMA モードが禁止時(SSR:DMA=0)、WSEL が"1", スレーブ送信中, 第二バイト以降で SSR:TDRE ビットが"1"の場合 (6) DMA モードが禁止時(SSR:DMA=0)、WSEL が"1", 受信 FIFO 禁止でスレーブ受信の場合 <DMA モードが許可の場合(SSR:DMA=1)> (7) DMA モードが許可時(SSR:DMA=1)、WSEL が"1", マスタ動作中, 第二バイト以降で SSR:TBI ビットが"1"のときに INT ビットに"1"を書き込んだ場合 < 9 ビット目> <DMA モードに関係ない場合> (1) 第一バイトでアービトレーションロストを検出した場合 (2) ストップ条件出力設定(マスタ動作中の MSS ビットへの"0"書込み)時以外に NACK を受信した場合 (3) WSEL=0 設定時、第二バイト以降でアービトレーションロストを検出した場合 (4) 第一バイトで予約アドレスを検出せずにマスタモードまたはスレーブモードの受信方向(IBSR:TRX=0)で受信 FIFO 許可時に受信 FIFO にデータがある場合

ビット名		機　　能
bit0	INT : 割込みフラグビット	<DMA モードが禁止の場合(SSR:DMA=0)> (5) DMA モードが禁止時(SSR:DMA=0)、第一バイトで予約アドレスを検出せずにマスタモードまたはスレーブモードの送信方向(IBSR:TRX=1)で SSR:TDRE ビットが"1"の場合 (6) DMA モードが禁止時(SSR:DMA=0)、第一バイトで予約アドレスを検出せずにマスタモードまたはスレーブモードの受信方向(IBSR:TRX=0)で受信 FIFO 禁止時に SSR:TDRE ビットが"1"の場合 (7) DMA モードが禁止時(SSR:DMA=0), WSEL=0 設定時、マスタモード動作中に第二バイト以降で SSR:TDRE ビットが"1"の場合 (8) DMA モードが禁止時(SSR:DMA=0), WSEL=0 設定時、スレーブ送信中に第二バイト以降で SSR:TDRE ビットが"1"の場合 (9) DMA モードが禁止時(SSR:DMA=0), WSEL=0 設定時、受信 FIFO 禁止でスレーブ受信の場合。ただし、予約アドレスを検出した第一バイトでのスレーブ受信では 9 ビット目では割込みは発生しません (10) DMA モードが禁止時(SSR:DMA=0), 受信 FIFO 許可, スレーブ受信のときに受信 FIFO が Full になった場合 <DMA モードが許可の場合(SSR:DMA=1)> (11) DMA モードが許可時(SSR:DMA=1)、第一バイトで予約アドレスを検出せずにスレーブモードの送信方向(IBSR:TRX=1)で SSR:TDRE ビットが"1"の場合 (12) DMA モードが許可時(SSR:DMA=1)、第一バイトで予約アドレスを検出せずにスレーブモードの受信方向(IBSR:TRX=0)で受信 FIFO 禁止時に SSR:TDRE ビットが"1"の場合 (13) DMA モードが許可時(SSR:DMA=1), WSEL=0 設定時、マスタモード動作中に第二バイト以降で SSR:TBI ビットが"1"のときに INT ビットに"1"を書き込んだ場合 <その他> (1) バスエラー検出 INT ビットのリセット条件 : (1) INT ビットへの"0"書込み (2) INT ビットが"1", ACT ビットが"1"のときに MSS ビットへの"0"書込み (3) INT ビットが"1", ACT ビットが"1"のときに SCC ビットへの"1"書込み DMA モードが禁止時(SSR:DMA=0)、INT ビットへの"1"書込みは無効です。 (注意事項) ・DMA モードが許可時(SSR:DMA=1)、マスタモード動作中に第二バイト以降で SSR:TBI ビットが"1"のときに INT ビットに"1"を書き込んだ場合、ステータス割込み(SIRQ＝1)は発生しません。 ・DMA モードが許可(SSR:DMA=1)時で SSR:TBI ビットが"1"で INT ビットが"0"のときに反復スタート条件を発行する場合は、INT ビットに"1"を書き込んだ後、INT ビットが"1"にセットされていることを確認してから TDR にスレーブアドレスを書き込み、SCC ビットに"1"を設定してください。 ・INT フラグが"1"にセットされている場合に、INT フラグに"0"を書き込むと I^2C バスのウェイトを解除します。 ・ISMK:EN ビットを"0"にした場合、受信タイミングによっては SSR:RDRF ビットと INT ビットが"1"になることがあります。この場合、受信データを読み出し、INT ビットをクリアしてください。 ・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。 ・受信 FIFO 許可時、マスタ受信動作で受信 FIFO が Full になっても INT ビットには"1"がセットされません。

40.4.5.2 シリアルステータスレジスタ ：SSR (Serial Status Register)

シリアルステータスレジスタのビット構成について説明します。

シリアルステータスレジスタ(SSR)は、送受信状態の確認を行います。

■ SSR ： アドレス Base_addr + 02H (アクセス ： バイト, ハーフワード,ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	REC	TSET	DMA	TBIE	ORE	RDRF	TDRE	TBI
初期値	0	0	0	0	0	0	1	1
属性	R0,W	R0,W	R/W	R/W	R,WX	R,WX	R,WX	R,WX

ビット名		機　　能
bit7	REC : 受信エラーフラグクリアビット	シリアルステータスレジスタ(SSR)のOREビットをクリアするビットです。 ・"1"書込みで、OREビットがクリアされます。 ・"0"書込みは、動作に影響しません。 読み出した場合、常に"0"が読み出されます。
bit6	TSET : 送信バッファエンプティフラグセットビット	シリアルステータスレジスタ(SSR)のTDREビットをセットするビットです。 ・"1"書込みで、TDREビットおよびDMAモードが許可(DMA=1)のときTBIビットがセットされます。 ・"0"書込みは、動作に影響しません。 読み出した場合、常に"0"が読み出されます。 (注意事項) IBCR:INTビットが"1"のときに本ビットに"1"を書き込んでください。
bit5	DMA: DMAモード許可ビット	DMAモードを禁止/許可するビットです。 ・本ビットを"1"にセットした場合、DMA転送に対応した割込み条件になります。 ・本ビットを"0"にセットした場合、通常転送時に割込み条件になります。 詳細は「表 40-13 I2Cインタフェースの割込み制御ビットと割込み要因」を参照ください。 (注意事項) ISMK:EN=0のときのみ本ビットを変更できます。
bit4	TBIE: 送信バスアイドル割込み許可ビット(DMAモードが許可のみ有効)	・CPUへの送信バスアイドル割込み要求出力を許可/禁止するビットです。 ・DMAモードが許可(DMA=1)でTBIEビットとTBIビットが"1"のとき、送信バスアイドル割込み要求を出力します。 ・DMAモードが禁止(DMA=0)のとき、本ビットは"0"となり、書込みを行っても、その書込みは無視され、"0"の状態を保持します。

ビット名		機　　能
bit3	ORE : オーバランエラー フラグビット	"0"読出し：オーバランエラーなし "1"読出し：オーバランエラーあり ・受信時にオーバランが発生すると"1"にセットされ、シリアルステータスレジスタ(SSR)の REC ビットに"1"を書き込むとクリアされます。 ・ORE ビットと SMR:RIE ビットが"1"の場合、受信割込み要求を出力します。 ・本フラグがセットされた場合、受信データレジスタ(RDR)は無効です。 ・受信 FIFO 使用時、本フラグがセットされた場合には受信データは受信 FIFO に格納されません。
bit2	RDRF : 受信データ フルフラグビット	"0"読出し：受信データレジスタ RDR がエンプティ "1"読出し：受信データレジスタ RDR にデータが存在する ・受信データレジスタ(RDR)の状態を示すフラグです。 ・RIE ビットと受信データフラグビット(RDRF)が"1"の場合、受信割込み要求を出力します。 ・RDR に受信データがロードされると、"1"にセットされ、受信データレジスタ(RDR)を読み出すと"0"にクリアされます。 ・データの 8 ビット目の SCL 立下りタイミングでセットされます。 ・NACK 応答*でもセットされます。 ・受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら RDRF が"1"にセットされます。 ・受信 FIFO 使用時は、受信 FIFO がエンプティになると"0"にクリアされます。 ・受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信せずに受信 FIFO にデータが残っていて受信アイドル状態が受信ボーレートクロックで 8 クロック以上続き、IBCR:BER ビットが"0"のとき RDRF が"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは"0"にリセットされ、再度 8 クロックをカウントします。 * : NACK 応答 ：アクノリッジ期間 I^2C バスの SDA が"H"であることを指します。 (注意事項) ・受信 FIFO 未使用時に DMA モード許可(DMA=1)で受信の場合、RDRF ビットが"1"で WSEL=0 設定時、ACK 送信後に SCL を"L"にし、RDRF ビットが"0"になると SCL が"L"の状態を解除します。 ・受信 FIFO 未使用時に DMA モード許可(DMA=1)で受信の場合、第 2 バイト以降で受信動作中(IBSR:TRX=0)、RDRF ビットが"1"で WSEL=1 設定時、1 バイトデータ受信直後に SCL を"L"にし、RDRF ビットが"0"になると SCL が"L"の状態を解除します。 ・受信 FIFO 使用時に DMA モード許可(DMA=1)で受信の場合、受信 FIFO がフルになると SCL を"L"にし、RDR より 1 回でもデータを読み出すと SCL が"L"の状態を解除します。

ビット名		機　　能
bit1	TDRE : 送信データエンプティフラグビット	"0"読出し ： 送信データレジスタ TDR にデータが存在する "1"読出し ： 送信データレジスタがエンプティ ・送信データレジスタ(TDR)の状態を示すフラグです。 ・TIE ビットと TDRE ビットが"1"の場合、送信割込み要求を出力します。 ・TDR に送信データを書き込むと、"0"となり TDR に有効なデータが存在していることを示します。データが送信シフトレジスタにロードされて送信が開始されると"1"となり TDR に有効なデータが存在していないことを示します。 ・シリアルステータスレジスタ(SSR)の TSET ビットに"1"を書き込むとセットされます。アービトレーションロスト、バスエラーなど検出した場合、TDRE ビットを"1"にセットしたいときに使用します。
bit0	TBI: 送信バスアイドルフラグビット(DMA モード許可のみ有効)	本ビットは DMA モード許可時(DMA=1)に I^2C が送信動作をしていないことを示すビットです。DMA モード許可(DMA=1)で第二バイト以降に TBI ビットが"1"になると、SCL を"L"にし、TBI ビットが"0"になると SCL の"L"の状態を解除します。 TBI ビットのセット条件 ： < 8 ビット目> (1) WSEL が"1", マスタ動作中、第二バイト以降で TDRE ビットが"1"の場合 (2) WSEL が"1", スレーブ送信中、第二バイト以降で SSR:TDRE ビットが"1"の場合 < 9 ビット目> (1) マスタ動作中、第一バイトで予約アドレスを検出せずに SSR:TDRE ビットが"1"の場合 (2) IBCR:WSEL が"0", マスタ動作中、第二バイト以降で SSR:TDRE ビットが"1"の場合 (3) IBCR:WSEL が"0", スレーブ送信中、第二バイト以降で SSR:TDRE ビットが"1"の場合 <その他> 送信バッファエンプティフラグセットビット(TSET)が"1"にセットされている場合 TBI ビットのリセット条件 ： (1) 送信データレジスタ(TDR)へ送信データを書き込んだ場合 本ビットが"1"で、送信バスアイドル割込許可(SCR:TBIE=1)されていると送信割込み要求を出力します。 ・DMA モードが禁止(DMA=0)時に、本ビットは不定となります。

40.4.5.3 I²C バスステータスレジスタ ：IBSR (I²C Bus Status Register)

I²C バスステータスレジスタのビット構成について説明します。

I²Cバスステータスレジスタ(IBSR)は､反復スタート, アクノリッジ, データ方向, アービトレーションロスト, ストップ条件, I²C バス状態, バスエラーを検出したことを示します。

■ IBSR ： アドレス Base_addr + 03H (アクセス ： バイト,ハーフワード,ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FBT	RACK	RSA	TRX	AL	RSC	SPC	BB
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R(RM1), W	R(RM1), W	R,WX

ビット名		機　　能
bit7	FBT : ファーストバイトビット	"0"読出し ： ファーストバイト以外 "1"読出し ： ファーストバイト送受信中 第一バイトを示すビットです。 FBT ビットのセット条件 ： (1) (反復)スタート条件を検出した場合 FBT ビットのクリア条件 ： (1) 2 バイト目の送受信 (2) ストップ条件検出 (3) I²C インタフェースの動作禁止(ISMK:EN ビット=0) (4) バスエラー検出(IBCR:BER ビット=1)
bit6	RACK : アクノリッジフラグビット	"0"読出し : "L"受信 "1"読出し : "H"受信 第一バイト､マスタモード時またはスレーブモード時に受信したアクノリッジをこのビットに示します。 RACK ビットの更新条件 (1) ファーストバイト時のアクノリッジ (2) マスタモードまたはスレーブモード時のデータのアクノリッジ RACK ビットのクリア条件(RACK ビット=0) (1) (反復)スタート条件検出 (2) I²C インタフェースの動作禁止(ISMK:EN ビット=0) (3) バスエラー検出(IBCR:BER ビット=1)

ビット名		機　　能
bit5	RSA : 予約アドレス検出ビット	"0"読出し ： 予約アドレス未検出 "1"読出し ： 予約アドレス検出 本ビットは予約アドレスを検出したことを示すビットです。 RSA ビットのセット条件(RSA=1) (1) 第一バイト目が(0000xxxx)または(1111xxxx)。"x"は"0"または"1"を示します。 RSA ビットのリセット条件(RSA=0) (1) (反復)スタート条件検出 (2) ストップ条件検出 (3) I^2C インタフェースの動作禁止(ISMK:EN ビット=0) (4) バスエラー検出(IBCR:BER ビット=1) 第一バイトで RSA ビットが"1"になるとその第一バイトの 8 ビット目の SCL の立下りで、FIFO 許可、禁止に関係なく割込みフラグ(IBCR:INT)を"1"にして SCL を"L"にします。このとき受信データを読み出し、スレーブとして動作させる場合には IBCR:ACKE を"1"に設定し、割込みフラグ(IBCR:INT)を"0"にクリアします。その後、TRX ビットが"0"であれば、スレーブとしてデータを受信します。途中でデータを受信させない場合には IBCR:ACKE ビットを"0"にします。それ以降、データを受信しません。 (注意事項) ・データ転送中に IBCR:ACKE を"0"にした場合には、ストップ条件または反復スタート条件を検出するまで IBCR:ACKE を"1"にすることは禁止です。 ・予約アドレス検出による割込み時、スレーブ送信を確認した場合、受信 FIFO が許可になっていると ACK 応答しますので受信 FIFO を禁止にし、IBCR:ACKE=0 にしてください。
bit4	TRX : データ方向ビット	"0"読出し ： 受信方向 "1"読出し ： 送信方向 本ビットはデータの方向を示すビットです。 TRX ビットのセット条件 ： (1) マスタモードで(反復)スタート条件を送信 (2) スレーブモードで第一バイトの 8 ビット目が"1"の場合(スレーブとして送信方向) TRX ビットのリセット条件 ： (1) アービトレーションロスト発生(AL=1) (2) スレーブモードでファーストバイトの 8 ビット目が"0"の場合(スレーブとして受信方向) (3) マスタモードでファーストバイトの 8 ビット目が"1"の場合(マスタとして受信方向) (4) ストップ条件検出 (5) マスタモード以外で(反復)スタート条件検出 (6) I^2C インタフェースの動作禁止(ISMK:EN ビット=0) (7) バスエラー検出(IBCR:BER ビット=1)

ビット名		機　　能
bit3	AL : アービトレーションロストビット	"0"読出し：アービトレーションロスト発生なし "1"読出し：アービトレーションロスト発生 本ビットはアービトレーションロストを示します。 AL ビットのセット条件 ： (1) マスタモード時出力しているデータと受信したデータが異なる場合 (2) MSS ビットに"1"を設定したが、スレーブとして動作している場合 (3) マスタモード時、第二バイト目以降のデータの 1 ビット目で反復スタート条件を検出した場合 (4) マスタモード時、第二バイト目以降のデータの 1 ビット目でストップ条件を検出した場合 (5) マスタモード時、反復スタート条件を発生させようとして発生できない場合 (6) マスタモード時、ストップ条件を発生させようとして発生できない場合 AL ビットのリセット条件 ： (1) MSS ビットへの"1"書込み (2) INT ビットへの"0"書込み (3) AL ビット=1, SPC ビット=1 のときに SPC ビットへの"0"書込み (4) I^2C インタフェースの動作禁止(ISMK:EN ビット=0) (5) バスエラー検出(IBCR:BER ビット=1)
bit2	RSC : 反復スタート条件確認ビット	"0"読出し：反復スタート条件未検出 "1"読出し：反復スタート条件検出 マスタモードまたはスレーブモード時に反復スタート条件を検出したことを示すビットです。 RSC ビットのセット条件 ： (1) スレーブモードまたはマスタモードで動作中にアクノリッジ後、反復スタート条件が検出された場合 RSC ビットのリセット条件 ： (1) RSC ビットへの"0"書込み (2) IBCR:MSS ビットへの"1"書込み (3) I^2C インタフェースの動作禁止(ISMK:EN ビット=0) 本ビットへの"1"書込みは無効となります。 (注意事項) ・予約アドレス検出によってスレーブモードとして受信動作中、アクノリッジ応答しなかった場合、スレーブモードを終了しますので次に反復スタート条件を検出しても本ビットに"1"はセットされません。 ・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。

ビット名		機　　能
bit1	SPC : ストップ条件確認ビット	"0"読出し : ストップ条件未検出 "1"読出し : (マスタ)ストップ条件検出またはストップ条件出力時のアービトレーションロスト発生 "1"読出し : (スレーブ)ストップ条件検出 マスタモードまたはスレーブモード時にストップ条件を検出したことを示すビットです。 SPC ビットのセット条件 : (1) スレーブモードまたはマスタモードで動作中にストップ条件が検出された場合 (2) マスタモード時、ストップ条件発生動作でアービトレーションロストが発生した場合 SPC ビットのリセット条件 : (1) 本ビットへの"0"書込み (2) IBCR:MSS ビットへの"1"書込み (3) I²C インタフェースの動作禁止(ISMK:EN ビット=0) 本ビットへの"1"書込みは無効です。 (注意事項) ・予約アドレス検出によってスレーブモードとして受信動作中、アクノリッジ応答しなかった場合、スレーブモードを終了しますので次にストップ条件を検出しても本ビットに"1"はセットされません。 ・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。
bit0	BB : バス状態ビット	"0"読出し : バスアイドル状態 "1"読出し : バス送受信状態 本ビットはバスの状態を示します。 BB ビットのセット条件 : (1) I²C バスの SDA または SCL で"L"を検出した場合 BB ビットのリセット条件 : (1) ストップ条件を検出した場合 (2) I²C インタフェースの動作禁止(ISMK:EN ビット=0) (3) バスエラー検出(IBCR:BER ビット=1)

40.4.5.4 受信データレジスタ/送信データレジスタ ：RDR/TDR (Receive Data Register / Transmit Data Register)

受信データレジスタ/送信データレジスタのビット構成について説明します。

受信データと送信データレジスタは同一アドレスに配置されています。読み出した場合は、受信データレジスタとして機能し、書き込んだ場合は送信データレジスタとして機能します。FIFO 動作許可の場合、RDR/TDR アドレスは FIFO 読出し、書込みアドレスとなります。

■ RDR/TDR ： アドレス Base_addr + 04H (アクセス ： バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	0	0	0	0	0	0	0	0
属性	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

リード

受信データレジスタ(RDR)は、シリアルデータ受信用のデータバッファレジスタです。

- シリアルデータライン(SDA)に送られてきたシリアルデータ信号がシフトレジスタで変換されて、受信データレジスタ(RDR)に格納されます。
- 第一バイト*1 を受信した場合、最下位ビット(RDR:D0)がデータ方向ビットとなります。
- 受信データが、受信データレジスタ(RDR)に格納されると、受信データフルフラグビット(SSR: RDRF)が"1"にセットされます。
- 受信データフルフラグビット(SSR:RDRF)は、受信データレジスタ(RDR)を読み出すと自動的に"0"にクリアされます。

*1 : 第一バイト : (反復)スタート条件後のデータを指します

＜注意事項＞

- 受信 FIFO 使用時は、受信 FIFO に所定のデータ数を受信したら SSR:RDRF が"1"にセットされます。
- 受信 FIFO 使用時は、受信 FIFO がエンプティになると SSR:RDRF が"0"にクリアされます。

ライト

送信データレジスタ(TDR)は、シリアルデータ送信用のデータバッファレジスタです。

- 送信データレジスタ(TDR)の値の MSB ファーストでシリアルデータライン(SDA 端子)に出力します。
- 第一バイトを送信する場合、最下位ビット(TDR:D0)がデータ方向ビットになります。
- 送信データエンプティフラグ(SSR:TDRE)は、送信データが送信データレジスタ(TDR)に書き込まれると、"0"にクリアされます。
- 送信データエンプティフラグ(SSR:TDRE)は、送信用シフトレジスタへ転送されると、"1"にセットされます。
- 次の送信データの書込みは、以下の条件のときに行ってください。
  (1) 割込みフラグ(IBCR:INT ビット)が"1"
  (2) バスエラーが発生していない(IBCR:BER ビット=0)
  (3) アクノリッジが ACK 応答(アクノリッジとして"0"受信)
- 送信 FIFO 禁止時、データエンプティフラグ(SSR:TDRE)が"0"のときは送信データレジスタ(TDR)に送信データを書き込むことはできません。
- 送信 FIFO 使用時、データエンプティフラグ(SSR:TDRE)が"0"であっても送信 FIFO の容量まで送信データを書き込むことが可能です。

＜注意事項＞

送信データレジスタは書込み専用のレジスタで、受信データレジスタは読出し専用のレジスタです。2 つのレジスタは同一アドレスに配置されているため書込み値と読出し値が異なります。したがって、INC/DEC 命令などリードモディファイライト(RMW)動作をする命令は使用できません。

40.4.5.5 ボーレートジェネレータレジスタ ：BGR (Baud rate Generator Register)

ボーレートジェネレータレジスタのビット構成について説明します。

ボーレートジェネレータレジスタ(BGR)は、シリアルクロックの分周比を設定します。

■ BGR ： アドレス Base_addr + 06H (アクセス ： ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	BGR[14:8]						
初期値	ー	0	0	0	0	0	0	0
属性	RX,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BGR[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit14～bit0] BGR[14:0] (Baud rate GeneratoR) : **ボーレートジェネレータビット**
ボーレート生成用内部リロードカウンタのリロード値を設定します。

本レジスタにリロード値を書き込むとリロードカウンタはカウントを開始します。

＜注意事項＞

- ボーレートジェネレータレジスタ(BGR)への書込みは、16 ビットアクセスで行ってください。
- ISMK レジスタの EN ビットが"0"のときにボーレートジェネレータレジスタの設定を行ってください。
- マスタモード, スレーブモードに関係なく、ボーレートを設定してください。
- 動作モード 4 (I^2C モード)では周辺クロック(PCLK)は 8MHz 以上で使用し、400kbps を超えるボーレートジェネレータの設定は禁止です。

40.4.5.6 7ビットスレーブアドレスマスクレジスタ :ISMK (I²C 7-bit Slave address Mask register)

7 ビットスレーブアドレスマスクレジスタのビット構成について説明します。

7 ビットスレーブアドレスマスクレジスタ(ISMK)は、スレーブアドレスの各ビットの比較をするか設定するレジスタです。

■ ISMK : アドレス Base_addr + 08H (アクセス : バイト,ハーフワード,ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EN	SM[6:0]						
初期値	0	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ビット名		機能
bit7	EN : I²C インタフェース許可ビット	I²C インタフェースの動作を許可/禁止するビットです。 "0"に設定した場合 : I²C インタフェースは動作禁止状態になります。 "1"に設定した場合 : I²C インタフェースが動作可能となります。 (注意事項) ・IBSR レジスタの BER ビットが"1"にセットされても、本ビットは"0"にクリアされません。 ・本ビットが"0"のときにボーレートジェネレータを設定してください。 ・本ビットが"0"のときに 7 ビットスレーブアドレスおよび 7 ビットスレーブマスクレジスタを設定してください。 ・I²C インタフェースが禁止される(EN=0)と送受信は直ちに禁止されます。 ・IBCR:MSS ビットに"0"を書き込んでストップコンディションを発生させた後に I²C インタフェースの動作を禁止する場合は、ストップコンディションの発生を確認した後、動作を禁止(EN=0)してください。 ・送信中に EN ビットを"0"にすると I²C バスの SDA/SCL にパルスが発生することがあります。 (注意事項) ・FIFO 許可の場合、FIFO 禁止にして EN ビットに"0"を書いてください。
bit6 ～ bit0	SM6～SM0 : スレーブアドレスマスクビット	7 ビットスレーブアドレスと受信したアドレスに対し、比較対象外にするかどうかを設定するビットです。 "1"を設定したビット : 比較する "0"を設定したビット : 一致したものとして処理する (注意事項) ・EN ビットが"0"のときに本レジスタを設定してください。

40.4.5.7 7 ビットスレーブアドレスレジスタ : ISBA (I²C 7-bit Slave Bus address register)

7 ビットスレーブアドレスレジスタのビット構成について説明します。

7 ビットスレーブアドレスレジスタ(ISBA)は、スレーブアドレスを設定するレジスタです。

■ ISBA : アドレス Base_addr + 09H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SAEN	SA[6:0]						
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

ビット名		機能
bit7	SAEN : スレーブアドレス許可ビット	スレーブアドレスの検出許可ビットです。 "0"を設定した場合 : スレーブアドレスを検出しません。 "1"を設定した場合 : ISBA, ISMK の設定と受信した第一バイトと比較を行います。
bit6～bit0	SA6～SA0 : 7 ビットスレーブアドレス	・7 ビットスレーブアドレスレジスタ(ISBA)は、スレーブアドレス検出許可(SAEN=1)されていれば、(反復)スタート条件検出後に受信した 7 ビットのデータが本レジスタと比較し、全ビットが一致すればスレーブモードとして動作し、ACK を出力します。そのとき、受信したスレーブアドレスは本レジスタにセットされます。(SAEN=0 であれば、ACK を出力しません) ・ISMK レジスタに"0"を設定したアドレスビットは比較対象外となります。 (注意事項) ・予約アドレスの設定は禁止です。 ・本レジスタは ISMK レジスタの EN ビットが"0"のときに設定してください。

40.5 UART の動作説明

UART の動作について示します。

40.5.1 UART の割込み

40.5.2 UART の動作

40.5.3 設定手順とプログラムフロー

40.5.1 UART の割込み

UART の割込みについて示します。

UART には、送受信割込みがあります。次に示す要因で割込み要求を発生させることができます。

- 受信データが受信データレジスタ(RDR)にセットされた場合、または受信エラーが発生した場合
- 送信データが送信データレジスタ(TDR)から送信用シフトレジスタに転送され、送信が開始された場合
- 送信バスアイドル(送信動作なし)
- 送信 FIFO データ要求

40.5.1.1 UART の割込み一覧

UART の割込み一覧について示します。

UART の割込み制御ビットと割込み要因は次のようになっています。

表 40-3 UART の割込み制御ビットと割込み要因

割込みの種類	割込要求フラグビット	フラグレジスタ	動作モード		割込み要因	割込み要因許可ビット	割込み要求フラグのクリア
			0	1			
受信	RDRF	SSR	○	○	1 バイト受信	SCR : RIE	受信データ(RDR)の読出し
					FBYTE 設定値分受信		受信 FIFO がエンプティになるまでの受信データ(RDR)の読出し
					FRIIE ビットが"1"で受信 FIFO に有効なデータが存在した状態で 8 ビット時間以上の受信アイドル状態検出		
	ORE	SSR	○	○	オーバランエラー		受信エラーフラグクリアビット(SSR:REC)への"1"書込み
	FRE	SSR	○	○	フレーミングエラー		
	PE	SSR	○	×	パリティエラー		
送信	TDRE	SSR	○	○	送信レジスタがエンプティ	SCR : TIE	送信データ(TDR)への書込み，または送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送)*
	TBI	SSR	○	○	送信動作なし	SCR : TBIE	送信データ(TDR)への書込み，または送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送)*
	FDRQ	FCR1	○	○	送信 FIFO がエンプティ	FCR1 : FTIE	FIFO 送信データ要求ビット(FCR1:FDRQ)への"0"書込み,または送信 FIFO がフル

* : TDRE ビットが"0"になってから TIE ビットを"1"にしてください。

40.5.1.2 受信割込み発生とフラグセットのタイミング

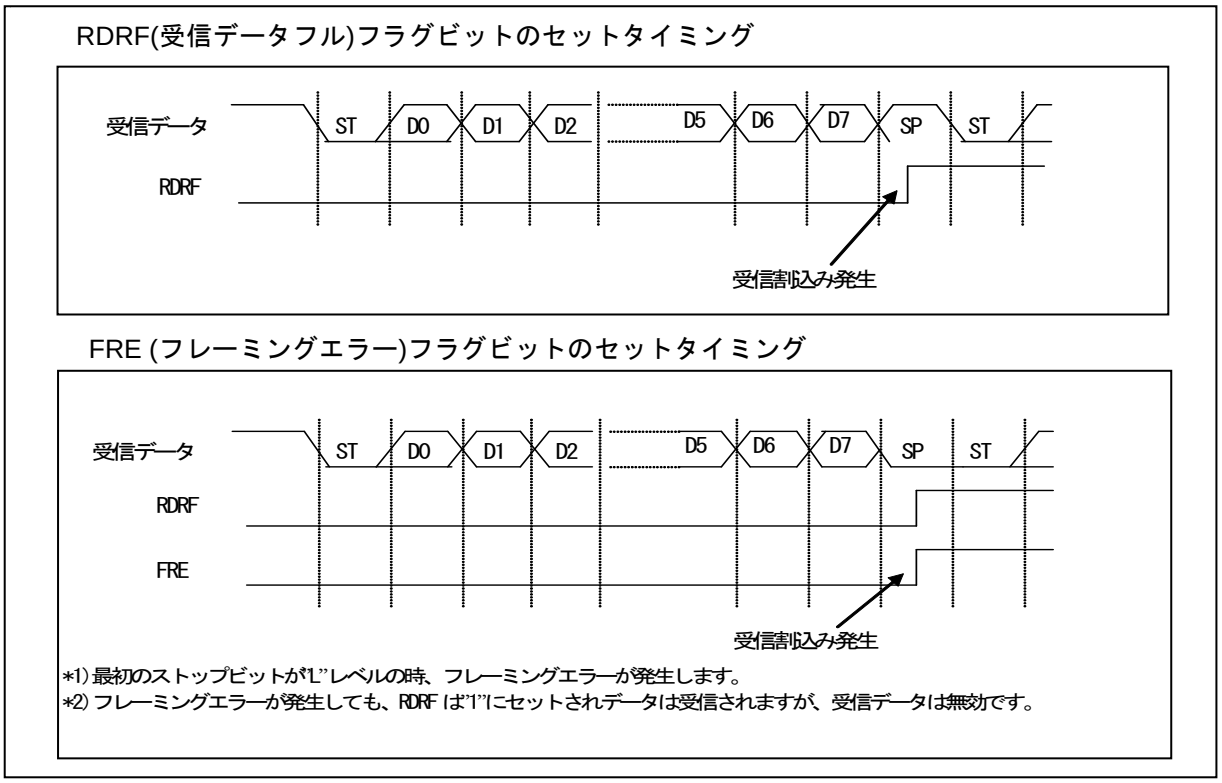
受信割込み発生とフラグセットのタイミングについて説明します。

受信時の割込みとしては、受信完了(SSR:RDRF)、および受信エラーの発生(SSR:PE, ORE, FRE)があります。

最初のストップビットの検出されることにより、受信データが受信データレジスタ(RDR)に格納されます。受信が完了したとき(SSR:RDRF=1)または受信エラーが発生(SSR:PE, ORE, FRE=1)すると各フラグがセットされます。そのとき、受信割込みが許可(SCR:RIE=1)されていると受信割込みが発生します。

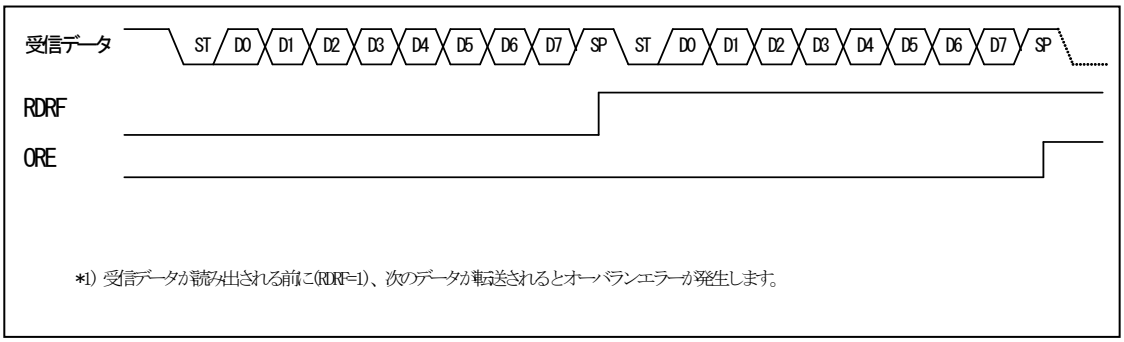
＜注意事項＞
受信エラーが発生した場合は、受信データレジスタ(RDR)のデータは無効となります。

図 40-5 各フラグビットのセットのタイミング



＜注意事項＞
受信時、ストップビットのサンプリングポイントと同時または 1～2 マシンクロック前にシリアルデータの立下りエッジ（ESCR:INV=0 の場合）または立上りエッジ(ESCR:INV=1 の場合）を検出するとそのエッジを無効にしたりして受信できなくなります。連続してフレームを出力する場合にはフレームの間隔を空けることを推奨します。

ORE (オーバランエラー)フラグビットのセットタイミング



*1) 受信データが読み出される前に(RDRF=1)、次のデータが転送されるとオーバランエラーが発生します。

40.5.1.3 受信 FIFO 使用時の割込み発生とフラグセットのタイミング

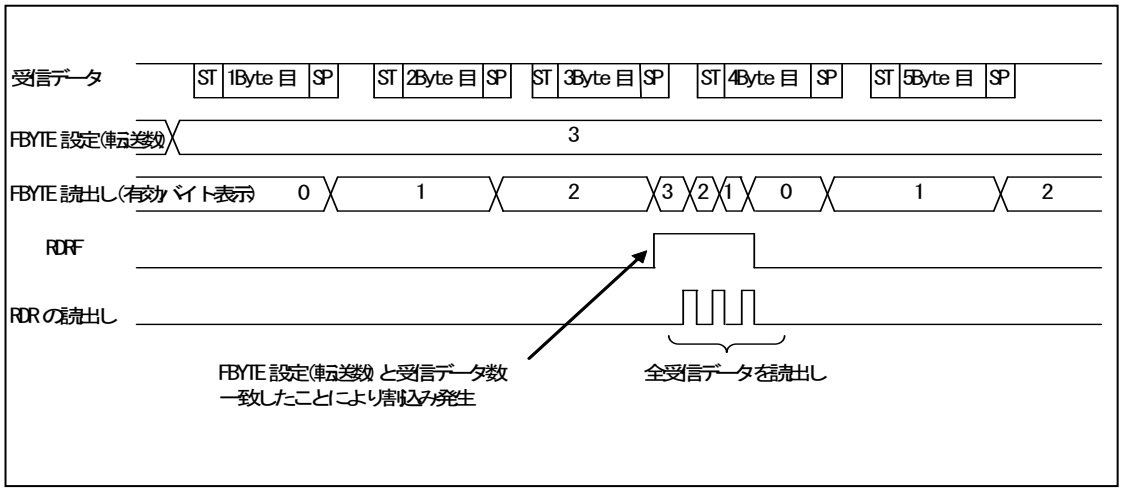
受信 FIFO 使用時の割込み発生とフラグセットのタイミングについて説明します。

受信 FIFO 使用時の割込みは、FBYTE レジスタ(FBYTE)の設定値分受信すると発生します。受信 FIFO 使用時の割込み発生は、FBYTE レジスタの設定値によって決定されます。

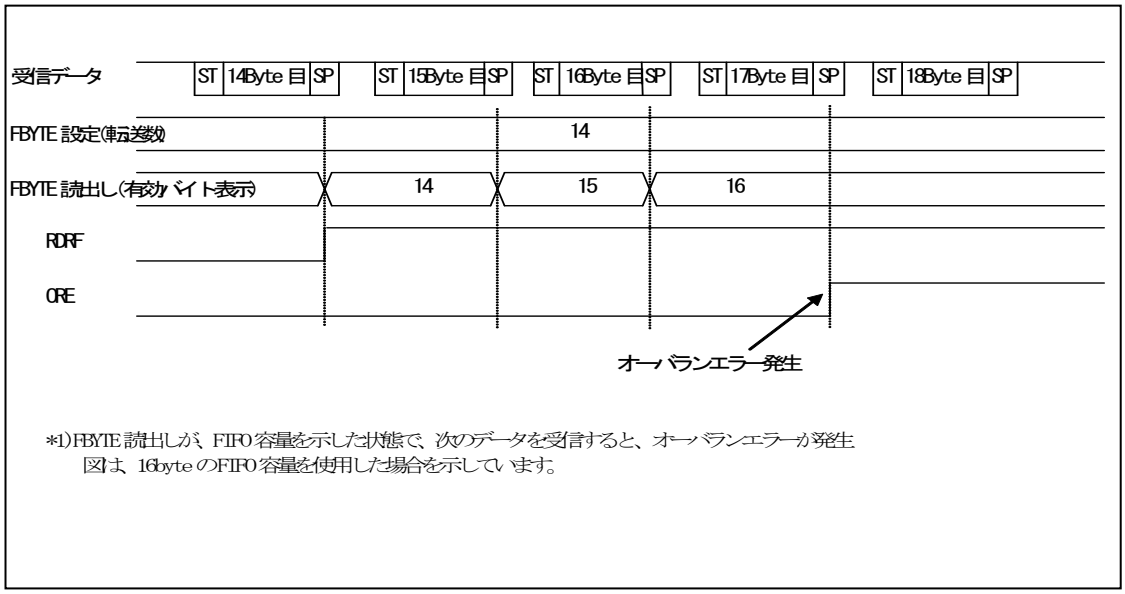
- FBYTE レジスタの転送数設定分のデータを受信すると、シリアルステータスレジスタの受信データフルフラグ(SSR:RDRF)が"1"にセットされます。このとき、受信割込み許可(SCR:RIE)されていると受信割込みを発生します。
- 受信 FIFO アイドル検出許可ビット(FCR1:FRIIE)が"1"で受信 FIFO に存在するデータ数が転送数に達しない場合、受信アイドル状態がボーレートクロックで 8 クロック以上続くと割込みフラグ(SSR:RDRF)が"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは 0 にリセットされ、再度 8 クロックをカウントします。受信 FIFO が禁止されるとそのカウンタは 0 にリセットされます。受信 FIFO にデータが残っている状態で受信 FIFO を許可すると再度、カウントを開始します。
- 受信 FIFO がエンプティになるまで受信データ(RDR)を読み出すと、受信データフルフラグ(SSR:RDRF)はクリアされます。
- 受信有効データ数表示が FIFO 容量を示した状態で、次のデータを受信するとオーバランエラー(SSR:ORE=1)が発生します。

図 40-6 FIFO 使用時のタイミング

受信 FIFO 使用時の受信割込み発生タイミング



ORE (オーバランエラー)フラグビットのセットタイミング



*1) FBYTE 読出しが、FIFO 容量を示した状態で、次のデータを受信すると、オーバランエラーが発生
図は、16byte の FIFO 容量を使用した場合を示しています。

40.5.1.4 送信割込み発生とフラグセットのタイミング

送信割込み発生とフラグセットのタイミングについて説明します。

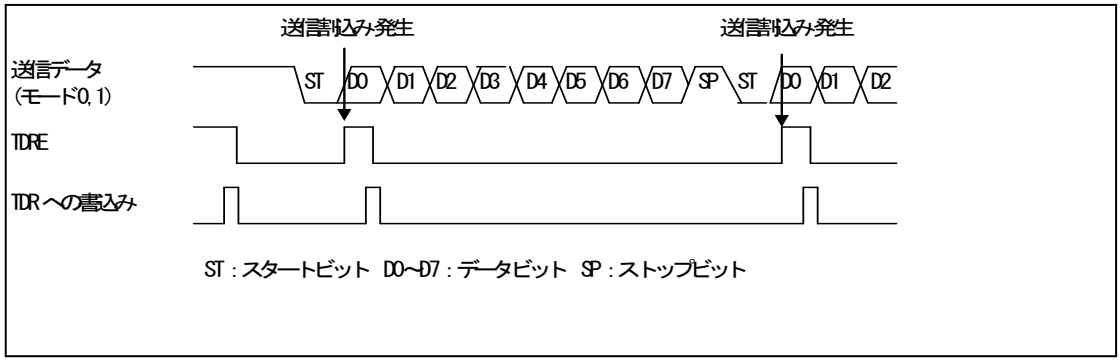
送信時の割込みとしては、送信データが、送信データレジスタ(TDR)から送信用シフトレジスタに転送され(SSR:TDRE=1)送信が開始された場合と、送信動作をしていないとき(SSR:TBI=1)に発生します。

送信データレジスタ(TDR)に書き込まれたデータが送信シフトレジスタに転送されると、次のデータの書込みが可能な状態(SSR:TDRE=1)になります。そのとき、送信割込みが許可(SCR:TIE=1)されていると、送信割込みが発生します。SSR:TDRE ビットは読出し専用ビットなので、送信データレジスタ(TDR)へのデータ書込みにより"0"にクリアされます。

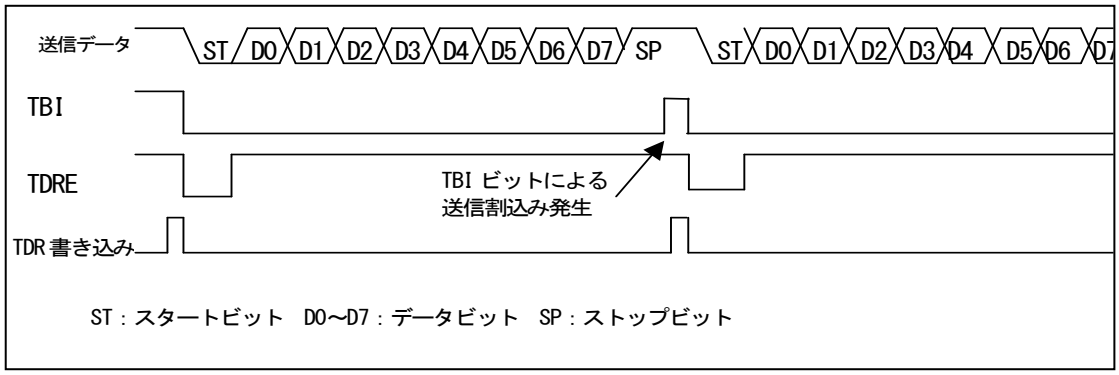
送信データレジスタが空(TDRE=1)で送信動作をしていないとき、SSR:TBI ビットは"1"にセットされます。このとき、送信バスアイドル割込み許可(SCR:TBIE=1)されていると、送信割込みが発生します。送信データレジスタ(TDR)に送信データを書き込むと SSR:TBI ビットおよび送信割込み要求はクリアされます。

図 40-7 送信割込みフラグのタイミング

送信データエンプティフラグ(TDRE) のセットタイミング



送信バスアイドルフラグ(TBI) のセットタイミング



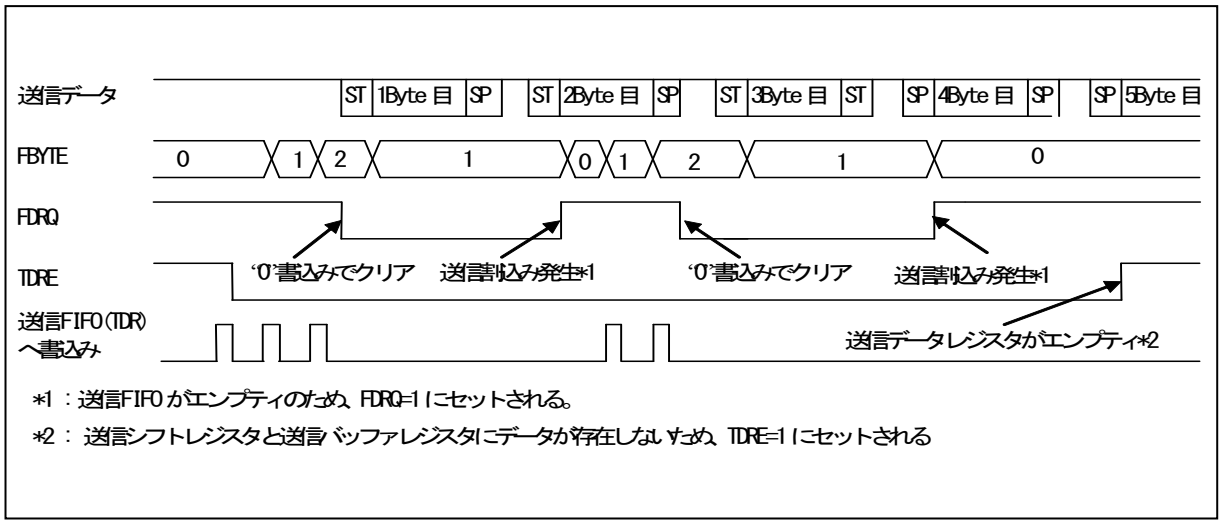
40.5.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング

送信 FIFO 使用時の割込み発生とフラグセットのタイミングについて説明します。

送信 FIFO 使用時の割込みは、送信 FIFO にデータが存在しないときに発生します。

- 送信 FIFO にデータが存在しない場合、FIFO 送信データ要求ビット(FCR1:FDRQ)が"1"にセットされます。このとき、FIFO 送信割込み許可(FCR1:FTIE=1)されていると送信割込みが発生します。
- 送信割込みが発生して送信 FIFO に必要なデータを書き込んだら、FIFO 送信データ要求ビット(FCR1:FDRQ)に"0"書き込んで割込み要求をクリアしてください。
- 送信 FIFO がフルになると FIFO 送信データ要求ビット(FCR1:FDRQ)は"0"になります。
- 送信 FIFO のデータの存在の確認は、FIFO バイトレジスタ(FBYTE)を読み出すことで確認できます。
- FBYTE=00_H のときは、送信 FIFO にデータが存在していないことを示します。

図 40-8 送信 FIFO 使用時の送信割込み発生タイミング



40.5.2 UART の動作

UART の動作について示します。

UART は、モード 0 の双方向シリアル非同期通信、モード 1 のマスタ/スレーブマルチプロセッサ通信で動作します。

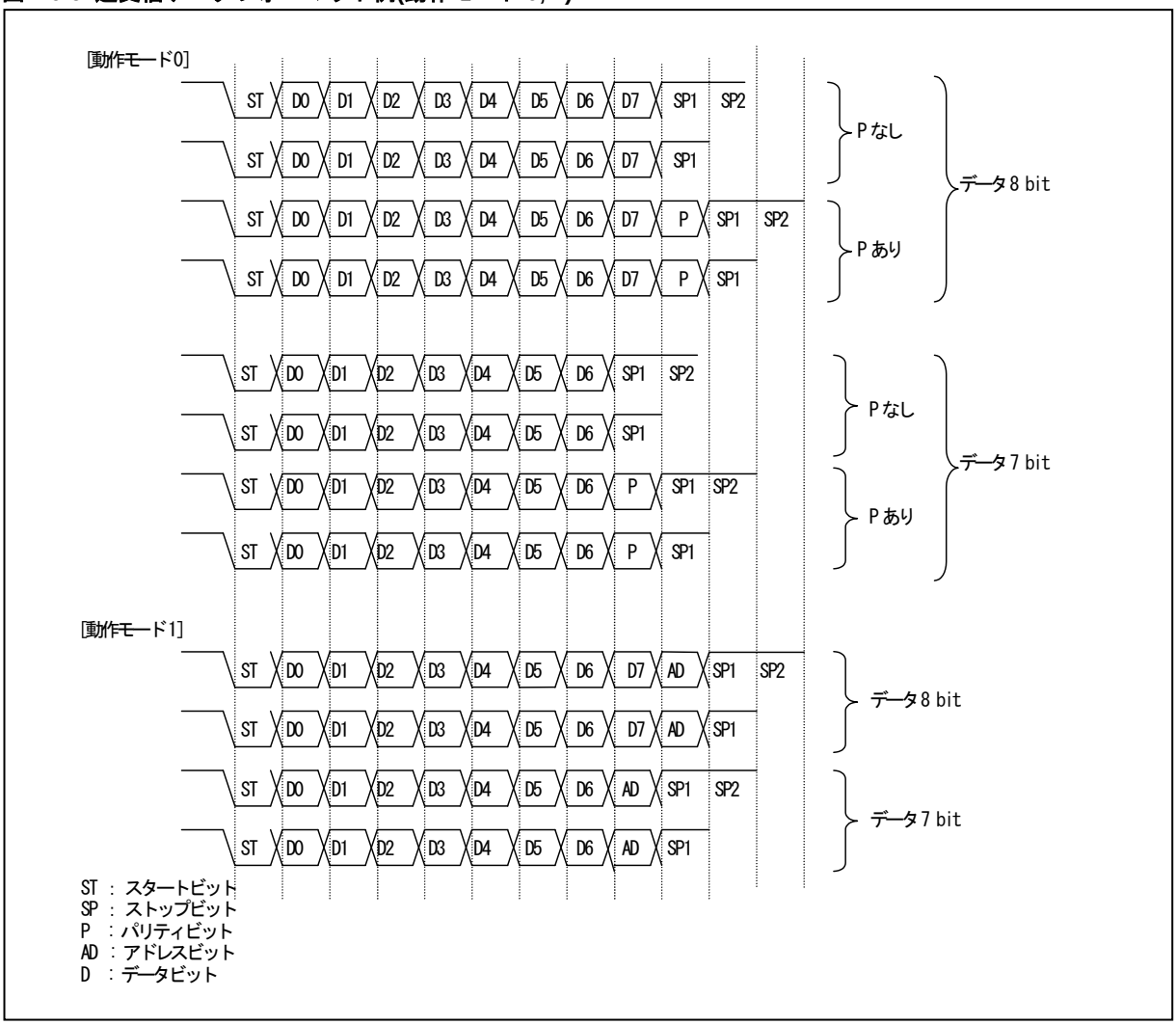
40.5.2.1 送受信データフォーマット

送受信データフォーマットについて説明します。

- 送受信データは、必ずスタートビットから始まり、指定されたデータビット長の送受信が行われ、少なくとも 1 ビットのストップビットで終了します。
- データ転送方向(LSB ファーストまたは MSB ファースト)は、シリアルモードレジスタ(SMR)の BDS ビットで決定されます。パリティありの場合、パリティビットは常に最終データビットと最初のストップビットの間に置かれます。
- 動作モード 0 (通常モード)では、パリティは、あり/なしの選択ができます。
- 動作モード 1 (マルチプロセッサモード)では、パリティは付加されず、AD ビットが付加されます。

動作モード 0, 1 の送受信データフォーマットを次に示します。

図 40-9 送受信データフォーマット例(動作モード 0, 1)



＜注意事項＞
- 図は、データ長 7, 8 ビットに設定した場合を示しています。(データ長は、動作モード 0 の場合、5～9 ビットまで設定できます。)
- シリアルモードレジスタ(SMR)の BDS ビットを"1"(MSB ファースト)に設定した場合、ビットは D7, D6, D5,・・・, D1, D0(P)の順で処理されます。
- データ長が、X ビット長に設定した場合、送受信データレジスタ(RDR/TDR)の下位 X ビットが有効になります。

40.5.2.2 送信動作

送信動作について説明します。

- シリアルステータスレジスタ(SSR)の送信データエンプティフラグビット(TDRE)が"1"であれば、送信データレジスタ(TDR)に送信データを書き込むことができます。(送信 FIFO が許可されている場合には TDRE=0 でも送信データを書くことは可能)
- 送信データを送信データレジスタ(TDR)に書き込むと、送信データエンプティフラグビット(TDRE)は"0"になります。
- シリアル制御レジスタの送信動作許可ビット(SCR:TXE)を"1"に設定すると、送信データは送信シフトレジスタにロードされスタートビットから順に送信が開始されます。
- 送信が開始されると、送信データエンプティフラグビット(SSR:TDRE)は再び"1"にセットされます。このとき、送信割込が許可(SCR:TIE=1)されていれば、送信割込みが発生します。割込み処理において、次の送信データを送信データレジスタに書き込むことができます。

＜注意事項＞
- 送信データエンプティフラグビット(SSR:TDRE)は初期値が"1"のため、送信割込みが許可(SCR:TIE)されると直ちに送信割込みが発生します。
- FIFO 送信データ要求ビット(FCR1:FDRQ)は初期値が"1"のため、FIFO 送信割込みが許可(FCR1: FTIE=1)されると直ちに送信割込みが発生します。

40.5.2.3 受信動作

受信動作について説明します。

- 受信動作が許可(SCR:RXE=1)されると、受信動作を行います。
- スタートビットを検出すると、拡張通信制御レジスタ(ESCR:PEN, P, L2, L1, L0)、およびシリアルモードレジスタ(SMR：BDS)に設定されているデータフォーマットにしたがって 1 フレームデータの受信が行われます。スタートビットの検出条件は、ノイズフィルタ(シリアルデータ入力を 3 回マシンクロックでサンプリングし、多数決)通過後に立下り (ESCR:INV=0 の場合) または立上り(ESCR:INV=1 の場合)を検出し、サンプリングポイントでその通過後のデータが"L"を検出した場合になります。
- 1 フレームの受信が完了すると、受信データフルフラグビット(SSR:RDRF)が"1"にセットされます。このとき、受信割込みが許可(SCR:RIE=1)されている場合、受信割込みが発生します。
- 受信データを読み出す際には、1 フレームデータの受信完了後に受信データを読み出し、シリアルステータスレジスタ(SSR)のエラーフラグの状態を確認してください。受信エラーが発生している場合には、エラー処理を行ってください。
- 受信データの読出しで、受信データフルフラグビット(SSR:RDRF)は"0"にクリアされます。
- 受信 FIFO が許可されている場合、受信 FBYTE に設定された分のフレームを受信すると受信データフルフラグビット(SSR:RDRF)は"1"にセットされます。
- 受信 FIFO アイドル検出許可ビット(FRIIE)が"1"で受信 FIFO に存在するデータ数が転送数に達しない場合、受信アイドル状態がボーレートクロックで 8 クロック以上続くと割込みフラグ(RDRF)が"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは 0 にリセットされ、再度 8 クロックをカウントします。受信 FIFO が禁止されるとそのカウンタは 0 にリセットされます。受信 FIFO にデータが残っている状態で受信 FIFO を許可すると再度、カウントを開始します。
- 受信 FIFO が許可されている場合、シリアルステータスレジスタ(SSR)のエラーフラグが"1"にセットされると受信 FIFO にはそのエラーが発生したデータは受信 FIFO に格納しません。また、そのとき受信データフルフラグビット(SSR:RDRF)を"1"にセットしません。(ただし、オーバランエラーの場合には RDRF フラグは"1"にセットされます。) 受信 FBYTE の表示はエラーが発生する前に正常に受信したデータ数を示しています。シリアルステータスレジスタ(SSR)のエラーフラグが"0"にクリアされないと受信 FIFO は許可されません。
- 受信 FIFO が許可されている場合、受信 FIFO にデータがなくなると受信データフルフラグビット(SSR:RDRF)は"0"にクリアされます。

＜注意事項＞

- 受信データレジスタ(RDR)のデータは、受信データレジスタフルフラグビット(SSR:RDRF)が"1"にセットされ、受信エラーが発生しなかった場合(SSR:PE, ORE, FRE=0)に有効となります。
- ノイズフィルタ(シリアルデータ入力を 3 回マシンクロックでサンプリングして多数決)は内蔵していますが、ノイズがフィルタを通過してしまうと間違ったデータを受信してしまいます。その対策としてノイズが本フィルタを通過しないようにボードを設計するか、ノイズが通過して問題にならない(例えば、最後にデータのチェックサムを付加してエラーが発生した場合には再送を行うなど)ように通信を行ってください。
- 受信時、ストップビットのサンプリングポイントと同時またはサンプリングポイントの 1～2 マシンクロック前にシリアルデータの立下りエッジ (ESCR:INV=0 の場合) または立上りエッジ(ESCR:INV=1 の場合) を検出するとそのエッジを無効などにしたりして受信できなくなります。連続してフレームを出力する場合にはフレームの間隔を空けることを推奨します。

40.5.2.4 クロック選択

クロック選択について説明します。

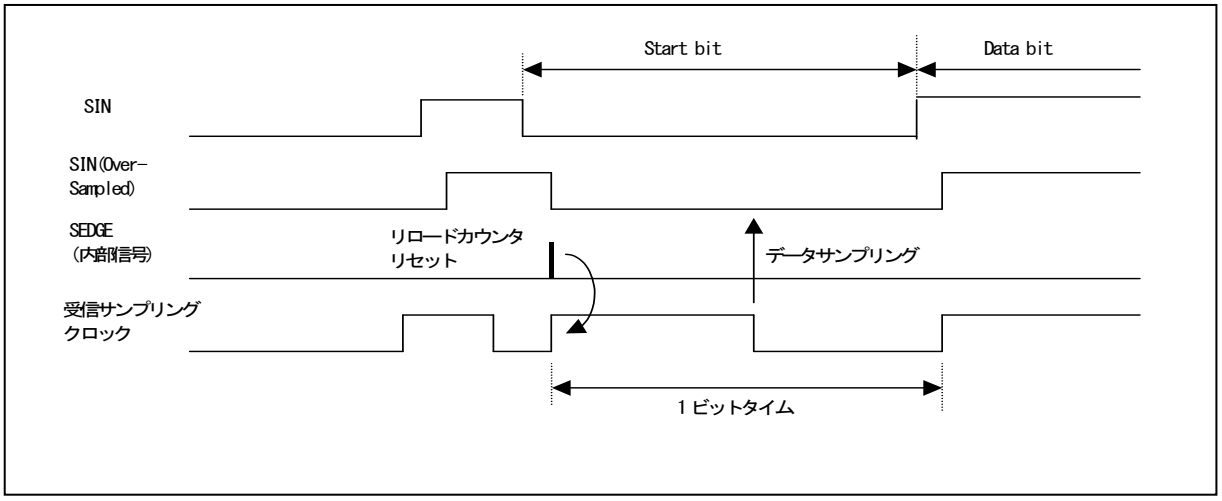
- 内部クロック、または外部クロックを使用できます。
- 外部クロックを使用する場合は、BGR:EXT=1 に設定します。この場合、外部クロックが、ボーレートジェネレータで分周されます。

40.5.2.5 スタートビット検出

スタートビット検出について説明します。

- 非同期モード時は、SIN 信号の立下りエッジによってスタートビットを認識します。このため受信動作許可(SCR:RXE=1)しても、SIN 信号の立下りエッジが入力されないと、受信動作を開始しません。
- スタートビットの立下りエッジを検出すると、ボーレートジェネレータの受信リロードカウンタはリセットされ、再リロードしカウントダウンを開始します。これによって、常にデータの中心でサンプリングします。

図 40-10 スタートビット検出



40.5.2.6 ストップビット

ストップビットについて説明します。

- 1 ビットから 4 ビット長を選択できます。
- 受信データフルフラグビット(SSR:RDRF)は、最初のストップビットを検出すると"1"にセットされます。

40.5.2.7 エラー検出

エラー検出について説明します。

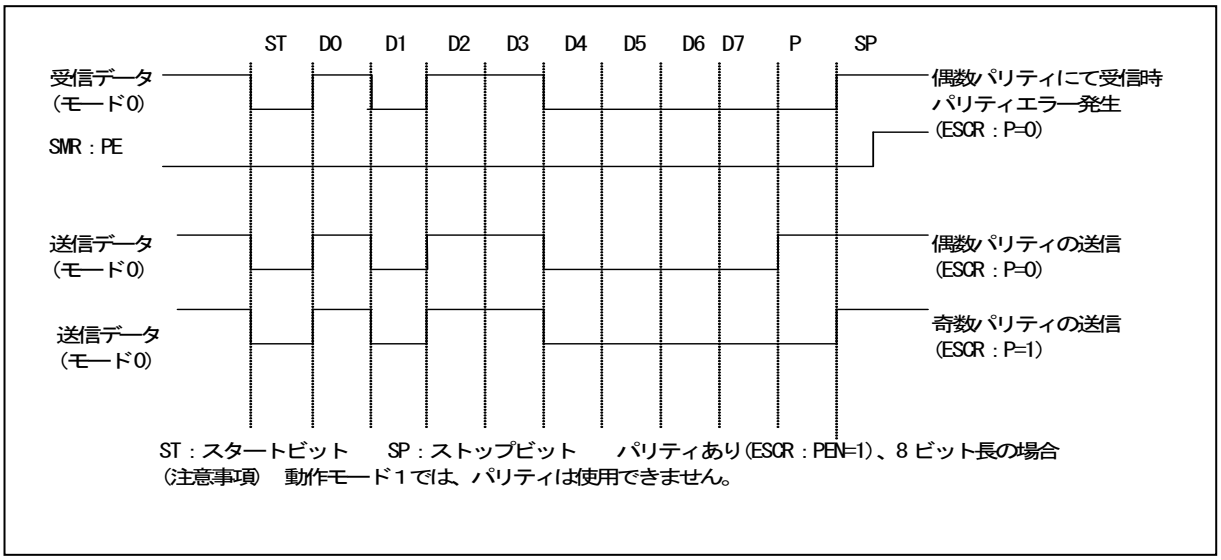
- 動作モード 0 では、パリティエラー, オーバランエラー, フレームエラーが検出できます。
- 動作モード 1 では、オーバランエラー, フレームエラーが検出できます。パリティエラーは検出できません。

40.5.2.8 パリティビット

パリティビットについて説明します。

- パリティビットの付加は、動作モード 0 の場合のみ設定できます。パリティ許可ビット(ESCR: PEN)でパリティの有無を、パリティ選択ビット(ESCR:P)で偶数パリティ/奇数パリティを設定できます。
- 動作モード 1 では、パリティを使用できません。

図 40-11 パリティ有効時の動作

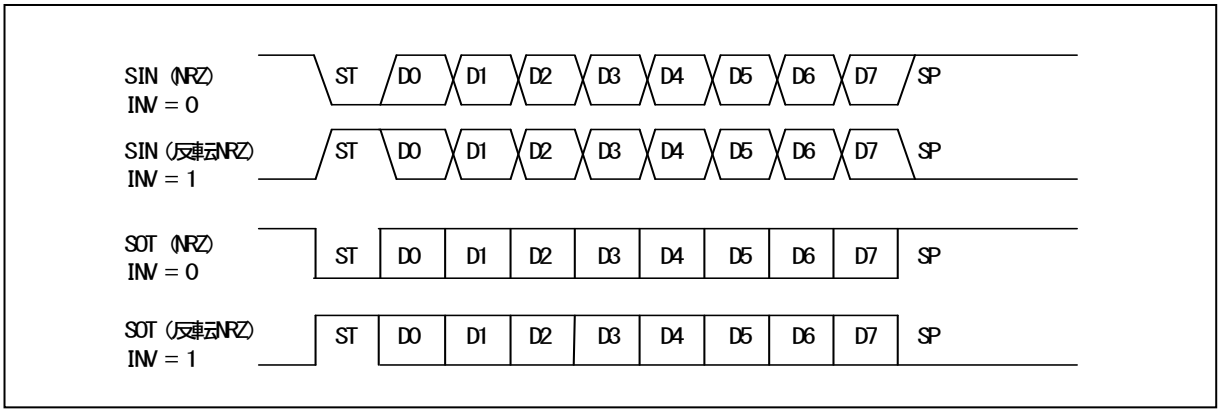


40.5.2.9 データ信号方式

データ信号方式について説明します。

- 拡張通信制御レジスタの INV ビットの設定によって、NRZ (Non Return to Zero)信号方式(ESCR: INV=0)、または反転 NRZ 信号方式(ESCR:INV=1)を選択できます。

図 40-12 NRZ (Non Return to zero)信号方式, および反転 NRZ 信号方式



40.5.2.10 データ転送方式

データ転送方式について示します。

データビット転送方法を LSB ファーストまたは MSB ファーストを選択できます。

40.5.2.11 UART ボーレート選択・設定

UART ボーレート選択・設定について説明します。

UART の送受信ボーレートジェネレータは、次のいずれかを選択できます。

- 専用ボーレートジェネレータ(リロードカウンタ)で内部クロックを分周して得られるボーレート
  2 つの内部リロードカウンタがあり、それぞれ送受信シリアルクロックに対応しています。ボーレートジェネレータレジスタ(BGR)で 15 ビットのリロード値を設定することにより、ボーレートを選択できます。

  リロードカウンタは、設定された値で内部クロックを分周します。

  クロックソースの設定は、内部クロックを選択(BGR:EXT=0)してください。

- 専用ボーレートジェネレータ(リロードカウンタ)で外部クロックを分周して得られるボーレート
  リロードカウンタのクロックソースに外部クロックを使用します。ボーレートジェネレータレジスタ(BGR)で 15 ビットのリロード値を設定することにより、ボーレートを選択できます。リロードカウンタは、設定された値で外部クロックを分周します。クロックソースの設定は、外部クロックとボーレートジェネレータクロック使用を選択(BGR:EXT=1)してください。

  本モードは特殊な周波数の発振子を分周して使用するケースを想定して用意されています。

＜注意事項＞
- 外部クロックの設定(EXT=1)は、リロードカウンタが停止した状態(BGR=15'h00)で行ってください。
- 外部クロックに設定(EXT=1)した場合、外部クロックの"H"幅, "L"幅は 2 周辺クロック(PCLK)以上必要です。

■ ボーレートの計算

2 つの 15 ビットリロードカウンタは、ボーレートジェネレータレジスタ(BGR)で設定します。

ボーレートの計算式を以下に示します。

(1)リロード値

$$V = \Phi / b - 1$$

V : リロード値
b : ボーレート
Φ : 内部クロック(周辺クロック(PCLK)), または外部クロック周波数

(2)計算例

内部クロック(周辺クロック(PCLK))16MHz, 内部クロック使用, ボーレート 19200 bps に設定する際のリロード値を計算します。

$$V = (16 \times 1{,}000{,}000) / 19200 - 1$$
$$= 832$$

このリロード値を用いた場合のボーレートは次のようになります。

$$b = (16 \times 1{,}000{,}000) / (832 + 1)$$
$$= 19208 \text{ bps}$$

(3)ボーレートの誤差

ボーレートの誤差は次の式によって求められます。

誤差(%) ＝ (計算値 － 目標値) / 目標値 × 100

(例)内部クロック 20MHz, 目標ボーレート 153600 bps に設定する場合

リロード値	＝ (20 × 1,000,000) / 153600 － 1	＝ 129
ボーレート(計算値)	＝ (20 × 1,000,000) / (129 + 1)	＝ 153846 bps
誤差(%)	＝ (153846 － 153600) / 153600 × 100	＝ 0.16(%)

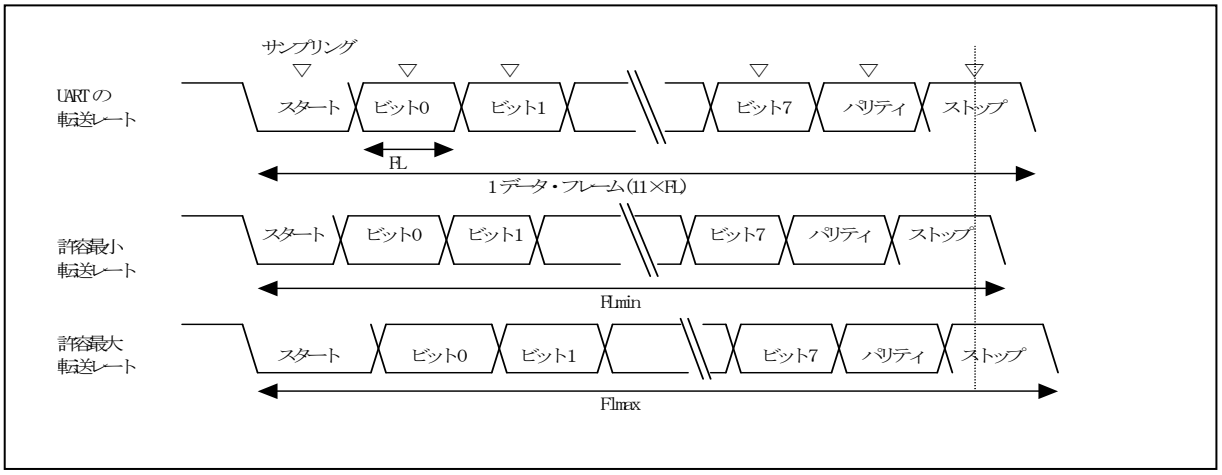
＜注意事項＞

- リロード値を"0"に設定するとリロードカウンタは停止します。
- リロード値が偶数の場合、受信シリアルクロックの"H"幅と"L"幅は"L"幅のほうが周辺クロック(PCLK)1 サイクル分長くなります。奇数の場合、シリアルクロックの"H"幅と"L"幅は同じになります。
- リロード値は 4 以上を設定してください。ただし、ボーレートの誤差とリロード値の設定によって正常にデータを受信できないことがあります。

■ 受信時のボーレートの許容誤差範囲

受信の際に、送信先のボーレートのずれがどの程度まで許容できるかを説明します。受信時のボーレート誤差は、下記に示す算出式を使用して、必ず許容誤差範囲内になるように設定してください。

図 40-13 受信時の許容ボーレート範囲



図に示すように、スタートビット検出後は BGR レジスタで設定したカウンタにより、受信データのサンプリング・タイミングが決定されます。このサンプリング・タイミングに最終データ(ストップビット)までが間に合えば正常に受信できます。これを 11 ビット受信に当てはめると理論上、次のようになります。

サンプリング・タイミングのマージンを周辺クロック(PCLK)(Φ)の 1 クロック分とすると、許容最小転送レート(FLmin)は次のようになります。

FLmin = (11bit×(V+1) － (V+1) / 2 +2) / Φ = (21V+25) / 2 Φ (s)

V : リロード値　Φ : 内部クロック(周辺クロック(PCLK))(Hz)

したがって、受信可能な送信先の最大ボーレート(BGmax)は次のようになります。

BGmax = 11/FLmin = 22 Φ / (21V+25)　(bps)

V : リロード値　Φ : 内部クロック(周辺クロック(PCLK))(Hz)

許容最大転送レート(FLmax)データを受信する場合、11 ビット目の受信データの始点にてサンプリングが行われます。よって、許容最大転送レート(FLmax)は次のようになります。

10/11×FLmax = (11bit×(V+1) – (V+1)/2)/Φ

V :リロード値　Φ : 内部クロック(周辺クロック(PCLK))(Hz)

FLmax = (21/20×11×(V+1))/Φ

サンプリング・タイミングのマージン(Φ)を 2 クロック分とすると、許容最大転送レート(FLmax)は次のようになります。

FLmax = (21/20×11×　(V+1)　-　2)/ Φ= (231V+191)/20Φ(s)

V :リロード値　Φ : 内部クロック(周辺クロック(PCLK))(Hz)

したがって、受信可能な送信先の最小ボーレート(BGmin)は次のようになります。

BGmin = 11 / FLmax = 220Φ / (231V+191) (bps)

V :リロード値　Φ : 内部クロック(周辺クロック(PCLK))(Hz)

前述の最小/最大ボーレート値の算出式から、UART と送信先とのボーレートの許容誤差を求めると次のようになります。

表 40-4 許容ボーレート誤差

リロード値	許容最大ボーレート誤差	許容最小ボーレート誤差
3	0%	0%
10	2.98%	-3.24%
50	4.37%	-4.44%
100	4.56%	-4.60%
200	4.66%	-4.68%
32767	4.76%	-4.76%

＜注意事項＞
受信の精度は、1 フレームのビット数, 内部クロック(周辺クロック(PCLK)), リロード値に依存します。内部クロックが高く、分周比が高くなるほど精度は高くなります。

■ 各内部クロック(周辺クロック(PCLK))・ボーレートに対するリロード値と誤差

表 40-5 各内部クロック(周辺クロック(PCLK))・ボーレートに対するリロード値と誤差

ボーレート (bps)	8 MHz		10 MHz		16 MHz		20 MHz		24 MHz		32MHz	
	Value	ERR	Value	ERR	Value	ERR	Value	ERR	Value	ERR	Value	ERR
4M	―	―	―	―	-	0	4	0	5	0	7	0
2.5M	―	―	-	0	―	―	―	―	―	―	―	―
2M	-	0	4	0	7	0	9	0	11	0	15	0
1M	7	0	9	0	15	0	19	0	23	0	31	0
500000	15	0	19	0	31	0	39	0	47	0	63	0
460800	―	―	―	―	―	―	―	―	51	−0.16	―	―
250000	31	0	39	0	63	0	79	0	95	0	127	0
230400	―	―	―	―	―	―	―	―	103	−0.16	―	―
153600	51	−0.16	64	−0.16	103	−0.16	129	−0.16	155	−0.16	207	−0.16
125000	63	0	79	0	127	0	159	0	191	0	255	0
115200	68	−0.64	86	0.22	138	0.08	173	0.22	207	−0.16	277	0.08
76800	103	−0.16	129	−0.16	207	−0.16	259	−0.16	311	−0.16	416	0.08
57600	138	0.08	173	0.22	277	0.08	346	−0.16	416	0.08	555	0.08
38400	207	−0.16	259	−0.16	416	0.08	520	0.03	624	0	832	−0.04
28800	277	0.08	346	＜0.01	554	−0.01	693	−0.06	832	−0.03	1110	−0.01
19200	416	0.08	520	0.03	832	−0.03	1041	0.03	1249	0	1666	0.02
10417	767	＜0.01	959	＜0.01	1535	＜0.01	1919	＜0.01	2303	＜0.01	3071	＜0.01
9600	832	0.04	1041	0.03	1666	0.02	2083	0.03	2499	0	3332	−0.01
7200	1110	＜0.01	1388	＜0.01	2221	＜0.01	2777	＜0.01	3332	＜0.01	4443	−0.01
4800	1666	0.02	2082	−0.02	3332	＜0.01	4166	＜0.01	4999	0	6666	＜0.01
2400	3332	＜0.01	4166	＜0.01	6666	＜0.01	8332	＜0.01	9999	0	13332	＜−0.01
1200	6666	＜0.01	8334	0.02	13332	＜0.01	16666	＜0.01	19999	0	26666	＜0.01
600	13332	＜0.01	16666	＜0.01	26666	＜0.01	―	―	―	―	―	―
300	26666	26666	＜0.01	―	―	―	―	―	―	―	―	―

・ Value : BGR レジスタの設定値 (10 進)
・ ERR : ボーレート誤差(%)

■ 外部クロック

ボーレートジェネレータレジスタ(BGR)のEXTビットに"1"を書き込むと、ボーレートジェネレータで外部クロックを分周します。

＜注意事項＞

外部クロック信号はUARTで内部クロックに同期します。したがって、同期化不可能な外部クロックの場合、動作が不安定になります。

リロードカウンタの機能

リロードカウンタには、送信リロードカウンタと受信リロードカウンタがあり、専用ボーレートジェネレータとして機能します。リロード値に対する15ビットレジスタから構成されており、外部クロックまたは内部クロックより送受信クロックを生成します。

カウントの開始

ボーレートジェネレータレジスタ(BGR)にリロード値を書き込むと、リロードカウンタはカウントを開始します。

再スタート

リロードカウンタは下記の条件で再スタートします。

- 送信/受信リロードカウンタ共通
  プログラマブルクリアビット(SCR:UPCL)に"1"を設定
- 受信リロードカウンタ
  非同期モードでのスタートビット立下りエッジ検出

40.5.3　設定手順とプログラムフロー

設定手順とプログラムフローについて示します。

40.5.3.1 動作モード 0 (1:1 接続)

40.5.3.2 動作モード 1 (1 : n 接続)

40.5.3.1　動作モード 0 (1:1 接続)

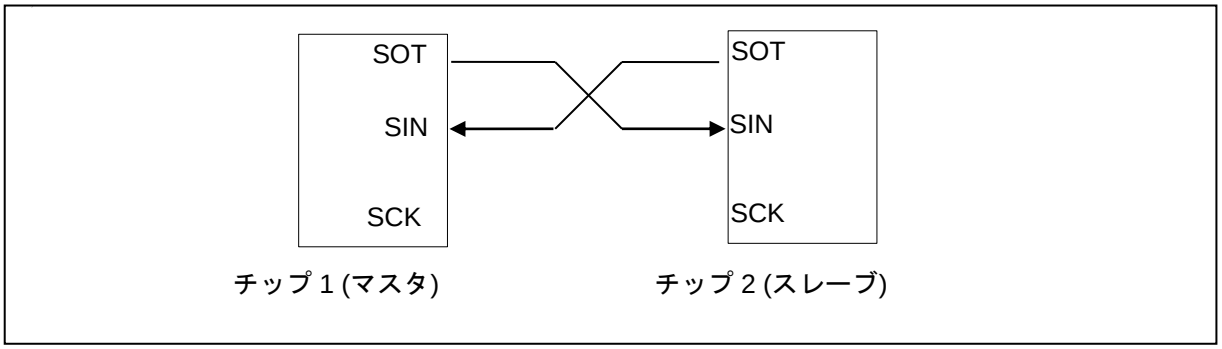
動作モード 0 (1:1 接続)について示します。

動作モード 0 では、非同期シリアル双方向通信をすることができます。

■ チップ間接続

動作モード 0 (通常モード)では、双方向通信を選択します。次に示すように 2 つの CPU を相互に接続します。

図 40-14 UART 動作モード 0 の双方向通信の接続例



■ フローチャート

図 40-15 双方向通信設定例(FIFO 未使用時)

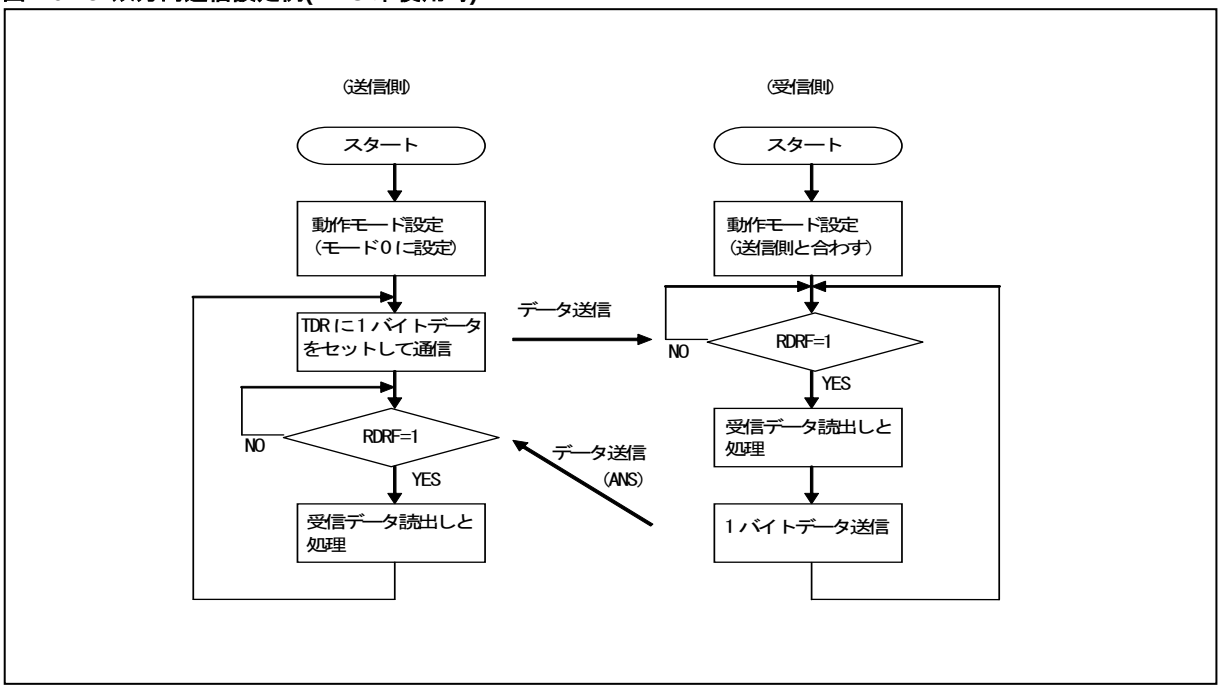
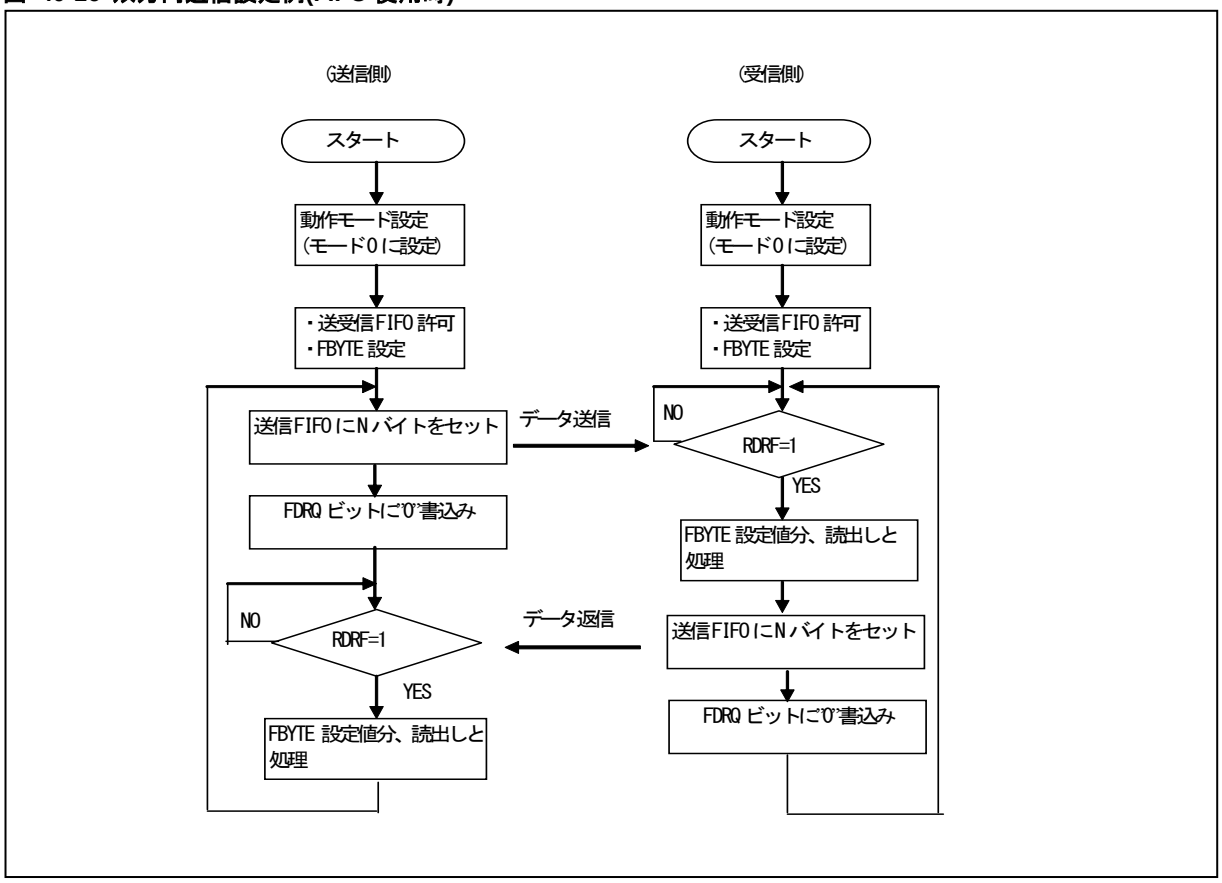


図 40-16 双方向通信設定例(FIFO 使用時)



40.5.3.2 動作モード 1 (1 : n 接続)

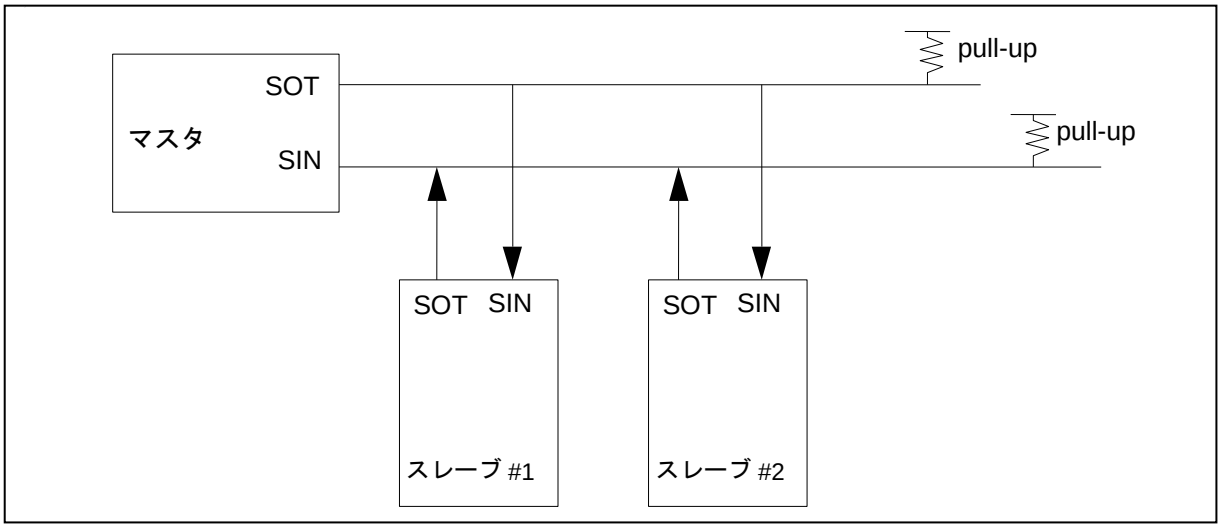
動作モード 1 (1 : n 接続)について説明します。

動作モード 1 (マルチプロセッサモード)では、複数 CPU のマスタ/スレーブ接続による通信が可能です。マスタ/スレーブとして使用できます。

■ チップ間接続

マスタ/スレーブ型通信では､図に示すように 2 本の共通通信ラインに 1 つのマスタ CPU と複数のスレーブ CPU を接続して通信システムを構成します。UART はマスタまたはスレーブのどちらでも使用できます。

図 40-17 UART のマスタ/スレーブ型通信の接続例



■ 機能選択

マスタ/スレーブ型通信では、次に示すように動作モードとデータ転送方式を選択してください。

表 40-6 マスタ/スレーブ型通信機能の選択

	動作モード		データ	パリティ	ストップビット	ビット方向
アドレス送受信	モード 1 (AD ビット送信)	モード 1 (AD ビット受信)	AD = 1 ＋ 7 または 8 ビットアドレス	なし	1 ビット または 2 ビット	LSB または, MSB ファースト
データ送受信			AD = 0 ＋ 7 または 8 ビットデータ			

＜注意事項＞

動作モード 1 では送受信データ(TDR/RDR)はワードアクセスで行ってください。

■ 通信手順

通信は、マスタ CPU がアドレスデータを送信することによって始まります。アドレスデータとは D8 ビットを "1"としたデータで、通信先となるスレーブ CPU を選択します。各スレーブ CPU はプログラムでアドレスデータを判断し、割り当てられたアドレスと一致した場合にマスタ CPU との通信(通常データ)をします。次に、マスタ/スレーブ型通信(マルチプロセッサモード)のフローチャートを示します。

図 40-18 マスタ/スレーブ型通信フローチャートの例 (FIFO 未使用時)

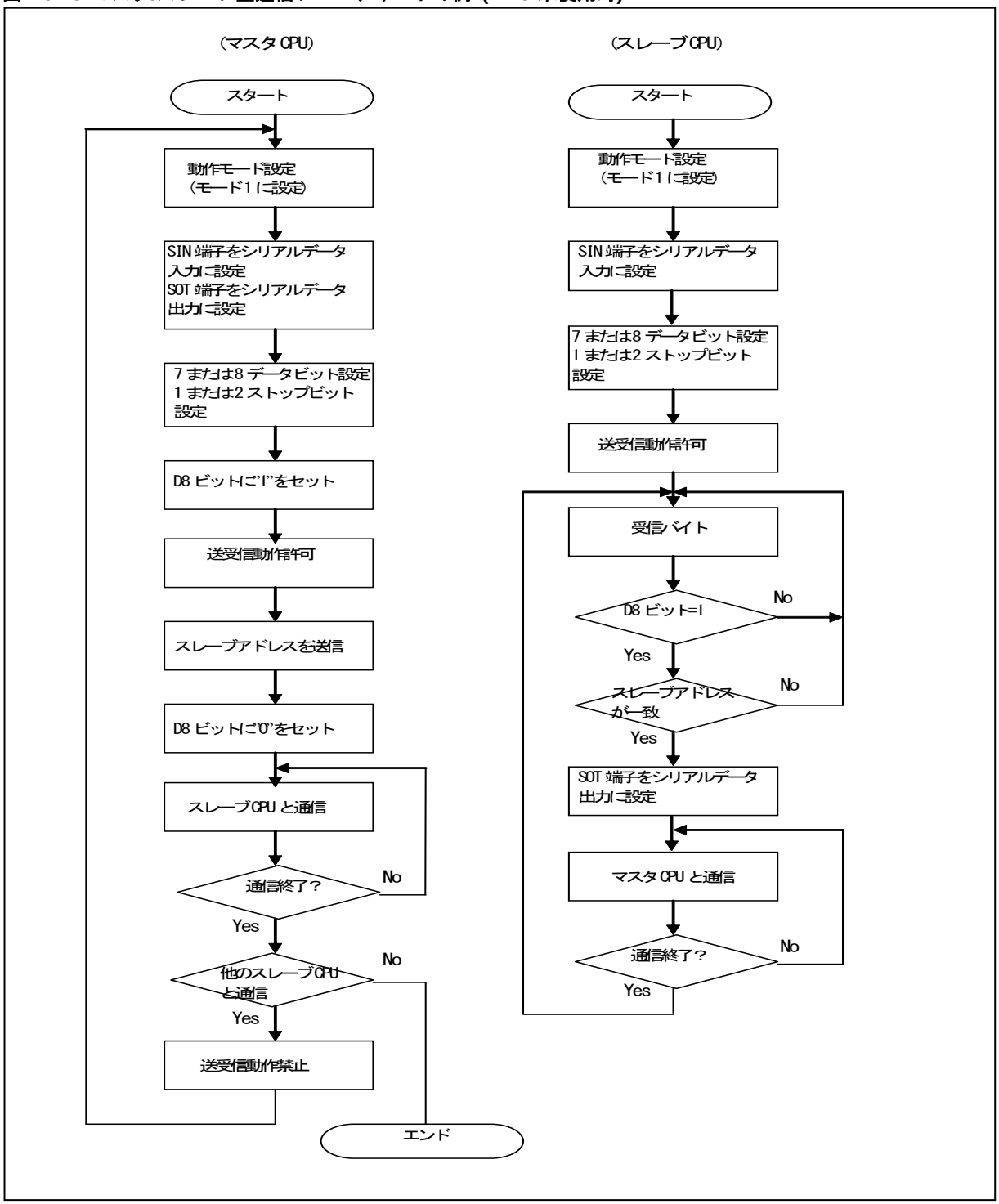
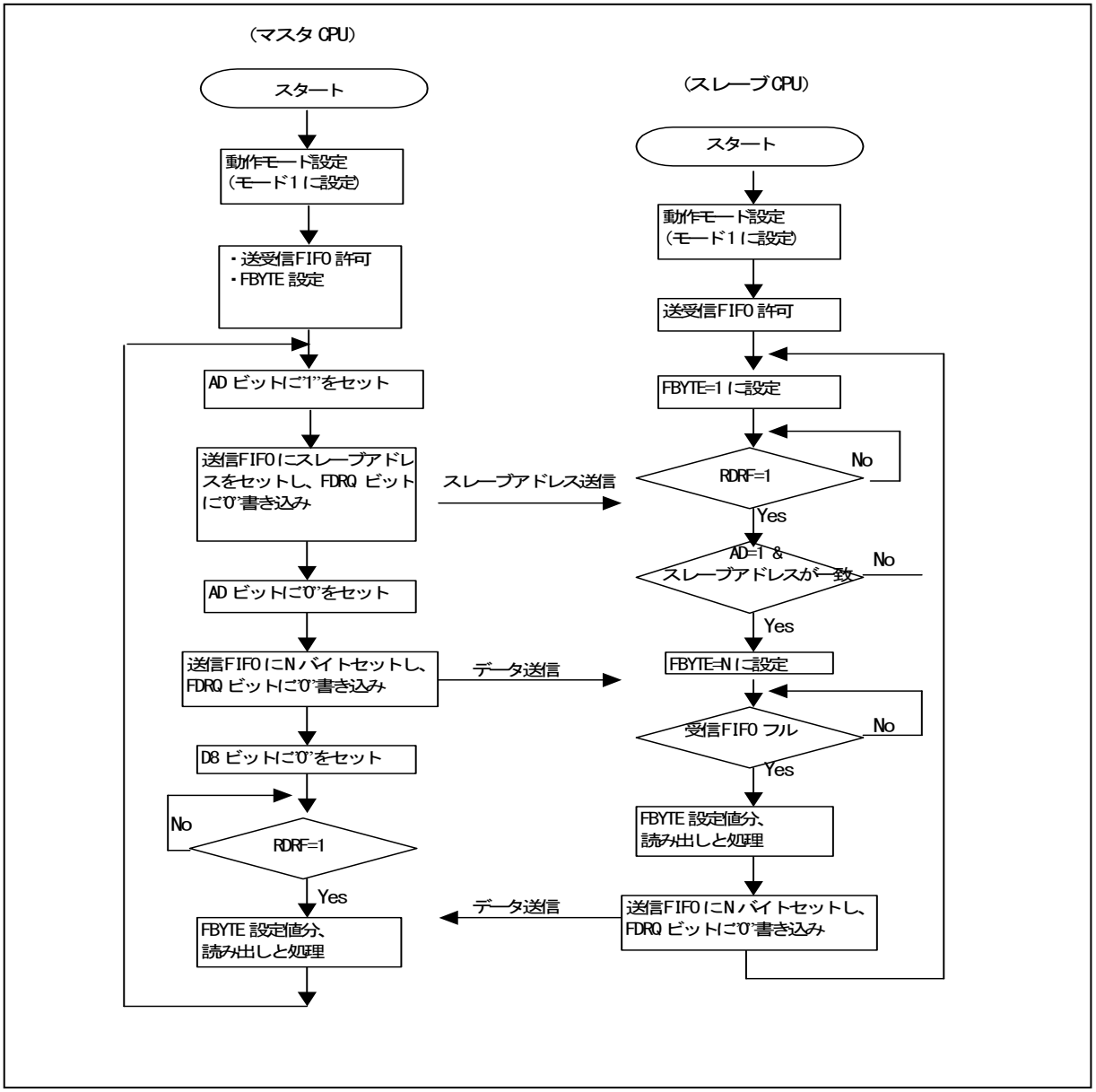


図 40-19 マスタ/スレーブ型通信フローチャートの例 (FIFO 使用時)



40.6 CSIO の動作説明

CSIO の動作について示します。

40.6.1 CSIO の割込み

40.6.2 CSIO の動作

40.6.3 設定手順とプログラムフロー

40.6.1 CSIO の割込み

CSIO の割込みについて説明します。

CSIO (クロック同期シリアルインタフェース)の割込みには、受信割込みと送信割込みがあり、次に示す要因で割込み要求を発生させることができます。

- 受信データが受信データレジスタ(RDR)にセットされた場合、または受信エラーが発生した場合
- 送信データが送信データレジスタ(TDR)から送信用シフトレジスタに転送され、送信が開始された場合
- 送信バスアイドル(送信動作なし)
- 送信 FIFO データ要求

40.6.1.1 CSIO の割込み一覧

CSIO の割込み一覧について示します。

表 40-7 CSIO の割込み制御ビットと割込み要因

割込みの種類	割込み要求フラグビット	フラグレジスタ	割込み要因	割込み要因許可ビット	割込み要求フラグのクリア
受信	RDRF	SSR	1 バイト受信	SCR:RIE	受信データ(RDR)の読出し
			FBYTE 設定値分受信		受信 FIFO がエンプティになるまでの受信データ(RDR)の読出し
			FRIIE ビットが〝1〞で受信 FIFO に有効なデータが存在した状態で 8 ビット時間以上の受信アイドル状態検出		
	ORE	SSR	オーバランエラー		受信エラーフラグクリアビット(SSR:REC)への"1"書込み
送信	TDRE	SSR	送信レジスタがエンプティ	SCR:TIE	送信データ(TDR)への書込み，または 送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送)*1
	TBI	SSR	送信動作なし	SCR:TBIE	送信データ(TDR)への書込み，または 送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送)*1
	FDRQ	FCR1	送信 FIFO がエンプティ	FCR1: FTIE	FIFO 送信データ要求ビット(FCR1:FDRQ)への"0"書込みまたは送信 FIFO がフル

*1 : TDRE ビットが"0"になってから TIE ビットを"1"にしてください。

40.6.1.2 受信割込み発生とフラグセットのタイミング

受信割込み発生とフラグセットのタイミングについて説明します。

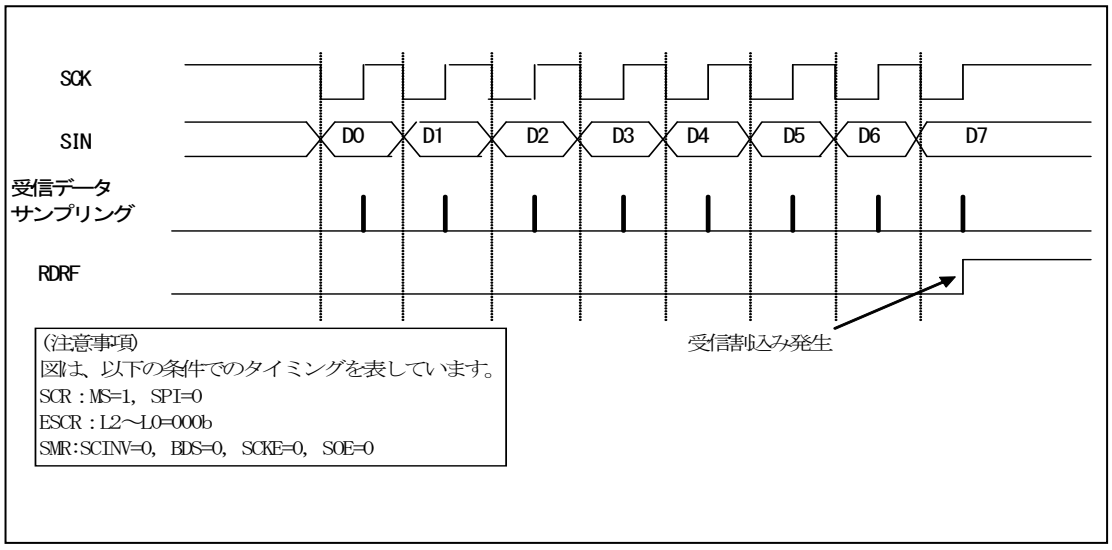
受信時の割込みとしては、受信完了(SSR:RDRF)、および受信エラーの発生(SSR:ORE)があります。

最終データビットを検出されることにより、受信データが受信データレジスタ(RDR)に格納されます。受信が完了したとき(SSR:RDRF=1)または受信エラーが発生(SSR:ORE=1)すると各フラグがセットされます。そのとき、受信割込みが許可(SCR:RIE=1)されていると受信割込みが発生します。

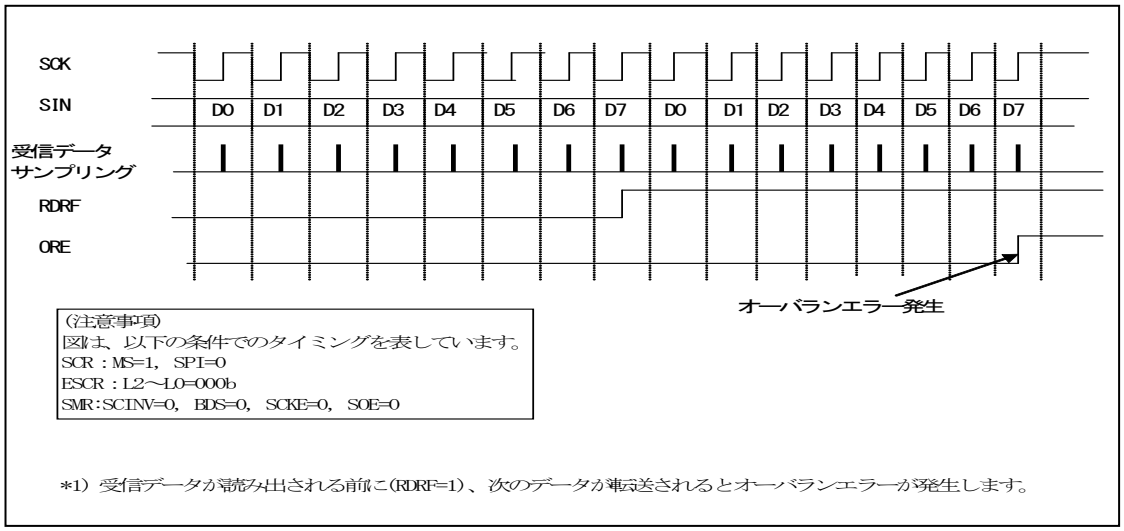
＜注意事項＞
受信エラーが発生した場合は、受信データレジスタ(RDR)のデータは無効となります。

図 40-20 フラグセットのタイミング

受信動作とフラグセットのタイミング



ORE (オーバランエラー) フラグセットタイミング



40.6.1.3 受信 FIFO 使用時の割込み発生とフラグセットのタイミング

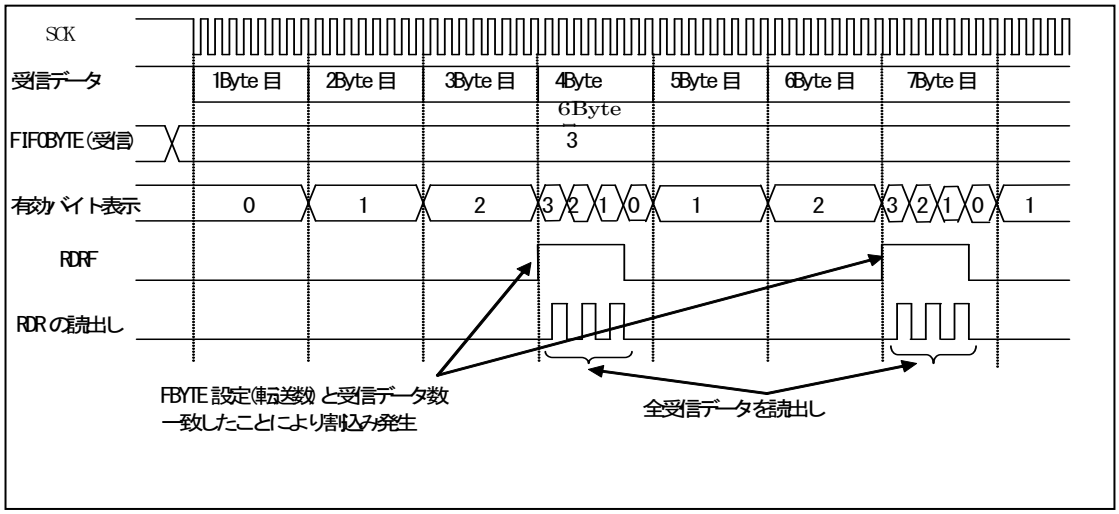
受信 FIFO 使用時の割込み発生とフラグセットのタイミングについて説明します。

受信 FIFO 使用時の割込みは、FBYTE レジスタ(FBYTE)の設定値分受信すると発生します。受信 FIFO 使用時の割込み発生は、FBYTE レジスタの設定値によって決定されます。

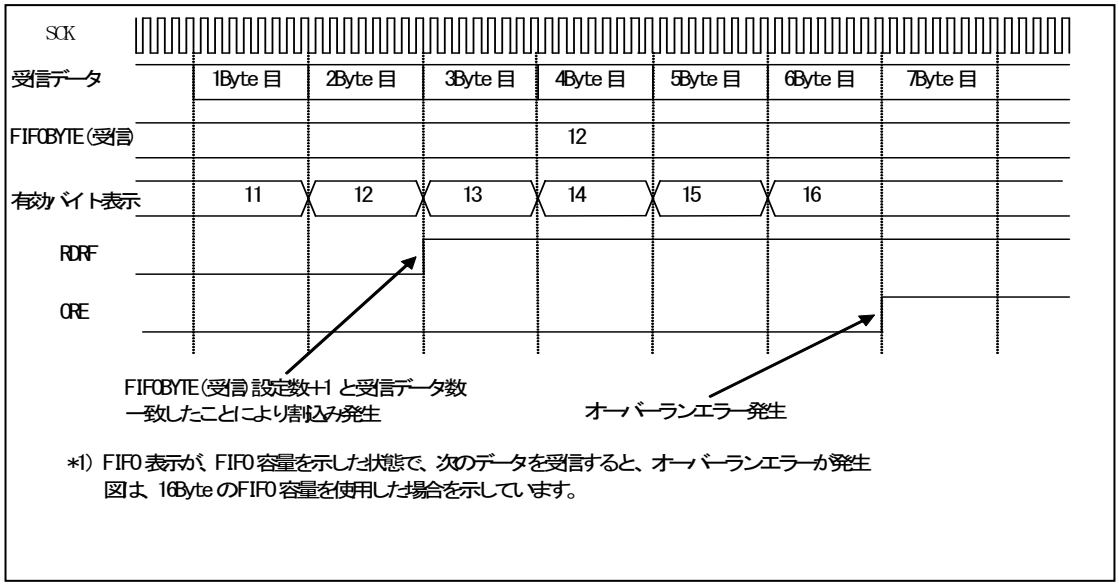
- FBYTE レジスタの転送数設定分のデータを受信するとシリアルステータスレジスタの受信データフルフラグ(SSR:RDRF)が"1"にセットされます。このとき、受信割込み許可(SCR:RIE)されていると受信割込みを発生します。
- 受信 FIFO アイドル検出許可ビット(FRIIE)が"1"で受信 FIFO に存在するデータ数が転送数に達しない場合、受信アイドル状態がボーレートクロックで 8 クロック以上続くと割込みフラグ(RDRF)が"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは 0 にリセットされ、再度 8 クロックをカウントします。受信 FIFO が禁止されるとそのカウンタは 0 にリセットされます。受信 FIFO にデータが残っている状態で受信 FIFO を許可すると再度、カウントを開始します。
- 受信 FIFO がエンプティになるまで受信データ(RDR)を読み出すと、受信データフルフラグ(SSR: RDRF)はクリアされます。
- 受信有効データ数表示が FIFO 容量を示した状態で、次のデータを受信するとオーバランエラー(SSR:ORE=1)が発生します。

図 40-21 割込み発生・フラグセットのタイミング

受信 FIFO 使用時の受信割込み発生タイミング



ORE (オーバランエラー)フラグビットのセットタイミング



*1) FIFO 表示が、FIFO 容量を示した状態で、次のデータを受信すると、オーバーランエラーが発生
図は、16Byte の FIFO 容量を使用した場合を示しています。

40.6.1.4 送信割込み発生とフラグセットのタイミング

送信割込み発生とフラグセットのタイミングについて説明します。

送信時の割込みとしては、送信データが、送信データレジスタ(TDR)から送信用シフトレジスタに転送され(SSR:TDRE=1)送信が開始された場合と、送信動作をしていないとき(SSR:TBI=1)に発生します。

送信データエンプティフラグ (SSR:TDRE) のセットタイミング

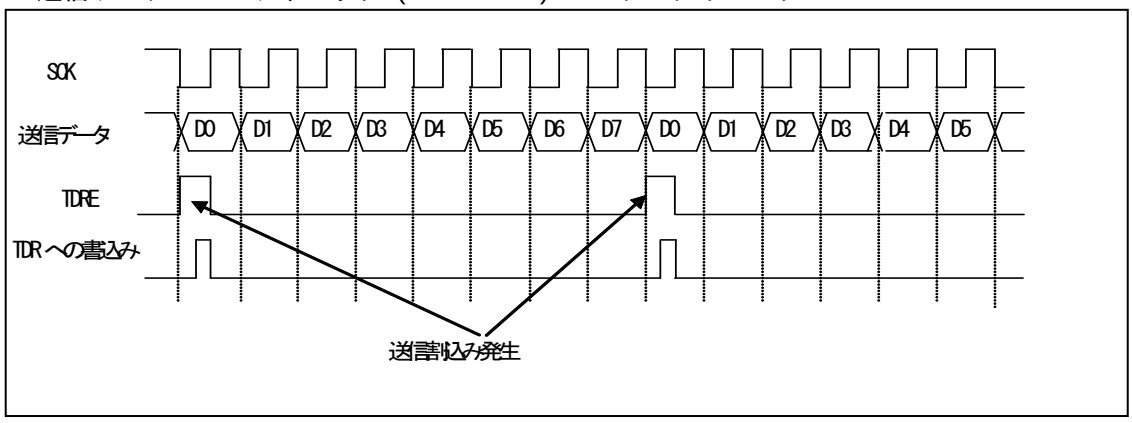
送信データレジスタ(TDR)に書込まれたデータが送信シフトレジスタに転送されると、次のデータの書込みが可能な状態(SSR:TDRE=1)になります。そのとき、送信割込みが許可(SCR:TIE=1)されていると、送信割込みが発生します。SSR:TDRE ビットはリードオンリビットなので、送信データレジスタ(TDR)へのデータ書込みにより"0"にクリアされます。

送信バスアイドルフラグ (SSR:TBI) のセットタイミング

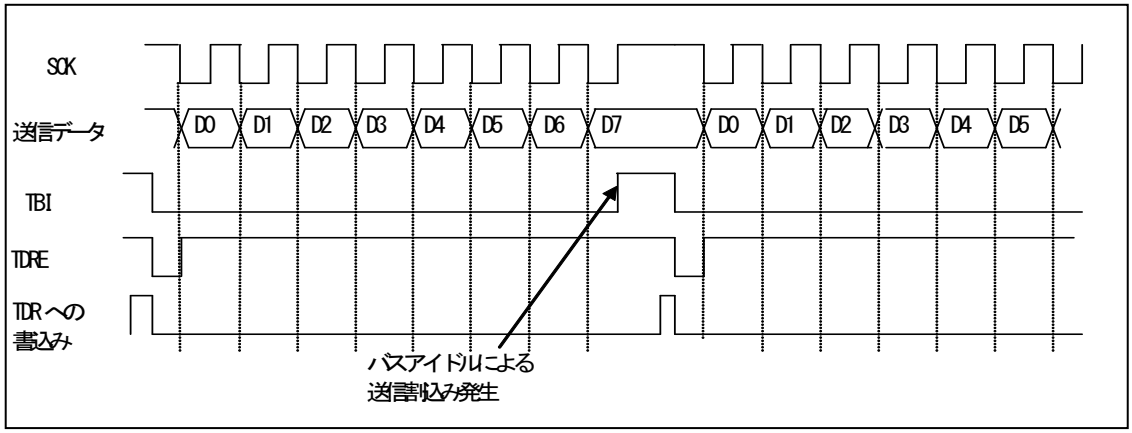
送信データレジスタがエンプティ(SSR:TDRE=1)で送信動作をしていないとき、SSR:TBI ビットは"1"にセットされます。このとき、送信バスアイドル割込み許可(SCR:TBIE=1)されていると、送信割込みが発生します。送信データレジスタ(TDR)に送信データをセットすると SSR:TBI ビットおよび送信割込み要求はクリアされます。

図 40-22 フラグセットのタイミング

送信データエンプティフラグ (SSR:TDRE) のセットタイミング



送信バスアイドルフラグ (SSR:TBI) のセットタイミング



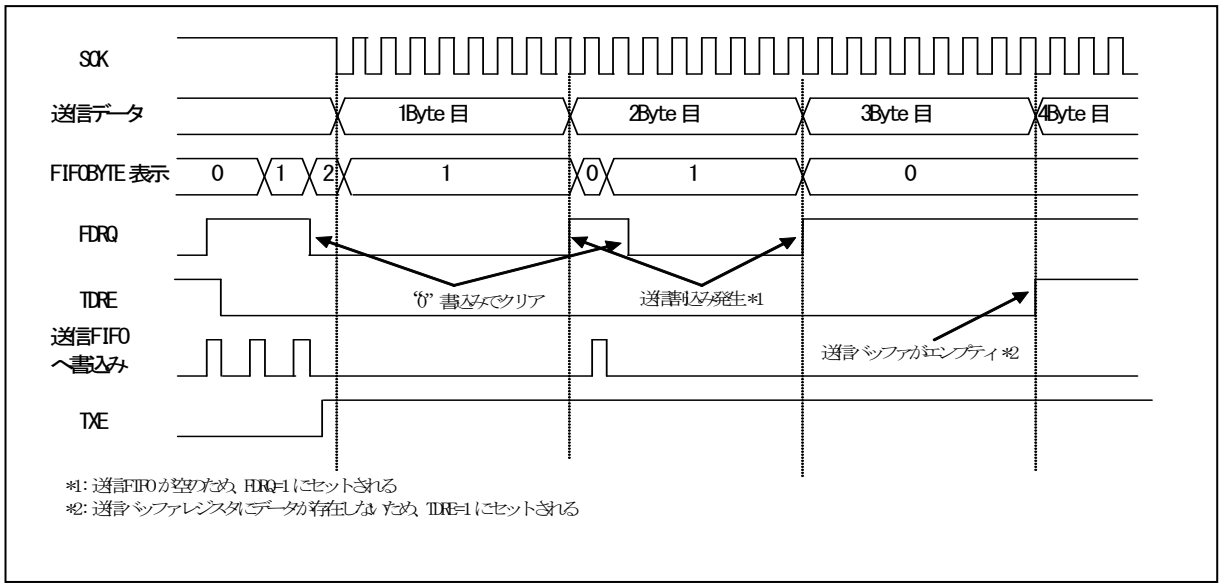
40.6.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング

送信 FIFO 使用時の割込み発生とフラグセットのタイミングについて説明します。

送信 FIFO 使用時の割込みは、送信 FIFO にデータが存在しないときに発生します。

- 送信 FIFO にデータが存在しない場合、FIFO 送信データ要求ビット(FCR1:FDRQ)が"1"にセットされます。このとき、FIFO 送信割込み許可(FCR1:FTIE=1)されていると送信割込みが発生します。
- 送信割込みが発生して送信 FIFO に必要なデータを書き込んだら、FIFO 送信データ要求ビット(FCR1:FDRQ)に"0"書き込んで割込み要求をクリアしてください。
- 送信 FIFO がフルになると FIFO 送信データ要求ビット(FCR1:FDRQ)は"0"になります。
- 送信 FIFO のデータの存在は、FIFO バイトレジスタ(FBYTE)を読み出すことで確認できます。FBYTE=00_H のときは、送信 FIFO にデータが存在していないことを示します。

図 40-23 割込み発生タイミング



40.6.2 CSIO の動作

CSIO の動作について示します。

40.6.2.1 ノーマル転送 (I)

40.6.2.2 ノーマル転送 (II)

40.6.2.3 SPI 転送 (I)

40.6.2.4 SPI 転送 (II)

40.6.2.5 ボーレートの生成

40.6.2.1 ノーマル転送 (I)

ノーマル転送 (I) について説明します。

■ 特長

	項目	説明
1	シリアルクロック(SCK)のマークレベル	"H"
2	送信データ出力タイミング	SCK の立下りエッジ
3	受信データのサンプリング	SCK の立上りエッジ
4	データ長	5～9 ビット

■ レジスタ設定

ノーマル転送 (I) に必要な、レジスタの設定値を以下に示します。

表 40-8 ノーマル転送 (I) レジスタ設定

	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SCR/SMR	UPCL	MS	SPI	RIE	TIE	TBIE	RXE	TXE	MD2	MD1	MD0	―	SCINV	BDS	SCKE	SOE
	0	1/0	0	*	*	*	*	*	0	1	0	―	0	*	1/0	1/0
SSR/ESCR	REC	―	―	―	ORE	RDRF	TDRE	TBI	SOP	―	―	WT1	WT0	L2	L1	L0
	0	―	―	―	―	―	―	―	0	―	―	*	*	*	*	*
TDR/RDR								D8	D7	D6	D5	D4	D3	D2	D1	D0
								*	*	*	*	*	*	*	*	*
BGR	―	BGR[14:8]							BGR[7:0]							
	―	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

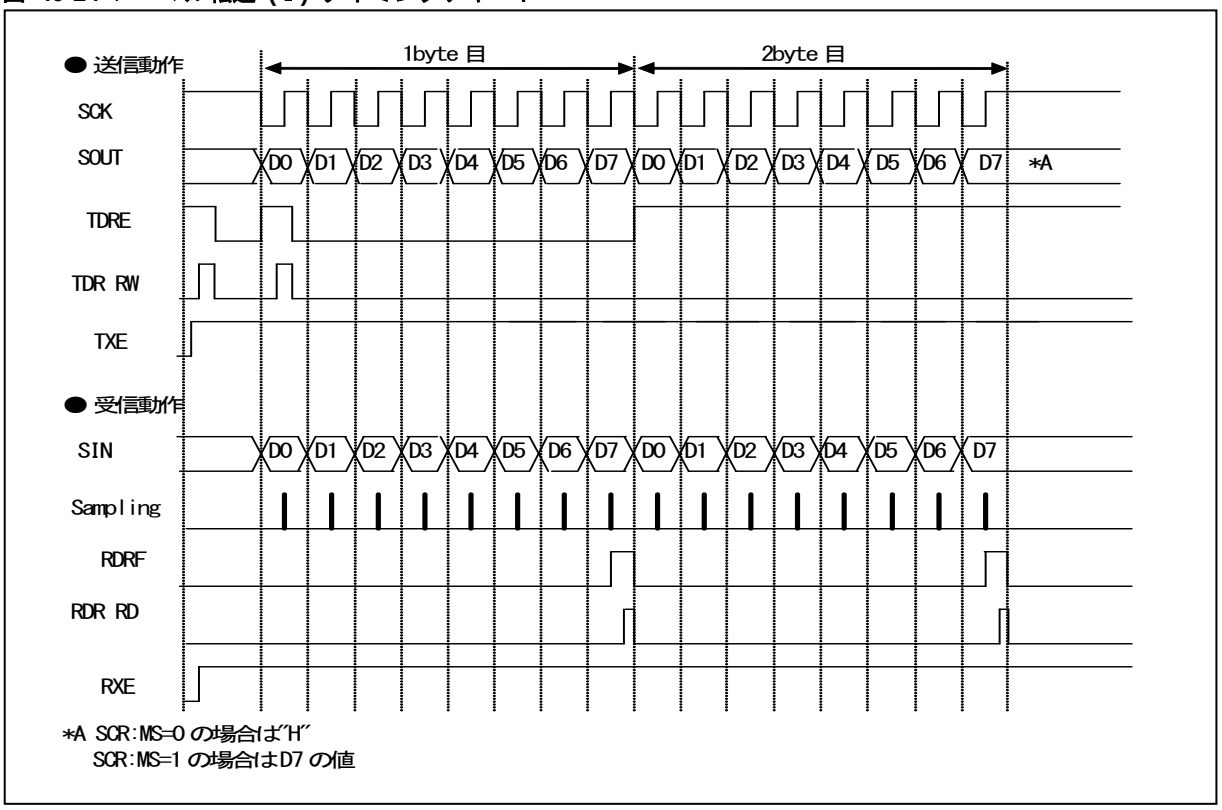
1 : 1 を設定　0 : 0 を設定
* : ユーザが決める設定

＜注意事項＞
上記ビットの設定値(1/0)は、マスタ動作、スレーブ動作で異なります。以下のように設定してください。

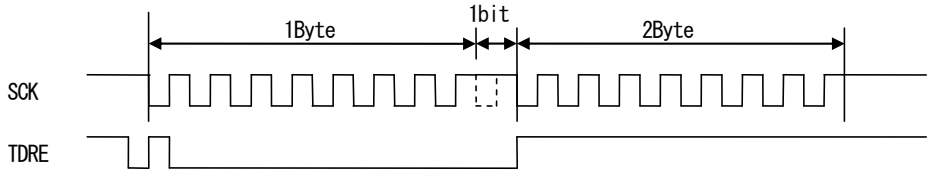
マスタ送信時　: SCR:MS=0, SMR:SCKE=1, SOE=1
マスタ受信時　: SCR:MS=0, SMR:SCKE=1, SOE=0
スレーブ送信時 : SCR:MS=1, SMR:SCKE=0, SOE=1
スレーブ受信時 : SCR:MS=1, SMR:SCKE=0, SOE=0

■ ノーマル転送（I）タイミングチャート

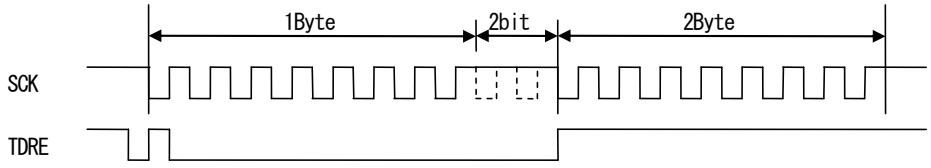
図 40-24 ノーマル転送（I）タイミングチャート



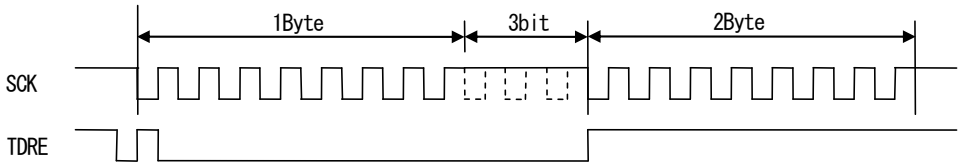
■ ESCR:WT1=0, ESCR:WT0=1(マスタ時)



■ ESCR:WT1=1, ESCR:WT0=0(マスタ時)



■ ESCR:WT1=1, ESCR:WT0=1(マスタ時)



■ 動作説明

[1] マスタ動作(SCR:MS=0, SMR:SCKE=1 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1), 送信動作許可(SCR:TXE=1)および受信動作禁止(SCR: RXE=0)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)出力の立下りエッジに同期して、送信データを出力します。

②最初の 1 ビット目の送信データが出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR: TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0), 送信動作許可(SCR:TXE=1)および受信動作許可(SCR: RXE=1)にし、TDR にダミーデータを書き込むとシリアルクロック出力(SCK)の立上りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

＜注意事項＞

■ 受信動作のみを行う場合、シリアルクロック(SCK)を出力させるために TDR にダミーデータを書いてください。

■ 送受信 FIFO 許可時は、転送させたいフレーム分 FBYTE レジスタに設定することで、設定値分のフレームのシリアルクロック(SCK)が出力されます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1)、受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)出力の立下りエッジに同期して、送信データを出力します。最初の 1 ビット目の送信データが出力されると SSR: TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)出力の立上りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

■ 連続データ送信または受信ウェイト動作

①連続データ送信または受信に対し、(ESCR:WT1, ESCR:WT0)= (0, 0)以外を設定した場合 Frame 間にウェイトが挿入されます。

[2] スレーブ動作(SCR:MS=1, SMR:SCKE=0 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1)および送信動作許可(SCR:TXE=1)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)入力の立下りエッジに同期して、送信データを出力します。

②最初の 1 ビット目の送信データが出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR: TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0)および受信動作許可(SCR:RXE=1)にすると、シリアルクロック入力(SCK)の立上りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1), 受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)入力の立下りエッジに同期して、送信データを出力します。最初の 1 ビット目の送信データが出力されると SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)入力の立上りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

40.6.2.2 ノーマル転送 (II)

ノーマル転送 (II) について説明します。

■ 特長

	項目	説明
1	シリアルクロック(SCK)のマークレベル	"L"
2	送信データ出力タイミング	SCK の立上りエッジ
3	受信データのサンプリング	SCK の立下りエッジ
4	データ長	5～9 ビット

■ レジスタ設定

ノーマル転送(Ⅱ)に必要な、レジスタの設定値を以下に示します。

表 40-9 ノーマル転送(Ⅱ)レジスタ設定

	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SCR/ SMR	UPCL	MS	SPI	RIE	TIE	TBIE	RXE	TXE	MD2	MD1	MD0	―	SCINV	BDS	SCKE	SOE
	0	1/0	0	*	*	*	*	*	0	1	0	―	1	*	1/0	1/0
SSR/ ESCR	REC	―	―	―	ORE	RDRF	TDRE	TBI	SOP	―	―	WT1	WT0	L2	L1	L0
	0	―	―	―	―	―	―	―	0	―	―	*	*	*	*	*
TDR/ RDR								D8	D7	D6	D5	D4	D3	D2	D1	D0
								*	*	*	*	*	*	*	*	*
BGR	―	BGR[14:8]							BGR[7:0]							
	―	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

1 : 1 を設定　0 : 0 を設定

* : ユーザが決める設定

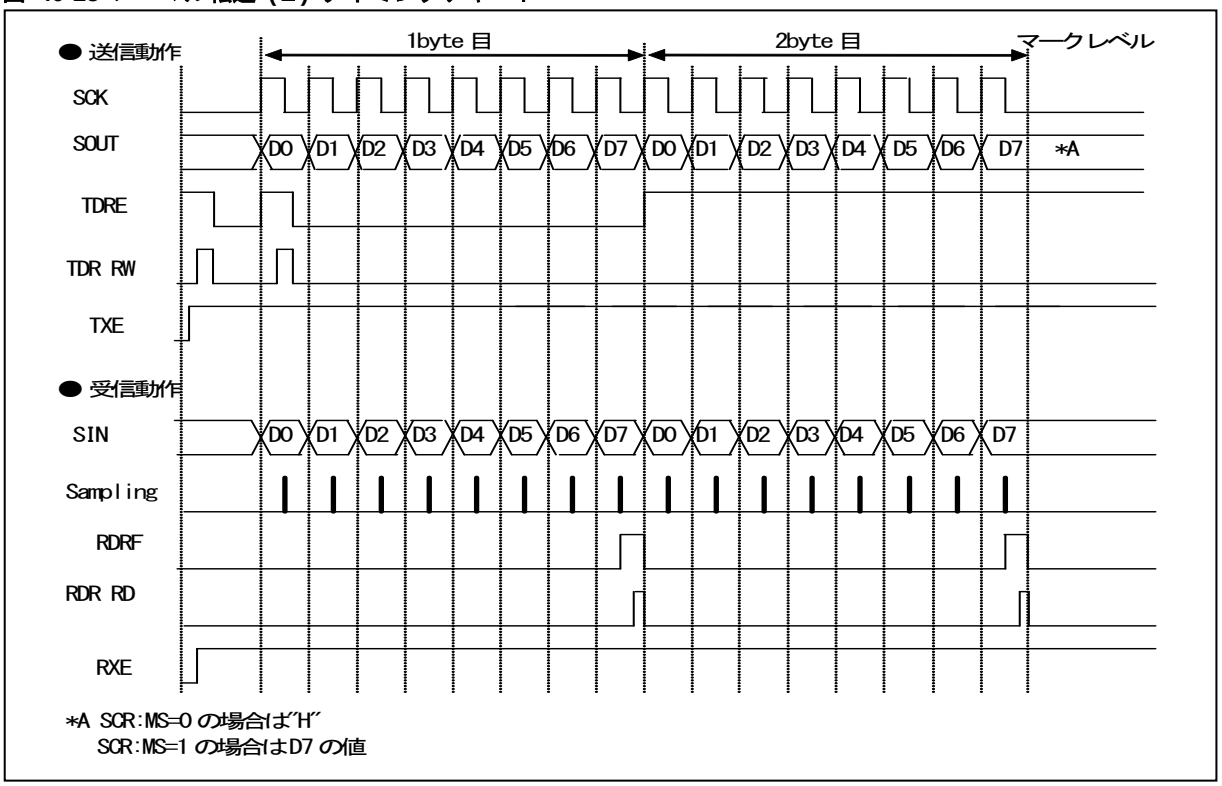
＜注意事項＞

上記ビットの設定値(1/0)は、マスタ動作，スレーブ動作で異なります。以下のように設定してください。

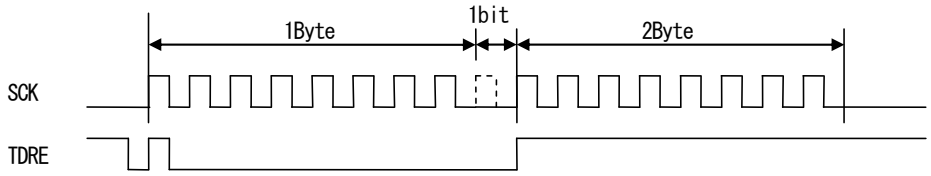
マスタ送信時　: SCR:MS=0, SMR:SCKE=1, SOE=1
マスタ受信時　: SCR:MS=0, SMR:SCKE=1, SOE=0
スレーブ送信時 : SCR:MS=1, SMR:SCKE=0, SOE=1
スレーブ受信時 : SCR:MS=1, SMR:SCKE=0, SOE=0

■ ノーマル転送（Ⅱ）タイミングチャート

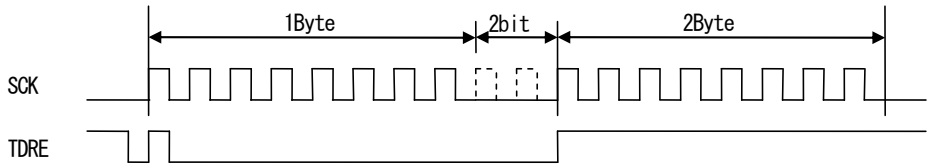
図 40-25 ノーマル転送（Ⅱ）タイミングチャート



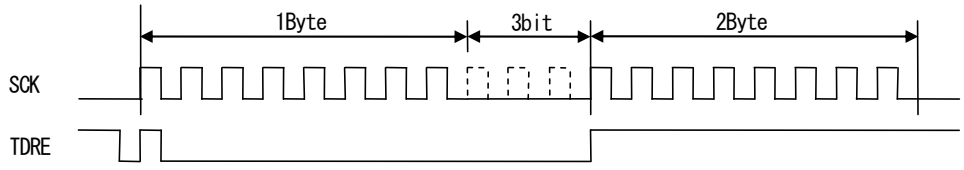
■ ESCR:WT1=0, ESCR:WT0=1(マスタ時)



■ ESCR:WT1=1, ESCR:WT0=0(マスタ時)



■ ESCR:WT1=1, ESCR:WT0=1(マスタ時)



■ 動作説明

[1] マスタ動作 (SCR:MS=0, SMR:SCKE=1 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1), 送信動作許可(SCR:TXE=1)および受信動作禁止(SCR: RXE=0)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)出力の立上りエッジに同期して、送信データを出力します。

②最初の 1 ビット目の送信データが出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR: TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0), 送信動作許可(SCR:TXE=1)および受信動作許可(SCR:RXE=1)にし、TDR にダミーデータを書き込むとシリアルクロック出力(SCK)の立下りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

＜注意事項＞

■ 受信動作のみを行う場合、シリアルクロック(SCK)を出力させるために TDR にダミーデータを書いてください。

■ 送受信 FIFO 許可時、転送させたいフレーム分 FBYTE レジスタに設定することで、設定値分のフレームのシリアルクロック(SCK)が出力されます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1), 受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)出力の立上りエッジに同期して、送信データを出力します。最初の 1 ビット目の送信データが出力されると SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)出力の立下りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

■ 連続データ送信または受信ウェイト動作

①連続データ送信または受信に対し、(ESCR:WT1, WT0)= (0, 0)以外を設定した場合 Frame 間にウェイトが挿入されます。

[2] スレーブ動作 (SCR:MS=1, SMR:SCKE=0 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1)および送信動作許可(SCR:TXE=1)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)入力の立上りエッジに同期して、送信データを出力します。

②最初の 1 ビット目の送信データが出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR: TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0)および受信動作許可(SCR:RXE=1)にすると、シリアルクロック入力(SCK)の立下りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1), 受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)入力の立上りエッジに同期して、送信データを出力します。最初の 1 ビット目の送信データが出力されると SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)入力の立下りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

40.6.2.3 SPI 転送 (I)

SPI 転送(I)について説明します。

■ 特長

	項目	説明
1	シリアルクロック (SCK) のマークレベル	"H"
2	送信データ出力タイミング	SCK の立上りエッジ
3	受信データのサンプリング	SCK の立下りエッジ
4	データ長	5～9 ビット

■ レジスタ設定

SPI 転送(Ⅰ)に必要な、レジスタの設定値を以下に示します。

表 40-10 SPI 転送(Ⅰ)レジスタ設定

	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SCR/ SMR	UPCL	MS	SPI	RIE	TIE	TBIE	RXE	TXE	MD2	MD1	MD0	―	SCINV	BDS	SCKE	SOE
	0	1/0	1	*	*	*	*	*	0	1	0	―	0	*	1/0	1/0
SSR/ ESCR	REC	―	―	―	ORE	RDRF	TDRE	TBI	SOP	―	―	WT1	WT0	L2	L1	L0
	0	―	―	―	―	―	―	―	0	―	―	*	*	*	*	*
TDR/ RDR								D8	D7	D6	D5	D4	D3	D2	D1	D0
								*	*	*	*	*	*	*	*	*
BGR	―	BGR[14:8]							BGR[7:0]							
	―	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

1 : 1 を設定　0 : 0 を設定
* : ユーザが決める設定

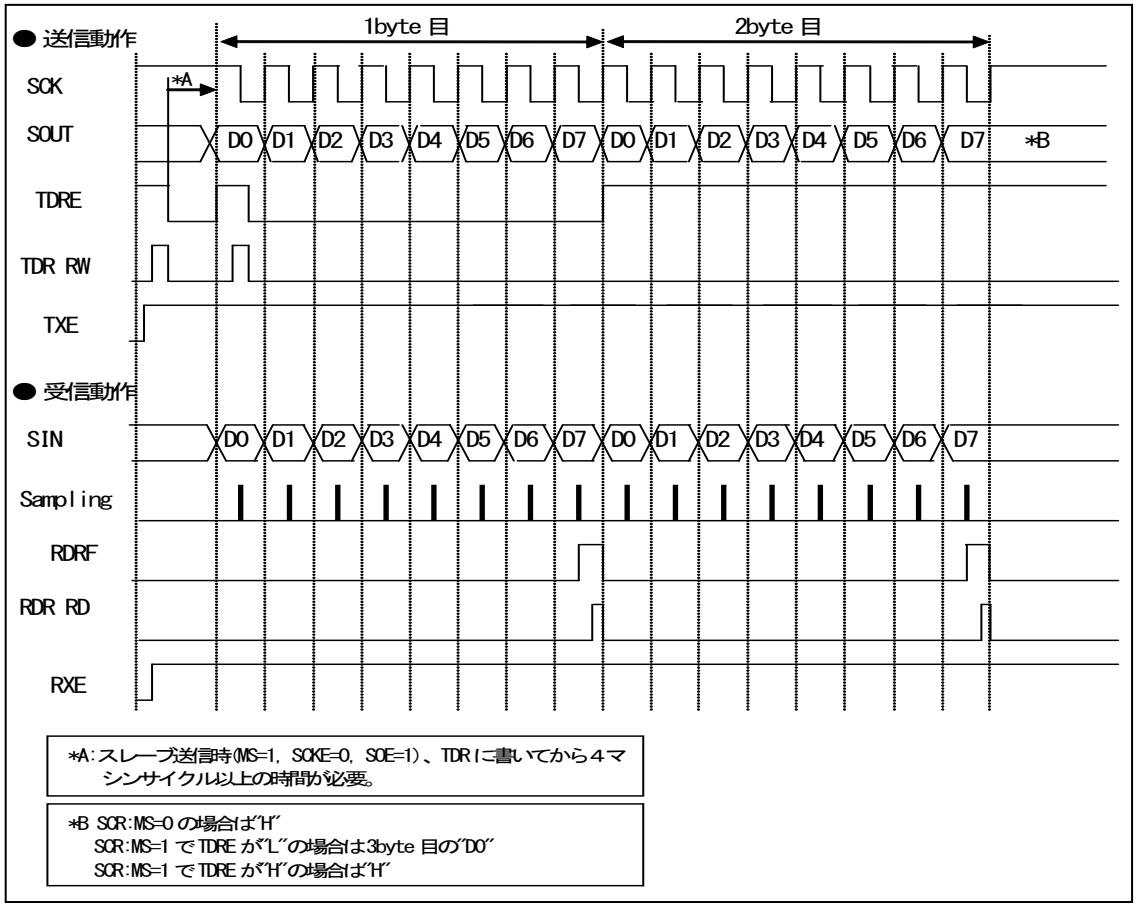
＜注意事項＞

上記ビットの設定値(1/0)は、マスタ動作, スレーブ動作で異なります。以下のように設定してください。

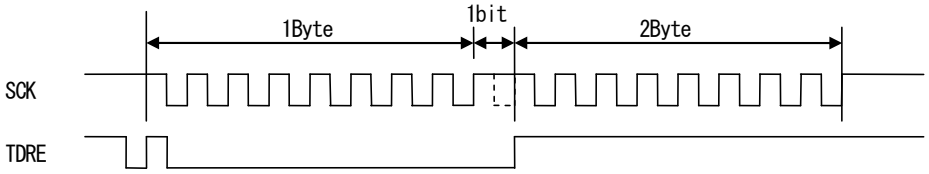
マスタ送信時　: SCR:MS=0, SMR:SCKE=1, SOE=1
マスタ受信時　: SCR:MS=0, SMR:SCKE=1, SOE=0
スレーブ送信時 : SCR:MS=1, SMR:SCKE=0, SOE=1
スレーブ受信時 : SCR:MS=1, SMR:SCKE=0, SOE=0

■ SPI 転送（I）タイミングチャート

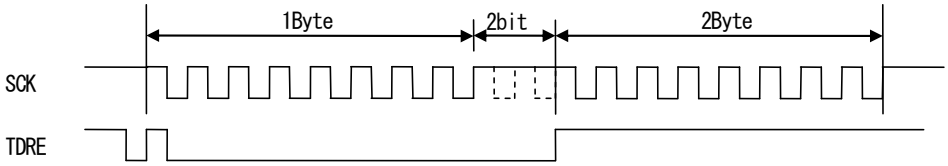
図 40-26 SPI 転送（I）タイミングチャート



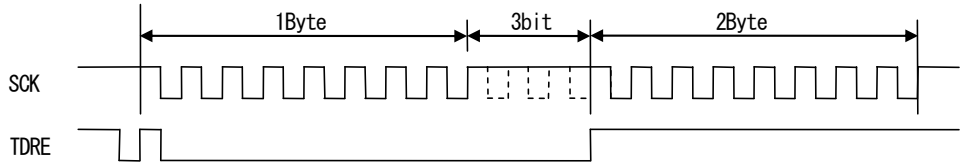
■ ESCR:WT1=0, ESCR:WTO=1(マスタ時)



■ ESCR:WT1=1, ESCR:WTO=0(マスタ時)



■ ESCR:WT1=1, ESCR:WTO=1(マスタ時)



■ 動作説明

[1] マスタ動作(SCR:MS=0, SMR:SCKE=1 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1), 送信動作許可(SCR:TXE=1)および受信動作禁止(SCR: RXE=0)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となり 1 ビット目が出力されます。その後、シリアルクロック(SCK)出力の立上りエッジに同期して、送信データを出力します。

②最初のシリアルクロックの立下りエッジの半サイクル前で、SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0), 送信動作許可(SCR:TXE=1)および受信動作許可(SCR: RXE=1)にし、TDR にダミーデータを書き込むとシリアルクロック出力(SCK)の立下りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

＜注意事項＞

■ 受信動作のみを行う場合、シリアルクロック(SCK)を出力させるために TDR にダミーデータを書いてください。

■ 送受信 FIFO 許可時、転送させたいフレーム分 FBYTE レジスタに設定することで、設定値分のフレームのシリアルクロック(SCK)が出力されます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1), 受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となり 1 ビット目が出力されます。その後、シリアルクロック(SCK)出力の立上りエッジに同期して、送信データを出力します。最初のシリアルクロックの立下りエッジの半サイクル前で、SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)出力の立下りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

■ 連続データ送信または受信ウェイト動作

①連続データ送信または受信に対し、(ESCR:WT1, WT0)= (0, 0)以外を設定した場合 Frame 間にウェイトが挿入されます。

[2] スレーブ動作(SCR : MS=1, SMR : SCKE=0 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1)および送信動作許可(SCR:TXE=1)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となり 1 ビット目が出力されます。その後、シリアルクロック(SCK)出力の立上りエッジに同期して、送信データを出力します。

②送信データの 1 ビット目が出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0)および受信動作許可(SCR:RXE=1)にすると、シリアルクロック入力(SCK)の立下りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。

このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1), 受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となり 1 ビット目が出力されます。その後、シリアルクロック(SCK)入力の立上りエッジに同期して、送信データを出力します。送信データの 1 ビット目が出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)入力の立下りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

■ 受信動作から送信動作への連続的な切換え

①シリアルデータ出力禁止(SMR:SOE=0), 受信割込み許可(SCR:RIE=1), 受信動作許可(SCR: RXE=1)および送信動作許可(SCR:TXE=1)にします。シリアルクロック(SCK)がマークレベル時に TDR にダミーデータを書き込むと、シリアルクロック入力(SCK)の立下りエッジで受信データをサンプリングします。

②受信動作を継続する場合、受信割込み要求後から次のシリアルクロック(SCK)の立上りまでに TDR にダミーデータを書き込んでください。

③受信動作から送信動作へ切え換える場合、受信割込み要求後から次のシリアルクロック(SCK)の立上りまでにシリアルデータ出力許可(SMR:SOE=1), 受信割込み禁止(SCR:RIE=0)および受信動作禁止(SCR:RXE=0)にし、TDR に送信データを書込みと受信動作終了後にシリアルクロックの立上りエッジに同期して送信データを出力します。

40.6.2.4 SPI 転送 (II)

SPI 転送(II)について説明します。

■ 特長

	項目	説明
1	シリアルクロック(SCK)のマークレベル	"L"
2	送信データ出力タイミング	SCK の立下りエッジ
3	受信データのサンプリング	SCK の立上りエッジ
4	データ長	5～9 ビット

■ レジスタ設定

SPI 転送 (Ⅱ) に必要な、レジスタの設定値を以下に示します。

表 40-11 SPI 転送 (Ⅱ) レジスタ設定

	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SCR/ SMR	UPCL	MS	SPI	RIE	TIE	TBIE	RXE	TXE	MD2	MD1	MD0	―	SCINV	BDS	SCKE	SOE
	0	1/0	1	*	*	*	*	*	0	1	0	―	1	*	1/0	1/0
SSR/ ESCR	REC	―	―	―	ORE	RDRF	TDRE	TBI	SOP	―	―	WT1	WT0	L2	L1	L0
	0	―	―	―	―	―	―	―	0	―	―	*	*	*	*	*
TDR/ RDR								D8	D7	D6	D5	D4	D3	D2	D1	D0
								*	*	*	*	*	*	*	*	*
BGR	―	BGR[14:8]							BGR[7:0]							
	―	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

1 : 1 を設定　0 : 0 を設定

* : ユーザが決める設定

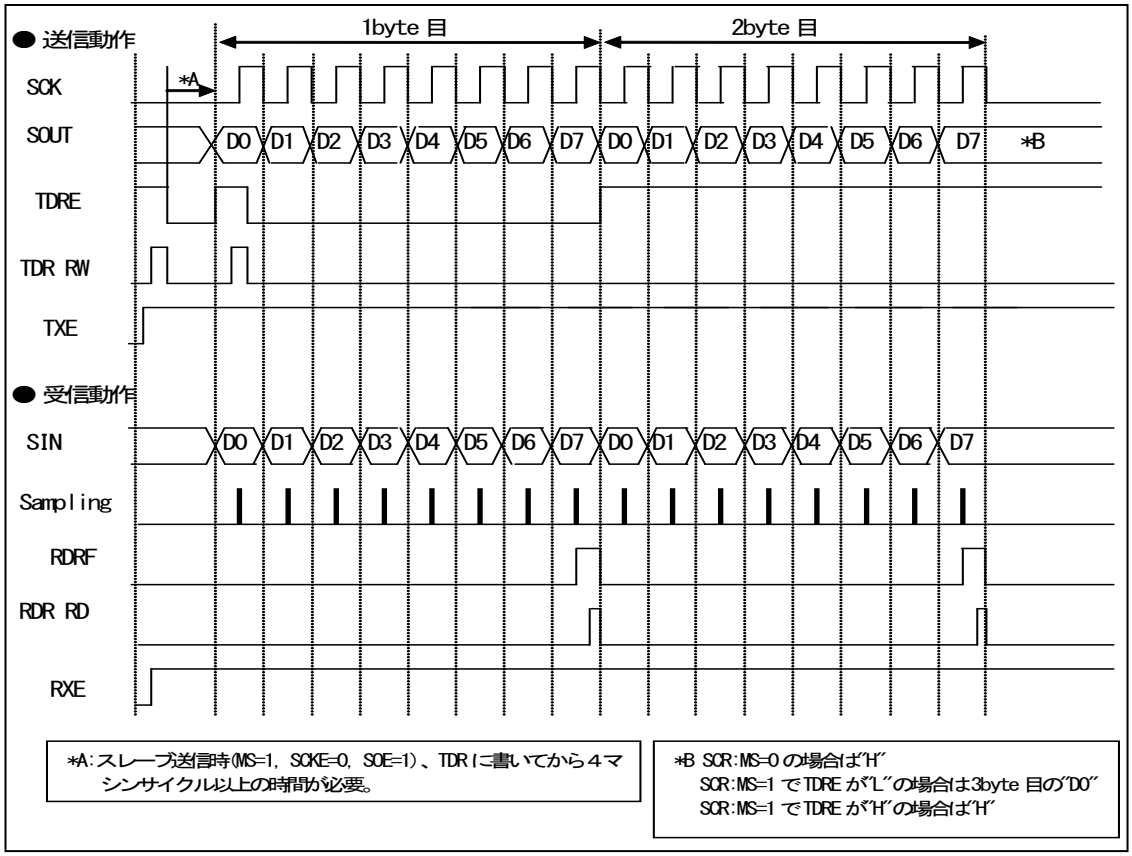
＜注意事項＞

上記ビットの設定値(1/0)は、マスタ動作, スレーブ動作で異なります。以下のように設定してください。

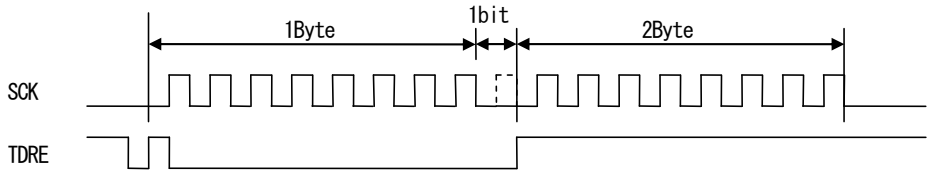
マスタ送信時　: SCR:MS=0, SMR:SCKE=1, SOE=1
マスタ受信時　: SCR:MS=0, SMR:SCKE=1, SOE=0
スレーブ送信時 : SCR:MS=1, SMR:SCKE=0, SOE=1
スレーブ受信時 : SCR:MS=1, SMR:SCKE=0, SOE=0

■ SPI 転送（Ⅱ）タイミングチャート

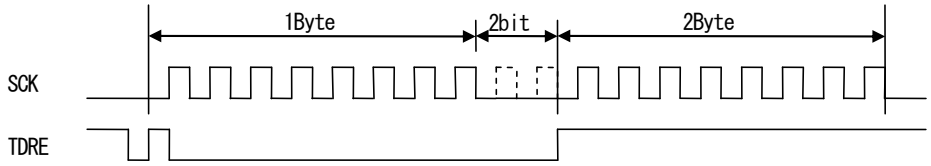
図 40-27 SPI 転送（Ⅱ）タイミングチャート



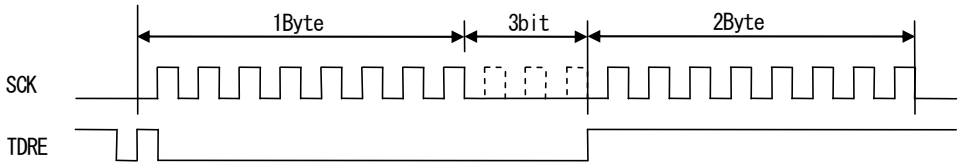
■ ESCR:WT1=0, ESCR:WTO=1(マスタ時)



■ ESCR:WT1=1, ESCR:WTO=0(マスタ時)



■ ESCR:WT1=1, ESCR:WTO=1(マスタ時)



■ 動作説明

[1] マスタ動作(SCR:MS=0, SMR:SCKE=1 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1), 送信動作許可(SCR:TXE=1)および受信動作禁止(SCR: RXE=0)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)出力の立下りエッジに同期して、送信データを出力します。

②最初のシリアルクロック(SCK)出力の立上りエッジの半サイクル前で、SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0), 送信動作許可(SCR:TXE=1)および受信動作許可(SCR: RXE=1)にし、TDR にダミーデータを書き込むとシリアルクロック出力(SCK)の立上りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

＜注意事項＞

■ 受信動作のみを行う場合、シリアルクロック(SCK)を出力させるために TDR にダミーデータを書いてください。

■ 送受信 FIFO 許可時、転送させたいフレーム分 FBYTE レジスタに設定することで、設定値分のフレームのシリアルクロック(SCK)が出力されます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1), 送受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となり 1 ビット目が出力されます。その後、シリアルクロック(SCK)出力の立下りエッジに同期して、送信データを出力します。最初のシリアルクロックの立上りエッジの半サイクル前で、SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)出力の立上りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

■ 連続データ送信または受信ウェイト動作

①連続データ送信または受信に対し、(ESCR:WT1, WT0)= (0, 0)以外を設定した場合 Frame 間にウェイトが挿入されます。

[2] スレーブ動作(SCR:MS=1, SMR:SCKE=0 に設定します)

■ 送信動作

①シリアルデータ出力許可(SMR:SOE=1)および送信動作許可(SCR:TXE=1)にし、TDR に送信データを書き込むと、SSR:TDRE=0 となりシリアルクロック(SCK)出力の立下りエッジに同期して、送信データを出力します。

②最初の 1 ビット目の送信データが出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR: TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

＜注意事項＞

送信動作許可後(SCR:TXE=1)、最初の TDR への送信データ書込みはシリアルクロック(SCK)がマークレベルのとき以外で行うと、1 ビット目のデータが出力されず、正常に送信動作を行いません。送信動作許可後(SCR:TXE=1)、最初の TDR への送信データ書込みはシリアルクロック(SCK)がマークレベルのときに行なってください。

■ 受信動作

①シリアルデータ出力禁止(SMR:SOE=0)および受信動作許可(SCR:RXE=1)にすると、シリアルクロック入力(SCK)の立上りエッジで、受信データをサンプリングします。

②最後のビットを受信すると、SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。

③受信データ(RDR)を読み出すと、SSR:RDRF は"0"にクリアされます。

■ 送受信動作

①送受信動作を同時に行う場合は、シリアルデータ出力許可(SMR:SOE=1), 受信動作許可(SCR:TXE, RXE=1)にします。

②TDR に送信データを書き込むと、SSR:TDRE=0 となり 1 ビット目が出力されます。その後、シリアルクロック(SCK)入力の立下りエッジに同期して、送信データを出力します。送信データの 1 ビット目が出力されると、SSR:TDRE=1 となり、送信割込み許可(SCR:TIE=1)されていると送信割込み要求を出力します。このとき、2 バイト目の送信データを書き込むことができます。

③受信データをシリアルクロック(SCK)入力の立上りエッジでサンプリングします。受信データの最後のビットを受信すると SSR:RDRF=1 となり、受信割込み許可(SCR:RIE=1)されていると、受信割込み要求を出力します。このとき、受信データ(RDR)を読み出すことができます。受信データを読み出すと SSR:RDRF は"0"にクリアされます。

■ 受信動作から送信動作への連続的な切換え

①シリアルデータ出力禁止(SMR:SOE=0), 受信割込み許可(SCR:RIE=1), 受信動作許可(SCR: RXE=1)および送信動作許可(SCR:TXE=1)にします。シリアルクロック(SCK)がマークレベル時に TDR にダミーデータを書き込むと、シリアルクロック入力(SCK)の立下りエッジで受信データをサンプリングします。

②受信動作を継続する場合、受信割込み要求後から次のシリアルクロック(SCK)の立上りまでに TDR にダミーデータを書き込んでください。

③受信動作から送信動作へ切え換える場合、受信割込み要求後から次のシリアルクロック(SCK)の立上りまでにシリアルデータ出力許可(SMR:SOE=1), 受信割込み禁止(SCR:RIE=0)および受信動作禁止(SCR:RXE=0)にし、TDR に送信データを書込みと受信動作終了後にシリアルクロックの立上りエッジに同期して送信データを出力します。

40.6.2.5 ボーレートの生成

ボーレートの生成について説明します。

専用ボーレートジェネレータは、マスタ動作時のみ機能します。ただし、受信 FIFO を使用する場合にはスレーブ動作時でも専用ボーレートジェネレータを設定してください。

専用ボーレートジェネレータの設定は、マスタ動作時とスレーブ動作時では異なります。

[1]マスタ動作時
専用ボーレートジェネレータで内部クロックを分周させて、ボーレートを選択します。

- 2 つの内部リロードカウンタがあり、それぞれ送受信シリアルクロックに対応しています。ボーレートジェネレータレジスタ(BGR)で 15 ビットのリロード値を設定することにより、ボーレートを選択できます。
- リロードカウンタは設定された値で内部クロックを分周します。

[2]スレーブ動作時
- スレーブ動作時(SCR : MS=1)は、専用ボーレートジェネレータは機能しません。 (クロック入力端子(SCK)から入力された、外部クロックを直接使用します。)

＜注意事項＞
受信 FIFO を使用する場合にはスレーブ動作時でも専用ボーレートジェネレータを設定してください。

■ ボーレートの計算

2 つの 15 ビットリロードカウンタは、ボーレートジェネレータレジスタ(BGR)で設定します。ボーレートの計算式を以下に示します。

(1) リロード値

$$V = \Phi / b - 1$$

V : リロード値　Φ : 周辺クロック(PCLK)周波数　b : ボーレート

(2) 計算例

周辺クロック(PCLK)16MHz, 内部クロック使用, ボーレート 19200bps に設定するには

リロード値 :

$$V = (16 \times 1{,}000{,}000) / 19200 - 1 = 832$$

よって、ボーレートは

$$b = (16 \times 1{,}000{,}000) / (832 + 1) = 19208 \text{ bps}$$

(3) ボーレートの誤差

ボーレートの誤差は次の式によって求められます。

誤差(%) ＝ (計算値－目標値) / 目標値 × 100

＜注意事項＞

- リロード値を"0"に設定するとリロードカウンタは停止します。
- リロード値が偶数の場合、シリアルクロックの"H"幅と"L"幅は SCINV ビットの設定によって以下のようになります。奇数の場合、シリアルクロックの"H"幅と"L"幅は同じになります。
  - SCINV=0 のとき、シリアルクロックの"H"幅が周辺クロック(PCLK)1 サイクル分長くなります。
  - SCINV=1 のとき、シリアルクロックの"L"幅が周辺クロック(PCLK)1 サイクル分長くなります。
- リロード値は 3 以上を設定してください。

■ リロードカウンタの機能

リロードカウンタには、送信リロードカウンタと受信リロードカウンタがあり、専用ボーレートジェネレータとして機能します。リロード値に対する 15 ビットレジスタから構成されており、内部クロックより送受信クロックを生成します。

■ カウントの開始

ボーレートジェネレータレジスタ(BGR)にリロード値を書き込むと、リロードカウンタはカウントを開始します。

■ 再スタート

リロードカウンタは下記の条件で再スタートします。

- 送信/受信リロードカウンタ共通
  - プログラマブルリセット(SCR:UPCL ビット)

40.6.3　設定手順とプログラムフロー

設定手順とプログラムフローについて示します。

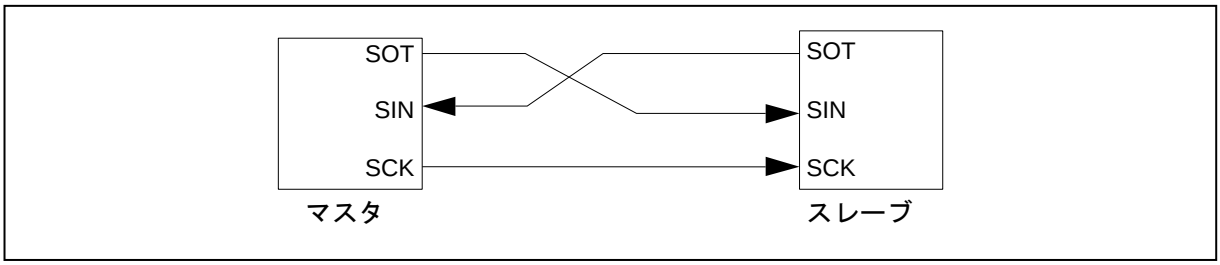
40.6.3.1 チップ間接続

40.6.3.2 フローチャート

40.6.3.1　チップ間接続

チップ間接続について示します。

図 40-28 CSIO チップ間接続例



40.6.3.2　フローチャート

フローチャートについて示します。

図 40-29 フローチャート例 (FIFO 未使用時)

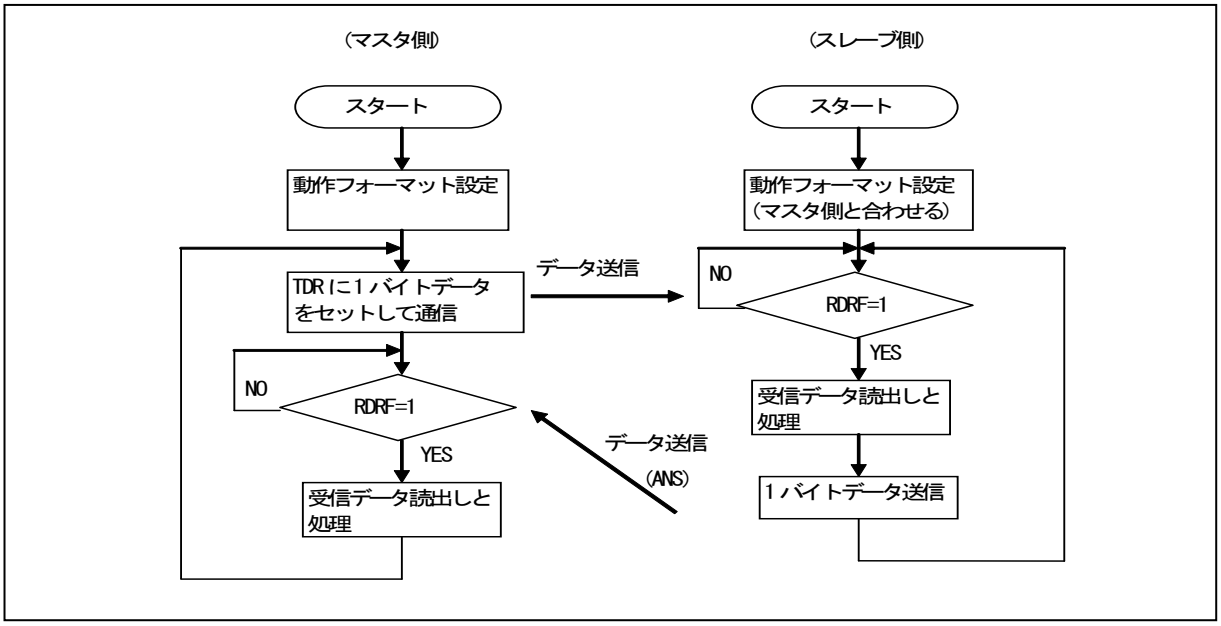
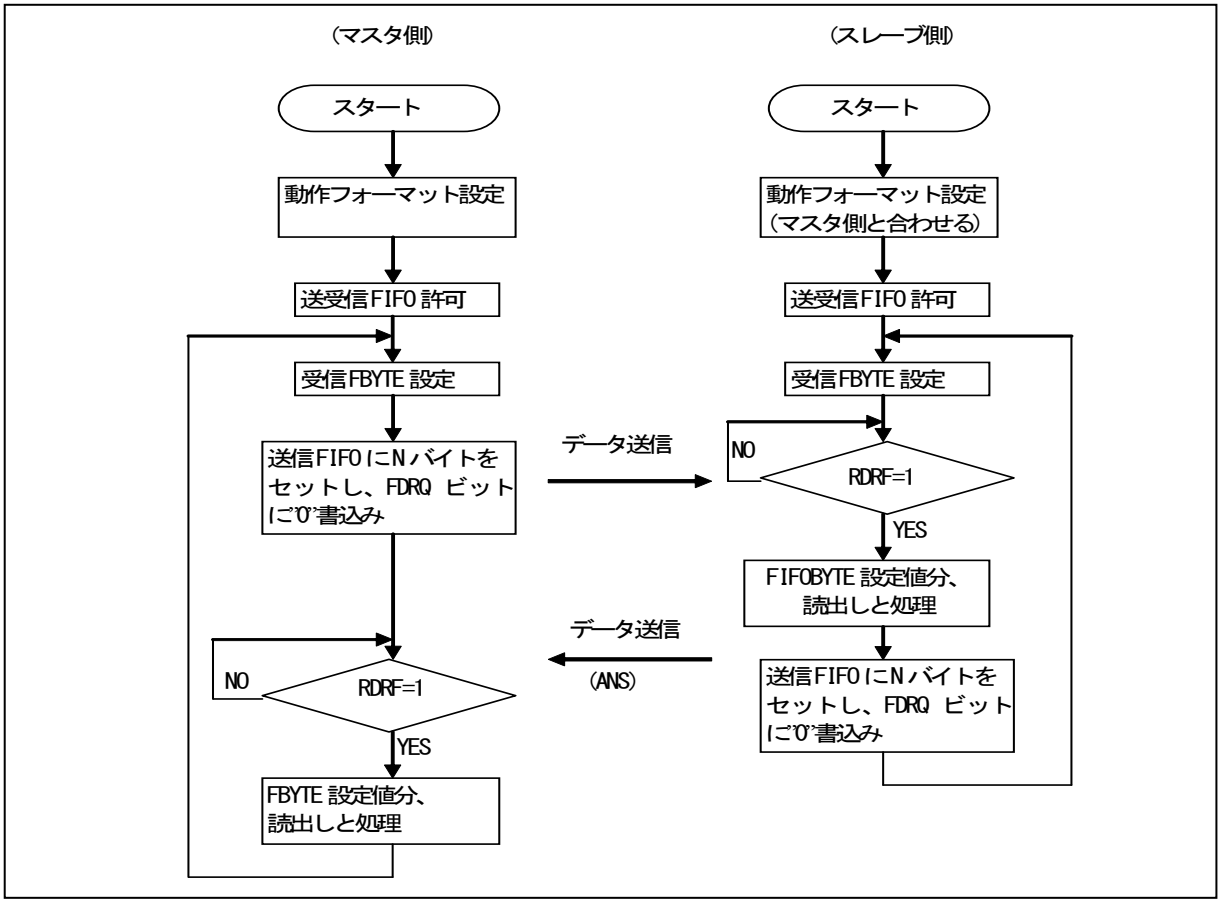


図 40-30 フローチャート例 (FIFO 使用時)



40.7 LIN-UART の動作説明

LIN-UART の動作について示します。

40.7.1 LIN-UART の割込み

40.7.2 LIN-UART の動作

40.7.3 設定手順とプログラムフロー

40.7.1 LIN-UART の割込み

LIN-UART の割込みについて説明します。

LIN-UART の割込みは、次に示す要因で割込み要求を発生させることができます。

- 受信データが受信データレジスタ(RDR)にセットされた場合、または受信エラーが発生した場合
- 送信データが送信データレジスタ(TDR)から送信用シフトレジスタに転送され、送信が開始された場合
- 送信バスアイドル(送信動作なし)
- 送信 FIFO データ要求
- LIN Synch Break 検出

40.7.1.1 LIN-UART インタフェースの割込み一覧

LIN-UART インタフェースの割込み一覧について示します。

LIN-UART の割込み制御ビットと割込み要因は次のようになっています。

表 40-12 LIN-UART の割込み制御ビットと割込み要因

割込みの種類	割込み要求フラグビット	フラグレジスタ	割込み要因	割込み要因許可ビット	割込み要求のクリア方法
受信	RDRF	SSR	1 バイト受信	SCR:RIE	受信データ(RDR)の読出し
			FBYTE 設定値分受信		受信 FIFO がエンプティになるまでの受信データ(RDR)の読出し
			FRIIE ビットが"1"で受信 FIFO に有効なデータが存在した状態で 8 ビット時間以上の受信アイドル状態検出		
	ORE	SSR	オーバランエラー		受信エラーフラグクリアビット(SSR:REC)への"1"書込み
	FRE	SSR	フレーミングエラー		
送信	TDRE	SSR	送信レジスタがエンプティ	SCR:TIE	送信データ(TDR)への書込み、または送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送) *1
	TBI	SSR	送信動作なし	SCR:TBIE	送信データ(TDR)への書込み、LIN Synch Break 設定ビット(LBR)への"1"書込み、または送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送) *1
	FDRQ	FCR1	送信 FIFO がエンプティ	FCR1:FTIE	FIFO 送信データ要求ビット(FCR1:FDRQ)への"0"書込み、または送信 FIFO がフル
ステータス	LBD	SSR	Lin Synch Break 検出	ESCR:LBIE	SSR : LBD ビットへの"0"書込み
インプットキャプチャ *2	ICP	ICS	Lin Synch Field の 1 回目の立下りエッジ	ICS:ICE	ICP をディセーブル
	ICP	ICS	Lin Synch Field の 5 回目の立下りエッジ		

*1 : TDRE ビットが"0"になってから TIE ビットをセットしてください。
*2 : レジスタについては『インプットキャプチャ』の章を参照してください。

＜注意事項＞

- ステータス割込みによる DMA 転送はサポートしていません。
- LIN synch break 検出を行う際には、LIN synch break 検出割込みを許可(LBIE=1)に設定した後、受信禁止(SCR:RXE=0)に設定してください。

40.7.1.2 受信割込み発生とフラグセットのタイミング

受信割込み発生とフラグセットのタイミングについて説明します。

受信時の割込みとしては、受信完了(SSR:RDRF), 受信エラーの発生(SSR:ORE, FRE), および LIN Synch Break 検出があります。

■ 受信割込み発生とフラグセットのタイミング

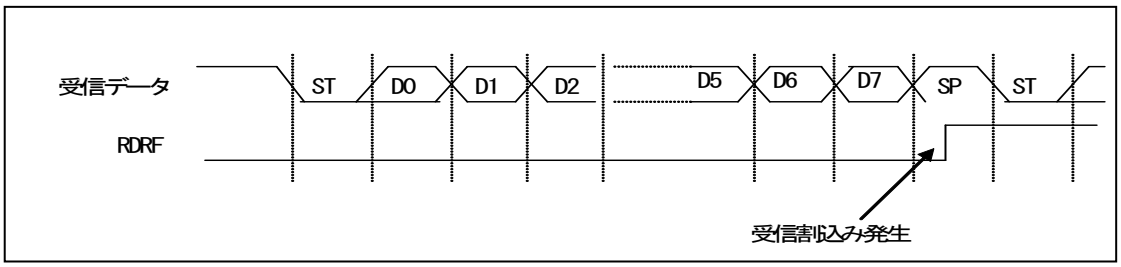
最初のストップビットの検出されることにより、受信データが受信データレジスタ(RDR)に格納されます。受信が完了したとき(SSR:RDRF=1)または受信エラーが発生(SSR:ORE, FRE=1)すると各フラグがセットされます。そのとき、受信割込みが許可(SCR:RIE=1)されていると受信割込みが発生します。

＜注意事項＞

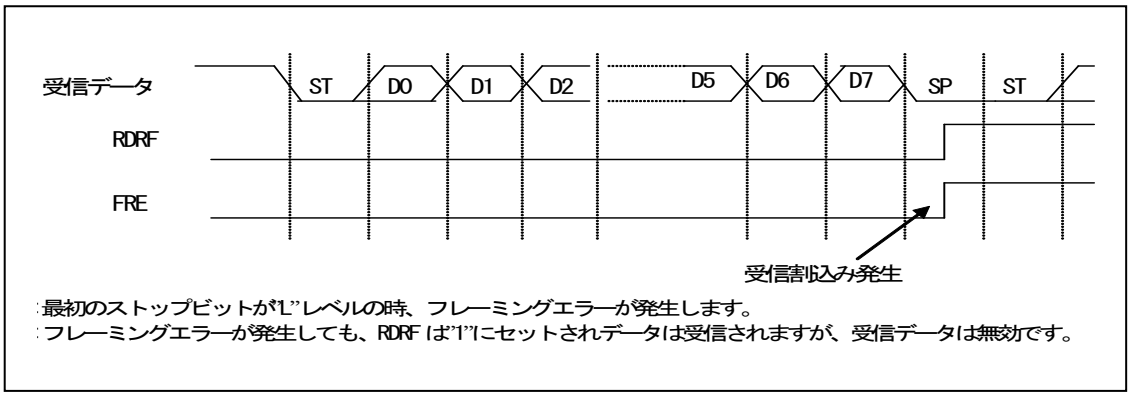
受信エラーが発生した場合は、受信データレジスタ(RDR)のデータは無効となります。

図 40-31 各フラグビットのセットのタイミング

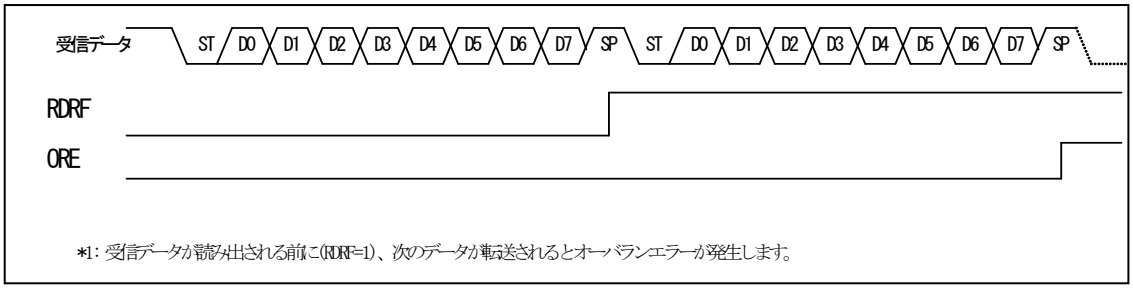
RDRF(受信データフル)フラグビットのセットタイミング



FRE (フレーミングエラー)フラグビットのセットタイミング



ORE (オーバランエラー)フラグビットのセットタイミング

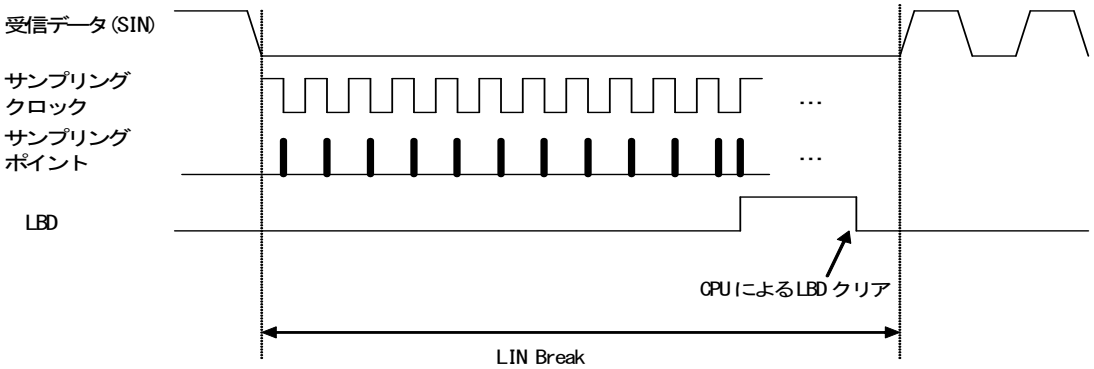


(注意事項) 受信時、ストップビットのサンプリングポイントと同時または 1～2 周辺クロック前にシリアルデータの立下りエッジを検出するとそのエッジを無効にしたりして受信できなくなります。連続してフレームを出力する場合にはフレームの間隔を空けることを推奨します。

■ LIN Synch Break 検出フラグ (LBD) のセットタイミング

スレーブ動作時(SCR:MS=1)に、シリアル入力(SIN)が 11 ビット幅以上"0"入力されると、LBD ビットは"1"にセットされます。このとき、LIN Synch Break 割込みが許可(ESCR:LBIE=1)されていると受信割込みを発生します。

図 40-32 LBD (LIN Synch Break 検出) フラグセットタイミング



受信データをサンプリングクロックの立下りで 11 ビット"L"を検出後、サンプリングクロックの立上りでLIN Break を検出する。LIN Break 検出時にLBD は"1"にセットされる。

40.7.1.3 受信 FIFO 使用時の割込み発生とフラグセットのタイミング

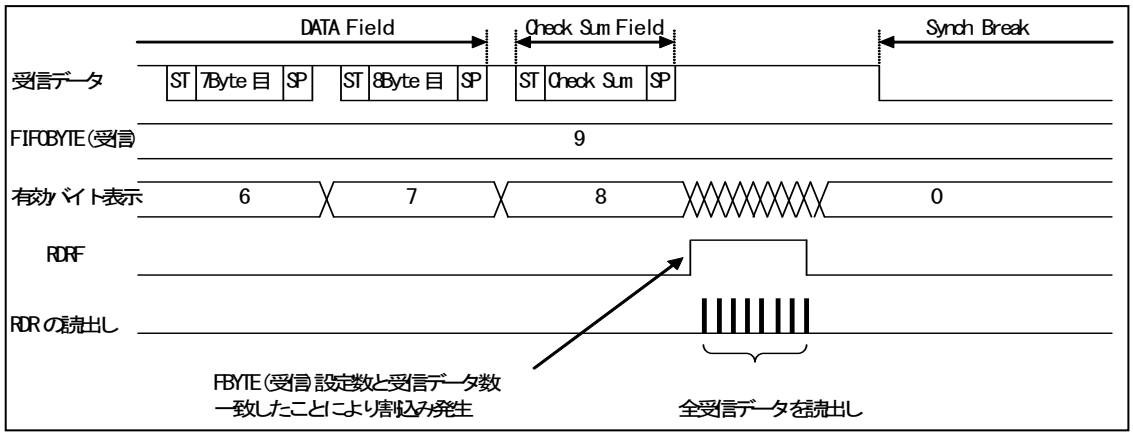
受信 FIFO 使用時の割込み発生とフラグセットのタイミングについて説明します。

受信 FIFO 使用時の割込みは、FBYTE レジスタ(FBYTE)の設定値分受信すると発生します。
受信 FIFO 使用時の割込み発生は、FBYTE レジスタの設定値によって決定されます。

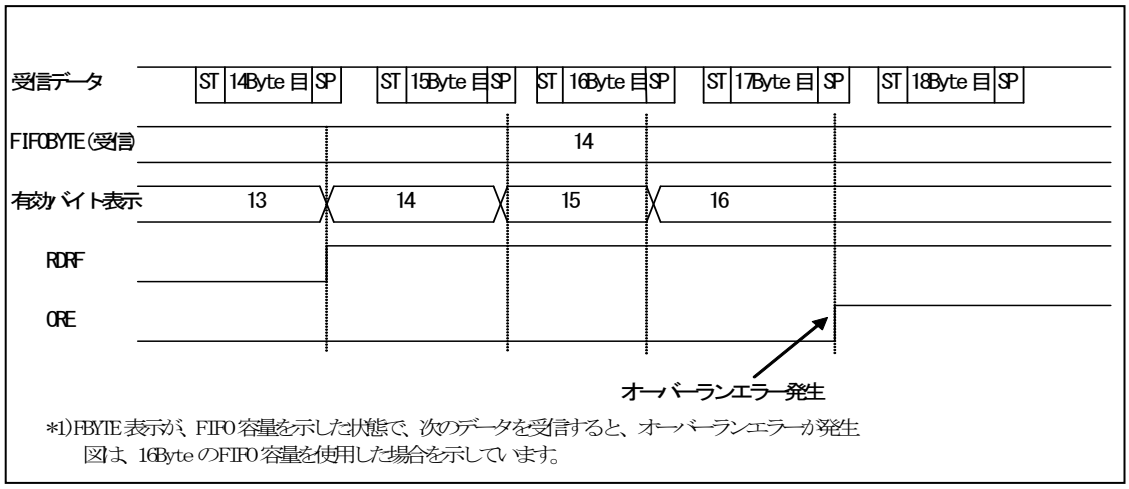
- FBYTE レジスタの転送数設定分のデータを受信するとシリアルステータスレジスタの受信データフルフラグ(SSR:RDRF)が"1"にセットされます。このとき、受信割込み許可(SCR : RIE)されていると受信割込みを発生します。
- 受信 FIFO アイドル検出許可ビット(FRIIE)が"1"で受信 FIFO に存在するデータ数が転送数に達しない場合、受信アイドル状態がボーレートクロックで 8 クロック以上続くと割込みフラグ(RDRF)が"1"にセットされます。8 クロックカウント中、RDR を読み出すとそのカウンタは 0 にリセットされ、再度 8 クロックをカウントします。受信 FIFO が禁止されるとそのカウンタは 0 にリセットされます。受信 FIFO にデータが残っている状態で受信 FIFO を許可すると再度、カウントを開始します。
- 受信 FIFO がエンプティになるまで受信データ(RDR)を読み出すと、受信データフルフラグ(SSR:RDRF)はクリアされます。
- 受信有効データ数表示が FIFO 容量を示した状態で次のデータを受信するとオーバランエラー(SSR:ORE=1)が発生します。

図 40-33 割込み発生タイミング

受信 FIFO 使用時の受信割込み発生タイミング



ORE (オーバランエラー)フラグビットのセットタイミング



*1)FBYTE 表示が、FIFO 容量を示した状態で、次のデータを受信すると、オーバーランエラーが発生
図は、16Byte の FIFO 容量を使用した場合を示しています。

40.7.1.4 送信割込み発生とフラグセットのタイミング

送信割込み発生とフラグセットのタイミングについて説明します。

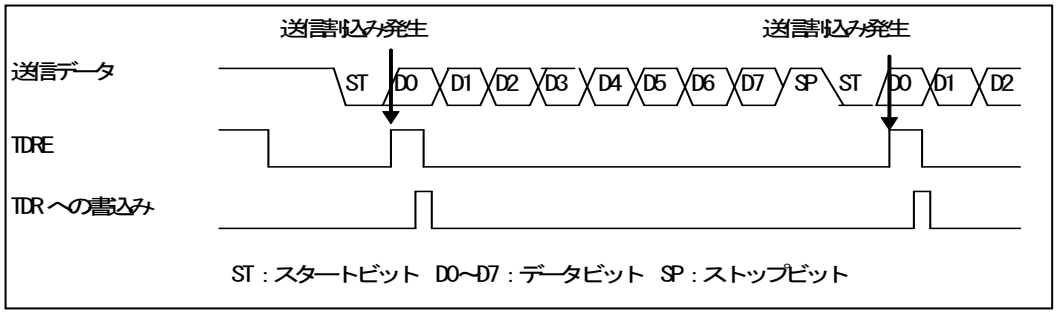
送信時の割込みとしては、送信データが、送信データレジスタ(TDR)から送信用シフトレジスタに転送され(SSR:TDRE=1)送信が開始された場合と、送信動作をしていないとき(SSR:TBI=1)に発生します。

- 送信データエンプティフラグ (SSR:TDRE) のセットタイミング
  送信データレジスタ(TDR)に書き込まれたデータが送信シフトレジスタに転送されると、次のデータの書込みが可能な状態(SSR:TDRE=1)になります。そのとき、送信割込みが許可(SCR: TIE=1)されていると、送信割込みが発生します。SSR:TDRE ビットはリードオンリビットなので、送信データレジスタ(TDR)へのデータ書込みにより"0"にクリアされます。

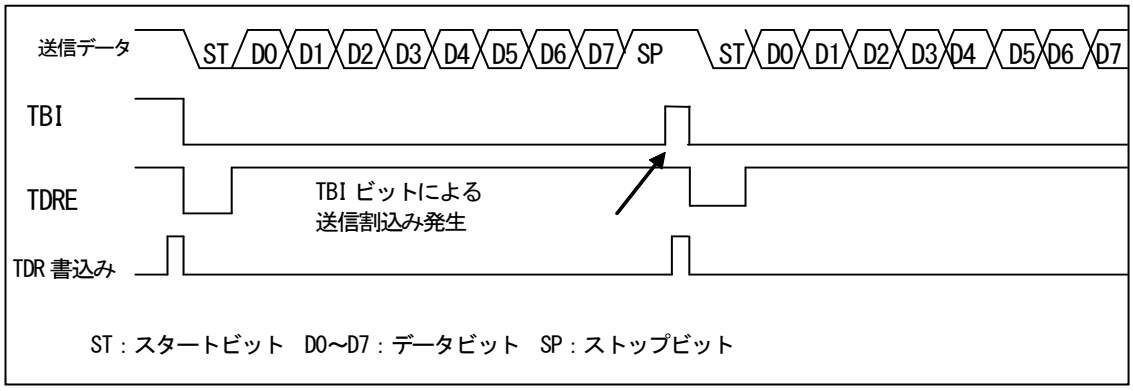
- 送信バスアイドルフラグ(TBI)のセットタイミング
  送信データレジスタがエンプティ(TDRE=1)で送信動作をしていないとき、SSR:TBI ビットは"1"にセットされます。このとき、送信バスアイドル割込み許可(SCR:TBIE=1)されていると、送信割込みが発生します。送信データレジスタ(TDR)に送信データをセッ トすると TBI ビットおよび送信割込み要求はクリアされます。

図 40-34 TDRE, TBI のセットタイミング

送信データエンプティフラグ(TDRE) のセットタイミング



送信バスアイドルフラグ(TBI) のセットタイミング



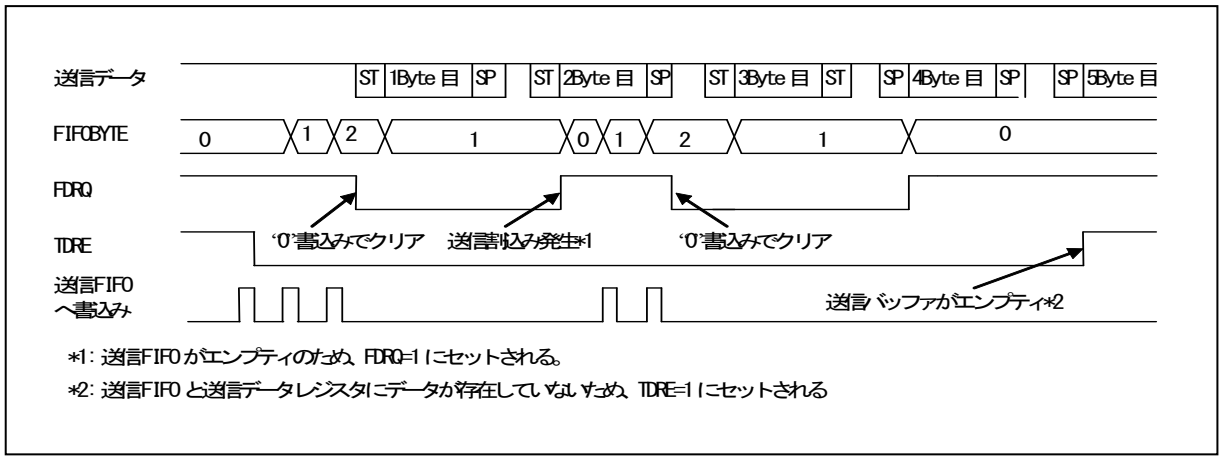
40.7.1.5 送信 FIFO 使用時の割込み発生とフラグセットのタイミング

送信 FIFO 使用時の割込み発生とフラグセットのタイミングについて説明します。

送信 FIFO 使用時の割込みは、送信 FIFO にデータが存在しないときに発生します。

- 送信 FIFO にデータが存在しない場合、FIFO 送信データ要求ビット(FCR1:FDRQ)が"1"にセットされます。このとき、FIFO 送信割込み許可(FCR1:FTIE=1)されていると送信割込みが発生します。
- 送信割込みが発生して送信 FIFO に必要なデータを書き込んだら、FIFO 送信データ要求ビット(FCR1:FDRQ)に"0"書き込んで割込み要求をクリアしてください。
- 送信 FIFO がフルになると FIFO 送信データ要求ビット(FCR1:FDRQ)は"0"になります。
- 送信 FIFO のデータの存在の確認は、FIFO バイトレジスタ(FBYTE)を読み出すことで確認できます。FBYTE=00_H のときは、送信 FIFO にデータが存在していないことを示します。

図 40-35 送信 FIFO 使用時の送信割込み発生タイミング



40.7.2 LIN-UART の動作

LIN-UART の動作について示します。

LIN-UART は、マスタ/スレーブ双方向 LIN 通信で動作します。

40.7.2.1 マスタデバイス動作

マスタデバイス動作について説明します。

■ デバイスの選択

マスタデバイスとして動作させるためには、SCR:MS ビットを"0"に設定します。

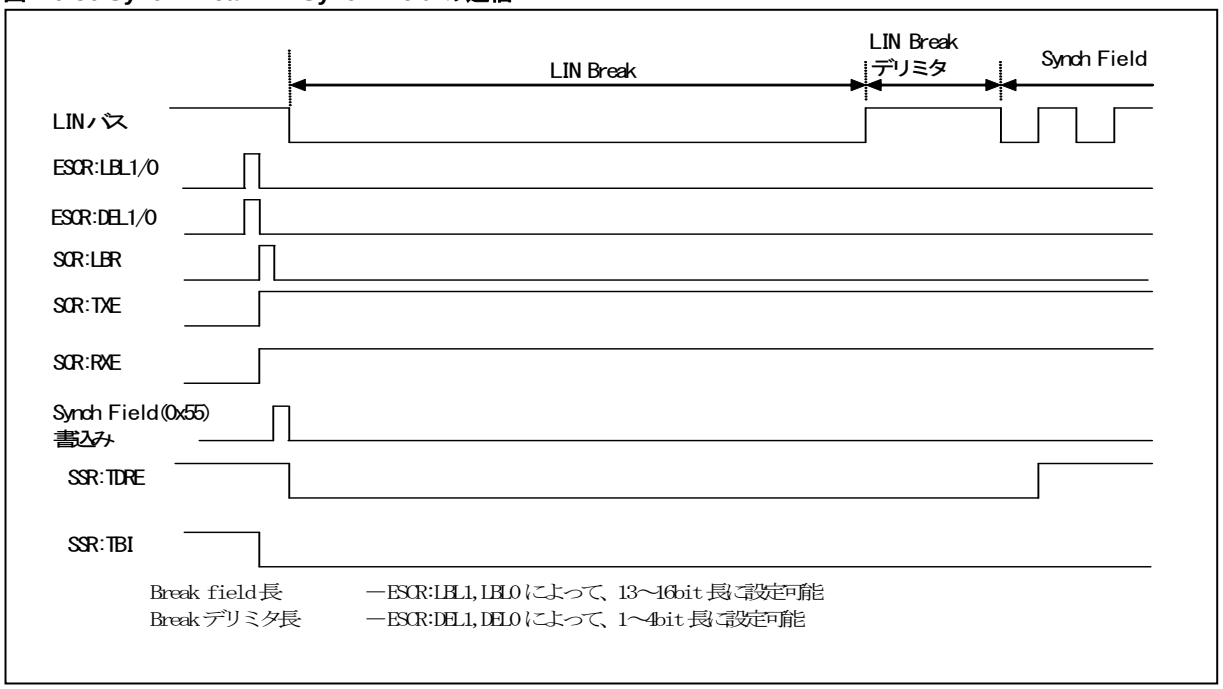
■ Synch Break 送信 ～ Synch Field 送信

- Synch Break 長を選択(ESCR:LBL1, LBL0)および、Synch Break デリミタ長の選択(ESCR:DEL1, DEL0)をします。
- 送信動作許可(SCR:TXE=1)し、SCR:LBR ビット(LIN Synch Break 設定ビット)に"1"を設定すると、Synch Break が送信されます。
- Synch Field は、送信データレジスタ(TDR)に 0x55 を書き込むことで送信されます。

＜注意事項＞

- SCR:LBR ビット(LIN Synch Break 設定ビット)に"1"を設定した後に、送信データレジスタ(TDR)に 0x55 を設定してください。
- SCR:RXE ビット(受信動作許可ビット)に"1"を設定していても Synch Break 部分は受信動作を行いません。

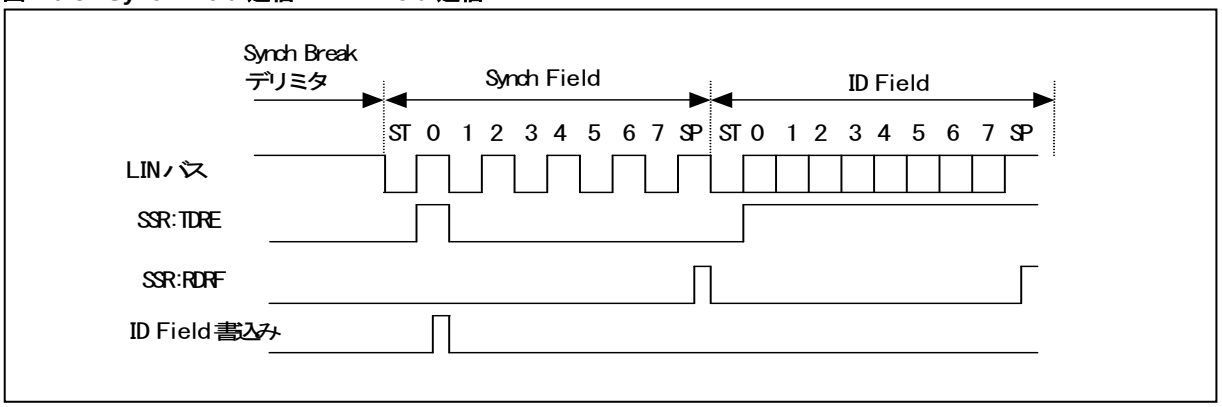
図 40-36 Synch Break ～ Synch Field の送信



■ Synch Field 送信 ～ ID Field 送信

- Synch Field (0x55)の最初の1ビット目が送信されると、SSR:TDRE (送信データエンプティ)ビットが"1"にセットされます。このとき、送信割込み許可(SCR:TIE=1)されていると、送信割込みが発生します。
- 送信割込みが発生したら、ID Field を送信データレジスタ(TDR)に書き込むことができます。
- 受信割込みが発生したら、送信データと受信したデータを比較し、エラーが発生していないことを確認します。
- ID Field は、データ長 8 ビットで、LSB ファーストで出力されます。

図 40-37 Synch Field 送信 ～ ID Field 送信



■ ID Field 送信 ～ DATA Field 送受信

DATA Field をスレーブデバイスに送信するか、受信するかを選択します。

- DATA Field を送信する場合
  ID Field の 1 ビット目が送信されると、SSR:TDRE=1 にセットされます。このとき、DATA Field の書込みが可能です。

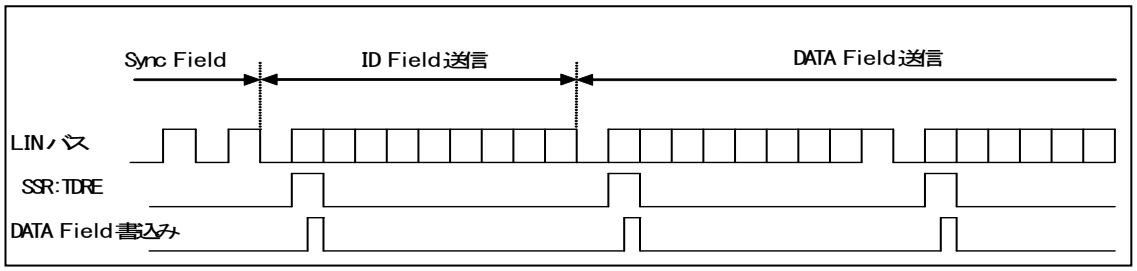
- DATA Field を受信する場合
  ID Field の 1 ビット目が送信されると、SSR : TDRE=1 にセットされますが、送信データを書き込まないでください。 また、送信割込み禁止(SCR:TIE=0)にしてください。

  DATA Field を受信すると、SSR:RDRF が"1"にセットされます。このとき、受信割込み許可(SCR: RIE=1)されていると受信割込みが発生します。

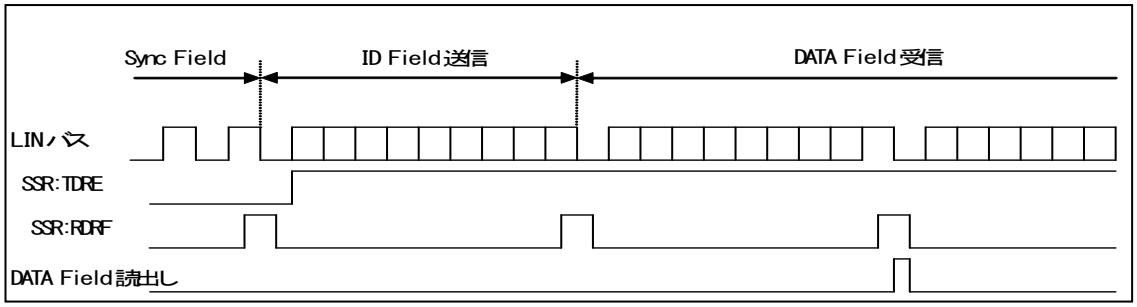
  スタートビットの検出条件は、ノイズフィルタ(シリアルデータ入力を 3 回マシンクロックでサンプリングし、多数決)通過後に立下りを検出し、サンプリングポイントでその通過後のデータが"L"を検出した場合になります。

図 40-38 ID Field 送信～DATA Field 送受信

ID Field 送信～DATA Field 送信



ID Field 送信～DATA Field 受信



＜注意事項＞

- ノイズフィルタ(シリアルデータ入力を3回マシンクロックでサンプリングして多数決)は内蔵していますが、ノイズが本フィルタを通過しないようにボードを設計するか、ノイズが通過して問題にならない(例えば、最後にデータのチェックサムを付加してエラーが発生した場合には再送を行うなど)ように通信を行ってください。
- 受信時、ストップビットのサンプリングポイントと同時または 1～2 マシンクロック前にシリアルデータの立下りエッジを検出するとそのエッジを無効などにしたりして受信できなくなります。連続してフレームを出力する場合にはフレームの間隔を空けることを推奨します。

■ FIFO 未使用時のタイミングチャート

図 40-39 LIN バスタイミング (DATA Field 送信時 : FIFO 未使用時)

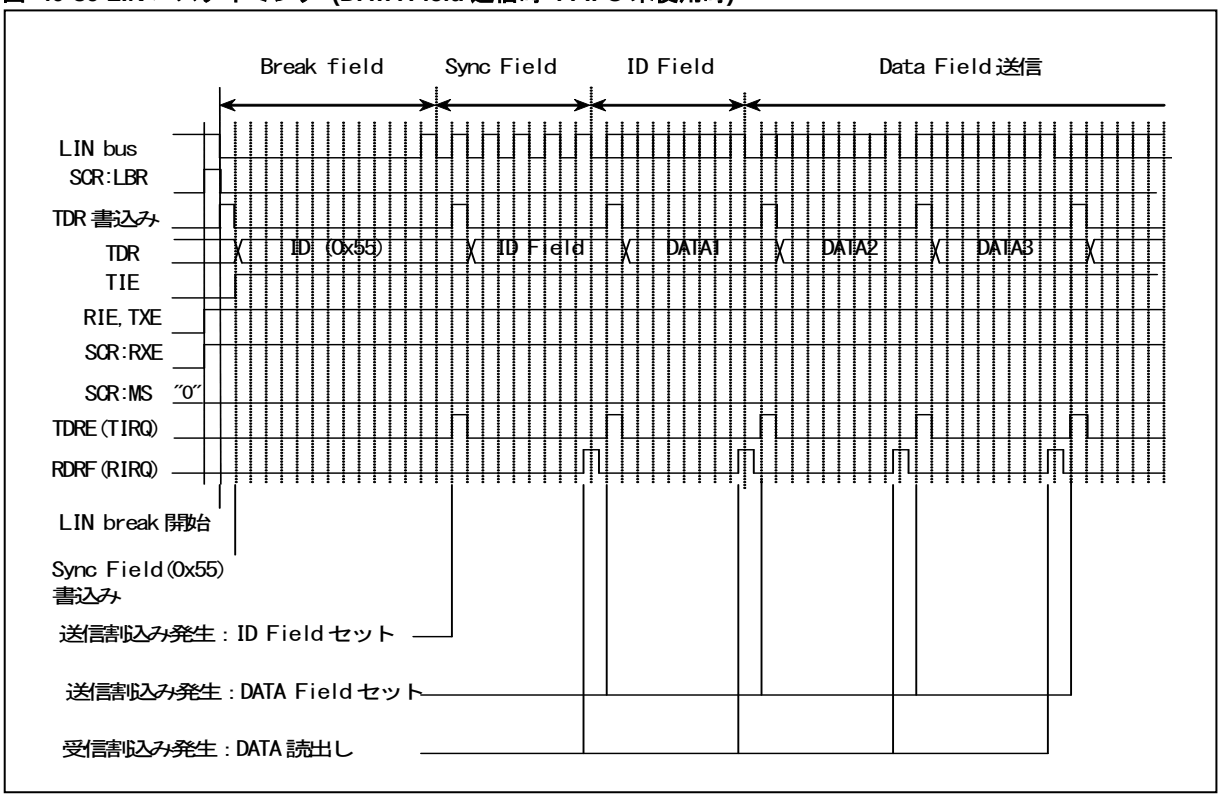
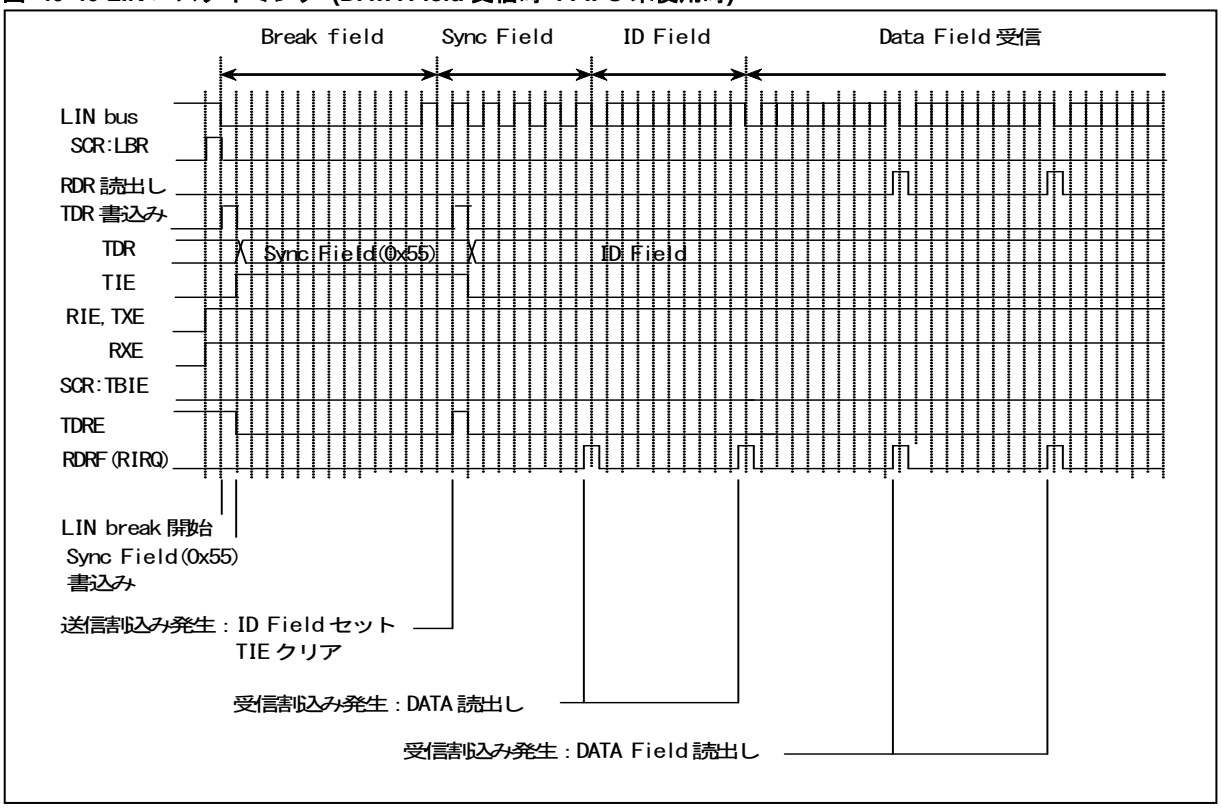


図 40-40 LIN バスタイミング (DATA Field 受信時 : FIFO 未使用時)



■ FIFO 使用時のタイミングチャート

図 40-41 LIN バスタイミング (DATA Field 送信時 : FIFO 使用時)

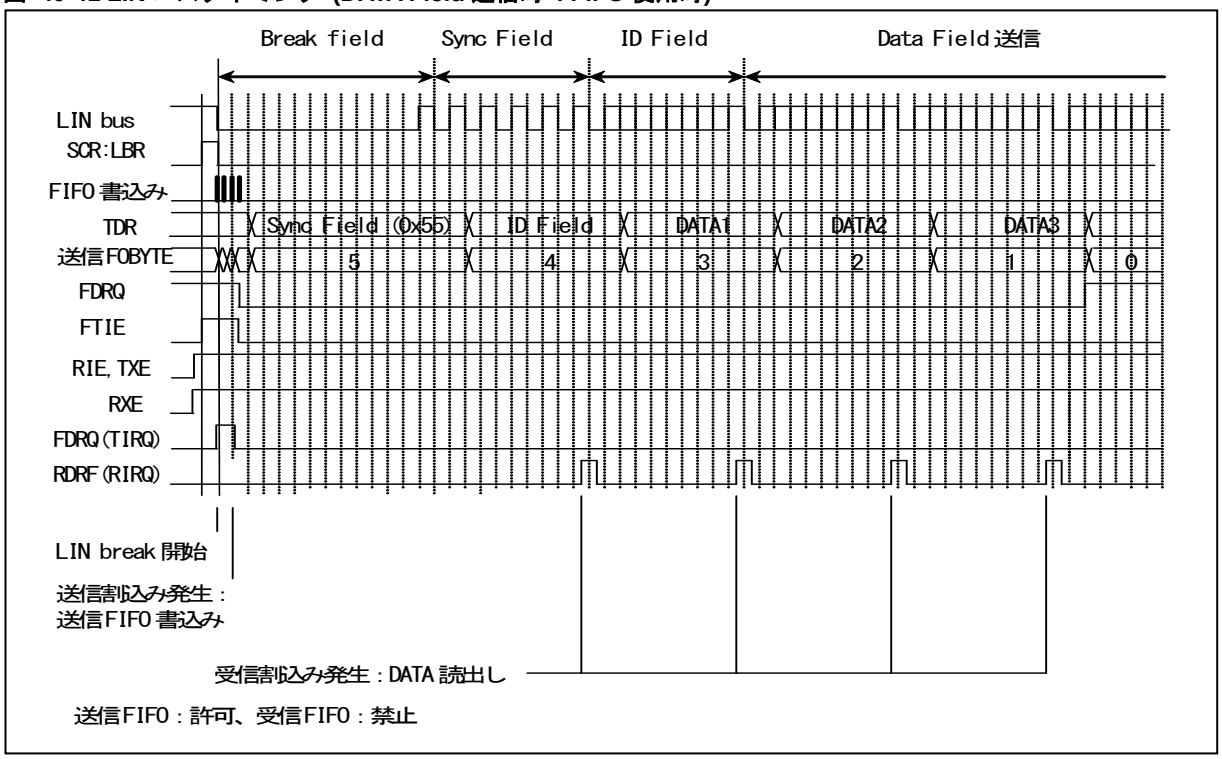
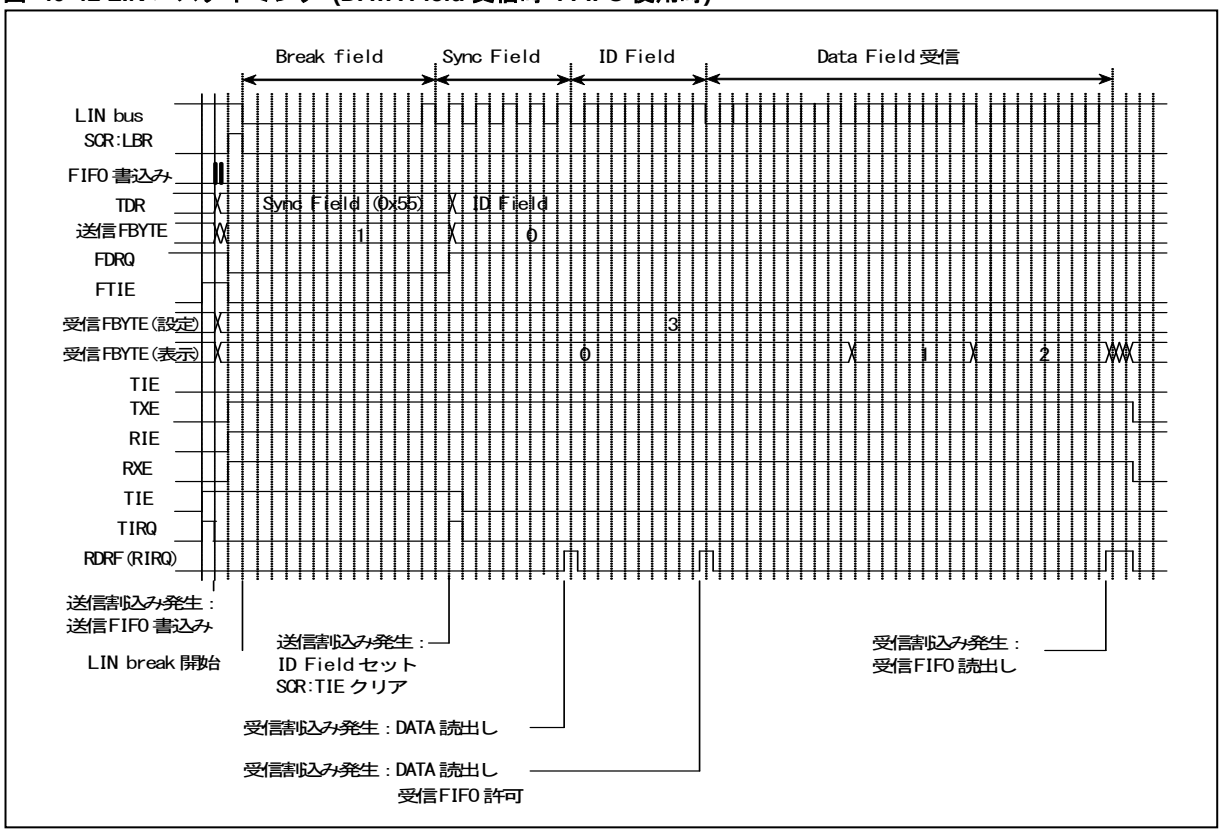


図 40-42 LIN バスタイミング (DATA Field 受信時 : FIFO 使用時)



40.7.2.2 スレーブデバイス動作

スレーブデバイス動作について説明します。

■ デバイスの選択

スレーブデバイスとして動作させるためには、SCR:MS ビットを"1"に設定します。

■ Synch Break Field 受信 ～ Synch Field 受信

①Synch Break が入力されると 11 ビット目で Synch Break 検出(SSR:LBD=1)されます。このとき、ESCR:LBIE ビットが"1"にセットされていると受信割込みを発生します。

②ここで、ICU の割込みを許可し両エッジ検出に設定します。

③Synch field の最初の立下りエッジを LIN-UART が検出すると、ICU に入力される内部信号(LSYN)を"H"にして ICU をスタートさせます。この内部信号(LSYN)は 5 番目の立下りエッジで"L"になります。

④ICU に入力される内部信号(LSYN)が、"H"の時間がボーレートを 8 倍した値になります。ボーレートの設定値は,次のようになります。

フリーランタイマがオーバフローしていない場合 :

$$BGR\ 値 = (b - a) \times Fe / (8 \times \phi) - 1$$

フリーランタイマがオーバフローした場合 :

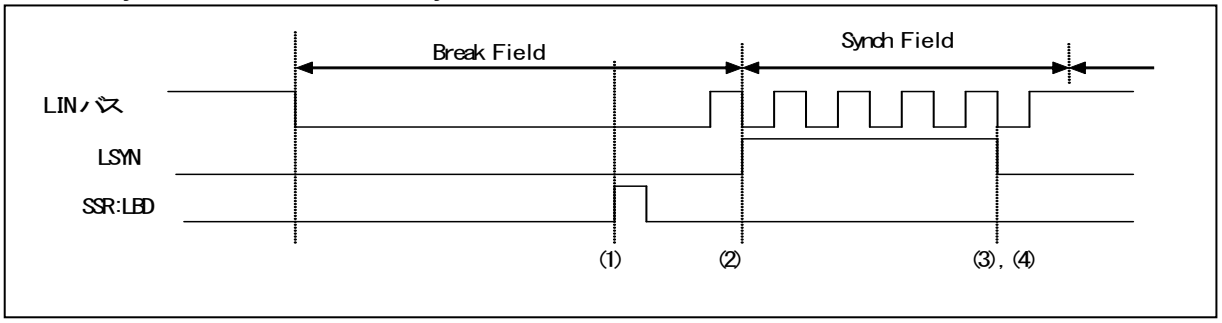
$$BGR\ 値 = (max + 1 + b - a) \times Fe / (8 \times \phi) - 1$$

max : フリーランタイマの最大値
a : 1 度目の割込み後の ICU データレジスタ値
b : 2 度目の割込み後の ICU データレジスタ値
ϕ : 周辺クロック(PCLK)周波数(MHz)
Fe : 外部クロック周波数(MHz)。内部クロック使用時(EXT=0), Fe= ϕとして計算

＜注意事項＞

Synch Break および Synch Field 時は、受信禁止(SCR:RXE=0)に設定してください。

図 40-43 Synch Break Field 受信 ～ Synch Field 受信



■ ID Field 受信 ～ DATA Field 送受信

ID Field を受信した後、マスタデバイスへ DATA Field を送信するか、受信するかを選択します。

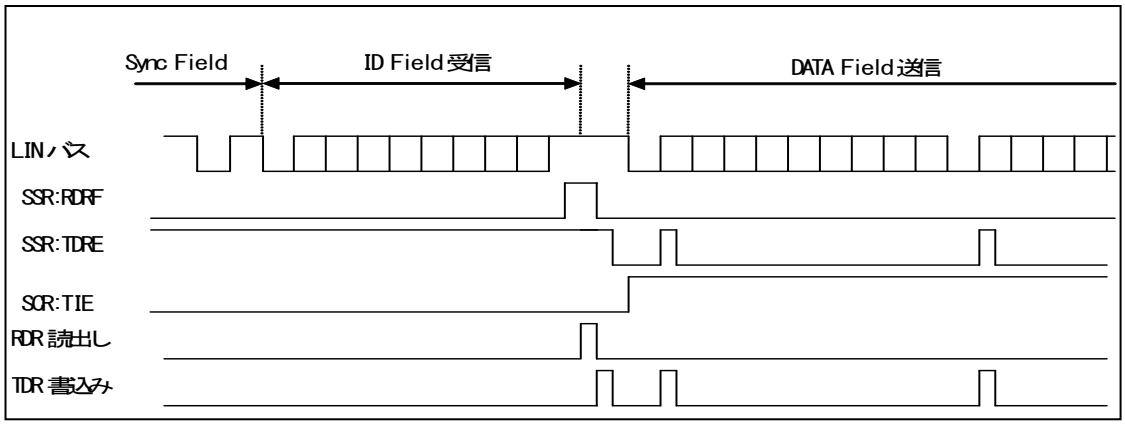
- DATA Field を送信する場合
  ID Field 受信後、送信データレジスタ(TDR)にデータを書き込んでください。このとき、送信割込み許可(SCR:TIE=1)しておきます。

- DATA Field を受信する場合
  DATA Field 受信ごとに、SSR:RDRF が"1"にセットされます。このとき、受信割込み許可(SCR: RDRF=1)されていると受信割込みが発生します。

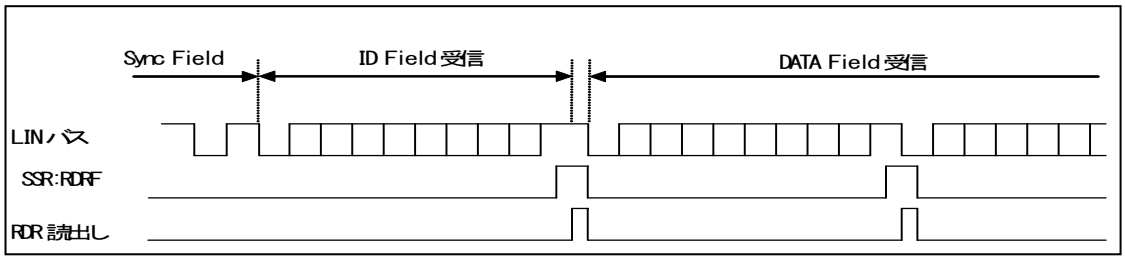
  スタートビットの検出条件は、ノイズフィルタ(シリアルデータ入力を 3 回マシンクロックでサンプリングし、多数決)通過後に立下りを検出し、サンプリングポイントでその通過後のデータが"L"を検出した場合になります。

図 40-44 ID Field 受信 ～ DATA Field 送受信

ID Field 受信 ～ DATA Field 送信



ID Field 受信 ～ DATA Field 受信



＜注意事項＞

- ノイズフィルタ(シリアルデータ入力を3回マシンクロックでサンプリングして多数決)は内蔵していますが、ノイズが本フィルタを通過しないようにボードを設計するか、ノイズが通過して問題にならない(例えば、最後にデータのチェックサムを付加してエラーが発生した場合には再送を行うなど)ように通信を行ってください。
- 受信時、ストップビットのサンプリングポイントと同時または 1～2 マシンクロック前にシリアルデータの立下りエッジを検出するとそのエッジを無効などにしたりして受信できなくなります。連続してフレームを出力する場合にはフレームの間隔を空けることを推奨します。

■ FIFO 未使用時のタイミングチャート

図 40-45 LIN バスタイミング (DATA Field 送信時 : FIFO 未使用時)

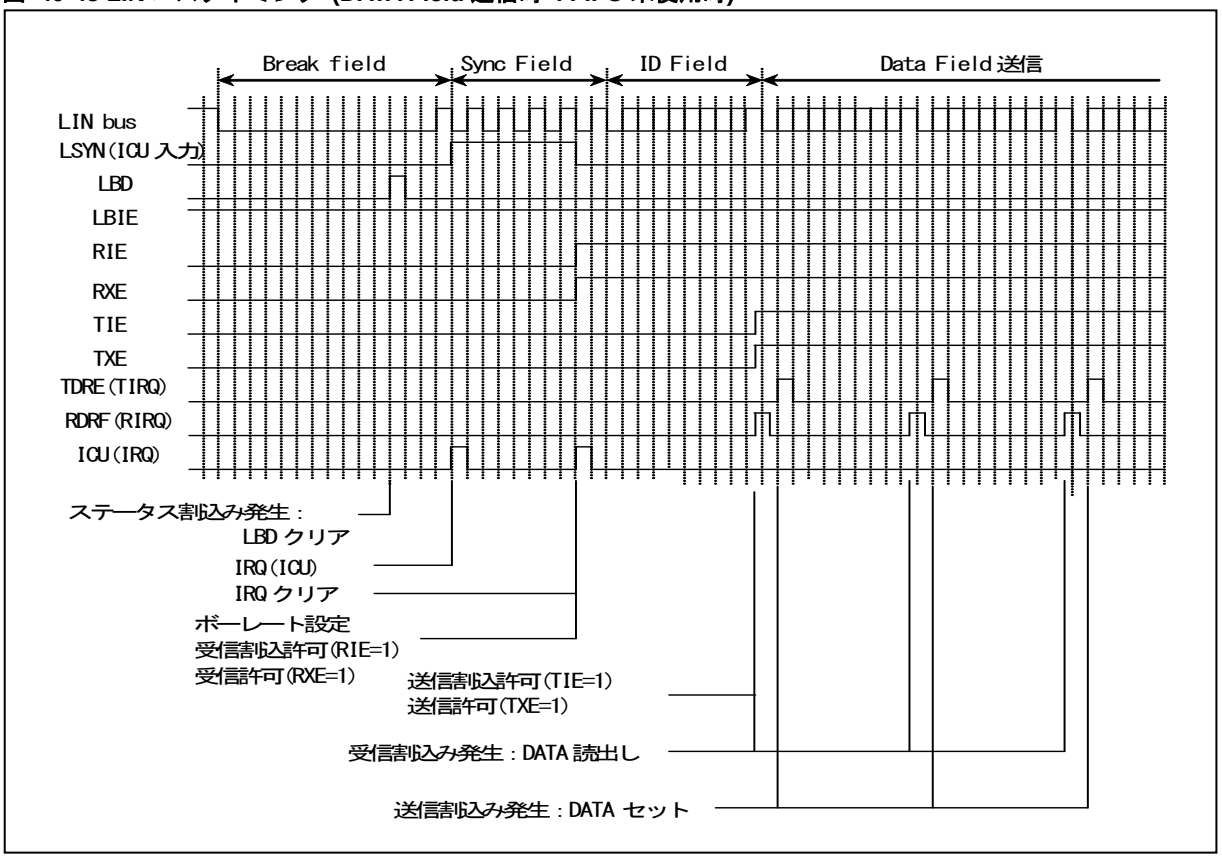
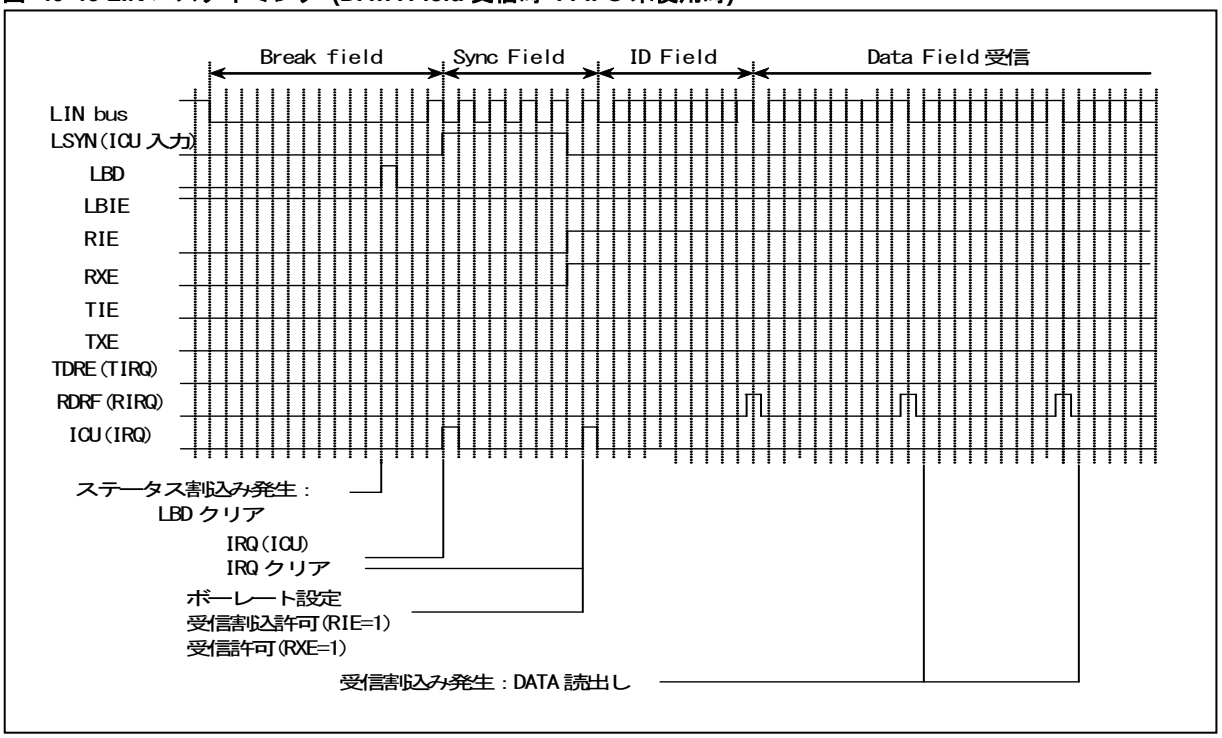


図 40-46 LIN バスタイミング (DATA Field 受信時 : FIFO 未使用時)



■ FIFO 使用時のタイミングチャート

図 40-47 LIN バスタイミング (DATA Field 送信時 : FIFO 使用時)

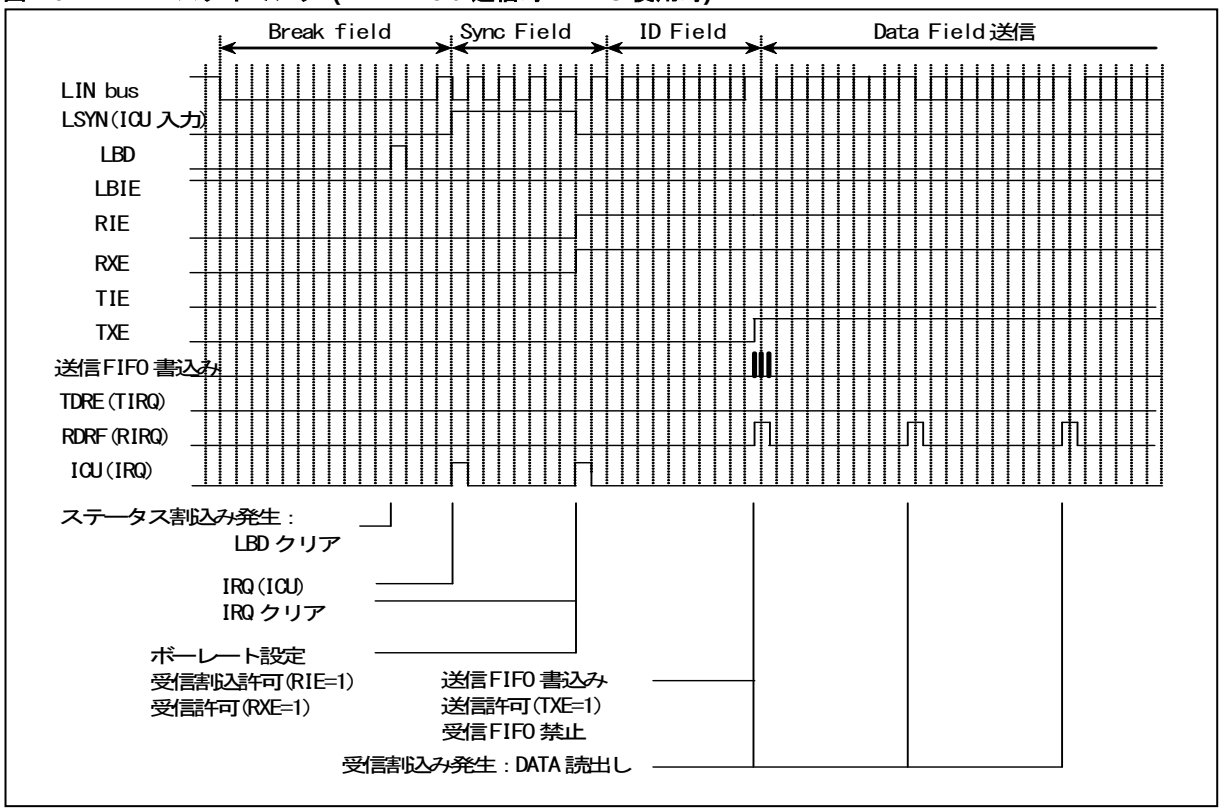
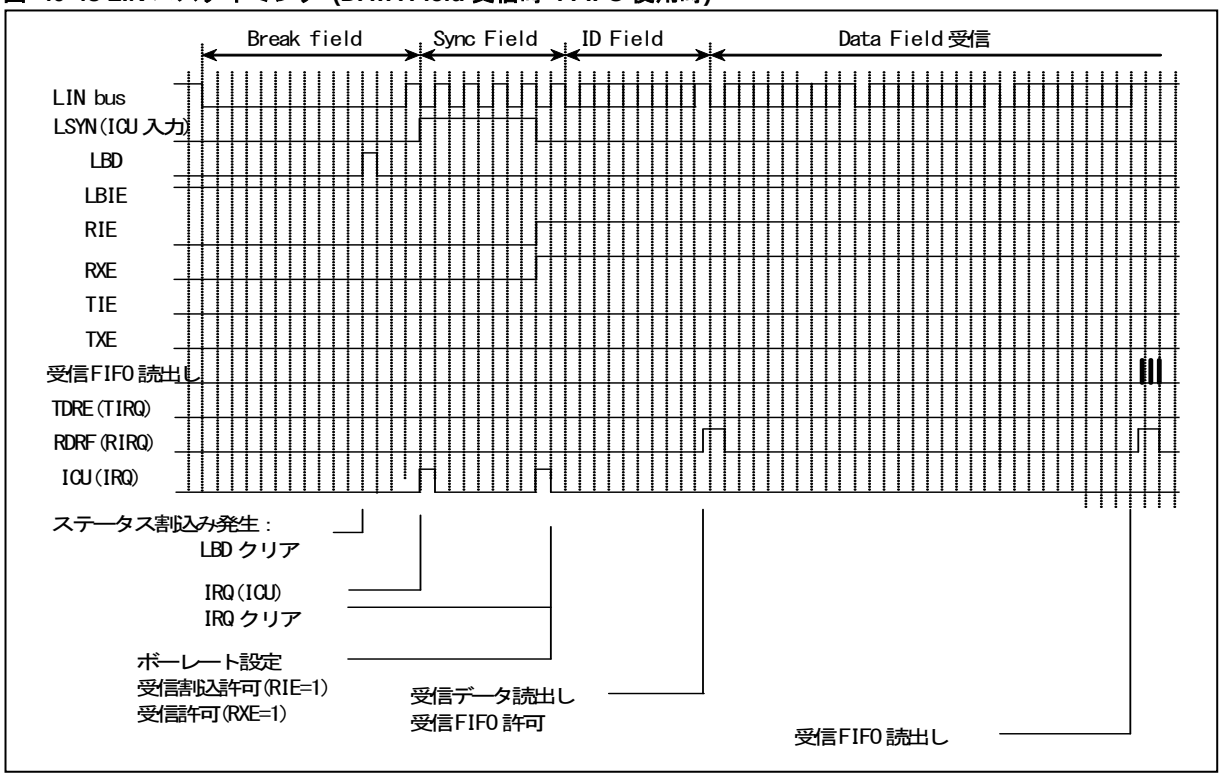


図 40-48 LIN バスタイミング (DATA Field 受信時 : FIFO 使用時)



40.7.2.3 LIN-UART ボーレート選択・設定

LIN-UART ボーレート選択・設定について示します。

LIN-UART では、

- 専用ボーレートジェネレータ(リロードカウンタ)で内部クロックを分周して得られるボーレート
- 専用ボーレートジェネレータ(リロードカウンタ)で外部クロックを分周して得られるボーレート

を使用できます。設定方法は UART 時(モード 0/1)と同一です。「40.5.2.11 UART ボーレート選択・設定」を参照してください。

40.7.3 設定手順とプログラムフロー

設定手順とプログラムフローについて示します。

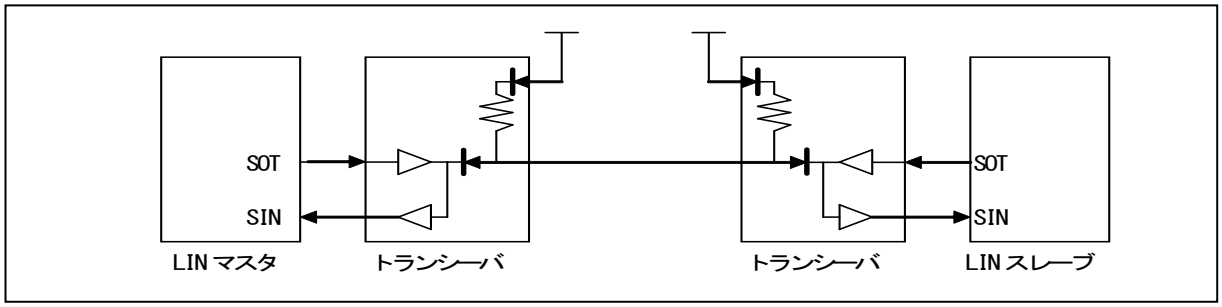
動作モード 3 (LIN 通信モード)では、LIN マスタシステムもしくは、LIN スレーブシステムに使用できます。

40.7.3.1 CPU 間接続

CPU 間接続について示します。

1 つの LIN マスタと LIN スレーブの通信システムを次に示します。マルチファンクションシリアルインタフェースは、LIN マスタまたは、LIN スレーブとして動作することができます。

図 40-49 LIN バスシステムの通信例



40.7.3.2 フローチャート例

フローチャート例について示します。

図 40-50 LIN 通信マスタモードフローチャート例 (FIFO 未使用)

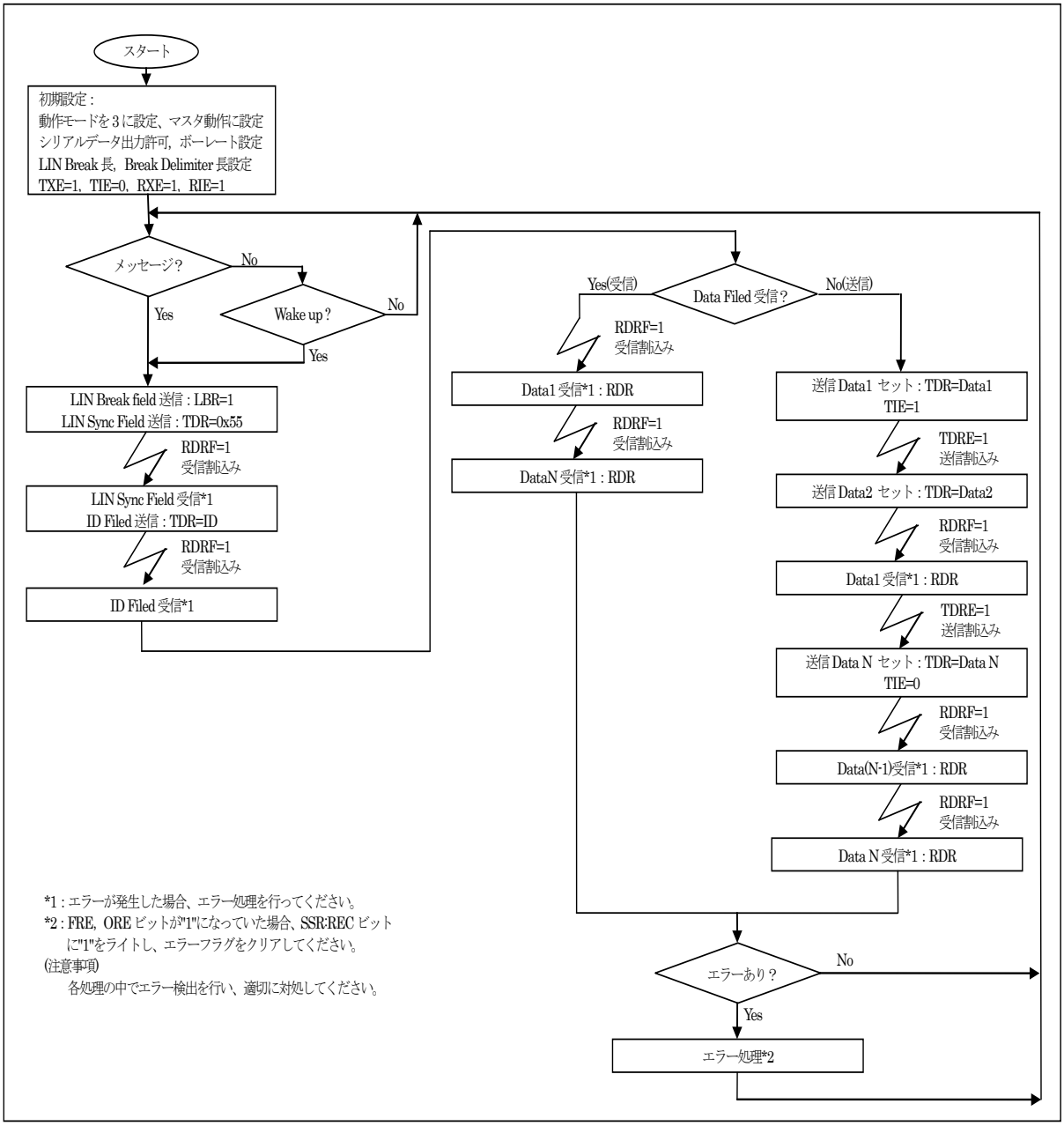
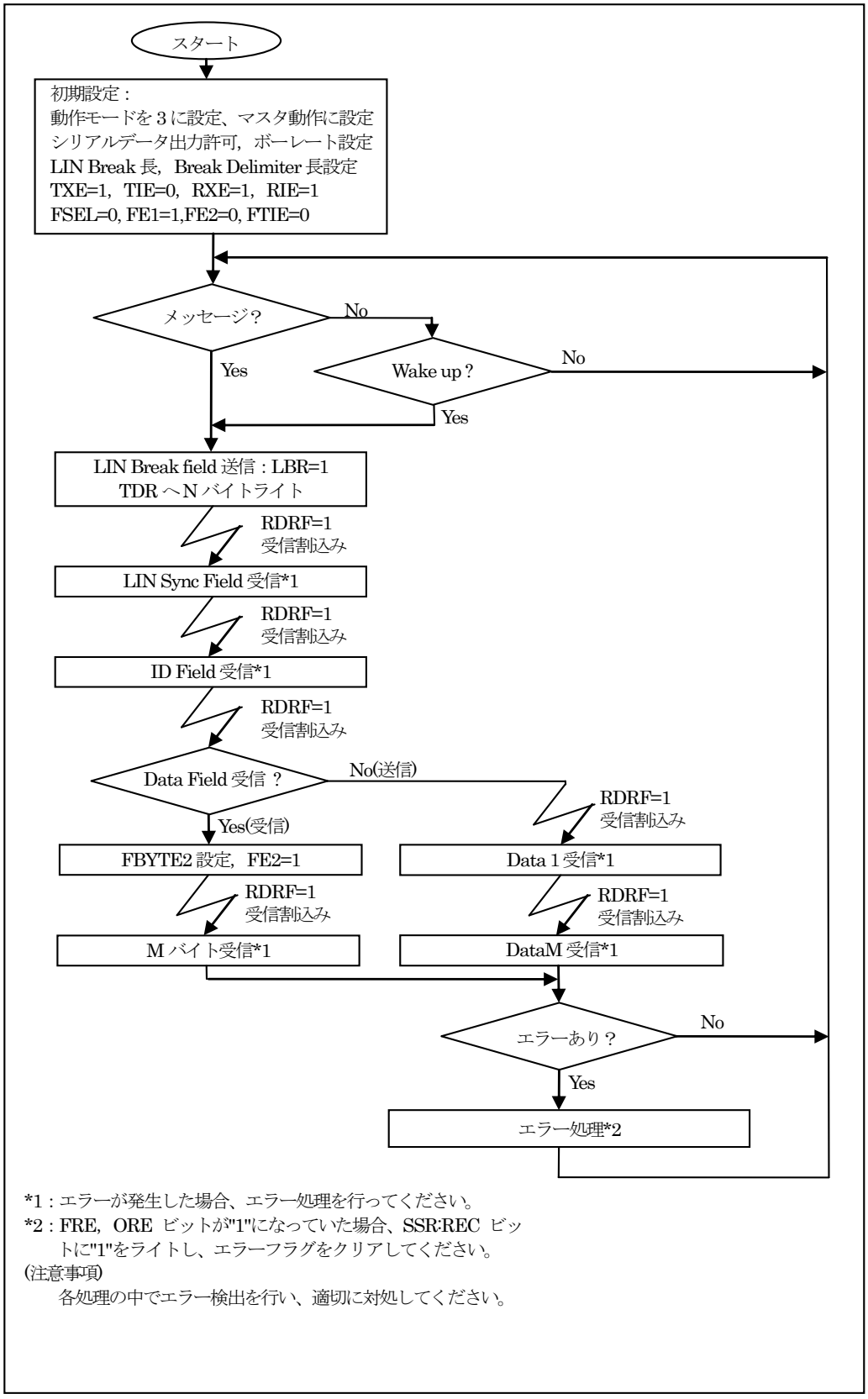


図 40-51 LIN 通信マスタモードフローチャート例 (FIFO 使用)



*1：エラーが発生した場合、エラー処理を行ってください。
*2：FRE，ORE ビットが"1"になっていた場合、SSR:REC ビットに"1"をライトし、エラーフラグをクリアしてください。
(注意事項)
各処理の中でエラー検出を行い、適切に対処してください。

図 40-52 LIN 通信スレーブモードフローチャート例 (FIFO 未使用)

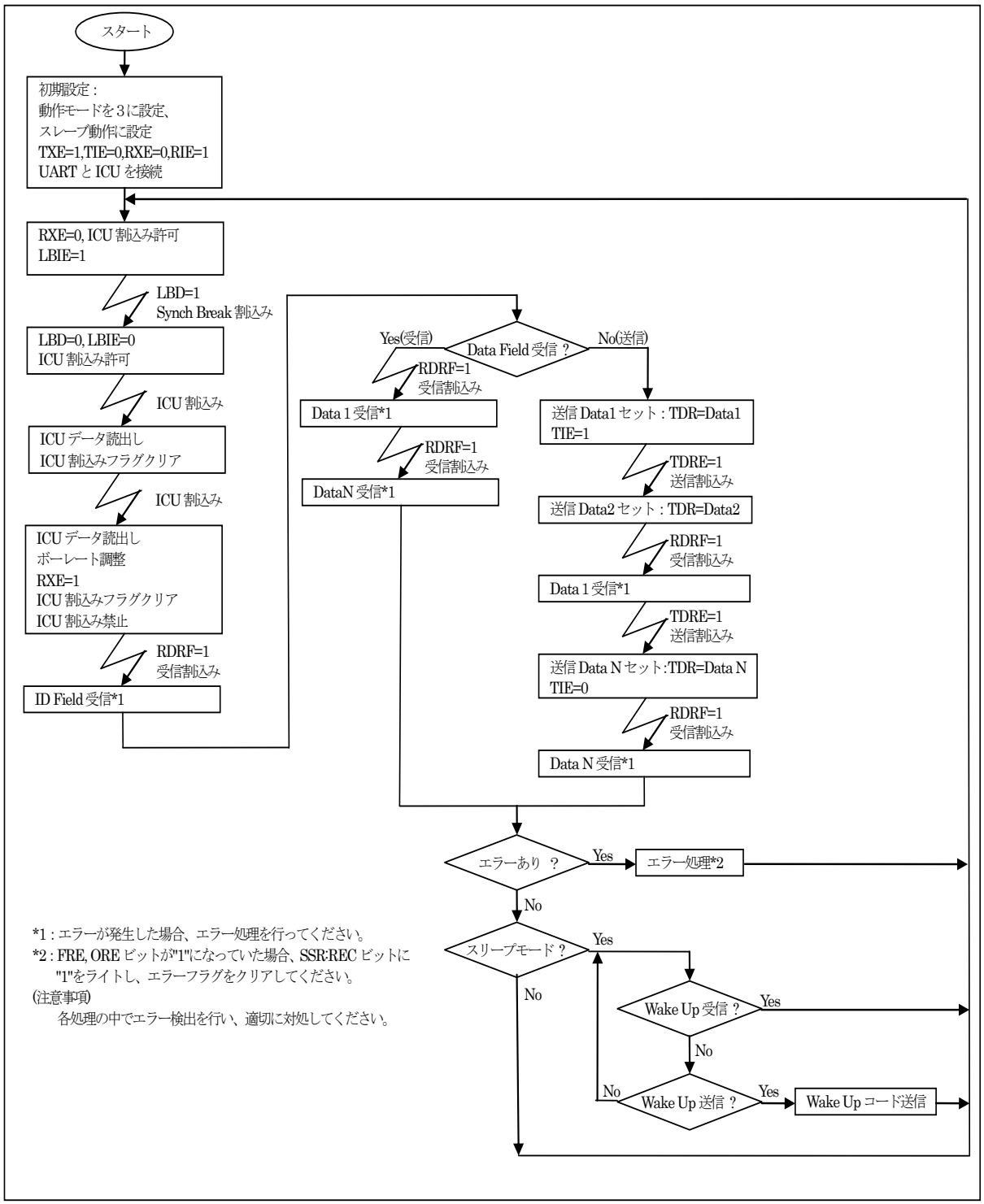
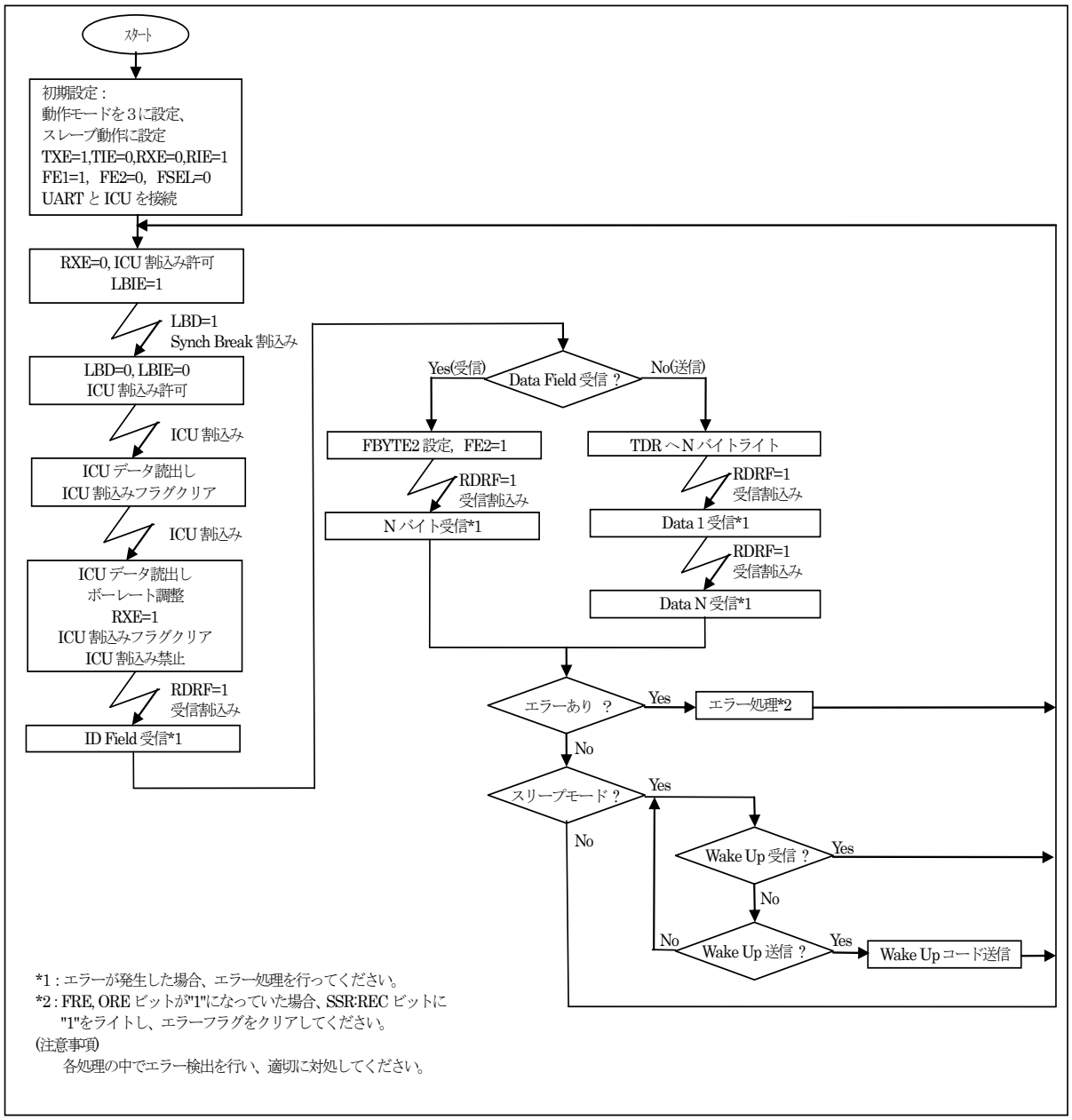


図 40-53 LIN 通信スレーブモードフローチャート例 (FIFO 使用)



40.8 I²C の動作説明

I²C の動作について示します。

40.8.1. I2C の割込み

40.8.2. I2C インタフェース通信の動作

40.8.3. I2C マスタモード

40.8.4. I2C スレーブモード

40.8.5. バスエラー

40.8.6. I2C のフローチャート例

40.8.1 I²C の割込み

I²C の割込みについて説明します。

I²C インタフェースの割込みは、次に示す要因で割込み要求を発生させることができます。

- 第一バイト送受信後/データ送受信後
- ストップ条件
- 反復スタート条件
- FIFO 送信データ要求
- FIFO 受信データ完了

■ I²C インタフェースの割込み一覧

I²C インタフェースの割込み制御ビットと割込み要因は次のようになっています。

表 40-13 I²C インタフェースの割込み制御ビットと割込み要因

割込みの種類	割込要求フラグビット	フラグレジスタ	割込み要因	割込み要因許可ビット	割込み要求フラグのクリア
ステータス	INT	IBCR	第一バイト送受信後*1 (SSR:DMA=1 のマスタ動作の場合は除く)	IBCR:INTE	割込みフラグビット(IBCR:INT)への"0"書込み
			データ送受信後*1 (SSR:DMA=0 の場合)		
			バスエラー検出		
			アービトレーションロスト検出		
			予約アドレス検出		
			NACK 受信		
			スレーブ受信動作時の受信 FIFO フル		受信 FIFO がエンプティになるまで受信データ読出し後、INT への"0"書込み
	SPC	IBSR	ストップ条件	IBCR:CNDE	SPC への"0"書込み
	RSC		反復スタート検出		RSC への"0"書込み
受信	RDRF	SSR	予約アドレス受信	SMR:RIE	受信データ(RDR)の読出し
			データ受信後		
			FBYTE 設定値分受信		受信 FIFO がエンプティになるまでの受信データ(RDR)の読出し
			FRIIE=1 で受信アイドル検出		
	ORE	SSR	オーバランエラー		受信エラーフラグビット(SSR:REC)への"1"書込み
送信	TDRE	SSR	送信レジスタがエンプティ	SMR:TIE	送信データ(TDR)への書込み, または送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送)*2
			送信バッファエンプティフラグセットビット(SSR:TSET)への"1"書込み		
	FDRQ	FCR1	送信 FIFO がエンプティ	FCR1:FTIE	FIFO 送信データ要求ビットへの"0"書込み、または送信 FIFO がフル
	TBI (SSR:DMA=1)	SSR	送信動作なし	SCR:TBIE	送信データ(TDR)への書込み,または送信 FIFO 動作許可ビットが"0"で送信 FIFO に有効なデータが存在しているときに送信 FIFO 動作許可ビットへの"1"書込み(送信再送)*3
			送信バッファエンプティフラグセットビット(SSR:TSET)への"1"書込み		

*1 : 正常なデータを送受信でき TDRE が"0"の場合、割込みは発生しません。これは DMA 転送をサポートするためです。データ送受信時に INT フラグを発生させたい場合には INT フラグがセットされるタイミングより前に TDRE ビットが"1"である必要があります。

*2 : TDRE ビットが"0"になってから TIE ビットを"1"にしてください。

*3 : SSR:TBI ビットが"0"になってから SSR:TBIE ビットを"1"にしてください。

＜注意事項＞

受信、およびステータス割込みによる DMA 転送はサポートしていません。

40.8.2 I²C インタフェース通信の動作

I²C インタフェース通信の動作について示します。

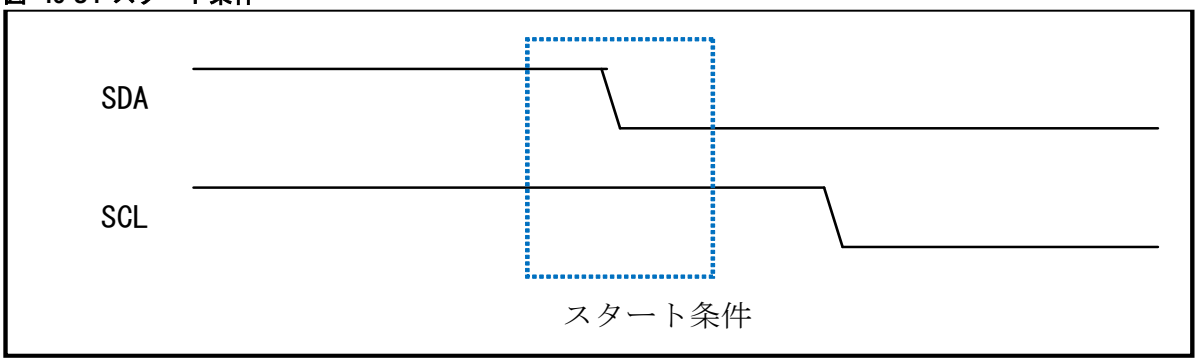
I²Cインタフェースは､2本の双方向バスライン，シリアルデータライン(SDA)およびシリアルクロックライン(SCL)を使用して通信を行います。

40.8.2.1 I²C バススタート条件

I²C バススタート条件について示します。

I²C バスの起動条件を以下に示します。

図 40-54 スタート条件

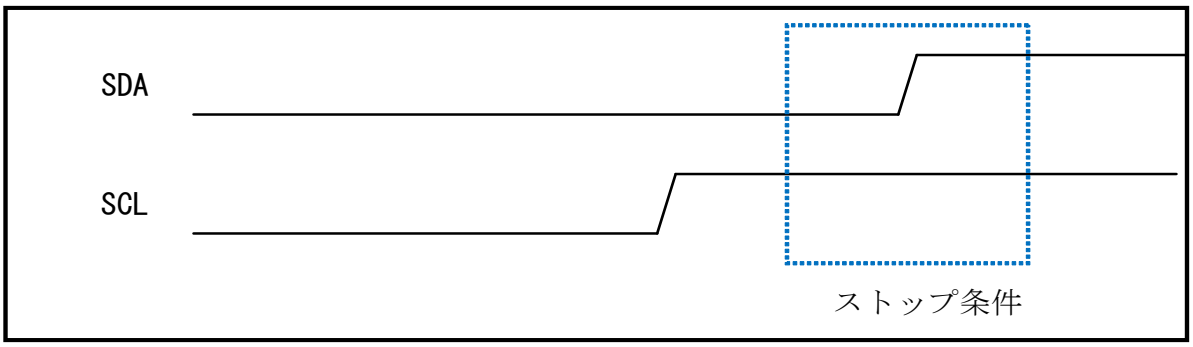


40.8.2.2 I²C バスストップ条件

I²C バスストップ条件について示します。

I²C バスのストップ条件を以下に示します。

図 40-55 ストップ条件

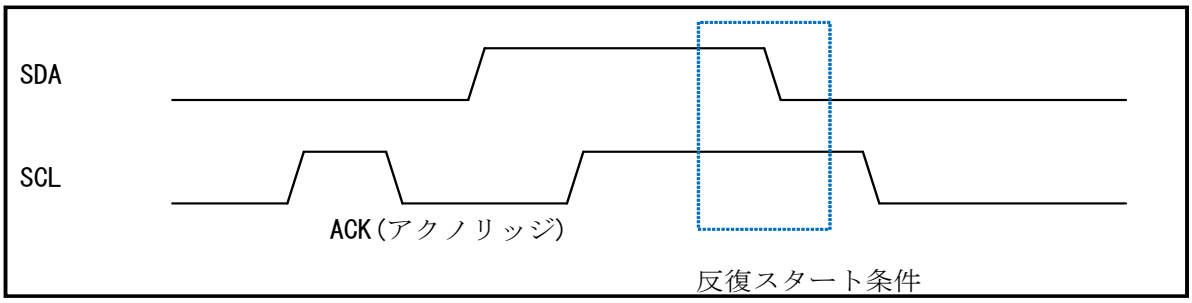


40.8.2.3 I²C バス反復スタート条件

I²C バス反復スタート条件について示します。

I²C バスの反復スタート条件を以下に示します。

図 40-56 反復スタート条件



40.8.2.4 I²C バスエラー

I²C バスエラーについて説明します。

I²Cバス上でデータの送受信中にストップ条件, (反復)スタート条件を検出するとバスエラーとして取り扱います。

- バスエラー発生条件
  バスエラーは以下の条件で IBCR:BER ビットを"1"にします。
  - □ 第一バイト転送中に(反復)スタート条件またはストップ条件を検出
  - □ データの 2～9 (アクノリッジ)ビット目で(反復)スタート条件またはストップ条件を検出
- バスエラー動作
  送受信による割込みフラグ(IBCR:INT)が"1"になったときに IBCR:BER ビットを確認し、IBCR:BER ビットが"1"の場合エラー処理を行ってください。IBCR:BER ビットは IBCR:INT ビットに"0"を書くことによってクリアされます。バスエラーによって IBCR:INT ビットは"1"にセットされますが、I²C バスの SCL を"L"にしてウェイト状態にはしません。

40.8.2.5 ボーレート生成

ボーレート生成について説明します。

専用ボーレートジェネレータは、シリアルクロックの周波数の設定を行います。

■ ボーレート選択

■ 専用ボーレートジェネレータ(リロードカウンタ)で内部クロックを分周して得られるボーレート

2つの内部リロードカウンタがあり、それぞれ送受信シリアルクロックに対応しています。ボーレートジェネレータレジスタ(BGR)で 15 ビットのリロード値を設定することにより、ボーレートを選択できます。リロードカウンタは、設定された値で内部クロックを分周します。

■ ボーレートの計算

2 つの 15 ビットリロードカウンタは、ボーレートジェネレータレジスタ(BGR)で設定します。ボーレートの計算式を以下に示します。

(1)リロード値

$$V = \Phi / b - 1$$

V : リロード値　　b : ボーレート　Φ : 内部クロック(周辺クロック(PCLK))周波数

ただし、I^2C バスの SCL の立上り時間によっては設定したボーレートが発生しませんのでリロード値を調整してください。

(2)計算例

内部クロック(周辺クロック(PCLK))周波数 16MHz, ボーレート 400kbps に設定する場合のリロード値は、次のようになります。

リロード値 :

$$V = (16 \times 1{,}000{,}000) / 400{,}000 - 1 = 39$$

よって、ボーレートは

$$b = (16 \times 1{,}000{,}000) / (39 + 1) = 400\text{kbps}$$

＜注意事項＞

■ ボーレートジェネレータレジスタ(BGR)への書込みは、16 ビットアクセスで行ってください。
■ ISMK レジスタの EN ビットが"0"のときにボーレートジェネレータレジスタの設定を行ってください。
■ 動作モード 4 (I^2C モード)では周辺クロック(PCLK)は 8MHz 以上で使用し、400kbps を超えるボーレートジェネレータの設定は禁止です。
■ リロード値を"0"に設定するとリロードカウンタは停止します。

■ 各ボーレート・各内部クロックに対するリロード値

表 40-14 各ボーレート・各内部クロックに対するリロード値

ボーレート [bps]	内部クロック (周辺クロック (PCLK))					
	8MHz	10MHz	16MHz	20MHz	24MHz	32MHz
400000	19	24	39	49	59	79
200000	39	49	79	99	119	159
100000	79	99	159	199	239	319

本数値は I²C バスの SCL 立上りが 0 s の場合です。I²C バスの SCL 立上りが遅い場合には上記数値より遅いボーレートになります。

■ リロードカウンタの機能

リロード値に対する 15 ビットレジスタから構成されており、内部クロックより送受信クロックを生成します。また、送信リロードカウンタのカウント値をボーレートジェネレータレジスタ(BGR)より読み出すことができます。

■ カウントの開始

ボーレートジェネレータレジスタ(BGR)にリロード値を書き込むと、リロードカウンタはカウントを開始します。

40.8.3 I²C マスタモード

I²C マスタモードについて示します。

マスタモードは、I²C バスにスタート条件を発生させ、I²C バスにクロックを出力します。I²C バスがアイドル状態(SCL="H", SDA="H")のとき、IBCR レジスタの MSS ビットに"1"を設定するとマスタモードになり、IBCR レジスタの ACT ビットが"1"になります。

40.8.3.1 スタート条件生成

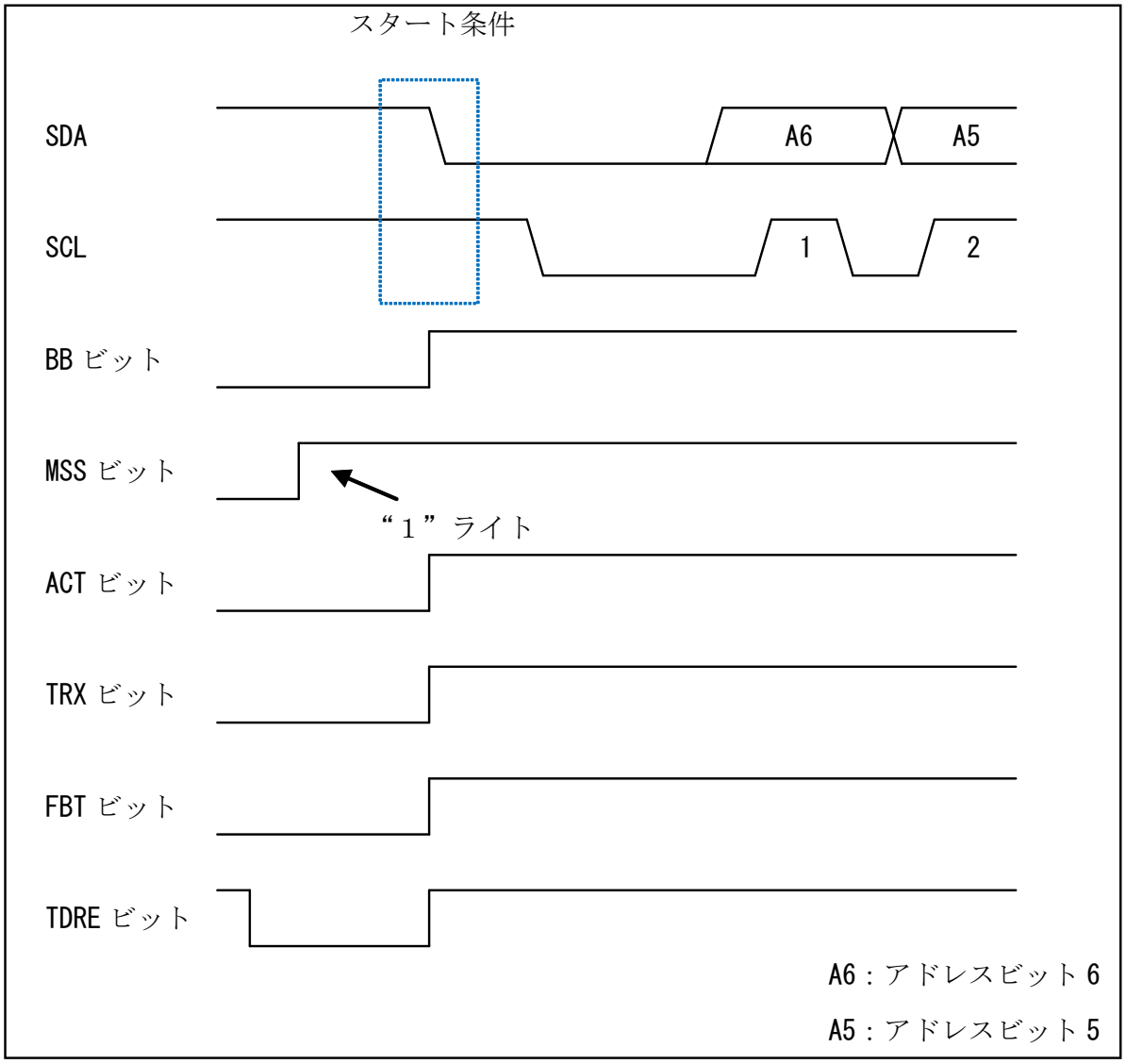
スタート条件生成について示します。

以下の条件でスタート条件が出力されます。

- SDA="H", SCL="H", ISMK:EN=1, IBSR:BB=0 のとき、IBCR:MSS ビットへ"1"書込み

I²C バスへスタート条件を出力すると IBCR:ACT ビットに"1"をセットします。その後、スタート条件を受信すると IBSR:BB ビットが"1"にセットされ、I²C バスは通信中であることを示します。

図 40-57 スタート条件出力および各ビットの関係



＜注意事項＞
動作モード 4 (I²C モード)では周辺クロック(PCLK)は 8MHz 以上で使用し、400kbps を超えるボーレートジェネレータの設定は禁止です。

40.8.3.2 スレーブアドレス出力

スレーブアドレス出力について示します。

スタート条件を出力すると TDR レジスタに設定されたデータを bit7 からアドレスとして出力します。FIFO 許可の場合、最初に書いた TDR レジスタのデータを出力します。bit0 はデータ方向ビット(R/W)として使用され、データ方向ビット(R/W)が"0"のとき、データはライト方向(マスタ→スレーブ)を示します。TDR レジスタへのアドレス設定は、IBCR:MSS=1 または IBCR:SCC=1 を書く前に行ってください。

図 40-58 アドレスおよびデータ方向 (FIFO 禁止の場合)

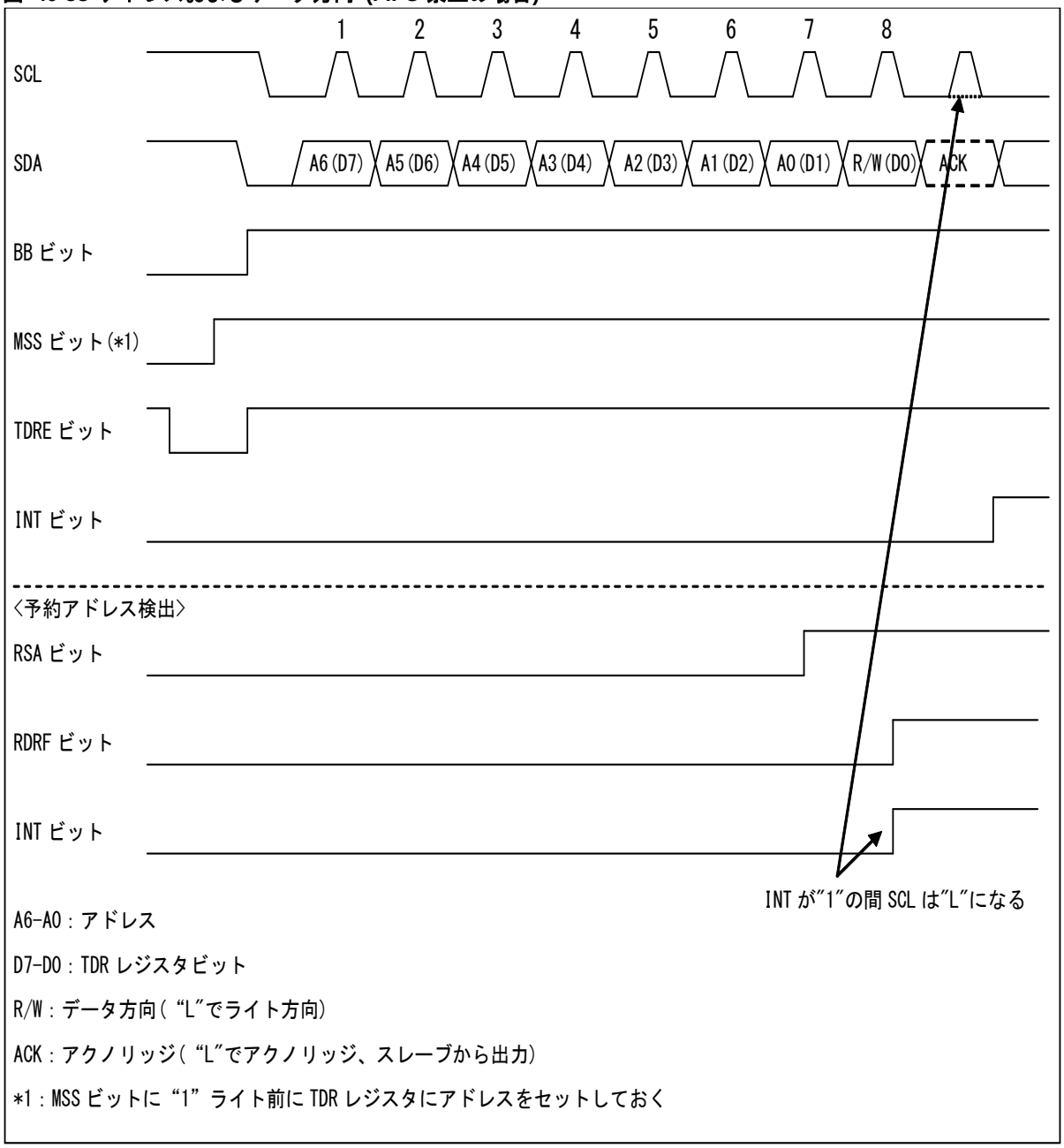
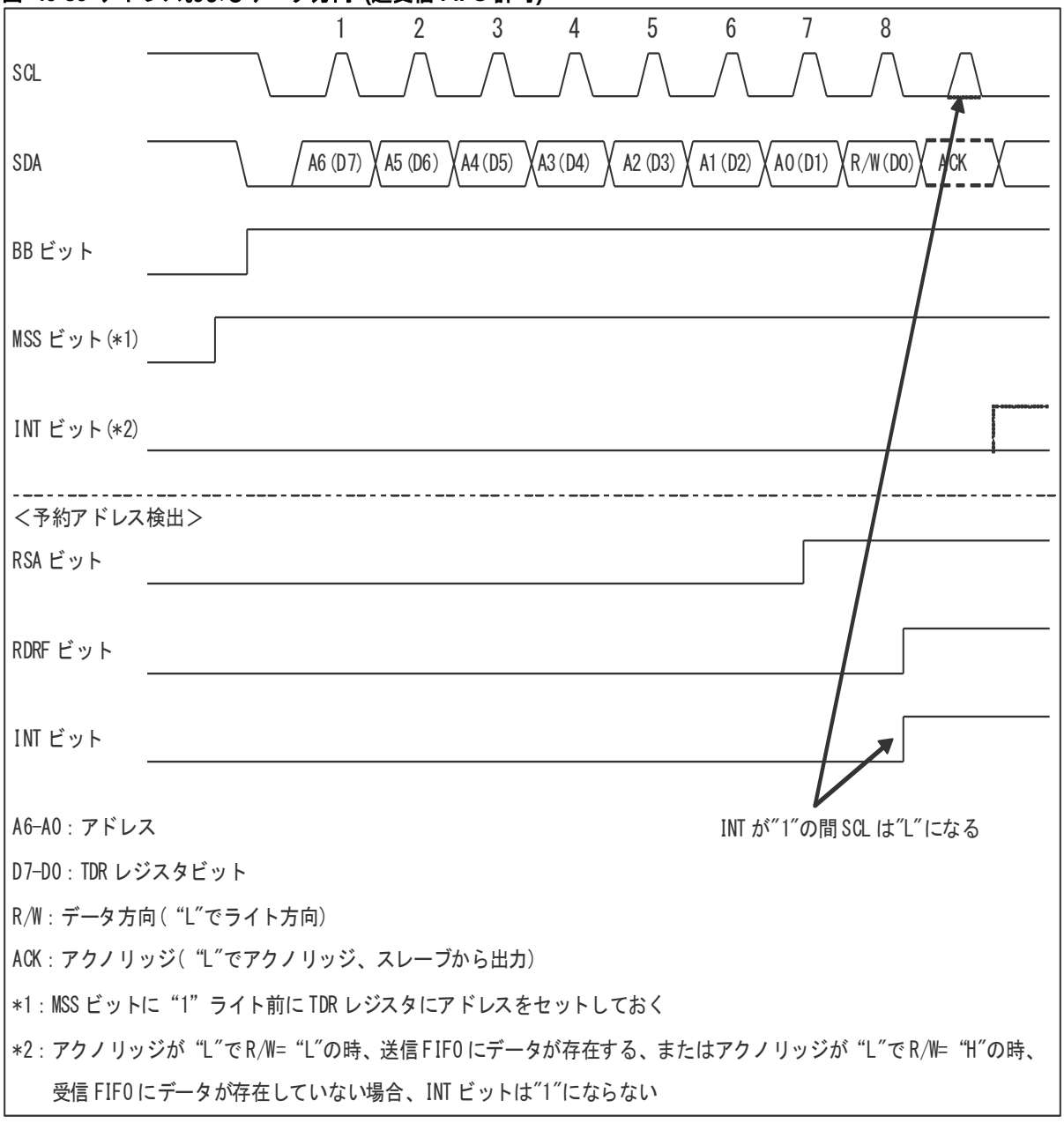


図 40-59 アドレスおよびデータ方向 (送受信 FIFO 許可)



A6-A0 : アドレス

D7-D0 : TDR レジスタビット

R/W : データ方向 (“L”でライト方向)

ACK : アクノリッジ (“L”でアクノリッジ、スレーブから出力)

*1 : MSS ビットに “1” ライト前に TDR レジスタにアドレスをセットしておく

*2 : アクノリッジが “L”で R/W= “L”の時、送信 FIFO にデータが存在する、またはアクノリッジが “L”で R/W= “H”の時、受信 FIFO にデータが存在していない場合、INT ビットは"1"にならない

40.8.3.3 第一バイト送信によるアクノリッジ受信

第一バイト送信によるアクノリッジ受信について説明します。

データ方向ビット(R/W)を出力すると I²C インタフェースは、スレーブからのアクノリッジを受信します。FIFO 許可と FIFO 禁止では、以下の動作になります。

表 40-15 DMA モード禁止時のアクノリッジ受信後の動作 (IBSR:RSA ビット=0, SSR:DMA ビット=0)

送信 FIFO	受信 FIFO	送信 FIFO 状態	受信 FIFO 状態	データ方向ビット (R/W)	アクノリッジ受信直後の動作	
					アクノリッジが ACK	アクノリッジが NACK
禁止	禁止	―	―	0	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
				1		
禁止	許可	―	データなし	0	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
			データあり		IBCR:INT ビットを"1"にしてウェイト	
			―	1	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	
許可	禁止	―	―	0	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
				1		
許可	許可	―	データなし	0	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
			データあり		IBCR:INT ビットを"1"にしてウェイト	
			―	1	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	

表 40-16 DMA モード許可時のアクノリッジ受信後の動作 (IBSR:RSA ビット=0, SSR:DMA ビット=1)

送信 FIFO	受信 FIFO	送信 FIFO 状態	受信 FIFO 状態	データ方向ビット (R/W)	アクノリッジ受信直後の動作	
					アクノリッジが ACK	アクノリッジが NACK
禁止	禁止	—	—	0	SSR:TDRE ビットが"1"であれば SSR:TBI ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、SSR:TBI ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
				1		
禁止	許可	—	データなし	0	SSR:TDRE ビットが"1"であれば SSR:TBI ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、SSR:TBI ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
			データあり		IBCR:INT ビットを"1"にしてウェイト	
			—	1	SSR:TDRE ビットが"1"であれば SSR:TBI ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、SSR:TBI ビットは"0"のままでウェイトなし	
許可	禁止	—	—	0	SSR:TDRE ビットが"1"であれば SSR:TBI ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、SSR:TBI ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
				1		
許可	許可	—	データなし	0	SSR:TDRE ビットが"1"であれば SSR:TBI ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、SSR:TBI ビットは"0"のままでウェイトなし	IBCR:INT ビットを"1"にしてウェイト
			データあり		IBCR:INT ビットを"1"にしてウェイト	
			—	1	SSR:TDRE ビットが"1"であれば SSR:TBI ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、SSR:TBI ビットは"0"のままでウェイトなし	

■ FIFO 禁止(送信 FIFO, 受信 FIFO 両方とも禁止, DMA モード禁止時 (SSR:DMA=0))

- IBSR:RSA ビットが"0"の場合、アクノリッジ受信後、SSR:TDRE ビットが"1"であれば割込みフラグ(IBCR:INT)を"1"にセットし、SCL を"L"に保持してウェイトします。ウェイトは割込みフラグに"0"を書くと割込みフラグが"0"になってウェイトを解除します。SSR:TDRE ビットが"0"であれば ACK を受信すれば割込みフラグを"1"にセットせずに SCL にクロックを発生します。
- IBSR:RSA ビットが"1"の場合、予約アドレス受信後(アクノリッジ前)、割込みフラグ(IBCR:INT)を"1"にセットし、SCL を"L"に保持してウェイトします。RDR レジスタ読出し後、IBCR:ACKE ビット, 送信データを設定し、割込みフラグに"0"を書くと割込みフラグが"0"になってウェイトを解除します。
- 受信したアクノリッジは IBSR:RACK ビットにセットされます。ウェイト中に IBSR:RACK ビットを確認し、NACK であれば、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を書いてストップ条件または反復スタート条件を発生させます。このとき、IBCR:INT ビットは自動的に"0"にクリアされます。

■ FIFO 許可 (DMA モード禁止時(SSR:DMA=0))

- IBCR:MSS ビットに"1"を設定する前に FIFO に以下の設定をする必要があります。
  - スレーブへ送信する場合(データ方向ビット=0)、スレーブアドレスなどを含むデータを送信 FIFO に設定
  - スレーブからデータを受信する場合(データ方向ビット=1)、FIFO バイト数レジスタに受信数を設定し、スレーブアドレスおよびデータ方向ビットと受信したいデータ数分ダミーで送信データレジスタに書込みを行う
- IBSR:RSA ビットが"0"の場合、アクノリッジ受信後、ACK であれば割込みフラグ(IBCR:INT)は"1"にセットせず、データ方向ビットにしたがってデータを送受信します(ウェイトなし)。NACK であれば、割込みフラグ(IBCR:INT)を"1"にセットし、SCL を"L"に保持してウェイトします。
- 受信したアクノリッジは IBSR:RACK ビットに格納されます。ウェイト中に IBSR:RACK ビットを確認し、NACK であれば、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を書いてストップ条件または反復スタート条件を発生させます。このとき、IBCR:INT ビットは自動的に"0"にクリアされます。

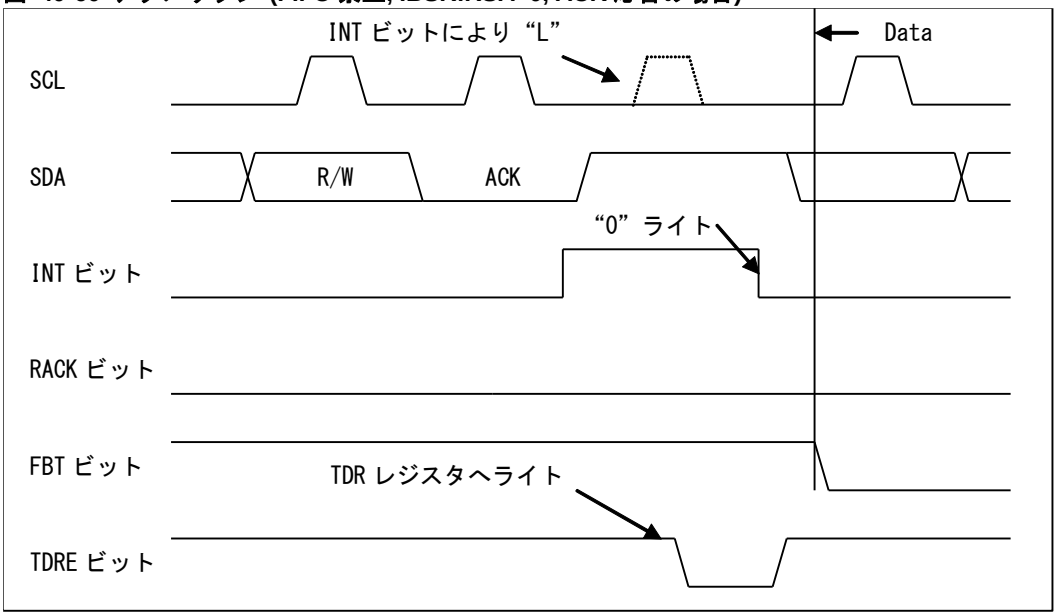
■ FIFO 禁止 (送信 FIFO, 受信 FIFO 両方とも禁止, DMA モード許可時 (SSR:DMA=1))

- IBSR:RSA ビットが"0"の場合、アクノリッジ受信後、SSR:TDRE ビットが"1"であれば送信バスアイドルフラグ(SSR:TBI)を"1"にセットし、SCL を"L"に保持してウェイトします。ウェイトは TDR レジスタに送信するデータを書くと送信バスアイドルフラグが"0"になってウェイトを解除します。SSR:TDRE ビットが"0"であれば ACK を受信すれば送信バスアイドルフラグ(SSR:TBI)を"1"にセットせずに SCL にクロックを発生します。
- IBSR:RSA ビットが"1"の場合、予約アドレス受信後(アクノリッジ前)、割込みフラグ(IBCR:INT)を"1"にセットし、SCL を"L"に保持してウェイトします。RDR レジスタ読出し後、IBCR:ACKE ビット、送信データを設定し、割込みフラグに"0"を書くと割込みフラグが"0"になってウェイトを解除します。
- 受信したアクノリッジは IBSR:RACK ビットにセットされます。ウェイト中に IBSR:RACK ビットを確認し、NACK であれば、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を書いてストップ条件または反復スタート条件を発生させます。このとき、IBCR:INT ビットは自動的に"0"にクリアされます。

■ FIFO 許可 (DMA モード許可時 (SSR:DMA=1))

- IBCR:MSS ビットに"1"を設定する前に FIFO に以下の設定をする必要があります。
  - スレーブへ送信する場合(データ方向ビット=0)、スレーブアドレスなどを含むデータを送信 FIFO に設定
  - スレーブからデータを受信する場合(データ方向ビット=1)、FIFO バイト数レジスタに受信数を設定し、スレーブアドレスおよびデータ方向ビットと受信したいデータ数分ダミーで送信データレジスタに書込みを行う
- IBSR:RSA ビットが"0"の場合、アクノリッジ受信後、ACK であれば割込みフラグ(IBCR:INT)は"1"にセットせず、データ方向ビットにしたがってデータを送受信します(ウェイトなし)。NACK であれば、割込みフラグ(IBCR:INT)を"1"にセットし、SCL を"L"に保持してウェイトします。
- 受信したアクノリッジは IBSR:RACK ビットに格納されます。ウェイト中に IBSR:RACK ビットを確認し、NACK であれば、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を書いてストップ条件または反復スタート条件を発生させます。このとき、IBCR:INT ビットは自動的に"0"にクリアされます。

図 40-60 アクノリッジ (FIFO 禁止, IBSR:RSA=0, ACK 応答の場合)



アドレスに対するウェイトは

- IBSR:RSA ビットが"0"の場合、アクノリッジ受信後
- IBSR:RSA ビットが"1"の場合、アクノリッジ受信前

になります。IBCR:WSEL の設定には依存しません

図 40-61 アクノリッジ (FIFO 禁止, IBSR:RSA=0, NACK 応答の場合)

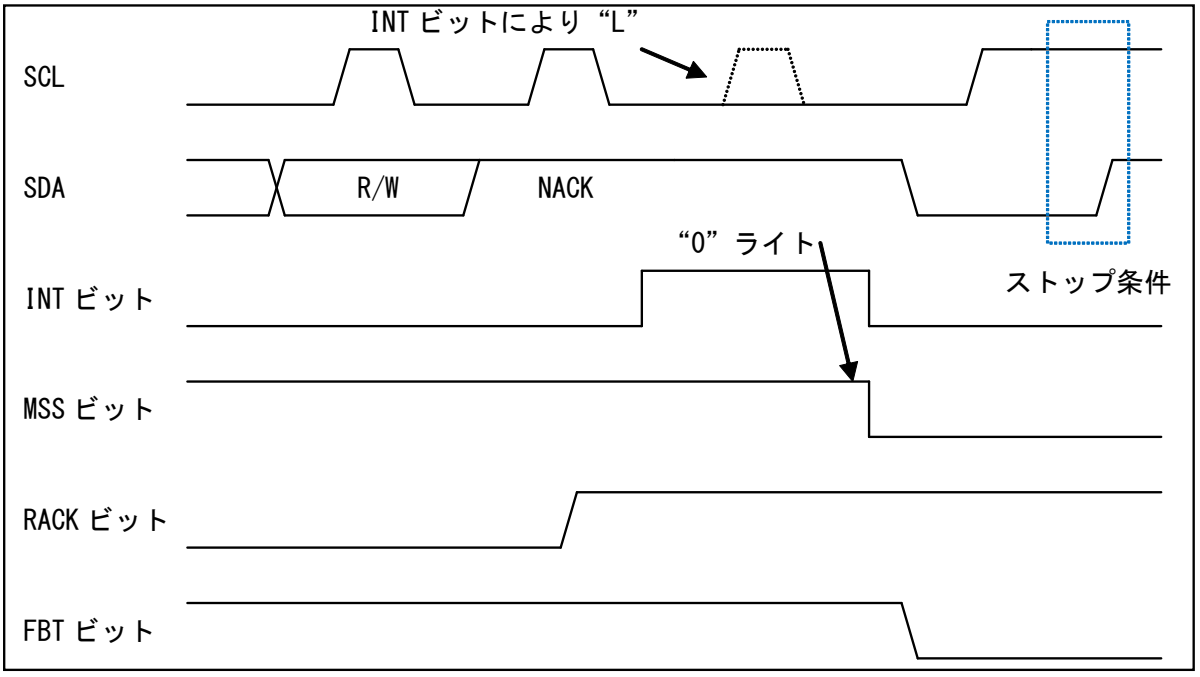


図 40-62 アクノリッジ (FIFO 禁止, IBSR:RSA=1, ACK 応答の場合)

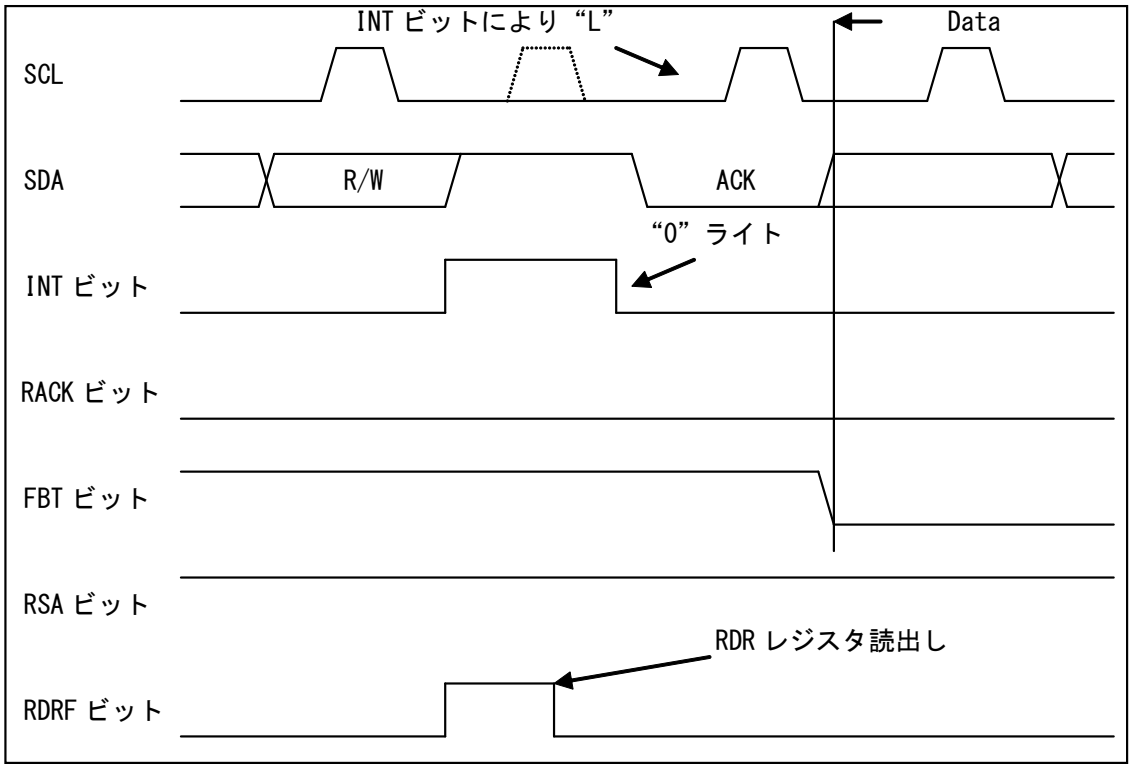


図 40-63 アクノリッジ (FIFO 禁止, IBSR:RSA=1, NACK 応答の場合)

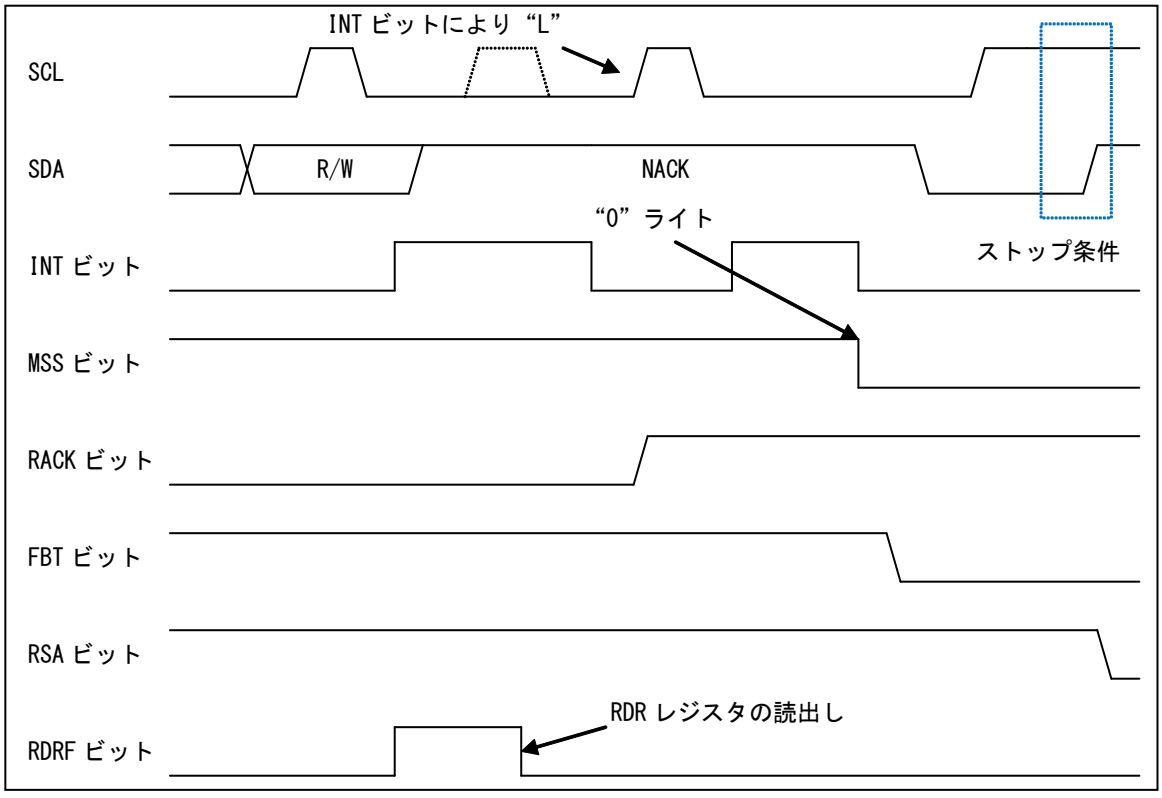
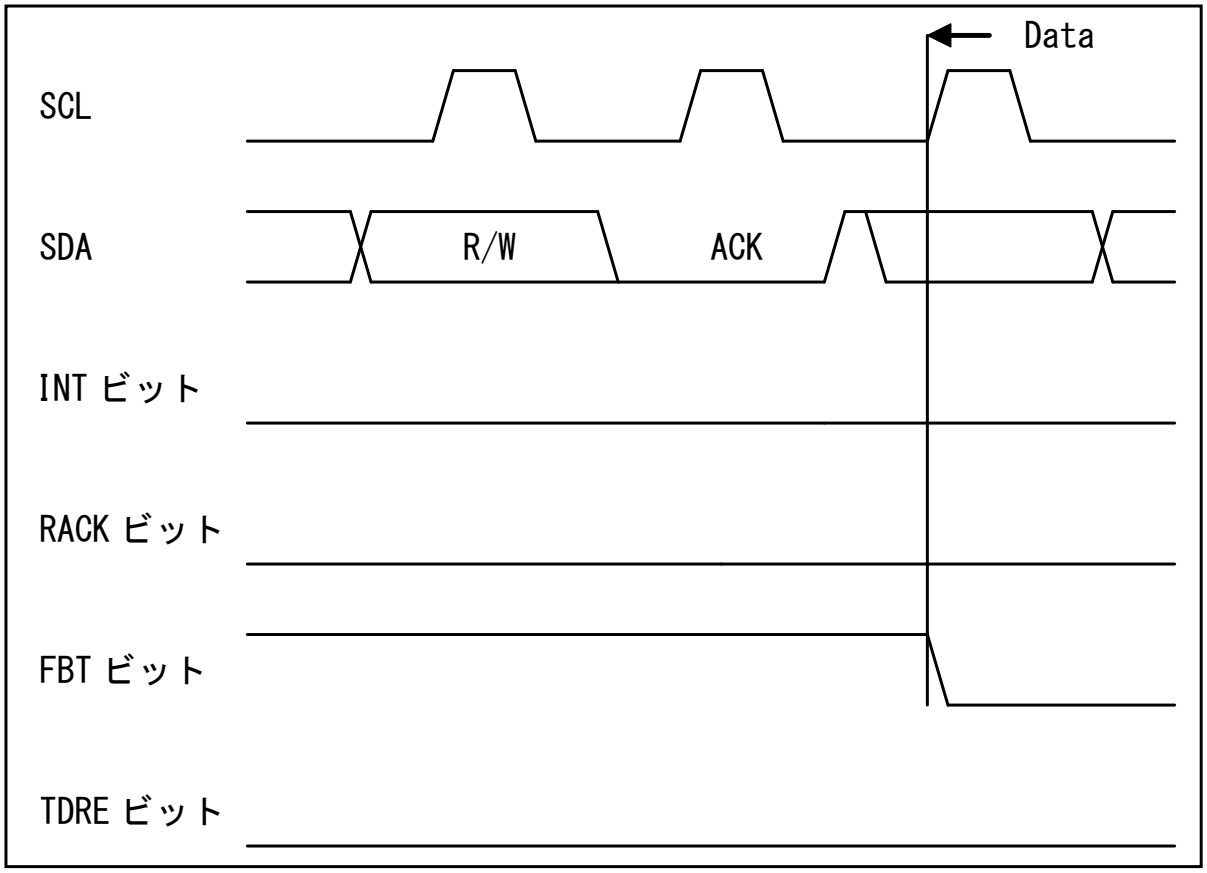


図 40-64 アクノリッジ (FIFO 許可，送信 FIFO データあり，受信 FIFO データなし, IBSR:RSA=0, ACK 応答の場合)



40.8.3.4 マスタによるデータ送信

マスタによるデータ送信について説明します。

データ方向ビット(R/W)が"0"の場合、データはマスタから送信します。1 バイト送信ごとにスレーブから ACK または NACK の応答があります。IBCR:WSEL ビットの設定によってウェイトの発生する場所が以下のようになります。

表 40-17 DMA モード禁止時 (SSR:DMA=0) のマスタデータ送信時の IBCR:WSEL ビット

WSEL	動作
0	第二バイト以降、SSR:TDRE ビットが"1"またはアービトレーションロスト検出でアクノリッジ後、割込みフラグ(IBCR:INT)を"1"、SCL を"L"にしてウェイト状態にします。FIFO 許可の場合アクノリッジ後、アービトレーションロスト検出または送信データレジスタに有効なデータがなくなった(SSR:TDRE=1)ときにアクノリッジ後割込みフラグ(IBCR:INT)を"1"にしてウェイト状態にします。
1	第二バイト以降、SSR:TDRE ビットが"1"またはアービトレーションロスト検出でマスタが 1 バイトのデータを送信後、割込みフラグ(IBCR:INT) "1"、SCL を"L"にしてウェイト状態にします。FIFO 許可の場合、アービトレーションロスト検出または送信データレジスタに有効なデータがなくなった(SSR:TDRE=1)ときにデータ送信後、割込みフラグ(IBCR:INT)を"1"にしてウェイト状態にします。

表 40-18 DMA モード許可時 (SSR:DMA=1) のマスタデータ送信時の IBCR:WSEL ビット

WSEL	動作
0	第二バイト以降、SSR:TDRE ビットが"1"でアクノリッジ後、送信バスアイドルフラグ(SSR:TBI)を"1"、SCL を"L"にしてウェイト状態にします。また、FIFO 許可の場合アクノリッジ後、送信データレジスタに有効なデータがなくなった(SSR:TDRE=1)ときにアクノリッジ後、送信バスアイドルフラグ(SSR:TBI)を"1"にしてウェイト状態にします。
1	第二バイト以降、SSR:TDRE ビットが"1"でマスタが 1 バイトのデータを送信後、送信バスアイドルフラグ(SSR:TBI)を"1"、SCL を"L"にしてウェイト状態にします。また、FIFO 許可の場合アクノリッジ後、送信データレジスタに有効なデータがなくなった(SSR:TDRE=1)ときにマスタが 1 バイトのデータを送信後、送信バスアイドルフラグ (SSR:TBI)を"1"にしてウェイト状態にします。

ただし、以下の場合、IBCR:WSEL の設定に依存せずにアクノリッジ後に割込みフラグ(IBCR:INT)をセットします。

- ストップ条件設定(IBCR:MSS=0, ACT=1)時以外に NACK を受信した場合

スレーブへデータを送信する場合の手順の一例を以下に示します。

- DMA モード禁止時(SSR:DMA=0)のスレーブへのデータ送信

(1) 予約アドレス以外への送信の場合

送信 FIFO が禁止されている場合

① Slave Address (データ方向ビットも含む)を TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後 ACK を受信し、割込みフラグ(IBCR:INT)が"1"になります。

③ TDR レジスタに送信するデータを書きます。

④ IBCR:WSEL ビット更新と共に割込みフラグ(IBCR:INT)に"0"を書込み、I²C バスのウェイトを解除します。

⑤ 1 バイト送信後 WSEL=0 の場合アクノリッジ受信後、IBCR:WSEL=1 の場合 1 バイト送信直後割込みフラグを"1"にして I²C バスをウェイトします。所定のデータ数を送信するまで③～⑤を繰り返します。ただし、IBCR:WSEL=1 のときウェイト解除後、NACK を受信した場合にはアクノリッジ受信後にもう 1 度割込みが発生し、バスをウェイトします。

⑥ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を発生させます。

送信 FIFO が許可されている場合

① Slave Address (データ方向ビットも含む)、送信データを TDR レジスタに書きます。

② IBCR:WSEL ビット設定と共に IBCR:MSS ビットに"1"を書きます。

③ 送信中に NACK 受信した場合、その直後に割込みフラグ(IBCR:INT)を"1"にし I²C バスをウェイトします。すべて ACK 応答を受信した場合、最終バイト送信後 IBCR:WSEL の設定にしたがって割込みフラグを"1"にして I²C バスをウェイトします。

④ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を生成させます。

(2) 予約アドレスへの送信の場合

送信 FIFO が禁止されている場合

① Slave Address として予約アドレスを TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後、割込みフラグ(IBCR:INT)が"1"になります。

③ RDR レジスタを読み出し、予約アドレスを確認します。(*1)

④ TDR レジスタに送信するデータを書きます。

⑤ IBCR:WSEL ビット更新と共に割込みフラグ(IBCR:INT)に"0"を書込み、I²C バスのウェイトを解除します。

⑥ 1 バイト送信後 IBCR:WSEL=0 の場合アクノリッジ受信後、IBCR:WSEL=1 の場合 1 バイト送信直後割込みフラグを"1"にして I²C バスをウェイトします。所定のデータ数を送信するまで④～⑥を繰り返します。ただし、IBCR:WSEL=1 のときウェイト解除後、NACK を受信した場合にはアクノリッジ受信後にもう 1 度割込みが発生し、バスをウェイトします。

⑦ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を発生させます。

送信 FIFO が許可されている場合

① Slave Address として予約アドレスを TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後、割込みフラグ(IBCR:INT)が"1"になります。

③ RDR レジスタを読み出し、予約アドレスを確認します。(*1)

④ TDR レジスタに全送信データ(送信 FIFO がフルとなる場合にはその状態になるまで)を書きます。

⑤ 送信中に NACK 受信した場合、その直後に割込みフラグ(IBCR:INT)を"1"にし I²C バスをウェイトします。すべて ACK 応答を受信した場合、最終バイト送信後 IBCR:WSEL の設定にしたがって割込みフラグ(IBCR:INT)を"1"にして I²C バスをウェイトします。

⑥ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を発生させます。

*1 : マルチマスタで予約アドレスがゼネラルコールの場合、アービトレーションロストが発生してスレーブとして動作する可能性がある場合、IBCR:ACKE ビットを"1"、IBCR:WSEL ビットを"1"にして次のデータでマスタとして動作するのか、スレーブとして動作するのか確認する必要があります。

■ DMA モード許可時(SSR:DMA=1)のスレーブへのデータ送信

(1) 予約アドレス以外への送信の場合

送信 FIFO が禁止されている場合

① Slave Address (データ方向ビットも含む)を TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後 ACK を受信し、送信バスアイドルフラグ(SSR:TBI)が"1"になります。

③ TDR レジスタに送信するデータを書きこみ、I²C バスのウェイトを解除します。

④ 1 バイト送信後 WSEL=0 の場合アクノリッジ受信後、IBCR:WSEL=1 の場合 1 バイト送信直後送信バスアイドルフラグ(SSR:TBI)を"1"にして I²C バスをウェイトします。

⑤ TDR レジスタに送信するデータを書き込み、I²C バスのウェイトを解除します。

⑥ 1 バイト送信後 WSEL=0 の場合アクノリッジ受信後、IBCR:WSEL=1 の場合 1 バイト送信直後送信バスアイドルフラグ(SSR:TBI)を"1"にして I²C バスをウェイトします。所定のデータ数を送信するまで⑤～⑥を繰り返します。ただし、IBCR:WSEL=1 のときウェイト解除後、NACK を受信した場合にはアクノリッジ受信後に割込みフラグ(IBCR:INT)が"1"になり、バスをウェイトします。

⑦ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定(*2)し、ストップ条件または反復スタート条件を発生させます。

送信 FIFO が許可されている場合

① Slave Address (データ方向ビットも含む)、送信データを TDR レジスタに書きます。

② IBCR:WSEL ビット設定と共に IBCR:MSS ビットに"1"を書きます。

③ 送信中に NACK 受信した場合、その直後に割込みフラグ(IBCR:INT)を"1"にし I²C バスをウェイトします。すべて ACK 応答を受信した場合、最終バイト送信後 IBCR:WSEL の設定にしたがって送信バスアイドルフラグ(SSR:TBI)を"1"にして I²C バスをウェイトします。

④ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定(*2)し、ストップ条件または反復スタート条件を生成させます。

(2) 予約アドレスへの送信の場合

送信 FIFO が禁止されている場合

① Slave Address として予約アドレスを TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後、割込みフラグ(IBCR:INT)が"1"になります。

③ RDR レジスタを読み出し、予約アドレスを確認します。(*1)

④ TDR レジスタに送信するデータを書きます。

⑤ IBCR:WSEL ビット更新と共に割込みフラグ(IBCR:INT)に"0"を書込み、I²C バスのウェイトを解除します。

⑥ 1 バイト送信後 IBCR:WSEL=0 の場合アクノリッジ受信後、IBCR:WSEL=1 の場合 1 バイト送信直後割込みフラグを"1"にして I²C バスをウェイトします。

⑦ TDR レジスタに送信するデータを書きこみ、I²C バスのウェイトを解除します。

⑧ 1 バイト送信後 IBCR:WSEL=0 の場合アクノリッジ受信後、IBCR:WSEL=1 の場合 1 バイト送信直後送信バスアイドルフラグ(SSR:TBI)を"1"にして I²C バスをウェイトします。所定のデータ数を送信するまで⑦～⑧を繰り返します。ただし、IBCR:WSEL=1 のときウェイト解除後、NACK を受信した場合にはアクノリッジ受信後にもう 1 度割込みが発生し、バスをウェイトします。

⑨ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定(*2)し、ストップ条件または反復スタート条件を発生させます。

送信 FIFO が許可されている場合

① Slave Address として予約アドレスを TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後、割込みフラグ(IBCR:INT)が"1"になります。

③ RDR レジスタを読み出し、予約アドレスを確認します。(*1)

④ TDR レジスタに全送信データ(送信 FIFO がフルとなる場合にはその状態になるまで)を書きます。

⑤ 送信中に NACK 受信した場合、その直後に割込みフラグ(IBCR:INT)を"1"にし I²C バスをウェイトします。すべて ACK 応答を受信した場合、最終バイト送信後 IBCR:WSEL の設定にしたがって割込みフラグ(IBCR:INT)を"1"にして I²C バスをウェイトします。

⑥ IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定(*2)し、ストップ条件または反復スタート条件を発生させます。

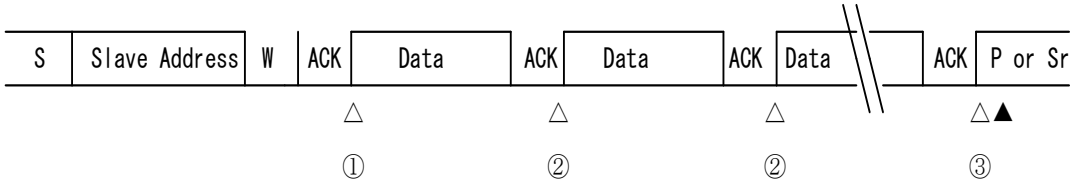
*1 : マルチマスタで予約アドレスがゼネラルコールの場合、アービトレーションロストが発生してスレーブとして動作する可能性がある場合、IBCR:ACKE ビットを"1"、IBCR:WSEL ビットを"1"にして次のデータでマスタとして動作するのか、スレーブとして動作するのか確認する必要があります。

*2: DMA モードが許可(SSR:DMA=1)で SSR:TBI ビットが"1"で IBCR:INT ビットが"0"のときに反復スタート条件を発行する場合は、IBCR:INT ビットに"1"を書き込んだ後、IBCR:INT ビットが"1"にセットされていることを確認してから TDR にスレーブアドレスを書き込み、IBCR:SCC ビットに"1"を設定してください。

＜注意事項＞

- 7ビットスレーブアドレスの検出を許可しているとき(ISBA:SAEN=1)にマスタモード時に7ビットスレーブアドレスを指定することは禁止です。
- 送受信中に IBCR レジスタを変更する場合、割込みフラグ(IBCR:INT)が"1"のときに変更してください。
- IBCR:WSEL ビットを変更した場合、次のデータの割込みフラグ(IBCR:INT)および DMA モードが許可時(SSR:DMA=1)での送信バスアイドルフラグ(SSR:TBI)の発生条件に使用されます。
- DMA モードが禁止時(SSR:DMA=0)、データ送信中で SSR:TDRE が"1"のときに TDR レジスタへ送信データを書き込み、ACK 応答を検出すると割込みフラグ(IBCR:INT)は"1"にならずにその書き込まれたデータが送信されます。
- DMA モード禁止時(SSR:DMA=0)、データ受信中に SSR:TDRE が"1"のときに TDR レジスタへ送信データを書き込み、ACK 応答した場合、割込みフラグ(IBCR:INT)は"1"にならずに SSR:RDRF のみ"1"になります(受信 FIFO 許可の場合、FBYTE レジスタ設定分受信した場合)。
- DMA 転送許可時(SSR:DMA=1)、データ送信中で SSR:TDRE が"1"のときに TDR レジスタへ送信データを書き込み、ACK 応答を検出すると送信バスアイドルフラグ(SSR:TBI)は"1"にならずにその書き込まれたデータが送信されます。
- DMA 転送許可時(SSR:DMA=1)、データ受信中に SSR:TDRE が"1"のときに TDR レジスタへ送信データを書き込み、ACK 応答した場合、送信バスアイドルフラグ(SSR:TBI)は"1"にならずに SSR:RDRF のみ"1"になります(受信 FIFO 許可の場合、FBYTE レジスタ設定分受信した場合)。

図 40-65 FIFO 禁止によるマスタ送信の割込み 1 (SSR:DMA=0, IBCR:WSEL=0, IBSR:RSA=0)



S：スタート条件， W：データ方向ビット(ライト方向)， P：ストップ条件， Sr：反復スタート条件，

△：INTE=“1”による割込み，▲：CNDE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

　－ TDR レジスタに送信データ書込み後、INT=“0”書込み

② 1バイト送信＋アクノリッジ受信により、割込み発生

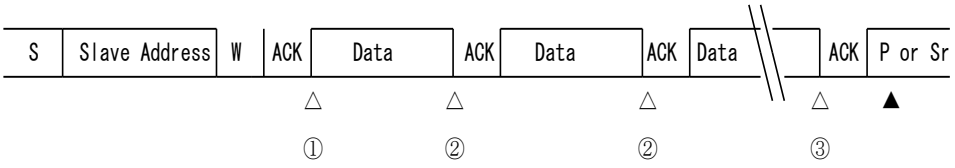
　－ TDR レジスタに送信データ書込み後、INT=“0”書込み

③ 1バイト送信＋アクノリッジ受信により、割込み発生

　－ MSS=“0”または MSS=“1”， SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDRE ビットは“1”

図 40-66 FIFO 禁止によるマスタ送信の割込み 2 (SSR:DMA=0, IBCR:WSEL=1, IBSR:RSA=0, ACK 応答)



S : スタート条件

W : データ方向ビット(ライト方向)

P : ストップ条件

Sr : 反復スタート条件

△ : INTE=“1”による割込み

▲ : CNDE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み後、INT=“0”書込み

② 1バイト送信により、割込み発生

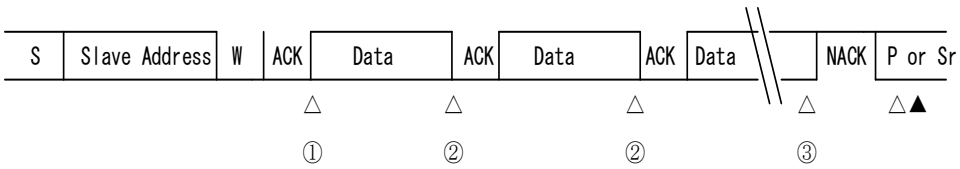
－ TDR レジスタに送信データを書込み後、INT=“0”書込み

③ 1バイト送信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDRE ビットは“1”

図 40-67 FIFO 禁止によるマスタ送信の割込み 3 (SSR:DMA=0, IBCR:WSEL=1, IBSR:RSA=0, NACK 応答)



S : スタート条件

W : データ方向ビット(ライト方向)

P : ストップ条件

Sr : 反復スタート条件

△ : INTE=“1”による割込み

▲ : CNDE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み後、INT=“0”書込み

② 1バイト送信により、割込み発生

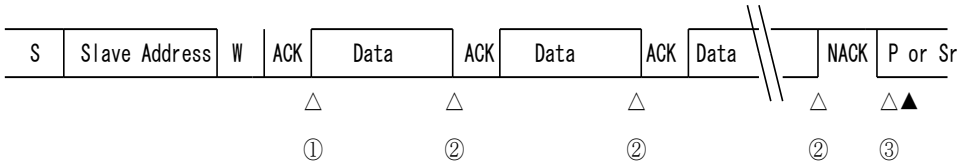
－ TDR レジスタに送信データを書込み後、INT=“0”書込み

③ 1バイト送信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDRE ビットは“1”

図 40-68 FIFO 禁止によるマスタ送信の割込み 4 (SSR:DMA=0, IBCR:WSEL=1", IBSR:RSA=0, 途中 NACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み後、INT=“0”書込み

② 1バイト送信により、割込み発生

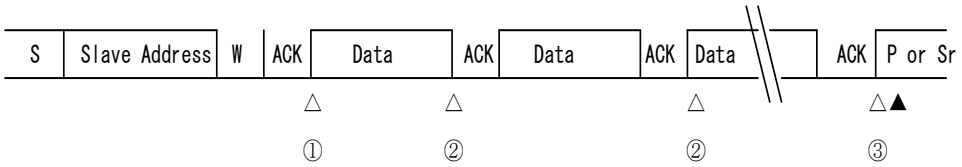
－ TDR レジスタに送信データを書込み後、INT=“0”書込み

③ NACK 応答により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDRE ビットは“1”

図 40-69 FIFO 禁止によるマスタ送信の割込み 5 (SSR:DMA=0, IBCR:WSEL=1->0, IBSR:RSA=0, ACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み後、INT=“0”書込み

② 1バイト送信により、割込み発生

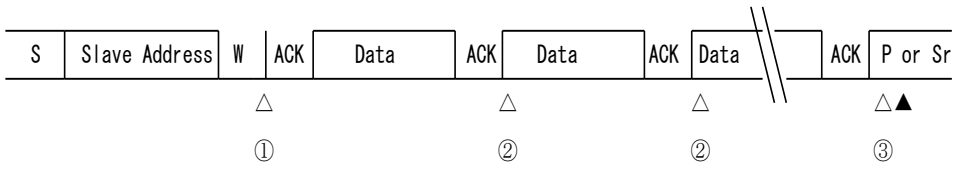
－ TDR レジスタに送信データを書込み後、WSEL="0", INT=“0”書込み

③ 1バイト送信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDRE ビットは“1”

図 40-70 FIFO 禁止によるマスタ送信の割込み 6 (SSR:DMA=0, IBCR:WSEL=0, IBSR:RSA=1)



S：スタート条件

W：データ方向ビット(ライト方向)

P：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① スレーブアドレス(予約アドレス)送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み後、INT=“0”書込み

② 1バイト送信＋アクノリッジ受信により、割込み発生

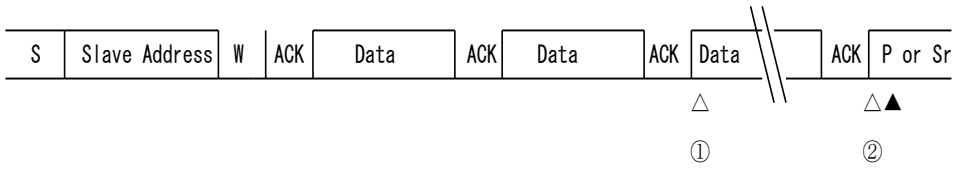
－ TDR レジスタに送信データを書込み後、INT=“0”書込み

③ 1バイト送信＋アクノリッジ受信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDRE ビットは“1”

図 40-71 FIFO 許可によるマスタ送信の割込み 7 (SSR:DMA=0, IBCR:WSEL=0, IBSR:RSA=0, ACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

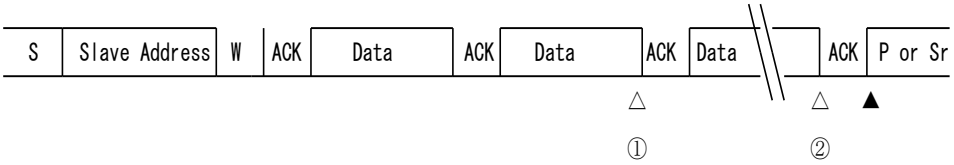
① 送信 FIFO がエンプティにより割込み発生

－ 送信 FIFO に送信データを書込み後、INT=“0”書込み

② 最終バイト送信(送信 FIFO がエンプティ)＋アクノリッジ受信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

図 40-72 FIFO 許可によるマスタ送信の割込み 8 (SSR:DMA=0, IBCR:WSEL=1, IBSR:RSA="0)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

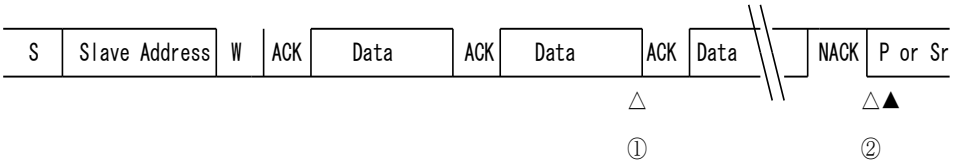
① 送信 FIFO がエンプティにより、割込み発生

－ 送信 FIFO に送信データを書込み後、INT=“0”書込み

② 最終バイト送信(送信 FIFO がエンプティ)により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

図 40-73 FIFO 許可によるマスタ送信の割込み 9 (SSR:DMA=0, IBCR:WSEL=1, IBSR:RSA=0, NACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

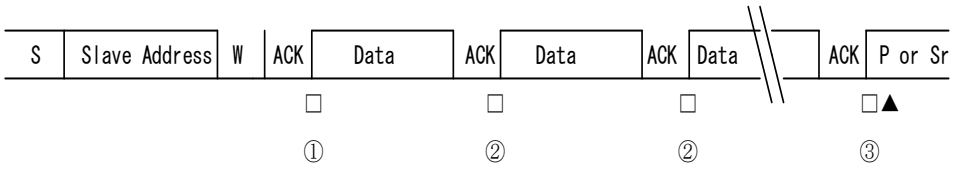
① 送信 FIFO がエンプティにより、割込み発生

－ 送信 FIFO に送信データを書込み後、INT=“0”書込み

② NACK 応答により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

図 40-74 FIFO 禁止によるマスタの割込み 10 (SSR:DMA=1, IBCR:WSEL=0, IBSR:RSA=0)



S : スタート条件

W : データ方向ビット(ライト方向)

P : ストップ条件

Sr : 反復スタート条件

▲ : CNDE=“1”による割込み

□ : TBIE=”1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データ書込み

② 1バイト送信＋アクノリッジ受信により、割込み発生

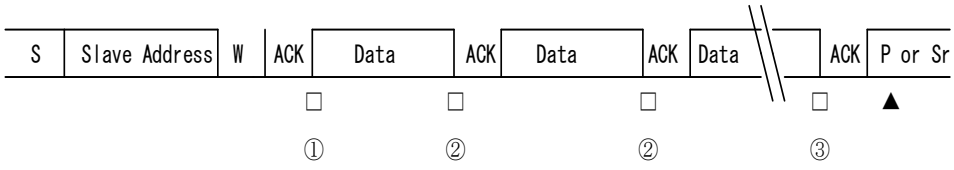
－ TDR レジスタに送信データ書込み

③ 1バイト送信＋アクノリッジ受信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(TBI)発生時、TDRE ビットは“1”

図 40-75 FIFO 禁止によるマスタ送信の割込み 11 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0, ACK 応答)



S : スタート条件

W : データ方向ビット(ライト方向)

P : ストップ条件

Sr : 反復スタート条件

▲ : CNDE=“1”による割込み

□ : TBIE=”1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み

② 1バイト送信により、割込み発生

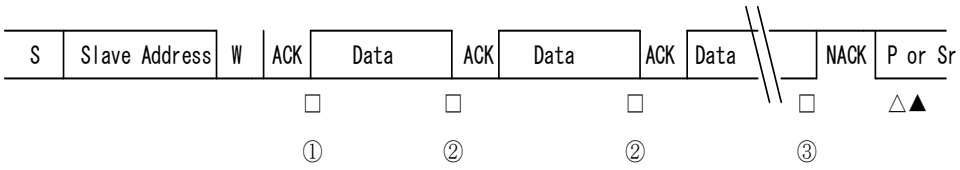
－ TDR レジスタに送信データを書込み

③ 1バイト送信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDRE ビットは“1”

図 40-76 FIFO 禁止によるマスタ送信の割込み 12 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0, NACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

□：TBIE=”1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み

② 1バイト送信により、割込み発生

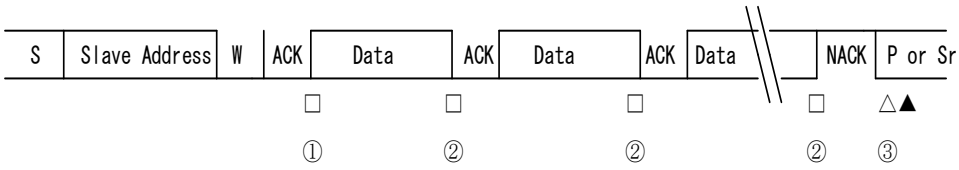
－ TDR レジスタに送信データを書込み

③ 1バイト送信により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT、TBI)発生時、TDRE ビットは“1”

図 40-77 FIFO 禁止によるマスタ送信の割込み 13 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0, 途中 NACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

□：TBIE=”1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み

② 1バイト送信により、割込み発生

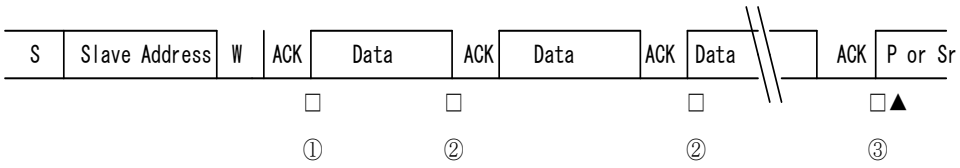
－ TDR レジスタに送信データを書込み

③ NACK 応答により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT、TBI)発生時、TDRE ビットは“1”

図 40-78 FIFO 禁止によるマスタ送信の割込み 14 (SSR:DMA=1, IBCR:WSEL=1->0, IBSR:RSA=0, ACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

▲：CNDE=“1”による割込み

□：TBIE=”1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み

② 1バイト送信により、割込み発生

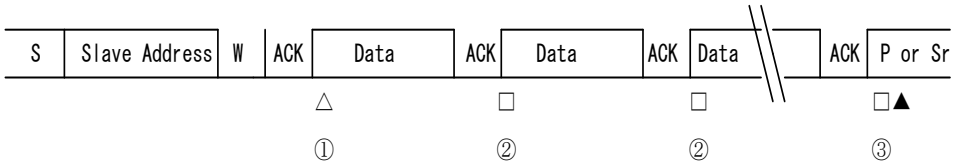
－ WSEL=”0”書込み後、TDR レジスタに送信データを書込み

③ 1バイト送信により、割込み発生

－ MSS=“0”又はMSS=“1”, SCC=“1”を設定

*) 割込みフラグ(TBIE)発生時、TDRE ビットは“1”

図 40-79 FIFO 禁止によるマスタの割込み 15 (SSR:DMA=1, IBCR:WSEL=0, IBSR:RSA=1)



S：スタート条件

W：データ方向ビット(ライト方向)

P：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

□：TBIE=”1”による割込み

① スレーブアドレス(予約アドレス)送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタに送信データを書込み後、INT=“0”書込み

② 1バイト送信＋アクノリッジ受信により、割込み発生

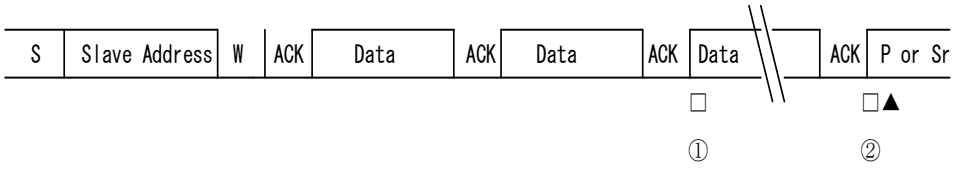
－ TDR レジスタに送信データを書込み

③ 1バイト送信＋アクノリッジ受信により、割込み発生

－ MSS=“0”又はMSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT、TBI)発生時、TDRE ビットは“1”

図 40-80 FIFO 許可によるマスタ送信の割込み 16 (SSR:DMA=1, IBCR:WSEL=0, IBSR:RSA=0, ACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

□：TBIE=”1”による割込み

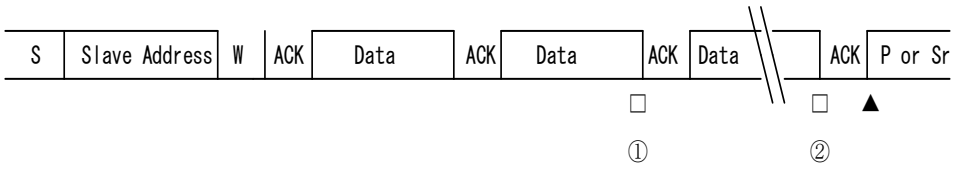
① 送信FIFOがエンプティにより割込み発生

－ 送信FIFOに送信データを書込み

② 最終バイト送信(送信FIFOがエンプティ)＋アクノリッジ受信により、割込み発生

－ MSS=“0”又はMSS=“1”，SCC=“1”を設定

図 40-81 FIFO 許可によるマスタ送信の割込み 17 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

□：TBIE=”1”による割込み

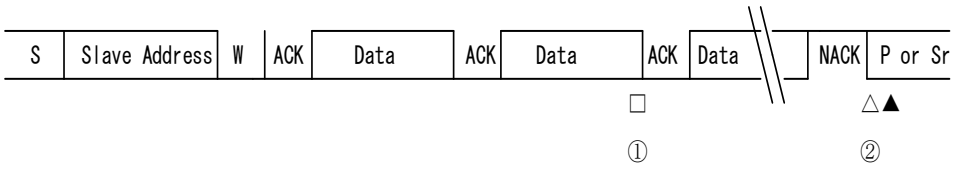
① 送信FIFOがエンプティにより、割込み発生

－ 送信FIFOに送信データを書込み

② 最終バイト送信(送信FIFOがエンプティ)により、割込み発生

－ MSS=“0”又はMSS=“1”，SCC=“1”を設定

図 40-82 FIFO 許可によるマスタ送信の割込み 18 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0, NACK 応答)



S ：スタート条件

W ：データ方向ビット(ライト方向)

P ：ストップ条件

Sr：反復スタート条件

△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

□：TDRE=”1”による割込み

① 送信 FIFO がエンプティにより、割込み発生

－ 送信 FIFO に送信データを書込み

② NACK 応答により、割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

40.8.3.5 マスタによるデータ受信

マスタによるデータ受信について説明します。

■ DMA モードが禁止の場合(SSR:DMA=0)

データ方向ビット(R/W)が"1"の場合、スレーブから送信されたデータを受信します。

FIFO 禁止の場合、マスタは SSR:TDRE ビットが"1"であれば 1 バイト受信ごとにウェイトを発生(IBCR:INT=1, SSR:RDRF=1)し、IBCR:WSEL ビットにしたがって IBCR レジスタの ACKE ビットの設定で ACK または NACK 応答します。SSR:TDRE ビットが"0"であれば、IBCR レジスタの ACKE ビットの設定で ACK 応答であればウェイトは発生せず(IBCR:INT=0)に次のデータを受信し、NACK応答であればウェイトが発生します(IBCR:INT=1)。

FIFO 許可の場合、受信バイト数設定と同じバイト数分を受信すると SSR:RDRF ビットがセットされます。割込みフラグはSSR:TDRE ビットが"1"のときにセットし、I²Cバスをウェイトします。IBCR:WSEL=0 の場合、SSR:TDRE ビットが"1"になると ACKE ビットの設定で NACK であれば NACK 応答して割込みフラグを"1"にします。IBCR:WSEL=1 の場合、最終バイト受信後ウェイトが発生しますのでそのウェイト中に IBCR:ACKE ビットを設定し、割込みフラグを"0"にクリアした後、IBCR:ACKE の設定にしたがって ACK または NACK 応答します。NACK 出力した場合でも受信データとして受信 FIFO に格納します。

割込みによるウェイトは以下を参照してください。

表 40-19 DMA モードが禁止時(SSR:DMA=0)のマスタデータ受信時の IBCR:WSEL ビット

WSEL	動作
0	第二バイト以降、SSR:TDRE ビットが"1"でアクノリッジ後、割込みフラグ(IBCR:INT)を"1"、SCL を"L"にしてウェイト状態にします。
1	第二バイト以降、SSR:TDRE ビットが"1"でマスタが 1 バイトのデータを受信後、割込みフラグ(IBCR:INT)を"1"、SCL を"L"にしてウェイト状態にします。

スレーブからデータを受信する場合の手順の一例を以下に示します。

■ 受信 FIFO が禁止されている場合

① Slave Address (データ方向ビットも含む)を TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後 ACK を受信し、割込みフラグ(IBCR:INT)が"1"になります。

③ IBCR:WSEL ビット更新と共に割込みフラグビット(IBCR:INT)に"0"を書込み、I²C バスのウェイトを解除します。

④ 1 バイト受信後 IBCR:WSEL=0 の場合アクノリッジ送信後、IBCR:WSEL=1 の場合 1 バイト受信直後割込みフラグを"1"にして I²C バスをウェイトします。所定のデータ数を受信するまで③～④を繰り返します。

⑤ 最終データ受信後、NACK を出力し、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を発生させます。

■ 送受信 FIFO が許可されている場合

① FBYTE レジスタに受信数を設定します。

② Slave Address (データ方向ビットも含む)と受信数分ダミーのデータを TDR レジスタに書きます。

③ IBCR:MSS ビットに"1"を書きます。

④ SSR:TDRE ビットが"0"の間、ACK 応答し、受信し続けます。その受信中に FBYTE に設定数分受信すると SSR:RDRF を"1"にします。SSR:RDRF が"1"になったところで RDR レジスタを読み出します。

⑤ SSR:TDRE ビットが"1"になると IBCR:WSEL=0 の場合 NACK 出力後、IBCR:WSEL=1 の場合 1 バイト受信直後割込みフラグを"1"にして I^2C バスをウェイトします。

⑥ IBCR:WSEL=1 の場合、IBCR:ACKE ビットを"0"に設定し、IBCR:WSEL=0 の場合 IBCR: ACKE ビットの設定は必要なく、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を発生させます。

■ DMA モードが許可の場合(SSR:DMA=1)

データ方向ビット(R/W)が"1"の場合、スレーブから送信されたデータを受信します。

FIFO 禁止の場合、マスタは SSR:TDRE ビットが"1"であれば 1 バイト受信ごとにウェイトを発生(SSR:TBI=1, SSR:RDRF=1)し、IBCR:WSEL ビットにしたがって IBCR レジスタの ACKE ビットの設定で ACK または NACK 応答します。SSR:TDRE ビットが"0"であれば、IBCR レジスタの ACKE ビットの設定で ACK 応答であればウェイトは発生せずに次のデータを受信し、NACK 応答であればウェイトが発生します(IBCR:INT=1)。

FIFO 許可の場合、受信バイト数設定と同じバイト数分を受信すると SSR:RDRF ビットがセットされます。送信バスアイドルフラグ(SSR:TBI)は SSR:TDRE ビットが"1"のときにセットし、I^2C バスをウェイトします。
IBCR:WSEL=0 の場合、SSR:TDRE ビットが"1"になると ACKE ビットの設定で NACK であれば NACK 応答して割込みフラグ(IBCR:INT)および送信バスアイドルフラグ(SSR:TBI)を"1"にします。IBCR:WSEL=1 の場合、最終バイト受信後ウェイト(SSR:TBI=1)が発生しますのでそのウェイト中に IBCR:ACKE ビットを設定し、送信バスアイドルフラグ(SSR:TBI)をクリア後、IBCR:ACKE の設定にしたがって ACK または NACK 応答します。NACK 出力した場合でも受信データとして受信 FIFO に格納します。

割込みによるウェイトは以下を参照してください。

表 40-20 DMA モードが許可時(SSR:DMA=1)のマスタデータ受信時の IBCR:WSEL ビット

WSEL	動作
0	第二バイト以降、SSR:TDRE ビットが"1"でアクノリッジ後、送信バスアイドルフラグ(SSR:TBI)を"1"、SCL を"L"にしてウェイト状態にします。 第二バイト以降、受信 FIFO 未使用時にアクノリッジ後に受信データフルフラグ(SSR:RDRF)が"1"セットされている場合、SCL を"L"にしてウェイト状態にします。
1	第二バイト以降、SSR:TDRE ビットが"1"でマスタが 1 バイトのデータを受信後、割込みフラグ(SSR:TBI)を"1"、SCL を"L"にしてウェイト状態にします。 第二バイト以降、受信 FIFO 未使用時に受信データフルフラグ(SSR:RDRF)が"1"にセットされるとデータ受信後、SCL を"L"にしてウェイト状態にします。

スレーブからデータを受信する場合の手順の一例を以下に示します。

■ 受信 FIFO が禁止されている場合

① Slave Address (データ方向ビットも含む)を TDR レジスタにセットし、IBCR:MSS ビットに"1"を書きます。

② Slave Address 送信後 ACK を受信し、送信バスアイドルフラグ(SSR:TBI)が"1"になります。

③ TDR レジスタに送信するデータを書きこみ、I^2C バスのウェイトを解除します。

④ 1 バイト受信後 IBCR:WSEL=0 の場合アクノリッジ送信後、IBCR:WSEL=1 の場合 1 バイト受信直後送信バスアイドルフラグ(SSR:TBI)および受信データフルフラグ(SSR:RDRF)(*2)を"1"にして I^2C バスをウェイトします。

⑤ IBCR:WSEL ビット更新し、RDR レジスタを読み出しダミーのデータを TDR レジスタに書きます。

⑥ 1 バイト受信後 IBCR:WSEL=0 の場合アクノリッジ送信後、IBCR:WSEL=1 の場合 1 バイト受信直後送信バスアイドルフラグ(SSR:TBI)および受信データフルフラグ(SSR:RDRF)(*2)を"1"にして I²C バスをウェイトします。 所定のデータ数を受信するまで⑤～⑥を繰り返します。

⑦ 最終データ受信後、NACK を出力し、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を発生させます。

■ 送受信 FIFO が許可されている場合

① FBYTE レジスタに受信数を設定します。

② Slave Address (データ方向ビットも含む)と受信数分ダミーのデータを TDR レジスタに書きます。

③ IBCR:WSEL=0 の場合は ACKE ビットの設定で NACK にし、IBCR:MSS ビットに"1"を書きます。

④ SSR:TDRE ビットが"0"の間、ACK 応答し、受信し続けます。その受信中に FBYTE に設定数分受信すると SSR:RDRF を"1"にします。SSR:RDRF が"1"になったところで RDR レジスタを読み出します。

⑤ SSR:TDRE ビットが"1"になると IBCR:WSEL=0 の場合 NACK 出力後、割込みフラグを"1"にして I²C バスをウェイトします。IBCR:WSEL=1 の場合 1 バイト受信直後送信バスアイドルフラグ(SSR:TBI)を"1"にして I²C バスをウェイトします

⑥ IBCR:WSEL=1 の場合、IBCR:ACKE ビットを"0"に設定し、IBCR:WSEL=0 の場合 IBCR: ACKE ビットの設定は必要なく、IBCR:MSS ビットに"0"または IBCR:SCC ビットに"1"を設定し、ストップ条件または反復スタート条件を発生させます。

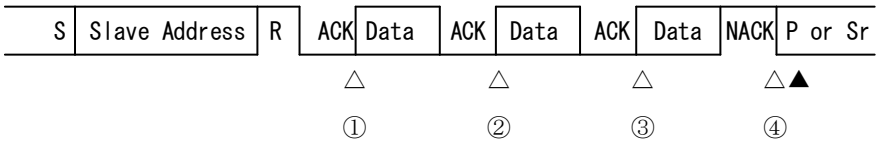
*1: DMA モードが許可(SSR:DMA=1)で SSR:TBI ビットが"1"で IBCR:INT ビットが"0"のときに反復スタート条件を発行する場合は、IBCR:INT ビットに"1"を書き込んだ後、IBCR:INT ビットが"1"にセットされていることを確認してから TDR にスレーブアドレスを書き込み、IBCR:SCC ビットに"1"を設定してください。

*2: IBCR:WSEL の設定に関係なく 1 バイト受信直後に受信データフルフラグ(SSR:RDRF)は"1"にセットされます。第二バイト以降で受信データフルフラグ(SSR:RDRF)が"1"にセットされているとき、IBCR:WSEL=0 の場合アクノリッジ送信後、IBCR:WSEL=1 の場合 1 バイト受信直後に I²C バスをウェイトします。

＜注意事項＞

- 7 ビットスレーブアドレスの検出を許可しているとき(ISBA:SAEN=1)にマスタモード時に 7 ビットスレーブアドレスを指定することは禁止です。
- SSR:TDRE が"0"のとき、オーバランエラーが発生しても IBCR:ACKE ビットの設定にしたがってアクノリッジを出力し、次の処理を行います。
- 送受信中に IBCR レジスタを変更する場合、割込みフラグ(IBCR:INT)が"1"または DMA モードが許可時(SSR:DMA=1)は送信バスアイドルフラグ(SSR:TBI1)が"1"のときに変更してください。
- DMA モードが禁止(SSR:DMA=0)でマスタ受信時、TDR レジスタにダミーデータを書き込み、割込みフラグ(IBCR:INT)が"1"になるタイミングで SSR:TDRE ビットが"0"の場合、割込みフラグ(IBCR:INT)は"0"のままで次のデータを受信します。
- DMA モードが許可(SSR:DMA=1)でマスタ受信時、TDR レジスタにダミーデータを書き込み、送信バスアイドルフラグ(SSR:TBI)が"1"になるタイミングで SSR:TDRE ビットが"0"の場合、送信バスアイドルフラグ(SSR:TBI)は"0"のままで次のデータを受信します。
- 受信 FIFO が許可、IBCR:WSEL=0 のときにデータを受信する場合、最終ビット受信後 SSR:RDRF ビットが"1"となり、ACK 送信後割込みフラグ(IBCR:INT)が"1"となります。

図 40-83 FIFO 禁止によるマスタ受信の割込み 1 (SSR:DMA=0, IBCR:WSEL=0, IBSR:RSA=0)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ INT=“0”書込みにより割込みが“0”にクリア

② 1バイト受信＋アクノリッジ送信により割込み発生

－ 受信データを読み出し後、INT=“0”書込み

③ 1バイト受信＋アクノリッジ送信により割込み発生

－ 受信データを読み出し後、ACKE=“0”に設定し、INT=“0”書込み

④ 1バイト受信＋アクノリッジ送信したことにより割込み発生

－ MSS=“0”又はMSS=“1”,SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDREビットは“1”

図 40-84 FIFO 禁止によるマスタ受信の割込み 2 (SSR:DMA=0, IBCR:WSEL=1, IBSR:RSA=0)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ INT=“0”書込みにより割込みが“0”にクリア

② 1バイト受信により割込み発生

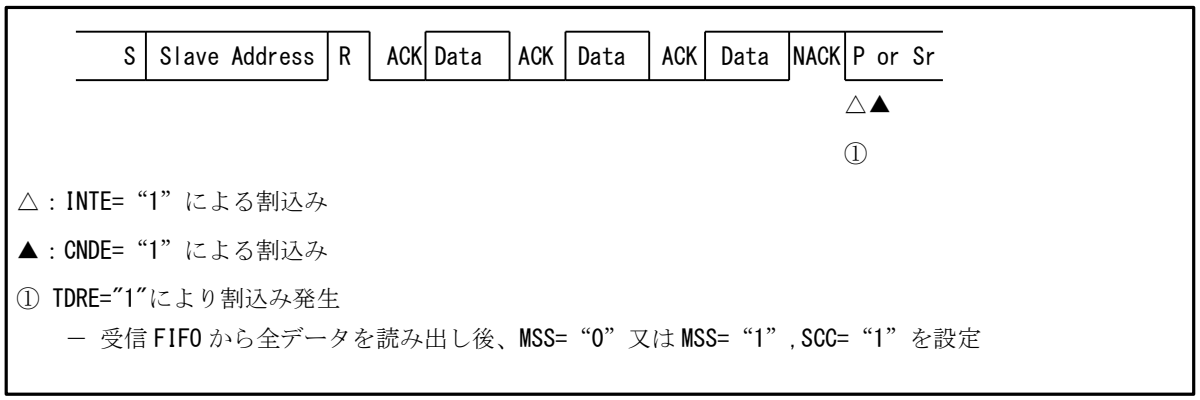
－ 受信データを読み出し後、INT=“0”書込み

③ 1バイト受信により割込み発生

－ 受信データを読み出し後、ACKE=“0”に設定し、MSS=“0”又はMSS=“1”,SCC=“1”を設定

*) 割込みフラグ(INT)発生時、TDREビットは“1”

図 40-85 FIFO 許可によるマスタ受信の割込み 3 (SSR:DMA=0, IBCR:WSEL=0, IBCR:ACKE=0, IBSR:RSA=0)



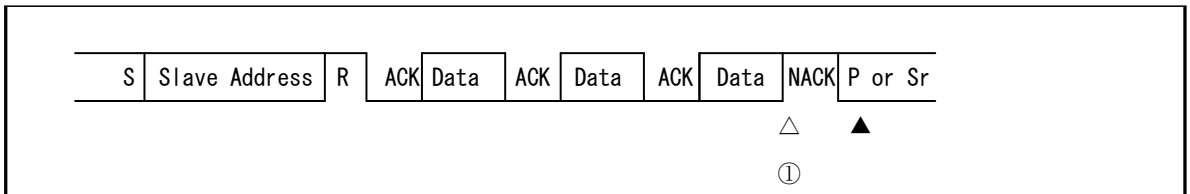
△ : INTE=“1”による割込み

▲ : CNDE=“1”による割込み

① TDRE=”1”により割込み発生

－ 受信 FIFO から全データを読み出し後、MSS=“0”又は MSS=“1”, SCC=“1”を設定

図 40-86 FIFO 許可によるマスタ受信の割込み 4 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0)



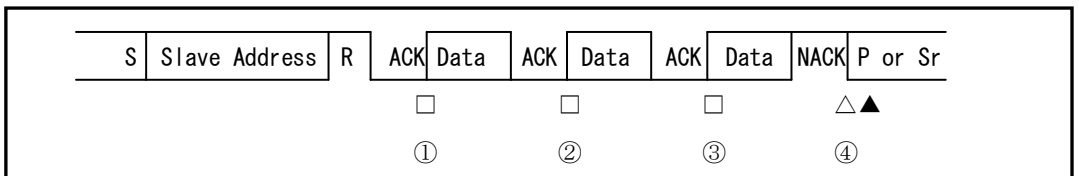
△ : INTE=“1”による割込み

▲ : CNDE=“1”による割込み

① TDRE=”1”により割込み発生

－ 受信 FIFO から全データを読み出し後、ACKE=“0”, MSS=“0”又は MSS=“1”, SCC=“1”を設定

図 40-87 FIFO 禁止によるマスタ受信の割込み 5 (SSR:DMA=1, IBCR:WSEL=0, IBSR:RSA=0)



△ : INTE=“1”による割込み

▲ : CNDE=“1”による割込み

□ : TBIE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタにダミーのデータを書込み

② 1バイト受信＋アクノリッジ送信により割込み発生

－ 受信データを読み出し後、TDR レジスタにダミーのデータを書込み

③ 1バイト受信＋アクノリッジ送信により割込み発生

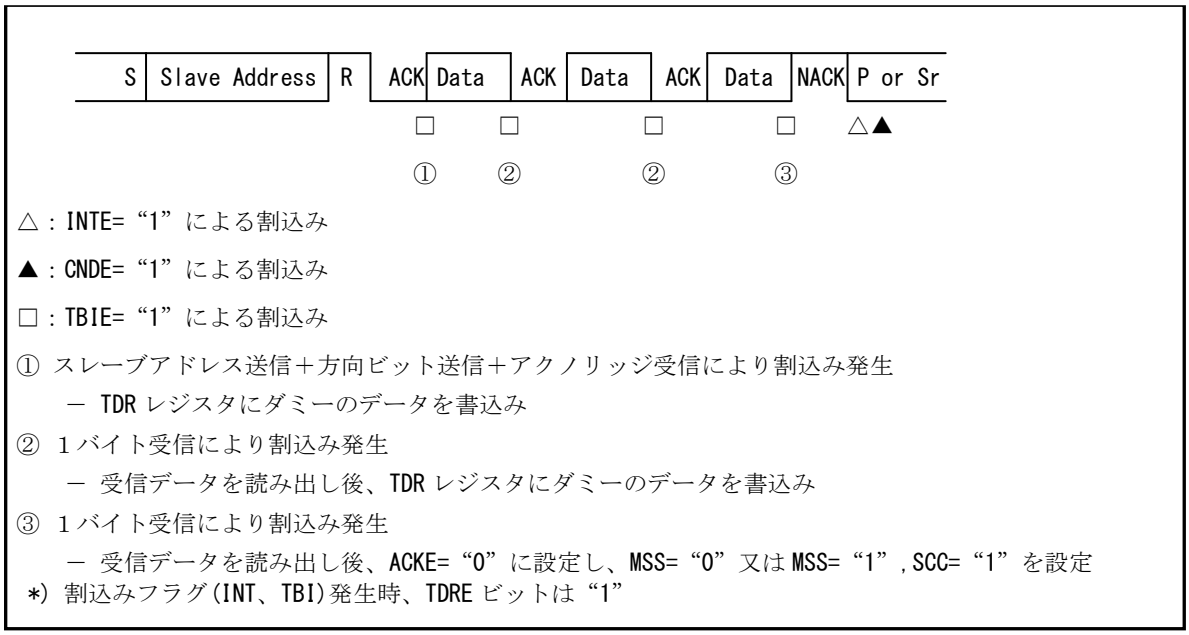
－ 受信データを読み出し後、ACKE=“0”に設定し、TDR レジスタにダミーのデータを書込み

④ 1バイト受信＋アクノリッジ送信したことにより割込み発生

－ MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT、TBI)発生時、TDRE ビットは“1”

図 40-88 FIFO 禁止によるマスタ受信の割込み 6 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

□：TBIE=“1”による割込み

① スレーブアドレス送信＋方向ビット送信＋アクノリッジ受信により割込み発生

－ TDR レジスタにダミーのデータを書込み

② 1バイト受信により割込み発生

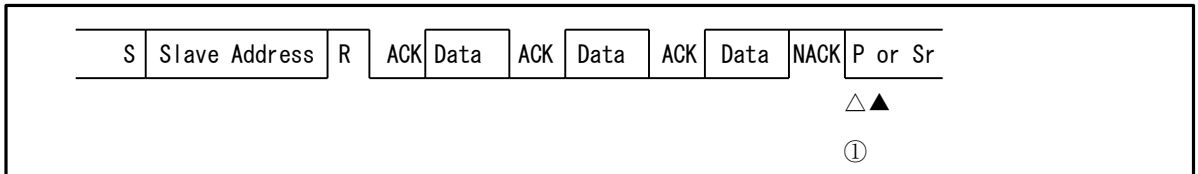
－ 受信データを読み出し後、TDR レジスタにダミーのデータを書込み

③ 1バイト受信により割込み発生

－ 受信データを読み出し後、ACKE=“0”に設定し、MSS=“0”又は MSS=“1”, SCC=“1”を設定

*) 割込みフラグ(INT、TBI)発生時、TDRE ビットは“1”

図 40-89 FIFO 許可によるマスタ受信の割込み 7 (SSR:DMA=1, IBCR:WSEL=0, IBCR:ACKE=0, IBSR:RSA=0)



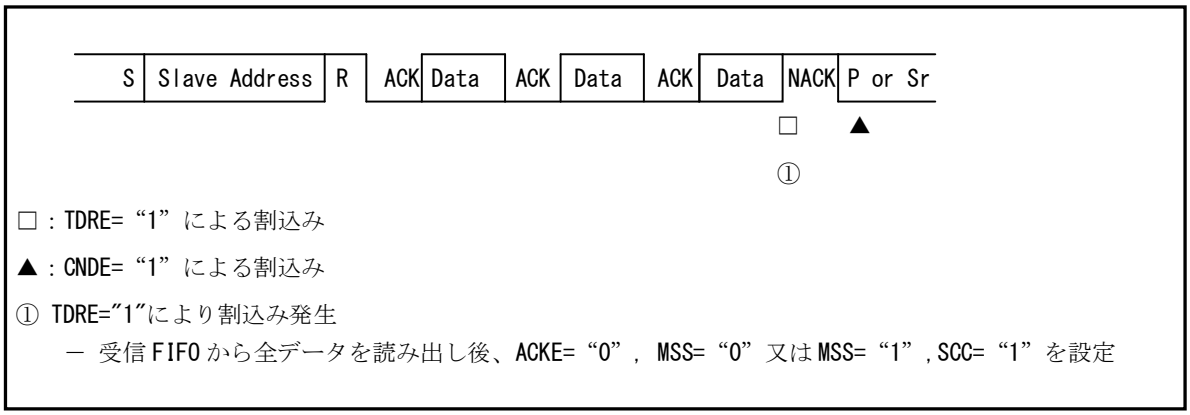
△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① TDRE="1"により割込み発生

－ 受信 FIFO から全データを読み出し後、MSS=“0”又は MSS=“1”, SCC=“1”を設定

図 40-90 FIFO 許可によるマスタ受信の割込み 8 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0)



40.8.3.6 アービトレーションロスト

アービトレーションロストについて説明します。

マスタがほかのマスタからのデータとデータが衝突し、送信したデータと異なるデータを受信した場合、アービトレーションロストと判断し IBCR:MSS ビットを"0"、IBSR:AL ビットを"1"にしてスレーブモードとして動作可能となります。IBSR:AL ビットは、以下の条件で"0"にクリアすることができます。

- IBCR:MSS ビットへの"1"書込み
- IBCR:INT ビットへの"0"書込み
- IBSR:AL ビット=1, IBSR:SPC ビット=1 のときに IBSR:SPC ビットへの"0"書込み
- I^2C インタフェースの禁止(ISMK:EN ビット=0)

アービトレーションロストが発生すると IBCR:WSEL の設定にしたがって割込みフラグ(IBCR:INT)を"1"にし、I^2C バスの SCL を"L"にします。

40.8.3.7 マスタモードのウェイト

マスタモードのウェイトについて説明します。

IBSR:BB ビットが"1"のときに IBCR:MSS ビットに"1"を設定するとスレーブモードとして動作していなければ IBSR:BB ビットが"1"の間、マスタモードをウェイトし、IBSR:BB ビットが"0"になってからスタート条件を送信します。マスタモードがウェイト中かどうかは IBCR:MSS ビットと IBCR:ACT ビットで判断できます(IBCR:MSS=1, IBCR:ACT=0 であればウェイト状態)。IBCR:MSS ビットに"1"を設定後、スレーブモードとして動作する場合、IBSR:AL ビットを"1"、IBCR:MSS ビットを"0"、IBCR:ACT ビットを"1"にします。

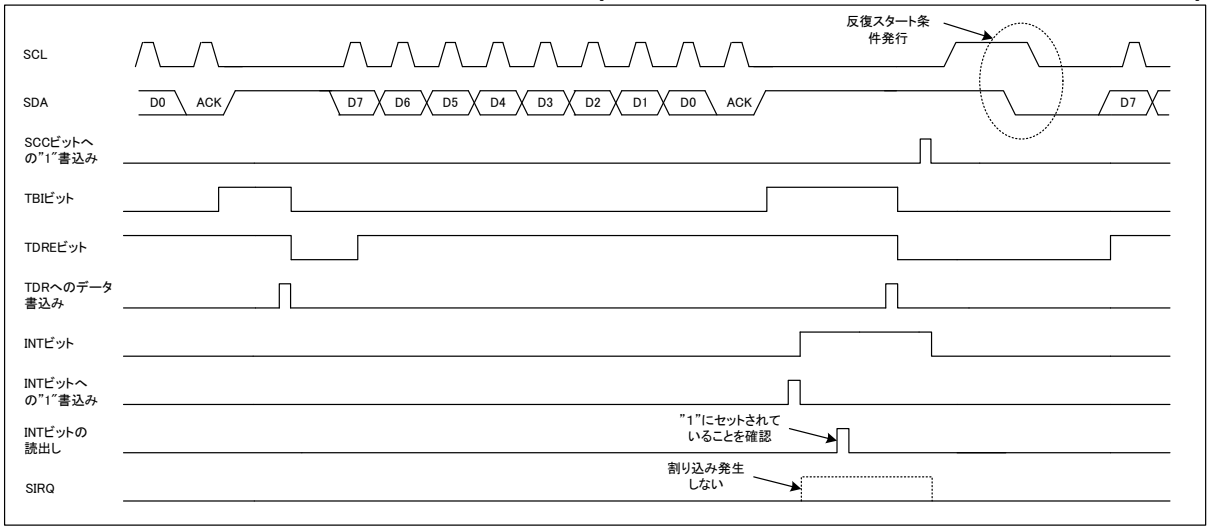
40.8.3.8 DMA モードが許可時(SSR:DMA=1)の反復スタート条件発行

DMA モードが許可時(SSR:DMA=1)の反復スタート条件について説明します。

送信バスアイドル中(SSR:TBI=1)で割込みフラグ(IBCR:INT)が"0"のときに、TDR レジスタにスレーブアドレスを書き込んだ場合、送信動作を開始してしまい、反復スタート条件を発行できません。

そのため送信バスアイドル中(SSR:TBI=1)で割込みフラグ(IBCR:INT)が"0"のときに、反復スタート条件を発行する場合は最初に割込みフラグ(IBCR:INT)へ"1"を書込み、その後割込みフラグ(IBCR:　INT)が"1"にセットされていることを確認します。このとき、SIRQ 割込みは発生しません。次に TDR レジスタにスレーブアドレスを書込み、その後反復スタートを発行(IBCR:SCC=1)します。

図 40-91 DMA モードが許可時の反復スタート条件発行(SSR:DMA=1, IBCR:WSEL=0, IBSR:RSA=0, ACK 応答)



40.8.4 I²C スレーブモード

I²C スレーブモードについて示します。

スレーブモードは(反復)スタート条件を検出し、ISBA レジスタと ISMK レジスタとの組み合わせと受信したアドレスが一致すると ACK 応答し、スレーブモードとして動作します。

＜注意事項＞
スタート条件検出後のアドレスデータの転送中、または bit2～bit9(アクノリッジビット) の転送中に、再度スタート条件を検出した場合、バスエラーを検出(IBCR:BER=1)し、受信を中断するため、次のデータ受信ができません。この場合、割込みフラグ(IBCR:INT)のクリア後にマスタからスタート条件の再送処理が必要になります。

40.8.4.1 スレーブアドレス一致検出

スレーブアドレス一致検出について示します。

(反復)スタート条件を検出すると次のデータの7ビットがアドレスとして受信します。ISMK レジスタで"1"がセットされているビットについて ISBA レジスタと受信アドレスの各ビットを比較し、一致した場合 ACK を出力します。

表 40-21 スレーブアドレスに対するアクノリッジ出力直後の動作

送信 FIFO	受信 FIFO	送信 FIFO 状態	受信 FIFO 状態	データ方向ビット (R/W)	アクノリッジ直後の動作	
					アクノリッジが ACK	アクノリッジが NACK
禁止	禁止	―	―	0	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にして**ウェイト**。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままで**ウェイト**なし	IBCR:INT ビットは"0"のままで**ウェイト**なし
				1		
禁止	許可	―	データなし	0	IBCR:INT ビットは"0"のままでウェイトなし	IBCR:INT ビットは"0"のままでウェイトなし。
			データあり		IBCR:INT ビットを"1"にしてウェイト	
			―	1	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	
許可	禁止	―	―	0	SSR:TDRE ビットが"1"であれば、IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	IBCR:INT ビットは"0"のままでウェイトなし
				1		
許可	許可	―	データなし	0	IBCR:INT ビットは"0"のままでウェイトなし	IBCR:INT ビットは"0"のままでウェイトなし
			データあり		IBCR:INT ビットを"1"にしてウェイト	
			―	1	SSR:TDRE ビットが"1"であれば IBCR:INT ビットを"1"にしてウェイト。SSR:TDRE ビットが"0"であれば、IBCR:INT ビットは"0"のままでウェイトなし	

■ 予約アドレス検出

第一バイト目で予約アドレス("0000xxxx"または"1111xxxx")と一致した場合、送受信 FIFO の許可に依存せずに 8 ビット目のデータ受信後、IBCR:INT ビットを"1"にして I^2C バスをウェイトします。このとき受信データを読み出し、スレーブとして動作させたい場合には IBCR:ACKE を"1"にセットしてデータ方向ビット(IBSR:TRX)を確認し、送信方向であれば送信データを TDR に書き込み、IBCR:INT ビットをクリアします。その後、スレーブとして動作します。IBCR:ACKE を"0"にした場合には、アクノリッジ出力後スレーブとして動作を行いません。

40.8.4.2 データ方向ビット

データ方向ビットについて示します。

アドレス受信後、データの送受信を決めるデータ方向ビットを受信します。このビットが"0"のときマスタからの送信を示し、スレーブとしてはデータを受信することになります。

40.8.4.3 スレーブによる受信

スレーブによる受信について説明します。

スレーブアドレスが一致しデータ方向ビットが"0"のとき、スレーブモードによる受信を示します。スレーブモードによる受信の手順の一例は以下のようになります。

- DMA モードが禁止(SSR:DMA=0)の場合
  受信 FIFO が禁止されている場合

  ① ACK 送信後、割込みフラグ(IBCR:INT)を"1"にして I²C バスをウェイトします。IBCR:MSS ビット, IBCR:ACT ビットと IBSR:FBT ビットでスレーブアドレス一致による割込みと判断し、IBCR:ACKE ビットに"1"、割込みフラグ(IBCR:INT)に"0"を書いて I²C バスのウェイトを解除します。 (「表 40-21 スレーブアドレスに対するアクノリッジ出力直後の動作」参照)

  ② 1 バイトのデータを受信後、IBCR:WSEL の設定にしたがって割込みフラグ(IBCR:INT)を"1"にして I²C バスをウェイトします。

  ③ RDR レジスタから受信したデータを読み出し、IBCR:ACKE ビットを設定後割込みフラグ(IBCR:INT)に"0"を書いて I²C バスのウェイトを解除します。

  ④ ストップ条件または反復スタート条件を検出するまで②～③を繰り返します。

  受信 FIFO が許可されている場合

  ① NACK の検出または 受信 FIFO がフルになると割込みフラグ(IBCR:INT)は"1"になり、I²C バスをウェイトします。ストップ条件, 反復スタート条件を検出した場合、IBSR:SPC ビット, IBSR:RSC ビットを"1"にして割込みフラグ(IBCR:INT)は"1"になりません(I²C バスのウェイトなし)。受信 FIFO は FBYTE レジスタの設定値と受信したデータ数が一致すると SSR:RDRF ビットを"1"にします。そのとき、SMR:RIE ビットが"1"になっていると受信割込みが発生します。

  ② 割込みフラグ(IBCR:INT)が"1"になった場合、RDR レジスタから受信したデータを読み出し、すべてのデータを読出し後割込みフラグに"0"を書いて I²C バスのウェイトを解除します。ストップ条件または反復スタート条件を検出した場合、受信したデータを RDR レジスタからすべて読み出し、IBSR:SPC ビットまたは IBSR:RSC ビットを"0"にクリアします。

- DMA モードが許可(SSR:DMA=1)の場合
  受信 FIFO が禁止されている場合

  ① ACK 送信後、割込みフラグ(IBCR:INT)を"1"にして I²C バスをウェイトします。 IBCR:MSS ビット, IBCR:ACT ビットと IBSR:FBT ビットでスレーブアドレス一致による割込みと判断し、IBCR:ACKE ビットに"1"、割込みフラグ(IBCR:INT)に"0"を書いて I²C バスのウェイトを解除します。 (「表 40-21 スレーブアドレスに対するアクノリッジ出力直後の動作」参照)

② 1 バイトのデータを受信後、1 バイト受信直後に受信データフルフラグ(SSR:RDRF)を"1"にセットします。受信データフルフラグ(SSR:RDRF)が"1"にセットされているとき、IBCR:WSEL=0 の場合アクノリッジ送信後、IBCR:WSEL=1 の場合 1 バイト受信直後に I²C バスをウェイトします。

③ IBCR:ACKE ビットを設定後 RDR レジスタから受信したデータを読出しにより受信データフルフラグ(SSR:RDRF)を"0"にクリアして I²C バスのウェイトを解除します。

④ ストップ条件または反復スタート条件を検出するまで②～③を繰り返します。

受信 FIFO が許可されている場合

① NACK の検出により割込みフラグ(IBCR:INT)は"1"になり I²C バスをウェイトします。受信 FIFO がフルになると、I²C バスをウェイトします。ストップ条件, 反復スタート条件を検出した場合、IBSR:SPC ビット, IBSR:RSC ビットを"1"にして割込みフラグ(IBCR:INT)は"1"になりません(I²C バスのウェイトなし)。受信 FIFO は FBYTE レジスタの設定値と受信したデータ数が一致すると SSR:RDRF ビットを"1"にします。そのとき、SMR:RIE ビットが"1"になっていると受信割込みが発生します。

② 割込みフラグ(IBCR:INT)が"1"になった場合、RDR レジスタから受信したデータを読み出し、すべてのデータを読み出し後割込みフラグに"0"を書いて I²C バスのウェイトを解除します。受信 FIFO がフルになった場合、RDR レジスタから 1 回でも受信したデータを読み出せば I²C バスのウェイトを解除します。ストップ条件または反復スタート条件を検出した場合、受信したデータを RDR レジスタからすべて読み出し、IBSR:SPC ビットまたは IBSR:RSC ビットを"0"にクリアします。

図 40-92 FIFO 禁止によるスレーブ受信の割込み 1 (SSR:DMA=0, IBCR:WSEL=0, IBSR:RSA=0)

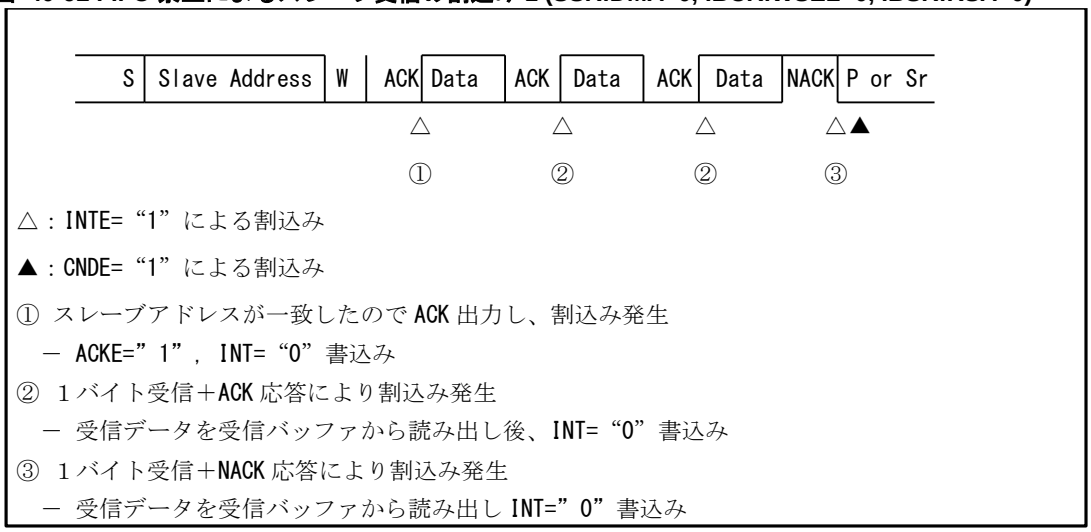
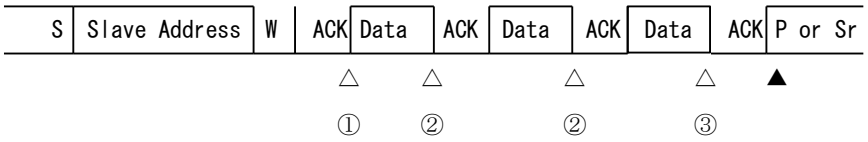


図 40-93 FIFO 禁止によるスレーブ受信の割込み 2 (SSR:DMA=0, IBCR:WSEL=1, IBSR:RSA=0)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① スレーブアドレスが一致したので ACK 出力し、割込み発生
　－ ACKE=“1”, INT=“0”書込み

② 1バイト受信により割込み発生
　－ 受信データを受信バッファから読み出し後、INT=“0”書込み

③ 1バイト受信により割込み発生
　－ 受信データを受信バッファから読み出し後、INT=”0”書込み

図 40-94 FIFO 禁止によるスレーブ受信の割込み 3 (SSR:DMA=0, IBCR:WSEL=1, IBSR:RSA=0)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① スレーブアドレスが一致したので ACK 出力し、割込み発生
　－ ACKE=“1”, INT=“0”書込み

② 1バイト受信により割込み発生
　－ 受信データを受信バッファから読み出し後、INT=“0”書込み

③ NACK 応答により割込み発生
　－ INT=“0”書込み

図 40-95 FIFO 許可によるスレーブ受信の割込み 4 (SSR:DMA=0, IBSR:RSA=0)

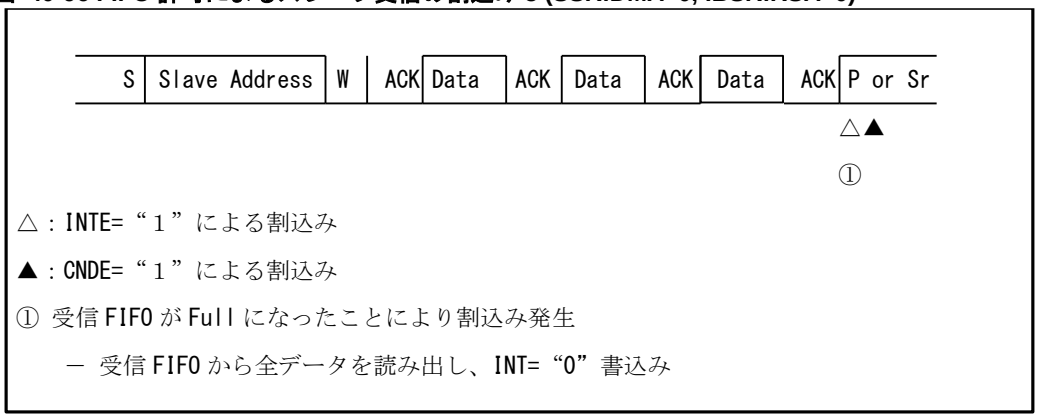


△：INTE=“１”による割込み

▲：CNDE=“１”による割込み

① ストップ条件又は反復スタート条件検出により割込み発生
　－ 受信 FIFO から全データの読み出し

図 40-96 FIFO 許可によるスレーブ受信の割込み 5 (SSR:DMA=0, IBSR:RSA=0)



△：INTE=“１”による割込み

▲：CNDE=“１”による割込み

① 受信 FIFO が Full になったことにより割込み発生

－ 受信 FIFO から全データを読み出し、INT=“0”書込み

図 40-97 FIFO 禁止によるスレーブ受信の割込み 6 (SSR:DMA=0, IBCR:WSEL=0, IBSR:RSA=1)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

① 予約アドレス(“0000xxxx”又は“1111xxxx”)が一致したので割込み発生

－ 受信データを読み出し、ACKE=“1”, INT=“0”書込み

② １バイト受信＋アクノリッジ出力により割込み発生

－ INT=“0”書込み

③ １バイト受信＋アクノリッジ出力により割込み発生

－ INT=”0”書込みにより割込み

図 40-98 FIFO 禁止によるスレーブ受信の割込み 7 (SSR:DMA=1, IBCR:WSEL=0, IBSR:RSA=0)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

■：RIE=“1”による割込み

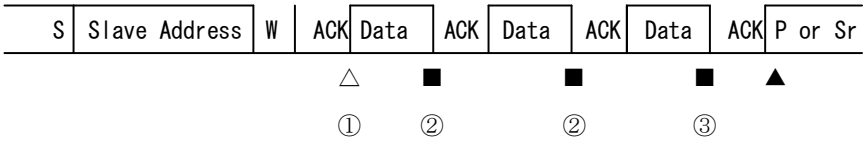
① スレーブアドレスが一致したので ACK 出力し、割込み発生
　－ ACKE=”1”, INT=“0”書込み

② 1 バイト受信により割込み発生(I2C バスはウエイトしない)
　－ 受信データを受信バッファから読み出し

③ ACK 応答により I2C バスウエイト
　－ 受信データを受信バッファから読み出し

④ 1 バイト受信＋NACK 応答により割込み発生
　－ 受信データを受信バッファから読み出し INT=”0”書込み

図 40-99 FIFO 禁止によるスレーブ受信の割込み 8 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0)



△：INTE=“1”による割込み

▲：CNDE=“1”による割込み

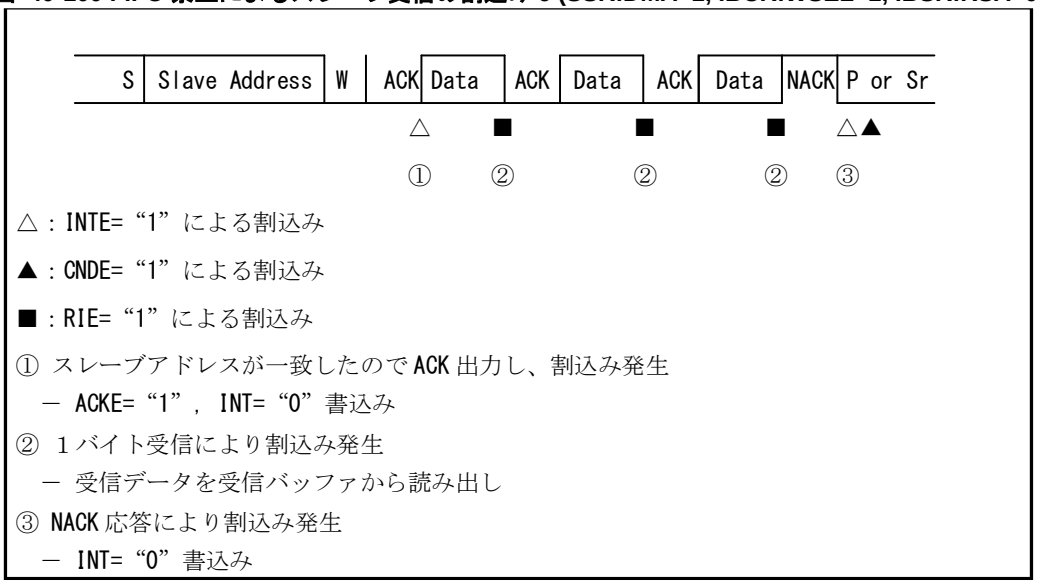
■：RIE=“1”による割込み

① スレーブアドレスが一致したので ACK 出力し、割込み発生
　－ ACKE=“1”, INT=“0”書込み

② 1 バイト受信により割込み発生
　－ 受信データを受信バッファから読み出し

③ 1 バイト受信により割込み発生
　－ 受信データを受信バッファから読み出し

図 40-100 FIFO 禁止によるスレーブ受信の割込み 9 (SSR:DMA=1, IBCR:WSEL=1, IBSR:RSA=0)



△ : INTE=“1”による割込み

▲ : CNDE=“1”による割込み

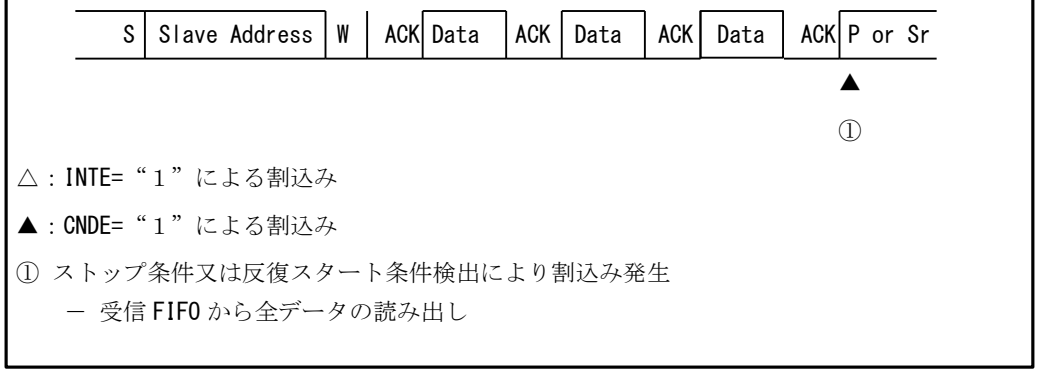
■ : RIE=“1”による割込み

① スレーブアドレスが一致したので ACK 出力し、割込み発生
　－ ACKE=“1”, INT=“0”書込み

② 1バイト受信により割込み発生
　－ 受信データを受信バッファから読み出し

③ NACK 応答により割込み発生
　－ INT=“0”書込み

図 40-101 受信 FIFO 許可によるスレーブ受信の割込み 10 (SSR:DMA=1, IBSR:RSA=0)

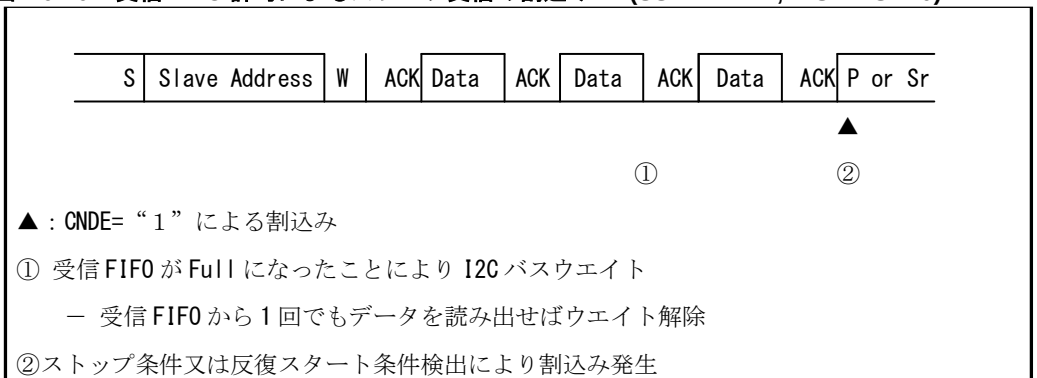


△ : INTE=“1”による割込み

▲ : CNDE=“1”による割込み

① ストップ条件又は反復スタート条件検出により割込み発生
　－ 受信 FIFO から全データの読み出し

図 40-102 受信 FIFO 許可によるスレーブ受信の割込み 11 (SSR:DMA=1, IBSR:RSA=0)

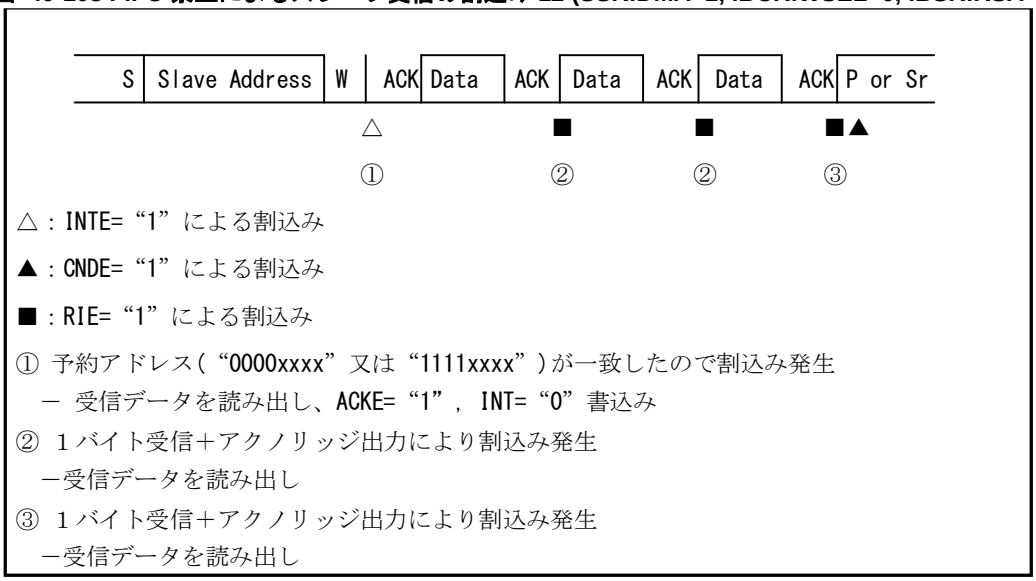


▲ : CNDE=“1”による割込み

① 受信 FIFO が Full になったことにより I2C バスウエイト
　－ 受信 FIFO から 1 回でもデータを読み出せばウエイト解除

②ストップ条件又は反復スタート条件検出により割込み発生
　－ 受信 FIFO から全データの読み出し

図 40-103 FIFO 禁止によるスレーブ受信の割込み 12 (SSR:DMA=1, IBCR:WSEL=0, IBSR:RSA=1)



40.8.4.4 スレーブによる送信

スレーブによる送信について説明します。

スレーブアドレスが一致しデータ方向ビットが"1"のとき、スレーブによる送信を示します。FIFO 禁止の場合、IBCR:WSEL の設定により､1バイト送信後またはアクノリッジ応答後に割込みフラグ(IBCR:INT)を"1"にし､ウェイトを発生します。(「表 40-21 スレーブアドレスに対するアクノリッジ出力直後の動作」参照)。

IBSR:RACK ビットによってマスタから出力されたアクノリッジを確認することができ、マスタから NACK 応答時、マスタが正しく受信できなかったか、データ受信の終了を示します。IBCR:WSEL=1 のときに NACK を検出した場合、割込みが発生しウェイトします。

40.8.5 バスエラー

バスエラーについて示します。

I²C バス上でデータの送受信中にストップ条件、(反復)スタート条件を検出するとバスエラーとして取り扱います。

40.8.5.1 バスエラー発生条件

バスエラー発生条件について示します。

バスエラーは以下の条件で IBCR:BER ビットを"1"にします。

- 第一バイト転送中に(反復)スタート条件またはストップ条件を検出
- データの 2～9 (アクノリッジ)ビット目で(反復)スタート条件またはストップ条件を検出

40.8.5.2 バスエラー動作

バスエラー動作について説明します。

送受信による割込みフラグ(IBCR:INT)が"1"になったときに IBCR:BER ビットを確認し、IBCR:BER ビットが"1"の場合エラー処理を行ってください。IBCR:BER ビットは IBCR:INT ビットに"0"を書くことによってクリアされます。

バスエラーによって IBCR:INT ビットは"1"にセットされますが、I²C バスの SCL を"L"にしてウェイト状態にはしません。

40.8.6 I²C のフローチャート例

I²C のフローチャート例について示します。

40.8.6.1 DMA モードが禁止時(SSR:DMA=0)の I²C フローチャート例 (FIFO 未使用時)

図 40-104 DMA モードが禁止時 (SSR:DMA=0) の I²C フローチャート例 (FIFO 未使用時) 1/3

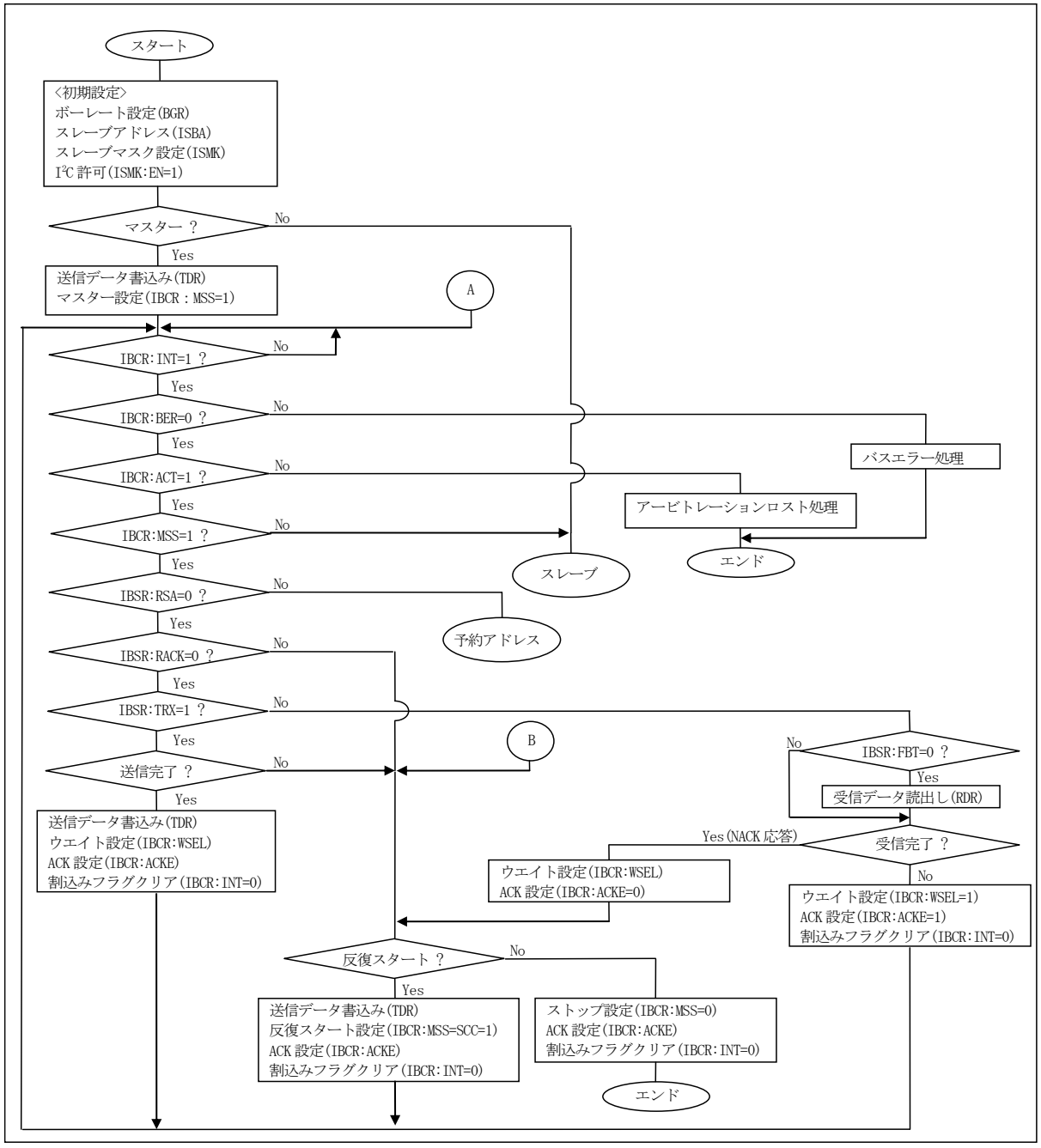


図 40-105 DMA モードが禁止時 (SSR:DMA=0) の I²C フローチャート例 (FIFO 未使用時) 2/3

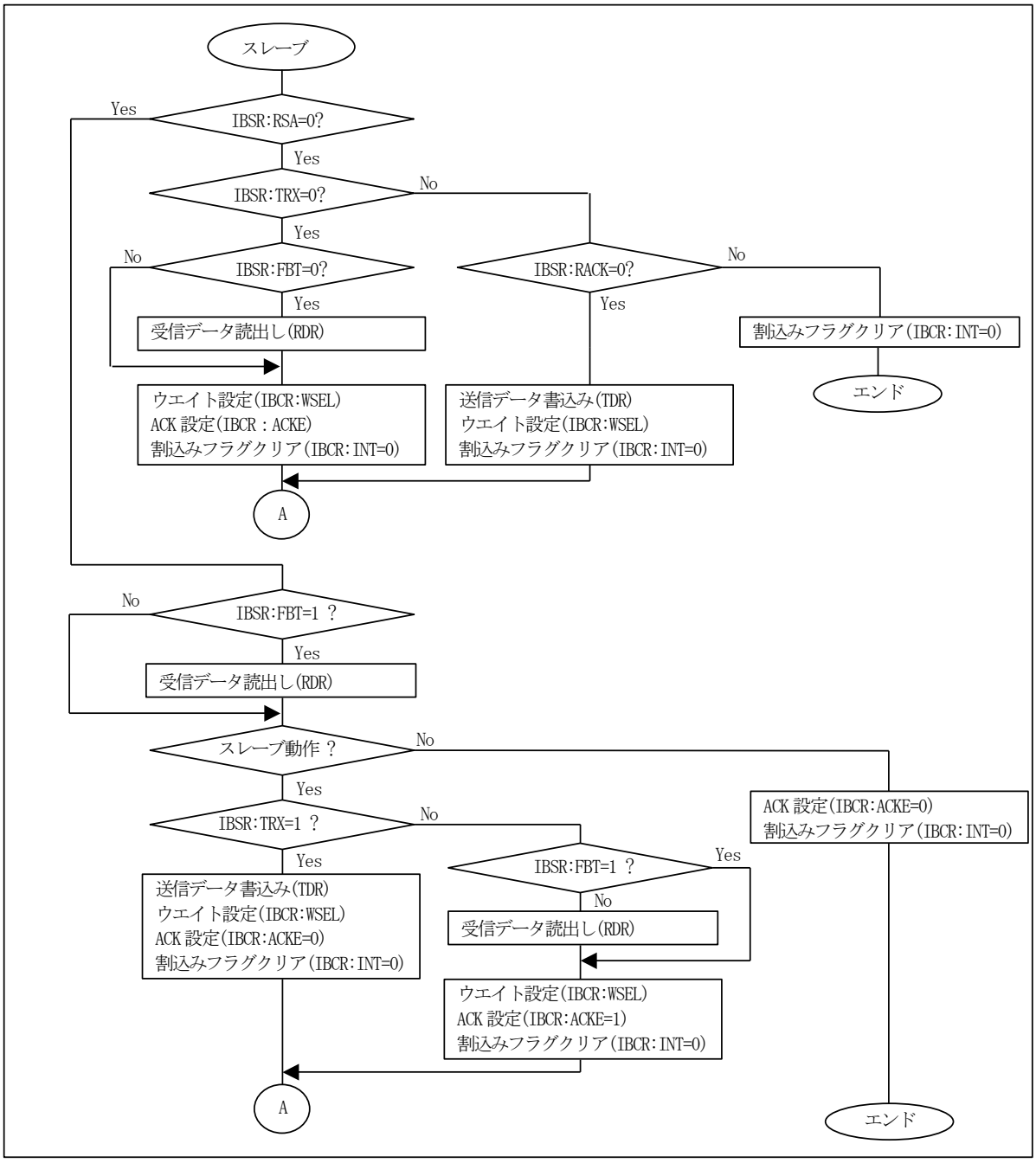
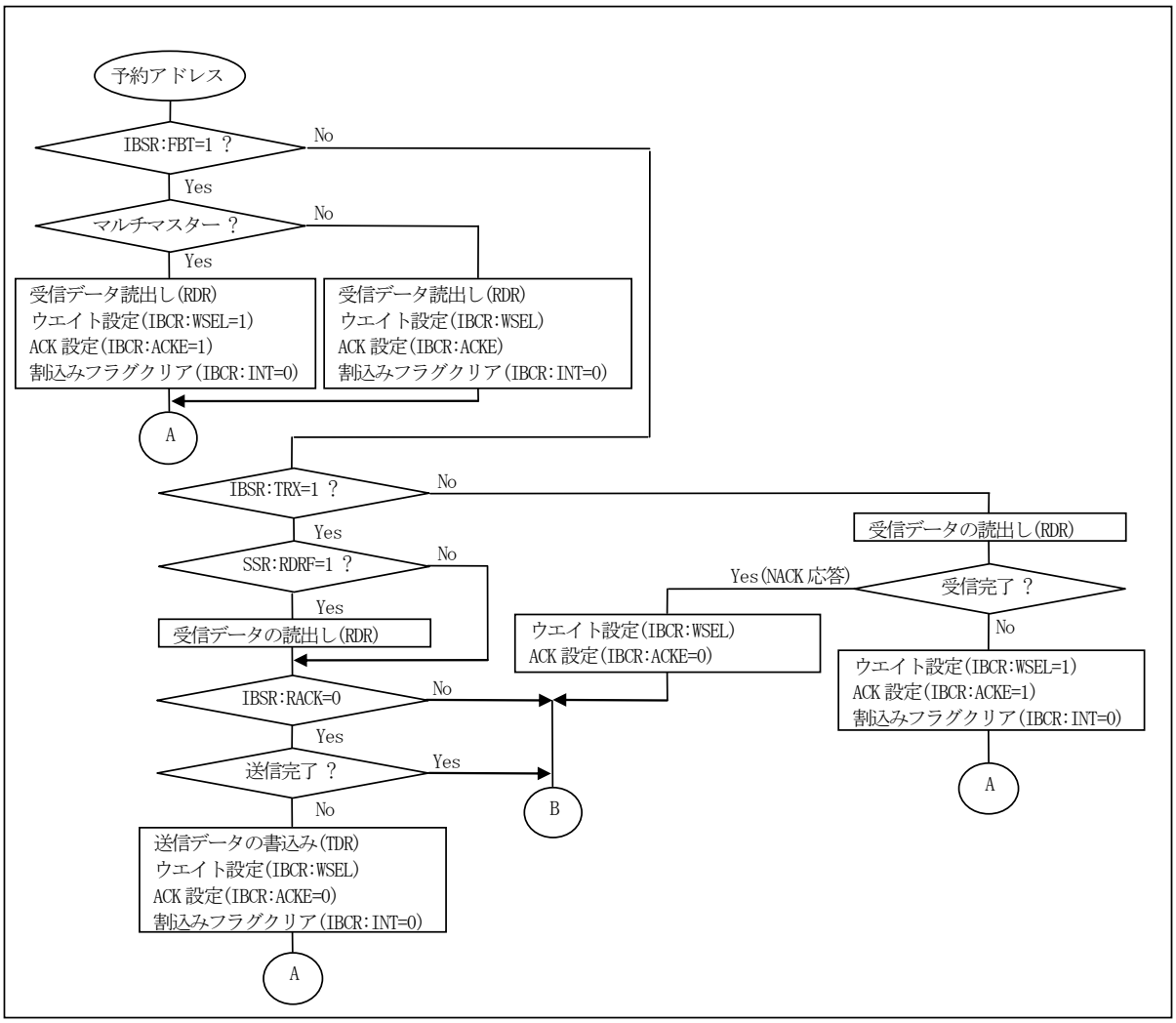


図 40-106 DMA モードが禁止時 (SSR:DMA=0) の I²C フローチャート例 (FIFO 未使用時) 3/3



40.8.6.2 DMA モードが許可時(SSR:DMA=1)の I²C フローチャート例 (FIFO 未使用時)

図 40-107 DMA モードが許可時 (SSR:DMA=1) の I²C フローチャート例 (FIFO 未使用時) 1/4

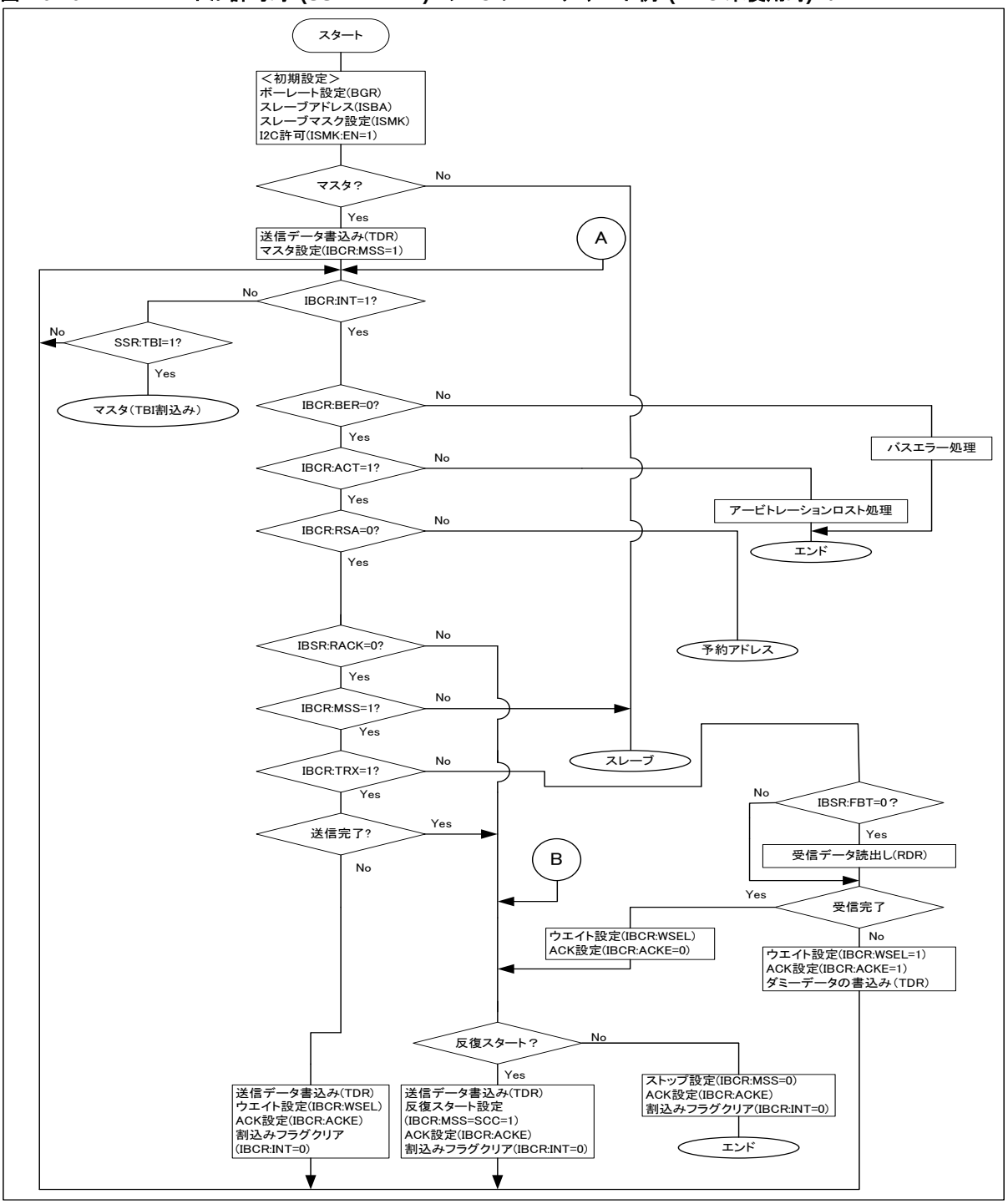


図 40-108 DMA モードが許可時 (SSR:DMA=1) の I²C フローチャート例 (FIFO 未使用時) 2/4

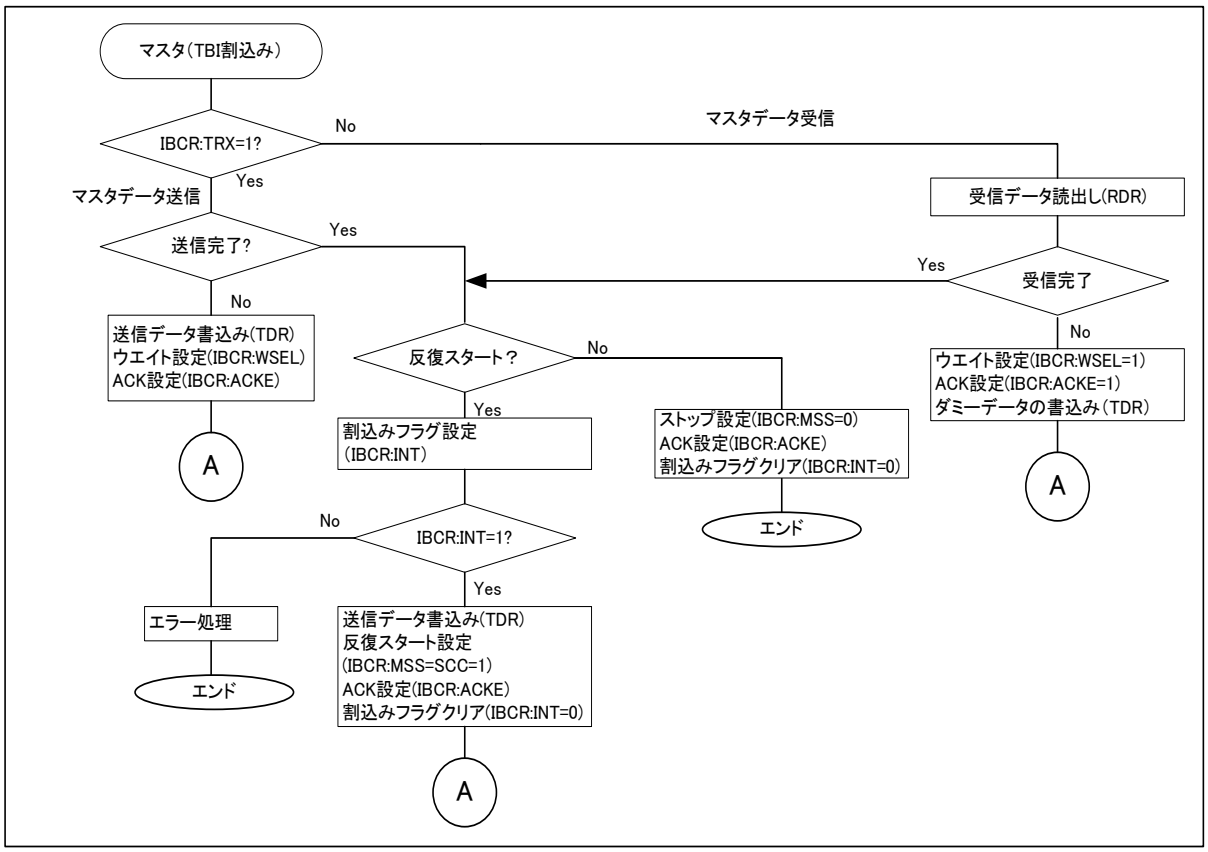


図 40-109 DMA モードが許可時 (SSR:DMA=1) の I²C フローチャート例 (FIFO 未使用時) 3/4

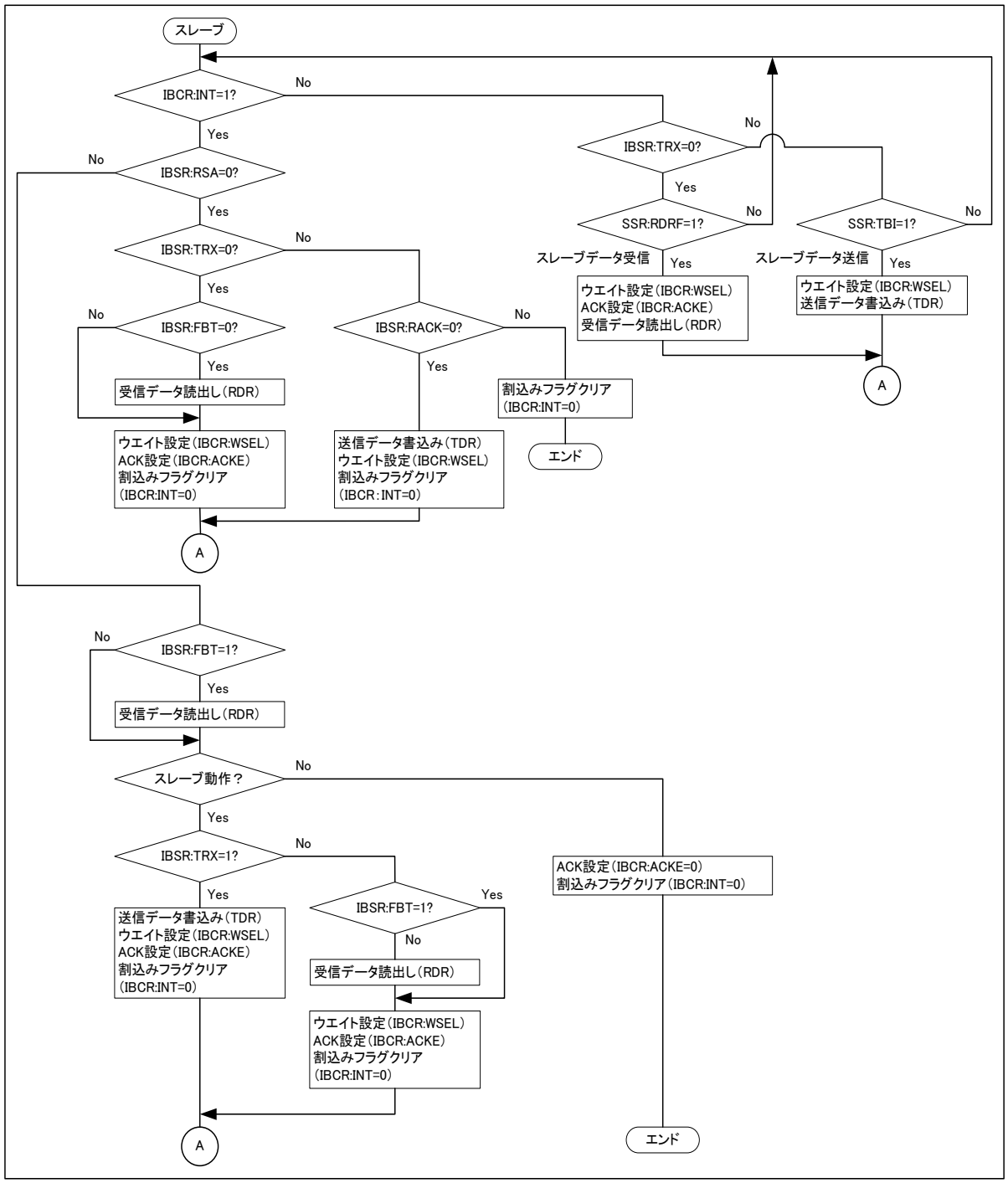
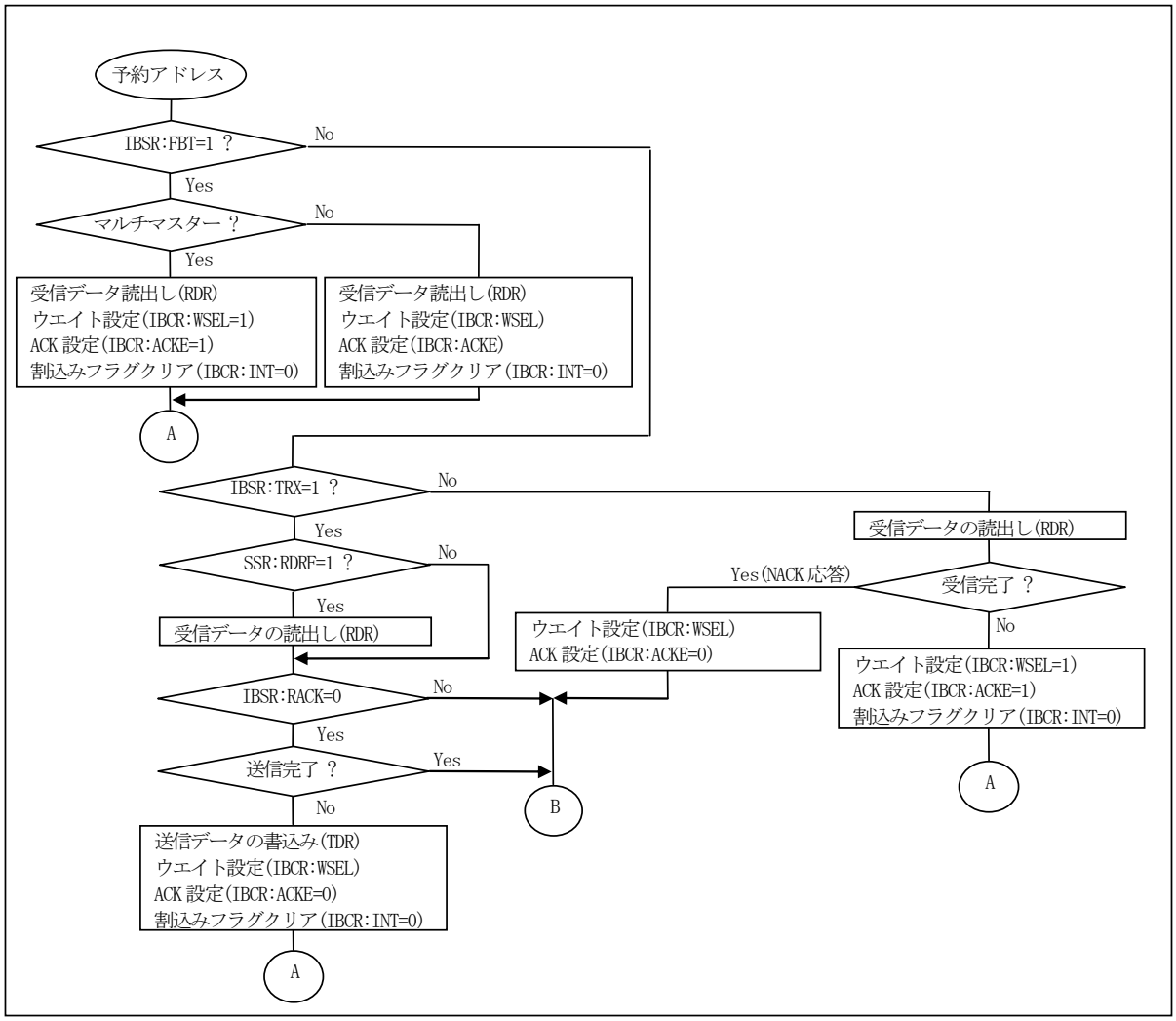


図 40-110 DMA モードが許可時 (SSR:DMA=1) の I²C フローチャート例 (FIFO 未使用時) 4/4



＜注意事項＞

フローは I²C モードによる動作設定概略を示すフローです。アプリケーションにあわせて、エラー処理などを考慮した処理をする必要があります。

41. LIN-UART

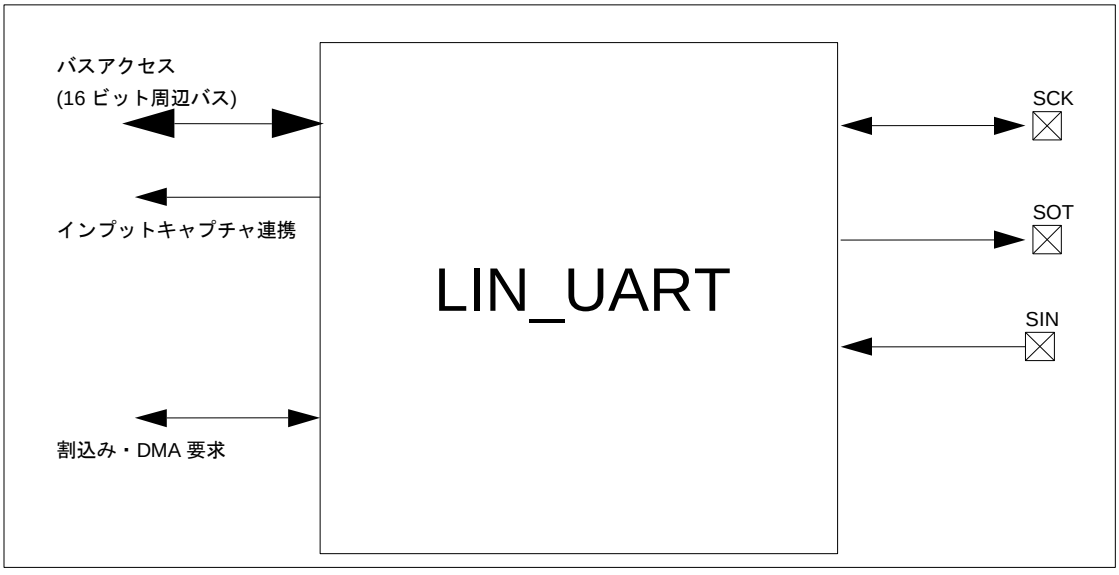
LIN-UART について説明します。

41.1 概要

LIN-UART の概要について説明します。

LIN(Local Interconnect Network)対応 UART(Universal Asynchronous Receiver and Transmitter)は、外部装置との非同期/同期通信を実現する、汎用のシリアルデータ通信インタフェースです。双方向通信機能(ノーマルモード), マスタ/スレーブ型通信機能(マルチプロセッサモード ： マスタ/スレーブ両方をサポート), LIN バスシステム(マスタ/スレーブ両方での動作)をサポートしています。

図 41-1 ブロックダイヤグラム (概要・1 チャネル分)



41.2 特長

LIN-UART の特長について説明します。

LIN-UART は、ほかの CPU や周辺装置、特に LIN デバイスとのデータ送受信に使用される汎用のシリアルデータ通信インタフェースです。

41.2.1 機能

LIN-UART の機能について説明します。

表 41-1 LIN-UART の機能

項目	機能
データバッファ	全二重バッファ
シリアル入力	5 回オーバサンプリングを行い、サンプリング値の多数決により受信値を決定します。(非同期モードのみ)
転送モード	・クロック同期 (スタート／ストップ同期，スタート／ストップビット選択) ・クロック非同期 (スタート／ストップビット使用可能)
ボーレート	・専用ボーレートジェネレータあり(15 ビットリロードカウンタから構成) ・外部クロック入力可能。またリロードカウンタで調節可能。
データ長	・7 ビット (同期モード・LIN モード以外) ・8 ビット
信号方式	NRZ (Non Return to Zero)方式
スタートビットタイミング	非同期モードでは，スタートビットの立下りエッジに同期
受信エラー検出	・フレーミングエラー ・オーバランエラー ・パリティエラー
割込み要求	・受信割込み (受信完了，受信エラー検出, LIN synch break 検出) ・送信割込み (送信データエンプティ) ・インプットキャプチャへの割込み要求(LIN synch field 検出：LSYN)
マスタ/スレーブ通信機能 （マルチプロセッサモード）	1 対複数 (1 マスタ，複数スレーブ)の通信が可能 (マスタとスレーブシステムの両方をサポート)
同期モード	マスタまたはスレーブ機能
端子アクセス	シリアル入出力端子の状態を、直接読書可能。
LIN バスオプション	・マスタデバイス動作 ・スレーブデバイス動作 ・LIN synch break 生成 ・LIN synch break 検出 ・LIN Synch field のスタート/ストップエッジをインプットキャプチャ 0,1,2,3,4,5 で検出 (『インプットキャプチャ』の章の『4.4 LIN SYNCH FIELD 切替えレジスタ : LSYNS』を参照してください。) ・LIN プロトコル Revision2.1 に対応
同期シリアルクロック	スタート/ストップビットで同期通信するために、SCK 端子に連続出力可能
クロック遅延オプション	クロック遅延用の特殊同期クロックモード (SPI に有効)

41.2.2 動作モード

LIN-UART の動作モードについて説明します。

LIN-UART には 4 つのモードがあり、動作モードはシリアルモードレジスタ(SMR)の MD0, MD1 ビットにより決定されます。モード 0 とモード 2 は双方向シリアル通信に、モード 1 はマスタ／スレーブ通信に使用されます。モード 3 は LIN マスタ／スレーブ通信用です。

表 41-2 LIN-UART の動作モード

動作モード		データ長		同期方式	ストップビット長	データビットフォーマット
		パリティなし	パリティあり			
0	ノーマルモード	7 または 8 ビット		非同期	1 ビットまたは 2 ビット	LSB ファーストまたは MSB ファースト
1	マルチプロセッサモード	7 ビットまたは 8 ビット+ 1 ビット (*1)	—			
2	ノーマルモード	8 ビット		同期	なし, 1 ビット, 2 ビット	
3	LIN モード	8 ビット	—	非同期	1 ビットまたは 2 ビット	LSB ファースト

—：設定禁止
*1："+1"はマルチプロセッサモードで通信制御用に使用されるアドレス/データ選択ビット(AD)です。

＜注意事項＞
モード 1 (マルチプロセッサモード)は, マスタ/スレーブ接続時にはマスタとスレーブ両方の動作に対応します。
モード 3 では，通信フォーマットが固定されます。
モードを変更した場合は，LIN-UART は送受信を停止して待機し，次の通信開始を待ちます。

以下の表に、シリアルモードレジスタ(SMR)の MD1, MD0 ビットによって設定される動作モードを示します。

表 41-3 モードビットの設定

MD1	MD0	モード	機能
0	0	0	非同期 (ノーマルモード)
0	1	1	非同期(マルチプロセッサモード)
1	0	2	同期 (ノーマルモード)
1	1	3	非同期 (LIN モード)

41.3 構成

LIN-UART の構成について説明します。

LIN-UART の構成について説明します。

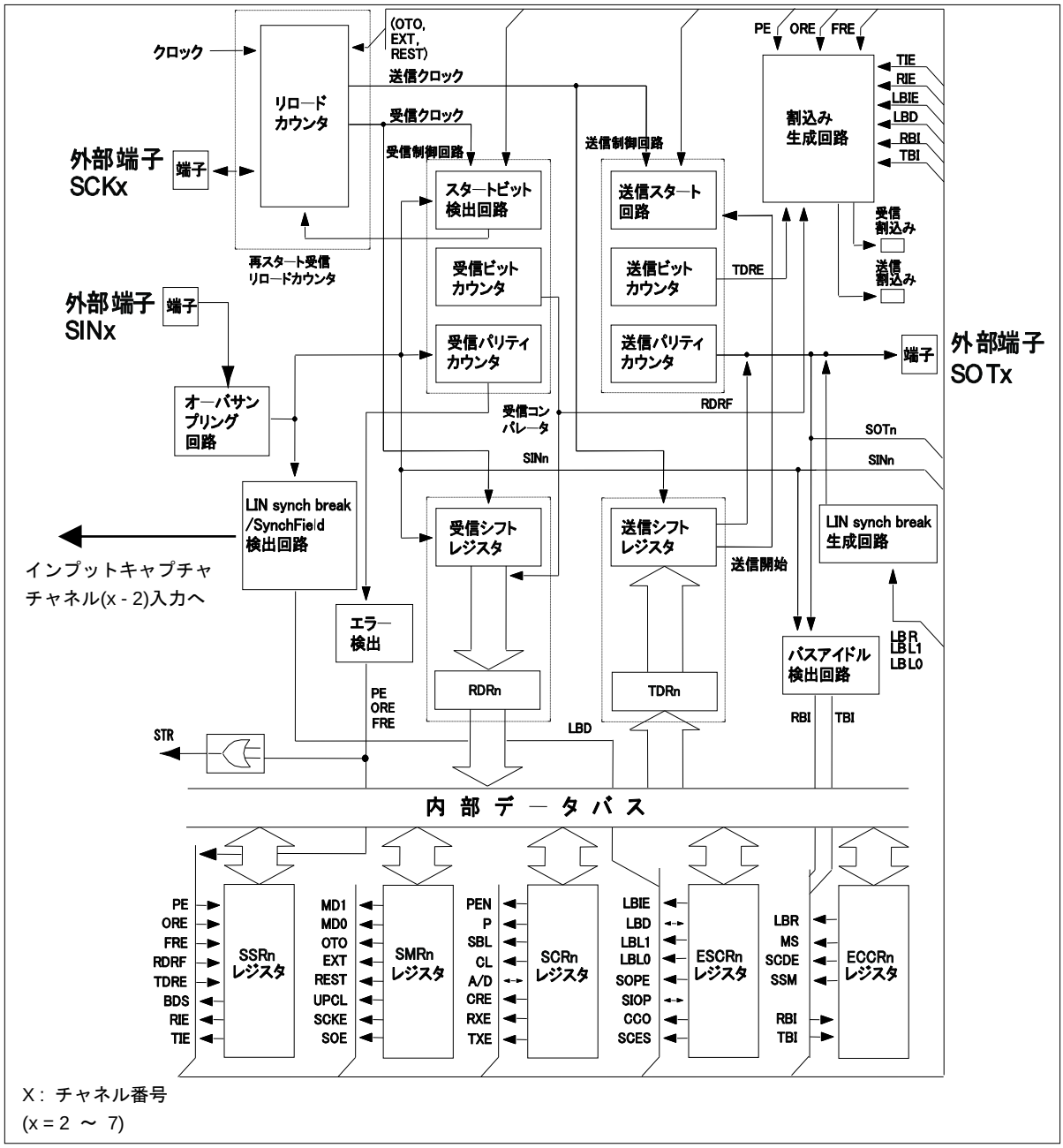
41.3.1 LIN-UART のブロックダイヤグラム

LIN-UART のブロックダイヤグラムについて説明します。

LIN-UART は下記のブロックにより構成されています。

- リロードカウンタ
- 受信制御回路
- 受信シフトレジスタ
- 受信データレジスタ (RDR)
- 送信制御回路
- 送信シフトレジスタ
- 送信データレジスタ (TDR)
- エラー検出回路
- オーバサンプリング回路
- 割込み生成回路
- LIN synch break/LIN synch field 検出回路
- バスアイドル検出回路
- 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)
- シリアルステータスレジスタ (SSR)
- 拡張通信コントロールレジスタ (ECCR)
- 拡張ステータス制御レジスタ (ESCR)

図 41-2 LIN-UART のブロックダイヤグラム



X : チャネル番号
(x = 2 ～ 7)

41.3.2　各ブロックの説明

LIN-UART の各ブロックについて説明します。

41.3.2.1. リロードカウンタ

41.3.2.2. 受信制御回路

41.3.2.3. 受信シフトレジスタ

41.3.2.4. 受信データレジスタ (RDR)

41.3.2.5. 送信制御回路

41.3.2.6. 送信シフトレジスタ

41.3.2.7. 送信データレジスタ (TDR)

41.3.2.8. エラー検出回路

41.3.2.9. オーバサンプリング回路

41.3.2.10. 割込み生成回路

41.3.2.11. LIN synch break/LIN synch field 検出回路

41.3.2.12. LIN synch break 生成回路

41.3.2.13. バスアイドル検出回路

41.3.2.14. シリアルモードレジスタ (SMR)

41.3.2.15. シリアルコントロールレジスタ (SCR)

41.3.2.16. シリアルステータスレジスタ (SSR)

41.3.2.17. 拡張ステータス制御レジスタ (ESCR)

41.3.2.18. 拡張通信コントロールレジスタ (ECCR)

41.3.2.1 リロードカウンタ

LIN-UART のリロードカウンタについて説明します。

リロードカウンタは専用のボーレートジェネレータとして機能します。送受信クロックは、外部クロックまたは内部クロックから生成されます。リロードカウンタにはリロード値として 15 ビットのレジスタがあります。送信リロードカウンタのカウント値は、BGR の値から読み出せます。

41.3.2.2 受信制御回路

LIN-UART の受信制御回路について説明します。

受信制御回路は、受信ビットカウンタ，スタートビット検出回路，受信パリティカウンタによって構成されます。

受信ビットカウンタは受信データをカウントします。指定されたデータ長の 1 データの受信が完了すると、受信データフルフラグビット(SSR:RDRF)をセットします。このとき、受信割込みが許可(SSR:RIE=1)されていれば、受信割込み要求を発生します。

スタートビット検出回路はシリアル入力信号からスタートビットを検出し，スタートビットの立下りエッジと同期してリロードカウンタへ信号を送出します。

受信パリティカウンタは受信データのパリティを計算します。

41.3.2.3 受信シフトレジスタ

LIN-UART の受信シフトレジスタについて説明します。

受信シフトレジスタは、SIN 端子から入力された受信データをビット単位でシフトして取り込みます。受信が完了すると、受信シフトレジスタは受信データを受信データレジスタ(RDR)へ転送します。

41.3.2.4 受信データレジスタ (RDR)

LIN-UART の受信データレジスタ (RDR)について説明します。

受信データレジスタでは受信データが保持されます。シリアル入力データは変換され、受信データレジスタへと格納されます。

41.3.2.5 送信制御回路

LIN-UART の送信制御回路について説明します。

送信制御回路は、送信ビットカウンタ，送信スタート回路，送信パリティカウンタによって構成されます。

送信ビットカウンタは送信データビットをカウントし、設定したデータ長に応じて 1 データを送信します。送信ビットカウンタが書込みデータの送信開始を示すと、シリアルステータスレジスタにフラグをセットします。このとき、送信割みが許可されていれば送信割込み要求を発生します。送信スタート回路は、データが TDR に書き込まれると送信を開始します。

送信パリティカウンタは、パリティありの場合には送信データのパリティビットを生成します。

41.3.2.6 送信シフトレジスタ

LIN-UART の送信シフトレジスタについて説明します。

送信シフトレジスタは、送信データレジスタ(TDR)へ書き込まれた送信データをシフトし、SOT 端子へデータをビット単位で出力します。

41.3.2.7 送信データレジスタ (TDR)

LIN-UART の送信データレジスタ (TDR)について説明します。

送信データレジスタに送信データを設定します。送信データレジスタに書き込まれたデータはシリアルデータへ変換され、出力されます。

41.3.2.8 エラー検出回路

LIN-UART のエラー検出回路について説明します。

受信終了時において、エラーがあったかどうか検出します。エラーが発生すると、対応するエラーフラグをセットします。

41.3.2.9 オーバサンプリング回路

LIN-UART のオーバサンプリング回路について説明します。

非同期モード動作では、マシンクロックで 5 回オーバサンプリングを行い、サンプリング値の多数決により受信値を決定します。また，同期モード動作では動作しません。

41.3.2.10 割込み生成回路

LIN-UART の割込み生成回路について説明します。

すべての割込み要因を制御します。対応する割込み許可ビットがセットされていると、直ちに割込みが発生します。

41.3.2.11 LIN synch break/LIN synch field 検出回路

LIN-UART の LIN synch break/LIN synch field 検出回路について説明します。

LIN マスタノードがメッセージヘッダを送信すると、LIN synch break を検出します。LIN synch break が検出されると、LBD フラグビットがセットされます。LIN Synch Field の 1 回目と 5 回目の立下りエッジを検出し，マスタノードが送信する実際のシリアルクロック同期を測定するために､インプットキャプチャへ内部信号(LSYN)を出力します。

41.3.2.12 LIN synch break 生成回路

LIN-UART の LIN synch break 生成回路について説明します。

拡張ステータス制御レジスタの LIN synch break 長選択ビットにより選択された長さの LIN synch break を生成します。

41.3.2.13 バスアイドル検出回路

LIN-UART のバスアイドル検出回路について説明します。

送受信が行われていないことを検出し、TBI, RBI フラグビットをセットします。

41.3.2.14 シリアルモードレジスタ (SMR)

シリアルモードレジスタについて説明します。

シリアルモードレジスタにより、下記の動作が行われます。

- LIN-UART 動作モードの選択
- クロック入力の選択
- 外部クロックが 1 対 1 接続または、リロードカウンタ接続であるかの選択
- 専用リロードタイマの再起動
- LIN-UART のソフトウェアリセット (レジスタの設定は保存)
- シリアルデータ端子 (SOT)への出力許可/禁止
- シリアルクロック端子 (SCK)への出力許可/禁止

41.3.2.15 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について説明します。

シリアルコントロールレジスタにより、下記の動作が行われます。

- パリティビットの有無
- パリティビットの選択
- ストップビット長の設定
- データ長の設定
- モード 1 でのフレームデータ形式の選択
- エラーフラグのクリア
- 送信許可/禁止
- 受信許可/禁止

41.3.2.16 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について説明します。

以下に動作機能を示します。

- 送受信やエラーの状態確認
- 転送方向 LSB ファースト/MSB ファーストの選択
- 送信割込み許可/禁止
- 受信割込み許可/禁止

41.3.2.17 拡張ステータス制御レジスタ (ESCR)

拡張ステータス制御レジスタについて説明します。

以下に動作機能を示します。

- LIN synch break 割込み許可/ 禁止
- LIN synch break 検出
- LIN synch break 長選択
- SINn, SOTn 端子への直接アクセス
- LIN-UART 同期クロックモードでの連続クロック出力設定
- サンプリングクロックエッジ選択

41.3.2.18 拡張通信コントロールレジスタ (ECCR)

拡張通信コントロールレジスタについて説明します。

以下に動作機能を示します。

- バスアイドル検出
- 同期クロック設定
- LIN synch break 生成

41.4 レジスタ

LIN-UART のレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子一覧

チャネル番号	Base_addr	外部端子名		
		SIN	SOT	SCK
2	0x00D0	SIN2_0 / SIN2_1	SOT2_0 / SOT2_1	SCK2_0 / SCK2_1
3	0x00D8	SIN3_0 / SIN3_1	SOT3_0 / SOT3_1	SCK3_0 / SCK3_1
4	0x00E0	SIN4_0 / SIN4_1	SOT4_0 / SOT4_1	SCK4_0 / SCK4_1
5	0x00E8	SIN5_0 / SIN5_1	SOT5_0 / SOT5_1	SCK5_0 / SCK5_1
6	0x00F0	SIN6_0/ SIN6_1	SOT6_0/SOT6_1	SCK6_0/ SCK6_1
7	0x00F8	SIN7_0/ SIN7_1	SOT7_0/SOT7_1	SCK7_0/ SCK7_1

ch.2～ch.7 は、IO リロケーション機能により、使用する外部端子を選択します。

■ レジスタマップ

表 41-4 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x00D0	SCR2	SMR2	SSR2	RDR2/TDR2	シリアルコントロールレジスタ 2 シリアルモードレジスタ 2 シリアルステータスレジスタ 2 受信/送信データレジスタ 2
0x00D4	ESCR2	ECCR2	BGR2		拡張ステータス制御レジスタ 2 拡張通信コントロールレジスタ 2 ボーレートジェネレータレジスタ 2
0x00D8	SCR3	SMR3	SSR3	RDR3/TDR3	シリアルコントロールレジスタ 3 シリアルモードレジスタ 3 シリアルステータスレジスタ 3 受信/送信データレジスタ 3
0x00DC	ESCR3	ECCR3	BGR3		拡張ステータス制御レジスタ 3 拡張通信コントロールレジスタ 3 ボーレートジェネレータレジスタ 3
0x00E0	SCR4	SMR4	SSR4	RDR4/TDR4	シリアルコントロールレジスタ 4 シリアルモードレジスタ 4 シリアルステータスレジスタ 4 受信/送信データレジスタ 4
0x00E4	ESCR4	ECCR4	BGR4		拡張ステータス制御レジスタ 4 拡張通信コントロールレジスタ 4 ボーレートジェネレータレジスタ 4
0x00E8	SCR5	SMR5	SSR5	RDR5/TDR5	シリアルコントロールレジスタ 5 シリアルモードレジスタ 5 シリアルステータスレジスタ 5 受信/送信データレジスタ 5
0x00EC	ESCR5	ECCR5	BGR5		拡張ステータス制御レジスタ 5 拡張通信コントロールレジスタ 5 ボーレートジェネレータレジスタ 5
0x00F0	SCR6	SMR6	SSR6	RDR6/TDR6	シリアルコントロールレジスタ 6 シリアルモードレジスタ 6 シリアルステータスレジスタ 6 受信/送信データレジスタ 6
0x00F4	ESCR6	ECCR6	BGR6		拡張ステータス制御レジスタ 6 拡張通信コントロールレジスタ 6 ボーレートジェネレータレジスタ 6
0x00F8	SCR7	SMR7	SSR7	RDR7/TDR7	シリアルコントロールレジスタ 7 シリアルモードレジスタ 7 シリアルステータスレジスタ 7 受信/送信データレジスタ 7
0x00FC	ESCR7	ECCR7	BGR7		拡張ステータス制御レジスタ 7 拡張通信コントロールレジスタ 7 ボーレートジェネレータレジスタ 7

41.4.1 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のビット構成について示します。

シリアルコントロールレジスタ(SCR)では、パリティビットの設定，ストップビット長とデータ長の選択，モード 1 におけるフレームデータフォーマットの選択，受信エラーフラグのクリア，送受信の許可/禁止を設定します。

■ SCR : アドレス Base_addr + 00H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PEN	P	SBL	CL	AD	CRE	RXE	TXE
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R,W	R0,W	R/W	R/W

[bit7] PEN : パリティ許可ビット

PEN	パリティ許可
0	パリティなし［初期値］
1	パリティあり

パリティビットの付加(送信時)と検出(受信時)を行うかどうかを設定します。

モード 0 またはモード 2 でスタート/ストップあり(ECCR:SSM="1")に設定した場合にのみ付加されます。モード 1, 3 では"0"に固定されます。

[bit6] P : パリティ選択ビット

P	パリティ選択
0	偶数パリティ［初期値］
1	奇数パリティ

パリティあり(PEN=1)の場合、偶数パリティ"0"、奇数パリティ"1"を選択します。

[bit5] SBL : ストップビット長選択ビット

SBL	ストップビット長
0	1 ビット［初期値］
1	2 ビット

動作モード 0，1，3(非同期)の場合、または動作モード 2(同期)でスタート/ストップあり(ECCR:SSM=1)に設定した場合のストップビット(送信データのフレームエンドマーク)のビット長を選択します。

＜注意事項＞

受信時は、ストップビットの 1 ビットでのみフレーミングエラーを検出します。

[bit4] CL : データ長選択ビット

CL	データ長選択
0	7 ビット［初期値］
1	8 ビット

送信／受信データ長を指定します。モード 2 およびモード 3 では"1"に固定されます。

[bit3] AD : アドレス／データ形式選択ビット

AD	アドレス／データ形式選択
0	データフレーム［初期値］
1	アドレスフレーム

マルチプロセッサモード(モード 1)でのデータ形式を指定します。読出しは、最後に受信したデータ形式の値になります。

＜注意事項＞

マルチプロセッサモード(モード 1)以外のモードでの AD ビット読出し値は不定です。AD ビットの使用については、『41.8. 使用上の注意事項』を参照してください。

[bit2] CRE: 受信エラーフラグクリアビット

CRE	受信エラークリア	
	書込み	読出し
0	影響なし［初期値］	読出し値は常に"0"
1	すべての受信エラー(PE, FRE, ORE)をクリア	

シリアルステータスレジスタ(SSR)の PE, FRE, ORE フラグがクリアされます。

＜注意事項＞

受信動作禁止(RXE=0)後に受信エラーフラグをクリアしてください。

[bit1] RXE : 受信許可ビット

RXE	受信許可
0	受信禁止［初期値］
1	受信許可

LIN-UART の受信動作を許可または禁止します。"0"に設定されている場合は、データフレーム受信は禁止されます。モード 3 の LIN synch break 検出は影響されません。

＜注意事項＞

- 受信中に受信動作を禁止 (RXE=0)した場合には、直ちに受信動作が停止します。この場合データは保証されません。
- 動作モード 2 で ECCR:MS=0 で、受信中に送信動作を禁止(TXE=0)にする場合は、受信動作(RXE=0)も禁止してください。

[bit0] TXE : 送信許可ビット

TXE	送信許可
0	送信禁止［初期値］
1	送信許可

LIN-UART の送信動作を許可/禁止します。

＜注意事項＞

送信中に送信動作を禁止 (TXE=0)した場合には、直ちに送信動作が停止します。この場合データは保証されません。

41.4.2 シリアルモードレジスタ (SMR)

シリアルモードレジスタ (SMR)のビット構成を示します。

シリアルモードレジスタ(SMR)は、動作モードとボーレートクロックを選択します。また、シリアルデータとクロックの端子への出力許可/禁止を設定するレジスタです。

■ SMR：アドレス Base_addr + 01_H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MD1	MD0	OTO	EXT	REST	UPCL	SCKE	SOE
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R0,W	R0,W	R/W	R/W

[bit7, bit6] MD1, MD0： 動作モード選択ビット

MD1	MD0	動作モード設定
0	0	モード 0：非同期ノーマルモード［初期値］
0	1	モード 1：非同期マルチプロセッサモード
1	0	モード 2：同期モード
1	1	モード 3：非同期 LIN モード

LIN-UART の動作モードを設定します。

＜注意事項＞

通信モードの変更は、LIN-UART の動作停止中に行ってください。送受信中にモード設定の変更をした場合、そのとき送受信されたデータは保証されません。送信データレジスタ(TDR)に書込み後にモード設定の変更を行った場合は、TDR の書込みは無効となり、送信データエンプティフラグがセット(SSR:TDRE=1)されます。

[Bit5] OTO: 1 to 1 外部クロック許可ビット

OTO	外部クロック許可
0	ボーレートジェネレータ(リロードカウンタ)使用［初期値］
1	外部クロックを直接使用

"1"に設定すると LIN-UART のシリアルクロックに外部クロックが直接使用することを許可します。モード 2(同期)のスレーブ動作時(ECCR:MS=1)に使用されます。

EXT=0 の場合、OTO ビットは"0"に固定されます。

[bit4] EXT： 外部クロック選択ビット

EXT	外部シリアルクロック許可
0	ボーレートジェネレータ(リロードカウンタ)を使用［初期値］
1	外部クロックシリアルクロックソース使用

リロードカウンタ用のクロックを選択できます。

[bit3] REST : リロードカウンタ再スタートビット

REST	リロードカウンタ再スタート	
	書込み	読出し
0	影響なし［初期値］	読出し値は常に"0"
1	カウンタ再スタート	

[bit2] UPCL : LIN-UART プログラマブルクリアビット(ソフトウェアリセット)

UPCL	LIN-UART プログラマブルクリア(ソフトウェアリセット)	
	書込み	読出し
0	影響なし［初期値］	読出し値は常に"0"
1	LIN-UART リセット	

"1"書込みで、LIN-UART は直ちにリセットされます。ただし、レジスタの設定値は保持されます。また、送信／受信は中断されます。

すべての送受信割込み要因 (TDRE, RDRF, LBD, PE, ORE, FRE) と LIN Break field 生成 bitto(LBR)は初期値になります。割込み禁止および送信禁止を設定した後、リセットを行ってください。

＜注意事項＞

- シリアル制御レジスタ(SCR)の TXE ビットが"0"のときに LIN-UART ソフトウェアリセット(UPCL=1)を実行してください。
- 動作モード 2 以外でプログラマブルリセット直後からシリアルデータ入力が"L"の状態が続いた場合、シリアルデータ入力が"H"になるまで、スタートビットおよび LIN Break field は検出されません。

[bit1] SCKE : シリアルクロック出力許可

SCKE	シリアルクロック出力許可
0	クロック入力端子［初期値］
1	シリアルクロック出力端子

シリアルクロック端子(SCK)の入出力を制御します。

"1"に設定した場合、モード 2 でクロックを出力します。

＜注意事項＞

- SCK 端子をシリアルクロック入力として使用する場合(SCKE = "0")、同時に外部クロック選択ビットを外部クロック側に設定してください(EXT = "1")
- SCK 端子はペリフェラル入出力端子として設定してください。設定方法は『I/O ポート』の章を参照してください。

[bit0] SOE : シリアルデータ出力許可ビット

SOE	シリアルデータ出力許可
0	出力無効［初期値］
1	シリアルデータ出力

LIN-UART からデータを送信する場合は、本ビットを"1"に設定してください。本ビットの初期値は"0"ですが、本ビットを"0"に設定する必要があるケースはありません。

41.4.3 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)のビット構成を示します。

シリアルステータスレジスタ(SSR)により、送信／受信状態，エラーの有無を確認できます。また、送信／受信割込みの制御も行います。

■ SSR：アドレス Base_addr + 02_H (アクセス: バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE
初期値	0	0	0	0	1	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R/W	R/W	R/W

[bit7] PE： パリティエラーフラグビット

PE	パリティエラー
0	パリティエラーなし［初期値］
1	受信中にパリティエラー発生

PEN=1 で受信時にパリティエラーが発生すると"1"にセットされ、シリアルコントロールレジスタ(SCR)の CRE ビットに"1"を書き込むと、"0"にクリアされます。

PE ビットと RIE ビットが"1"の場合は，受信割込み要求が出力されます。

フラグがセットされた場合は、受信データレジスタ(RDR)のデータは無効です。

[bit6] ORE： オーバランエラーフラグビット

ORE	オーバランエラー
0	オーバランエラーなし［初期値］
1	受信中にオーバランエラー発生

受信中にオーバランエラーが発生すると、"1"にセットされます。シリアルコントロールレジスタ(SCR)の CRE ビットへ"1"を書き込むと、"0"にクリアされます。

ORE ビットと RIE ビットが"1"の場合は、受信割込み要求が出力されます。

フラグがセットされた場合は、受信データレジスタ(RDR)のデータは無効です。

[bit5] FRE： フレーミングエラーフラグビット

FRE	フレーミングエラー
0	フレーミングエラーなし［初期値］
1	受信中にフレーミングエラー発生

受信中にフレーミングエラーが発生すると、"1"にセットされます。シリアルコントロールレジスタ(SCR)の CRE ビットへ"1"を書き込むと、"0"にクリアされます。

FRE ビットと RIE ビットが"1"の場合は、受信割込み要求が出力されます。

フラグがセットされた場合は，受信データレジスタ(RDR)のデータは無効です。

＜注意事項＞

- SCR: SBL=1 に設定しても、ストップビットの 1 ビット目のみフレーミングエラーを検出します。
- 動作モード2以外でフレーミングエラーを検出し､そのままシリアルデータ入力が"L"の状態が続いた場合、シリアルデータ入力が"H"になるまで、スタートビットは検出されません。

[bit4] RDRF : 受信データフルフラグビット

RDRF	受信データレジスタフル
0	受信データレジスタにデータなし［初期値］
1	受信データレジスタにデータあり

受信データレジスタ(RDR)の状態を示します。

受信データが RDR へ格納されると"1"にセットされます。RDR が読み出すと"0"にクリアされます。

RDRF ビットと RIE ビットが"1"の場合は、受信割込み要求が出力されます。

[bit3] TDRE : 送信データエンプティフラグビット

TDRE	送信データレジスタエンプティ
0	送信データレジスタにデータあり
1	送信データレジスタにデータなし［初期値］

送信データレジスタ(TDR)の状態を示します。

送信データを TDR へ書き込むと"0"となり、送信データがあることを示します。データが送信シフトレジスタへ格納され送信が開始されると、"1"にセットされます。

TDRE ビットと TIE ビットが"1"の場合は、送信割込み要求が出力されます。

TDRE ビットが"1"のとき、拡張通信制御レジスタ(ECCR)の LBR ビットに"1"を設定すると TDRE ビットは"0"になり、LIN synch break 生成後 TDR レジスタに有効なデータがない場合"1"になります。

＜注意事項＞

初期状態では、TDRE=1 になっています。
送信データレジスタ(TDR)への書込みにより TDRE=0 とした後に、モード設定(SMR:MD[1:0])の変更を行うと、送信データは無効となり TDRE=1 となります。

[bit2] BDS : 転送方向選択ビット

BDS	ビット方向設定
0	LSB ファースト(最下位ビットから転送)［初期値］
1	MSB ファースト(最上位ビットから転送)

シリアル転送データの転送を LSB ファースト(BDS = "0")または MSB ファースト(BDS = "1")のいずれかに選択できます。

モード 3 では"0"に固定されます。

＜注意事項＞

受信データレジスタの読出し／書込み中に，受信データの上位側と下位側は入れ替えられます。データが RDR に書き込まれた後、BDS ビットの値を書き換えると、データは無効となります。

[bit1] RIE : 受信割込み要求許可ビット

RIE	受信割込み許可
0	受信割込み禁止［初期値］
1	受信割込み許可

CPU への受信割込み要求出力の許可/禁止を制御します。

RIE ビットと受信データフラグビット(RDRF)が"1"もしくはエラーフラグ(PE, ORE, FRE)が"1"にセットされると、受信割込み要求が出力されます。

[bit0] TIE : 送信割込み要求許可ビット

TIE	送信割込み許可
0	送信割込み禁止［初期値］
1	送信割込み許可

CPU への送信割込み要求出力の許可/禁止を制御します。

TIE ビットと TDRE ビットが"1"の場合、送信割込み要求が出力されます。

41.4.4　受信／送信データレジスタ (RDR/TDR)

受信／送信データレジスタ (RDR/TDR)について示します。

受信データレジスタ(RDR)は受信データを、送信データレジスタ(TDR)は送信データを保持します。RDR, TDR は同一のアドレスに配置されています。

41.4.4.1　受信データレジスタ (RDR)

受信データレジスタ(RDR)のビット構成を示します。

■ 受信データレジスタ (RDR) :
RDR ：アドレス Base_addr + 03H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit7～bit0] D[7:0] : データレジスタ

アクセス	データレジスタ
読出し	受信データレジスタからの読出し

受信データレジスタ(RDR) は、シリアルデータ受信用のデータバッファレジスタです。

シリアル入力端子 (SIN) に送られてきたシリアルデータ信号がシフトレジスタで変換されて，受信データレジスタ(RDR) に格納されます。

データ長が 7 ビットの場合は、上位 1 ビット(RDR:D7) は"0" となります。

受信データが, 受信データレジスタ(RDR) に格納されると、受信データフルフラグビット(SSR:RDRF) が"1" にセットされます。受信割込みが許可されている場合は(SSR:RIE=1)、受信割込み要求を発生します。

受信データレジスタ(RDR) は、受信データフルフラグビット(SSR:RDRF) が"1" の状態で読出してください。受信データフルフラグビット(SSR:RDRF) は、受信データレジスタ(RDR) を読み出すと自動的に"0" にクリアされます。また、受信割込みが許可されていて、エラーが生じていない場合には、受信割込みもクリアされます。

受信エラーが発生(SSR:PE, ORE, FRE のいずれかが"1") した場合、受信データレジスタ(RDR) のデータは無効となります。

41.4.4.2 送信データレジスタ (TDR)

送信データレジスタ(TDR)のビット構成について示します。

■ 送信データレジスタ (TDR) :
TDR：アドレス Base_addr + 03H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	－	－	－	－	－	－	－	－
属性	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W	RX,W

[bit7～bit0] D[7:0] : データレジスタ

アクセス	データレジスタ
書込み	送信データレジスタへの書込み

送信データが送信許可状態で送信データレジスタへ書き込まれると、データは送信シフトレジスタへ転送後シリアルデータへと変換され、シリアルデータ出力端子(SOT)から送信されます。データ長が 7 ビットの場合は、最上位ビット(D7)は送信されません。

TDR レジスタへ送信データが書き込まれると、送信データエンプティフラグビット(SSR の TDRE ビット)が"0"にクリアされます。送信シフトレジスタへの転送が完了し,送信が開始されると，TDRE ビットは'1'にセットされます。TDRE ビット"1"であれば、次の送信データを TDR レジスタへ書き込むことができます。送信割込み要求が許可されている場合は、送信割込みが発生します。送信割込みの発生、もしくは TDRE ビットが"1"であれば、次のデータを書き込んでください。

＜注意事項＞

TDR は書込み専用、RDR は読出し専用のレジスタです。このレジスタは同一のアドレスに配置されているため、読出し値と書込み値は異なります。そのため、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。

41.4.5 拡張ステータス制御レジスタ (ESCR)

拡張ステータス制御レジスタのビット構成について示します。

拡張ステータス制御レジスタでは LIN 機能を設定できます。また、SIN, SOT 端子への直接アクセスと LIN-UART 同期クロックモードの設定ができます。

■ ESCR：アドレス Base_addr + 04H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	LBIE	LBD	LBL1	LBL0	SOPE	SIOP	CCO	SCES
初期値	0	0	0	0	0	X	0	0
属性	R/W	R(RM1),W	R/W	R/W	R/W	R,W	R/W	R/W

[bit7] LBIE : LIN synch break 検出割込み許可ビット

LBIE	LIN synch break 検出割込み許可
0	LIN synch break 割込み禁止［初期値］
1	LIN synch break 割込み許可

LIN synch break 検出フラグ(LBD)の読出しが"1"のとき、割込みが許可(LBIE=1)されると割込みを発生します。

動作モード 1,2 では"0"に固定されます。

[bit6] LBD : LIN synch break 検出フラグビット

LBD	LIN synch break 検出	
	書込み	読出し
0	LIN synch break 検出フラグのクリア	LIN synch break の検出なし［初期値］
1	影響なし	LIN synch break の検出あり

LIN synch break が検出(シリアル入力が 11 ビットタイム以上"0"になる)されると、"1"にセットされます。"0"書込みにより LBD フラグビットはクリアされ、LIN synch break 検出割込みが許可されている場合は、割込みもクリアします。

リードモディファイライト系命令では常に"1"が返されますが、この場合は LIN synch break 検出を意味しません。

＜注意事項＞

LIN synch break 検出を行う際には、LIN synch break 検出割込みを許可(LBIE=1)に設定した後，受信禁止(SCR:RXE=0)に設定してください。

[bit5, bit4] LBL1, LBL0 : LIN synch break 長選択ビット

LBL1	LBL0	LIN synch break 長
0	0	13 ビット長［初期値］
0	1	14 ビット長
1	0	15 ビット長
1	1	16 ビット長

LIN-UART で生成される LIN synch break のシリアルビット長が定義されます。LIN synch break の受信では、常に 11 ビット固定です。

[bit3] SOPE : シリアル出力端子直接アクセス許可ビット

SOPE	シリアル出力端子直接アクセス
0	シリアル出力端子直接アクセス禁止［初期値］
1	シリアル出力端子直接アクセス許可

シリアルデータの出力が許可(SMR:SOE=1)されているときに本ビットに"1"を設定すると、SOT 端子への直接アクセスを許可します。

詳細は、「表 41-5 SOPE と SIOP の機能」を参照してください。

＜注意事項＞

SCR:TXE=1 で、送信中または動作モード 2 のスレーブモードのときは本ビットの設定に関らず、SOT 端子への直接アクセスはできません。

[bit2] SIOP : シリアル入出力端子直接アクセス許可ビット

SIOP	シリアル入出力端子アクセス	
	書込み（SOPE が"1"の場合）	読出し
0	SOT 端子は"0"出力	SIN 端子の値を読出し
1	SOT 端子は"1"出力［初期値］	

通常読出し命令では、SIN 端子の値を返します。書込みは SOT 端子の値を設定します。リードモディファイライト系命令では、SOT 端子の値を返します。

詳細は、下表を参照してください。

表 41-5 SOPE と SIOP の機能

SOPE	SIOP	SIOP への書込み	SIOP からの読出し
0	R/W	影響なし(書込み値は保持)	SIN の値を返す
1	R/W	SOT 端子へ "0"または"1"書込み	SIN の値を返す
0	RMW	影響なし(書込み値は保持)	SOT の値を返す
1	RMW	SOT 端子へ "0"または"1"書込み	SOT の値を返す

[bit1] CCO : 連続クロック出力許可ビット

CCO	連続クロック出力（モード 2）
0	連続クロック出力禁止［初期値］
1	連続クロック出力許可

LIN-UART がモード 2(同期モード)のマスタ設定(ECCR:MS=0)で、SCK 端子がクロック出力として設定されている場合に、連続シリアルクロック出力を許可します。

＜注意事項＞

SCK端子をクロック出力設定にしてください(SMR:SCKE=1)。CCO ビットに"1"を設定する場合，スタート/ストップビット付加設定(ECCR:SSM=1) にします。

動作モード 0、1、3 と動作モード 2 のスレーブ設定のときは、本ビットに"0"を設定してください。

シリアルクロック出力許可中（SMR:SCKE="1"）で以下の条件で CCO と SCES ビットを設定した場合、シリアルクロック出力切り替え直後に所定のクロック幅がシリアルクロック出力端子(SCK 端子)に出力されないことがありますが、その後、正常に出力されます。

- CCO ビットが"1"の状態で SCES ビットを変更した場合
- CCO ビットと SCES ビットを同時に変更した場合
- CCO ビットを"1"から"0"に変更した場合

[bit0] SCES : サンプリングシリアルクロックエッジ選択ビット

SCES	サンプリングシリアルクロックエッジ選択
0	クロックの立上りエッジでサンプリング（通常）［初期値］
1	クロックの立下りエッジでサンプリング（反転クロック）

LIN-UART がモード 2(同期モード)のスレーブ設定(ECCR:MS=1)のとき、SCES を"1"に設定すると、サンプリングエッジが立上りエッジから立下りエッジへと切り替わります。

モード 2(同期モード)のマスタ設定(ECCR:MS=0)で、SCK 端子がクロック出力として設定されている場合、内部シリアルクロックと出力クロック信号が反転します。

モード 0, 1, 3 のときは"0"に設定してください。

＜注意事項＞

本ビットに”1”を設定している場合、ソフトウェアリセットを禁止します。また、送受信を禁止時に本ビットを変更してください。

41.4.6 拡張通信コントロールレジスタ (ECCR)

拡張通信コントロールレジスタ(ECCR)のビット構成について示します。

拡張通信コントロールレジスタ(ECCR)では、バスアイドル検出設定、同期クロック設定、LIN synch break 生成ができます。

■ ECCR：アドレス Base_addr + 05H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	LBR	MS	SCDE	SSM	予約	RBI	TBI
初期値	0	0	0	0	0	0	X	X
属性	R/W0	R0,W	R/W	R/W	R/W	R/W0	R,WX	R,WX

[bit7] 予約
必ず"0"を書き込んでください。

[bit6] LBR : LIN synch break 生成ビット

LBR	LIN synch break 生成	
	書込み	読出し
0	影響なし［初期値］	読出し値は常に"0"
1	LIN synch break 生成	

モード 3 の場合に、LBR ビットに"1"が書き込まれると、ESCR の LBL1, LBL0 で設定した長さの LIN synch break が生成されます。動作モード 0 のときは、"0"に設定してください。

[bit5] MS : マスタ／スレーブモード選択ビット

MS	マスタ／スレーブ機能(モード 2)
0	マスタモード（シリアルクロック生成）［初期値］
1	スレーブモード（外部シリアルクロック受信）

モード2(同期)でマスタまたはスレーブモードを選択します。マスタモード選択時は同期クロックを生成します。スレーブモード選択時は、外部シリアルクロックを受信します。

動作モード 0, 1, 3 のときは"0"に固定されます。

MS ビットを変更する際は、送信禁止(SCR:TXE=0)のときに変更してください。

＜注意事項＞

スレーブモード選択時は、クロックソースを外部クロックとして 1 対 1 外部クロック入力に設定してください（SMR の SCKE="0", EXT="1", OTO="1"）。

[bit4] SCDE : シリアルクロック遅延許可ビット

SCDE	シリアルクロック遅延許可(モード 2)
0	クロック遅延禁止［初期値］
1	クロック遅延許可

モード 2 のマスタモード動作時に, SCDE ビットを"1"に設定すると図 41-15 送信クロックの遅延シリアルクロック出力に示すような遅延したシリアルクロックを出力します。

動作モード 0, 1, 3 のときは"0"に固定されます。

[bit3] SSM : スタート／ストップビット許可

SSM	スタート/ストップビット許可(モード 2)
0	スタート／ストップビットなし［初期値］
1	スタート／ストップビットあり

モード 2 でスタートビットとストップビットを同期データフォーマットに付加します。

動作モード 0, 1, 3 のときは"0"に固定されます。

[bit2] 予約

必ず"0"を書き込んでください。

[bit1] RBI : 受信バスアイドルフラグビット

RBI	受信バスアイドル
0	受信動作中
1	受信動作なし

SIN 端子が"H"レベルで，受信動作がない場合に，"1"にセットされます。

モード 2 では受信バスアイドル検出機能は使用できません。

[bit0] TBI : 送信バスアイドルフラグビット

TBI	送信バスアイドル
0	送信動作中
1	送信動作なし

SOT 端子に送信動作がない場合は，"1"にセットされます。

モード 2 でスレーブモード選択した場合、送信バスアイドル検出機能は使用できません。

41.4.7 ボーレートジェネレータレジスタ (BGR)

ボーレートジェネレータレジスタ (BGR)のビット構成について示します。

ボーレートジェネレータレジスタ(BGR)は, シリアルクロックの分周比を設定します。また, 送信リロードカウンタの正確な値を読み出すことも可能です。

■ BGR：アドレス Base_addr + 06H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	—	B14	B13	B12	B11	B10	B09	B08
初期値	—	0	0	0	0	0	0	0
属性	R0,WX	R,W	R,W	R,W	R,W	R,W	R,W	R,W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	B07	B06	B05	B04	B03	B02	B01	B00
初期値	0	0	0	0	0	0	0	0
属性	R,W	R,W	R,W	R,W	R,W	R,W	R,W	R,W

[bit15] - : 未定義ビット
常に"0"が読み出せます。書込みは動作に影響ありません。

[bit14～bit8] B[14:08] : ボーレートジェネレータレジスタ 1

B14-08	ボーレートジェネレータレジスタ 1
書込み	リロード値の bit14～bit8 をカウンタへ書込み
読出し	カウンタのビット値 14～8 の読出し

[bit7～bit0] B[07:00] : ボーレートジェネレータレジスタ 0

B07-00	ボーレートジェネレータレジスタ 0
書込み	リロード値の bit7～bit0 をカウンタへ書込み
読出し	カウンタのビット値 7～0 の読出し

ボーレートジェネレータレジスタ(BGR)では, シリアルクロックの分周比を設定します。

カウントするリロード値の書込みと送信リロードカウンタ値の読出しが可能です。

ボーレートジェネレータレジスタ(BGR)にリロード値を書き込むとリロードカウンタはカウントを開始します。

41.5 割込み

割込みについて示します。

41.5.1. 概要

41.5.2. 受信割込み生成とフラグセットタイミング

41.5.3. 送信割込み生成とフラグタイミング

41.5.1 概要

概要について示します。

LIN-UART には受信割込みと送信割込みがあります。下記のいずれかの場合において割込み要求が生成されます。

- 受信データの受信データレジスタ(RDR)への格納、もしくは受信エラーの発生
- 送信データの送信データレジスタ(TDR)から送信シフトレジスタへの転送
- LIN synch break 検出

41.5.1.1 LIN-UART の割込み

LIN-UART の割込みについて示します。

下表に割込みコントロールビットと割込み要因を示します。

表 41-6 LIN-UART の割込みコントロールビットと割込み要因

受信/送信/インプットキャプチャ	割込み要求フラグビット	フラグレジスタ	動作モード				割込み要因	割込み要因許可ビット	割込み要求のクリア方法
			0	1	2	3			
受信	RDRF	SSR	○	○	○	○	RDR への受信データ書込み	SSR:RIE	受信データの読出し
	ORE	SSR	○	○	○	○	オーバランエラー		受信エラークリアビット(SSR:CRE) への"1"書込み
	FRE	SSR	○	○	△	○	フレーミングエラー		
	PE	SSR	○	×	△	×	パリティエラー		
	LBD	ESCR	×	×	×	○	LIN-Synch-Break 検出	ESCR:LBIE	ESCR の LBD ビットへの"0"書込み
送信	TDRE	SSR	○	○	○	○	送信レジスタエンプティ	SSR:TIE	送信データの書込み、LIN synch break 生成ビット(ECCR:LBR) への"1"書込み
インプットキャプチャ(ch.0～ch.5)	ICP	ICS	×	×	×	○	LIN synch field の最初の立下りエッジ	ICS:ICE	ICS:ICP ビットへの"0"書込み
	ICP	ICS	×	×	×	○	LIN synch field の 5 回目の立下りエッジ	ICS:ICE	

○：使用可能
△：ECCR の SSM ビットが"1"のときに使用可能
×：使用不可

41.5.1.2 受信割込み

受信割込みについて示します。

受信モード時に下記の一つが発生すると、シリアルステータスレジスタ(SSR)の対応するフラグビットが"1"にセットされます。

- **データ受信完了：RDRF**

受信データの受信シフトレジスタから受信データレジスタ(RDR)への転送、読出し

- **オーバランエラー：ORE**

RDRF="1"で、RDR が CPU によって読み出されず、次の受信データが受信シフトレジスタから受信データレジスタ(RDR)へ転送された場合 (ORE=1)

- **フレーミングエラー：FRE**

ストップビット 1 ビット目でシリアルデータが"L"を検出された場合。(FRE=1)

- **パリティエラー：PE**

パリティ検出エラー (PE=1)

受信割込みが許可(SSR:RIE = "1")されているときに、上記のフラグが 1 つでも"1"となると、受信割込み要求が発生します。

受信データレジスタ(RDR)が読み出されると、RDRF フラグは自動的に"0"にクリアされます。シリアルコントロールレジスタ(SCR)の受信エラーフラグクリアビット(CRE)に"1"が書き込まれると、エラーフラグはすべて"0"にクリアされます。

＜注意事項＞

CRE ビットは書込み専用で、"1"を書き込むと 1 クロックサイクルの期間"1"を保持します。

41.5.1.3 送信割込み

送信割込みについて示します。

送信データが送信データレジスタ(TDR)から送信シフトレジスタへ転送され、送信が開始されるとシリアルステータスレジスタ(SSR)の送信データレジスタエンプティフラグビット(TDRE)が"1"にセットされます。この場合、SSR の送信割込み許可ビット(TIE)が設定されていると、割込み要求が発生します。

＜注意事項＞

リセット後の TDRE の初期値は"1"です。したがって、TIE フラグが"1"に設定されると直ちに送信割り込みが発生します。また、TDRE は送信データレジスタ(TDR)への書込み、または LIN synch break 生成ビット(ECCR:LBR)への"1"書込みによってのみ TDRE フラグはクリアされます。

41.5.1.4 LIN synch break 割込み

LIN synch break 割込みについて示します。

LIN-UART がモード 3 で LIN スレーブとして動作する場合に機能します。

シリアル入力バスが 11 ビットタイム以上"0" (ドミナント) になると、拡張ステータス制御レジスタ(ESCR)の LIN synch break 検出フラグビット(LBD)が"1"にセットされます。LIN synch break 割込みと LBD フラグは、LBD フラグの"0"書込みでクリアされます。LIN synch field でキャプチャ割込みが発生する前に LBD フラグをクリアしてください。

LIN synch break 検出を行う際には、受信禁止(SCR:RXE=0)にする必要があります。

41.5.1.5 LIN synch field エッジ検出割込み

LIN synch field エッジ検出割込みについて示します。

モード 3 で LIN スレーブとして動作する場合に機能します。

LIN synch break 検出後、内部信号は LIN synch field の 1 回目の立下りエッジで"1"にセットされ、5 回目の立下りエッジ後に"0"にセットされます。インプットキャプチャ(ICP) を LSYN 入力(LSYNS0:LSYN=1)、両エッジ検出(ICS:EG1, EG0=11)、割込み許可(ICS:ICE=1)の設定にすると、LSYN(内部信号)の立上りエッジと立下りエッジで割込みが発生します。

インプットキャプチャで検出されたカウント値の差は、マスタのシリアルクロック 8 ビット分に相当し、新しいボーレートを計算することができます。スタートビットの立下りエッジを検出すると、リロードカウンタは自動的に再スタートします。

以下に各割込み発生タイミングを示します。

図 41-3 LIN synch break 検出とフラグセットタイミング

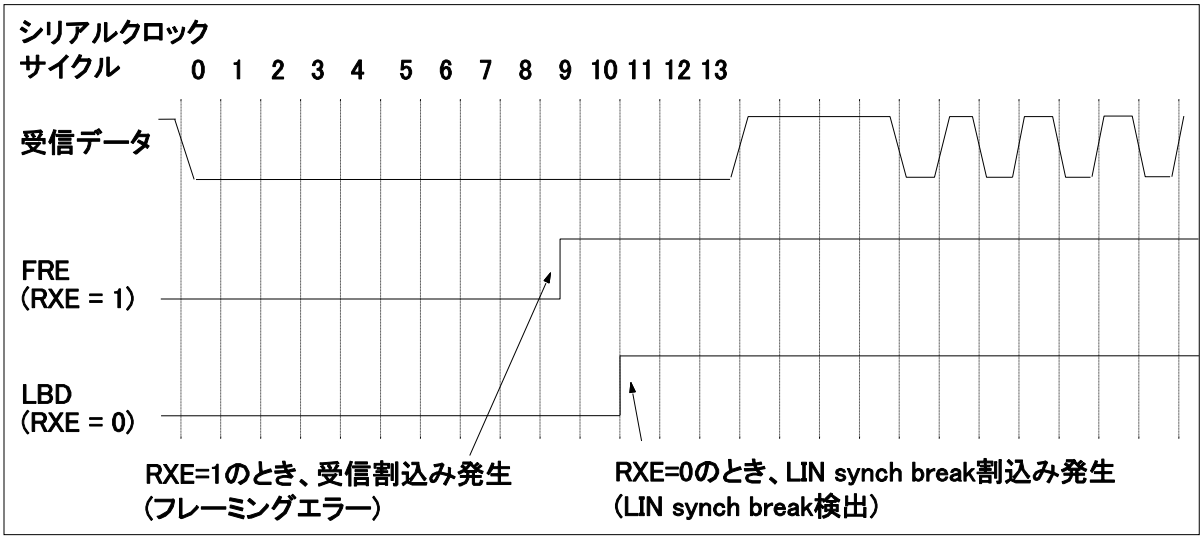
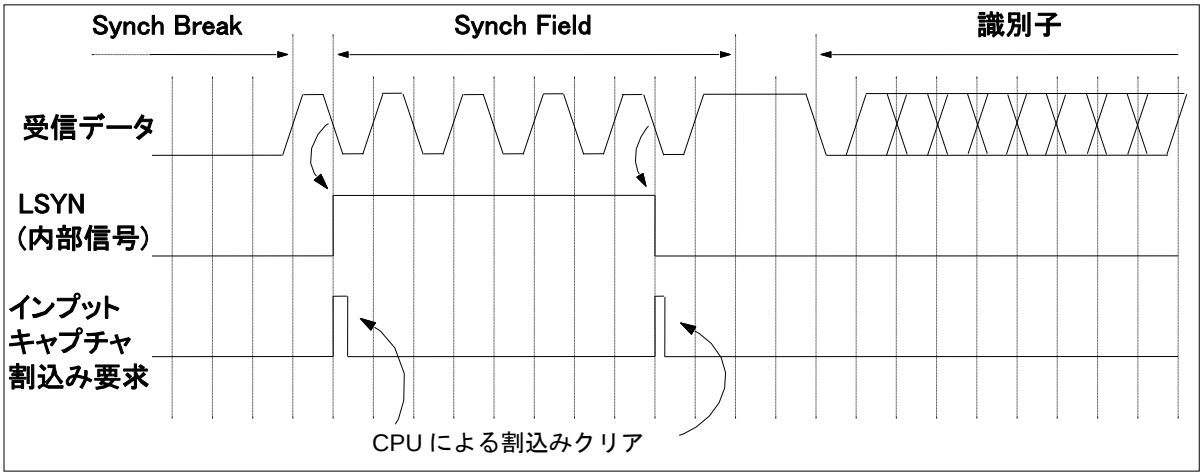


図 41-4 LIN synch field エッジ検出割込みタイミング



41.5.2　受信割込み生成とフラグセットタイミング

受信割込み生成とフラグセットタイミングについて示します。

受信割込み要因、受信完了 (SSR の RDRF ビット)と、受信エラーの発生 (SSR の PE, ORE, FRE ビット) について説明します。

41.5.2.1　受信割込み生成とフラグセットタイミング

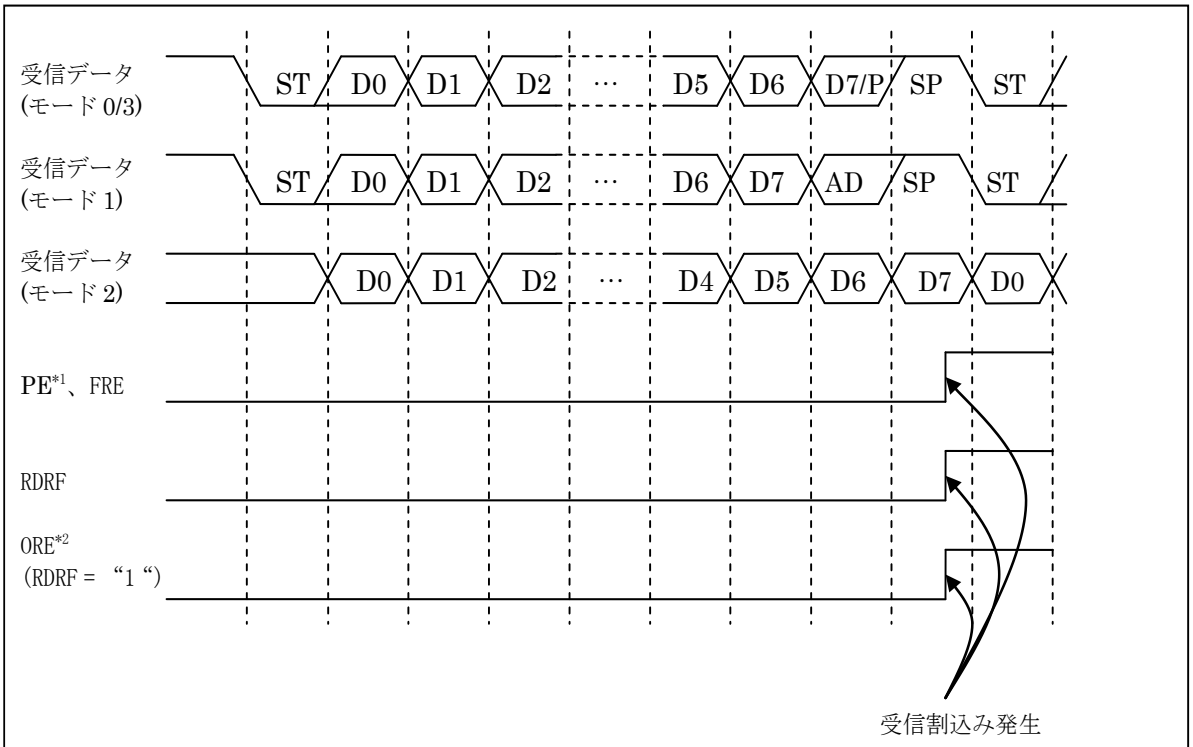
受信割込み生成とフラグセットタイミングについて示します。

動作モード 0, 1, 2 (SSM=1), 3 で最初のストップビットの検出、または動作モード 2 (SSM=0) で最終データビットが検出されることにより、受信データが受信データレジスタ (RDR) に格納されます。受信が完了したとき (SSR:RDRF=1) または受信エラーが発生 (SSR:PE, ORE, FRE=1) すると各フラグがセットされます。そのとき、受信割込みが許可 (SSR:RIE=1) されていると受信割込みが発生します。

＜注意事項＞

受信エラーが発生した場合は、いずれのモードにおいても受信データレジスタの内容は無効となります。

図 41-5 受信動作とフラグセットタイミング例

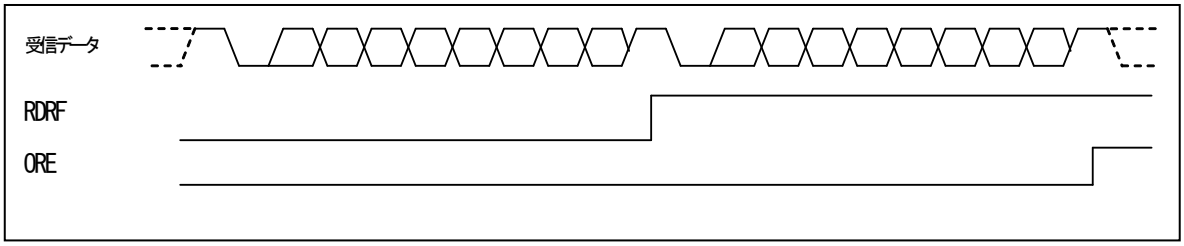


*1：動作モード 1、3 で PE フラグは常に”0”となります。
*2：受信データが読み出される前に(RDRF=1)、次のデータが転送されるとオーバーランエラーが発生します。
ST：スタートビット　SP：ストップビット　AD：モード 1(マルチプロセッサ)アドレスデータ選択ビット

＜注意事項＞

図 41-5 受信動作とフラグセットタイミング例にはモード 0 とモード 3 でのすべての受信オプションは示されていません。
ここでは、"7p1"と"8N1"です（p="偶数パリティ"または"奇数パリティ"）。

図 41-6 ORE 設定タイミング



41.5.3　送信割込み生成とフラグタイミング

送信割込み生成とフラグタイミングについて示します。

送信時、送信データが送信データレジスタ(TDR)から送信用シフトレジスタに転送され、送信が開始されると割込みが発生します。

41.5.3.1　送信割込み生成とフラグタイミング

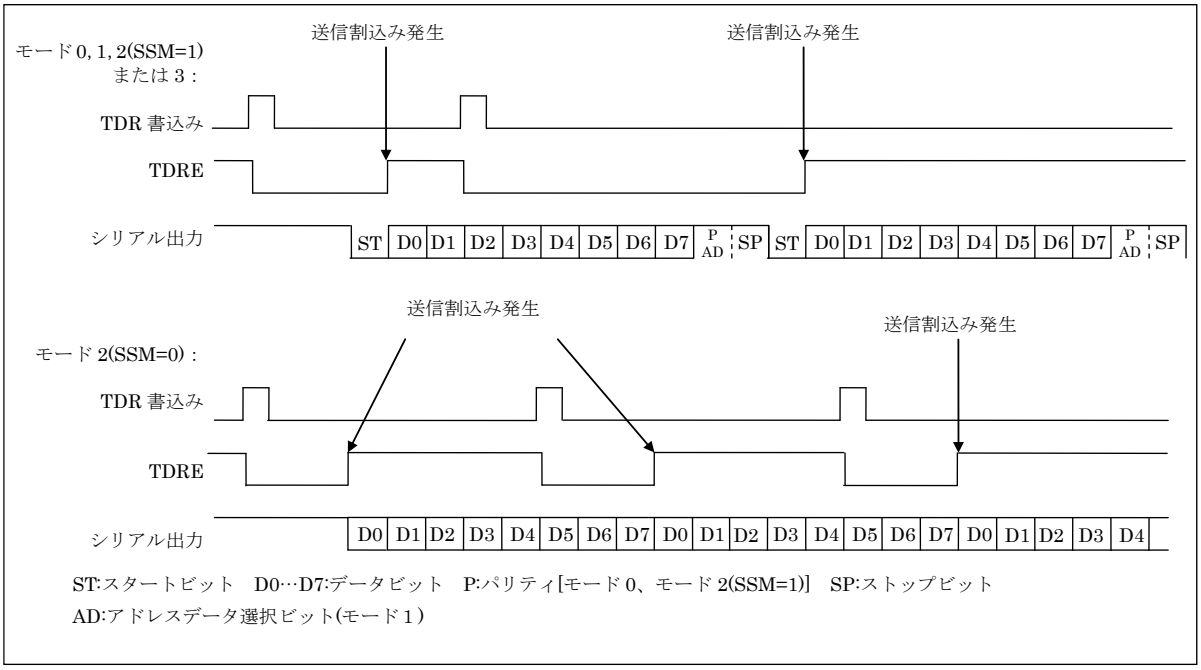
送信割込み生成とフラグタイミングについて示します。

送信データが送信データレジスタ (TDR) に書き込まれたデータが送信シフトレジスタに転送され、送信が開始されると次のデータの書込みが可能な状態(SSR:TDRE=1)になります。そのとき、送信割込みが許可(SSR:TIE=1)されていると、送信割込みが発生します。

TDRE ビットは、送信データが送信データレジスタ (TDR) へのデータ書込みにより"0"にクリアされます。

LIN-UART の送信動作とフラグ設定タイミングの一例を、図 41-7 送信動作とフラグ設定タイミング例に示します。

図 41-7 送信動作とフラグ設定タイミング例



＜注意事項＞

図 41-7 送信動作とフラグ設定タイミング例の例には、モード 0 の送信オプションのすべては示されていません。ここでは、"8p1"(p="偶数パリティ"または"奇数パリティ")です。モード 3 とモード 2 で SSM ビットが"0"の場合、パリティとアドレス/データ選択ビットは付与されません。

41.5.3.2 送信割込み要求生成タイミング

送信割込み要求生成タイミングについて示します。

送信割込みが許可 (SSR の TIE ビットが"1")されている場合に TDRE フラグが"1"になると、送信割込み要求が生成されます。

＜注意事項＞

TDRE の初期値は"1"です。したがって、送信割込みが許可(TIE="1")されると、送信完了割込みが直ちにセットされます。TDRE は読出し専用です。送信データレジスタ(TDR)への書込み、または LIN synch break 生成ビット(ECCR:LBR)への"1"書込みによってのみ TDRE フラグはクリアされます。

41.6 ボーレート

ボーレートについて示します。

LIN-UART のシリアルクロックとして下記のいずれかを選択できます。

- 専用ボーレートジェネレータ (リロードカウンタ)
- 外部クロックをボーレートジェネレータに入力 (リロードカウンタ)
- 外部クロック (SCK 端子入力クロックの直接使用)

41.6.1 ボーレート選択

ボーレート選択について示します。

図 41-8 にボーレート選択回路を示します。下記の 3 つからボーレートを選択可能です。

41.6.1.1 専用ボーレートジェネレータ (リロードカウンタ) で内部クロックを分周して得られるボーレート

専用ボーレートジェネレータ(リロードカウンタ)で内部クロックを分周して得られるボーレートについて示します。

2 つの内部リロードカウンタがあり、それぞれ送受信シリアルクロックに対応しています。ボーレートジェネレータレジスタ(BGR)で 15 ビットのリロード値を設定することにより、ボーレートを選択できます。リロードカウンタは、設定された値で内部クロックを分周します。

非同期モード、同期モード(マスタ)時に使用します。

クロックソースの設定は、内部クロックとボーレートジェネレータクロック使用を選択(SMR:EXT=0, OTO=0)してください。

41.6.1.2 専用ボーレートジェネレータ (リロードカウンタ) で外部クロックを分周して得られるボーレート

専用ボーレートジェネレータ(リロードカウンタ)で外部クロックを分周して得られるボーレートについて示します。

リロードカウンタのクロックソースに外部クロックを使用します。

ボーレートジェネレータレジスタ(BGR)で 15 ビットのリロード値を設定することにより、ボーレートを選択できます。

リロードカウンタは、設定された値で外部クロックを分周します。

非同期モード時に使用します。

クロックソースの設定は、外部クロックとボーレートジェネレータクロック使用を選択(SMR:EXT=1, OTO=0)してください。

41.6.1.3 外部クロック (1 対 1 モード) によるボーレート

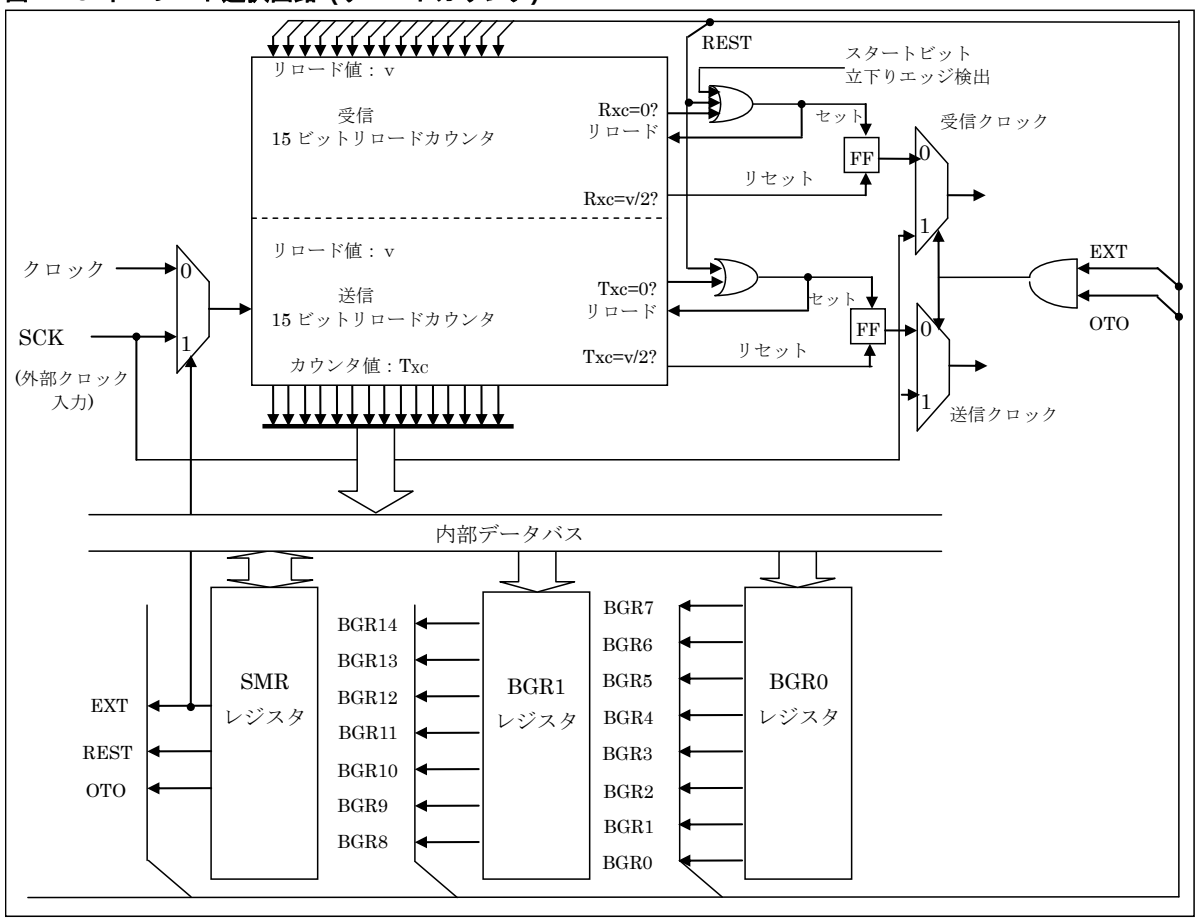
外部クロック(1 対 1 モード)によるボーレートについて示します。

LIN-UART のクロック入力端子(SCK)から入力されたクロックを、そのままボーレートとして使用します(同期モード 2 スレーブ動作(ECCR:MS=1))。

同期モード(スレーブ)時に使用します。

クロックソースの設定は、外部クロックと外部クロック直接使用を選択(SMR:EXT=1, OTO=1)してください。

図 41-8 ボーレート選択回路 (リロードカウンタ)



41.6.2　ボーレートの設定

ボーレートの設定について示します。

ボーレートの設定方法と、シリアルクロック周波数の算出結果を示します。

41.6.2.1　ボーレートの算出

ボーレートの算出について示します。

2 つの 15 ビットリロードカウンタは、ボーレートジェネレータレジスタ(BGR)で設定します。

ボーレートの計算式を以下に示します。

リロード値　　：　v=(Φ/b)-1
ボーレート値　　：　b=Φ/(v+1)
v：リロード値　　b：ボーレート　　Φ：クロック周波数

● 計算例

クロックが 16MHz で、目的のボーレートが 19200bps の場合、リロード値"v"は以下のようにして算出できます。

リロード値：

$$v = \left(\frac{16 \times 10^6}{19200} \right) - 1 = 832$$

よって実際のボーレートは下記のように計算できます。

$$b = \frac{\Phi}{(v + 1)} = \frac{16 \times 10^6}{833} = 19207.6831$$

＜注意事項＞

リロード値を"0"に設定すると、リロードカウンタは停止します。したがって、最小の分周比は 2 分周となります。
非同期モードで送受信する際、5 回オーバサンプリングして受信値を決定するため、リロード値を最低でも 4 に設定する必要があります。

41.6.2.2 クロック周波数ごとのボーレート設定例

クロック周波数ごとのボーレート設定例について示します。

次表に、クロック周波数とボーレートの設定例を示します。

表 41-7 クロックごとのボーレート設定例

ボーレート	8MHz		16MHz		24MHz		32MHz		40MHz	
	値	dev.	値	dev.	値	dev.	値	dev.	値	dev.
4M	-	-	-	-	5	0	7	0	9	0
2M	-	-	7	0	11	0	15	0	19	0
1M	7	0	15	0	23	0	31	0	39	0
500000	15	0	31	0	47	0	63	0	79	0
460800	-	-	-	-	51	-0.16	68	-0.64	86	0.22
250000	31	0	63	0	95	0	127	0	159	0
230400	-	-	-	-	103	-0.16	138	0.08	173	0.22
153600	51	-0.16	103	-0.16	155	-0.16	207	-0.16	259	-0.16
125000	63	0	127	0	191	0	255	0	319	0
115200	68	-0.64	138	0.08	207	-0.16	277	0.08	346	-0.06
76800	103	-0.16	207	-0.16	311	-0.16	416	0.08	520	0.32
57600	138	0.08	277	0.08	416	0.08	555	0.08	693	-0.06
38400	207	-0.16	416	0.08	624	0	832	-0.04	1041	0.03
28800	277	0.08	554	-0.01	832	-0.03	1110	-0.01	1388	0.01
19200	416	0.08	832	-0.03	1249	0	1666	0.02	2082	-0.02
10417	767	0	1535	0	2303	0	3071	0	3839	0.003
9600	832	-0.04	1666	0.02	2499	0	3332	-0.01	4166	0.008
7200	1110	-0.01	2221	-0.01	3332	-0.01	4443	-0.01	5555	0.008
4800	1666	0.02	3332	-0.01	4999	0	6666	0	8332	0.004
2400	3332	-0.01	6666	0	9999	0	13332	0	16666	0.002
1200	6666	0	13332	0	19999	0	26666	0	-	-
600	13332	0	26666	0	-	-	-	-	-	-
300	26666	0	-	-	-	-	-	-	-	-

＜注意事項＞

偏差の単位は"%"です。
最大同期ボーレートは、マシンクロックの 6 分周です。

41.6.2.3　外部クロックの使用

外部クロックの使用について示します。

シリアルモードレジスタ(SMR)の EXT ビットに"1"を書き込むと、外部クロックが選択されます。ボーレートジェネレータで外部クロックは内部クロックと同じように使用できます。

同期モード 2 でスレーブ動作を使用する際に、1 対 1 外部クロック入力モード(SMR:OTO=1)を選択します。外部クロックを直接シリアルクロックへ入力します。

＜注意事項＞

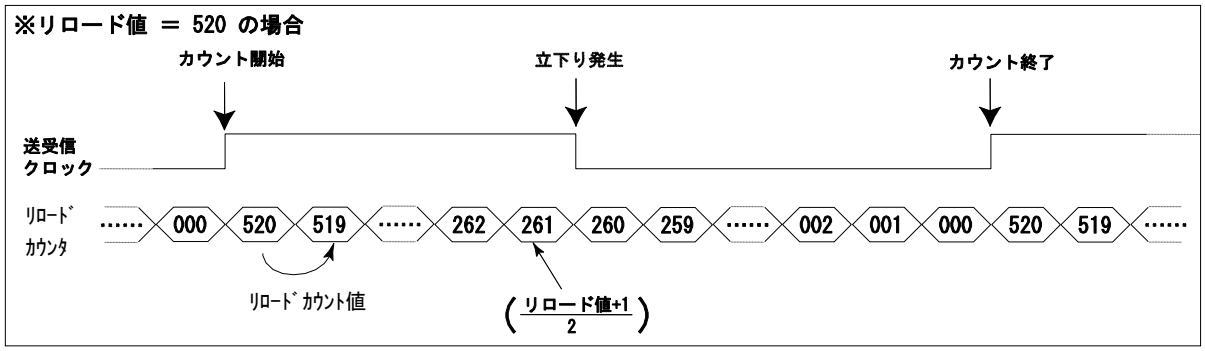
外部クロック信号は LIN-UART で内部クロックに同期します。したがって、同期不可能な外部クロックの場合、LIN-UART は誤動作します。

41.6.2.4　リロードカウンタの動作

リロードカウンタの動作について示します。

送受信リロードカウンタの動作例を示します。

図 41-9 リロードカウンタのカウント例



＜注意事項＞

シリアルクロック信号の立下りエッジはリロード値を 2 で割った値|(v + 1) / 2|をカウントした 後に発生します。

41.6.3　リロードカウンタ

リロードカウンタについて示します。

専用ボーレートジェネレータとして機能する 15 ビットのリロードカウンタです。

外部クロックまたは内部クロックより送受信クロックを生成します。

また送信リロードカウンタのカウント値をボーレートジェネレータレジスタ(BGR)より読み出すことができます。

41.6.3.1　リロードカウンタの機能

リロードカウンタの機能について示します。

リロードカウンタには、送信リロードカウンタと受信リロードカウンタがあり、専用ボーレートジェネレータとして機能します。リロード値に対する 15 ビットレジスタから構成されており、外部クロックまたは内部クロックより送受信クロックを生成します。また、送信リロードカウンタのカウント値をボーレートジェネレータレジスタ(BGR)より読み出すことができます。

41.6.3.2　カウントの開始

カウントの開始について示します。

ボーレートジェネレータレジスタ(BGR)にリロード値を書き込むと、リロードカウンタはカウントを開始します。

41.6.3.3　再スタート

再スタートについて示します。

LIN-UART リセット(SMR:UPCL への"1" 書込み) または再スタートを設定(SMR:REST への"1"書込み)すると、2 つのリロードカウンタは再スタートします。

受信リロードカウンタについては、非同期モードでスタートビットの立下りエッジが検出されることでも再スタートされ、受信データに受信シフトレジスタを同期させます。

41.6.3.4　カウンタのクリア

カウンタのクリアについて示します。

リセットによりボーレートジェネレータレジスタ(BGR)のリロード値とリロードカウンタが"00_H"にクリアされ、リロードカウンタは停止します。

LIN-UART リセット(SMR:UPCL への"1"書込み)でカウンタ値は一時"00_H"にクリアされますが、リロード値は保持されていますのでリロードカウンタは再スタートします。再スタートの設定(SMR:REST への"1"書込み)ではカウンタ値は"00_H"にクリアされません。

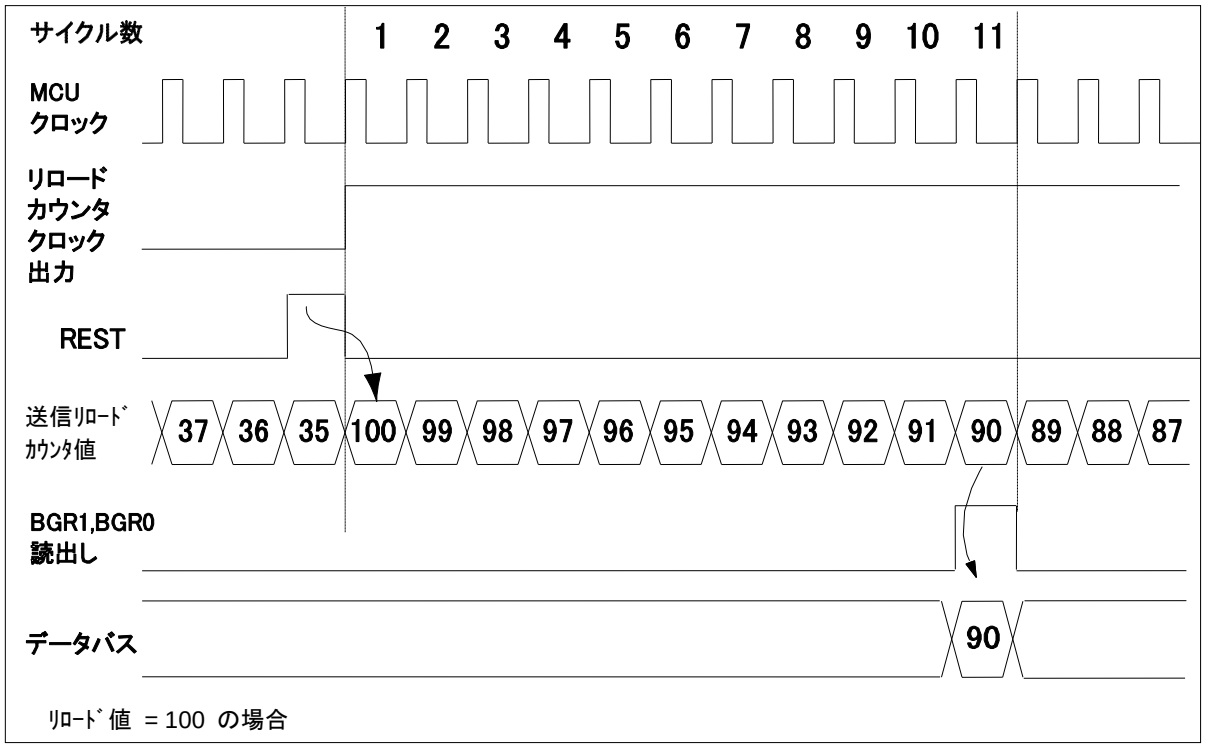
41.6.3.5 簡易タイマとしての利用

簡易タイマとしての利用について示します。

送信リロードカウンタは簡易的なタイマとしても使用できます。

図 41-10 リロードカウンタ再起動例に簡易タイマの利用例を示します。

図 41-10 リロードカウンタ再起動例



例では、REST 後の MCU クロックサイクル数 (cyc) は以下のようになります。

cyc = v - c + 1 = 100 - 90 + 1 = 11
v：リロード値　　c：読出しカウンタ値

41.7 動作説明

動作について示します。

LIN-UART は動作モード 0 では、通常双方向シリアル通信として動作します。モード 1 ではマスタ/スレーブとしてマルチプロセッサ通信を行います。モード 2 およびモード 3 ではマスタ/スレーブとして双方向通信を行います。

41.7.1 概要

動作について示します。

41.7.1.1. 動作モード

41.7.1.2. CPU 間の接続方法

41.7.1.3. 同期方式

41.7.1.4. 信号方式

41.7.1.5. 送受信動作の開始

41.7.1.6. 送受信動作の停止

41.7.1.7. 送受信中の停止

41.7.1.1 動作モード

動作モードについて説明します。

LIN-UART にはモード 0～3 の 4 つの動作モードがあります。下表に、CPU 間接続方法とデータ転送に応じて選択可能な動作モードを示します。

表 41-8 LIN-UART の動作モード

動作モード		データ長		同期方式	ストップビット長	データビットフォーマット
		パリティなし	パリティあり			
0	ノーマルモード	7 または 8 ビット		非同期	1 ビットまたは 2 ビット	LSB ファーストまたは MSB ファースト
1	マルチプロセッサモード	7 ビットまたは 8 ビット+ 1 ビット(*1)	—			
2	ノーマルモード	8 ビット		同期	なし, 1 ビット, 2 ビット	
3	LIN モード	8 ビット	—	非同期	1 ビットまたは 2 ビット	LSB ファースト

—:設定不可
(*1):"+1"はマルチプロセッサモードで通信制御用に使用されるアドレス/データ選択ビット(AD)です。

＜注意事項＞
モード 1(マルチプロセッサモード)は、マスタ/スレーブ接続時にはマスタとスレーブ両方の動作に対応します。
モード 3 では、通信フォーマットが固定されます。
モードの切換えは、LIN-UART の送受信・待受け動作をすべて解除した後に行い、LIN-UART をリセット(SMR:UPCL=1)してください。

41.7.1.2 CPU 間の接続方法

CPU 間の接続方法について示します。

外部クロック 1 対 1 接続(ノーマルモード)とマスタ/スレーブ型接続(マルチプロセッサモード)のどちらかを選択できます。どちらの方式でも、データ長, パリティ有無, 同期方式などは、すべての CPU で統一しておく必要があり、動作モードを次のように選択します。

下記に示すように動作モードを選択してください。

- 1 対 1 接続の場合 : 2 つの CPU で動作モード 0, 動作モード 2 のいずれかの同じ方式を採用する必要があります。非同期方式では動作モード 0, 同期方式では動作モード 2 を選択してください。また、動作モード 2 では 1 つの CPU をマスタへ、もう 1 つの CPU をスレーブへ設定してください。
- マスタ/スレーブ型接続の場合 : 動作モード 1 を選択します。マスタ/スレーブシステムとして使用してください。

41.7.1.3 同期方式

同期方式について示します。

非同期方式において、受信クロックは受信スタートビットの立下りエッジに同期します。同期方式では、マスタのクロック信号またはマスタとして動作したときのクロック信号によって同期させることができます。

41.7.1.4 信号方式

信号方式について示します。

NRZ(Non Return to Zero)形式です。

41.7.1.5 送受信動作の開始

送受信動作の開始について示します。

送信動作許可ビット(SCR:TXE)を"1"に設定すると、送信動作を開始します。

受信動作許可ビット(SCR:RXE)を"1"に設定すると、受信動作を開始します。

41.7.1.6 送受信動作の停止

送受信動作の停止について示します。

送信動作許可ビット(SCR:TXE)を"0"に設定すると、送信動作を停止します。

受信動作許可ビット(SCR:RXE)を"0"に設定すると、受信動作を停止します。

41.7.1.7 送受信中の停止

送受信中の停止について示します。

送受信中に送受信動作を禁止(SCR2:TXE, RXE=0)した場合は、直ちに送受信動作が停止します。この場合データは保証されません。

41.7.2　非同期モード (動作モード 0, 1)

非同期モード (動作モード 0, 1) について示します。

動作モード 0 (ノーマルモード) または動作モード 1(マルチプロセッサモード)で使用する場合、転送方式は非同期となります。

41.7.2.1　送受信データフォーマット

送受信データフォーマットについて説明します。

送受信データは必ずスタートビット("L"レベル)から始まり、指定されたデータビット長の送受信が行われ、ストップビット("H"レベル)で終了します。

ビット転送方向(LSB ファーストまたは MSB ファースト)はシリアルステータスレジスタ(SSR)の BDS ビットで決定されます。パリティありの場合、パリティビットは常に最終データビットと最初のストップビットの間に置かれます。

- 動作モード 0 では、データ長は 7 ビットまたは 8 ビットを選択します。パリティは、あり/なしの選択ができます。ストップビット長(1 または 2)を選択できます。
- 動作モード 1 では、データ長は 7 または 8 ビットで、パリティは付加されず、アドレス/データビットが付加されます。ストップビット長(1 または 2)を選択できます。

転送フレームのビット長計算式は、以下のようになります。

長さ = 1 + d + p + s
(d=データビット数[7 または 8], p=パリティ[0 または 1], s=ストップビットの数[1 または 2])

図 41-11 転送データ形式 (動作モード 0, 1)



＜注意事項＞

シリアルステータスレジスタ(SSR)の BDS ビットが"1"に設定(MSB ファースト)されると、ビットストリームは、D7, D6, ・・・, D1, D0, (P) として処理されます。また、データ長 7 ビットの場合は D6, ・・・, D1, D0, (P)の順序で処理されます。

41.7.2.2 送信動作

送信動作について示します。

送信データレジスタ(TDR) に送信データがない状態(SSR.TDRE=1)のときに，送信データレジスタ(TDR)へ送信データを書き込みます。その後，送信動作を許可(SCR.TXE=1)すると、送信が開始されます。送信データエンプティフラグビット(SSR.TDRE)は、送信データレジスタ(TDR)に送信データを書き込むと"0"になります。

送信データが送信データレジスタ(TDR)から送信シフトレジスタに転送されると、送信データエンプティフラグビット(SSR:TDRE)が再び"1"にセットされます。このとき、送信割込みが許可(SSR:TIE=1)されていれば、送信割込み要求が発生します。割込み処理において、次の送信データを送信データレジスタ(TDR)に書き込むことができます。データ長が 7 ビットに設定(CL=0) されている場合、転送方向選択ビット(BDS) の設定(LSB ファーストまたは MSB ファースト）にかかわらず、TDR の MSB が未使用ビットになります。

＜注意事項＞

送信データエンプティフラグビット(SSR:TDRE)の初期値が"1"になっているため、送信割込みが許可(SSR:TIE=1)されると直ちに割込みが発生します。

41.7.2.3 受信動作

受信動作について示します。

受信動作が許可(SCR:RXE=1)されていると、受信動作を行います。スタートビットを検出すると、シリアル制御レジスタ(SCR)で設定しているデータフォーマットにしたがって1フレームデータの受信が行われます。エラーが発生した場合にはエラーフラグ(SSR:PE, ORE, FRE)がセットされます。1 フレームの受信が完了すると、受信データは受信シフトレジスタから受信データレジスタ(RDR)へ転送され、受信データレジスタフルフラグビット(SSR:RDRF)が"1"にセットされます。このとき、受信割込み要求が許可(SSR:RIE=1)されていると受信割込み要求を出力します。

受信データを読み出す際には、1 フレームデータの受信完了後に、エラーフラグの状態を確認し、正常に受信が行われていれば受信データレジスタ(RDR)から受信データを読み出してください。受信エラーが発生している場合には，エラー処理を行ってください。

受信データの読出しで、受信データレジスタフルフラグビット(SSR:RDRF)は"0"にクリアされます。データ長が 7 ビットに設定(CL=0) されている場合、転送方向選択ビット(BDS) の設定(LSB ファーストまたは MSB ファースト）にかかわらず、TDR の MSB が未使用ビットになります。

＜注意事項＞

受信データレジスタ(RDR)のデータは、受信データレジスタフルフラグビット(SSR:RDRF)が"1"にセットされ、エラーが発生しなかった(SSR:PE, ORE, FRE=0)場合に有効となります。

41.7.2.4 使用クロック

使用クロックについて示します。

内部クロックまたは外部クロックを使用します。ボーレートは、ボーレートジェネレータを選択(SMR:EXT=0 または 1, OTO=0)してください。

41.7.2.5 ストップビット

ストップビットについて示します。

送信時に、1 ビットまたは 2 ビットのストップビットを選択できます。2 ビットのストップビットを選択したときは、最初のストップビットのみ受信時に検出されます。

最初のストップビットが検出されると受信データレジスタフルフラグ(SSR:RDRF)が"1"になります。その後スタートビットが検出されない場合、受信バスアイドルフラグ(ECCR:RBI)が"1"になり、受信動作がないことを示します。

41.7.2.6 エラー検出

エラー検出について示します。

モード 0 では、パリティエラー、オーバランエラー、フレーミングエラーが検出できます。

モード 1 では、オーバランエラー、フレーミングエラーが検出できます。パリティエラーは検出できません。

41.7.2.7 パリティ

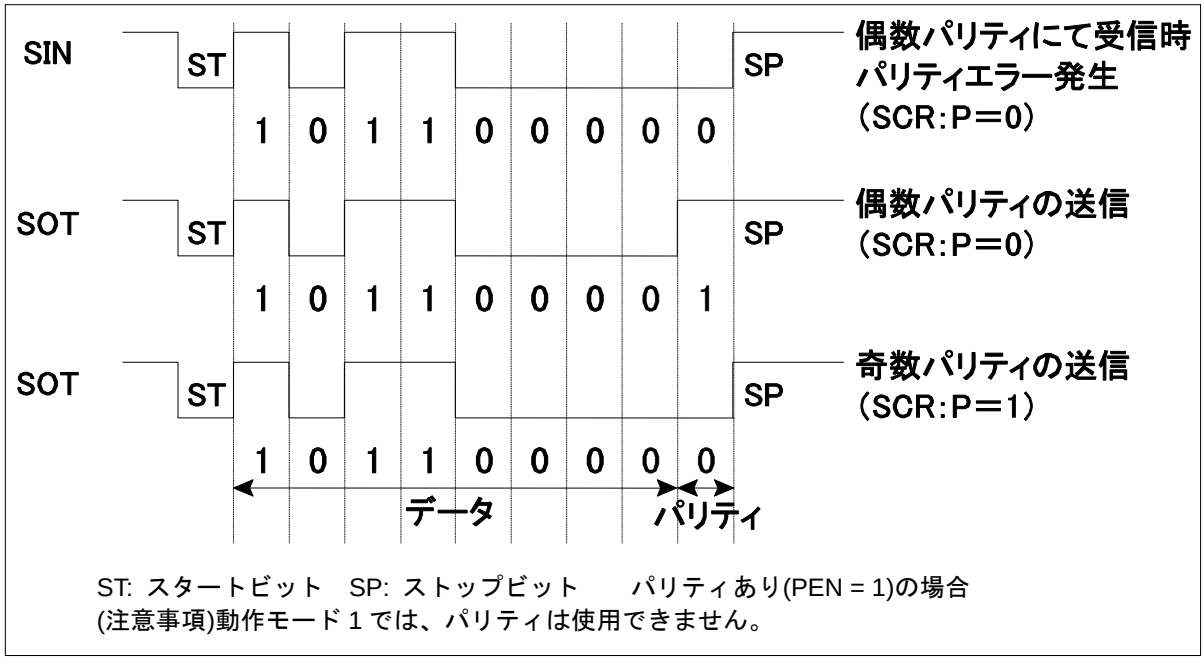
パリティについて示します。

パリティビットの付加(送信時)、検出(受信時)を設定できます。

パリティ許可ビット(SCR:PEN)でパリティの有無を、パリティ選択ビット(SCR:P)で偶数/奇数パリティを選択できます。

動作モード 1 では、パリティを使用できません。

図 41-12 パリティ有効時の送受信データ



41.7.2.8 データ信号方式

データ信号方式について示します。

NRZ データフォーマットです。

41.7.2.9 データ転送方法

データ転送方法について示します。

データビット転送方法を LSB または MSB ファーストに選択できます。

41.7.3　同期モード (動作モード 2)

同期モード(動作モード 2)について示します。

LIN-UART 動作モード 2（ノーマルモード）で使用する場合、転送方式はクロック同期となります。

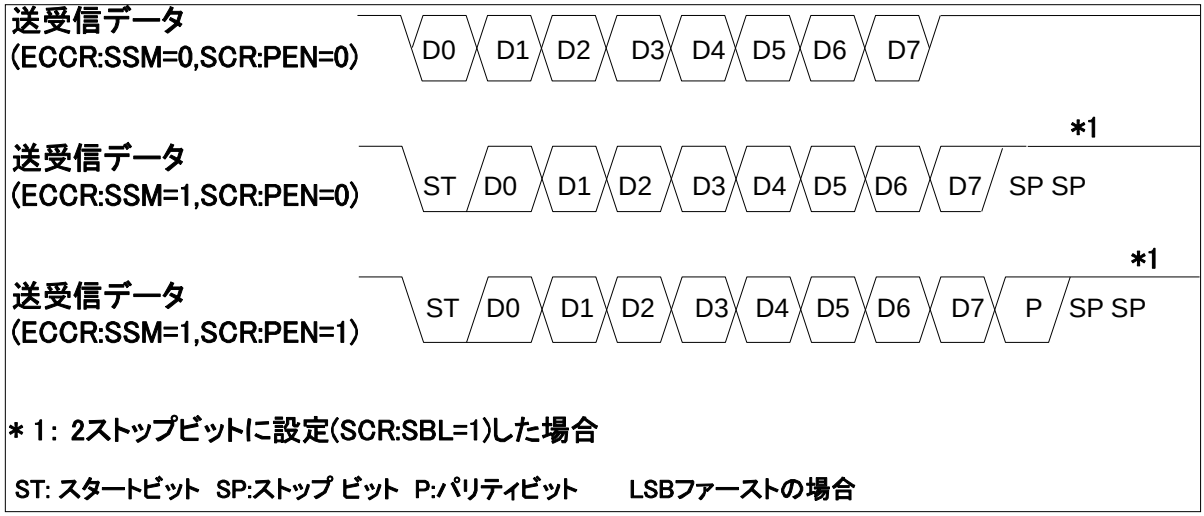
41.7.3.1　送受信データフォーマット

送受信データフォーマットについて示します。

同期モードでは、8 ビットデータを送受信し、スタートビット, ストップビットの有無を選択(ECCR:SSM)できます。またスタート/ストップあり(ECCR:SSM=1)のとき、パリティビットの有無を選択(SCR:PEN)できます。

以下に、同期モード使用時のデータフォーマットを示します。

図 41-13 送受信データフォーマット（動作モード 2）



41.7.3.2　マスタ/スレーブ設定

マスタ/スレーブ設定について示します。

モード 2 では、マスタとスレーブの設定ができます。

マスタ(ECCR:MS=0)は、シリアルクロックを発生します。

スレーブ(ECCR:MS=1)は、外部クロックを受信します。外部クロックを選択し、1 対 1 外部入力に設定(SMR:EXT, OTO=1)してください。

41.7.3.3 サンプリングエッジ選択

サンプリングエッジ選択について示します。

データビットをサンプリングする際、サンプリングエッジを選択できます。

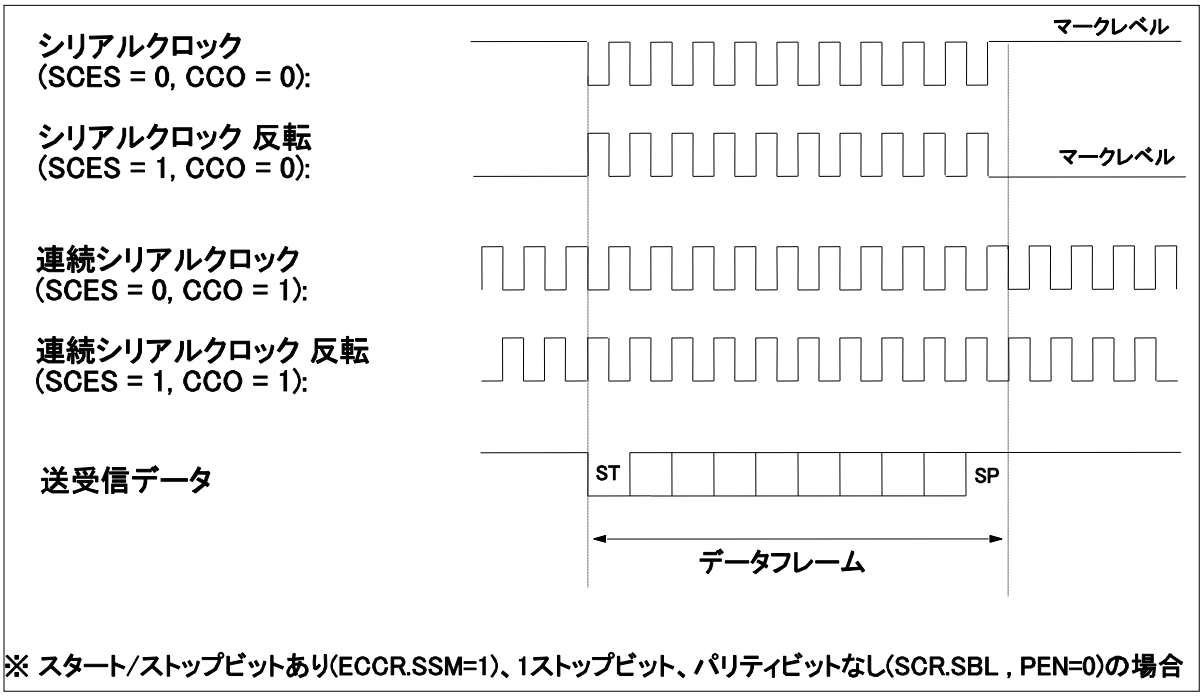
- 立上りエッジでサンプリング(ESCR:SCES=0)：ノーマルクロック
- 立下りエッジでサンプリング(ESCR:SCES=1)：クロック反転

送受信クロックとして，シリアルクロック(ノーマル/遅延)と連続シリアルクロックを選択できます。

マスタモードで、連続シリアルクロックを使用していない場合(ESCR:CCO=0)にクロックを反転すると(ESCR:SCES=1)、クロック信号のマークレベルは"L"になります。

以下に、サンプリングエッジの選択による反転クロックを示します。

図 41-14 サンプリングエッジ選択による反転クロック



※ スタート/ストップビットあり(ECCR.SSM=1)、1ストップビット、パリティビットなし(SCR.SBL , PEN=0)の場合

41.7.3.4 クロック供給

クロック供給について示します。

クロック同期モードでは、送受信ビット数に等しい数のクロックの供給が必要になります。

＜注意事項＞

スタート/ストップビットありの通信時は、クロックサイクル数と、スタート/ストップビットが付加された数が一致していなければなりません。

41.7.3.5 使用クロック

使用クロックについて示します。

マスタでは、内部クロックを使用します。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発生します。

ボーレートは、ボーレートジェネレータを選択(SMR:EXT=0, OTO=0)してください。

スレーブでは、外部クロックを使用します。送信側の送信データレジスタにデータがあることを確認した後、外部から正確に 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(SCES=0 のとき"H", SCES=1 のとき"L")にする必要があります。

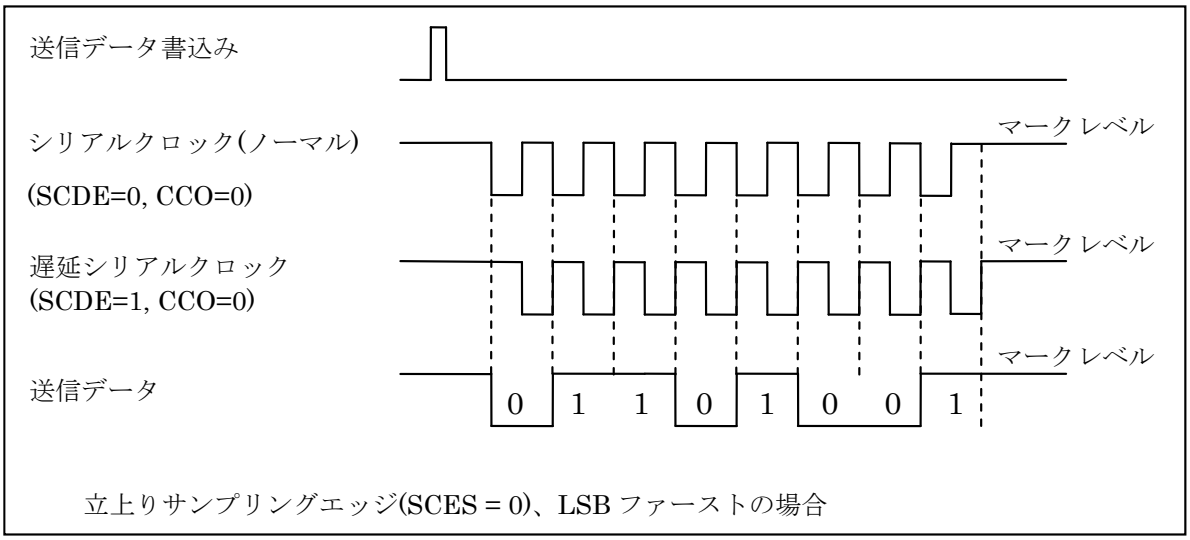
ボーレートは、外部クロック(1 対 1)を選択(SMR:EXT=1, OTO=1)してください。

41.7.3.6 遅延シリアルクロック

遅延シリアルクロックについて説明します。

ECCR の SCDE ビットを"1"に設定すると、以下に示すように遅延した送信クロックを出力します。本機能は、受信側のデバイスが、クロックの立上りエッジまたは立下りエッジでデータをサンプリングする場合に必要となります。

図 41-15 送信クロックの遅延シリアルクロック出力



＜注意事項＞

- 送受信クロックに連続シリアルクロックが選択されているとき(ESCR:CCO=1)は，シリアルクロック(ノーマル/遅延)を設定(ECCR:SCDE)しても連続シリアルクロックになり，遅延しません。
- 同期モードのスレーブモード(ECCR：MS＝1)で使用する場合は、シリアルクロック遅延許可ビット(ECCR:SCDE)は"0"にしてください。

41.7.3.7 連続シリアルクロック

連続シリアルクロックについて示します。

連続シリアルクロックを選択した場合は、マスタの SCK 端子からシリアルクロックが連続的に出力されます。また連続シリアルクロックを使用する際には、送受信の開始/終了を明確にするために、スタート/ストップビットを必ず付加(ECCR:SSM=1)してください。

41.7.3.8 パリティ

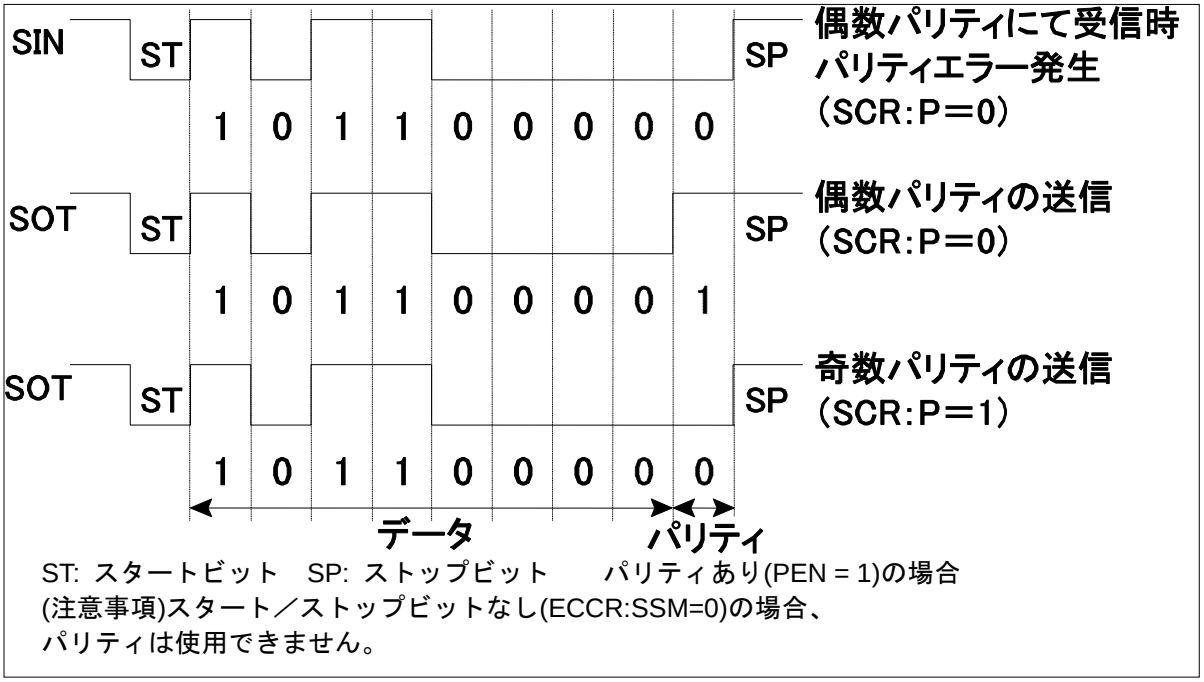
パリティについて示します。

パリティビットの付加 (送信時), 検出 (受信時) を設定できます。

パリティ許可ビット(SCR:PEN)でパリティの有無を、パリティ選択ビット (SCR:P) で偶数/奇数パリティを選択できます。

スタート/ストップビットなしのときは、パリティを使用できません。

図 41-16 パリティ有効時の送受信データ



ST: スタートビット　SP: ストップビット　　パリティあり(PEN = 1)の場合
(注意事項)スタート／ストップビットなし(ECCR:SSM=0)の場合、
パリティは使用できません。

41.7.3.9 データ信号方式

データ信号方式について示します。

NRZ データフォーマットです。

41.7.3.10 ストップビット

ストップビットについて示します。

送信時に、1 ビットまたは 2 ビットのストップビットを選択できます。2 ビットのストップビットを選択したときは最初のストップビットのみ受信時に検出されます。

最初のストップビットが検出されると受信データレジスタフルフラグ(SSR:RDRF) が"1" になります。その後スタートビットが検出されない場合、受信バスアイドルフラグ(ECCR:RBI) が"1" になり、受信動作がないことを示します。

41.7.3.11 エラー検出

エラー検出について示します。

スタート/ストップビットがない場合(ECCR:SSM=0)は、オーバランエラーのみ検出されます。

スタート/ストップビットとパリティビットがある場合は、パリティエラー，オーバランエラー，フレーミングエラーが検出できます。

41.7.3.12 通信開始

通信開始について示します。

送信データレジスタ(TDR)への書込みで通信を開始します。受信する場合でも通信を開始するために、シリアルデータ出力を禁止(SMR:SOE=0)した後、必ず仮のデータを送信データレジスタ(TDR)へ書き込む必要がありますので注意してください。

41.7.3.13 通信終了

通信終了について示します。

1 フレームのデータ送受信が終了すると、受信データレジスタフルフラグビット(SSR:RDRF)が"1"にセットされます。受信後はエラーフラグを確認し、通信が正常に行われたどうか判断してください。

＜注意事項＞
連続クロックとスタート/ストップビットを使用して、非同期モードのような二重通信方式が可能です。

41.7.3.14 データ転送方法

データ転送方法について示します。

データビット転送方法を LSB または MSB ファーストに選択できます。

41.7.4 LIN モード (動作モード 3)

LIN モード(動作モード 3)について示します。

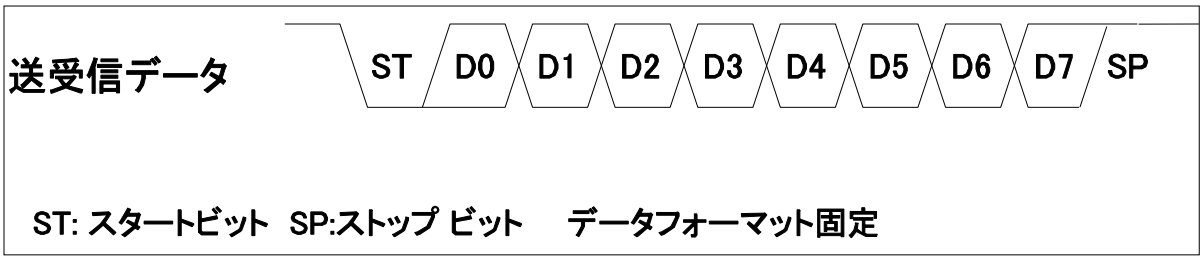
LIN-UART 動作モード 3 で LIN マスタ/スレーブ機能が動作します。転送方式は非同期となります。

41.7.4.1 送受信データフォーマット

送受信データフォーマットについて示します。

動作モード 3 では、データフォーマットが固定されます。8 ビットデータを送受信し、スタート/ストップビットが付加され LSB ファーストになります。またパリティビットは付加されません。

図 41-17 送受信データフォーマット



41.7.4.2 LIN マスタ動作

LIN マスタ動作について示します。

LIN マスタモードは、マスタにスレーブを同期させるためマスタですべてのボーレートを決定します。

LIN 通信の開始は、マスタからスレーブへ LIN synch break を送信します。LIN synch break は SOT 端子に"L"を 13～16 ビット発生させます。LIN synch break 長を選択(ESCR:LBL1/LBL0)し、LIN synch break を生成(ECCR:LBR=1)してください。

LIN synch break の後、LIN synch field(55_H)を送信します。LIN synch break 生成(ECCR:LBR への"1"書込み)により、送信データあり(SSR:TDRE=0)となりますが、送信データレジスタ(TDR)へ 55_H を書き込むことができます。これにより、LIN synch break 後の送信割込みは発生しません。LIN synch field 後、非同期通信を行います。LIN synch field(55_H)が送信された後、非同期通信を行います。

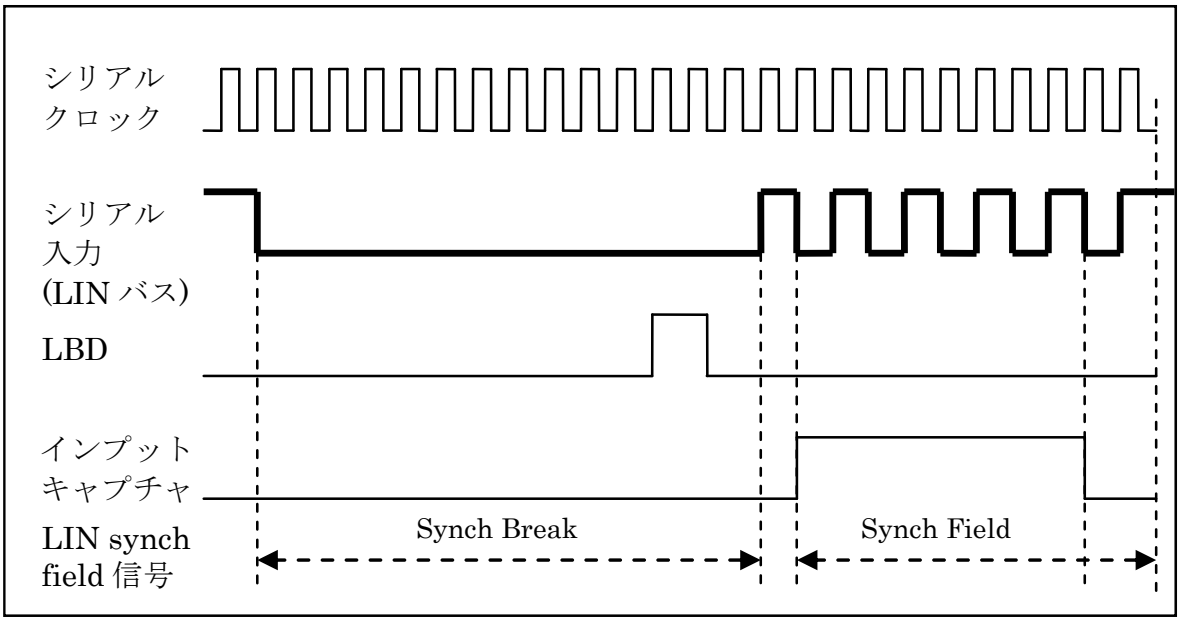
41.7.4.3 LIN スレーブ動作

LIN スレーブ動作について示します。

LIN スレーブモードは、マスタボーレートに同期させます。バス(シリアル入力)が 11 ビットタイム以上"0"になると LIN マスタの LIN synch break が検出(ESCR:LBD=1) されます。LIN synch break 検出を行う際には、受信禁止(SCR:RXE=0)または受信割込み禁止(SSR:RIE=0)にする必要があります。このとき、LIN synch break 割込みが許可(ESCR:LBIE=1)されていると、割込みが発生します。LIN synch break 検出フラグビット(ESCR:LBD)の"0"書込みで割込みがクリアされます。LIN synch break 検出後，内部信号は LIN synch field の 1 回目の立下りエッジで"1"にセットされ、5 回目の立下りエッジ後に"0"にセットされます。両方のエッジ検出時に、インプットキャプチャ割込みが許可(ICS:ICE=1)されていると割込みが発生します。LIN synch field 検出時の内部信号は、マスタのシリアルクロック 8 ビット分に相当し、インプットキャプチャでカウントされます。その後、非同期通信を行うことができます。「41.7.2 非同期モード (動作モード 0, 1)」を参照してください。

以下に、LIN 通信フレームの典型的なスタートと LIN-UART の動作を示します。

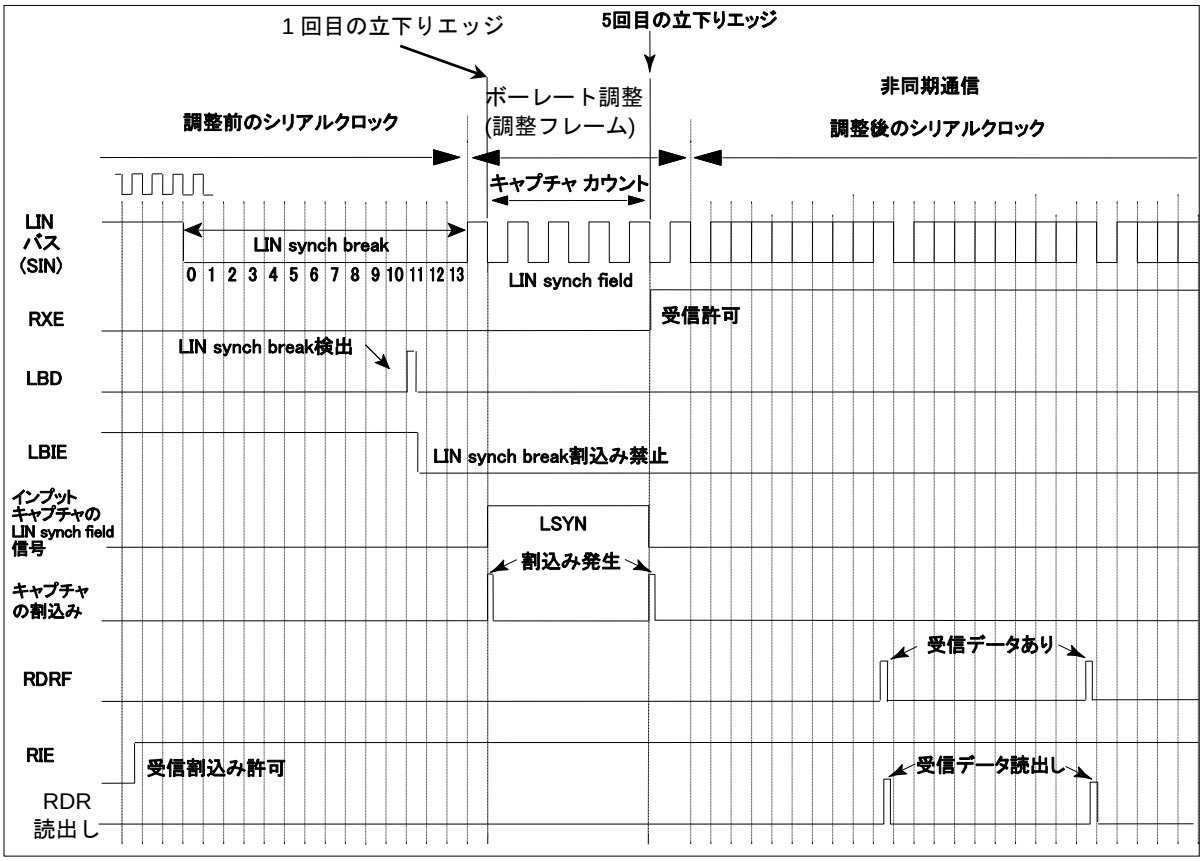
図 41-18 LIN スレーブ動作



41.7.4.4 LIN バスタイミング

LIN バスタイミングについて示します。

図 41-19 LIN バスタイミングと LIN-UART の信号



41.7.4.5 ボーレートの計算について

ボーレートの計算について説明します。

例として LIN-UART ch.3 の動作を以下に説明します。Synch field の最初の立下りエッジを LIN-UART ch.3 が検出すると、インプットキャプチャ(ICU1)に入力される内部信号を"H"にして ICU1 をスタートさせます。この内部信号は 5 番目の立下りエッジで"L"になります。ICU1 は LIN モードに設定されている必要があります(LSYNS0:LSYN1)。また、ICU1 の割込みイネーブル設定(ICS01:ICE1)、両エッジ検出に設定する必要があります(ICS01:EG11, ICS01:EG10)。ICU1 入力信号が"1"の時間がボーレートを 8 倍した値となります。ボーレート設定値は下式で算出できます。

● フリーランタイマがオーバフローしていない場合

BGR 値={(b-a)×Fe/(8×φ)}-1

● フリーランタイマがオーバフローした場合

BGR 値= {(max+1+b-a)×Fe/(8×φ)}-1

max : フリーランタイマの最大値
a : 1 度目の割込み後の ICU データレジスタ値
b : 2 度目の割込み後の ICU データレジスタ値
φ: マシンクロック周波数(MHz)
Fe : 外部クロック周波数(MHz)
内部ボーレートジェネレータ使用時(EXT=0), Fe=φとして計算

＜注意事項＞

上記のように LIN スレーブモード時、Sync field で新たに計算された BGR 値がボーレート±15%以上の誤差が生じた場合は、ボーレートの設定は行わないでください。

LIN-UART と ICU の関係につきましては、『フリーランタイマ』の章の『 5. 動作説明』の章および『インプットキャプチャ』の章の『 5. 動作説明』を参照してください。

41.7.4.6 使用クロック

使用クロックについて示します。

内部クロックを使用します。ボーレートは、ボーレートジェネレータを選択 (SMR:EXT=0 または 1, OTO=0)してください。

41.7.4.7 データ信号方式

データ信号方式について示します。

NRZ データフォーマットです。

41.7.4.8 ストップビット

ストップビットについて示します。

送信時に、1 ビットまたは 2 ビットのストップビットを選択できます。2 ビットのストップビットを選択したときは最初のストップビットのみ受信時に検出されます。

最初のストップビットが検出されると受信データレジスタフルフラグ(SSR:RDRF) が"1" になります。その後スタートビットが検出されない場合、受信バスアイドルフラグ(ECCR:RBI) が"1" になり、受信動作がないことを示します。

41.7.4.9 エラー検出

エラー検出について示します。

オーバランエラー、フレーミングエラーが検出できます。

41.7.5　シリアル端子への直接アクセス

シリアル端子への直接アクセスについて示します。

LIN-UART は送信端子(SOT)，受信端子(SIN)へ、直接アクセスすることができます。

LIN-UART は、プログラマがシリアル入出力端子に直接アクセスすることができます。

シリアル入力端子(SIN)の状態をシリアル入出力端子直接アクセスビット(ESCR:SIOP)で読み出すことができます。

シリアル出力端子(SOT)への直接書込みを可能(ESCR:SOPE=1)にし，シリアル入出力端子直接アクセスビット(ESCR:SIOP)へ"0"または"1"を書き込んだ後、シリアル出力を許可(SMR:SOE=1)にすると, シリアル出力端子(SOT)の値を任意に設定することができます。

LIN モードの場合は、送信したデータの読出し、または物理的な LIN バス線信号が間違っていたときのエラー処理に使用できます。

＜注意事項＞

送信動作がない(送信シフトレジスタが空のとき)場合のみアクセス可能です。出力端子にアクセスする前(SMR:SOE=1)に、シリアル入出力端子直接アクセスビット(ESCR:SIOP)へ値を書き込んでください。これは, SIOP ビットが以前の値を保持しているため、期待しないレベルの信号が出力されることを防ぐためです。リードモディファイライト動作時は読出しサイクル内の SOT 端子の値を返します。

41.7.6 双方向通信機能 (ノーマルモード)

双方向通信機能(ノーマルモード)について示します。

動作モード 0, 2 では通常のシリアル双方向通信が可能です。動作モード 0 は非同期通信、動作モード 2 は同期通信の選択ができます。

以下に、ノーマルモード (動作モード 0, 2)での LIN-UART の設定を示します。

図 41-20 動作モード 0, 2 での LIN-UART の設定

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR, SMR	PEN	P	SBL	CL	AD	CRE	RXE	TXE	MD1	MD0	OTO	EXT	REST	UPCL	SCKE	SOE
モード 0 →	◎	◎	◎	◎	×	◎	◎	◎	0	0	0	◎	◎	◎	0	◎
モード 2 (MS=0) →	□	□	□	+	×	◎	◎	◎	1	0	0	◎	◎	◎	1	◎
モード 2 (MS=1) →	□	□	□	+	×	◎	◎	◎	1	0	1	1	◎	◎	0	◎

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSR, TDR/RDR	PE	ORE	FRE	RDRF	TDRE	BDS	RIE	TIE	変換データ設定(書込み時) /受信データ保持(読出し時)							
モード 0 →	◎	◎	◎	◎	◎	◎	◎	◎								
モード 2 (MS=0) →	□	◎	□	◎	◎	◎	◎	◎								
モード 2 (MS=1) →	□	◎	□	◎	◎	◎	◎	◎								

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ESCR, ECCR	LBIE	LBD	LBL1	LBL0	SOPE	SIOP	CCO	SCES	予約	LBR	MS	SCDE	SSM	予約	RBI	TBI
モード 0 →	×	×	×	×	◎	◎	×	+	0	0	+	+	+	0	◎	◎
モード 2 (MS=0) →	+	×	×	×	◎	◎	◎	◎	0	+	0	◎	◎	0	×	◎
モード 2 (MS=1) →	+	×	×	×	◎	◎	0	◎	0	+	1	0	◎	0	×	×

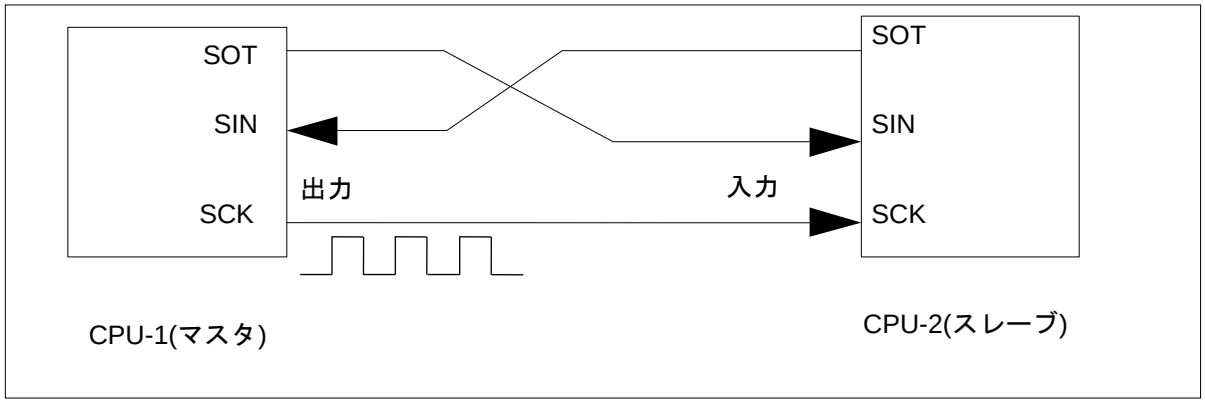
◎ : 使用可能ビット
× : 未使用ビット
1 : "1"を設定
0 : "0"を設定
□ : SSM=1 (同期スタート/ストップビット付加) の場合に使用。
+ : 自動的に正しく設定されるビット

41.7.6.1 CPU 間の接続

CPU 間の接続について示します。

以下に LIN-UART モード 2 での 2CPU 間接続を示します。

図 41-21 LIN-UART 動作モード 2 双方向通信の接続例

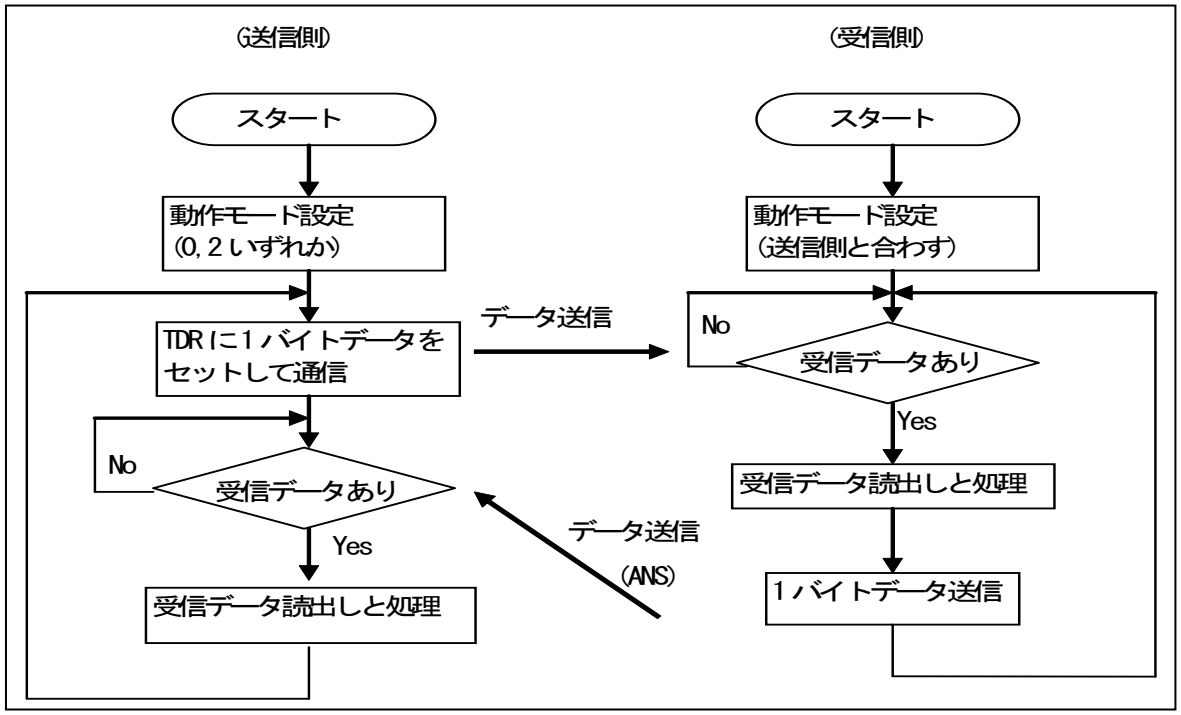


41.7.6.2 通信手順

通信手順について示します。

通信は、送信データが準備できたときに送信側から任意なタイミングで開始します。受信側で送信データを受取ると、定期的に ANS (例では 1 バイトごと)を返します。以下に双方向通信フローチャートの一例を示します。

図 41-22 双方向通信フローチャートの例



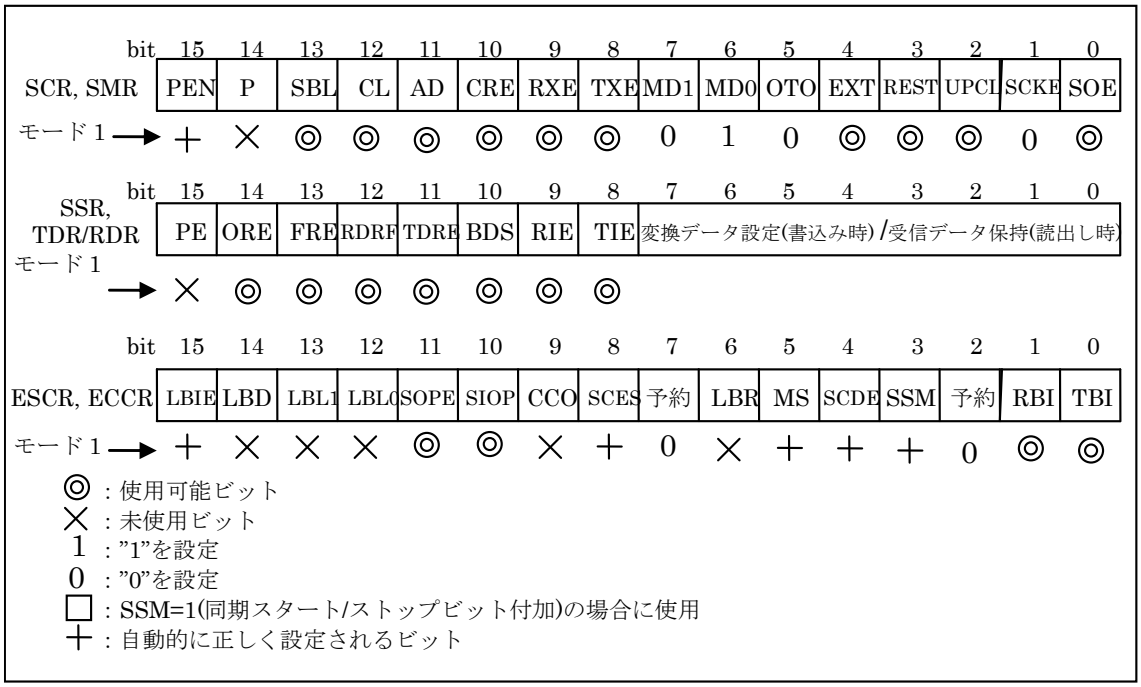
41.7.7　マスタ／スレーブ通信機能 (マルチプロセッサモード)

マスタ／スレーブ通信機能(マルチプロセッサモード)について示します。

動作モード 1 で、複数 CPU のマスタ／スレーブモード接続による通信が可能です。マスタまたはスレーブとして使用できます。

以下に、マルチプロセッサモード (動作モード 1)での LIN-UART の設定を示します。

図 41-23 動作モード 1 での LIN-UART の設定

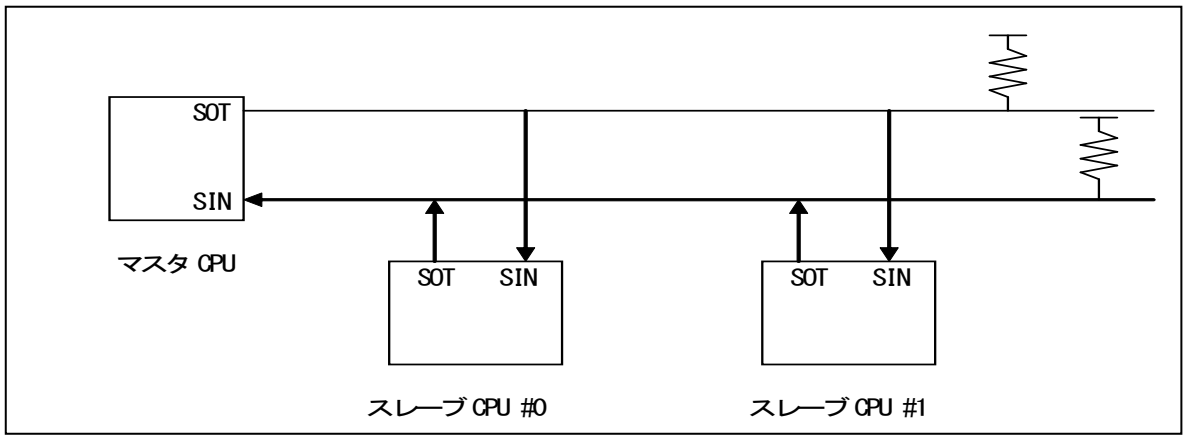


41.7.7.1 CPU 間の接続

CPU 間の接続について示します。

以下に、2本の通信線で接続されたマスタ CPU と複数のスレーブ CPU からなる通信システムを示します。LIN-UART はマスタまたはスレーブのどちらでも使用できます。

図 41-24 LIN-UART マスタースレーブ通信の接続例



41.7.7.2 機能選択

機能選択について示します。

マスタ/スレーブ型通信では、下表に示すように動作モードとデータ転送方式を選択してください。

表 41-9 マスタースレーブ通信機能の設定

	動作モード		データ	パリティ	同期方法	ストップビット	ビット方向
	マスタ CPU	スレーブ CPU					
アドレス送受信	モード 1 (AD ビット送受信)		AD="1" + 7 または 8 ビット アドレス	なし	非同期	1 ビット または 2 ビット	LSB または MSB ファースト
データ送受信			AD="0" + 7 または 8 ビット データ				

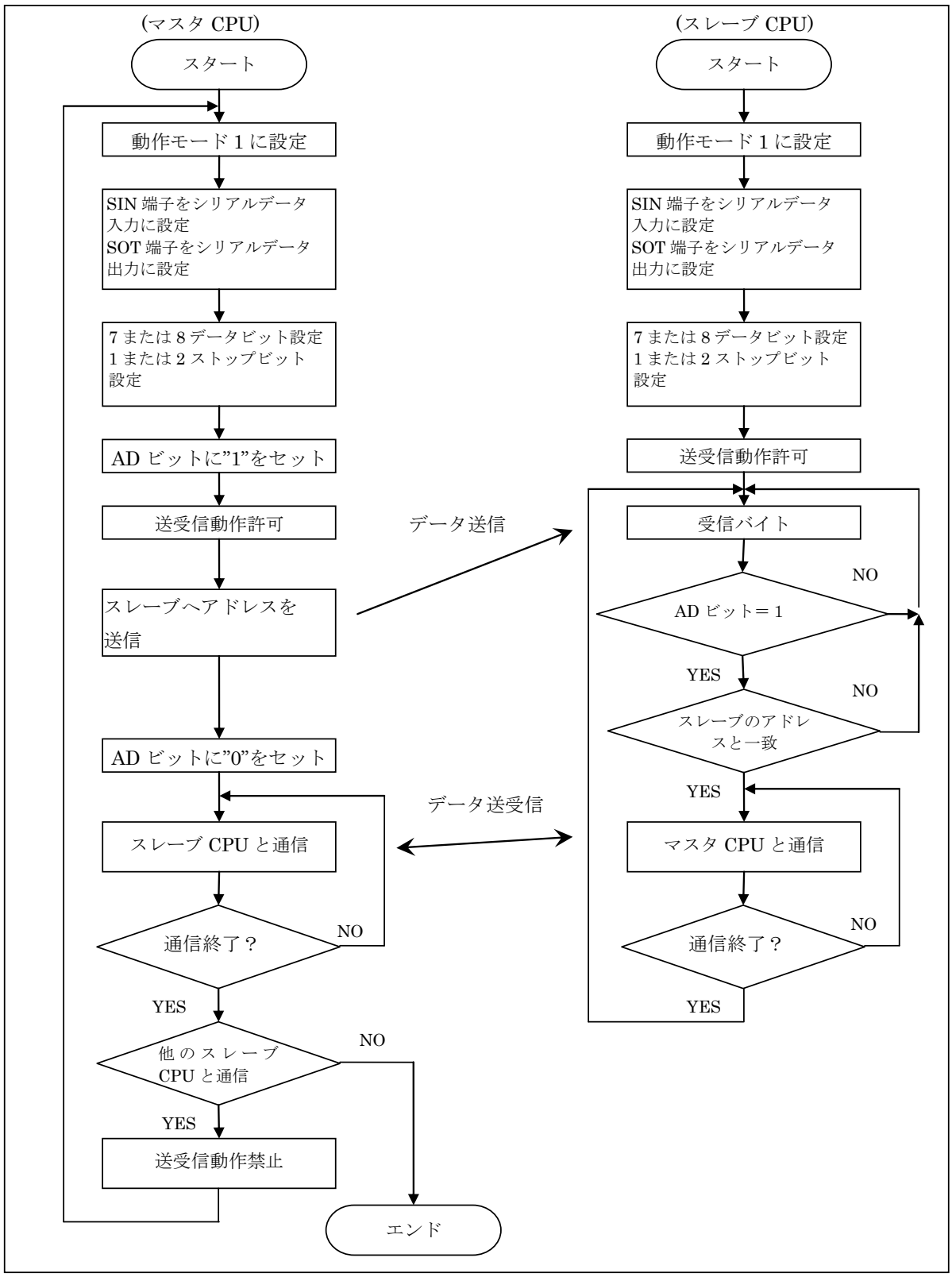
41.7.7.3 通信手順

通信手順について示します。

通信はマスタ CPU がアドレスデータを送信することによって始まります。アドレスデータとは AD ビットを"1"としたデータで、通信先となるスレーブ CPU を選択します。各スレーブ CPU はプログラムでアドレスデータを判断し、割当てられたアドレスと一致した場合にマスタ CPU との通信をします。

以下に、マスタ／スレーブ型通信(マルチプロセッサモード)のフローチャート例を示します。

図 41-25 マスタ／スレーブ通信のフローチャート



41.7.8 LIN 通信機能

LIN 通信機能について示します。

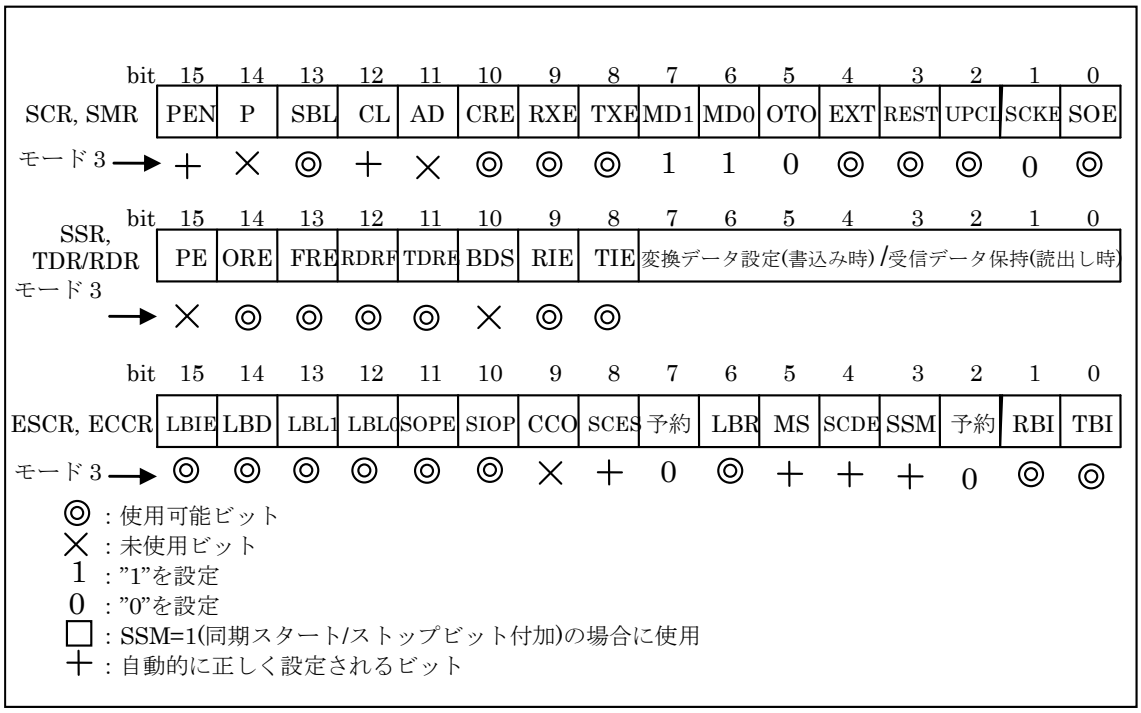
LIN-UART 通信で LIN デバイスは、LIN マスタ/スレーブシステムが使用できます。

41.7.8.1 LIN マスタ/スレーブ通信機能

LIN マスタ/スレーブ通信機能について示します。

以下に、LIN 通信モード(動作モード 3)での LIN-UART の設定を示します。

図 41-26 動作モード 3(LIN)での LIN-UART の設定



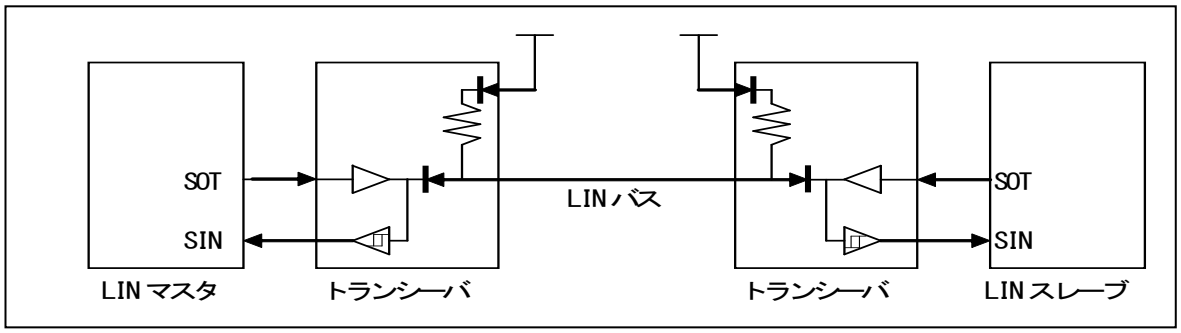
41.7.8.2 LIN デバイスの接続

LIN デバイスの接続について示します。

以下に、LIN マスタデバイスと LIN スレーブデバイスの接続を示します。

LIN-UART は LIN マスタまたは LIN スレーブとして設定可能です。

図 41-27 LIN バスシステムの接続例



41.7.9 LIN 通信モード (動作モード 3) LIN-UART サンプルフローチャート

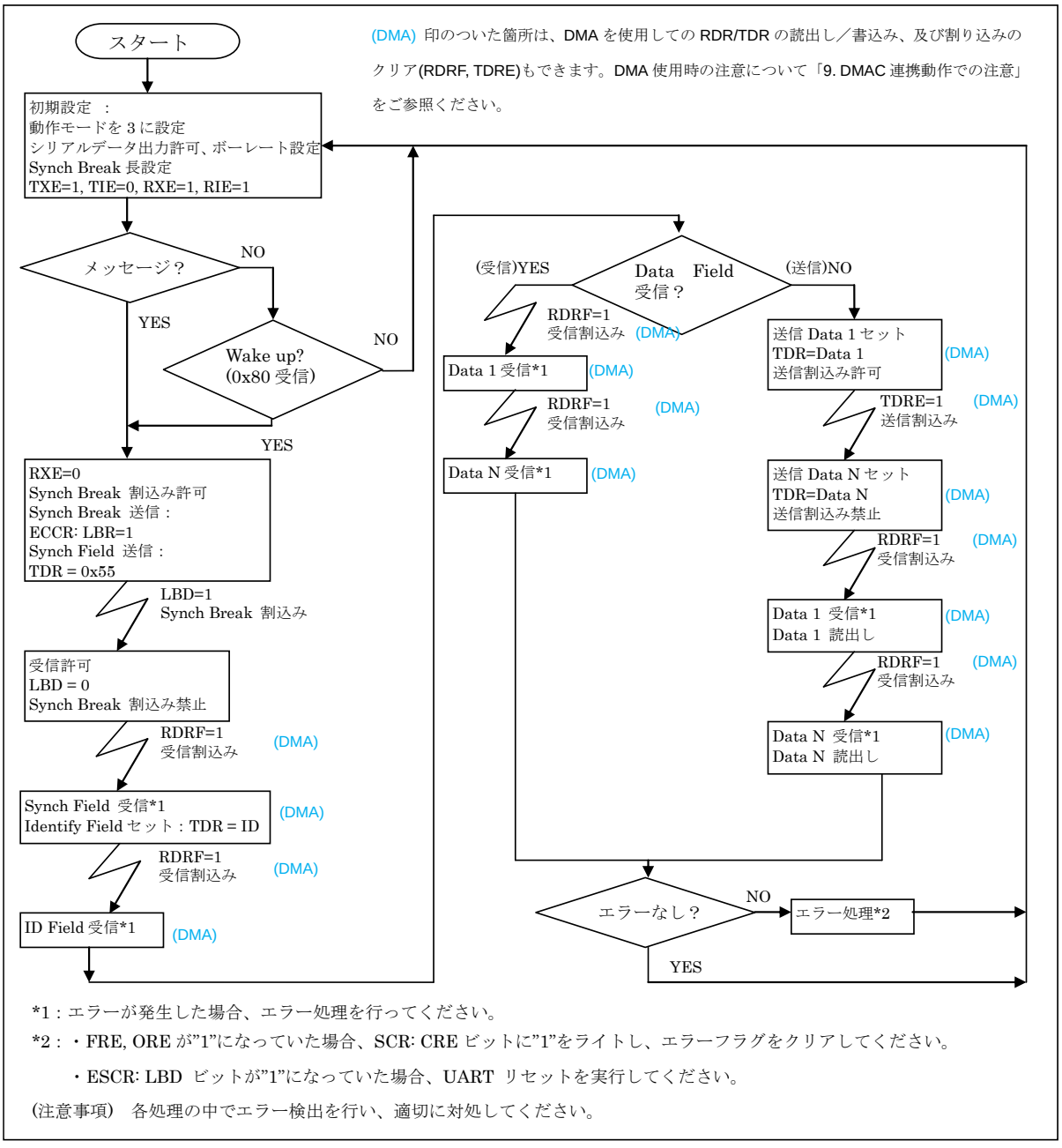
LIN 通信モード (動作モード 3) LIN-UART サンプルフローチャートについて示します。

LIN 通信モードにおける LIN-UART のフローチャート例を示します。

41.7.9.1 マスタデバイスとしての LIN-UART

マスタデバイスとしての LIN-UART について示します。

図 41-28 LIN マスタモードでの LIN-UART フローチャート



*1：エラーが発生した場合、エラー処理を行ってください。

*2：・FRE, ORE が”1”になっていた場合、SCR: CRE ビットに”1”をライトし、エラーフラグをクリアしてください。

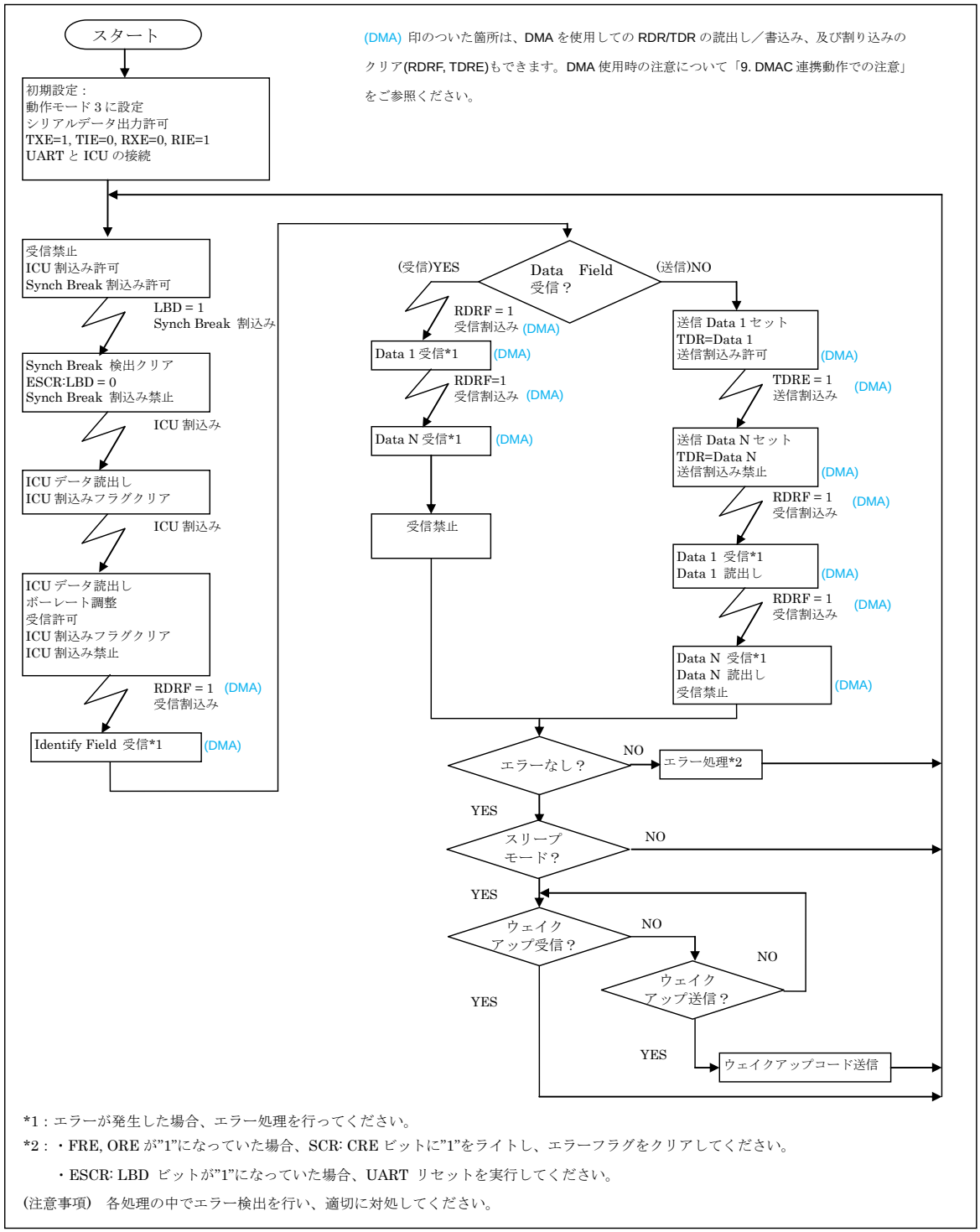
・ESCR: LBD ビットが”1”になっていた場合、UART リセットを実行してください。

(注意事項) 各処理の中でエラー検出を行い、適切に対処してください。

41.7.9.2 スレーブデバイスとしての LIN-UART

スレーブデバイスとしての LIN-UART について示します。

図 41-29 LIN スレーブモードでの LIN-UART フローチャート



*1：エラーが発生した場合、エラー処理を行ってください。

*2：・FRE, ORE が”1”になっていた場合、SCR: CRE ビットに”1”をライトし、エラーフラグをクリアしてください。

・ESCR: LBD ビットが”1”になっていた場合、UART リセットを実行してください。

(注意事項) 各処理の中でエラー検出を行い、適切に対処してください。

41.8 使用上の注意事項

使用上の注意事項について示します。

LIN-UART 使用時における注意事項を示します。

41.8.1 動作許可

動作許可について示します。

LIN-UART には送信，受信のそれぞれに対してシリアルコントロールレジスタ(SCR)に TXE(送信)，RXE(受信)の動作許可ビットがあります。デフォルト(初期値)では、送受信とも動作禁止であるため、転送前に動作を許可する必要があります。また必要に応じて動作禁止にし、転送を中止することもできます。

41.8.2 通信モードの設定

通信モードの設定について示します。

通信モードの設定は、LIN-UART の動作停止中にしてください。送受信中にモード設定をした場合、そのとき送受信されたデータは保証されません。

通信モードの設定後、SMR:UPCL(プログラマブルクリアビット)によってソフトウェアリセットを実行してください。

41.8.3 送信割込み許可のタイミング

送信割込み許可のタイミングについて示します。

送信データエンプティフラグビット(SSR:TDRE)はデフォルト(初期値)が"1"(送信データなし，送信データ書込み許可)であるため、送信割込み要求を許可(SSR:TIE=1)すると直ちに、送信割込み要求が発生します。必ず送信データを設定した後、TIE フラグを"1"にしてください。

41.8.4 動作設定の変更

動作設定の変更について示します。

スタート/ストップビットの付加またはデータフォーマットを変更した場合など、設定を変更した後にはLIN-UARTのリセットを推薦します。

シリアルモードレジスタ(SMR) の設定と同時に LIN-UART のリセット(SMR:UPCL=1)を行っても、正しい動作設定は保証されません。そのため、シリアルモードレジスタ(SMR)のビット設定を行った後、再度 LIN-UART をリセット(SMR:UPCL=1)することを推奨します。

41.8.5 LIN synch break 検出

LIN synch break 検出について示します。

LIN synch break の送信時間はマスタ，スレーブ間の発振精度誤差により変化します。スレーブは、11 シリアルビット長以上の LIN synch break を検出できます。

モード 3(LIN モード)時、シリアル入力が 11 ビット幅以上"0"になると、LIN synch break を検出(ESCR:LBD=1)し、LIN-UART は synch field 待ちになります。

このため、LIN synch break 以外のところでシリアル入力が 11 ビット以上"0"になった場合、LIN-UART は、synch break が入力されたものと認識(LBD=1)し、synch field 待ちになります。この場合、LIN-UART リセット(SMR:UPCL=1)を実行してください。

41.8.6 LIN スレーブ設定

LIN スレーブ設定について示します。

LIN synch break の最小 13 ビット長を確実に検出するために、LIN スレーブを開始するときは、必ず最初の LIN synch break を受信する前にボーレートを設定してください。

41.8.7 プログラムの互換性

プログラムの互換性について示します。

LIN-UART が古い FJ-UART と類似していますが、プログラムの互換性はありません。プログラミングの型は同じときもありますが、レジスタの構造は異なります。さらに、ボーレートの設定はあらかじめ設定された値を選出する代わりにリロード値によって現在決定されます。

41.8.8 アドレス/データ形式選択ビット (SCR:AD)

アドレス/データ形式選択ビットについて示します。

- シリアルコントロールレジスタ(SCR)の AD ビットは、書込み時には送信のアドレス/データ選択設定を行い、読出し時には最後に受信した AD ビットの値を返します。内部的には送信/受信の AD ビット値は個別のレジスタに保存されています。
- リードモディファイライト系命令では、送信された AD ビットデータの値が読み出されます。
- 送信動作時 (TDRE ビットが"0" から"1" になるとき)送信用 AD ビットも送信データレジスタ(TDR) のデータとともに、送信シフトレジスタにロードされますので送信用 AD ビットは送信データレジスタ(TDR) への書込み前に AD ビットを設定してください。

41.8.9 LIN-UART ソフトウェアリセット

LIN-UART ソフトウェアリセットについて示します。

シリアルコントロールレジスタ(SCR)の TXE ビットが"0"のときに、LIN-UART ソフトウェアリセット(SMR:UPCL=1)を実行してください。

41.8.10 LIN Synch Field のインプットキャプチャでの検出について

LIN Synch Field のインプットキャプチャでの検出について示します。

インプットキャプチャの LSYNS0 レジスタの設定が必要になります。『インプットキャプチャ』の章を参照してください。

41.8.11 バスアイドル検出機能

バスアイドル検出機能について示します。

動作モード 2 では受信バスアイドル検出機能は使用できません。また、スレーブモード選択した場合送信バスアイドル検出機能も使用できません。

41.9 DMAC 連携動作での注意

DMAC 連携動作での注意について示します。

LIN-UART 送信，受信の各割込みは DMAC 転送要因に割り当てされ、DMA データ転送機能による送信データの書込み，受信データの読出しが可能です。

41.9.1 送信動作

送信動作について説明します。

LIN-UART 送信動作を開始(SCR:TXE=1)ならびに送信割込み要求許可ビットを有効(SSR:TIE=1)にする前に、送信データレジスタ(TDR)へのダミーライト(任意のデータの書込み)を行ってください。さらに TDR のデータを破棄するために LIN-UART ソフトリセットを発行(SMR:UPCL=1)してください。

以前に行った LIN-UART 転送動作(DMAC 使用しない場合も含む)によって、DMAC への割込み要求が正常に発行できない状態に陥る場合があります。上記動作では、DMAC への割込み要求を発行するための状態復帰をするための操作です。

41.9.2 受信動作

受信動作について説明します。

LIN-UART 受信動作を開始(SCR:RXE=1)ならびに受信割込み許可ビットを有効(SSR:RIE=1)にする前に、受信データレジスタ(RDR)の読出しを行ってください。

LIN-UART 受信時のエラー発生などにより、不要となった受信データが受信データレジスタ(RDR)に残り、この後 DMAC への割込み要求が正常に発行できない状態となります。

LIN-UART ソフトリセット発行(SMR:UPCL=1)により受信データを無効とすることができますが、以後の DMA 転送を正常に行うため受信データレジスタ(RDR)の読出しを行ってください。

42. CAN

CAN について説明します。

42.1 概要

CAN の概要について説明します。

本シリーズは、CAN を 3 チャネル搭載しています。

CAN は、シリアル通信用の標準プロトコルである CAN プロトコル ver2.0A/B に準拠しており、自動車や FA などの工業分野に広く使用されています。

42.2 特長

CAN の特長について説明します。

本シリーズの CAN には以下の特長があります。

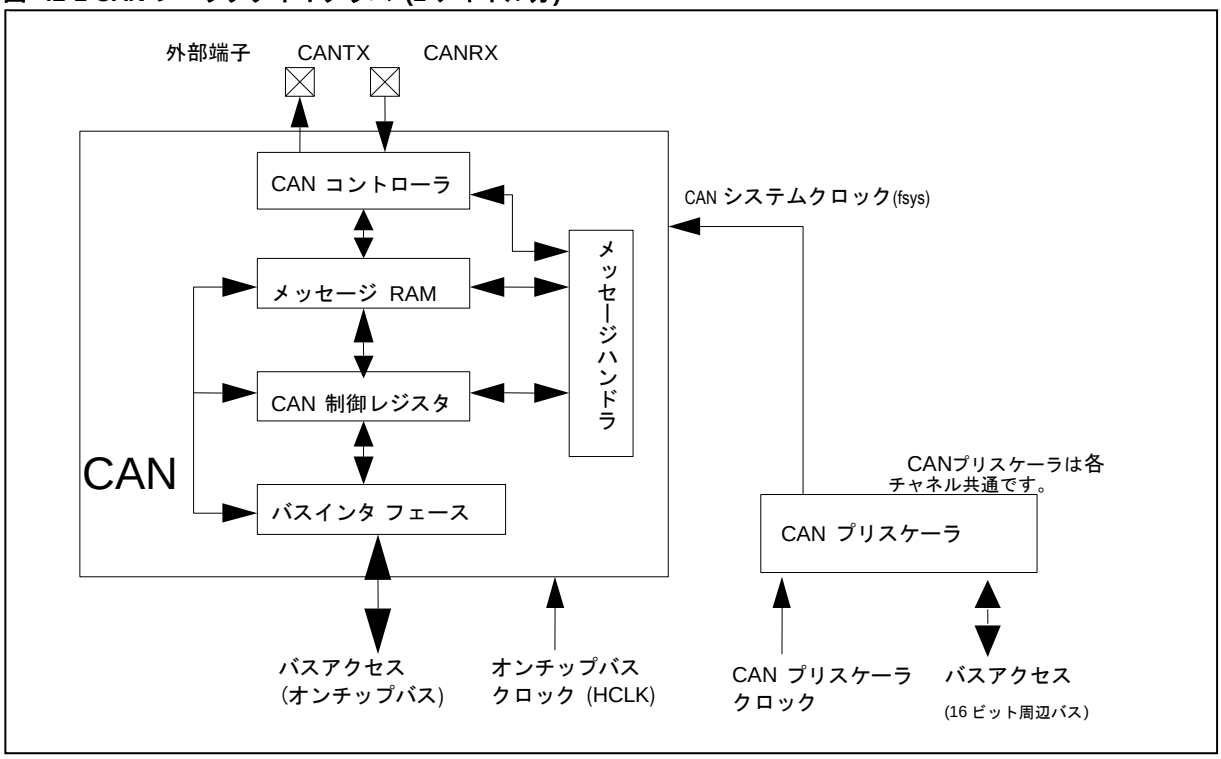
- CAN プロトコル ver2.0A/B をサポート
- 1MBit/s までのビットレートをサポート
- メッセージオブジェクトごとの識別マスク
- プログラマブル FIFO モードをサポート
- マスク可能な割込み
- 自己テスト動作用プログラマブルループバックモードをサポート
- インタフェースレジスタを使用してメッセージバッファへのリード・ライト
- 32/64/128 のメッセージバッファをサポート。型格、チャネルにより異なりますので、『概要』の章を参照してください。

42.3 構成

CAN の構成について説明します。

以下に CAN のブロックダイヤグラムを示します。

図 42-1 CAN ブロックダイヤグラム (1 チャネル分)



CAN コントローラ

CAN プロトコルと送受信メッセージ転送のためのシリアル/パラレル変換用のシリアルレジスタを制御します。

メッセージ RAM

メッセージオブジェクトを格納します。

メッセージハンドラ

メッセージ RAM と CAN コントローラを制御します。

CPU インタフェース

FR 内部バスのインタフェースを制御します。

CAN プリスケーラ

CAN システムクロック(fsys)を生成します。

42.4 レジスタ

CAN のレジスタに示します。

42.4.1 概要

CAN のレジスタ概要について説明します。

CAN には、以下のレジスタがあります。

- CAN 制御レジスタ(CTRLR)
- CAN ステータスレジスタ(STATR)
- CAN エラーカウンタ(ERRCNT)
- CAN ビットタイミングレジスタ(BTR)
- CAN 割込みレジスタ(INTR)
- CAN テストレジスタ(TESTR)
- CAN プリスケーラ拡張レジスタ(BRPER)
- IFx コマンド要求レジスタ(IFxCREQ)
- IFx コマンドマスクレジスタ(IFxCMSK)
- IFx マスクレジスタ 1, 2(IFxMSK1, IFxMSK2)
- IFx アービトレーション 1, 2(IFxARB1, IFxARB2)
- IFx メッセージ制御レジスタ(IFxMCTR)(IFxMCTR)
- IFx データレジスタ A1, A2, B1, B2(IFxDTA1, IFxDTA2, IFxDTB1, IFxDTB2)
- CAN 送信要求レジスタ 1, 2(TREQR1, TREQR2)
- CAN New Data レジスタ 1, 2(NEWDT1, NEWDT2)
- CAN 割込みペンディングレジスタ 1, 2(INTPND1, INTPND2)
- CAN メッセージ有効レジスタ 1, 2(MSGVAL1, MSGVAL2)
- CAN クロックプリスケーラレジスタ(CANPRE)

CAN レジスタは、256 バイト(64 ワード)のアドレス空間が割り当てられ、バイトもしくはワードのアクセスが可能です。メッセージ RAM への CPU のアクセスは、メッセージインタフェースレジスタを介して行います。

■ ベースアドレス (Base_addr)・外部端子一覧

チャネル番号	Base_addr	外部端子名	
		CANTX	CANRX
0	0x2000	TX0	RX0
1	0x2100	TX1	RX1
2	0x2200	TX2	RX2

■ 全体コントロールレジスター覧

表 42-1 全体コントロールレジスター覧

アドレス	レジスタ				備考
	+0	+1	+2	+3	
Base_addr + 00_H	CAN 制御レジスタ (CTRLR)		CAN ステータスレジスタ (STATR)		STAR: BOff, EWarn,Epass =Read Only RxOk,TxOk,LEC= Read/Write
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	予約ビット	CTRLR 参照	予約ビット	STATR 参照	
	Reset: 00_H	Reset: 01_H	Reset: 00_H	Reset: 00_H	
Base_addr + 04_H	CAN エラーカウンタ (ERRCNT)		CAN ビットタイミングレジスタ (BTR)		ERRCNT: Read Only BTR: Init(CTLR)=CCE (CTRLR)="1"時 Write 可能
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	RP, REC[6:0]	TEC[7:0]	TSeg2[2:0], TSeg1[3:0]	SJW[1:0], BRP[5:0]	
	Reset: 00_H	Reset: 00_H	Reset: 23_H	Reset: 01_H	
Base_addr + 08_H	CAN 割込みレジスタ (INTR)		CAN テストレジスタ (TESTR)		INTR: Read Only TESTR: Test(CTRLR)="1" 時、Write 可能. "Rx"は CAN_RX 端子のレベル値を示す。
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	IntId[15:8]	IntId[7:0]	予約ビット	TESTR 参照	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H & 0br0000000	
Base_addr + $0C_H$	CAN プリスケーラ拡張レジスタ (BRPER)		予約ビット		BRPER: CCE(CTLR)="1" 時 Write 可能
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	予約ビット	BRPE[3:0]	-	-	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H	

■ メッセージインタフェースレジスター覧

表 42-2 メッセージインタフェースレジスター覧

アドレス	レジスタ				備考
	+0	+1	+2	+3	
Base_addr + 10_H	IF 1 コマンド要求レジスタ (IF1CREQ)		IF1 コマンドマスクレジスタ (IF1CMSK)		
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	BUSY	Mess. No. [5:0]	予約ビット	IF1CMSK 参照	
	Reset: 00_H	Reset: 01_H	Reset: 00_H	Reset: 00_H	
Base_addr + 14_H	IF1 マスクレジスタ 2 (IF1MSK2)		IF1 マスクレジスタ 1 (IF1MSK1)		
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	MXtd. MDir, Msk[28:24]	Msk[23:16]	Msk[15:8]	Msk[7:0]	
	Reset: FF_H	Reset: FF_H	Reset: FF_H	Reset: FF_H	
Base_addr + 18_H	IF1 アービトレーションレジスタ 2 (IF1ARB2)		IF1 アービトレーションレジスタ 1 (IF1ARB1)		
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	MsgVal, Xtd, Dir,ID[28:24]	ID[23:16]	ID[15:8]	ID[7:0]	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H	
Base_addr + $1C_H$	IF1 メッセージコントロールレジスタ (IF1MCTR)		予約ビット		
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	IF1MCTR 参照	IF1MCTR 参照	-	-	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H	
Base_addr + 20_H	IF1 データ A レジスタ 1 (IF1DTA1)		IF1 データ A レジスタ 2 (IF1DTA2)		バイト配列順序: Big Endian
	bit[7:0]	bit[15:8]	bit[7:0]	bit[15:8]	
	Data[0]	Data[1]	Data[2]	Data[3]	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H	
Base_addr + 24_H	IF1 データ B レジスタ 1 (IF1DTB1)		IF1 データ B レジスタ 2 (IF1DTB2)		バイト配列順序: Big Endian
	bit[7:0]	bit[15:8]	bit[7:0]	bit[15:8]	
	Data[4]	Data[5]	Data[6]	Data[7]	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H	
Base_addr + 30_H	IF1 データ A レジスタ 2 (IF1DTA2)		IF1 データ A レジスタ 1 (IF1DTA1)		バイト配列順序: Little Endian
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	Data[3]	Data[2]	Data[1]	Data[0]	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H	
Base_addr + 34_H	IF1 データ B レジスタ 2 (IF1DTB2)		IF1 データ B レジスタ 1 (IF1DTB1)		バイト配列順序: Little Endian
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	Data[7]	Data[6]	Data[5]	Data[4]	
	Reset: 00_H	Reset: 00_H	Reset: 00_H	Reset: 00_H	

アドレス	レジスタ				備考
	+0	+1	+2	+3	
Base_addr + 40H	IF2 コマンド要求レジスタ (IF2CREQ)		IF2 コマンドマスクレジスタ (IF2CMSK)		
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	BUSY	Mess. No. [5:0]	予約ビット	IF2CMSK 参照	
	Reset: 00H	Reset: 01H	Reset: 00H	Reset: 00H	
Base_addr + 44H	IF2 マスクレジスタ 2 (IF2MSK2)		IF2 マスクレジスタ 1 (IF2MSK1)		
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	MXtd. MDir, Msk[28:24]	Msk[23:16]	Msk[15:8]	Msk[7:0]	
	Reset: FFH	Reset: FFH	Reset: FFH	Reset: FFH	
Base_addr + 48H	IF2 アービトレーションレジスタ 2 (IF2ARB2)		IF2 アービトレーションレジスタ 1 (IF2ARB1)		
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	MsgVal, Xtd, Dir,ID[28:24]	ID[23:16]	ID[15:8]	ID[7:0]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + 4CH	IF2 メッセージコントロールレジスタ (IF2MCTR)		予約ビット		
	bit[15:8]	bit[7:0]	bit[7:0]	bit[15:8]	
	IF2MCTR 参照	IF2MCTR 参照	-	-	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + 50H	IF2 データ A レジスタ 1 (IF2DTA1)		IF2 データ A レジスタ 2 (IF2DTA2)		バイト配列順序: Big Endian
	bit[7:0]	bit[15:8]	bit[7:0]	bit[15:8]	
	Data[0]	Data[1]	Data[2]	Data[3]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + 54H	IF2 データ B レジスタ 1 (IF2DTB1)		IF2 データ B レジスタ 2 (IF2DTB2)		バイト配列順序: Big Endian
	bit[7:0]	bit[15:8]	bit[7:0]	bit[15:8]	
	Data[4]	Data[5]	Data[6]	Data[7]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + 60H	IF2 データ A レジスタ 2 (IF2DTA2)		IF2 データ A レジスタ 1 (IF2DTA1)		バイト配列順序: Little Endian
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	Data[3]	Data[2]	Data[1]	Data[0]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + 64H	IF2 データ B レジスタ 2 (IF2DTB2)		IF2 データ B レジスタ 1 (IF2DTB1)		バイト配列順序: Little Endian
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	Data[7]	Data[6]	Data[5]	Data[4]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	

■ メッセージハンドラレジスター覧

表 42-3 メッセージハンドラレジスター覧

アドレス	レジスタ				備考
	+0	+1	+2	+3	
Base_addr + 80H	CAN 送信要求レジスタ 2 (TREQR2)		CAN 送信要求レジスタ 1 (TREQR1)		INTR1,2: Read Only
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	TxRqst[32:25]	TxRqst[24:17]	TxRqst[16:9]	TxRqst[8:1]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + 84H Base_addr + 88H Base_addr + 8CH	32 メッセージバッファ以上をサポートするための予約領域 (CAN 送信要求レジスタ (TREQR1, TREQR2)参照) TREQ3～TREQ4 : 64 メッセージバッファサポートの場合 TREQ3～TREQ6 : 96 メッセージバッファサポートの場合 TREQ3～TREQ8 : 128 メッセージバッファサポートの場合				
Base_addr + 90H	CAN New Data レジスタ 2 (NEWDT2)		CAN New Data レジスタ 1 (NEWDT1)		NEWDT1,2: Read Only
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	NewDat[32:25]	NewDat[24:17]	NewData[16:9]	NewData[8:1]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + 94H Base_addr + 98H Base_addr + 9CH	32 メッセージバッファ以上をサポートするための予約領域 (CAN データ更新レジスタ (NEWDT1, NEWDT2)参照) NEWDT3～NEWDT4 : 64 メッセージバッファサポートの場合 NEWDT3～NEWDT6 : 96 メッセージバッファサポートの場合 NEWDT3～NEWDT8 : 128 メッセージバッファサポートの場合				
Base_addr + A0H	CAN 割込みペンディングレジスタ 2 (INTPND2)		CAN 割込みペンディングレジスタ 1 (INTPND1)		INTPND1,2: Read Only
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	IntPnd[32:25]	IntPnd[24:17]	IntPnd[16:9]	IntPnd[8:1]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + A4H Base_addr + A8H Base_addr + ACH	32 メッセージバッファ以上をサポートするための予約領域 (CAN 割込みペンディングレジスタ(INTPND1, INTPND2)参照) INTPND3～INTPND4 : 64 メッセージバッファサポートの場合 INTPND3～INTPNT6 : 96 メッセージバッファサポートの場合 INTPND3～INTPND8 : 128 メッセージバッファサポートの場合				
Base_addr + B0H	CAN メッセージ有効レジスタ 2 (MSGVAL2)		CAN メッセージ有効レジスタ 1 (MSGVAL1)		MSGVAL1,2: Read Only
	bit[15:8]	bit[7:0]	bit[15:8]	bit[7:0]	
	MsgVal[32:25]	MsgVal[24:17]	MsgVal[16:9]	MsgVal[8:1]	
	Reset: 00H	Reset: 00H	Reset: 00H	Reset: 00H	
Base_addr + B4H Base_addr + B8H Base_addr + BCH	32 メッセージバッファ以上をサポートするための予約領域 (CAN メッセージ有効レジスタ (MSGVAL1, MSGVAL2)参照) MSGVAL3～MSGVAL4 : 64 メッセージバッファサポートの場合 MSGVAL3～MSGVAL6 : 96 メッセージバッファサポートの場合 MSGVAL3～MSGVAL8 : 128 メッセージバッファサポートの場合				

■ クロックプリスケーラレジスタ

表 42-4 クロックプリスケーラレジスタ

アドレス	レジスタ				備考
	+0	+1	+2	+3	
00_04A4$_H$	CANPRE	-	-	-	CAN Prescaler
	bit[3:0]	-	-	-	
	CANPRE[3:0]	-	-	-	
	Reset: 00$_H$	-	-	-	

■ 全体コントロールレジスタ

- CAN 制御レジスタ(CTRLR)
- CAN ステータスレジスタ(STATR)
- CAN エラーカウンタ(ERRCNT)
- CAN ビットタイミングレジスタ(BTR)
- CAN 割込みレジスタ(INTR)
- CAN テストレジスタ(TESTR)
- CAN プリスケーラ拡張レジスタ(BRPER)

■ メッセージインタフェースレジスタ

- IFx コマンド要求レジスタ(IFxCREQ)
- IFx コマンドマスクレジスタ(IFxCMSK)
- IFx マスクレジスタ 1, 2(IFxMSK1, IFxMSK2)
- IFx アービトレーションレジスタ 1, 2(IFxARB1, IFxARB2)
- IFx メッセージ制御レジスタ(IFxMCTR)
- IFx データレジスタ A1, A2, B1, B2(IFxDTA1, IFxDTA2, IFxDTB1, IFxDTB2)

■ メッセージハンドラレジスタ

- CAN 送信要求レジスタ 1, 2(TREQR1, TREQR2)
- CAN データ更新レジスタ 1, 2(NEWDT1, NEWDT2)
- CAN 割込みペンディングレジスタ 1, 2(INTPND1, INTPND2)
- CAN メッセージ有効レジスタ 1, 2(MSGVAL1, MSGVAL2)

■ プリスケーラレジスタ

- CAN クロックプリスケーラレジスタ(CANPRE)

42.4.2 全体コントロールレジスタ

全体コントロールレジスタについて示します。

全体コントロールレジスタは、CAN プロトコル制御および動作モードを制御し、ステータス情報を提供します。

- CAN 制御レジスタ(CTRLR)
- CAN ステータスレジスタ(STATR)
- CAN エラーカウンタ(ERRCNT)
- CAN ビットタイミングレジスタ(BTR)
- CAN 割込みレジスタ(INTR)
- CAN テストレジスタ(TESTR)
- CAN プリスケーラ拡張レジスタ(BRPER)

42.4.2.1 CAN 制御レジスタ (CTRLR)

CAN 制御レジスタのビット構成について説明します。

CAN コントローラの動作モードを制御します。

■ CAN 制御レジスタ(上位バイト) : アドレス Base_addr+00H (アクセス:バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

■ CAN 制御レジスタ(下位バイト) : アドレス Base_addr+01H (アクセス:バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	Test	CCE	DAR	予約	EIE	SIE	IE	Init
初期値	0	0	0	0	0	0	0	1
属性	R/W	R/W	R/W	R0,W0	R/W	R/W	R/W	R/W

[bit15～bit8] 予約
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit7] Test : テストモード許可ビット

Test	機能
0	通常動作 [初期値]
1	テストモード

＜注意事項＞
Test ビットに"1"を設定する場合、Iint ビットが"1"のときに設定してください。

[bit6] CCE : ビットタイミングレジスタ書込み許可ビット

CCE	機能
0	CAN ビットタイミングレジスタ(BTR)および CAN プリスケーラ拡張レジスタ(BRPER)への書込みを禁止します [初期値]
1	CAN ビットタイミングレジスタ(BTR)および CAN プリスケーラ拡張レジスタ(BRPER)への書込みを許可します。Init ビットが"1"のときに有効です。

[bit5] DAR : 自動再送禁止ビット

DAR	機能
0	調停負けまたはエラー検出時のメッセージの自動再送を許可します [初期値]
1	自動再送を禁止します

CAN コントローラは調停負けまたは転送中のエラー検出によりフレームの自動再送を行います。自動再送する場合は、DAR ビットに"0"を設定します。CAN を Time Triggered CAN 環境で動作させるためには、DAR ビットに"1"を設定する必要があります。

＜注意事項＞

DAR ビットに"1"を設定した場合は、メッセージオブジェクト(メッセージオブジェクトについては「42.4.4 メッセージオブジェクト」を参照してください)の TxRqst ビットと NewDat ビットの値は以下のようになります。

- フレーム送信を開始したとき、メッセージオブジェクトの TxRqst ビットが"0"にクリアされますが、NewDat ビットは 1"にセットされたままです。
- フレーム送信が正常終了すると NewDat ビットは"0"にクリアされます。
  送信が調停負けもしくはエラー検出すると、NewDat ビットは"1"にセットされたままです。送信を再開するためには、TxRqst ビットに"1"を設定してください。
- フレーム送出中(TxRqst=1) に CAN 制御レジスタ(CTRLR)の DAR ビットを"0"から"1"に変更すると送出中のフレームがもう一度送出されますので DAR ビットを変更する場合、INIT ビットが"1" のときに変更してください。
- DAR ビットに"1"を設定して、複数のメッセージバッファによる送信を行った場合、以下の動作になります。
- CAN がフレーム送信開始前および送信中に * 他の* メッセージバッファの TxRqst に"1"が設定された場合(複数のメッセージバッファの TxRqst に"1" が設定された場合)、フレーム送信の開始時、設定されたすべての TxRqst が"0"にリセットされ、その中で優先順位の高いメッセージバッファのデータが送信されます。
- フレーム送信が正常終了すると送信されたメッセージバッファの NewDat が"0"にリセットされ、そのときメッセージバッファの TxIE が"1"であれば、メッセージオブジェクトの IntPnd は"1" にセットされます。
- ほかのメッセージバッファはフレーム送信開始時、TxRqst が"0"にリセットされたためにフレーム送信は行われません。NewDat または IntPnd によって送信されたメッセージバッファを確認後、送信したいメッセージバッファに対し、再度 TxRqst と NewDat に"1"をセットする必要があります。

[bit4] 予約

常に"0"が読み出せます。書込みは"0"を設定してください。

[bit3] EIE : エラー割込みコード許可ビット

EIE	機能
0	CAN ステータスレジスタ(STATR)の Boff ビットまたは EWarn ビットの変化により、CAN 割込みレジスタ(INTR)への割込みコードの設定を禁止します。 [初期値]
1	CAN ステータスレジスタ(STATR)の Boff ビットまたは EWarn ビットの変化により、CAN 割込みレジスタ(INTR)へのステータス割込みコードの設定を許可します。

[bit2] SIE : ステータス割込みコード許可ビット

SIE	機能
0	CAN ステータスレジスタ(STATR)の TxOk, RxOk または LEC ビットの変化により、CAN 割込みレジスタ(INTR)への割込みコードの設定を禁止します。 [初期値]
1	CAN ステータスレジスタ(STATR)の TxOk, RxOk または LEC ビットの変化により、CAN 割込みレジスタ(INTR)へのステータス割込みコードの設定を許可します。CPU からの書込みによって発生した TxOk, RxOk, LEC ビットの変化は CAN 割込みレジスタ(INTR)には設定されません。

[bit1] IE : 割込み許可ビット

IE	機能
0	割込みの発生を禁止します。 [初期値]
1	割込みの発生を許可します。

[bit0] Init : 初期化ビット

Init	機能
0	CAN コントローラの初期化解除後に動作します。
1	CAN コントローラを初期化し、動作を停止します。 [初期値]

＜注意事項＞

- バスオフリカバリシーケンスは、Init ビットの設定/解除によって短縮はできません。デバイスがバスオフになると、CAN コントローラ自身が Init ビットを"1"にセットし、すべてのバス動作を停止します。バスオフ状態から Init ビットを"0"にクリアすると、バスアイドルが連続的に 129 回(11 ビットのレセッシブを 1 回とする)発生するまでバス動作を停止状態にします。バスオフリカバリシーケンス実行後、エラーカウンタをリセットします。
- バスオフリカバリシーケンス中に Init ビットに"1" を設定し、再度 Init ビットに"0"を設定した場合、バスオフリカバリシーケンスを最初から実施します(11 ビットのレセッシブを 1 回として 129 回再度実施)。
- CAN ビットタイミングレジスタ(BTR)を設定する場合は、Init および CCE ビットに"1"を設定してください。
- 送受信途中で Init ビットに"1"を設定した場合、即送受信を中止します。
- 低消費電力モード(停止モード、クロックモード)へ遷移する前、および供給クロックを変更する前に Init ビットに"1"を書込んで CAN コントローラを初期状態にしなければなりません。
- 次のレジスタにより、CAN インタフェースに供給するクロックの分周比を変更する場合は、Init ビットに"1"を設定し、CAN コントローラを停止した状態で行ってください。
  - ☐ CAN ビットタイミングレジスタ(BTR)
  - ☐ CAN プリスケーラ拡張レジスタ(BRPER)
  - ☐ CAN プリスケーラレジスタ(CANPRE)
- 送信中、Init ビットに"1"を設定する場合、送信完了後に Init ビットに"1"を設定してください。もし、送信中、Init ビットに"1"を設定した場合、Init ビットに"0"を設定してから 2 ビット時間経過後に送信の設定(TxRqst="1")を行ってください。

42.4.2.2 CAN ステータスレジスタ (STATR)

CAN ステータスレジスタのビット構成について説明します。

CAN ステータスおよび CAN バス状態を表示します。

■ CAN ステータスレジスタ(上位バイト) : アドレス Base_addr+02H(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

■ CAN ステータスレジスタ(下位バイト) : アドレス Base_addr+03H(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BOff	EWarn	EPass	RxOk	TxOk	LEC[2:0]		
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,W	R,W	R,WX	R,WX	R,WX

[bit15 ～bit8] 予約
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit7] BOff : バスオフビット

BOff	機能
0	CAN コントローラはバスオフ状態でないことを示します。 [初期値]
1	CAN コントローラはバスオフ状態であることを示します。

[bit6] EWarm : ワーニングビット

EWarn	機能
0	送信と受信カウンタが共に 96 未満であることを示します [初期値]
1	送信または受信カウンタが 96 以上であることを示します。

[bit5] EPass : エラーパッシブビット

EPass	機能
0	送信と受信カウンタが共に 128 未満(エラーアクティブ状態)であることを示します。 [初期値]
1	受信カウンタは RP ビット="1"、送信カウンタが 128 以上(エラーパッシブ状態)であることを示します。

[bit4] RxOk : メッセージ正常受信ビット

RxOk	機能
0	CAN バス上で正常なメッセージ通信が行われていない またはバスアイドル状態であることを示します。 [初期値]
1	CAN バス上で正常なメッセージ通信が行われたことを示します。

[bit3] TxOk : メッセージ正常送信ビット

TxOk	機能
0	バスアイドル状態もしくは正常なメッセージ送信が行われていないことを示します。 [初期値]
1	正常なメッセージ送信が行われたことを示します。

＜注意事項＞

RxOk および TxOk ビットは、"0"書込みでのみクリアされます。

[bit2～bit0]LEC[2:0]: ラストエラーコードビット

LEC[2:0]	状態	機能
000	正常	正常に送信または受信されたことを示します。 [初期値]
001	Stuff エラー	メッセージ内において6ビット以上連続してドミナントまたはレセッシブを検出したことを示します
010	Form エラー	受信されたフレームの固定フォーマット部が間違って検出したことを示します。
011	Ack エラー	送信メッセージに対し、ほかのノードからアクノリッジされなかったことを示します。
100	Bit1 エラー	調停フィールドを除くメッセージの送信データにおいて、レセッシブを送信したにも関らずドミナントを検出したことを示します。
101	Bit0 エラー	メッセージの送信データにおいて、ドミナントを送信したにも関らずレセッシブを検出したことを示します。 バスリカバリ中には、11 ビットのレセッシブを検出するごとにセットされます。このビットを読み出すことによりバスリカバリシーケンスを監視できます。
110	CRC エラー	受信したメッセージの CRC データと計算した CRC の結果が一致しなかったことを示します。
111	未検出	LEC ビットに"111_B"を設定した後、LEC 値が"111_B"を読み出した場合、その期間は送受信しなかったことを示します。(バスアイドル状態)

LEC ビットは、CAN バス上で発生した最後のエラーを示すコードを保持します。メッセージがエラーなしで転送(受信/送信)完了すると"0"にクリアされます。未検出コード"111_B"は、コード更新をチェックするために使用することができます。

＜注意事項＞

- ステータス割込みコード(8000_H)は、EIE ビットが"1"のときに BOff または EWarn ビットが変化した場合、もしくは SIE ビットが"1"のときに RxOk, TxOk または LEC ビットが変化した場合、CAN 割込みレジスタ(INTR)に設定されます。
- RxOk, TxOk ビットはフラグの値がプログラムによる書込みによって更新されますので、CAN コントローラによってセットされた RxOk, TxOk ビットの値が変更されてしまいます。もし RxOk, TxOk ビットを使用する場合には、RxOk または TxOk ビットが"1"にセットされてから(45×BT)時間以内にクリアしてください。BT は 1 ビットタイムを示します。
- SIE ビットが"1"のとき、LEC ビットの変化による割込みが発生した場合には CAN ステータスレジスタ(STATR)に書込まないでください。
- EPass ビットの変化、RxOk, TxOk および LEC ビットへの書込み動作では CAN 割込みレジスタ(INTR)のエラーコード割込みのセットは行われません。
- BOff ビットが"1" になった場合、EPass ビット、EWarn ビットは"1"の状態になっています。また、EPass ビットが"1" になった場合、EWarn ビットは"1"の状態になっています。
- CAN ステータスレジスタ(STATR)を読み出すことにより、CAN 割込みレジスタ(INTR)のステータス割込み(8000_H)はクリアされます。

42.4.2.3 CAN エラーカウンタ (ERRCNT)

CAN エラーカウンタのビット構成について説明します。

受信エラーパッシブ表示および受信エラーカウンタ、送信エラーカウンタを示します。

■ CAN エラーカウンタレジスタ(上位バイト) : アドレス Base_addr+04H(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	RP	REC[6:0]						
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN エラーカウンタレジスタ(下位バイト) : アドレス Base_addr+05H(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TEC[7:0]							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit15] RP : 受信エラーパッシブ表示

RP	機能
0	受信エラーカウンタはエラーパッシブ状態でないことを示します。[初期値]
1	受信エラーカウンタは CAN 仕様で定義されているエラーパッシブ状態に到達したことを示します。

[bit14～bit8] REC[6:0] : 受信エラーカウンタ

受信エラーカウンタ値。受信エラーカウンタ値の範囲は 0～127 です。

受信エラーカウンタが 128 以上になる場合、RP ビットに"1"を設定し、受信エラーカウンタは更新されません。

例)

REC[6:0]=127、 受信エラーで+8 される場合、その結果は、RP=1, REC[6:0]=127

REC[6:0]=126、 受信エラーで+8 される場合、その結果は、RP=1, REC[6:0]=126

REC[6:0]=119、 受信エラーで+8 される場合、その結果は、RP=0, REC[6:0]=127

[bit7～bit0] TEC[7:0] : **送信エラーカウンタ**

送信エラーカウンタ値。送信エラーカウンタ値の範囲は 0～255 です。
送信エラーカウンタが 256 以上になる場合、CAN 制御レジスタの Init ビットに"1"を設定し、送信エラーカウンタは更新されません。

例)
TEC[7:0]=255、 送信エラーで+8 される場合、その結果は、Init=1, TEC[7:0]=255
TEC[7:0]=254、 送信エラーで+8 される場合、その結果は、Init=1, TEC[7:0]=254
TEC[7:0]=247、 送信エラーで+8 される場合、その結果は、Init=0, TEC[7:0]=255

42.4.2.4 CAN ビットタイミングレジスタ (BTR)

CAN ビットタイミングレジスタのビット構成について説明します。

プリスケーラおよびビットタイミングを設定します。

■ CAN ビットタイミングレジスタ(上位バイト) : アドレス Base_addr+06H(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	TSeg2			TSeg1			
初期値	0	0	1	0	0	0	1	1
属性	R0,W0	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ CAN ビットタイミングレジスタ(下位バイト) : アドレス Base_addr+07H(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	SJW		BRP					
初期値	0	0	0	0	0	0	0	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15] 予約
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit14～bit12] TSeg2 : タイムセグメント 2 設定ビット
有効設定値は 0～7 です。TSeg2+1 ビットの値がタイムセグメント 2 になります。
タイムセグメント 2 は、CAN 仕様のフェーズバッファセグメント(PHASE_SEG2)に相当します。

[bit11～bit8] TSeg1: タイムセグメント 1 設定ビット
有効設定値は 1～15 です。0 の設定は禁止です。TSeg1+1 ビットの値がタイムセグメント 1 になります。
タイムセグメント 1 は、CAN 仕様のプロパゲーションセグメント(PROP_SEG)＋フェーズバッファセグメント 1(PHASE_SEG1)に相当します。

[bit7, bit6] SJW : 再同期化ジャンプ幅設定ビット
有効設定値は 0～3 です。SJW+1 ビットの値が再同期ジャンプ幅となります。

[bit5～bit0] BRP : ボーレートプリスケーラ設定ビット
有効設定値は 0～63 です。BRP+1 ビットの値がボーレートプリスケーラになります。
システムクロック(fsys)を分周して、CAN コントローラの基本単位時間(tq)を決定します。

＜注意事項＞
CAN 制御レジスタ(CTRLR)の CCE ビットと Init ビットが"1"にセットされているときに、CAN ビットタイミングレジスタ(BTR)および CAN プリスケーラ拡張レジスタ(BRPER)を設定してください。

42.4.2.5 CAN 割込みレジスタ (INTR)

CAN 割込みレジスタのビット構成について説明します。

メッセージ割込みコードおよびステータス割込みコードが確認できます。

■ CAN 割込みレジスタ (上位バイト) : アドレス Base_addr+08H (アクセス : ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	IntId15～IntId8							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN 割込みレジスタ(下位バイト) : アドレス Base_addr+09H (アクセス : ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IntId7～IntId0							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

IntId	機能
0000_H	割込みなし
(32msg の場合) 0001_H ～ 0020_H (64msg の場合) 0001_H ～ 0040_H (128msg の場合) 0001_H ～ 0080_H	割込み要因はメッセージオブジェクトの番号を示します (メッセージ割込みコード)
(32msg の場合) 0021_H ～ $7FFF_H$ (64msg の場合) 0041_H ～ $7FFF_H$ (128msg の場合) 0081_H ～ $7FFF_H$	未使用
8000_H	CAN ステータスレジスタ(STATR)の変化による割込みを示します (ステータス割込みコード)
8001_H ～ $FFFF_H$	未使用

複数の割込みコードが保留中である場合、CAN 割込みレジスタ(INTR)は優先度の高い割込みコードを示します。割込みコードが CAN 割込みレジスタ(INTR)にセットされていても優先度の高い割込みコードが発生した場合には、CAN 割込みレジスタ(INTR)は優先度の高い割込みコードに更新されます。

優先度の高い割込みコードは、ステータス割込みコード(8000_H)、メッセージ割込み(0001_H, 0002_H, 0003_H, ……, 0020_H) の順になります。(32msg の場合。64, 128msg も同様)

IntId[15:0]ビットが 0000_H 以外で、CAN 制御レジスタ(CTRLR)の IE ビットが"1"にセットされると、CPU への割込み信号がアクティブになります。IntId[15:0]ビットの値が 0000_H になる(割込み要因がリセットされる)もしくは CAN 制御レジスタ(CTRLR)の IE ビットが"0"にリセットされると、割込み信号はインアクティブになります。

対象となるメッセージオブジェクト(メッセージオブジェクトについては「42.4.4 メッセージオブジェクト」を参照してください)の IntPnd ビットを"0"にクリアすることでメッセージ割込みコードはクリアされます。

ステータス割込みコードは CAN ステータスレジスタ(STATR)を読み出すことでクリアされます。

42.4.2.6 CAN テストレジスタ (TESTR)

CAN テストレジスタのビット構成について説明します。

テストモードの設定および RX 端子のモニタを行います。動作については、「42.5.7 テストモード」を参照してください。

■ CAN テストレジスタ(上位バイト) : アドレス Base_addr+0A$_H$ (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

■ CAN テストレジスタ(下位バイト) : アドレス Base_addr+0B$_H$ (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	Rx	Tx1	Tx0	LBack	Silent	Basic	予約	予約
初期値	r	0	0	0	0	0	0	0
属性	R,WX	R/W	R/W	R/W	R/W	R/W	R0,W0	R0,W0

bit7 の Rx の初期値 (r) は、CAN バス上のレベルが表示されます。

[bit15～bit8] 予約
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit7] Rx : Rx 端子モニタビット

Rx	機能
0	CAN バスはドミナントであることを示します。
1	CAN バスはレセッシブであることを示します。

[bit6, bit5] Tx1, Tx0 : TX 端子コントロールビット

Tx1, Tx0	機能
00	通常動作 [初期値]
01	サンプリングポイントが TX 端子に出力されます。
10	TX 端子にドミナントを出力します。
11	TX 端子にレセッシブを出力します。

[bit4] LBack : ループバックモード

LBack	機能
0	ループバックモードを禁止します。 [初期値]
1	ループバックモードを許可します。

[bit3] Silent : サイレントモード

Silent	機能
0	サイレントモードを禁止します。 [初期値]
1	サイレントモードを許可します。

[bit2] Basic : ベーシックモード

Basic	機能
0	ベーシックモードを禁止します。 [初期値]
1	ベーシックモードを許可します。 IF1 レジスタは送信メッセージとして、IF2 レジスタは受信メッセージとして使用されます。

[bit1, bit0] 予約

常に"0"が読み出されます。書込みは"0"を設定してください。

＜注意事項＞

- CAN制御レジスタ(CTRLR)のTestビットを"1"にセットした後、レジスタへ書き込んでください。テストモードが有効となるのは、CAN 制御レジスタ(CTRLR)の Test ビットが"1"のときです。途中で CAN 制御レジスタ(CTRLR)の Test ビットを"0"にするとテストモードから通常モードになります。
- Tx ビットを"00"以外に設定した場合、メッセージを送信できません。

42.4.2.7 CAN プリスケーラ拡張レジスタ (BRPER)

CAN プリスケーラ拡張レジスタのビット構成について説明します。

CAN ビットタイミングで設定したプリスケーラと組み合わせることにより、CAN コントローラで使用するプリスケーラを拡張します。

■ CAN プリスケーラ拡張レジスタ(上位バイト) : アドレス Base_addr+0C$_H$(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

■ CAN プリスケーラ拡張レジスタ(下位バイト) : アドレス Base_addr+0D$_H$(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	予約	予約	予約	BRPE			
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W	R/W

[bit15～bit4] 予約
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit3～bit0] BRPE : ボーレートプリスケーラ拡張ビット
CAN ビットタイミングレジスタ(BTR)の BRP ビットと BRPE ビットを組み合わせることにより、1023 までボーレートプリスケーラを拡張できます。

{BRPE(MSB: 4 ビット),BRP(LSB: 6 ビット)}＋1 の値が CAN コントローラのプリスケーラとなります。

42.4.3 メッセージインタフェースレジスタ

メッセージインタフェースレジスタについて説明します。

CPU からメッセージ RAM へのアクセスを制御するために 2 組のメッセージインタフェースレジスタを提供します。

メッセージ RAM への CPU アクセスを制御するために使用される 2 組のメッセージインタフェースレジスタがあります。この 2 組のレジスタは、転送された(する)データ(メッセージオブジェクト)をバッファすることで、メッセージ RAM への CPU アクセスと CAN コントローラからのアクセスの競合を回避します。メッセージオブジェクト(メッセージオブジェクトについては、「42.4.4 メッセージオブジェクト」を参照してください)は、メッセージインタフェースレジスタとメッセージ RAM 間を一度に転送します。

テストベーシックモードを除き、2 組のメッセージインタフェースレジスタの機能は同一で、独立して動作可能です。例えば、IF1 のメッセージインタフェースレジスタをメッセージ RAM への書込み動作中に、IF2 のメッセージインタフェースレジスタをメッセージ RAM からの読出しに使用することも可能です。表 4-5 に 2 組のメッセージインタフェースレジスタを示します。

メッセージインタフェースレジスタは、コマンドレジスタ(コマンド要求、コマンドマスクレジスタ)と、このコマンドレジスタによって制御されるメッセージバッファレジスタ(マスク、アービトレーション、メッセージ制御、データレジスタ)から構成されます。コマンドマスクレジスタは、データ転送の方向とメッセージオブジェクトのどの部分が転送されるのかを示します。コマンド要求レジスタは、メッセージ番号の選択と、コマンドマスクレジスタに設定された動作を行います。

表 42-5 IF1、IF2 メッセージインタフェースレジスタ

アドレス	IF1 レジスタセット	アドレス	IF2 レジスタセット
Base_addr+ 10_H	IF1 コマンド要求	Base_addr+ 40_H	IF2 コマンド要求
Base_addr+ 12_H	IF1 コマンドマスク	Base_addr+ 42_H	IF2 コマンドマスク
Base_addr+ 14_H	IF1 マスク 2	Base_addr+ 44_H	IF2 マスク 2
Base_addr+ 16_H	IF1 マスク 1	Base_addr+ 46_H	IF2 マスク 1
Base_addr+ 18_H	IF1 アービトレーション 2	Base_addr+ 48_H	IF2 アービトレーション 2
Base_addr+ $1A_H$	IF1 アービトレーション 1	Base_addr+ $4A_H$	IF2 アービトレーション 1
Base_addr+ $1C_H$	IF1 メッセージ制御	Base_addr+ $4C_H$	IF2 メッセージ制御
Base_addr+ 20_H	IF1 データ A1	Base_addr+ 50_H	IF2 データ A1
Base_addr+ 22_H	IF1 データ A2	Base_addr+ 52_H	IF2 データ A2
Base_addr+ 24_H	IF1 データ B1	Base_addr+ 54_H	IF2 データ B1
Base_addr+ 26_H	IF1 データ B2	Base_addr+ 56_H	IF2 データ B2

42.4.3.1 IFx コマンド要求レジスタ (IFxCREQ)

IFx コマンド要求レジスタのビット構成について説明します。

メッセージ RAM のメッセージ番号の選択とメッセージ RAM とメッセージバッファレジスタ間の転送を行います。また、テストのベーシックモードでは、IF1 を送信制御用に IF2 を受信制御用として使用します。

■ IFx コマンド要求レジスタ(上位バイト) : アドレス Base_addr+10H & Base_addr+40H (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	BUSY	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R/W	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

■ IFx コマンド要求レジスタ(下位バイト) : アドレス Base_addr+11H & Base_addr+41H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	Message Number							
初期値	0	0	0	0	0	0	0	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IFx コマンド要求レジスタ(IFxCREQ)へメッセージ番号を書き込むとすぐにメッセージ RAM とメッセージバッファレジスタ(マスク、アービトレーション、メッセージ制御、データレジスタ)とのメッセージ転送が開始されます。この書込み動作で、BUSY ビットが"1"にセットされ転送処理中であることを示します。その転送が終了すると、BUSY ビットが"0"にリセットされます。

BUSY ビットが"1"のときに、CPU からメッセージインタフェースレジスタへアクセスが発生すると、BUSY ビットが"0"になるまで(コマンド要求レジスタ書込み後、クロックで 3～6 サイクル期間)、CPU をウェイトさせます。

テストのベーシックモードでは、BUSY ビットの使用方法が異なります。IF1 コマンド要求レジスタは、送信メッセージとして使用され、BUSY ビットに"1"をセットすることによりメッセージ送信開始を指示します。メッセージ転送が正常終了すると、BUSY ビットは"0"にリセットされます。また、BUSY ビットを"0"にリセットすることにより、いつでもメッセージ転送を中断させることが可能です。

IF2 コマンド要求レジスタは、受信メッセージとして使用され、BUSY ビットを"1"にセットすることにより、受信したメッセージを IF2 メッセージインタフェースレジスタに格納します。

[bit15] BUSY : **ビジーフラグビット**
(1) テストベーシックモード以外

BUSY	機能
0	メッセージインタフェースレジスタとメッセージ RAM 間でデータ転送処理を行っていないことを示します。[初期値]
1	メッセージインタフェースレジスタとメッセージ RAM 間でデータ転送処理中であることを示します。

(2) テストベーシックモード

IF1 コマンド要求レジスタ

BUSY	機能
0	メッセージ送信を禁止します。
1	メッセージ送信を許可します。

IF2 コマンド要求レジスタ

BUSY	機能
0	メッセージ受信を禁止します。
1	メッセージ受信を許可します。

[bit14～bit8] **予約**
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit7～bit0] Message Number : **メッセージ番号** (32 message buffer CAN 用)

Message Number	機能
00_H	設定禁止です。設定した場合、20_H として解釈され、20_H が読み出されます。
01_H ～ 20_H	処理を行うメッセージ番号を設定します。
21_H ～ $3F_H$	設定禁止です。設定した場合、01_H～$1F_H$ として解釈され、解釈された値が読み出されます。

[bit7～bit0] Message Number : **メッセージ番号** (64 message buffer CAN 用)

Message Number	機能
00_H	設定禁止です。設定した場合、40_H として解釈され、40_H が読み出されます。
01_H ～ 40_H	処理を行うメッセージ番号を設定します。
41_H ～ FF_H	設定禁止です。設定した場合、01_H～$3F_H$ として解釈され、解釈された値が読み出されます。

[bit7～bit0] Message Number : **メッセージ番号** (128 message buffer CAN 用)

Message Number	機能
00_H	設定禁止です。設定した場合、80_H として解釈され、80_H が読み出されます。
01_H ～ 80_H	処理を行うメッセージ番号を設定します。
81_H ～ FF_H	設定禁止です。設定した場合、01_H～$7F_H$ として解釈され、解釈された値が読み出されます。

＜注意事項＞

BUSY ビットは、リード/ライトが可能です。テストのベーシックモード時以外は、このビットに何を書いても動作に影響しません。(ベーシックモードについては、「42.5.7 テストモード」を参照してください)

42.4.3.2 IFx コマンドマスクレジスタ (IFxCMSK)

IFx コマンドマスクレジスタのビット構成について説明します。

メッセージインタフェースレジスタとメッセージ RAM 間の転送方向を制御し、どのデータを更新するかを設定します。また、テストのベーシックモードでレジスタは無効となります。

■ IFx コマンドマスクレジスタ(上位バイト) : アドレス Base_addr+12H & Base_addr+42H(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	予約	予約	予約	予約	予約	予約	予約
初期値	0	0	0	0	0	0	0	0
属性	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0	R0,W0

■ IFx コマンドマスクレジスタ(下位バイト) : アドレス Base_addr+13H & Base_addr+43H(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	WR/RD	Mask	Arb	Control	CIP	TxRqst/ NewDat	Data A	Data B
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit8] 予約
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit7] WR/RD : ライト/リード制御ビット

WR/RD	機能
0	メッセージ RAM からデータを読み出すことを示します。メッセージ RAM からの読出しは IFx コマンド要求レジスタ(IFxCREQ)への書込みによって実行されます。メッセージ RAM から読み出すデータは、Mask, Arb, Control, CIP, TxRqst/NewDat, Data A, Data B ビットの設定に依存します。 [初期値]
1	メッセージ RAM へデータを書き込むことを示します。メッセージ RAM への書き込みは IFx コマンド要求レジスタ(IFxCREQ)への書き込みによって実行されます。メッセージ RAM への書き込みデータは、Mask, Arb, Control, CIP,TxRqst/NewDat, Data A, Data B ビットの設定に依存します。

＜注意事項＞
リセット後、メッセージ RAM のデータは不定です。メッセージ RAM のデータが不定状態でメッセージ RAM のデータを読み出すことは禁止です。

IFx コマンドマスクレジスタ(IFxCMSK)の bit6～bit0 は、転送方向(WR/RD ビット)の設定により異なった意味になります。

(1) 転送方向がライトの場合(WR/RD="1")

[bit6] Mask : マスクデータ更新ビット

Mask	機能
0	メッセージオブジェクト*1 のマスクデータ(ID マスク + MDir + MXtd)を更新しないことを示します。 [初期値]
1	メッセージオブジェクト*1 のマスクデータ(ID マスク + MDir + MXtd)を更新することを示します。

[bit5] Arb : アービトレーションデータ更新ビット

Arb	機能
0	メッセージオブジェクト*1 のアービトレーションデータ(ID + Dir + Xtd + MsgVal)を更新しないことを示します。 [初期値]
1	メッセージオブジェクト*1 のアービトレーションデータ(ID + Dir + Xtd + MsgVal)を更新することを示します。

[bit4] Control : 制御データ更新ビット

Control	機能
0	メッセージオブジェクト*1 の制御データ(IFx メッセージ制御レジスタ(IFxMCTR))を更新しないことを示します。 [初期値]
1	メッセージオブジェクト*1 の制御データ(IFx メッセージ制御レジスタ(IFxMCTR))を更新することを示します。

[bit3] CIP : 割込みクリアビット

"0"または"1"を設定しても CAN コントローラへの動作に影響を与えません。

[bit2] TxRqst/NewDat : メッセージ送信要求ビット

TxRqst/ NewDat	機能
0	メッセージオブジェクト*1 および CAN 送信要求レジスタ(TREQR)の TxRqst ビットを変更しないことを示します。 [初期値]
1	メッセージオブジェクト*1 および CAN 送信要求レジスタ(TREQR)の TxRqst ビットに"1"を設定(送信要求)することを示します。

[bit1] Data A : データ 0-3 更新ビット

Data A	機能
0	メッセージオブジェクト*1 のデータ 0-3 を更新しないことを示します。 [初期値]
1	メッセージオブジェクト*1 のデータ 0-3 を更新することを示します。

[bit0] Data B : データ 4-7 更新ビット

Data B	機能
0	メッセージオブジェクト*1 のデータ 4-7 を更新しないことを示します。 [初期値]
1	メッセージオブジェクト*1 のデータ 4-7 を更新することを示します。

*1 :「42.4.4 メッセージオブジェクト」を参照してください。

＜注意事項＞

- IFx コマンドマスクレジスタ(IFxCMSK)の TxRqst/NewDat ビットが"1"に設定されると、IFx メッセージ制御レジスタ(IFxMCTR)の TxRqst ビットの設定は無効となります。
- テストのベーシックモードでレジスタは無効となります。

(2) 転送方向がリードの場合(WR/RD="0")

[bit6] Mask : マスクデータ更新ビット

Mask	機能
0	メッセージオブジェクト*1 から IFx マスクレジスタ 1, 2(IFxMSK1, IFxMSK2)へデータ(ID マスク + MDir + MXtd)を転送しないことを示します。 [初期値]
1	メッセージオブジェクト*1 から IFx マスクレジスタ 1, 2(IFxMSK1, IFxMSK2)へデータ(ID マスク + MDir + MXtd)を転送することを示します。

[bit5] Arb : アービトレーションデータ更新ビット

Arb	機能
0	メッセージオブジェクト*1 から IFx アービトレーション 1,2(IFxARB1, IFxARB2)へデータ(ID+ Dir + Xtd + MsgVal)を転送しないことを示します。 [初期値]
1	メッセージオブジェクト*1 から IFx アービトレーション 1,2(IFxARB1, IFxARB2)へデータ(ID+ Dir + Xtd + MsgVal)を転送することを示します。

[bit4] Control : 制御データ更新ビット

Control	機能
0	メッセージオブジェクト*1 から IFx メッセージ制御レジスタ(IFxMCTR)へデータを転送しないことを示します。 [初期値]
1	メッセージオブジェクト*1 から IFx メッセージ制御レジスタ(IFxMCTR)へデータを転送することを示します。

[bit3] CIP : 割込みクリアビット

CIP	機能
0	メッセージオブジェクト*1 および CAN 割込みペンディングレジスタ(INTPND)の IntPnd ビットを保持することを示します。 [初期値]
1	メッセージオブジェクト*1 および CAN 割込みペンディングレジスタ(INTPND)の IntPnd ビットを"0"にクリアすることを示します。

[bit2] TxRqst/NewDat : データ更新ビット

TxRqst/ NewDat	機能
0	メッセージオブジェクト*1 および CAN データ更新レジスタの NewDat ビットを保持することを示します。 [初期値]
1	メッセージオブジェクト*1 および CAN データ更新レジスタの NewDat ビットを"0"にクリアすることを示します。

[bit1] Data A : データ 0-3 更新ビット

Data A	機能
0	メッセージオブジェクト*1 および CAN データレジスタ A1, A2 のデータを保持することを示します。 [初期値]
1	メッセージオブジェクト*1 および CAN データレジスタ A1, A2 のデータを更新することを示します。

[bit0] Data B : データ 4-7 更新ビット

Data B	機能
0	メッセージオブジェクト*1 および CAN データレジスタ B1, B2 のデータを保持することを示します。 [初期値]
1	メッセージオブジェクト*1 および CAN データレジスタ B1, B2 のデータを更新することを示します。

*1 : 「42.4.4 メッセージオブジェクト」を参照してください。

＜注意事項＞

- メッセージオブジェクトへの読出しアクセスにより、IntPnd および NewDat ビットは"0"にリセットすることが可能です。ただし、IFx メッセージ制御レジスタ(IFxMCTR)の IntPnd および NewDat ビットには、読出しアクセスによりリセットされる前の IntPnd, NewDat ビットが格納されます。
- テストのベーシックモードでは無効となります。

42.4.3.3 IFx マスクレジスタ 1,2 (IFxMSK1, IFxMSK2)

IFx マスクレジスタ 1,2 のビット構成について説明します。

メッセージ RAM のメッセージオブジェクトマスクデータをライト/リードするために用いられます。また、テストのベーシックモードでは、設定されているマスクデータは無効となります。

各ビットの機能については、「42.4.4 メッセージオブジェクト」に記述されています。

■ IFx マスクレジスタ 2(上位バイト) : アドレス Base_addr+14H & Base_addr+44H(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	MXtd	MDir	予約	Msk28～Msk24				
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R1,W1	R/W	R/W	R/W	R/W	R/W

■ IFx マスクレジスタ 2(下位バイト) : アドレス Base_addr+15H & Base_addr+45H(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	Msk23～Msk16							
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ IFx マスクレジスタ 1(上位バイト) : アドレス Base_addr+16H & Base_addr+46H(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	Msk15～Msk8							
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ IFx マスクレジスタ 1(下位バイト) : アドレス Base_addr+17H & Base_addr+47H(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	Msk7～Msk0							
初期値	1	1	1	1	1	1	1	1
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

「予約ビット(IFx マスクレジスタ 2 の bit13)は"1"が読み出され、書込み時は"1"を書き込んでください。

42.4.3.4 IFx アービトレーションレジスタ 1,2 (IFxARB1, IFxARB2)

IFx アービトレーションレジスタ 1,2 のビット構成について説明します。

メッセージ RAM のメッセージオブジェクトアービトレーションデータをライト/リードするために用いられます。また、テストのベーシックモードでは無効となります。

各ビットの機能については、「42.4.4 メッセージオブジェクト」に記述されています。

■ IFx アービトレーションレジスタ 2(上位バイト) : アドレス Base_addr+18H & Base_addr+48H(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	MsgVal	Xtd	Dir	ID28～ID24				
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ IFx アービトレーションレジスタ 2(下位バイト) : アドレス Base_addr+19H & Base_addr+49H(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ID23～ID16							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ IFx アービトレーションレジスタ 1(上位バイト) : アドレス Base_addr+1AH & Base_addr+4AH(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	ID15～ID8							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ IFx アービトレーションレジスタ 1(下位バイト) : アドレス Base_addr+1BH & Base_addr+4BH(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ID7～ID0							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

＜注意事項＞

送信途中でメッセージオブジェクトの MsgVal ビットを"0"にクリアした場合、送信が完了した時点で CAN ステータスレジスタ(STATR)の TxOk ビットは"1"になりますが、メッセージオブジェクトおよび CAN 送信要求レジスタ(TREQR)の TxRqst ビットは"0"にクリアされませんのでメッセージインタフェースレジスタによって TxRqst ビットを"0"にクリアしてください。

42.4.3.5 IFx メッセージ制御レジスタ (IFxMCTR)

IFx メッセージ制御レジスタのビット構成について説明します。

メッセージ RAM のメッセージオブジェクト制御データをライト/リードするために用いられます。また、テストのベーシックモードでは、IF1 メッセージ制御レジスタは無効となります。IF2 メッセージ制御レジスタの NewDat と MsgLst は通常の動作を行い、DLC[3:0]ビットは受信したメッセージの DLC を表示します。その他の制御ビットは無効("0")として動作します。

各ビットの機能については、「42.4.4 メッセージオブジェクト」に記述されています。

■ IFx メッセージ制御レジスタ(上位バイト) : アドレス Base_addr+1CH & Base_addr+4CH(アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	NewDat	MsgLst	IntPnd	UMask	TxIE	RxIE	RmtEn	TxRqst
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ IFx メッセージ制御レジスタ(下位バイト) : アドレス Base_addr+1D$_H$ & Base_addr+4D$_H$(アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	EoB	予約	予約	予約	DLC 3-0			
初期値	0	0	0	0	0	0	0	0
属性	R/W	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W	R/W

＜注意事項＞

TxRqst, NewDat および IntPnd ビットは、IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD ビットの設定によって以下のような動作になります。

- 転送方向がライトの場合(IFx コマンドマスクレジスタ(IFxCMSK):WR/RD="1")
  - ☐ IFx コマンドマスクレジスタ(IFxCMSK)の TxRqst/NewDat が"0"に設定されている場合のみ、TxRqst ビットが有効となります。
- 転送方向がリードの場合(IFx コマンドマスクレジスタ(IFxCMSK):WR/RD="0")
  - ☐ IFx コマンドマスクレジスタ(IFxCMSK)の CIP ビットを"1"、IFx コマンド要求レジスタ(IFxCREQ)への書込みによりメッセージオブジェクトおよびCAN割込みペンディングレジスタ(INTPND)の IntPnd ビットをリセットした場合、レジスタにはリセットされる前の IntPnd ビットが格納されます。
  - ☐ IFx コマンドマスクレジスタ(IFxCMSK)の TxRqst/NewDat ビットを"1"、IFx コマンド要求レジスタ(IFxCREQ)への書込みによりメッセージオブジェクトおよび CAN データ更新レジスタの NewDat ビットをリセットした場合、レジスタにはリセットする前の NewDat ビットが格納されます。

42.4.3.6 IFx データレジスタ A1,A2,B1,B2 (IFxDTA1, IFxDTA2, IFxDTB1, IFxDTB2)

IFx データレジスタ A1,A2,B1,B2 のビット構成について説明します。

メッセージ RAM のメッセージオブジェクト送受信データをライト/リードするために用いられます。データフレームの送受信のみ使用され、リモートフレームの送受信には使用されません。

	addr+0	addr+1	addr+2	addr+3
IFx Message Data A1 (addresses 20_H & 50_H)	Data(0)	Data(1)		
IFx Message Data A2 (addresses 22_H & 52_H)			Data(2)	Data(3)
IFx Message Data B1 (addresses 24_H & 54_H)	Data(4)	Data(5)		
IFx Message Data B2 (addresses 26_H & 56_H)			Data(6)	Data(7)
IFx Message Data A2 (addresses 30_H & 60_H)	Data(3)	Data(2)		
IFx Message Data A1 (addresses 32_H & 62_H)			Data(1)	Data(0)
IFx Message Data B2 (addresses 34_H & 64_H)	Data(7)	Data(6)		
IFx Message Data B1 (addresses 36_H & 66_H)			Data(5)	Data(4)

■ IFx データレジスタ：

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	Data							
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

送信メッセージデータの設定
設定したデータは、MSB(bit7, bit15)より開始して Data(0), Data(1), …, Data(7)の順で送信されます。

受信メッセージデータ
受信メッセージデータは、MSB(bit7, bit15)より開始して Data(0), Data(1), …, Data(7)の順で格納されます。

＜注意事項＞

- 受信メッセージデータが 8 バイトより少ない場合は、データレジスタの残りのバイトには不定データが書き込まれます。
- メッセージオブジェクトへの転送は、Data A もしくは Data B の 4 バイト単位で行われますので、4 バイトのうちある一部のデータだけを更新することはできません。

42.4.4 メッセージオブジェクト

メッセージオブジェクトについて説明します。

メッセージ RAM には、32 (チャネルによっては 64 または 128 まで)のメッセージオブジェクトがあります。メッセージ RAM への CPU アクセスと CAN コントローラからのアクセスの競合を回避するために、CPU はメッセージオブジェクトへ直接、アクセスはできません。これらのアクセスは、IFx メッセージインタフェースレジスタ経由で行います。

メッセージオブジェクトの構成と機能を説明します。

42.4.4.1 メッセージオブジェクトの構成

メッセージオブジェクトの構成について説明します。

以下にメッセージオブジェクトの構成を示します。

表 42-6 メッセージオブジェクトの構成

UMask	Msk28-0	MXtd	MDir	EoB	NewDat		MsgLst	RxIE	TxIE	IntPnd	RmtEn	TxRqst
MsgVal	ID28-0	Xtd	Dir	DLC3-0	Data0	Data1	Data2	Data3	Data4	Data5	Data6	Data7

＜注意事項＞

メッセージオブジェクトは、CAN 制御レジスタ(CTRLR)の Init ビット、ハードウェアリセットでは初期化されません。ハードウェアリセットの場合、ハードウェアリセット解除後、CPU によってメッセージ RAM を初期化するかメッセージ RAM の MsgVal ビットを"0"にしてください。

42.4.4.2 メッセージオブジェクトの機能

メッセージオブジェクトの機能について説明します。

ID28-ID0, Xtd, Dir ビットは、メッセージを送信する場合、ID とメッセージの種類に使用されます。メッセージを受信する場合、Msk28-Msk0, MXtd, MDir ビットと共に受容フィルタで使用されます。
受容フィルタを通過したデータフレームもしくはリモートフレームの ID, IDE,RTR, DLC および DATA は、メッセージオブジェクトの ID28-ID0, Xtd, Dir, DLC3-DLC0, Data7-Data0 に格納されます。Xtd は拡張フレームか標準フレームかを示し、Xtd が"1" の場合、29 ビット ID(拡張フレーム)を、Xtd が"0" の場合は、11 ビット ID(標準フレーム)を受信することになります。
受信したデータフレームもしくはリモートフレームが 1 つ以上のメッセージオブジェクトと一致した場合は、一致した最小のメッセージ番号に格納されます。(詳細は、「42.5.3 メッセージ受信動作」の受信メッセージの受容フィルタを参照してください。)

MsgVal: 有効メッセージビット

MsgVal	機能
0	メッセージオブジェクトは無効です。 メッセージの送受信は行いません。
1	メッセージオブジェクトは有効です。 メッセージの送受信が可能となります。

＜注意事項＞

- CAN 制御レジスタ(CTRLR)の Init ビットを"0"にリセットする前と、ID28-ID0, Xtd, Dir, DLC3-0 を変更する前には、メッセージオブジェクトの MsgVal ビットを必ず初期化してください。
- 送信中に MsgVal ビットを"0"にすると送信が完了した時点で CAN ステータスレジスタ(STATR)の TxOk ビットは"1"になりますが、メッセージオブジェクトおよび CAN 送信要求レジスタ(TREQR)の TxRqst ビットは"0"にクリアされませんのでメッセージインタフェースレジスタによって TxRqst ビットを"0"にクリアしてください。

UMask: 受容マスク許可ビット

UMask	機能
0	Msk28-Msk0, MXtd, MDir を使用しません。
1	Msk28-Msk0, MXtd, MDir を使用します。

＜注意事項＞

- CAN 制御レジスタ(CTRLR)の Init ビットが"1"のとき、あるいは MsgVal ビットが"0"のときに UMask ビットを変更してください。
- Dir ビットが"1"かつ RmtEn ビットが"0"のとき、UMask ビットの設定により動作が異なります。
  - UMask ビットが"1"の場合は、受容フィルタを通過しリモートフレームを受信したとき、TxRqst ビットを"0"にリセットします。そのとき、受信した ID, IDE, RTR, DLC はメッセージオブジェクトに格納し、NewDat ビットは"1"にセットされ、データは変更しません。(データフレームのように取り扱います)
  - UMask ビットが"0"の場合は、リモートフレーム受信に対し、TxRqst ビットはそのまま保持し、リモートフレームを無視します。

ID28～ID0: メッセージ ID

	機能
ID28 - ID0	29 ビット ID(拡張フレーム)を指示します。
ID28 - ID18	11 ビット ID(標準フレーム)を指示します。

Msk28～Msk0: ID マスク

Msk	機能
0	メッセージオブジェクトの ID と対応するビットをマスクします。
1	メッセージオブジェクトの ID と対応するビットをマスクしません。

Xtd: 拡張 ID 許可ビット

Xtd	機能
0	メッセージオブジェクトは 11 ビット ID(標準フレーム)が使用されます。
1	メッセージオブジェクトは 29 ビット ID(拡張フレーム)が使用されます。

MXtd: 拡張 ID マスクビット

MXtd	機能
0	メッセージオブジェクトの Xtd ビットに設定した値と、受信したフレームの IDE ビットの値の比較を行いません。受信したフレームの IDE ビットにしたがって標準フレームの ID として比較するか、拡張フレームの ID として比較するか決定します。
1	メッセージオブジェクトの Xtd ビットに設定した値と、受信したフレームの IDE ビットの値の比較を行います。

＜注意事項＞

11 ビット ID(標準フレーム)がメッセージオブジェクトに設定されると、受信したデータフレームの ID は、ID28～ID18 へ書き込まれます。ID マスクは、Msk28～Msk18 が使用されます。

Dir: メッセージ方向ビット

Dir	機能
0	受信方向を示します。 TxRqst ビットが"1"にセットされると、リモートフレームの送信を行い、TxRqst ビットが"0"のときは、受容フィルタを通過したデータフレームを受信します。
1	送信方向を示します。 TxRqst ビットが"1"にセットされると、データフレームを送信し、TxRqst ビットが"0"で RmtEn が"1"にセットされている場合、受容フィルタを通過したリモートフレームの受信によって、CAN コントローラ自身が TxRqst を"1"にセットします。

MDir: メッセージ方向マスクビット

MDir	機能
0	受容フィルタでメッセージ方向ビット(Dir)をマスクします。
1	受容フィルタでメッセージ方向ビット(Dir)をマスクしません。

＜注意事項＞

MDir ビットは常に"1"を設定してください。

EoB: エンドオブバッファビット(詳細は、「5.4 FIFO バッファ機能」参照)

EoB	機能
0	メッセージオブジェクトは FIFO バッファとして使用され、最終メッセージでないことを示します。
1	単一メッセージオブジェクトまたはFIFOバッファの最終メッセージオブジェクトを示します。

＜注意事項＞

- EoB ビットは、2～32 メッセージの FIFO バッファを構成するために使用します。
- 単一メッセージオブジェクト(FIFO を使用しない場合)は、必ず EoB ビットに"1"を設定してください。

NewDat: データ更新ビット

NewDat	機能
0	有効なデータがないことを示します。
1	有効なデータがあることを示します。

MsgLst: メッセージロスト

MsgLst	機能
0	メッセージロストは発生していません。
1	メッセージロストが発生しています。

＜注意事項＞

MsgLst ビットは Dir ビットが"0"のとき(受信方向)のみ有効です。

RxIE: 受信割込みフラグイネーブルビット

RxIE	機能
0	フレーム受信成功後、IntPnd ビットは変更されません。
1	フレーム受信成功後、IntPnd ビットが"1"にセットされます。

TxIE: 送信割込みフラグイネーブルビット

TxIE	機能
0	フレーム送信成功後、IntPnd ビットは変更されません。
1	フレーム送信成功後、IntPnd ビットが"1"にセットされます。

IntPnd: 割込みペンディングビット

IntPnd	機能
0	割込み要因がありません。
1	割込み要因があります。 ほかに優先度の高い割込みがない場合は、CAN 割込みレジスタ(INTR)の IntId ビットはこのメッセージオブジェクトを示します。

RmtEn: リモートイネーブル

RmtEn	機能
0	リモートフレームの受信で、TxRqst ビットは変更されません。
1	Dir ビットが"1"でリモートフレームを受信すると、TxRqst ビットが"1"にセットされます。

＜注意事項＞

Dir ビットが"1"かつ RmtEn ビットが"0"のとき、UMask ビットの設定により動作が異なります

- UMask が"1"の場合は、受容フィルタを通過しリモートフレームを受信したとき、TxRqst ビットを"0"にリセットします。そのとき、受信した ID, IDE, RTR, DLC はメッセージオブジェクトに格納し、NewDat ビットは"1"にセットされ、データは変更しません。(データフレームのように取り扱います)
- UMask が"0"の場合は、リモートフレーム受信に対し、TxRqst ビットはそのまま保持し、リモートフレームを無視します。

TxRqst: 送信要求ビット

TxRqst	機能
0	送信アイドル状態(送信中でもないし、送信待ち状態でもない)を示します。
1	送信中または、送信待ちであることを示します。

DLC3-0: データ長コード

DLC3-0	機能
0-8	データフレーム長は 0～8 バイトです。
9-15	設定禁止です。 設定された場合は、8 バイト長となります。

＜注意事項＞

データフレームを受信すると DLC ビットには、受信した DLC が格納されます。

Data 0-7: データ 0-7

	機能
Data 0	CAN データフレームの最初のデータバイト
Data 1	CAN データフレームの 2 番目のデータバイト
Data 2	CAN データフレームの 3 番目のデータバイト
Data 3	CAN データフレームの 4 番目のデータバイト
Data 4	CAN データフレームの 5 番目のデータバイト
Data 5	CAN データフレームの 6 番目のデータバイト
Data 6	CAN データフレームの 7 番目のデータバイト
Data 7	CAN データフレームの 8 番目のデータバイト

＜注意事項＞

- CAN バスへのシリアル出力は、MSB(bit7 もしくは bit15)より出力します。
- 受信メッセージデータが 8 バイトより少ない場合は、データレジスタの残りのバイトには不定が書き込まれます。
- メッセージオブジェクトへの転送は、Data A もしくは Data B の 4 バイト単位で行われますので、4 バイトのうちある一部のデータだけを更新することはできません。

42.4.5 メッセージハンドラレジスタ

メッセージハンドラレジスタについて示します。

メッセージハンドラレジスタは､すべて読出し専用です｡メッセージオブジェクトのTxRqst, NewDat, IntPnd, MsgValビットとIntIdビットは、ステータスを表示します。

- CAN送信要求レジスタ 1, 2 (TREQR1, TREQR2)
- CAN データ更新レジスタ 1, 2 (NEWDT1, NEWDT2)
- CAN 割込みペンディングレジスタ 1,2 (INTPND1, INTPND2)
- CAN メッセージ有効レジスタ 1, 2(MSGVAL1, MSGVAL2)

42.4.5.1 CAN 送信要求レジスタ (TREQR1, TREQR2)

CAN 送信要求レジスタのビット構成について説明します。

全メッセージオブジェクトのTxRqstビットを表示します。TxRqstビットを読み出すことにより、どのメッセージオブジェクトの送信要求がペンディング中であるか確認することができます。

■ CAN送信要求レジスタ2(上位バイト) : アドレスBase_addr+ 80H (アクセス:バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TxRqst32～TxRqst25							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN送信要求レジスタ2(下位バイト) : アドレスBase_addr+81H (アクセス:バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TxRqst24～TxRqst17							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN送信要求レジスタ1(上位バイト) : アドレスBase_addr+82H (アクセス:バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	TxRqst16～TxRqst9							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN送信要求レジスタ 1(下位バイト) : アドレス Base_addr+83H (アクセス:バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	TxRqst8～TxRqst1							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

TxRqst32-1: 送信要求ビット

TxRqst32-1	機能
0	送信アイドル状態(送信中でもないし、送信待ち状態でもない)を示します。
1	送信中または、送信待ちであることを示します。

TxRqst ビットのセット/リセット条件を以下に示します。

セット条件

IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD に"1", TxRqst に"1"を設定し、IFx コマンド要求レジスタ(IFxCREQ)への書込みにより特定オブジェクトの TxRqst にセットできます。

IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD に"1", TxRqst に"0",Control に"1" を設定し、IFx メッセージ制御レジスタ(IFxMCTR)の TxRqst に"1"を設定した場合、IFx コマンド要求レジスタ(IFxCREQ)への書込みにより特定オブジェクトの TxRqst にセットできます。

Dir ビットが"1", RmtEn ビットが"1"に設定し、受容フィルタを通過したリモートフレームの受信によりセットされます。

リセット条件

IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD に"1", TxRqst に"0", Control に"1" を設定し、IFx メッセージ制御レジスタ(IFxMCTR)の TxRqst に"0"を設定した場合、IFx コマンド要求レジスタ(IFxCREQ)への書込みにより特定オブジェクトの TxRqst をリセットできます。

フレームの送信が正常終了すると、リセットされます。

Dir が"1", RmtEN が"0", UMask が"1"の場合、受容フィルタを通過したリモートフレームの受信によりリセットされます。

32 メッセージバッファ以上を搭載する CAN マクロにおける送信要求ビットの確認については以下の表を参照してください。

表 42-7 32 メッセージバッファ以上を搭載する CAN マクロにおける送信要求ビット

		addr + 0	addr + 1	addr + 2	addr + 3
TREQR 4 & 3	TxRqst 64-33 (address 84_H)	TxRqst64-57	TxRqst56-49	TxRqst48-41	TxRqst40-33
TREQR 6 & 5	TxRqst 96-65 (address 88_H)	TxRqst96-89	TxRqst88-81	TxRqst80-73	TxRqst72-65
TREQR 8 & 7	TxRqst 128-97 (address $8C_H$)	TxRqst128-121	TxRqst120-113	TxRqst112-105	TxRqst104-97

＜注意事項＞

- 優先順位が最下位のメッセージバッファを送信に使用している場合、TXRqst に"1"を設定し、送信中止のため、TxRqst に"0"を設定した場合、そのタイミングによっては、再度 TxRqst に"1" を設定したとき、以下のいずれかのイベントが発生するまで、メッセージが送信されないことがあります。
  - ☐ CAN バス上に有効なメッセージが流れる
  - ☐ ほかのメッセージバッファに対して送信要求が発行される
  - ☐ Init ビットによって CAN が初期化される

  システム上、もし送信を中止する状況が発生する場合、送信メッセージバッファとして優先順位が最下位のメッセージバッファを使用しないか、送信中止後、上記のいずれかのイベントを発生させた後、再度 TxRqst に"1"を設定してください。
- TxRqst ビットが"1"のときに ID28-0, DLC3-0, Xtd, Data7-0 のメッセージオブジェクトを変更すると変更前後のメッセージオブジェクトが混在して送出されたり、変更後のメッセージオブジェクトが送出されなかったりする可能性があるので TxRqst ビットが"0"のときにそれらを変更してください。

42.4.5.2 CAN データ更新レジスタ (NEWDT1, NEWDT2)

CAN データ更新レジスタのビット構成について説明します。

全メッセージオブジェクトの NewDat ビットを表示します。NewDat ビットを読み出すことにより、どのメッセージオブジェクトのデータが更新されたか確認することができます。

■ CAN データ更新レジスタ 2(上位バイト) : アドレス Base_addr+ 90_H (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	NewDat32～NewDat25							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN データ更新レジスタ 2(下位バイト) : アドレス Base_addr+ 91_H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	NewDat24～NewDat17							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN データ更新レジスタ 1(上位バイト) : アドレス Base_addr+ 92_H (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	NewDat16～NewDat9							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN データ更新レジスタ 1(下位バイト) : アドレス Base_addr+ 93_H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	NewDat8～NewDat1							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

NewDat32-1: データ更新ビット

NewDat32-1	機能
0	有効なデータがないことを示します
1	有効なデータがあることを示します

NewDat ビットのセット/リセット条件を以下に示します。

セット条件

IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD に"1", Control に"1"を設定し、IFx メッセージ制御レジスタ(IFxMCTR)の NewDat に"1"を設定して、IFx コマンド要求レジスタ(IFxCREQ)の書込みにより特定オブジェクトにセットできます。

受容フィルタを通過したデータフレームの受信によりセットされます。

Dir が"1", RmtEN が"0", UMask が"1"の場合、受容フィルタを通過したリモートフレームの受信によりセットされます。

リセット条件

IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD に"0", NewDat に"1"を設定した場合、IFx コマンド要求レジスタ(IFxCREQ)の書込みにより特定オブジェクトの NewDat をリセットできます。

IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD に"1", Control に"1"を設定し、IFx メッセージ制御レジスタ(IFxMCTR)の NewDat を"0"に設定した場合、IFx コマンド要求レジスタ(IFxCREQ)の書込みにより特定オブジェクトの NewDat をリセットできます。

送信用シフトレジスタ(内部レジスタ)へデータを転送終了後、リセットされます。

32 メッセージバッファ以上を搭載する CAN マクロにおけるデータ更新ビットの確認については以下の表を参照してください。

表 42-8 32 メッセージバッファ以上を搭載する CAN マクロにおけるデータ更新ビット

		addr + 0	addr + 1	addr + 2	addr + 3
NEWDT 4 & 3	NewDat 64-33 (address 94_H)	NewDat64-57	NewDat56-49	NewDat48-41	NewDat40-33
NEWDT 6 & 5	NewDat 96-65 (address 98_H)	NewDat96-89	NewDat88-81	NewDat80-73	NewDat72-65
NEWDT 8 & 7	NewDat 128-97 (address $9C_H$)	NewDat128-121	NewDat120-113	NewDat112-105	NewDat104-97

42.4.5.3 CAN 割込みペンディングレジスタ(INTPND1, INTPND2)

CAN 割込みペンディングレジスタのビット構成について説明します。

全メッセージオブジェクトの IntPnd ビットを表示します。IntPnd ビットを読み出すことにより、どのメッセージオブジェクトが割込みペンディング中であるか確認することができます。

■ CAN 割込みペンディングレジスタ 2(上位バイト) : アドレス Base_addr+ $A0_H$ (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	IntPnd32～IntPnd25							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN 割込みペンディングレジスタ 2(下位バイト) : アドレス Base_addr+ $A1_H$ (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IntPnd24～IntPnd17							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN 割込みペンディングレジスタ 1(上位バイト) : アドレス Base_addr+ $A2_H$ (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	IntPnd16～IntPnd9							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN 割込みペンディングレジスタ 1(下位バイト) : アドレス Base_addr+ $A3_H$ (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	IntPnd8～IntPnd1							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

IntPnd32-1: 割込みペンディングビット

IntPnd32-1	機能
0	割込み要因がありません
1	割込み要因があります

IntPnd ビットのセット/リセット条件を以下に示します。

セット条件

TxIE が"1"にセットされている場合、フレームの正常送信完了によりセットされます。

RxIE が"1"にセットされている場合、受容フィルタを通過したフレームの正常受信完了によりセットされます。

IFx コマンドマスクレジスタの WR/RD に"1", Control に"1", IFx メッセージ制御レジスタの IntPnd に"1"を設定して、IFx コマンド要求レジスタの書込みにより、特定オブジェクトの IntPnd をセットできます。

リセット条件

IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD に"0", CIP に"1"を設定した場合、IFx コマンド要求レジスタ(IFxCREQ)の書込みにより特定オブジェクトの IntPnd をリセットできます。IFx コマンドマスクレジスタの WR/RD に"1", Control に"1"を設定し、IFx メッセージ制御レジスタの IntPnd を"0"に設定した場合、IFx コマンド要求レジスタの書込みにより特定オブジェクトの IntPnd をリセットできます。

32 メッセージバッファ以上を搭載する CAN マクロにおける割込みペンディングビットの確認については以下の表を参照してください。

表 42-9 32 メッセージバッファ以上を搭載する CAN マクロにおける割込みペンディングビット

		addr + 0	addr + 1	addr + 2	addr + 3
INTPND 4 & 3	IntPnd 64-33 (address $A4_H$)	IntPnd64-57	IntPnd56-49	IntPnd48-41	IntPnd40-33
INTPND 6 & 5	IntPnd 96-65 (address $A8_H$)	IntPnd96-89	IntPnd88-81	IntPnd80-73	IntPnd72-65
INTPND 8 & 7	IntPnd 128-97 (address AC_H)	IntPnd128-121	IntPnd120-113	IntPnd112-105	IntPnd104-97

42.4.5.4 CAN メッセージ有効レジスタ (MSGVAL1, MSGVAL2)

CAN メッセージ有効レジスタのビット構成について説明します。

全メッセージオブジェクトの MsgVal ビットを表示します。MsgVal ビットを読み出すことにより、どのメッセージオブジェクトが有効であるかチェックすることができます。

■ CAN メッセージ有効レジスタ 2(上位バイト) : アドレス Base_addr+B0H (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	MsgVal32～MsgVal 25							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN メッセージ有効レジスタ 2(下位バイト) : アドレス Base_addr+B1H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MsgVal24～MsgVal17							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN メッセージ有効レジスタ 1(上位バイト) : アドレス Base_addr+B2H (アクセス : バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	MsgVal16～MsgVal9							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

■ CAN メッセージ有効レジスタ 1(下位バイト) : アドレス Base_addr+B3H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MsgVal8～MsgVal1							
初期値	0	0	0	0	0	0	0	0
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

MsgVal32-1: メッセージ有効ビット

MsgVal 32-1	機能
0	メッセージオブジェクトは無効です メッセージの送受信は行いません
1	メッセージオブジェクトは有効です メッセージの送受信が可能となります

MsgVal ビットのセット/リセット条件を以下に示します。

セット条件

IFx コマンドマスクレジスタの WR/RD に"1", Arb に"1"を設定し、IFx アービトレーションレジスタ 2 の MsgVal ビットに"1"を設定し、IFx コマンド要求レジスタ(IFxCREQ)への書込みにより特定オブジェクトの MsgVal ビットをセットできます。

リセット条件

IFx コマンドマスクレジスタの WR/RD に"1", Arb に"1"を設定し、IFx アービトレーションレジスタ 2 の MsgVal ビットに"0"を設定し、IFx コマンド要求レジスタ(IFxCREQ)の書込みにより特定オブジェクトの MsgVal ビットをクリアできます。

32 メッセージバッファ以上を搭載する CAN マクロにおけるメッセージ有効ビットの確認については以下の表を参照してください。

表 42-10 32 メッセージバッファ以上を搭載する CAN マクロにおけるメッセージ有効ビット

		addr + 0	addr + 1	addr + 2	addr + 3
MSGVAL 4 & 3	MsgVal 64-33 (address $B4_H$)	MsgVal64-57	MsgVal56-49	MsgVal48-41	MsgVal40-33
MSGVAL 6 & 5	MsgVal 96-65 (address $B8_H$)	MsgVal96-89	MsgVal88-81	MsgVal80-73	MsgVal72-65
MSGVAL 8 & 7	MsgVal 128-97 (address BC_H)	MsgVal128-121	MsgVal120-113	MsgVal112-105	MsgVal104-97

42.4.6 CAN プリスケーラレジスタ (CANPRE)

CAN プリスケーラレジスタのビット構成について説明します。

CAN システムクロック(fsys)生成プリスケーラを設定するものです。詳細は「42.5.6 ビットタイミング・CAN システムクロック (fsys) 生成」を参照してください。このレジスタの値を変更する場合は、CAN 制御レジスタ(CTRLR)の初期化ビット(Init)を"1"にセットし、すべてのバス動作を停止してください。

■ CAN プリスケーラレジスタ : アドレス 04A4H (アクセス : バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約	予約			CANPRE3	CANPRE2	CANPRE1	CANPRE0
初期値	0	0	0	0	0	0	0	0
属性	R/W0	R0,W0	R0,W0	R0,W0	R/W	R/W	R/W	R/W

[bit7] 予約
必ず"0"を書き込んでください。

[bit6～bit4] 予約
常に"0"が読み出されます。書込みは"0"を設定してください。

[bit3～bit0] CANPRE[3:0] : CAN プリスケーラ設定ビット

CANPRE [3:0]	機能	入力 CAN プリスケーラクロック 80MHz 時	入力 CAN プリスケーラクロック 64MHz 時	入力 CAN プリスケーラクロック 48MHz 時
0000	CANクロックとしてシステムクロックの 1/1 周期が選択されます。(初期値: CANPRE[3:0]=0000)	80MHz	64MHz	48MHz
0001	CANクロックとしてシステムクロックの 1/2 周期が選択されます。	40MHz	32MHz	24MHz
001x	CANクロックとしてシステムクロックの 1/4 周期が選択されます。	20MHz	16MHz	12MHz
01xx	CANクロックとしてシステムクロックの 1/8 周期が選択されます。	10MHz	8MHz	6MHz
1000	CANクロックとしてシステムクロックの 2/3 周期が選択されます。クロックの Duty は 67% となります。	53.3MHz	42.7MHz	32MHz
1001	CANクロックとしてシステムクロックの 1/3 周期が選択されます。	26.7MHz	21.4MHz	16MHz
1010	CANクロックとしてシステムクロックの 1/6 周期が選択されます。	13.3MHz	10.7MHz	8MHz
1011	CANクロックとしてシステムクロックの 1/12 周期が選択されます。	6.7MHz	5.4MHz	4MHz
110x	CANクロックとしてシステムクロックの 1/5 周期が選択されます。	16.0MHz	12.8MHz	9.6MHz
111x	CANクロックとしてシステムクロックの 1/10 周期が選択されます。	8.0MHz	6.4MHz	4.8MHz

＜注意事項＞

- CAN プリスケーラ設定ビットの変更は、CAN 制御レジスタ(CTRLR)の初期化ビットを"1"にセットし、すべてのバス動作を停止させた後に行ってください。
- レジスタの設定により CAN インタフェースに供給するクロックは 16MHz 以下としてください。

42.5 動作説明

CAN の動作について説明します。

CAN には以下の機能があります。

- メッセージオブジェクト
- メッセージ送信動作
- メッセージ受信動作
- FIFO バッファ機能
- 割込み機能
- ビットタイミング
- テストモード
- ソフトウェア初期化

42.5.1 メッセージオブジェクト

メッセージオブジェクトについて示します。

メッセージ RAM のメッセージオブジェクトとインタフェースについて説明します。

42.5.1.1 メッセージオブジェクト

メッセージオブジェクトについて説明します。

メッセージ RAM のメッセージオブジェクト設定(MsgVal, NewDat, IntPnd, TxRqst ビットを除く)は、ハードウェアリセットによって初期化されません。そのためメッセージオブジェクトを CPU で初期化するか、MsgVal ビットを無効(MsgVal="0")に設定してください。また、CAN ビットタイミングレジスタ(BTR) と CAN プリスケーラ拡張レジスタ(BRPER) の設定は CAN 制御レジスタ(CTRLR) の Init ビットが"1"、CCE ビットが"1"のとき行ってください。

メッセージオブジェクトの設定は、メッセージインタフェースレジスタ(IFx マスクレジスタ(IFCMSK), IFx アービトレーションレジスタ(IFxARB), IFx メッセージ制御レジスタ(IFxMCTR), IFx データレジスタ(IFxDTx))に設定した後、IFx コマンド要求レジスタ(IFxCREQ)へのメッセージ番号の書込みにより、そのインタフェースレジスタのデータが指定されたメッセージオブジェクトへ転送されます。

CAN 制御レジスタ(CTRLR)の Init ビットが"0"にクリアされると CAN コントローラは動作を開始します。受容フィルタを通過した受信メッセージは、メッセージ RAM へ格納されます。送信要求が保留されているメッセージは、メッセージ RAM から CAN コントローラのシフトレジスタへ転送され、CAN バスへの送信が行われます。

CPU は、メッセージインタフェースレジスタ経由で受信メッセージの読出しおよび、送信メッセージの更新を行います。また、CAN 制御レジスタ(CTRLR)および、IFx メッセージ制御レジスタ(IFxMCTR)(メッセージオブジェクト)の設定に応じて、CPU への割込みが行われます。

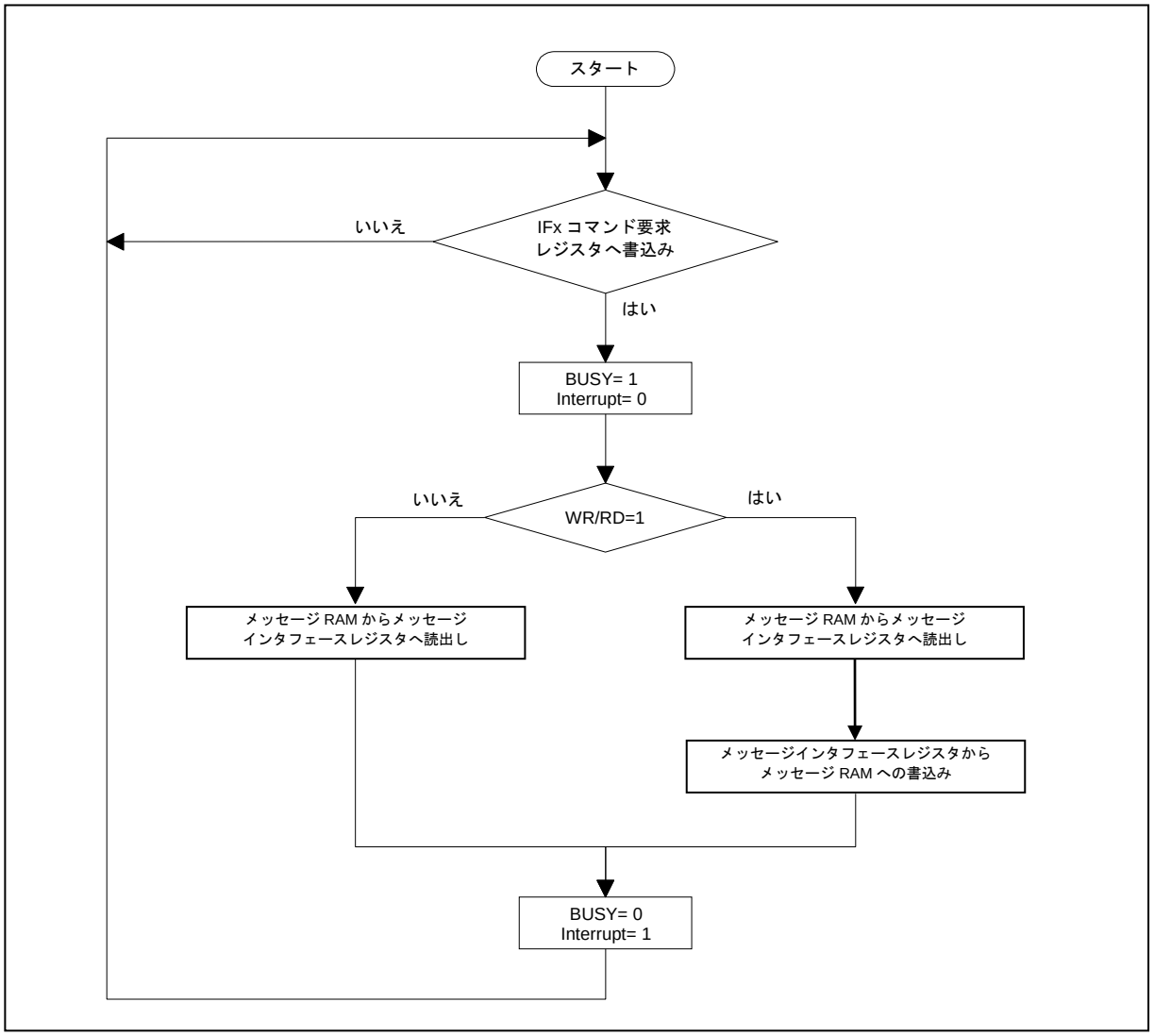
42.5.1.2 メッセージ RAM とのデータ送受信

メッセージ RAM とのデータ送受信について説明します。

メッセージインタフェースレジスタとメッセージ RAM とのデータ転送を開始すると、IFx コマンド要求レジスタ(IFxCREQ)の BUSY ビットを"1"にセットします。転送完了後、BUSY ビットは"0"にクリアされます。(図 5-1 参照)

IFx コマンドマスクレジスタ(IFxCMSK)は、一つのメッセージオブジェクトの全データ転送か、データの部分転送を行うかを設定します。メッセージ RAM の構造上、メッセージオブジェクトの単一ビット/バイトの書込みは不可能となっており、常に一つのメッセージオブジェクトの全データをメッセージ RAM へ書き込みます。したがって、メッセージインタフェースレジスタからメッセージ RAM へのデータ転送は、リードモディファイライトサイクルを実行しています。

図 42-2 メッセージインタフェースレジスタとメッセージ RAM のデータ転送



42.5.2 メッセージ送信動作

メッセージ送信動作について示します。

送信メッセージオブジェクトの設定方法および送信動作について説明します。

42.5.2.1 メッセージ送信

メッセージ送信について説明します。

メッセージインタフェースレジスタとメッセージ RAM 間でデータ転送がない場合、CAN メッセージ有効レジスタ(MSGVAL)の MsgVal ビットと CAN 送信要求レジスタ(TREQR)の TxRqst ビットを評価します。送信要求を保留している中で、最高優先度の有効であるメッセージオブジェクトが、送信用のシフトレジスタへ転送が行われます。そのときメッセージオブジェクトの NewDat ビットは"0"にクリアされます。

正常に送信が完了したとき、メッセージオブジェクトに新たなデータがない(NewDat="0")場合は、TxRqst ビットは"0"にリセットされます。TxIE ビットが"1"にセットされている場合は、送信成功後に IntPnd ビットが"1"にセットされます。CAN コントローラが CAN バス上で調停に負けた場合、あるいは転送中にエラーが発生した場合は、CAN バスがアイドルになり次第、直ちにメッセージの再送信が行われます。

42.5.2.2 送信優先度

送信優先度について示します。

メッセージオブジェクトの送信優先度は、メッセージ番号によって決定します。メッセージオブジェクト 1 が最高の優先度で、メッセージオブジェクト 32(搭載している最大メッセージオブジェクト番号)が最低優先度となります。したがって、2 つ以上の送信要求が保留されていると、対応するメッセージオブジェクトの小さい番号順に転送が行われます。

＜注意事項＞

- 優先順位が最下位のメッセージバッファを送信に使用している場合、TXRqst に"1"を設定し、送信中止のため、TxRqst に"0"を設定した場合、そのタイミングによっては、再度 TxRqst に"1" を設定したとき、以下のいずれかのイベントが発生するまで、メッセージが送信されないことがあります。
  - ☐ CAN バス上に有効なメッセージが流れる
  - ☐ ほかのメッセージバッファに対して送信要求が発行される
  - ☐ Init ビットによって CAN が初期化される

  システム上、もし送信を中止する状況が発生する場合、送信メッセージバッファとして優先順位が最下位のメッセージバッファを使用しないか、送信中止後、上記のいずれかのイベントを発生させた後、再度 TxRqst に"1" を設定してください。
- TxRqst ビットが"1"のときに ID28-0, DLC3-0, Xtd, Data7-0 のメッセージオブジェクトを変更すると変更前後のメッセージオブジェクトが混在して送出されたり、変更後のメッセージオブジェクトが送出されなかったり可能性があるので TxRqst ビットが"0"のときにそれらを変更してください。

42.5.2.3 送信メッセージオブジェクトの設定

送信メッセージオブジェクトの設定について説明します。

以下に送信オブジェクトの初期化方法を示します。

表 42-11 送信メッセージオブジェクトの初期化

MsgVal	Arb	Data	Mask	EoB	Dir	NewDat	MsgLst	RxIE	TxIE	IntPnd	RmtEn	TxRqst
1	appl.	appl.	appl.	1	1	0	0	0	appl.	0	appl.	0

IFx アービトレーションレジスタ(ID28-0 と Xtd ビット)は、アプリケーションで与えられ、送信メッセージの ID およびメッセージの種類を定義します。

標準フレーム(11 ビット ID)を設定した場合は、ID28～ID18 を使用し、ID17～ID0 は無効となります。拡張フレーム(29 ビット ID)を設定した場合は、ID28～ID0 を使用します。

TxIE ビットに"1"をセットすると、メッセージオブジェクトの送信成功後に IntPnd ビットが"1"にセットされます。

RmtEn ビットに"1"をセットすると、一致するリモートフレームを受信後、TxRqst ビットに"1"をセットし、データフレームを自動的に送信します。

データレジスタ(DLC3-0, Data0-7)の設定は、アプリケーションで与えられます。

Umask="1"のとき、IFx マスクレジスタ(Msk28-0, UMask, MXtd, MDir ビット)は、マスク設定によりグループ化された ID を持つリモートフレームを受信し、その後、送信を許可(TxRqst ビットに"1"をセット)するために使用されます。詳細は、「42.5.3 メッセージ受信動作」のリモートフレームを参照してください。

＜注意事項＞

IFx マスクレジスタの Dir ビットをマスク許可に設定することは禁止です。

42.5.2.4 送信メッセージオブジェクトの更新

送信メッセージオブジェクトの更新について説明します。

CPU は、送信メッセージオブジェクトのデータをメッセージインタフェースレジスタ経由で、更新できます。

送信メッセージオブジェクトのデータは、対応する IFx データレジスタ(IFxDTx)の 4 バイト単位(IFx データレジスタ A(IFxDTAx), IFx データレジスタ B(IFxDTBx)単位)でデータ書込みが行われます。そのため 1 バイトだけ送信メッセージオブジェクトを変更することはできません。

8 バイトのデータを更新するときは、まず IFx コマンドマスクレジスタ(IFxCMSK)へ 0087_H の書込みを行います。そして、IFx コマンド要求レジスタ(IFxCREQ)へメッセージ番号の書込みにより、送信メッセージオブジェクトのデータ更新(8 バイトデータ)と TxRqst ビットへの"1"書込みが同時に行われます。

NewDat ビットと TxRqst ビットが共に"1"に設定された場合、送信を開始すると NewDat ビットは"0"にリセットされます。

＜注意事項＞

- データを更新する場合は、IFx データレジスタ A(IFxDTAx)もしくは IFx データレジスタ B (IFxDTBx)の 4 バイト単位で行ってください。
- TxRqst ビットが"1" のときに ID28-0, DLC3-0, Xtd, Data7-0 のメッセージオブジェクトを変更すると変更前後のメッセージオブジェクトが混在して送出されたり、変更後のメッセージオブジェクトが送出されなかったりする可能性があるので TxRqst ビットが"0"のときにそれらを変更してください。

42.5.3 メッセージ受信動作

メッセージ受信動作について示します。

受信メッセージオブジェクトの設定方法および受信動作について説明します。

42.5.3.1 受信メッセージの受容フィルタ

受信メッセージの受容フィルタについて示します。

メッセージのアービトレーション/コントロールフィールド(ID＋IDE＋RTR＋DLC)が CAN コントローラ受信用シフトレジスタへ完全にシフトされると、有効メッセージオブジェクトとの一致比較のためにメッセージ RAM のスキャンを開始します。

このとき、メッセージ RAM のメッセージオブジェクトから調停フィールドとマスクデータ(MsgVal, UMask, NewDat, EoB を含む)がロードされ、メッセージオブジェクトとシフトレジスタの調停フィールドがマスクデータを含んで比較されます。

この動作は、"メッセージオブジェクトとシフトレジスタの調停フィールドが一致検出されるまで"、または "メッセージ RAM の最終ワードに到達するまで"、繰り返し実行します。一致が検出されると、メッセージ RAM のスキャンは停止され、受信フレームのタイプ(データフレームまたはリモートフレーム)により、CAN コントローラは処理を行います。

42.5.3.2 受信優先度

受信優先度について示します。

メッセージオブジェクトの受信優先度は、メッセージ番号によって決定します。メッセージオブジェクト 1 が最高の優先度で、メッセージオブジェクト 32 (搭載している最大メッセージオブジェクト番号)が最低優先度となります。したがって、受容フィルタで 2 つ以上一致した場合は、メッセージ番号の小さい番号が受信メッセージオブジェクトとなります。

42.5.3.3 データフレーム受信

データフレーム受信について示します。

CAN コントローラは、受容フィルタで一致したメッセージオブジェクトのメッセージ RAM へ、シフトレジスタから受信メッセージを転送し格納します。この格納するデータは、データバイトだけではなく、すべてのアービトレーションフィールドおよびデータ長コードも格納します。これは、IFx マスクレジスタがマスク設定されている場合でも実行されます。(ID とデータバイトを保持するために格納されます。)

NewDat ビットは、新たなデータが受信されると"1"にセットされます。CPU がメッセージオブジェクトを読出したときには、NewDat ビットを"0"にリセットしてください。メッセージ受信時に、既に NewDat ビットが"1"にセットされている場合は、その前のデータが失われたことになり、MsgLst ビットが"1"にセットされます。

RxIE ビットが"1"にセットされている場合、メッセージバッファを受信すると CAN 割込みペンディングレジスタ(INTPND)の IntPnd ビットに"1"をセットします。そのとき、そのメッセージオブジェクトの TxRqst ビットは"0"にクリアされます。これは、リモートフレーム送信処理中に、要求データフレームを受信した場合、送信処理を防ぐために行われます。

42.5.3.4 リモートフレーム

リモートフレームについて示します。

リモートフレーム受信時の動作は、下記の 3 つの処理があります。一致するメッセージオブジェクトの設定より、リモートフレーム受信時の処理が選択されます。

1. Dir="1"(送信方向), RmtEn="1", UMask="1"または"0"
   一致したリモートフレームの受信を行い、このメッセージオブジェクトの TxRqst ビットのみ"1"にセットされ、リモートフレームに対するデータフレームの自動返信(送信)を行います。(TxRqst ビット以外のメッセージオブジェクトは変更されません。)
2. Dir="1"(送信方向), RmtEn="0", UMask="0"
   受信したリモートフレームがメッセージオブジェクトと一致しても受信しないでリモートフレームを無効にします。(このメッセージオブジェクトの TxRqst ビットは変更されません。)
3. Dir="1"(送信方向), RmtEn="0", UMask="1"
   受信したリモートフレームがメッセージオブジェクトと一致した場合、このメッセージオブジェクトの TxRqst ビットは"0"にリセットされ、リモートフレームは受信データフレームのように処理されます。受信したアービトレーションフィールドとコントロールフィールド(ID + IDE + RTR + DLC)は、メッセージ RAM のメッセージオブジェクトへ格納され、このメッセージオブジェクトの NewDat ビットが"1"にセットされます。メッセージオブジェクトのデータフィールドは変更されません。

42.5.3.5 受信メッセージオブジェクトの設定

受信メッセージオブジェクトの設定について示します。

以下に受信メッセージオブジェクトの初期化方法を示します。

表 42-12 受信メッセージオブジェクトの初期化

MsgVal	Arb	Data	Mask	EoB	Dir	NewDat	MsgLst	RxIE	TxIE	IntPnd	RmtEn	TxRqst
1	appl.	appl.	appl.	1	0	0	0	appl.	0	0	0	0

IFx アービトレーションレジスタ(ID28-0, Xtd ビット)は、アプリケーションによって与えられ、受容フィルタに用いられる受信メッセージ ID とメッセージの種類を定義します。

標準フレーム(11 ビット ID)を設定した場合は、ID28～ID18 を使用し、ID17～ID0 は無効となります。また、標準フレームが受信されると、ID17～ID0 は"0"にリセットされます。拡張フレーム(29 ビット ID)を設定した場合は、ID28～ID0 を使用します。

RxIE ビットが"1"にセットされた場合、受信データフレームがメッセージオブジェクトへ格納されると IntPnd ビットが"1"にセットされます。

データ長コード(DLC3-0)は、アプリケーションによって与えられます。CAN コントローラが、受信データフレームをメッセージオブジェクトへ格納するとき、受信データ長コードと 8 バイトのデータを格納します。データ長コードが 8 未満の場合は、メッセージオブジェクトの残りのデータは不定データが書き込まれます。

Umask="1"のとき、IFx マスクレジスタ(Msk28-0, UMask, MXtd, MDir ビット)は、マスク設定によりグループ化された ID を持つデータフレームの受信を許可するために使用します。詳細は、「5.3 メッセージ受信動作」のデータフレーム受信を参照してください。

＜注意事項＞

IFx マスクレジスタの Dir ビットのマスク設定は禁止です。

42.5.3.6 受信メッセージの処理

受信メッセージの処理について説明します。

CPU は、メッセージインタフェースレジスタを介して、受信メッセージをいつでも読み出すことが可能です。

通常、IFx コマンドマスクレジスタ(IFxCMSK)に"007F$_H$"を書き込みます。次にメッセージオブジェクトのメッセージ番号を IFx コマンド要求レジスタ(IFxCREQ)に書き込みます。この手順によって、指定されたメッセージ番号の受信メッセージをメッセージ RAM からメッセージインタフェースレジスタに転送します。このとき、IFx コマンドマスクレジスタ(IFxCMSK)の設定により、メッセージオブジェクトの NewDat ビットと IntPnd ビットを"0"にクリアすることが可能です。

受信メッセージの処理は、受容フィルタにより一致した場合、メッセージを受信します。メッセージオブジェクトで受容フィルタのマスクを使用している場合は、マスク設定されたデータが受容フィルタから除外され、メッセージを受信するか判断します。

NewDat ビットは、メッセージオブジェクトが最後に読み出されてから、新しいメッセージが受信されたかを示します。

MsgLst ビットは、受信したデータがメッセージオブジェクトから読み出されない状態で次の受信データを受信したために前のデータを失ってしまったことを示します。MsgLst ビットは自動的にリセットされません。

リモートフレーム送信処理中に、受容フィルタにより一致するデータフレームが受信された場合には、TxRqst ビットは自動的に"0"にクリアされます。

42.5.4 FIFO バッファ機能

FIFO バッファ機能について示します。

受信メッセージ処理におけるメッセージオブジェクトの FIFO バッファの構成および動作について説明します。

42.5.4.1 FIFO バッファの構成

FIFO バッファの構成について説明します。

FIFO バッファの受信メッセージオブジェクトの構成は、EoB ビットを除いて、受信メッセージオブジェクトの構成と同じです。 (「42.5.3 メッセージ受信動作」の受信メッセージオブジェクトの設定を参照してください)

FIFO バッファは、2 つ以上の受信メッセージオブジェクトを連結して使用します。この FIFO バッファへ受信メッセージを格納するためには、受信メッセージオブジェクトの ID とマスクを使用する場合はそれらの設定を一致させなければなりません。

FIFO バッファの最初の受信メッセージオブジェクトは、優先順位の高いメッセージ番号の小さい番号となります。FIFO バッファの最後の受信メッセージオブジェクトは、EoB ビットに"1"をセットし、FIFO バッファブロックの終わりを示す必要があります。(FIFO バッファ構成を使用するメッセージオブジェクトの最終メッセージオブジェクト以外は、EoB ビットに"0"を設定してください。)

＜注意事項＞

- FIFO バッファで使用するメッセージオブジェクトの ID とマスク設定は必ず同じ設定にしてください。
- FIFO バッファを使用しない場合は、必ず EoB ビットに"1"を設定してください。

42.5.4.2 FIFO バッファによるメッセージ受信

FIFO バッファによるメッセージ受信について説明します。

受信メッセージが、FIFO バッファの ID と一致した場合は、最小メッセージ番号の FIFO バッファの受信メッセージオブジェクトへ格納されます。

FIFO バッファの受信メッセージオブジェクトへメッセージが格納されると、この受信メッセージオブジェクトの NewDat ビットが"1"にセットされます。EoB ビットが"0"の受信メッセージオブジェクトへ NewDat ビットをセットすると、CAN コントローラによる FIFO バッファ書込みは、最後の受信メッセージオブジェクト(EoB="1")に到達するまで、受信メッセージオブジェクトは保護され、書込みが行われません。

最終 FIFO バッファまで有効なデータが格納された状態で受信メッセージオブジェクトの NewDat ビットに"0"書込み(書込み保護の解除)が行われないと次に受信されたメッセージが最終メッセージオブジェクトへ書き込まれ、メッセージは上書きされてしまいます。

42.5.4.3 FIFO バッファからの読出し

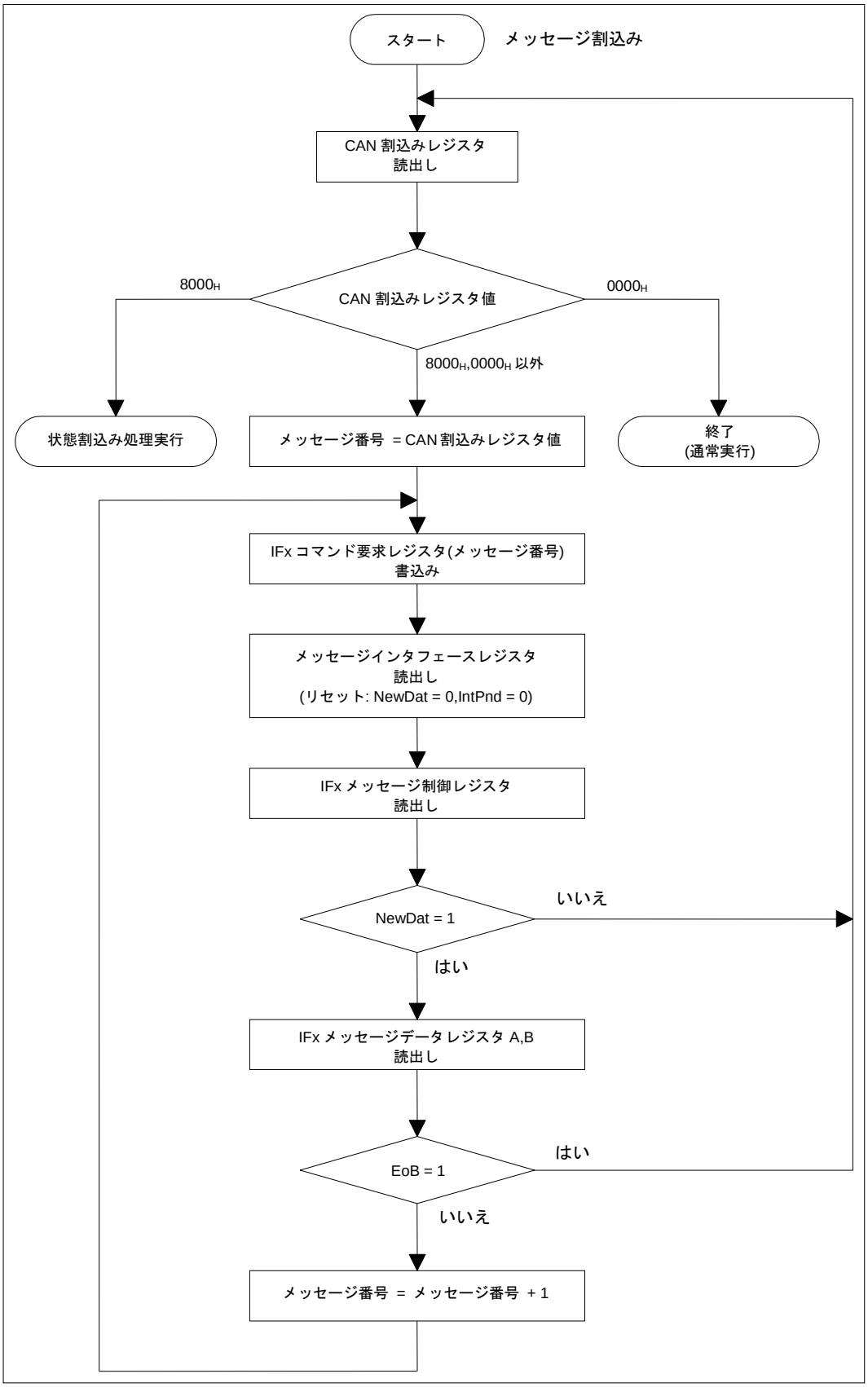
FIFO バッファからの読出しについて説明します。

CPUが受信メッセージオブジェクトの内容を読み出すには、IFx コマンド要求レジスタ(IFxCREQ)へ受信メッセージ番号を書き込むことによって、メッセージインタフェースレジスタに転送され読み出すことができます。このとき、IFx コマンドマスクレジスタ(IFxCMSK)の WR/RD を"0"(リード)および、TxRqst/NewDat=1, IntPnd=1 に設定し、NewDat ビットと IntPnd ビットを"0"にリセットしてください。

FIFO バッファの機能を保証するために、FIFO バッファの受信メッセージオブジェクトは、必ず最小のメッセージ番号から読み出してください。

以下に FIFO バッファで連結されるメッセージオブジェクトの CPU の処理方法を示します。

図 42-3 FIFO バッファの CPU 処理



42.5.5 割込み機能

割込み機能について説明します。

ステータス割込み(IntId=8000_H)およびメッセージ割込み(IntId=メッセージ番号)による割込み処理について説明します。

複数の割込みが保留中である場合、CAN 割込みレジスタ(INTR)は、保留中の最高優先度の割込みコードを示します。割込みコードの設定された時間順は無視され、常に優先順位の高い割込みコードが表示されます。CPU がクリアするまで割込みコードは保持されます。
ステータス割込み(IntId ビットの 8000_H)は、最高優先度となります。
メッセージ割込みの優先度は、メッセージ番号の小さいメッセージが高く、大きいメッセージが低くなります。
メッセージ割込みは、メッセージオブジェクトの IntPnd ビットのクリアによってクリアされます。ステータス割込みは、CAN ステータスレジスタ(STATR)の読出しでクリアされます。
CAN 割込みペンディングレジスタ(INTPND)の IntPnd ビットは、割込みの有無を示します。保留中の割込みがないときは、IntPnd ビットは"0"を示します。
CAN 制御レジスタ(CTRLR)の IE ビットおよび、IFx メッセージ制御レジスタ(IFxMCTR)の TxIE ビット, RxIE ビットに"1"をセットしている状態で、IntPnd ビットが"1"になると CPU への割込み信号がアクティブになります。
割込み信号は、CAN 割込みペンディングレジスタ(INTPND)が"0"にクリアされる(割込み要因リセット)か、CAN 制御レジスタ(CTRLR)の IE ビットが"0"にリセットされるまでアクティブ状態を保持します。
CAN 割込みレジスタ(INTR)の 8000_H は、CAN コントローラによって CAN ステータスレジスタ(STATR)が更新されたことを示し、この割込みが最高優先度となります。CAN ステータスレジスタ(STATR)の更新による割込みは、CAN 制御レジスタ(CTRLR)の EIE ビットと SIE ビットにより CAN 割込みレジスタ(INTR)への設定許可または禁止の制御ができます。また、CPU への割込み信号の制御は、CAN 制御レジスタ(CTRLR)の IE ビットで行うことができます。
CAN ステータスレジスタ(STATR)の RxOk ビット, TxOk ビット, LEC ビットは、CPU からの書込みにより更新(リセット)することができますが、その書込みにより割込みのセットまたははリセットを行うことはできません。
CAN 割込みレジスタ(INTR)の 8000_H, 0000_H 以外は、メッセージ割込みが保留中であることを示し、優先度の高い保留中のメッセージ割込みを示します。

CAN 割込みレジスタ(INTR)は、IE がリセットされた場合でも更新されます。

CPU へのメッセージ割込みの原因は、CAN 割込みレジスタ(INTR)または CAN 割込みペンディングレジスタ(INTPND)で確認することができます。(「42.4.5 メッセージハンドラレジスタ 」参照)メッセージ割込みをクリアする場合、同時にメッセージデータを読み出すことは可能であり、CAN 割込みレジスタ(INTR)で示されているメッセージ割込みをクリアすると次に優先度の高い割込みが CAN 割込みレジスタ(INTR)にセットされ、次の割込み処理を待つことになります。割込みがない場合には、CAN 割込みレジスタ(INTR)は 0000_H を示します。

＜注意事項＞

- ステータス割込み(IntId=8000_H)は、CAN ステータスレジスタ(STATR)の読出しアクセスにより割込みクリアされます。
- CAN ステータスレジスタ(STATR)の書込みアクセスによる、ステータス割込み(IntId=8000_H)は発生しません。

42.5.6 ビットタイミング・CAN システムクロック (fsys) 生成

ビットタイミング・CAN システムクロック(fsys)生成について説明します。

ビットタイミングについての概要と CAN コントローラにおけるビットタイミングについて説明します。

CAN ネットワークの各 CAN ノードは、それぞれクロック発振器(通常は水晶発振器)を持っています。ビットタイムのタイムパラメータは、CAN ノードごとに個別に構成できます。CAN ノードの発振周期(fosc)が異なっていても、共通のビットレートを作り出せます。
これらの発振器の周波数は、温度や電圧の変化、コンポーネントの悪化により少し異なります。その変動が発振器の許容範囲(df)内である限りは、CAN ノードはビットストリームへ再同期化することで異なるビットレートを補償できます。
CAN 仕様に応じて、ビットタイムは 4 つの区分に分けられ(図 5-4 ビットタイミング参照), 同期化部(Sync_Seg), 伝送時間部(Prop_Seg), フェイズバッファ部 1 (Phase_Seg1), フェイズバッファ部 2 (Phase_Seg2)で構成されます。それぞれの区分は、プログラマブルな時間量(表 5-3 CAN ビットタイムのパラメータ参照)から成ります。ビットタイムの基本単位時間(tq)は、CAN のシステムクロック(fsys)とボーレートプリスケーラ(BRP)で定義されます。

$$tq = BRP / fsys$$

CAN システムクロック(fsys)は、以下の図により生成されます。同期化部の Sync_Seg は、CAN バスのエッジを期待するビットタイム内のタイミングとなります。伝送時間部の Prop_Seg は、CAN ネットワーク内の物理的遅延時間を補償します。フェイズバッファ部の Phase_Seg1, Phase_Seg2 は、サンプリングポイントを指定します。再同期化ジャンプ幅(SJW)は、エッジフェーズエラーを補償するために再同期化時のサンプリングポイントの移動幅を定義します。

図 42-4 CAN システムクロック(fsys)生成図

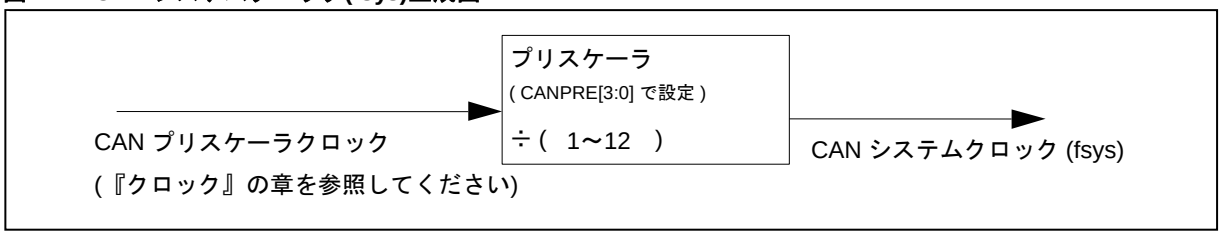


図 42-5 ビットタイミング

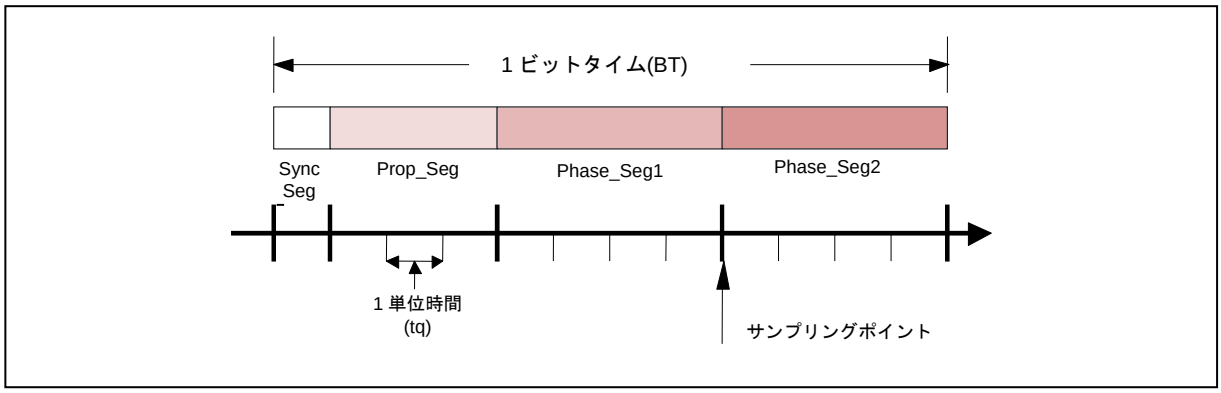


表 42-13 CAN ビットタイムのパラメータ

パラメータ	レンジ	機能
BRP	[1-32]	時間量の長さ tq の定義
Sync_Seg	1 tq	固定長　システムクロックへの同期化
Prop_Seg	[1-8] tq	物理遅延時間の補償
Phase_Seg1	[1-8] tq	サンプルポイント前のエッジフェーズエラー保証 同期化により一時的に長くなることがあります
Phase_Seg2	[1-8] tq	サンプルポイント後のエッジフェーズエラー保証 同期化により一時的に短くなることがあります
SJW	[1-4] tq	再同期化ジャンプ幅 どちらかのフェイズバッファ部より長くなることはありません。

次に CAN コントローラにおけるビットタイミングを示します。

図 42-6 CAN コントローラにおけるビットタイミング

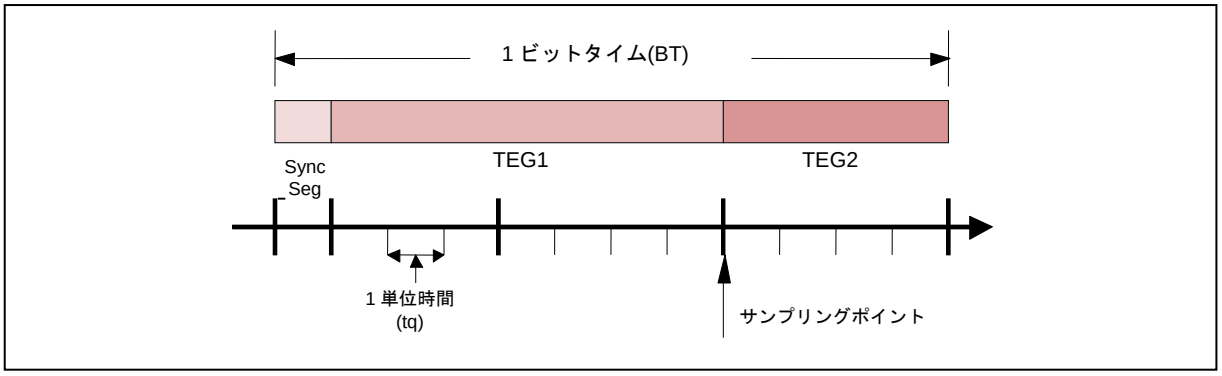


表 42-14 CAN コントローラのパラメータ

パラメータ	レンジ	機能
BRPE,BRP	[0-1023]	時間量の長さ tq の定義 ビットタイミングレジスタおよびプレスケーラ拡張レジスタにより最大 1024 までプリスケーラを拡張できます
Sync_Seg	1 tq	システムクロックへの同期化 固定長
TSEG1	[1-15] tq	サンプリングポイント前のタイムセグメントです Prop_Seg と Phase_Seg1 に相当します ビットタイミングレジスタにより制御可能です
TSEG2	[0-7] tq	サンプリングポイント後のタイムセグメントです Phase_Seg2 に相当します ビットタイミングレジスタにより制御可能です
SJW	[0-3] tq	再同期化ジャンプ幅です ビットタイミングレジスタにより制御可能です

各パラメータの関係を以下に示します。

tq=([BRPE, BRP]+1) / fsys

BT=SYNC_SEG + TEG1 + TEG2

=(1 + (TSEG1 + 1) + (TSEG2 + 1)) ×tq

=(3 + TSEG1 + TSEG2) × tq

42.5.7 テストモード

テストモードについて示します。

テストモードの設定方法および動作について説明します。

42.5.7.1 テストモード設定

テストモード設定について示します。

CAN 制御レジスタ(CTRLR)の Test ビットに"1"をセットすることでテストモードなります。テストモードに設定すると、CAN テストレジスタ(TESTR)の Tx1, Tx0, LBack, Silent, Basic ビットが有効となります。

CAN 制御レジスタ(CTRLR)の Test ビットを"0"にリセットすることにより、すべてのテストレジスタ機能を無効にします。

42.5.7.2 サイレントモード

サイレントモードについて説明します。

CANテストレジスタ(TESTR)の Silent ビットを"1"にセットすることにより、CAN コントローラをサイレントモードに設定できます。

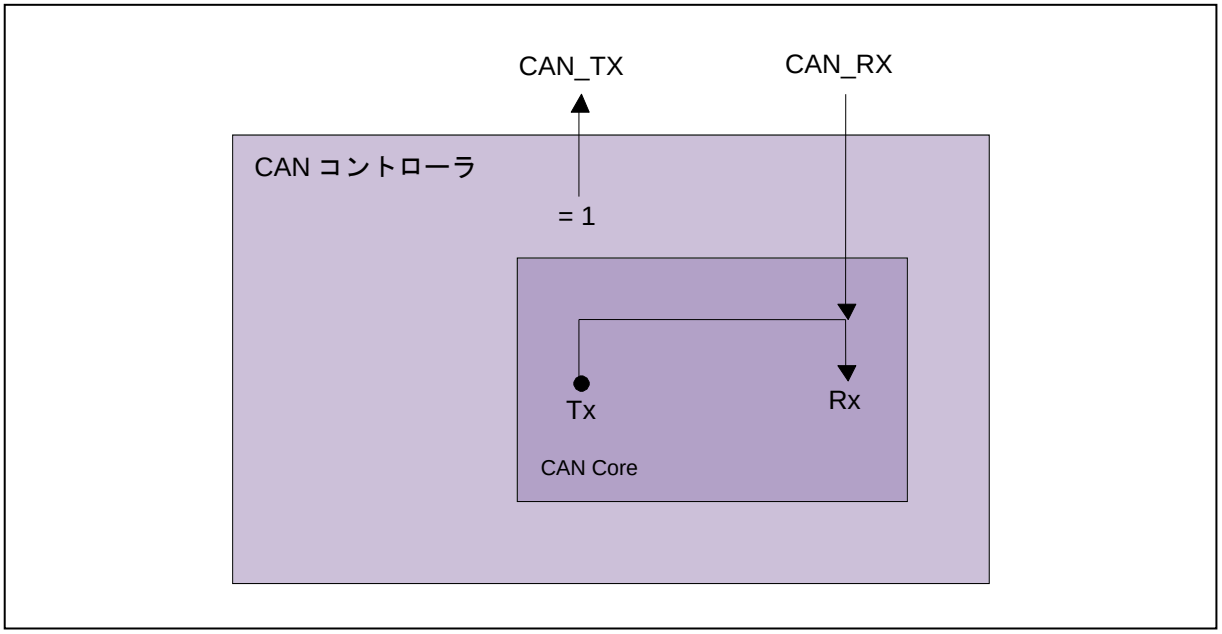
サイレントモードでは、データフレームとリモートフレームを受信可能であるが、CAN バス上にはレセッシブのみ出力し、メッセージおよび ACK の送信を行いません。

CAN コントローラがドミナントビット(ACK, オーバロードフラグ, アクティブエラーフラグ)の送信を要求された場合、CAN コントローラ内部の折り返し回路で RX 側に送信されます。この動作では、CAN バス上においてレセッシブ状態であっても、受信側では CAN コントローラ内部で折り返し送信されたドミナントビットを受信することになります。

サイレントモードでは、ドミナントビット(ACK, エラーフラグ)送信による影響がない状態で、CAN バス上のトラフィック解析ができます。

以下にサイレントモードでの信号 CAN_TX と CAN_RX の CAN コントローラへの接続を示します。

図 42-7 サイレントモードでの CAN コントローラ



42.5.7.3 ループバックモード

ループバックモードについて説明します。

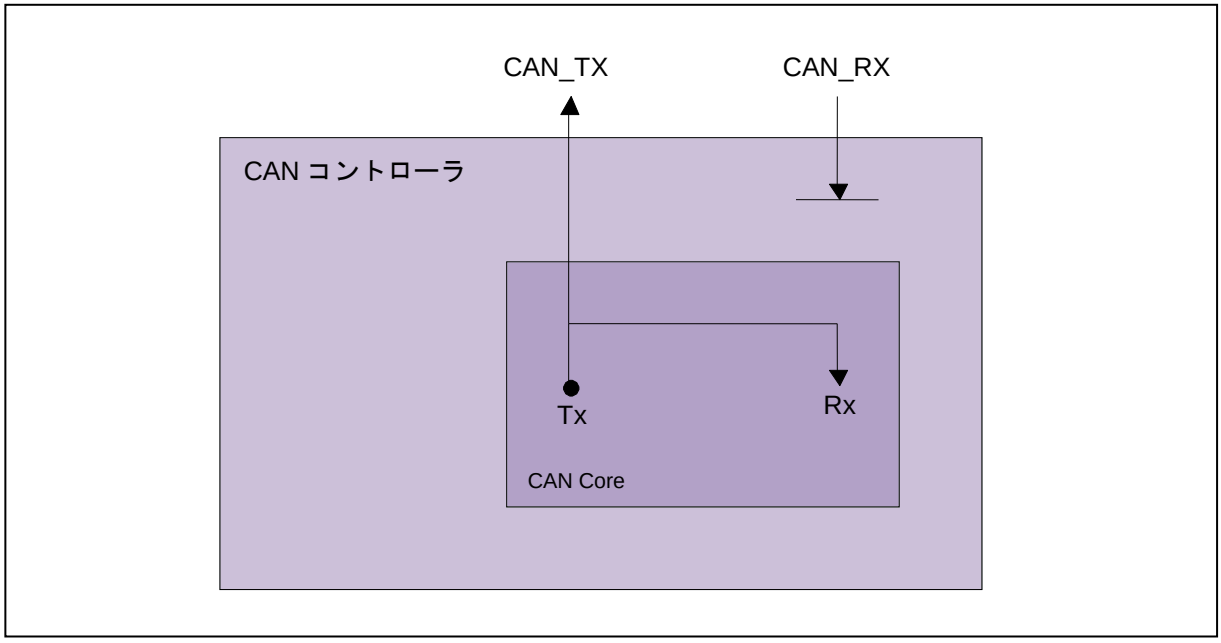
CAN テストレジスタ(TESTR)の LBack ビットを"1"にセットすることにより、CAN コントローラをループバックモードに設定できます。

ループバックモードは、自己診断機能に使用できます。

ループバックモードでは、CAN コントローラ内部で TX 側と RX 側が接続され、CAN コントローラが送信したメッセージを、RX 側で受信したメッセージとして扱い、受容フィルタを通過したメッセージは、受信バッファに格納します。

以下にループバックモードでの信号 CAN_TX と CAN_RX の CAN コントローラへの接続を示します。

図 42-8 ループバックモードの CAN コントローラ



＜注意事項＞

外部信号から独立するため、データ/リモートフレームのアクノリッジスロットでのドミナントビットはサンプリングされません。そのため通常、CAN コントローラはアクノリッジエラーを発生しますが、テストモードではアクノリッジエラーを発生しないようにしています。

42.5.7.4 サイレントモードとループバックモードの結合

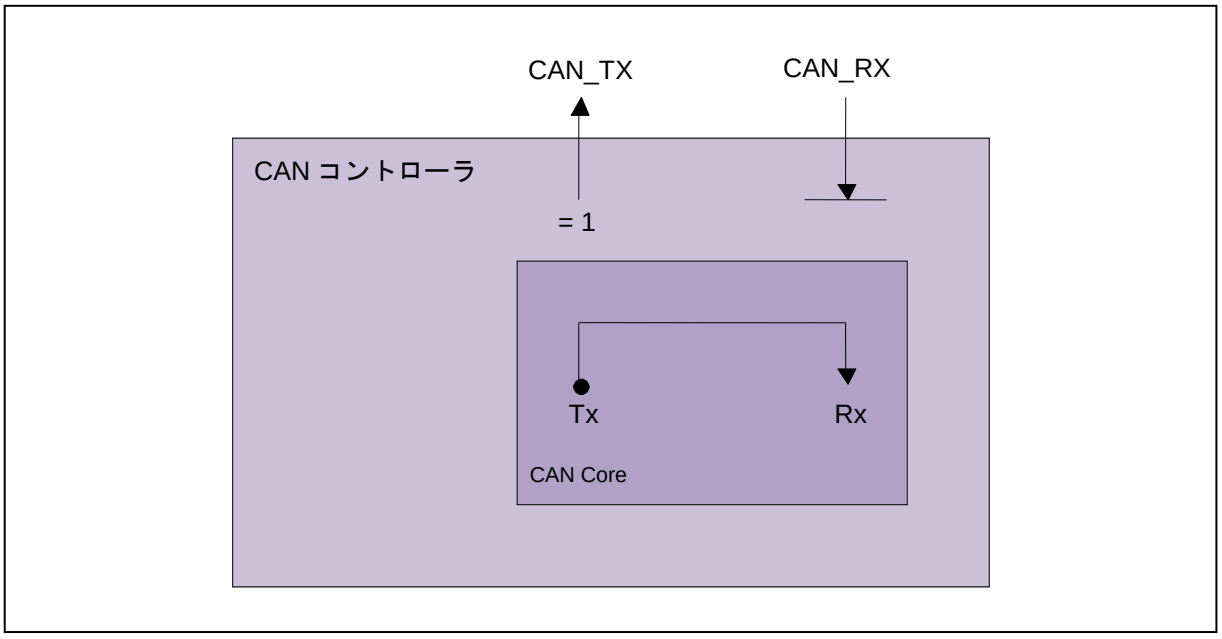
サイレントモードとループバックモードの結合について説明します。

CAN テストレジスタ(TESTR)の LBack ビットと Silent ビットを同時に"1"にセットすることにより、ループバックモードとサイレントモードを結合させ動作することが可能です。

このモードは、「ホットセルフテスト」用に使用できます。「ホットセルフテスト」とは、CAN コントローラがループバックモードでテストしたときに、CAN_TX 端子にはレセッシブの固定出力、CAN_RX 端子からの入力は無効となりますので、CAN システムの動作に影響がないことを意味しています。

以下にサイレントモードとループバックモードの結合したときの信号 CAN_TX と CAN_RX の CAN コントローラへの接続を示します。

図 42-9 サイレントモードとループバックモードの結合された CAN コントローラ



42.5.7.5 ベーシックモード

ベーシックモードについて説明します。

CANテストレジスタ(TESTR)のBasicビットを"1"にセットすることにより、CANコントローラをベーシックモードに設定できます。

ベーシックモードでは、CAN コントローラは、メッセージ RAM を使用せずに動作します。

IF1 メッセージインタフェースレジスタは、送信制御用として使用されます。

メッセージ送信を行う場合、まず、IF1 メッセージインタフェースレジスタに送信する内容を設定します。次に、IF1 コマンド要求レジスタの BUSY ビットに"1"をセットすることで送信要求します。BUSY ビットが"1"にセットされている間、IF1 メッセージインタフェースレジスタをロックしている、または送信が保留されていることを示します。

BUSY ビットに"1"がセットされると CAN コントローラは以下の動作を行います。

CAN バスがバスアイドルになるとすぐに、IF1 メッセージインタフェースレジスタの内容を、送信用シフトレジスタへロードし、送信を開始します。正常に送信完了すると、BUSY ビットが"0"にリセットされ、ロックされていた IF1 メッセージインタフェースレジスタを開放します。

送信が保留されているときに、IF1 コマンド要求レジスタの BUSY ビットを"0"にリセットすることでいつでも中断できます。また、送信中に BUSY ビットを"0"にリセットすると、調停負けやエラーなどの場合に行われる再送信を停止します。

IF2 メッセージインタフェースレジスタは、受信制御用として使用されます。

メッセージの受信は、受容フィルタを使用せずにすべて受信します。IF2 コマンド要求レジスタの BUSY ビットを"1"にセットすることにより、受信したメッセージの内容を読み出すことが可能となります。

BUSY ビットに"1"がセットされると CAN コントローラは以下の動作を行います。

- 受信したメッセージ (受信用のシフトレジスタの内容) を受容フィルタなしで、IF2 メッセージインタフェースレジスタへ格納します。

新しいメッセージが IF2 メッセージインタフェースレジスタに格納された場合、CAN コントローラが NewDat ビットを"1"にセットします。また、NewDat ビットが"1"のときに、さらに新しいメッセージを受信した場合は、CAN コントローラが MsgLst を"1"にセットします。

＜注意事項＞

- ベーシックモードでは、制御/状態ビットに関係するすべてのメッセージオブジェクトと IFx コマンドマスクレジスタ(IFxCMSK)の制御モード設定は無効となります。
- コマンド要求レジスタのメッセージ番号は無効です。
- IF2 メッセージ制御レジスタの NewDat ビットと MsgLst ビットは通常時と同様に動作し、DLC3-0 は受信された DLC を示し、ほかの制御ビットは"0"として読み出されます。

42.5.7.6 端子 CAN_TX のソフトウェア制御

端子 CAN_TX のソフトフェア制御について説明します。

CAN 送信端子である CAN_TX は、4 つの出力機能があります。

- シリアルデータ出力(通常出力)
- CAN コントローラのビットタイミングをモニタするための、CAN サンプリングポイント信号出力
- ドミナント固定出力
- レセッシブ固定出力

ドミナントおよびレセッシブの固定出力は、CAN 受信端子の CAN_RX モニタ機能と共に、CAN バスの物理層の確認に使用できます。

CAN_TX 端子の出力モードは、CAN テストレジスタ(TESTR)の Tx1 と Tx0 ビットにより制御可能です。

＜注意事項＞
CAN メッセージ送信もしくはループバックモード, サイレントモード, ベーシックモードを使用する際は、CAN_TX をシリアルデータ出力に設定する必要があります。

42.5.8 ソフトウェア初期化

ソフトウェア初期化について説明します。

ソフトウェアによる初期化について説明します。

ソフトウェアでの初期化要因を以下に示します。

- ハードウェアリセット
- CAN 制御レジスタ(CTRLR)の Init ビットの設定
- バスオフ状態への遷移

ハードウェアによるリセットは、メッセージ RAM (MsgVal, NewDat, IntPnd, TxRqst ビットを除く)以外すべて初期化されます。メッセージ RAM はハードウェアによるリセット後、CPU によって初期化するかメッセージ RAM の MsgVal を"0"にしてください。また、ビットタイミングレジスタを設定する場合には、CAN 制御レジスタ(CTRLR)の Init ビットを"0"にクリアする前に設定してください。

CAN 制御レジスタ(CTRLR)の Init ビットは、以下の条件で"1"にセットされます。

- CPU からの"1"書込み
- ハードウェアリセット
- バスオフ

Init ビットが"1"にセットされると、CAN バスの全メッセージ送受信は停止され、CAN バス出力の CAN_TX 端

子はレセッシブ出力となります。(CAN_TX テストモードは除く)

Init ビットに"1"をセットすると、エラーカウンタは変化しませんし、レジスタも変更されません。

CAN 制御レジスタ(CTRLR)の Init ビットと CCE ビットが"1"にセットされると、ボーレート制御用のビットタイミングレジスタとプリスケーラ拡張レジスタへの設定が可能となります。

Init ビットを"0"にリセットすることによりソフトウェア初期化を終了します。また、Init ビットを"0"にすることは、CPU からのアクセスでしか実行できません。

Init ビットが"0"にリセットされてから、連続した 11 ビットのレセッシブの発生 (=バスアイドル) を待つことにより、CAN バス上のデータ転送と同期化した後、メッセージの転送が行われます。

通常動作中にメッセージオブジェクトの Msk, ID, XTD, EoB, RmtEn を変更する場合、MsgVal を無効に設定してから変更してください。

42.6 制限事項

CAN の制限事項について説明します。

42.6.1 INIT ビット

INIT ビットについて示します。

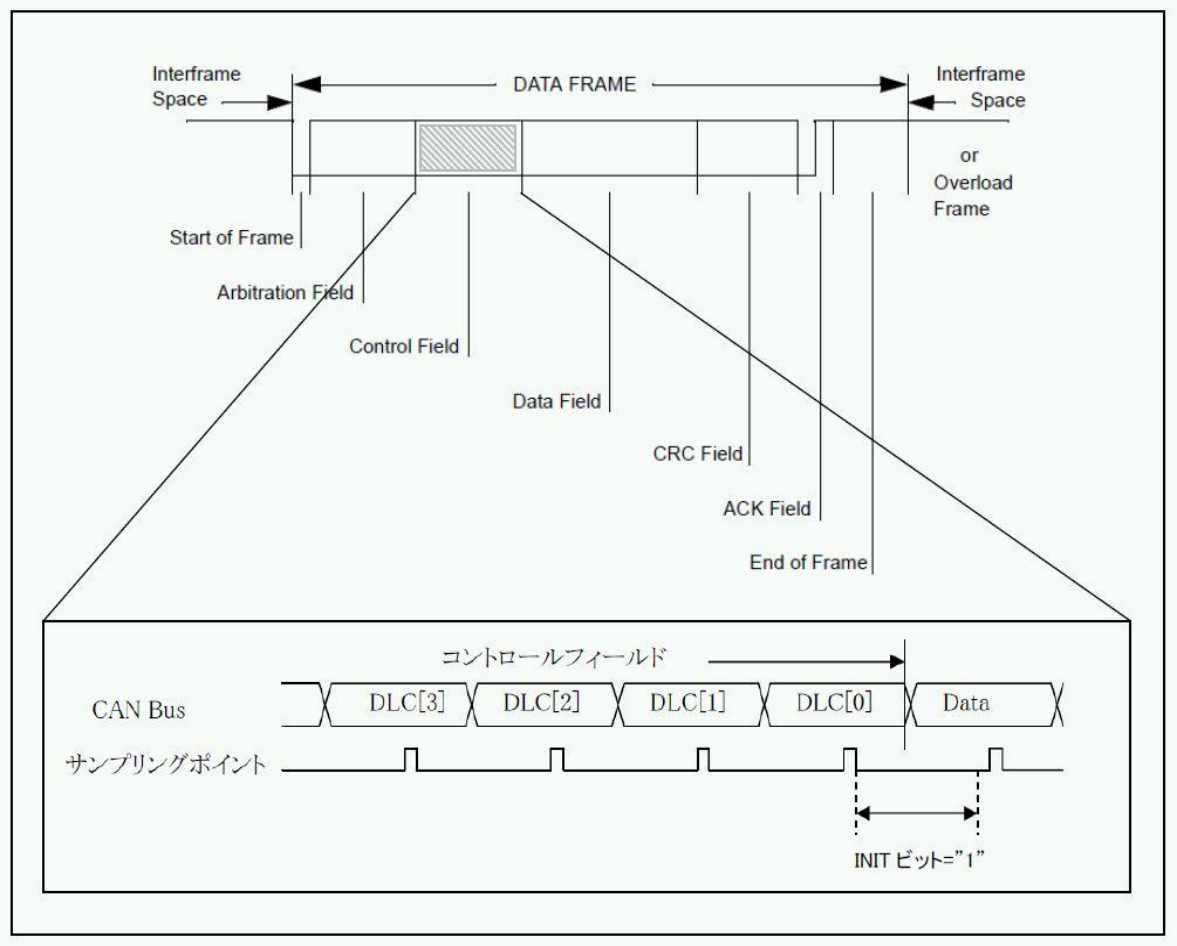
42.6.1.1. 制限事項

42.6.1.2. 回避方法

42.6.1.1 制限事項

制限事項について説明します。

コントロールフィールドの最終ビットを送信中に CAN コントロールレジスタ(CTRLR)の INIT ビット="1"を設定(図 6-1)すると、INIT ビットをクリアした後、最初に送信されるフレームのデータフィールドが 1 ビット分左へシフトされたものとなります。
なお、それ以降のメッセージは正しく送信されます。
また、データ長がゼロのリモートフレームとデータフレームは、本タイミングで INIT ビットを設定しても影響を受けません。



42.6.1.2 回避方法

回避方法について説明します。

以下のいずれかの方法で、本制限を回避してください。

1. CAN 制御レジスタ(CTRLR)の INIT ビットに"1"を設定する場合
   送信完了後すぐに CAN 制御レジスタ(CTRLR)の INIT ビットに"1"を設定してください。

2. 送信中に CAN 制御レジスタ(CTRLR)の INIT ビットに"1"を設定し、その後 INIT ビットに"0"を設定し送信したい場合
   まずは、INIT ビットに"1"を設定し、送信要求ビット(TxRqst)が"1"に設定されているメッセージバッファに対し送信キャンセル(TxRqst ビットに"0"を設定)を実行してから、INIT ビットに"0"を設定してください。
   その後、CAN の 2 ビット時間経過後、送信したメッセージバッファの送信要求ビット(TxRqst)に"1"を設定してください。

43. D/A コンバータ

D/A コンバータについて説明します。

43.1 概要

D/A コンバータの概要について説明します。

D/A コンバータは, デジタル信号をアナログ信号に変換する周辺機能です。本製品は 8 ビット D/A コンバータを 2 チャネル内蔵しています。

43.2 特長

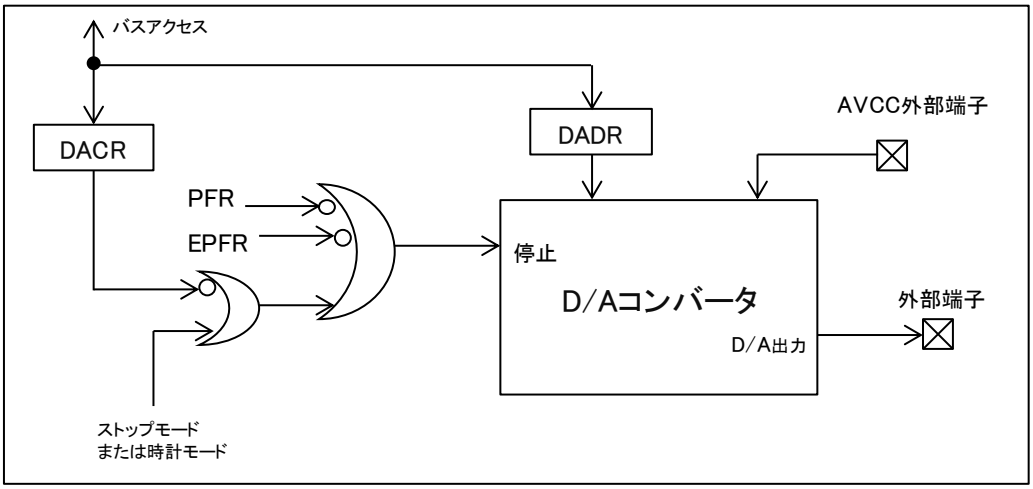
D/A コンバータの特長について説明します。

- パワーダウン機能
  D/A コンバータからの出力が禁止されているときに, 電力を落とすパワーダウン機能を内蔵しています。

43.3 構成

D/A コンバータの構成を示します。

図 43-1 ブロックダイヤグラム



43.4 レジスタ

D/A コンバータのレジスタについて説明します。

■ ベースアドレス (Base_addr)・外部端子表

チャネル番号	Base_addr	外部端子
0	0x023C	DAO0
1	0x023E	DAO1

■ レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x023C	DACR0	DADR0	DACR1	DADR1	DA コントロールレジスタ 0 DA データレジスタ 0 DA コントロールレジスタ 1 DA データレジスタ 1

43.4.1　DA コントロールレジスタ ：DACR (DA Control Register)

DA コントロールレジスタのビット構成について示します。

DAO 端子からの出力を許可するレジスタです。

■ DACR： アドレス Base_addr (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							DAE
初期値	-	-	-	-	-	-	-	0
属性	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	R/W

[bit0] DAE (DA Enable) : DA 出力許可ビット

0: 出力禁止

1: 出力許可

＜注意事項＞

ポート機能として D/A コンバータを選択しているときに、このビットを"0" に設定すると PFR/EPFR の設定にかかわらず、ポート機能としての D/A コンバータは無効になります。

また、ストップモードや時計モードに入ると、D/A コンバータの端子機能は、強制的に無効になります。

11 章 I/O ポートの 5.2 章に書かれているとおり、D/A コンバータの出力機能は最優先です。そのため、D/A コンバータ機能が無効になったときに、A/D, PPG, OCU, SG など D/A コンバータ以外の機能が有効になっていると、それらの機能が有効になるので注意してください。PFR/EPFR/ADCH で、D/A コンバータ以外のリソース機能が選択されていない場合は､汎用ポート機能となりますので PDR/DDR にて端子状態を選択することが可能です。

43.4.2　DA データレジスタ ：DADR (DA DATA Register)

DA データレジスタのビット構成について示します。

DAO 端子からの出力電圧を設定するレジスタです。このレジスタに格納された値を元に D/A コンバータからの出力電圧が算出されます。

■ DADR： アドレス Base_addr + 01H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	DA[7:0]							
初期値	X	X	X	X	X	X	X	X
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] DA[7:0] (DA) : DA 出力値

DA[7:0]	出力電圧
00000000	0/256 × AVCC
00000001	1/256 × AVCC
00000010	2/256 × AVCC
～	～
11111101	253/256 × AVCC
11111110	254/256 × AVCC
11111111	255/256 × AVCC

AVCC : AVCC 外部端子からの入力電圧

＜注意事項＞

このレジスタはリセットしても初期化されません。

43.5 動作説明

D/A コンバータの動作について説明します。

D/A コンバータは、D/A データレジスタ(DADR)に書き込まれた値を元に出力電圧を算出し、DAO 端子からアナログ電圧を出力します。

D/A データレジスタ(DADR)の DA7～DA0 ビットに値を書き込み、D/A コントロールレジスタ(DACR0)の DAE ビットに"1"を書き込むと、D/A コンバータからアナログ信号が出力されます。

ポート機能として D/A コンバータを選択しているときに、本ビットを"0"に設定すると PFR/EPFR の設定にかかわらず、ポート機能としての D/A コンバータは無効になります。また、ストップモードや時計モードに入ると、D/A コンバータの端子機能は、強制的に無効になります。11 章 I/O ポートの 5.2 章に書かれているとおり、D/A コンバータの出力機能は最優先です。そのため、D/A コンバータ機能が無効になったときに、A/D, PPG, OCU, SG など D/A コンバータ以外の機能が有効になっていると、それらの機能が有効になるので注意してください。PFR/EPFR/ADCH で、D/A コンバータ以外のリソース機能が選択されていない場合は、汎用ポート機能となりますので PDR/DDR にて端子状態を選択することが可能です。

D/A コンバータの変換結果の外部端子への出力は、D/A コントロールレジスタ(DACR)の DAE ビットのほかに PFR レジスタと EPFR レジスタにより端子を DA 出力に設定する必要があります。

43.6 注意事項

D/A コンバータの注意事項について説明します。

DA 出力値は、PDDR レジスタでは読み出せません。

44. A/D コンバータ

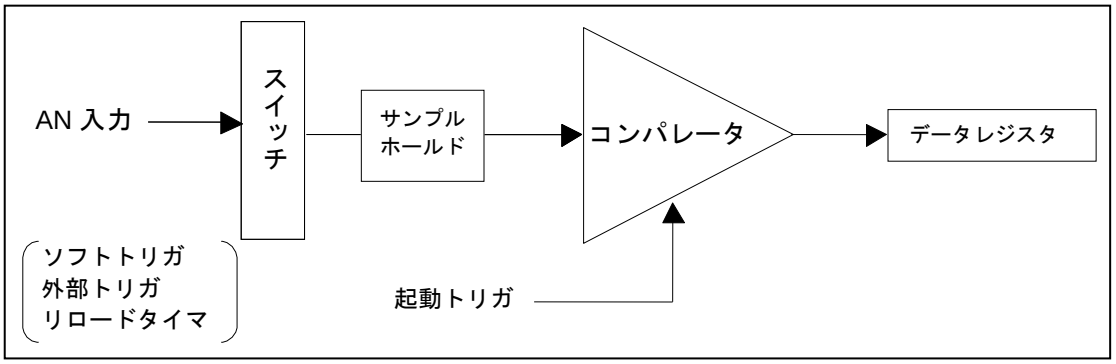
A/D コンバータについて説明します。

44.1 概要

A/D コンバータの概要について説明します。

A/D コンバータは、アナログ入力電圧をデジタル値に変換するものです。

変換モードには、単発変換モード，連続変換モード，停止変換モードがあり、各々変換動作としてシングル変換動作，スキャン変換動作があります。起動要因は各種トリガ (ソフトトリガ/外部トリガ/リロードタイマ) から選択できます。



44.2 特長

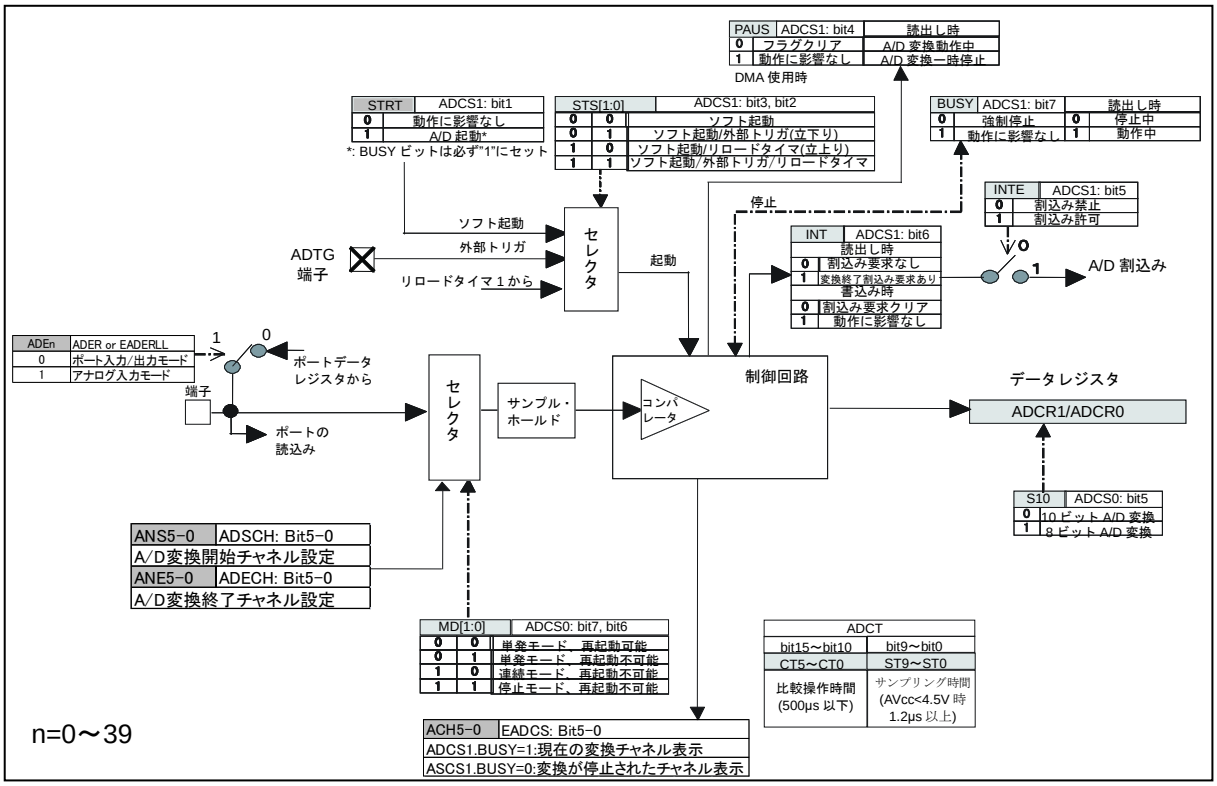
A/D コンバータの特長について説明します。

- 変換方式：サンプルホールド回路付き RC 型逐次比較変換方式
- 個数 ：1 (A/D コンバータ 入力 -- 40 チャネル: AN0～AN39）
- 変換時間 ：
  最小 3µs (サンプルホールド時間を含む)
  変換時間=サンプリング＋変換
- 分解能 ：8/10 ビット分解能
- 変換モード ：
  単発変換モード：指定チャネルを 1 サイクル変換
  連続変換モード：指定チャネルを繰り返し変換
  停止変換モード：指定チャネルを変換したら一時停止して次の起動が掛かるまで待機 (変換開始の同期が可能)
- 変換動作 ： 上記変換モードの変換動作として以下があります。
  シングル変換動作 ：1 チャネルを選択変換
  スキャン変換動作 ： 連続した複数のチャネルを変換。最大 40 チャネルプログラム可能
- 起動要因 ：
  ソフトトリガ (ADCS1:STRT)
  外部トリガ, 立下り (ADTG 端子)
  リロードタイマ, 立上り(リロードタイマ 1)
- 割込み要求 ：
  A/D 変換終了時には、CPU に対して A/D 変換終了の割込み要求を発生
- 割込み ： 変換終了割込み
- 機能停止 ：A/D 変換動作の強制停止が可能

44.3 構成

A/D コンバータの構成について説明します。

図 44-1 構成図



44.4 レジスタ

A/D コンバータのレジスタについて説明します。

表 44-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x00A0	ADER				アナログ入力許可レジスタ
0x00A4	ADCS1	ADCS0	ADCR1	ADCR0	A/D 制御ステータスレジスタ上位 A/D 制御ステータスレジスタ下位 データレジスタ上位 データレジスタ下位
0x00A8	ADCT		ADSCH	ADECH	変換時間設定レジスタ A/D 開始チャネル設定レジスタ A/D 終了チャネル設定レジスタ
0x00AC	予約	EADERLL	EADCS	予約	拡張アナログ入力許可レジスタ(ch.39～ch.32) 拡張 A/D 制御ステータスレジスタ

44.4.1　アナログ入力許可レジスタ ：ADER

アナログ入力許可レジスタのビット構成について説明します。

使用する端子をアナログ入力に設定します。(ch.31～ch.0)

■ ADERH：アドレス 00A0H (アクセス：バイト，ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	ADE31	ADE30	ADE29	ADE28	ADE27	ADE26	ADE25	ADE24
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ADE23	ADE22	ADE21	ADE20	ADE19	ADE18	ADE17	ADE16
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ ADERL：アドレス 00A2H (アクセス：バイト，ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	ADE15	ADE14	ADE13	ADE12	ADE11	ADE10	ADE9	ADE8
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit0] ADE[31:0] : アナログ入力許可

ADEn	意味
0	ポート入力/出力モード
1	アナログ入力モード

n=0～31

開始チャネル、終了チャネルのアナログ入力許可レジスタ(ADER)には必ず"1" を設定してください。

44.4.2 A/D 制御ステータスレジスタ (上位) : ADCS1

A/D 制御ステータスレジスタ (上位) のビット構成について説明します。

A/D コンバータの制御およびステータス表示を行うレジスタです。

■ ADCS1：アドレス 00A4H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	BUSY	INT	INTE	PAUS	STS1	STS0	STRT	予約
初期値	0	0	0	0	0	0	0	0
属性	R (RM1),W	R (RM1),W	R/W	R,W	R/W	R/W	R (RM0),W	R0,W0

[bit7] BUSY : 強制停止指示ビット/動作確認ビット

BUSY	読出し時	書込み時
0	A/D コンバータは停止中	A/D コンバータを強制停止します
1	A/D コンバータは動作中	動作に影響なし

- A/D 変換起動でセットされます。
- 単発モードでは、最終チャネルの変換終了でクリアされます。
- 連続，停止モード時は、"0"を書込みで停止するまでクリアされません。
- 強制終了とソフト起動を同時に行わないでください（BUSY=0, STRT=1）。
- ソフトトリガが起動(STRT ビットに"1"書込み）を行うとき、強制停止指示ビットも必ず"1"を書き込んでください。（同時に"1"にしないと起動がかかりません）

[bit6] INT : A/D 変換終了フラグ/割込み要求

INT	読出し時	書込み時
0	割込み要求なし	フラグのクリア
1	割込み要求あり (A/D 変換終了，すべてスキャン変換終了)	動作に影響なし

＜注意事項＞

"0"書込みクリアは A/D 停止中に行ってください。

[bit5] INTE : A/D 割込み要求許可

INTE	意味
0	割込み要求禁止
1	割込み要求許可

A/D 割込み要求許可ビット(INTE)と、A/D 割込み要求フラグ(INT)が"1"のとき、割込みが発生します。

[bit4] PAUS : A/D 一時停止フラグ

PAUS	読出し時	書込み時
0	A/D 変換動作中	フラグのクリア
1	A/D 変換動作一時停止	動作に影響なし

- A/D 変換結果を格納するレジスタが 1 つのため連続で変換した場合、変換結果を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転送しなければ次の変換データが格納されないようになっています。この間 A/D 変換動作は停止します。転送終了後、INT ビットをクリアすると A/D は変換を再開します。
- クリアは"0"書込みでのみ可能です。(転送終了ではクリアされません。)
- INT ビットに"1"がセットされた状態のまま次の A/D 変換を実行すると PAUS ビットに"1"がセットされます。(前の A/D 変換の Data を保護するため)そのため、 1 つのチャネルの A/D 変換後、次の A/D 変換前に必ず INT ビットをクリアする必要があります。
- 変換データの保護機能に関しては「44.5.3. 変換モード」を参照してください。

[bit3, bit2] STS1, STS0 : A/D 変換の起動要因の選択

STS1	STS0	起動要因
0	0	ソフトトリガ
0	1	外部トリガ (立下り) または ソフトトリガ
1	0	リロードタイマ出力 (立上り) または ソフトトリガ
1	1	外部トリガ (立下り) 、リロードタイマ出力 (立上り) または ソフトトリガ

- 起動要因を複数指定している場合は最初にきた要因で起動します。
- A/D 変換中に発生した起動要因は単発変換モード(ADCS0:MD[1:0]="00")と停止変換モード(ADCS0:MD[1:0]="11")の変換再開時に有効です。
- 単発変換モード(ADCS0:MD[1:0]="01"), 連続変換モード, 停止変換モードで再起動する場合は A/D 変換動作をいったん停止(BUSY="0")させてから行ってください。
- 起動要因の設定は書き換えられると直ちに変わりますので、A/D 変換中に書き換えるときには注意してください。
- 外部端子トリガは立下りエッジを検出します。外部トリガ入力レベルが"L"のときにビットを書き換えて外部端子トリガ起動に設定すると A/D が起動する場合があります。
- タイマ選択時は、16 ビットリロードタイマ 1 が選択されます。

[bit1] STRT : A/D 変換のソフトトリガ

STRT	機能
0	動作に影響なし
1	A/D コンバータを起動します(ソフトトリガ)

- ソフトトリガ起動をするときは、強制停止指示ビット (BUSY)も必ず"1"にしてください。もしも、強制停止指示ビット(BUSY)を同時に"0"にした場合 A/D は起動しません。
- 再起動をかけるときは､BUSY に"0"を書き込んで強制停止をおこなった後､再び"1"を書き込んでください。
- 連続モード時および停止モード時は動作機能上、再起動はかかりません。"1"を書き込む前に BUSY ビットをチェックしてください。(BUSY ビットをクリアしてから起動してください。)
- 強制終了とソフト起動を同時に行わないでください(BUSY=0, STRT=1)。

[bit0] 予約

- 必ず"0"を書き込んでください。

44.4.3 A/D 制御ステータスレジスタ (下位) : ADCS0

A/D 制御ステータスレジスタ (下位)のビット構成について説明します

A/D コンバータの制御およびステータスの確認を行うレジスタです。

＜注意事項＞
A/D 変換動作中に書き換えないようにしてください。

■ ADCS0 ：アドレス 00A5H (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	MD1	MD0	S10	予約				
初期値	0	0	0	-	-	-	-	-
属性	R/W	R/W	R/W	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX

[bit7, bit6] MD1, MD0 : 動作モード設定

MD1	MD0	動作モード
0	0	単発変換モード、動作中の再起動はすべて可能
0	1	単発変換モード、動作中の再起動不可能
1	0	連続変換モード、動作中の再起動不可能
1	1	停止変換モード、動作中の再起動不可能

- ■ 単発変換モード：
  ADSCH:ANS[5:0]の設定チャネルから ADECH:ANE[5:0]の設定チャネルまで A/D 変換を連続して行いすべてのチャネルの変換が終了したら停止します。
- ■ 連続変換モード：
  ADSCH:ANS[5:0]の設定チャネルから ADECH:ANE[5:0]の設定チャネルまで A/D 変換を繰り返し行います。
- ■ 停止変換モード：
  ADSCH:ANS[5:0]の設定チャネルから ADECH:ANE[5:0]の設定チャネルまで 1 つのチャネルごとに 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。
  - □ 連続変換モード，停止変換モードで A/D 変換を起動すると、BUSY ビットで強制終了するまで変換動作を続けます。
  - □ 強制終了は BUSY ビットに"0"を書き込むことにより行われます。
  - □ 強制終了後の起動時は ADSCH:ANS[5:0]の設定チャネルから変換されます。
  - □ 単発，連続，停止の各変換モードでの再起動不可能はタイマ，外部トリガ，ソフトすべての起動に適用されます。

[bit5] S10 : 分解能設定

S10	設定
0	10 ビット A/D 変換
1	8 ビット A/D 変換

8 ビット A/D 変換の結果は ADCR0 に格納されます。

[bit4～bit0] 予約
書込み、読出しともに効果はありません。

44.4.4　データレジスタ ：ADCR0,ADCR1

データレジスタのビット構成について説明します。

データレジスタ(ADCR0, ADCR1)は、変換の結果として生成されたデジタル値を格納するために使用されるものです。ADCR0 は下位 8 ビットを格納し、ADCR1 は変換結果の最上位の 2 ビットを格納します。

データレジスタの値を 1 回の変換終了ごとに更新します。

データレジスタには、通常、最終変換値を格納しています。

■ ADCR1：アドレス 00A6H (アクセス：バイト，ハーフワード，ワード)

■ ADCR0：アドレス 00A7H (アクセス：バイト，ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約						D9	D8
初期値	—	—	—	—	—	—	X	X
属性	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	D7	D6	D5	D4	D3	D2	D1	D0
初期値	X	X	X	X	X	X	X	X
属性	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

変換データの保護機能を利用することができます。「44.5.3. 変換モード」を参照してください。

＜注意事項＞

ADCR1(0x0000A6)と ADCR0(0x0000A7)にバイトアクセスを行う場合は、ADCR1 へアクセス後、ADCR0 へアクセスを行ってください。変換結果の上書きが ADCR0 へのリードを待って行われるため、ADCR0、ADCR1 の順でアクセスを行うと、ADCR1 レジスタの値を読み出す前に、変換結果が上書きされる可能性があります。

＜注意事項＞

割込みクリアが先に行われ、変換結果のリードが行われていない状態では、次の変換が終了した場合、変換結果を上書きせずに変換結果がリードされるまで待ちます。リードを先に行った場合は、割込みクリアを行うまで変換結果の上書きを待ちます。

44.4.5　変換時間設定レジスタ ：ADCT

変換時間設定レジスタのビット構成について説明します。

サンプリング時間および比較時間を制御します。A/D の変換時間を設定するレジスタです。

＜注意事項＞
A/D 変換動作中に書き換えないようにしてください。

■ 推奨設定値

最適な変換時間を達成するために、以下の設定が推奨されます。(AVCC ≧ 4.5V)

周辺クロック(PCLK) (MHz)	比較操作時間 (CT5～CT0)	サンプリング時間 (ST9～ST0)	変換時間(μs)
16	000011　(03_H)	0000010110　(016_H)	2.125+1.375=3.500
24	000100　(04_H)	0000100001　(021_H)	1.833+1.375=3.208
32	000110　(06_H)	0000101100　($02C_H$)	2.000+1.375=3.063

■ 変換時間設定レジスタ ：ADCT (ADCT0, ADCT1)

● ADCT1 ：アドレス $00A8_H$ (アクセス：バイト，ハーフワード，ワード)

● ADCT0 ：アドレス $00A9_H$ (アクセス：バイト，ハーフワード，ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	CT5	CT4	CT3	CT2	CT1	CT0	ST9	ST8
初期値	0	0	0	1	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ST7	ST6	ST5	ST4	ST3	ST2	ST1	ST0
初期値	0	0	1	0	1	1	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit15～bit10] CT5～CT0 : 比較操作時間のクロック分周値設定

- ☐　CT5～CT0 の設定を"000001" (01H)にすると分周なし= PCLK になります。
- ☐　CT5～CT0 の設定を"000000" (00_H)にしないでください。

＜注意事項＞
周辺クロックとして、拡散クロックを使用する場合はベースクロックの拡散時最速周波数に、非拡散クロックを使用する場合は PLL クロック周波数に下記制限があります。(「ベースクロック」「PLL クロック」については、クロック章の図 1-1 および図 3-5 を参照ください。)

ベースクロックもしくは PLL クロックの周波数が 32MHz より速い場合は、周辺クロックの分周設定を、必ず 2 分周以上に設定してください。

ベースクロックもしくは PLL クロックの周波数が 80MHz より速い場合は、周辺クロック(PCLK2)の分周設定にかかわらず、必ず CT5～CT0 の設定を"000010"(02h) 以上に設定してください。

加えて、比較操作時間のクロックは、8～17MHz になるように CT5～CT0 を設定してください。

[bit9～bit0] ST9～ST0 : アナログ入力サンプリング時間設定
リセットにより"0000101100" ($02C_H$)に初期化されます。
ST9-ST0 へ以下の値を設定することは禁止です。必ず 3 以上の値を設定してください。
"0000000010"(02_H), "0000000001"(01_H), "0000000000"(00_H)

44.4.6 A/D 開始/終了チャネル設定レジスタ ：ADSCH, ADECH

A/D 開始／終了チャネル設定レジスタのビット構成について説明します。

A/D 変換の開始チャネルと終了チャネルの設定レジスタです。

A/D 変換動作中に書き換えないでください。

■ A/D 開始チャネル設定レジスタ ：ADSCH

● ADSCH ：アドレス 00AA$_H$ (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	—	—	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
初期値	—	—	0	0	0	0	0	0
属性	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W

■ A/D 終了チャネル設定レジスタ ：ADECH

● ADECH ：アドレス 00AB$_H$ (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	—	—	ANE5	ANE4	ANE3	ANE2	ANE1	ANE0
初期値	—	—	0	0	0	0	0	0
属性	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W

[bit7, bit6] - : **未定義**
常に"0"が読み出されます。書込みは動作に影響ありません。

[bit5～bit0] ANS5～ANS0/ANE5～ANE0 : **開始/終了チャネル**

ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	開始チャネル
ANE5	ANE4	ANE3	ANE2	ANE1	ANE0	終了チャネル
0	0	0	0	0	0	AN0
0	0	0	0	0	1	AN1
0	0	0	0	1	0	AN2
0	0	0	0	1	1	AN3
0	0	0	1	0	0	AN4
0	0	0	1	0	1	AN5
0	0	0	1	1	0	AN6
0	0	0	1	1	1	AN7
0	0	1	0	0	0	AN8
0	0	1	0	0	1	AN9
0	0	1	0	1	0	AN10
0	0	1	0	1	1	AN11
0	0	1	1	0	0	AN12
0	0	1	1	0	1	AN13
0	0	1	1	1	0	AN14
0	0	1	1	1	1	AN15
0	1	0	0	0	0	AN16

ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	開始チャネル
ANE5	ANE4	ANE3	ANE2	ANE1	ANE0	終了チャネル
0	1	0	0	0	1	AN17
0	1	0	0	1	0	AN18
0	1	0	0	1	1	AN19
0	1	0	1	0	0	AN20
0	1	0	1	0	1	AN21
0	1	0	1	1	0	AN22
0	1	0	1	1	1	AN23
0	1	1	0	0	0	AN24
0	1	1	0	0	1	AN25
0	1	1	0	1	0	AN26
0	1	1	0	1	1	AN27
0	1	1	1	0	0	AN28
0	1	1	1	0	1	AN29
0	1	1	1	1	0	AN30
0	1	1	1	1	1	AN31
1	0	0	0	0	0	AN32
1	0	0	0	0	1	AN33
1	0	0	0	1	0	AN34
1	0	0	0	1	1	AN35
1	0	0	1	0	0	AN36
1	0	0	1	0	1	AN37
1	0	0	1	1	0	AN38
1	0	0	1	1	1	AN39
上記以外						設定禁止

A/D 変換の開始チャネルと終了チャネルを設定します。

- ☐ 同一のチャネルが ANS5～ANS0 および ANE5～ANE0 に書き込まれると、変換は 1 つのチャネルに対してのみ行われます (シングル変換)。
- ☐ 連続モード、停止モードを設定している場合は本ビット群で設定されたチャネルの変換が終わると ANS5～ANS0 で設定された開始チャネルに戻ります。
- ☐ 設定チャネルが ANS > ANE の場合は ANS から変換が始まり ch.39 まで変換したら ch.0 に戻り ANE 迄変換します。
- ☐ リセットにより本ビット群は ANS="000000", ANE="000000"に初期化されます。

  例：チャネル設定 ANS=ch.6、ANE=ch.3 で単発モードのとき変換は次の順序で行われます。

  ch.6 → ch.7 → ch.8 → ･･･ → ch.39 → ch.0 → ch.1 → ch.2 → ch.3

44.4.7　拡張アナログ入力許可レジスタ ：EADERLL

拡張アナログ入力許可レジスタについて説明します。

使用する端子をアナログ入力に設定します。(ch.39～ch.32)

■ EADERLL： アドレス 00AD$_H$ (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	ADE39	ADE38	ADE37	ADE36	ADE35	ADE34	ADE33	ADE32
初期値	0	0	0	0	0	0	0	0
属性	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

[bit7～bit0] ADE[39:32] : アナログ入力許可

ADEn	意味
0	ポート入力/出力モード
1	アナログ入力モード

n=32～39

開始チャネル、終了チャネルのアナログ入力許可レジスタには必ず"1" を設定してください。

44.4.8　拡張 A/D 制御ステータスレジスタ ：EADCS

拡張 A/D 制御ステータスレジスについて説明します。

現在の変換チャネルまたは変換が停止されたチャネルを表示します。

■ EADCS： アドレス 00AE$_H$ (アクセス：バイト，ハーフワード，ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約		ACH5	ACH4	ACH3	ACH2	ACH1	ACH0
初期値	-	-	0	0	0	0	0	0
属性	R0,WX	R0,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX

[bit5～bit0] ACH5～ACH0： アナログ変換チャネル

ACH5	ACH4	ACH3	ACH2	ACH1	ACH0	変換チャネル
0	0	0	0	0	0	AN0
0	0	0	0	0	1	AN1
0	0	0	0	1	0	AN2
0	0	0	0	1	1	AN3
0	0	0	1	0	0	AN4
0	0	0	1	0	1	AN5
0	0	0	1	1	0	AN6
0	0	0	1	1	1	AN7
0	0	1	0	0	0	AN8
0	0	1	0	0	1	AN9
0	0	1	0	1	0	AN10
0	0	1	0	1	1	AN11
0	0	1	1	0	0	AN12
0	0	1	1	0	1	AN13
0	0	1	1	1	0	AN14
0	0	1	1	1	1	AN15
0	1	0	0	0	0	AN16
0	1	0	0	0	1	AN17
0	1	0	0	1	0	AN18
0	1	0	0	1	1	AN19
0	1	0	1	0	0	AN20
0	1	0	1	0	1	AN21
0	1	0	1	1	0	AN22
0	1	0	1	1	1	AN23
0	1	1	0	0	0	AN24
0	1	1	0	0	1	AN25
0	1	1	0	1	0	AN26
0	1	1	0	1	1	AN27
0	1	1	1	0	0	AN28
0	1	1	1	0	1	AN29
0	1	1	1	1	0	AN30
0	1	1	1	1	1	AN31
1	0	0	0	0	0	AN32
1	0	0	0	0	1	AN33
1	0	0	0	1	0	AN34
1	0	0	0	1	1	AN35
1	0	0	1	0	0	AN36
1	0	0	1	0	1	AN37
1	0	0	1	1	0	AN38
1	0	0	1	1	1	AN39

ADCS1:BUSY	読込み時のチャネル状態
1 (A/D 変換中)	現在の変換チャネル
0 (強制終了時)	変換が停止されたチャネル

44.5 動作説明

A/D コンバータの動作について説明します。

A/D の動作モードについて、以下に説明します。

44.5.1 シングル変換動作

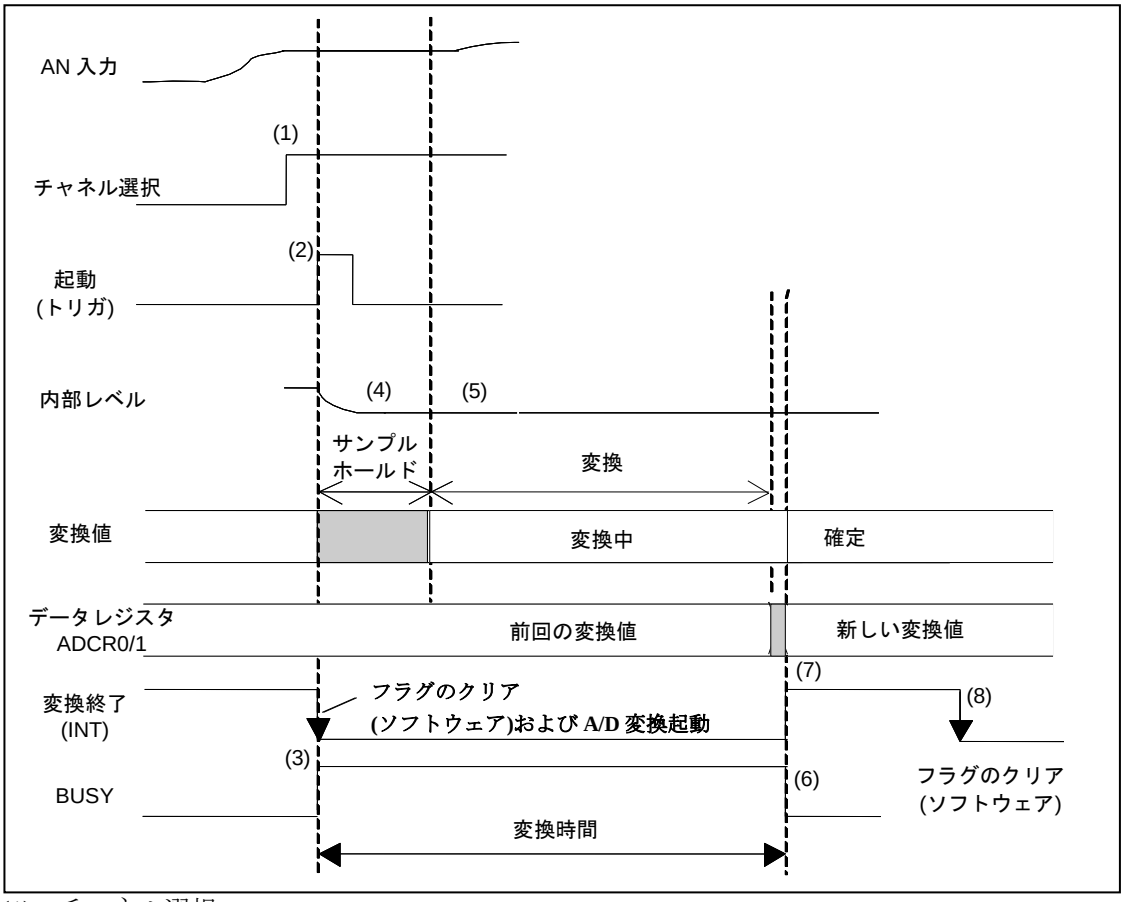
44.5.2 スキャン変換動作

44.5.3 変換モード

44.5.1 シングル変換動作

シングル変換の動作について説明します。

シングル変換動作

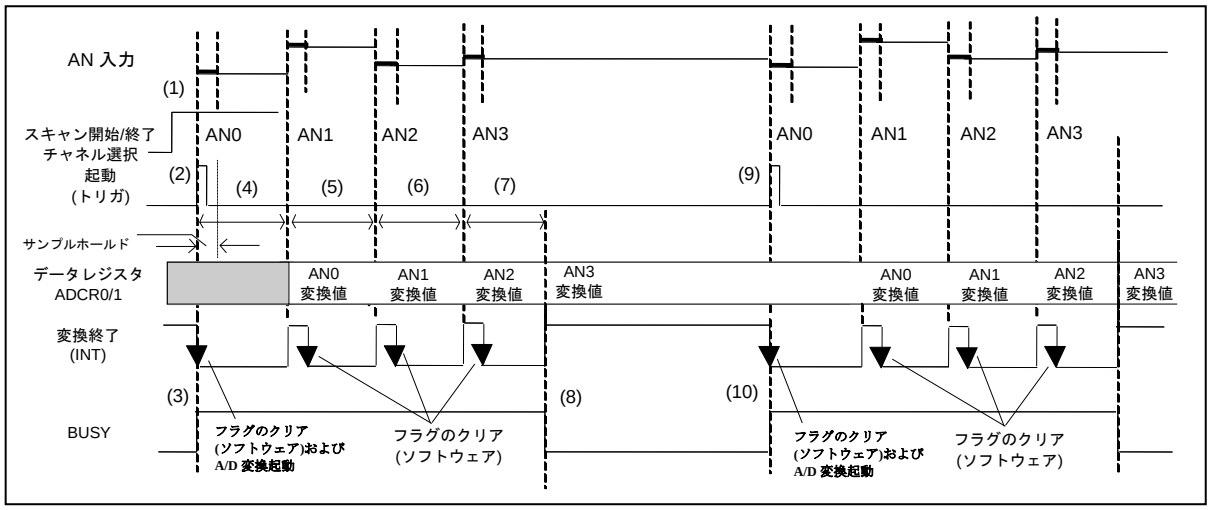


(1) チャネル選択
(2) A/D 変換起動 (トリガ入力 : ソフトトリガ/リロードタイマ/外部トリガ)
(3) INT フラグクリア, BUSY フラグセット
(4) サンプルホールド
(5) 変換
(6) 変換終了, INT フラグセット, BUSY フラグクリア
(7) 変換値のデータレジスタへの格納
(8) ソフトウェアによる INT フラグのクリア

44.5.2 スキャン変換動作

スキャン変換の動作について説明します。

スキャン変換動作



(1) 起動チャネル選択
(2) A/D 起動 (トリガ : ソフトトリガ/リロードタイマ/外部トリガ)
(3) INT フラグクリア, BUSY フラグセット
(4) 1. AN0 変換
2. INT フラグセット(AN0 の変換データ格納)
(5) 1. AN1 変換
2. AN0 の変換結果を転送
3. INT フラグクリア
4. INT フラグセット(AN1 の変換データ格納)
(6) 1. AN2 変換
2. AN1 の変換結果を転送
3. INT フラグクリア
4. INT フラグセット(AN2 の変換データ格納)
(7) 1. AN3 変換
2. AN2 の変換結果を転送
3. INT フラグクリア
4. INT フラグセット(AN3 の変換データ格納)
(8) INT フラグのセット, BUSY フラグのクリア
(9) 次の A/D 起動
(10) INT フラグクリア, BUSY フラグのセット

44.5.3 変換モード

変換モードについて説明します。

A/D コンバータは逐次比較方式で作動し、分解能は 10 ビットまたは 8 ビットを選択することができます。

この A/D コンバータは変換結果格納用のレジスタが 1 つ (16 ビット)であるため, 変換データレジスタ (ADCR0 および ADCR1)は変換が完了するたびに書き換えられます。このため、A/D コンバータ単独では連続変換処理には適しませんので DMA を使って変換データをメモリに転送しながら変換することを推奨いたします。動作モードを次に説明します。

■ 単発変換モード

単発変換モードは，ANS ビットと ANE ビットで設定されたアナログ入力を順に変換していき ANE ビットで設定された終了チャネルまで変換が終わると A/D は動作を停止します。開始チャネルと終了チャネルが同じとき(ANS＝ANE)は一方のチャネルのみの変換動作になります。

【例】

- ANS＝000000，ANE＝000011
  開始 → AN0 → AN1 → AN2 → AN3 → 終了
- ANS＝000010，ANE＝000010
  開始 → AN2 → 終了

■ 連続変換モード

連続変換モードは，ANS ビットと ANE ビットで設定されたアナログ入力を順に変換していき ANE ビットで設定された終了チャネルまで変換が終わると ANS のアナログ入力に戻り変換動作を続けます。開始チャネルと終了チャネルが同じとき(ANS＝ANE)は一方のチャネルのみの変換を続けます。

【例】

- ANS＝000000，ANE＝000011
  開始 → AN0 → AN1 → AN2 → AN3 → AN0 ーー→ 繰り返し
- ANS＝000010，ANE＝000010
  開始 → AN2 → AN2 → AN2 ーー→ 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→強制終了)。強制終了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 (強制終了した場合には変換レジスタに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、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 チャネル変換するたびに変換動作を一時停止します。一時停止を解除するにはもう一度起動をかけることにより行われます。

ANE ビットで設定された終了チャネルまで変換が終わると ANS のアナログ入力に戻り変換動作を続けます。開始チャネルと終了チャネルが同じ(ANS=ANE)であれば変換は一方のチャネルに対してのみ実行されます。

【例】

- ANS＝000000，ANE＝000011
  開始 → AN0 → 停止 → 起動 → AN1 → 停止 → 起動 → AN2 → 停止 → 起動 → → AN3 → 停止 → 起動 → AN0 ーー→ 繰り返し
- ANS＝000010，ANE＝000010
  開始 → AN2 → 停止 → 起動 → AN2 → 停止 → 起動 → AN2 ーー→ 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。

44.6 設定

A/D コンバータの設定について説明します。

表 44-2 A/D を使うために必要な設定-単発変換モード

設定	設定レジスタ	設定方法
モード選択 (単発変換)	A/D 制御 (ADCS0)	44.7.1 参照
ビット長選択		44.7.2 参照
チャネルの選択		44.7.3 参照
変換時間の設定	変換時間設定 (ADCT)	44.7.4 参照
AN 端子を入力に設定	アナログ入力許可 (ADER) 拡張アナログ入力許可(EADERLL)	44.7.5 参照
A/D 起動トリガ選択	A/D 制御 (ADCS1)	44.7.6 参照
A/D 起動トリガ発生 ソフトトリガ → ソフトトリガビットの設定		44.7.6 参照
リロードタイマ → リロードタイマ立上り出力	『リロードタイマ』の章を参照してください。	
外部トリガ →ADTG 端子にトリガを入力	外部より入力	
変換終了フラグ確認	A/D 制御 (ADCS1)	44.7.8 参照
変換値読出し	データレジスタ (ADCR0, ADCR1)	44.7.9 参照

表 44-3 A/D を使うために必要な設定-連続変換モード

設定	設定レジスタ	設定方法
モード選択 (連続変換)	A/D 制御 (ADCS0, ADCS1)	44.7.1 参照
ビット長選択		44.7.2 参照
開始チャネルの選択		44.7.3 参照
変換時間の設定	変換時間設定 (ADCT)	44.7.4 参照
AN 端子を入力に設定	アナログ入力許可 (ADER) 拡張アナログ入力許可(EADERLL)	44.7.5 参照
A/D 起動トリガ選択	A/D 制御 (ADCS1)	44.7.6 参照
A/D 起動トリガ発生 ソフトトリガ → ソフトトリガビットの設定		44.7.7 参照
リロードタイマ → リロードタイマ立上り出力	『リロードタイマ』の章を参照してください。	
外部トリガ →ADTG 端子にトリガを入力	外部より入力	
変換終了フラグ確認	A/D 制御 (ADCS1)	44.7.8 参照
変換値読出し	データレジスタ (ADCR0, ADCR1)	44.7.9 参照

表 44-4 A/D 動作の強制停止

設定	設定レジスタ	設定方法
強制停止	A/D 制御 (ADCS1)	44.7.10 参照

表 44-5 A/D 割込みを行うために必要な項目

設定	設定レジスタ	設定方法
A/D 割込みベクタ, A/D 割込みレベルの設定	『割込み制御(割込みコントローラ)』の章を参照してください。	44.7.11 参照
A/D 割込み要因選択 (A/D 変換終了)	A/D 制御レジスタ (ADCS1)	44.7.12 参照
A/D 割込み設定 割込み要求のクリア 割込み要求の許可		44.7.13 参照

44.7 Q&A

A/D コンバータの Q&A について説明します。

44.7.1. 変換モードの種類と設定方法は?

44.7.2. ビット長を指定するには?

44.7.3. チャネルを選択するには?

44.7.4. 変換時間を設定するには?

44.7.5. アナログ端子入力を有効にするには?

44.7.6. A/D コンバータの起動方法を選択するには?

44.7.7. A/D コンバータを起動するには?

44.7.8. 変換終了を確認するには?

44.7.9. 変換値を読み出すには?

44.7.10. A/D 変換動作を強制的に停止させるには?

44.7.11. 割込み関連レジスタは??

44.7.12. 割込みの種類は?

44.7.13. 割込みを許可/禁止/クリアするには?

44.7.1 変換モードの種類と設定方法は?

変換モードの種類と設定方法について説明します。

変換には、下記の 4 つがあります。

- 指定したチャネル分を 1 サイクル変換して終了する単発変換モード(動作中の再起動可能)
- 指定したチャネル分を 1 サイクル変換して終了する単発変換モード(動作中の再起動不可能)
- 指定したチャネル分を繰り返し変換する連続変換モード
- 指定したチャネル分までの変換を 1 チャネルごとに行い一時停止する停止変換モード

動作モード設定ビット (ADCS0:MD[1:0]) にて設定してください。

動作モード	動作モード設定ビット (MD[1:0])
単発変換モード(動作中の再起動可能)にするには	"00"にする
単発変換モード(動作中の再起動不可能)にするには	"01"にする
連続変換モードにするには	"10"にする
停止変換モードにするには	"11"にする

44.7.2 ビット長を指定するには?

ビット長の指定について説明します。

変換結果の格納ビット数の設定 (ADCS0:S10) にて行ってください。

動作	変換結果の格納ビット数 (S10)
データレジスタに 10 ビットで格納するには	"0"にする
データレジスタに 8 ビットで格納するには	"1"にする

44.7.3 チャネルを選択するには?

チャネルの選択について説明します。

変換するチャネルを A/D 開始チャネル設定ビット (ADSCH:ANS[5:0])と A/D 終了チャネル設定ビット (ADECH:ANE[5:0]) にて指定してください。

A/D 変換の開始チャネルを指定してください。

A/D 変換開始チャネル	チャネル選択ビット(ANS[5:0])
AN0 を指定するには	"000000"にする
AN1 を指定するには	"000001"にする
AN2 を指定するには	"000010"にする
AN3 を指定するには	"000011"にする
AN4 を指定するには	"000100"にする
AN5 を指定するには	"000101"にする
AN6 を指定するには	"000110"にする
AN7 を指定するには	"000111"にする
AN8 を指定するには	"001000"にする
AN9 を指定するには	"001001"にする
AN10 を指定するには	"001010"にする
AN11 を指定するには	"001011"にする
AN12 を指定するには	"001100"にする
AN13 を指定するには	"001101"にする
AN14 を指定するには	"001110"にする
AN15 を指定するには	"001111"にする
AN16 を指定するには	"010000"にする
AN17 を指定するには	"010001"にする
AN18 を指定するには	"010010"にする
AN19 を指定するには	"010011"にする
AN20 を指定するには	"010100"にする
AN21 を指定するには	"010101"にする
AN22 を指定するには	"010110"にする
AN23 を指定するには	"010111"にする
AN24 を指定するには	"011000"にする
AN25 を指定するには	"011001"にする
AN26 を指定するには	"011010"にする
AN27 を指定するには	"011011"にする
AN28 を指定するには	"011100"にする
AN29 を指定するには	"011101"にする
AN30 を指定するには	"011110"にする
AN31 を指定するには	"011111"にする
AN32 を指定するには	"100000"にする
AN33 を指定するには	"100001"にする
AN34 を指定するには	"100010"にする
AN35 を指定するには	"100011"にする
AN36 を指定するには	"100100"にする
AN37 を指定するには	"100101"にする
AN38 を指定するには	"100110"にする
AN39 を指定するには	"100111"にする

A/D 変換の終了チャネルを指定してください。

A/D 変換終了チャネル	チャネル選択ビット(ANE[5:0])
AN0 を指定するには	"000000"にする
AN1 を指定するには	"000001"にする
AN2 を指定するには	"000010"にする
AN3 を指定するには	"000011"にする
AN4 を指定するには	"000100"にする
AN5 を指定するには	"000101"にする
AN6 を指定するには	"000110"にする
AN7 を指定するには	"000111"にする
AN8 を指定するには	"001000"にする
AN9 を指定するには	"001001"にする
AN10 を指定するには	"001010"にする
AN11 を指定するには	"001011"にする
AN12 を指定するには	"001100"にする
AN13 を指定するには	"001101"にする
AN14 を指定するには	"001110"にする
AN15 を指定するには	"001111"にする
AN16 を指定するには	"010000"にする
AN17 を指定するには	"010001"にする
AN18 を指定するには	"010010"にする
AN19 を指定するには	"010011"にする
AN20 を指定するには	"010100"にする
AN21 を指定するには	"010101"にする
AN22 を指定するには	"010110"にする
AN23 を指定するには	"010111"にする
AN24 を指定するには	"011000"にする
AN25 を指定するには	"011001"にする
AN26 を指定するには	"011010"にする
AN27 を指定するには	"011011"にする
AN28 を指定するには	"011100"にする
AN29 を指定するには	"011101"にする
AN30 を指定するには	"011110"にする
AN31 を指定するには	"011111"にする
AN32 を指定するには	"100000"にする
AN33 を指定するには	"100001"にする
AN34 を指定するには	"100010"にする
AN35 を指定するには	"100011"にする
AN36 を指定するには	"100100"にする
AN37 を指定するには	"100101"にする
AN38 を指定するには	"100110"にする
AN39 を指定するには	"100111"にする

44.7.4　変換時間を設定するには?

変換時間の設定について説明します。

変換時間設定レジスタ ADCT で設定してください。
(式 1) サンプリング時間＝ ST 設定値 × $1/F_{PCLK}$
(式 2) 比較操作時間＝ CT 設定値 × $1/F_{PCLK}$ × 10 + $4/F_{PCLK}$
(式 3) 変換時間 (トータル) ＝ サンプリング時間＋比較操作時間

設定項目	制御ビット	推奨値(F_{PCLK})			備考
		16MHz 時	24MHz 時	32MHz 時	
サンプリング時間を設定するには	(ST[9:0])	"0000010110" (1.375µs)	"0000100001" (1.375µs)	"0000101100" (1.375µs)	1.2µs以上に設定してください (AVcc<4.5V 時)
比較操作時間を設定するには	(CT[5:0])	000011 (2.125µs)	000100 (1.833µs)	000110 (2.000µs)	500µs 以下に設定してください

＜注意事項＞
ST[9:0]の設定値は A/D のサンプリング時間が必要サンプリング以上になるように設定してください。

必要サンプリング時間については「44.4.5. 変換時間設定レジスタ : ADCT」を参照してください。

44.7.5　アナログ端子入力を有効にするには?

アナログ端子入力を有効にするには?について説明します。

アナログ入力許可レジスタ(ADER)または拡張アナログ入力許可レジスタ(EADERLL)で行ってください。

動作	制御ビット	設定
AN0 端子を入力にするには	ADER:ADE0	"1"にする
AN1 端子を入力にするには	ADER:ADE1	"1"にする
AN2 端子を入力にするには	ADER:ADE2	"1"にする
AN3 端子を入力にするには	ADER:ADE3	"1"にする
AN4 端子を入力にするには	ADER:ADE4	"1"にする
AN5 端子を入力にするには	ADER:ADE5	"1"にする
AN6 端子を入力にするには	ADER:ADE6	"1"にする
AN7 端子を入力にするには	ADER:ADE7	"1"にする
AN8 端子を入力にするには	ADER:ADE8	"1"にする
AN9 端子を入力にするには	ADER:ADE9	"1"にする
AN10 端子を入力にするには	ADER:ADE10	"1"にする
AN11 端子を入力にするには	ADER:ADE11	"1"にする
AN12 端子を入力にするには	ADER:ADE12	"1"にする
AN13 端子を入力にするには	ADER:ADE13	"1"にする
AN14 端子を入力にするには	ADER:ADE14	"1"にする
AN15 端子を入力にするには	ADER:ADE15	"1"にする
AN16 端子を入力にするには	ADER:ADE16	"1"にする
AN17 端子を入力にするには	ADER:ADE17	"1"にする
AN18 端子を入力にするには	ADER:ADE18	"1"にする
AN19 端子を入力にするには	ADER:ADE19	"1"にする
AN20 端子を入力にするには	ADER:ADE20	"1"にする
AN21 端子を入力にするには	ADER:ADE21	"1"にする
AN22 端子を入力にするには	ADER:ADE22	"1"にする
AN23 端子を入力にするには	ADER:ADE23	"1"にする
AN24 端子を入力にするには	ADER:ADE24	"1"にする
AN25 端子を入力にするには	ADER:ADE25	"1"にする
AN26 端子を入力にするには	ADER:ADE26	"1"にする
AN27 端子を入力にするには	ADER:ADE27	"1"にする
AN28 端子を入力にするには	ADER:ADE28	"1"にする
AN29 端子を入力にするには	ADER:ADE29	"1"にする
AN30 端子を入力にするには	ADER:ADE30	"1"にする
AN31 端子を入力にするには	ADER:ADE31	"1"にする
AN32 端子を入力にするには	EADERLL:ADE32	"1"にする
AN33 端子を入力にするには	EADERLL:ADE33	"1"にする
AN34 端子を入力にするには	EADERLL:ADE34	"1"にする
AN35 端子を入力にするには	EADERLL:ADE35	"1"にする
AN36 端子を入力にするには	EADERLL:ADE36	"1"にする
AN37 端子を入力にするには	EADERLL:ADE37	"1"にする
AN38 端子を入力にするには	EADERLL:ADE38	"1"にする
AN39 端子を入力にするには	EADERLL:ADE39	"1"にする

44.7.6 A/D コンバータの起動方法を選択するには?

A/D コンバータの起動方法の選択について説明します。

起動トリガには、下記の 3 種類があります。

ソフトトリガ
リロードタイマの立上り信号
外部トリガ入力の立下り信号

起動トリガの設定は、起動要因選択ビット (ADCS1:STS[1:0])で行ってください。

A/D 起動要因	起動要因選択ビット(STS[1:0])
ソフトトリガを指定するには	"00"にする
外部トリガ/ソフトトリガを指定するには	"01"にする
リロードタイマ/ソフトトリガを指定するには	"10"にする
外部トリガ/リロードタイマ/ソフトトリガを指定するには	"11"にする

A/D コンバータは、選択した要因の中で最初にきた要因で起動します。

44.7.7 A/D コンバータを起動するには?

A/D コンバータの起動について説明します。

ソフトトリガの発生方法
ソフトトリガは A/D 変換ソフトトリガビット (ADCS1:STRT) で行ってください。

動作	A/D 変換ソフトトリガビット (STRT)
ソフトトリガを発生させるには	"1"を書き込む

リロードタイマ 1 での起動方法
リロードタイマの設定と起動が必要です。詳細は『リロードタイマ』の章を参照してください。

リロードタイマのアンダフローによって、リロードタイマの出力信号が立上りになると、起動トリガが発生します。

外部トリガによる起動方法
外部トリガは外部トリガ入力端子 ADTG で行ってください。

ADTG 端子をペリフェラル入力に設定してください。設定方法は『I/O ポート』の章を参照してください。

動作	設定
ADTG 端子をトリガ入力にするには	端子をペリフェラル入力に設定。 設定方法は『I/O ポート』の章を参照してください。

44.7.8 変換終了を確認するには?

変換終了の確認について説明します。

変換終了を確認する方法は、下記の 2 つがあります。

A/D 変換終了割込み要求フラグ(ADCS1:INT)で確認する方法

(INT)	意味
読み出した値が"0"の場合	A/D 変換終了割込み要求なし
読み出した値が"1"の場合	A/D 変換終了割込み要求あり

動作確認ビット (ADCS1:BUSY)で確認する方法

(BUSY)	設定
読み出した値が"0"の場合	A/D 変換終了(停止中)
読み出した値が"1"の場合	A/D 変換中

44.7.9 変換値を読み出すには?

変換値の読出しについて説明します。

変換値は、データレジスタ ADCR0、ADCR1 から読み出せます。

ADCR0 は下位 8 ビットを格納し、ADCR1 は変換結果の最上位 2 ビットを格納します。

データレジスタの値は 1 回の変換終了ごとに更新されます。通常、最終変換値を格納します。

動作	レジスタ
10 ビット変換値を読み出すには	ADCR1, ADCR0 レジスタの読出し
8 ビット変換値を読み出すには	ADCR0 レジスタの読出し

44.7.10 A/D 変換動作を強制的に停止させるには?

A/D 変換動作の強制停止について説明します。

強制停止ビット(ADCS1:BUSY)で行ってください。

動作	強制停止ビット(BUSY)
A/D 変換動作を強制的に停止させるには	"0"を書き込む

強制停止ビット(BUSY)に、"1"を書き込んでも、A/D の動作に影響はありません。

44.7.11 割込み関連レジスタは??

割込み関連レジスタについて説明します。

A/D 割込みベクタ、A/D 割込みレベルの設定

A/D 番号、割込みレベル、ベクタの関係は下表のとおりです。

割込みレベル、割込みベクタの詳細については『割込み制御(割込みコントローラ)』の章を参照してください。

割込みベクタ (デフォルト)	割込みレベル設定ビット(ICR[4:0])
#48 Address: 0FFF3C$_H$	割込みレベルレジスタ(ICR32) Address: 00460$_H$

44.7.12 割込みの種類は?

割込みの種類について説明します。

割込み要因は A/D 変換終了のみです。選択のビットはありません。

44.7.13 割込みを許可/禁止/クリアするには?

割込みの許可/禁止/クリアについて説明します。

割込み要求の許可ビット、割込み要求フラグ

割込み許可の設定は、割込み要求許可ビット(ADCS1:INTE)にて行ってください。

動作	割込み要求許可ビット(INTE)
割込み要求を禁止するには	"0"にする
割込み要求を許可するには	"1"にする

割込み要求のクリアは、割込み要求フラグ(ADCS1:INT) にて行ってください。

動作	割込要求フラグ(INT)
割込み要求をクリアするには	"0"を書き込む (「17.7.7. 起動トリガの種類と選択方法は?」参照)

44.8 サンプルプログラム

サンプルプログラムについて説明します。

設定手順例 1

AN0 より入力したレベルをA/D 変換する例 (シングル変換、ソフトトリガ) を以下に示します。

<初期設定>

・ポート

	レジスタ名. ビット名
ポートの A/D 入力選択	ADERL. AN7-0

・A/D 開始/ 終了チャネル設定

	レジスタ名. ビット名
変換開始チャネルの設定	ADSCH .ANS5-0
変換終了チャネルの設定	ADECH .ANE5-0

・A/D 開始/ 終了チャネル設定

	レジスタ名. ビット名
変換時間の設定	ADCT .CT5-0 .ST9-8 .ST7-0

・A/D 制御

	レジスタ名. ビット名
AN0 の制御 割込み要求フラグのクリア ≫ 割込み禁止≫ 起動トリガ選択≫ 変換モード選択≫ ビット長の選択≫	ADCS1 .BUSY .INT .INTE .PAUS .STS .STRT . 予約ビット .MD[1:0] .S10 . 予約ビット

・割込み関連

	レジスタ名. ビット名
A/D 割込みレベルの設定	ICR32
I フラグの設定	(CCR)

<A/D 起動>

・A/D 制御

	レジスタ名. ビット名
A/D 割込み許可	ADCS1 .INT .INTE
A/D0 のソフト起動	ADCS1 .BUSY .STRT

<割込み>

・変換値の読出し

	レジスタ名. ビット名
割込み禁止, 割込み要求フラグクリア	ADCS1 .INT .INTE
変換値の読出し	D9 - D0
割込み許可	ADCS1 .INTE

<割込みベクタ>
ベクタテーブルの設定

<その他>
(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 1

```
void AD_sample_1(void)
{
        AD_INITIAL();
        AD_ch0_start();
}

void AD_INITIAL(void)
{
        IO_ADERL = 0x01;        /* AN0 のみ A/D 入力 */

        IO_ADSCH = 0x0000;      /* AN0 設定 */
                                /* 000000 */
        IO_ADECH = 0x0000;      /* AN0 設定 */
                                /* 000000 */


        IO_ADCT0 = 0x0816;      /* 値は推奨値(16MHz 時) */
                                /* 000010 */
                                /* 00 */
                                /* 00010110 */


        IO_ADCS1.hword= 0x8000; /* 設定値 : 10000000 00000000 (bit)*/
                                 /* Bit15=1:      ( 影響なし)*/
                                 /* Bit14=0:      割込み要求クリア */
                                 /* Bit13=0:      割込み禁止 */
                                 /* Bit12=0:      フラグクリア*/
                                 /* Bit11-10=00:  ソフトトリガ*/
                                 /* Bit9=0:       ( 影響なし) */
                                 /* Bit8=0:       */
                                 /* Bit7-6=00:    単発変換 */
                                 /* Bit5=0:       10 bits */
                                 /* Bit4-0=00000: */


      IO_ICR[32].bit.ICR =32;   /* 値は任意 */
      __EI();                   /* 割込み許可 */
}

AD_ch0_start()
{
      IO_ADCS1.hword= 0x2000;
                                /* Bit6=0: AD 割込みフラグクリア */
                                /* Bit5=1: AD 割込み許可 */
      IO_ADCS1.hword= 0xF200;
                                /* Bit7=1: "1" 書込みが必要 */
                                /* Bit1=1: ソフト起動 */
}
__interrupt void AD_ch0_int()   /* */
{
      IO_ADCS1.hword = 0x8000;
                                /* Bit6=0: AD 割込みフラグクリア */
                                /* Bit5=0: AD 割込み禁止 */
      [ 任意の格納場所]= ADCR1,ADCR0; /* 変換値の格納 */
      IO_ADCS1.hword= 0xA000;
                                /* Bit5=1: AD 割込み許可 */
}

ベクタテーブルにて割込みルーチンの指定が必要
#pragma intvect AD_ch0_int 48
```

設定手順例 2

AN1 からAN3 より入力したレベルをA/D 変換する（ スキャン変換、 外部トリガ）。
(ADTG に外部トリガ(立下り) 入力が必要です)

〈初期設定〉

・ポート

	レジスタ名. ビット名
ポートの A/D 入力選択	ADERL. AN7-0
外部トリガポート設定	『I/O ポート』の章を参照

・A/D 開始/ 終了チャネル設定

	レジスタ名. ビット名
変換開始チャネルの設定	ADSCH .ANS5-0
変換終了チャネルの設定	ADECH .ANE5-0

・A/D 開始/ 終了チャネル設定

	レジスタ名. ビット名
変換時間の設定	ADCT .CT5-0 .ST9-8 .ST7-0

・A/D 制御

	レジスタ名. ビット名
AN0 の制御	ADCS1 .BUSY
割込み要求フラグのクリア ≫	.INT
割込み禁止≫	.INTE
	.PAUS
起動トリガ選択≫	.STS .STRT
変換モード選択≫	. 予約ビット
ビット長の選択≫	.MD[1:0]
	.S10
	. 予約ビット

・割込み関連

	レジスタ名. ビット名
A/D 割込みレベルの設定	ICR32
I フラグの設定	(CCR)

〈A/D 起動〉

・A/D 制御

	レジスタ名. ビット名
A/D 割込み許可	ADCS1 .INT .INTE

〈割込み〉

・変換値の読出し

	レジスタ名. ビット名
割込み禁止, 割込み要求フラグクリア	ADCS1 .INT .INTE
変換値の読出し	D9 - D0
割込み許可	ADCS1 .INTE

〈割込みベクタ〉
ベクタテーブルの設定

〈その他〉
(注意事項)
事前にクロック関連の設定および__set_il(数値) の設定が必要です。『クロック』の章および『割込み制御(割込みコントローラ)』の章を参照してください。

プログラム例 2 (条件：PCLK=16MHz)

```
void AD_sample_2(void)
{
        AD_1to3_INITIAL();
        AD_ch1to3_start();
}

void AD_1to3_INITIAL(void)
{
        IO_ADERL = 0x0E;        /* AN1 から AN3 のみ A/D 入力 */
                                /* ADTG 端子をペリフェラル入力に設定してください */
        PORT_SETTING_ADTG_IN();

        IO_ADSCH = 0x0001;      /* AN1 設定 */
                                /* 000001 */
        IO_ADECH = 0x0003;      /* AN3 設定 */
                                /* 000011 */

        IO_ADCT0 = 0x0816;      /* 値は推奨値(16MHz 時) */
                                /* 000010 */
                                /* 00 */
                                /* 00010110 */


        IO_ADCS1.hword= 0x8800; /* 設定値：10001000 00000000 (bit)*/
                                /* Bit15=1:     ( 影響なし)*/
                                /* Bit14=0:     割込み要求クリア */
                                /* Bit13=0:     割込み禁止 */
                                /* Bit12=0:     フラグクリア*/
                                /* Bit11-10=01: 外部トリガ*/
                                /* Bit9=0:      ( 影響なし) */
                                /* Bit8=0:      */
                                /* Bit7-6=00:   単発変換 */
                                /* Bit5=0:      10 bits */
                                /* Bit4-0=00000: */


      IO_ICR[32].bit.ICR =32;   /* 値は任意 */
      __EI();                   /* 割込み許可 */
}

AD_ch01to3_start()
{
      IO_ADCS1.hword= 0xB400;
                                /* Bit6=0: AD 割込みフラグクリア */
                                /* Bit5=1: AD 割込み許可 */
}

__interrupt void AD_ch01to3_int()    /* AN3 変換後に割込みがかかる。 */
{
      IO_ADCS1.hword = 0x9400;
                                /* Bit6=0: AD 割込みフラグクリア */
                                /* Bit5=0: AD 割込み禁止 */
      [ 任意の格納場所]= ADCR1,ADCR0; /* 変換値の格納 */
      IO_ADCS1.hword= 0xA400;
                                /* Bit5=1: AD 割込み許可 */
}

ベクタテーブルにて割込みルーチンの指定が必要
#pragma intvect AD_ch01to3_int 48
```

設定手順例 3

AN1 からAN3 より入力したレベルをA/D 変換する（スキャン変換、外部トリガ、DMA 使用（割込みによる要求・DMA チャネル 0））。
（ADTG に外部トリガ（ 立下り）入力が必要です）

<初期設定>

・ポート	レジスタ名．ビット名
ポートの A/D 入力選択	ADERL．AN7-0
外部トリガポート設定	DDR7.P70

・A/D 開始/ 終了チャネル設定	レジスタ名．ビット名
変換開始チャネルの設定	ADSCH .ANS5-0
変換終了チャネルの設定	ADECH .ANE5-0

・A/D 開始/ 終了チャネル設定	レジスタ名．ビット名
変換時間の設定	ADCT .CT5-0 .ST9-8 .ST7-0

・A/D 制御	レジスタ名．ビット名
ANO の制御	ADCS1 .BUSY
割込み要求フラグのクリア ≫	.INT
割込み禁止≫	.INTE
	.PAUS
起動トリガ選択≫	.STS .STRT
変換モード選択≫	. 予約ビット
ビット長の選択≫	.MD[1:0]
	.S10
	. 予約ビット

・割込み関連	レジスタ名．ビット名
A/D 割込みレベルの設定	ICR32

・DMA 関連	レジスタ名．ビット名
DMA 転送要求の発生・クリア（ICSEL の設定は不要）	IORR
DMA チャンネル 0 設定	DCCR
DMA チャンネル 0 転送元	DSAR
DMA チャンネル 0 転送先	DDAR
DMA チャンネル 0 転送回数	DTCR
DMA チャンネル 0 ステータスクリア	DCSR
DMA 許可	DMACR.DME

<A/D 起動>

・A/D 制御	レジスタ名．ビット名
A/D 割込み許可	ADCS1 .INT .INTE

プログラム例 3　　（条件：PCLK=16MHz）

```
void AD_sample_3(void)
{
        AD _1to3_INITIAL();
        DMA_Setting();
        AD_ch1to3_start();
}

void AD_1to3_INITIAL(void)
{
        IO_ADERL = 0x0E;                /* AN1 から AN3 のみ A/D 入力 */
        IO_PORT1.IO_DDR7.bit. P70 = 0; /* DDR7.P70 を入力に設定*/

        IO_ADSCH = 0x0001;      /* AN1 設定 */
                                /* 000001 */
        IO_ADECH = 0x0003;      /* AN3 設定 */
                                /* 000011 */

        IO_ADCT0 = 0x0816;      /* 値は推奨値(16MHz 時) */
                                /* 000010 */
                                /* 00 */
                                /* 00010110 */


        IO_ADCS1.hword= 0x8400; /* 設定値：10001000 00000000 (bit)*/
                                 /* Bit15=1:     （影響なし)*/
                                 /* Bit14=0:     割込み要求クリア */
                                 /* Bit13=0:     割込み禁止 */
                                 /* Bit12=0:     フラグクリア*/
                                 /* Bit11-10=01: 外部トリガ*/
                                 /* Bit9=0:       （影響なし）*/
                                 /* Bit8=0:      */
                                 /* Bit7-6=00:   単発変換 */
                                 /* Bit5=0:      10 bits */
                                 /* Bit4-0=00000: */


      IO_ICR[32].bit.ICR =31;    /* 値は任意 */
}

void DMA_Setting(){
      IO_DMA0_DCCR.word= 0x00000000    // チャネル 0 ディセーブル
      IO_DMA0_DCSR.hword= 0x0000       // チャネル 0 ステータスクリア
      IO_DMA0_DSAR.word=               // 転送元＝ ADC データレジスタ
        &IO_ADCR1.hword;               //
      IO_DMA0_DDAR.word= ( 任意 );     // SRAM 内のバッファ（ ソフトウェアで FIFO)
      IO_DMA0_DTCR.hword= 0x0100;      // 256 個まで。
      IO_DMAREQCLR_IORR0.byte=         // IO 転送要求レジスタ
        (( 0x30 - 0x10 ) + 0x40);      // ベクタ番号#0x30、IOE ビットセット
      IO_DMA0_DCCR.word= 0x8010B010;   // チャネル 0 許可、割込みによる要求、
                                        // ブロック転送、転送元アドレス固定、ST=1
                                        // 転送先アドレスインクリメント
                                        // 1 ブロック＝ 2 バイト×１回
      IO_DMACR.word= 0x80000000        // DMA 許可
}

AD_ch01to3_start()
{
      IO_ADCS1.hword= 0xB400;
                                 /* Bit6=0: AD 割込みフラグクリア */
                                 /* Bit5=1: AD 割込み許可 */
}
```

44.9 注意事項

A/D コンバータの注意事項について説明します。

A/D コンバータを使用するにあたっての注意点を示します。

■ 電源投入順序
A/D コンバータの電源 (AVcc, AVRH)投入および、アナログ入力への電圧の印加は、必ず MCU 電源 (Vcc)を投入したあとに行ってください。

VCC ≧ Avcc ≧ AVRH
Avcc ≧ AN (アナログ印加電圧) ≧ Vss
を守ってください。

アナログ入力端子の入力インピーダンスについて A/D コンバータは、サンプルホールド回路を内蔵しており、A/D 変換の起動後、アナログ入力端子の電圧をサンプルホールド用コンデンサに取り込みます。したがって、アナログ入力の外部回路の出力インピーダンスが高い場合、アナログ入力サンプリング期間内にアナログ入力電圧が安定しないことがあります。そのため、外部回路の出力インピーダンスは十分低くしてください。

もしも、外部回路の出力インピーダンスを低くすることができない場合は、サンプリング時間を十分延ばしてください。

$|AVRH-AV_{SS}|$ が小さくなるにしたがって、相対的に誤差は大きくなります。

44.10 A/D コンバータの用語の定義

A/D コンバータの用語の定義について説明します。

分解能
A/D 変換器により識別可能なアナログ変化。

直線性誤差
ゼロトランジション点(00 0000 0000 ⇔ 00 0000 0001)と
フルスケールトランジション点(11 1111 1110 ⇔ 11 1111 1111) と
を結んだ直線と実際の変換特性との偏差。

微分直線性誤差
出力コードを 1LSB 変化させるのに必要な入力電圧の理想値からの偏差。

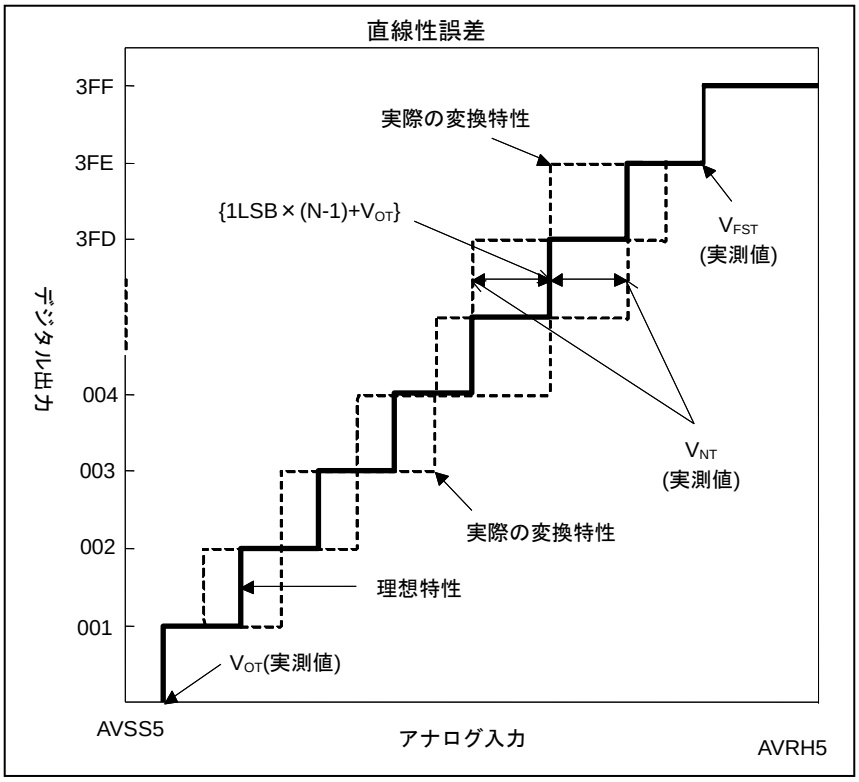
$$1LSB = \frac{V_{FST} - V_{OT}}{1022} \quad [V]$$

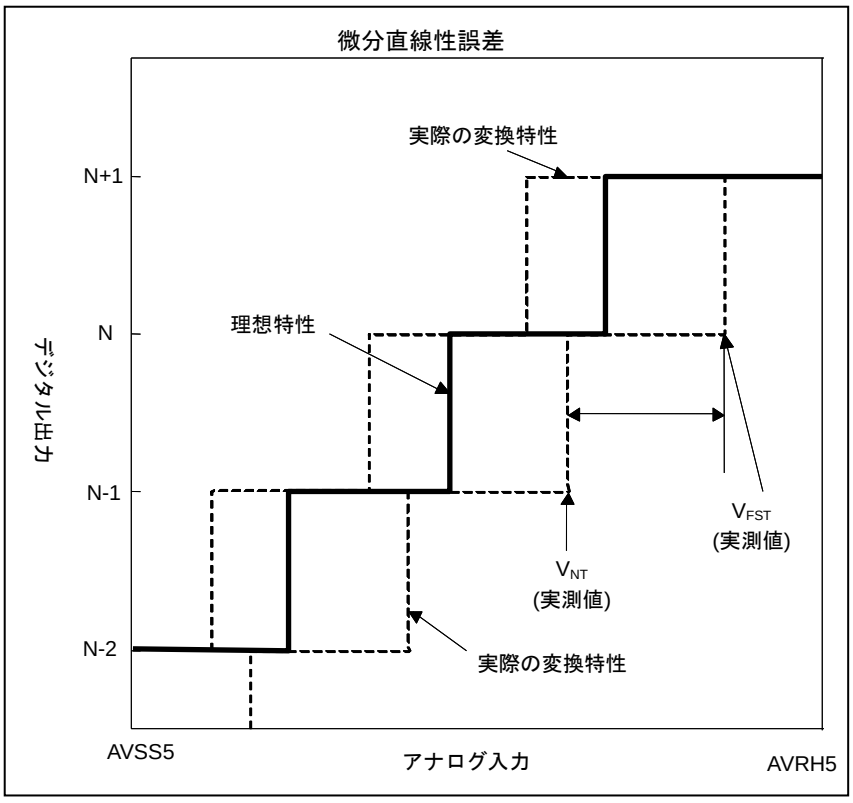
V_{OT} :デジタル出力が $(000)_H$ から $(001)_H$ に遷移する電圧
V_{FST}:デジタル出力が $(3FE)_H$ から $(3FF)_H$ に遷移する電圧

$$\text{デジタル出力 N の直線性誤差} = \frac{V_{NT} - \{1LSB \times (N-1) + V_{OT}\}}{1LSB} \quad [LSB]$$

$$\text{デジタル出力 N の微分直線性誤差} = \frac{V_{(N+1)T} - V_{NT}}{1LSB} - 1 \quad [LSB]$$

V_{NT}:デジタル出力が (N+1) から N に遷移する電圧


直線性誤差
3FF
3FE
3FD
004
003
002
001
デジタル出力
実際の変換特性
{1LSB × (N-1)+VOT}
VFST
(実測値)
VNT
(実測値)
実際の変換特性
理想特性
VOT(実測値)
AVSS5
アナログ入力
AVRH5


微分直線性誤差
N+1
N
N-1
N-2
デジタル出力
実際の変換特性
理想特性
VFST
(実測値)
VNT
(実測値)
実際の変換特性
AVSS5
アナログ入力
AVRH5

総合誤差

実際の値と理論値との差をいい、ゼロトランジション誤差/フルスケールトランジション誤差/直線性誤差を含む誤差。

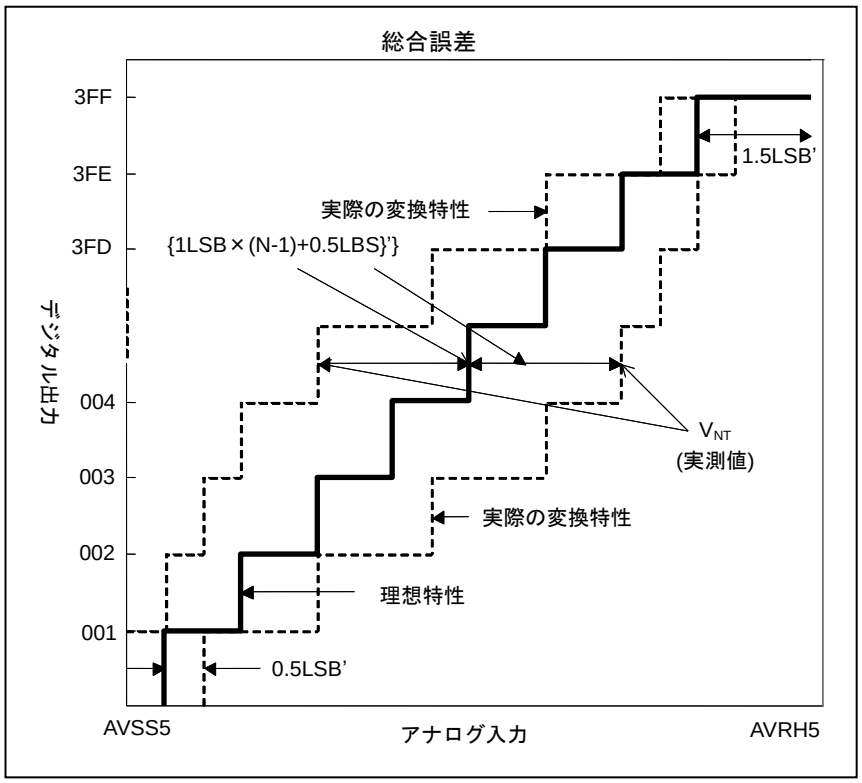
$$1LSB'(\text{理想値}) = \frac{AVRH - AV_{SS}}{1024} \quad [V]$$

$$V_{OT}'(\text{理想値}) = AV_{SS} + 0.5LSB' \quad [V]$$

$$V_{FST}'(\text{理想値}) = AVRH - 1.5LSB' \quad [V]$$

$$\text{デジタル出力 N の総合誤差} = \frac{V_{NT} - \{1LSB' \times (N-1) + 0.5LSB'\}}{1LSB'}$$

V_{NT}：デジタル出力が（N+1）から N に遷移する電圧



45. フラッシュメモリ

フラッシュメモリについて説明します。

45.1 概要

フラッシュメモリの概要について説明します。

本シリーズに内蔵されているフラッシュメモリの容量は

576K バイト(512K+64K バイト)　　：CY91F575
1088K バイト(1024K+64K バイト)　：CY91F577
1600K バイト(1536K+64K バイト)　：CY91F578
2112K バイト(2048K+64K バイト)　：CY91F579

です。ECC(Error Correction Code) が付加されています。

45.2 特長

フラッシュメモリの特長について説明します。

● 使用可能容量：

CY91F575 ： 576K バイト (大セクタ部 128K× 4 バイト+ 小セクタ部 16K×4 バイト)
CY91F577 ：1088K バイト (大セクタ部 128K× 8 バイト+ 小セクタ部 16K×4 バイト)
CY91F578 ：1600K バイト (大セクタ部 128K×12 バイト+ 小セクタ部 16K×4 バイト)
CY91F579 ：2112K バイト (大セクタ部 128K×16 バイト+ 小セクタ部 16K×4 バイト)

本シリーズには ECC 符号格納のため、上記に加え 4 バイトにつき 6 ビットのフラッシュメモリが搭載されています。

大容量セクタは 64K バイト、小容量セクタは 8K バイトで構成されますが、連続する領域を使用する場合には、2 つのセクタが交互に現れるので、大容量セクタ 128K バイト、小容量セクタ 16K バイト単位となります。

● 高速動作：

ワード (32 ビット) 単位での読出しが 80MHz・1 サイクルで可能。

● 外部からの書込み：

ROM ライタにより可能

● 動作モード：

① CPU-ROM モード
(CPU ／ DMA がフラッシュメモリにアクセス。読出しのみ)
② CPU プログラミングモード
(CPU がフラッシュメモリにアクセス。読出し・書き込み・消去)
③ フラッシュメモリモード (外部からフラッシュメモリにアクセス可能)

● CPU による読出し/ 書込み/ 消去 (自動アルゴリズム *) が可能

● ROM ライタによる読出し/ 書込み/ 消去 (自動アルゴリズム *) が可能

● セキュリティ機能

- 第三者によるフラッシュメモリ内容読出し阻止のため、セキュリティ ON 時に、外部からの命令フェッチ後の動作や、チップ消去以外の書込み・消去を抑止
- オンチップデバッガ(OCD) 使用により、パスワード認証後にセキュリティ ON 時の場合でも OCD を使用した外部からの読出し可能。

● ECC (Error Correction Code) 機能

- 1 ワード中の 1 ビットまでの誤りを訂正する ECC(Error Correction Code) セキュリティ機能があります。(2 ビット誤り検出機能は搭載していません。) 誤りは読出し中に自動で訂正されます。また、ECC 符号はフラッシュメモリへの書込み時に自動で付加されます。誤り訂正による読出しサイクルペナルティはありませんので、ソフトウェア開発の際に誤り訂正ペナルティを考慮する必要はありません。
- チップイレーズ/セクタイレーズされた状態でデータを読み出すとイレーズ状態"FFFF"が読み出されます。ただしこのとき、ECC エラー訂正の発生を示すフラッシュステータスレジスタ(FSTR)の FECCERR ビットがセットされます。

*: 自動アルゴリズム=Embedded Algorithm ™

45.3 構成

フラッシュメモリの構成について説明します。

45.3.1 ブロックダイヤグラム

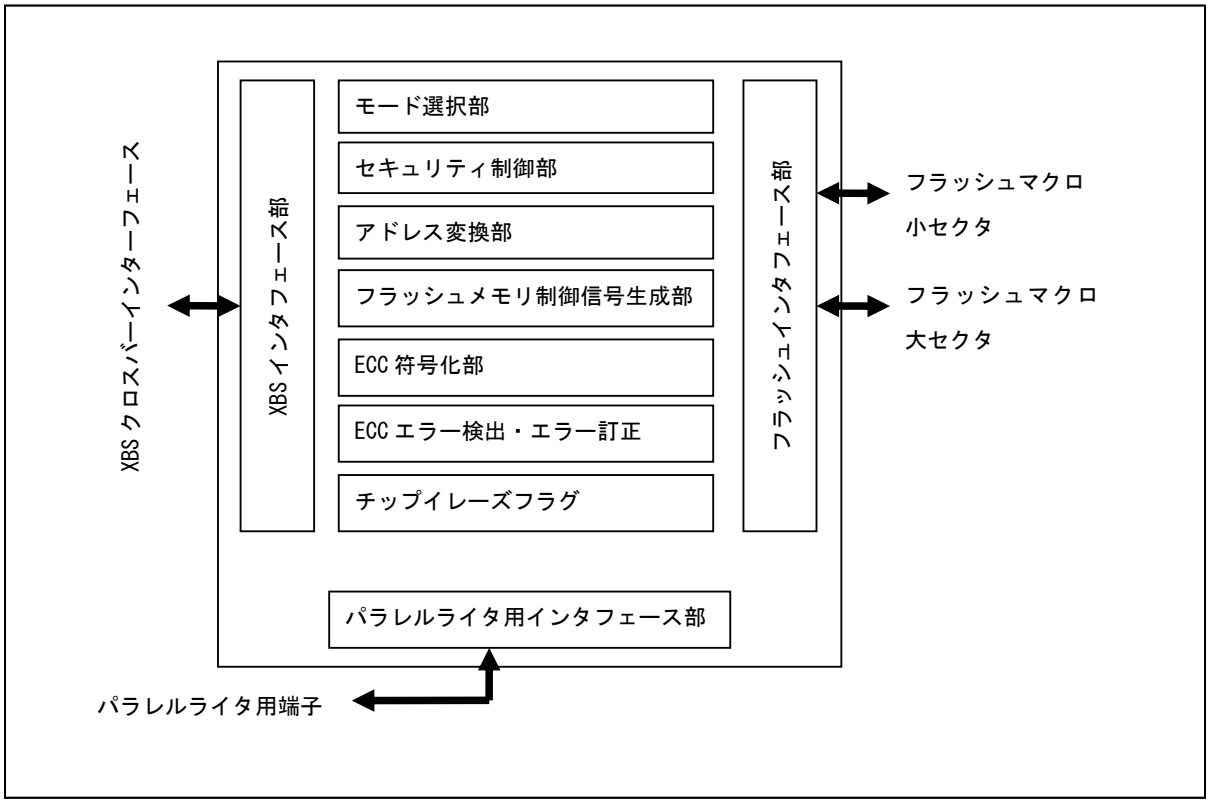
45.3.2 セクタ構成図

45.3.3 セクタ番号・フラッシュマクロ番号対応表

45.3.1 ブロックダイヤグラム

フラッシュメモリのブロックダイヤグラムについて示します。

図 45-1 ブロックダイヤグラム



45.3.2　セクタ構成図

フラッシュメモリのセクタ構成図について示します。

図 45-2 セクタ構成図 (1024KB+64KB)



図 45-3 セクタ構成図 (512KB+64KB)

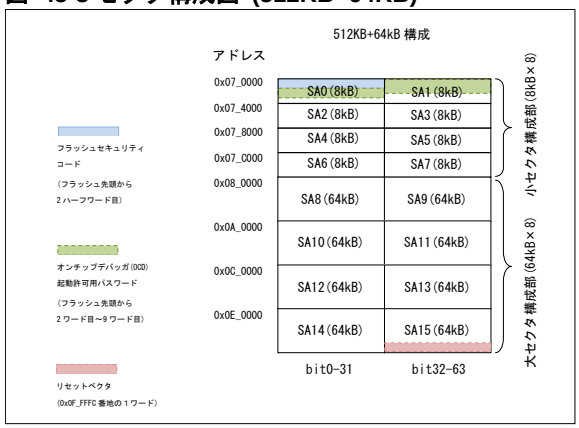


図 45-4 セクタ構成図 (1536KB+64KB)

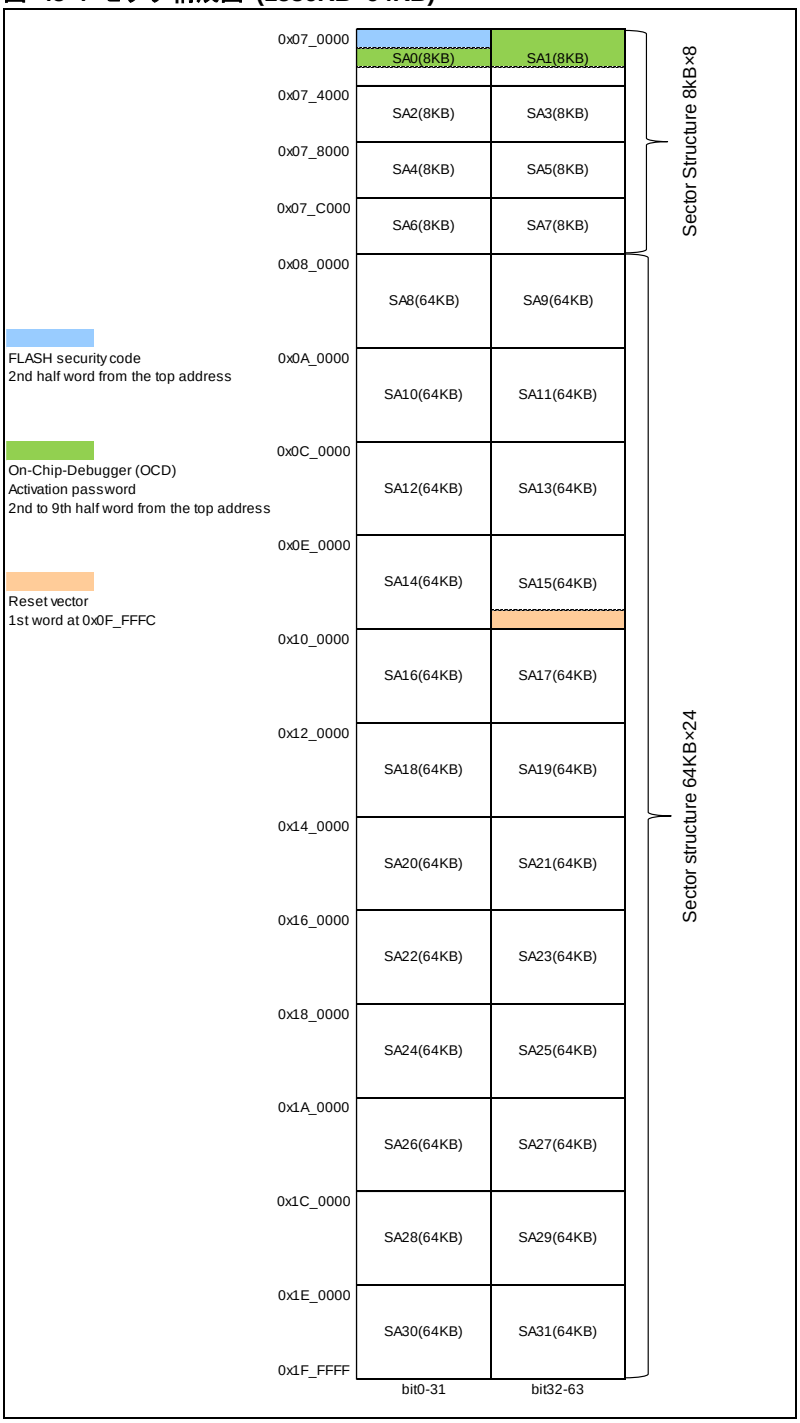
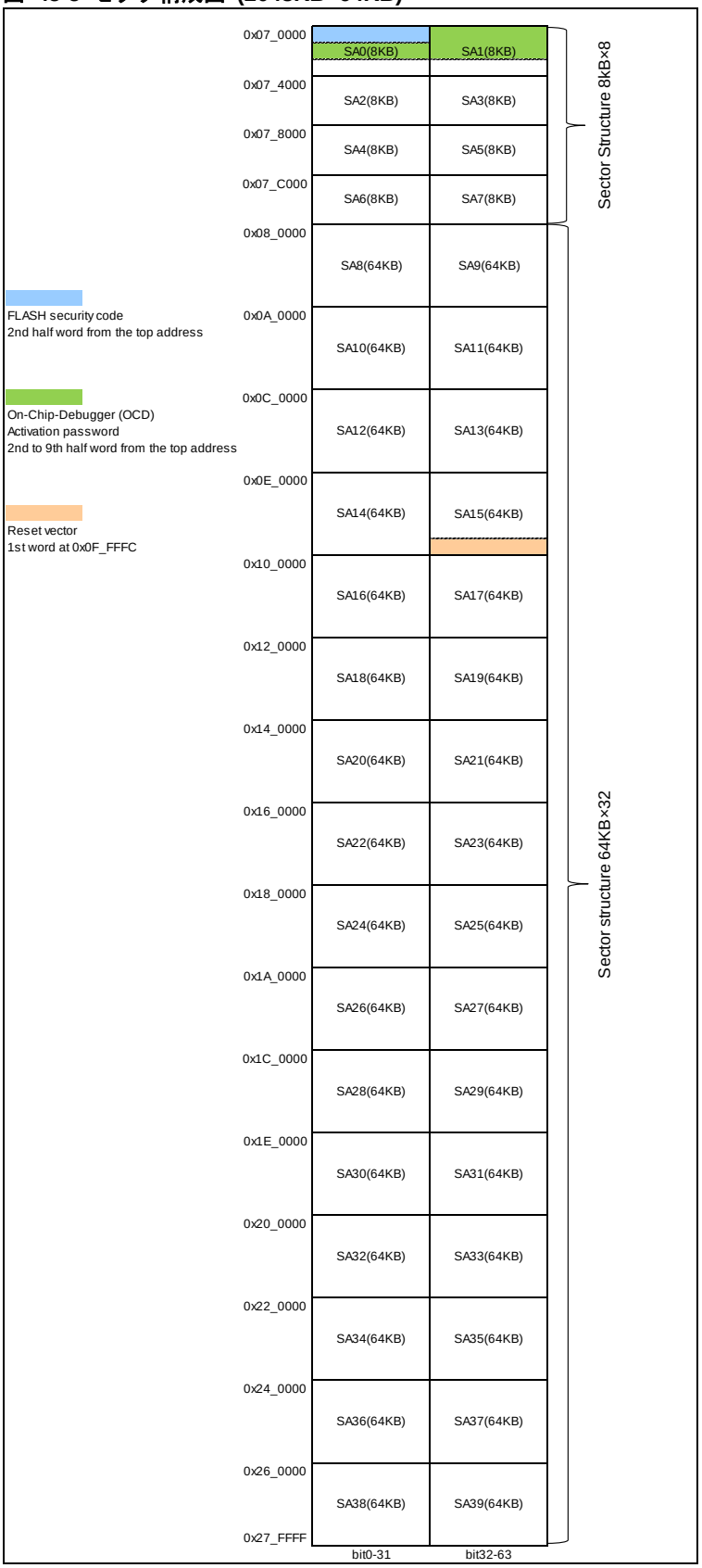


図 45-5 セクタ構成図 (2048KB+64KB)



＜注意事項＞

- FixedVector 機能により、リセットベクタは 0x0F_FFFC 番地に書き込まれた値ではなく、フラッシュメモリの先頭アドレス+0x0024 番地が返されます。詳細は『FixedVector 機能』の章を参照してください。
- オンチップデバッガ(OCD)起動許可用パスワード設定については、『オンチップデバッガ(OCD)』の章を参照してください。オンチップデバッガ(OCD)のセキュリティ機能を使用しない場合は、この領域には何も書き込まず、フラッシュ消去直後の初期状態(全ビット=1)のままとしてください。

45.3.3 セクタ番号・フラッシュマクロ番号対応表

フラッシュメモリのセクタ構成図について示します。

表 45-1 セクタ番号表

セクタ番号	アドレス	セクタサイズ	備考
SA0	0x07_0000 ～ 0x07_3FFB (下位 32bit)	8KB	フラッシュセキュリティコード領域 (0x07_0002 ～0x07_0003) デバッガ(OCD)起動許可用パスワード領域 (0x07_0008 ～0x07_000B, 0x07_0010 ～0x07_0013, 0x07_0018 ～0x07_001B, 0x07_0020 ～0x07_0023)
SA1	0x07_0004 ～ 0x07_3FFF (上位 32bit)	8KB	デバッガ(OCD)起動許可用パスワード領域 (0x07_0004 ～0x07_0007, 0x07_000C ～0x07_000F, 0x07_0014 ～0x07_0017, 0x07_001C ～0x07_001F)
SA2	0x07_4000 ～ 0x07_7FFB (下位 32bit)	8KB	
SA3	0x07_4004 ～ 0x07_7FFF (上位 32bit)	8KB	
SA4	0x07_8000 ～ 0x07_BFFB (下位 32bit)	8KB	
SA5	0x07_8004 ～ 0x07_BFFF (上位 32bit)	8KB	
SA6	0x07_C000 ～ 0x07_FFFB (下位 32bit)	8KB	
SA7	0x07_C004 ～ 0x07_FFFF (上位 32bit)	8KB	
SA8	0x08_0000 ～ 0x09_FFFB (下位 32bit)	64KB	
SA9	0x08_0004 ～ 0x09_FFFF (上位 32bit)	64KB	
SA10	0x0A_0000 ～ 0x0B_FFFB (下位 32bit)	64KB	
SA11	0x0A_0004 ～ 0x0B_FFFF (上位 32bit)	64KB	
SA12	0x0C_0000 ～ 0x0D_FFFB (下位 32bit)	64KB	
SA13	0x0C_0004 ～ 0x0D_FFFF (上位 32bit)	64KB	
SA14	0x0E_0000 ～ 0x0F_FFFB (下位 32bit)	64KB	
SA15	0x0E_0004 ～ 0x0F_FFFF (上位 32bit)	64KB	リセットベクタ位置 (0x0F_FFFC～0x0F_FFFF)
SA16	0x10_0000 ～ 0x11_FFFB (下位 32bit)	64KB	
SA17	0x10_0004 ～ 0x11_FFFF (上位 32bit)	64KB	

セクタ番号	アドレス	セクタサイズ	備考
SA18	0x12_0000 ～ 0x13_FFFB (下位 32bit)	64KB	
SA19	0x12_0004 ～ 0x13_FFFF (上位 32bit)	64KB	
SA20	0x14_0000 ～ 0x15_FFFB (下位 32bit)	64KB	
SA21	0x14_0004 ～ 0x15_FFFF (上位 32bit)	64KB	
SA22	0x16_0000 ～ 0x17_FFFB (下位 32bit)	64KB	
SA23	0x16_0004 ～ 0x17_FFFF (上位 32bit)	64KB	
SA24	0x18_0000 ～ 0x19_FFFB (下位 32bit)	64KB	
SA25	0x18_0004 ～ 0x19_FFFF (上位 32bit)	64KB	
SA26	0x1A_0000 ～ 0x1B_FFFB (下位 32bit)	64KB	
SA27	0x1A_0004 ～ 0x1B_FFFF (上位 32bit)	64KB	
SA28	0x1C_0000 ～ 0x1D_FFFB (下位 32bit)	64KB	
SA29	0x1C_0004 ～ 0x1D_FFFF (上位 32bit)	64KB	
SA30	0x1E_0000 ～ 0x1F_FFFB (下位 32bit)	64KB	
SA31	0x1E_0004 ～ 0x1F_FFFF (上位 32bit)	64KB	
SA32	0x20_0000 ～ 0x21_FFFB (下位 32bit)	64KB	
SA33	0x20_0004 ～ 0x21_FFFF (上位 32bit)	64KB	
SA34	0x22_0000 ～ 0x23_FFFB (下位 32bit)	64KB	
SA35	0x22_0004 ～ 0x23_FFFF (上位 32bit)	64KB	
SA36	0x24_0000 ～ 0x25_FFFB (下位 32bit)	64KB	
SA37	0x24_0004 ～ 0x25_FFFF (上位 32bit)	64KB	
SA38	0x26_0000 ～ 0x27_FFFB (下位 32bit)	64KB	
SA39	0x26_0004 ～ 0x27_FFFF (上位 32bit)	64KB	

45.4 レジスタ

フラッシュメモリのレジスタについて説明します。

表 45-2 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0840	FCTLR		予約	FSTR	フラッシュ制御レジスタ フラッシュステータスレジスタ
0x2308	FLIFCTLR	予約	FLIFFER1	FLIFFER2	フラッシュインタフェース制御レジスタ フラッシュインタフェースフィーチャ拡張レジスタ 1 フラッシュインタフェースフィーチャ拡張レジスタ 2

45.4.1 フラッシュ制御レジスタ ：FCTLR (Flash ConTroL Register)

フラッシュ制御レジスタのビット構成について示します。

Flash へのアクセス制御を設定します。

■ FCTLR：アドレス 0840H (アクセス: ハーフワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	FWE	予約		FSZ[1:0]		FAW[1:0]	
初期値	1	0	-	-	1	0	0	0
属性	R1,WX	R/W	RX,W0	RX,W0	R/W	R/W	R/W	R/W

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	FDSBL	予約		RDYF	予約			
初期値	0	-	-	0	-	-	-	-
属性	R/W	RX,W0	RX,W0	R/W	RX,W0	RX,W0	RX,W0	RX,W0

[bit15] 予約
予約ビットです。このビットは常に"1"が読み出されます。書込みは動作に影響しません。

[bit14] FWE (Flash Write Enable) Flash 書込み許可
Flash への書込み許可ビットです。このビットをセットして、CPU プログラミングモードに設定します。書込みが許可されたかどうかは FSTR:FRDY ビットで確認してください。

本ビットを設定すると、FLASH メモリへのデータフェッチに対し、ECC によるエラー検出・データ訂正機能が無効になります。

FWE	説明
0	Flash 書込み禁止(初期値)
1	Flash 書込み許可

＜注意事項＞
FLASH に書込みを行う場合、FLASH からの命令フェッチは禁止です。

[bit13, bit12] 予約
予約ビットです。読出し値は不定です。書込み時は"0"を書き込んでください。

[bit11, bit10] FSZ[1:0] (Flash write access SiZe) : Flash 書込みアクセスサイズ設定
CPU モード時の FLASH ライトアクセスサイズを指定します。必ず設定したビット幅で書き込んでください。読出しアクセスサイズは、本ビットの影響を受けません。読出し時 FLASH マクロに対しては常に 32bit Read を行います。FAW ビットにより wait サイクルを挿入した場合は、64bit 読出しアクセスとなります。

FSZ[1:0]	説明
00	8 ビット
01/10/11	16 ビット

[bit9, bit8] FAW[1:0] (FLASH Access Wait) : FLASH アクセス・ウェイト設定
CPU モード時の FLASH アクセスに対するウェイト・サイクルを設定します。フラッシュメモリの読出し時間は 12.5 ns なので、80MHz を超える動作周波数でフラッシュメモリにアクセスすることは禁止です。

FAW により wait サイクルを挿入する際は、クロックを高速にする前に、本ビットを設定してください。また、wait サイクルを削除する際は、クロックを低速にしてから、本ビットを設定してください。

FAW[1:0]	説明
00	0 サイクル (初期値)
01	1 サイクル
10/11	設定禁止

＜注意事項＞
本ビットにより、1wait サイクルを設定すると、ワイルドレジスタ機能が使用できません。ワイルドレジスタ機能を使用する場合は、コアの動作速度を 80MHz 以下とし、FAW ビットの設定値を 2'b00(0cycle)としてください。

[bit7] FDSBL (Flash DiSaBLe) : Flash Disable 指示
Flash をアクセス禁止状態(読出し・書込み共に)に設定します。

FDSBL	説明
0	Flash アクセス許可(初期値)
1	Flash アクセス禁止

[bit6, bit5] 予約
予約ビットです。読出し値は不定です。書込み時は"0" を書き込んでください。

[bit4] RDYF (ReaDY Flag) : 分岐アクセス時の RDY ネゲート指示
分岐アクセス時のウェイト・サイクル挿入を指示します。本ビットが"1"に設定されると、分岐アクセス時は、ウェイト・サイクルが挿入されます。これは、分岐した際の処理サイクルを合わせるためです。本ビットが"0"の場合は、分岐アクセス発生時、FLASH I/F の内部状態により、ウェイト・サイクルの制御が行われます。分岐アクセス受け付け時に、サイクルタイムを保障する必要がない場合、ウェイト・サイクルは挿入しません。サイクルタイムを保障する必要がある場合は、ウェイト・サイクルを挿入します。

RDYF	説明
0	FLASH I/F の状態に依る(初期値)
1	ウェイト・サイクル挿入

[bit3～bit0] 予約
予約ビットです。読出し値は不定です。書込み時は"0"を書き込んでください。

45.4.2 フラッシュステータスレジスタ ：FSTR (Flash STatus Register)

フラッシュステータスレジスタのビット構成について示します。

Flash の状態を表示します。

■ FSTR：アドレス 0843H (アクセス: バイト)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					FECCERR	FHANG	FRDY
初期値	-	-	-	-	-	0	0	1
属性	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	R,W	R,WX	R,WX

[bit7～bit3] 予約
予約ビットです。読出し値は不定です。書込みは動作に影響しません。

[bit2] FECCERR (Flash ECC Error coRRection) : データ読出し ECC 訂正発生
CPU 命令読出し以外でのフラッシュメモリ読出しの際に ECC エラー訂正が発生すると、本ビットがセットされます。"0" 書込みで本ビットをクリアします。

FECCERR	読出し	書込み
0	データ読出し時の ECC による訂正は発生していません(初期値)	本ビットをクリアします
1	データ読出し時に ECC エラー訂正が発生しました	動作に影響しません

1 ワード中に 2 ビット以上の誤りがある場合は、本ビットの読出し値は不定です。

CPU 命令読出しの際に ECC エラー訂正が発生しても、本ビットはセットされません。

ECC エラーと 0 書込みが同時に発生した場合は 0 書込みが優先されます。

[bit1] FHANG (Flash HANG) : Flash HANG 状態
フラッシュメモリの HANG 状態を示します。

FHANG	説明
0	通常状態
1	HANGUP 状態

タイミング超過(「[bit5] TLOV : (タイミングリミット超過フラグビット)」参照)すると HANG 状態になります。このビットが"1" になった場合はリセットコマンド(「45.5.3.1. コマンドシーケンス」参照)を発行してください。

自動アルゴリズムのコマンド発行直後は正しい値を読み出せない場合は、コマンド発行後 1 回目の本ビット読出し値は無視してください。

フラッシュメモリマクロを 2 つ以上搭載する品種においては、いずれかのマクロで HANGUP 状態となるとこのビットが"1"になります。
対象品種(搭載フラッシュメモリサイズが 1088KB を超える品種)

[bit0] FRDY (Flash ReaDY) : Flash 書込み許可
自動アルゴリズムでフラッシュメモリの書込み/ 消去動作が実行中か完了しているかを示します。動作中の場合、フラッシュメモリへデータを書き込んだりデータを消去したりすることはできません。

FRDY	説明
0	動作中(書込み/ 消去不可、ステータス読出し可能)
1	動作完了(書込み/ 消去可能、読出し可能)

自動アルゴリズムのコマンド発行直後に正しい値を読み出せない場合は、コマンド発行後 1 回目の本ビット読出し値は無視してください。

フラッシュメモリマクロを 2 つ以上搭載する品種においては、両方のマクロで書込み/消去動作が完了するとこのビットが"1"になります。
対象品種(搭載フラッシュメモリサイズが 1088KB を超える品種)

45.4.3 フラッシュインタフェース制御レジスタ ：FLIFCTLR(Flash I/F Control Register)

フラッシュインタフェース制御レジスタのビット構成について示します。

Flash I/F を制御します。プログラムフラッシュ、ワークフラッシュ共用のレジスタです。

■ FLIFCTLR：アドレス 2308H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			DFWDSBL	予約		ECCDSBL1	ECCDSBL0
初期値	-	-	-	0	-	0	0	0
属性	RX,WX	RX,WX	RX,WX	R/W	RX,WX	R/W0	R/W	R/W

[bit7～bit5] 予約
予約ビットです。読出し値は不定です。書込みは動作に影響しません。

[bit4] DFWDSBL (Data Fetch Wait cycle Disable) : データフェッチ・ウェイトサイクル無効
本ビットを"1"に設定すると、ウェイト設定時のデータフェッチ時に挿入されるウェイト・サイクルを無効にします。ただし、サイクルタイム保証のためのウェイト・サイクルは無効にすることはできません。

DFWDSBL	説明
0	ウェイト・サイクル有効(初期値)
1	ウェイト・サイクル無効

＜注意事項＞
FLIFCTLR.DFWDSBL ビットを"1"から"0"に変更する場合、かならず FCTLR.FAW="00"に 設定してから行ってください。

[bit3] 予約
予約ビットです。読出し値は不定です。書込みは動作に影響しません。

[bit2] 予約
予約ビットです。必ず"0"を書込んでください。

[bit1] ECCDSBL1(ECC Disable1) : ECC 機能無効 1
CPU モード時にワークフラッシュメモリへのライトアクセス、データフェッチに対し、ECC 機能の有効/無効を設定します。

ECCDSBL1	説明
0	ECC 機能有効(初期値)
1	ECC 機能無効

[bit0] ECCDSBL0(ECC Disable0) : ECC 機能無効 0
CPU モード時にプログラムフラッシュメモリへのライトアクセス、データフェッチに対し、ECC 機能の有効/無効を設定します。

ECCDSBL0	説明
0	ECC 機能有効(初期値)
1	ECC 機能無効

45.4.4 フラッシュインタフェースフィーチャ拡張レジスタ 1 : FLIFFER1(Flash I/F Feature Extension Register 1)

フラッシュインタフェースフィーチャ拡張レジスタ 1 のビット構成について示します。

予備レジスタです。本レジスタに書込む場合は 0xFF を書き込んでください。

■ FLIFFER1 : アドレス 230AH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							
初期値	1	1	1	1	1	1	1	1
属性	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1

[bit7～bit0] 予約
予約ビットです。0xFF を書き込んでください。

45.4.5 フラッシュインタフェースフィーチャ拡張レジスタ 2 : FLIFFER2(Flash I/F Feature Extension Register 2)

フラッシュインタフェースフィーチャ拡張レジスタ 2 のビット構成について示します。

予備レジスタです。本レジスタに書き込む場合は 0xFF を書き込んでください。

■ FLIFFER2 : アドレス 230BH (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							
初期値	1	1	1	1	1	1	1	1
属性	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1	R/W1

[bit7～bit0] 予約
予約ビットです。0xFF を書き込んでください。

45.5 動作説明

フラッシュメモリの動作について説明します。

45.5.1. アクセスモード設定

45.5.2. CPU によるフラッシュメモリ書込み

45.5.3. 自動アルゴリズム

45.5.4. リセットコマンド

45.5.5. 書込みコマンド

45.5.6. チップ消去コマンド

45.5.7. セクタ消去コマンド

45.5.8. セクタ消去一時停止コマンド

45.5.9. セキュリティ機能

45.5.10. フラッシュメモリの使用上の注意

45.5.1　アクセスモード設定

アクセスモードの設定について示します。

本シリーズのフラッシュメモリには以下の 3 モードがあります。本項にて①②の設定方法を説明します。③については、ご使用の ROM ライタの説明書を参照してください。

① CPU-ROM モード(CPU がフラッシュメモリにアクセス。読出しのみ、バイト/ハーフワード/ワードアクセス)
② CPU プログラミングモード(CPU がフラッシュメモリにアクセス。読み書き、Half-word アクセスのみ)
③ フラッシュメモリモード (外部からフラッシュメモリにアクセス可能)

45.5.1.1　CPU-ROM モードへの設定

CPU-ROM モードへの設定について示します。

フラッシュ制御レジスタ(FCTLR)の FWE ビットが"0"のとき、CPU-ROM モードです。CPU-ROM モードでは、フラッシュステータスレジスタ(FSTR)の FRDY ビットが"1"のとき、フラッシュメモリからの読出しが可能になります。CPU-ROM モードでは、フラッシュメモリへの書込みは不可能です。リセット解除後はこの状態になります。

45.5.1.2　CPU プログラミングモードへの設定

CPU プログラミングモードへの設定について示します。

フラッシュ制御レジスタ(FCTLR)の FWE ビットが"1"のとき、CPU プログラミングモードです。CPU プログラミングモードでは、フラッシュステータスレジスタ(FSTR)の FRDY ビットが"1"のとき、フラッシュメモリからの読出しおよび書込みが可能になります。

45.5.2　CPU によるフラッシュメモリ書込み

CPU によるフラッシュメモリ書込みについて説明します。

CPU プログラミングモードに設定した後、自動アルゴリズムを利用して消去・書込みを行います。本品種では 1 ワードごとの ECC(Error Correction Code) が付加されますので、1 ワードごとに書込みをしてください。以下の手順で 1 ハーフワードを 2 連続に、1 ワードごとに書込みをしてください。この手順に従わない場合、ECC 計算されずにフラッシュメモリに書き込まれてしまうため、書いた値を正しく読み出せません。

① Flash アクセスサイズ設定を 16 ビットに設定します。(FCTLR.FSZ[1:0]=01)

② 書込みコマンドを発行します。書込みアドレス= PA　書込みデータ= PD[31:16] 書込みコマンドについては「45.5.5. 書込みコマンド」を参照してください。

③ 書込みが終了するまで、ハードウェアシーケンスフラグを読出します。ハードウェアシーケンスフラグ読出しについては「45.5.3.2 自動アルゴリズム実行状態」を参照してください。

④ 書込みコマンドを発行します。書込みアドレス= PA+2　書込みデータ= PD[15:0]　この際、ハードウェアが自動で ②の PD[31:16]とあわせて ECC 符号の計算を行い、ECC 符号の書込みも自動で同時に行われます。

⑤ 書込みが終了するまで、ハードウェアシーケンスフラグを読み出します。

⑥ 書込みデータがまだある場合は②に戻ります。すべて書込み完了した場合は、⑦へ進みます。

⑦ CPU-ROM モードに設定します。

⑧ 書き込んだ値を読み出して、正しい値を読み出せるか確認してください。また、正しい値が読みだせた場合でも FSTR:FECCERR ビットを確認して ECC 訂正がされていないか確認してください。ECC 訂正が発生していた場合はフラッシュメモリ消去からやり直してください。

PA　　：書込み対象アドレス(ワードアラインド)
PD[31:0]　：書込みデータ
PD[31:16]　：書込みデータ上位 16 ビット分
PD[15:0]　：書込みデータ下位 16 ビット分

45.5.3 自動アルゴリズム

自動アルゴリズムについて説明します。

CPU プログラミングモードを利用する場合、フラッシュメモリへの書込み/ 消去は自動アルゴリズムを起動して行います。本項にて自動アルゴリズムについて説明します。

45.5.3.1 コマンドシーケンス

コマンドシーケンスについて説明します。

フラッシュメモリへ1回～6回連続でハーフワード(16 ビット)のデータを書き込むと自動アルゴリズムが起動します。これをコマンドとよびます。コマンドシーケンスを以下に示します。

表 45-3 コマンドシーケンス

コマンド	書込み回数	1 回目		2 回目		3 回目		4 回目		5 回目		6 回目	
		アドレス [11:0]	データ [7:0]	アドレス [11:0]	データ [7:0]	アドレス [11:0]	データ [7:0]	アドレス [11:0]	データ [7:0]	アドレス [11:0]	データ [7:0]	アドレス [11:0]	データ [7:0]
リセット	1	任意	$F0_H$										
読出し	1	RA	RD										
書込み	4	$x554_H$	AA_H	$yAA8_H$	55_H	$x554_H$	$A0_H$	PA	PD				
チップ消去	6	$x554_H$	AA_H	$yAA8_H$	55_H	$x554_H$	80_H	$x554_H$	AA_H	$yAA8_H$	55_H	$x554_H$	10_H
セクタ消去	6	$x554_H$	AA_H	$yAA8_H$	55_H	$x554_H$	80_H	$x554_H$	AA_H	$yAA8_H$	55_H	SA	30_H
セクタ消去一時停止	1	任意	$B0_H$										
セクタ消去再開	1	任意	30_H										

* 表中のデータ表記は下位 8 ビット分のみを表記しています。上位 8 ビットは任意です。コマンドはハーフワードまたはバイトで書込んでください。
* 表中のアドレス表記は下位 16 ビット分のみです。上位 16 ビット分は、対象となるフラッシュマクロのアドレス範囲のうちの任意のアドレスを指定してください。

x：1,3,5,7,9,B,D,F
y：0,2,4,6,8,A,C,E

PA：書込みアドレス (ハーフワードアラインド)
PD：書込みデータ (ハーフワードで書き込んでください。)
SA：セクタアドレス (消去対象となるセクタのアドレス範囲の内の任意のアドレスを指定してください。)
RA：読出しアドレス
RD：読出しデータ (読出し幅は任意です。)

＜注意事項＞
コマンドアドレス、セクタ消去コマンド発行時に入力するセクタアドレス(SA),の最下位 2 ビット分は以下のようにしてください。

- ハーフワードアクセス時: 2'b00
- バイトアクセス時: 2'b01 または 2'b11

例 1 バイトアクセス時、コマンドアドレス=(標準コマンドアドレスの最下位 2 ビットを 2'b01 に変更)とする場合

$yAA8_H \rightarrow yAA9_H$、$x554_H \rightarrow x555_H$、SA → {SA[31:2], 2'b01}
(SA:セクタ消去コマンド発行時に入力する消去対象セクタ内の任意のアドレス)

例 2 バイトアクセス時、コマンドアドレス=(標準コマンドアドレスの最下位 2 ビットを 2'b11 に変更)とする場合

$yAA8_H \rightarrow yAAB_H$、$x554_H \rightarrow x557_H$、SA → {SA[31:2], 2'b11}
(SA:セクタ消去コマンド発行時に入力する消去対象セクタ内の任意のアドレス)

＜注意事項＞
誤ったアドレス値やデータ値を書き込んだ場合や、誤ったシーケンスで書込みした場合、それまで書き込んだコマンドはクリアされます。

■ リセットコマンド

リセットコマンドを対象 FLASH メモリに送るとそれまでに入力した**エラー! 参照元が見つかりません。**に示す各コマンド入力をキャンセルし、再び 1 回目からコマンド入力し直すことができます。

ただし、各コマンドを最後まで入力し自動アルゴリズムが起動すると、本リセットコマンドでは自動アルゴリズムを中止することはできません。

自動アルゴリズムの実行がタイミングリミットを超過した場合のみ、リセットコマンドを入力するとフラッシュメモリがリセット状態へ復帰します。

■ 読出しコマンド

読出しコマンドを対象セクタに送るとフラッシュメモリを読み出せます。読出しコマンドを発行すると、フラッシュメモリは、他のコマンドが発行されるまで読出し状態を保ちます。

■ プログラム（書込み）コマンド

書込みコマンドを対象セクタに 4 回連続して送ると自動アルゴリズムを起動してフラッシュメモリにデータを書き込めます。データの書込みはどのようなアドレスの順番でもセクタの境界を越えても行えます。

CPU プログラミングモードでは、ハーフワード/バイトで書込みが行えます。4 回目の書込みが終了すると、自動アルゴリズムが起動し、フラッシュメモリへの自動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、外部からフラッシュメモリを制御する必要はありません。

実際の動作については、「45.5.5. 書込みコマンド」を参照してください。

＜注意事項＞

- ハーフワードで書き込む場合、4 回目の書込みコマンド(書込みデータサイクル)を奇数番地に書き込むと、書込みが正しく行われません。必ず偶数番地に書込んでください。
- 1 回の書込みコマンドシーケンスではハーフワードのデータ 1 つしか書き込めません。複数のデータを書き込みたい場合は、1 データに 1 度書込みコマンドシーケンスを発行してください。
- セキュリティ ON 時においては、フラッシュの書込みには制限があります。詳細は「5.9.4 セキュリティ ON 時のフラッシュアクセス制限」を参照してください。

■ チップ消去コマンド

チップ消去コマンドを対象セクタに 6 回連続して送るとフラッシュメモリの全セクタを一括で消去できます。6 回目の書込みが終了すると、自動アルゴリズムが起動しチップ消去動作が開始されます。自動消去アルゴリズムが起動するとすべてのチップ消去する前に、フラッシュメモリがチップ内のすべてのセルに"0" を書込んで、マージンを検証(プリプログラム)するため、チップ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は、外部からフラッシュメモリを制御する必要はありません。

実際の動作については、「45.5.6. チップ消去コマンド」を参照してください。

■ セクタ消去コマンド

セクタ消去コマンドを対象セクタに 6 回連続して送るとフラッシュメモリのセクタを消去できます。6 回目の書込みが終了し、40µs 経過(タイムアウト期間)すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。複数のセクタを消去したい場合は、40µs(タイムアウト期間)以内に消去するセクタのアドレスに消去コード(30_H)を書き込んでください。タイムアウト期間内に次のセクタが入力されないと、セクタ消去コマンドが無効になる場合があります。自動消去アルゴリズムが起動するとセクタ消去する前に、フラッシュメモリが消去するセクタのセルに"0" を書き込んで、マージンを検証(プリプログラム)するため、セクタ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は，外部からフラッシュメモリを制御する必要はありません。

実際の動作については、「45.5.7. セクタ消去コマンド」を参照してください。

＜注意事項＞

セキュリティ ON 時においては、フラッシュのセクタ消去には制限があります。詳細は「5.9.4 セキュリティ ON 時のフラッシュアクセス制限」を参照してください。

■ セクタ消去一時停止コマンド

セクタイレーズ実行中またはコマンドタイムアウト中に、セクタ消去一時停止コマンドを送ると、セクタ消去一時停止状態（セクタイレーズサスペンド状態）に移行できます。

セクタ消去一時停止状態では、消去対象でないセクタのメモリセルの読出し動作が可能となります。ただし、新たな書込みおよび消去コマンドは受け付けられません。

セクタ消去一時停止状態から、中断していた消去動作を再開するには、消去再開コマンドを送ります。

消去再開コマンドが受け付けられると、セクタ消去状態に戻り消去動作を再開します。

コマンドタイムアウト状態から本状態に遷移した場合であっても、イレーズレジュームコマンドが正常にライトされると、コマンドタイムアウト状態には遷移せず、セクタ消去状態に遷移して直ちにセクタ消去動作を再開します。

実際の動作については、「45.5.8 セクタ消去一時停止コマンド」を参照してください。

＜注意事項＞

セクタ消去一時停止コマンドの発行からセクタ消去動作を停止して消去対象でないセクタからの読出しが可能となるまで最大 16.7µs 要します。

読出し可能状態になったかどうかは、FLASH ステータスレジスタ(FSTR)の FRDY ビットまたはハードウェアシーケンスフラグの TOGG1 で確認できます。

45.5.3.2 自動アルゴリズム実行状態

自動アルゴリズム実行状態について説明します。

フラッシュメモリでは、書込みや消去を自動アルゴリズムで行うため、自動アルゴリズムが実行中かどうかを FLASH ステータスレジスタ(FSTR)の FRDY ビットで、動作状態をハードウェアシーケンスフラグで確認できます。

■ ハードウェアシーケンスフラグ

自動アルゴリズムの状態を示すフラグです。FLASH ステータスレジスタ(FSTR)の FRDY ビットが"0" のときに、フラッシュメモリの任意のアドレスを読み出すと動作状態を確認できます。ハードウェアシーケンスフラグのビット構成を次に示します。

図 45-6 ハードウェアシーケンスフラグのビット構成

ハーフワードアクセスの場合

bit15	bit14	bit13	bit12	bit11	bit10	bit 9	bit 8
未定義	未定義	未定義	未定義	未定義	未定義	未定義	未定義
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
DPOLL	TOGG1	TLOV	未定義	SETI	TOGG2	未定義	未定義

バイトアクセスの場合

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
DPOLL	TOGG1	TLOV	未定義	SETI	TOGG2	未定義	未定義

＜注意事項＞

- ワードアクセスで読み出すことはできません。必ず CPU プログラミングモードのときに、ハーフワードかバイトアクセスで読み出してください。
- CPU ROM モードで任意のアドレスを読み出しても、ハードウェアシーケンスフラグを読み出すことはできません。

● 各ビットとフラッシュメモリの状態

ハードウェアシーケンスフラグの各ビットの状態とフラッシュメモリ状態の対応を次に示します。

表 45-4 フラグとフラッシュメモリ状態の対応

状態		DPOLL	TOGG1	TLOV	SETI	TOGG2
実行中	書込み中	反転データ (*1)	トグル	0	0	-
	セクタ/チップ消去中	0	トグル	0	1	-
タイムリミット超過	書込みコマンド	反転データ (*1)	トグル	1	0	-
	セクタ/チップ消去コマンド	0	トグル	1	1	-
セクタ消去一時停止	消去対象セクタ	-	-	-	-	トグル

*1 読み出される値については、「 ● ビット説明」を参照してください。

● ビット説明

[bit15～bit8] 未定義ビット

[bit7] DPOLL : データポーリングフラグビット
書込み/消去対象アドレスを指定してハードウェアシーケンスフラグを読み出すと、データポーリング機能で自動アルゴリズムが実行中かどうかをこのビットに示します。

読み出される値は動作状態によって異なります。

(1) 書込み時

書込み中：	最後に書き込まれたデータの bit7 の値と逆の値（反転データ） が読み出されます。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。
書込み終了後：	ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit7 の値が読み出されます。

(2) セクタ消去時

セクタ消去実行中：	消去中のセクタから"0" が読み出されます。
セクタ消去後：	必ず"1" が読み出されます。

(3) チップ消去時

チップ消去実行中：	必ず"0" が読み出されます。
チップ消去後：	必ず"1" が読み出されます。

(4) セクタ消去一時停止時

一時停止状態（未完了）：	セクタ消去一時停止セクタから、"0"が読み出されます。
セクタ消去動作完了：	セクタ消去一時停止セクタから、"1"が読み出されます。

＜注意事項＞
自動アルゴリズムを起動しているときは、指定したアドレスのデータを読み出すことはできません。このビットで自動アルゴリズムの動作が終了していることを確認してから、データを読み出してください。

[bit6] TOGG1 : トグルフラグ 1 ビット
任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムが実行中かどうかをこのビットで示します。読み出される値は動作状態によって異なります。

書込み/ セクタ消去/ チップ消去時

書込み/ セクタ消去/ チップ消去中：	連続でこのビットを読み出すと、"1"と"0"が交互に読み出されます(トグル動作)。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。
書込み/ セクタ消去/ チップ消去終了後：	ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit6 の値が読み出されます。

[bit5] TLOV :**タイミングリミット超過フラグビット**
任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムの実行時間がフラッシュメモリ内部で規定している時間(内部パルスの回数)を超過したかどうかをこのビットで示します。読み出される値は動作状態によって異なります。

書込み/ セクタ消去/ チップ消去時
次の値が読み出されます。

"0"	**規定時間内**
"1"	**規定時間を越えている**

このビットが"1" のときに、DPOLL ビットや TOGG1 ビットが自動アルゴリズム実行中であることを示していると、書込みや消去に失敗したことになります。

例えば、フラッシュメモリでは"0" が書き込まれているデータを"1" に書き換えることができないため、 "0"が書き込まれているアドレスに"1"を書き込もうとすると、フラッシュメモリがロックされ自動アルゴリズムが終了しません。この場合は、DPOLL ビットの値は無効のままになり、TOGG1 ビットからは"1" と"0" が交互に読み出され続けます。この状態のまま規定時間を超えたときに、このビットが"1" に変わります。このビットが"1" になった場合はリセットコマンドを発行してください。

＜注意事項＞
このビットが"1" の場合は、フラッシュメモリが正しく使用されなかったことを示しています。フラッシュメモリの不良ではありません。リセットコマンドを発行してから適切な処理を行ってください。

[bit4] **未定義ビット**

[bit3] SETI :**セクタ消去タイマフラグビット**
セクタ消去時は、セクタ消去コマンドを発行してから実際にセクタ消去が開始されるまでには、40µs のタイムアウト期間が必要です。任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、セクタ消去コマンドのタイムアウト期間中かどうかをこのビットで示します。読み出される値は動作状態によって異なります。

セクタ消去時 :
セクタ消去時に次のセクタ消去コードを入力する前に、このビットを確認することで次のセクタ消去コードが受け付けられる状態かどうかを確認できます。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスせず、 次の値が読み出されます。

"0"	セクタ消去ウェイト期間中(次のセクタ消去コード(0x30) を受け付けられます。)
"1"	セクタ消去ウェイト期間を超過している(このときに、DPOLL ビットや TOGG1 ビットが自動アルゴリズム実行中であることを示していると、フラッシュメモリ内部の消去が開始されています。この場合、セクタ消去コード(0x30) 以外のコマンドはフラッシュメモリ内部の消去が完了するまで無視されます。)

[bit2] TOGG2 : トグルフラグ 2 ビット

セクタ消去サスペンド状態において、消去対象でないセクタに対しては読出し(リード)を行うことができますが、消去対象セクタに対してはリードができません。本フラグは、セクタ消去一時停止中でリードアドレスが消去対象セクタの場合、出力データがトグルして、消去対象セクタであることを示します。

消去対象セクタへの読出し	連続でこのビットを読み出すと、"1"と"0"が交互に読み出されます(トグル動作)。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。
消去対象でないセクタへの読出し	指定したアドレスのデータを読み出します。

[bit1,bit0] **未定義ビット**

45.5.4　リセットコマンド

リセットコマンドについて説明します。

リセットコマンドを対象 FLASH メモリに送るとフラッシュメモリをリセット状態にできます。この状態は、フラッシュメモリの初期状態のため、電源を投入したときやコマンドが正常終了すると、フラッシュメモリは常にリセット状態に戻ります。電源投入時はデータ読出しコマンドを発行する必要はありません。また、リセット状態では通常の読出しアクセスでデータを読み出したり、CPU からプログラムアクセスしたりできるため、データを読み出すときにリセットコマンドを発行する必要はありません。

45.5.5　書込みコマンド

書込みコマンドについて示します。

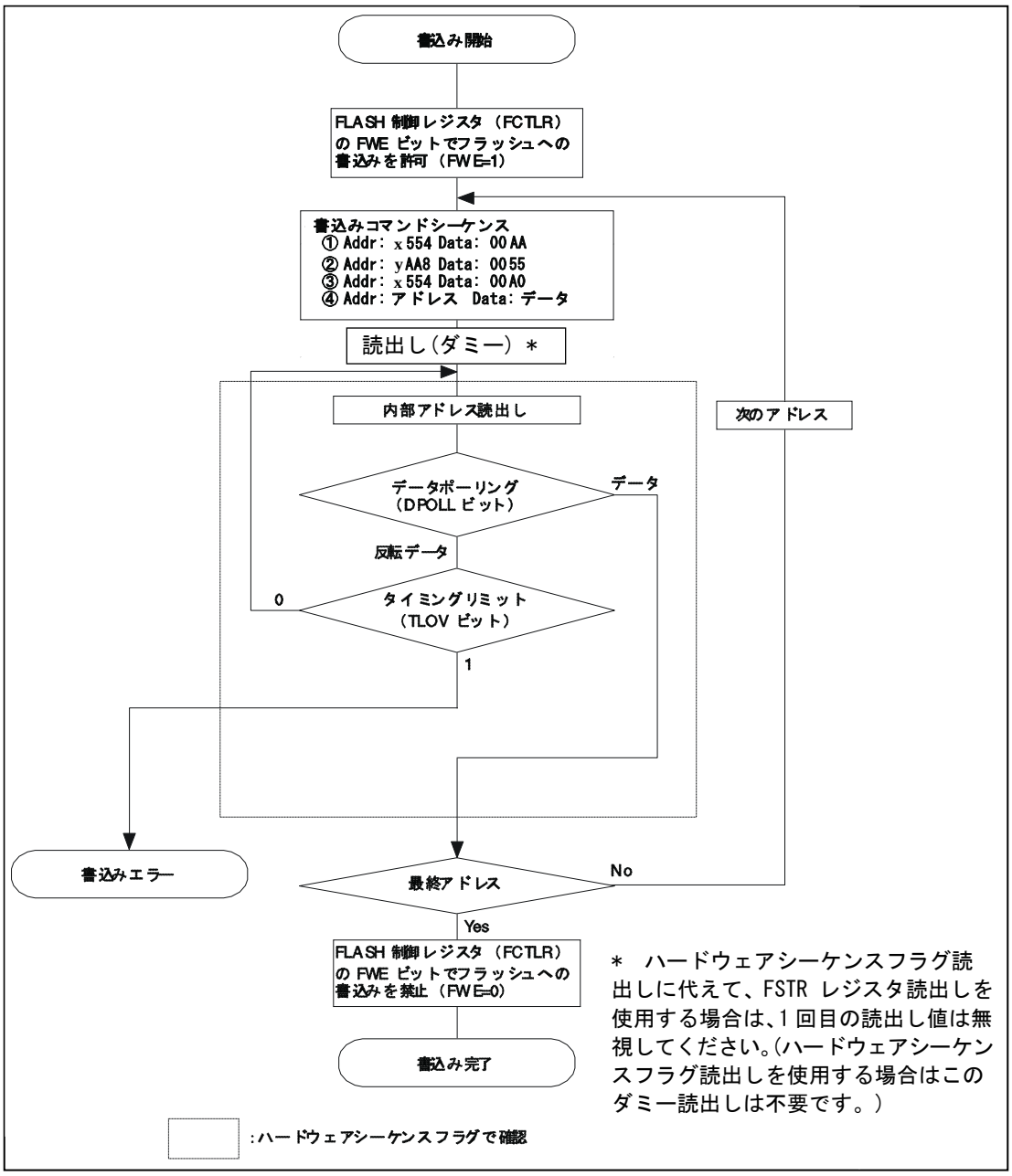
次の順番で書込みを行います。

①書込みコマンドを対象セクタに連続して送る
自動アルゴリズムが起動されフラッシュメモリへデータが書き込まれます。書込みコマンド発行後は外部からフラッシュメモリを制御する必要はありません。

②書込みを行ったアドレスにリードアクセスする
読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7(DPOLL ビット)が書き込んだ値と一致していると、フラッシュメモリへの書込みが終了したことになります。書込みが終了していない場合は、最後に書き込んだデータの bit7 の値と逆の値(反転データ)が読み出されます。

フラッシュメモリへの書込み動作例を次に示します。

図 45-7 書込み手順例


書込み開始
FLASH 制御レジスタ（FCTLR）の FWE ビットでフラッシュへの書込みを許可（FWE=1）
書込みコマンドシーケンス
① Addr: x554 Data: 00AA
② Addr: yAA8 Data: 0055
③ Addr: x554 Data: 00A0
④ Addr: アドレス Data: データ
読出し（ダミー）＊
内部アドレス読出し
次のアドレス
データポーリング（DPOLL ビット）
データ
反転データ
タイミングリミット（TLOV ビット）
0
1
書込みエラー
最終アドレス
No
Yes
FLASH 制御レジスタ（FCTLR）の FWE ビットでフラッシュへの書込みを禁止（FWE=0）
書込み完了
:ハードウェアシーケンスフラグで確認
＊ ハードウェアシーケンスフラグ読出しに代えて、FSTR レジスタ読出しを使用する場合は、1 回目の読出し値は無視してください。（ハードウェアシーケンスフラグ読出しを使用する場合はこのダミー読出しは不要です。）

＜注意事項＞

- 書込みが終了すると、フラッシュメモリは読出しモードに戻るため、書込みアドレスを受け付けなくなります。
- 書込みコマンドについては、「45.5.3 自動アルゴリズム」を参照してください。
- ハードウェアシーケンスフラグの DPOLL ビットは、TLOV ビットと同時に値が変わるので TLOV ビットが"1" の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG1 ビット、TLOV ビットが"1"に変わると同時にトグル動作を停止します。そのため、TLOV ビットが"1"の場合でも TOGG1 ビットを再度確認する必要があります。
- フラッシュメモリへは、どのようなアドレスの順番でも、またセクタの境界を越えても書き込めますが、1 回の書込みコマンドシーケンスではハーフワードのデータ 1 つしか書き込めません。複数のデータを書き込みたい場合は、1 データに 1 度書込みコマンドシーケンスを発行してください。
- 一度、 "0"が書き込まれたデータを"1"に戻すことはできません。"0"を"1"に書き換えると、以下のいずれかになります。
  - □ 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
  - □ 書込み規定時間を超え、ハードウェアシーケンスフラグビットの TLOV ビットが"1"に変わる
  - □ "1"が書き込まれたように見える
    ただし、"1"が書き込まれたように見えた場合でも、実際のデータは"0"のままのため読出し/ リセットモードでデータを読み出すと"0" が読み出されます。データを"1"に戻したい場合は、チップ消去かセクタ消去を行ってください。
- 書込み動作中はフラッシュメモリに書き込まれたすべてのコマンドが無視されます。
- 書込み動作中に本デバイスがリセットされた場合は、書き込んでいるデータは保証されません。
- 本シリーズでは ECC ビット付加のため、必ず 16 ビットを 2 回、32 ビットで書込みを行う必要があります。手順については「45.5.2. CPU によるフラッシュメモリ書込み」を参照してください。

45.5.6　チップ消去コマンド

チップ消去コマンドについて示します。

チップ消去コマンドにて、フラッシュメモリの消去対象フラッシュマクロを一括して消去できます。

チップ消去コマンドを対象 FLASH メモリに連続して送ると自動アルゴリズムを起動して、全セクタを一括で消去できます。チップ消去コマンドについては、「45.5.3 自動アルゴリズム」を参照してください。

① チップ消去コマンドを消去対象フラッシュマクロのセクタに連続して送る
自動アルゴリズムが起動されフラッシュメモリへデータが書き込まれます。

② 消去対象フラッシュマクロの任意のアドレスにリードアクセスする
読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOLL ビット)が"1" だと、チップ消去が終了したことになります。

チップ消去に必要な時間は「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。
チップ消去動作が終了すると、フラッシュメモリは読出し/リセットモードに戻ります。

＜注意事項＞

- 自動消去アルゴリズムが起動するとすべてのチップ消去する前に、フラッシュメモリがチップ内のすべてのセルに"0" を書き込んで、マージンを検証(プリプログラム)するため、チップ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は外部からフラッシュメモリを制御する必要もありません。
- セキュリティ ON 時においては、フラッシュの消去には手順に制限があります。詳細は「45.5.9.3 フラッシュセキュリティ解除方法」を参照してください。
- 1 回の消去コマンドで消去されるフラッシュメモリマクロは 1 つです。フラッシュメモリマクロを 2 つ以上搭載する品種においては、それぞれのマクロに消去コマンドが必要です。
  対象品種（搭載フラッシュメモリサイズが 1088KB を超える品種）

45.5.7　セクタ消去コマンド

セクタ消去コマンドについて示します。

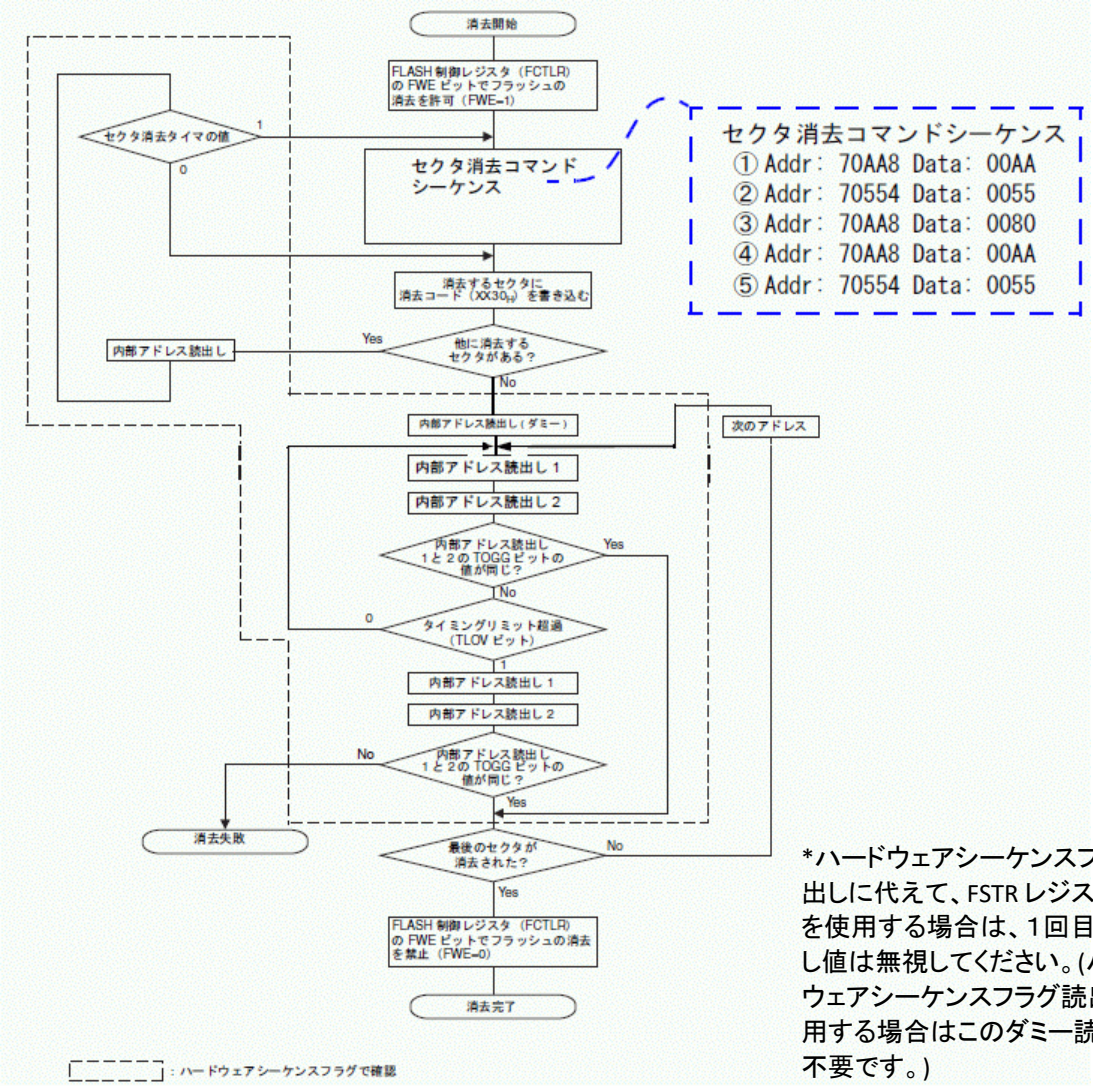
フラッシュメモリ内のセクタを選択して、選択したセクタのデータのみを消去できます。複数のセクタを同時に指定することもできます。次の順番でセクタ消去を行います。

① セクタ消去コマンドを対象セクタに連続して送る
40µs 経過(タイムアウト期間)すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。複数のセクタを消去したい場合は、40µs(タイムアウト期間)以内に消去するセクタのアドレスに消去コード(30_H)を書き込んでください。タイムアウト期間経過後に書き込んでも、セクタ消去コマンドが無効になる場合があります。

②任意のアドレスにリードアクセスする
読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOLL ビット)が"1" だと、セクタ消去が終了したことになります。また、TOGG1 ビットを利用してセクタ消去が完了したかどうかを確認することもできます。

確認動作に TOGG1 ビットを使用した場合を例にとって、セクタ消去手順例を次に示します。

図 45-8 セクタ消去手順例



*ハードウェアシーケンスフラグ読出しに代えて、FSTR レジスタ読出しを使用する場合は、1回目の読出し値は無視してください。(ハードウェアシーケンスフラグ読出しを使用する場合はこのダミー読出しは不要です。)

＜注意事項＞

- セクタ消去に必要な時間は「(セクタ消去時間＋セクタ書込み時間（プリプログラム)) ×セクタ数」となります。
- セクタ消去動作が終了すると、フラッシュメモリは読出し/ リセットモードに戻ります。
- セクタ消去コマンドについては、「45.5.3 自動アルゴリズム」を参照してください。
- ハードウェアシーケンスフラグの DPOLL ビットは、 TLOV ビットと同時に値が変わるので TLOV ビットが"1" の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG1 ビットは、 TLOV ビットが"1" に変わると同時にトグル動作を停止します。そのため、 TLOV ビットが"1" の場合でも TOGG1 ビットを再度確認する必要があります。
- タイムアウト期間を含むセクタ消去中に、セクタ消去コマンド以外のコマンドを発行すると、フラッシュメモリが読出し / リセット状態になります。この場合、フラッシュメモリがリセットされるので、コマンドが発行される 1 つ前または複数のセクタ消去コマンドが無効になります。セクタ消去を行う場合は、最初からセクタ消去コマンドを発行しなおしてください。
- 自動消去アルゴリズムが起動すると、セクタ消去する前に、フラッシュメモリが消去するセルに"0" を書き込んで、マージンを検証（プリプログラム）するため、セクタ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は外部からフラッシュメモリを制御する必要もありません。

45.5.8　セクタ消去一時停止コマンド

セクタ消去一時停止コマンドについて示します。

セクタ消去実行中またはコマンドタイムアウト中に、セクタ消去を一時的に停止させることができます。

セクタ消去一時停止状態では、消去対象でないセクタのメモリセルの読出し動作が可能となります。ただし、新たな書込みおよび消去コマンドは受け付けられません。

セクタ消去を一時停止させるには、対象 FLASH マクロの任意のアドレスにセクタ消去一時停止コマンドを送ります。

セクタ消去が停止したあとは、対象 FLASH マクロからの読出し動作が許可されます。

このとき、セクタ消去一時停止中のセクタからは、ハードウェアシーケンスフラグが読み出されます。

セクタ消去一時停止状態に入ると、以下の状態になります。

- セクタ消去中にトグルする TOGG1 ビットが、セクタ消去一時停止状態ではトグルしない。
- フラッシュステータスレジスタの FRDY が"1"になる。

これらを利用して、セクタ消去一時停止状態に入った事を確認可能です。

＜注意事項＞
セクタ消去一時停止コマンドの発行からセクタ消去動作を停止して消去対象でないセクタからの読出しが可能となるまで最大 16.7µs 要します。

セクタ消去一時停止中も、ハードウェアシーケンスフラグの Bit2:TOGG2 はトグルしますので、本ビットを利用して停止中セクタの確認が可能です。

セクタ消去一時停止状態から、中断していた消去動作を再開するには、表 45-3 コマンドシーケンスのセクタ消去再開コマンドを送ります。

セクタ消去再開コマンドは、セクタ消去一時停止状態でのみ受け付けられます。

セクタ消去一時停止状態になったことを確認してからコマンドを送ってください。

消去再開コマンドが受け付けられると、セクタ消去状態に戻り消去動作を再開します。

45.5.9　セキュリティ機能

セキュリティ機能について示します。

本フラッシュメモリには、セキュリティ機能が搭載されています。セキュリティ機能が OFF のときは、制限なくフラッシュメモリを使用できますが、セキュリティ機能が ON のときは、外部バスからの命令フェッチ後の動作や、チップ消去以外の書込み・消去が抑止されます。制限内容については「45.5.9.4 セキュリティ ON 時のフラッシュアクセス制限」を参照してください。

45.5.9.1　リセット解除時におけるフラッシュセキュリティ ON/OFF 判別

リセット解除時におけるフラッシュセキュリティ ON/OFF 判別について示します。

本シリーズのフラッシュインタフェースは、リセット解除後にフラッシュセキュリティコード領域の 2 バイトを読み出します。その値が 0x0001 の場合は、セキュリティ ON になり、その後のフラッシュメモリへのアクセス制限が発生します。それ以外の値の場合は、セキュリティ OFF になります。

45.5.9.2　フラッシュセキュリティ設定方法

フラッシュセキュリティ設定方法について示します。

フラッシュセキュリティコード領域(図 45-2 セクタ構成図 (1024KB+64KB)などを参照してください)に 0x0001 が書き込まれた後にリセットの入力・解除が行われると、セキュリティ ON になります。一度セキュリティ ON になると、フラッシュメモリ領域全体を消去しない限りセキュリティ OFF にはなりません。

45.5.9.3　フラッシュセキュリティ解除方法

フラッシュセキュリティ解除方法について示します。

次の順序で、チップ消去コマンドをすべてのフラッシュマクロに対して行ってください。

1. ワークフラッシュを消去します。
2. フラッシュセキュリティコードが格納されていないプログラムフラッシュを消去します。
3. フラッシュセキュリティコードが格納されているプログラムフラッシュを消去します。

上記で示したように、プログラムフラッシュの消去は最後に行ってください。そうしない場合はプログラムフラッシュへの消去コマンドは無視されます。また、各消去の間にリセットが入った場合は 1. からやり直してください。

＜注意事項＞
ユーザモード(内部 FLASH 起動)では、任意のフラッシュマクロに対して消去コマンドの発行、フラッシュマクロ内のデータ消去が可能です。
フラッシュマクロに格納されるデータ保護の観点から、各フラッシュマクロに対するチップ消去の順序は上記のとおり実施することを推奨します。

45.5.9.4 セキュリティ ON 時のフラッシュアクセス制限

セキュリティ ON 時のフラッシュアクセス制限について示します。

セキュリティ ON 時には、起動モードにより以下で示す制限が発生します。

表 45-5 セキュリティ ON 時のアクセス制限

動作モード	アクセス制限
ユーザ・外バス	通常状態(後述のフラッシュセキュリティ違反によるアクセス制限がされていない状態)では、セキュリティ情報領域(FLASH メモリの先頭 9 ワード)に対する書込みはキャンセルされます。また、セクタ 0,1 に対するセクタイレーズコマンドも無視されます。 オンチップバス領域に対して命令フェッチが行なわれるとフラッシュセキュリティ違反リセット要因によるリセット要求が発行されます。以後、フラッシュメモリへのアクセスは受け付けません。 リセットで通常状態に復帰します。
上記以外 (ライタなど)	フラッシュメモリに対するアクセスを制限します。 読出しはデータをマスクし 0xFFFF_FFFF を返却します。書込みコマンド、セクタ消去コマンドは無視されます。 チップ消去コマンドは受け付けます。「45.5.9.3 フラッシュセキュリティ解除方法」を参照してください。

また、セキュリティ ON 時には、セキュリティ情報格納領域(フラッシュメモリの先頭 9 ワード)へのデータ読出し行った場合は

- データアクセスエラーとなり、不正命令例外またはデータアクセスエラー割込みが発生します。(詳細は「FR Family FR81 32 ビット・マイクロコントローラ プログラミングマニュアル」を参照してください。)
- 読出し値として 0xFFFFFFFF が返されます。

ただし、OCD ツール接続時の、OCDU からのアクセスまたはデバッグステート時の読出しの場合はこの制限はありません。

45.5.10 フラッシュメモリの使用上の注意

フラッシュメモリの使用上の注意について示します。

- 書込み中に本デバイスがリセットされた場合は，書き込んでいるデータは保証されません。
- FLASH 制御レジスタ(FCTLR)の FWE ビットで CPU プログラミングモードを設定(FWE=1)したきは、フラッシュメモリ上のプログラムを実行しないでください。正常な値を取り出せずにプログラムが暴走します。
- FLASH 制御レジスタ(FCTLR)の FWE ビットで CPU プログラミングモードを設定(FWE=1)し、フラッシュメモリ上に割込みベクタテーブルがある場合は、割込み要求を発生させないでください。正常な値を取り出せずにプログラムが暴走します。
- 本品種では ECC ビット付加のため、必ず 16 ビットを 2 回、32 ビットで書込みを行う必要があります。手順については「45.5.2. CPU によるフラッシュメモリ書込み」を参照してください。
- 複数マクロへの同時(並行) コマンド発行はしないでください。ハードウェアシーケンスフラグまたは FRDY ビットでコマンド完了を確認してから、次のマクロへのコマンドを投入してください。
- オンチップデバッガ(OCD)のパスワードによる認証が完了すると、セキュリティ ON 時の場合でも OCD を使用して外部からフラッシュメモリの内容を読み出すことができます。第三者による読出しを停止したい場合は、オンチップデバッガ(OCD)起動許可用のパスワードを必ず設定してください。
- FLASH プログラム/イレーズ中にスタンバイ状態に遷移することは禁止です。
- 本フラッシュメモリは、ECC を搭載しているため、既に何らかの値が書き込まれているアドレスへのデータ上書きは行えません。

46. ワークフラッシュメモリ

ワークフラッシュメモリについて説明します。

46.1 概要

ワークフラッシュメモリの概要について説明します。

本シリーズに内蔵されているフラッシュメモリの容量は 64K バイトです。ECC(Error Correction Code)が付加されています。

46.2 特長

ワークフラッシュメモリの特長について説明します。

● 使用可能容量：

■ CY91F575/CY91F577/CY91F578/CY91F579 ：64K バイト (8k バイト×8 セクタ)

本シリーズは ECC 符号格納のため、上記に加え 4 バイトにつき 6 ビットのフラッシュメモリが搭載されています。

● 高速動作：

■ ワード (32 ビット) 単位での読出しが 80MHz・2 サイクルで可能。

● 外部からの書込み：

■ ROM ライタにより可能

● 動作モード：

14. CPU-ROM モード
(CPU/DMA がフラッシュメモリにアクセス。読出しのみ)
データアクセスのみが可能です。インストラクションフェッチはできません。

15. CPU プログラミングモード
(CPU がフラッシュメモリにアクセス。読出し・書込み・消去)

16. フラッシュメモリモード
(外部からフラッシュメモリにアクセス可能)

● セキュリティ機能

■ 第三者によるフラッシュメモリ内容読出し阻止のため、セキュリティ ON 時に、外部からの命令フェッチ後の動作や、チップ消去以外の書込み・消去を抑止

■ オンチップデバッガ(OCD) 使用により、パスワード認証後にセキュリティ ON 時の場合でも OCD を使用した外部からの読出し可能。

● ECC (Error Correction Code) 機能

■ 1 ワード中の 1 ビットまでの誤りを訂正する ECC(Error Correction Code) セキュリティ機能があります。(2 ビット誤り検出機能は搭載していません。) 誤りは読出し中に自動で訂正されます。また、ECC 符号はフラッシュメモリへの書込み時に自動で付加されます。誤り訂正による読出しサイクルペナルティはありませんので、ソフトウェア開発の際に誤り訂正ペナルティを考慮する必要はありません。

■ チップイレーズ/セクタイレーズされた状態でデータを読み出すとイレーズ状態"FFFF"が読み出されます。ただしこのとき、ECC エラー訂正の発生を示すワークフラッシュステータスレジスタ (DFSTR) の DFECCERR ビットがセットされます。

*: 自動アルゴリズム=Embedded Algorithm

46.3 構成

ワークフラッシュメモリの構成について説明します。

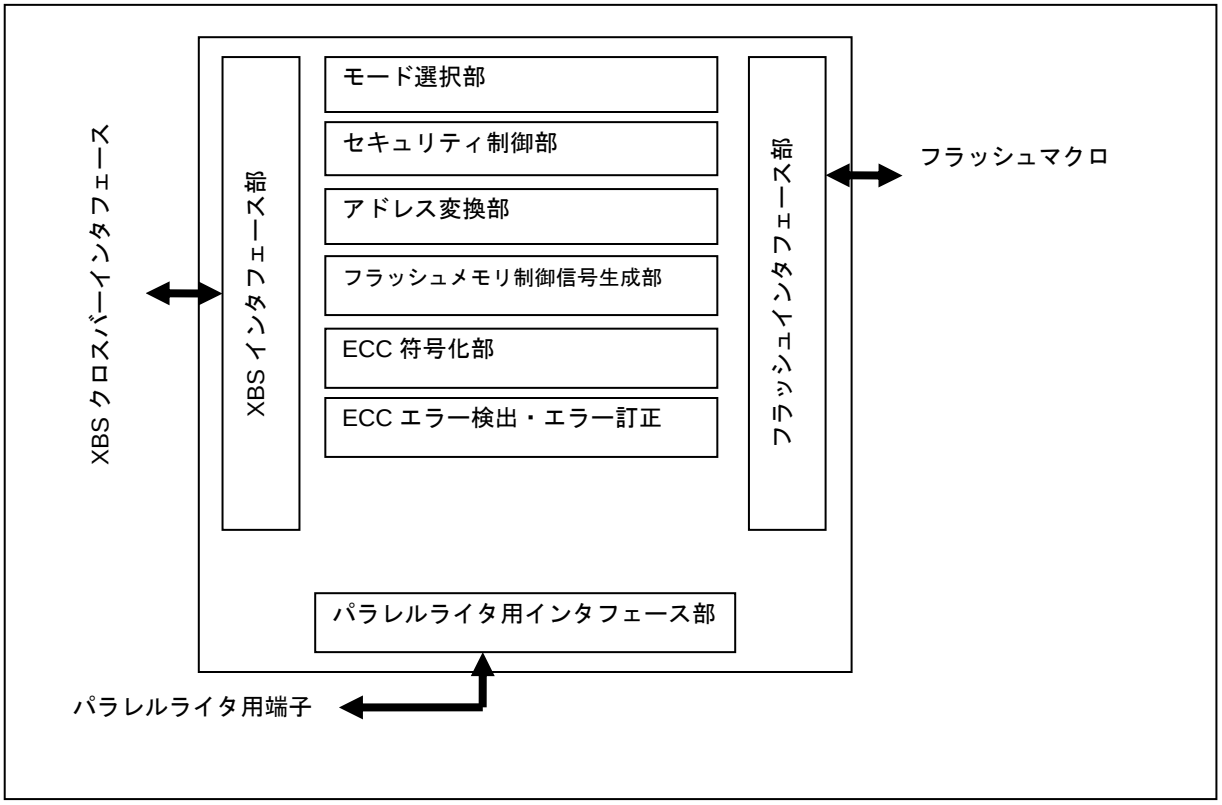
46.3.1. ブロックダイヤグラム

46.3.2. セクタ構成図

46.3.1 ブロックダイヤグラム

ワークフラッシュメモリのブロックダイヤグラムについて示します。

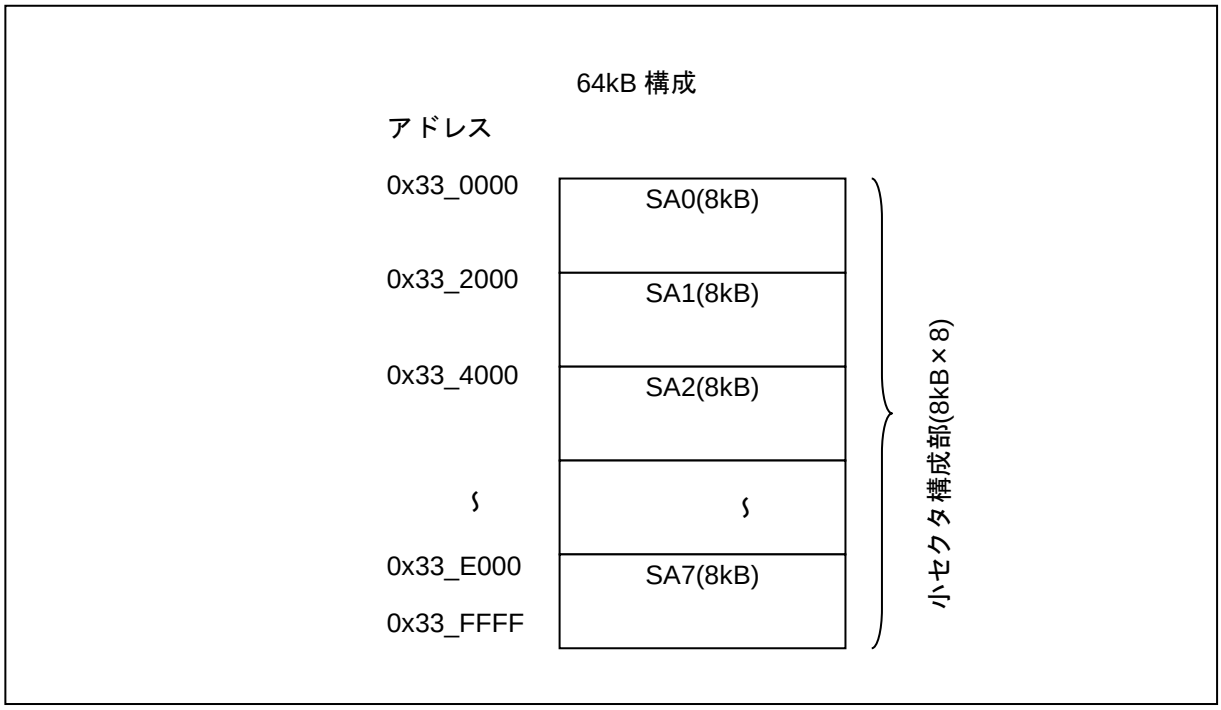
図 46-1 ブロックダイヤグラム (64KB 品種)



46.3.2 セクタ構成図

ワークフラッシュメモリのセクタ構成図について示します。

図 46-2 セクタ構成図



46.4 レジスタ

ワークフラッシュメモリのレジスタについて説明します。

表 46-1 レジスタマップ

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x2300	DFCTLR		予約	DFSTR	ワークフラッシュ制御レジスタ ワークフラッシュステータスレジスタ
0x2308	FLIFCTLR	予約	予約	予約	フラッシュインタフェース制御レジスタ

46.4.1 ワークフラッシュ制御レジスタ ：DFCTLR (WorkFlash ConTroL Register)

ワークフラッシュ制御レジスタのビット構成について示します。

ワークフラッシュへのアクセス制御を設定します。

■ DFCTLR：アドレス 2300H (アクセス: バイト,ハーフワード,ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約	FWE	予約					
初期値	-	0	-	-	-	-	-	-
属性	RX,WX	R/W	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							
初期値	-	-	-	-	-	-	-	-
属性	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX

[bit15] 予約

予約ビットです。読出し値は不定です。書込みは動作に影響しません。

[bit14] FWE (Flash Write Enable) : Flash 書込み許可

CPU モード時にワークフラッシュ への書込みを許可する制御ビットです。

本ビットを設定すると、ワークフラッシュメモリへのデータフェッチに対し、ECC によるエラー検出・データ訂正機能が無効になります。

FWE	説明
0	Flash 書込み不許可(初期値)
1	Flash 書込み許可

[bit13～bit0] 予約

予約ビットです。読出し値は不定です。書込みは動作に影響しません。

46.4.2 ワークフラッシュステータスレジスタ ：DFSTR (WorkFlash STatus Register)

ワークフラッシュステータスレジスタのビット構成について示します。

ワークフラッシュの状態を表示します。

■ DFSTR：アドレス 2303H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約					DFECCERR	DFHANG	DFRDY
初期値	-	-	-	-	-	0	0	1
属性	RX,WX	RX,WX	RX,WX	RX,WX	RX,WX	R/W	R,WX	R,WX

[bit7～bit3] 予約
予約ビットです。読出し値は不定です。書込みは動作に影響しません。

[bit2] DFECCERR (WorkFlash ECC Error coRRection) : データ読出し ECC 訂正発生
CPU モードにおけるワークフラッシュへのデータリード時に ECC エラーが発生したことを示します。このビットは"0"書込みでクリアされます。ECC エラーと"0"書込みが同時に発生した場合は"0"書込みが優先されます。

DFECCERR	読出し	書込み
0	データ読出し時の ECC による訂正は発生していません (初期値)	本ビットをクリアします
1	データ読出し時に ECC エラー訂正が発生しました	動作に影響しません

1 ワード中に 2 ビット以上の誤りがある場合は、本ビットの読出し値は不定です。

[bit1] DFHANG (WorkFlash HANG) : ワークフラッシュ HANG 状態
ワークフラッシュメモリの HANG 状態を示します。タイミング超過(「[bit5] : TLOV : (タイミングリミット超過フラグビット)」参照)すると HANG 状態になります。このビットが"1" になった場合はリセットコマンド(「46.5.3.1 コマンドシーケンス」参照)を発行してください。

自動アルゴリズムのコマンド発行直後は正しい値を読み出せない場合があるので、コマンド発行後 1 回目の本ビット読出し値は無視してください。

DFHANG	説明
0	通常状態
1	HANGUP 状態

[bit0] DFRDY (WorkFlash ReaDY) : **ワークフラッシュ書込み許可**

自動アルゴリズムでフラッシュメモリの書込み/ 消去動作が実行中か完了しているかを示します。動作中の場合、フラッシュメモリへデータを書き込んだりデータを消去したりすることはできません。

DFRDY	説明
0	動作中(書込み/ 消去不可、ステータス読出し可能)
1	動作完了(書込み/ 消去可能、読出し可能)

自動アルゴリズムのコマンド発行直後は正しい値を読み出せない場合があるので、コマンド発行後 1 回目の本ビット読出し値は無視してください。

46.4.3 フラッシュインタフェース制御レジスタ ：FLIFCTLR (Flash I/F Control Register)

フラッシュインタフェース制御レジスタのビット構成について示します。

Flash I/F を制御します。プログラムフラッシュ、ワークフラッシュ共用のレジスタです。

■ FLIFCTLR：アドレス 2308H (アクセス: バイト, ハーフワード, ワード)

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約			DFWDSBL	予約		ECCDSBL1	ECCDSBL0
初期値	-	-	-	0	-	0	0	0
属性	RX,WX	RX,WX	RX,WX	R/W	RX,WX	R/W0	R/W	R/W

[bit7～bit5] 予約
予約ビットです。読出し値は不定です。書込みは動作に影響しません。

[bit4] DFWDSBL (Data Fetch Wait cycle Disable) : データフェッチ・ウェイトサイクル無効
本ビットを"1"に設定すると、ウェイト設定時のデータフェッチ時に挿入されるウェイトサイクルを無効にします。ただし、サイクルタイム保証のためのウェイトサイクルは無効にすることはできません。

DFWDSBL	説明
0	ウェイトサイクル有効(初期値)
1	ウェイトサイクル無効

[bit3] 予約
予約ビットです。読出し値は不定です。書込みは動作に影響しません。

[bit2] 予約
予約ビットです。必ず"0"を書込んでください。

[bit1] ECCDSBL1(ECC Disable1) : ECC 機能無効 1
CPU モード時にワークフラッシュメモリへのライトアクセス、データフェッチに対し、ECC 機能の有効/無効を設定します。

ECCDSBL1	説明
0	ECC 機能有効(初期値)
1	ECC 機能無効

[bit0] ECCDSBL0(ECC Disable0) : ECC 機能無効 0
CPU モード時にプログラムフラッシュメモリへのライトアクセス、データフェッチに対し、ECC 機能の有効/無効を設定します。

ECCDSBL0	説明
0	ECC 機能有効(初期値)
1	ECC 機能無効

46.5 動作説明

ワークフラッシュメモリの動作について説明します。

Flash 領域へのアクセス方法について説明します。

46.5.1. アクセスモード設定

46.5.2. CPU によるフラッシュメモリ書込み

46.5.3. 自動アルゴリズム

46.5.4. リセットコマンド

46.5.5. 書込みコマンド

46.5.6. チップ消去コマンド

46.5.7. セクタ消去コマンド

46.5.8. セクタ消去一時停止コマンド

46.5.9. セキュリティ機能

46.5.10. フラッシュメモリの使用上の注意

46.5.1　アクセスモード設定

アクセスモード設定について示します。

本シリーズのフラッシュメモリには以下の 3 モードがあります。本項にて 1. 2.の設定方法を説明します。3.については、ご使用の ROM ライタの説明書を参照してください。

1. CPU-ROM モード
   (CPU がフラッシュメモリにアクセス。読出しのみ、バイト/ハーフワード/ワードアクセス)
2. CPU プログラミングモード
   (CPU がフラッシュメモリにアクセス。読出し/書込み、ハーフワードアクセスのみ)
3. フラッシュメモリモード (外部からフラッシュメモリにアクセス可能)

46.5.1.1　CPU-ROM モードへの設定

CPU-ROM モードへの設定について示します。

ワークフラッシュ制御レジスタ(DFCTLR)の FWE ビットが"0"のとき、CPU-ROM モードです。CPU-ROM モードでは、ワークフラッシュステータスレジスタ(DFSTR)の DFRDY ビットが"1"のとき、フラッシュメモリからの読み出しが可能になります。CPU-ROM モードでは、フラッシュメモリへの書込みはできませんです。リセット解除後はこの状態になります。

46.5.1.2　CPU プログラミングモードへの設定

CPU プログラミングモードへの設定について示します。

ワークフラッシュ制御レジスタ(DFCTLR)の FWE ビットが"1"のとき、CPU プログラミングモードです。CPU プログラミングモードでは、ワークフラッシュステータスレジスタ(DFSTR)の DFRDY ビットが"1"のとき、フラッシュメモリからの読出しおよび書込みが可能になります。

46.5.2　CPU によるフラッシュメモリ書込み

CPU によるフラッシュメモリ書込みについて示します。

CPU プログラミングモードに設定した後、自動アルゴリズムを利用して消去・書込みを行います。本品種では 1 ワードごとの ECC(Error Correction Code) が付加されますので、1 ワードごとの書込みを行う必要があります。以下の手順で 1 ハーフワードを 2 連続に、1 ワードごとに書込みを行ってください。この手順に従わない場合、ECC 計算されずにフラッシュメモリに書き込まれてしまうため、書き込んだ値を正しく読み出せません。

1. Flash アクセスサイズ設定を 16 ビットに設定します。(FCTLR:FSZ[1:0]=01)
   FCTLR は、『フラッシュメモリ』の章を参照してください。
2. 書込みコマンドを発行します。書込みアドレス= PA 書込みデータ= PD[31:16]
   書込みコマンドについては「46.5.5 書込みコマンド」を参照してください。
3. 書込みが終了するまで、ハードウェアシーケンスフラグを読み出します。
   ハードウェアシーケンスフラグ読出しについては「46.5.3.2 自動アルゴリズム実行状態」を参照してください。
4. 書込みコマンドを発行します。書込みアドレス= PA+2　書込みデータ= PD[15:0]
   この際、ハードウェアが自動で 2.の PD[31:16]とあわせて ECC 符号の計算を行い、ECC 符号の書込みも自動で同時に行われます。
5. 書込みが終了するまで、ハードウェアシーケンスフラグを読み出します。
6. 書込みデータがまだある場合は 2.に戻ります。すべて書込み完了した場合は 7.へ進みます。
7. CPU-ROM モードに設定します。
8. 書込みした値を読み出して、正しい値を読めるか確認してください。また、正しい値が読めた場合でも DFSTR:DFECCERR ビットを確認して ECC 訂正がされていないか確認してください。ECC 訂正が発生していた場合はフラッシュメモリ消去からやり直してください。

PA: 書込み対象アドレス(ワードアラインド)
PD[31:0]: 書込みデータ
PD[31:16]: 書込みデータ上位 16 ビット分
PD[15:0]: 書込みデータ下位 16 ビット分

46.5.3 自動アルゴリズム

自動アルゴリズムについて示します。

CPU プログラミングモードを利用する場合、フラッシュメモリへの書込み/ 消去は自動アルゴリズムを起動して行います。本項にて自動アルゴリズムについて説明します。

46.5.3.1 コマンドシーケンス

コマンドシーケンスについて説明します。

フラッシュメモリへ 1 回～ 6 回連続でハーフワード(16 ビット)のデータを書き込むと自動アルゴリズムが起動します。これをコマンドとよびます。コマンドシーケンスを以下に示します。

表 46-2 コマンドシーケンス

コマンド	書込み回数	1 回目		2 回目		3 回目		4 回目		5 回目		6 回目	
		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
リセット	1	任意	$F0_H$										
読出し	1	RA	RD										
書込み	4	$AA8_H$	AA_H	554_H	55_H	$AA8_H$	$A0_H$	PA	PD				
チップ消去	6	$AA8_H$	AA_H	554_H	55_H	$AA8_H$	80_H	$AA8_H$	AA_H	554_H	55_H	$AA8_H$	10_H
セクタ消去	6	$AA8_H$	AA_H	554_H	55_H	$AA8_H$	80_H	$AA8_H$	AA_H	554_H	55_H	SA	30_H
セクタ消去一時停止	1	任意	$B0_H$										
セクタ消去再開	1	任意	30_H										

* 表中のデータ表記は下位 8 ビット分のみを表記しています。上位 8 ビットは任意です。コマンドはハーフワードまたはバイトで書き込んでください。

* 表中のアドレス表記は下位 12 ビット分のみです。上位 20 ビット分は、対象となるフラッシュマクロのアドレス範囲のうちの任意のアドレスを指定してください。

PA：書込みアドレス (ハーフワードアラインド)

PD：書込みデータ (16 ビットで書き込んでください。)

SA：セクタアドレス (消去対象となるセクタのアドレス範囲の内の任意のアドレスを指定してください。)

RA：読出しアドレス

RD：読出しデータ (読出し幅は任意です。)

＜注意事項＞
コマンドアドレス、セクタ消去コマンド発行時に入力するセクタアドレス(SA)、の最下位 2 ビット分は以下のようにしてください。

- ハーフワードアクセス時: 2'b00
- バイトアクセス時: 2'b01 または 2'b11

例 1 バイトアクセス時、コマンドアドレス=(標準コマンドアドレスの最下位 2 ビットを 2'b01 に変更)とする場合

$AA8_H \rightarrow AA9_H$、$554_H \rightarrow 555_H$、SA → {SA[31:2], 2'b01}
(SA:セクタ消去コマンド発行時に入力する消去対象セクタ内の任意のアドレス)

例 2 バイトアクセス時、コマンドアドレス=(標準コマンドアドレスの最下位 2 ビットを 2'b11 に変更)とする場合

$AA8_H \rightarrow AAB_H$、$554_H \rightarrow 557_H$、SA → {SA[31:2], 2'b11}
(SA:セクタ消去コマンド発行時に入力する消去対象セクタ内の任意のアドレス)

＜注意事項＞
誤ったアドレス値やデータ値をライトした場合や、誤ったシーケンスでライトした場合、それまでライトしたコマンドはクリアされます。

■ リセットコマンド

リセットコマンドを対象 FLASH メモリに送るとそれまでに入力した表 46-2 コマンドシーケンスに示す各コマンド入力をキャンセルし、再び 1 回目からコマンド入力し直すことができます。

ただし、各コマンドを最後まで入力し自動アルゴリズムが起動すると、本リセットコマンドでは自動アルゴリズムを中止することはできません。

自動プログラムアルゴリズムの実行がタイミングリミットを超過した場合のみ、リセットコマンドを入力するとフラッシュメモリがリセット状態へ復帰します。

■ 読出しコマンド

読出しコマンドを対象セクタに送るとフラッシュメモリを読み出しできます。読出しコマンドを発行すると、フラッシュメモリは、ほかのコマンドが発行されるまで読出し状態を保ちます。

■ プログラム(書込み)コマンド

書込みコマンドを対象セクタに 4 回連続して送ると自動アルゴリズムを起動してフラッシュメモリにデータを書き込めます。データの書込みはどのようなアドレスの順番でもセクタの境界を越えても行えます。CPU プログラミングモードでは、ハーフワードで書込みを行います。4 回目の書込みが終了すると、自動アルゴリズムが起動し、フラッシュメモリへの自動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、外部からフラッシュメモリを制御する必要はありません。

実際の動作については、「46.5.5 書込みコマンド」を参照してください。

＜注意事項＞

- ハーフワードで書き込む場合、4 回目の書込みコマンド(書込みデータサイクル)を奇数番地に書き込むと、書込みが正しく行われません。必ず偶数番地に書き込んでください。
- 1 回の書込みコマンドシーケンスではハーフワードのデータ 1 つしか書き込めません。複数のデータを書き込みたい場合は、 1 データに 1 度書込みコマンドシーケンスを発行してください。
- セキュリティ ON 時においては、フラッシュの書込みには制限があります。詳細は「5.9.4 セキュリティ ON 時のフラッシュアクセス制限」を参照してください。

■ チップ消去コマンド

チップ消去コマンドを対象セクタに 6 回連続して送るとフラッシュメモリの全セクタを一括で消去できます。6 回目の書込みが終了すると、自動アルゴリズムが起動しチップ消去動作が開始されます。自動消去アルゴリズムが起動するとすべてのチップ消去する前に、フラッシュメモリがチップ内のすべてのセルに"0" を書き込んで、マージンを検証(プリプログラム)するため、チップ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は、外部からフラッシュメモリを制御する必要はありません。

実際の動作については、「46.5.6 チップ消去コマンド」を参照してください。

■ セクタ消去コマンド

セクタ消去コマンドを対象セクタに 6 回連続して送るとフラッシュメモリのセクタを消去できます。6 回目の書込みが終了し、40µs 経過(タイムアウト期間)すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。複数のセクタを消去したい場合は、40µs(タイムアウト期間)以内に消去するセクタのアドレスに消去コード(30_H)を書き込んでください。タイムアウト期間内に次のセクタが入力されないと、セクタ消去コマンドが無効になる場合があります。自動消去アルゴリズムが起動するとセクタ消去する前に、フラッシュメモリが消去するセクタのセルに"0" を書き込んで、マージンを検証(プリプログラム)するため、セクタ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は、外部からフラッシュメモリを制御する必要はありません。

実際の動作については、「46.5.7 セクタ消去コマンド」を参照してください。

＜注意事項＞

セキュリティ ON 時においては、フラッシュのセクタ消去には制限があります。詳細は「5.9.4 セキュリティ ON 時のフラッシュアクセス制限」を参照してください。

■ セクタ消去一時停止コマンド

セクタイレーズ実行中またはコマンドタイムアウト中に、セクタ消去一時停止コマンドを送ると、セクタ消去一時停止状態（セクタイレーズサスペンド状態）に移行できます。

セクタ消去一時停止状態では、消去対象でないセクタのメモリセルの読み出し動作が可能となります。ただし、新たな書き込みおよび消去コマンドは受け付けられません。

セクタ消去一時停止状態から、中断していた消去動作を再開するには、消去再開コマンドを送ります。

消去再開コマンドが受け付けられると、セクタ消去状態に戻り消去動作を再開します。

コマンドタイムアウト状態から本状態に遷移した場合であっても、イレーズレジュームコマンドが正常にライトされると、コマンドタイムアウト状態には遷移せず、セクタイレーズ状態に遷移して直ちにセクタ消去動作を再開します。

実際の動作については、「46.5.8 セクタ消去一時停止コマンド」を参照してください。

＜注意事項＞

セクタ消去一時停止コマンドの発行からセクタ消去動作を停止して消去対象でないセクタからの読み出しが可能となるまで最大 16.7µs 要します。

読み出し可能状態になったかどうかは、ワークフラッシュステータスレジスタ(DFSTR)の DFRDY ビットまたはハードウェアシーケンスフラグの TOGG1 で確認できます。

46.5.3.2 自動アルゴリズム実行状態

自動アルゴリズム実行状態について示します。

フラッシュメモリでは、書込みや消去を自動アルゴリズムで行うため、自動アルゴリズムが実行中かどうかを、ワークフラッシュステータスレジスタ(DFSTR)の DFRDY ビット で、動作状態をハードウェアシーケンスフラグで確認できます。

■ ハードウェアシーケンスフラグ

自動アルゴリズムの状態を示すフラグです。ワークフラッシュステータスレジスタ(DFSTR)の DFRDY ビットが "0" のときに、フラッシュメモリの任意のアドレスを読み出すと動作状態を確認できます。ハードウェアシーケンスフラグのビット構成を次に示します。

図 46-3 ハードウェアシーケンスフラグのビット構成

ハーフワードアクセスの場合

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
未定義	未定義	未定義	未定義	未定義	未定義	未定義	未定義

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DPOLL	TOGG1	TLOV	未定義	SETI	TOGG2	未定義	未定義

バイトアクセスの場合

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DPOLL	TOGG1	TLOV	未定義	SETI	TOGG2	未定義	未定義

＜注意事項＞

- ワードアクセスで読み出すことはできません。必ず、CPU プログラミングモードのときにハーフワードかバイトアクセスで読み出してください。
- CPU ROM モードで任意のアドレスを読み出しても、ハードウェアシーケンスフラグを読み出すことはできません。

● 各ビットとフラッシュメモリの状態

ハードウェアシーケンスフラグの各ビットの状態とフラッシュメモリ状態の対応を次に示します。

表 46-3 フラグとフラッシュメモリ状態の対応

状態		DPOLL	TOGG1	TLOV	SETI	TOGG2
実行中	書込み中	反転データ (*1)	トグル	0	0	-
	セクタ/チップ消去中	0	トグル	0	1	-
タイムリミット超過	書込みコマンド	反転データ (*1)	トグル	1	0	-
	セクタ/チップ消去コマンド	0	トグル	1	1	-
セクタ消去一時停止	消去対象セクタ	-	-	-	-	トグル

*1：読み出される値については、「● ビット説明」を参照してください。

● ビット説明

[bit15～bit8] 未定義ビット

[bit7] DPOLL：(データポーリングフラグビット)

書き込み/消去対象アドレスを指定してハードウェアシーケンスフラグを読み出すと、データポーリング機能で自動アルゴリズムが実行中かどうかをこのビットに示します。
読み出される値は動作状態によって異なります。

1. 書込み時

書込み中	最後に書き込まれたデータの bit7 の値と逆の値(反転データ) が読み出されます。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。
書込み終了後	ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit7 の値が読み出されます。

2. セクタ消去時

セクタ消去実行中	消去中のセクタから"0" が読み出されます。
セクタ消去後	必ず"1" が読み出されます。

3. チップ消去時

チップ消去実行中	必ず"0" が読み出されます。
チップ消去後	必ず"1" が読み出されます。

4. セクタ消去一時停止時

一時停止状態(未完了)	セクタ消去一時停止セクタから、"0"が読み出されます。
セクタ消去動作完了	セクタ消去一時停止セクタから、"1"が読み出されます。

＜注意事項＞

自動アルゴリズムを起動しているときは、指定したアドレスのデータを読み出すことはできません。このビットで自動アルゴリズムの動作が終了していることを確認してから、データを読み出してください。

[bit6] TOGG1：(トグルフラグ 1 ビット)

任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムが実行中かどうかをこのビットで示します。読み出される値は動作状態によって異なります。

書込み/ セクタ消去/ チップ消去時

書込み/ セクタ消去/ チップ消去中：	連続でこのビットを読み出すと、"1" と"0" が交互に読み出されます(トグル動作)。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。
書込み/ セクタ消去/ チップ消去終了後：	ハードウェアシーケンスフラグを読み出すために指定したアドレスの bit6 の値が読み出されます。

[bit5] TLOV：(タイミングリミット超過フラグビット)
任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、自動アルゴリズムの実行時間がフラッシュメモリ内部で規定している時間(内部パルスの回数)を超過したかどうかをこのビットで示します。読み出される値は動作状態によって異なります。

書込み/ セクタ消去/ チップ消去時

次の値が読み出されます。

"0"	規定時間内
"1"	規定時間を超えている

このビットが"1" のときに、 DPOLL ビットや TOGG1 ビットが自動アルゴリズム実行中であることを示していると、書込みや消去に失敗したことになります。

例えば、フラッシュメモリでは"0" が書き込まれているデータを"1" に書き換えることができないため、"0" が書き込まれているアドレスに"1" を書き込もうとすると、フラッシュメモリがロックされ自動アルゴリズムが終了しません。この場合は、DPOLL ビットの値は無効のままになり、TOGG1 ビットからは"1" と"0" が交互に読み出され続けます。この状態のまま規定時間を超えたときに、このビットが"1" に変わります。このビットが"1" になった場合はリセットコマンドを発行してください。

＜注意事項＞
このビットが"1" の場合は、フラッシュメモリが正しく使用されなかったことを示しています。フラッシュメモリの不良ではありません。リセットコマンドを発行してから適切な処理を行ってください。

[bit4] 未定義ビット

[bit3] SETI (セクタ消去タイマフラグビット)
セクタ消去時は、セクタ消去コマンドを発行してから実際にセクタ消去が開始されるまでには、40µs のタイムアウト期間が必要です。任意のアドレスを指定してハードウェアシーケンスフラグを読み出すと、セクタ消去コマンドのタイムアウト期間中かどうかをこのビットで示します。読み出される値は動作状態によって異なります。

セクタ消去時：
セクタ消去時に次のセクタ消去コードを入力する前に、このビットを確認することで次のセクタ消去コードが受け付けられる状態かどうかを確認できます。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスせず、 次の値が読み出されます。

"0"	セクタ消去ウェイト期間中(次のセクタ消去コード(0x30) を受け付けられます。)
"1"	セクタ消去ウェイト期間を超過している(このときに、DPOLL ビットや TOGG1 ビットが自動アルゴリズム実行中であることを示していると、フラッシュメモリ内部の消去が開始されています。この場合、セクタ消去コード(0x30) 以外のコマンドはフラッシュメモリ内部の消去が完了するまで無視されます。)

[bit2] TOGG2：(トグルフラグ 2 ビット)
セクタイレーズサスペンド状態において、消去対象でないセクタに対しては読出し(リード)を行うことができますが、消去対象セクタに対してはリードができません。本フラグは、セクタ消去一時停止中でリードアドレスが消去対象セクタの場合、出力データがトグルして、消去対象セクタであることを示します。

消去対象セクタへの読出し	連続でこのビットを読み出すと、"1" と"0" が交互に読み出されます(トグル動作)。ハードウェアシーケンスフラグを読み出すために指定したアドレスにはアクセスされません。
消去対象でないセクタへの読出し	指定したアドレスのデータを読み出します。

[bit1, bit0] 未定義ビット

46.5.4　リセットコマンド

リセットコマンドについて示します。

リセットコマンドを対象 FLASH メモリに送るとフラッシュメモリをリセット状態にできます。この状態は、フラッシュメモリの初期状態のため、電源を投入したときやコマンドが正常終了すると、フラッシュメモリは常にリセット状態に戻ります。電源投入時はデータ読出しコマンドを発行する必要はありません。また、リセット状態では通常の読出しアクセスでデータを読み出したり、CPU からプログラムアクセスしたりできるため、データを読み出すときにセットコマンドを発行する必要はありません。

46.5.5　書込みコマンド

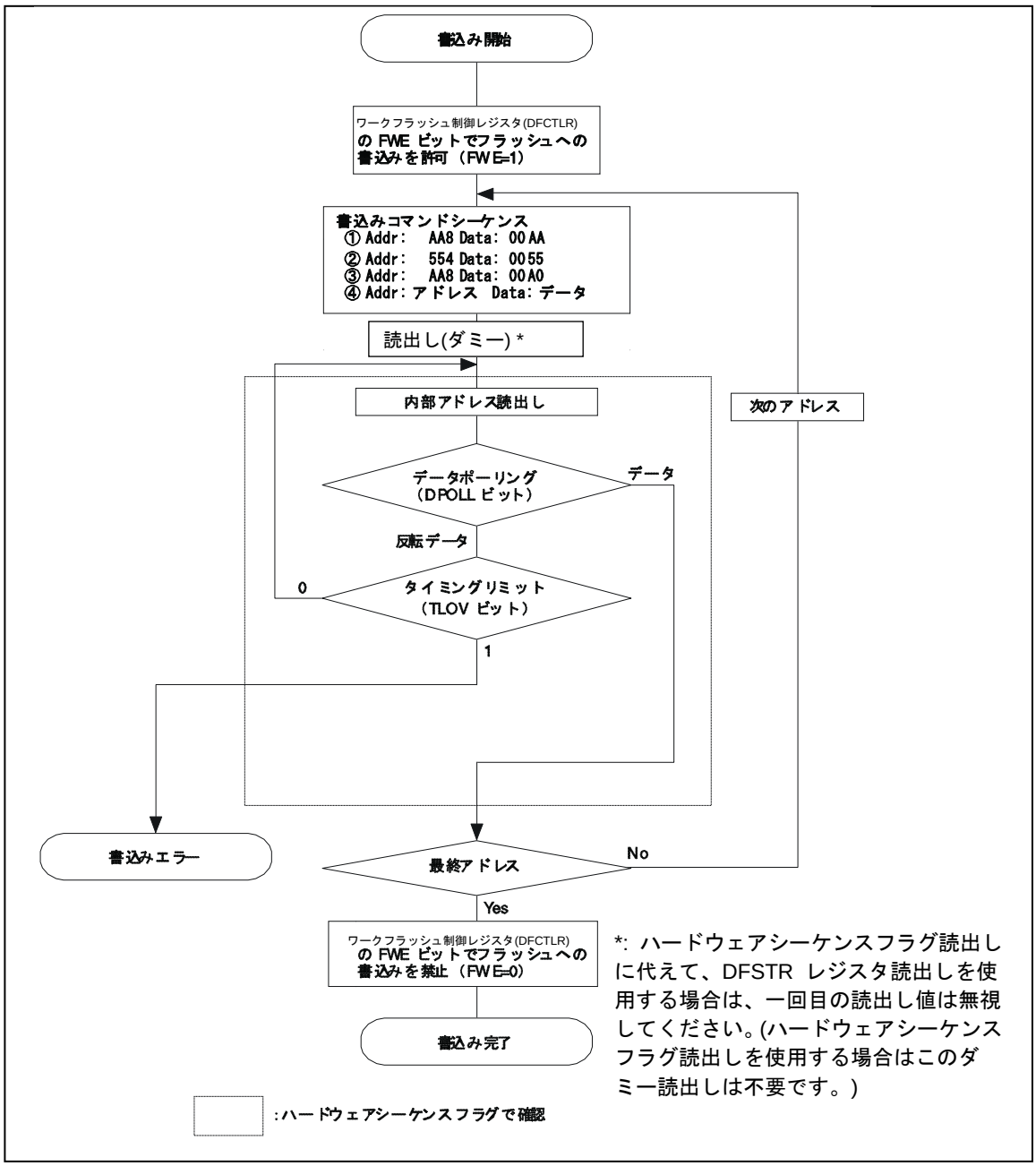
書込みコマンドについて示します。

次の順番で書込みを行います。

1. 書込みコマンドを対象セクタに連続して送る
   自動アルゴリズムが起動されフラッシュメモリへデータが書き込まれます。書込みコマンド発行後は外部からフラッシュメモリを制御する必要はありません。
2. 書込みを行ったアドレスにリードアクセスする
   読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOLL ビット)が書き込んだ値と一致していると、フラッシュメモリへの書込みが終了したことになります。書込みが終了していない場合は、最後に書き込んだデータの bit7 の値と逆の値(反転データ)が読み出されます。

フラッシュメモリへの書込み動作例を次に示します。

図 46-4 書込み手順例



＜注意事項＞

- 書込みが終了すると、フラッシュメモリは読出しモードに戻るため、書込みアドレスを受け付けなくなります。
- 書込みコマンドについては、「46.5.3 自動アルゴリズム」を参照してください。
- ハードウェアシーケンスフラグの DPOLL ビットは、TLOV ビットと同時に値が変わるので TLOV ビットが"1" の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG1 ビット、TLOV ビットが"1" に変わると同時にトグル動作を停止します。そのため、TLOV ビットが"1"の場合でも TOGG1 ビットを再度確認する必要があります。

- フラッシュメモリへは、どのようなアドレスの順番でも、またセクタの境界を越えても書き込めますが、 1 回の書込みコマンドシーケンスではハーフワードのデータ 1 つしか書き込めません。複数のデータを書き込みたい場合は、 1 データに 1 度書込みコマンドシーケンスを発行してください。
- 一度、 "0" が書き込まれたデータを"1" に戻すことはできません。"0" を"1" に書き換えると、以下のいずれかになります。
  - □ 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
  - □ 書込み規定時間を超え、ハードウェアシーケンスフラグビットの TLOV ビットが"1" に変わる
  - □ "1" が書き込まれたように見える

  ただし、"1" が書き込まれたように見えた場合でも、実際のデータは"0"のままのため読出し/ リセットモードでデータを読み出すと"0" が読み出されます。データを"1" に戻したい場合は、チップ消去かセクタ消去を行ってください。

- 書込み動作中はフラッシュメモリに書き込まれたすべてのコマンドが無視されます。
- 書込み中に本デバイスがリセットされた場合は、書き込んでいるデータは保証されません。
- 本シリーズでは ECC ビット付加のため、必ず 16 ビットを 2 回、32 ビットで書込みを行う必要があります。手順については「46.5.2 CPU によるフラッシュメモリ書込み」を参照してください。

46.5.6　チップ消去コマンド

チップ消去コマンドについて示します。

チップ消去コマンドにて、フラッシュメモリの消去対象フラッシュマクロを一括して消去できます。

チップ消去コマンドを対象 FLASH メモリに連続して送ると自動アルゴリズムを起動して、全セクタを一括で消去できます。チップ消去コマンドについては、「46.5.3 自動アルゴリズム」を参照してください。

1. チップ消去コマンドを消去対象フラッシュマクロのセクタに連続して送る
   自動アルゴリズムが起動されフラッシュメモリへデータが書き込まれます。

2. 消去対象フラッシュマクロの任意のアドレスにリードアクセスする
   読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOLL ビット)が"1" だと、チップ消去が終了したことになります。

   チップ消去に必要な時間は「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。チップ消去動作が終了すると、フラッシュメモリは読出し/リセットモードに戻ります。

＜注意事項＞

- 自動消去アルゴリズムが起動するとすべてのチップ消去する前に、フラッシュメモリがチップ内のすべてのセルに"0" を書き込んで、マージンを検証(プリプログラム)するため、チップ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は外部からフラッシュメモリを制御する必要もありません。
- セキュリティ ON 時においては、フラッシュの消去には手順に制限があります。詳細は「45.5.9.3 フラッシュセキュリティ解除方法」を参照してください。

46.5.7　セクタ消去コマンド

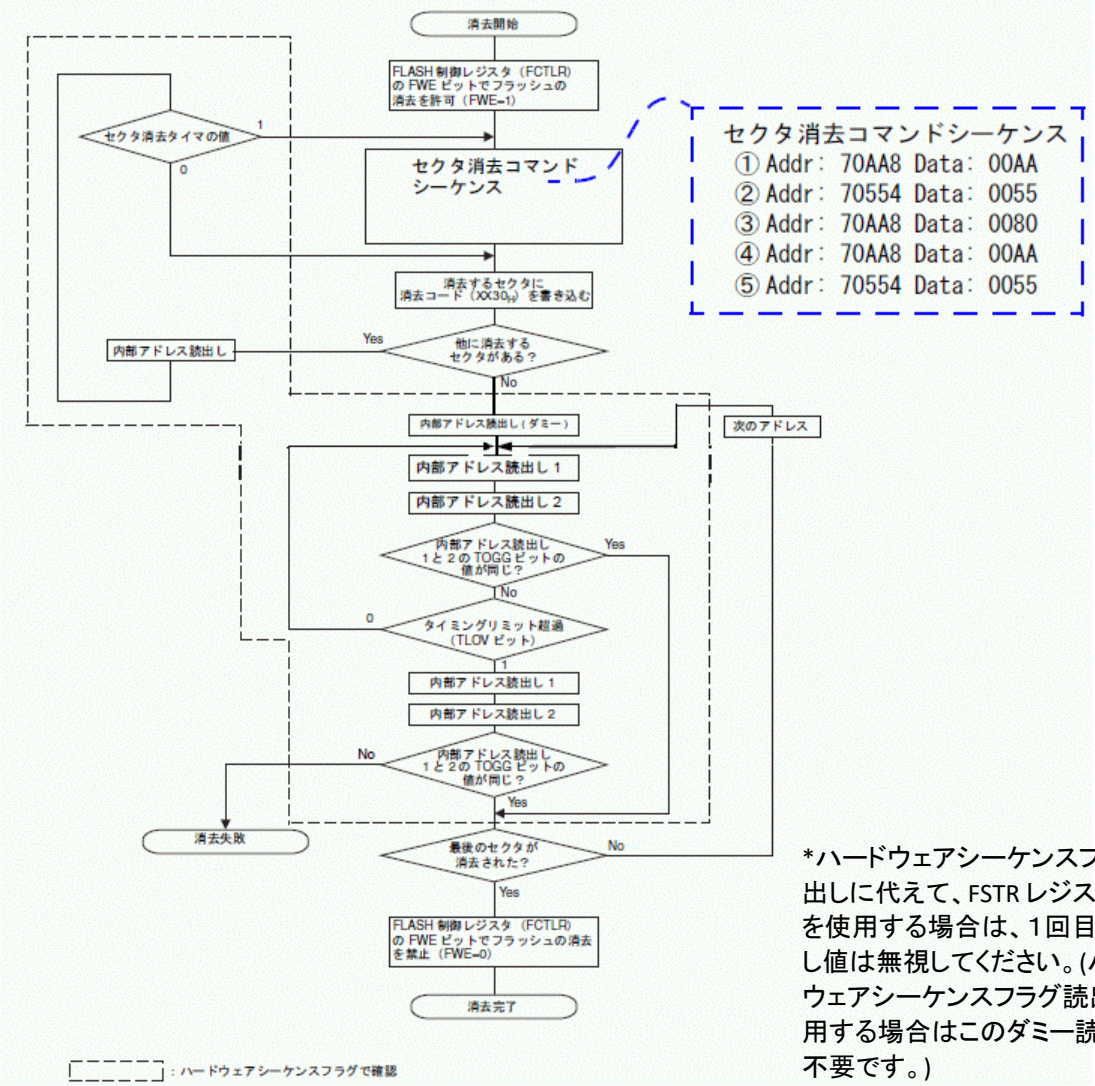
セクタ消去コマンドについて示します。

フラッシュメモリ内のセクタを選択して、選択したセクタのデータのみを消去できます。複数のセクタを同時に指定することもできます。次の順番でセクタ消去を行います。

1. セクタ消去コマンドを対象セクタに連続して送る
   40µs 経過(タイムアウト期間)すると、自動アルゴリズムが起動しセクタ消去動作が開始されます。複数のセクタを消去したい場合は、40µs (タイムアウト期間)以内に消去するセクタのアドレスに消去コード(30_H)を書き込んでください。タイムアウト期間経過後に書き込んでも、セクタ消去コマンドが無効になる場合があります。

2. 任意のアドレスにリードアクセスする
   読み出したデータはハードウェアシーケンスフラグになります。そのため、読み出したデータの bit7 (DPOLL ビット)が"1" だと、セクタ消去が終了したことになります。また、TOGG1 ビットを利用してセクタ消去が完了したかどうかを確認することもできます。
   確認動作に TOGG1 ビットを使用した場合を例にとって、セクタ消去手順例を次に示します。

図 46-5 セクタ消去手順例



*ハードウェアシーケンスフラグ読出しに代えて、FSTR レジスタ読出しを使用する場合は、1回目の読出し値は無視してください。(ハードウェアシーケンスフラグ読出しを使用する場合はこのダミー読出しは不要です。)

＜注意事項＞

- セクタ消去に必要な時間は「(セクタ消去時間＋セクタ書込み時間(プリプログラム))×セクタ数」となります。
- セクタ消去動作が終了すると、フラッシュメモリは読出し/ リセットモードに戻ります。
- セクタ消去コマンドについては、「46.5.3 自動アルゴリズム」を参照してください。
- ハードウェアシーケンスフラグの DPOLL ビットは、 TLOV ビットと同時に値が変わるので TLOV ビットが"1" の場合でも再度確認する必要があります。
- ハードウェアシーケンスフラグの TOGG1 ビットは、 TLOV ビットが"1" に変わると同時にトグル動作を停止します。そのため、 TLOV ビットが"1" の場合でも TOGG1 ビットを再度確認する必要があります。
- タイムアウト期間を含むセクタ消去中に、セクタ消去コマンド以外のコマンドを発行すると、フラッシュメモリが読出し / リセット状態になります。この場合、フラッシュメモリがリセットされるので、コマンドが発行される 1 つ前または複数のセクタ消去コマンドが無効になります。セクタ消去を行う場合は、最初からセクタ消去コマンドを発行しなおしてください。
- 自動消去アルゴリズムが起動すると、セクタ消去する前に、フラッシュメモリが消去するセルに"0" を書き込んで、マージンを検証(プリプログラム)するため、セクタ消去前に、フラッシュメモリに書込みを行う必要はありません。また、マージン検証中は外部からフラッシュメモリを制御する必要もありません。

46.5.8　セクタ消去一時停止コマンド

セクタ消去一時停止コマンドについて示します。

セクタイレーズ実行中またはコマンドタイムアウト中に、セクタ消去を一時的に停止させることができます。

セクタ消去一時停止状態では、消去対象でないセクタのメモリセルの読み出し動作が可能となります。ただし、新たな書き込みおよび消去コマンドは受け付けられません。

セクタ消去を一時停止させるには、対象 FLASH マクロの任意のアドレスにセクタ消去一時停止コマンドを送ります。

セクタ消去が停止したあとは、対象 FLASH マクロからの読み出し動作が許可されます。

このとき、セクタ消去一時停止中のセクタからは、ハードウェアシーケンスフラグが読み出されます。

セクタ消去一時停止状態に入ると、以下の状態になります。

- セクタ消去中にトグルする TOGG1 ビットが、セクタ消去一時停止状態ではトグルしない。
- ワークフラッシュステータスレジスタの DFRDY が"1"になる。

これらを利用して、セクタ消去一時停止状態に入った事を確認可能です。

＜注意事項＞
セクタ消去一時停止コマンドの発行からセクタ消去動作を停止して消去対象でないセクタからの読み出しが可能となるまで最大 16.7 μ s 要します。

セクタ消去一時停止中も、ハードウェアシーケンスフラグの Bit2:TOGG2 はトグルしますので、本ビットを利用して停止中セクタの確認が可能です。

セクタ消去一時停止状態から、中断していた消去動作を再開するには、表 46-2 コマンドシーケンスのセクタ消去再開コマンドを送ります。

セクタ消去再開コマンドは、セクタ消去一時停止状態でのみ受け付けられます。

セクタ消去一時停止状態になったことを確認してからコマンドを送ってください。

消去再開コマンドが受け付けられると、セクタ消去状態に戻り消去動作を再開します。

46.5.9　セキュリティ機能

セキュリティ機能について示します。

本フラッシュメモリには、セキュリティ機能が搭載されています。セキュリティ機能が OFF のときは、制限なくフラッシュメモリを使用できますが、セキュリティ機能が ON のときは、外部バスからの命令フェッチ後の動作や、チップ消去以外の書込み・消去が抑止されます。制限内容については『46.5.9.4 セキュリティ ON 時のフラッシュアクセス制限』を参照してください。

46.5.9.1　リセット解除時におけるフラッシュセキュリティ ON/OFF 判別

リセット解除時におけるフラッシュセキュリティ ON/OFF 判別について示します。

本品種のフラッシュインタフェースは、リセット解除後にフラッシュセキュリティコード領域の 2 バイトを読出します。その値が 0x0001 の場合は、セキュリティ ON になり、その後のフラッシュメモリへのアクセス制限が発生します。それ以外の値の場合は、セキュリティ OFF になります。

46.5.9.2　フラッシュセキュリティ設定方法

フラッシュセキュリティ設定方法について示します。

フラッシュセキュリティコード領域(『フラッシュメモリ』の章の『図 46-2 セクタ構成図』を参照してください)に 0x0001 が書き込まれた後にリセットの入力・解除が行われると、セキュリティ ON になります。一度セキュリティ ON になると、フラッシュメモリ領域全体を消去しない限り、セキュリティ OFF にはなりません。

46.5.9.3　フラッシュセキュリティ解除方法

フラッシュセキュリティ解除方法について示します。

次の順序で、チップ消去コマンドをすべてのフラッシュマクロに対して行ってください。

16. ワークフラッシュを消去します。
17. フラッシュセキュリティコードが格納されていないプログラムフラッシュを消去します。
18. フラッシュセキュリティコードが格納されているプログラムフラッシュを消去します。

上記で示したように、プログラムフラッシュの消去は最後に行ってください。そうしない場合はプログラムフラッシュへの消去コマンドは無視されます。また、各消去の間にリセットが入った場合は 1.からやり直してください。

＜注意事項＞
ユーザモード(内部 FLASH 起動)では、任意のフラッシュマクロに対して消去コマンドの発行、フラッシュマクロ内のデータ消去が可能です。
フラッシュマクロに格納されるデータ保護の観点から、各フラッシュマクロに対するチップ消去の順序は上記のとおり実施することを推奨します。

46.5.9.4 セキュリティ ON 時のフラッシュアクセス制限

セキュリティ ON 時のフラッシュアクセス制限について示します。

セキュリティ ON 時には、起動モードにより以下で示す制限が発生します。

表 46-4 セキュリティ ON 時のアクセス制限

動作モード	アクセス制限
ユーザ・外バス	通常状態(後述のフラッシュセキュリティ違反によるアクセス制限がされていない状態)では FLASH メモリに対するアクセス制限はありません。 オンチップバス領域に対して命令フェッチが行なわれるとフラッシュセキュリティ違反リセット要因によるリセット要求が発行されます。以後、フラッシュメモリへのアクセスは受け付けません。 リセットで通常状態に復帰します。
上記以外 (ライタなど)	フラッシュメモリに対するアクセスを制限します。 読出しはデータをマスクし 0xFFFF_FFFF を返却します。書込みコマンド、セクタ消去コマンドは無視されます。 チップ消去コマンドは受け付けます。「45.5.9.3 フラッシュセキュリティ解除方法」を参照してください。

46.5.10 フラッシュメモリの使用上の注意

フラッシュメモリの使用上の注意について示します。

- 書込み中に本デバイスがリセットされた場合は、書き込んでいるデータは保証されません。
- ワークフラッシュ制御レジスタ(DFCTLR)の FWE ビットで CPU プログラミングモードを設定(FWE=1)したきは、フラッシュメモリ上のプログラムを実行しないでください。正常な値を取り出せずにプログラムが暴走します。
- ワークフラッシュ制御レジスタ(DFCTLR)の FWE ビットで CPU プログラミングモードを設定(FWE=1)し、フラッシュメモリ上に割込みベクタテーブルがある場合は、割込み要求を発生させないでください。正常な値を取り出せずにプログラムが暴走します。
- 本品種では ECC ビット付加のため、必ず 16 ビットを 2 回、32 ビットで書込みを行う必要があります。手順については「46.5.2 CPU によるフラッシュメモリ書込み」を参照してください。
- 複数マクロへの同時(並行) コマンド発行はしないでください。ハードウェアシーケンスフラグまたは DFRDY ビットでコマンド完了を確認してから、次のマクロへのコマンドを投入してください。
- オンチップデバッガ(OCD) のパスワードによる認証が完了すると、セキュリティ ON 時の場合でも OCD を使用して外部からフラッシュメモリの内容を読み出すことができます。第三者による読出しを止したい場合は、オンチップデバッガ(OCD)起動許可用のパスワードを必ず設定してください。
- フラッシュプログラム/イレーズ中にスタンバイ状態に遷移することは禁止です。
- 本フラッシュメモリは、ECC を搭載しているため、既に何らかの値が書き込まれているアドレスへのデータ上書きは行えません。

47. HS_SPI

本シリーズの HS_SPI (High-Speed Serial Parallel Interface) は使用できません。

47.1 概要

HS_SPI の概要について説明します。

HS_SPI モジュールは様々なオペレーティング・モードをサポートすることにより、デファクトスタンダードである SPI プロトコルを使用するシリアル周辺デバイス対応します。また、新しい仕様であるデュアルビットおよびクワッドビット SPI プロトコルにも対応します。次節で HS_SPI の特長について説明します。

(注意事項)

本シリーズでは HS_SPI は使用禁止です。

47.2 特長

HS_SPI の特長について説明します。

- デュアルビット/クワッドビットモードの SPI 動作と同様にレガシーモードをサポート
- マスタモード時、最大 4 個のスレーブデバイスをサポート
- CY91570 シリーズにおける各モードのシリアルクロック動作周波数範囲
  - コマンドシーケンサモード (最大 16MHz)
  - ダイレクトモード/マスタ動作 (最大 16MHz)
  - ダイレクトモード/スレーブ動作 (最大 8MHz)
- スレーブ選択ごとにシリアルクロックの転送速度，スレーブ選択信号のアクティブレベル，極性，および位相の設定が可能
- 「コマンドシーケンサ」モード時、外部シリアルフラッシュとシリアル SRAM のデバイスは本製品のアドレス空間にメモリ配置することができます。
- 「コマンドシーケンサ」モードでは、オンチップバスマスタ(CPU コア、DMAC)によるメモリアクセスは HS_SPI によって自動的にシリアルメモリのリード/ライトコマンドで変換されます。
- 「ダイレクト」モードでは、FIFO インタフェースを介して HS_SPI を標準 SPI として使用します。
- ダイレクトモード動作にて、2 つの CY91570 シリーズデバイスをインタフェースできます。

47.3 構成

HS_SPI の構成について説明します。

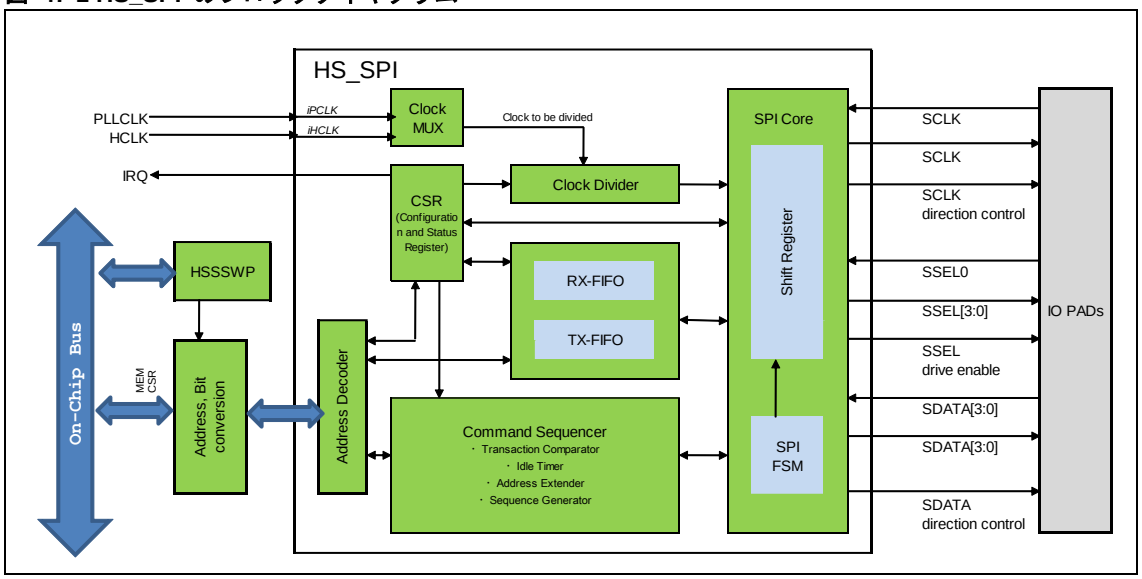
47.3.1 ブロックダイヤグラム

47.3.2 用語/略語

47.3.1 ブロックダイヤグラム

HS_SPI モジュールのブロックダイヤグラムを示します。

図 47-1 HS_SPI のブロックダイヤグラム



● HS_SPI レジスタ設定

HS_SPI の動作は Configuration and Status Register(CSR)の設定によって制御と監視ができます。 CSR の詳細は、「47.4. レジスタ」を参照してください。

● SPI コアとクロック分周器

SPI コアはプロトコルエンジンであり、シリアルインタフェースの駆動とサンプリングを行います。SPI 通信に関連する属性(シリアルクロック周波数, シリアルクロック位相, 極性など)は CSR で設定されます。ホスト側では、HS_SPI がダイレクトモードで動作しているか、コマンドシーケンサモードで動作しているかにより、SPI コアは FIFO, またはコマンドシーケンサに内部接続します。SPI マスタ動作時、HS_SPI は 4 つの SPI スレーブに対してシリアル転送を開始でき、これらの SPI スレーブは 4 つのスレーブ選択信号 (SSEL0, SSEL1, SSEL2,SSEL3)に接続されます。SPI スレーブ動作時、HS_SPI は外部の SPI マスタによって開始されたシリアル転送に対応できます。このとき、SSEL0 端子はアサートされます。内部クロック分周器は、HS_SPI が SPI マスタとして動作するとき、シリアルクロック出力(SCLK)を生成します。「クロックマルチプレックス」はクロック選択器であり、オンチップバスクロック(iHCLK)とペリフェラルクロック(iPCLK)のどちらか一方を CSR の設定によりクロック分周器への入力とします。

● オンチップバスインタフェースとアドレスデコード

CY91570 シリーズのオンチップバスマスタは、オンチップバスを介して HS_SPI モジュールにアクセスできます。アドレスデコーダはオンチップバスのアドレス信号をデコードします。レジスタへのアクセスの場合、アドレスは、そのまま CSR に送られます。 コマンドシーケンサモードの場合、オンチップバスアクセスが、4 つのスレーブ選択信号の 1 つに接続されているシリアルメモリデバイスへのアクセスであれば、メモリアドレスはコマンドシーケンサに渡されます。

● TX-FIFO と RX-FIFO

HS_SPI は内部に 2 つの FIFO メモリを備え、ここにデータを一時的に格納します。1 つが送信データ用で、ほかの 1 つが受信データ用となります。各 FIFO は 16 段でデータ幅は 32 ビットです。これらの FIFO は、HS_SPI のダイレクトモードで使用されます。

● コマンドシーケンサ

コマンドシーケンサは、外部シリアルメモリデバイスを本製品のアドレス空間に割り付けます。このブロックは、「トランザクション比較器(Transaction Comparator)」，「アドレス拡張器(Address Extender)」，「アイドルタイマ(Idle Timer)」および「シーケンス発生器(Sequence Generator)」から成ります。メモリ配置されたシリアルメモリにアクセスするときは、シーケンス発生器が外部メモリデバイスへのアクセスを開始します。シリアルメモリデバイスのアドレスは「アドレス拡張器」により生成されます。アドレス拡張器はオンチップバスのアドレス信号のいくつかのビットを HS_SPI アドレス拡張レジスタと組み合わせます。シリアルインタフェース上のメモリのリード・ライト動作の一つが完了すると、シーケンス発生器はオンチップバスマスタにそのことを伝え、シリアルクロックを切断し、対応するスレーブ選択信号出力のアサートをアイドルタイマで定められた期間保持します。アイドルタイムアウトの期間に、トランザクション比較器がそれまでと同じアドレスと、アクセスタイプ(リード・ライト)の外部シリアルメモリへのアクセス指示を受け取ると、シーケンス発生器はシリアルクロックの生成を開始し、次のシリアルメモリの領域にアクセスを開始します。

47.3.2 用語/略語

HS_SPI の用語/略語について示します。

用　　語	意　　味
バイト時間	バイト時間とは、SPI インタフェースにおいて 1 バイトのデータを転送するのに要する時間を表します。1 バイト時間は、クワッドビット SPI プロトコルでは SCLK の 2 周期、デュアルビット SPI プロトコルでは SCLK の 4 周期、レガシーSPI プロトコルでは SCLK の 8 周期になります。
CSR	HS_SPI の制御/ステータスレジスタ
MEM	HS_SPI のメモリ領域
DMA	ダイレクトメモリアクセス
FIFO	ファーストイン, ファーストアウト
Half Word	16 ビットのデータ
Word	32 ビットのデータ
iHCLK	HS_SPI のバスインタフェースのクロック (CY91570 シリーズのオンチップバスクロック)
iPCLK	シリアルクロック発生の別クロックソース CY91570 シリーズの HS_SPI では、PLLCLK を意味します。『クロック』の章を参照してください。
SPI	シリアル・ペリフェラル・インタフェース
TX	送信 (Transmit)
RX	受信 (Receive)

47.4 レジスタ

レジスタについて説明します。

■ ベースアドレスと外部端子

チャネル#	ベース_アドレス	外部端子								
		SCLK	SSEL0	SSEL1	SSEL2	SSEL3	SDATA0 (*1)	SDATA1 (*1)	SDATA2	SDATA3
0	0x2000_0000	SPI_CLK	SPI_CS0	SPI_CS1	SPI_CS2	SPI_CS3	SPI_SIO0 / SPI_SIO1	SPI_SIO1 / SPI_SIO0	SPI_SIO2	SPI_SIO3

*1 : EPFR52 レジスタで設定されます。詳細は、『I/O ポート』の章を参照してください。

■ HS_SPI アドレススワップ制御レジスタ

アドレス	+0	+1	+2	+3	レジスタ機能
0x2000_0400	HSSSWP	予約			HS_SPI アドレススワップ制御レジスタ

■ HS_SPI CSR 領域

アドレス	[31:24] (*1) +0 (*2) +3 (*3)	[23:16](*1) +1 (*2) +2 (*3)	[15:8](*1) +2 (*2) +1 (*3)	[7:0](*1) +3 (*2) +0 (*3)	レジスタ機能
0x2000_0000	HSSPIn_MCTRL				HS_SPI モジュール制御レジスタ
0x2000_0004	HSSPIn_PCC0				HS_SPI ペリフェラル通信設定レジスタ 0
0x2000_0008	HSSPIn_PCC1				HS_SPI ペリフェラル通信設定レジスタ 1
0x2000_000C	HSSPIn_PCC2				HS_SPI ペリフェラル通信設定レジスタ 2
0x2000_0010	HSSPIn_PCC3				HS_SPI ペリフェラル通信設定レジスタ 3
0x2000_0014	HSSPIn_TXF				HS_SPI TX 割込みフラグレジスタ
0x2000_0018	HSSPIn_TXE				HS_SPI TX 割込み許可レジスタ
0x2000_001C	HSSPIn_TXC				HS_SPI TX 割込みクリアレジスタ
0x2000_0020	HSSPIn_RXF				HS_SPI RX 割込みフラグレジスタ
0x2000_0024	HSSPIn_RXE				HS_SPI RX 割込み許可レジスタ
0x2000_0028	HSSPIn_RXC				HS_SPI RX 割込みクリアレジスタ
0x2000_002C	HSSPIn_FAULTF				HS_SPI Fault 割込みフラグレジスタ
0x2000_0030	HSSPIn_FAULTC				HS_SPI Fault 割込みクリアレジスタ
0x2000_0034	予約		HSSPIn_DMDMAEN	HSSPIn_DMCFG	HS_SPI ダイレクトモード DMA 許可レジスタ HS_SPI ダイレクトモード設定レジスタ

アドレス	[31:24] (*1) +0 (*2) +3 (*3)	[23:16](*1) +1 (*2) +2 (*3)	[15:8](*1) +2 (*2) +1 (*3)	[7:0](*1) +3 (*2) +0 (*3)	レジスタ機能
0x2000_0038	HSSPIn_DMTRP	HSSPIn_DMPSEL	HSSPIn_DMSTOP	HSSPIn_DMSTART	HS_SPI ダイレクトモード転送プロトコルレジスタ HS_SPI ダイレクトモードペリフェラル選択レジスタ HS_SPI ダイレクトモードストップレジスタ HS_SPI ダイレクトモードスタートレジスタ
0x2000_003C	HSSPIn_DMBCS		HSSPIn_DMBCC		HS_SPI ダイレクトモードバイトカウントステータスレジスタ HS_SPI ダイレクトモードバイトカウント制御レジスタ
0x2000_0040	HSSPIn_DMSTATUS				HS_SPI ダイレクトモードステータスレジスタ
0x2000_0044	予約		HSSPIn_RXBITCNT	HSSPIn_TXBITCNT	HS_SPI 受信ビットカウントレジスタ HS_SPI 送信ビットカウントレジスタ
0x2000_0048	HSSPIn_RXSHIFT				HS_SPI RX シフトレジスタ
0x2000_004C	HSSPIn_FIFOCFG				HS_SPI FIFO 設定レジスタ
0x2000_0050	HSSPIn_TXFIFO0				HS_SPI TX FIFO レジスタ 0
0x2000_0054	HSSPIn_TXFIFO1				HS_SPI TX FIFO レジスタ 1
0x2000_0058	HSSPIn_TXFIFO2				HS_SPI TX FIFO レジスタ 2
0x2000_005C	HSSPIn_TXFIFO3				HS_SPI TX FIFO レジスタ 3
0x2000_0060	HSSPIn_TXFIFO4				HS_SPI TX FIFO レジスタ 4
0x2000_0064	HSSPIn_TXFIFO5				HS_SPI TX FIFO レジスタ 5
0x2000_0068	HSSPIn_TXFIFO6				HS_SPI TX FIFO レジスタ 6
0x2000_006C	HSSPIn_TXFIFO7				HS_SPI TX FIFO レジスタ 7
0x2000_0070	HSSPIn_TXFIFO8				HS_SPI TX FIFO レジスタ 8
0x2000_0074	HSSPIn_TXFIFO9				HS_SPI TX FIFO レジスタ 9
0x2000_0078	HSSPIn_TXFIFO10				HS_SPI TX FIFO レジスタ 10
0x2000_007C	HSSPIn_TXFIFO11				HS_SPI TX FIFO レジスタ 11
0x2000_0080	HSSPIn_TXFIFO12				HS_SPI TX FIFO レジスタ 12
0x2000_0084	HSSPIn_TXFIFO13				HS_SPI TX FIFO レジスタ 13
0x2000_0088	HSSPIn_TXFIFO14				HS_SPI TX FIFO レジスタ 14
0x2000_008C	HSSPIn_TXFIFO15				HS_SPI TX FIFO レジスタ 15
0x2000_0090	HSSPIn_RXFIFO0				HS_SPI TX FIFO レジスタ 0
0x2000_0094	HSSPIn_RXFIFO1				HS_SPI TX FIFO レジスタ 1

アドレス	[31:24] (*1) +0 (*2) +3 (*3)	[23:16](*1) +1 (*2) +2 (*3)	[15:8](*1) +2 (*2) +1 (*3)	[7:0](*1) +3 (*2) +0 (*3)	レジスタ機能
0x2000_0098	HSSPIn_RXFIFO2				HS_SPI TX FIFO レジスタ 2
0x2000_009C	HSSPIn_RXFIFO3				HS_SPI TX FIFO レジスタ 3
0x2000_00A0	HSSPIn_RXFIFO4				HS_SPI TX FIFO レジスタ 4
0x2000_00A4	HSSPIn_RXFIFO5				HS_SPI TX FIFO レジスタ 5
0x2000_00A8	HSSPIn_RXFIFO6				HS_SPI TX FIFO レジスタ 6
0x2000_00AC	HSSPIn_RXFIFO7				HS_SPI TX FIFO レジスタ 7
0x2000_00B0	HSSPIn_RXFIFO8				HS_SPI TX FIFO レジスタ 8
0x2000_00B4	HSSPIn_RXFIFO9				HS_SPI TX FIFO レジスタ 9
0x2000_00B8	HSSPIn_RXFIFO10				HS_SPI TX FIFO レジスタ 10
0x2000_00BC	HSSPIn_RXFIFO11				HS_SPI TX FIFO レジスタ 11
0x2000_00C0	HSSPIn_RXFIFO12				HS_SPI TX FIFO レジスタ 12
0x2000_00C4	HSSPIn_RXFIFO13				HS_SPI TX FIFO レジスタ 13
0x2000_00C8	HSSPIn_RXFIFO14				HS_SPI TX FIFO レジスタ 14
0x2000_00CC	HSSPIn_RXFIFO15				HS_SPI TX FIFO レジスタ 15
0x2000_00D0	HSSPIn_CSCFG				HS_SPI コマンドシーケンサ設定レジスタ
0x2000_00D4	HSSPIn_CSITIME				HS_SPI コマンドシーケンサアイドルタイムレジスタ
0x2000_00D8	HSSPIn_CSAEXT				HS_SPI コマンドシーケンサアドレス拡張レジスタ
0x2000_00DC	HSSPIn_RDCSDC1		HSSPIn_RDCSDC0		HS_SPI リードコマンドシーケンスデータ制御レジスタ 1 HS_SPI リードコマンドシーケンスデータ制御レジスタ 0
0x2000_00E0	HSSPIn_RDCSDC3		HSSPIn_RDCSDC2		HS_SPI リードコマンドシーケンスデータ制御レジスタ 3 HS_SPI リードコマンドシーケンスデータ制御レジスタ 2
0x2000_00E4	HSSPIn_RDCSDC5		HSSPIn_RDCSDC4		HS_SPI リードコマンドシーケンスデータ制御レジスタ 5 HS_SPI リードコマンドシーケンスデータ制御レジスタ 4
0x2000_00E8	HSSPIn_RDCSDC7		HSSPIn_RDCSDC6		HS_SPI リードコマンドシーケンスデータ制御レジスタ 7 HS_SPI リードコマンドシーケンスデータ制御レジスタ 6

アドレス	[31:24] (*1) +0 (*2) +3 (*3)	[23:16](*1) +1 (*2) +2 (*3)	[15:8](*1) +2 (*2) +1 (*3)	[7:0](*1) +3 (*2) +0 (*3)	レジスタ機能
0x2000_00EC	HSSPIn_WRCSDC1		HSSPIn_WRCSDC0		HS_SPI ライトコマンドシーケンスデータ制御レジスタ 1 HS_SPI ライトコマンドシーケンスデータ制御レジスタ 0
0x2000_00F0	HSSPIn_WRCSDC3		HSSPIn_WRCSDC2		HS_SPI ライトコマンドシーケンスデータ制御レジスタ 3 HS_SPI ライトコマンドシーケンスデータ制御レジスタ 2
0x2000_00F4	HSSPIn_WRCSDC5		HSSPIn_WRCSDC4		HS_SPI ライトコマンドシーケンスデータ制御レジスタ 5 HS_SPI ライトコマンドシーケンスデータ制御レジスタ 4
0x2000_00F8	HSSPIn_WRCSDC7		HSSPIn_WRCSDC6		HS_SPI ライトコマンドシーケンスデータ制御レジスタ 7 HS_SPI ライトコマンドシーケンスデータ制御レジスタ 6
0x2000_00FC ～ 0x2000_03FC	予約				予約

*1: HSSSWP:WSWP=0 のときに、ワードアクセスで CSR ををアクセスした場合のビット位置
*2: HSSSWP:ADSWP=0 かつ HSSSWP:BSWP=1 のときに、バイトアクセスで CSR をアクセスした場合のアドレスオフセット
*3: HSSSWP:ADSWP=1 かつ HSSSWP:BSWP=0 のときに、バイトアクセスで CSR をアクセスした場合のアドレスオフセット

47.4.1 HS_SPI モジュール制御レジスタ ：HSSPIn_MCTRL (HS_SPI Module Control Register)

HS_SPI モジュール制御レジスタは HS_SPI モジュールを制御します。モジュール許可ビット，コマンドシーケンサ許可ビットなどの重要なビットを含んでいます。このレジスタを用いてモジュール動作の許可/禁止を行うことができます。

■ HS_SPI モジュール制御レジスタ ：HSSPIn_MCTRL (HS_SPI Module Control Register)

図 47-2 HS_SPI モジュール制御レジスタのビット構成

HSSPIn_MCTRL																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	MES	CDSS	予約	CSEN	MEN
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R/W	R0,W0	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-1 HSSPIn_MCTRL レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit[7:5]	read0	
bit4	MES	モジュール許可状態 : Module Enable Status (MES) 0: モジュールを完全に無効にし、省電力モードにします。 1: モジュールを有効にします。
bit3	CDSS	クロック分周ソース選択 : Clock Division Source Select (CDSS) HS_SPI がマスタモードである場合、内部クロック分周器はオンチップバスクロック(iHCLK)かペリフェラルクロック(iPCLK)を選択して、これを分周します。CDSS ビットで、クロック分周器がどちらのクロックを分周するかを決めます。また、このビットはスレーブモードでは使用されません。 0: クロック分周器は iHCLK を分周します。 1: クロック分周器は iPCLK を分周します。 コマンドシーケンサモード(CSEN=0)のときは本ビットを 0 に設定してください。 PLL クロックが停止または安定化を待っている間、このビットを"1"に設定することは禁止です。
bit2	予約	必ず"0"を書き込んでください。
bit1	CSEN	コマンドシーケンサ許可 : Command Sequencer Enable (CSEN) 0: ダイレクトモード許可, コマンドシーケンサ禁止 1: ダイレクトモード禁止, コマンドシーケンサ許可
bit0	MEN	モジュール許可 : Module Enable (MEN) 0: モジュール動作禁止 HS_SPI モジュールは省電力モードになります。すべてのシリアル入出力信号が HS_SPI によって Hi-Z 状態となります。 (端子機能として HS_SPI が選択されている場合) 1: モジュール動作許可 HS_SPI の各レジスタ設定後、本ビットを"1"に設定することにより、HS_SPI は動作可能となります。 ソフトウェアは本ビットをリセットした場合: a) ダイレクトモード:マスタ動作では、スレーブ選択を解除した後のSPI転送を停止します。スレーブ動作では、スレーブ選択が解除された後はSPI転送に対応しません。スレーブ選択が解除された後、HS_SPIモジュール内部へのiHCLKとiPCLK供給を停止し、HS_SPIモジュールは省電力モードに入ります。 b) コマンドシーケンサモード: メモリマップされたデバイスへのオンチップバスアクセスを受け取ると、HS_SPIは未配置メモリアクセスフォルト割込みを発生します。シリアルインタフェース上のいかなる命令も発生しません。スレーブ選択が解除された後、HS_SPI内部へのiHCLKとiPCLKの両クロック供給を停止し、省電力モードに入ります。

47.4.2 HS_SPI ペリフェラル通信設定レジスタ 0~3 : HSSPIn_PCC0~3 (HS_SPI Peripheral Communication Configuration Register 0~3)

HS_SPI ペリフェラル通信設定レジスタ 0~3 はスレーブ選択 0~3 のシリアル通信に関連する属性を制御します。ソフトウェアは HS_SPI の対応するスレーブ選択信号(0~3)に接続されるシリアルペリフェラルの通信属性に一致した属性値を、これらのレジスタに設定しなければなりません。HS_SPI がダイレクトモード, またはコマンドシーケンサモードで SPI マスタとして設定されると、4 つのレジスタすべてが使用されます。HS_SPI がダイレクトモードで SPI スレーブとして設定されると、HSSPIn_PCC0 レジスタだけが使用されます。
ここでは、HSSPIn_PCC0 レジスタのみ説明します。ほかのレジスタ(HSSPIn_PCC1, HSSPIn_PCC2, および HSSPIn_PCC3)も、同様のビット構成になっています。

■ HS_SPI ペリフェラル通信設定レジスタ 0 (HSSPIn_PCC0)

図 47-3 HS_SPI ペリフェラル通信設定レジスタ 0 のビット構成

HSSPIn_PCC0																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	SAFESYNC	CDRS[6]	CDRS[5]	CDRS[4]	CDRS[3]	CDRS[2]	CDRS[1]	CDRS[0]	Read0	SDIR	SS2CD[1]	SS2CD[0]	SSPOL	RTM	ACES	CPOL	CPHA
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-2 HSSPIn_PCC0 レジスタのビット説明

ビット名		説明
bit[31:24]	read0	
bit[23:17]	read0	
bit16	SAFESYNC	ペリフェラル 0 のための安全同期 : Safe Synchronisation for Peripheral 0 (SAFESYNC) HS_SPI がダイレクトモード, またはコマンドシーケンサモードで SPI マスタとして設定されているとき、このビットを設定します。 0: モジュールは通常動作します。シリアル転送をしている期間は、SP_SPI はデータの安全同期を実行しません。 1: ペリフェラル 0 とのシリアル通信が行われている間、モジュールはデータの安全同期のため、設定された遅延を付加します。 有効な SAFESYNC ビット設定については、「47.5.1.6. シリアルクロック周波数」を参照してください。

ビット名		説明
bit[15:9]	CDRS[6:0]	ペリフェラル0のクロック分周比選択 ：Clock Division Ratio Select of Peripheral 0 (CDRS[6:0]) HS_SPI が、ダイレクトモード, またはコマンドシーケンサモードで SPI マスタとして設定されているとき、このフィールドで、内部クロック分周器のクロック分周比を設定します。HS_SPI が SPI スレーブモードとして設定されているとき、このフィールドは使用しません。 0: 設定禁止 1: 2 分周 2: 4 分周 3: 6 分周 ... 127: 254 分周 一般に、クロック周波数(Fo)はソースクロック周波数(Fi)を CDSR の 2 倍の値で割ることにより得られます(CDRS はゼロ以外)。 Fo = Fi / (2 × CDRS) CDRS 値は、分周後のシリアルクロックの周波数がオンチップバスクロックの周波数よりも高くならないように、設定しなければなりません。 設定可能な分周比については「47.5.1.6. シリアルクロック周波数」を参照してください。
bit8	read0	
bit7	SDIR	ペリフェラル 0 のシフト方向 ：Shift Direction of Peripheral 0 (SDIR) 本ビットは、フィールド内におけるビット転送の順序(すなわちシフト方向)を決めます。データレジスタにおける最上位ビット(MSB)と最下位ビット(LSB)の位置は SDIR ビットの設定内容に係わらず変わりません。データレジスタへのリード/ライトアクセスでは、bit0 が常に最下位ビット(LSB)になります。 0: 最上位ビット(MSB)を最初に送信 1: 最下位ビット(LSB)を最初に送信
bit[6:5]	SS2CD[1:0]	スレーブ選択からペリフェラル0のクロック遅延: Slave-Select to Clock Delay of Peripheral 0 (SS2CD[1:0]) これらのビットはダイレクトモード, またはコマンドシーケンサモードで HS_SPI が SPI マスタに設定されているときのみ使用できます。 これらのビットによって、スレーブデバイスに対するセットアップ時間を定めます。SCLK のトグルを遅らせることにより、HS_SPI はスレーブへのデータ送信をチップセレクトアクティブエッジから SCLK のサイクル単位で遅らせます。 HSSPIn_PCC0~3:CPHA=0 の場合、スレーブ選択のアサートと SCLK の最初のエッジの間の遅れは、SCLK で(SS2CD ＋ 0.5)サイクルとなります。 HSSPIn_PCC0~3:CPHA=1 の場合、スレーブ選択のアサートと SCLK の最初のエッジの間の遅れは、SCLK で(SS2CD)サイクルとなります。 スレーブ選択が有効になった場合、スレーブは SS2CD ビットにより決められた遅延時間以内にデータ転送の準備をする必要があります。

ビット名		説明
bit4	SSPOL	ペリフェラル 0 のスレーブ選択極性 ：Slave Select Polarity of Peripheral 0 (SSPOL) 本ビットはスレーブ選択信号(SSEL0)の極性を決定します。 0: SSEL0 はデフォルト状態の間"H"を保持します。信号は"L"でアクティブです。 1: SSEL0 はデフォルト状態の間"L"を保持します。信号は"H"でアクティブです。
bit3	RTM	タイミング補償されたクロックを用いたペリフェラル 0 からのデータ取込み ：Use retimed clock for Capturing the data from Peripheral 0 (RTM) HS_SPI に接続されたシリアルデバイスの Setup/Hold マージンが厳しい場合は、このビットを"1"にセットしてください。このビットは HS_SPI がダイレクトモード, またはコマンドシーケンスモードで SPI マスタとして設定されている場合のみ有効です。 0: タイミング補正されたクロックを使わずに、シリアルデータを取り込みます。 1: タイミング補正されたクロックを使って、シリアルデータを取り込みます。 HS_SPI シリアルインタフェースのクロック動作モードは、CPHA, CPOL, ACES, RTM ビットの組み合わせにより決定します。
bit2	ACES	ペリフェラル 0 のアクティブクロックエッジは同一である ：Active Clock Edges are Same on Peripheral 0 (ACES) このビットはデータの送信に使われるクロックエッジとデータ受信のクロックエッジが同じであるか、別であるかを決定します。このビットは HS_SPI がダイレクトモード, またはコマンドシーケンスモードで SPI マスタとして設定されている場合のみ有効です。 0: データの送信/受信でクロックの異なる(反対の)エッジを使用します。 1: データの送信/受信でクロックの同じエッジを使用します。 HS_SPI シリアルインタフェースのクロック動作モードは、CPHA, CPOL, ACES, RTM ビットの組み合わせにより決定します。 ダイレクトモード時のレガシーTX-and-RX の設定に対し、ACES=1 の設定はサポートされていません。ダイレクトモード時でレガシーTX-and-RX 設定の場合(HSSPIn_DMTRP:TRP[3:0]=0000)は、ACES を 0 に設定してください。
bit1	CPOL	ペリフェラル 0 のクロック極性 ：Clock Polarity of Peripheral 0 (CPOL) 0:デフォルト状態の間、SCLK を L に保持します。 1:デフォルト状態の間、SCLK を H に保持します。 HS_SPI シリアルインタフェースのクロック動作モードは、CPHA, CPOL, ACES, RTM ビットの組み合わせにより決定します。
bit0	CPHA	ペリフェラル 0 のクロック位相 ：Clock Phase of Peripheral 0 (CPHA) 0: 入力データは、シリアルクロックの奇数番目のエッジでサンプリングされます。 1: 入力データは、シリアルクロックの偶数番目のエッジでサンプリングされます。 HS_SPI シリアルインタフェースのクロック動作モードは、CPHA, CPOL, ACES, RTM ビットの組み合わせにより決定します。

47.4.3 HS_SPI TX 割込みフラグレジスタ (HSSPIn_TXF)

HS_SPI TX 割込みフラグレジスタは TX 割込みの要因を示します。これらの割込みフラグはモジュールが許可(HSSPIn_MCTRL:MES=1)され、ダイレクトモードの場合のみ設定できます。ソフトウェアはこれらの割込みを許可し TX 割込み発生を待つか、本レジスタのポーリングを行います。

■ HS_SPI TX 割込みフラグレジスタ (HSSPIn_TXF)

図 47-4 HS_SPI TX 割込みフラグレジスタのビット構成

HSSPIn_TXF																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	TSSRS	TFMTS	TFLETS	TFUS	TFOS	TFES	TFFS
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-3 HSSPIn_TXF レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit7	read0	
bit6	TSSRS	スレーブ選択解除 : Slave Select Released (TSSRS) この割込みフラグは、スレーブ選択信号が SPI マスタにより解除されたことを示します。 HSSPIn_TXE:TSSRE が有効の場合、TX 割込みが発生します。
bit5	TFMTS	TX-FIFO フィルレベルしきい値超 : TX-FIFO Fill Level is More Than Threshold (TFMTS) この割込みフラグは、受信データが TX-FIFO の設定しきい値を超えているとき(HSSPIn_DMSTATUS:TXFLEVEL ＞ HSSPIn_FIFOCFG:TXFTH)、オンチップバスクロックに同期してセットされます。 HSSPIn_TXE:TFMTE が有効の場合、TX 割込みが発生します。
bit4	TFLETS	TX-FIFO フィルレベルしきい値以下 : TX-FIFO Fill Level is Less Than or Equal to Threshold (TFLETS) この割込みフラグは、受信データが TX-FIFO の設定しきい値に等しいか、またはそれ以下のとき(HSSPIn_DMSTATUS:TXFLEVEL ≦ HSSPIn_FIFOCFG:TXFTH)、オンチップバスクロックに同期してセットされます。 HSSPIn_TXE:TFLETE が有効の場合、TX 割込みが発生します。

ビット名		説明
bit3	TFUS	TX-FIFO アンダラン : TX-FIFO Underrun (TFUS) この割込みフラグは TX-FIFO のアンダランを示します。TX-FIFO アンダランは TX-FIFO が空のときに、TX-FIFO の読み出しにより発生します。また、スレーブモード動作中に起こることもあります。HSSPIn_TXE:TFUE が有効の場合、TX 割込みが発生します。
bit2	TFOS	TX-FIFO オーバラン : TX-FIFO Overrun (TFOS) この割込みフラグはTX-FIFOがオーバランしていることを示します。TX-FIFO オーバランは TX-FIFO がフルで HSSPIn_TXFIFO0~15 レジスタが書き込まれたときに起こります。 HSSPIn_TXE:TFOE が有効の場合、TX 割込みが発生します。
bit1	TFES	TX-FIFO とシフトレジスタが空 : TX-FIFO and Shift Register is Empty (TFES) この割込みフラグは、TX-FIFO と SPI コアの TX シフトレジスタが共に空のときに、オンチップバスクロックに同期してセットされます。 HSSPIn_TXE:TFEE が有効の場合、TX 割込みが発生します。
bit0	TFFS	TX-FIFO フル : TX-FIFO Full (TFFS) この割込みフラグは TX-FIFO がフルのときに、オンチップバスクロックに同期してセットされます。 HSSPIn_TXE:TFFE が有効の場合、TX 割込みが発生します。

47.4.4 HS_SPI TX 割込み許可レジスタ (HSSPIn_TXE)

HS_SPI TX 割込み許可レジスタは、HSSPIn_TXF レジスタで示される割込み要因が有効の場合、TX 割込みを発生するかどうかを決定します。TX 割込み信号の発生のため、これらのフラグを有効にする必要があります。

■ HS_SPI TX 割込み許可レジスタ (HSSPIn_TXE)

図 47-5 HS_SPI TX 割込み許可レジスタのビット構成

HSSPIn_TXE																													
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	TSSRE	TFMTE	TFLETE	TFUE	TFOE	TFEE	TFFE
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-4 HSSPIn_TXE レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit7	read0	
bit6	TSSRE	スレーブ選択解除割込み許可 : Slave Select Released Interrupt Enable (TSSRE) このビットは HSSPIn_TXF:TSSR 割込み要因による TX 割込み生成を制御します。 0: HSSPIn_TXF:TSSRS 割込み要因による TX 割込み生成を禁止します。 1: HSSPIn_TXF:TSSRS 割込み要因による TX 割込み生成を許可します。
bit5	TFMTE	TX-FIFO フィルレベルしきい値割込み許可超:TX-FIFO Fill Level is More Than Threshold Interrupt Enable (TFMTE) このビットは HSSPIn_TXF:TFMTS 割込み要因による TX 割込み生成を制御します。 0: HSSPIn_TXF:TFMTS 割込み要因による TX 割込み生成を禁止します。 1: HSSPIn_TXF:TFMTS 要因による TX 割込み生成を許可します。
bit4	TFLETE	TX-FIFO フィルレベルしきい値割込み許可以下 : TX-FIFO Fill Level is Less Than or Equal To Threshold Interrupt Enable (TFLETE) このビットは HSSPIn_TXF:TFLETS 割込み要因による TX 割込み生成を制御します。 0: HSSPIn_TXF:TFLETS 割込み要因による TX 割込み生成を禁止します。 1: HSSPIn_TXF:TFLETS 要因による TX 割込み生成を許可します。

ビット名		説明
bit3	TFUE	TX-FIFO アンダラン割込み許可: TX-FIFO Underrun Interrupt Enable (TFUE) このビットは HSSPIn_TXF:TFUS 割込み要因による TX 割込み生成を制御します。 0: HSSPIn_TXF:TFUS 割込み要因による TX 割込み生成を禁止します。 1: HSSPIn_TXF:TFUS 割込み要因による TX 割込み生成を許可します。
bit2	TFOE	TX-FIFO オーバラン割込み許可 : TX-FIFO Overrun Interrupt Enable (TFOE) このビットは HSSPIn_TXF:TFOS 割込み要因による TX 割込み生成を制御します。 0: HSSPIn_TXF:TFOS 割込み要因による TX 割込み生成を禁止します。 1: HSSPIn_TXF:TFOS 割込み要因による TX 割込み生成を許可します。
bit1	TFEE	TX-FIFO 空割込み許可 : TX-FIFO Empty Interrupt Enable (TFEE) このビットは HSSPIn_TXF:TFES 割込み要因による TX 割込み生成を制御します。 0: HSSPIn_TXF:TFES 割込み要因による TX 割込み生成を禁止します。 1: HSSPIn_TXF:TFES 割込み要因による TX 割込み生成を許可します。
bit0	TFFE	TX-FIFO フル割込み許可 : TX-FIFO Full Interrupt Enable (TFFE) このビットは HSSPIn_TXF:TFFS 割込み要因による TX 割込み生成を制御します。 0: HSSPIn_TXF:TFFS 割込み要因による TX 割込み生成を禁止します。 1: HSSPIn_TXF:TFFS 割込み要因による TX 割込み生成を許可します。

47.4.5 HS_SPI TX 割込みクリアレジスタ (HSSPIn_TXC)

HS_SPI TX 割込みクリアレジスタにより HSSPIn_TXF レジスタに保持された割込みフラグをクリアします。このレジスタのビットに“1”を書き込むと、HSSPIn_TFX レジスタの対応するフラグがクリアされます。

■ HS_SPI TX 割込みクリアレジスタ (HSSPIn_TXC)

図 47-6 HS_SPI TX 割込みクリアレジスタのビット構成

HSSPIn_TXC																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	TSSRC	TFMTC	TFLETC	TFUC	TFOC	TFEC	TFFC
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-5 HSSPIn_TXC レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit7	read0	
bit6	TSSRC	スレーブ選択解除割込みクリア ：Slave Select Released Interrupt Clear (TSSRC) このビットは、HSSPIn_TXF:TSSRS 割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_TXF:TSSRS フラグをクリアします。 読出し値は"0"です。
bit5	TFMTC	TX-FIFO フィルレベルしきい値超割込みクリア:TX-FIFO Fill Level More Than Threshold Interrupt Clear (TFMTC) このビットは、HSSPIn_TXF:TFMTS 割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_TXF:TFMTS 割込みフラグをクリアします。 読出し値は"0"です。
bit4	TFLETC	TX-FIFO フィルレベルしきい値以下割込みクリア：TX-FIFO Fill Level Less Than or Equal to Threshold Interrupt Clear (TFLETC) このビットは、HSSPIn_TXF:TFLETS 割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_TXF:TFLETS 割込みフラグをクリアします。 読出し値は"0"です。
bit3	TFUC	TX-FIFO アンダラン割込みクリア ：TX-FIFO Underrun Interrupt Clear (TFUC) このビットは、HSSPIn_TXF:TFUS 割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_TXF:TFUS 割込みフラグをクリアします。 読出し値は"0"です。

ビット名		説明
bit2	TFOC	TX-FIFO オーバラン割込みクリア : TX-FIFO Overrun Interrupt Clear (TFOC) このビットは、HSSPIn_TXF:TFOS 割込みフラグをクリアするために使用します。 0 書込み: 無効。 1 書込み: HSSPIn_TXF:TFOS 割込みフラグをクリアします。 読出し値は"0"です。
bit1	TFEC	TX-FIFO エンプティ割込みクリア : TX-FIFO Empty Interrupt Clear (TFEC) このビットは、HSSPIn_TXF:TFES 割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_TXF:TFES 割込みフラグをクリアします。 読出し値は"0"です。
bit0	TFFC	TX-FIFO フル割込みクリア : TX-FIFO Full Interrupt Clear (TFFC) このビットは、HSSPIn_TXF:TFFS 割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_TXF:TFFS 割込みフラグをクリアします。 読出し値は"0"です。

47.4.6 HS_SPI RX 割込みフラグレジスタ (HSSPIn_RXF)

HS_SPI RX割込みフラグレジスタはRX割込みの要因を示します。モジュールが動作許可(HSSPIn_MCTRL:MES=1)であり、ダイレクトモード時のみ各要因は意味を持ちます。
ソフトウェアは、これらの割込みを許可し RX 割込み発生を待つか、本レジスタのポーリングを行います。

■ HS_SPI RX 割込みフラグレジスタ (HSSPIn_RXF)

図 47-7 HS_SPI RX 割込みフラグレジスタのビット構成

HSSPIn_RXF																														
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	RSSRS	RFMTS	RFLETS	RFUS	RFOS	RFES	RFFS
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-6 HSSPIn_RXF レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit7	read0	
bit6	RSSRS	スレーブ選択解除 : Slave Select Released (RSSRS) この割込みフラグは、スレーブ選択信号が SPI マスタにより解除されたことを示します。 HSSPIn_RXE:RSSRE が有効の場合、RX 割込み信号が発生します。
bit5	RFMTS	RX-FIFO フィルレベルしきい値超 : RX-FIFO Fill Level is More Than Threshold (RFMTS) RX-FIFO のフィルレベルが設定されたしきい値を超えている (HSSPIn_DMSTATUS:RXFLEVEL>HSSPIn_FIFOCFG:RXFT)場合、この割込みフラグはオンチップバスクロックに同期してセットされます。 HSSPIn_RXE:RFMTE が有効の場合、RX 割込み信号が発生します。
bit4	RFLETS	RX-FIFO フィルレベルしきい値以下 : RX-FIFO Fill Level is Less Than or Equal to Threshold (RFLETS) RX-FIFOのフィルレベルが設定されたしきい値以下 (HSSPIn_DMSTATUS:RXFLEVEL≦HSSPIn_FIFOCFG:RXFT)の場合、この割込みフラグはオンチップバスクロックごに同期してセットされます。 HSSPIn_RXE:RFLETE が有効の場合、RX割込みが発生します。

ビット名		説明
bit3	RFUS	RX-FIFO アンダラン ：RX-FIFO Underrun (RFUS) この割込みフラグは RX-FIFO がアンダランを示します。RX-FIFO アンダランは RX-FIFO が空のときに(オンチップバスマスタが)HSSPIn_RXFIFO0~15 レジスタの読み出しにより発生します。 HSSPIn_RXE:RFUE が有効な場合、RX 割込みが発生します。
bit2	RFOS	RX-FIFO オーバラン ：RX-FIFO Overrun (RFOS) この割込みフラグは RX-FIFO がオーバランであることを示します。RX-FIFO オーバランは RX-FIFO がフルのときに、RX-FIFO に書込みにより発生します。また、スレーブモード動作の際も発生することがあります。 HSSPIn_RXE:RFOE が有効な場合、RX 割込みが発生します。
bit1	RFES	RX-FIFO エンプティ ：RX-FIFO Empty (RFES) この割込みフラグは RX-FIFO が空のときにオンチップバスクロックに同期してセットされます。 HSSPIn_RXE:RFEE が有効な場合、RX 割込みが発生します。
bit0	RFFS	RX-FIFO フル ：RX-FIFO Full (RFFS) この割込みフラグは RX-FIFO がフルのときにオンチップバスクロックに同期してセットされます。 HSSPIn_RXE:RFFE が有効な場合、RX 割込みが発生します。

47.4.7 HS_SPI RX 割込み許可レジスタ (HSSPIn_RXE)

HS_SPI RX 割込み許可レジスタは HSSPIn_RXF レジスタで示される割込み要因が有効場合、RX 割込みを発生するかどうかを決定します。
RX 割込み信号の発生のため、これらのフラグを有効にする必要があります。

■ HS_SPI RX 割込み許可レジスタ (HSSPIn_RXE)

図 47-8 HS_SPI RX 割込み許可レジスタのビット構成

HSSPIn_RXE																														
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	RSSRE	RFMTE	RFLETE	RFUE	RFOE	RFEE	RFFE
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-7 HSSPIn_RXE レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit7	read0	
bit6	RSSRE	スレーブ選択解除割込み許可 ：Slave Select Released Interrupt Enable (RSSRE) このビットで HSSPIn_RXF:RSSRS 割込み要因による RX 割込み生成を制御します。 0: HSSPIn_RXF:RSSRS 割込み要因による RX 割込み生成を禁止します。 1: HSSPIn_RXF:RSSRS 割込み要因による RX 割込み生成を許可します。
bit5	RFMTE	RX-FIFO フィルレベルしきい値超割込み許可 ：RX-FIFO Fill Level is More Than Threshold Interrupt Enable (RFMTE) このビットで HSSPIn_RXF:RFMTS 割込み要因による RX 割込み生成を制御します。 0: HSSPIn_RXF:RFMTS 割込み要因による RX 割込み生成を禁止します。 1: HSSPIn_RXF:RFMTS 割込み要因による RX 割込み生成を許可します。
bit4	RFLETE	RX-FIFO フィルレベルしきい値以下割込み許可：RX-FIFO Fill Level is Less Than or Equal To Threshold Interrupt Enable (RFLETE) このビットで HSSPIn_RXF:RFLETS 割込み要因による RX 割込み生成を制御します。 0: HSSPIn_RXF:RFLETS 割込み要因による RX 割込み生成を禁止します。 1: HSSPIn_RXF:RFLETS 割込み要因による RX 割込み生成を許可します。

ビット名		説明
bit3	RFUE	RX-FIFO アンダラン割込み許可 : RX-FIFO Underrun Interrupt Enable (RFUE) このビットで HSSPIn_RXF:RFUS 割込み要因による RX 割込み生成を制御します。 0: HSSPIn_RXF:RFUS 割込み要因による RX 割込み生成を禁止します。 1: HSSPIn_RXF:RFUS 割込み要因による RX 割込み生成を許可します。
bit2	RFOE	RX-FIFO オーバラン割込み許可 : RX-FIFO Overrun Interrupt Enable (RFOE) このビットで HSSPIn_RXF:RFOS 割込み要因による RX 割込み生成を制御します。 0: HSSPIn_RXF:RFOS 割込み要因による RX 割込み生成を禁止します。 1: HSSPIn_RXF:RFOS 割込み要因による RX 割込み生成を許可します。
bit1	RFEE	RX-FIFO エンプティ割込み許可 : RX-FIFO Empty Interrupt Enable (RFEE) このビットで HSSPIn_RXF:RFES 割込み要因による RX 割込み生成を制御します。 0: HSSPIn_RXF:RFES 割込み要因による RX 割込み生成を禁止します。 1: HSSPIn_RXF:RFES 割込み要因による RX 割込み生成を許可します。
bit0	RFFE	RX-FIFO フル割込み許可 : RX-FIFO Full Interrupt Enable (RFFE) このビットで HSSPIn_RXF:RFFS 割込み要因による RX 割込み生成を制御します。 0: HSSPIn_RXF:RFFS 割込み要因による RX 割込み生成を禁止します。 1: HSSPIn_RXF:RFFS 割込み要因による RX 割込み生成を許可します。

47.4.8 HS_SPI RX 割込みクリアレジスタ (HSSPIn_RXC)

HS_SPI RX 割込みクリアレジスタにより HSSPIn_RXF レジスタに保持された割込みフラグをクリアします。このレジスタのビットに“1”を書き込むと、HSSPIn_RXF レジスタの対応するフラグをクリアします。

■ HS_SPI RX 割込みクリアレジスタ (HSSPIn_RXC)

図 47-9 HS_SPI RX 割込みクリアレジスタのビット構成

HSSPIn_RXC																														
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	RSSRC	RFMTC	RFLETC	RFUC	RFOC	RFEC	RFFC
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-8 HSSPIn_RXC レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit7	read0	
bit6	RSSRC	スレーブ選択解除割込みクリア ：Slave Select Released Interrupt Clear (RSSRC) このビットは、HSSPIn_RXF:RSSRS 割込みフラグのクリアに使用します。 0: 無効 1: HSSPIn_RXF:RSSRS 割込みフラグをクリアします。 読出し値は"0"です。
bit5	RFMTC	RX-FIFO フィルレベルしきい値超割込みクリア ：RX-FIFO Fill Level More Than Threshold Interrupt Clear (RFMTC) このビットは、HSSPIn_RXF:RFMTS 割込みフラグのクリアに使用します。 0: 無効。 1: HSSPIn_RXF:RFMTS 割込みフラグをクリアします。 読出し値は"0"です。
bit4	RFLETC	RX-FIFO フィルレベルしきい値以下割込みクリア ：RX-FIFO Fill Level Less Than or Equal to Threshold Interrupt Clear (RFLETC) このビットは、HSSPIn_RXF:RFLETS 割込みフラグのクリアに使用します。 0: 無効。 1: HSSPIn_RXF:RFLETS 割込みフラグをクリアします。 読出し値は"0"です。
bit3	RFUC	RX-FIFO アンダラン割込みクリア ：RX-FIFO Underrun Interrupt Clear (RFUC) このビットは、HSSPIn_RXF:RFUS 割込みフラグのクリアに使用します。 0: 無効。 1: HSSPIn_RXF:RFUS 割込みフラグをクリアします。 読出し値は"0"です。

ビット名		説明
bit2	RFOC	RX-FIFO オーバラン割込みクリア : RX-FIFO Overrun Interrupt Clear (RFOC) このビットは、HSSPIn_RXF:RFOS 割込みフラグのクリアに使用します。 0: 無効。 1: HSSPIn_RXF:RFOS 割込みフラグをクリアします。 読出し値は"0"です。
bit1	RFEC	RX-FIFO エンプティ割込みクリア : RX-FIFO Empty Interrupt Clear (RFEC) このビットは、HSSPIn_RXF:RFES 割込みフラグのクリアに使用します。 0: 無効。 1: HSSPIn_RXF:RFES 割込みフラグをクリアします。 読出し値は"0"です。
bit0	RFFC	RX-FIFO フル割込みクリア : RX-FIFO Full Interrupt Clear (RFFC) このビットは、HSSPIn_RXF:RFFS 割込みフラグのクリアに使用します。 0: 無効。 1: HSSPIn_RXF:RFFS 割込みフラグをクリアします。 読出し値は"0"です。

47.4.9 HS_SPI フォルト割込みフラグレジスタ (HSSPIn_FAULTF)

HS_SPI フォルト割込みフラグレジスタはフォルト割込みの要因を示します。このフォルト割込みは HS_SPI ではマスクできません。
フォルトが発生した場合は、ソフトウェアによる適切な処理を行ってください。

■ HS_SPI フォルト割込みフラグレジスタ (HSSPIn_FAULTF)

図 47-10 HS_SPI フォルト割込みフラグレジスタのビット構成

HSSPIn_FAULTF																														
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	WAFS	UMAFS
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

＜注意事項＞
CY91570 シリーズでは、フォルト割込み要因は CPU に通知されません。フォルト割込み要因の発生を確認するためにこのレジスタをポーリングする必要があります。

表 47-9 HSSPIn_FAULTF レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit[7:2]	read0	
bit1	WAFS	書込みアクセスフォルト : Write Access Fault (WAFS) この割込みフラグは書込み Write Access Fault (WAFS)が起こったことを示します。この割込みフラグは HS_SPI モジュールではマスクできません。 コマンドシーケンスモードで、HSSPIn_CSCFG:SRAM=0 かつオンチップバスマスタが HS_SPI のメモリ領域へのライトアクセスが発生すると、本ビットがセットされます。
bit0	UMAFS	未配置メモリアクセスフォルト : Unmapped Memory Access Fault (UMAFS) この割込みフラグは Unmapped Memory Access Fault (UMAFS)がおこったことを示します。この割込みフラグは HS_SPI モジュールではマスクできません。 下記いずれかの発生でセットされます。 (a) ダイレクトモード(HSSPIn_MCTRL:CSEN=0)のとき、HS_SPI メモリ領域へのオンチップバスのアクセスが検出された。 (b) コマンドシーケンスモード(HSSPIn_MCTRL:CSEN=1)のとき、HSSPIn_CSCFG:SSEL0EN ～ SSEL3EN により許可されていないメモリデバイスへのオンチップバスのアクセスが検出された。 (c) コマンドシーケンスモード(HSSPIn_MCTRL:CSEN=1)のとき、HSSPIn_CSCFG:MSEL フィールドで設定された 4 つのスレーブ選択上に配置されたメモリ領域の外へのオンチップバスからのアクセスが発生した。 (d) モジュール無効(HSSPIn_MCTRL:MEN=0)のとき、HS_SPI メモリ領域に対してオンチップバスからアクセスが発生した。

47.4.10 HS_SPI フォルト割込みクリアレジスタ (HSSPIn_FAULTC)

HS_SPI フォルト割込みクリアレジスタは、HSSPIn_FAULTF レジスタに保持された割込みフラグをクリアするために使用されます。
このレジスタのビットに”1”を書き込むことによって、HSSPIn_FAULTF レジスタの対応するフラグをクリアすることができます。

■ HS_SPI フォルト割込みクリアレジスタ (HSSPIn_FAULTC)

図 47-11 HS_SPI フォルト割込みクリアレジスタのビット構成

HSSPIn_FAULTC																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	WAFC	UMAFC
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,W	R0,W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-10 HSSPIn_FAULTC レジスタのビット説明

ビット名		説明
bit[31:8]	read0	
bit[7:2]	read0	
bit1	WAFC	書込みアクセスフォルト割込みクリア ：Write Access Fault Interrupt Clear (WAFC) このビットは HSSPIn_FAULTF:WAFS 割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_FAULTF:WAFS 割込みフラグをクリアします。 読出し値は"0"です。
bit0	UMAFC	未配置メモリアクセスフォルト割込みクリア ：Unmapped Memory Access Fault Interrupt Clear (UMAFC) このビットはHSSPIn_FAULTF:UMAFS割込みフラグをクリアするために使用します。 0: 無効。 1: HSSPIn_FAULTF:UMAFS 割込みフラグをクリアします。 読出し値は"0"です。

47.4.11 HS_SPI ダイレクトモード設定レジスタ (HSSPIn_DMCFG)

HS_SPI ダイレクトモード設定レジスタにより、下記の HS_SPI 動作パラメータを設定します。

- マスタモード動作またはスレーブモード動作
- ソフトウェアのフロー制御またはスレーブ選択停止のバイトカウンタモード

このレジスタは、HS_SPI モジュールがダイレクトモード時のみ使用できます。

■ HS_SPI ダイレクトモード設定レジスタ (HSSPIn_DMCFG)

図 47-12 HS_SPI ダイレクトモード設定レジスタのビット構成

HSSPIn_DMCFG							
7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	予約	SSDC	MST
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W0	R/W	R/W
0	0	0	0	0	0	0	1

Access : Byte, Half-word, Word

表 47-11 HSSPIn_DMCFG レジスタのビット説明

ビット名		説明
bit[7:3]	read0	
bit2	予約	必ず"0"を書き込んでください。
bit1	SSDC	スレーブ選択アサート停止制御 : Slave Select Deassertion Control (SSDC) このビットは HS_SPI がマスタ動作時(HSSPIn_DMCFG:MST=1)のみに使用します。 SSDC ビットは HS_SPI がマスタとして動作するとき、どのようにスレーブ選択を停止するか決定します。 0: ソフトウェアフロー制御 HSSPIn_DMSTOP:STOP ビットにより、スレーブ選択停止となります。 1: バイトカウンタモード HSSPIn_DMBCC:BCC により、スレーブ選択停止となります。
bit0	MST	マスタモード : Master Mode (MST) 0: HS_SPI はスレーブモードとなります。 1: HS_SPI はマスタモードとなります。

47.4.12 HS_SPI ダイレクトモード DMA 許可レジスタ (HSSPIn_DMDMAEN)

このレジスタの機能は、CY91570 シリーズではサポートされていません。
ライトする場合は"0"を書き込んでください。

■ HS_SPI ダイレクトモード DMA 許可レジスタ (HSSPIn_DMDMAEN)

図 47-13 HS_SPI ダイレクトモード DMA 許可レジスタのビット構成

HSSPIn_DMDMAEN							
7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	TXDMAEN	RXDMAEN
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W0	R/W0
0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-12 HSSPIn_DMDMAEN レジスタのビット説明

ビット名		機能
bit[7:2]	Read0	
bit1	TXDMAEN	このビットには必ず"0"を書き込んでください。
bit0	RXDMAEN	このビットには必ず"0"を書き込んでください。

47.4.13 HS_SPI ダイレクトモードスタートレジスタ (HSSPIn_DMSTART)

HS_SPI がマスタとして動作しているとき、シリアル転送開始を HS_SPI ダイレクトモードスタートレジスタにより制御することができます。
このレジスタは、HS_SPI モジュールがダイレクトモードでマスタとして動作しているときにのみ使用することができます。

■ HS_SPI ダイレクトモードスタートレジスタ (HSSPIn_DMSTART)

図 47-14 HS_SPI ダイレクトモードスタートレジスタのビット構成

HSSPIn_DMSTART							
7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	START
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,W
0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-13 HSSPIn_DMSTART レジスタのビット説明

ビット名		説明
bit[7:1]	read0	
bit0	START	転送スタート: Start Transfer (START) 本ビットは HS_SPI がマスタモード(HSSPIn_DMCFG:MST=1)のときにのみ使用可能です。 0: 無効。 1: このビットを設定します。 HS_SPI はシリアル転送を始めるとき、このビットを"0"にリセットします。 このビットが既に"1"に設定されているときに、再度"1"を書き込むことは実行中のシリアル転送があったとしても何の影響も与えません。

47.4.14 HS_SPI ダイレクトモードストップレジスタ (HSSPIn_DMSTOP)

HS_SPI ダイレクトモードストップレジスタは、ソフトウェアフロー制御モード(HSSPIn_DMCFG:SSDC=0)が選択されているときに、シリアル転送を停止するのに使用されます。
このレジスタは、HS_SPI モジュールがダイレクトモードでマスタとして動作しているときにのみ使用されます。
STOP ビットがいったん設定されると、ソフトウェアは(HS_SPI がスレーブ選択出力のアサート停止により)実行中のシリアル転送を停止するまで、このビットをクリアしないでください。

■ HS_SPI ダイレクトモードストップレジスタ (HSSPIn_DMSTOP)

図 47-15 HS_SPI ダイレクトモードストップレジスタのビット構成

HSSPIn_DMSTOP							
7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	STOP
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W
0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-14 HSSPIn_DMSTOP レジスタのビット説明

ビット名		説明
bit[7:1]	read0	
bit0	STOP	ストップビット : Stop Bit (STOP) このビットは HS_SPI がマスタとして動作し(HSSPIn_DMCFG:MST=1)、HSSPIn_DMCFG:SSDC=0 のときにのみ使用することができます。 このビットはスレーブ選択出力の停止のため使用されます。 0: HS_SPI はスレーブ選択出力を停止しない。 1: HS_SPI の状況によりスレーブ選択出力を停止する。 TX-Only モード時: 停止が設定され、TX-FIFO のすべての内容が転送される。 RX-Only モード時: 停止が設定され、実行中のシフトレジスタへの書込みが完了する。 TX-and-RX モード時: 停止が設定され、TX-FIFO のすべての内容が転送される。

47.4.15 HS_SPI ダイレクトモードペリフェラル選択レジスタ (HSSPIn_DMPSEL)

HS_SPI ダイレクトモードペリフェラル選択レジスタは、マスタモードでシリアル転送を開始するため、4 つの周辺スレーブ選択信号から 1 つの信号を選択するのに使用されます。
このレジスタは HS_SPI モジュールがダイレクトモード時のみ使用できます。

■ HS_SPI ダイレクトモードペリフェラル選択レジスタ (HSSPIn_DMPSEL)

図 47-16 HS_SPI ダイレクトモードペリフェラル選択レジスタのビット構成

HSSPIn_DMPSEL							
7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	PSEL[1]	PSEL[0]
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W
0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-15 HSSPIn_DMPSEL レジスタのビット説明

ビット名		説明
bit[7:2]	read0	
bit[1:0]	PSEL[1:0]	ペリフェラル選択 : Peripheral Select (PSEL[1:0]) これらのビットは、HS_SPI がマスタモード(HSSPIn_DMCFG:MST=1)である場合のみ使用されます。 PSEL ビットは、SSEL3～SSEL0 の 4 つのスレーブ選択出力信号のうちどの信号をシリアル転送に使用するかを決めます。 00: スレーブ選択信号 0 01: スレーブ選択信号 1 10: スレーブ選択信号 2 11: スレーブ選択信号 3

47.4.16 HS_SPI ダイレクトモード転送プロトコルレジスタ (HSSPIn_DMTRP)

HS_SPI ダイレクトモード転送プロトコルレジスタはシリアル転送のプロトコルを設定します。
このレジスタは HS_SPI モジュールがダイレクトモード時のみ使用できます。

■ HS_SPI ダイレクトモード転送プロトコルレジスタ (HSSPIn_DMTRP)

図 47-17 HS_SPI ダイレクトモード転送プロトコルレジスタのビット構成

HSSPIn_DMTRP							
7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	TRP[3]	TRP[2]	TRP[1]	TRP[0]
R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-16 HSSPIn_DMTRP レジスタのビット説明

ビット名		説明
bit[7:4]	read0	
bit[3:0]	TRP[3:0]	転送プロトコル : Transfer Protocol (TRP[3:0]) TRP[3:2] ビットは Rx-Only, Tx-Only または TX-and-RX のデュプレックス設定を示します。 TRP[1:0] ビットはレガシー, デュアル, クワッドのうちどのプロトコルが使われているかを示します。 0000: TX-and-RX (レガシーモード時) 0100: RX Only (レガシーモード時) 0101: RX Only (デュアルモード時) 0110: RX Only (クワッドモード時) 1000: TX Only (レガシーモード時) 1001: TX Only (デュアルモード時) 1010: TX Only (クワッドモード時) 上記以外のビット構成は予約で、デフォルトはレガシーモードの TX-and-RX です。

47.4.17 HS_SPI ダイレクトモードバイトカウント制御レジスタ (HSSPIn_DMBCC)

HS_SPI ダイレクトモードバイトカウント制御レジスタは、フロー制御のバイトカウンタモードが選択されている(HSSPIn_DMCFG:SSDC=1)とき、シリアル転送で転送するバイト数を設定します。
このレジスタは、HS_SPI モジュールがダイレクトモード動作で、マスタとして機能するよう設定されている場合にのみ使用することができます。

■ HS_SPI ダイレクトモードバイトカウント制御レジスタ (HSSPIn_DMBCC)

図 47-18 HS_SPI ダイレクトモードバイトカウント制御レジスタのビット構成

HSSPIn_DMBCC															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BCC[15]	BCC[14]	BCC[13]	BCC[12]	BCC[11]	BCC[10]	BCC[9]	BCC[8]	BCC[7]	BCC[6]	BCC[5]	BCC[4]	BCC[3]	BCC[2]	BCC[1]	BCC[0]
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-17 HSSPIn_DMBCC レジスタのビット説明

ビット名		説明
bit[15:0]	BCC[15:0]	バイトカウント制御 : Byte Count Control (BCC[15:0]) これらのビットは、ダイレクトモードで HSSPIn_DMCFG:SSDC=1 のときに HS_SPI がマスタとして動作する(HSSPIn_DMCFG:MST=1)場合に使用します。BCC ビットに送信, 受信, または送受信されるバイト数を設定します。 バイト数は FIFO 幅の単位にしなければなりません。 (例 : HSSPIn_FIFOCFG.FWIDTH[1:0] = 10 のとき、BCC の値は 0x0000, 0x0003, 0x0006, 0x0009, 0x000C, 0x000F, …。) BCC の値は転送の開始時にダウンカウンタにロードされ、1 バイトがシリアルに転送されるごとにカウンタはデクリメントされます。HS_SPI はダウンカウンタの値がゼロになるとトランザクションを完了し、スレーブ選択のアサートを停止します。

47.4.18 HS_SPI ダイレクトモードバイトカウントステータスレジスタ (HSSPIn_DMBCS)

HS_SPI ダイレクトモードバイトカウントレジスタは読出し専用のレジスタで、実行中のシリアル転送で引き続き送信する必要のある残りのバイト数を確認するために使用します。
このレジスタは、HS_SPI モジュールがダイレクトモード動作でマスタとして設定され、フロー制御としてバイトカウンタモードが選択されている場合のみ有効です。

■ HS_SPI ダイレクトモードバイトカウントステータスレジスタ (HSSPIn_DMBCS)

図 47-19 HS_SPI ダイレクトモードバイトカウントステータスレジスタのビット構成

HSSPIn_DMBCS															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BCS[15]	BCS[14]	BCS[13]	BCS[12]	BCS[11]	BCS[10]	BCS[9]	BCS[8]	BCS[7]	BCS[6]	BCS[5]	BCS[4]	BCS[3]	BCS[2]	BCS[1]	BCS[0]
R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-18 HSSPIn_DMBCS レジスタのビット説明

ビット名		説明
bit[15:0]	BCS[15:0]	バイトカウント状態 : Byte Count Status (BCS[15:0]) 本ビットは読出し専用で、ダイレクトモードでマスタとして動作 (HSSPIn_DMCFG:MST=1) し、HSSPIn_DMCFG: SSDC=1 の場合のみ有効です。 BCS ビットは実行中のシリアル転送でまだシリアルに送信，受信，または送受信されていないバイト数を示します。

47.4.19 HS_SPI ダイレクトモードステータスレジスタ (HSSPIn_DMASTATUS)

HS_SPI ダイレクトモードステータスレジスタは、TX/RX パスがアクティブ状態かアイドル状態か、TX/RX FIFO のフィルレベルの現状値などのステータスビットを含みます。
このレジスタは HS_SPI モジュールがダイレクトモードに設定されているときにのみ使用されます。

■ HS_SPI ダイレクトモードステータスレジスタ (HSSPIn_DMSTATUS)

図 47-20 HS_SPI ダイレクトモードステータスレジスタのビット構成

HSSPIn_DMSTATUS																													
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	TXFLEVEL[4]	TXFLEVEL[3]	TXFLEVEL[2]	TXFLEVEL[1]	TXFLEVEL[0]	Read0	Read0	Read0	RXFLEVEL[4]	RXFLEVEL[3]	RXFLEVEL[2]	RXFLEVEL[1]	RXFLEVEL[0]	Read0	Read0	Read0	Read0	Read0	Read0	TXACTIVE	RXACTIVE
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R,WX	R,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-19 HSSPIn_DMSTATUS レジスタのビット説明

ビット名		説明
bit[31:24]	read0	
bit[23:21]	read0	
bit[20:16]	TXFLEVEL[4:0]	TX-FIFO の現フィルレベル : Current fill level of TX-FIFO (TXFLEVEL[4:0]) これらのビットは TX-FIFO の現在のフィルレベルを示します。
bit[15:13]	read0	
bit[12:8]	RXFLEVEL[4:0]	RX-FIFO の現フィルレベル : Current fill level of RX-FIFO (RXFLEVEL[4:0]) これらのビットは RX-FIFO の現在のフィルレベルを示します。
bit[7:2]	read0	
bit1	TXACTIVE	TX アクティブ : TX Active (TXACTIVE) 送信の状態を示します。 0: シリアル送信は実行されていません。 1: シリアル送信を実行中です。
bit0	RXACTIVE	RX アクティブ : RX Active (RXACTIVE) 受信の状態を示します。 0: シリアル受信は実行されていません。 1: シリアル受信を実行中です。

47.4.20 HS_SPI送信ビットカウントレジスタ (HSSPIn_TXBITCNT)

シリアル転送が停止, または終了すると、HS_SPI 転送ビットカウントレジスタには TX シフトレジスタからの未送信ビットの数が保持されます。
このレジスタは HS_SPI モジュールがダイレクトモードに設定されている場合にのみ使用されます。

■ HS_SPI 送信ビットカウントレジスタ (HSSPIn_TXBITCNT)

図 47-21 HS_SPI 送信ビットカウントレジスタのビット構成

HSSPIn_TXBITCNT							
7	6	5	4	3	2	1	0
Read0	Read0	TXBITCNT[5]	TXBITCNT[4]	TXBITCNT[3]	TXBITCNT[2]	TXBITCNT[1]	TXBITCNT[0]
R0,WX	R0,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-20 HSSPIn_TXBITCNT レジスタのビット説明

ビット名		説明
bit[7:6]	read0	
bit[5:0]	TXBITCNT[5:0]	TX ビットカウント : TX Bit Count (TXBITCNT[5:0]) これらのビットは TX シフトレジスタからの未送信ビット数を示します。 0: 送信するビットはありません。 1: 1 ビットが未送信です。 ... 31: 31 ビットが未送信です。 32: 32 ビットが未送信です。 HSSPIn_TXBITCNT レジスタは、HS_SPI がマスタモードで転送が停止したとき、またはスレーブ選択解除割込みフラグ(HSSPIn_RXF:RSSRS)がセットされ転送が終了したときに更新されます。

47.4.21 HS_SPI 受信ビットカウントレジスタ (HSSPIn_RXBITCNT)

シリアル転送の終了後、HS_SPI 受信ビットカウントレジスタには RX シフトレジスタに受信されたビット数が保持されます。これらのビットは RX FIFO メモリに送り込まれていません。
このレジスタは、HS_SPI モジュールがスレーブモードに設定されているときにのみ使用されます。

■ HS_SPI 受信ビットカウントレジスタ (HSSPIn_RXBITCNT)

図 47-22 HS_SPI 受信ビットカウントレジスタのビット構成

HSSPIn_RXBITCNT							
7	6	5	4	3	2	1	0
Read0	Read0	RXBITCNT[5]	RXBITCNT[4]	RXBITCNT[3]	RXBITCNT[2]	RXBITCNT[1]	RXBITCNT[0]
R0,WX	R0,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-21 HSSPIn_RXBITCNT レジスタのビット説明

Bit name		説明
bit[7:6]	read0	
bit[5:0]	RXBITCNT[5:0]	RX ビットカウント: RX Bit Count (RXBITCNT[5:0]) これらのビットは RX シフトレジスタの中の有効ビット数を示します。 0: 有効ビットは存在しません。 1: 1 ビットが有効 ... 31: 31 ビットが有効 32: 32 ビットが有効 このレジスタはスレーブモード時のみ使用できます。 スレーブモードでは、スレーブ選択信号のアサート停止により転送が終了したとき、HSSPIn_RXSHIFT レジスタは受信データに更新されます。また、HSSPIn_RXBITCNT レジスタは HSSPIn_RXSHIFT レジスタの中の有効ビット数に更新されます。HSSPIn_RXSHIFT レジスタと HSSPIn_RXBITCNT レジスタを読み取ることにより、RX-FIFO メモリに送り込まれていない RX データを確認することができます。

47.4.22 HS_SPI RX シフトレジスタ (HSSPIn_RXSHIFT)

HS_SPI RX シフトレジスタは読出し専用のレジスタです。ダイレクトモードでのシリアル転送が停止，または終了したとき、シリアルインタフェース経由で HS_SPI が受け取ったデータでまだ RX-FIFO に送り込まれていないデータをこのレジスタに格納します。
このレジスタはダイレクトモードでのみ使用できます。

■ HS_SPI RX シフトレジスタ (HSSPIn_RXSHIFT)

図 47-23 HS_SPI RX シフトレジスタのビット構成

HSSPIn_RXSHIFT																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXSHIFT[31]	RXSHIFT[30]	RXSHIFT[29]	RXSHIFT[28]	RXSHIFT[27]	RXSHIFT[26]	RXSHIFT[25]	RXSHIFT[24]	RXSHIFT[23]	RXSHIFT[22]	RXSHIFT[21]	RXSHIFT[20]	RXSHIFT[19]	RXSHIFT[18]	RXSHIFT[17]	RXSHIFT[16]	RXSHIFT[15]	RXSHIFT[14]	RXSHIFT[13]	RXSHIFT[12]	RXSHIFT[11]	RXSHIFT[10]	RXSHIFT[9]	RXSHIFT[8]	RXSHIFT[7]	RXSHIFT[6]	RXSHIFT[5]	RXSHIFT[4]	RXSHIFT[3]	RXSHIFT[2]	RXSHIFT[1]	RXSHIFT[0]
R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Word

＜注意事項＞
このレジスタは必ずワードアクセスでアクセスしてください。バイト/ハーフワードよるアクセスは不定データを返します。

表 47-22 HSSPIn_RXSHIFT レジスタのビット説明

ビット名		説明
bit[31:0]	RXSHIFT[31:0]	RX シフトレジスタ : RX Shift Register (RXSHIFT[31:0]) このレジスタは、スレーブモードでのみ使用できます。 スレーブモードでは、スレーブ選択信号のアサート停止により転送が終了したとき、HSSPIn_RXSHIFT レジスタは受信データに更新されます。また、HSSPIn_RXBITCNT レジスタは HSSPIn_RXSHIFT レジスタの中の有効ビット数に更新されます。HSSPIn_RXSHIFT レジスタと HSSPIn_RXBITCNT レジスタを読み取ることにより、RX-FIFO メモリに送り込まれていない RX データを確認することができます。

47.4.23 HS_SPI TX-FIFO レジスタ (HSSPIn_TXFIFO0~15)

HS_SPI TX-FIFO レジスタはデータを TX-FIFO メモリに送り込むために使用されます。16 個のレジスタはすべて連続してレジスタマップに配置されています。16 個のレジスタの機能は同一です。
これらのレジスタへのアクセスは、32 ビットアクセスのみです。これらのレジスタへライトアクセスすると、TX-FIFO メモリには 33 ビットが送り込まれます。33 番目のビットは CTRL ビットであり HSSPIn_FIFOCFG:TXCTRL ビットと同じ値が書き込まれます。

■ HS_SPI TX-FIFO レジスタ 0 (HSSPIn_TXFIFO0)

図 47-24 HS_SPI TX-FIFO レジスタ 0 のビット構成

HSSPIn_TXFIFO0																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXDATA[31]	TXDATA[30]	TXDATA[29]	TXDATA[28]	TXDATA[27]	TXDATA[26]	TXDATA[25]	TXDATA[24]	TXDATA[23]	TXDATA[22]	TXDATA[21]	TXDATA[20]	TXDATA[19]	TXDATA[18]	TXDATA[17]	TXDATA[16]	TXDATA[15]	TXDATA[14]	TXDATA[13]	TXDATA[12]	TXDATA[11]	TXDATA[10]	TXDATA[9]	TXDATA[8]	TXDATA[7]	TXDATA[6]	TXDATA[5]	TXDATA[4]	TXDATA[3]	TXDATA[2]	TXDATA[1]	TXDATA[0]
R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W	R0,W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Word

＜注意事項＞
このレジスタは必ずワードアクセスでアクセスしてください。バイト/ハーフワードよるアクセスは不定データが TX-FIFO に入ります。

表 47-23 HSSPIn_TXFIFO0 レジスタのビット説明

ビット名		説明
bit[31:0]	TXDATA[31:0]	TX-FIFO レジスタ 0 : TX-FIFO Register 0 (TXDATA[31:0]) この 32 ビットレジスタへ書き込むと TX－FIFO メモリの次のロケーションにデータが書き込まれ、TX-FIFO ライトポインタを 1 つ進めます。 TX-FIFO ワードの 33 番目のビットである TXCTRL ビットには HSSPIn_FIFOCFG:TXCTRL ビットの値が書き込まれます。 TX-FIFO メモリの設定幅がどのようなものであれ、32 ビット幅のアクセスしか許可されません。TX-FIFO メモリの設定幅が 32 ビットに満たないとき、HS_SPI は設定幅を超えた最上位ビットを転送しません。例えば、FIFO メモリの設定幅が 8 ビットのとき、TXDATA[7:0]ビットのみが HS_SPI により転送され、TXDATA[31:08]ビットは転送されません。したがって、この場合、ソフトウェアはこれらの転送されない上位ビットの位置に転送が必要な有効データを書き込まないようにしなければなりません。 TX-FIFO メモリがフルのときにこのレジスタにライトアクセスすると、新しいデータを TX-FIFO に押し込み、TX-FIFO オーバランが発生します。TX-FIFO オーバランになる条件が発生したときは、シリアルラインを経由して転送されたデータの整合性は保証されません。このレジスタに書き込む前に、ソフトウェアはオーバランを避けるため、TX-FIFO メモリがフルでないことを確認しておくことが必要です。

47.4.24 HS_SPI RX-FIFO レジスタ (HSSPIn_RXFIFO0~15)

これらの HS_SPI RX-FIFO レジスタはデータを RX-FIFO メモリから取り出すために使用されます。16 個のレジスタはすべて連続してレジスタマップに配置されています。16 個のレジスタの機能は同一です。レジスタへのリードアクセスは 32 ビットアクセスのみ許可されます。これらのレジスタへのリードアクセスにより、RX-FIFO からデータを読み出しが可能です。

■ HS_SPI RX-FIFO レジスタ 0 (HSSPIn_RXFIFO0)

図 47-25 HS_SPI RX-FIFO レジスタ 0 のビット構成

HSSPIn_RXFIFO0																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDATA[31]	RXDATA[30]	RXDATA[29]	RXDATA[28]	RXDATA[27]	RXDATA[26]	RXDATA[25]	RXDATA[24]	RXDATA[23]	RXDATA[22]	RXDATA[21]	RXDATA[20]	RXDATA[19]	RXDATA[18]	RXDATA[17]	RXDATA[16]	RXDATA[15]	RXDATA[14]	RXDATA[13]	RXDATA[12]	RXDATA[11]	RXDATA[10]	RXDATA[9]	RXDATA[8]	RXDATA[7]	RXDATA[6]	RXDATA[5]	RXDATA[4]	RXDATA[3]	RXDATA[2]	RXDATA[1]	RXDATA[0]
R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX	R,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Word

表 47-24 HSSPIn_RXFIFO0 レジスタのビット説明

ビット名		説明
bit[31:0]	RXDATA[31:0]	RX-FIFO レジスタ 0 : RX-FIFO Register 0 (RXDATA[31:0]) このレジスタを読み出すと、RX-FIFO リードポインタが指示した RX-FIFO メモリのロケーションから 1 ワード分のデータが返されます。レジスタへのリードアクセス後、RX-FIFO リードポインタは増加し、その増加分は発生したリードサイクルに応じた値となります。 RX-FIFO メモリの設定幅がどのようなものであれ、32 ビット幅のリードアクセスしか許可されません。FIFO メモリの設定幅が 32 ビットに満たないとき、FIFO の記憶場所から読み出されなかった上位ビットは無効なデータとなります。例えば、FIFO メモリの設定幅が 8 ビットのとき、RXDATA[7:0]ビットのみが有効であり、RXDATA[31:08]ビットは論理 0 を返します。したがって、この場合ソフトウェアはこれらの読み出されない上位ビットからのデータを使わないようにしなければなりません。 RX-FIFO メモリがエンプティのときにこのレジスタにリードアクセスすると、RX-FIFO から無効なデータが取り出されます。リードサイクルがオンチップバスマスタにより起動されていた場合は RX-FIFO アンダラン割込みイベント (HSSPIn_RXF:RFUS)が発生します。

47.4.25 HS_SPI FIFO 設定レジスタ (HSSPIn_FIFOCFG)

HS_SPI FIFO 設定レジスタは TX-FIFO メモリと RX-FIFO メモリの動作を設定します。FIFO メモリのしきい値と幅を設定することができます。
ソフトウェアはまたこのレジスタの TXFLSH と RXFLSH ビットで FIFO メモリの内容を破棄することができます。

■ HS_SPI FIFO 設定レジスタ (HSSPIn_FIFOCFG)

図 47-26 HS_SPI FIFO 設定レジスタのビット構成

HSSPIn_FIFOCFG																													
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	TXFLSH	RXFLSH	TXCTRL	FWIDTH[1]	FWIDTH[0]	TXFTH[3]	TXFTH[2]	TXFTH[1]	TXFTH[0]	RXFTH[3]	RXFTH[2]	RXFTH[1]	RXFTH[0]
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,W	R0,W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1

Access : Byte, Half-word, Word

表 47-25 HSSPIn_FIFOCFG レジスタのビット説明

ビット名		説明
bit[31:16]	read0	
bit[15:13]	read0	
bit12	TXFLSH	TX-FIFO フラッシュ : TX-FIFO Flush (TXFLSH) 本ビットにより TX-FIFO メモリの内容を破棄します。 0: 無効。 1: TX-FIFOメモリ内容を破棄します。 読出し値は"0"です。
bit11	RXFLSH	RX-FIFO フラッシュ : RX-FIFO Flush (RXFLSH) 本ビットにより RX-FIFO メモリの内容を破棄します。 0: 無効。 1: RX-FIFO メモリ内容を破棄します。 読出し値は"0"です。

ビット名		説明
bit10	TXCTRL	TX-FIFO に書き込むための TXCTRL ビット : TXCTRL bit to be written to TX-FIFO (TXCTRL) HSSPIn_TXFIFO0~15 レジスタへの書込みを行うとき、TX-FIFO ワードの 33 番目のビット(TXCTRL ビット)はこのビットの値となります。 HSSPIn_DMTRP:TRP ビットが以下のモードのいずれかにプログラムされている場合は、HSSPIn_FIFOCFG:TXCTRL ビットを設定してください。 a) TX Only (デュアルモードのとき) b) TX Only (クワッドモードのとき) HSSPIn_DMTRP:TRP ビットが上記以外のモードのときは、TXCTRL ビットを " 0"に設定してください。 HSSPIn_TXFIFO0~15 レジスタに書き込む前に TX-FIFO 内の送信データに対して TXCTRL ビットをセットするか否かを判断し、本ビットを設定する必要があります。
bit[9:8]	FWIDTH[1:0]	FIFO 幅 : FIFO Width (FWIDTH[1:0]) これらのビットは FIFO 幅を示します。FIFO の設定幅に応じて SPI コアのシフトレジスタの使用可能サイズも変わります。 00: TX-FIFO, RX-FIFO およびシフトレジスタは 8 ビット幅。 01: TX-FIFO, RX-FIFO およびシフトレジスタは 16 ビット幅。 10: TX-FIFO, RX-FIFO およびシフトレジスタは 24 ビット幅。 11: TX-FIFO, RX-FIFO およびシフトレジスタは 32 ビット幅。
bit[7:4]	TXFTH[3:0]	TX-FIFO しきい値 : TX-FIFO Threshold Level (TXFTH[3:0]) これらのビットは TX-FIFO のしきい値レベルを設定します。
bit[3:0]	RXFTH[3:0]	RX-FIFO しきい値 : RX-FIFO Threshold Level (RXFTH[3:0]) これらのビットは RX-FIFO のしきい値レベルを設定します。

47.4.26 HS_SPI コマンドシーケンサ設定レジスタ (HSSPIn_CSCFG)

HS_SPI コマンドシーケンサ設定レジスタは HS_SPI モジュールのコマンドシーケンサを設定します。
コマンドシーケンサモードを有効にする前に、本レジスタの設定が必要です。転送プロトコルの属性，メモリ書込みの許可・禁止, HS_SPI に接続されるメモリデバイスのサイズなどを本レジスタで設定できます。

■ HS_SPI コマンドシーケンサ設定レジスタ (HSSPIn_CSCFG)

図 47-27 HS_SPI コマンドシーケンサ設定レジスタのビット構成

HSSPIn_CSCFG																													
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	MSEL[3]	MSEL[2]	MSEL[1]	MSEL[0]	Read0	Read0	Read0	Read0	SSEL3EN	SSEL2EN	SSEL1EN	SSEL0EN	Read0	Read0	Read0	Read0	Read0	MBM[1]	MBM[0]	SRAM
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-26 HSSPIn_CSCFG コマンドシーケンサ設定レジスタのビット説明

ビット名		説明
bit[31:24]	read0	
bit[23:20]	read0	
bit[19:16]	MSEL[3:0]	メモリデバイス選択ビット : Memory Device Selection bits (MSEL[3:0]) これらのビットは各スレーブ選択信号に関連付けられるオンチップバスアドレス空間の範囲を示します。また、選択された各デバイスのメモリバンクのサイズも示します。 これらのビットはコマンドシーケンサにより次の用途に使用されます。 (a) 4 つのスレーブ選択出力信号のどれをメモリ配置されたシリアル転送用にアサートするかを選択します。 (b) 選択されたメモリデバイスの各メモリバンクのサイズを選択する。 詳細は「47.5.3. コマンドシーケンサモード」を参照してください。
bit[15:12]	read0	
bit11	SSEL3EN	スレーブ選択 3 許可 : Slave Select 3 Enable (SSEL3EN) 0: スレーブ選択 3 に割り当てられたメモリ領域へのアクセスは非配置メモリアクセスフォルトとなります。 1: スレーブ選択 3 に割り当てられたシリアルメモリデバイスへのアクセスを許可します。
bit10	SSEL2EN	スレーブ選択 2 許可 : Slave Select 2 Enable (SSEL2EN) 0: スレーブ選択 2 に割り当てられたメモリ領域へのアクセスは非配置メモリアクセスフォルトとなります。 1: スレーブ選択 2 に割り当てられたシリアルメモリデバイスへのアクセスを許可します。

ビット名		説明
bit9	SSEL1EN	スレーブ選択 1 許可 ：Slave Select 1 Enable (SSEL1EN) 0: スレーブ選択 1 に割り当てられたメモリ領域へのアクセスは非配置メモリアクセスフォルトとなります。 1: スレーブ選択 1 に割り当てられたシリアルメモリデバイスへのアクセスを許可します。
bit8	SSEL0EN	スレーブ選択 0 許可 ：Slave Select 0 Enable (SSEL0EN) 0: スレーブ選択 0 に割り当てられたメモリ領域へのアクセスは非配置メモリアクセスフォルトとなります。 1: スレーブ選択 0 に割り当てられたシリアルメモリデバイスへのアクセスを許可します。
bit[7:3]	read0	
bit[2:1]	MBM[1:0]	マルチビットモード ：Multi Bit Mode (MBM[1:0]) 00 : コマンドシーケンサによるメモリデバイスアクセスはレガシーSPI プロトコルを使用します。リードデータは SDATA[0]でサンプリングされます。メモリ命令，アドレス，その他の制御情報は SDATA[1]から送信されます。ほかのシリアルデータラインの出力は抑止されます。 01 : コマンドシーケンサによるメモリデバイスアクセスは半二重デュアルビット SPI プロトコルを使用します。リードデータは SDATA[1:0]でサンプリングされます。メモリ命令，アドレス，その他の制御情報は SDATA[1:0]から送信されます。 10: コマンドシーケンサによるメモリデバイスアクセスはクワッドビット SPI プロトコルを使用します。リードデータは SDATA[3:0]でサンプリングされます。メモリ命令，アドレス，その他の制御情報は SDATA[3:0]から送信されます。 11: 予約
bit0	SRAM	シリアル SRAM/シリアルフラッシュメモリタイプ選択 ：Serial SRAM / Serial Flash Memory Type Select (SRAM) このビットはシリアル SRAM デバイスが HS_SPI 接続された場合に設定します。 0: シリアルメモリデバイスを接続します。デバイスへの書込みは抑止されます。 1: シリアル SRAM メモリを接続します。デバイスへの書込みが可能です。

47.4.27 HS_SPI コマンドシーケンサアイドルタイムレジスタ (HSSPIn_CSITIME)

HS_SPI コマンドシーケンサアイドルタイムレジスタは、コマンドシーケンサのアイドルタイムアウト時間を設定します。
コマンドシーケンサモードを許可する前に、ソフトウェアでこのタイムアウト時間の設定が必要です。

■ HS_SPI コマンドシーケンサアイドルタイムレジスタ (HSSPIn_CSITIME)

図 47-28 HS_SPI コマンドシーケンサアイドルタイムレジスタのビット構成

HSSPIn_CSITIME																													
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	ITIME[15]	ITIME[14]	ITIME[13]	ITIME[12]	ITIME[11]	ITIME[10]	ITIME[9]	ITIME[8]	ITIME[7]	ITIME[6]	ITIME[5]	ITIME[4]	ITIME[3]	ITIME[2]	ITIME[1]	ITIME[0]
R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Access : Byte, Half-word, Word

表 47-27 HSSPIn_CSITIME レジスタのビット説明

ビット名		説明
bit[31:16]	read0	
bit[15:0]	ITIME[15:0]	アイドル時間 : Idle Time (ITIME[15:0]) これらのビットはコマンドシーケンサモード(HSSPIn_MCTRL:CSEN=1)のときに使用されます。 HS_SPI は、シリアルインタフェースを通して、要求された回数のメモリの読出し/書込みを完了すると、スレーブ選択信号のアサート状態を保持します。アイドル期間内にシリアルメモリデバイスへのアクセスが検出されない場合は、HS_SPI がスレーブ選択信号のアサートを停止します。シリアルメモリアクセスのすべてがリードアクセス、またはすべてがライトアクセス、といった同じタイプのアクセスの場合や、アクセス領域が連続している、また、アクセスが既定のアイドル期間内に発生する場合のパフォーマンス向上に寄与します。アイドル時間の単位は、オンチップバスクロックの周期単位となります。

47.4.28 HS_SPI コマンドシーケンサアドレス拡張レジスタ (HSSPIn_CSAEXT)

HS_SPI コマンドシーケンサアドレス拡張レジスタは、コマンドシーケンサが扱うメモリ領域を拡張するために使用されます。
本レジスタにより、仮想的に 16GB までのシリアルメモリにアクセスすることができます。アドレス拡張を使用しない場合には、このレジスタのすべてのビットを”0”に設定しなければなりません。

■ HS_SPI コマンドシーケンサアドレス拡張レジスタ (HSSPIn_CSAEXT)

図 47-29 HS_SPI コマンドシーケンサアドレス拡張レジスタのビット構成

HSSPIn_CSAEXT																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AEXT[31]	AEXT[30]	AEXT[29]	AEXT[28]	AEXT[27]	AEXT[26]	AEXT[25]	AEXT[24]	AEXT[23]	AEXT[22]	AEXT[21]	AEXT[20]	AEXT[19]	AEXT[18]	AEXT17]	AEXT[16]	AEXT[15]	AEXT[14]	AEXT[13]	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0	Read0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-28 HSSPIn_CSAEXT レジスタのビット説明

ビット名		説明
bit[31:13]	AEXT[31:13]	アドレス拡張ビット : Address Extension Bits (AEXT[31:13]) これらのビットはコマンドシーケンサモード(HSSPIn_MCTRL:CSEN=1)において使用します。 HSSPIn_CSAEXT レジスタの [31:13]ビットが、コマンドシーケンサが生成するメモリアドレスの上位 19 ビットとなります。コマンドシーケンサモードのとき HS_SPI により生成された各スレーブ選択のメモリアドレスは、本ビット値とオンチップバスアドレスバスで決まるアドレス値を組み合わせた値となります。詳細は「47.5.3. コマンドシーケンサモード」を参照してください。 アドレス拡張を行わないときは、本ビットを 0x00000000 に設定してください。
bit[12:0]	read0	

47.4.29 HS_SPI リードコマンドシーケンスデータ/制御レジスタ 0~7 (HSSPIn_RDCSDC0~7)

HS_SPI リードコマンドシーケンスデータ/制御レジスタ 0~7 はメモリ読み込み操作を行うためにコマンドシーケンサが生成するシリアル転送処理フェーズを設定する 8 組のデータ/制御レジスタです。これらのレジスタはコマンドシーケンサモードでのみ使用されます。
ここでは、HSSPIn_RDCSDC0 レジスタを説明します。ほかのレジスタのビット構成も同じです。

■ HS_SPI リードコマンドシーケンスデータ / 制御レジスタ 0 (HSSPIn_RDCSDC0)

図 47-30 HS_SPI リードコマンドシーケンスデータ / 制御レジスタ 0 のビット構成

HSSPIn_RDCSDC0															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDCSDATA[7]	RDCSDATA[6]	RDCSDATA[5]	RDCSDATA[4]	RDCSDATA[3]	RDCSDATA[2]	RDCSDATA[1]	RDCSDATA[0]	Read0	Read0	Read0	Read0	Read0	Read0	Read0	DEC
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-29 HSSPIn_RDCSDC0 レジスタのビット説明

ビット名		説明
bit[15:8]	RDCSDATA[7:0]	メモリリードトランザクション用コマンドシーケンスデータ/制御バイト : Command Sequencer Data/Control Byte for Memory-Read transactions (RDCSDATA[7:0]) これらのビットは HSSPIn_RDCSDC0:DEC ビットが"0"のとき、シリアルインタフェースに送信される 8 ビットのデータとなります。 HSSPIn_RDCSDC0:DEC ビットが“1”のとき、RDCSDATA[2:0]ビットは以下の内容を意味します。 RDCSDATA[2:0]=000: シリアルメモリアドレスのアドレスビット [07:00]を送信します。 RDCSDATA[2:0]=001: シリアルメモリアドレスのアドレスビット[15:08] を送信します。 RDCSDATA[2:0]=010: シリアルメモリアドレスのアドレスビット [23:16]を送信します。 RDCSDATA[2:0]=011: シリアルメモリアドレスのアドレスビット[31:24]を送信します。 RDCSDATA[2:0]=100: Hi-Z バイト(SDATA[3:0]信号は 1 バイト時間の間トライステートです。) RDCSDATA[2:0]=101: Hi-Z ニブル(RDCSDATA[7:4]ビットの送信には SDATA 出力のトライステートが 4 ビット時間続きます。) RDCSDATA[2:0]=111: リストの最後です。 RDCSDATA[2:0]の上記以外の値は「予約」となります。使用禁止です。
bit[7:1]	read0	
bit0	DEC	デコード : Decode (DEC) 0: HSSPIn_RDCSDC0:RDCSDATA の値をそのまま送信します。 1: HSSPIn_RDCSDC0:RDCSDATA[2:0]を制御コードとして使用します。

47.4.30 HS_SPI ライトコマンドシーケンスデータ/制御レジスタ 0~7 (HSSPIn_WRCSDC0~7)

HS_SPI ライトコマンドシーケンスデータ/制御レジスタ 0~7 はメモリ書込み動作を行うためにコマンドシーケンサが生成するシリアル転送処理フェーズを設定する 8 組のデータ/制御レジスタです。これらのレジスタはコマンドシーケンサモードでのみ使用されます。
ここでは、HSSPIn_WRCSDC0 レジスタを説明します。ほかのレジスタのビット構成も同じです。

■ HS_SPI ライトコマンドシーケンスデータ / 制御レジスタ 0 (HSSPIn_WRCSDC0)

図 47-31 HS_SPI ライトコマンドシーケンスデータ/制御レジスタ 0 のビット構成

HSSPIn_WRCSDC0															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WRCSDATA[7]	WRCSDATA[6]	WRCSDATA[5]	WRCSDATA[4]	WRCSDATA[3]	WRCSDATA[2]	WRCSDATA[1]	WRCSDATA[0]	Read0	Read0	Read0	Read0	Read0	Read0	Read0	DEC
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R0,WX	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Access : Byte, Half-word, Word

表 47-30 HSSPIn_WRCSDC0 レジスタのビット説明

ビット名		説明
bit[15:8]	WRCSDATA[7:0]	メモリライトトランザクション用コマンドシーケンスデータ/制御バイト：Command Sequencer Data/Control Byte for Memory-Write transactions (WRCSDATA[7:0]) これらのビットは HSSPIn_WRCSDC0:DEC ビットが"0"のとき、シリアルインタフェースに送信される 8 ビットのデータとなります。 HSSPIn_WRCSDC0:DEC ビットが“1”のとき、WRCSDATA[2:0]ビットは以下の内容を意味します。 WRCSDATA[2:0]=000: シリアルメモリアドレスのアドレスビット[07:00]を送信します。 WRCSDATA[2:0]=001: シリアルメモリアドレスのアドレスビット[15:08]を送信します。 WRCSDATA[2:0]=010: シリアルメモリアドレスのアドレスビット[23:16]を送信します。 WRCSDATA[2:0]=011: シリアルメモリアドレスのアドレスビット[31:24]を送信します。 WRCSDATA[2:0]=100: Hi-Z バイト(SDATA[3:0]信号は 1 バイト時間の間トライステートです。) WRCSDATA[2:0]=101: Hi-Z ニブル(WRCSDATA[7:4]ビットの送信には SDATA 出力のトライステートが 4 ビット時間続きます。) WRCSDATA[2:0]=111: リストの最後です。 WRCSDATA[2:0]の上記以外の値は「予約」となります。使用禁止です。
bit[7:1]	read0	
bit0	DEC	デコード：Decode (DEC) 0: HSSPIn_WRCSDC0:WRCSDATA の値をそのまま送信します。 1: HSSPIn_WRCSDC0:WRCSDATA[2:0]を制御コードとして使用します。

47.4.31 HS_SPI アドレススワップ制御レジスタ (HSSSWPR)

HS_SPI アドレススワップ制御レジスタは、オンチップバスマスタから HS_SPI メモリ領域と CSR 領域へのバスアクセスのためのアドレスとエンディアン変換を制御します。
このレジスタは HS_SPI メモリ領域と CSR 領域に対してのみ有効です。

■ HS_SPI アドレススワップ制御レジスタ (HSSSWPR)

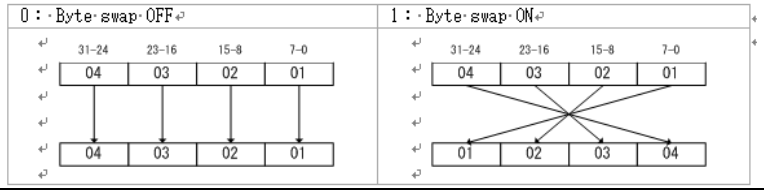
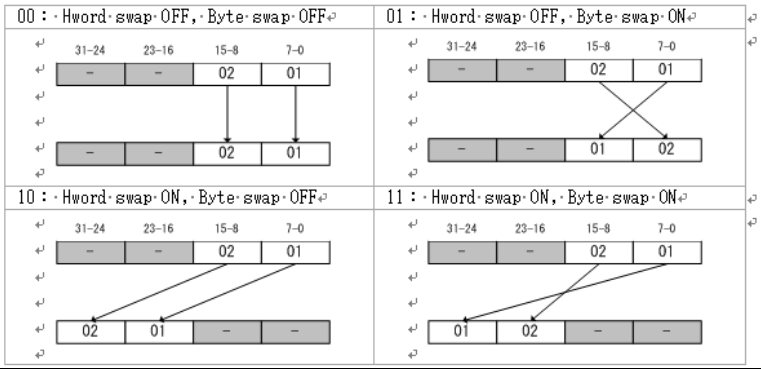
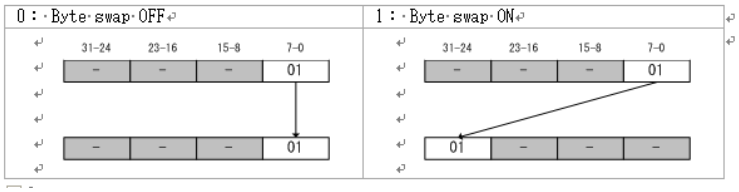
図 47-32 HS_SPI アドレススワップ制御レジスタのビット構成

HSSSWPR							
31	30	29	28	27	26	25	24
予約	予約	予約	ADSWP	WSWP	HSWP[1]	HSWP[0]	BSWP
R0,WX	R0,WX	R0,WX	R/W	R/W	R/W	R/W	R/W
0	0	0	0	1	1	1	1

Access : Byte, Half-word, Word

表 47-31 HSSSWPR レジスタのビット説明

ビット名		機能
bit[7:5]	予約	このレジスタへの書込みは、無効です。 読出し時は、このレジスタは常に"0"を返します。
bit4	ADSWP	アドレス入換え : Address Swap このビットが"1"に設定されているとき、オンチップバスマスタからのアクセスアドレスは下記のように変換されます。 ワードアクセス: 変換なし [例] 0x2000_0000 → 0x2000_0000 ハーフワードアクセス: アクセスアドレスの下位(LSB)の 2 番目のビットをトグルします。 [例] 0x2000_0004 → 0x2000_0006 バイトアクセス: アクセスアドレスの下位(LSB)の 1 番目と 2 番目のビットをトグルします。 [例] 0x2000_0008 → 0x2000_000B このビットが"0"に設定されているとき、すべてのアクセスアドレスはそのまま HS_SPI メモリと CSR 領域へ出力されます。

<table>
<tr><th colspan="2">ビット名</th><th>機能</th></tr>
<tr><td>bit3</td><td>Word Swap
(WSWP)</td><td>このビットはオンチップバスマスタから HS_SPI メモリと CSR 領域へのワードアクセスのビット部変換を有効にします。
<table>
<tr><th>WSWP</th><th>ワードアクセススワップ制御</th></tr>
<tr><td>0</td><td>Byte swap OFF</td></tr>
<tr><td>1</td><td>Byte swap ON</td></tr>
</table>

</td></tr>
<tr><td>bit[2:1]</td><td>Half-word Swap
(HSWP)</td><td>このビットはオンチップバスマスタから HS_SPI メモリと CSR 領域へのハーフワードアクセスのビットフィールド変換を有効にします。
メモリと CSR 領域へのハーフワードアクセスを行う場合、ソフトウェアはこのビットを適切に設定することが必要です。
<table>
<tr><th>HSWP</th><th>ハーフワードアクセススワップ制御</th></tr>
<tr><td>00</td><td>Hword swap OFF, Byte swap OFF</td></tr>
<tr><td>01</td><td>Hword swap OFF, Byte swap ON</td></tr>
<tr><td>10</td><td>Hword swap ON, Byte swap OFF</td></tr>
<tr><td>11</td><td>Hword swap ON, Byte swap ON</td></tr>
</table>

</td></tr>
<tr><td>bit0</td><td>Byte Swap
(BSWP)</td><td>このビットはオンチップバスマスタから HS_SPI メモリと CSR 領域へのバイトアクセスのビットフィールド変換を有効にします。
メモリと CSR 領域へのバイトアクセスを行う場合、ソフトウェアはこのビットを適切に設定することが必要です。
<table>
<tr><th>BSWP</th><th>バイトアクセススワップ制御</th></tr>
<tr><td>0</td><td>Byte swap OFF</td></tr>
<tr><td>1</td><td>Byte swap ON</td></tr>
</table>

</td></tr>
</table>

47.5 機能説明

HS_SPI の動作について説明します。

47.5.1. HS_SPI モードの動作と設定

47.5.2. ダイレクトモード

47.5.3. コマンドシーケンサモード

47.5.1 HS_SPI モードの動作と設定

HS_SPI モードの動作と設定について示します。

47.5.1.1. 動作

47.5.1.2. クロッキングモード

47.5.1.3. タイミング補正クロック

47.5.1.4.SPI プロトコル

47.5.1.5. シフト方向

47.5.1.6. シリアルクロック周波数

47.5.1.1 動作モード

動作モードについて示します。

HS_SPI はダイレクトモードとコマンドシーケンサモードの 2 つの動作モードを設定できます。

ダイレクトモードでは、CPU コアは送信するデータを TX-FIFO へ直接書き込むことができます。同様に、CPU コアはシリアルインタフェース経由で受け取ったデータを RX-FIFO とシフトレジスタから直接読み込むことができます。SPI コアはシリアルインタフェース経由でデータを直接 FIFO メモリへ、または FIFO メモリから転送します。ダイレクトモードでは、HS_SPI は CSR の設定により SPI マスタ、または SPI スレーブとして機能します。ダイレクトモードについては「47.5.2. ダイレクトモード」で説明します。

コマンドシーケンサモードでは、HS_SPI は SPI マスタとしてのみ機能します。このモードでは HS_SPI は外部のシリアルフラッシュまたはシリアル SRAM デバイスを CPU コアのアドレス空間に配置します。この方法で、最大 4 個のシリアルメモリデバイスを 4 組のスレーブ選択出力のそれぞれに配置することができます。CPU コアまたはほかのオンチップバスマスタが、これらのシリアルメモリデバイスにアクセスするためにオンチップバス転送を開始すると、HS_SPI は対応するメモリの読み込み、または書込み動作を行うためシリアル転送を開始します。HS_SPI が外部デバイスをアクセスするときまで、オンチップバス転送は動作しません。コマンドシーケンサモードについては「47.5.3. コマンドシーケンサモード」で説明します。

47.5.1.2 クロッキングモード

クロッキングモードについて示します。

HSSPIn_PCC0~3:CPOL, HSSPIn_PCC0~3:CPHA および HSSPIn_PCC0~3:ACES ビットの設定により、各周辺デバイスは最大 8 個のクロッキングモードを使用できます。これらのビットと HSSPIn_PCC0~3:RTM ビットにより、シリアル SPI クロックに対する HS_SPI のシリアル入力と出力のタイミングを決定します。下記のクロッキングモードの表で説明します。

表 47-32 クロッキングモード

モード	ACES (同一アクティブクロックエッジ)	CPOL (クロック極性)	CPHA (クロック位相)	説明
モード0	0	0	0	出力データは、まずシリアルクロックの立上りエッジの1/2サイクル前に駆動され、その後は立下りエッジで駆動されます。
				入力データは、シリアルクロックの立上りエッジでサンプリングされます。
モード1		0	1	出力データは、シリアルクロックの立上りエッジで駆動されます。
				入力データは、シリアルクロックの立下りエッジでサンプリングされます。
モード2		1	0	出力データは、まずシリアルクロックの立下りエッジの1/2サイクル前に駆動され、その後は立上りエッジで駆動されます。
				入力データは、シリアルクロックの立下りエッジでサンプリングされます。
モード3		1	1	出力データは、シリアルクロックの立下りエッジで駆動されます。
				入力データは、立上りエッジでサンプリングされます。
モード4(*)	1	0	0	出力データは、まずシリアルクロックの立上りエッジの1/2サイクル前に駆動され、その後は立下りエッジで駆動されます。
				入力データは、シリアルクロックの立下りエッジでサンプリングされます。

モード	ACES (同一アクティブクロックエッジ)	CPOL (クロック極性)	CPHA (クロック位相)	説明
モード 5(*)	1	0	1	出力データは、シリアルクロックの立上りエッジで駆動されます。
				入力データは、まずシリアルクロックの立下りエッジの1/2サイクル後にサンプリングされ、その後は立上りエッジでサンプリングされます。
モード 6(*)		1	0	出力データは、まずシリアルクロックの立下りエッジの1/2サイクル前に駆動され、その後は立上りエッジで駆動されます。
				入力データは、シリアルクロックの立上りエッジでサンプリングされます。
モード 7(*)		1	1	出力データは、シリアルクロックの立下りエッジで駆動されます。
				入力データは、まずシリアルクロックの立上りエッジの1/2サイクル後にサンプリングされ、その後は立下りエッジでサンプリングされます。

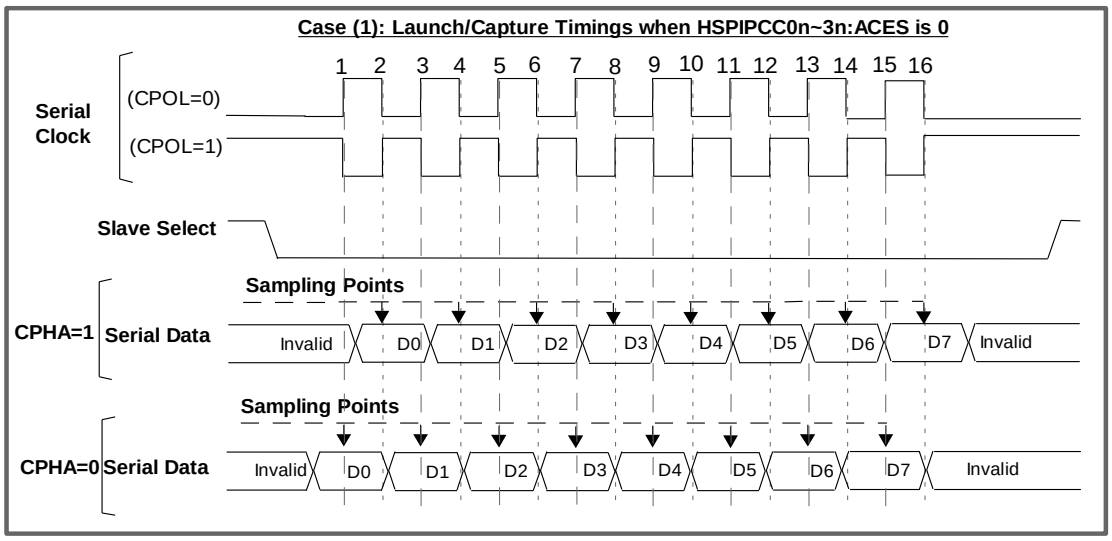
(*)レガシーTX-and-RX プロトコル(HSSPIn_DMTRP:TRP[3:0]=0000)では、モード 4, 5, 6, 7 はサポートされません。

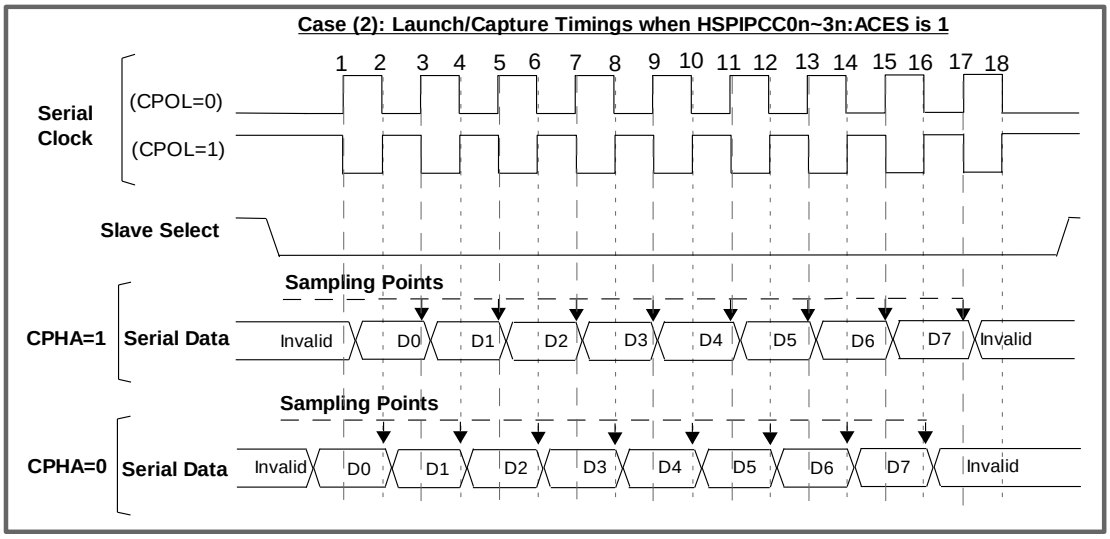
シリアルデータとシリアルクロックを示すタイミング波形を ACES, CPOL,CPHA ビットの組み合わせごとに以下の図に示します。

図 47-1 に示すように、一般的な HSSPIn_PCC0~3:ACES=0 の設定では、データ駆動とサンプリングのタイミングがクロック周期の半サイクルしか離れていないのに対し、HSSPIn_PCC0~3:ACES=1 の設定では、1 サイクル離れています。したがって、HSSPIn_PCC0~3:ACES ビットがセットされると、転送にはさらに 1 クロックサイクル分の時間が必要となります。受信モードでの転送開始時に、HS_SPI は最初のサンプリング時点でのデータサンプリングを省略し、次のサンプリング時点からデータサンプリングを開始します。この最初のサンプリング時点でのデータサンプリングの省略はタイミング補正モードでの正確なシリアルデータを得るために行われます。

HSSPIn_PCC0~3:ACES がセットされているときに、CSR を経由して送信と受信を同時に許可する場合は、受信開始の時点で挿入された追加のクロックサイクルの間に、HS_SPI マスタに接続されている SPI スレーブにもデータを送信してしまう、ということ注意してください。したがって、ACES を設定している最中の不要なデータの送信を避けるため、受信を許可している間、送信を禁止してください。

図 47-33 シリアルインタフェースクロックのクロッキングモード





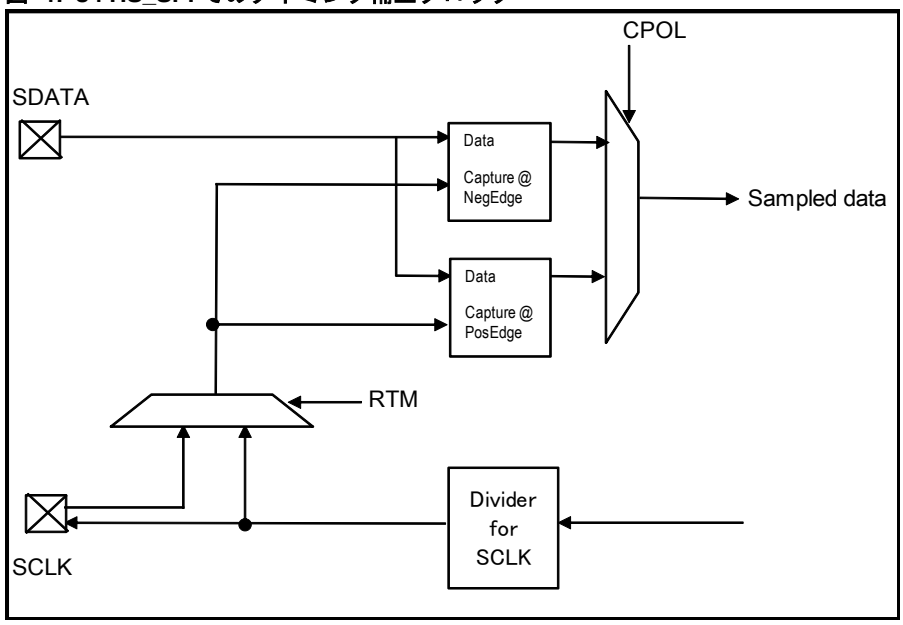
図に示すように、ACES="1"設定時には、リモートデバイスが正確にシリアルデータを取り込めるよう、追加の 1 サイクルが必要です。

47.5.1.3 タイミング補正クロック

タイミング補正クロックについて示します。

シリアルフラッシュデバイスにはHS_SPIでシリアルデータを取得するためのセットアップの余裕が非常に厳しいものもあります。HS_SPI がそのようなメモリデバイスに接続されているときは、データセットアップ違反がシリアルデータ入力を取得するレジスタで発生する可能性があります。そのようなシリアルメモリから有効なデータを取得するために、タイミング補正クロックモードを用いる必要があります。タイミング補正クロックモードは HSSPIn_PCC0~3:RTM ビットで設定できます。

図 47-34 HS_SPI でのタイミング補正クロック



47.5.1.4 SPI プロトコル

SPI プロトコルについて示します。

HS_SPI は、新しいデュアルビット SPI やクワッドビット SPI プロトコルとともに、レガシーSPI もサポートします。ダイレクトモードの場合は、HSSPIn_DMTRP:TRP[1:0]ビットで HS_SPI がレガシー，デュアルビット，またはクワッドビットのいずれを使用するのかを指定します。コマンドシーケンスモードの場合は、HSSPIn_CSCFG:MBM ビットで HS_SPI がレガシー，デュアルビット，またはクワッドビットのいずれを使用するのかを指定します。デュアルビットとクワッドビット SPI プロトコルは、新世代のシリアルフラッシュメモリデバイスとのインタフェースに使用します。

● レガシーSPI プロトコル

レガシーSPI プロトコルは全二重のプロトコルです。HS_SPI がレガシーSPI プロトコルでマスタモードに設定されているときは、データを単線(SDATA[1])経由で受け取ることができ、また同時に、データを単線(SDATA[0])経由で送信することができます。HS_SPI がレガシーSPI プロトコルでスレーブモードに設定されているときは、データを単線(SDATA[0])経由で受け取ることができ、また同時に、データを単線(SDATA[1])経由で送信することができます。レガシーSPI プロトコルが使用されている間は、HS_SPI が未使用のデータライン(SDATA[2:3])をトライステートにします。ダイレクトモードで HSSPIn_DMTRP:TRP ビットが"TX-and-RX in Legacy Mode", "TX-Only in Legacy Mode", または"RX-Only in Legacy Mode"のいずれかに設定されているときは、HS_SPI は全二重のレガシーSPI プロトコルを使用します。

● デュアルビットプロトコル

デュアルビット SPI プロトコルでは、半二重方式の 2 本のシリアルデータライン(SDATA[1:0])が使われます。データの送信と受信は同時に発生することはありません。デュアルビット SPI プロトコルが使われている間は、HS_SPI は未使用のデータライン(SDATA[2:3])をトライステートにします。

ダイレクトモードで HSSPIn_DMTRP:TRP ビットが"TX-Only in Dual Mode", または"RX-Only in Dual Mode"のいずれかに設定されているときは、デュアルビット SPI プロトコルが使われます。

● クワッドビットプロトコル

クワッドビット SPI プロトコルの場合は、4 本すべてのシリアルデータライン(SDATA[3:0])が、半二重方式で使用されます。データの送信と受信が同時に発生することはありません。

ダイレクトモードで HSSPIn_DMTRP:TRP ビットが"TX-Only in Quad Mode", または"RX-Only in Quad Mode"のいずれかに設定されているときは、クワッドビット SPI プロトコルが使われます。

47.5.1.5 シフト方向

シフト方向について示します。

HS_SPI ペリフェラルコミュニケーション設定レジスタ(HSSPIn_PCC0~3)には、シフトレジスタのシフト方向を指定するビット(SDIR)があります。

HSSPIn_PCC0~3:SDIR=0 のとき、シフトレジスタの最上位ビット(MSB)が初めに転送され、また、初めに受信したデータはシフトレジスタの最下位ビット(LSB)にシフトされます。つまり、シフトレジスタは左方向へシフトされます。HSSPIn_PCC0~3:SDIR=1 のとき、シフトレジスタの最下位ビット(LSB)が初めに転送され、また、初めに受信したデータはシフトレジスタの最上位ビット(MSB)にシフトされます。つまり、シフトレジスタは右方向へシフトされます。データレジスタへの読出し/書込みアクセスのときは、HSSPIn_PCC0~3:SDIR ビットの設定値に関係なく、常にデータの最下位ビットが bit0 にきます。

レガシー, デュアルビット, またはクワッドビット SPI プロトコルのどれかが使われているときに、CPOL=0, CPHA=0, SDIR=0, FWIDTH=0 のときのシフト方向と CPOL=0, CPHA=0, SDIR=1, FWIDTH=0 のときのシフト方向は　シフトレジスタのデータがシリアルデータラインへシフトするのか、またはシリアルデータラインからシフトするのかを示します。下図の波形は HSSPIn_PCC0~3:CPOL=0, HSSPIn_PCC0~3:CPHA=0, HSSPIn_FIFOCFG:FWIDTH=0 の条件を想定したものです。図は送信データが TX-FIFO メモリからシフトレジスタにロードされたことを示しています。しかしながら、送信データのソースは HSSPIn_RDCSDC0~7:RDCSDATA や HSSPIn_WRCSDC0~7:WRCSDATA といったほかのレジスタの場合もあり得ることに注意してください。

図 47-35 シフト方向 (CPOL=0, CPHA=0, SDIR=0, FWIDTH=0 の場合)

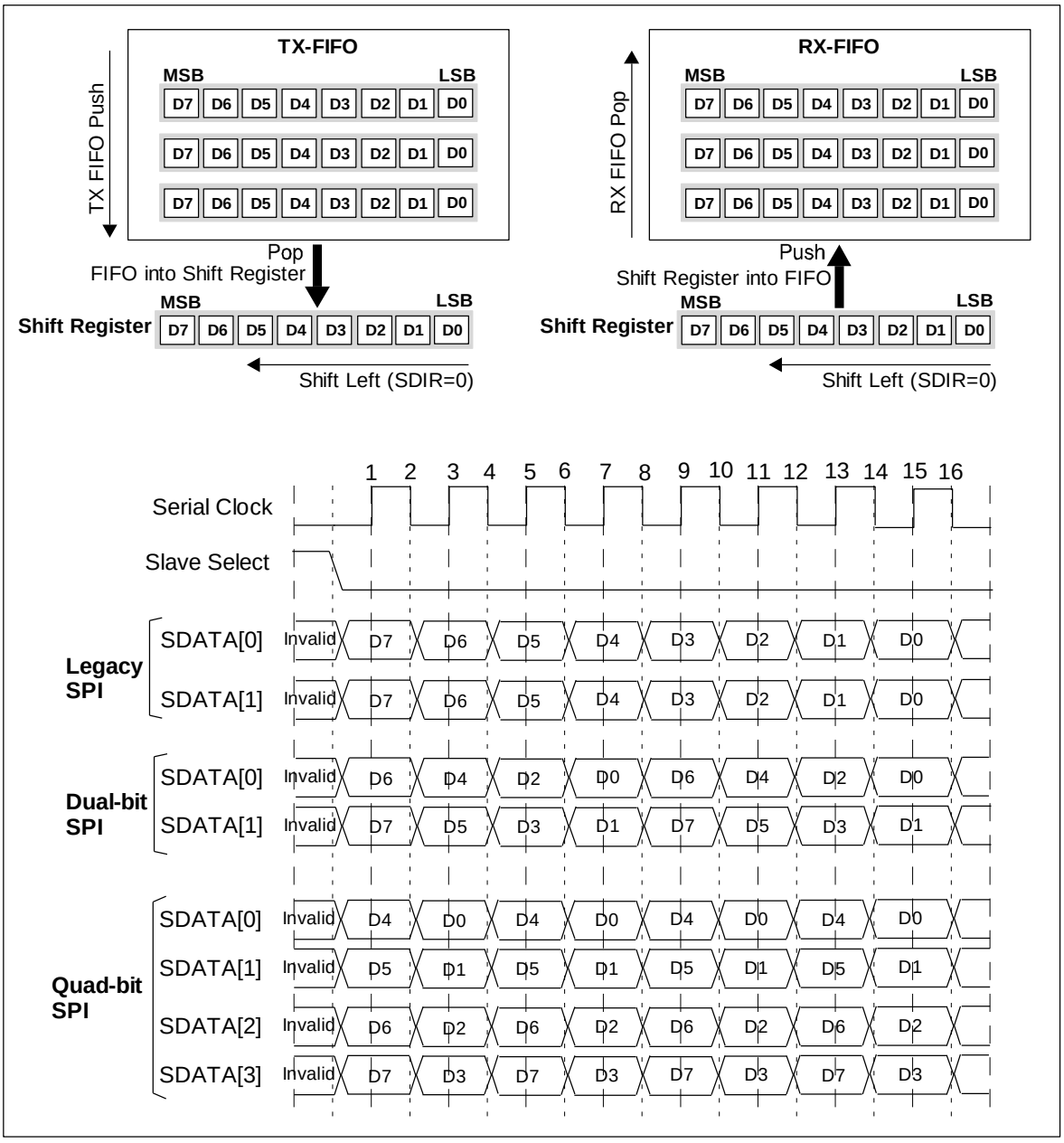
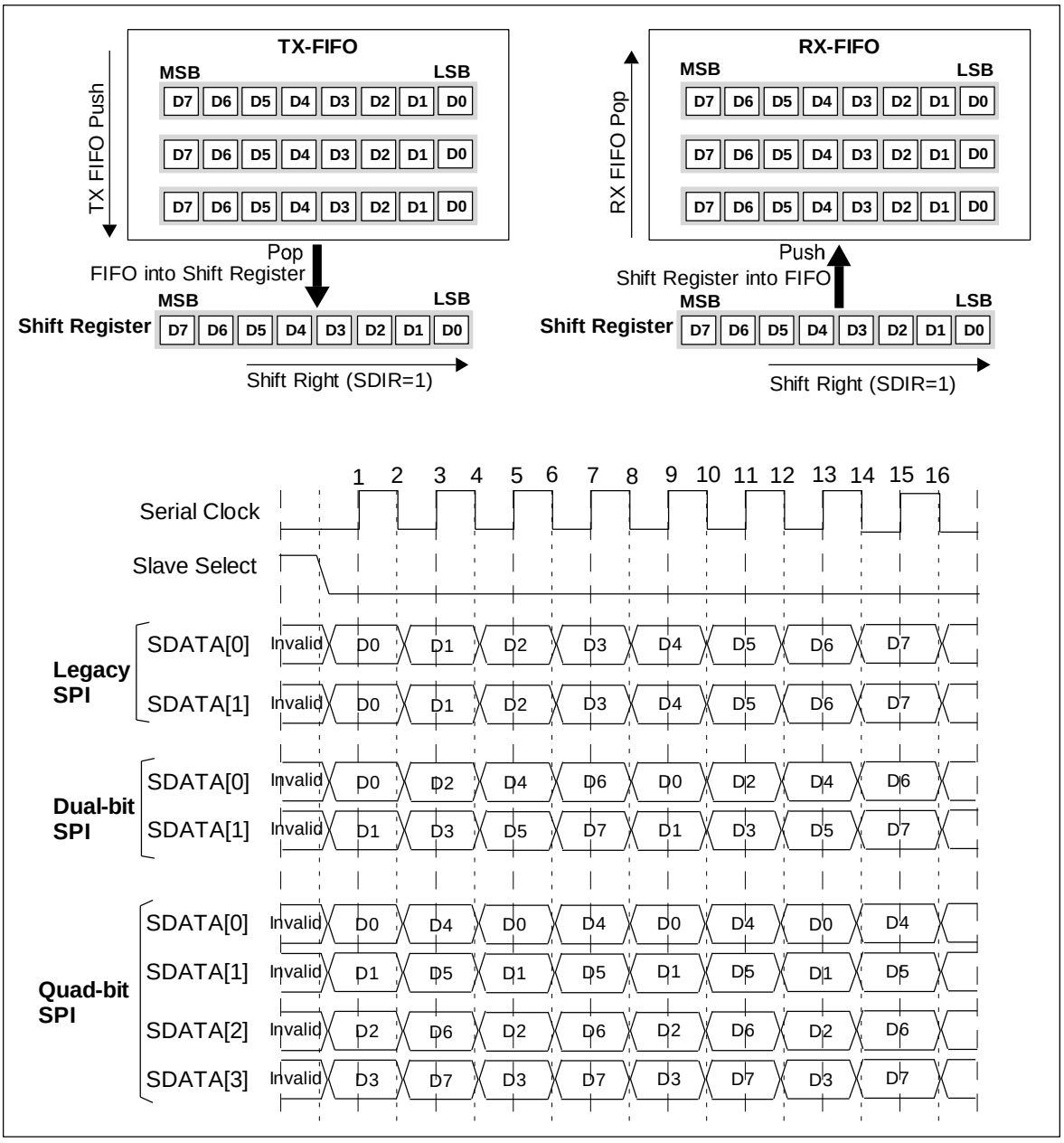


図 47-36 シフト方向 (CPOL=0, CPHA=0, SDIR=1, FWIDTH=0 の場合)



47.5.1.6 シリアルクロック周波数

シリアルクロック周波数について示します。

HS_SPI がマスタモードの場合、SCLK は内部でオンチップバスクロック(iHCLK)か周辺クロック(iPCLK)を分割することにより生成されます。HSSPIn_MCTRL:CDSS レジスタで 4 つのスレーブ選択のそれぞれのソースクロックの選択を行います。内部クロック分周器のクロック分周比は HSSPIn_PCC0~3 レジスタでプログラム可能です。

＜注意事項＞
コマンドシーケンスモードでは、HSSPIn_MCTRL:CDSS は"0"に設定されている必要があります。

HSSPIn_PCC0~3:CDRS でクロックの分周比を決めます。クロック分周器により生成されるクロックの周波数(Fo)は次の式であらわされます。

Fo = Fi / (2 × HSSPIn_PCC0~3:CDRS)

ここで、Fi は HSSPIn_MCTRL:CDSS で選択したソースクロックの周波数です。

このセクションの略語 :
Fsclk : SCLK の周波数
Fhclk : iHCLK の周波数

■ ダイレクトモード,マスタ動作時の SCLK 周波数範囲と SAFESYNC ビットの条件

下表は最大 Fsclk の範囲と HS_SPI がダイレクトモード, マスタ動作時に HSSPIn_PCC0~3: SAFESYNC ビットに"1"を設定することが必要となる条件を示しています。

表 47-33 SCLK 周波数範囲と SAFESYNC ビットの条件 (ダイレクトモード)

モード	シフトレジスタの幅	プロトコル	SAFESYNC を”1”に設定する条件	最大 SCLK 周波数
ダイレクトモード, マスタ	8 ビット	レガシー	SAFESYNC は不要	下記の 2 値のうち小さい方の値: ・(1/2) Fhclk ・16MHz
		デュアルビット	Fsclk = (1/2) Fhclk	
		クワッドビット	Fsclk ≧ (1/5) Fhclk	
	16 ビット	レガシー	SAFESYNC は不要	
		デュアルビット	SAFESYNC は不要	
		クワッドビット	Fsclk = (1/2) Fhclk	
	24 ビット	レガシー	SAFESYNC は不要	
		デュアルビット		
		クワッドビット		
	32 ビット	レガシー	SAFESYNC は不要	
		デュアルビット		
		クワッドビット		

SCLK のソースとして HCLK が選択されていない場合(CDSS=1)は、表に示した周波数レンジに対し余裕のある設定とするよう注意してください。

■ コマンドシーケンサモード時の SCLK 周波数範囲と SAFESYNC ビットの条件

下表は最大 Fsclk の範囲と HS_SPI がコマンドシーケンス動作時に HSSPIn_PCC0~3:SAFESYNC ビットに"1"を設定することが必要となる条件を示しています。

表 47-34 SCLK 周波数範囲と SAFESYNC ビットの条件(コマンドシーケンサモード)

動作モード	メモリマッピング転送のオンチップバス転送サイズ	プロトコル	SAFESYNC を 1 に設定する条件	最大 SCLK 周波数
コマンドシーケンサ	8 ビット	レガシー	SAFESYNC は不要	下記の 2 値のうち、小さい方の値: ・(1/4) Fhclk ・16MHz
		デュアルビット		
		クワッドビット	Fsclk ≧ (1/6) Fhclk	
	16 ビット	レガシー	SAFESYNC は不要	下記の 2 値のうち、小さい方の値: ・(1/4) Fhclk ・16MHz
		デュアルビット		
		クワッドビット	Fsclk≧ (1/6) Fhclk	
	32 ビット	レガシー	SAFESYNC は不要	下記の 2 値のうち、小さい方の値: ・(1/2) Fhclk ・16MHz
		デュアルビット	Fsclk = (1/2) Fhclk	
		クワッドビット	Fsclk≧ (1/4) Fhclk	

コマンドシーケンサモードが選択されている場合は、HSSPIn_MCTRL:CDSS ビットを"1"に設定することは禁止されていることに注意してください。

■ ダイレクトモード, スレーブ動作時の SCLK 周波数

下表は最大 Fsclk の範囲を示しています。スレーブモードで動作中に、シリアルクロックの周波数が図 47-35 の最大 SCLK 周波数欄に示されている値よりも大きいときは、データの内容は保証されません。したがって、スレーブモードのときは、SPI マスタが生成するシリアルクロック周波数が図 47-35 の最大 SCLK 周波数欄に示される最大値よりも低い値となるように設定する必要があります。

スレーブモードでは SAFESYNC ビットは必要ありません。

表 47-35 スレーブモード時の最大 SCLK 周波数

モード	シフトレジスタの幅	プロトコル	SAFESYNC 値	最大 SCLK 周波数
ダイレクトモード, スレーブ	8 ビット	レガシー	セーフ同期はスレーブモードでは適用しません。	下記の 2 値のうち、小さい方の値: ・(1/2) Fhclk ・8MHz
		デュアルビット		下記の 2 値のうち、小さい方の値: ・(1/4) Fhclk ・8MHz
		クワッドビット		下記の 2 値のうち、小さい方の値: ・(1/10) Fhclk ・8MHz
	16 ビット	レガシー		下記の 2 値のうち、小さい方の値: ・(1/2) Fhclk ・8MHz
		デュアルビット		
		クワッドビット		下記の 2 値のうち、小さい方の値: ・(1/4) Fhclk ・8MHz
	24 ビット	レガシー		下記の 2 値のうち、小さい方の値: ・(1/2) Fhclk ・8MHz
		デュアルビット		
		クワッドビット		
	32 ビット	レガシー		
		デュアルビット		
		クワッドビット		

SCLK 入力は HCLK に同期していないため、表に示された周波数レンジに対し余裕のある設定とするよう注意してください。

■ マスタモード時の SAFESYNC ビットの効果

SAFESYNC ビットによって、RX/TX レジスタと TX/RX-FIFO の間の正確なデータ転送のためのウェイトステートの挿入が有効になります。

HSSPIn_PCC0~3:SAFESYNC ビットがセットされると、HS_SPI マスタは内部でデータの同期をとっている間、実行中のシリアル転送を断続的に停止します。この停止はシリアルクロックの 3 周期分の時間続きます。シリアルインタフェースは以下の 3 つの条件が同時に成立する場合のみ、内部データのセーフ同期のため停止させられます。

- HS_SPI がマスタモード(HSSPIn_DMCFG:MST=1'b1, または HSSPIn_MCTRL:CSEN=1'b1)であること
- HSSPIn_PCC0～3:SAFESYNC ビットが 1 に設定されていること
- 「シフトレジスタの幅が 8 ビットでシリアルインタフェースがデュアルビットかクワッドビットに設定されていること」, または「シフトレジスタの幅が 16 ビットでシリアルインタフェースがクワッドビットモードに設定されていること」

このように、HSSPIn_PCC0~3:SAFESYNC ビットが“1”に設定されていると、HS_SPI の SPI コアは、そのときに使用しているシフトレジスタの幅により、データの転送単位ごとにセーフ同期を行うかどうかを決定します。単に HSSPIn_PCC0~3:SAFESYNC ビットを設定しただけでは、SPI コアが送信中のデータ転送単位ごとにセーフ

同期のためのウェイトステートを挿入するわけではありません。SPI コアは、シフトレジスタの幅と使用している SPI プロトコル（レガシー/デュアル/クワッド）に関係する特定の条件が満足された場合にのみ、ウェイトステートを追加します。これにより、セーフ同期による時間の損失はあるものの、シリアル転送の帯域幅を著しく損なうことがないよう保証します。

47.5.2 ダイレクトモード

ダイレクトモードについて示します。

ダイレクトモードでは、CPU コア(または DMA コントローラ)がシリアルインタフェース上のシリアル転送を直接制御します。ダイレクトモードへは、HSSPIn_MCTRL レジスタの CSEN ビットを使って遷移します。ダイレクトモードでは、HS_SPI はシリアルインタフェース経由で送受信するデータの一時保管に内部 FIFO を使用します。このダイレクトモードの動作を説明します。

47.5.2.1 内部 FIFO

内部 FIFO について説明します。

HS_SPI は内部に 2 つの FIFO を持ち、ここにデータを一時格納します。一方はデータの送信用、他方はデータの受信用となります。

HS_SPI でのシリアル転送の動作は、HSSPIn_DMTRP:TRP の設定により TX-Only, RX-Only, TX-and-RX のいずれかになります。この結果、HS_SPI が FIFO を 1 つだけ使うのか、両方使うのかが決まります。

HS_SPI が TX-Only の動作に設定されると、TX-FIFO が使用されます。HS_SPI が RX-Only の動作に設定されると、RX-FIFO が使用されます。HS_SPI が TX-and-RX 動作に設定されると、両方の FIFO が使用されます。

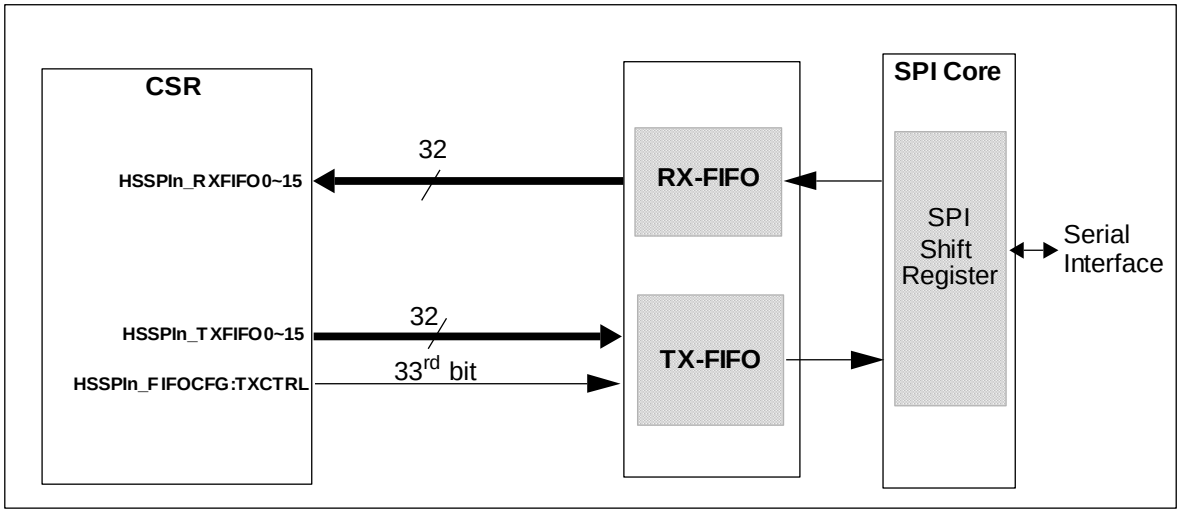
● FIFO サイズ

どちらの FIFO も 16 段、データ幅 32 ビットの構成です。ただし、ソフトウェアは HSSPIn_FIFOCFG: FWIDTH で TX-FIFO と RX-FIFO の有効データ幅を設定できます。

SPI コアのシフトレジスタは 32 ビット幅です。FIFO の幅が HSSPIn_FIFOCFG:FWIDTH フィールドで変更されると、シフトレジスタが使用する幅も対応して変更されます。

詳細は図 47-37 を参照してください。

図 47-37 ダイレクトモードの HS_SPI



32 ビットのデータ幅のほかに、TX-FIFO の各段には 33 番目の制御ビットがあります。TXCTRL ビットとよばれるこのビットにより、SPI コアが TX-FIFO のデータを送り出すのか、シリアルデータラインをトライステートにするのかを決めます。TXCTRL ビットが"1"に設定されると、HS_SPI は対応する TX-FIFO 段のデータのビット"0"をさらにデコードします。TXCTRL ビットと TX-FIFO データのビット"0"とのすべての組み合わせを以下

の「送信時のシリアルデータ出力のトライステート化」に示します。

表 47-36 送信時のトライステートシリアルデータ出力

TXCTRL	TX-FIFO データのビット0	説明
0	-	シリアルデータ出力ラインは対応するデータを送っている間、トライステートにはなりません。
1	0	シリアルデータ出力ラインは 1 バイト時間トライステートになります。対応する TX-FIFO 段のデータは送信されません。
1	1	HSSPIn_PCC0~3:SDIR ビットに設定されたシフト方向に係わらず、データ送信は以下の順序で発生します。 対応する TX-FIFO 段からデータビット[7:4]を送信します。 このデータの送信方向は HSSPIn_PCC0~3:SDIR ビットの設定により決まります。 SDATA 出力ラインは 4 ビット時間トライステートになります。

HSSPIn_DMBCC:BCC 機能や、表 5-2 で示す制限事項に対しては、TXCTRL ビットが設定されている TX-FIFO 段のデータは 1 バイト幅と考えてください。市販のクワッドビット SPI メモリの中には、インタフェースをとるために、ダミーサイクルの間やコマンドシーケンスのモードビットの下位ニブルの送信の間、シリアルデータ(SDATA)ラインをトライステートにする必要があるものがあります。ソフトウェアは、TX-FIFO のデータワードと TXCTRL ビットを使ってダイレクトモードでコマンドシーケンスを生成しインタフェースをとることで、このようなケースに対応可能です。

● FIFO アクセス

FIFO の設定幅に係わらず、HSSPIn_RXFIFO0~15 と HSSPIn_TXFIFO0~15 レジスタには 32 ビットワードアクセスのみが可能です。

CSR の HSSPIn_RXFIFO0~15 レジスタへ読出しアクセスを行うと、RX-FIFO からすぐに 1 ワードを読み出します。RX-FIFO 幅が 8 ビットのときは、HSSPIn_RXFIFO0～15 レジスタから読み出された上位 24 ビットは論理“0”となります。同様に、FIFO 幅が 16 ビットや 24 ビットのときは HSSPIn_RXFIFO0～15 レジスタから読み出された未使用のビットは論理“0”となります。

CSR の HSSPIn_TXFIFO0~15 レジスタへ書込みアクセスを行うと、1 ワードのデータと TXCTRL ビット(HSSPIn_FIFOCFG:TXCTRL 参照)を TX-FIFO に送り込みます。しかし、HS_SPI がデータをシリアルライン上で送信する際には、HSSPIn_TXFIFO0~15 レジスタから読み取ったデータの最下位ビットのみが送られます。送信される最下位ビットの数は TX-FIFO の設定幅で決まります。HS_SPI は送信されなかった上位ビットについては無視します。

● RX-Data のアクセス

HS_SPI が SDATA ライン上で受け取ったシリアルデータは、SPI コアのシフトレジスタでパラレル化された上で RX-FIFO に送り込まれます。

転送が完了、すなわち、スレーブ選択ラインのアサートが解除された場合や、マスタモードで転送が停止されると、HSSPIn_RXSHIFT レジスタはパラレル化されたデータに更新され、HSSPIn_RXBITCNT:RXBITCNT は HSSPIn_RXSHIFT レジスタで有効であったビット数に 更新されます。HSSPIn_TXF:TSSRS や HSSPIn_RXF:RSSRS 割込みフラグがセットされた場合、ソフトウェアは HSSPIn_RXSHIFT と HSSPIn_RXBITCNT:RXBITCNT を読取ることにより、RX-FIFO に送り込まれなかった RX データを確認することができます。

47.5.2.2 サービス要求

サービス要求について示します。

ダイレクトモードで動作しているとき、CPU コアへの割込み要求は、TX-FIFO と RX-FIFO が現在どれだけ埋まっているか(フィルレベル)と、そのしきい値の設定にしたがって起動されます。DMA コントローラをデータ転送に使った場合も同様です。CY91570 シリーズでは HS_SPI は DMA コントローラと接続が可能で、ダイレクトモードで動作中は、DMA コントローラと HS_SPI の TX-FIFO や RX-FIFO との間でデータのブロック転送を行えます。割込みフラグは実行中の SPI 転送が終了したときに設定されます。

● FIFO レベルに起因する割込み要求

2 つの FIFO のフィルレベルは HSSPIn_DMSTATUS:TXFLEVEL と HSSPIn_DMSTATUS:RXFLEVEL で参照可能です。FIFO のフィルレベルと CPU コアによって設定されたしきい値に基づいて、HS_SPI は割込み要求を発生します。

TX-FIFO のフィルレベル(HSSPIn_DMSTATUS:TXFLEVEL)が HSSPIn_FIFOCFG:TXFTH で設定された TX-FIFO しきい値以下ならば、「TX-FIFO フィルレベルしきい値以下」(HSSPIn_TXF:TFLETS)割込みフラグがセットされます。

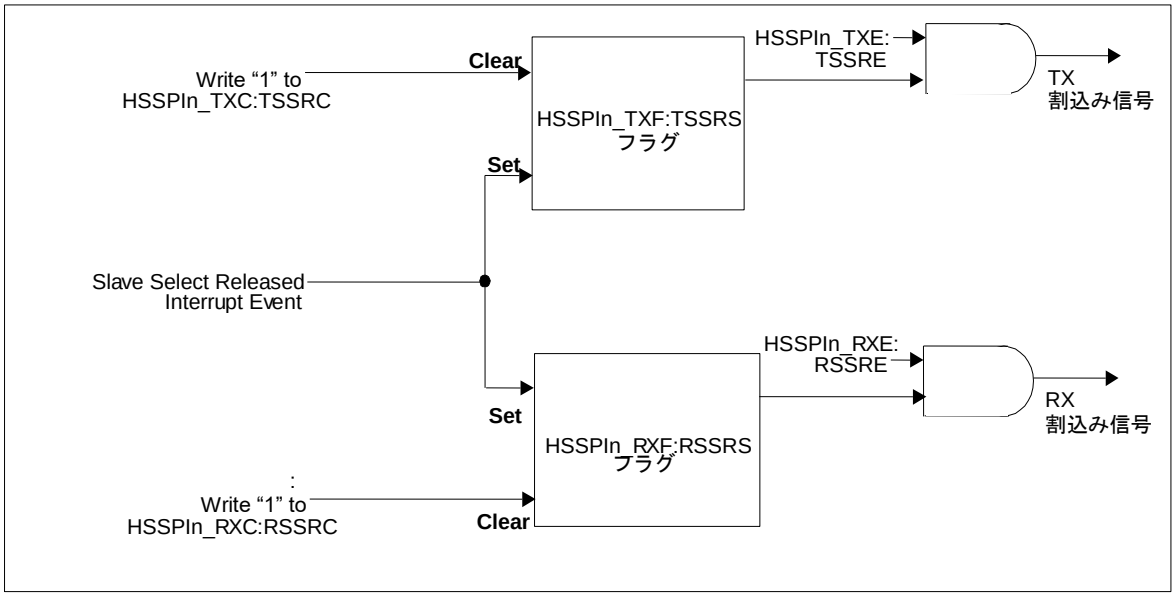
RX-FIFO のフィルレベル(HSSPIn_DMSTATUS:RXFLEVEL)が HSSPIn_FIFOCFG:RXFTH で設定された RX-FIFO しきい値を超えているときは、「RX-FIFO フィルレベルしきい値超」(HSSPIn_RXF:RFMTS)割込みフラグがセットされます。

HS_SPI が TX-Only 動作に設定されていると、RX-FIFO は使用されません。HS_SPI が RX-Only 動作に設定されていると、TX-FIFO は使用されません。

● 転送終了に起因する割込み要求

ダイレクトモードで動作中に、スレーブ選択ラインのアサートを解除すると、HS_SPI が割込みを起動します。スレーブ選択解除のイベントは 2 つの割込みフラグ、HSSPIn_TXF:TSSRS と HSSPIn_RXF:RSSRS のセット要因となります。これらのフラグはそれぞれ割込みクリアビットと割込み許可ビットを持っています。2 つの割込みフラグは別々の割込み信号として出力されます。図 47-38 にその様子を示します。

図 47-38 「スレーブ選択解除」割込みイベントの設定



47.5.2.3 マスタモードにおける SPI 転送

マスタモードにおける SPI 転送について示します。

HS_SPI は SPI マスタのとき、4 つの SPI スレーブ選択ラインのうち HSSPIn_DMPSEL:PSEL フィールドが選択した 1 つのラインへ転送を開始します。

● マスタモードにおける HS_SPI 通信属性

シリアルインタフェースでの通信には、シリアルインタフェースクロックの周波数，極性，位相，スレーブ選択ラインの極性、といったいくつかの属性があります。これらの通信属性は個々の SPI デバイスによって異なります。HS_SPI がダイレクトモードでマスタとして動作しているとき、それぞれ属性が異なる最大 4 つのスレーブとインタフェースをとることができます。
これらのデバイス固有の通信属性は CSR の HSSPIn_PCC0~3 レジスタで設定できます。

● シリアル転送の開始

HS_SPI が有効(HSSPIn_MCTRL:MEN=1)であり、ダイレクトモード(HSSPIn_MCTRL:CSEN=0)でマスタ(HSSPIn_DMCFG:MST=1)として機能しているとき、HSSPIn_DMSTART:START ビットが“1”にセットされると、HS_SPI によりシリアル転送が開始されます。

HSSPIn_DMTRP:TRP が転送許可の状態に設定されていて、かつ、HSSPIn_DMSTART: START ビットが"1"に設定された状態で TX-FIFO が空のとき、HS_SPI は TX-FIFO がソフトウェアにより書き込まれるまでシリアル転送を開始しません。

TX-FIFO がソフトウェアにより書き込まれるまで HS_SPI がシリアル転送の開始を待っている間に、HSSPIn_MCTRL:MEN ビットが"0"にリセットされたり、カウンタモードが使われた場合には、下記のようになります。

* HSSPIn_MCTRL:MEN ビットがソフトウェアにより"0"にリセットされると、次の転送は開始せず、HS_SPI モジュールの無効化が優先されます。
1. 転送長の制御にカウンタモードが使用されると、シリアル転送が HS_SPI により開始されるときに、HSSPIn_DMBCS レジスタは HSSPIn_DMBCC レジスタの値を上書きされます。それ以前の HSSPIn_DMBCS レジスタの値は"0"です。

HSSPIn_DMSTART:START ビットがいったん"1"にセットされると、ソフトウェアではリセットできませんので、注意が必要です。HS_SPI モジュールは、シリアル転送を開始後にこのビットをリセットします。START ビットが"1"にセットされているときに、"1"を書き込んでも何も変わりません。シリアル転送が既に開始されていて START ビットが"0"になっているときに、このビットに"1"を書き込んでも、進行中の転送に影響を与えません。進行中の転送が終了した後に新しいシリアル転送が始められます。

● TX-DATA の不足または RX-FIFO の空き不足による停止

標準 SPI プロトコルで規定されているように、スレーブ選択をアサートし続けたり、シリアルクロックを切断することにより、処理中の転送を停止することができます。TX-FIFO への書込みを待っている間、または RX-FIFO が読み出されるのを待っている間、HS_SPI は自動的にシリアルクロックを切断します。HS_SPI マスタが TX-Only, RX-Only, または TX-and-RX のいずれの動作を行っているかにより、HS_SPI のシリアルクロック切断, 転送停止には 3 つのケースがあります。

- TX-Only モード: TX-FIFO およびシフトレジスタが空のとき、シリアルクロックは切断される。
- RX-Only モード: RX-FIFO およびシフトレジスタがフルのとき、シリアルクロックは切断される。
- TX-and-RX モード: TX-FIFO およびシフトレジスタが空のとき、または RX-FIFO およびシフトレジスタがフルのとき、シリアルクロックは切断される。

FIFO リソース不足により HS_SPI がシリアルクロックを切断し処理中の転送を停止しても、対応するスレーブ選択ラインは、転送が終了していないことをスレーブに対し示すためアサートの状態を保ちます。FIFO リソースが使用可能状態になると、HS_SPI はシリアルクロックのトグリングを開始し、停止した転送を自動的に再開します。

● 転送長の制御

マスタモードでは、転送長(スレーブ選択ラインをアサート停止するまでの時間)の制御には 2 つの方法があります。

- カウンタモード
- ソフトウェアフロー制御モード

これらのモードは HSSPIn_DMCFG:SSDC ビットで選択できます。

カウンタモードでは、CPU コアは転送するバイトの数を HSSPIn_DMBCC:BCC に設定しなければなりません。HS_SPI 転送が始まるとき、HS_SPI は転送されるバイトの数を数え、HSSPIn_DMBCC:BCC に示されていたバイト数が転送された後でスレーブ選択信号を解除します。

ソフトウェアフロー制御モードでは、CPU コアは HSSPIn_DMSTOP:STOP ビットを使って転送長を制御します。HS_SPI マスタが X-Only, RX-Only, または TX-and-RX のいずれの動作を行っているかにより、スレーブ選択出力のアサート停止は以下のように制御されます。

- ☐ TX-Only モード: HSSPIn_DMSTOP:STOP ビットがセットされ、TX-FIFO のすべてのコンテンツが転送されると転送完了。
- ☐ RX-Only モード: HSSPIn_DMSTOP:STOP ビットがセットされ、受信したシリアルデータを再構築するために SPI コアで使用されるシフトレジスタにあるすべてのビットがシフトインすると転送完了。
- ☐ TX-and-RX モード: HSSPIn_DMSTOP:STOP ビットがセットされ、TX-FIFO のすべてのコンテンツが転送されると転送完了。

47.5.2.4 スレーブモードにおける SPI 転送

スレーブモードにおける SPI 転送について示します。

HS_SPI がスレーブとして設定(HSSPIn_DMCFG:MST=0)されている場合には、スレーブ選択入力(SSEL0)がアサートされると、HS_SPI は外部マスタが開始した転送に対応します。

● スレーブモードにおける HS_SPI 通信属性

シリアルインタフェースでの通信には、シリアルインタフェースクロックの極性，位相，スレーブ選択ラインの極性、といったいくつかの属性があります。これらの通信属性は個々の SPI デバイスによって異なります。HS_SPI がダイレクトモード動作でスレーブとして動作しているとき、通信属性値は CSR の HSSPIn_PCC0 レジスタで設定できます。

スレーブモードのときは、外部マスタがシリアルクロック(SCLK)を駆動します。

● シリアル転送中に FIFO リソースが不足する場合

HS_SPI は、スレーブとして機能しているときはシリアル転送を制御できません。したがって、ソフトウェアは、HSSPIn_TXF:TFLETS, HSSPIn_TXF:TFMTS, HSSPIn_RXF:RFLETS, HSSPIn_RXF:RFMTS といった割込みフラグを監視し、FIFO リソースの管理を行う必要があります。

HS_SPI が送信に設定(HSSPIn_DMTRP:TRP で TX-Only，または TX-and-RX に設定)されているときに、TX-FIFO が空になった場合は、HS_SPI は HSSPIn_TXF:TFES 割込みフラグをセットします。スレーブ選択がアサートされ、クロックが動作している限り、TX-FIFO が空になった後でも TX-FIFO からのデータ読み取りを続け(その結果、TX-FIFO はアンダランとなる)、シリアルデータライン上に TX-FIFO から得たデータを送り続けます。

HS_SPI が受信に設定(HSSPIn_DMTRP:TRP で RX-Only，または TX-and-RX に設定)されているときに、RX-FIFO がフルになると、HS_SPI は HSSPIn_RXF:RFFS 割込みフラグをセットし、RX-FIFO に SDATA ラインで受信したシリアルデータの重ね書きを続けます。

47.5.3　コマンドシーケンサモード

コマンドシーケンサモードではHS_SPIはSPIマスタとして動作し、外部シリアルメモリデバイスとのインタフェースを行います。4 本のスレーブ選択ラインのそれぞれで同一タイプのシリアルフラッシュやシリアル SRAM デバイスを割り当てます。CPU コアやオンチップバス上のバスマスタがメモリアクセスを行うと、HS_SPI が自動的にシリアルメモリの読出し/書込み命令に変換します。この HS_SPI のコマンドシーケンサモードの動作について説明します。

47.5.3.1　メモリの割当て

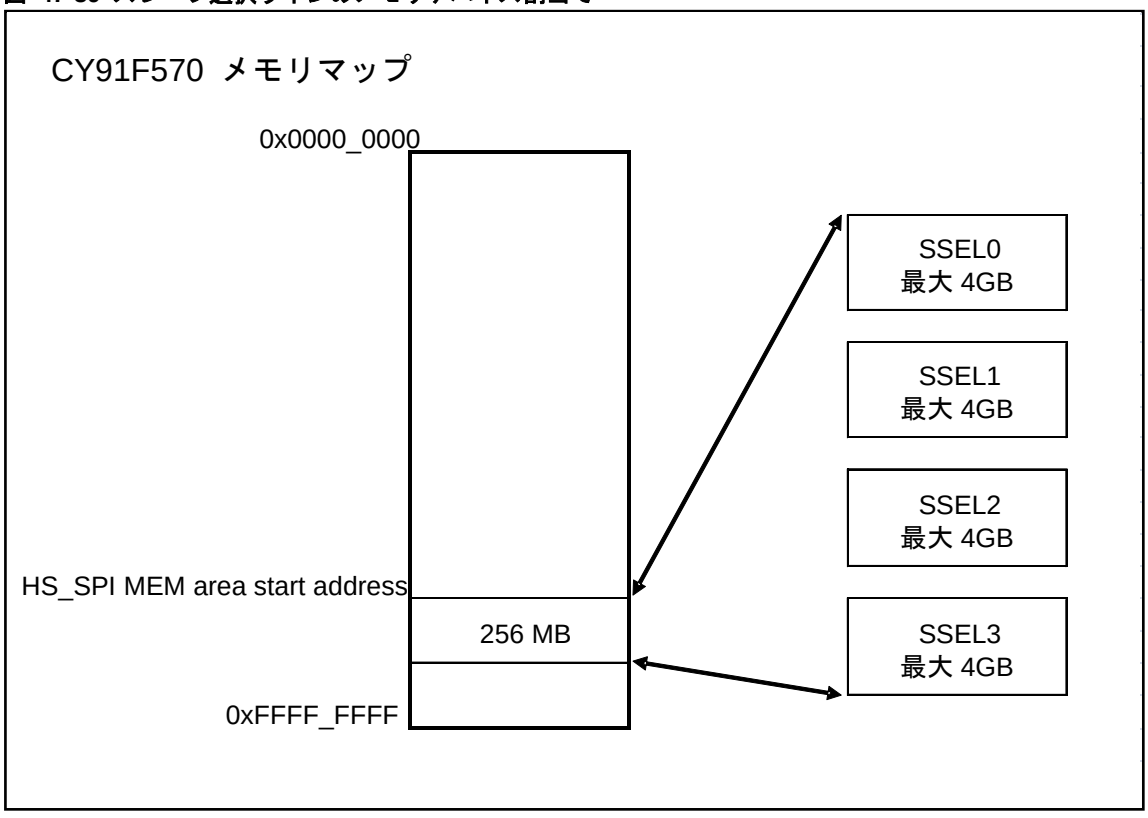
メモリの割当てについて示します。

コマンドシーケンサモードを使って、HS_SPIの4つのスレーブ選択出力上のシリアルフラッシュかシリアルSRAMメモリデバイスを最大4つまで、メモリに割り当てることができます。メモリ割り当てを行ったデバイスはすべて同一ファミリでなければなりません。

コマンドシーケンサモードでは、最大 4 つまでの外部メモリデバイスを割り当てるために、HS_SPI には 256MB のメモリ空間が割り振られています。各スレーブ選択はコマンドシーケンサのアドレス拡張機構を用いて、理論的には最大 4GB(32 ビットのアドレスバス)のメモリにアクセスすることが可能です。アドレス拡張機構により、19 ビットアドレス拡張レジスタ(HSSPIn_CSAEXT レジスタ)の上位ビットとオンチップバスアドレスバスの数ビットを連結して、各スレーブ選択上でアクセスできる 32 ビットアドレスを構成できます。この機能の詳細はのちほど説明します。

CY91570 シリーズの 256MB のアドレス空間は、実際には図 47-39 に示すように、外部シリアルメモリの 16GB に割り当てられています。

図 47-39 スレーブ選択ラインのメモリデバイス割当て



● スレーブの選択

HSSPIn_CSCFG:MSEL は、各スレーブ選択ラインに関連するオンチップバスアドレス空間のサイズを表しています。HSSPIn_CSCFG:MSEL の値と、CPU コア、または DMA コントローラのような他のオンチップバスマスタによってオンチップバスアドレスバス上に出力されたアドレスを基にして、HS_SPI コマンドシーケンサは 4 つのスレーブ選択ラインのどれをアサートするかを決めます。詳細は表 47-37 を参照してください。

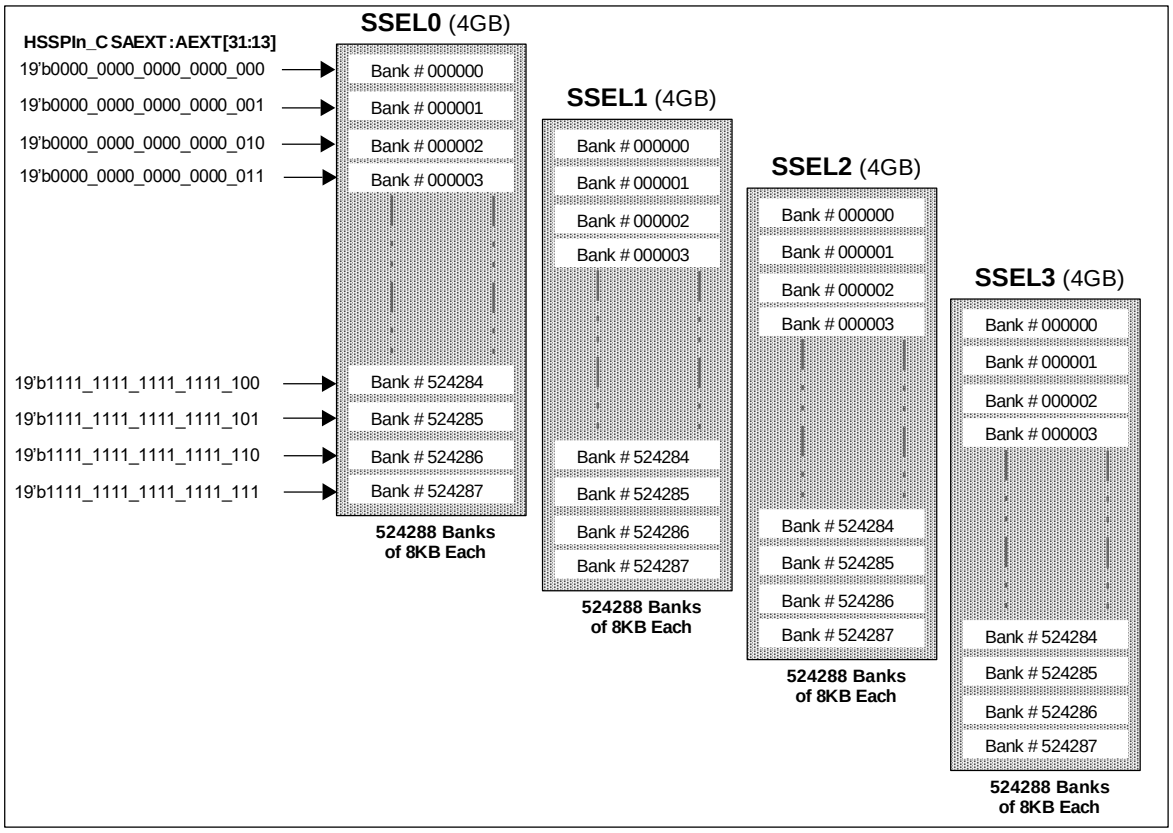
例として、HSSPIn_CSCFG:MSEL が、各スレーブ選択と関連するオンチップバスアドレス空間が 8KB であることを示していると仮定ます。もし、オンチップバスアドレスが「HS_SPI Memory Base Address」と「HS_SPI Memory Base Address + 8KB」の間であるならば、スレーブ選択 0 がアサートされます。もし、オンチップバスアドレスが「HS_SPI Memory Base Address + 8KB」と「HS_SPI Memory Base Address + 16KB」の間であるならば、スレーブ選択 1 がアサートされます。同様にしてスレーブ選択が決められますが、オンチップバスアドレスが「HS_SPI Memory Base Address + 32KB」を超える場合は、アドレスは範囲外となり HSSPIn_FAULTF:UMAF 割込みフラグがセットされます。

● 32 ビットメモリアドレスの生成

CY91570 シリーズの 256MB のアドレス空間からスレーブ選択ライン上の 4GB アドレス空間への割当ては、アドレス拡張機構により実現します。各メモリデバイスは複数のメモリバンクから構成されていると考えると分かりやすいでしょう。各バンクのサイズは HSSPIn_CSCFG:MSEL で設定可能です。HSSPIn_CSAEXT レジスタの値を変えることで、バンクを選択することができます。毎回 HSSPIn_CSAEXT レジスタをプログラムしなおすことにより、選択したスレーブの別のバンクにアクセスすることができます。これにより、異なる複数のバンクを通して最大 4GB のメモリデバイスのアドレス指定をすることができます。

HSSPIn_CSCFG:MSEL フィールドが"0000"にプログラムされたとき、4GB のデバイスがそれぞれどのようにして 524288 個のバンクを構成するかを図 47-40 に示します。

図 47-40 各スレーブ選択/スレーブ内バンク上の 4G バイトのアドレッシング(HSSPIn_CSCFG:MSEL=0000)



HS_SPI がオンチップバス上にて受け取ったオンチップバスアドレスの下位ビットは、アドレス拡張ビットにより選択されたバンク内のオフセットとして使われます。

アドレス拡張レジスタの適切な数のビットとオンチップバスアドレスバスの適切なビットの数を結合して、シリアルインタフェース上のメモリにアクセスする 32 ビットのアドレスが得られます。表 47-37 に CY91570 シリーズのアドレス空間へのメモリアドレス割当てを示します。

表 47-37 CY91570 シリーズのアドレス空間へのメモリアドレス割当て

HSSPIn_CSCFG:MSEL [3:0]	各スレーブ選択のメモリバンクサイズ/各スレーブ選択関連のオンチップバスアドレス範囲サイズ	使用可能なスレーブ選択ラインの数	スレーブ選択上のメモリバンク選択のための HSSPIn_CSAEXT レジスタからの使用ビット数	バンクのメモリ位置アドレス指定のためのオンチップバスアドレスからの使用ビット数
0000	8K バイト	SSEL0, SSEL1, SSEL2, SSEL3	AEXT[31:13]	HADDR[12:0]
0001	16K バイト		AEXT[31:14]	HADDR[13:0]
0010	32K バイト		AEXT[31:15]	HADDR[14:0]
0011	64K バイト		AEXT[31:16]	HADDR[15:0]
0100	128K バイト		AEXT[31:17]	HADDR[16:0]
0101	256K バイト		AEXT[31:18]	HADDR[17:0]
0110	512K バイト		AEXT[31:19]	HADDR[18:0]
0111	1M バイト		AEXT[31:20]	HADDR[19:0]
1000	2M バイト		AEXT[31:21]	HADDR[20:0]
1001	4M バイト		AEXT[31:22]	HADDR[21:0]
1010	8M バイト		AEXT[31:23]	HADDR[22:0]
1011	16M バイト		AEXT[31:24]	HADDR[23:0]
1100	32M バイト		AEXT[31:25]	HADDR[24:0]
1101	64M バイト		AEXT[31:26]	HADDR[25:0]
1110	128M バイト	SSEL0 と SSEL1 のみ	AEXT[31:27]	HADDR[26:0]
1111	256M バイト	SSEL0 のみ	AEXT[31:28]	HADDR[27:0]

.

右端の 2 列は、CY91570 シリーズのアドレス空間からメモリアドレス割付けに際して、最終的にシリアルメモリの 32 ビットアドレスを得るため HSSPIn_CSAEXT:AEXT とオンチップバスアドレス(HADDR)のどのビットを結合するかを示しています。

この方法で生成された最終的なメモリアドレスは 32 ビットアドレスですが、ソフトウェアはメモリの読出し/書込みコマンドシーケンスのアドレスフェーズの間に、この 32 ビットアドレスからシリアルメモリデバイスへ送るバイト数を決めることができることに注意してください。

47.5.3.2 コマンドシーケンスの開始

コマンドシーケンスの開始について説明します。

コマンドシーケンサがメモリ割付けされたシリアルデバイスに対するオンチップバス上の読取りアクセスを受け付けると、4 つのスレーブ選択ラインの 1 つに対し対応するメモリ読取りコマンドを開始し、受け取ったデータをパラレル化し、読取りデータを返します。

同様に、メモリ展開したシリアルデバイスへのオンチップバス上の書込みアクセスを受け取ると、4 つのスレーブ選択ラインの 1 つに対し、対応するメモリ書込みコマンドを開始し、書き込むデータを SDATA ラインにシリアルに転送します。

HS_SPI がメモリ読取りコマンドを開始し、シリアルメモリデバイスから読取りデータを受け取ると、HS_SPI のオンチップバスのスレーブポートはオンチップバスにウェイトステートを挿入します。同様に、シリアルメモリ書込みシーケンスでもウェイトステートが挿入されます。

HS_SPI は、オンチップバスマスタが発行した直前のアドレスとオンチップバス転送タイプを覚えています。新しいトランザクションのアドレスが連続していない場合、またはコマンドが切換わった(読出し→書込み，または書込み→読出し)場合、シリアルインタフェース上に新しいコマンドが発行されます。

47.5.3.3 オンチップバスのアイドルによるタイムアウト

オンチップバスのアイドルによるタイムアウトについて説明します。

シリアルデバイスがコマンドシーケンサモードでアクセスされたのち、HS_SPI はシリアルトランザクションが終わっても、スレーブ選択ラインのアサートを続けます。HSSPIn_CSITIME:ITIME で定義された期間以内に、直前の処理と連続したメモリアドレスへの同一命令(読出し，書込み)の新しいオンチップバストランザクションが検出されると、HS_SPI はシリアルデバイス上で新しいコマンドシーケンスフェーズを始める代わりにそれまでのシリアル転送を続けます。これによって、アクセス時間が短縮されます。メモリアドレスがその直前のアドレスと連続していなかったり、読出し，書込みといった命令が直前のものと異なるオンチップバスアクセスが発生したりすると、アイドル期間が終了する前であっても、HS_SPI はスレーブ選択のアサートを停止し(実行中の転送の終了を意味します)、新たな転送を開始します。アイドル期間中にそれまでのメモリアドレスと連続するオンチップバスアクセスがない場合は、アイドルタイムが切れたのちに、HS_SPI はスレーブ選択のアサートを停止して、転送を終了します。

このようにして、HSSPIn_CSITIME:ITIME を用いることにより、任意の期間同様のシリアルトランザクションを続けることによりメモリ割付けしたアクセスの全体としての性能を向上させています。

HSSPIn_CSITIME:ITIME で設定される時間の単位はオンチップバスクロック入力のサイクル数です。

47.5.3.4 CSR におけるコマンドシーケンスの設定

CSR におけるコマンドシーケンスの設定について説明します。

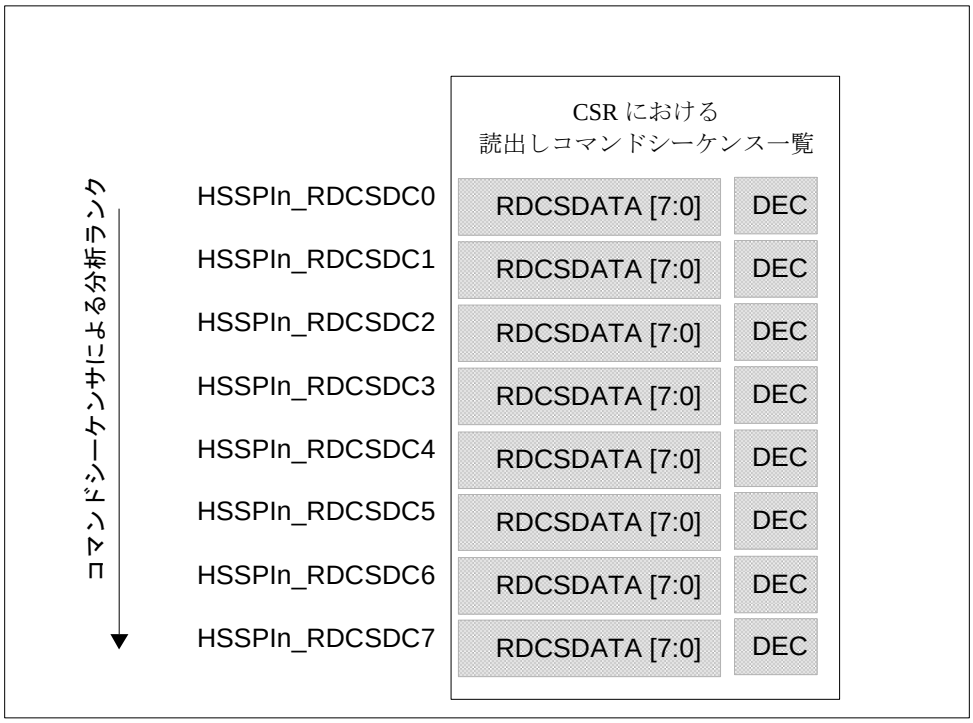
コマンドシーケンサはメモリ読出しアクセスをサポートします。また、シリアル SRAM デバイスがメモリ割り当てされているなら、HSSPIn_CSCFG:SRAM=1 の設定により、コマンドシーケンサによる書込みアクセスも可能となります。

コマンドシーケンスモードの HS_SPI によって生成されるコマンドフェーズのシーケンス(命令フェーズ, アドレスフェーズ, データフェーズ)は、HS_SPI の初期化の間に CSR のソフトウェアにより設定されます。

● メモリ読出しコマンドシーケンスの生成

メモリ読出しトランザクションに関するコマンドフェーズのシーケンスは、8 つの HSSPIn_RDCSDC[0:7]レジスタのリストで設定できます。リストの中の 8 つのレジスタは HSSPIn_RDCSDC0 から HSSPIn_RDCSDC07 までそれぞれ分析されます。詳細は図 47-41 を参照してください。

図 47-41 メモリ読出しコマンドシーケンス一覧



これらレジスタのそれぞれにある DEC ビットはデータバイト (RDCSDATA[2:0]) を表 47-38 に示されているようにデコードする必要があるか、または RDCSDATA[7:0]のデータバイトをそのまま転送するかを示しています。

表 47-38 読出しコマンドシーケンスリストのデコーディング

DEC	RDCSDATA [2:0]	説明
0	-	RDCSDATA[7:0]をそのまま送信する
1	000	シリアルメモリにアクセスするためのアドレスビット[07:00]を送信する。
1	001	シリアルメモリにアクセスするためのアドレスビット[15:08]を送信する。
1	010	シリアルメモリにアクセスするためのアドレスビット[23:16]を送信する。
1	011	シリアルメモリにアクセスするためのアドレスビット[31:24]を送信する。
1	100	1 バイト時間の間、SDATA 出力ラインをトライステートにする
1	101	HSSPIn_PCC0~3:SDIR ビットで設定されたシフト方向の如何によらず、送信は次の順に行う (1) RDCSDATA[07:04]をそのまま送信する。送信方向は HSSPIn_PCC0~3:SDIR ビットの設定値による。 (2) 4 ビット時間の間、SDATA 出力ラインをトライステートにする。
1	111	リストの最終項

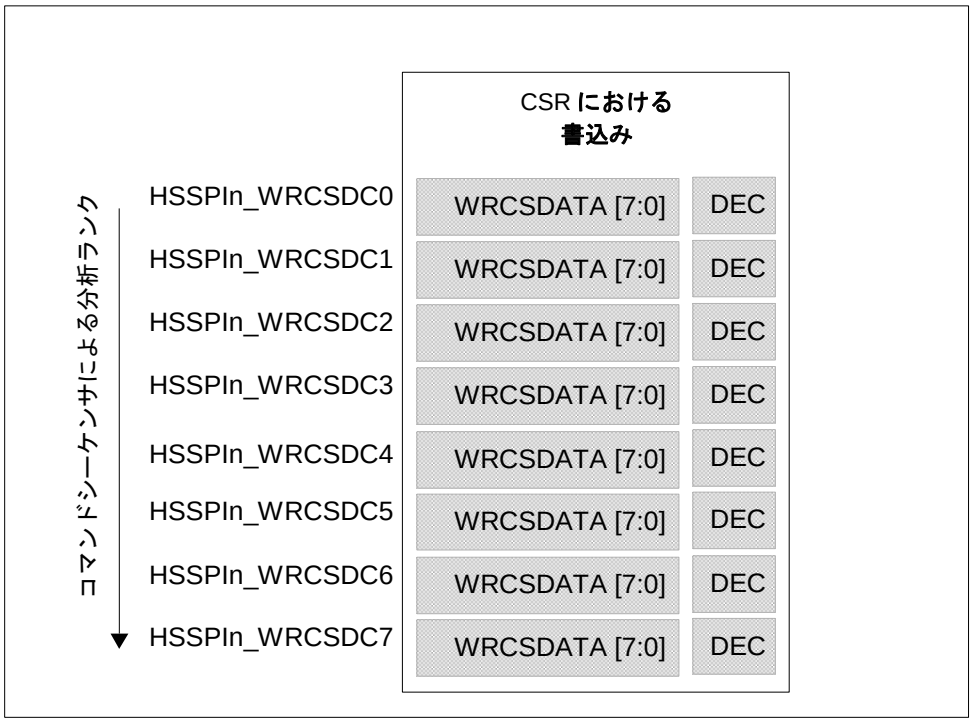
コマンドシーケンサは「End of List(リスト最終項)」を受け取るか、または HSSPIn_RDCSDC7 レジスタが終了するかのいずれか早い時点で、データ読出しサイクルに切り替わります。データ読出しサイクルの間、SDATA ライン上のシリアルデータはサンプリングされ、オンチップバス読出しトランザクションへの応答として、パラレル化されたデータがオンチップバスマスタへ戻されます。

● メモリ書込みコマンドシーケンスの生成

メモリ書込みコマンドシーケンスは HSSPIn_CSCFG:SRAM ビットにより有効とされているときにのみ、コマンドシーケンサにより開始されます。

メモリ書込みトランザクションに関するコマンドフェーズのシーケンスは、8 つの HSSPIn_WRCSDC[0:7]レジスタのリストで設定できます。リストの中の 8 つのレジスタは HSSPIn_WRCSDC0 から HSSPIn_WRCSDC07 までそれぞれ分析されます。詳細は図 47-42 を参照してください。

図 47-42 メモリ書込みコマンドシーケンスリスト



これらレジスタのそれぞれにある DEC ビットはデータバイト(WRCSDATA[2:0])を表 47-39 に示されているようにデコードする必要があるか、または WRCSDATA[7:0]のデータバイトをそのまま転送するかを示しています。

表 47-39 書込みコマンドシーケンスリストのデコーディング

DEC	WRCSDATA [2:0]	説明
0	- ‘	WRCSDATA[07:00] をそのまま送信する
1	000	シリアルメモリにアクセスするためのアドレスビット[07:00]を送信する。
1	001	シリアルメモリにアクセスするためのアドレスビット[15:08]を送信する。
1	010	シリアルメモリにアクセスするためのアドレスビット[23:16]を送信する。
1	011	シリアルメモリにアクセスするためのアドレスビット[31:24]を送信する。
1	100	1 バイト時間の間、SDATA 出力ラインをトライステートにする
1	101	HSSPIn_PCC0~3:SDIR ビットで設定されたシフト方向の如何によらず、送信は次の順に行う (1) RDCSDATA[07:04]をそのまま送信する。送信方向は HSSPIn_PCC0~3:SDIR ビットの設定値による。 (2) 4 ビット時間の間、SDATA 出力ラインをトライステートにする。
1	111	リストの最終項

コマンドシーケンサは「End of List(リスト最終項)」を受け取るか、または HSSPIn_WRCSDC7 レジスタが終了するかの、いずれか早い時点で、データ書込みサイクルに切り換わります。データ書込みサイクルの間、オンチップバス書込みデータバス(HWDATA)からのパラレルデータは、設定された SPI プロトコルにしたがい SDATA ライン経由でシリアルに送信されます。

47.6 HS_SPI の使用上の注意事項

本節は”Programmer’s Guide”であり、HS_SPI モジュールをプログラムする上で注意すべき点について説明します。HS_SPI モジュールのプログラムする前に、本節に示すガイドラインの一読をお願いします。

47.6.1 一般的な使用上の注意事項

一般的な使用上の注意事項について説明します。

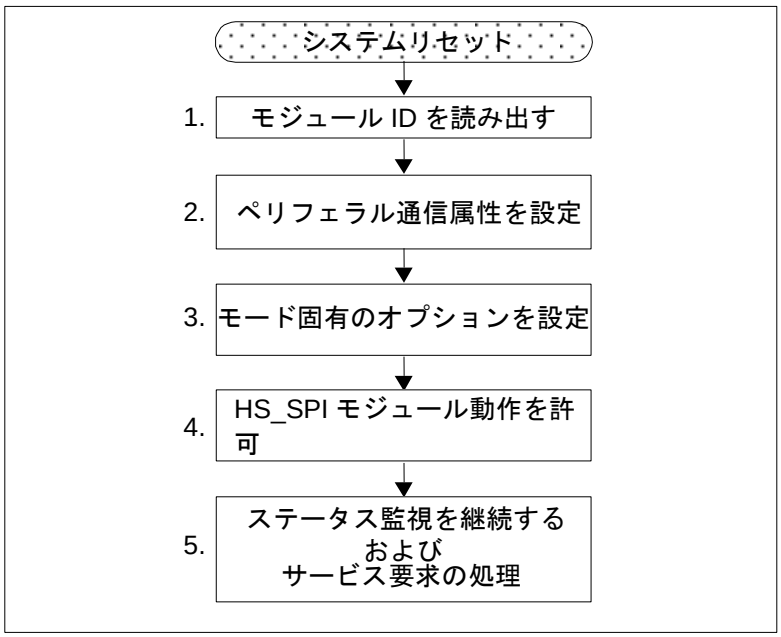
- シリアルトランザクションに関連するパラメータと制御ビットは、トランザクションの途中で切り替えてはいけません。HSSPIn_PCC0~3 レジスタの属性の選択，ダイレクトモード動作とコマンドシーケンサモード動作の切り替え，マスタモード動作とスレーブ動作の切り替え，などがこれにあたります。これらの切替えは、HS_SPI モジュールが無効(HSSPIn_MCTRL:MEN=0)となり、実行中のシリアル転送が完了(HSSPIn_TXF:TSSRS=1 または HSSPIn_RXF:RSSRS=1)したのちにのみ、行なってください。SSPIn_DMSTATUS:TXACTIVE ビットと HSSPIn_DMSTATUS:RXACTIVE ビットを読むことにより、ソフトウェアは HS_SPI モジュールがすべての転送を終了したことを確認できます。
- ダイレクトモード動作である種のシリアルフラッシュメモリデバイスの転送プロトコルをエミュレートするときは、先頭のバイトをレガシーモードで転送し、その後、残りのデータバイトをデュアルビットモードかクワッドビットモードで転送することが必要となることがあります。このような場合には、シリアル転送の処理中に、すなわち、スレーブ選択ライン SSEL0~3 の 1 つがアサートされている間に、HSSPIn_DMTRP レジスタを変更することが必要になる可能性があります。シリアル転送開始後に、HSSPIn_DMTRP レジスタの再設定が必要になったときは、「●TX-DATA の不足または RX-FIFO の空き不足による停止」の項で説明したように、実行中のシリアル転送をいったん停止した後に再設定することができます。
- しかしながら、シリアル転送を停止している間に、HSSPIn_DMTRP を「TX-Only Legacy」から「RX-Only Legacy」モードへと切り替えるプログラム変更をしてはいけません。「RX-Only Legacy」から「TX-Only Legacy」への切り替えも禁止です。転送プロトコルを「TX-Only Legacy」から「RX-Only Legacy」へ切り替える代わりに、転送の開始前にソフトウェアで HSSPIn_DMTRP を「TX and RX Legacy」モードにプログラムを変更し、その後、受信が不要なら受け取ったデータのバイトを無視したり、送信が不要ならダミーのデータを送信したりすることで代用できます。
- ダイレクトモード動作のときは、HS_SPI の内部 FIFO がオーバランしたり、アンダランしたりしないようソフトウェアが監視する必要があります。FIFO がオーバランやアンダランとなると、FIFO のフィルレベル(HSSPIn_DMSTATUS:TXFLEVEL と HSSPIn_DMSTATUS:RXFLEVEL)はもはや意味をなさず、ソフトウェアは FIFO の内容を破棄しなければなりません。
- ソフトウェアは、FIFO をシリアル転送に使用する前に、SSPIn_FIFOCFG:RXFLSH と HSSPIn_FIFOCFG:TXFLSH ビットを使って、対応する FIFO の内容を破棄しなければなりません。FIFO の内容を破棄することにより、その直前の転送などに起因する不要なデータが残っていることを防ぎます。
- ダイレクトモードでのマスタ動作時には、転送が終わるたびに、RX-FIFO がフルになっていなければ、RX シフトレジスタからのデータを RX-FIFO にに送り込みます。
  - シリアル転送が終了しかけている間に RX FIFO がフルになってしまうと、シリアル転送は停止し、残りのデータは HS_SPI が停止している間、RX シフトレジスタに残されます。RX FIFO に空きができると、HS_SPI は停止状態から抜け、RX シフトレジスタの残りのデータを RX-FIFO に送り込み、スレーブ選択ラインを解除します。残りのデータは、RX シフトレジスタの状態に関係なく、RX-FIFO に送り込まれます。
  - このように、ダイレクトモード時のマスタ動作では、RX シフトレジスタにデータが残ることはありません。HSSPIn_RXBITCNT レジスタのビットカウントは常に“0”で、受信したデータはすべて RX-FIFO に送り込まれます。
- ダイレクトモード時のスレーブ動作では、転送が停止すると、HS_SPI が最後に受け取ったデータは RX-FIFO に送り込まれず RX シフトレジスタの中に残ります。ソフトウェアは HSSPIn_RXSHIFT レジスタの残りのデータと、HSSPIn_RXBITCNT レジスタのビットカウントを読み取らねばなりません。

47.6.2 HS_SPI モジュールのプログラム手順

HS_SPI モジュールのプログラム手順について示します。

図 47-43 は HS_SPI モジュールを使用するときの一般的な手順を示しています。

図 47-43 プログラムフローチャート(一般手順)



1. CY91570シリーズには1つのHS_SPIモジュールしか搭載しておらず、モジュールID機能をサポートしていません。引き続きドライブソフトウェアにHS_SPI CSRとMEMベースアドレスをセットしてください。
2. 次に、HS_SPIで接続されているシリアルデバイスとのペリフェラル通信に関連する属性を設定します。マスタモードでは、HS_SPIは最大4個までのシリアルデバイスと接続できます。スレーブモードでは、HS_SPIは1個のマスタデバイスと接続できます。シリアル通信に関連したクロック極性, クロック位相, 転送周波数(クロック分周比とクロックソース選択ビット), スレーブ選択信号の極性などの属性は、HSSPIn_PCC0, HSSPIn_PCC1, HSSPIn_PCC2, HSSPIn_PCC3などのレジスタで設定します。これらの属性はHS_SPIが実際に接続するリモートシリアルデバイスの仕様に合わせる必要があります。これらの設定はHS_SPIモジュールがアクティブな間は変更してはいけません。ソフトウェアで設定値を変更するときは、まずHS_SPIモジュールを無効とし、処理中のシリアル転送の終了後に行ってください。
3. HS_SPIはHSSPIn_MCTRL:CSENビットでダイレクトモードかコマンドシーケンサモードに設定できます。どちらのモードが使われるかによって、ソフトウェアはモードに特有なレジスタを設定する必要があります。ダイレクトモードに特有なレジスタは次のものです。HSSPIn_TXF, HSSPIn_TXE, HSSPIn_TXC, HSSPIn_RXE, HSSPIn_RXF, HSSPIn_RXE, HSSPIn_RXC, HSSPIn_DMCFG, HSSPIn_DMSTATUS, HSSPIn_RXSHIFT, HSSPIn_TXFIFO0～15, HSSPIn_RXFIFO0～15, HSSPIn_FIFOCFG
   コマンドシーケンサモードに特有なレジスタは次のものです。
   HSSPIn_CSCFG, HSSPIn_CSITIME, HSSPIn_CSAEXT, HSSPIn_RDCSDC0～7, HSSPIn_WRCSDC0～7
4. すべてのモジュール特有の設定を行ったのち、HSSPIn_MCTRL:MENビットを"1"に設定し、HS_SPIモジュールを有効にしてください。
5. HS_SPIモジュールが有効になると、通常の動作を開始します。ソフトウェアは種々のステータスビットを使ってHS_SPIモジュールの状態を監視しなければなりません。HS_SPIモジュールがサービス要求を開始するように設定されていれば、モジュールは周期的にサービス要求(割込み)を起動します。ソフトウェアはHS_SPIの正常な動作を保証するためにこれらの要求を処理します。

47.6.3　ダイレクトモード動作での HS_SPI の注意事項

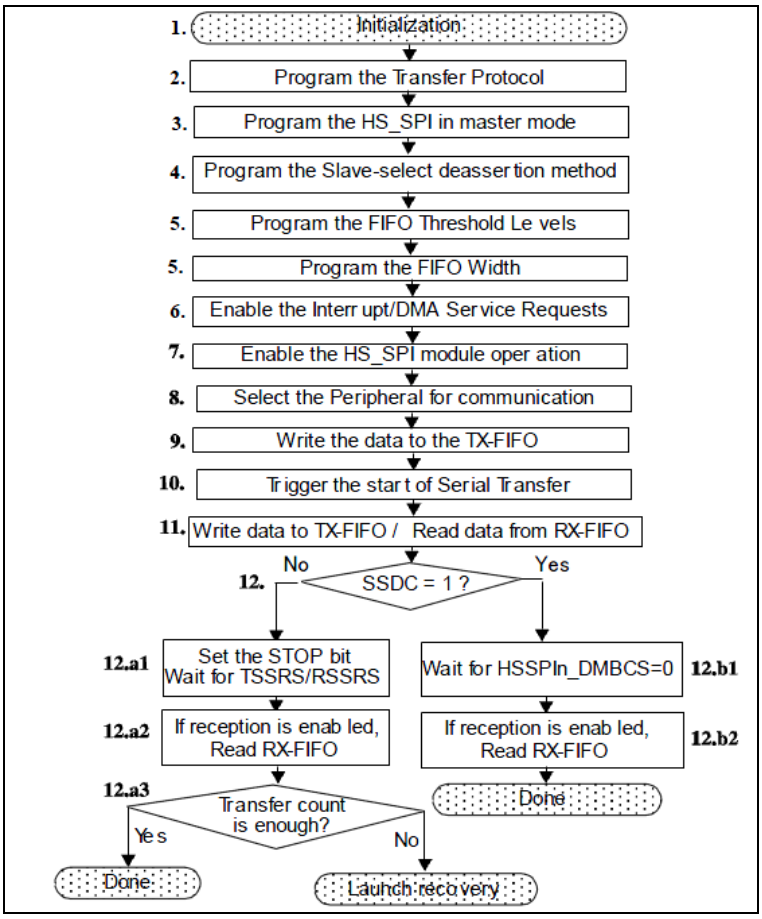
ダイレクトモード動作での HS_SPI の注意事項について示します。

47.6.3.1　マスタモード

マスタモードについて示します。

図 47-44にダイレクトモード動作時にHS_SPIモジュールをマスタとして使用するときの一般的な手順を示します。

図 47-44 プログラムフローチャート:ダイレクトモード動作時のマスタの HS_SPI



* DMA サービス要求は CY91570 シリーズの HS_SPI ではサポートされていません。

- システムリセットの後、ソフトウェアはHSSPIn_PCC0～3レジスタのペリフェラル通信関連の属性をセットして、HS_SPIモジュールを初期化します。これらの属性をHS_SPIとシリアルに接続しているリモートシリアルデバイスの仕様に合わせることが重要です。また、HSSPIn_MCTRL:CSENビットが"0"にリセットされていることを確認してください。
- 次に、HSSPIn_DMTRP:TRPで転送プロトコルを設定します。HS_SPIシリアル転送がレガシー，デュアルビット，クワッドビットSPIプロトコルのどれを使用するのか、また、HS_SPIが送信のみ，受信のみ，または送受信のどれを使用するのかを設定します。

- 次に、HSSPIn_DMCFG:MST ビットをセットします。これで HS_SPI がスレーブデバイス，またはマスタデバイスのどちらとして使われるのかが決まります。
- HSSPIn_DMCFG:SSDCを設定します。これによってスレーブ選択出力のアサート停止方法を選択します。バイトカウンタモードに設定すると、シリアル転送するバイト数としてHSSPIn_DMBCC:BCCの値を取り込みます。ソフトウェアフロー制御に設定したときは、予定したデータの送受信が終了したのち、ソフトウェアでHSSPIn_DMSTOP:STOPビットをセットする必要があります。
- HSSPIn_FIFOCFGレジスタでFIFOのしきい値レベルを設定します。これらのレベルを設定することでサービス要求のアサートを制御することができます。また、HSSPIn_FIFOCFG: FWIDTHでFIFO幅を設定します。
- サービス要求を設定します。TX-FIFO書込み要求による割込みを有効とするには、HSSPIn_TXEレジスタのビットを設定してください。RX-FIFOの読出し要求による割込みを有効にするには、HSSPIn_RXEレジスタのビットを設定してください。
- 以上でHS_SPIをダイレクトモード動作時させる初期化の手順は完了です。モジュールを有効にするため、HSSPIn_MCTRL:MENビットをセットしてください。
- HS_SPIが転送を開始するペリフェラルをHSSPIn_DMPSEL:PSELで選択してください。
- HS_SPIがHSSPIn_DMTRP:TRPでTX-Only，またはTX-and-RXモード動作に設定されている場合は、HSSPIn_TXFIFO0~15レジスタにアクセスして、送信データをTX-FIFOに書き込みます。HSSPIn_TXFIFO0~15レジスタへ書き込む前に、HSSPIn_FIFOCFG:TXCTRLの値を必要に応じて修正します。一般的には、すなわち、TX-FIFOに書き込まれたデータがそのままの形で送信される場合は、HSSPIn_FIFOCFG:TXCTRLビットを"0"にリセットします。HS_SPIが1バイト時間または4ビット時間の間シリアルデータラインをトライステートとするように設定された場合のみ、HSSPIn_FIFOCFG:TXCTRLビットを"1"にセットし、HSSPIn_TXFIFO0~15への書込みを行わなければなりません。
- HS_SPIがマスタに設定されているときは、HSSPIn_DMSTART:STARTビットをセットするとシリアルトランザクションを開始します。シリアルトランザクションが始まり、HSSPIn_DMTRP:TRPで送信が有効になると、HS_SPIはTX-FIFOを読み出してシフトレジスタにロードします。シフトレジスタはHSSPIn_PCC0~3:SDIRフィールドの設定に基づいて左または右方向にシフトし、送信データはシリアルラインへシフトします。HS_SPIがHSSPIn_DMTRP:TRPで受信動作を有効にすると、HS_SPIは受信したデータをシフトレジスタにシリアルにシフトしてこのデータをパラレル化します。パラレル化された受信データはRX-FIFOへ送られます。
- TX-FIFOレベルが閾値よりも低いか、HS_SPI RX-FIFOレベルがしきい値を超えているときはいつでも、HS_SPIはサービス要求をアサートします。ソフトウェアはFIFOへの書込みや読出しによって、HS_SPIが通常の動作を行えるよう保証しなければなりません。割込み処理が完了したら、ソフトウェアはHSSPIn_TXCレジスタかHSSPIn_RXCレジスタを使って割込み要求をクリアする必要があります。
- ソフトウェアがシリアル転送を停止するには、ソフトウェアフロー制御モードとバイトカウントモードの2つのモードがあり、HSSPIn_DMCFG:SSDCビットの設定で決まります。

ソフトウェアフロー制御の場合：

- a1) ソフトウェアはHSSPIn_TXF:TSSRSフラグかHSSPIn_RXF:RSSRSフラグがセットされるのを待ちます。これはスレーブ選択が解除されたことを示します。
- a2) HSSPIn_DMTRPレジスタにおいて受信が有効な場合、ソフトウェアはRX-FIFOから受信データを読出します。
- a3) ソフトウェアフロー制御モードを使って転送されるデータのバイト数が十分でない場合、ソフトウェアはやり直しの処理を行わなければなりません。

バイトカウンタモードの場合：

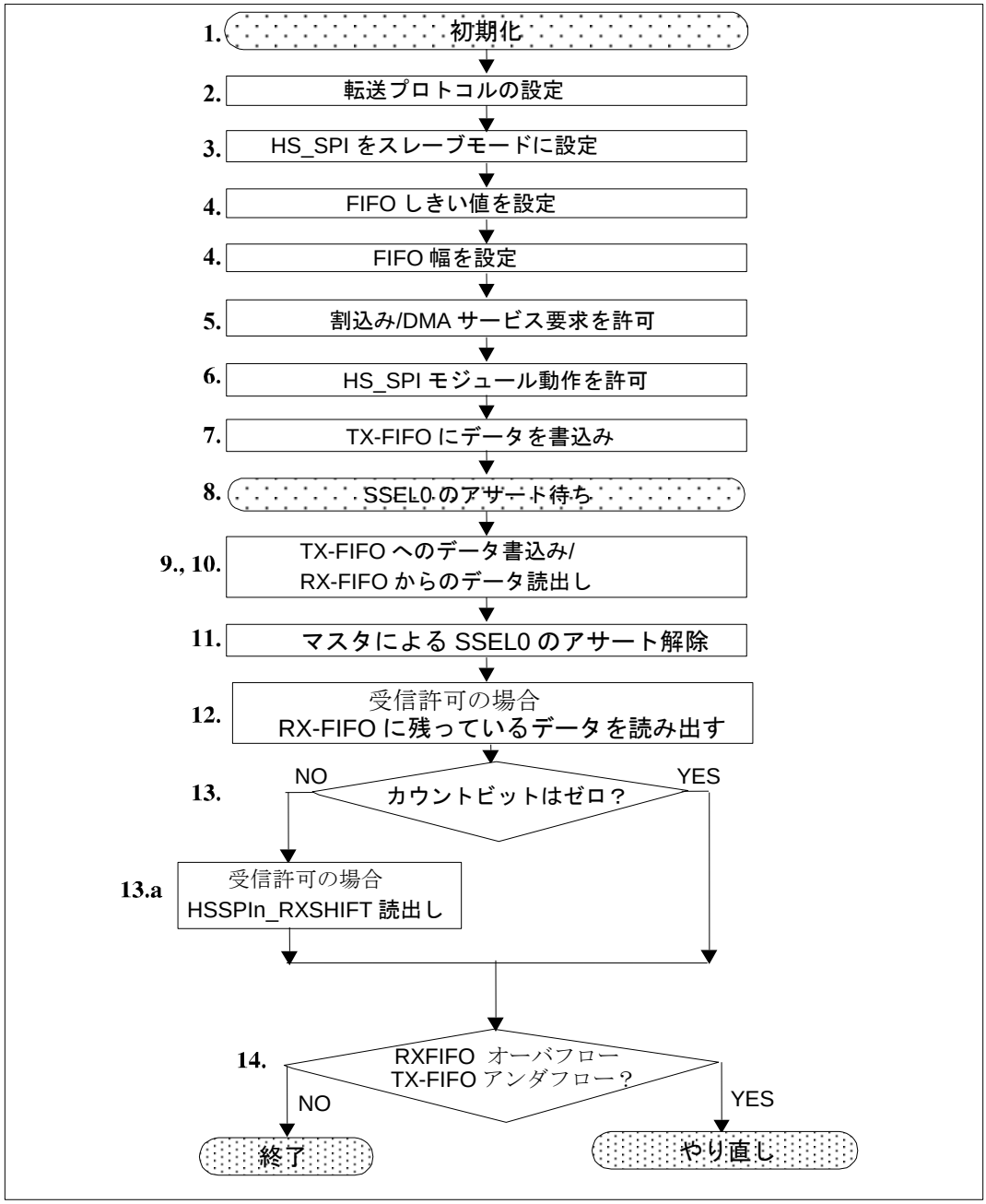
- b1) ソフトウェアはHSSPIn_DMBCSレジスタの値が"0"になるまで待ちます。
- b2) HSSPIn_DMTRPレジスタで受信が有効な場合、ソフトウェアはRX-FIFOから受信データを読出します。
- 正常動作時、ソフトウェアは手順の 9 項から 12 項までを繰り返します。1 項の初期化に戻ることもあります。
- ダイレクトモード動作とコマンドシーケンサモード動作を切り換えたり、シリアル転送に直接影響するパラメータを再設定したりするときは、ソフトウェアはまず処理中の転送を停止し、HS_SPI モジュールを無効にします。HS_SPI モジュールによるすべての転送が終了したことを確認した後でなければ、ソフトウェアはモジュールを再設定できません。転送の終了は HSSPIn_DMSTATUS:TXACTIVE ビットと HSSPIn_DMSTATUS: RXACTIVE ビットを読み出して確認します。

47.6.3.2 スレーブモード

スレーブモードについて示します。

図 47-45 にダイレクトモード動作時に HS_SPI モジュールをスレーブとして使用するときの一般的な手順を示します。

図 47-45 プログラムフローチャート : ダイレクトモード動作の HS_SPI スレーブ



* : CY91570 シリーズの HS_SPI は DMA サービス要求をサポートしていません。

- システムリセット後、ソフトウェアは HSSPIn_PCC0 レジスタのペリフェラル通信に関連した属性をセットし、HS_SPI モジュールを初期化する必要があります。これらの属性は HS_SPI がシリアルに接続しているリモートシリアルデバイスの仕様と一致している必要があります。また、HSSPIn_MCTRL:CSEN ビットが"0"にリセットされていることを確認してください。
- 次に、HSSPIn_DMTRP:TRP で転送プロトコルを設定します。レガシー, デュアルビット, クワッドビット SPI プロトコルのいずれを使用するのか、また HS_SPI は送信のみ, 受信のみ, 送受信の両方のいずれの機能を使用するのかを設定します。
- 次に、HSSPIn_DMCFG:MST ビットを設定します。これにより HS_SPI がスレーブデバイスとして使われるのか、マスタデバイスとして使われるのかを決めます。HS_SPI はスレーブデバイスとして使われると、SSEL0 入力から外部マスタが発信したシリアル転送にのみに応答します。したがって、スレーブモードで使用するときは、HS_SPI は HSSPIn_PCC0 レジスタのペリフェラル通信属性のみを用います。
- HSSPIn_FIFOCFG レジスタを設定して、FIFO のしきい値レベルをセットします。レベルの設定により、サービス要求のアサート条件を制御できます。また、HSSPIn_FIFOCFG:FWIDTH を設定して、FIFO 幅を選択してください。
- サービス要求を設定します。TX-FIFO の書込み要求に対する割込み要求を有効にするには、HSSPIn_TXE レジスタのビットを設定してください。RX-FIFO の読出し要求に対する割込み要求を有効にするには、HSSPIn_RXE レジスタのビットを設定してください。
- 以上でダイレクトモード動作の HS_SPI の初期化の手順は完了です。モジュールを有効にするため HSSPIn_MCTRL:MEN ビットをセットしてください。
- HS_SPI が HSSPIn_DMTRP:TRP で TX-Only モード動作, または TX-and-RX モード動作に設定されている場合、HSSPIn_TXFIFO0~15 レジスタにアクセスして、転送データを TX-FIFO に書き込んでください。HSSPIn_TXFIFO0~15 に書き込む前に、HSSPIn_FIFOCFG:TXCTRL の値を必要に応じて修正してください。通常の、TX-FIFO に書き込まれたデータがそのまま送信される場合は、HSSPIn_FIFOCFG:TXCTRL ビットを必ず“0”にリセットしてください。HS_SPI が 1 バイト時間か 4 ビット時間の間シリアルデータラインをトライステートにするよう指示された場合のみ、HSSPIn_FIFOCFG:TXCTRL ビットを“1”にセットし、HSSPIn_TXFIFO0~15 への書込みアクセスを行ってください。
- HS_SPI がスレーブに設定されているときは、HS_SPI はリモートマスタが SSEL0 入力をアサートするのを待ちます。
- シリアルトランザクションが始まり、HSSPIn_DMTRP:TRP で送信が有効になっていると、HS_SPI は TX-FIFO を読み出し、シフトレジスタにロードします。シフトレジスタは HSSPIn_PCC0~3:SDIR の設定に基づいて左または右方向へシフトされ、送信データはシリアルラインへシフトして送りだされます。HS_SPI が HSSPIn_DMTRP:TRP で受信動作が有効になっていると、HS_SPI は受信データをシリアルレジスタにシリアルにシフトしながらパラレル化します。シフトレジスタでパラレル化された受信データは RX-FIFO へ送り込まれます。
- TX-FIFO のレベルが閾値を下回るか、HS_SPI の RX-FIFO のレベルが閾値を超えるときは、HS_SPI はサービス要求をアサートします。ソフトウェアは FIFO を読み出したり、書き込んだりして HS_SPI の動作を行えるよう保証しなければなりません。割込み処理が完了したら、ソフトウェアは HSSPIn_TXC か HSSPIn_RXC レジスタを使って割込みサービス要求をクリアします。
- スレーブ動作の場合、転送の終了を示すためリモートマスタがスレーブ選択のアサート解除を行います。
- 受信が有効ならば、ソフトウェアは RX-FIFO から受信データを読み出す必要があります。
- スレーブモード動作においては、転送が停止すると、HS_SPI が受信したがまだ RX-FIFO に送り込まれていないデータは RX シフトレジスタに残ります。RX シフトレジスタで有効なビット数は RX ビットカウントレジスタに表示されます。したがって、受信が有効であるかどうかを確認するためソフトウェアは HSSPIn_RXBITCNT レジスタを読み出す必要があります。

13. a HSSPIn_RXBITCNT の値が“0”以外であれば、ソフトウェアは HSSPIn_RXSHIFT レジスタから残されているデータを読み出す必要があります。

- スレーブモードのときは、ソフトウェアは RX-FIFO がオーバフローしていないこと、および TX-FIFO がアンダランしていないことを監視していなければなりません。もしも、RX-FIFO がオーバフローの場合や TX-FIFO がアンダランの場合は、ソフトウェアはやり直しの処理を行わなければなりません。

- 通常の動作の場合、ソフトウェアは手順の 8 項から 14 項の動作を繰り返しますが、場合によっては初期化の手順に戻ることもあります。
- ダイレクトモードからコマンドシーケンサモードに切り換えたり、シリアル転送に直接影響するパラメータを再設定したりする場合、ソフトウェアはまず処理中の転送を停止し、HS_SPI モジュールを無効にします。このとき、コマンドシーケンサモードへの切り替えはスレーブモード動作からマスタモード動作への切り替えも伴うことに注意してください。HS_SPI モジュールによるすべての転送が終了したことを確認した後でなければ、ソフトウェアはモジュールを再設定できません。転送の終了は HSSPIn_DMSTATUS:TXACTIVE ビットと HSSPIn_DMSTATUS: RXACTIVE ビットを読み出して確認します。

47.6.4　メモリ領域に割り当てて使用する場合

メモリ領域に割り当てたメモリの使用について示します。

コマンドシーケンサモードでメモリ領域に割り当てたシリアルメモリを使用する場合には、下記の注意点を守ってください。

47.6.4.1　使用上の規則と注意事項

使用上の規則と注意事項について示します。

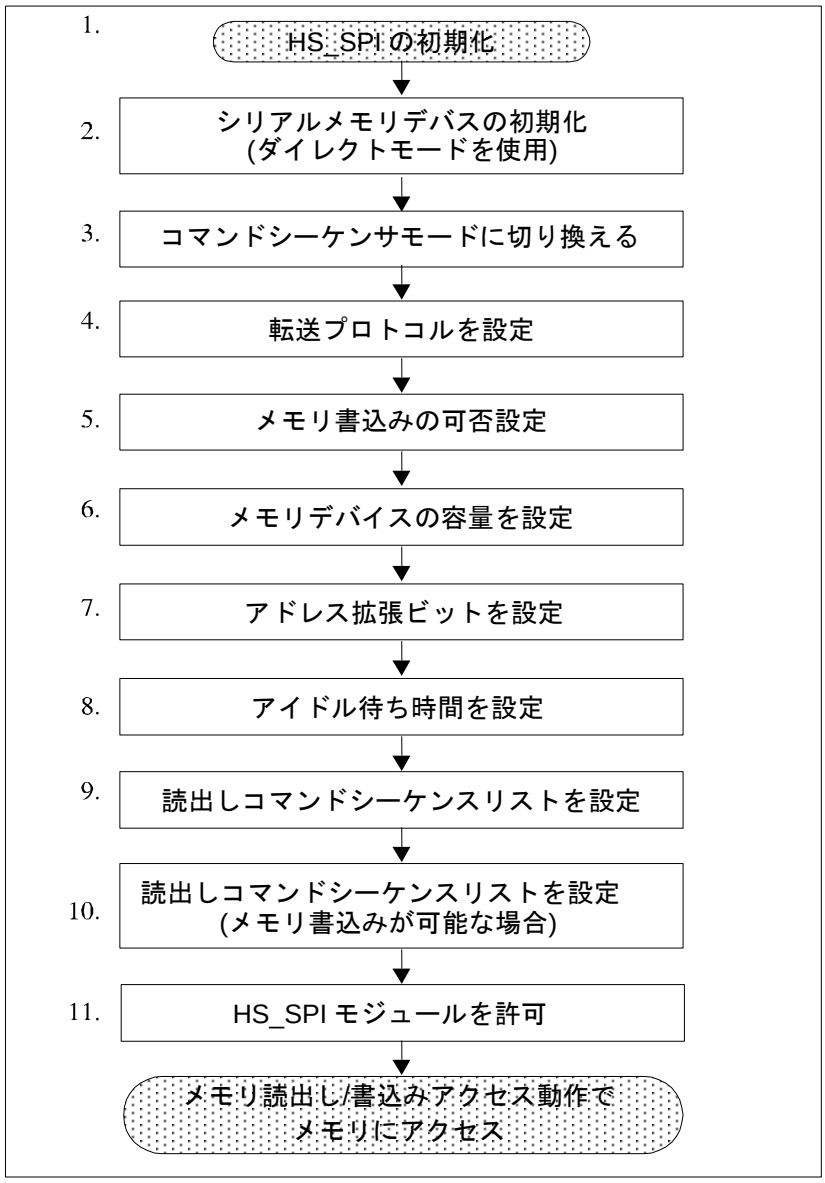
- コマンドシーケンサモードでは、HS_SPI に接続されるシリアルメモリデバイスはすべて同一ファミリ製品である必要があります。異なるメーカのシリアルメモリデバイスを混在することは避けてください。
- メモリデバイスが同じファミリ製品に属するものあっても、それぞれメモリサイズが異なるものを接続するときは、HSSPIn_CSCFG:MSEL の値を決める際には、最大容量のメモリデバイスを考慮する必要があります。ここで、HS_SPI がメモリ割付けされたシリアルメモリデバイスに送信する最終メモリアドレスのバイト数は、コマンドシーケンスリスト(HSSPIn_RDCSDC0~7 と HSSPIn_WRCSDC0~7 レジスタ)で設定されるということに注意が必要です。コマンドシーケンサモードでは、32 ビットアドレス体系のメモリデバイスと、同じファミリ製品でも 24 ビットアドレス体系のほかのメモリデバイスを接続することはできません。これは 24 ビットアドレス体系のメモリデバイスのアドレスフェーズは 3 サイクルのみなので、32 ビットアドレスのものと一緒に使うことができないためです。
- メモリデバイスが 21 ビットアドレスのみを利用し、HS_SPI のコマンドシーケンスがメモリデバイスへ 24 ビットのアドレスを送信するように設定されていると、ソフトウェアは上位 3 ビット[23:21] に HSSPIn_CSAEXT レジスタを使って“0”を書き込みます。使用されない上位 3 ビットが"0"にリセットされない場合は、HS_SPI に接続しているシリアルメモリデバイスのアドレスポインタはこれらをラップし、予期せぬ結果を生じることがあります。
- シリアル SRAM デバイスはバーストモードで動作するように設定されている場合にのみ、バースト動作をサポートします。しかし、コマンドシーケンサは常に SRAM デバイスはバーストモードであることを想定しています。HS_SPI のコマンドシーケンサモードを有効にする前に、ソフトウェアはダイレクトモード動作でシリアル SRAM デバイスをバーストモード動作に設定しておく必要があります。
- コマンドシーケンサモードでは、転送開始後は、レガシー, デュアルビット, クワッドビットモード間の切り替えはできません。このために、最近の仕様のシリアルフラッシュデバイスには、「連続読出しモード」で動作できるように初期化されなければ、コマンドシーケンサを有効にできないものがあります。メモリデバイスは HS_SPI のダイレクトモード動作を用いて連続読出しモードに設定できます。

47.6.4.2 プログラムフローチャート

プログラムフローチャートについて示します。

HS_SPI を使って、CY91570 シリーズのアドレス空間にメモリデバイスのアドレスを割り付ける際の一般的な手順を図 47-46 を示します。

図 47-46 シリアルデバイスをメモリ割付けするときの一般的な手順



- システムリセットの後、ソフトウェアは HSSPIn_PCC0, HSSPIn_PCC1, HSSPIn_PCC2 と HSSPIn_PCC3 レジスタのペリフェラル通信に関連した属性をセットして HS_SPI モジュールを初期化します。これらの属性は、HS_SPI がシリアル接続するリモートシリアルデバイスの仕様と同じであること必要があります。シリアルメモリデバイスがコマンドシーケンサモードでメモリ割付けされるときは、すべてのメモリデバイスがファミリ製品である必要があります。したがって、すべての 4 つのペリフェラル通信設定レジスタ HSSPIn_PCC0~3 は同一の設定値でなければなりません。

- 次に、メモリ割り付けを行うシリアルデバイスの初期化をおこないます。初期化はデバイスに固有で、レジスタセットの制御ビット/ステータスビットの設定を含む場合があります。例えば、クワッド SPI シリアルメモリデバイスを使用するために、デバイスを高性能モード(すなわちクワッドモード)にセットすることなどが考えられます。詳細は接続を予定しているシリアルメモリデバイスのデータシートで確認してください。このシリアルメモリデバイスの初期化は HS_SPI のダイレクトモードで行ってください。
- ダイレクトモードでシリアルデバイスの初期化を行った後、HS_SPI をコマンドシーケンサモードで再設定してください。ダイレクトモード動作からコマンドシーケンサモードに切り替えるため、ソフトウェアはまず、実行中の転送を停止し、HS_SPI モジュールを無効にします。HSSPIn_DMSTATUS:TXACTIVE と HSSPIn_DMSTATUS:RXACTIVE ビットを読み出して、HS_SPI モジュールがそのすべての転送を終了したことを確認したのち、初めてソフトウェアはモジュールをコマンドシーケンサモードに設定できます。
- 次は、転送プロトコルの設定です。すなわち、HSSPIn_CSCFG:MBM フィールドで、HS_SPI シリアル転送にレガシー, デュアルビット, クワッドビット SPI プロトコルのどれを使うかを決めます。
- シリアル SRAM デバイスが接続されている場合は、HSSPIn_CSCFG:SRAM ビットによりこれらのメモリ割付けされたデバイスへの書込みアクセスを許可することが可能です。シリアルフラッシュデバイスには書込みアクセスを許可してはいけません。
- HSSPIn_CSCFG:MSEL にシリアル転送を始めるメモリデバイスの選択に使用されるオンチップバスアドレス空間を設定してください。アドレス生成の詳細については、「47.5.3. コマンドシーケンサモード」を参照してください。
- 割付けされたメモリのアクセス用に生成されたアドレスで 16GB の拡張メモリ領域を使う場合には、HSSPIn_CSAEXT レジスタの設定が必要となります。アドレス生成の詳細については、「47.5.3. コマンドシーケンサモード」を参照してください。
- HSSPIn_CSITIME:ITIME は、割付けされたメモリへのアクセス全体の性能を向上させるときに使います。これは、直前のシリアルトランザクションを任意の時間続けることによって実現されます。例えば、連続なメモリアドレスへ同じタイプ(読出し/書込み)のアクセスがあるときは、HS_SPI は新たなコマンドやアドレスサイクルを発行することなく、同じシリアル転送を処理します。これによりアクセス時間が減少します。HSSPIn_CSITIME:ITIME を適切なアイドルタイムアウト値に設定してください。
- 読出しコマンドシーケンスレジスタ(HSSPIn_RDCSDC0～7)のリストを、接続されているメモリデバイスへのメモリ読出しコマンドのシーケンスと同じになるよう設定してください。読出しコマンドシーケンスの詳細は、各デバイスのデータシートを参照してください。
- HSSPIn_CSCFG:SRAM ビットでメモリ書込みアクセスを有効にする場合には、書込みコマンドシーケンスレジスタ(HSSPIn_WRCSDC0～7)のリストを、接続されているメモリデバイスのメモリ書込みコマンドのシーケンスと同じになるよう設定してください。書込みコマンドシーケンスの詳細は、各デバイスのデータシートを参照してください。
- 以上で、メモリ割付けデバイスをアクセスするための HS_SPI モジュールの設定は完了です。HSSPIn_MCTRL:MEN ビットで HS_SPI モジュールを有効にしてください。それにより、オンチップバスアクセスと外部メモリとのアドレス割当てを行い、シリアルインタフェース上で読出し、書込みシーケンスの生成が始まります。

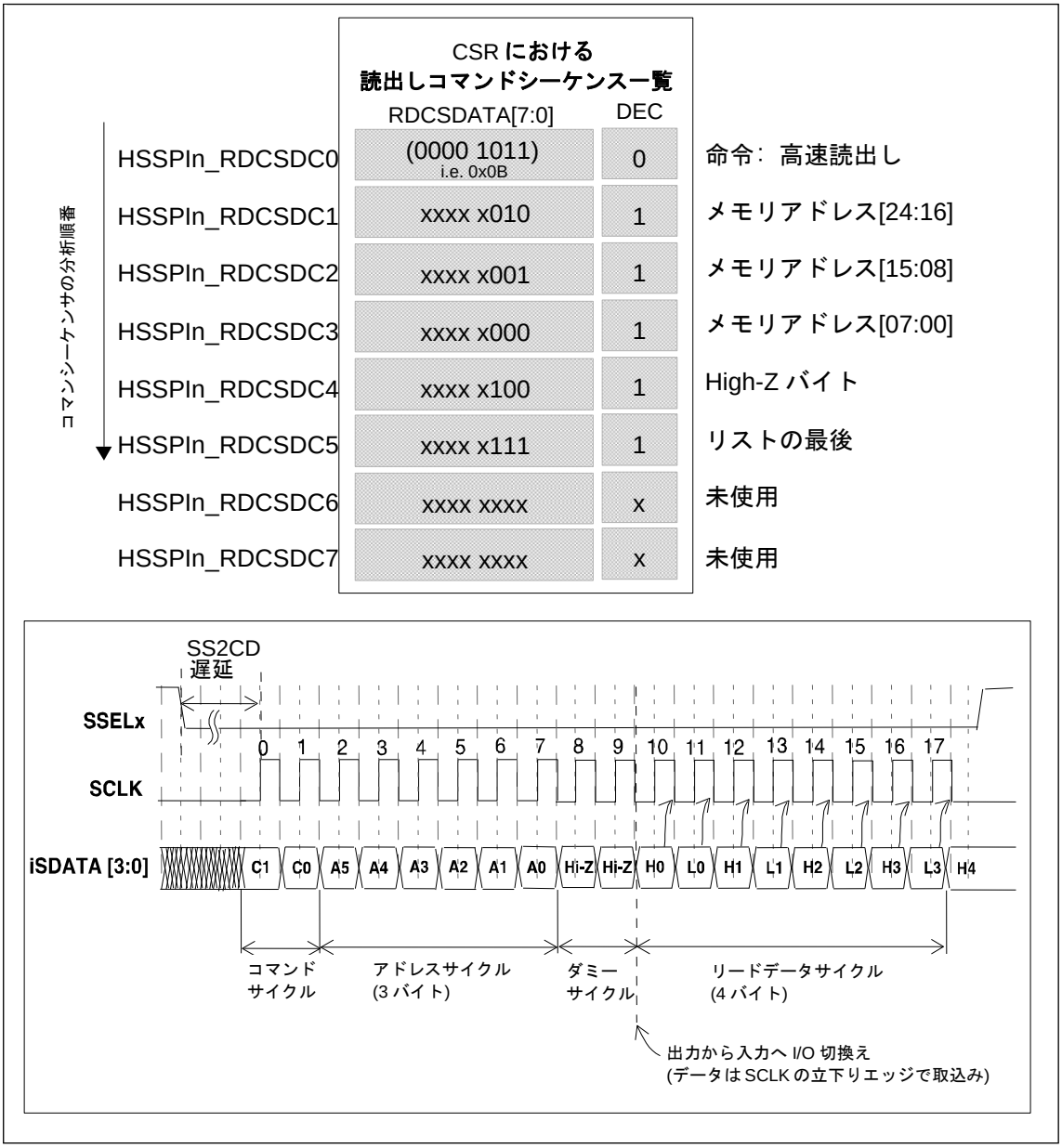
47.6.4.3 コマンドシーケンサのタイミングダイヤグラム

コマンドシーケンサのタイミングダイヤグラムについて示します。

図 47-47 に、コマンドシーケンサのシリアルメモリの読出しコマンドシーケンス生成過程の例を示します。読出しコマンドシーケンスリストは図に示してあるように HSSPIn_RDCSDC0 レジスタから HSSPIn_RDCS5 レジスタまでに設定してあると仮定します。
コマンドシーケンサはリストを分析し、HSSPIn_RDCS0 レジスタから始め、「47.5.3. コマンドシーケンサモード」で説明してあるように命令を実行します。

図 47-47 はクロックモード 0 のときの、読出しコマンドシーケンスに対応するタイミング図です。

図 47-47 読出しコマンドシーケンスの図解 (モード 0):タイミング図付き



47.7 注意事項

HS_SPI ダイレクトモード時のクワッド SPI プロトコルシリアルフラッシュデバイス読出しの注意事項について示します。

47.7.1 概要

概要について示します。

現代の高速シリアルフラッシュデバイスは、クワッド SPI プロトコルシリアルインタフェースを装備しています。HS_SPI はコマンドシーケンサモードだけでなく、ダイレクトモードでもクワッド SPI プロトコル読み出しを開始できます。

図 47-48 に示したように、クワッド SPI プロトコルを使ってシリアルフラッシュデバイスを読み出すには以下の手順にしたがってください。

- レガシープロトコルで命令とアドレスを送ります。
- ダミーサイクルを送ります。その間、シリアルフラッシュデバイスはデータラインの方向を入力から出力に切り替えます。HS_SPI はダミーサイクル中にデータライン(D0--D3) を駆動してはいけません。
- Quad-RX プロトコルで読み出しデータを受け取ります。
- 以上の間、CS ピンはアサートされている必要があります。

HS_SPI のダイレクトモードを使ったときの手順を 2 つの例を使って説明します。

図 47-48 SPI フラッシュクワッド読出しコマンド例

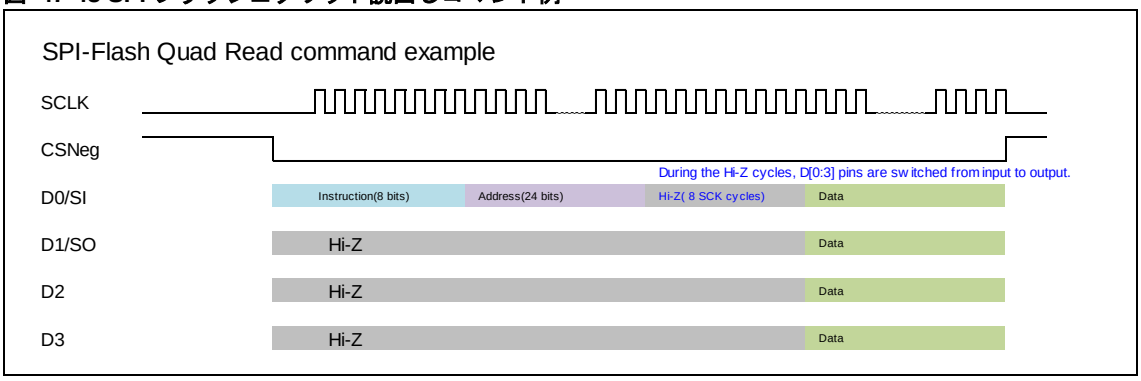


図 47-49 SPI フラッシュメモリとのレガシーSPI プロトコルを使用した接続

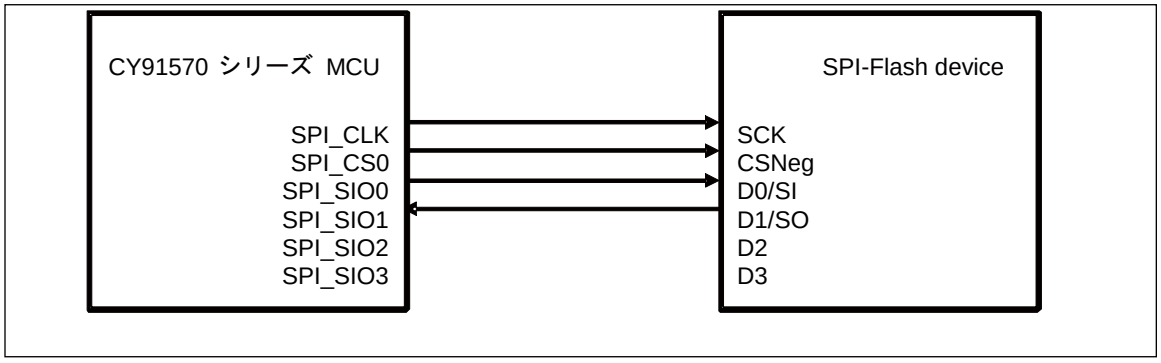
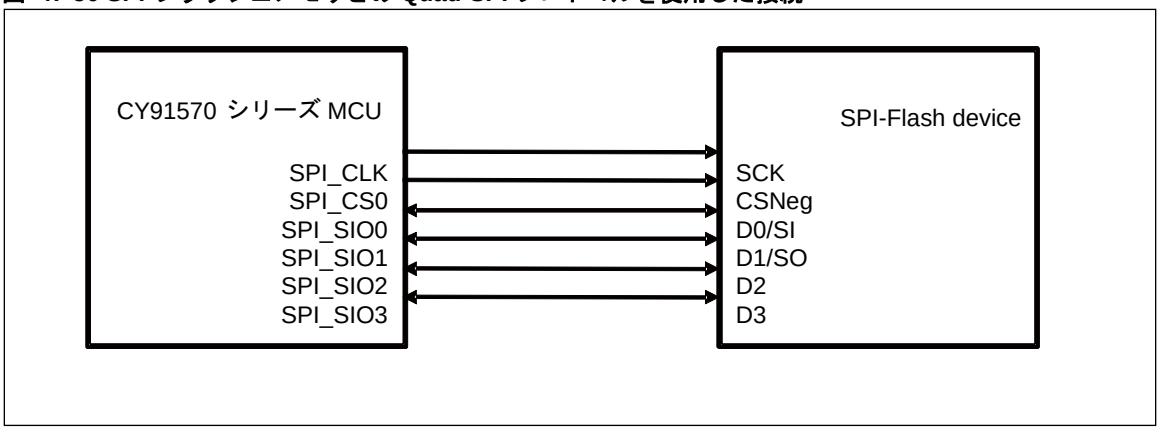


図 47-50 SPI-フラッシュメモリとの Quad-SPI プロトコルを使用した接続

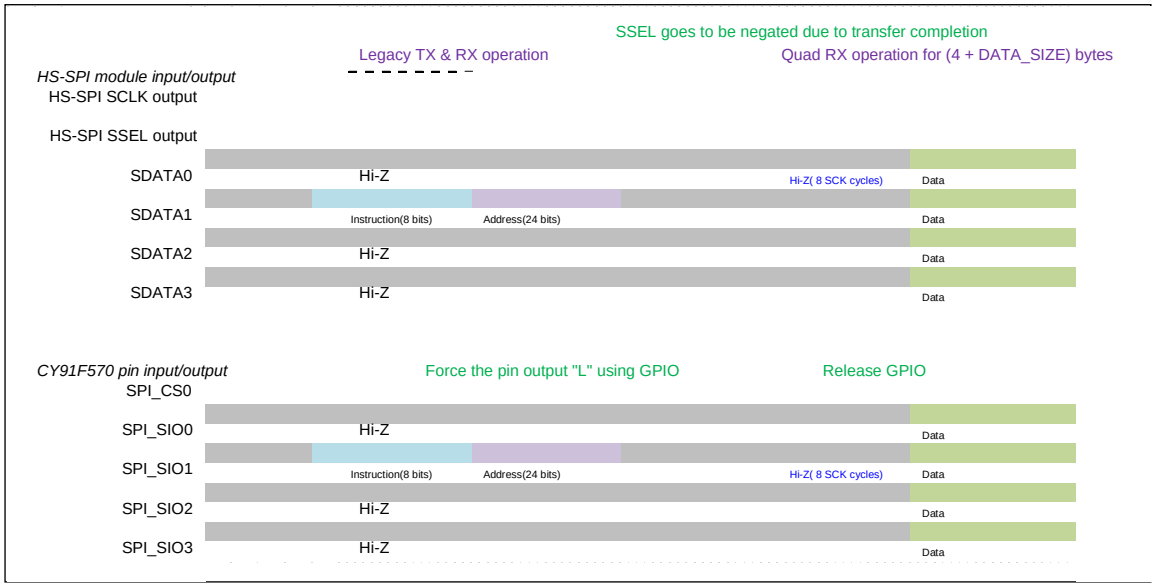


47.7.2 例 1 : 2 つの SPI 転送の開始

例 1 では、図 47-51 に示すように、クワッド読出しコマンドを作るためにレガシープロトコル SPI 転送とクワッド RX プロトコル SPI 転送を開始します。

- TX-FIFO に命令とアドレスをセットします。
  バイトカウンタモード(HSPIDMCFG:SSDC=1)で、4 バイト転送(HSPIDMBCC=4)
- レガシープロトコルで転送を開始します
- レガシープロトコル転送実施中に SPI_CS0 ピン機能を GPIO に切り替えます。これは転送完了時の SSEL 取消しに対する対策のためです。
- レガシープロトコル転送が完了します。
- HSPIDMBCC へ(4 + DATA_SIZE) バイトをセットします。
- Quad-RX プロトコルにおける転送を開始します。
- ソフトウェアは必要に応じて RX-FIFO の内容を取り出します。最初の 8 バイトのデータは命令, アドレス, ダミーサイクルに対応する RX データであるため無視しなければなりません。
- Quad-RX プロトコル転送の間に、HSSPI_CS ピン機能を GPIO から HS-SPI SSEL 出力に戻します。
- Quad-RX プロトコル転送が終了します。

図 47-51 クワッド読出しコマンド例 –SPI 開始と GPIO を使った強制チップ選択

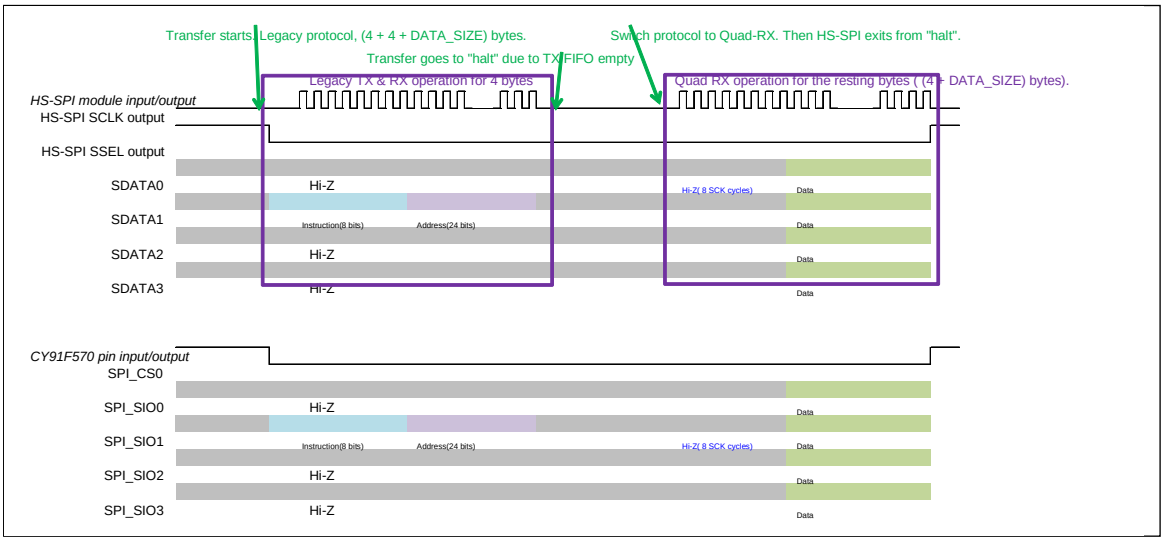


47.7.3 例 2： 転送におけるプロトコル切換え

例 2 では、図 47-52 に示すように、ソフトウェアがレガシープロトコルで SPI 転送を開始し、プロトコルを Quad-RX に切り換えます。

- TX-FIFO に命令とアドレスをセットします。
  バイトカウンタモード(HSPIDMCFG:SSDC=1), (4 + 4 + DATA_SIZE)バイト転送(HSPIDMBCC=(4 + 4 + DATA_SIZE))
- レガシープロトコルにおける転送を開始します。
- レガシープロトコルによる転送は、TX-FIFO のアンダフローにより「停止」になります。「停止」については、「●TX-DATA の不足、または RX-FIFO スペース不足による転送の停止」を参照してください。
- ソフトウェアがプロトコルを Quad-RX に切り換えます。
- 転送を再開します。
- ソフトウェアは必要に応じ RX－FIFO の情報を取り出します。最初の 8 バイトのデータは命令，アドレス，ダミーサイクルに対応する RX データであるため無視しなければなりません。
- Quad-RX プロトコル転送が終了します。

図 47-52 クワッド読出しコマンド例 – プロトコル切換えに「停止」を使用



48. オンチップデバッガ (OCD)

オンチップデバッガ (OCD) について説明します。

48.1 概要

オンチップデバッガ(OCD)の概要について説明します。

本シリーズでのオンチップデバッガ(OCD)の概要ならびに仕様制限を説明します。

OCDU は、FR81 においてオンチップデバッグ機能を提供するためのデバイス内蔵デバッグサポートユニットです。OCDU は、デバッガの基本機能(CPU 実行/ブレーク制御、CPU レジスタ/メモリ/IO のアクセス)、小規模デバッグ支援機能(イベント、実行時間測定、トレースなど)、セキュリティ機能を搭載してします。

48.2 特長

オンチップデバッガ(OCD)の特長について説明します。

- 一線式デバッグツール I/F
- デバッグセキュリティ機能
- デバッグモード制御機能
- 実行制御機能
  - 各種ステータス表示機能(チップステータス、CPU ステータスなど)
  - デバッグコマンド実行制御機能
  - 小規模デバッグメインメモリ(8 バイト=4 命令)
  - CPU レジスタ退避レジスタ(PC/PS)
  - PC モニタ機能
  - リセット機能
    - チップリセット(INIT)
    - CPU リセット(RST)
- ブレーク機能
  - ステップ実行ブレーク
  - イベントトリガブレーク
  - 強制ブレーク
  - ガーデッドアクセスブレーク
  - トレース終了ブレーク
  - 実行開始アドレス直後の割込み受付制御
- デバッグ DMA 機能(DDMA 機能)
  - 各種転送モードのサポート(アドレスモード、Verify モード、DEBUG I/F のバースト転送)
- イベント機能
  - コードイベント：8
  - 条件付コードイベント：2
  - データイベント：8
  - 割込みイベント：2
  - ユーザイベント：2
  - イベントシーケンサ：2 レベル＋リセット
- 実行時間測定タイマ機能
  - Go-Break 間測定
  - トリガ間測定(単発測定/累積測定)
- トレース機能
  - 特殊ステートトレース
  - 分岐トレース
  - データトレース
  - トレースディレイ
  - トレースフレーム数：512

48.3 構成

オンチップデバッガ (OCD) の構成について示します。

図 48-1 OCDU のブロックダイヤグラム

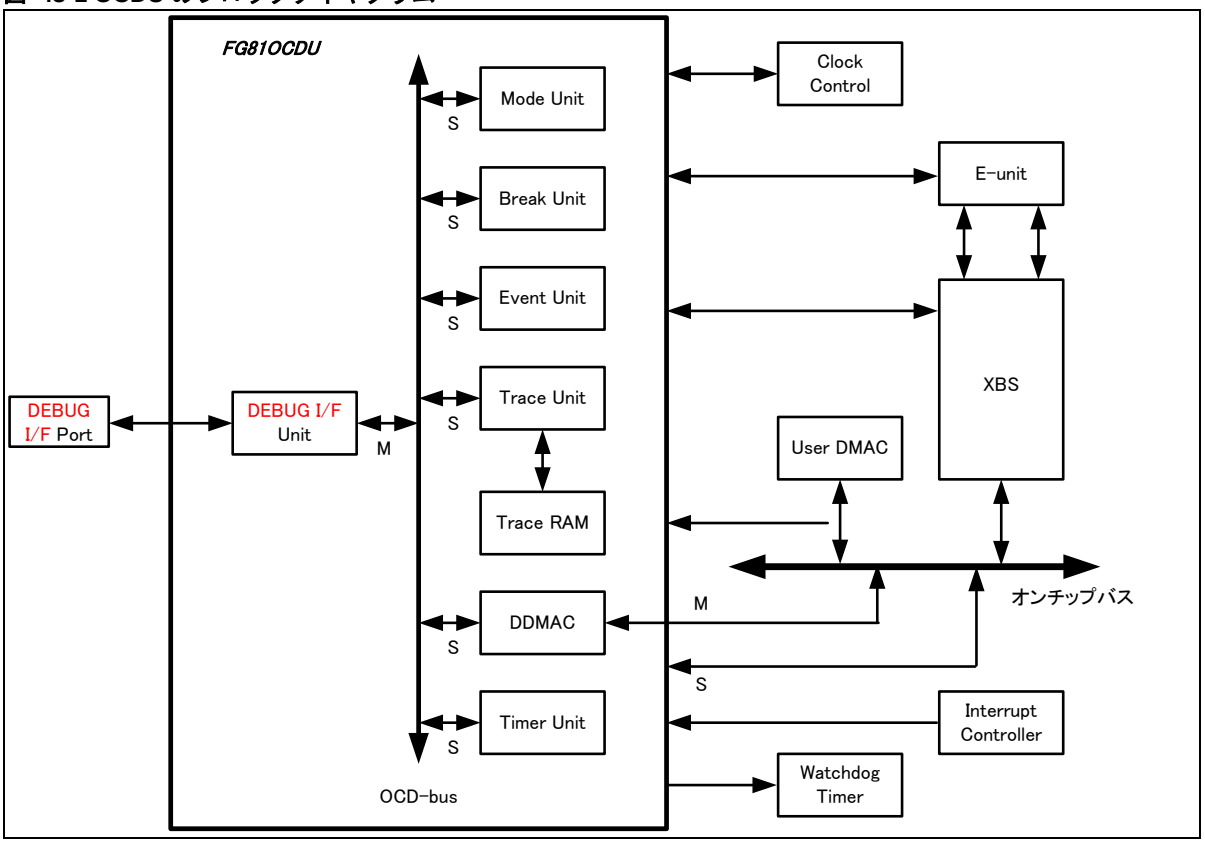
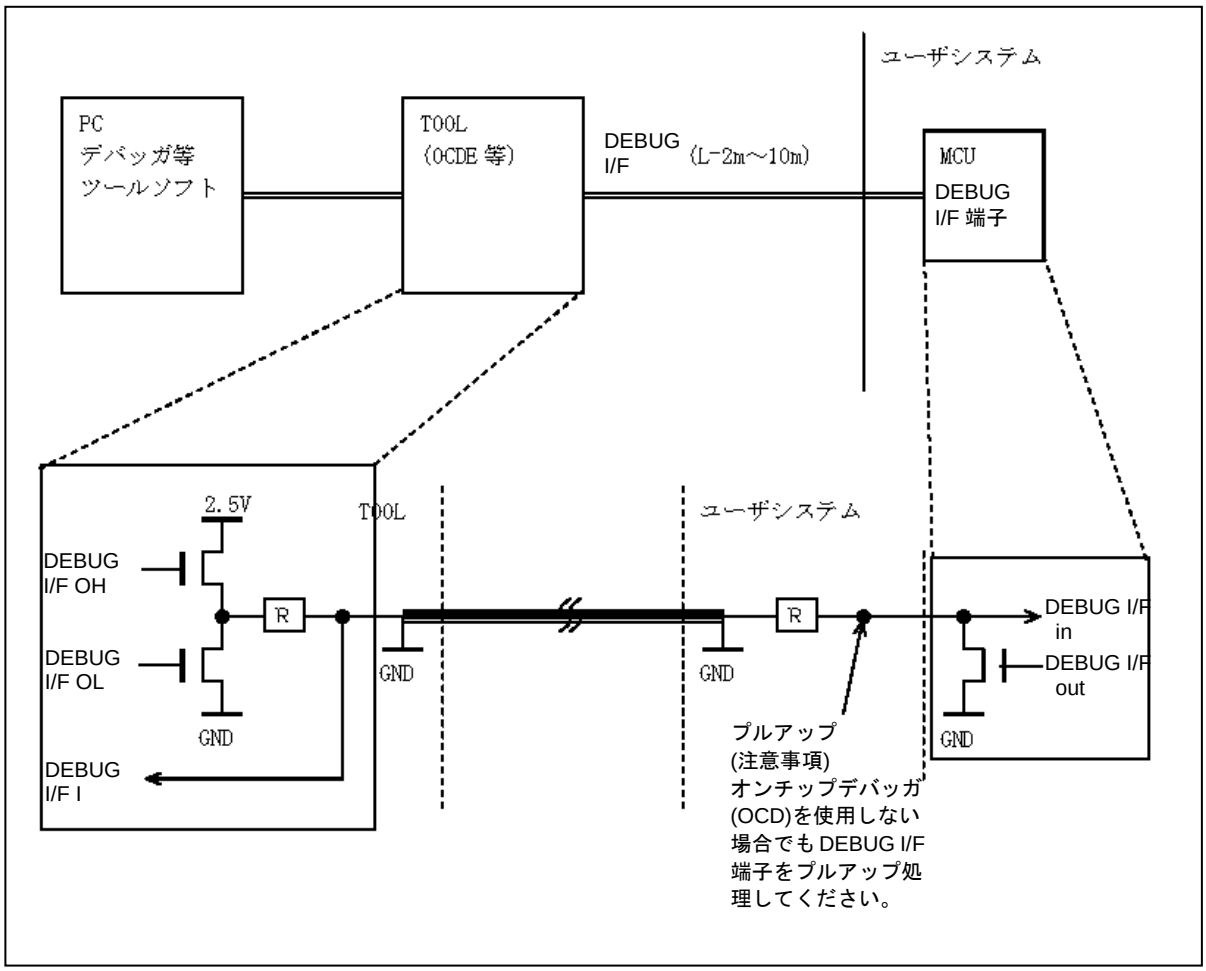


図 48-2 OCD 接続図


ユーザシステム
PC
デバッガ等
ツールソフト
TOOL
(OCDE 等)
DEBUG
I/F
(L=2m～10m)
MCU
DEBUG
I/F 端子
2.5V
TOOL
ユーザシステム
DEBUG
I/F OH
DEBUG
I/F OL
DEBUG
I/F I
R
R
GND
GND
GND
GND
DEBUG I/F
in
DEBUG I/F
out
プルアップ
(注意事項)
オンチップデバッガ
(OCD)を使用しない
場合でも DEBUG I/F
端子をプルアップ処
理してください。

48.3.1 DEBUG I/F 用クロック

DEBUG I/F 用クロックについて示します。

DEBUG I/F 用クロックのクロック接続構成については、『クロック』の章を参照してください。

48.3.1.1 DEBUG I/F 用メインクロック (M_MCLK)

DEBUG I/F 用メインクロック (M_MCLK) について示します。

OCD ツール接続時、DEBUG I/F 用メインクロック (M_MCLK) にはメインクロック (MCLK) が供給されます。

OCD ツール未接続時、DEBUG I/F 用メインクロック (M_MCLK) は停止します。

48.3.1.2 DEBUG I/F 用 PLL クロック (M_PCLK)

DEBUG I/F 用 PLL クロック (M_PCLK) について示します。

OCD ツール接続時で高速 UART モードおよび位相変調 UART モード選択時、DEBUG I/F 用 PLL クロック (M_PCLK) には PLL クロック (PLLCLK) が供給されます。

OCD ツール未接続時、DEBUG I/F 用 PLL クロック(M_PCLK)は停止します。

48.4 レジスタ

オンチップデバッガ(OCD)のレジスタについて説明します。

48.4.1. DBG レジスタ

48.4.2. ユーザ IO レジスタ

48.4.1 DBG レジスタ

DBG レジスタのビット構成について示します。

表 48-1 レジスタマップ (DBG レジスタ)

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0xFF00	DSUCR		予約		DSU 制御レジスタ

■ DSU 制御レジスタ(DSUCR)

フリーランモードにおいて、DSU の制御を行うレジスタです。

詳細は弊社営業担当にお問い合わせください。

● DSUCR：アドレス FF00H (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	X	X	X	X	X	X	X	X
属性	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							DSU
初期値	X	X	X	X	X	X	X	0
属性	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	R,W

48.4.2 ユーザ IO レジスタ

ユーザ IO レジスタのビット構成について示します。

表 48-2 レジスタマップ (ユーザ IO レジスタ)

アドレス	レジスタ				レジスタ機能
	+0	+1	+2	+3	
0x0BFC	予約		UER		ユーザイベントレジスタ

■ ユーザイベントレジスタ (UER)

ユーザイベントの検出を行うレジスタです。

詳細は弊社営業担当にお問い合わせください。

● UER：アドレス 0BFE$_H$ (アクセス: バイト, ハーフワード, ワード)

	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
	予約							
初期値	X	X	X	X	X	X	X	X
属性	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	予約							UEVT
初期値	X	X	X	X	X	X	X	X
属性	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W0	RX,W

48.5 動作説明

オンチップデバッガ(OCD)の動作について説明します。

48.5.1. OCDU 動作モード

48.5.2. DEBUG I/F 概要

48.5.3. 本シリーズの OCD ツール接続時での仕様制限

48.5.4. 本シリーズの OCD-DSU ID コードおよび実装タイプ情報

48.5.1 OCDU 動作モード

OCDU 動作モードについて示します。

48.5.1.1. 動作モードについて

48.5.1.2. 動作モード状態遷移

48.5.1.1 動作モードについて

動作モードについて示します。

OCDU の動作モードにはエミュレータモードとフリーランモードがあります。

■ エミュレータモード(デバッグ走行状態)
エミュレータモードには、デバッグ命令を実行するためのデバッグステートとユーザプログラムを実行するためのユーザステートがある。デバッグステートで RETI 命令が実行されるとユーザステートへ遷移し、ユーザステートでブレークされるとデバッグステートへ遷移する。

■ フリーランモード(通常走行状態)
ユーザプログラムのみが走行するモード。

48.5.1.2 動作モード状態遷移

動作モード状態遷移について示します。

INIT 解除時 (INIT を伴う RST 含む) は、チップリセットシーケンスにおける DEBUG I/F からのモードコマンドにしたがって、エミュレータモードのデバッグステートか、フリーランモードに遷移します。

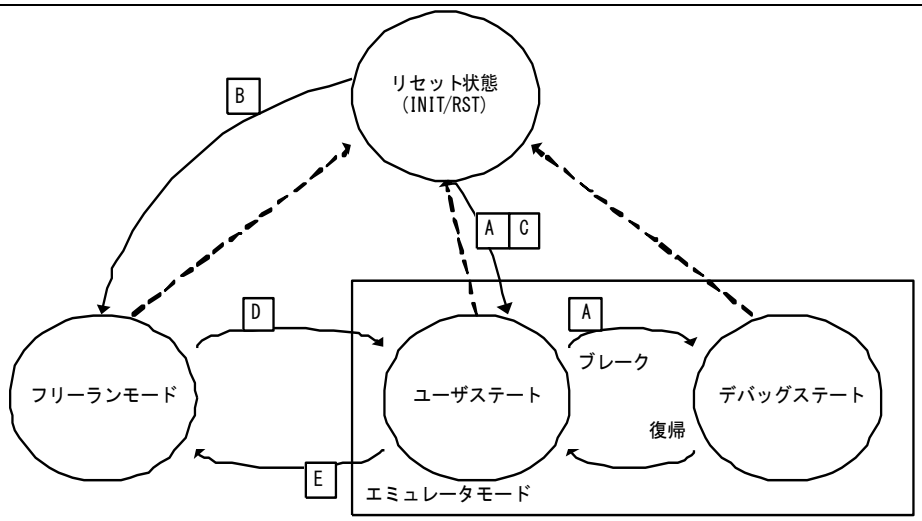
RST 解除時 (INIT は伴わない) は、RST 発生前の動作モードに遷移します。ただし、ユーザステートで RST が発生した後、強制ブレーク要求があった場合は、RST 解除時にエミュレータモードのデバッグステートへ遷移します。

また、OCD レジスタの制御により、フリーランモードとエミュレータモードのユーザステート間の遷移が可能となります。

リセット状態からデバッグステートへ遷移する際、まずユーザステートへ遷移します。この際、OCDU がブレーク要求をすることで、リセット状態 → ユーザステート→ (ブレーク) → デバッグステートへと遷移します。

各遷移条件を以下に示します。

図 48-3 OCDU 動作モード遷移図



[破線の遷移]
リセット（INIT、RST）が発生した場合の遷移。すべての状態からリセット状態へ遷移する。
[実線の遷移]
A:①INIT が伴うリセット発生後、INIT 解除時の DEBUG I/F のチップリセットシーケンスにおいてモードコマンドを正常に受信した場合の遷移。
②デバッグステートで INIT が伴わないリセット発生後、RST 解除時の遷移。
③ユーザステートで INIT が伴わないリセット発生後、RST 解除前に強制ブレーク、または擬似オンザフライブレーク要求があった場合の RST 解除時の遷移。
B:①INIT が伴うリセット発生後、INIT 解除時の DEBUG I/F のチップリセットシーケンスにおいてモードコマンドを正常に受信しなかった場合の遷移。②フリーランモードで INIT が伴わないリセット発生後、RST 解除時の遷移。
C:ユーザステートで INIT が伴わないリセット発生後、RST 解除時の遷移。
D:フリーランモードで OCDU 動作許可のとき、E_MSTSR:DMODE ビットへ"1"書込みした場合の遷移。
E:ユーザステートで、E_MSTSR:DMODE ビットへ"0"書込みした場合の遷移。

48.5.2 DEBUG I/F 概要

DEBUG I/F 概要について示します。

DEBUG I/F はシングルワイヤのデバッグインタフェースであり、MCU とツールを 1 線 (+GND) で接続します。MCU は 1 端子をデバッグインタフェース用に使用します。

DEBUG I/F は、双方向端子であり、通信と特殊シーケンスの機能を有します。通信はシリアル伝送方式 (UART) で、通信のボーレートは通常 UART モードでは MCU のメイン源発振クロックをベースとした分周クロックであり、高速 UART モードおよび位相変調 UART (マンチェスタエンコード UART) では PLL クロックをベースとした分周クロックとします。特殊シーケンスには、チップリセットシーケンスとストールがあります。チップリセットシーケンスでは MCU が INIT 発生を通知する機能と、INIT 解除後に起動するデバッグモードを検出する機能があります。ストール機能には、ツールからの通信ストール要求と強制ブレーク要求、MCU からの通信エラー通知があります。

以下に主な DEBUG I/F の機能を示します。

- チップリセットシーケンス機能(INIT 通知、モードコマンド)
- UART 機能 (通常 UART、高速 UART、位相変調 UART)
- ストール要求 (通信ストール要求、強制ブレーク要求、通信エラー通知)

DEBUG I/F の双方向端子は、N-ch オープンドレイン出力により実現され、DEBUG I/F 端子はユーザシステム上でプルアップされます。ツール接続時は、ツールによりプルアップされます。

ツールの接続は図 48-2 OCD 接続図を参照してください。

48.5.2.1 チップリセットシーケンス

チップリセットシーケンスについて示します。

INIT が発生した場合、OCDU は DEBUG I/F の仕様に従い、チップリセットシーケンスを実行します。チップリセットシーケンスを実行する基準クロックは、通常 UART のサンプリングクロック(メイン源発振クロックの 8 分周クロック)です。

チップリセットシーケンスは、以下の 5 つのフェーズで構成されます。

- 開始フェーズ
- INIT フェーズ
- レベルセンスフェーズ
- モードエントリフェーズ
- 終了フェーズ

● 開始フェーズ

発生した INIT が解除された時点から通常 UART のサンプリングクロック 32 サイクル間は、開始フェーズです。OCDU は、このフェーズで特別な動作を行いません。

● INIT 通知フェーズ

開始フェーズ終了してから通常 UART のサンプリングクロック 480 サイクル間は、INIT 通知フェーズです。OCDU は、このフェーズにいる間、DEBUG I/F に L を出力し、INIT の発生をツールに通知します。

● レベルセンスフェーズ

INIT 通知フェーズ終了から通常 UART のサンプリングクロック 256 サイクル間は、レベルセンスフェーズです。OCDU は、このフェーズで特別な動作を行いません。

● モードエントリフェーズ

レベルセンスフェーズ終了から通常 UART のサンプリングクロック 256 サイクル間は、モードエントリフェーズです。OCDU は、このフェーズからツールからのモードコマンドの受信を開始します。

このフェーズでモードコマンドの受信開始を検出(UART 受信でスタートビットを検出)した場合、OCDU はエミュレータモード(デバッグステート)で起動します。その後、正常なモードコマンド(受信エラーなし、かつ、モードコード一致)を受信すると、これ以降 OCDU は後述するレジスタアクセスコマンドの受付が可能になります。正常なモードコマンド(受信エラーまたはモードコード不一致)を受信しなかった場合、OCDU は INIT 要求を発生させ、INIT 解除後に再度チップリセットシーケンスを実行します。

このフェーズでモードコマンドの受信開始を検出(UART 受信でスタートビットを検出)しなかった場合、OCDU はフリーランモードで起動します。

モードエントリフェーズ開始直後にモードコマンドを受信させる場合、UART 受信サンプリングクロックで 1 サイクル幅以上 DEBUG I/F に H が入力されるのを待って、モードコマンドを受信させる必要があります。これが満足されない場合、モードコマンド受信のスタートビットを正常検出できず、正しくモードエントリできない可能性があります。

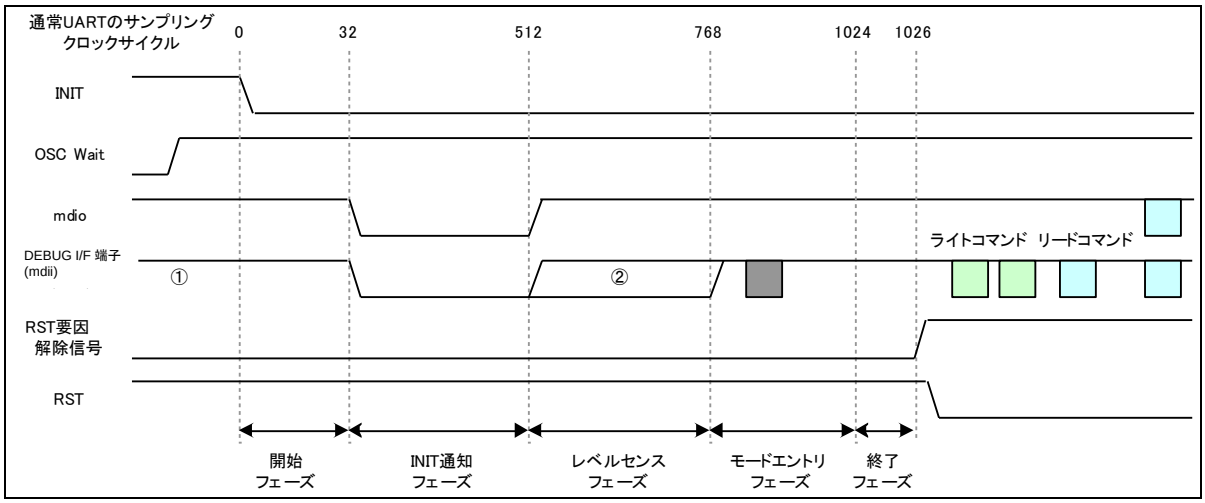
● 終了フェーズ

モードエントリフェーズ終了から通常UARTのサンプリングクロック2サイクル間は、終了フェーズです。OCDUはこのフェーズで特別な動作を行いません。OCDU は終了フェーズが終わった時点で『CHAPTER 7: リセット』の章の『7.5.4.3 リセット(RST)』記載のリセット発行シーケンスを実行します。RST 要因を解除します。

チップリセットシーケンスの通常 UART のサンプリングクロックのサイクル数とフェーズの関係は次のようになります。

チップリセットシーケンスのフェーズ	開始フェーズ	INIT通知フェーズ	レベルセンスフェーズ	モードエントリフェーズ	終了フェーズ
INIT 解除からの通常 UART のサンプリングクロックサイクル	1 - 32	33 - 512	513 - 768	769 - 1024	1025, 1026

以下に、チップリセットシーケンスを示します。



OSC Wait ：メイン源発振クロックの発振安定。INIT は、発振安定確認後に解除される。
① ：ツールのプルアップにより、DEBUG I/F は H レベルとなる。
② ：DEBUG I/F はユーザシステム上のプルアップのレベルとなる。

48.5.2.2 セキュリティ機能

セキュリティ機能について示します。

OCDU にはセキュリティ機能を搭載しています。OCDU は、CPU のメモリ空間にあるデバッグセキュリティ領域に記憶されるセキュリティ情報の設定により、セキュリティ機能を有効にすることができます。セキュリティ機能を有効にすると、セキュリティロック状態となります。これを解除するために、セキュリティ情報で設定したパスワードを指定長回数、E_SLPR レジスタに書き込むことで、セキュリティをアンロックできます。

■ セキュリティ情報

デバッグセキュリティ領域は、搭載する Flash の先頭アドレス+4 から+33 の 30 バイトに配置されます。OCDU はセキュリティシーケンスで、この領域を参照します。

デバッグセキュリティ領域には、以下のセキュリティ情報があります。

- セキュリティパスワード長 (PW 長)
  デバッグセキュリティ領域の先頭アドレスにある 16bit 長データであり、下位 4bit が有効な PW 長です。上位 12bit は動作に影響がありません。PW 長が 0x0 または 0xF の場合、セキュリティは無効です。PW 長が 0x1～0xE (1～14) の場合、セキュリティは有効です。

- セキュリティパスワード (PW)
  デバッグセキュリティ領域にある 16bit 長のデータで、1～14 個あります。PW 長アドレスの次のアドレスから順に、PW1、PW2、･･･PW14 と配置されます(下図参照)。セキュリティが有効(PW 長 : 1～14)の場合、PW 長の値が有効な PW を示します。
  (例えば、PW 長が 8 の場合、PW1～PW8 が有効、PW9～PW14 が無効。)

アドレス	15 … 0
ROM/Flash 先頭アドレス +4	PW 長
ROM/Flash 先頭アドレス +6	PW1
ROM/Flash 先頭アドレス +8	PW2
･･･	･･･
ROM/Flash 先頭アドレス +32	PW14

＜注意事項＞

オンチップデバッガ(OCD)のセキュリティ機能を使用しない場合は、この領域には何も書き込まず、フラッシュ消去直後の初期状態(全 bit=1)のままとしてください。

48.5.3 本シリーズの OCD ツール接続時での仕様制限

本シリーズの OCD ツール接続時での仕様制限について示します。

OCD ツール接続時には、次の制限が発生します。

48.5.3.1 クロック設定

クロック設定について示します。

OCD 高速 UART ならびに位相変調 UART 通信時には、PLL は継続して発振しているため、次の PLL 設定レジスタの変更は有効になりません。ただし OCD ツール未接続時と同様に、書込みならびに読出しが可能です。

- PLLCR.ODS
- PLLCR.PMS
- PLLCR.PDS
- CCPSDIVR.PODS
- CCPLLFBR.IDIV

48.5.3.2 スタンバイモード

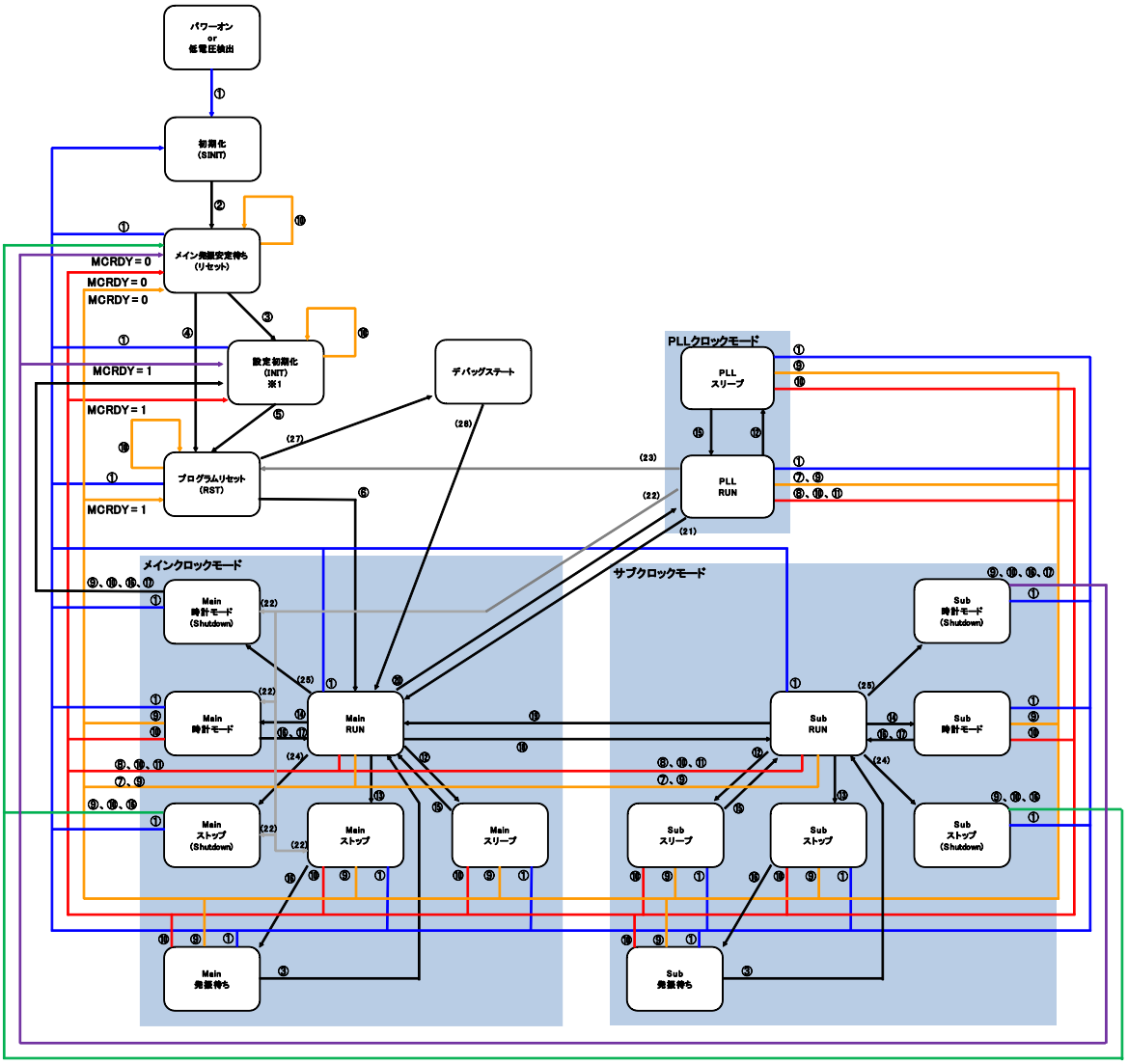
スタンバイモードについて示します。

- 時計モードに移行しても、OCD 高速 UART ならびに位相変調 UART 通信許可時には PLL は発振停止しません。
- ストップモードに移行しても、メインクロックは発振停止しません。OCD 高速 UART ならびに位相変調 UART 通信許可時には PLL は発振停止しません。サブクロック発振時は、サブクロックは停止せずに発振継続します。
- 上記制限による、OCD ツール未接続時と動作の違いのある機能を示します。
  - ☐ PLL 停止時(CSELR:PCEN=0)、または OCD 高速 UART ならびに位相変調 UART 通信許可時は CAN は時計モード・ストップモード時でも動作継続します。(CPU による処理のない範囲での動作です。)
  - ☐ LCD コントローラはストップモード時には時計モード時と同様の動作をします。
  - ☐ リアルタイムクロックはストップモード時でも動作継続します。
  - ☐ RTC/WDT1 補正(キャリブレーション)のカウンタ動作はストップモード時でも継続します。
- 次の機能は、上記制限を受けずに OCD ツール未接続時と同様の動作になります。
  - ☐ メインタイマ、サブタイマはストップモード時にクリアされる構造になっているため、ストップモード時は動作しません。
- PLL クロック発振継続のため、時計モード時の消費電力は OCD ツール未接続時と比べて大きくなります。
- PLL クロック、メインクロック、サブクロック発振継続のため、ストップモード時の消費電力は OCD ツール未接続時と比べて大きくなります。

48.5.3.3 クロック・リセット状態遷移

クロック・リセット状態遷移について示します。

図 48-4 デバイス状態



① パワーオンリセット または 内部低電圧検出 または 外部リセットと NMI の同時アサート
② パワーオンリセット解除 かつ 内部低電圧解除 かつ 外部リセットと NMI の同時アサート解除
③ 発振安定待ち終了
④ 発振安定待ち終了(リセット要因が⑦、⑨の場合)
⑤ INIT解除
⑥ RST解除
⑦ ソフトリセット
⑧ ソフトウォッチドッグリセット(イレギュラーを含む) or ソフトリセット(イレギュラー)
⑨ 外部リセット入力(NMI 無効) or 外部低電圧検出
⑩ 外部リセット入力(NMI 無効＋イレギュラー) or 外部低電圧検出(イレギュラー)
⑪ ハードウォッチドッグリセット(イレギュラーを含む)
⑫ スリープモード(命令書き込み)
⑬ ストップモード(命令書き込み)
⑭ 時計モード(命令書き込み)
⑮ 割込み(⑱、⑰も含む)
⑯ 割込み(クロックを必要としない)/NMI
⑰ メインタイマ割込み/サブタイマ割込み/RTC割込み
⑱ メイン → サブ切り替え(命令書き込み)
⑲ サブ → メイン切り替え(命令書き込み)
⑳ メイン → PLL切り替え(命令書き込み)
(21) PLL → メイン切り替え(命令書き込み)
(22) 不正スタンバイモード移行
(23) 不正スタンバイモード移行検出リセット
(24) ストップモードかつShutDown(命令書き込み)
(25) 時計モードかつShutDown(命令書き込み)

(27) RST解除(設定初期化(INIT)を伴いチップリセットシーケンスにおいてモードコマンドを正常に受信した場合 or 強制ブレーク要求があった場合)
(28) ブレークリターン
※ (26) は欠番です。

*1: 時計モード(Shutdown)からの復帰時とストップモード(Shutdown)からの復帰時には、リセットされないレジスタがあります。詳細は、『消費電力制御』の章の『電源遮断・通常スタンバイ制御の制限事項』を参照してください。

＜注意事項＞

1 系統品種ではサブクロック入力がないため、サブクロックモードには遷移しません。

48.5.3.4 仕様制限まとめ

仕様制限まとめについて示します。

1) 通信モード(*1)：通常 UART
(注意事項) シャットダウンスタンバイモードのデバッグを行う場合、OCD ツール未接続で行なってください。

リセット要因	OCD ツール未接続時との差		備考
	初期化範囲	処理時間	
パワーオンリセット	なし	あり	リセット解除後エミュレータモード(デバッグステート)へ遷移
RSTX 端子入力(イレギュラー)			
RSTX 端子入力			降圧回路切替え安定待ち時間(*2)なし Main/Sub ストップモード、Main/Sub 時計モードからの復帰のみ
RSTX 端子入力(+NMIX 端子入力)			リセット解除後エミュレータモード(デバッグステート)へ遷移
ウォッチドッグリセット 0(イレギュラー)			
ウォッチドッグリセット 0			
ウォッチドッグリセット 1(イレギュラー)			
ウォッチドッグリセット 1			
外部低電圧検出リセット(イレギュラー)			
外部低電圧検出リセット			降圧回路切替え安定待ち時間(*2)なし Main/Sub ストップモード、Main/Sub 時計モードからの復帰のみ
不正スタンバイモード移行検出リセット(イレギュラー)			リセット解除後エミュレータモード(デバッグステート)へ遷移
不正スタンバイモード移行検出リセット		なし	
内部低電圧検出リセット		あり	リセット解除後エミュレータモード(デバッグステート)へ遷移
フラッシュセキュリティ違反リセット(イレギュラー)			
フラッシュセキュリティ違反リセット		なし	
ソフトウェアリセット(イレギュラー)		あり	リセット解除後エミュレータモード(デバッグステート)へ遷移
ソフトウェアリセット		なし	

割込み要因	OCD ツール未接続時との処理時間差	備考
すべての割込み	あり	降圧回路切替え安定待ち時間(*2)なし Main/Sub ストップモード、Main/Sub 時計モードからの復帰のみ

リセット関連以外のデバイス状態	OCD ツール未接続時との動作差異	備考
Main RUN/Main スリープモード	なし	
PLL RUN/PLL スリープモード		
Sub RUN/sub スリープモード		
Main/sub ストップモード	あり	降圧回路固定 メイン発振継続 サブ発振継続 動作継続(LCD コントローラ、リアルタイムクロック、RTC/WDT1 補正カウンタ動作)
Main/sub 時計モード		降圧回路固定 メイン発振継続 Sub 時計モード

2) 通信モード(*1)：高速 UART/位相変調 UART

リセット要因	OCD ツール未接続時との差		備考
	初期化範囲	処理時間	
パワーオンリセット	なし	あり	リセット解除後エミュレータモード(デバッグステート)へ遷移
RSTX 端子入力（イレギュラー）			
RSTX 端子入力			降圧回路切替え安定待ち時間(*2)なし Main/Sub ストップモード、Main/Sub 時計モードからの復帰のみ
RSTX 端子入力(+NMIX 端子入力)			リセット解除後エミュレータモード(デバッグステート)へ遷移
ウォッチドッグリセット 0（イレギュラー）			
ウォッチドッグリセット 0			
ウォッチドッグリセット 1（イレギュラー）			
ウォッチドッグリセット 1			
外部低電圧検出リセット(イレギュラー)			
外部低電圧検出リセット			降圧回路切替え安定待ち時間(*2)なし Main/Sub ストップモード、Main/Sub 時計モードからの復帰のみ
不正スタンバイモード移行検出リセット（イレギュラー）	なし	あり	リセット解除後エミュレータモード(デバッグモード)へ遷移
不正スタンバイモード移行検出リセット		なし	
内部低電圧検出リセット フラッシュセキュリティ違反リセット（イレギュラー）	なし	あり	リセット解除後エミュレータモード(デバッグステート)へ遷移
フラッシュセキュリティ違反リセット		なし	
ソフトウェアリセット（イレギュラー）		あり	リセット解除後エミュレータモード(デバッグステート)へ遷移
ソフトウェアリセット		なし	

割込み要因	OCD ツール未接続時との処理時間差	備考
すべての割込み	あり	降圧回路切替え安定待ち時間(*2)なし Main/Sub ストップモード、Main/Sub 時計モードからの復帰のみ

リセット関連以外のデバイス状態	OCD ツール未接続時との動作差異	備考
Main RUN/main スリープモード	Yes	PLL 発振継続
PLL RUN/PLL スリープモード	なし	
Sub RUN/sub スリープモード	あり	メイン発振継続 PLL 発振継続
Main/sub ストップモード		降圧回路固定 メイン発振継続 サブ発振継続 PLL 発振継続(不正スタンバイモード移行検出無効) 動作継続(CAN、LCD コントローラ、リアルタイムクロック、RTC/WDT1 補正カウンタ動作)
Main/sub 時計モード		降圧回路固定 メイン発振継続 Sub 時計モード 動作継続(CAN)

*1: 通信モード設定に関しては『SOFTUNE Workbench 操作マニュアル』を参照してください。
*2: 降圧回路安定待ち時間 約 6µs

48.5.4 本シリーズの OCD-DSU ID コードおよび実装タイプ情報

本シリーズの OCD-DSU ID コードおよび実装タイプ情報について示します。

表 48-3 本シリーズの OCD-DSU ID コード

ID 名称	bit 幅	対応 ID レジスタ名称	OCD 空間でのアドレス	値	備考
メーカ ID	16	E_IDMCR	0x000	0x0400	サイプレスコード
CPU ファミリ ID	16	E_IDFCR	0x001	0x0200	FR81E/FR81S
DSU タイプ ID	8	E_IDVCR	0x003	0x06	
DSU バージョン ID	4	E_IDVCR	0x003	0x2	
デバイス ID	16	E_IDDCR	0x002	0x0018	CY91F575/7
				0x0019	CY91F578/9
デバイス バージョン ID	4	E_IDVCR	0x003	0x0	CY91F575/7
				0x1	CY91F578/9

表 48-4 本品種の実装タイプ情報

品種名	コードイベント数	データイベント数	データイベント(大小比較)	シーケンサイベント	トレース
CY91F577 CY91F575	8	8	○	○	512 フレーム

49. 付録

付録について示します。

49.1. メモリマップ

49.2. I/O マップ

49.3. 割込みベクター覧

49.4. CPU 状態における端子状態

49.5. 主な変更内容

49.1 メモリマップ

メモリマップについて示します。

CY91F578/9 のメモリマップはデータシートを参照してください。

図 A-1 メモリマップ CY91F577

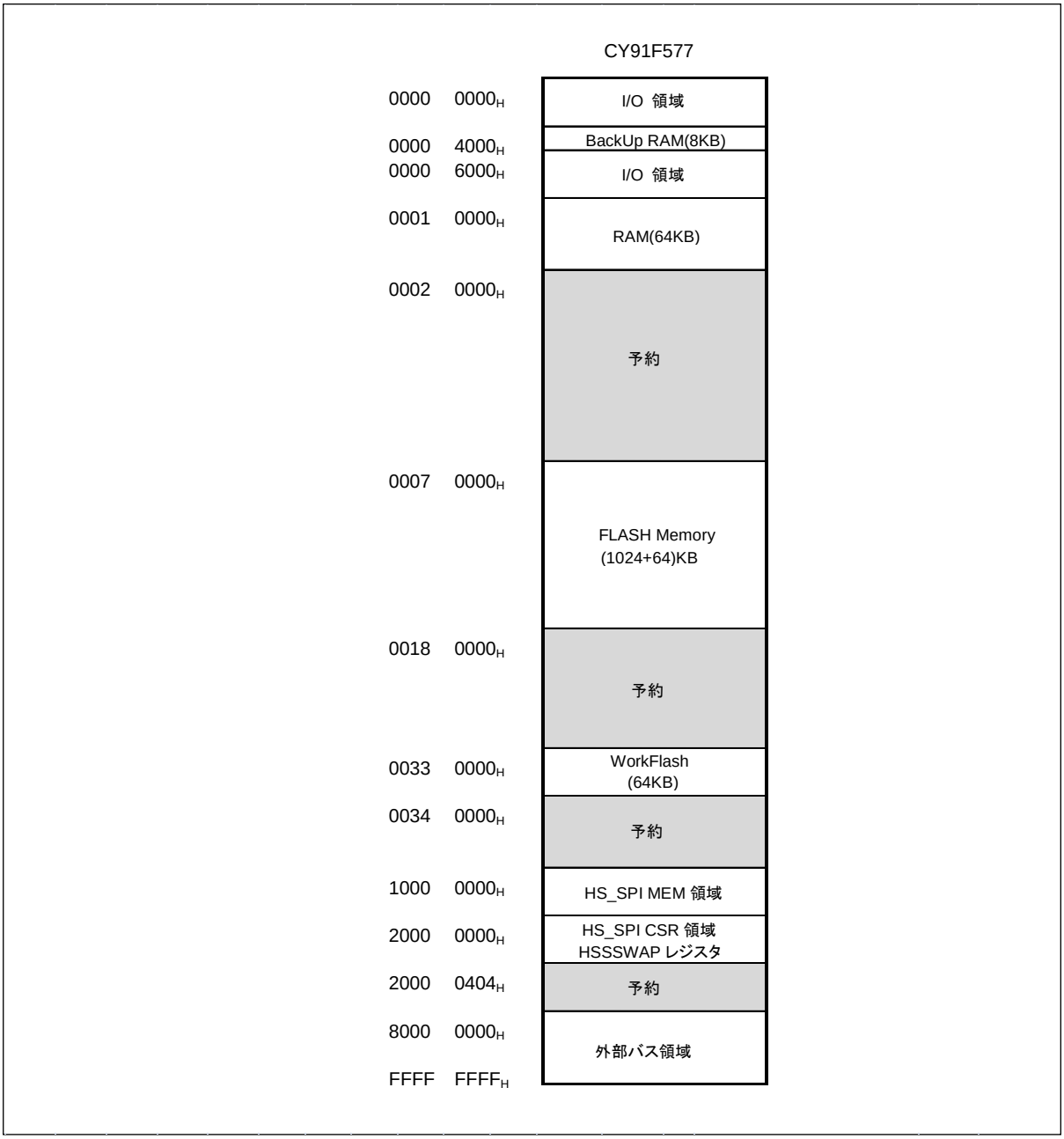
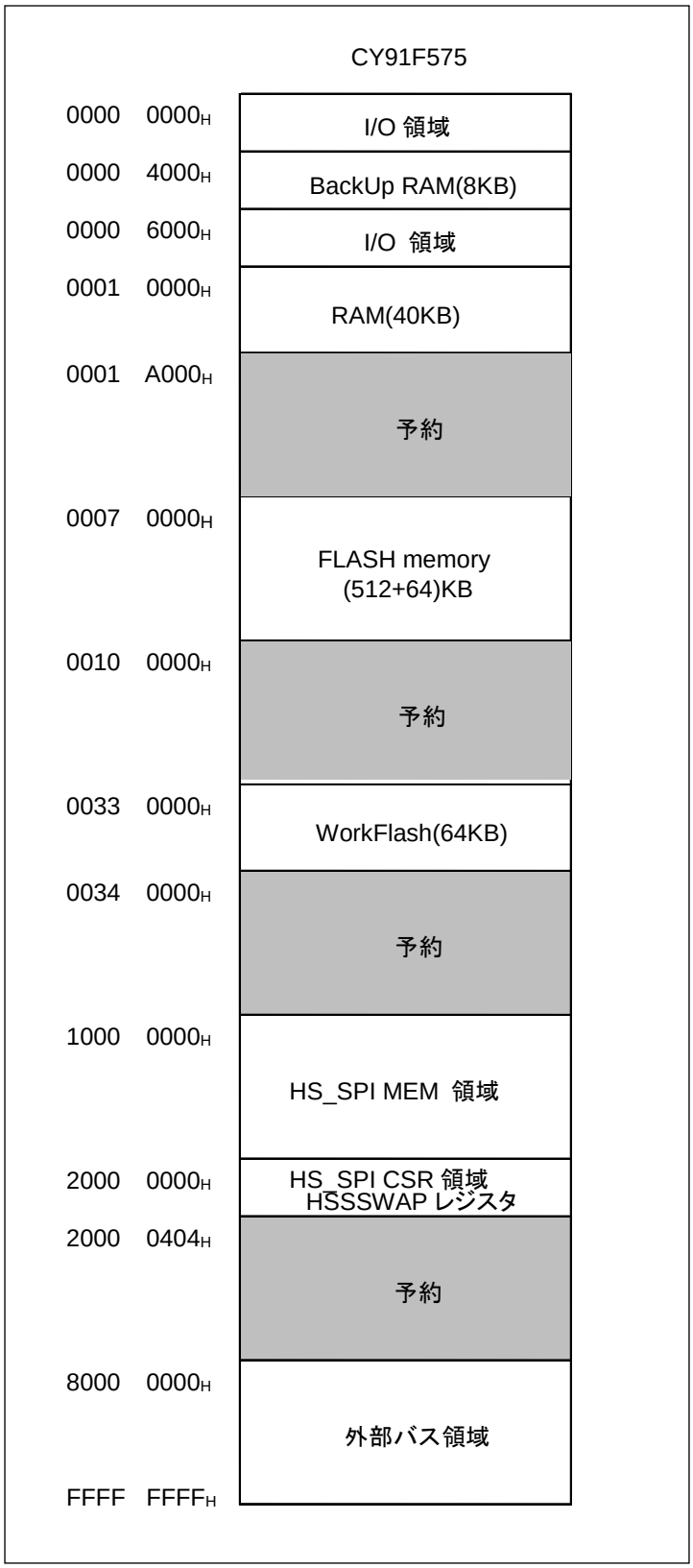


図 A-2 メモリマップ CY91F575



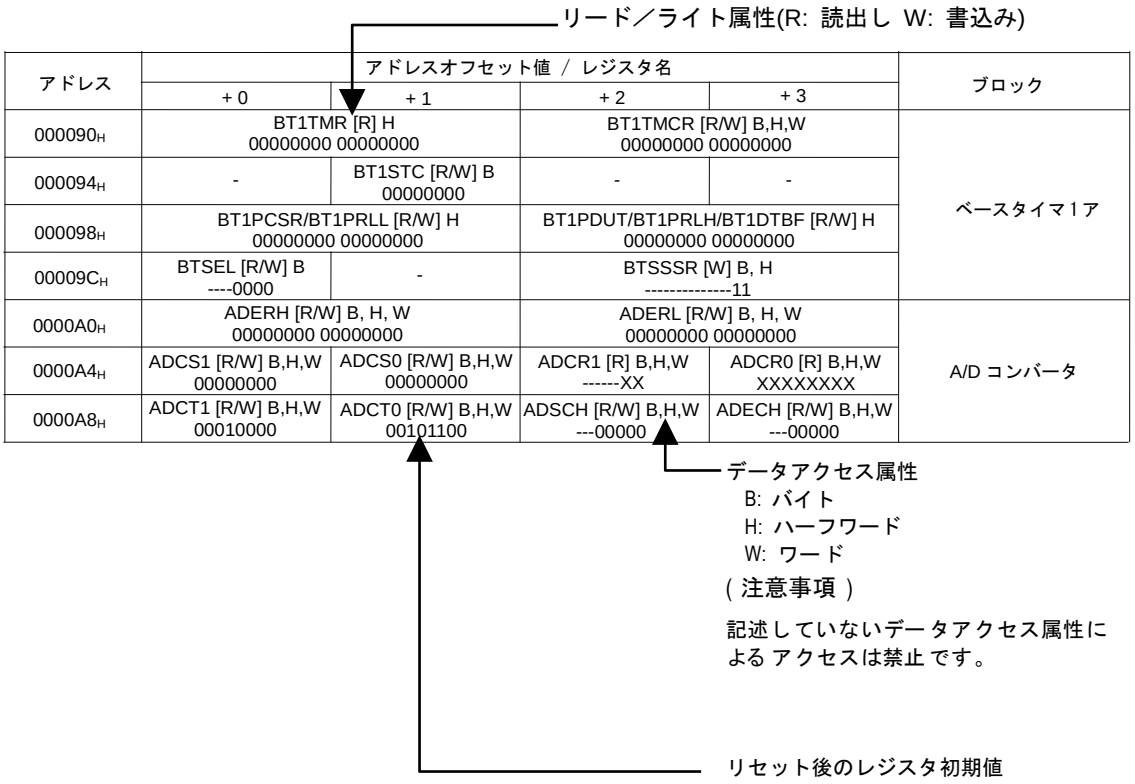
49.2 I/O マップ

IO マップについて示します。

メモリ空間と周辺リソースの各レジスタの対応を示します。

CY91F578/9 の IO マップはデータシートを参照してください。

図 B-1 I/O マップの見方



リセット後のレジスタ初期値の表記の意味を以下に示します。

- "1": 初期値"1"
- "0": 初期値"0"
- "X": 初期値不定
- "-": 予約ビット/未定義ビット
- "*":設定により初期値が"0"または"1"に変化

＜注意事項＞

記載のないアドレスへのアクセスは禁止です。

表 B-1 : I/O マップ (CY91F575/577)

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000000_H	PDR00[R/W] B,H,W XXXXXXXX	PDR01[R/W] B,H,W XXXXXXXX	PDR02[R/W] B,H,W XXXXXXXX	PDR03[R/W] B,H,W XXXXXXXX	ポートデータレジスタ
000004_H	PDR04[R/W] B,H,W XXXXXXXX	PDR05[R/W] B,H,W XXXXXXXX	PDR06[R/W] B,H,W XXXXXXXX	PDR07[R/W] B,H,W XXXXXXXX	
000008_H	PDR08[R/W] B,H,W XXXXXXXX	PDR09[R/W] B,H,W XXXXXXXX	PDR10[R/W] B,H,W XXXXXXXX	PDR11[R/W] B,H,W XXXXXXXX	
$00000C_H$	PDR12[R/W] B,H,W XXXXXXXX	PDR13[R/W] B,H,W XX-XXXXX	—	—	
000010_H ～ 000038_H	—	—	—	—	予約
$00003C_H$	WDTCR0[R/W] B,H,W -0--0000	WDTCPR0[W] B,H,W 00000000	WDTCR1[R] B,H,W ----0110	WDTCPR1[W] B,H,W 00000000	ウォッチドッグタイマ[S]
000040_H	—	—	—	—	予約
000044_H	DICR [R/W] B -------0	—	—	—	遅延割込み
000048_H	TMRLRA4 [R/W] H XXXXXXXX XXXXXXXX		TMR4 [R] H XXXXXXXX XXXXXXXX		リロードタイマ 4
$00004C_H$	TMRLRB4 [R/W] H XXXXXXXX XXXXXXXX		TMCSR4 [R/W] B, H,W 00000000 0-000000		
000050_H	TMRLRA5 [R/W] H XXXXXXXX XXXXXXXX		TMR5 [R] H XXXXXXXX XXXXXXXX		リロードタイマ 5
000054_H	TMRLRB5 [R/W] H XXXXXXXX XXXXXXXX		TMCSR5 [R/W] B, H,W 00000000 0-000000		
000058_H	TMRLRA6 [R/W] H XXXXXXXX XXXXXXXX		TMR6 [R] H XXXXXXXX XXXXXXXX		リロードタイマ 6
$00005C_H$	TMRLRB6 [R/W] H XXXXXXXX XXXXXXXX		TMCSR6 [R/W] B, H,W 00000000 0-000000		
000060_H	TMRLRA0 [R/W] H XXXXXXXX XXXXXXXX		TMR0 [R] H XXXXXXXX XXXXXXXX		リロードタイマ 0
000064_H	TMRLRB0 [R/W] H XXXXXXXX XXXXXXXX		TMCSR0 [R/W] B, H,W 00000000 0-000000		
000068_H ～ $00007C_H$	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000080H	BT0TMR[R] H 00000000 00000000		BT0TMCR[R/W]H -0000000 00000000		ベースタイマ 0
000084H	—	BT0STC[R/W] B 0000-000	—	—	
000088H	BT0PCSR/BT0PRLL[R/W] H XXXXXXXX XXXXXXXX		BT0PDUT/BT0PRLH/BT0DTBF [R/W] H XXXXXXXX XXXXXXXX		
00008CH	—	—	—	—	予約
000090H	BT1TMR[R] H 00000000 00000000		BT1TMCR[R/W]H -0000000 00000000		ベースタイマ 1
000094H	—	BT1STC[R/W] B 0000-000	—	—	
000098H	BT1PCSR/BT1PRLL[R/W] H 00000000 00000000		BT1PDUT/BT1PRLH/BT1DTBF [R/W] H 00000000 00000000		
00009CH	BTSEL01[R/W] B ----0000	—	BTSSSR[W] B,H -------- ------11		ベースタイマ 0,1
0000A0H	ADERH [R/W]B, H, W 00000000 00000000		ADERL [R/W]B, H, W 00000000 00000000		A/D コンバータ
0000A4H	ADCS1 [R/W] B, H,W 0000000-	ADCS0 [R/W] B, H,W 000-----	ADCR1 [R] B, H,W ------XX	ADCR0 [R] B, H,W XXXXXXXX	
0000A8H	ADCT1 [R/W] B, H,W 00010000	ADCT0 [R/W] B, H,W 00101100	ADSCH [R/W] B, H,W --000000	ADECH [R/W] B, H,W --000000	
0000ACH	—	EADERLL [R/W] B, H,W 00000000	EADCS [R] B, H,W --000000	—	
0000B0H	SCR0/(IBCR0) [R/W] B,H,W 0--00000	SMR0 [R/W] B,H,W 000-0000	SSR0 [R/W] B,H,W 0-000011	ESCR0/(IBSR0) [R/W] B,H,W -0000000	Multi-UART0 *1 下位 8 ビットにアクセスする場合のみ、バイトアクセス可能 *2 リセット直後は I²C モードでないため予約
0000B4H	RDR0/(TDR0)[R/W] B,H,W *1 -------0 00000000		BGR0 [R/W] H,W 00000000 00000000		
0000B8H	— / (ISMK0) [R/W] B,H,W -------- *2	— / (ISBA0) [R/W] B,H,W -------- *2	—	—	
0000BCH	FCR10 [R/W] B,H,W ---00100	FCR00 [R/W] B,H,W -0000000	FBYTE20 [R/W] B,H,W 00000000	FBYTE10 [R/W] B,H,W 00000000	

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0000C0H	SCR1/(IBCR1) [R/W] B,H,W 0--00000	SMR1 [R/W] B,H,W 000-0000	SSR1 [R/W] B,H,W 0-000011	ESCR1/(IBSR1) [R/W] B,H,W -0000000	Multi-UART1 *1 下位8ビットにアクセスする場合のみ、バイトアクセス可能 *2 リセット直後はI²Cモードでないため予約
0000C4H	RDR1/(TDR1)[R/W] B,H,W *1 -------0 00000000		BGR1 [R/W] H,W 00000000 00000000		
0000C8H	— / (ISMK1) [R/W] B,H,W -------- *2	— / (ISBA1) [R/W] B,H,W -------- *2	—	—	
0000CCH	FCR11 [R/W] B,H,W ---00100	FCR01 [R/W] B,H,W -0000000	FBYTE21 [R/W] B,H,W 00000000	FBYTE11 [R/W] B,H,W 00000000	
0000D0H	SCR2 [R/W] B, H, W 00000000	SMR2 [R/W] B, H, W 00000000	SSR2 [R/W] B, H, W 00001000	RDR2 /TDR2 [R/W] B, H, W 00000000	LIN-UART2
0000D4H	ESCR2 [R/W] B, H, W 00000X00	ECCR2 [R/W] B, H, W -0000-XX	BGR2 [R/W] B, H, W -0000000 00000000		
0000D8H	SCR3 [R/W] B, H, W 00000000	SMR3 [R/W] B, H, W 00000000	SSR3 [R/W] B, H, W 00001000	RDR3 /TDR3 [R/W] B, H, W 00000000	LIN-UART3
0000DCH	ESCR3 [R/W] B, H, W 00000X00	ECCR3 [R/W] B, H, W -0000-XX	BGR3 [R/W] B, H, W -0000000 00000000		
0000E0H	SCR4 [R/W] B, H, W 00000000	SMR4 [R/W] B, H, W 00000000	SSR4 [R/W] B, H, W 00001000	RDR4 /TDR4 [R/W] B, H, W 00000000	LIN-UART4
0000E4H	ESCR4 [R/W] B, H, W 00000X00	ECCR4 [R/W] B, H, W -0000-XX	BGR4 [R/W] B, H, W -0000000 00000000		
0000E8H	SCR5 [R/W] B, H, W 00000000	SMR5 [R/W] B, H, W 00000000	SSR5 [R/W] B, H, W 00001000	RDR5 /TDR5 [R/W] B, H, W 00000000	LIN-UART5
0000ECH	ESCR5 [R/W] B, H, W 00000X00	ECCR5 [R/W] B, H, W -0000-XX	BGR5 [R/W] B, H, W -0000000 00000000		
0000F0H	SCR6 [R/W] B, H, W 00000000	SMR6 [R/W] B, H, W 00000000	SSR6 [R/W] B, H, W 00001000	RDR6 /TDR6 [R/W] B, H, W 00000000	LIN-UART6
0000F4H	ESCR6 [R/W] B, H, W 00000X00	ECCR6 [R/W] B, H, W -0000-XX	BGR6 [R/W] B, H, W -0000000 00000000		
0000F8H	SCR7 [R/W] B, H, W 00000000	SMR7 [R/W] B, H, W 00000000	SSR7 [R/W] B, H, W 00001000	RDR7 /TDR7 [R/W] B, H, W 00000000	LIN-UART7
0000FCH	ESCR7 [R/W] B, H, W 00000X00	ECCR7 [R/W] B, H, W -0000-XX	BGR7 [R/W] B, H, W -0000000 00000000		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000100H	TMRLRA1 [R/W] H XXXXXXXX XXXXXXXX		TMR1 [R] H XXXXXXXX XXXXXXXX		リロードタイマ 1
000104H	TMRLRB1 [R/W] H XXXXXXXX XXXXXXXX		TMCSR1 [R/W] B, H,W 00000000 0-000000		
000108H	TMRLRA2 [R/W] H XXXXXXXX XXXXXXXX		TMR2 [R] H XXXXXXXX XXXXXXXX		リロードタイマ 2
00010CH	TMRLRB2 [R/W] H XXXXXXXX XXXXXXXX		TMCSR2 [R/W] B, H,W 00000000 0-000000		
000110H	TMRLRA3 [R/W] H XXXXXXXX XXXXXXXX		TMR3 [R] H XXXXXXXX XXXXXXXX		リロードタイマ 3
000114H	TMRLRB3 [R/W] H XXXXXXXX XXXXXXXX		TMCSR3 [R/W] B, H,W 00000000 0-000000		
000118H ～ 00011CH	—	—	—	—	予約
000120H	OCCP6 [R/W] W 00000000 00000000 00000000 00000000				アウトプットコンペア 6,7
000124H	OCCP7 [R/W] W 00000000 00000000 00000000 00000000				
000128H	OCFS67 [R/W] B, H, W ------11	—	OCSH67[R/W] B, H, W ---0--00	OCSL67[R/W] B, H, W 0000--00	
00012CH	OCCP8 [R/W] W 00000000 00000000 00000000 00000000				アウトプットコンペア 8,9
000130H	OCCP9 [R/W] W 00000000 00000000 00000000 00000000				
000134H	OCFS89 [R/W] B, H, W ------11	—	OCSH89[R/W] B, H, W ---0--00	OCSL89[R/W] B, H, W 0000--00	
000138H	OCCP10 [R/W] W 00000000 00000000 00000000 00000000				アウトプットコンペア 10,11
00013CH	OCCP11 [R/W] W 00000000 00000000 00000000 00000000				
000140H	OCFS1011 [R/W] B, H, W ------11	—	OCSH1011[R/W] B, H, W ---0--00	OCSL1011[R/W] B, H, W 0000--00	
000144H	GCN13 [R/W] H 00110010 00010000		—	GCN23 [R/W] B ----0000	PPG12,13,14,15 制御
000148H	GCN14 [R/W] H 00110010 00010000		—	GCN24 [R/W] B ----0000	PPG16,17,18,19 制御

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
00014C$_H$	GCN15 [R/W] H 00110010 00010000		—	GCN25 [R/W] B ----0000	PPG20, 21, 22, 23 制御
000150$_H$	PTMR11 [R] H,W 11111111 11111111		PCSR11 [W] H,W XXXXXXXX XXXXXXXX		PPG11
000154$_H$	PDUT11 [W] H,W XXXXXXXX XXXXXXXX		PCN11 [R/W] B,H,W 0000000- 000000-0		
000158$_H$	PTMR12 [R] H,W 11111111 11111111		PCSR12 [W] H,W XXXXXXXX XXXXXXXX		PPG12
00015C$_H$	PDUT12 [W] H,W XXXXXXXX XXXXXXXX		PCN12 [R/W] B,H,W 0000000- 000000-0		
000160$_H$	PTMR13 [R] H,W 11111111 11111111		PCSR13 [W] H,W XXXXXXXX XXXXXXXX		PPG13
000164$_H$	PDUT13 [W] H,W XXXXXXXX XXXXXXXX		PCN13 [R/W] B,H,W 0000000- 000000-0		
000168$_H$	PTMR14 [R] H,W 11111111 11111111		PCSR14 [W] H,W XXXXXXXX XXXXXXXX		PPG14
00016C$_H$	PDUT14 [W] H,W XXXXXXXX XXXXXXXX		PCN14 [R/W] B,H,W 0000000- 000000-0		
000170$_H$	PTMR15 [R] H,W 11111111 11111111		PCSR15 [W] H,W XXXXXXXX XXXXXXXX		PPG15
000174$_H$	PDUT15 [W] H,W XXXXXXXX XXXXXXXX		PCN15 [R/W] B,H,W 0000000- 000000-0		
000178$_H$	PTMR16 [R] H,W 11111111 11111111		PCSR16 [W] H,W XXXXXXXX XXXXXXXX		PPG16
00017C$_H$	PDUT16 [W] H,W XXXXXXXX XXXXXXXX		PCN16 [R/W] B,H,W 0000000- 000000-0		
000180$_H$	PTMR17 [R] H,W 11111111 11111111		PCSR17 [W] H,W XXXXXXXX XXXXXXXX		PPG17
000184$_H$	PDUT17 [W] H,W XXXXXXXX XXXXXXXX		PCN17 [R/W] B,H,W 0000000- 000000-0		
000188$_H$	PTMR18 [R] H,W 11111111 11111111		PCSR18 [W] H,W XXXXXXXX XXXXXXXX		PPG18
00018C$_H$	PDUT18 [W] H,W XXXXXXXX XXXXXXXX		PCN18 [R/W] B,H,W 0000000- 000000-0		
000190$_H$	PTMR19 [R] H,W 11111111 11111111		PCSR19 [W] H,W XXXXXXXX XXXXXXXX		PPG19
000194$_H$	PDUT19 [W] H,W XXXXXXXX XXXXXXXX		PCN19 [R/W] B,H,W 0000000- 000000-0		
000198$_H$	PTMR20 [R] H,W 11111111 11111111		PCSR20 [W] H,W XXXXXXXX XXXXXXXX		PPG20
00019C$_H$	PDUT20 [W] H,W XXXXXXXX XXXXXXXX		PCN20 [R/W] B,H,W 0000000- 000000-0		
0001A0$_H$	PTMR21 [R] H,W 11111111 11111111		PCSR21 [W] H,W XXXXXXXX XXXXXXXX		PPG21
0001A4$_H$	PDUT21 [W] H,W XXXXXXXX XXXXXXXX		PCN21 [R/W] B,H,W 0000000- 000000-0		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0001A8H	PTMR22 [R] H,W 11111111 11111111		PCSR22 [W] H,W XXXXXXXX XXXXXXXX		PPG22
0001ACH	PDUT22 [W] H,W XXXXXXXX XXXXXXXX		PCN22 [R/W] B,H,W 0000000- 000000-0		
0001B0H	PTMR23 [R] H,W 11111111 11111111		PCSR23 [W] H,W XXXXXXXX XXXXXXXX		PPG23
0001B4H	PDUT23 [W] H,W XXXXXXXX XXXXXXXX		PCN23 [R/W] B,H,W 0000000- 000000-0		
0001B8H ～ 0001FCH	—	—	—	—	予約
000200H	PWC20 [R/W] H,W ------XX XXXXXXXX		PWC10 [R/W] H,W ------XX XXXXXXXX		ステッピングモータ制御 0
000204H	—	PWC0 [R/W] B -00000--	PWS20 [R/W] B,H,W -0000000	PWS10 [R/W] B,H,W --000000	
000208H	PWC21 [R/W] H,W ------XX XXXXXXXX		PWC11 [R/W] H,W ------XX XXXXXXXX		ステッピングモータ制御 1
00020CH	—	PWC1 [R/W] B -00000--	PWS21 [R/W] B,H,W -0000000	PWS11 [R/W] B,H,W --000000	
000210H	PWC22 [R/W] H,W ------XX XXXXXXXX		PWC12 [R/W] H,W ------XX XXXXXXXX		ステッピングモータ制御 2
000214H	—	PWC2 [R/W] B -00000--	PWS22 [R/W] B,H,W -0000000	PWS12 [R/W] B,H,W --000000	
000218H	PWC23 [R/W] H,W ------XX XXXXXXXX		PWC13 [R/W] H,W ------XX XXXXXXXX		ステッピングモータ制御 3
00021CH	—	PWC3 [R/W] B -00000--	PWS23 [R/W] B,H,W -0000000	PWS13 [R/W] B,H,W --000000	
000220H	PWC24 [R/W] H,W ------XX XXXXXXXX		PWC14 [R/W] H,W ------XX XXXXXXXX		ステッピングモータ制御 4
000224H	—	PWC4 [R/W] B -00000--	PWS24 [R/W] B,H,W -0000000	PWS14 [R/W] B,H,W --000000	
000228H	PWC25 [R/W] H,W ------XX XXXXXXXX		PWC15 [R/W] H,W ------XX XXXXXXXX		ステッピングモータ制御 5
00022CH	—	PWC5 [R/W] B -00000--	PWS25 [R/W] B,H,W -0000000	PWS15 [R/W] B,H,W --000000	
000230H ～ 000238H	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
00023C$_H$	DACR0 [R/W] B,H,W -------0	DADR0 [R/W] B,H,W XXXXXXXX	DACR1 [R/W] B,H,W -------0	DADR1 [R/W] B,H,W XXXXXXXX	DA コンバータ
000240$_H$	CPCLR0 [R/W] W 11111111 11111111 11111111 11111111				フリーランタイマ 0
000244$_H$	TCDT0 [R/W] W 00000000 00000000 00000000 00000000				
000248$_H$	TCCSH0 [R/W]B,H,W 0-----00	TCCSL0 [R/W]B,H,W -1-00000	—		
00024C$_H$	CPCLR1 [R/W] W 11111111 11111111 11111111 11111111				フリーランタイマ 1
000250$_H$	TCDT1 [R/W] W 00000000 00000000 00000000 00000000				
000254$_H$	TCCSH1 [R/W]B,H,W 0-----00	TCCSL1 [R/W]B,H,W -1-00000	—		
000258$_H$	—	—	—	—	予約
00025C$_H$	GCN10 [R/W] H 00110010 00010000		—	GCN20 [R/W] B ----0000	PPG0, 1, 2, 3 制御
000260$_H$	GCN11 [R/W] H 00110010 00010000		—	GCN21 [R/W] B ----0000	PPG4, 5, 6, 7 制御
000264$_H$	GCN12 [R/W] H 00110010 00010000		—	GCN22 [R/W] B ----0000	PPG8, 9, 10, 11 制御
000268$_H$	—	—	—	PPGDIV [R/W] B ------00	PPG0
00026C$_H$	PTMR0 [R] H,W 11111111 11111111		PCSR0 [W] H,W XXXXXXXX XXXXXXXX		
000270$_H$	PDUT0 [W] H,W XXXXXXXX XXXXXXXX		PCN0 [R/W] B, H,W 0000000- 000000-0		
000274$_H$	PTMR1 [R] H,W 11111111 11111111		PCSR1 [W] H,W XXXXXXXX XXXXXXXX		PPG1
000278$_H$	PDUT1 [W] H,W XXXXXXXX XXXXXXXX		PCN1 [R/W] B,H,W 0000000- 000000-0		
00027C$_H$	PTMR2 [R] H,W 11111111 11111111		PCSR2 [W] H,W XXXXXXXX XXXXXXXX		PPG2
000280$_H$	PDUT2 [W] H,W XXXXXXXX XXXXXXXX		PCN2 [R/W] B,H,W 0000000- 000000-0		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000284H	PTMR3 [R] H,W 11111111 11111111		PCSR3 [W] H,W XXXXXXXX XXXXXXXX		PPG3
000288H	PDUT3 [W] H,W XXXXXXXX XXXXXXXX		PCN3 [R/W] B,H,W 0000000- 000000-0		
00028CH	PTMR4 [R] H,W 11111111 11111111		PCSR4 [W] H,W XXXXXXXX XXXXXXXX		PPG4
000290H	PDUT4 [W] H,W XXXXXXXX XXXXXXXX		PCN4 [R/W] B,H,W 0000000- 000000-0		
000294H	PTMR5 [R] H,W 11111111 11111111		PCSR5 [W] H,W XXXXXXXX XXXXXXXX		PPG5
000298H	PDUT5 [W] H,W XXXXXXXX XXXXXXXX		PCN5 [R/W] B,H,W 0000000- 000000-0		
00029CH	PTMR6 [R] H,W 11111111 11111111		PCSR6 [W] H,W XXXXXXXX XXXXXXXX		PPG6
0002A0H	PDUT6 [W] H,W XXXXXXXX XXXXXXXX		PCN6 [R/W] B,H,W 0000000- 000000-0		
0002A4H	PTMR7 [R] H,W 11111111 11111111		PCSR7 [W] H,W XXXXXXXX XXXXXXXX		PPG7
0002A8H	PDUT7 [W] H,W XXXXXXXX XXXXXXXX		PCN7 [R/W] B,H,W 0000000- 000000-0		
0002ACH	PTMR8 [R] H,W 11111111 11111111		PCSR8 [W] H,W XXXXXXXX XXXXXXXX		PPG8
0002B0H	PDUT8 [W] H,W XXXXXXXX XXXXXXXX		PCN8 [R/W] B,H,W 0000000- 000000-0		
0002B4H	PTMR9 [R] H,W 11111111 11111111		PCSR9 [W] H,W XXXXXXXX XXXXXXXX		PPG9
0002B8H	PDUT9 [W] H,W XXXXXXXX XXXXXXXX		PCN9 [R/W] B,H,W 0000000- 000000-0		
0002BCH	PTMR10 [R] H,W 11111111 11111111		PCSR10 [W] H,W XXXXXXXX XXXXXXXX		PPG10
0002C0H	PDUT10 [W] H,W XXXXXXXX XXXXXXXX		PCN10 [R/W] B,H,W 0000000- 000000-0		
0002C4H	IPCP0 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				インプットキャプチャ 0, 1
0002C8H	IPCP1 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0002CCH	ICFS01 [R/W] B, H, W ------00	—	LSYNS0 [R/W] B,H,W --000000	ICS01 [R/W] B, H, W 00000000	

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0002D0H	IPCP2 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				インプットキャプチャ 2, 3
0002D4H	IPCP3 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0002D8H	ICFS23 [R/W] B, H, W ------00	—	—	ICS23 [R/W] B, H, W 00000000	
0002DCH	IPCP4 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				インプットキャプチャ 4, 5
0002E0H	IPCP5 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0002E4H	ICFS45 [R/W] B, H, W ------00	—	—	ICS45 [R/W] B, H, W 00000000	
0002E8H	OCCP0 [R/W] W 00000000 00000000 00000000 00000000				アウトプットコンペア 0, 1
0002ECH	OCCP1 [R/W] W 00000000 00000000 00000000 00000000				
0002F0H	OCFS01 [R/W] B, H, W ------11	—	OCSH01[R/W] B, H, W ---0--00	OCSL01[R/W] B, H, W 0000--00	
0002F4H	OCCP2 [R/W] W 00000000 00000000 00000000 00000000				アウトプットコンペア 2, 3
0002F8H	OCCP3 [R/W] W 00000000 00000000 00000000 00000000				
0002FCH	OCFS23 [R/W] B, H, W ------11	—	OCSH23[R/W] B, H, W ---0--00	OCSL23[R/W] B, H, W 0000--00	
000300H ～ 00030CH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000310H	—	—	MPUCR [R/W] H 000000-0 ----0100		MPU [S] (この領域へは CPU のみアクセス可能)
000314H	—	—	—	—	
000318H	—				
00031CH	—	—	—		
000320H	DPVAR [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000324H	—	—	DPVSR [R/W] H -------- 00000--0		
000328H	DEAR [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
00032CH	—	—	DESR [R/W] H -------- 00000--0		
000330H	PABR0 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
000334H	—	—	PACR0 [R/W] H 000000-0 00000--0		
000338H	PABR1 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
00033CH	—	—	PACR1 [R/W] H 000000-0 00000--0		
000340H	PABR2 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
000344H	—	—	PACR2 [R/W] H 000000-0 00000--0		
000348H	PABR3 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
00034CH	—	—	PACR3 [R/W] H 000000-0 00000--0		
000350H	PABR4 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
000354H	—	—	PACR4 [R/W] H 000000-0 00000--0		
000358H	PABR5 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
00035CH	—	—	PACR5 [R/W] H 000000-0 00000--0		
000360H	PABR6 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000364H	—	—	PACR6 [R/W] H 000000-0 00000--0		MPU [S] (この領域へは CPU のみアクセス可能)
000368H	PABR7 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
00036CH	—	—	PACR7 [R/W] H 000000-0 00000--0		
000370H	PABR8 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				MPU [S] (MPU 12ch または 16ch 搭載品種のみ) (この領域へは CPU のみアクセス可能)
000374H	—	—	PACR8 [R/W] H 000000-0 00000--0		
000378H	PABR9[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
00037CH	—	—	PACR9 [R/W] H 000000-0 00000--0		
000380H	PABR10 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
000384H	—	—	PACR10 [R/W] H 000000-0 00000--0		
000388H	PABR11 [R/W] ,W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
00038CH	—	—	PACR11 [R/W] H 000000-0 00000--0		
000390H	PABR12 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				MPU [S] (MPU 16ch 搭載品種のみ) (この領域へは CPU のみアクセス可能)
000394H	—	—	PACR12 [R/W] H 000000-0 00000--0		
000398H	PABR13 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
00039CH	—	—	PACR13 [R/W] H 000000-0 00000--0		
0003A0H	PABR14 [R/W]W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
0003A4H	—	—	PACR14 [R/W] H 000000-0 00000--0		
0003A8H	PABR15 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXX0000				
0003ACH	—	—	PACR15 [R/W] H 000000-0 00000--0		
0003B0H ～ 0003FCH	—	—	—	—	予約[S]

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000400H	ICSEL0[R/W] B, H, W -----000	ICSEL1[R/W] B, H, W -----000	ICSEL2[R/W] B, H, W -------0	ICSEL3[R/W] B, H, W -------0	DMA 転送要求の発生・クリア
000404H	ICSEL4[R/W] B, H, W -------0	ICSEL5[R/W] B, H, W -------0	ICSEL6[R/W] B, H, W -----000	ICSEL7[R/W] B, H, W -----000	
000408H	ICSEL8[R/W] B, H, W ------00	ICSEL9[R/W] B, H, W ------00	ICSEL10 [R/W]B, H, W ------00	ICSEL11[R/W] B, H, W ------00	DMA 転送要求の発生・クリア
00040CH	ICSEL12[R/W] B, H, W ------00	ICSEL13[R/W] B, H, W -------0	ICSEL14 [R/W]B, H, W -------0	ICSEL15[R/W] B, H, W -------0	
000410H	ICSEL16[R/W] B, H, W -------0	ICSEL17[R/W] B, H, W -------0	ICSEL18 [R/W]B, H, W -------0	ICSEL19[R/W] B, H, W -----000	
000414H	ICSEL20[R/W] B, H, W -----000	ICSEL21[R/W] B, H, W ------00	ICSEL22 [R/W]B, H, W ------00	—	
000418H	IRPR0H[R] B, H, W 00------	IRPR0L[R] B, H, W 00------	IRPR1H[R] B, H, W 00------	IRPR1L[R] B, H, W 00------	割込み要求一括読出しレジスタ
00041CH	IRPR2H[R] B, H, W 00------	IRPR2L[R] B, H, W 00------	IRPR3H[R] B, H, W 000000--	IRPR3L[R] B, H, W 000000--	
000420H	IRPR4H[R] B, H, W 0000----	IRPR4L[R] B, H, W 0000----	IRPR5H[R] B, H, W 0000----	IRPR5L[R] B, H, W 000-----	
000424H	IRPR6H[R] B, H, W --000---	IRPR6L[R] B, H, W 00000---	IRPR7H[R] B, H, W -0000---	IRPR7L[R] B, H, W ------00	
000428H	IRPR8H[R] B, H, W 000-----	IRPR8L[R] B, H, W 000-----	IRPR9H[R] B, H, W 00------	IRPR9L[R] B, H, W 00------	
00042CH	IRPR10H[R] B, H, W 00------	IRPR10L[R] B, H, W 00------	IRPR11H[R] B, H, W 00------	IRPR11L[R] B, H, W 00------	
000430H	IRPR12H[R] B, H, W 000000--	IRPR12L[R] B, H, W 000000--	IRPR13H[R] B, H, W 000-----	IRPR13L[R] B, H, W 00000---	
000434H	IRPR14H[R] B, H, W 00000000	IRPR14L[R] B, H, W 00000000	IRPR15H[R] B, H, W 000-----	—	
000438H ～ 00043CH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000440_H	ICR00 [R/W] B, H, W ---11111	ICR01 [R/W] B, H, W ---11111	ICR02 [R/W] B, H, W ---11111	ICR03 [R/W] B, H, W ---11111	割込み制御[S]
000444_H	ICR04 [R/W] B, H, W ---11111	ICR05 [R/W] B, H, W ---11111	ICR06 [R/W] B, H, W ---11111	ICR07 [R/W] B, H, W ---11111	
000448_H	ICR08 [R/W] B, H, W ---11111	ICR09 [R/W] B, H, W ---11111	ICR10 [R/W] B, H, W ---11111	ICR11 [R/W] B, H, W ---11111	
$00044C_H$	ICR12 [R/W] B, H, W ---11111	ICR13 [R/W] B, H, W ---11111	ICR14 [R/W] B, H, W ---11111	ICR15 [R/W] B, H, W ---11111	
000450_H	ICR16 [R/W] B, H, W ---11111	ICR17 [R/W] B, H, W ---11111	ICR18 [R/W] B, H, W ---11111	ICR19 [R/W] B, H, W ---11111	
000454_H	ICR20 [R/W] B, H, W ---11111	ICR21 [R/W] B, H, W ---11111	ICR22 [R/W] B, H, W ---11111	ICR23 [R/W] B, H, W ---11111	
000458_H	ICR24 [R/W] B, H, W ---11111	ICR25 [R/W] B, H, W ---11111	ICR26 [R/W] B, H, W ---11111	ICR27 [R/W] B, H, W ---11111	
$00045C_H$	ICR28 [R/W] B, H, W ---11111	ICR29 [R/W] B, H, W ---11111	ICR30 [R/W] B, H, W ---11111	ICR31 [R/W] B, H, W ---11111	
000460_H	ICR32 [R/W] B, H, W ---11111	ICR33 [R/W] B, H, W ---11111	ICR34 [R/W] B, H, W ---11111	ICR35 [R/W] B, H, W ---11111	
000464_H	ICR36 [R/W] B, H, W ---11111	ICR37 [R/W] B, H, W ---11111	ICR38 [R/W] B, H, W ---11111	ICR39 [R/W] B, H, W ---11111	
000468_H	ICR40 [R/W] B, H, W ---11111	ICR41 [R/W] B, H, W ---11111	ICR42 [R/W] B, H, W ---11111	ICR43 [R/W] B, H, W ---11111	
$00046C_H$	ICR44 [R/W] B, H, W ---11111	ICR45 [R/W] B, H, W ---11111	ICR46 [R/W] B, H, W ---11111	ICR47 [R/W] B, H, W ---11111	
000470_H ～ $00047C_H$	—	—	—	—	予約[S]
000480_H	RSTRR [R] B,H,W XXXX--XX	RSTCR [R/W] B,H,W 111----0	STBCR [R/W] B,H,W * 000---11	—	リセット制御[S] 消費電力制御[S] * : STBCR への DMA による書込みは禁止です。
000484_H	—	—	—	—	予約 [S]

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000488H	DIVR0 [R/W] B,H,W 000-----	DIVR1 [R/W] B,H,W 0001----	DIVR2 [R/W] B,H,W 0011----	—	クロック制御[S]
00048CH	—	—	—	—	予約 [S]
000490H	IORR0[R/W] B, H, W -0000000	IORR1[R/W] B, H, W -0000000	IORR2[R/W] B, H, W -0000000	IORR3[R/W] B, H, W -0000000	周辺からの DMA 転送要求[S]
000494H	IORR4[R/W] B, H, W -0000000	IORR5[R/W] B, H, W -0000000	IORR6[R/W] B, H, W -0000000	IORR7[R/W] B, H, W -0000000	
000498H	IORR8[R/W] B, H, W -0000000	IORR9[R/W] B, H, W -0000000	IORR10[R/W] B, H, W -0000000	IORR11[R/W] B, H, W -0000000	
00049CH	IORR12[R/W] B, H, W -0000000	IORR13[R/W] B, H, W -0000000	IORR14[R/W] B, H, W -0000000	IORR15[R/W] B, H, W -0000000	
0004A0H	—	—	—	—	予約
0004A4H	CANPRE [R/W] B,H,W ----0000	—	—	—	CAN プリスケーラ
0004A8H ～ 0004B4H	—	—	—	—	予約
0004B8H	CUCR0 [R/W] B,H,W -------- ---0--00		CUTD0 [R/W] B,H,W 10000000 00000000		RTC/WDT1 補正 (キャリブレーション)
0004BCH	CUTR0 [R] B,H,W -------- 00000000 00000000 00000000				
0004C0H	—	—	—	—	
0004C4H	CUCR1 [R/W] B,H,W -------- ---0--00		CUTD1[R/W] B,H,W 11000011 01010000		
0004C8H	CUTR1 [R] B,H,W -------- 00000000 00000000 00000000				
0004CCH	CRTR [R/W] B,H,W 01111111	—	—	—	
0004D0H ～ 0004DCH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0004E0H	SCR8/(IBCR8) [R/W] B,H,W 0--00000	SMR8 [R/W] B,H,W 000-0000	SSR8 [R/W] B,H,W 0-000011	ESCR8/(IBSR8) [R/W] B,H,W -0000000	Multi-UART8 *1 下位8ビットにアクセスする場合のみ、バイトアクセス可能
0004E4H	RDR8/(TDR8)[R/W] B,H,W *1 -------0 00000000		BGR8 [R/W] H,W 00000000 00000000		
0004E8H	—	—	—	—	
0004ECH	FCR18 [R/W] B,H,W ---00100	FCR08 [R/W] B,H,W -0000000	FBYTE28 [R/W] B,H,W 00000000	FBYTE18 [R/W] B,H,W 00000000	
0004F0H	SCR9/(IBCR9) [R/W] B,H,W 0--00000	SMR9 [R/W] B,H,W 000-0000	SSR9 [R/W] B,H,W 0-000011	ESCR9/(IBSR9) [R/W] B,H,W -0000000	Multi-UART9 *1 下位8ビットにアクセスする場合のみ、バイトアクセス可能
0004F4H	RDR9/(TDR9)[R/W] B,H,W *1 -------0 00000000		BGR9 [R/W] H,W 00000000 00000000		
0004F8H	—	—	—	—	
0004FCH	FCR19 [R/W] B,H,W ---00100	FCR09 [R/W] B,H,W -0000000	FBYTE29 [R/W] B,H,W 00000000	FBYTE19 [R/W] B,H,W 00000000	
000500H ～ 00050CH	—	—	—	—	予約
000510H	CSELR [R/W] B,H,W 001---00	CMONR [R] B,H,W 001---00	MTMCR [R/W] B,H,W 00001111	STMCR [R/W] B,H,W 0000-111	クロック制御[S]
000514H	PLLCR [R/W] B,H,W -------- 11110000		CSTBR [R/W] B,H,W -0000000	PTMCR [R/W] B,H,W 00------	
000518H	—	—	CPUAR [R/W] B,H,W 0----XXX	—	リセット[S]
00051CH	—	—	—	—	予約 [S]

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000520H	CCPSSELR [R/W] B,H,W -------0	—	—	CCPSDIVR [R/W] B,H,W -000-000	クロック制御 2
000524H	—	CCPLLFBR [R/W] B,H,W -0000000	CCSSFBR0 [R/W] B,H,W --000000	CCSSFBR1 [R/W] B,H,W ---00000	
000528H	—	CCSSCCR0 [R/W] B,H,W ----0000	CCSSCCR1[R/W] H,W 000----- --------		
00052CH	—	CCCGRCR0 [R/W] B,H,W 00----00	CCCGRCR1 [R/W] B,H,W 00000000	CCCGRCR2 [R/W] B,H,W 00000000	
000530H	CCRTSELR [R/W] B,H,W 0------0	—	CCPMUCR0 [R/W] B,H,W 0-----00	CCPMUCR1 [R/W] B,H,W 0--00000	
000534H	—	—	—	—	
000538H	—	—	—	—	
00053CH	—	—	—	—	
000540H ～ 00054CH	—	—	—	—	予約
000550H	EIRR0[R/W] B,H,W XXXXXXXX	ENIR0[R/W] B,H,W 00000000	ELVR0[R/W] B,H,W 00000000 00000000		外部割込み (INT0～INT7)
000554H	EIRR1[R/W] B,H,W XXXXXXXX	ENIR1[R/W] B,H,W 00000000	ELVR1[R/W] B,H,W 00000000 00000000		外部割込み (INT8～INT15)
000558H	—	—	—	—	予約
00055CH	—	—	WTDR[R/W] H 00000000 00000000		リアルタイムクロック
000560H	—	WTCRH [R/W] B ------00	WTCRM [R/W] B,H 00000000	WTCRL [R/W] B,H ----00-0	
000564H	—	WTBRH [R/W] B --XXXXXX	WTBRM [R/W] B XXXXXXXX	WTBRL [R/W] B XXXXXXXX	
000568H	WTHR [R/W] B,H ---00000	WTMR [R/W] B,H --000000	WTSR [R/W] B --000000	—	
00056CH	—	CSVCR[R/W] B -001110-, -001010- *3	—	—	クロックスーパバイザ
000570H ～ 00057CH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000580H	REGSEL [R/W] B,H,W 0110011-	—	—	—	レギュレータ制御
000584H	LVD5R [R/W] B,H,W -------1	LVD5F [R/W] B,H,W 0-100--1	LVD [R/W] B,H,W 01000--0	—	低電圧検出
000588H ～ 00058CH	—	—	—	—	予約
000590H	PMUSTR [R/W] B,H,W 0-----1X	PMUCTLR [R/W] B,H,W 0-00----	PWRTMCTL [R/W] B,H,W -----011	—	PMU
000594H	PMUINTF0 [R/W] B,H,W 00000000	PMUINTF1 [R/W] B,H,W 00000000	PMUINTF2 [R/W] B,H,W 0000----	—	
000598H	—	—	—	—	
00059CH ～ 0005A4H	—	—	—	—	予約
0005A8H	LCDCMR [R/W] B,H,W 0-------	LCRS [R/W] B,H,W 00000000	LCR0 [R/W] B,H,W 00010000	LCR1 [R/W] B,H,W --------	LCD コントローラ
0005ACH	VRAM0[R/W] B,H,W 00000000	VRAM1[R/W] B,H,W 00000000	VRAM2[R/W] B,H,W 00000000	VRAM3[R/W] B,H,W 00000000	
0005B0H	VRAM4[R/W] B,H,W 00000000	VRAM5[R/W] B,H,W 00000000	VRAM6[R/W] B,H,W 00000000	VRAM7[R/W] B,H,W 00000000	
0005B4H	VRAM8[R/W] B,H,W 00000000	VRAM9[R/W] B,H,W 00000000	VRAM10[R/W] B,H,W 00000000	VRAM11[R/W] B,H,W 00000000	
0005B8H	VRAM12[R/W] B,H,W 00000000	VRAM13[R/W] B,H,W 00000000	VRAM14[R/W] B,H,W 00000000	VRAM15[R/W] B,H,W 00000000	
0005BCH	LDR0[R/W] B,H,W -------0	LDR1[R/W] B,H,W 00000000	—	—	
0005C0H ～ 0005FCH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000600H	ASR0 [R/W] W 00000000 00000000 -------- 1111-001				外部バスインタフェース[S]
000604H	ASR1 [R/W] W XXXXXXXX XXXXXXXX -------- XXXX-XX0				
000608H	ASR2 [R/W] W XXXXXXXX XXXXXXXX -------- XXXX-XX0				
00060CH	ASR3 [R/W] W XXXXXXXX XXXXXXXX -------- XXXX-XX0				
000610H ～ 00063CH	—	—	—	—	予約[S]
000640H	ACR0 [R/W] W -------- -------- -------- 01--00--				外部バスインタフェース[S]
000644H	ACR1 [R/W] W -------- -------- -------- XX--XX--				
000648H	ACR2 [R/W] W -------- -------- -------- XX--XX--				
00064CH	ACR3 [R/W] W -------- -------- -------- XX--XX--				
000650H ～ 00067CH	—	—	—	—	予約[S]
000680H	AWR0 [R/W] W ----1111 00000000 11110000 00000-0-				外部バスインタフェース[S]
000684H	AWR1 [R/W] W ----XXXX XXXXXXXX XXXXXXXX XXXXX-X-				
000688H	AWR2 [R/W] W ----XXXX XXXXXXXX XXXXXXXX XXXXX-X-				
00068CH	AWR3 [R/W] W ----XXXX XXXXXXXX XXXXXXXX XXXXX-X-				
000690H ～ 00070CH	—	—	—	—	予約 (～0006FFH[S])
000710H	BPCCRA[R/W] B 00000000	BPCCRB[R/W] B 00000000	BPCCRC[R/W] B 00000000	—	バスパフォーマンスカウンタ
000714H	BPCTRA [R/W] W 00000000 00000000 00000000 00000000				
000718H	BPCTRB [R/W] W 00000000 00000000 00000000 00000000				
00071CH	BPCTRC [R/W] W 00000000 00000000 00000000 00000000				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000720$_H$ ～ 0007F8$_H$	—	—	—	—	予約
0007FC$_H$	BMODR[R] B, H, W XXXXXXXX	—	—	—	動作モード
000800$_H$ ～ 00083C$_H$	—	—	—	—	予約[S]
000840$_H$	FCTLR[R/W] H -0--1000 0--0----		—	FSTR[R/W] B -----001	フラッシュメモリレジスタ [S]
000844$_H$	—	—	—	—	予約 [S]
000848$_H$	—	—	—	—	予約 [S]
00084C$_H$	—	—	—	—	
000850$_H$	—	—	—	—	
000854$_H$	—	—	—	—	
000858$_H$	—	—	WREN[R/W] H 00000000 00000000		ワイルドレジスタ [S]
00085C$_H$	—	—	—	—	予約 [S]
000860$_H$	—	—	—	—	
000864$_H$	—	—	—	—	
000868$_H$	—	—	—	—	
00086C$_H$	—	—	—	—	
000870$_H$	—	—	—	—	
000874$_H$	—	—	—	—	
000878$_H$	—	—	—	—	
00087C$_H$	—	—	—	—	
000880$_H$	WRAR00 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				ワイルドレジスタ [S]
000884$_H$	WRDR00 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000888$_H$	WRAR01 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
00088C$_H$	WRDR01 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000890$_H$	WRAR02 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
000894$_H$	WRDR02 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000898$_H$	WRAR03 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
00089C$_H$	WRDR03 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0008A0H	WRAR04 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				ワイルドレジスタ [S]
0008A4H	WRDR04 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008A8H	WRAR05 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008ACH	WRDR05 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008B0H	WRAR06 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008B4H	WRDR06 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008B8H	WRAR07 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008BCH	WRDR07 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008C0H	WRAR08 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008C4H	WRDR08 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008C8H	WRAR09 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008CCH	WRDR09 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008D0H	WRAR10 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008D4H	WRDR10 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008D8H	WRAR11 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008DCH	WRDR11 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008E0H	WRAR12 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008E4H	WRDR12 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008E8H	WRAR13 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008ECH	WRDR13 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008F0H	WRAR14 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				
0008F4H	WRDR14 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
0008F8H	WRAR15 [R/W] W -------- --XXXXXX XXXXXXXX XXXXXX--				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0008FC$_H$	WRDR15 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				ワイルドレジスタ [S]
000900$_H$ ～ 000BF8$_H$	—	—	—	—	予約
000BFC$_H$	—	—	UER [W] B,H,W -------- -------X		OCDU
000C00$_H$	DCCR0[R/W] W 0----000 --00--00 00000000 0-000000				DMA コントローラ [S]
000C04$_H$	DCSR0[R/W] H 0------- -----000		DTCR0[R/W] H 00000000 00000000		
000C08$_H$	DSAR0[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C0C$_H$	DDAR0 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C10$_H$	DCCR1 [R/W] W 0----000 --00--00 00000000 0-000000				
000C14$_H$	DCSR1 [R/W] H 0------- -----000		DTCR1 [R/W] H 00000000 00000000		
000C18$_H$	DSAR1 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C1C$_H$	DDAR1 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C20$_H$	DCCR2 [R/W] W 0----000 --00--00 00000000 0-000000				
000C24$_H$	DCSR2 [R/W] H 0------- -----000		DTCR2 [R/W] H 00000000 00000000		
000C28$_H$	DSAR2 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C2C$_H$	DDAR2 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C30$_H$	DCCR3[R/W] W 0----000 --00--00 00000000 0-000000				
000C34$_H$	DCSR3 [R/W] H 0------- -----000		DTCR3 [R/W] H 00000000 00000000		
000C38$_H$	DSAR3 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C3C$_H$	DDAR3 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
$000C40_H$	DCCR4 [R/W] W 0----000 --00--00 00000000 0-000000				DMA コントローラ [S]
$000C44_H$	DCSR4 [R/W] H 0------- -----000		DTCR4 [R/W] H 00000000 00000000		
$000C48_H$	DSAR4[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C4C_H$	DDAR4[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C50_H$	DCCR5 [R/W] W 0----000 --00--00 00000000 0-000000				
$000C54_H$	DCSR5 [R/W] H 0------- -----000		DTCR5 [R/W] H 00000000 00000000		
$000C58_H$	DSAR5 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C5C_H$	DDAR5 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C60_H$	DCCR6 [R/W] W 0----000 --00--00 00000000 0-000000				
$000C64_H$	DCSR6 [R/W] H 0------- -----000		DTCR6 [R/W] H 00000000 00000000		
$000C68_H$	DSAR6 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C6C_H$	DDAR6 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C70_H$	DCCR7 [R/W] W 0----000 --00--00 00000000 0-000000				
$000C74_H$	DCSR7 [R/W] H 0------- -----000		DTCR7 [R/W] H 00000000 00000000		
$000C78_H$	DSAR7 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C7C_H$	DDAR7 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$000C80_H$	DCCR8 [R/W] W 0----000 --00--00 00000000 0-000000				
$000C84_H$	DCSR8 [R/W] H 0------- -----000		DTCR8 [R/W] H 00000000 00000000		
$000C88_H$	DSAR8 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000C8C$_H$	DDAR8 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				DMA コントローラ [S]
000C90$_H$	DCCR9 [R/W] W 0----000 --00--00 00000000 0-000000				
000C94$_H$	DCSR9 [R/W] H 0------- -----000		DTCR9 [R/W] H 00000000 00000000		
000C98$_H$	DSAR9 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000C9C$_H$	DDAR9 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CA0$_H$	DCCR10 [R/W] W 0----000 --00--00 00000000 0-000000				
000CA4$_H$	DCSR10[R/W] H 0------- -----000		DTCR10[R/W] H 00000000 00000000		
000CA8$_H$	DSAR10 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CAC$_H$	DDAR10 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CB0$_H$	DCCR11[R/W] W 0----000 --00--00 00000000 0-000000				
000CB4$_H$	DCSR11 [R/W] H 0------- -----000		DTCR11 [R/W] H 00000000 00000000		
000CB8$_H$	DSAR11 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CBC$_H$	DDAR11 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CC0$_H$	DCCR12 [R/W] W 0----000 --00--00 00000000 0-000000				
000CC4$_H$	DCSR12 [R/W] H 0------- -----000		DTCR12 [R/W] H 00000000 00000000		
000CC8$_H$	DSAR12 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CCC$_H$	DDAR12 [R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CD0$_H$	DCCR13 [R/W] W 0----000 --00--00 00000000 0-000000				
000CD4$_H$	DCSR13[R/W] H 0------- -----000		DTCR13[R/W] H 00000000 00000000		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000CD8H	DSAR13[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				DMA コントローラ [S]
000CDCH	DDAR13[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CE0H	DCCR14[R/W] W 0----000 --00--00 00000000 0-000000				
000CE4H	DCSR14[R/W] H 0------- -----000		DTCR14[R/W] H 00000000 00000000		
000CE8H	DSAR14[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CECH	DDAR14[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CF0H	DCCR15[R/W] W 0----000 --00--00 00000000 0-000000				
000CF4H	DCSR15[R/W] H 0------- -----000		DTCR15[R/W] H 00000000 00000000		
000CF8H	DSAR15[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000CFCH	DDAR15[R/W] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000D00H ～ 000DF0H	—	—	—	—	予約[S]
000DF4H	—	—	DNMIR[R/W] B 0------0	DILVR[R/W] B ---11111	DMA コントローラ [S]
000DF8H	DMACR[R/W] W 0------- -------- 0------- --------				
000DFCH	—	—	—	—	予約[S]

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000E00H	DDR00[R/W] B,H,W 00000000	DDR01[R/W] B,H,W 00000000	DDR02[R/W] B,H,W 00000000	DDR03[R/W] B,H,W 00000000	データ方向レジスタ
000E04H	DDR04[R/W] B,H,W 00000000	DDR05[R/W] B,H,W -0000000	DDR06[R/W] B,H,W 00000000	DDR07[R/W] B,H,W 00000000	
000E08H	DDR08[R/W] B,H,W 00000000	DDR09[R/W] B,H,W 00000000	DDR10[R/W] B,H,W 00000000	DDR11[R/W] B,H,W 00000000	
000E0CH	DDR12[R/W] B,H,W 00000000	DDR13[R/W] B,H,W 00-00000	—	—	
000E10H ～ 000E1CH	—	—	—	—	予約
000E20H	PFR00[R/W] B,H,W 00000000	PFR01[R/W] B,H,W 00000000	PFR02[R/W] B,H,W 00000000	PFR03[R/W] B,H,W 10000000	ポート機能レジスタ
000E24H	PFR04[R/W] B,H,W 11111111	PFR05[R/W] B,H,W 11111111	PFR06[R/W] B,H,W 00000000	PFR07[R/W] B,H,W 00000000	
000E28H	PFR08[R/W] B,H,W 00000000	PFR09[R/W] B,H,W 0-000000	PFR10[R/W] B,H,W 00000000	PFR11[R/W] B,H,W 00000000	
000E2CH	PFR12[R/W] B,H,W 00000000	PFR13[R/W] B,H,W 00-00000	—	—	
000E30H ～ 000E3CH	—	—	—	—	予約
000E40H	PDDR00[R] B,H,W XXXXXXXX	PDDR01[R] B,H,W XXXXXXXX	PDDR02[R] B,H,W XXXXXXXX	PDDR03[R] B,H,W XXXXXXXX	入力データダイレクトリードレジスタ
000E44H	PDDR04[R] B,H,W XXXXXXXX	PDDR05[R] B,H,W XXXXXXXX	PDDR06[R] B,H,W XXXXXXXX	PDDR07[R] B,H,W XXXXXXXX	
000E48H	PDDR08[R] B,H,W XXXXXXXX	PDDR09[R] B,H,W XXXXXXXX	PDDR10[R] B,H,W XXXXXXXX	PDDR11[R] B,H,W XXXXXXXX	
000E4CH	PDDR12[R] B,H,W XXXXXXXX	PDDR13[R] B,H,W XX-XXXXX	—	—	
000E50H ～ 000E5CH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
$000E60_H$	EPFR00[R/W] B,H,W 00000000	EPFR01[R/W] B,H,W 00000000	EPFR02[R/W] B,H,W ---00000	EPFR03[R/W] B,H,W ---00000	拡張ポート機能レジスタ
$000E64_H$	EPFR04[R/W] B,H,W ---00000	EPFR05[R/W] B,H,W ---00000	EPFR06[R/W] B,H,W ---00000	EPFR07[R/W] B,H,W ---00000	
$000E68_H$	EPFR08[R/W] B,H,W ---00000	EPFR09[R/W] B,H,W ---00000	EPFR10[R/W] B,H,W -0000000	EPFR11[R/W] B,H,W --000000	
$000E6C_H$	EPFR12[R/W] B,H,W --000000	EPFR13[R/W] B,H,W --000000	EPFR14[R/W] B,H,W --000000	EPFR15[R/W] B,H,W -0000000	拡張ポート機能レジスタ
$000E70_H$	EPFR16[R/W] B,H,W 00000000	EPFR17[R/W] B,H,W 00000000	EPFR18[R/W] B,H,W 10000000	EPFR19[R/W] B,H,W 11111111	
$000E74_H$	EPFR20[R/W] B,H,W 11111111	EPFR21[R/W] B,H,W 00000000	EPFR22[R/W] B,H,W 00000000	EPFR23[R/W] B,H,W 00000000	
$000E78_H$	EPFR24[R/W] B,H,W -----000	EPFR25[R/W] B,H,W -----000	EPFR26[R/W] B,H,W ----0000	EPFR27[R/W] B,H,W ---00000	
$000E7C_H$	EPFR28[R/W] B,H,W ----0000	EPFR29[R/W] B,H,W 00000000	EPFR30[R/W] B,H,W 00000000	EPFR31[R/W] B,H,W 00000000	
$000E80_H$	EPFR32[R/W] B,H,W 00000000	EPFR33[R/W] B,H,W ---00000	EPFR34[R/W] B,H,W ---00000	EPFR35[R/W] B,H,W ---00000	
$000E84_H$	EPFR36[R/W] B,H,W ---00000	EPFR37[R/W] B,H,W 00000000	EPFR38[R/W] B,H,W ---00000	EPFR39[R/W] B,H,W 00000000	
$000E88_H$	EPFR40[R/W] B,H,W --000000	EPFR41[R/W] B,H,W -----000	EPFR42[R/W] B,H,W ------00	EPFR43[R/W] B,H,W 00000000	
$000E8C_H$	EPFR44[R/W] B,H,W 00000000	EPFR45[R/W] B,H,W 00000000	EPFR46[R/W] B,H,W --000000	EPFR47[R/W] B,H,W -------0	
$000E90_H$	—	—	—	—	
$000E94_H$	EPFR52[R/W] B,H,W -------0	EPFR53[R/W] B,H,W ---00000	EPFR54[R/W] B,H,W ----0000	—	
$000E98_H$ ～ $000E9C_H$	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000EA0H	PPCR00[R/W] B,H,W 11111111	PPCR01[R/W] B,H,W 11111111	PPCR02[R/W] B,H,W 11111111	PPCR03[R/W] B,H,W 11111111	ポートプルアップ/プルダウン制御レジスタ
000EA4H	PPCR04[R/W] B,H,W 11111111	PPCR05[R/W] B,H,W 11111111	PPCR06[R/W] B,H,W 11111111	PPCR07[R/W] B,H,W 11111111	
000EA8H	PPCR08[R/W] B,H,W 11111111	PPCR09[R/W] B,H,W 11111111	PPCR10[R/W] B,H,W 11111111	PPCR11[R/W] B,H,W 11111111	
000EACH	PPCR12[R/W] B,H,W 11111111	PPCR13[R/W] B,H,W 11-11111	—	—	
000EB0H ～ 000EBCH	—	—	—	—	予約
000EC0H	PPER00[R/W] B,H,W 00000000	PPER01[R/W] B,H,W 00000000	PPER02[R/W] B,H,W 00000000	PPER03[R/W] B,H,W 00000000	ポートプルアップ/プルダウン許可レジスタ
000EC4H	PPER04[R/W] B,H,W 00000000	PPER05[R/W] B,H,W 00000000	PPER06[R/W] B,H,W 00000000	PPER07[R/W] B,H,W 00000000	
000EC8H	PPER08[R/W] B,H,W 00000000	PPER09[R/W] B,H,W 00000000	PPER10[R/W] B,H,W 00000000	PPER11[R/W] B,H,W 00000000	
000ECCH	PPER12[R/W] B,H,W 00000000	PPER13[R/W] B,H,W 00-00000	—	—	
000ED0H ～ 000EDCH	—	—	—	—	予約
000EE0H	PILR00[R/W] B,H,W 11111111	PILR01[R/W] B,H,W 11111111	PILR02[R/W] B,H,W 11111111	PILR03[R/W] B,H,W 11111111	ポート入力レベル選択レジスタ
000EE4H	PILR04[R/W] B,H,W 11111111	PILR05[R/W] B,H,W 11111111	PILR06[R/W] B,H,W 11111111	PILR07[R/W] B,H,W 11111111	
000EE8H	PILR08[R/W] B,H,W 11111111	PILR09[R/W] B,H,W 11111111	PILR10[R/W] B,H,W 11111111	PILR11[R/W] B,H,W 11111111	
000EECH	PILR12[R/W] B,H,W 11111111	PILR13[R/W] B,H,W 11-11111	—	—	
000EF0H ～ 000EFCH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000F00H	EPILR00[R/W] B,H,W 00000000	EPILR01[R/W] B,H,W 00000000	EPILR02[R/W] B,H,W 00000000	EPILR03[R/W] B,H,W 00000000	拡張ポート入力レベル選択レジスタ
000F04H	EPILR04[R/W] B,H,W 00000000	EPILR05[R/W] B,H,W 00000000	EPILR06[R/W] B,H,W 00000000	EPILR07[R/W] B,H,W 00000000	
000F08H	EPILR08[R/W] B,H,W 00000000	EPILR09[R/W] B,H,W 00000000	EPILR10[R/W] B,H,W 00000000	EPILR11[R/W] B,H,W 00000000	
000F0CH	EPILR12[R/W] B,H,W 00000000	EPILR13[R/W] B,H,W 00-00000	—	—	
000F10H ～ 000F1CH	—	—	—	—	予約
000F20H	PODR00[R/W] B,H,W 00000000	PODR01[R/W] B,H,W 00000000	PODR02[R/W] B,H,W 00000000	PODR03[R/W] B,H,W 00000000	ポート出力駆動レジスタ
000F24H	PODR04[R/W] B,H,W 00000000	PODR05[R/W] B,H,W 00000000	PODR06[R/W] B,H,W 00000000	PODR07[R/W] B,H,W 00000000	
000F28H	PODR08[R/W] B,H,W 00000000	PODR09[R/W] B,H,W 00000000	PODR10[R/W] B,H,W 00000000	PODR11[R/W] B,H,W 00000000	ポート出力駆動レジスタ
000F2CH	PODR12[R/W] B,H,W 00000000	PODR13[R/W] B,H,W 00-00000	—	—	
000F30H	—	—	—	—	予約
000F34H	—	EPODR01 [R/W] B,H,W 00000000	EPODR02 [R/W] B,H,W 00000000	EPODR03 [R/W] B,H,W -0000000	拡張ポート出力駆動レジスタ
000F38H	EPODR06 [R/W] B,H,W 00000000	EPODR07 [R/W] B,H,W 00000000	EPODR08 [R/W] B,H,W 00000000	—	
000F3CH	—	—	—	—	予約
000F40H	PORTEN [R/W] B,H,W -------0	—	—	—	ポート入力許可レジスタ
000F44H ～ 000F6CH	—	—	—	—	予約
000F70H	RCRH0[W] H,W XXXXXXXX	RCRL0[W] B,H,W XXXXXXXX	UDCRH0[R] H,W 00000000	UDCRL0[R] B,H,W 00000000	アップダウンカウンタ 0
000F74H	CCR0[R/W] B,H 00000000 -0001000		—	CSR0[R/W] B 00000000	

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000F78H ～ 000F7CH	—	—	—	—	予約
000F80H	RCRH1[W] H,W XXXXXXXX	RCRL1[W] B,H,W XXXXXXXX	UDCRH1 [R] H,W 00000000	UDCRL1[R] B,H,W 00000000	アップダウンカウンタ 1
000F84H	CCR1[R/W] B,H 00000000 -0001000		—	CSR1[R/W] B 00000000	
000F88H ～ 000F8CH	—	—	—	—	予約
000F90H	OCCP4 [R/W] W 00000000 00000000 00000000 00000000				アウトプットコンペア 4,5
000F94H	OCCP5 [R/W] W 00000000 00000000 00000000 00000000				
000F98H	OCFS45 [R/W] B, H, W ------11	—	OCSH45 [R/W] B, H, W ---0--00	OCSL45[R/W] B, H, W 0000--00	
000F9CH	—	—	—	—	予約
000FA0H	CPCLR2 [R/W] W 11111111 11111111 11111111 11111111				フリーランタイマ 2
000FA4H	TCDT2 [R/W] W 00000000 00000000 00000000 00000000				
000FA8H	TCCSH2 [R/W] B,H,W 0-----00	TCCSL2 [R/W] B,H,W -1-00000	—		
000FACH	CPCLR3 [R/W] W 11111111 11111111 11111111 11111111				フリーランタイマ 3
000FB0H	TCDT3 [R/W] W 00000000 00000000 00000000 00000000				
000FB4H	TCCSH3 [R/W] B,H,W 0-----00	TCCSL3 [R/W] B,H,W -1-00000	—		
000FB8H	CPCLR4 [R/W] W 11111111 11111111 11111111 11111111				フリーランタイマ 4
000FBCH	TCDT4 [R/W] W 00000000 00000000 00000000 00000000				
000FC0H	TCCSH4 [R/W] B,H,W 0-----00	TCCSL4 [R/W] B,H,W -1-00000	—		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
000FC4H	CPCLR5 [R/W] W 11111111 11111111 11111111 11111111				フリーランタイマ 5
000FC8H	TCDT5 [R/W] W 00000000 00000000 00000000 00000000				
000FCCH	TCCSH5 [R/W]B,H,W 0-----00	TCCSL5 [R/W]B,H,W -1-00000	—		
000FD0H	IPCP6 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				インプットキャプチャ 6,7
000FD4H	IPCP7 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000FD8H	ICFS67 [R/W] B, H, W ------00	—	LSYNS1 [R/W] B,H,W ----0000	ICS67 [R/W] B, H, W 00000000	
000FDCH	IPCP8 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				インプットキャプチャ 8,9
000FE0H	IPCP9 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000FE4H	ICFS89 [R/W] B, H, W ------00	—	—	ICS89 [R/W] B, H, W 00000000	
000FE8H	IPCP10 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				インプットキャプチャ 10,11
000FECH	IPCP11 [R] W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
000FF0H	ICFS1011 [R/W] B, H, W ------00	—	—	ICS1011 [R/W] B, H, W 00000000	
000FF4H ～ 000FFCH	—	—	—	—	予約
001000H	SACR [R/W] B,H,W -------0	PICD [R/W] B,H,W ----0011	—	—	クロック制御
001004H ～ 00103CH	—	—	—	—	予約
001040H	—	SGDER0 [R/W] B,H,W 00000000	SGCR0[R/W] B,H,W -0000-0- 000--000		サウンドジェネレータ 0
001044H	SGAR0[R/W] B,H,W 00000000 00000000		SGFR0[R/W] B,H,W 00000000	SGNR0[R/W] B,H,W 00000000	
001048H	SGTCR0[R/W] B,H,W 00000000	SGIDR0[R/W] B,H,W 00000000	SGPCR0[R/W] B,H,W 00000000 11111111		
00104CH	SGDMAR0[W] B,H,W 00000000 00000000 00000000 00000000				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
001050H ～ 00105CH	—	—	—	—	予約
001060H	—	SGDER1[R/W] B,H,W 00000000	SGCR1[R/W] B,H,W -0000-0- 000--000		サウンドジェネレータ 1
001064H	SGAR1[R/W] B,H,W 00000000 00000000		SGFR1[R/W] B,H,W 00000000	SGNR1[R/W] B,H,W 00000000	
001068H	SGTCR1[R/W] B,H,W 00000000	SGIDR1[R/W] B,H,W 00000000	SGPCR1[R/W] B,H,W 00000000 11111111		
00106CH	SGDMAR1[W] B,H,W 00000000 00000000 00000000 00000000				
001070H ～ 00107CH	—	—	—	—	予約
001080H	—	SGDER2[R/W] B,H,W 00000000	SGCR2[R/W] B,H,W -0000-0- 000--000		サウンドジェネレータ 2
001084H	SGAR2[R/W] B,H,W 00000000 00000000		SGFR2[R/W] B,H,W 00000000	SGNR2[R/W] B,H,W 00000000	
001088H	SGTCR2[R/W] B,H,W 00000000	SGIDR2[R/W] B,H,W 00000000	SGPCR2[R/W] B,H,W 00000000 11111111		
00108CH	SGDMAR2[W] B,H,W 00000000 00000000 00000000 00000000				
001090H ～ 00109CH	—	—	—	—	予約
0010A0H	—	SGDER3[R/W] B,H,W 00000000	SGCR3[R/W] B,H,W -0000-0- 000--000		サウンドジェネレータ 3
0010A4H	SGAR3[R/W] B,H,W 00000000 00000000		SGFR3[R/W] B,H,W 00000000	SGNR3[R/W] B,H,W 00000000	
0010A8H	SGTCR3[R/W] B,H,W 00000000	SGIDR3[R/W] B,H,W 00000000	SGPCR3[R/W] B,H,W 00000000 11111111		
0010ACH	SGDMAR3[W] B,H,W 00000000 00000000 00000000 00000000				
0010B0H ～ 0010BCH	—	—	—	—	予約

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0010C0H	—	SGDER4[R/W] B,H,W 00000000	SGCR4[R/W] B,H,W -0000-0- 000--000		サウンドジェネレータ 4
0010C4H	SGAR4[R/W] B,H,W 00000000 00000000		SGFR4[R/W] B,H,W 00000000	SGNR4[R/W] B,H,W 00000000	
0010C8H	SGTCR4[R/W] B,H,W 00000000	SGIDR4[R/W] B,H,W 00000000	SGPCR4[R/W] B,H,W 00000000 11111111		
0010CCH	SGDMAR4[W] B,H,W 00000000 00000000 00000000 00000000				
0010D0H ～ 00112CH	—	—	—	—	予約
001130H	—	—	—	CRCCR[R/W] B,H,W -0000000	CRC 演算
001134H	CRCINIT[R/W] B,H,W 11111111 11111111 11111111 11111111				
001138H	CRCIN[R/W] B,H,W 00000000 00000000 00000000 00000000				
00113CH	CRCR[R] B,H,W 11111111 11111111 11111111 11111111				
001140H ～ 001FFCH	—	—	—	—	予約
002000H	CTRLR0 [R/W] B,H,W --------000-0001		STATR0[R/W] B,H,W -------- 00000000		CAN0 (64msb)
002004H	ERRCNT0 [R] B,H,W 00000000 00000000		BTR0[R/W] B,H,W -0100011 00000001		
002008H	INTR0 [R] B,H,W 00000000 00000000		TESTR0[R/W] B,H,W -------- X00000--		
00200CH	BRPER0 [R/W] B,H,W -------- ----0000		—		
002010H	IF1CREQ0 [R/W] B,H,W 0------- 00000001		IF1CMSK0 [R/W] B,H,W -------- 00000000		
002014H	IF1MSK20 [R/W] B,H,W 11-11111 11111111		IF1MSK10 [R/W] B,H,W 11111111 11111111		
002018H	IF1ARB20 [R/W] B,H,W 00000000 00000000		IF1ARB10 [R/W] B,H,W 00000000 00000000		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
00201C$_H$	IF1MCTR0 [R/W] B,H,W 00000000 0---0000		—		CAN0 (64msb)
002020$_H$	IF1DTA10 [R/W] B,H,W 00000000 00000000		IF1DTA20[R/W] B,H,W 00000000 00000000		
002024$_H$	IF1DTB10 [R/W] B,H,W 00000000 00000000		IF1DTB20 [R/W] B,H,W 00000000 00000000		
002028$_H$, 00202C$_H$	予約				
002030$_H$, 002034$_H$	予約 (IF1 データミラー)				
002038$_H$, 00203C$_H$	予約				
002040$_H$	IF2CREQ0 [R/W] B,H,W 0------- 00000001		IF2CMSK0 [R/W] B,H,W -------- 00000000		
002044$_H$	IF2MSK20 [R/W] B,H,W 11-11111 11111111		IF2MSK10 [R/W] B,H,W 11111111 11111111		
002048$_H$	IF2ARB20 [R/W] B,H,W 00000000 00000000		IF2ARB10 [R/W] B,H,W 00000000 00000000		
00204C$_H$	IF2MCTR0 [R/W] B,H,W 00000000 0---0000		—		
002050$_H$	IF2DTA10 [R/W] B,H,W 00000000 00000000		IF2DTA20 [R/W] B,H,W 00000000 00000000		
002054$_H$	IF2DTB10 [R/W] B,H,W 00000000 00000000		IF2DTB20 [R/W] B,H,W 00000000 00000000		
002058$_H$, 00205C$_H$	予約				
002060$_H$, 002064$_H$	予約 (IF2 データミラー)				
002068$_H$ ～ 00207C$_H$	予約				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
002080$_H$	TREQR20 [R] B,H,W 00000000 00000000		TREQR10 [R] B,H,W 00000000 00000000		CAN0 (64msb)
002084$_H$	TREQR40 [R] B,H,W 00000000 00000000		TREQR30 [R] B,H,W 00000000 00000000		
002088$_H$	—		—		
00208C$_H$	—		—		
002090$_H$	NEWDT20 [R] B,H,W 00000000 00000000		NEWDT10 [R] B,H,W 00000000 00000000		
002094$_H$	NEWDT40 [R] B,H,W 00000000 00000000		NEWDT30 [R]B,H,W 00000000 00000000		
002098$_H$	—		—		
00209C$_H$	—		—		
0020A0$_H$	INTPND20 [R] B,H,W 00000000 00000000		INTPND10 [R] B,H,W 00000000 00000000		
0020A4$_H$	INTPND40 [R] B,H,W 00000000 00000000		INTPND30 [R] B,H,W 00000000 00000000		
0020A8$_H$	—		—		
0020AC$_H$	—		—		
0020B0$_H$	MSGVAL20 [R] B,H,W 00000000 00000000		MSGVAL10 [R] B,H,W 00000000 00000000		
0020B4$_H$	MSGVAL40 [R] B,H,W 00000000 00000000		MSGVAL30 [R] B,H,W 00000000 00000000		
0020B8$_H$	—		—		
0020BC$_H$	—		—		
0020C0$_H$ ～ 0020FC$_H$	予約				
002100$_H$	CTRLR1 [R/W] B,H,W --------000-0001		STATR1[R/W] B,H,W -------- 00000000		CAN1 (32msb)
002104$_H$	ERRCNT1 [R] B,H,W 00000000 00000000		BTR1[R/W] B,H,W -0100011 00000001		
002108$_H$	INTR1 [R] B,H,W 00000000 00000000		TESTR1[R/W] B,H,W -------- X00000--		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
00210C$_H$	BRPER1 [R/W] B,H,W -------- ----0000		—		CAN1 (32msb)
002110$_H$	IF1CREQ1 [R/W] B,H,W 0------- 00000001		IF1CMSK1 [R/W] B,H,W -------- 00000000		
002114$_H$	IF1MSK21 [R/W] B,H,W 11-11111 11111111		IF1MSK11 [R/W] B,H,W 11111111 11111111		
002118$_H$	IF1ARB21 [R/W] B,H,W 00000000 00000000		IF1ARB11 [R/W] B,H,W 00000000 00000000		
00211C$_H$	IF1MCTR1 [R/W] B,H,W 00000000 0---0000		—		
002120$_H$	IF1DTA11 [R/W] B,H,W 00000000 00000000		IF1DTA21 [R/W] B,H,W 00000000 00000000		
002124$_H$	IF1DTB11 [R/W] B,H,W 00000000 00000000		IF1DTB21 [R/W] B,H,W 00000000 00000000		
002128$_H$, 00212C$_H$	予約				
002130$_H$, 002134$_H$	予約 (IF1 データミラー)				
002138$_H$, 00213C$_H$	予約				
002140$_H$	IF2CREQ1 [R/W] B,H,W 0------- 00000001		IF2CMSK1 [R/W] B,H,W -------- 00000000		
002144$_H$	IF2MSK21 [R/W] B,H,W 11-11111 11111111		IF2MSK11 [R/W] B,H,W 11111111 11111111		
002148$_H$	IF2ARB21 [R/W] B,H,W 00000000 00000000		IF2ARB11 [R/W] B,H,W 00000000 00000000		
00214C$_H$	IF2MCTR1 [R/W] B,H,W 00000000 0---0000		—		
002150$_H$	IF2DTA11 [R/W] B,H,W 00000000 00000000		IF2DTA21 [R/W] B,H,W 00000000 00000000		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
002154_H	IF2DTB11 [R/W] B,H,W 00000000 00000000		IF2DTB21 [R/W] B,H,W 00000000 00000000		CAN1 (32msb)
002158_H, $00215C_H$	予約				
002160_H, 002164_H	予約 (IF2 データミラー)				
002168_H ～ $00217C_H$	予約				
002180_H	TREQR21 [R] B,H,W 00000000 00000000		TREQR11 [R] B,H,W 00000000 00000000		
002184_H	—		—		
002188_H	—		—		
$00218C_H$	—		—		
002190_H	NEWDT21 [R] B,H,W 00000000 00000000		NEWDT11 [R] B,H,W 00000000 00000000		
002194_H	—		—		
002198_H	—		—		
$00219C_H$	—		—		
$0021A0_H$	INTPND21 [R] B,H,W 00000000 00000000		INTPND11 [R] B,H,W 00000000 00000000		
$0021A4_H$	—		—		
$0021A8_H$	—		—		
$0021AC_H$	—		—		
$0021B0_H$	MSGVAL21 [R] B,H,W 00000000 00000000		MSGVAL11 [R] B,H,W 00000000 00000000		
$0021B4_H$	—		—		
$0021B8_H$	—		—		
$0021BC_H$	—		—		
$0021C0_H$ ～ $0021FC_H$	予約				

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
002200H	CTRLR2 [R/W] B,H,W -------- 000-0001		STATR2[R/W] B,H,W -------- 00000000		CAN2 (32msb)
002204H	ERRCNT2[R] B,H,W 00000000 00000000		BTR2[R/W] B,H,W -0100011 00000001		
002208H	INTR2[R] B,H,W 00000000 00000000		TESTR2[R/W] B,H,W -------- X00000--		
00220CH	BRPER2 [R/W] B,H,W -------- ----0000		—		
002210H	IF1CREQ2[R/W] B,H,W 0------- 00000001		IF1CMSK2[R/W] B,H,W -------- 00000000		
002214H	IF1MSK22 [R/W] B,H,W 11-11111 11111111		IF1MSK12[R/W] B,H,W 11111111 11111111		
002218H	IF1ARB22 [R/W] B,H,W 00000000 00000000		IF1ARB12[R/W] B,H,W 00000000 00000000		
00221CH	IF1MCTR2[R/W] B,H,W 00000000 0---0000		—		
002220H	IF1DTA12 [R/W] B,H,W 00000000 00000000		IF1DTA22[R/W] B,H,W 00000000 00000000		
002224H	IF1DTB12 [R/W] B,H,W 00000000 00000000		IF1DTB22[R/W] B,H,W 00000000 00000000		
002228H, 00222CH	予約				
002230H, 002234H	予約 (IF1 データミラー)				
002238H, 00223CH	予約				
002240H	IF2CREQ2[R/W] B,H,W 0------- 00000001		IF2CMSK2[R/W] B,H,W -------- 00000000		
002244H	IF2MSK22 [R/W] B,H,W 11-11111 11111111		IF2MSK12[R/W] B,H,W 11111111 11111111		
002248H	IF2ARB22[R/W] B,H,W 00000000 00000000		IF2ARB12[R/W] B,H,W 00000000 00000000		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
00224C$_H$	IF2MCTR2[R/W] B,H,W 00000000 0---0000		—		CAN2 (32msb)
002250$_H$	IF2DTA12[R/W] B,H,W 00000000 00000000		IF2DTA22[R/W] B,H,W 00000000 00000000		
002254$_H$	IF2DTB12[R/W] B,H,W 00000000 00000000		IF2DTB22[R/W] B,H,W 00000000 00000000		
002258$_H$, 00225C$_H$	予約				
002260$_H$, 002264$_H$	予約 (IF2 data データミラー)				
002268$_H$ ～ 00227C$_H$	予約				
002280$_H$	TREQR22[R] B,H,W 00000000 00000000		TREQR12[R] B,H,W 00000000 00000000		
002284$_H$	—		—		
002288$_H$	—		—		
00228C$_H$	—		—		
002290$_H$	NEWDT22[R] B,H,W 00000000 00000000		NEWDT12[R] B,H,W 00000000 00000000		
002294$_H$	—		—		
002298$_H$	—		—		
00229C$_H$	—		—		
0022A0$_H$	INTPND22[R] B,H,W 00000000 00000000		INTPND12[R] B,H,W 00000000 00000000		
0022A4$_H$	—		—		
0022A8$_H$	—		—		
0022AC$_H$	—		—		
0022B0$_H$	MSGVAL22[R] B,H,W 00000000 00000000		MSGVAL12[R] B,H,W 00000000 00000000		

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
0022B4H	—		—		CAN2 (32msb)
0022B8H	—		—		
0022BCH	—		—		
0022C0H ～ 0022FCH	—	—	—	—	予約
002300H	DFCTLR[R/W] B,H,W -0------ --------		—	DFSTR[R/W] B,H,W -----001	ワークフラッシュ
002304H	—	—	—	—	
002308H	FLIFCTLR [R/W] B,H,W ---0--00	—	FLIFFER1 [R/W] B,H,W --------	FLIFFER2 [R/W] B,H,W --------	フラッシュ/ワークフラッシュ
00230CH ～ 0023FCH	—	—	—	—	予約
002400H	SEEARX[R] B,H,W --000000 00000000		DEEARX[R] B,H,W --000000 00000000		XBS RAM ECC 制御レジスタ
002404H	EECSRX[R/W] B,H,W ----0000	—	EFEARX [R/W] B,H,W --000000 00000000		
002408H	—	EFECRX [R/W] B,H,W -------0 00000000 00000000			
00240CH ～ 002FFCH	—	—	—	—	予約
003000H	SEEARA[R] B,H,W -----000 00000000		DEEARA[R] B,H,W -----000 00000000		Backup RAM ECC 制御レジスタ
003004H	EECSRA[R/W] B,H,W ----0000	—	EFEARA[R/W] B,H,W -----000 00000000		
003008H	—	EFECRA [R/W] B,H,W -------0 00000000 00000000			
00300CH ～ 003FFCH	—	—	—	—	予約
004000H ～ 005FFCH	Backup-RAM				Backup RAM 領域
006000H ～ 00FEFCH	—	—	—	—	予約 (00F000H～[S])

アドレス	アドレスオフセット値 / レジスタ名				ブロック
	+0	+1	+2	+3	
$00FF00_H$	DSUCR [R/W] B,H,W -------- -------0		—	—	OCDU [S]
$00FF04_H$ ～ $00FF0C_H$	—	—	—	—	予約 [S]
$00FF10_H$	PCSR [R/W] B,H,W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				OCDU [S]
$00FF14_H$	PSSR [R/W] B,H,W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				
$00FF18_H$ ～ $00FFF4_H$	—	—	—	—	予約 [S]
$00FFF8_H$	EDIR1 [R] B,H,W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				OCDU [S]
$00FFFC_H$	EDIR0 [R] B,H,W XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX				

[S] : システムレジスタです。ユーザモードでこれらのレジスタに読み書きを行った場合は、
不正命令例外(データアクセスエラー) を発生させます。
*3：初期値は型格によって異なります。詳しくは、『CHAPTER 31: クロックスーパバイザ』の章の CSVCR レジスタを確認ください。

49.3 割込みベクター覧

割込みベクター覧について示します。

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を示します。

表 C-1 割込みベクタ

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	RN *1
	10 進	16 進				
リセット	0	00	-	3FC$_H$	000FFFFC$_H$	-
システム予約	1	01	-	3F8$_H$	000FFFF8$_H$	-
システム予約	2	02	-	3F4$_H$	000FFFF4$_H$	-
システム予約	3	03	-	3F0$_H$	000FFFF0$_H$	-
システム予約	4	04	-	3EC$_H$	000FFFEC$_H$	-
FPU 例外	5	05	-	3E8$_H$	000FFFE8$_H$	-
命令アクセス保護違反例外	6	06	-	3E4$_H$	000FFFE4$_H$	-
データアクセス保護違反例外	7	07	-	3E0$_H$	000FFFE0$_H$	-
データアクセスエラー割込み	8	08	-	3DC$_H$	000FFFDC$_H$	-
INTE 命令	9	09	-	3D8$_H$	000FFFD8$_H$	-
命令ブレーク	10	0A	-	3D4$_H$	000FFFD4$_H$	-
システム予約	11	0B	-	3D0$_H$	000FFFD0$_H$	-
システム予約	12	0C	-	3CC$_H$	000FFFCC$_H$	-
システム予約	13	0D	-	3C8$_H$	000FFFC8$_H$	-
不正命令例外	14	0E	-	3C4$_H$	000FFFC4$_H$	-
NMI 要求/ XBS RAM ダブルビットエラー発生/ Backup RAM ダブルビットエラー発生	15	0F	15 (F$_H$) 固定	3C0$_H$	000FFFC0$_H$	-
外部割込み 0-7	16	10	ICR00	3BC$_H$	000FFFBC$_H$	0
外部割込み 8-15	17	11	ICR01	3B8$_H$	000FFFB8$_H$	1
リロードタイマ 0/1/4/5	18	12	ICR02	3B4$_H$	000FFFB4$_H$	2(*2)
リロードタイマ 2/3/6	19	13	ICR03	3B0$_H$	000FFFB0$_H$	3(*2)
マルチファンクションシリアル インタフェース ch.0 (受信完了)/ マルチファンクションシリアル インタフェース ch.0 (ステータス)	20	14	ICR04	3AC$_H$	000FFFAC$_H$	4 (*3)
マルチファンクションシリアル インタフェース ch.0 (送信完了)	21	15	ICR05	3A8$_H$	000FFFA8$_H$	5
マルチファンクションシリアル インタフェース ch.1 (受信完了)/ マルチファンクションシリアル インタフェース ch.1 (ステータス)	22	16	ICR06	3A4$_H$	000FFFA4$_H$	6 (*3)
マルチファンクションシリアル インタフェース ch.1 (送信完了)	23	17	ICR07	3A0$_H$	000FFFA0$_H$	7
LIN-UART2 (受信完了)	24	18	ICR08	39CH	000FFF9C$_H$	8
LIN-UART2 (送信完了)	25	19	ICR09	398$_H$	000FFF98$_H$	9
LIN-UART3 (受信完了)	26	1A	ICR10	394$_H$	000FFF94$_H$	10
LIN-UART3 (送信完了)	27	1B	ICR11	390$_H$	000FFF90$_H$	11
LIN-UART4 (受信完了)	28	1C	ICR12	38C$_H$	000FFF8C$_H$	12

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	RN *1
	10 進	16 進				
LIN-UART4 (送信完了)	29	1D	ICR13	388_H	$000FFF88_H$	13
LIN-UART5 (受信完了)	30	1E	ICR14	384_H	$000FFF84_H$	14
LIN-UART5 (送信完了)	31	1F	ICR15	380_H	$000FFF80_H$	15
LIN-UART6 (受信完了)	32	20	ICR16	$37C_H$	$000FFF7C_H$	16
LIN-UART6 (送信完了)	33	21	ICR17	378_H	$000FFF78_H$	17
CAN0	34	22	ICR18	374_H	$000FFF74_H$	-
CAN1	35	23	ICR19	370_H	$000FFF70_H$	-
CAN2/アップダウンカウンタ 0/ アップダウンカウンタ 1	36	24	ICR20	$36C_H$	$000FFF6C_H$	-
リアルタイムクロック	37	25	ICR21	368_H	$000FFF68_H$	-
サウンドジェネレータ 0 / LIN-UART7 (受信完了)	38	26	ICR22	364_H	$000FFF64_H$	22
サウンドジェネレータ 1 / LIN-UART7 (送信完了)	39	27	ICR23	360_H	$000FFF60_H$	23
PPG0/1/10/11/20/21	40	28	ICR24	$35C_H$	$000FFF5C_H$	24
PPG2/3/12/13/22/23	41	29	ICR25	358_H	$000FFF58_H$	25
PPG4/5/14/15	42	2A	ICR26	354_H	$000FFF54_H$	26
PPG6/7/16/17	43	2B	ICR27	350_H	$000FFF50_H$	27
PPG8/9/18/19	44	2C	ICR28	$34C_H$	$000FFF4C_H$	28
マルチファンクションシリアル インタフェース ch.8 (受信完了)/ マルチファンクションシリアル インタフェース ch.8 (ステータス) / HS_SPI 受信割込み要求	45	2D	ICR29	348_H	$000FFF48_H$	29 (*4)
メインタイマ/サブタイマ/PLL タイマ / マルチファンクションシリアル インタフェース ch.8(送信完了)/ HS_SPI 送信割込み要求	46	2E	ICR30	344_H	$000FFF44_H$	30 (*4)
クロックキャリブレーション ユニット(サブ発振) / サウンドジェネレータ 4/ マルチファンクションシリアル インタフェース ch.9 (受信完了) / マルチファンクションシリアル インタフェース ch.9 (ステータス)	47	2F	ICR31	340_H	$000FFF40_H$	31 (*5)
A/D コンバータ	48	30	ICR32	$33C_H$	$000FFF3C_H$	32
クロックキャリブレーションユニット(CR 発振) / マルチファンクションシリアル インタフェース ch.9 (送信完了)	49	31	ICR33	338_H	$000FFF38_H$	33 (*5)
フリーランタイマ 0/2/4	50	32	ICR34	334_H	$000FFF34_H$	-
フリーランタイマ 1/3/5	51	33	ICR35	330_H	$000FFF30_H$	-
ICU0/6 (取込み)	52	34	ICR36	$32C_H$	$000FFF2C_H$	36
ICU1/7 (取込み)	53	35	ICR37	328_H	$000FFF28_H$	37
ICU2/8(取込み)	54	36	ICR38	324_H	$000FFF24_H$	38
ICU3/9(取込み)	55	37	ICR39	320_H	$000FFF20_H$	39
ICU4/10(取込み)	56	38	ICR40	$31C_H$	$000FFF1C_H$	40
ICU5/11(取込み)	57	39	ICR41	318_H	$000FFF18_H$	41
OCU0/1/6/7/10/11(一致)	58	3A	ICR42	314_H	$000FFF14_H$	42
OCU2/3/4/5/8/9(一致)	59	3B	ICR43	310_H	$000FFF10_H$	43

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	RN *1
	10 進	16 進				
ベースタイマ 0 IRQ0 / ベースタイマ 0 IRQ1 / サウンドジェネレータ 2	60	3C	ICR44	$30C_H$	$000FFF0C_H$	44
ベースタイマ 1 IRQ0 / ベースタイマ 1 IRQ1 / サウンドジェネレータ 3 / XBS RAM シングルビットエラー発生/ Backup RAM シングルビットエラー発生	61	3D	ICR45	308_H	$000FFF08_H$	45 (*6)
DMAC0/1/2/3/4/5/6/7/8/9/10/11/12/13/14/15	62	3E	ICR46	304_H	$000FFF04_H$	-
遅延割込み	63	3F	ICR47	300_H	$000FFF00_H$	-
システム予約 (REALOS®*7 にて使用)	64	40	-	$2FC_H$	$000FFEFC_H$	-
システム予約 (REALOS にて使用)	65	41	-	$2F8_H$	$000FFEF8_H$	-
INT 命令で使用	66 | 255	42 | FF	-	$2F4_H$ | 000_H	$000FFEF4_H$ | $000FFC00_H$	-

*1 : RN 番号割り当てのないペリフェラルからの割込みによる DMA 転送要求には対応していません。
*2 : リロードタイマ ch.4,5,6 は割込みによる DMA 転送に対応していません。
*3 : マルチファンクションシリアルインタフェースのステータスは、I^2C 受信による DMA 転送に対応していません。
*4 : HS_SPI は割込みによる DMA 転送には対応していません。
*5 : クロックキャリブレーションユニットは割込みによる DMA 転送には対応していません。
*6 : RAM ECC ビットエラーによる割込みによる DMA 転送には対応していません。
*7 : REALOS はサイプレスの登録商標です。

49.4 CPU 状態における端子状態

CPU 状態における端子状態について示します。

表 D-1 端子状態表

Pin Number	Pin Function Port Number/Bidirection Function /Output Function/Input Function	Pad I/O Type	Input level	Active Level	Pull-up/Down	External Reset Factor 1(*1) While the reset factor is active Issuing internal reset	External Reset Factor 1(*1) After the reset factor has gone Issuing internal reset (including oscillation stabilization waiting)	External Reset Factor 1(*1) After the reset factor has gone After internal reset releasing	External Reset Factor 2(*2) While the reset factor is active Before internal reset issuance	External Reset Factor 2(*2) While the reset factor is active Issuing internal reset	External Reset Factor 2(*2) After the reset factor has gone Before internal reset issuance	External Reset Factor 2(*2) After the reset factor has gone Issuing internal reset (including oscillation stabilization waiting)	External Reset Factor 2(*2) After the reset factor has gone After internal reset releasing	Internal Reset Factor(*4)	Internal Reset Factor(*5) Issuing internal reset (including oscillation stabilization waiting)	Internal Reset Factor(*5) After internal reset releasing	Sleep Mode SLVL[1:0]=0x	Sleep Mode SLVL[1:0]=1x	Stop Mode SLVL[1:0]=0x	Stop Mode SLVL[1:0]=1x	Watch Mode SLVL[1:0]=0x	Watch Mode SLVL[1:0]=1x	After entry to SPL=1
2	P015/DI9_0/SEG5/CD1_1/INT13_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Status before entering shutdown is retained.	Status before entering shutdown is retained.	Last status retained(*6)	Last status retained	Last status retained	Hi-Z/Input Blocked(*10)	Last status retained	Hi-Z/Input Blocked(*10)	Yes
3	P016/DI0_0/SEG6/CD2_1/INT14_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
4	P017/DI1_0/SEG7/CD3_1/INT15_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
5	P020/ADI/SEG8/ICU6_0/OCU0_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
6	P021/CSI0/SEG9/ICU7_0/OCU1_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
7	P022/CSI1/SEG10_/ICU8_0/OCU2_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
8	P023/RDX/SEG11/ICU9_0/OCU3_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
9	P024/WRDX/SEG12/ICU10_0/OCU4_1_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
10	P025/WR1X/SEG13_/ICU11_0/OCU5_1_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
11	P026/A00/SEG14/SPI_CS3/SIN6_1/OCU6_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
12	P027/A01/SEG15/SPI_CS2/SOT6_1/OCU8_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
13	P030/A02/SEG16/SPI_CS1/SCK6_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
14	P031/A03/SEG17/SPI_CS0/SIN6_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
15	P032/A04/SEG18/SPI_SIO3/SOT6_0/OCU7_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
16	P033/A05/SEG19/SPI_SIO2/SCK6_0/OCU9_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
17	P034/A06/SEG20/SPI_SIO1/SIN6_0/OCU9_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
18	P035/A07/SEG21/SPI_SIO0/SIN7_0/OCU8_1	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
19	P036/A08/SEG22/PPG11_0/SPI_CLK/SCK6_0	H	CMOS/AMTTL	-	D	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
20	P037/A09/SEG23/ST0/PPG12_0/SIN0_0	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
21	P040/A10/SEG24/ST1/PPG13_0/SOT1_0	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
22	P041/A11/SEG25/ST2/PPG14_0/SCK7_0	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
23	P042/A12/SEG26/ST3/PPG15_0/AIN0_0	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
24	P043/A13/SEG27/ST4/BIN0_0/SCK4_0/OCU0_1	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
25	P044/A14/SEG28/ST5/ZIN0_0/SOT4_0/OCU1_1	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
26	P045/A15/SEG29/ST6/AIN1_0/SIN4_0	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
27	P046/A16/SEG30/ST7/BIN1_0/SOT4_0	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
28	P047/A17/SEG31/ST8/ZIN1_0/SCK4_2	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
29	P050/A18/COM0/OCU8_1	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
30	P051/A19/COM1/OCU9_1	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
31	P052/A20/COM2/OCU10_1	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
32	P053/A21/COM3/OCU11_1	I	CMOS/AMTTL	-	UD	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"/Input Blocked	Output "L"	Output "L"/Input Blocked	Output "L"	Output "L"	Output "L"/Input Blocked	Output "L"/Input Blocked									
33	P054/SYSCLK/VD/FRCK6_1	I2	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
34	P055/CS2X/V1/FRCK1_1	I2	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
35	P056/CS2X/V2/FRCK2_1	I2	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
36	P057/RDY/V3/FRCK3_1	I3	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
37	P060/PWM1P0/AN0	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
38	P061/PWM1M0/AN0_0/SIN1_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
39	P062/PWM2P0/AN10/ZIN1_1/SOT1_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
40	P063/PWM2M0/AN11/BIN1_1/SCK1_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
41	P064/PWM1P1/AN12/AIN1_1/SIN0_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
42	P065/PWM1M1/AN13/ZIN0_1/SOT0_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
43	P066/PWM2P1/AN14/BIN0_1/SCK0_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
44	P067/PWM2M1/AN15/AIN0_1/SIN5_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
45	P070/PWM1P2/AN16/SOT5_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
46	P071/PWM1M2/AN17/SCK5_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
47	P072/PWM2P2/AN18/ICU11_1/SIN8_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
48	P073/PWM2M2/AN19/ICU10_1/SOT8_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
49	P074/PWM1P3/AN20/PPG10_1/ICU8_1/SCK8_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
50	P075/PWM1M3/AN21/PPG13_1/ICU9_1/SIN7_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
51	P076/PWM2P3/AN22/PPG14_1/ICU7_1/SOT7_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
52	P077/PWM2M3/AN23/PPG15_1/ICU6_1/SCK7_1	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
53	P080/PWM1P4/AN24/SIN8_0/PPG16_0/AIN0_2	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
54	P081/PWM1M4/AN25/SOT8_0/PPG17_0/BIN0_2	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
55	P082/PWM2P4/AN26/SCK8_0/PPG18_0/ZIN0_2	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
56	P083/PWM3M4/AN27/ICU3_2/PPG19_0	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
57	P084/PWM1P5/AN28/ICU2_2/PPG20_0	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
58	P085/PWM1M5/AN29/ICU2_2/PPG21_0	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
59	P086/PWM2P5/AN30/ICU3_2/PPG22_0	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
60	P087/PWM2M5/AN31/ICU4_2/PPG23_0	K	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
61	P090/ADTG/PPG8_2	M	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
62	P100/SIN4_1/AN0/TIN0_1/PPG8_0	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
63	P101/SOT4_1/AN1/TIN1_1/PPG9_0	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
64	P102/SCK4_1/AN2/TIN2_1/PPG10_0/ICU6_2	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
65	P103/SIN5_1/AN3/TIN3_1/PPG11_0/ICU7_2	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
66	P104/SOT5_1/AN4/TOT0_1/PPG12_1/ICU8_2	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
67	P105/SCK5_1/AN5/TOT1_1/PPG13_1/ICU9_2	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
68	P106/AN6/PPG4_1/ICU10_2/SOA4_1	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
69	P107/ANT/PPG5_1/DA01/ICU11_2/SOD4_1	L	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
70	P123/OCU3_0/PPG8_2/DA00/AN29	S	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
71	P122/OCU2_0/SCK5_0/TOT3_0/PPG17_2/AN28	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
72	P121/FRCK0_0/SIN5_0/INT7_0/TOT2_0/PPG16_2/AN27	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
73	P120/FRCK1_0/SIN5_0/INT6_0/TOT1_0/PPG15_2/AN26	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
74	P117/SCK4_0/TIN2_0/SIG2/TIO4/FRCK2_0/AN25	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
75	P118/SOT4_0/TIN3_0/SOA1/FRCK3_0/AN24	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
76	P115/SIN4_0/TIN2_0/SIG2/FRCK4_0/AN33	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
77	P114/SCK3_0/TIN1_0/ICU5_1/SOA2/TRG3/AN32	J	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
78	P137 (*13)	M	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
79	P136 (*13)	M	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
80	P097/INIT/SOT1_0/INT8_0/TIN0_0/ICU4_1/PPG0_1	M	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
81	P094/SCK1/SIN3_0/INT15_0/ICU1_1/PPG6_1	M	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
82	P093/SOA1/SOT2_0/INT14_0/ICU3_1/PPG5_1	M	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
83	P092/SOA2/SCK2_0/INT13_0/TOT3_1/ICU2_1/PPG7_1	M	CMOS/AMTTL	-	UD	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Enabled	Hi-Z/Input Enabled	Hi-Z/Input Blocked	Hi-Z/Input Blocked									
84	P091/SOA0/SIN2_0/INT12_0/TOT2_1/ICU0_1/PPG4_1	M	CMOS/AMTTL	-	UD	Hi-Z/

49.5 主な変更内容

ページ	場所	変更箇所
Revision 3.0		
-	-	Initial release
Revision 3.1		
-	-	社名変更および記述フォーマットの変換
-	-	空白ページ削除とそれに伴うページ番号修正
28～30	1.1.8	URL の記述を削除
90	2.2.1	URL の記述を追加

改訂履歴

Document Revision History

ドキュメント名: FR ファミリ FR81S, CY91570 シリーズ 32 ビット・マイクロコントローラ ハードウェアマニュアル ドキュメント番号: 002-05575			
版	**発行日**	**変更者**	**変更内容**
**	01/28/2015	TORS	サイプレスとしてスパンション MB91570_MN705-00011-3v1-Jをドキュメントコード 002-05575 に登録しました。 本版の内容およびフォーマットに変更はありません。
*A	05/11/2017	TORS	最新のテンプレートへ更新しました。
*B	01/17/2019	TORS	シリーズ名と型格を変更しました。 MB91570 -> CY91570 パッケージ外形寸法図を変更しました。