

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以って NEC エレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日
ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事事務の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。

標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット

高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）

特定水準： 航空機器、航空宇宙機器、海中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

改訂一覧は表紙をクリックして直接ご覧になれます。
改訂一覧は改訂箇所をまとめたものであり、詳細については、
必ず本文の内容をご確認ください。

SH7660

ハードウェアマニュアル

ルネサス32ビットRISCマイクロコンピュータ
SuperH RISC engineファミリ

SH7660

HD6417660

本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりましては、事前に弊社営業窓口で最新の情報をご確認頂きますとともに、弊社ホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意下さい。
5. 本資料に記載した情報は、正確を期すため慎重に制作したのですが、万一本資料の記述の誤りに起因する損害がお客様に生じた場合においても、弊社はその責任を負いません。
6. 本資料に記載の製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断して下さい。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会下さい。なお、上記用途に使用されたことにより発生した損害等について弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないで下さい。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
 - 1) 生命維持装置。
 - 2) 人体に埋め込み使用するもの。
 - 3) 治療行為（患部切り出し、薬剤投与等）を行なうもの。
 - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて製品をご使用された場合の故障および事故につきましては、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。弊社製品の故障または誤動作が生じた場合も人身事故、火災事故、社会的損害などを生じさせないよう、お客様の責任において冗長設計、延焼対策設計、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特にマイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願い致します。
11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断り致します。
13. 本資料に関する詳細についてのお問い合わせ、その他お気づきの点等がございましたら弊社営業窓口までご照会下さい。

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認ください。

同じグループのマイコンでも型名が違うと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

本書の構成

本書は、以下の構成で制作しています。

1. 製品ご使用上の注意事項
2. 本書の構成
3. はじめに
4. 目次
5. 概要
6. 各機能モジュールの説明
 - ・ CPU およびシステム制御系
 - ・ 内蔵周辺モジュール

各モジュールの機能説明の構成は、モジュール毎に異なりますが、一般的には、特長、 入出力端子、 レジスタの説明、 動作説明、 使用上の注意事項、等の節で構成されています。

本 LSI を用いた応用システムを設計する際、注意事項を十分確認の上設計してください。各章の本文中には説明に対する注意事項と、各章の最後には使用上の注意事項があります。必ずお読みください。(使用上の注意事項は必要により記載されます。)

7. レジスタ一覧
8. 電気的特性
9. 付録
10. 本版で改訂された箇所 (改訂版のみ適用)

改訂来歴は、前版の記載内容について訂正・追加された主な箇所についてまとめたものです。改訂内容の全てについて記載したものではありませんので、詳細については、本書の本文上でご確認ください。

11. 索引

はじめに

SH7660 は、ルネサス テクノロジオリジナルの RISC 方式の CPU をコアにして、システム構成に必要な周辺機能を集積した RISC マイクロプロセッサです。

対象者 このマニュアルは、SH7660 を用いた応用システムを設計するユーザーを対象としています。
このマニュアルを使用される読者には、電気回路、論理回路、およびマイクロコンピュータに関する基本的な知識を必要とします。

目的 このマニュアルは、SH7660 のハードウェア機能と電気的特性をユーザーに理解していただくことを目的にしています。
なお、実行命令の詳細については、「SH-3/SH-3E/SH3-DSP プログラミングマニュアル」に記載しておりますので、あわせてご覧ください。

読み方

- 対象製品と製品略称の表記について。

本書では、下記の製品について説明をします。

製品分類と製品略称	基本製品型名
基本分類	
SH7660	HD6417660

- 機能全体を理解しようとするとき。
目次にしたがって読んでください。
本書は、大きく分類すると、CPU、システム制御機能、周辺機能、電気的特性の順に構成されています。

- CPU機能の詳細を理解したいとき。
別冊の「SH-3/SH-3E/SH3-DSP プログラミングマニュアル」を参照してください。

凡例 レジスタ表記 : シリアルコミュニケーションなど、同一または類似した機能が複数チャンネルに存在する場合に、次の表記を使用します。

XXX_N (XXX は基本レジスタ名称、N はチャンネル番号)

ビット表記 : 左側が上位ビット、右側が下位ビットの順に表記します。

数字の表記 : 2進数は B'xxxx、16進数は H'xxxx、10進数は xxxx

信号の表記 : ローアクティブの信号にはオーバーバーを付けます。xxxx

関連資料一覧 ウェブ・サイトに最新資料を掲載しています。ご入手の資料が最新版であるかを確認してください。
(<http://japan.renesas.com/>)

- SH7660に関するユーザズマニュアル

資料名	資料番号
SH7660 ハードウェアマニュアル	本マニュアル
SH-3/SH-3E/SH3-DSP プログラミングマニュアル	ADJ - 602 - 120

- 開発ツール関連ユーザズマニュアル

資料名	資料番号
SuperH RISC engine C/C++ コンパイラ、アセンブラ、最適化リンケージエディタ V.9.00 ユーザズマニュアル	RJJ10B0156
SuperH RISC engine High-Performance Embedded Workshop 3 ユーザズマニュアル	RJJ10B0028
SuperH RISC engine High-Performance Embedded Workshop 3 チュートリアル	RJJ10B0026

- アプリケーションノート

資料名	資料番号
SuperH RISC engine C/C++コンパイラパッケージアプリケーションノート	RJJ05B0557

略語の説明

ALU	Arithmetic Logic Unit 演算論理回路
ASE	Adaptive System Evaluator 適応システム評価
AUD	Advanced User Debugger アドバンスユーザデバッガ
bps	bit per second ビット数/秒
BSC	Bus State Controller バスステートコントローラ
CODEC	Coder-Decoder コーデック
CPG	Clock Pulse Generator クロック発振器
CPU	Central Processing Unit 中央制御装置
DAC	Digital to Analog Converter デジタル - アナログ変換器
DMAC	Direct Memory Access Controller ダイレクトメモリアクセスコントローラ
DSP	Digital Signal Processor デジタルシグナルプロセッサ
ESD	Electrostatic Discharge 静電放電
FIFO	First-In First-Out 先入れ先出しレジスタ
Hi-Z	High Impedance ハイインピーダンス
H-UDI	User Debug Interface ユーザデバッグインタフェース
INTC	Interrupt Controller 割り込みコントローラ
LSB	Least Significant Bit 最下位ビット
MSB	Most Significant Bit 最上位ビット

PC	Program Counter プログラムカウンタ
PFC	Pin Function Controller ピンファンクションコントローラ
PLL	Phase Locked Loop 位相ロックループ
RAM	Random Access Memory ランダムアクセスメモリ
RF	Radio Frequency 無線周波数
RISC	Reduced Instruction Set Computer 縮小命令セットコンピュータ
ROM	Read Only Memory リードオンリーメモリ
SIOF	Serial Input Output with FIFO シリアルインタフェース FIFO 付き
SCIF	Serial Communication Interface with FIFO FIFO 内蔵シリアルコミュニケーションインタフェース
SOF	Start Of Frame フレーム開始信号
TAP	Test Access Port テスト端子
T.B.D.	To Be Determined 未定
TLB	Translation Lookaside Buffer 変換ルックアサイドバッファ
UBC	User Break Controller ユーザブレイクコントローラ
USB	Universal Serial Bus ユニバーサルシリアルバス
WDT	Watch Dog Timer ウォッチドッグタイマ

本版で改訂された箇所

修正項目	ページ	修正内容（詳細はマニュアル参照）										
全体	-	用語を変更 （修正前）シリアル I/O （修正後）シリアルインタフェース （修正前）D/A 変換器 （修正後）D/A コンバータ										
	-	HD157100NP を削除										
1.1 SH7660 の特長 表 1.1 SH7660 の特長	1-2	表を修正 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%;">項 目</th> <th style="width: 70%;">特 長</th> </tr> </thead> <tbody> <tr> <td>DSP ユニット</td> <td> <ul style="list-style-type: none"> • 16 ビット命令、および 32 ビット命令の混在可能 • 32 または 40 ビットの内部データバスを内蔵 • 乗算器、ALU、バレルシフトに対応 • 16 ビット×16 ビットに対応する 32 ビットワンサイクル乗算器を内蔵 • 大容量の DSP データレジスタファイルをサポート • 6 本の 32 ビットデータレジスタ • 2 本の 40 ビットデータレジスタ • DSP データバス用の拡張ハードアーキテクチャをサポート • 2 本のデータバス • 1 本の命令バス • 最大 4 つのバラレル演算を実行可能 • ALU、乗算、2 つのロード/ストア • 2 つのメモリアクセス用のアドレスを生成するための 2 本のアドレスユニットを装備 • DSP データアドレッシングモードをサポート • インクリメント、およびインデクシング（モジュロアドレッシングあり/なし） • ゼロオーバーヘッドリピートループ制御に対応 • 条件付実行命令に対応 • ユーザ DSP モードおよび特権 DSP モードをサポート </td> </tr> </tbody> </table>	項 目	特 長	DSP ユニット	<ul style="list-style-type: none"> • 16 ビット命令、および 32 ビット命令の混在可能 • 32 または 40 ビットの内部データバスを内蔵 • 乗算器、ALU、バレルシフトに対応 • 16 ビット×16 ビットに対応する 32 ビットワンサイクル乗算器を内蔵 • 大容量の DSP データレジスタファイルをサポート • 6 本の 32 ビットデータレジスタ • 2 本の 40 ビットデータレジスタ • DSP データバス用の拡張ハードアーキテクチャをサポート • 2 本のデータバス • 1 本の命令バス • 最大 4 つのバラレル演算を実行可能 • ALU、乗算、2 つのロード/ストア • 2 つのメモリアクセス用のアドレスを生成するための 2 本のアドレスユニットを装備 • DSP データアドレッシングモードをサポート • インクリメント、およびインデクシング（モジュロアドレッシングあり/なし） • ゼロオーバーヘッドリピートループ制御に対応 • 条件付実行命令に対応 • ユーザ DSP モードおよび特権 DSP モードをサポート 						
項 目	特 長											
DSP ユニット	<ul style="list-style-type: none"> • 16 ビット命令、および 32 ビット命令の混在可能 • 32 または 40 ビットの内部データバスを内蔵 • 乗算器、ALU、バレルシフトに対応 • 16 ビット×16 ビットに対応する 32 ビットワンサイクル乗算器を内蔵 • 大容量の DSP データレジスタファイルをサポート • 6 本の 32 ビットデータレジスタ • 2 本の 40 ビットデータレジスタ • DSP データバス用の拡張ハードアーキテクチャをサポート • 2 本のデータバス • 1 本の命令バス • 最大 4 つのバラレル演算を実行可能 • ALU、乗算、2 つのロード/ストア • 2 つのメモリアクセス用のアドレスを生成するための 2 本のアドレスユニットを装備 • DSP データアドレッシングモードをサポート • インクリメント、およびインデクシング（モジュロアドレッシングあり/なし） • ゼロオーバーヘッドリピートループ制御に対応 • 条件付実行命令に対応 • ユーザ DSP モードおよび特権 DSP モードをサポート 											
1.2 ブロック図	1-7	記号説明を修正 USBH/USBF : USB ホスト / USB ファンクション コントローラ										
1.3 ピン配置図 表 1.2 ピンごとの機能および初期状態一覧表	1-12	表を修正 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 20%;">端子番号</th> <th style="width: 30%;">端子名</th> <th style="width: 50%;">機 能</th> </tr> </thead> <tbody> <tr> <td>J1</td> <td>TRST</td> <td>H-UDI リセット入力。内部でプルアップされています。本端子への信号供給については、「21.9 基板設計の制約事項」を参照ください。</td> </tr> </tbody> </table>	端子番号	端子名	機 能	J1	TRST	H-UDI リセット入力。内部でプルアップされています。本端子への信号供給については、「21.9 基板設計の制約事項」を参照ください。				
端子番号	端子名	機 能										
J1	TRST	H-UDI リセット入力。内部でプルアップされています。本端子への信号供給については、「21.9 基板設計の制約事項」を参照ください。										
1.4 端子の機能 表 1.3 SH7660 端子機能	1-18	表を修正 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">分類</th> <th style="width: 10%;">端子名</th> <th style="width: 10%;">入出力</th> <th style="width: 20%;">名称</th> <th style="width: 50%;">機 能</th> </tr> </thead> <tbody> <tr> <td>割り込み</td> <td>IRQOUT</td> <td>出力</td> <td>割り込み要求通知</td> <td>割り込み要因が発生したことを外部デバイスに通知できます。</td> </tr> </tbody> </table>	分類	端子名	入出力	名称	機 能	割り込み	IRQOUT	出力	割り込み要求通知	割り込み要因が発生したことを外部デバイスに通知できます。
分類	端子名	入出力	名称	機 能								
割り込み	IRQOUT	出力	割り込み要求通知	割り込み要因が発生したことを外部デバイスに通知できます。								
	-	「6.3.4 スリープモード」を削除										
	-	「7.3.4 スリープモード」を削除										

修正項目	ページ	修正内容（詳細はマニュアル参照）																
8.3.4 割り込みコントロールレジスタ 2 (ICR2)	8-8	<p>説明を追加</p> <p>ICR2 は、外部割り込み入力端子 IRQ7 に対して立ち上がりエッジ、立ち下がりエッジ、ローレベル、およびハイレベルの検出モードを個別に指定する 16 ビットのレジスタです。ただし、本 LSI では IRQ7 には、USBF (USB ファンクションコントローラ) のスタンバイ復帰割り込み要求信号が接続されていますので、別の用途には使用できません。そのため、検出方法は立ち下がりエッジのみに固定されているので、これ以外は設定しないでください。</p> <p>表を修正</p> <table border="1"> <thead> <tr> <th>ビット 2n+1</th> <th>ビット 2n</th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> <td>割り込み要求を IRQ7 入力の立ち上がりエッジで検出する。ただし、本 LSI では設定禁止。</td> </tr> <tr> <td>1</td> <td>0</td> <td>割り込み要求を IRQ7 入力のローレベルで検出する。ただし、本 LSI では設定禁止。</td> </tr> <tr> <td>1</td> <td>1</td> <td>割り込み要求を IRQ7 入力のハイレベルで検出する。ただし、本 LSI では設定禁止。</td> </tr> </tbody> </table> <p>注を削除</p>	ビット 2n+1	ビット 2n		0	1	割り込み要求を IRQ7 入力の立ち上がりエッジで検出する。ただし、本 LSI では設定禁止。	1	0	割り込み要求を IRQ7 入力のローレベルで検出する。ただし、本 LSI では設定禁止。	1	1	割り込み要求を IRQ7 入力のハイレベルで検出する。ただし、本 LSI では設定禁止。				
ビット 2n+1	ビット 2n																	
0	1	割り込み要求を IRQ7 入力の立ち上がりエッジで検出する。ただし、本 LSI では設定禁止。																
1	0	割り込み要求を IRQ7 入力のローレベルで検出する。ただし、本 LSI では設定禁止。																
1	1	割り込み要求を IRQ7 入力のハイレベルで検出する。ただし、本 LSI では設定禁止。																
8.4.2 IRQ 割り込み	8-11	<p>説明を修正</p> <p>ICR1、ICR2 に上書きする際、IRQ 端子の状態によっては IRQ 割り込みが誤検出される可能性があります。これを避けるため、まず接続先を停止するか、接続先で信号マスクをかけ IRQ 端子の状態を非アクティブ側としておき、変化しないようにしてください。それから、割り込みマスクレジスタ IMR0 の設定、ステータスレジスタ SR のブロック (BL) ビットの設定等を実行して割り込みをマスク状態にしてから、割り込み要求レジスタ 0 (IRR0) を読み出した後に 0 を書き込んで不正な割り込みをクリアしたうえで、ICR1、ICR2 を上書きしてください。その後に接続先の起動やマスクの解除を順次してください。</p>																
10.4.3 DMA 転送要求 (3) 内蔵周辺モジュールリクエストモード	10-18	<p>説明を修正</p> <p>DMA 転送要求信号は、SCIF0、SCIF1、USBF からの送信データエンブティ転送要求と受信データフル転送要求および SIOF からの送信データ転送リクエストと受信データ転送リクエストがあります。</p>																
表 10.6 RS3-0 ビットによる内蔵周辺モジュールリクエストモードの選択	10-19	<p>表を修正</p> <table border="1"> <thead> <tr> <th rowspan="2">CHCR</th> <th colspan="2">DMARS</th> <th rowspan="2">DMA 転送要求元</th> <th rowspan="2">DMA 転送要求信号</th> </tr> <tr> <th>RS[3:0]</th> <th>MID</th> <th>RID</th> </tr> </thead> <tbody> <tr> <td rowspan="2">1000</td> <td rowspan="2">010101</td> <td>01</td> <td>SIOF 送信部</td> <td>TDREQ (送信データ転送リクエスト)</td> </tr> <tr> <td>10</td> <td>SIOF 受信部</td> <td>RDREQ (受信データ転送リクエスト)</td> </tr> </tbody> </table>	CHCR	DMARS		DMA 転送要求元	DMA 転送要求信号	RS[3:0]	MID	RID	1000	010101	01	SIOF 送信部	TDREQ (送信データ転送リクエスト)	10	SIOF 受信部	RDREQ (受信データ転送リクエスト)
CHCR	DMARS			DMA 転送要求元	DMA 転送要求信号													
	RS[3:0]	MID	RID															
1000	010101	01	SIOF 送信部	TDREQ (送信データ転送リクエスト)														
		10	SIOF 受信部	RDREQ (受信データ転送リクエスト)														

修正項目	ページ	修正内容（詳細はマニュアル参照）																																	
10.4.5 DMA 転送の種類	10-29	表を修正 <table border="1" data-bbox="600 343 1203 844"> <thead> <tr> <th data-bbox="600 343 691 407">アドレスモード</th> <th data-bbox="691 343 1081 382">転送区間</th> <th data-bbox="1081 343 1203 407">使用可能チャンネル</th> </tr> </thead> <tbody> <tr> <td data-bbox="600 407 691 440">デュアル</td> <td data-bbox="691 407 1081 440">外部メモリと外部メモリ</td> <td data-bbox="1081 407 1203 440">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 440 1081 473">外部メモリとメモリマップト外部デバイス</td> <td data-bbox="1081 440 1203 473">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 473 1081 537">メモリマップト外部デバイスとメモリマップト外部デバイス</td> <td data-bbox="1081 473 1203 537">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 537 1081 569">外部メモリと内蔵周辺モジュール</td> <td data-bbox="1081 537 1203 569">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 569 1081 633">メモリマップト外部デバイスと内蔵周辺モジュール</td> <td data-bbox="1081 569 1203 633">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 633 1081 666">内蔵周辺モジュールと内蔵周辺モジュール</td> <td data-bbox="1081 633 1203 666">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 666 1081 699">X/Yメモリ、UメモリとX/Yメモリ、Uメモリ</td> <td data-bbox="1081 666 1203 699">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 699 1081 763">X/Yメモリ、Uメモリとメモリマップト外部デバイス</td> <td data-bbox="1081 699 1203 763">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 763 1081 795">X/Yメモリ、Uメモリと内蔵周辺モジュール</td> <td data-bbox="1081 763 1203 795">0~3^{※3}</td> </tr> <tr> <td></td> <td data-bbox="691 795 1081 844">X/Yメモリ、Uメモリと外部メモリ</td> <td data-bbox="1081 795 1203 844">0~3^{※3}</td> </tr> </tbody> </table>	アドレスモード	転送区間	使用可能チャンネル	デュアル	外部メモリと外部メモリ	0~3 ^{※3}		外部メモリとメモリマップト外部デバイス	0~3 ^{※3}		メモリマップト外部デバイスとメモリマップト外部デバイス	0~3 ^{※3}		外部メモリと内蔵周辺モジュール	0~3 ^{※3}		メモリマップト外部デバイスと内蔵周辺モジュール	0~3 ^{※3}		内蔵周辺モジュールと内蔵周辺モジュール	0~3 ^{※3}		X/Yメモリ、UメモリとX/Yメモリ、Uメモリ	0~3 ^{※3}		X/Yメモリ、Uメモリとメモリマップト外部デバイス	0~3 ^{※3}		X/Yメモリ、Uメモリと内蔵周辺モジュール	0~3 ^{※3}		X/Yメモリ、Uメモリと外部メモリ	0~3 ^{※3}
アドレスモード	転送区間	使用可能チャンネル																																	
デュアル	外部メモリと外部メモリ	0~3 ^{※3}																																	
	外部メモリとメモリマップト外部デバイス	0~3 ^{※3}																																	
	メモリマップト外部デバイスとメモリマップト外部デバイス	0~3 ^{※3}																																	
	外部メモリと内蔵周辺モジュール	0~3 ^{※3}																																	
	メモリマップト外部デバイスと内蔵周辺モジュール	0~3 ^{※3}																																	
	内蔵周辺モジュールと内蔵周辺モジュール	0~3 ^{※3}																																	
	X/Yメモリ、UメモリとX/Yメモリ、Uメモリ	0~3 ^{※3}																																	
	X/Yメモリ、Uメモリとメモリマップト外部デバイス	0~3 ^{※3}																																	
	X/Yメモリ、Uメモリと内蔵周辺モジュール	0~3 ^{※3}																																	
	X/Yメモリ、Uメモリと外部メモリ	0~3 ^{※3}																																	
10.4.6 バスサイクルのステート数とDREQ端子のサンプリングタイミング (2) DREQ端子のサンプリングタイミング	10-32	説明を修正 8ビット、16ビット外部デバイスに16バイト転送やロングワードアクセスしたり、8ビット外部デバイスにワードアクセスをする場合は、DMA転送単位が複数のバスサイクルに分割されます。 このとき、バスサイクル間でCSがネゲートする設定を選択していると、データをアライメントするためにCSと同様にDACK出力およびTEND出力が分割されます。この例を図10.18に示します。																																	
	10-33	説明を追加 この分割されたDACK出力ではDREQのサンプリングを正しく検出できません。 外部リクエスト使用時には、下記いずれかの方法でDACK(CS _n)が分割されるのを回避してください。 1. DACK(CS _n)が分割されない設定にする。(例えばDMACのCHCR:RS3~RS0の設定を外部リクエスト、シングルアドレスモードに設定、BSCのCMNCR:DMAIWAに0を設定、転送対象空間のCSnBCR:IWW1、IWW0およびIWRRS1、IWRRS0にサイクル間ウェイト0を設定、CSnWCRのWMを外部ウェイト入力無視に設定) 2. 1転送単位の転送サイズを外部デバイスのバス幅以下に設定する。																																	
10.5 使用上の注意事項 (4) Uメモリ、およびX/Yメモリへのアクセスについて (5) 外部リクエスト使用時の注意事項	10-34	説明を追加																																	

修正項目	ページ	修正内容（詳細はマニュアル参照）														
11.1 特長 図 11.1 CPG のブロック図	11-2	MCCR を削除														
	11-4	「(9) メモリクロックコントロールレジスタ」を削除														
11.3 クロック動作モード 表 11.3 クロックモードと FRQCR 値の可能な組み合わせ	11-7	表を修正 <table border="1"> <thead> <tr> <th>モード</th> <th>FRQCR レジスタ値</th> <th>入力クロック / 水晶発振子 周波数範囲</th> <th>CKIO 端子 周波数範囲</th> </tr> </thead> <tbody> <tr> <td rowspan="3">7</td> <td>H'1202</td> <td>20MHz ~ 30MHz</td> <td>20MHz ~ 30MHz</td> </tr> <tr> <td>H'1204</td> <td>20MHz ~ 40MHz</td> <td>20MHz ~ 40MHz</td> </tr> <tr> <td>H'1206</td> <td>20MHz ~ 40MHz</td> <td>20MHz ~ 40MHz</td> </tr> </tbody> </table>	モード	FRQCR レジスタ値	入力クロック / 水晶発振子 周波数範囲	CKIO 端子 周波数範囲	7	H'1202	20MHz ~ 30MHz	20MHz ~ 30MHz	H'1204	20MHz ~ 40MHz	20MHz ~ 40MHz	H'1206	20MHz ~ 40MHz	20MHz ~ 40MHz
モード	FRQCR レジスタ値	入力クロック / 水晶発振子 周波数範囲	CKIO 端子 周波数範囲													
7	H'1202	20MHz ~ 30MHz	20MHz ~ 30MHz													
	H'1204	20MHz ~ 40MHz	20MHz ~ 40MHz													
	H'1206	20MHz ~ 40MHz	20MHz ~ 40MHz													
11.4 レジスタの説明	11-8	メモリクロックコントロールレジスタ (MCCR) を削除 注を修正 【注】* ■ STBCR、STBCR2～STBCR4 の説明は、「第 13 章 低消費電力モード」を参照してください。														
11.5.1 通信率の変更	11-10	注を削除														
11.6 ボード設計上の注意事項 図 11.3 PLL 発振回路使用時の注意	11-12	図を修正 														
13.2 レジスタの説明	13-4	メモリクロックコントロールレジスタ (MCCR) を削除														
	-	「13.2.5 メモリクロックコントロールレジスタ (MCCR)」を削除														
15.4.10 SPI モード 図 15.22 SPI データ / クロックのタイミング 1 (CPHA = 0)	15-45	図を修正 														

修正項目	ページ	修正内容（詳細はマニュアル参照）
15.5 使用上の注意事項	15-46	<p>説明を追加</p> <p>5. SIIER の TDMAE ビットを"1"にして、送信データ DMA 転送要求を有効にしたときに、SIFCTR の TFWM ビットを 3'b111（送信 FIFO の空き容量が 1 段以上のときに転送要求を発行する）の設定にした場合、DMA 転送により FIFO が FULL になったにも関わらず、誤って次の DMA 転送要求を出してしまい、送信 FIFO がオーバーフローしてしまう場合があります。このため、送信データ DMA 転送要求使用時には、TFWM ビットは、3'b000～3'b110 までの範囲で設定してください。</p> <p>6. マスタモード 2（ステレオ出力）で使用した場合、SICTR レジスタの FSE ビットに"0"を設定し SIOFSYNC の出力を許可しない状態から、FSE ビットに"1"を設定し SIOFSYNC の出力を許可したとき、先頭フレームの SIOFSYNC 信号の立ち上がりが 1 ビット早くなり High 期間が長くなる場合（送信開始時または送信再開時）と 1 ビット遅くなり High 期間が短くなる場合（送信再開時）があります。また、送信停止時の FSE ビットを"0"とするタイミングが、SIOFSYNC 信号の High 期間の場合には、SIOFSYNC 信号の High 期間が短くなる場合があります。マスタモード 2 を使用する場合には、ステレオデータ転送の開始時と終了時に、無音データを挿入するか外部コーデックのミュート制御をすることをお勧めします。</p>
16.5 割り込み要因と DMAC	16-43	<p>説明を修正</p> <p>1. 発生要因に対応する割り込みイネーブルビット（TIE、RIE）を 1 にセットしてください。</p> <p>2. その他の割り込みイネーブルビット（TSIE、ERIE、BRIE、DRIE）は 0 にセットしてください。</p>
19.1 特長	19-1	<p>説明を修正</p> <ul style="list-style-type: none"> • OpenHCI バージョン 1.0 レジスタセット準拠 • ユニバーサルシリアルバス（Universal Serial Bus）バージョン 2.0 対応 • フルスピード（12Mbps）をサポート <p>注を追加</p> <p>【注】本 LSI では 1 ポートのみの入出力対応のためルートハブ機能はサポートしておりませんが、内部回路構成はルートハブ+1 ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。</p>
19.3.2 HcControl レジスタ（HCTLR）	19-4、19-5	<p>ルートハブに関する注を追加</p> <p>【注】*1 本 LSI では 1 ポートのみの入出力対応のためルートハブ機能はサポートしておりませんが、内部回路構成はルートハブ+1 ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。</p> <p>*2 ED：Endpoint Descriptor</p>

修正項目	ページ	修正内容（詳細はマニュアル参照）									
19.3.3 HcCommandStatus レジスタ (HCSR)	19-7、 19-8	ルートをハブに関する注を追加 【注】*1 TD : Transfer Descriptor *2 本 LSI では 1 ポートのみの入出力対応のためルートをハブ機能はサポートしていませんが、内部回路構成はルートをハブ + 1 ポートとなっております。このため、レジスタ構成等にルートをハブの概念が入っております。									
19.3.19 HcRhDescriptorA レジスタ (HRDRA)	19-22、 19-23	ルートをハブに関する注を追加 【注】* 本 LSI では 1 ポートのみの入出力対応のためルートをハブ機能はサポートしていませんが、内部回路構成はルートをハブ + 1 ポートとなっております。このため、レジスタ構成等にルートをハブの概念が入っております。									
	19-22	説明を修正 <table border="1"> <thead> <tr> <th>ビット</th> <th>ビット名</th> <th>初期値</th> <th>R/W</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>11</td> <td>OCPM</td> <td>0</td> <td>R/W</td> <td>Over Current Protection Mode このビットは、ルートをハブポート*の過電流ステータスの通知方法を選択します。リセット時に、このビットは PSM ビットと同じモードを反映します。NOCP ビットがクリアされた場合、このビットは有効となります。 0 : 過電流ステータスはすべてのダウンストリームポートに対し集合的に報告される（ただし、使用禁止：「19.5.7 過電流検出機能について」参照） 1 : 過電流ステータスはポートごとに通知される</td> </tr> </tbody> </table>	ビット	ビット名	初期値	R/W	説明	11	OCPM	0	R/W
ビット	ビット名	初期値	R/W	説明							
11	OCPM	0	R/W	Over Current Protection Mode このビットは、ルートをハブポート*の過電流ステータスの通知方法を選択します。リセット時に、このビットは PSM ビットと同じモードを反映します。NOCP ビットがクリアされた場合、このビットは有効となります。 0 : 過電流ステータスはすべてのダウンストリームポートに対し集合的に報告される（ただし、使用禁止：「19.5.7 過電流検出機能について」参照） 1 : 過電流ステータスはポートごとに通知される							
19.3.20 HcRhDescriptorB レジスタ (HRDRB)	19-24	ルートをハブに関する注を追加 【注】* 本 LSI では 1 ポートのみの入出力対応のためルートをハブ機能はサポートしていませんが、内部回路構成はルートをハブ + 1 ポートとなっております。このため、レジスタ構成等にルートをハブの概念が入っております。									
19.3.21 HcRhStatus レジスタ (HRSR)	19-25、 19-26	ルートをハブに関する注を追加 【注】* 本 LSI では 1 ポートのみの入出力対応のためルートをハブ機能はサポートしていませんが、内部回路構成はルートをハブ + 1 ポートとなっております。このため、レジスタ構成等にルートをハブの概念が入っております。									
	19-26	説明を修正 <table border="1"> <thead> <tr> <th>ビット</th> <th>ビット名</th> <th>初期値</th> <th>R/W</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>OCI</td> <td>0</td> <td>R/W</td> <td>Over Current Indicator (使用禁止：「19.5.7 過電流検出機能について」参照)</td> </tr> </tbody> </table>	ビット	ビット名	初期値	R/W	説明	1	OCI	0	R/W
ビット	ビット名	初期値	R/W	説明							
1	OCI	0	R/W	Over Current Indicator (使用禁止：「19.5.7 過電流検出機能について」参照)							
19.3.22 HcRhPortStatus レジスタ (HRPSR)	19-27、 19-29 ~ 19-31	ルートをハブに関する注を追加 【注】* 本 LSI では 1 ポートのみの入出力対応のためルートをハブ機能はサポートしていませんが、内部回路構成はルートをハブ + 1 ポートとなっております。このため、レジスタ構成等にルートをハブの概念が入っております。									

修正項目	ページ	修正内容（詳細はマニュアル参照）										
19.5.5 周辺クロックについて	19-33	追加										
19.5.6 Uメモリ、およびXYメモリへのアクセスについて	19-33	追加										
19.5.7 過電流検出機能について	19-34	追加										
19.5.8 USB Bus Resetの発行について	19-34	追加										
20.8.7 ソフトウェアスタンバイモードへの遷移と解除時の注意 図20.21 ソフトウェアスタンバイモードへの遷移と解除のフロー例	20-56	図を修正 ソフトウェアスタンバイモードからの解除フロー例 										
23.2 ポート B 図 23.2 ポート B	23-3	図を修正 										
26.1 特長	26-1	説明を修正 ...に準拠したシリアル入出力インタフェースです。 なお、TCK 入力は外部でプルアップして使用してください。 本 LSI の H-UDI はバウンダリスキャンをサポートし、エミュレータの接続にも使用されます。										
26.2 入出力端子 表 26.1 端子構成	26-2	表を修正 <table border="1" data-bbox="600 1213 1197 1300"> <thead> <tr> <th>端子名</th> <th>入出力</th> <th>説明</th> </tr> </thead> <tbody> <tr> <td>TCK</td> <td>入力</td> <td>シリアルデータ入出力用クロック端子 外部でプルアップしてください。データはこのクロックに同期してデータ入力端子 (TDI) から H-UDI にシリアルに供給され、データ出力端子 (TDO) から出力されます。</td> </tr> </tbody> </table>	端子名	入出力	説明	TCK	入力	シリアルデータ入出力用クロック端子 外部でプルアップしてください。データはこのクロックに同期してデータ入力端子 (TDI) から H-UDI にシリアルに供給され、データ出力端子 (TDO) から出力されます。				
端子名	入出力	説明										
TCK	入力	シリアルデータ入出力用クロック端子 外部でプルアップしてください。データはこのクロックに同期してデータ入力端子 (TDI) から H-UDI にシリアルに供給され、データ出力端子 (TDO) から出力されます。										
26.3.3 バウンダリスキャンレジスタ (SDBSR) 表 26.3 本 LSI の端子とバウンダリスキャンレジスタの対応	26-5	表を修正 <table border="1" data-bbox="642 1348 1156 1499"> <thead> <tr> <th>ビット名</th> <th>端子名</th> <th>入出力</th> </tr> </thead> <tbody> <tr> <td>68</td> <td rowspan="3">USB_OVR_CRNT/USB_VBUS</td> <td>IN</td> </tr> <tr> <td>69</td> <td>Control</td> </tr> <tr> <td>70</td> <td>OUT</td> </tr> </tbody> </table>	ビット名	端子名	入出力	68	USB_OVR_CRNT/USB_VBUS	IN	69	Control	70	OUT
ビット名	端子名	入出力										
68	USB_OVR_CRNT/USB_VBUS	IN										
69		Control										
70		OUT										
	26-7	表を修正 <table border="1" data-bbox="642 1547 1156 1692"> <thead> <tr> <th>ビット名</th> <th>端子名</th> <th>入出力</th> </tr> </thead> <tbody> <tr> <td>213</td> <td rowspan="3">D10</td> <td>IN</td> </tr> <tr> <td>214</td> <td>Control</td> </tr> <tr> <td>215</td> <td>OUT</td> </tr> </tbody> </table>	ビット名	端子名	入出力	213	D10	IN	214	Control	215	OUT
ビット名	端子名	入出力										
213	D10	IN										
214		Control										
215		OUT										

修正項目	ページ	修正内容（詳細はマニュアル参照）																									
26.6 使用上の注意事項	26-14	<p>説明を追加</p> <p>5. ASE モード (ASEMD0 = Low に設定) において、ASEBRKAK* = Low、TRST = High でのパワーオンリセットは有効、ASEBRKAK = High、TRST = High でのパワーオンリセットは無効となりますので、ご注意ください。詳細を表 26.5 に記載します。</p> <p>表 26.5 本 LSI での ASE モードにおけるパワーオンリセット動作</p> <table border="1"> <thead> <tr> <th>各端子の信号状態</th> <th>パワーオンリセット信号アサート時の LSI 動作</th> </tr> </thead> <tbody> <tr> <td>ASEMD0 = Low TRST = High ASEBRKAK = Low</td> <td>パワーオンリセットは有効</td> </tr> <tr> <td>ASEMD0 = Low TRST = High ASEBRKAK = High</td> <td>パワーオンリセットは無効</td> </tr> </tbody> </table> <p>なお、本現象は ASEMD0 = Low を設定する場合（例えば E10A 使用時）に発生しますが、ASEMD0 = High を設定する通常動作の場合は、本現象は発生しません。</p> <p>【注】* E10A がシステム全体の制御を行っているかを示す信号です。E10A がシステム全体の制御を行っているとき、Low を出力します。</p>	各端子の信号状態	パワーオンリセット信号アサート時の LSI 動作	ASEMD0 = Low TRST = High ASEBRKAK = Low	パワーオンリセットは有効	ASEMD0 = Low TRST = High ASEBRKAK = High	パワーオンリセットは無効																			
各端子の信号状態	パワーオンリセット信号アサート時の LSI 動作																										
ASEMD0 = Low TRST = High ASEBRKAK = Low	パワーオンリセットは有効																										
ASEMD0 = Low TRST = High ASEBRKAK = High	パワーオンリセットは無効																										
27.1 レジスタアドレス一覧（機能モジュールごと、マニュアル章番号順）	27-4	メモリクロックコントロールレジスタを削除																									
27.2 レジスタビット一覧	27-14	MCCR を削除																									
27.3 各動作モードにおけるレジスタの状態	27-26	MCCR を削除																									
28.2 DC 特性 表 28.2 DC 特性(2-a)【DAC、USB 関連端子を除く】	28-4	<p>表を修正</p> <table border="1"> <thead> <tr> <th>項 目</th> <th>記号</th> <th>Min.</th> <th>Typ.</th> <th>Max.</th> <th>単位</th> <th>測定条件</th> </tr> </thead> <tbody> <tr> <td rowspan="2">入力 High レベル電圧</td> <td>EXTAL</td> <td>$V_{CC} - 0.3$ 2.0</td> <td>-</td> <td>$V_{CC} + 0.3$ $V_{CC} + 0.3$</td> <td rowspan="2">V</td> <td rowspan="2">HD157102NP の BRCLK 接続時</td> </tr> <tr> <td>EXTAL2</td> <td>$V_{CC} - 0.3$</td> <td>-</td> <td>$V_{CC} + 0.3$</td> </tr> <tr> <td>出力 Low レベル電圧</td> <td>RDI_TXTRDATA, RDI_RXBDW_OUT, RCL_SPI_CLK, RCL_SPI_ENB, RCL_SPI_TXRX, RDI_CTRL3</td> <td>-</td> <td>-</td> <td>0.55</td> <td>V</td> <td>$V_{CC} = 2.8 \sim 3.0V$, $I_{CC} = 2.0mA$</td> </tr> </tbody> </table>	項 目	記号	Min.	Typ.	Max.	単位	測定条件	入力 High レベル電圧	EXTAL	$V_{CC} - 0.3$ 2.0	-	$V_{CC} + 0.3$ $V_{CC} + 0.3$	V	HD157102NP の BRCLK 接続時	EXTAL2	$V_{CC} - 0.3$	-	$V_{CC} + 0.3$	出力 Low レベル電圧	RDI_TXTRDATA, RDI_RXBDW_OUT, RCL_SPI_CLK, RCL_SPI_ENB, RCL_SPI_TXRX, RDI_CTRL3	-	-	0.55	V	$V_{CC} = 2.8 \sim 3.0V$, $I_{CC} = 2.0mA$
項 目	記号	Min.	Typ.	Max.	単位	測定条件																					
入力 High レベル電圧	EXTAL	$V_{CC} - 0.3$ 2.0	-	$V_{CC} + 0.3$ $V_{CC} + 0.3$	V	HD157102NP の BRCLK 接続時																					
	EXTAL2	$V_{CC} - 0.3$	-	$V_{CC} + 0.3$																							
出力 Low レベル電圧	RDI_TXTRDATA, RDI_RXBDW_OUT, RCL_SPI_CLK, RCL_SPI_ENB, RCL_SPI_TXRX, RDI_CTRL3	-	-	0.55	V	$V_{CC} = 2.8 \sim 3.0V$, $I_{CC} = 2.0mA$																					
28.3.1 クロックタイミング 表 28.5 クロックタイミング	28-7	<p>注を修正</p> <p>【注】*4 図 28.8 ~ 図 28.11 に記載されている「RF 起動時間 (T_{xtl})」(HD157100NP および HD157102NP の内蔵発振回路の安定時間)は、HD157102NP のデータシートでいずれも 6msec (Typical) となっています。</p>																									
28.3.6 シンクロナス DRAM タイミング 図 28.42 シンクロナス DRAM パワーオンシーケンス(モードライトタイミング、TRP = 2 サイクル)	28-42	<p>図を修正</p>																									

修正項目	ページ	修正内容（詳細はマニュアル参照）																		
28.3.9 SCIF モジュール信号タイミング 表 28.10 SCIF モジュール信号タイミング	28-54	表を修正 <table border="1" data-bbox="605 343 1195 407"> <thead> <tr> <th data-bbox="605 343 865 363">項目</th> <th data-bbox="865 343 906 363">記号</th> <th data-bbox="906 343 961 363">Min.</th> <th data-bbox="961 343 1044 363">Max.</th> <th data-bbox="1044 343 1112 363">単位</th> <th data-bbox="1112 343 1195 363">参照図</th> </tr> </thead> <tbody> <tr> <td data-bbox="605 363 865 388">SCIF 受信データセットアップ時間（クロック同期）</td> <td data-bbox="865 363 906 388">t_{US}</td> <td data-bbox="906 363 961 388">$2 \times t_{\text{DEC}}$</td> <td data-bbox="961 363 1044 388"></td> <td data-bbox="1044 363 1112 388">ns</td> <td data-bbox="1112 363 1195 388">28.61</td> </tr> <tr> <td data-bbox="605 388 865 407">SCIF 受信データホールド時間（クロック同期）</td> <td data-bbox="865 388 906 407">t_{HD}</td> <td data-bbox="906 388 961 407">$2 \times t_{\text{DEC}}$</td> <td data-bbox="961 388 1044 407"></td> <td data-bbox="1044 388 1112 407">ns</td> <td data-bbox="1112 388 1195 407"></td> </tr> </tbody> </table>	項目	記号	Min.	Max.	単位	参照図	SCIF 受信データセットアップ時間（クロック同期）	t_{US}	$2 \times t_{\text{DEC}}$		ns	28.61	SCIF 受信データホールド時間（クロック同期）	t_{HD}	$2 \times t_{\text{DEC}}$		ns	
項目	記号	Min.	Max.	単位	参照図															
SCIF 受信データセットアップ時間（クロック同期）	t_{US}	$2 \times t_{\text{DEC}}$		ns	28.61															
SCIF 受信データホールド時間（クロック同期）	t_{HD}	$2 \times t_{\text{DEC}}$		ns																

すべての商標および登録商標は、それぞれの所有者に帰属します。

目次

1.	概要	1-1
1.1	SH7660の特長	1-1
1.2	ブロック図	1-7
1.3	ピン配置図	1-8
1.4	端子の機能	1-17
2.	CPU	2-1
2.1	処理状態と処理モード	2-1
2.1.1	処理状態	2-1
2.1.2	処理モード（ユーザモード / 特権モード）	2-2
2.2	メモリマップ	2-3
2.2.1	論理アドレス空間	2-3
2.2.2	物理アドレス空間	2-5
2.2.3	外部アドレス空間	2-6
2.3	レジスタの説明	2-7
2.3.1	汎用レジスタ	2-9
2.3.2	システムレジスタ	2-10
2.3.3	プログラムカウンタ	2-10
2.3.4	コントロールレジスタ	2-11
2.4	データ形式	2-13
2.4.1	レジスタのデータ形式	2-13
2.4.2	メモリ上でのデータ形式	2-14
2.5	命令の特長	2-15
2.5.1	命令の実行方法	2-15
2.5.2	アドレッシングモード	2-17
2.5.3	命令形式	2-20
2.6	命令セット	2-22
2.6.1	機能別命令セット	2-22
2.6.2	オペレーションコードマップ	2-32
3.	DSP 演算ユニット	3-1
3.1	DSP拡張機能	3-1
3.2	DSPモードのリソース	3-3
3.2.1	処理モード	3-3

3.2.2	DSP モードのメモリマップ	3-3
3.2.3	CPU のレジスタセット	3-3
3.2.4	DSP レジスタ	3-6
3.3	CPU 拡張命令	3-7
3.3.1	リピート制御命令	3-7
3.3.2	拡張リピート制御命令	3-15
3.4	DSP データ転送命令	3-18
3.4.1	汎用レジスタ	3-21
3.4.2	DSP データアドレッシング	3-23
3.4.3	モジュロアドレッシング	3-24
3.4.4	メモリのデータ形式	3-26
3.4.5	ダブル、シングルデータ転送命令の命令フォーマット	3-26
3.5	DSP データ演算命令	3-28
3.5.1	DSP レジスタ	3-28
3.5.2	DSP 命令の命令セット	3-31
3.5.3	DSP タイプデータ形式	3-35
3.5.4	ALU 固定小数点算術演算	3-36
3.5.5	ALU 整数演算	3-41
3.5.6	ALU 論理演算	3-42
3.5.7	固定小数点乗算	3-44
3.5.8	シフト演算	3-45
3.5.9	MSB 検出命令	3-49
3.5.10	丸め演算	3-52
3.5.11	オーバフロー防止機能	3-53
3.5.12	ローカルデータ移動命令	3-54
3.5.13	オペランドの競合	3-55
3.6	DSP 拡張機能命令セット	3-56
3.6.1	CPU 拡張命令セット	3-56
3.6.2	ダブルデータ転送命令セット	3-57
3.6.3	シングルデータ転送命令セット	3-58
3.6.4	DSP データ演算命令セット	3-60
3.6.5	DSP モードでのオペレーションコードマップ	3-64
4.	例外処理	4-1
4.1	レジスタの説明	4-1
4.1.1	TRAPA 例外レジスタ (TRA)	4-2
4.1.2	例外事象レジスタ (EXPEVT)	4-2
4.1.3	割り込み事象レジスタ 2 (INTEVT2)	4-3
4.1.4	例外アドレスレジスタ (TEA)	4-3
4.2	例外処理の機能	4-3

4.2.1	例外処理の流れ	4-3
4.2.2	例外処理ベクタアドレス	4-4
4.2.3	例外コード	4-4
4.2.4	例外要求と BL ビットの関係 (多重例外防止)	4-4
4.2.5	例外要因の受け付けタイミングと優先順位	4-5
4.3	個別例外の動作説明	4-7
4.3.1	リセット	4-7
4.3.2	一般例外	4-8
4.4	DSP 拡張機能有効時の例外処理	4-11
4.4.1	不当命令例外、スロット不当命令例外	4-11
4.4.2	CPU アドレスエラー	4-11
4.4.3	リピート制御中の例外	4-11
4.5	使用上の注意事項	4-16
5.	キャッシュ	5-1
5.1	特長	5-1
5.1.1	キャッシュの構成	5-1
5.2	レジスタの説明	5-3
5.2.1	キャッシュ制御レジスタ 1 (CCR1)	5-3
5.2.2	キャッシュ制御レジスタ 2 (CCR2)	5-4
5.3	動作説明	5-6
5.3.1	キャッシュの検索	5-6
5.3.2	リード動作	5-7
5.3.3	プリフェッチ動作	5-8
5.3.4	ライト動作	5-8
5.3.5	ライトバックバッファ	5-8
5.3.6	キャッシュと外部メモリとのコヒーレンシ	5-9
5.4	メモリ割り付けキャッシュの構成	5-9
5.4.1	アドレスアレイ	5-9
5.4.2	データアレイ	5-10
5.4.3	使用例	5-12
6.	X/Y メモリ	6-1
6.1	特長	6-1
6.2	動作説明	6-2
6.2.1	CPU からのアクセス	6-2
6.2.2	DSP からのアクセス	6-2
6.2.3	I バスマスタモジュールからのアクセス	6-3
6.3	使用上の注意事項	6-3
6.3.1	ページ競合	6-3

6.3.2	バス競合	6-3
6.3.3	キャッシュの設定	6-3
6.3.4	アドレスエラー	6-4
6.3.5	リセット解除後のメモリの内容	6-4
7.	Uメモリ	7-1
7.1	特長	7-1
7.2	動作説明	7-2
7.2.1	CPU からのアクセス	7-2
7.2.2	DSP からのアクセス	7-2
7.2.3	Iバスマスタモジュールからのアクセス	7-2
7.3	使用上の注意事項	7-3
7.3.1	ページ競合	7-3
7.3.2	バス競合	7-3
7.3.3	キャッシュの設定	7-3
7.3.4	アドレスエラー	7-4
7.3.5	リセット解除後のメモリの内容	7-4
8.	割り込みコントローラ (INTC)	8-1
8.1	特長	8-1
8.2	入出力端子	8-3
8.3	レジスタの説明	8-3
8.3.1	割り込み優先レベル設定レジスタ A ~ H (IPRA ~ IPRH)	8-4
8.3.2	割り込みコントロールレジスタ 0 (ICR0)	8-6
8.3.3	割り込みコントロールレジスタ 1 (ICR1)	8-7
8.3.4	割り込みコントロールレジスタ 2 (ICR2)	8-8
8.3.5	割り込み要求レジスタ 0 (IRR0)	8-9
8.3.6	割り込みマスクレジスタ 0、1、4~6、9 (IMR0、1、4~6、9)	8-9
8.3.7	割り込みマスククリアレジスタ 0、1、4~6、9 (IMCR0、1、4~6、9)	8-10
8.4	割り込み要因	8-11
8.4.1	NMI 割り込み	8-11
8.4.2	IRQ 割り込み	8-11
8.4.3	内蔵周辺モジュール割り込み	8-12
8.4.4	割り込み例外処理および優先順位	8-12
8.5	動作説明	8-14
8.5.1	割り込み動作の流れ	8-14
8.5.2	多重割り込み	8-16
9.	バスステートコントローラ (BSC)	9-1
9.1	特長	9-1

9.2	入出力端子	9-4
9.3	エリアの概要	9-5
9.3.1	空間分割	9-5
9.3.2	シャドウ空間	9-5
9.3.3	アドレスマップ	9-6
9.3.4	データバス幅	9-7
9.4	レジスタの説明	9-7
9.4.1	共通コントロールレジスタ (CMNCR)	9-8
9.4.2	CS _n 空間バスコントロールレジスタ (CS _n BCR) (n=0、3、4)	9-9
9.4.3	CS _n 空間ウェイトコントロールレジスタ (CS _n WCR) (n=0、3、4)	9-12
9.4.4	SDRAM コントロールレジスタ (SDCR)	9-21
9.4.5	リフレッシュタイムコントロール/ステータスレジスタ (RTC _{SR})	9-24
9.4.6	リフレッシュタイムカウンタ (RTCNT)	9-25
9.4.7	リフレッシュタイムコンスタントレジスタ (RTCOR)	9-25
9.4.8	リセットウェイトカウンタ (RWCNT)	9-26
9.5	動作説明	9-26
9.5.1	エンディアン/アクセスサイズとデータアライメント	9-26
9.5.2	通常空間インタフェース	9-29
9.5.3	アクセスウェイト制御	9-34
9.5.4	\overline{CS}_n アサート期間拡張	9-36
9.5.5	SDRAM インタフェース	9-37
9.5.6	バースト ROM インタフェース	9-59
9.5.7	バイト選択付き SRAM インタフェース	9-60
9.5.8	アクセスサイクル間ウェイト	9-64
9.5.9	バスアービトレーション	9-65
9.5.10	使用上の注意事項	9-68
10.	ダイレクトメモリアクセスコントローラ (DMAC)	10-1
10.1	特長	10-1
10.2	入出力端子	10-3
10.3	レジスタの説明	10-3
10.3.1	DMA ソースアドレスレジスタ (SAR)	10-4
10.3.2	DMA ディスティネーションアドレスレジスタ (DAR)	10-4
10.3.3	DMA トランスファカウンタレジスタ (DMATCR)	10-4
10.3.4	DMA チャンネルコントロールレジスタ (CHCR)	10-5
10.3.5	DMA イニシャルアドレスレジスタ (IAR)	10-10
10.3.6	DMA オペレーションレジスタ (DMAOR)	10-11
10.3.7	DMA 拡張リソースセクタ 0、1 (DMARS0、DMARS1)	10-13
10.4	動作説明	10-15
10.4.1	転送フロー	10-15

10.4.2	リピートモード転送	10-17
10.4.3	DMA 転送要求	10-17
10.4.4	チャンネルの優先順位	10-20
10.4.5	DMA 転送の種類	10-23
10.4.6	バスサイクルのステート数と DREQ 端子のサンプリングタイミング	10-30
10.5	使用上の注意事項	10-34
11.	クロックパルス発振器 (CPG)	11-1
11.1	特長	11-1
11.2	入出力端子	11-4
11.3	クロック動作モード	11-5
11.4	レジスタの説明	11-8
11.4.1	周波数制御レジスタ (FRQCR)	11-8
11.5	周波数変更方法	11-10
11.5.1	逡倍率の変更	11-10
11.5.2	分周率の変更	11-10
11.6	ボード設計上の注意事項	11-11
11.7	使用上の注意事項	11-12
11.7.1	$\overline{\text{RESETP}}$ 端子を用いたリセットについて	11-12
12.	ウォッチドッグタイマ (WDT)	12-1
12.1	特長	12-1
12.2	レジスタの説明	12-3
12.2.1	ウォッチドッグタイマカウンタ (WTCNT)	12-3
12.2.2	ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR)	12-3
12.2.3	レジスタアクセス時の注意	12-5
12.3	動作説明	12-6
12.3.1	ソフトウェアスタンバイ解除の手順	12-6
12.3.2	周波数変更の手順	12-6
12.3.3	ウォッチドッグタイマモードの使用法	12-7
12.3.4	インターバルタイマモードの使用法	12-7
13.	低消費電力モード	13-1
13.1	特長	13-1
13.1.1	低消費電力モードの種類	13-1
13.1.2	リセット	13-2
13.1.3	入出力端子	13-3
13.2	レジスタの説明	13-4
13.2.1	スタンバイコントロールレジスタ (STBCR)	13-4
13.2.2	スタンバイコントロールレジスタ 2 (STBCR2)	13-5

13.2.3	スタンバイコントロールレジスタ 3 (STBCR3)	13-6
13.2.4	スタンバイコントロールレジスタ 4 (STBCR4)	13-6
13.3	動作説明	13-8
13.3.1	スリープモード	13-8
13.3.2	ソフトウェアスタンバイモード	13-8
13.3.3	モジュールスタンバイ機能	13-10
13.3.4	各種モード間の状態遷移	13-11
13.3.5	内蔵レギュレータをオフ (外部 VDD 印加) する方法	13-11
14.	タイマユニット (TMU)	14-1
14.1	特長	14-1
14.2	レジスタの説明	14-3
14.2.1	タイマスタートレジスタ (TSTR)	14-4
14.2.2	タイマコントロールレジスタ (TCR)	14-5
14.2.3	タイマコンスタントレジスタ (TCOR)	14-6
14.2.4	タイマカウンタ (TCNT)	14-6
14.3	動作説明	14-6
14.3.1	カウンタの動作	14-6
14.4	割り込み	14-8
14.4.1	ステータスフラグのセットタイミング	14-8
14.4.2	ステータスフラグのクリアタイミング	14-9
14.4.3	割り込み要因と優先順位	14-9
14.5	使用上の注意事項	14-10
15.	FIFO 付きシリアルインタフェース (SIOF)	15-1
15.1	特長	15-1
15.2	入出力端子	15-2
15.3	レジスタの説明	15-3
15.3.1	モードレジスタ (SIMDR)	15-4
15.3.2	コントロールレジスタ (SICTR)	15-6
15.3.3	送信データレジスタ (SITDR)	15-8
15.3.4	受信データレジスタ (SIRDR)	15-8
15.3.5	送信制御データレジスタ (SITCR)	15-9
15.3.6	受信制御データレジスタ (SIRCR)	15-10
15.3.7	ステータスレジスタ (SISTR)	15-10
15.3.8	割り込み許可レジスタ (SIHER)	15-15
15.3.9	FIFO コントロールレジスタ (SIFCTR)	15-16
15.3.10	クロックセレクトレジスタ (SISCR)	15-18
15.3.11	送信データアサインレジスタ (SITDAR)	15-19
15.3.12	受信データアサインレジスタ (SIRDAR)	15-20

15.3.13	制御データアサインレジスタ (SICDAR)	15-21
15.3.14	SPI 制御レジスタ (SPICR)	15-22
15.4	動作説明	15-24
15.4.1	シリアルクロック	15-24
15.4.2	シリアルタイミング	15-25
15.4.3	転送データフォーマット	15-26
15.4.4	転送データのレジスタ割り付け	15-27
15.4.5	制御データインタフェース	15-30
15.4.6	FIFO	15-31
15.4.7	送受信手順	15-33
15.4.8	割り込み	15-38
15.4.9	送受信タイミング	15-39
15.4.10	SPI モード	15-43
15.5	使用上の注意事項	15-46
16.	FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)	16-1
16.1	特長	16-1
16.2	入出力端子	16-3
16.3	レジスタの説明	16-4
16.3.1	レシーブシフトレジスタ (SCRSR)	16-5
16.3.2	レシーブ FIFO データレジスタ (SCFRDR)	16-5
16.3.3	トランスミットシフトレジスタ (SCTSR)	16-5
16.3.4	トランスミット FIFO データレジスタ (SCFTDR)	16-6
16.3.5	シリアルモードレジスタ (SCSMR)	16-6
16.3.6	シリアルコントロールレジスタ (SCSCR)	16-8
16.3.7	FIFO エラー数レジスタ (SCFER)	16-12
16.3.8	シリアルステータスレジスタ (SCSSR)	16-13
16.3.9	ビットレートレジスタ (SCBRR)	16-17
16.3.10	FIFO コントロールレジスタ (SCFCR)	16-19
16.3.11	FIFO データ数レジスタ (SCFDR)	16-21
16.3.12	トランスミットデータストップレジスタ (SCTDSR)	16-22
16.4	動作説明	16-22
16.4.1	概要	16-22
16.4.2	調歩同期式モード	16-22
16.4.3	調歩同期式モードのシリアル動作	16-23
16.4.4	クロック同期式モード	16-32
16.4.5	クロック同期式モードのシリアル動作	16-33
16.5	割り込み要因とDMAC	16-43
16.6	使用上の注意事項	16-44

17. ブート機能 (BOOT)	17-1
17.1 特長	17-1
17.2 入出力端子	17-2
17.3 レジスタの説明	17-3
17.4 動作説明	17-3
17.4.1 ブートモードのアドレス空間	17-3
17.4.2 ブート処理の実行手順	17-4
17.5 使用上の注意事項	17-7
18. USB ビンマルチプレクスコントローラ (USBPM)	18-1
18.1 特長	18-1
18.2 入出力端子	18-2
18.3 レジスタの説明	18-3
18.3.1 EXCPG 制御レジスタ (EXCPGCR)	18-3
18.4 外部回路例	18-5
18.4.1 USB ファンクションと外部回路の接続例	18-5
18.4.2 USB ホストと外部回路の接続例	18-7
18.5 使用上の注意事項	18-8
19. USB ホストコントローラ (USBH)	19-1
19.1 特長	19-1
19.2 入出力端子	19-1
19.3 レジスタの説明	19-2
19.3.1 HcRevision レジスタ (HREVR)	19-3
19.3.2 HcControl レジスタ (HCTLR)	19-3
19.3.3 HcCommandStatus レジスタ (HCSR)	19-6
19.3.4 HcInterruptStatus レジスタ (HISR)	19-8
19.3.5 HcInterruptEnable レジスタ (HIER)	19-10
19.3.6 HcInterruptDisable レジスタ (HIDR)	19-12
19.3.7 HcHCCA レジスタ (HHCCAR)	19-14
19.3.8 HcPeriodCurrentED レジスタ (HPCEDR)	19-14
19.3.9 HcControlHeadED レジスタ (HCHEDR)	19-15
19.3.10 HcControlCurrentED レジスタ (HCCEDR)	19-15
19.3.11 HcBulkHeadED レジスタ (HBHEDR)	19-15
19.3.12 HcBulkCurrentED レジスタ (HBCEDR)	19-15
19.3.13 HcDoneHeadED レジスタ (HDHEDR)	19-16
19.3.14 HcFmInterval レジスタ (HFIR)	19-16
19.3.15 HcFmRemaining レジスタ (HFRR)	19-18
19.3.16 HcFmNumber レジスタ (HFNR)	19-19
19.3.17 HcPeriodicStart レジスタ (HPSR)	19-20

19.3.18	HcLSThreshold レジスタ (HLSTR)	19-21
19.3.19	HcRhDescriptorA レジスタ (HRDRA)	19-22
19.3.20	HcRhDescriptorB レジスタ (HRDRB)	19-24
19.3.21	HcRhStatus レジスタ (HRSR)	19-25
19.3.22	HcRhPortStatus レジスタ (HRPSR)	19-26
19.4	データ格納フォーマット	19-31
19.4.1	転送データの格納フォーマット	19-31
19.4.2	ディスクリプタの格納フォーマット	19-31
19.5	使用上の注意事項	19-31
19.5.1	リセット制御の制約	19-31
19.5.2	ロースピードに関して	19-32
19.5.3	共有メモリへのアクセス動作に関する制約 1	19-32
19.5.4	共有メモリへのアクセス動作に関する制約 2	19-33
19.5.5	周辺クロックについて	19-33
19.5.6	U メモリ、および X/Y メモリへのアクセスについて	19-34
19.5.7	過電流検出機能について	19-34
19.5.8	USB Bus Reset の発行について	19-34
20.	USB ファンクションコントローラ (USBF)	20-1
20.1	特長	20-1
20.2	入出力端子	20-3
20.3	レジスタの説明	20-4
20.3.1	割り込みフラグレジスタ 0 (IFR0)	20-5
20.3.2	割り込み選択レジスタ 0 (ISR0)	20-11
20.3.3	割り込みイネーブルレジスタ 0 (IER0)	20-14
20.3.4	EP0i データレジスタ (EPDR0i)	20-16
20.3.5	EP0o データレジスタ (EPDR0o)	20-17
20.3.6	EP0s データレジスタ (EPDR0s)	20-17
20.3.7	EP1 データレジスタ (EPDR1)	20-17
20.3.8	EP2i データレジスタ (EPDR2i)	20-18
20.3.9	EP2o データレジスタ (EPDR2o)	20-18
20.3.10	EP3i データレジスタ (EPDR3i)	20-18
20.3.11	EP3o データレジスタ (EPDR3o)	20-19
20.3.12	EP4 データレジスタ (EPDR4)	20-19
20.3.13	EP5 データレジスタ (EPDR5)	20-19
20.3.14	EP6 データレジスタ (EPDR6)	20-19
20.3.15	EP0o 受信データサイズレジスタ (EPSZ0o)	20-20
20.3.16	EP2o 受信データサイズレジスタ (EPSZ2o)	20-20
20.3.17	EP3o 受信データサイズレジスタ (EPSZ3o)	20-20
20.3.18	EP6 受信データサイズレジスタ (EPSZ6)	20-20

20.3.19	トリガレジスタ (TRG)	20-21
20.3.20	データステータスレジスタ (DASTS)	20-22
20.3.21	FIFO クリアレジスタ (FCLR)	20-23
20.3.22	DMA 転送設定レジスタ (DMA)	20-24
20.3.23	エンドポイントストールレジスタ (EPSTL)	20-25
20.3.24	コンフィグレーションバリュージスタ (CVR)	20-26
20.3.25	タイムスタンプレジスタ (TSR)	20-26
20.3.26	コントロールレジスタ (CTRL)	20-27
20.3.27	エンドポイント情報レジスタ (EPIRn0 ~ 5)	20-28
20.4	動作説明	20-35
20.4.1	ケーブル接続時	20-35
20.4.2	ケーブル切断時	20-36
20.4.3	コントロール転送	20-36
20.4.4	EP1、4 インタラプトイン転送	20-42
20.4.5	EP2i、5 バルクイン転送 (2 面 FIFO)	20-43
20.4.6	EP2o、6 バルクアウト転送 (2 面 FIFO)	20-44
20.4.7	EP3i アイソクロナスイン転送	20-45
20.4.8	EP3o アイソクロナスアウト転送	20-47
20.5	USB標準コマンドとクラス / ベンダーコマンドの処理	20-49
20.5.1	コントロール転送で送信されるコマンドの処理	20-49
20.6	ストール動作	20-50
20.6.1	概要	20-50
20.6.2	アプリケーションが強制的にストールさせたい場合	20-50
20.6.3	USB ファンクションコントローラが自動的にストールさせる場合	20-52
20.7	USBファンクションコントローラと外部回路の接続例	20-53
20.8	使用上の注意事項	20-54
20.8.1	セットアップデータ受信について	20-54
20.8.2	FIFO のクリアについて	20-54
20.8.3	データレジスタの読み出し / 書き込みについて	20-54
20.8.4	EP0 に関する割り込み要因の割り当てについて	20-54
20.8.5	DMA 転送設定時の FIFO クリアについて	20-55
20.8.6	TR 割り込み使用時の注意	20-55
20.8.7	ソフトウェアスタンバイモードへの遷移と解除時の注意	20-55
20.8.8	バスパワー機能を使用する場合のメインクロックについて	20-57
20.8.9	周辺クロックについて	20-57
21.	Bluetooth™ インタフェース (BT)	21-1
21.1	特長	21-1
21.2	入出力端子	21-3
21.3	レジスタの説明	21-3

21.4	RF ICとの接続	21-4
21.4.1	RF IC との接続	21-4
21.5	Voice Codec ICとの接続.....	21-5
21.5.1	Voice Codec (STLC7550) との接続	21-5
21.5.2	Voice Codec (MC145483) との接続	21-6
21.6	低周波数クロック発振器の接続.....	21-7
21.6.1	RTCSEL1 ビットによる選択機能	21-7
21.6.2	周波数変換回路の動作説明.....	21-8
21.6.3	外部水晶振動子接続時の注意	21-9
21.7	パワーオンリセット/クロックリジューム制御機能.....	21-9
21.7.1	パワーオンリセット	21-9
21.7.2	クロックリジューム制御.....	21-10
21.8	Bluetooth HCI/TCIコマンドとAPI.....	21-10
21.9	基板設計の制約事項	21-11
22.	D/A コンバータ (DAC)	22-1
22.1	特長.....	22-1
22.2	入出力端子	22-2
22.3	レジスタの説明	22-2
22.3.1	D/A データレジスタ 0、1 (DADR_0、1)	22-2
22.3.2	D/A コントロールレジスタ (DACR)	22-3
22.4	動作説明.....	22-4
23.	I/O ポート.....	23-1
23.1	ポートA.....	23-1
23.1.1	レジスタの説明	23-1
23.1.2	ポート A データレジスタ (PADR)	23-2
23.2	ポートB	23-3
23.2.1	レジスタの説明	23-3
23.2.2	ポート B データレジスタ (PBDR)	23-3
23.3	ポートD.....	23-5
23.3.1	レジスタの説明	23-5
23.3.2	ポート D データレジスタ (PDDR)	23-5
23.4	ポートE.....	23-7
23.4.1	レジスタの説明	23-7
23.4.2	ポート E データレジスタ (PEDR)	23-7
23.5	ポートG.....	23-9
23.5.1	レジスタの説明	23-9
23.5.2	ポート G データレジスタ (PGDR)	23-9
23.6	ポートH.....	23-11

23.6.1	レジスタの説明	23-11
23.6.2	ポート H データレジスタ (PHDR)	23-11
24.	ピンファンクションコントローラ (PFC)	24-1
24.1	概要	24-1
24.2	レジスタの説明	24-3
24.2.1	ポート A コントロールレジスタ (PACR)	24-3
24.2.2	ポート B コントロールレジスタ (PBCR)	24-4
24.2.3	ポート D コントロールレジスタ (PDCR)	24-6
24.2.4	ポート E コントロールレジスタ (PECR)	24-7
24.2.5	ポート G コントロールレジスタ (PGCR)	24-8
24.2.6	ポート H コントロールレジスタ (PHCR)	24-9
24.2.7	ピンセレクトレジスタ A (PSELA)	24-10
24.2.8	I/O バッファ Hi-Z 制御レジスタ A (HIZCRA)	24-11
24.2.9	ノイズキャンセラ制御レジスタ (NCCR)	24-13
25.	ユーザブレイクコントローラ (UBC)	25-1
25.1	特長	25-1
25.2	レジスタの説明	25-3
25.2.1	ブレイクアドレスレジスタ A (BARA)	25-3
25.2.2	ブレイクアドレスマスクレジスタ A (BAMRA)	25-4
25.2.3	ブレイクバスサイクルレジスタ A (BBRA)	25-4
25.2.4	ブレイクアドレスレジスタ B (BARB)	25-5
25.2.5	ブレイクアドレスマスクレジスタ B (BAMRB)	25-6
25.2.6	ブレイクデータレジスタ B (BDRB)	25-6
25.2.7	ブレイクデータマスクレジスタ B (BDMRB)	25-7
25.2.8	ブレイクバスサイクルレジスタ B (BBRB)	25-8
25.2.9	ブレイクコントロールレジスタ (BRCR)	25-9
25.2.10	実行回数ブレイクレジスタ (BETR)	25-11
25.2.11	ブランチソースレジスタ (BRSR)	25-13
25.2.12	ブランチデスティネーションレジスタ (BRDR)	25-14
25.3	動作説明	25-15
25.3.1	ユーザブレイク動作の流れ	25-15
25.3.2	命令フェッチサイクルでのブレイク	25-16
25.3.3	データアクセスサイクルでのブレイク	25-17
25.3.4	X メモリまたは Y メモリバスサイクルでのブレイク	25-18
25.3.5	シーケンシャルブレイク	25-18
25.3.6	退避されるプログラムカウンタの値	25-19
25.3.7	PC トレース	25-20
25.3.8	使用例	25-20

25.4	使用上の注意事項	25-25
26.	ユーザデバッグインタフェース (H-UDI)	26-1
26.1	特長	26-1
26.2	入出力端子	26-2
26.3	レジスタの説明	26-3
26.3.1	バイパスレジスタ (SDBPR)	26-3
26.3.2	インストラクションレジスタ (SDIR)	26-3
26.3.3	バウンダリスキャンレジスタ (SDBSR)	26-4
26.3.4	ID レジスタ (SDID)	26-8
26.4	動作説明	26-9
26.4.1	TAP コントローラ	26-9
26.4.2	リセット構成	26-10
26.4.3	TDO 出力タイミング	26-10
26.5	バウンダリスキャン	26-11
26.5.1	サポートする命令	26-11
26.5.2	注意事項	26-12
26.6	使用上の注意事項	26-13
27.	レジスタ一覧	27-1
27.1	レジスタアドレス一覧 (機能モジュールごと、マニュアル章番号順)	27-2
27.2	レジスタビット一覧	27-8
27.3	各動作モードにおけるレジスタの状態	27-24
28.	電気的特性	28-1
28.1	絶対最大定格	28-1
28.2	DC特性	28-3
28.3	AC特性	28-6
28.3.1	クロックタイミング	28-6
28.3.2	制御信号タイミング	28-13
28.3.3	AC バスタイミング仕様	28-16
28.3.4	基本タイミング	28-18
28.3.5	パースト ROM タイミング	28-25
28.3.6	シンクロナス DRAM タイミング	28-26
28.3.7	周辺モジュール信号タイミング	28-49
28.3.8	SIOF モジュール信号タイミング	28-50
28.3.9	SCIF モジュール信号タイミング	28-54
28.3.10	USB モジュール信号タイミング	28-55
28.3.11	USB トランシーバタイミング	28-56
28.3.12	ブルートゥースインタフェース (BT) タイミング	28-57

28.3.13	AC 特性測定条件.....	28-60
28.4	D/Aコンバータ特性.....	28-61
付録	付録-1
A.	端子状態表.....	付録-1
B.	外部水晶発振子使用上の注意.....	付録-5
B.1	推奨回路図.....	付録-5
B.2	基板設計時のご注意.....	付録-6
C.	外形寸法図.....	付録-7
索引	索引-1

図目次

1. 概要	1-1
図 1.1 SH7660 の構成	1-7
図 1.2 ピン配置図	1-8
2. CPU	2-1
図 2.1 処理の状態遷移図	2-2
図 2.2 論理アドレス空間	2-3
図 2.3 P4 領域	2-4
図 2.4 物理アドレス空間	2-5
図 2.5 外部アドレス空間と実装空間（エリア 0 の場合）	2-6
図 2.6 処理モード別のレジスタ構成	2-8
図 2.7 汎用レジスタの構成	2-10
図 2.8 システムレジスタおよびプログラムカウンタの構成	2-11
図 2.9 コントロールレジスタの構成	2-13
図 2.10 メモリ上のデータ形式（ビッグエンディアン）	2-14
図 2.11 メモリ上のデータ形式（リトルエンディアン）	2-15
3. DSP 演算ユニット	3-1
図 3.1 DSP 命令の命令形式	3-2
図 3.2 DSP モードでの CPU レジスタ	3-4
図 3.3 DSP レジスタの構成	3-7
図 3.4 DSP レジスタとバスの接続	3-19
図 3.5 汎用レジスタ（DSP モード）	3-22
図 3.6 DSP 命令による並列処理プログラムの例	3-33
図 3.7 条件付き演算とデータ転送命令の例	3-34
図 3.8 データ形式	3-36
図 3.9 ALU 固定小数点算術演算フロー	3-37
図 3.10 演算シーケンスの例	3-38
図 3.11 キャリー / ボローモードでの DC ビット生成の例	3-39
図 3.12 負値モードでの DC ビット生成の例	3-39
図 3.13 オーバフローモードでの DC ビット生成の例	3-40
図 3.14 ALU 整数演算フロー	3-41
図 3.15 ALU 論理演算フロー	3-43
図 3.16 固定小数点乗算フロー	3-44

図 3.17	算術シフト演算フロー.....	3-46
図 3.18	論理シフト演算フロー.....	3-48
図 3.19	PDMSB 演算フロー.....	3-50
図 3.20	丸め演算フロー.....	3-52
図 3.21	丸め演算の定義.....	3-53
図 3.22	ローカルデータ移動命令のフロー.....	3-54
4.	例外処理.....	4-1
図 4.1	レジスタのビット構成.....	4-2
5.	キャッシュ.....	5-1
図 5.1	キャッシュの構成.....	5-2
図 5.2	キャッシュの検索方法.....	5-7
図 5.3	ライトバックバッファの構成.....	5-9
図 5.4	メモリ割り付けキャッシュアクセスのアドレス、データ指定方法.....	5-11
8.	割り込みコントローラ (INTC).....	8-1
図 8.1	INTC のブロック図.....	8-2
図 8.2	割り込み動作フローチャート.....	8-15
9.	バスステートコントローラ (BSC).....	9-1
図 9.1	BSC のブロック図.....	9-3
図 9.2	論理アドレス空間と物理アドレス空間.....	9-5
図 9.3	通常空間基本アクセス (アクセスウェイト 0).....	9-29
図 9.4	通常空間連続アクセス例 1 データバス幅 16 ビット、ロングワードアクセス、CSnWCR.WM ビット = 0 (アクセスウェイト 0、サイクル間ウェイト 0).....	9-31
図 9.5	通常空間連続アクセス例 2 データバス幅 16 ビット、ロングワードアクセス、CSnWCR.WM ビット = 1 (アクセスウェイト 0、サイクル間ウェイト 0).....	9-32
図 9.6	16 ビットデータ幅 SRAM 接続例.....	9-33
図 9.7	8 ビットデータ幅 SRAM 接続例.....	9-33
図 9.8	通常空間アクセスのウェイトタイミング (ソフトウェアウェイトのみ).....	9-34
図 9.9	通常空間アクセスのウェイトタイミング (WAIT 信号によるウェイト挿入).....	9-35
図 9.10	CSn アサート期間拡張.....	9-36
図 9.11	16 ビットデータ幅 SDRAM 接続例.....	9-37
図 9.12	パーストリード基本タイミング (オートプリチャージ).....	9-42
図 9.13	パーストリードウェイト指定タイミング (オートプリチャージ).....	9-43
図 9.14	シングルリードの基本タイミング (オートプリチャージ).....	9-44
図 9.15	パーストライト基本タイミング (オートプリチャージ).....	9-45
図 9.16	シングルライト基本タイミング (オートプリチャージ).....	9-46
図 9.17	パーストリードタイミング (オートプリチャージなし).....	9-48

図 9.18	バーストリードタイミング (バンクアクティブ、同一ロウアドレス)	9-48
図 9.19	バーストリードタイミング (バンクアクティブ、異なるロウアドレス)	9-49
図 9.20	シングルライトタイミング (オートプリチャージなし)	9-50
図 9.21	シングルライトタイミング (バンクアクティブ、同一ロウアドレス)	9-51
図 9.22	シングルライトタイミング (バンクアクティブ、異なるロウアドレス)	9-52
図 9.23	オートリフレッシュタイミング	9-53
図 9.24	セルフリフレッシュタイミング	9-54
図 9.25	低周波数モードでのアクセスタイミング	9-56
図 9.26	パワーダウンモードでのアクセスタイミング	9-57
図 9.27	SDRAM モードレジスタ書き込みタイミング (JEDEC 準拠)	9-59
図 9.28	バースト ROM アクセス (データバス幅 8 ビット 4 バイト転送 (バースト数 4)、 初回アクセスウェイト 2、2 回目以降アクセスウェイト 1)	9-60
図 9.29	BAS=0 バイト選択付き SRAM 基本アクセスタイミング	9-61
図 9.30	BAS=1 バイト選択付き SRAM 基本アクセスタイミング	9-62
図 9.31	16 ビットデータ幅バイト選択付き SRAM 接続例	9-63
図 9.32	不具合発生時波形	9-64
図 9.33	バスアービトレーション (マスタモード)	9-67
図 9.34	マスタとスレーブの接続例	9-68
10.	ダイレクトメモリアクセスコントローラ (DMAC)	10-1
図 10.1	DMAC ブロック図	10-2
図 10.2	DMA 転送フローチャート	10-16
図 10.3	ラウンドロビンモード	10-21
図 10.4	ラウンドロビンモードでのチャネル優先順位	10-22
図 10.5	デュアルアドレスモードのデータフロー	10-24
図 10.6	デュアルアドレスモードの DMA 転送タイミング例 (転送元: 通常メモリ、転送先: 通常メモリ)	10-25
図 10.7	シングルアドレスモードのデータフロー	10-26
図 10.8	シングルアドレスモードの DMA 転送タイミング例	10-26
図 10.9	サイクルスチール通常モードの DMA 転送例 (デュアルアドレス、DREQ ローレベル検出)	10-27
図 10.10	サイクルスチールインターミットモードの DMA 転送例 (デュアルアドレス、DREQ ローレベル検出)	10-28
図 10.11	バーストモードの DMA 転送例 (デュアルアドレス、DREQ ローレベル検出)	10-28
図 10.12	複数チャネルが動作する場合のバス状態	10-30
図 10.13	サイクルスチールモード・エッジ検出時の DREQ 入力検出タイミング	10-31
図 10.14	サイクルスチールモード・レベル検出時の DREQ 入力検出タイミング	10-31
図 10.15	バーストモード・エッジ検出時の DREQ 入力検出タイミング	10-31
図 10.16	バーストモード・レベル検出時の DREQ 入力検出タイミング	10-32
図 10.17	DMA 転送終了信号タイミング (サイクルスチール・レベル検出)	10-32

図 10.18	BSC 通常メモリアクセス (ノーウエイト、アイドルサイクル 1、16 ビットデバイスへのロングワードアクセス)	10-33
11.	クロックパルス発振器 (CPG)	11-1
図 11.1	CPG のブロック図.....	11-2
図 11.2	水晶発振子使用時の注意.....	11-11
図 11.3	PLL 発振回路使用時の注意.....	11-12
12.	ウォッチドッグタイマ (WDT)	12-1
図 12.1	WDT のブロック図	12-2
図 12.2	WTCNT および WTCR への書き込み	12-6
13.	低消費電力モード.....	13-1
図 13.1	STBCR の STBY ビットとソフトウェアスタンバイモードの解除.....	13-10
図 13.2	各モード間の状態遷移図.....	13-11
14.	タイマユニット (TMU)	14-1
図 14.1	TMU のブロック図.....	14-2
図 14.2	カウント動作設定手順例.....	14-7
図 14.3	オートリロードカウンタの動作.....	14-7
図 14.4	内部クロック動作時のカウントタイミング	14-8
図 14.5	UNF のセットタイミング.....	14-8
図 14.6	ステータスフラグのクリアタイミング.....	14-9
15.	FIFO 付きシリアルインタフェース (SIOF)	15-1
図 15.1	SIOF のブロック図.....	15-2
図 15.2	シリアルクロック供給.....	15-24
図 15.3	シリアルデータ同期タイミング.....	15-25
図 15.4	SIOF 送受信タイミング.....	15-26
図 15.5	送受信データビットアライメント.....	15-28
図 15.6	制御データビットアライメント.....	15-29
図 15.7	制御データインタフェース(スロット位置).....	15-30
図 15.8	制御データインタフェース (セカンダリ FS)	15-31
図 15.9	マスタ時送信動作例	15-33
図 15.10	マスタ時受信動作例	15-34
図 15.11	スレーブ時送信動作例.....	15-35
図 15.12	スレーブ時受信動作例.....	15-36
図 15.13	送受信タイミング (8 ビットモノラル - 1)	15-39
図 15.14	送受信タイミング (8 ビットモノラル - 2)	15-40
図 15.15	送受信タイミング (16 ビットモノラル - 1)	15-40

図 15.16	送受信タイミング (16 ビットステレオ - 1)	15-41
図 15.17	送受信タイミング (16 ビットステレオ - 2)	15-41
図 15.18	送受信タイミング (16 ビットステレオ - 3)	15-42
図 15.19	送受信タイミング (16 ビットステレオ - 4)	15-42
図 15.20	送受信タイミング (16 ビットステレオ)	15-43
図 15.21	SPI モードの構成例.....	15-43
図 15.22	SPI データ/クロックのタイミング 1 (CPHA = 0)	15-45
図 15.23	SPI データ/クロックのタイミング 2 (CPHA = 1)	15-45
16.	FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)	16-1
図 16.1	SCIF のブロック図.....	16-3
図 16.2	SCIF の初期化フローチャートの例	16-25
図 16.3	シリアル送信のフローチャートの例	16-26
図 16.4	送信時の動作例 (8 ビットデータ/パリティあり / 1 ストップビットの例)	16-27
図 16.5	送信データストップ機能の動作例.....	16-28
図 16.6	送信データストップ機能のフローチャート	16-28
図 16.7	シリアル受信のフローチャートの例 (1)	16-29
図 16.8	シリアル受信のフローチャートの例 (2)	16-30
図 16.9	SCIF の受信時の動作例 (8 ビットデータ/パリティあり / 1 ストップビットの例)	16-31
図 16.10	$\overline{\text{CTS}}$ 制御の動作例.....	16-32
図 16.11	$\overline{\text{RTS}}$ 制御の動作例.....	16-32
図 16.12	クロック同期式通信のデータフォーマット	16-33
図 16.13	送信動作時の初期化フローチャートの例 (1)	16-34
図 16.13	受信動作時の初期化フローチャートの例 (2)	16-35
図 16.13	同時送受信時の初期化フローチャートの例 (3)	16-36
図 16.14	送信動作時のフローチャートの例 (初期化後 1 回目の送信動作) (1)	16-37
図 16.14	送信動作時のフローチャートの例 (2 回目以降の送信動作) (2)	16-38
図 16.15	受信動作時のフローチャートの例 (初期化後 1 回目の受信動作) (1)	16-39
図 16.15	受信動作時のフローチャートの例 (2 回目以降の受信動作) (2)	16-40
図 16.16	同時送受信動作時のフローチャートの例 (初期化後 1 回目の送受信動作) (1)	16-41
図 16.16	同時送受信動作時のフローチャートの例 (2 回目以降の送受信動作) (2)	16-42
図 16.17	調歩同期モードの受信データサンプリングタイミング	16-45
17.	ブート機能 (BOOT)	17-1
図 17.1	BOOT 機能のブロック図.....	17-2
図 17.2	ブートモードでの外部アドレス空間	17-3
図 17.3	ブート処理の実行手順.....	17-6
18.	USB ビンマルチプレクスコントローラ (USBPM)	18-1
図 18.1	USB 機能のブロック図.....	18-2

図 18.2	USB ファンクションコントローラと外部回路の接続例 1	18-6
図 18.3	USB ファンクションコントローラと外部回路の接続例 2	18-6
図 18.4	USB ホストコントローラと外部回路の接続例	18-8
19.	USB ホストコントローラ (USBH)	19-1
図 19.1	ステータスライトバックの実施タイミング	19-32
20.	USB ファンクションコントローラ (USBF)	20-1
図 20.1	USB ファンクションコントローラのブロック図	20-3
図 20.2	エンドポイント構成例 (1)	20-32
図 20.3	エンドポイント構成例 (2)	20-34
図 20.4	ケーブル接続時の動作	20-35
図 20.5	ケーブル切断時の動作	20-36
図 20.6	コントロール転送における各転送ステージ	20-36
図 20.7	セットアップステージの動作	20-37
図 20.8	データステージ (コントロールイン時) の動作	20-38
図 20.9	データステージ (コントロールアウト時) の動作	20-39
図 20.10	ステータスステージ (コントロールイン時) の動作	20-40
図 20.11	ステータスステージ (コントロールアウト時) の動作	20-41
図 20.12	EP1 インタラプトイン転送の動作	20-42
図 20.13	EP2i バルクイン転送の動作	20-43
図 20.14	EP2o バルクアウト転送の動作	20-44
図 20.15	EP3i アイソクロナスイン転送の動作	20-45
図 20.16	EP3o アイソクロナスアウト転送の動作	20-47
図 20.17	アプリケーションで強制的にストールさせたい場合	20-51
図 20.18	USB ファンクションコントローラが自動的にストールさせた場合	20-52
図 20.19	USB ファンクションコントローラと外部回路の接続例	20-53
図 20.20	TR 割り込みフラグのセットタイミング	20-55
図 20.21	ソフトウェアスタンバイモードへの遷移と解除のフロー例	20-56
21.	ブルートゥースインタフェース (BT)	21-1
図 21.1	BT のブロック図	21-2
図 21.2	RF IC との接続例	21-4
図 21.3	Voice Codec (STLC7550) との接続例	21-5
図 21.4 (1)	Voice Codec (MC145483) との接続例	21-6
図 21.4 (2)	Voice Codec (MC145483) と接続した場合のタイミング	21-6
図 21.5	周波数変換回路の機能	21-8
図 21.6	32.768kHz から変換した疑似 32kHz クロックの誤差	21-8
図 21.7	水晶発振子使用時の注意	21-9
図 21.8	リセット信号のタイミング	21-10

図 21.9	本 LSI が起動しない場合	21-11
図 21.10	対策案 1	21-12
図 21.11	対策案 2	21-12
図 21.12	本 LSI を単独で使用する場合の対策案	21-13
図 21.13	対策案 3	21-13
22.	D/A コンバータ (DAC)	22-1
図 22.1	DAC のブロック図	22-1
図 22.2	D/A 変換動作の例	22-4
23.	I/O ポート	23-1
図 23.1	ポート A	23-1
図 23.2	ポート B	23-3
図 23.3	ポート D	23-5
図 23.4	ポート E	23-7
図 23.5	ポート G	23-9
図 23.6	ポート H	23-11
25.	ユーザブレークコントローラ (UBC)	25-1
図 25.1	UBC のブロック図	25-2
26.	ユーザデバッグインタフェース (H-UDI)	26-1
図 26.1	H-UDI ブロック図	26-1
図 26.2	TAP コントローラ状態遷移図	26-9
図 26.3	H-UDI データ転送タイミング	26-10
図 26.4	相互干渉しないリセット系信号の設計例	26-13
28.	電气的特性	28-1
	電源投入シーケンス	28-2
図 28.1	EXTAL クロック入力タイミング	28-8
図 28.2	CKIO クロック入力タイミング	28-8
図 28.3	CKIO クロック出力タイミング	28-8
図 28.4	スタンバイ復帰時発振安定時間 (リセットによる復帰)	28-9
図 28.5	スタンバイ復帰時発振安定時間 (NMI、IRQ による復帰)	28-9
図 28.6	パワーオンリセット時のタイミングチャート (電源 ON から発振するクロックを RDI_REFCLK_IN 端子および EXTAL 端子に供給した場合)	28-10
図 28.7	通常のリセット時のタイミングチャート (電源 ON から発振するクロックを RDI_REFCLK_IN 端子および EXTAL 端子に供給した場合)	28-10

図 28.8	パワーオンリセット時のタイミングチャート (弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子のみに接続し、EXTAL 端子には電源 ON から発振する、BRCLK とは別のクロックを供給した場合)	28-11
図 28.9	通常のリセット時のタイミングチャート (弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子のみに接続し、EXTAL 端子には電源 ON から発振する、BRCLK とは別のクロックを供給した場合)	28-11
図 28.10	パワーオンリセット時のタイミングチャート (弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子および EXTAL 端子に接続し、電源投入時の $\overline{\text{TRST}}$ 端子および $\overline{\text{RESETP}}$ 端子に供給する信号を個別に供給した場合)	28-12
図 28.11	通常のリセット時のタイミングチャート (弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子および EXTAL 端子に接続し、電源投入時の $\overline{\text{TRST}}$ 端子および $\overline{\text{RESETP}}$ 端子に供給する信号を個別に供給した場合)	28-12
図 28.12	リセットおよび $\overline{\text{BOOT-E}}$ 入力タイミング	28-14
図 28.13	割り込み信号入力タイミング	28-14
図 28.14	$\overline{\text{IRQOUT}}$ タイミング	28-14
図 28.15	$\overline{\text{REFOUT}}$ タイミング	28-15
図 28.16	バス権解放タイミング	28-15
図 28.17	スタンバイ時の端子ドライブタイミング	28-15
図 28.18	通常空間基本バスサイクル (ノーウェイト)	28-18
図 28.19	通常空間基本バスサイクル (ソフトウェアウェイト 1)	28-19
図 28.20	通常空間基本バスサイクル (非同期外部ウェイト 1 挿入)	28-20
図 28.21	通常空間基本バスサイクル (ソフトウェアウェイト 1、非同期外部ウェイト有効 (WM ビット=0)、アイドルサイクルなし設定)	28-21
図 28.22	通常空間 CS 拡張バスサイクル (SW = 1 サイクル、HW = 1 サイクル、非同期外部ウェイト 1 挿入)	28-22
図 28.23	バイト選択付き SRAM バスサイクル (SW = 1 サイクル、HW = 1 サイクル、非同期外部ウェイト 1 挿入、BAS = 0 (ライトサイクル UB、LB コントロール))	28-23
図 28.24	バイト選択付き SRAM バスサイクル (SW = 1 サイクル、HW = 1 サイクル、非同期外部ウェイト 1 挿入、BAS = 1 (ライトサイクル WE コントロール))	28-24
図 28.25	パースト ROM リードバスサイクル (ソフトウェアウェイト 1、非同期外部ウェイト 1 挿入、パーストウェイト 1、2 パースト)	28-25
図 28.26	シンクロナス DRAM シングルリードバスサイクル (オートプリチャージモード、CAS レイテンシ 2、TRCD = 1 サイクル、TRP = 1 サイクル)	28-26
図 28.27	シンクロナス DRAM シングルリードバスサイクル (オートプリチャージモード、CAS レイテンシ 2、TRCD = 2 サイクル、TRP = 2 サイクル)	28-27
図 28.28	シンクロナス DRAM パーストリードバスサイクル (シングルリード × 4) (オートプリチャージモード、CAS レイテンシ 2、TRCD = 1 サイクル、TRP = 2 サイクル)	28-28
図 28.29	シンクロナス DRAM パーストリードバスサイクル (シングルリード × 4) (オートプリチャージモード、CAS レイテンシ 2、TRCD = 2 サイクル、TRP = 1 サイクル)	28-29
図 28.30	シンクロナス DRAM シングルライトバスサイクル (オートプリチャージモード、TRWL = 1 サイクル)	28-30

☒ 28.31	シンクロナス DRAM シングルライトバスサイクル (オートプリチャージモード、TRCD=3 サイクル、TRWL=1 サイクル)	28-31
☒ 28.32	シンクロナス DRAM パーストライトバスサイクル (シングルライト×4) (オートプリチャージモード、TRCD=1 サイクル、TRWL=1 サイクル)	28-32
☒ 28.33	シンクロナス DRAM パーストライトバスサイクル (シングルライト×4) (オートプリチャージモード、TRCD=2 サイクル、TRWL=1 サイクル)	28-33
☒ 28.34	シンクロナス DRAM パーストリードバスサイクル (シングルリード×4) (バンクアクティブモード：ACTV+READ コマンド、CAS レイテンシ 2、 TRCD=1 サイクル)	28-34
☒ 28.35	シンクロナス DRAM パーストリードバスサイクル (シングルリード×4) (バンクアクティブモード：READ コマンド、同一ロウアドレス、CAS レイテンシ 2、 TRCD=1 サイクル)	28-35
☒ 28.36	シンクロナス DRAM パーストリードバスサイクル (シングルリード×4) (バンクアクティブモード：PRE+ACTV+READ コマンド、異なるロウアドレス、 CAS レイテンシ 2、TRCD=1 サイクル)	28-36
☒ 28.37	シンクロナス DRAM パーストライトバスサイクル (シングルライト×4) (バンクアクティブモード、ACTV+WRIT コマンド、TRCD=1 サイクル)	28-37
☒ 28.38	シンクロナス DRAM パーストライトバスサイクル (シングルライト×4) (バンクアクティブモード、ACTV+WRIT コマンド、TRCD=1 サイクル)	28-38
☒ 28.39	シンクロナス DRAM パーストライトバスサイクル (シングルライト×4) (バンクアクティブモード、PRE+ACTV+WRIT コマンド、TRCD=1 サイクル)	28-39
☒ 28.40	シンクロナス DRAM オートリフレッシュタイミング (TRP=2 サイクル)	28-40
☒ 28.41	シンクロナス DRAM セルフリフレッシュタイミング (TRP=2 サイクル)	28-41
☒ 28.42	シンクロナス DRAM パワーオンシーケンス (モードライトタイミング、TRP=2 サイクル)	28-42
☒ 28.43	シンクロナス DRAM 低周波モードでのアクセスタイミング (オートプリチャージモード、TRWL=1 サイクル)	28-43
☒ 28.44	シンクロナス DRAM 低周波モードオートリフレッシュタイミング (TRP=2 サイクル)	28-44
☒ 28.45	シンクロナス DRAM 低周波モードセルフリフレッシュタイミング (TRP=2 サイクル)	28-45
☒ 28.46	シンクロナス DRAM 低周波モードパワーオンシーケンス (モードライトタイミング、TRP=2 サイクル)	28-46
☒ 28.47	シンクロナス DRAM パワーダウンモードライトリードバスサイクル (オートプリチャージモード、TRCD=1 サイクル、TRP=1 サイクル、 TRWL=1 サイクル)	28-47
☒ 28.48	シンクロナス DRAM パワーダウンモードリードライトバスサイクル (オートプリチャージモード、TRCD=1 サイクル、TRP=1 サイクル、 TRWL=1 サイクル)	28-48
☒ 28.49	I/O ポートタイミング	28-49
☒ 28.50	DREQ 入力タイミング (DREQ ローレベル検出)	28-49
☒ 28.51	DACK、TEND 出力タイミング	28-49
☒ 28.52	SIOF_MCLK 入力タイミング	28-50
☒ 28.53	SIOF 送受信タイミング (マスタモード 1・立ち下がりサンプリング時)	28-51
☒ 28.54	SIOF 送受信タイミング (マスタモード 1・立ち上がりサンプリング時)	28-51

図 28.55	SIOF 送受信タイミング (マスタモード 2・立ち下がりサンプリング時)	28-52
図 28.56	SIOF 送受信タイミング (マスタモード 2・立ち上がりサンプリング時)	28-52
図 28.57	SIOF 送受信タイミング (スレーブモード 1・スレーブモード 2 時)	28-53
図 28.58	SIOF 送受信タイミング (SPI モード、CPHA = 0、SSAST[1 : 0] = 01)	28-53
図 28.59	SIOF 送受信タイミング (SPI モード、CPHA = 1、SSAST[1 : 0] = 01)	28-54
図 28.60	SCIF 入力クロックサイクル	28-54
図 28.61	SCIF クロック同期式モード時の入出力タイミング	28-55
図 28.62	USB クロックタイミング	28-55
図 28.63	USB トランシーバタイミング	28-56
図 28.64	USB トランシーバ特性評価回路 (Full Speed 時)	28-56
図 28.65	ブルートゥースインタフェースモジュールクロックタイミング	28-57
図 28.66	ブルートゥースインタフェースモジュールローパワークロックタイミング	28-58
図 28.67	ブルートゥースインタフェース (BT) Voice Codec(STLC7550)インタフェース信号の タイミング	28-59
図 28.68	ブルートゥースインタフェース (BT) Voice Codec(MC145483)インタフェース信号の	28-59
図 28.69	ブルートゥースインタフェース (BT) RF 用 SPI インタフェース信号のタイミング	28-60
図 28.70	ブルートゥースインタフェース (BT) 受信データタイミング	28-60
図 28.71	出力負荷回路	28-61
付録	付録-1
図 B.1	TBP-208A 外形寸法図	付録-7

表目次

1. 概要	1-1
表 1.1 SH7660 の特長	1-2
表 1.2 ピンごとの機能および初期状態一覧表	1-9
表 1.3 SH7660 端子機能	1-17
2. CPU	2-1
表 2.1 レジスタの初期値	2-7
表 2.2 アドレッシングモードと実効アドレス	2-17
表 2.3 命令形式	2-20
表 2.4 機能別命令	2-22
表 2.5 データ転送命令	2-25
表 2.6 算術演算命令	2-26
表 2.7 論理演算命令	2-27
表 2.8 シフト命令	2-28
表 2.9 分岐命令	2-28
表 2.10 システム制御命令	2-29
表 2.11 オペレーションコードマップ	2-32
3. DSP 演算ユニット	3-1
表 3.1 論理アドレス空間	3-3
表 3.2 各処理モードにおける SR の各ビットの動作説明	3-6
表 3.3 RS および RE のアドレス設定ルール	3-10
表 3.4 リピート制御命令	3-10
表 3.5 リピート制御マクロ	3-11
表 3.6 DSP モード拡張システム制御命令	3-12
表 3.7 リピート制御中の PC の値 (RC[11:0] \geq 2 の場合)	3-14
表 3.8 拡張リピート制御命令	3-17
表 3.9 DSP モード拡張システム制御命令	3-21
表 3.10 データ転送命令の関係	3-23
表 3.11 モジュールアドレッシング制御命令	3-25
表 3.12 ダブルデータ転送の命令形式	3-26
表 3.13 シングルデータ転送命令の命令形式	3-27
表 3.14 DSP 命令のデスティネーションレジスタ	3-29
表 3.15 DSP 命令のソースレジスタ	3-29

表 3.16	DSR レジスタのビットの説明.....	3-30
表 3.17	DSP 命令の命令形式.....	3-32
表 3.18	DSP 命令のオペランドとレジスタの対応.....	3-32
表 3.19	DC ビットの更新ルール.....	3-33
表 3.20	NOPX と NOPY の命令コードの例.....	3-35
表 3.21	ALU 固定小数点算術演算の種類.....	3-37
表 3.22	オペランドとレジスタの対応.....	3-38
表 3.23	ALU 整数演算の種類.....	3-42
表 3.24	ALU 論理演算の種類.....	3-43
表 3.25	固定小数点乗算の種類.....	3-44
表 3.26	オペランドの柔軟性.....	3-45
表 3.27	シフト演算の種類.....	3-46
表 3.28	PDMSB 命令の定義.....	3-51
表 3.29	PDMSB 命令の種類.....	3-51
表 3.30	丸め演算の種類.....	3-53
表 3.31	固定小数点算術演算用オーバフロー防止機能の定義.....	3-53
表 3.32	整数算術演算用オーバフロー防止機能の定義.....	3-54
表 3.33	ローカルデータ移動命令の種類.....	3-54
表 3.34	オペランド柔軟性.....	3-55
表 3.35	DSP モード拡張システム制御命令.....	3-56
表 3.36	ダブルデータ転送命令.....	3-57
表 3.37	シングルデータ転送命令.....	3-58
表 3.38	DSP データ転送のオペランドとレジスタとの対応.....	3-59
表 3.39	DSP データ演算命令.....	3-60
表 3.40	オペレーションコードマップ.....	3-65
4.	例外処理.....	4-1
表 4.1	例外事象一覧.....	4-6
表 4.2	リピートループ中の命令位置と制約の種別.....	4-12
表 4.3	リピート制御中に生じた再実行型例外に対するの SPC の値 (SR.RC[11:0]>=2 の場合).....	4-13
表 4.4	リピートループでの例外受け付けの制限.....	4-15
表 4.5	リピート制御中のメモリアクセス例外で特殊な例外を生じる命令 (SR.RC[11:0]>=1).....	4-15
5.	キャッシュ.....	5-1
表 5.1	LRU ビットと置き換えられるウェイ (キャッシュロック機能を使用しない場合).....	5-3
表 5.2	PREF 命令がキャッシュミスした場合に置き換えられるウェイ.....	5-5
表 5.3	PREF 命令以外がキャッシュミスした場合に置換されるウェイ.....	5-5
表 5.4	LRU ビットと置き換えられるウェイ (W2LOCK=1 かつ W3LOCK=0 の場合).....	5-6
表 5.5	LRU ビットと置き換えられるウェイ (W2LOCK=0 かつ W3LOCK=1 の場合).....	5-6
表 5.6	LRU ビットと置き換えられるウェイ (W2LOCK=1 かつ W3LOCK=1 の場合).....	5-6

6.	X/Y メモリ	6-1
表 6.1	X/Y メモリ論理アドレス.....	6-1
表 6.2	キャッシュの設定	6-3
7.	U メモリ	7-1
表 7.1	U メモリ論理アドレス.....	7-1
表 7.2	キャッシュの設定	7-3
8.	割り込みコントローラ (INTC)	8-1
表 8.1	端子構成	8-3
表 8.2	割り込み要求要因と IPRA ~ IPRH.....	8-5
表 8.3	割り込み要因と IMR0 ~ IMR9 および IMCR0 ~ IMCR9 の対応	8-10
表 8.4	割り込み例外処理要因と優先順位	8-13
9.	バスステートコントローラ (BSC)	9-1
表 9.1	端子構成	9-4
表 9.2	外部アドレス空間のアドレスマップ	9-6
表 9.3	16 ビット外部デバイス / ビッグエンディアンのアクセスとデータアライメント	9-27
表 9.4	8 ビット外部デバイス / ビッグエンディアンのアクセスとデータアライメント	9-27
表 9.5	16 ビット外部デバイス / リトルエンディアンのアクセスとデータアライメント	9-28
表 9.6	8 ビット外部デバイス / リトルエンディアンのアクセスとデータアライメント	9-28
表 9.7	CS3BCR.BSZ[1:0]、SDCR.A3ROW[1:0]、SDCR.A3COL[1:0]と アドレスマルチプレクスの関係 (1)	9-38
表 9.8	CS3BCR.BSZ[1:0]、SDCR.A3ROW[1:0]、SDCR.A3COL[1:0]と アドレスマルチプレクスの関係 (2)	9-39
表 9.9	CS3BCR.BSZ[1:0]、SDCR.A3ROW[1:0]、SDCR.A3COL[1:0]と アドレスマルチプレクスの関係 (3)	9-40
表 9.10	アクセスサイズとバースト数の関係	9-41
表 9.11	SDRAM モードレジスタライト時のアクセスアドレス	9-58
表 9.12	データバス幅およびアクセスサイズとバースト数の関係	9-60
10.	ダイレクトメモリアクセスコントローラ (DMAC)	10-1
表 10.1	端子構成	10-3
表 10.2	DMARS の設定	10-14
表 10.3	RS ビットによる外部リクエストモードの選択	10-17
表 10.4	DL、DS ビットによる外部リクエスト検出の選択	10-18
表 10.5	DL、DS ビットによる外部リクエスト検出の選択	10-18
表 10.6	RS3 ~ 0 ビットによる内蔵周辺モジュールリクエストモードの選択	10-19
表 10.7	サポートできる DMA 転送	10-23
表 10.8	DMA 転送区間とリクエストモード、バスモードとの関連一覧	10-29

11. クロックパルス発振器 (CPG)	11-1
表 11.1 端子構成	11-4
表 11.2 クロック動作モード	11-5
表 11.3 クロックモードと FRQCR 値の可能な組み合わせ	11-6
13. 低消費電力モード	13-1
表 13.1 低消費電力モードの状態	13-2
表 13.2 端子構成	13-3
表 13.3 ソフトウェアスタンバイモード時のレジスタの状態	13-9
14. タイマユニット (TMU)	14-1
表 14.1 TMU の割り込み要因	14-9
15. FIFO 付きシリアルインタフェース (SIOF)	15-1
表 15.1 端子構成	15-2
表 15.2 転送モードごとの動作	15-5
表 15.3 SIOF シリアルクロック周波数例	15-24
表 15.4 シリアル転送モード	15-26
表 15.5 フレーム長	15-27
表 15.6 送信データ音声モード	15-28
表 15.7 受信データ音声モード	15-29
表 15.8 制御データ ch.数設定	15-29
表 15.9 送信要求発行条件	15-32
表 15.10 受信要求発行条件	15-32
表 15.11 送受信リセット	15-37
表 15.12 SIOF 割り込み要因	15-38
表 15.13 SPI 送受信の動作状態	15-44
16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)	16-1
表 16.1 端子構成	16-3
表 16.2 SCSMR の設定値とシリアル送信 / 受信フォーマット	16-23
表 16.3 シリアル送信 / 受信フォーマット	16-24
表 16.4 SCIF 割り込み要因	16-44
17. ブート機能 (BOOT)	17-1
表 17.1 端子構成	17-2
表 17.2 ブート機能使用時のクロック周波数と SCIF の転送レート	17-7

18.	USB ビンマルチプレクスコントローラ (USBPM)	18-1
表 18.1	端子構成 (アナログトランシーバ信号)	18-2
表 18.2	端子構成 (電源制御信号)	18-3
表 18.3	端子構成 (クロック制御信号)	18-3
19.	USB ホストコントローラ (USBH)	19-1
表 19.1	端子構成	19-1
20.	USB ファンクションコントローラ (USBF)	20-1
表 20.1	端子構成	20-3
表 20.2	設定可能値の制約一覧	20-30
表 20.3	エンドポイント構成例 (1)	20-31
表 20.4	エンドポイント構成情報設定例 (1)	20-32
表 20.5	エンドポイント構成例 (2)	20-33
表 20.6	エンドポイント構成情報設定例 (2)	20-34
表 20.7	アプリケーション側でのコマンドデコード	20-49
21.	Bluetooth インタフェース (BT)	21-1
表 21.1	端子構成	21-3
表 21.2	選択条件と機能一覧	21-7
22.	D/A コンバータ (DAC)	22-1
表 22.1	端子構成	22-2
23.	I/O ポート	23-1
表 23.1	ポート A データレジスタ (PADR) の読み出し / 書き込み動作	23-2
表 23.2	ポート B データレジスタ (PBDR) の読み出し / 書き込み動作	23-4
表 23.3	ポート D データレジスタ (PDDR) の読み出し / 書き込み動作	23-6
表 23.4	ポート E データレジスタ (PEDR) の読み出し / 書き込み動作	23-8
表 23.5	ポート G データレジスタ (PGDR) の読み出し / 書き込み動作	23-10
表 23.6	ポート H データレジスタ (PHDR) の読み出し / 書き込み動作	23-12
24.	ピンファンクションコントローラ (PFC)	24-1
表 24.1	マルチプレクス一覧表	24-1
25.	ユーザブレイクコントローラ (UBC)	25-1
表 25.1	ブレイクアドレスレジスタの指定	25-5
表 25.2	ブレイクデータレジスタの指定	25-7
表 25.3	データアクセスサイクルアドレスおよびオペランドサイズの比較条件	25-17

26. ユーザデバッグインタフェース (H-UDI)	26-1
表 26.1 端子構成	26-2
表 26.2 JTAG コマンド	26-4
表 26.3 本 LSI の端子とバウンダリスキャンレジスタの対応	26-5
表 26.4 リセット構成	26-10
表 26.5 本 LSI での ASE モードにおけるパワーオンリセット動作	26-14
28. 電気的特性	28-1
表 28.1 絶対最大定格	28-1
表 28.2 DC 特性 (1) 【共通項目】	28-3
表 28.2 DC 特性 (2-a) 【DAC、USB 関連端子を除く】	28-4
表 28.2 DC 特性 (2-b) 【USB 関連端子】	28-5
表 28.2 DC 特性 (2-c) 【USB トランシーバ関連端子】	28-5
表 28.3 出力許容電流値	28-6
表 28.4 動作周波数	28-6
表 28.5 クロックタイミング	28-6
表 28.6 制御信号タイミング	28-13
表 28.7 バスタイミング	28-16
表 28.8 周辺モジュール信号タイミング	28-49
表 28.9 SIOF モジュール信号タイミング	28-50
表 28.10 SCIF モジュール信号タイミング	28-54
表 28.11 USB モジュールクロックタイミング	28-55
表 28.12 USB モジュールクロックタイミング	28-55
表 28.13 USB トランシーバタイミング (Full Speed 動作時)	28-56
表 28.14 ブルートゥースインタフェースモジュールクロックタイミング	28-57
表 28.15 ブルートゥースインタフェースモジュールクロックタイミング	28-57
表 28.16 ブルートゥースインタフェースモジュールローパワークロックタイミング	28-57
表 28.17 ブルートゥースインタフェースモジュールローパワークロックタイミング	28-58
表 28.18 ブルートゥースインタフェース (BT) Voice Codec インタフェース信号のタイミング	28-58
表 28.19 ブルートゥースインタフェース (BT) RF 用 SPI インタフェース信号のタイミング	28-59
表 28.20 受信データタイミング	28-60
表 28.21 D/A コンバータ特性	28-61
付録	付録-1
表 A.1 端子状態	付録-1

1. 概要

1.1 SH7660 の特長

本 LSI は、32 ビット RISC タイプ SuperH アーキテクチャの CPU とデジタル信号処理 (DSP) 拡張機能をコアとし、16k バイトのキャッシュメモリ、16k バイトの X/Y メモリ、大容量 128k バイトの U メモリ、およびシステム構成に必要な割り込みコントローラなどを集積した RISC マイクロプロセッサです。

内蔵 DMAC (Direct Memory Access Controller) による高速データ転送や、外部メモリアクセスサポート機能による各種メモリへの直結が可能です。さらに、USB (ホスト/ファンクション) や 921k ビット/秒の高速非同期シリアルインタフェース回路、音声/オーディオ CODEC 用シリアルインタフェース回路、DA コンバータなどのシステムコンフィギュレーションに最適で強力な周辺機能を内蔵しています。

本 LSI はまた、Bluetooth™ のインタフェースも周辺機能の 1 つとしてサポートしています。ファームウェアを含めて HCI 階層までの Bluetooth プロトコルスタック機能を標準で備えており、さらに上位プロトコルスタックや各種アプリケーションプロファイルを搭載して、Bluetooth ベースバンド機能を一括して実現できる性能を持っています。HCI 階層以下のプロトコルスタック機能を扱うための API を用意していますので、従来の周辺機能と同様に、容易に本 Bluetooth インタフェースを扱うことができます。本 Bluetooth インタフェースは、ルネサス テクノロジ製の RF チップ (HD157102NP) と直結可能です。

【注】 Bluetooth は Bluetooth SIG, Inc., U.S.A. の登録商標であり、ルネサス テクノロジはライセンス契約によりこれを使用しております。

本 LSI の特長を表 1.1 に示します。

1. 概要

表 1.1 SH7660 の特長

項 目	特 長
CPU	<ul style="list-style-type: none"> • ルネサス テクノロジ独自の SuperH アーキテクチャ • SH-1、SH-2、および SH-3 とオブジェクトコードレベルで互換性あり • 32 ビット内部データバスを内蔵 • 豊富なレジスタ群をサポート <ul style="list-style-type: none"> 汎用レジスタ：32 ビットのレジスタを 16 本内蔵（8 本の 32 ビットバンクレジスタを含む） コントロールレジスタ：32 ビットのレジスタを 5 本内蔵 システムレジスタ：32 ビットのレジスタを 4 本内蔵 • RISC タイプ命令セットをサポート <ul style="list-style-type: none"> 命令長は、コード効率改善のための 16 ビット固定長を使用 ロードストアアーキテクチャ 遅延分岐命令 C 言語指向の命令セット • パレルシフト命令、積和演算命令をサポート • 基本命令の命令実行時間は、1 命令 / サイクルで対応 • 論理アドレス空間は、4G バイトに対応 • 5 段パイプラインをサポート
DSP ユニット	<ul style="list-style-type: none"> • 16 ビット命令、および 32 ビット命令の混在可能 • 32 または 40 ビットの内部データバスを内蔵 • 乗算器、ALU、パレルシフトに対応 • 16 ビット x 16 ビットに対応する 32 ビットワンサイクル乗算器を内蔵 • 大容量の DSP データレジスタファイルをサポート <ul style="list-style-type: none"> 6 本の 32 ビットデータレジスタ 2 本の 40 ビットデータレジスタ • DSP データバス用の拡張ハーバードアーキテクチャをサポート <ul style="list-style-type: none"> 2 本のデータバス 1 本の命令バス • 最大 4 つのパラレル演算を実行可能 <ul style="list-style-type: none"> ALU、乗算、2 つのロード / ストア • 2 つのメモリアクセス用のアドレスを生成するための 2 本のアドレスユニットを装備 • DSP データアドレッシングモードをサポート <ul style="list-style-type: none"> インクリメント、およびインデクシング（モジュロアドレッシングあり / なし） • ゼロオーバーヘッドリピートループ制御に対応 • 条件付実行命令に対応 • ユーザ DSP モードおよび特権 DSP モードをサポート

項 目	特 長
キャッシュ メモリ	<ul style="list-style-type: none"> • 16k バイトキャッシュ、命令とデータの混在可能 • 256 エントリ、4 ウェイセットアソシアティブ、および 16 バイトブロック長に対応 • ライトバック、ライトスルー、および LRU 置換アルゴリズムに対応 • 1 段階ライトバックバッファを内蔵 • 最大 2 つのウェイをロック可能
XY メモリ	<ul style="list-style-type: none"> • 3 本の独立した読み出しおよび書き込みポート CPU からの 8、16、および 32 ビットアクセス DSP からの最大 2 つの 16 ビットのアクセス DMAC からの 8、16、および 32 ビットアクセス USBH からの 8 および 32 ビットアクセス • トータル 16k バイト (4k バイト × 4) のメモリを内蔵 • 割り込み要求は、なし • DMA 転送要求は、なし (転送元または転送先としてアクセス可)
U メモリ	<ul style="list-style-type: none"> • 2 本の独立した読み出しまたは書き込みポート CPU からの 8、16、および 32 ビットアクセス DSP からの 16 および 32 ビットアクセス DMAC からの 8、16、および 32 ビットアクセス USBH からの 8 および 32 ビットアクセス • 128k バイトの大容量メモリを内蔵 • 割り込み要求は、なし • DMA 転送要求は、なし (転送元または転送先としてアクセス可)
割り込み コントローラ (INTC)	<ul style="list-style-type: none"> • 5 本の外部割り込み端子 (NMI、IRQ4 ~ IRQ2、IRQ0) をサポート • 1 本の割り込み要求通知端子 (IRQOUT) をサポート • 内蔵周辺割り込みは、モジュールごとに優先順位を設定可能 • 立ち下がり、立ち上がり、ハイレベル、およびローレベルの選択可能
バスステート コントローラ (BSC)	<ul style="list-style-type: none"> • 物理アドレス空間は、それぞれ最大 16M バイトの 3 つの領域エリア 0、エリア 3、エリア 4 をサポート • 各エリアには独立に次の機能を設定可能： バスサイズ (8、16 ビット)。ただし、各エリアごとにサポートサイズは異なります。 アクセスウェイトサイクル数 (リード/ライトで独立ウェイト設定可のエリアあり) アイドルウェイトサイクル設定 (同一エリア/別エリア) エリアごとに接続するメモリを指定することによって、SRAM、バイト選択付き SRAM、SDRAM、バースト ROM との直結が可能 該当する領域にチップセレクト信号 ($\overline{CS0}$、$\overline{CS3}$、$\overline{CS4}$) を出力 (\overline{CSn} アサート/ネゲートタイミングをプログラミングで選択可) • SDRAM リフレッシュ機能 オートリフレッシュおよびセルフリフレッシュモードをサポート • SDRAM バーストアクセス機能 • ビッグエンディアンまたはリトルエンディアンを設定可能

1. 概要

項 目	特 長
DMA コントローラ (DMAC)	<ul style="list-style-type: none"> • 4チャンネルを内蔵 (うち1チャンネルは外部リクエスト受け付け可能) • バーストモードおよびサイクルスチールモードをサポート • インタミットtentサイクルスチールモードをサポート
クロックパルス 発振器 (CPG)	<ul style="list-style-type: none"> • クロックモードは、入力クロックを外部入力 (EXTAL または CKIO) および水晶発振子から選択可能 • 3種類のクロック生成に対応 内部クロック : 120MHz バスクロック : 60MHz 周辺クロック : 30MHz • パワーダウンモードのサポート ソフトウェアスタンバイモード スリープモード モジュールスタンバイモード • 3種類のクロックモード (PLL1 と PLL2 の通倍率選択、およびクロックと水晶発振子の選択が可能) • 1種類の分周器動作モード
ウォッチドッグ タイマ (WDT)	<ul style="list-style-type: none"> • 1チャンネルのウォッチドッグタイマを内蔵 • ウォッチドッグタイマモードとインターバルタイマモードを選択可能 • インターバルタイマモード時、割り込み発生可能
タイマユニット (TMU)	<ul style="list-style-type: none"> • 32ビットタイマ 3チャンネル内蔵 • 各チャンネルとも、4種類のカウンタ入力クロックを選択可能 • オートリロード方式の32ビットダウンカウンタ • P でのプリスケアラ内蔵 • 割り込み要求 : 32ビットダウンカウンタのアンダフローの発生で割り込み要求を発生
FIFO 付き シリアル インタフェース (SIOF)	<ul style="list-style-type: none"> • 1チャンネル内蔵 • FIFO容量 32ビット×16段 (送信用、受信用に独立して持つ) • 8ビット / 16ビット / 16ビットステレオ音声入出力対応 • 同期方法はフレーム同期パルス / 左右 ch.切り替えに対応 • CODEC 制御データインタフェースに対応 • リニア / オーディオ / A-Law、μ-Law CODEC チップに接続可能 • マスタ / スレーブ両モードに対応 • P および外部端子からのサンプリングレートクロック (最大 48kHz) を入力選択可能 • P でのプリスケアラ内蔵 • モジュールスタンバイ機能あり • 割り込み要求あり、DMAC 要求あり

項 目	特 長
FIFO 内蔵 シリアル コミュニケーション インタフェース (SCIF0、SCIF1)	<ul style="list-style-type: none"> • 2チャンネル内蔵 • 調歩同期式モードとクロック同期式モードをサポート • 送受信 FIFO おのおの 64 バイト内蔵 • 高速 UART • CTS/RTS 対応サポート • P でのプリスケラ内蔵 • 割り込み要求あり、DMAC 要求あり
ブート機能 (BOOT)	<ul style="list-style-type: none"> • パワーオンリセット時、通常モード/ブートモード切り替え • SCIF0 からの初期書き込みプログラム自動取り込み 内蔵メモリ (U メモリ) へのプログラムダウンロード • ダウンロードしたプログラムのオートラン機能
USB ホスト コントローラ (USBH)	<ul style="list-style-type: none"> • OHCI Ver.1.0 準拠のレジスタセット • USB2.0 Full Speed 対応 • 合計 127 個のエンドポイントサポート • コントロール/バルク/インタラプト/アイソクロナスモードをサポート • バスマスタコントローラ • USB トランシーバ 1 ポート (USB ファンクションコントローラと共通) • 割り込み要求あり
USB ファンクション コントローラ (USBF)	<ul style="list-style-type: none"> • USB2.0 Full Speed 対応 • USB トランシーバ 1 ポート (USB ホストコントローラと共通) • 合計 6 エンドポイントサポート、エンドポイント番号は切り替え可能 • アイソクロナスサポート (計 2 エンドポイント) • コントロール (エンドポイント 0)、バルク転送方式 (計 2 エンドポイント)、インタラプト (計 1 エンドポイント) をサポート • USB 標準コマンドをサポートし、クラスおよびベンダコマンドはファームウェアで処理 • エンドポイント用の FIFO バッファを内蔵 (バルク、アイソクロナス: 128 バイト/エンドポイント) • 割り込み要求あり、DMAC 要求あり
ブルートゥース インタフェース (BT)	<ul style="list-style-type: none"> • Bluetooth 規格 v1.2 に対応 • ルネサス テクノロジ RF IC (HD157102NP) との直結インタフェースをサポート • 2 種類の Voice Codec IC (STLC7550、MC145483) との直結インタフェースをサポート • 4 種類 (A-law/μ-law/CVSD/Linear PCM) の Voice CODEC に対応 • ACL/SCO リンクをサポート (SCO over HCI 可能) • 拡散変調は、周波数ホッピング方式で 79 チャンネルホッピング • RTC (Real Time Clock) 用のクロック (32.768kHz) を低消費電力状態用クロックとして利用可能 • 低消費電力状態で使用する低周波数クロックを水晶発振器モード/外部直接入力モードから選択可能 • Hold/Sniff/Park 3 種類の低消費電力状態をサポート

1. 概要

項目	特長
D/A コンバータ (DAC)	<ul style="list-style-type: none">• 8 ビット、2 チャンネル• 出力範囲：0 ~ AVcc
I/O ポート / ピンファンクション コントローラ (PFC)	<ul style="list-style-type: none">• 入出力兼用ポートはビット毎に入出力切り替え可能
ユーザブ레이크 コントローラ (UBC)	<ul style="list-style-type: none">• アドレス、データ値、アクセスタイプ、およびデータサイズはすべてブ레이크条件として設定可能• シーケンシャルブ레이크機能をサポート• 2本のブ레이크チャンネルを内蔵
ユーザデバッグ インタフェース (H-UDI)	<ul style="list-style-type: none">• E10A エミュレータをサポート• リアルタイム分岐トレースに対応• 高速エミュレーションプログラム実行用 1k バイトの RAM を内蔵
パッケージ	<ul style="list-style-type: none">• BGA 208 ピン (ピンピッチ : 0.65mm)
電源電圧	<ul style="list-style-type: none">• I/O : 3.0 ~ 3.6V (一部 2.7 ~ 3.0V)内部 : 内蔵レギュレータによる給電方式、または 1.4V ~ 1.6V 外部給電方式選択可能
動作温度範囲	<ul style="list-style-type: none">• -40 ~ +85

- 【注】
1. 本 LSI は MMU/TLB 機能をサポートしていませんので、SH-3 の CPU に備えられている LDTLB 命令は機能しません。
 2. 本 LSI 内蔵の AUD、ASERAM は、デバッグ用のモジュールなので、本マニュアルでは、説明を省略します。詳しくは開発関連ユーザズマニュアルを参照してください。

1.2 ブロック図

図 1.1 に SH7660 のブロック構成図を示します。

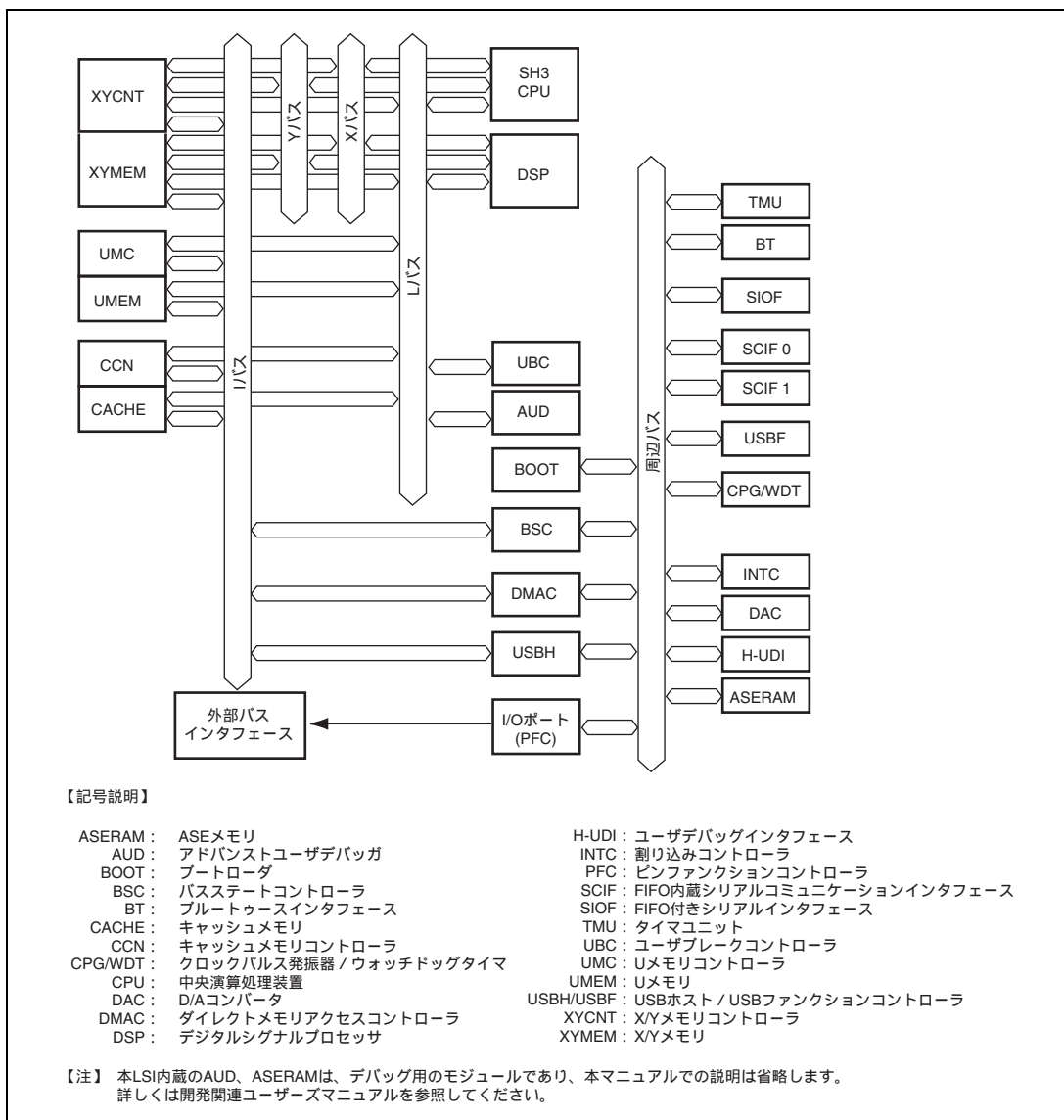


図 1.1 SH7660 の構成

1. 概要

1.3 ピン配置図

図 1.2 にピン配置図、表 1.2 にピン毎の機能および初期状態一覧を示します。

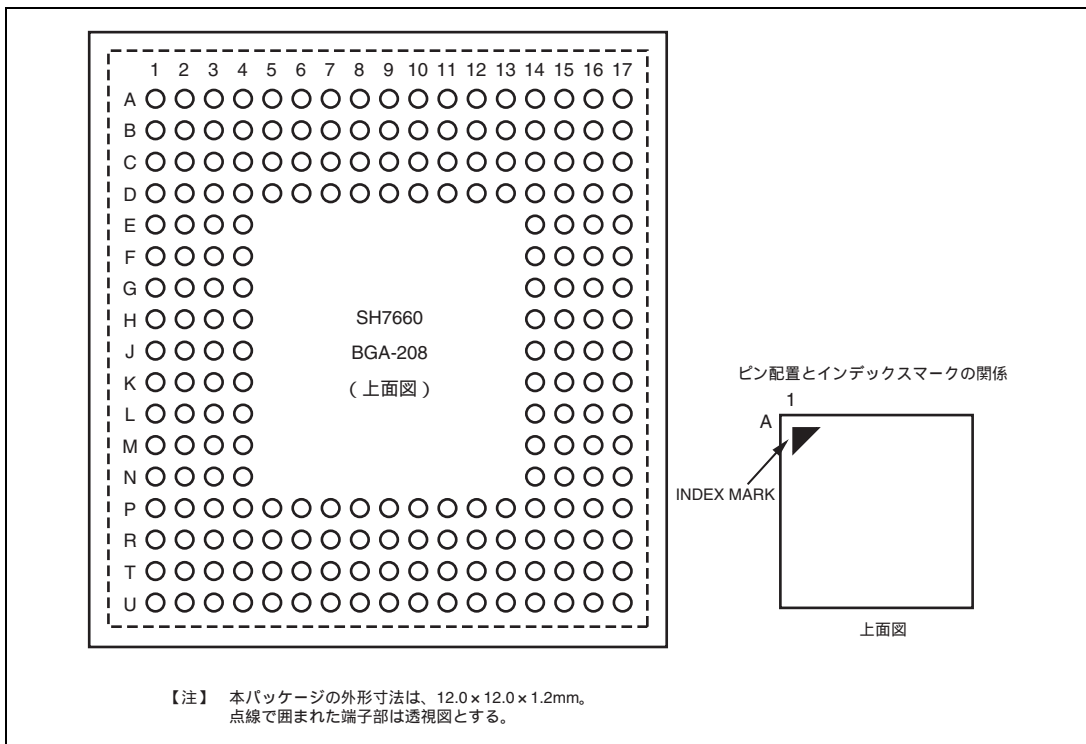


図 1.2 ピン配置図

表 1.2 ピンごとの機能および初期状態一覧表

端子番号	端子名	機 能
A1	PTH4/SCIF1_CTS	ポート H / SCIF1 用 CTS 入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
A2	NMI	ノンマスクプル割り込み要求。使用しない場合は、ハイに固定してください。
A3	PTB1/IRQ2	ポート B / 外部割り込み要求入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
A4	RTCSEL0	テスト端子です。必ずローに固定してください。
A5	NC	予約。どこにも接続しないでください。
A6	NC	予約。どこにも接続しないでください。
A7	AVss(DAC)	D/A コンバータ (DAC) 用アナログ電源 (0V)
A8	XTAL	水晶発振器端子
A9	EXTAL	外部クロック入力 / 水晶発振器端子。モード 7 のときはプルアップしてください。
A10	EXTAL2	Bluetooth 低消費電力状態用外部クロック入力 / 水晶発振器端子。未使用時はプルアップしてください。
A11	RDI_TXTRDATA* ¹	RF IC 用送受信データ入出力。初期状態は出力となります。未使用時はプルアップしてください。
A12	Vss_28	RF IC との入出力端子用電源 (0V)
A13	RDI_REFCLK_IN* ¹	RF IC とのデータ入出力用および USB 用クロック入力* ³ およびリセットホールドカウンタ用クロック入力* ⁴ 。常時入力待ち状態です。
A14	RCI_SPI_TXRX* ¹	RF IC との SPI シリアルデータ入出力。初期状態は出力となります。未使用時はプルアップしてください。
A15	USB_N	USB 内蔵トランシーバ D-入出力。初期状態はハイインピーダンスです。未使用時はプルダウンしてください。
A16	USB_P	USB 内蔵トランシーバ D+入出力。初期状態はハイインピーダンスです。未使用時はプルダウンしてください。
A17	AVss(USB)	USB トランシーバ用アナログ電源 (0V)
B1	NC	予約。どこにも接続しないでください。
B2	ASEMD0	ASE モード制御入力。通常はハイに固定してください。
B3	PTG0/SCIF0_SCK/IRQ3	ポート G / SCIF0 用クロック入出力 / 外部割り込み要求入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
B4	RREF	テスト端子です。プルアップしてください。
B5	NC	予約。どこにも接続しないでください。
B6	Vss(DLL)	内部レギュレータ用電源 (DLL) (0V)
B7	DA0	D/A コンバータ (DAC) 出力 (チャンネル 0)
B8	Vcc(I/O)	入出力用電源 (3.3V)
B9	Vss	電源 (0V)

1. 概要

端子番号	端子名	機能
B10	XTAL2	Bluetooth 低消費電力状態用水晶発振器端子。未使用時は OPEN にしてください。
B11	RDI_RXBDW_OUT* ¹	RF IC 用パケット制御出力
B12	Vcc_28	RF IC との入出力端子用電源 (2.8V/3.3V)
B13	RCI_SPI_CLK* ¹	RF IC との SPI インタフェースクロック出力
B14	NC	予約。どこにも接続しないでください。
B15	NC	予約。どこにも接続しないでください。
B16	NC	予約。どこにも接続しないでください。
B17	NC	予約。どこにも接続しないでください。
C1	PTH1/SCIF1_TXD	ポート H / SCIF1 用送信データ出力。初期状態は出力ハイインピーダンスです。
C2	PTH2/SCIF1_RXD	ポート H / SCIF1 用受信データ入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
C3	PTH3/SCIF1_RTS	ポート H / SCIF1 用 RTS 出力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
C4	PTB0/IRQ0	ポート B / 外部割込み要求入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
C5	Vss(SREG)	内部レギュレータ (サブ) 用電源 (0V)
C6	Vdd(DLL)	内部レギュレータからの電源出力 / 内部電源入力 (DLL) (1.5V)
C7	DA1	D/A コンバータ (DAC) 出力 (チャンネル 1)
C8	Vss	電源 (0V)
C9	NC	予約。どこにも接続しないでください。
C10	Vss	電源 (0V)
C11	RDI_CTRL3* ¹	RF IC 内の発振器をイネーブルにするストロブ出力
C12	Vss_28	RF IC との入出力端子用電源 (0V)
C13	RCI_SPI_ENB* ¹	RF IC との SPI インタフェースイネーブル出力
C14	AVcc(USB)	USB トランシーバ用アナログ電源 (3.3V)
C15	NC	予約。どこにも接続しないでください。
C16	UCLK	USB 用外部クロック入力。初期状態は内部 DLL クロックが有効となり、この端子は内部で入力固定されます。未使用時はプルアップしてください。
C17	USB_OVR_CRNT/USB_VBUS	USB 過電流検知入力 / USB ケーブル接続モニタ入力。未使用時はプルダウンしてください。
D1	PTG2/SCIF0_RXD	ポート G / SCIF0 用受信データ入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
D2	PTG3/SCIF0_RTS	ポート G / SCIF0 用 RTS 出力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
D3	PTG4/SCIF0_CTS	ポート G / SCIF0 用 CTS 入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。

端子番号	端子名	機能
D4	PTH0/SCIF1_SCK/IRQ4	ポート H / SCIF1 用クロック入出力 / 外部割込み要求入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
D5	Vcc(SREG)	内部レギュレータ (サブ) 用電源 (3.3V)
D6	Vcc(DLL)	内部レギュレータ用電源 (DLL) (3.3V)
D7	Vss	電源 (0V)
D8	AVcc(DAC)	D/A コンバータ (DAC) 用アナログ電源 (3.3V)
D9	Vcc(I/O)	入出力用電源 (3.3V)
D10	NC	予約。どこにも接続しないでください。
D11	RDI_CTRL4*1	RF IC 用リセット制御および低消費電力制御出力
D12	Vcc_28	RF IC との入出力端子用電源 (2.8V/3.3V)
D13	RESETP	パワーオンリセット入力
D14	USB_PWR_EN/USB_PULLUP	USB 電源投入許可制御出力 / USB プルアップ制御出力。初期状態はロー出力です。
D15	Vcc(PLL)	内部レギュレータ用電源 (PLL1/PLL2) (3.3V)
D16	Vdd(PLL)	内部レギュレータからの電源出力 / 内部電源入力 (PLL1/PLL2) (1.5V)
D17	Vss(PLL)	内部レギュレータ用電源 (PLL1/PLL2) (0V)
E1	PTA6/SIOF_SS2	ポート A / SIOF SPI モードスレーブデバイス選択。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
E2	PTG1/SCIF0_TXD	ポート G / SCIF0 用送信データ出力。初期状態は出力ハイインピーダンスです。
E3	Vss	電源 (0V)
E4	Vcc(I/O)	入出力用電源 (3.3V)
E14	Vcc(内部)	内部レギュレータ用電源 (3.3V)
E15	Vdd	内部レギュレータからの電源出力 / 内部電源入力 (1.5V)
E16	Vss	電源 (0V)
E17	CKIO	システムクロック入出力
F1	PTA5/SIOF_SS1	ポート A / SIOF SPI モードスレーブデバイス選択。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
F2	Vss	電源 (0V)
F3	Vdd	内部レギュレータからの電源出力 / 内部電源入力 (1.5V)
F4	Vcc(内部)	内部レギュレータ用電源 (3.3V)
F14	Vcc(I/O)	入出力用電源 (3.3V)
F15	Vss	電源 (0V)
F16	NC	予約。どこにも接続しないでください。
F17	MD5	エンディアン設定入力

1. 概要

端子番号	端子名	機能
G1	PTA2/SIOF_RXD(MISO)	ポート A / SIOF 用受信データ入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
G2	PTA3/SIOF_SYNC(SIOF_SS0)	ポート A / SIOF 用フレーム同期信号出力 (SIOF SPI モードスレーブデバイス選択)。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
G3	PTA4/SIOF_SCK(SCK)	ポート A / SIOF 用シリアルクロック入出力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
G4	NC	予約。どこにも接続しないでください。
G14	MD2	クロックモード設定入力
G15	MD1	クロックモード設定入力
G16	PTE6/RDI_CTRL2 ^{*1} /IRQOUT	ポート E / RF IC 用外部パワーアンプ制御出力 / 割り込み要求通知。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
G17	PTE0/VC1_CODEC_PWRDWN ^{*2} /TEND	ポート E / Voice Codec IC 用パワーダウン制御出力 / DMA 転送終了フラグ出力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
H1	TEST_REG	内部電源 (V _{DD} = 1.5V) 選択入力。内部レギュレータ使用時は、ハイ固定してください。
H2	NC	予約。どこにも接続しないでください。
H3	PTA0/SIOF_MCLK	ポート A / SIOF 用マスタクロック入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
H4	PTA1/SIOF_TXD(MOSI)	ポート A / SIOF 用送信データ出力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
H14	Vcc(内部)	内部レギュレータ用電源 (3.3V)
H15	V _{DD}	内部レギュレータからの電源出力 / 内部電源入力 (1.5V)
H16	V _{SS}	電源 (0V)
H17	PTE1/VC1_SCO_TX ^{*2} /DACK	ポート E / Voice Codec IC からの送信データ入力 / DMA 転送要求受付。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
J1	TRST	H-UDI リセット入力。内部でプルアップされています。本端子への信号供給については、「21.9 基板設計の制約事項」を参照ください。
J2	ASEBRKAK	ASE ブレークアクトリッジ出力
J3	TDO	H-UDI データ出力
J4	TEST	テスト用入力。常時ハイに固定してください。
J14	Vcc(I/O)	入出力用電源 (3.3V)
J15	PTE2/VC1_SCO_RX ^{*2} /DREQ	ポート E / Voice Codec IC への受信データ出力 / DMA 転送要求入力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。

端子番号	端子名	機能
J16	PTE4/VCI_SCO_SYNC_OUT* ² / BREQ	ポート E / Voice Codec IC 用フレーム同期信号出力 / バス権要求入力。初期状態はバス権要求入力が有効です。外部でプルアップしてください。
J17	PTE3/VCI_SCO_CLK_OUT* ² / BACK	ポート E / Voice Code IC 用クロック出力 / バス解放アクノリッジ出力。初期状態はバス解放アクノリッジ出力が有効です。
K1	TDI	H-UDI テストデータ入力。内部でプルアップされています。
K2	TCK	H-UDI クロック入力。外部でプルアップしてください。
K3	Vss	電源 (0V)
K4	Vcc(I/O)	入出力用電源 (3.3V)
K14	AUDATA3	AUD データバス (ビット 3) 出力
K15	AUDSYNC	AUD 同期信号出力
K16	PTE5/VCI_HWC* ²	ポート E / Voice Codec IC 用動作モード選択出力。初期状態はポート入力固定、出力ハイインピーダンスですが、内部プルアップ MOS がオン状態となります。
K17	Vss	電源 (0V)
L1	TMS	H-UDI テストモードスイッチ。内部でプルアップされています。
L2	Vss	電源 (0V)
L3	VDD	内部レギュレータからの電源出力 / 内部電源入力 (1.5V)
L4	Vcc(内部)	内部レギュレータ用電源 (3.3V)
L14	Vcc(I/O)	入出力用電源 (3.3V)
L15	Vss	電源 (0V)
L16	AUDATA1	AUD データバス (ビット 1) 出力
L17	AUDATA2	AUD データバス (ビット 2) 出力
M1	NC	予約。どこにも接続しないでください。
M2	BOOT_E	ブートモードイネーブル制御入力。内部でプルアップされています。
M3	CS0	チップセレクト 0 出力
M4	CS3	チップセレクト 3 出力
M14	PTB7/BS	ポート B / バスサイクル開始信号出力。初期状態はバスサイクル開始信号出力が有効です。
M15	RD	リードストロブ出力
M16	AUDCK	AUD クロック出力
M17	AUDATA0	AUD データバス (ビット 0) 出力
N1	CS4	チップセレクト 4 出力
N2	D0	データバス (ビット 0)
N3	D1	データバス (ビット 1)
N4	D2	データバス (ビット 2)
N14	Vcc(内部)	内部レギュレータ用電源 (3.3V)
N15	VDD	内部レギュレータからの電源出力 / 内部電源入力 (1.5V)

1. 概要

端子番号	端子名	機能
N16	Vss	電源 (0V)
N17	PTB6/ $\overline{\text{REFOUT}}$	ポート B / バス解放要求出力。初期状態はバス解放要求出力が有効です。
P1	D3	データバス (ビット 3)
P2	D4	データバス (ビット 4)
P3	Vcc(I/O)	入出力用電源 (3.3V)
P4	D5	データバス (ビット 5)
P5	D13	データバス (ビット 13)
P6	Vcc(I/O)	入出力用電源 (3.3V)
P7	A3	アドレスバス (ビット 3)
P8	Vcc(I/O)	入出力用電源 (3.3V)
P9	A8	アドレスバス (ビット 8)
P10	Vcc(I/O)	入出力用電源 (3.3V)
P11	A15	アドレスバス (ビット 15)
P12	Vcc(I/O)	入出力用電源 (3.3V)
P13	PTD3/A21	ポート D / アドレスバス (ビット 21)。初期状態は A21 出力が有効です。
P14	PTB4/ $\overline{\text{RAS}}$	ポート B / $\overline{\text{RAS}}$ (SDRAM) 出力。初期状態は $\overline{\text{RAS}}$ 出力が有効です。
P15	Vcc(I/O)	入出力用電源 (3.3V)
P16	Vss	電源 (0V)
P17	$\overline{\text{WAIT}}$	ハードウェアウェイト要求。内部でプルアップされています。
R1	Vss	電源 (0V)
R2	REG_PD_MAIN	内部電源 ($V_{DD} = 1.5V$) を外部入力時は、ハイ固定してください。その他のときは、どこにも接続しないでください。
R3	D10	データバス (ビット 10)
R4	Vcc(I/O)	入出力用電源 (3.3V)
R5	D14	データバス (ビット 14)
R6	Vss	電源 (0V)
R7	A4	アドレスバス (ビット 4)
R8	Vss	電源 (0V)
R9	A11	アドレスバス (ビット 11)
R10	REG_PD_BGR	内部電源 ($V_{DD} = 1.5V$) を外部入力時は、ハイ固定してください。その他のときは、どこにも接続しないでください。
R11	A14	アドレスバス (ビット 14)
R12	A18	アドレスバス (ビット 18)
R13	PTD2/A20	ポート D / アドレスバス (ビット 20)。初期状態は A20 出力が有効です。
R14	Vcc(I/O)	入出力用電源 (3.3V)
R15	PTB2/ $\overline{\text{CKE}}$	ポート B / クロックイネーブル (SDRAM) 出力。初期状態は $\overline{\text{CKE}}$ 出力が有効です。

1. 概要

端子番号	端子名	機能
R16	WE0/DQM0	D7～D0 セレクト信号 / DQM (SDRAM) 出力
R17	WE1/DQM1	D15～D8 セレクト信号 / DQM (SDRAM) 出力
T1	D6	データバス (ビット 6)
T2	D7	データバス (ビット 7)
T3	NC	予約。どこにも接続しないでください。
T4	D11	データバス (ビット 11)
T5	D15	データバス (ビット 15)
T6	A1	アドレスバス (ビット 1)
T7	A5	アドレスバス (ビット 5)
T8	REG_PD_VREF	内部電源 (V _{DD} = 1.5V) を外部入力時は、ハイ固定してください。その他のときは、どこにも接続しないでください。
T9	A9	アドレスバス (ビット 9)
T10	NC	予約。どこにも接続しないでください。
T11	A13	アドレスバス (ビット 13)
T12	A17	アドレスバス (ビット 17)
T13	PTD1/A19	ポート D / アドレスバス (ビット 19)。初期状態は A19 出力が有効です。
T14	PTD5/A23	ポート D / アドレスバス (ビット 23)。初期状態は A23 出力が有効です。
T15	PTB5/RD/W \bar{R}	ポート B / リード / ライト出力。初期状態はリード / ライト出力が有効です。
T16	VBB	テスト端子です。V _{SS} に接続してください。
T17	NC	予約。どこにも接続しないでください。
U1	D8	データバス (ビット 8)
U2	D9	データバス (ビット 9)
U3	V _{SS}	電源 (0V)
U4	D12	データバス (ビット 12)
U5	PTD0/A0	ポート D / アドレスバス (ビット 0)。初期状態は A0 出力が有効です。
U6	A2	アドレスバス (ビット 2)
U7	A6	アドレスバス (ビット 6)
U8	A7	アドレスバス (ビット 7)
U9	A10	アドレスバス (ビット 10)
U10	A12	アドレスバス (ビット 12)
U11	V _{SS}	電源 (0V)
U12	A16	アドレスバス (ビット 16)
U13	V _{SS}	電源 (0V)
U14	PTD4/A22	ポート D / アドレスバス (ビット 22)。初期状態は A22 出力が有効です。
U15	V _{SS}	電源 (0V)

1. 概要

端子番号	端子名	機能
U16	PTB3/CAS	ポート B / CAS (SDRAM) 出力。初期状態は CAS 出力が有効です。
U17	VBBENB	テスト端子です。Vss に接続してください。

- 【注】
- *1 RF IC と接続される端子です。具体的接続方法は「第 21 章 ブルートゥースインタフェース (BT)」を参照してください。
 - *2 Voice Codec IC と接続される端子です。具体的接続方法は「第 21 章 ブルートゥースインタフェース (BT)」を参照してください。
 - *3 USB ピンマルチプレクスコントローラの EXCPG 制御レジスタ (EXCPGCR) の USBCLKSEL ビットを 0 にすることで RDI_REFCLK_IN から入力したクロックが USB 用クロックとして有効となります。ただし内蔵通信回路を使用した場合、クロック精度は USB 規格の値を保証できません。本 LSI の USB 機能を使用する場合は、上記の方法よりも、UCLK 端子から USB の規格に適合した精度を持つ 48MHz のクロックを入力することを推奨します。
 - *4 本 LSI 内部には、RDI_REFCLK_IN から入力されるクロックを使ったリセットホールドカウンタが備えられているため Bluetooth 機能を使用しない場合においても、本端子には必ずクロック供給をするようにしてください。本端子に供給するクロックは、10 ~ 13MHz の範囲であれば EXTAL 端子に供給するクロックと同一のもので対応可能です。

1.4 端子の機能

各端子の機能を表 1.3 に示します。

表 1.3 SH7660 端子機能

分類	端子名	入出力	名称	機能
電源	VDD* ¹	出力 / 入力	電源	LSI 内蔵レギュレータの出力または内部電源の入力です。 TEST_REG 端子で選択します。
	Vcc(内部)	入力	電源	内部レギュレータ用電源です。すべての Vcc 端子をシステムの電源に接続してください。開放端子があると動作しません。
	Vcc(I/O)	入力	電源	入出力端子用電源です。すべての Vcc 端子をシステムの電源に接続してください。開放端子があると動作しません。
	Vss	入力	グラウンド	グラウンド端子です。すべての Vss 端子をシステム電源 (0V) に接続してください。開放端子があると動作しません。
	TEST_REG	入力	電源選択	LSI 内蔵レギュレータ選択用端子です。ハイの場合、内蔵レギュレータ選択で、ローの場合内部電源 (V _{DD} = 1.5V) 端子が外部入力となります。この端子は動作中には変化させないでください。
	REG_PD_MAIN	入力 / NC	電源制御	LSI 内蔵レギュレータ制御用端子です。内部電源 (V _{DD} = 1.5V) を外部入力時は、ハイ固定してください。その他の時は、どこにも接続しないでください。
	REG_PD_BGR	入力 / NC	電源制御	LSI 内蔵レギュレータ制御用端子です。内部電源 (V _{DD} = 1.5V) を外部入力時は、ハイ固定してください。その他の時は、どこにも接続しないでください。
REG_PD_VREF	入力 / NC	電源制御	LSI 内蔵レギュレータ制御用端子です。内部電源 (V _{DD} = 1.5V) を外部入力時は、ハイ固定してください。その他の時は、どこにも接続しないでください。	
クロック	Vcc(PLL)	入力	PLL1/PLL2 用電源	内蔵 PLL1/PLL2 発振器用の電源端子です。
	V _{DD} (PLL)* ¹	出力 / 入力	電源	内蔵 PLL1/PLL2 発振器用の LSI 内蔵レギュレータの出力または内部電源の入力です。TEST_REG 端子で選択します。
	Vss(PLL)	入力	PLL1/PLL2 共用グラウンド	内蔵 PLL1/PLL2 発振器共通のグラウンド端子です。
	EXTAL	入力	クリスタル入力 (クロック入力)	水晶発振子を接続します。また、EXTAL 端子は外部クロックを入力することができます。水晶発振子を接続する場合、および外部クロック入力の接続例については、「第 11 章 クロックパルス発振器 (CPG)」を参照してください。
	XTAL	出力	クリスタル出力	水晶発振子を接続します。水晶発振子を接続する場合、および外部クロック入力の接続例については、「第 11 章 クロックパルス発振器 (CPG)」を参照してください。
	CKIO	入出力	クロック入出力	外部デバイスにシステムクロックを供給します。または外部クロックを入力として使用してください。

1. 概要

分類	端子名	入出力	名称	機能
動作モード コントロール	MD5、 MD2、 MD1	入力	モード制御	動作モードを設定します。これらの端子は動作中には変化させないでください。 MD2、MD1はクロックモード設定用、MD5はエンディアン設定用です。
システム 制御	RESETP	入力	パワーオン リセット	この端子がローレベルになると、パワーオンリセット状態になります。
	BOOT_E	入力	ブート制御	ブートモードイネーブル制御入力。パワーオンリセット時、本端子はローの場合は、リセット終了後、ブートモードになり、ブートローダが起動します。ハイの場合は、通常のリセットシーケンスが起動します。
	BREQ	入力	バス権要求	外部デバイスがバス権の解放を要求するときにローレベルにします。
	BACK	出力	バス権要求 アクノリッジ	バス権を外部に解放したことを示します。BREQ信号を出力したデバイスは、BACK信号を受けて、バス権を獲得したことを知ることができます。
割り込み	NMI	入力	ノンマスクابل 割り込み要求	ノンマスクابل割り込み要求端子です。使用しない場合はハイレベルに固定してください。
	IRQ4 ~ IRQ2、 IRQ0	入力	割り込み要求 4~2、0	マスク可能な割り込み要求端子です。 レベル入力、エッジ入力の選択が可能です。エッジ入力の場合、立ち上がり、立ち下がり、両エッジの選択が可能です。
	IRQOUT	出力	割り込み要求通知	割り込み要因が発生したことを外部デバイスに通知できます。
アドレスバス	A23 ~ A0	出力	アドレスバス	アドレスを出力します。
データバス	D15 ~ D0	入出力	データバス	16ビットの双方向バスです。
バス制御	$\overline{CS0}$ 、 $\overline{CS3}$ 、 $\overline{CS4}$	出力	チップセレクト 0、3、4	外部メモリまたはデバイスのためのチップセレクト信号です。
	\overline{RD}	出力	読み出し	外部のデバイスから読み出すことを示します。
	$\overline{RD}/\overline{WR}$	出力	リード/ライト	リード/ライト信号端子です。
	\overline{BS}	出力	バス開始	バスサイクル開始信号端子です。
	$\overline{WE1}$	出力	上位側書き込み	外部メモリまたはデバイスのデータのビット15~8に書き込みすることを示します。
	$\overline{WE0}$	出力	下位側書き込み	外部メモリまたはデバイスのデータのビット7~0に書き込みすることを示します。
	CAS	出力	データ イネーブル	SDRAMのCAS信号です。
	RAS	出力	データ イネーブル	SDRAMのRAS信号です。
	CKE	出力	クロック イネーブル	SDRAMのクロックイネーブル信号端子です。
	DQM1	出力	DQマスク	SDRAMのデータビット15~8を選択することを示します。
DQM0	出力	DQマスク	SDRAMのデータビット7~0を選択することを示します。	

分類	端子名	入出力	名称	機能
バス制御	REFOUT	出力	バス解放要求	バス権解放中にリフレッシュ要求が発生したことを示します。
	WAIT	入力	ウェイト	外部空間をアクセスするときのバスサイクルにウェイトサイクルを挿入させる入力です。
ダイレクト	DREQ	入力	DMA 転送要求	外部からの DMA 転送要求の入力端子です。
メモリアクセス コントローラ (DMAC)	DACK	出力	DMA 転送要求 受け付け	外部からの DMA 転送要求に対する要求受け付け出力端子です。
	TEND	出力	DMA 転送終了	DMA 転送が終了したことを示します。
FIFO 付き シリアル インタフェース (SIOF)	SIOF_TXD (MOSI)	出力	送信データ	送信データ用の端子です。
	SIOF_RXD (MISO)	入力	受信データ	受信データ用の端子です。
	SIOF_SCK (SCK)	入出力	シリアル クロック	クロック入出力端子です。
	SIOF_MCLK	入力	マスタ クロック	マスタクロック入力端子です。
	SIOF_SYNC (SIOF_SS0)	入出力	フレーム同期信号 (スレーブデバイス 0 選択)	フレーム同期信号端子です。 (SPI モード時、スレーブデバイス 0 を選択する端子です。)
	SIOF_SST1	出力	スレーブデバイス 1 選択	SPI モード時、スレーブデバイス 1 を選択する端子 です。
	SIOF_SS2	出力	スレーブデバイス 2 選択	SPI モード時、スレーブデバイス 2 を選択する端子 です。
FIFO 内蔵 シリアル コミュニケーション インタフェース (SCIF0、1)	SCIF0_TXD、 SCIF1_TXD	出力	送信データ	送信データ用の端子です。
	SCIF0_RXD、 SCIF1_RXD	入力	受信データ	受信データ用の端子です。
	SCIF0_SCK、 SCIF1_SCK	入出力	シリアル クロック	クロック入出力端子です。
	SCIF0_RTS SCIF1_RTS	出力	送信要求	モデムコントロール端子です。
	SCIF0_CTS SCIF1_CTS	入力	送信可	モデムコントロール端子です。

1. 概要

分類	端子名	入出力	名称	機能
USB	UCLK	入力	USB クロック	USB 用クロック入力端子 (48MHz 入力)
	USB_PWR_EN/ USB_PULLUP	出力	USB 電圧制御	USB 電源投入許可制御 (USBH 使用時) USB プルアップ制御 (USBF 使用時)
	USB_OVR_CRNT/ USB_VBUS	入力	USB 電源検知	過電流検出 (USBH 使用時) USB ケーブル接続モニタ (USBF 使用時)
	USB_N	入出力	D-入出力	USB 内蔵トランシーバ D-
	USB_P	入出力	D+入出力	USB 内蔵トランシーバ D+
	AVcc(USB)	入力	USB トランシーバ 用アナログ電源	USB トランシーバ用のアナログ電源端子です。
	AVss(USB)	入力	USB トランシーバ 用アナロググランド	USB トランシーバ用のアナロググランド端子です。 システムの電源 (Vss) に接続してください。
	Vcc(DLL)	入力	USB クロック逡倍 回路用 3.3V 電源	USB 用 48MHz クロック逡倍回路用電源端子です。 システムの電源 (Vcc) に接続してください。
	Vdd(DLL)* ¹	出力 / 入力	電源	USB 用 48MHz クロック逡倍回路用の LSI 内蔵レギュレータの出力または内部電源の入力です。 TEST_REG 端子で選択します。
	Vss(DLL)	入力	USB クロック逡倍 回路用グランド	USB 用 48MHz クロック逡倍回路用グランド端子です。 システムの電源 (Vss) に接続してください。
ブルートゥース インタフェース (BT)	RDI_TXTRDATA* ²	入出力	送受信データバス	RF IC とのデータの送受信のためのバスです。
	RDI_RXBDW_ OUT* ²	出力	パケット制御	BT のパケット処理状態を RF IC に伝える信号です。
	RDI_REFCLK_IN* ²	入力	BT クロック入力	ブルートゥースインタフェースの動作クロックを入力してください。
	RDI_CTRL2* ²	出力	RF IC 制御 2	クラス 1 対応時に外部パワーアンプのパワーコントロールに使用します。
	RDI_CTRL3* ²	出力	RF IC 制御 3	RF IC 内の発振器をイネーブルにするストロブ信号です。
	RDI_CTRL4* ²	出力	RF IC 制御 4	RF IC 用リセット制御および低消費電力制御出力です。
	RCI_SPI_CLK* ²	出力	SPI クロック	シリアルインタフェースクロック
	RCI_SPI_TXRX* ²	入出力	SPI データ	シリアルインタフェースデータ
	RCI_SPI_ENB* ²	出力	SPI イネーブル	シリアルインタフェースのイネーブル信号
	VCI_SCO_CLK_ OUT* ³	出力	VCI クロック	Voice Codec IC にクロックを供給します。
	VCI_SCO_SYNC_ OUT* ³	出力	VCI 同期	Voice Codec IC にフレーム同期信号を供給します。
	VCI_SCO_TX* ³	入力	SCO 送信 データ	送信する SCO データを Voice Codec IC から入力します。

分類	端子名	入出力	名称	機能
Bluetooth インタフェース (BT)	VCI_SCO_RX* ³	出力	SCO 受信 データ	受信した SCO データを Voice Codec IC へ出力します。
	VCI_HWC* ³	出力	VCI モード選択	Voice Codec IC (STLC7550) の動作モードを選択します。
	VCI_CODEEC_PW RDWN* ³	出力	VCI パワーダウン	Voice Codec IC のパワーダウン制御を行います。
	RTCSEL0	入力	テスト端子	ロー固定してください。
	RREF	入力	テスト端子	プルアップしてください。
	EXTAL2 XTAL2	入力 出力	Low Power クロック水晶発振子 接続	Low Power クロックのため、水晶発振子を接続する端子です。 RTCSEL0 がハイのとき、これらの端子は非活性状態です。
	Vcc_28	入力	電源	RF IC 接続端子用電源です。RF IC と同じ電圧の電源を供給してください。
	Vss_28	入力	グランド	RF IC 接続端子用グランド端子です。
D/A コンバータ (DAC)	DA1、DA0	出力	アナログ 出力端子	D/A コンバータのアナログ出力端子です。
	AVcc(DAC)	入力	D/A コンバータ用 アナログ電源	D/A コンバータの電源端子です。D/A コンバータを使用しない場合はポート用電源 (Vcc) に接続してください。
	AVss(DAC)	入力	D/A コンバータ用 アナロググランド	D/A コンバータのグランド端子です。システムの電源 (Vss) に接続してください。
I/O ポート	PTA6 ~ PTA0	入出力	汎用ポート	7 ビットの汎用入出力ポート端子です。
	PTB7 ~ PTB0	入出力	汎用ポート	8 ビットの汎用入出力ポート端子です。
	PTD5 ~ PTD0	入出力	汎用ポート	6 ビットの汎用入出力ポート端子です。
	PTE6 ~ PTE0	入出力	汎用ポート	7 ビットの汎用入出力ポート端子です。
	PTG4 ~ PTG0	入出力	汎用ポート	5 ビットの汎用入出力ポート端子です。
	PTH4 ~ PTH0	入出力	汎用ポート	5 ビットの汎用入出力ポート端子です。
ユーザデバッグ インタフェース (H-UDI)	TCK	入力	テスト クロック	テストクロック入力端子です。
	TMS	入力	テストモード セレクト	テストモードセレクト信号入力端子です。
	TDI	入力	テストデータ 入力	インストラクションとデータのシリアル入力端子です。
	TDO	出力	テストデータ 出力	インストラクションとデータのシリアル出力端子です。
	TRST	入力	テスト リセット	初期化信号入力端子です。

1. 概要

分類	端子名	入出力	名称	機能
アドバンスト ユーザデバッグ (AUD)	AUDATA3 ~ AUDATA0	出力	AUD データ	分岐先アドレス出力端子です。
	AUDCK	出力	AUD クロック	同期クロック出力端子です。
	AUDSYNC	出力	AUD 同期信号	データ先頭位置認識信号出力端子です。
E10A インタフェース	ASEBRKAK	出力	ブレークモードアクノリ ッジ	E10A エミュレータがブレークモードに入ったこ とを示します。 E10A との接続については、「SH7660 E10A エミ ュレータ」を参照してください。
	ASEMD0	入力	ASE モード	ASE モードを設定します。

- 【注】 *1 V_{DD} と V_{DD} (PLL)、V_{DD} (DLL)は直接短絡せず、できる限り分離して、ノイズ対策を施してください。
- *2 RF IC と接続される端子です。具体的接続方法は「第 21 章 ブルートゥースインタフェース (BT)」を参照してください。
- *3 Voice Codec IC と接続される端子です。具体的接続方法は「第 21 章 ブルートゥースインタフェース (BT)」を参照してください。

2. CPU

2.1 処理状態と処理モード

2.1.1 処理状態

CPU の処理状態には、リセット状態、例外処理状態、プログラム実行状態、および低消費電力状態の 4 種類があります。

(1) リセット状態

CPU がリセットされている状態です。リセットの種類には、パワーオンリセットとマニュアルリセットの 2 種類があります。リセットの詳細については、「第 4 章 例外処理」を参照してください。

パワーオンリセットでは、本 LSI 内部の全モジュールのレジスタや内部状態が初期化されます。マニュアルリセットでは、バススタートコントローラ (BSC) などの一部のモジュールでレジスタの値が保持されます。詳細については、「第 27 章 レジスタ一覧」を参照してください。

CPU の内部状態やレジスタは、パワーオンリセットでもマニュアルリセットでも初期化されます。初期化完了後は、H'A000 0000 に分岐してユーザが作成したリセット処理プログラムに制御が渡され、実行を開始します。

(2) 例外処理状態

一般例外や割り込みの例外処理要因によって、CPU が処理状態の流れを変える過渡的な状態です。プログラムカウンタ (PC) を退避プログラムカウンタ (SPC) に、ステータスレジスタ (SR) を退避ステータスレジスタ (SSR) に退避します。ベクタベースレジスタ (VBR) の内容にベクタオフセットを加算したアドレスへ分岐し、ユーザが作成した例外処理プログラムに制御が渡され、実行を開始します。

例外処理状態の詳細については、「第 4 章 例外処理」を参照してください。

(3) プログラム実行状態

CPU が順次プログラムを実行している状態です。

(4) 低消費電力状態

CPU の動作が停止し、消費電力が抑えられている状態です。SLEEP 命令で低消費電力状態になります。低消費電力状態の詳細については、「第 13 章 低消費電力モード」を参照してください。

処理状態間の遷移を図 2.1 に示します。

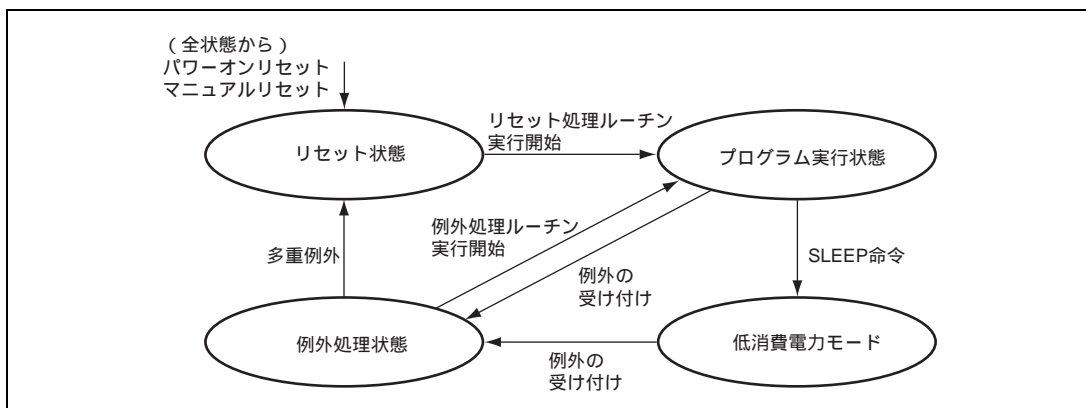


図 2.1 処理の状態遷移図

2.1.2 処理モード（ユーザモード / 特権モード）

CPUの処理モードには、ユーザモードと特権モードの2つがあります。ステータスレジスタ（SR）の処理モードビット（MD）で処理モードが決定され、MDビットが0のときユーザモードになり、1のとき特権モードになります。リセット状態や例外処理状態に遷移することにより、CPUは自動的に特権モードに遷移します。CPUが特権モードにあるとき、レジスタやアドレス空間のすべてのリソースをアクセスすることができます。処理モードごとにCPUがアクセス可能なレジスタやアドレス空間の違いについては、「2.2 メモリマップ」、「2.3 レジスタの説明」を参照してください。

ステータスレジスタ（SR）のMDビットに0を書き込むことで、CPUの処理モードは、ユーザモードになります。CPUがユーザモードにあるとき、SRを含む一部のレジスタやアドレス空間の一部がユーザプログラムからアクセスできなくなり、またシステムを制御する命令が実行できなくなります。この機構により、ユーザプログラムからシステムのリソースを保護することができます。処理モードをユーザモードから特権モードに変更するには、例外処理状態に遷移する必要があります。^{*1*2}

【注】 *1 ユーザモードから特権モードのサービスルーチンを呼び出す目的で、無条件トラップ命令（TRAPA）がサポートされています。

*2 ユーザモードから特権モードへ遷移する際、SRおよびプログラムカウンタ（PC）が退避されます。これらを復帰することにより、ユーザモードのプログラムを再開することができます。この例外処理プログラムからの復帰用にRTE命令がサポートされています。

2.2 メモリマップ

本 LSI が搭載している CPU は、前述のように処理モードによって見えるアドレス空間が異なり、CPU が扱うアドレス空間が必ずしも物理的なアドレス空間と一致しているとは限りません。そこで、CPU の出力するアドレス空間を論理アドレス空間、物理的なアドレス空間を物理アドレス空間と呼んで区別しています。さらに、物理アドレス空間のうち、本 LSI の外部にアクセス可能な空間を外部アドレス空間と呼びます。以後、これらの名称を使って説明します。

2.2.1 論理アドレス空間

CPU の出力するアドレス空間を論理アドレス空間といいます。本 LSI の CPU は、32 ビットの論理アドレス空間をサポートし、4G バイトの論理アドレス空間を用いてシステムのリソースにアクセスします。ユーザのプログラムやデータは、この論理アドレス空間上でアクセスされます。CPU から出力されるアドレスは、ハードウェアとしては L バス上（「第 1 章 概要」の「図 1.1 SH7660 の構成」を参照）に流れているアドレスになります。論理アドレス空間は、図 2.2 に示すとおり、いくつかの領域に分けて管理されています。

特権モードでは、P0 領域から P4 領域の 4G バイトの空間にアクセスすることが可能です。

ユーザモードでは、U0 領域の 2G バイトの空間にアクセス可能です。また、SR レジスタの DSP ビットが 1 の場合は、Uxy 領域の 16M バイトの空間もアクセス可能になります。ユーザモードで U0 および Uxy 領域以外にアクセスした場合は、アドレスエラーとなります。

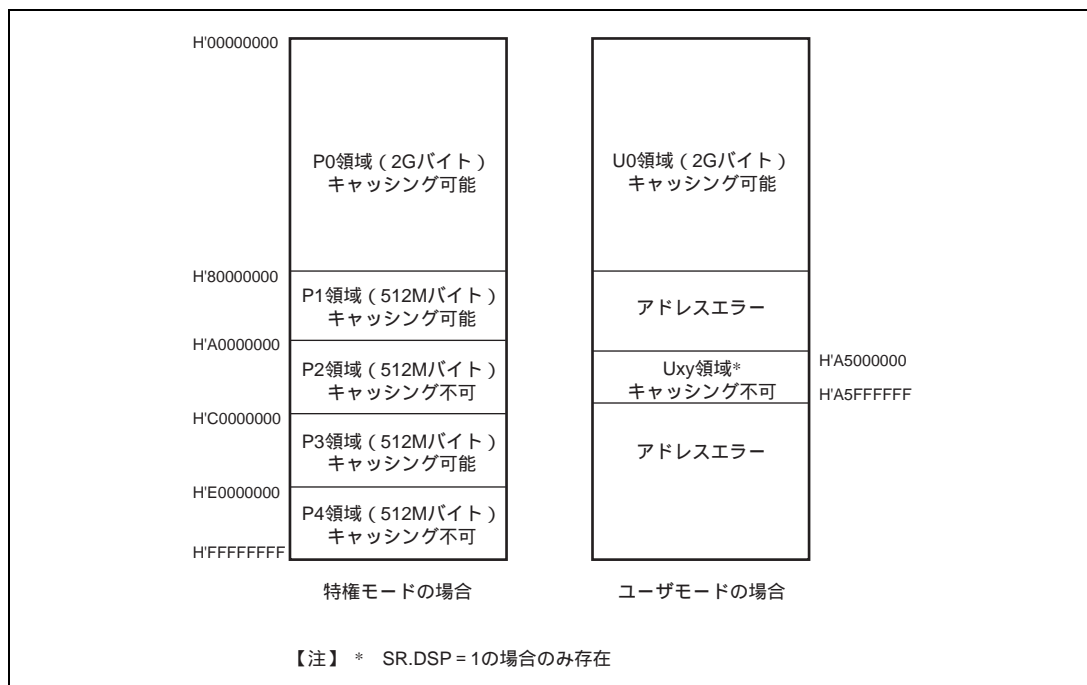


図 2.2 論理アドレス空間

2. CPU

(1) P0/U0 領域

CPU が特権モードにあるとき P0 領域、ユーザモードにあるとき U0 領域と呼びます。P0 および U0 領域は、キャッシュを用いたアクセスが可能な領域です*。

(2) P1 領域

P1 領域は、キャッシングの対象となる特権領域として定義されています*。通常、この領域にはオペレーティングシステム (OS) のカーネル、および例外処理のハンドラなど、特権モードで高速に動作するプログラムを配置します。

(3) P2 領域

P2 領域は、キャッシングの対象とならない特権領域として定義されています。リセット状態から呼び出されるリセット処理プログラムは、P2 領域の先頭 (H'A0000000) から記述します。通常、この領域にはシステムの初期設定ルーチンなど、OS の起動に必要なプログラムを配置します。また、本 LSI の一部の内蔵 I/O をアクセスする場合には、P2 領域にプログラムを配置しておく必要があります。

(4) P3 領域

P3 領域は、キャッシングの対象となる特権領域として定義されています*。

(5) P4 領域

P4 領域は、制御空間で、キャッシングの対象にならず、特権モードのみでアクセス可能です。P4 領域の詳細を図 2.3 に示します。本 LSI の内蔵 I/O の一部は、この空間に配置されています。この空間に配置されている具体的な内蔵 I/O については、「第 27 章 レジスタ一覧」を参照してください。

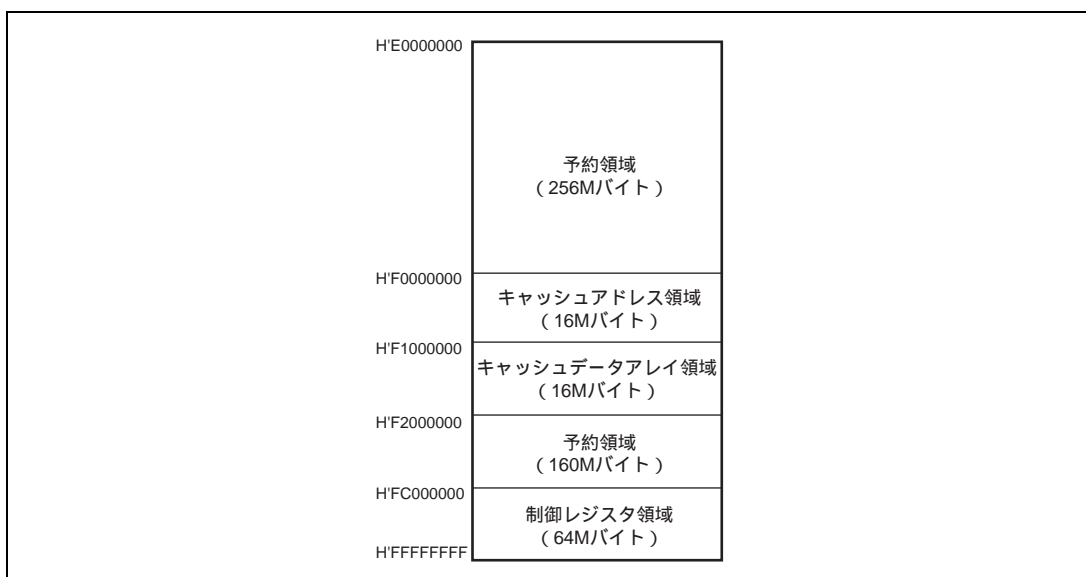


図 2.3 P4 領域

(6) Uxy 領域

Uxy 領域は、ユーザモードにおいて、SR レジスタの DSP ビットが 1 のときに使用可能となる、本 LSI の内蔵メモリにマッピングされる領域です。ユーザモードにおいて、DSP ビットが 0 のときにこの領域にアクセスするとアドレスエラーとなります。この領域はキャッシュを用いたアクセスは行えません。Uxy 領域の詳細については「第 6 章 X/Y メモリ」および「第 7 章 U メモリ」を参照してください。DSP ビットについては「第 3 章 DSP 演算ユニット」を参照してください。

【注】 * キャッシュを用いるか否かは、キャッシュコントロールレジスタ (CCR1) の CE ビットに従います。

2.2.2 物理アドレス空間

CPU から出力されたアドレスをアドレス変換した後の物理的なアドレス空間を物理アドレス空間と呼びます。ハードウェアとしては I バス上に流れているアドレスになります。本 CPU は論理アドレスと同様に 32 ビットの物理アドレス空間をサポートしていますが、図 2.4 に示すように、32 ビットのうち上位 3 ビットがマスクされてシャドウとして扱われ、29 ビットが実際の物理アドレス空間へのアクセスに使用されて 0.5G バイトの物理メモリをアクセスできます。したがって、上位 3 ビットには通常、000 にマスクされたアドレスがそのまま物理アドレスになります*。物理アドレス空間の詳細については「第 9 章 バスステートコントローラ (BSC)」を参照してください。

DMAC 等の CPU 以外のバスマスタは、I バスに直接接続されているため、論理アドレス空間は扱わずに直接物理アドレス空間を扱います。

【注】 * 論理アドレスで上位 3 ビットが 111 になっていた場合のみ (特権モードの P4 領域に相当)、内蔵 I/O をアクセスするために、I バスの上位 3 ビットにも 111 がそのまま出て行きます。

H'00000000	29ビット (512Mバイト) 物理アドレス空間
H'20000000	29ビット物理アドレス空間 のシャドウ空間
H'40000000	29ビット物理アドレス空間 のシャドウ空間
H'60000000	29ビット物理アドレス空間 のシャドウ空間
H'80000000	29ビット物理アドレス空間 のシャドウ空間
H'A0000000	29ビット物理アドレス空間 のシャドウ空間
H'C0000000	29ビット物理アドレス空間 のシャドウ空間
H'E0000000	内蔵I/O領域
H'FFFFFFFF	

図 2.4 物理アドレス空間

2.2.3 外部アドレス空間

物理アドレス空間は図 2.5 に示すように、エリアと呼ばれる 8 つの領域に分かれています。このうち、エリア 1 は内蔵 I/O 空間として使用され、本 LSI の内蔵 I/O の多くはこの領域にマッピングされています*。本 LSI ではエリア 2 およびエリア 5~エリア 7 は予約領域です。残りの 3 つのエリアが外部アドレス空間になります。外部アドレス空間は、各エリアごとに異なった種類のメモリを接続することができます（詳細については「第 9 章 パスステートコントローラ（BSC）」を参照してください）。

CPU から見た場合、通常は論理アドレスの上位 3 ビットがマスクされてシャドウとして扱われていますので、例えば特権モードの場合、論理アドレス空間の P0 領域の H'00000100 番地、P1 領域の H'80000100 番地、P2 領域の H'A0000100 番地、および P3 領域の H'C0000100 番地は、すべて同一の外部アドレス空間のエリア 0 の H'00000100 番地にマッピングされます。ただし、P4 領域はマッピングの対象にならず、P4 領域をアクセスしても外部アドレス空間へのアクセスとはなりません。

外部メモリが接続できる 3 つのエリア（エリア 0、エリア 3、エリア 4）は、それぞれ 64M バイトの外部アドレス空間ですが、本 LSI はアドレス端子を 24 ピン備えていますので、それぞれのエリアには最大 16M バイトのメモリが実装可能です。図 2.5 にはエリア 0 に対応する実装空間を示しています。

【注】 * 外部アドレス空間のエリア 1 にマッピングされている内蔵 I/O をアクセスする場合には、キャッシングの対象でない論理アドレス空間の P2 領域からアクセスしてください。

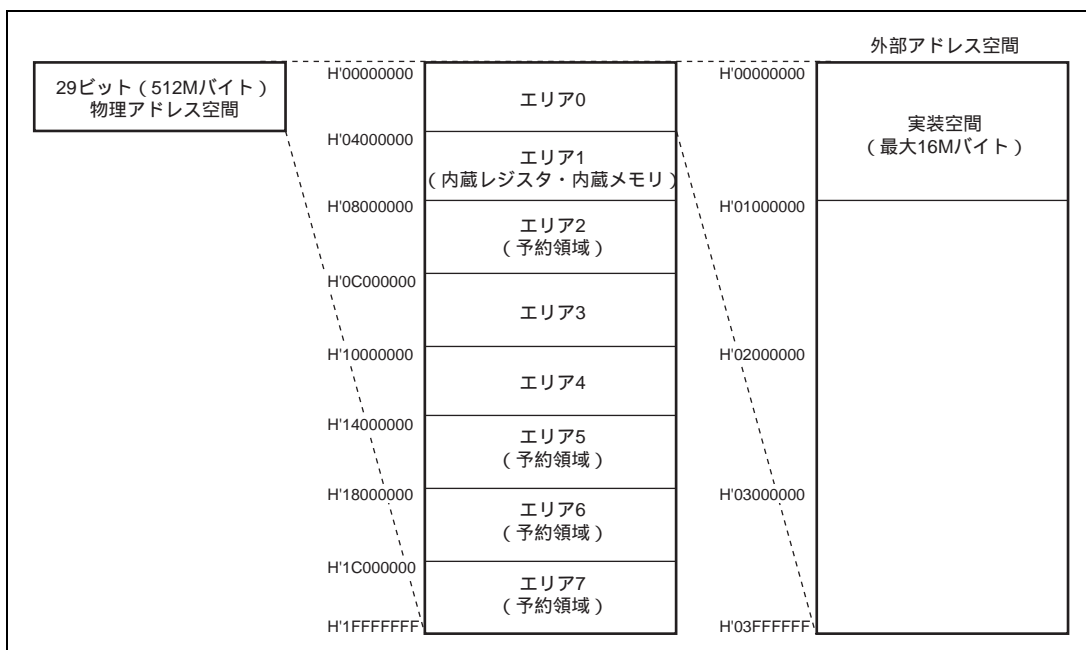


図 2.5 外部アドレス空間と実装空間（エリア 0 の場合）

2.3 レジスタの説明

本 LSI の CPU は、32 ビットのレジスタを 33 本保有しています。これらは、汎用レジスタ (24 本)、コントロールレジスタ (5 本)、システムレジスタ (3 本)、およびプログラムカウンタ (1 本) に分類されます。

(1) 汎用レジスタ

汎用レジスタには、R0_BANK0~R7_BANK0、R0_BANK1~R7_BANK1、および R8~R15 の 24 本のレジスタがあります。R0~R7 はバンク構成になっており、どちらのバンクが汎用レジスタとして使用されるかは、処理モード、およびステータスレジスタ (SR) のレジスタバンクビット (RB) により決定されます。

(2) システムレジスタ

システムレジスタには、積和レジスタ (MACH、MACL)、およびプロシージャレジスタ (PR) があり、処理モードによらずアクセスできます。

(3) プログラムカウンタ

プログラムカウンタ (PC) は、「現在実行中の命令アドレス+4」の値を保持します。

(4) コントロールレジスタ

コントロールレジスタには、ステータスレジスタ (SR)、グローバルベースレジスタ (GBR)、退避ステータスレジスタ (SSR)、退避プログラムカウンタ (SPC)、およびベクタベースレジスタ (VBR) があります。

このうち、GBR だけはユーザモードからでもアクセスできますが、他のコントロールレジスタは特権モードでのみアクセスできます。

リセット後のレジスタの値を表 2.1 に示します。また、処理モード別のレジスタ構成を図 2.6 に示します。

表 2.1 レジスタの初期値

区分	レジスタ	初期値*
汎用レジスタ	R0_BANK0 ~ R7_BANK0 R0_BANK1 ~ R7_BANK1 R8 ~ R15	不定
システムレジスタ	MACH、MACL、PR	不定
プログラムカウンタ	PC	H'A0000000
コントロールレジスタ	SR	MD ビットは 1、RB ビットは 1、BL ビットは 1、I3~I0 は B'1111 (HF)、DSP ビットは 0、予約ビットは 0、その他は不定
	GBR、SSR、SPC	不定
	VBR	H'00000000

【注】 * パワーオンリセットおよびマニュアルリセット後の初期値です。

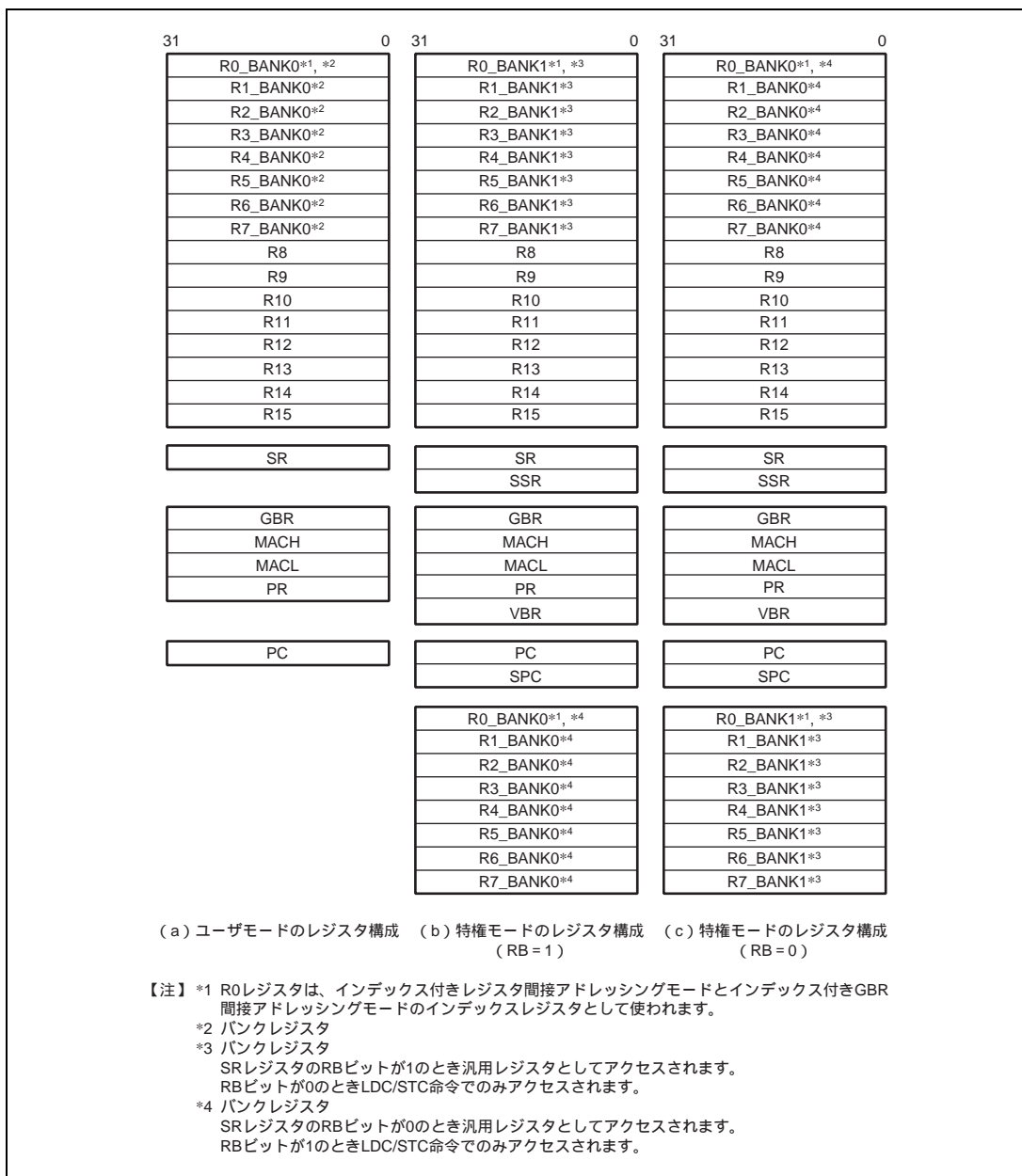


図 2.6 処理モード別のレジスタ構成

2.3.1 汎用レジスタ

汎用レジスタには、R0_BANK0～R7_BANK0、R0_BANK1～R7_BANK1、およびR8～R15の24本のレジスタがあります。図2.7に汎用レジスタの構成を示します。R0～R7はバンク構成になっており、どちらのバンクが汎用レジスタとして使用されるかは処理モード、およびステータスレジスタ（SR）のレジスタバンクビット（RB）により決定されます。汎用レジスタとして選択されているバンクはR0～R7という名前でアクセスされ、選択されていないバンクのレジスタはR0_BANK～R7_BANKという名前で、コントロールレジスタのロード命令（LDC）とストア命令（STC）を用いてアクセスします。

ユーザモードの場合は、RBビットの値によらずバンク0が選択されます。R0_BANK0～R7_BANK0とR8～R15の合計16本のレジスタが汎用レジスタR0～R15としてアクセスされます。バンク1のレジスタR0_BANK1～R7_BANK1は、アクセスできません。

例外処理状態に遷移し特権モードになると、同時にRBビットに1がセットされ、バンク1が選択されます。この場合は、バンク1のR0_BANK1～R7_BANK1とR8～R15の合計16本のレジスタを汎用レジスタR0～R15としてアクセスすることができます。例外処理状態に遷移することで自動的にバンクが切り替わるため、例外処理ルーチンでR0～R7の値を退避する必要はありません。バンク0のR0_BANK0～R7_BANK0は、LDCまたはSTC命令により、R0_BANK～R7_BANKとしてアクセスします。

特権モードでは、RBビットに0を書くことで、バンク0を汎用レジスタとして使用するよう選択することも可能です。この場合、バンク0のR0_BANK0～R7_BANK0とR8～R15の合計16本のレジスタを汎用レジスタR0～R15としてアクセスすることができます。バンク1のR0_BANK1～R7_BANK1は、LDCまたはSTC命令により、R0_BANK～R7_BANKとしてアクセスします。

汎用レジスタR0～R15は、ほぼすべての命令で等価な機能のレジスタとして使用されますが、一部の命令では、R0レジスタを暗黙的に使用したり、R0のみがソースレジスタやデスティネーションレジスタになる場合があります。

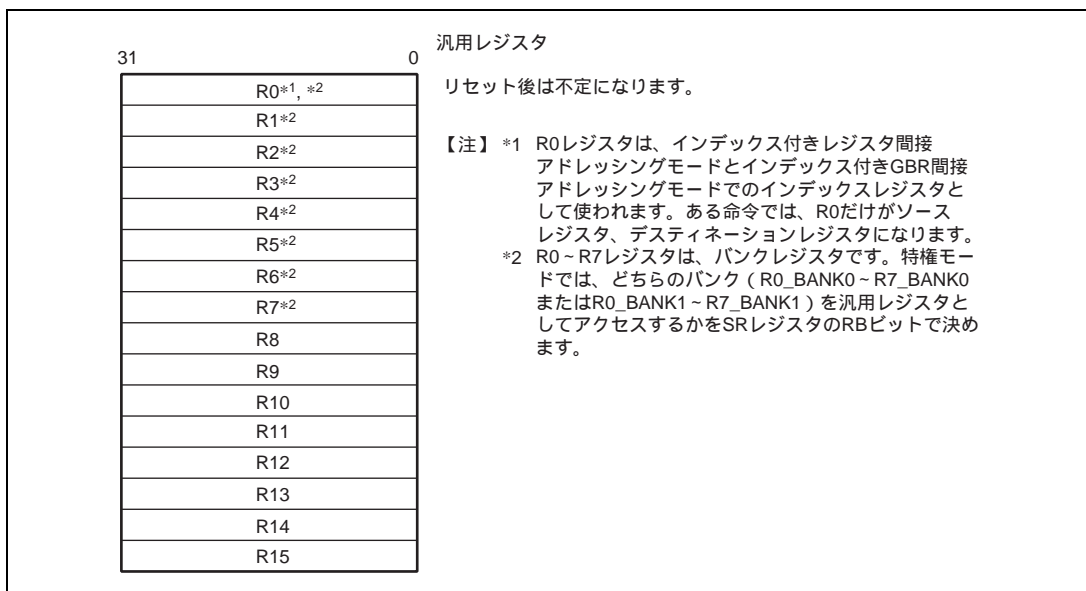


図 2.7 汎用レジスタの構成

2.3.2 システムレジスタ

システムレジスタは、LDS または STS 命令でアクセスできます。システムレジスタには、下記に示す 2 つのレジスタがあります。図 2.8 にシステムレジスタの構成を示します。

(1) 積和レジスタ

積和レジスタには、積和演算命令および乗算命令の結果を格納します。また、積和演算命令では加算値を格納する場所としても使用されます。リセット状態では不定になります。

積和レジスタは、上位 32 ビットを格納する積和上位レジスタ (MACH) と下位 32 ビットを格納する積和下位レジスタ (MACL) からなっています。

(2) プロシジャーレジスタ

プロシジャーレジスタ (PR) には、BSR、BSRF、および JSR 命令を用いたサブルーチンコールのときの戻りアドレスが格納されます。PR 中の戻りアドレスは、サブルーチンからの復帰命令 (RTS) によってプログラムカウンタ (PC) に復元されます。リセット状態では、不定になります。

2.3.3 プログラムカウンタ

プログラムカウンタ (PC) には、実行中の命令アドレス+4 の値が格納されます。図 2.8 に PC の構成を示します。PC の値を直接読み出す命令はありませんが、例外処理状態では退避プログラムカウンタ (SPC) に、サブルーチンコール時にはプロシジャーレジスタ (PR) にその値が退避されます。また、アドレッシングモードとして PC 相対アドレッシングが提供されています。

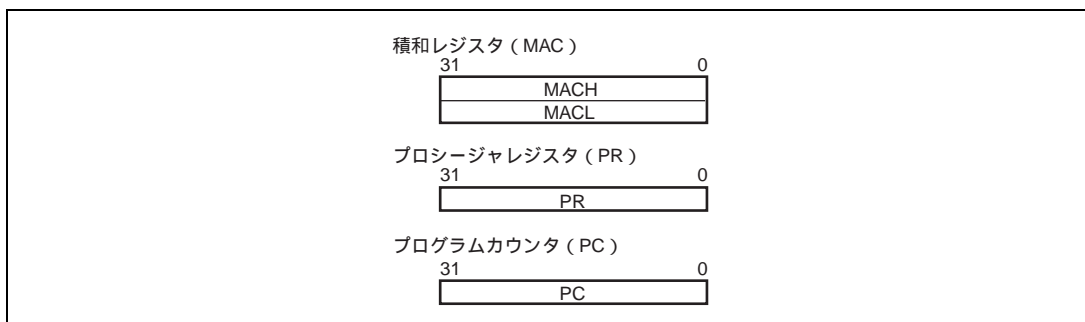


図 2.8 システムレジスタおよびプログラムカウンタの構成

2.3.4 コントロールレジスタ

コントロールレジスタは、特権モードのとき LDC または STC 命令を使ってアクセスできます。ただし、グローバルベースレジスタ (GBR) は、ユーザモードでもアクセスすることができます。コントロールレジスタには、下記に示す 5 つのレジスタがあります。

(1) ステータスレジスタ (SR)

SR には、システムの状態を表す各種情報を格納します。SR は、特権モードでのみアクセスできます。

ビット	ビット名	初期値	R/W	説明
31		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
30	MD	1	R/W	処理モード CPU の処理モードを表します。 0 : ユーザモード 1 : 特権モード リセット状態および例外処理状態になると 1 がセットされます。
29	RB	1	R/W	レジスタバンク 汎用レジスタ R0 ~ R7 は、バンクレジスタです。 特権モードで使用する汎用レジスタのバンクを指定します。 0 : R0_BANK0 ~ R7_BANK0 と R8 ~ R15 が汎用レジスタになり、 R0_BANK1 ~ R7_BANK1 は LDC または STC 命令でアクセスできます。 1 : R0_BANK1 ~ R7_BANK1 と R8 ~ R15 が汎用レジスタになり、 R0_BANK0 ~ R7_BANK0 は LDC または STC 命令でアクセスできます。 リセット状態および例外処理状態になると 1 がセットされます。

2. CPU

ビット	ビット名	初期値	R/W	説明
28	BL	1	R/W	ブロック 0: 割り込みやユーザブ레이크を受け付けます。 1: 割り込みやユーザブ레이크の受付を抑止します。 リセット状態および例外処理状態になると1がセットされます。
27~10		すべて0	R	リザーブビット* ² 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
9	M		R/W	Mビット* ¹
8	Q		R/W	Qビット* ¹ DIV0S、DIV0U、およびDIV1命令で使用します。これらの命令を実行することで、ユーザモードでも値を変更できます。リセット状態での初期値は、不定です。例外処理状態になっても、値は変化しません。
7	I3	1	R/W	割り込みマスクビット
6	I2	1	R/W	割り込み要求マスクレベルを表す4ビットデータです。
5	I1	1	R/W	割り込みが発生しても、割り込み受け付けレベルに変化しません。
4	I0	1	R/W	リセット状態でB'1111に初期化されますが、例外処理状態になっても値は変化しません。
3		0	R	リザーブビット* ²
2		0	R	読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
1	S		R/W	飽和モード* ¹ 乗算命令および積和命令での飽和モードを指定します。ユーザモードからは、SETSおよびCLRS命令を用いることで値を設定できます。 リセット状態での初期値は、不定です。例外処理状態になっても、値は変化しません。
0	T		R/W	Tビット* ¹ 比較演算命令での真または偽条件や、キャリまたはポロー付き演算命令でのキャリまたはポローの有無を表します。ユーザモードからは、SETTおよびCLRT命令を用いることで値を設定できます。リセット状態での初期値は、不定です。例外処理状態になっても、値は変化しません。

【注】 *1 M、Q、S、Tビットはユーザモードで専用命令によってセット/クリアが可能です。他のビットは、特権モードで読み出しまたは書き込みが可能です。

*2 DSPをお使いの場合は、本ビットは拡張されますので「3.2.3 CPUレジスタセット」を参照してください。

(2) 退避ステータスレジスタ (SSR)

このレジスタは、特権モードでのみアクセスできます。例外処理状態へ遷移する場合には、ステータスレジスタ (SR) の値が退避されます。リセット状態での初期値は、不定です。

(3) 退避プログラムカウンタ (SPC)

このカウンタは、特権モードでのみアクセスできます。例外処理状態へ遷移する場合には、プログラムカウン

タ (PC) の値が退避されます。リセット状態の初期値は、不定です。

(4) グローバルベースレジスタ (GBR)

GBR 間接アドレッシングのベースレジスタとして参照されます。リセット状態の初期値は、不定です。

(5) ベクタベースレジスタ (VBR)

このレジスタは、特権モードでのみアクセスできます。リセット状態以外から例外処理状態に遷移した場合は、分岐先のベースアドレスとして参照されます。詳細については、「第4章 例外処理」を参照してください。リセット状態に遷移することで、H'00000000 に初期化されます。

コントロールレジスタの構成を図 2.9 に示します。

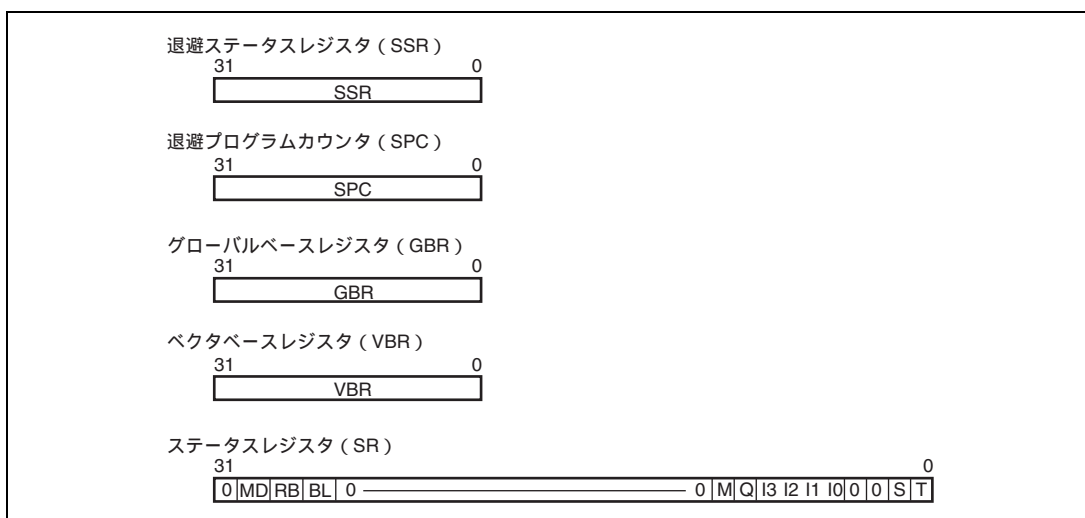
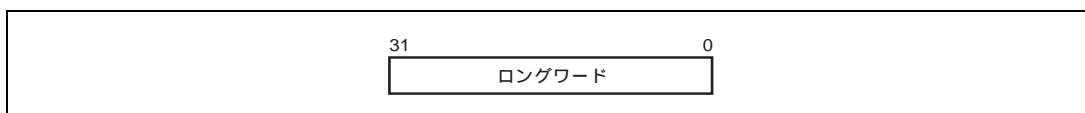


図 2.9 コントロールレジスタの構成

2.4 データ形式

2.4.1 レジスタのデータ形式

レジスタオペランドのデータのサイズは、常にロングワード (32 ビット) です。メモリ上のデータをレジスタへロードするとき、メモリオペランドのデータサイズがバイト (8 ビット) もしくはワード (16 ビット) の場合は、ロングワードに符号拡張して、レジスタに格納します。



2.4.2 メモリ上でのデータ形式

バイト、ワード、ロングワードのデータ形式があります。メモリは8ビットのバイト、16ビットのワード、および32ビットのロングワードいずれの形でもアクセスすることができます。32ビットに満たないメモリオペランドは、符号拡張されてレジスタに格納されます。

ワードオペランドはワード境界（2バイト刻みの偶数番地：2n番地）から、ロングワードオペランドはロングワード境界（4バイト刻みの偶数番地：4n番地）からアクセスしてください。これを守らない場合は、アドレスエラーとなり、例外処理状態に遷移します。バイトオペランドは、どの番地からでもアクセスできます。

ワードオペランドやロングワードオペランドをアクセスする場合、レジスタ上でワード、ロングワードを構成する各バイト位置がメモリ上でどのアドレスに対応するかは、エンディアン方式（ビッグエンディアン/リトルエンディアンの別）により異なります。

図 2.10 に、ビッグエンディアンの場合の対応を示します。ビッグエンディアンでは、レジスタの最上位バイト側がアドレスの小さい番地に、最下位バイト側がアドレスの大きい番地に対応します。例として、汎用レジスタ R1 で示されるアドレスに汎用レジスタの値をロングワードでストアした場合は、R0 の最上位バイトが R1 で示されるアドレスにストアされ、R0 の最下位バイトが (R1+3) で示されるアドレスにストアされます。

本 LSI の内蔵デバイスが持つメモリ割り付けレジスタは、ビッグエンディアンとしてアクセスされます。ただし、ロングワードサイズのレジスタをワードまたはバイトでアクセスすることが可能か、ワードサイズのレジスタをバイトでアクセスすることが可能かは、レジスタによって異なります。

【注】 本 LSI の CPU 命令コードは、メモリ上にワードサイズとして格納されている必要があります。ビッグエンディアンの場合は、命令コードはメモリ上にワード境界から上位バイトおよび下位バイトの順番に格納してください。

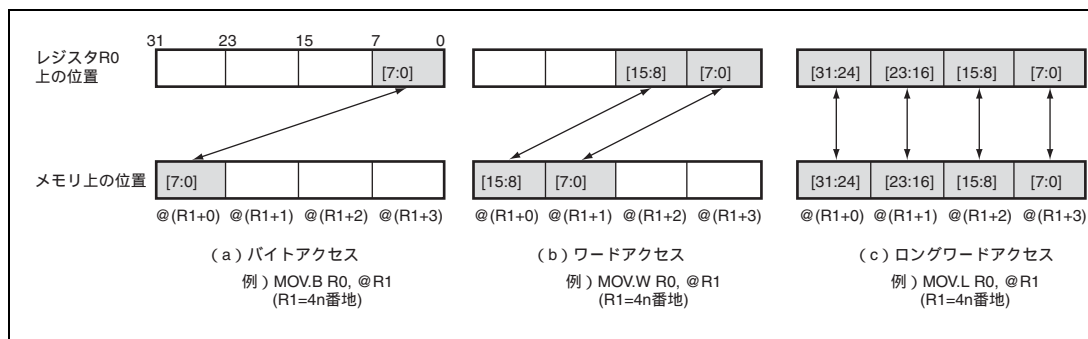


図 2.10 メモリ上のデータ形式（ビッグエンディアン）

データフォーマットとしてリトルエンディアンを選択することも可能です。エンディアン方式は、パワーオンリセット時に外部端子（MD5 端子）で設定してください。MD5 端子がローレベルの場合はビッグエンディアンに、MD5 端子がハイレベルの場合はリトルエンディアンに設定されます。エンディアン方式は、動的には変更できません。

リトルエンディアンでは、レジスタの最上位バイト側がアドレスの大きい番地に、最下位バイト側がアドレスの小さい番地に対応します（図 2.11）。例として、汎用レジスタ R1 で示されるアドレスに汎用レジスタの値を

ロングワードでストアした場合、リトルエンディアンでは R0 の最上位バイトが (R1+3) で示されるアドレスにストアされ、R0 の最下位バイトが R1 で示されるアドレスにストアされます。

エンディアンをリトルエンディアンに設定した場合、本 LSI の持つ内蔵メモリへのアクセスはリトルエンディアンで行われますが、メモリ空間に割り付けられたレジスタへのアクセスはビッグエンディアンで行われます。ただし、ロングワードサイズのレジスタをワードまたはバイトでアクセスすることが可能か、ワードサイズのレジスタをバイトでアクセスすることが可能かは、レジスタによって異なります。

【注】 本 LSI の CPU 命令コードは、メモリ上にワードサイズとして格納されている必要があります。リトルエンディアンの場合は、命令コードはメモリ上にワード境界から下位バイトおよび上位バイトの順に格納してください。

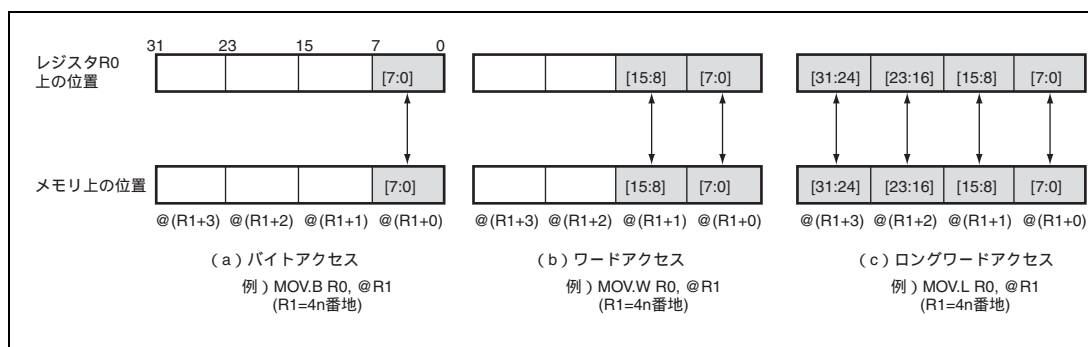


図 2.11 メモリ上のデータ形式 (リトルエンディアン)

2.5 命令の特長

2.5.1 命令の実行方法

(1) 命令長

命令セットは、16 ビット固定長命令で逐次パイプラインで実行されます。逐次パイプラインによりほとんどの命令が 1 サイクルの時間で実行されます。すべてのデータは 32 ビットのロングワード単位で処理されます。メモリは、8 ビットのバイト、16 ビットのワード、および 32 ビットのロングワード単位にアクセスされます。このとき、バイトとワードは、符号拡張されて 32 ビットのロングワードで処理されます。リテラル定数は、算術演算 (MOV、ADD および CMP/EQ 命令) では符号が拡張され、論理演算 (TST、AND、OR および XOR 命令) ではゼロ拡張されて処理されます。

(2) ロードストアアーキテクチャ

ロードストアアーキテクチャを採用しているため、基本演算はレジスタ間で処理されます。メモリ上のデータを扱うときは、転送命令でレジスタにロードしてから演算命令で実行してください。ただし、論理演算の AND のようなビット操作命令は、実際にメモリ上で処理されます。

2. CPU

(3) 遅延分岐

無条件分岐は、遅延分岐として処理されます。分岐によるパイプラインの乱れを最小にするため遅延分岐命令の次の命令を先に実行してから分岐します。例を以下に示します。条件付き分岐命令には、遅延分岐命令と通常分岐命令の2通りがあります。

```
BRA          TRGET
ADD          R1,R0      ;TRGET へ分岐する前に ADD 命令が実行されます。
```

(4) Tビット

ステータスレジスタ (SR) の T ビットは、比較命令の結果を表すのに使用され、条件分岐が発生するか否かを決定する真偽状態 (TRUE または FALSE) として読み出されます。処理速度を向上させるため、T ビットは特定の命令でのみ変更されます。T ビットの使い方の例を以下に示します。

```
ADD          #1,R0      ;T ビットは ADD 命令では変更されません。
CMP/EQ      #0,R0      ;T ビットは R0 が 0 のとき 1 にセットされます。
BT          TRGET      ;T ビットが 1 (R0=0) のとき TRGET に分岐します。
```

(5) リテラル定数

バイト長のリテラル定数は、イミディエイトデータとして命令コードの中に直接挿入されます。16 ビット固定長の命令コードなので、ワードまたはロングワードのリテラル定数は命令コードの中に直接挿入されず、メインメモリにテーブルの形で格納されます。メモリテーブルは、ディスプレースメント付きの PC 相対アドレッシングを使った MOV 命令でアクセスします。例を以下に示します。

```
MOV.W      @(disp,PC),R0
```

(6) 絶対アドレス

ワードおよびロングワードリテラル定数と同じように、絶対アドレスもメインメモリにテーブルの形で格納してください。絶対アドレスの値はレジスタに転送され、オペランドのアクセスはインデックス付きレジスタ間接アドレッシングで指定されます。絶対アドレスは、ワードおよびロングワードのイミディエイトデータと同じように、命令実行中にレジスタに格納されます。

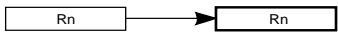
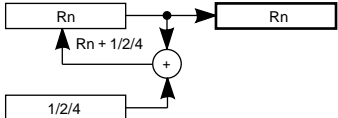
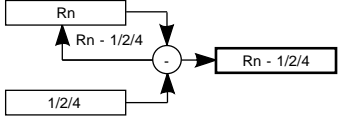
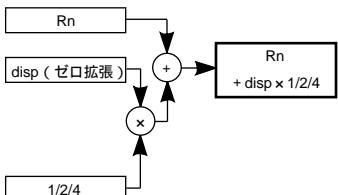
(7) 16 ビットと 32 ビットディスプレースメント

16 ビットおよび 32 ビットディスプレースメントでデータを参照するときは、あらかじめディスプレースメントの値をメモリ上のテーブルに格納してください。絶対アドレスと全く同じようにディスプレースメントの値がレジスタに転送され、オペランドのアクセスはインデックス付きレジスタ間接アドレッシングで指定されます。絶対アドレスはワードおよびロングワードのイミディエイトデータと同じように、命令実行中にレジスタに格納されます。

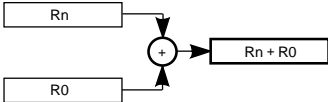
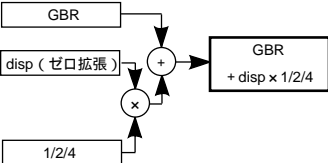
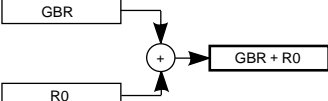
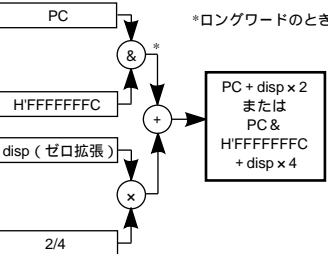
2.5.2 アドレッシングモード

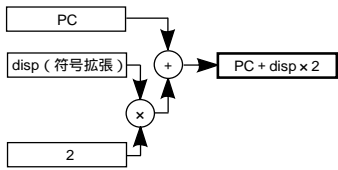
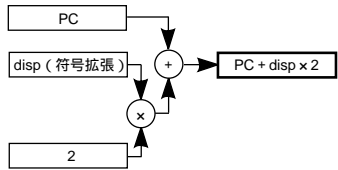
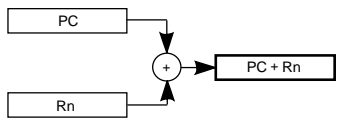
アドレッシングモードと実効アドレスの計算方法を表 2.2 に示します。

表 2.2 アドレッシングモードと実効アドレス

アドレッシングモード	命令フォーマット	実効アドレスの計算方法	計算式
レジスタ直接	Rn	実効アドレスはレジスタ Rn です。 (オペランドはレジスタ Rn の内容です。)	-
レジスタ間接	@Rn	実効アドレスはレジスタ Rn の内容です。 	Rn
ポストインクリメントレジスタ間接	@Rn+	実効アドレスはレジスタ Rn の内容です。命令実行後 Rn に定数を加算します。定数はオペランドサイズがバイトのとき 1、ワードのとき 2、ロングワードのとき 4 です。 	Rn 命令実行後 バイト : Rn + 1 Rn ワード : Rn + 2 Rn ロングワード : Rn + 4 Rn
プリデクリメントレジスタ間接	@ - Rn	実効アドレスは、あらかじめ定数を減算したレジスタ Rn の内容です。定数はバイトのとき 1、ワードのとき 2、ロングワードのとき 4 です。 	バイト : Rn - 1 Rn ワード : Rn - 2 Rn ロングワード : Rn - 4 Rn (計算後の Rn で命令実行)
ディスプレースメント付きレジスタ間接	@ (disp:4, Rn)	実効アドレスはレジスタ Rn に 4 ビットディスプレースメント disp を加算した内容です。disp はゼロ拡張後、オペランドサイズによってバイトで 1 倍、ワードで 2 倍、ロングワードで 4 倍します。 	バイト : Rn + disp ワード : Rn + disp x 2 ロングワード : Rn + disp x 4

2. CPU

アドレッシングモード	命令フォーマット	実効アドレスの計算方法	計算式
インデックス付きレジスタ間接	@ (R0, Rn)	<p>実効アドレスはレジスタ Rn に R0 を加算した内容です。</p> 	$Rn + R0$
ディスプレースメント付き GBR 間接	@ (disp:8, GBR)	<p>実効アドレスはレジスタ GBR に 8 ビットディスプレースメント disp を加算した内容です。disp はゼロ拡張後、オペランドサイズによってバイトで 1 倍、ワードで 2 倍、ロングワードで 4 倍します。</p> 	バイト : $GBR + disp$ ワード : $GBR + disp \times 2$ ロングワード : $GBR + disp \times 4$
インデックス付き GBR 間接	@ (R0, GBR)	<p>実効アドレスはレジスタ GBR に R0 を加算した内容です。</p> 	$GBR + R0$
ディスプレースメント付き PC 相対	@ (disp:8, PC)	<p>実効アドレスはレジスタ PC に 8 ビットディスプレースメント disp を加算した内容です。disp はゼロ拡張後、オペランドサイズによってワードで 2 倍、ロングワードで 4 倍します。さらにロングワードのときは PC の下位 2 ビットをマスクします。</p> 	ワード : $PC + disp \times 2$ ロングワード : $PC \& HFFFFFFFC + disp \times 4$

アドレッシングモード	命令フォーマット	実効アドレスの計算方法	計算式
PC 相対	disp:8	<p>実効アドレスはレジスタ PC に 8 ビットディスプレイメント disp を符号拡張後 2 倍し、加算した内容です。</p> 	$PC + disp \times 2$
	disp:12	<p>実効アドレスはレジスタ PC に 12 ビットディスプレイメント disp を符号拡張後 2 倍し、加算した内容です。</p> 	$PC + disp \times 2$
	Rn	<p>実効アドレスはレジスタ PC に Rn を加算した内容です。</p> 	$PC + Rn$
イミディエイト	#imm:8	TST, AND, OR, XOR 命令の 8 ビットイミディエイト imm はゼロ拡張します。	-
	#imm:8	MOV, ADD, CMP/EQ 命令の 8 ビットイミディエイト imm は符号拡張します。	-
	#imm:8	TRAPA 命令の 8 ビットイミディエイト imm はゼロ拡張後、4 倍します。	-

【注】 下記のディスプレイメント (disp) を伴うアドレッシングモードにおいて、本マニュアルのアセンブラ記述は、オペランドサイズに応じたスケーリング ($\times 1$ 、 $\times 2$ 、 $\times 4$) を行う前の値を表記しています。これは、本 LSI の動作を明確にするためです。実際のアセンブラの記述は、各アセンブラの表記ルールを参照してください。

@ (disp:4, Rn) ;ディスプレイメント付きレジスタ間接

@ (disp:8, GBR) ;ディスプレイメント付き GBR 間接

@ (disp:8, PC) ;ディスプレイメント付き PC 相対

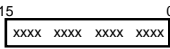
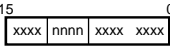
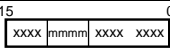
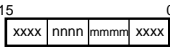
disp : 8, disp :12 ;PC 相対

2.5.3 命令形式

表 2.3 に命令形式、ソースオペランド、およびデスティネーションオペランドの意味を示します。命令コードによりオペランドの意味が異なります。記号は次のとおりです。

xxxx : 命令コード
 mmmm : ソースレジスタ
 nnnn : デスティネーションレジスタ
 iiii : イミディエイトデータ
 dddd : ディスプレースメント

表 2.3 命令形式

命令形式	ソースオペランド	デスティネーション オペランド	命令の例
0 形式			NOP
n 形式		nnnn: レジスタ直接	MOVT Rn
		コントロールレジスタ またはシステムレジスタ	nnnn: レジスタ直接 STS MACH,Rn
		コントロールレジスタ またはシステムレジスタ	nnnn: プリデクリメン トレジスタ間接 STC.L SR,@-Rn
m 形式		mmmm: レジスタ直接	コントロールレジスタ またはシステムレジスタ LDC Rm,SR
		mmmm: ポストインクリメ ントレジスタ間接	コントロールレジスタ またはシステムレジスタ LDC.L @Rm+,SR
		mmmm: レジスタ間接	JMP @Rm
		mmmm: Rm を用いた PC 相 対	BRAF Rm
nm 形式		mmmm: レジスタ直接	nnnn: レジスタ直接 ADD Rm,Rn
		mmmm: レジスタ直接	nnnn: レジスタ間接 MOV.L Rm,@Rn
		mmmm: ポストインクリメ ントレジスタ間接 (積和演算) nnnn: *ポストインクリメン トレジスタ間接 (積和演算)	MACH,MACL MAC.W @Rm+,@Rn+
		mmmm: ポストインクリメ ントレジスタ間接	nnnn: レジスタ直接 MOV.L @Rm+,Rn

命令形式		ソースオペランド	デスティネーション オペランド	命令の例
nm 形式		mmmm: レジスタ 直接	nnnn: プリデクリメント レジスタ間接	MOV.L Rm,@-Rn
		mmmm: レジスタ 直接	nnnn: インデックス 付きレジスタ間接	MOV.L Rm,@(R0,Rn)
md 形式		mddddddd: ディスプレイスメン ト付きレジスタ間接	R0 (レジスタ直接)	MOV.B @(disp,Rm),R0
nd4 形式		R0 (レジスタ直接)	nnnndddd: ディスプレイスメン ト付き レジスタ間接	MOV.B R0,@(disp,Rn)
nmd 形式		mmmm: レジスタ 直接	nnnndddd: ディスプレイスメン ト付き レジスタ間接	MOV.L Rm,@(disp,Rn)
		mddddddd: ディスプレイスメン ト付きレジスタ間接	nnnn: レジスタ直接	MOV.L @(disp,Rm),Rn
d 形式		ddddddd: ディスプレイスメン ト付き GBR 間接	R0 (レジスタ直接)	MOV.L @(disp,GBR),R0
		R0 (レジスタ直接)	ddddddd: ディスプレイスメン ト付き GBR 間接	MOV.L R0,@(disp,GBR)
		ddddddd: ディスプレイスメン ト付き PC 相対	R0 (レジスタ直接)	MOVA @(disp,PC),R0
		ddddddd: PC 相対		BF label
d12 形式		ddddddddddd: PC 相対		BRA label (label=disp+PC)
nd8 形式		ddddddd: ディスプレイスメン ト付き PC 相対	nnnn: レジスタ直接	MOV.L @(disp,PC),Rn
i 形式		iiiiiii: イミディエイト	インデックス付き GBR 間接	AND.B #imm,@(R0,GBR)
		iiiiiii: イミディエイト	R0 (レジスタ直接)	AND #imm,R0
		iiiiiii: イミディエイト		TRAPA #imm
ni 形式		iiiiiii: イミディエイト	nnnn: レジスタ直接	ADD #imm,Rn

【注】 * 積和命令では、nnnn はソースレジスタです。

2.6 命令セット

2.6.1 機能別命令セット

機能別の命令を表 2.4 に示します。

表 2.4 機能別命令

機能	命令の種類	オペコード	詳細機能	命令数
データ転送命令	5	MOV	データ転送	39
		MOVA	実効アドレスの転送	
		MOVT	T ビットの転送	
		SWAP	上位と下位の交換	
		XTRCT	連結レジスタの中央切り出し	
算術演算命令	21	ADD	2 進加算	33
		ADDC	キャリ付き 2 進加算	
		ADDV	オーバフロー付き 2 進加算	
		CMP/cond	比較	
		DIV1	除算	
		DIV0S	符号付き除算の初期化	
		DIV0U	符号なし除算の初期化	
		DMULS	符号付き倍精度乗算	
		DMULU	符号なし倍精度乗算	
		DT	デクリメントとテスト	
		EXTS	符号拡張	
		EXTU	ゼロ拡張	
		MAC	積和演算、倍精度積和演算	
		MUL	倍精度乗算 (32×32 ビット)	
		MULS	符号付き乗算 (16×16 ビット)	
		MULU	符号なし乗算 (16×16 ビット)	
		NEG	符号反転	
		NEGC	ポロ-付き符号反転	
		SUB	2 進減算	
		SUBC	ポロ-付き 2 進減算	
SUBV	アンダフロー付き 2 進減算			

機能	命令の種類	オペコード	詳細機能	命令数
論理演算命令	6	AND	論理積演算	14
		NOT	ビット反転	
		OR	論理和演算	
		TAS	メモリテストとビットセット	
		TST	論理積演算のTビットセット	
		XOR	排他的論理和演算	
シフト命令	12	ROTCL	Tビット付き1ビット左回転	16
		ROTCR	Tビット付き1ビット右回転	
		ROTL	1ビット左回転	
		ROTR	1ビット右回転	
		SHAD	ダイナミック算術シフト	
		SHAL	1ビット左算術シフト	
		SHAR	1ビット右算術シフト	
		SHLD	ダイナミック論理シフト	
		SHLL	1ビット左論理シフト	
		SHLLn	nビット左論理シフト	
		SHLR	1ビット右論理シフト	
		SHLRn	nビット右論理シフト	
分岐命令	9	BF	条件分岐、遅延付き条件分岐 (T=0で分岐)	11
		BT	条件分岐、遅延付き条件分岐 (T=1で分岐)	
		BRA	無条件分岐	
		BRAF	無条件分岐	
		BSR	サブルーチンプロシージャへの分岐	
		BSRF	サブルーチンプロシージャへの分岐	
		JMP	無条件分岐	
		JSR	サブルーチンプロシージャへの分岐	
RTS	サブルーチンプロシージャからの復帰			
システム制御命令	14	CLRMAC	MACレジスタのクリア	74
		CLRS	Sビットのクリア	
		CLRT	Tビットのクリア	
		LDC	コントロールレジスタへのロード	
		LDS	システムレジスタへのロード	
		NOP	無操作	
		PREF	データキャッシュへのプリフェッチ	
		RTE	例外処理からの復帰	
		SETS	Sビットのセット	
		SETT	Tビットのセット	

2. CPU

機能	命令の種類	オペコード	詳細機能	命令数
システム制御命令	14	SLEEP	低消費電力モードへの遷移	74
		STC	コントロールレジスタからのストア	
		STS	システムレジスタからのストア	
		TRAPA	トラップ例外処理	
	計 67			187

CPU 命令の命令コード、動作、および実行ステートを、以下の形式で分類別に、表 2.5～表 2.10 に説明します。

命令	命令コード	動作の概略	特権	実行ステート	T ビット
ニーモニックで表示しています。	MSB LSB の順で表示しています。	動作の概略を表示しています。	特権命令を示しています。	ノーウェイトのときの値です。*1	命令実行後の、T ビットの値を表示しています。
記号の説明	記号の説明	記号の説明			記号の説明 : 変化しない。
OP: Sz SRC, DEST	mmmm: ソースレジスタ	(xx): メモリオペランド			
OP: オペコード	nnnn: デスティネーションレジスタ	M/Q/T: SR 内のフラグビット			
Sz: サイズ		&: ビット毎の論理積			
SRC: ソース	0000: R0	: ビット毎の論理和			
DEST: デスティネーション	0001: R1	^: ビット毎の排他的論理和			
Rm: ソースレジスタ	~: ビット毎の論理否定			
Rn: デスティネーションレジスタ	1111: R15	<<n: 左 n ビットシフト			
imm: イミディエイトデータ	iiii: イミディエイトデータ	>>n: 右 n ビットシフト			
disp: ディスプレースメント	dddd: ディスプレースメント*2				

【注】 *1 命令の実行ステートについて

表に示した実行ステートは、最少値です。実際は、下記の条件により、命令実行ステート数が増加します。

- (1) 命令フェッチとデータアクセスの競合が起こる場合
- (2) ロード命令（メモリ レジスタ）のデスティネーションレジスタと、その直後の命令が使うレジスタとが同一の場合

*2 命令のオペランドサイズなどに応じてスケールリング（×1、×2、×4）されます。

表 2.5 データ転送命令

命令	命令コード	動作	特権	実行 ステート	Tビット
MOV #imm,Rn	1110nnnniiiiiiii	imm 符号拡張 Rn		1	
MOV.W @(disp,PC),Rn	1001nnnnndddd	(disp × 2+PC) 符号拡張 Rn		1	
MOV.L @(disp,PC),Rn	1101nnnnndddd	(disp × 4+PC) Rn		1	
MOV Rm,Rn	0110nnnnnnmm0011	Rm Rn		1	
MOV.B Rm,@Rn	0010nnnnnnmm0000	Rm (Rn)		1	
MOV.W Rm,@Rn	0010nnnnnnmm0001	Rm (Rn)		1	
MOV.L Rm,@Rn	0010nnnnnnmm0010	Rm (Rn)		1	
MOV.B @Rm,Rn	0110nnnnnnmm0000	(Rm) 符号拡張 Rn		1	
MOV.W @Rm,Rn	0110nnnnnnmm0001	(Rm) 符号拡張 Rn		1	
MOV.L @Rm,Rn	0110nnnnnnmm0010	(Rm) Rn		1	
MOV.B Rm,@-Rn	0010nnnnnnmm0100	Rn-1 Rn, Rm (Rn)		1	
MOV.W Rm,@-Rn	0010nnnnnnmm0101	Rn-2 Rn, Rm (Rn)		1	
MOV.L Rm,@-Rn	0010nnnnnnmm0110	Rn-4 Rn, Rm (Rn)		1	
MOV.B @Rm+,Rn	0110nnnnnnmm0100	(Rm) 符号拡張 Rn, Rm+1 Rm		1	
MOV.W @Rm+,Rn	0110nnnnnnmm0101	(Rm) 符号拡張 Rn, Rm+2 Rm		1	
MOV.L @Rm+,Rn	0110nnnnnnmm0110	(Rm) Rn, Rm+4 Rm		1	
MOV.B R0,@(disp,Rn)	10000000nnndddd	R0 (disp+Rn)		1	
MOV.W R0,@(disp,Rn)	10000001nnndddd	R0 (disp × 2+Rn)		1	
MOV.L Rm,@(disp,Rn)	0001nnnnnnmmddd	Rm (disp × 4+Rn)		1	
MOV.B @(disp,Rm),R0	10000100nnndddd	(disp+Rm) 符号拡張 R0		1	
MOV.W @(disp,Rm),R0	10000101nnndddd	(disp × 2+Rm) 符号拡張 R0		1	
MOV.L @(disp,Rm),Rn	0101nnnnnnmmddd	(disp × 4+Rm) Rn		1	
MOV.B Rm,@(R0,Rn)	0000nnnnnnmm0100	Rm (R0+Rn)		1	
MOV.W Rm,@(R0,Rn)	0000nnnnnnmm0101	Rm (R0+Rn)		1	
MOV.L Rm,@(R0,Rn)	0000nnnnnnmm0110	Rm (R0+Rn)		1	
MOV.B @(R0,Rm),Rn	0000nnnnnnmm1100	(R0+Rm) 符号拡張 Rn		1	
MOV.W @(R0,Rm),Rn	0000nnnnnnmm1101	(R0+Rm) 符号拡張 Rn		1	
MOV.L @(R0,Rm),Rn	0000nnnnnnmm1110	(R0+Rm) Rn		1	
MOV.B R0,@(disp,GBR)	11000000ddddddd	R0 (disp+GBR)		1	
MOV.W R0,@(disp,GBR)	11000001ddddddd	R0 (disp × 2+GBR)		1	
MOV.L R0,@(disp,GBR)	11000010ddddddd	R0 (disp × 4+GBR)		1	
MOV.B @(disp,GBR),R0	11000100ddddddd	(disp+GBR) 符号拡張 R0		1	
MOV.W @(disp,GBR),R0	11000101ddddddd	(disp × 2+GBR) 符号拡張 R0		1	
MOV.L @(disp,GBR),R0	11000110ddddddd	(disp × 4+GBR) R0		1	

2. CPU

命令	命令コード	動作	特権	実行 ステート	Tビット
MOVA @ (disp,PC),R0	11000111dddddddd	disp × 4+PC R0		1	
MOVT Rn	0000nnnn00101001	T Rn		1	
SWAP.B Rm,Rn	0110nnnnnnmmmm1000	Rm 下位 2 バイトの上下バイト交換 Rn		1	
SWAP.W Rm,Rn	0110nnnnnnmmmm1001	Rm 上下ワード交換 Rn		1	
XTRCT Rm,Rn	0010nnnnnnmmmm1101	Rm と Rn の中央 32 ビット Rn		1	

表 2.6 算術演算命令

命令	命令コード	動作	特権	実行 ステート	Tビット
ADD Rm,Rn	0011nnnnnnmmmm1100	Rn+Rm Rn		1	
ADD #imm,Rn	0111nnnnnniiiiiiii	Rn+imm Rn		1	
ADDC Rm,Rn	0011nnnnnnmmmm1110	Rn+Rm+T Rn, キャリ T		1	キャリ
ADDV Rm,Rn	0011nnnnnnmmmm1111	Rn+Rm Rn, オーバフロー T		1	オーバフロー
CMP/EQ #imm,R0	10001000iiiiiiii	R0=imm のとき 1 T		1	比較結果
CMP/EQ Rm,Rn	0011nnnnnnmmmm0000	Rn=Rm のとき 1 T		1	比較結果
CMP/HS Rm,Rn	0011nnnnnnmmmm0010	無符号で Rn Rm のとき 1 T		1	比較結果
CMP/GE Rm,Rn	0011nnnnnnmmmm0011	有符号で Rn Rm のとき 1 T		1	比較結果
CMP/HI Rm,Rn	0011nnnnnnmmmm0110	無符号で Rn>Rm のとき 1 T		1	比較結果
CMP/GT Rm,Rn	0011nnnnnnmmmm0111	有符号で Rn>Rm のとき 1 T		1	比較結果
CMP/PL Rn	0100nnnn00010101	Rn>0 のとき 1 T		1	比較結果
CMP/PZ Rn	0100nnnn00010001	Rn 0 のとき 1 T		1	比較結果
CMP/STR Rm,Rn	0010nnnnnnmmmm1100	いずれかのバイトが等しいとき 1 T		1	比較結果
DIV1 Rm,Rn	0011nnnnnnmmmm0100	1 ステップ除算 (Rn ÷ Rm)		1	計算結果
DIV0S Rm,Rn	0010nnnnnnmmmm0111	Rn の MSB Q, Rm の MSB M, M^Q T		1	計算結果
DIV0U	000000000011001	0 M/Q/T		1	0
DMULS.L Rm,Rn	0011nnnnnnmmmm1101	符号付きで Rn × Rm MACH, MACL 32 × 32 64 ビット		2 (~ 5) *	
DMULU.L Rm,Rn	0011nnnnnnmmmm0101	符号なしで Rn × Rm MACH, MACL 32 × 32 64 ビット		2 (~ 5) *	
DT Rn	0100nnnn00010000	Rn-1 Rn, Rn が 0 のとき 1 T, Rn が 0 以外 のとき 0 T		1	比較結果
EXTS.B Rm,Rn	0110nnnnnnmmmm1110	Rm をバイトから符号拡張 Rn		1	
EXTS.W Rm,Rn	0110nnnnnnmmmm1111	Rm をワードから符号拡張 Rn		1	
EXTU.B Rm,Rn	0110nnnnnnmmmm1100	Rm をバイトからゼロ拡張 Rn		1	
EXTU.W Rm,Rn	0110nnnnnnmmmm1101	Rm をワードからゼロ拡張 Rn		1	

命令	命令コード	動作	特権	実行 ステート	Tビット
MAC.L	@Rm+, @Rn+	0000nnnnnnmmmm1111 符号付きで (Rn) × (Rm) + MAC MAC, Rn+4 Rn, Rm+4 Rm 32 × 32+64 64 ビット		2 (~5)*	
MAC.W	@Rm+, @Rn+	0100nnnnnnmmmm1111 符号付きで (Rn) × (Rm) + MAC MAC, Rn+2 Rn, Rm+2 Rm 16 × 16+64 64 ビット		2 (~5)*	
MUL.L	Rm, Rn	0000nnnnnnmmmm0111 Rn × Rm MACL 32 × 32 32 ビット		2 (~5)*	
MULS.W	Rm, Rn	0010nnnnnnmmmm1111 符号付きで Rn × Rm MACL 16 × 16 32 ビット		1 (~3)*	
MULU.W	Rm, Rn	0010nnnnnnmmmm1110 符号なしで Rn × Rm MACL 16 × 16 32 ビット		1 (~3)*	
NEG	Rm, Rn	0110nnnnnnmmmm1011 0-Rm Rn		1	
NEGC	Rm, Rn	0110nnnnnnmmmm1010 0-Rm-T Rn, ポロー T		1	ポロー
SUB	Rm, Rn	0011nnnnnnmmmm1000 Rn-Rm Rn		1	
SUBC	Rm, Rn	0011nnnnnnmmmm1010 Rn-Rm-T Rn, ポロー T		1	ポロー
SUBV	Rm, Rn	0011nnnnnnmmmm1011 Rn-Rm Rn, アンダフロー T		1	アンダフロー

【注】 * 命令の直後に MACH/MACL レジスタから演算結果を読み出すときは、() 内に示されたステート数が必要です。

表 2.7 論理演算命令

命令	命令コード	動作	特権	実行 ステート	Tビット
AND	Rm, Rn	0010nnnnnnmmmm1001 Rn & Rm Rn		1	
AND	#imm, R0	11001001iiiiiiii R0 & imm R0		1	
AND.B	#imm, @ (R0, GBR)	11001101iiiiiiii (R0+GBR) & imm (R0+GBR)		3	
NOT	Rm, Rn	0110nnnnnnmmmm0111 ~Rm Rn		1	
OR	Rm, Rn	0010nnnnnnmmmm1011 Rn Rm Rn		1	
OR	#imm, R0	11001011iiiiiiii R0 imm R0		1	
OR.B	#imm, @ (R0, GBR)	11001111iiiiiiii (R0+GBR) imm (R0+GBR)		3	
TAS.B	@Rn	0100nnnn00011011 (Rn) が 0 のとき 1 T, 1 MSB of (Rn)		4	テスト結果
TST	Rm, Rn	0010nnnnnnmmmm1000 Rn & Rm, 結果が 0 のとき 1 T		1	テスト結果
TST	#imm, R0	11001000iiiiiiii R0 & imm, 結果が 0 のとき 1 T		1	テスト結果
TST.B	#imm, @ (R0, GBR)	11001100iiiiiiii (R0+GBR) & imm, 結果が 0 のとき 1 T		3	テスト結果
XOR	Rm, Rn	0010nnnnnnmmmm1010 Rn ^ Rm Rn		1	
XOR	#imm, R0	11001010iiiiiiii R0 ^ imm R0		1	
XOR.B	#imm, @ (R0, GBR)	11001110iiiiiiii (R0+GBR) ^ imm (R0+GBR)		3	

2. CPU

表 2.8 シフト命令

命令	命令コード	動作	特権	実行 ステート	Tビット
ROTL Rn	0100nnnn00000100	T Rn MSB		1	MSB
ROTR Rn	0100nnnn00000101	LSB Rn T		1	LSB
ROTCL Rn	0100nnnn00100100	T Rn T		1	MSB
ROTCR Rn	0100nnnn00100101	T Rn T		1	LSB
SHAD Rm, Rn	0100nnnnnnmm1100	Rm 0 のとき、Rn<<Rm Rn Rm<0 のとき、Rn>>Rm [MSB Rn]		1	
SHAL Rn	0100nnnn00100000	T Rn 0		1	MSB
SHAR Rn	0100nnnn00100001	MSB Rn T		1	LSB
SHLD Rm, Rn	0100nnnnmmmm1101	Rm 0 のとき、Rn<<Rm Rn Rm<0 のとき、Rn>>Rm [0 Rn]		1	
SHLL Rn	0100nnnn00000000	T Rn 0		1	MSB
SHLR Rn	0100nnnn00000001	0 Rn T		1	LSB
SHLL2 Rn	0100nnnn00001000	Rn<<2 Rn		1	
SHLR2 Rn	0100nnnn00001001	Rn>>2 Rn		1	
SHLL8 Rn	0100nnnn00011000	Rn<<8 Rn		1	
SHLR8 Rn	0100nnnn00011001	Rn>>8 Rn		1	
SHLL16 Rn	0100nnnn00101000	Rn<<16 Rn		1	
SHLR16 Rn	0100nnnn00101001	Rn>>16 Rn		1	

表 2.9 分岐命令

命令	命令コード	動作	特権	実行 ステート	Tビット
BF disp	10001011dddddddd	T=0 のとき disp × 2+PC へ分岐 T=1 のとき nop		3/1*	
BF/S disp	10001111dddddddd	遅延分岐、T=0 のとき disp × 2+PC へ分岐 T=1 のとき nop		2/1*	
BT disp	10001001dddddddd	T=1 のとき disp × 2+PC へ分岐 T=0 のとき nop		3/1*	
BT/S disp	10001101dddddddd	遅延分岐、T=1 のとき disp × 2+PC へ分岐 T=0 のとき nop		2/1*	
BRA disp	1010dddddddddddd	遅延分岐、disp × 2+PC へ分岐		2	
BRAF Rm	0000mmmm00100011	遅延分岐、Rm+PC へ分岐		2	
BSR disp	1011dddddddddddd	遅延分岐、遅延スロット命令の次命令のアド レス PR, disp × 2+PC へ分岐		2	

命令	命令コード	動作	特権	実行 ステート	Tビット
BSRF Rm	0000mmmmmm00000011	遅延分岐、遅延スロット命令の次命令のアドレス PR, Rm+PC へ分岐		2	
JMP @Rm	0100mmmmmm00101011	遅延分岐、Rm へ分岐		2	
JSR @Rm	0100mmmmmm00001011	遅延分岐、遅延スロット命令の次命令のアドレス PR, Rm へ分岐		2	
RTS	0000000000001011	遅延分岐、PR へ分岐		2	

【注】 * 分岐しないときは、1ステートになります。

表 2.10 システム制御命令

命令	命令コード	動作	特権	実行 ステート	Tビット
CLRMAC	0000000000101000	0 MACH,MACL		1	
CLRS	0000000001001000	0 S		1	
CLRT	0000000000001000	0 T		1	0
LDC Rm,SR	0100mmmmmm00001110	Rm SR	特権	6	LSB
LDC Rm,GBR	0100mmmmmm00011110	Rm GBR		4	
LDC Rm,VBR	0100mmmmmm00101110	Rm VBR	特権	4	
LDC Rm,SSR	0100mmmmmm00111110	Rm SSR	特権	4	
LDC Rm,SPC	0100mmmmmm01001110	Rm SPC	特権	4	
LDC Rm,R0_BANK	0100mmmmmm10001110	Rm R0_BANK	特権	4	
LDC Rm,R1_BANK	0100mmmmmm10011110	Rm R1_BANK	特権	4	
LDC Rm,R2_BANK	0100mmmmmm10101110	Rm R2_BANK	特権	4	
LDC Rm,R3_BANK	0100mmmmmm10111110	Rm R3_BANK	特権	4	
LDC Rm,R4_BANK	0100mmmmmm11001110	Rm R4_BANK	特権	4	
LDC Rm,R5_BANK	0100mmmmmm11011110	Rm R5_BANK	特権	4	
LDC Rm,R6_BANK	0100mmmmmm11101110	Rm R6_BANK	特権	4	
LDC Rm,R7_BANK	0100mmmmmm11111110	Rm R7_BANK	特権	4	
LDC.L @Rm+,SR	0100mmmmmm00000111	(Rm) SR, Rm+4 Rm	特権	8	LSB
LDC.L @Rm+,GBR	0100mmmmmm00010111	(Rm) GBR, Rm+4 Rm		4	
LDC.L @Rm+,VBR	0100mmmmmm00100111	(Rm) VBR, Rm+4 Rm	特権	4	
LDC.L @Rm+,SSR	0100mmmmmm00110111	(Rm) SSR, Rm+4 Rm	特権	4	
LDC.L @Rm+,SPC	0100mmmmmm01000111	(Rm) SPC, Rm+4 Rm	特権	4	
LDC.L @Rm+,R0_BANK	0100mmmmmm10000111	(Rm) R0_BANK, Rm+4 Rm	特権	4	
LDC.L @Rm+,R1_BANK	0100mmmmmm10010111	(Rm) R1_BANK, Rm+4 Rm	特権	4	
LDC.L @Rm+,R2_BANK	0100mmmmmm10100111	(Rm) R2_BANK, Rm+4 Rm	特権	4	
LDC.L @Rm+,R3_BANK	0100mmmmmm10110111	(Rm) R3_BANK, Rm+4 Rm	特権	4	

2. CPU

命令	命令コード	動作	特権	実行 ステート	Tビット	
LDC.L	@Rm+,R4_BANK	0100mrrrrnn11000111	(Rm) R4_BANK, Rm+4 Rm	特権	4	
LDC.L	@Rm+,R5_BANK	0100mrrrrnn11010111	(Rm) R5_BANK, Rm+4 Rm	特権	4	
LDC.L	@Rm+,R6_BANK	0100mrrrrnn11100111	(Rm) R6_BANK, Rm+4 Rm	特権	4	
LDC.L	@Rm+,R7_BANK	0100mrrrrnn11110111	(Rm) R7_BANK, Rm+4 Rm	特権	4	
LDS	Rm,MACH	0100mrrrrnn00001010	Rm MACH		1	
LDS	Rm,MACL	0100mrrrrnn00011010	Rm MACL		1	
LDS	Rm,PR	0100mrrrrnn00101010	Rm PR		1	
LDS.L	@Rm+,MACH	0100mrrrrnn00000110	(Rm) MACH, Rm+4 Rm		1	
LDS.L	@Rm+,MACL	0100mrrrrnn00010110	(Rm) MACL, Rm+4 Rm		1	
LDS.L	@Rm+,PR	0100mrrrrnn00100110	(Rm) PR, Rm+4 Rm		1	
NOP		0000000000001001	無操作		1	
PREF	@Rm	0000mrrrrnn10000011	(Rm) キャッシュ		1	
RTE		0000000000101011	遅延分岐、SSR SR、SPC へ分岐	特権	5	
SETS		0000000001011000	1 S		1	
SETT		0000000000011000	1 T		1	1
SLEEP		0000000000011011	スリープ	特権	4 ^{*1}	
STC	SR,Rn	0000nrrrrnn00000010	SR Rn	特権	1	
STC	GBR,Rn	0000nrrrrnn00010010	GBR Rn		1	
STC	VBR,Rn	0000nrrrrnn00100010	VBR Rn	特権	1	
STC	SSR, Rn	0000nrrrrnn00110010	SSR Rn	特権	1	
STC	SPC,Rn	0000nrrrrnn01000010	SPC Rn	特権	1	
STC	R0_BANK,Rn	0000nrrrrnn10000010	R0_BANK Rn	特権	1	
STC	R1_BANK,Rn	0000nrrrrnn10010010	R1_BANK Rn	特権	1	
STC	R2_BANK,Rn	0000nrrrrnn10100010	R2_BANK Rn	特権	1	
STC	R3_BANK,Rn	0000nrrrrnn10110010	R3_BANK Rn	特権	1	
STC	R4_BANK,Rn	0000nrrrrnn11000010	R4_BANK Rn	特権	1	
STC	R5_BANK,Rn	0000nrrrrnn11010010	R5_BANK Rn	特権	1	
STC	R6_BANK,Rn	0000nrrrrnn11100010	R6_BANK Rn	特権	1	
STC	R7_BANK,Rn	0000nrrrrnn11110010	R7_BANK Rn	特権	1	
STC.L	SR,@-Rn	0100nrrrrnn00000011	Rn-4 Rn,SR (Rn)	特権	1	
STC.L	GBR,@-Rn	0100nrrrrnn00010011	Rn-4 Rn,GBR (Rn)		1	
STC.L	VBR,@-Rn	0100nrrrrnn00100011	Rn-4 Rn,VBR (Rn)	特権	1	

命令	命令コード	動作	特権	実行 ステート	Tビット	
STC.L	SSR,@-Rn	0100nnnn00110011	Rn-4 Rn, SSR (Rn)	特権	1	
STC.L	SPC,@-Rn	0100nnnn01000011	Rn-4 Rn, SPC (Rn)	特権	1	
STC.L	R0_BANK,@-Rn	0100nnnn10000011	Rn-4 Rn, R0_BANK (Rn)	特権	1	
STC.L	R1_BANK,@-Rn	0100nnnn10010011	Rn-4 Rn, R1_BANK (Rn)	特権	1	
STC.L	R2_BANK,@-Rn	0100nnnn10100011	Rn-4 Rn, R2_BANK (Rn)	特権	1	
STC.L	R3_BANK,@-Rn	0100nnnn10110011	Rn-4 Rn, R3_BANK (Rn)	特権	1	
STC.L	R4_BANK,@-Rn	0100nnnn11000011	Rn-4 Rn, R4_BANK (Rn)	特権	1	
STC.L	R5_BANK,@-Rn	0100nnnn11010011	Rn-4 Rn, R5_BANK (Rn)	特権	1	
STC.L	R6_BANK,@-Rn	0100nnnn11100011	Rn-4 Rn, R6_BANK (Rn)	特権	1	
STC.L	R7_BANK,@-Rn	0100nnnn11110011	Rn-4 Rn, R7_BANK (Rn)	特権	1	
STS	MACH,Rn	0000nnnn00001010	MACH Rn		1	
STS	MACL,Rn	0000nnnn00011010	MACL Rn		1	
STS	PR,Rn	0000nnnn00101010	PR Rn		1	
STS.L	MACH,@-Rn	0100nnnn00000010	Rn-4 Rn, MACH (Rn)		1	
STS.L	MACL,@-Rn	0100nnnn00010010	Rn-4 Rn, MACL (Rn)		1	
STS.L	PR,@-Rn	0100nnnn00100010	Rn-4 Rn, PR (Rn)		1	
TRAPA	#imm	11000011iiiiiiii	無条件トラップ例外発生*2		8	

【注】 *1 スリープ状態に遷移するまでの最小ステート数です。

*2 「第4章 例外処理」を参照してください。

1. 命令の実行ステートについて

表に示した実行ステートは、最少値です。実際は、下記の条件により、命令実行ステート数が増加します。

(1) 命令フェッチとデータアクセスの競合が起こる場合

(2) ロード命令(メモリ レジスタ)のデスティネーションレジスタと、その直後の命令が使うレジスタとが同一の場合

2. 下記のディスプレースメント (disp) を伴うアドレッシングモードにおいて、本マニュアルのアセンブラ記述は、オペランドサイズに応じたスケールリング (×1、×2、×4) を行う前の値を表記しています。これは、本 LSI の動作を明確にするためです。実際のアセンブラの記述は、各アセンブラの表記ルールを参照してください。

@ (disp:4, Rn) ;ディスプレースメント付きレジスタ間接

@ (disp:8, GBR) ;ディスプレースメント付き GBR 間接

@ (disp:8, PC) ;ディスプレースメント付き PC 相対

disp : 8, disp : 12 ;PC 相対

2. CPU

2.6.2 オペレーションコードマップ

オペレーションコードマップを表 2.11 に示します。

表 2.11 オペレーションコードマップ

命令コード				Fx: 0000	Fx: 0001	Fx: 0010	Fx: 0011 ~ 1111
MSB		LSB		MD: 00	MD: 01	MD: 10	MD: 11
0000	Rn	Fx	0000				
0000	Rn	Fx	0001				
0000	Rn	00MD	0010	STC SR, Rn	STC GBR, Rn	STC VBR, Rn	STC SSR, Rn
0000	Rn	01MD	0010	STC SPC, Rn			
0000	Rn	10MD	0010	STC R0_BANK, Rn	STC R1_BANK, Rn	STC R2_BANK, Rn	STC R3_BANK, Rn
0000	Rn	11MD	0010	STC R4_BANK, Rn	STC R5_BANK, Rn	STC R6_BANK, Rn	STC R7_BANK, Rn
0000	Rm	00MD	0011	BSRF Rm		BRAF Rm	
0000	Rm	10MD	0011	PREF @Rm			
0000	Rn	Rm	01MD	MOV.B Rm, @ (R0, Rn)	MOV.W Rm, @ (R0, Rn)	MOV.L Rm, @ (R0, Rn)	MUL.L Rm, Rn
0000	0000	00MD	1000	CLRT	SETT	CLRMAC	
0000	0000	01MD	1000	CLRS	SETS		
0000	0000	Fx	1001	NOP	DIV0U		
0000	0000	Fx	1010				
0000	0000	Fx	1011	RTS	SLEEP	RTE	
0000	Rn	Fx	1000				
0000	Rn	Fx	1001			MOVT Rn	
0000	Rn	Fx	1010	STS MACH, Rn	STS MACL, Rn	STS PR, Rn	
0000	Rn	Fx	1011				
0000	Rn	Rm	11MD	MOV.B @ (R0, Rm) , Rn	MOV.W @ (R0, Rm) , Rn	MOV.L @ (R0, Rm) , Rn	MAC.L @Rm + , @Rn +
0001	Rn	Rm	disp	MOV.L Rm, @ (disp:4, Rn)			
0010	Rn	Rm	00MD	MOV.B Rm, @Rn	MOV.W Rm, @Rn	MOV.L Rm, @Rn	
0010	Rn	Rm	01MD	MOV.B Rm, @ Rn	MOV.W Rm, @ Rn	MOV.L Rm, @ Rn	DIV0S Rm, Rn
0010	Rn	Rm	10MD	TST Rm, Rn	AND Rm, Rn	XOR Rm, Rn	OR Rm, Rn
0010	Rn	Rm	11MD	CMP/STR Rm, Rn	XTRCT Rm, Rn	MULU.W Rm, Rn	MULSW Rm, Rn
0011	Rn	Rm	00MD	CMP/EQ Rm, Rn		CMP/HS Rm, Rn	CMP/GE Rm, Rn
0011	Rn	Rm	01MD	DIV1 Rm, Rn	DMULU.L Rm, Rn	CMP/HI Rm, Rn	CMP/GT Rm, Rn
0011	Rn	Rm	10MD	SUB Rm, Rn		SUBC Rm, Rn	SUBV Rm, Rn
0011	Rn	Rm	11MD	ADD Rm, Rn	DMULS.L Rm, Rn	ADDC Rm, Rn	ADDV Rm, Rn
0100	Rn	Fx	0000	SHLL Rn	DT Rn	SHAL Rn	
0100	Rn	Fx	0001	SHLR Rn	CMP/PZ Rn	SHAR Rn	
0100	Rn	Fx	0010	STS.L MACH, @ Rn	STS.L MACL, @ Rn	STS.L PR, @ Rn	

命令コード				Fx: 0000	Fx: 0001	Fx: 0010	Fx: 0011 ~ 1111
MSB	LSB			MD: 00	MD: 01	MD: 10	MD: 11
0100	Rn	00MD	0011	STC.L SR, @ Rn	STC.L GBR, @ Rn	STC.L VBR, @ Rn	STC.L SSR, @ Rn
0100	Rn	01MD	0011	STC.L SPC, @ Rn			
0100	Rn	10MD	0011	STC.L R0_BANK, @ Rn	STC.L R1_BANK, @ Rn	STC.L R2_BANK, @ Rn	STC.L R3_BANK, @ Rn
0100	Rn	11MD	0011	STC.L R4_BANK, @ Rn	STC.L R5_BANK, @ Rn	STC.L R6_BANK, @ Rn	STC.L R7_BANK, @ Rn
0100	Rn	Fx	0100	ROTL Rn		ROTCL Rn	
0100	Rn	Fx	0101	ROTR Rn	CMP/PL Rn	ROTCR Rn	
0100	Rm	Fx	0110	LDS.L @Rm+, MACH	LDS.L @Rm+, MACL	LDS.L @Rm+, PR	
0100	Rm	00MD	0111	LDC.L @Rm+, SR	LDC.L @Rm+, GBR	LDC.L @Rm+, VBR	LDC.L @Rm+, SSR
0100	Rm	01MD	0111	LDC.L @Rm+, SPC			
0100	Rm	10MD	0111	LDC.L @Rm+, R0_BANK	LDC.L @Rm+, R1_BANK	LDC.L @Rm+, R2_BANK	LDC.L @Rm+, R3_BANK
0100	Rm	11MD	0111	LDC.L @Rm+, R4_BANK	LDC.L @Rm+, R5_BANK	LDC.L @Rm+, R6_BANK	LDC.L @Rm+, R7_BANK
0100	Rn	Fx	1000	SHLL2 Rn	SHLL8 Rn	SHLL16 Rn	
0100	Rn	Fx	1001	SHLR2 Rn	SHLR8 Rn	SHLR16 Rn	
0100	Rm	Fx	1010	LDS Rm, MACH	LDS Rm, MACL	LDS Rm, PR	
0100	Rm/ Rn	Fx	1011	JSR @Rm	TAS.B @Rn	JMP @Rm	
0100	Rn	Rm	1100	SHAD Rm, Rn			
0100	Rn	Rm	1101	SHLD Rm, Rn			
0100	Rm	00MD	1110	LDC Rm, SR	LDC Rm, GBR	LDC Rm, VBR	LDC Rm, SSR
0100	Rm	01MD	1110	LDC Rm, SPC			
0100	Rm	10MD	1110	LDC Rm, R0_BANK	LDC Rm, R1_BANK	LDC Rm, R2_BANK	LDC Rm, R3_BANK
0100	Rm	11MD	1110	LDC Rm, R4_BANK	LDC Rm, R5_BANK	LDC Rm, R6_BANK	LDC Rm, R7_BANK
0100	Rn	Rm	1111	MAC.W @Rm+, @Rn+			
0101	Rn	Rm	disp	MOV.L @(disp:4, Rm), Rn			
0110	Rn	Rm	00MD	MOV.B @Rm, Rn	MOV.W @Rm, Rn	MOV.L @Rm, Rn	MOV Rm, Rn
0110	Rn	Rm	01MD	MOV.B @Rm+, Rn	MOV.W @Rm+, Rn	MOV.L @Rm+, Rn	NOT Rm, Rn
0110	Rn	Rm	10MD	SWAP.B Rm, Rn	SWAP.W Rm, Rn	NEGC Rm, Rn	NEG Rm, Rn
0110	Rn	Rm	11MD	EXTU.B Rm, Rn	EXTU.W Rm, Rn	EXTS.B Rm, Rn	EXTS.W Rm, Rn
0111	Rn	imm		ADD # imm : 8, Rn			

2. CPU

命令コード				Fx: 0000	Fx: 0001	Fx: 0010	Fx: 0011 ~ 1111
MSB		LSB		MD: 00	MD: 01	MD: 10	MD: 11
1000	00MD	Rn	disp	MOV.B R0, @ (disp: 4, Rn)	MOV.W R0, @ (disp: 4, Rn)		
1000	01MD	Rm	disp	MOV.B @ (disp:4, Rm) , R0	MOV.W @ (disp: 4, Rm) , R0		
1000	10MD	imm/disp		CMP/EQ #imm:8, R0	BT disp: 8		BF disp: 8
1000	11MD	imm/disp			BT/S disp: 8		BF/S disp: 8
1001	Rn	disp		MOV.W @ (disp : 8, PC) , Rn			
1010	disp			BRA disp: 12			
1011	disp			BSR disp: 12			
1100	00MD	imm/disp		MOV.B R0, @ (disp: 8, GBR)	MOV.W R0, @ (disp: 8, GBR)	MOV.L R0, @ (disp: 8, GBR)	TRAPA #imm: 8
1100	01MD	disp		MOV.B @ (disp: 8, GBR) , R0	MOV.W @ (disp: 8, GBR) , R0	MOV.L @ (disp: 8, GBR) , R0	MOVA @ (disp: 8, PC) , R0
1100	10MD	imm		TST #imm: 8, R0	AND #imm: 8, R0	XOR #imm: 8, R0	OR #imm: 8, R0
1100	11MD	imm		TST.B #imm: 8, @ (R0, GBR)	AND.B #imm: 8, @ (R0, GBR)	XOR.B #imm: 8, @ (R0, GBR)	OR.B #imm: 8, @ (R0, GBR)
1101	Rn	disp		MOV.L @ (disp: 8, PC) , Rn			
1110	Rn	imm		MOV #imm:8, Rn			
1111	*****						

【注】 なお、詳細については「SH-3/SH-3E/SH3-DSP プログラミングマニュアル」を参照してください。

3. DSP 演算ユニット

3.1 DSP 拡張機能

本 LSI では、DSP ユニットおよび DSP ユニットに直結された X/Y メモリを内蔵しており、それらを制御する拡張命令セットが提供されています。拡張される命令セットは、次の 4 つのグループに分けられます (図 3.1)。

(1) CPU 用の拡張システム制御命令

DSP 拡張機能が有効になると CPU 用の拡張システム制御命令として以下の命令が利用できるようになります。

- リピートループを制御するための命令や、リピートループ制御用のコントロールレジスタに対するアクセス命令が追加されます。ゼロオーバーヘッドリピート制御機構を使用することによりループ構造のプログラムを効率的に実行することができるようになります。本機能に関しては、「3.3 CPU 拡張命令」で詳しく説明します。
- モジュロアドレッシングを制御する命令、およびコントロールレジスタをアクセスする命令が追加されます。循環構造を持つデータ構造にアクセスできる機能をモジュロアドレッシングと呼びます。これらの命令については、「3.4 DSP データ転送命令」で詳しく説明します。
- DSP ユニットのレジスタに対するアクセス命令が追加されます。DSP ユニットのいくつかのレジスタを CPU のシステムレジスタであるかのように操作することが可能になります。これらの命令については、「3.4 DSP データ転送命令」で詳しく説明します。

(2) DSP ユニットのレジスタと内蔵 X/Y メモリ間のデータ転送命令

DSP ユニットのレジスタと内蔵 X/Y メモリ間のデータ転送命令は、ダブルデータ転送命令とも呼ばれます。このグループの命令のコード長は、CPU 命令と同様に 16 ビットです。DSP ユニットと DSP ユニットに直結された内蔵 X/Y メモリのデータ転送を行います。このグループの命令は、DSP ユニット用の演算命令と組み合わせて記述することが可能です。このグループの命令については、「3.4 DSP データ転送命令」で詳しく説明します。

(3) DSP ユニットのレジスタと全論理アドレス空間の間のデータ転送命令

DSP ユニットのレジスタと全論理アドレス空間の間のデータ転送命令は、シングルデータ転送命令とも呼ばれます。このグループの命令のコード長は、CPU 命令同様に 16 ビットです。DSP ユニットと全論理アドレス空間の間でデータ転送を行います。このグループの命令については、「3.4 DSP データ転送命令」で詳しく説明します。

3. DSP 演算ユニット

(4) DSP ユニット用の命令

DSP ユニット用の命令は、DSP 命令と呼ばれます。この命令は、DSP ユニットを用いたデジタル信号処理演算を高速に実行するために用意されています。この命令のコード長は、32 ビットです。DSP 命令のフィールドは、A フィールドと B フィールドに分かれています。A フィールドにはダブルデータ転送命令の機能を記述することができ、B フィールドには ALU 演算命令、および乗算命令を記述することができます。記述されたこれらの命令は並列に実行され、同時に 4 つの処理 (ALU 演算、乗算、および 2 つのデータ転送) を実行することができます。

このグループの命令については、「3.5 DSP データ演算命令」で詳しく説明します。

- 【注】
1. 32 ビット命令コードは、16 ビットの命令コードが 2 個連続したものとして扱われます。このため、32 ビット命令もワード境界から配置することができます。32 ビットの命令コードは、メモリ上にワードサイズ単位で、上位ワード、下位ワードの順番に格納してください。
 2. リトルエンディアンの場合は、命令コードの上位ワードと下位ワードがそれぞれワード単位でアクセスされるものとして、メモリ上に格納してください。

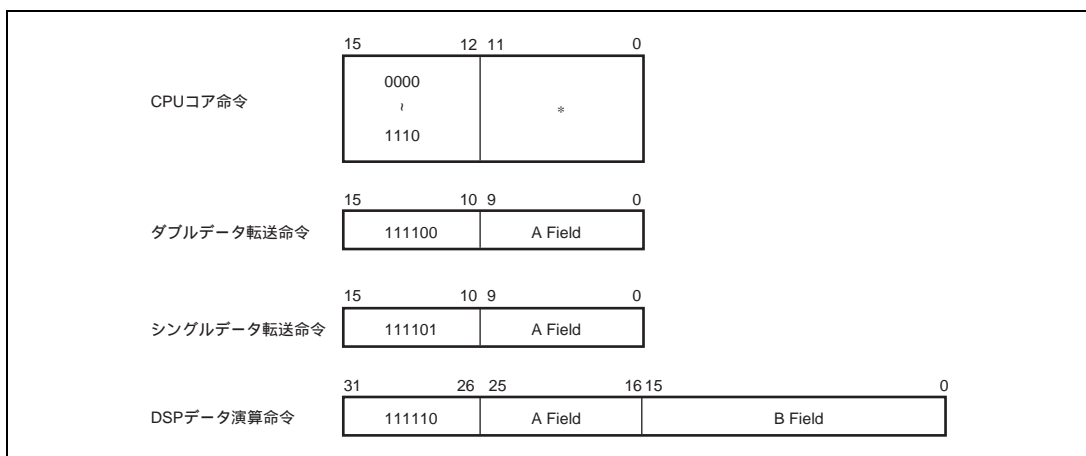


図 3.1 DSP 命令の命令形式

3.2 DSP モードのリソース

3.2.1 処理モード

CPU の処理モードは、ステータスレジスタ (SR) のモードビット (MD) および DSP ビット (DSP) により、次の表のように拡張されます。

MD ビット	DSP ビット	処理モード	説 明	
			特権保護されたリソースのアクセスや 特権命令の実行	DSP 拡張機能
0	0	ユーザモード	不可	無効
0	1	ユーザ DSP モード	不可	有効
1	0	特権モード	可能	無効
1	1	特権 DSP モード	可能	有効

このように、DSP ビットによる DSP 拡張機能の制御は、MD ビットによる制御と独立に作用します。ただし、DSP ビットは特権モードでのみ値の変更が可能であり、DSP モードの変更を行うには特権モードや特権 DSP モードへの遷移が必要になります。

3.2.2 DSP モードのメモリマップ

DSP モードのときは、論理アドレス空間の P2 領域 (H'A0000000 ~ H'BFFFFFFF) の一部 (H'A5000000 ~ H'A5FFFFFFF) がユーザ DSP モードでもアクセス可能になります (表 3.1)。ユーザ DSP モードでアクセスするときは、この領域を U_{xy} 領域と呼びます。X/Y メモリは、この領域に配置され、ユーザ DSP モードでもアクセスが可能です。

表 3.1 論理アドレス空間

アドレス範囲	領域名称	保護モード	説 明
H'A5000000 ~ H'A5FFFFFFF	P2/U _{xy}	特権モード または DSP モード	16M バイト論理空間、キャッシング不可。 特権モードおよび DSP モード (特権 DSP モードおよびユーザ DSP モード) でアクセス可能。

3.2.3 CPU のレジスタセット

DSP モードでは、CPU のステータスレジスタ (SR) に制御ビットが拡張され、リピートスタートレジスタ (RS)、リピートエンドレジスタ (RE)、およびモジュロレジスタ (MOD) の 3 つのコントロールレジスタが拡張されます (図 3.2)。

3. DSP 演算ユニット

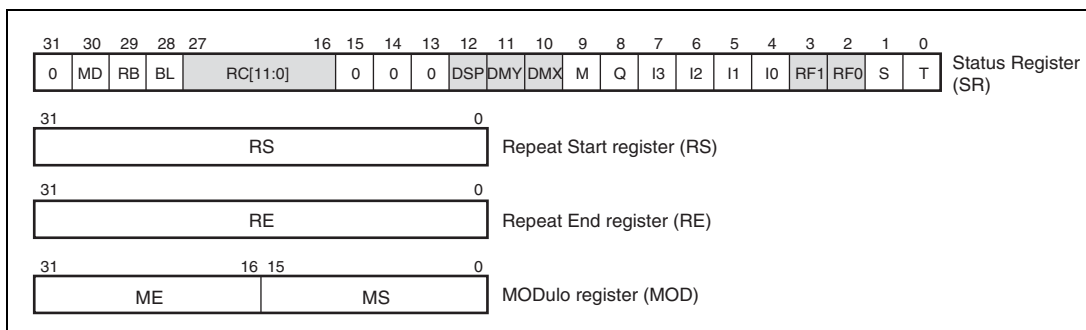


図 3.2 DSP モードでの CPU レジスタ

(1) ステータスレジスタ (SR) の拡張

DSP モードでは、SR に以下に示す制御ビットが拡張されます。これらのビットを DSP 拡張ビットと呼びます。DSP 拡張ビットは、DSP モードでのみ有効です。

ビット	ビット名	初期値	R/W	説明
31~28				拡張前の部分については、「第 2 章 CPU」を参照してください。
27~16	RC11~RC0	すべて 0	R/W	リピートカウンタ リピートカウンタは、リピート命令で制御されるリピートループの、残り実行回数を保持しています。このビットは、特権モード、特権 DSP モード、ユーザ DSP モードで更新可能です。リセット状態に遷移することにより 0 に初期化されます。例外処理状態に遷移しても値は変化しません。
15~13				拡張前の部分については、「第 2 章 CPU」を参照してください。
12	DSP	0	R/W	DSP ビット DSP ビットは、DSP 拡張機能の有効および無効を制御します。このビットに 1 を書き込むことで DSP 拡張機能が有効になります。このビットは特権モードおよび特権 DSP モードでのみ更新可能で、ユーザ DSP モードでは更新できません。リセット状態に遷移することにより 0 に初期化されます。例外処理状態に遷移しても値は変化しません。
11	DMY	0	R/W	モジュロ制御 モジュロ制御ビットは、X/Y メモリへのアクセス命令でのモジュロアドレッシングの有効・無効を制御します。これらのビットは特権モード、特権 DSP モード、ユーザ DSP モードで更新可能です。リセット状態に遷移することにより 0 に初期化されます。例外処理状態に遷移しても値は変化しません。
10	DMX	0	R/W	
9~4				拡張前の部分については、「第 2 章 CPU」を参照してください。
3	RF1	0	R/W	リピートフラグ リピートフラグビットは、リピート制御命令によって使用されます。これらのビットは、特権モード、特権 DSP モード、およびユーザ DSP モードで更新可能です。リセット状態に遷移することにより、0 に初期化されます。例外処理状態に遷移しても値は変化しません。
2	RF0	0	R/W	
1~0				拡張前の部分については、「第 2 章 CPU」を参照してください。

【注】 ステータスレジスタに値を代入する場合、リザーブビット部分 (ビット 31、15~13) には必ず 0 を書きこんでください。

(2) リピートスタートレジスタ (RS)

RS には、ゼロオーバーヘッドリピート制御機構で制御されるリピートループの先頭の命令アドレスが格納されます。RS は、DSP モードでアクセスできます。リセット状態に遷移したときの初期値は、不定です。例外処理状態に遷移しても値は変化しません。

(3) リピートエンドレジスタ (RE)

RE には、ゼロオーバーヘッドリピート制御機構で制御されるリピートループの最終命令の実行を検出するためのアドレスが格納されます。RE は、DSP モードでのみアクセスできます。リセット状態に遷移することにより、0 に初期化されます。例外処理状態に遷移しても値は変化しません。

(4) モジュールレジスタ (MOD)

MOD には、上位 16 ビットにモジュールアドレッシングの終了アドレスを、下位 16 ビットにモジュールアドレッシングの開始アドレスを格納します。MOD レジスタの上位 16 ビットを ME レジスタ、下位 16 ビットを MS レジスタと表現する場合があります。MOD は、DSP モードでのみアクセスできます。リセット状態に遷移したときの初期値は、不定です。例外処理状態に遷移しても値は変化しません。

これらのレジスタは、コントロールレジスタへのロード (LDC) およびストア (STC) 命令でアクセスできます。RS、RE、および MOD に対する LDC と STC 命令は、特権 DSP モードとユーザ DSP モードで使用可能になります。

SR に対する LDC と STC 命令は、本来 MD ビットが 1 の場合にのみ使用可能な命令ですが、ユーザ DSP モードにおいても使用可能になります。ただし、値を書き替えられる制御ビットは、RC11 ~ RC0、RF1 ~ RF0、DMX、および DMY に限定されます。LDC と STC 命令使用時の SR の詳細は、下記のとおりです。

- ユーザモード時は、SR に対する LDC 命令と STC 命令は不当命令例外となります。
- 特権モードと特権 DSP モード時は、SR の全ビットが更新できます。
- ユーザ DSP モード時は、SR は STC 命令で読み出し可能です。

ユーザ DSP モード時は、SR への LDC 命令発行は可能ですが、DSP 拡張ビットのみ更新できます。

3. DSP 演算ユニット

表 3.2 各処理モードにおける SR の各ビットの動作説明

ビット	特権モード	ユーザモード	特権 DSP モード	ユーザ DSP モード	専用命令による DSP 関連ビットへのアクセス	リセット後の初期値
	MD=1 & DSP=0	MD=0 & DSP=0	MD=1 & DSP=1	MD=0 & DSP=1		
MD	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		1
RB	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		1
BL	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		1
RC [11:0]	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : OK	SETRC 命令	B'000000000000
DSP	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		0
DMY	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : OK		0
DMX	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : OK		0
Q	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		X
M	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		X
I[3:0]	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		B'1111
RF[1:0]	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : OK	SETRC 命令	X
S	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		X
T	S : OK、L : OK	S、L : 不当命令	S : OK、L : OK	S : OK、L : NG		X

S : STC 命令

L : LDC 命令

OK : STC と LDC 動作を許可します。

不当命令 : 実行すると不当命令例外が発生します。

NG : 前の値を保持します。変化しません。

X : 不定

例外処理状態に遷移すると、DSP モードでの拡張ビットも含めた SR の全制御ビットが SSR へ待避されます。復帰時には、拡張 DSP ビットも含めて全制御ビットを回復してください。リポート制御を例外処理前の状態に復帰する必要がある場合には、RS と RE レジスタを例外処理前の値に回復してください。モジュロ制御を例外処理前の状態に復帰する必要がある場合には、MOD レジスタを例外処理前の値に回復してください。

3.2.4 DSP レジスタ

DSP ユニットの、8 つのデータレジスタ (A0、A1、X0、X1、Y0、Y1、M0、および M1) と 1 つのステータスレジスタ (DSR) を持っています。図 3.3 に DSP レジスタの構成を示します。これらは、すべて 32 ビット幅のレジスタです。レジスタ A0 および A1 は、8 ビット幅のガードビットレジスタ (A0G および A1G) と組み合わせ、40 ビット幅のレジスタとしても使用されます。DSR は、DSP データ演算結果の状態 (ゼロ、負、など) を保持し、また CPU の T ビットに類似した DC ビットを持っています。各ビットの詳細は、「3.5 DSP データ演算命令」を参照してください。

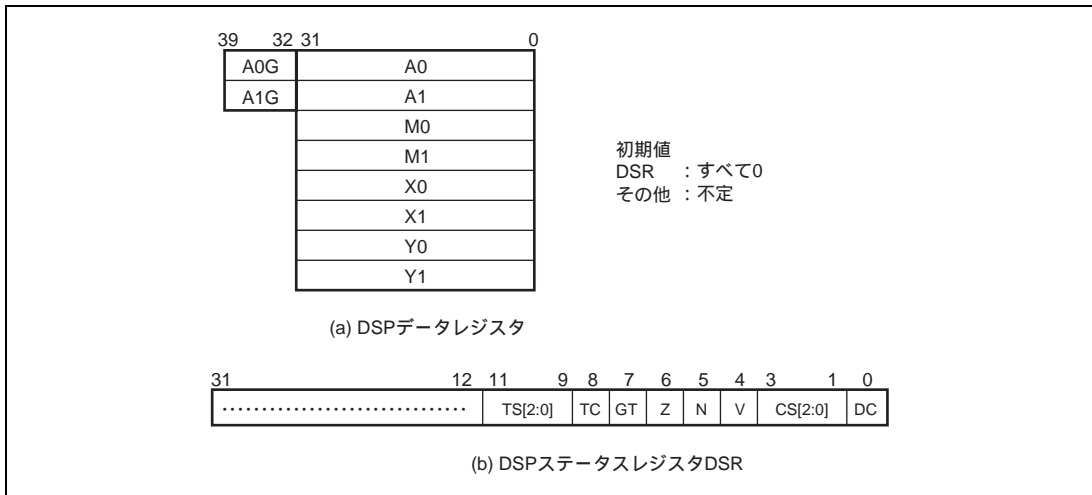


図 3.3 DSP レジスタの構成

3.3 CPU 拡張命令

3.3.1 リピート制御命令

DSP モードでは、リピートループを効率的に実行するための特別な制御機能（ゼロオーバーヘッドリピート制御機構）が使用できます。この機能を使用することで、比較命令や分岐命令のオーバーヘッドなしにループ構造のプログラムを実行することができます。リピートループを記述するには、表 3.4 に示すリピート制御命令、表 3.5 に示すリピート制御マクロ、表 3.6 に示す DSP モード拡張システム制御命令などを使います。

(1) リピートループのプログラム例

以下にリピートループを使用したプログラム例を示します。

- (例1) 4命令以上のリピートループ

```

LDRS RptStart      ; RS レジスタに、リピート開始命令のアドレスをセットします。
LDRE RptDtct+4    ; RE レジスタに、リピート検出命令のアドレス+4 をセットします。
SETRC #4          ; SR レジスタの RC[11:0] に、リピート回数(4) をセットします。
Instr0            ; SETRC 命令から【リピート開始命令】までには少なくとも1個の
                  ; 命令が必要です。

RptStart:        instr1      ; 【リピート開始命令】
                .....      ;
                .....      ;

RptDtct:         instr(N-3)  ; 【リピート検出命令】リピート最終命令から3命令前の命令がリ
                  ; ピート検出命令になります。

RptEnd2:         instr(N-2)  ;
RptEnd1:         instr(N-1)  ;
RptEnd:          instrN     ; 【リピート最終命令】

```

3. DSP 演算ユニット

このプログラムの例では、RptStart のアドレスにある命令 (instr1) から RptEnd のアドレスに配置された命令 (instrN) までが 4 回繰り返し実行されます。繰り返し実行されるプログラム範囲をリピートループと呼び、その開始と終了命令をそれぞれリピート開始命令、リピート最終命令と呼びます。CPU は、命令を順次実行しながら、特定の命令の実行完了を検出することにより、リピートループの制御を開始します。この命令をリピート検出命令と呼びます。4 命令以上のリピートループでは、リピート最終命令から 3 命令前の命令がリピート検出命令になります。4 命令のリピートループでは RptStart 命令と RptDct 命令は同じ命令になります。

リピートループの制御には、DSP モードで拡張されるコントロールレジスタ RE、RS、および SR レジスタの RC[11:0]、RF[1:0]ビットが使用されます。また、これらのレジスタに値を設定するには、それぞれ LDRE、LDRS、SETRC 命令を使用します。

- リピートエンドレジスタ (RE)

REには、LDRE命令で値をセットします。リピート検出命令のアドレス+4を設定します。4命令以上のリピートループでは、リピート最終命令から3命令前の命令がリピート検出命令になります。3命令以下のリピートループについては、後述します。

- リピートスタートレジスタ (RS)

RSには、LDRS命令で値をセットします。4命令以上のリピートループでは、リピート開始命令のアドレスをセットします。3命令以下のリピートループでは特殊なアドレスを設定しますが、これについては後述します。

- リピートカウンタ (SRのRC[11:0])

SETRC命令により、繰り返し回数をセットします。リピートループ実行中は、繰り返しの残り回数を保持しています。

- リピートフラグ (SRのRF[1:0])

RFは、SETRC命令実行時に、RSおよびREレジスタに設定された値の関係から自動的に設定され、リピートループの命令数の情報を保持しています。通常、ユーザが値を変更することはありません。

CPU は、RE レジスタとプログラムカウンタ (PC) の値を常に比較しながら命令を実行します。PC は、命令アドレス+4 の値を保持していますので、リピート検出命令実行時に両者が一致することで、リピート検出命令が検出されます。リピート検出命令の実行が分岐せずに完了し、かつ RC[11:0]>0 である場合に、リピート制御が行われます。リピート最終命令の実行完了時に RC[11:0]>=2 であれば、RC[11:0]を 1 減じた後、RS レジスタに設定されたアドレスへ制御を移します。RC[11:0]==1 (または 0) であれば RC[11:0]を 0 にした後、リピート最終命令の次の命令へ制御を移します。

リピートループを構成する命令数が 3、2、または 1 命令の場合のプログラム例を (例 2) (例 3) (例 4) に示します。これらの場合、リピート検出命令はリピート開始命令の直前の命令になります。また、RS レジスタにはリピート命令数を示す特殊な値を設定します。

- (例2) 3命令リピートループ

```

LDRS RptDtct+4 ; RS レジスタに、リピート検出命令のアドレス+4 をセットします。
LDRE RptDtct+4 ; RE レジスタに、リピート検出命令のアドレス+4 をセットします。
SETRC #4 ; SR レジスタの RC[11:0] に、リピート回数 (4) をセットします。
; SETRC 命令実行時に、RE-RS==0 であれば 3 命令リピートと認識
; されます。

RptDtct: instr0 ; 【リピート検出命令】リピート開始命令の直前の命令がリピート検
; 出命令になります。LDRC 命令とリピート開始命令の間には少なく
; とも 1 命令が必要です。

RptStart: instr1 ; 【リピート開始命令】
; instr2 ;
RptEnd: instr3 ; 【リピート最終命令】

```

- (例3) 2命令リピートループ

```

LDRS RptDtct+6 ; RS レジスタに、リピート検出命令のアドレス+6 をセットします。
LDRE RptDtct+4 ; RE レジスタに、リピート検出命令のアドレス+4 をセットします。
SETRC #4 ; SR レジスタの RC[11:0] に、リピート回数 (4) をセットします。
; SETRC 実行時に RE-RS==2 であれば 2 命令リピートと認識され
; ます。

RptDtct: instr0 ; 【リピート検出命令】リピート開始命令の直前の命令がリピート検
; 出命令になります。LDRC 命令とリピート開始命令の間には少なく
; とも 1 命令が必要です。

RptStart: instr1 ; 【リピート開始命令】
RptEnd: instr2 ; 【リピート最終命令】

```

- (例4) 1命令リピートループ

```

LDRS RptDtct+8 ; RS レジスタに、リピート検出命令のアドレス+8 をセットします。
LDRE RptDtct+4 ; RE レジスタに、リピート検出命令のアドレス+4 をセットします。
SETRC #4 ; SR レジスタの RC[11:0] に、リピート回数 (4) をセットします。
; SETRC 実行時に RE-RS==4 であれば 1 命令リピートと認識され
; ます。

RptDtct: instr0 ; 【リピート検出命令】リピート開始命令の直前の命令がリピート検
; 出命令になります。LDRC 命令とリピート開始命令の間には少なく
; とも 1 命令が必要です。

RptStart:
RptEnd: instr1 ; 【リピート開始命令】==【リピート最終命令】

```

3、2、および 1 命令リピートの場合には、RS レジスタにリピートループ中の命令を示すための特殊なアドレスを設定します。SETRC 命令を実行した際に、RE-RS が計算され、この結果が 0、-2、-4 のとき、それぞれ 3 命令、

3. DSP 演算ユニット

2 命令、1 命令のリピートループとして認識されます。

リピート検出命令の実行が分岐せずに完了し、かつ RC[11:0]>0 である場合には、リピート検出命令の次の命令をリピート開始命令として、認識されたリピート命令数分の命令を繰り返し実行します。リピート終了命令実行完了時に RC[11:0]>=2 であれば、RC[11:0]を 1 減じた後、リピート開始命令へ制御を移します。RC[11:0]=1 (または 0) であれば RC[11:0]を 0 にした後、リピート最終命令の次の命令へ制御を移します。

【注】 RE-RS の値が正の場合に、CPU は 4 命令以上のリピートループと認識し、リピートループを制御します (4 命令以上のリピートループの場合、RE-RS の値は常に正の値になります。(例 1) のプログラム例を参考にしてください)。RE-RS の値が正の値、0、-2、-4 以外の値になった場合の動作は保証しません。

表 3.3 に、RS、RE に設定すべきアドレスを纏めます。

表 3.3 RS および RE のアドレス設定ルール

レジスタ	リピートループ中の命令数			
	1	2	3	>=4
RS	RptStart0 +8	RptStart0 +6	RptStart0 +4	RptStart
RE	RptStart0 +4	RptStart0 +4	RptStart0 +4	RptEnd3+4

【注】 ここでは、次のラベルを使用しています。

RptStart : リピート開始命令のアドレス

RptStart0 : リピート開始命令の 1 命令前の命令アドレス

RptEnd3 : リピート最終命令の 3 つ前の命令アドレス

(2) リピート制御命令およびリピート制御マクロ

リピートループを記述するには、前項 (1) で例示したように、LDRS および LDRE 命令でそれぞれ RS と RE レジスタに適切なアドレスを設定した後、SETRC 命令でリピート回数を指定してください。SETRC のオペランドとしては、8 ビットのイミディエート値または汎用レジスタが使用できます。RC に 256 を超える値を設定するには、SETRC Rm タイプの命令を使用してください。

表 3.4 リピート制御命令

命令	動作	実行 ステート
LDRS @(disp,PC)	(disp × 2+PC)を算出し、RS レジスタに値を設定します。	1
LDRE @(disp,PC)	(disp × 2+PC)を算出し、RE レジスタに値を設定します。	1
SETRC #imm	8 ビット定数 imm を SR レジスタの RC[11:0]に設定し、SR レジスタの RF[1:0]にリピート命令数を示す情報を設定します。RC[11:0]には、0 から 255 までの値が設定できます。	1
SETRC Rm	Rm レジスタの[11:0]を SR レジスタの RC[11:0] に設定し、SR レジスタの RF[1:0]にリピート命令数を示す情報を設定します。RC[11:0]には、0 から 4095 までの値が設定できます。	1

RS および RE レジスタには、表 3.3 に示したルールに従って適切なアドレスを設定する必要があります。SH アセンブラでは、この問題を処理するためにリピート制御マクロ (REPEAT) が提供されています。

表 3.5 リピート制御マクロ

命令	動作	実行 ステート
REPEAT RptStart, RptEnd, #imm	RptStart をリピート開始命令、RptEnd をリピート最終命令とし、8 ビットのイミディエート値 imm をリピート回数とするリピートループを設定します。適切に変換された LDRS、LDRE、および SETRC の 3 命令に展開されます。	3
REPEAT RptStart, RptEnd, Rm	RptStart をリピート開始命令、RptEnd をリピート最終命令とし、Rm の[11:0]をリピート回数とするリピートループを設定します。適切に変換された LDRS、LDRE、および SETRC の 3 命令に展開されます。	3

リピート制御マクロを使用することで、前述した (例 1) ~ (例 4) は、それぞれ次に示す (例 5) ~ (例 8) のように簡略に記述することができます。

- (例 5) 4命令以上のリピートループの記述例... (例 1) に示した命令列に展開されます。

```

REPEAT RptStart, RptEnd, #4
    instr0          ;
RptStart: instr1   ; 【リピート開始命令】
    .....         ;
    .....         ;
    instr(N-3)     ; 【リピート検出命令】
    instr(N-2)     ;
    instr(N-1)     ;
RptEnd:   instrN   ; 【リピート最終命令】

```

- (例 6) 3命令リピートループの記述例... (例 2) に示した命令列に展開されます。

```

REPEAT RptStart, RptEnd, #4
    instr0          ; 【リピート検出命令】
RptStart: instr1   ; 【リピート開始命令】
    instr2          ;
RptEnd:   instr3   ; 【リピート最終命令】

```

3. DSP 演算ユニット

- (例7) 2命令リピートループ... (例3) に示した命令列に展開されます。

```

REPEAT RptStart, RptEnd, #4
    instr0          ; 【リピート検出命令】
RptStart: instr1   ; 【リピート開始命令】
RptEnd:   instr2   ; 【リピート最終命令】

```

- (例8) 1命令リピートループ... (例4) に示した命令列に展開されます。

```

REPEAT RptStart, RptEnd, #4
    instr0          ; 【リピート検出命令】
RptStart:
RptEnd:   instr1   ; 【リピート開始命令】 == 【リピート最終命令】

```

DSP モードでは、RS および RE レジスタの値を操作するシステム制御命令 (LDC と STC 命令) が拡張されます。また、SR レジスタの RC[11:0] および RF[1:0] ビットは、SR レジスタに対する LDC と STC 命令で制御できます。これらの命令は、リピートループ中に例外を受け付ける場合に使用してください。RS と RE レジスタおよび SR レジスタの RC[11:0] と RF[1:0] ビットを退避した後、回復することでリピートループを再開することができます。ただし、リピートループ中の例外の受け付けには、制限事項があります。詳しくは、「3.3.1 (3) リピート制御中の制限事項」および「第 4 章 例外処理」を参照してください。

表 3.6 DSP モード拡張システム制御命令

命令	動作	実行ステート
STC RS,Rn	RS Rn	1
STC RE,Rn	RE Rn	1
STC.L RS,@-Rn	Rn-4 Rn, RS (Rn)	1
STC.L RE,@-Rn	Rn-4 Rn, RE (Rn)	1
LDC.L @Rn+,RS	(Rn) RS, Rn+4 Rn	4
LDC.L @Rn+,RE	(Rn) RE, Rn+4 Rn	4
LDC Rn,RS	Rn RS	4
LDC Rn,RE	Rn RE	4

(3) リピート制御中の制限事項

(a) リピート制御命令の配置

LDRS および LDRE 命令を実行した後に SETRC 命令を実行してください。また、SETRC 命令とリピート開始命令の間には少なくとも 1 命令が必要です。

(b) リピート検出命令に続く命令以降の不当命令

リピート検出命令の次の命令からリピート最終命令の間に、以下に示す命令を配置すると不当命令例外を発生します。

- 分岐命令

BRA、BSR、BT、BF、BT/S、BF/S、BSRF、RTS、BRAf、RTE、JSR、JMP、TRAPA

- リピート制御命令

SETRC、LDRS、LDRE

- SR、RS、REに対するロード命令

LDC Rn,SR、LDC @Rn+,SR、LDC Rn,RE、LDC @Rn+,RE、LDC Rn,RS、LDC @Rn+,RS

【注】 1~3命令のリピートループの場合はリピートループ中の全命令が、4命令以上のリピートループの場合はリピート最終命令を含む3命令がこの制約の範囲となります。

(c) リピートループ中の禁止命令(4命令以上のリピートループ)

4命令以上のリピートループのリピート開始命令からリピート検出命令までの間には、以下の命令を配置しないでください。配置した場合の動作は、保証されません。

- リピート制御命令

SETRC、LDRS、LDRE

- SR、RS、REに対するロード命令

LDC Rn,SR、LDC @Rn+,SR、LDC Rn,RE、LDC @Rn+,RE、LDC Rn,RS、LDC @Rn+,RS

【注】 多重のリピートループは、保障されません。最内部のループをリピート制御命令で記述し、外部のループはDTおよびBF/S命令等で実現してください。

(d) リピート検出命令の次命令以降への分岐および例外受理に関する制限

CPUがリピートループを認識するためには、リピート検出命令の実行が分岐せずに完了することが必要です。

したがって、リピート検出命令の次以降の命令に分岐した場合はリピートループが認識されず、リピート最終命令を実行してもリピート開始命令に制御が移りません。この場合は、RC[11:0]の値も変化しません。

- リピートループ中で条件分岐命令を使用する場合は、リピート検出命令以前の命令を分岐先に指定してください。
- リピートループ中でサブルーチンコールを使用する場合は、サブルーチンコール命令の遅延スロット命令がリピート検出命令より前になるように配置してください。

ここでの分岐には、例外処理ルーチンからの復帰を含みます。復帰アドレスがリピート検出命令の次命令以降になる例外が発生するとリピート制御が正しく復帰されません。このため、リピート検出命令からリピート最終命令の間では例外の受け付けを制限しています。割り込みなどのCPUが受け付けを保留し得る例外は、保留します。CPUが受け付けを保留できない例外の場合は例外処理状態へ遷移しますが、もとの実行状態へ復帰することができません。詳細は、「第4章 例外処理」を参照してください。

3. DSP 演算ユニット

- 【注】
1. リピート検出命令が TRAPA 命令である場合は、復帰アドレスがリピート検出命令の次命令となり、リピート制御を正しく復帰できません。TRAPA 命令は、復帰アドレスがリピート検出命令より後のアドレスとなるため、リピート検出命令より前に配置してください。
 2. リピート検出命令以降の命令が SLEEP 命令である場合は、低消費電力状態への遷移および割り込み等による例外の受け付けは正しく行われますが、リピート制御を正しく復帰できません。SLEEP 命令は、リピート検出命令より前に配置してください。

(e) リピート検出命令からの分岐

リピート検出命令が遅延分岐命令の遅延スロット命令である場合や分岐命令そのものである場合は、分岐命令で分岐しなかったときにリピートループが認識されます。分岐したときはリピート制御は行われず、分岐先命令を実行します。

(f) リピート制御中のプログラムカウンタ

RC[11:0]>=2 の場合、リピート検出命令の 2 命令以降ではプログラムカウンタ (PC) の値が通常とは異なります。1、2、3 命令リピートループの場合、リピート検出命令の次の命令 (リピート開始命令) では通常と同じ値 (= 命令アドレス+4) を指しますが、その次の命令以降でもリピート開始命令の PC の値と同じ値を取り続けます。4 命令以上のリピートループの場合は、リピート検出命令の次の命令までは通常と同じ値を取りますが、その次の命令では RS、さらにその次の命令 (リピート最終命令) では RS+2 の値を取ります。これらの通常と異なった PC の設定値に関しては保証されません。

このため、リピート検出命令の 2 命令以降では PC 相対アドレッシング命令が正しく動作せず、その結果を保証しません。

- PC 相対アドレッシング命令

MOVA @(disp, PC), Rn

MOV.W @(disp, PC), Rn

MOV.L @(disp, PC), Rn

(MOV #imm, Rn をアセンブラが、MOV.W @(disp, PC), Rn や MOV.L @(disp, PC), Rn に展開する場合も含まれます。)

表 3.7 リピート制御中の PC の値 (RC[11:0]>=2 の場合)

ラベル	リピートループ中の命令数			
	1	2	3	>=4
RptDtct	RptDtct+4	RptDtct+4	RptDtct+4	RptDtct+4
RptDtct1	RptDtct1+4	RptDtct1+4	RptDtct1+4	RptDtct1+4
RptDtct2	-	RptDtct1+4	RptDtct1+4	RS
RptDtct3	-	-	RptDtct1+4	RS+2

【注】 ここでは、次のラベルを使用しています。

RptDtct : リピート検出命令のアドレス

RptDtct1 : リピート検出命令の 1 命令後の命令アドレス。(1、2、および 3 命令リピートでは RptStart です。)

RptDct2 : リピート検出命令の 2 命令後の命令アドレス。

RptDct3 : リピート検出命令の 3 命令後の命令アドレス。

(g) リピートカウンタとリピート制御

CPU は、常に RE と PC との比較を行いながらプログラムを実行しています。SR レジスタの RC[11:0] が 0 以外で PC が RE に一致すると、リピート制御が機能します。

- RC>=2 の場合は、リピート最終命令実行後、リピート開始命令に制御が移行します。リピート最終命令の実行完了により RC が 1 減じられます。(a) ~ (f) の制限がかかります。
- RC==1 の場合は、リピート最終命令実行後、RC が 0 になり、後続命令へ制御が移行します。RC==1 の場合も (a) ~ (f) の制限がかかります。
- RC==0 の場合は、リピート検出命令を実行しても、リピート制御は機能しません。リピートループは通常の命令列として 1 回実行され、リピート最終命令を実行してもリピート開始命令へは制御が移行しません。

3.3.2 拡張リピート制御命令

「3.3.1 リピート制御命令」で提供されるリピート制御機構には、いくつかの制約事項があります。この制約を軽減するため、表 3.8 に示すリピート制御命令が拡張されています。これらの命令は、従来の SH-DSP アーキテクチャには存在しない命令で、互換性を重視する場合には従来のリピート制御命令を使用してください。この意味で従来のリピート制御命令を互換リピート制御命令と呼びます。

(1) 拡張リピート制御命令のプログラム例

以下に拡張リピート制御命令を使用したプログラム例を示します。

• (例1) 4命令以上のリピートループ

```

LDRS RptStart      ; RS レジスタに、リピート開始命令のアドレスをセットします。
LDRE RptEnd        ; RE レジスタに、リピート最終命令のアドレスをセットします。
LDRC #4            ; SR レジスタの RC[11:0] に、リピート回数 (4) をセットします。
instr0              ; LDRC 命令から【リピート開始命令】までには少なくとも 1 個の命令
                    ; が必要です。

RptStart: instr1    ; 【リピート開始命令】
                ;
                ;
instr(N-3)         ; 【リピート検出命令】リピート最終命令から 3 命令前の命令がリ
                    ; ピート検出命令になります。

instr(N-2)         ;
instr(N-1)         ;

RptEnd: instrN     ; 【リピート最終命令】

```

3. DSP 演算ユニット

- (例2) 3命令リピートループ

```
LDRS RptStart      ; RS レジスタに、リピート開始命令のアドレスをセットします。
LDRE RptEnd        ; RE レジスタに、リピート最終命令のアドレスをセットします。
LDRC #4            ; SR レジスタの RC[11:0]に、リピート回数(4)をセットします。
instr0             ; 【リピート検出命令】リピート開始命令の直前の命令がリピート検
                  ; 出命令になります。LDRC 命令とリピート開始命令の間には少なく
                  ; とも1命令が必要です。

RptStart:  instr1   ; 【リピート開始命令】
           instr2   ;
RptEnd:    instr3   ; 【リピート最終命令】
```

- (例3) 2命令リピートループ

```
LDRS RptStart      ; RS レジスタに、リピート開始命令のアドレスをセットします。
LDRE RptEnd        ; RE レジスタに、リピート最終命令のアドレスをセットします。
LDRC #4            ; SR レジスタの RC[11:0]に、リピート回数(4)をセットします。
instr0             ; 【リピート検出命令】リピート開始命令の直前の命令がリピート検
                  ; 出命令になります。LDRC 命令とリピート開始命令の間には少なく
                  ; とも1命令が必要です。

RptStart:  instr1   ; 【リピート開始命令】
RptEnd:    instr2   ; 【リピート最終命令】
```

- (例4) 1命令リピートループ

```
LDRS RptStart      ; RS レジスタに、リピート開始命令のアドレスをセットします。
LDRE RptEnd        ; RE レジスタに、リピート最終命令のアドレスをセットします。
LDRC #4            ; SR レジスタの RC[11:0]に、リピート回数(4)をセットします。
instr0             ; 【リピート検出命令】リピート開始命令の直前の命令がリピート検
                  ; 出命令になります。LDRC 命令とリピート開始命令の間には少なく
                  ; とも1命令が必要です。

RptStart:
RptEnd:    instr1   ; 【リピート開始命令】==【リピート最終命令】
```

拡張リピート制御命令では、リピート命令数によらず、RS レジスタにリピート開始命令のアドレスを、RE レジスタにリピート最終命令のアドレスを格納します。また、SETRC 命令の代わりに LDRC 命令を用いることで、拡張リピート制御が行われます。拡張リピート制御が行われている場合は、リピート最終命令を実行することでリピートループが認識できます。このため、分岐や例外に対する制約がありません。

(2) 拡張リピート制御命令

拡張リピート制御命令を用いてリピートループを記述するには、前節で例示したように、LDRS と LDRE 命令でそれぞれ RS と RE レジスタにそれぞれリピート開始命令およびリピート最終命令を指定します。LDRS と LDRE 命令は、互換リピート制御用の命令をそのまま使用します。その後、LDRC 命令でリピート回数を指定してください。LDRC のオペランドとしては、8 ビットのイミディエート値または汎用レジスタが使用できます。RC に 256 を超える値を設定するには、LDRC Rm タイプの命令を使用してください。

表 3.8 拡張リピート制御命令

命令	動作	実行ステート
LDRS @(disp,PC)	(disp × 2+PC)を算出し、RS レジスタに値を設定します。	1
LDRE @(disp,PC)	(disp × 2+PC)を算出し、RE レジスタに値を設定します。	1
LDRC #imm	8 ビット定数 imm を SR レジスタの RC[11:0]に設定し、SR レジスタの RF[1:0]にリピート命令数を示す情報を設定します。RC[11:0]には、0 から 255 までの値を設定できます。 拡張リピート制御中を示すために、RE レジスタのビット 0 に 1 がセットされます。	1
LDRC Rm	Rm レジスタの[11:0]を SR レジスタの RC[11:0]に設定し、SR レジスタの RF[1:0]にリピート命令数を示す情報を設定します。RC[11:0]には、0 から 4095 までの値を設定できます。 拡張リピート制御中を示すために、RE レジスタのビット 0 に 1 がセットされます。	1

LDRC 命令を実行することで、以後 CPU は拡張リピート制御を行います。拡張リピート制御中であることを示すために、LDRC 命令の実行により RE レジスタのビット 0 に 1 がセットされます。例外処理等で RE レジスタの値を変更する場合は、ビット 0 も正確に退避した後に回復してください。SR レジスタの RC[11:0]、DSP、RF[1:0] ビット、および RE と RS レジスタを退避した後に回復することで拡張リピート制御として処理を復帰することができます。

(3) 拡張リピート制御中の制限事項

(a) 拡張リピート制御命令の配置

LDRS および LDRE 命令を実行した後に LDRC 命令を実行してください。また、LDRC 命令とリピート開始命令の間には、少なくとも 1 命令が必要です。

(b) リピート最終命令での不当命令

リピート最終命令として次の命令を配置すると不当命令例外が発生します。

• 遅延分岐命令

BRA、BSR、BT/S、BF/S、BSRF、RTS、BRAf、RTE、JSR、JMP

• リピート制御命令

SETRC、LDRS、LDRE、LDRC

3. DSP 演算ユニット

- SR、RS、REに対するロード命令

LDC Rn,SR, LDC @Rn+,SR, LDC Rn,RE, LDC @Rn+,RE, LDC Rn,RS, LDC @Rn+,RS

【注】 非遅延の分岐命令 (BT、BF、TRAPA) は、最終命令として配置しても構いません。また、遅延分岐命令の遅延スロットがリポート最終命令となっても構いません。これらの場合、分岐したときも分岐しなかったときも RC[11:0]の値は 1 減じられます。分岐しなかったときはリポート開始命令へ、分岐したときには分岐先へ制御が移行します。

(c) リポートカウンタとリポート制御

CPU は、常にリポートエンドレジスタ (RE) と PC-4 (命令のアドレス) との比較を行いながらプログラムを実行しています。RE レジスタのビット 0 が 1 であり、SR レジスタの RC[11:0]が 0 以外で PC-4[31:1]と RE[31:1]が一致すると、拡張リポート制御が行われます。

- $RC \geq 2$ の場合は、リポート最終命令実行後、リポート開始命令に制御が移行します。リポート最終命令の実行完了により RC が 1 減じられます。
- $RC = 1$ の場合は、リポート最終命令実行後、RC が 0 になり、後続命令へ制御が移行します。
- $RC = 0$ の場合は、リポート検出命令を実行しても、リポート制御は機能しません。リポートループは通常の命令列として 1 回実行され、リポート最終命令を実行してもリポート開始命令へは制御が移行しません。

3.4 DSP データ転送命令

DSP モードでは、DSP ユニットのレジスタに対するデータ転送命令が追加されます。追加されるデータ転送命令は、次の 3 種類に分類されます。

DSP ユニットは、X メモリと Y メモリに X バスと Y バスと呼ばれる専用バスで接続されており、これらのバスを用いたデータ転送命令を使用することで、X/Y メモリとの間で同時に 2 個のデータを転送することができます (ダブルデータ転送命令)。このダブルデータ転送命令は、DSP データ演算命令と組み合わせて記述することができ、データ転送およびデータ演算を並列に実行することが可能です。

また、DSP ユニットは、L バスと呼ばれる CPU が使用するバスとも接続されており、DSR を除く全レジスタは CPU の生成する論理アドレス空間全てにアクセスすることができます (シングルデータ転送命令)。シングルデータ転送命令は、DSP データ演算命令と組み合わせることはできず、また一度にアクセスできるデータは 1 個だけになります。

さらに DSP ユニットのレジスタのうちいくつかは、CPU のシステムレジスタとして扱われ、これらを制御するためのシステム制御命令が追加されています。CPU の汎用レジスタとの間は、データ転送用のバス (C バス) で接続されています。

いずれのタイプのデータ転送命令でも、アクセスするアドレスは CPU が生成し、出力します。これらの命令に対しては CPU の汎用レジスタのいくつかはアドレス生成に使用され、また独特のアドレッシングモードを有します。

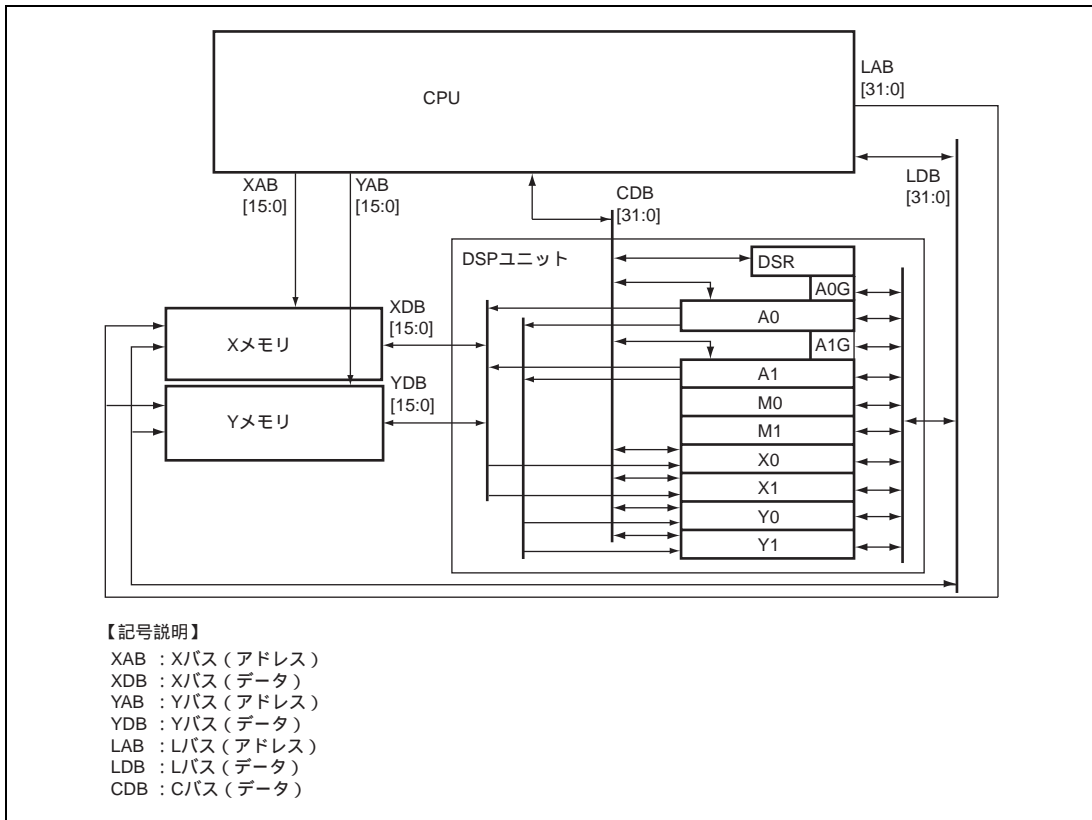


図 3.4 DSP レジスタとバスの接続

(1) ダブルデータ転送命令 (MOVX.W、MOVY.W、MOVX.L、および MOVY.L)

ダブルデータ転送命令の命令形式を表 3.12 に示します。

ダブルデータ転送命令では、X メモリに対するアクセス、および Y メモリに対するアクセスを同時に記述することができます。このアクセスにはそれぞれ X バスと Y バスと呼ばれる専用バスを使用します。CPU の命令フェッチは L バスを使用するため、CPU の命令フェッチとの間でバス競合が生じることもありません。

X メモリに対するロード命令はデスティネーションオペランドとして X0 と X1 レジスタのどちらか 1 つを指定し、Y メモリに対するロード命令はデスティネーションオペランドとして Y0 と Y1 レジスタのどちらか 1 つを指定できます。X メモリと Y メモリのいずれのストア命令もソースオペランドとして A0 と A1 レジスタのどちらか 1 つを指定することができます。ダブルデータ転送命令はワードデータ (16 ビット) のみを扱います。ワードデータ転送命令を実行すると、レジスタオペランドの上位ワード側が用いられます。ワードデータロード命令の場合は、データはデスティネーションレジスタの上位ワード側にデータが読み込まれ、デスティネーションの低位ワード側が自動的に 0 クリアされます。

ダブルデータ転送命令では、DSP データ演算命令を並列して記述することができます。ただし、実行する演算命令に条件付き命令を指定した場合でも、指定した条件はどのデータ転送命令に対しても影響しません。条件付

3. DSP 演算ユニット

き命令については、「3.5 DSP データ演算命令」を参照してください。

ダブルデータ転送命令は、X メモリか Y メモリのみアクセスすることができます。その他のメモリ空間はアクセスすることができません。また、X バスと Y バスは、それぞれ 16 ビット (64k バイト) のアドレス空間を持っており、L バスのアドレス空間の H'A500 0000 ~ H'A500 FFFF および H'A501 0000 ~ H'A501 FFFF の範囲に対応します。この範囲は、P2 領域または Uxy 領域に含まれるため、キャッシュの影響を受けません。

(2) シングルデータ転送命令

シングルデータ転送命令の命令形式を表 3.13 に示します。

シングルデータ転送命令は、任意のメモリアドレス空間にアクセスできます。DSR を除く DSP ユニットのすべてのレジスタ*をソースオペランド、デスティネーションオペランドに指定することができます。ガードビットレジスタ (A0G および A1G) も独立したレジスタとして指定することができます。シングルデータ転送命令では、L バス (LAB および LDB) を使用するので、CPU の扱うすべての論理アドレス空間をアクセスすることができます。キャッシュ可能領域を指定し、かつキャッシュが利用可能な場合には、キャッシングの対象になります。X と Y メモリは、CPU の出力する論理アドレス空間 (L バス) の一部としてマッピングされており、シングルデータ転送命令でアクセスすることができます。ただし、CPU の命令フェッチは L バスを使用するため、データ転送と命令フェッチの間でバス競合が発生する可能性があります。

シングルデータ転送は、ワードとロングワードのいずれも扱うことができます。ワードデータ転送命令を実行すると、レジスタオペランドの上位ワード側が用いられます。ワードデータロード命令の場合は、データはデスティネーションレジスタの上位ワード側に読み込まれ、デスティネーションの下位ワード側は自動的に 0 でクリアされます。ガードビット部分がサポートされている場合には、符号ビットが拡張されて格納されます。ロングワードデータロード命令の場合は、データはデスティネーションレジスタの上位ワード側と下位ワード側に読み込まれ、ガードビットがあれば、符号ビットが拡張されて格納されます。ガードレジスタストア命令の場合は、符号ビットが LDB の上位 24 ビットに拡張されて LDB バスに読み出されます。

【注】 * DSR レジスタは、システムレジスタとして定義されているので、LDS、STS 命令でのデータの転送が可能です。

1. すべてのデータ転送命令は、パイプラインの MA ステージで実行されます。
2. すべてのデータ転送命令は、DSR レジスタの条件コードビットのどれも更新しません。

(3) システム制御命令

DSP ユニットのレジスタのうち DSR、A0、X0、X1、Y0、Y1 レジスタは、CPU のシステムレジスタとして扱うことができ、STS と LDS 命令によって汎用レジスタやメモリとの間でデータ転送を行うことができます。これらのシステム制御命令は、CPU レジスタの PR、MACH、MACL と全く同じように扱うことができ、アドレッシングも同一です。

表 3.9 DSP モード拡張システム制御命令

命令	動作	実行ステート
STS DSR,Rn	DSR Rn	1
STS A0,Rn	A0 Rn	1
STS X0,Rn	X0 Rn	1
STS X1,Rn	X1 Rn	1
STS Y0,Rn	Y0 Rn	1
STS Y1,Rn	Y1 Rn	1
STS.L DSR,@-Rn	Rn-4 Rn, DSR (Rn)	1
STS.L A0,@-Rn	Rn-4 Rn, A0 (Rn)	1
STS.L X0,@-Rn	Rn-4 Rn, X0 (Rn)	1
STS.L X1,@-Rn	Rn-4 Rn, X1 (Rn)	1
STS.L Y0,@-Rn	Rn-4 Rn, Y0 (Rn)	1
STS.L Y1,@-Rn	Rn-4 Rn, Y1 (Rn)	1
LDS.L @Rn+,DSR	(Rn) DSR, Rn+4 Rn	1
LDS.L @Rn+,A0	(Rn) A0, Rn+4 Rn	1
LDS.L @Rn+,X0	(Rn) X0, Rn+4 Rn	1
LDS.L @Rn+,X1	(Rn) X1, Rn+4 Rn	1
LDS.L @Rn+,Y0	(Rn) Y0, Rn+4 Rn	1
LDS.L @Rn+,Y1	(Rn) Y1, Rn+4 Rn	1
LDS Rn,DSR	Rn DSR	1
LDS Rn,A0	Rn A0	1
LDS Rn,X0	Rn X0	1
LDS Rn,X1	Rn X1	1
LDS Rn,Y0	Rn Y0	1
LDS Rn,Y1	Rn Y1	1

3.4.1 汎用レジスタ

DSP タイプの命令では、汎用レジスタ 16 本のうち 10 本のレジスタがダブルデータ転送命令とシングルデータ転送命令で特別なアドレスポインタおよびインデックスレジスタとして使用されます。DSP タイプ命令でのレジスタの目的を表すもう 1 つの記号を [] 内に示します。

- ダブルデータ転送命令 (XメモリとYメモリに同時にアクセスする場合)

ダブルデータ転送命令は、XとYメモリに同時にアクセスできます。XとYメモリのアドレスを指定するために、次の2つのアドレスポインタセットを用意しています。

アクセス対象	アドレスポインタ	インデックスレジスタ
Xメモリ (MOVX.W)	R4,R5 [Ax]	R8 [Ix]
Yメモリ (MOVY.W)	R6,R7 [Ay]	R9 [Iy]

3. DSP 演算ユニット

- シングルデータ転送命令

シングルデータ転送命令では、Lバスを使用して全ての論理アドレス空間をアクセスできます。次のアドレスポインタとインデックスレジスタを使用します。

アクセス対象	アドレスポインタ	インデックスレジスタ
全論理空間 (MOVS.W/L)	R4,R5,R2,R3 [As]	R8 [Is]

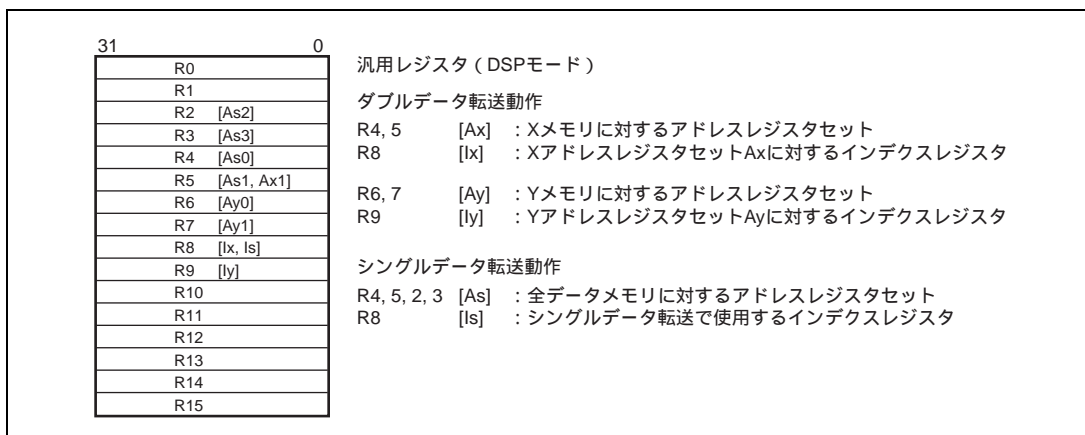


図 3.5 汎用レジスタ (DSP モード)

アセンブラでは R0~R9 の記号名 (シンボル) を使います。DSP データ転送命令では、次のようなレジスタの別名 (エイリアス) を使うこともできます。アセンブラでは、次のように書きます。

```
Ix: .REG (R8)
```

名前 Ix が R8 の別名になります。そのほか、次のように別名を付けます。

```
Ax0: .REG (R4)
```

```
Ax1: .REG (R5)
```

```
Ix: .REG (R8)
```

```
Ay0: .REG (R6)
```

```
Ay1: .REG (R7)
```

```
Iy: .REG (R9)
```

```
As0: .REG (R4) ;これは、シングルデータ転送のために別名が必要なときの定義です。
```

```
As1: .REG (R5) ;これは、シングルデータ転送のために別名が必要なときの定義です。
```

```
As2: .REG (R2)
```

```
As3: .REG (R3)
```

```
Is: .REG (R8) ;これは、シングルデータ転送のために別名が必要なときの定義です。
```

3.4.2 DSP データアドレッシング

ダブルデータ転送命令およびシングルデータ転送命令の関係を表 3.10 に示します。

表 3.10 データ転送命令の関係

	ダブルデータ転送命令	シングルデータ転送命令
		MOVX.W MOVY.W
アドレスレジスタ	Ax : R4, R5 Ay : R6, R7	As : R2, R3, R4, R5
インデックス レジスタ	Ix : R8 Iy : R9	Is : R8
アドレッシング	Nop/Inc(+2)/インデクス加算 : ポストインクリメント	Nop/Inc(+2,+4)/インデクス加算 : ポストインクリメント
		Dec(-2,-4) : プリデクリメント
モジュロ アドレッシング	可能	不可
データバス	XDB, YDB	LDB
データ長	16 ビット (ワード)	16 ビット/32 ビット (ワード/ロングワード)
バス競合	なし	あり
アクセス可能メモリ	X, Y メモリ	すべてのメモリ空間
ソースレジスタ	Dx, Dy : A0, A1	Ds : A0/A1, M0/M1, X0/X1, Y0/Y1, A0G, A1G
デスティネーション レジスタ	Dx : X0/X1 Dy : Y0/Y1	Ds : A0/A1, M0/M1, X0/X1, Y0/Y1, A0G, A1G

(1) ダブルデータ転送命令のアドレッシングモード

ダブルデータ転送命令には、次の 3 つのアドレッシングモードがあります。

- 更新無し

AxとAyレジスタがアドレスポインタです。@Axと@Ayへのアクセスが行われ、AxとAyの値は更新されません。

- インクリメント

AxとAyレジスタがアドレスポインタです。@Axと@Ayへのアクセス後、+2が加算されます(ポスト更新)。

- インデクスレジスタ加算

AxとAyレジスタがアドレスポインタです。@Axと@Ayへのアクセス後、それぞれIxとIyレジスタの値が加算されます(ポスト更新)。ダブルデータ転送命令にデクリメントアドレッシングはありませんが、デクリメントさせるためには-2をインデクスレジスタに設定し、インデクスレジスタ加算アドレッシングを指定します。

3. DSP 演算ユニット

X と Y メモリアドレスリングの場合は、アドレスポインタのビット 0 が無効になります。X と Y メモリアドレスリングの場合は、アドレスポインタとインデックスレジスタのビット 0 には 0 を書き込んでください。1 を書き込んだ場合は、動作の保証はできません。

X と Y バスを使用して X メモリと Y メモリへアクセスする場合は、 A_x と A_y の上位ワード側は無視されます。また、 A_{y+2} と A_{y+1y} の結果は、 A_y の下位ワード側に格納され、上位ワード側は元の値が保持されます。ただし、 A_{x+2} と A_{x+1x} の演算は 32 ビットで行われ、上位ワード側が変化する場合もあります。

(2) シングルデータ転送命令のアドレッシングモード

シングルデータ転送命令には、次の 4 つのデータアドレッシングモードがあります。

- 更新無し

A_s レジスタがアドレスポインタです。@ A_s へのアクセスが行われますが、 A_s は更新されません。

- インクリメント

A_s レジスタがアドレスポインタです。@ A_s へのアクセス後、転送サイズに応じて +2 または +4 が加算されます (ポスト更新)。

- インデックスレジスタ加算

A_s レジスタがアドレスポインタです。@ A_s へのアクセス後、 I_s レジスタの値が加算されます (ポスト更新)。

- デクリメント

A_s レジスタがアドレスポインタです。データ転送前に -2 または -4 が加算 (+2 または +4 が減算) されます (プリ更新)。

シングルデータ転送命令では、アドレスの 32 ビット全てが有効です。

3.4.3 モジュロアドレッシング

ダブルデータ転送命令では、モジュロアドレッシングを使用することができます。モジュロアドレッシング制御命令を表 3.11 に示します。モジュロアドレッシングモードが設定されている場合は、アドレスポインタの値がすでに設定されたモジュロ終了アドレスになると、アドレスポインタはモジュロ開始アドレスになります。

モジュロアドレッシングの制御には、DSP モードで拡張されるモジュロレジスタ (MOD) および SR レジスタの DMX と DMY ビットを使用します。

MOD レジスタにモジュロアドレス領域の開始アドレスと終了アドレスを格納します。MOD レジスタの上位ワード側にモジュロ開始アドレス (MS) を、MOD レジスタの下位ワード側にモジュロ終了アドレス (ME) を格納します。MOD レジスタに対する LDC 命令および STC 命令が拡張されます。

SR レジスタの DMX ビットをセットすると X アドレスレジスタが、DMY ビットをセットすると Y アドレスレジスタがそれぞれモジュロアドレッシングモードになります。モジュロアドレッシングは、X と Y アドレスレジスタどちらかに対してだけ有効です。両方を同時にモジュロアドレッシングモードにすることはできません。したがって、DMX と DMY ビットを同時にセットしないでください。万一同時にセットされた場合には、DMY ビットのみ有効となります。DMX と DMY ビットは、SR レジスタに対する STC 命令および LDC 命令で設定できま

す。

モジュロアドレッシング制御中に例外を受理した場合は、MOD レジスタ、および SR レジスタの DMX と DMY ビットを退避してください。復帰時にこれらを回復することにより、モジュロアドレッシング制御に復帰することができます。

表 3.11 モジュロアドレッシング制御命令

命令	動作	実行ステート
STC MOD,Rn	MOD Rn	1
STC.L MOD,@-Rn	Rn-4 Rn,MOD (Rn)	1
LDC.L @Rn+,MOD	(Rn) MOD,Rn+4 Rn	4
LDC Rn,MOD	Rn MOD	4

モジュロアドレッシングの使用例を以下に示します。

```

MOV.L #H'70047000,R10 ; MS=H'7000 ME=H'7004 として
LDC R10, MOD          ; MOD レジスタに ME:MS を設定します。
STC SR,R10            ;
MOV.L #H'FFFFFF3FF, R11
MOV.L #H'00000400, R12;
AND R11,R10           ;
OR R12,R10            ;
LDC R10,SR            ; SR.DMX=1, SR.DMY=0。X モジュロアドレッシングモードを設定。
MOV.L #H'A5007000,R4
MOVX.W @R4+,X0        ;R4: H'A5007000 H'A5007002
MOVX.W @R4+,X0        ;R4: H'A5007002 H'A5007004
MOVX.W @R4+,X0        ;R4: H'A5007004 H'A5007000 (ME と一致したので、MS が設定
                       ;されます)
MOVX.W @R4+,X0        ;R4: H'A5007000 H'A5007002

```

MS と ME にモジュロ開始アドレスとモジュロ終了アドレスを指定した後に DMX または DMY ビットを 1 にセットします。

DMX または DMY ビットで指定された X または Y データ転送命令が実行されると、アドレスレジスタの更新前の値が ME と比較されます*。データ転送の後、比較結果が ME と一致していた場合、アドレスレジスタの更新後の値として、MS のモジュロ開始アドレスが代入されます。

X または Y データ転送命令のアドレッシングタイプが「更新無し」の場合は、たとえ ME と一致しても MS への復帰は行われません。また、X または Y データ転送命令のアドレッシングタイプが「インデクスレジスタ加算」の場合は、アドレスポインタは ME と一致せずにその値を超えてしまうことがあります。この場合は、アドレスポインタはモジュロ開始アドレスには戻りません。

最大のモジュロサイズは、64k バイトです。これは、X と Y メモリをアクセスするには十分です。

3. DSP 演算ユニット

【注】 * モジュロアドレッシングに限らず、XとYメモリアドレッシング時は、ビット0が無視されます。アドレスポインタ、インデックスレジスタ、MS、およびMEのビット0には、必ず0を書き込んでください。1を書き込んだ場合は、動作の保証はできません。

3.4.4 メモリのデータ形式

DSP 命令で扱えるメモリのデータ形式は、ワードとロングワードに分けられます。MOVX.W 命令で 2n 以外のアドレスから始まるワードデータをアクセスしようとしたり、または MOVX.L、LDS.L、および STS.L 命令で 4n 以外のアドレスから始まるロングワードにアクセスしようとするるとアドレスエラーが発生します。このような場合は、アクセスするデータは保証されません。

MOVX.W および MOVY.W 命令により 2n 以外のアドレスから始まるワードデータをアクセスしたときは、アドレスエラーが発生しません。必ず 2n 境界にアドレスを設定してください。2n 以外のアドレスに設定した場合は、アクセスするデータは保証されません。

3.4.5 ダブル、シングルデータ転送命令の命令フォーマット

ダブルデータ転送命令の命令形式を表 3.12 に、シングルデータ転送命令の命令形式を表 3.13 に示します。

表 3.12 ダブルデータ転送の命令形式

分類	ニーモニック	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Xメモリ データ 転送	NOPX	1	1	1	1	0	0	0		0		0		0	0		
	MOVX.W @Ax,Dx							Ax		Dx		0			0	1	
	MOVX.W @Ax+,Dx														1	0	
	MOVX.W @Ax+lx,Dx														1	1	
	MOVX.W Da,@Ax									Da		1			0	1	
	MOVX.W Da,@Ax+														1	0	
	MOVX.W Da,@Ax+lx														1	1	
Yメモリ データ 転送	NOPY	1	1	1	1	0	0		0		0		0			0	0
	MOVY.W @Ay,Dy								Ay		Dy		0			0	1
	MOVY.W @Ay+,Dy															1	0
	MOVY.W @Ay+ly,Dy															1	1
	MOVY.W Da,@Ay										Da		1			0	1
	MOVY.W Da,@Ay+															1	0
	MOVY.W Da,@Ay+ly															1	1

【注】 Ax : 0=R4, 1=R5 Ay : 0=R6, 1=R7 Dx : 0=X0, 1=X1 Dy : 0=Y0, 1=Y1 Da : 0=A0, 1=A1

表 3.13 シングルデータ転送命令の命令形式

分類	ニーモニック	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0										
シングル データ 転送	MOVS.W @-As,Ds	1	1	1	1	0	1	As	Ds	0:(*)	0	0	0	0	0	0	0										
	MOVS.W @As,Ds																	0:R4	1:(*)	0	1						
	MOVS.W @As+,Ds																	1:R5	2:(*)	1	0						
	MOVS.W @As+Is,Ds																	2:R2	3:(*)	1	1						
	MOVS.W Ds,@-As																	3:R3	4:(*)	0	0	0	1				
	MOVS.W Ds,@As																		5:A1	0	1						
	MOVS.W Ds,@As+																		6:(*)	1	0						
	MOVS.W Ds,@As+Is																		7:A0	1	1						
	MOVS.L @-As,Ds																		8:X0	0	0	1	0				
	MOVS.L @As,Ds																		9:X1	0	1						
	MOVS.L @As+,Ds																		A:Y0	1	0						
	MOVS.L @As+Is,Ds																		B:Y1	1	1						
	MOVS.L Ds,@-As																		C:M0	0	0	1	1				
	MOVS.L Ds,@As																		D:A1G	0	1						
	MOVS.L Ds,@As+																		E:M1	1	0						
	MOVS.L Ds,@As+Is																		F:A0G	1	1						

【注】 * システム予約コード

3.5 DSP データ演算命令

3.5.1 DSP レジスタ

DSP レジスタとして 8 つのデータレジスタ (A0、A1、X0、X1、Y0、Y1、M0、および M1) と 1 つのコントロールレジスタ (DSR) を持っています (図 3.3)。

これらの DSP レジスタには、4 種類の命令でアクセスします。1 つ目は、DSP データ演算命令です。DSP 固定小数点算術演算命令のソースレジスタに A0 または A1 を使用するときは、ガードビット (ビット 39~32) を使用します。デスティネーションレジスタに A0 または A1 を使用するときは、ガードビットのビット 39~32 が有効になります。DSP 固定小数点算術演算命令がソースレジスタに A0 または A1 以外の DSP レジスタを使用するときは、ソースレジスタの値をビット 39~32 に符号拡張して使用します。A0 または A1 以外の DSP レジスタをデスティネーションレジスタに使用するときは、結果のビット 39~32 は破棄されます。

2 つ目は X または Y データ転送命令の MOVX.W と MOVY.W です。この命令は、16 ビットの X と Y データバス (図 3.4) を用い X と Y メモリにアクセスします。この命令でロードまたはストアされるレジスタは、常に上位 16 ビット (ビット 31~16) です。X0 および X1 は X メモリロードのデスティネーションレジスタ、Y0 および Y1 は Y メモリロードのデスティネーションレジスタになることができますが、他のレジスタはこの命令のデスティネーションレジスタになることはできません。データをレジスタの上位 16 ビット (ビット 31~16) に読み込むとき、レジスタの下位 16 ビット (ビット 15~0) は自動的に 0 にクリアされます。A0 と A1 はこの動作によって X、Y メモリに格納することができますが、他のレジスタは格納することができません。

3 つ目は、シングルデータ転送命令の MOV.S.W および MOV.S.L です。この命令は、LDB (図 3.4) によって任意のメモリにアクセスできます。すべての DSP レジスタは LDB につながり、データ転送のソースおよびデスティネーションレジスタになることができます。これには、ワードおよびロングワードアクセスモードがあります。ワードモードでは、A0G と A1G を除く DSP レジスタの上位 16 ビット (ビット 31~16) がロードあるいはストアの対象になります。ワードモードで A0G と A1G 以外のレジスタにデータを読み込むと、レジスタの下位 16 ビットは 0 にクリアされます。A0 または A1 の場合は、データはビット 39~32 に符号拡張され、その下位 16 ビットは 0 にクリアされます。ワードモードで A0G または A1G がデスティネーションレジスタの場合は、データは 8 ビットレジスタに読み込まれますが、A0 または A1 は 0 にクリアされません。ロングワードモードでデスティネーションレジスタが A0 または A1 のときは、データは 39~32 に符号拡張されます。

4 つ目は、システム制御命令の LDS と STS、および LDS.L と STS.L です。DSP レジスタの DSR、A0、X0、X1、Y0 および Y1 は、システムレジスタとして扱われます。これらのレジスタ用に、CPU の汎用レジスタとシステムレジスタ間のデータ転送命令およびメモリアクセス命令があります。

表 3.14 および表 3.15 は、DSP 命令で使用するレジスタのデータタイプを示します。命令コードの制限のため、表に示すレジスタの中には使用できない演算もあります。たとえば、PMULS はソースレジスタに A1 を使用できますが、A0 は使用できません。これらの表は、レジスタの選択性の詳細については省略しています。

表 3.14 DSP 命令のデスティネーションレジスタ

レジスタ		命令	ガードビット		レジスタビット			
			39	32	31	16	15	0
A0、A1	DSP 演算	固定小数点、PSHA、PMULS	(符号拡張) 40ビット結果					
		整数、PDMSB	(符号拡張) 24ビット結果				0クリア	
		論理、PSHL	0クリア		16ビット結果		0クリア	
	データ 転送	MOVS.W	符号拡張		16ビットデータ		0クリア	
		MOVS.L	符号拡張		32ビットデータ			
A0G、A1G	データ	MOVS.W	8ビットデータ		更新しない			
	転送	MOVS.L	8ビットデータ		更新しない			
X0、X1 Y0、Y1	DSP 演算	固定小数点、PSHA、PMULS	32ビット結果					
		整数、論理、PDMSB、PSHL					16ビット結果	
M0、M1	データ 転送	MOVX/Y.W、MOVS.W			16ビットデータ		0クリア	
		MOVS.L			32ビットデータ			

表 3.15 DSP 命令のソースレジスタ

レジスタ		命令	ガードビット		レジスタビット			
			39	32	31	16	15	0
A0、A1	DSP 演算	固定小数点、PDMSB、PSHA	40ビットデータ					
		整数	24ビットデータ					
		論理、PSHL、PMULS			16ビットデータ			
	データ 転送	MOVX/Y.W、MOVS.W			16ビットデータ			
		MOVS.L			32ビットデータ			
A0G、A1G	データ	MOVS.W	8ビットデータ					
	転送	MOVS.L	8ビットデータ					
X0、X1 Y0、Y1 M0、M1	DSP 演算	固定小数点、PDMSB、PSHA	符号*		32ビットデータ			
		整数	符号*		16ビットデータ			
論理、PSHL、PMULS				16ビットデータ				
	データ 転送	MOVS.W			16ビットデータ			
		MOVS.L			32ビットデータ			

【注】 * データを符号拡張し、ALUに入力する。

DSP ユニットの、1つのコントロールレジスタ兼用の DSP ステータスレジスタ (DSR) を持っています。DSR は、DSP データ演算結果の状態 (ゼロ、負、など) を保持します。DSR は、また CPU の T ビットに類似した DC ビットを持っており、状態を示します。条件付き DSP データ処理命令は、この DC ビットに基づいて実行を制御します。この制御は、DSP ユニットの命令にのみ影響を与えます。すなわち、DSP レジスタのみを制御し、アドレスレジスタの更新、およびロードやストア命令などの CPU の命令を制御することはできません。DC 状態選択

3. DSP 演算ユニット

ビット (CS [2:0]) には、DC ビットに反映する条件を指定します。

PMULS、MOVX、MOVY、および MOVS を除く無条件 DSP タイプのデータ演算命令は条件フラグと DC ビットを更新しますが、MAC 命令を含む CPU 命令はどれも DC ビットを更新しません。条件付き DSP タイプデータ演算命令も DSR を更新することはありません。

表 3.16 DSR レジスタのビットの説明

ビット	ビット名	初期値	R/W	説明
31~12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
11~9	TS2~TS0	すべて 0	R/W	T ビット状態選択 TC ビットが 1 のとき、SR レジスタの T ビットに設定する演算結果状態を選択するモードを指定します。ただし、SR レジスタの S ビットが 1 のときもオーバーフロー検出を行います。 000: キャリ/ポローモード 001: 負値モード 010: ゼロ値モード 011: オーバフローモード 100: 符号付き大モード 101: 符号付き以上モード 110: 予約 (設定禁止) 111: 予約 (設定禁止)
8	TC	0	R/W	TC ビット 0: SR レジスタの T ビットは DSP 命令に依存しません。 1: SR レジスタの T ビットは DSP 命令実行時、DSR レジスタの T S ビットの状態により変化します。ただし、条件付き DSP 命令実行時は、T ビットは変化しません。
7	GT	0	R/W	符号付き大ビット 演算結果が正 (ゼロを除く)、またはオペランド 1 がオペランド 2 より大きいことを表します。 1: 演算結果が正、またはオペランド 1 がオペランド 2 より大きい
6	Z	0	R/W	ゼロビット 演算結果がゼロ (0)、またはオペランド 1 がオペランド 2 と等しいことを表します。 1: 演算結果がゼロ (0)、または等しい
5	N	0	R/W	負値ビット 演算結果が負、またはオペランド 1 がオペランド 2 より小さいことを表します。 1: 演算結果が負、またはオペランド 1 がオペランド 2 より小さい

ビット	ビット名	初期値	R/W	説明
4	V	0	R/W	オーバフロービット 演算結果がオーバフローしたことを表します。 1：演算結果がオーバフロー
3~1	CS	すべて 0	R/W	DC ビット状態選択 DC ビットに設定する演算結果状態を選択するためのモードを指定します。 000：キャリ/ボローモード 001：負値モード 010：ゼロモード 011：オーバフローモード 100：符号付き大モード 101：符号付き以上モード 110：予約（設定禁止） 111：予約（設定禁止）
0	DC	0	R/W	DSP 状態ビット CS ビットで指定されたモードで演算結果の状態を設定します。 0：指定されたモードの状態が成立しない。 1：指定されたモードの状態が成立する。 PADDC または PSUBC 命令実行後の DC ビットは、CS ビットに関係なくキャリ/ボローモードで演算結果の状態を設定します。

DSR は、システムレジスタに割り当てられます。DSR には、次のロードまたはストア命令が用意されています。

```
STS DSR, Rn;
STS.L DSR, @-Rn;
LDS Rn, DSR;
LDS.L @Rn+, DSR;
```

STS 命令で DSR を読み出すとき、上位ビット（ビット 31～ビット 16）はすべて 0 になります。

3.5.2 DSP 命令の命令セット

DSP 命令は、DSP ユニットで処理されるデジタル信号処理の命令です。これらの命令は 32 ビット長の命令コードで、複数の命令を並列に実行します。命令コードは A フィールド、B フィールドの 2 つに分かれており、A フィールドにはダブルデータ転送命令を指定し、B フィールドにはシングルまたはダブルデータ演算命令を指定します。命令は独立して指定することができ、実行も独立に実行されます。

B フィールドのデータ演算命令は 3 種類に分かれています。ダブルデータ演算命令、条件付きシングルデータ演算命令、および無条件シングルデータ演算命令の 3 つです。DSP 命令の命令形式を表 3.17 に示します。それぞれのオペランドは独立に DSP レジスタから選べます。DSP 命令のオペランドとレジスタの対応を表 3.18 に示し

3. DSP 演算ユニット

ます。

A フィールドのみで B フィールドのない並列処理 (DSP データ演算命令がないデータ転送のみの並列処理) も可能です。

表 3.17 DSP 命令の命令形式

分類		命令形式
ダブルデータ演算命令		ALUop. Sx, Sy, Du MLTop. Se, Sf, Dg
シングルデータ演算命令	条件付きシングルデータ演算命令	DCT ALUop. Sx, Sy, Dz
		DCF ALUop. Sx, Sy, Dz
		DCT ALUop. Sx, Dz
		DCF ALUop. Sx, Dz
		DCT ALUop. Sy, Dz
		DCF ALUop. Sy, Dz
無条件シングルデータ演算命令		ALUop. Sx, Sy, Dz ALUop. Sx, Dz ALUop. Sy, Dz MLTop. Se, Sf, Dg

表 3.18 DSP 命令のオペランドとレジスタの対応

レジスタ	ALU、シフト演算				乗算演算		
	Sx	Sy	Dz	Du	Se	Sf	Dg
A0	Yes		Yes	Yes			Yes
A1	Yes		Yes	Yes	Yes	Yes	Yes
M0		Yes	Yes				Yes
M1		Yes	Yes				Yes
X0	Yes		Yes	Yes	Yes	Yes	
X1	Yes		Yes		Yes		
Y0		Yes	Yes	Yes	Yes	Yes	
Y1		Yes	Yes			Yes	

DSP 命令を書くときは、最初に B フィールドの命令を書いて、次に A フィールドの命令を書きます。DSP 命令による並列処理プログラム例を図 3.6 に示します。A フィールドのみで B フィールドのない並列処理 (DSP データ演算命令がないデータ転送のみの並列処理) も可能です。

PADD	A0, M0, A0	PMULS	X0, Y0, M0	MOVX.W @R4+, X0	MOVY.W @R6+, Y0
DCF	PINC	M1, A1		MOVX.W @R5+R8, X0	MOVY.W @R7+, Y1
PCMP	M1, M0			MOVX.W @R4, X1	[NOPY]

図 3.6 DSP 命令による並列処理プログラムの例

ここで、[]は省略可能な部分を表します。

NOPX と NOPY のノーオペレーション命令は、省略可能です。DSP 命令の B フィールドの詳細は、「3.6.4 DSP データ演算命令セット」を参照してください。

DSR レジスタの条件コードビット (DC) は、常に無条件の ALU またはシフト演算命令の結果に基づいて更新されます。条件付き命令の場合は、DC ビットを更新しません。乗算命令も DC ビットを更新しません。DC ビットの更新は、DSR レジスタの CS[2:0]ビットにより行われます。表 3.19 に DC ビットの更新ルールについて示します。

表 3.19 DC ビットの更新ルール

CS [2:0]			条件モード	説 明
0	0	0	キャリーまたはポローモード	ALU 算術演算の結果、キャリーまたはポローが発生した場合は、DC ビットがセットされます。それ以外は 0 クリアされます。 シフト命令 (PSHA または PSHL) の実行時、最後にシフトアウトしたビットデータが DC ビットにコピーされます。 ALU 論理演算の実行時は、DC ビットは常に 0 クリアされます。
0	0	1	負値モード	ALU またはシフト (PSHA) 算術演算の実行時は、ガードビット部分を含めて結果の MSB が DC ビットにコピーされます。 ALU またはシフト (PSHL) 論理演算の実行時、ガードビット部分を除く結果の MSB が DC ビットにコピーされます。
0	1	0	ゼロ値モード	ALU またはシフト演算の結果がすべてゼロの場合は、DC ビットがセットされます。それ以外は 0 クリアされます。
0	1	1	オーバフローモード	ALU またはシフト (PSHA) の算術演算結果がガードビット部分を除いたデスティネーションレジスタの範囲を越える場合は、DC ビットがセットされます。それ以外は 0 クリアされます。 ALU またはシフト (PSHL) の論理演算の実行時は、DC ビットは常に 0 クリアされます。
1	0	0	符号付き大モード	このモードは符号付き以上モードに類似していますが、結果がすべて 0 の場合は、DC は 0 クリアされます。 DC = $\sim(\text{負値} \wedge \text{オーバレンジ}) \vee \text{ゼロ値}$; 算術演算の場合 DC = 0 ; 論理演算の場合

3. DSP 演算ユニット

CS [2:0]			条件モード	説明
1	0	1	符号付き以上モード	<p>ALU またはシフト (PSHA) の算術演算の結果がガードビットを含んだデスティネーションレジスタの範囲を超える場合 (オーバーレンジと呼ぶ) は、定義は負値モードと同じになります。オーバーレンジでない場合は、定義は負値モードの逆になります。</p> <p>ALU またはシフト (PSHL) の論理演算の実行時は、DC ビットは常に 0 クリアされます。</p> <p>DC = ~(負値^オーバーレンジ) ; 算術演算の場合</p> <p>DC = 0 ; 論理演算の場合</p>
1	1	0	予約 (設定禁止)	
1	1	1	予約 (設定禁止)	

- 条件付き演算とデータ転送

このクラスに属する命令の中には前記のように、条件付きで実行することができるものがあります。ただし、指定した条件は命令の B フィールドに対してのみ有効であって、並列して指定したデータ転送命令には有効ではありません。図 3.7 に例を示します。

DCT	PADD	X0, Y0, A0	MOVX.W	@R4+, X0	MOVY.W	A0, @R6+R9
<条件が真の場合>						
実行前:	X0=H'33333333,	Y0=H'55555555,	A0=H'123456789A,			
	R4=H'00008000,	R6=H'00005000,	R9=H'00000004,			
	(R4)=1111	(R6)=2222				
実行後:	X0=H'11110000,	Y0=H'55555555,	A0=H'0088888888,			
	R4=H'00008002,	R6=H'00005004,	R9=H'00000004,			
	(H'00008000)=H'1111	(H'00008233)=H'2222				
<条件が偽の場合>						
実行前:	X0=H'33333333,	Y0=H'55555555,	A0=H'123456789A,			
	R4=H'00008000,	R6=H'00005000,	R9=H'00000004,			
	(R4)=1111	(R6)=2222				
実行後:	X0=H'11110000,	Y0=H'55555555,	A0=H'123456789A,			
	R4=H'00008002,	R6=H'00005004,	R9=H'00000004,			
	(H'00008000)=H'1111	(H'00008233)=H'2222				

図 3.7 条件付き演算とデータ転送命令の例

- NOPXおよびNOPYの命令コードの割当て

DSP データ演算命令と同時に並列処理されるデータ転送命令がないときは、データ転送命令に NOPX または NOPY 命令を書くか、あるいは命令を省略することもできます。NOPX または NOPY 命令を書くかあるいは省略しても命令コードは同じです。また、DSP データ演算命令がなく、並列処理されるデータ転送命令だけのときも同様です。このときの、NOPX と NOPY の命令コードの例を表 3.20 に示します。

表 3.20 NOPX と NOPY の命令コードの例

命令	コード
PADD X0, Y0, A0 MOVX.W @R4+, X0 MOVY.W @R6+R9, Y0	1111100000001011 1011000100000111
PADD X0, Y0, A0 NOPX MOVY.W @R6+R9, Y0	1111100000000011 1011000100000111
PADD X0, Y0, A0 NOPX NOPY	1111100000000000 1011000100000111
PADD X0, Y0, A0 NOPX	1111100000000000 1011000100000111
PADD X0, Y0, A0	1111100000000000 1011000100000111
MOVX.W @R4+, X0 MOVY.W @R6+R9, Y0	1111000000001011
MOVX.W @R4+, X0 NOPY	1111000000001000
MOVS.W @R4+, X0	1111010010001000
NOPX MOVY.W @R6+R9, Y0	1111000000000011
MOVY.W @R6+R9, Y0	1111000000000011
NOPX NOPY	1111000000000000
NOP	000000000001001

3.5.3 DSP タイプデータ形式

本 LSI は、命令によって異なるデータ形式を持っています。ここでは DSP タイプ命令用のデータ形式について解説します。

図 3.8 に 2 進小数点の位置の異なる 3 つの DSP タイプのデータ形式を、また参考として、ビット 0 の右側に 2 進小数点を持つ CPU タイプのデータ形式を示します。

DSP タイプ固定小数点データ形式は、ビット 31 とビット 30 の間に 2 進小数点があります。DSP タイプ整数形式は、ビット 16 とビット 15 の間に 2 進小数点があります。DSP タイプ論理形式には、2 進小数点はありません。データ形式の有効なデータ長は、命令および DSP レジスタによって異なります。

3. DSP 演算ユニット

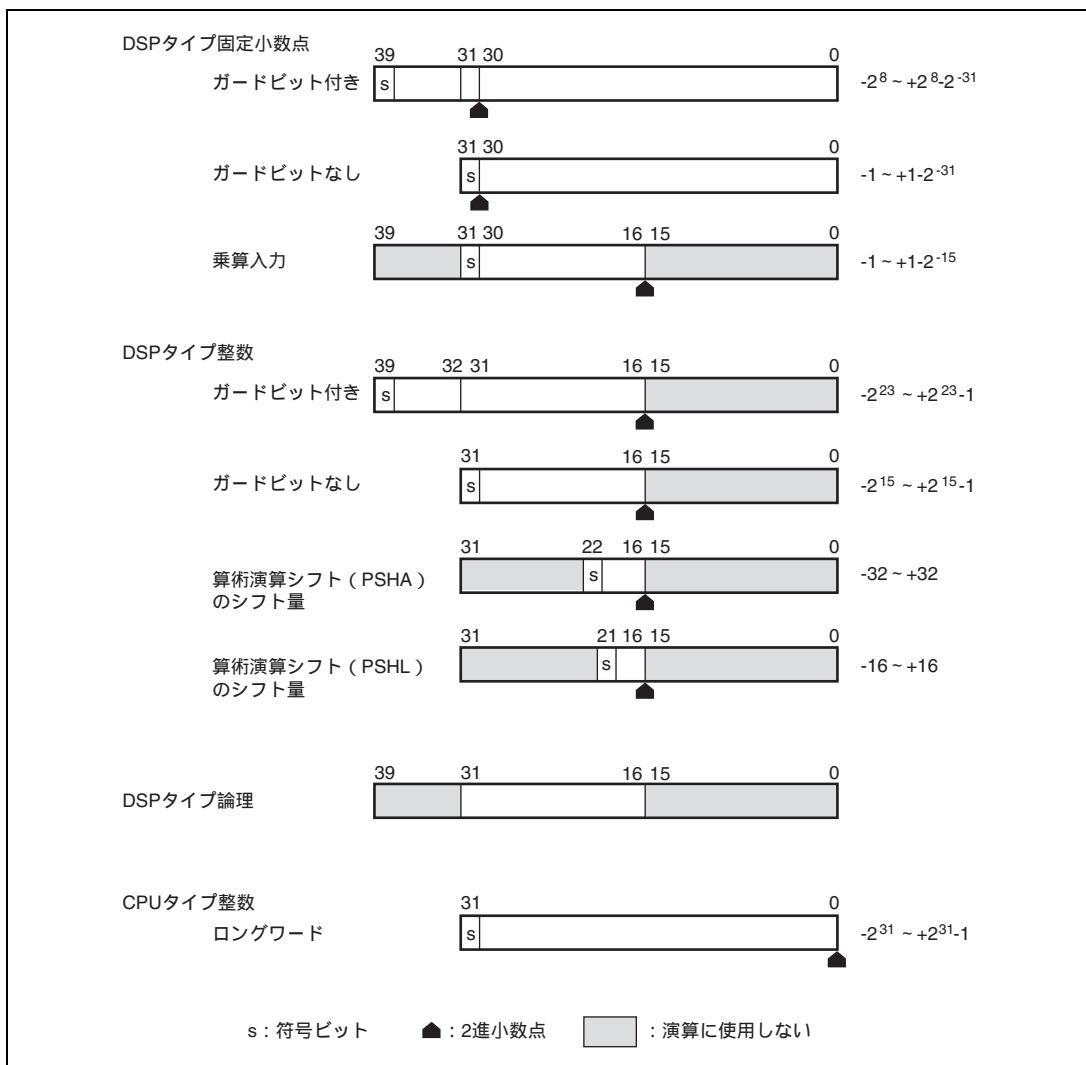


図 3.8 データ形式

算術演算シフト (PSHA) 命令のシフト量は、 $-64 \sim +63$ を表す 7 ビットのフィールドを持っていますが、 $-32 \sim +32$ が有効な数です。また論理演算シフトのシフト量も 6 ビットのフィールドを持っていますが、 $-16 \sim +16$ が有効な数です。無効なシフト量を指定した場合、結果は保証されません。

3.5.4 ALU 固定小数点算術演算

図 3.9 に ALU 固定小数点算術演算フローを示します。表 3.21 はこの演算の種々のタイプを示し、表 3.22 は各オペランドとレジスタの対応を示します。

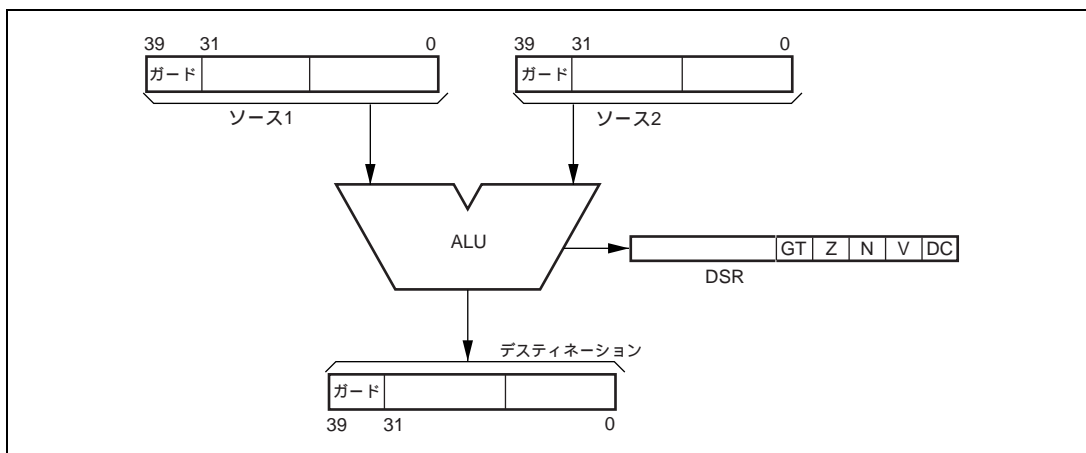


図 3.9 ALU 固定小数点算術演算フロー

【注】 ALU 固定小数点算術演算は、基本的に 40 ビット演算であり、32 ビットの基本精度部分および 8 ビットのガードビット部分から構成されます。したがって、ガードビット部分を提供していないレジスタをソースオペランドに指定すると、符号ビットがガードビット部分にコピーされます。ガードビット部分を提供していないレジスタをデスティネーションオペランドに指定すると、演算結果の下位 32 ビットがデスティネーションレジスタに入力されます。

ALU 固定小数点算術演算は、レジスタ間で実行されます。各ソースオペランドおよびデスティネーションオペランドは、DSP レジスタの 1 つから独立して選択されます。ガードビットを持つレジスタをオペランドに指定すると、ガードビットも含めてこれらの演算が実行されます。これらの演算は、図 3.10 に示すように DSP ステージで実行されます。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

表 3.21 ALU 固定小数点算術演算の種類

ニーモニック	機能	ソース 1	ソース 2	デスティネーション
PADD	加算	Sx	Sy	Dz または Du
PSUB	減算	Sx	Sy	Dz または Du
PADDC	キャリー付き加算	Sx	Sy	Dz
PSUBC	ボロー付き減算	Sx	Sy	Dz
PCMP	比較	Sx	Sy	-
PCOPY	データコピー	Sx	all 0	Dz
		all 0	Sy	Dz
PABS	絶対値	Sx	all 0	Dz
		all 0	Sy	Dz
PNEG	符号反転	Sx	all 0	Dz
		all 0	Sy	Dz
PCLR	クリア	all 0	all 0	Dz

3. DSP 演算ユニット

表 3.22 オペランドとレジスタの対応

レジスタ	Sx	Sy	Dz	Du
A0	Yes		Yes	Yes
A1	Yes		Yes	Yes
M0		Yes	Yes	
M1		Yes	Yes	
X0	Yes		Yes	Yes
X1	Yes		Yes	
Y0		Yes	Yes	Yes
Y1		Yes	Yes	

図 3.10 に示すように、ALU 演算 PADD X0,Y0,A0 と同じラインで並列処理プログラムされたデータ転送命令 MOVX.W @R4+,X0 により MA ステージでメモリから X0 に読み込まれたデータは、データ転送命令のデスティネーションオペランド X0 が ALU 演算のソースオペランド X0 と同一であってもこの ALU 演算用のソースオペランドのデータとしては使用されません。この場合は、前の命令 MOVX.W @R4+,X0 の結果が ALU 演算のソースオペランドのデータとして用いられた後にデータ転送命令 MOVX.W @R4+,X0 のデスティネーションオペランド X0 としてデータが更新されます。

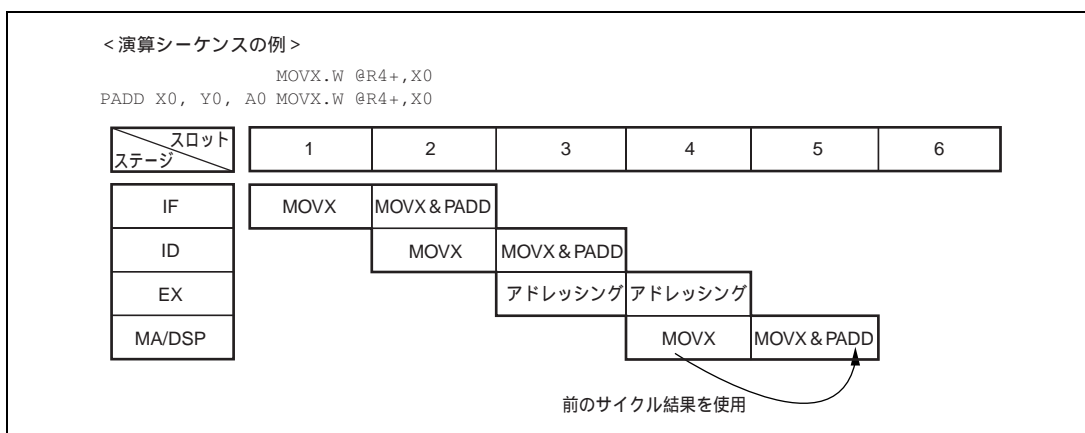


図 3.10 演算シーケンスの例

ALU 算術演算を実行するたびに、DSR の DC、N、Z、V、および GT ビットは基本的に演算結果に従って更新されます。ただし、条件付き命令の場合は、指定条件が真で演算が実行されてもこれらのビットは更新されません。無条件命令の場合は、これらは演算結果に従って常に更新されます。DC ビットの定義は、DSR レジスタの CS[2:0]ビット (DC ビット状態選択ビット) で指定します。DC ビットは、以下ようになります。

(1) キャリー/ポローモード (CS [2:0] = B'000 の場合)

DC ビットは、ガードビット部分を除いた演算結果の最上位ビットからキャリーまたはポローが発生したことを示します。いくつかの例を図 3.11 に示します。このモードがデフォルトです。PABS および PNEG 命令では、入力データが負のとき LSB に 1 を加算するため、キャリービットが発生する場合があります。

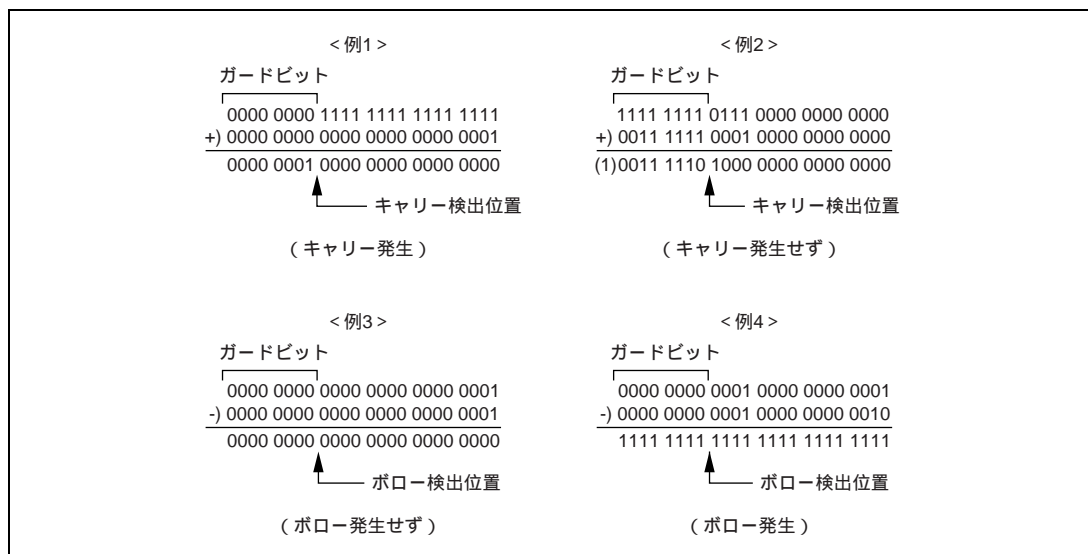


図 3.11 キャリー/ポローモードでの DC ビット生成の例

(2) 負値モード (CS [2:0] = B'001 の場合)

DC フラグは、演算結果の MSB と同じ状態を示します。結果が負の数ときは、DC ビットは 1 を示します。結果が 0 または正の数ときは、DC ビットは 0 を示します。ALU は常に 40 ビットの算術演算を実行するので、正か負かを検出する符号ビットはデスティネーションオペランドに関係なく常に演算結果の MSB から得られます。いくつかの例を図 3.12 に示します。

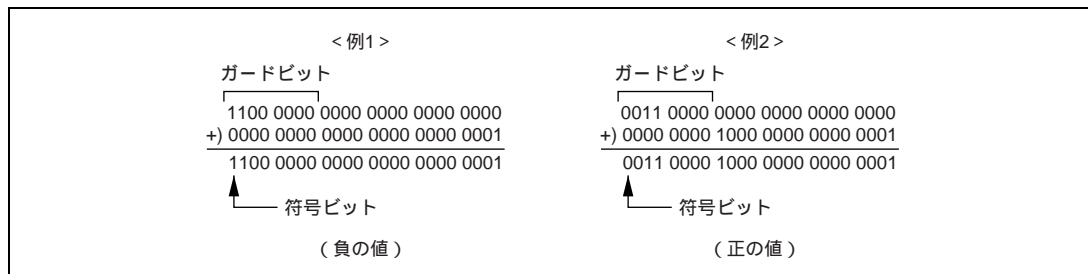


図 3.12 負値モードでの DC ビット生成の例

3. DSP 演算ユニット

(3) ゼロ値モード (CS [2:0]=B'010 の場合)

DC フラグは、演算結果がゼロであるか否かを示します。結果がゼロの場合は、DC ビットは 1 を示します。結果がゼロでない場合は、DC ビットは 0 を示します。

(4) オーバフローモード (CS [2:0]=B'011 の場合)

DC ビットは、結果にオーバフローが発生したか否かを示します。ガードビットを除き演算の結果がデスティネーションレジスタの範囲を越える場合は、DC ビットが 1 にセットされます。ガードビットがある場合でも、DC ビットはガードビットがない場合の結果を示します。したがって、ガードビットの部分が大きな数を表すために使用される場合は、DC ビットは常に 1 にセットされます。オーバーフローモードでの DC ビット生成の例を図 3.13 に示します。

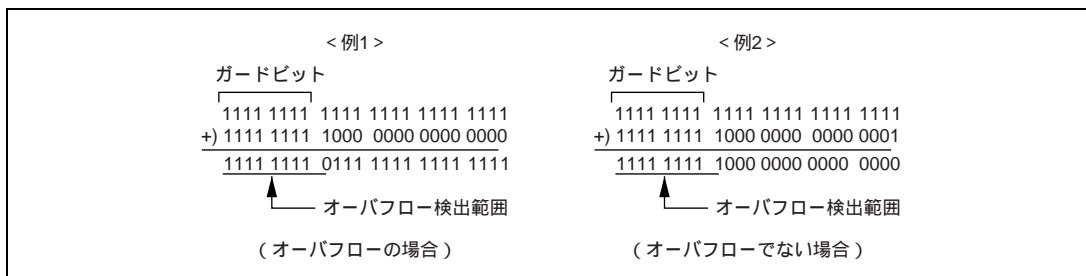


図 3.13 オーバフローモードでの DC ビット生成の例

(5) 符号付き大モード (CS [2:0]=B'100 の場合)

DC ビットは、比較演算 PCMP の結果、ソース 1 データ (符号付き) がソース 2 データ (符号付き) より大きいかどうかを示します。したがって、PCMP 演算は、条件付き演算をこの条件モードで実行するときにあらかじめ実行する必要があります。ソース 1 データがソース 2 データより大きい場合は、比較演算の結果は普通正の値なので、このモードは前述の負値モードに類似しています。ただし、ソース 1 データがソース 2 データより大きくても、比較演算の結果がガードビットを含めたデスティネーションオペランドの範囲を越える場合 (「オーバーレンジ」と呼ぶ) は、結果の符号ビットは負の値を示します。この条件モードでは、この特殊な場合を考慮した上で DC ビットを更新します。次の式は、この条件を得る定義を示します。

$$DC = \sim\{(\text{負値} \wedge \text{オーバーレンジ}) \mid \text{ゼロ値}\}$$

PCMP 演算をこの条件モードで実行する場合は、DC ビットの結果は CPU 命令の CMP/GT 演算の T ビットの結果と同じです。

(6) 符号付き以上モード (CS [2:0]=B'101 の場合)

DC ビットは、比較演算 PCMP の結果、ソース 1 データ (符号付き) がソース 2 データ (符号付き) 以上であるかどうかを示します。したがって、PCMP 演算は、条件付き演算をこの条件モードで実行するときにあらかじめ実行する必要があります。このモードは前述の「符号付き大モード」と類似していますが、このモードには等

しい場合も含まれます。次の式は、この条件を得る定義を示します。

$$DC = \sim(\text{負値} \wedge \text{オーバーレンジ})$$

PCMP 演算をこの条件モードで実行する場合は、DC ビットの結果は CPU 命令の CMP/GE 演算の T ビット結果と同じです。

N ビットは、CS [2:0] ビットが負値モードとしてセットされる DC ビットと常と同じ状態を示します。上記の負値モード部分を参照してください。Z ビットは、CS [2:0] ビットがゼロ値モードとしてセットされる DC ビットと常と同じ状態を示します。上記のゼロ値モード部分を参照してください。V ビットは、CS [2:0] ビットがオーバーフローモードとしてセットされる DC ビットと常と同じ状態を示します。上記のオーバーフローモード部分を参照してください。GT ビットは、CS [2:0] ビットが符号付き大モードとしてセットされる DC ビットと常と同じ状態を示します。上記の符号付き大モード部分を参照してください。

【注】 DC ビットは、PADDC と PSUBC 命令では、CS[2:0]の状態に関係なく常にキャリー/ボローモードとなります。

- オーバフローの防止機能

SR レジスタの S ビットは、DSP ユニットのどの ALU 固定小数点算術演算に対しても有効です。詳細については、「3.5.11 オーバフロー防止機能」を参照してください。

3.5.5 ALU 整数演算

図 3.14 に ALU 整数演算フローを示します。表 3.23 にこの演算の種類を示します。各オペランドのレジスタとの対応は、表 3.22 に示した ALU 固定小数点算術演算と同じです。

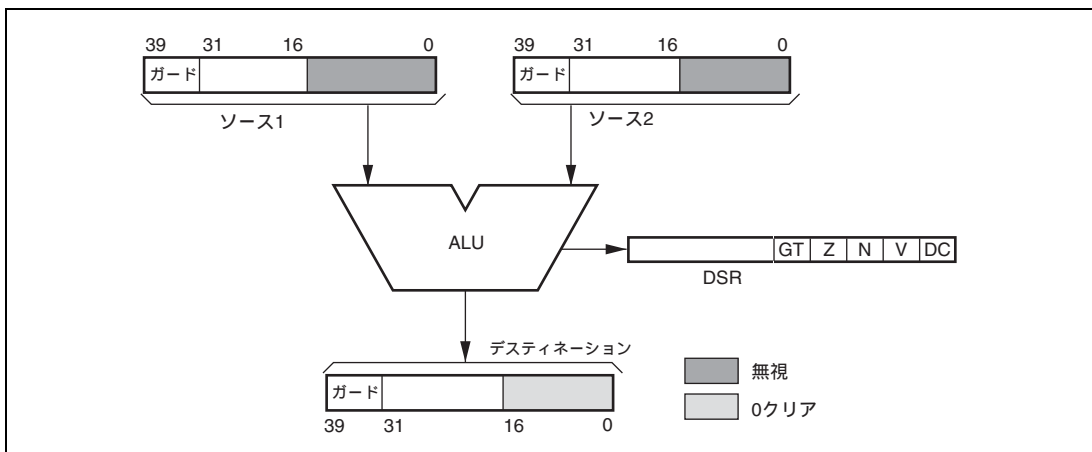


図 3.14 ALU 整数演算フロー

3. DSP 演算ユニット

表 3.23 ALU 整数演算の種類

ニーモニック	機能	ソース 1	ソース 2	デスティネーション
PINC	1 ずつインクリメント	Sx	+ 1	Dz
		+ 1	Sy	Dz
PDEC	1 ずつデクリメント	Sx	- 1	Dz
		- 1	Sy	Dz

【注】 ALU 整数演算は、基本的に 24 ビット演算、すなわち上位 16 ビットの基本精度および 8 ビットのガードビット部分からなります。したがって、ガードビット部分を提供していないレジスタをソースオペランドに指定すると、符号ビットはガードビット部分にコピーされます。ガードビット部分を提供していないレジスタをデスティネーションオペランドに指定すると、演算結果のガードビットを除いた上位ワードがデスティネーションレジスタの上位ワードに入力されます。

ALU 整数演算では、ソースオペランドの下位ワードは無視され、デスティネーションオペランドの下位ワードは自動的にクリアされます。ガードビット部分がサポートされている場合は、ALU 整数演算で有効です。その他は、基本的に ALU 固定小数点算術演算と同じです。ただし、表 3.23 に示すように、この種の演算は 2 種類の命令しか提供されません。したがって、第 2 オペランドは、実質的には +1 か -1 かのいずれかとなります。ワードデータを DSP ユニットのレジスタに読み込むと、上位ワードデータとして入力されます。ガードビットがあるレジスタをオペランドに指定すると、ガードビットも有効です。これらの演算は、ALU 固定小数点算術演算同様、図 3.10 に示すように DSP ステージで実行されます。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

ALU 整数演算を実行するたびに、DSR レジスタの DC、N、Z、V、および GT ビットは、基本的に演算結果に従って更新されます。これは ALU 固定小数点算術演算と同じですが、各ソースオペランドとデスティネーションオペランドの下位ワードはそれらを生成するためには使用しません。詳細については、「3.5.4 ALU 固定小数点算術演算」を参照してください。

条件付き命令の場合は、指定した条件が真であり演算が実行されてもこれらのビットは更新されません。無条件命令の場合、これらは、演算結果に従って常に更新されます。詳細については、「3.5.4 ALU 固定小数点算術演算」を参照してください。

- オーバフローの防止機能

SR レジスタの S ビットは、DSP ユニットのすべての ALU 整数演算で有効です。詳細については、「3.5.11 オーバフロー防止機能」を参照してください。

3.5.6 ALU 論理演算

図 3.15 に ALU 論理演算フローを示します。表 3.24 にこの演算の種類を示します。各オペランドとレジスタの対応は、表 3.21 に示した ALU 固定小数点算術演算と同じです。

ALU 論理演算は、レジスタ間で実行します。各ソースオペランドおよびデスティネーションオペランドは、DSP レジスタの 1 つから独立して選択されます。図 3.15 に示すように、この種の演算は、各オペランドの上位ワードのみを使用します。ソースオペランドの下位ワードとガードビットは無視され、デスティネーションオペランド

の下位ワードとガードビットは自動的にクリアされます。これらの演算もまた図 3.10 に示すように DSP ステージで実行されます。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

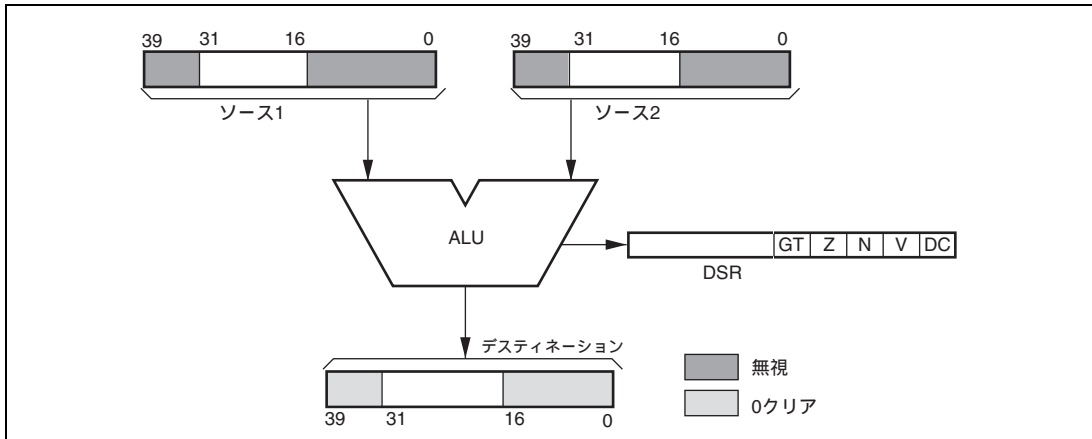


図 3.15 ALU 論理演算フロー

表 3.24 ALU 論理演算の種類

ニーモニック	機能	ソース 1	ソース 2	デスティネーション
PAND	論理 AND	Sx	Sy	Dz
POR	論理 OR	Sx	Sy	Dz
PXOR	論理排他的 OR	Sx	Sy	Dz

ALU 論理演算を実行するたびに、DSR レジスタの DC、N、Z、V、および GT ビットは、基本的に演算結果に従って更新されます。条件付き命令の場合は、指定条件が真で演算が実行されてもこれらのビットは更新されません。無条件命令の場合は、これらは演算結果に従って常に更新されます。DC ビットの定義は、DSR レジスタの CS[2:0]ビット (DC ビット状態選択ビット) で指定します。DC ビットの結果は、次のとおりです。

(1) キャリー / ボローモード (CS [2:0] = B'000 の場合)

DCビットは常に0にクリアされます。

(2) 負値モード (CS [2:0] = B'001 の場合)

演算結果のビット31の値がDCビットに読み込まれます。

(3) ゼロ値モード (CS [2:0] = B'010 の場合)

演算結果がゼロのときDCビットは1にセットされ、それ以外は0にクリアされます。

(4) オーバフローモード (CS [2:0] = B'011 の場合)

DCビットは常に0にクリアされます。

3. DSP 演算ユニット

(5) 符号付き大モード (CS [2:0] = B'100 の場合)

DCビットは常に0にクリアされます。

(6) 符号付き以上モード (CS [2:0] = B'101 の場合)

DCビットは常に0にクリアされます。

N ビットは、CS [2:0]ビットが負値モードとしてセットされる DC ビットと常に同じ状態を示します。上記の負値モード部分を参照してください。Z ビットは、CS [2:0]ビットがゼロ値モードとしてセットされる DC ビットと常に同じ状態を示します。上記のゼロ値モード部分を参照してください。V ビットは、CS [2:0]ビットがオーバフローモードとしてセットされる DC ビットと常に同じ状態を示します。上記のオーバフローモード部分を参照してください。GT ビットは、CS [2:0]ビットが符号付き大モードとしてセットされる DC ビットと常に同じ状態を示します。上記の符号付き大モード部分を参照してください。

3.5.7 固定小数点乗算

図 3.16 に固定小数点乗算命令のフローを示します。表 3.25 にこの演算の種類を示します。表 3.26 に各オペランドとレジスタの対応を示します。DSP ユニットの乗算は、シングルワード符号付き単精度乗算です。固定小数点乗算は、図 3.10 に示すように DSP ステージで実行されます。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

倍精度乗算が必要な場合は、CPU のダブルワード乗算命令を活用することができます。

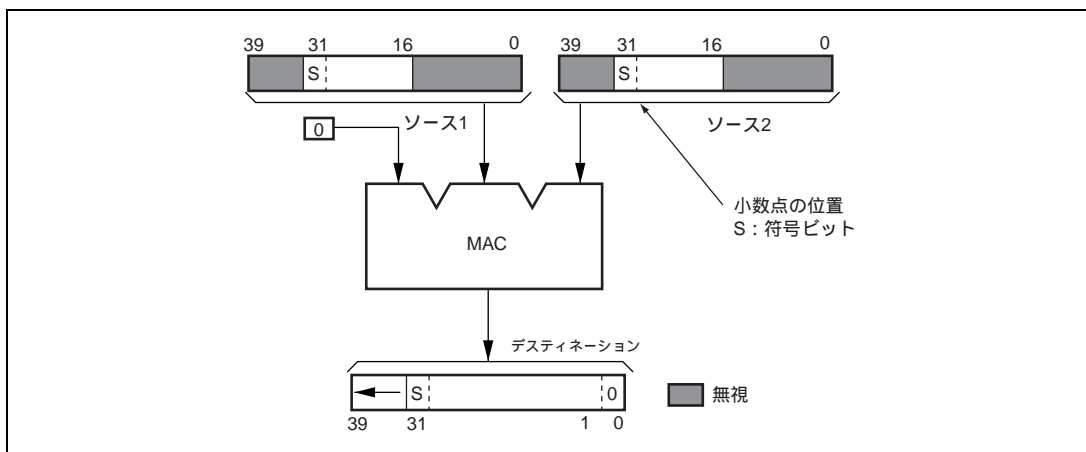


図 3.16 固定小数点乗算フロー

表 3.25 固定小数点乗算の種類

ニーモニック	機能	ソース 1	ソース 2	デスティネーション
PMULS	符号付き乗算	Se	Sf	Dg

表 3.26 オペランドの柔軟性

レジスタ	Se	Sf	Dg
A0			Yes
A1	Yes	Yes	Yes
M0			Yes
M1			Yes
X0	Yes	Yes	
X1	Yes		
Y0	Yes	Yes	
Y1		Yes	

【注】 乗算は、基本的に 32 ビットの演算結果を生成します。したがって、ガードビット部分を提供するレジスタをデスティネーションオペランドに指定するとガードビット部分には、演算結果のビット 31 がコピーされます。

DSP ユニット側の乗算は、整数ではなく固定小数点算術演算です。したがって、乗数および被乗数それぞれの上位ワードが図 3.16 に示すように MAC ユニットに入力されます。CPU 命令の乗算では、両ソースオペランドの下位ワードが MAC ユニットに入力されます。演算結果も CPU 命令の場合とは異なります。CPU 命令の乗算結果は、デスティネーションの LSB に揃えられますが、DSP ユニットの固定小数点乗算結果は MSB に揃えられ、固定小数点乗算結果の LSB は常に 0 になります。

この固定小数点乗算は、1 サイクルで実行されます。乗算演算は、常に無条件で実行されますが、DSR レジスタの DC、N、Z、V、および GT の条件コードビットには影響しません。

- オーバフローの防止機能

SR レジスタの S ビットは DSP ユニットのこの乗算に対して有効です。詳細については、「3.5.11 オーバフロー防止機能」を参照してください。

S ビットが 0 の場合は、 $H'8000 * H'8000 ((-1.0) * (-1.0))$ 演算を符号付き固定小数点乗算として実行するときだけオーバーフローが発生します。結果は、 $H'00\ 8000\ 0000$ ですが、 $(+1.0)$ を意味しません。S ビットが 1 の場合は、オーバーフロー防止機能が働いて結果は $H'00\ 7FFF\ FFFF$ となります。

3.5.8 シフト演算

シフト演算は、パレルシフトし、シフト量オペランドとしてレジスタ値またはイミディエイト値を使用することができます。他のソースオペランドとデスティネーションオペランドは、レジスタで指定します。シフト演算には、算術シフトおよび論理シフトの 2 種類があります。表 3.27 にこの演算の種類を示します。イミディエイトオペランドを除き、各オペランドのレジスタとの対応は、表 3.21 に示すように ALU 固定小数点算術演算と同じです。

3. DSP 演算ユニット

表 3.27 シフト演算の種類

ニーモニック	機能	ソース 1	ソース 2	デスティネーション
PSHA Sx, Sy, Dz	算術シフト	Sx	Sy	Dz
PSHL Sx, Sy, Dz	論理シフト	Sx	Sy	Dz
PSHA #Imm1, Dz	イミディエイト付き 算術シフト	Dz	Imm1	Dz
PSHL #Imm2, Dz	イミディエイト付き 論理シフト	Dz	Imm2	Dz

-32<=Imm1<=+32₇、-16<=Imm2<=+16

(1) 算術シフト：

図 3.17 に算術シフト演算フローを示します。

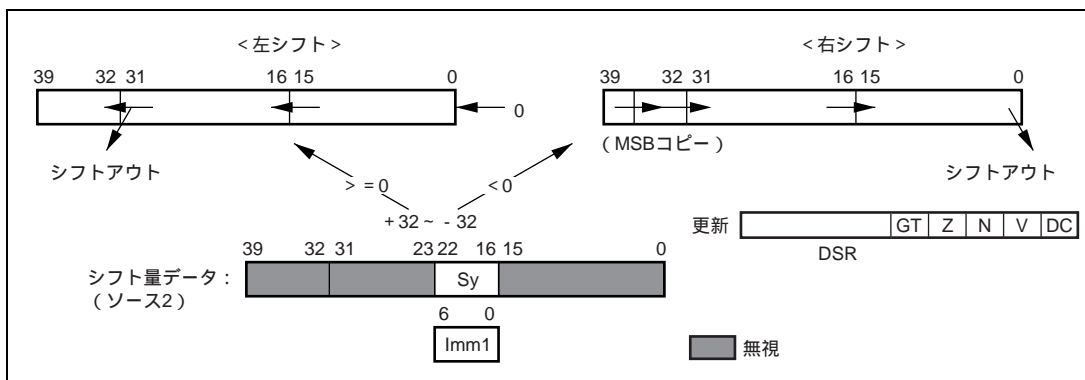


図 3.17 算術シフト演算フロー

【注】 算術シフト演算は、基本的に 40 ビット演算、すなわち 32 ビットの基本精度、8 ビットのガードビット部分から構成されます。したがって、ガードビット部分を提供していないレジスタをソースオペランドに指定すると、符号ビットがガードビット部分にコピーされます。ガードビット部分を提供していないレジスタをデスティネーションオペランドに指定すると、演算結果の下位 32 ビットがデスティネーションレジスタに入力されます。

この算術シフト演算においては、ソース 1 オペランドとデスティネーションオペランドは全ビット有効になります。シフト量は、整数部分としてソース 2 オペランドで指定します。ソース 2 オペランドは、レジスタまたはイミディエイトオペランドで指定することができます。利用可能なシフト範囲は、-32 から +32 までです。ここで負の値は右シフト、正の値は左シフトを意味します。ソース 2 オペランドとしては、-64 から +63 までを指定することができますが、無効なシフト値が指定された場合は、結果は保証されません。イミディエイトオペランド命令を持つシフトの場合は、ソース 1 オペランドはデスティネーションのレジスタと同じレジスタでなければなりません。この演算は、ALU 固定小数点算術演算同様に、図 3.10 に示すような DSP ステージで実行します。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

算術シフト演算を実行するたびに、DSR レジスタの DC、N、Z、V、および GT ビットは、基本的に演算結果に

従って更新されます。ただし、条件付き命令の場合は、指定条件が真で演算が実行されてもこれらのビットは更新されません。無条件の命令の場合は、これらは演算結果に従って常に更新されます。DC ビットの定義は DSR レジスタの CS[2:0]ビット (DC ビット状態選択ビット) で指定します。DC ビットの結果は、次のとおりです。

1. キャリー / ボローモード (CS [2:0] = B'000の場合)

DCビットは、演算結果として最後にシフトアウトしたデータを示します。

2. 負値モード (CS [2:0] = B'001の場合)

DCビットは、演算結果が負の値のとき1にセットされ、ゼロまたは正の値のときに0クリアされます。

3. ゼロ値モード (CS [2:0] = B'010の場合)

DCビットは、演算結果がゼロのとき1にセットされます。それ以外は、0クリアされます。

4. オーバフローモード (CS [2:0] = B'011の場合)

オーバーフローが発生したときに1にセットされます。

5. 符号付き大モード (CS [2:0] = B'100の場合)

DCビットは、常に0にクリアされます。

6. 符号付き以上モード (CS [2:0] = B'101の場合)

DCビットは、常に0にクリアされます。

N ビットは、CS [2:0]ビットが負値モードとしてセットされる DC ビットと常に同じ状態を示します。上記の負値モード部分を参照してください。Z ビットは、CS [2:0]ビットがゼロ値モードとしてセットされる DC ビットと常に同じ状態を示します。上記のゼロ値モード部分を参照してください。V ビットは、CS [2:0]ビットがオーバーフローモードとしてセットされる DC ビットと常に同じ状態を示します。上記のオーバーフローモード部分を参照してください。GT ビットは、CS [2:0]ビットが符号付き大モードとしてセットされる DC ビットと常に同じ状態を示します。上記の符号付き大モード部分を参照してください。

- オーバフローの防止機能

SR レジスタの S ビットは、DSP ユニットのどの算術シフト演算に対しても有効です。詳細については、「3.5.11 オーバフロー防止機能」を参照してください。

3. DSP 演算ユニット

(2) 論理シフト :

図 3.18 に論理シフト演算フローを示します。

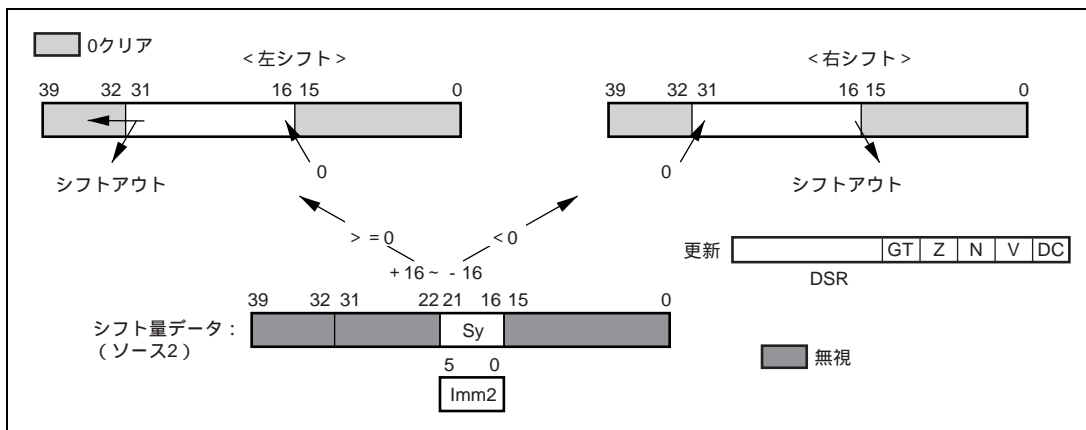


図 3.18 論理シフト演算フロー

図 3.18 に示すように、論理シフト演算は、ソース 1 の上位ワードとデスティネーションオペランドを使用します。ソースオペランドの下位ワードとガードビット部分は無視され、デスティネーションオペランドの下位ワードとガードビット部分は、ALU 論理演算同様、0 にクリアされます。シフト量は、整数データとしてソース 2 オペランドに指定します。ソース 2 オペランドでは、レジスタまたはイミディエイトオペランドに指定できます。利用可能なシフト範囲は、-16 から +16 です。ここで、負の値は右シフト、正の値は左シフトを意味します。任意のソース 2 オペランドは、-32 から +31 までを指定することができますが、無効なシフト値を指定すると、結果は保証されません。イミディエイトオペランド命令を持つシフトの場合は、ソース 1 オペランドはデスティネーションのレジスタと同じレジスタでなければなりません。これらの演算は図 3.10 に示すように DSP ステージで実行します。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

論理シフト演算を実行するたびに、DSR レジスタの DC、N、Z、V、および GT ビットは、基本的に演算結果に従って更新されます。条件付き演算の場合は、指定条件が真で演算が実行されてもこれらのビットは更新されません。無条件の演算の場合は、これらは演算結果で常に更新されます。DC ビットの定義は、DSR レジスタの CS[2:0] ビット (DC ビット状態選択ビット) で指定します。DC ビットの結果は、次のとおりです。

1. キャリー / ボローモード (CS [2:0] = B'000 の場合)

DC ビットは、演算結果として最後にシフトアウトしたデータを示します。

2. 負値モード (CS [2:0] = B'001 の場合)

DC ビットは、演算結果のビット 31 の値が格納されます。

3. ゼロ値モード (CS [2:0] = B'010 の場合)

DC ビットは、演算結果がゼロのとき 1 にセットされます。それ以外は、0 にクリアされます。

4. オーバフローモード (CS [2:0] = B'011の場合)

DCビットは、常に0にクリアされます。

5. 符号付き大モード (CS [2:0] = B'100の場合)

DCビットは、常に0にクリアされます。

6. 符号付き以上モード (CS [2:0] = B'101の場合)

DCビットは、常に0にクリアされます。

N ビットは、CS [2:0]ビットが負値モードとしてセットされる DC ビットと常に同じ状態を示します。上記の負値部分を参照してください。Z ビットは、CS [2:0]ビットがゼロ値モードとしてセットされる DC ビットと常に同じ状態を示します。上記のゼロ値モード部分を参照してください。V ビットは、CS [2:0]ビットがオーバフローモードとしてセットされる DC ビットと常に同じ状態を示しますが、この演算では常にクリアされます。GT ビットも同じです。

3.5.9 MSB 検出命令

MSB 検出命令 (PDMSB : Detect Most Significant Bit) は、正規化のためのシフト量を計算するために使用されます。図 3.19 に PDMSB 命令のフローを、表 3.28 に演算の定義を示します。表 3.29 にこの演算の種類を示します。各オペランドのレジスタとの対応は、表 3.21 に示した ALU 固定小数点算術演算と同じです。

【注】 MSB 検出命令の結果は、ALU 整数演算と同様、基本的に 24 ビット、すなわち上位 16 ビットの基本精度と 8 ビットのガードビット部分です。ガードビット部分を提供していないレジスタをデスティネーションオペランドに指定すると、演算結果の上位ワードがデスティネーションレジスタに入力されます。

図 3.19 に示すように、PDMSB 命令はソースオペランドとしてフルサイズのデータを使用しますが、正規化用のシフト量データは「3.5.8 シフト演算」で述べたように整数データでなければならないので、デスティネーションオペランドは整数演算結果と見なされます。これらの演算は、図 3.10 に示すように DSP ステージで実行されます。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

PDMSB 命令を実行するたびに、DSR レジスタの DC、N、Z、V、および GT ビットは、基本的に演算結果に従って更新されます。条件付き命令の場合は、指定条件が真で演算が実行されてもこれらのビットは更新されません。無条件命令の場合は、これらは演算結果で常に更新されます。

3. DSP 演算ユニット

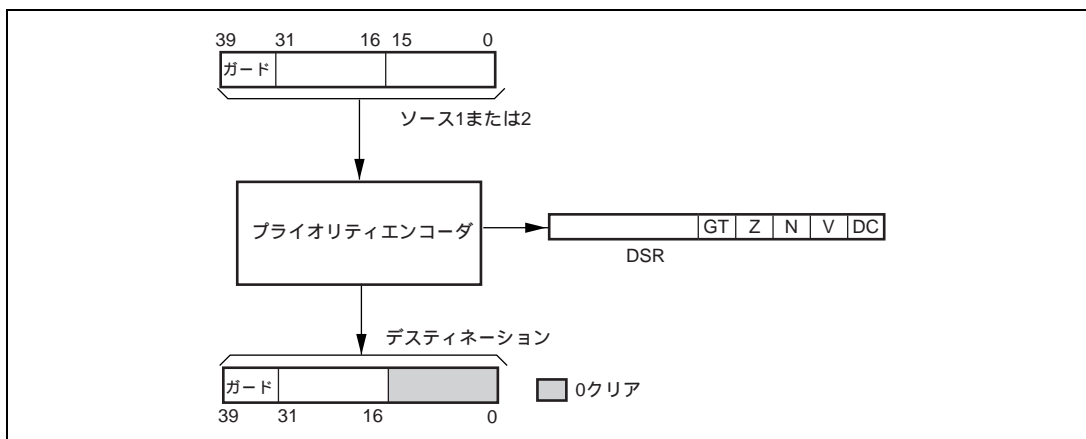


図 3.19 PDMSB 演算フロー

DC ビットの定義は DSR レジスタの CS[2:0] ビット (DC ビット状態選択ビット) で選択します。DC ビットの結果は、次のとおりです。

(1) キャリー/ボローモード (CS [2:0] = B'000 の場合)

DCビットは、常に0にクリアされます。

(2) 負値モード (CS [2:0] = B'001 の場合)

DCビットは、演算結果が負の値のとき1にセットされ、ゼロまたは正の値のとき0にクリアされます。

(3) ゼロ値モード (CS [2:0] = B'010 の場合)

DCビットは、演算結果がゼロのとき1にセットされます。それ以外は、0にクリアされます。

(4) オーバフローモード (CS [2:0] = B'011 の場合)

DCビットは、常に0にクリアされます。

(5) 符号付き大モード (CS [2:0] = B'100 の場合)

DCビットは、演算結果が正の値のときに1にセットされます。それ以外は、0にクリアされます。

(6) 符号付き以上モード (CS [2:0] = B'101 の場合)

DCビットは、演算結果が正またはゼロのとき1にセットされます。それ以外は、0にクリアされます。

表 3.28 PDMSB 命令の定義

ソースデータ														DST 用の結果									
ガードビット					上位ワード					下位ワード				ガードビット	上位ワード								
39	38	-	33	32	31	30	29	28	-	3	2	1	0	39-32	31-22	21	20	19	18	17	16	10進数	
0	0	-	0	0	0	0	0	0	-	0	0	0	0	all 0	all 0	0	1	1	1	1	1	1	+31
0	0	-	0	0	0	0	0	0	-	0	0	0	0	all 0	all 0	0	1	1	1	1	0	0	+30
0	0	-	0	0	0	0	0	0	-	0	0	1	*	all 0	all 0	0	1	1	1	0	1	0	+29
0	0	-	0	0	0	0	0	0	-	0	1	*	*	all 0	all 0	0	1	1	1	0	0	0	+28
:					:								:										
0	0	-	0	0	0	0	0	1	-	*	*	*	*	all 0	all 0	0	0	0	0	1	0	0	+2
0	0	-	0	0	0	0	1	*	-	*	*	*	*	all 0	all 0	0	0	0	0	0	1	0	+1
0	0	-	0	0	0	1	*	*	-	*	*	*	*	all 0	all 0	0	0	0	0	0	0	0	0
0	0	-	0	0	1	*	*	*	-	*	*	*	*	all 1	all 1	1	1	1	1	1	1	1	-1
0	0	-	0	1	*	*	*	*	-	*	*	*	*	all 1	all 1	1	1	1	1	1	0	0	-2
:					:								:										
0	1	-	*	*	*	*	*	*	-	*	*	*	*	all 1	all 1	1	1	1	0	0	0	0	-8
1	0	-	*	*	*	*	*	*	-	*	*	*	*	all 1	all 1	1	1	1	0	0	0	0	-8
:					:								:										
1	1	-	1	0	*	*	*	*	-	*	*	*	*	all 1	all 1	1	1	1	1	1	0	0	-2
1	1	-	1	1	0	*	*	*	-	*	*	*	*	all 1	all 1	1	1	1	1	1	1	1	-1
1	1	-	1	1	1	0	*	*	-	*	*	*	*	all 0	all 0	0	0	0	0	0	0	0	0
1	1	-	1	1	1	1	0	*	-	*	*	*	*	all 0	all 0	0	0	0	0	0	1	0	+1
1	1	-	1	1	1	1	1	0	-	*	*	*	*	all 0	all 0	0	0	0	0	1	0	0	+2
:					:								:										
1	1	-	1	1	1	1	1	1	-	1	0	*	*	all 0	all 0	0	1	1	1	0	0	0	+28
1	1	-	1	1	1	1	1	1	-	1	1	0	*	all 0	all 0	0	1	1	1	0	1	0	+29
1	1	-	1	1	1	1	1	1	-	1	1	1	0	all 0	all 0	0	1	1	1	1	0	0	+30
1	1	-	1	1	1	1	1	1	-	1	1	1	1	all 0	all 0	0	1	1	1	1	1	1	+31

【注】 * Don't care ビットを意味します。

表 3.29 PDMSB 命令の種類

ニーモニック	機能	ソース 1	ソース 2	デスティネーション
PDMSB	MSB 検出	Sx	-	Dz
		-	Sy	Dz

N ビットは、CS [2:0] ビットが負値モードとしてセットされる DC ビットと常に同じ状態を示します。上記の負

3. DSP 演算ユニット

値モード部分を参照してください。Z ビットは、CS [2:0]ビットがゼロ値モードとしてセットされる DC ビットと常と同じ状態を示します。上記のゼロ値モード部分を参照してください。V ビットは、CS [2:0]ビットがオーバフローモードとしてセットされる DC ビットと常と同じ状態を示します。上記のオーバフローモード部分を参照してください。GT ビットは、CS [2:0]ビットが符号付き大モードとしてセットされる DC ビットと常と同じ状態を示します。上記の符号付き大モード部分を参照してください。

3.5.10 丸め演算

DSP ユニットの、32 ビットから 16 ビットに丸める丸め機能を提供します。ガードビットがある場合は、40 ビットから 24 ビットに丸めます。丸め命令を実行するときは、ソースオペランドに H'00008000 が加算された後、下位ワードが 0 クリアされます。図 3.20 に丸め演算フローを示します。図 3.21 に丸め演算の定義を示します。また、表 3.30 に丸め演算の種類を示します。各オペランドのレジスタとの対応は、表 3.21 に示した ALU 固定小数点算術演算と同じです。

図 3.21 に示すように、丸め演算は、ソースオペランド、デスティネーションオペランド両方に対してフルサイズデータを使用します。これらの演算は、図 3.10 に示すような DSP ステージで実行されます。DSP ステージは、メモリアクセスを行う MA ステージと同ステージです。

丸め演算を実行するたびに、DSR レジスタの DC、N、Z、V、および GT ビットは、基本的に演算結果に従って更新されます。条件付き命令の場合は、指定条件が真で演算が実行されてもこれらのビットは更新されません。無条件命令の場合は、これらは演算結果で常に更新されます。DC ビットの定義は、DSR レジスタの CS[2:0]ビット (DC ビット状態選択ビット) で指定します。これらの状態コードビットの結果は ALU 固定小数点算術演算と同じです。

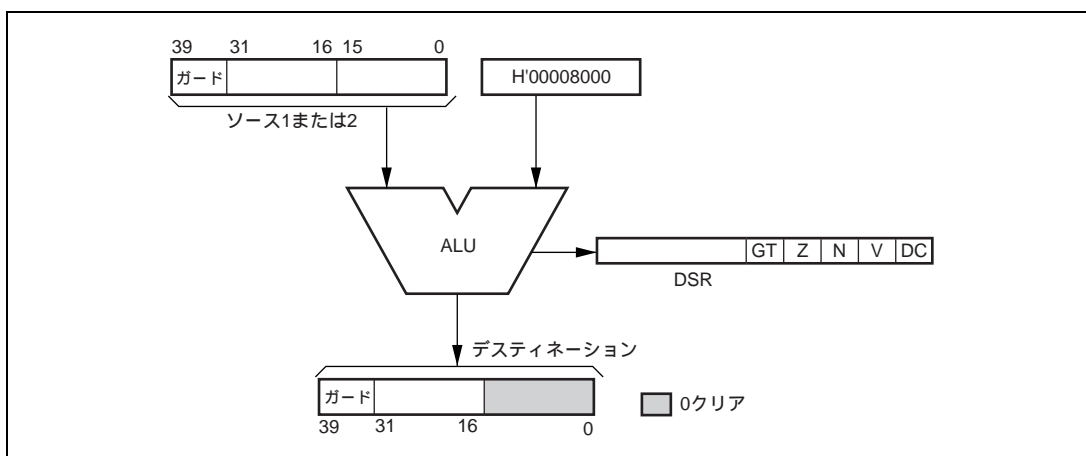


図 3.20 丸め演算フロー

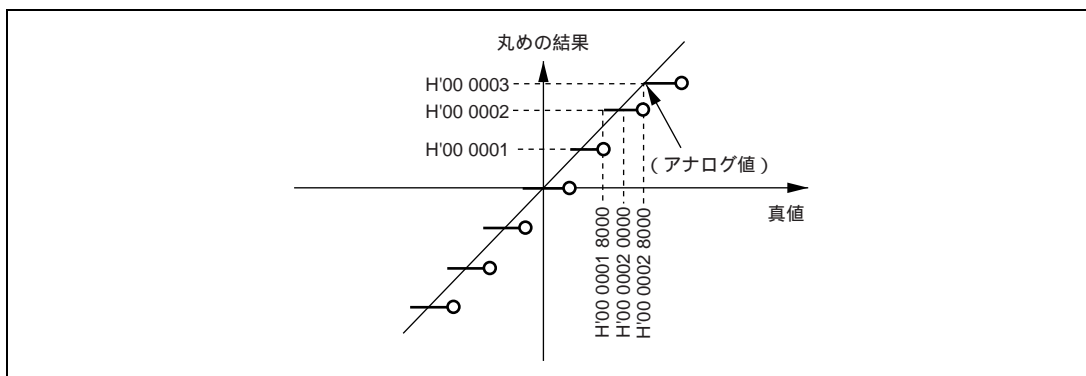


図 3.21 丸め演算の定義

表 3.30 丸め演算の種類

ニーモニック	機能	ソース 1	ソース 2	デスティネーション
PRND	丸め	Sx	-	Dz
		-	Sy	Dz

- オーバフロー防止機能

SR レジスタの S ビットは DSP ユニットの任意の丸め演算で有効です。詳細については、「3.5.11 オーバフロー防止機能」を参照してください。

3.5.11 オーバフロー防止機能

SR レジスタの S ビットは、オーバフロー防止機能イネーブルビットとして使用します。この S ビットは、CPU 命令の乗算および MAC 演算を含め、DSP ユニットで実行する算術演算に有効です。演算結果がガードビット部分のない 2 の補数の表記範囲を超える場合は、算術演算はオーバフローします。表 3.31 に「3.5.7 固定小数点乗算」で説明した固定小数点乗算を含めて、固定小数点算術演算に対するオーバフロー保護の定義を示します。表 3.32 には整数算術演算に対するオーバフロー防止機能の定義を示します。整数算術演算の飽和値の下位ワードは Don't care です。下位ワードの値は、保証されません。

オーバフロー防止機能が有効になっているときは、オーバフローは発生することはありません。したがって、V ビットは、0 にクリアされます。オーバフローモードを CS [2:0] ビットで選択するときも DC ビットは 0 にクリアされます。

表 3.31 固定小数点算術演算用オーバフロー防止機能の定義

符号	オーバフロー条件	固定値	16 進表記
正	結果 $> 1 - 2^{-31}$	$1 - 2^{-31}$	00 7FFF FFFF
負	結果 < -1	-1	FF 8000 0000

3. DSP 演算ユニット

表 3.32 整数算術演算用オーバーフロー防止機能の定義

符号	オーバーフロー条件	固定値	16 進表記
正	結果 > $2^{15} - 1$	$2^{15} - 1$	00 7FFF ****
負	結果 < $- 2^{15}$	$- 2^{15}$	FF 8000 ****

【注】 * Don't care を意味します。

3.5.12 ローカルデータ移動命令

本 LSI には、CPU 命令の乗算 / 積和演算 (MAC) をサポートするための MACL と MACH の 2 つレジスタがあります。これらのレジスタは、他の DSP レジスタとのローカルデータ移動命令により、テンポラリレジスタとして活用することができます。図 3.22 にローカルデータ移動命令のフローを示します。表 3.33 にはこの命令の種類を示します。

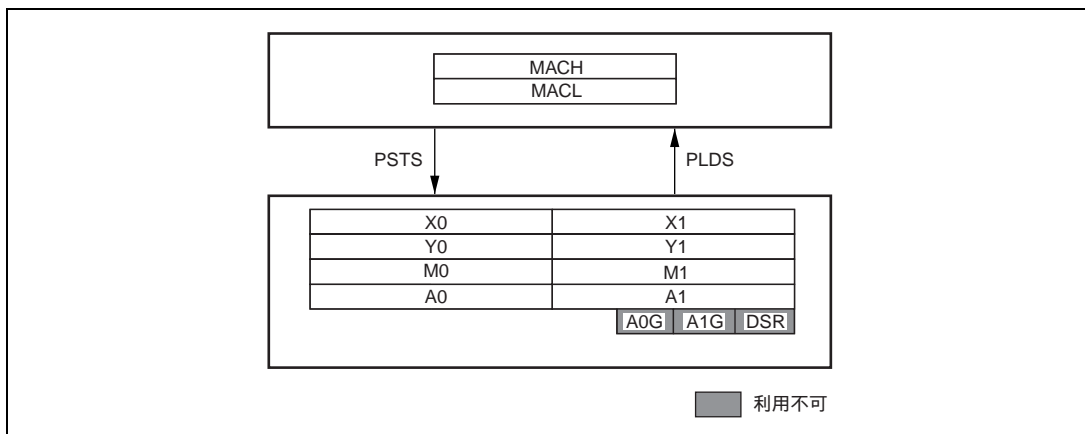


図 3.22 ローカルデータ移動命令のフロー

表 3.33 ローカルデータ移動命令の種類

ニーモニック	機能	オペランド
PLDS	DSP レジスタから MACL/H へのデータ移動	Dz
PSTS	MACL/H から DSP レジスタへのデータ移動	Dz

この命令は、他の転送命令と非常に似ています。A0 および A1 レジスタのいずれかを PSTS のデスティネーションオペランドとして指定すると、符号ビットが該当するガードビット部分 A0G または A1G に符号拡張されて格納されます。命令結果にかかわらず、DSR レジスタの DC ビットおよびその他の条件コードビットは、更新されません。この命令は、条件付きとしても動作します。ローカルデータ移動命令は、MOVX と MOVY で並列して指定することができます。

3.5.13 オペランドの競合

同一のデスティネーションオペランドを複数の DSP データ演算命令による並列処理で指定すると、データの競合が発生します。表 3.34 に各命令のオペランドとレジスタの対応を示します。

表 3.34 オペランド柔軟性

		Xメモリロード 命令			Yメモリロード 命令			6オペランド ALU命令			3オペランド 乗算命令			3オペランド ALU命令		
		Ax	Ix	Dx	Ay	Iy	Dy	Sx	Sy	Du	Se	Sf	Dg	Sx	Sy	Dz
DSP レジスタ	A0							*1		*2			*2	*1		*1
	A1							*1		*2	*1	*1	*2	*1		*1
	M0								*1				*1		*1	*1
	M1								*1				*1		*1	*1
	X0			*2				*1		*2	*1	*1		*1		*2
	X1			*2				*1		*2	*1			*1		*2
	Y0						*2		*1	*2	*1	*1			*1	*2
	Y1						*2		*1			*1			*1	*2

【注】 *1 オペランドに利用可能なレジスタ

*2 オペランドに利用可能なレジスタ（オペランド競合の場合）

オペランド競合問題には、3つの場合があります。

- ALU演算と乗算命令が同じデスティネーションオペランド（DuおよびDg）を指定する場合
- X側ロードとALU命令が同じデスティネーションオペランド（Dxおよび（DuまたはDz））を指定する場合
- Y側ロードとALU命令が同じデスティネーションオペランド（Dyおよび（DuまたはDz））を指定する場合

競合した場合の結果は、保証されません。

3. DSP 演算ユニット

3.6 DSP 拡張機能命令セット

3.6.1 CPU 拡張命令セット

表 3.35 DSP モード拡張システム制御命令

命令	命令コード	動作	実行 ステート	T ビット
SETRC #imm	10000010iiiiiii	imm RC (of SR)	1	-
SETRC Rn	0100nnnn00010100	Rn[11:0] RC(of SR)	1	-
LDRS @(disp,PC)	10001100ddddddd	(disp × 2+PC) RS	1	-
LDRE @(disp,PC)	10001110ddddddd	(disp × 2+PC) RE	1	-
STC MOD,Rn	0000nnnn01010010	MOD Rn	1	-
STC RS,Rn	0000nnnn01100010	RS Rn	1	-
STC RE,Rn	0000nnnn01110010	RE Rn	1	-
STS DSR,Rn	0000nnnn01101010	DSR Rn	1	-
STS A0,Rn	0000nnnn01111010	A0 Rn	1	-
STS X0,Rn	0000nnnn10001010	X0 Rn	1	-
STS X1,Rn	0000nnnn10011010	X1 Rn	1	-
STS Y0,Rn	0000nnnn10101010	Y0 Rn	1	-
STS Y1,Rn	0000nnnn10111010	Y1 Rn	1	-
STS.L DSR,@-Rn	0100nnnn01100010	Rn-4 Rn, DSR (Rn)	1	-
STS.L A0,@-Rn	0100nnnn01110010	Rn-4 Rn, A0 (Rn)	1	-
STS.L X0,@-Rn	0100nnnn10000010	Rn-4 Rn, X0 (Rn)	1	-
STS.L X1,@-Rn	0100nnnn10010010	Rn-4 Rn, X1 (Rn)	1	-
STS.L Y0,@-Rn	0100nnnn10100010	Rn-4 Rn, Y0 (Rn)	1	-
STS.L Y1,@-Rn	0100nnnn10110010	Rn-4 Rn, Y1 (Rn)	1	-
STC.L MOD,@-Rn	0100nnnn01010011	Rn-4 Rn, MOD (Rn)	1	-
STC.L RS,@-Rn	0100nnnn01100011	Rn-4 Rn, RS (Rn)	1	-
STC.L RE,@-Rn	0100nnnn01110011	Rn-4 Rn, RE (Rn)	1	-
LDS.L @Rn+,DSR	0100nnnn01100110	(Rn) DSR, Rn+4 Rn	1	-
LDS.L @Rn+,A0	0100nnnn01110110	(Rn) A0, Rn+4 Rn	1	-
LDS.L @Rn+,X0	0100nnnn10000110	(Rn) X0, Rn+4 Rn	1	-
LDS.L @Rn+,X1	0100nnnn10010110	(Rn) X1, Rn+4 Rn	1	-
LDS.L @Rn+,Y0	0100nnnn10100110	(Rn) Y0, Rn+4 Rn	1	-
LDS.L @Rn+,Y1	0100nnnn10110110	(Rn) Y1, Rn+4 Rn	1	-
LDC.L @Rn+,MOD	0100nnnn01010111	(Rn) MOD, Rn+4 Rn	4	-
LDC.L @Rn+,RS	0100nnnn01100111	(Rn) RS, Rn+4 Rn	4	-

命令	命令コード	動作	実行 ステート	Tビット
LDC.L @Rn+,RE	0100nnnn01110111	(Rn) RE, Rn+4 Rn	4	-
LDS Rn,DSR	0100nnnn01101010	Rn DSR	1	-
LDS Rn,A0	0100nnnn01111010	Rn A0	1	-
LDS Rn,X0	0100nnnn10001010	Rn X0	1	-
LDS Rn,X1	0100nnnn10011010	Rn X1	1	-
LDS Rn,Y0	0100nnnn10101010	Rn Y0	1	-
LDS Rn,Y1	0100nnnn10111010	Rn Y1	1	-
LDC Rn,MOD	0100nnnn01011110	Rn MOD	4	-
LDC Rn,RS	0100nnnn01101110	Rn RS	4	-
LDC Rn,RE	0100nnnn01111110	Rn RE	4	-

3.6.2 ダブルデータ転送命令セット

表 3.36 ダブルデータ転送命令

命令		命令コード	動作	実行 ステート	DC
Xメモリ データ 転送	NOPX	1111000*0*0*00**	Xメモリノークセス	1	-
	MOVX.W @Ax, Dx	111100A*D*0*01**	(Ax) DxのMSW, 0 DxのLSW	1	-
	MOVX.W @Ax+, Dx	111100A*D*0*10**	(Ax) DxのMSW, 0 DxのLSW, Ax+2 Ax	1	-
	MOVX.W @Ax+Ix, Dx	111100A*D*0*11**	(Ax) DxのMSW, 0 DxのLSW, Ax+Ix Ax	1	-
	MOVX.W Da, @Ax	111100A*D*1*01**	DaのMSW (Ax)	1	-
	MOVX.W Da, @Ax+	111100A*D*1*10**	DaのMSW (Ax), Ax+2 Ax	1	-
	MOVX.W Da, @Ax+Ix	111100A*D*1*11**	DaのMSW (Ax), Ax+Ix Ax	1	-
Yメモリ データ 転送	NOPY	111100*0*0*0**00	Yメモリノークセス	1	-
	MOVY.W @Ay, Dy	111100*A*D*0**01	(Ay) DyのMSW, 0 DyのLSW	1	-
	MOVY.W @Ay+, Dy	111100*A*D*0**10	(Ay) DyのMSW, 0 DyのLSW, Ay+2 Ay	1	-
	MOVY.W @Ay+Iy, Dy	111100*A*D*0**11	(Ay) DyのMSW, 0 DyのLSW, Ay+Iy Ay	1	-
	MOVY.W Da, @Ay	111100*A*D*1**01	DaのMSW (Ay)	1	-
	MOVY.W Da, @Ay+	111100*A*D*1**10	DaのMSW (Ay), Ay+2 Ay	1	-
	MOVY.W Da, @Ay+Iy	111100*A*D*1**11	DaのMSW (Ay), Ay+Iy Ay	1	-

3. DSP 演算ユニット

3.6.3 シングルデータ転送命令セット

表 3.37 シングルデータ転送命令

命令	命令コード	動作	実行 ステート	DC	区分
MOVS.W @-As, Ds	111101AADDDDD0000	As-2 As, (As) Ds の MSW, 0 Ds の LSW	1	-	
MOVS.W @As, Ds	111101AADDDDD0100	(As) Ds の MSW, 0 Ds の LSW	1	-	
MOVS.W @As+, Ds	111101AADDDDD1000	(As) Ds の MSW, 0 Ds の LSW, As+2 As	1	-	
MOVS.W @As+Ix, Ds	111101AADDDDD1100	(As) Ds の MSW, 0 Ds の LSW, As+Ix As	1	-	
MOVS.W Ds, @-As	111101AADDDDD0001	As-2 As, Ds の MSW (As)	1	-	*
MOVS.W Ds, @As	111101AADDDDD0101	Ds の MSW (As)	1	-	*
MOVS.W Ds, @As+	111101AADDDDD1001	Ds の MSW (As), As+2 As	1	-	*
MOVS.W Ds, @As+Ix	111101AADDDDD1101	Ds の MSW (As), As+Ix As	1	-	*
MOVS.L @-As, Ds	111101AADDDDD0010	As-4 As, (As) Ds	1	-	
MOVS.L @As, Ds	111101AADDDDD0110	(As) Ds	1	-	
MOVS.L @As+, Ds	111101AADDDDD1010	(As) Ds, As+4 As	1	-	
MOVS.L @As+Ix, Ds	111101AADDDDD1110	(As) Ds, As+Ix As	1	-	
MOVS.L Ds, @-As	111101AADDDDD0011	As-4 As, Ds (As)	1	-	
MOVS.L Ds, @As	111101AADDDDD0111	Ds (As)	1	-	
MOVS.L Ds, @As+	111101AADDDDD1011	Ds (As), As+4 As	1	-	
MOVS.L Ds, @As+Ix	111101AADDDDD1111	Ds (As), As+Ix As	1	-	

【注】 * ガードビットレジスタ A0G と A1G をソースオペランド Ds に指定した場合は、データは LDB[7:0]バスに出力され、符号ビットは上位ビット [31:8]に転記されます。

DSP データ転送のオペランドとレジスタとの対応を表 3.38 に示します。

表 3.38 DSP データ転送のオペランドとレジスタとの対応

レジスタ		Ax	Ix	Dx	Ay	Iy	Dy	Da	As	Ds
CPU レジスタ	R0									
	R1									
	R2 (As2)								Yes	
	R3 (As3)								Yes	
	R4 (Ax0, As0)	Yes							Yes	
	R5 (Ax1, As1)	Yes							Yes	
	R6 (Ay0)				Yes					
	R7 (Ay1)				Yes					
	R8 (Ix)		Yes							
	R9 (Iy)					Yes				
DSP レジスタ	A0							Yes		Yes
	A1							Yes		Yes
	M0									Yes
	M1									Yes
	X0			Yes						Yes
	X1			Yes						Yes
	Y0						Yes			Yes
	Y1						Yes			Yes
	A0G									Yes
	A1G									Yes

3. DSP 演算ユニット

3.6.4 DSP データ演算命令セット

表 3.39 DSP データ演算命令

命令	命令コード	動作	実行 状態	DC
PMULS Se, Sf, Dg	111110***** 0100eeff0000gg00	Se*Sf Dg (符号付き)	1	-
PADD Sx, Sy, Du PMULS Se, Sf, Dg	111110***** 0111eeffxxyygguu	Sx+Sy Du Se*Sf Dg (符号付き)	1	*
PSUB Sx, Sy, Du PMULS Se, Sf, Dg	111110***** 0110eeffxxyygguu	Sx-Sy Du Se*Sf Dg (符号付き)	1	*
PADD Sx, Sy, Dz	111110***** 10110001xxyyzzzz	Sx+Sy Dz	1	*
DCT PADD Sx, Sy, Dz	111110***** 10110010xxyyzzzz	If DC=1, Sx+Sy Dz If DC=0, nop	1	-
DCF PADD Sx, Sy, Dz	111110***** 10110011xxyyzzzz	If DC=0, Sx+Sy Dz If DC=1, nop	1	-
PSUB Sx, Sy, Dz	111110***** 10100001xxyyzzzz	Sx-Sy Dz	1	*
DCT PSUB Sx, Sy, Dz	111110***** 10100010xxyyzzzz	If DC=1, Sx-Sy Dz If DC=0, nop	1	-
DCF PSUB Sx, Sy, Dz	111110***** 10100011xxyyzzzz	If DC=0, Sx-Sy Dz If DC=1, nop	1	-
PSHA Sx, Sy, Dz	111110***** 10010001xxyyzzzz	If Sy>=0, Sx<<Sy Dz (算術シフト) If Sy<0, Sx>>Sy Dz	1	*
DCT PSHA Sx, Sy, Dz	111110***** 10010010xxyyzzzz	If DC=1 & Sy>=0, Sx<<Sy Dz(算術シフト) If DC=1 & Sy<0, Sx>>Sy Dz If DC=0, nop	1	-
DCF PSHA Sx, Sy, Dz	111110***** 10010011xxyyzzzz	If DC=0 & Sy>=0, Sx<<Sy Dz(算術シフト) If DC=0 & Sy<0, Sx>>Sy Dz If DC=1, nop	1	-
PSHL Sx, Sy, Dz	111110***** 10000001xxyyzzzz	If Sy>=0, Sx<<Sy Dz (論理シフト) If Sy<0, Sx>>Sy Dz	1	*
DCT PSHL Sx, Sy, Dz	111110***** 10000010xxyyzzzz	If DC=1 & Sy>=0, Sx<<Sy Dz(論理シフト) If DC=1 & Sy<0, Sx>>Sy Dz If DC=0, nop	1	-
DCF PSHL Sx, Sy, Dz	111110***** 10000011xxyyzzzz	If DC=0 & Sy>=0, Sx<<Sy Dz(論理シフト) If DC=0 & Sy<0, Sx>>Sy Dz If DC=1, nop	1	-
PCOPY Sx, Dz	111110***** 11011001xx00zzzz	Sx Dz	1	*

3. DSP 演算ユニット

命令	命令コード	動作	実行 状態	DC
PCOPY S_y, D_z	111110***** 1111100100yyzzzz	$S_y D_z$	1	*
DCT PCOPY S_x, D_z	111110***** 11011010xx00zzzz	If DC=1, $S_x D_z$ If DC=0, nop	1	-
DCT PCOPY S_y, D_z	111110***** 1111101000yyzzzz	If DC=1, $S_y D_z$ If DC=0, nop	1	-
DCF PCOPY S_x, D_z	111110***** 11011011xx00zzzz	If DC=0, $S_x D_z$ If DC=1, nop	1	-
DCF PCOPY S_y, D_z	111110***** 1111101100yyzzzz	If DC=0, $S_y D_z$ If DC=1, nop	1	-
PDMSB S_x, D_z	111110***** 10011101xx00zzzz	$S_x D_z$ 正規化のためのカウントシフト値	1	*
PDMSB S_y, D_z	111110***** 1011110100yyzzzz	$S_y D_z$ 正規化のためのカウントシフト値	1	*
DCT PDMSB S_x, D_z	111110***** 10011110xx00zzzz	If DC=1, 正規化のためのカウントシフト値 $S_x D_z$ If DC=0, nop	1	-
DCT PDMSB S_y, D_z	111110***** 1011111000yyzzzz	If DC=1, 正規化のためのカウントシフト値 $S_y D_z$ If DC=0, nop	1	-
DCF PDMSB S_x, D_z	111110***** 10011111xx00zzzz	If DC=0, 正規化のためのカウントシフト値 $S_x D_z$ If DC=1, nop	1	-
DCF PDMSB S_y, D_z	111110***** 1011111100yyzzzz	If DC=0, 正規化のためのカウントシフト値 $S_y D_z$ If DC=1, nop	1	-
PINC S_x, D_z	111110***** 10011001xx00zzzz	S_x の MSW+1 D_z	1	*
PINC S_y, D_z	111110***** 1011100100yyzzzz	S_y の MSW+1 D_z	1	*
DCT PINC S_x, D_z	111110***** 10011010xx00zzzz	If DC=1, S_x の MSW+1 D_z If DC=0, nop	1	-
DCT PINC S_y, D_z	111110***** 1011101000yyzzzz	If DC=1, S_y の MSW+1 D_z If DC=0, nop	1	-
DCF PINC S_x, D_z	111110***** 10011011xx00zzzz	If DC=0, S_x の MSW+1 D_z If DC=1, nop	1	-
DCF PINC S_y, D_z	111110***** 1011101100yyzzzz	If DC=0, S_y の MSW+1 D_z If DC=1, nop	1	-
PNEG S_x, D_z	111110***** 11001001xx00zzzz	0- $S_x D_z$	1	*

3. DSP 演算ユニット

命令	命令コード	動作	実行 ステート	DC
PNEG S_y, D_z	111110***** 1110100100yyzzzz	0- S_y D_z	1	*
DCT PNEG S_x, D_z	111110***** 11001010xx00zzzz	If DC=1, 0- S_x D_z If DC=0, nop	1	-
DCT PNEG S_y, D_z	111110***** 1110101000yyzzzz	If DC=1, 0- S_y D_z If DC=0, nop	1	-
DCF PNEG S_x, D_z	111110***** 11001011xx00zzzz	If DC=0, 0- S_x D_z If DC=1, nop	1	-
DCF PNEG S_y, D_z	111110***** 1110101100yyzzzz	If DC=0, 0- S_y D_z If DC=1, nop	1	-
POR S_x, S_y, D_z	111110***** 10110101xxyyzzzz	S_x S_y D_z	1	*
DCT POR S_x, S_y, D_z	111110***** 10110110xxyyzzzz	If DC=1, S_x S_y D_z If DC=0, nop	1	-
DCF POR S_x, S_y, D_z	111110***** 10110111xxyyzzzz	If DC=0, S_x S_y D_z If DC=1, nop	1	-
PAND S_x, S_y, D_z	111110***** 10010101xxyyzzzz	S_x & S_y D_z	1	*
DCT PAND S_x, S_y, D_z	111110***** 10010110xxyyzzzz	If DC=1, S_x & S_y D_z If DC=0, nop	1	-
DCF PAND S_x, S_y, D_z	111110***** 10010111xxyyzzzz	If DC=0, S_x & S_y D_z If DC=1, nop	1	-
PXOR S_x, S_y, D_z	111110***** 10100101xxyyzzzz	S_x ^ S_y D_z	1	*
DCT PXOR S_x, S_y, D_z	111110***** 10100110xxyyzzzz	If DC=1, S_x ^ S_y D_z If DC=0, nop	1	-
DCF PXOR S_x, S_y, D_z	111110***** 10100111xxyyzzzz	If DC=0, S_x ^ S_y D_z If DC=0, nop	1	-
PDEC S_x, D_z	111110***** 10001001xx00zzzz	S_x [39:16]-1 D_z	1	*
DCT PDEC S_x, D_z	111110***** 10001010xx00zzzz	If DC=1, S_x [39:16]-1 D_z If DC=0, nop	1	-
DCF PDEC S_x, D_z	111110***** 10001011xx00zzzz	If DC=0, S_x [39:16]-1 D_z If DC=1, nop	1	-
PDEC S_y, D_z	111110***** 1010100100yyzzzz	S_y [31:16]-1 D_z	1	*

3. DSP 演算ユニット

命令	命令コード	動作	実行 ステート	DC
DCT PDEC Sy, Dz	111110***** 1010101000yyzzzz	If DC=1, Sy [31:16]-1 Dz If DC=0, nop	1	-
DCF PDEC Sy, Dz	111110***** 1010101100yyzzzz	If DC=0, Sy [31:16]-1 Dz If DC=1, nop	1	-
PCLR Dz	111110***** 100011010000zzzz	h'00000000 Dz	1	*
DCT PCLR Dz	111110***** 100011100000zzzz	If DC=1, h'00000000 Dz If DC=0, nop	1	-
DCF PCLR Dz	111110***** 100011110000zzzz	If DC=0, h'00000000 Dz If DC=1, nop	1	-
PSHA #imm, Dz	111110***** 00010iiiiiiiizzzz	If imm>=0, Dz<<imm Dz (算術シフト) If imm<0, Dz>>imm Dz	1	*
PSHL #imm, Dz	111110***** 00000iiiiiiiizzzz	If imm>=0, Dz<<imm Dz (論理シフト) If imm<0, Dz>>imm Dz	1	*
PSTS MACH, Dz	111110***** 110011010000zzzz	MACH Dz	1	-
DCT PSTS MACH, Dz	111110***** 110011100000zzzz	If DC=1, MACH Dz If DC=0, nop	1	-
DCF PSTS MACH, Dz	111110***** 110011110000zzzz	If DC=0, MACH Dz If DC=1, nop	1	-
PSTS MACL, Dz	111110***** 110111010000zzzz	MACL Dz	1	-
DCT PSTS MACL, Dz	111110***** 110111100000zzzz	If DC=1, MACL Dz If DC=0, nop	1	-
DCF PSTS MACL, Dz	111110***** 110111110000zzzz	If DC=0, MACL Dz If DC=1, nop	1	-
PLDS Dz, MACH	111110***** 111011010000zzzz	Dz MACH	1	-
DCT PLDS Dz, MACH	111110***** 111011100000zzzz	If DC=1, Dz MACH If DC=0, nop	1	-
DCF PLDS Dz, MACH	111110***** 111011110000zzzz	If DC=0, Dz MACH If DC=1, nop	1	-
PLDS Dz, MACL	111110***** 111111010000zzzz	Dz MACL	1	-
DCT PLDS Dz, MACL	111110***** 111111100000zzzz	If DC=1, Dz MACL If DC=0, nop	1	-
DCF PLDS Dz, MACL	111110***** 111111110000zzzz	If DC=0, Dz MACL If DC=1, nop	1	-

3. DSP 演算ユニット

命令	命令コード	動作	実行 ステート	DC
PADDC Sx, Sy, Dz	111110***** 10110000xxyyzzzz	Sx+Sy+DC Dz キャリー DC	1	キャリー
PSUBC Sx, Sy, Dz	111110***** 10100000xxyyzzzz	Sx-Sy-DC Dz ボロー DC	1	ボロー
PCMP Sx, Sy	111110***** 10000100xxyy0000	Sx-Sy DC を更新*	1	*
PABS Sx, Dz	111110***** 10001000xx0zzzz	If Sx<0, 0-Sx Dz If Sx>=0, Sx Dz	1	*
PABS Sy, Dz	111110***** 101010000yyzzzz	If Sy<0, 0-Sy Dz If Sy>=0, Sy Dz	1	*
PRND Sx, Dz	111110***** 10011000xx0zzzz	Sx+h'00008000 Dz h'0000 Dz の LSW	1	*
PRND Sy, Dz	111110***** 101110000yyzzzz	Sy+h'00008000 Dz h'0000 Dz の LSW	1	*

【注】 * 表 3.19 を参照。

命令コード欄の上段中*****が A フィールド、下段が B フィールドを示します。

3.6.5 DSP モードでのオペレーションコードマップ

DSP モードで拡張される命令コードを含んだオペレーションコードマップを表 3.40 に示します。

表 3.40 オペレーションコードマップ

命令コード				Fx: 0000	Fx: 0001	Fx: 0010	Fx: 0011 ~ 1111
MSB		LSB		MD: 00	MD: 01	MD: 10	MD: 11
0000	Rn	Fx	0000				
0000	Rn	Fx	0001				
0000	Rn	00MD	0010	STC SR, Rn	STC GBR, Rn	STC VBR, Rn	STC SSR, Rn
0000	Rn	01MD	0010	STC SPC, Rn	STC MOD, Rn	STC RS, Rn	STC RE, Rn
0000	Rn	10MD	0010	STC R0_BANK, Rn	STC R1_BANK, Rn	STC R2_BANK, Rn	STC R3_BANK, Rn
0000	Rn	11MD	0010	STC R4_BANK, Rn	STC R5_BANK, Rn	STC R6_BANK, Rn	STC R7_BANK, Rn
0000	Rm	00MD	0011	BSRF Rm		BRAF Rm	
0000	Rm	10MD	0011	PREF @Rm			
0000	Rn	Rm	01MD	MOV.B Rm, @(R0, Rn)	MOV.W Rm, @(R0, Rn)	MOV.L Rm, @(R0, Rn)	MUL.L Rm, Rn
0000	0000	00MD	1000	CLRT	SETT	CLRMAC	
0000	0000	01MD	1000	CLRS	SETS		
0000	0000	10MD	1000				
0000	0000	11MD	1000				
0000	0000	Fx	1001	NOP	DIV0U		
0000	0000	Fx	1010				
0000	0000	Fx	1011	RTS	SLEEP	RTE	
0000	Rn	Fx	1000				
0000	Rn	Fx	1001			MOVT Rn	
0000	Rn	00MD	1010	STS MACH, Rn	STS MACL, Rn	STS PR, Rn	
0000	Rn	01MD	1010			STS DSR, Rn	STS A0, Rn
0000	Rn	10MD	1010	STS X0, Rn	STS X1, Rn	STS Y0, Rn	STS Y1, Rn
0000	Rn	Fx	1011				
0000	Rn	Rm	11MD	MOV.B @(R0, Rm), Rn	MOV.W @(R0, Rm), Rn	MOV.L @(R0, Rm), Rn	MAC.L @Rm + ,@Rn +
0001	Rn	Rm	disp	MOV.L Rm, @(disp:4, Rn)			
0010	Rn	Rm	00MD	MOV.B Rm, @Rn	MOV.W Rm, @Rn	MOV.L Rm, @Rn	
0010	Rn	Rm	01MD	MOV.B Rm, @ Rn	MOV.W Rm, @ Rn	MOV.L Rm, @ Rn	DIV0S Rm, Rn
0010	Rn	Rm	10MD	TST Rm, Rn	AND Rm, Rn	XOR Rm, Rn	OR Rm, Rn
0010	Rn	Rm	11MD	CMP/STR Rm, Rn	XTRCT Rm, Rn	MULU.W Rm, Rn	MULSW Rm, Rn
0011	Rn	Rm	00MD	CMP/EQ Rm, Rn		CMP/HS Rm, Rn	CMP/GE Rm, Rn
0011	Rn	Rm	01MD	DIV1 Rm, Rn	DMULU.L Rm, Rn	CMP/HI Rm, Rn	CMP/GT Rm, Rn
0011	Rn	Rm	10MD	SUB Rm, Rn		SUBC Rm, Rn	SUBV Rm, Rn
0011	Rn	Rm	11MD	ADD Rm, Rn	DMULS.L Rm, Rn	ADDC Rm, Rn	ADDV Rm, Rn
0100	Rn	Fx	0000	SHLL Rn	DT Rn	SHAL Rn	

3. DSP 演算ユニット

命令コード				Fx: 0000	Fx: 0001	Fx: 0010	Fx: 0011 ~ 1111	
MSB		LSB		MD: 00	MD: 01	MD: 10	MD: 11	
0100	Rn	Fx	0001	SHLR Rn	CMP/PZ Rn	SHAR Rn		
0100	Rn	Fx	0010	STS.L MACH, @ Rn	STS.L MACL, @ Rn	STS.L PR, @ Rn		
0100	Rn	00MD	0011	STC.L SR, @ Rn	STC.L GBR, @ Rn	STC.L VBR, @ Rn	STC.L SSR, @ Rn	
0100	Rn	01MD	0011	STC.L SPC, @ Rn	STC.L MOD, @ Rn	STC.L RS, @ Rn	STC.L RE, @ Rn	
0100	Rn	10MD	0011	STC.L R0_BANK, @ Rn	STC.L R1_BANK, @ Rn	STC.L R2_BANK, @ Rn	STC.L R3_BANK, @ Rn	
0100	Rn	11MD	0011	STC.L R4_BANK, @ Rn	STC.L R5_BANK, @ Rn	STC.L R6_BANK, @ Rn	STC.L R7_BANK, @ Rn	
0100	Rn	Fx	0100	ROTL Rn	SETRC Rn	ROTCL Rn		
0100	Rn	Fx	0101	ROTR Rn	CMP/PL Rn	ROTCR Rn		
0100	Rm	00MD	0110	LDS.L @Rm+, MACH	LDS.L @Rm+, MACL	LDS.L @Rm+, PR		
0100	Rm	01MD	0110			LDS.L @Rm+, DSR	LDS.L @Rm+, A0	
0100	Rm	10MD	0110	LDS.L @Rm+, X0	LDS.L @Rm+, X1	LDS.L @Rm+, Y0	LDS.L @Rm+, Y1	
0100	Rm	00MD	0111	LDC.L @Rm+, SR	LDC.L @Rm+, GBR	LDC.L @Rm+, VBR	LDC.L @Rm+, SSR	
0100	Rm	01MD	0111	LDC.L @Rm+, SPC	LDC.L @Rm+, MOD	LDC.L @Rm+, RS	LDC.L @Rm+, RE	
0100	Rm	10MD	0111	LDC.L @Rm+, R0_BANK	LDC.L @Rm+, R1_BANK	LDC.L @Rm+, R2_BANK	LDC.L @Rm+, R3_BANK	
0100	Rm	11MD	0111	LDC.L @Rm+, R4_BANK	LDC.L @Rm+, R5_BANK	LDC.L @Rm+, R6_BANK	LDC.L @Rm+, R7_BANK	
0100	Rn	Fx	1000	SHLL2 Rn	SHLL8 Rn	SHLL16 Rn		
0100	Rn	Fx	1001	SHLR2 Rn	SHLR8 Rn	SHLR16 Rn		
0100	Rm	00MD	1010	LDS Rm, MACH	LDS Rm, MACL	LDS Rm, PR		
0100	Rm	01MD	1010			LDS Rm, DSR	LDS Rm, A0	
0100	Rm	10MD	1010	LDS Rm, X0	LDS Rm, X1	LDS Rm, Y0	LDS Rm, Y1	
0100	Rm/Rn	Fx	1011	JSR @Rm	TAS.B @Rn	JMP @Rm		
0100	Rn	Rm	1100	SHAD Rm, Rn				
0100	Rn	Rm	1101	SHLD Rm, Rn				
0100	Rm	00MD	1110	LDC Rm, SR	LDC Rm, GBR	LDC Rm, VBR	LDC Rm, SSR	
0100	Rm	01MD	1110	LDC Rm, SPC	LDC Rm, MOD	LDC Rm, RS	LDC Rm, RE	
0100	Rm	10MD	1110	LDC Rm, R0_BANK	LDC Rm, R1_BANK	LDC Rm, R2_BANK	LDC Rm, R3_BANK	
0100	Rm	11MD	1110	LDC Rm, R4_BANK	LDC Rm, R5_BANK	LDC Rm, R6_BANK	LDC Rm, R7_BANK	
0100	Rn	Rm	1111	MAC.W @Rm+, @Rn+				
0101	Rn	Rm	disp	MOV.L @(disp:4, Rm), Rn				
0110	Rn	Rm	00MD	MOV.B @Rm, Rn	MOV.W @Rm, Rn	MOV.L @Rm, Rn	MOV Rm, Rn	
0110	Rn	Rm	01MD	MOV.B @Rm+, Rn	MOV.W @Rm+, Rn	MOV.L @Rm+, Rn	NOT Rm, Rn	

3. DSP 演算ユニット

命令コード				Fx: 0000	Fx: 0001	Fx: 0010	Fx: 0011 ~ 1111
MSB		LSB		MD: 00	MD: 01	MD: 10	MD: 11
0110	Rn	Rm	10MD	SWAP.B Rm, Rn	SWAP.W Rm, Rn	NEGC Rm, Rn	NEG Rm, Rn
0110	Rn	Rm	11MD	EXTU.B Rm, Rn	EXTU.W Rm, Rn	EXTS.B Rm, Rn	EXTS.W Rm, Rn
0111	Rn		imm	ADD #imm : 8, Rn			
1000	00MD	Rn	disp	MOV.B	MOV.W	SETRC #imm	
		imm		R0, @(disp: 4, Rn)	R0, @(disp: 4, Rn)		
1000	01MD	Rm	disp	MOV.B @(disp:4, Rm), R0	MOV.W @(disp: 4, Rm), R0		
1000	10MD		imm/disp	CMP/EQ #imm:8, R0	BT disp: 8		BF disp: 8
1000	11MD		imm/disp	LDRS @(disp:8,PC)	BT/S disp: 8	LDRE @(disp:8,PC)	BF/S disp: 8
1001	Rn		disp	MOV.W @(disp : 8, PC), Rn			
1010			disp	BRA disp : 12			
1011			disp	BSR disp: 12			
1100	00MD		imm/disp	MOV.B R0, @(disp: 8, GBR)	MOV.W R0, @(disp: 8, GBR)	MOV.L R0, @(disp: 8, GBR)	TRAPA #imm: 8
1100	01MD		disp	MOV.B @(disp: 8, GBR), R0	MOV.W @(disp: 8, GBR), R0	MOV.L @(disp: 8, GBR), R0	MOVA @(disp: 8, PC), R0
1100	10MD		imm	TST #imm: 8, R0	AND #imm: 8, R0	XOR #imm: 8, R0	OR #imm: 8, R0
1100	11MD		imm	TST.B #imm: 8, @(R0, GBR)	AND.B #imm: 8, @(R0, GBR)	XOR.B #imm: 8, @(R0, GBR)	OR.B #imm: 8, @(R0, GBR)
1101	Rn		disp	MOV.L @(disp: 8, PC), Rn			
1110	Rn		imm	MOV #imm:8, Rn			
1111	00**		*****	MOVX.W, MOVY.W デブルデータ転送命令			
1111	01**		*****	MOVS.W, MOVS.L シングルデータ転送命令			
1111	10**		*****	MOVX.W, MOVY.W デブルデータ転送命令; DSP 並列演算命令付き(32ビット命令)			
1111	11**		*****				

- 【注】
1. 詳細については、「SH-3/SH-3E/SH3-DSP プログラミングマニュアル」を参照してください。
 2. 網掛けの命令コードは、DSP 拡張命令です。SR.DSP=1 の場合のみ使用可能です。

3. DSP 演算ユニット

4. 例外処理

例外処理とは、通常のプログラムの処理から離れて、通常とは異なる処理をすることをいいます。たとえば、定義されていない命令コードを実行しようとしたり、CPU の処理モードによって保護されている命令を実行しようとした場合に、適切な処置をすることで、元のプログラムに復帰したり、異常を報告して終了するなどの制御が必要になります。また、LSI に内蔵されたモジュールや LSI 外部に接続されたモジュールから CPU に対して要求された処理の制御が必要になる場合もあります。

このような機能をサポートするために、ユーザが作成した例外処理ルーチンに制御を移し、その処置をすることを例外処理と呼びます。本 LSI では、例外処理の要求を一般例外および割り込みの 2 種類に分類して処理します。ユーザは、例外処理の要求に応じた例外処理ルーチンを配置することで、必要な処理を実行した後、元のプログラムの実行を再開することができます。

リセットの入力も、通常のプログラムの実行を中断して、レジスタの初期化を行った後にリセットベクタへ制御の流れを移します。この動作も例外処理の一つとみなすことができます。本章では、その動作を概説します。ただし、本章での「例外処理」とは、一般例外および割り込みに対する処理を表します。

また、本章では、割り込みに関しては割り込み要求に対する処理のみを記述しています。割り込み要求の発生方法に関しては、「第 8 章 割り込みコントローラ (INTC)」を参照してください。

4.1 レジスタの説明

例外処理で使用するレジスタには、以下の 4 つがあります。初期値が不定のレジスタは、ソフトウェアにより初期化してください。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については、「第 27 章 レジスタ一覧」を参照してください。

- TRAPA例外レジスタ (TRA)
- 例外事象レジスタ (EXPEVT)
- 割り込み事象レジスタ 2 (INTEVT2)
- 例外アドレスレジスタ (TEA)

各レジスタのビット構成を図 4.1 に示します。

4. 例外処理

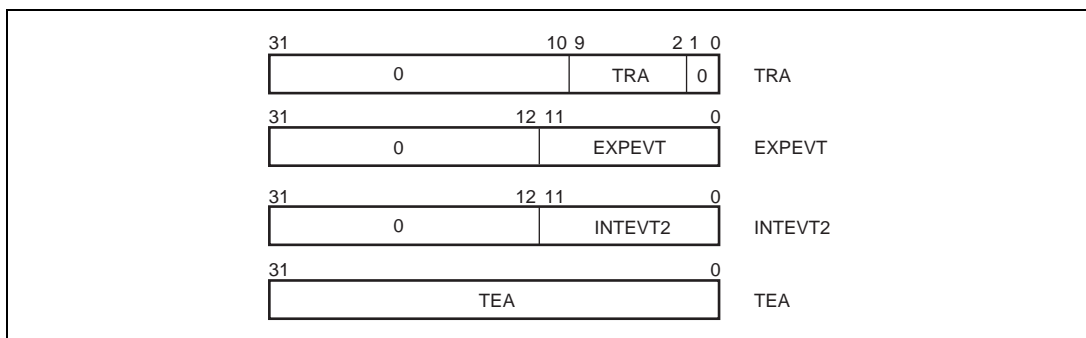


図 4.1 レジスタのビット構成

4.1.1 TRAPA 例外レジスタ (TRA)

TRA は、H'FFFF FFD0 番地に配置されていて、TRAPA 命令の 8 ビットイミディエイトデータ (imm) から構成されています。TRA は、TRAPA 命令実行時にハードウェアにより自動的に設定されます。TRA は、ソフトウェアからも変更が可能ですが、ビット 9~2 のみが書き換えできます。

ビット	ビット名	初期値	R/W	説明
31~10		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
9~2	TRA		R/W	8 ビットイミディエイトデータ
1、0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

4.1.2 例外事象レジスタ (EXPEVT)

EXPEVT は、H'FFFF FFD4 番地に配置されていて、例外コード 12 ビットから構成されています。EXPEVT に設置される例外コードは、リセットと一般例外事象による例外コードです。例外コードは、例外発生時にハードウェアにより自動的に設定されます。EXPEVT は、ソフトウェアからも変更が可能ですが、ビット 11~ビット 0 のみが書き換えできます。

ビット	ビット名	初期値	R/W	説明
31~12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
11~0	EXPEVT	*	R/W	12 ビットの例外コード

【注】 * パワーオンリセット時には H'000 が、マニュアルリセット時には H'020 が設定されます。

4.1.3 割り込み事象レジスタ 2 (INTEVT2)

INTEVT2 は、H'A4000000 番地に配置されていて、例外コード 12 ビットから構成されています。INTEVT2 に設定される例外コードは、割り込み要求による例外コードです。例外コードは、例外発生時にハードウェアにより自動的に設定されます。INTEVT2 は、ソフトウェアからは変更できません。

ビット	ビット名	初期値	R/W	説 明
31~12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
11~0	INTEVT2		R	12 ビットの割り込み要求による例外コード

4.1.4 例外アドレスレジスタ (TEA)

TEA は、H'FFFFFFFC 番地に配置されていて、メモリアクセスに関連した例外が発生した際に、例外を発生させた論理アドレスが格納されます。TEA は、ソフトウェアからも変更できます。

ビット	ビット名	初期値	R/W	説 明
31~0	TEA	0	R/W	例外を発生させた論理アドレス

4.2 例外処理の機能

4.2.1 例外処理の流れ

例外処理では、例外処理の後に復帰すべきアドレスおよびステータスレジスタ (SR) の内容をそれぞれ退避プログラムカウンタ (SPC) および退避ステータスレジスタ (SSR) に退避し、ベクタアドレスに従って対応する例外処理ルーチンに制御を渡し、実行します。例外発生時の状態に復帰する場合は、例外処理ルーチンで復帰命令 (RTE) を実行します。これにより SSR の内容が SR に回復され、例外発生時の状態に戻り、その後 SPC に退避されたアドレスに制御を移します。基本的な例外処理の流れは、次のとおりです。

例外要求が発生し、CPU がそれを受け付けると、次の 1~8 が実行されます。

1. 例外処理の後に、復帰すべき命令のアドレスが SPC に退避されます。
2. SR の内容が、SSR に退避されます。
3. SR のブロックビット (BL) が 1 に設定され、後続の例外要求がマスクされます。
4. SR のモードビット (MD) が 1 に設定され、特権モードに切り換わります。
5. SR のレジスタバンクビット (RB) が 1 に設定されます。
6. 例外要因が一般例外の場合は、要因の例外コードが EXPEVT に、例外要因が割り込みの場合は要因の例外コードが INTEVT2 に書き込まれます。
7. TRAPA 命令を実行した場合は、TRA に TRAPA 命令で指定した 8 ビットイミディエイト値が設定されます。ま

4. 例外処理

た、メモリアクセスに関する例外の場合は、TEAに例外を発生した論理アドレスが書き込まれます。

8. 決められた例外処理のベクタアドレスに分岐して、例外処理ルーチンの命令が実行されます。

1~8は連続して行われ、後述の多重例外受け付けの場合を除き、この期間で別の例外を受け付けることはありません。

一般例外の例外処理ルーチンでは EXPEVT の値から、割り込み処理の例外処理ルーチンでは INTEVT2 の値から例外要因を判定し、必要な例外処理を行ってください。例外処理ルーチンの処理が終了すると、RTE 命令を実行することでプログラムの実行を再開できます。RTE 命令では、次の 1~3 が実行されます。

1. SSRに退避された内容をSRに回復し、例外処理前の処理状態に戻ります。
2. RTE命令の遅延スロットの命令を実行します。*
3. SPCに退避されたアドレスへ制御を移します。

1~3は連続して行われ、この期間で別の例外を受け付けることはありません。また、RTE 命令の実行前に SPC や SSR を操作することで、例外処理前とは異なった状態に復帰することもできます。

【注】 * RTE遅延スロット命令が実行されるCPU処理モードに関しては、「4.5 使用上の注意事項」を参照してください。

4.2.2 例外処理ベクタアドレス

一般例外の割り込みのベクタアドレスは、ベクタベースアドレスにベクタオフセットの値を加えた値で決めます。一般例外のベクタオフセットは、H'0000 0100 です。割り込みのベクタアドレスのオフセットは、H'0000 0600 です。ベクタベースアドレスは、ベクタベースレジスタ (VBR) にソフトウェアで設定します。ベクタベースアドレスは、論理アドレスエリア (P1 または P2 領域) に設定してください。

4.2.3 例外コード

各例外事象を区別するために例外コードが決められており、リセットと一般例外のときには EXPEVT のビット 11~0 に、割り込み要求のときには INTEVT2 に書き込まれます。割り込みの例外コード(要因コードの)詳細は、「第 8 章 割り込みコントローラ (INTC)」を参照してください。リセットおよび一般例外に対する例外コードを表 4.1 に示します。

4.2.4 例外要求と BL ビットの関係 (多重例外防止)

SR の BL ビットは、リセットや例外の受け付けで 1 に設定されます。BL ビットが 1 のときは、一般例外の割り込みの受け付けが次のように制限されて、多重に例外が受け付けられるのを防止します。

割り込みの要求は、BL ビットが 1 の間は保留されます。ユーザが BL ビットを 0 にした時点で割り込み要求が受け付けられます。ただし、CPU が低消費電力状態にある場合は、SR の BL ビットが 1 であっても、割り込みを受け付け、低消費電力状態から復帰します。

DMA アドレスエラーも同様に BL ビットが 1 の間は保留され、BL ビットが 0 になった時点で例外要求が受け付けられます。一方、BL ビットが 1 の期間中に生じたユーザブレイク要求は無視され、保留されません。したがって、BL ビットを 0 にしても、要求された例外は、受け付けられません。

BLビットが1の期間にDMAアドレスエラーおよびユーザブ레이크以外の一般例外要求が生じた場合には、CPUはリセット後の状態と同じになり、リセットベクタ（H'A0000000）へ処理を移行します（多重例外）。ただし、これは通常のリセットとは異なり、CPU以外のモジュールは初期化されず、EXPEVT、SPC、およびSSRの値は、不定となります。また、LSI外部でこの状態を検出することもできません。

例外処理を多重に受け付け可能にするためには、例外受け付け後にBLビットが1の間にSPCとSSRを退避させ、その後SRのBLビットを0でクリアします。SPCとSSRを回復する場合には、これらを回復する前にSRのBLビットを1に設定してください。

4.2.5 例外要因の受け付けタイミングと優先順位

（1）命令同期型・命令非同期型例外要求

リセットおよび割り込みは、プログラムの流れに関係無く非同期に入る例外要求です。一般例外の内、DMAアドレスエラーと或る条件下でのユーザブ레이크も非同期の例外要求となります。これらの例外要求は、その要求をどの命令で受け付けるかを予測することはできません。

その他の一般例外は、その例外要求に対応する命令が一意に決定します。

（2）再実行型、完了型例外

すべての例外は、再実行型および完了型の2種類に分類されます。再実行型の例外を受け付けると、例外を受け付けた命令の実行を中断し、そのアドレスをSPCに退避します。例外処理から復帰すると、例外を生じた命令から実行を再開します。完了型例外では、例外を受け付けた命令の実行を完了し、次の命令のアドレスを退避して例外処理を実行します。

ただし、遅延分岐命令と遅延スロットの間では、次のように処理されます。遅延スロットで検出された再実行型例外は、遅延分岐命令実行前に受け付けられます。遅延分岐命令または遅延スロットで検出された完了型例外は、遅延分岐命令実行後、分岐先命令の実行前に受け付けられます。ここでいう遅延スロットとは、遅延無条件分岐命令の次の命令や、遅延条件分岐命令が成立するときの次の命令を示します。遅延条件分岐で分岐しなかった場合は、通常と同じように処理されます。

（3）優先順位と判定順位

すべての例外要求は、2つ以上の例外が同時に発生したときに受け付ける優先順位が決められています。リセット、一般例外、および割り込みの例外要求は、ここで挙げられた順位で優先順位が高くなっています。リセットは、CPUがいかなる状態であろうと受け付けられます。また、割り込みは、リセットや一般例外の要求が無い場合に限り受け付けられます。

同一の命令で複数の一般例外要求が生じる場合は、次の順序で判定されます。

1. 直前の命令で生じた完了型例外*
2. 命令実行前ユーザブ레이크 【再実行型】
3. 命令フェッチに関する例外（CPUアドレスエラー）【再実行型】

4. 例外処理

4. 命令デコードの結果生じる例外（一般不当命令例外、スロット不当命令例外【再実行型】、および無条件トラップ【完了型】）
5. データアクセスに関する例外（CPUアドレスエラー）【再実行型】
6. 無条件トラップ【完了型】
7. 命令実行前以外のユーザブレイク【完了型】
8. DMAアドレスエラー【完了型】

【注】 * ある命令で完了型例外を受けつけた場合は、次の命令の実行前に例外処理に移行しますが、この処理は次の命令で生じる例外の判定を行う前に優先して処理されます。

一度に受け付けることの出来る例外要求は1個ですが、順次、例外要求を受け付けていくことですべての例外要求を処理することができます。

表 4.1 例外事象一覧

例外種別	実行命令の終了状態	例外事象	優先順位 *1	判定順位	BL=1 ときの 処理	例外要因 コード*5	ベクタ オフセット
リセット (命令非同期)	中断	パワーオンリセット	1	1	リセット	H'000	-
		マニュアルリセット	1	2	リセット	H'020	-
一般例外 (命令同期)	再実行	ユーザブレイク（命令実行前）	2	0	無視	H'1E0	H'0000100
		CPU アドレスエラー（命令アクセス）*4	2	1	リセット	H'0E0	H'0000100
		一般不当命令例外	2	2	リセット	H'180	H'0000100
		スロット不当命令例外	2	2	リセット	H'1A0	H'0000100
		CPU アドレスエラー（データ読み出し/書き込み）*4	2	3	リセット	H'0E0/H'100	H'0000100
	完了	無条件トラップ（TRAPA 命令）	2	4	リセット	H'160	H'0000100
		ユーザブレイク（命令実行後、アドレス）	2	5	無視	H'1E0	H'0000100
一般例外 (命令非同期)	完了	ユーザブレイク（データブレイク、Iバスブレイク）	2	5	無視	H'1E0	H'0000100
		DMA アドレスエラー	2	6	保留	H'5C0	H'0000100
割り込み (命令非同期)	完了	ノンマスカブル割り込み（NMI）	3*2	-	保留	- *3	H'0000600
		外部ハードウェア割り込み（IRQ 割り込み）	4*2	-	保留	- *3	H'0000600
		H-UDI 割り込み	4*2	-	保留	- *3	H'0000600
		内蔵周辺モジュール割り込み	4*2	-	保留	- *3	H'0000600

【注】 *1 優先順位は、高い方から低い方に1から4で指定されます。リセットは、すべての要求に優先します。割り込みは、一般例外要求が無い場合のみ受け付けることができます。

*2 複数の割り込み要因間の優先順位は、「第8章 割り込みコントローラ（INTC）」を参照してください。

*3 割り込みを受理しても例外事象レジスタ（EXPEVT）の値は変化せず、割り込み事象レジスタ2（INTEVT2）に割り込み要求元を示す割り込み要因の例外コードが設定されます。「第8章 割り込みコントローラ（INTC）」を参照してください。

- *4 リピートループ中の特定の箇所でこれらの例外が発生すると特殊な例外コードおよび特殊なベクタオフセットになる場合があります。
- *5 H'040、H'060、H'080、H'0A0、H'0C0、H'0D0、H'120、H'140、H'3E0の例外要因コードは、予約されています。

4.3 個別例外の動作説明

個別の例外処理動作について、発生条件および発生時のプロセッサの動作を説明します。本節では、リセットおよび一般例外について述べます。割り込みの動作に関しては、「第8章 割り込みコントローラ (INTC)」を参照してください。

4.3.1 リセット

(1) パワーオンリセット

条件：

$\overline{\text{RESETP}}$ 端子*1 ローレベル、WDTによるパワーオンリセット要求

動作：

EXPEVTにH'000を設定し、VBR、SRの初期化を行い、PC = H'A0000000に分岐します*2。

初期化により、VBRレジスタはH'00000000にセットされます。SRレジスタは、MD、RB、BLビットが1にセットされ、DSPビットが0にクリアされ、割り込みマスクビット (I3~I0) がB'1111にセットされます。

CPUおよび内蔵周辺モジュールの初期化を行います。詳細は、各章のレジスタの説明を参照してください。電源投入時には必ず $\overline{\text{RESETP}}$ 端子ローレベルによるリセットを行ってください。

【注】 *1 $\overline{\text{RESETP}}$ 端子を用いたリセットについては、注意事項があります。詳細については「11.7.1 $\overline{\text{RESETP}}$ 端子を用いたリセットについて」を参照してください。

*2 パワーオンリセット時、BOOT_E 端子がローアサートされていると、アドレスはH'A000 0000ですが、ブート機能にあるROM領域に分岐し、ブート処理が始まります（「第17章 ブート機能 (BOOT)」を参照）。

(2) マニュアルリセット

条件：

WDTによるマニュアルリセット要求

動作：

EXPEVTにH'020を設定し、VBR、SRの初期化を行い、PC = H'A0000000に分岐します。*

初期化により、VBRレジスタはH'00000000にセットされます。SRレジスタは、MD、RB、BLビットが1にセットされ、DSPビットが0にクリアされ、割り込みマスクビット (I3~I0) がB'1111にセットされます。

CPUおよび内蔵周辺モジュールの初期化を行います。パワーオンリセットとマニュアルリセットでは、初期化されるレジスタが異なります。詳細は、各章のレジスタの説明を参照してください。

【注】 * マニュアルリセット時もパワーオンリセット時と同様、 $\overline{\text{BOOT_E}}$ 端子がローアサートされていると、ブート機能にあるROM領域に分岐し、ブート処理が始まります。

4. 例外処理

4.3.2 一般例外

(1) CPU アドレスエラー

条件：

- 奇数アドレス ($4n+1$ 、 $4n+3$) から命令フェッチ
- ワードデータをワード境界以外 ($4n+1$ 、 $4n+3$) からアクセス
- ロングワードデータをロングワードデータ境界以外 ($4n+1$ 、 $4n+2$ 、 $4n+3$) からアクセス
- ユーザモードで論理空間のH'80000000 ~ H'FFFFFFFの領域をアクセス

種別：

命令同期、再実行型

退避アドレス：

命令フェッチの場合： 本例外を発生させた命令フェッチ先のアドレス。

データアクセスの場合： 本例外を発生させた命令のアドレス。

ただし、遅延スロットにある場合には、遅延分岐命令のアドレス。

例外コード：

読み出しで例外が発生した場合： H'0E0

書き込みで発生した場合： H'100

特記：

本例外を発生させた論理アドレス (32ビット) をTEAに設定します。

(2) 一般不当命令例外

条件：

- 遅延分岐命令の遅延スロット以外で、命令の定義されていない命令コード (未定義コード) を実行した場合
遅延分岐命令：JMP、JSR、BRA、BRA_F、BSR、BSR_F、RTS、RTE、BT/S、BF/S

【注】 未定義コードについては、「第2章 CPU」の「表 2.11 オペレーションコードマップ」を参照してください。ただし、命令コード H'FC00 ~ H'FFFF の未定義コードの例外処理動作のみを保証し、その他の未定義コードでは動作を保証しません。

- 遅延スロット以外にある特権命令をユーザモードでデコードした場合

特権命令：LDC、STC、RTE、SLEEP

ただし、LDC/STCでGBRをアクセスする命令は特権命令ではありません。

種別：

命令同期、再実行型

退避アドレス：

本例外を発生させた命令のアドレス

例外コード：

H'180

特記：

なし

(3) スロット不当命令例外

条件：

- 遅延スロットにある未定義コードをデコードした場合

遅延分岐命令：JMP、JSR、BRA、BRAf、BSR、BSRf、RTS、RTE、BT/S、BF/S

- 遅延スロットにある特権命令をユーザモードでデコードした場合

特権命令：LDC、STC、RTE、SLEEP

ただし、LDC/STCでGBRにアクセスする命令は、特権命令ではありません。

- 遅延スロット内でPCを書き換える命令をデコードした場合

PCを書き換える命令：JMP、JSR、BRA、BRAf、BSR、BSRf、RTS、RTE、BT、BF、BT/S、BF/S、TRAPA、
LDC Rm,SR、LDC.L @Rm+,SR

種別：

命令同期、再実行型

退避アドレス：

遅延分岐命令が配置されたアドレス

例外コード：

H'1A0

特記：

なし

(4) 無条件トラップ

条件：

TRAPA命令の実行

種別：

命令同期、完了型

退避アドレス：

TRAPA命令の次命令のアドレス

例外コード：

H'160

4. 例外処理

特記：

完了型の例外のため、TRAPA命令の次命令のアドレスをSPCに退避します。TRAPA命令中の8ビットのイミディエイト値をTRA[9:2]に設定します。

(5) ユーザブレイクポイントトラップ

条件：

ユーザブレイクコントローラに設定したブレイク条件が成立した場合

種別：

命令実行前ブレイク (Lバス) の場合： 命令同期、再実行型。

オペランドブレイク (Lバス) の場合： 命令同期、完了型。

データブレイク (Lバス) の場合： 命令非同期、完了型。

Iバスでのブレイクの場合： 命令非同期、完了型。

退避アドレス：

再実行型の場合： ブレイク成立命令のアドレス。

ただし、遅延スロットにある場合には遅延分岐命令のアドレス。

完了型の場合： ブレイク要求を受け付けた命令の次命令のアドレス。

ただし、遅延スロットにある場合には、遅延分岐命令の分岐先のアドレス。

例外コード：

H'1E0

特記：

ユーザブレイクコントローラの詳細は、「第25章 ユーザブレイクコントローラ (UBC)」を参照してください。

(6) DMA アドレスエラー

条件：

- ワードデータをワード境界以外 ($4n+1$ 、 $4n+3$) からアクセス
- ロングワードデータをロングワードデータ境界以外 ($4n+1$ 、 $4n+2$ 、 $4n+3$) からアクセス

種別：

命令非同期、完了型

退避アドレス：

本例外を受け付けた命令の次命令のアドレスが退避されます。ただし、遅延スロットにある場合には、遅延分岐命令の分岐先のアドレスが退避されます。

例外コード：

H'5C0

特記：

ダイレクトメモリアクセスコントローラ (DMAC) に上記条件に該当する不正なアドレスが設定され、実際の転送動作が行われた際に例外要求が生じます。DMACの転送動作は、CPUの命令動作とは非同期に行われるため、例外要求も命令非同期になります。DMACの動作の詳細は、「第10章 ダイレクトメモリアクセスコントローラ (DMAC)」を参照してください。

【注】 USBホストはハードウェアがアドレスを自動的に認識して適当なビット長のデータアクセスを実行するので、アドレスエラーは発生しません。

4.4 DSP 拡張機能有効時の例外処理

DSPモード (SRレジスタのDSPビットが1) のとき、例外処理の受け付け条件や例外処理が変更されるものがあります。

4.4.1 不当命令例外、スロット不当命令例外

DSPモードでは、DSP拡張命令を実行することが可能になります。逆に、SRレジスタのDSPビットが0のときにこれらの命令コードを実行しようとすると、不当命令例外になります。

また、DSPモードでは、SRレジスタに対するSTCおよびLDC命令がユーザモードでも実行可能になります (ただし、更新可能なビットはDSP拡張ビットの内、RC[11:0]、DMX、DMY、およびRF[1:0]のみです)。

4.4.2 CPU アドレスエラー

DSPモードでは、P2空間の一部 (U_{xy}領域: H'A5000000 ~ H'A5FFFFFF) がユーザモードでもアクセスでき、CPUアドレスエラーが発生しません。

4.4.3 リピート制御中の例外

リピート制御中に発生する例外要求に対する扱いや例外を受け付けた際の処理は、通常の状態とは異なり、場合によっては例外を受け付けなかったり、受け付けても正しく復帰できなかったりする場合があります。この制約は、リピートカウンタが1以上で、リピート検出命令からリピート最終命令までの間の命令で生じる場合があります。本節では、この期間をリピート制御中と呼びます。以下に、リピートループ中の命令数が4以上、3、2、および1の場合のプログラム例を示します。本節では、リピート検出命令やその命令アドレスをRptDtctと記述します。同様に、リピート検出命令の1,2,および3命令先の命令やその命令アドレスをそれぞれRptDtct1, RptDtct2, およびRptDtct3と記述します。

また、下記例中の【A】、【B】、【C1】、および【C2】は、各種の制約が生じる命令を表します。「表 4.2 リピートループ中の命令位置と制約の種別」にその概要を示します。

4. 例外処理

表 4.2 リピートループ中の命令位置と制約の種類

命令位置	SPC* ¹	不当命令* ²	割り込み、ブレーク等* ³	CPU アドレスエラー* ⁴
【A】				
【B】			保留	
【C1】		追加	保留	命令 / データ
【C2】	不正	追加	保留	命令 / データ

【注】 *1 SR.RC[11:0]>=2 の場合に例外が発生すると、SPC に特殊なアドレスが設定されます。

*2 SR.RC[11:0]>=1 の場合に不当命令になり得る命令種が増えます。

*3 SR.RC[11:0]>=1 の場合に、割り込みやブレーク、DMA アドレスエラーの要求が保留されます。

*4 SR.RC[11:0]>=1 の場合に、特殊な例外コードが設定されます。

• (例1) 4命令以上のリピートループ

```

LDRS RptStart      ; 【A】
LDRE RptDtct+4    ; 【A】
SETRC #4          ; 【A】
instr0            ; 【A】
RptStart: instr1   ; 【A】 【リピート開始命令】
                  ; 【A】
                  ; 【A】
RptDtct: RptDtct  ; 【B】 【リピート検出命令】 リピート最終命令から 3 命令前の命令
                  ; 【B】 がリピート検出命令になります。
RptDtct1          ; 【C1】
RptDtct2          ; 【C2】
RptEnd: RptDtct3  ; 【C2】 【リピート最終命令】
instrNext        ; 【A】

```

• (例2) 3命令リピートループ

```

LDRS RptDtct+4    ; 【A】
LDRE RptDtct+4    ; 【A】
SETRC #4          ; 【A】
RptDtct: RptDtct  ; 【B】 【リピート検出命令】 リピート開始命令の直前の命令がリピー
                  ; 【B】 ト検出命令になります。
RptStart: RptDtct1 ; 【C1】 【リピート開始命令】
RptDtct2          ; 【C2】
RptEnd: RptDtct3  ; 【C2】 【リピート最終命令】
instrNext        ; 【A】

```

• (例3) 2命令リピートループ

```

LDRS RptDtct+6 ; 【A】
LDRE RptDtct+4 ; 【A】
SETRC #4 ; 【A】
RptDtct: RptDtct ; 【B】【リピート検出命令】リピート開始命令の直前の命令がリピート検出命令になります。
RptStart: RptDtct1 ; 【C1】【リピート開始命令】
RptEnd: RptDtct2 ; 【C2】【リピート最終命令】
instrNext ; 【A】

```

• (例4) 1命令リピートループ

```

LDRS RptDtct+8 ; 【A】
LDRE RptDtct+4 ; 【A】
SETRC #4 ; 【A】
RptDtct: RptDtct ; 【B】【リピート検出命令】リピート開始命令の直前の命令がリピート検出命令になります。
RptStart:
RptEnd: RptDtct1 ; 【C1】【リピート開始命令】==【リピート最終命令】
instrNext ; 【A】

```

(1) リピート制御中の例外で退避される SPC

リピート制御中に例外を受け付け、SR レジスタのリピートカウンタ (RC[11:0]) が 2 以上である場合は、退避されるアドレスが復帰すべき値を正しく示さない場合があります。また、例外処理から復帰した際にリピート制御を正しく再実行するには、その復帰アドレスはリピート検出命令以前である必要があります。従って、リピート制御中に例外を受け付けた場合は、リピート検出命令での再実行型例外を除き、正しくリピート制御に復帰することはできません。

表 4.3 リピート制御中に生じた再実行型例外に対するの SPC の値 (SR.RC[11:0]>=2 の場合)

例外発生命令	リピートループ中の命令数			
	1	2	3	>=4
RptDtct	RptDtct	RptDtct	RptDtct	RptDtct
RptDtct1	RptDtct1	RptDtct1	RptDtct1	RptDtct1
RptDtct2	-	RptDtct1	RptDtct1	RS-4
RptDtct3	-	-	RptDtct1	RS-2

【注】 ここでは、次のラベルを使用しています。

RptDtct : リピート検出命令のアドレス

RptDtct1 : リピート検出命令の 1 命令後の命令アドレス (1,2,3 命令リピートでは RptStart です。)

RptDtct2 : リピート検出命令の 2 命令後の命令アドレス。

4. 例外処理

RptDct3: リピート検出命令の3命令後の命令アドレス。

RS: リピート開始命令のアドレス。

表中の網掛けの箇所再実行型の例外処理がなされた場合は、本来退避されるべきアドレスと異なったアドレスがSPCに退避されます。SR.RC[11:0]が1または0の場合は、正しいアドレスが退避されます。

(2) リピート制御中の不当命令例外

リピート制御中の RptDct1 以降の命令において下記の命令を使用すると一般不当命令例外を発生します。SPCに待避されるアドレスは「4.4.3 (1) リピート制御中の例外で退避される SPC」を参照してください。

- 分岐命令

BRA, BSR, BT, BF, BT/S, BF/S, BSRF, RTS, BRAF, RTE, JSR, JMP, TRAPA

- リピート制御命令

SETRC, LDRS, LDRE

- SR, RS, REに対するロード命令

LDC Rn,SR, LDC @Rn+,SR, LDC Rn,RE, LDC @Rn+,RE, LDC Rn,RS, LDC @Rn+,RS

【注】 1~3命令のリピートループの場合はリピートループ中の全命令が、4命令以上のリピートループの場合はリピート終了命令を含む3命令がこの制約の範囲となります。

(3) リピート制御中に保留される例外要求

リピート制御中、割り込みや一部の例外要求はその受け付けを保留し、例外処理から復帰できない命令で例外を受け付けることを防止しています。リピートループのプログラム(例1)~(例4)を参照してください。【B】および【C】(【C1】または【C2】)と記した命令で発生した例外は、それぞれ次のように処理されます。

- 割り込み、DMAアドレスエラー

【B】の命令および【C】の命令とも、例外を受け付けず、また、要求も保存されません。次に【A】の命令を実行したときに改めて要求を検出し、受け付けます*。(例1)~(例4)に示されるように、4命令以下のリピートループの実行中には、割り込みおよびDMAアドレスエラーは受け付けられません。

【注】 * CPUが受け付け可能な状態になるまで、割り込みコントローラ(INTC)やダイレクトメモリアクセスコントローラ(DMAC)で要求が保存されます。

- 実行前ユーザブ레이크

【B】の命令では受け付けられ、SPCに【B】の命令のアドレスがセットされます。【C】の命令では受け付けられませんが、要求は保存され、次に【A】の命令、あるいは【B】の命令が実行される際にその命令の実行前に受け付けられます。このときのSPCはその【A】あるいは【B】の命令のアドレスになります。

- 実行後ユーザブ레이크

【B】の命令、【C】の命令とも、例外は受け付けられませんが、要求は保存され、次に【A】の命令あるいは【B】の命令を実行する際に、その命令の実行前に受け付けられます。このときのSPCはその【A】あるいは【B】の命令のアドレスになります。

表 4.4 リピートループでの例外受け付けの制限

	B の命令	C の命令
割り込み	受け付けません	受け付けません
DMA アドレスエラー	受け付けません	受け付けません
実行前ユーザブレイク	受け付けます	受け付けません
実行後ユーザブレイク	受け付けません	受け付けません

(4) リピート制御中の CPU アドレスエラー

リピート制御中に CPU アドレスエラーが発生した場合は、例外を受け付けますが、リピートループ中であることを示す例外コード (H'070) が EXPEVT に設定されます。リピート検出命令の次命令からリピート最終命令においては命令アクセスおよびデータアクセスに対する例外がこの例外コードを生成します。

また、SPC に退避されるアドレスは「4.4.3 (1) リピート制御中の例外で退避される SPC」に準じて設定されます。

この例外が生じた場合は、例外処理の後リピート制御に正しく復帰することができません。リピートループを正しく実行するには、リピート制御中に CPU アドレスエラーが発生しないようにしてください。

【注】 1~3 命令のリピートループの場合はリピート検出命令およびリピートループ中の全命令が、4 命令以上のリピートループの場合はリピート終了命令を含む 3 命令がこの制約の範囲となります。SR.RC[11:0]>=1 の場合に、この制約が生じます。

表 4.5 リピート制御中のメモリアクセス例外で特殊な例外を生じる命令 (SR.RC[11:0]>=1)

例外発生命令	リピートループ中の命令数			
	1	2	3	>=4
RptDct				
RptDct1	命令 / データアクセス	命令 / データ	命令 / データ	命令 / データ
RptDct2	-	命令 / データ	命令 / データ	命令 / データ
RptDct3	-	-	命令 / データ	命令 / データ

【注】 ここでは、次のラベルを使用しています。

RptDct : リピート検出命令

RptDct1 : リピート検出命令の 1 命令後の命令

RptDct2 : リピート検出命令の 2 命令後の命令

RptDct3 : リピート検出命令の 3 命令後の命令

4.5 使用上の注意事項

1. RTE命令の遅延スロットに配置された命令は、SSRに退避されていた値がSRに復帰された後に実行されます。命令アクセスに関する例外の受け付け判定は復帰前のSRの値に応じて決定され、その他の例外の受け付け判定は復帰後のSRによる処理モードやBLビットに依存して決定されます。完了型の例外に関してはRTEの分岐先の実行前に受け付けられますが、再実行型の例外が発生すると動作が保証されません。
2. RTE命令の遅延スロットに配置された命令では、ユーザブレイクの受け付けは行われません。
3. LDC命令によりSRレジスタのMDやBLビットを操作した場合は、その次命令から新しいSRレジスタの値で例外の受け付けを再判定します*。完了型例外では次命令の実行後に例外が受け付けられますが、完了型例外のうち、割り込みおよびDMAアドレスエラーに関しては次命令の実行前に受け付けを行います。

【注】 * SR に対する LDC 命令が実行されると、後続命令への命令フェッチが再び行われ、新しい SR の値で命令フェッチ例外の再評価が行われます。

5. キャッシュ

5.1 特長

- 容量：16kバイトキャッシュ
- 構成：命令とデータ混在、4ウェイセットアソシアティブ
- ロック機能：ウェイ2およびウェイ3はロック可能
- ラインサイズ：16バイト
- エントリ数：256エントリ/ウェイ
- ライト方式：領域グループごとにライトバック方式とライトスルー方式から選択可能。
 - グループ1（P0、P3、U0領域）
 - グループ2（P1領域）
- 置換方式：LRU置換アルゴリズムを採用

5.1.1 キャッシュの構成

キャッシュは、命令とデータ混在型の4ウェイセットアソシアティブ方式です。4つのウェイ（バンク）で構成され、各々のウェイは、アドレスとデータに分かれています。

アドレスとデータは、各々256のエントリで構成されます。エントリのデータをラインと呼びます。1ラインは、16バイト（4バイト×4）です。1ウェイあたりのデータ容量は、4kバイト（16バイト×256エントリ）で、キャッシュ全体（4ウェイ）では16kバイトの容量となります。

キャッシュの構成を図5.1に示します。

5. キャッシュ

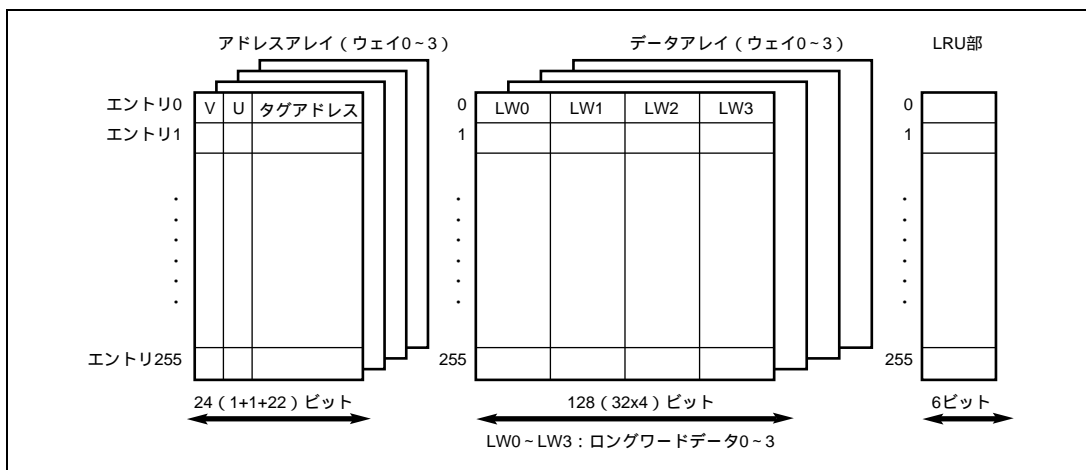


図 5.1 キャッシュの構成

(1) アドレスアレイ

V ビットは、エントリのデータが有効かどうかを表します。V ビットが 1 で有効を、0 で無効を表します。

U ビットは、ライトバックモードでそのエントリに書き込みがあったことを表します。U ビットが 1 で書き込みありを、0 で書き込みなしを表します。

タグアドレスは、外部メモリのアクセスに使用される物理アドレスを保持します。キャッシュ検索時の比較に使用される 22 ビット (メモリへのアクセスアドレスビット 31 ~ 10) からなります。

本 LSI では、物理アドレス 32 ビットの上位 3 ビットをシャドウとして利用するため、タグアドレスの上位 3 ビットに 0 が入ります (「第 9 章 バスステートコントローラ (BSC)」を参照)。

V および U ビットは、パワーオンリセットで 0 に初期化されますが、マニュアルリセットでは初期化されません。タグアドレスは、パワーオンリセットおよびマニュアルリセットでは初期化されません。

(2) データアレイ

データアレイは、16 バイトの命令またはデータを保持します。キャッシュへのエントリの登録は、ライン単位 (16 バイト単位) で行います。

データアレイは、パワーオンリセットおよびマニュアルリセットで初期化されません。

(3) LRU

4 ウェイセットアソシアティブ方式では、エントリアドレスが同じ命令とデータを 4 つまでキャッシュに登録できます。エントリを登録するとき、4 つのウェイのうち、どのウェイに登録するかを LRU ビットで表します。LRU ビットは、6 ビットからなり、ハードウェアで制御します。ウェイ選択のアルゴリズムとして、最も以前にアクセスされたウェイを選ぶ LRU (Least Recently Used) アルゴリズムを使用しています。

キャッシュミスの際にリプレースされるウェイは、6 ビットの LRU ビットによって指定されます。キャッシュロック機能を使用しない場合の LRU ビットとリプレースされるウェイの関係を表 5.1 に示します (キャッシュロ

ック機能を使用する場合に関しては、「5.2.2 キャッシュ制御レジスタ2 (CCR2)」を参照してください)。表 5.1 に示した以外の LRU ビットをソフトウェアで指定した場合は、キャッシュは正しく動作しません。LRU ビットをソフトウェアで変更するときは、表 5.1 に示すパターンを設定してください。

LRU ビットは、パワーオンリセットで B'000000 に初期化されますが、マニュアルリセットでは初期化されません。

表 5.1 LRU ビットと置き換えられるウェイ (キャッシュロック機能を使用しない場合)

LRU (ビット 5-0)	置き換えられるウェイ
000000、000100、010100、100000、110000、110100	3
000001、000011、001011、100001、101001、101011	2
000110、000111、001111、010110、011110、011111	1
111000、111001、111011、111100、111110、111111	0

5.2 レジスタの説明

キャッシュには以下のレジスタがあります。これらのレジスタのアドレスおよびアクセスサイズについては「第 27 章 レジスタ一覧」を参照してください。

- キャッシュ制御レジスタ1 (CCR1)
- キャッシュ制御レジスタ2 (CCR2)

5.2.1 キャッシュ制御レジスタ 1 (CCR1)

キャッシュは、CCR1 の CE ビットでイネーブルまたはディスエーブルを指定します。また、CCR1 には、キャッシュの全エントリの無効化を制御する CF ビット、ライトスルーモードとライトバックモードを切り換える WT ビット、および CB ビットがあります。CCR1 の内容を変更するプログラムは、キャッシングしないアドレス空間に配置してください。

ビット	ビット名	初期値	R/W	説明
31-4	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込み時も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
3	CF	0	R/W	キャッシュフラッシュ 1 を書き込むと、キャッシュの全エントリの V、U、および LRU ビットを 0 にクリア (フラッシュ) します。読み出すと 0 が読み出されます。フラッシュの際は、外部メモリへの書き戻しは行いません。
2	CB	0	R/W	ライトバック P1 領域のライトバックまたはライトスルーの切り替え 0: ライトスルーモード 1: ライトバックモード

5. キャッシュ

ビット	ビット名	初期値	R/W	説明
1	WT	0	R/W	ライトスルー P0、U0、および P3 領域のライトバックまたはライトスルーの切り替え 0: ライトバックモード 1: ライトスルーモード
0	CE	0	R/W	キャッシュ有効 キャッシュ機能を使用するかどうかを表します。 0: 使用しない 1: 使用する

5.2.2 キャッシュ制御レジスタ 2 (CCR2)

CCR2 は、キャッシュロック機能を制御するレジスタです。キャッシュロック機能は、DSP モード時のみ有効です。DSP モードとは、CPU の SR レジスタ (ステータスレジスタ) の DSP ビット (ビット 12) = 1 の状態を言います。非 DSP モード (DSP ビット = 0) では、キャッシュロック機能は無効です。

DSP モード時にプリフェッチ命令 (PREF @Rn) を実行し、キャッシュミスした場合は、CCR2 のビット 9、8 (W3LOAD、W3LOCK) およびビット 1、0 (W2LOAD、W2LOCK) の設定に従って Rn が指し示した 1 ライン分のデータをキャッシュに取り込みます。プリフェッチ命令を実行した場合の各ビットの設定と置換されるウェイの関係は、表 5.2 に示すとおりです。一方、プリフェッチ命令を実行しキャッシュヒットした場合は、新たなデータの取り込みは行われず、既に有効となっているエントリが保持されます。たとえば、Rn が指し示す 1 ライン分のデータが既にウェイ 0 に存在する状態において、DSP モードで、W3LOAD = 1 かつ W3LOCK = 1 と設定し、プリフェッチ命令を実行した場合は、キャッシュヒットとなり、ウェイ 3 へのデータの取り込みは行われません。

DSP モード時の、プリフェッチ命令以外でのキャッシュアクセスでは、W3LOCK、および W2LOCK ビットによって置換されるウェイが制限されます。CCR2 の各ビットの設定と置換されるウェイの関係は、表 5.3 に示すとおりです。

CCR2 の内容を変更するプログラムは、キャッシングしないアドレス空間に配置してください。

ビット	ビット名	初期値	R/W	説明
31 ~ 10	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込み時も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
9	W3LOAD	0	R/W	ウェイ 3 ロード (W3LOAD)
8	W3LOCK	0	R/W	ウェイ 3 ロック (W3LOCK) W3LOCK=1、W3LOAD=1、かつ DSP モードの場合は、プリフェッチ命令でキャッシュミスしたデータは常にウェイ 3 に読み込まれます。その他のすべての条件では、プリフェッチしたデータは LRU の示すウェイに読み込まれます。

ビット	ビット名	初期値	R/W	説明
7~2	-	すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込み時も常に0にしてください。 1を書いた場合の動作の保証はできません。
1	W2LOAD	0	R/W	ウェイ2ロード (W2LOAD)
0	W2LOCK	0	R/W	ウェイ2ロック (W2LOCK) W2LOCK=1、W2LOAD=1、かつ DSP モードの場合は、プリフェッチ命令でキャッシュミスしたデータは常にウェイ2に読み込まれます。その他のすべての条件では、プリフェッチしたデータはLRUの示すウェイに読み込まれます。

【注】 W2LOAD および W3LOAD は、同時に 1 にセットしないでください。

表 5.2 PREF 命令がキャッシュミスした場合に置き換えられるウェイ

DSP ビット	W3LOAD	W3LOCK	W2LOAD	W2LOCK	置き換えられるウェイ
0	*	*	*	*	LRU に従う (表 5.1)
1	*	0	*	0	LRU に従う (表 5.1)
1	*	0	0	1	LRU に従う (表 5.4)
1	0	1	*	0	LRU に従う (表 5.5)
1	0	1	0	1	LRU に従う (表 5.6)
1	0	*	1	1	ウェイ 2
1	1	1	0	*	ウェイ 3

【注】 * : Don't care

W3LOAD = 1 かつ W2LOAD = 1 には、設定しないでください

表 5.3 PREF 命令以外がキャッシュミスした場合に置換されるウェイ

DSP ビット	W3LOAD	W3LOCK	W2LOAD	W2LOCK	置き換えられるウェイ
0	*	*	*	*	LRU に従う (表 5.1)
1	*	0	*	0	LRU に従う (表 5.1)
1	*	0	*	1	LRU に従う (表 5.4)
1	*	1	*	0	LRU に従う (表 5.5)
1	*	1	*	1	LRU に従う (表 5.6)

【注】 * : Don't care

W3LOAD = 1 かつ W2LOAD = 1 には、設定しないでください

5. キャッシュ

表 5.4 LRU ビットと置き換えられるウェイ (W2LOCK=1 かつ W3LOCK=0 の場合)

LRU (ビット 5-0)	置き換えられるウェイ
000000, 000001, 000100, 010100, 100000, 100001, 110000, 110100	3
000011, 000110, 000111, 001011, 001111, 010110, 011110, 011111	1
101001, 101011, 111000, 111001, 111011, 111100, 111110, 111111	0

表 5.5 LRU ビットと置き換えられるウェイ (W2LOCK=0 かつ W3LOCK=1 の場合)

LRU (ビット 5-0)	置き換えられるウェイ
000000, 000001, 000011, 001011, 100000, 100001, 101001, 101011	2
000100, 000110, 000111, 001111, 010100, 010110, 011110, 011111	1
110000, 110100, 111000, 111001, 111011, 111100, 111110, 111111	0

表 5.6 LRU ビットと置き換えられるウェイ (W2LOCK=1 かつ W3LOCK=1 の場合)

LRU (ビット 5-0)	置き換えられるウェイ
000000, 000001, 000011, 000100, 000110, 000111, 001011, 001111, 010100, 010110, 011110, 011111	1
100000, 100001, 101001, 101011, 110000, 110100, 111000, 111001, 111011, 111100, 111110, 111111	0

5.3 動作説明

5.3.1 キャッシュの検索

キャッシュがイネーブルのとき (CCR1 レジスタの CE ビット=1)、P0、P1、P3、および U0 領域の命令またはデータにアクセスすると、キャッシュが検索され、目的の命令またはデータがキャッシュに存在するか調べられます。キャッシュの検索方法の概念図を図 5.2 に示します。キャッシュは物理キャッシュで、タグアドレスには、物理アドレスを保持します。

メモリへのアクセスアドレス (論理) のビット 11~4 でエントリを選択し、そのエントリのタグアドレスを読み出します。メモリへのアクセスアドレス (ビット 31~10) と、アドレスアレイから読み出した物理アドレス (タグアドレス) を比較します。アドレスの比較は、4 ウェイとも行います。比較の結果一致しており、かつ、比較されたエントリが有効である (V=1) 場合には、キャッシュヒットとなります。それ以外の場合は、キャッシュミスとなります。ウェイ 1 がヒットした場合を図 5.2 に示します。

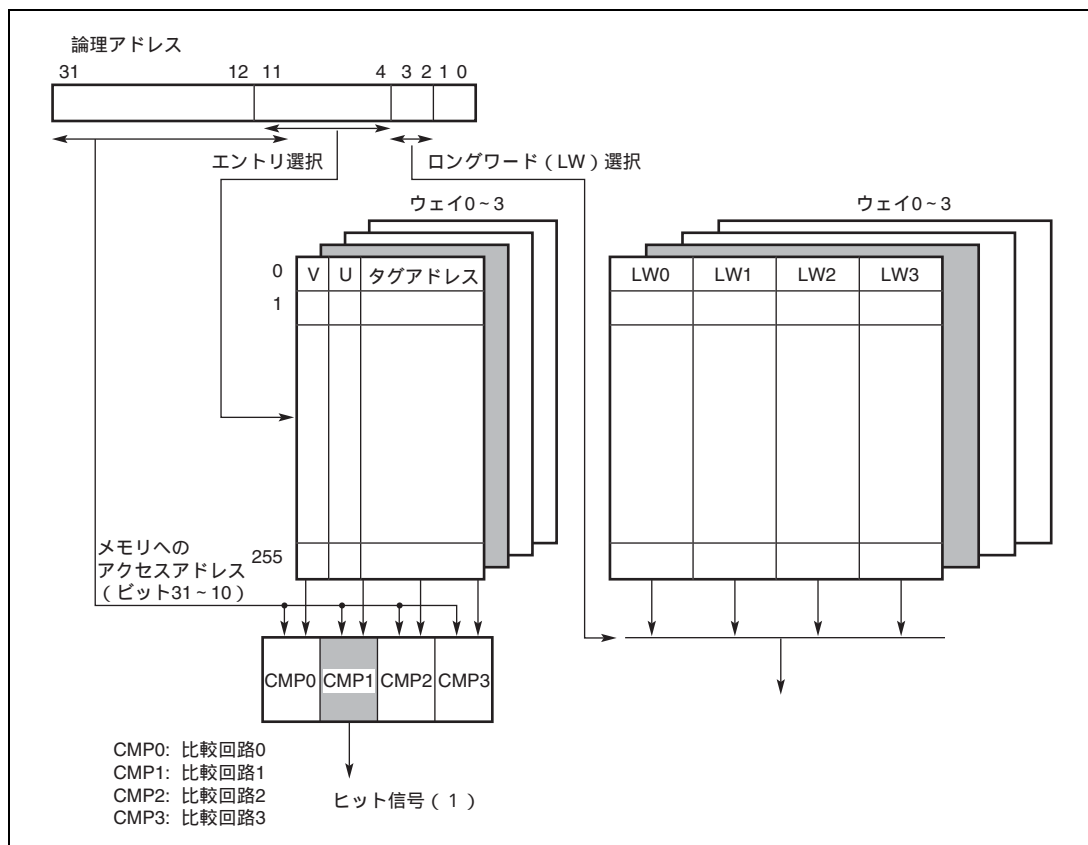


図 5.2 キャッシュの検索方法

5.3.2 リード動作

(1) リードヒット

キャッシュから CPU に命令またはデータが転送されます。ヒットしたウェイが最新となるように、LRU が更新されます。

(2) リードミス

外部バスサイクルを起動し、エントリを更新します。置換するウェイは、表 5.3 に従います。エントリの更新の単位は、16 バイトです。外部メモリから目的の命令またはデータがキャッシュに登録されると同時に、CPU にその命令またはデータが転送されます。キャッシュに登録されるときに、U ビットが 0 に、V ビットが 1 にセットされ、置換されたウェイが最新となるように LRU が更新されます。ライトバックモードでエントリの更新によって置換されるエントリの U ビットが 1 の場合には、そのエントリがライトバックバッファに転送されてから、キャッシュ更新サイクルを開始します。キャッシュ更新サイクルが終了後、ライトバックバッファに転送したエントリをメモリへ書き戻します。書き戻しの単位は、16 バイトです。

5.3.3 プリフェッチ動作

(1) プリフェッチヒット

ヒットしたウェイが最新となるように LRU が更新されます。その他のキャッシュの内容は、変更されません。CPU への命令またはデータの転送は、行われません。

(2) プリフェッチミス

CPU への命令またはデータの転送が行われず、置換するウェイは表 5.2 に従います。その他の動作はリードミスの場合と同じです。

5.3.4 ライト動作

(1) ライトヒット

ライトバックモードでは、キャッシュにデータがライトされ、外部メモリへのライトサイクルは発行されません。ライトされたエントリの U ビットが 1 にセットされ、ヒットしたウェイが最新になるように LRU が更新されます。

ライトスルーモードでは、キャッシュにデータがライトされ、外部メモリへのライトサイクルが発行されます。ライトされたエントリの U ビットは更新されず、ヒットしたウェイが最新になるように LRU が更新されます。

(2) ライトミス

ライトバックモードでは、ライトミス時に外部バスサイクルを起動し、エントリを更新します。置換するウェイは、表 5.3 に従います。エントリの更新によって置換えられるエントリの U ビットが 1 の場合には、そのエントリがライトバックバッファに転送されてから、キャッシュ更新サイクルを開始します。キャッシュにデータがライトされ、U ビットが 1 にセットされ、かつ V ビットも 1 にセットされます。置換したウェイが最新になるように LRU が更新されます。キャッシュ更新サイクル終了後は、ライトバックバッファに転送したエントリをメモリへ書き戻します。書き戻しの単位は、16 バイトです。

ライトスルーモードでは、ライトミス時にキャッシュへのライトを行わず、外部メモリにのみライトを行います。

5.3.5 ライトバックバッファ

ライトバックモードで置換えられるエントリの U ビットが 1 のとき、外部メモリへの書き戻しが必要になります。性能向上のため、置換えられるエントリをまずライトバックバッファに転送し、キャッシュへ新エントリの取り込みを書き戻しに優先させます。キャッシュへの新エントリの取り込み終了後は、ライトバックバッファが外部メモリへの書き戻しを行います。この書き戻し中は、キャッシュはアクセス可能です。

ライトバックバッファは、キャッシュの 1 ライン分のデータ (16 バイト) とその物理アドレスを保持可能です。ライトバックバッファの構成を図 5.3 に示します。

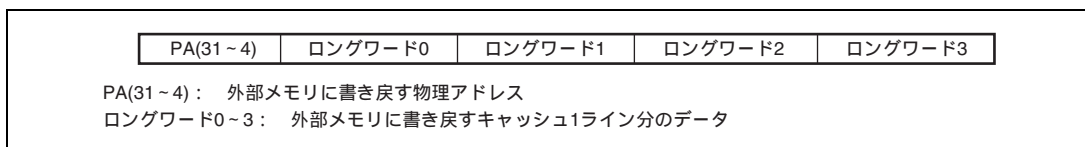


図 5.3 ライトバックバッファの構成

5.3.6 キャッシュと外部メモリとのコヒーレンシ

キャッシュと外部メモリとのコヒーレンシは、ソフトウェアで保証してください。

本 LSI と他の装置との共有メモリをキャッシングするアドレス空間に配置する場合には、必要に応じてメモリ割り付けキャッシュを操作し、無効化およびライトバックを行ってください。本 LSI 内の CPU と DMAC との共有メモリについても同様に行ってください。

5.4 メモリ割り付けキャッシュの構成

キャッシュをソフトウェアで管理するために、特権モードにおいて、MOV 命令により、キャッシュの内容の読み出し、および書き込みが可能です。キャッシュは、論理アドレス空間の P4 領域に割り付けられています。アドレスアレイは H'F000 0000 ~ H'FOFF FFFF に、データアレイは H'F100 0000 ~ H'F1FF FFFF に割り付けられています。アドレスアレイおよびデータアレイともアクセスサイズはロングワード固定であり、命令フェッチは行えません。

5.4.1 アドレスアレイ

アドレスアレイは、H'F0000000 ~ H'FOFF FFFF に割り付けられています。アドレスアレイのアクセスには、32 ビットのアドレスの指定（読み出したまたは書き込み時）と 32 ビットのデータの指定（書き込み時）が必要です。アドレスにはアクセスするエントリを選択するための情報を指定し、データにはアドレスアレイに書き込むタグアドレス、V ビット、U ビットおよび LRU ビットを指定します。

アドレスには、エントリを選択するためのエントリアドレス、ウェイを選択するための W、連想動作の有無を指定する A、およびアドレスアレイアクセスを示す H'F0 を指定します。W は、B'00 がウェイ 0 を、B'01 がウェイ 1 を、B'10 がウェイ 2 を、B'11 がウェイ 3 を表します。

データには、タグアドレス、LRU ビット、U ビット、および V ビットを指定します。

アドレスおよびデータのフォーマットについては、図 5.4 を参照してください。タグアドレスの上位 3 ビット（ビット 31 ~ 29）には、常に 0 を指定してください。

アドレスアレイに対しては、次の 3 種類の操作が可能です。

(1) アドレスアレイリード

アドレスに指定されたエントリアドレスおよびウェイに対応するエントリからタグアドレス、LRU ビット、U ビット、および V ビットを読み出します。リードの場合は、アドレスに指定される連想ビット（A ビット）は 1 でも 0 でも連想動作は行いません。

5. キャッシュ

(2) アドレスアレイライト (連想なし)

アドレスに指定されたエントリアドレスおよびウェイに対応するエントリに対して、データで指定されたタグアドレス、LRU ビット、U ビット、および V ビットを書き込みます。アドレスの連想ビット (A ビット) は、0 にしてください。書き込みを U ビットが 1、V ビットが 1 のキャッシュラインに対して行った場合は、そのキャッシュラインの書き戻しを行った後に、データで指定されたタグアドレス、LRU ビット、U ビット、および V ビットを書き込みます。ただし、V ビットに 0 を書き込むときは、必ずそのエントリの U ビットにも 0 を書き込んでください。

(3) アドレスアレイライト (連想あり)

アドレスの連想ビット (A ビット) を 1 にしてライトした場合は、アドレスで指定されたエントリの 4 ウェイ全てに対して、データで指定されたタグアドレスとの間で一致判定が行われます。一致判定の結果ヒットしたウェイに対して、データで指定された U ビットと V ビットをエントリに書き込みます。ただし、タグアドレスと LRU ビットは、変更されません。どのウェイにもヒットしなかった場合は、書き込みを行わず、ノーオペレーションとなります。本動作は、キャッシュの特定エントリの無効化に用いられます。このときヒットしたエントリの U ビットが 1 だった場合は、書き戻しが発生します。ただし、V ビットに 0 を書き込むときは、必ずそのエントリの U ビットにも 0 を書き込んでください。

5.4.2 データアレイ

データアレイは、HF100 0000 ~ HF1FF FFFF に割り付けられています。データアレイのアクセスには、32 ビットのアドレスの指定 (読み出しまたは書き込み時) と 32 ビットのデータの指定 (書き込み時) が必要です。アドレスにはアクセスするエントリを選択するための情報を指定し、データにはデータアレイに書き込むロングワードデータを指定します。

アドレスにはエントリを選択するためのエントリアドレス、1 ライン (16 バイト) 中のロングワード位置を示す L、ウェイを指定するための W、およびデータアレイアクセスを示す HF1 を指定します。L は、B'00 がロングワード 0 を、B'01 がロングワード 1 を、B'10 がロングワード 2 を、B'11 がロングワード 3 を表します。W は、B'00 がウェイ 0 を、B'01 がウェイ 1 を、B'10 がウェイ 2 を、B'11 がウェイ 3 を表します。アクセスはロングワードサイズ固定なので、アドレスのビット 1~0 には B'00 を指定してください。

アドレスおよびデータのフォーマットについては、図 5.4 を参照してください。

データアレイに対しては、次の 2 種類の操作が可能です。なお、この操作によってアドレスアレイの情報を変更されることはありません。

(1) データアレイリード

アドレスに指定されたエントリアドレスおよびウェイに対応するエントリから、アドレスの L で指定されたデータを読み出します。

(2) データレイライト

アドレスに指定されたエントリアドレスおよびウェイに対応するエントリのうち、アドレスのLで指定された位置に、データで指定されたロングワードデータを書き込みます。

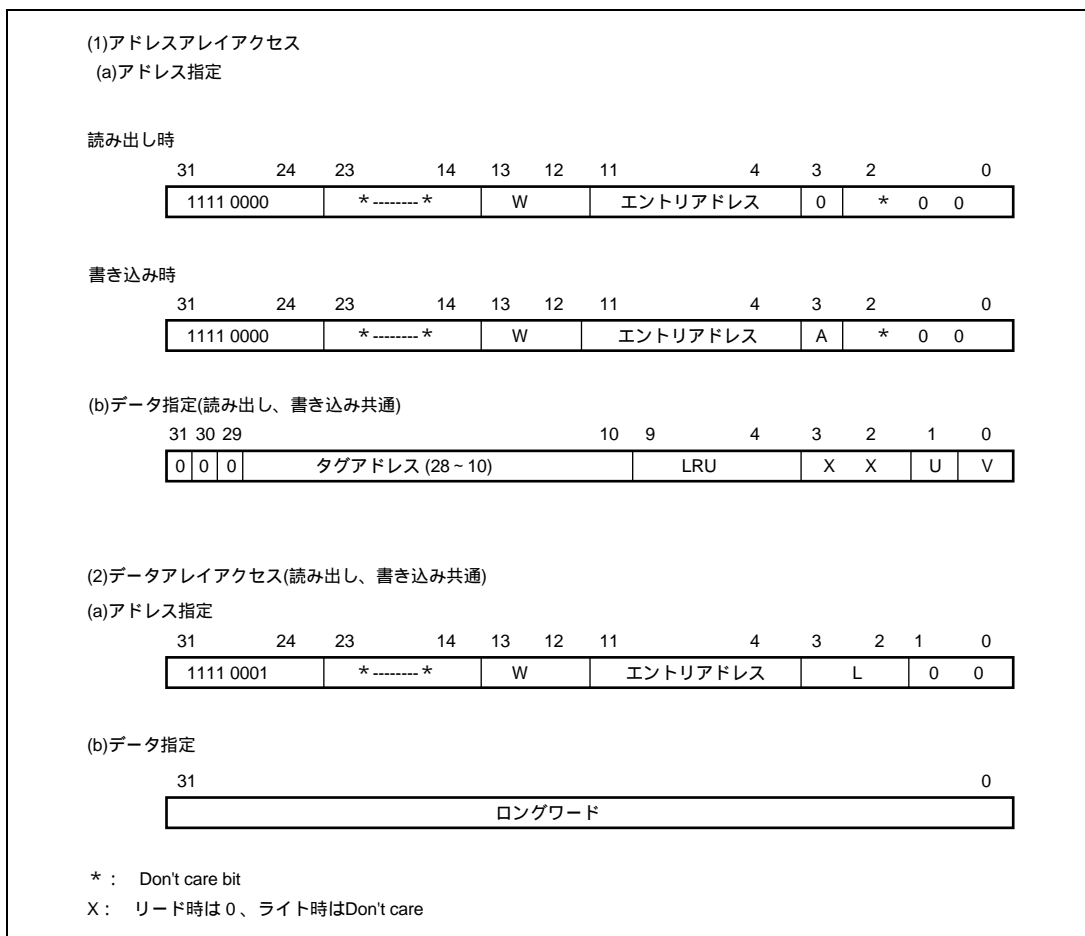


図 5.4 メモリ割り付けキャッシュアクセスのアドレス、データ指定方法

5.4.3 使用例

(1) 特定エントリの無効化

キャッシュの特定エントリの無効化は、メモリ割り付けキャッシュアクセスにおいてそのエントリのVビットに0を書き込むことで実現できます。Aビットを1とし、書き込みデータで指定されるタグアドレスをエントリアドレスで選択されたキャッシュ中のタグアドレスと比較し、一致したときにそのエントリのUビットが1だった場合はそのエントリがライトバックされ、書き込みデータで指定されたVビットおよびUビットを書き込みます。一致しない場合は、ノーオペレーションです。

以下に、R0に書き込みデータを、R1にアドレスを指定した場合の例を示します。

```
; R0 = H'0110 0010; アドレスアレイアクセスのデータ指定、タグアドレス = B'0 0001 0001 0000 0000 00、LRU = B'00 0001、U = 0、V = 0
```

```
; R1 = H'F000 0088; アドレスアレイアクセスのアドレス指定、ウェイ = B'00、エントリアドレス = B'00001000、A = 1
```

```
;
```

```
MOV.L R0, @R1
```

(2) 特定エントリのデータ部の読み出し

特定エントリのデータ部の読み出しは、メモリ割り付けキャッシュアクセスで可能です。図 5.4 のデータアレイのデータ部に示されるロングワードが、レジスタに読み出されます。

以下に、R0にアドレスを指定し、R1に読み出す例を示します。

```
; R0 = H'F100 004C; データアレイアクセスのアドレス指定、ウェイ = B'00、エントリアドレス = B'00000100、L = B'11
```

```
;
```

```
MOV.L @R0, R1 ; ロングワード3が読み出されます。
```

6. X/Y メモリ

本 LSI は、おのおの 8k バイトの X メモリおよび Y メモリモジュールを内蔵しており、命令やデータを格納することができます。

6.1 特長

- ページ：
Xメモリが2ページ（ページ0および1）とYメモリが2ページ（ページ0および1）の合計4ページが存在します。
- メモリマップ：
本メモリは、論理アドレス空間、物理アドレス空間にそれぞれ配置されています。
論理アドレス空間内では、表6.1に示されるP2/Uxy領域内のアドレスに配置されます。これらのアドレスは、CPUの動作モードによって、P2（SR.MD=1の場合）、あるいはUxy（SR.MD=0かつSR.DSP=1の場合）となります。

表 6.1 X/Y メモリ論理アドレス

ページ	メモリサイズ（4ページ合計）16k バイト
Xメモリ ページ0	H'A5007000 ~ H'A5007FFF
Xメモリ ページ1	H'A5008000 ~ H'A5008FFF
Yメモリ ページ0	H'A5017000 ~ H'A5017FFF
Yメモリ ページ1	H'A5018000 ~ H'A5018FFF

一方、物理アドレス空間内では、表6.1に示す論理アドレスはエリア1の一部に配置されます。物理アドレスでアクセスを行う場合は、表6.1に示すアドレスの上位3bitを0としたアドレスを使用します。

- ポート：
各ページは、3本の独立した読み出しまたは書き込みのポートを持ち、各バスと接続されています。XメモリはIバス、Xバス、およびLバスと、YメモリはIバス、Yバス、およびLバスと接続されています。論理アドレス空間からのアクセスにはLバス、Xバス、Yバスが、物理アドレス空間からのアクセスにはIバスが使用されます。
- 優先順位：
同じページに対して異なるバスから同時にアクセス要求があった場合には、優先順位に従ってアクセスが処理されます。優先順位は、高い順にXメモリではIバス、Xバス、Lバスとなり、YメモリではIバス、Yバス、Lバスとなります。

6.2 動作説明

6.2.1 CPU からのアクセス

CPU からは、L バスまたは I バス経由で 8/16/32 ビットのアクセスが可能です。

CPU からのアクセス手段として、論理アドレスにより L バスからアクセスを行う方法と、物理アドレスにより I バスからアクセスを行う方法があります。L バスからのアクセスは、ページ競合が発生しない限り 1 サイクルアクセスになります。I バスからのアクセスは、複数サイクル必要となります。CPU の動作モードに応じてそれぞれ以下ようになります。

(1) 特権モードおよび特権 DSP モード (SR.MD=1)

このモードでは、P0 および P2 領域からアクセスすることができます。

(2) ユーザ DSP モード (SR.MD=0 かつ SR.DSP=1)

このモードでは、U0 および U_{xy} 領域からアクセスすることができます。

(3) ユーザモード (SR.MD=0 かつ SR.DSP=0)

このモードでは、U0 領域からアクセスすることができます。

6.2.2 DSP からのアクセス

DSP からは、L バスまたは I バス経由で 16/32 ビットのアクセスと、X バスおよび Y バス経由で 16 ビットのアクセスが可能です。

DSP からのアクセスは、命令の種類によりアクセス方法が異なります。

X データ転送命令および Y データ転送命令は、常に X バスおよび Y バスからのアクセスになります。この場合は、ページ競合が発生しない限り 1 サイクルアクセスになります。また、X バスからの X メモリアクセスと、Y バスからの Y メモリアクセスは、同時に行うことができます。

シングルデータ転送命令は、アクセス手段として、論理アドレスにより L バスからアクセスを行う方法と、物理アドレスにより I バスからアクセスを行う方法があります。L バスからのアクセスは、ページ競合が発生しない限り 1 サイクルアクセスになります。I バスからのアクセスは、複数サイクル必要となります。CPU の動作モードに応じてそれぞれ以下ようになります。

(1) 特権 DSP モード (SR.MD=1 かつ SR.DSP=1)

このモードでは、P0 および P2 領域からアクセスすることができます。

(2) ユーザ DSP モード (SR.MD=0 かつ SR.DSP=1)

このモードでは、U0 および U_{xy} 領域からアクセスすることができます。

6.2.3 Iバスマスタモジュールからのアクセス

DMAC、USBHなどのIバスマスタモジュールからの本メモリへのアクセスは、常に物理アドレスバスであるIバスからのアクセスとなります。

DMACは8/16/32ビットのアクセスを、USBHは8/32ビットのアクセスを行います。アドレスはP4領域(A31～A29=B'111)以外では、上位3ビットが内部で強制的にB'000とされるので、P4領域以外の論理アドレスを設定しても、Iバスからのアクセスとなります。

6.3 使用上の注意事項

6.3.1 ページ競合

同じページに対して異なるバスから同時にアクセス要求が発生した場合は、ページ競合となります。各アクセスは正しく完了しますが、このような競合はメモリアccessの性能低下を招きます。したがって、できるだけ競合が起こらないようにソフトウェアでの対策を推奨します。たとえば、各バスごとに異なるメモリや異なるページをアクセスすれば競合は発生しません。

6.3.2 バス競合

Iバスは、複数のバスマスタモジュールによる共有バスです。このため、Iバス経由のアクセスは、Iバス上で他のIバスマスタモジュールとの間でバス競合が発生する場合があります。バス競合が発生するとメモリアccessの性能低下を招きますので、できるだけ競合が起こらないようにソフトウェアでの対策を推奨します。たとえば、CPUによる本メモリアccessでは、Iバス経由を避け、P2領域またはU_{xy}領域からLバス経由でアクセスすることによって、Iバス上での競合を回避できます。

6.3.3 キャッシュの設定

CPUとDSPからキャッシュを利用してIバス経由で本メモリアccessした場合には、動作を保証しません。キャッシュを有効(CCR1.CE=1)にして使用する場合には、P2またはU_{xy}領域からLバス経由でアクセスしてください。高い性能が必要なプログラムでは、P2またはU_{xy}領域からアクセスすることを推奨します。以上の関係を表6.2にまとめます。

表 6.2 キャッシュの設定

設定	アドレス領域とアクセスの可否	
	P0、U0	P2、U _{xy}
0		
1	x	

【注】 : 可 (推奨)

: 可

x : 不可

6. X/Y メモリ

6.3.4 アドレスエラー

X/Y メモリに対してアドレスエラーを起こす書き込みを行った場合、X/Y メモリの内容が壊れる場合があります。

6.3.5 リセット解除後のメモリの内容

パワーオンリセット解除後、マニュアルリセット解除後および $\overline{\text{TRST}}$ 端子アサート解除後のメモリの内容は保証されません。

7. Uメモリ

本 LSI は、128k バイトの U メモリモジュールを内蔵しており、命令やデータを格納することができます。

7.1 特長

- ページ：

最大2ページ（ページ0、1）存在します。

- メモリマップ：

本メモリは、論理アドレス空間および物理アドレス空間にそれぞれ配置されています。

論理アドレス空間内では、表7.1に示されるP2/U_{xy}領域内のアドレスに配置されています。これらのアドレスは、CPUの動作モードによってP2（SR.MD=1の場合）あるいはU_{xy}（SR.MD=0かつSR.DSP=1の場合）となります。

表 7.1 Uメモリ論理アドレス

ページ	メモリサイズ（合計）128k バイト(2 ページ)
Uメモリ ページ0	H'A55F0000 ~ H'A55FFFFFF
Uメモリ ページ1	H'A5600000 ~ H'A560FFFF

一方、物理アドレス空間内では、表7.1に示す論理アドレスはエリア1の一部に配置されます。物理アドレスでアクセスを行う場合は、表7.1に示すアドレスの上位3ビットを0としたアドレスを使用します。

- ポート：

各ページは、2本の独立した読み出しまたは書き込みポートを持ち、Iバス、およびLバスと接続されています。論理アドレス空間からのアクセスにはLバスが、また物理アドレス空間からのアクセスにはIバスが使用されます。

- 優先順位：

同じページに対して異なるバスから同時にアクセス要求があった場合には、優先順位に従ってアクセスが処理されます。優先順位は、高い順にIバス、Lバスとなります。

7.2 動作説明

7.2.1 CPUからのアクセス

CPUからは、LバスまたはIバス経由で8/16/32ビットのアクセスが可能です。

CPUからのアクセス手段として、論理アドレスによりLバスからアクセスを行う方法と、物理アドレスによりIバスからアクセスを行う方法があります。Lバスからのアクセスは、ページ競合が発生しない限り1サイクルアクセスになります。Iバスからのアクセスは、複数サイクル必要となります。CPUの動作モードに応じてそれぞれ以下ようになります。

(1) 特権モードおよび特権 DSP モード (SR.MD=1)

このモードでは、P0 および P2 領域からアクセスすることができます。

(2) ユーザ DSP モード (SR.MD=0 かつ SR.DSP=1)

このモードでは、U0 および U_{xy} 領域からアクセスすることができます。

(3) ユーザモード (SR.MD=0 かつ SR.DSP=0)

このモードでは、U0 領域からアクセスすることができます。

7.2.2 DSPからのアクセス

DSPからは、シングルデータ転送命令でのみアクセス可能です。LバスまたはIバス経由で16/32ビットのアクセスを行います。Xデータ転送命令およびYデータ転送命令では、アクセスできません。

DSPからのアクセス手段として、論理アドレスによりLバスからアクセスを行う方法と、物理アドレスによりIバスからアクセスを行う方法があります。Lバスからのアクセスは、ページ競合が発生しない限り1サイクルアクセスになります。Iバスからのアクセスは、複数サイクル必要となります。CPUの動作モードに応じてそれぞれ以下ようになります。

(1) 特権 DSP モード (SR.MD=1 かつ SR.DSP=1)

このモードでは、P0 および P2 領域からアクセスすることができます。

(2) ユーザ DSP モード (SR.MD=0 かつ SR.DSP=1)

このモードでは、U0 および U_{xy} 領域からアクセスすることができます。

7.2.3 Iバスマスタモジュールからのアクセス

DMAC、USBHなどのIバスマスタモジュールからの本メモリへのアクセスは、常に物理アドレスバスであるIバスからのアクセスとなります。

DMACは8/16/32ビットのアクセスを、USBHは8/32ビットのアクセスを行います。アドレスはP4領域(A31~A29=B'111)以外では、上位3ビットが内部で強制的にB'000とされるので、P4領域以外の論理アドレスを設定しても、Iバスからのアクセスとなります。

7.3 使用上の注意事項

7.3.1 ページ競合

同じページに対して異なるバスから同時にアクセス要求が発生した場合は、ページ競合となります。各アクセスは正しく完了しますが、このような競合はメモリアccessの性能低下を招きます。したがって、できるだけ競合が起こらないようにソフトウェアでの対策を推奨します。たとえば、各バスごとに異なるメモリや異なるページをアクセスすれば競合は発生しません。

7.3.2 バス競合

Iバスは、複数のバスマスタモジュールによる共有バスです。このため、Iバス経由のアクセスは、Iバス上で他のIバスマスタモジュールとの間でバス競合が発生する場合があります。バス競合が発生するとメモリアccessの性能低下を招きますので、できるだけ競合が起こらないようにソフトウェアでの対策を推奨します。たとえば、CPUによる本メモリアccessでは、Iバス経由を避け、P2領域またはU_{xy}領域からLバス経由でアクセスすることによって、Iバス上での競合を回避できます。

7.3.3 キャッシュの設定

CPUとDSPからキャッシュを利用してIバス経由で本メモリアccessした場合には、動作を保証しません。キャッシュを有効(CCR1.CE=1)にして使用する場合には、P2またはU_{xy}領域からLバス経由でアクセスしてください。高い性能が必要なプログラムでは、P2またはU_{xy}領域からアクセスすることを推奨します。以上の関係を表7.2にまとめます。

表 7.2 キャッシュの設定

設定	アドレス領域とアクセスの可否	
	P0、U0	P2、U _{xy}
CCR1.CE = 0		
CCR1.CE = 1	x	

【注】 : 可 (推奨)

: 可

x : 不可

7. Uメモリ

7.3.4 アドレスエラー

Uメモリに対してアドレスエラーを起こす書き込みを行った場合、Uメモリの内容が壊れる場合があります。

7.3.5 リセット解除後のメモリの内容

パワーオンリセット解除後、マニュアルリセット解除後および $\overline{\text{TRST}}$ 端子アサート解除後のメモリ内容は保証されません。

8. 割り込みコントローラ (INTC)

割り込みコントローラ (INTC) は、割り込み要因の優先順位を判定し、CPU への割り込み要求を制御します。INTC には、各割り込みの優先順位を設定するためのレジスタがあり、ユーザがこのレジスタに設定した優先順位に従って、割り込み要求が処理されます。

8.1 特長

- 割り込み優先順位を16レベル設定可能

割り込み優先レベル設定レジスタにより、内蔵周辺モジュール、IRQ割り込みの優先順位を割り込み要求元別に16レベルまで設定することができます。

- NMIノイズキャンセル機能

NMI端子の状態を示すNMI入力レベルビットを持っています。割り込み例外サービスルーチンでこのビットを読むことにより端子状態を確認でき、ノイズキャンセラとして使用できます。

- IRQ割り込みを設定可能

ローレベル、ハイレベル、立ち上がり、立ち下がり

- 割り込み要求通知信号を外部へ出力可能 ($\overline{\text{IRQOUT}}$ 端子)

外部割り込みや内蔵周辺モジュール割り込み要求が発生したことを外部バスマスタに知らせることにより、バス権を要求することができます。

- 割り込みの禁止・許可を設定可能

割り込みマスクレジスタ、割り込みマスククリアレジスタを持ち、割り込み要因別に割り込みを禁止または許可することができます。

INTC のブロック図を図 8.1 に示します。

8. 割り込みコントローラ (INTC)

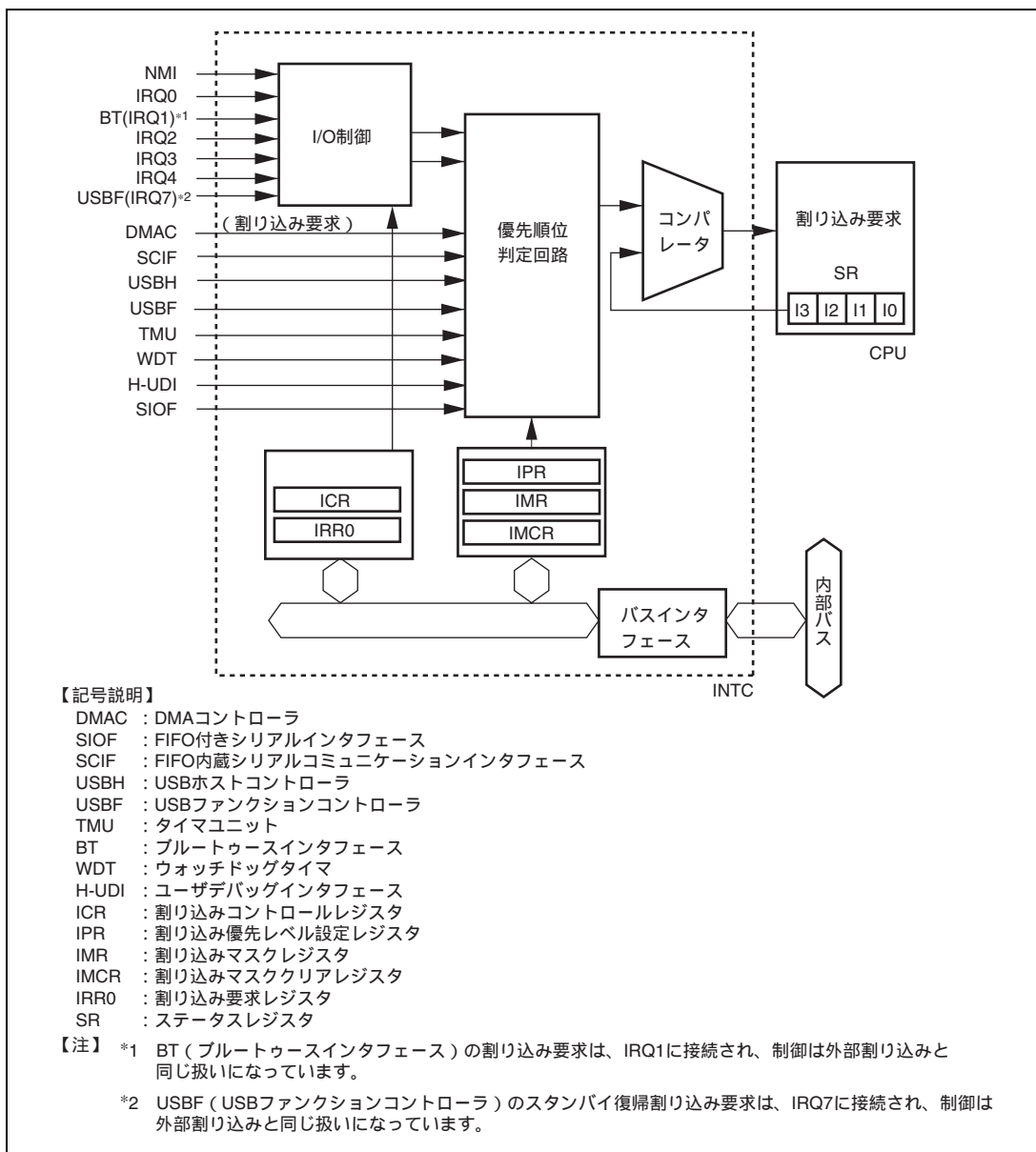


図 8.1 INTC のブロック図

8.2 入出力端子

割り込みコントローラの端子構成を表 8.1 に示します。

表 8.1 端子構成

端子名	入出力	機能
NMI	入力	ノンマスクابل割り込み入力端子 マスク不可能な割り込み要求信号の入力
IRQ7、IRQ4～IRQ0* ¹	入力	割り込み入力端子 割り込み要求信号の入力
IRQOUT* ²	出力	割り込み要求通知端子 割り込み要求が発生したことを通知する信号

【注】 *1 IRQ1 はブルートゥースインタフェース (BT) の割り込み要求として使われていますので外部端子としてはサポートされていません。また、IRQ7 は、スタンバイ復帰割り込み要求として使われていますので外部端子としてはサポートされていません。

*2 NMI または H-UDI 割り込み要求が発生し、CPU の割り込み応答時間が短い場合には、アサートされないことがあります。

8.3 レジスタの説明

割り込みコントローラには、以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態によるレジスタの状態については、「第 27 章 レジスタ一覧」を参照してください。

- 割り込みコントロールレジスタ0 (ICR0)
- 割り込みコントロールレジスタ1 (ICR1)
- 割り込みコントロールレジスタ2 (ICR2)
- 割り込み優先レベル設定レジスタA (IPRA)
- 割り込み優先レベル設定レジスタB (IPRB)
- 割り込み優先レベル設定レジスタC (IPRC)
- 割り込み優先レベル設定レジスタD (IPRD)
- 割り込み優先レベル設定レジスタE (IPRE)
- 割り込み優先レベル設定レジスタF (IPRF)
- 割り込み優先レベル設定レジスタG (IPRG)
- 割り込み優先レベル設定レジスタH (IPRH)
- 割り込み要求レジスタ0 (IRR0)
- 割り込みマスクレジスタ0 (IMR0)
- 割り込みマスクレジスタ1 (IMR1)
- 割り込みマスクレジスタ4 (IMR4)

8. 割り込みコントローラ (INTC)

- 割り込みマスクレジスタ5 (IMR5)
- 割り込みマスクレジスタ6 (IMR6)
- 割り込みマスクレジスタ9 (IMR9)
- 割り込みマスククリアレジスタ0 (IMCR0)
- 割り込みマスククリアレジスタ1 (IMCR1)
- 割り込みマスククリアレジスタ4 (IMCR4)
- 割り込みマスククリアレジスタ5 (IMCR5)
- 割り込みマスククリアレジスタ6 (IMCR6)
- 割り込みマスククリアレジスタ9 (IMCR9)

8.3.1 割り込み優先レベル設定レジスタ A~H (IPRA~IPRH)

IPRA~IPRHは、内蔵周辺モジュール、IRQ割り込みに対して0~15までの優先順位レベルをセットする16ビットの読み出し/書き込みレジスタです。

ビット	ビット名	初期値	R/W	説明
15	IPR15	0	R/W	4ビット単位で、各割り込み要因の割り込み優先レベルを設定します。詳細は表 8.2 割り込み要求要因と IPRA~IPRHを参照してください。
14	IPR14	0	R/W	
13	IPR13	0	R/W	
12	IPR12	0	R/W	
11	IPR11	0	R/W	
10	IPR10	0	R/W	
9	IPR9	0	R/W	
8	IPR8	0	R/W	
7	IPR7	0	R/W	
6	IPR6	0	R/W	
5	IPR5	0	R/W	
4	IPR4	0	R/W	
3	IPR3	0	R/W	
2	IPR2	0	R/W	
1	IPR1	0	R/W	
0	IPR0	0	R/W	

表 8.2 割り込み要求要因と IPRA ~ IPRH

レジスタ名	ビット 15~12	ビット 11~8	ビット 7~4	ビット 3~0
IPRA	TMU0	TMU1	TMU2	予約*1
IPRB	WDT	予約*1	予約*1	予約*1
IPRC	IRQ3	IRQ2	IRQ1 (BT 用)*2	IRQ0
IPRD	IRQ7 (USBF スタンバイ復帰用)*3	予約*1	予約*1	IRQ4
IPRE	DMAC	予約*1	予約*1	予約*1
IPRF	予約*1	予約*1	USBF	USBH
IPRG	SCIF0	SCIF1	予約*1	予約*1
IPRH	予約*1	SIOF	予約*1	予約*1

【注】 *1 予約：読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

*2 IRQ1 には、BT (ブルートゥースインタフェース) の割り込み要求信号が接続されていますので、別の用途には使用できません。

*3 IRQ7 には、USBF (USB ファンクションコントローラ) のスタンバイ復帰割り込み要求信号が接続されていますので、別の用途には使用できません。

表 8.2 に示すように、各レジスタには 4 組の内蔵周辺モジュール、または IRQ 割り込みが割り当てられます。4 ビットグループ (ビット 15~12、ビット 11~8、ビット 7~4、ビット 3~0) は、H'0 (0000) ~ HF (1111) の値に設定することができます。設定 H'0 は優先順位レベル 0 (要求マスク) を意味し、HF は優先順位レベル 15 (最高レベル) です。

8. 割り込みコントローラ (INTC)

8.3.2 割り込みコントロールレジスタ 0 (ICR0)

ICR0は外部割り込み入力端子NMIの入力検出モードを設定し、NMI端子に対する入力信号レベルを示します。

ビット	ビット名	初期値	R/W	説明
15	NMIL	0/1*	R	NMI入力レベル NMI端子に信号の入力レベルをセットします。このビットを読み出してNMI端子レベルを判断することができます。このビットは、変更不可能です。 0: NMI入力レベルはロー 1: NMI入力レベルはハイ
14~9	-	すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
8	NMIE	0	R/W	NMIエッジ選択 NMI入力の立下りまたは立上がりエッジのどちらで割り込み要求信号を検出するかを選択します。 0: NMI入力の立下りエッジで割り込み要求信号を検出 1: NMI入力の立上がりエッジで割り込み要求信号を検出
7~0	-	すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。

【注】 * NMI入力がハイレベル時は1、NMI入力がローレベル時は0になります。

8.3.3 割り込みコントロールレジスタ 1 (ICR1)

ICR1 は、外部割り込み入力端子 IRQ4 ~ IRQ0 に対して立ち上がりエッジ、立ち下がりエッジ、ローレベル、およびハイレベルの検出モードを個別に指定する 16 ビットのレジスタです。

ビット	ビット名	初期値	R/W	説明																		
15	-	0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。																		
14	IRQE*	1	R/W	割り込み要求イネーブル IRQ7、IRQ4 ~ IRQ0 端子を 6 本の独立した割り込み端子として使用許可 / 禁止を選択します。 0 : 6 本の独立した割り込み端子 IRQ7、IRQ4 ~ IRQ0 として使用許可 1 : IRQ7、IRQ4 ~ IRQ0 端子は割り込み端子として使用禁止																		
13 ~ 10	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。																		
9	IRQ41S	0	R/W	IRQn センスセレクト IRQ4 ~ IRQ0 端子に対する割り込み信号を立ち下がりエッジ、立ち上がりエッジ、ローレベル、ハイレベルのどれで検出するかを選択します。																		
8	IRQ40S	0	R/W																			
7	IRQ31S	0	R/W																			
6	IRQ30S	0	R/W																			
5	IRQ21S	0	R/W																			
4	IRQ20S	0	R/W																			
3	IRQ11S	0	R/W																			
2	IRQ10S	0	R/W																			
1	IRQ01S	0	R/W																			
0	IRQ00S	0	R/W																			
<table border="1"> <thead> <tr> <th>ビット 2n+1</th> <th>ビット 2n</th> <th></th> </tr> </thead> <tbody> <tr> <td>IRQn1S</td> <td>IRQn0S</td> <td></td> </tr> <tr> <td>0</td> <td>0</td> <td>割り込み要求を IRQn 入力の立ち下がりエッジで検出する</td> </tr> <tr> <td>0</td> <td>1</td> <td>割り込み要求を IRQn 入力の立ち上がりエッジで検出する</td> </tr> <tr> <td>1</td> <td>0</td> <td>割り込み要求を IRQn 入力のローレベルで検出する</td> </tr> <tr> <td>1</td> <td>1</td> <td>割り込み要求を IRQn 入力のハイレベルで検出する</td> </tr> </tbody> </table> <p>【記号説明】 n=0~4</p>					ビット 2n+1	ビット 2n		IRQn1S	IRQn0S		0	0	割り込み要求を IRQn 入力の立ち下がりエッジで検出する	0	1	割り込み要求を IRQn 入力の立ち上がりエッジで検出する	1	0	割り込み要求を IRQn 入力のローレベルで検出する	1	1	割り込み要求を IRQn 入力のハイレベルで検出する
ビット 2n+1	ビット 2n																					
IRQn1S	IRQn0S																					
0	0	割り込み要求を IRQn 入力の立ち下がりエッジで検出する																				
0	1	割り込み要求を IRQn 入力の立ち上がりエッジで検出する																				
1	0	割り込み要求を IRQn 入力のローレベルで検出する																				
1	1	割り込み要求を IRQn 入力のハイレベルで検出する																				

【注】 * リセット後の初期化ルーチンでステータスレジスタ (SR) の BL ビットを 0 クリアする前に必ず IRQE ビットを 0 に設定し、以後本ビットを変更しないでください。BL ビットが 0 の状態で IRQE ビットを 1 にした状態は動作保証ができません。

8. 割り込みコントローラ (INTC)

8.3.4 割り込みコントロールレジスタ 2 (ICR2)

ICR2 は、外部割り込み入力端子 IRQ7 に対して立ち上がりエッジ、立ち下がりエッジ、ローレベル、およびハイレベルの検出モードを個別に指定する 16 ビットのレジスタです。ただし、本 LSI では IRQ7 には、USBF (USB ファンクションコントローラ) のスタンバイ復帰割り込み要求信号が接続されていますので、別の用途には使用できません。そのため、検出方法は立ち下がりエッジのみに固定されているので、これ以外は設定しないでください。

ビット	ビット名	初期値	R/W	説明																		
15~4	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。																		
3 2	IRQ71S IRQ70S	0 0	R/W R/W	IRQ7 センスセレクト IRQ7 端子に対する割り込み信号を立ち下がりエッジ、立ち上がりエッジ、ローレベル、ハイレベルのどれで検出するかを選択します。 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ビット 2n+1</th> <th>ビット 2n</th> <th></th> </tr> <tr> <th>IRQ71S</th> <th>IRQ70S</th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>割り込み要求 (USBF スタンバイ復帰割り込み要求) を IRQ7 入力の立ち下がりエッジで検出する</td> </tr> <tr> <td>0</td> <td>1</td> <td>割り込み要求を IRQ7 入力の立ち上がりエッジで検出する。ただし、本 LSI では設定禁止。</td> </tr> <tr> <td>1</td> <td>0</td> <td>割り込み要求を IRQ7 入力のローレベルで検出する。ただし、本 LSI では設定禁止。</td> </tr> <tr> <td>1</td> <td>1</td> <td>割り込み要求を IRQ7 入力のハイレベルで検出する。ただし、本 LSI では設定禁止。</td> </tr> </tbody> </table> <p>【注】 USBF スタンバイ復帰割り込み要求を検出しない場合は、IPRD レジスタのビット 15~12 を 0 にするが、IMR0 のビット 7 を 1 にするがし、IRQ7 割り込み要求をマスクしてください。</p>	ビット 2n+1	ビット 2n		IRQ71S	IRQ70S		0	0	割り込み要求 (USBF スタンバイ復帰割り込み要求) を IRQ7 入力の立ち下がりエッジで検出する	0	1	割り込み要求を IRQ7 入力の立ち上がりエッジで検出する。ただし、本 LSI では設定禁止。	1	0	割り込み要求を IRQ7 入力のローレベルで検出する。ただし、本 LSI では設定禁止。	1	1	割り込み要求を IRQ7 入力のハイレベルで検出する。ただし、本 LSI では設定禁止。
ビット 2n+1	ビット 2n																					
IRQ71S	IRQ70S																					
0	0	割り込み要求 (USBF スタンバイ復帰割り込み要求) を IRQ7 入力の立ち下がりエッジで検出する																				
0	1	割り込み要求を IRQ7 入力の立ち上がりエッジで検出する。ただし、本 LSI では設定禁止。																				
1	0	割り込み要求を IRQ7 入力のローレベルで検出する。ただし、本 LSI では設定禁止。																				
1	1	割り込み要求を IRQ7 入力のハイレベルで検出する。ただし、本 LSI では設定禁止。																				
1, 0	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にするか、IMR0 のビット 7 を 1 にするかしてください。1 を書いた場合の動作の保証はできません。																		

8.3.5 割り込み要求レジスタ 0 (IRR0)

IRR0 は、外部入力端子 IRQ7、IRQ4*~IRQ0 からの割り込み要求を示す 8 ビットレジスタです。

【注】 * 本 LSI では、IRQ7、IRQ1 は内部モジュール専用として使われているため、実際には外部端子としてはサポートされていません。

ビット	ビット名	初期値	R/W	説明
7	IRQ7R* ²	0	R/W	IRQn 割り込み要求
6* ¹	-	0	R	IRQn 端子に割り込み要求が入力されているかを示します。
5* ¹	-	0	R	IRQn 端子がエッジ検出モードに設定されている場合、割り込み要求は
4	IRQ4R* ²	0	R/W	IRQnR ビットの 1 を読み出した後に、0 を書き込むことでクリアされます。
3	IRQ3R* ²	0	R/W	IRQn 端子がレベル検出モードに設定されている場合、割り込み要求が入力
2	IRQ2R* ²	0	R/W	されているかどうかを示します。IRQn 端子の入力値でのみ、セット/クリアすることができます。
1	IRQ1R* ²	0	R/W	
0	IRQ0R* ²	0	R/W	IRQnR 0 : IRQn 端子に割り込み要求なし 1 : IRQn 端子に割り込み要求あり 【記号説明】 n=0~4、7

【注】 *1 ビット 6~5 は予約ビットです。読み出し時には不定が読み出されます。書き込むときは常に 0 を書き込んでください。1 を書いた場合の動作の保証はできません。

*2 リセット後の初期化ルーチンで割り込みコントロールレジスタ 1 (ICR1) の IRQE ビットを 0 に設定した後、IRQ7R、IRQ4R~IRQ0R ビットを 0 クリアしてください。

8.3.6 割り込みマスクレジスタ 0、1、4~6、9 (IMR0、1、4~6、9)

IMR0、1、4~6、9 は、読み出し/書き込み可能な 8 ビットのレジスタで、IRQ および内蔵周辺モジュール割り込み要求をマスク設定します。割り込み要因をマスク設定する際、IRQ および周辺モジュールの動作状態によっては割り込み要求を誤検出する場合があります。これを避けるため、割り込みが発生しない状態で割り込みマスクレジスタ (IMR0、1、4~6、9) を設定し、かつ設定後に読み出しを実行してください。

表 8.3 に本レジスタと各割り込み要因の関係を示します。

8. 割り込みコントローラ (INTC)

ビット	ビット名	初期値	R/W	説明									
7	IM7	0	R/W	割り込みマスク 割り込み要因と割り込みマスクレジスタの対応関係を表 8.3 に示します。 <table border="1" style="margin: 10px auto;"> <thead> <tr> <th>IMn</th> <th>読み出し時</th> <th>書き込み時</th> </tr> </thead> <tbody> <tr> <td>1</td> <td colspan="2">該当ビットの割り込み要因をマスクする</td> </tr> <tr> <td>0</td> <td>マスクしない</td> <td>なにもしない</td> </tr> </tbody> </table> 【記号説明】 n = 7 ~ 0	IMn	読み出し時	書き込み時	1	該当ビットの割り込み要因をマスクする		0	マスクしない	なにもしない
IMn	読み出し時	書き込み時											
1	該当ビットの割り込み要因をマスクする												
0	マスクしない	なにもしない											
6	IM6	0	R/W										
5	IM5	0	R/W										
4	IM4	0	R/W										
3	IM3	0	R/W										
2	IM2	0	R/W										
1	IM1	0	R/W										
0	IM0	0	R/W										

表 8.3 割り込み要因と IMR0 ~ IMR9 および IMCR0 ~ IMCR9 の対応

レジスタ名	ビット名 (機能名)							
	7	6	5	4	3	2	1	0
IMR0/IMCR0	IRQ7 (USBFS スタンバイ 復帰割り込み 要求)			IRQ4	IRQ3	IRQ2	IRQ1(BT)	IRQ0
IMR1/IMCR1					DEI3	DEI2	DEI1	DEI0
					(DMAC)			
IMR4/IMCR4		TUNI2	TUNI1	TUNI0	ITI			
		(TMU2)		(TMU1)	(TMU0)	(WDT)		
IMR5/IMCR5							SCIF1	SCIF0
							(SCIF1)	(SCIF0)
IMR6/IMCR6							SIOFI	
							(SIOF)	
IMR9/IMCR9						USBFI1	USBFI0	USBHI
						(USBF)		(USBH)

【注】 - 予約ビット：読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

8.3.7 割り込みマスククリアレジスタ 0、1、4~6、9 (IMCR0、1、4~6、9)

IMCR0、1、4~6、9 は、書き込み可能な 8 ビットのレジスタで、IRQ および内蔵周辺モジュール割り込み要求のマスク設定をクリアします。表 8.3 に本レジスタと各割り込み要因の関係を示します。

ビット	ビット名	初期値	R/W	説明						
7	IMC7		W	割り込みマスククリア 割り込み要因と割り込みマスククリアレジスタの対応関係を表 8.3 に示します。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>IMCn</th> <th>書き込み時</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>割り込みマスクレジスタ IMRn の該当ビットをクリアする</td> </tr> <tr> <td>0</td> <td>なにもしない</td> </tr> </tbody> </table> 【記号説明】 n = 7 ~ 0	IMCn	書き込み時	1	割り込みマスクレジスタ IMRn の該当ビットをクリアする	0	なにもしない
IMCn	書き込み時									
1	割り込みマスクレジスタ IMRn の該当ビットをクリアする									
0	なにもしない									
6	IMC6		W							
5	IMC5		W							
4	IMC4		W							
3	IMC3		W							
2	IMC2		W							
1	IMC1		W							
0	IMC0		W							

8.4 割り込み要因

割り込み要因は、NMI、IRQ、内蔵周辺モジュールの 3 つに分類されます。各割り込みの優先順位は割り込み優先レベル値 (16 ~ 0) で表され、レベル 16 が最高で、レベル 1 が最低です。レベル 0 に設定すると、その割り込みはマスクされ、割り込み要求は無視されます。

8.4.1 NMI 割り込み

NMI 割り込みは、レベル 16 の最優先の割り込みです。ステータスレジスタ (SR) の BL ビットが 0 ならば NMI 割り込みは受け付けられます。NMI 割り込みはエッジ検出です。ただし、スリープまたはスタンバイモード中は、BL ビットに関わりなく割り込みが受け付けられます。割り込みコントロールレジスタ 0 (ICR0) の NMI エッジセレクトビット (NMIE) は、立ち上がりエッジ / 立ち下がりエッジの選択を行います。

エッジ入力割り込み検出では、周辺クロック (P) ベースで 2 サイクル以上のパルス幅が必要です。NMI 割り込み例外処理によって、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) が影響されることはありません。

NMI 割り込みを使用して、スリープ、スタンバイ状態から復帰できます。

8.4.2 IRQ 割り込み

IRQ 割り込みは、IRQ7、IRQ4 ~ IRQ0 端子からレベルまたはエッジで入力されます。優先レベルは、割り込み優先レベル設定レジスタ C、D (IPRC、IPRD) にレベル 0 ~ 15 の範囲で設定できます。

IRQ 割り込みをエッジセンスで使用する場合は、IRR0 の対応するビットをソフトウェアで 1 であることを読み出した後に 0 を書き込み、割り込み要因をクリアしてください。

ICR1、ICR2 に上書きする際、IRQ 端子の状態によっては IRQ 割り込みが誤検出される可能性があります。これを避けるため、まず接続先を停止するか、接続先で信号マスクをかけ IRQ 端子の状態を非アクティブ側としておき、変化しないようにしてください。それから、割り込みマスクレジスタ IMR0 の設定、ステータスレジスタ SR のブロック (BL) ビットの設定等を実行して割り込みをマスク状態にしてから、割り込み要求レジスタ 0 (IRR0) を読み出した後に 0 を書き込んで不正な割り込みをクリアしたうえで、ICR1、ICR2 を上書きしてください。その後に接続先の起動やマスクの解除を順次してください。

8. 割り込みコントローラ (INTC)

エッジ入力割り込み検出では、周辺クロック (P) ベースで2 サイクル以上のパルス幅を必要とします。

IRQ 割り込みをレベルセンスで使用する場合は、CPU がサンプリングするまで端子レベルを保持する必要があります。したがって、割り込みハンドラ内で割り込み要因をクリアしてください。

IRQ 割り込み処理ルーチンによって、ステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) が影響されることはありません。

8.4.3 内蔵周辺モジュール割り込み

内蔵周辺モジュール割り込みは、次のようなモジュールで発生する割り込みです*。

- DMAコントローラ (DMAC)
- FIFO付きシリアルインタフェース (SIOF)
- FIFO内蔵シリアルコミュニケーションインタフェース (SCIF0、SCIF1)
- USBホストコントローラ (USBH)
- USBファンクションコントローラ (USBF)
- タイムユニット (TMU)
- ウォッチドッグタイマ (WDT)
- ユーザデバッグインタフェース (H-UDI)

【注】 * ブルートゥースインタフェース (BT) で発生する割り込みは内蔵周辺モジュール割り込みとしてではなく、外部割り込み (IRQ1) の端子に本 LSI 内部で接続されています。

また、USB ファンクションコントローラ (USBF) のスタンバイ復帰割り込みは、内蔵周辺モジュール割り込みとしてではなく、外部割り込み (IRQ7) の端子に本 LSI 内部で接続されています。

割り込み要因ごとに異なる割り込みベクタが割り当てられてはいませんが、要因は割り込み事象レジスタ 2 (INTEVT2) に反映されますので、INTEVT2 レジスタの値をオフセットとして分岐することにより、容易に要因を判定できます。

H-UDI を除く各モジュールの優先順位レベル (0 ~ 15) は、割り込み優先レベル設定レジスタ A ~ H (IPRA ~ IPRH) に優先順位レベル値を書き込むことによって、セットすることができます。H-UDI の優先順位レベルは 15 (固定) です。

ステータスレジスタの割り込みマスクビット (I3 ~ I0) は内蔵周辺モジュール割り込み処理の影響を受けません。

8.4.4 割り込み例外処理および優先順位

割り込み要因は、NMI、IRQ、内蔵周辺モジュールの 3 つに分類されます。各割り込みの優先順位は割り込み優先レベル値 (16 ~ 0) で表され、レベル 16 が最高で、レベル 1 が最低です。レベル 0 に設定すると、その割り込みはマスクされ、割り込み要求は無視されます。

割り込み要因と割り込み事象レジスタ 2 (INTEVT2) の例外コード、割り込み優先順位を表 8.4 に示します。

各割り込み要因に応じて割り込み事象レジスタ 2 (INTEVT2) に、それぞれ異なる例外コードが割り当てられ

ます。例外処理ルーチンの先頭アドレスは、各割り込み要因で共通です。このため、割り込み要因を識別するために、例外処理ルーチンの先頭で、INTEVT2 レジスタの値を使って分岐させます。たとえば INTEVT2 レジスタの値をオフセットにして分岐させます。

内蔵周辺モジュール、IRQ 割り込みの優先順位は、割り込み優先レベル設定レジスタによって、優先レベル 15 ~ 0 の範囲で任意に設定できます。リセットによって、内蔵周辺モジュール、IRQ 割り込みの優先順位は優先レベル 0 に設定されます。

複数の割り込み要因の優先順位を同じレベルに設定した場合、それらの割り込みが同時に発生したときは、表 8.4 に示す「デフォルト優先順位」に従って処理されます。

表 8.4 割り込み例外処理要因と優先順位

割り込み要因		例外コード	割り込み 優先順位 (初期値)	IPR (ビット番号)	IPR 設定 ユニット内の 優先順位	デフォルト 優先順位
NMI		H'1C0	16	-	-	高 ↑ ↓ 低
H-UDI		H'5E0	15	-	-	
IRQ	IRQ0	H'600	0-15 (0)	IPRC (3-0)	-	
	IRQ1 (BT)	H'620	0-15 (0)	IPRC (7-4)	-	
	IRQ2	H'640	0-15 (0)	IPRC (11-8)	-	
	IRQ3	H'660	0-15 (0)	IPRC (15-12)	-	
	IRQ4	H'680	0-15 (0)	IPRD (3-0)	-	
	IRQ7 (USBF スタンバイ復帰)	H'6E0	0-15 (0)	IPRD (15-12)	-	
DMAC	DEI0	H'800	0-15 (0)	IPRE (15-12)	高 ↑ ↓ 低	
	DEI1	H'820				
	DEI2	H'840				
	DEI3	H'860				
USBH	USBHI	H'A00	0-15 (0)	IPRF (3-0)	-	
USBF	USI0	H'A20	0-15 (0)	IPRF (7-4)	高 低	
	USI1	H'A40				
SCIF0	SCIFI0	H'C00	0-15 (0)	IPRG (15-12)	-	
SCIF1	SCIFI1	H'C20	0-15 (0)	IPRG (11-8)	-	
SIOF	SIOFI	H'CA0	0-15 (0)	IPRH (11-8)	-	
TMU0	TUNI0	H'400	0-15 (0)	IPRA (15-12)	-	
TMU1	TUNI1	H'420	0-15 (0)	IPRA (11-8)	-	
TMU2	TUNI2	H'440	0-15 (0)	IPRA (7-4)	-	
WDT	ITI	H'560	0-15 (0)	IPRB (15-12)	-	

8.5 動作説明

8.5.1 割り込み動作の流れ

割り込み発生時の動作の流れを以下に説明します。また、図 8.2 に動作フローを示します。

1. 割り込みコントローラに対して、各割り込み要求元から割り込み要求信号が送られます。
2. 割り込みコントローラでは、送られた割り込み要求の中から割り込み優先レベルレジスタ A ~ H (IPRA ~ IPRH) に従って、最も優先順位の高い割り込みが選択され、それより優先順位の低い割り込みは保留されます。このとき、同一優先順位に設定された割り込み、または同一モジュール内の割り込みが複数発生した場合は、表 8.4 に従って、最も優先順位の高い割り込みが選択されます。
3. 割り込みコントローラで選択された割り込みの優先レベルと CPU のステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) とが比較されます。I3 ~ I0 ビットのレベルより高い優先順位の割り込みだけが受け付けられ、CPU へ割り込み要求信号が送られます。
4. 検出タイミング : INTC は周辺クロック (P) に同期して動作し、CPU に割り込み要求を通知します。CPU は、命令の切れ目で割り込みを受け付けます。
5. 割り込み事象レジスタ 2 (INTEVT2) に割り込み要因コードがセットされます。
6. ステータスレジスタ (SR) とプログラムカウンタ (PC) が、それぞれ SSR と SPC に退避されます。
7. SR のブロックビット (BL)、レジスタバンクビット (RB)、モードビット (MD) が 1 にセットされます。
8. 割り込み関連の例外処理ルーチンの先頭番地 (ベクタベースレジスタ (VBR) に設定された値と H'00000600 の和) にジャンプします。このジャンプは遅延分岐ではありません。

例外処理ルーチンでは、割り込み要因を識別するために、たとえば INTEVT2 レジスタの値をオフセットとして分岐します。これにより、容易に割り込み要因別の処理ルーチンへ分岐できます。

- 【注】
1. 本 LSI では、割り込みを受け付けても CPU のステータスレジスタ (SR) の割り込みマスクビット (I3 ~ I0) は変化しません。
 2. 割り込み要因フラグは、割り込みハンドラ中でクリアしてください。
クリアしたはずの割り込み要因を誤まって再度受け付けないようにするために、クリア後要因フラグをリードし、その後 RTE 命令を実行します。

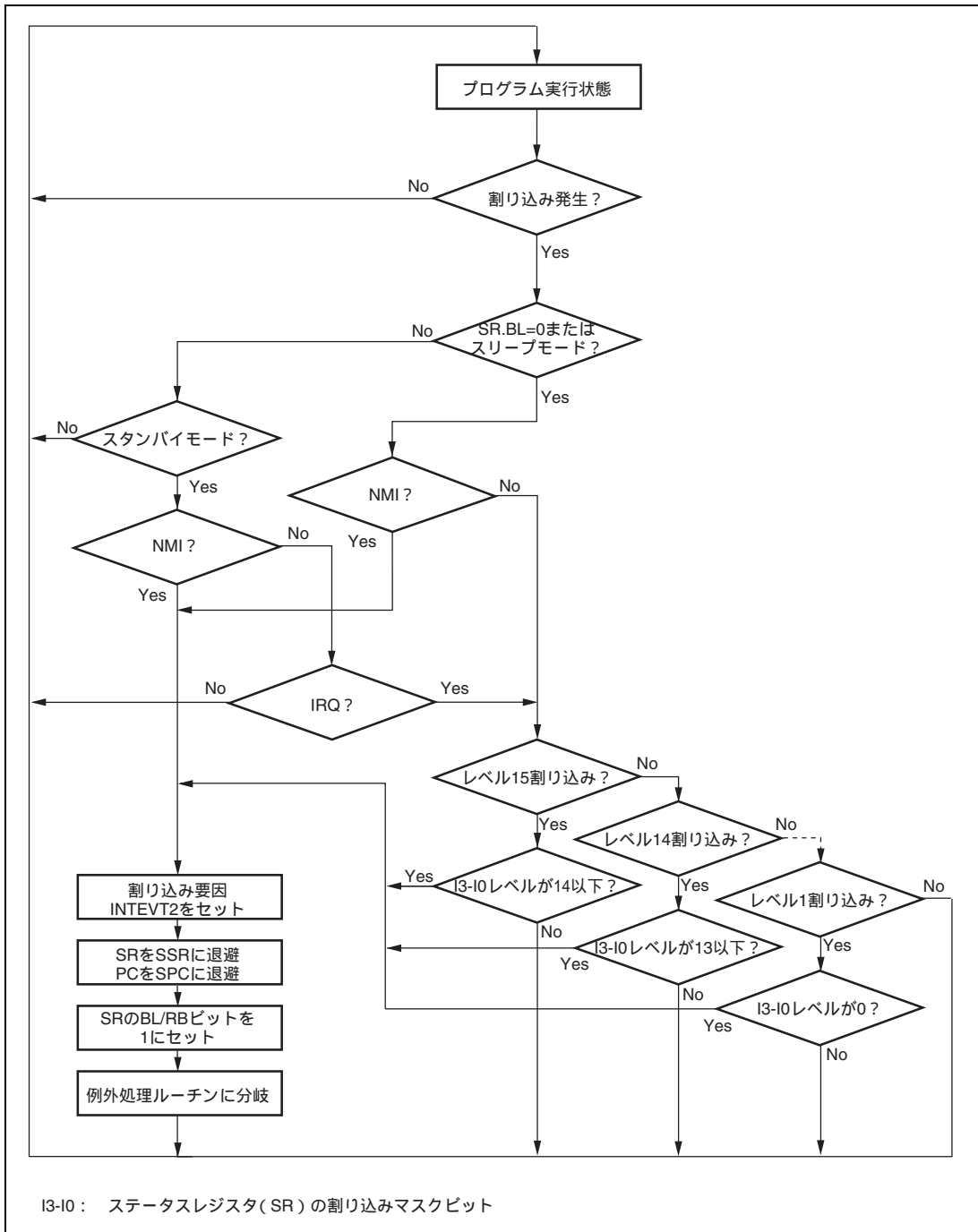


図 8.2 割り込み動作フローチャート

8.5.2 多重割り込み

多重割り込みを行う場合は、割り込み処理ルーチンの構造を以下のようにします。

1. 割り込み要因を判定するために、INTEVT2レジスタの値をオフセットとして、各割り込み要因の割り込み処理ルーチンに分岐します。
2. 各割り込み処理ルーチン中で、該当割り込み要因をクリアします。
3. SPC、SSRをメモリに退避します。
4. SRのBLビットをクリアします。このとき、SRの割り込みマスクビットも受け付けた割り込みレベルに設定します。
5. このあと、実際に行いたい処理を書きます。
6. RTE命令を実行します。

割り込み処理ルーチンを上記の構造にすることにより、4. の直後の時点で多重割り込みがあった場合、より優先レベルの高いものは受け付けられます。

9. バスステートコントローラ (BSC)

本 LSI に内蔵のバスステートコントローラ (BSC) は、外部アドレス空間に接続された各種メモリ、外部デバイスに対し制御信号を出力します。これにより、SRAM、SDRAM などの各種メモリ、および外部デバイスを直接接続することができます。

9.1 特長

BSC には、次の特長があります。

1. 外部アドレス空間

- CS0、CS3、およびCS4の各空間をそれぞれ最大16Mバイトまで、および外部アドレス空間の合計を48Mバイトまでサポート
- 空間ごとに、通常空間インタフェース、バイト選択付きSRAMインタフェース、バーストROMインタフェース、およびSDRAMインタフェースのメモリ種類を指定可能*¹
- 空間ごとに、データバス幅 (8ビットまたは16ビット) を選択可能*²
- 空間ごとに、ウェイトステートの挿入を制御可能
- リードアクセス、ライトアクセスごとにウェイトステートの挿入を制御可能
- 連続するアクセスがリード - ライト (同一空間または別空間)、リード - リード (同一空間または別空間)、および先頭サイクルがライトの場合の5種類独立にアイドルサイクルを設定可能

2. 通常空間インタフェース

- 通常メモリ (SRAM、ROM) 等との直結が可能なインタフェースをサポート

3. バーストROMインタフェース

ページモード機能を有するROM (ページFlash ROM) を高速にアクセス可能

4. SDRAMインタフェース

- CS3空間にSDRAMを設定可能
- ロウアドレスまたはカラムアドレスのマルチプレクス出力をサポート
- シングル読み出しまたはシングル書き込みによる効率的なアクセスが可能
- バンクアクティブモードによる高速アクセスが可能
- オートリフレッシュとセルフリフレッシュのサポート

5. バイト選択付きSRAMインタフェース

- バイト選択付きSRAMとの直結が可能なインタフェースをサポート

9. バスステートコントローラ (BSC)

6. バスアービトレーション

- すべての資源を他のCPUと共有し、外部からのバス権要求を受け、バス使用許可を出力可能

7. リフレッシュ機能

- オートリフレッシュとセルフリフレッシュをサポート
- リフレッシュ用カウンタ、クロック選択により、リフレッシュ間隔を設定可能
- リフレッシュ回数設定 (1、2、4、6、および8) による集中リフレッシュが可能

【注】 *1 CSn 空間によっては、一部のメモリ種類のインタフェースをサポートしていません。空間ごとにサポートしているメモリインタフェースの種類については、「9.3 エリアの概要」を参照してください。

*2 CS0 空間のデータバス幅は 16 ビット固定です。

BSCのブロック図を図9.1に示します。

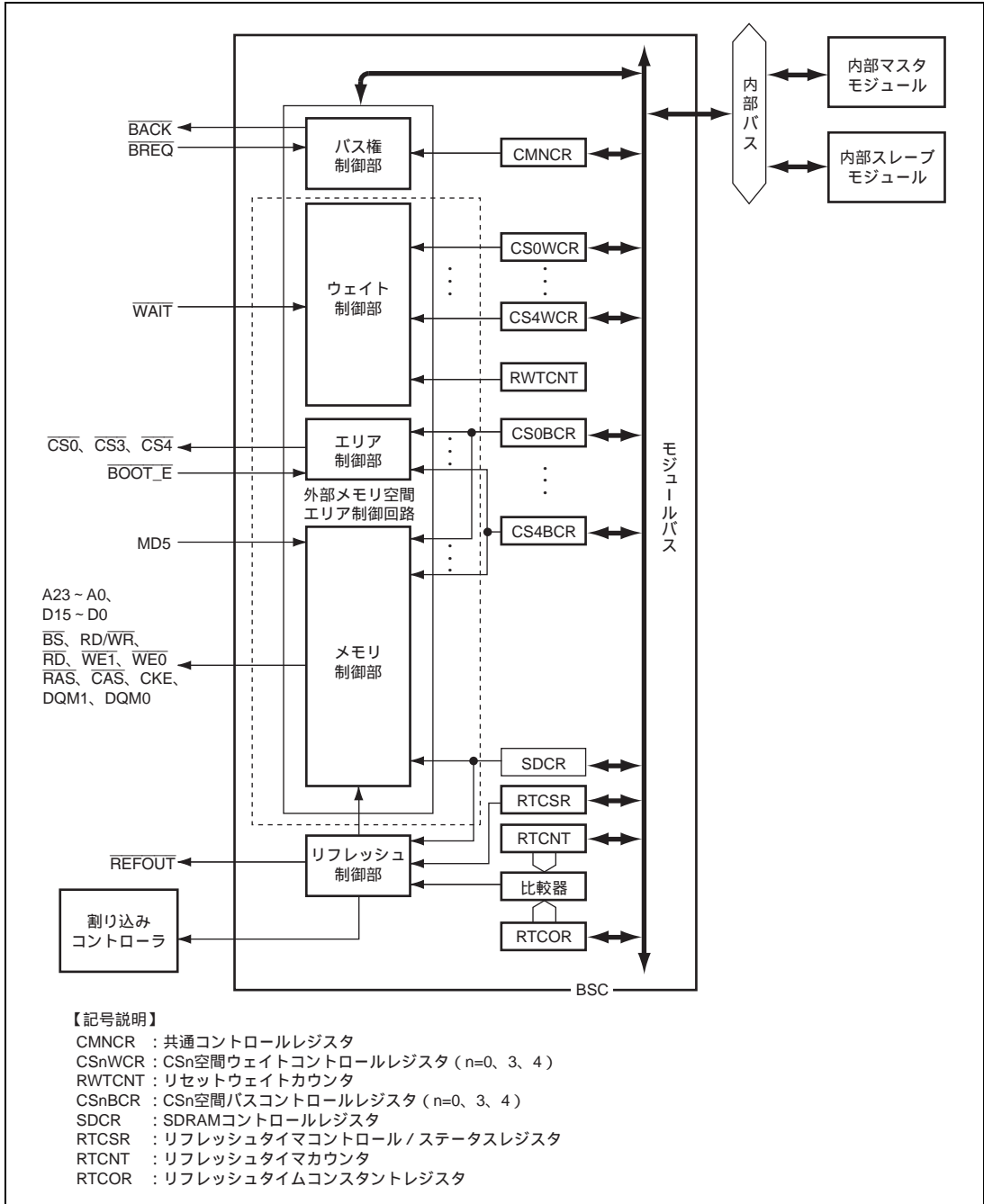


図9.1 BSCのブロック図

9. バスステートコントローラ (BSC)

9.2 入出力端子

BSC の端子構成を表 9.1 に示します。

表 9.1 端子構成

端子名	入出力	機能
A23 ~ A0	出力	アドレスバス
D15 ~ D0	入出力	データバス
\overline{BS}	出力	バスサイクルの開始を示す信号 通常空間およびバースト ROM アクセス時にアサートされる。SDRAM アクセス時は、 \overline{CAS} と同タイミングでアサートされる。
$\overline{CS0}$, $\overline{CS3}$, $\overline{CS4}$	出力	チップセレクト
$\overline{RD}/\overline{WR}$	出力	リードまたはライト信号 SDRAM およびバイト選択付き SRAM 接続時は、 \overline{WE} 端子に接続
\overline{RD}	出力	リードパルス信号 (リードデータ出力許可信号)
$\overline{WE1}/\overline{DQM1}$	出力	$\overline{WE1}$: D15 ~ D8 対応のバイト書き込み指示 バイト選択付き SRAM 接続時は、バイトセレクト端子に接続 $\overline{DQM1}$: D15 ~ D8 対応の選択信号 SDRAM 接続時は、 $\overline{DQM1}$ 端子に接続
$\overline{WE0}/\overline{DQM0}$	出力	$\overline{WE0}$: D7 ~ D0 対応のバイト書き込み指示 バイト選択付き SRAM 接続時は、バイトセレクト端子に接続 $\overline{DQM0}$: D7 ~ D0 対応の選択信号 SDRAM 接続時は、 $\overline{DQM0}$ 端子に接続
\overline{RAS}	出力	SDRAM 接続時は、 \overline{RAS} 端子に接続
\overline{CAS}	出力	SDRAM 接続時は、 \overline{CAS} 端子に接続
\overline{CKE}	出力	SDRAM 接続時は、 \overline{CKE} 端子に接続
\overline{WAIT}	入力	外部ウェイト入力
\overline{BREQ}	入力	バス権要求入力
\overline{BACK}	出力	バス使用許可出力
MD5	入力	エンディアン指定 0: ビッグエンディアン 1: リトルエンディアン
\overline{REFOUT}	出力	バス解放時リフレッシュ実行要求出力
$\overline{BOOT_E}$	入力	ブートイネーブル (詳細は、「第 17 章 ブート機能 (BOOT)」を参照)

9.3 エリアの概要

9.3.1 空間分割

本 LSI は、アーキテクチャとして 32 ビットの論理アドレス空間を有しています。このうち上位 3 ビットでキャッシュアクセス方法を表します (詳細は、「第 5 章 キャッシュ」を参照してください)。残り 29 ビットは、512M バイトの物理アドレス空間にマッピングされますが、この物理アドレス空間はさらに 8 つのエリアに分割されています。BSC は、この 29 ビットの物理アドレス空間に対し制御を行います。図 9.2 に論理アドレス空間から物理アドレス空間へのマッピングを示します。エリア 1 は内蔵 I/O 空間として使用され、エリア 2 およびエリア 5~7 は予約領域です。残りの 3 つのエリア (エリア 0、エリア 3、エリア 4) が外部アドレス空間になります。

本 LSI は、表 9.2 に示すように 8 つのエリアに分割された物理アドレス空間のうち、外部アドレス空間の 3 つのエリアにそれぞれ各種メモリを接続でき、各々に対応してチップセレクト信号 ($\overline{CS0}$ 、 $\overline{CS3}$ 、 $\overline{CS4}$) を出力します。たとえば、エリア 0 のアクセス時には $\overline{CS0}$ が、エリア 4 のアクセス時に $\overline{CS4}$ がアサートされます。

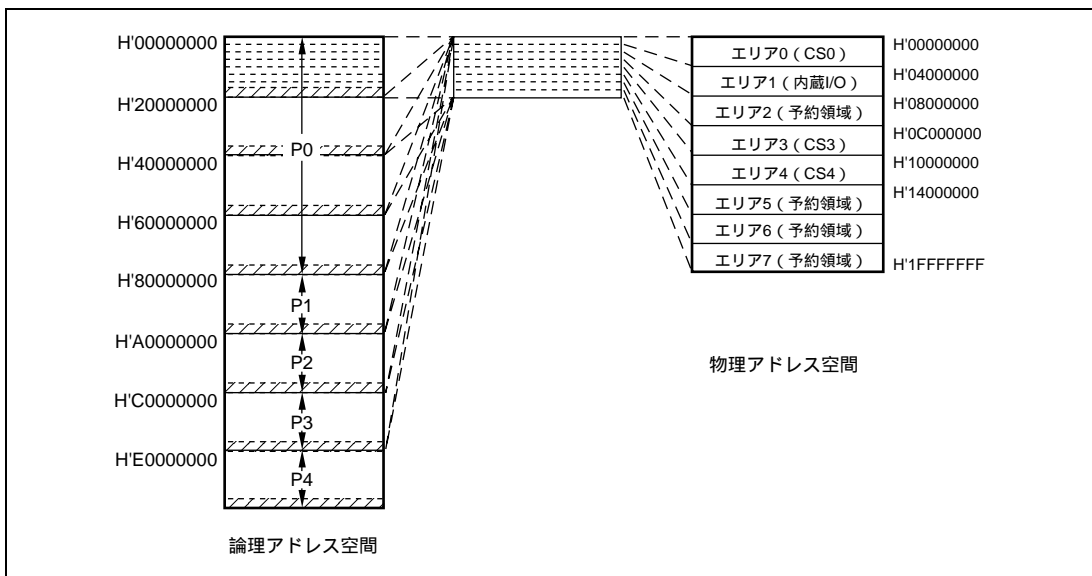


図 9.2 論理アドレス空間と物理アドレス空間

9.3.2 シャドウ空間

エリア 0、3 および 4 は、物理アドレスの A28 ~ A26 でデコードされ、A28 ~ A26 = B'000、B'011 および B'100 のエリアに対応します。アドレスの A31 ~ A29 は、無視されます。このため、たとえば、エリア 0 のアドレスの範囲は H'00000000 ~ H'03FFFFFF なのに対し、H'20000000 × n (n=1 ~ 6) を加えた P0 ~ P3 領域の論理アドレス空間はシャドウ空間となります。さらに、アドレスバスの外部端子としては、A23 ~ A0 をサポートしているので、アドレス A25 ~ A24 は、無視されます。このため、たとえば、エリア 0 では、H'01000000 ~ H'03FFFFFF もシャドウ

9. バスステートコントローラ (BSC)

空間となります。

P4 領域 (H'E0000000 ~ H'FFFFFFF) は、内蔵 I/O 領域であり、内蔵レジスタ等のアドレスが割り付けてあります。シャドウ空間にはなりません。

9.3.3 アドレスマップ

外部アドレス空間は合計 48M バイトあり、これを 3 つのエリアに分割して使用します。接続されるメモリの種類およびデータバス幅は、各部分空間ごとに指定します。外部アドレス空間のアドレスマップは、表 9.2 のとおりです。

表 9.2 外部アドレス空間のアドレスマップ

空間	接続可能なメモリの種類	物理アドレス	容量	アクセスサイズ
エリア 0 (CS0 空間)	通常空間*1	H'00000000 ~ H'00FFFFFF	16M バイト	8、16、32*2
	バースト ROM	H'01000000 ~ H'03FFFFFF	シャドウ	
		H'00000000 ~ H'03FFFFFF + H'20000000 × n + H'20000000 × n	シャドウ	(n : 1~6)
エリア 1	内蔵 I/O	H'04000000 ~ H'07FFFFFF		
		H'04000000 ~ H'07FFFFFF + H'20000000 × n + H'20000000 × n		(n : 1~6)
エリア 2	予約領域*4	H'08000000 ~ H'0BFFFFFF + H'20000000 × n + H'20000000 × n		(n : 1~6)
エリア 3 (CS3 空間)	通常空間*1	H'0C000000 ~ H'0CFFFFFF	16M バイト	8、16、32*3
	バイト選択付き SRAM SDRAM	H'0D000000 ~ H'0FFFFFFF	シャドウ	
		H'0C000000 ~ H'0FFFFFFF + H'20000000 × n + H'20000000 × n	シャドウ	(n : 1~6)
エリア 4 (CS4 空間)	通常空間*1	H'10000000 ~ H'10FFFFFF	16M バイト	8、16、32*3
	バースト ROM バイト選択付き SRAM	H'11000000 ~ H'13FFFFFF	シャドウ	
		H'10000000 ~ H'13FFFFFF + H'20000000 × n + H'20000000 × n	シャドウ	(n : 1~6)
エリア 5	予約領域*4	H'14000000 ~ H'17FFFFFF + H'20000000 × n + H'20000000 × n		(n : 1~6)
エリア 6	予約領域*4	H'18000000 ~ H'1BFFFFFF + H'20000000 × n + H'20000000 × n		(n : 0~6)
エリア 7	予約領域*4	H'1C000000 ~ H'1FFFFFFF + H'20000000 × n + H'20000000 × n		(n : 0~7)

【注】 *1 通常空間は、SRAM、ROM 等のインタフェースを持つ外部デバイスを接続できます。

*2 エリア 0 の空間 (チップセレクト信号 $\overline{CS0}$ アサート時にセレクト) はデータバス幅 16 ビット固定です。

*3 CSn 空間バスコントロールレジスタ (CSnBCR) でデータバス幅を指定します。

エリア 3 の空間で SDRAM 接続時のデータバス幅は 16 ビットに設定してください。

*4 予約領域はアクセスしないでください。アクセスした場合は、動作の保証はできません。

9.3.4 データバス幅

本 LSI のエリア 3 およびエリア 4 のデータバス幅は、CS_n 空間バスコントロールレジスタ (CS_nBCR, n=3~4) にて空間ごとに 8 ビット、16 ビットから選べます。エリア 0 のデータバス幅は、16 ビットに固定されています。

詳しくは「9.4.2 CS_n 空間バスコントロールレジスタ (CS_nBCR)」を参照してください。

9.4 レジスタの説明

BSC には、以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については、「第 27 章 レジスタ一覧」を参照してください。

メモリとのインタフェースの設定が終了するまでは、CS0 空間以外はアクセスしないでください。CS0 空間以外にアクセスする場合、アクセス対象となる空間に対応する CS_n 空間バスコントロールレジスタ (CS_nBCR) を先に設定し、その後で CS_n 空間ウェイトコントロールレジスタ (CS_nWCR) を設定してください。

- 共通コントロールレジスタ (CMNCR)
- CS0空間バスコントロールレジスタ (CS0BCR)
- CS3空間バスコントロールレジスタ (CS3BCR)
- CS4空間バスコントロールレジスタ (CS4BCR)
- CS0空間ウェイトコントロールレジスタ (CS0WCR)
- CS3空間ウェイトコントロールレジスタ (CS3WCR)
- CS4空間ウェイトコントロールレジスタ (CS4WCR)
- SDRAMコントロールレジスタ (SDCR)
- リフレッシュタイムコントロール/ステータスレジスタ (RTCSR)
- リフレッシュタイムカウンタ (RTCNT)
- リフレッシュタイムコンスタントレジスタ (RTCOR)
- リセットウェイトカウンタ (RWCNT)
- CS3空間SDRAMモードレジスタ (SDMR3)*

【注】 * 本レジスタの実体は SDRAM 内にあります。本レジスタ空間をアクセスすることにより、SDRAM 内のレジスタに書き込まれます。詳しくは、「9.5.5 (12) パワーオンシ - ケンス」を参照してください。

9. バスステートコントローラ (BSC)

9.4.1 共通コントロールレジスタ (CMNCR)

CMNCR は、各エリアに共通の制御を行う 32 ビットのレジスタです。

CMNCR は、パワーオンリセット時に初期化されますが、マニュアルリセットおよびスタンバイモード時は初期化されずに前の値を保持します。本レジスタの初期設定が終了するまではエリア 0 以外の外部メモリをアクセスしないでください。

ビット	ビット名	初期値	R/W	説明
31~8		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
7 6	DMAIW1 DMAIW0	0 0	R/W R/W	DMA シングルアドレス転送時のアクセスサイクル間ウェイト指定 DMA シングルアドレス転送時に DACK 付き外部デバイスをアクセスした後に挿入するアイドルサイクル数を指定します。 アイドルサイクルの挿入の方法は、後述の DMAIWA ビットの指定により異なります。 00 : アイドルサイクルなし 01 : 1 アイドルサイクル挿入 10 : 2 アイドルサイクル挿入 11 : 4 アイドルサイクル挿入
5	DMAIWA	0	R/W	DMA シングルアドレス転送時のアクセスサイクル間ウェイト挿入方法指定 DMAIW1 と DMAIW0 ビットで指定したアイドルサイクルの挿入方法を指定します。 本ビットが 0 の場合は、DACK 付き外部デバイスがデータバスをドライブ後、本 LSI を含む他のデバイスがデータバスをドライブするときにアイドルサイクルを挿入します。DACK 付き外部デバイスが連続してデータバスをドライブする場合は、アイドルサイクルを挿入しません。本ビットが 1 の場合は、DACK 付き外部デバイスへのアクセスが連続する場合でも、1 回のアクセス終了後必ずアイドルサイクルが挿入されます。 0 : DACK 付き外部デバイスがデータバスをドライブ後、他のデバイスがデータバスをドライブするときにアイドルサイクルを挿入 1 : DACK 付き外部デバイスアクセス後、常にアイドルサイクルを挿入
4		1	R	リザーブビット 読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。 0 を書いた場合の動作の保証はできません。

ビット	ビット名	初期値	R/W	説明
3	ENDIAN	0/1*	R	エンディアンフラグ 本ビットは、パワーオンリセット時にエンディアン設定の外部端子 (MD5) の値をサンプリングし、その値を初期値とします。 全空間のエンディアンは、このビットで決定されます。本ビットは、読み出し専用です。 0: パワーオンリセット時にエンディアン設定の外部端子 (MD5) がローレベルであり、本 LSI がビッグエンディアンとして動作します。 1: パワーオンリセット時にエンディアン設定の外部端子 (MD5) がハイレベルであり、本 LSI がリトルエンディアンとして動作します。
2		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
1	HIZMEM	0	R/W	Hi-Z メモリコントロール A23 ~ A0、 \overline{BS} 、 \overline{CSn} 、 $\overline{RD}/\overline{WR}$ 、 $\overline{WEn}/\overline{DQMn}$ 、および \overline{RD} のスタンバイモード時の端子状態を指定します。 0: スタンバイモード時にハイインピーダンス 1: スタンバイモード時にドライブ
0	HIZCNT	0	R/W	Hi-Z コントロール スタンバイモード時およびバス権解放時の CKIO、CKE、 \overline{RAS} 、 \overline{CAS} の状態を指定します。 0: CKIO、CKE、 \overline{RAS} 、 \overline{CAS} は、スタンバイモード時およびバス権解放時にハイインピーダンス 1: CKIO、CKE、 \overline{RAS} 、 \overline{CAS} は、スタンバイモード時およびバス権解放時にドライブ

【注】 * エンディアンを指定する外部端子 (MD5) の値をパワーオンリセット時にサンプリングします。
ビッグエンディアンのときは 0、リトルエンディアンのときは 1 となります。

9.4.2 CSn 空間バスコントロールレジスタ (CSnBCR) (n = 0、3、4)

CSnBCR は、各空間に接続するメモリの種類、空間のデータバス幅、およびアクセスサイクル間ウェイト数を設定する 32 ビットのレジスタです。

CSnBCR はパワーオンリセット時に初期化されますが、マニュアルリセットおよびスタンバイモード時は初期化されずに前の値を保持します。本レジスタの初期設定が終了するまでは、エリア 0 以外の外部メモリをアクセスしないでください。

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
31、30		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
29 28	IWW1 IWW0	1 1	R/W R/W	ライト - リード / ライト - ライトサイクル間アイドル指定* ¹ 空間に接続されたメモリをアクセスした後に挿入するアイドルサイクル数を指定します。 対象となるサイクルは、ライト - リードサイクルとライト - ライトサイクルの場合です。 00 : アイドルサイクルなし 01 : 1 アイドルサイクル挿入 10 : 2 アイドルサイクル挿入 11 : 4 アイドルサイクル挿入
27		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
26 25	IWRWD1 IWRWD0	1 1	R/W R/W	別空間リード - ライトサイクル間アイドル指定* ¹ 空間に接続されたメモリをアクセスした後に挿入するアイドルサイクル数を指定します。 対象となるサイクルは、連続するアクセスが本空間リード - 別空間ライトサイクルの場合です。 00 : アイドルサイクルなし 01 : 1 アイドルサイクル挿入 10 : 2 アイドルサイクル挿入 11 : 4 アイドルサイクル挿入
24		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
23 22	IWRWS1 IWRWS0	1 1	R/W R/W	同一空間リード - ライトサイクル間アイドル指定* ¹ 空間に接続されたメモリをアクセスした後に挿入するアイドルサイクル数を指定します。 対象となるサイクルは、連続するアクセスが本空間リード - 同一空間ライトサイクルの場合です。 00 : アイドルサイクルなし 01 : 1 アイドルサイクル挿入 10 : 2 アイドルサイクル挿入 11 : 4 アイドルサイクル挿入
21		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
20 19	IWRRD1 IWRRD0	1 1	R/W R/W	別空間リード - リードサイクル間アイドル指定*1 空間に接続されたメモリをアクセスした後に挿入するアイドルサイクル数を指定します。 対象となるサイクルは、連続するアクセスが本空間リード - 別空間リードサイクルの場合です。 00 : アイドルサイクルなし 01 : 1 アイドルサイクル挿入 10 : 2 アイドルサイクル挿入 11 : 4 アイドルサイクル挿入
18		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
17 16	IWRRS1 IWRRS0	1 1	R/W R/W	同一空間リード - リードサイクル間アイドル指定 空間に接続されたメモリをアクセスした後に挿入するアイドルサイクル数を指定します。 対象となるサイクルは、連続するアクセスが本空間リード - 同一空間リードサイクルの場合です。 00 : アイドルサイクルなし 01 : 1 アイドルサイクル挿入 10 : 2 アイドルサイクル挿入 11 : 4 アイドルサイクル挿入
15		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
14 13 12	TYPE2 TYPE1 TYPE0	0 0 0	R/W R/W R/W	メモリ種類指定*2 空間に接続するメモリの種類を設定します。 000 : 通常空間 001 : パースト ROM 010 : 設定禁止 011 : バイト選択付き SRAM 100 : SDRAM 101 : 設定禁止 110 : 設定禁止 111 : 設定禁止 エリアごとの接続可能なメモリの種類は、表 9.2 を参照してください。
11		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
10	BSZ1	1	R/W	データバス幅指定*3
9	BSZ0	0/1	R/W	空間のデータバス幅を指定します。 エリア 3、エリア 4 の設定は以下のとおりです。 00 : 設定禁止 01 : 8 ビット 10 : 16 ビット 11 : 設定禁止 【注】CS0BCR の初期値は 10、他の CSnBCR の初期値は 11 になりますので、有効な値に設定してください。 エリア 0 のデータバス幅は、16 ビット固定です。本ビットの設定は無視されます。
8~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

- 【注】 *1 実際の動作では、本ビットの指定以外の要因により、アイドルサイクルが挿入される場合があります。
- *2 メモリ種類の指定で、指定可能なエリア以外に設定した場合、動作の保証はできません。
- *3 エリア 0 以外のデータバス幅は、リセット後の初期値は設定禁止状態となりますので、該当空間アクセス前に 8 ビットまたは 16 ビットに設定してください。

9.4.3 CSn 空間ウェイトコントロールレジスタ (CSnWCR) (n=0, 3, 4)

CSnWCR は、メモリアクセスに関する各種ウェイトサイクルの設定を行う、32 ビットのレジスタです。本レジスタのビット構成は、CSn 空間バスコントロールレジスタ (CSnBCR) に設定したメモリ種類により、以下のように変わります。対象となるエリアをアクセスする前に設定してください。本レジスタは、CSnBCR レジスタを設定後に設定してください。

CSnWCR は、パワーオンリセット時に初期化されますが、マニュアルリセットおよびスタンバイモード時は初期化されずに前の値を保持します。

(1) 通常空間、バイト選択付き SRAM の場合

• CS0WCR

ビット	ビット名	初期値	R/W	説明
31~13		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
12	SW1	0	R/W	アドレス、 \overline{CSn} アサート \overline{RD} 、 \overline{WEn} アサート遅延サイクル数 アドレス、 \overline{CSn} アサートから \overline{RD} 、 \overline{WEn} アサートまでの遅延サイクル数を指定します。 00 : 0.5 サイクル 01 : 1.5 サイクル 10 : 2.5 サイクル 11 : 3.5 サイクル
11	SW0	0	R/W	
10	WR3	1	R/W	アクセスウェイトサイクル数 リード/ライトアクセスに必要なウェイトサイクル数を指定します。 0000 : 0 サイクル 0001 : 1 サイクル 0010 : 2 サイクル 0011 : 3 サイクル 0100 : 4 サイクル 0101 : 5 サイクル 0110 : 6 サイクル 0111 : 8 サイクル 1000 : 10 サイクル 1001 : 12 サイクル 1010 : 14 サイクル 1011 : 18 サイクル 1100 : 24 サイクル 1101 : 設定禁止 1110 : 設定禁止 1111 : 設定禁止
9	WR2	0	R/W	
8	WR1	1	R/W	
7	WR0	0	R/W	
6	WM	0	R/W	外部ウェイトマスク指定 外部ウェイト入力を有効にするか無視するかを指定します。 アクセスウェイトサイクル数が 0 の場合でも、本ビットの設定は有効です。 0 : 外部ウェイト入力有効 1 : 外部ウェイト入力無視
5~2		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
1	HW1	0	R/W	\overline{RD} 、 \overline{WEn} ネゲート アドレス、 \overline{CSn} ネゲート遅延サイクル数 \overline{RD} 、 \overline{WEn} ネゲートから、アドレス、 \overline{CSn} ネゲートまでの遅延サイクル数を指定します。 00 : 0.5 サイクル 01 : 1.5 サイクル 10 : 2.5 サイクル 11 : 3.5 サイクル
0	HW0	0	R/W	

9. バスステートコントローラ (BSC)

• CS3WCR

ビット	ビット名	初期値	R/W	説明
31~21		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
20	BAS	0	R/W	バイト選択付き SRAM バイトアクセス選択 バイト選択付き SRAM インタフェース時の \overline{WEn} および RD/ \overline{WR} 信号のタイミングを設定します。 0: \overline{WEn} は、リード/ライトタイミングでアサート RD/ \overline{WR} は、ライトアクセスサイクル中アサート 1: \overline{WEn} はリード/ライトアクセスサイクル中アサート RD/ \overline{WR} は、ライトタイミングでアサート
19~11		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
10	WR3	1	R/W	アクセスウェイトサイクル数
9	WR2	0	R/W	リード/ライトアクセスに必要なウェイトサイクル数を指定します。
8	WR1	1	R/W	0000: 0 サイクル
7	WR0	0	R/W	0001: 1 サイクル 0010: 2 サイクル 0011: 3 サイクル 0100: 4 サイクル 0101: 5 サイクル 0110: 6 サイクル 0111: 8 サイクル 1000: 10 サイクル 1001: 12 サイクル 1010: 14 サイクル 1011: 18 サイクル 1100: 24 サイクル 1101: 設定禁止 1110: 設定禁止 1111: 設定禁止
6	WM	0	R/W	外部ウェイトマスク指定 外部ウェイト入力の有効にするか無視するかを指定します。 アクセスウェイトサイクル数が0の場合でも、本ビットの設定は有効です。 0: 外部ウェイト入力有効 1: 外部ウェイト入力無視
5~0		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。

9. バスステートコントローラ (BSC)

• CS4WCR

ビット	ビット名	初期値	R/W	説明
31~21		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
20	BAS	0	R/W	バイト選択付き SRAM バイトアクセス選択 バイト選択付き SRAM インタフェース時の $\overline{WE_n}$ および RD/\overline{WR} 信号のタイミングを設定します。 0: $\overline{WE_n}$ は、リード/ライトタイミングでアサート RD/\overline{WR} は、ライトアクセスサイクル中アサート 1: $\overline{WE_n}$ はリード/ライトアクセスサイクル中アサート RD/\overline{WR} は、ライトタイミングでアサート
19		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
18 17 16	WW2 WW1 WW0	0 0 0	R/W R/W R/W	ライトアクセスウェイトサイクル数 ライトアクセスに必要なサイクル数を指定します。 000: WR3~WR0 設定 (リードアクセスウェイト) と同じサイクル 001: 0 サイクル 010: 1 サイクル 011: 2 サイクル 100: 3 サイクル 101: 4 サイクル 110: 5 サイクル 111: 6 サイクル
15~13		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
12 11	SW1 SW0	0 0	R/W R/W	アドレス、 $\overline{CS_n}$ アサート \overline{RD} 、 $\overline{WE_n}$ アサート遅延サイクル数 アドレス、 $\overline{CS_n}$ アサートから \overline{RD} 、 $\overline{WE_n}$ アサートまでの遅延サイクル数を指定します。 00: 0.5 サイクル 01: 1.5 サイクル 10: 2.5 サイクル 11: 3.5 サイクル

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説 明
10	WR3	1	R/W	アクセスウェイトサイクル数
9	WR2	0	R/W	リード/ライトアクセスに必要なウェイトサイクル数を指定します。
8	WR1	1	R/W	0000 : 0 サイクル
7	WR0	0	R/W	0001 : 1 サイクル
				0010 : 2 サイクル
				0011 : 3 サイクル
				0100 : 4 サイクル
				0101 : 5 サイクル
				0110 : 6 サイクル
				0111 : 8 サイクル
				1000 : 10 サイクル
				1001 : 12 サイクル
				1010 : 14 サイクル
				1011 : 18 サイクル
				1100 : 24 サイクル
				1101 : 設定禁止
				1110 : 設定禁止
				1111 : 設定禁止
6	WM	0	R/W	外部ウェイトマスク指定 外部ウェイト入力を有効にするか無視するかを指定します。 アクセスウェイトサイクル数が0の場合でも、本ビットの設定は有効です。 0 : 外部ウェイト入力有効 1 : 外部ウェイト入力無視
5~2		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
1	HW1	0	R/W	\overline{RD} 、 \overline{WEn} ネゲート アドレス、 \overline{CSn} ネゲート遅延サイクル数
0	HW0	0	R/W	\overline{RD} 、 \overline{WEn} ネゲートから、アドレス、 \overline{CSn} ネゲートまでの遅延サイクル数を指定します。 00 : 0.5 サイクル 01 : 1.5 サイクル 10 : 2.5 サイクル 11 : 3.5 サイクル

(2) パーストROMの場合

• CS0WCR

ビット	ビット名	初期値	R/W	説 明
31~18		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
17	BW1	0	R/W	パーストウェイトサイクル数 パーストリードアクセス時の2回目以降のアクセスサイクルに挿入するウェイトサイクル数を指定します。 00:0サイクル 01:1サイクル 10:2サイクル 11:3サイクル
16	BW0	0	R/W	
15~11		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
10	W3	1	R/W	アクセスウェイトサイクル数 ライトサイクルおよび1回目のリードのアクセスサイクルに挿入するウェイトサイクル数を指定します。 0000:0サイクル 0001:1サイクル 0010:2サイクル 0011:3サイクル 0100:4サイクル 0101:5サイクル 0110:6サイクル 0111:8サイクル 1000:10サイクル 1001:12サイクル 1010:14サイクル 1011:18サイクル 1100:24サイクル 1101:設定禁止 1110:設定禁止 1111:設定禁止
9	W2	0	R/W	
8	W1	1	R/W	
7	W0	0	R/W	

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
6	WM	0	R/W	外部ウェイトマスク指定 外部ウェイト入力を有効にするか無視するかを指定します。 アクセスウェイトサイクル数が0の場合でも、本ビットの設定は有効です。 0: 外部ウェイト入力有効 1: 外部ウェイト入力無視
5~0		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。

• CS4WCR

ビット	ビット名	初期値	R/W	説明
31~18		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
17 16	BW1 BW0	0 0	R/W R/W	バーストウェイトサイクル数 バーストリードアクセス時の2回目以降のアクセスサイクルに挿入するウェイトサイクル数を指定します。 00: 0 サイクル 01: 1 サイクル 10: 2 サイクル 11: 3 サイクル
15~13		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
12 11	SW1 SW0	0 0	R/W R/W	アドレス、 \overline{CSn} アサート \overline{RD} 、 \overline{WEn} アサート遅延サイクル数 アドレス、 \overline{CSn} アサートから、 \overline{RD} 、 \overline{WEn} アサートまでの遅延サイクル数を指定します。 00: 0.5 サイクル 01: 1.5 サイクル 10: 2.5 サイクル 11: 3.5 サイクル

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説 明	
10	W3	1	R/W	アクセスウェイトサイクル数	
9	W2	0	R/W	ライトサイクルおよび1回目のリードアクセスサイクルに挿入するウェイトサイクル数を指定します。 0000 : 0 サイクル 0001 : 1 サイクル 0010 : 2 サイクル 0011 : 3 サイクル 0100 : 4 サイクル 0101 : 5 サイクル 0110 : 6 サイクル 0111 : 8 サイクル 1000 : 10 サイクル 1001 : 12 サイクル 1010 : 14 サイクル 1011 : 18 サイクル 1100 : 24 サイクル 1101 : 設定禁止 1110 : 設定禁止 1111 : 設定禁止	
8	W1	1	R/W		
7	W0	0	R/W		
6	WM	0	R/W		外部ウェイトマスク指定 外部ウェイト入力を有効にするか無視するかを指定します。 アクセスウェイトサイクル数が0の場合でも、本ビットの設定は有効です。 0 : 外部ウェイト入力有効 1 : 外部ウェイト入力無視
5~2		すべて0	R		リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
1	HW1	0	R/W		RD、WE \bar{n} ネゲート アドレス、CS \bar{n} ネゲート遅延サイクル数 RD、WE \bar{n} ネゲートから、アドレス、CS \bar{n} ネゲートまでの遅延サイクル数を指定します。 00 : 0.5 サイクル 01 : 1.5 サイクル 10 : 2.5 サイクル 11 : 3.5 サイクル
0	HW0	0	R/W		

9. バスステートコントローラ (BSC)

(3) SDRAM の場合

• CS3WCR

ビット	ビット名	初期値	R/W	説 明
31~15		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
14 13	TRP1 TRP0	0 0	R/W R/W	オートプリチャージ / PRE コマンド ACTV コマンドサイクル数 オートプリチャージの起動、または PRE コマンド発行後から、同一バンクに対する ACTV コマンド発行までの最小サイクル数を指定します。 00 : 1 サイクル 01 : 2 サイクル 10 : 3 サイクル 11 : 4 サイクル
12		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
11 10	TRCD1 TRCD0	0 1	R/W R/W	ACTV コマンド READ(A) / WRIT(A) コマンドサイクル数 ACTV コマンド発行後、READ(A) / WRIT(A) コマンド発行までの最小サイクル数を指定します。 00 : 1 サイクル 01 : 2 サイクル 10 : 3 サイクル 11 : 4 サイクル
9		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
8 7	A3CL1 A3CL0	1 0	R/W R/W	CAS レイテンシ CAS レイテンシを指定します。 00 : 1 サイクル 01 : 2 サイクル 10 : 3 サイクル 11 : 4 サイクル
6, 5		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

ビット	ビット名	初期値	R/W	説明
4 3	TRWL1 TRWL0	0 0	R/W R/W	WRITA / WRIT コマンド オートプリチャージ / PRE コマンドサイクル数 WRITA または WRIT コマンド発行後に、オートプリチャージが起動されるまでのサイクル数、または PRE コマンド発行までのサイクル数を指定します。 00 : 0 サイクル 01 : 1 サイクル 10 : 2 サイクル 11 : 3 サイクル
2		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
1 0	TRC1 TRC0	0 0	R/W R/W	REF コマンド / セルフリフレッシュ解除 ACTV コマンドサイクル数 REF コマンド発行後、またはセルフリフレッシュ解除後から ACTV コマンド発行までの最小サイクル数を指定します。 00 : 3 サイクル 01 : 4 サイクル 10 : 6 サイクル 11 : 9 サイクル

9.4.4 SDRAM コントロールレジスタ (SDCR)

SDCR は、32 ビットのレジスタで、SDRAM のリフレッシュ方法やアクセス方法、および接続する SDRAM の種類を指定します。

SDCR は、パワーオンリセット時に初期化されますが、マニュアルリセットおよびスタンバイモード時は初期化されずに前の値を保持します。RFSH、RMODE 以外の各ビットは、パワーオンリセット後の初期化設定時に書き込みを行い、以降は値を変更しないでください。RFSH、RMODE ビットに対して書き込みを行う際は、他のビットは変化されずに同じ値を書き込んでください。シンクロナス DRAM 使用時に本レジスタの設定が終了するまで、エリア 3 をアクセスしないでください。

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
31~13		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
12	SLOW	0	R/W	低周波数モード SDRAM に対するコマンド、アドレス、ライトデータを出力するタイミング、およびリードデータを取り込むタイミングを指定します。 本ビットを1に設定すると、コマンド、アドレス、ライトデータ、およびリードデータのホールド時間を延長することができます。このモードは、SDRAM を低周波数で動作させるときに適したモードです。 0: SDRAM に対するコマンド、アドレス、およびライトデータを CKIO の立ち上がり同期して出力する。SDRAM からのリードデータを CKIO の立ち上がり同期して取り込む。 1: SDRAM に対するコマンド、アドレス、およびライトデータを CKIO の立ち下がり同期して出力する。SDRAM からのリードデータを CKIO の立ち下がり同期して取り込む。
11	RFSH	0	R/W	リフレッシュ制御 SDRAM に対してリフレッシュを行うかどうかを指定します。 0: リフレッシュしない。 1: リフレッシュする。
10	RMODE	0	R/W	リフレッシュ制御 RFSH ビットが1のとき、オートリフレッシュを行うのかセルフリフレッシュを行うのかを指定します。 RFSH ビットを1かつ本ビットを1に設定すると、その直後にセルフリフレッシュモードに入ります。RFSH ビットを1かつ本ビットを0に設定すると、RTCSR、RTCNT、および RTCOR レジスタに設定した内容に従いオートリフレッシュを行います。 0: オートリフレッシュを行う。 1: セルフリフレッシュを行う。
9	PDOWN	0	R/W	パワーダウンモード SDRAM 以外のメモリアクセス終了後に、SDRAM をパワーダウンモードにするかどうかを指定します。 本ビットは1に設定すると、SDRAM アクセス終了後、SDRAM 以外のメモリアクセスをトリガとして、CKE 端子をローレベルにして SDRAM をパワーダウンモードにします。 0: SDRAM をパワーダウンモードにしない。 1: SDRAM をパワーダウンモードにする。

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
8	BACTV	0	R/W	バンクアクティブモード オートプリチャージモード (READA および WRITA コマンドを使用) でアクセスするか、バンクアクティブモード (READ および WRIT コマンドを使用) でアクセスするかを指定します。 0: オートプリチャージモード (READA および WRITA コマンドを使用) 1: バンクアクティブモード (READ および WRIT コマンドを使用) 【注】 バンクアクティブモードを指定したとき、データバス幅は、16 ビットに設定してください。
7~5		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
4 3	A3ROW1 A3ROW0	0 0	R/W R/W	ロウアドレスビット数 ロウアドレスのビット数を指定します。 00: 11 ビット 01: 12 ビット 10: 設定禁止 11: 設定禁止
2		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
1 0	A3COL1 A3COL0	0 0	R/W R/W	コラムアドレスビット数 コラムアドレスのビット数を指定します。 00: 8 ビット 01: 9 ビット 10: 設定禁止 11: 設定禁止

9. バスステートコントローラ (BSC)

9.4.5 リフレッシュタイマコントロール/ステータスレジスタ (RTCSR)

RTCSR は、32 ビットのレジスタで、SDRAM のリフレッシュに関する各種設定を行います。

書き込み時には、書き込みデータの上位 16 ビットを H'A55A としてライトプロテクトを解除してください。

ビット	ビット名	初期値	R/W	説明
31~8		すべて 0	R/W	ライトプロテクト 読み出すと常に 0 が読み出されます。書き込み時には、書き込みデータの上位 16 ビットを H'A55A としてライトプロテクトを解除してください。下位 8 ビットは常に 0 にしてください。その他の値を書いた場合の動作の保証はできません。
7	CMF	0	R/W	コンペアマッチフラグ リフレッシュタイマカウンタ (RTCNT) とリフレッシュタイムコンスタントレジスタ (RTCOR) の値が一致したことを示すステータスフラグです。 次の条件でセット/クリアされます。 0: クリア条件: CMF=1 の状態で RTCSR を読み出した後、CMF に 0 を書き込んだとき 1: セット条件: RTCNT=RTCOR になったとき
6		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
5	CKS2	0	R/W	クロックセレクト
4	CKS1	0	R/W	リフレッシュタイマカウンタ (RTCNT) をカウントアップするクロックを選択します。
3	CKS0	0	R/W	000: カウントアップ停止 001: B /4 010: B /16 011: B /64 100: B /256 101: B /1024 110: B /2048 111: B /4096

ビット	ビット名	初期値	R/W	説明
2	RRC2	0	R/W	リフレッシュ回数
1	RRC1	0	R/W	リフレッシュタイムカウンタ (RTCNT) とリフレッシュタイムコンスタントレジスタ (RTCOR) の値が一致してリフレッシュ要求が発生したとき、連続してリフレッシュを行う回数を指定します。
0	RRC0	0	R/W	本機能により、リフレッシュを発生させる周期を長くすることができます。 000 : 1 回 001 : 2 回 010 : 4 回 011 : 6 回 100 : 8 回 101 : 設定禁止 110 : 設定禁止 111 : 設定禁止

9.4.6 リフレッシュタイムカウンタ (RTCNT)

RTCNT は、8 ビットのカウンタレジスタで、RTCSR レジスタの CKS2 ~ CKS0 ビットで選択したクロックによりカウントアップされます。RTCNT と RTCOR レジスタの値が一致すると、RTCNT は 0 にクリアされます。また、255 までカウントアップすると次は 0 に戻ります。

書き込み時には、書き込みデータの上位 16 ビットを H'A55A としてライトプロテクトを解除してください。

ビット	ビット名	初期値	R/W	説明
31 ~ 8		すべて 0	R/W	ライトプロテクト 読み出すと常に 0 が読み出されます。書き込み時には、書き込みデータの上位 16 ビットを H'A55A としてライトプロテクトを解除してください。下位 8 ビットは常に 0 にしてください。その他の値を書いた場合の動作の保証はできません。
7 ~ 0		すべて 0	R/W	8 ビットのカウンタ

9.4.7 リフレッシュタイムコンスタントレジスタ (RTCOR)

RTCOR は、32 ビットのレジスタです。RTCOR と RTCNT カウンタの値が一致すると、RTCSR レジスタの CMF ビットが 1 にセットされ、RTCNT は 0 にクリアされます。

SDCR レジスタの RFSH ビットが 1 にセットされている場合は、この一致信号によってリフレッシュ要求が発生します。リフレッシュ要求は、実際にリフレッシュ動作が行われるまで保持されます。次の一致までにリフレッシュ要求が処理されない場合は、前の要求は無効となります。

書き込み時には、書き込みデータの上位 16 ビットを H'A55A としてライトプロテクトを解除してください。

9. バスステートコントローラ (BSC)

ビット	ビット名	初期値	R/W	説明
31~8		すべて0	R/W	ライトプロテクト 読み出すと常に0が読み出されます。書き込み時には、書き込みデータの上位16ビットをH'A55Aとしてライトプロテクトを解除してください。下位8ビットは常に0にしてください。その他の値を書いた場合の動作の保証はできません。
7~0		すべて0	R/W	リフレッシュタイム リフレッシュタイムカウンタ(RTCNT)のカウンタアップ上限値を設定します。

9.4.8 リセットウェイトカウンタ (RWCNT)

RWCNTは、7ビットのカウンタレジスタです。パワーオンリセット解除後、CKIO信号に同期してカウントアップを開始し、レジスタの値がH'007Fになるとカウントアップを停止します。本レジスタがカウントアップを行っている間は、外部バスへのアクセスが待たされます。フラッシュメモリ等のリセット解除から最初のアクセスまでの最小時間を確保するために、本レジスタは存在します。本レジスタへの読み出し、書き込みは行えません。

9.5 動作説明

9.5.1 エンディアン / アクセスサイズとデータアライメント

本LSIでは、バイトデータの並び方を上位バイト (MSByte) が0番地側になるビッグエンディアン、下位バイト (LSByte) が0番地側になるリトルエンディアンのいずれもサポートしています。この切り換えは、外部ピン (MD5ピン) でパワーオンリセット時に設定します。パワーオンリセット時、MD5端子がローレベルのときビッグエンディアンになり、MD5端子がハイレベルのときリトルエンディアンになります。

また、データバス幅は、通常メモリ、バイト選択付きSRAMとしては8ビットおよび16ビット幅の2種類から選べ (ただしエリア0では16ビット幅固定)、SDRAMは16ビット幅固定です。データのアライメントは、各デバイスのデータバス幅およびエンディアンにあわせて行われます。したがって、8ビット幅のデバイスからロングワードデータを読み出すためには、4回の読み出し動作が必要です。本LSIでは、それぞれのインタフェース間で、データのアライメントおよびデータ長の変換を自動的に行います。

エンディアンとデバイスのデータ幅とアクセスの単位との関係を表9.3~表9.6に示します。

表 9.3 16 ビット外部デバイス / ビッグエンディアンのアクセスとデータアライメント

オペレーション		データバス		ストロープ信号	
		D15~D8	D7~D0	$\overline{WE1}/DQM1$	$\overline{WE0}/DQM0$
0 番地バイトアクセス		データ 7~0	-	アサート	-
1 番地バイトアクセス		-	データ 7~0	-	アサート
2 番地バイトアクセス		データ 7~0	-	アサート	-
3 番地バイトアクセス		-	データ 7~0	-	アサート
0 番地ワードアクセス		データ 15~8	データ 7~0	アサート	アサート
2 番地ワードアクセス		データ 15~8	データ 7~0	アサート	アサート
0 番地ロング ワードアクセス	1 回目 (0 番地)	データ 31~24	データ 23~16	アサート	アサート
	2 回目 (1 番地)	データ 15~8	データ 7~0	アサート	アサート

表 9.4 8 ビット外部デバイス / ビッグエンディアンのアクセスとデータアライメント

オペレーション		データバス		ストロープ信号	
		D15~D8	D7~D0	$\overline{WE1}/DQM1$	$\overline{WE0}/DQM0$
0 番地バイトアクセス		-	データ 7~0	-	アサート
1 番地バイトアクセス		-	データ 7~0	-	アサート
2 番地バイトアクセス		-	データ 7~0	-	アサート
3 番地バイトアクセス		-	データ 7~0	-	アサート
0 番地 ワードアクセス	1 回目 (0 番地)	-	データ 15~8	-	アサート
	2 回目 (1 番地)	-	データ 7~0	-	アサート
2 番地 ワードアクセス	1 回目 (2 番地)	-	データ 15~8	-	アサート
	2 回目 (3 番地)	-	データ 7~0	-	アサート
0 番地ロング ワードアクセス	1 回目 (0 番地)	-	データ 31~24	-	アサート
	2 回目 (1 番地)	-	データ 23~16	-	アサート
	3 回目 (2 番地)	-	データ 15~8	-	アサート
	4 回目 (3 番地)	-	データ 7~0	-	アサート

9. バスステートコントローラ (BSC)

表 9.5 16 ビット外部デバイス/リトルエンディアンのアクセスとデータアライメント

オペレーション		データバス		ストローク信号	
		D15~D8	D7~D0	$\overline{WE1}/DQM1$	$\overline{WE0}/DQM0$
0番地バイトアクセス			データ7~0		アサート
1番地バイトアクセス		データ7~0		アサート	
2番地バイトアクセス			データ7~0		アサート
3番地バイトアクセス		データ7~0		アサート	
0番地ワードアクセス		データ15~8	データ7~0	アサート	アサート
2番地ワードアクセス		データ15~8	データ7~0	アサート	アサート
0番地ロング ワードアクセス	1回目(0番地)	データ15~8	データ7~0	アサート	アサート
	2回目(2番地)	データ31~24	データ23~16	アサート	アサート

表 9.6 8 ビット外部デバイス/リトルエンディアンのアクセスとデータアライメント

オペレーション		データバス		ストローク信号	
		D15~D8	D7~D0	$\overline{WE1}/DQM1$	$\overline{WE0}/DQM0$
0番地バイトアクセス			データ7~0		アサート
1番地バイトアクセス			データ7~0		アサート
2番地バイトアクセス			データ7~0		アサート
3番地バイトアクセス			データ7~0		アサート
0番地ワード アクセス	1回目(0番地)		データ7~0		アサート
	2回目(1番地)		データ15~8		アサート
2番地ワード アクセス	1回目(2番地)		データ7~0		アサート
	2回目(3番地)		データ15~8		アサート
0番地ロング ワードアクセス	1回目(0番地)		データ7~0		アサート
	2回目(1番地)		データ15~8		アサート
	3回目(2番地)		データ23~16		アサート
	4回目(3番地)		データ31~24		アサート

9.5.2 通常空間インタフェース

(1) 基本タイミング

通常空間アクセスは、おもにバイト選択端子のない SRAM の直結を考慮してストロープ信号を出力します。バイト選択付き端子のある SRAM を使用する場合は、「9.5.7 バイト選択付き SRAM インタフェース」を参照してください。図 9.3 に通常空間アクセスの基本タイミングを示します。ウェイトのない通常アクセスは、2 サイクルで終了します。 \overline{BS} 信号はバスサイクルの開始を表し、1 サイクルアサートされます。

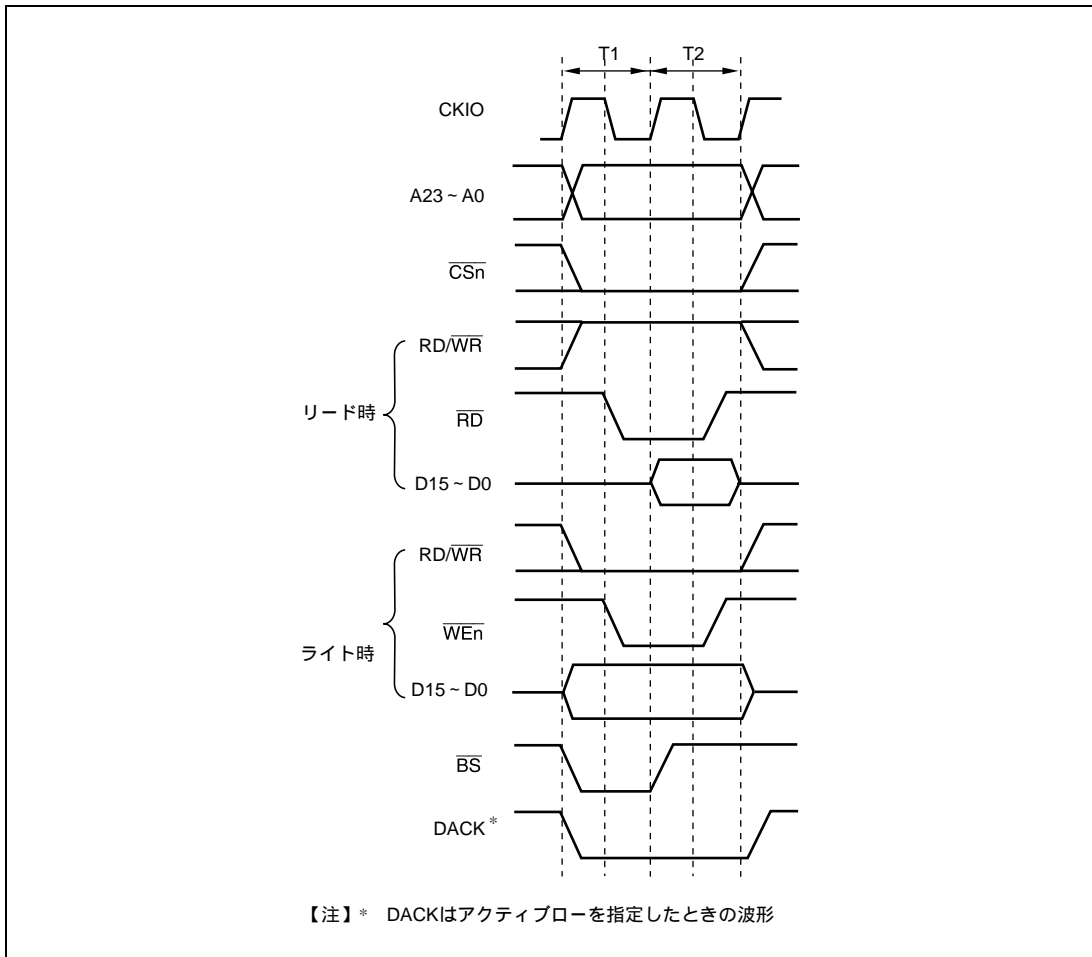


図 9.3 通常空間基本アクセス (アクセスウェイト 0)

9. バスステートコントローラ (BSC)

リード時は、外部バスに対してアクセスサイズの指定がありません。アドレスの最下位ビットに正しいアクセス開始アドレスが出力されていますが、アクセスサイズの指定がないので、16ビットデバイスでは16ビットを、8ビットデバイスでは8ビットを常に読み出すことになります。ライト時には、書き込みを行うバイトの \overline{WE} 信号のみがアサートされます。

キャッシュフィル/コピーバックのための読み出し/書き込みは、設定したデータバス幅に従い、合計16バイトを連続して行います。この途中ではバス権を解放しません。バイトまたはワードオペランドアクセス時および奇数ワード境界への分岐時のキャッシュミスに関しても、本LSI外部インタフェース上は必ずロングワードアクセスでアクセスを行います。ライトスルー領域の書き込み、およびキャッシュ非対象領域の読み出し/書き込みに関しては、実際のアクセスサイズに従ってアクセスを行います。

データバスにバッファを設ける場合には、 \overline{RD} 信号を用いてリードデータの出力制御を行う必要があります。 $\overline{RD}/\overline{WR}$ 信号は、アクセスを行っていないときはリード状態(ハイレベル出力)となっているため、これを用いて外付けデータバッファの制御を行うと出力が衝突する危険性があるので注意が必要です。

図9.4、図9.5に通常空間連続アクセスの例を示します。CSnWCR.WMビットを0に設定すると、外部ウェイトを評価するために1サイクル T_{nop} が挿入されます(図9.4)。しかし、CSnWCR.WMビットを1に設定すると、外部ウェイトが無視され T_{nop} サイクルの挿入を抑制することができます(図9.5)。図9.6、図9.7にSRAMの接続例を示します。本LSIのアドレス空間はバイト単位で割り当てられていますので、ロングワード単位のデータを扱う場合(図9.6)や、ワード単位のデータを扱う場合(図9.7)には、外部メモリのアドレスと本LSIのアドレスのビット位置が、ずれた接続をすることになりますので、注意してください。

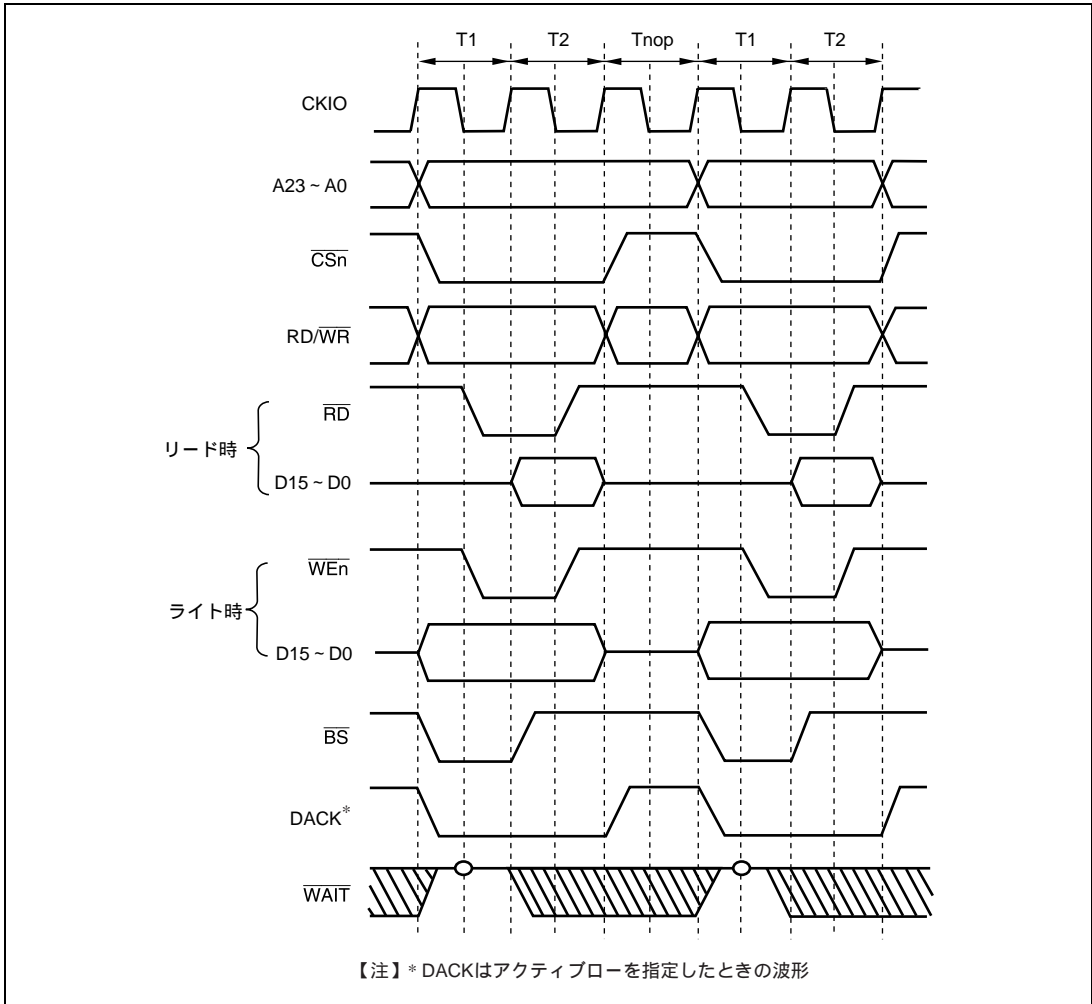


図 9.4 通常空間連続アクセス例 1
 データバス幅 16 ビット、ロングワードアクセス、CSnWCR.WM ビット = 0
 (アクセスウェイト 0、サイクル間ウェイト 0)

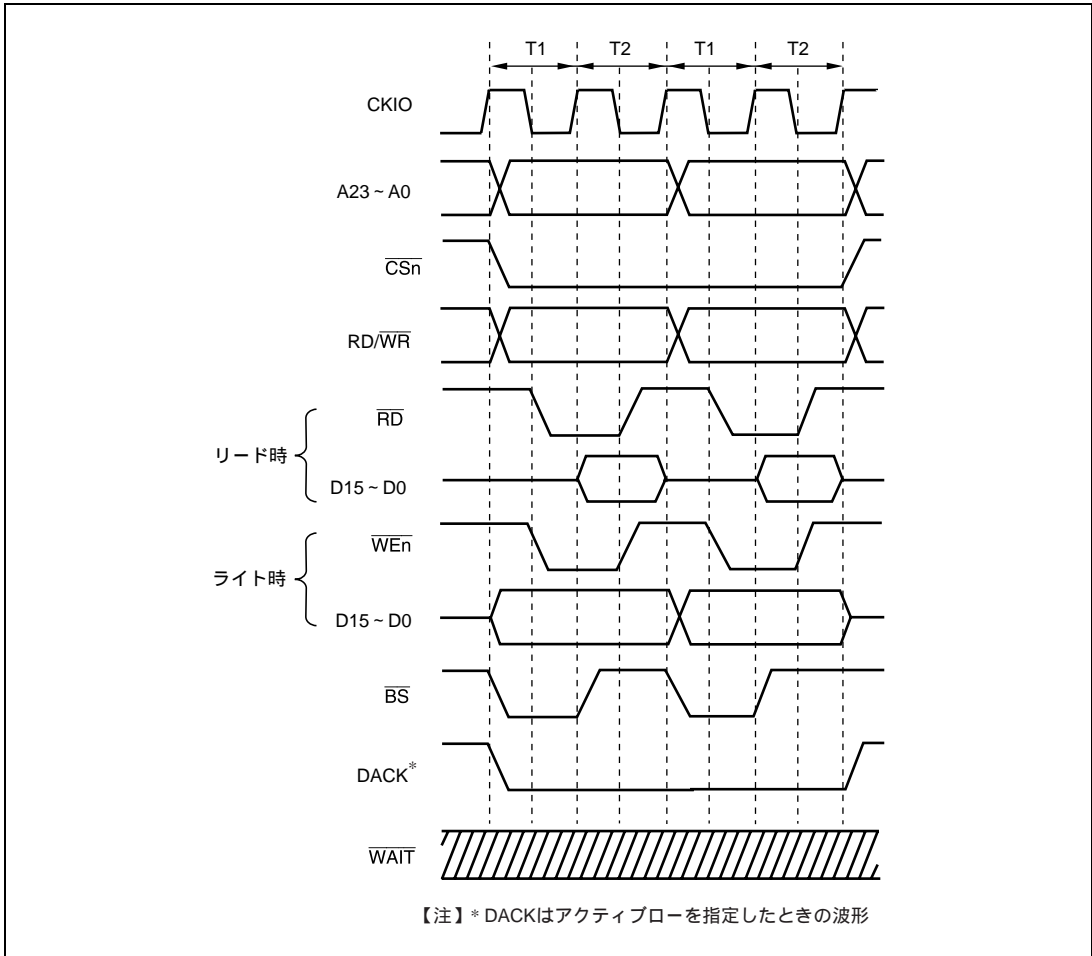


図 9.5 通常空間連続アクセス例 2
 データバス幅 16 ビット、ロングワードアクセス、 $\overline{CSnWCR.WM}$ ビット = 1
 (アクセスウェイト 0、サイクル間ウェイト 0)

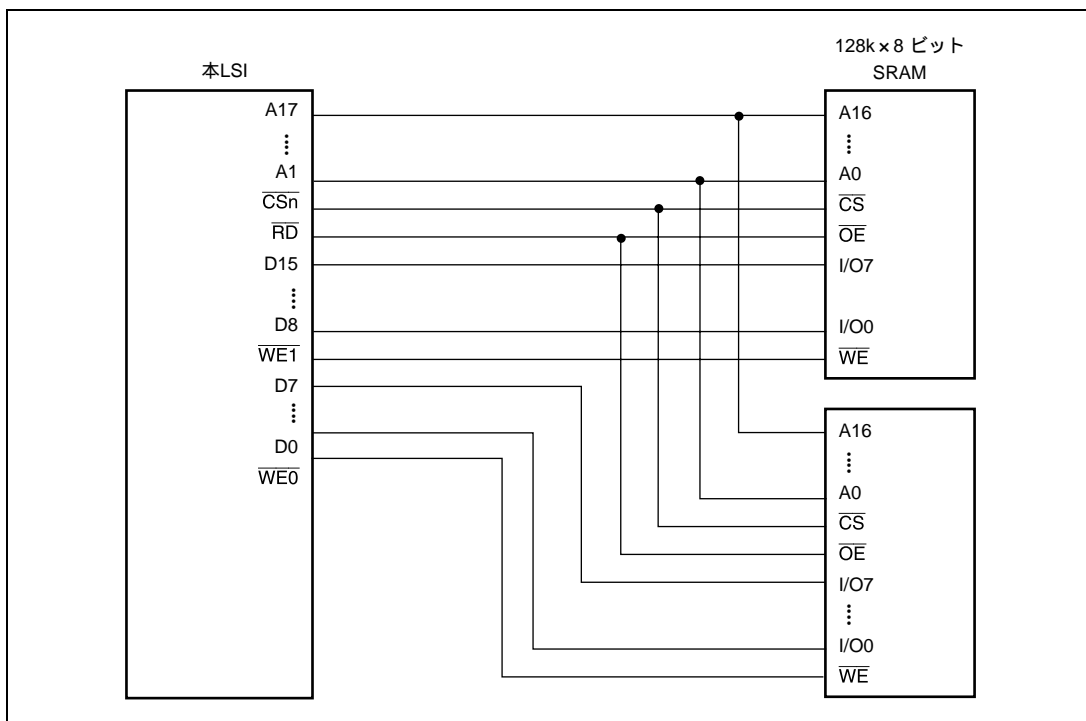


図 9.6 16 ビットデータ幅 SRAM 接続例

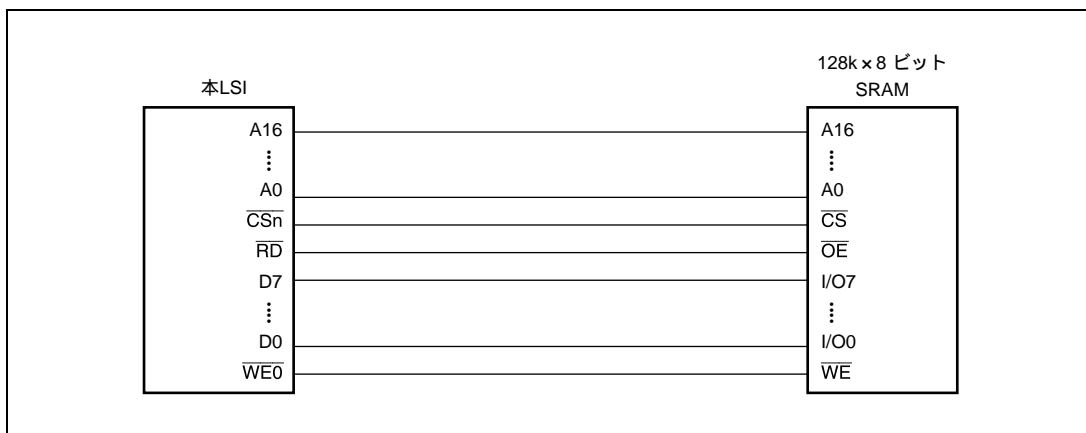


図 9.7 8 ビットデータ幅 SRAM 接続例

9.5.3 アクセスウェイト制御

CSnWCR の WR3、WR2、WR1、および W0 ビットの設定により、通常空間アクセスのウェイトサイクルの挿入を制御できます。エリア 4 では、リードアクセスとライトアクセスで独立にウェイトサイクルを挿入することが可能です。その他のエリアのアクセスウェイトは、リードおよびライトサイクルで共通となります。図 9.8 に示す通常空間のアクセスでは、Tw のサイクルがウェイトサイクルとして指定サイクル数だけ挿入されます。

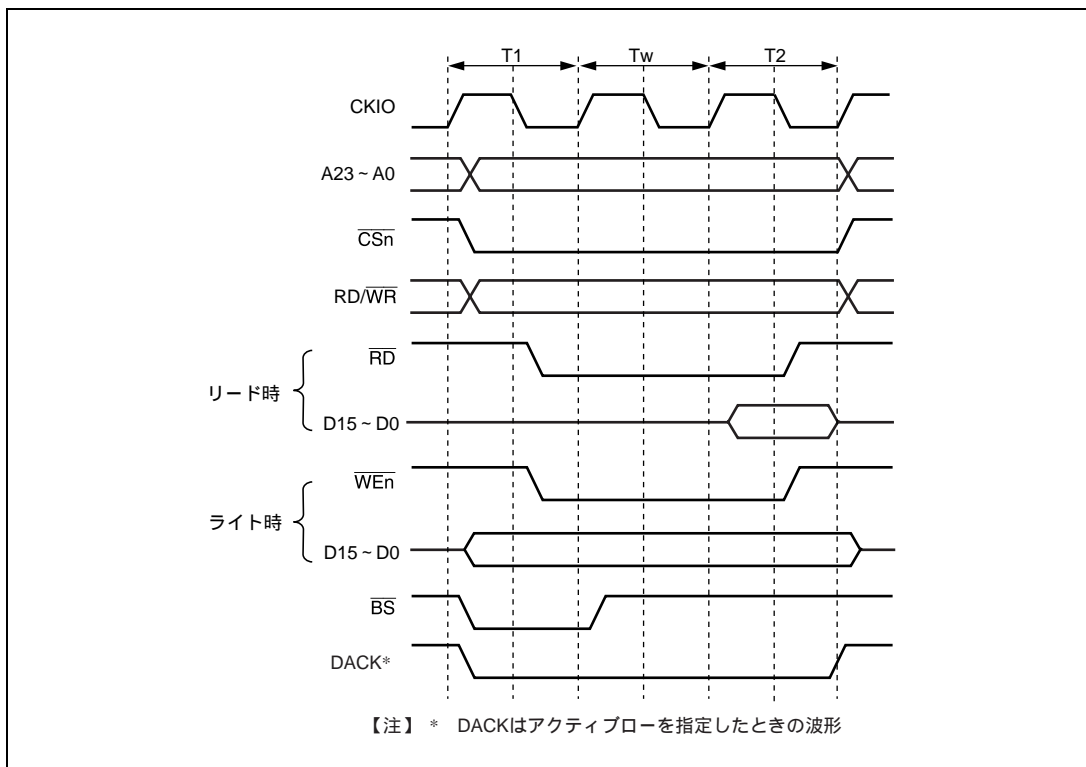


図 9.8 通常空間アクセスのウェイトタイミング (ソフトウェアウェイトのみ)

CSnWCR の WM ビットを 0 としたときには、外部からのウェイト入力 $\overline{\text{WAIT}}$ 信号もサンプリングされます。 $\overline{\text{WAIT}}$ 信号のサンプリングを図 9.9 に示します。ソフトウェアウェイトとして 2 サイクルのウェイトを指定しています。 $\overline{\text{WAIT}}$ 信号は、T1 または Tw サイクルから T2 サイクルに移行する際に、CKIO の立ち下がりでサンプリングされます。

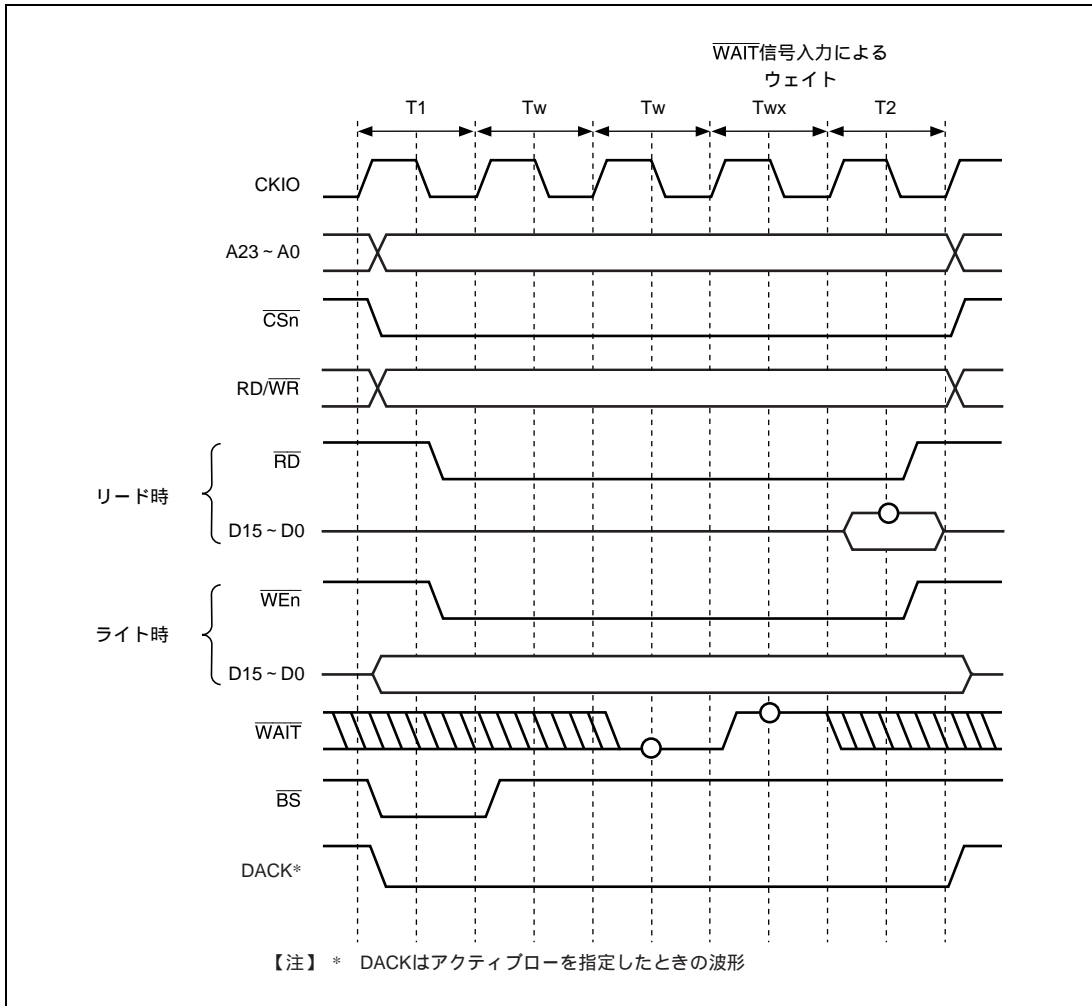


図 9.9 通常空間アクセスのウェイトタイミング (WAIT 信号によるウェイト挿入)

9.5.4 \overline{CSn} アサート期間拡張

\overline{CSnWCR} の SW1 と SW0 ビットの設定により、 \overline{CSn} アサートから \overline{RD} と $\overline{WE_n}$ アサートまでのサイクル数を指定できます。また、HW1 と HW0 ビットの設定により、 \overline{RD} と $\overline{WE_n}$ ネゲートから \overline{CSn} ネゲートまでのサイクル数を指定できます。これにより、外部デバイスとのフレキシブルなインタフェースがとれます。例を図 9.10 に示します。Th および Tf サイクルが通常サイクルの前と後ろにそれぞれ付加されています。これらのサイクルでは、 \overline{RD} と $\overline{WE_n}$ 以外はアサートされますが、 \overline{RD} と $\overline{WE_n}$ はアサートされません。また、データは Tf サイクルまで延長して出力されるので、書き込み動作の遅いデバイスなどに有効です。

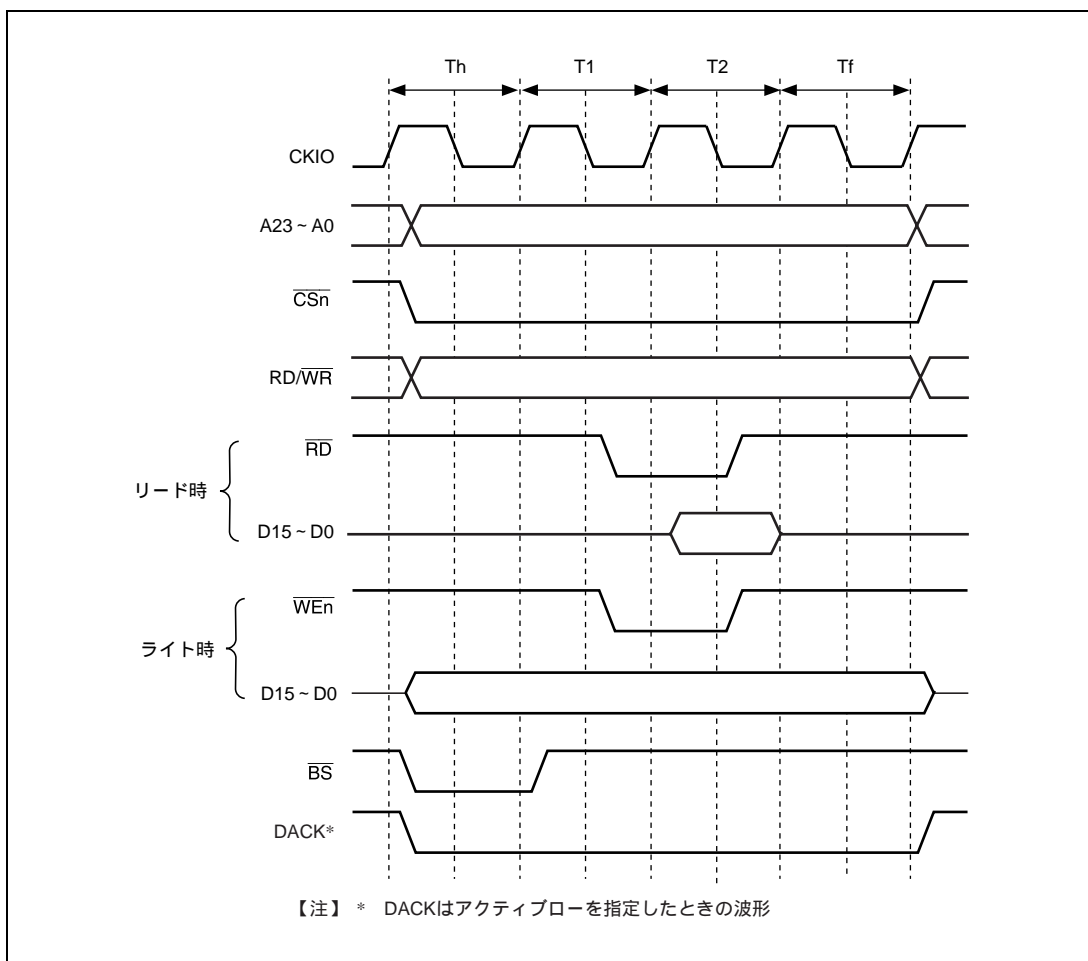


図 9.10 \overline{CSn} アサート期間拡張

9.5.5 SDRAM インタフェース

(1) SDRAM 直結インタフェース

本 LSI に接続可能な SDRAM は、ロウアドレスが 11/12/13 ビット、カラムアドレスが 8/9/10 ビット、バンク数が 4 以下、リード・ライトコマンドサイクルで A10 端子をプリチャージモードの設定に使用する製品です。

SDRAM を直結するための制御信号は、 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 、 $\text{RD}/\overline{\text{WR}}$ 、DQM1、DQM0、CKE、および $\overline{\text{CS3}}$ です。 $\overline{\text{CS3}}$ を除く信号は各空間に共通であり、CKE を除く信号は $\overline{\text{CS3}}$ がアサートされているときのみ有効になります。

SDRAM を接続する空間のデータバス幅は、16 ビットに設定してください。

SDRAM の動作モードとしては、バーストリード/シングルライト (バースト長 1) とバーストリード/バーストライト (バースト長 1) をサポートしています。

$\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 、 $\text{RD}/\overline{\text{WR}}$ 、および特定のアドレス信号によって、SDRAM に対するコマンドが指定されます。コマンドには、NOP、オートリフレッシュ (REF)、セルフリフレッシュ (SELF)、全バンクプリチャージ (PALL)、指定バンクプリチャージ (PRE)、バンクアクティブ (ACTV)、リード (READ)、プリチャージ付きリード (READA)、ライト (WRIT)、プリチャージ付きライト (WRITA)、およびモードレジスタ書き込み (MRS) があります。

アクセスするバイトの指定は、DQM1 および DQM0 によって行われます。該当する DQMn がローレベルのバイトに対してリード/ライトが行われます。DQMn とアクセスするバイトの関係は、「9.5.1 エンディアン/アクセスサイズとデータアライメント」を参照してください。

図 9.11 に本 LSI と SDRAM との接続例を示します。本 LSI のアドレス空間はバイト単位で割り当てられていますので、ロングワード単位のデータを扱う場合や、ワード単位のデータを扱う場合には、外部メモリのアドレスと本 LSI のアドレスのビット位置が、ずれた接続をすることになりますので、注意してください。

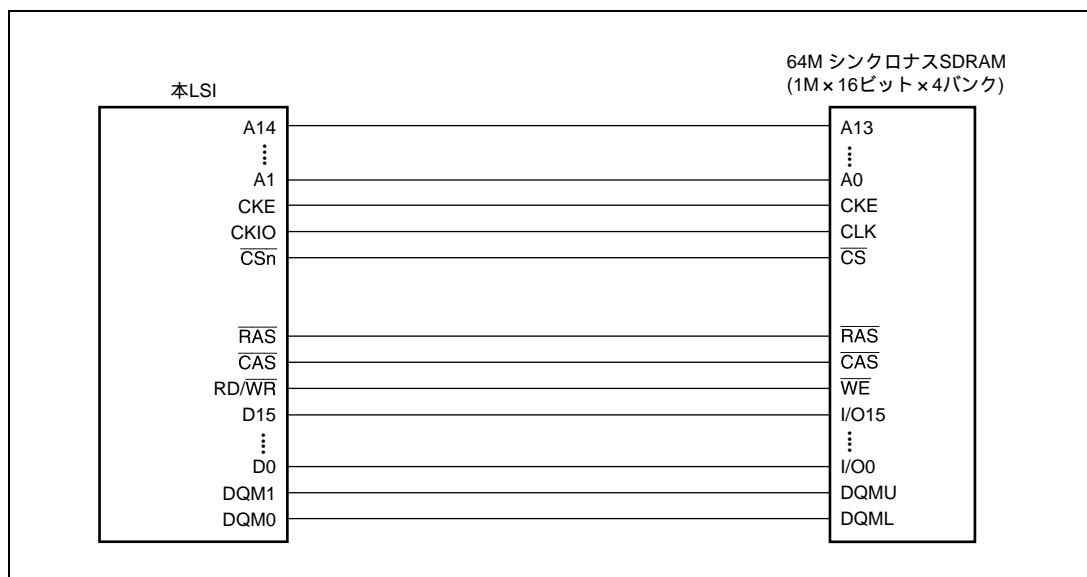


図 9.11 16 ビットデータ幅 SDRAM 接続例

9. バスステートコントローラ (BSC)

(2) アドレスマルチプレクス

CS3BCR の BSZ[1:0]ビット、SDCR の A3ROW[1:0]ビット、および A3COL[1:0]ビットの設定に従って、外付けのアドレスマルチプレクス回路なしに SDRAM を接続できるようにアドレスのマルチプレクスを行います。表 9.7 ~ 表 9.9 に BSZ[1:0]、A3ROW[1:0]、および A3COL[1:0]の設定とアドレス端子に出力されるビットの関係を示します。この表以外の設定は、行わないでください。この表以外の設定を行った場合の動作は、保証されません。A23 ~ A18 は、マルチプレクスを行わず常に本来のアドレスが出力されています。

データバス幅が 16 ビットするとき (BSZ[1:0]=B'10) は、SDRAM の A0 端子はワードアドレスの指定を行います。したがって、SDRAM の A0 端子を本 LSI の A1 端子に接続し、以下 A1 端子を A2 端子にという順で接続してください。

表 9.7 CS3BCR.BSZ[1:0]、SDCR.A3ROW[1:0]、SDCR.A3COL[1:0]とアドレスマルチプレクスの関係 (1)

設定				
CS3BCR.BSZ[1:0]	SDCR.A3ROW[1:0]	SDCR.A3COL[1:0]		
B'10 (データバス幅： 16 ビット)	B'00 (ロウアドレス ビット数：11 ビット)	B'00 (コラムアドレス ビット数：8 ビット)		
本 LSI の 出力端子	ロウアドレス 出力サイクル	コラムアドレス 出力サイクル	シンクロナス DRAM の端子	機能
A17	A25	A17		未使用
A16	A24	A16		
A15	A23	A15		
A14	A22	A14		
A13	A21*2	A21*2	A12(BA1)	バンク指定
A12	A20*2	A20*2	A11(BA0)	
A11	A19	L/H*1	A10/AP	アドレス / プリチャージ指定
A10	A18	A10	A9	アドレス
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		未使用
接続メモリ例				
16M ビット品 (512K ワード×16 ビット×2 バンク、コラム 8 ビット品) 1 個				

【注】 *1 L/H はコマンド指定に使われるビットであり、アクセスモードによってローまたはハイに固定されます。

*2 バンクアドレス指定

表 9.8 CS3BCR.BSZ[1:0]、SDCR.A3ROW[1:0]、SDCR.A3COL[1:0]とアドレスマルチプレクスの関係 (2)

設定				
CS3BCR.BSZ[1:0]	SDCR.A3ROW[1:0]	SDCR.A3COL [1:0]		
B'10 (データバス幅 : 16 ビット)	B'01 (ロウアドレス ビット数 : 12 ビット)	B'00 (カラムアドレス ビット数 : 8 ビット)		
本 LSI の 出力端子	ロウアドレス 出力サイクル	カラムアドレス 出力サイクル	シンクロナス DRAM の端子	機能
A17	A25	A17		未使用
A16	A24	A16		
A15	A23	A15		
A14	A22* ²	A22* ²	A13(BA1)	バンク指定
A13	A21* ²	A21* ²	A12(BA0)	
A12	A20	A12	A11	アドレス
A11	A19	L/H* ¹	A10/AP	アドレス/ プリチャージ指定
A10	A18	A10	A9	アドレス
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		未使用
接続メモリ例				
64M ビット品 (1M ワード×16 ビット×4 バンク、カラム 8 ビット品) 1 個				

【注】 *1 L/H はコマンド指定に使われるビットであり、アクセスモードによってローまたはハイに固定されます。

*2 バンクアドレス指定

9. バスステートコントローラ (BSC)

表 9.9 CS3BCR.BSZ[1:0]、SDCR.A3ROW[1:0]、SDCR.A3COL[1:0]とアドレスマルチプレクスの関係 (3)

設定				
CS3BCR.BSZ[1:0]	SDCR.A3ROW[1:0]	SDCR.A3COL[1:0]		
B'10 (データバス幅 : 16 ビット)	B'01 (ロウアドレス ビット数 : 12 ビット)	B'01 (コラムアドレス ビット数 : 9 ビット)		
本 LSI の 出力端子	ロウアドレス 出力サイクル	コラムアドレス 出力サイクル	シンクロナス DRAM の端子	機能
A17	A26	A17		未使用
A16	A25	A16		
A15	A24	A15		
A14	A23* ²	A23* ²	A13(BA1)	バンク指定
A13	A22* ²	A22* ²	A12(BA0)	
A12	A21	A12	A11	アドレス
A11	A20	L/H* ¹	A10/AP	アドレス/ プリチャージ指定
A10	A19	A10	A9	アドレス
A9	A18	A9	A8	
A8	A17	A8	A7	
A7	A16	A7	A6	
A6	A15	A6	A5	
A5	A14	A5	A4	
A4	A13	A4	A3	
A3	A12	A3	A2	
A2	A11	A2	A1	
A1	A10	A1	A0	
A0	A9	A0		
接続メモリ例				
128M ビット品 (2M ワード×16 ビット×4 バンク、コラム 9 ビット品) 1 個				

【注】 *1 L/H はコマンド指定に使われるビットであり、アクセスモードによってローまたはハイに固定されます。

*2 バンクアドレス指定

(3) パーストリード

本 LSI でパーストリードが発生する条件は以下のとおりです。

1. データバス幅よりもリードのアクセスサイズが大きいとき
2. キャッシュミス時の16バイト転送時
3. DMACでの16バイト転送 (ノンキャッシュ領域アクセス) のとき

本 LSI は、SDRAM に対し常にバースト長 1 でアクセスします。たとえば、16 ビットのデータバスに接続された SDRAM から連続して 16 バイト分のデータを読み出すときは、バースト長 1 のリードを 8 回連続して行います。このときのアクセスをバースト数 8 と呼びます。表 9.10 にアクセスサイズとバースト数の関係を示します。

表 9.10 アクセスサイズとバースト数の関係

データバス幅	アクセスサイズ	バースト数
16 ビット	8 ビット	1
	16 ビット	1
	32 ビット	2
	16 バイト	8

パーストリード時のタイミングチャートを図 9.12 と図 9.13 に示します。図 9.12 では、ACTV コマンド出力を行なう Tr サイクルに続いて、READ コマンドを Tc1 ~ Tc7 サイクルに、READA コマンドを Tc8 サイクルに発行し、Td1 から Td8 のサイクルに外部クロック (CKIO) の立ち上がりでリードデータを受け取ります。Tap サイクルは SDRAM 内部で READA コマンドによるオートプリチャージの完了を待つサイクルであり、この間は同一のバンクに対して新たなコマンドの発行は行ないません。ただし、別の CS 空間や同じ SDRAM の異なるバンクに対するアクセスは可能です。CS3WCR の TRP1 および TRP0 の指定によって Tap のサイクル数を決定します。

本 LSI では、様々な周波数で SDRAM と接続するために CS3WCR の各ビットを設定することによりウェイトサイクルを挿入することができます。各種ウェイトの設定を行なった例が図 9.13 となります。ACTV コマンド出力サイクル Tr から READA コマンド出力サイクル Tc1 までのサイクル数は、CS3WCR の TRCD1 および TRCD0 ビットによって指定することができます。TRCD1 および TRCD0 の設定が 2 サイクル以上の場合、Tr サイクルと Tc1 サイクルの間に NOP コマンド発行サイクル Trw サイクルが挿入されます。READA コマンド出力サイクル Tc1 からリードデータ取りこみサイクル Td1 までのサイクル数は、CS3WCR の A3CL1 と A3CL0 ビットによって指定することができます。このサイクル数は、シンクロナス DRAM の CAS レイテンシに相当します。シンクロナス DRAM の CAS レイテンシの仕様は 3 サイクルまでですが、本 LSI では、1 サイクルから 4 サイクルまで設定できます。これは、本 LSI とシンクロナス DRAM の間にラッチを含む回路を設けて接続するためのものです。

9. バスステートコントローラ (BSC)

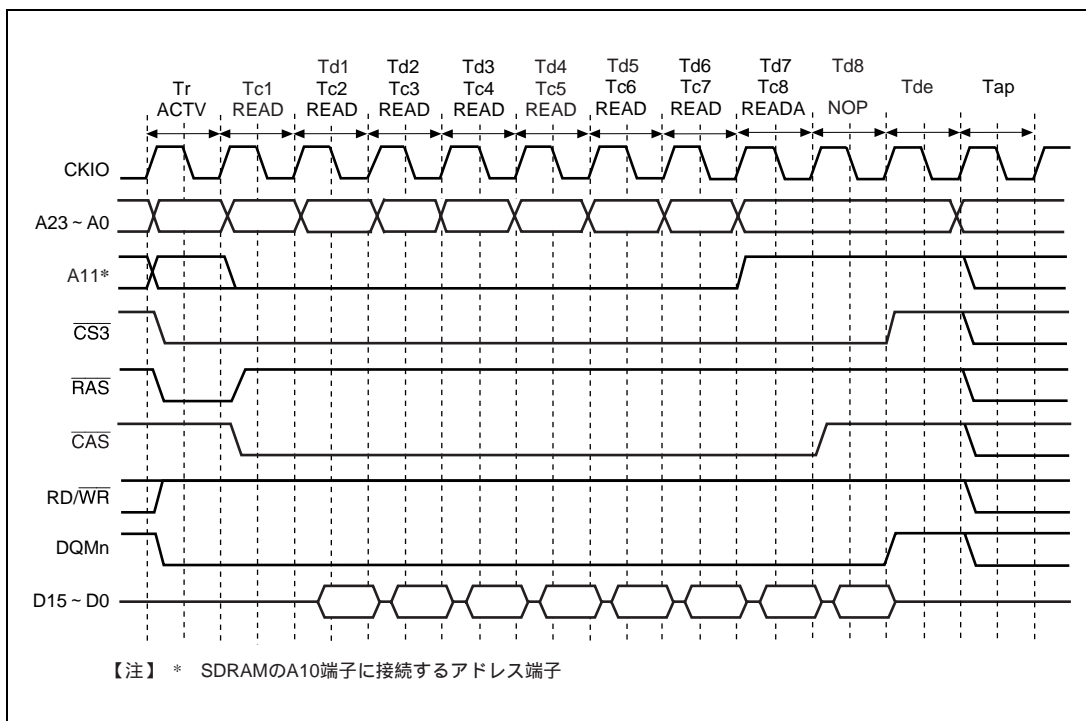


図 9.12 バーストリード基本タイミング (オートプリチャージ)

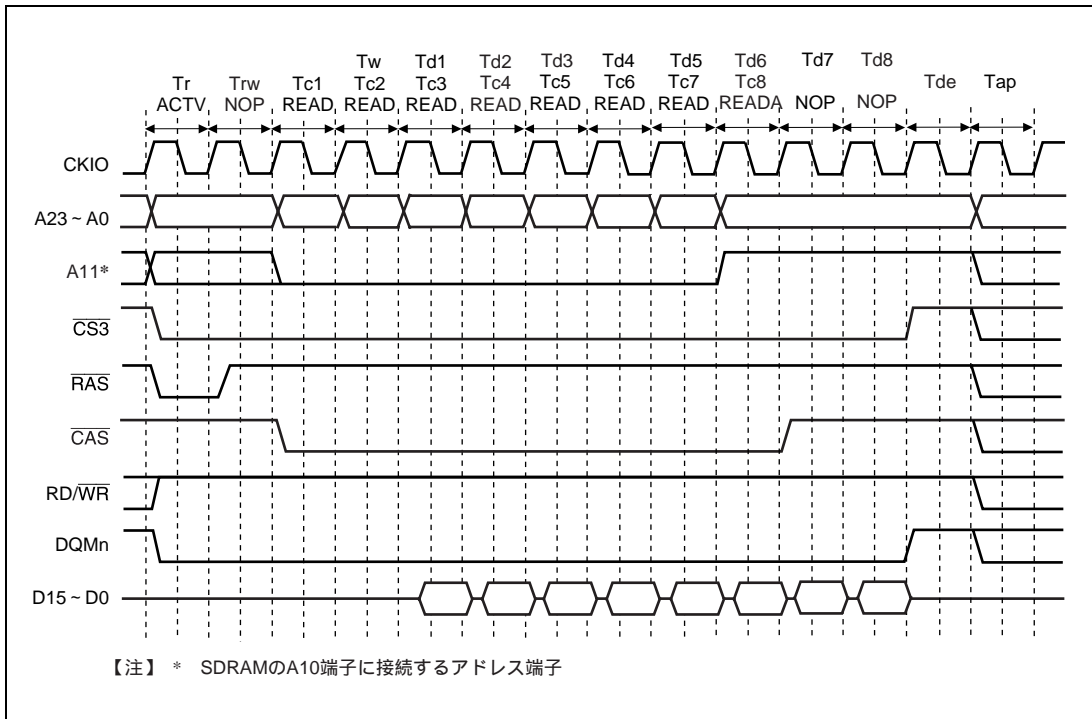


図 9.13 バーストリードウェイト指定タイミング (オートプリチャージ)

(4) シングルリード

ノンキャッシュ領域でかつデータバス幅がアクセスサイズ以上のときは、リードアクセスは1回で終了します。これをシングルリードと呼びます。SDRAMは、バーストリード/シングルライトのモードでバースト長1に設定しているので必要なデータのみ出力します。このため、キャッシュスルー領域をアクセスしても無駄なバスサイクルは、発生しません。

シングルリードの基本タイミングチャートを図 9.14 に示します。

9. バスステートコントローラ (BSC)

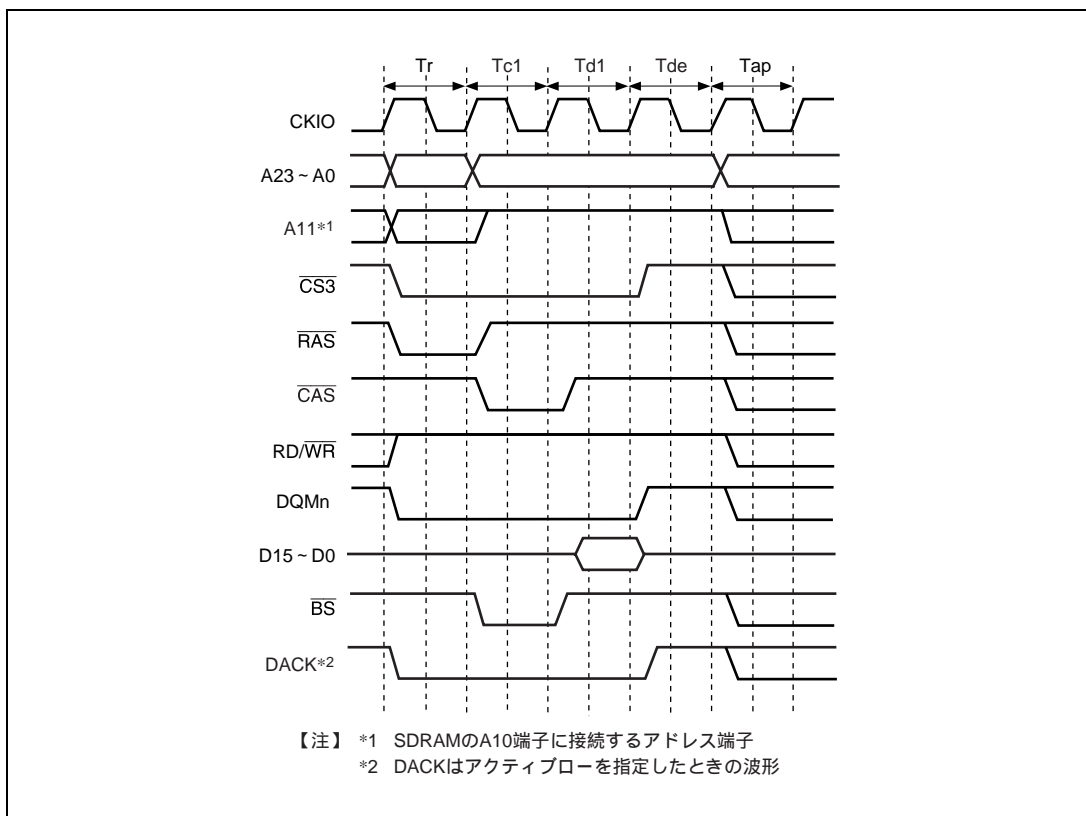


図 9.14 シングルリードの基本タイミング (オートプリチャージ)

(5) パーストライト

本 LSI でパーストライトが発生する条件は、以下のとおりです。

1. データバス幅よりもライトのアクセスサイズが大きいとき
2. キャッシュのコピーバックが発生したとき
3. DMACでの16バイト転送 (ノンキャッシュ領域アクセス) のとき

本 LSI は、SDRAM に対し常にパースト長 1 でアクセスします。たとえば、16 ビットのデータバスに接続された SDRAM から連続して 16 バイト分のデータを書き込むときは、パースト長 1 のライトを 8 回連続して行います。アクセスサイズとパースト数の関係は、表 9.10 に従います。図 9.15 にパーストライト時のタイミングチャートを示します。パーストライトでは ACTV コマンド出力を行う T_r サイクルに続いて WRIT コマンドを $T_{c1} \sim T_{c7}$ サイクルに、オートプリチャージを行う WRITA コマンドを T_{c8} サイクルに発行します。ライトサイクルでは、ライトデータはライトコマンドと同時に出力します。オートプリチャージ付きライトコマンド出力後は、オートプリチャージが起動されるまでの時間待 T_{rw1} サイクル、そしてオートプリチャージの完了を待 T_{ap} サイクルが続きます。 T_{ap} サイクルは SDRAM 内部で WRITA コマンドによるオートプリチャージの完了を待 T_{ap} サイクルで

あり、この間は同一のバンクに対して新たなコマンドの発行は行いません。ただし、別の CS_n 空間や同じ SDRAM の異なるバンクに対するアクセスは可能です。Trwl サイクルは CS3WCR の TRWL1 と TRWL0 ビット、および Tap サイクルは CS3WCR の TRP1 と TRP0 ビットの指定で決定されます。

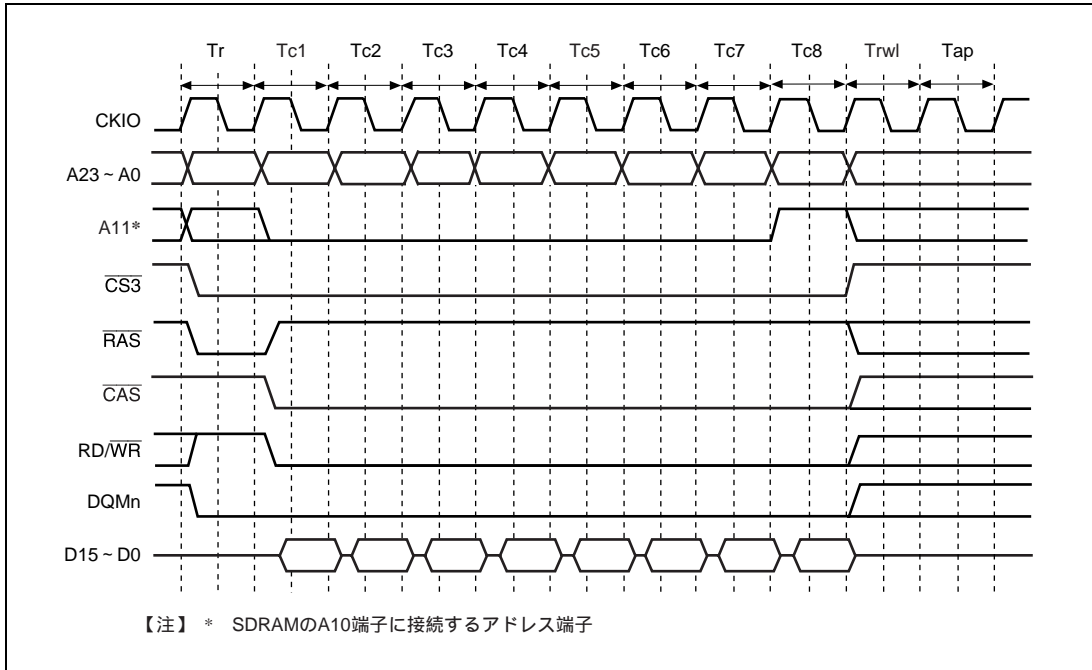


図 9.15 バーストライト基本タイミング (オートプリチャージ)

9. バスステートコントローラ (BSC)

(6) シングルライト

ノンキャッシュ領域でかつデータバス幅がアクセスサイズ以上のときは、ライトアクセスは1回で終了します。これをシングルライトと呼びます。シングルライトの基本タイミングチャートを図9.16に示します。

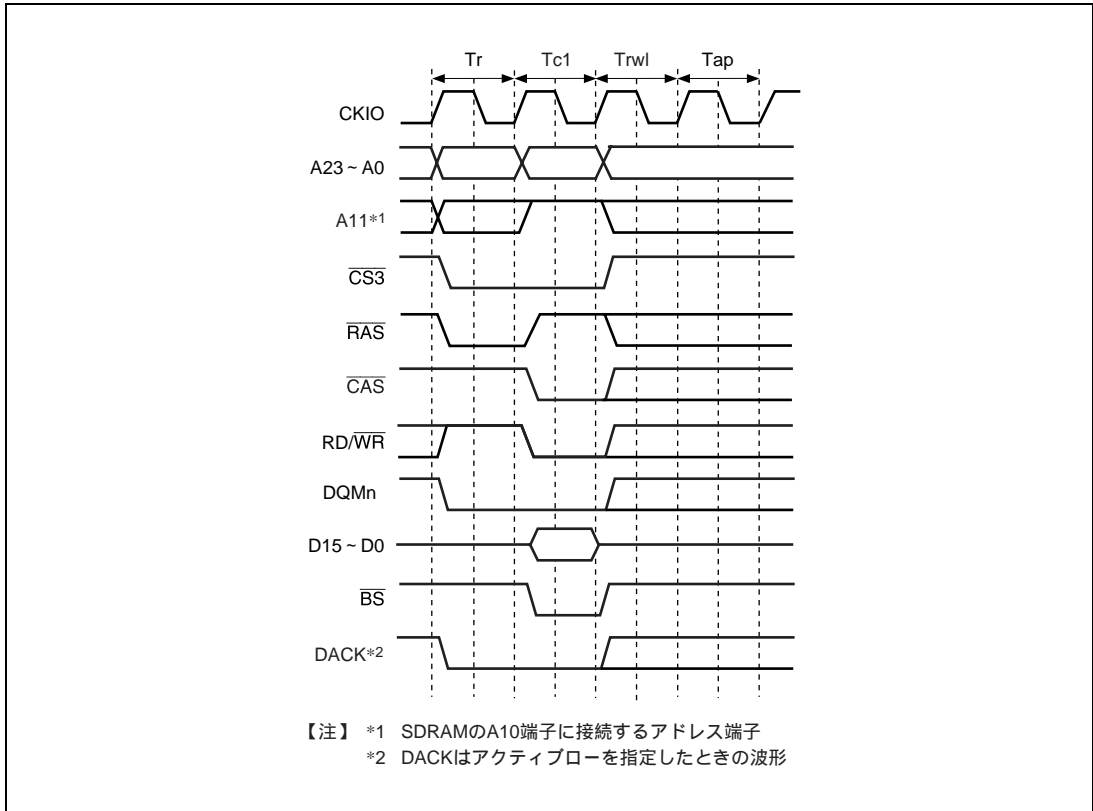


図 9.16 シングルライト基本タイミング (オートプリチャージ)

(7) バンクアクティブ

同一のロウアドレスに対するアクセスを高速実行するため、SDRAM のバンク機能を使用することができます。SDCR の BACTV ビットが 1 の場合は、オートプリチャージなしのコマンド (READ または WRIT) を使用してアクセスを行います。これをバンクアクティブ機能といいます。

バンクアクティブ機能を用いた場合は、アクセスが終了してもプリチャージは行われません。同じバンクの同じロウアドレスにアクセスする場合は、ACTV コマンドを発行せずに、直ちに READ または WRIT コマンドを発行することができます。SDRAM の内部は複数のバンクに分かれているので、それぞれのバンクで 1 つずつのロウアドレスをアクティブ状態としておくことができます。次のアクセスが異なるロウアドレスに対するものであった場合には、最初に PRE コマンドを発行して当該バンクのプリチャージを行い、プリチャージ完了後 ACTV コマンド、READ または WRIT コマンドの順に発行します。異なるロウアドレスに対するアクセスが続く場合には、アクセス要求があってからプリチャージを行うため、かえってアクセス時間が延びてしまう可能性があります。PRE コマンド発行から ACTV コマンド発行までのサイクル数は、CS3WCR の TRP[1:0] ビットで指定します。

書き込みの場合は、オートプリチャージを行うと、WRITA コマンド発行後 $Trwl + Tap$ サイクルの間同一バンクに対してコマンドを発行できません。バンクアクティブモードを用いると、同一ロウアドレスの場合には続けて READ または WRIT コマンドを発行することができます。したがって、1 つの書き込みごとに $Trwl + Tap$ サイクルだけサイクル数を短縮することができます。

各バンクをアクティブ状態にしておける時間 ($tRAS$) には、制限があります。プログラムの実行によって、この制限を守る周期で異なるロウアドレスにアクセスする保証がない場合には、リフレッシュ周期を $tRAS$ 以下に設定する必要があります。

図 9.17 にオートプリチャージのないバーストリードサイクルを、図 9.18 には同一のロウアドレスに対するバーストリードサイクルを、図 9.19 には異なるロウアドレスに対するバーストリードサイクルを示します。同様に、図 9.20 にオートプリチャージのないシングルライトサイクルを、図 9.21 に同一のロウアドレスに対するシングルライトサイクルを、図 9.22 には異なるロウアドレスに対するシングルライトサイクルを示します。

図 9.18 において READ コマンドを発行する Tc サイクルに先立って、何も行わない $Tnop$ サイクルが挿入されています。これは SDRAM からのデータリード時に、読み出しバイト指定を行う $DQMn$ 信号について、2 サイクルのレイテンシを守るために挿入されています。CAS レイテンシが 2 以上の場合には、 Tc サイクル以降に $DQMn$ 信号をアサートしても 2 サイクルのレイテンシが守られるので、 $Tnop$ サイクルの挿入は行われません。

バンクアクティブ機能が設定されている空間のそれぞれのバンクに対するアクセスのみを見た場合は、同一のロウアドレスに対するアクセスが続く限り、図 9.17 または図 9.20 で始まり、図 9.18 または図 9.21 を繰り返します。間に別の空間や別のバンクに対するアクセスがあっても影響しません。バンクアクティブ中に別のロウアドレスに対するアクセスが発生した場合は、図 9.18 または図 9.21 の代わりに図 9.19 または図 9.22 のバスサイクルを行います。バンクアクティブモードでも、リフレッシュサイクルの後またはバスアービトレーションによるバス解放の後には、すべてのバンクが非アクティブな状態になります。

9. バスステートコントローラ (BSC)

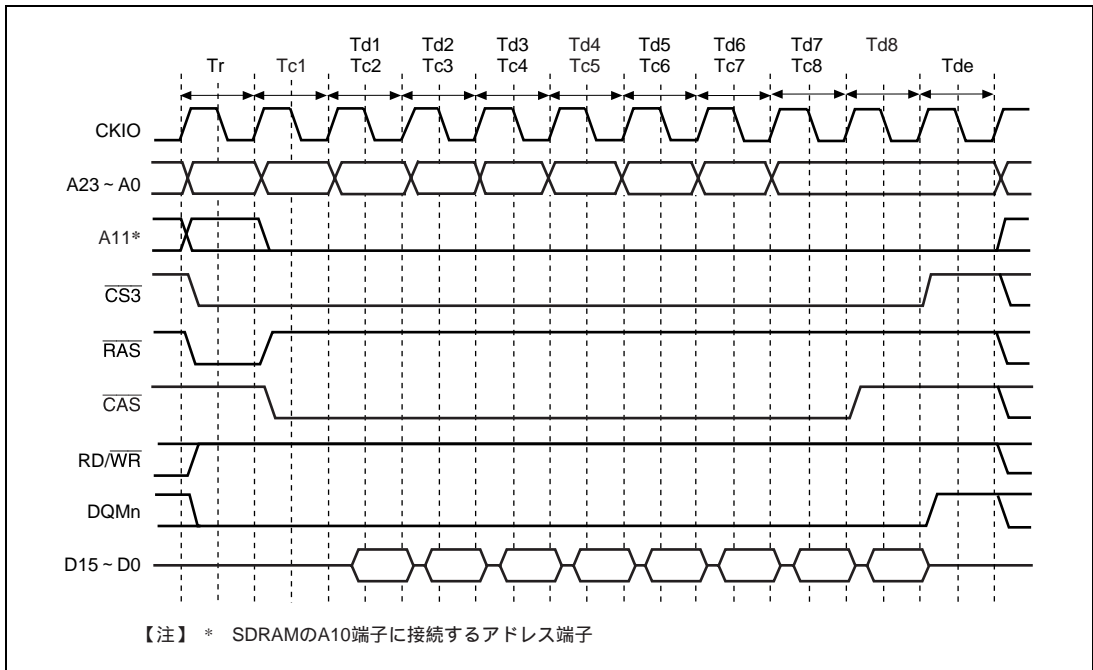


図 9.17 バーストリードタイミング (オートプリチャージなし)

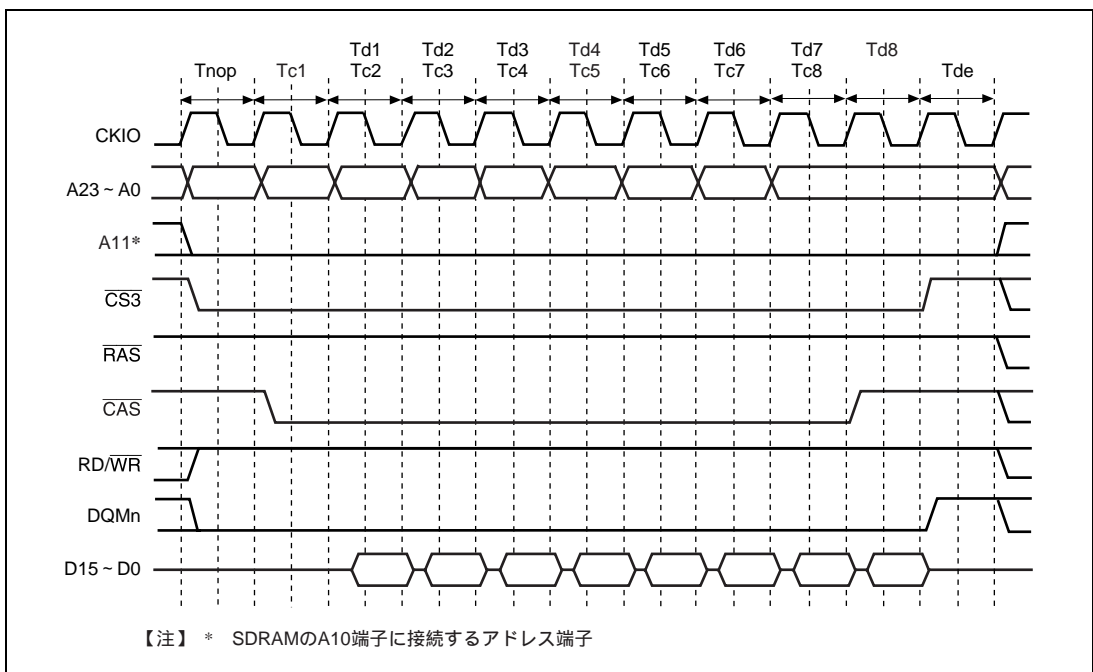


図 9.18 バーストリードタイミング (バンクアクティブ、同一ロウアドレス)

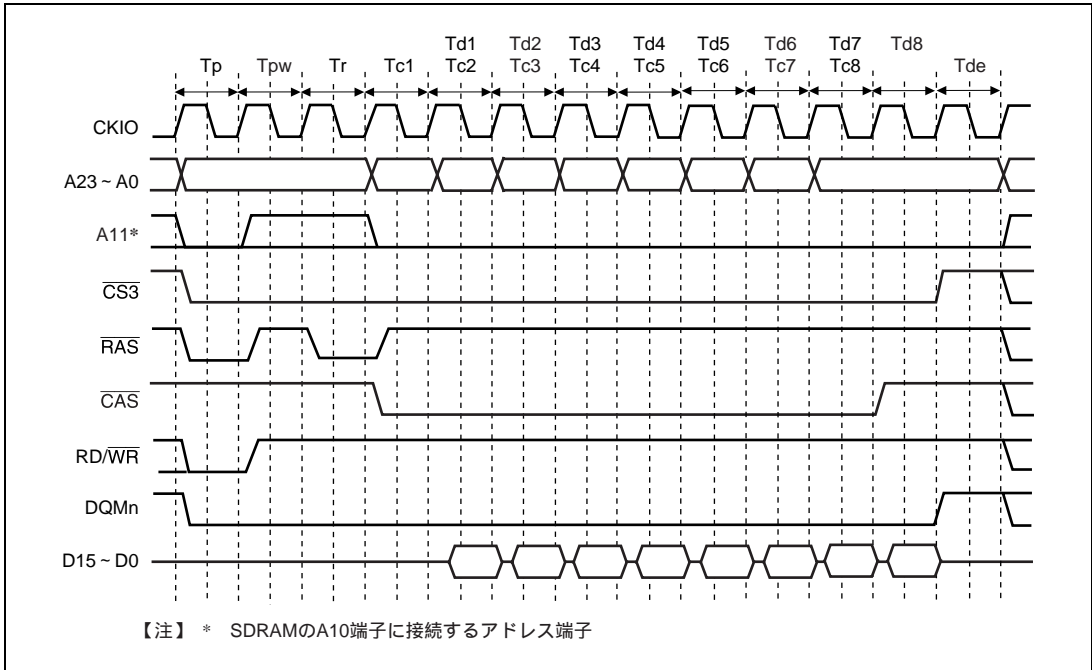


図 9.19 バーストリードタイミング (バンクアクティブ、異なるロウアドレス)

9. バスステートコントローラ (BSC)

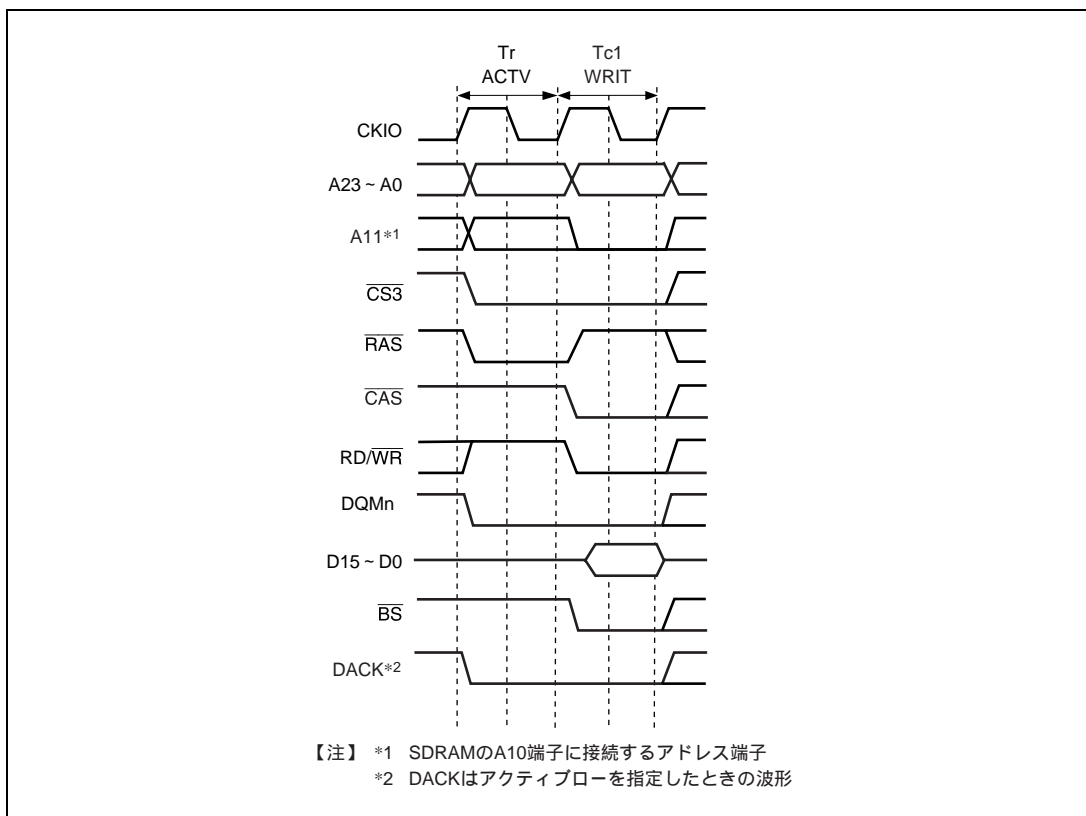


図 9.20 シングルライトタイミング (オートプリチャージなし)

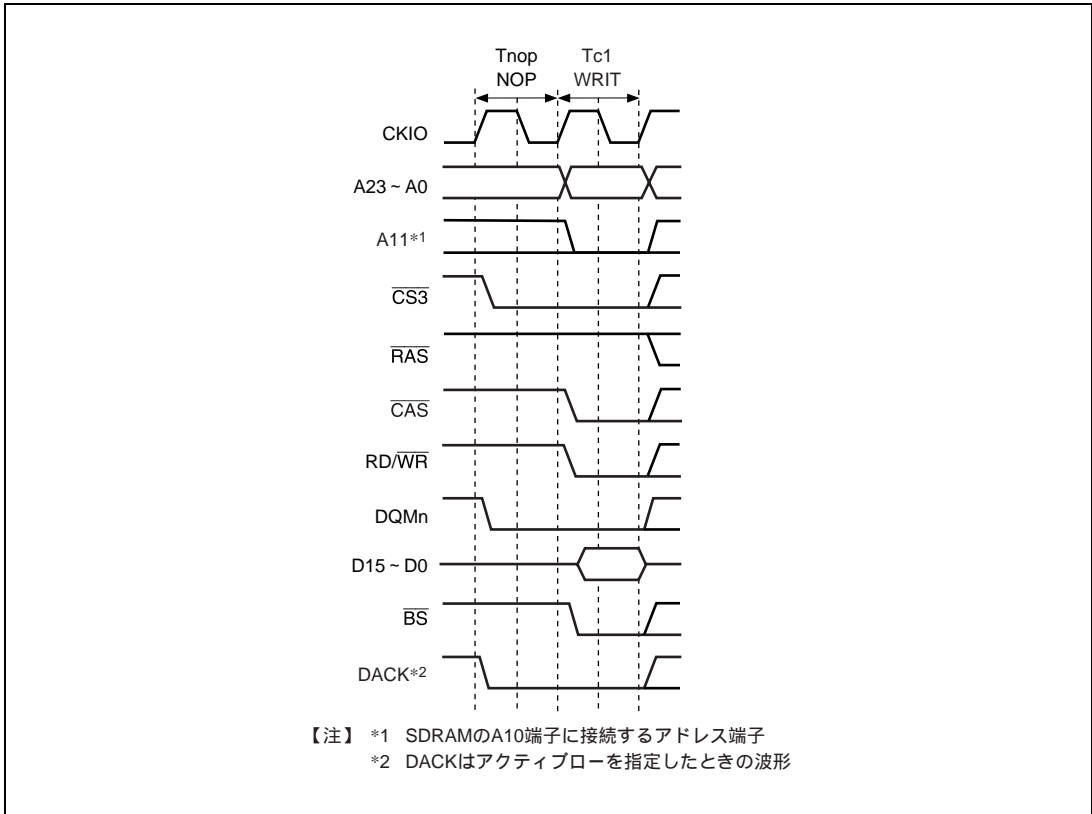


図 9.21 シングルライトタイミング (バンクアクティブ、同一ロウアドレス)

9. バスステートコントローラ (BSC)

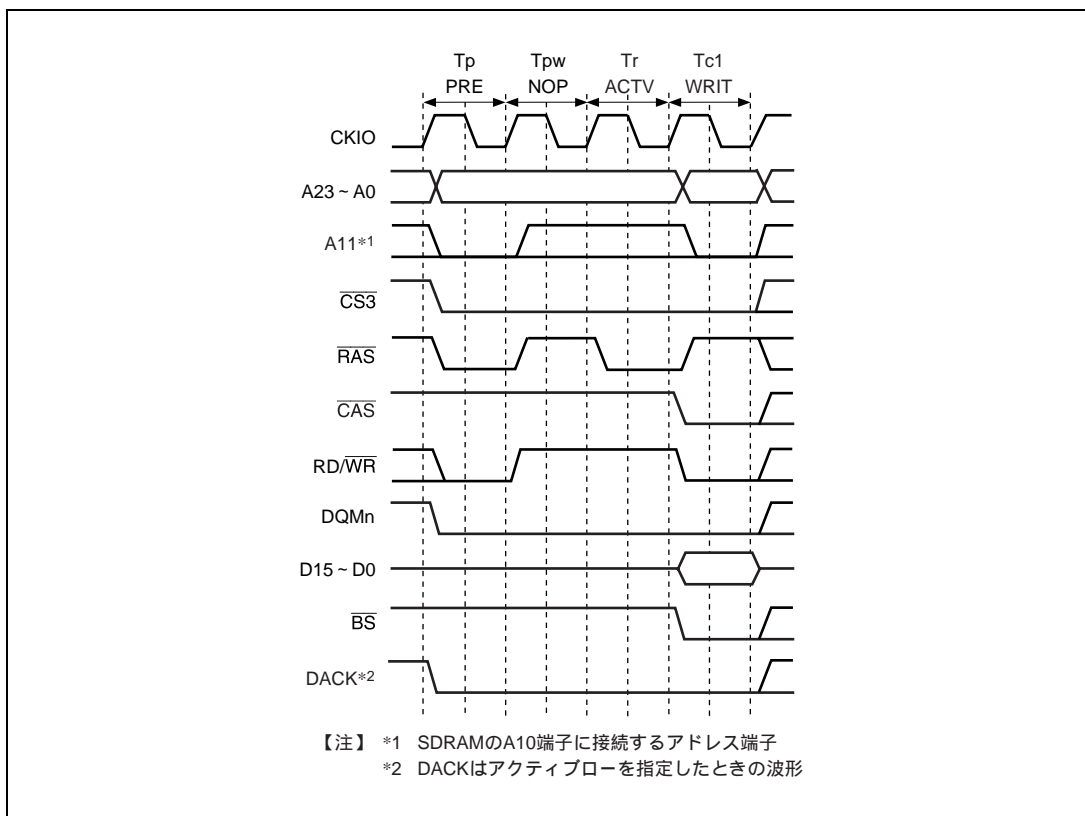


図 9.22 シングルライトタイミング (バンクアクティブ、異なるロウアドレス)

(8) リフレッシュ

BSC は、SDRAM のリフレッシュを制御する機能を備えています。SDCR の RFSH ビットを 1 に、RMODE ビットを 0 に設定することによって、オートリフレッシュを行うことができます。また、RTCSCR の RRC[2:0] ビットを設定することにより、連続してリフレッシュを発生させることができます。さらに、長時間 SDRAM にアクセスしないときは、RFSH ビットと RMODE ビットを共に 1 にすることによって、消費電力が少ないセルリフレッシュを起動することができます。

(a) オートリフレッシュ

RTCSCR の CKS[2:0] ビットで選択した入力クロックと、RTCOR に設定した値とで決まる間隔で RTCSCR の RRC[2:0] に設定した回数のリフレッシュが行われます。使用する SDRAM のリフレッシュ間隔規定を満たすように、各レジスタの設定を行ってください。最初に RTCOR、RTCNT、SDCR の RFSH ビットおよび RMODE ビットの設定を行い、次いで RTCSCR の CKS[2:0] ビットおよび RRC[2:0] ビットの設定を行ってください。CKS[2:0] によって入力クロックを選択すると、RTCNT はそのときの値からカウントアップを開始します。RTCNT の値は常に RTCOR の値と比較されており、両者の値が一致するとリフレッシュ要求が発生し、RRC[2:0] に設定された回数分のオートリフレッシュが実行されます。同時に RTCNT は 0 にクリアされ、カウントアップが再開されます。

図 9.23 にオートリフレッシュサイクルのタイミングを示します。オートリフレッシュが起動されると、プリチャージ中のバンクがある場合は、その完了を待った後、すべてのバンクをアクティブ状態からプリチャージ状態にするため、 T_p サイクルで PALL コマンドを発行します。次いで、CS3WCR の TRP[1:0] ビットで設定された数のアイドルサイクル挿入後、REF コマンドを T_{rr} サイクルに発行します。 T_{rr} サイクル後 CS3WCR の TRC[1:0] ビットで指定されるサイクル数の間は、新たなコマンドの発行は行いません。SDRAM のリフレッシュサイクル時間の規定 (t_{RC}) を満たすように TRC[1:0] を設定する必要があります。CS3WCR の TRP[1:0] ビットの設定値が 2 サイクル以上の場合、 T_p サイクルと T_{rr} サイクルの間に NOP サイクルが挿入されます。

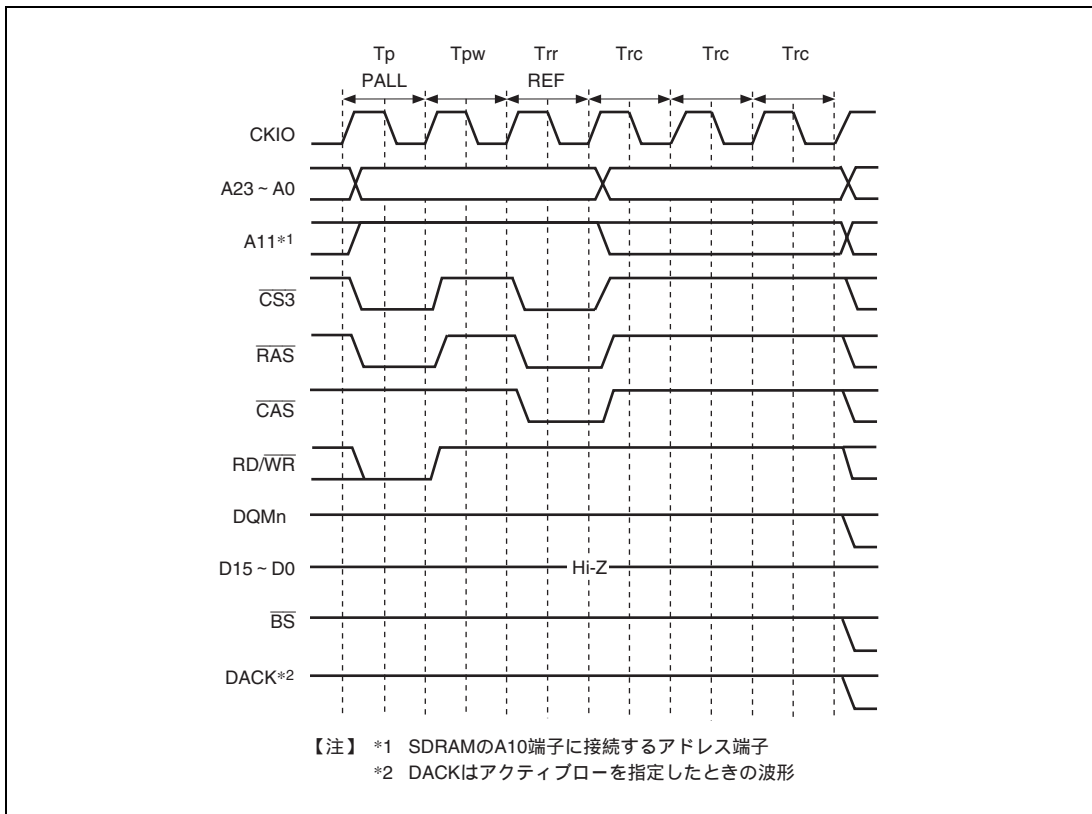


図 9.23 オートリフレッシュタイミング

(b) セルフリフレッシュ

セルフリフレッシュは、SDRAM の内部でリフレッシュタイミングとリフレッシュアドレスを生成する一種のスタンバイモードです。SDCR の RFSH ビットと RMODE ビットをともに 1 にすることによって起動します。セルフリフレッシュが起動されると、プリチャージ中のバンクがある場合は、その完了を待った後、 T_p サイクルで PALL コマンドを発行します。次いで、CS3WCR の TRP[1:0] ビットで設定されたアイドルサイクルを挿入後、SELF コマンドを発行します。セルフリフレッシュ状態の間は、SDRAM にアクセスすることができません。セルフリフ

9. バスステートコントローラ (BSC)

レッシュの解除は、RMODE ビットを 0 にすることによって行われます。セルフリフレッシュ解除後、CS3WCR の TRC[1:0]ビットで指定されるサイクル数の間はコマンドの発行を行いません。

セルフリフレッシュのタイミングを図 9.24 に示します。セルフリフレッシュ解除後、直ちにオートリフレッシュが正しい間隔で行われるように設定してください。オートリフレッシュの設定をしている状態からセルフリフレッシュにした場合は、セルフリフレッシュ解除時に RFSH=1、RMODE=0 とすれば、オートリフレッシュが再開されます。セルフリフレッシュ解除からオートリフレッシュ開始までに時間がかかる場合には、(RTCOR の値 - 1) を RTCNT に設定することにより、直ちにオートリフレッシュを開始することができます。

セルフリフレッシュに設定した後は、本 LSI をスタンバイ状態にした場合にもセルフリフレッシュ状態は継続され、割り込みによるスタンバイ状態からの復帰後もセルフリフレッシュ状態が保持されます。

マニュアルリセットによってもセルフリフレッシュ状態が解除されることはありません。

パワーオンリセットの場合には、BSC のレジスタが初期化されるため、セルフリフレッシュ状態が解除されます。セルフリフレッシュ解除後、SDRAM に対して CKE のハイレベル保持期間を満たす必要があります。SDRAM パワーダウンモード (SDCR.PDOWN ビット=1) 使用時は、1 度パワーダウンモードを解除してからセルフリフレッシュモードへ遷移させてください。

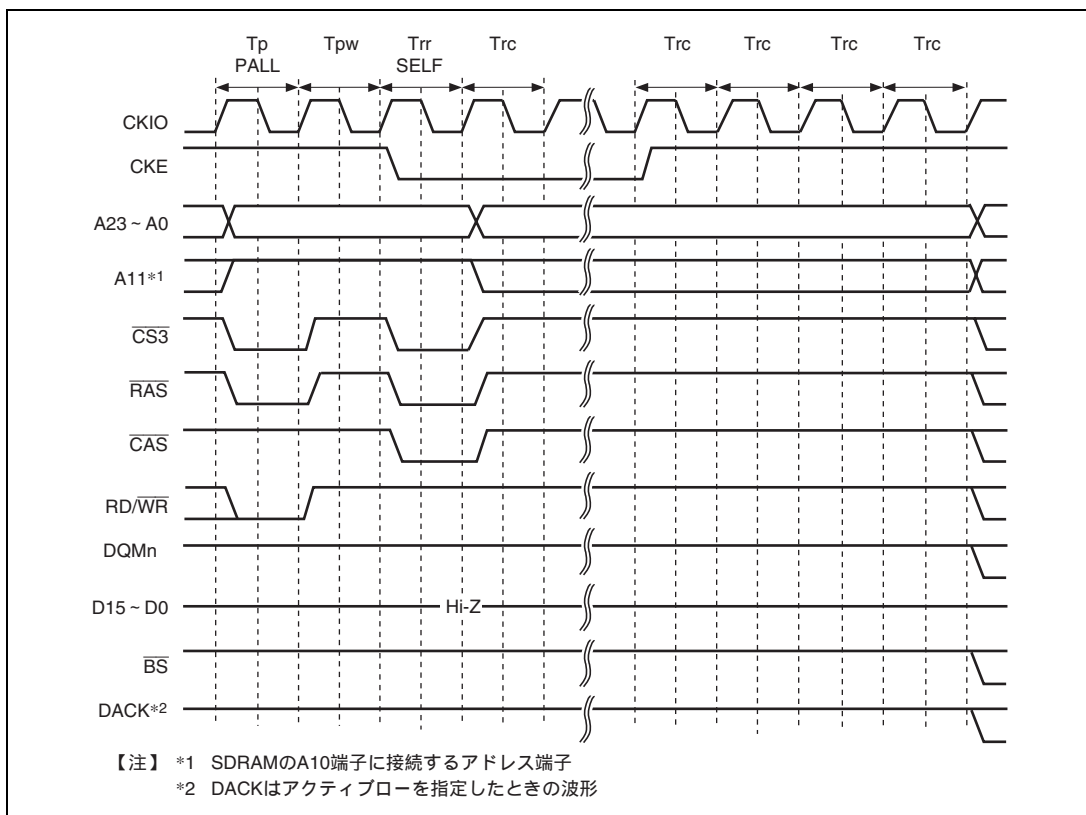


図 9.24 セルフリフレッシュタイミング

(9) リフレッシュ要求とバスサイクルの関係

バスサイクル実行中にリフレッシュ要求が発生した場合は、リフレッシュの実行はバスサイクルの完了まで待たされます。また、バスアービトレーション機能でバスを解放しているときにリフレッシュ要求が発生した場合は、バス権を獲得するまでリフレッシュの実行は待たされます。本 LSI はリフレッシュの実行が待たされている間にバス権を要求する信号として、 $\overline{\text{REFOUT}}$ 端子を設けています。バス権を獲得するまで、 $\overline{\text{REFOUT}}$ をアサート (ローレベル) し続けます。

リフレッシュの実行を待たされている状態で新たなリフレッシュ要求が発生した場合には、前のリフレッシュ要求は消滅します。リフレッシュを正しく行うためには、リフレッシュ間隔よりも長いバスサイクルや、バス権の占有が起らないようにする必要があります。

セルフリフレッシュ中にバス権要求が発生しても、バスの解放はセルフリフレッシュが解除されるまで行われません。

9. バスステートコントローラ (BSC)

(10) 低周波数モード

SDCR の SLOW ビットを 1 に設定すると、コマンド、アドレス、ライトデータの出力および、リードデータの取り込みを、SDRAM を低周波数で動作させるのに適したタイミングで行います。

図 9.25 に低周波数モードでのアクセスタイミングを示します。このモードでは、コマンド、アドレス、ライトデータを通常より半サイクル遅い CKIO の立ち下がりに同期して出力します。また、リードデータを通常より半サイクル早い CKIO の立ち下がりで取り込みます。これにより、コマンド、アドレス、ライトデータ、リードデータのホールド時間を延長することができます。

SLOW ビットを 1 に設定して高周波数で SDRAM を動作させると、コマンド、アドレス、ライトデータ、リードデータのセットアップ時間が確保できなくなる可能性があります。SLOW ビットの設定は、動作周波数やボードのタイミング設計を考慮して決定してください。

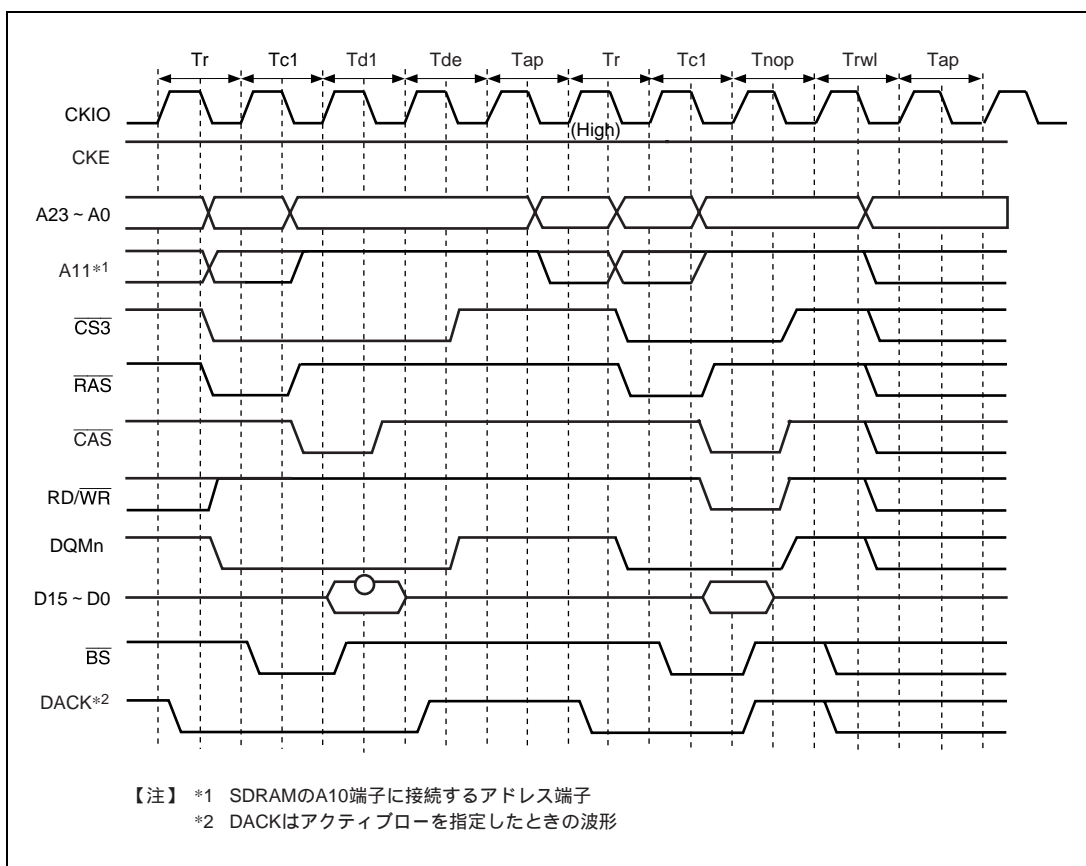


図 9.25 低周波数モードでのアクセスタイミング

(11) パワーダウンモード

SDCR の PDOWN ビットを 1 に設定すると、非アクセス時には CKE を L レベルにして SDRAM をパワーダウンモードに移行させます。これにより非アクセス時の消費電力を大幅に抑えることができます。パワーダウンモード中のアクセス発生時には SDRAM のパワーダウンモードを解除するために CKE 信号をアサートするサイクルが挿入されるため、1 サイクルのオーバーヘッドが発生します。また、SDRAM アクセス終了後は、SDRAM 以外のメモリのアクセスを行うことで、再度パワーダウンモードに移行できます (CKE が L レベルになります)。図 9.26 にパワーダウンモードでのアクセスタイミングを示します。

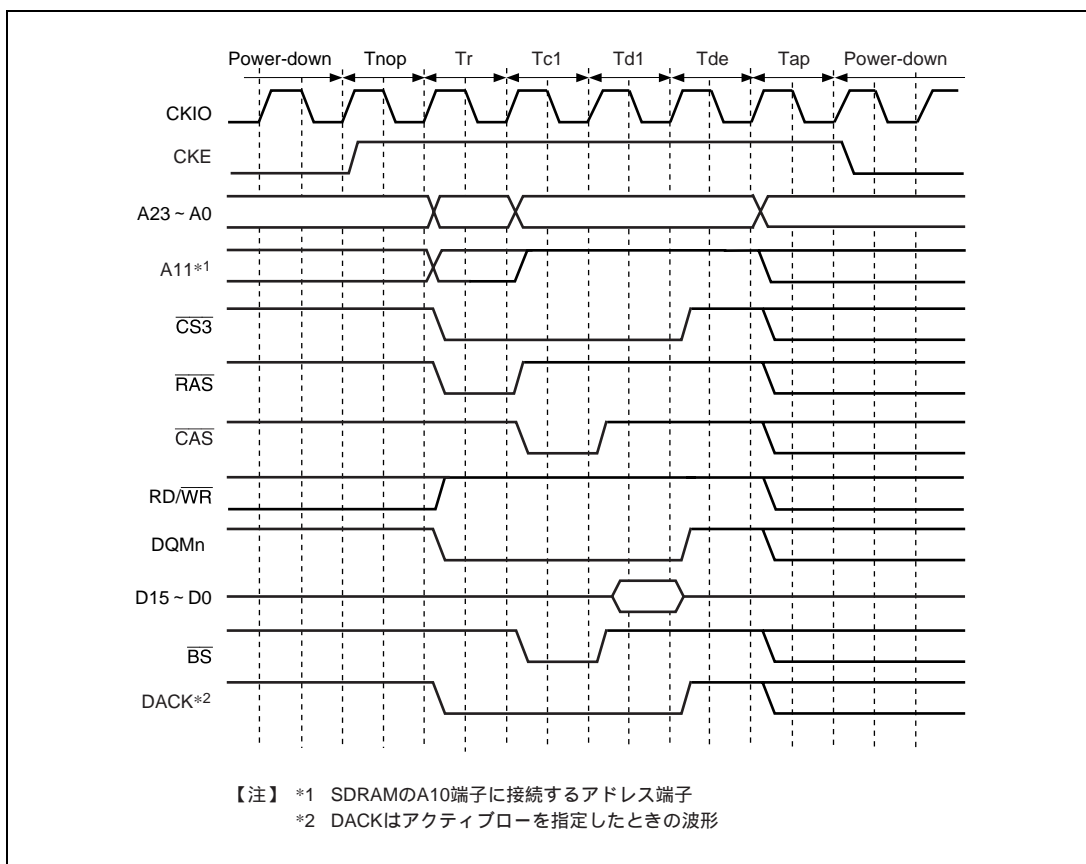


図 9.26 パワーダウンモードでのアクセスタイミング

9. バスステートコントローラ (BSC)

(12) パワーオンシーケンス

SDRAMを使用するためには、パワーオン後、SDRAMに対してモード設定を行う必要があります。SDRAMの初期化を正しく行うためには、まずBSCのレジスタを設定した後、SDMR3レジスタをアクセスすることによりSDRAMのモードレジスタに対する書き込みを行います。SDRAMのモードレジスタの設定は $\overline{CS3}$ 、 \overline{RAS} 、 \overline{CAS} 、および $\overline{RD}/\overline{WR}$ の組み合わせで、その時点のアドレス信号の値がSDRAMに取り込まれます。設定したい値をXとすると $X + H'A4FD5000$ 番地にワードライトを行うことによって、値XがSDRAM内のモードレジスタに書き込まれます。このときライトデータは、無視されます。本LSIでサポートしているバーストリード/シングルライト(バースト長1)またはバーストリード/バーストライト(バースト長1)、CASレイテンシ2~3、ラップタイプ=シーケンシャル、およびバースト長1を設定するには、表9.11に示すアクセスアドレスに任意のデータをワードライトします。このとき、外部アドレス端子のA12以上のビットには0が出力されます。

表 9.11 SDRAM モードレジスタライト時のアクセスアドレス

• エリア3設定 (SDMR3)

バーストリード/シングルライト(バースト長1)の場合

データバス幅	CAS レイテンシ	アクセスアドレス	外部アドレス端子
16 ビット	2	H'A4FD5440	H'0000440
	3	H'A4FD5460	H'0000460

バーストリード/バーストライト(バースト長1)の場合

データバス幅	CAS レイテンシ	アクセスアドレス	外部アドレス端子
16 ビット	2	H'A4FD5040	H'0000040
	3	H'A4FD5060	H'0000060

モードレジスタ設定タイミングを図9.27に示します。まず全バンクプリチャージコマンド(PALL)を発行し、次いでオートリフレッシュコマンド(REF)を8回発行します。そして最後に、モードレジスタ書き込みコマンド(MRS)を発行します。PALLと1回目のREFの間にCS3WCRのTRP[1:0]ビットに設定した数のアイドルサイクルが挿入され、REFとREFおよび、8回目のREFとMRSの間にCS3WCRのTRC[1:0]ビットに設定した数のアイドルサイクルが挿入されます。また、MRSと次に発行するコマンドの間に1サイクル以上のアイドルサイクルが挿入されます。

SDRAMは全バンクプリチャージ(PALL)に先立って、電源投入後に一定のアイドル時間を確保しなければなりません。必要なアイドル時間は、ご使用になるSDRAMのマニュアルを参照してください。リセット信号のパルス幅がこのアイドル時間より長い場合には、直ちにモードレジスタの設定を行っても問題ありませんが、短い場合は注意する必要があります。

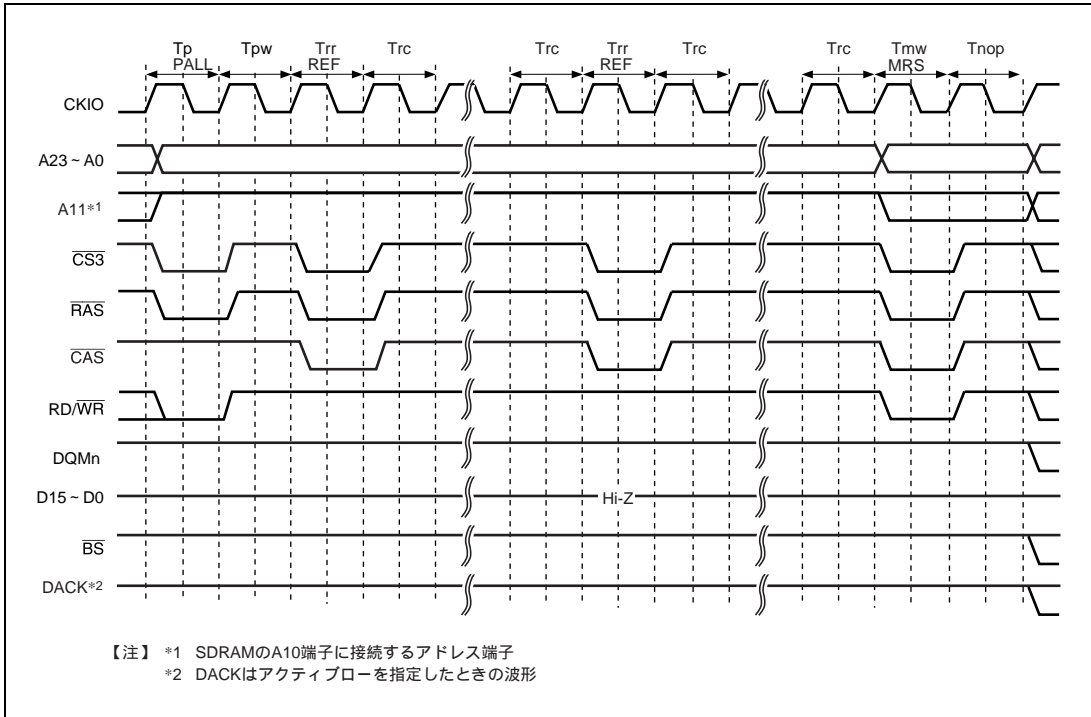


図 9.27 SDRAM モードレジスタ書き込みタイミング (JEDEC 準拠)

9.5.6 バースト ROM インタフェース

バースト ROM インタフェースは、バーストモードあるいはページモード等と呼ばれるアドレスの切り替えによって、高速に読み出しのできる機能を有するメモリをアクセスするためのものです。基本的には通常空間インタフェースと同じようなアクセスを行います。最初のサイクルを終了する際に \overline{RD} 信号のネゲートを行わず、アドレスのみを切り替えて、2 回目以降のアクセスを行います。2 回目以降のアクセスでは、アドレスの変化が CKIO 信号の立ち下がりになります。

最初のアクセスサイクルに対しては、CSnWCR の W[3:0]ビットに設定された数のウェイトサイクルが挿入されます。2 回目以降のアクセスサイクルに対しては、CSnWCR の BW[1:0]ビットに設定された数のウェイトサイクルが挿入されます。

バースト ROM アクセス時は、 \overline{BS} 信号は最初のアクセスサイクルに対してのみアサートされます。また、外部ウェイト入力も最初のアクセスサイクルにのみ有効です。

バースト ROM インタフェースでバースト動作を行わないシングルアクセスおよびライトアクセス時は、通常空間インタフェースと同じアクセスタイミングになります。

表 9.12 にデータバス幅およびアクセスサイズとバースト数の関係を、図 9.28 にタイムチャートを示します。

9. バスステートコントローラ (BSC)

表 9.12 データバス幅およびアクセスサイズとバースト数の関係

データバス幅	アクセスサイズ	バースト数	アクセス回数
8 ビット	8 ビット	1	1
	16 ビット	2	1
	32 ビット	4	1
	16 バイト	16	1
16 ビット	8 ビット	1	1
	16 ビット	1	1
	32 ビット	2	1
	16 バイト	8	1

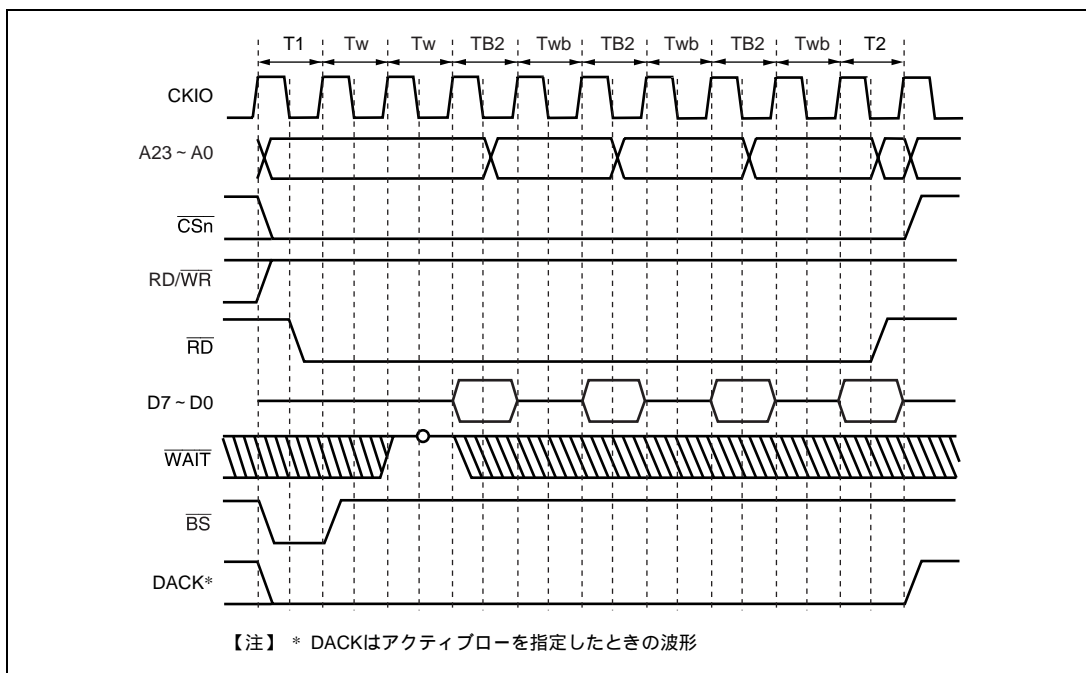


図 9.28 バースト ROM アクセス
 (データバス幅 8 ビット 4 バイト転送 (バースト数 4)、
 初回アクセスウェイト 2、2 回目以降アクセスウェイト 1)

9.5.7 バイト選択付き SRAM インタフェース

(1) 基本アクセスタイミング

バイト選択付き SRAM インタフェースは、リードまたはライトいずれのバスサイクルでもバイト選択端子 ($\overline{WE_n}$) を出力するメモリインタフェースです。このインタフェースは 16 ビットのデータ端子を持ち、UB ある

いはLBのような上位バイト選択端子および下位バイト選択端子のあるSRAMをアクセスするためのものです。

CSnWCRのBASビットが0(初期値)のとき、バイト選択付きSRAMインタフェースのライトアクセスタイミングは、通常空間インタフェースと同一です。一方、リード動作では、 \overline{WEn} 端子のタイミングが通常空間インタフェースと異なり、 \overline{WEn} 端子からバイト選択信号を出力します。図9.29に基本アクセスタイミングを示します。特にライト時は、バイト選択端子(\overline{WEn})のタイミングでメモリに書き込まれます。ご使用になるメモリのデータシートをご確認ください。

CSnWCRのBASビットが1のとき、 \overline{WEn} 端子とRD/ \overline{WR} 端子のタイミングが変化します。図9.30に基本アクセスタイミングを示します。特にライト時は、ライトイネーブル端子(RD/ \overline{WR})のタイミングでメモリに書き込まれます。RD/ \overline{WR} のネグートタイミングからのライトデータのホールドタイミングは、CSnWCRのHW[1:0]ビットを設定することにより確保してください。図9.31にバイト選択付きSRAMの接続例を示します。

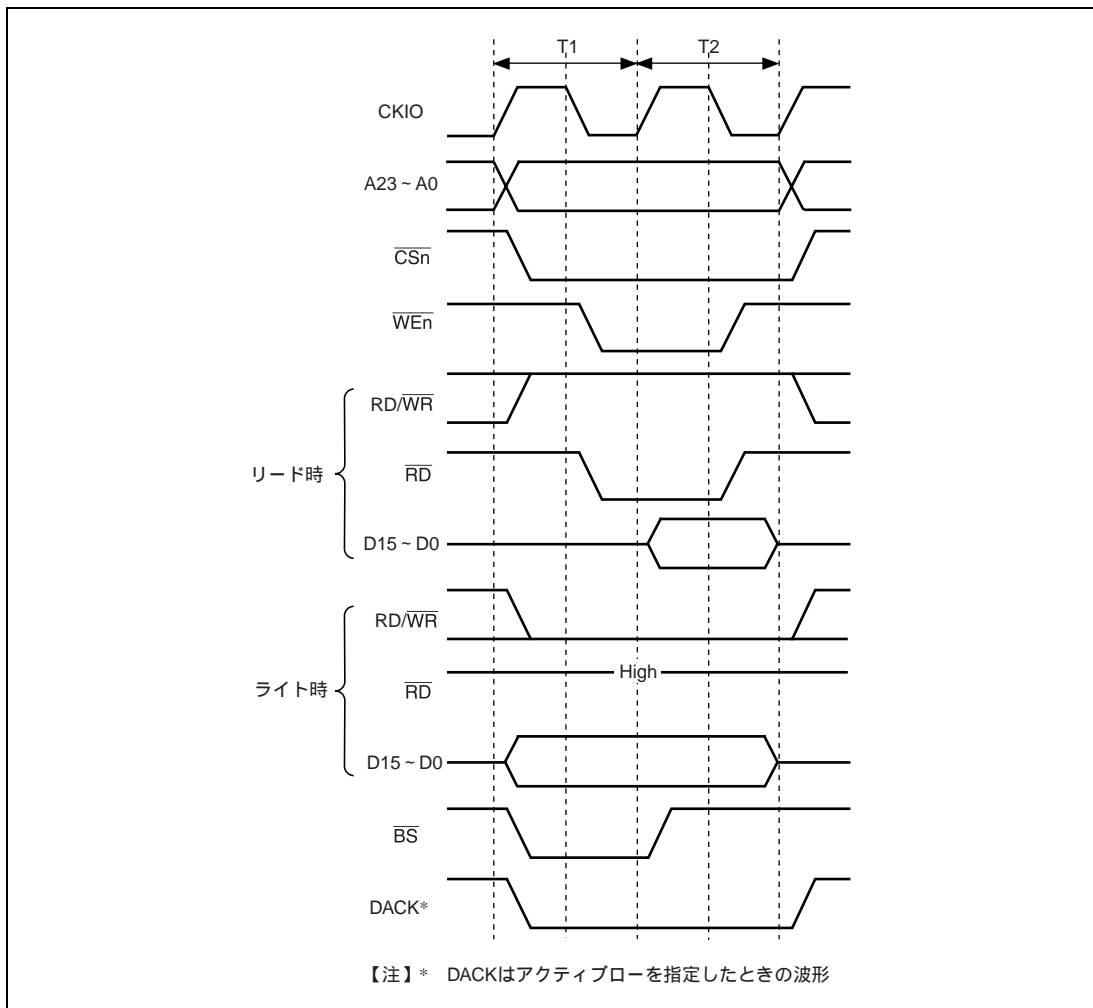


図 9.29 BAS=0 バイト選択付きSRAM 基本アクセスタイミング

9. バスステートコントローラ (BSC)

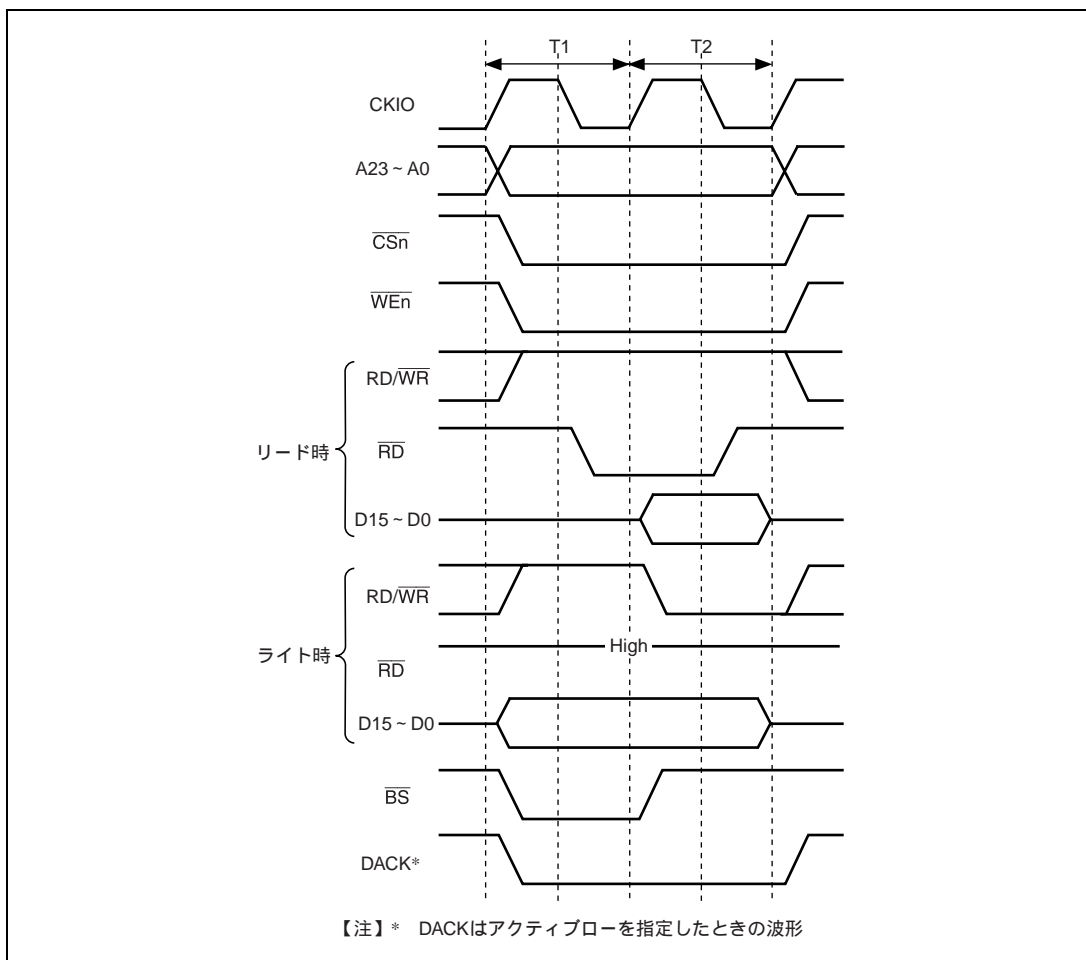


図 9.30 BAS=1 バイト選択付き SRAM 基本アクセスタイミング

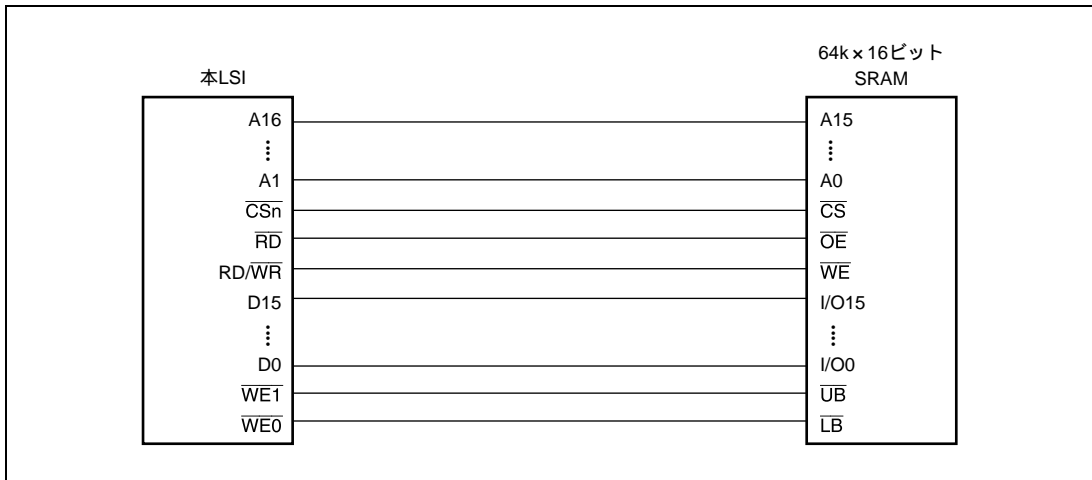


図 9.31 16 ビットデータ幅バイト選択付き SRAM 接続例

(2) バイト選択付き SRAM インタフェースの制限事項

オートリフレッシュ中の SDRAM 使用時は、バイト選択付き SRAM インタフェースにおいて BAS ビットを 1 に設定しないでください。この場合、リフレッシュが発行できなくなります。

【不具合現象】

リフレッシュ前にバイト選択付き (BAS = 1) のアクセスがあった場合、リフレッシュシーケンス先頭の PALL コマンドサイクルにおいて、RD/WR のローレベルが出力されず ACTV コマンドになってしまいます。本現象により、その後の REF コマンドから SDRAM の動作が保証できなくなります。

【回避策】

1. SDRAM と混在させる疑似 SRAM は、UB、LB コントロールライトサポート品を使用してください。
2. フラッシュメモリ混載の MCP を使用するために、MCP の WE 端子に本 LSI の RD/WR 端子を入力する必要がある場合は、以下の手順でアクセスしてください。
 - SDRAM をセルフリフレッシュに遷移
 - フラッシュメモリ接続エリアの CSnBCR、CSnWCR をバイト選択付き SRAM および WE ライトコントロール (BAS = 1) に設定
 - フラッシュメモリへのライトアクセスを開始

フラッシュメモリへのライトアクセス終了後、SDRAM のセルフリフレッシュを解除する場合は、必ず UB、LB コントロール (BAS = 0) に設定してからセルフリフレッシュを解除してください。

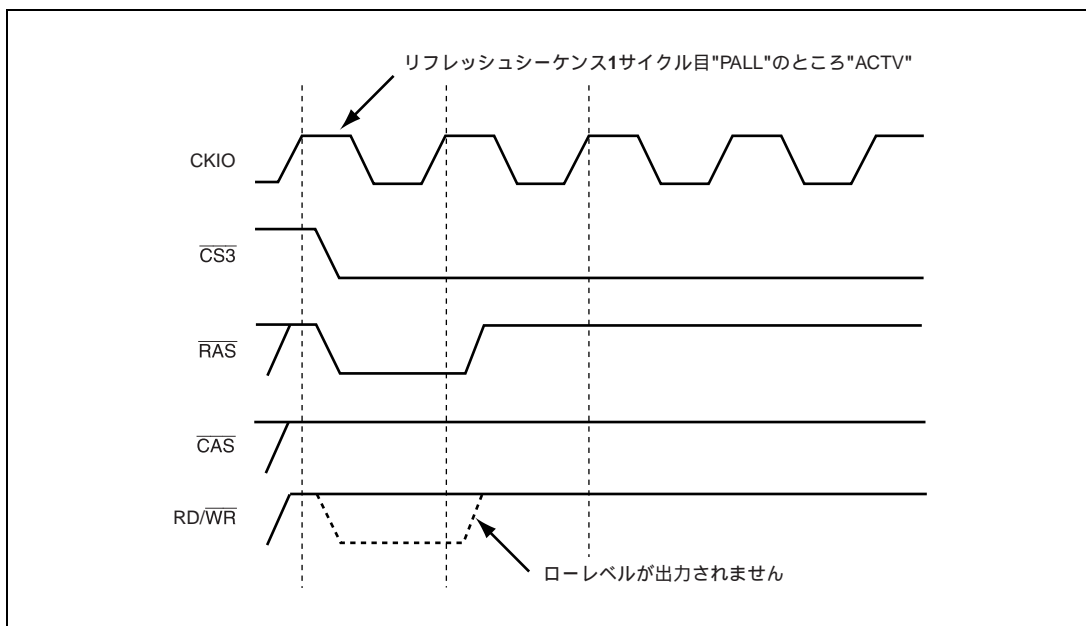


図 9.32 不具合発生時波形

9.5.8 アクセスサイクル間ウェイト

本 LSI の動作周波数が高くなってきたため、低速なデバイスからのデータ出力が完了した際のデータバッファのオフが間に合わず、次のデバイスのデータ出力と衝突してデバイスの信頼度を低下させたり、誤動作を引き起こす場合があります。これを防止するため、連続するアクセス間にアクセスサイクル間のウェイトを挿入して、データの衝突を回避する機能を設けました。

アクセスサイクル間ウェイトのサイクル数は、CSnBCR の IWW[1:0]ビット、IWRWD[1:0]ビット、IWRWS[1:0]ビット、IWRRD[1:0]ビット、IWRRS[1:0]、および CMNCR の DMAIW[1:0]ビット、DMAIWA ビットで指定します。アクセスサイクル間ウェイト (アイドルサイクル) は、以下の条件のとき挿入が可能です。

1. 連続するアクセスがライト - リード、ライト - ライトの場合
2. 連続するアクセスが別空間でかつリード - ライトの場合
3. 連続するアクセスが同一空間でかつリード - ライトの場合
4. 連続するアクセスが別空間でかつリード - リードの場合
5. 連続するアクセスが同一空間でかつリード - リードの場合
6. DMAシングル転送による外部デバイスによるデータ出力サイクル後の本LSIを含む別デバイスによるデータ出力の場合 (DMAIWA=0)
7. DMAシングル転送による外部デバイスによるデータ出力サイクル後にアクセス発生の場合 (DMAIWA=1)

9.5.9 バスアービトレーション

バスアービトレーションでは、通常状態でバス権を有し、他のデバイスからのバス権要求を受けてバスの解放を行う場合と、通常状態でバス権を持っておらず、外部アクセスが必要なたびにバス権要求する場合があります。(本マニュアルでは、通常状態でバス権を持っている場合をマスタモードと呼びます。)本 LSI は、マスタモードをサポートしています。

マスタとスレーブとの間でバス権を受け渡す際、接続されているデバイスの誤動作を防ぐため、バス解放に先立ってすべてのバス制御信号をネゲート状態にします。バス権を受け取る場合にも、バス制御信号はネゲート状態からドライブを開始します。バス権を受け渡すマスタとスレーブで同じ値に信号をドライブするので、出力バッファの衝突は回避できます。バス制御信号のハイインピーダンス状態での外来ノイズによる誤動作を防ぐために、これらの制御信号にはブルアップ抵抗が必要となります。

バス権の受け渡しはバスサイクルの切れ目で行われます。バス権を要求されたとき、バスサイクルを行っていない場合は、直ちにバス権の解放を行います。バスサイクル最中の場合は、バスサイクルが完了するまで待ち、バス権の解放を行います。LSI 外部から見るとバスサイクルを行っていない場合でも、アクセスサイクル間ウェイトを挿入するなど、内部的にはバスサイクルが開始されている場合があるため、 $\overline{\text{CSn}}$ 信号その他のバス制御信号を見て、直ちにバスが解放されるかどうかを判断することはできません。バス解放が行われない状態を以下に示します。

1. キャッシュミスによる16バイト転送中
2. キャッシュのコピーバック中
3. TAS命令のリードサイクルとライトサイクル間
4. データバス幅がアクセスサイズよりも小さいことによって生じる複数のバスサイクル(たとえば8ビットデータバス幅のメモリにロングワードアクセスを行う場合のバスサイクル間)
5. DMACでの16バイト転送時

$\overline{\text{BREQ}}$ および $\overline{\text{BACK}}$ のハンドシェイクによる外部バス解放には、少なからずオーバーヘッドが存在します。スレーブモジュールのタスクが多い場合は、一度のバス権獲得により、複数回のバスサイクルを実行するようにシステム設計を行なってください。バス権の受け渡しに必要なサイクルを減らすことによりシステム設計が容易になります。

(1) マスタモード

マスタモードでは、バス権要求を受けない限りバス権を保有しています。外部からのバス権要求 $\overline{\text{BREQ}}$ のアサート(ローレベル)を受け、実行中のバスサイクルが終わり次第バスの解放を行い、バス使用許可 $\overline{\text{BACK}}$ をアサート(ローレベル)します。スレーブがバスを解放したことを示す $\overline{\text{BREQ}}$ のネゲート(ハイレベル)を受けて $\overline{\text{BACK}}$ をネゲート(ハイレベル)し、バスの使用を再開します。

SDRAM は、アクティブなバンクがある場合、全バンクプリチャージコマンド (PALL) を発行し、これを完了させた後バスの解放を行います。

具体的なバス解放シーケンスは、次のとおりです。

9. バスステートコントローラ (BSC)

1. CKIOの立ち上がりに同期して、アドレスバスおよびデータバスをハイインピーダンスにします。
2. 0.5サイクル後にバス使用許可信号をCKIOの立ち下がりに同期してアサートします。
3. これに続くCKIOの立ち上がりで、バス制御信号 (\overline{BS} 、 \overline{CSn} 、 \overline{RAS} 、 \overline{CAS} 、 \overline{WE} / \overline{DQMn} 、 \overline{RD} 、および $\overline{RD}/\overline{WR}$) をハイインピーダンスにします。これらのバス制御信号は、遅くともハイインピーダンスにする1サイクル前には、ハイレベルにされています。
4. バス権要求信号のサンプリングは、CKIOの立ち下がりで行います。

スレーブからバス権を再獲得するときのシーケンスは、次のとおりです。

1. \overline{BREQ} のネゲートをCKIOの立ち下がりで検出すると、1.5サイクル後にバス制御信号は、ハイレベルでドライブを開始します。
2. これに続くクロックの立ち下がりで、バス使用許可信号をネゲートします。アドレスバスおよびデータバスのドライブを開始するのは、これに続くCKIOの立ち上がりです。
3. バス制御信号をアサートしてバスサイクルを実際に開始するのは、最も早い場合にはアドレスおよびデータ信号をドライブするのと同じクロックの立ち上がりからです。

マスタモードのバスアービトレーションタイミングを図 9.33 に示します。

ユーザが個別に設計したスレーブによっては、アービトレーションによるオーバーヘッドを減少させるため、連続して複数回のバスアクセスを発生しようとする場合があります。このようなケースでは、SDRAM の確実なリフレッシュを行うため、スレーブのバス占有時間がリフレッシュ周期を超えることなく、バス権を解放するように設計を行う必要があります。そこで、本 LSI では、リフレッシュの実行が待たされている間にバス権を要求する信号として \overline{REFOUT} 端子を用意しています。そして、リフレッシュの実行が待たされている間バス権を獲得するまで、 \overline{REFOUT} をアサート (ローレベル) します。外部のスレーブデバイスがこれを受けてバス権を解放すれば、バス権が本 LSI に戻り、リフレッシュを実行することができます。

また、 \overline{BREQ} 、 \overline{BACK} のハンドシェイクによる外部バス解放には、少なからずオーバーヘッドが存在します。スレーブモジュールのタスクが多い場合は一度のバス権獲得により、複数回のバスサイクルを実行するようにシステム設計を行ってください。バス権の受け渡しに必要なサイクルを減らすことによりシステム設計が容易になります。

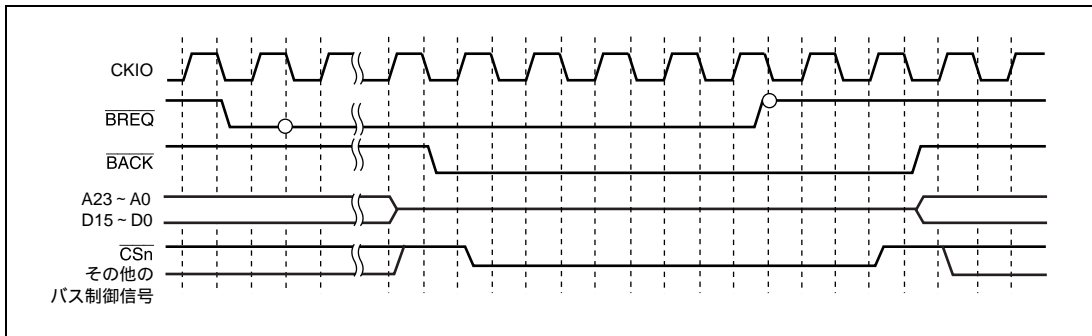


図 9.33 バスアービトレーション (マスタモード)

(2) マスタとスレーブの協調

マスタとスレーブの接続例を図 9.34 に示します。

マスタとスレーブで矛盾なくシステムリソースを制御するためには、役割分担を適切に行う必要があります。SDRAM は、その使用に先立って初期化を行わなければなりません。また、低消費電力を実現するためのスタンバイ動作を行う場合にも、役割分担を行う必要があります。

本 LSI では、初期化、リフレッシュ、スタンバイなどのすべての制御をマスタモードのデバイスが行うように設計されています。2 プロセッサ構成のマスタ・スレーブ直接接続のときは、メモリへのアクセスを除くすべての処理をマスタが行います。スレーブ側のプロセッサが SDRAM のように、使用に先立って初期化が必要なデバイスを初期化が完了する前にアクセスしないようにする必要があります。具体的な方法としては、スレーブのリセット解除をマスタから行うように外部回路を設ける方法、初期化の必要のない SRAM などに初期化終了後にマスタがフラグをセットすることとしスレーブはこのフラグがセットされたことを確認してからアクセスを開始する方法、マスタからスレーブに割り込みがかけられるように外部回路を設け初期化終了後にマスタからスレーブに割り込みをかけることによってスレーブを待機状態から解放する方法などがあります。

SDRAM をセルフリフレッシュモードにした場合は、マスタはバスの解放を行いません。また、スタンバイモードへの遷移や、周波数変更のためにマスタのクロックが停止する場合は、マスタはバスの解放を行うことができません。このとき、スレーブからのバス権要求が発生しないように、スレーブをスリープさせる等スレーブのアクセスサイクルが発生しないようにしてください。

9. バスステートコントローラ (BSC)

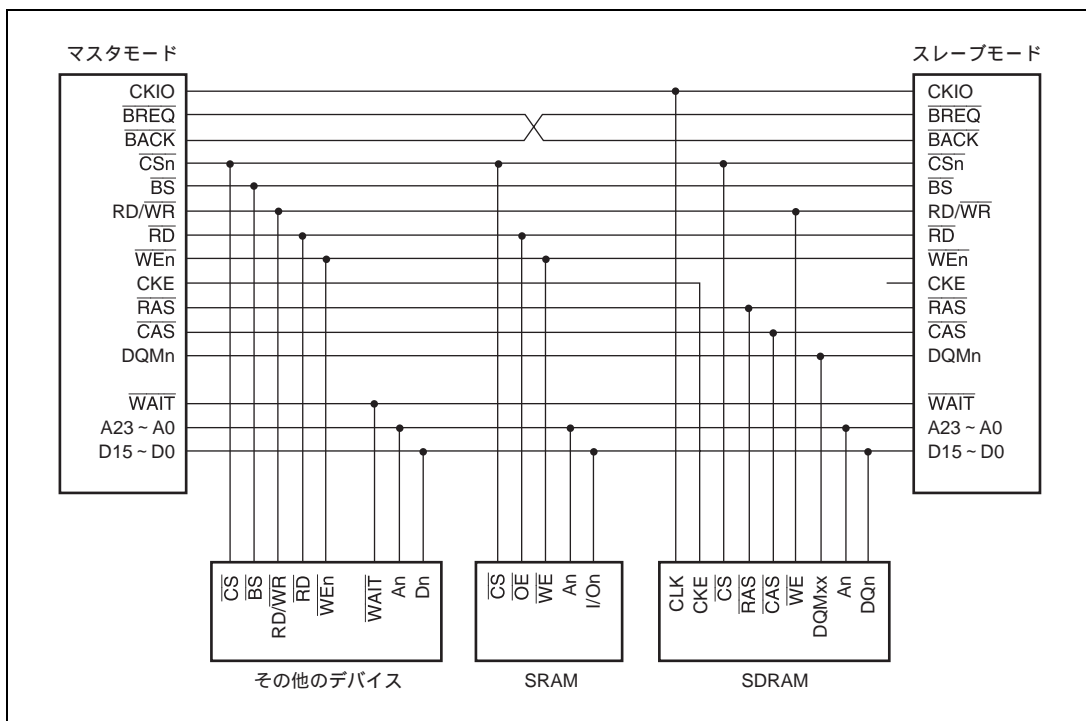


図 9.34 マスタとスレープの接続例

9.5.10 使用上の注意事項

(1) リセット

BSC は、パワーオンリセットでのみ完全な初期化が行われます。パワーオンリセット時には、バスサイクルの途中であるなしに関わらず直ちにすべての信号をネゲートし、出力バッファをオフにします。また、制御レジスタはすべて初期化されます。スタンバイ、スリープ、およびマニュアルリセットでは、バスステートコントローラの制御レジスタの初期化は一切行われません。マニュアルリセットが行われると、現在実行中のバスサイクルはそのバスサイクルに限り終了まで実行され、その後はアクセス待ちの状態になります。キャッシュおよびその他の LSI 内部バスマスタモジュールの 16 バイト転送を実行中の場合は、バスマスタがマニュアルリセットによってアクセス要求を取り消すため、ロングワード単位でアクセスが打ち切られます。このためキャッシュフィル時にマニュアルリセットを入れる場合は、キャッシュの内容は保証されなくなりますので注意してください。マニュアルリセット中も RTCNT のカウントアップが行われるためリフレッシュ要求が発生し、リフレッシュサイクルが起動されます。また、マニュアルリセット中でも BREQ によるアービトレーション要求を受け付けます。

フラッシュメモリには、リセット解除後から最初のアクセスまでの最小時間を規定しているものがあります。バスステートコントローラは、この最小時間を確保するために、7 ビットのカウンタ (RWTCNT) を用意しています。パワーオンリセットによりこのカウンタは、0 クリアされます。パワーオンリセット解除後は CKIO 信号に同期してカウントアップされ、カウンタの値が H'007F になるまで外部アクセスを発生しません。マニュアルリセ

ットでは、カウンタの値はクリアされません。

(2) LSI 内部バスマスタからみたアクセス

本 LSI の内部は、キャッシュバス (L バス)、内部バス (I バス)、および周辺バスの 3 つのバスに分割されています。CPU およびキャッシュメモリはキャッシュバスに、CPU を除く内部バスマスタおよび BSC は内部バスに、低速な周辺モジュールは周辺バスにそれぞれ接続されています。また、キャッシュメモリ以外の内蔵メモリと UBC、AUD 等のデバッグモジュールはキャッシュバスと内部バスの双方に接続されています。キャッシュバスから内部バスのアクセスは行えますが、逆は行えません。このため以下のようなことが発生します。

CPU を除く内部バスマスタ (DMAC 等) からキャッシュメモリ以外の内蔵メモリへのアクセスは行えますが、キャッシュメモリへのアクセスは行えません。CPU を除く内部バスマスタにより外部メモリへの書き込みが行われ、その結果として外部メモリの内容とキャッシュの内容に食い違いが発生することがあります。CPU を除く内部バスマスタにより外部メモリへの書き込みが行われた場合、その番地へのデータがキャッシュにある可能性があるときには、キャッシュメモリのページをソフトウェアで行う必要があります。

CPU が読み出しアクセスを開始し、それがキャッシュ領域の場合はキャッシュの検索が行われます。キャッシュにデータが保持されている場合は、これを取りこみアクセスは完了します。キャッシュ内にデータが無い場合には、内部バスを介してキャッシュデータのフィルを行うため、4 つの連続したロングワードリードが起動されます。バイトまたはワードオペランドアクセス時および奇数ワード境界 ($4n+2$) への分岐時のミスヒットに関して、本 LSI 外部インタフェース上は必ず 4 つのロングワードアクセスでフィルを行います。キャッシュスルー領域に関しては、実際のアクセスアドレスに従ってアクセスを行います。アクセスが偶数ワード境界 ($4n$) への命令フェッチの場合にはロングワードアクセス、奇数ワード境界 ($4n+2$) への命令フェッチの場合にはワードアクセスとなります。

非キャッシュ領域および内蔵周辺モジュールの読み出しサイクルの場合は、その判定後内部バスを介して読み出しサイクルが起動されます。読み出しデータは、キャッシュバスを経由して CPU に送られます。

書き込みサイクルがキャッシュ領域に対するものであった場合は、キャッシュのライト方式により動作が異なります。

ライトバックモード時は、キャッシュの検索を行い該当アドレスのデータがあった場合にはキャッシュに書き込みを行います。実際のメモリへの書き込みは、該当アドレスの置き換えが発生するまで行われません。該当アドレスのデータが無かった場合には、キャッシュの更新が行われます。まず置き換え対象となるデータを内部バッファへ退避し、次に該当アドレスのデータを含む 16 バイトのデータ読み出しを行い、該当アドレスのデータを更新します。それに続き、最初に退避した 16 バイトのデータの書き戻しサイクルが行われます。

ライトスルーモード時は、キャッシュの検索を行い該当アドレスのデータがあった場合にはキャッシュへの書き込みと並行して内部バスを経由して実際の書き込みが行われます。該当アドレスのデータが無かった場合には、キャッシュの更新は行わずに内部バスを経由して実際の書き込みのみ行われます。

BSC には一段のライトバッファがあるため、ライトサイクルでは本 LSI 外部のバスサイクルが完了しなくても内部バスを別のアクセスに使用することができます。本 LSI 外部の低速メモリに対して書き込みを行った後に、内蔵周辺モジュールに対する読み出しまたは書き込みを行う場合は、低速メモリへの書き込みの完了を待たずに内蔵周辺モジュールへのアクセスが可能です。

9. バスステートコントローラ (BSC)

読み出しでは、常に動作の完了まで CPU は待たされるので、実際のデバイスに対するデータの書き込みが完了したことを確認してから処理を続行したい場合は、続けて同じアドレスに対するダミーの読み出しアクセスを行うと書き込みの終了を確認できます。

DMAC 等の別のバスマスタからのアクセスでも同様に BSC のライトバッファは働きます。したがって、デュアルアドレスの DMA 転送を行う場合は、書き込みサイクルの完了を待たずに次の読み出しサイクルの起動がかけられます。ただし、DMA のソースアドレスとディスティネーションアドレスがともに外部メモリ空間である場合には、前の書き込みサイクルが完了するまで次の読み出しサイクルの開始は待たされます。

(3) 内蔵周辺モジュールのアクセス

内蔵周辺モジュールのレジスタへのアクセスは、内部バスから周辺クロック (P) で 2 サイクル以上かかりま
す。システム設計の見積もり時には、ご注意ください。

10. ダイレクトメモリアクセスコントローラ (DMAC)

本 LSI は、ダイレクトメモリアクセスコントローラ (DMAC) を内蔵しています。DMAC は、DACK (転送要求受け付け信号) 付き外部デバイス、外部メモリ、内蔵メモリ、メモリマップト外部デバイス、内蔵周辺モジュール間のデータ転送を、CPU に代わって高速に行うことができます。

10.1 特長

- チャンネル数：4チャンネル (うち1チャンネルは外部リクエスト受け付け可能)
外部リクエストはch0で受付可能です。
- アドレス空間：アーキテクチャ上は4Gバイト
- 転送データ長：バイト、ワード (2バイト)、ロングワード (4バイト)、16バイト (ロングワード×4)
- 最大転送回数：16,777,216 (24ビット) 回
- アドレスモード：シングルアドレスモードとデュアルアドレスモードから選択可能
- 転送要求：
外部リクエスト、内蔵周辺モジュールリクエスト、オートリクエストの3種類から選択可能
内蔵周辺モジュールリクエストを発行できるものは以下のモジュールがあります。
SCIF0、SCIF1、SIOF、USBF
- バスモード：サイクルスチールモード (通常モードとインターミッテントモード) とバーストモードから選択可能
- 優先順位：チャンネル優先順位固定モードとラウンドロビンモードから選択可能
- 割り込み要求：データ転送終了時にCPUへ割り込み要求を発生可能
- 外部リクエスト検出：DREQ入力のローまたはハイレベル検出、および立ち上がりまたは立ち下がりエッジ検出から選択可能
- 転送要求受付信号・転送終了信号：DACKおよびTENDはアクティブレベルを設定可能

10. ダイレクトメモリアクセスコントローラ (DMAC)

DMACのブロック図を図10.1に示します。

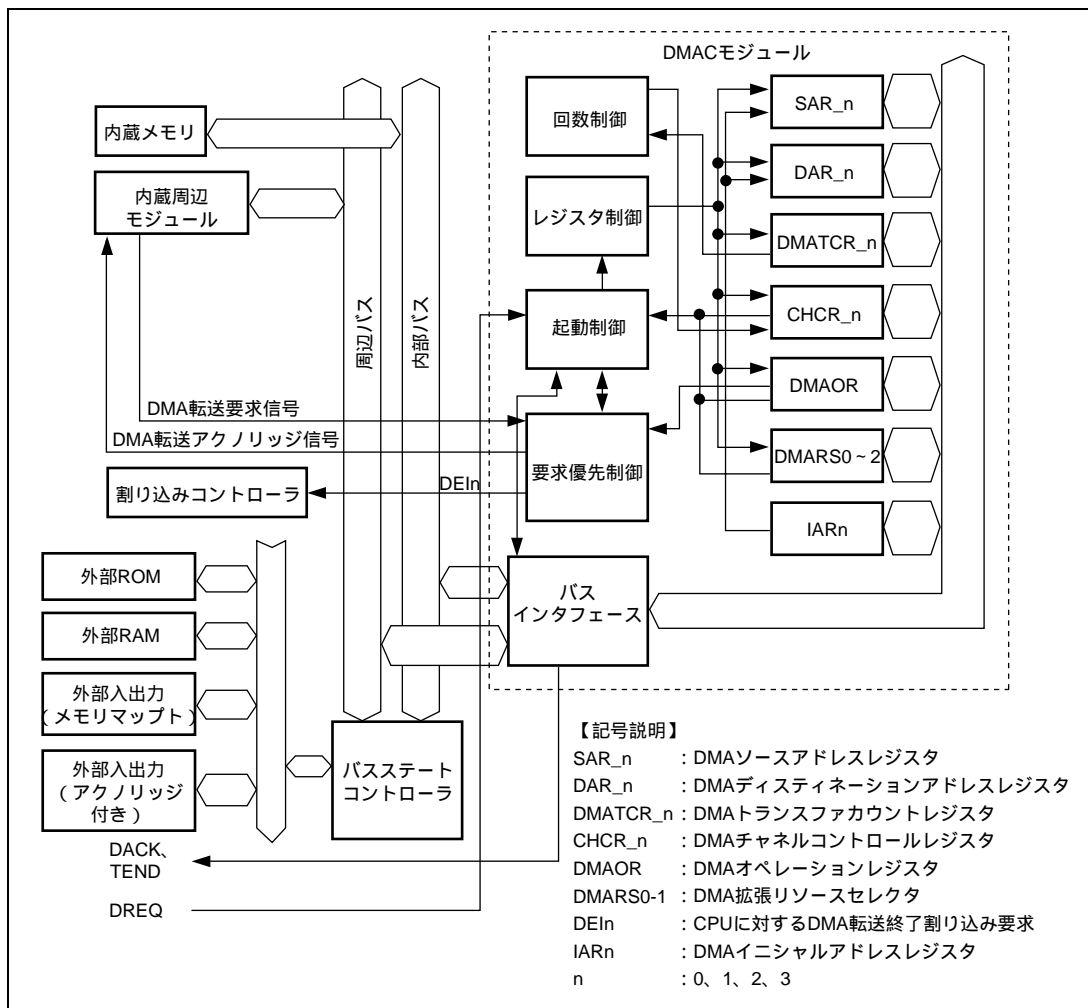


図 10.1 DMAC ブロック図

10.2 入出力端子

DMACの端子構成を表10.1に示します。

表 10.1 端子構成

チャンネル	端子名	入出力	機能
0	DREQ	入力	DMA 転送要求 外部デバイスからチャンネル0へのDMA 転送要求入力
	DACK	出力	DMA 転送要求受付 DMAC チャンネル0から外部デバイスへのDMA 転送要求受付出力
	TEND	出力	DMA 転送終了 DMAC チャンネル0のDMA 転送終了出力

10.3 レジスタの説明

DMACには以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第27章 レジスタ一覧」を参照してください。各チャンネルのレジスタについては、チャンネル0のSARはSAR_0のように表記しています。

(1) チャンネル0

- DMAソースアドレスレジスタ_0 (SAR_0)
- DMAディスティネーションアドレスレジスタ_0 (DAR_0)
- DMAトランスファカウントレジスタ_0 (DMATCR_0)
- DMAチャンネルコントロールレジスタ_0 (CHCR_0)
- DMAイニシャルアドレスレジスタ_0 (IAR_0)

(2) チャンネル1

- DMAソースアドレスレジスタ_1 (SAR_1)
- DMAディスティネーションアドレスレジスタ_1 (DAR_1)
- DMAトランスファカウントレジスタ_1 (DMATCR_1)
- DMAチャンネルコントロールレジスタ_1 (CHCR_1)
- DMAイニシャルアドレスレジスタ_1 (IAR_1)

(3) チャンネル2

- DMAソースアドレスレジスタ_2 (SAR_2)
- DMAディスティネーションアドレスレジスタ_2 (DAR_2)
- DMAトランスファカウントレジスタ_2 (DMATCR_2)

10. ダイレクトメモリアクセスコントローラ (DMAC)

- DMAチャンネルコントロールレジスタ₂ (CHCR₂)
- DMAイニシャルアドレスレジスタ₂ (IAR₂)

(4) チャンネル3

- DMAソースアドレスレジスタ₃ (SAR₃)
- DMAディスティネーションアドレスレジスタ₃ (DAR₃)
- DMAトランスファカウントレジスタ₃ (DMATCR₃)
- DMAチャンネルコントロールレジスタ₃ (CHCR₃)
- DMAイニシャルアドレスレジスタ₃ (IAR₃)

(5) 共通

- DMAオペレーションレジスタ (DMAOR)
- DMA拡張リソースセクタ0 (DMARS0)
- DMA拡張リソースセクタ1 (DMARS1)

10.3.1 DMA ソースアドレスレジスタ (SAR)

SAR は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送元のアドレスを指定します。DMA 転送中は、次の転送元アドレスを示しています。シングルアドレスモードにおいて、転送元が DACK 付きの外部デバイスの転送をする場合には SAR は無視されます。

16 ビット、32 ビット幅のデータ転送を行う場合は、それぞれ、16 ビット、32 ビット境界のアドレスを指定してください。16 バイト単位で転送を行う場合は、16 バイト境界に値を設定してください。

リセット時の値は不定になり、スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持します。

10.3.2 DMA ディスティネーションアドレスレジスタ (DAR)

DAR は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送先のアドレスを指定します。DMA 転送中は、次の転送先アドレスを示しています。シングルアドレスモードにおいて、転送先が DACK 付きの外部デバイスの転送をする場合には DAR は無視されます。

16 ビット、32 ビット幅のデータ転送を行う場合は、それぞれ、16 ビット、32 ビット境界のアドレスを指定してください。16 バイト単位で転送を行う場合は、16 バイト境界に値を設定してください。

リセット時の値は不定になり、スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持します。

10.3.3 DMA トランスファカウントレジスタ (DMATCR)

DMATCR は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送回数を指定します。転送回数は、設定値が H'00000001 のときは 1 回、H'00FFFFFF のときは 16,777,215 回で、H'00000000 のときは 16,777,216 回(最大転送回数)になります。DMA 転送中は、残りの転送回数を示しています。

DMATCR の上位 8 ビットは、読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を

書いた場合の動作の保証はできません。

16 バイト転送のときは、16 バイト転送 1 回 (128 ビット) で 1 回のカウントをします。

リセット時の値は不定になり、スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持します。

10.3.4 DMA チャンネルコントロールレジスタ (CHCR)

CHCR は、読み出し / 書き込み可能な 32 ビットのレジスタで、DMA 転送モードを制御します。

リセット時は初期化され、スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持します。

ビット	ビット名	初期値	R/W	説明
31~27	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
26	RPT	0	R	リピートモード DMA 転送をリピートモードで行うか、通常モードで行うかを指定します。 0 : DMA 転送を通常モードで行う。 1 : DMA 転送をリピートモードで行う。
25	RAS	0	R	リピート領域対象 リピート領域を転送元とするか、転送先とするかを設定します。 0 : 転送先をリピート領域とする。 1 : 転送元をリピート領域とする。
24	DA	0	R	DMA アシンクロナス DREQ が同期信号か、非同期信号かを指定します。 本ビットは CHCR_0 でのみ有効です。CHCR_1~3 ではリザーブビットとなります。書き込む値は常に 0 にしてください。1 を書いた場合の動作は保証できません。 0 : DREQ は非同期信号 1 : DREQ は同期信号
23	DO	0	R/W	DMA オーバランビット DREQ をオーバラン 0 で検出するか、オーバラン 1 で検出するかを選択します。 本ビットは CHCR_0 でのみ有効です。CHCR_1~3 ではリザーブビットとなります。書き込む値は常に 0 にしてください。1 を書いた場合の動作は保証できません。 0 : DREQ をオーバラン 0 で検出 1 : DREQ をオーバラン 1 で検出

10. ダイレクトメモリアクセスコントローラ (DMAC)

ビット	ビット名	初期値	R/W	説明
22	TL	0	R/W	<p>トランスファエンドレベル</p> <p>TEND 信号をハイアクティブにするかローアクティブにするかを指定します。</p> <p>本ビットは CHCR_0 でのみ有効です。CHCR_1~3 ではリザーブビットとなります。書き込む値は常に 0 にしてください。1 を書いた場合の動作は保証できません。</p> <p>0 : TEND をローアクティブ出力 1 : TEND をハイアクティブ出力</p>
21~18	-	すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>
17	AM	0	R/W	<p>アクノリッジモード</p> <p>デュアルアドレスモードで、DACK をデータ読み出しサイクルで出力するか、書き込みサイクルで出力するかを選択します。</p> <p>シングルアドレスモード時は、本ビットの指定に関係なく DACK は常に出力されます。</p> <p>本ビットは CHCR_0 でのみ有効です。CHCR_1~3 ではリザーブビットとなります。書き込む値は常に 0 にしてください。1 を書いた場合の動作は保証できません。</p> <p>0:読み出しサイクルで DACK を出力(デュアルアドレスモード) 1:書き込みサイクルで DACK を出力(デュアルアドレスモード)</p>
16	AL	0	R/W	<p>アクノリッジレベル</p> <p>DACK 信号をハイアクティブにするかローアクティブにするかを指定します。</p> <p>本ビットは CHCR_0 でのみ有効です。CHCR_1~3 ではリザーブビットとなります。書き込む値は常に 0 にしてください。1 を書いた場合の動作は保証できません。</p> <p>0 : DACK をローアクティブ出力 1 : DACK をハイアクティブ出力</p>

10. ダイレクトメモリアクセスコントローラ (DMAC)

ビット	ビット名	初期値	R/W	説明
15 14	DM1 DM0	0 0	R/W R/W	<p>ディスティネーションアドレスモード</p> <p>DMA 転送先のアドレスの増減を指定します (シングルアドレスモードにおいて、DACK 付き外部デバイスへの転送をする場合には、DM1、DM0 ビットは無視されます)。</p> <p>00: ディスティネーションアドレスは固定 (16 バイト単位転送時は設定禁止)</p> <p>01: ディスティネーションアドレスは増加 (バイト単位転送時は +1、ワード単位転送時は +2、ロングワード単位転送時は +4、16 バイト単位転送時は +16)</p> <p>10: ディスティネーションアドレスは減少 (バイト単位転送時は -1、ワード単位転送時は -2、ロングワード単位転送時は -4、16 バイト単位転送時は設定禁止)</p> <p>11: 設定禁止</p>
13 12	SM1 SM0	0 0	R/W R/W	<p>ソースアドレスモード</p> <p>DMA 転送元のアドレスの増減を指定します (シングルアドレスモードにおいて、DACK 付き外部デバイスから転送をする場合には、SM1、SM0 ビットは無視されます)。</p> <p>00: ソースアドレスは固定 (16 バイト単位転送時は設定禁止)</p> <p>01: ソースアドレスは増加 (バイト単位転送時は +1、ワード単位転送時は +2、ロングワード単位転送時は +4、16 バイト転送時は +16)</p> <p>10: ソースアドレスは減少 (バイト単位転送時は -1、ワード単位転送時は -2、ロングワード単位転送時は -4、16 バイト転送時は設定禁止)</p> <p>11: 設定禁止</p>

10. ダイレクトメモリアクセスコントローラ (DMAC)

ビット	ビット名	初期値	R/W	説明	
11	RS3	0	R/W	リソースセレクト	
10	RS2	0	R/W	DMAC に入力する転送要求元を指定します。	
9	RS1	0	R/W	転送要求元の変更は、必ず DMA イネーブルビット (DE) が 0 の状態で行ってください。 0000 : 外部リクエスト、デュアルアドレスモード 0001 : 設定禁止 0010 : 外部リクエスト、シングルアドレスモード 外部アドレス空間 DACK 付き外部デバイス 0011 : 外部リクエスト、シングルアドレスモード DACK 付き外部デバイス 外部アドレス空間 0100 : オートリクエスト 0101 : 設定禁止 0110 : 設定禁止 0111 : 設定禁止 1000 : DMA 拡張リソースセクタ 1001 : 設定禁止 1010 : 設定禁止 1011 : 設定禁止 1100 : 設定禁止 1101 : 設定禁止 1110 : 設定禁止 1111 : 設定禁止 【注】外部リクエストの指定は CHCR_0 のみ有効です。CHCR_1 ~ 3 では、外部リクエスト指定は設定できません。	
8	RS0	0	R/W		
7	DL	0	R/W		DREQ レベル (DL)、DREQ セレクト (DS)
6	DS	0	R/W		DREQ 入力の検出方法と検出レベルを選択します。 本ビットは CHCR_0 のみ有効です。CHCR_1 ~ 3 ではリザーブビットとなります。書き込む値は常に 0 にしてください。1 を書いた場合の動作は保証できません。 またチャンネル 0 でも転送要求元を内蔵周辺モジュール、またはオートリクエストに指定した場合、本ビットは無効です。 00 : ローレベル検出 01 : 立ち下がりエッジ検出 10 : ハイレベル検出 11 : 立ち上がりエッジ検出

10. ダイレクトメモリアクセスコントローラ (DMAC)

ビット	ビット名	初期値	R/W	説明
5	TB	0	R/W	<p>トランスファバスモード</p> <p>DMA 転送のバスモードを選択します。</p> <p>0 : サイクルスチールモード</p> <p>1 : パーストモード</p>
4 3	TS1 TS0	0 0	R/W R/W	<p>トランスファサイズ</p> <p>DMA 転送の単位を選択します。</p> <p>転送元または転送先が、転送サイズが指定された内蔵周辺モジュールのレジスタの場合には、必ずその転送サイズを選んでください。</p> <p>00 : バイト単位</p> <p>01 : ワード (2 バイト) 単位</p> <p>10 : ロングワード (4 バイト) 単位</p> <p>11 : 16 バイト単位転送 (ロングワード 4 回転送)</p>
2	IE	0	R/W	<p>インタラプトイネーブル</p> <p>DMA 転送終了時に CPU に割り込み要求するかどうかを指定します。</p> <p>IE ビットを 1 にセットした場合、TE ビットがセットされると、CPU に対し割り込み (DEI) を要求します。</p> <p>0 : 割り込み要求を禁止</p> <p>1 : 割り込み要求を許可</p>
1	TE	0	R/(W)*	<p>トランスファエンドフラグ</p> <p>DMA トランスファカウントレジスタ (DMATCR) の値が 0 になり、DMA 転送が終了すると、TE ビットは 1 にセットされます。</p> <p>DMATCR が 0 にならないときに、NMI 割り込み、DMA アドレスエラーによって転送が終了した場合、および DE ビット、DMA オペレーションレジスタ (DMAOR) の DME ビットをクリアして転送を終了させた場合には、TE ビットはセットされません。TE ビットをクリアするには、TE ビットの 1 を読み出してから 0 を書き込みます。</p> <p>TE ビットがセットされていると、DE ビットを 1 にしていても転送は許可されません。</p> <p>0 : DMA 転送中または DMA 転送中断</p> <p>【クリア条件】TE ビットの 1 を読み出してから 0 を書き込む。</p> <p>1 : (DMATCR=0 により) DMA 転送終了</p>

10. ダイレクトメモリアクセスコントローラ (DMAC)

ビット	ビット名	初期値	R/W	説明
0	DE	0	R/W	DMA イネーブル DMA 転送を許可または禁止します。 オートリクエストモードでは、DE ビットおよび DMAOR の DME ビットを 1 にセットすると転送を開始します。ただし、TE ビット、DMAOR の NMIF ビット、AE ビットのすべてが 0 であることが必要です。外部リクエスト、周辺モジュールリクエストでは、DE ビットと DME ビットを 1 にセットした後で、さらに該当デバイスまたは周辺モジュールから DMA 転送要求があると転送を開始します。ただし、この場合にもオートリクエストモードと同じく、TE ビット、NMIF ビット、AE ビットのすべてが 0 であることが必要です。DE ビットを 0 にクリアすると、転送を中断することができます。 0 : DMA 転送を禁止 1 : DMA 転送を許可

【注】 * フラグをクリアするための 0 書き込みのみ可能です。

10.3.5 DMA イニシャルアドレスレジスタ (IAR)

IAR は、読み出し / 書き込み可能な 32 ビットレジスタで、DMA 転送をリピートするときに SAR または DAR に設定するアドレスの初期値を指定します。

16 ビット、32 ビット幅のデータ転送を行う場合は、それぞれ、16 ビット、32 ビット境界のアドレスを指定してください。16 バイト単位で転送を行う場合は、16 バイト境界に値を設定してください。

リセット時の値は不定になり、スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持しません。

10.3.6 DMA オペレーションレジスタ (DMAOR)

DMAOR は、読み出しまたは書き込み可能な 32 ビットのレジスタで、DMA 転送時のチャネルの優先順位を指定します。また、DMA の転送状態 (ステータス) も示します。

リセット時の値は不定になり、スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持します。

ビット	ビット名	初期値	R/W	説明
31, 30	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
29	CMS1	0	R/W	サイクルスチールモードセレクト サイクルスチールモード時に通常モードとインターミットモードを選択します。 インターミットモードを有効にするためには、全チャネルのバスモードがサイクルスチールモードであることが必要です。 00: 通常モード 01: 設定禁止 10: インターミットモード 16 外部バスクロック 16 クロックに 1 回 DMA 転送を実行 11: インターミットモード 64 外部バスクロック 64 クロックに 1 回 DMA 転送を実行
28	CMS0	0	R/W	
27, 26	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
25	PR1	0	R/W	プライオリティーモード 同時に複数のチャネルに転送要求があった場合に、実行するチャネルの優先順位を決定するビットです。 00: 固定モード 1: CH0 > CH1 > CH2 > CH3 01: 固定モード 2: CH0 > CH2 > CH3 > CH1 10: チャネル選択ラウンドロビンモード: RCn ビットの状態が優先順位に反映されます。 11: 全チャネルラウンドロビンモード
24	PR0	0	R/W	
23~19	-	0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

10. ダイレクトメモリアクセスコントローラ (DMAC)

ビット	ビット名	初期値	R/W	説明
18	AE	0	R/(W)*	<p>アドレスエラーフラグ</p> <p>DMAC によるアドレスエラーが発生したことを示します。</p> <p>AE ビットがセットされると、CHCR の DE ビットと DMAOR の DME ビットを 1 にセットしても、DMA 転送は許可されません。</p> <p>AE ビットをクリアするには、AE ビットの 1 を読み出してから 0 を書き込みます。</p> <p>0 : DMAC によるアドレスエラーなし</p> <p>【クリア条件】AE ビットの 1 を読み出してから 0 を書き込む。</p> <p>1 : DMAC によるアドレスエラー発生</p>
17	NMIF	0	R/(W)*	<p>NMI フラグ</p> <p>NMI 割り込みが発生したことを示します。</p> <p>NMIF ビットがセットされると、CHCR の DE ビットと DMAOR の DME ビットを 1 にセットしても、DMA 転送は許可されません。</p> <p>NMIF ビットをクリアするには、NMIF ビットの 1 を読み出してから 0 を書き込みます。</p> <p>NMI が入力されたとき、実行中の DMA 転送の 1 転送単位までは行われず、DMAC が動作していないときに、NMI 割り込みが入力されても、NMIF ビットは 1 にセットされます。</p> <p>0 : NMI 割り込みなし</p> <p>【クリア条件】NMIF ビットの 1 を読み出してから 0 を書き込む。</p> <p>1 : NMI 割り込み発生</p>
16	DME	0	R/W	<p>DMA マスタイネーブル</p> <p>すべてのチャンネルの DMA 転送を許可または禁止します。</p> <p>DME ビットおよび CHCR の DE ビットを 1 にセットすると、DMA 転送が許可されます。ただし、転送を行うチャンネルの CHCR にある TE ビットと DMAOR の NMIF ビット、AE ビットのすべてが 0 である必要があります。DME ビットをクリアすると、すべてのチャンネルの DMA 転送が中断されます。</p> <p>0 : 全チャンネルの DMA 転送を禁止</p> <p>1 : 全チャンネルの DMA 転送を許可</p>
15~0		すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>

【注】 * フラグをクリアするための 0 書き込みのみ可能です。

10.3.7 DMA 拡張リソースセクタ 0、1 (DMARS0、DMARS1)

DMARS は、読み出しまたは書き込み可能な 16 ビットレジスタで、各チャンネルごとに周辺モジュールからの DMA 転送要求元を指定します。DMARS0 はチャンネル 0 および 1、DMARS1 はチャンネル 2 および 3 を設定します。

リセット時は初期化され、スタンバイモードおよびモジュールスタンバイ機能使用時には値を保持します。

各モジュールからの転送要求は、表 10.2 の MID、RID を設定します。表 10.2 以外の MID / RID を設定したときの動作は保証できません。DMARS レジスタからの転送要求は、CHCR_0~3 レジスタのリソースセレクトビット (RS3~0) =B'1000 に設定したときのみ有効です。B'1000 以外の場合は、DMARS を設定しても転送要求元として受け付けられません。

- DMARS0 の設定

ビット	ビット名	初期値	R/W	説明
15	C1MID5	0	R/W	DMA チャンネル 1 転送要求元モジュール ID5~0 (MID) 表 10.2 参照
14	C1MID4	0	R/W	
13	C1MID3	0	R/W	
12	C1MID2	0	R/W	
11	C1MID1	0	R/W	
10	C1MID0	0	R/W	
9	C1RID1	0	R/W	DMA チャンネル 1 転送要求元レジスタ ID1~0 (RID) 表 10.2 参照
8	C1RID0	0	R/W	
7	C0MID5	0	R/W	DMA チャンネル 0 転送要求元モジュール ID5~0 (MID) 表 10.2 参照
6	C0MID4	0	R/W	
5	C0MID3	0	R/W	
4	C0MID2	0	R/W	
3	C0MID1	0	R/W	
2	C0MID0	0	R/W	
1	C0RID1	0	R/W	DMA チャンネル 0 転送要求元レジスタ ID1~0 (RID) 表 10.2 参照
0	C0RID0	0	R/W	

10. ダイレクトメモリアクセスコントローラ (DMAC)

• DMARS1の設定

ビット	ビット名	初期値	R/W	説明
15	C3MID5	0	R/W	DMA チャンネル 3 転送要求元モジュール ID5~0 (MID) 表 10.2 参照
14	C3MID4	0	R/W	
13	C3MID3	0	R/W	
12	C3MID2	0	R/W	
11	C3MID1	0	R/W	
10	C3MID0	0	R/W	
9	C3RID1	0	R/W	DMA チャンネル 3 転送要求元レジスタ ID1~0 (RID) 表 10.2 参照
8	C3RID0	0	R/W	
7	C2MID5	0	R/W	DMA チャンネル 2 転送要求元モジュール ID5~0 (MID) 表 10.2 参照
6	C2MID4	0	R/W	
5	C2MID3	0	R/W	
4	C2MID2	0	R/W	
3	C2MID1	0	R/W	
2	C2MID0	0	R/W	
1	C2RID1	0	R/W	DMA チャンネル 2 転送要求元レジスタ ID1~0 (RID) 表 10.2 参照
0	C2RID0	0	R/W	

表 10.2 DMARS の設定

周辺モジュール	1 チャンネル分の 設定値 (MID+RID)	MID	RID	機能
SCIF0	H'21	B'001000	B'01	送信
	H'22		B'10	受信
SCIF1	H'29	B'001010	B'01	送信
	H'2A		B'10	受信
SIOF	H'55	B'010101	B'01	送信
	H'56		B'10	受信
USBF	H'73	B'011100	B'11	送受信 0
	H'70		B'00	送受信 1

10.4 動作説明

DMAC は DMA 転送要求があると決められたチャネルの優先順位に従って転送を開始し、転送終了条件が満たされると転送を終了します。転送要求にはオートリクエスト、外部リクエスト、内蔵周辺モジュールリクエストの3種類のモードがあります。バスモードはバーストモードとサイクルスチールモードを選択することができます。

10.4.1 転送フロー

DMA ソースアドレスレジスタ (SAR)、DMA デスティネーションアドレスレジスタ (DAR)、DMA トランスファカウントレジスタ (DMATCR)、DMA チャネルコントロールレジスタ (CHCR)、DMA オペレーションレジスタ (DMAOR)、DMA 拡張リソースセレクト (DMARS) に目的の転送条件設定後、DMAC は以下の順序でデータを転送します。

1. 転送許可状態かどうか (DE=1、DME=1、TE=0、AE=0、NMIF=0) をチェックします。
2. 転送許可状態で転送要求が発生すると1転送単位のデータ (TS0、TS1ビットの設定により決定) を転送します。オートリクエストモードの場合は、DEビットおよびDMEビットが1にセットされると自動的に転送を開始します。1回の転送を行うごとにDMATCRの値を1デクリメントします。具体的な転送フローは、アドレスモード、バスモードにより異なります。
3. 指定された回数の転送を終える (DMATCRの値が0になる) と、転送を正常に終了します。このときCHCRのIEビットに1がセットしてあれば、CPUにDEI割込みが発生します。ただし、RPT=1を設定しておく、SARもしくはDARにIARの値が設定されDMATCRが書き込まれた初期値に戻り、TEをクリアするだけで続けて転送を実行できます。
4. DMACによるアドレスエラーかNMI割込みが発生した場合には、転送を中断します。またCHCRのDEビットかDMAORのDMEビットを0にしても転送を中断します。

図 10.2 に上記のフローチャートを示します。

10. ダイレクトメモリアクセスコントローラ (DMAC)

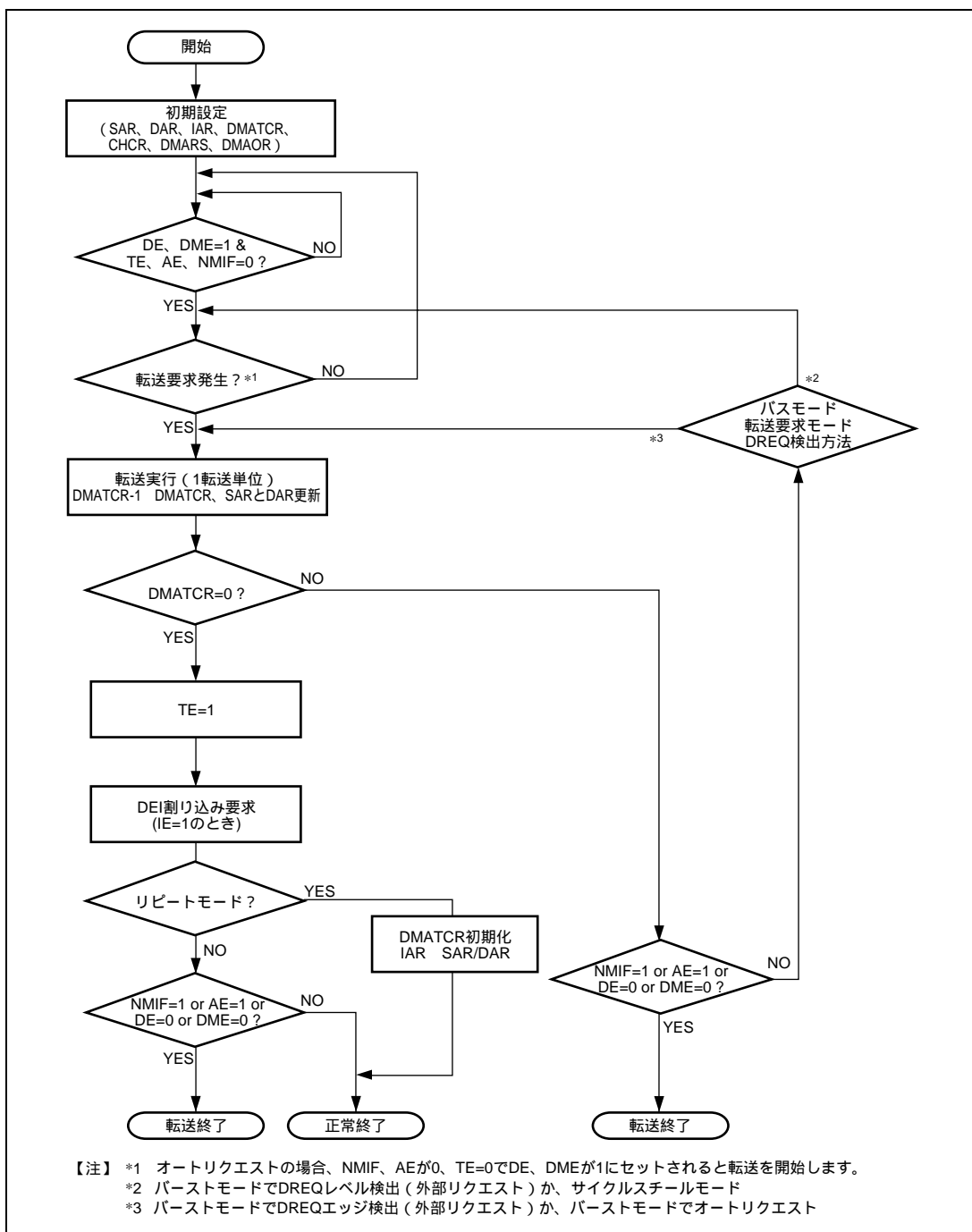


図 10.2 DMA 転送フローチャート

10.4.2 リピートモード転送

DMACのリピートモードは、CHCR_n レジスタの RPT ビットに 1 を設定することによって利用することができます。リピートモードが設定されて、転送が終了する(DMATCR_n が 0 になる)と、CHCR_n の IE ビットが設定してあれば DE1 割り込みを発生させた上で、CHCR_n の RAS ビットで指定された SAR_n もしくは DAR_n に IAR_n の値をコピーし、DMATCR_n を初期値に戻します。CHCR_n レジスタの TE ビットをクリアすると、連続して転送を実行することができます。

10.4.3 DMA 転送要求

DMA 転送要求はデータの転送元または転送先に発生させるのが基本的な使い方ですが、転送元でも転送先でもない外部デバイスや内蔵周辺モジュールに発生させる使い方でもできます。

転送要求にはオートリクエスト、外部リクエスト、内蔵周辺モジュールリクエストの 3 種類があります。転送要求の選択は DMA チャンネルコントロールレジスタ 0~3 (CHCR_0~3) の RS3~RS0 ビットおよび DMARS0、DMARS1 レジスタによって行います。

(1) オートリクエストモード

オートリクエストモードはメモリ同士の転送や、転送要求を発生できない内蔵周辺モジュールとメモリ転送のように、転送要求信号が外部から来ない場合に、DMAC 内部で自動的に転送要求信号を発生するモードです。CHCR_0~3 の DE ビットおよび DMAOR の DME ビットを 1 にセットすると転送が開始されます。ただし CHCR_0~3 の TE ビット、DMAOR の AE ビット、NMIF ビットがすべて 0 である必要があります。

(2) 外部リクエストモード

外部リクエストモードは、LSI の外部デバイスからの転送要求信号 (DREQ) によって転送を開始させるモードです。DMA チャンネル 0 のみ有効です。システムに応じて表 10.3 に示すモードの中から 1 つを選んで使います。DMA 転送が許可されているとき (DE=1、DME=1、TE=0、AE=0、NMIF=0) に DREQ が入力されると DMA 転送が開始されます。

表 10.3 RS ビットによる外部リクエストモードの選択

RS3	RS2	RS1	RS0	アドレスモード	転送元	転送先
0	0	0	0	デュアルアドレスモード	任意	任意
0	0	1	0	シングルアドレスモード	外部メモリまたは メモリマップト外部デバイス	DACK 付き外部デバイス
			1		DACK 付き外部デバイス	外部メモリまたは メモリマップト外部デバイス

DREQ をエッジで検出するかレベルで検出するかは、表 10.4 に示す CHCR_0 の DREQ レベル (DL) ビットと DREQ セレクト (DS) ビットで選択します。転送要求元は必ずしもデータの転送元か転送先である必要はありません。

10. ダイレクトメモリアクセスコントローラ (DMAC)

表 10.4 DL、DS ビットによる外部リクエスト検出の選択

CHCR		外部リクエスト検出方法
DL	DS	
0	0	ローレベル検出
	1	立ち下がり検出
1	0	ハイレベル検出
	1	立ち上がり検出

DREQ が受け付けられると DREQ 端子は要求受け付け不可能状態 (不感帯) となります。受け付けた DREQ に対するアクノリッジ DACK を出力した後、再び DREQ 端子は要求を受け付けることが可能になります。

DREQ をレベル検出で使う場合、DACK を出力して次の DREQ を検出するタイミングによって、リクエストと同じ回数の転送を実行して中断する場合 (オーバーラン 0) と、リクエストより 1 つ多い回数の転送を実行して中断する場合 (オーバーラン 1) があります。オーバーランを 0 にするか 1 にするかは、CHCR の DO ビットで選択します (表 10.5)。

表 10.5 DL、DS ビットによる外部リクエスト検出の選択

CHCR	外部リクエスト
DO	
0	オーバーラン 0
1	オーバーラン 1

(3) 内蔵周辺モジュールリクエストモード

内蔵周辺モジュールリクエストモード (表 10.6) では、内蔵周辺モジュールからの DMA 転送要求信号によって転送が実行されます。

DMA 転送要求信号は、SCIF0、SCIF1、USBF からの送信データエンプティ転送要求と受信データフル転送要求および SIOF からの送信データ転送リクエストと受信データ転送リクエストがあります。

内蔵周辺モジュールリクエストモード選択時に、DMA 転送許可状態 (DE=1、DME=1、TE=0、AE=0、NMIF=0) ならば、転送要求信号によって転送が実行されます。

転送要求を SCIF の送信データエンプティ転送要求に設定した場合、転送先を当該 SCIF のトランスミットデータレジスタとする必要があります。同様に転送要求を SCIF の受信データフル転送要求に設定した場合、転送元を当該 SCIF のレシーブデータレジスタとする必要があります。これらは SIOF、USBF も同様です。

10. ダイレクトメモリアクセスコントローラ (DMAC)

表 10.6 RS3~0 ビットによる内蔵周辺モジュールリクエストモードの選択

CHCR RS[3:0]	DMARS		DMA 転送要求元	DMA 転送要求信号	転送元	転送先	バスモード
	MID	RID					
1000	001000	01	SCIF0 送信部	TDFE (送信 FIFO データエンプティ)	任意	SCFTDR_0	サイクル スチール
		10	SCIF0 受信部	RDF (受信 FIFO データフル)	SCFRDR_0	任意	サイクル スチール
	001010	01	SCIF1 送信部	TDFE (送信 FIFO データエンプティ)	任意	SCFTDR_1	サイクル スチール
		10	SCIF1 受信部	RDF (受信 FIFO データフル)	SCFRDR_1	任意	サイクル スチール
	010101	01	SIOF 送信部	TDREQ (送信データ転送 リクエスト)	任意	SIOF/SITDR_0	サイクル スチール
		10	SIOF 受信部	RDREQ (受信データ転送 リクエスト)	SIOF/ SIRDR_0	任意	サイクル スチール
	011100	11	USBF 送信部 0	送信データ エンプティ要求 0	任意	EPDR2i、5	サイクル スチール
			USBF 受信部 0	受信データ フル要求 0	EPDR2o、6	任意	サイクル スチール
		00	USBF 送信部 1	送信データ エンプティ要求 1	任意	EPDR2i、5	サイクル スチール
			USBF 受信部 1	受信データ フル要求 1	EPDR2o、6	任意	サイクル スチール

10.4.4 チャンネルの優先順位

DMAC は同時に複数のチャンネルに対して転送要求があった場合には、決められた優先順位に従って転送を行います。チャンネルの優先順位は固定モード 1、固定モード 2、チャンネル選択ラウンドロビンモード、全チャンネルラウンドロビンモードの 4 種類のモードから選択できます。モードの選択は DMAOR の PR1、PR0 ビットにより行います。

(1) 固定モード

固定モード 1 および 2 ではチャンネルの優先順位は変化しません。

各モードの優先順位は以下の通りです。

- 固定モード 1 : CH0>CH1>CH2>CH3
- 固定モード 2 : CH0>CH2>CH3>CH1

これらの選択は DMAOR の PR1、PR0 ビットにより行います。

(2) ラウンドロビンモード

ラウンドロビンモードでは、1 つのチャンネルで 1 転送単位 (バイト、ワード、ロングワード、または 16 バイト単位) の転送が終了するごとにそのチャンネルの優先順位が一番低くなるように優先順位を変更します。この動作を図 10.3 に示します。なおリセット直後のラウンドロビンモードの優先順位は、CH0>CH1>CH2>CH3 です。

ラウンドロビンモードを指定した場合、複数のチャンネルのバスモードでサイクルスチールモードとバーストモードを混在させないでください。

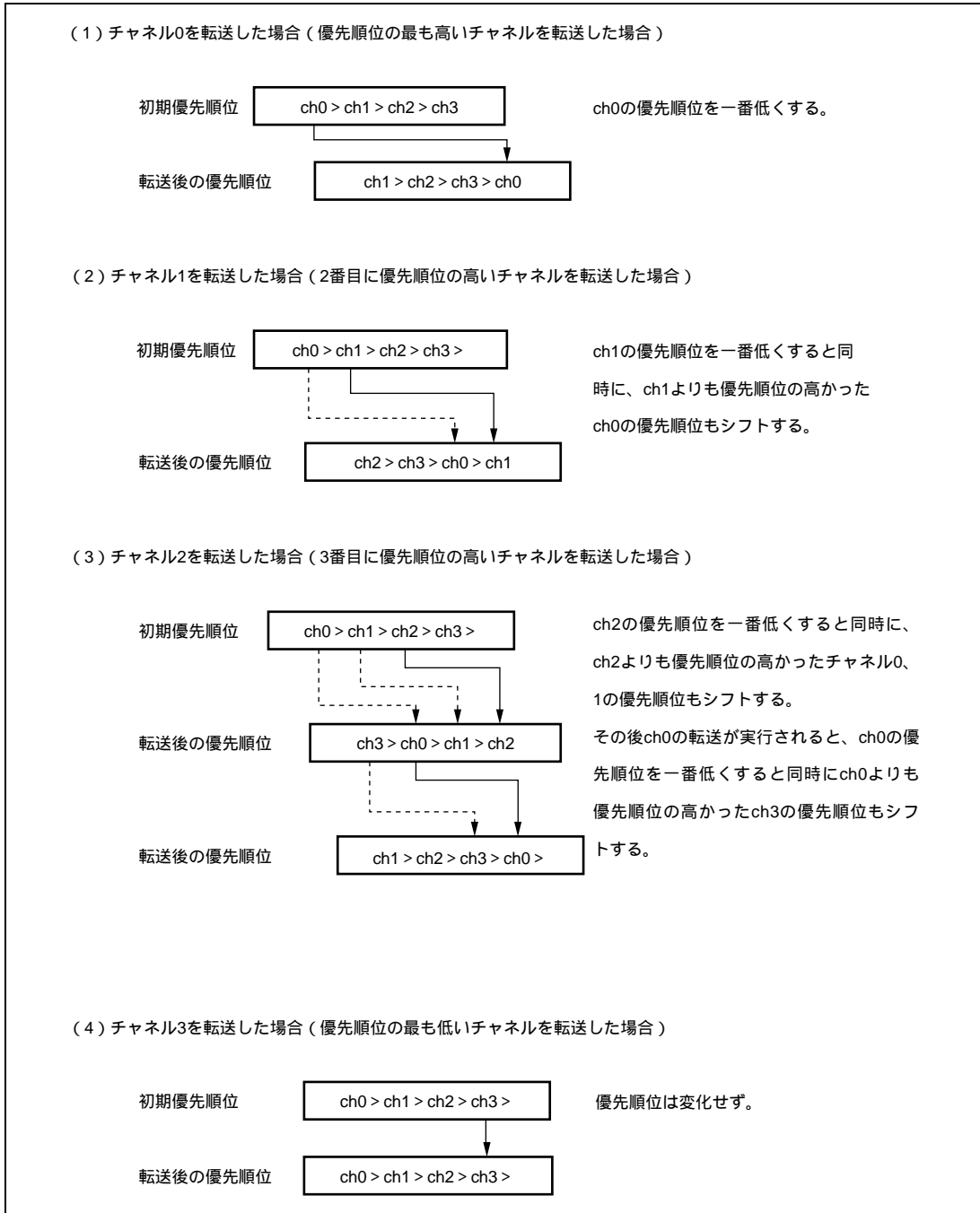


図 10.3 ラウンドロビンモード

10. ダイレクトメモリアクセスコントローラ (DMAC)

図 10.4 にチャンネル 0 とチャンネル 3 に同時に転送要求が発生し、チャンネル 0 の転送中にチャンネル 1 の転送要求が発生した場合のチャンネルの優先順位の変化を示します。この場合の DMAC の動作は以下のようになります。

1. チャンネル0とチャンネル3に同時に転送要求が発生します。
2. チャンネル0のほうがチャンネル3より優先順位が高いので、チャンネル0の転送を開始します（チャンネル3は転送待ち）。
3. チャンネル0の転送中にチャンネル1に転送要求が発生します（チャンネル1とチャンネル3は転送待ち）。
4. チャンネル0の転送を終了すると、チャンネル0の優先順位を一番低くします。
5. この時点でチャンネル1のほうがチャンネル3より優先順位が高いので、チャンネル1の転送を開始します（チャンネル3は転送待ち）。
6. チャンネル1の転送を終了すると、チャンネル1の優先順位を一番低くします。
7. チャンネル3の転送を開始します。
8. チャンネル3の転送を終了すると、チャンネル3の優先順位が一番低くなるように、チャンネル3と一緒にチャンネル2の優先順位を低くします。

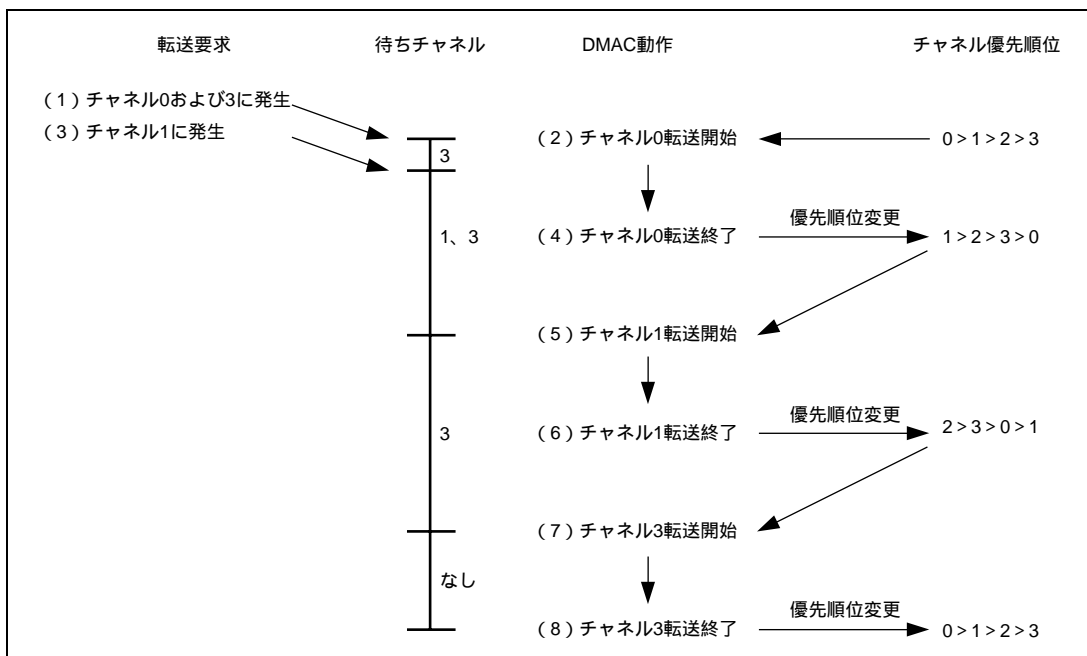


図 10.4 ラウンドロビンモードでのチャンネル優先順位

10.4.5 DMA 転送の種類

DMA 転送は、転送元と転送先を何回のバスサイクルでアクセスするかによって、シングルアドレスモード転送とデュアルアドレスモード転送に分けられます。具体的な転送動作タイミングは、バスモードによって異なります。バスモードには、サイクルスチールモードとバーストモードがあります。表 10.7 に DMAC がサポートできる転送を示します。

表 10.7 サポートできる DMA 転送

転送元	転送先				
	DACK 付き 外部デバイス	外部メモリ	メモリマップト 外部デバイス	内蔵周辺 モジュール	XY メモリ U メモリ
DACK 付き 外部デバイス	不可	デュアル、 シングル	デュアル、 シングル	不可	不可
外部メモリ	デュアル、 シングル	デュアル	デュアル	デュアル	デュアル
メモリマップト 外部デバイス	デュアル、 シングル	デュアル	デュアル	デュアル	デュアル
内蔵周辺 モジュール	不可	デュアル	デュアル	デュアル	デュアル
XY メモリ U メモリ	不可	デュアル	デュアル	デュアル	デュアル

- 【注】
1. デュアル：デュアルアドレスモード
 2. シングル：シングルアドレスモード
 3. 内蔵周辺モジュールは、ロングワードサイズのアクセスを許可しているレジスタに限り 16 バイト転送ができます。

(1) アドレスモード

(a) デュアルアドレスモード

デュアルアドレスモードは、転送元と転送先をともにアドレスによってアクセス（選択）する場合に使うモードです。転送元と転送先は外部でも内部でもかまいません。このモードでは、DMAC は、読み出しサイクルで転送元を、書き込みサイクルで転送先をアクセスし、2 つのバスサイクルで転送を行います。このとき、転送データは一時的に DMAC に格納されます。例えば、図 10.5 のような外部メモリ同士の転送では、読み出しサイクルで一方の外部メモリからデータが DMAC に読み出され、続く書き込みサイクルでそのデータがもう一方の外部メモリに書き込まれます。

10. ダイレクトメモリアクセスコントローラ (DMAC)

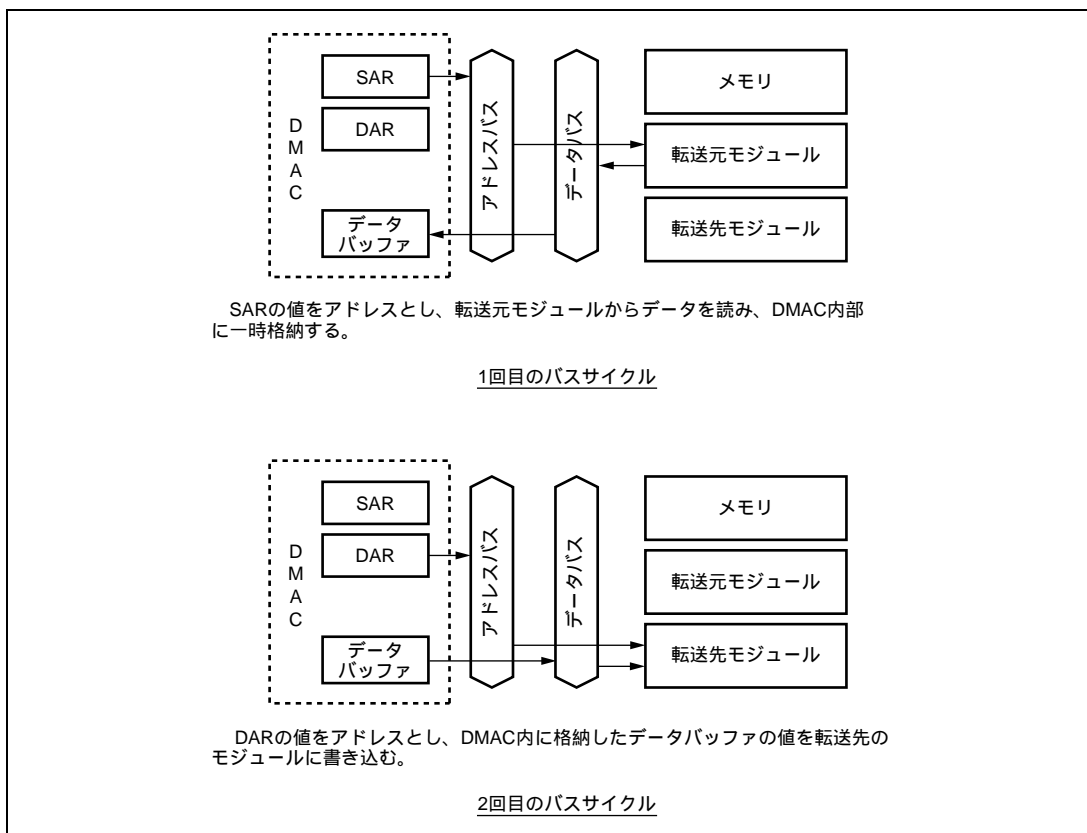


図 10.5 デュアルアドレスモードのデータフロー

転送要求は、オートリクエスト、外部リクエスト、内蔵周辺モジュールリクエストのいずれでも可能です。デュアルアドレスモードでは、DACK はリードサイクルあるいはライトサイクルに出力可能です。リードサイクルとライトサイクルのどちらに出力するかは CHCR の AM ビットによって設定可能です。

図 10.6 にデュアルアドレスモードでの DMA 転送タイミング例を示します。

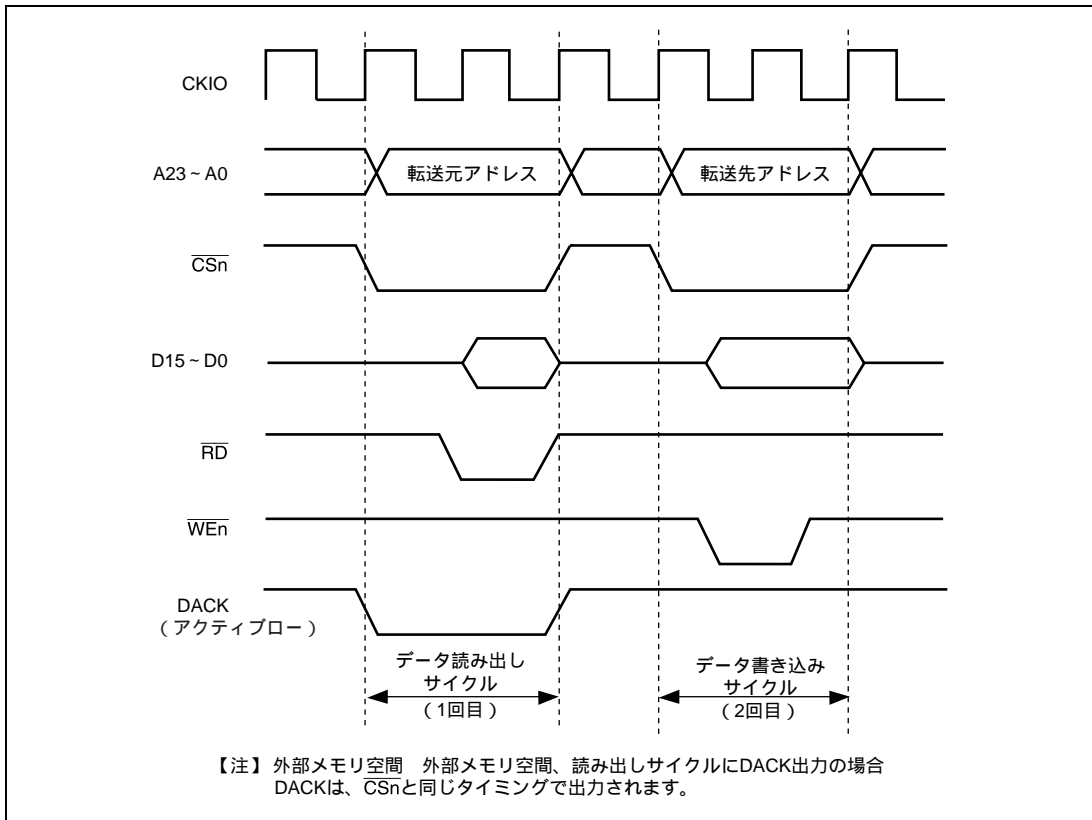


図 10.6 デュアルアドレスモードのDMA転送タイミング例
(転送元：通常メモリ、転送先：通常メモリ)

(b) シングルアドレスモード

シングルアドレスモードは、転送元と転送先がともに外部で、そのうち的一方をDACK信号によってアクセス(選択)し、もう一方をアドレスによってアクセスする場合に使うモードです。このモードでは、DMACは、転送要求受付信号DACKを一方の外部デバイスに出力してアクセスすると同時に、転送相手にアドレスを出して、1つのバスサイクルでDMA転送を行います。たとえば、図10.7のような外部メモリとDACK付き外部デバイスとの転送では、外部デバイスがデータバスにデータを出力するのと同じバスサイクルでそのデータが外部メモリに書き込まれます。

10. ダイレクトメモリアクセスコントローラ (DMAC)

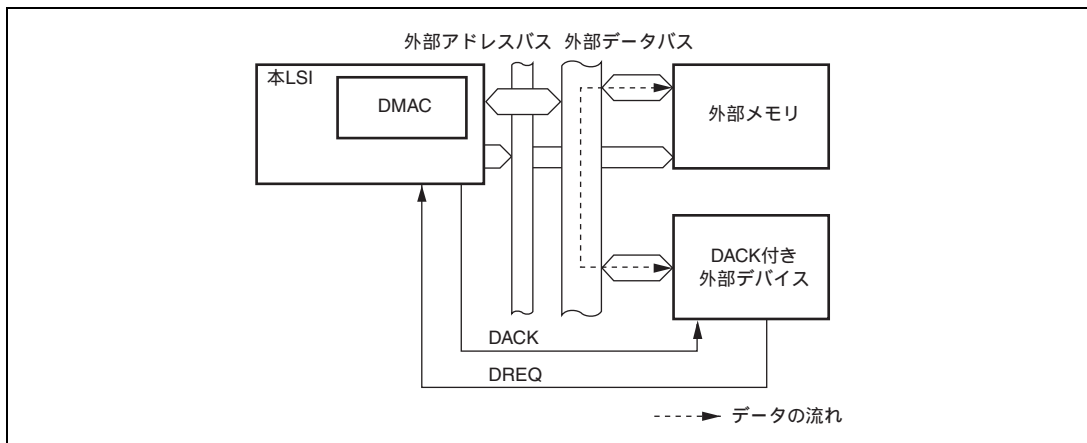


図 10.7 シングルアドレスモードのデータフロー

シングルアドレスモードで可能な転送は、1) DACK 付き外部デバイスとメモリマップト外部デバイス間転送、2) DACK 付き外部デバイスと外部メモリ間転送です。いずれの場合も転送要求は外部リクエスト (DREQ) のみです。

図 10.8 にシングルアドレスモードでの DMA 転送タイミング例を示します。

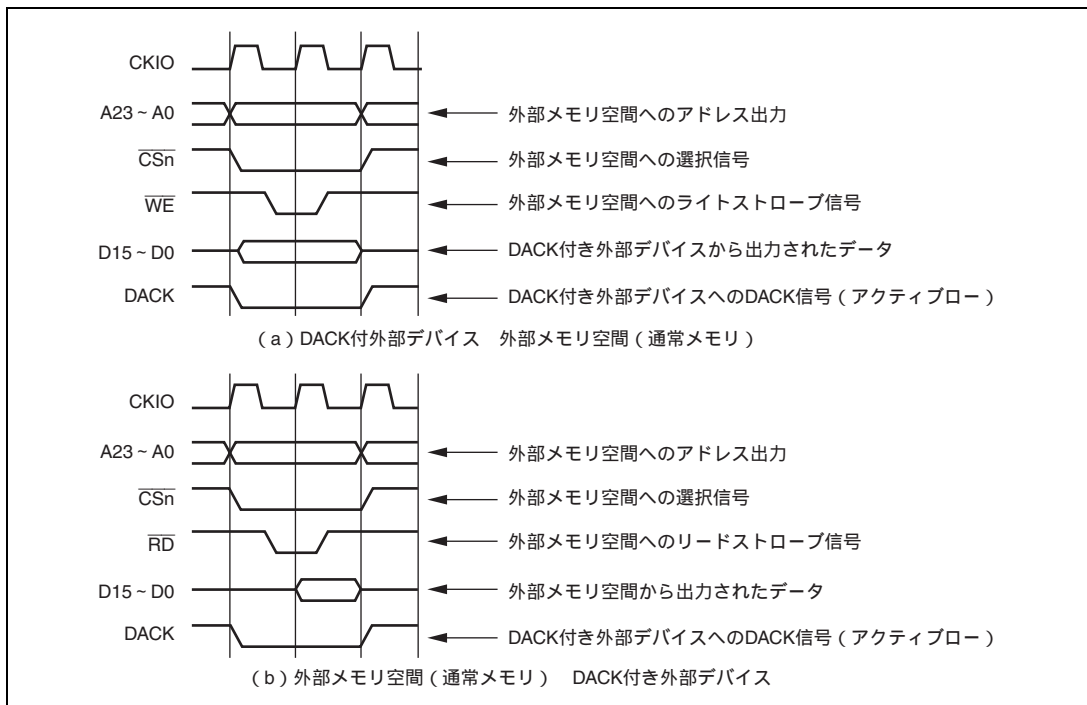


図 10.8 シングルアドレスモードの DMA 転送タイミング例

(2) バスモード

バスモードにはサイクルスチールモードとバーストモードがあります。モードの選択は CHCR の TB ビットによって行います。

(a) サイクルスチールモード

• 通常モード

サイクルスチールの通常モードでは、DMAC は一回の転送単位 (バイト、ワード、ロングワード、または 16 バイト単位) の転送を終了するたびにバス権を他のバスマスタに渡します。その後転送要求があれば、他のバスマスタからバス権を取り戻し、再び 1 転送単位の転送を行い、その転送を終了するとまたバス権を他のバスマスタに渡します。これを転送終了条件が満たされるまで繰り返します。

サイクルスチール通常モードは、転送要求元、転送元、転送先にかかわらずすべての転送区間で使えます。

図 10.9 にサイクルスチール通常モードでの DMA 転送タイミング例を示します。図の例での転送条件は以下のとおりです。

1. デュアルアドレスモード
2. DREQ ローレベル検出

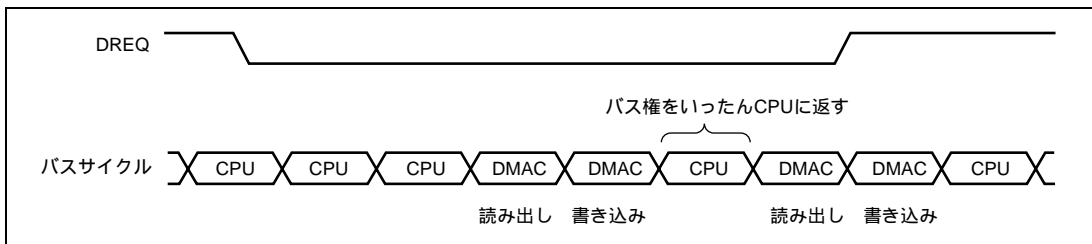


図 10.9 サイクルスチール通常モードの DMA 転送例 (デュアルアドレス、DREQ ローレベル検出)

• インターミットモード 16、インターミットモード 64

サイクルスチールのインターミットモードでは、DMAC は一回の転送単位 (バイト、ワード、ロングワード、または 16 バイト単位) の転送を終了するたびにバス権を他のバスマスタに渡します。その後転送要求があれば、B カウントで 16 クロックまたは 64 クロック待った後に、他のバスマスタからバス権を取り戻し、再び 1 転送単位の転送を行い、その転送を終了するとまたバス権を他のバスマスタに渡します。これを転送終了条件が満たされるまで繰り返します。このため DMA 転送によるバス占有割合をサイクルスチール通常モードに比べ、低く抑えることが可能です。

DMAC が再びバス権を取り戻すときに、キャッシュミスによるエントリの更新などが行われているときなどは、DMA 転送がさらに待たされる場合があります。

インターミットモードは、転送要求元、転送元、転送先にかかわらずすべての転送区間で使えますが、すべてのチャンネルのバスモードがサイクルスチールモードである必要があります。

図 10.10 にサイクルスチールインターミットモードでの DMA 転送タイミング例を示します。図の例での転送条件は以下のとおりです。

10. ダイレクトメモリアクセスコントローラ (DMAC)

1. デュアルアドレスモード
2. DREQローレベル検出

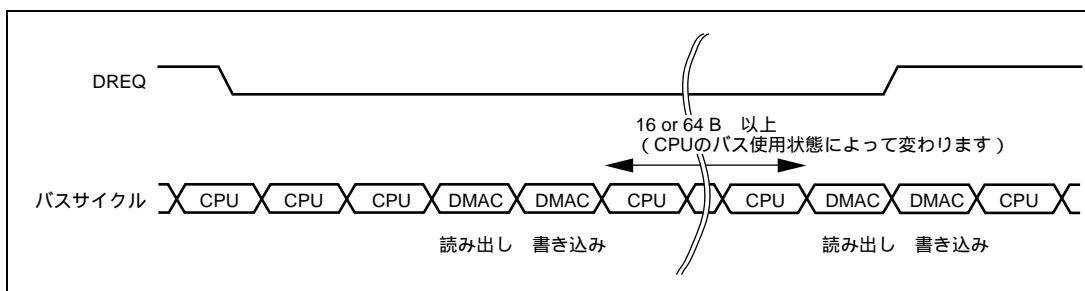


図 10.10 サイクルスチールインターミットモードのDMA転送例
(デュアルアドレス、DREQローレベル検出)

(b) バーストモード

バーストモードではDMACは一度バス権を取ると、転送終了条件が満たされるまでバス権を解放せずに転送を続けます。ただし、外部リクエストモードで、DREQをレベルで検出する場合には、DREQがアクティブなレベルでなくなると、転送終了条件が満たされていない場合でも、すでに要求を受け付けたDMAC転送要求を終了後に他のバスマスタにバス権を渡します。

バーストモードは、内蔵周辺モジュールが転送要求元となっている場合、使用できません。

図 10.11 にバーストモードでのDMA転送タイミング例を示します。

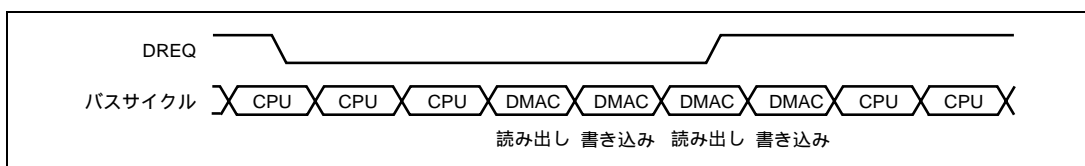


図 10.11 バーストモードのDMA転送例 (デュアルアドレス、DREQローレベル検出)

(3) DMA転送区間とリクエストモード、バーストモードの関係

表 10.8 にDMA転送区間とリクエストモード、バーストモードなどの関連事項を示します。

10. ダイレクトメモリアクセスコントローラ (DMAC)

表 10.8 DMA 転送区間とリクエストモード、バスモードとの関連一覧

アドレスモード	転送区間	リクエストモード	バスモード	転送サイズ (ビット)	使用可能チャンネル
デュアル	DACK 付き外部デバイスと外部メモリ	外部	B/C	8/16/32/128	0
	DACK 付き外部デバイスとメモリマップト外部デバイス	外部	B/C	8/16/32/128	0
	外部メモリと外部メモリ	外部 / オート	B/C	8/16/32/128	0 ~ 3 ^{*3}
	外部メモリとメモリマップト外部デバイス	外部 / オート	B/C	8/16/32/128	0 ~ 3 ^{*3}
	メモリマップト外部デバイスとメモリマップト外部デバイス	外部 / オート	B/C	8/16/32/128	0 ~ 3 ^{*3}
	外部メモリと内蔵周辺モジュール	すべて可 ^{*1}	C	8/16/32/128 ^{*2}	0 ~ 3 ^{*3}
	メモリマップト外部デバイスと内蔵周辺モジュール	すべて可 ^{*1}	C	8/16/32/128 ^{*2}	0 ~ 3 ^{*3}
	内蔵周辺モジュールと内蔵周辺モジュール	すべて可 ^{*1}	C	8/16/32/128 ^{*2}	0 ~ 3 ^{*3}
	X/Y メモリ、U メモリと X/Y メモリ、U メモリ	外部 / オート	B/C	8/16/32/128	0 ~ 3 ^{*3}
	X/Y メモリ、U メモリとメモリマップト外部デバイス	外部 / オート	B/C	8/16/32/128	0 ~ 3 ^{*3}
	X/Y メモリ、U メモリと内蔵周辺モジュール	すべて可 ^{*1}	C	8/16/32/128 ^{*2}	0 ~ 3 ^{*3}
	X/Y メモリ、U メモリと外部メモリ	外部 / オート	B/C	8/16/32/128	0 ~ 3 ^{*3}
シングル	DACK 付き外部デバイスと外部メモリ	外部	B/C	8/16/32/128	0
	DACK 付き外部デバイスとメモリマップト外部デバイス	外部	B/C	8/16/32/128	0

【記号説明】

B: パーストモード

C: サイクルスチールモード

【注】 *1 外部リクエスト、オートリクエスト、内蔵周辺モジュールリクエストのいずれでも可能。ただし、内蔵周辺モジュールリクエストの場合、転送要求元のレジスタは、転送元または転送先のレジスタである必要があります。

*2 転送元または転送先である内蔵周辺モジュールのレジスタで許されるアクセスサイズ。

*3 転送要求が外部リクエストの場合にはチャンネル 0 のみ。

(4) バスモードとチャンネルの優先順位

優先順位固定モード (CH0>CH1) において、チャンネル 1 がパーストモードで転送中でも、それより優先順位の高いチャンネル 0 に転送要求が発生すると、ただちにチャンネル 0 の転送を開始します。

このとき、チャンネル 0 もパーストモードの場合は優先順位の高いチャンネル 0 の転送がすべて終了してから、チャンネル 1 が転送を継続します。

また、チャンネル 0 がサイクルスチールモードの場合、まず優先順位の高いチャンネル 0 が 1 転送単位の転送を行った後、バス権を解放せずに連続してチャンネル 1 が転送されます。その後も、チャンネル 0 チャンネル 1 チャンネル 0 チャンネル 1 というように交互に転送が行われます。つまりバス状態は、サイクルスチールモード転送終了後の

10. ダイレクトメモリアクセスコントローラ (DMAC)

CPU サイクルがバーストモード転送に置き換わった形になります(以後バーストモードの優先実行と呼びます)。この例を図 10.12 に示します。競合するバーストモードが複数チャンネルある場合は、その中で一番優先順位の高いチャンネルが優先実行されます。

DMA 転送を複数チャンネルで行う場合は、競合するすべてのバースト転送が終了するまでバス権はバスマスタに解放しません。

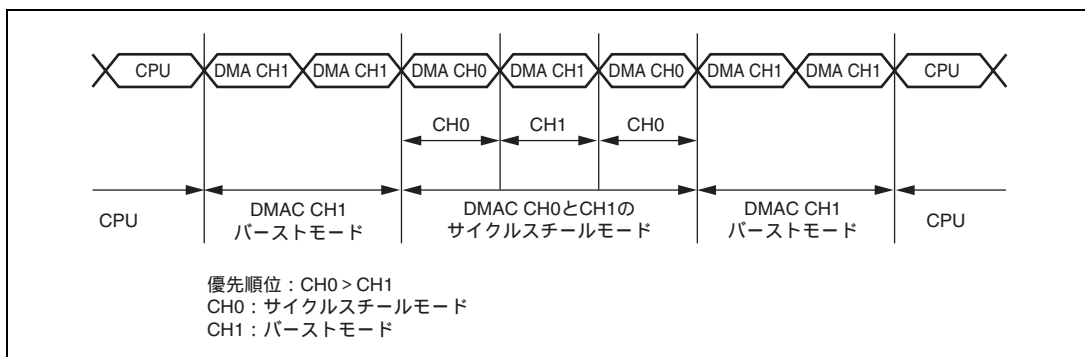


図 10.12 複数チャンネルが動作する場合のバス状態

ラウンドロビンモードでは、図 10.3 に示した仕様で優先順位が変化します。ただしバスモードがサイクルスチールのチャンネルとバーストモードのチャンネルを混在することはできません。

10.4.6 バスサイクルのステート数と DREQ 端子のサンプリングタイミング

(1) バスサイクルのステート数

DMAC がバスマスタのときのバスサイクルのステート数は、CPU がバスマスタのときと同様にバスステートコントローラ (BSC) で制御されます。詳しくは、「第 9 章 バスステートコントローラ (BSC)」を参照してください。

(2) DREQ 端子のサンプリングタイミング

各バスモードに対する DREQ 入力のサンプリングタイミングを図 10.13、図 10.14、図 10.15、および図 10.16 に示します。

10. ダイレクトメモリアクセスコントローラ (DMAC)

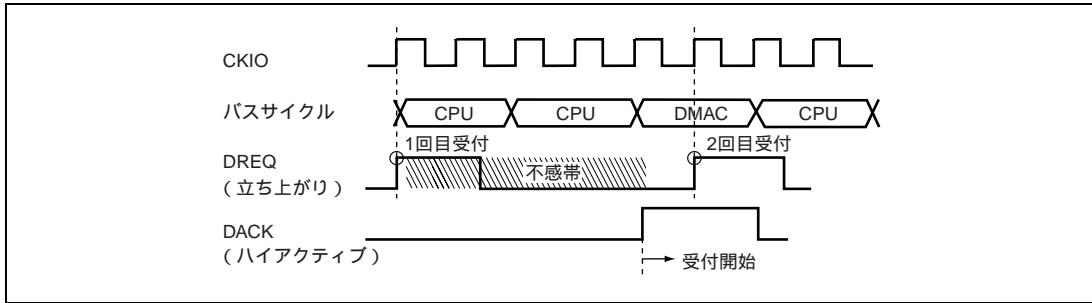


図 10.13 サイクルスチールモード・エッジ検出時の DREQ 入力検出タイミング

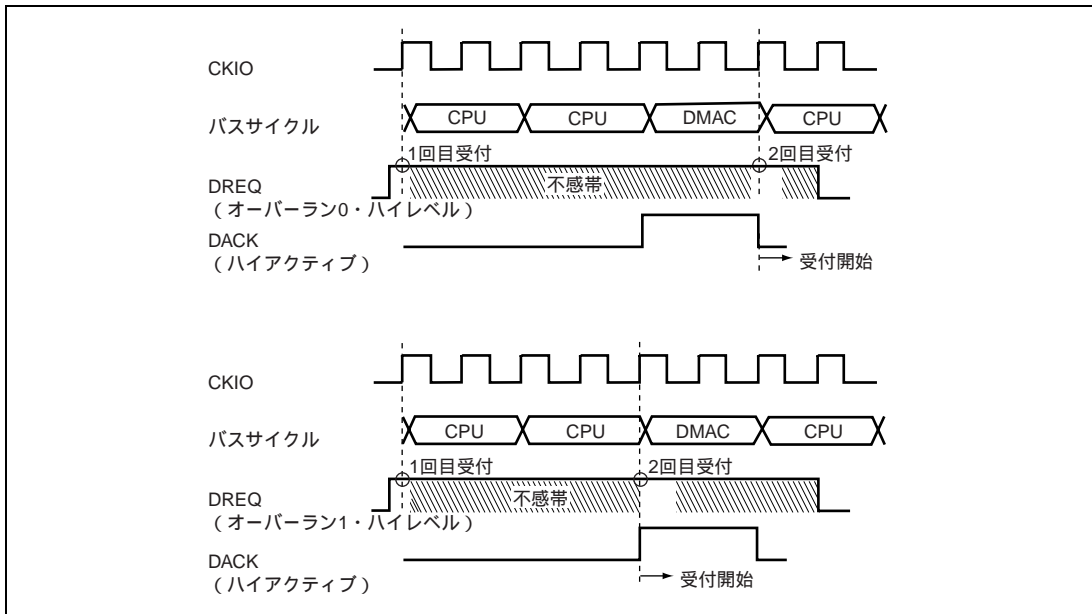


図 10.14 サイクルスチールモード・レベル検出時の DREQ 入力検出タイミング

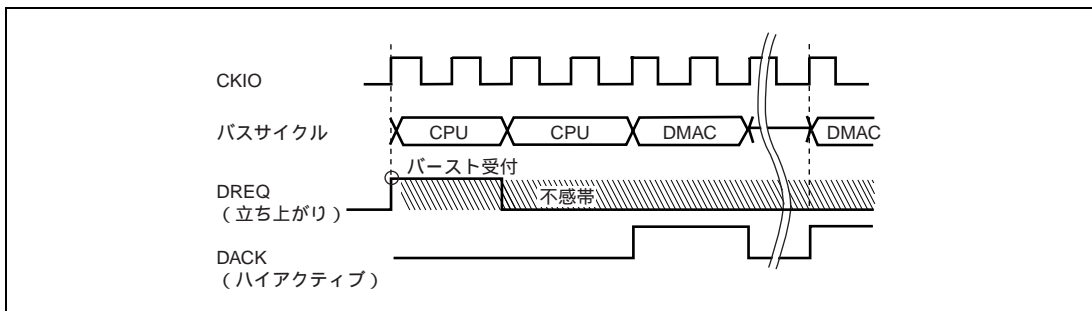


図 10.15 バーストモード・エッジ検出時の DREQ 入力検出タイミング

10. ダイレクトメモリアクセスコントローラ (DMAC)

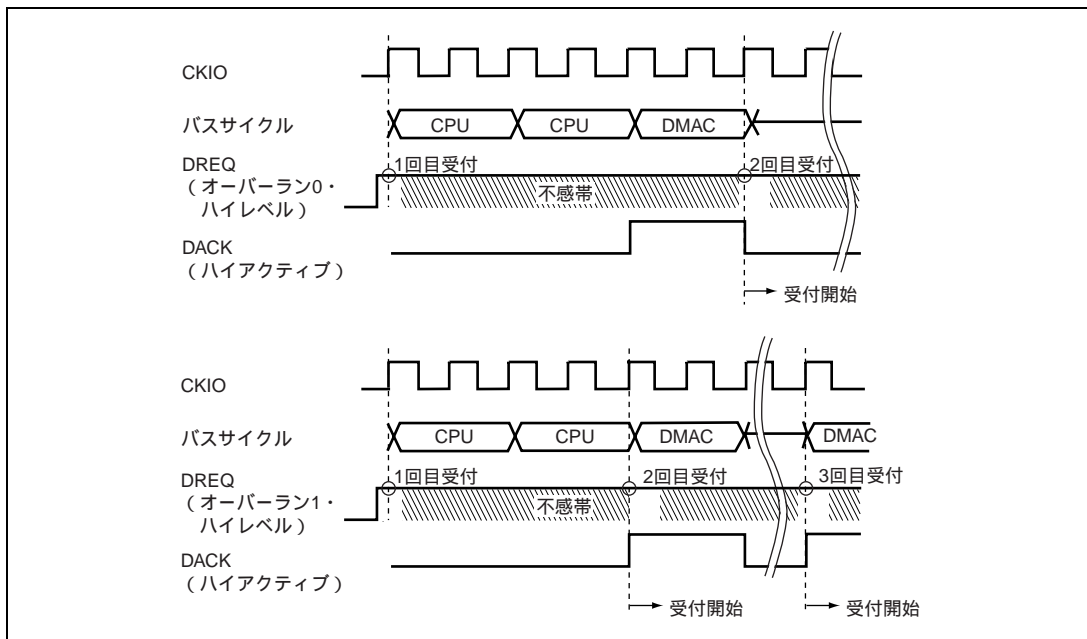


図 10.16 バーストモード・レベル検出時の DREQ 入力検出タイミング

図 10.17 に TEND 出力のタイミングを示します。

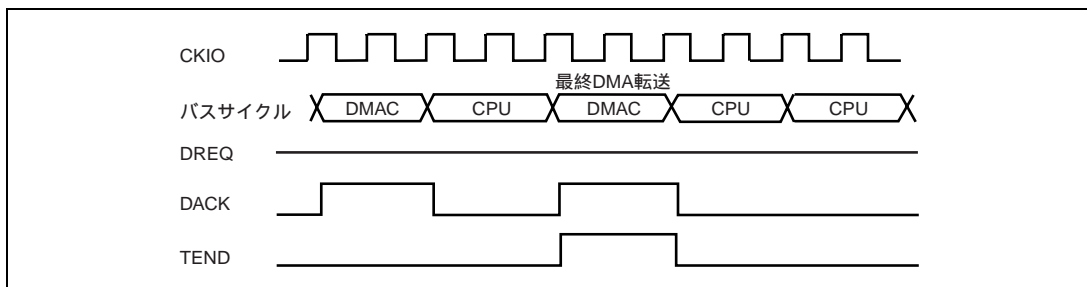


図 10.17 DMA 転送終了信号タイミング (サイクルスチール・レベル検出)

8 ビット、16 ビット外部デバイスに 16 バイト転送やロングワードアクセスしたり、8 ビット外部デバイスにワードアクセスをする場合は、DMA 転送単位が複数のバスサイクルに分割されます。

このとき、バスサイクル間で CS がネゲートする設定を選択している場合、データをアライメントするために CS と同様に DACK 出力および TEND 出力が分割されます。この例を図 10.18 に示します。

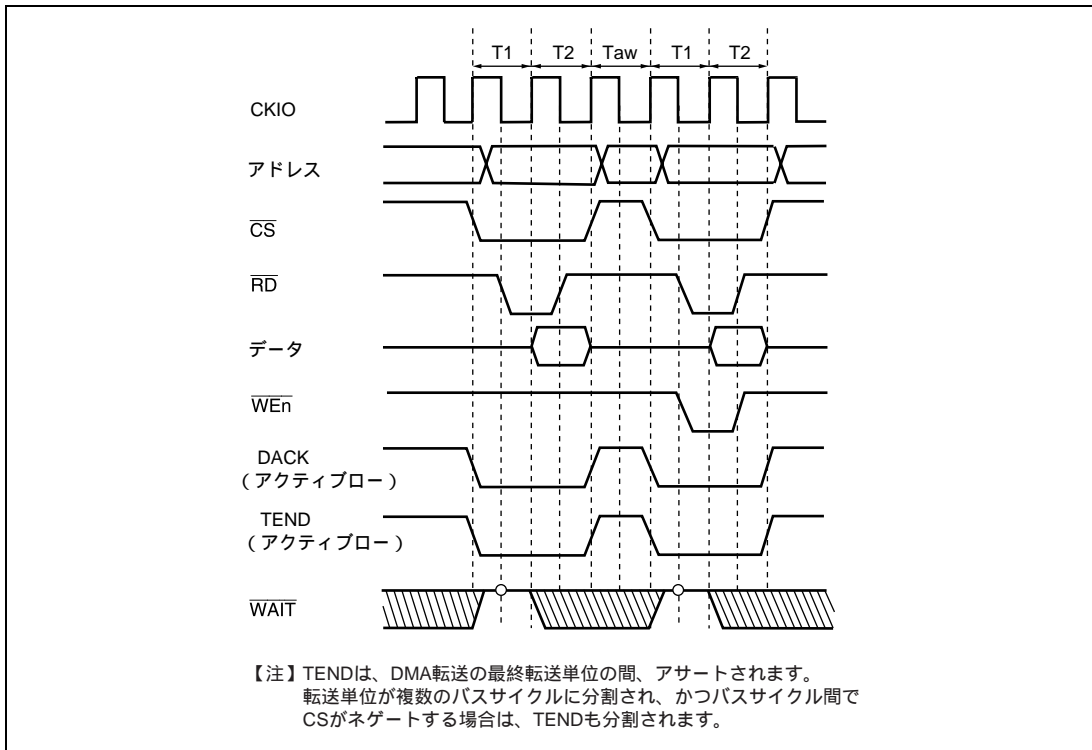


図 10.18 BSC 通常メモリアクセス (ノーウエイト、アイドルサイクル1、16ビットデバイスへのロングワードアクセス)

この分割された DACK 出力では DREQ のサンプリングを正しく検出できません。

外部リクエスト使用時には、下記いずれかの方法で DACK (CS_n) が分割されるのを回避してください。

1. DACK (CS_n) が分割されない設定にする。(例えばDMACのCHCR: RS3 ~ RS0の設定を外部リクエスト、シングルアドレスモードに設定、BSCのCMNCR: DMAIWAに0を設定、転送対象空間のCSnBCR: IWW1、IWW0およびIWRRS1、IWRRS0にサイクル間ウェイト0を設定、CSnWCRのWMを外部ウェイト入力無視に設定)
2. 1転送単位の転送サイズを外部デバイスのバス幅以下に設定する。

10.5 使用上の注意事項

(1) チャネルコントロールレジスタ (CHCR_n) の書き換え

DMAC 各チャネルのチャネルコントロールレジスタ (CHCR_n) を書き換えるときは、まず当該チャネルの DE ビットを 0 クリアしてから書き換えてください。

(2) 低消費電力モードへの移行

DMA 転送動作中にソフトウェアスタンバイモードに入ったり、DMAC のモジュールスタンバイビットをセットしてモジュールスタンバイ状態にしないでください。ソフトウェアスタンバイモードに移行するときや、モジュールスタンバイ状態にするときは、すべてのチャネルの DE ビットを 0 クリアした後に移行してください。

(3) 転送要求元が内蔵周辺モジュールの場合

転送要求元が内蔵周辺モジュールの場合、バーストモードは使用できませんので、ご注意ください。

(4) U メモリ、および X/Y メモリへのアクセスについて

本 LSI 内蔵の U メモリ、および X/Y メモリへアクセスする場合、物理アドレスによる I バスからのアクセスを行う方法で実施してください。論理アドレスによる L バスからのアクセスを行う方法では正常に動作しません。

(5) 外部リクエスト使用時の注意事項

8 ビット、16 ビット外部デバイスに 16 バイト転送やロングワードアクセスしたり、8 ビット外部デバイスにワードアクセスをする場合は、DMA 転送単位が複数のバスサイクルに分割されます。

このとき、バスサイクル間で CS がネゲートする設定を選択していると、データをアライメントするために CS と同様に DACK 出力および TEND 出力が分割されます。この分割された DACK 出力では DREQ のサンプリングを正しく検出できません。外部リクエスト使用時には、下記いずれかの方法で DACK (CS_n) が分割されるのを回避してください。

1. DACK (CS_n) が分割されない設定にする。(例えば DMAC の CHCR : RS3 ~ RS0 の設定を外部リクエスト、シングルアドレスモードに設定、BSC の CMNCR : DMAIWA に 0 を設定、転送対象空間の CSnBCR : IWW1、IWW0 および IWRRS1、IWRRS0 にサイクル間ウェイト 0 を設定、CSnWCR の WM を外部ウェイト入力無視に設定)
2. 1 転送単位の転送サイズを外部デバイスのバス幅以下に設定する。

11. クロックパルス発振器 (CPG)

本 LSI は、クロックパルス発振器 (CPG) を内蔵しており、内部クロック (I)、周辺クロック (P)、およびバスクロック (B) を生成します。CPG は、発振器、PLL 回路、および分周回路で構成されます。

11.1 特長

- 3種類のクロックモード

使用する周波数範囲、PLLのオン/オフ、および水晶発振子直付けか外部クロック入力かによって、3種類のクロックモードから選択できます。

- 3種類のクロック

CPU、キャッシュで使用する内部クロック (I) と、周辺モジュールで使用する周辺クロック (P)、さらに外部バスインタフェースで使用するバスクロック (B =CKIO) を独立に生成できます。

- 周波数変更機能

CPG内部のPLL (Phase Locked Loop) 回路や分周回路により、内部クロックと周辺クロックの周波数を独立に変更できます。周波数変更は、周波数制御レジスタ (FRQCR) の設定により、ソフトウェアで行います。

- 低消費電力モードの制御

スリープモード、ソフトウェアスタンバイモードでのクロック停止、およびモジュールスタンバイ機能での特定モジュールの停止が可能です。

CPG のブロック図を図 11.1 に示します。

11. クロックパルス発振器 (CPG)

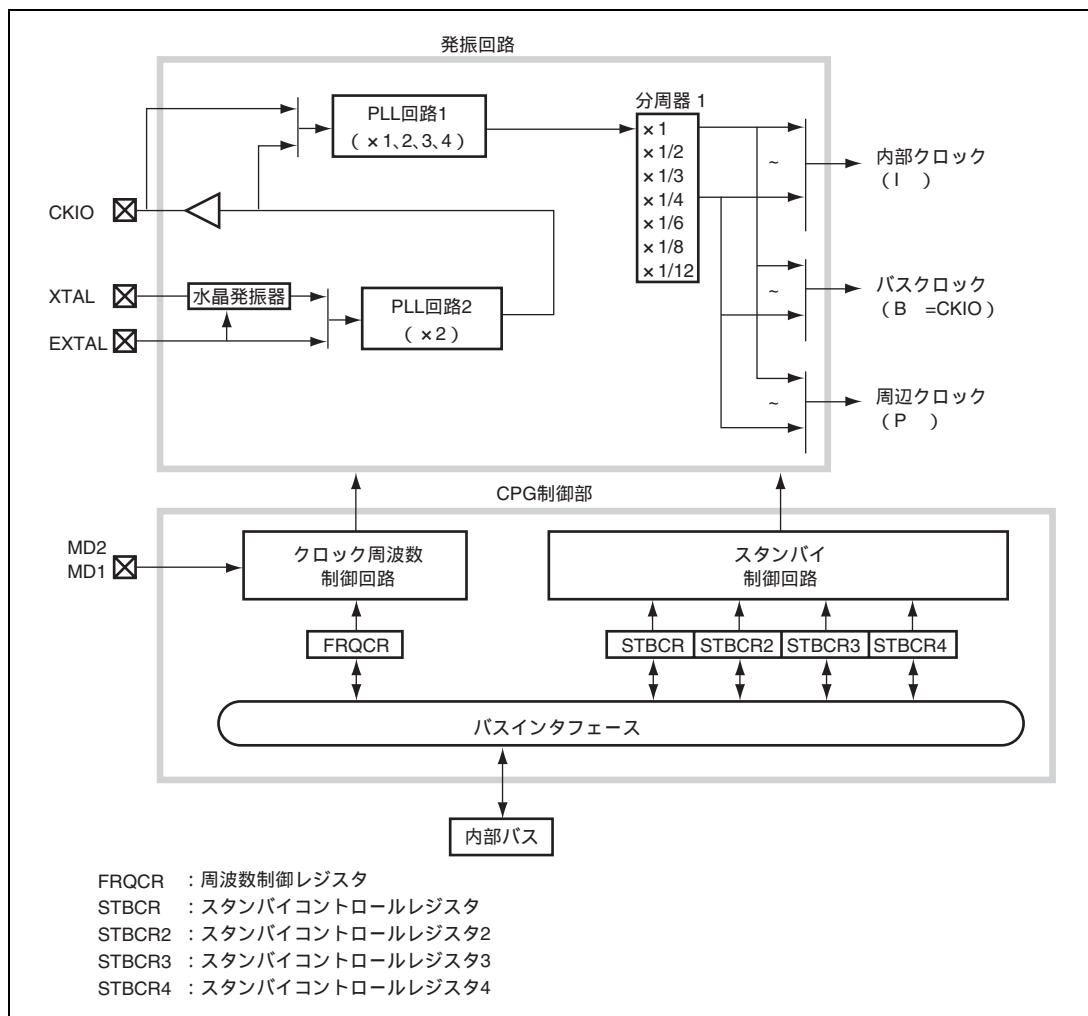


図 11.1 CPG のブロック図

CPG の各ブロックは、次のように機能します。

(1) PLL 回路 1

PLL 回路 1 は、入力クロック周波数を 1 倍、2 倍、3 倍、または 4 倍に逡倍する機能を持ちます。逡倍率は、周波数制御レジスタ (FRQCR) で設定します。このとき、内部クロック、バスクロック、周辺クロックの立ち上がりエッジの位相は、CKIO 端子の立ち上がりエッジの位相に一致するように制御されます。

(2) PLL 回路 2

PLL 回路 2 は、水晶発振器または EXTAL 端子からの入力クロック周波数を 2 倍に逡倍する機能を持ちます。逡倍率は、クロック動作モードにより固定されます。クロック動作モードは、MD1、MD2 端子で設定します。クロック動作モードについては、表 11.2 を参照してください。

(3) 水晶発振器

XTAL または EXTAL 端子に水晶発振子を接続して使用する場合の発振回路です。水晶発振器は、クロック動作モード 6 に設定することにより使用可能となります。

(4) 分周器 1

分周器 1 は、内部クロック、バスクロック、および周辺クロックで使用する動作周波数のクロックを生成する機能を持ちます。動作周波数は、PLL 回路 1 の出力周波数に対して、CKIO 端子のクロック周波数より低くならない範囲で、1 倍、1/2 倍、1/3 倍、1/4 倍、1/6 倍、1/8 倍、または 1/12 倍の選択が可能です。分周率は、周波数制御レジスタ (FRQCR) で設定します。*

【注】 * 内部クロックと周辺クロックは、周波数制御レジスタ (FRQCR) でおのこの独立に選択が可能になっていますが、選択可能な設定の組み合わせでも、おのこの動作周波数の上限値を超える設定をした場合の動作は保証されません。バスクロックは CKIO 端子のクロック周波数と同一である必要があるため、分周率は PLL 回路 1 の逡倍率から自動的に設定されます。

(5) クロック周波数制御回路

クロック周波数制御回路は、MD1、MD2 端子、周波数制御レジスタ (FRQCR) によりクロック周波数を制御します。

(6) スタンバイ制御回路

スタンバイ制御回路は、クロック切り換え時またはスリープモードやスタンバイモード時の内蔵発振回路の状態、および他のモジュールの状態を制御します。

(7) 周波数制御レジスタ

周波数制御レジスタ (FRQCR) には、スタンバイ時の CKIO 端子からのクロック出力の有無、PLL 回路 1 の周波数逡倍率、内部クロック、および周辺クロックの周波数分周率の各制御ビットが割り当てられています。

11. クロックパルス発振器 (CPG)

(8) スタンバイコントロールレジスタ

スタンバイコントロールレジスタ (STBCR、STBCR2～STBCR4) には、低消費電力モードの各制御ビットが割り当てられています。STBCRn については、「第 13 章 低消費電力モード」を参照してください。

11.2 入出力端子

CPG の端子構成を表 11.1 に示します。

表 11.1 端子構成

端子名	入出力	機能
MD1	入力	モード制御端子 クロック動作モードを設定します。
MD2	入力	モード制御端子 クロック動作モードを設定します。
XTAL	出力	クリスタル出力端子 水晶発振子を接続します。
EXTAL	入力	クリスタル入力端子 (クロック入力端子) 水晶発振子を接続します。または外部クロック入力端子として使用します。
CKIO	入出力	クロック入出力端子 外部クロック入力、または外部クロック出力端子として使用します。

11.3 クロック動作モード

モード制御端子 (MD2、MD1) の組み合わせとクロック動作モードの関係を表 11.2 に示します。

クロック動作モードの使用可能周波数範囲を表 11.3 に示します。

表 11.2 クロック動作モード

モード	端子組み合わせ		クロック入出力		PLL 回路 2	PLL 回路 1	CKIO の周波数
	MD2	MD1	供給源	出力			
5	1	0	EXTAL	CKIO	ON ($\times 2$)	ON ($\times 1, 2, 3, 4$)	(EXTAL) $\times 2$
6	0	0	水晶 発振子	CKIO	ON ($\times 2$)	ON ($\times 1, 2, 3, 4$)	(水晶) $\times 2$
7	1	1	CKIO		OFF	ON ($\times 1, 2, 3, 4$)	(CKIO)

- モード5:

EXTAL 端子から外部クロックを入力し、PLL 回路 2 で周波数を 2 倍に通倍して本 LSI 内部に供給するので、外部で生成するクロック周波数が低くてすみません。入力クロック周波数は、10MHz から 30MHz まで使用でき、CKIO の周波数レンジは 20MHz から 60MHz となります。

- モード6:

内蔵水晶発振器を動作させ、発振周波数を PLL 回路 2 で 2 倍に通倍して本 LSI 内部に供給するので、使用する水晶発振子の周波数が低くてすみません。水晶発振子は発振周波数が 10MHz から 30MHz のものを使用でき、CKIO の周波数レンジは 20MHz から 60MHz となります。

- モード7:

このモードでは CKIO 端子が入力になり、この端子に外部クロックを入力して、PLL 回路 1 で波形整形および設定により周波数通倍を行い、本 LSI に供給します。シンクロナス DRAM を接続するのに適したモードです。入力クロック周波数 (CKIO の周波数レンジ) は、20MHz から 60MHz となります。

11. クロックパルス発振器 (CPG)

表 11.3 クロックモードと FRQCR 値の可能な組み合わせ

モード	FRQCR レジスタ値	PLL 回路 1	PLL 回路 2	クロック比 (I : B : P)	入力クロック / 水晶発振子 周波数範囲	CKIO 端子 周波数範囲
5、6	H'1000	ON(x1)	ON(x2)	2 : 2 : 2	10MHz ~ 15MHz	20MHz ~ 30MHz
	H'1001* ²	ON(x1)	ON(x2)	2 : 2 : 1	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1002	ON(x1)	ON(x2)	2 : 2 : 2/3	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1003* ¹	ON(x1)	ON(x2)	2 : 2 : 1/2	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1004	ON(x1)	ON(x2)	2 : 2 : 1/3	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1005	ON(x1)	ON(x2)	2 : 2 : 1/4	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1006	ON(x1)	ON(x2)	2 : 2 : 1/6	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1101	ON(x2)	ON(x2)	4 : 2 : 2	10MHz ~ 15MHz	20MHz ~ 30MHz
	H'1103	ON(x2)	ON(x2)	4 : 2 : 1	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1104	ON(x2)	ON(x2)	4 : 2 : 2/3	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1105	ON(x2)	ON(x2)	4 : 2 : 1/2	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1106	ON(x2)	ON(x2)	4 : 2 : 1/3	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1111	ON(x2)	ON(x2)	2 : 2 : 2	10MHz ~ 15MHz	20MHz ~ 30MHz
	H'1113	ON(x2)	ON(x2)	2 : 2 : 1	10MHz ~ 30MHz	20MHz ~ 60MHz
	H'1202	ON(x3)	ON(x2)	6 : 2 : 2	10MHz ~ 15MHz	20MHz ~ 30MHz
	H'1204	ON(x3)	ON(x2)	6 : 2 : 1	10MHz ~ 20MHz	20MHz ~ 40MHz
	H'1206	ON(x3)	ON(x2)	6 : 2 : 1/2	10MHz ~ 20MHz	20MHz ~ 40MHz
	H'1303	ON(x4)	ON(x2)	8 : 2 : 2	10MHz ~ 15MHz	20MHz ~ 30MHz
	H'1305	ON(x4)	ON(x2)	8 : 2 : 1	10MHz ~ 15MHz	20MHz ~ 30MHz
	H'1306	ON(x4)	ON(x2)	8 : 2 : 2/3	10MHz ~ 15MHz	20MHz ~ 30MHz
H'1313	ON(x4)	ON(x2)	4 : 2 : 2	10MHz ~ 15MHz	20MHz ~ 30MHz	
7	H'1000	ON(x1)	OFF	1 : 1 : 1	20MHz ~ 30MHz	20MHz ~ 30MHz
	H'1001* ²	ON(x1)	OFF	1 : 1 : 1/2	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1002	ON(x1)	OFF	1 : 1 : 1/3	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1003* ¹	ON(x1)	OFF	1 : 1 : 1/4	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1004	ON(x1)	OFF	1 : 1 : 1/6	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1005	ON(x1)	OFF	1 : 1 : 1/8	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1006	ON(x1)	OFF	1 : 1 : 1/12	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1101	ON(x2)	OFF	2 : 1 : 1	20MHz ~ 30MHz	20MHz ~ 30MHz
	H'1103	ON(x2)	OFF	2 : 1 : 1/2	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1104	ON(x2)	OFF	2 : 1 : 1/3	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1105	ON(x2)	OFF	2 : 1 : 1/4	20MHz ~ 60MHz	20MHz ~ 60MHz
	H'1106	ON(x2)	OFF	2 : 1 : 1/6	20MHz ~ 60MHz	20MHz ~ 60MHz

11. クロックパルス発振器 (CPG)

モード	FRQCR レジスタ値	PLL 回路 1	PLL 回路 2	クロック比 (I : B : P)	入力クロック / 水晶発振子 周波数範囲	CKIO 端子 周波数範囲
7	H'1111	ON(x2)	OFF	1 : 1 : 1	20MHz ~ 30MHz	20MHz ~ 30MHz
	H'1202	ON(x3)	OFF	3 : 1 : 1	20MHz ~ 30MHz	20MHz ~ 30MHz
	H'1204	ON(x3)	OFF	3 : 1 : 1/2	20MHz ~ 40MHz	20MHz ~ 40MHz
	H'1206	ON(x3)	OFF	3 : 1 : 1/4	20MHz ~ 40MHz	20MHz ~ 40MHz
	H'1303	ON(x4)	OFF	4 : 1 : 1	20MHz ~ 30MHz	20MHz ~ 30MHz
	H'1305	ON(x4)	OFF	4 : 1 : 1/2	20MHz ~ 30MHz	20MHz ~ 30MHz
	H'1306	ON(x4)	OFF	4 : 1 : 1/3	20MHz ~ 30MHz	20MHz ~ 30MHz
	H'1313	ON(x4)	OFF	2 : 1 : 1	20MHz ~ 30MHz	20MHz ~ 30MHz
H'1333	ON(x4)	OFF	1 : 1 : 1	20MHz ~ 30MHz	20MHz ~ 30MHz	

- 【注】
- 分周器 1 の入力、PLL 回路 1 の出力になります。
 - PLL 回路 1 の入力、モード 7 のときは CKIO 端子からの入力、それ以外のモード 5、6 では PLL 回路 2 の出力になります。
 - 内部クロックの周波数は、CKIO 端子の周波数に PLL 回路 1 の周波数通倍率と分周器 1 の分周率を掛けた周波数になります。内部クロック周波数は、120MHz 以下に設定してください。ただし、CKIO 端子の周波数より低く設定しないでください。
 - 周辺クロックの周波数は、CKIO 端子の周波数に PLL 回路 1 の周波数通倍率と分周器 1 の分周率を掛けた周波数になります。周辺クロック周波数は、30MHz 以下に設定してください。また、CKIO 端子の周波数より高く設定しないでください。
 - PLL 回路 1 の通倍率には、 $\times 1$ 、 $\times 2$ 、 $\times 3$ または $\times 4$ が選択できます。分周器 1 の分周率には、 $\times 1$ 、 $\times 1/2$ 、 $\times 1/3$ 、 $\times 1/4$ 、 $\times 1/6$ 、 $\times 1/8$ 、または $\times 1/12$ が選択できます。これらは、それぞれ周波数制御レジスタ (FRQCR) で設定します。
 - PLL 回路 1 の出力周波数は、CKIO 端子の周波数に PLL 回路 1 の通倍率を掛けた周波数になります。この周波数は、120MHz 以下で使用してください。
 - バスクロックの周波数は、常に CKIO 端子の周波数と等しくなるように設定されます。
 - 表 11.3 のクロック比は、入力クロック周波数を 1 としたときの各周波数の比率です。
 - 表 11.3 にある組み合わせ以外では、本 LSI の動作は保証できません。
 - *1 各モードでのリセット直後の初期値を表します。
 - *2 ブート機能使用時に設定可能な値であることを意味します。
これ以外の値を設定してブート機能を使用しないでください。
また、このときの入力クロック周波数に応じて設定できる SCIF の転送レートに制限があります。詳細は、「17.5 (2) ブート機能使用時のクロック周波数と SCIF の転送レートについて」を参照してください。

11.4 レジスタの説明

CPG には以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態によるレジスタの状態については、「第 27 章 レジスタ一覧」を参照してください。

- 周波数制御レジスタ (FRQCR)
- スタンバイコントロールレジスタ (STBCR、STBCR2~STBCR4)*

【注】 * STBCR、STBCR2~STBCR4 の説明は、「第 13 章 低消費電力モード」を参照してください。

11.4.1 周波数制御レジスタ (FRQCR)

FRQCR は、読み出しまたは書き込み可能な 16 ビットのレジスタで、スタンバイ時の CKIO 端子からクロック出力の有無、PLL 回路 1 の周波数通倍率、内部クロック、および周辺クロックの周波数分周率の指定ができます。FRQCR は、ワードアクセスのみ可能です。

このレジスタは、 $\overline{\text{RESETP}}$ 端子からのリセット時に初期化されます。その他のリセットおよびスタンバイモード時には、前の値を保持しています。

ビット	ビット名	初期値	R/W	説明
15~13		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
12	CKOEN	1	R/W	クロック出力イネーブル CKOEN は、CMNCR の HIZCNT ビットが 1 に設定されているとき、スタンバイ解除時に CKIO 端子から、クロックを出力するか、CKIO 端子をレベル固定するかを選択します。 CMNCR の HIZCNT ビットを 1、かつ本ビットを 0 に設定した場合は、CKIO 端子がローレベルに固定されます。これにより、スタンバイ解除時の不安定な CKIO クロックによって外部回路が誤動作することを防ぐことができます。クロック動作モード 7 のときは、このビットの値によらず CKIO 端子が入力になります。 0：スタンバイ解除時に CKIO 端子をローレベルに固定する。 1：CKIO 端子からクロックを出力する。
11		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

11. クロックパルス発振器 (CPG)

ビット	ビット名	初期値	R/W	説 明
10 9 8	STC2 STC1 STC0	0 0 0	R/W R/W R/W	周波数逡倍率 000 : ×1 倍 001 : ×2 倍 010 : ×3 倍 011 : ×4 倍 100 : 設定禁止 101 : 設定禁止 110 : 設定禁止 111 : 設定禁止
7		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
6 5 4	IFC2 IFC1 IFC0	0 0 0	R/W R/W R/W	内部クロック周波数の分周率 PLL 回路 1 の出力周波数に対しての内部クロック周波数の分周率を指定します。 000 : ×1 倍 001 : ×1/2 倍 010 : 設定禁止 011 : ×1/4 倍 100 : 設定禁止 101 : 設定禁止 110 : 設定禁止 111 : 設定禁止
3		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
2 1 0	PFC2 PFC1 PFC0	0 1 1	R/W R/W R/W	周辺クロック周波数の分周率 PLL 回路 1 の出力周波数に対しての周辺クロック周波数の分周率を指定します。 000 : ×1 倍 001 : ×1/2 倍 010 : ×1/3 倍 011 : ×1/4 倍 100 : ×1/6 倍 101 : ×1/8 倍 110 : ×1/12 倍 110 : 設定禁止 111 : 設定禁止

【注】 設定禁止のコードを設定した場合は動作保証されません。

11.5 周波数変更方法

内部クロックおよび周辺クロックの周波数を変更するには、PLL 回路 1 の通倍率を変える方法と、分周器 1 の分周率を変える方法があります。これらはいずれも周波数制御レジスタ (FRQCR) によってソフトウェアで制御します。以下にこれらの方法について示します。

11.5.1 通倍率の変更

PLL 回路 1 の通倍率を変更する場合は、PLL 安定時間が必要になります。内蔵 WDT により安定時間のカウントを行います。

1. 初期状態では、PLL回路1の通倍率は×1になっています。
2. WDTに、指定された発振安定時間になるように値をセットし、WDTを停止します。次の設定が必要です。
WTCSR.TME = 0 : WDTの停止
WTCSR.CKS[2:0] : WDTカウントクロックの分周率
WTCNT : カウンタの初期値
3. STC[2:0]ビットを目的とする値に設定します。同時にIFC[2:0]、PFC[2:0]ビットに分周率を設定することも可能です。
4. 本LSI内部は、一時的に停止し、WDTのカウントアップを開始します。内部クロック、バスクロック、および周辺クロックが停止し、WDTにのみクロックが供給されます。また、CKIO端子にはクロックが出力され続けます。
5. WDTのカウントオーバーフローで設定されたクロックが供給され始め、本LSIは動作を再開します。WDTはオーバーフロー後、停止します。

11.5.2 分周率の変更

分周率変更のみで、同時に通倍率の変更を行わない場合は、WDT によるカウントは行いません。

1. 初期状態では、IFC[2:0] = B'000、PFC[2:0] = B'011になっています。
2. IFC[2:0]、PFC[2:0]ビットを目的とする値に設定します。クロックモードやPLL回路1の通倍率との関係で設定可能な値は限られます。誤った値を設定すると本LSIは誤動作するので注意してください。
3. 直ちに設定されたクロックに切り替わります。

11.6 ボード設計上の注意事項

(1) 外部水晶発振器使用時の注意

水晶発振器と容量 CL1、CL2 およびダンピング抵抗 R は、できるだけ XTAL 端子と EXTAL 端子の近くに置いてください。また、誘導を避け、正しい発振を行うために、振動子に付加するコンデンサの接地点は共通にし、これらの部品の近くには配線パターンを置かないでください。

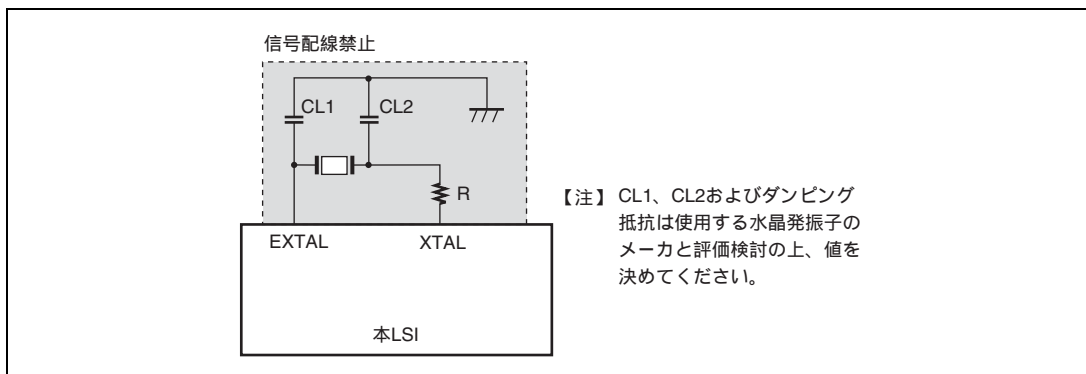


図 11.2 水晶発振器使用時の注意

(2) バイパスコンデンサについての注意

V_{SS} と V_{CC} のペア毎に積層セラミックコンデンサをパスコンとして入れてください。

パスコンは、LSI の電源端子の近くに実装してください。

容量値だけでなく、周波数特性は LSI の動作周波数に合った特性を持つものを使用してください。

11. クロックパルス発振器 (CPG)

(3) PLL 発振回路使用時の注意

PLL 用 Vcc と Vss の接続パターンはボード電源供給端子からの配線長を短くし、パターン幅は広くしてインダクタンス成分を減らしてください。

クロックモード 7 のときは、EXTAL 端子をプルアップし、XTAL 端子は解放にしてください。

PLL のアナログ電源系はノイズ等に敏感であるため、他の電源との干渉によってシステム全体として誤動作を生じさせる可能性があります。このため、本アナログ電源系と Vcc のデジタル電源系は、極力基板上で同一リソースを供給しないでください。

なお、内蔵レギュレータオフ (VDD 外部印加) 選択時は、上記 PLL 用 Vcc は PLL 用 VDD と読み替えてください。

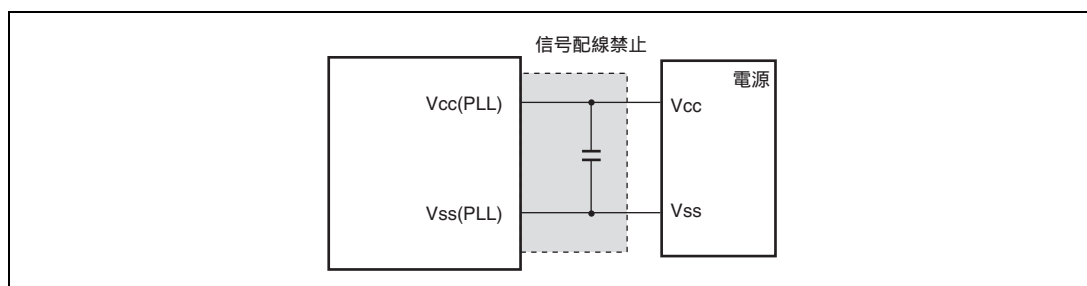


図 11.3 PLL 発振回路使用時の注意

11.7 使用上の注意事項

11.7.1 $\overline{\text{RESETP}}$ 端子を用いたリセットについて

本 LSI 内部には、RDL_REFCLK_IN 端子から入力されるクロックを使ったリセットホールドカウンタが備えられているため、Bluetooth 機能を使用しない場合においても、RDL_REFCLK_IN 端子には必ずクロック供給をするようにしてください。RDL_REFCLK_IN 端子に供給するクロックは、10 ~ 13MHz の範囲であれば EXTAL 端子に供給するクロックと同一のもので対応可能です。

12. ウォッチドッグタイマ (WDT)

本 LSI は、ウォッチドッグタイマ (WDT) を内蔵しており、システムの暴走などによりカウンタ値が書き換えられずにオーバーフローした場合に、LSI の内部をリセットすることができます。

WDT は、1 チャネルのタイマであり、周辺クロックを入力とし、ソフトウェアスタンバイモードや周波数変更時の一時的なスタンバイ状態の解除の際に、クロック安定時間のカウンタに使用します。また、通常のウォッチドッグタイマまたはインターバルタイマとしても使用可能です。

12.1 特長

- クロック安定時間の確保に使用可能
ソフトウェアスタンバイモード、クロック周波数変更時の一時的なスタンバイ状態の解除時に使用します。
- ウォッチドッグタイマモードとインターバルタイマモードを切り替え可能
- ウォッチドッグタイマモード時、内部リセットを発生
カウンタオーバーフローにより、本 LSI 内部をリセットします。
- インターバルタイマモード時、割り込みを発生
カウンタオーバーフローにより、インターバルタイマ割り込みを発生します。
- 8種類のカウント入力クロックを選択可能
周辺クロックを分周した8種類のクロック ($\times 1 \sim \times 1/4096$) から選択できます。
- リセットの種類として、パワーオンリセットとマニュアルリセットを選択できます。

12. ウォッチドッグタイマ (WDT)

WDT のブロック図を図 12.1 に示します。

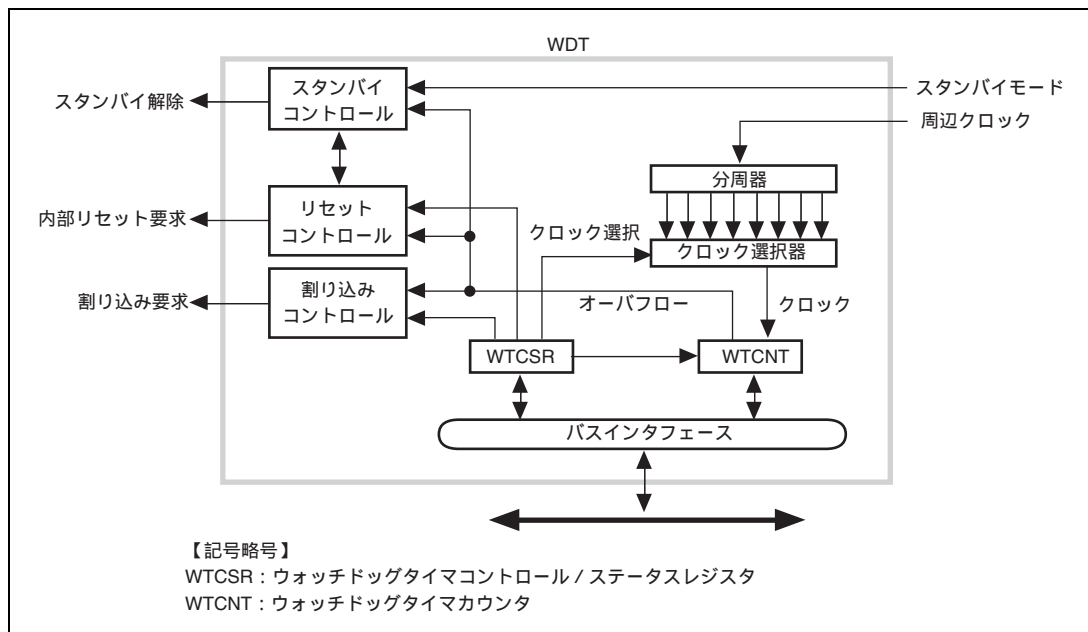


図 12.1 WDT のブロック図

12.2 レジスタの説明

WDT には、以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については、「第 27 章 レジスタ一覧」を参照してください。

- ウォッチドッグタイマカウンタ (WTCNT)
- ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR)

12.2.1 ウォッチドッグタイマカウンタ (WTCNT)

WTCNT は、読み出しおよび書き込みが可能な 8 ビットレジスタです。選択されたクロックでカウントアップするカウンタです。オーバーフローすると、ウォッチドッグタイマモードのときはリセットが発生し、インターバルタイマモードのときは割り込みが発生します。WTCNT は、 $\overline{\text{RESETP}}$ 端子によるリセット時のみ、H'00 に初期化されます。

WTCNT への書き込みは、上位バイトを H'5A にしてワードサイズで行ってください。読み出しは、バイトサイズで行ってください。

【注】 本レジスタは、誤って書き換えられないように、書き込み方法が一般のレジスタと異なります。詳しくは、「12.2.3 レジスタアクセス時の注意」を参照してください。

12.2.2 ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR)

WTCSR は、読み出しおよび書き込みが可能な 8 ビットレジスタです。WTCSR は、カウントに使用するクロックの選択を行うビット、オーバーフローフラグ、およびイネーブルビットからなります。

WDT オーバフローによる内部リセット時には、WTCSR の値は保持されます。 $\overline{\text{RESETP}}$ 端子によるリセット時のみ初期化されます。ソフトウェアスタンバイ解除時のクロック安定時間のカウントに使用するときには、カウンタオーバーフロー後、値が保持されます。

WTCSR への書き込みは、上位バイトを H'A5 にしてワードサイズで行ってください。読み出しは、バイトサイズで行ってください。

【注】 本レジスタは、誤って書き換えられないように、書き込み方法が一般のレジスタと異なります。詳しくは、「12.2.3 レジスタアクセス時の注意」を参照してください。

12. ウォッチドッグタイマ (WDT)

ビット	ビット名	初期値	R/W	説明
7	TME	0	R/W	<p>タイマイネーブル</p> <p>タイマ動作の開始または停止を設定します。ソフトウェアスタンバイモードやクロック周波数変更時に WDT を使用する場合には、このビットを 0 にしてください。</p> <p>0: タイマディスエーブル カウントアップを停止し、WTCNT の値を保持する。</p> <p>1: タイマイネーブル</p>
6	WT/IT	0	R/W	<p>タイマモードセレクト</p> <p>ウォッチドッグタイマとして使用するか、インターバルタイマとして使用するかを指定します。</p> <p>0: インターバルタイマモード 1: ウォッチドッグタイマモード</p> <p>【注】 WDT の動作中に WT/IT を書き換えるとカウントアップが正しく行われないことがあります。</p>
5	RSTS	0	R/W	<p>リセットセレクト</p> <p>ウォッチドッグタイマモードで WTCNT がオーバーフローしたときのリセットの種類を指定します。インターバルタイマモードの場合は、設定値は無視されます。</p> <p>0: パワーオンリセット 1: マニュアルリセット</p>
4	WOVF	0	R/W	<p>ウォッチドッグタイマオーバーフロー</p> <p>ウォッチドッグタイマモードで WTCNT がオーバーフローしたことを示します。インターバルタイマモードでは、セットされません。</p> <p>0: オーバフローなし 1: ウォッチドッグタイマモードで WTCNT がオーバーフローした</p>
3	IOVF	0	R/W	<p>インターバルタイマオーバーフロー</p> <p>インターバルタイマモードで WTCNT がオーバーフローしたことを示します。ウォッチドッグタイマモードでは、セットされません。</p> <p>0: オーバフローなし 1: インターバルタイマモードで WTCNT がオーバーフローした</p>

ビット	ビット名	初期値	R/W	説明	
2	CKS2	0	R/W	クロックセレクト	
1	CKS1	0	R/W	周辺クロック(P)を分周して得られる8種類のクロックから、WTCTのカウンタに使用するクロックを選択します。かっこ内に示すオーバーフロー周期は、周辺クロック P = 15MHz の場合の値です。 クロック分周比 オーバフロー周期 000 : 1 (17 μs) 001 : 1/4 (68 μs) 010 : 1/16 (273 μs) 011 : 1/32 (546 μs) 100 : 1/64 (1.09 ms) 101 : 1/256 (4.36 ms) 110 : 1/1024 (17.48 ms) 111 : 1/4096 (69.91 ms)	
0	CKS0	0	R/W		
					【注】 WDT の動作中に CKS2 ~ CKS0 ビットを書き換えると、カウンタアップが正しく行われず場合があります。CKS2 ~ CKS0 ビットを書き換える場合は、必ず WDT を停止させてください。

12.2.3 レジスタアクセス時の注意

ウォッチドッグタイマカウンタ (WTCNT)、ウォッチドッグタイマコントロール/ステータスレジスタ (WTCSR)は、容易に書き換えられないように、書き込み方法が一般のレジスタと異なります。次の方法で、読み出したり書き込みを行ってください。

- WTCNTおよびWTCSRへの書き込み

WTCNTおよびWTCSRへ書き込むときは、必ずワード転送命令を使用してください。バイト転送およびロングワード転送命令では、書き込めません。

図12.2に示すように、WTCNTへ書き込むときは上位バイトをH'5Aにし、下位バイトを書き込みデータにして転送してください。WTCSRへ書き込むときは上位バイトをH'A5にし、下位バイトを書き込みデータにして転送してください。このように転送すると、下位バイトのデータがWTCNTまたはWTCSRへ書き込まれません。

12. ウォッチドッグタイマ (WDT)

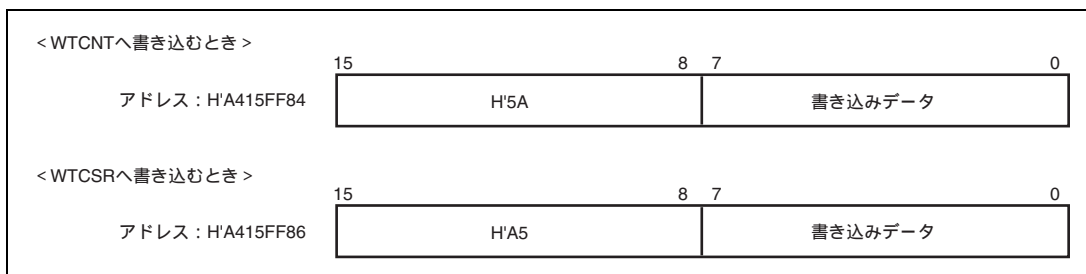


図 12.2 WTCNT および WTCSR への書き込み

12.3 動作説明

12.3.1 ソフトウェアスタンバイ解除の手順

WDT は、ソフトウェアスタンバイモードを NMI などの割り込みで解除する場合に使用します。一例として NMI でソフトウェアスタンバイモードを解除する手順を以下に示します。(リセットで解除する場合は、WDT は動作しないため、クロックが安定するまで $\overline{\text{RESETP}}$ 端子をローレベルに保ってください。)

- ソフトウェアスタンバイモードへの遷移前に、必ず WTCSR の TME ビットを 0 に設定してください。TME ビットが 1 に設定されていると、カウントオーバーフロー時に誤ってリセットまたはインターバルタイマ割り込みが発生することがあります。
- WTCSR の CKS2 ~ CKS0 ビットに使用するカウントクロックの種類と WTCNT に初期値を設定しておきます。これらの値は、カウントオーバーフローまでの時間がクロック発振安定時間以上になるように設定してください。WT/IT ビットの設定値には依存しません。
- SLEEP 命令実行によりソフトウェアスタンバイモードに遷移して、クロックは停止します。
- NMI 信号変化のエッジなどの割り込み検出により、WDT がカウントを開始します。
- WDT がカウントオーバーフローすると、CPG がクロック供給を開始して、本 LSI が動作を再開します。このとき、WTCSR の WOVF および IOVF ビットは、セットされません。
- WDT は H'00 からカウントを継続するので、割り込み処理プログラムの中で STBCR の STBY ビットを 0 にしてください。これにより、WDT が停止します。STBY ビットが 1 のままだと、WDT が H'80 までカウントした時点で再びソフトウェアスタンバイモードに入ります。このソフトウェアスタンバイモードは、パワ - オンリセットで解除してください。

12.3.2 周波数変更の手順

PLL を使用した周波数変更時に、WDT を使用します。分周器の切り替えのみによる周波数変更の場合は、WDT は使用しません。

1. 周波数変更前に、必ずWTCSRのTMEビットを0に設定してください。TMEビットが1に設定されていると、カウントオーバーフロー時に誤ってリセットまたはインターバルタイマ割り込みが発生することがあります。
2. WTCSRのCKS2～CKS0ビットに使用するカウントクロックの種類とWTCNTに初期値を設定しておきます。これらの値は、カウントオーバーフローまでの時間がクロック発振安定時間以上になるように設定してください。WT/ITビットの設定値には依存しません。
3. 周波数制御レジスタ (FRQCR) のSTC2～STC0ビットを書き換えると、プロセッサ内部は一時的に停止して、WDTはカウントを開始します。
4. WDTはカウントオーバーフローすると、CPGがクロック供給を再開して、本LSIが動作を再開します。このときは、WTCSRのWOVFおよびIOVFビットはセットされません。
5. カウンタは、H'00の値で停止します。
6. 周波数変更命令の後、WTCNTを書き換える場合には、WTCNTを読み出してH'00になっていることを確認してから書き換えてください。

12.3.3 ウォッチドッグタイマモードの使用法

1. WTCSRのTMEビットに0を設定してください。TMEビットが1に設定されていると、カウントオーバーフロー時に誤ってリセットまたはインターバルタイマ割り込みが発生することがあります。
2. WTCSRのWT/ITビットに1を設定して、RSTSビットにリセットの種類、CKS2～CKS0ビットにカウントクロックの種類、およびWTCNTに初期値を設定します。
3. WTCSRのTMEビットに1をセットすると、ウォッチドッグタイマモードでカウントを開始します。
4. ウォッチドッグタイマモードで動作中は、カウンタがオーバーフローしないように定期的にカウンタをH'00に書き換えてください。
5. カウンタがオーバーフローすると、WDTはWTCSRのWOVFビットを1にセットして、RSTSビットで指定された種類のリセットを発生します。この後、カウンタはカウントを続行します。

12.3.4 インターバルタイマモードの使用法

インターバルタイマモードで動作中は、カウンタがオーバーフローするたびにインターバルタイマ割り込みを発生します。したがって、一定時間ごとに割り込みを発生させることができます。

1. WTCSRのTMEビットに0を設定してください。TMEビットが1に設定されていると、カウントオーバーフロー時に誤ってリセットまたはインターバルタイマ割り込みが発生することがあります。
2. WTCSRのWT/ITビットに0をセットして、CKS2～CKS0ビットにカウントクロックの種類、WTCNTに初期値を設定します。
3. WTCSRのTMEビットに1をセットするとインターバルタイマモードでカウントを開始します。

12. ウォッチドッグタイム (WDT)

4. WDTは、カウンタがオーバーフローするとWTCSRのIOVFビットに1をセットし、インターバルタイム割り込み要求をINTCに送ります。カウンタはカウントを続行します。
5. インターバルタイム割り込み要求を例外処理ルーチンで解除するには、WTCSRのIOVFビットにセットされたフラグをクリアしてください。割り込み例外処理の動作については「第4章 例外処理」および「第8章 割り込みコントローラ (INTC)」を参照してください。

13. 低消費電力モード

本 LSI は、低消費電力モードとしてスリープモード、ソフトウェアスタンバイモード、モジュールスタンバイ機能を内蔵しています。低消費電力モードでは、内蔵周辺モジュールの一部と CPU が機能を停止します。これにより、消費電力を低減させることができます。低消費電力モードは、リセットまたは割り込みによって解除されます。

また、外部から内部電源 VDD(=1.5V)を供給することが可能であれば、内蔵レギュレータを停止して、さらに本 LSI の消費電力を低減できます。

13.1 特長

- スリープモード/ソフトウェアスタンバイモード/モジュールスタンバイ機能をサポートします。
- 内蔵レギュレータをオフして、VDD(=1.5V)を外部から印加することもできます。

13.1.1 低消費電力モードの種類

低消費電力モードには、次のようなモード、機能があります。

1. スリープモード
2. ソフトウェアスタンバイモード
3. モジュールスタンバイ機能 (キャッシュ、X/Yメモリ、Uメモリ、DMAC、UBC、H-UDI、および内蔵周辺モジュール)*

【注】 * USBH/USBF もモジュールスタンバイ機能を備えていますが、クロックソースが他のモジュールとは異なっているため、独立の制御機構を持っています。詳細については、「第 18 章 USB ビンマルチプレクスコントローラ(USBPM)」を参照してください。

プログラム実行状態から各モードへ遷移する条件、各モードでの CPU や周辺モジュールなどの状態、および各モードの解除方法を、表 13.1 に示します。

なお、内蔵レギュレータのオン/オフは、各モードの動作に影響を与えません。

13. 低消費電力モード

表 13.1 低消費電力モードの状態

低消費電力モード	遷移条件	状態							解除方法
		CPG	CPU	CPUレジスタ	内蔵メモリ	内蔵周辺モジュール	端子	外部メモリ	
スリープモード	STBCRのSTBYビットが0の状態 でSLEEP命令を 実行	動作	停止	保持	停止 (内容は 保持)	動作	保持	オート リフレッシュ されます	<ul style="list-style-type: none"> • 割り込み • リセット
ソフトウェアスタンバイモード	STBCRのSTBYビットが1の状態 でSLEEP命令を 実行	停止	停止	保持	停止 (内容は 保持)	停止	*	セルフ リフレッシュ にしてくだ さい	<ul style="list-style-type: none"> • NMI、IRQ • $\overline{\text{RESETP}}$ リセ ット
モジュールスタンバイ機能	STBCRのMSTPビットを1とする	動作	動作	保持	指定モジ ュールが 停止 (内容は 保持)	指定モジ ュールが 停止	保持	オート リフレッシュ されます	<ul style="list-style-type: none"> • MSTP ビット を0にクリア • パワーオン リセット

【注】* 「付録 A. 端子状態表」を参照してください。

ASE モード時にはエミュレータから H-UDI コマンドで解除する方法があります。詳細は、各エミュレータのマニュアルを参照してください。

13.1.2 リセット

リセットは、電源を入れたとき、またはいわゆる初期化状態から再実行したいときに使われます。リセットは、パワーオンリセットとマニュアルリセットの2種類があります。パワーオンリセットの場合は、すべての実行中の処理が中断され、いかなる未処理の事象も取り消されて、リセット処理が直ちに実行されます。一方、マニュアルリセットの場合は、外部メモリの内容を保持するための処理等は継続します。パワーオンリセットおよびマニュアルリセットが発生する条件を以下に示します。

(1) パワーオンリセット発生条件

1. $\overline{\text{RESETP}}$ 端子にローレベルを入力する。
2. WTCSCRのWT/ITビットに1を、RSTSビットに0をセットした状態でWDTのカウンタを開始し、カウンタがオーバーフローしたとき（WDTによるパワーオンリセット要求）。
3. H-UDIリセットが発生したとき。（H-UDIリセットについては、「第26章 ユーザデバッグインタフェース（H-UDI）」を参照してください。）

(2) マニュアルリセット発生条件

WTCSCRのWT/ITビットに1を、RSTSビットに1をセットした状態でWDTのカウンタを開始し、カウンタがオーバーフローしたとき（WDTによるマニュアルリセット要求）

13.1.3 入出力端子

内蔵レギュレータのオン/オフ制御に関する端子を表 13.2 に示します。

表 13.2 端子構成

端子名	入出力	機能
RESETP	入力	パワーオンリセット 端子にローレベルを入力することにより、パワーオンリセット処理へ遷移します。
TEST_REG	入力	LSI 内蔵レギュレータ選択用端子です。ハイの場合、内蔵レギュレータ選択です。 この端子は動作中には変化させないでください。
REG_PD_MAIN	入力 / NC	内部電源(VDD=1.5V)を外部入力時は、ハイ固定してください。その他の時は、どこにも接続しないでください。
REG_PD_BGR	入力 / NC	内部電源(VDD=1.5V)を外部入力時は、ハイ固定してください。その他の時は、どこにも接続しないでください。
REG_PD_VREF	入力 / NC	内部電源(VDD=1.5V)を外部入力時は、ハイ固定してください。その他の時は、どこにも接続しないでください。

13.2 レジスタの説明

低消費電力モード関連で使用するレジスタには、以下のものがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については、「第 27 章 レジスタ一覧」を参照してください。

- スタンバイコントロールレジスタ (STBCR)
- スタンバイコントロールレジスタ2 (STBCR2)
- スタンバイコントロールレジスタ3 (STBCR3)
- スタンバイコントロールレジスタ4 (STBCR4)

13.2.1 スタンバイコントロールレジスタ (STBCR)

STBCR は、低消費電力モードの状態を指定します。STBCR レジスタは、読み出したり書き込み可能な 8 ビットのレジスタです。パワーオンリセット時は初期化されますが、マニュアルリセットでは前の値を保持します。

ビット	ビット名	初期値	R/W	説明
7	STBY	0	R/W	ソフトウェアスタンバイ ソフトウェアスタンバイモードへの遷移を指定します。 0: SLEEP 命令の実行で、スリープモードへ遷移 1: SLEEP 命令の実行で、ソフトウェアスタンバイモードへ遷移
6, 5		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
4	STBXTL	0	R/W	スタンバイクリスタル スタンバイモード時の水晶発振器の停止 / 発振を指定します。 0: スタンバイ状態で水晶発振器の発振が停止します。 1: スタンバイ状態でも水晶発振器は発振継続します。
3		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
2	MSTP2	0	R/W	モジュールストップ 2 MSTP2 ビットを 1 にセットすると内蔵周辺モジュールのうち、タイマユニット (TMU) へのクロックの供給の停止を指定します。 0: TMU は動作 1: TMU へのクロックの供給を停止
1, 0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

13.2.2 スタンバイコントロールレジスタ 2 (STBCR2)

STBCR2 は、読み出したり書き込み可能な 8 ビットのレジスタで、低消費電力モード時の各周辺モジュールの動作を制御します。STBCR2 は、パワーオンリセット時に初期化されますが、マニュアルリセットでは前の値を保持します。

ビット	ビット名	初期値	R/W	説明
7	MSTP10	0	R/W	モジュールストップ 10 MSTP10 ビットを 1 にセットすると H-UDI へのクロックの供給を停止します。 0 : H-UDI は動作 1 : H-UDI へのクロックの供給を停止
6	MSTP9	0	R/W	モジュールストップ 9 MSTP9 ビットを 1 にセットすると UBC へのクロックの供給を停止します。 0 : UBC は動作 1 : UBC へのクロックの供給を停止
5	MSTP8	0	R/W	モジュールストップ 8 MSTP8 ビットを 1 にセットすると DMAC へのクロックの供給を停止します。 0 : DMAC は動作 1 : DMAC へのクロックの供給を停止
4	MSTP7	0	R/W	モジュールストップ 7 MSTP7 ビットを 1 にセットすると DSP へのクロックの供給を停止します。 0 : DSP は動作 1 : DSP へのクロックの供給を停止
3		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
2	MSTP5	0	R/W	モジュールストップ 5 MSTP5 ビットを 1 にセットするとキャッシュメモリへのクロックの供給を停止します。 0 : キャッシュメモリは動作 1 : キャッシュメモリへのクロックの供給を停止
1	MSTP4	0	R/W	モジュールストップ 4 MSTP4 ビットを 1 にセットすると U メモリへのクロックの供給を停止します。 0 : U メモリは動作 1 : U メモリへのクロックの供給を停止

13. 低消費電力モード

ビット	ビット名	初期値	R/W	説明
0	MSTP3	0	R/W	モジュールストップ 3 MSTP3 ビットを 1 にセットすると X/Y メモリへのクロックの供給を停止します。 0 : X/Y メモリは動作 1 : X/Y メモリへのクロックの供給を停止

13.2.3 スタンバイコントロールレジスタ 3 (STBCR3)

STBCR3 は、読み出しまたは書き込み可能な 8 ビットレジスタで、低消費電力時の各モジュールの動作を制御します。STBCR3 は、パワーオンリセット時に初期化されますが、マニュアルリセットでは前の値を保持します。

ビット	ビット名	初期値	R/W	説明
7	MSTP37	0	R/W	モジュールストップ 37 MSTP37 ビットを 1 にセットすると SIOF へのクロックの供給を停止します。 0 : SIOF は動作 1 : SIOF へのクロック供給を停止
6~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

13.2.4 スタンバイコントロールレジスタ 4 (STBCR4)

STBCR4 は、読み出しまたは書き込み可能な 8 ビットレジスタで、低消費電力時の各モジュールの動作を制御します。STBCR4 は、パワーオンリセット時に初期化されますが、マニュアルリセットでは前の値を保持します。

ビット	ビット名	初期値	R/W	説明
7		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
6	MSTP46	0	R/W	モジュールストップ 46 MSTP46 ビットを 1 にセットすると USB へのクロックの供給を停止します。 0 : USB は動作 1 : USB へのクロック供給を停止 【注】 本ビットに 1 をセットすると、本ビットのクリアは、パワーオンリセットでのみ有効となります。0 を書き込んで、本ビットは 0 にクリアされません。

ビット	ビット名	初期値	R/W	説明
5、4		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
3	MSTP43	0	R/W	モジュールストップ 43 MSTP43 ビットを1にセットするとBOOTへのクロックの供給を停止します。 0: BOOTは動作 1: BOOTへのクロック供給を停止 【注】本ビットを1から0に変更したとき（BOOTへのクロック供給停止状態からクロック供給を開始したとき）、実際にBOOTモジュールがアクセス可能状態になるまでには一定の時間がかかります。ウォーミングアップ時間確保のため、本ビットを1から0に変更した場合は、STBCR4レジスタを読み出した後、BOOTモジュールへのアクセスを行ってください。
2	MSTP42	0	R/W	モジュールストップ 42 MSTP42 ビットを1にセットするとDACへのクロックの供給を停止します。 0: DACは動作 1: DACへのクロック供給を停止
1	MSTP41	0	R/W	モジュールストップ 41 MSTP41 ビットを1にセットするとSCIF1へのクロックの供給を停止します。 0: SCIF1は動作 1: SCIF1へのクロック供給を停止
0	MSTP40	0	R/W	モジュールストップ 40 MSTP40 ビットを1にセットするとSCIF0へのクロックの供給を停止します。 0: SCIF0は動作 1: SCIF0へのクロック供給を停止

13.3 動作説明

13.3.1 スリープモード

(1) スリープモードへの遷移

STBCR レジスタの STBY ビットが 0 の状態で SLEEP 命令を実行すると、プログラム実行状態からスリープモードに遷移します。CPU は SLEEP 命令実行後に停止しますが、CPU のレジスタ内容は保持されます。内蔵周辺モジュールは、動作を続けます。CKIO 端子には、クロックが出力され続けます。

(2) スリープモードの解除

スリープモードは、割り込み (NMI、IRQ、内蔵周辺) およびリセットにより、解除されます。

スリープモード中は SR レジスタの BL ビットが 1 でも割り込みを受け付けますので、必要ならば SLEEP 命令実行前に SPC と SSR をスタックに退避してください。

- 割り込みによる解除

NMI、IRQ、および内蔵周辺の各割り込みが発生すると、スリープモードが解除され割り込み例外処理が実行されます。INTEVT2 レジスタには、割り込み要因に対応したコードがセットされます。

- リセットによる解除

パワーオンリセットおよびマニュアルリセットにより、スリープモードは解除されます。

13.3.2 ソフトウェアスタンバイモード

(1) ソフトウェアスタンバイモードへの遷移

STBCR レジスタの STBY ビットが 1 の状態で SLEEP 命令を実行すると、プログラム実行状態からソフトウェアスタンバイモードに遷移します。ソフトウェアスタンバイモードでは、CPU だけでなくクロックや内蔵周辺モジュールも停止します。CKIO 端子からのクロックは、FRQCR レジスタの CKOEN ビットおよび CMNCR レジスタの HIZCNT ビットの設定を組み合わせることで出力状態を選択することができます。

CPU およびキャッシュのレジスタ内容は、保持されます。内蔵周辺モジュールのレジスタに関しては初期化されるものがあります。ソフトウェアスタンバイモード時の周辺モジュールのレジスタ状態を表 13.3 に示します。

表 13.3 ソフトウェアスタンバイモード時のレジスタの状態

モジュール	初期化レジスタ	内容が保持されるレジスタ
割り込みコントローラ (INTC)	-	全レジスタ
バスステートコントローラ (BSC)	-	全レジスタ
DMAC	-	全レジスタ
クロックパルス発振器 (CPG)	-	全レジスタ
タイマユニット (TMU)	TSTR レジスタ	TSTR 以外のレジスタ
SIOF、SCIF0、SCIF1	-	全レジスタ
USBPM、USBH、USBF	-	全レジスタ
D/A コンバータ (DAC)	-	全レジスタ
I/O ポート	-	全レジスタ
ユーザブ레이크コントローラ (UBC)	-	全レジスタ
H-UDI	-	全レジスタ

ソフトウェアスタンバイモードへ遷移する手順を以下に示します。

1. WDTのタイマコントロールレジスタ (WTCSR) のTMEビットを0にし、WDTを停止させます。
2. WDTのタイマカウンタ (WTCNT) に0をセットし、WTCSRレジスタのCKS2 ~ CKS0 ビットに指定された発振安定時間になるように、値を設定します。
3. STBCRレジスタのSTBYビットに1を設定した後、SLEEP 命令を実行させます。

(2) ソフトウェアスタンバイモードの解除

ソフトウェアスタンバイモードは、割り込み (NMI、IRQ) および $\overline{\text{RESETP}}$ 端子からのリセットにより、解除されます。

• 割り込みによる解除

内蔵 WDT によるホットスタートができます。NMI、IRQ の各割り込みが検出されると、WDT のタイマコントロール / ステータスレジスタに設定されている時間が経過した後、LSI 全体にクロックが供給され、ソフトウェアスタンバイモードが解除されます。この後割り込み例外処理が実行され、割り込み要因に対応したコードが INTEVT2 に設定されます。割り込み処理ルーチンに分岐後、STBCR レジスタの STBY ビットをクリアしてください。WDT は、自動的に停止します。STBY ビットをクリアしないと、WDT は動作を継続し WTCNT が H'80 に達した時点でソフトウェアスタンバイモード*に遷移してしまいます。

ソフトウェアスタンバイモード中は、SR レジスタの BL ビットが 1 のときでも割り込みを受け付けますので、必要ならば SLEEP 命令実行前に SPC と SSR をスタックに退避してください。

FRQCR レジスタの CKOEN ビットを 1 かつ CMNCR レジスタの HIZCNT ビットを 1 に設定した場合、割り込み検出直後からソフトウェアスタンバイモードが解除されるまでの間、CKIO 端子のクロック出力の位相が不安定になることがあります。

【注】 * このソフトウェアスタンバイモードは、 $\overline{\text{RESETP}}$ 端子からのリセットで解除してください。

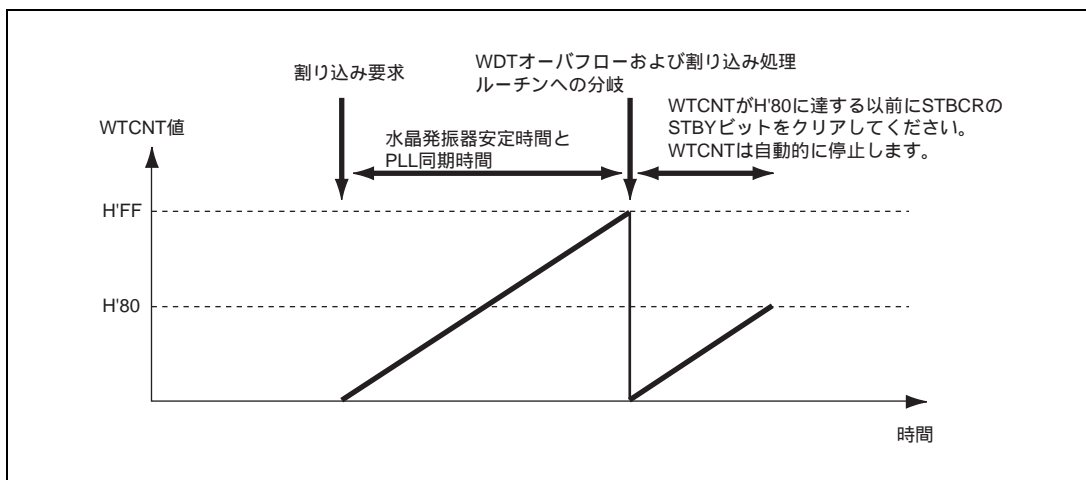


図 13.1 STBCR の STBY ビットとソフトウェアスタンバイモードの解除

- リセットによる解除

$\overline{\text{RESETP}}$ 端子を用いたリセットにより、ソフトウェアスタンバイモードは解除されます。

$\overline{\text{RESETP}}$ 端子は、クロックの発振が安定するまで、ローレベルを保持してください。

クロック動作モード 7 以外は、CKIO 端子は、内部のクロックが出力され続けます。

13.3.3 モジュールスタンバイ機能

(1) モジュールスタンバイ機能への遷移

スタンバイコントロールレジスタの各 MSTP ビットに 1 をセットすることで、それぞれ対応した内蔵周辺モジュールへのクロック供給を停止させることができます。この機能を使用することで、ノーマルモード時およびスリープ時の消費電力を低減させることができます。遷移前には必ずそのモジュールをディスエーブル状態にし、モジュールスタンバイ状態にしてください。

モジュールスタンバイ状態では、内蔵周辺モジュールの外部端子は内蔵周辺モジュールにより異なります。レジスタの状態は、ソフトウェアスタンバイモード時と同じです。表 13.3 を参照してください。

(2) モジュールスタンバイ機能の解除

モジュールスタンバイ機能の解除は、各 MSTP ビットを 0 にクリアするか、パワーオンリセットにより行います。

各 MSTP ビットを 0 にクリアしてモジュールスタンバイ機能を解除する場合は、該当ビットを読み出して 0 クリアされたことを確認してください。

13.3.4 各種モード間の状態遷移

各種モード間の状態遷移を図 13.2 に示します。

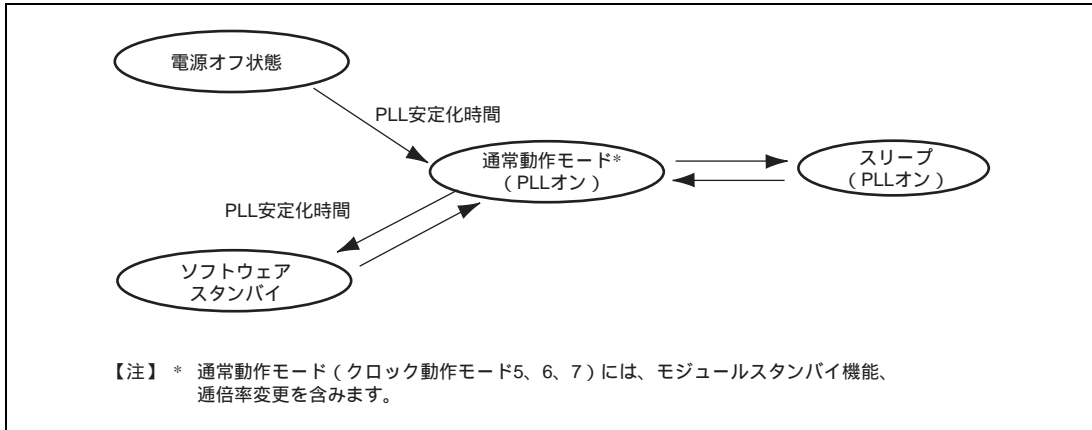


図 13.2 各モード間の状態遷移図

13.3.5 内蔵レギュレータをオフ (外部 VDD 印加) する方法

外部から内部電源 $V_{DD}(=1.5V)$ を供給することが可能であれば、内蔵レギュレータを停止して、さらに本 LSI の消費電力を低減できます。

TEST_REG 端子をロー、REG_PD_MAIN 端子、REG_PD_BGR 端子、REG_PD_VREF 端子をハイにすると内蔵レギュレータがオフされ V_{DD} 端子から 1.5V を供給することが可能となります。

これにより内蔵レギュレータでの内部損失分を低減できます。

14. タイマユニット (TMU)

本 LSI は、3 チャンネルの 32 ビットタイマにより構成されるタイマユニット (TMU) を内蔵しています。

14.1 特長

- オートリロード方式の32ビットダウンカウンタを搭載
- 任意の時点で書き込み / 読み出し可能な、オートリロード用の32ビットコンスタントレジスタおよび32ビットダウンカウンタを搭載
- 32ビットダウンカウンタのアンダフローの発生で割り込み要求を発生 (H'00000000 H'FFFFFF)
- 各チャンネルとも、4種類のカウンタ入力クロックを選択可能
P /4、P /16、P /64、P /256

14. タイムユニット (TMU)

TMUのブロック図を図 14.1 に示します。

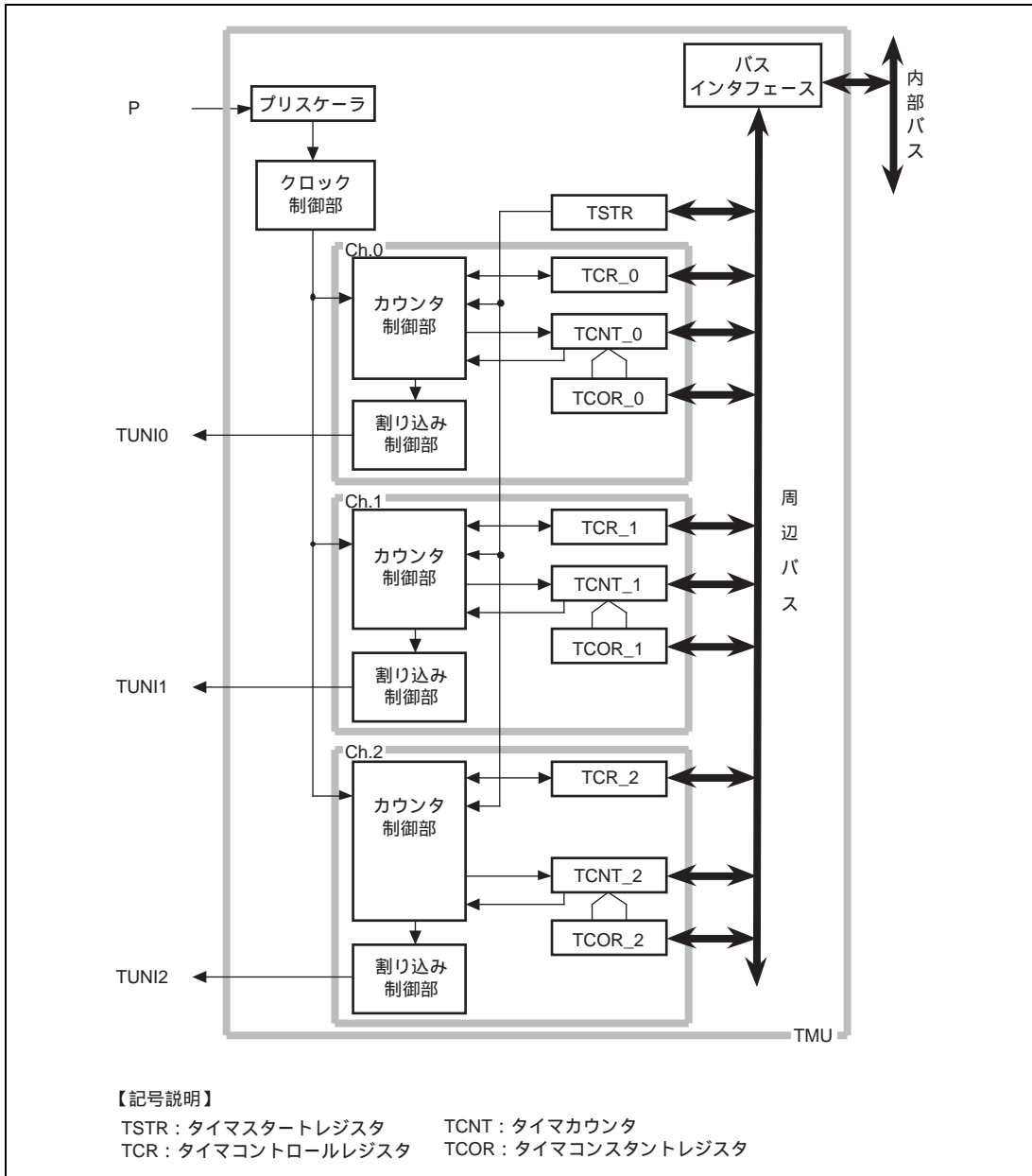


図 14.1 TMUのブロック図

14.2 レジスタの説明

TMU には以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。各チャンネルのレジスタ名については、チャンネル 0 の TCOR は TCOR_0 のように表記しています。

(1) 共通

- タイマスタートレジスタ (TSTR)

(2) チャンネル 0

- タイマコンスタントレジスタ_0 (TCOR_0)
- タイマカウンタ_0 (TCNT_0)
- タイマコントロールレジスタ_0 (TCR_0)

(3) チャンネル 1

- タイマコンスタントレジスタ_1 (TCOR_1)
- タイマカウンタ_1 (TCNT_1)
- タイマコントロールレジスタ_1 (TCR_1)

(4) チャンネル 2

- タイマコンスタントレジスタ_2 (TCOR_2)
- タイマカウンタ_2 (TCNT_2)
- タイマコントロールレジスタ_2 (TCR_2)

14. タイムユニット (TMU)

14.2.1 タイムスタートレジスタ (TSTR)

TSTR は、タイムカウンタ (TCNT) を動作させるか、停止させるかを選択する、読み出し / 書き込み可能な 8 ビットのレジスタです。

ビット	ビット名	初期値	R/W	説明
7-3		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
2	STR2	0	R/W	カウンタスタート 2 タイムカウンタ 2 (TCNT_2) を動作させるか、停止させるかを選択します。 0 : TCNT_2 のカウント動作は停止 1 : TCNT_2 はカウント動作
1	STR1	0	R/W	カウンタスタート 1 タイムカウンタ 1 (TCNT_1) を動作させるか、停止させるかを選択します。 0 : TCNT_1 のカウント動作は停止 1 : TCNT_1 はカウント動作
0	STR0	0	R/W	カウンタスタート 0 タイムカウンタ 0 (TCNT_0) を動作させるか、停止させるかを選択します。 0 : TCNT_0 のカウント動作は停止 1 : TCNT_0 はカウント動作

14.2.2 タイマコントロールレジスタ (TCR)

TCR は、TCNT カウンタの制御および割り込みの制御を行うレジスタで、読み出し / 書き込み可能な 16 ビットレジスタです。

TCR は、タイマカウンタ (TCNT) のアンダフロー発生を示すフラグが 1 にセットされたときの割り込み発生制御、カウンタクロック選択を行います。

ビット	ビット名	初期値	R/W	説明
15~9		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
8	UNF	0	R/(W)*	アンダフローフラグ TCNT のアンダフローを示すフラグです。 0 : TCNT がアンダフローを起こしていない 【クリア条件】 UNF に 0 を書き込んだとき 1 : TCNT がアンダフローを起こした 【セット条件】 TCNT がアンダフローを起こした
7、6		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
5	UNIE	0	R/W	アンダフロー割り込み制御 TCNT のアンダフローの発生を示すステータスフラグ UNF が 1 にセットされたときに割り込み発生を許可するかどうかを制御します。 0 : UNF による割り込み (TUNI) を禁止 1 : UNF による割り込み (TUNI) を許可
4~2		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
1	TPSC1	0	R/W	タイマプリスケアラ
0	TPSC0	0	R/W	TCNT のカウンタクロックを選択します。 00 : P /4 でカウント 01 : P /16 でカウント 10 : P /64 でカウント 11 : P /256 でカウント

【注】 * フラグをクリアするために 0 のみ書き込むことができます。

14. タイマユニット (TMU)

14.2.3 タイマコンスタントレジスタ (TCOR)

TCOR は、TCNT カウンタのアンダフローが発生したとき、TCNT カウンタにセットする値を指定します。

TCOR レジスタは、読み出し / 書き込み可能な 32 ビットレジスタです。

初期値は、H'FFFFFFF です。

14.2.4 タイマカウンタ (TCNT)

TCNT は、入力したクロックにより、カウントダウン動作を行います。入力するクロックは、タイマコントロールレジスタ (TCR) の TPSC1 ~ TPSC0 ビットにより選択します。

TCNT のカウントダウン動作の結果、アンダフロー (H'00000000 ~ H'FFFFFFF) が発生すると、対応するチャネルのタイマコントロールレジスタ (TCR) のアンダフローフラグ (UNF) がセットされます。また、同時に TCNT カウンタ自体には、タイマコンスタントレジスタ (TCOR) の値がセットされ、セットされた値からカウントダウン動作を続けます。

初期値は、H'FFFFFFF です。

14.3 動作説明

各チャネルには、32 ビットのタイマカウンタ (TCNT) と 32 ビットのタイマコンスタントレジスタ (TCOR) があります。TCNT カウンタは、ダウンカウント動作を行います。オートリロード機能によって、周期カウント動作が可能です。

14.3.1 カウンタの動作

タイマスタートレジスタ (TSTR) の STR0 ~ STR2 ビットを 1 にセットすると、対応するチャネルのタイマカウンタ (TCNT) はカウント動作を開始します。TCNT カウンタがアンダフローすると対応するタイマコントロールレジスタ (TCR) の UNF フラグがセットされます。このとき、TCR レジスタの UNIE ビットが 1 ならば、CPU に割り込みを要求します。また、このとき TCNT カウンタには TCOR レジスタから値がコピーされ、ダウンカウント動作を継続します。

(1) カウント動作の設定手順例

カウント動作の設定手順例を図 14.2 に示します。

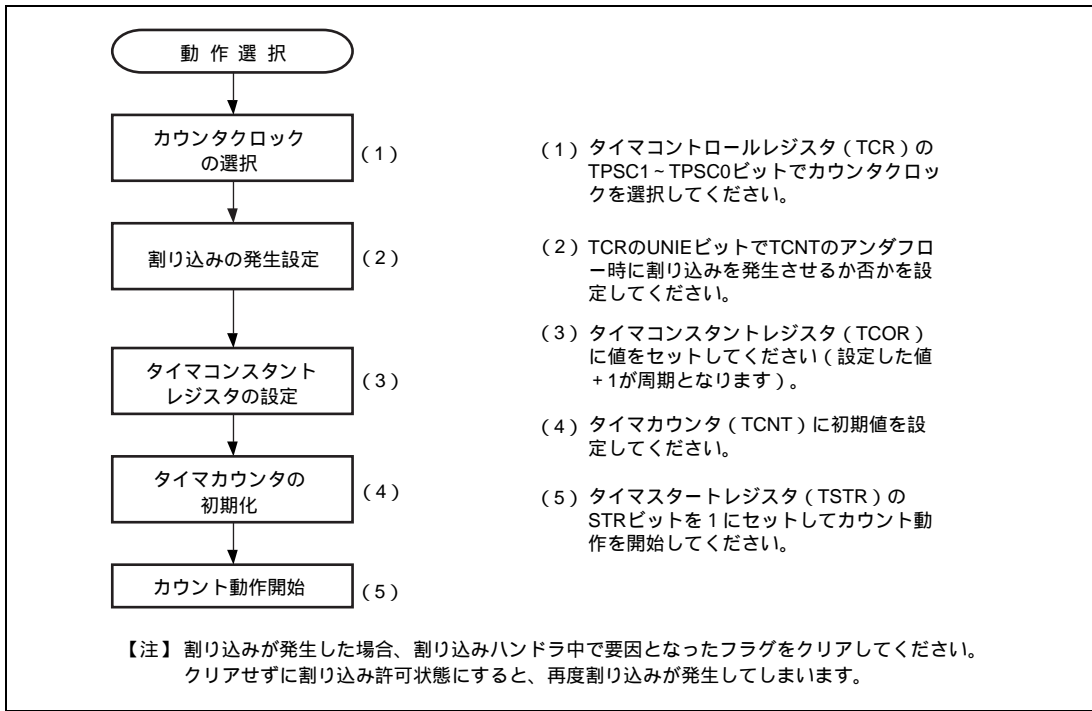


図 14.2 カウント動作設定手順例

(2) オートリロードカウンタ動作

TCNT のオートリロード動作を、図 14.3 に示します。

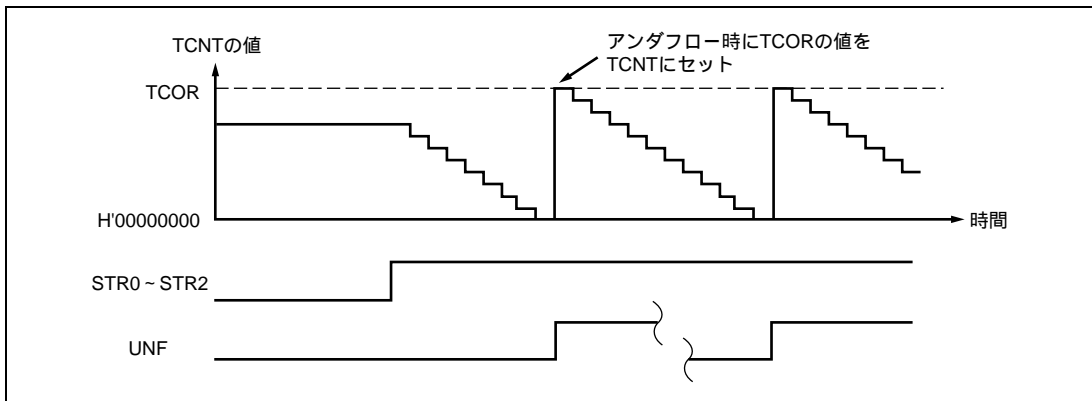


図 14.3 オートリロードカウンタの動作

14. タイマユニット (TMU)

(3) TCNT のカウントタイミング

TCR レジスタの TPSC1 ~ TPSC0 ビットにより、周辺モジュール用クロックを分周した 4 種類のクロック (P /4、P /16、P /64、P /256) が選択できます。このときのタイミングを図 14.4 に示します。

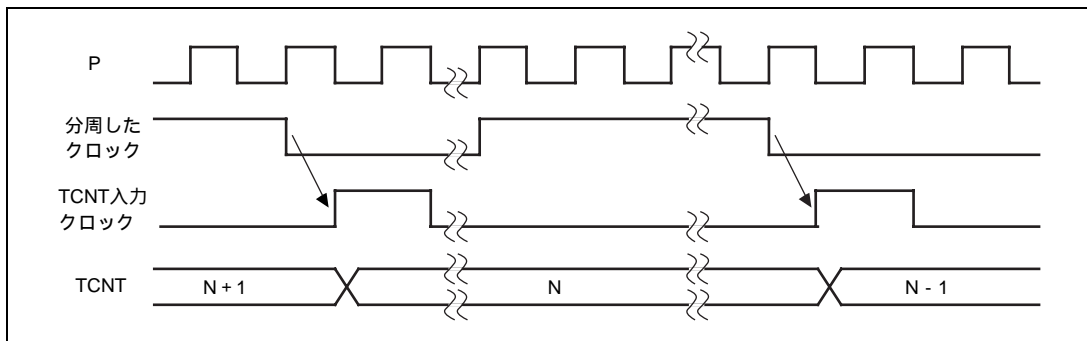


図 14.4 内部クロック動作時のカウントタイミング

14.4 割り込み

TMU の割り込み要因は、アンダフロー割り込み (TUNI) です。

14.4.1 ステータスフラグのセットタイミング

UNF ビットは、TCNT カウンタがアンダフローしたときに 1 にセットされます。このときのタイミングを図 14.5 に示します。

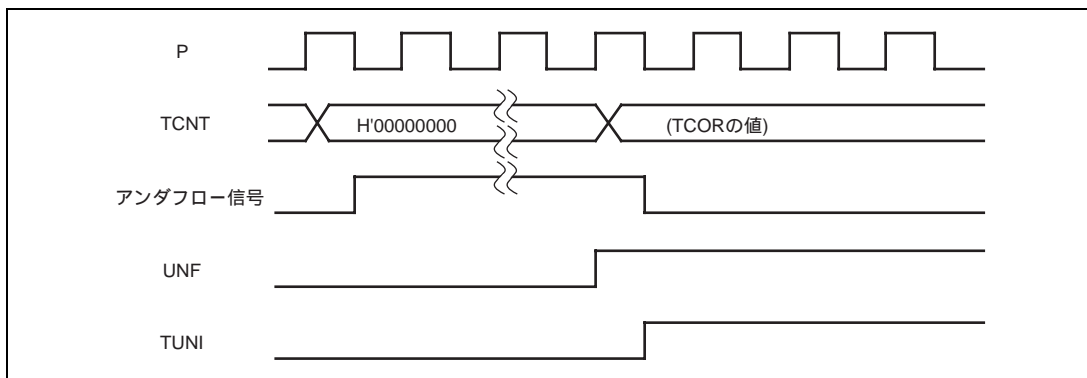


図 14.5 UNF のセットタイミング

14.4.2 ステータスフラグのクリアタイミング

ステータスフラグはCPUから0を書き込むとクリアされます。このときのタイミングを図14.6に示します。

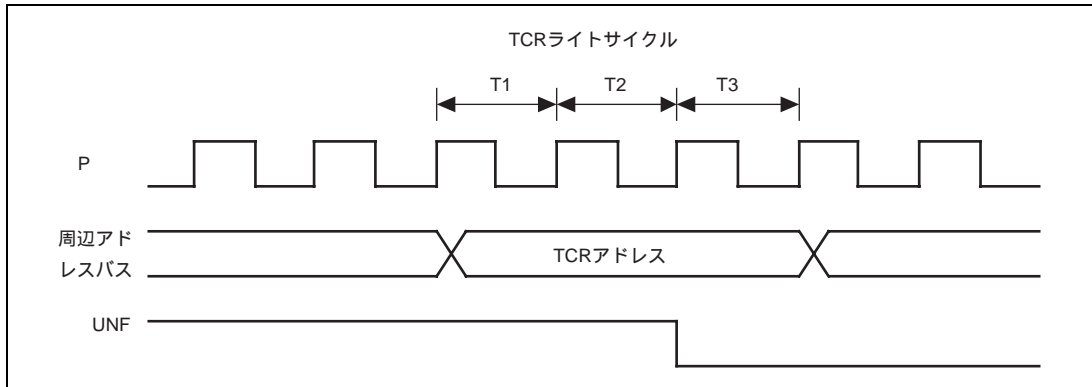


図 14.6 ステータスフラグのクリアタイミング

14.4.3 割り込み要因と優先順位

TMUは各チャンネルごとにアンダフロー割り込みを発生します。割り込み要求フラグが1にセットされ、かつ割り込み許可ビットが1にセットされているとき、当該割り込みが要求されます。これらの割り込みは、割り込み事象レジスタ (INTEVT2) にコードがセットされますので、そのコードにしたがって割り込み処理を行ってください。

チャンネル間の優先順位は、割り込みコントローラにより変更可能です。詳細は「第4章 例外処理」、「第8章 割り込みコントローラ (INTC)」を参照してください。

TMUの割り込み要因を表14.1に示します。

表 14.1 TMUの割り込み要因

チャンネル	割り込み要因	内容	優先順位
0	TUNI0	アンダフロー割り込み 0	高 ↑ ↓ 低
1	TUNI1	アンダフロー割り込み 1	
2	TUNI2	アンダフロー割り込み 2	

14.5 使用上の注意事項

(1) レジスタの書き込みについて

レジスタの書き込み時に、タイマのカウンタ動作の同期処理は行っておりません。レジスタの書き込みの際には、必ずタイマスタートレジスタ (TSTR) の該当チャネルのスタートビット (STR2~STR0) をクリアして、タイマのカウンタ動作を停止させてください。

(2) レジスタの読み出しについて

レジスタの読み出し時に、タイマのカウンタ動作との同期処理を行っています。タイマカウンタ動作とレジスタの読み込み処理が同時に行われた場合は、同期処理により TCNT カウンタのカウンタダウン動作前の値が読み出されます。

15. FIFO 付きシリアルインタフェース (SIOF)

本 LSI は、FIFO 付きクロック同期シリアルインタフェースモジュール (SIOF : Serial Input Output with FIFO) を内蔵しています。SIOF は、シリアルペリフェラルインタフェースバス (SPI) によるシリアル通信が可能です。

15.1 特長

- シリアル転送

FIFO容量32ビット×16段 (送受信独立)

8ビット / 16ビット / 16ビットステレオ音声出力に対応

データの送信はMSBが先頭 (MSB First)

サンプリングレート最大48kHzに対応

同期方法はフレーム同期パルス / 左右ch.切り換えに対応

CODEC制御データインタフェースに対応

リニア / オーディオ / A-Law、 μ -Law CODECチップに接続可能

マスタ / スレーブ両モードに対応

- シリアルクロック

クロックソースとして外部端子入力と内部クロック (P) からの選択が可能

- 割り込み : 1種類

- DMA転送

送信の転送要求による、DMA転送を用いた送信動作に対応

- SPIモード

マスタモード固定で、SPIスレーブデバイスと全二重による連続通信が可能

データサンプリングとしてSCKの立ち下がりエッジ / 立ち上がりエッジを選択

送信タイミングとしてSCKのクロックフェーズを選択

3つのスレーブデバイスを選択

送受信データ長は8ビット固定

15. FIFO 付きシリアルインタフェース (SIOF)

図 15.1 に SIOF のブロック図を示します。

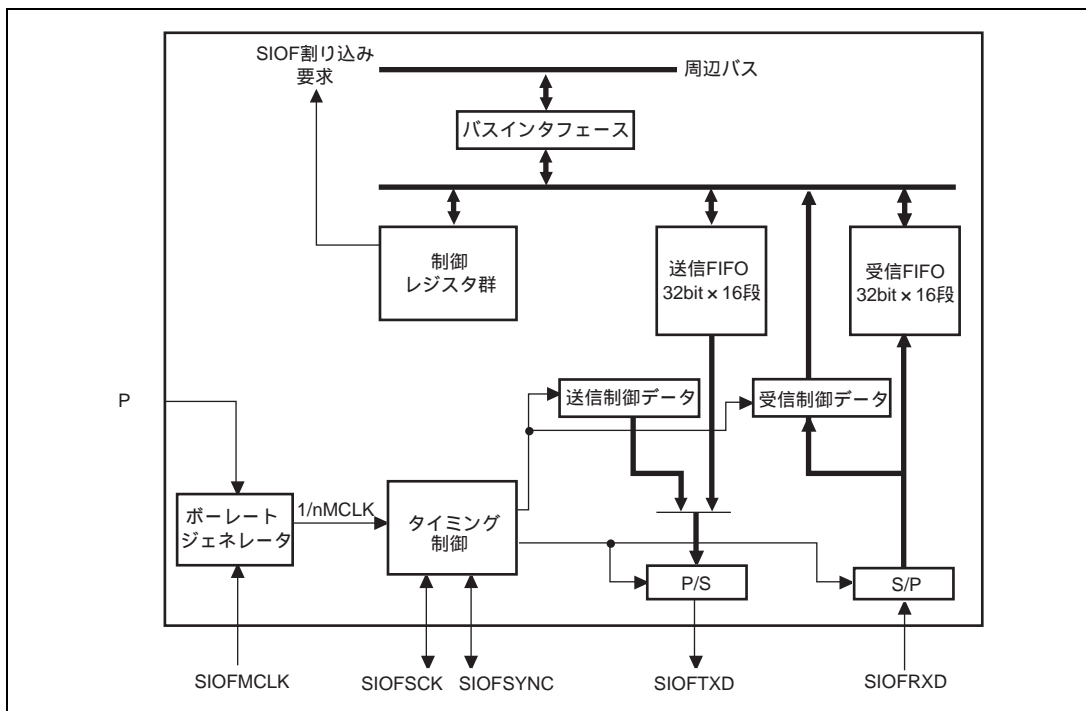


図 15.1 SIOF のブロック図

15.2 入出力端子

本モジュールの端子構成を表 15.1 に示します。

表 15.1 端子構成

端子名	略称*	入出力	機能
SIOF_MCLK	SIOFMCLK	入力	マスタクロック入力
SIOF_SCK (SCK)	SIOFSCK (SCK)	入出力	シリアルクロック (送受信共通) SPI モード時、出力固定
SIOF_SYNC (SIOF_SS0)	SIOFSYNC (SS0)	入出力	フレーム同期信号 (送受信共通) SPI モード時、出力固定かつスレプデバイス 0 を選択
(SIOF_SS1)	(SS1)	出力	SPI モード時、スレプデバイス 1 を選択
(SIOF_SS2)	(SS2)	出力	SPI モード時、スレプデバイス 2 を選択
SIOF_TXD (MOSI)	SIOFTXD (MOSI)	出力	送信データ
SIOF_RXD (MISO)	SIOFRXD (MISO)	入力	受信データ

【注】 * 本文中では略称 (SIOFMCLK、SIOFSCK、SIOFSYNC、SIOFTXD、SIOFRXD) を使用します。また、SPI モードでは、SCK、SS0、SS1、SS2、MOSI、MISO の名称を使用します。

15.3 レジスタの説明

本モジュールには、以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- モードレジスタ (SIMDR)
- コントロールレジスタ (SICTR)
- 送信データレジスタ (SITDR)
- 受信データレジスタ (SIRDR)
- 送信制御データレジスタ (SITCR)
- 受信制御データレジスタ (SIRCR)
- ステータスレジスタ (SISTR)
- 割り込み許可レジスタ (SIIER)
- FIFOコントロールレジスタ (SIFCTR)
- クロックセレクトレジスタ (SISCR)
- 送信データアサインレジスタ (SITDAR)
- 受信データアサインレジスタ (SIRDAR)
- 制御データアサインレジスタ (SICDAR)
- SPI制御レジスタ (SPICR)

15. FIFO 付きシリアルインタフェース (SIOF)

15.3.1 モードレジスタ (SIMDR)

SIMDR は、読み出し / 書き込み可能な 16 ビットのレジスタで、SIOF の動作モードを設定します。

ビット	ビット名	初期値	R/W	説明
15	TRMD1	1	R/W	転送モード 表 15.2 に示す転送モードを選択します。 00: スレープモード 1 01: スレープモード 2 10: マスタモード 1 11: マスタモード 2
14	TRMD0	0	R/W	
13	SYNCAT	0	R/W	SIOFSYNC 端子有効タイミング SIOFSYNC 信号を同期パルスで出力する場合の出力位置を示します。 0: フレームの先頭ビットデータ 1: スロットの最終ビットデータ
12	REDG	0	R/W	受信データサンプリングエッジ 0: SIOFRXD を SIOFSCK の立ち下がりエッジでサンプリングする。 (SIOFTXD は SIOFSCK の立ち上がりエッジで送出します。) 1: SIOFRXD を SIOFSCK の立ち上がりエッジでサンプリングする。 (SIOFTXD は SIOFSCK の立ち下がりエッジで送出します。) 【注】本ビットは、マスタモード時のみ有効となります。
11	FL3	0	R/W	フレーム長 00xx: データ長は 8 ビット、フレーム長は 8 ビット 0100: データ長は 8 ビット、フレーム長は 16 ビット 0101: データ長は 8 ビット、フレーム長は 32 ビット 0110: データ長は 8 ビット、フレーム長は 64 ビット 0111: データ長は 8 ビット、フレーム長は 128 ビット 10xx: データ長は 16 ビット、フレーム長は 16 ビット 1100: データ長は 16 ビット、フレーム長は 32 ビット 1101: データ長は 16 ビット、フレーム長は 64 ビット 1110: データ長は 16 ビット、フレーム長は 128 ビット 1111: データ長は 16 ビット、フレーム長は 256 ビット 【注】データ長 8 ビットを選択した場合、制御データの送受信は行えません。 x: Don't care
10	FL2	0	R/W	
9	FL1	0	R/W	
8	FL0	0	R/W	
7	TXDIZ	0	R/W	送信無効時* SIOFTXD 端子の出力 0: 無効時 1 出力 1: 無効時ハイインピーダンス状態 【注】* 無効時とは、ディスエーブル時および送信データ、制御データとして割り当てていないスロットを送出する場合です。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
6	RCIM	0	R/W	受信制御データ割り込みモード 0 : SIRCR レジスタの内容が変化したときに SISTR レジスタの RCRDY ビットをセットする。 1 : SIRCR レジスタへの制御データ受信タイミングごとに SISTR レジスタの RCRDY ビットをセットする。
5	SYNCAC	0	R/W	SIOFSYNC 端子極性 SIOFSYNC 信号をマスタモード時の同期パルスで出力する場合に有効となります。 0 : ハイアクティブ 1 : ローアクティブ
4	SYNCDL	0	R/W	SIOFSYNC 端子に対しデータ端子ビットの遅延 SIOFSYNC 信号が同期パルスの場合に有効となります。 またスレーブモードでの送信時は、1 ビット遅延のみ有効です。 0 : ビット遅延なし 1 : 1 ビット遅延
3~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

表 15.2 に、転送モードごとの動作を示します。

表 15.2 転送モードごとの動作

転送モード	マスタ/スレーブ	SIOFSYNC	ビット遅延	制御データ方式*
スレーブモード 1	スレーブ	同期パルス	SYNCDL ビット	スロット位置
スレーブモード 2	スレーブ	同期パルス		セカンダリ FS
マスタモード 1	マスタ	同期パルス		スロット位置
マスタモード 2	マスタ	L/R	なし	非対応

【注】 * 制御データ方式については、FL ビットに 1xxx を設定した場合に有効となります。(x : Don't care)

15. FIFO 付きシリアルインタフェース (SIOF)

15.3.2 コントロールレジスタ (SICTR)

SICTR は、読み出し / 書き込み可能な 16 ビットのレジスタで、SIOF の動作状態を設定します。

ビット	ビット名	初期値	R/W	説明
15	SCKE	0	R/W	シリアルクロック出力イネーブル 本ビットはマスタモード時に有効となります。 0 : SIOFSCK の出力を禁止 (0 を出力する) 1 : SIOFSCK の出力を許可 本ビットに 1 を設定すると、SIOF はポーレートジェネレータを初期化し、動作を開始すると同時に SIOFSCK にポーレートジェネレータで生成したクロックを出力します。 本ビットは、モジュールストップ時に初期化されます。
14	FSE	0	R/W	フレーム同期信号出力イネーブル 本ビットはマスタモード時に有効となります。 0 : SIOFSYNC の出力を禁止 (0 を出力する) 1 : SIOFSYNC の出力を許可 本ビットに 1 を設定すると、SIOF はフレームカウンタを初期化し、動作を開始します。 本ビットは、モジュールストップ時に初期化されます。
13~10		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
9	TXE	0	R/W	送信イネーブル 0 : SIOFTXD からのデータ送出手を禁止 1 : SIOFTXD からのデータ送出手を許可 • 本ビットへの設定は、次のフレーム先頭 (フレーム同期信号の立ち上がり) 時に有効となります。 • 本ビットへの 1 設定が有効になると、SIOF は SIFCTR レジスタの TFWM ビットの設定に従い、送信転送要求を発行します。送信 FIFO にデータが格納されると、SIOFTXD から送信データの送出手を開始します。 • 送信リセット時に初期化されます。 本ビットは、モジュールストップ時に初期化されます。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
8	RXE	0	R/W	<p>受信イネーブル</p> <p>0 : SIOFRXD からのデータ受信を禁止 1 : SIOFRXD からのデータ受信を許可</p> <ul style="list-style-type: none"> 本ビットへの設定は、次のフレーム先頭(フレーム同期信号の立ち上がり)時に有効となります。 本ビットへの1設定が有効になると、SIOFはSIOFRXDからの受信データの取り込みを開始します。受信FIFOにデータが格納されると、SIFCTRのRFWMビットの設定に従い、受信転送要求を発行します。 受信リセット時に初期化されます。 <p>本ビットは、モジュールストップ時に初期化されます。</p>
7~2		すべて0	R	<p>リザーブビット</p> <p>読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。</p>
1	TXRST	0	R/W	<p>送信リセット</p> <p>0 : 送信動作をリセットしない 1 : 送信動作をリセットする</p> <ul style="list-style-type: none"> 本ビットへの設定は、直ちに有効となります。また、初期化されるレジスタを設定する前に、本ビットに0を設定してください。 本ビットへの1設定が有効になると、SIOFは直ちにSIOFTXDからの送信データを1とし、送信データ系のレジスタ、送信関連のステータスを初期化します。初期化の対象は下記のレジスタとビットです。 <ol style="list-style-type: none"> SITDR レジスタ SITCR レジスタ 送信FIFOライトポインタ SISTR レジスタのTCRDY、TFEMP、TDREQ ビット 本ビットは、リセット動作を完了するとSIOFが自動的にクリアするため、読み出し時の値は常に0です。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
0	RXRST	0	R/W	受信リセット 0: 受信動作をリセットしない 1: 受信動作をリセットする <ul style="list-style-type: none"> 本ビットへの設定は、直ちに有効となります。また、初期化されるレジスタを設定する前に、本ビットに 0 を設定してください。 本ビットへの 1 設定が有効になると、SIOF は SIOFRXD からの受信を停止し、受信データ系のレジスタ、受信関連のステータスを初期化します。初期化の対象は下記のレジスタとビットです。 <ol style="list-style-type: none"> SIRDR レジスタ SIRCR レジスタ 受信 FIFO リードポインタ SISTR レジスタの RCRDY、RFFUL、RDREQ ビット 本ビットは、リセット動作を完了すると SIOF が自動的にクリアするため、読み出し時の値は常に 0 です。

15.3.3 送信データレジスタ (SITDR)

SITDR は、書き込み専用の 32 ビットのレジスタで、SIOF の送信データを設定します。

本レジスタは、「第 27 章 レジスタ一覧」に示す初期化条件のほか、SICTR の TXRST ビットにより送信リセットされた場合も、初期化されます。

本レジスタは、モジュールストップ時に初期化されます。

ビット	ビット名	初期値	R/W	説明
31~16	SITDL 15~0	すべて 0	W	左 ch.送信データ 左 ch.データとして SIOFTXD から送出するデータを設定します。送信フレームにおける左 ch.データの位置は、SITDAR の TDLA ビットへの設定値に従います。 <ul style="list-style-type: none"> 本ビットは SITDAR の TDLE ビットに 1 を設定した場合に有効となります。
15~0	SITDR 15~0	すべて 0	W	右 ch.送信データ 右 ch.データとして SIOFTXD から送出するデータを設定します。送信フレームにおける右 ch.データの位置は、SITDAR の TDRA ビットへの設定値に従います。 <ul style="list-style-type: none"> 本ビットは SITDAR の TDRE ビットに 1 で、かつ SITDAR の TLREP ビットに 0 を設定した場合に有効となります。

15.3.4 受信データレジスタ (SIRDR)

SIRDR は、読み出し専用の 32 ビットのレジスタで、SIOF の受信データの読み出しを行います。本レジスタには受信 FIFO のデータが格納されます。

本レジスタは、「第 27 章 レジスタ一覧」に示す初期化条件のほか、SICTR の RXRST ビットにより受信リセ

ットされた場合も、初期化されます。

ビット	ビット名	初期値	R/W	説 明
31~16	SIRD 15~0	すべて0	R	左 ch.受信データ 左 ch.データとして SIOFRXD から受信したデータを格納します。受信フレームにおける左 ch.データの位置は、SIRDAR の RDLA ビットへの設定値に従います。 • 本ビットは SIRDAR の RDLE ビットに 1 を設定した場合に有効となります。
15~0	SIRD 15~0	すべて0	R	右 ch.受信データ 右 ch.データとして SIOFRXD から受信したデータを格納します。受信フレームにおける右 ch.データの位置は、SIRDAR の RDRA ビットへの設定値に従います。 • 本ビットは SIRDAR の RDRE ビットに 1 を設定した場合に有効となります。

15.3.5 送信制御データレジスタ (SITCR)

SITCR は、読み出し/書き込み可能な 32 ビットのレジスタで、SIOF の送信制御データを設定します。本レジスタへの設定は、SIMDR の FL ビットに 1xxx (x : Don't care) を設定したときに有効となります。

本レジスタは、「第 27 章 レジスタ一覧」に示す初期化条件のほか、SICTR の TXRST ビットにより送信リセットされた場合も、初期化されます。

本レジスタは、モジュールストップ時に初期化されます。

ビット	ビット名	初期値	R/W	説 明
31~16	SITC0 15~0	すべて0	R/W	制御 0ch.送信データ 制御 0ch.送信データとして SIOFTXD から送出するデータを設定します。送受信フレームにおける制御 0ch.データの位置は、SICDAR の CD0A ビットへの設定値に従います。 • 本ビットは SICDAR の CD0E ビットに 1 を設定した場合に有効となります。
15~0	SITC1 15~0	すべて0	R/W	制御 1ch.送信データ 制御 1ch.送信データとして SIOFTXD から送出するデータを設定します。送受信フレームにおける制御 1ch.データの位置は、SICDAR の CD1A ビットへの設定値に従います。 • 本ビットは SICDAR の CD1E ビットに 1 を設定した場合に有効となります。

15. FIFO 付きシリアルインタフェース (SIOF)

15.3.6 受信制御データレジスタ (SIRCR)

SIRCR は、読み出し/書き込み可能な 32 ビットのレジスタで、SIOF の受信制御データが格納されます。本レジスタへの設定は、SIMDR の FL ビットに 1xxx (x : Don't care) を設定したときに有効となります。

本レジスタは、「第 27 章 レジスタ一覧」に示す初期化条件のほか、SICTR の RXRST ビットにより受信リセットされた場合も、初期化されます。

ビット	ビット名	初期値	R/W	説明
31~16	SIRC0 15~0	すべて 0	R/W	制御 0ch.受信データ 制御 0ch.受信データとして SIOFRXD から受信したデータを格納します。 送受信フレームにおける制御 0ch.データの位置は、SICDAR の CD0A ビットへの設定値に従います。 • 本ビットは SICDAR の CD0E ビットに 1 を設定した場合に有効となります。
15~0	SIRC1 15~0	すべて 0	R/W	制御 1ch.受信データ 制御 1ch.受信データとして SIOFRXD から受信したデータを格納します。 送受信フレームにおける制御 1ch.データの位置は、SICDAR の CD1A ビットへの設定値に従います。 • 本ビットは SICDAR の CD1E ビットに 1 を設定した場合に有効となります。

15.3.7 ステータスレジスタ (SISTR)

SISTR は、読み出し専用の 16 ビットのレジスタで、SIOF の状態を表示します。本レジスタの各ビットは、SIHER レジスタの対応するビットに 1 を設定した場合に、SIOF の割り込み要因となります。

本レジスタは、モジュールストップ時に初期化されます。

ビット	ビット名	初期値	R/W	説明
15		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
14	TCRDY	0	R	送信制御データレディ 0 : SITCR レジスタへの書き込みが可能でない 1 : SITCR レジスタへの書き込みが可能である • 本ビットが 0 時に SITCR への書き込みを行うと SICTR は上書きされ、以前の内容は SIOFTXD から送出されません。 • 本ビットは SICTR の TXE ビットが 1 のときに有効となります。 • 本ビットは状態を表示するビットであり、SITCR への書き込みを行うと SIOF がクリアします。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
13	TFEMP	0	R	<p>送信 FIFO エンプティ</p> <p>0: 送信 FIFO が空でない 1: 送信 FIFO が空である</p> <ul style="list-style-type: none"> 本ビットは SICTR の TXE ビットが 1 のときに有効となります。 本ビットは状態を表示するビットであり、SITDR への書き込みが行われると SIOF がクリアします。 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
12	TDREQ	0	R	<p>送信データ転送リクエスト</p> <p>0: 送信 FIFO の空き領域が SIFCTR の TFWM ビットによる設定値を超えていない 1: 送信 FIFO の空き領域が SIFCTR の TFWM ビットによる設定値を超えた</p> <p>送信データ転送リクエストは、送信 FIFO の空き領域が SIFCTR の TFWM ビットによる設定値を超えたときに発行します。</p> <p>DMAC による送信データ転送を用いる場合には、DMAC による 1 回のアクセスによって、本ビットは必ずクリアされます。DMAC アクセス後も本ビットの設定条件が満たされている場合には、SIOF は再び本ビットに 1 を表示します。</p> <ul style="list-style-type: none"> 本ビットは SICTR の TXE ビットが 1 のときに有効となります。 本ビットは状態を表示するビットであり、送信 FIFO の空き領域が SIFCTR の TFWM ビットによる設定値よりも少なくなると SIOF がクリアします。 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
11		0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>
10	RCRDY	0	R	<p>受信制御データレディ</p> <p>0: SIRCR レジスタに有効データが格納されていない 1: SIRCR レジスタに有効データが格納されている</p> <ul style="list-style-type: none"> 本ビットが 1 のときに再び SIRCR への有効データの書き込みが発生した場合、SIRCR には最新のデータが上書きされます。 本ビットは SICTR の RXE ビットが 1 のときに有効となります。 本ビットは状態を表示するビットであり、SIRCR の読み出しを行うと SIOF がクリアします。 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
9	RFFUL	0	R	<p>受信 FIFO フル</p> <p>0: 受信 FIFO が満杯でない 1: 受信 FIFO が満杯である</p> <ul style="list-style-type: none"> • 本ビットは SICTR の RXE ビットが 1 のときに有効となります。 • 本ビットは状態を表示するビットであり、SIRDR の読み出しを行うと SIOF がクリアします。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
8	RDREQ	0	R	<p>受信データ転送リクエスト</p> <p>0: 受信 FIFO の有効領域が SIFCTR の RFWM ビットによる設定値を超えていない 1: 受信 FIFO の有効領域が SIFCTR の RFWM ビットによる設定値を超えた</p> <p>受信データ転送リクエストは、受信 FIFO の有効領域が SIFCTR の RFWM ビットによる設定値を超えたときに発行します。</p> <p>DMAC による受信データ転送を用いる場合には、DMAC による 1 回のアクセスによって、本ビットは必ずクリアされます。DMAC アクセス後も本ビットの設定条件が満たされている場合には、SIOF は再び本ビットに 1 を表示します。</p> <ul style="list-style-type: none"> • 本ビットは SICTR の RXE ビットが 1 のときに有効となります。 • 本ビットは状態を表示するビットであり、受信 FIFO の有効領域が SIFCTR の RFWM ビットによる設定値よりも少なくなると SIOF がクリアします。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
7、6		すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
5	SAERR	0	R/W	<p>スロットアサインエラー</p> <p>0 : スロットアサインエラー未発生 1 : スロットアサインエラー発生</p> <p>スロットアサインエラーとは、SITDAR、SIRDAR、SICDAR の設定に重複があったことを表します。</p> <p>スロットアサインエラー発生時、SIOF は SIOFTXD への送信および SIOFRXD からの受信を行いません。ただし、SICTR の TXE ビット、RXE ビットのクリアは行いません。</p> <ul style="list-style-type: none"> • 本ビットは SICTR の TXE ビットもしくは RXE ビットが 1 のときに有効となります。 • 本ビットに 1 を書き込むと、内容がクリアされます。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
4	FSERR	0	R/W	<p>フレーム同期エラー</p> <p>0 : フレーム同期エラー未発生 1 : フレーム同期エラー発生</p> <p>フレーム同期エラーとは、転送データ、制御データの送受信完了前に、次のフレーム同期タイミングとなったことを表します。</p> <p>フレーム同期エラー発生時、SIOF は転送可能なスロットに対してのみ送受信を行います。</p> <ul style="list-style-type: none"> • 本ビットは SICTR の TXE ビットもしくは RXE ビットが 1 のときに有効となります。 • 本ビットに 1 を書き込むと、内容がクリアされます。0 書き込みは無効です。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
3	TFOVF	0	R/W	<p>送信 FIFO オーバフロー</p> <p>0 : 送信 FIFO オーバフロー未発生 1 : 送信 FIFO オーバフロー発生</p> <p>送信 FIFO オーバフローとは、送信 FIFO が満杯時に SITDR への書き込みが発生したことを表します。</p> <p>送信 FIFO オーバフロー時、SIOF はオーバフローとなった書き込みを無効とします。</p> <ul style="list-style-type: none"> • 本ビットは SICTR の TXE ビットが 1 のときに有効となります。 • 本ビットに 1 を書き込むと、内容がクリアされます。0 書き込みは無効です。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
2	TFUDF	0	R/W	<p>送信 FIFO アンダフロー</p> <p>0: 送信 FIFO アンダフロー未発生 1: 送信 FIFO アンダフロー発生</p> <p>送信 FIFO アンダフローとは、送信 FIFO がエンプティ時に送信動作によるロードが発生したことを表します。</p> <p>送信 FIFO アンダフロー時、SIOF は前回送出データを繰り返して送出します。</p> <ul style="list-style-type: none"> • 本ビットは SICTR の TXE ビットが 1 のときに有効となります。 • 本ビットに 1 を書き込むと、内容がクリアされます。0 書き込みは無効です。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
1	RFUDF	0	R/W	<p>受信 FIFO アンダフロー</p> <p>0: 受信 FIFO アンダフロー未発生 1: 受信 FIFO アンダフロー発生</p> <p>受信 FIFO アンダフローとは、受信 FIFO が空時に SIRDR の読み出しが発生したことを表します。</p> <p>受信 FIFO アンダフロー時、SIRDR から読み出したデータの値は保証しません。</p> <ul style="list-style-type: none"> • 本ビットは SICTR の RXE ビットが 1 のときに有効となります。 • 本ビットに 1 を書き込むと、内容がクリアされます。0 書き込みは無効です。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。
0	RFOVF	0	R/W	<p>受信 FIFO オーバフロー</p> <p>0: 受信 FIFO オーバフロー未発生 1: 受信 FIFO オーバフロー発生</p> <p>受信 FIFO オーバフローとは、受信 FIFO がフル時に受信動作による書き込みが発生したことを表します。</p> <p>受信 FIFO オーバフロー時、オーバフローとなった受信データは消失します。</p> <ul style="list-style-type: none"> • 本ビットは SICTR の RXE ビットが 1 のときに有効となります。 • 本ビットに 1 を書き込むと、内容がクリアされます。0 書き込みは無効です。 • 本ビットによる割り込み発行が許可されている場合、SIOF 割り込みを発行します。

15.3.8 割り込み許可レジスタ (SIER)

SIER は、読み出し / 書き込み可能な 16 ビットのレジスタで、SIOF 割り込みの発行を許可します。本レジスタの各ビットに 1 を設定した場合に、SISTR の対応するビットに 1 がセットされると、SIOF は割り込みを発行します。

ビット	ビット名	初期値	R/W	説明
15	TDMAE	0	R/W	送信データ DMA 転送要求イネーブル 割り込みを CPU への割り込み / DMA 転送要求として伝えます。要因としては、TDREQE が設定できます。 0 : CPU への割り込みとして使用 1 : DMAC への DMA 転送要求として使用
14	TCRDYE	0	R/W	送信制御データレディイネーブル 0 : 送信制御データレディによる割り込みを禁止 1 : 送信制御データレディによる割り込みを許可
13	TFEMPE	0	R/W	送信 FIFO エンプティイネーブル 0 : 送信エンプティによる割り込みを禁止 1 : 送信エンプティによる割り込みを許可
12	TDREQE	0	R/W	送信データ転送リクエストイネーブル 0 : 送信データ転送リクエストによる割り込みを禁止 1 : 送信データ転送リクエストによる割り込みを許可
11	RDMAE	0	R/W	受信データ DMA 転送要求イネーブル 割り込みを CPU への割り込み / DMA 転送要求として伝えます。要因としては、RDREQE が設定できます。 0 : CPU への割り込みとして使用 1 : DMAC への DMA 転送要求として使用
10	RCRDYE	0	R/W	受信制御データレディイネーブル 0 : 受信制御データレディによる割り込みを禁止 1 : 受信制御データレディによる割り込みを許可
9	RFFULE	0	R/W	受信 FIFO フルイネーブル 0 : 受信 FIFO フルによる割り込みを禁止 1 : 受信 FIFO フルによる割り込みを許可
8	RDREQE	0	R/W	受信データ転送リクエストイネーブル 0 : 受信データ転送リクエストによる割り込みを禁止 1 : 受信データ転送リクエストによる割り込みを許可
7, 6		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
5	SAERRE	0	R/W	スロットアサインエラーイネーブル 0: スロットアサインエラーによる割り込みを禁止 1: スロットアサインエラーによる割り込みを許可
4	FSERRE	0	R/W	フレーム同期エラーイネーブル 0: フレーム同期エラーによる割り込みを禁止 1: フレーム同期エラーによる割り込みを許可
3	TFOVFE	0	R/W	送信 FIFO オーバフローイネーブル 0: 送信 FIFO オーバフローによる割り込みを禁止 1: 送信 FIFO オーバフローによる割り込みを許可
2	TFUDFE	0	R/W	送信 FIFO アンダフローイネーブル 0: 送信 FIFO アンダフローによる割り込みを禁止 1: 送信 FIFO アンダフローによる割り込みを許可
1	RFUDFE	0	R/W	受信 FIFO アンダフローイネーブル 0: 受信 FIFO アンダフローによる割り込みを禁止 1: 受信 FIFO アンダフローによる割り込みを許可
0	RFOVFE	0	R/W	受信 FIFO オーバフローイネーブル 0: 受信 FIFO オーバフローによる割り込みを禁止 1: 受信 FIFO オーバフローによる割り込みを許可

15.3.9 FIFO コントロールレジスタ (SIFCTR)

SIFCTR は、読み出し / 書き込み可能な 16 ビットのレジスタで、送受信 FIFO の転送が可能なエリアを表示します。

ビット	ビット名	初期値	R/W	説明
15	TFWM2	0	R/W	送信 FIFO ウォータマーク
14	TFWM1	0	R/W	000: 送信 FIFO の空き領域が 16 段のときに転送要求を発行する
13	TFWM0	0	R/W	001: 設定禁止 010: 設定禁止 011: 設定禁止 100: 送信 FIFO の空き領域が 12 段以上のときに転送要求を発行する 101: 送信 FIFO の空き領域が 8 段以上のときに転送要求を発行する 110: 送信 FIFO の空き領域が 4 段以上のときに転送要求を発行する 111: 送信 FIFO の空き領域が 1 段以上のときに転送要求を発行する • 送信 FIFO の転送要求は、SISTR の TDREQE ビットで行います。 • 本ビットへの設定にかかわらず、送信 FIFO は常に 16 段の FIFO として動作を行います。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明
12	TFUA4	1	R	送信 FIFO 使用可能エリア CPU または DMAC が転送可能な語数を B'00000 (満杯) ~ B'10000 (空) で表示します。
11	TFUA3	0	R	
10	TFUA2	0	R	
9	TFUA1	0	R	
8	TFUA0	0	R	
7	RFWM2	0	R/W	受信 FIFO ウォータマーク 000 : 受信 FIFO の有効データ領域が 1 段以上のときに転送要求を発行する 001 : 設定禁止 010 : 設定禁止 011 : 設定禁止 100 : 受信 FIFO の有効データ領域が 4 段以上のときに転送要求を発行する 101 : 受信 FIFO の有効データ領域が 8 段以上のときに転送要求を発行する 110 : 受信 FIFO の有効データ領域が 12 段以上のときに転送要求を発行する 111 : 受信 FIFO の有効データ領域が 16 段のときに転送要求を発行する • 受信 FIFO の転送要求は、SISTR の RDREQE ビットで行います。 • 本ビットへの設定にかかわらず、受信 FIFO は常に 16 段の FIFO として動作を行います。
6	RFWM1	0	R/W	
5	RFWM0	0	R/W	
4	RFUA4	0	R	
3	RFUA3	0	R	
2	RFUA2	0	R	受信 FIFO 使用可能エリア CPU または DMAC が転送可能な語数を B'00000 (空) ~ B'10000 (満杯) で表示します。
1	RFUA1	0	R	
0	RFUA0	0	R	

15. FIFO 付きシリアルインタフェース (SIOF)

15.3.10 クロックセレクトレジスタ (SISCR)

SISCR は、読み出し / 書き込み可能な 16 ビットのレジスタで、マスタ時のシリアルクロック生成条件を設定します。本レジスタへの設定は、SIMDR の TRMD1、TRMD0 ビットに B'10 もしくは B'11 が設定されているときに有効です。

ビット	ビット名	初期値	R/W	説明
15	MSSEL	1	R/W	マスタクロックソース選択 0 : マスタクロックとして SIOFMCLK 端子入力信号を使用 1 : マスタクロックとして P を使用 マスタクロックとは、ボーレートジェネレータに入力するクロックを指します。
14	MSIMM	1	R/W	マスタクロック直接選択 0 : シリアルクロックとしてボーレートジェネレータ出力クロックを使用 1 : シリアルクロックとしてマスタクロックをそのまま使用
13		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
12	BRPS4	0	R/W	プリスケアラ設定 ボーレートジェネレータのプリスケアラのカウンタ値によりマスタクロックの分周比を設定します。 設定の範囲は B'00000 (× 1/1) ~ B'11111 (× 1/32) となります。
11	BRPS3	0	R/W	
10	BRPS2	0	R/W	
9	BRPS1	0	R/W	
8	BRPS0	0	R/W	
7~3		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
2	BRDV2	0	R/W	ボーレートジェネレータ分周比 ボーレートジェネレータにおける出力段の分周比を設定します。 000 : プリスケアラ出力 × 1/2 001 : プリスケアラ出力 × 1/4 010 : プリスケアラ出力 × 1/8 011 : プリスケアラ出力 × 1/16 100 : プリスケアラ出力 × 1/32 101 : 設定禁止 110 : 設定禁止 111 : プリスケアラ出力 × 1/1* • ボーレートジェネレータの最終分周比は、BRPS × BRDV で決定します(最大 1/1024)。 【注】 * 本設定は、BRPS[4:0] = B'00000 のときのみ有効です。
1	BRDV1	0	R/W	
0	BRDV0	0	R/W	

15.3.11 送信データアサインレジスタ (SITDAR)

SITDAR は、読み出し / 書き込み可能な 16 ビットのレジスタで、フレーム内の送信データ位置 (スロット No.) を設定します。

ビット	ビット名	初期値	R/W	説明
15	TDLE	0	R/W	送信左 ch. データイネーブル 0 : 左 ch. データの送信を許可しない 1 : 左 ch. データの送信を許可する
14 ~ 12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
11 10 9 8	TDLA3 TDLA2 TDLA1 TDLA0	0 0 0 0	R/W R/W R/W R/W	送信左 ch. データアサイン 3 ~ 0 送信フレームにおける左 ch. データの位置を B'0000 (0) から B'1110 (14) で設定します。 1111 : 設定禁止 • 左 ch. データの送信データは SITDR の SITDL ビットに設定します。
7	TDRE	0	R/W	送信右 ch. データイネーブル 0 : 右 ch. データの送信を許可しない 1 : 右 ch. データの送信を許可する
6	TLREP	0	R/W	送信左 ch. リピート 0 : 右 ch. データとして SITDR の SITDR ビット設定値を送出する。 1 : 右 ch. データとして SITDR の SITDL ビット設定値を繰り返して送 出する。 • 本ビットの設定は TDRE ビットが 1 のとき有効となります。 • ビットに 1 を設定した場合、SITDR ビットへの設定は無視されます。
5, 4		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
3 2 1 0	TDRA3 TDRA2 TDRA1 TDRA0	0 0 0 0	R/W R/W R/W R/W	送信右 ch. データアサイン 送信フレームにおける右 ch. データの位置を B'0000 (0) から B'1110 (14) で設定します。 1111 : 設定禁止 • 右 ch. データの送信データは SITDR の SITDR ビットに設定します。

15. FIFO 付きシリアルインタフェース (SIOF)

15.3.12 受信データアサインレジスタ (SIRDAR)

SIRDAR は、読み出し / 書き込み可能な 16 ビットのレジスタで、フレーム内の受信データ位置 (スロット No.) を設定します。

ビット	ビット名	初期値	R/W	説明
15	RDLE	0	R/W	受信左 ch. データイネーブル 0 : 左 ch. データの受信を許可しない 1 : 左 ch. データの受信を許可する
14 ~ 12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
11	RDLA3	0	R/W	受信左 ch. データアサイン 受信フレームにおける左 ch. データの位置を B'0000 (0) から B'1110 (14) で設定します。 1111 : 設定禁止 • 左 ch. データの受信データは SIRDAR の SIRDLE ビットに格納されます。
10	RDLA2	0	R/W	
9	RDLA1	0	R/W	
8	RDLA0	0	R/W	
7	RDRE	0	R/W	受信右 ch. データイネーブル 0 : 右 ch. データの受信を許可しない 1 : 右 ch. データの受信を許可する
6 ~ 4		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
3	RDRA3	0	R/W	受信右 ch. データアサイン 受信フレームにおける右 ch. データの位置を B'0000 (0) から B'1110 (14) で設定します。 1111 : 設定禁止 • 右 ch. データの受信データは SIRDAR の SIRDRE ビットに格納されます。
2	RDRA2	0	R/W	
1	RDRA1	0	R/W	
0	RDRA0	0	R/W	

15.3.13 制御データアサインレジスタ (SICDAR)

SICDAR は、読み出し / 書き込み可能な 16 ビットのレジスタで、フレーム内の制御データ位置 (スロット No.) を設定します。本レジスタへの設定は、SIMDR の FL ビットに 1xxx (x : Don't care) を設定したときに有効となります。

ビット	ビット名	初期値	R/W	説明
15	CD0E	0	R/W	制御 0ch. データイネーブル 0 : 制御 0ch. データの送受信を許可しない 1 : 制御 0ch. データの送受信を許可する
14~12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
11	CD0A3	0	R/W	制御 0ch. データアサイン 送受信フレームにおける制御 0ch. データの位置を B'0000 (0) から B'1110 (14) で設定します。 1111 : 設定禁止 • 制御 0ch. データの送信データは SITCR の SITD0 ビットに設定します。 • 制御 0ch. データの受信データは SIRCR の SIRD0 ビットに格納されます。
10	CD0A2	0	R/W	
9	CD0A1	0	R/W	
8	CD0A0	0	R/W	
7	CD1E	0	R/W	制御 1ch. データイネーブル 0 : 制御 1ch. データの送受信を許可しない 1 : 制御 1ch. データの送受信を許可する
6~4		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
3	CD1A3	0	R/W	制御 1ch. データアサイン 送受信フレームにおける制御 1ch. データの位置を B'0000 (0) から B'1110 (14) で設定します。 1111 : 設定禁止 • 制御 1ch. データの送信データは SITCR の SITD1 ビットに設定します。 • 制御 1ch. データの受信データは SIRCR の SIRD1 ビットに格納されます。
2	CD1A2	0	R/W	
1	CD1A1	0	R/W	
0	CD1A0	0	R/W	

15. FIFO 付きシリアルインタフェース (SIOF)

15.3.14 SPI 制御レジスタ (SPICR)

SPICR は、読み出し / 書き込み可能な 16 ビットのレジスタで、SPI の動作モードを設定します。

ビット	ビット名	初期値	R/W	説明
15	SPIM	0	R/W	SPI モード SIOF の動作モードを選択します。 0 : SIOF として動作します。 1 : SIOF は、SPI のマスタモードで動作します。
14		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
13	CPHA	0	R/W	SPI クロックフェーズ SPI のクロックフェーズを選択します。 0 : SCK の第 1 エッジでデータをサンプリングします。 1 : SCK の第 2 エッジでデータをサンプリングします。
12	CPOL	0	R/W	SPI クロック極性 SPI のクロック極性を選択します。 0 : SCK はアクティブ・ハイクロックで、アイドル状態でローになります。 1 : SCK はアクティブ・ローロックで、アイドル状態でハイになります。
11		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
10	SS2E*	0	R/W	スレーブデバイス 2 (SS2) イネーブル 0 : スレーブデバイス 2 を選択しません 1 : スレーブデバイス 2 を選択します
9	SS1E*	0	R/W	スレーブデバイス 1 (SS1) イネーブル 0 : スレーブデバイス 1 を選択しません 1 : スレーブデバイス 1 を選択します
8	SS0E*	0	R/W	スレーブデバイス 0 (SS0) イネーブル 0 : スレーブデバイス 0 を選択しません 1 : スレーブデバイス 0 を選択します
7, 6		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

15. FIFO 付きシリアルインタフェース (SIOF)

ビット	ビット名	初期値	R/W	説明																														
5 4	SSAST1 SSAST0	0 0	R/W R/W	<p>SS アサートのセットアップ</p> <p>SS の SCK に対するセットアップ・タイミングを設定します。</p> <p>CPHA = 0 の場合 (単位: SCK クロック)</p> <table border="1"> <thead> <tr> <th>SSAST[1:0]</th> <th>SS セットアップ</th> <th>SS ホールド</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>0.5 クロック</td> <td>0 クロック</td> </tr> <tr> <td>01</td> <td>1 クロック</td> <td>0.5 クロック</td> </tr> <tr> <td>10</td> <td>1.5 クロック</td> <td>1 クロック</td> </tr> <tr> <td>11</td> <td>2 クロック</td> <td>1.5 クロック</td> </tr> </tbody> </table> <p>CPHA = 1 の場合 (単位: SCK クロック)</p> <table border="1"> <thead> <tr> <th>SSAST[1:0]</th> <th>SS セットアップ</th> <th>SS ホールド</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>0 クロック</td> <td>0.5 クロック</td> </tr> <tr> <td>01</td> <td>0.5 クロック</td> <td>1 クロック</td> </tr> <tr> <td>10</td> <td>1 クロック</td> <td>1.5 クロック</td> </tr> <tr> <td>11</td> <td>1.5 クロック</td> <td>2 クロック</td> </tr> </tbody> </table>	SSAST[1:0]	SS セットアップ	SS ホールド	00	0.5 クロック	0 クロック	01	1 クロック	0.5 クロック	10	1.5 クロック	1 クロック	11	2 クロック	1.5 クロック	SSAST[1:0]	SS セットアップ	SS ホールド	00	0 クロック	0.5 クロック	01	0.5 クロック	1 クロック	10	1 クロック	1.5 クロック	11	1.5 クロック	2 クロック
SSAST[1:0]	SS セットアップ	SS ホールド																																
00	0.5 クロック	0 クロック																																
01	1 クロック	0.5 クロック																																
10	1.5 クロック	1 クロック																																
11	2 クロック	1.5 クロック																																
SSAST[1:0]	SS セットアップ	SS ホールド																																
00	0 クロック	0.5 クロック																																
01	0.5 クロック	1 クロック																																
10	1 クロック	1.5 クロック																																
11	1.5 クロック	2 クロック																																
3, 2		すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。</p>																														
1 0	FLD1 FLD0	0 0	R/W R/W	<p>フレーム遅延</p> <p>フレーム間隔のアイドル状態の最小時間を SCK クロック数で指定します。</p> <p>00: SCK の 0 クロック遅延 \overline{SS} のローアサートを継続した状態で連続 SPI 通信します。</p> <p>01: SCK の 1 クロック遅延 10: SCK の 2 クロック遅延 11: SCK の 3 クロック遅延</p>																														

【注】 * SS0E、SS1E、SS2E ビットを同時に 1 に設定しないでください。

15.4 動作説明

15.4.1 シリアルクロック

(1) マスタ/スレーブ

SIOF のクロックモードとして下記の 2 モードがあります。

- スレーブモード : SIOFSCK、SIOFSYNCは入力
- マスタモード : SIOFSCK、SIOFSYNCは出力

(2) ポーレートジェネレータ

SIOF マスタ時には、ポーレートジェネレータ (BRG) を用いてシリアルクロックを生成します。ポーレートジェネレータの分周比は、1/1 ~ 1/1024 です。

図 15.2 にシリアルクロック供給系統図を示します。

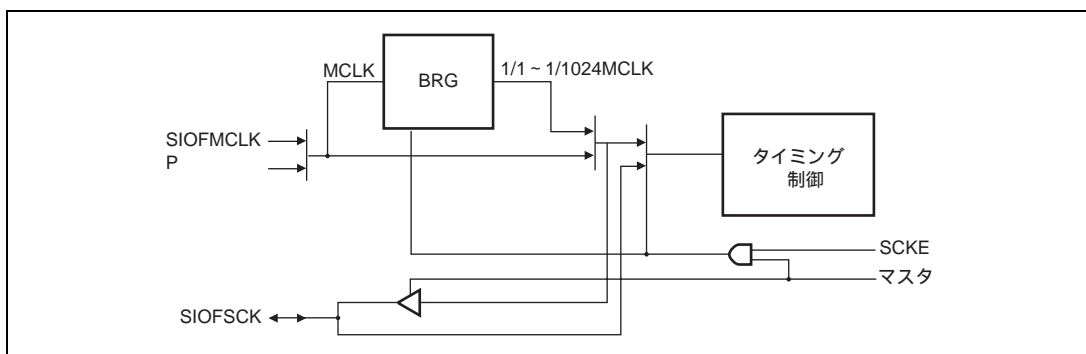


図 15.2 シリアルクロック供給

また、表 15.3 にシリアルクロック周波数の例を示します。

表 15.3 SIOF シリアルクロック周波数例

フレーム長	サンプリングレート		
	8kHz	44.1kHz	48kHz
32 ビット	256kHz	1.4112MHz	1.536MHz
64 ビット	512kHz	2.8224MHz	3.072MHz
128 ビット	1.024MHz	5.6448MHz	6.144MHz
256 ビット	2.048MHz	11.289MHz	12.289MHz

15.4.2 シリアルタイミング

(1) SIOFSYNC

SIOFSYNC はフレーム同期信号 (FS) です。転送モードによって下記 2 通りの機能を有します。

- 同期パルス：フレーム先頭を表す1ビット幅のパルス
- L/R：ステレオデータの左ch. (L) をハイレベル、右ch. (R) をローレベルで表す1/2フレーム幅のパルス

図 15.3 に SIOFSYNC による同期タイミングを示します。

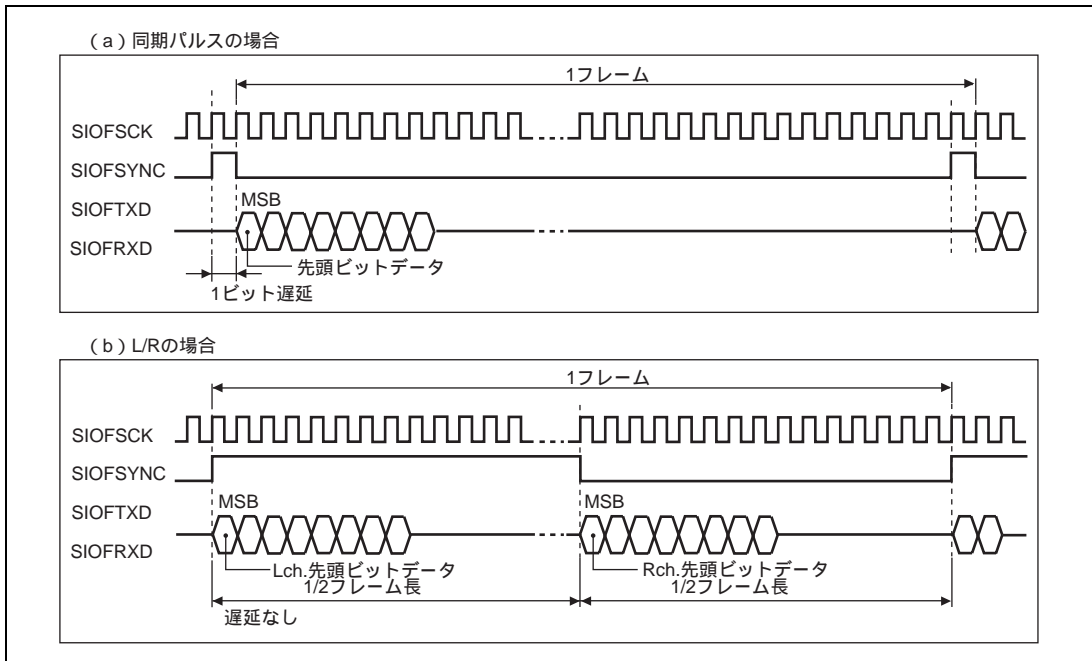


図 15.3 シリアルデータ同期タイミング

15. FIFO 付きシリアルインタフェース (SIOF)

(2) 送受信タイミング

SIOFSCK に対する SIOFTXD の送信タイミングおよび SIOFRXD の受信タイミングは、サンプリングタイミングとして下記 2 通りの設定が可能です。送受信タイミングの設定は、SIMDR の REDG ビットに行います。

- 立ち下がりサンプリング
- 立ち上がりサンプリング

図 15.4 に送受信タイミングを示します。

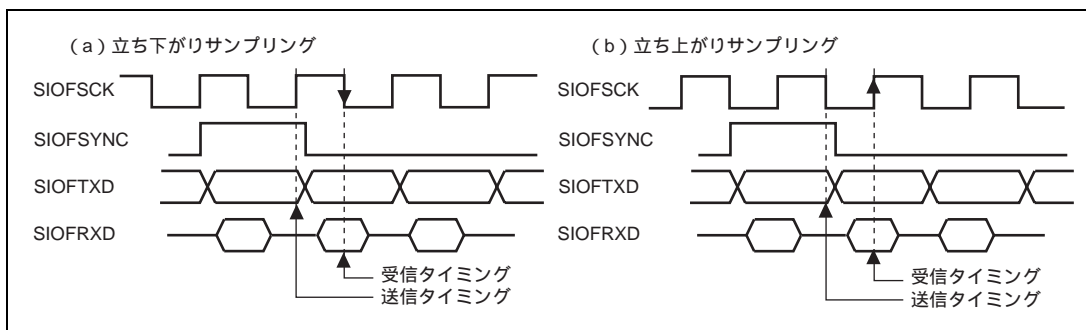


図 15.4 SIOF 送受信タイミング

15.4.3 転送データフォーマット

SIOF は、下記の転送を行います。

- 送受信データ 8ビット / 16ビット / 16ビットステレオの転送データ
- 制御データ 16ビット長 (専用レジスタを用いてインタフェースを行います)

(1) 転送モード

SIOF は、転送モードとして表 15.4 に示すように 4 通りのモードを持っています。転送モードの設定は SIMDR の TRMD1 ~ TRMD0 ビットに行います。

表 15.4 シリアル転送モード

転送モード	SIOFSYNC	ビット遅延	制御データ
スレーブモード 1	同期パルス	SYNCDL ビット	スロット位置
スレーブモード 2	同期パルス		セカンダリ FS
マスタモード 1	同期パルス		スロット位置
マスタモード 2	L/R	なし	非対応

(2) フレーム長

SIOF が転送を行うフレームの長さは SIMDR の FL3 ~ FL0 ビットに対して設定を行います。表 15.5 に設定値とフレーム長の関係を示します。

表 15.5 フレーム長

FL3~FL0	スロット長	1 フレームあたりのビット	対応する転送データ
00xx	8	8	8 ビットモノラル
0100	8	16	8 ビットモノラル
0101	8	32	8 ビットモノラル
0110	8	64	8 ビットモノラル
0111	8	128	8 ビットモノラル
10xx	16	16	16 ビットモノラル
1100	16	32	16 ビットモノラル/ステレオ
1101	16	64	16 ビットモノラル/ステレオ
1110	16	128	16 ビットモノラル/ステレオ
1111	16	256	16 ビットモノラル/ステレオ

【注】 x : Don't care

(3) スロット位置

SIOF は、1 フレームにおける送信データ、受信データ、制御データ (送受信共通) の位置をスロット番号でそれぞれ個別に設定することが可能です。設定は下記レジスタに対して行います。

- 送信データ : SITDAR
- 受信データ : SIRDAR
- 制御データ : SICDAR

制御データは、データ長 16 ビットのときのみ有効となります。また、制御データは送信と受信で必ず同じスロットに割り当てられます。

15.4.4 転送データのレジスタ割り付け

(1) 送受信データ

送受信データの書き込み/読み出しは下記レジスタに対して行います。

- 送信データ書き込み : SITDR (32ビットアクセス)
- 受信データ読み出し : SIRDR (32ビットアクセス)

図 15.5 に送受信データと SITDR、SIRDR のビットアライメントを示します。

15. FIFO 付きシリアルインタフェース (SIOF)

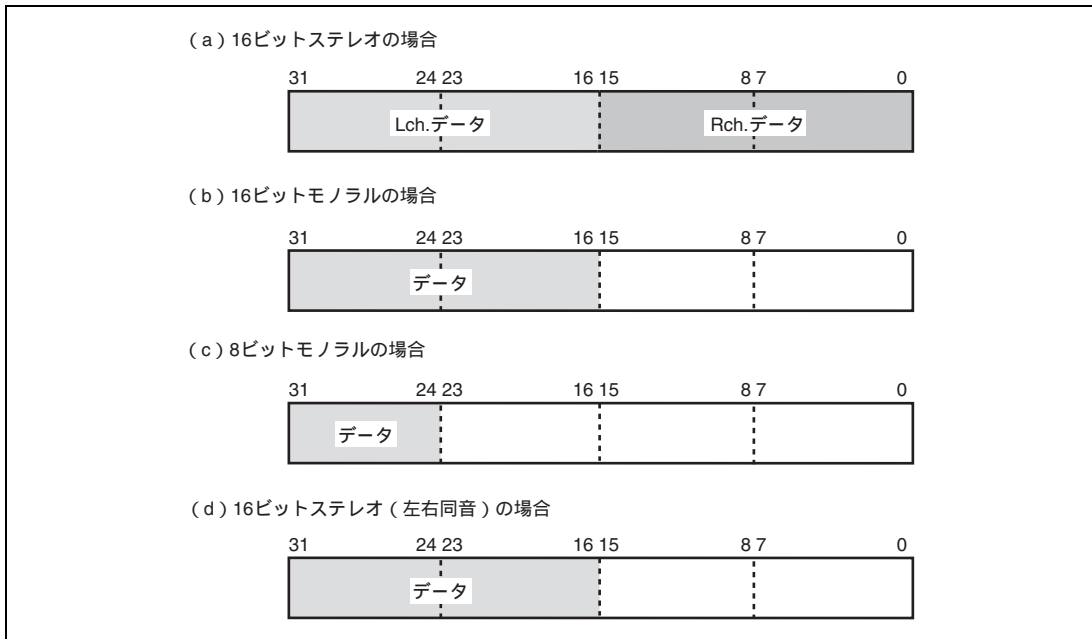


図 15.5 送受信データビットアライメント

【注】 ハッチングの部分だけが有効なデータとして送受信されます。このため、8ビットデータ時にはバイト転送、16ビットデータ時にはワード転送でアクセスを行う必要があります。ハッチングがかかっていない領域のデータは送受信の対象となりません。

送信データのモノラル/ステレオの設定は、SITDARのTDLEビットおよびTDREビットに対して行います。受信データのモノラル/ステレオの設定は、SIRDARのRDLEビットおよびRDREビットに対して行います。また、送信データステレオ時の左右同音出力は、SITDARのTLREPビットに設定を行います。表 15.6 に送信データにおける音声モードの設定を、表 15.7 に受信データにおける音声モードの設定を示します。

表 15.6 送信データ音声モード

モード \ ビット	TDLE	TDRE	TDREP
モノラル	1	0	x
ステレオ	1	1	0
左右同音	1	1	1

【注】 x : Don't care

表 15.7 受信データ音声モード

モード \ ビット	RDLE	RDRE
モノラル	1	0
ステレオ	1	1

【注】 受信データには、左右同音モードは存在しません。

8ビットモノラルで送受信を行う場合には、左チャンネル側を使用してください。

(2) 制御データ

制御データの書き込み / 読み出しは下記レジスタに対して行います。

- 送信制御データ書き込み : SITCR (32ビットアクセス)
- 受信制御データ読み出し : SIRCR (32ビットアクセス)

図 15.6 に送受信制御データと SITCR、SIRCR のビットアライメントを示します。

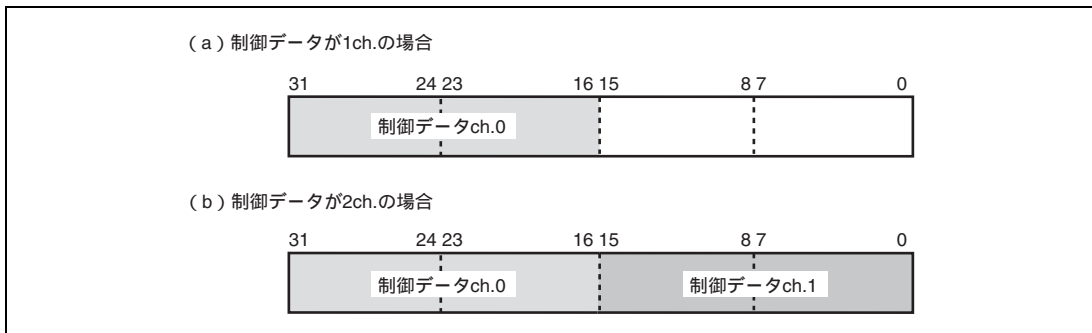


図 15.6 制御データビットアライメント

制御データの ch.数の設定は、SICDAR の CD0E ビットおよび CD1E ビットに対して行います。

表 15.8 に制御データにおける ch.数の設定を示します。

表 15.8 制御データ ch.数設定

ch.数 \ ビット	CD0E	CD1E
1	1	0
2	1	1

【注】 制御データを1チャンネルだけ用いる場合には、チャンネル0側を使用してください。

15. FIFO 付きシリアルインタフェース (SIOF)

15.4.5 制御データインタフェース

制御データは、CODEC への制御コマンド出力と CODEC の状態入力を行います。SIOF は、制御データのインタフェース方式として、下記 2 方式に対応します。

- スロット位置による制御
- セカンダリFSによる制御

制御データは、データ長として 16 ビットを選択した場合に有効となります。

(1) スロット位置による制御 (マスタモード 1、スレープモード 1)

制御データのスロット位置を指定して、SIOF が送受信を行う全フレームで制御データの送受信を行う方式です。SIOF マスタ、スレープのどちらでも使用可能です。図 15.7 にスロット位置による制御データインタフェースのタイミング例を示します。

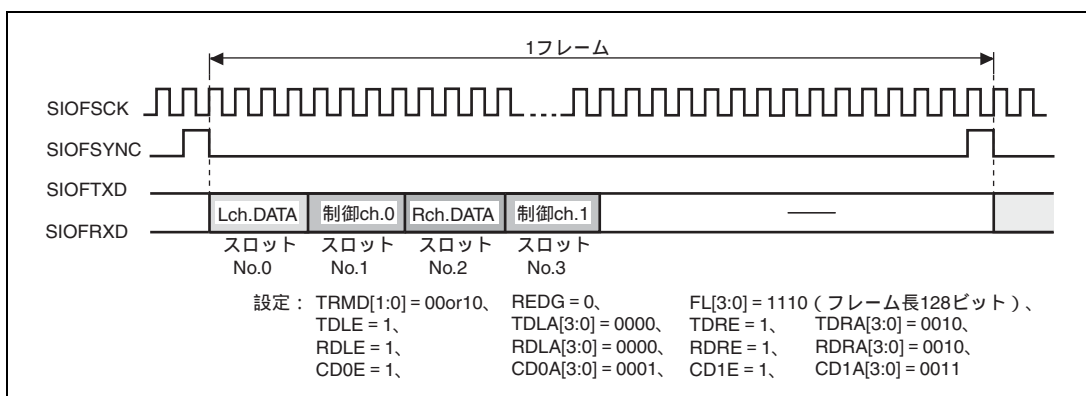


図 15.7 制御データインタフェース(スロット位置)

(2) セカンダリ FS による制御 (スレープモード 2)

SIOFSYNC をフレーム同期 (FS) として出力する CODEC が、本来の FS 出力位置でない 1/2 フレーム時間経過後に、制御データ送受信専用のセカンダリ FS を出力して、制御データの送受信を行う方式です。SIOF スレープのみ使用可能です。セカンダリ FS による制御データインタフェースの手順を以下に示します。

- 通常の送信データは LSB = 0 で送出 (SIOF が強制的に 0 とする)
制御データ送信を行う場合には LSB = 1 の送信データ送出 (SITCDR への書き込みで SIOF が 1 とする)
- CODEC はセカンダリ FS を出力する
- SIOF はセカンダリ FS に同期して制御データ (SITCDR 設定データ) の送信および受信 (SIRCDR へ格納) を行う

図 15.8 にセカンダリ FS による制御データインタフェースのタイミング例を示します。

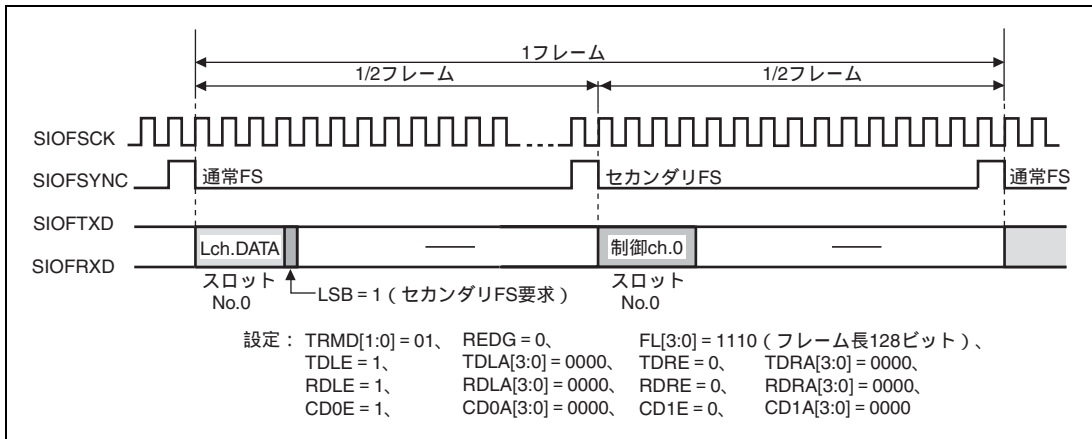


図 15.8 制御データインタフェース (セカンダリ FS)

15.4.6 FIFO

(1) 概要

SIOF 送受信 FIFO の特徴を以下に示します。

- 送受信それぞれ32ビット×16段の容量
- CPU、DMACのアクセスサイズにかかわらず、1回のリード/ライトサイクルでポインタは更新されます。
(1段のアクセスを複数回に分割することは不可能です。)

(2) 転送要求

FIFO の転送要求は、以下に記す割り込み要因として CPU および DMAC に出力することが可能です。

- 送信要求：TDREQ (送信用割り込み要因)
- 受信要求：RDREQ (受信用割り込み要因)

送受信 FIFO の転送要求を発行する条件は、それぞれ個別に設定が可能です。送信 FIFO の転送要求の条件は SIFCTR の TFWM2 ~ TFWM0 ビットに、受信 FIFO の転送要求は RFWM2 ~ RFWM0 ビットに設定します。表 15.9 に送信要求発行条件を、表 15.10 に受信要求発行条件を示します。

15. FIFO 付きシリアルインタフェース (SIOF)

表 15.9 送信要求発行条件

TFWM2 ~ 0	リクエスト段数	送信要求発行	使用領域
000	1	空き領域が 16 段	小 ▲ ▼ 大
100	4	空き領域が 12 段以上	
101	8	空き領域が 8 段以上	
110	12	空き領域が 4 段以上	
111	16	空き領域が 1 段以上	

表 15.10 受信要求発行条件

RFWM2 ~ 0	リクエスト段数	受信要求発行	使用領域
000	1	有効データが 1 段以上	小 ▲ ▼ 大
100	4	有効データが 4 段以上	
101	8	有効データが 8 段以上	
110	12	有効データが 12 段以上	
111	16	有効データが 16 段	

データ領域もしくは空き領域が上記段数を超えた場合でも、FIFO の容量は常に 16 段使用可能です。したがってオーバーフローもしくはアンダフローエラーは、データ領域もしくは空き領域が 16 段を超えた場合に発行されません。

また、転送要求は FIFO が空もしくは満杯にならなくても、上記条件を満たさなくなった時点で解除します。

(3) 段数表示

送受信 FIFO の使用状況を、下記内容でレジスタに表示します。

- 送信FIFO：SIFCTRのTFUA4～TFUA0ビットに空き領域の段数を表示
- 受信FIFO：SIFCTRのRFUA4～RFUA0ビットに有効データの段数を表示

上記内容は、CPU もしくは DMAC が転送可能なデータ数を表します。

15.4.7 送受信手順

(1) マスタ時送信

図 15.9 にマスタ時送信の設定例および動作を示します。

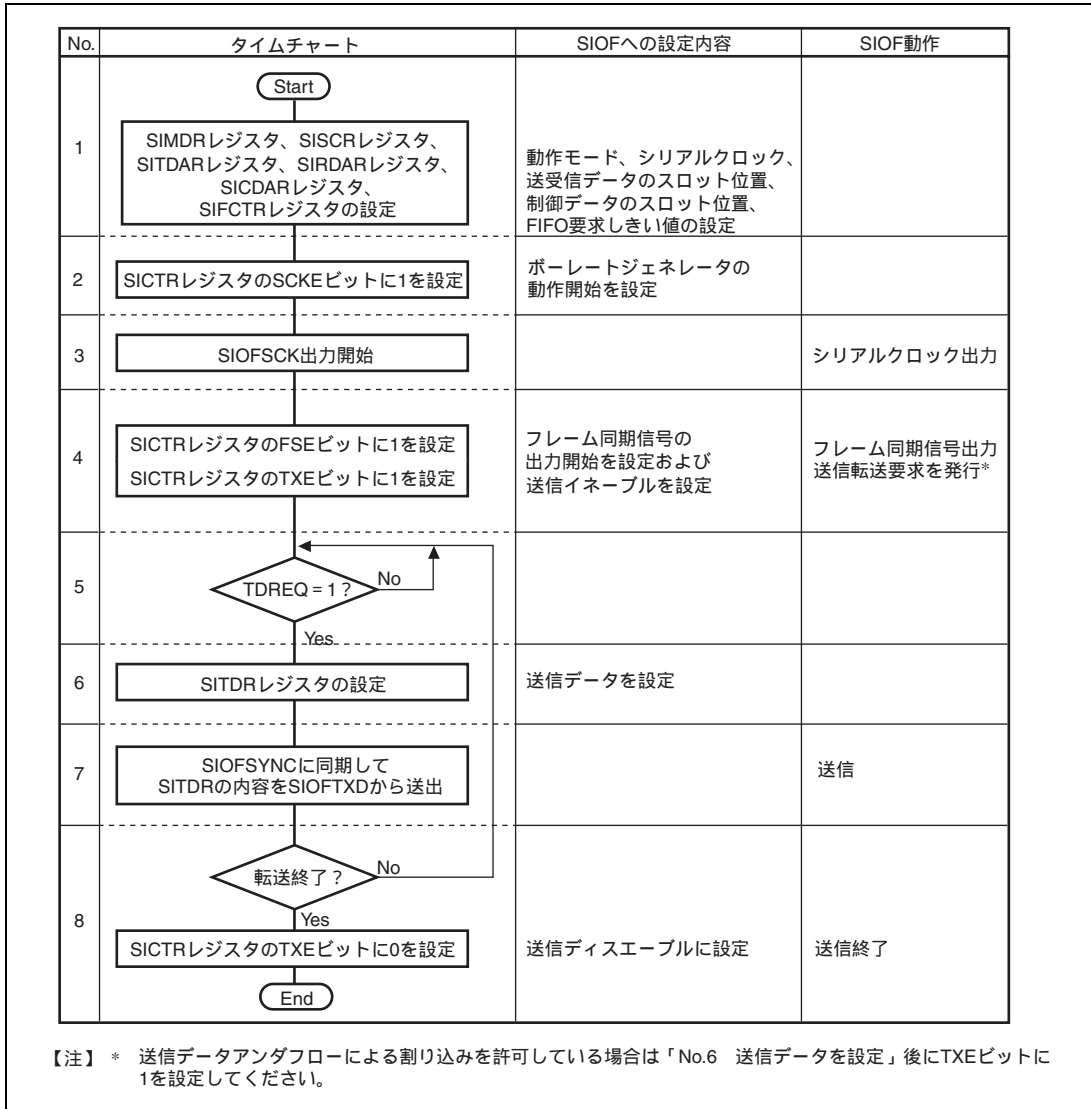


図 15.9 マスタ時送信動作例

15. FIFO 付きシリアルインタフェース (SIOF)

(2) マスタ時受信

図 15.10 にマスタ時受信の設定例および動作を示します。

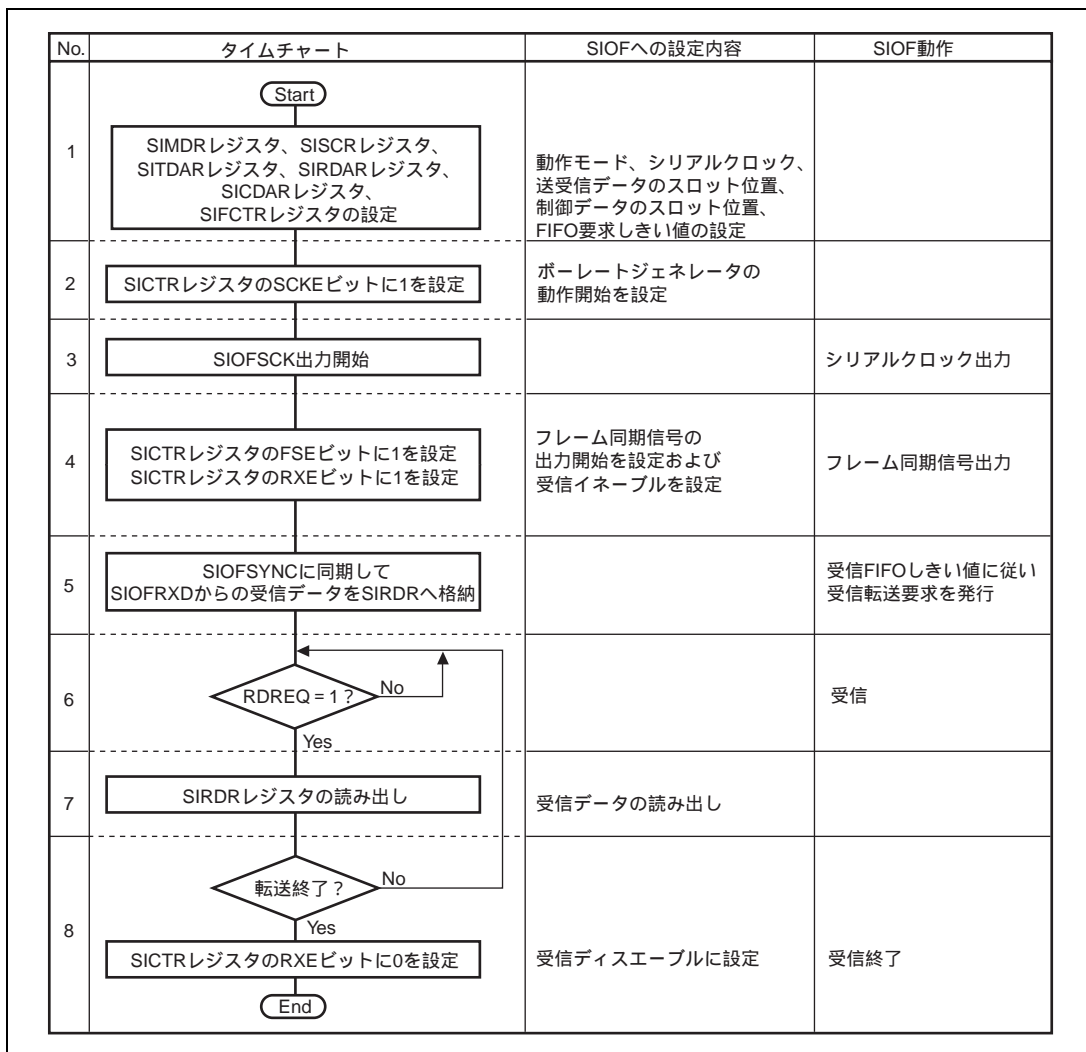


図 15.10 マスタ時受信動作例

(3) スレープ時送信

図 15.11 にスレープ時送信の設定例および動作を示します。

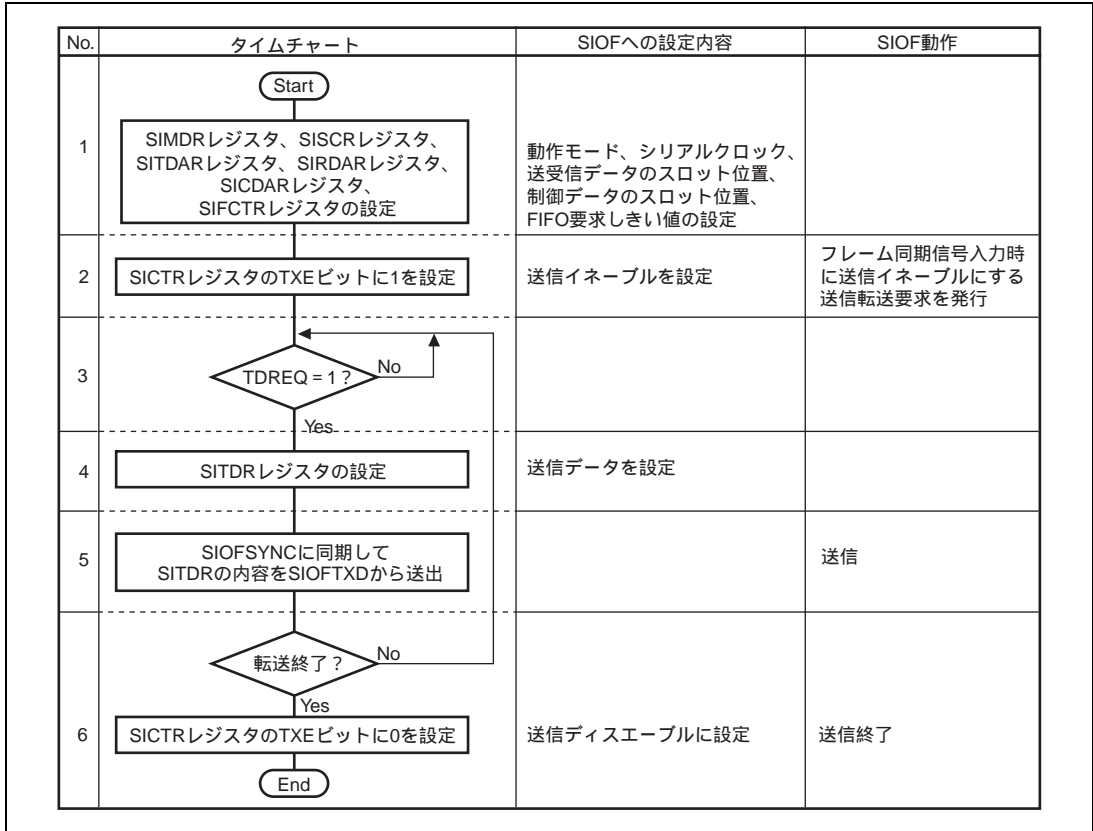


図 15.11 スレープ時送信動作例

15. FIFO 付きシリアルインタフェース (SIOF)

(4) スレープ時受信

図 15.12 にスレープ時受信の設定例および動作を示します。

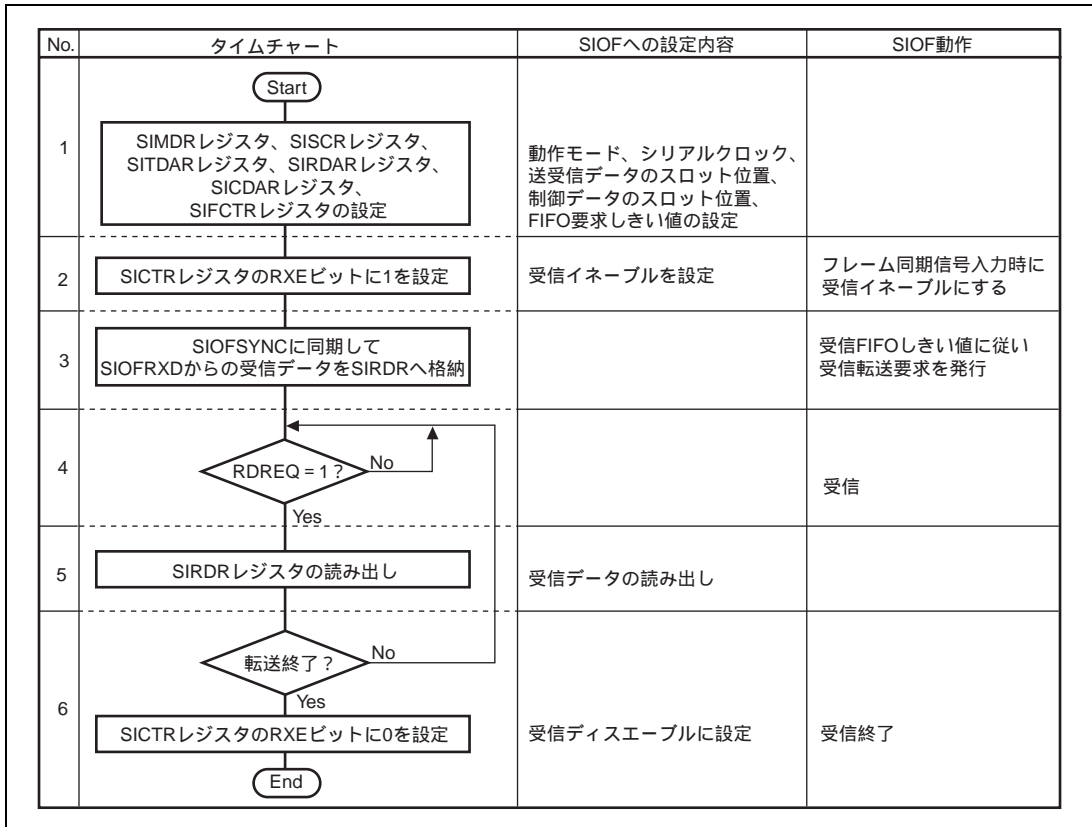


図 15.12 スレープ時受信動作例

(5) 送受信リセット

SIOF は、下記ビットに 1 を設定することにより、送信部と受信部を個別にリセットすることが可能です。

- 送信リセット：(SICTRのTXRSTビット)
- 受信リセット：(SICTRのRXRSTビット)

表 15.11 に送受信リセットで初期化される内容を示します。

表 15.11 送受信リセット

種類	初期化対象
送信リセット	SITDR 送信 FIFO ライトポインタ SISTR の TCRDY ビット、TFEMP ビット、TDREQ ビット SICTR の TXE ビット
受信リセット	SIRDR 受信 FIFO リードポインタ SISTR の RCRDY ビット、RFFUL ビット、RDREQ ビット SICTR の RXE ビット

(6) モジュールストップ

SIOF は、モジュールストップ時には送受信動作を停止します。またその際、下記内容を初期化します。

- SITDR
- SITCR
- 受信FIFOリードポインタ
- 送信FIFOライトポインタ
- SISTR
- SICTR

15. FIFO 付きシリアルインタフェース (SIOF)

15.4.8 割り込み

SIOF は、1 種類の割り込みを持っています。

(1) 割り込み要因

割り込みは、それぞれ複数の要因によって発行することができます。各要因は、SISTR に SIOF ステータスとして表示します。表 15.12 に SIOF 割り込み要因一覧を示します。

表 15.12 SIOF 割り込み要因

No.	区分	ビット名	機能名称	内 容
1	送信	TDREQ	送信 FIFO 転送要求	送信 FIFO に設定値以上のデータが格納された
2		TFEMP	送信 FIFO エンプティ	送信 FIFO が空である
3	受信	RDREQ	受信 FIFO 転送要求	受信 FIFO に設定値以上のデータが格納された
4		RFFUL	受信 FIFO フル	受信 FIFO が満杯である
5	制御	TCRDY	送信制御データレディ	送信制御データレジスタに書き込みが可能である
6		RCRDY	受信制御データレディ	受信制御データレジスタに有効な値が格納された
7	エラー	TFUDF	送信 FIFO アンダフロー	送信 FIFO が空時にシリアルデータ送出タイミングが来た
8		TFOVF	送信 FIFO オーバフロー	送信 FIFO が満杯時に送信 FIFO に書き込みを行った
9		RFOVF	受信 FIFO オーバフロー	受信 FIFO が満杯時にシリアルデータを受信した
10		RFUDF	受信 FIFO アンダフロー	受信 FIFO が空時に受信 FIFO 読み出しを行った
11		FSERR	フレーム同期エラー	設定ビット数以前に同期信号が入力された (スレーブ時)
12		SAERR	スロットアサインエラー	シリアルデータ、制御データのスロット設定が重複している

割り込み要因によって割り込みを発行するか否かは、SIER への設定によって決定します。SIER の対応するビットに 1 が設定してある割り込み要因が 1 にセットされたときに、SIOF 割り込みを発行します。

(2) 送受信区分について

送信区分の要因、受信区分の要因は状態を表す信号であり、いったんセットされてもその後状態が変化すると SIOF が自動的にクリアします。

ただし、DMA 転送を用いた場合には、DMA 転送が終了した時点で、DMA 転送要求は必ず 1 サイクル期間 0 レベルとなります。

(3) エラー発生時の処理

SISTR にステータスとして表示する各エラー発生時、SIOF は下記動作を行います。

- 送信FIFOアンダフロー (TFUDF)

直前の送信データを再び送出します。

- 送信FIFOオーバフロー (TFOVF)

送信FIFOの内容は保護され、オーバフローとなった書き込みは無視します。

- 受信FIFOオーバーフロー (RFOVF)
オーバーフローとなったデータが廃棄され、消失します。
 - 受信FIFOアンダフロー (RFUDF)
不定値がバス上に出力されます。
 - フレーム同期エラー (FSERR)
エラーとなった同期信号 (FSYN) に従って、内部カウンタはリセットされます。
 - スロットアサインエラー (SAERR)
1. シリアルデータと制御データが重複した場合には、シリアルデータが優先されます。
 2. 制御データ同士が重複した場合には、送受信データは保証されません。

15.4.9 送受信タイミング

SIOF のシリアル送受信の例を図 15.13 ~ 図 15.20 に示します。

(1) 8 ビットモノラルの場合 (その 1)

同期パルス方式、立ち下がりサンプリング、送信データ、受信データはスロット No.0、フレーム長は 8 ビット

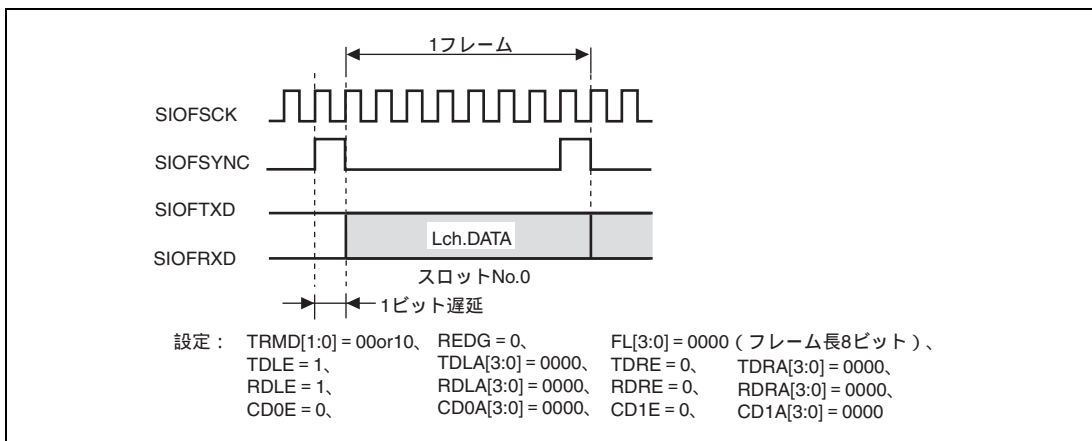


図 15.13 送受信タイミング (8 ビットモノラル - 1)

15. FIFO 付きシリアルインタフェース (SIOF)

(2) 8ビットモノラルの場合 (その2)

同期パルス方式、立ち下がりサンプリング、送信データ、受信データはスロット No.0、フレーム長は 16 ビット

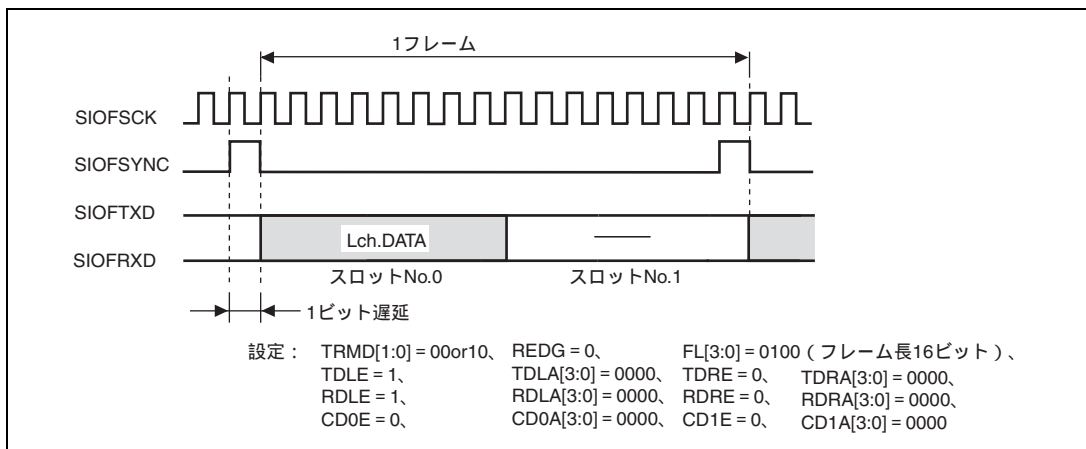


図 15.14 送受信タイミング (8ビットモノラル - 2)

(3) 16ビットモノラルの場合 (その1)

同期パルス方式、立ち下がりサンプリング、送信データ、受信データはスロット No.0、フレーム長は 64 ビット

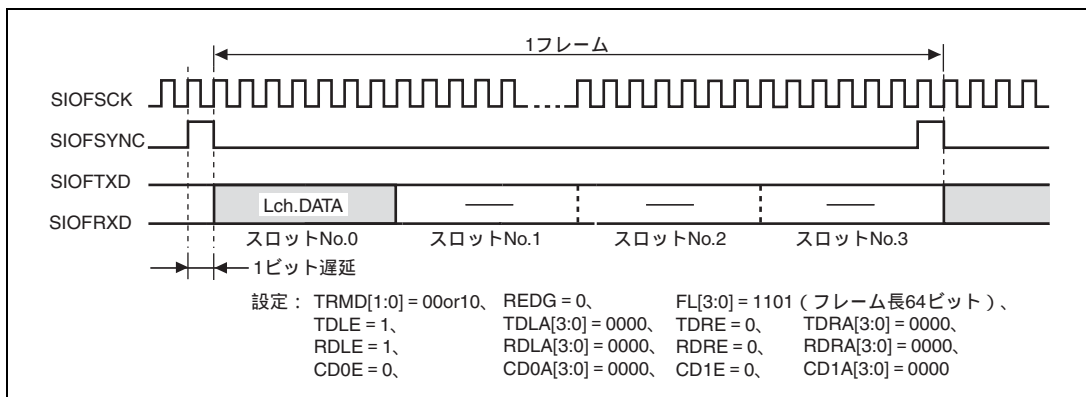


図 15.15 送受信タイミング (16ビットモノラル - 1)

(4) 16 ビットステレオの場合 (その 1)

L/R 方式、立ち上がりサンプリング、Lch.データはスロット No.0、Rch.データはスロット No.1、フレーム長は 32 ビット

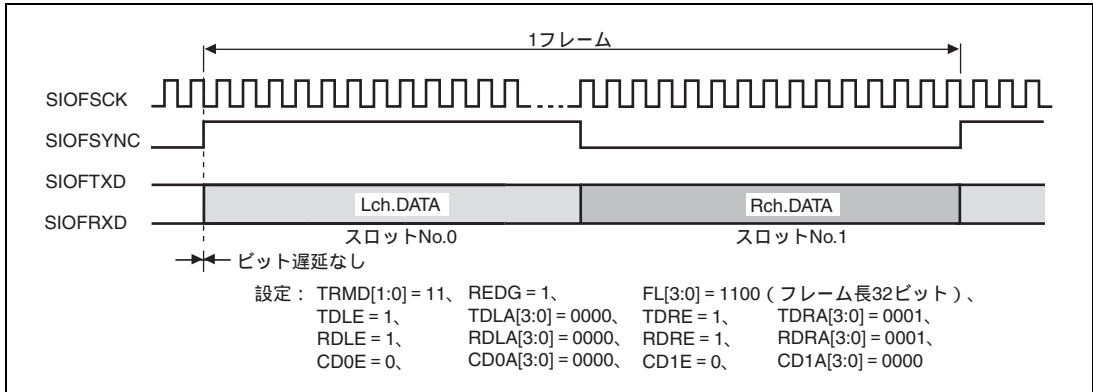


図 15.16 送受信タイミング (16 ビットステレオ - 1)

(5) 16 ビットステレオの場合 (その 2)

L/R 方式、立ち上がりサンプリング、Lch.送信データはスロット No.0、Lch.受信データはスロット No.1、Rch.送信データはスロット No.2、Rch.受信データはスロット No.3、フレーム長は 64 ビット

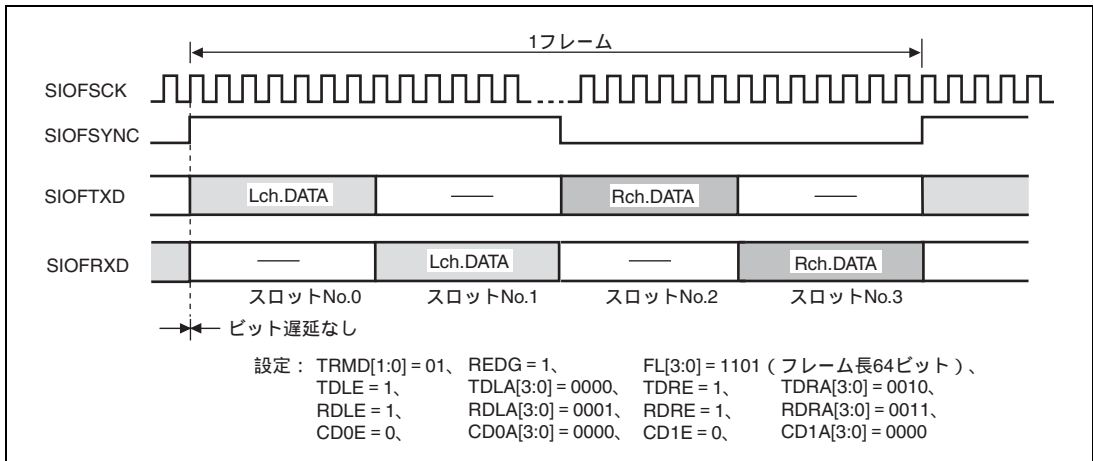


図 15.17 送受信タイミング (16 ビットステレオ - 2)

15. FIFO 付きシリアルインタフェース (SIOF)

(6) 16 ビットステレオの場合 (その3)

同期パルス方式、立ち下がりサンプリング、Lch.データはスロット No.0、Rch.データはスロット No.1、制御 ch.データ 0 はスロット No.2、制御 ch.データ 1 はスロット No.3、フレーム長は 128 ビット

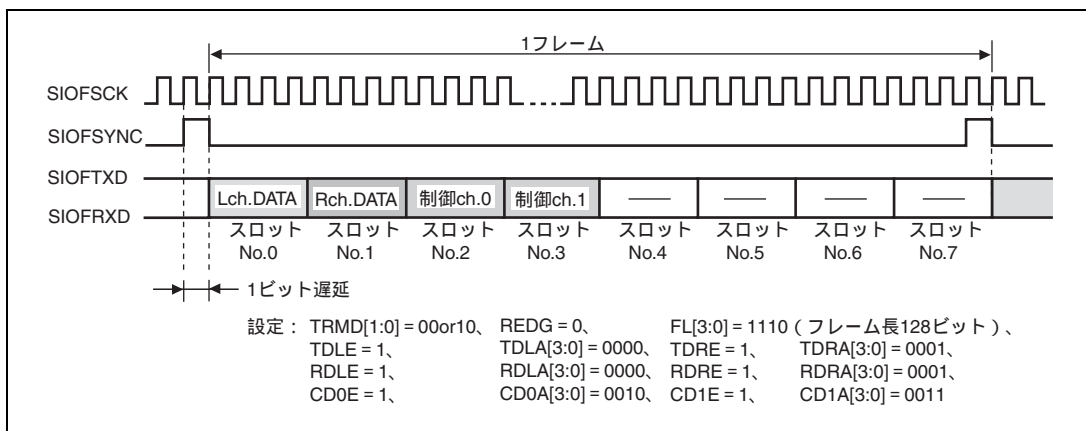


図 15.18 送受信タイミング (16 ビットステレオ - 3)

(7) 16 ビットステレオの場合 (その4)

同期パルス方式、立ち下がりサンプリング、Lch.データはスロット No.0、Rch.データはスロット No.2、制御 ch.データ 0 はスロット No.1、制御 ch.データ 1 はスロット No.3、フレーム長は 128 ビット

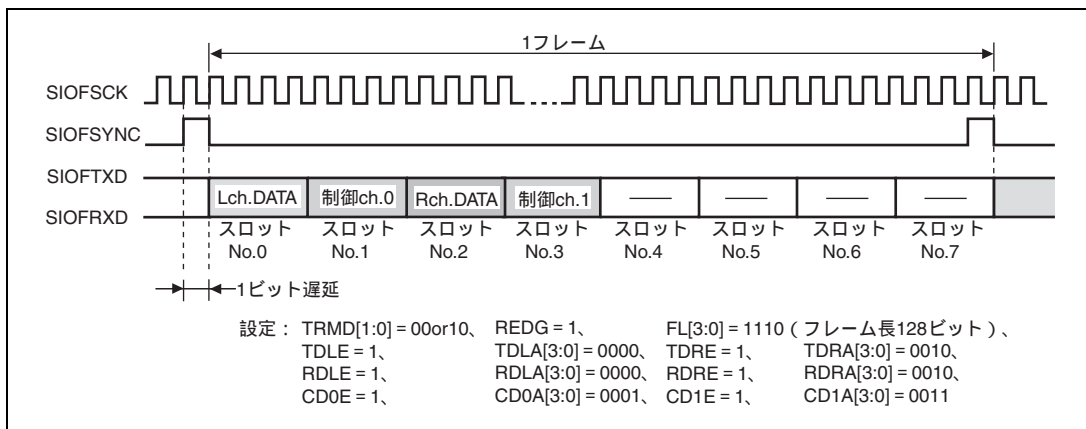


図 15.19 送受信タイミング (16 ビットステレオ - 4)

(8) 各スロット終了時に同期パルスを出力するモードの場合 (SYNCAT ビットが 1 のとき)

同期パルス方式、立ち下がりサンプリング、Lch.データはスロット No.0、Rch.データはスロット No.1、制御 ch.データ 0 はスロット No.2、制御 ch.データ 1 はスロット No.3、フレーム長は 128 ビット

また、本モードでは、スロット No.0 に必ず有効データを設定してください。

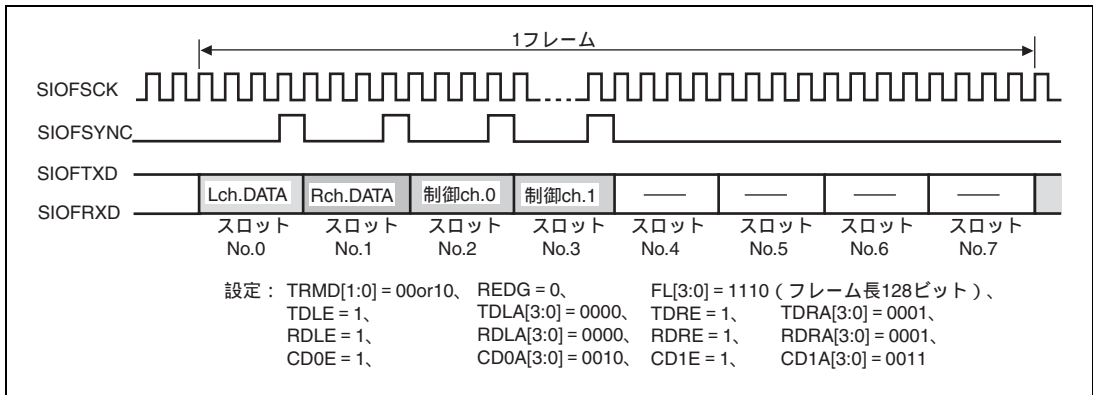


図 15.20 送受信タイミング (16 ビットステレオ)

15.4.10 SPI モード

SPICR の設定により、SIOF を SPI モードで動作させることができます。

(1) 構成例

図 15.21 に SPI モード時の構成例を示します。

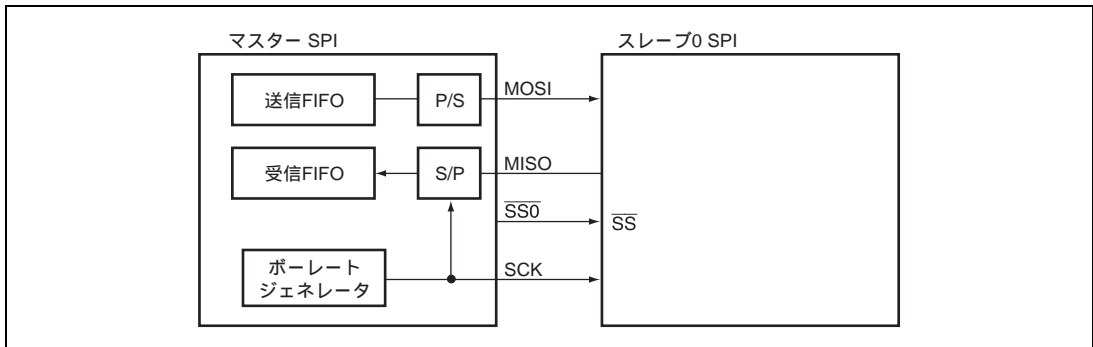


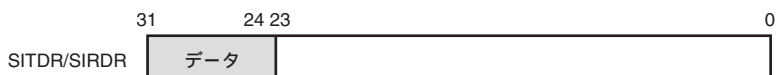
図 15.21 SPI モードの構成例

15. FIFO 付きシリアルインタフェース (SIOF)

(2) SPI 動作

表 15.13 に SPI 送受信の動作状態を示します。

SPI モード時の送受信するデータ長は 8 ビット固定で、SITDR、SIRDR の上位 8 ビットが有効なデータとして、送受信されます。



ハッチングの部分だけが有効なデータとして送受信されます。

割り込み要求の要因は、送信データ転送要求 (TDREQ)、送信 FIFO エンプティ (TFEMP)、受信データ転送要求 (RDREQ)、受信 FIFO フル (RFFUL)、受信 FIFO オーバフロー (RFOVF) のみを有効とし、割り込み許可レジスタ (SIER) で禁止 / 許可を選択します。これ以外の割り込み要因は常に禁止状態にしてください。

DMA 転送要求は、送信データ DMA 転送要求 (TDMA)、受信データ DMA 転送要求 (RDMA) を有効とし、割り込み許可レジスタで禁止 / 許可を選択します。

SPI のボーレートは、SISCR を使用して設定します。

表 15.13 SPI 送受信の動作状態

TXE	RXE	TDMAE	RDMAE	SPI 送受信動作
0	0	Don't Care	Don't Care	送受信不可
0	1	0	1	半二重受信 送信 FIFO が動作せず、MOSI はダミーデータを送信します。 MISO の受信データは、受信 FIFO に格納され、DMA 転送されます。 RE = 1 の間、受信動作を継続し、受信 FIFO がフルになると以降の受信データを無視し、受信 FIFO オーバフロー (RFOVF) のステータスがセットされます。
1	0	0	0	半二重送信 送信 FIFO のデータが MOSI から送信されます。 受信 FIFO が動作せず、MISO データを無視します。 送信 FIFO が空になると送信動作を終了します。
		1	0	半二重送信 送信 FIFO に DMA 転送されたデータは MOSI から送信されます。受信 FIFO が動作せず、MISO データを無視します。 送信 FIFO が空になると送信動作を終了します。
1	1	0	0	全二重通信 送受信 FIFO が同時に動作します。 送信 FIFO にデータが存在すると送受信動作を行います。 送信 FIFO が空になると送受信動作を終了します。

【注】 SPI モードでは、上記以外の設定を禁止します。

半二重受信時 (送信無効時) の場合には、MOSI 出力の値を SIMDR の TXDIZ ビットで制御することができます。

TXDIZ = 0 : 送信無効時、MOSI は 1 出力

TXDIZ = 1 : 送信無効時、MOSI はハイインピーダンス状態

(3) シリアルクロックのタイミング

図 15.22、図 15.23 に SPI 動作時のデータ/クロックのタイミング図を示します。図に示すように、シリアルクロックのフェーズと極性により 4 通りのシリアル転送フォーマットを選択できます。

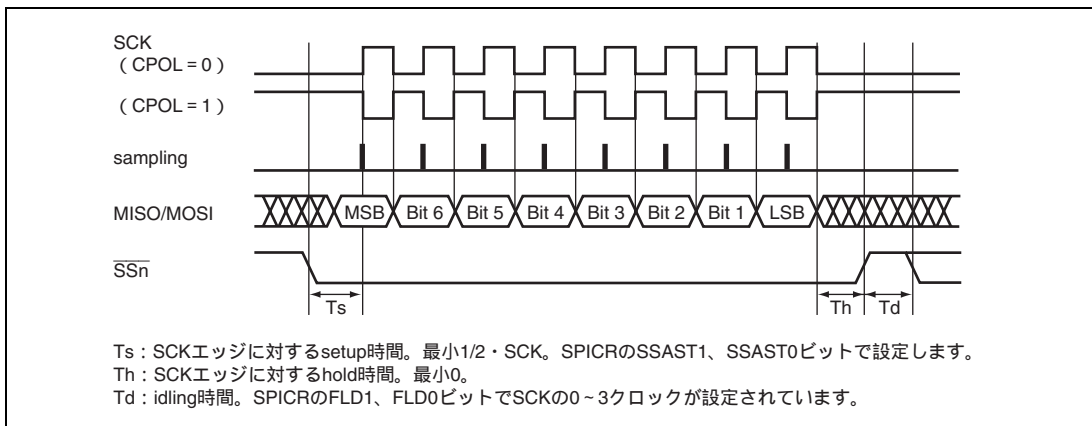


図 15.22 SPI データ/クロックのタイミング 1 (CPHA = 0)

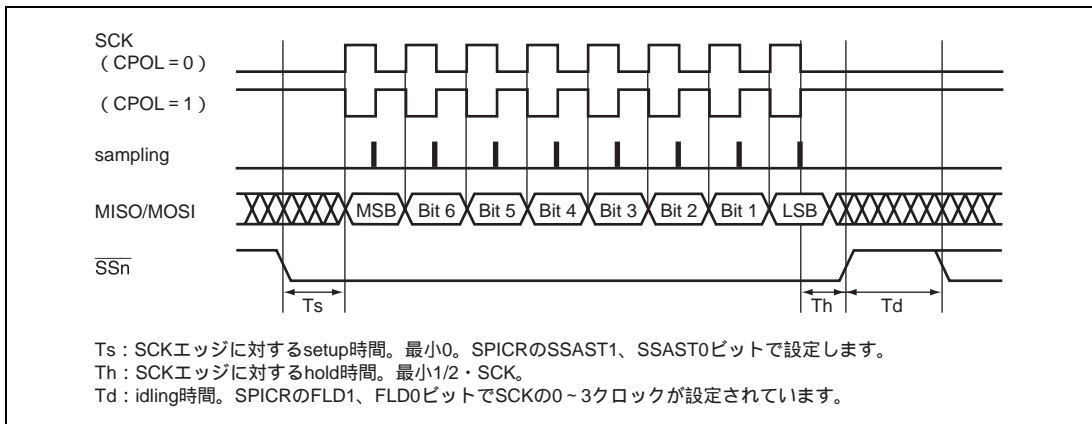


図 15.23 SPI データ/クロックのタイミング 2 (CPHA = 1)

15.5 使用上の注意事項

1. 送受信動作中にソフトウェアスタンバイモードに入ったり、SIOFのモジュールスタンバイビットをセットしてモジュールスタンバイ状態にしないでください。ソフトウェアスタンバイモードに移行するときや、モジュールスタンバイ状態にするときは、必ずSICTRレジスタのTXE、RXEビットが共にクリア（送受信ディスエーブル状態）した後に移行してください。
2. SIOFのSIOF_TXD端子は、TXEビットがクリアされていても出力状態になります。SIOF_RXD端子は、RXEビットがクリアされていても入力イネーブル状態になります。ソフトウェアスタンバイモード時、SIOF_TXD端子はハイインピーダンス状態で、SIOF_RXD端子は入力固定状態になります。ただし、SIOF_TXD端子は、PTA1ポートとピンマルチプレクスされており、パワーオンリセット直後の初期状態は、入力がかつブルアップMOSがオン状態です。SIOF_RXD端子はPTA2ポートとピンマルチプレクスされており、パワーオンリセット直後の初期状態は、入力がかつブルアップMOSがオン状態です。SIOF_TXD、SIOF_RXD端子に接続する信号線の処理には注意が必要です。
3. SIOFには、16ビットの独立した2つのデータを取り扱う32ビットのレジスタがいくつか存在します。MD5端子により、ビッグエンディアン/リトルエンディアンを切り替える場合、これらのレジスタとのデータ転送ではメモリ上での上位ワードと下位ワードの順序がレジスタ内では入れ替わる場合がありますので、注意が必要です。
4. SIOFがDMA転送の転送先あるいは転送元になる場合、一回の転送単位はロングワードのみとなります。
5. SIHERのTDMAEビットを"1"にして、送信データDMA転送要求を有効にしたときに、SIFCTRのTFWMビットを3'b111（送信FIFOの空き容量が1段以上のときに転送要求を発行する）の設定にした場合、DMA転送によりFIFOがFULLになったにも関わらず、誤って次のDMA転送要求を出してしまい、送信FIFOがオーバーフローしてしまう場合があります。このため、送信データDMA転送要求使用時には、TFWMビットは、3'b000～3'b110までの範囲で設定してください。
6. マスタモード2（ステレオ出力）で使用した場合、SICTRレジスタのFSEビットに"0"を設定しSIOFSYNCの出力を許可しない状態から、FSEビットに"1"を設定しSIOFSYNCの出力を許可したとき、先頭フレームのSIOFSYNC信号の立ち上がりが1ビット早くなりHigh期間が長くなる場合（送信開始時または送信再開時）と1ビット遅くなりHigh期間が短くなる場合（送信再開時）があります。また、送信停止時のFSEビットを"0"とするタイミングが、SIOFSYNC信号のHigh期間の場合には、SIOFSYNC信号のHigh期間が短くなる場合があります。マスタモード2を使用する場合には、ステレオデータ転送の開始時と終了時に、無音データを挿入するか外部コーデックのミュート制御をすることをお勧めします。

16. FIFO 内蔵シリアルコミュニケーション インタフェース (SCIF)

本 LSI は、2 チャンネルの FIFO バッファ内蔵のシリアルコミュニケーションインタフェース (SCIF : Serial Communication Interface with FIFO) を内蔵しています。

SCIF は、調歩同期式とクロック同期式のシリアル通信ができます。

送受信に FIFO レジスタをおのおの 64 段内蔵しており、効率の良い高速連続通信を行うことができます。

16.1 特長

- 調歩同期式モード

キャラクタ単位で同期をとる調歩同期方式でシリアルデータの通信を行います。Universal Asynchronous Receiver/Transmitter (UART) や Asynchronous Communication Interface Adapter (ACIA) など標準の調歩同期式通信用 LSI とのシリアルデータ通信が可能です。

シリアルデータ通信フォーマットを 8 種類のフォーマットから選択できます。

データ長 : 7 ビット、または 8 ビット

ストップビット長 : 1 ビット、または 2 ビット

パリティ : 偶数パリティ、奇数パリティ、またはパリティなし

LSB ファースト

受信エラーの検出 : パリティエラー、フレーミングエラー、オーバランエラーを検出

ブレークの検出 : フレーミングエラーが発生し、引き続き 1 フレーム長以上スペース 0 (ローレベル) の場合、ブレークを検出します。

- クロック同期式モード

クロックに同期してシリアルデータ通信を行います。クロック同期式通信機能を持つ他の LSI とのシリアルデータ通信が可能です。

データ長 : 8 ビット

LSB ファースト

- 全二重通信が可能

独立した送信部と受信部を備えているので、送信と受信を同時に行うことができます。

また、送信部、および受信部ともに 64 段の FIFO バッファ構造になっていますのでシリアルデータの高速連続送信、連続受信ができます。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

- 内蔵ボーレートジェネレータにより任意のビットレートを選択可能
- 送受信クロックソースを、ボーレートジェネレータからの内部クロック、またはSCK端子からの外部クロックから選択可能。
- 調歩同期式モード時は6種類の割り込み要因
送信データストップ割り込み、送信FIFOデータエンプティ、受信FIFOデータフル、受信エラー（フレーミングエラー/パリティエラー）割り込み、ブレーク受信割り込み、レシーブデータレディ割り込みの6種類の割り込み要因があります。各割り込み要因による割り込みのベクタは共通です。
- クロック同期式モード時は2種類の割り込み要因
送信FIFOデータエンプティと受信FIFOデータフルの2種類の割り込み要因があります。各割り込み要因による割り込みのベクタは共通です。
- 送信FIFOデータエンプティ時と送信データストップ時、受信FIFOデータフル時にDMAコントローラ（DMAC）を起動させてデータの転送を行うことができます。ただし、送信FIFOデータエンプティと送信データストップのDMAC要求は共通です。
- モデムコントロール機能（ $\overline{\text{CTS}}$ 、 $\overline{\text{RTS}}$ ）を内蔵しています。
- 送信データストップ機能（調歩同期式モードのみ）を内蔵しています。
- SCIFを使用しないときは、消費電力低減のためSCIFに対してクロックの供給を止めて動作を停止させることができます。
- 送受信FIFOレジスタ内のデータ数および受信FIFOレジスタ内の受信データの受信エラー数を知ることができます。

図 16.1 に SCIF のブロック図を示します。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

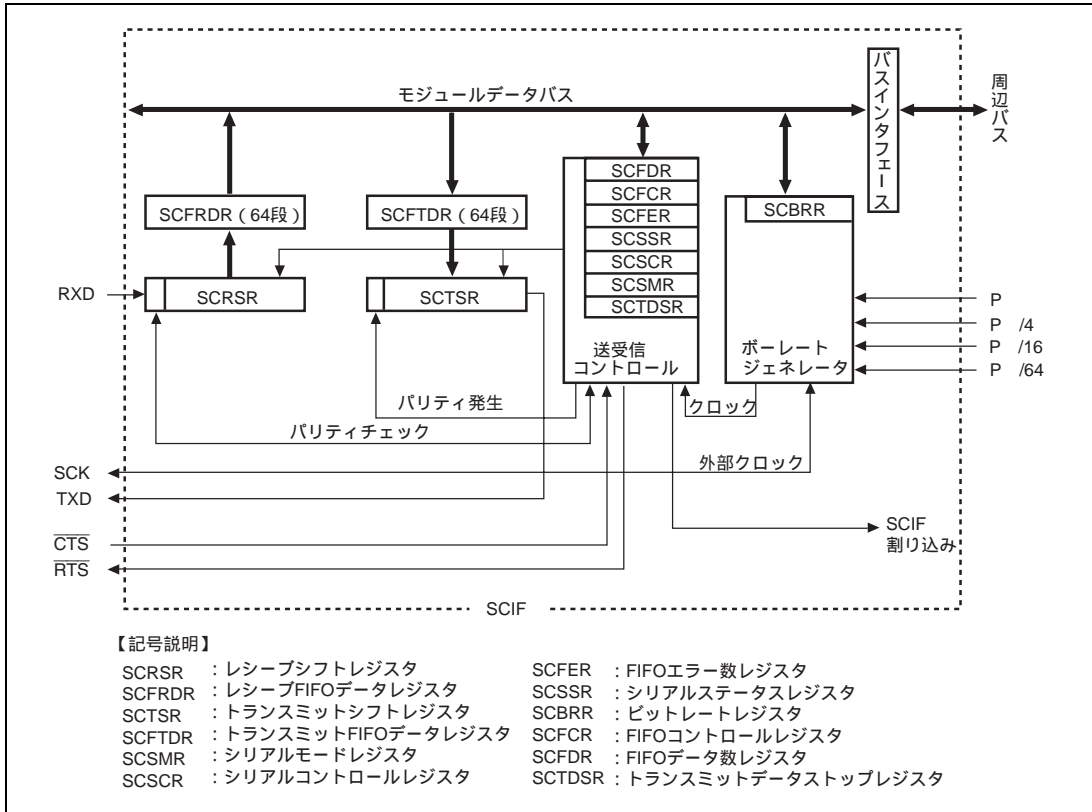


図 16.1 SCIF のブロック図

16.2 入出力端子

SCIF の端子構成を表 16.1 に示します。

表 16.1 端子構成

チャネル	端子名称	略称*1	入出力	機能
0	SCIF0_SCK	SCK	入出力	シリアルクロック端子 クロック入出力
	SCIF0_RXD	RXD*2	入力	レシーブデータ端子 受信データ入力
	SCIF0_TXD	TXD*2	出力	トランスミットデータ端子 送信データ出力
	SCIF0_CTS	CTS	入力	モデムコントロール端子 送信可
	SCIF0_RTS	RTS	出力	モデムコントロール端子 送信要求

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

チャンネル	端子名称	略称* ¹	入出力	機 能
1	SCIF1_SCK	SCK	入出力	シリアルクロック端子 クロック入出力
	SCIF1_RXD	RXD* ²	入力	レシーブデータ端子 受信データ入力
	SCIF1_TXD	TXD* ²	出力	トランスミットデータ端子 送信データ出力
	SCIF1_CTS	CTS	入力	モデムコントロール端子 送信可
	SCIF1_RTS	RTS	出力	モデムコントロール端子 送信要求

【注】 *1 本文中では、チャンネルを省略し、総合の名称として SCK、RXD、TXD、CTS、RTS を使用します。

*2 SCIF の動作設定を SCSCR の TE、RE ビットで行うことにより、シリアル端子として機能します。

16.3 レジスタの説明

本モジュールには以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

(1) チャンネル 0

- シリアルモードレジスタ_0 (SCSMR_0)
- ビットレートレジスタ_0 (SCBRR_0)
- シリアルコントロールレジスタ_0 (SCSCR_0)
- トランスミットデータストップレジスタ_0 (SCTDSR_0)
- FIFOエラー数レジスタ_0 (SCFER_0)
- シリアルステータスレジスタ_0 (SCSSR_0)
- FIFOコントロールレジスタ_0 (SCFCR_0)
- FIFOデータ数レジスタ_0 (SCFDR_0)
- トランスミットFIFOデータレジスタ_0 (SCFTDR_0)
- レシーブFIFOデータレジスタ_0 (SCFRDR_0)

(2) チャンネル 1

- シリアルモードレジスタ_1 (SCSMR_1)
- ビットレートレジスタ_1 (SCBRR_1)
- シリアルコントロールレジスタ_1 (SCSCR_1)
- トランスミットデータストップレジスタ_1 (SCTDSR_1)
- FIFOエラー数レジスタ_1 (SCFER_1)

- シリアルステータスレジスタ_1 (SCSSR_1)
- FIFOコントロールレジスタ_1 (SCFCR_1)
- FIFOデータ数レジスタ_1 (SCFDR_1)
- トランスミットFIFOデータレジスタ_1 (SCFTDR_1)
- レシーブFIFOデータレジスタ_1 (SCFRDR_1)

16.3.1 レシーブシフトレジスタ (SCRSR)

SCRSR は、シリアルデータを受信するためのレジスタです。

SCIF は、SCRSR に RXD 端子から入力されたシリアルデータを LSB (ビット 0) から受信した順にセットし、パラレルデータに変換します。1 バイトのデータ受信を終了すると、データは自動的にレシーブ FIFO データレジスタである SCFRDR へ転送されます。

CPU から直接 SCRSR の読み出し / 書き込みをすることはできません。

16.3.2 レシーブ FIFO データレジスタ (SCFRDR)

SCFRDR は、受信したシリアルデータを格納する 8 ビット長の 64 段 FIFO レジスタ (略称: 受信 FIFO) です。

SCIF は、1 バイトのシリアルデータの受信が終了すると、レシーブシフトレジスタ (SCRSR) から SCFRDR へ受信したシリアルデータを転送して格納し、受信動作を完了します。このあと、SCRSR は受信可能になり、レシーブ FIFO データレジスタがいっぱいになる 64 バイトデータまで連続した受信動作が可能です。

SCFRDR は、読み出し専用レジスタですので CPU から書き込むことはできません。

また、レシーブ FIFO データレジスタに受信データがない状態で読み出した値は不定値になります。レシーブ FIFO データレジスタ内の受信データがいっぱいになると、以降のシリアルデータは失われます。

ビット	ビット名	初期値	R/W	説明
7~0	SCFRD7~0	不定	R	シリアル受信データ用 FIFO

16.3.3 トランスミットシフトレジスタ (SCTSR)

SCTSR は、シリアルデータを送信するためのレジスタです。

SCIF は、トランスミット FIFO データレジスタ (SCFTDR) から送信データをいったん SCTSR に転送し、LSB (ビット 0) から順に TXD 端子に送り出すことでシリアルデータ送信を行います。

1 バイトのデータ送信を終了すると自動的に SCFTDR から SCTSR へ次の送信データを転送し、送信を開始します。

CPU から、直接 SCTSR の読み出し / 書き込みをすることはできません。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

16.3.4 トランスミット FIFO データレジスタ (SCFTDR)

SCFTDR は、シリアル送信するデータを格納する 8 ビット長の 64 段 FIFO レジスタ (略称: 送信 FIFO) です。SCIF は、送信データが SCFTDR に書き込まれたとき、トランスミットシフトレジスタ (SCTSR) が空ならば、SCFTDR に書き込まれた送信データを SCTSR に転送してシリアル送信を開始します。

SCFTDR は、書き込み専用レジスタですので CPU から読み出すことはできません。

SCFTDR 内の送信データが 64 バイトでいっぱいになるときは次のデータを書き込むことはできません。書き込んだデータは無視されます。

ビット	ビット名	初期値	R/W	説明
7~0	SCFTD7~0	不定	W	シリアル送信データ用 FIFO

16.3.5 シリアルモードレジスタ (SCSMR)

SCSMR は、読み出し / 書き込み可能な 16 ビットのレジスタで、SCIF のシリアル通信フォーマットの設定と、ポーレートジェネレータのクロックソース、サンプリングレートを選択します。

ビット	ビット名	初期値	R/W	説明
15~11	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
10	SRC2	0	R/W	サンプリングコントロール
9	SRC1	0	R/W	サンプリングレートを選択します。本設定は、調歩同期式モード時のみ有効となります。
8	SRC0	0	R/W	000 : サンプリングレート 1/16 001 : サンプリングレート 1/5 010 : サンプリングレート 1/7 011 : サンプリングレート 1/11 100 : サンプリングレート 1/13 101 : サンプリングレート 1/17 110 : サンプリングレート 1/19 111 : サンプリングレート 1/27
7	C/A	0	R/W	コミュニケーションモード SCIF の動作モードを調歩同期式モードとクロック同期式モードのいずれかから選択します。 0 : 調歩同期式モード 1 : クロック同期式モード

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
6	CHR	0	R/W	<p>キャラクタレングス</p> <p>データ長を 7 ビット / 8 ビットデータのいずれかから選択します。</p> <p>本ビットは調歩同期モード時のみ有効です。クロック同期モードでは、CHR ビットの設定にかかわらず、データ長は 8 ビットデータ固定です。</p> <p>0 : 8 ビットデータ</p> <p>1 : 7 ビットデータ*</p> <p>【注】* 7 ビットデータを選択した場合、トランスミット FIFO データレジスタ (SCFTDR) の MSB (ビット 7) は送信されません。</p>
5	PE	0	R/W	<p>パリティイネーブル</p> <p>送信時にパリティビットの付加を、受信時にパリティビットのチェックを行うかどうかを選択します。本設定は、調歩同期モード時のみ有効です。クロック同期モードでは、PE ビットの設定にかかわらずパリティビットの付加およびチェックは行いません。</p> <p>0 : パリティビットの付加、およびチェックを禁止</p> <p>1 : パリティビットの付加、およびチェックを許可*</p> <p>【注】* PE ビットに 1 をセットすると送信時には、O/E ビットで指定した偶数、または奇数パリティを送信データに付加して送信します。受信時には、受信したパリティビットが O/E ビットで指定した偶数、または奇数パリティになっているかどうかをチェックします。</p>
4	O/E	0	R/W	<p>パリティモード</p> <p>パリティの付加やチェックを偶数パリティ、または奇数パリティのいずれで行うかを選択します。O/E ビットの設定は、PE ビットに 1 を設定しパリティビットの付加やチェックを許可した時のみ有効になります。クロック同期モードや、調歩同期モードで、パリティの付加やチェックを禁止している場合には、O/E ビットの設定は無効です。</p> <p>0 : 偶数パリティ*¹</p> <p>1 : 奇数パリティ*²</p> <p>【注】 *1 偶数パリティに設定すると送信時には、パリティビットと送信キャラクタを合わせて、その中の 1 の数の合計が偶数になるようにパリティビットを付加して送信します。 受信時には、パリティビットと受信キャラクタを合わせて、その中の 1 の数の合計が偶数であるかどうかをチェックします。</p> <p>*2 奇数パリティに設定すると送信時には、パリティビットと送信キャラクタを合わせて、その中の 1 の数の合計が奇数になるようにパリティビットを付加して送信します。 受信時には、パリティビットと受信キャラクタを合わせて、その中の 1 の数の合計が奇数であるかどうかをチェックします。</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
3	STOP	0	R/W	<p>ストップビットレングス</p> <p>ストップビットの長さを1ビット/2ビットのいずれかから選択します。</p> <p>なお、受信時にはSTOPビットの設定にかかわらず、受信したストップビットの1ビット目のみをチェックします。ストップビットの2ビット目が1の場合は、ストップビットとして扱いますが、0の場合は、次の送信キャラクタのスタートビットとして扱います。</p> <p>本設定は調歩同期式モード時のみ有効です。クロック同期式モードでは、ストップビットは付加されませんので、このビットの設定は無効です。</p> <p>0: 1ストップビット*¹</p> <p>1: 2ストップビット*²</p> <p>【注】 *1 送信時には、送信キャラクタの最後尾に1ビットの1(ストップビット)を付加して送信します。</p> <p>*2 送信時には、送信キャラクタの最後尾に2ビットの1(ストップビット)を付加して送信します。</p>
2	-	0	R	<p>リザーブビット</p> <p>読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。</p>
1 0	CKS1 CKS0	0 0	R/W R/W	<p>クロックセレクト</p> <p>内蔵ポーレートジェネレータのクロックソースを選択します。</p> <p>00: P</p> <p>01: P /4</p> <p>10: P /16</p> <p>11: P /64</p>

16.3.6 シリアルコントロールレジスタ (SCSCR)

SCSCRは、読み出し/書き込み可能な16ビットのレジスタで、SCIFの送信/受信動作、割り込み要求の許可/禁止、および送信/受信クロックソースの選択を行います。

ビット	ビット名	初期値	R/W	説明
15	TDRQE	0	R/W	<p>トランスミットデータ転送要求イネーブル</p> <p>送信時のTIE=1かつ送信FIFOデータエンプティ発生時に、送信FIFOデータエンプティ割り込み/DMA転送要求とするかを切り替えます。</p> <p>0: CPUへ割り込み要求を発行します</p> <p>1: DMACへ送信データ転送要求を発行します</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
14	RDRQE	0	R/W	<p>レシーブデータ転送要求イネーブル</p> <p>受信時の RIE = 1 かつ受信 FIFO データフル発生時に、受信 FIFO データフル割り込み / DMA 転送要求とするかを切り替えます。</p> <p>0 : CPU へ割り込み要求を発行します</p> <p>1 : DMAC へ受信データ転送要求を発行します</p>
13、12	-	すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>
11	TSIE	0	R/W	<p>トランスミットデータストップインタラプトイネーブル</p> <p>SCFCR の TSE ビットがイネーブルでかつ SCSSR の TSF フラグが 1 にセットされたとき、トランスミットデータストップ要因による割り込みの発生を許可 / 禁止します。</p> <p>0 : 送信データストップ割り込みを禁止*</p> <p>1 : 送信データストップ割り込みを許可</p> <p>【注】* 割り込み要求の解除は、TSF フラグの 1 を読み出したあと、0 にクリアするかまたは TSIE を 0 にクリアすることで行うことができます。</p>
10	ERIE	0	R/W	<p>受信エラーインタラプトイネーブル</p> <p>SCSSR の ER フラグが 1 にセットされたとき、受信エラー（フレーミングエラー / パリティエラー）要因による割り込みの発生を許可 / 禁止します。</p> <p>0 : 受信エラー割り込みを禁止*</p> <p>1 : 受信エラー割り込みを許可</p> <p>【注】* 割り込み要求の解除は、ER フラグの 1 を読み出したあと、0 にクリアするか、ERIE ビットを 0 にクリアすることで行えます。</p>
9	BRIE	0	R/W	<p>ブ레이크インタラプトイネーブル</p> <p>SCSSR の BRK フラグが 1 にセットされたとき、ブ레이크受信要因による割り込みの発生を許可 / 禁止します。</p> <p>0 : ブ레이크受信割り込みを禁止*</p> <p>1 : ブ레이크受信割り込みを許可</p> <p>【注】* 割り込み要求の解除は、BRK フラグの 1 を読み出したあと、0 にクリアするか、BRIE ビットを 0 にクリアすることで行えます。</p>
8	DRIE	0	R/W	<p>レシーブデータレディインタラプトイネーブル</p> <p>SCSSR の DR フラグが 1 にセットされたとき、レシーブデータレディ要因による割り込みの発生を許可 / 禁止します。</p> <p>0 : レシーブデータレディ割り込みを禁止*</p> <p>1 : レシーブデータレディ割り込みを許可</p> <p>【注】* 割り込み要求の解除は、DR フラグの 1 を読み出したあと、0 にクリアするか、DRIE ビットを 0 にクリアすることで行えます。</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
7	TIE	0	R/W	<p>トランスミットインタラプトイネーブル</p> <p>SCSSR の TDFE フラグが 1 にセットされたときに、送信 FIFO データエンブティ要因による割り込み要求の発生を許可 / 禁止します。</p> <p>0 : 送信 FIFO データエンブティ割り込み要求を禁止*</p> <p>1 : 送信 FIFO データエンブティ割り込み要求を許可</p> <p>【注】* 割り込み要求の解除は、SCFTDR に送信トリガ設定数より多い送信データを書き込み、TDFE フラグの 1 を読み出したあと、0 にクリアするか、または TIE を 0 にクリアすることで行うことができます。</p>
6	RIE	0	R/W	<p>レシーブインタラプトイネーブル</p> <p>SCSSR の RDF フラグが 1 にセットされたときの受信 FIFO データフル要因による割り込み要求の発生を許可 / 禁止します。</p> <p>0 : 受信 FIFO データフル割り込み要求を禁止*</p> <p>1 : 受信 FIFO データフル割り込み要求を許可</p> <p>【注】* 割り込み要求の解除は、RDF フラグの 1 を読み出したあと、0 にクリアするか、RIE ビットを 0 にクリアすることで行えます。</p>
5	TE	0	R/W	<p>トランスミットイネーブル</p> <p>シリアル送信動作の開始を許可 / 禁止します。</p> <p>0 : 送信動作を禁止</p> <p>1 : 送信動作を許可*</p> <p>【注】* TE ビットを 1 にセットする前に必ずシリアルモードレジスタ (SCSMR)、FIFO コントロールレジスタ (SCFCR) の設定を行い、送信フォーマットを決定し、送信 FIFO をリセットしてください。TE ビットがイネーブル状態のまま、上記各レジスタの設定を変更した場合は、動作の保証はできません。</p>
4	RE	0	R/W	<p>レシーブイネーブル</p> <p>シリアル受信動作の開始を許可 / 禁止します。</p> <p>0 : 受信動作を禁止*¹</p> <p>1 : 受信動作を許可*²</p> <p>【注】 *¹ RE ビットを 0 にクリアしても DR、ER、BRK、RDF、FER、PER、ORER の各フラグは影響を受けず、状態を保持しますので注意してください。</p> <p>*² RE ビットを 1 にセットする前に必ずシリアルモードレジスタ (SCSMR)、FIFO コントロールレジスタ (SCFCR) の設定を行い、受信フォーマットを決定し、受信 FIFO をリセットしてください。RE ビットがイネーブル状態のまま、上記各レジスタの設定を変更した場合は、動作の保証はできません。</p>
3、2	-	すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説 明
1	CKE1	0	R/W	クロックイネーブル
0	CKE0	0	R/W	<p>クロックソースを設定します。SCSMR で SCIF の動作モードを決定する前に、必ず CKE1、CKE0 ビットの設定をしてください。</p> <p>00：内部クロック / SCK 端子は入力端子（入力信号は無視）</p> <p>01：内部クロック / SCK 端子は同期クロック出力*¹</p> <p>10：外部クロック / SCK 端子はクロック入力*²</p> <p>11：外部クロック / SCK 端子はクロック入力*²</p> <p>内蔵ポーレートジェネレータでデータをサンプリングする場合は、CKE1、0 ビットを B'00（内部クロック / SCK0 端子は入力端子（入力信号は無視）と設定してください。</p> <p>また、SCK 端子をポートとして使用する場合、CKE1、0 ビットを B'00 に設定してください。</p> <p>【注】 *1 クロック同期式モードの場合、ビットレートと同じ周波数のクロックを出力</p> <p>*2 調歩同期式モードの場合、サンプリングレートに合わせたクロックを入力してください。たとえば、サンプリングレートが 1/16 の場合、ビットレートの 8 倍の周波数のクロックを入力してください。</p> <p>外部クロックを入力しない場合は、CKE1、CKE0 ビットを B'00 または B'01 に設定してください。</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

16.3.7 FIFO エラー数レジスタ (SCFER)

SCFER は、読み出し専用の 16 ビットのレジスタで、データの受信エラー（フレーミングエラー / パリティエラー）数を示します。

ビット	ビット名	初期値	R/W	説明
15、14	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。
13	PER5	0	R	パリティエラー数 調歩同期モード時に、レシーブ FIFO データレジスタ (SCFRDR) に格納されている受信データでパリティエラーの発生しているデータ数を示します。 SCSSR の ER ビットがセットされたあと、ビット 13~8 で示される値がパリティエラー発生データ数を表示します。 SCFRDR の 64 バイトの受信データすべてがパリティエラーをとまなう場合、PER5~PER0 は 0 を表示します。
12	PER4	0	R	
11	PER3	0	R	
10	PER2	0	R	
9	PER1	0	R	
8	PER0	0	R	
7、6	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。
5	FER5	0	R	フレーミングエラー数 調歩同期モード時に、レシーブ FIFO データレジスタ (SCFRDR) に格納されている受信データでフレーミングエラーの発生しているデータ数を示します。 SCSSR の ER ビットがセットされたあと、ビット 5~0 で示される値がフレーミングエラーの発生しているデータ数を表示します。 SCFRDR の 64 バイトの受信データすべてがフレーミングエラーをとまなう場合、FER5~FER0 は 0 を表示します。
4	FER4	0	R	
3	FER3	0	R	
2	FER2	0	R	
1	FER1	0	R	
0	FER0	0	R	

16.3.8 シリアルステータスレジスタ (SCSSR)

SCSSR は、読み出し / 書き込み可能な 16 ビットのレジスタで、ステータスを示します。

ただし、ORER、TSF、ER、TDFE、BRK、RDF、DR の各フラグへ 1 を書き込むことはできません。また、これらを 0 にクリアするためには、あらかじめ 1 を読み出ししておく必要があります。また、TEND フラグ、FER フラグおよび PER フラグは読み出し専用であり、書き込むことはできません。

ビット	ビット名	初期値	R/W	説明
15~10	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
9	ORER	0	R/(W)*	オーバランエラーフラグ 受信時にオーバランエラーが発生したことを示します。 本ビットは調歩同期モード時のみ有効となります。 0: 受信中、または正常に受信を完了したことを表示* ¹ 【クリア条件】 (1) パワーオンリセット、マニュアルリセット時 (2) ORER = 1 の状態を読み出したあと、0 を書き込んだとき 1: 受信時にオーバランエラーが発生したことを表示* ² 【セット条件】 受信 FIFO フルの状態で次のシリアル受信を完了したとき 【注】 *1 SCSSR の RE ビットを 0 にクリアしたときには、ORER フラグは影響を受けず以前の状態を保持します。 *2 SCFRDR ではオーバランエラーが発生する前の受信データを保持し、あとから受信したデータが失われます。さらに、ORER = 1 にセットされた状態で、以降のシリアル受信を続けることはできません。
8	TSF	0	R/(W)*	トランスミットデータストップフラグ 送信データ数が、SCTDSR の設定値と一致したことを示します。 0: 送信データ数が SCTDSR の値と一致していない 【クリア条件】 (1) パワーオンリセット、マニュアルリセット時 (2) TSF = 1 の状態を読み出したあと、0 を書き込んだとき 1: 送信データ数が SCTDSR の値と一致

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
7	ER	0	R/(W)*	<p>レシーブエラー</p> <p>調歩同期モードで、受信時にフレーミングエラー、パリティエラーが発生したことを示します。*1</p> <p>0: 受信時にフレーミングエラーまたはパリティエラーが発生していないことを表示</p> <p>【クリア条件】</p> <p>(1) パワーオンリセット、マニュアルリセット時</p> <p>(2) ER = 1 の状態を読み出したあと、0 を書き込んだとき</p> <p>1: 受信時にフレーミングエラーまたはパリティエラーが発生したことを表示</p> <p>【セット条件】</p> <p>(1) 受信終了時に受信データの最後尾のストップビットが1であるかどうかをチェックし、ストップビットが0であったとき*2</p> <p>(2) 受信時の受信データとパリティビットを合わせた1の数がシリアルモードレジスタ (SCSMR) の O/E ビットで指定した偶数 / 奇数パリティの設定と一致しなかったとき</p> <p>【注】 *1 SCSCR の RE ビットを0にクリアしたときには、ER フラグは影響を受けず以前の状態を保持します。レシーブエラーが発生しても受信データは SCFRDR に転送され、受信動作を続けます。</p> <p>SCFRDR から読み出したデータに受信エラーがあるかどうかは、SCSSR の FER、PER ビットで判定できます。</p> <p>*2 ストップレングスが2ビットのときは1ビット目のストップビットが1であるかどうかのみを判定し2ビット目のストップビットはチェックしません。</p>
6	TEND	1	R	<p>トランスミットエンド</p> <p>送信キャラクタの最後尾ビットの送信時に SCFTDR に有効なデータがなく、送信を終了したことを示します。</p> <p>0: 送信中であることを表示</p> <p>【クリア条件】</p> <p>SCFTDR へデータを書き込んだとき</p> <p>1: 送信を終了したことを表示</p> <p>【セット条件】</p> <p>1バイトのシリアル送信キャラクタの送信時に SCFTDR に送信データがないとき</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
5	TDFE	1	R/(W)*	<p>トランスミット FIFO データエンプティ</p> <p>トランスミット FIFO データレジスタ (SCFTDR) からトランスミットシフトレジスタ (SCTSR) にデータ転送が行われ、SCFTDR 内のデータ数が FIFO コントロールレジスタ (SCFCR) の TTRG1、TTRG0 ビットで設定した送信トリガデータ数以下になり、SCFTDR に送信データを書き込むことが可能になったことを示します。</p> <p>0: SCFTDR に送信トリガ設定数より多い送信データが書き込まれていることを表示</p> <p>【クリア条件】</p> <p>SCFTDR に送信トリガ設定数を超える送信データを書き込み、TDFE = 1 の状態を読み出したあと、0 を書き込んだとき</p> <p>1: SCFTDR の送信データ数が送信トリガ設定数以下であることを表示</p> <p>【セット条件】</p> <p>(1) パワーオンリセット、マニュアルリセット時</p> <p>(2) SCFTDR の送信データ数が送信動作によって送信トリガ設定数以下になったとき*</p> <p>【注】* SCFTDR は 64 バイトの FIFO レジスタですので TDFE = 1 で書き込むことができる最大データ数は、(64 - 送信トリガ設定数) になります。これより多くデータを書き込んだ場合は無視されます。また、SCFTDR 内のデータ数は SCFCR に示されます。</p>
4	BRK	0	R/(W)*	<p>ブ레이크検出</p> <p>調歩同期モードで、受信データのブ레이크信号の検出を示します。</p> <p>0: ブ레이크信号を受信していないことを表示</p> <p>【クリア条件】</p> <p>(1) パワーオンリセット、マニュアルリセット時</p> <p>(2) BRK = 1 の状態を読み出したあと、0 を書き込んだとき</p> <p>1: ブ레이크信号を受信したことを表示*</p> <p>【セット条件】</p> <p>フレーミングエラーをとまなうデータを受信したとき、引き続き 1 フレーム長以上スペース 0 (ローレベル) の場合</p> <p>【注】* ブ레이크検出すると検出後の受信データ (H'00) の SCFCR 転送は停止します。</p> <p>ブ레이크が終了し、受信信号がマーク 1 に戻ると受信データの転送が再開します。</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
3	FER	0	R	<p>フレーミングエラー表示</p> <p>調歩同期式モードで、レシーブ FIFO データレジスタ (SCFRDR) から読み出したデータのフレーミングエラーを表示します。</p> <p>0: SCFRDR から読み出した受信データにフレーミングエラーがないことを表示</p> <p>【クリア条件】</p> <p>(1) パワーオンリセット、マニュアルリセット時</p> <p>(2) SCFRDR 読み出しデータにフレーミングエラーなし</p> <p>1: SCFRDR から読み出した受信データにフレーミングエラーが発生していることを表示</p> <p>【セット条件】</p> <p>SCFRDR 読み出しデータにフレーミングエラーあり</p>
2	PER	0	R	<p>パリティエラー表示</p> <p>調歩同期式モードで、レシーブ FIFO データレジスタ (SCFRDR) から読み出したデータのパリティエラーを表示します。</p> <p>0: SCFRDR から読み出した受信データにパリティエラーがないことを表示</p> <p>【クリア条件】</p> <p>(1) パワーオンリセット、マニュアルリセット時</p> <p>(2) SCFRDR 読み出しデータにパリティエラーなし</p> <p>1: SCFRDR から読み出した受信データにパリティエラーが発生していることを表示</p> <p>【セット条件】</p> <p>SCFRDR 読み出しデータにパリティエラーあり</p>
1	RDF	0	R/(W)*	<p>レシーブ FIFO データフル</p> <p>受信したデータがレシーブシフトレジスタ (SCRSR) からレシーブ FIFO データレジスタ (SCFRDR) に転送され、SCFRDR 内の受信データ数が、FIFO コントロールレジスタ (SCFCR) の RTRG1、RTRG0 ビットで設定した受信トリガデータ数以上になったことを示します。</p> <p>0: SCFRDR 内の受信データ数が受信トリガ設定数より少ないことを表示</p> <p>【クリア条件】</p> <p>(1) パワーオンリセット、マニュアルリセット時</p> <p>(2) SCFRDR 内の受信データ数が受信トリガ設定数より少なくなるまで SCFRDR を読み出し、RDF = 1 を読み出したあと、0 を書き込んだとき</p> <p>1: SCFRDR 内の受信データ数が受信トリガ設定数以上であることを表示</p> <p>【セット条件】</p> <p>SCFRDR に受信トリガ設定数以上の受信データが格納されたとき*</p> <p>【注】* SCFRDR は 64 バイトの FIFO レジスタです。RDF = 1 で少なくとも受信トリガ設定数のデータを読み出すことができます。SCFRDR が空の状態ではデータを読み出すと不定値が読み出されます。なお SCFRDR 内の受信データ数は SCFDR の下位ビットに示されます。</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
0	DR	0	R/(W) *	<p>レシーブデータレディ</p> <p>調歩同期式モードで、レシーブ FIFO データレジスタ (SCFRDR) に受信トリガ設定数未満のデータがあり、かつ次のデータが来ないことを示します。</p> <p>0: 受信中または正常に受信完了して SCFRDR に受信データが残っていないことを表示。</p> <p>【クリア条件】</p> <p>(1) パワーオンリセット、マニュアルリセット時</p> <p>(2) SCFRDR 内の受信データをすべて読み出し、DR = 1 を読み出したあと、0 を書き込んだとき</p> <p>1: 次の受信データが来ないことを表示</p> <p>【セット条件】</p> <p>SCFRDR に受信トリガ設定数未満のデータがあり、かつ次のデータが来ないとき*</p> <p>【注】 * DR ビットのセットタイミングは、SCSMR のサンプリングコントロールビットの設定に関わらず、サンプリングレート 1/16 での 15etu 後となります。</p> <p>【記号説明】 etu (Elementary Time Unit : 1 ビットの転送期間の略)</p>

【注】 * フラグをクリアするための 0 書き込みのみ可能です。

16.3.9 ビットレートレジスタ (SCBRR)

SCBRR は、読み出し / 書き込み可能な 8 ビットのレジスタで、シリアルモードレジスタ (SCSMR) の CKS1、CKS0 ビットで選択されるボーレートジェネレータの動作クロックと合わせて、シリアル送信 / 受信のビットレートを設定します。

ビット	ビット名	初期値	R/W	説明
7~0	SCBRD7~0	H'FF	R/W	ビットレート設定

SCBRR の設定値は以下の計算式で求められます。

【調歩同期式モードのとき】

(サンプリングレート 1/16 の場合)

$$N = \frac{P}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(サンプリングレート 1/5 の場合)

$$N = \frac{P}{10 \times 2^{2n-1} \times B} \times 10^6 - 1$$

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

(サンプリングレート 1/11 の場合)

$$N = \frac{P}{22 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(サンプリングレート 1/13 の場合)

$$N = \frac{P}{26 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(サンプリングレート 1/27 の場合)

$$N = \frac{P}{54 \times 2^{2n-1} \times B} \times 10^6 - 1$$

【クロック同期モードのとき】

$$N = \frac{P}{4 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B : ビットレート (bit/s)

N : ボーレートジェネレータの SCBRR の設定値

調歩同期モード時 (0 N 255)

クロック同期モード時 (1 N 255)

P : 周辺モジュール動作周波数 (MHz)

n : ボーレートジェネレータ入力クロック (n=0、1、2、3)

(n とクロックの関係は、次表を参照してください)

n	クロック	SCSMR の設定値	
		CKS1	CKS0
0	P	0	0
1	P /4	0	1
2	P /16	1	0
3	P /64	1	1

調歩同期モードのビットレート誤差は、以下の計算式で求められます。

(サンプリングレート 1/16 の場合)

$$\text{誤差 (\%)} = \left(\frac{P \times 10^6}{(1+N) \times B \times 32 \times 2^{2n-1}} - 1 \right) \times 100$$

(サンプリングレート 1/5 の場合)

$$\text{誤差 (\%)} = \left(\frac{P \times 10^6}{(1+N) \times B \times 10 \times 2^{2n-1}} - 1 \right) \times 100$$

(サンプリングレート 1/11 の場合)

$$\text{誤差 (\%)} = \left(\frac{P \times 10^6}{(1+N) \times B \times 22 \times 2^{2n-1}} - 1 \right) \times 100$$

(サンプリングレート 1/13 の場合)

$$\text{誤差 (\%)} = \left(\frac{P \times 10^6}{(1+N) \times B \times 26 \times 2^{2n-1}} - 1 \right) \times 100$$

(サンプリングレート 1/27 の場合)

$$\text{誤差 (\%)} = \left(\frac{P \times 10^6}{(1+N) \times B \times 54 \times 2^{2n-1}} - 1 \right) \times 100$$

16.3.10 FIFO コントロールレジスタ (SCFCR)

SCFCR は、読み出し / 書き込み可能な 16 ビットのレジスタで、送信、受信おのこの FIFO レジスタのデータ数リセット、およびトリガデータ数の設定を行います。また、ループバックテストの許可ビットを含んでいます。

ビット	ビット名	初期値	R/W	説明
15	TSE	0	R/W	トランスミットデータストップイネーブル 送信データストップ機能を許可 / 禁止します。本機能は、調歩同期式でのみ設定可能です。クロック同期式では本機能はサポートしていないので、本ビットは 0 に設定してください。 0 : 送信データストップ機能を禁止 1 : 送信データストップ機能を許可
14	TCRST	0	R/W	トランスミットカウントリセット 送信カウンタを 0 にクリアします。本ビットは、送信データストップ機能使用時のみ有効となります。 0 : 送信カウントリセットを禁止* 1 : 送信カウントリセットを許可 (0 にクリア) 【注】* パワーオンリセット、マニュアルリセット時には送信カウントリセット (0 にクリア) が行われます。
13~11		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
10 9 8	RSTRG2 RSTRG1 RSTRG0	0 0 0	R/W R/W R/W	<p>RTS 出力アクティブトリガ</p> <p>レシーブ FIFO データレジスタ (SCFRDR) 内に格納された受信データ数が以下に示すトリガ設定数以上になったとき、RTS 信号はハイレベルになります。</p> <p>000 : 63 001 : 1 010 : 8 011 : 16 100 : 32 101 : 48 110 : 54 111 : 60</p>
7 6	RTRG1 RTRG0	0 0	R/W R/W	<p>レシーブ FIFO データ数トリガ</p> <p>シリアルステータスレジスタ (SCSSR) のレシーブデータフル (RDF) フラグをセットする受信データ数を設定するビットです。</p> <p>レシーブ FIFO データレジスタ (SCFRDR) 内に格納された受信データ数が以下に示す受信トリガ設定数以上になったとき RDF フラグをセットします。</p> <p>00 : 1 01 : 16 10 : 32 11 : 48</p>
5 4	TTRG1 TTRG0	0 0	R/W R/W	<p>トランスミット FIFO データ数トリガ</p> <p>シリアルステータスレジスタ (SCSSR) のトランスミット FIFO データレジスタエンプティ (TDFE) フラグをセットする残りの送信データ数を設定するビットです。</p> <p>送信動作によりトランスミット FIFO データレジスタ (SCFTDR) 内の送信データ数が、以下に示す送信トリガ設定数以下になったとき TDFE フラグをセットします。</p> <p>00 : 32 (32) 01 : 16 (48) 10 : 2 (62) 11 : 0 (64)</p> <p>【注】 () 内の値はフラグ発生時の SCFTDR の空き数を示します。</p>
3	MCE	0	R/W	<p>モデムコントロールイネーブル</p> <p>モデムコントロール信号 CTS、RTS を有効にします。</p> <p>本ビットは、調歩同期式モード時のみ有効です。</p> <p>0 : モデム信号を無効* 1 : モデム信号を有効</p> <p>【注】 * CTS は入力値にかかわらず無効に、RTS は 0 に固定します。</p>

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
2	TFRST	0	R/W	トランスミット FIFO データレジスタリセット トランスミット FIFO データレジスタ内の送信データを無効とし、空の状態にリセットします。 0: リセット動作を禁止* 1: リセット動作を許可 【注】* パワーオンリセット、マニュアルリセット時にはリセット動作が行われます。
1	RFRST	0	R/W	レシーブ FIFO データレジスタリセット レシーブ FIFO データレジスタ内の受信データを無効とし、空の状態にリセットします。 0: リセット動作を禁止* 1: リセット動作を許可 【注】* パワーオンリセット、マニュアルリセット時にはリセット動作が行われます。
0	LOOP	0	R/W	ループバックテスト 送信出力端子 (TXD) と受信入力端子 (RXD)、 \overline{RTS} 端子と \overline{CTS} 端子を内部で接続し、ループバックテストを可能にします。 0: ループバックテストを禁止 1: ループバックテストを許可

16.3.11 FIFO データ数レジスタ (SCFDR)

SCFDR は、読み出し専用の 16 ビットのレジスタで、トランスミット FIFO データレジスタ (SCFTDR) およびレシーブ FIFO データレジスタ (SCFRDR) 内に格納されているデータ数を示します。

ビット 14~8 で SCFTDR 内の送信データ数を、ビット 6~0 で SCFRDR 内の受信データ数を示します。

ビット	ビット名	初期値	R/W	説明
15	-	0	R	リザーブビット 読み出すと常に 0 が読み出されます。
14	T6	0	R	SCFTDR 内に格納されている未送信のデータ数を示します。 H'00 は送信データがないことを、H'40 は SCFTDR にいっぱいの送信データが格納されていることを示します。
13	T5	0	R	
12	T4	0	R	
11	T3	0	R	
10	T2	0	R	
9	T1	0	R	
8	T0	0	R	
7	-	0	R	リザーブビット 読み出すと常に 0 が読み出されます。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

ビット	ビット名	初期値	R/W	説明
6	R6	0	R	SCFRDR 内に格納されている受信データ数を示します。 H'00 は受信データがないことを、H'40 は SCFRDR にいっぱい受信データが格納されていることを示します。
5	R5	0	R	
4	R4	0	R	
3	R3	0	R	
2	R2	0	R	
1	R1	0	R	
0	R0	0	R	

16.3.12 トランスミットデータストップレジスタ (SCTDSR)

SCTDSR は、読み出し / 書き込み可能な 8 ビットレジスタで、送信データ数を設定します。本レジスタは FIFO コントロールレジスタ (SCFCR) の TSE ビットがイネーブル時のみ有効となります。送信動作は、本レジスタで設定したデータ数を送信すると停止します。設定可能な値は、H'00 (1 バイト) ~ H'FF (256 バイト) です。本機能は調歩同期式モードでのみ有効です。

本レジスタの初期値は H'FF です。

16.4 動作説明

16.4.1 概要

SCIF は、キャラクタ単位で同期をとりながら通信する調歩同期モードと、クロックに同期してシリアル通信を行うクロック同期式モードをサポートしています。

送受信おののに 64 段のバッファ (FIFO) を内蔵しており、CPU のオーバヘッドを減らし、高速連続通信が可能です。

16.4.2 調歩同期式モード

以下に調歩同期式モードについて説明します。

送受信フォーマットの選択は、シリアルモードレジスタ (SCSMR) で行います。これを表 16.2 に示します。また、SCIF のクロックソースは、シリアルコントロールレジスタ (SCSCR) の CKE1、CKE0 ビットで決まります。

- データ長：7ビット / 8ビットから選択可能
- パリティの付加および1ビット / 2ビットのストップビットの付加を選択可能 (これらの組み合わせにより送信 / 受信フォーマット、およびキャラクタ長を決定)
- 受信時にフレーミングエラー、パリティエラー、オーバランエラー、レシーブFIFOデータフル、レシーブデータレディ、およびブレークの検出が可能
- 送受信FIFOレジスタおののの格納データ数を表示

- クロックソース：内部クロック / 外部クロックから選択可能

内部クロックを選択した場合：

ボーレートジェネレータのクロックで動作

外部クロックを選択した場合：

サンプリングレートに合わせたクロックを入力することが必要。たとえば、サンプリングレートが1/16の場合、ビットレートの8倍の周波数のクロックを入力することが必要

(内蔵ボーレートジェネレータを使用しない)

表 16.2 SCSMR の設定値とシリアル送信 / 受信フォーマット

SCSMR の設定値			モード	SCIF の送信 / 受信フォーマット		
ビット 6	ビット 5	ビット 3		データ長	パリティ ビット	ストップ ビット長
CHR	PE	STOP				
0	0	0	調歩同期式 モード	8 ビット データ	なし	1 ビット
		1				2 ビット
	1	0			あり	1 ビット
		1			2 ビット	
1	0	0		7 ビット データ	なし	1 ビット
		1				2 ビット
	1	0			あり	1 ビット
		1			2 ビット	

16.4.3 調歩同期式モードのシリアル動作

(1) 送信 / 受信フォーマット

設定可能な送信 / 受信フォーマットを、表 16.3 に示します。

送信 / 受信フォーマットは 8 種類あり、シリアルモードレジスタ (SCSMR) の設定により選択できます。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

表 16.3 シリアル送信 / 受信フォーマット

SCSMRの設定			シリアル送信 / 受信フォーマットとフレーム長											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START 8ビットデータ STOP											
		1	START 8ビットデータ STOP								STOP	STOP		
	1	0	START 8ビットデータ P								STOP			
		1	START 8ビットデータ P								STOP	STOP		
1	0	0	START 7ビットデータ STOP											
		1	START 7ビットデータ STOP							STOP	STOP			
	1	0	START 7ビットデータ P							STOP				
		1	START 7ビットデータ P							STOP	STOP			

(2) クロック

SCIFの送受信クロックは、シリアルコントロールレジスタ (SCSCR) の CKE1、CKE0 ビットの設定により、内蔵ポーレートジェネレータの生成した内部クロック、または SCK 端子から入力された外部クロックの2種類から選択できます。

外部クロックを SCK 端子に入力する場合には、サンプリングレートに合わせたクロックを入力してください。たとえば、サンプリングレートが 1/16 の場合、ビットレートの 8 倍の周波数のクロックを入力してください。

(3) データの送信 / 受信動作

(a) SCIF の初期化

データの送信 / 受信前には、まず SCSCR の TE ビット、および RE ビットを 0 にクリアしたあと、以下の順で SCIF を初期化してください。

通信フォーマットの変更などの場合には必ず、TE ビットおよび RE ビットを 0 にクリアしてから次の手順で変更を行ってください。TE ビットを 0 にクリアすると、トランスミットシフトレジスタ (SCTSR) が初期化されます。TE、RE ビットを 0 にクリアしても、シリアルステータスレジスタ (SCSSR)、トランスミット FIFO データレジスタ (SCFTDR) および、レシーブ FIFO データレジスタ (SCFRDR) の内容は保持されますので注意してください。TE ビットの 0 クリアは、送信データをすべて送信し SCSSR の TEND ビットが 1 にセットされたあとに行ってください。送信中でも 0 クリア可能ですが、送信中のデータは 0 クリア後、ハイインピーダンス状態になります。また再度 TE ビットを 1 にセットして送信開始する前に SCFCR の TFRST ビットをいったん 1 にセットして SCFTDR をリセットしてください。

外部クロックを使用している場合には、動作が不確実にになりますので初期化を含めた動作中にクロックを止めないでください。

図 16.2 に SCIF の初期化フローチャートの例を示します。

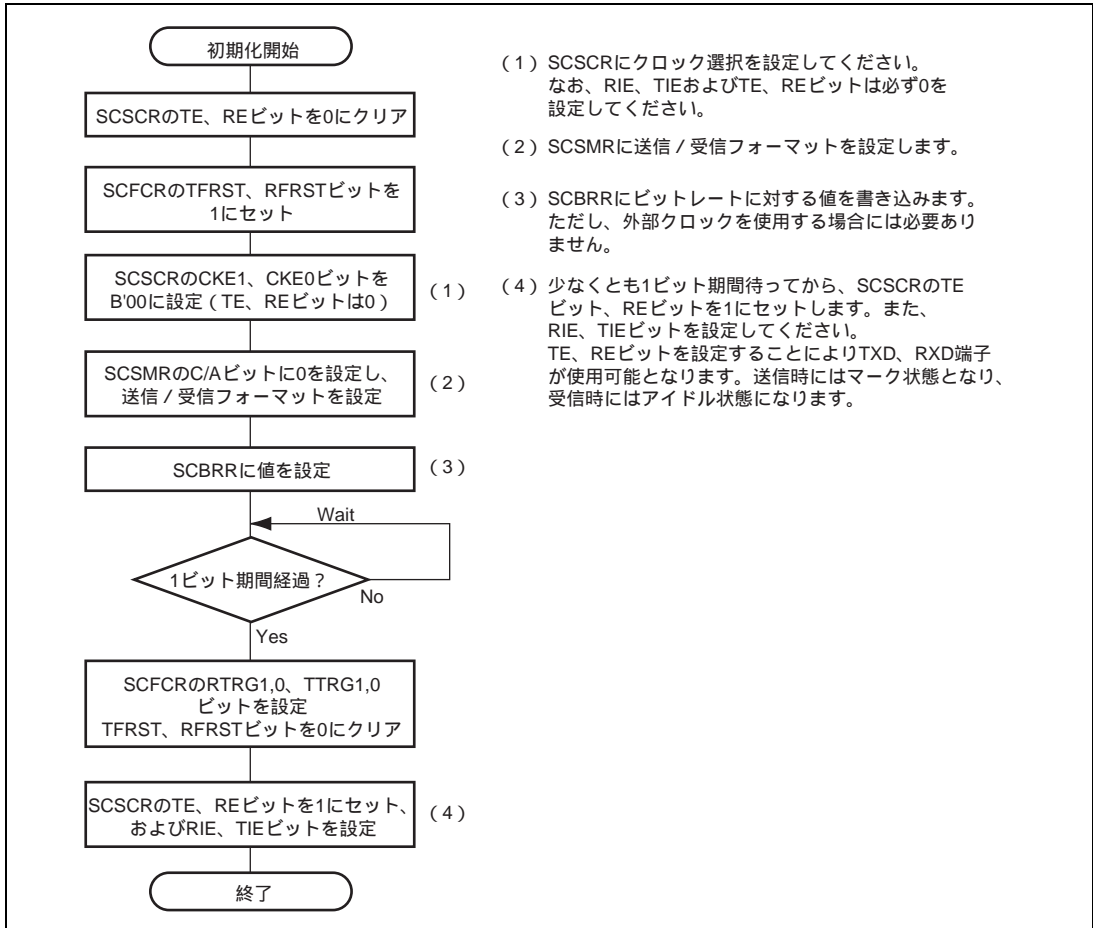


図 16.2 SCIF の初期化フローチャートの例

(b) シリアルデータ送信

図 16.3 にシリアル送信のフローチャートの例を示します。

シリアルデータ送信は、SCIF を送信動作可能状態に設定したあと、以下の手順を参考に行ってください。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

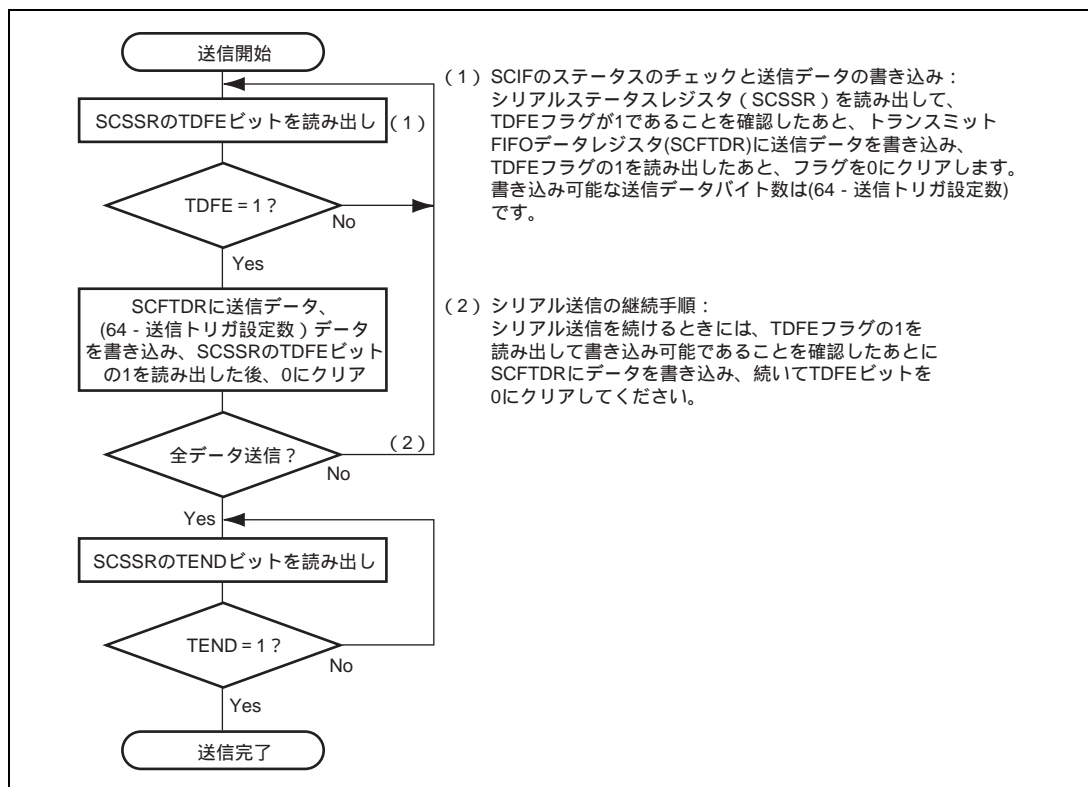


図 16.3 シリアル送信のフローチャートの例

SCIF はシリアル送信時に以下のように動作します。

1. SCIFは、トランスミットFIFOデータレジスタ (SCFTDR) にデータが書き込まれると、SCFTDRからトランスミットシフトレジスタ (SCTSR) にデータを転送し、送信を開始します。SCFTDRにはシリアルステータスレジスタ (SCSSR) のTDFEフラグが1であることを確認して送信データを書き込んでください。書き込み可能なデータ数は少なくとも (64 - 送信トリガ設定) 数です。
2. SCFTDRからSCTSRへデータが転送され、送信を開始すると、SCFTDRに送信データがなくなるまで連続して送信動作を続けます。途中、SCFTDR内の送信データ数がFIFOコントロールレジスタ (SCFCR) で設定した送信トリガ数以下になったとき、TDFEフラグをセットします。

このとき、シリアルコントロールレジスタ (SCSCR) のTIEビットが1にセットされていると送信FIFOデータエンプティ要因による割り込み要求を発生します。

送信データストップ機能使用時、トランスミットデータストップレジスタ (SCTDSR) に設定したデータ数と一致すると、送信動作を停止し、シリアルステータスレジスタ (SCSSR) のTSFフラグをセットします。

このとき、シリアルコントロールレジスタ (SCSCR) のTSIEビットが1にセットされていると送信データストップ要因による割り込み要求を発生します。ただし、送信FIFOデータエンプティ割り込みと送信データス

トップ割り込みのベクタは共通です。

シリアル送信データは、以下の順に TXD 端子から送り出されます。

- スタートビット：1ビットの0が出力されます。
- 送信データ：8ビット、または7ビットのデータがLSBから順に出力されます。
- パリティビット（偶数パリティ、または奇数パリティ）が出力されます。
なお、パリティビットを出力しないフォーマットも選択できます。
- ストップビット：1ビットまたは2ビットの1（ストップビット）が出力されます。
- マーク状態：次の送信を開始するスタートビットを送り出すまで1を出力し続けます。

3. SCIFは、ストップビットを送出するタイミングでSCFTDRの送信データをチェックします。

データがあるとSCFTDRからSCTSRにデータを転送し、ストップビットを送り出したあと、次フレームのシリアル送信を開始します。

送信データがないとシリアルステータスレジスタ（SCSSR）のTENDフラグに1をセットし、ストップビットを送り出したあと、1を出力するマーク状態になります。

調歩同期式モードの送信時の動作例を図 16.4 に示します。

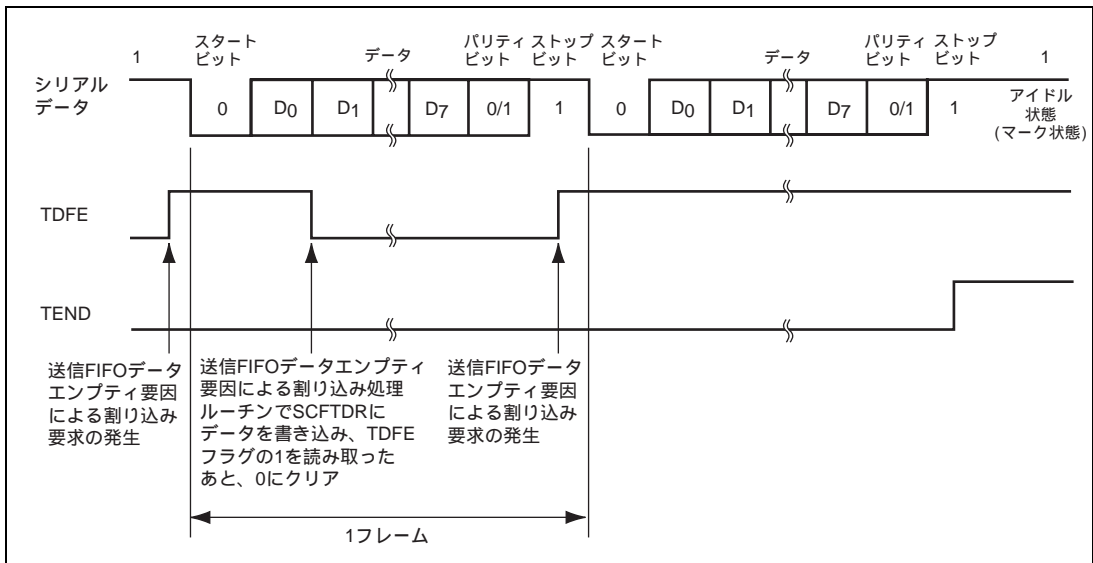


図 16.4 送信時の動作例（8ビットデータ/パリティあり/1ストップビットの例）

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

送信データストップ機能

送信データストップ機能とは、SCTDSRレジスタの値と送信データ数が一致すると、送信動作を停止する機能です。TSIEビット（割り込みイネーブルビット）をセットしておくことで、割り込みの発生及びDMACの起動を行うことができます。

送信データストップ機能の動作例を図 16.5 に示します。

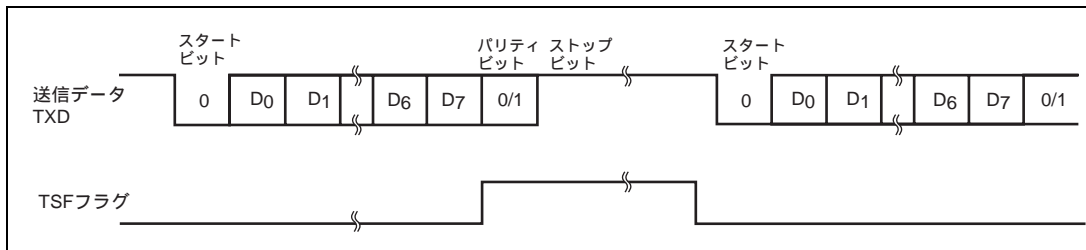


図 16.5 送信データストップ機能の動作例

次に、送信データストップ機能のフローチャートを図 16.6 に示します。

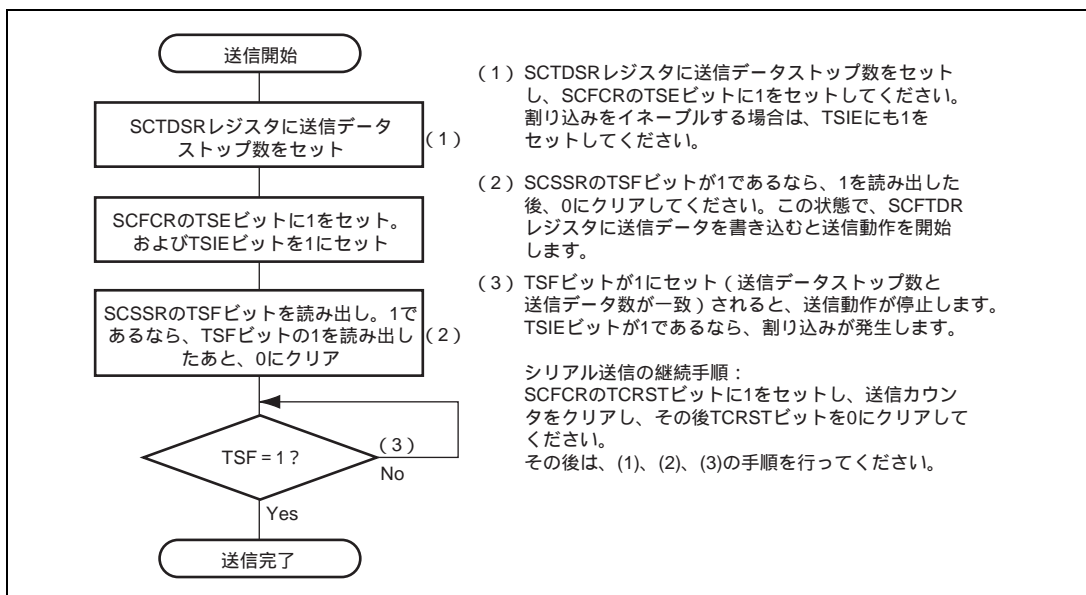


図 16.6 送信データストップ機能のフローチャート

(c) シリアルデータ受信

図 16.7、図 16.8 にシリアル受信フローチャートの例を示します。

シリアルデータ受信は、SCIF を受信動作可能状態に設定したあと、以下の手順に従い行ってください。

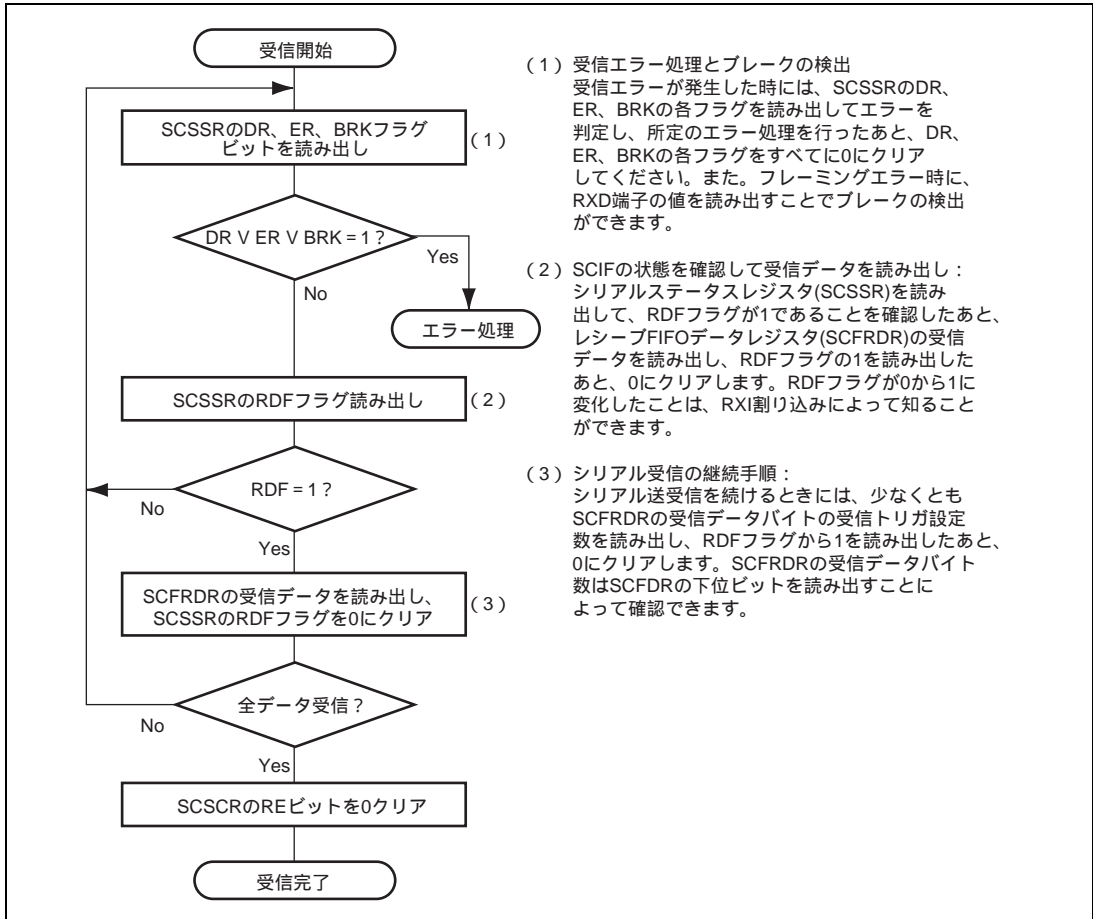


図 16.7 シリアル受信のフローチャートの例 (1)

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

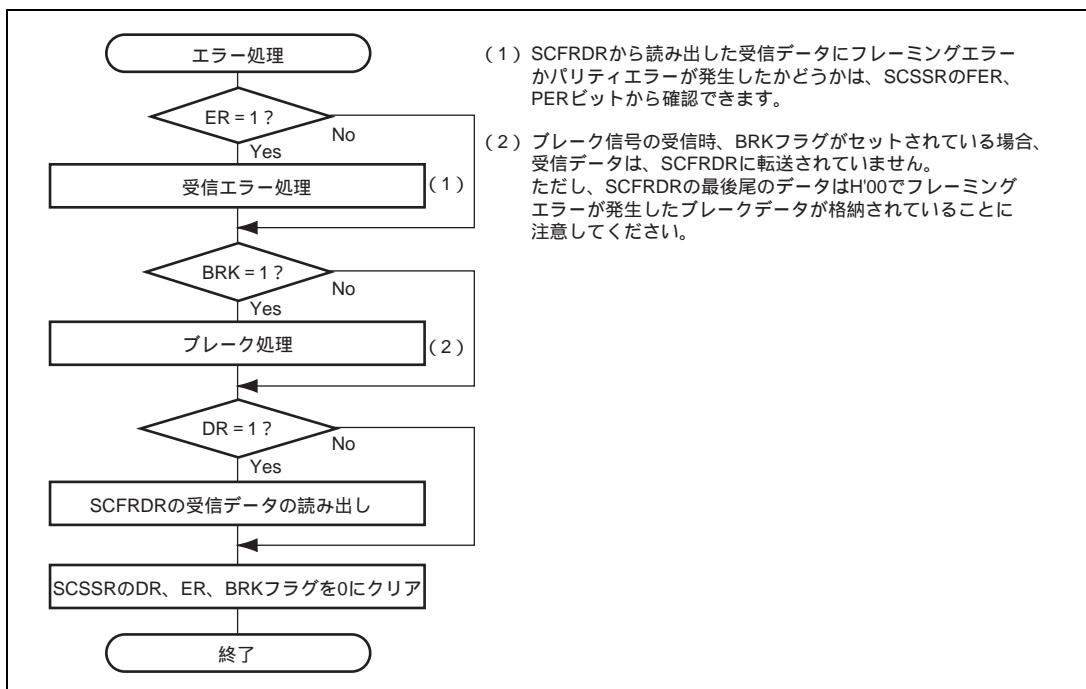


図 16.8 シリアル受信のフローチャートの例 (2)

SCIF は受信時に以下のように動作します。

1. SCIFは通信回線を監視し、スタートビットの0を検出すると内部を同期化し、受信を開始します。
2. 受信したデータをSCSSRのLSBからMSBの順に格納します。
3. パリティビット、およびストップビットを受信します。

受信後、SCIF は以下のチェックを行います。

- ストップビットチェック：ストップビットが1であるかをチェックします。
ただし、2ストップビットの場合、1ビット目のストップビットのみをチェックします。
- 受信データをレシフシフトレジスタ (SCSSR) からSCFRDRに転送できる状態であるかをチェックします。
- ブレークチェック：BRKフラグが0であり、ブレーク状態でないことをチェックします。

以上のチェックがパスしたとき、SCFRDR に受信データが格納されます。

【注】 受信エラー (フレーミングエラー / パリティエラー) が発生しても受信動作を続けます。

4. RDFフラグが1になったとき、SCSCRのRIEビットが1にセットされていると受信FIFOデータフル要因による割り込み要求が発生します。

また、ERフラグが1になったとき、SCSCRのERIEビットが1にセットされていると受信エラー要因による割り込み要求が発生します。

BRKフラグが1になったとき、SCSCRのBRIEビットが1にセットされていると、ブ레이크受信要因による割り込み要求が発生します。

DRフラグが1になったとき、SCSCRのDRIEビットが1にセットされていると、レシーブデータレディ要因による割り込み要求が発生します。

各要因で発生する割り込みのベクタは共通です。

調歩同期式モード受信時の動作例を図 16.9 に示します。

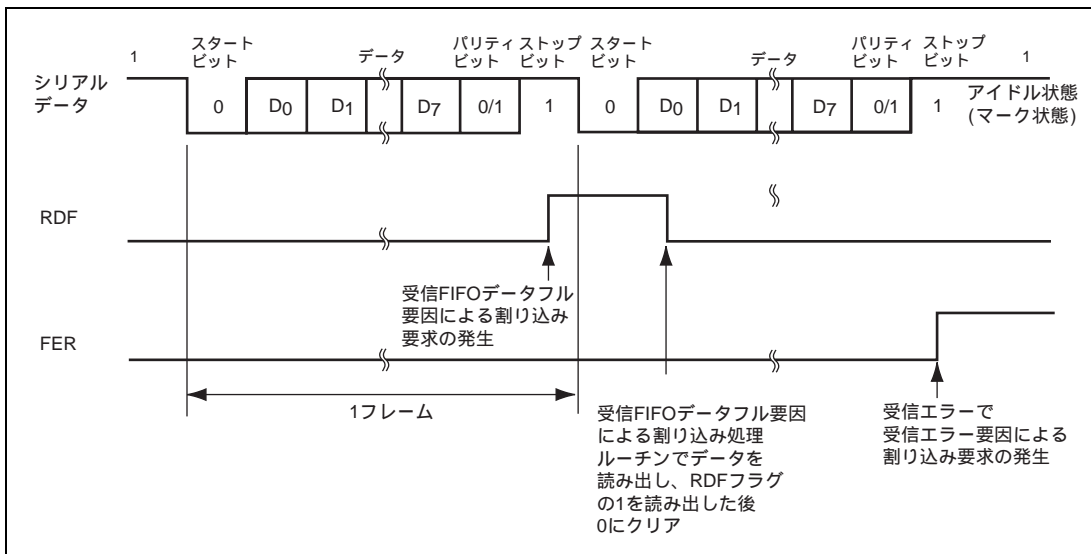


図 16.9 SCIF の受信時の動作例 (8 ビットデータ / パリティあり / 1 ストップビットの例)

(d) モデム機能使用時の送受信

モデム機能を使用した場合、 $\overline{\text{CTS}}$ 入力値に応じて送信を停止、再開することができます。 $\overline{\text{CTS}}$ が 1 にセットされると、送信中である場合 1 フレームの送信後マーク状態になります。 $\overline{\text{CTS}}$ が 0 にセットされると、次の送信データがスタートビットを先頭に出力されます。

$\overline{\text{CTS}}$ 制御の動作例を図 16.10 に示します。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

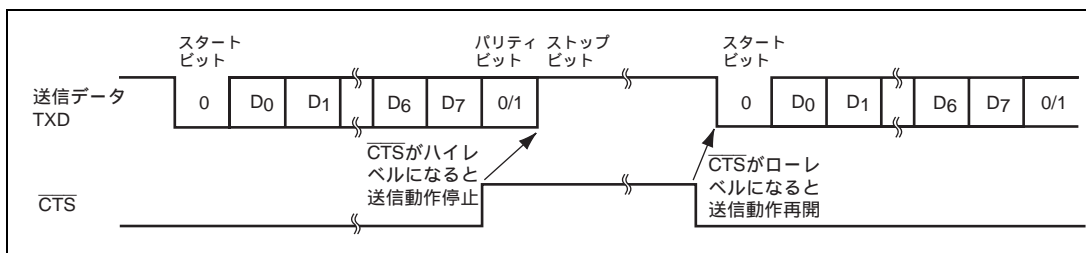


図 16.10 $\overline{\text{CTS}}$ 制御の動作例

また、モデム機能を使用した場合、受信 FIFO (SCFRDR) が $\overline{\text{RTS}}$ 出力トリガ数以上になったとき、 $\overline{\text{RTS}}$ 信号はハイレベルになります。

次に $\overline{\text{RTS}}$ 制御の動作例を図 16.11 に示します。

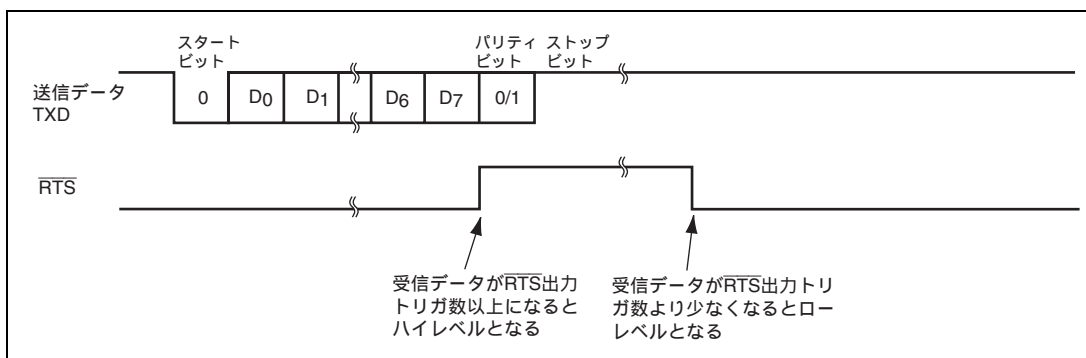


図 16.11 $\overline{\text{RTS}}$ 制御の動作例

16.4.4 クロック同期式モード

以下にクロック同期式モードについて説明します。

送受信おののに 64 段の FIFO バッファを内蔵しており、CPU のオーバーヘッドを減らし、高速連続通信が可能です。

動作クロックソースの選択は、シリアルモードレジスタ (SCSMR) で行います。また、SCIF のクロックソースは、シリアルコントロールレジスタ (SCSCR) の CKE1、CKE0 で決まります。

- 送受信フォーマット：8ビットデータ固定
- 送受信FIFOレジスタおののの格納データ数を表示
- SCIFのクロックソース：内部クロック / 外部クロックから選択可能

内部クロックを使用した場合：

SCIFはポーレートジェネレータのクロックで動作し、同期クロックを外部へ出力

外部クロックを選択した場合：

SCK端子から入力された外部周期クロックで動作

16.4.5 クロック同期式モードのシリアル動作

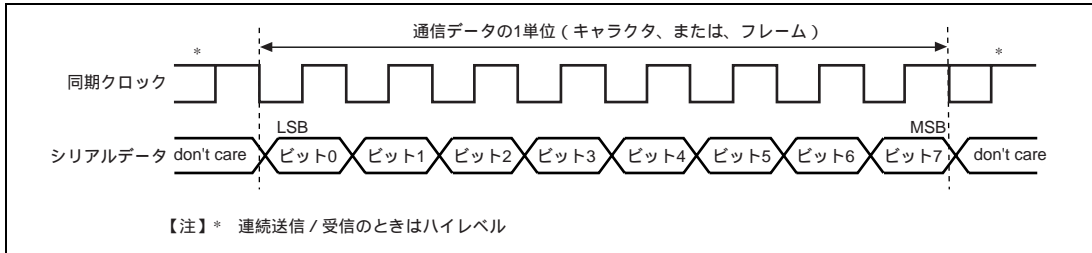


図 16.12 クロック同期式通信のデータフォーマット

クロック同期式シリアル通信では、通信回線のデータは同期クロックの立ち下がりから次の立ち下がりまで出力されます。また、同期クロックの立ち上がりでデータの確定が保証されます。

シリアル通信の1キャラクタは、データのLSBから始まり最後にMSBが出力されます。MSB出力後の通信回線の状態はMSBの状態を保ちます。

クロック同期式モードでは、SCIFは同期クロックの立ち上がりに同期してデータを受信します。

(1) 送信 / 受信フォーマット

8ビットデータ固定です。

パリティビットやマルチプロセッサビットの付加はできません。

(2) クロック

SCSCRのCKE1、CKE0ビットの設定により内蔵ポーレートジェネレータの生成した内部クロックまたはSCK端子から入力された外部周期クロックの2種類から選択できます。

同期クロックは1キャラクタの送受信で8パルス出力され、送信 / 受信を行わないときにはハイレベルに固定されます。ただし、受信のみの動作のときは、REビットが1にセットされている間、同期クロックは出力し続けます。

1キャラクタ単位で同期クロックをハイレベルに固定したいときは、受信するデータ数と同じ数のデータをトランスミットFIFOデータレジスタ(SCFTDR)へ書き込み、同時にTE、REビットを1にセットして送信データをダミーで送信してください。送信データ数が送信されると同期クロックはハイレベルに固定されます。

(3) データの送信 / 受信動作

(a) SCIFの初期化

データの送信 / 受信前には、まずSCSCRのTEビット、およびREビットを0にクリアしたあと、以下の順でSCIFを初期化してください。

クロックソースの変更などの場合には必ず、TEビットおよびREビットを0にクリアしてから次の手順で変更を行ってください。TEビットを0にクリアすると、トランスミットシフトレジスタ(SCTSR)が初期化されます。TE、REビットを0にクリアしても、シリアルステータスレジスタ(SCSSR)、トランスミットFIFOデータレジスタ(SCFTDR)および、レシーブFIFOデータレジスタ(SCFRDR)の内容は保持されますので注意してくださ

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

い。TE ビットの 0 クリアは、送信データをすべて送信し SCSSR の TEND ビットが 1 にセットされたあとに行ってください。送信途中には TE ビットを 0 にクリアしないでください。TE ビットを 0 にクリアすると、TXD 端子はハイインピーダンス状態となります。また再度 TE ビットを 1 にセットして送信開始する前に SCFCR の TFRST ビットをいったん 1 にセットして SCFTDR をリセットしてください。

図 16.13 に SCIF の初期化フローチャートの例を示します。

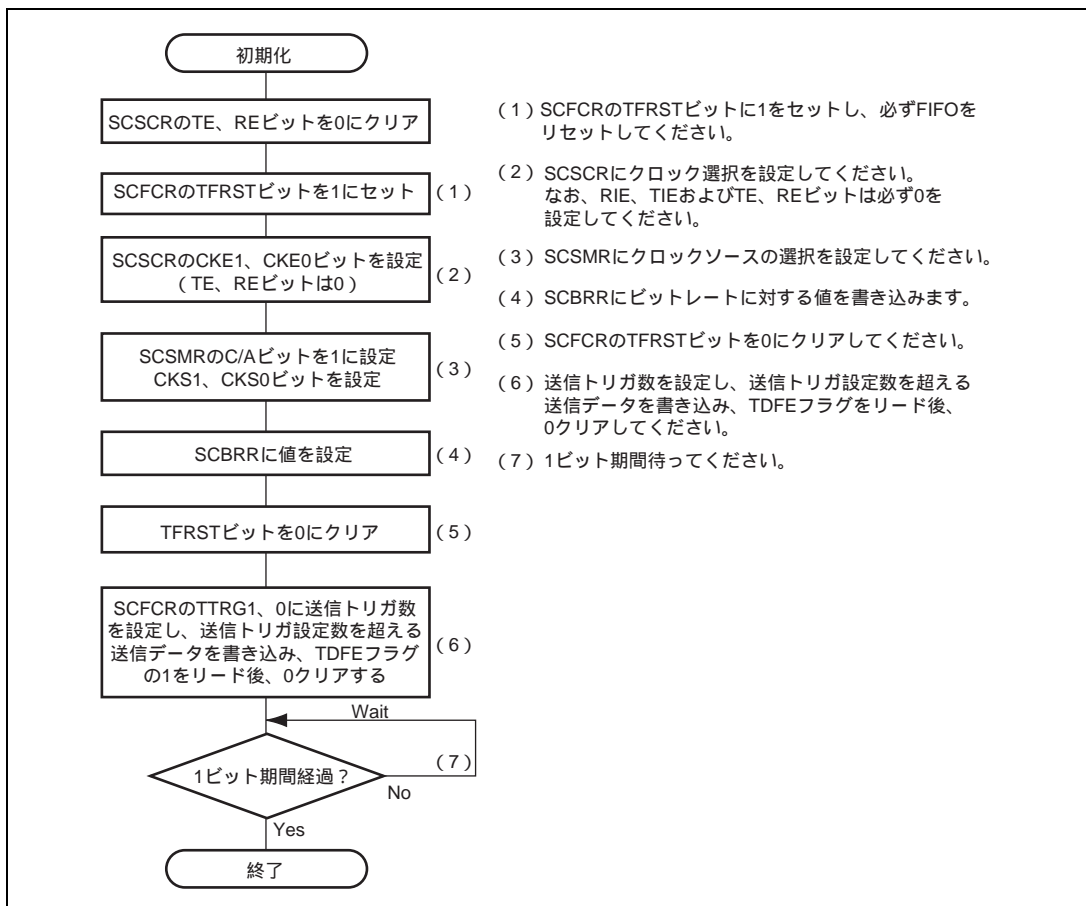


図 16.13 送信動作時の初期化フローチャートの例 (1)

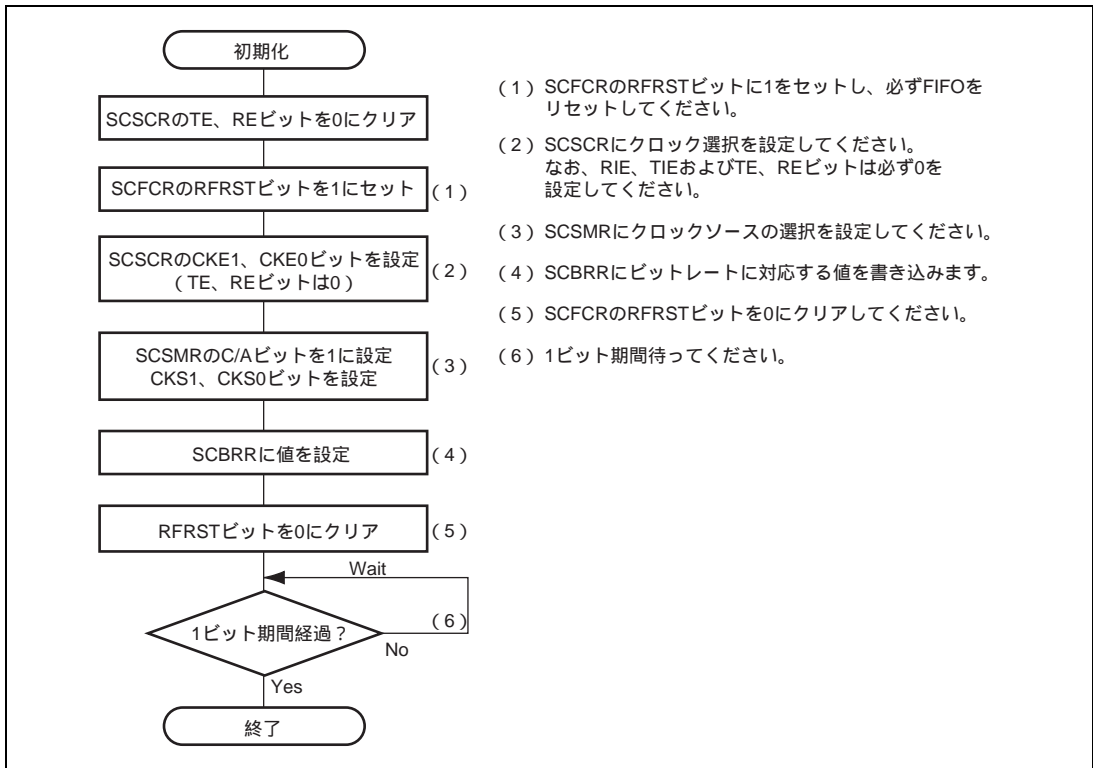


図 16.13 受信動作時の初期化フローチャートの例 (2)

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

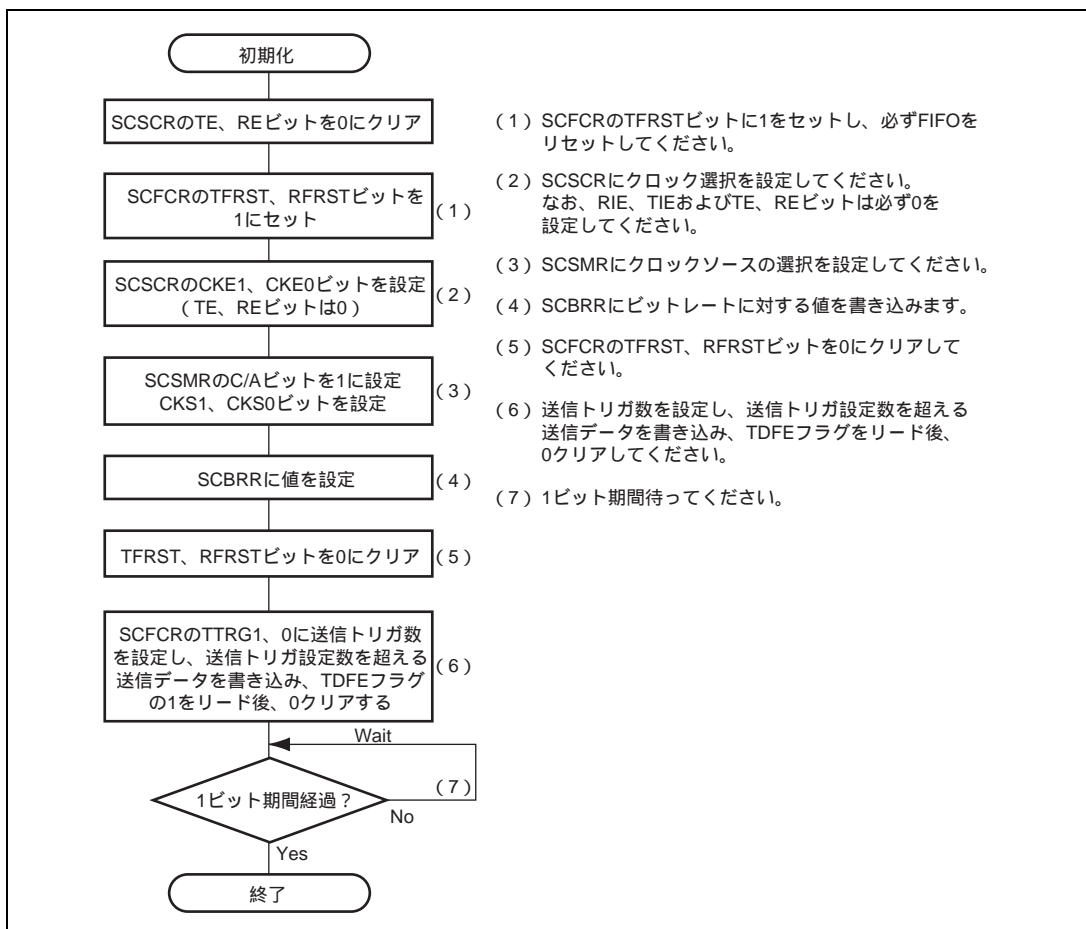


図 16.13 同時送受信時の初期化フローチャートの例 (3)

(b) シリアルデータ送信

図 16.14 にシリアル送信のフローチャートの例を示します。

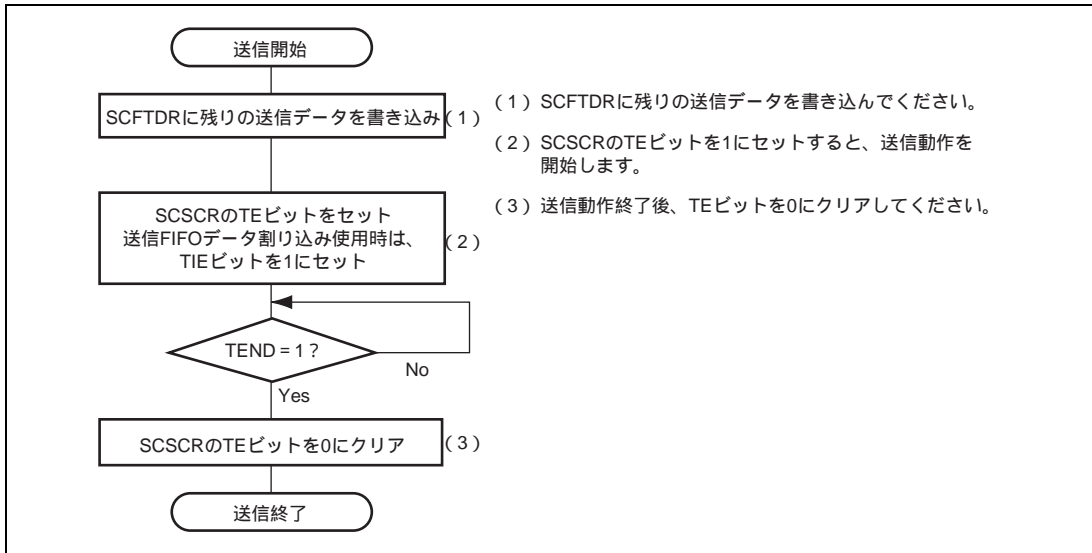


図 16.14 送信動作時のフローチャートの例 (初期化後 1 回目の送信動作) (1)

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

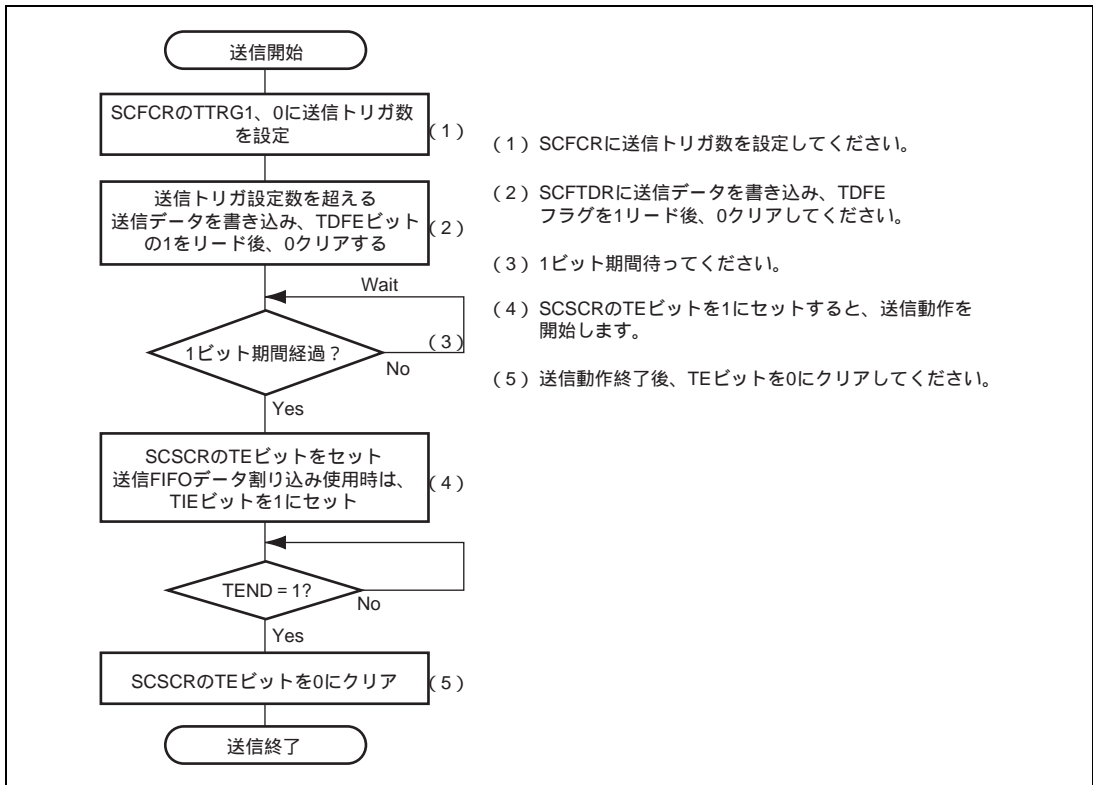


図 16.14 送信動作時のフローチャートの例 (2回目以降の送信動作) (2)

(c) シリアルデータ受信

図 16.15 にシリアル受信フローチャートの例を示します。

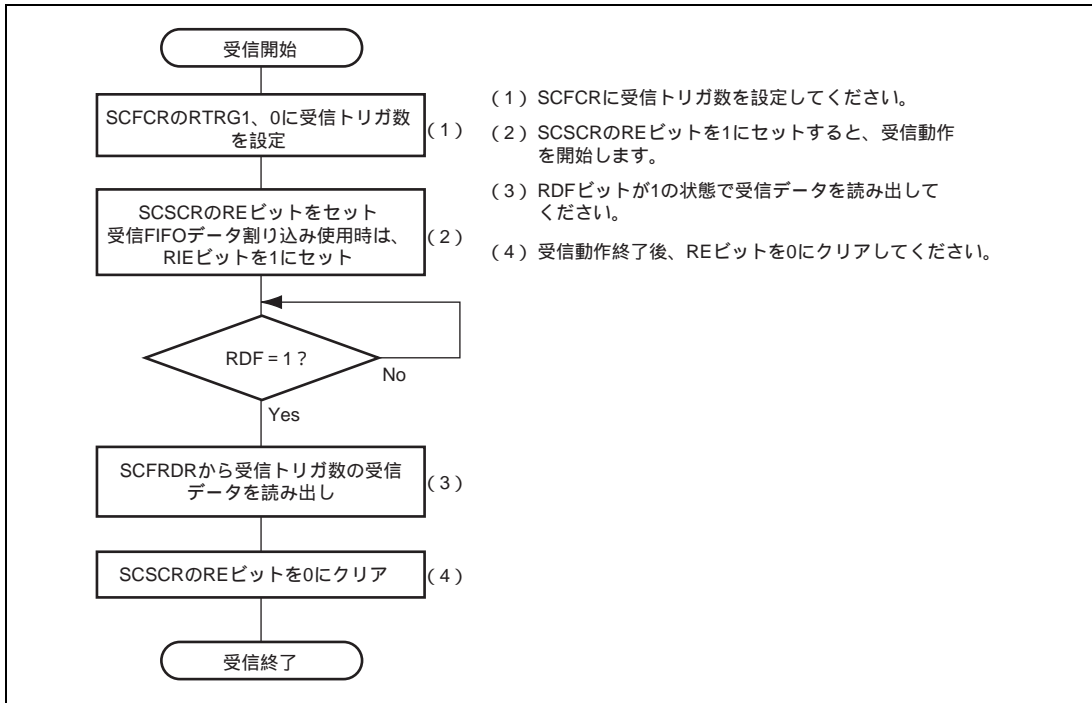


図 16.15 受信動作時のフローチャートの例 (初期化後 1 回目の受信動作) (1)

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

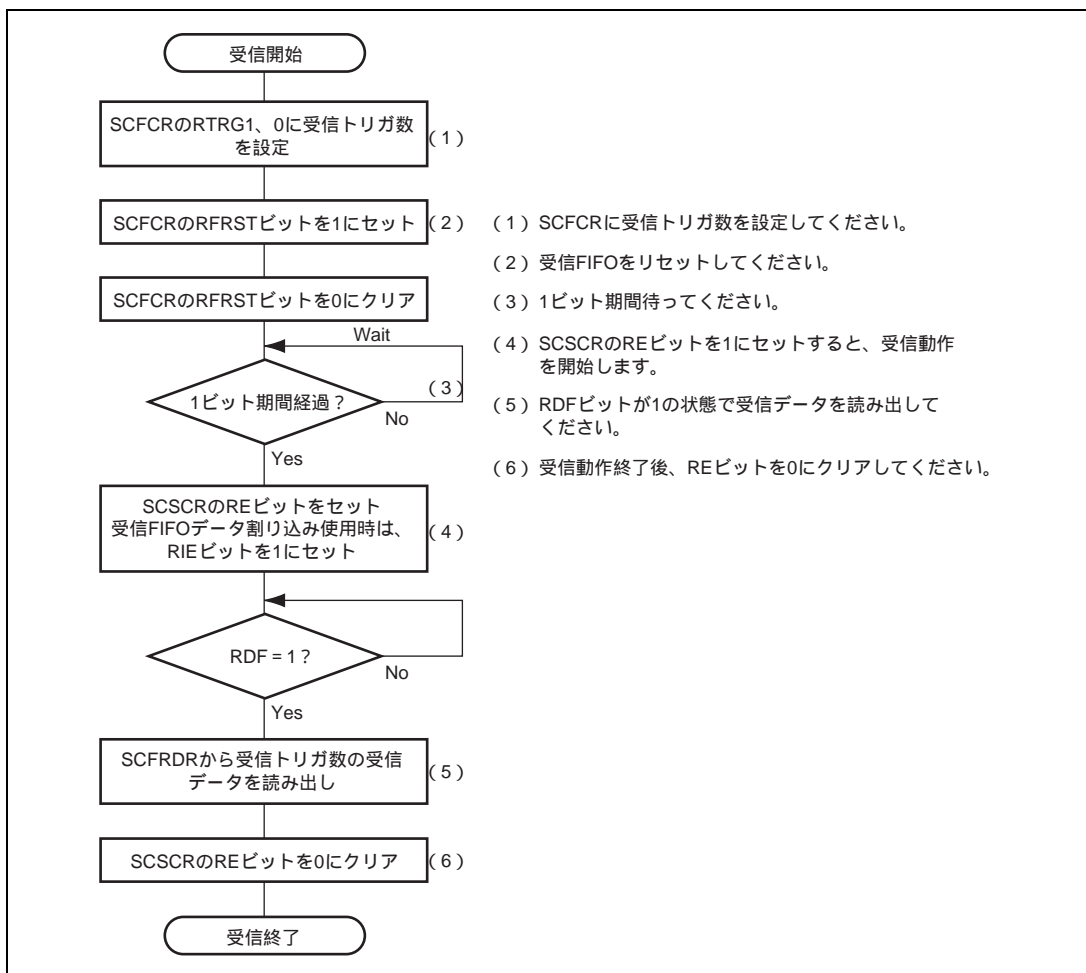


図 16.15 受信動作時のフローチャートの例 (2 回目以降の受信動作) (2)

(d) シリアルデータ送受信同時動作

図 16.16 にシリアル送受信同時動作のフローチャートの例を示します。

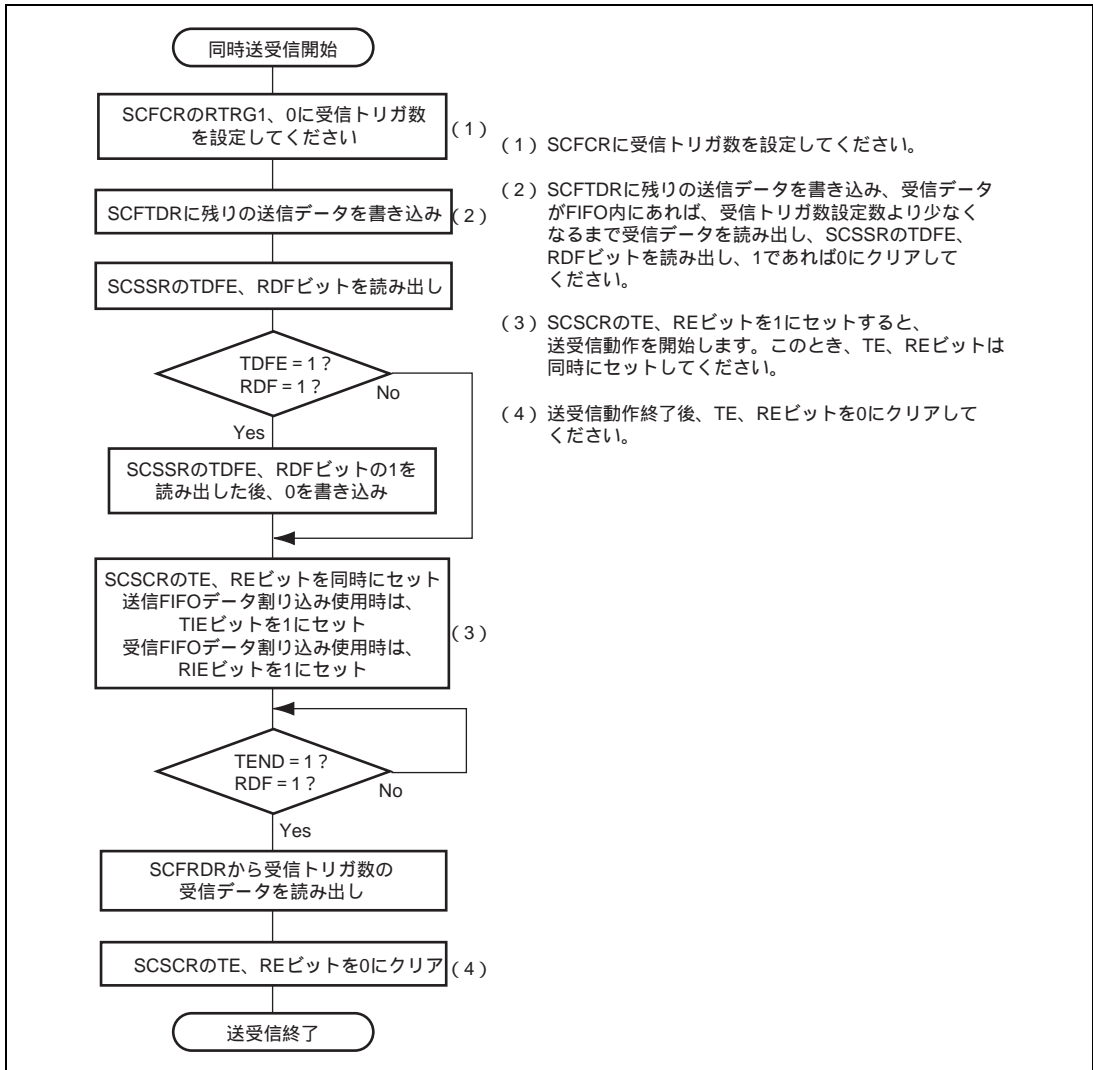


図 16.16 同時送受信動作時のフローチャートの例 (初期化後 1 回目の送受信動作) (1)

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

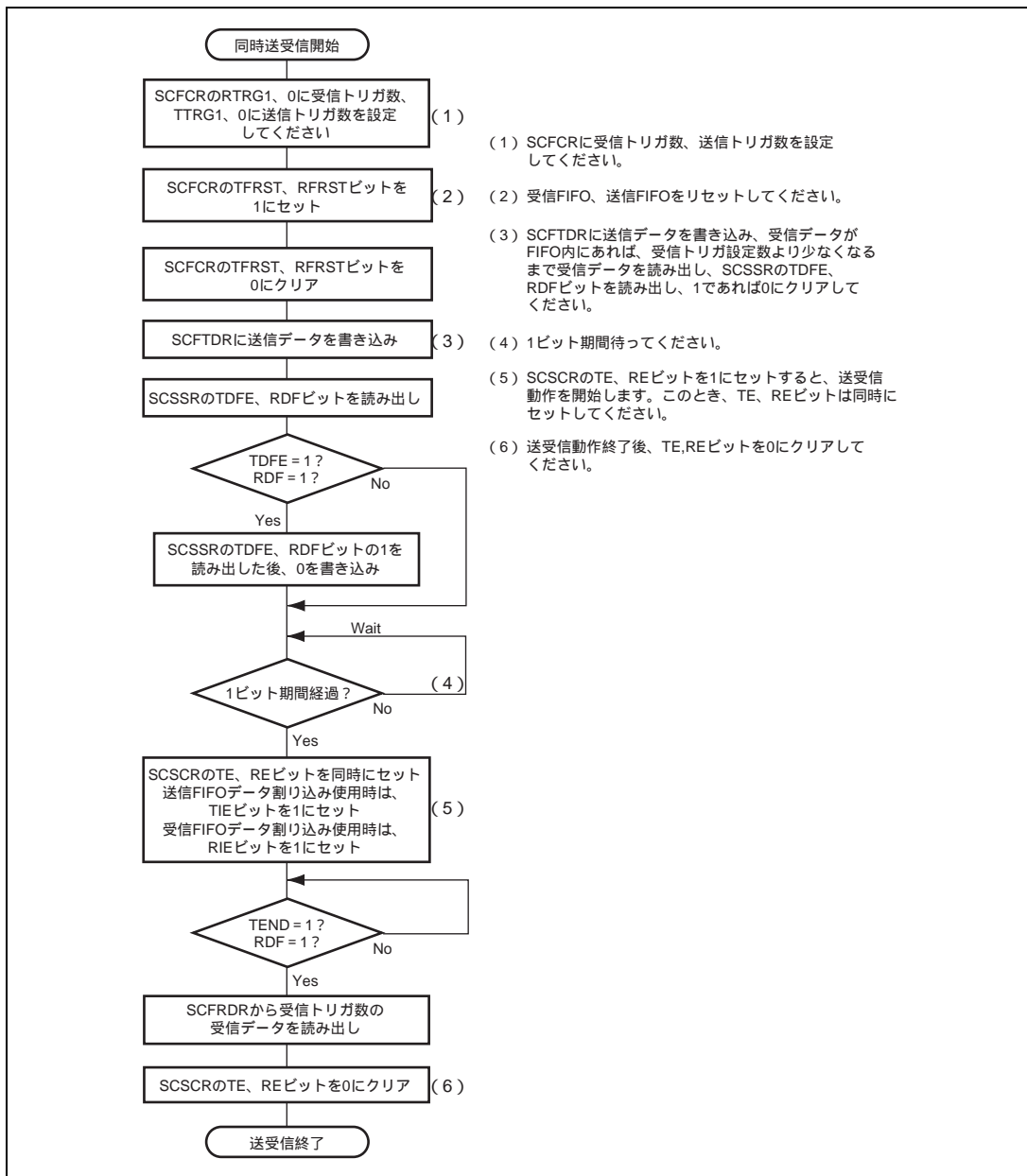


図 16.16 同時送受信動作時のフローチャートの例 (2回目以降の送受信動作) (2)

16.5 割り込み要因と DMAC

SCIF は、調歩同期式モード時には、送信 FIFO データエンプティ要因による割り込み要求、送信データストップ要因による割り込み要求、受信エラー要因による割り込み要求、受信 FIFO データフル要因による割り込み要求、ブレイク受信要因による割り込み要求、レシープデータレディ要因による割り込み要求の 6 種類の割り込み要因をサポートしています。クロック同期式モード時には、送信 FIFO データエンプティ割り込み要求、受信 FIFO データフル割り込み要求の 2 種類の割り込みをサポートしています。ただし、各要因による割り込みのベクタは共通です。

表 16.4 に各割り込み要因を示します。各割り込み要因は、SCSSR の TIE、RIE、ERIE、BRIE、DRIE、TSIE ビットで、許可または禁止ができます。

SCSSR の TDFE フラグが 1 にセットされると、送信 FIFO データエンプティ要因による割り込み要求が発生します。SCSSR の TSF フラグが 1 にセットされると、送信データストップ要因による割り込み要求が発生します。送信 FIFO データエンプティ要因による割り込み要求と送信データストップ要因による割り込み要求で DMAC を起動してデータ転送を行うことができます。ただし、送信 FIFO データエンプティ要因と送信データストップ要因による DMAC 要求は共通です。

SCSSR の RDF フラグが 1 にセットされると、受信 FIFO データフル要因による割り込み要求が発生します。受信 FIFO データフル要因による割り込み要求で DMAC を起動してデータ転送を行うことができます。

DMAC を使って送受信を行う場合は、先に DMAC を設定し、イネーブル状態にしてから SCIF の設定を行ってください。DMAC の設定方法は「第 10 章 ダイレクトメモリアクセスコントローラ (DMAC)」を参照してください。

SCSSR の ER フラグが 1 にセットされた場合、または、
SCSSR の BRK フラグが 1 にセットされた場合、または、
SCSSR の DR フラグが 1 にセットされた場合、または、
SCSSR の TSF フラグが 1 にセットされた場合には、割り込み要求が発生します。

ただし、各要因による割り込みのベクタは共通となります。また、同一要因により、DMAC 起動と割り込みを同時に発生させることはできません。DMAC 起動を使用する場合は、以下の手順をとってください。

1. 発生要因に対応する割り込みイネーブルビット (TIE、RIE) を 1 にセットしてください。
2. その他の割り込みイネーブルビット (TSIE、ERIE、BRIE、DRIE) は 0 にセットしてください。

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

表 16.4 SCIF 割り込み要因

割り込み要因の内容	DMAC の起動
受信エラー (ER) またはブ레이크 (BRK) による割り込み	不可
受信 FIFO データフル (RDF) またはデータレディ (DR) による割り込み	可* ¹
送信 FIFO データエンpty (TDFE) または送信データストップ (TSF) による割り込み	可* ²

【注】 *¹ 受信 FIFO データフル要因による割り込み要求でのみ DMAC の起動が行えます。

- *² 送信 FIFO データエンpty (TDFE) または送信データストップ (TSF) による割り込み要求で DMAC の起動が行えます。送信データストップ (TSF) により DMAC を起動した場合のクリアは以下になります。
- (1) CPU で TSF フラグをリードしたとき。
 - (2) 送信 FIFO がいっぱいになったとき。

優先順位、SCIF 以外の割り込みとの関係は、「第 4 章 例外処理」を参照してください。

16.6 使用上の注意事項

SCIF を使用する際は、以下のことに注意してください。

(1) SCFTDR への書き込みと TDFE フラグについて

シリアルステータスレジスタ (SCSSR) の TDFE フラグはトランスミット FIFO データレジスタ (SCFTDR) 内に書き込んだ送信データ数が、FIFO コントロールレジスタ (SCFCR) の TTRG1、0 ビットで設定した送信トリガ数以下になったときセットされます。TDFE がセットされたあと、SCFTDR の空データ数まで送信データを書き込むことができ、効率よい連続送信が可能となります。

しかし TDFE フラグは SCFTDR に書き込まれているデータ数が送信トリガ数以下の場合には、1 を読み出し後、0 にクリアしても再び 1 にセットされます。TDFE のクリアは送信トリガ数より多い送信データが SCFTDR に格納されたときに行ってください。

SCFTDR 内の送信データ数は FIFO データ数レジスタ (SCFDR) のビット 14~8 で知ることができます。

(2) SCFRDR の読み出しと RDF フラグについて

シリアルステータスレジスタ (SCSSR) の RDF フラグは、レシーブ FIFO データレジスタ (SCFRDR) 内の受信データ数が FIFO コントロールレジスタ (SCFCR) の RTRG1、0 ビットで設定した受信トリガ数以上になったときセットします。RDF がセットされたあと、SCFRDR からトリガ数分の受信データを読み出すことで効率のよい連続受信が可能です。

ただし、読み出し後も SCFRDR 内のデータ数がトリガ数以上の場合、RDF フラグを 0 にクリアしても再び 1 にセットされますので、すべての受信データを読み出したあと、RDF フラグの 1 を読み出し 0 にクリアしてください。

SCFRDR 内の受信データ数は FIFO データ数レジスタ (SCFDR) のビット 6~0 で知ることができます。

(3) ブレークの検出と処理について

フレーミングエラー (FER) 検出時に RXD 端子の値を直接読み出すことによっても、ブレークを検出できます。ブレークでは、RXD 端子からの入力が入力がすべて 0 になりますので、FER フラグがセットされ、またパリティエラー (PER) もセットされる場合があります。

SCIF は、ブレークを受信したあとは、SCFRDR への受信データの転送は停止しますが、受信動作は続けています。

(4) 受信データサンプリングタイミングと受信マージン

サンプリングレートを 1/16 とした場合を例にあげて説明します。SCIF は転送レートの 8 倍の周波数の基本クロックで動作します。

受信時に SCIF は、スタートビットの立ち下がり基本クロックでサンプリングして、内部を同期化します。また、受信データを基本クロックの 8 クロック目の立ち上がりエッジで内部に取り込みます。

これを図 16.17 に示します。

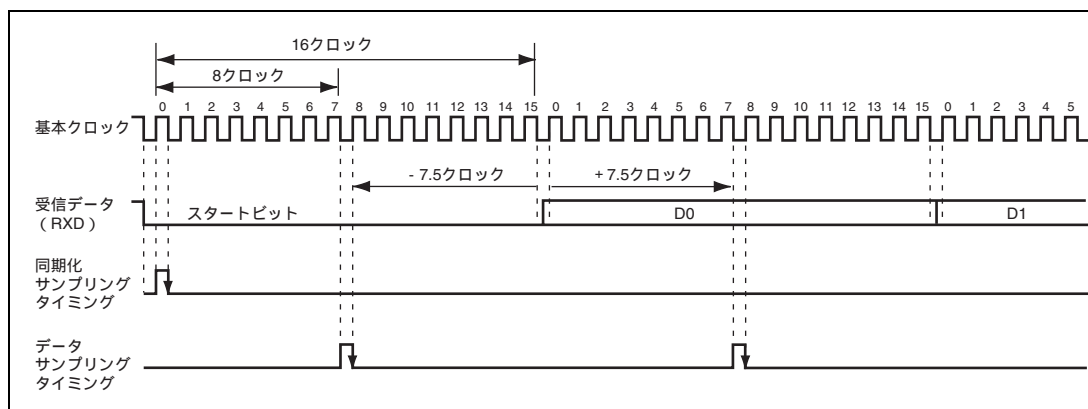


図 16.17 調歩同期式モードの受信データサンプリングタイミング

したがって、調歩同期式モードでの受信マージンは式 (1) のように表わすことができます。

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N} (1+F) \right| \times 100\% \cdots \cdots \text{式 (1)}$$

M : 受信マージン (%)

N : クロックに対するビットレートの比 (N = 16)

D : クロックデューティ (D = 0 ~ 1.0)

L : フレーム長 (L = 9 ~ 12)

F : クロック周波数の偏差の絶対値

16. FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)

式 (1) で、 $F=0$ 、 $D=0.5$ とすると、受信マージンは式 (2) より 46.875% となります。

$D=0.5$ 、 $F=0$ のとき

$$M = (0.5 - 1 / (2 \times 16)) \times 100\%$$

$$= 46.875\% \quad \dots \dots \dots \text{式 (2)}$$

ただし、この値はあくまでも計算上の値ですので、システム設計の際には 20 ~ 30% の余裕を持たせてください。

(5) SCIF の初期化

通信フォーマットの変更などの場合には必ず、TE ビットおよび RE ビットを 0 にクリアしてから次の手順で変更を行ってください。TE ビットを 0 にクリアするとトランスミットシフトレジスタ (SCTSR) が、RE ビットを 0 にクリアするとレシーブシフトレジスタ (SCRSR) が初期化されます。TE、RE ビットを 0 にクリアしても、シリアルステータスレジスタ (SCSSR)、トランスミット FIFO データレジスタ (SCFTDR) および、レシーブ FIFO データレジスタ (SCFRDR) の内容は保持されますので注意してください。TE ビットの 0 クリアは、送信データをすべて送信し SCSSR の TEND ビットが 1 にセットされた後に行ってください。送信中でも 0 クリア可能ですが、送信中のデータは 0 クリア後、ハイインピーダンス状態になります。また再度 TE ビットを 1 にセットして送信開始する前に SCFCR の TFRST ビットを 1 にセットして SCFTDR をリセットしてください。

外部クロックを使用している場合には、動作が不確実になりますので初期化を含めた動作中にクロックを止めないでください。

(6) 低消費電力モードへの移行

シリアル送受信動作中にソフトウェアスタンバイモードに入ったり、SCIF のモジュールスタンバイビットをセットしてモジュールスタンバイ状態にしないでください。ソフトウェアスタンバイモードに移行するときや、モジュールスタンバイ状態にするときは、必ずシリアルコントロールレジスタ (SCSCR) の TE、RE ビットを共にクリア (送受信ディスエーブル状態) した後に移行してください。

(7) 送受信ディスエーブル時の送受信端子状態

SCIF モジュールの TXD 端子は、TE ビットがクリアされているときはハイインピーダンス状態になります。RXD 端子は RE ビットがクリアされているときは入力固定状態になります。TXD 端子に接続する信号線の処理には注意が必要です。接続する先が常時入力状態の場合は、信号線にプルアップ素子を接続してください。

17. ブート機能 (BOOT)

本 LSI は、外付けフラッシュメモリへの初期書き込みに必要なファームウェアをオンチップメモリにダウンロードするブート機能 (BOOT) を備えています。本機能を使うことにより、フラッシュメモリを基板に実装後でも本 LSI を通してプログラムを更新したり、内容をダンプしたりすることが可能となります。また、本 LSI とフラッシュメモリその他をベアチップで 1 つのパッケージに封止するときにも、フラッシュメモリへの書き込みのためにアドレスバスやデータバスをパッケージの外へ出す必要がなく、ピン数を大幅に節約できます。

17.1 特長

- SCIF0からの初期書き込みプログラム自動取り込み

パワーオンリセット後、まず外付けフラッシュメモリへの初期書き込みプログラムをSCIF0から内蔵メモリ (Uメモリ) に取り込む動作を開始します。転送方式はUARTによる調歩同期方式です。これにより、接続する外付けフラッシュメモリの種類に応じて任意の初期書き込みプログラムをダウンロードすることが可能です。ダウンロードするプログラムは、本章で定められている仕様に準拠すれば、外付けフラッシュメモリへの初期書き込み以外のプログラムにすることも可能です。取り込むプログラムのサイズは最大8,192バイトまでの範囲で自由に決められます。

- ビット転送レート自動合せ込み

SCIF0から取り込むプログラムのビット転送レートを自動的に合せ込む機能を備えています。

- 取り込んだプログラムは内蔵メモリ上で高速実行

SCIF0から取り込んだプログラムは内蔵メモリ上に展開されるため、高速に実行可能です。

- プログラムのダウンロード終了後、自動的にダウンロードしたプログラムを実行開始

所定のプログラムをダウンロード後、ダウンロードしたプログラムの先頭番地 (H'A55F0000) に分岐し、自動的に実行を開始します。

【注】 BOOT 機能は 1 つの回路モジュール構成となっており、モジュールスタンバイ機能によるパワー制御の対象モジュールの 1 つになっています。BOOT 機能は、ブート処理時以外は不要です。通常の動作状態やスリープ状態での消費電流を低減したい場合は、スタンバイコントロールレジスタ 4 (STBCR4) の MSTP43 ビットを 1 にセットすることにより、BOOT モジュールへのクロック供給を停止状態に設定することをお勧めします。モジュールスタンバイ機能の詳細については、「第 13 章 低消費電力モード」を参照してください。

図 17.1 にブート機能 (BOOT) のブロック図を示します。バスステートコントローラ (BSC) の入出力端子 (A23 ~ A0、D15 ~ D0、 $\overline{WE1}$ ~ $\overline{WE0}$ 、 $\overline{CS0}$ 、 \overline{RD}) を外付けフラッシュメモリに接続し、SCIF0 の入出力端子 (SCIF0_RXD、SCIF0_TXD) を外部ホストに接続します。

17. ブート機能 (BOOT)

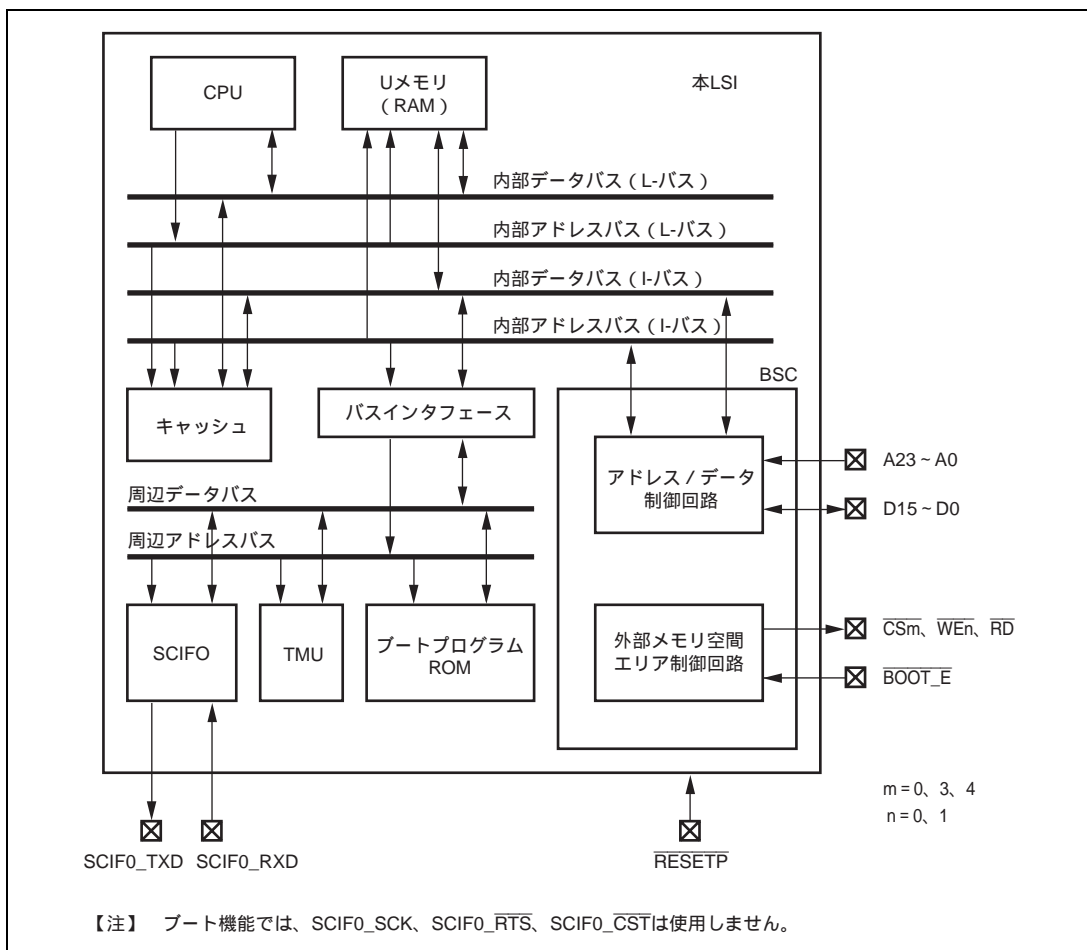


図 17.1 BOOT 機能のブロック図

17.2 入出力端子

ブート機能 (BOOT) 専用の端子を表 17.1 に示します。外付けフラッシュメモリを接続する端子は「第 9 章 バスステートコントローラ (BSC)」、SCIF0 に関する端子は「第 16 章 FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF)」、パワーオンリセット端子 ($\overline{\text{RESETP}}$) に関する詳細は「第 4 章 例外処理」を参照してください。

表 17.1 端子構成

端子名	入出力	機能
BOOT_E	入力	ブートイネーブル入力 リセット中のみ切り替え可能です。

17.3 レジスタの説明

ブート機能 (BOOT) には、CPU からアクセス可能なレジスタはありません。

17.4 動作説明

17.4.1 ブートモードのアドレス空間

パワーオンリセット ($\overline{\text{RESETP}}$ 端子 = ロー) 時、 $\overline{\text{BOOT_E}}$ 端子をローにアサートすることにより、リセット終了後、本 LSI はブートモードに入ります。ブートモードでは、本 LSI は通常のリセットベクタへの分岐処理は開始されず、ブート機能が立ち上がり、本 LSI に内蔵されているブートプログラムが起動します。

リセットによる初期状態のため、CPU は特権モードとなっています。特権モードについての詳細は、「2.1 処理状態と処理モード」を参照してください。ブートモードでは通常の特権モードと異なり、外部アドレス空間のエリア 0 領域が図 17.2 に示すように 2 つの領域に分割され、H'00000000 ~ H'01FFFFFF はブート機能の専用領域となり、ユーザによるアクセスは禁止状態になります。ブートプログラムはここに配置されています。H'02000000 ~ H'03FFFFFF は通常モードの場合と同様にエリア 0 領域となります。

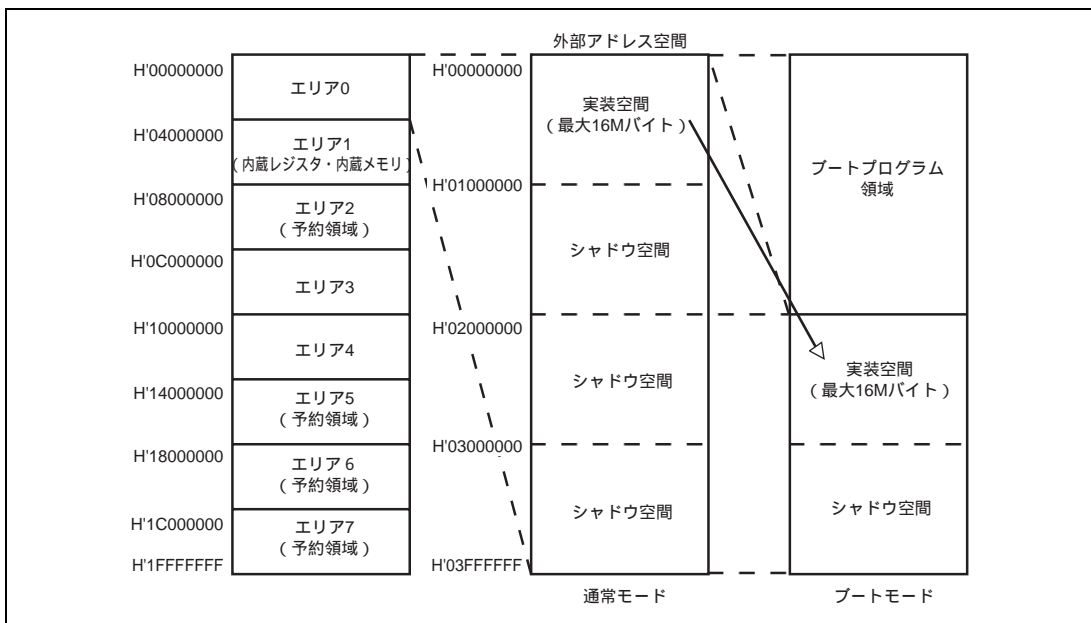


図 17.2 ブートモードでの外部アドレス空間

外付けフラッシュメモリに初期書き込みを行う場合、パワーオンリセット解除後リセットベクタが必ず H'A0000000 (CPU から見える論理アドレス) に分岐し、外部アドレス空間のエリア 0 にアクセスしますので、初期書き込みプログラムはエリア 0 の先頭アドレスから書き込む必要がありますが、上記のとおり、ブートモードでは H'00000000 ~ H'01FFFFFF はブート機能の専用領域となっていますので、H'02000000 を先頭アドレスとして

17. ブート機能 (BOOT)

書き込みを行います。本 LSI は実装可能な領域が最大 16M バイトであり、H'02000000 ~ H'02FFFFFF 領域は H'00000000 ~ H'00FFFFFF 領域のシャドウとなるため、H'02000000 から書き込むことにより、ブートモード解除後の通常モードでは H'00000000 番地からアクセス可能となります。

【注】 * BOOT_E 端子入力はリセット期間中以外は変化させないでください。リセット期間以外で入力を変更した場合は、動作の保証はできません。

17.4.2 ブート処理の実行手順

ブート機能を使用するためには、外付けフラッシュメモリへの初期書き込みプログラムをあらかじめ外部ホスト側に準備する必要があります。初期書き込みプログラムは、使用する外付けフラッシュメモリの書き込みアルゴリズムに沿ったプログラムを準備してください。

ブートプログラムは SCIF0 を使った外部ホストとのデータ送受信の準備を開始します。SCIF0 の必要な初期設定終了後、自動的に外部ホストと最適な送受信ビットレートに調整します。ブート処理の実行手順を図 17.3 に示します。

(1) SCIF0 ビット転送レートの自動合せ込み動作

本 LSI は、9.6kbps から 38.4kbps の間で最適なビット転送レートを自動的に合わせ込みます。ブートモードで本 LSI が起動すると、本 LSI は外部ホストから連続送信される調歩同期式のデータ (H'00) のロー期間を測定します。ロー期間の測定にはタイマユニット (TMU) のチャンネル 0 を使用しています。このとき外部ホストは本 LSI の SCIF0 との受信 / 送信フォーマットを「8 ビットデータ、1 ストップビット、パリティ無し」に設定してください。ブートプログラムは測定したロー期間から外部ホストの送信するビットレートを算出し、最適な設定を行った後、ビットレート調整終了を通知するために外部ホストへ H'00 を 1 バイト送信します。外部ホストはこのビットレート調整終了通知を正常に受信したら、本 LSI へ H'55 を 1 バイト送信してください。本 LSI は H'55 を受信すると、H'AA を 1 バイト送信します。

【注】 受信が正常に行われなかった場合は無限ループに入り、反応しなくなります。再度ブートモードをリセットしてから起動し、上述の操作を行ってください。

(2) 初期書き込みプログラムの転送動作

ビット転送レートの自動合わせ込み動作を終了すると、ブートプログラムは、初期書き込みプログラムのダウンロード動作を開始します。

SCIF0 からダウンロードする初期書き込みプログラムのサイズは、最大 8,192 バイト (H'2000) まで対応可能です*¹。ダウンロードした初期書き込みプログラムは、内蔵メモリ (Uメモリ) 上の H'055F0000 ~ H'055F1FFF (物理アドレス) に格納されます。ブートプログラムはダウンロード動作終了後、自動的に H'055F0000 に分岐する仕様になっています*²。

ブートプログラムは H'055FFFC0 ~ H'055FFFFF (物理アドレス) をワークエリアとして使用していますが、ダウンロード動作終了後は、ダウンロードした初期書き込みプログラムによって使用することは可能です。

【注】 *¹ 初期書き込みプログラムのサイズを H'0000 または H'2001 以上の値に設定すると、正常動作できなくなりますので、設定しないでください。設定した場合、動作の保証はできません。

*² ブートモード時に初期書き込みプログラムで書き込むユーザプログラムの初期設定ルーチンは、必ずエリア 0 の H'02000000 から格納するようにしてください。これにより、通常モード時、パワーオンリセット直後にまずエリア 0 の先頭番地 (H'00000000) に分岐するので、初期設定が可能になります。

エリア 0 でユーザプログラムが実装可能な領域は、図 17.2 に示しているように、H'02000000 ~ H'02FFFFFF (通常モード時の H'00000000 ~ H'00FFFFFF) の最大 16M バイトになります。

本 LSI はエリア 0 の他にエリア 3、エリア 4 にもそれぞれ最大 16M バイトの外部メモリの実装が可能です。対応した初期書き込みプログラムを準備することにより、必要に応じてエリア 3、エリア 4 にも外付けフラッシュメモリを拡張できます。

17. ブート機能 (BOOT)

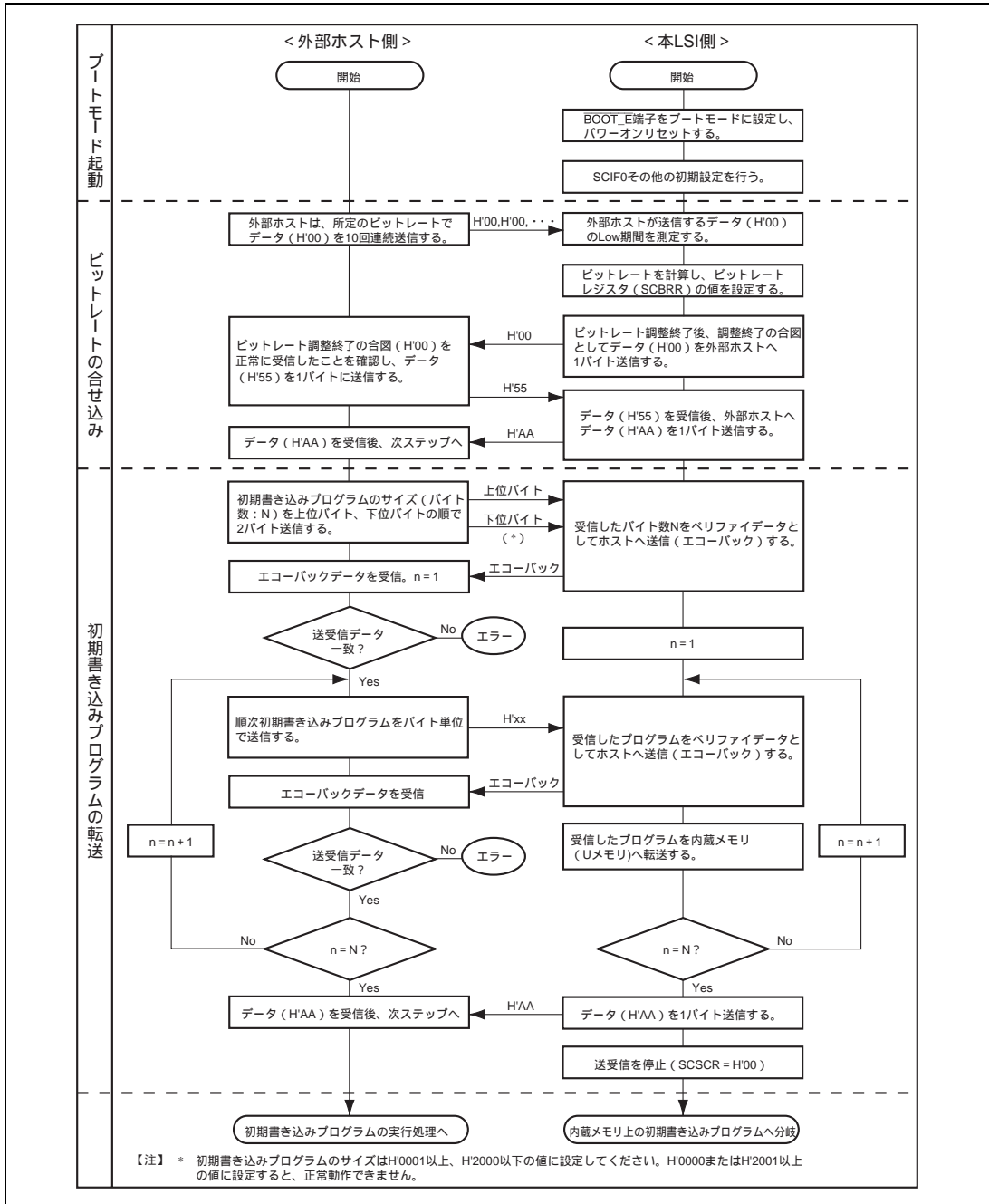


図 17.3 ブート処理の実行手順

17.5 使用上の注意事項

(1) ブート機能使用時のエンディアンについて

ブート機能はブートROM内に格納されているファームウェアによって動作しますが、ビッグエンディアンで動作しますので、パワーオンリセット($\overline{\text{RESETP}}$ 端子 = ロー)時、 $\overline{\text{BOOT_E}}$ 端子をローにアサートするときはMD5端子を必ずロー(ビッグエンディアンモード)にしてください。ダウンロード終了後、ブート機能は内蔵メモリ(Uメモリ)にダウンロードされた外付けフラッシュメモリへの初期書き込みプログラムの先頭番地に分岐しますので、初期書き込みプログラムもビッグエンディアンで動作するようにしておく必要があります。通常モードでの動作をリトルエンディアンで実行する場合は、フラッシュメモリへの初期書き込み時にコードのアライメントに注意してください。

(2) ブート機能使用時のクロック周波数と SCIF の転送レートについて

ブート機能使用時に外部ホストとの転送を行うSCIFの転送レートは、表17.2に示すような制限があります。外部ホスト転送用プログラム(たとえばフラッシュメモリなど)での転送レート設定には注意してください。

表 17.2 ブート機能使用時のクロック周波数と SCIF の転送レート

CPG 設定	入力クロック周波数	SCIF 転送レート
モード5 クロック比 I : B : P = 2:2:1	10MHz 以上	9600bps
	13MHz 以上	19200bps
	16MHz 以上	38400bps
モード7 クロック比 I : B : P = 1:1:1/2	20MHz 以上	9600bps
	26MHz 以上	19200bps
	32MHz 以上	38400bps

17. ブート機能 (BOOT)

18. USB ピンマルチプレクスコントローラ (USBPM)

18.1 特長

USB ピンマルチプレクスコントローラは USB ホストコントローラ、または USB ファンクションコントローラから USB トランシーバへのデータパスを制御します。さらに、本 LSI の USB ピンマルチプレクスコントローラは、USB ホストコントローラおよび USB ファンクションコントローラに供給するクロック信号を内部生成または外部から入力する選択制御機能を持っています。

USB ホストコントローラと USB ファンクションコントローラは、EXCPG 制御レジスタ (EXCPGCR) で制御されるマルチプレクサを介して、USB トランシーバに接続されています。そのため、USB トランシーバは USB ホストコントローラまたは USB ファンクションコントローラのいずれか一方に使用されることとなります。

本 LSI は USB2.0 規格で規定された 48MHz のクロックを RDI_REFCLK_IN 端子より入力される 13MHz クロックから内蔵通倍回路によって生成し、USB ホストコントローラおよび USB ファンクションコントローラに供給します。外部に 48MHz のクロック供給源がある場合、外部から直接供給することも可能です。前者の内蔵通倍回路を使う機能は、本 LSI に直結される RF IC 等から安定したクロックを受け取る場合に、新たな 48MHz の TCXO を外部に接続せずに済む利点がありますが、一定のジッタが発生しますので、その点の考慮が必要です。内蔵通倍回路からのクロックを使った場合、他の USB 機器との接続性の確認を行ってから使用してください。

18. USB ピンマルチプレクスコントローラ (USBPM)

図 18.1 に、本 LSI に内蔵された USB ホストコントローラ、USB ファンクションコントローラ、内蔵された 1 ポートのアナログ USB トランシーバおよびクロック制御回路の接続を示します。

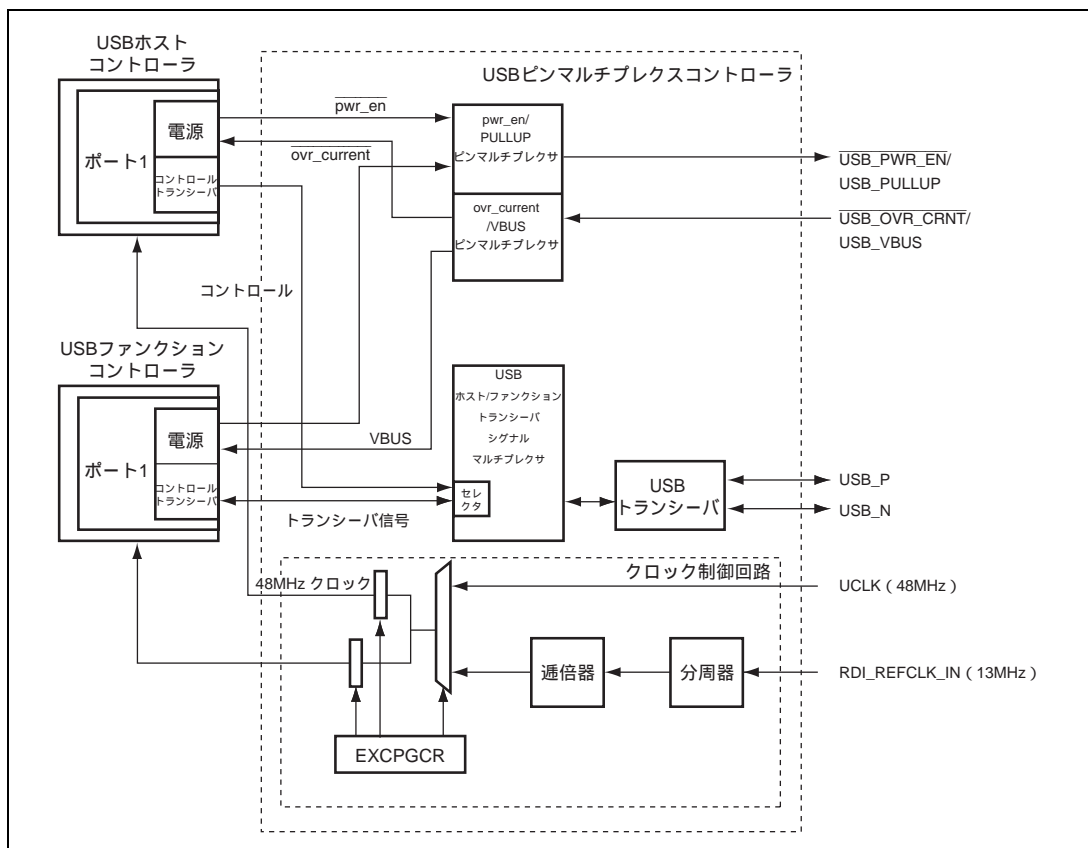


図 18.1 USB 機能のブロック図

18.2 入出力端子

USB ピンマルチプレクスコントローラには表 18.1、表 18.2、表 18.3 に示すような端子があります。

表 18.1 端子構成 (アナログトランシーバ信号)

端子名	入出力	機能
USB_P	入出力	P 端子 D+ポートトランシーバ端子
USB_N	入出力	N 端子 D-ポートトランシーバ端子

【注】 表 18.1 に示された端子は、1 ポートの USB ホストコントローラ端子、または 1 ポートの USB ファンクションコントローラ端子として使用することが可能です。未使用時はプルダウンしてください。

18. USB ピンマルチプレクスコントローラ (USBPM)

表 18.2 端子構成 (電源制御信号)

端子名	入出力	機能
USB_PWR_EN/ USB_PULLUP	出力	パワーストア端子 / ブルアップ制御端子 電源投入許可制御 (USBH 用) / USB ブルアップ制御 (USBF 用) *
USB_OVR_CRNT/ USB_VBUS	入力	オーバーカレント端子 / VBUS 端子 オーバーカレント検出 (USBH 用) / USB ケーブル接続モニタ (USBF 用) *

【注】 表 18.2 に示された端子は、USB の電源制御のために使用されます。(*印)の端子は USB ホストコントローラ (USBH)、および USB ファンクションコントローラ (USBF) の機能がマルチプレクスされています。

表 18.3 端子構成 (クロック制御信号)

端子名	入出力	機能
RDI_REFCLK_IN	入力	ブルートゥースインタフェース用クロック端子 ブルートゥースインタフェース用クロック (13MHz) を入力するための端子 (内蔵逡倍回路によって 48MHz を生成します。)
UCLK	入力	外部クロック端子 外部から直接 48MHz クロックを入力するための端子

【注】 USBH/USBF の 48MHz クロックは、RDI_REFCLK_IN または UCLK のどちらかを選択して使用します。

18.3 レジスタの説明

本モジュールには以下のレジスタがあります。このレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- EXCPG制御レジスタ (EXCPGCR)

18.3.1 EXCPG 制御レジスタ (EXCPGCR)

EXCPGCR は、USB ホストコントローラ、USB ファンクションコントローラへの 48MHz クロックを供給するための制御およびモジュールリセットを行うためのレジスタです。

ビット	ビット名	初期値	R/W	説明
7~5	-	すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

18. USB ピンマルチプレクスコントローラ (USBPM)

ビット	ビット名	初期値	R/W	説明
4	USBVALID	0	R	<p>USB ステータス</p> <p>USB モジュール (ホスト/ファンクションコントローラ共) が使用可能かどうかを示すステータスビットです。読み出しのみ可能です。書き込みアクセスは無視されます。USB モジュールは、パワーオンリセット直後、ソフトウェアスタンバイからの復帰直後、および USBRESET ビットによるリセット直後は、しばらく使用できない状態が発生します。USB モジュールを使用するときには、このビットが使用可能状態であることを確認の後、使用してください。</p> <p>【注】 USBHSTP ビットや USBFSTP ビットがセットされていても内蔵通信器は停止しませんので、USBVALID ビットは 1 の状態を維持します。</p> <p>0 : USB モジュールは使用不可 1 : USB モジュールは使用可能</p>
3	USBRESET	1	R/W	<p>USB リセット</p> <p>USB モジュールのリセットビットです。</p> <p>USB モジュール (ホスト/ファンクションコントローラ共) をリセットするために使用します。このビットに 0 を書き込むと、約 1 μs 後に USBVALID ビットが 0 になります。USBVALID ビットが 0 になったことを確認した後、このビットを 1 にセットしてください。USB モジュールはこのビットが 1 にセットされた後、リセットシーケンスを開始し、終了後、USBVALID ビットが 1 になります。この手順に従わなかった場合、USB モジュールの正常な初期化動作は保証できません。</p> <p>USBHSTP ビットおよび USBFSTP ビットを使用して、USB モジュールへのクロック供給を停止した後に再度クロック供給を行う際、USB モジュールは不安定な状態になる場合がありますので、USB モジュールを使用するときは必ず、本ビットでリセットしてから使用してください。</p> <p>0 : USB モジュールをリセット 1 : 通常状態</p>
2	USBHSTP	0	R/W	<p>モジュールストップ USBH</p> <p>USB ホストコントローラのストップビットです。</p> <p>USB ホストコントローラを動作させるクロックの供給/停止を制御するために使用します。本ビットに 0 を書き込むと、USB ホストコントローラにクロックが供給され、1 を書き込むと、クロック供給が停止します。</p> <p>USBHSTP ビットと USBFSTP ビットが共に 0 または共に 1 の場合、USB トランシーバは非活性状態となり、USB_P、USB_N 出力端子はハイインピーダンスとなり、<u>USB_PWR_EN</u>/ USB_PULLUP 端子はロー出力になります。USB ホストまたはファンクション機能を使うときは、最初に必ず USBHSTP ビットと USBFSTP ビットのどちらか一方のみを 0 に設定してください。USB ホスト機能と USB ファンクション機能を同時に使うことは出来ません。USBHSTP ビットと USBFSTP ビットを共に 0 にしたまま、モジュールの制御レジスタをアクセスした場合の動作は保証できません。</p> <p>USB ホストコントローラへのクロック供給が停止している期間内の USB ホストコントローラへのレジスタアクセスは無視されます。</p> <p>0 : USB ホストコントローラは動作 1 : USB ホストコントローラへのクロック供給を停止</p>

18. USB ピンマルチプレクスコントローラ (USBPM)

ビット	ビット名	初期値	R/W	説明
1	USBFSTP	0	R/W	<p>モジュールストップ USBF</p> <p>USB ファンクションコントローラのストップビットです。</p> <p>USB ファンクションコントローラを動作させるクロックの供給 / 停止を制御するために使用します。このビットに 0 を書き込むと、USB ファンクションコントローラにクロックが供給され、1 を書き込むと、クロック供給が停止します。USBHSTP ビットと USBFSTP ビットが共に 0 または共に 1 の場合、USB トランシーバは非活性状態となり、USB_P、USB_N 出力端子はハイインピーダンスとなり、USB_PWR_EN/USB_PULLUP 端子はロー出力になります。USB ホストまたはファンクション機能を使うときは、最初に必ず USBHSTP ビットと USBFSTP ビットのどちらか一方のみを 0 に設定してください。USB ホスト機能と USB ファンクション機能を同時に使うことはできません。USBHSTP ビットと USBFSTP ビットを共に 0 にしたまま、モジュールの制御レジスタをアクセスした場合の動作は保証できません。</p> <p>USB ファンクションコントローラへのクロック供給が停止している期間内の USB ファンクションコントローラへのレジスタアクセスは行わないでください。行った際の動作は保証できません。</p> <p>0 : USB ファンクションコントローラは動作 1 : USB ファンクションコントローラへのクロック供給を停止</p>
0	USBCLKSEL	0	R/W	<p>USB クロック選択</p> <p>USB 用クロックのソース選択ビットです。</p> <p>USB クロックを内部で生成するか、外部から入力するかを選択するために使用します。このビットを切り替えた場合は、必ず USBRESET ビットを使って USB ホスト / ファンクションコントローラ両モジュールの初期化を行ってください。</p> <p>0 : USB クロックは内部で生成 1 : USB クロックは UCLK 端子から供給</p>

18.4 外部回路例

18.4.1 USB ファンクションと外部回路の接続例

図 18.2 ~ 図 18.3 に USB ファンクションコントローラを選択した場合の本 LSI と外部回路との接続例を示します。USB ファンクションコントローラの使用に際しては、USB_VBUS 端子に信号を入力する必要があります。USB_VBUS 端子は USB ホストコントローラの $\overline{\text{USB_OVR_CRNT}}$ 端子とピンマルチプレクスされており、EXCPGCR レジスタの USBFSTP ビットに 0、USBHSTP ビットに 1 を書き込むことにより、USB_VBUS 端子の機能が選択されます。USB_VBUS 端子の状態により、USB ファンクションコントローラはケーブルの接続 / 切断を認識します。また、USB ホストコントローラ / ハブに対して接続されたことを通知するために、D+ 端子のプルアップを行う必要があります。図 18.2 ~ 図 18.3 の回路例では、プルアップの制御に USB_PULLUP 端子を使用しています。

18. USB ピンマルチプレクスコントローラ (USBPM)

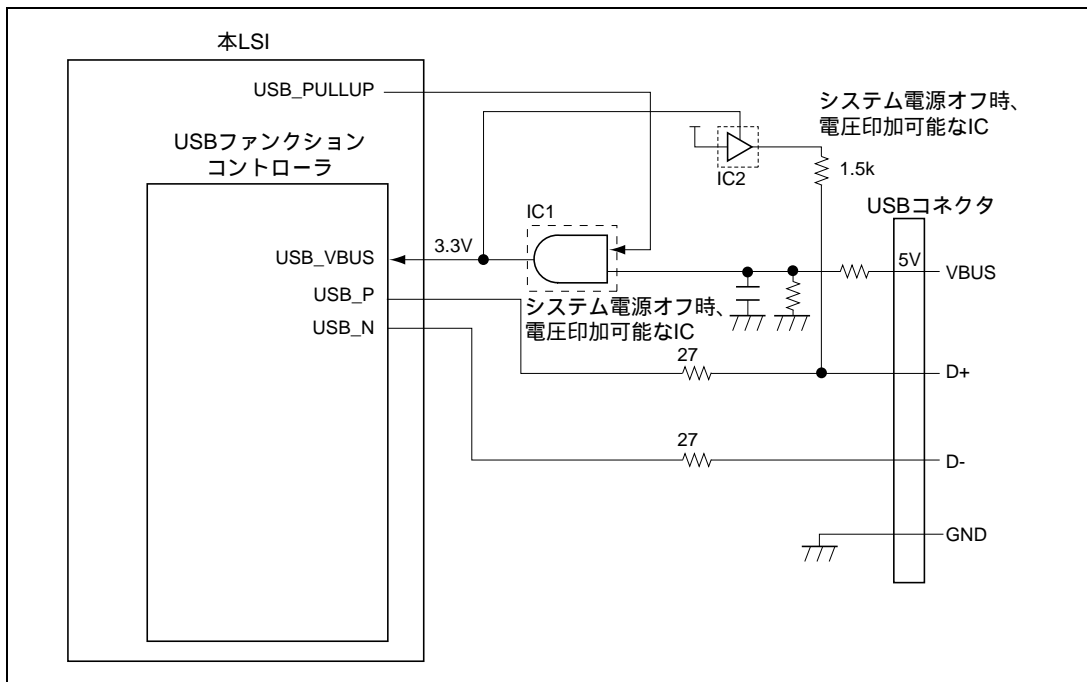


図 18.2 USB ファンクションコントローラと外部回路の接続例 1

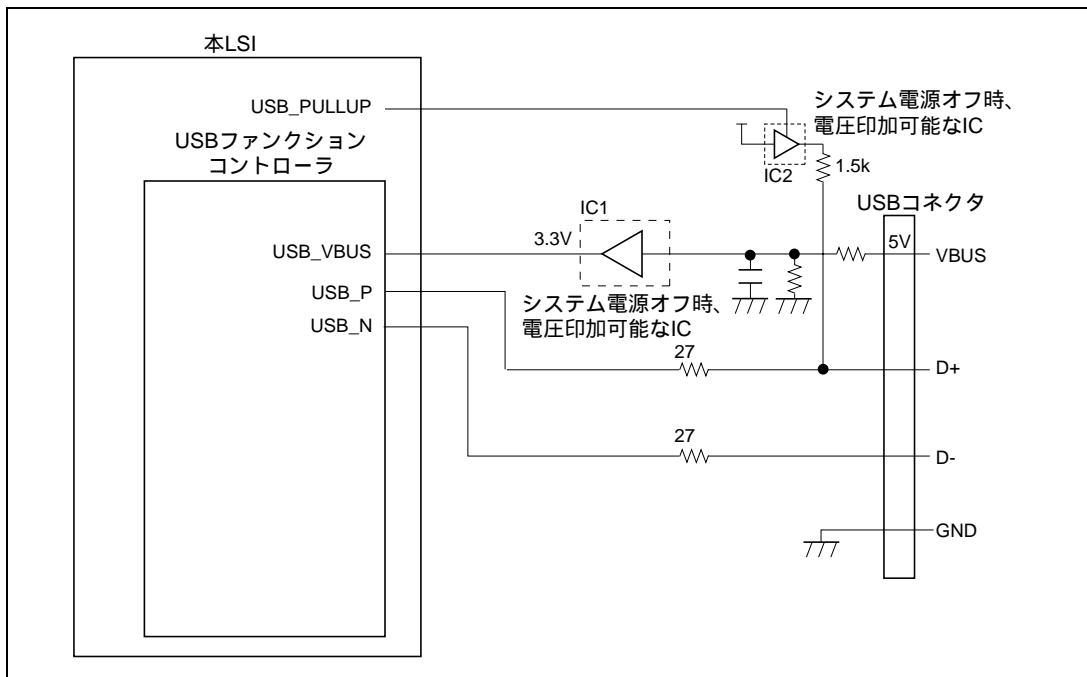


図 18.3 USB ファンクションコントローラと外部回路の接続例 2

- D+のプルアップ制御

USBホストコントローラ/ハブへの接続通知(D+プルアップ)を禁止したい場合(優先度の高い処理中、初期化処理中など)、システムではD+のプルアップをUSB_PULLUP端子を用いて制御してください。図18.2に示す回路例のようにD+のプルアップ制御信号、およびUSB_VBUS端子入力信号はUSB_PULLUP端子とUSBケーブルVBUSを用いて(AND回路)制御してください。図18.2の回路例ではUSB_PULLUP端子がローレベルのとき、D+プルアップ禁止となります(USB_PWR_EN/USB_PULLUP端子の初期値はローになります)。また、図18.3に示す回路例のように直接USB_PULLUP端子でプルアップ制御をする場合は、必ずUSB_VBUS端子がハイレベルになったことを確認してから、D+をプルアップしてください。プルアップ制御用IC(図18.2~図18.3のIC2)には、システム電源オフ時に電圧印加が可能なIC(HD74LVIG126A等)を使用してください(本LSI搭載UDCコアは、USB_VBUS端子=ローレベルのとき、D+、D-の状態にかかわらずパワードステートを保持します)。

- USBケーブル接続/切断の検出

本LSI搭載のUSBファンクションコントローラはハードウェアにてUSBのステート等を管理しているため、接続/切断を認識するUSB_VBUS信号が必要となります。USB_VBUSはUSBケーブル内の電源信号(VBUS)を用いますが、USBファンクションコントローラ(本LSI搭載システム)が電源オフ時、USBホスト/ハブにケーブルが接続されると、USBホストコントローラ/ハブから電圧(5V)が印加されてしまいます。

そのため、図18.2~図18.3のIC1には、システム電源オフ時に電圧印加が可能なIC(HD74LVIG08A等)を使用してください。

18.4.2 USBホストと外部回路の接続例

図18.4にUSBホストコントローラを選択した場合の本LSIと外部回路の接続例を示します。USBホストコントローラの使用に際しては、USBの電源バス制御用に別途LSIを用意する必要があります(図18.4中のUSB電源制御LSIに相当します)。該当LSIには、USB規格を満たす電源供給容量を持ち、過電流保護機能を有するものを選定してください。USB_OVR_CRNT端子への入力は、過電流検出時ローレベルとなるようシステムを構築してください。

18. USB ピンマルチプレクスコントローラ (USBPM)

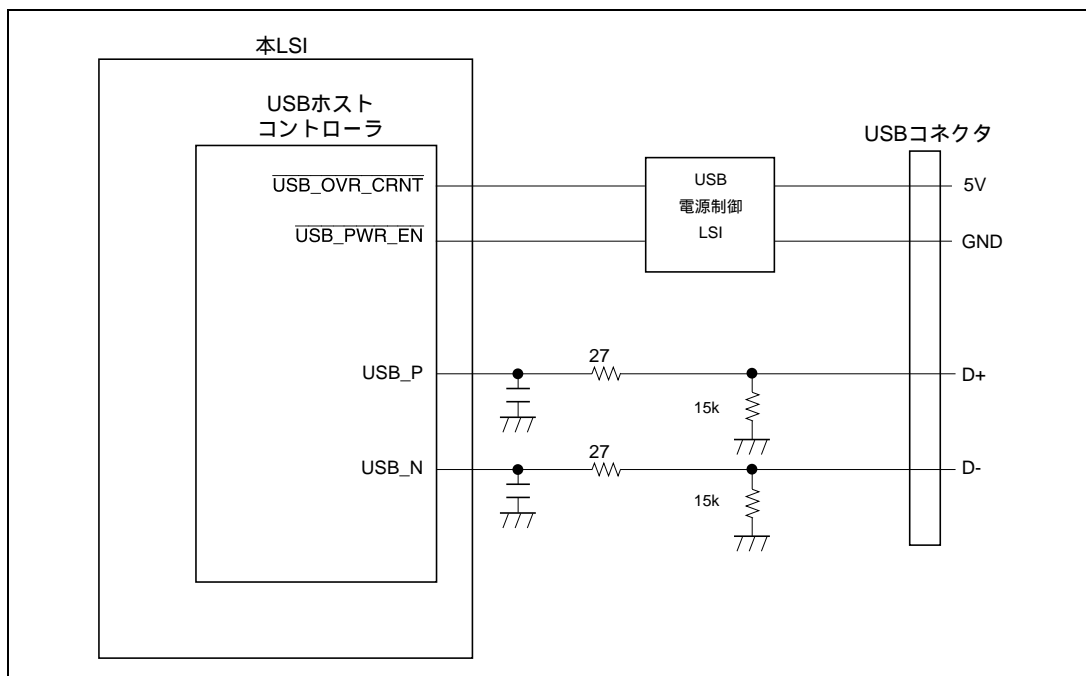


図 18.4 USB ホストコントローラと外部回路の接続例

18.5 使用上の注意事項

(1) USB ホスト / ファンクションと外部回路の接続例に関して

本章記載の外部回路との接続例はあくまで参考例であり、これにより動作を保証するものではありません。

また、外部からのサージ、および ESD ノイズ対策がシステム的に必要な場合、保護ダイオード等で対策してください。

(2) USB に供給されるクロックに関して

USB クロックを内蔵逡倍回路で生成する場合、クロックのソースは CPU クロックのソースである EXTAL 端子からの入力クロックではなく、ブルートゥースの RF IC とのデータ入出力用およびブルートゥースインタフェース用のクロックである RDI_REFCLK_IN 端子からの入力クロックから生成しています。ただし、内蔵逡倍回路を使用した場合、クロック精度は USB 規格の値を保証できません。本 LSI の USB 機能を使用する場合には、上記の方法よりも、UCLK 端子から USB の規格に適合した精度を持つ 48MHz のクロックを入力することを推奨します。

(3) 規格外の市販 USB ケーブル使用による不具合について

実際に市販されている USB ケーブルには、規格外の製品が存在しており、一部の規格外のケーブルを使用すると接続が中断する現象が確認されています。

規格に適合しているケーブルをお使いください。

19. USB ホストコントローラ (USBH)

本 LSI の USB ホストコントローラモジュール (以後、ホストコントローラと称します) は、ユニバーサルシリアルバス (Universal Serial Bus) バージョン 2.0 に対応し、OpenHCI をサポートしています。

USB の OpenHCI 仕様は、ホストコントローラのレジスタ仕様の詳細を規定しています。このホストコントローラのデバイスドライバおよびハードウェアの開発にあたっては、OpenHCI の仕様書に従って開発を行ってください。

19.1 特長

本 LSI のホストコントローラには、以下のような特長があります。

- OpenHCIバージョン1.0レジスタセット準拠
- ユニバーサルシリアルバス (Universal Serial Bus) バージョン2.0対応
- フルスピード (12Mbps) をサポート
- 過電流検出機構をサポート
- 最大127エンドポイントをサポート

【注】 本 LSI では 1 ポートのみの入出力対応のためルートハブ機能はサポートしておりませんが、内部回路構成はルートハブ + 1 ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。

19.2 入出力端子

ホストコントローラの端子構成を表 19.1 に示します。

各端子の詳細な設定方法については、「第 18 章 USB ピンマルチプレクスコントローラ (USBPM)」を参照してください。

表 19.1 端子構成

端子名	入出力	機能
UCLK	入力	外部クロック端子 外部から直接 48MHz クロックを入力するための端子
RDI_REFCLK_IN	入力	外部クロック端子 外部から 13MHz のクロックを入力します。本信号は内部で逡倍され、48MHz のクロックが生成されます。
USB_PWR_EN/ USB_PULLUP	出力	パワーイネーブル端子 / プルアップ制御端子 電源投入許可制御 (USBH 用) / USB プルアップ制御 (USBF 用)
USB_OVR_CRNT/ USB_VBUS	入力	オーバーカレント端子 / VBUS 端子 オーバーカレント検出 (USBH 用) / USB ケーブル接続モニタ (USBF 用)

19. USB ホストコントローラ (USBH)

端子名	入出力	機能
USB_P	入出力	P 端子 D+ポートトランシーバ端子
USB_N	入出力	N 端子 D-ポートトランシーバ端子

【注】 48MHz クロックは、RDL_REFCLK_IN または UCLK のどちらかを選択して使用します。

19.3 レジスタの説明

ホストコントローラ (USBH) には、以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- HcRevision レジスタ (HREVR)
- HcControl レジスタ (HCTLR)
- HcCommandStatus レジスタ (HCSR)
- HcInterruptStatus レジスタ (HISR)
- HcInterruptEnable レジスタ (HIER)
- HcInterruptDisable レジスタ (HIDR)
- HcHCCA レジスタ (HHCCAR)
- HcPeriodCurrentED レジスタ (HPCEDR)
- HcControlHeadED レジスタ (HCHEDR)
- HcControlCurrentED レジスタ (HCCEDR)
- HcBulkHeadED レジスタ (HBHEDR)
- HcBulkCurrentED レジスタ (HBCEDR)
- HcDoneHeadED レジスタ (HDHEDR)
- HcFmInterval レジスタ (HFIR)
- HcFmRemaining レジスタ (HFRR)
- HcFmNumber レジスタ (HFNR)
- HcPeriodicStart レジスタ (HPSR)
- HcLSThreshold レジスタ (HLSTR)
- HcRhDescriptorA レジスタ (HRDRA)
- HcRhDescriptorB レジスタ (HRDRB)
- HcRhStatus レジスタ (HRSR)
- HcRhPortStatus1 レジスタ (HRPSR)

19.3.1 HcRevision レジスタ (HREVR)

ホストコントローラのため実行される HCI 仕様のバージョン番号 (BCD 表現) を含んでいます。値 H'10 はバージョン 1.0 に相当します。この仕様準拠の HCI インプリメンテーションはすべて H'10 の値を持ちます。

ビット	ビット名	初期値	R/W	説明
31~8		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
7	REV7	0	R	Revision バージョン番号 (BCD 表現) H'10 : バージョン 1.0
6	REV6	0	R	
5	REV5	0	R	
4	REV4	1	R	
3	REV3	0	R	
2	REV2	0	R	
1	REV1	0	R	
0	REV0	0	R	

19.3.2 HcControl レジスタ (HCTLR)

ホストコントローラのための操作モードを指定します。

ビット	ビット名	初期値	R/W	説明
31~9		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
8	IR	0	R/W	Interrupt Routine HcInterruptStatus において登録されたイベントによって発生する割り込みの発行を決定します。ホストコントローラドライバは、ハードウェアリセットと同時にこのビットをクリアしますが、ソフトウェアリセットと同時に変更しません。ホストコントローラドライバは、ホストコントローラの所有権を示すために、タグとしてこのビットを用います。 0 : すべての割り込みは正常なホストバス割り込みメカニズムに発行 1 : 割り込みは SMI に発行

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
7 6	HCFS1 HCFS0	0 0	R/W R/W	<p>Host Controller Function Status</p> <p>ホストコントローラドライバは、ホストコントローラが HcInterruptStatus の SF ビットを読み出して SOF を送り始めたかどうかを判定します。ホストコントローラは Usb Suspend 状態のときだけ、このビットを変更できません。ホストコントローラは、ダウンストリームポートから resume 信号を検出した後に、UsbSuspend 状態から UsbResume 状態に遷移可能です。ホストコントローラは、ハードウェアリセットの後に UsbReset となるように、ソフトウェアリセットの後に UsbSuspend 状態になります。前者ではルートハブ*¹がリセットされます。</p> <p>00 : USB Reset 01 : USB Resume 10 : USB Operational 11 : USB Suspend</p>
5	BLE	0	R/W	<p>Bulk List Enable</p> <p>次のフレームでバルクリストの処理を可能にするためにセットされます。ホストコントローラは、リストの処理が決まった場合、このビットをチェックします。ディスエーブルのときには、ホストコントローラドライバはリストを修正できます。HcBulkCurrentED が削除される ED*²を指し示している場合、ホストコントローラドライバは、リスト処理を再可能化する前に、HcBulkCurrentED を更新してポインタを早めてください。</p> <p>0 : バルクリスト処理をしない 1 : バルクリスト処理をする</p>
4	CLE	0	R/W	<p>Control List Enable</p> <p>次のフレームでコントロールリストの処理を可能にするためにセットされます。ホストコントローラドライバによりクリアする場合、次の SOF の後にコントロールリストの処理を行いません。ホストコントローラは、リストを処理するときは必ず、このビットをチェックする必要があります。ディスエーブルのときには、ホストコントローラドライバはリストを修正できません。HcControlCurrentED が削除される ED を指し示している場合、ホストコントローラドライバは、リスト処理の再可能化の前に HcControlCurrentED を更新してポインタを早める必要があります。</p> <p>0 : コントロールリスト処理をしない 1 : コントロールリスト処理をする</p>

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
3	IE	0	R/W	<p>Isochronous Enable</p> <p>isochronous ED の処理をイネーブル/ディスエーブルにするために、ホストコントローラドライバにより用いられます。フレームにおいて周期的なリストを処理する間、ホストコントローラは、isochronous ED を見つけると (F=1)、このビットのステータスをチェックします。セットされている(イネーブル)場合、ホストコントローラはEDを処理し続けます。クリアされている(ディスエーブル)場合、ホストコントローラは、周期的なリスト(現在の isochronous ED のみ含んでいます)の処理を停止させて、バルク/コントロールリストを処理し始めます。このビットのセット状態は、次のフレームにおいて有効であることが保証されています(現在のフレームではありません)。</p> <p>0 : アイソクロナス ED を処理しない 1 : アイソクロナス ED を処理する</p>
2	PLE	0	R/W	<p>Periodic List Enable</p> <p>次のフレームの周期的なリストの処理を可能にするために本ビットをセットします。ホストコントローラドライバによってクリアする場合、周期的なリストの処理は次の SOF の後では発生しません。ホストコントローラがリストの処理を始める前に、ホストコントローラはこのビットをチェックする必要があります。</p> <p>0 : 周期的なリストの処理は次の SOF の後には発生しない 1 : 周期的なリストの処理は次の SOF の後に発生する</p>
1 0	CBSR1 CBSR0	0 0	R/W R/W	<p>Control Bulk Service Ratio</p> <p>コントロールとバルク ED のサービス比率を指定します。非周期的なリストをどれかを処理する前に、ホストコントローラは、別のコントロール ED を供給し続けるか、またはバルク ED に切り換えるか決定する場合、コントロール ED を処理したかどうかの内部計算によって指定した比率を比較する必要があります。内部計算は、フレーム境界を越えるときに保持されます。リセットの場合には、ホストコントローラドライバは、この値を元に戻す必要があります。</p> <p>00 : 1:1 01 : 2:1 10 : 3:1 11 : 4:1</p>

【注】 *1 本 LSI では 1 ポートのみでの出力対応のためルートハブ機能はサポートしていませんが、内部回路構成はルートハブ+1ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。

*2 ED : Endpoint Descriptor

19.3.3 HcCommandStatus レジスタ (HCSR)

ホストコントローラの現在のステータスを反映するだけでなく、ホストコントローラドライバにより発行されたコマンドを受け取るために、ホストコントローラは HcCommandStatus レジスタを使用します。ホストコントローラドライバにとって、セットするために書き込むレジスタです。1 を書き込まれたビットはセットされ、0 を書き込まれたビットは変化しないままであることを、ホストコントローラは保証する必要があります。ホストコントローラドライバは、前もって発行されたコマンドにより複数の明確なコマンドをホストコントローラに配付できます。ホストコントローラドライバは、すべてのビットに対して通常のリードアクセスができます。

SOC0~1 ビットは、ホストコントローラが Scheduling Overrun エラーを検出したフレームの数を示します。EOF の前に周期的なリストが完了しないときに、これは表示されます。Scheduling Overrun エラーを検出すると、ホストコントローラはカウンタをインクリメントし、HcInterruptStatus レジスタの SO フィールドをセットします。

ビット	ビット名	初期値	R/W	説明
31~18		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
17	SOC1	0	R/W	Scheduling Overrun Count 各 SchedulingOverrun エラー発生時に本ビットはインクリメントされます。B'00 と初期設定され、B'11 に戻ります。HcInterruptStatus の SO ビットがセットされていても、SchedulingOverrun を検出すると、本ビットはインクリメントされます。これは、どの持続的スケジューリング問題も監視できるように、ホストコントローラドライバが使用します。
16	SOC0	0	R/W	
15~4		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
3	OCR	0	R/W	Ownership Change Request ホストコントローラの制御を変更するために本ビットは、OS ホストコントローラドライバによりセットされます。セットされると、ホストコントローラが HcInterruptStatus の OC ビットをセットします。変更後に、本ビットがクリアされ、OS ホストコントローラドライバからの次の要求までクリア状態となります。 0 : チェンジ後、このビットがクリアされ、OS ホストコントローラドライバからの次の要求までクリアされた状態 1 : HcInterruptStatus レジスタの OC ビットをセット

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
2	BLF	0	R/W	<p>Bulk List Filled</p> <p>バルクリストに TD*¹ が存在していることを示すために本ビットは使用されます。TD を ED に付加するとき、ホストコントローラドライバにより、リストに設定されます。</p> <p>ホストコントローラがリストのヘッダの処理を開始すると、BLF をチェックします。BLF が 0 であるかぎり、ホストコントローラはリストの処理を開始しません。BLF が 1 のとき、ホストコントローラはリスト処理を開始して、BLF を 0 にセットします。ホストコントローラがリスト上に TD を検出すると、ホストコントローラは、BLF を 1 に設定します。TD がリストに存在せず、ホストコントローラドライバが BLF を設定しないと、ホストコントローラが、リスト処理を完成します。バルクリスト処理を停止すると、BLF は 0 のままです。</p> <p>0 : リストは処理されない 1 : リストを処理</p>
1	CLF	0	R/W	<p>Control List Filled</p> <p>コントロールリスト TD が存在していることを示すために本ビットは使用されます。ホストコントローラドライバが TD をコントロールリストの ED に付加すると、ホストコントローラドライバにより設定されます。</p> <p>ホストコントローラがコントロールリストのヘッダの処理を開始すると、ホストコントローラは CLF をチェックします。CLF が 0 であるかぎり、ホストコントローラはコントロールリスト処理を開始しません。CLF が 1 になると、ホストコントローラはコントロールリスト処理を開始し、CLF を 0 にセットします。ホストコントローラが TD をリスト上で検出すると、ホストコントローラは、CLF を 1 にセットします。もし TD をコントロールリストに検出できず、ホストコントローラドライバが CLF をセットしないと、ホストコントローラが、コントロールリストの処理を完了します。コントロールリスト処理を停止すると、CLF は 0 のままです。</p> <p>0 : リストは処理されない 1 : リストを処理</p>
0	HCR	0	R/W	<p>Host Controller Request</p> <p>ホストコントローラのソフトウェアリセットを開始するために、本ビットはホストコントローラドライバによりセットされます。ホストコントローラの機能上の状態にかかわらず、次の状態を除いて、operational レジスタの大部分がリセットされる UsbSuspend 状態に遷移します。たとえば、HcControl の IR フィールドおよびホストパスのないアクセスは許されます。このビットはリセット操作の完了と同時にホストコントローラによりクリアされます。リセット操作は 10μs 内で完了されなければなりません。このビットは、セットされるときに、ルートハブ*² をリセットせずに、次のリセット信号はダウンストリームポートに出力されません。</p> <p>0 : リセット操作の完了と同時にホストコントローラによりクリア 1 : USB Suspend 状態</p>

【注】 *1 TD : Transfer Descriptor

19. USB ホストコントローラ (USBH)

*2 本 LSI では 1 ポートのみ の 入出力 対応 のため ルート ハブ 機能 は サポート して あり ませ んが、 内部 回路 構成 は ルート ハブ +1 ポート と なっ て あり ます。 この ため、 レジスタ 構成 等 に ルート ハブ の 概念 が 入っ て あり ます。

19.3.4 HcInterruptStatus レジスタ (HISR)

ハードウェア割り込みを起こすさまざまなイベントのステータスを示します。イベントが発生すると、ホストコントローラは、このレジスタ中の対応するビットをセットします。ビットが 1 になると、HcInterruptEnable レジスタ (「19.3.5 HcInterruptEnable レジスタ (HIER)」を参照) において割り込みがイネーブルされている状態で、MIE ビットがセットされると、ハードウェア割り込みが発生します。割り込みが発生すると、割り込み事象レジスタ 2 (INTEVT2) にホストコントローラ割り込み (USBHI) の例外コード (HA00) が設定され、例外処理が始まります。割り込み事象レジスタ 2 (INTEVT2) に関する詳細は、「第 4 章 例外処理」を、ホストコントローラ割り込み (USBHI) の例外コードおよび割り込み処理に関する詳細は、「第 8 章 割り込みコントローラ (INTC)」を参照してください。

ホストコントローラドライバは、クリアされるビットポジションに 1 を書き込むことで、このレジスタの特定ビットをクリアします。ホストコントローラドライバは、これらのどのビットもセットできません。ホストコントローラはビットをクリアすることはありません。

ビット	ビット名	初期値	R/W	説明
31		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
30	OC	0	R/W	Ownership Change ホストコントローラドライバが HcCommandStatus レジスタの OCR ビットをセットするとき、このビットがセットされます。このイベントは、マスクされないと、直ちにシステム管理割り込み (SMI) を生成します。SMI 端子がないと、このビットは 0 になります。 0 : HcCommandStatus レジスタの OCR ビットがセットされていない 1 : HcCommandStatus レジスタの OCR ビットがセットされている
29~7		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
6	RHSC	0	R/W	Root Hub Status Change HcRhStatus レジスタの内容、または HcRhPortStatus レジスタの内容 [ダウンストリームポート番号] のうち、いずれかの内容が変わったときには、このビットがセットされます。 0 : HcRhStatus レジスタ、または HcRhPortStatus レジスタの内容に変化がない 1 : HcRhStatus レジスタ、または HcRhPortStatus レジスタの内容が変化

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
5	FNO	0	R/W	<p>Frame Number Overflow</p> <p>HcFmNumber レジスタの MSB (ビット 15) が、0 から 1 に、または 1 から 0 に値が変化するとき、HccaFrameNumber が更新された後に、セットされます。</p> <p>0 : HcFmNumber レジスタの MSB、または HccaFrameNumber は更新されていない</p> <p>1 : HcFmNumber レジスタの MSB、または HccaFrameNumber が更新された</p>
4	UE	0	R/W	<p>Unrecoverable Error</p> <p>USB と関連しないシステムエラーが検出されたとき、セットされます。ホストコントローラがリセットされると、ホストコントローラドライバはこのビットをクリアします。</p> <p>0 : システムエラー未発生</p> <p>1 : システムエラーを検出</p>
3	RD	0	R/W	<p>Resume Detected</p> <p>USB のデバイスが resume 信号を出力していることをホストコントローラが検出したとき、このビットがセットされます。ホストコントローラドライバが UsbResume state をセットすると、このビットはセットされません。</p> <p>0 : resume 信号を検出していない</p> <p>1 : resume 信号を検出</p>
2	SF	0	R/W	<p>Start of Frame</p> <p>このビットは、各フレームの開始時、および HccaFrameNumber の更新後に、ホストコントローラによりセットされます。ホストコントローラは、SOF トークンも同時に生成します。</p> <p>0 : 各フレームが開始されていない、または HccaFrameNumber が更新されていない</p> <p>1 : 各フレームの開始、および HccaFrameNumber の更新</p>
1	WDH	0	R/W	<p>Write back Done Head</p> <p>ホストコントローラが HcDoneHead を HccaDoneHead に書き込んだすぐ後に、このビットがセットされます。このビットがクリアされるまで、HccaDoneHead は更新されません。ホストコントローラドライバは、HccaDoneHead の内容を保存した後にだけ、このビットをクリアしてください。</p> <p>0 : 1 にセットした後、クリアしたとき</p> <p>1 : HcDoneHead を HccaDoneHead に書き込んだとき</p>
0	SO	0	R/W	<p>Scheduling Overrun</p> <p>現在のフレームにおいて、HccaFrameNumber の更新の後、USB スケジュールがオーバーランしたとき、このビットはセットされます。SchedulingOverrun はまた、HcCommandStatus の SOC0 ~ 1 ビットをインクリメントします。</p> <p>0 : USB スケジュールはオーバーランしていない</p> <p>1 : USB スケジュールがオーバーランした</p>

19.3.5 HcInterruptEnable レジスタ (HIER)

HcInterruptEnable レジスタの各イネーブルビットは、HcInterruptStatus レジスタの関連したイベントによる割り込みの発生許可を制御します。HcInterruptStatus レジスタのいずれかのイベントビットがセットされたとき、HcInterruptEnable レジスタの相当するビットがセットされているかつ、MasterInterruptEnable (MIE) ビットがセットされていると CPU に対してハードウェア割り込みが要求されます。その結果、割り込み事象レジスタ 2 (INTEVT2) にホストコントローラ割り込み (USBHI) の例外コード (H'A00) が設定され、例外処理が始まります (なお、USBHI 例外コードは割り込み生成イベントの内容にかかわらず共通して使用されます)。そのためホストコントローラドライバで割り込み発生を検出する場合に USBHI 例外コードを利用することができます。割り込み事象レジスタ 2 (INTEVT2) に関する詳細は、「第 4 章 例外処理」を、ホストコントローラ割り込み (USBHI) の例外コードおよび割り込み処理に関する詳細は、「第 8 章 割り込みコントローラ (INTC)」を参照してください。

HcInterruptEnable レジスタは、1 を書き込むことによって相当するビットはセットされますが、0 を書き込んでも変化しません。そのため、特定のビットをセットしたいときには、該当ビットのみ 1 としたデータを書き込むことにより、他のビットの状態をチェックすること無しに独立にセットすることが可能です。

HcInterruptEnable レジスタは後述の HcInterruptDisable レジスタと連動しており、特定のビットを 0 クリアしたい場合は、HcInterruptDisable レジスタの該当ビットに 1 を書き込むことにより、HcInterruptEnable レジスタ側の該当ビットが自動的に 0 クリアされます。

ビット	ビット名	初期値	R/W	説明
31	MIE	0	R/W	Master Interrupt Enable このビットに 1 をセットすると、このレジスタの他のビットにおいて指定されたイベントによる割り込み生成を可能にします。これはマスタ割り込みを可能にするためにホストコントローラドライバにより使用されます。 HcInterruptDisable レジスタの MIE ビットに 1 が書き込まれると、このビットは 0 クリアされます。 0: このビットは無視されます 1: 他のビットで指定されたイベントによる割り込み生成可
30	OC	0	R/W	Ownership Change Interrupt Enable このビットに 1 をセットすると、Ownership Change イベントによる割り込み生成を可能にします。HcInterruptDisable レジスタの OC ビットに 1 が書き込まれると、このビットは 0 クリアされます。 0: このビットは無視されます 1: Ownership Change イベントによる割り込み生成可
29~7		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
6	RHSC	0	R/W	<p>Root Hub Status Change Interrupt Enable</p> <p>このビットに1をセットすると、Root Hub Status Change イベントによる割り込み発生を可能にします。HcInterruptDisable レジスタの RHSC ビットに1が書き込まれると、このビットは0クリアされます。</p> <p>0 : このビットは無視されます 1 : Root Hub Status Change イベントによる割り込み発生可</p>
5	FNO	0	R/W	<p>Frame Number Overflow Interrupt Enable</p> <p>このビットに1をセットすると、Frame Number Overflow イベントによる割り込み発生を可能にします。HcInterruptDisable レジスタの FNO ビットに1が書き込まれると、このビットは0クリアされます。</p> <p>0 : このビットは無視されます 1 : Frame Number Overflow イベントによる割り込み発生可</p>
4	UE	0	R/W	<p>Unrecoverable Error Interrupt Enable</p> <p>このビットに1をセットすると、Unrecoverable Error イベントによる割り込み発生を可能にします。HcInterruptDisable レジスタの UE ビットに1が書き込まれると、このビットは0クリアされます。</p> <p>0 : このビットは無視されます 1 : Unrecoverable Error イベントによる割り込み発生可</p>
3	RD	0	R/W	<p>Resume Detected Interrupt Enable</p> <p>このビットに1をセットすると、Resume Detected イベントによる割り込み発生を可能にします。HcInterruptDisable レジスタの RD ビットに1が書き込まれると、このビットは0クリアされます。</p> <p>0 : このビットは無視されます 1 : Resume Detected イベントによる割り込み発生可</p>
2	SF	0	R/W	<p>Start of Frame Interrupt Enable</p> <p>このビットに1をセットすると、Start of Frame イベントによる割り込み発生を可能にします。HcInterruptDisable レジスタの SF ビットに1が書き込まれると、このビットは0クリアされます。</p> <p>0 : このビットは無視されます 1 : Start of Frame イベントによる割り込み発生可</p>
1	WDH	0	R/W	<p>Write back Done Head Interrupt Enable</p> <p>このビットに1をセットすると、Write back Done Head イベントによる割り込み発生を可能にします。HcInterruptDisable レジスタの WDH ビットに1が書き込まれると、このビットは0クリアされます。</p> <p>0 : このビットは無視されます 1 : Write back Done Head イベントによる割り込み発生可</p>

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
0	SO	0	R/W	Scheduling Overrun Interrupt Enable このビットに 1 をセットすると、Scheduling Overrun イベントによる割り込み発生を可能にします。HcInterruptDisable レジスタの SO ビットに 1 が書き込まれると、このビットは 0 クリアされます。 0 : このビットは無視されます 1 : Scheduling Overrun イベントによる割り込み発生可

19.3.6 HcInterruptDisable レジスタ (HIDR)

HcInterruptDisable レジスタの各ディスエーブルビットは、HcInterruptStatus レジスタの関連したイベントによる割り込みマスクを制御します。HcInterruptStatus レジスタのいずれかのイベントビットがセットされたとき、HcInterruptDisable レジスタの相当するビットがセットされているか、または HcInterruptDisable レジスタの MIE ビットがセットされていると、CPU に対するハードウェア割り込み要求は発生しません。

このレジスタに 1 を書き込むことによって HcInterruptEnable レジスタの相当するビットはクリア (ディスエーブル状態) されますが、0 を書き込むことによって HcInterruptEnable レジスタの対応するビットは変化しません。そのため、特定のビットをディスエーブル状態にしたいときには、該当ビットのみ 1 としたデータを書き込むことにより、他のビットの状態をチェックせずに独立に任意のビットをディスエーブル状態にすることが可能です。

HcInterruptDisable レジスタは前述の HcInterruptEnable レジスタと連動しています。したがって、特定のビットを 0 クリアしたい (イネーブル状態にしたい) 場合は、HcInterruptEnable レジスタの該当ビットに 1 を書き込むことにより、HcInterruptDisable レジスタ側の該当ビットが自動的に 0 クリアされます。HcInterruptDisable レジスタは読み出した際に、その時点での HcInterruptEnable レジスタの内容が反映されます。

ビット	ビット名	初期値	R/W	説明
31	MIE	0	R/W	Master Interrupt Disable このビットに 1 をセットすると、このレジスタの他のビットにおいて指定されたイベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの MIE ビットに 1 が書き込まれると、このビットは 0 クリアされます。 0 : このビットは無視されます 1 : 他のビットで指定されたイベントによる割り込み発生不可
30	OC	0	R/W	Ownership Change Interrupt Disable このビットに 1 をセットすると、Ownership Change イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの OC ビットに 1 が書き込まれると、このビットは 0 クリアされます。 0 : このビットは無視されます 1 : Ownership Change イベントによる割り込み発生不可
29~7		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
6	RHSC	0	R/W	<p>Root Hub Status Change Interrupt Disable</p> <p>このビットに 1 をセットすると、Root Hub Status Change イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの RHSC ビットに 1 が書き込まれると、このビットは 0 クリアされます。</p> <p>0 : このビットは無視されます 1 : Root Hub Status Change イベントによる割り込み発生不可</p>
5	FNO	0	R/W	<p>Frame Number Overflow Interrupt Disable</p> <p>このビットに 1 をセットすると、Frame Number Overflow イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの FNO ビットに 1 が書き込まれると、このビットは 0 クリアされます。</p> <p>0 : このビットは無視されます 1 : Frame Number Overflow イベントによる割り込み発生不可</p>
4	UE	0	R/W	<p>Unrecoverable Error Interrupt Disable</p> <p>このビットに 1 をセットすると、Unrecoverable Error イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの UE ビットに 1 が書き込まれると、このビットは 0 クリアされます。</p> <p>0 : このビットは無視されます 1 : Unrecoverable Error イベントによる割り込み発生不可</p>
3	RD	0	R/W	<p>Resume Detected Interrupt Disable</p> <p>このビットに 1 をセットすると、Resume Detected イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの RD ビットに 1 が書き込まれると、このビットは 0 クリアされます。</p> <p>0 : このビットは無視されます 1 : Resume Detected イベントによる割り込み発生不可</p>
2	SF	0	R/W	<p>Start of Frame Interrupt Disable</p> <p>このビットに 1 をセットすると、Start of Frame イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの SF ビットに 1 が書き込まれると、このビットは 0 クリアされます。</p> <p>0 : このビットは無視されます 1 : Start of Frame イベントによる割り込み発生不可</p>
1	WDH	0	R/W	<p>Write back Done Head Interrupt Disable</p> <p>このビットに 1 をセットすると、Write back Done Head イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの WDH ビットに 1 が書き込まれると、このビットは 0 クリアされます。</p> <p>0 : このビットは無視されます 1 : Write back Done Head イベントによる割り込み発生不可</p>

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
0	SO	0	R/W	Scheduling Overrun Interrupt Disable このビットに 1 をセットすると、Scheduling Overrun イベントによる割り込み発生をマスクします。HcInterruptEnable レジスタの SO ビットに 1 が書き込まれると、このビットは 0 クリアされます。 0 : このビットは無視されます 1 : Scheduling Overrun イベントによる割り込み発生不可

19.3.7 HcHCCA レジスタ (HHCCAR)

HcHCCA レジスタはホストコントローラコミュニケーションエリアの物理アドレスを含んでいます。ホストコントローラドライバは、HcHCCA レジスタにすべて 1 を書き込むことと、HcHCCA レジスタの内容を読み出すことによって、アラインメント制限を決定します。アラインメントは、下位ビットの 0 の数を調べることで判定します。最小のアラインメントは 256 バイトです。したがって、0 から 7 までのビットは、読み出されたときに、常に 0 にしておかなければなりません。このエリアは、ホストコントローラとホストコントローラドライバによりアクセスされる、コントロール構造と割り込みテーブルを保持するために使用されます。

ビット	ビット名	初期値	R/W	説明
31~8	HCCA	0	R/W	Host Controller Communication Area の物理アドレス
7~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19.3.8 HcPeriodCurrentED レジスタ (HPCEDR)

HcPeriodCurrentED レジスタは、現在の Isochronous ED あるいは Interrupt ED の物理アドレスを含んでいます。

ビット	ビット名	初期値	R/W	説明
31~4	PCED	0	R	Period Current ED 現在の Isochronous ED あるいは Interrupt ED の物理アドレス
3~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19.3.9 HcControlHeadED レジスタ (HCHEDR)

HcControlHeadED レジスタは、コントロールリストにおいて、最初の ED の物理アドレスを含んでいます。

ビット	ビット名	初期値	R/W	説明
31~4	CHED	0	R/W	Control Head ED コントロールリストにおける最初の ED の物理アドレス
3~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19.3.10 HcControlCurrentED レジスタ (HCCEDR)

HcControlCurrentED レジスタは、コントロールリストにおいて、現在の ED の物理アドレスを含んでいます。

ビット	ビット名	初期値	R/W	説明
31~4	CCED	0	R/W	Control Current ED コントロールリストにおける現在の ED の物理アドレス
3~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19.3.11 HcBulkHeadED レジスタ (HBHEDR)

HcBulkHeadED レジスタは、バルクリストにおいて最初の ED の物理アドレスを含んでいます。

ビット	ビット名	初期値	R/W	説明
31~4	BHED	0	R/W	Bulk Head ED バルクリストにおける最初の ED の物理アドレス
3~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19.3.12 HcBulkCurrentED レジスタ (HBCEDR)

HcBulkCurrentED レジスタは、バルクリストにおいて、現在の ED の物理アドレスを含んでいます。バルクリストがラウンドロビン方式で供給されると、エンドポイントはそれらの挿入に従ってリストに配列されます。

ビット	ビット名	初期値	R/W	説明
31~4	BCED	0	R/W	Bulk Current ED バルクリストにおける現在の ED の物理アドレス
3~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19. USB ホストコントローラ (USBH)

19.3.13 HcDoneHeadED レジスタ (HDHEDR)

HcDoneHead レジスタは、Done queue に付加された、直前に完了した TD の物理アドレスを含んでいます。正常な動作において、ホストコントローラドライバは、その内容が HCCA に周期的に書き込まれるようにこのレジスタを読み出す必要はありません。

ビット	ビット名	初期値	R/W	説明
31~4	DH	0	R	Done Head Done queue に付加された、直前に完了した TD の物理アドレス
3~0		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19.3.14 HcFmInterval レジスタ (HFIR)

HcFmInterval レジスタは、フレームのビットタイム間隔 (すなわち 2 つの連続的な SOF 間) を示す 14 ビット値と、scheduling overrun を発生させずにホストコントローラが送受信するフルスピードでの最大パケットサイズを示す 15 ビットの値を含んでいます。ホストコントローラドライバは、各 SOF において、現在の値の上に新しい値を書き込んで、フレームインターバルの微調整を行います。

ビット	ビット名	初期値	R/W	説明
31	FIT	0	R/W	FrameIntervalToggle ホストコントローラドライバは、新しい値を FrameInterval にロードするときはいつでも、このビットをトグルします。
30	FSMPS14	0	R/W	FSLargestDataPacket
29	FSMPS13	0	R/W	このビットは、各フレームの最初に Largest Data Packet Counter にロードされる値を指定します。カウンタ値は、scheduling overrun を発生させずに、与えられたどの時間でも、ホストコントローラにより一つのトランザクションに送受信され得るビットの最も大きいデータ量を表しています。フィールド値はホストコントローラドライバにより計算されます。
28	FSMPS12	0	R/W	
27	FSMPS11	0	R/W	
26	FSMPS10	0	R/W	
25	FSMPS9	0	R/W	
24	FSMPS8	0	R/W	
23	FSMPS7	0	R/W	
22	FSMPS6	0	R/W	
21	FSMPS5	0	R/W	
20	FSMPS4	0	R/W	
19	FSMPS3	0	R/W	
18	FSMPS2	0	R/W	
17	FSMPS1	0	R/W	
16	FSMPS0	0	R/W	
15、14		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説 明
13	FI13	1	R/W	FrameInterval
12	FI12	0	R/W	このビットは、ビットタイムの2つの連続的なSOF間の間隔を指定します。標準値は、H'2EDF (11,999) にセットされます。 ホストコントローラドライバは、ホストコントローラをリセットする前に、このフィールドの現在値を保存する必要があります。これにより、HcCommandStatus レジスタのHCR ビットをセットすることによって、ホストコントローラによってこのビットをその標準値にリセットします。ホストコントローラドライバは、リセットシーケンスの完了と同時に保存された値に戻すことを選ぶことができます。
11	FI11	1	R/W	
10	FI10	1	R/W	
9	FI9	1	R/W	
8	FI8	0	R/W	
7	FI7	1	R/W	
6	FI6	1	R/W	
5	FI5	0	R/W	
4	FI4	1	R/W	
3	FI3	1	R/W	
2	FI2	1	R/W	
1	FI1	1	R/W	
0	FI0	1	R/W	

19. USB ホストコントローラ (USBH)

19.3.15 HcFmRemaining レジスタ (HFRR)

HcFmRemaining レジスタは、現在のフレームに残っているビットタイムを示す 14 ビットのダウンカウンタです。

ビット	ビット名	初期値	R/W	説明
31	FRT	0	R	Frame Remaining Toggle FR が 0 に達するときは常に、このビットは HcFmInterval レジスタの FIT ビットからロードされます。このビットは FrameInterval と FrameRemaining レジスタを同期させるためにホストコントローラドライバにより使われません。
30 ~ 14		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
13	FR13	0	R	Frame Remaining このカウンタは各ビットタイムでデクリメントされます。0 に達するとき、次のビットタイム境界において、HcFmInterval レジスタにより指定された FrameInterval ビットの値をロードすることによって本カウンタはリセットされます。UsbOperational 状態に入るときに、ホストコントローラは HcFmInterval レジスタの FrameInterval ビットを再度読み込み、次の SOF から更新された値を使います。
12	FR12	0	R	
11	FR11	0	R	
10	FR10	0	R	
9	FR9	0	R	
8	FR8	0	R	
7	FR7	0	R	
6	FR6	0	R	
5	FR5	0	R	
4	FR4	0	R	
3	FR3	0	R	
2	FR2	0	R	
1	FR1	0	R	
0	FR0	0	R	

19.3.16 HcFmNumber レジスタ (HFNR)

HcFmNumber レジスタは 16 ビットカウンタです。ホストコントローラとホストコントローラドライバにおいて発生するイベント間のタイミングの参照を示します。ホストコントローラドライバは、このレジスタにおいて指定された 16 ビット値を使い、レジスタへの頻繁なアクセスを必要とせずに 32 ビットフレーム数を生成します。

ビット	ビット名	初期値	R/W	説明
31 ~ 16		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
15	FN15	0	R	Frame Number HcFmRemaining が再ロードされるときにインクリメントされます。H'FFFF に達すると、その後 H'0 に戻ります。UsbOperational 状態に入るときに、自動的にインクリメントされます。ホストコントローラが各フレーム境界において、FN15 ~ 0 ビットをインクリメントし、SOF を送った後、ホストコントローラがそのフレームの中で最初の ED を読み出す前に、その内容が HCCA に書かれます。HCCA に書いた後、ホストコントローラは HcInterruptStatus レジスタの SF ビットをセットします。
14	FN14	0	R	
13	FN13	0	R	
12	FN12	0	R	
11	FN11	0	R	
10	FN10	0	R	
9	FN9	0	R	
8	FN8	0	R	
7	FN7	0	R	
6	FN6	0	R	
5	FN5	0	R	
4	FN4	0	R	
3	FN3	0	R	
2	FN2	0	R	
1	FN1	0	R	
0	FN0	0	R	

19. USB ホストコントローラ (USBH)

19.3.17 HcPeriodicStart レジスタ (HPSR)

HcPeriodicStart レジスタは、ホストコントローラが周期的なりスト処理を開始する最も早い時間を決定するための、14 ビットのプログラム可能な値を持っています。

ビット	ビット名	初期値	R/W	説明
31 ~ 14		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
13	PS13	0	R/W	Periodic Start ハードウェアリセット後、このフィールドはクリアされます。その後ホストコントローラの初期設定間に、ホストコントローラドライバによりセットされます。この値は、HcFmInterval レジスタからおおよそ 10% off した値に計算されます。通常の値は、H'2A2F です。HcFmRemaining レジスタが指定された値に達すると、周期的なりスト処理は、コントロール/バルク処理よりプライオリティを持ちます。したがって、ホストコントローラは、現在のコントロール/バルクトランザクションを完了後に、周期的なりスト処理を開始します。
12	PS12	0	R/W	
11	PS11	0	R/W	
10	PS10	0	R/W	
9	PS9	0	R/W	
8	PS8	0	R/W	
7	PS7	0	R/W	
6	PS6	0	R/W	
5	PS5	0	R/W	
4	PS4	0	R/W	
3	PS3	0	R/W	
2	PS2	0	R/W	
1	PS1	0	R/W	
0	PS0	0	R/W	

19.3.18 HcLSThreshold レジスタ (HLSTR)

HcLSThreshold レジスタは、EOF の前に最大 8 バイトの LS パケットの転送を行うかどうかを決めるために、ホストコントローラにより用いられる 12 ビットの値を含んでいます。ホストコントローラとホストコントローラドライバによって、この値を変更することはできません。なお、本 LSI ではロースピードはサポートしておりません。

ビット	ビット名	初期値	R/W	説明
31~12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
11	LST11	0	R/W	LS Threshold このフィールドは、ロースピードトランザクションの開始前に FrameRemaining ビットと比較される値を含んでいます。FrameRemaining ビットの値がこのビットの値以上である場合のみトランザクションを開始します。値はトランスミッションとセットアップオーバーヘッドを考慮してホストコントローラドライバが計算します。
10	LST10	1	R/W	
9	LST9	1	R/W	
8	LST8	0	R/W	
7	LST7	0	R/W	
6	LST6	0	R/W	
5	LST5	1	R/W	
4	LST4	0	R/W	
3	LST3	1	R/W	
2	LST2	0	R/W	
1	LST1	0	R/W	
0	LST0	0	R/W	

19.3.19 HcRhDescriptorA レジスタ (HRDRA)

HcRhDescriptorA レジスタは、ルートハブ*のフィーチャーを表す2つのレジスタで、そのうちの1番目のレジスタです。ディスクリプタ長(11)、ディスクリプタタイプ(TBD)、ハブの Class Descriptor の hub controller current ビット(0)は、ホストコントローラドライバによりエミュレートします。その他すべてのビットは HcRhDescriptorA レジスタと HcRhDescriptorB レジスタに配置されます。

ビット	ビット名	初期値	R/W	説明
31	POTPGT7	0	R/W	PowerOnToPowerGoodTime このビットは、ホストコントローラドライバが、ルートハブ*のパワーオンポートにアクセスするまでの待ち時間を指定します。インプリメンテーション固有です。時間の単位は2msです。時間は POTPGT×2ms として計算されます。
30	POTPGT6	0	R/W	
29	POTPGT5	0	R/W	
28	POTPGT4	0	R/W	
27	POTPGT3	0	R/W	
26	POTPGT2	0	R/W	
25	POTPGT1	1	R/W	
24	POTPGT0	0	R/W	
23~13		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
12	NOCP	1	R/W	No Over Current Protection このビットは、ルートハブポート*の過電流ステータスの通知方法を選択します。このビットがクリアされるときに、OCPM ビットは、グローバル通知/ポート毎通知を指定します。 0: 過電流ステータスはすべてのダウンストリームポートに対し集散的に通知される 1: 過電流保護はサポートされない
11	OCPM	0	R/W	Over Current Protection Mode このビットは、ルートハブポート*の過電流ステータスの通知方法を選択します。リセット時に、このビットは PSM ビットと同じモードを反映します。NOCP ビットがクリアされた場合、このビットは有効となります。 0: 過電流ステータスはすべてのダウンストリームポートに対し集散的に報告される(ただし、使用禁止:「19.5.7 過電流検出機能について」参照) 1: 過電流ステータスはポートごとに通知される
10	DT	0	R	Device Type このビットは、ルートハブ*が合成デバイスではないことを示します。このビットは、常に0をセットしてください。

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
9	NPS	1	R/W	<p>No Power Switching</p> <p>このビットは電源切り替えのサポート、またはポートに常に電力を供給するかを選択します。インプリメンテーション固有です。このビットがクリアされるとき、PSM ビットはグローバル/ポート切り替えを指定します。</p> <p>【注】初期値が1であるため、ポートを電源切替可能にする場合は、あらかじめクリア(ホストコントローラドライバで0書き込み)してください。</p> <p>0: ポートは電源切り替え可能</p> <p>1: ホストコントローラがパワーオン時には、ポートは常にパワーオン</p>
8	PSM	0	R/W	<p>Power Switching Mode</p> <p>このビットは、ルートハブポート*のパワー切り替えの制御方法を指定します。インプリメンテーション固有です。NPS ビットがクリアされる場合のみ、このビットは有効です。</p> <p>0: すべてのポートに同時に電力を供給します。</p> <p>1: 個々のポートに個々に電力を供給します。このモードでは、ポートパワーは、グローバルスイッチまたはポートごとスイッチのいずれかにより制御します。HcRhDescriptorB レジスタのPPCM ビットがセットされると、ポートはポートパワーコマンドにだけ応答します(セット/クリアポートパワー)。ポートマスクをクリアする場合、ポートはグローバル電源スイッチだけにより制御されます(セット/クリアグローバルパワー)。</p>
7	NDP7	0	R	<p>Number Downstream Ports</p> <p>これらのビットは、ルートハブ*によってサポートされたダウンストリームポートの数を指定します。インプリメンテーション固有です。ポートの最小数は1です。OpenHCI がサポートするポートの最大数は15です。</p>
6	NDP6	0	R	
5	NDP5	0	R	
4	NDP4	0	R	
3	NDP3	0	R	
2	NDP2	0	R	
1	NDP1	1	R	
0	NDP0	0	R	

【注】 * 本 LSI では 1 ポートのみの入出力対応のためルートハブ機能はサポートしていませんが、内部回路構成はルートハブ+1ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。

19.3.20 HcRhDescriptorB レジスタ (HRDRB)

HcRhDescriptorB レジスタは、ルートハブ*のフィーチャーを表す2つのレジスタで、そのうちの2番目のレジスタです。これらのビットは、システムインプリメンテーションに適合するように、初期設定の際に設定してください。

ビット	ビット名	初期値	R/W	説明
31~18		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
17	PPCM	0	R/W	Port Power Control Mask このビットは、ポートが、HcRhDescriptorA レジスタの PSM ビットがセットされるとき、グローバルなパワーコントロールコマンドにより影響されるかどうかを示します。セット時は、ポートのパワー状態は、ポートごとのパワーコントロールにより影響されます(セット/クリアポートパワー)。クリア時は、ポートはグローバル電源スイッチにより制御します(セット/クリアグローバルパワー)。もしデバイスがグローバル切り替えモード(PSM ビット=0)に設定されているならば、このビットは有効ではありません。 【注】 HcRhDescriptorA レジスタの NPS ビットをクリアし、全ポートをパワーオフ(PortPowerStatus=0)してから、本ビットをセットしてください。 0: グローバル電源スイッチにより制御します 1: ポート#1 はポートごとのパワーコントロールにより影響されます
16~2		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
1	DR	0	R/W	Device Removable これらのビットは、ルートハブ*のポート専用のビットです。クリア時に、取り付けられたデバイスが取り外し可能になります。セット時には、取り付けられたデバイスは取り外さないでください。 0: ポート#1 に付属しているデバイスは、取り外し可能 1: ポート#1 に付属しているデバイスは、取り外し不可
0		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。

【注】 * 本 LSI では 1 ポートのみの入出力対応のためルートハブ機能はサポートしておりませんが、内部回路構成はルートハブ+1ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。

19.3.21 HcRhStatus レジスタ (HRSR)

HcRhStatus レジスタは2つのパーツに分割されます。ロングワードの下位ワードは、ハブステータスビットを表し、上位ワードは、ハブステータスチェンジビットを表しています。リザーブビットには、0をセットしてください。

ビット	ビット名	初期値	R/W	説明
31	CRWE	0	W	Clear Remote Wakeup Enable 1を書くことによりクリアされます。0をセットした場合、クリアされません。
30~18		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。
17	OCIC	0	R/W	Over Current Indicator Change このレジスタのOCIフィールドが変更されると、このビットはホストコントローラによりセットされます。ホストコントローラドライバは、1を書いてこのビットをクリアします。0をセットした場合、クリアされません。
16	LPSC	0	R/W	(読み出し時) LocalPowerStatusChange ルートハブ*はローカルなパワーステータス機能をサポートしません。したがって、このビットは常に0が読み出されます。 (書き込み時) SetGlobalPower グローバルなパワーモード (HcRhDescriptorA レジスタのPSM ビット=0) において、このビットは、すべてのポートをパワーオンするために、1を書き込みます (HcRhPortStatus レジスタのPPS ビットをセット)。ポートごとにパワーモードにおいて、HcRhDescriptorB レジスタのPPCM ビットがセットされていないポートにだけPPS ビットをセットします。0を書き込んだ場合、クリアされません。
15	DRWE	0	R/W	(読み出し時) DeviceRemoteWakeupEnable このビットは resume イベントとして HcRhPortStatus レジスタの CSC ビットをイネーブルし、UsbSuspend から UsbResume への状態変遷、および ResumeDetected 割り込みをセットします。 0 : ConnectStatusChange が remote wakeup イベントでない 1 : ConnectStatusChange が remote wakeup イベント (書き込み時) SetRemoteWakeupEnable 1を書き込むことによりセットします。0を書き込んだ場合、セットしません。
14~2		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。1を書いた場合の動作の保証はできません。

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
1	OCI	0	R/W	Over Current Indicator (使用禁止: 「19.5.7 過電流検出機能について」参照)
0	LPS	0	R/W	(読み出し時) LocalPowerStatus ルートハブ*はローカルなパワーステータス機能をサポートしません。したがって、このビットは常に0が読み出されます。 (書き込み時) Clear Global Power グローバルパワーモード (HcRhDescriptorA レジスタの PSM ビット=0) において、このビットは、すべてのポートをパワーオフするために、1を書き込みます (HcRhPortStatus レジスタの PPS ビットをクリア)。ポートごとのパワーモードにおいては、PPS ビットを、HcRhDescriptorB レジスタの PPCM ビットがセットされていないポートに対してだけクリアします。0を書き込んだ場合、無効です。

【注】 * 本 LSI では 1 ポートのみの入出力対応のためルートハブ機能はサポートしていませんが、内部回路構成はルートハブ+1ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。

19.3.22 HcRhPortStatus レジスタ (HRPSR)

HcRhPortStatus レジスタは、ポートのベース制御とポートイベントの通知に使用します。上位ワードがステータス変化を反映するのに対して、下位ワードは、ポートステータスを反映するために使用します。ステータスビットの中には、特別な書き込みをするものがあります (下記参照)。change port status への書き込みが行われているときにトランザクション (ハンドシェイクを通るトークン) が進行中ならば、結果として発生しているポートステータス変化は、トランザクションが完了するまで延期しなくてはなりません。リザーブビットには、常に0を書き込んでください。

ビット	ビット名	初期値	R/W	説明
31~21		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
20	PRSC	0	R/W	Port Reset Status Change このビットは、10ms のポートリセット信号の終了時にセットされます。ホストコントローラドライバは、このビットをクリアするために1を書き込みます。0を書き込んだ場合は、クリアされません。 0: ポートリセットは完了していない 1: ポートリセットは完了

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
19	OCIC	0	R/W	<p>Port Over Current Indicator Change</p> <p>ポートごとのベースにおいて過電流条件を通知するとき、このビットは有効です。ルートハブ*が POCI ビットを変更するときには、このビットがセットされます。ホストコントローラドライバは、このビットをクリアするために、1 を書き込みます。0 を書き込んだ場合、クリアされません。</p> <p>0 : PortOverCurrentIndicator の変更なし 1 : PortOverCurrentIndicator が変更された</p>
18	PSSC	0	R/W	<p>Port Suspend Status Change</p> <p>すべての resume シーケンスが完了したとき、このビットがセットされます。このシーケンスは、20ms の resume パルス、LS EOP、および 3 ms resynchronization 遅延を含みます。ホストコントローラドライバは、このビットをクリアするために、1 を書き込みます。0 を書き込んだ場合、クリアされません。PRSC ビットが設定されるときも、このビットがクリアされません。</p> <p>0 : resume は完成されていない 1 : resume は完成</p>
17	PESC	0	R/W	<p>Port Enable Status Change</p> <p>ハードウェアイベントにより、PES ビットがクリア時に、このビットがセットされます。ホストコントローラドライバの書き込みによる変更ではセットされません。ホストコントローラドライバは、このビットをクリアするために、1 を書き込みます。0 を書き込んだ場合、クリアされません。</p> <p>0 : PortEnableStatus の変化なし 1 : PortEnableStatus の変化あり</p>
16	CSC	0	R/W	<p>Connect Status Change</p> <p>接続または切断イベントが発生するとこのビットがセットされます。ホストコントローラドライバは、このビットをクリアするために、1 を書き込みます。0 を書き込んだ場合、クリアされません。SetPortReset、SetPortEnable、または SetPortSuspend を書き込むときに CCS ビットがクリアされた場合、ポートの電源切断時の書き込みは起こらないので、接続ステータスを再評価することをドライバに示すためにセットされます。</p> <p>【注】 HcRhDescriptorB レジスタの DR ビットがセットされている場合、このビットは、デバイスが取り付けられることをシステムに通知するために、ルートハブプリセット*の後にだけセットされます。</p> <p>0 : CurrentConnectStatus の変化なし 1 : CurrentConnectStatus の変化あり</p>
15 ~ 10		すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込み値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
9	LSDA	0	R/W	<p>(読み出し時) LowSpeedDeviceAttached</p> <p>このビットは、このポートに取り付けられたデバイスのスピードを示します。セット時は、ロースピードデバイスがこのポートに取り付けられています。クリア時は、フルスピードデバイスがこのポートに取り付けられています。CCS ビットがセットされているときのみ、このビットが有効です。</p> <p>0 : フルスピードデバイスを接続 1 : ロースピードデバイスを接続</p> <p>(書き込み時) ClearPortPower</p> <p>ホストコントローラドライバは、このビットに 1 を書き込むことによって、PPS ビットをクリアします。0 を書き込んだ場合、クリアしません。なお、本 LSI ではロースピードはサポートしておりません。</p>
8	PPS	1	R/W	<p>(読み出し時) PortPowerStatus</p> <p>このビットは、実行されるパワー切り替えモードにかかわらずポートのパワーステータスを反映します。ただし、HcRhDescriptorA レジスタの NoPowerSwitching ビットの初期値が 1 であるため、本ビットは最初は 1 固定となっています。以下のようにパワー切り替えを行う場合は、あらかじめ NoPowerSwitching ビットをクリアする必要があります。</p> <p>過電流条件が検出されると、このビットはクリアされます。ホストコントローラドライバは、SetPortPower または SetGlobalPower に書き込みをしてこのビットをセットします。また、ホストコントローラドライバは、ClearPortPower または ClearGlobalPower に書き込みをしてこのビットをクリアします。HcRhDescriptorA レジスタの PSM ビットと HcRhDescriptorB レジスタの PPCM ビットによって、どのパワーコントロールスイッチが使用可能であるかが決定されます。グローバルな切り替えモード (PSM ビット=0) では、セット/クリアグローバルパワーだけにより、このビットを制御します。ポートごとのパワー切り替えモード (PSM ビット=1) のときに、そのポートの PPCM ビットをセットする場合、セット/クリアポートパワーコマンドだけが使用可能です。マスクを設定していない場合、セット/クリアグローバルパワーコマンドを使用可能です。ポートパワーがディセーブル時には、CCS ビット、PES ビット、PSS ビット、および PRS ビットがリセットされます。</p> <p>【注】パワー切り替えをサポートしていないと、このビットは常に 1 が読み出されます。</p> <p>0 : ポートパワーはオフ 1 : ポートパワーはオン</p> <p>(書き込み時) SetPortPower</p> <p>ホストコントローラドライバは、PPS ビットをセットするために、1 を書き込みます。0 を書き込んだ場合、セットされません。</p>
7~5		すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
4	PRS	0	R/W	<p>(読み出し時) PortResetStatus</p> <p>SetPortReset への書き込みによりこのビットがセットされると、ポートリセット信号が出力されます。リセット完了時は、PRSC ビットをセットするとき、このビットがクリアされます。CCS ビットがクリアされている場合、このビットはセットされません。</p> <p>0 : ポートリセット信号はアクティブでない 1 : ポートリセット信号はアクティブ</p> <p>(書き込み時) SetPortReset</p> <p>ホストコントローラドライバは、このビットに 1 を書き込んで、ポートリセット信号をセットします。0 を書き込んだ場合、セットされません。CCS ビットがクリアされていると、この書き込みにより PRS ビットはセットされませんが、代わりに、CSC ビットをセットします。これにより、未接続ポートのリセットをドライバに通知します。</p>
3	POCI	0	R/W	<p>(読み出し時) PortOverCurrentIndicator</p> <p>ポートごとのベースにおいて過電流条件を通知する方式において、ルートハブ*が配置されているときのみ、このビットは有効になります。ポートごとの過電流通知をサポートしていないと、このビットは 0 にクリアされます。本ビットをクリアすると、すべてのパワー制御はこのポートにおいて正常となります。セットされた場合、このポートにおいて過電流状態が存在することになります。このビットは常に過電流の入力信号を反映します。</p> <p>0 : 過電流状態なし 1 : 過電流状態を検出</p> <p>(書き込み時) ClearSuspendStatus</p> <p>ホストコントローラドライバは、リジューム開始のために、1 を書き込みます。0 を書き込んだ場合、無効です。PSS ビットがセットされているときのみ、リジュームを開始します。</p>

19. USB ホストコントローラ (USBH)

ビット	ビット名	初期値	R/W	説明
2	PSS	0	R/W	<p>(読み出し時) PortSuspendStatus</p> <p>このビットは、ポートが中断処理中、または再開シーケンス中であることを示します。SetSuspendState の書き込みによりセットし、リジューム間隔の終わりで PSSC ビットをセットするときにクリアされます。CCS ビットがクリアされると、このビットはセットできません。ポートリセットの終了時において PRSC ビットがセットされる時、またはホストコントローラが USB リジューム状態になったときに、このビットはクリアされます。もしアップストリームでリジュームが行われている場合、ホストコントローラに通知されます。</p> <p>0 : ポートは中断されない 1 : ポートは中断</p> <p>(書き込み時) SetPortSuspend</p> <p>ホストコントローラドライバは、このビットに 1 を書き込んで PSS ビットをセットします。0 を書き込んだ場合、セットされません。また、CCS ビットがクリアされているとき、この書き込みにより PSS ビットをセットせずに、CSC ビットをセットします。これにより、未接続ポートが中断処理されたことをドライバに通知します。</p>
1	PES	0	R/W	<p>(読み出し時) PortEnableStatus</p> <p>このビットは、ポートがイネーブルか、ディスエーブルかどうかを示します。過電流条件および切断イベント、パワーオフスイッチ、または babble のような動作上のバスエラーを検出すると、ルートハブ*はこのビットをクリアできます。この変更により、PESB ビットがセットされます。ホストコントローラドライバは、SetPortEnable に書き込みしてこのビットをセットし、または ClearPortEnable に書き込んでこのビットをクリアします。CCS ビットをクリアすると、このビットはセットできません。また、PRSC ビットをセットするポートリセットの完了時、または PSSC ビットをセットするポートサスペンドの完了時にも、このビットはセットされます。</p> <p>0 : ポートはディスエーブル 1 : ポートはイネーブル</p> <p>(書き込み時) SetPortEnable</p> <p>ホストコントローラドライバは、1 を書き込んで PES ビットをセットします。0 を書き込んだ場合、セットされません。CCS ビットをクリアした場合、この書き込みにより PES ビットはセットされず、代わりに、CSC ビットをセットします。これは、未接続ポートをイネーブルにする試みをしたことをドライバに通知します。</p>

ビット	ビット名	初期値	R/W	説明
0	CCS	0	R/W	<p>(読み出し時) CurrentConnectStatus</p> <p>このビットはダウンストリームポートの現在の状態を反映します。</p> <p>【注】取り付けられたデバイスが取り外し可能ではないときには (DeviceRemoveable)、このビットは、常に 1 が読まれます。</p> <p>0 : どのデバイスも未接続 1 : デバイスを接続</p> <p>(書き込み時) ClearPortEnable</p> <p>ホストコントローラドライバは、PES ビットをクリアするために、1 を書き込みます。0 を書き込んだ場合、クリアされません。CCS ビットはどの書き込みによっても影響を受けません。</p>

【注】 * 本 LSI では 1 ポートのみでの入力対応のためルートハブ機能はサポートしていませんが、内部回路構成はルートハブ+1 ポートとなっております。このため、レジスタ構成等にルートハブの概念が入っております。

19.4 データ格納フォーマット

19.4.1 転送データの格納フォーマット

USB ホストコントローラのデータ格納フォーマット (エンディアン) は、MD5 端子による CPU のエンディアン指定と連動しています。したがって、CPU をビッグエンディアンで動作させる場合は、USB ホストコントローラの転送データもビッグエンディアンとして扱ってください。また、CPU をリトルエンディアンで動作させる場合は、USB ホストコントローラの転送データもリトルエンディアンとして扱ってください。

19.4.2 ディスクリプタの格納フォーマット

USB ホストコントローラの各転送トランザクションを定義する ED (Endpoint Descriptor) および TD (Transfer Descriptor) は、各 Dword がメモリのロングワード境界 (アドレス $4n \sim 4n+3$) と符号するように配置してください。なお、この場合はエンディアンの違いは影響しません。

19.5 使用上の注意事項

19.5.1 リセット制御の制約

USB ステートが Operational 状態でソフトウェアリセット (HCSR レジスタの HCR ビットに 1 をセット) の発行、もしくは、USB ステートを Operational 状態から Reset 状態 (HCTLR レジスタの HCFS1/0 ビットに 00 を設定) へ遷移させるときは、下記の手順で行ってください。

- 各リスト処理を中止する。(HCTLR レジスタの BLE ビット、CLE ビット、IE ビット、PLE ビットを 0 にクリアする)
- USB ホストコントローラからの SOF 割り込みを 2 回待つ。(HISR レジスタの SF ビット割り込み)

19. USB ホストコントローラ (USBH)

3. HCTLRレジスタのHCFS1/0ビットを11に設定して、USBステートをSuspend状態にする。
4. 所望のリセット処理を実行する。

19.5.2 ロースピードに関して

本 LSI 搭載の USB ホストコントローラはロースピードはサポートしていません。

19.5.3 共有メモリへのアクセス動作に関する制約 1

- 内容

ホストがショートパケットを受信完了した後、共有メモリに対してステータスライトバック書き込みを行います。このとき、ホストのアクセスする共有メモリが他のメモリアccessによりバスビジー状態になっており、同一フレーム (フレーム長は 1ms) 内にホストからのアクセスができない場合、ステータスライトバックが正しく行われません (図 19.1 参照)。

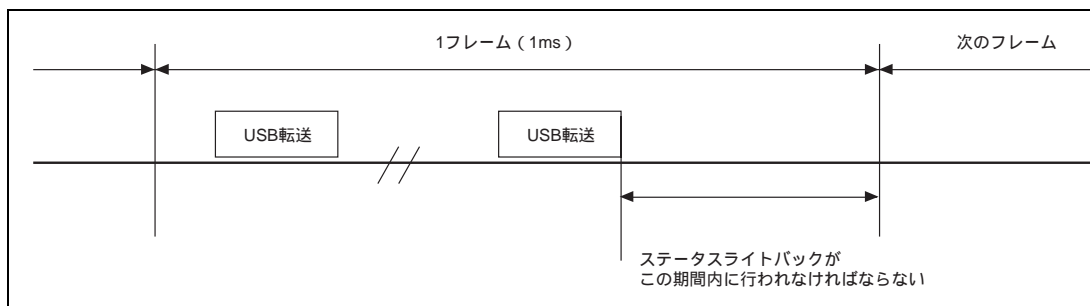


図 19.1 ステータスライトバックの実施タイミング

- 発生した場合の動作

転送が正しく完了しているにもかかわらず、ホストの共有メモリ上は転送が完了していない状態となります。その結果、ホストは不正な転送要求を再度デバイスに対して発行します。

- 発生条件

次の場合に発生します。ショートパケット受信以外では発生しません。

1. フレーム中の最後のパケット かつ
2. それがショートパケット受信 かつ
3. ステータスライトバック書き込みが同一フレーム内にできないとき

- 対応策

本不具合は、8 バイト分のステータスライトバックが同一フレーム内に完了しないときに発生します。よって、

他のバス使用権を持つモジュールの占有時間の調整を行い、17 μ s 以内にホストから必ず共有メモリアクセスが行えるように設定してください。

【注】 17 μ s は OHCI 規格書に記載されているフレーム最後のアイドル期間 210 ビット (=17.5 μ s) から算出しています。

19.5.4 共有メモリへのアクセス動作に関する制約 2

- 内容

アイソクロナスアウト転送時、ホストはアイソクロナス転送に必要な送信データを 12Mbps のレートで共有メモリから読み込みます。このとき、ホストのアクセスする共有メモリが他のメモリアクセスによりバスビジー状態になっており、送信データの読み込みができない場合、それ以降のメモリアクセスが行われなくなります。その結果、まだ未処理の TD*が残っていても転送が開始されません。

【注】 * TD:OHCI 規格書で定義された転送ディスクリプタ (Transfer Descriptor)。転送に必要な情報が格納されている。

- 発生した場合の動作

それ以降のすべての転送が中断します。

- 発生条件

次の場合に発生します。アイソクロナスアウト転送以外では発生しません。

1. アイソクロナスアウト転送パケットを送信中 かつ
2. 12Mbpsのレートで送信データが共有メモリから読み込めないとき

- 対応策

本不具合は、アイソクロナスアウト転送パケットの送信中に、送信データが 12Mbps のレートで連続して読み込めないときに発生します。ここで、ホストのデータアクセス仕様により送信を開始してから次の 16 バイト分の送信データが必要になるまでの時間は、12MHz 換算で 128 クロック期間 (= 10.7 μ s) となります。よって、アイソクロナスアウト転送を行う場合は、他のバス使用権を持つモジュールの占有時間の調整を行い、10 μ s 以内にホストから必ず共有メモリアクセスが行えるように設定してください。

19.5.5 周辺クロックについて

本 USB ホストコントローラを使用する際は、本 LSI の周辺モジュールで使用する周辺クロック (P) の周波数を 13MHz より大きい周波数で使用してください。周辺クロックの周波数が 13MHz 以下では正常に動作しません。周辺クロックの詳細な設定方法については「第 11 章 クロックパルス発振器 (CPG)」を参照してください。

19.5.6 Uメモリ、およびX/Yメモリへのアクセスについて

本 LSI 内蔵の U メモリ、および X/Y メモリへアクセスする場合、物理アドレスによる I バスからのアクセスを行う方法で実施してください。論理アドレスによる L バスからのアクセスを行う方法では正常に動作しません。

19.5.7 過電流検出機能について

本 USB ホストにおいて、過電流検出 / 保護回路を外付けし、USBOVC 端子から検出結果 (Low アクティブ) を入力して過電流検出機能を使用する場合は、HRDRA レジスタ (H'A4490048 番地) の NOCP ビット (12bit) を "0" に設定し、同レジスタの OCPM ビット (11bit) を "1" に設定して、HRPSR レジスタ (H'A4490054 番地) の POCl (3bit) に USBOVC 端子からの検出結果を反映させて、ご使用ください。

19.5.8 USB Bus Reset の発行について

レジューム (サスペンド) 動作を行う場合に使用する USB Bus Reset は、必ず HRPSR レジスタ (H'A4490054 番地) の PRS ビット (4bit) による Port Reset を使用してください。

20. USB ファンクションコントローラ (USBF)

本 LSI は、USB ファンクションコントローラ (USBF) を内蔵しています。

20.1 特長

- USB2.0に対応したUDC (USB Device Controller) を内蔵、USBプロトコルを自動処理。エンドポイント0に対するUSB標準コマンドを自動処理 (一部のコマンドとクラス/ベンダコマンドはファームウェアでデコードし、処理する必要があります。)
- 転送スピード : フルスピード

USB ホストが使用するエンドポイント番号と本 USB ファンクションコントローラが提供するエンドポイント FIFO 番号 (転送方式、転送方向は固定) の対応をエンドポイント情報レジスタ (EPIRn0~5) で設定することで任意のエンドポイント構成が可能です。

エンドポイント名	名称 記号	エンド ポイント FIFO 番号	転送タイプ	最大 パケット サイズ	FIFO バッファ 容量 (バイト)	DMA 転送	用途
エンドポイント 0	EP0s	H'00	セットアップ	8	8	-	Enumration
	EP0i	H'00	コントロールイン	64	64	-	
	EP0o	H'00	コントロールアウト	64	64	-	
エンドポイント 1	EP1	H'01	インタラプトイン	16	16	-	モデム、 パケット
エンドポイント 2i	EP2i	H'02	バルクイン	64	128	可能	
エンドポイント 2o	EP2o	H'03	バルクアウト	64	128	可能	
エンドポイント 3i	EP3i	H'05	アイソクロナスイン	64	128	-	AT コマンド 制御、音声
エンドポイント 3o	EP3o	H'06	アイソクロナスアウト	60	120	-	
エンドポイント 4	EP4	H'07	インタラプトイン	16	16	-	SD/MMC メモ리카ード
エンドポイント 5	EP5	H'08	バルクイン	32	64	可能	
エンドポイント 6	EP6	H'09	バルクアウト	32	64	可能	

20. USB ファンクションコントローラ (USBF)

- 割り込み要求：USB送受信に必要な各種割り込み信号を生成
- クロック：EXCPGによる内部システムクロック / 外部入力 (48MHz) 選択可能
- 低消費電力モード
USBケーブル切断時、UDC内部クロック停止による低消費電力化が可能
サスペンド状態へ自動遷移 / 自動復帰
- パワーモード：セルフパワー、バスパワー
- エンドポイントの構成：任意のエンドポイント構成が設定可能です (Bluetooth規格対応のエンドポイント構成も可能)。

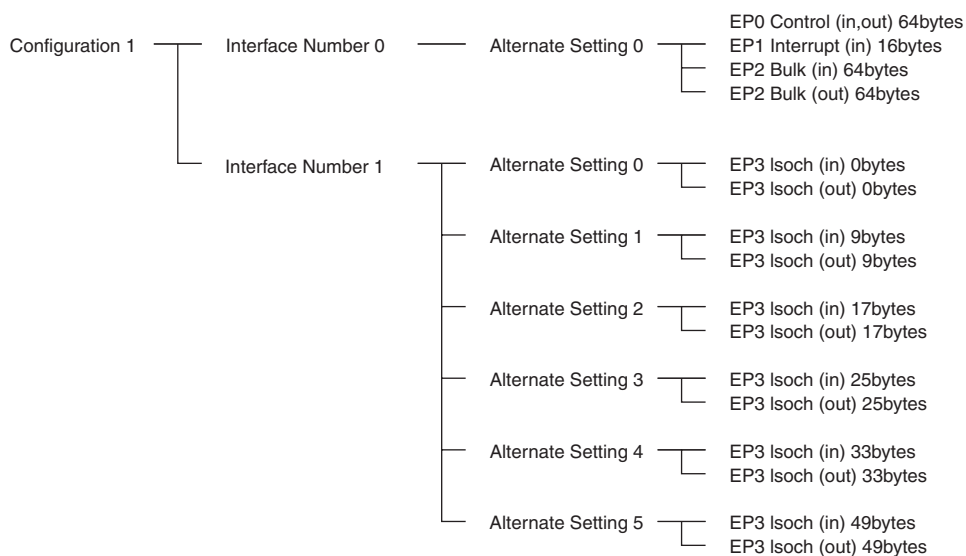


図 20.1 に USBF のブロック図を示します。

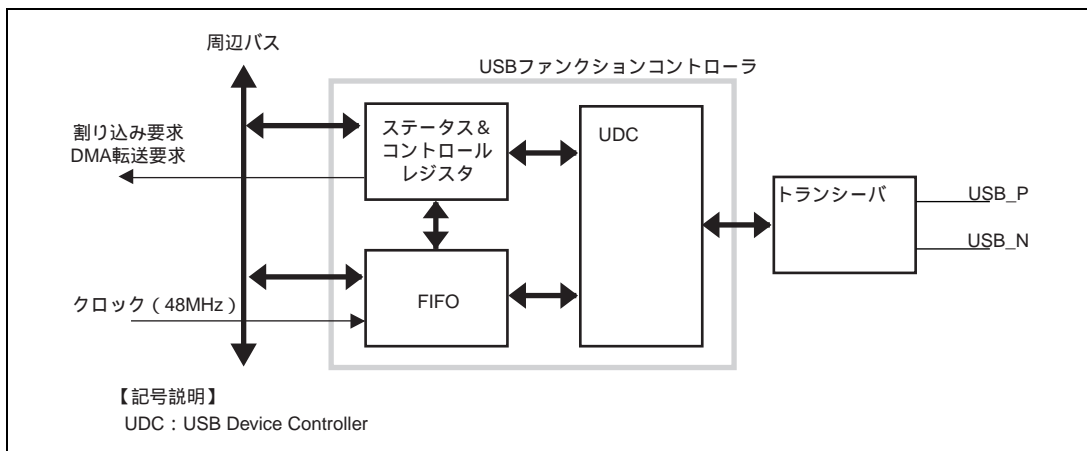


図 20.1 USB ファンクションコントローラのブロック図

20.2 入出力端子

USBF の端子構成を表 20.1 に示します。

各端子の詳細な設定方法については、「第 18 章 USB ピンマルチプレクスコントローラ (USBPM)」を参照してください。

表 20.1 端子構成

端子名	入出力	機能
UCLK	入力	外部クロック端子 外部から直接 48MHz クロックを入力するための端子
USB_PWR_EN/ USB_PULLUP	出力	パワーマネージメント端子 / プルアップ制御端子 電源投入許可制御 (USBH 用) / プルアップ制御出力 (USBF 用)
USB_OVR_CRNT/ USB_VBUS	入力	オーバercurrent端子 / VBUS 端子 オーバercurrent検出 (USBH 用) / USB ケーブル接続モニタ (USBF 用)
USB_P	入出力	P 端子 D+ ポートトランシーバ端子
USB_N	入出力	N 端子 D- ポートトランシーバ端子

20.3 レジスタの説明

USBF には、以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- 割り込みフラグレジスタ0 (IFR0)
- 割り込み選択レジスタ0 (ISR0)
- 割り込みイネーブルレジスタ0 (IER0)
- EP0iデータレジスタ (EPDR0i)
- EP0oデータレジスタ (EPDR0o)
- EP0sデータレジスタ (EPDR0s)
- EP1データレジスタ (EPDR1)
- EP2iデータレジスタ (EPDR2i)
- EP2oデータレジスタ (EPDR2o)
- EP3iデータレジスタ (EPDR3i)
- EP3oデータレジスタ (EPDR3o)
- EP4データレジスタ (EPDR4)
- EP5データレジスタ (EPDR5)
- EP6データレジスタ (EPDR6)
- EP0o受信データサイズレジスタ (EPSZ0o)
- EP2o受信データサイズレジスタ (EPSZ2o)
- EP3o受信データサイズレジスタ (EPSZ3o)
- EP6受信データサイズレジスタ (EPSZ6)
- トリガレジスタ (TRG)
- データステータスレジスタ (DASTS)
- FIFOクリアレジスタ (FCLR)
- DMA転送設定レジスタ (DMA)
- エンドポイントストールレジスタ (EPSTL)
- コンフィグレーションバリュージェスタ (CVR)
- タイムスタンプレジスタ (TSR)
- コントロールレジスタ (CTLR)
- エンドポイント情報レジスタ (EPIRn0~5)

20.3.1 割り込みフラグレジスタ 0 (IFR0)

EP0i、EP0o、EP1、EP2i、EP2o、EP3i、EP3o、EP4～6、およびバスリセット、セットアップコマンド受信、VBUS、SUS/RES、SOF、SETC、SETI 割り込みフラグレジスタです。各フラグが 1 にセットされ割り込みイネーブルレジスタ 0 (IER0) の対応するビットで割り込みが許可されていれば、割り込み選択レジスタ 0 (ISR0) の対応ビットにより割り当てられた INT 端子から割り込みが出力されます。割り込み要求と割り込み信号の設定は、ISR0 で設定します。フラグのクリアは 0 を書き込むことで行います。1 書き込みは無効で、何も行われません。

ビット	ビット名	初期値	R/W	説明
31		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
30	BRST	0	R/W	バスリセット 【セット条件】 USB バス上でバスリセット信号を検出したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
29	SETUP TS	0	R/W	セットアップコマンド受信完了 【セット条件】 ファンクションでコマンドデコードを行う 8 バイトのデータが正常に受信され、ファンクションからホストに ACK ハンドシェイクを返したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
28	VBUSMN	0	R* ²	USB 接続ステータス USB_VBUS 端子の状態をモニタするステータスビットです。USB_VBUS 端子の状態を反映します。
27	VBUSF	0	R/W	USB 切断検出 【セット条件】 ファンクションが USB バスに接続されたとき、および切断されたとき 【注】接続 / 切断の検出には本モジュールの USB_VBUS 端子を使用します。 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説明
26	SURSS	0	R* ¹	Suspend/Resume ステータス バスの状態を示すステータスビットです。 1: サスペンド状態 0: 通常状態
25	SURES	0	R/W	Suspend/Resume 検出 【セット条件】 バスが通常状態からサスペンド状態、またはサスペンド状態から通常状態へ遷移したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
24	CFDN	0	R/W	エンドポイント情報ロード終了 【セット条件】 EPIR レジスタに書き込んだエンドポイント情報が本モジュール内で設定完了したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき 【注】本モジュールはエンドポイント情報の設定が完了した後、USB ファンクションコントローラとして正常に動作します。
23	SOF	0	R/W	SOF パケット 【セット条件】 SOF パケットを検出したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
22	SETC	0	R/W	Set Configuration コマンド検出 【セット条件】 Set Configuration コマンドを検出したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
21	SETI	0	R/W	Set Interface コマンド検出 【セット条件】 Set Interface コマンドを検出したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説 明
20	EP6 FULL	0	R* ¹	<p>EP6 (Bulk out) FIFO フル</p> <p>【セット条件】</p> <p>EP6 の FIFO は 2 面構成になっており、1 面以上の FIFO が一杯のとき</p> <p>【クリア条件】</p> <p>(1) リセット時</p> <p>(2) 2 面とも空きになったとき</p> <p>【注】ステータスビットのためクリアはできません</p>
19	EP5 EMPTY	1	R* ¹	<p>EP5 (Bulk in) FIFO エンプティ</p> <p>【セット条件】</p> <p>(1) リセット時</p> <p>(2) EP5 の FIFO は 2 面構成になっており、1 面以上の FIFO が空のとき</p> <p>【クリア条件】</p> <p>FIFO が 2 面とも空でなくなったとき</p> <p>【注】ステータスビットのためクリアはできません</p>
18	EP5 TR	0	R/W	<p>EP5 (Bulk in) 転送リクエスト</p> <p>【セット条件】</p> <p>EP5 にホストから IN トークンが発行され、2 面とも FIFO が空のとき</p> <p>【クリア条件】</p> <p>(1) リセット時</p> <p>(2) CPU で 0 を書き込んだとき</p>
17	EP4 TR	0	R/W	<p>EP4 (Int) 転送リクエスト</p> <p>【セット条件】</p> <p>EP4 にホストから IN トークンが発行され、FIFO が空のとき</p> <p>【クリア条件】</p> <p>(1) リセット時</p> <p>(2) CPU で 0 を書き込んだとき</p>
16	EP4 TS	0	R/W	<p>EP4 (Int) 送信完了</p> <p>【セット条件】</p> <p>EP4 にホストに送信するデータを書き込んだ後、ファンクションからホストに正常にデータが転送され、ACK ハンドシェイクが返ってきたとき</p> <p>【クリア条件】</p> <p>(1) リセット時</p> <p>(2) CPU で 0 を書き込んだとき</p>
15、14		すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説明
13	EP3o TF	0	R/W	<p>EP3o (Iso out) 異常受信</p> <p>EP3o の FIFO 状態を示すフラグです。ホストからのデータ受信が終了し、次の SOF パケット受信後に読み出し可能となった FIFO 面の状態を示します。</p> <p>【セット条件】 ホストからの転送データを EP3o で異常受信 (パケットエラー) したとき</p> <p>【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき</p>
12	EP3o TS	0	R/W	<p>EP3o (Iso out) 正常受信</p> <p>EP3o の FIFO 状態を示すフラグです。ホストからのデータ受信が終了し、次の SOF パケット受信後に読み出し可能となった FIFO 面の状態を示します。</p> <p>【セット条件】 ホストからの転送データを EP3o で正常受信したとき</p> <p>【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき</p>
11	EP3i TR	0	R/W	<p>EP3i (Iso in) 送信リクエスト</p> <p>【セット条件】 EP3i にホストから IN トークンが発行されたときに送信される FIFO が空のとき</p> <p>【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき</p>
10	EP3i TS	0	R/W	<p>EP3i (Iso in) 正常送信</p> <p>EP3i の FIFO 状態を示すフラグです。</p> <p>SOF パケット受信後、自動的に FIFO の面が切り替わりますが、このとき CPU からの書き込みが可能な FIFO 面は、1 つ前 (SOF 受信前) のフレームでホストにデータ送信を行った面です。本ビットはこの 1 つ前の送信状態を示します。</p> <p>【セット条件】 1 つ前のフレームで送信が正常に行われたとき</p> <p>【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき</p>
9、8		すべて 0	R	<p>リザーブビット</p> <p>読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。</p>

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説 明
7	EP2o FULL	0	R* ¹	EP2o (Bulk out) FIFO フル 【セット条件】 EP2o の FIFO は 2 面構成になっており、1 面以上の FIFO が一杯のとき 【クリア条件】 (1) リセット時 (2) 2 面とも空きになったとき 【注】ステータスビットのためクリアはできません
6	EP2i EMPTY	1	R* ¹	EP2i (Bulk in) FIFO エンプティ 【セット条件】 (1) リセット時 (2) EP2i の FIFO は 2 面構成になっており、1 面以上の FIFO が空のとき 【クリア条件】 FIFO が 2 面とも空でなくなったとき
5	EP2i TR	0	R/W	EP2i (Bulk in) 転送リクエスト 【セット条件】 EP2i にホストから IN トークンが発行され、2 面とも FIFO が空のとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
4	EP1 TR	0	R/W	EP1 (Int) 転送リクエスト 【セット条件】 EP1 にホストから IN トークンが発行され、FIFO が空のとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
3	EP1 TS	0	R/W	EP1 (Int) 送信完了 【セット条件】 EP1 にホストに送信するデータを書き込んだ後、ファンクションからホストに正常にデータが転送され、ACK ハンドシェイクが返ってきたとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説明
2	EP0o TS	0	R/W	EP0o 受信完了 【セット条件】 ホストから EP0o へ正常にデータが受信されファンクションからホストに ACK ハンドシェイクを返したとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
1	EP0i TR	0	R/W	EP0i 転送リクエスト 【セット条件】 EP0i にホストから IN トークンが発行され、FIFO が空のとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき
0	EP0i TS	0	R/W	EP0i 送信完了 【セット条件】 ホストに転送するデータを EP0i に書き込んだ後、ファンクションからホストに正常にデータが送信され、ACK ハンドシェイクが返ってきたとき 【クリア条件】 (1) リセット時 (2) CPU で 0 を書き込んだとき

- 【注】 *1 EP2o、6 の FIFO フルおよび EP2i、5 の FIFO エンプティビットは FIFO 状態を示すステータスビットなのでクリアはできません。
- *2 USB 接続ステータス (ビット 28) は USB_VBUS 端子の状態を示すビットなのでクリアはできません。割り込みの出力も行いません。

20.3.2 割り込み選択レジスタ 0 (ISR0)

IFR0 の各割り込み要求を出力する割り込み要求端子を設定します。USBF には 2 本の割り込み要求信号 (INT0N、INT1N) があり、割り込みコントローラ (INTC) に割り込み要求信号を出力します。対応するビットに 0 をセットすると INT0N 割り込み要求信号が出力され、1 をセットすると INT1N 割り込み要求信号が出力されます。INT0N 割り込み要求信号は割り込みコントローラ (INTC) の USBF10 に、INT1N 割り込み要求信号は USBF11 にそれぞれ対応しています。

初期値では、割り込みフラグレジスタの各割り込み要因は INT0N 割り込み要求信号から出力されます。

ビット	ビット名	初期値	R/W	説明
31		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
30	BRST IS	0	R/W	BRST 割り込み選択 0 : BRST (バスリセット) 割り込み要求を INT0N から出力 1 : BRST (バスリセット) 割り込み要求を INT1N から出力
29	SETUP TS IS	0	R/W	SETUP 割り込み選択 0 : SETUP (セットアップデータ受信完了) 割り込み要求を INT0N から出力 1 : SETUP (セットアップデータ受信完了) 割り込み要求を INT1N から出力
28		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
27	VBUSF IS	0	R/W	VBUSF 割り込み選択 0 : VBUSF 割り込み要求を INT0N から出力 1 : VBUSF 割り込み要求を INT1N から出力
26		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
25	SURES IS	0	R/W	SURES 割り込み選択 0 : SURES 割り込み要求を INT0N から出力 1 : SURES 割り込み要求を INT1N から出力
24	CFDN IS	0	R/W	CFDN 割り込み選択 0 : CFDN 割り込み要求を INT0N から出力 1 : CFDN 割り込み要求を INT1N から出力
23	SOF IS	0	R/W	SOF 割り込み選択 0 : SOF 割り込み要求を INT0N から出力 1 : SOF 割り込み要求を INT1N から出力

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説明
22	SETC IS	0	R/W	SETC 割り込み選択 0 : SETC 割り込み要求を INT0N から出力 1 : SETC 割り込み要求を INT1N から出力
21	SETI IS	0	R/W	SETI 割り込み選択 0 : SETI 割り込み要求を INT0N から出力 1 : SETI 割り込み要求を INT1N から出力
20	EP6 FULL IS	0	R/W	EP6 FULL 割り込み選択 0 : EP6 FULL 割り込み要求を INT0N から出力 1 : EP6 FULL 割り込み要求を INT1N から出力
19	EP5 EMPTY IS	0	R/W	EP5 EMPTY 割り込み選択 0 : EP5 EMPTY 割り込み要求を INT0N から出力 1 : EP5 EMPTY 割り込み要求を INT1N から出力
18	EP5 TR IS	0	R/W	EP5 TR 割り込み選択 0 : EP5 TR (転送リクエスト) 割り込み要求を INT0N から出力 1 : EP5 TR (転送リクエスト) 割り込み要求を INT1N から出力
17	EP4 TR IS	0	R/W	EP4 TR 割り込み選択 0 : EP4 TR (転送リクエスト) 割り込み要求を INT0N から出力 1 : EP4 TR (転送リクエスト) 割り込み要求を INT1N から出力
16	EP4 TS IS	0	R/W	EP4 TS 割り込み選択 0 : EP4 TS (受信完了) 割り込み要求を INT0N から出力 1 : EP4 TS (受信完了) 割り込み要求を INT1N から出力
15、14		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
13	EP3o TF IS	0	R/W	EP3o TF 割り込み選択 0 : EP3o TF 割り込み要求を INT0N から出力 1 : EP3o TF 割り込み要求を INT1N から出力
12	EP3o TS IS	0	R/W	EP3o TS 割り込み選択 0 : EP3o TS (受信完了) 割り込み要求を INT0N から出力 1 : EP3o TS (受信完了) 割り込み要求を INT1N から出力
11	EP3i TR IS	0	R/W	EP3i TR 割り込み選択 0 : EP3i TR (転送リクエスト) 割り込み要求を INT0N から出力 1 : EP3i TR (転送リクエスト) 割り込み要求を INT1N から出力
10	EP3i TS IS	0	R/W	EP3i TS 割り込み選択 0 : EP3i TS (受信完了) 割り込み要求を INT0N から出力 1 : EP3i TS (受信完了) 割り込み要求を INT1N から出力

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説 明
9、8		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
7	EP2o FULL IS	0	R/W	EP2o FULL 割り込み選択 0 : EP2o FULL 割り込み要求を INT0N から出力 1 : EP2o FULL 割り込み要求を INT1N から出力
6	EP2i EMPTY IS	0	R/W	EP2i EMPTY 割り込み選択 0 : EP2i EMPTY 割り込み要求を INT0N から出力 1 : EP2i EMPTY 割り込み要求を INT1N から出力
5	EP2i TR IS	0	R/W	EP2i TR 割り込み選択 0 : EP2i TR (転送リクエスト) 割り込み要求を INT0N から出力 1 : EP2i TR (転送リクエスト) 割り込み要求を INT1N から出力
4	EP1 TR IS	0	R/W	EP1 TR 割り込み選択 0 : EP1 TR (転送リクエスト) 割り込み要求を INT0N から出力 1 : EP1 TR (転送リクエスト) 割り込み要求を INT1N から出力
3	EP1 TS IS	0	R/W	EP1 TS 割り込み選択 0 : EP1 TS (受信完了) 割り込み要求を INT0N から出力 1 : EP1 TS (受信完了) 割り込み要求を INT1N から出力
2	EP0o TS IS	0	R/W	EP0o TS 割り込み選択 0 : EP0o TS (受信完了) 割り込み要求を INT0N から出力 1 : EP0o TS (受信完了) 割り込み要求を INT1N から出力
1	EP0i TR IS	0	R/W	EP0i TR 割り込み選択 0 : EP0i TR (転送リクエスト) 割り込み要求を INT0N から出力 1 : EP0i TR (転送リクエスト) 割り込み要求を INT1N から出力
0	EP0i TS IS	0	R/W	EP0i TS 割り込み選択 0 : EP0i TS (受信完了) 割り込み要求を INT0N から出力 1 : EP0i TS (受信完了) 割り込み要求を INT1N から出力

20.3.3 割り込みイネーブルレジスタ 0 (IER0)

IFR0 の各割り込みを許可 / 禁止するレジスタです。各割り込みに対応するビットに 1 が設定されていて、IFR0 レジスタの該当割り込みフラグが 1 にセットされると、ISR0 で設定された INT0N または INT1N 端子がローアサートされ、割り込み要求が発行されます。

ビット	ビット名	初期値	R/W	説明
31		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
30	BRST IE	0	R/W	BRST 割り込みイネーブル 0 : BRST (バスリセット) の割り込み要求を禁止 1 : BRST (バスリセット) の割り込み要求を許可
29	SETUP TS IE	0	R/W	SETUP TS 割り込みイネーブル 0 : SETUP TS (セットアップデータ受信完了) の割り込み要求を禁止 1 : SETUP TS (セットアップデータ受信完了) の割り込み要求を許可
28		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
27	VBUSF IE	0	R/W	VBUSF 割り込みイネーブル 0 : VBUSF の割り込み要求を禁止 1 : VBUSF の割り込み要求を許可
26		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
25	SURES IE	0	R/W	SURES 割り込みイネーブル 0 : SURES の割り込み要求を禁止 1 : SURES の割り込み要求を許可
24	CFDN IE	0	R/W	CFDN 割り込みイネーブル 0 : CFDN の割り込み要求を禁止 1 : CFDN の割り込み要求を許可
23	SOF IE	0	R/W	SOF 割り込みイネーブル 0 : SOF の割り込み要求を禁止 1 : SOF の割り込み要求を許可
22	SETC IE	0	R/W	SETC 割り込みイネーブル 0 : SETC の割り込み要求を禁止 1 : SETC の割り込み要求を許可

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説明
21	SET1 IE	0	R/W	SET1 割り込みイネーブル 0 : SET1 の割り込み要求を禁止 1 : SET1 の割り込み要求を許可
20	EP6 FULL IE	0	R/W	EP6 FULL 割り込みイネーブル 0 : EP6 FULL の割り込み要求を禁止 1 : EP6 FULL の割り込み要求を許可
19	EP5 EMPTY IE	0	R/W	EP5 EMPTY 割り込みイネーブル 0 : EP5 EMPTY の割り込み要求を禁止 1 : EP5 EMPTY の割り込み要求を許可
18	EP5 TR IE	0	R/W	EP5 TR 割り込みイネーブル 0 : EP5 TR (転送リクエスト) の割り込み要求を禁止 1 : EP5 TR (転送リクエスト) の割り込み要求を許可
17	EP4 TR IE	0	R/W	EP4 TR 割り込みイネーブル 0 : EP4 TR (転送リクエスト) の割り込み要求を禁止 1 : EP4 TR (転送リクエスト) の割り込み要求を許可
16	EP4 TS IE	0	R/W	EP4 TS 割り込みイネーブル 0 : EP4 TS (受信完了) の割り込み要求を禁止 1 : EP4 TS (受信完了) の割り込み要求を許可
15、14		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
13	EP3o TF IE	0	R	EP3o TF 割り込みイネーブル 0 : EP3o TF の割り込み要求を禁止 1 : EP3o TF の割り込み要求を許可
12	EP3o TS IE	0	R	EP3o TS 割り込みイネーブル 0 : EP3o TS (受信完了) の割り込み要求を禁止 1 : EP3o TS (受信完了) の割り込み要求を許可
11	EP3i TR IE	0	R/W	EP3i TR 割り込みイネーブル 0 : EP3i TR (転送リクエスト) の割り込み要求を禁止 1 : EP3i TR (転送リクエスト) の割り込み要求を許可
10	EP3i TS IE	0	R/W	EP3i TS 割り込みイネーブル 0 : EP3i TS (受信完了) の割り込み要求を禁止 1 : EP3i TS (受信完了) の割り込み要求を許可
9、8		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説明
7	EP2o FULL IE	0	R/W	EP2o FULL 割り込みイネーブル 0 : EP2o FULL の割り込み要求を禁止 1 : EP2o FULL の割り込み要求を許可
6	EP2i EMPTY IE	0	R/W	EP2i EMPTY 割り込みイネーブル 0 : EP2i EMPTY の割り込み要求を禁止 1 : EP2i EMPTY の割り込み要求を許可
5	EP2i TR IE	0	R/W	EP2i TR 割り込みイネーブル 0 : EP2i TR (転送リクエスト) の割り込み要求を禁止 1 : EP2i TR (転送リクエスト) の割り込み要求を許可
4	EP1 TR IE	0	R/W	EP1 TR 割り込みイネーブル 0 : EP1 TR (転送リクエスト) の割り込み要求を禁止 1 : EP1 TR (転送リクエスト) の割り込み要求を許可
3	EP1 TS IE	0	R/W	EP1 TS 割り込みイネーブル 0 : EP1 TS (受信完了) の割り込み要求を禁止 1 : EP1 TS (受信完了) の割り込み要求を許可
2	EP0o TS IE	0	R/W	EP0o TS 割り込みイネーブル 0 : EP0o TS (受信完了) の割り込み要求を禁止 1 : EP0o TS (受信完了) の割り込み要求を許可
1	EP0i TR IE	0	R/W	EP0i TR 割り込みイネーブル 0 : EP0i TR (転送リクエスト) の割り込み要求を禁止 1 : EP0i TR (転送リクエスト) の割り込み要求を許可
0	EP0i TS IE	0	R/W	EP0i TS 割り込みイネーブル 0 : EP0i TS (受信完了) の割り込み要求を禁止 1 : EP0i TS (受信完了) の割り込み要求を許可

20.3.4 EP0i データレジスタ (EPDR0i)

エンドポイント0の送信用8ビット長64バイトFIFOバッファです。コントロールインに対するパケット分の送信データを保存します。1パケット分のデータを書き込み、TRGのEP0i PKTE ビットをセットすることで送信データが確定します。データを送信した後、ホストからACKハンドシェイクが返ってくると、IFR0のEP0i TS ビットがセットされます。このFIFOバッファは、FCLRのEP0i CLR ビットにより初期化することができます。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	W	コントロールイン転送用のデータレジスタ

20.3.5 EP0o データレジスタ (EPDR0o)

エンドポイント0の受信用8ビット長64バイトFIFOバッファです。セットアップコマンドを除くエンドポイント0の受信データが格納されます。データを正常に受信すると、IFR0のEP0o TS ビットがセットされ、受信バイト数がEPSZ0o レジスタに表示されます。データを読み出した後、TRGのEP0o RDFN ビットをセットすることで、次のパケットを受信可能となります。このFIFOバッファは、FIFOクリアレジスタ(FCLR)のEP0o CLR ビットにより初期化することができます。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	R	コントロールアウト転送用のデータレジスタ

20.3.6 EP0s データレジスタ (EPDR0s)

エンドポイント0のセットアップコマンド専用のデータレジスタです。セットアップステージで送られた8バイトのコマンドデータが格納されます。ただし、マイコン(ファームウェア)で処理するコマンドのみ受信します。本モジュールが自動処理するコマンドを受信した場合は格納されません。

セットアップコマンドは必ず受信する必要があるため、バッファ内にデータが残っている場合でも新しいデータによって上書きされます。つまり、コマンドを読み出している間に次のコマンドの受信が開始された場合、受信が優先となり読み出したデータは無効になります。

【注】EPDR0s レジスタは必ず8バイト単位で読み出してください。途中で読み出しを中止すると次のセットアップで受信したデータが正常に読み出せません。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	R	コントロールアウト転送時のセットアップコマンドが格納されるレジスタ

20.3.7 EP1 データレジスタ (EPDR1)

エンドポイント1の送信用8ビット長16バイトFIFOバッファです。エンドポイント1のインタラプト転送における1パケット分の送信データを保持します。1パケット分のデータを書き込み、TRGのEP1 PKTE ビットをセットすることで送信データが確定します。1パケット分のデータを正常に送信し、ホストからACKハンドシェークを受信するとIFR0のEP1 TS ビットがセットされます。このFIFOバッファは、FCLRのEP1 CLR ビットにより初期化することができます。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	W	エンドポイント1転送用のデータレジスタ

20.3.8 EP2i データレジスタ (EPDR2i)

エンドポイント 2i の送信用 8 ビット長 128 バイト FIFO バッファです。最大パケットサイズの 2 倍の容量を持っていて、2 面構成になっています。この FIFO バッファに送信データを書き込み、TRG の EP2i PKTE ビットをセットすることで 1 パケット分の送信データが確定し、2 面構成のバッファが切り替わります。この FIFO バッファへの送信データは DMA 転送が可能です。この FIFO バッファは FCLR の EP2i CLR ビットにより初期化することができます。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	W	エンドポイント 2i 転送用のデータレジスタ

20.3.9 EP2o データレジスタ (EPDR2o)

エンドポイント 2o の受信用 8 ビット長 128 バイト FIFO バッファです。最大パケットサイズの 2 倍の容量を持っていて、2 面構成になっています。受信バイト数は EPSZ2o レジスタに表示されます。データを読み出した後、TRG の EP2o RDFN ビットに 1 を書き込むことで、読み出した面のバッファが再受信可能になります。この FIFO バッファの受信データは DMA 転送が可能です。この FIFO バッファは、FCLR の EP2o CLR ビットにより初期化することができます。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	R	エンドポイント 2o 転送用のデータレジスタ

20.3.10 EP3i データレジスタ (EPDR3i)

エンドポイント 3i の送信用 8 ビット長 128 バイト FIFO バッファです。最大パケットサイズの 2 倍の容量を持っていて、2 面構成になっています。この FIFO バッファに送信データを書き込んだ後、SOF パケットを受信することで、1 パケット分の送信データが確定し、2 面構成のバッファが切り替わります。この FIFO バッファは、FCLR の EP3i CLR ビットにより初期化することができます。

【注】SOF パケットを受信できなかった場合、SOF マーカ機能により SOF 割り込みは発生しますが、SOF を受信できなかったフレームのデータは無効となります。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	W	エンドポイント 3i 転送用のデータレジスタ

20.3.11 EP3₀ データレジスタ (EPDR3₀)

エンドポイント 3₀ の受信用 8 ビット長 120 バイト FIFO バッファです。最大パケットサイズの 2 倍の容量を持っていて、2 面構成になっています。受信バイト数は EPSZ3₀ レジスタに表示されます。SOF パケットを受信したときに受信データが確定されます。したがって、受信データは次の SOF パケット受信までに必ずすべて読み出してください。次の SOF パケットを受信すると、自動的に FIFO の面が切り替わり、1 つ前のデータは読み出せなくなります。この FIFO バッファは、FCLR の EP3₀ CLR ビットにより初期化することができます。

【注】 SOF パケットを受信できなかった場合、SOF マーカ機能により SOF 割り込みは発生しますが、SOF を受信できなかったフレームのデータは無効となります。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	R	エンドポイント 3 ₀ 転送用のデータレジスタ

20.3.12 EP4 データレジスタ (EPDR4)

エンドポイント 4 送信用 8 ビット長 16 バイト FIFO バッファです。エンドポイント 4 インタラプト転送における 1 パケット分の送信データを保持します。1 パケット分のデータを書き込み、TRG の EP4 PKTE ビットをセットすることで送信データが確定します。1 パケット分のデータを正常に送信し、ホストから ACK ハンドシェイクを受信すると IFR0 の EP4 TS ビットがセットされます。この FIFO バッファは FCLR の EP4 CLR ビットにより初期化することができます。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	W	エンドポイント 4 転送用のデータレジスタ

20.3.13 EP5 データレジスタ (EPDR5)

エンドポイント 5 の送信用 8 ビット長 64 バイト FIFO バッファです。最大パケットサイズの 2 倍の容量を持っていて、2 面構成になっています。この FIFO バッファに送信データを書き込み、TRG の EP5 PKTE ビットをセットすることで 1 パケット分の送信データが確定し、2 面構成のバッファが切り替わります。この FIFO バッファへの送信データは DMA 転送が可能です。この FIFO バッファは、FCLR の EP5 CLR ビットにより初期化することができます。

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	W	エンドポイント 5 転送用のデータレジスタ

20.3.14 EP6 データレジスタ (EPDR6)

エンドポイント 6 の受信用 8 ビット長 64 バイト FIFO バッファです。最大パケットサイズの 2 倍の容量を持っていて、2 面構成になっています。受信バイト数は EPSZ6 レジスタに表示されます。データを読み出した後、TRG の EP6 RDFN ビットに 1 を書き込むことで、読み出した面のバッファが再受信可能になります。この FIFO バッファの受信データは DMA 転送が可能です。この FIFO バッファは FCLR の EP6 CLR ビットにより初期化することができます。

20. USB ファンクションコントローラ (USBF)

ビット	ビット名	初期値	R/W	説明
7~0	D7~D0	不定	R	エンドポイント6転送用のデータレジスタ

20.3.15 EP0o 受信データサイズレジスタ (EPSZ0o)

エンドポイント 0o の受信データサイズレジスタです。ホストから受信したバイト数を示します。

ビット	ビット名	初期値	R/W	説明
7~0	-	すべて0	R	エンドポイント 0o の受信バイト数

20.3.16 EP2o 受信データサイズレジスタ (EPSZ2o)

エンドポイント 2o の受信データサイズレジスタです。ホストから受信したバイト数を示します。エンドポイント 2o の FIFO は 2 面構成になっています。本レジスタに示される受信データサイズは、現在選択されている (CPU で読み出せる) 面のサイズです。

ビット	ビット名	初期値	R/W	説明
7~0	-	すべて0	R	エンドポイント 2o の受信バイト数

20.3.17 EP3o 受信データサイズレジスタ (EPSZ3o)

エンドポイント 3o の受信データサイズレジスタです。ホストから受信したバイト数を示します。エンドポイント 3o の FIFO は 2 面構成になっています。本レジスタに示される受信データサイズは、現在選択されている (CPU で読み出せる) 面のサイズです。

ビット	ビット名	初期値	R/W	説明
7~0	-	すべて0	R	エンドポイント 3o の受信バイト数

20.3.18 EP6 受信データサイズレジスタ (EPSZ6)

エンドポイント 6 の受信データサイズレジスタです。ホストから受信したバイト数を示します。エンドポイント 6 の FIFO は 2 面構成になっています。本レジスタに示される受信データサイズは、現在選択されている (CPU で読み出せる) 面のサイズです。

ビット	ビット名	初期値	R/W	説明
7~0	-	すべて0	R	エンドポイント 6 の受信バイト数

20.3.19 トリガレジスタ (TRG)

EP0s、EP0i、EP0o、EP1、EP2i、EP2o、EP4、EP5、EP6 の各エンドポイントの FIFO に対する送受信許可トリガを与えるワンショットレジスタです。与えるトリガは IN FIFO レジスタに対するパケットイネーブルと OUT FIFO レジスタに対する読み出し完了トリガです。各エンドポイントに対応するビットに 1 書き込みを行うとトリガが与えられます。1 書き込みの後、自動的に 0 になります。

ビット	ビット名	初期値	R/W	説明
15				リザーブビット 書き込み値は常に 0 にしてください。1 を書いた場合の動作の保証はできません。
14	EP6 RDFN		W	EP6 読み出し完了
13	EP5 PKTE		W	EP5 パケットイネーブル
12	EP4 PKTE		W	EP4 パケットイネーブル
11~7		すべて		リザーブビット 書き込み値は常に 0 にしてください。1 を書いた場合の動作の保証はできません。
6	EP2o RDFN		W	EP2o 読み出し完了
5	EP2i PKTE		W	EP2i パケットイネーブル
4	EP1 PKTE		W	EP1 パケットイネーブル
3				リザーブビット 書き込み値は常に 0 にしてください。1 を書いた場合の動作の保証はできません。
2	EP0s RDFN		W	EP0s 読み出し完了
1	EP0o RDFN		W	EP0o 読み出し完了
0	EP0i PKTE		W	EP0i パケットイネーブル

20.3.20 データステータスレジスタ (DASTS)

送信用 FIFO データレジスタ内にデータがあるかどうかを示すレジスタです。送信用 FIFO に書き込んだデータが、TRG の PKTE ビットへ 1 を書き込み、有効になったとき 1 にセットされます。また、有効にしたデータがすべてホストに送信されたとき、0 にクリアされます。エンドポイント 2i、5 の 2 面構成 FIFO の場合、2 面ともデータがなくなったとき 0 にクリアされます。

ビット	ビット名	初期値	R/W	説 明
7		0	R	リザーブビット 読み出すと常に 0 が読み出されます。
6	EP5 DE	0	R	EP5 データタイネーブル
5	EP4 DE	0	R	EP4 データタイネーブル
4		0	R	リザーブビット 読み出すと常に 0 が読み出されます。
3	EP2i DE	0	R	EP2i データタイネーブル
2	EP1 DE	0	R	EP1 データタイネーブル
1		0	R	リザーブビット 読み出すと常に 0 が読み出されます。
0	EP0i DE	0	R	EP0i データタイネーブル

20.3.21 FIFO クリアレジスタ (FCLR)

各エンドポイントの FIFO をクリアするワンショットレジスタです。各ビットに 1 書き込みをすることで、対応する FIFO 内のデータをクリアすることができます。各ビットは 1 書き込み後、自動的に 0 に戻ります。

送信用 FIFO の場合、FIFO にデータを書き込み TRG の PKTE ビットに 1 書き込みしていないデータ、および 1 書き込みによって有効になったデータをクリアすることができます。受信用 FIFO の場合、まだ受信中で確定していないデータ、および受信が完了したデータをクリアすることができます。

EP2i、EP2o、EP3i、EP3o、EP5、EP6 の 2 面構成 FIFO は 2 面ともクリアされます。

また、このクリア指示によって、対応する割り込みフラグはクリアされません。送信中のクリアは行わないでください。

ビット	ビット名	初期値	R/W	説明
15				リザーブビット 書き込む値は常に 0 にしてください。1 を書いた場合の動作の保証はできません。
14	EP6 CLR		W	EP6 クリア
13	EP5 CLR		W	EP5 クリア
12	EP4 CLR		W	EP4 クリア
11				リザーブビット 書き込む値は常に 0 にしてください。1 を書いた場合の動作の保証はできません。
10	EP3o CLR		W	EP3o クリア
9	EP3i CLR		W	EP3i クリア
8、7		すべて		リザーブビット 書き込む値は常に 0 にしてください。1 を書いた場合の動作の保証はできません。
6	EP2o CLR		W	EP2o クリア
5	EP2i CLR		W	EP2i クリア
4	EP1 CLR		W	EP1 クリア
3、2		すべて		リザーブビット 書き込む値は常に 0 にしてください。1 を書いた場合の動作の保証はできません。
1	EP0o CLR		W	EP0o クリア
0	EP0i CLR		W	EP0i クリア

20. USB ファンクションコントローラ (USBF)

20.3.22 DMA 転送設定レジスタ (DMA)

DMA 転送可能なエンドポイント EP2i、EP2o、EP5、EP6 のデータレジスタに対して内蔵 DMAC のデュアルアドレス転送を使用する場合に設定します。

ビット	ビット名	初期値	R/W	説明
7	EP6 DMAE	0	R/W	EP6DMA 許可 EP6 に対する DMA 転送を許可します。
6	EP6 DMAS	0	R/W	EP6DMA 要求選択 EP6 に対する DMA 転送要求出力を選択します。
5	EP5 DMAE	0	R/W	EP5DMA 許可 EP5 に対する DMA 転送を許可します。
4	EP5 DMAS	0	R/W	EP5DMA 要求選択 EP5 に対する DMA 転送要求出力を選択します。
3	EP2o DMAE	0	R/W	EP2o DMA 許可 EP2o に対する DMA 転送を許可します。
2	EP2o DMAS	0	R/W	EP2o DMA 要求選択 EP2o に対する DMA 転送要求出力を選択します。
1	EP2i DMAE	0	R/W	EP2iDMA 許可 EP2i に対する DMA 転送を許可します。
0	EP2i DMAS	0	R/W	EP2iDMA 要求選択 EP2i に対する DMA 転送要求出力を選択します。

20.3.23 エンドポイントストールレジスタ (EPSTL)

各エンドポイントをストールさせるレジスタです。ストールビットに1がセットされたエンドポイントは1を書き込んだ次の転送からホストに対してストールハンドシェイクを返します。

エンドポイント0に対するストールビットは、ファンクションでコマンドデコードを行う8バイトのデータ受信時に自動的に解除され、EP0 STL ビットはクリアされます。また、IFR0 の SETUP TS ビットが1セットされている場合、EP0 STL ビットへの1書き込みは無視されます。詳細動作は「20.6 ストール動作」を参照してください。

ビット	ビット名	初期値	R/W	説明
15~11		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
10	EP6 STL	0	R/W	EP6 ストール EP6 ストールを設定します。
9	EP5 STL	0	R/W	EP5 ストール EP5 ストールを設定します。
8	EP4 STL	0	R/W	EP4 ストール EP4 ストールを設定します。
7		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
6	EP3o STL	0	R/W	EP3o ストール EP3o ストールを設定します。
5	EP3i STL	0	R/W	EP3i ストール EP3i ストールを設定します。
4		0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も常に0にしてください。 1を書いた場合の動作の保証はできません。
3	EP2o STL	0	R/W	EP2o ストール EP2o ストールを設定します。
2	EP2i STL	0	R/W	EP2i ストール EP2i ストールを設定します。
1	EP1 STL	0	R/W	EP1 ストール EP1 ストールを設定します。
0	EP0 STL	0	R/W	EP0 ストール EP0 ストールを設定します。

20. USB ファンクションコントローラ (USBF)

20.3.24 コンフィグレーションバリュレジスタ (CVR)

ホストから「Set Configuration/Set Interface」コマンドを正常に受信したとき、セットされる「Configuration/Interface/Alternate」値が格納されるレジスタです。

ビット	ビット名	初期値	R/W	説明
7	CNFV	0	R	コンフィグレーション値 Set Configuration コマンドを受け取ったときのコンフィグレーション設定値が格納されます。 CNFV の更新は IFR0 レジスタの SETC = 1 セット時です。
6	INTV3	0	R	インタフェース値 Set Interface コマンドを受け取ったときのインタフェース設定値が格納されます。 INTV の更新は IFR0 レジスタの SETI = 1 セット時です。
5	INTV2	0	R	
4	INTV1	0	R	
3	INTV0	0	R	
2	ALTV2	0	R	オルタネート値
1	ALTV1	0	R	Set Interface コマンドを受け取ったときのオルタネート設定値が格納されます。 ALTV の更新は IFR0 レジスタの SETI = 1 セット時です。
0	ALTV0	0	R	

20.3.25 タイムスタンプレジスタ (TSR)

現在のタイムスタンプ値が格納されるレジスタです。タイムスタンプの更新は、IFR0 の SOF ビットに 1 がセットされたときです。

SOF マーカ機能を有効にしたとき、および SOF パケットが破損した場合のタイムスタンプは前回と同じ値になります。

ビット	ビット名	初期値	R/W	説明
15~11		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。
10	D10	0	R	タイムスタンプデータ
9	D9	0	R	
8	D8	0	R	
7	D7	0	R	
6	D6	0	R	
5	D5	0	R	
4	D4	0	R	
3	D3	0	R	
2	D2	0	R	
1	D1	0	R	
0	D0	0	R	

20.3.26 コントロールレジスタ (CTRL)

USB ファンクションコントローラの各種設定を行うレジスタです。

アイソクロナス転送を使用する場合、SOF マーカ機能は必ずイネーブルにしてください。

ビット	ビット名	初期値	R/W	説 明
7、6		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
5	PULLUPE	0	R/W	プルアップイネーブル USB_PULLUP 端子の出力値を制御します。
4	RWUPS	0	R	リモートウェイクアップステータス ホストからのリモートウェイクアップの禁止 / 許可を示すステータスビットです。Set Feature/Clear Feature リクエストによる Device Remote Wakeup でホストからのリモートウェイクアップが禁止されている場合 0 を示し、許可されると 1 を示します。
3	RSME	0	R/W	リジュームイネーブル サスペンド状態を解除 (リモートウェイクアップを実行) するビットです。 1 を書き込むとリジューム要求が発行されます。使用する場合は、最低 12MHz で 1 クロック以上 1 を保持し再び 0 にクリアしてください。
2	PWMD	0	R/W	パワーモード選択 0: セルフパワーモードで使用 1: バスパワーモードで使用
1	ASCE	0	R/W	自動ストールクリアイネーブル 本ビットに 1 を設定すると、ストールハンドシェイクをホストに返信した後、返信したエンドポイントのストール設定ビット (EPSTLR/EPxSTL) を自動的にクリアします。全エンドポイント共通なのでエンドポイントごとの制御はできません。0 を設定した場合、各エンドポイントのストール設定ビットのクリアはソフトで行ってください。 なお、本ビットをイネーブルにする場合は、必ず EPSTL レジスタの 1 セットより先にイネーブルにしてください。
0	SOFME	0	R/W	SOF マーカ機能イネーブル SOF マーカ機能を制御します。 1 を設定すると SOF パケットが破損しても、約 1ms ごとに SOF 割り込みフラグを 1 にセットします。1 を設定する場合は、IFR0 の SOF 割り込みフラグを正常に検出してから行ってください。また、サスペンド検出時には必ず 0 クリアし、その後の 1 セットも SOF を検出してから行ってください。

20.3.27 エンドポイント情報レジスタ (EPIRn0~5)

各エンドポイントの構成情報を設定するレジスタです。これらの情報は1エンドポイントに対して6バイト必要です。データの書き込みはエンドポイント0から順に行ってください。また、6(バイト)×19(エンドポイント) = 114バイト以上書き込まないでください。本情報は、パワーオンリセット時に1回書き込んでください。それ以降は書き込まないでください。

以下に1エンドポイント分の書き込みデータを説明します。本レジスタは同一アドレスに順番に書き込むため、レジスタは1つですが説明上登録番号n(nは0~18)用の書き込みデータをEPIRn0~EPIRn5(EPIR[登録番号][書き込み順序])と表記します。書き込みはEPIR00から順番に行ってください。

• EPIRn0

ビット	ビット名	初期値	R/W	説明
7~4	D7~D4	不定	W	エンドポイント番号 【設定可能範囲】B'0000~B'0110 【注】B'0111~B'1111は設定禁止
3~0	D3~D0	不定	W	エンドポイントが属する Configuration 番号 【設定可能範囲】B'0000 or B'0001 【注】B'0010~B'1111は設定禁止

• EPIRn1

ビット	ビット名	初期値	R/W	説明
7~4	D7~D4	不定	W	エンドポイントが属する Interface 番号 【設定可能範囲】B'0000~B'0010 【注】B'0011~B'1111は設定禁止
3~0	D3~D0	不定	W	エンドポイントが属する Alternate 番号 【設定可能範囲】 Interface 番号 (D7~D4) の値により D3~D0 の設定可能範囲が異なります。 D7~D4 = B'0000 : D3~D0 = B'0000 D7~D4 = B'0001 : D3~D0 = B'0000~B'0101 D7~D4 = B'0010 : D3~D0 = B'0000 【注】上記以外は設定禁止

20. USB ファンクションコントローラ (USBF)

• EPIR n 2

ビット	ビット名	初期値	R/W	説 明
7、6	D7、D6	不定	W	リザーブビット 書き込む値は常に0にしてください。1を書いた場合の動作の保証はできません。
5、4	D5、D4	不定	W	エンドポイントの転送方式 【設定可能範囲】 B'00：コントロール B'01：アイソクロナス B'10：バルク B'11：インタラプト
3	D3	不定	W	エンドポイントの転送方向 【設定可能範囲】 B'0：OUT B'1：IN
2~0	D2~D0	不定	W	リザーブビット 書き込む値は常に0にしてください。1を書いた場合の動作の保証はできません。

• EPIR n 3

ビット	ビット名	初期値	R/W	説 明
7~1	D7~D1	不定	W	エンドポイントの最大バケットサイズ 【設定可能範囲】 B'00000000 ~ B'10000000 【注】 B'10000001 ~ B'11111111 は設定禁止
0	D0	不定	W	リザーブビット 書き込む値は常に0にしてください。1を書いた場合の動作の保証はできません。

• EPIR n 4

ビット	ビット名	初期値	R/W	説 明
7~0	D7~D0	不定	W	リザーブビット 書き込む値は常に0にしてください。1を書いた場合の動作の保証はできません。

• EPIR n 5

ビット	ビット名	初期値	R/W	説 明
7~0	D7~D0	不定	W	エンドポイント FIFO 番号 【設定可能範囲】 H'00 ~ H'03, H'05 ~ H'09 【注】 H'04, H'0A ~ H'FF は設定禁止

上記 EPIR n 0 に設定するエンドポイント番号とは、USB ホストが使用するエンドポイント番号です。また、EPIR

20. USB ファンクションコントローラ (USBF)

n 5 に設定するエンドポイント FIFO 番号とは、この USB ファンクションコントローラが物理的に備えている各エンドポイント付属の FIFO の番号です。本 USB ファンクションコントローラの各エンドポイントの持つ機能と FIFO 番号との対応については、「20.1 特長」の対応図を参照してください。なお、上記 EPIRn0 ~ 5 に書き込む設定値には以下の制約があるので注意してください。

1. 各エンドポイントFIFOは、各転送方式・方向・最大パケットサイズに対応した専用ハードウェアで最適化しているため、必ず下表20.2に示す転送方式・方向・最大パケットサイズに設定する。
(例) エンドポイントFIFO番号1は「インタラプト転送」「最大パケットサイズ16バイト」以外の設定はできない。エンドポイントFIFO番号5は「アイソクロナス転送」「IN」以外の設定はできないが「最大パケットサイズ」は0~64バイトの間で任意設定可能。
2. エンドポイント番号0とエンドポイントFIFO番号0は必ず1対1に対応させる。
3. エンドポイントFIFO番号0の最大パケットサイズは64のみ設定可能。
4. エンドポイントFIFO番号0は最大パケットサイズのみ設定可能で残りのデータはすべて0。
5. エンドポイントFIFO番号1および7の最大パケットサイズは16のみ設定可能。
6. エンドポイントFIFO番号2および3の最大パケットサイズは64のみ設定可能。
7. エンドポイントFIFO番号8および9の最大パケットサイズは32のみ設定可能。
8. エンドポイントFIFO番号5の最大パケットサイズは0~64の範囲で設定可能。
9. エンドポイントFIFO番号6の最大パケットサイズは0~60の範囲で設定可能。
10. アイソクロナス転送設定時、同一のエンドポイントに対して0~5の範囲でAlternateを使用できるが、必ず同じエンドポイントFIFO番号に割り当てる。
11. エンドポイント情報の設定は最大19個まで可能。
12. エンドポイント情報は必ず19個分書き込まなければならない。
13. 使用しないエンドポイントの情報はすべて0を書き込む。

表 20.2 に設定可能な転送方式・転送方向・最大パケットサイズの制約一覧を示します。

表 20.2 設定可能値の制約一覧

エンドポイント FIFO 番号	最大パケットサイズ	転送方式	転送方向
0	64 バイト	コントロール	
1	16 バイト	インタラプト	IN
2	64 バイト	バルク	IN
3	64 バイト	バルク	OUT
5	0~64 バイト	アイソクロナス	IN
6	0~60 バイト	アイソクロナス	OUT

20. USB ファンクションコントローラ (USBF)

エンドポイント FIFO 番号	最大パケットサイズ	転送方式	転送方向
7	16 バイト	インタラプト	IN
8	32 バイト	バルク	IN
9	32 バイト	バルク	OUT

以下に具体的な設定例を示します。

(1) 設定例 1

ブルートゥースの「HCI USB TRANSPORT LAYER」で推奨している例です。登録番号 16~18 のデータにはすべて 0 を書き込みます。

表 20.3 エンドポイント構成例 (1)

EP 番号	Conf.	Int.	Alt.	転送方式	転送方向	最大パケットサイズ	EP FIFO 番号
0				コントロール	IN/OUT	64 バイト	0
1	1	0	0	インタラプト	IN	16 バイト	1
2	1	0	0	バルク	IN	64 バイト	2
2	1	0	0	バルク	OUT	64 バイト	3
3	1	1	0	アイソクロナス	IN	0 バイト	5
3	1	1	0	アイソクロナス	OUT	0 バイト	6
3	1	1	1	アイソクロナス	IN	9 バイト	5
3	1	1	1	アイソクロナス	OUT	9 バイト	6
3	1	1	2	アイソクロナス	IN	17 バイト	5
3	1	1	2	アイソクロナス	OUT	17 バイト	6
3	1	1	3	アイソクロナス	IN	25 バイト	5
3	1	1	3	アイソクロナス	OUT	25 バイト	6
3	1	1	4	アイソクロナス	IN	33 バイト	5
3	1	1	4	アイソクロナス	OUT	33 バイト	6
3	1	1	5	アイソクロナス	IN	49 バイト	5
3	1	1	5	アイソクロナス	OUT	49 バイト	6

20. USB ファンクションコントローラ (USBF)

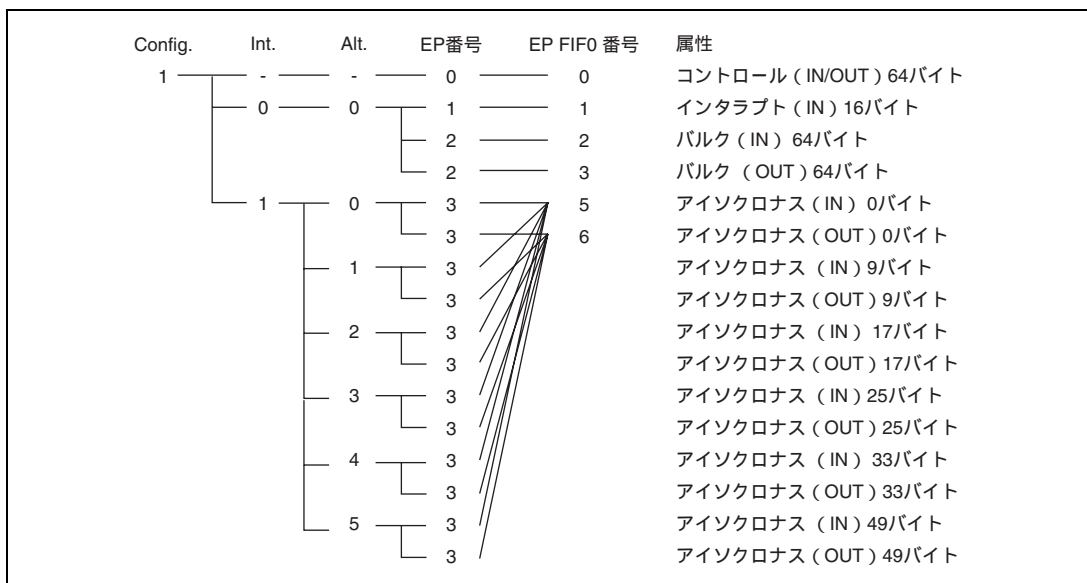


図 20.2 エンドポイント構成例 (1)

表 20.4 エンドポイント構成情報設定例 (1)

N	EPIR[N]0	EPIR[N]1	EPIR[N]2	EPIR[N]3	EPIR[N]4	EPIR[N]5
0	00	00	00	80	00	00
1	11	00	38	20	00	01
2	21	00	28	80	00	02
3	21	00	20	80	00	03
4	31	10	18	00	00	05
5	31	10	10	00	00	06
6	31	11	18	12	00	05
7	31	11	10	12	00	06
8	31	12	18	22	00	05
9	31	12	10	22	00	06
10	31	13	18	32	00	05
11	31	13	10	32	00	06
12	31	14	18	42	00	05
13	31	14	10	42	00	06
14	31	15	18	62	00	05
15	31	15	10	62	00	06
16	00	00	00	00	00	00
17	00	00	00	00	00	00
18	00	00	00	00	00	00

(2) 設定例 2

使用しないエンドポイント情報がある場合の別の例です。登録番号 13～18 のデータにはすべて 0 を書き込みます。

表 20.5 エンドポイント構成例 (2)

EP 番号	Conf.	Int.	Alt.	転送方式	転送方向	最大パケットサイズ	EP FIFO 番号
0				コントロール	IN/OUT	64 バイト	0
1	1	0	0	インタラプト	IN	16 バイト	1
2	1	0	0	バルク	IN	64 バイト	2
2	1	0	0	バルク	OUT	64 バイト	3
3	1	1	0	アイソクロナス	IN	0 バイト	5
3	1	1	0	アイソクロナス	OUT	0 バイト	6
3	1	1	1	アイソクロナス	IN	32 バイト	5
3	1	1	1	アイソクロナス	OUT	32 バイト	6
3	1	1	2	アイソクロナス	IN	64 バイト	5
3	1	1	2	アイソクロナス	OUT	60 バイト	6
4	1	2	0	インタラプト	IN	16 バイト	7
5	1	2	0	バルク	IN	32 バイト	8
6	1	2	0	バルク	OUT	32 バイト	9

20. USB ファンクションコントローラ (USBF)

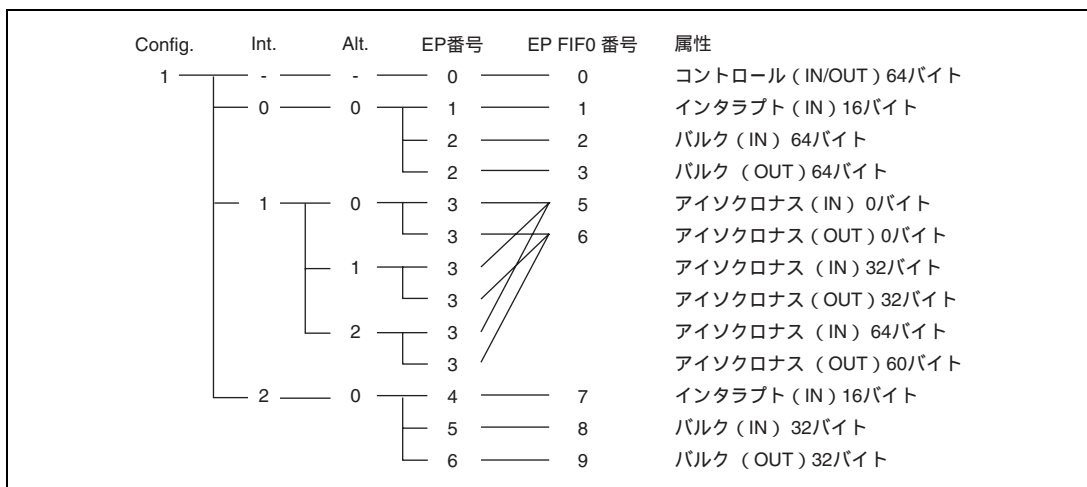


図 20.3 エンドポイント構成例 (2)

表 20.6 エンドポイント構成情報設定例 (2)

n	EPIR[n]0	EPIR[n]1	EPIR[n]2	EPIR[n]3	EPIR[n]4	EPIR[n]5
0	00	00	00	80	00	00
1	11	00	38	20	00	01
2	21	00	28	80	00	02
3	21	00	20	80	00	03
4	31	10	18	00	00	05
5	31	10	10	00	00	06
6	31	11	18	40	00	05
7	31	11	10	40	00	06
8	31	12	18	80	00	05
9	31	12	10	78	00	06
10	41	20	38	20	00	07
11	51	20	28	40	00	08
12	61	20	20	40	00	09
13	00	00	00	00	00	00
14	00	00	00	00	00	00
15	00	00	00	00	00	00
16	00	00	00	00	00	00
17	00	00	00	00	00	00
18	00	00	00	00	00	00

20.4 動作説明

20.4.1 ケーブル接続時

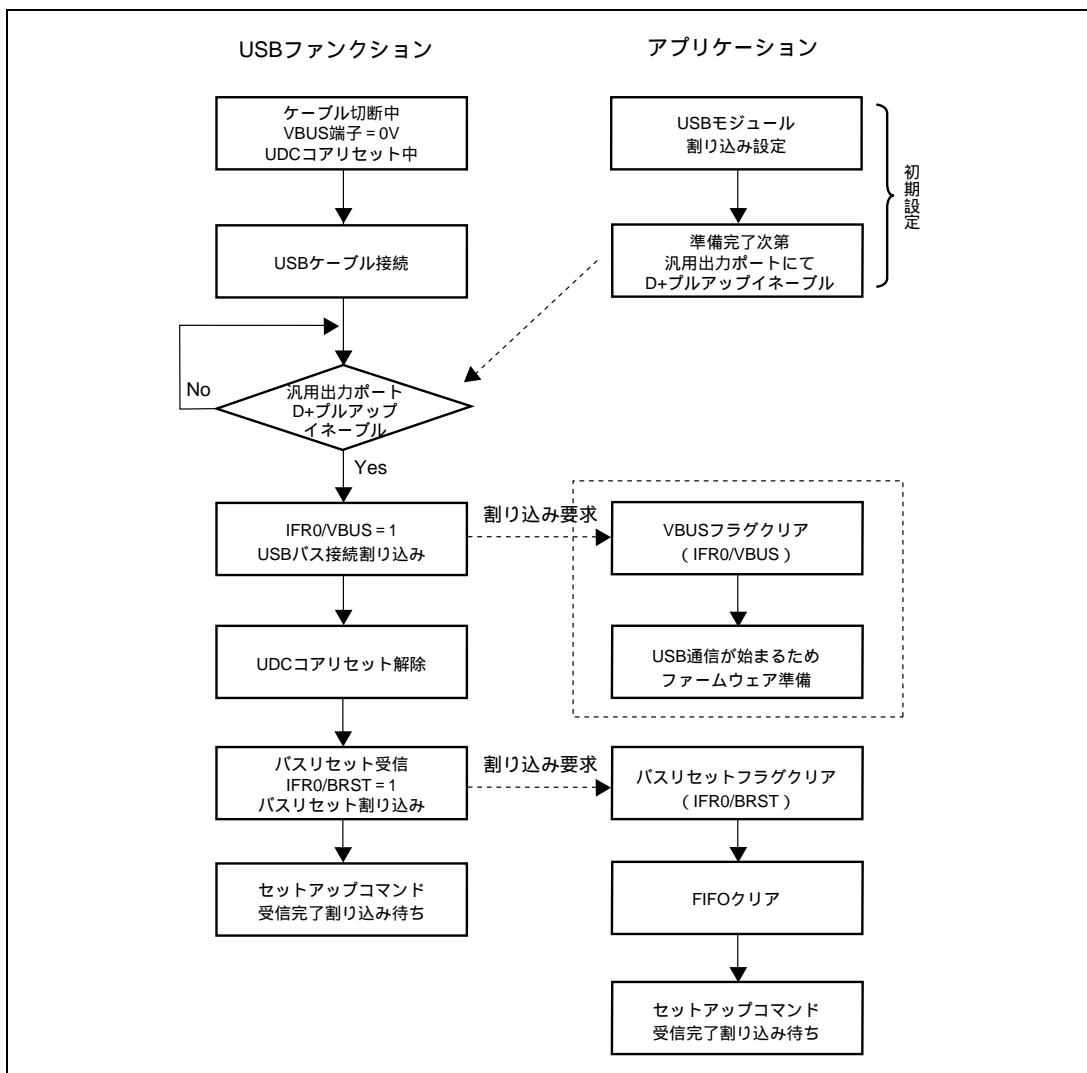


図 20.4 ケーブル接続時の動作

上記フローは、「20.7 USB ファンクションコントローラと外部回路の接続例」時の動作を示しています。

USB ケーブル接続を検出する必要がないアプリケーションでは、USB バス接続割り込みによる処理は不要です。バスリセット割り込みにて準備してください。

詳細は「20.7 USB ファンクションコントローラと外部回路の接続例」を参照してください。

20.4.2 ケーブル切断時

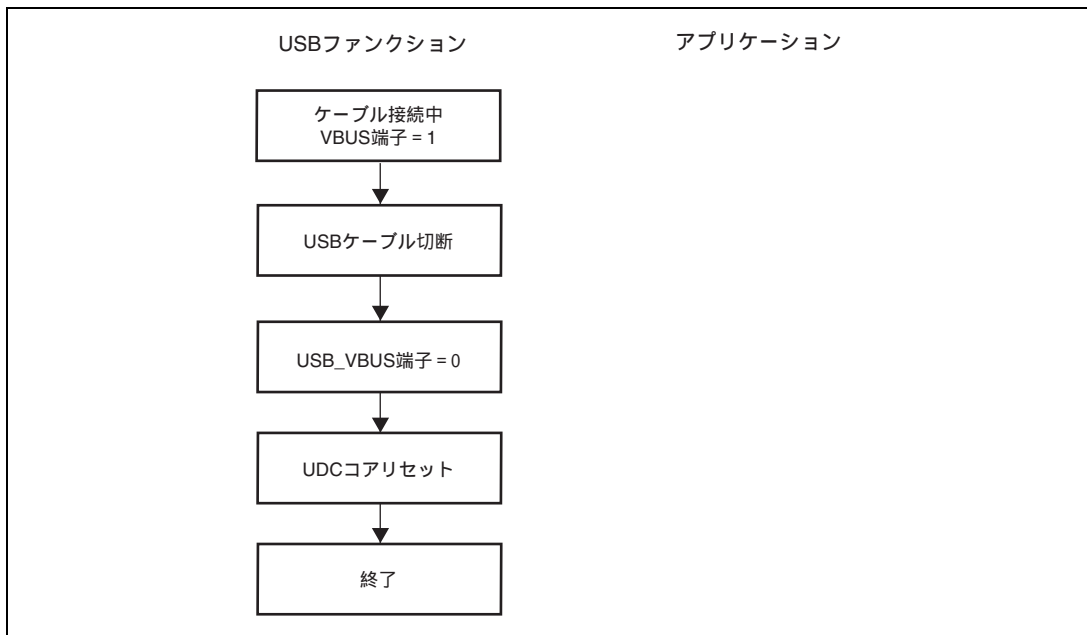


図 20.5 ケーブル切断時の動作

上記フローは、「20.7 USB ファンクションコントローラと外部回路の接続例」時の動作を示しています。詳細は「20.7 USB ファンクションコントローラと外部回路の接続例」を参照してください。

20.4.3 コントロール転送

コントロール転送は、セットアップ、データ（無い場合もある）、ステータスの3つのステージで構成されます（図 20.6）。また、データステージは、複数のバストランザクションで構成されます。以下に、各ステージごとの動作フローを示します。

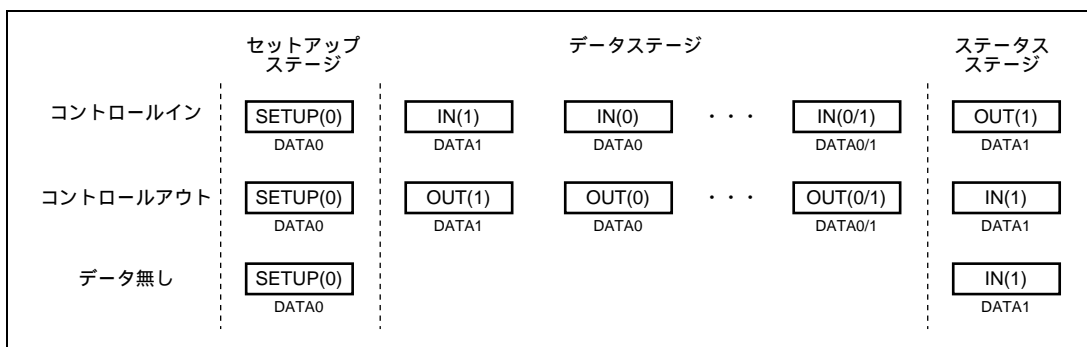


図 20.6 コントロール転送における各転送ステージ

(1) セットアップステージ

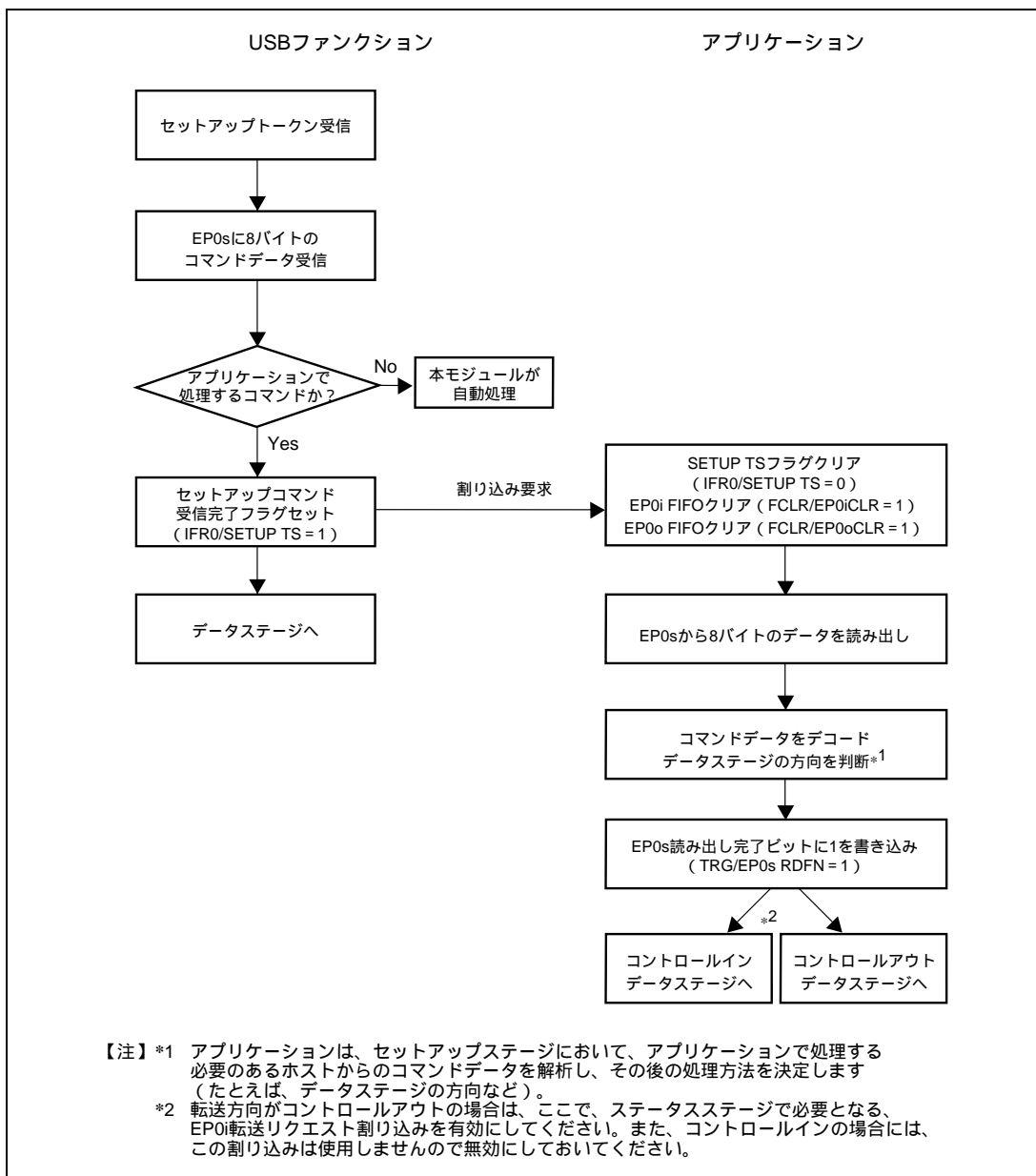


図 20.7 セットアップステージの動作

(2) データステージ (コントロールイン時)

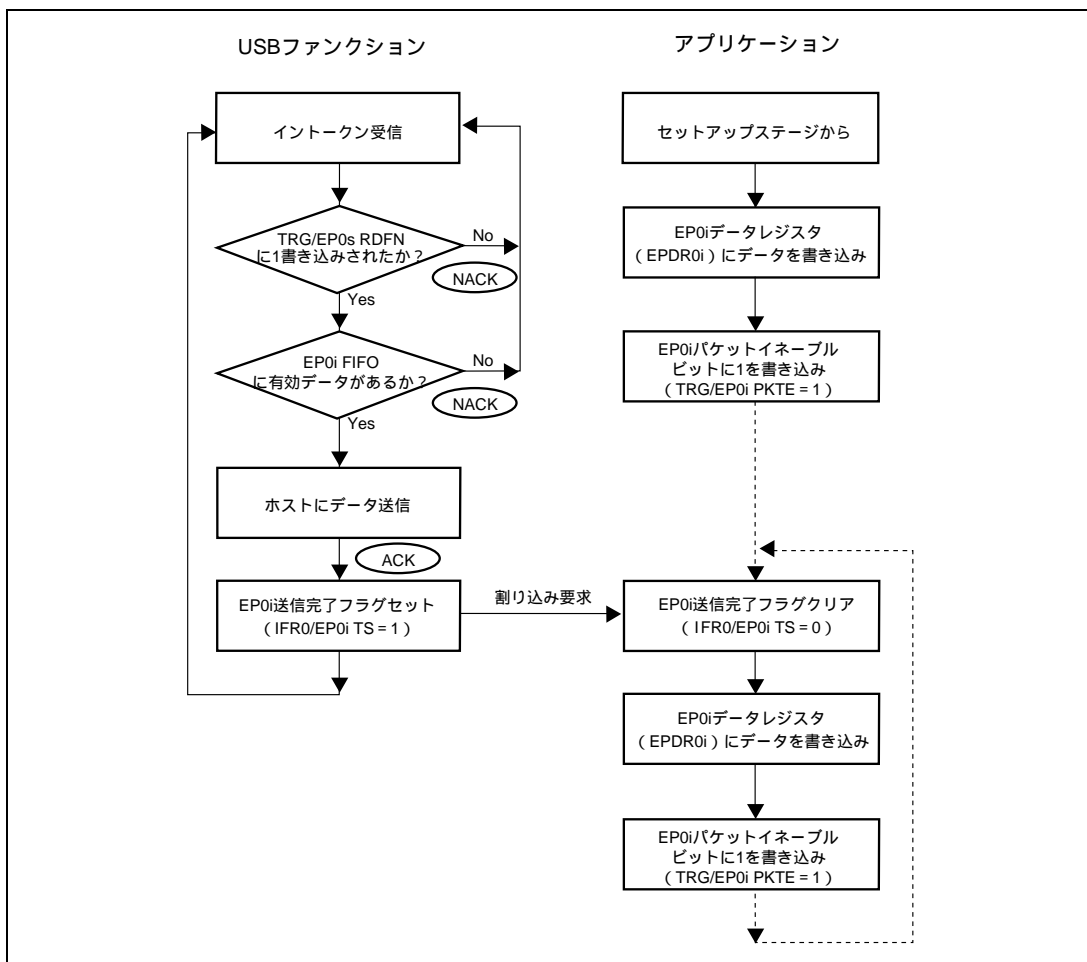


図 20.8 データステージ (コントロールイン時) の動作

アプリケーションは、まずセットアップステージにおいて、ホストからのコマンドデータを解析し、その後のデータステージの方向を判断します。コマンドデータの解析結果により、データステージがイン転送の場合、ホストに送りたいデータの1パケット分をFIFOに書き込みます。さらに送りたいデータがある場合、最初に書きこんだデータがホストに送られた後 (IFR0/EP0i TS = 1)、FIFOにデータを書き込みます。

データステージの終わりは、ホストがアウトトークンを送信し、ステータスステージに入ったことで判断します。

【注】 ファンクションが送信するデータのサイズが、ホストから要求されたデータサイズより小さい場合、ファンクションは、最大パケットサイズより短いパケットをホストに返すことで、データステージの終了を示します。また、ファンクションが送信するデータのサイズが、最大パケットサイズの整数倍の場合には、0 レングスパケットを送信して、データステージの最後を示します。

(3) データステージ (コントロールアウト時)

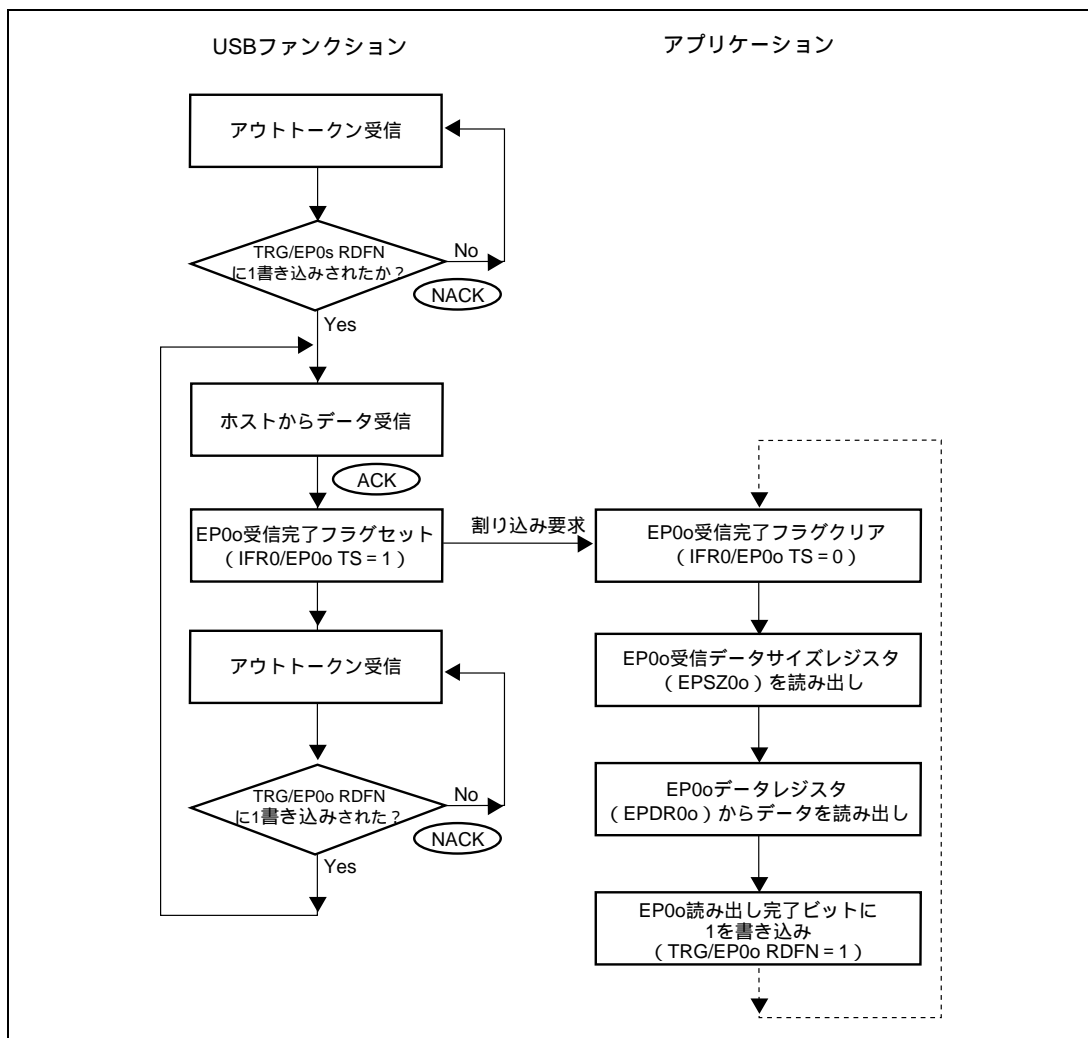


図 20.9 データステージ (コントロールアウト時) の動作

アプリケーションは、まずセットアップステージにおいて、ホストからのコマンドデータを解析し、その後のデータステージの方向を判断します。コマンドデータの解析結果により、データステージがアウト転送の場合、ホストからのデータを待ち、データ受信後 (IFR0/EP0o TS = 1)、FIFO からデータを読み出します。次にアプリケーションは、EP0o 読み出し完了ビットに 1 を書き込み、受信 FIFO を空にして、次のデータ受信を待ちます。データステージの終わりは、ホストがイントークンを送信し、ステータスステージに入ったことで判断します。

20. USB ファンクションコントローラ (USBF)

(4) ステータスステージ (コントロールイン時)

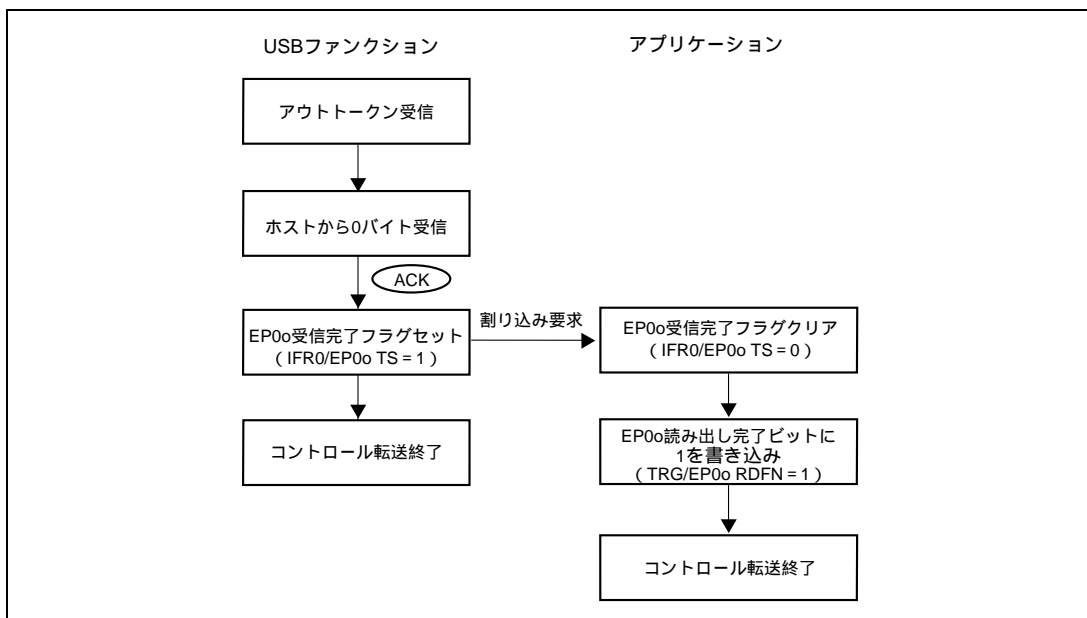


図 20.10 ステータスステージ (コントロールイン時) の動作

コントロールイン時のステータスステージは、ホストからのアウトトークンで始まります。アプリケーションは、ホストからの0バイトデータを受信して、コントロール転送を終了します。

(5) ステータスステージ (コントロールアウト時)

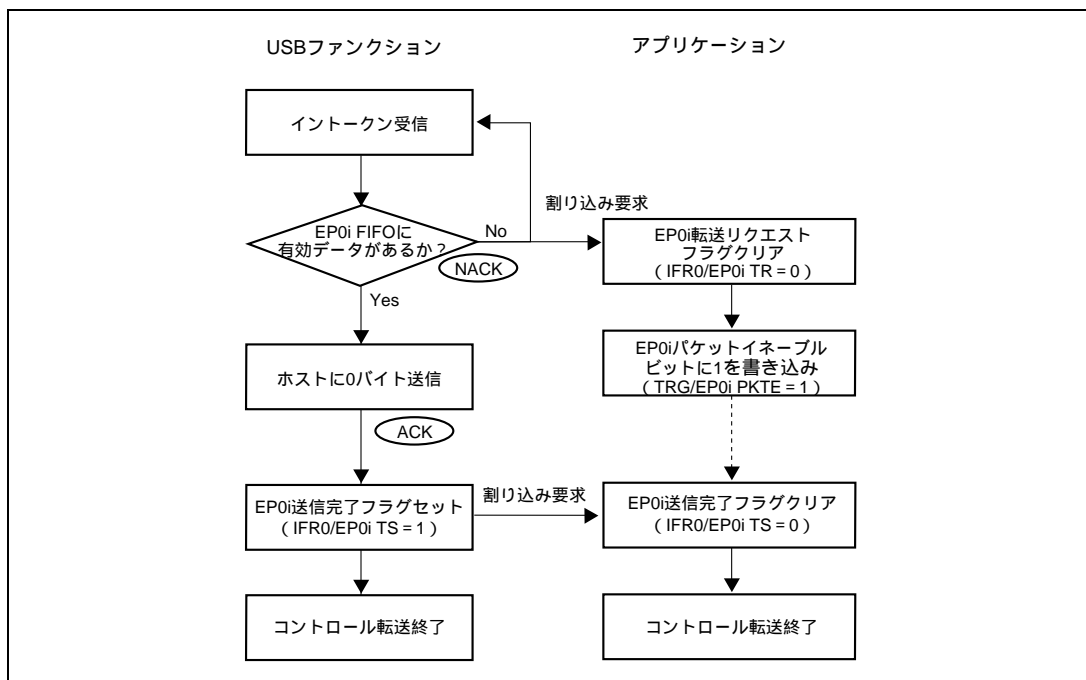


図 20.11 ステータスステージ (コントロールアウト時) の動作

コントロールアウト時のステータスステージは、ホストからのイントークンで始まります。ステータスステージの始まりのイントークン受信時には、まだ EP0i FIFO にはデータが入っていないので、EP0i 転送リクエスト割り込みが入ります。アプリケーションは、この割り込みによりステータスステージが開始されたことを認識します。次に、ホストに 0 バイトデータを送信するために、EP0i FIFO にデータを書き込まず、EP0i パケットイネーブルビットに 1 を書き込みます。これにより、次のイントークンでホストに 0 バイトデータが送信され、コントロール転送が終了します。

ただし、アプリケーションが、データステージにかかわる処理をすべて終了した後、EP0i パケットイネーブルビットに 1 を書き込んでください。

20.4.4 EP1、4 インタラプトイン転送

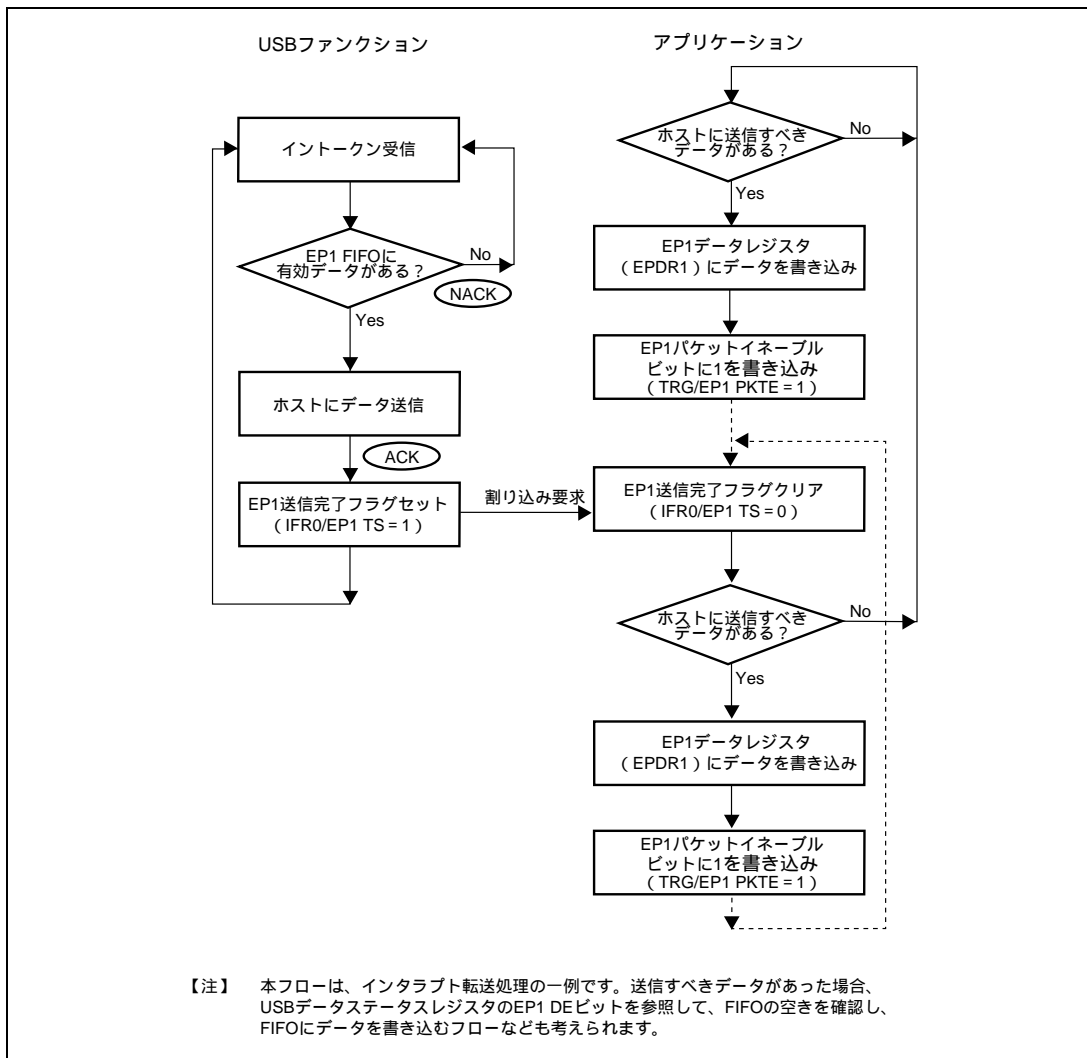


図 20.12 EP1 インタラプトイン転送の動作

20.4.5 EP2i、5 バルクイン転送 (2 面 FIFO)

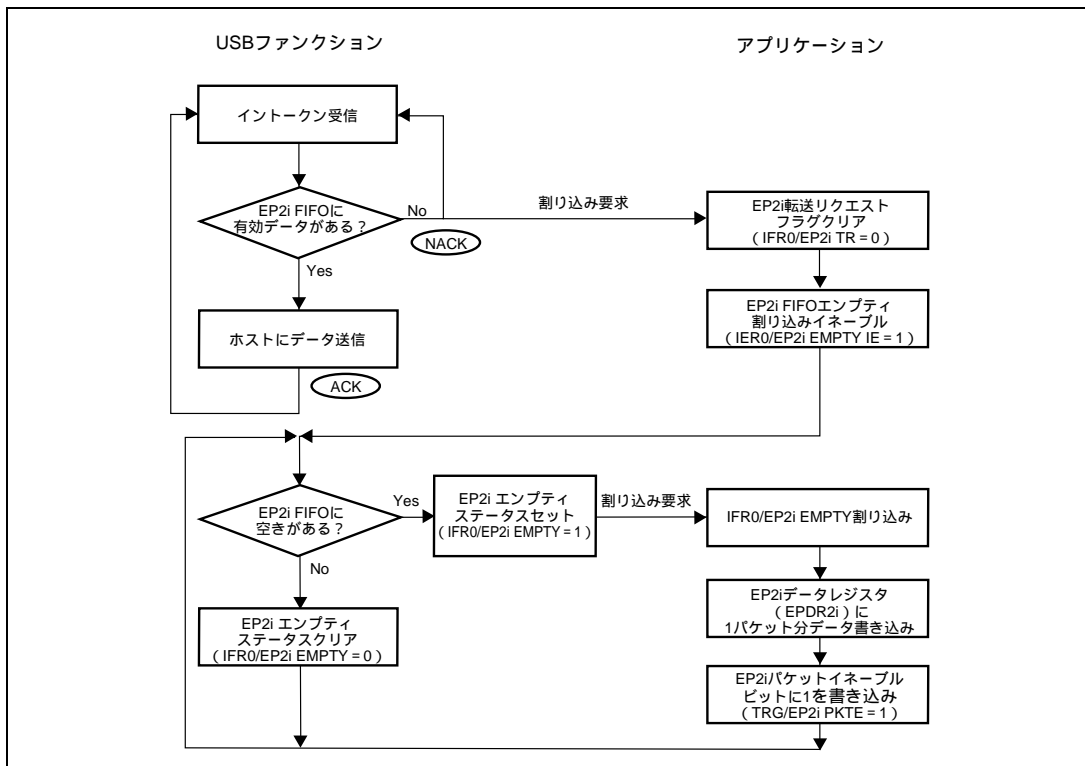


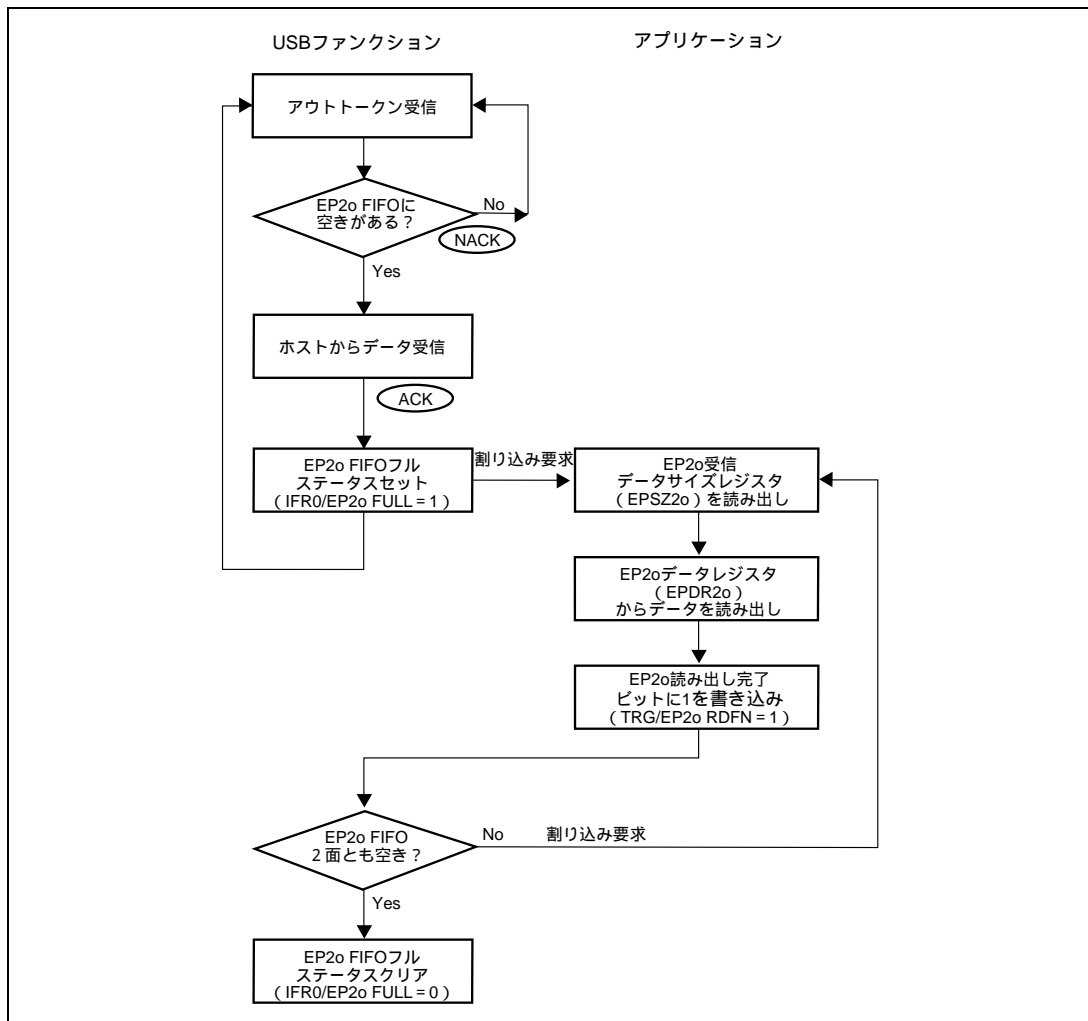
図 20.13 EP2i バルクイン転送の動作

EP2i は 64 バイトの FIFO を 2 面持っています。しかし、ユーザは 2 面あることを意識することなく、データ送信および送信データの書き込みができます。ただし、1 回のデータ書き込みは 1 面ごとに行ってください。たとえば、2 面とも FIFO が空の場合でも、連続して 128 バイトデータを書き込み後、まとめて EP2i PKTE ビットのセットを行うことはできません。必ず 64 バイトの書き込みごとに EP2i PKTE ビットのセットを行ってください。

バルクイン転送を行いたい場合、まず最初のイントークンで FIFO 内に有効データが存在しないので、IFR0/EP2i TR ビット割り込みが要求されます。その割り込みで、IER0/EP2i EMPTY IE ビットに 1 を書き込み、EP2i FIFO エンティ割り込みを許可します。最初は、EP2i の 2 面の FIFO は共に空になっているので、EP2i FIFO エンティ割り込みがすぐに発生します。

この割り込みを使って、送信するデータをデータレジスタに書き込みます。最初 1 面分の送信データを書き込んだ後は、他方の FIFO が空いているので、すぐ他方の面に送信データを書き込みます。2 面とも一杯になった場合、IFR0/EP2i EMPTY ビットが 0 になります。1 面でも空であれば IFR0/EP2i EMPTY ビットが 1 セットされます。データ送信完了後、ホストから ACK が返ってきたら、データ送信を行った FIFO が空になります。このとき、他方の FIFO に有効な送信データが用意されている場合は、連続して送信動作が行えます。

すべての送信が完了後、IER0/EP2i EMPTY IE ビットに 0 を書き込み、割り込み要求を禁止にしてください。

20.4.6 EP2_o、6 バルクアウト転送 (2 面 FIFO)図 20.14 EP2_o バルクアウト転送の動作

EP2_o は 64 バイトの FIFO を 2 面持っています。しかし、ユーザは 2 面あることを意識することなく、データ受信および受信データの読み出しができます。

FIFO が 1 面でも受信完了すると、IFR0/EP2_o FULL ビットがセットされます。FIFO が 2 面とも空の状態での最初の受信後は、他方の FIFO が空いているので、すぐ次のパケットを受信することができます。2 面とも一杯になった場合、ホストには自動的に NACK が返信されます。データ受信後、受信データの読み出しが終了したら、TRG/EP2_o RDFN ビットに 1 を書き込みます。この操作によって、今読み出しを終えた FIFO が空になり、次のパケットを受信可能な状態になります。

20.4.7 EP3i アイソクロナスイン転送

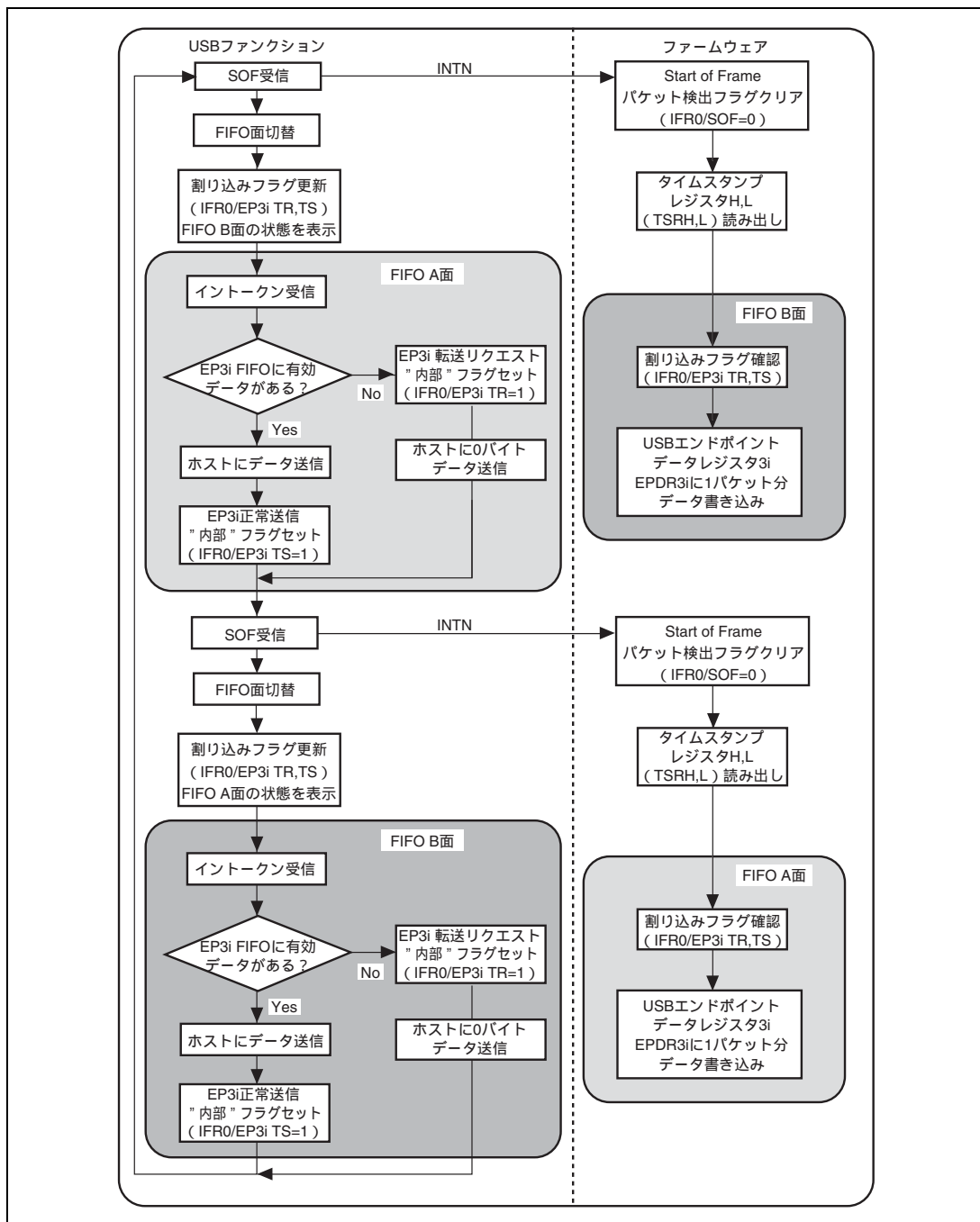


図 20.15 EP3i アイソクロナスイン転送の動作

20. USB ファンクションコントローラ (USBF)

EP3i は、最大 64 バイトの FIFO を 2 面持っています。しかし、ユーザは 2 面あることを意識することなく、データ送信および送信データの書き込みができます (ただし、フローチャートでは説明のために A 面、B 面と呼びます)。

アイソクロナス転送は、1 フレーム (1ms) に 1 回だけ、転送が発生するため、SOF を受信すると、ハードが自動で、FIFO の面を切り替えます (SOF がエラーにより受信できない場合でも、SOF マーカ機能をイネーブルにすれば、1ms 周期でハードが自動で FIFO の面を切り替えます)。

SOF 受信で、FIFO の面を切り替えるため、USB ファンクションがホストにデータを送信する FIFO とファームウェアが送信データを書き込む FIFO は、同フレーム内では面が異なり、FIFO の読み出しと書き込みが競合することはありません。したがって、ファームウェアが書き込むデータは、1 つ後のフレームで送信するデータとなります。また、SOF 受信により FIFO の面が自動的に切り替わるので、必ずデータの書き込みは、フレーム内に完了してください。

USB ファンクションは、イントークン受信後、FIFO 内にホストへ送信するデータがある場合、データをホストに送信し、TS 内部フラグを 1 にセットします。FIFO にデータが無ければ TR 内部フラグを 1 にセットし、ホストに 0 バイトデータを送信します。内部フラグ情報 (TR、TS) は SOF 受信でファームウェアにて読み出すことが可能な IFR0 フラグ情報 (TR、TS) として自動的に更新されます。

ファームウェアでは、まず SOF 割り込みによりアイソクロナス転送の処理ルーチン呼び出し、タイムスタンプのチェックを行います。その後 1 パケット分のデータを FIFO に書き込みます。この書き込んだデータは、次のフレームでホストに送信されます。また、前のフレームで転送が正常に行われなかったかどうかは、IFR0 レジスタのフラグ情報 (TR、TS) を読み出して判断することができます。

20.4.8 EP3o アイソクロナスアウト転送

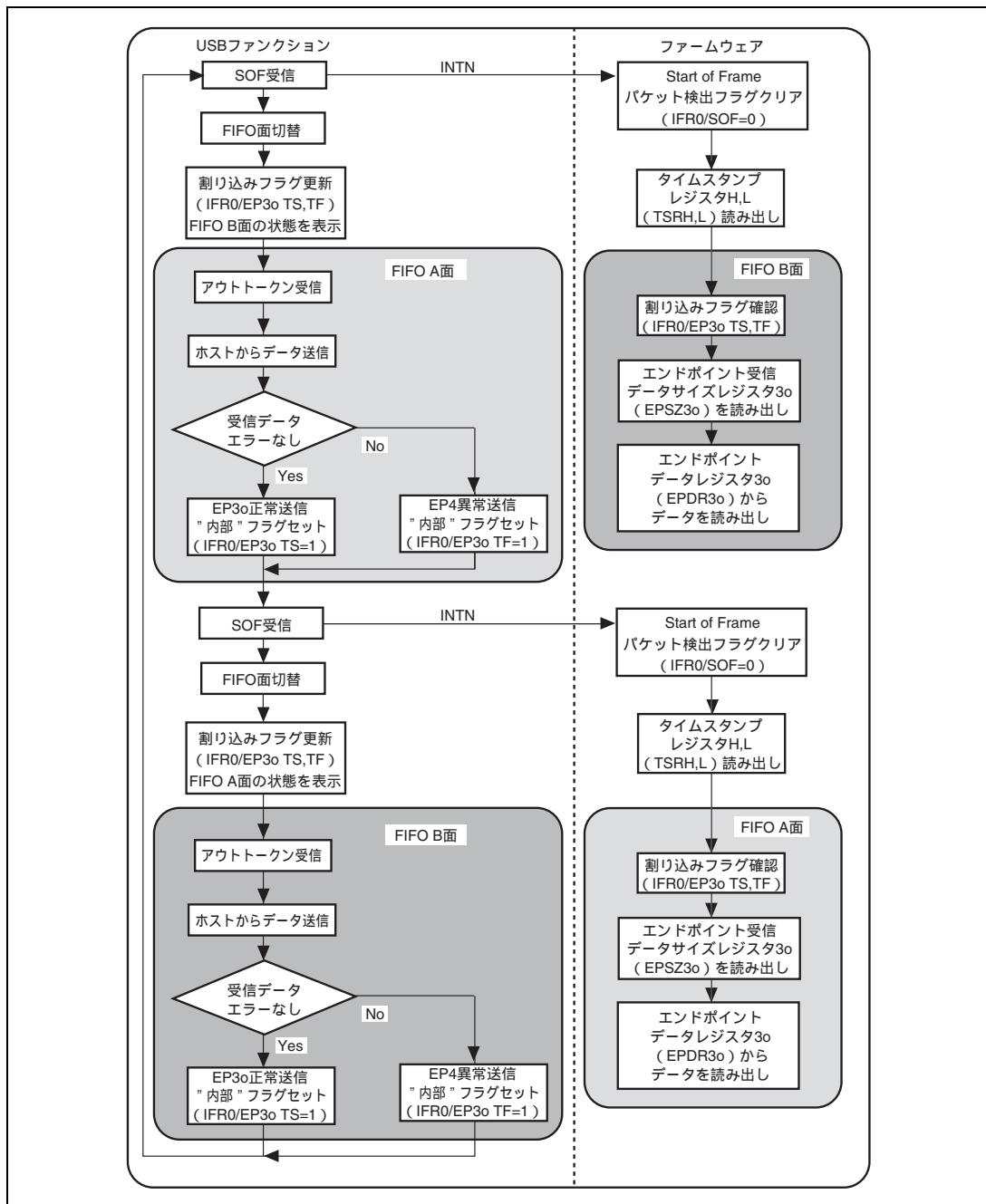


図 20.16 EP3o アイソクロナスアウト転送の動作

20. USB ファンクションコントローラ (USBF)

EP3₀ は、最大 60 バイトの FIFO を 2 面持っています。しかし、ユーザは 2 面あることを意識することなく、データ送信および受信データの読み出しができます。(ただし、フローチャートでは説明のために A 面、B 面と呼びます)。

アイソクロナス転送は、1 フレーム (1ms) に 1 回だけ、転送が発生するため、SOF を受信すると、ハードが自動で、FIFO の面を切り替えます (SOF がエラーにより受信できない場合でも、SOF マーカ機能をイネーブルにすれば、1ms 周期でハードが自動で FIFO の面を切り替えます)。

SOF 受信で、FIFO の面を切り替えるため、USB ファンクションがホストからデータを受信する FIFO とファームウェアが受信データを読み出す FIFO は、同フレーム内では面が異なり、FIFO の読み出しと書き込みが競合することはありません。したがって、ファームウェアが書き込むデータは、1 つ前のフレームで受信したデータとなります。また、SOF 受信により FIFO の面が自動的に切り替わるので、必ずデータの読み出しは、フレーム内に完了してください。

USB ファンクションは、アウトトークン受信後、ホストからデータを受信しますが、このとき、データにエラーがあった場合、TF 内部フラグを 1 にセットします。データにエラーがなければ TS 内部フラグを 1 にセットします。内部フラグ情報 (TF、TS) は、SOF 受信でファームウェアにて読み出すことが可能な IFR0 フラグ情報 (TF、TS) として自動的に更新されます。

ファームウェアでは、まず SOF 割り込みによりアイソクロナス転送の処理ルーチンを呼び出し、タイムスタンプのチェックを行います。その後、データを FIFO から読み出します。またそのデータにエラーがあるかどうかは、IFR0 レジスタのフラグ情報 (TS、TF) を読み出して判断します。このときの IFR0 のフラグ情報は、現在読み出し可能な FIFO の面のステータスを表しています。

20.5 USB 標準コマンドとクラス / ベンダーコマンドの処理

20.5.1 コントロール転送で送信されるコマンドの処理

コントロール転送でホストから送信されてくるコマンドによっては、アプリケーション側でデコードを行い、コマンドの処理を行う必要があります。以下の表 20.7 にアプリケーション側でのコマンドデコードについて示します。

表 20.7 アプリケーション側でのコマンドデコード

アプリケーション側でデコードの必要無し	アプリケーション側でデコードの必要あり
Clear Feature	Get Descriptor
Get Configuration	Class/Vendor コマンド
Get Interface	Synch Frame
Get Status	Set Descriptor
Set Address	
Set Configuration	
Set Feature	
Set Interface	

アプリケーション側でデコードする必要のない場合、コマンドデコード、データステージ、ステータスステージ処理は自動的に行われます。したがって、ユーザは何もする必要はありません。また、割り込みも発生しません。

アプリケーション側でデコードする必要がある場合には、本モジュールはコマンドを EP0s の FIFO に保存します。正常受信完了後、IFR0/SETUP TS フラグがセットされ、割り込み要求を発生します。この割り込みルーチンの中で EP0s のデータレジスタ (EPDR0S) から 8 バイトのデータを読み出し、ファームウェアでデコードしてください。その後、デコードの結果により、必要となるデータステージ、ステータスステージの処理を行ってください。

20.6 ストール動作

20.6.1 概要

本モジュールでのストール動作について説明します。本モジュールのストール機能には、次の2つの場合があります。

1. アプリケーションが何らかの理由で強制的にエンドポイントをストールさせる場合
2. USBの規格違反によって本モジュール内部で自動的にストールする場合

本モジュール内には、各エンドポイントの状態(ストールか否か)を保持した内部状態ビットを持っています。ホストからトランザクションが送られてきたとき、本モジュールはこの内部状態ビットを参照してホストにストールを返すかどうか判断します。このビットは、アプリケーションでは解除できません。解除する場合はホストから Clear Feature コマンドを使ってクリアしてください。ただし、EP0 に対する内部状態ビットはセットアップコマンド受信時のみ自動的にクリアされます。

20.6.2 アプリケーションが強制的にストールさせたい場合

アプリケーションが本モジュールに対してストール要求するレジスタ EPSTL を使用します。アプリケーションが特定のエンドポイントをストールさせたい場合、EPSTL の該当ビットをセットします(図 20.17 の 1-1)。このとき、内部状態ビットは変化しません。次に、ホストから EPSTL の該当ビットがセットされているエンドポイントに対してトランザクションが送られてきたとき、本モジュールは内部状態ビットを参照し、セットされていなければ EPSTL の該当ビットを参照します(図 20.17 の 1-2)。ここで、EPSTL の該当ビットがセットされていれば、本モジュールは内部状態ビットをセットし、ホストに対してストールハンドシェイクを返します(図 20.17 の 1-3)。EPSTL の該当ビットがセットされていなければ、内部状態ビットは変化せず、トランザクションが受け付けられます。

一度、内部状態ビットがセットされた後は、EPSTL に関係なく、ホストから Clear Feature コマンドでクリアされるまで内部状態ビットは保持されます。Clear Feature コマンドで該当ビットがクリアされても(図 20.17 の 3-1) EPSTL のビットがセットされている間は、該当エンドポイントに対するトランザクションが行われるたびに内部状態ビットがセットされるため、本モジュールはストールハンドシェイクを返します(図 20.17 の 1-2)。したがって、ストールを解除するためには、EPSTL の該当ビットをアプリケーションでクリアし、さらに Clear Feature コマンドで内部状態ビットをクリアする必要があります(図 20.17 の 2-1、2-2、2-3)。

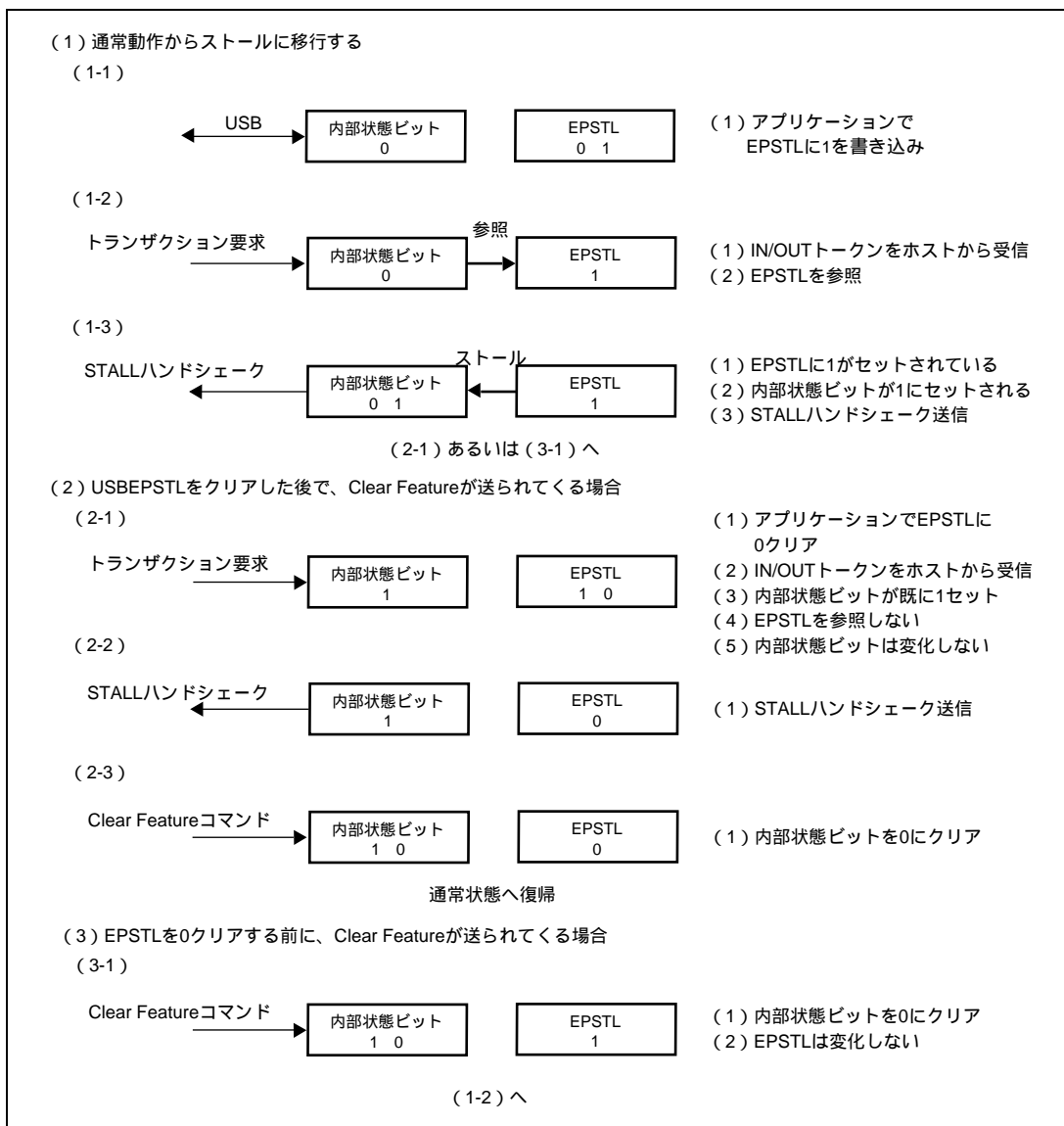


図 20.17 アプリケーションで強制的にストールさせたい場合

20.6.3 USB ファンクションコントローラが自動的にストールさせる場合

Set Feature コマンドでストール設定した場合、あるいはUSBの規格違反があった場合は、EPSTLに関係なく本モジュールが自動的に該当エンドポイントの内部状態ビットをセットし、ストールハンドシェイクを返します(図20.18の1-1)。

一度、内部状態ビットがセットされた後は、EPSTLに関係なく、ホストから Clear Feature コマンドでクリアされるまで、内部状態ビットは保持されます。Clear Feature コマンドで該当ビットがクリアされた後は、EPSTLを参照するようになります(図20.18の3-1)。内部状態ビットがセットされている間は、該当エンドポイントに対するトランザクションが行われても、内部状態ビットがセットされているため、本モジュールはストールハンドシェイクを返します(図20.18の2-1、2-2)。したがって、ストールを解除するには、Clear Feature コマンドで内部状態ビットをクリアする必要があります(図20.18の3-1)。もし、アプリケーションによってEPSTLをセットしている場合は、EPSTLもクリアしてください(図20.18の2-1)。

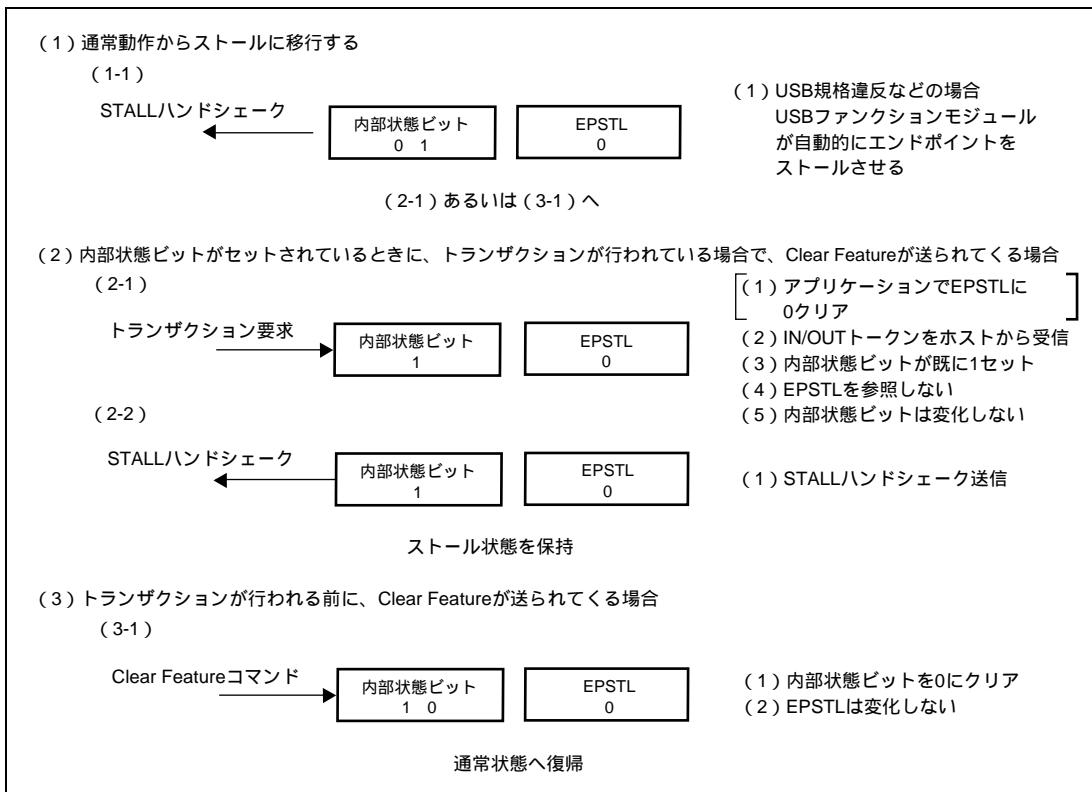


図 20.18 USB ファンクションコントローラが自動的にストールさせた場合

20.7 USB ファンクションコントローラと外部回路の接続例

(1) D+のプルアップ制御

USB ホスト / ハブへの接続通知 (D+プルアップ) を遅延させたい (優先度の高い処理中、初期化処理中等) システムでは、D+のプルアップを汎用出力ポートを用いて制御してください。しかし、USB ケーブルがすでにホスト / ハブに接続され D+のプルアップを禁止している場合、D+、D-が共にローレベル (ホスト / ハブ側は D+、D-を共にプルダウン) となり、USB ファンクションコントローラはホストから USB パスリセットを受信したものと誤って認識してしまいます。そのため、下記の回路例のように D+のプルアップ制御信号および USB_VBUS 端子入力信号は USB_PULLUP 端子出力と USB ケーブル VBUS を用いて (AND 回路) 制御してください (本 LSI の UDC コアは、USB_VBUS 端子がローレベルのとき、D+、D-の状態にかかわらずパワードステートを保持します)。

(2) USB ケーブル接続 / 切断の検出

本モジュールはハードウェアにて USB のステート等を管理しているため、接続 / 切断を認識する VBUS 信号が必要となります。VBUS は USB ケーブル内の電源信号 (VBUS) を用いますが、ファンクション (本 LSI 搭載システム) が電源オフ時、USB ホスト / ハブにケーブルが接続されると、USB ホスト / ハブから電圧 (5V) が印加されてしまいます。そのため、システム電源オフ時に電圧印加が可能な IC (HD74LV1G08A、2G08A 等) を外部に搭載してください。

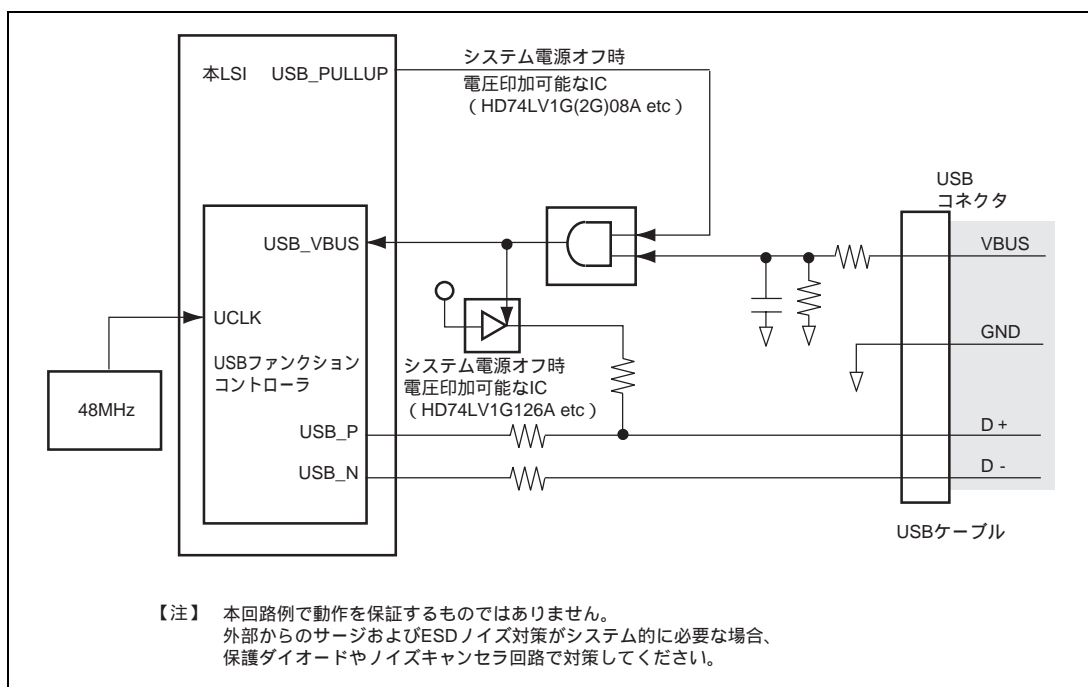


図 20.19 USB ファンクションコントローラと外部回路の接続例

20.8 使用上の注意事項

20.8.1 セットアップデータ受信について

8 バイトのセットアップデータ受信を行う EPDR0 レジスタは以下の点に注意してください。

1. USBではセットアップコマンドを必ず受信することになっているため、CPU側からの読み出しよりも、USBバス側からの書き込みが優先になっています。受信完了後にCPUでデータの読み出しを行っている最中に、次のセットアップコマンドの受信が開始された場合、書き込みを優先にするためCPU側からの読み出しを強制的に無効にします。したがって、受信開始後に読み出しされる値は不定値になります。
2. EPDR0sは必ず8バイト単位で読み出ししてください。途中で読み出しを中止すると次のセットアップで受信したデータが正常に読み出せません。

20.8.2 FIFO のクリアについて

USB ケーブル接続後、通信途中で抜かれた場合、受信中あるいは送信中のデータが FIFO 内に残っている場合があります。したがって、ケーブル再接続後は、速やかに FIFO のクリアを行ってください。

なお、ホストからデータ受信中あるいはホストに対してデータ送信中の FIFO クリアは行わないでください。

20.8.3 データレジスタの読み出し / 書き込みについて

本 USB ファンクションコントローラのデータレジスタの読み出し / 書き込みをする際は、以下の点に注意してください。

(1) 受信用データレジスタ

受信用データレジスタは、有効な受信データ数以上読み出さないでください。即ち、受信データサイズレジスタに示されるバイト数以上の読み出しをしないでください。2面 FIFO を持つ受信用データレジスタの場合も1回に読み出せる最大データ数は最大パケットサイズです。現在有効になっている面のデータの読み出しを終了したら、必ず TRG/RDFN ビットに1を書き込んでください。この操作を行うことで、他方の面に切り替わり、新しいバイト数が受信データサイズに反映され、次のデータの読み出しが可能になります。

(2) 送信用データレジスタ

送信用データレジスタは、最大パケットサイズ以上書き込まないでください。2面 FIFO を持つ送信用データレジスタの場合も、1回の書き込みは必ず最大パケットサイズ以内にしてください。データの書き込み後、TRG/PKTE ビットに1を書き込むと本モジュール内で面が切り替わり、他方の面に対する次のデータの書き込みが可能になります。したがって、2面分連続でデータの書き込みを行わないでください。

20.8.4 EP0 に関する割り込み要因の割り当てについて

本モジュールの IFR0 レジスタに割り当てられた EP0 に関する割り込み要因(ビット 0、1、2、29)は、必ず ISR0 レジスタで同じ割り込み端子に割り当ててください。その他の割り込み要因には特に制約はありません。

20.8.5 DMA 転送設定時の FIFO クリアについて

エンドポイント 2_o およびエンドポイント 6 において、DMA 転送をイネーブルにしているときは、データレジスタのクリアはできません。クリアを行う場合は、DMA 転送を解除してから行ってください。

20.8.6 TR 割り込み使用時の注意

バルクイン転送には転送要求割り込み (TR 割り込み) がありますが、本割り込みを使用するときは次の点に注意してください。

TR 割り込みフラグは、USB ホストから IN トークンが送られてきたとき、該当 EP の FIFO にデータがないときにセットされます。しかし、図 20.20 に示すようなタイミングの場合、連続して TR 割り込みが発生します。このような場合でも誤動作しないように注意してください。

【注】 本モジュールは IN トークン受信時、該当 EP の FIFO にデータが無い場合に NACK 判定を行います。TR 割り込みフラグは NACK ハンドシェイク送信後にセットされます。したがって、TRG/PKTE ビットの書き込みが次の IN トークンより遅れた場合、TR 割り込みフラグが再度セットされます。

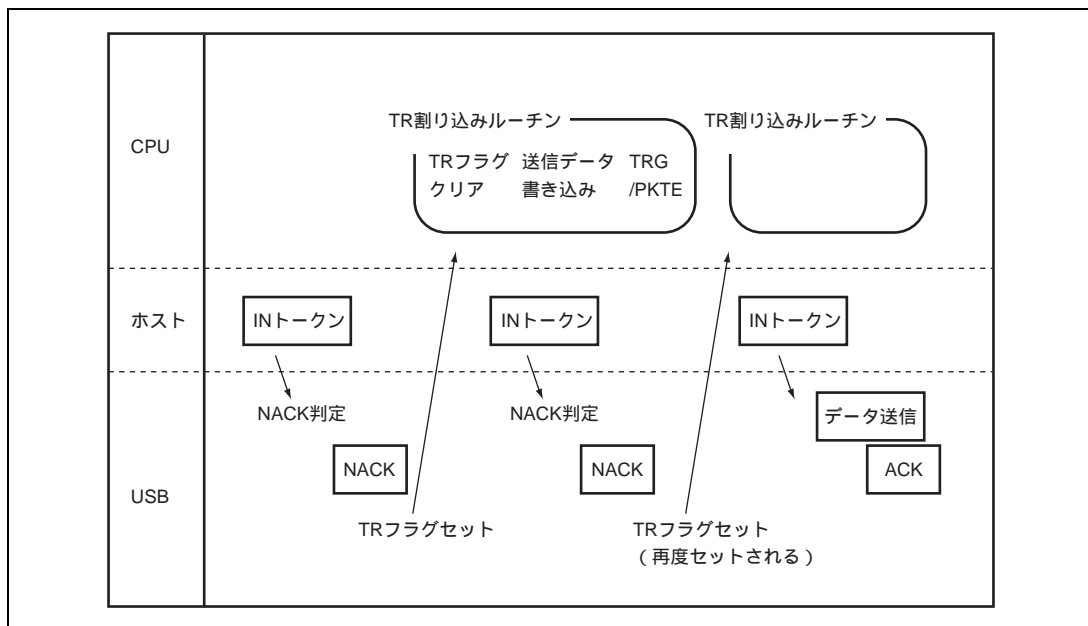


図 20.20 TR 割り込みフラグのセットタイミング

20.8.7 ソフトウェアスタンバイモードへの遷移と解除時の注意

ソフトウェアスタンバイモードへの遷移、および解除のフロー例を図 20.21 に示します。

ソフトウェアスタンバイ状態が解除され、再度 USBF をアクセスするときは、USB クロック (48MHz) の動作安定時間を待つようにしてください。

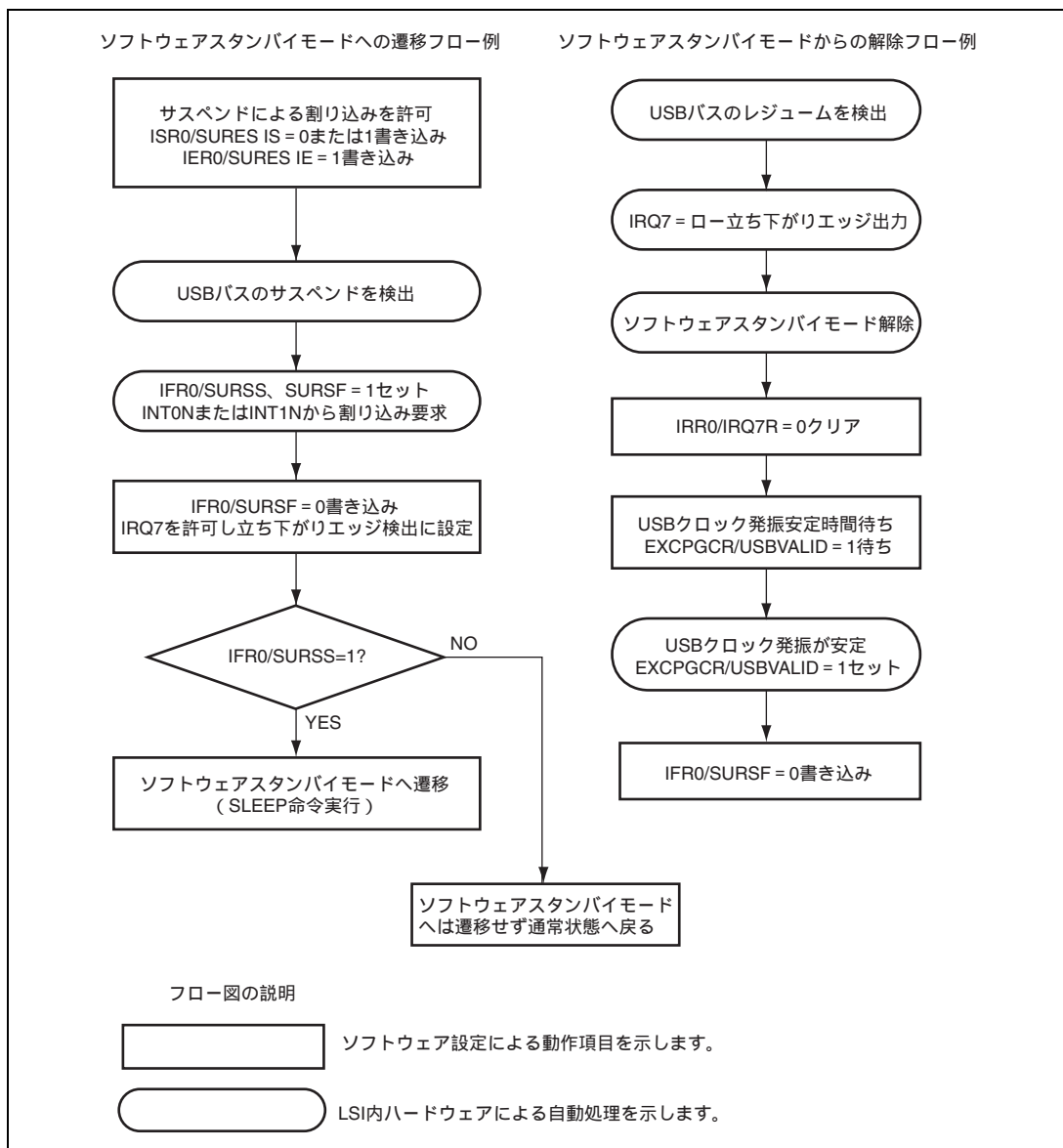


図 20.21 ソフトウェアスタンバイモードへの遷移と解除のフロー例

20.8.8 バスパワード機能を使用する場合のメインクロックについて

本 LSI のメインクロック (EXTAL から入力) をブルートゥースインタフェース用のクロック (RDI_REFCLK_IN から入力) と同じようにブルートゥース用 RF チップから入力すると、サスペンド状態からの復帰時にクロック供給を再開する手段がありませんので、バスパワード機能を使用する場合は常時アクティブ状態のクロックをメインクロックとして使用する必要があります。また、サスペンド時の消費電力を最小限に抑えるため、メインクロックは水晶振動子を EXTAL および XTAL 端子に接続する機能を使用することをお勧めします。

20.8.9 周辺クロックについて

本 USB ファンクションコントローラを使用する際は、本 LSI の周辺モジュールで使用する周辺クロック (P) の周波数を 13MHz より大きい周波数で使用してください。周辺クロックの周波数が 13MHz 以下では正常に動作しません。周辺クロックの詳細な設定方法については「第 11 章 クロックパルス発振器 (CPG)」を参照してください。

21. ブルートゥースインタフェース (BT)

本 LSI は、ブルートゥースベースバンド処理に必要なブルートゥースインタフェース(BT)を内蔵しています。本 BT は、ブルートゥース規格 (ver1.2) に従い、ハードウェアとファームウェアとの組み合わせで構成されており、ホストコントローラインタフェース (HCI:Host Controller Interface) 階層までの下位プロトコルスタック機能を備えています。本 LSI は本機能を搭載することにより、ベースバンド階層 (BB) およびリンクマネージメントプロトコル階層 (LMP) について LOGO 認証を取得予定です。

上位プロトコルスタックを構築するために、ブルートゥース規格 (ver1.2) で規定されている HCI コマンドをサポートしているほか、独自に TCI (Test Command Interface) コマンドをサポートしています。実際の上位プロトコルスタックと下位プロトコルスタックとのコミュニケーションのための関数を API (Application Interface) として用意しています。独自サポートの TCI コマンドおよび API 関数の詳細資料がご入り用の場合は、別途弊社営業窓口までご相談ください。

21.1 特長

- ブルートゥース規格 (v1.2) に対応
- 拡散変調は周波数ホッピング方式で79チャンネルホッピング
- ACL (Asynchronous Connection Less) およびSCO (Synchronous Connection Oriented) リンクをサポート
- ACL、SCO共にHCIの上位階層から制御可能
- リンク・接続の確立の制御系パケット、データの送受信パケット、全タイプのパケットをサポート
- 接続形態は 1 : 1 通信 (point to point)、ピコネット (point to multi-point) をサポート
- Voice Codec方式ではA-law - μ -law、CVSD - Linear PCM、Linear PCM - A-law/ μ -law、A-law/ μ -law - CVSD 相互変換が可能
- Encryption/Decryption
- Hold / Sniff / Park モードの3つの低消費電力状態をコントロール
- ルネサス テクノロジ製RF IC (HD157102NP) と直結可能なインタフェースをサポート
- CPUを介さずに以下の2種類のVoice Codec ICと直結可能なインタフェースをサポート
 - (1) STMicroelectronics社製STLC7550
 - (2) Motorola社製MC145483
- 低消費電力状態で使用する低周波数クロックとして、32.768kHzの水晶発振子を利用可能な周波数補正機能をサポート
- HCIコマンドのほかに独自のTCIコマンド、API関数をサポート

21. ブルートゥースインタフェース (BT)

BTのブロック図を図 21.1 に示します。

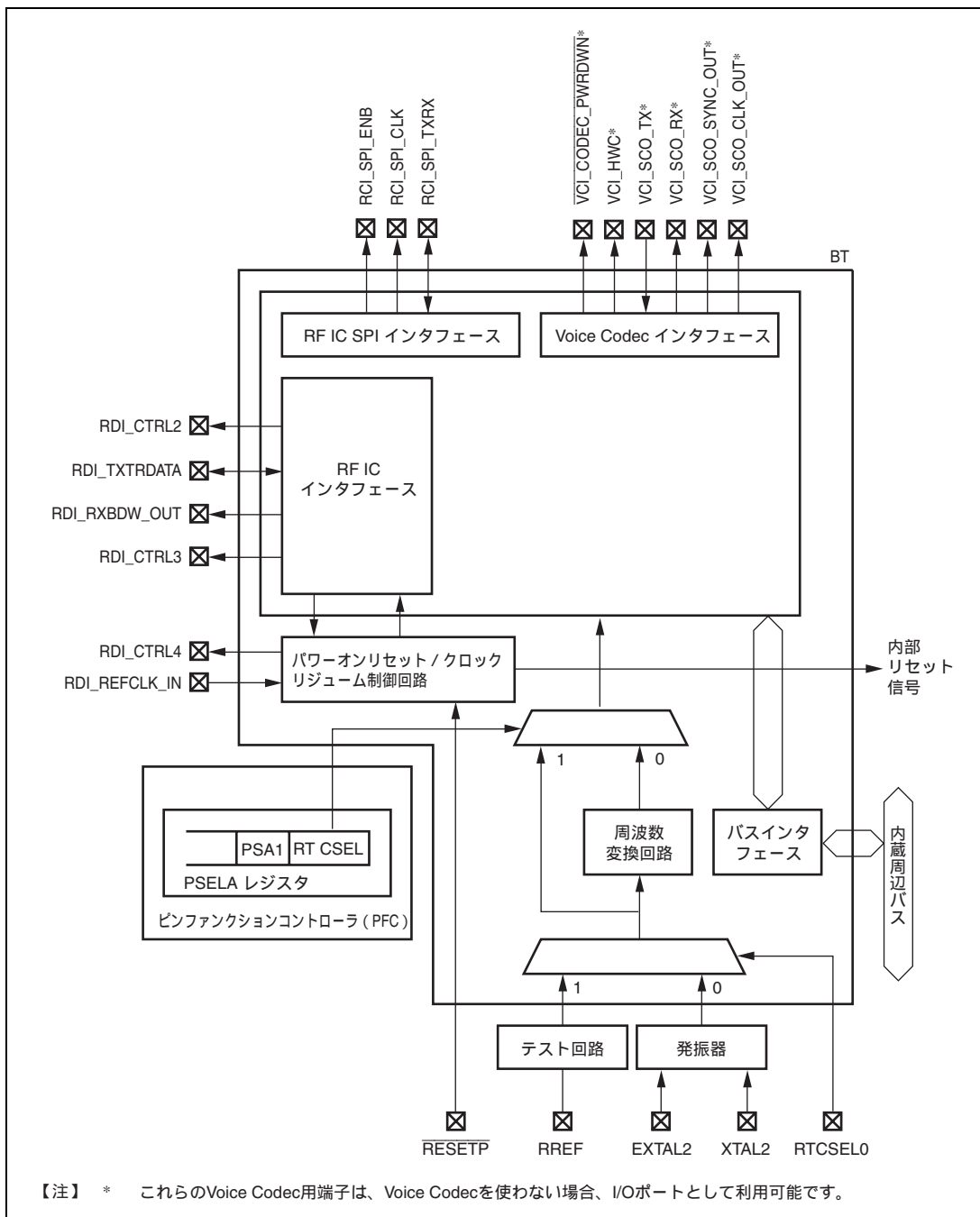


図 21.1 BTのブロック図

21.2 入出力端子

BTの入出力端子を表 21.1 に示します。

表 21.1 端子構成

区分	端子名	入出力	機能
RF IC インタフェース	RDI_CTRL2	出力	クラス 1 対応用外付けパワーアンプ制御出力
	RDI_TXTRDATA	入出力	RF IC 用送受信データ入出力
	RDI_RXBDW_OUT	出力	RF IC 用パケット制御出力
	RDI_REFCLK_IN	入力	RF IC とのデータ入出力用クロックおよび BT 用メインクロック入力
	RDI_CTRL3	出力	RF IC 内発振器をイネーブルにするストロープ出力
	RDI_CTRL4	出力	RF IC 用リセット制御、インタフェース用電源および低消費電力制御出力
RF IC との SPI インタフェース	RDI_SPI_TXRX	入出力	SPI シリアルデータ入出力
	RDI_SPI_CLK	出力	RF IC との SPI インタフェースクロック出力
	RDI_SPI_ENB	出力	RF IC との SPI インタフェースイネーブル出力
Voice Codec インタフェース	VCI_SCO_CLK_OUT	出力	Voice Codec IC 用クロック出力* ¹
	VCI_SCO_SYNC_OUT	出力	Voice Codec IC 用フレーム同期信号出力* ¹ * ³
	VCI_SCO_RX	出力	Voice Codec IC への受信データ出力* ¹
	VCI_SCO_TX	入力	Voice Codec IC からの送信データ入力* ¹
	VCI_HWC	出力	Voice Codec IC 用動作モード選択出力* ¹
	VCI_CODECPWRDWN	出力	Voice Codec IC 用パワーダウン制御出力* ¹
クロック選択	RTCSEL0	入力	テスト端子* ²
	RREF	入力	テスト端子* ²
	EXTAL2	入力	水晶発振子接続端子
	XTAL2	入力	水晶発振子接続端子

【注】 *¹ Voice Codec インタフェース端子は、Voice Codec を接続しない場合、I/O ポート (ポート E) として使うことができます。詳細については、「第 23 章 I/O ポート」を参照してください。

*² RTCSEL0 端子はロー固定、RREF 端子は Vcc ヘブルアップしてお使いください。

*³ VCI_SCO_SYNC_OUT 端子は、BREQ 端子とマルチプレクスされていますので、電源投入時の誤動作を防ぐために、プルアップしてご使用ください。

21.3 レジスタの説明

BTにあるレジスタは、ユーザから直接アクセスするものではなく、すべて付属するファームウェアがアクセスしますので、ここでは説明を記載していません。付属ファームウェアの機能については、別冊のマニュアルを参照してください。

端子から入力されるパワーオンリセットでのみ初期化されます。WDT の発生するパワーオンリセットとマニュアルリセットでは初期化されません。

21.4 RF IC との接続

BT は、ルネサス テクノロジ製 RF IC (HD157102NP) と直結可能なインタフェースを備えています。

21.4.1 RF IC との接続

図 21.2 に本 LSI とルネサス テクノロジ製 RF IC (HD157102NP) との接続例を示します。パワーオンリセットが印加されると、リセット信号が BT のパワーオンリセット / クロックリジューム制御回路に送られ、RDI_CTRL4 信号から RF IC に伝えられます。RDI_CTRL4 信号は、そのまま RF IC 側のインタフェース部分の電源供給端子となります。並行してリセット状態をホールドし、外部からのリセット信号がネゲートされた後も引き続き本 LSI の内部リセット状態を保持します。

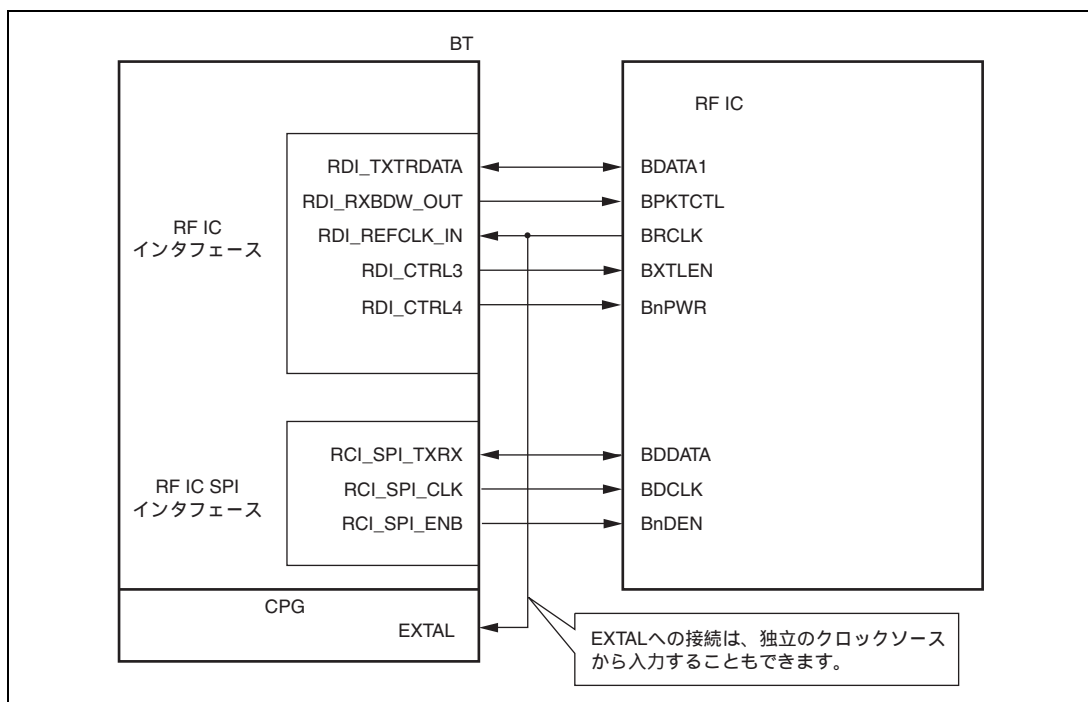


図 21.2 RF IC との接続例

RF IC はリセットによる初期化終了後、クロック信号 (RDI_REFCLK_IN) を出力し始めます。BT のパワーオンリセット / クロックリジューム制御回路は RF IC からのクロックを受けてカウンタ回路がカウントを開始し、所定のカウンタ後、リセット保持状態を解除します。

RF IC から送られてくるクロック信号 (RDI_REFCLK_IN) は、本 LSI 内蔵 CPG の EXTAL 端子に接続することにより、本 LSI 全体のメインクロックとして利用することも可能です。これにより、ブルートゥースのシステム全体を構成するのに必要な水晶発振器の個数を節約することができます。この場合、RDI_REFCLK_IN は 13MHz です。本 LSI 全体のメインクロック周波数は自動的に 13MHz となります。

EXTAL 端子に RDI_REFCLK_IN と同じクロックを接続せずに独立のクロックを入力することも可能です*。その場合、メインクロック周波数は本 LSI が保証する範囲で自由に入力周波数を選ぶことが可能です。入力可能なクロック周波数の範囲については、「第 28 章 電気的特性」を参照してください。EXTAL 端子の機能の詳細については、「第 11 章 クロックパルス発振器 (CPG)」を参照してください。

【注】 * 本 LSI が内蔵している USBF のバスパワー機能を使用する場合は、必ず RDI_REFCLK_IN に入力するクロックから独立したクロックを入力し、常時クロック入力にアクティブ状態にしてください。

21.5 Voice Codec IC との接続

BT は、CPU を介さずに SCO リンクで直接 RF IC とデータをやり取りする機能を備えており、外部の Voice Codec との接続としては、STMicroelectronics 社製 STLC7550 と MOTOROLA 社製 MC145483 との接続をサポートしています。

21.5.1 Voice Codec (STLC7550) との接続

図 21.3 に BT と Voice Codec (STLC7550) との接続例を示します。STLC7550 側の SCLK 端子は使用しません。

VCI_SCO_SYNC_OUT 端子は、 $\overline{\text{BREQ}}$ 端子とマルチプレクスされていますので、電源投入時の誤動作を防ぐために、プルアップしてご使用ください。

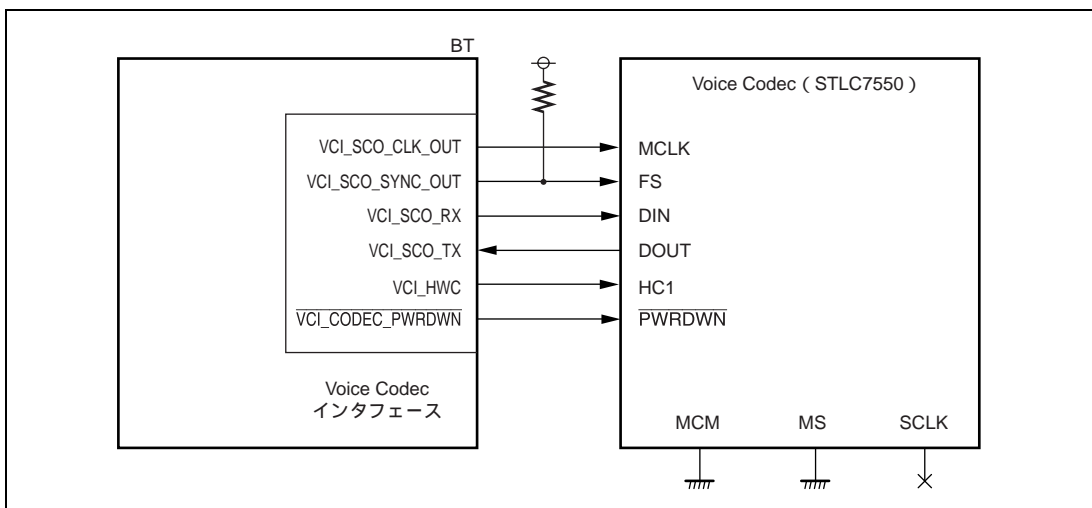


図 21.3 Voice Codec (STLC7550) との接続例

21. ブルートゥースインタフェース (BT)

21.5.2 Voice Codec (MC145483) との接続

図 21.4 (1) に BT と Voice Codec (MC145483) との接続例を示します。VCI_SCO_SYNC_OUT 端子は、 $\overline{\text{BREQ}}$ 端子とマルチプレクスされていますので、電源投入時の誤動作を防ぐために、プルアップしてご使用ください。また、タイミングを図 21.4 (2) に示します。

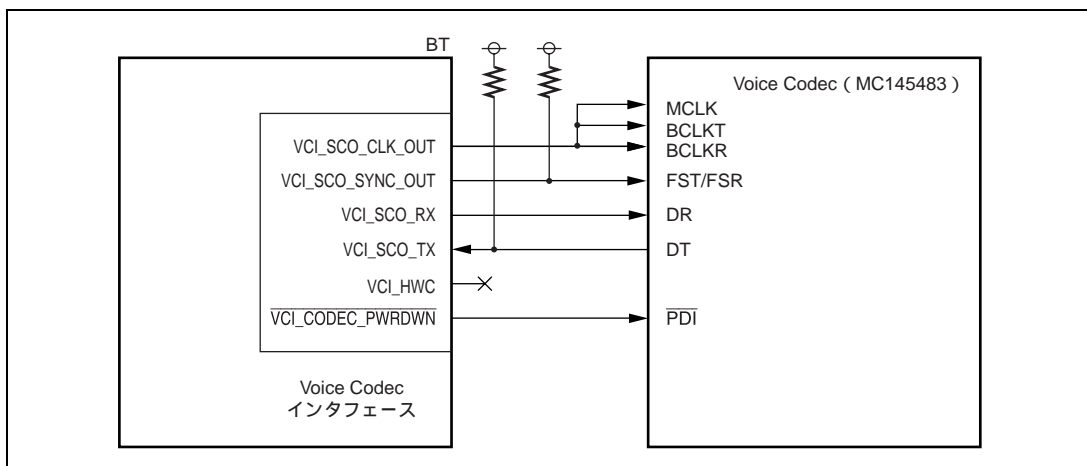


図 21.4 (1) Voice Codec (MC145483) との接続例

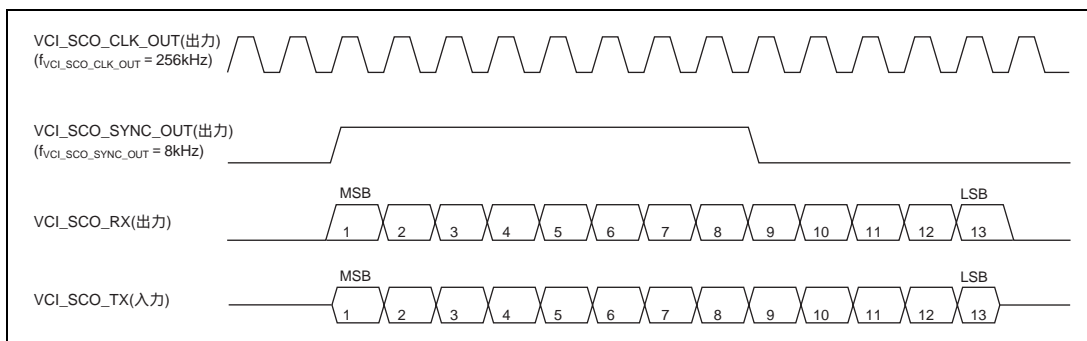


図 21.4 (2) Voice Codec (MC145483) と接続した場合のタイミング

21.6 低周波数クロック発振器の接続

ブルートゥースの低消費電力状態のモード (Sniff/Park/Hold) に対応するため、BT は外部から 32kHz または 32.768kHz のクロックを受け取るインタフェースを持っています。BT にクロックを供給する方法としては以下の 4 つがあり、ピンセレクトレジスタ (PSELA) の RTCSEL1 ビットによって選択します。ピンセレクトレジスタ A (PSELA) についての詳細は、「第 24 章 ピンファンクションコントローラ (PFC)」を参照してください。

1. EXTAL2とXTAL2端子に32.0kHzの水晶振動子を接続し、内部発振器を動作させて供給。
2. EXTAL2とXTAL2端子に32.768kHzの水晶振動子を接続し、内部発振器を動作させ、さらに周波数変換回路で擬似的に32kHzを生成して供給する。
3. EXTAL2から32.0kHzのクロックを直接入力する。
4. EXTAL2から32.768kHzのクロックを直接入力し、さらに周波数変換回路で擬似的に32kHzを生成して供給する。

表 21.2 に選択条件と機能の一覧を示します。

表 21.2 選択条件と機能一覧

RTCSEL1 ビット	機 能
1	EXTAL2/XTAL2 端子に 32.0kHz 水晶振動子を接続し、内部発振器を動作させて生成して入力します。もしくは、EXTAL2 より 32kHz クロックを直接入力して使用します。
0	EXTAL2/XTAL2 端子に 32.768kHz の水晶振動子を接続し、内部発振器を動作させて生成して入力します。もしくは、EXTAL2 より 32.768kHz クロックを直接入力して使用します。

21.6.1 RTCSEL1 ビットによる選択機能

ピンセレクトレジスタ A (PSELA) の RTCSEL1 ビットにより、BT にある周波数変換回路を動作させるかどうかを選択します。周波数変換回路は、32.768kHz から疑似的な 32kHz クロックを生成します。この周波数変換回路を動作させることにより、EXTAL2 と XTAL2 端子に 32.0kHz の水晶振動子を接続する代わりに、リアルタイムクロック (RTC) 用に使われる 32.768kHz の水晶振動子を接続することができます。

RTCSEL1 ビットに 0 をセットすると、周波数変換回路が動作します。RTCSEL1 ビットに 1 をセットすると、周波数変換回路は非活性状態となり、32.768kHz の周波数をバイパスします。RTCSEL0 端子をハイにした場合は、RTCSEL1 ビットは常に 1 として使用してください。RTCSEL1 ビットの初期状態は 0 です。ピンセレクトレジスタ A (PSELA) についての詳細は、「第 24 章 ピンファンクションコントローラ (PFC)」を参照してください。

21.6.2 周波数変換回路の動作説明

32.768kHz の 128 クロック分は 32kHz の 125 クロック分に相当します。これを利用して図 21.5 に示すように 32.768kHz のクロック信号を各ステートごとにカウントし、128 クロックを 43(ステート 0)、42(ステート 1)、43(ステート 2) に区切り、それぞれの区間ごとにクロック 1 つの立ち下がりを取り去り、128 クロックのサイクルで 125 クロックとなるようにします。このよう調整すると、32kHz 信号 125 クロック分の区間で、元の 32.768kHz 信号は 125 クロックとなり、32000 クロック / 秒となります。調整した信号により 32kHz クロックを生成します。

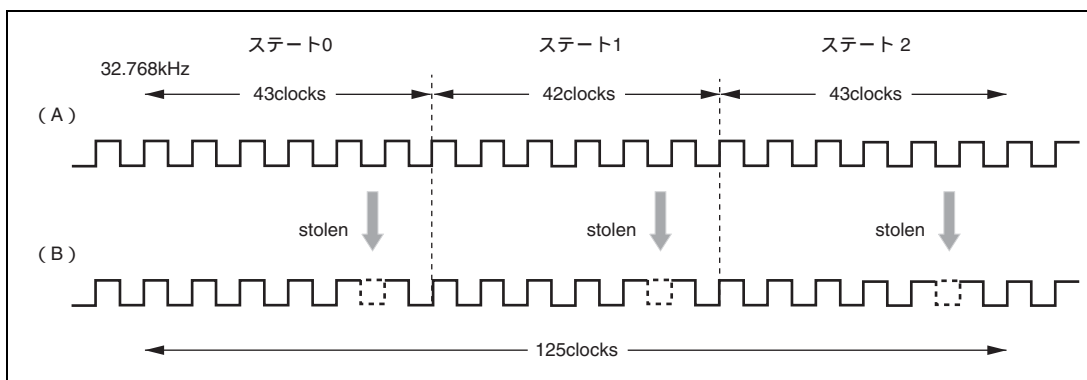


図 21.5 周波数変換回路の機能

32kHz のクロック信号の実経過時間と 32.768kHz から生成した 32kHz クロック信号の実経過時間との時間的誤差は図 21.6 のようになります。125 カウントごとに (A)、(B) は一致しており、最大の誤差は $30\mu\text{s}$ 程度となります。

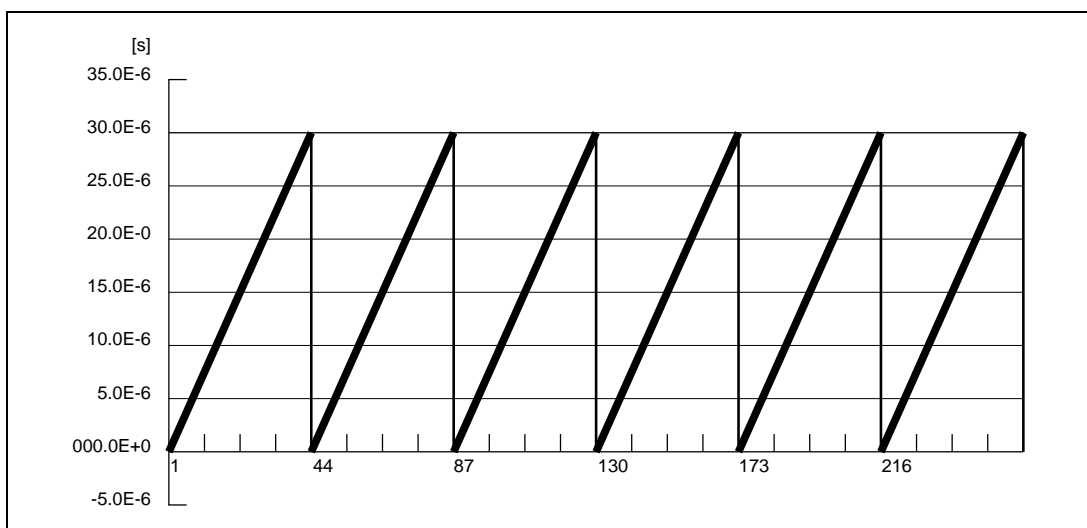


図 21.6 32.768kHz から変換した疑似 32kHz クロックの誤差

21.6.3 外部水晶振動子接続時の注意

外部水晶振動子を EXTAL2、XTAL2 に接続するときの注意事項を図 21.7 に示します。

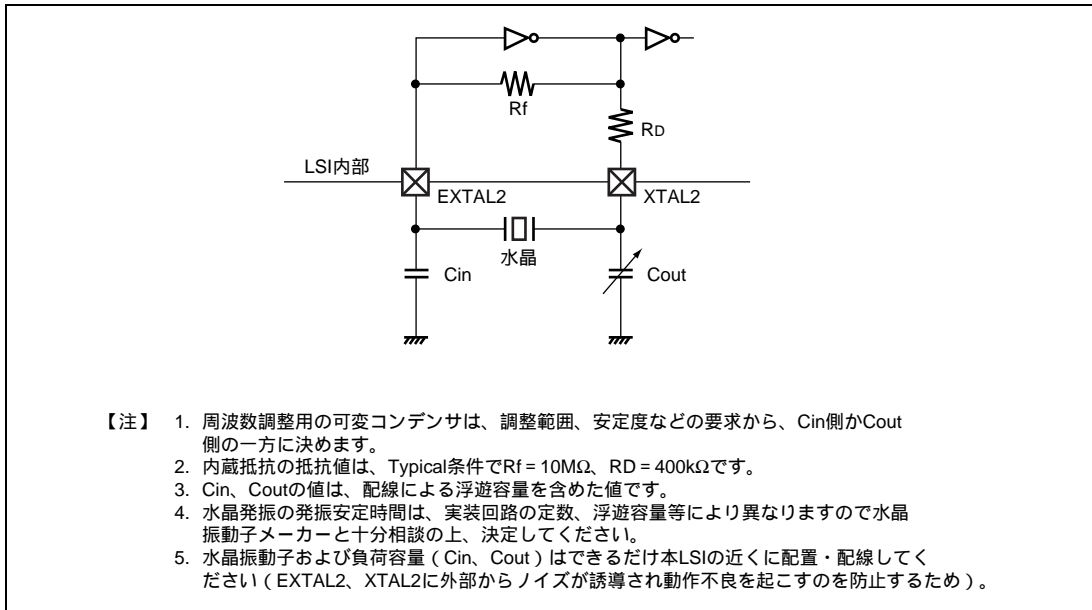


図 21.7 水晶発振子使用時の注意

21.7 パワーオンリセット/クロックリジューム制御機能

本 LSI は RF IC と直結し、かつ必要な外付け部品を最小限にするための機能を BT に備えています。

21.7.1 パワーオンリセット

本 LSI はルネサス テクノロジ製 RF IC (HD157102NP) から供給される RF IC とのインタフェース用タイミングクロックをそのまま本 LSI のメインクロックとしても利用可能です (図 21.2 を参照)。ルネサス テクノロジ製 RF IC はパワーオンリセットをネゲートして一定の時間経過後にクロック出力を開始しますが、本 LSI はパワーオンリセット中にクロックを供給し、内蔵発振器の発振を安定させる必要があります。そのため外部からのリセット入力ネゲートされた後も内部でリセットを保持し、外部からのクロック供給開始後、所定の時間カウントしてから内部リセット信号をネゲートする機能を備えています。タイミングチャートを以下に示します。

内部リセット信号は 2 種類あり、USB 用のリセット信号とそれ以外のモジュール用の内部リセット信号に分かれています。USB 用のリセット信号のアサート期間は他のモジュールとは異なります。図 21.8 にタイミングチャートを示します。

21. ブルートゥースインタフェース (BT)

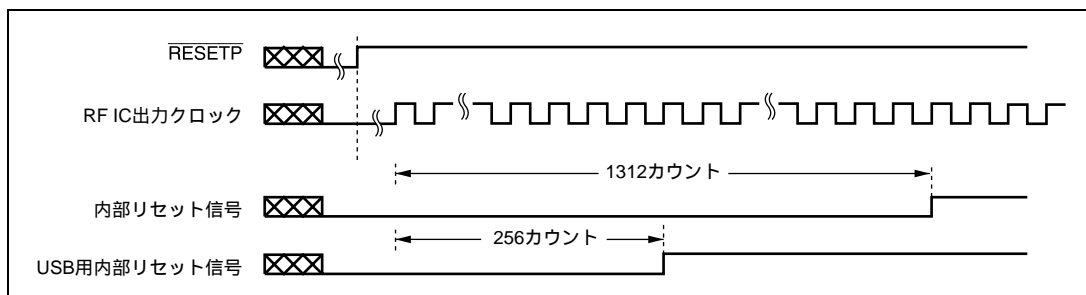


図 21.8 リセット信号のタイミング

21.7.2 クロックリジューム制御

低消費電力状態において、本 LSI から RF IC に対するコマンドで、RF IC から供給されるクロック出力を停止させることができます。このとき、RF IC から供給されるクロック出力を本 LSI のメインクロックとして使用している場合、本 LSI の動作も停止することになりますが、BT は別に供給されている 32kHz または 32.768kHz のクロックで動作することになります。この状態から復帰するために、本 LSI はクロックリジューム制御機能を備えています。クロックリジューム制御は次のシーケンスを行います。

1. BTより割り込みを発生
2. BTよりRF ICに起動コマンドを発行
3. INTCが割り込みを受け付け
4. CPU起動

21.8 Bluetooth HCI/TCI コマンドと API

本 LSI は、上位プロトコルスタックを構築するために、ブルートゥース規格 (ver1.2) で規定されている HCI (Host Control Interface) コマンドおよび TCI (Test Control Interface) コマンドをサポートしている他、独自に TCI コマンドをサポートしています。ブルートゥース規格 (ver1.2) で規定されている HCI コマンドおよび TCI コマンドについての詳細は、ブルートゥース規格 (ver1.2) を参照してください。独自サポートの TCI コマンドの詳細資料が御入用の場合は、別途弊社営業窓口まで御相談ください。

また、本 LSI は HCI より下位層のプロトコルスタックファームウェアをライブラリとして用意しており、開発したプログラムにブルートゥースインタフェースを容易に追加できるようにしています。また、本ライブラリのアプリケーションインタフェース (API) は、ブルートゥース規格の H4 をベースとしているので、さまざまなブルートゥースプロトコルスタックと親和性が良く、容易に本 LSI に移植できます。本ライブラリのアプリケーションインタフェース (API) として、関数が用意されています。それぞれの関数の詳細資料が御入用の場合は、別途弊社営業窓口まで御相談ください。

21.9 基板設計の制約事項

- 現象

本 LSI への電源投入 (パワーオンリセット) 時においてリセットシーケンスが正しく行われず、デバイスが起動しない可能性があります。

この現象が起こる条件は、

- (a) 本 LSI の EXTAL 端子の入力を、弊社製 Bluetooth 用 RF IC (HD157102NP) からのクロック出力 (BRCLK) から供給した場合

および

- (b) 電源投入 (パワーオンリセット) 時に本 LSI の $\overline{\text{RESETP}}$ 端子と $\overline{\text{TRST}}$ 端子に同一のパワーオンリセット信号が入力される構成の場合

上記 (a) および (b) が同時に成り立つ基板設計の場合に限ります (図 21.9 参照)

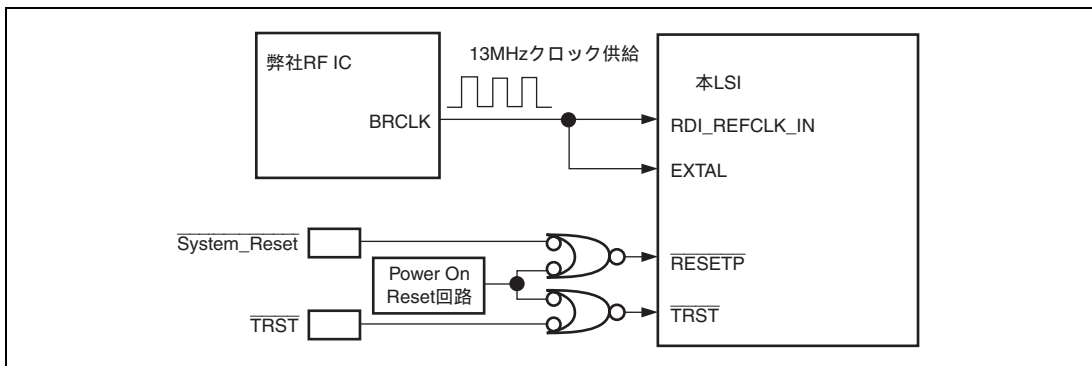


図 21.9 本 LSI が起動しない場合

- 対策

本 LSI のリセットシーケンスを正常に動作させるために、下記の 1. ~ 3. のいずれかの対策を基板上に施してください。

- 図 21.10 に示すように $\overline{\text{TRST}}$ 端子にスイッチを設け、通常動作時は $\overline{\text{TRST}}$ 端子を GND 固定とし、またバウンダリスキャンおよび E10A 使用時は $\overline{\text{TRST}}$ 端子への信号入力が可能になるようにします。なお、バウンダリスキャンおよび E10A をご使用にならない基板の場合は、スイッチを設けず $\overline{\text{TRST}}$ 端子を GND 固定することもできます。なお、 $\overline{\text{TRST}}$ 端子にはバウンダリスキャンの規格により LSI 内部でプルアップ抵抗を接続しておりますので、GND 固定した場合には 30 ~ 110 μA 程度のリーク電流が生じます。

21. ブルートゥースインタフェース (BT)

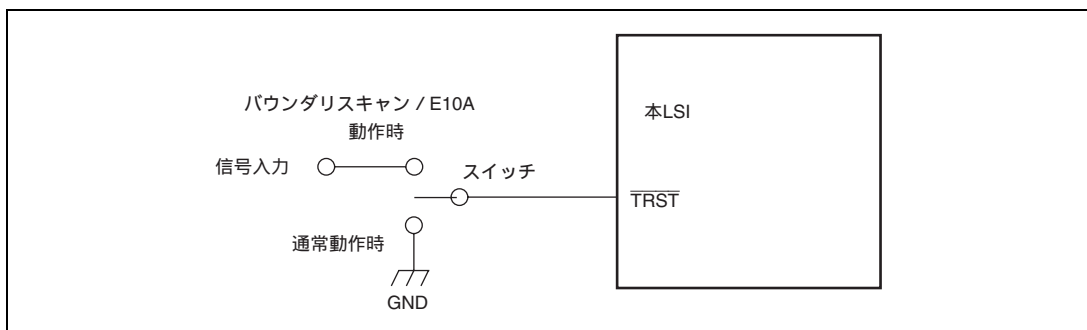


図 21.10 対策案 1

2. 図21.11に示すように、弊社製Bluetooth用RF IC (HD157102NP) からのクロック出力 (BRCLK) は RDI_REFCLK_IN端子のみに接続し、EXTAL端子にはRDI_REFCLK_IN端子に供給するクロックとは別のクロック源を入力するようにします。本LSIへのEXTAL、 $\overline{\text{RESETP}}$ 、 $\overline{\text{TRST}}$ のタイミングチャートおよびAC特性については、「28.3.1 クロックタイミング」の図28.8、28.9および表28.5を参照してください。
- また図21.12のように、本LSIを単独で使用する場合についてのEXTAL、 $\overline{\text{RESETP}}$ 、 $\overline{\text{TRST}}$ のタイミングチャートおよびAC特性については、「28.3.1 クロックタイミング」の図28.6、28.7および表28.5を参照してください。

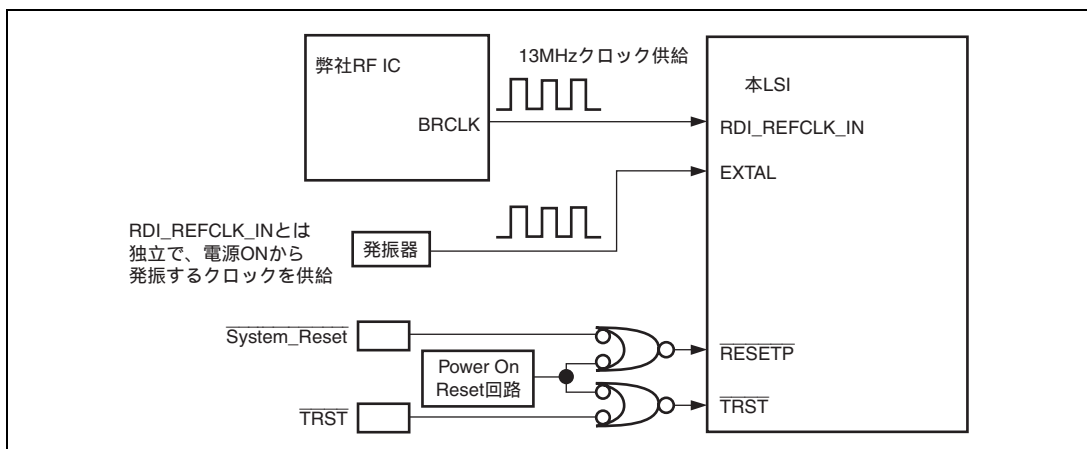


図 21.11 対策案 2

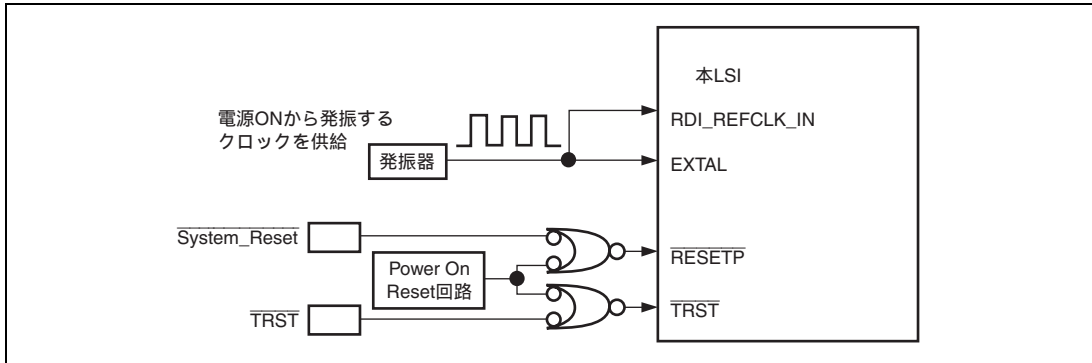


図 21.12 本 LSI を単独で使用する場合の対策案

3. 図21.13に示すように、電源投入時の $\overline{\text{TRST}}$ 端子および $\overline{\text{RESETP}}$ 端子に供給する信号を個別に供給するようにします。この場合の $\overline{\text{TRST}}$ および $\overline{\text{RESETP}}$ のタイミングチャートおよびAC特性については、「28.3.1 クロックタイミング」の図28.10、28.11および表28.5を参照してください。

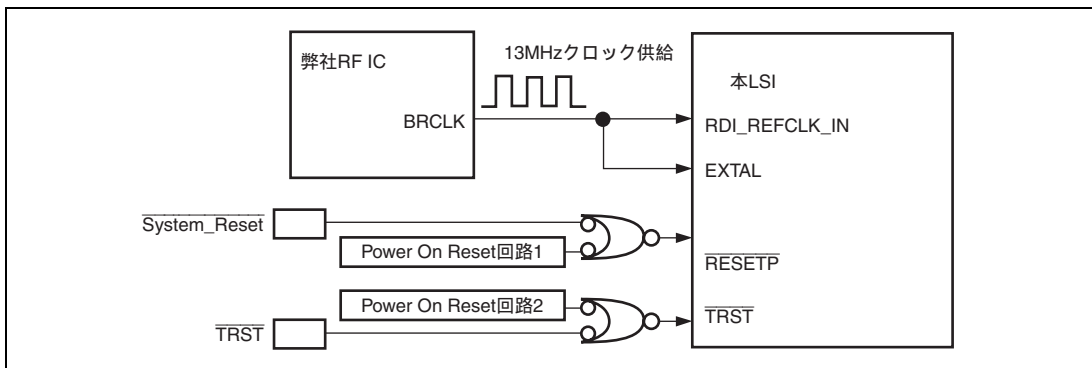


図 21.13 対策案 3

22. D/A コンバータ (DAC)

本 LSI には 2 チャンネルの D/A コンバータ (DAC) を搭載しています。

22.1 特長

- 8ビットの解像度
- 出力チャンネル：2チャンネル
- 変換時間：最小10 μ s (容量性負荷：20pF)
- 出力電圧：0V \sim AV_{cc} (アナログ部電源)

DAC のブロック図を図 22.1 に示します。

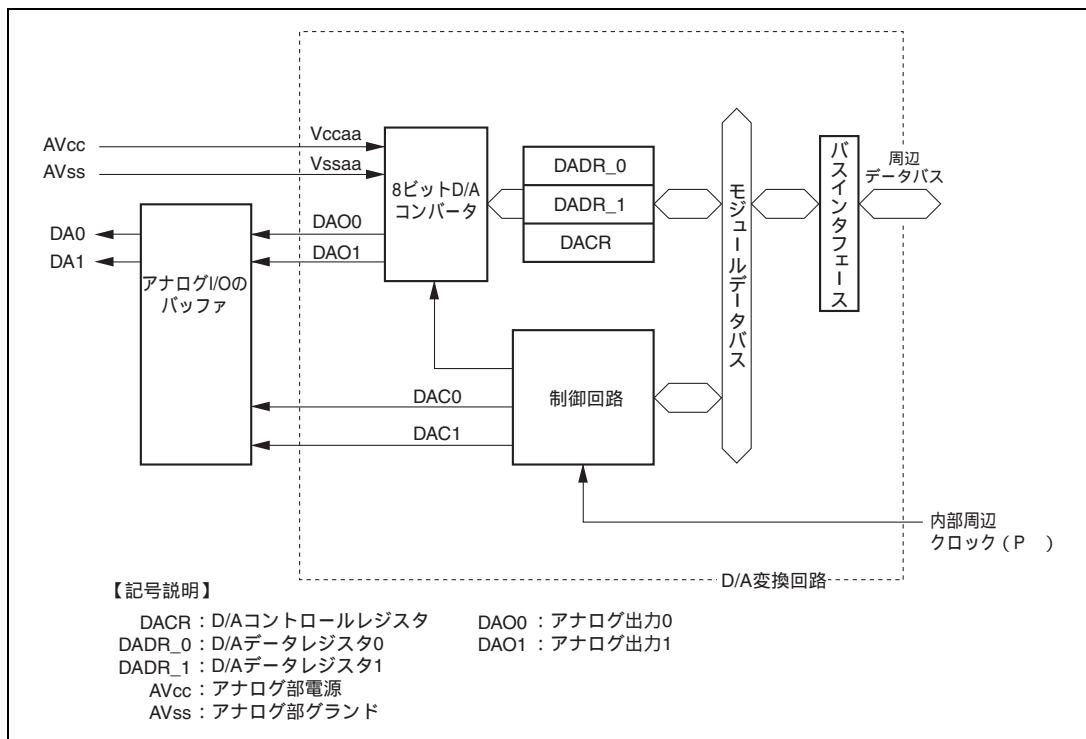


図 22.1 DAC のブロック図

22.2 入出力端子

DAC の端子構成を表 22.1 に示します。

表 22.1 端子構成

端子名	入出力	機能
AVcc	-	アナログ部の電源および D/A 変換基準電圧
AVss	-	アナログ部のグランド
DA0	出力	チャンネル 0 のアナログ出力
DA1	出力	チャンネル 1 のアナログ出力

22.3 レジスタの説明

DAC には、以下のレジスタがあります。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- D/A データレジスタ 0 (DADR_0)
- D/A データレジスタ 1 (DADR_1)
- D/A コントロールレジスタ (DACR)

22.3.1 D/A データレジスタ 0、1 (DADR_0、1)

DADR_0、DADR_1 は、変換データを格納する 8 ビットの読み出し / 書き込み可能なレジスタです。DACR の D/A 出力イネーブル (DAOE1、DAOE0) が 1 のとき、DADR_0、DADR_1 に設定された値を常に D/A 変換して、アナログ出力端子 (DA0、DA1) へ出力します。DADR_0、DADR_1 は、リセット時 H'00 に初期化されます。ソフトウェアスタンバイ、およびモジュールスタンバイでは初期化されません。

ビット	ビット名	初期値	R/W	説明
7~0		H'00	R/W	変換データを格納する 8 ビットのレジスタです。

22.3.2 D/A コントロールレジスタ (DACR)

DACR は、DAC の動作を制御する 8 ビットの読み出し / 書き込み可能なレジスタです。DACR は、リセット時 H'3F に初期化されます。ソフトウェアタンバイ、およびモジュールスタンバイでは初期化されません。

ビット	ビット名	初期値	R/W	説明
7	DAOE1	0	R/W	チャンネル 1 の D/A 変換とアナログ出力を制御します。 0 : チャンネル 1 の D/A 変換とアナログ出力 (DA1) を禁止 1 : チャンネル 1 の D/A 変換とアナログ出力 (DA1) を許可
6	DAOE0	0	R/W	チャンネル 0 の D/A 変換とアナログ出力を制御します。 0 : チャンネル 0 の D/A 変換とアナログ出力 (DA0) を禁止 1 : チャンネル 0 の D/A 変換とアナログ出力 (DA0) を許可
5~0		すべて 1	R	リザーブビット 読み出すと常に 1 が読み出されます。書き込む値も常に 1 にしてください。 0 を書いた場合の動作の保証はできません。

22.4 動作説明

DAC には、独立して変換を実行できる 2 つの D/A 変換回路を内蔵しています。

DAC は DACR でアナログ出力を許可状態のとき、常に D/A 変換が実行されます。DADR_0、DADR_1 の値を変更すると、新しいデータの変換を直ちに開始します。DACR の DAOE1 ビット、DAOE0 ビットを 1 にセットすると変換結果が出力されます。

チャンネル 0 の D/A 変換の例を次に示します。タイミングは図 22.2 に示します。

1. 変換するデータを DADR_0 に書き込みます。
2. DACR の DAOE0 ビットを 1 にセットします。D/A 変換を開始し、DA0 は出力許可になります。変換結果は変換時間後に出力されます。出力値は、 $(\text{DADR_0内容} / 256) \times AV_{CC}$ です。この変換結果の出力は、DADR_0 の値を変更するか、DACR の DAOE0 ビットを 0 にクリアするまで続きます。
3. DADR_0 の値を変更すると、変換を再び開始し、結果は変換時間の後で出力されます。
4. DACR の DAOE0 ビットを 0 にクリアすると、アナログ出力 (DA0) は禁止 (Hi-Z) になります。

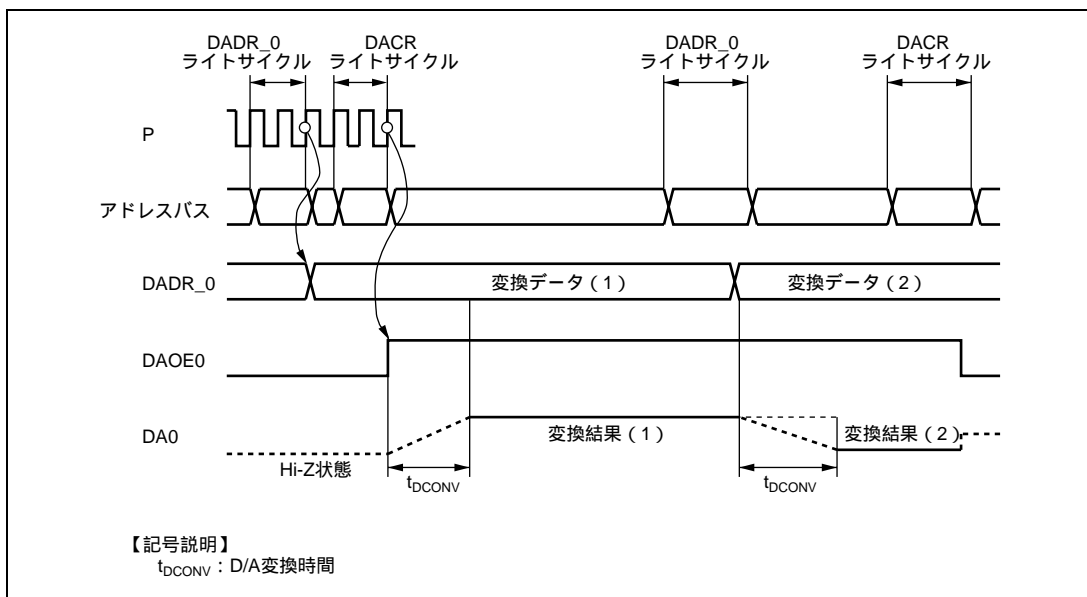


図 22.2 D/A 変換動作の例

23. I/O ポート

本 LSI には 6 本の I/O ポート (ポート A、B、D、E、G および H) があります。それぞれのポートの端子は、すべて他の端子機能 (ピンファンクションコントローラ (PFC) で端子機能とプルアップ MOS 制御の選択を行います) を兼ねているマルチプレクス端子です。ポートはそれぞれ、端子のデータを格納するためのデータレジスタを 1 本ずつ持っています。

23.1 ポート A

ポート A は図 23.1 に示すような端子構成を持つ 7 ビットの入出力ポートです。各端子には入力プルアップ MOS があり、PFC のポート A コントロールレジスタ (PACR) で制御します。

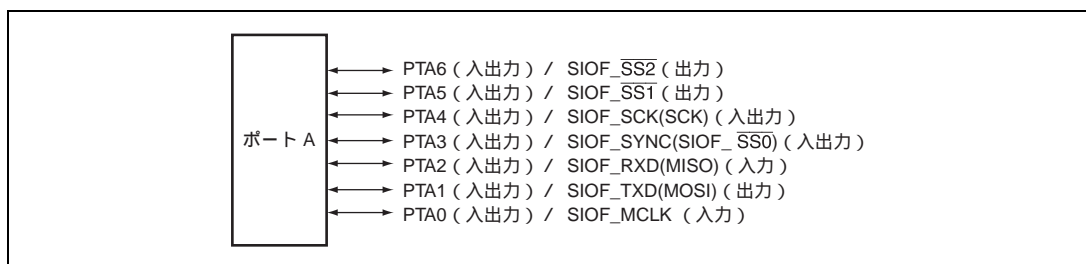


図 23.1 ポート A

23.1.1 レジスタの説明

ポート A には以下のレジスタがあります。このレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- ポート A データレジスタ (PADR)

23.1.2 ポート A データレジスタ (PADR)

PADR は、読み出し / 書き込み可能な 7 ビットのレジスタで、端子 PTA6 ~ PTA0 のデータを格納します。PA6DT ~ PA0DT ビットは PTA6 ~ PTA0 端子に対応しています。端子機能が汎用出力ポートの場合には、ポートを読み出すと、対応する PADR ビットの値が直接読み出されます。端子機能が汎用入力ポートの場合には、ポートを読み出すと、対応する端子レベルが読み出されます。

ビット	ビット名	初期値	R/W	説明
7		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 としてください。 1 を書いた場合の動作の保証はできません。
6	PA6DT	0	R/W	表 23.1 に PADR の機能を示します。
5	PA5DT	0	R/W	
4	PA4DT	0	R/W	
3	PA3DT	0	R/W	
2	PA2DT	0	R/W	
1	PA1DT	0	R/W	
0	PA0DT	0	R/W	

表 23.1 ポート A データレジスタ (PADR) の読み出し / 書き込み動作

PACR* ¹ の状態		端子状態	読み出しの場合	書き込みの場合
PAnMD1* ²	PAnMD0* ²			
0	0	その他の機能	PADR の値	PADR に書き込めるが、端子の状態に影響しない
	1	出力	PADR の値	書き込み値が端子から出力される
1	0	入力 (プルアップ MOS オン)	端子の状態	PADR に書き込めるが、端子の状態に影響しない
	1	入力 (プルアップ MOS オフ)	端子の状態	PADR に書き込めるが、端子の状態に影響しない

【注】 *1 ポート A コントロールレジスタ (PACR) については、「第 24 章 ピンファンクションコントローラ (PFC)」を参照してください。

*2 n = 0 ~ 6

23.2 ポート B

ポート B は図 23.2 に示すような端子構成を持つ 8 ビットの入出力ポートです。各端子には入力プルアップ MOS があり、PFC のポート B コントロールレジスタ (PBCR) で制御します。

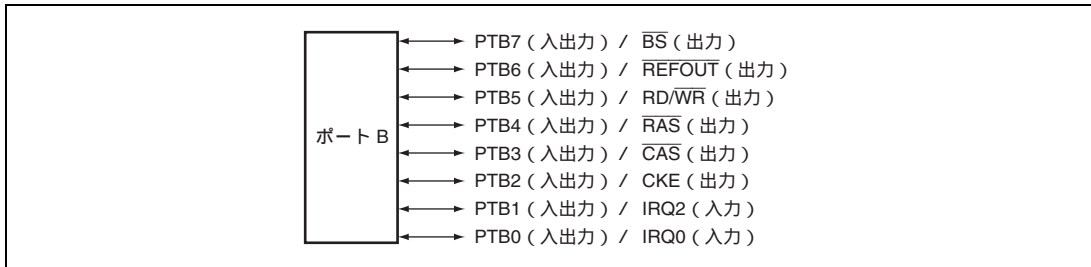


図 23.2 ポート B

23.2.1 レジスタの説明

ポート B には以下のレジスタがあります。このレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- ポート B データレジスタ (PBDR)

23.2.2 ポート B データレジスタ (PBDR)

PBDR は、読み出し / 書き込み可能な 8 ビットのレジスタで、端子 PTB7 ~ PTB0 のデータを格納します。PB7DT ~ PB0DT ビットは PTB7 ~ PTB0 端子に対応しています。端子機能が汎用出力ポートの場合には、ポートを読み出すと、対応する PBDR ビットの値が直接読み出されます。端子機能が汎用入力ポートの場合には、ポートを読み出すと、対応する端子レベルが読み出されます。

ビット	ビット名	初期値	R/W	説明
7	PB7DT	0	R/W	表 23.2 に PBDR の機能を示します。
6	PB6DT	0	R/W	
5	PB5DT	0	R/W	
4	PB4DT	0	R/W	
3	PB3DT	0	R/W	
2	PB2DT	0	R/W	
1	PB1DT	0	R/W	
0	PB0DT	0	R/W	

23. I/O ポート

表 23.2 ポート B データレジスタ (PBDR) の読み出し / 書き込み動作

PBCR ^{*1} の状態		端子状態	読み出しの場合	書き込みの場合
PBnMD1 ^{*2}	PBnMD0 ^{*2}			
0	0	その他の機能	PBDR の値	PBDR に書き込めるが、端子の状態に影響しない
	1	出力	PBDR の値	書き込み値が端子から出力される
1	0	入力 (プルアップ MOS オン)	端子の状態	PBDR に書き込めるが、端子の状態に影響しない
	1	入力 (プルアップ MOS オフ)	端子の状態	PBDR に書き込めるが、端子の状態に影響しない

【注】 *1 ポート B コントロールレジスタ (PBCR) については、「第 24 章 ピンファンクションコントローラ (PFC)」を参照してください。

*2 n = 0 ~ 7

23.3 ポート D

ポート D は図 23.3 に示すような端子構成を持つ 6 ビットの入出力ポートです。各端子には入力プルアップ MOS があり、PFC のポート D コントロールレジスタ (PDCR) で制御します。

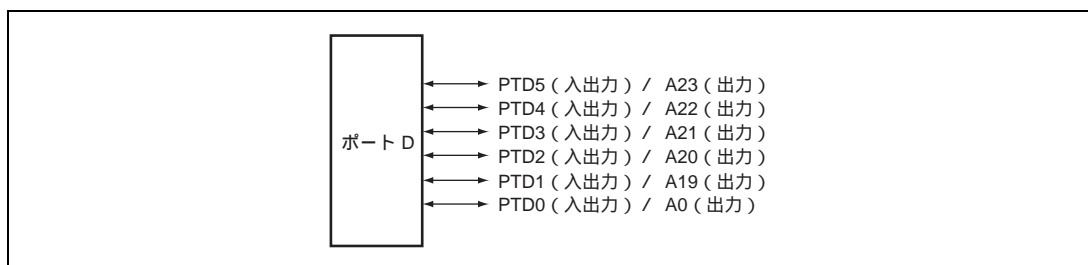


図 23.3 ポート D

23.3.1 レジスタの説明

ポート D には以下のレジスタがあります。このレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- ポート D データレジスタ (PDDR)

23.3.2 ポート D データレジスタ (PDDR)

PDDR は、読み出し / 書き込み可能な 6 ビットのレジスタで、端子 PTD5 ~ PTD0 のデータを格納します。PD5DT ~ PD0DT ビットは PTD5 ~ PTD0 端子に対応しています。端子機能が汎用出力ポートの場合には、ポートを読み出すと、対応する PDDR ビットの値が直接読み出されます。端子機能が汎用入力ポートの場合には、ポートを読み出すと、対応する端子レベルが読み出されます。

ビット	ビット名	初期値	R/W	説明
7, 6		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 としてください。 1 を書いた場合の動作の保証はできません。
5	PD5DT	0	R/W	表 23.3 に PDDR の機能を示します。
4	PD4DT	0	R/W	
3	PD3DT	0	R/W	
2	PD2DT	0	R/W	
1	PD1DT	0	R/W	
0	PD0DT	0	R/W	

23. I/O ポート

表 23.3 ポート D データレジスタ (PDDR) の読み出し / 書き込み動作

PDCR の状態* ¹		端子状態	読み出しの場合	書き込みの場合
PDnMD1* ²	PDnMD0* ²			
0	0	その他の機能	PDDR の値	PDDR に書き込めるが、端子の状態に影響しない
	1	出力	PDDR の値	書き込み値が端子から出力される
1	0	入力 (プルアップ MOS オン)	端子の状態	PDDR に書き込めるが、端子の状態に影響しない
	1	入力 (プルアップ MOS オフ)	端子の状態	PDDR に書き込めるが、端子の状態に影響しない

【注】 *1 ポート D コントロールレジスタ (PDCR) については、「第 24 章 ビンファンクションコントローラ (PFC)」を参照してください。

*2 n = 0 ~ 5

23.4 ポート E

ポート E は図 23.4 に示すような端子構成を持つ 7 ビットの入出力ポートです。各端子には入力プルアップ MOS があり、PFC のポート E コントロールレジスタ (PECR) で制御します。

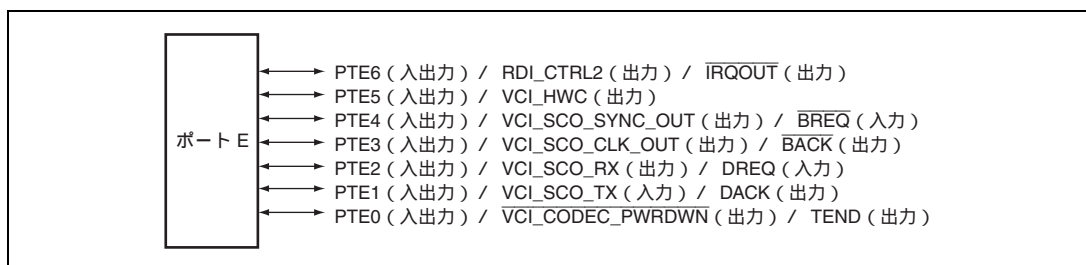


図 23.4 ポート E

23.4.1 レジスタの説明

ポート E には以下のレジスタがあります。このレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- ポート E データレジスタ (PEDR)

23.4.2 ポート E データレジスタ (PEDR)

PEDR は、読み出し / 書き込み可能な 7 ビットのレジスタで、端子 PTE6 ~ PTE0 のデータを格納します。PE6DT ~ PE0DT ビットは PTE6 ~ PTE0 端子に対応しています。端子機能が汎用出力ポートの場合には、ポートを読み出すと、対応する PEDR ビットの値が直接読み出されます。端子機能が汎用入力ポートの場合には、ポートを読み出すと、対応する端子レベルが読み出されます。

ビット	ビット名	初期値	R/W	説明
7		0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 としてください。 1 を書いた場合の動作の保証はできません。
6	PE6DT	0	R/W	表 23.4 に PEDR の機能を示します。
5	PE5DT	0	R/W	
4	PE4DT	0	R/W	
3	PE3DT	0	R/W	
2	PE2DT	0	R/W	
1	PE1DT	0	R/W	
0	PE0DT	0	R/W	

23. I/O ポート

表 23.4 ポート E データレジスタ (PEDR) の読み出し / 書き込み動作

PECR の状態*1		端子状態	読み出しの場合	書き込みの場合
PEnMD1*2	PEnMD0*2			
0	0	その他の機能	PEDR の値	PEDR に書き込めるが、端子の状態に影響しない
	1	出力	PEDR の値	書き込み値が端子から出力される
1	0	入力 (プルアップ MOS オン)	端子の状態	PEDR に書き込めるが、端子の状態に影響しない
	1	入力 (プルアップ MOS オフ)	端子の状態	PEDR に書き込めるが、端子の状態に影響しない

【注】 *1 ポート E コントロールレジスタ (PECR) については、「第 24 章 ビンファンクションコントローラ (PFC) 」を参照してください。

*2 n = 0 ~ 6

23.5 ポート G

ポート G は図 23.5 に示すような端子構成を持つ 5 ビットの入出力ポートです。各端子には入力プルアップ MOS があり、PFC のポート G コントロールレジスタ (PGCR) で制御します。

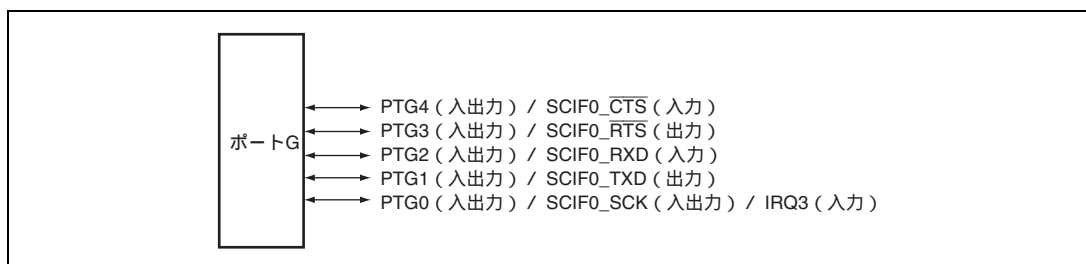


図 23.5 ポート G

23.5.1 レジスタの説明

ポート G には以下のレジスタがあります。このレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- ポート G データレジスタ (PGDR)

23.5.2 ポート G データレジスタ (PGDR)

PGDR は、読み出し / 書き込み可能な 5 ビットのレジスタで、端子 PTG4 ~ PTG0 のデータを格納します。PG4DT ~ PG0DT ビットは PTG4 ~ PTG0 端子に対応しています。端子機能が汎用出力ポートの場合には、ポートを読み出すと、対応する PGDR ビットの値が直接読み出されます。端子機能が汎用入力ポートの場合には、ポートを読み出すと、対応する端子レベルが読み出されます。

FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF) のシリアルコントロールレジスタ₀ (SCSCR₀) の TE ビットまたは RE ビットを 1 にセットすると PGCR レジスタの設定に関係なく、SCIF₀_TXD または SCIF₀_RXD 端子が優先的に選択されます。

23. I/O ポート

ビット	ビット名	初期値	R/W	説明
7~5		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 としてください。 1 を書いた場合の動作の保証はできません。
4	PG4DT	0	R/W	表 23.5 に PGDR の機能を示します。
3	PG3DT	0	R/W	
2	PG2DT	0	R/W	
1	PG1DT	0	R/W	
0	PG0DT	0	R/W	

表 23.5 ポート G データレジスタ (PGDR) の読み出し / 書き込み動作

PGCR の状態*1		端子状態	読み出しの場合	書き込みの場合
PGnMD1*2	PGnMD0*2			
0	0	その他の機能	PGDR の値	PGDR に書き込めるが、端子の状態に影響しない
	1	出力	PGDR の値	書き込み値が端子から出力される
1	0	入力 (プルアップ MOS オン)*3	端子の状態	PGDR に書き込めるが、端子の状態に影響しない
	1	入力 (プルアップ MOS オフ)*3	端子の状態	PGDR に書き込めるが、端子の状態に影響しない

【注】 *1 ポート G コントロールレジスタ (PGCR) については、「第 24 章 ビンファンクションコントローラ (PFC)」を参照してください。

*2 n = 0 ~ 4

*3 n = 1 のときは出力ハイインピーダンス状態になります。

23.6 ポート H

ポート H は図 23.6 に示すような端子構成を持つ 5 ビットの入出力ポートです。各端子には入力プルアップ MOS があり、PFC のポート H コントロールレジスタ (PHCR) で制御します。

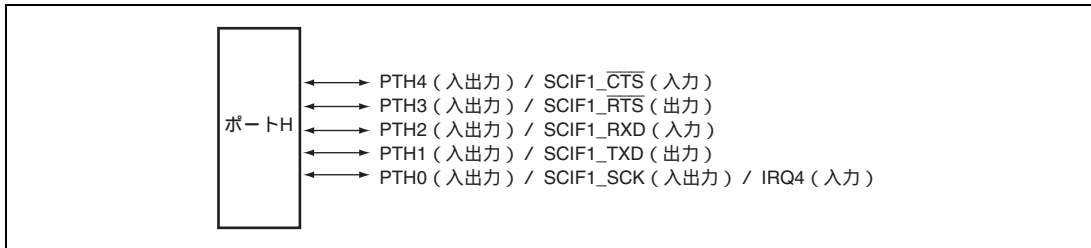


図 23.6 ポート H

23.6.1 レジスタの説明

ポート H には以下のレジスタがあります。このレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- ポート H データレジスタ (PHDR)

23.6.2 ポート H データレジスタ (PHDR)

PHDR は、読み出し / 書き込み可能な 5 ビットのレジスタで、端子 PTH4 ~ PTH0 のデータを格納します。PH4DT ~ PH0DT ビットは PTH4 ~ PTH0 端子に対応しています。端子機能が汎用出力ポートの場合には、ポートを読み出すと、対応する PHDR ビットの値が直接読み出されます。端子機能が汎用入力ポートの場合には、ポートを読み出すと、対応する端子レベルが読み出されます。

FIFO 内蔵シリアルコミュニケーションインタフェース (SCIF) のシリアルコントロールレジスタ₁ (SCSCR₁) の TE ビットまたは RE ビットを 1 にセットすると、PHCR レジスタの設定に関係なく、SCIF1_TXD または SCIF1_RXD 端子機能が優先的に選択されます。

23. I/O ポート

ビット	ビット名	初期値	R/W	説明
7~5		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 としてください。 1 を書いた場合の動作の保証はできません。
4	PH4DT	0	R/W	表 23.6 に PHDR の機能を示します。
3	PH3DT	0	R/W	
2	PH2DT	0	R/W	
1	PH1DT	0	R/W	
0	PH0DT	0	R/W	

表 23.6 ポート H データレジスタ (PHDR) の読み出し / 書き込み動作

PHCR の状態* ¹		端子状態	読み出しの場合	書き込みの場合
PHnMD1* ²	PHnMD0* ²			
0	0	その他の機能	PHDR の値	PHDR に書き込めるが、端子の状態に影響しない
	1	出力	PHDR の値	書き込み値が端子から出力される
1	0	入力 (プルアップ MOS オン)* ³	端子の状態	PHDR に書き込めるが、端子の状態に影響しない
	1	入力 (プルアップ MOS オフ)* ³	端子の状態	PHDR に書き込めるが、端子の状態に影響しない

【注】 *1 ポート H コントロールレジスタ (PHCR) については、「第 24 章 ビンファンクションコントローラ (PFC)」を参照してください。

*2 n = 0 ~ 4

*3 n = 1 のときは出力ハイインピーダンス状態になります。

24. ピンファンクションコントローラ (PFC)

24.1 概要

本 LSI では、いくつかの端子に I/O ポートの機能がマルチプレクスされています (I/O ポートの詳細については「第 23 章 I/O ポート」を参照してください)。ピンファンクションコントローラ (PFC) は、I/O ポートとマルチプレクスされている端子の機能とその入出力の方向を選ぶためのレジスタで構成されています。端子機能と入出力の方向は、LSI の動作モードに関係なく端子ごとに個別に選択することができます。表 24.1 に、本 LSI のマルチプレクス端子を示します。表 24.1 において、ハッチングの機能がリセット直後から使用できる端子機能です。

表 24.1 マルチプレクス一覧表

ポート	機能 1 (関連モジュール)	機能 2 (関連モジュール)	機能 3 (関連モジュール)
A	PTA6 入出力 (ポート)	SIOF_SS2 出力 (SIOF)	-
A	PTA5 入出力 (ポート)	SIOF_SS1 出力 (SIOF)	-
A	PTA4 入出力 (ポート)	SIOF_SCK(SCK)入出力 (SIOF)	-
A	PTA3 入出力 (ポート)	SIOF_SYNC(SIOF_SS0)入出力 (SIOF)	-
A	PTA2 入出力 (ポート)	SIOF_RXD(MISO)入力 (SIOF)	-
A	PTA1 入出力 (ポート)	SIOF_TXD(MOSI)出力 (SIOF)	-
A	PTA0 入出力 (ポート)	SIOF_MCLK 入力 (SIOF)	-
B	PTB7 入出力 (ポート)	BS 出力 (BSC)	-
B	PTB6 入出力 (ポート)	REFOUT 出力 (BSC)	-
B	PTB5 入出力 (ポート)	RD/WR 出力 (BSC)	-
B	PTB4 入出力 (ポート)	RAS 出力 (BSC)	-
B	PTB3 入出力 (ポート)	CAS 出力 (BSC)	-
B	PTB2 入出力 (ポート)	CKE 出力 (BSC)	-
B	PTB1 入出力 (ポート)	IRQ2 入力 (INTC)	-
B	PTB0 入出力 (ポート)	IRQ0 入力 (INTC)	-
D	PTD5 入出力 (ポート)	A23 出力 (BSC)	-
D	PTD4 入出力 (ポート)	A22 出力 (BSC)	-
D	PTD3 入出力 (ポート)	A21 出力 (BSC)	-
D	PTD2 入出力 (ポート)	A20 出力 (BSC)	-
D	PTD1 入出力 (ポート)	A19 出力 (BSC)	-
D	PTD0 入出力 (ポート)	A0 出力 (BSC)	-

24. ピンファンクションコントローラ (PFC)

ポート	機能 1 (関連モジュール)	機能 2 (関連モジュール)	機能 3 (関連モジュール)
E	PTE6 入出力 (ポート)	RDI_CTRL2 出力 (BT)	IRQOUT 出力 (INTC)
E	PTE5 入出力 (ポート)	VCI_HWC 出力 (BT)	-
E	PTE4 入出力 (ポート)	VCI_SCO_SYNC_OUT 出力 (BT)	BREQ 入力 (BSC)
E	PTE3 入出力 (ポート)	VCI_SCO_CLK_OUT 出力 (BT)	BACK 出力 (BSC)
E	PTE2 入出力 (ポート)	VCI_SCO_RX 出力 (BT)	DREQ 入力 (DMAC)
E	PTE1 入出力 (ポート)	VCI_SCO_TX 入力 (BT)	DACK 出力 (DMAC)
E	PTE0 入出力 (ポート)	VCI_CODEC_PWRDWN 出力 (BT)	TEND 出力 (DMAC)
G	PTG4 入出力 (ポート)	SCIF0_CTS 入力 (SCIF0)	-
G	PTG3 入出力 (ポート)	SCIF0_RTS 出力 (SCIF0)	-
G	PTG2 入出力 (ポート)	SCIF0_RXD 入力 (SCIF0)	-
G	PTG1 入出力 (ポート)	SCIF0_TXD 出力 (SCIF0)	-
G	PTG0 入出力 (ポート)	SCIF0_SCK 入出力 (SCIF0)	IRQ3 入力 (INTC)
H	PTH4 入出力 (ポート)	SCIF1_CTS 入力 (SCIF1)	-
H	PTH3 入出力 (ポート)	SCIF1_RTS 出力 (SCIF1)	-
H	PTH2 入出力 (ポート)	SCIF1_RXD 入力 (SCIF1)	-
H	PTH1 入出力 (ポート)	SCIF1_TXD 出力 (SCIF1)	-
H	PTH0 入出力 (ポート)	SCIF1_SCK 入出力 (SCIF1)	IRQ4 入力 (INTC)

24.2 レジスタの説明

PFC のレジスタを以下に示します。これらのレジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。

- ポートAコントロールレジスタ (PACR)
- ポートBコントロールレジスタ (PBCR)
- ポートDコントロールレジスタ (PDCR)
- ポートEコントロールレジスタ (PECR)
- ポートGコントロールレジスタ (PGCR)
- ポートHコントロールレジスタ (PHCR)
- ピンセレクトレジスタA (PSELA)
- I/OバッファHi-Z制御レジスタA (HIZCRA)
- ノイズキャンセラ制御レジスタ (NCCR)

【注】 PFC モジュールの各レジスタは、マニュアルリセットでは初期化されず、値を保持します。

24.2.1 ポート A コントロールレジスタ (PACR)

PACR は、読み出し / 書き込み可能な 16 ビットのレジスタで、端子機能と入力プルアップ MOS 制御を選択します。

ビット	ビット名	初期値*	R/W	説明
15、14	-	すべて 0	R	リザーブビット 読み出すと 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
13 12	PA6MD1 PA6MD0	1 0	R/W R/W	PTA6 モード 00 : その他の機能 (表 24.1 参照) 01 : ポート出力 10 : ポート入力 (プルアップ MOS : オン) 11 : ポート入力 (プルアップ MOS : オフ)
11 10	PA5MD1 PA5MD0	1 0	R/W R/W	PTA5 モード 00 : その他の機能 (表 24.1 参照) 01 : ポート出力 10 : ポート入力 (プルアップ MOS : オン) 11 : ポート入力 (プルアップ MOS : オフ)

24. ピンファンクションコントローラ (PFC)

ビット	ビット名	初期値*	R/W	説明
9	PA4MD1	1	R/W	PTA4 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
8	PA4MD0	0	R/W	
7	PA3MD1	1	R/W	PTA3 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
6	PA3MD0	0	R/W	
5	PA2MD1	1	R/W	PTA2 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
4	PA2MD0	0	R/W	
3	PA1MD1	1	R/W	PTA1 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
2	PA1MD0	0	R/W	
1	PA0MD1	1	R/W	PTA0 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
0	PA0MD0	0	R/W	

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

24.2.2 ポート B コントロールレジスタ (PBCR)

PBCR は、読み出し / 書き込み可能な 16 ビットのレジスタで、端子機能と入力プルアップ MOS 制御を選択します。

ビット	ビット名	初期値*	R/W	説明
15	PB7MD1	0	R/W	PTB7 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
14	PB7MD0	0	R/W	

24. ピンファンクションコントローラ (PFC)

ビット	ビット名	初期値*	R/W	説明
13	PB6MD1	0	R/W	PTB6 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
12	PB6MD0	0	R/W	
11	PB5MD1	0	R/W	PTB5 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
10	PB5MD0	0	R/W	
9	PB4MD1	0	R/W	PTB4 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
8	PB4MD0	0	R/W	
7	PB3MD1	0	R/W	PTB3 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
6	PB3MD0	0	R/W	
5	PB2MD1	0	R/W	PTB2 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
4	PB2MD0	0	R/W	
3	PB1MD1	1	R/W	PTB1 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
2	PB1MD0	0	R/W	
1	PB0MD1	1	R/W	PTB0 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
0	PB0MD0	0	R/W	

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

24. ピンファンクションコントローラ (PFC)

24.2.3 ポート D コントロールレジスタ (PDCR)

PDCR は、読み出し / 書き込み可能な 16 ビットのレジスタで、端子機能と入力プルアップ MOS 制御を選択します。

ビット	ビット名	初期値*	R/W	説明
15~12	-	すべて 0	R	リザーブビット 読み出すと 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
11 10	PD5MD1 PD5MD0	0 0	R/W R/W	PTD5 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
9 8	PD4MD1 PD4MD0	0 0	R/W R/W	PTD4 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
7 6	PD3MD1 PD3MD0	0 0	R/W R/W	PTD3 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
5 4	PD2MD1 PD2MD0	0 0	R/W R/W	PTD2 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
3 2	PD1MD1 PD1MD0	0 0	R/W R/W	PTD1 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
1 0	PD0MD1 PD0MD0	0 0	R/W R/W	PTD0 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

24.2.4 ポート E コントロールレジスタ (PECR)

PECR は、読み出し / 書き込み可能な 16 ビットのレジスタで、端子機能と入力プルアップ MOS 制御を選択します。

ビット	ビット名	初期値*	R/W	説明
15、14	-	すべて 0	R	リザーブビット 読み出すと 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
13 12	PE6MD1 PE6MD0	1 0	R/W R/W	PTE6 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
11 10	PE5MD1 PE5MD0	1 0	R/W R/W	PTE5 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
9 8	PE4MD1 PE4MD0	0 0	R/W R/W	PTE4 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
7 6	PE3MD1 PE3MD0	0 0	R/W R/W	PTE3 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
5 4	PE2MD1 PE2MD0	1 0	R/W R/W	PTE2 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
3 2	PE1MD1 PE1MD0	1 0	R/W R/W	PTE1 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)

24. ピンファンクションコントローラ (PFC)

ビット	ビット名	初期値*	R/W	説明
1	PE0MD1	1	R/W	PTE0 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
0	PE0MD0	0	R/W	

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

24.2.5 ポート G コントロールレジスタ (PGCR)

PGCR は、読み出し / 書き込み可能な 16 ビットのレジスタで、端子機能と入力プルアップ MOS 制御を選択します。

ビット	ビット名	初期値*	R/W	説明
15~10	-	すべて 0	R	リザーブビット 読み出すと 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
9	PG4MD1	1	R/W	PTG4 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
8	PG4MD0	0	R/W	
7	PG3MD1	1	R/W	PTG3 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
6	PG3MD0	0	R/W	
5	PG2MD1	1	R/W	PTG2 モード SCSCR_0 の RE ビット=0 の場合 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ) SCSCR_0 の RE ビット=1 の場合 PG2MD1、PG2MD0 にかかわらず、その他の機能 (SCIF0_RXD) となります。
4	PG2MD0	0	R/W	

24. ピンファンクションコントローラ (PFC)

ビット	ビット名	初期値*	R/W	説明
3 2	PG1MD1 PG1MD0	1 0	R/W R/W	PTG1 モード SCSCR_0 の TE ビット=0 の場合 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: 出力ハイインピーダンス状態 11: 出力ハイインピーダンス状態 SCSCR_0 の TE ビット=1 の場合 PG1MD1、PG1MD0 にかかわらず、その他の機能 (SCIF0_TXD) となります。
1 0	PG0MD1 PG0MD0	1 0	R/W R/W	PTG0 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

24.2.6 ポート H コントロールレジスタ (PHCR)

PHCR は、読み出し / 書き込み可能な 16 ビットのレジスタで、端子機能と入力プルアップ MOS 制御を選択します。

ビット	ビット名	初期値*	R/W	説明
15~10	-	すべて 0	R	リザーブビット 読み出すと 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
9 8	PH4MD1 PH4MD0	1 0	R/W R/W	PTH4 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)
7 6	PH3MD1 PH3MD0	1 0	R/W R/W	PTH3 モード 00: その他の機能 (表 24.1 参照) 01: ポート出力 10: ポート入力 (プルアップ MOS: オン) 11: ポート入力 (プルアップ MOS: オフ)

24. ピンファンクションコントローラ (PFC)

ビット	ビット名	初期値*	R/W	説明
5 4	PH2MD1 PH2MD0	1 0	R/W R/W	PTH2 モード SCSCR_1 の RE ビット=0 の場合 00：その他の機能（表 24.1 参照） 01：ポート出力 10：ポート入力（プルアップ MOS：オン） 11：ポート入力（プルアップ MOS：オフ） SCSCR_1 の RE ビット=1 の場合 PH2MD1、PH2MD0 にかかわらず、その他の機能（SCIF1_RXD）となります。
3 2	PH1MD1 PH1MD0	1 0	R/W R/W	PTH1 モード SCSCR_1 の TE ビット=0 の場合 00：その他の機能（表 24.1 参照） 01：ポート出力 10：出力ハイインピーダンス状態 11：出力ハイインピーダンス状態 SCSCR_1 の TE ビット=1 の場合 PH1MD1、PH1MD0 にかかわらず、その他の機能（SCIF1_TXD）となります。
1 0	PH0MD1 PH0MD0	1 0	R/W R/W	PTH0 モード 00：その他の機能（表 24.1 参照） 01：ポート出力 10：ポート入力（プルアップ MOS：オン） 11：ポート入力（プルアップ MOS：オフ）

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

24.2.7 ピンセレクトレジスタ A (PSELA)

PSELA は、2 つの目的を持っている読み出し / 書き込み可能な 16 ビットのレジスタです。

ビット 0 (RTCSEL1) はブルートゥースインタフェース (BT) で使われる低周波数クロックのソース周波数を選択します。32.000kHz を選択した場合は、そのまま BT で使われますが、32.768kHz を選択した場合は、擬似的に 32.000kHz のクロックに周波数補正を行う回路が働きます。詳細について、「第 21 章 ブルートゥースインタフェース (BT)」を参照してください。

ビット 1~8 は「その他の機能」を 2 つ以上マルチプレクスしている端子の機能を選択します。これらの端子において、一方の「その他の機能」を使用する場合は、PSELA の該当ビットを設定した後にポートコントロールレジスタを「その他の機能」に設定してください。

設定例：PTE6/RDI_CTRL2/IRQOUT 端子において、IRQOUT 機能を使用する場合

1. PSELA.PSA8 ビットに 1 を書き込み。
2. ポート E コントロールレジスタ PECR の (PE6MD1、PE6MD0) ビットを (0、0) (その他の機能) に設定。

24. ピンファンクションコントローラ (PFC)

ビット	ビット名	初期値*	R/W	説明
15-9		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も必ず0にしてください。 1を書いた場合の動作の保証はできません。
8	PSA8	0	R/W	RDI_CTRL2/IRQOUTの選択 0: RDI_CTRL2を選択 1: IRQOUTを選択
7	PSA7	1	R/W	VCI_SCO_SYNC_OUT/BREQの選択 0: VCI_SCO_SYNC_OUTを選択 1: BREQを選択
6	PSA6	1	R/W	VCI_SCO_CLK_OUT/BACKの選択 0: VCI_SCO_CLK_OUTを選択 1: BACKを選択
5	PSA5	0	R/W	VCI_SCO_RX/DREQの選択 0: VCI_SCO_RXを選択 1: DREQを選択
4	PSA4	0	R/W	VCI_SCO_TX/DACKの選択 0: VCI_SCO_TXを選択 1: DACKを選択
3	PSA3	0	R/W	VCI_CODEC_PWRDWN/TENDの選択 0: VCI_CODEC_PWRDWNを選択 1: TENDを選択
2	PSA2	0	R/W	SCIF0_SCK/IRQ3の選択 0: SCIF0_SCKを選択 1: IRQ3を選択
1	PSA1	0	R/W	SCIF1_SCK/IRQ4の選択 0: SCIF1_SCKを選択 1: IRQ4を選択
0	RTCSEL 1	0	R/W	低周波数クロックの周波数選択 0: 32.768kHzを選択 1: 32.000kHzを選択

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

24.2.8 I/O バッファ Hi-Z 制御レジスタ A (HIZCRA)

HIZCRA は読み出し / 書き込み可能な 16 ビットのレジスタで、端子の Hi-Z 制御およびプルアップ制御を機能単位ごとに行います*。

【注】 * ポート n コントロールレジスタ (PnCR) でポート入力 (プルアップ MOS : オン) が選択されている場合、HIZCRA レジスタで出力ハイインピーダンスに設定されていても、プルアップ MOS はオン状態を維持します。

24. ピンファンクションコントローラ (PFC)

ビット	ビット名	初期値*1	R/W	説明
15~9		すべて0	R	リザーブビット 読み出すと常に0が読み出されます。書き込む値も必ず0にしてください。 1を書いた場合の動作の保証はできません。
8	HIZA8	0	R/W	SIOF 関連信号 (SIOF_RXD(MISO)/ SIOF_TXD(MOSI)/ SIOF_SYNC(SIOF_SS0)/SIOF_SS1/ SIOF_SS2) がマルチプレクスされている端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる
7	HIZA7	0	R/W	SIOF 関連信号 (SIOF_MCLK/ SIOF_SCK(SCK)) がマルチプレクスされている端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる
6	HIZA6	0	R/W	SCIF0 関連信号 (SCIF0_RXD/SCIF0_TXD/SCIF0_RTS/SCIF0_CTS) がマルチプレクスされている端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる
5	HIZA5	0	R/W	SCIF0 関連信号 (SCIF0_SCK) がマルチプレクスされている端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる
4	HIZA4	0	R/W	SCIF1 関連信号 (SCIF1_RXD/SCIF1_TXD/SCIF1_RTS/SCIF1_CTS) がマルチプレクスされている端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる
3	HIZA3	0	R/W	SCIF1 関連信号 (SCIF1_SCK) がマルチプレクスされている端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる
2	HIZA2	0	R/W	RD/WR / RAS / CAS / CKE がマルチプレクスされている端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる
1	HIZA1	0	R/W	IRQ2 端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる*
0	HIZA0	0	R/W	IRQ0 端子のハイインピーダンス制御を行います。 0 : I/O バッファは通常動作 1 : I/O バッファは入力固定、出力ハイインピーダンスとなる**

【注】 *1 マニュアルリセットでは初期化されず、レジスタの値を保持します。

*2 本設定では、シュミット入力が入ります。

24.2.9 ノイズキャンセラ制御レジスタ (NCCR)

NCCR は、読み出し / 書き込み可能な 16 ビットのレジスタで、ノイズキャンセラのオン / オフ制御を行います。

ビット	ビット名	初期値*	R/W	説明
15~1		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も必ず 0 にしてください。 1 を書いた場合の動作の保証はできません。
0	NCCR0	0	R/W	$\overline{\text{TRST}}$ 、NMI、IRQ のノイズキャンセラ制御を行います。 0 : ノイズキャンセラオフ 1 : ノイズキャンセラオン

【注】 * マニュアルリセットでは初期化されず、レジスタの値を保持します。

25. ユーザブレイクコントローラ (UBC)

ユーザブレイクコントローラ (UBC) は、プログラムデバッグを容易にする機能を提供します。この機能を使用することにより、セルフモニタデバッグを容易に作成でき、インサーキットエミュレータを使用しなくても、本 LSI 単体で手軽にプログラムをデバッグできます。UBC に設定できるブレイク条件には、命令フェッチまたはデータの読み出し書き込み、データのサイズ、データの内容、アドレスの値、および命令フェッチのときの停止タイミングがあります。

25.1 特長

1. 次のようなブレイク比較条件を設定できます

ブレイクチャンネル数 : 2チャンネル (チャンネルAとB)

ユーザブレイクは、チャンネルA、B独立に、または連続した (シーケンシャル) 一つの条件として設定することができます (シーケンシャルブレイク設定 : チャンネルAのブレイク条件が一致した後チャンネルBのブレイク条件の一致が発生し、しかも両者が同じバスサイクルで発生しないとき)。

• アドレス

アドレス32ビットの比較はビットごとにマスク可能で、ユーザは下位12ビット (4kページ)、下位10ビット (1kページ)、あるいは任意の大きさのページ等でアドレスをマスク可能です。

4本のアドレスバス (Lバスアドレス (LAB)、Iバスアドレス (IAB)、Xメモリアドレスバス (XAB) およびYメモリアドレスバス (YAB)) の1つを選択できます。

• データ

チャンネルBのみ、32ビットマスク可能。

4本のデータバス (Lバスデータ (LDB)、Iバスデータ (IDB)、Xメモリデータバス (XDB)、およびYメモリデータバス (YDB)) の1つを選択可能です。

• バスサイクル

命令フェッチまたはデータアクセス

• 読み出しまたは書き込み

• オペランドサイズ

バイト、ワード、およびロングワードをサポート。

2. ユーザ指定のユーザブレイク条件例外処理ルーチンを実行可能。

3. 命令フェッチサイクルにおいて、ブレイクを命令の実行の前に設定するか、後に設定するかを指定可能。

25. ユーザブレイクコントローラ (UBC)

4. ブレイク条件 (チャンネルBに対してのみ) として、最大 $2^{12} - 1$ 回まで繰り返し回数を指定可能。
5. 8組の分岐元 / 分岐先バッファをサポート。

UBC のブロック図を図 25.1 に示します。

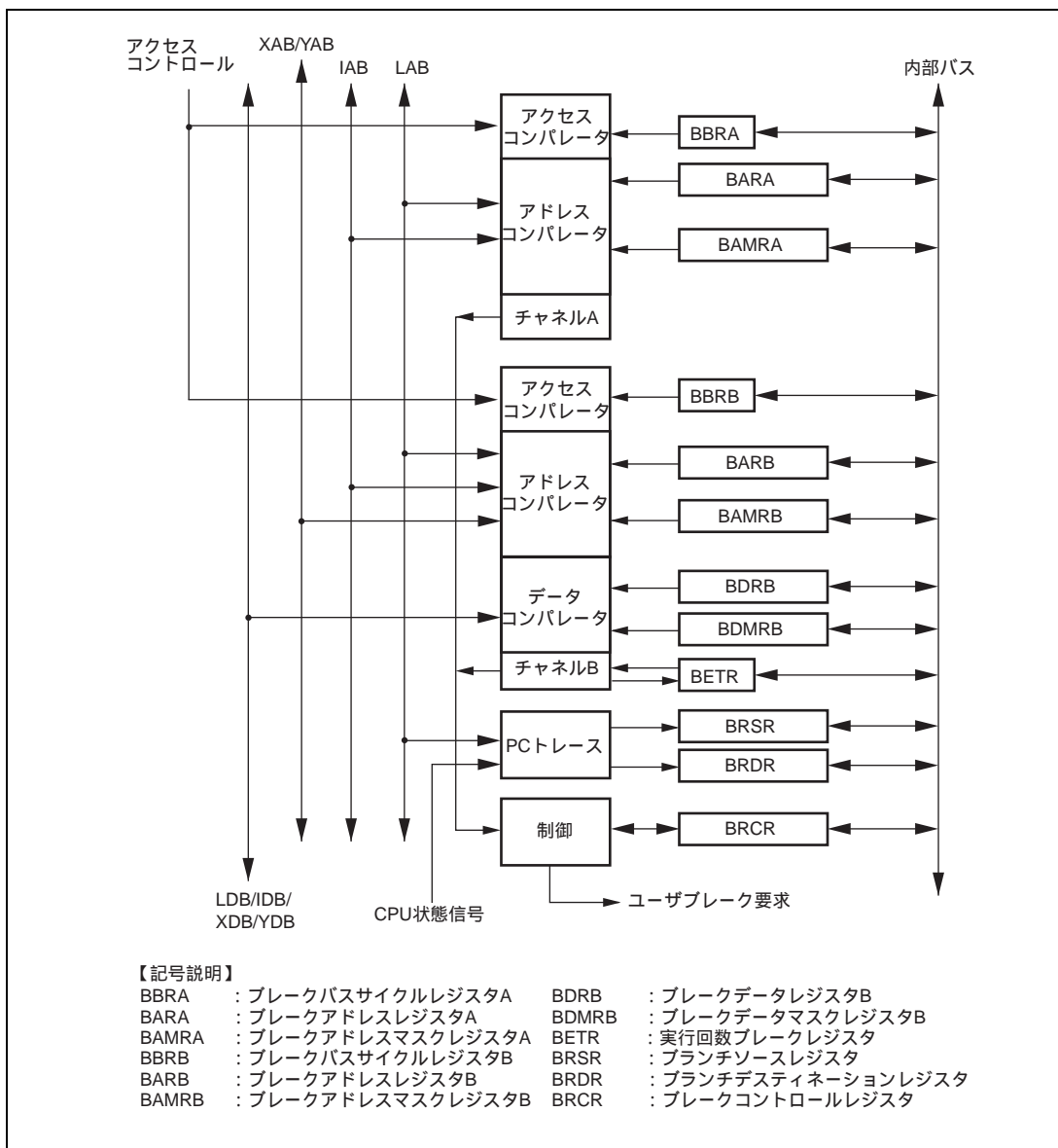


図 25.1 UBC のブロック図

25.2 レジスタの説明

UBC には以下のレジスタがあります。これらのレジスタのアドレスおよびアクセスサイズについては「第 27 章 レジスタ一覧」を参照してください。

- ブレークアドレスレジスタA (BARA)
- ブレークアドレスマスクレジスタA (BAMRA)
- ブレークバスサイクルレジスタA (BBRA)
- ブレークアドレスレジスタB (BARB)
- ブレークアドレスマスクレジスタB (BAMRB)
- ブレークバスサイクルレジスタB (BBRB)
- ブレークデータレジスタB (BDRB)
- ブレークデータマスクレジスタB (BDMRB)
- ブレークコントロールレジスタ (BRCR)
- 実行回数ブレークレジスタ (BETR)
- ブランチソースレジスタ (BRSR)
- ブランチデスティネーションレジスタ (BRDR)

25.2.1 ブレークアドレスレジスタ A (BARA)

BARA は、32 ビットの読み出しまたは書き込み可能なレジスタです。BARA は、チャンネル A のブレーク条件とするアドレスを指定します。

ビット	ビット名	初期値	R/W	説 明
31~0	BAA31~ BAA0	すべて 0	R/W	ブレークアドレス A チャンネル A のブレーク条件を指定する LAB または IAB のアドレスを格納します。

25. ユーザブレイクコントローラ (UBC)

25.2.2 ブ레이크アドレスマスクレジスタ A (BAMRA)

BAMRA は 32 ビットの読み出しまたは書き込み可能なレジスタです。BAMRA は、BARA によって指定されるブレイクアドレスビットのうち、マスクするビットを指定します。

ビット	ビット名	初期値	R/W	説明
31~0	BAMA31 ~ BAMA0	すべて 0	R/W	ブレイクアドレスマスク A BARA (BAA31~BAA0) によって指定されるチャンネル A のブレイクアドレスビットのうち、マスクするビットを指定します。 0: ブレイクアドレスビット BAA _n は、ブレイク条件に含まれる 1: ブレイクアドレスビット BAA _n はマスクされ、ブレイク条件に含まれない 【注】 n=31~0

25.2.3 ブレイクバスサイクルレジスタ A (BBRA)

BBRA は、チャンネル A のブレイク条件として (1) L バスサイクルまたは I バスサイクル、(2) 命令フェッチまたはデータアクセス、(3) 読み出しまたは書き込み、および (4) オペランドサイズを指定する 16 ビットの読み出しまたは書き込み可能なレジスタです。

ビット	ビット名	初期値	R/W	説明
15~8		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
7 6	CDA1 CDA0	0 0	R/W R/W	L バスサイクル / I バスサイクルセレクト A チャンネル A ブレイク条件のバスサイクルとして L バスサイクルまたは I バスサイクルを選択します。 00: 条件比較を行わない 01: ブレイク条件は、L バスサイクル 10: ブレイク条件は、I バスサイクル 11: ブレイク条件は、L バスサイクル
5 4	IDA1 IDA0	0 0	R/W R/W	命令フェッチ / データアクセスセレクト A チャンネル A ブレイク条件のバスサイクルとして命令フェッチサイクルまたはデータアクセスサイクルを選択します。 00: 条件比較を行わない 01: ブレイク条件は、命令フェッチサイクル 10: ブレイク条件は、データアクセスサイクル 11: ブレイク条件は、命令フェッチサイクルまたはデータアクセスサイクル

ビット	ビット名	初期値	R/W	説明
3	RWA1	0	R/W	読み出し / 書き込みセレクト A チャンネル A ブ레이크条件のバスサイクルとして読み出しサイクルまたは書き込みサイクルを選択します。 00 : 条件比較を行わない 01 : ブ레이크条件は、読み出しサイクル 10 : ブ레이크条件は、書き込みサイクル 11 : ブ레이크条件は、読み出しサイクルまたは書き込みサイクル
2	RWA0	0	R/W	
1	SZA1	0	R/W	オペランドサイズセレクト A チャンネル A ブ레이크条件のバスサイクルのオペランドサイズを選択します。 00 : ブ레이크条件には、オペランドサイズを含まない 01 : ブ레이크条件は、バイトアクセス 10 : ブ레이크条件は、ワードアクセス 11 : ブ레이크条件は、ロングワードアクセス
0	SZA0	0	R/W	

25.2.4 ブ레이크アドレスレジスタ B (BARB)

BARB は、32 ビットの読み出しまたは書き込み可能なレジスタです。チャンネル B のブ레이크条件とするアドレスを指定します。ブ레이크条件 B の対象とするアドレスバスは 4 種類あり、ブ레이크バスサイクルレジスタ B (BBRB) の制御ビット CDB1、CDB0、XYE および XY5 により選択します。

ビット	ビット名	初期値	R/W	説明
31~0	BAB31~BAB0	すべて 0	R/W	ブ레이크アドレス B チャンネル B のブ레이크条件を指定するアドレスを指定します。 BBRB により I バスまたは L バスを選択した場合は、BAB31~BAB0 に IAB または LAB のアドレスを指定します。 BBRB により X メモリを選択した場合は、BAB31~BAB17 に XAB のビット 15~1 の値を設定します。このとき、BAB16~BAB0 の値は任意です。 BBRB により Y メモリを選択した場合は、BAB15~BAB1 に YAB のビット 15~1 の値を設定します。このとき、BAB31~BAB16 および BAB0 の値は任意です。

表 25.1 ブ레이크アドレスレジスタの指定

BBRB でのバス選択	BAB31~BAB17	BAB16	BAB15~BAB1	BAB0
L バス	LAB31~LAB0			
I バス	IAB31~IAB0			
X バス	XAB15~XAB1	don't care	don't care	don't care
Y バス	don't care	don't care	YAB15~YAB1	don't care

25. ユーザブレイクコントローラ (UBC)

25.2.5 ブ레이크アドレスマスクレジスタ B (BAMRB)

BAMRB は、32 ビットの読み出しまたは書き込み可能なレジスタです。BAMRB は、BARB で指定するブレイクアドレスビットのうち、マスクするビットを指定します。

ビット	ビット名	初期値	R/W	説明
31~0	BAMB31 ~ BAMB0	すべて 0	R/W	ブレイクアドレスマスク B BARB (BAB31~BAB0) によって指定されるチャンネル B のブレイクアドレスビットのうち、マスクするビットを指定します。 0: ブレイクアドレスビット BABn は、ブレイク条件に含まれる 1: ブレイクアドレスビット BABn はマスクされ、ブレイク条件に含まれない 【注】n=31~0

25.2.6 ブレイクデータレジスタ B (BDRB)

BDRB は、32 ビットの読み出しまたは書き込み可能なレジスタです。ブレイク条件 B の対象とするデータバスは 4 種類あり、ブレイクバスサイクルレジスタ B (BBRB) の制御ビット CDB1、CDB0、XYE および XYX により選択します。

ビット	ビット名	初期値	R/W	説明
31~0	BDB31~ BDB0	すべて 0	R/W	ブレイクデータビット B チャンネル B のブレイク条件を指定するデータを格納します。 BBRB により I バスを選択した場合は、BDB31~BDB0 に IDB のブレイクデータを指定します。 BBRB により L バスを選択した場合は、BDB31~BDB0 に LDB のブレイクデータを指定します。 BBRB により X メモリを選択した場合は、BDB31~BDB16 に XDB のビット 15~0 のブレイクデータを設定します。このとき、BDB15~BDB0 の値は任意です。 BBRB により Y メモリを選択した場合は、BDB15~BDB0 に YDB のビット 15~0 のブレイクデータを設定します。このとき、BDB31~BDB16 の値は任意です。

表 25.2 ブ레이크データレジスタの指定

BBRB でのバス選択	BDB31 ~ BDB16	BDB15 ~ BDB0
Lバス	LDB31 ~ LDB0	
Iバス	IDB31 ~ IDB0	
Xバス	XDB15 ~ XDB0	don't care
Yバス	don't care	YDB15 ~ YDB0

- 【注】
1. ブ레이크条件にデータバスの値を含める場合は、オペランドサイズを指定してください。
 2. ブ레이크条件としてバイトサイズを指定する場合は、BDRB におけるブ레이크データとして、ビット 15~8 とビット 7~0 に同一のバイトデータをセットしてください。
 3. MOV.S.W @-As,Ds、MOV.S.W @As,Ds、MOV.S.W @As+,Ds、および MOV.S.W @As+Ix,Ds 命令に対する L バス上のブ레이크条件として、データバスの値を含める場合は、ビット 31~16 にデータをセットしてください。

25.2.7 ブ레이크データマスクレジスタ B (BDMRB)

BDMRB は、32 ビットの読み出しまたは書き込み可能なレジスタです。BDMRB は、BDRB で指定するブ레이크データビットのうちマスクするビットを指定します。

ビット	ビット名	初期値	R/W	説 明
31~0	BDMB31 ~ BDMB0	すべて 0	R/W	ブ레이크データマスク B BDRB (BDB31 ~ BDB0) によって指定されるチャンネル B のブ레이크データビットのうちマスクするビットを指定します。 0: ブ레이크データビット BDBn は、ブ레이크条件に含まれる 1: ブ레이크データビット BDBn はマスクされ、ブ레이크条件に含まれない 【注】n=31~0

- 【注】
1. ブ레이크条件にデータバスの値を含める場合は、オペランドサイズを指定してください。
 2. ブ레이크条件としてバイトサイズを指定する場合は、BDMRB におけるブ레이크マスクデータとして、ビット 15~8 とビット 7~0 に同一のバイトデータをセットしてください。
 3. MOV.S.W @-As,Ds、MOV.S.W @As,Ds、MOV.S.W @As+,Ds、および MOV.S.W @As+Ix,Ds 命令に対する L バス上のブ레이크条件としてデータバスの値を含める場合は、ビット 31~16 にマスクデータをセットしてください。

25. ユーザブレイクコントローラ (UBC)

25.2.8 ブレークバスサイクルレジスタ B (BBRB)

BBRB は、チャンネル B のブレイク条件として (1) X バスまたは Y バス、(2) L バスサイクルまたは I バスサイクル、(3) 命令フェッチまたはデータアクセス、(4) 読み出しまたは書き込み、および (5) オペランドサイズを指定する 16 ビットの読み出しまたは書き込み可能なレジスタです。

ビット	ビット名	初期値	R/W	説明
15~10		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
9	XYE	0	R/W	X メモリまたは Y メモリバスをチャンネル B のブレイク条件として選択します。ただし、CDB1,CDB0 で L バスを選択した場合にのみ有効です。X および Y の選択は XYS ビットで行います。 0 : チャンネル B ブレイク条件用に L バスを選択 1 : チャンネル B ブレイク条件用に X/Y メモリバスを選択
8	XYS	0	R/W	X バスまたは Y バスをチャンネル B ブレイク条件のバスとして選択します。 0 : チャンネル B ブレイク条件用に X バスを選択 1 : チャンネル B ブレイク条件用に Y バスを選択
7	CDB1	0	R/W	L バスサイクル / I バスサイクルセレクト B チャンネル B ブレイク条件のバスサイクルとして L バスサイクルまたは I バスサイクルを選択します。 00 : 条件比較を行わない 01 : ブレイク条件は、L バスサイクル 10 : ブレイク条件は、I バスサイクル 11 : ブレイク条件は、L バスサイクル
6	CDB0	0	R/W	
5	IDB1	0	R/W	命令フェッチ / データアクセスセレクト B チャンネル B ブレイク条件のバスサイクルとして命令フェッチサイクルまたはデータアクセスサイクルを選択します。 00 : 条件比較を行わない 01 : ブレイク条件は、命令フェッチサイクル 10 : ブレイク条件は、データアクセスサイクル 11 : ブレイク条件は、命令フェッチサイクルまたはデータアクセスサイクル
4	IDB0	0	R/W	
3	RWB1	0	R/W	読み出し / 書き込みセレクト B チャンネル B ブレイク条件のバスサイクルとして読み出しサイクルまたは書き込みサイクルを選択します。 00 : 条件比較を行わない 01 : ブレイク条件は、読み出しサイクル 10 : ブレイク条件は、書き込みサイクル 11 : ブレイク条件は、読み出しサイクルまたは書き込みサイクル
2	RWB0	0	R/W	

ビット	ビット名	初期値	R/W	説明
1	SZB1	0	R/W	オペランドサイズセレクト B
0	SZB0	0	R/W	チャンネル B ブレイク条件のバスサイクルのオペランドサイズを選択します。 00: ブレイク条件は、オペランドサイズを含まない 01: ブレイク条件は、バイトアクセス 10: ブレイク条件は、ワードアクセス 11: ブレイク条件は、ロングワードアクセス

25.2.9 ブレイクコントロールレジスタ (BRCR)

BRCR は、次の条件を設定します。

1. チャンネル A と B を 2 つの独立したチャンネル条件か、あるいは 1 つの連続した条件として使用するかを指定します。
2. ブレイクを命令実行の前に設定するか後に設定するかを指定します。
3. チャンネル B 比較条件に実行回数を含めるかどうかを指定します。
4. チャンネル B 比較条件にデータバスの値を含めるかどうかを決定します。
5. PC トレースをイネーブルにします。

BRCR は、ブレイク条件一致フラグと種々のブレイク条件をセットするためのビットを持つ 32 ビットの読み出しまたは書き込み可能なレジスタです。

ビット	ビット名	初期値	R/W	説明
31 ~ 16		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。 1 を書いた場合の動作の保証はできません。
15	SCMFCA	0	R/W	L バスサイクル条件一致フラグ A チャンネル A にセットしたブレイク条件の L バスサイクル条件を満足すると、このフラグは 1 にセットされます。このフラグをクリアするには、このビットに 0 を書き込みます。 0: チャンネル A に対する L バスサイクル条件不一致 1: チャンネル A に対する L バスサイクル条件一致
14	SCMFCB	0	R/W	L バスサイクル条件一致フラグ B チャンネル B にセットしたブレイク条件の L バスサイクル条件を満足すると、このフラグは 1 にセットされます。このフラグをクリアするには、このビットに 0 を書き込みます。 0: チャンネル B に対する L バスサイクル条件不一致 1: チャンネル B に対する L バスサイクル条件一致

25. ユーザブレークコントローラ (UBC)

ビット	ビット名	初期値	R/W	説明
13	SCMFDA	0	R/W	Iバスサイクル条件一致フラグ A チャンネル A にセットしたブレーク条件の Iバスサイクル条件を満足すると、このフラグは 1 にセットされます。このフラグをクリアするには、このビットに 0 を書き込みます。 0 : チャンネル A に対する Iバスサイクル条件不一致 1 : チャンネル A に対する Iバスサイクル条件一致
12	SCMFDB	0	R/W	Iバスサイクル条件一致フラグ B チャンネル B にセットしたブレーク条件の Iバスサイクル条件を満足すると、このフラグは 1 にセットされます。このフラグをクリアするには、このビットに 0 を書き込みます。 0 : チャンネル B に対する Iバスサイクル条件不一致 1 : チャンネル B に対する Iバスサイクル条件一致
11	PCTE	0	R/W	PC トレースイネーブル 0 : PC トレースを禁止 1 : PC トレースを許可
10	PCBA	0	R/W	PC ブレークセレクト A チャンネル A に対する命令フェッチサイクルのブレークタイミングが命令実行の前か後かを選択します。 0 : チャンネル A の PC ブレークを命令実行前に設定 1 : チャンネル A の PC ブレークを命令実行後に設定
9、8		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
7	DBEB	0	R/W	データブレークイネーブル B データバス条件がチャンネル B のブレーク条件に含まれるかどうかを選択します。 0 : データバス条件がチャンネル B のブレーク条件に含まれない 1 : データバス条件がチャンネル B のブレーク条件に含まれる
6	PCBB	0	R/W	PC ブレークセレクト B チャンネル B に対する命令フェッチサイクルのブレークタイミングが命令実行の前か後かを選択します。 0 : チャンネル B の PC ブレークを命令実行前に設定 1 : チャンネル B の PC ブレークを命令実行後に設定
5、4		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。

ビット	ビット名	初期値	R/W	説明
3	SEQ	0	R/W	シーケンス条件セレクト チャンネル A および B の 2 つの条件が独立した条件であるか連続した条件であるかを選択します。 0: 独立した条件下でチャンネル A とチャンネル B を比較 1: 連続した条件下でチャンネル A とチャンネル B を比較 (チャンネル A、次にチャンネル B)
2、1		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
0	ETBE	0	R/W	実行回数ブ레이크イネーブル チャンネル B に対してのみ実行回数ブ레이크条件を有効にします。このビットが 1 の場合、生じたブ레이크条件の数が BETR レジスタで指定した実行回数と等しくなったときユーザブ레이크割り込みが出されます。 0: チャンネル B の実行回数ブ레이크条件を無効にする 1: チャンネル B の実行回数ブ레이크条件を有効にする

25.2.10 実行回数ブ레이크レジスタ (BETR)

BETR は、16 ビットの読み出しまたは書き込み可能なレジスタです。チャンネル B の実行回数ブ레이크条件を有効にすると、このレジスタはブ레이크を行う回数を指定します。最大値は、 $2^{12} - 1$ 回です。ブ레이크条件を満たすたびに BETR は、1 ずつデクリメントされます。BETR が H'0001 になった後は、ブ레이크条件を満たすとブ레이크が出されます。

ビット	ビット名	初期値	R/W	説明
15~12		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。1 を書いた場合の動作の保証はできません。
11~0	BET11~ BET0	すべて 0	R/W	実行回数

【注】 チャンネル B のブ레이크条件を命令フェッチサイクルでのブ레이크に設定し、ブ레이크する命令が下記の命令に該当する場合、1 回のブ레이크で BETR は 1 ずつデクリメントされません。デクリメントされる値は次のようになります。

25. ユーザブ레이크コントローラ (UBC)

命令	カウントダウン値
RTE	4
DMULS.L Rm,Rn	2
DMULU.L Rm,Rn	2
MAC.L @Rm+,@Rn+	2
MAC.W @Rm+,@Rn+	2
MUL.L Rm,Rn	3
AND.B #imm,@(R0,GBR)	3
OR.B #imm,@(R0,GBR)	3
TAS.B @Rn	3
TST.B #imm,@(R0,GBR)	3
XOR.B #imm,@(R0,GBR)	3
LDC Rm,SR	4
LDC Rm,GBR	4
LDC Rm,VBR	4
LDC Rm,SSR	4
LDC Rm,SPC	4
LDC Rm,R0_BANK	4
LDC Rm,R1_BANK	4
LDC Rm,R2_BANK	4
LDC Rm,R3_BANK	4
LDC Rm,R4_BANK	4
LDC Rm,R5_BANK	4
LDC Rm,R6_BANK	4
LDC Rm,R7_BANK	4
LDC.L @Rm+,SR	6
LDC.L @Rm+,GBR	4
LDC.L @Rm+,VBR	4
LDC.L @Rm+,SSR	4
LDC.L @Rm+,SPC	4
LDC.L @Rm+,R0_BANK	4
LDC.L @Rm+,R1_BANK	4
LDC.L @Rm+,R2_BANK	4
LDC.L @Rm+,R3_BANK	4
LDC.L @Rm+,R4_BANK	4
LDC.L @Rm+,R5_BANK	4

LDC.L @Rm+,R6_BANK	4
LDC.L @Rm+,R7_BANK	4
LDC.L @Rn+,MOD	4
LDC.L @Rn+,RS	4
LDC.L @Rn+,RE	4
LDC Rn,MOD	4
LDC Rn,RS	4
LDC Rn,RE	4
BSR label	2
BSRF Rm	2
JSR @Rm	2

25.2.11 ブランチソースレジスタ (BRSR)

BRSR は、32 ビットの読み出し専用レジスタです。BRSR は、分岐元命令のアドレスのビット 27~0 を格納します。BRSR は、分岐発生時に 1 にセットされるフラグビットを持っています。このフラグビットは、BRSR の読み出し時、PC トレース禁止状態からの許可設定時、またはパワーオンリセットで 0 にクリアされます。その他のビットは、リセットによっては初期化されません。8 本の BRSR レジスタはキュー構造を持ち、格納したレジスタは分岐ごとにシフトされます。

ビット	ビット名	初期値	R/W	説 明
31	SVF	0	R	BRSR 有効フラグ 分岐元のアドレスが格納されているかどうかを示します。分岐元アドレスが取り出されるとき、このフラグは 1 にセットされます。このフラグは、BRSR を読み出すことによって 0 にクリアされます。 0 : BRSR レジスタの値は無効 1 : BRSR レジスタの値は有効
30~ 28		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
27~0	BSA27~ BSA0		R	分岐元アドレス これらのビットは、分岐元アドレスのビット 27~0 を格納します。

25.2.12 ブランチデスティネーションレジスタ (BRDR)

BRDR は、32 ビットの読み出し専用レジスタです。BRDR は、分岐先命令のアドレスのビット 27~0 を格納します。BRDR は、分岐発生時に 1 に設定されるフラグビットを持っています。このフラグビットは、BRDR の読み出し時、PC トレース禁止状態からの許可設定時、またはパワーオンリセットで 0 にクリアされます。その他のビットは、リセットによっては初期化されません。8 本の BRDR レジスタはキュー構造を持ち、格納したレジスタは分岐ごとにシフトされます。

ビット	ビット名	初期値	R/W	説明
31	DVF	0	R	BRDR 有効フラグ 分岐先アドレスが格納されているかどうかを示します。分岐先アドレスが取り出されるときこのフラグは 1 にセットされます。このフラグは、BRDR を読み出すことによって 0 にクリアされます。 0 : BRDR レジスタの値は無効 1 : BRDR レジスタの値は有効
30~28		すべて 0	R	リザーブビット 読み出すと常に 0 が読み出されます。書き込む値も常に 0 にしてください。
27~0	BDA27~ BDA0		R	分岐元アドレス これらのビットは、分岐先アドレスのビット 27~0 を格納します。

25.3 動作説明

25.3.1 ユーザブレイク動作の流れ

ブレイク条件の設定からユーザブレイク例外処理までの動作の流れは、次のとおりです。

1. ブレイクアドレスは、ブレイクアドレスレジスタ (BARAまたはBARB) にセットします。マスクするアドレスは、ブレイクアドレスマスクレジスタ (BAMRAまたはBAMRB) にセットします。ブレイクデータは、ブレイクデータレジスタ (BDRB) にセットします。マスクするデータは、ブレイクデータマスクレジスタ (BDMRB) にセットします。バスブレイク条件は、ブレイクバスサイクルレジスタ (BBRAまたはBBRB) にセットします。BBRAまたはBBRBの3つの制御ビットペア、すなわちLバスサイクルまたはIバスサイクルセレクト、命令フェッチまたはデータアクセスセレクト、読み出しまたは書き込みセレクトの3つの制御ビットペアのどれか1つでも00の場合は、ユーザブレイクは発生しません。ブレイク制御は、BRCRのビットにセットします。他のすべてのブレイク関連レジスタの設定後にBBRAまたはBBRBの設定を行ってください。
2. ブレイク条件を満足すると、UBCはユーザブレイク要求をCPUに通知するとともに、それぞれのチャンネルに対するLバス条件一致フラグ (SCMFCAまたはSCMFCB) およびIバス条件一致フラグ (SCMFDAまたはSCMFDB) をセットします。チャンネルBでX/Yメモリバスを指定した場合は、条件一致フラグはSCMFCBが使用されます。
3. 設定条件の一致または不一致をチェックするため、該当する条件一致フラグ (SCMFCA、SCMFDA、SCMFCB、SCMFDB) を使用することができます。フラグは条件一致によりセットされますが、リセットされません。フラグを再び使用できるようにするためには、まず0を書き込まなければなりません。
4. チャンネルAおよびチャンネルBで設定したブレイクがほぼ同時に発生する場合があります。CPUに対するブレイク割り込み要求は1つだけであっても、これらの2つのブレイクチャンネル一致フラグは2つともセットされる場合があります。
5. ブレイク条件としてIバスを選択した場合は、次のことに注意してください。
 - Iバスには、CPUとDMACを含む複数のバスマスタが接続されています。UBCはすべてのバスマスタの生成するバスサイクルを監視し、条件一致判定を行います。
 - Iバスで使用されるアドレスは、物理アドレスです。ブレイクアドレスレジスタ (BARAおよびBARB) には物理アドレスを設定してください。CPUがLバス上で発行した論理アドレスのバスサイクルは、物理アドレスに変換されたのちにIバスに出力されます。(アドレス変換機構が有効な場合は、MMUによるアドレス変換も行われます。)
 - CPUがLバス上で発行したデータアクセスサイクルは、論理アドレスがキャッシング対象で無い場合はLバスで指定したサイズのバスサイクルが発行され、アドレスも丸められることはありません。
 - CPUがLバス上で発行した命令フェッチサイクルは、論理アドレスがキャッシング対象で無い場合もすべてロングワードサイズとなり、アドレスはロングワード境界に丸められます。

25. ユーザブレイクコントローラ (UBC)

- CPUがLバス上で発行した論理アドレスがキャッシング対象のアドレスでありかつキャッシュミスした場合は、キャッシュフィルサイクルとしてIバスに発行されます。この場合はロングワードサイズで発行され、アドレスもロングワード境界に丸められた値が使用されます。ただし、ライトスルーモードにおけるライトミス時にはキャッシュフィルは行われず、Lバス上で指定したサイズのバスサイクルが発行され、アドレスも丸められることはありません。また、ライトバックモードの場合は、リードフィルサイクルに伴ってライトバックサイクルが発行されるときがあります。これもロングワードサイズのバスサイクルで、アドレスもロングワード境界に丸められています。
 - CPUのLバス上での命令フェッチに起因したIバスサイクル(リードフィルサイクルを含む)をIバスでの命令フェッチサイクル、それ以外をデータアクセスサイクルと定義します。
 - DMACが発行するIバスサイクルは、データアクセスサイクルのみです。
 - Iバスにブレイク条件を設定している場合は、CPUが実行した命令に起因したIバスサイクルで条件が一致したときでも、どの命令でブレイクを受け付けるかを一意に決定することはできません。
6. CPUのステータスレジスタ (SR) のブロックビット (BL) がIの期間は、すべてのブレイクは受け付けられません。ただし、条件判定は行われ、一致した場合は条件一致フラグがセットされます。

25.3.2 命令フェッチサイクルでのブレイク

1. ブレイクバスサイクルレジスタ (BBRAまたはBBRB) にLバス/命令フェッチ/読み出し/ワードまたはロングワードが設定されると、ブレイク条件はLバスの命令フェッチになります。命令実行の前にブレイクするか後にブレイクするかは、該当するチャンネルに対するブレイクコントロールレジスタ (BRCR) のPCBAまたはPCBBビットで選択できます。ブレイク条件として命令フェッチサイクルを設定する場合は、ブレイクアドレスレジスタ (BARAまたはBARB) のLSBを0にクリアしてください。このビットが1にセットされているとブレイクは発生しません。
2. 命令フェッチによるブレイクがその命令を実行する前に行われるように設定されている状態で条件が一致した場合は、命令がフェッチされて命令を実行することが確定した時点でブレイクが生じます。したがって、この機能はオーバラン (分岐または割り込みの遷移中にフェッチされ、しかも実行されない命令) によってフェッチされる命令には使用できません。遅延分岐命令の遅延スロットに対してこの種のブレイクが設定されると、遅延分岐命令の実行前にブレイクが発生します。

【注】遅延条件分岐命令が分岐しなかった場合は、その後続命令は遅延スロットとはみなされません。

3. ブレイク条件でブレイクが命令実行後に起こるように設定している場合は、ブレイク条件と一致した命令が実行され、次の命令の実行前にブレイクが発生します。実行前のブレイクの場合と同様、これはオーバランフェッチ命令では使用できません。遅延分岐命令およびその遅延スロットに対してこの種のブレイクが設定されると、分岐先の最初の命令までブレイクは発生しません。
4. 命令フェッチサイクルがチャンネルBに設定されるとブレイクデータレジスタB (BDRB) は、無視されます。したがって、命令フェッチサイクルのブレイクには、ブレイクデータを設定することはできません。

5. 命令フェッチサイクルでのブレイクにおいてIバスを設定した場合は、Iバス上の命令フェッチサイクルに対する条件判定が行われます。詳細は、「25.3.1 ユーザブレイク動作の流れ」の5.の項を参照してください。

25.3.3 データアクセスサイクルでのブレイク

1. データアクセスブレイクにおいて、ブレイク条件としてIバスを指定した場合は、実行された命令によりアクセスされた論理アドレス(およびデータ)に対して条件比較を行いブレイクを発生します。ブレイク条件としてIバスを指定した場合は、Iバス上のCPUを含めたすべてのバスマスタが発行するデータアクセスサイクルの物理アドレス(およびデータ)に対して条件比較を行いブレイクを発生します。Iバス上に発行されるCPUのバスサイクルに関しては、「25.3.1 ユーザブレイク動作の流れ」の5.の項を参照してください。
2. 表25.3にデータアクセスサイクルアドレスと各オペランドサイズについての比較条件の関係を示します。

表 25.3 データアクセスサイクルアドレスおよびオペランドサイズの比較条件

アクセスサイズ	比較アドレス
ロングワード	ブレイクアドレスレジスタのビット31~2とアドレスバスのビット31~2を比較
ワード	ブレイクアドレスレジスタのビット31~1とアドレスバスのビット31~1を比較
バイト	ブレイクアドレスレジスタのビット31~0とアドレスバスのビット31~0を比較

これは、たとえばブレイクアドレスレジスタ (BARA / BARB) にアドレスH'00001003を設定するとき、ブレイク条件を満足するバスサイクルには、(他のすべての条件が満足されると仮定した場合) 以下が含まれることを意味します。

H'00001000 でのロングワードアクセス

H'00001002 でのワードアクセス

H'00001003 でのバイトアクセス

3. チャンネルBのブレイク条件にデータ値が含まれる場合

ブレイク条件にデータ値が含まれる場合は、ブレイクバスサイクルレジスタB (BBRB) にロングワード、ワード、またはバイトをオペランドサイズとして指定します。データ値がブレイク条件に含まれる場合は、アドレス条件とデータ条件が一致するときブレイクが発生します。この場合、バイトデータを指定するためには、ブレイクデータレジスタB (BDRB) とブレイクデータマスクレジスタB (BDMRB) のビット15~8、ビット7~0の2バイトに同じデータをセットします。ワードまたはバイトのオペランドサイズを選択すると、BDRBとBDMRBのビット31~16は、無視されます。ただし、MOVS.W @-As,Ds、MOVS.W @As,Ds、MOVS.W @As+,Ds、MOVS.W @As+Ix,Ds命令に対するブレイク条件にデータ値を含む場合は、BDRBとBDMRBのビット31~16にワードデータをセットします(ビット15~0は無視されます)。

4. PREF命令は、アクセスデータのないロングワードの読み出しアクセスとして扱います。したがって、PREF命令のブレイク条件にデータ値が含まれる場合は、ブレイクは発生しません。

25. ユーザブレイクコントローラ (UBC)

5. Lバスを選択している場合は、条件が一致した命令の実行を完了し、次の命令を実行する直前にブレイクが発生します。ただし、条件にデータ値を含める場合は、条件が一致した命令の次の命令の実行完了後になる場合もあります。Iバスを選択している場合は、ブレイクの発生する命令を特定することはできません。また、遅延分岐命令やその遅延スロットでこの種のブレイクが発生した場合は、分岐先の最初の命令までブレイクは発生しません。

25.3.4 XメモリまたはYメモリバスサイクルでのブレイク

1. XメモリバスサイクルまたはYメモリバスサイクルに対するブレイク条件は、チャンネルBでのみ指定できます。BBRBのXYEを1にセットした場合は、XメモリまたはYメモリバス上のブレイクアドレスおよびブレイクデータが選択されます。BBRBのXYSを指定することによってXメモリバスまたはYメモリバスのどちらかを選択する必要があります。ブレイク条件には、XメモリとYメモリを同時に含めることはできません。ブレイク条件は、ブレイクバスサイクルレジスタB (BBRB) でLバス、データアクセスサイクル、読み出または書き込みアクセス、オペランドサイズをワード、またはオペランドサイズを指定しないに設定することによって、XメモリバスサイクルまたはYメモリバスサイクルに適用されます。
2. ブレイク条件としてXメモリアドレスを選択するときはBARBとBAMRBの上位16ビットにXメモリアドレスを指定し、Yメモリアドレスを選択するときは下位16ビットにYメモリアドレスを指定してください。BDRBとBDMRBに対するXメモリデータまたはYメモリデータの指定は、同様の方法で行います。
3. XメモリまたはYメモリバスのデータアクセスブレイクの発生タイミングは、Lバスのデータアクセスブレイクの場合と同じになります。詳細は、「25.3.3 データアクセスサイクルでのブレイク」の5の項を参照してください。

25.3.5 シーケンシャルブレイク

1. BRRCRのSEQビットを1にセットすると、チャンネルAブレイク条件が一致した後、チャンネルBブレイク条件が一致するときにシーケンシャルブレイクが発生します。チャンネルAブレイク条件が一致する前にチャンネルBブレイク条件が一致すると、ユーザブレイクは発生しません。また、チャンネルAとチャンネルBのブレイク条件が同時に一致したときも、シーケンシャルブレイクは発生しません。シーケンシャルブレイク指定時、チャンネルA条件が一致し、かつチャンネルB条件が一致していないときにチャンネルA一致をクリアしたい場合は、BRRCRレジスタのSEQビットに0を書き込みます。
2. シーケンシャルブレイク指定では、Lバス、Iバス、Xバス、またはYバスを選択でき、実行回数ブレイク条件も指定することができます。たとえば、実行回数ブレイク条件を指定すると、チャンネルAブレイク条件一致後、チャンネルBブレイク条件がBETR = H'0001のときに一致するとブレイク条件が満たされます。

25.3.6 退避されるプログラムカウンタの値

ブレイク発生時は、実行を再開すべき命令のアドレスを SPC に退避し、例外処理状態に移行します。ブレイク条件として L バスを指定している場合は、ブレイクの発生する命令を一意に決定することができます (ブレイク条件にデータを含む場合を除く)。ブレイク条件として I バスを指定している場合は、ブレイクの発生する命令を一意に決定することはできません。

1. 命令フェッチを (命令実行前) ブレイク条件として指定する場合

SPC には、ブレイク条件と一致した命令のアドレスが退避されます。条件が一致した命令は実行されず、その前にブレイクが発生します。ただし、遅延スロット命令で条件が一致した場合は、遅延分岐命令のアドレスが SPC に退避されます。

2. 命令フェッチを (命令実行後) ブレイク条件として指定する場合

SPC には、ブレイク条件と一致した命令の次の命令のアドレスが退避されます。条件が一致した命令は実行され、次の命令の実行前にブレイクが発生します。遅延分岐命令やその遅延スロットで一致した場合は、それらの命令は実行され、分岐先のアドレスが SPC に退避されます。

3. データアクセス (アドレスのみ) をブレイク条件として指定する場合

SPC には、ブレイク条件に一致した命令の直後の命令のアドレスが退避されます。条件に一致した命令が実行され、次の命令の実行前にブレイクが発生します。ただし、遅延スロットで条件が一致した場合は、分岐先のアドレスが SPC に退避されます。

4. データアクセス (アドレス + データ) をブレイク条件として指定する場合

データ値がブレイク条件に追加されると、ブレイク条件に一致した命令の次の命令が、その次の命令のアドレスが SPC に退避されます。ブレイクが発生する場所は、正確に決定することはできません。

遅延スロット命令で条件が一致した場合は、分岐先アドレスが SPC に退避されます。また、条件に一致した命令の次命令が分岐命令である場合は、分岐命令や遅延スロットの実行まで完了した後にブレイクが生じる場合があります。この場合も SPC には、分岐先のアドレスが退避されます。

25.3.7 PC トレース

1. PCトレースは、BRCRのPCTEを1にセットすることによってイネーブルになります。分岐（分岐命令および割り込み例外）が発生すると、分岐元アドレスと分岐先アドレスがそれぞれBRSRとBRDRに格納されます。
2. BRSR, BRDRに格納される値は、分岐の種類によってそれぞれ次のようになります。
 - 分岐命令により分岐が生じる場合は、分岐命令のアドレスがBRSRに、分岐先命令のアドレスがBRDRに格納されます。
 - 割り込みや一般例外により分岐が生じる場合は、例外発生により保存されるSPCの値がBRSRに、例外処理ルーチンの先頭アドレスがBRDRに格納されます。

DSP拡張機能のリピートループを使用している場合は、リピート終了命令からリピート開始命令に制御が移ってもこの動作は分岐とはみなされず、BRSRとBRDRには格納されません。

3. BRSRとBRDRは、8組のキュー構造からなっています。PCトレースレジスタに格納されたアドレスの読み出し時、キューの先頭を最初に読み出します。BRSRとBRDRは、リードポインタを共有します。BRSR、BRDRの順で読み出してください。キューは、BRDRの読み出し後のみシフトされます。BRCRのPCTEビットをオフからオンに切り換えると、キューの値は無効になります。

25.3.8 使用例

(1) Lバス命令フェッチサイクルに指定したブ레이크条件

(例 1-1)

- レジスタ指定

BARA = H'00000404、BAMRA = H'00000000、BBRA = H'0054、BARB = H'00008010、BAMRB = H'00000006、
BBRB = H'0054、BDRB = H'00000000、BDMRB = H'00000000、BRCR = H'00300400

指定条件： チャンネルA / チャンネルB独立モード

<チャンネル A>

アドレス： H'00000404、アドレスマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ（命令実行後） / 読み出し（オペランドサイズは条件に含まれません）

<チャンネル B>

アドレス： H'00008010、アドレスマスク： H'00000006

データ： H'00000000、データマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ（命令実行前） / 読み出し（オペランドサイズは条件に含まれません）

ユーザブレイクは、アドレスH'00000404の命令実行後、またはアドレスH'00008010 ~ H'00008016の命令の実行前に発生します。

(例 1-2)

- レジスタ指定

BARA = H'00037226、BAMRA = H'00000000、BBRA = H'0056、BARB = H'0003722E、BAMRB = H'00000000、
BBRB = H'0056、BDRB = H'00000000、BDMRB = H'00000000、
BRRCR = H'00000008

指定条件： チャンネルA / チャンネルBシーケンシャルモード

<チャンネル A>

アドレス： H'00037226、アドレスマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 読み出し / ワード

<チャンネル B>

アドレス： H'0003722E、アドレスマスク： H'00000000

データ： H'00000000、データマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 読み出し / ワード

アドレスH'00037226の命令が実行された後、アドレスH'0003722Eの命令実行前にユーザブレイクが発生しません。

(例 1-3)

- レジスタ指定

BARA = H'00027128、BAMRA = H'00000000、BBRA = H'005A、BARB = H'00031415、BAMRB = H'00000000、
BBRB = H'0054、BDRB = H'00000000、BDMRB = H'00000000、BRRCR = H'00300000

指定条件： チャンネルA / チャンネルB独立モード

<チャンネル A>

アドレス： H'00027128、アドレスマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 書き込み / ワード

<チャンネル B>

アドレス： H'00031415、アドレスマスク： H'00000000

データ： H'00000000、データマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 読み出し (オペランドサイズは条件に含まれません)

チャンネルAでは、命令フェッチは書き込みサイクルではないのでユーザブレイクは生じません。チャンネルBでは、命令フェッチは偶数アドレスに対して実行されるのでユーザブレイクは発生しません。

25. ユーザブレイクコントローラ (UBC)

(例 1-4)

- レジスタ指定

BARA = H'00037226、BAMRA = H'00000000、BBRA = H'005A、BARB = H'0003722E、BAMRB = H'00000000、
BBRB = H'0056、BDRB = H'00000000、BDMRB = H'00000000、BRCR = H'00000008

指定条件： チャンネルA / チャンネルBシーケンシャルモード

<チャンネル A>

アドレス： H'00037226、アドレスマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 書き込み / ワード

<チャンネル B>

アドレス： H'0003722E、アドレスマスク： H'00000000

データ： H'00000000、データマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 読み出し / ワード

チャンネルAで命令フェッチは書き込みサイクルではないので、シーケンシャル条件一致は生じません。したがって、ユーザブレイクは発生しません。

(例 1-5)

- レジスタ指定

BARA = H'00000500、BAMRA = H'00000000、BBRA = H'0057、BARB = H'00001000、BAMRB = H'00000000、
BBRB = H'0057、BDRB = H'00000000、BDMRB = H'00000000、BRCR = H'00300001、BETR = H'0005

指定条件：チャンネルA / チャンネルB独立モード

<チャンネル A>

アドレス： H'00000500、アドレスマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 読み出し / ロングワード

<チャンネル B>

アドレス： H'00001000、アドレスマスク： H'00000000

データ： H'00000000、データマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 読み出し / ロングワード

実行回数ブレイクイネーブル (5回)

チャンネルAでは、ユーザブレイクはアドレスH'00000500の命令の実行前に生じます。チャンネルBでは、ユーザブレイクは、アドレスH'00001000の命令を4回実行した後、5回目の命令実行前に生じます。

(例 1-6)

- レジスタ指定

BARA = H'00008404、BAMRA = H'0000FFFF、BBRA = H'0054、BARB = H'00008010、BAMRB = H'00000006、
BBRB = H'0054、BDRB = H'00000000、BDMRB = H'00000000、BR CR = H'00000400

指定条件： チャンネルA / チャンネルB独立モード

<チャンネル A>

アドレス： H'00008404、アドレスマスク： H'0000FFFF

バスサイクル： Lバス / 命令フェッチ (命令実行後) / 読み出し (オペランドサイズは条件には含まれません)

<チャンネル B>

アドレス： H'00008010、アドレスマスク： H'00000006

データ： H'00000000、データマスク： H'00000000

バスサイクル： Lバス / 命令フェッチ (命令実行前) / 読み出し (オペランドサイズは条件には含まれません)

ユーザブ레이크は、アドレスH'00008000 ~ H'00008FFEの命令の実行後、または、アドレスH'00008010 ~ H'00008016の命令の実行前に生じます。

(2) Lバスデータアクセスサイクルに指定したブ레이크条件

(例 2-1)

- レジスタ指定

BARA = H'00123456、BAMRA = H'00000000、BBRA = H'0064、BARB = H'000ABCDE、BAMRB = H'000000FF、
BBRB = H'006A、BDRB = H'0000A512、BDMRB = H'00000000、BR CR = H'00000080

指定条件： チャンネルA / チャンネルB独立モード

<チャンネル A>

アドレス： H'00123456、アドレスマスク： H'00000000

バスサイクル： Lバス / データアクセス / 読み出し (オペランドサイズは条件に含まれません)

<チャンネル B>

アドレス： H'000ABCDE、アドレスマスク： H'000000FF

データ： H'0000A512、データマスク： H'00000000

バスサイクル： Lバス / データアクセス / 書き込み / ワード

チャンネルAでは、ユーザブ레이크は、アドレスH'00123454に対するロングワードの読み出し、アドレスH'00123456に対するワード読み出し、あるいはアドレスH'00123456に対するバイト読み出しで生じます。チャンネルBでは、ユーザブ레이크はH'000ABC00 ~ H'000ABCFEにワードH'A512を書き込むときに生じます。

25. ユーザブ레이크コントローラ (UBC)

(例 2-2)

- レジスタ指定

BARA = H'01000000、BAMRA = H'00000000、BBRA = H'0066、BARB = H'0000F000、BAMRB = H'FFFF0000、
BBRB = H'036A、BDRB = H'00004567、BDMRB = H'00000000、BRCR = H'00300080

指定条件： チャンネルA / チャンネルB独立モード

<チャンネル A>

アドレス： H'01000000、アドレスマスク： H'00000000

バスサイクル： Lバス / データアクセス / 読み出し / ワード

<チャンネル B>

Yアドレス： H'0000F000、アドレスマスク： H'FFFF0000

データ： H'00004567、データマスク： H'00000000

バスサイクル： Yバス / データアクセス / 書き込み / ワード

チャンネルAでは、ユーザブ레이크は、メモリ空間のアドレスH'01000000に対するワード読み出しで生じます。チャンネルBでは、ユーザブ레이크はYメモリ空間のアドレスH'0000F000にワードH'4567を書き込むときに生じます。XメモリまたはYメモリ空間は、モード指定によって変更することができます。

(3) Iバスデータアクセスサイクルに指定されたブ레이크条件

(例 3-1)

- レジスタ指定：

BARA = H'00314156、BAMRA = H'00000000、BBRA = H'0094、BARB = H'00055555、BAMRB = H'00000000、
BBRB = H'00A9、BDRB = H'00007878、BDMRB = H'00000F0F、BRCR = H'00000080

指定条件： チャンネルA / チャンネルB独立モード

<チャンネル A>

アドレス： H'00314156、アドレスマスク： H'00000000

バスサイクル： Iバス / 命令フェッチ / 読み出し (オペランドサイズは条件に含まれません)

<チャンネル B>

アドレス： H'00055555、アドレスマスク： H'00000000

データ： H'00000078、データマスク： H'0000000F

バスサイクル： Iバス / データアクセス / 書き込み / バイト

チャンネルAでは、ユーザブ레이크はメモリ空間のアドレスH'00314156に対する命令フェッチで生じます。チャンネルBでは、ユーザブ레이크はIバス上でバイトH'7*をアドレスH'00055555に書き込むときに生じます。

25.4 使用上の注意事項

1. UBCのレジスタの読み出しまたは書き込みは、Iバス経由で行われます。したがって、UBCのレジスタを書き換える命令を実行してから実際にその値が反映されるまでの期間は、所望のブレイクが発生しない場合があります。UBCレジスタが変更されるタイミングを知るためには、最後に書き込んだレジスタを読み出してください。それ以降の命令は、新しく書き込んだレジスタ値に対して有効です。
2. UBCはLバスサイクルとIバスサイクルを同じチャンネルで監視することはできません。
3. シーケンシャルブレイクの指定における注意事項は、次のとおりです。
 - シーケンシャルブレイクの設定時、Aチャンネル一致が発生後Bチャンネル一致が発生するとき、条件一致が発生します。したがって、チャンネルA一致とチャンネルB一致が同時に発生するバスサイクルが設定されてもブレイクは、発生しません。
4. ユーザブレイクと他の例外が同一命令で発生した場合は、「第4章 例外処理」の表4.1に定められた優先順位で判定が行われます。より高い優先度の例外が発生した場合は、ユーザブレイクは発生しません。
 - 命令実行前ブレイクは他のどの例外よりも優先して受け付けられます。
 - 命令実行後ブレイクやデータアクセスブレイクは、より優先度の高い再実行型の例外（命令実行前ブレイクを含む）と同時に発生した場合、再実行型の例外が受け付けられ、条件一致を示すフラグもセットされません（ただし、5項に示す例外事項があります）。例外処理により再実行型の例外要因が解消され、同命令が再実行されて完了する時点で改めてブレイクが発生し、フラグがセットされます。
 - 命令実行後ブレイクやデータアクセスブレイクが、より優先度の高い完了型の例外（TRAPA）と同時に発生した場合は、ブレイクは発生しませんが、条件一致を示すフラグはセットされます。
5. 4項の例外事項として、次の注意事項があります。
 - データアクセスによりCPUアドレスエラーが発生する命令において命令実行後ブレイクやデータアクセスブレイクが成立する場合は、ブレイクに優先してCPUアドレスエラーが発生します。この際、UBCの条件一致フラグもセットされます。
6. 遅延スロットでブレイクが発生する場合は、次の注意事項があります。
 - RTE命令の遅延スロット命令に対して命令実行前ブレイクを設定した場合は、RTE命令の分岐先の実行前までブレイクは発生しません。
7. UBCモジュールスタンバイ時は、ユーザブレイク機能を使用できません。また、モジュールスタンバイ中は、UBCレジスタを読み書きしないでください。読み書きした場合は、その値は保証されません。
8. DSP拡張機能のリピートループを使用している場合は、リピートループ全体あるいは一部の命令を実行中にブレイク条件が一致してもブレイクの発生が保留される場合があります。詳細は、「第4章 例外処理」を参照してください。

26. ユーザデバッグインタフェース (H-UDI)

本 LSI は、バウンダリスキャン機能やエミュレータのサポートのため、ユーザデバッグインタフェース (H-UDI) およびアドバンスドユーザデバッグ (AUD) を内蔵しています。

本章では、H-UDI のバウンダリスキャン機能について概要を説明します。本 LSI に組み込まれている JTAG 機能の詳細については、別紙 BSDL (Boundary-Scan Description Language) 記述の資料を参照ください。AUD を初めとするエミュレータ機能の詳細については、各エミュレータのユーザーズマニュアルを参照してください。

26.1 特長

ユーザデバッグインタフェース (H-UDI) は、JTAG (Joint Test Action Group, IEEE Std.1149.1 and IEEE Standard Test Access Port and Boundary-Scan Architecture) に準拠したシリアル入出力インタフェースです。

なお、TCK 入力は外部でプルアップして使用してください。

本 LSI の H-UDI はバウンダリスキャンをサポートし、エミュレータの接続にも使用されます。

エミュレータを使用する場合は、H-UDI の機能を使用しないでください。H-UDI のバウンダリスキャン機能を使う場合には、必ず ASEMD0 をハイ固定にしてください。エミュレータとの接続方法は、エミュレータのマニュアルを参照してください。

H-UDI のブロック図を図 26.1 に示します。

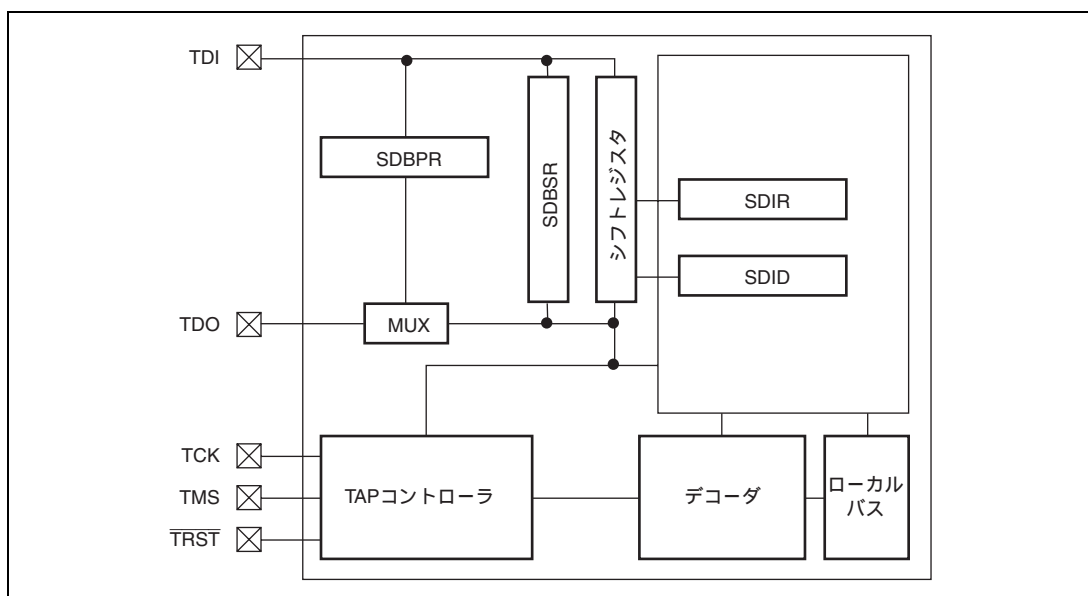


図 26.1 H-UDI ブロック図

26.2 入出力端子

表 26.1 に H-UDI の端子構成を示します。

表 26.1 端子構成

端子名	入出力	説明
TCK	入力	シリアルデータ入出力用クロック端子 外部でプルアップしてください。データはこのクロックに同期してデータ入力端子 (TDI) から H-UDI にシリアルに供給され、データ出力端子 (TDO) から出力されます。
TMS	入力	モードセレクト入力端子 TCK に同期してこの信号を変化させることによって TAP 制御回路の状態が決まります。プロトコルは JTAG 規格 (IEEE Std.1149.1) に準拠しています。
TRST	入力	リセット入力端子 TCK とは非同期で入力を受け付けローレベルで H-UDI をリセットします。H-UDI 機能の利用の有無にかかわらず、電源投入時に TRST を一定期間ローレベルにしなければなりません。これは JTAG の規格と異なります。リセット構成の詳細については「26.4.2 リセット構成」を参照してください。
TDI	入力	シリアルデータ入力端子 TCK に同期してこの端子を変化させることによって H-UDI にデータを送ります。
TDO	出力	シリアルデータ出力端子 TCK に同期してこの端子を読み出すことによって H-UDI からデータを読み取ります。データ出力タイミングは SDIR に設定されているコマンドの種類により異なります。詳細については、「26.3.2 インストラクションレジスタ (SDIR)」を参照してください。
ASEMD0	入力	ASE モードセレクト端子 RESETP 端子アサート期間中に、ASEMD0 端子にローレベルを入力すると ASE モードになり、ハイレベルを入力すると通常モードになり、JTAG の機能が使えます。ASE モードでは、エミュレータ専用の機能が使用可能になります。ASEMD0 端子への入力は、High/Low どちらかに固定してください。*
ASEBRKAK AUDSYNC AUDATA3~0 AUDCK	出力	エミュレータ専用の端子

【注】 * パウダリスキャン機能を使うときは、必ず ASEMD0 を High 固定にしてください。

26.3 レジスタの説明

H-UDI には以下のレジスタがあります。レジスタのアドレスおよび各処理状態におけるレジスタの状態については「第 27 章 レジスタ一覧」を参照してください。*

- バイパスレジスタ (SDBPR)
- インストラクションレジスタ (SDIR)
- バウンダリスキャンレジスタ (SDBSR)
- IDレジスタ (SDID)

【注】 * H-UDI のレジスタが CPU からアクセス可能になるのは、ASE モード時のエミュレータ機能が有効になっているときのみとなっています。これらのレジスタは通常モード (バウンダリスキャン機能を使用するモード) では、JTAG 機能専用回路の一部として機能しますので、CPU からアクセスしないでください。

26.3.1 バイパスレジスタ (SDBPR)

SDBPR は 1 ビットのレジスタです。H-UDI の端子からのみアクセス可能です。CPU からはアクセスすることはできません。SDIR をバイパスモードにセットすると、SDBPR は H-UDI 端子の TDI と TDO の間に接続されます。初期値は不定です。

26.3.2 インストラクションレジスタ (SDIR)

通常モードのとき、SDIR は 3 ビットのレジスタです。H-UDI の端子からのみ書き込み可能です。CPU からはアクセスすることはできません。「第 27 章 レジスタ一覧」に記載されているアドレスを使って CPU から読み出しても、不定が出力されます。初期状態でこのレジスタは JTAG IDCODE になっています。このレジスタに予約となっているコマンドをセットした場合の動作は保証できません。

ASE モードのとき、SDIR は CPU からは読み出しのみ可能な 16 ビットのレジスタです。JTAG の機能は無効になります。ASE モードでの SDIR の機能については、エミュレータのユーザーズマニュアルを参照してください。

ビット	ビット名	初期値	R/W	説明
2	TI2	1	-	テストインストラクション 2~0 JTAG のインストラクションは、TDI からのシリアル入力によって SDIR に転送されます。CPU からは R/W アクセスできません。 コマンドは表 26.2 を参照してください。
1	TI1	0	-	
0	TI0	0	-	

26. ユーザデバッグインタフェース (H-UDI)

表 26.2 JTAG コマンド

ビット 2~0			説 明
TI2	TI1	TI0	
0	0	0	JTAG EXTEST
0	0	1	JTAG SAMPLE/PRELOAD
0	1	0	予約
0	1	1	予約
1	0	0	JTAG IDCODE (初期値)
1	0	1	予約
1	1	0	予約
1	1	1	JTAG BYPASS

26.3.3 バウンダリスキャンレジスタ (SDBSR)

SDBSR は、本 LSI の入出力端子の制御を行うために PAD 上に配置されたシフトレジスタです。初期値は不定です。CPU からはアクセスできません。

EXTEST、SAMPLE/PRELOAD コマンドを用いて、JTAG 規格に対応したバウンダリスキャンテストを行うことができます。表 26.3 に本 LSI の端子とバウンダリスキャンレジスタの対応を示します。

表 26.3 本 LSI の端子とバウンダリスキャンレジスタの対応

ビット名	端子名	入出力	ビット名	端子名	入出力
0	ASEBRKAK	IN	36	PTH1/SCIF1_TXD	IN
1		Control	37		Control
2		OUT	38		OUT
3	PTA0/SIOF_MCLK	IN	39	PTH2/SCIF1_RXD	IN
4		Control	40		Control
5		OUT	41		OUT
6	PTA1/SIOF_TXD	IN	42	PTH3/SCIF1_RTS	IN
7		Control	43		Control
8		OUT	44		OUT
9	PTA2/SIOF_RXD	IN	45	PTH4/SCIF1_CTS	IN
10		Control	46		Control
11		OUT	47		OUT
12	PTA3/SIOF_SYNC(SIOF_SS0)	IN	48	NMI	IN
13		Control	49	PTH0/SCIF1_SCK/IRQ4	IN
14		OUT	50		Control
15	PTA4/SIOF_SCK	IN	51		PTG0/SCIF0_SCK/IRQ3
16		Control	52	Control	
17		OUT	53	OUT	
18	PTA5/SIOF_SS1	IN	54	PTB1/IRQ2	IN
19		Control	55		Control
20		OUT	56		OUT
21	PTA6/SIOF_SS2	IN	57	PTB0/IRQ0	IN
22		Control	58		Control
23		OUT	59		OUT
24	PTG1/SCIF0_TXD	IN	60	RTCSEL0	IN
25		Control	61		Control
26		OUT	62		OUT
27	PTG2/SCIF0_RXD	IN	63	USB_PWR_EN/USB_PULLUP	IN
28		Control	64		Control
29		OUT	65		OUT
30	PTG3/SCIF0_RTS	IN	66	UCLK	IN
31		Control	67		Control
32		OUT	68		OUT
33	PTG4/SCIF0_CTS	IN	69	USB_OVR_CRNT/USB_VBUS	IN
34		Control	70		Control
35		OUT	71		OUT
				CKIO	IN

26. ユーザデバッグインタフェース (H-UDI)

ビット名	端子名	入出力	ビット名	端子名	入出力
72	CKIO	Control	109	AUDATA1	OUT
73		OUT	110	AUDATA0	IN
74	MD5	IN	111		Control
75	MD2	IN	112		OUT
76	MD1	IN	113	AUDCK	IN
77	PTE6/RDI_CTRL2/IRQOUT	IN	114		Control
78		Control	115		OUT
79		OUT	116	\overline{RD}	IN
80	PTE0/VCI_CODEC_PWRDWN/TEND	IN	117		Control
81		Control	118		OUT
82		OUT	119	PTB7/BS	IN
83	PTE1/VCI_SCO_TX/DACK	IN	120		Control
84		Control	121		OUT
85		OUT	122	PTB6/REFOUT	IN
86	PTE2/VCI_SCO_RX/DREQ	IN	123		Control
87		Control	124		OUT
88		OUT	125	\overline{WAIT}	IN
89	PTE3/VCI_SCO_CLK_OUT/BACK	IN	126	$\overline{WE1/DQM1}$	IN
90		Control	127		Control
91		OUT	128		OUT
92	PTE4/VCI_SCO_SYNC_OUT/BREQ	IN	129	$\overline{WE0/DQM0}$	IN
93		Control	130		Control
94		OUT	131		OUT
95	PTE5/VCI_HWC	IN	132	PTB2/CKE	IN
96		Control	133		Control
97		OUT	134		OUT
98	AUDSYNC	IN	135	PTB3/ \overline{CAS}	IN
99		Control	136		Control
100		OUT	137		OUT
101	AUDATA3	IN	138	PTB4/ \overline{RAS}	IN
102		Control	139		Control
103		OUT	140		OUT
104	AUDATA2	IN	141	PTB5/RD/ \overline{WR}	IN
105		Control	142		Control
106		OUT	143		OUT
107	AUDATA1	IN	144	PTD5/A23	IN
108		Control	145		Control

26. ユーザデバッグインタフェース (H-UDI)

ビット名	端子名	入出力	ビット名	端子名	入出力
146	PTD5/A23	OUT	183	A6	OUT
147	PTD4/A22	IN	184	A5	Control
148		Control	185		OUT
149		OUT	186		Control
150	PTD3/A21	IN	187	A4	OUT
151		Control	188		Control
152		OUT	189		A3
153	PTD2/A20	IN	190	A2	Control
154		Control	191		OUT
155		OUT	192		Control
156	PTD1/A19	IN	193	A1	OUT
157		Control	194		Control
158		OUT	195		PTD0/A0
159	A18	Control	196	PTD0/A0	Control
160		OUT	197		OUT
161	A17	Control	198	D15	IN
162		OUT	199		Control
163	A16	OUT	200		D14
164		Control	201	IN	
165	A15	OUT	202	D14	Control
166		Control	203		OUT
167	A14	OUT	204	D13	IN
168		Control	205		Control
169	A13	OUT	206		D13
170		Control	207	D12	
171	A12	OUT	208	D12	Control
172		Control	209		OUT
173	A11	OUT	210	D11	IN
174		Control	211		Control
175	A10	OUT	212		D11
176		Control	213	D10	
177	A9	OUT	214	D10	Control
178		Control	215		OUT
179	A8	OUT	216	D9	IN
180		Control	217		Control
181	A7	OUT	218		D9
182		Control	219	D8	

26. ユーザデバッグインタフェース (H-UDI)

ビット名	端子名	入出力	ビット名	端子名	入出力
220	D8	Control	238	D2	Control
221		OUT	239		OUT
222	D7	IN	240	D1	IN
223		Control	241		Control
224		OUT	242		OUT
225	D6	IN	243	D0	IN
226		Control	244		Control
227		OUT	245		OUT
228	D5	IN	246	$\overline{CS4}$	IN
229		Control	247		Control
230		OUT	248		OUT
231	D4	IN	249	$\overline{CS3}$	IN
232		Control	250		Control
233		OUT	251		OUT
234	D3	IN	252	$\overline{CS0}$	IN
235		Control	253		Control
236		OUT	254		OUT
237	D2	IN	255	BOOT_E	IN

26.3.4 ID レジスタ (SDID)

SDID は 32 ビットレジスタで、バージョン、パーツ番号、製造者番号および固定コードから構成されています。このレジスタは、通常モードでも CPU から SDIDH/SDIDL の 2 つの 16 ビットレジスタとして読み出し可能です。書き込みはできません。

H-UDI 端子側からのアクセスはできません。

ビット	ビット名	初期値	R/W	説明
31~0	DID31~ DID0	説明を 参照	R	デバイス ID31~0 JTAG に規定されている ID レジスタです。本 LSI では、H'0028200F (初期値) です。ただし、上位 4 ビットはチップのバージョンにより変更されることがあります。 SDIDH はビット 31~16 に対応します。 SDIDL はビット 15~0 に対応します。

26.4 動作説明

26.4.1 TAP コントローラ

図 26.2 に TAP コントローラの内部状態を示します。JTAG で規定されている状態遷移に対応しています。

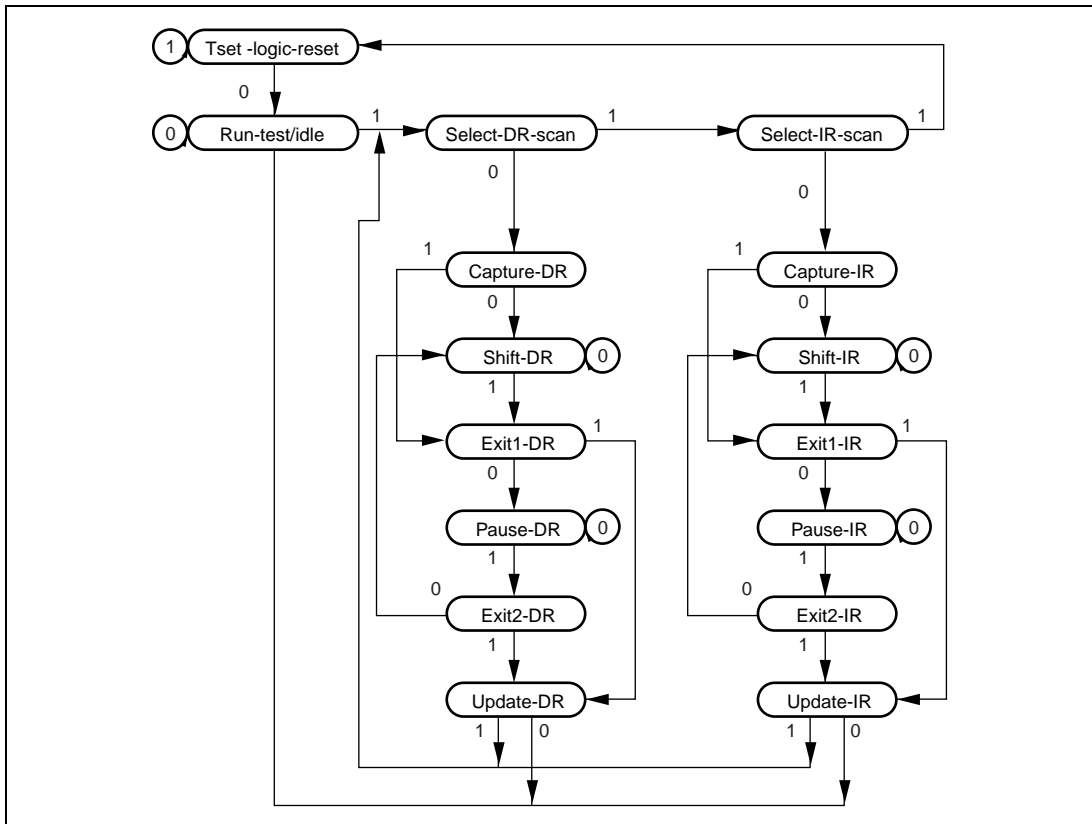


図 26.2 TAP コントローラ状態遷移図

26. ユーザデバッグインタフェース (H-UDI)

26.4.2 リセット構成

表 26.4 に本 LSI のリセット構成を示します。

表 26.4 リセット構成

ASEMD0*1	RESETP	TRST	チップ状態
H	L	L	通常リセットおよび H-UDI リセット
		H	通常リセット
	H	L	H-UDI のみリセット
		H	通常動作
L	L	L	リセットホールド*2
		H	通常リセット
	H	L	H-UDI のみリセット
		H	通常動作

【注】 *1 通常モードと ASE モードの設定を選択

ASEMD0 = H、通常モード

ASEMD0 = L、ASE モード

*2 ASE モード時、リセットホールドは一定サイクル期間、 $\overline{\text{RESETP}}$ と $\overline{\text{TRST}}$ 端子をローレベルにセットすることによって有効になります。この状態で $\overline{\text{RESETP}}$ をハイレベルにセットしても CPU は起動しません。その後 $\overline{\text{TRST}}$ をハイレベルにセットすると、H-UDI 動作が有効になりますが CPU は起動しません。リセットホールド状態は次のようにして解除されます。

別の $\overline{\text{RESETP}}$ アサート (パワーオンリセット)

$\overline{\text{TRST}}$ の再アサート

26.4.3 TDO 出力タイミング

通常モードで TDO から出力するデータの切替タイミングは、TCK の立ち下がりエッジに同期して変化します。これは JTAG 規格のタイミングです。ASE モードでエミュレータ機能を使用しているときは、TDO から出力するデータの切替タイミングは変わります。詳細はエミュレータのユーザーズマニュアルを参照してください。

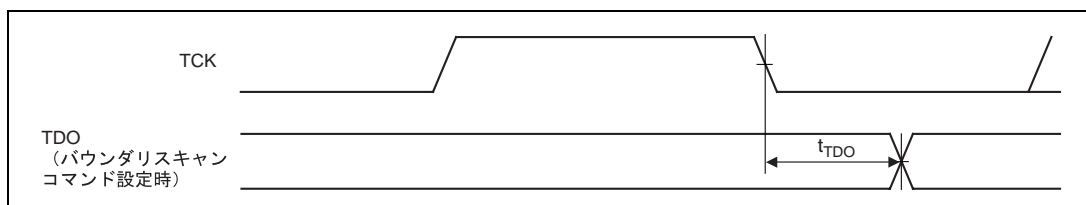


図 26.3 H-UDI データ転送タイミング

26.5 バウンダリスキャン

ASEMD0 がハイの時、H-UDI の各端子は JTAG 用の端子として機能します。本 LSI 上の全てのバウンダリスキャン・セルは、TDI から入り、TDO から出る 1 本のシフトレジスタとして接続され、バウンダリスキャンレジスタ (SDBSR) として構成されています。ボード上でバウンダリスキャン機能をサポートしている全ての LSI の TDO と TDI をバウンダリスキャンレジスタが 1 本のシフトレジスタとしてつながれるように接続します。この方式により、ボードの LSI 間の配線、LSI の実装状態等のテストが可能になります。

26.5.1 サポートする命令

本 LSI では、JTAG 規格で定義される 3 つの必須命令 (BYPASS、SAMPLE/PRELOAD、EXTEST) と、1 つのオプション命令 (IDCODE) をサポートします。

(1) BYPASS

BYPASS 命令は、バイパスレジスタを動作させる必須の標準命令です。この命令はシフトパスを短縮してプリント基板上の他の LSI のシリアルデータを転送高速化するためのものです。この命令の実行中、テスト回路はシステム回路に何も影響を与えません。命令コードは B'111 です。

(2) SAMPLE/PRELOAD

SAMPLE/PRELOAD 命令は本 LSI の内部回路からバウンダリスキャンレジスタに値を入力し、スキャンパスから出力したり、スキャンパスにデータをロードする命令です。本命例実行中本 LSI の入力ピンはそのまま内部回路に伝達され、内部回路の値はそのまま出力ピンから外部へ出力されます。本命令の実行により本 LSI のシステム回路は何の影響も受けません。命令コードは B'001 です。

SAMPLE 動作では、入力ピンから内部回路へ転送される値や内部回路から出力端子へ転送される値のスナッチショットをバウンダリスキャンレジスタに取り込みスキャンパスから読み出します。スナッチショットの取り込みは Capture-DR 状態の TCK の立ち上がりに同期して行われます。スナッチショットの取り込みは本 LSI の通常動作を妨げずに行われます。

PRELOAD 動作では、EXTEST 命令に先立ちスキャンパスからバウンダリスキャンレジスタのパラレル出力ラッチに初期値を設定します。PRELOAD 動作がないと、EXTEST 命令を実行するとき、最初のスキャンシーケンスが完了する (出力ラッチへの転送) までの間出力ピンから不定値が出力される (EXTEST 命令では出力ピンに常にパラレル出力ラッチを出力する) ことになります。

(3) EXTEST

本命令では、本 LSI をプリント基板に実装したとき、外部回路をテストするためのものです。本命令の実行時、出力ピンはバウンダリスキャンレジスタからテストデータ (SAMPLE/PRELOAD 命令ですでに設定されています) をプリント基板へ出力するために使用され、入力ピンはプリント基板からバウンダリスキャンレジスタにテスト結果を取り込むために使用されます。EXTEST 命令を N 回用いてテストを行うとき、N 回目のテストデータは (N-1) 回目のスキャンアウトのときにスキャンインされます。

本命令の Capture-DR 状態で出力ピンのバウンダリスキャンレジスタにロードされたデータは外部回路のテス

26. ユーザデバッグインタフェース (H-UDI)

トには使用されません。(シフト動作で入れ換えます)。

命令コードは B'000 です。

(4) IDCODE

H-UDI 端子から SDIR にコマンドを設定することにより、H-UDI 端子を JTAG で規定されている IDCODE モードに設定できます。H-UDI を初期化した場合($\overline{\text{TRST}}$ のアサート、または TAP を Test-Logic-Reset 状態にする場合)は、IDCODE モードになります。命令コードは B'100 です。

26.5.2 注意事項

1. クロック関連信号 (EXTAL2、XTAL2、EXTAL、XTAL) はバウンダリスキャンの対象外です。
2. リセット関連信号 ($\overline{\text{RESETP}}$) はバウンダリスキャンの対象外です。
3. H-UDI関連信号 (TCK、TDI、TDO、TMS、 $\overline{\text{TRST}}$ 、 $\overline{\text{ASEMD0}}$) はバウンダリスキャンの対象外です。
4. ブルートゥースインタフェース (BT) 関連信号の一部 (RDI_TXTRDATA、RDI_RXBDW_OUT、RDI_REFCLK_IN、RDI_CTRL3、RDI_CTRL4、RFSEL、RCI_SPI_TXRX、RCL_SPI_CLK、RCI_SPI_ENB、RREF) はバウンダリスキャンの対象外です。
5. テスト関連信号の一部 ($\overline{\text{TEST_REG}}$ 、 $\overline{\text{TEST}}$) はバウンダリスキャンの対象外です。
6. アナログ信号 (USB_P、USB_N、DA0、DA1) はバウンダリスキャンの対象外です。
7. EXTESTコマンドを設定時には、 $\overline{\text{RESETP}}$ 端子はローレベル固定にしてください。
8. バウンダリスキャン実行中は、 $\overline{\text{ASEMD0}}$ 端子をハイレベルに固定してください。

26.6 使用上の注意事項

1. H-UDIコマンドは、一度セットされると他のコマンドがH-UDIから再発行されないかぎり変更されません。同じコマンドを連続して与える場合は、チップ動作に影響のないコマンド (BYPASS等) を一度設定してから再度コマンドを設定する必要があります。
2. スタンバイモードではチップ動作が中断されるためH-UDIコマンドは受け付けられません。また、スタンバイモードの前後でTAPの状態を保持するためには、スタンバイモード遷移の際、TCKをハイレベルにしておく必要があります。
3. H-UDIはエミュレータの接続に使用されます。したがって、エミュレータを使用する場合には、H-UDIの機能は使用できません。
4. バウンダリスキャン機能としての $\overline{\text{TRST}}$ 端子はオプション扱いになっています。そのため他のLSIではパワーオンリセット機能を内蔵し、 $\overline{\text{TRST}}$ 端子が無いか、あるいはあってもパワーオンリセット時の初期化は不要のものもありますが、本LSIではパワーオンリセット時には $\overline{\text{TRST}}$ 端子でTAPコントローラも同時に初期化する必要があります。 $\overline{\text{TRST}}$ に印加するパワーオンリセット信号に付いては、以下の様な考慮をしてください。
 - 電源投入時に必ずリセット信号を印加してください。
 - ボードテストの $\overline{\text{TRST}}$ 信号がLSIのシステム側の動作に影響を与えないように、回路を分離してください。ボードテスト中にTAPコントローラのリセットのためボードテストが非同期に $\overline{\text{TRST}}$ 信号を生成する可能性がありますので、回路が分離されていない場合、 $\overline{\text{TRST}}$ 信号がボードテスト中にLSI動作に影響を与える (非同期にシステムリセットが掛かってしまうなど) 可能性があります。
 - 逆に、LSIのシステムリセットが、ボードテストの $\overline{\text{TRST}}$ 信号に影響を与えないように、回路を分離してください。ボードテスト中にボードテストがLSIのシステムリセット信号の接続テストのため、システムリセット信号をバウンダリスキャンから制御しますが、回路が分離されていない場合、この動作でTAPコントローラがリセットされてしまい、正常にテストが行えない可能性があります。

図 26.4 に接続例を示します。

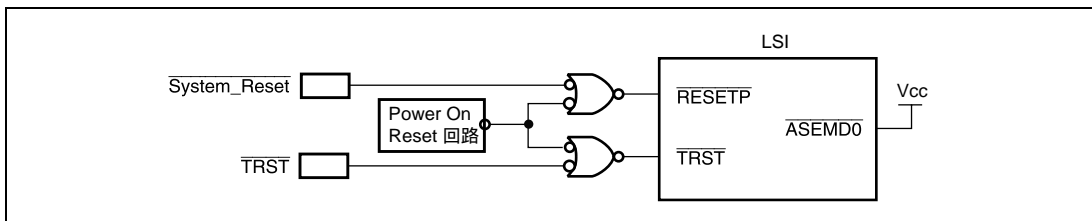


図 26.4 相互干渉しないリセット系信号の設計例

26. ユーザデバッグインタフェース (H-UDI)

5. ASEモード ($\overline{\text{ASEMD0}} = \text{Low}$ に設定)において、 $\overline{\text{ASEBRKAK}}^* = \text{Low}$ 、 $\overline{\text{TRST}} = \text{High}$ でのパワーオンリセットは有効、 $\overline{\text{ASEBRKAK}} = \text{High}$ 、 $\overline{\text{TRST}} = \text{High}$ でのパワーオンリセットは無効となりますので、ご注意ください。詳細を表26.5に記載します。

表 26.5 本 LSI での ASE モードにおけるパワーオンリセット動作

各端子の信号状態	パワーオンリセット信号アサート時の LSI 動作
$\overline{\text{ASEMD0}} = \text{Low}$ $\overline{\text{TRST}} = \text{High}$ $\overline{\text{ASEBRKAK}} = \text{Low}$	パワーオンリセットは有効
$\overline{\text{ASEMD0}} = \text{Low}$ $\overline{\text{TRST}} = \text{High}$ $\overline{\text{ASEBRKAK}} = \text{High}$	パワーオンリセットは無効

なお、本現象は $\overline{\text{ASEMD0}} = \text{Low}$ を設定する場合 (例えばE10A使用時) に発生しますが、 $\overline{\text{ASEMD0}} = \text{High}$ を設定する通常動作の場合は、本現象は発生しません。

- 【注】 * E10A がシステム全体の制御を行っているかを示す信号です。E10A がシステム全体の制御を行っているとき、Low を出力します。

27. レジスタ一覧

本章では、内蔵 I/O レジスタについて、次の形で説明しています。

1. レジスタアドレス一覧（機能モジュールごと、マニュアル章番号順）
 - 機能モジュールごとに、マニュアルの章番号の順に表記しています。
 - 本リストに記載されていないリザーブアドレスは、アクセスしないでください。
 - アドレスは、16ビットまたは32ビットの場合、ビッグエンディアンを前提としてMSB側のアドレスを表記しています。
2. レジスタのビット一覧
 - 「レジスタアドレス一覧（機能モジュールごと、マニュアル章番号順）」の並びに従って、ビットの構成を表記しています。
 - リザーブビットのビット名称部は、「 」で表記しています。
 - ビット名称部が空白のものは、そのレジスタ全体がカウンタやデータに割り付けられていることを表しています。
 - 16ビットまたは32ビットのレジスタの場合は、MSB側のビットから表記しています。
バイトの記載順序は、ビッグエンディアンを前提としています。
3. 各動作モードにおけるレジスタの状態
 - 「レジスタアドレス一覧（機能モジュールごと、マニュアル章番号順）」の並びに従って、レジスタの状態を表記しています。
 - 初期化時の各ビットの状態は、該当する章のレジスタの説明を参照してください。
 - 基本的な動作モード時のレジスタの状態を示しており、内蔵モジュール固有のリセットなどがある場合は、内蔵モジュールの章を参照してください。

27. レジスタ一覧

27.1 レジスタアドレス一覧(機能モジュールごと、マニュアル章番号順)

アクセスサイズは、ビット数を表しています。

【注】 未定義およびリザーブアドレスのアクセスは、禁止します。これらのレジスタをアクセスした時の動作および継続する動作については保証できませんので、アクセスしないようにしてください。

レジスタ名称	略称	ビット数	論理アドレス	モジュール	アクセスサイズ
割り込み事象レジスタ 2	INTEVT2	32	H'A400 0000	例外処理	32
TRAPA 例外レジスタ	TRA	32	H'FFFF FFD0		32
例外事象レジスタ	EXPEVT	32	H'FFFF FFD4		32
例外アドレスレジスタ	TEA	32	H'FFFF FFFC		32
キャッシュ制御レジスタ 1	CCR1	32	H'FFFF FFEC	キャッシュ	32
キャッシュ制御レジスタ 2	CCR2	32	H'A400 00B0		32
割り込み優先レベル設定レジスタ A	IPRA	16	H'A414 FEE2	INTC	16
割り込み優先レベル設定レジスタ B	IPRB	16	H'A414 FEE4		16
割り込み優先レベル設定レジスタ C	IPRC	16	H'A414 0016		16
割り込み優先レベル設定レジスタ D	IPRD	16	H'A414 0018		16
割り込み優先レベル設定レジスタ E	IPRE	16	H'A414 001A		16
割り込み優先レベル設定レジスタ F	IPRF	16	H'A408 0000		16
割り込み優先レベル設定レジスタ G	IPRG	16	H'A408 0002		16
割り込み優先レベル設定レジスタ H	IPRH	16	H'A408 0004		16
割り込みマスクレジスタ 0	IMR0	8	H'A408 0040		8
割り込みマスクレジスタ 1	IMR1	8	H'A408 0042		8
割り込みマスクレジスタ 4	IMR4	8	H'A408 0048		8
割り込みマスクレジスタ 5	IMR5	8	H'A408 004A		8
割り込みマスクレジスタ 6	IMR6	8	H'A408 004C		8
割り込みマスクレジスタ 9	IMR9	8	H'A408 0052		8
割り込みマスククリアレジスタ 0	IMCR0	8	H'A408 0060		8
割り込みマスククリアレジスタ 1	IMCR1	8	H'A408 0062		8
割り込みマスククリアレジスタ 4	IMCR4	8	H'A408 0068		8
割り込みマスククリアレジスタ 5	IMCR5	8	H'A408 006A		8
割り込みマスククリアレジスタ 6	IMCR6	8	H'A408 006C		8
割り込みマスククリアレジスタ 9	IMCR9	8	H'A408 0072		8
割り込みコントロールレジスタ 0	ICR0	16	H'A414 FEE0		16
割り込みコントロールレジスタ 1	ICR1	16	H'A414 0010		16
割り込みコントロールレジスタ 2	ICR2	16	H'A414 0012		16
割り込み要求レジスタ 0	IRR0	8	H'A414 0004		8
共通コントロールレジスタ	CMNCR	32	H'A4FD 0000	BSC	32

27. レジスタ一覧

レジスタ名称	略称	ビット数	論理アドレス	モジュール	アクセスサイズ
CS0 空間バスコントロールレジスタ	CS0BCR	32	H'A4FD 0004	BSC	32
CS3 空間バスコントロールレジスタ	CS3BCR	32	H'A4FD 000C		32
CS4 空間バスコントロールレジスタ	CS4BCR	32	H'A4FD 0010		32
CS0 空間ウェイトコントロールレジスタ	CS0WCR	32	H'A4FD 0024		32
CS3 空間ウェイトコントロールレジスタ	CS3WCR	32	H'A4FD 002C		32
CS4 空間ウェイトコントロールレジスタ	CS4WCR	32	H'A4FD 0030		32
SDRAM コントロールレジスタ	SDCR	32	H'A4FD 0044		32
リフレッシュタイマコントロール/ ステータスレジスタ	RTCSR	32	H'A4FD 0048		32
リフレッシュタイマカウンタ	RTCNT	32	H'A4FD 004C		32
リフレッシュタイマコンスタントレジスタ	RTCOR	32	H'A4FD 0050		32
CS3 空間 SDRAM モードレジスタ	SDMR3		H'A4FD 5xxx*1		16
DMA ソースアドレスレジスタ_0	SAR_0	32	H'A401 0020	DMAC	16 / 32
DMA ディスティネーションアドレス レジスタ_0	DAR_0	32	H'A401 0024		16 / 32
DMA トランスファカウントレジスタ_0	DMATCR_0	32	H'A401 0028		16 / 32
DMA チャンネルコントロールレジスタ_0	CHCR_0	32	H'A401 002C		8 / 16 / 32
DMA イニシャルアドレスレジスタ_0	IAR_0	32	H'A401 0000		16 / 32
DMA ソースアドレスレジスタ_1	SAR_1	32	H'A401 0030		16 / 32
DMA ディスティネーションアドレス レジスタ_1	DAR_1	32	H'A401 0034		16 / 32
DMA トランスファカウントレジスタ_1	DMATCR_1	32	H'A401 0038		16 / 32
DMA チャンネルコントロールレジスタ_1	CHCR_1	32	H'A401 003C		8 / 16 / 32
DMA イニシャルアドレスレジスタ_1	IAR_1	32	H'A401 0004		16 / 32
DMA ソースアドレスレジスタ_2	SAR_2	32	H'A401 0040		16 / 32
DMA ディスティネーションアドレス レジスタ_2	DAR_2	32	H'A401 0044		16 / 32
DMA トランスファカウントレジスタ_2	DMATCR_2	32	H'A401 0048		16 / 32
DMA チャンネルコントロールレジスタ_2	CHCR_2	32	H'A401 004C		8 / 16 / 32
DMA イニシャルアドレスレジスタ_2	IAR_2	32	H'A401 0008		16 / 32
DMA ソースアドレスレジスタ_3	SAR_3	32	H'A401 0050		16 / 32
DMA ディスティネーションアドレス レジスタ_3	DAR_3	32	H'A401 0054		16 / 32
DMA トランスファカウントレジスタ_3	DMATCR_3	32	H'A401 0058		16 / 32
DMA チャンネルコントロールレジスタ_3	CHCR_3	32	H'A401 005C		8 / 16 / 32
DMA イニシャルアドレスレジスタ_3	IAR_3	32	H'A401 000C		16 / 32
DMA オペレーションレジスタ	DMAOR	32	H'A401 0060	8 / 16 / 32	

27. レジスタ一覧

レジスタ名称	略称	ビット数	論理アドレス	モジュール	アクセスサイズ	
DMA 拡張リソースセクタ 0	DMARS0	16	H'A409 0000	DMAC	16	
DMA 拡張リソースセクタ 1	DMARS1	16	H'A409 0004		16	
周波数制御レジスタ* ³	FRQCR	16	H'A415 FF80	CPG	16	
ウォッチドッグタイマカウンタ	WTCNT	8	H'A415 FF84	WDT	8 / 16 * ²	
ウォッチドッグタイマコントロール/ ステータスレジスタ	WTCSR	8	H'A415 FF86		8 / 16 * ²	
スタンバイコントロールレジスタ* ³	STBCR	8	H'A415 FF82	低消費電力	8	
スタンバイコントロールレジスタ 2	STBCR2	8	H'A415 FF88		8	
スタンバイコントロールレジスタ 3	STBCR3	8	H'A40A 0000		8	
スタンバイコントロールレジスタ 4	STBCR4	8	H'A40A 0004		8	
タイマスタートレジスタ	TSTR	8	H'A412 FE92	TMU	8	
タイマコンスタントレジスタ_0	TCOR_0	32	H'A412 FE94		32	
タイマカウンタ_0	TCNT_0	32	H'A412 FE98		32	
タイマコントロールレジスタ_0	TCR_0	16	H'A412 FE9C		16	
タイマコンスタントレジスタ_1	TCOR_1	32	H'A412 FEA0		32	
タイマカウンタ_1	TCNT_1	32	H'A412 FEA4		32	
タイマコントロールレジスタ_1	TCR_1	16	H'A412 FEA8		16	
タイマコンスタントレジスタ_2	TCOR_2	32	H'A412 FEAC		32	
タイマカウンタ_2	TCNT_2	32	H'A412 FEB0		32	
タイマコントロールレジスタ_2	TCR_2	16	H'A412 FEB4		16	
モードレジスタ	SIMDR	16	H'A442 0000		SIOF	16
クロックセレクトレジスタ	SISCR	16	H'A442 0002			16
送信データアサインレジスタ	SITDAR	16	H'A442 0004	16		
受信データアサインレジスタ	SIRDAR	16	H'A442 0006	16		
制御データアサインレジスタ	SICDAR	16	H'A442 0008	16		
コントロールレジスタ	SICTR	16	H'A442 000C	16		
FIFO コントロールレジスタ	SIFCTR	16	H'A442 0010	16		
ステータスレジスタ	SISTR	16	H'A442 0014	16		
割り込み許可レジスタ	SIER	16	H'A442 0016	16		
送信データレジスタ	SITDR	32	H'A442 0020	32		
受信データレジスタ	SIRDR	32	H'A442 0024	32		
送信制御データレジスタ	SITCR	32	H'A442 0028	32		
受信制御データレジスタ	SIRCR	32	H'A442 002C	32		
SPI 制御レジスタ	SPICR	16	H'A4420030	16		
シリアルモードレジスタ_0	SCSMR_0	16	H'A443 0000	SCIF_0		16
ビットレートレジスタ_0	SCBRR_0	8	H'A443 0004	(チャンネル 0)		8

27. レジスタ一覧

レジスタ名称	略称	ビット数	論理アドレス	モジュール	アクセスサイズ
シリアルコントロールレジスタ_0	SCSCR_0	16	H'A443 0008	SCIF_0 (チャンネル0)	16
トランスミットデータストップレジスタ_0	SCTDSR_0	8	H'A443 000C		8
FIFO エラー数レジスタ_0	SCFER_0	16	H'A443 0010		16
シリアルステータスレジスタ_0	SCSSR_0	16	H'A443 0014		16
FIFO コントロールレジスタ_0	SCFCR_0	16	H'A443 0018		16
FIFO データ数レジスタ_0	SCFDR_0	16	H'A443 001C		16
トランスミット FIFO データレジスタ_0	SCFTDR_0	8	H'A443 0020		8
レシーブ FIFO データレジスタ_0	SCFRDR_0	8	H'A443 0024		8
シリアルモードレジスタ_1	SCSMR_1	16	H'A445 0000	SCIF_1 (チャンネル1)	16
ビットレートレジスタ_1	SCBRR_1	8	H'A445 0004		8
シリアルコントロールレジスタ_1	SCSCR_1	16	H'A445 0008		16
トランスミットデータストップレジスタ_1	SCTDSR_1	8	H'A445 000C		8
FIFO エラー数レジスタ_1	SCFER_1	16	H'A445 0010		16
シリアルステータスレジスタ_1	SCSSR_1	16	H'A445 0014		16
FIFO コントロールレジスタ_1	SCFCR_1	16	H'A445 0018		16
FIFO データ数レジスタ_1	SCFDR_1	16	H'A445 001C		16
トランスミット FIFO データレジスタ_1	SCFTDR_1	8	H'A445 0020	8	
レシーブ FIFO データレジスタ_1	SCFRDR_1	8	H'A445 0024	8	
EXCPG 制御レジスタ	EXCPGCR	8	H'A447 0000	USBPM	8
HcRevision レジスタ	HREVR	32	H'A449 0000	USBH	32
HcControl レジスタ	HCTLR	32	H'A449 0004		32
HcCommandStatus レジスタ	HCSR	32	H'A449 0008		32
HcInterruptStatus レジスタ	HISR	32	H'A449 000C		32
HcInterruptEnable レジスタ	HIER	32	H'A449 0010		32
HcInterruptDisable レジスタ	HIDR	32	H'A449 0014		32
HcHCCA レジスタ	HHCCAR	32	H'A449 0018		32
HcPeriodCurrentED レジスタ	HPCEDR	32	H'A449 001C		32
HcControlHeadED レジスタ	HCHEDR	32	H'A449 0020		32
HcControlCurrentED レジスタ	HCCEDR	32	H'A449 0024		32
HcBulkHeadED レジスタ	HBHEDR	32	H'A449 0028		32
HcBulkCurrentED レジスタ	HBCEDR	32	H'A449 002C		32
HcDoneHeadED レジスタ	HDHEDR	32	H'A449 0030		32
HcFmInterval レジスタ	HFIR	32	H'A449 0034		32
HcFmRemaining レジスタ	HFRR	32	H'A449 0038		32
HcFmNumber レジスタ	HFNR	32	H'A449 003C		32
HcPeriodicStart レジスタ	HPSR	32	H'A449 0040	32	
HcLSThreshold レジスタ	HLSTR	32	H'A449 0044	32	

27. レジスタ一覧

レジスタ名称	略称	ビット数	論理アドレス	モジュール	アクセスサイズ	
HcRhDescriptorA レジスタ	HRDRA	32	H'A449 0048	USBH	32	
HcRhDescriptorB レジスタ	HRDRB	32	H'A449 004C		32	
HcRhStatus レジスタ	HRSR	32	H'A449 0050		32	
HcRhPortStatus1 レジスタ	HRPSR	32	H'A449 0054		32	
割り込みフラグレジスタ 0	IFR0	32	H'A448 0000	USBF	32	
割り込み選択レジスタ 0	ISR0	32	H'A448 0010		32	
割り込みイネーブルレジスタ 0	IER0	32	H'A448 0020		32	
EP0i データレジスタ	EPDR0i	64B*4	H'A448 0030		8	
EP0o データレジスタ	EPDR0o	64B*4	H'A448 0034		8	
EP0s データレジスタ	EPDR0s	8B*4	H'A448 0038		8	
EP1 データレジスタ	EPDR1	16B*4	H'A448 003C		8	
EP2i データレジスタ	EPDR2i	128B*4	H'A448 0040		8	
EP2o データレジスタ	EPDR2o	128B*4	H'A448 0044		8	
EP3i データレジスタ	EPDR3i	128B*4	H'A448 004C		8	
EP3o データレジスタ	EPDR3o	120B*4	H'A448 0050		8	
EP4 データレジスタ	EPDR4	16B*4	H'A448 0054		8	
EP5 データレジスタ	EPDR5	64B*4	H'A448 0058		8	
EP6 データレジスタ	EPDR6	64B*4	H'A448 005C		8	
EP0o 受信データサイズレジスタ	EPSZ0o	8	H'A448 0080		8	
EP2o 受信データサイズレジスタ	EPSZ2o	8	H'A448 0084		8	
EP3o 受信データサイズレジスタ	EPSZ3o	8	H'A448 0088		8	
EP6 受信データサイズレジスタ	EPSZ6	8	H'A448 008C		8	
トリガレジスタ	TRG	16	H'A448 00A0		16	
データステータスレジスタ	DASTS	8	H'A448 00A4		8	
FIFO クリアレジスタ	FCLR	16	H'A448 00A8		16	
DMA 転送設定レジスタ	DMA	8	H'A448 00AC		8	
エンドポイントストールレジスタ	EPSTL	16	H'A448 00B0		32*5	
コンフィグレーションバリュールレジスタ	CVR	8	H'A448 00B4		8	
タイムスタンプレジスタ	TSR	16	H'A448 00B8		16	
コントロールレジスタ	CTLR	8	H'A448 00BC		8	
エンドポイント情報レジスタ	EPIR	8	H'A448 00C0		8	
D/A データレジスタ 0	DADR_0	8	H'A440 0000		DAC	8
D/A データレジスタ 1	DADR_1	8	H'A440 0001			8
D/A コントロールレジスタ	DACR	8	H'A440 0002			8
ポート A データレジスタ	PADR	8	H'A405 0120		I/O ポート	8
ポート B データレジスタ	PBDR	8	H'A405 0122			8
ポート D データレジスタ	PDDR	8	H'A405 0126	8		

レジスタ名称	略称	ビット数	論理アドレス	モジュール	アクセスサイズ
ポートEデータレジスタ	PEDR	8	H'A405 0128	I/Oポート	8
ポートGデータレジスタ	PGDR	8	H'A405 012C		8
ポートHデータレジスタ	PHDR	8	H'A405 012E		8
ポートAコントロールレジスタ	PACR	16	H'A405 0100	PFC	16
ポートBコントロールレジスタ	PBCR	16	H'A405 0102		16
ポートDコントロールレジスタ	PDCR	16	H'A405 0106		16
ポートEコントロールレジスタ	PECR	16	H'A405 0108		16
ポートGコントロールレジスタ	PGCR	16	H'A405 010C		16
ポートHコントロールレジスタ	PHCR	16	H'A405 010E		16
ピンセレクトレジスタA	PSELA	16	H'A405 0140		16
I/OバッファHi-Z制御レジスタA	HIZCRA	16	H'A405 0146		16
ノイズキャンセラ制御レジスタ	NCCR	16	H'A405 014C		16
ブ레이크データレジスタB	BDRB	32	H'FFFF FF90		UBC
ブ레이크データマスクレジスタB	BDMRB	32	H'FFFF FF94	32	
ブ레이크コントロールレジスタ	BRCR	32	H'FFFF FF98	32	
実行回数ブ레이크レジスタ	BETR	16	H'FFFF FF9C	16	
ブ레이크アドレスレジスタB	BARB	32	H'FFFF FFA0	32	
ブ레이크アドレスマスクレジスタB	BAMRB	32	H'FFFF FFA4	32	
ブ레이크バスサイクルレジスタB	BBRB	16	H'FFFF FFA8	16	
ブランチソースレジスタ	BRSR	32	H'FFFF FFAC	32	
ブ레이크アドレスレジスタA	BARA	32	H'FFFF FFB0	32	
ブ레이크アドレスマスクレジスタA	BAMRA	32	H'FFFF FFB4	32	
ブ레이크バスサイクルレジスタA	BBRA	16	H'FFFF FFB8	16	
ブランチデスティネーションレジスタ	BRDR	32	H'FFFF FFBC	32	
インストラクションレジスタ*	SDIR	16	H'A410 0200	H-UDI	
IDレジスタ	SDID/SDIDH	16	H'A410 0214		16
IDレジスタ	SDID/SDIDL	16	H'A410 0216		16

【注】 *1 SDRAMモードレジスタへのアクセス制御で、xxxは設定値に依存します。

*2 読み出しは8ビット、書き込みは16ビットです。

*3 本レジスタへの書き込み直後に SLEEP 命令を発行すると、正しくレジスタの内容が反映されない可能性があります。必ずレジスタの読み出しをしてから SLEEP 命令を発行するようにしてください。

*4 USBFモジュールの各エンドポイントデータレジスタのビット数は、FIFOバッファの最大値を表しています。

*5 エンドポイントストールレジスタ (EPSTL) の読み出し、書き込みは32ビットサイズでアクセスしてください。この場合、下位2バイトが有効データの位置になります。

*6 H-UDIインタフェースからのアクセスの場合は、3ビットのレジスタです。

27. レジスタ一覧

27.2 レジスタビット一覧

内蔵モジュールのレジスタのアドレスとビット名を以下に示します。

16 ビットおよび 32 ビットレジスタは、8 ビットずつ 2 段または 4 段で表しています。

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
INTEVT2									例外処理
					INTEVT2	INTEVT2	INTEVT2	INTEVT2	
	INTEVT2	INTEVT2	INTEVT2	INTEVT2	INTEVT2	INTEVT2	INTEVT2	INTEVT2	
TRA									
							TRA	TRA	
	TRA	TRA	TRA	TRA	TRA	TRA			
EXPEVT									
					EXPEVT	EXPEVT	EXPEVT	EXPEVT	
	EXPEVT	EXPEVT	EXPEVT	EXPEVT	EXPEVT	EXPEVT	EXPEVT	EXPEVT	
TEA	TEA	TEA	TEA	TEA	TEA	TEA	TEA	TEA	
	TEA	TEA	TEA	TEA	TEA	TEA	TEA	TEA	
	TEA	TEA	TEA	TEA	TEA	TEA	TEA	TEA	
	TEA	TEA	TEA	TEA	TEA	TEA	TEA	TEA	
CCR1									キャッシュ
					CF	CB	WT	CE	
							W3LOAD	W3LOCK	
CCR2									
							W2LOAD	W2LOCK	
IPRA	IPR15	IPR14	IPR13	IPR12	IPR11	IPR10	IPR9	IPR8	INTC
	IPR7	IPR6	IPR5	IPR4					
IPRB	IPR15	IPR14	IPR13	IPR12					
IPRC	IPR15	IPR14	IPR13	IPR12	IPR11	IPR10	IPR9	IPR8	
	IPR7	IPR6	IPR5	IPR4	IPR3	IPR2	IPR1	IPR0	

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
IPRD	IPR15	IPR14	IPR13	IPR12					INTC
					IPR3	IPR2	IPR1	IPR0	
IPRE	IPR15	IPR14	IPR13	IPR12					
IPRF									
	IPR7	IPR6	IPR5	IPR4	IPR3	IPR2	IPR1	IPR0	
IPRG	IPR15	IPR14	IPR13	IPR12	IPR11	IPR10	IPR9	IPR8	
IPRH					IPR11	IPR10	IPR9	IPR8	
IMR0	IM7			IM4	IM3	IM2	IM1	IM0	
IMR1					IM3	IM2	IM1	IM0	
IMR4		IM6	IM5	IM4	IM3				
IMR5							IM1	IM0	
IMR6							IM1		
IMR9						IM2	IM1	IM0	
IMCR0	IMC7			IMC4	IMC3	IMC2	IMC1	IMC0	
IMCR1					IMC3	IMC2	IMC1	IMC0	
IMCR4		IMC6	IMC5	IMC4	IMC3				
IMCR5							IMC1	IMC0	
IMCR6							IMC1		
IMCR9						IMC2	IMC1	IMC0	
ICR0	NMIL							NMIE	
ICR1		IRQE					IRQ41S	IRQ40S	
	IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S	
ICR2									
					IRQ71S	IRQ70S			
IRR0	IRQ7R			IRQ4R	IRQ3R	IRQ2R	IRQ1R	IRQ0R	
CMNCR									BSC
	DMAIW1	DMAIW0	DMAIWA		ENDIAN		HIZMEM	HIZCNT	

27. レジスタ一覧

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
CS0BCR			IWW1	IWW0		IWRWD1	IWRWD0		BSC
	IWRWS1	IWRWS0		IWRRD1	IWRRD0		IWRRS1	IWRRS0	
		TYPE2	TYPE1	TYPE0		BSZ1	BSZ0		
CS3BCR			IWW1	IWW0		IWRWD1	IWRWD0		
	IWRWS1	IWRWS0		IWRRD1	IWRRD0		IWRRS1	IWRRS0	
		TYPE2	TYPE1	TYPE0		BSZ1	BSZ0		
CS4BCR			IWW1	IWW0		IWRWD1	IWRWD0		
	IWRWS1	IWRWS0		IWRRD1	IWRRD0		IWRRS1	IWRRS0	
		TYPE2	TYPE1	TYPE0		BSZ1	BSZ0		
CS0WCR (バースト ROM 以外)									
				SW1	SW0	WR3	WR2	WR1	
	WR0	WM					HW1	HW0	
CS0WCR (バースト ROM 使用時)							BW1	BW0	
						W3	W2	W1	
	W0	WM							
CS3WCR (SDRAM 以外)				BAS					
						WR3	WR2	WR1	
	WR0	WM							
							BW1	BW0	
CS3WCR (SDRAM 使用時)									
		TRP1	TRP0		TRCD1	TRCD0		A3CL1	
	A3CL0			TRWL1	TRWL0		TRC1	TRC0	
CS4WCR (バースト ROM 以外)				BAS		WW2	WW1	WW0	
				SW1	SW0	WR3	WR2	WR1	
	WR0	WM					HW1	HW0	

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
CS4WCR (バースト ROM 使用時)									BSC
							BW1	BW0	
				SW1	SW0	W3	W2	W1	
	W0	WM					HW1	HW0	
SDCR									
				SLOW	RFSH	RMODE	PDOWN	BACTV	
				A3ROW1	A3ROW0		A3COL1	A3COL0	
RTCSR									
	CMF		CKS2	CKS1	CKS0	RRC2	RRC1	RRC0	
RTCNT									
RTCOR									
SDMR3									
SAR_0									DMAC
DAR_0									
DMATCR_0									

27. レジスタ一覧

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール		
CHCR_0						RPT	RAS	DA	DMAC		
	DO	TL					AM	AL			
	DM1	DM0	SM1	SM0	RS3	RS2	RS1	RS0			
	DL	DS	TB	TS1	TS0	IE	TE	DE			
IAR_0										DMAC	
SAR_1											DMAC
DAR_1									DMAC		
DMATCR_1										DMAC	
CHCR_1						RPT	RAS	DA			DMAC
	DO	TL					AM	AL			
	DM1	DM0	SM1	SM0	RS3	RS2	RS1	RS0			
	DL	DS	TB	TS1	TS0	IE	TE	DE			
IAR_1									DMAC		
SAR_2										DMAC	

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
DAR_2									DMAC
DMATCR_2									
CHCR_2						RPT	RAS	DA	
	DO	TL					AM	AL	
	DM1	DM0	SM1	SM0	RS3	RS2	RS1	RS0	
	DL	DS	TB	TS1	TS0	IE	TE	DE	
IAR_2									
SAR_3									
DAR_3									
DMATCR_3									
CHCR_3						RPT	RAS	DA	
	DO	TL					AM	AL	
	DM1	DM0	SM1	SM0	RS3	RS2	RS1	RS0	
	DL	DS	TB	TS1	TS0	IE	TE	DE	

27. レジスタ一覧

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
IAR_3									DMAC
DMAOR			CMS1	CMS0			PR1	PR0	DMAC
						AE	NMIF	DME	
DMARS0	C1MID5	C1MID4	C1MID3	C1MID2	C1MID1	C1MID0	C1RID1	C1RID0	DMAC
	C0MID5	C0MID4	C0MID3	C0MID2	C0MID1	C0MID0	C0RID1	C0RID0	
DMARS1	C3MID5	C3MID4	C3MID3	C3MID2	C3MID1	C3MID0	C3RID1	C3RID0	
	C2MID5	C2MID4	C2MID3	C2MID2	C2MID1	C2MID0	C2RID1	C2RID0	
FRQCR				CKOEN		STC2	STC1	STC0	CPG
		IFC2	IFC1	IFC0		PFC2	PFC1	PFC0	
WTCNT									WDT
WTCSR	TME	WT/IT	RSTS	WOVF	IOVF	CKS2	CKS1	CKS0	
STBCR	STBY			STBXTL		MSTP2			低消費 電力
STBCR2	MSTP10	MSTP9	MSTP8	MSTP7		MSTP5	MSTP4	MSTP3	
STBCR3	MSTP37								
STBCR4		MSTP46			MSTP43	MSTP42	MSTP41	MSTP40	
TSTR						STR2	STR1	STR0	TMU
TCOR_0									
TCNT_0									TMU
TCR_0								UNF	TMU
			UNIE				TPSC1	TPSC0	

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
TCOR_1									TMU
TCNT_1									
TCR_1								UNF	
			UNIE				TPSC1	TPSC0	
TCOR_2									
TCNT_2									
TCR_2								UNF	
			UNIE				TPSC1	TPSC0	
SIMDR	TRMD1	TRMD0	SYNCAT	REDG	FL3	FL2	FL1	FL0	SIOF
	TXDIZ	RCIM	SYNCAC	SYNCDL					
SISCR	MSEL	MSIMM		BRPS4	BRPS3	BRPS2	BRPS1	BRPS0	
						BRDV2	BRDV1	BRDV0	
SITDAR	TDLE				TDLA3	TDLA2	TDLA1	TDLA0	
	TDRE	TLREP			TDRA3	TDRA2	TDRA1	TDRA0	
SIRDAR	RDLE				RDLA3	RDLA2	RDLA1	RDLA0	
	RDRE				RDRA3	RDRA2	RDRA1	RDRA0	
SICDAR	CD0E				CD0A3	CD0A2	CD0A1	CD0A0	
	CD1E				CD1A3	CD1A2	CD1A1	CD1A0	
SICTR	SCKE	FSE					TXE	RXE	
							TXRST	RXRST	
SIFCTR	TFWM2	TFWM1	TFWM0	TFUA4	TFUA3	TFUA2	TFUA1	TFUA0	
	RFWM2	RFWM1	RFWM0	RFUA4	RFUA3	RFUA2	RFUA1	RFUA0	

27. レジスタ一覧

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール	
SISTR		TCRDY	TFEMP	TDREQ		RCRDY	RFFUL	RDREQ	SIOF	
			SAERR	FSERR	TFOVF	TFUDF	RFUDF	RFOVF		
SIER	TDMAE	TCRDYE	TFEMPE	TDREQE	RDMAE	RCRDYE	RFFULE	RDREQE		
			SAERRE	FSERRE	TFOVFE	TFUDFE	RFUDFE	RFOVFE		
SITDR	SITDL15	SITDL14	SITDL13	SITDL12	SITDL11	SITDL10	SITDL9	SITDL8		
	SITDL7	SITDL6	SITDL5	SITDL4	SITDL3	SITDL2	SITDL1	SITDL0		
	SITDR15	SITDR14	SITDR13	SITDR12	SITDR11	SITDR10	SITDR9	SITDR8		
	SITDR7	SITDR6	SITDR5	SITDR4	SITDR3	SITDR2	SITDR1	SITDR0		
SIRDR	SIRDL15	SIRDL14	SIRDL13	SIRDL12	SIRDL11	SIRDL10	SIRDL9	SIRDL8		
	SIRDL7	SIRDL6	SIRDL5	SIRDL4	SIRDL3	SIRDL2	SIRDL1	SIRDL0		
	SIRDR15	SIRDR14	SIRDR13	SIRDR12	SIRDR11	SIRDR10	SIRDR9	SIRDR8		
	SIRDR7	SIRDR6	SIRDR5	SIRDR4	SIRDR3	SIRDR2	SIRDR1	SIRDR0		
SITCR	SITC015	SITC014	SITC013	SITC012	SITC011	SITC010	SITC09	SITC08		
	SITC07	SITC06	SITC05	SITC04	SITC03	SITC02	SITC01	SITC00		
	SITC115	SITC114	SITC113	SITC112	SITC111	SITC110	SITC19	SITC18		
	SITC17	SITC16	SITC15	SITC14	SITC13	SITC12	SITC11	SITC10		
SIRCR	SIRC015	SIRC014	SIRC013	SIRC012	SIRC011	SIRC010	SIRC09	SIRC08		
	SIRC07	SIRC06	SIRC05	SIRC04	SIRC03	SIRC02	SIRC01	SIRC00		
	SIRC115	SIRC114	SIRC113	SIRC112	SIRC111	SIRC110	SIRC19	SIRC18		
	SIRC17	SIRC16	SIRC15	SIRC14	SIRC13	SIRC12	SIRC11	SIRC10		
SPICR	SPIM		CPHA	CPOL		SS2E	SS1E	SS0E		
			SSAST1	SSAST0			FLD1	FLD0		
SCSMR_0						SRC2	SRC1	SRC0	SCIF_0	
	C/A	CHR	PE	O/E	STOP		CKS1	CKS0		
SCBRR_0	SCBRD7	SCBRD6	SCBRD5	SCBRD4	SCBRD3	SCBRD2	SCBRD1	SCBRD0		
SCSCR_0	TDRQE	RDRQE			TSIE	ERIE	BRIE	DRIE		
	TIE	RIE	TE	RE			CKE1	CKE0		
SCTDSR_0										
SCFER_0			PER5	PER4	PER3	PER2	PER1	PER0		
			FER5	FER4	FER3	FER2	FER1	FER0		
SCSSR_0							ORER	TSF		
	ER	TEND	TDFE	BRK	FER	PER	RDF	DR		
SCFCR_0	TSE	TCRST				RSTRG2	RSTRG1	RSTRG0		
	RTRG1	RTRG0	TTRG1	TTRG0	MCE	TFRST	RFRST	LOOP		

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
SCFDR_0		T6	T5	T4	T3	T2	T1	T0	SCIF_0
		R6	R5	R4	R3	R2	R1	R0	
SCFTDR_0	SCFTD7	SCFTD6	SCFTD5	SCFTD4	SCFTD3	SCFTD2	SCFTD1	SCFTD0	
SCFRDR_0	SCFRD7	SCFRD6	SCFRD5	SCFRD4	SCFRD3	SCFRD2	SCFRD1	SCFRD0	
SCSMR_1						SRC2	SRC1	SRC0	SCIF_1
	C/A	CHR	PE	O/E	STOP		CKS1	CKS0	
SCBRR_1	SCBRD7	SCBRD6	SCBRD5	SCBRD4	SCBRD3	SCBRD2	SCBRD1	SCBRD0	
SCSCR_1	TDRQE	RDRQE			TSIE	ERIE	BRIE	DRIE	
	TIE	RIE	TE	RE			CKE1	CKE0	
SCTDSR_1									
SCFER_1			PER5	PER4	PER3	PER2	PER1	PER0	
			FER5	FER4	FER3	FER2	FER1	FER0	
SCSSR_1							ORER	TSF	
	ER	TEND	TDFE	BRK	FER	PER	RDF	DR	
SCFCR_1	TSE	TCRST				RSTRG2	RSTRG1	RSTRG0	
	RTRG1	RTRG0	TTRG1	TTRG0	MCE	TFRST	RFRST	LOOP	
SCFDR_1		T6	T5	T4	T3	T2	T1	T0	
		R6	R5	R4	R3	R2	R1	R0	
SCFTDR_1	SCFTD7	SCFTD6	SCFTD5	SCFTD4	SCFTD3	SCFTD2	SCFTD1	SCFTD0	
SCFRDR_1	SCFRD7	SCFRD6	SCFRD5	SCFRD4	SCFRD3	SCFRD2	SCFRD1	SCFRD0	
EXCPGCR				USBVALID	USBRESET	USBHSTP	USBFSTP	USBCLKSEL	USBPM
HREVR									USBH
HCTLR								IR	
	HCFS1	HCFS0	BLE	CLE	IE	PLE	CBSR1	CBSR0	
HCSR							SOC1	SOC0	
					OCR	BLF	CLF	HCR	

27. レジスタ一覧

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
HISR		OC							USBH
		RHSC	FNO	UE	RD	SF	WDH	SO	
HIER	MIE	OC							
		RHSC	FNO	UE	RD	SF	WDH	SO	
HIDR	MIE	OC							
		RHSC	FNO	UE	RD	SF	WDH	SO	
HHCCAR									
HPCEDR									
HCHEDR									
HCCEDR									
HBHEDR									

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
HBCEDR									USBH
HDHEDR									
HFIR	FIT	FSMPS14	FSMPS13	FSMPS12	FSMPS11	FSMPS10	FSMPS9	FSMPS8	
	FSMPS7	FSMPS6	FSMPS5	FSMPS4	FSMPS3	FSMPS2	FSMPS1	FSMPS0	
			FI13	FI12	FI11	FI10	FI9	FI8	
	FI7	FI6	FI5	FI4	FI3	FI2	FI1	FI0	
HFRR	FRT								
			FR13	FR12	FR11	FR10	FR9	FR8	
	FR7	FR6	FR5	FR4	FR3	FR2	FR1	FR0	
HFNR									
	FN15	FN14	FN13	FN12	FN11	FN10	FN9	FN8	
	FN7	FN6	FN5	FN4	FN3	FN2	FN1	FN0	
HPSR									
			PS13	PS12	PS11	PS10	PS9	PS8	
	PS7	PS6	PS5	PS4	PS3	PS2	PS1	PS0	
HLSTR									
					LST11	LST10	LST9	LST8	
	LST7	LST6	LST5	LST4	LST3	LST2	LST1	LST0	
HRDRA	POTPGT7	POTPGT6	POTPGT5	POTPGT4	POTPGT3	POTPGT2	POTPGT1	POTPGT0	
				NOCP	OCPM	DT	NPS	PSM	
	NDP7	NDP6	NDP5	NDP4	NDP3	NDP2	NDP1	NDP0	

27. レジスタ一覧

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
HRDRB									USBF
							PPCM		
							DR		
HRSR	CRWE							OCIC LPSC	
	DRWE							OCI LPS	
HRPSR				PRSC	OCIC	PSSC	PESC	CSC	
							LSDA	PPS	
				PRS	POCI	PSS	PES	CCS	
IFR0		BSRT	SETUP TS	VBUSMN	VBUSF	SURSS	SURES	CFDN	
	SOF	SETC	SETI	EP6 FULL	EP5 EMPTY	EP5 TR	EP4 TR	EP4 TS	
			EP3o TF	EP3o TS	EP3I TR	EP3I TS			
	EP2o FULL	EP2i EMPTY	EP2I TR	EP1 TR	EP1 TS	EP0o TS	EP0I TR	EP0I TS	
ISR0		BSRT IS	SETUP TS IS		VBUSF IS		SURES IS	CFDN IS	
	SOF IS	SETC IS	SETI IS	EP6 FULL IS	EP5 EMPTY IS	EP5TR IS	EP4TR IS	EP4TS IS	
			EP3o TF IS	EP3o TS IS	EP3I TR IS	EP3I TS IS			
	EP2o FULL IS	EP2I EMPTY IS	EP2I TR IS	EP1 TR IS	EP1 TS IS	EP0o TS IS	EP0I TR IS	EP0I TS IS	
IER0		BSRT IE	SETUP TS IE		VBUSF IE		SURES IE	CFDN IE	
	SOF IE	SETC IE	SETI IE	EP6 FULL IE	EP5 EMPTY IE	EP5 TR IE	EP4 TR IE	EP4 TS IE	
			EP3o TF IE	EP3o TS IE	EP3I TR IE	EP3I TS IE			
	EP2o FULL IE	EP2I EMPTY IE	EP2I TR IE	EP1 TR IE	EP1 TS IE	EP0o TS IE	EP0i TR IE	EP0i TS IE	
EPDR0i	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR0o	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR0s	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR1	D7	D6	D5	D4	D3	D2	D1	D0	

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
EPDR2i	D7	D6	D5	D4	D3	D2	D1	D0	USBF
EPDR2o	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR3i	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR3o	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR4	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR5	D7	D6	D5	D4	D3	D2	D1	D0	
EPDR6	D7	D6	D5	D4	D3	D2	D1	D0	
EPSZ0o	D7	D6	D5	D4	D3	D2	D1	D0	
EPSZ2o	D7	D6	D5	D4	D3	D2	D1	D0	
EPSZ3o	D7	D6	D5	D4	D3	D2	D1	D0	
EPSZ6	D7	D6	D5	D4	D3	D2	D1	D0	
TRG		EP6 RDFN	EP5 PKTE	EP4 PKTE					
		EP2o RDFN	EP2i PKTE	EP1 PKTE		EP0s RDFN	EP0o RDFN	EP0i PKTE	
DASTS		EP5 DE	EP4 DE		EP2i DE	EP1 DE		EP0i DE	
FCLR		EP6 CLR	EP5 CLR	EP4 CLR		EP3o CLR	EP3i CLR		
		EP2o CLR	EP2i CLR	EP1 CLR			EP0o CLR	EP0i CLR	
DMA	EP6 DMAE	EP6 DMAS	EP5 DMAE	EP5 DMAS	EP2o DMAE	EP2o DMAS	EP2i DMAE	EP2i DMAS	
EPSTL						EP6STL	EP5STL	EP4STL	
		EP3o STL	EP3i STL		EP2o STL	EP2i STL	EP1 STL	EP0 STL	
CVR	CNFV	INTV3	INTV2	INTV1	INTV0	ALTV2	ALTV1	ALTV0	
TSR						D10	D9	D8	
	D7	D6	D5	D4	D3	D2	D1	D0	
CTRL			PULLPE	RWUPS	RSME	PWMD	ASCE	SOFME	
EPIR	D7	D6	D5	D4	D3	D2	D1	D0	
DADR_0									DAC
DADR_1									
DACR	DAOE1	DAOE0							
PADR		PA6DT	PA5DT	PA4DT	PA3DT	PA2DT	PA1DT	PA0DT	I/O ポート
PBDR	PB7DT	PB6DT	PB5DT	PB4DT	PB3DT	PB2DT	PB1DT	PB0DT	
PDDR			PD5DT	PD4DT	PD3DT	PD2DT	PD1DT	PD0DT	
PEDR		PE6DT	PE5DT	PE4DT	PE3DT	PE2DT	PE1DT	PE0DT	
PGDR				PG4DT	PG3DT	PG2DT	PG1DT	PG0DT	

27. レジスタ一覧

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
PHDR				PH4DT	PH3DT	PH2DT	PH1DT	PH0DT	I/O ポート
PACR			PA6MD1	PA6MD0	PA5MD1	PA5MD0	PA4MD1	PA4MD0	PFC
	PA3MD1	PA3MD0	PA2MD1	PA2MD0	PA1MD1	PA1MD0	PA0MD1	PA0MD0	
PBCR	PB7MD1	PB7MD0	PB6MD1	PB6MD0	PB5MD1	PB5MD0	PB4MD1	PB4MD0	
	PB3MD1	PB3MD0	PB2MD1	PB2MD0	PB1MD1	PB1MD0	PB0MD1	PB0MD0	
PDCR					PD5MD1	PD5MD0	PD4MD1	PD4MD0	
	PD3MD1	PD3MD0	PD2MD1	PD2MD0	PD1MD1	PD1MD0	PD0MD1	PD0MD0	
PECR			PE6MD1	PE6MD0	PE5MD1	PE5MD0	PE4MD1	PE4MD0	
	PE3MD1	PE3MD0	PE2MD1	PE2MD0	PE1MD1	PE1MD0	PE0MD1	PE0MD0	
PGCR							PG4MD1	PG4MD0	
	PG3MD1	PG3MD0	PG2MD1	PG2MD0	PG1MD1	PG1MD0	PG0MD1	PG0MD0	
PHCR							PH4MD1	PH4MD0	
	PH3MD1	PH3MD0	PH2MD1	PH2MD0	PH1MD1	PH1MD0	PH0MD1	PH0MD0	
PSELA								PSA8	
	PSA7	PSA6	PSA5	PSA4	PSA3	PSA2	PSA1	RTCSEL1	
HIZCRA								HIZA8	
	HIZA7	HIZA6	HIZA5	HIZA4	HIZA3	HIZA2	HIZA1	HIZA0	
NCCR									
								NCCR0	
BDRB	BDB31	BDB30	BDB29	BDB28	BDB27	BDB26	BDB25	BDB24	UBC
	BDB23	BDB22	BDB21	BDB20	BDB19	BDB18	BDB17	BDB16	
	BDB15	BDB14	BDB13	BDB12	BDB11	BDB10	BDB9	BDB8	
	BDB7	BDB6	BDB5	BDB4	BDB3	BDB2	BDB1	BDB0	
BDMRB	BDMB31	BDMB30	BDMB29	BDMB28	BDMB27	BDMB26	BDMB25	BDMB24	
	BDMB23	BDMB22	BDMB21	BDMB20	BDMB19	BDMB18	BDMB17	BDMB16	
	BDMB15	BDMB14	BDMB13	BDMB12	BDMB11	BDMB10	BDMB9	BDMB8	
	BDMB7	BDMB6	BDMB5	BDMB4	BDMB3	BDMB2	BDMB1	BDMB0	
BRCR									
	SCMFCA	SCMFCA	SCMFDA	SCMFDB	PCTE	PCBA			
	DBEB	PCBB			SEQ			ETBE	
BETR					BET11	BET10	BET9	BET8	
	BET7	BET6	BET5	BET4	BET3	BET2	BET1	BET0	

レジスタ略称	ビット 31/ 23/15/ 7	ビット 30/ 22/14/ 6	ビット 29/ 21/13/ 5	ビット 28/ 20/12/ 4	ビット 27/ 19/11/ 3	ビット 26/ 18/10/ 2	ビット 25/ 17/ 9/ 1	ビット 24/ 16/ 8/ 0	モジュール
BARB	BAB31	BAB30	BAB29	BAB28	BAB27	BAB26	BAB25	BAB24	UBC
	BAB23	BAB22	BAB21	BAB20	BAB19	BAB18	BAB17	BAB16	
	BAB15	BAB14	BAB13	BAB12	BAB11	BAB10	BAB9	BAB8	
	BAB7	BAB6	BAB5	BAB4	BAB3	BAB2	BAB1	BAB0	
BAMRB	BAMB31	BAMB30	BAMB29	BAMB28	BAMB27	BAMB26	BAMB25	BAMB24	
	BAMB23	BAMB22	BAMB21	BAMB20	BAMB19	BAMB18	BAMB17	BAMB16	
	BAMB15	BAMB14	BAMB13	BAMB12	BAMB11	BAMB10	BAMB9	BAMB8	
	BAMB7	BAMB6	BAMB5	BAMB4	BAMB3	BAMB2	BAMB1	BAMB0	
BBRB							XYE	XY5	
	CDB1	CDB0	IDB1	IDB0	RWB1	RWB0	SZB1	SZB0	
BR5R	SVF				BSA27	BSA26	BSA25	BSA24	
	BSA23	BSA22	BSA21	BSA20	BSA19	BSA18	BSA17	BSA16	
	BSA15	BSA14	BSA13	BSA12	BSA11	BSA10	BSA9	BSA8	
	BSA7	BSA6	BSA5	BSA4	BSA3	BSA2	BSA1	BSA0	
BARA	BAA31	BAA30	BAA29	BAA28	BAA27	BAA26	BAA25	BAA24	
	BAA23	BAA22	BAA21	BAA20	BAA19	BAA18	BAA17	BAA16	
	BAA15	BAA14	BAA13	BAA12	BAA11	BAA10	BAA9	BAA8	
	BAA7	BAA6	BAA5	BAA4	BAA3	BAA2	BAA1	BAA0	
BAMRA	BAMA31	BAMA30	BAMA29	BAMA28	BAMA27	BAMA26	BAMA25	BAMA24	
	BAMA23	BAMA22	BAMA21	BAMA20	BAMA19	BAMA18	BAMA17	BAMA16	
	BAMA15	BAMA14	BAMA13	BAMA12	BAMA11	BAMA10	BAMA9	BAMA8	
	BAMA7	BAMA6	BAMA5	BAMA4	BAMA3	BAMA2	BAMA1	BAMA0	
BBRA									
	CDA1	CDA0	IDA1	IDA0	RWA1	RWA0	SZA1	SZA0	
BRDR	DVF				BDA27	BDA26	BDA25	BDA24	
	BDA23	BDA22	BDA21	BDA20	BDA19	BDA18	BDA17	BDA16	
	BDA15	BDA14	BDA13	BDA12	BDA11	BDA10	BDA9	BDA8	
	BDA7	BDA6	BDA5	BDA4	BDA3	BDA2	BDA1	BDA0	
SDIR*						TI2	TI1	TI0	H-UDI
SDID/SDIDH	DID31	DID30	DID29	DID28	DID27	DID26	DID25	DID24	
	DID23	DID22	DID21	DID20	DID19	DID18	DID17	DID16	
SDID/SDIDL	DID15	DID14	DID13	DID12	DID11	DID10	DID9	DID8	
	DID7	DID6	DID5	DID4	DID3	DID2	DID1	DID0	

【注】 * SDIR は H-UDI 端子から書き込みのみ可能です。

27. レジスタ一覧

27.3 各動作モードにおけるレジスタの状態

レジスタ略称	パワーオン リセット* ¹	マニュアル リセット* ¹	ソフトウェア スタンバイ* ¹	モジュール スタンバイ* ¹	スリープ	モジュール
INTEVT2	初期化	初期化	保持	保持	保持	例外処理
TRA	初期化	初期化	保持	保持	保持	
EXPEVT	初期化	初期化	保持	保持	保持	
TEA	初期化	初期化	保持	保持	保持	
CCR1	初期化	初期化	保持	保持	保持	キャッシュ
CCR2	初期化	初期化	保持	保持	保持	
IPRA	初期化	初期化	保持	保持	保持	INTC
IPRB	初期化	初期化	保持	保持	保持	
IPRC	初期化	初期化	保持	保持	保持	
IPRD	初期化	初期化	保持	保持	保持	
IPRE	初期化	初期化	保持	保持	保持	
IPRF	初期化	初期化	保持	保持	保持	
IPRG	初期化	初期化	保持	保持	保持	
IPRH	初期化	初期化	保持	保持	保持	
IMR0	初期化	初期化	保持	保持	保持	
IMR1	初期化	初期化	保持	保持	保持	
IMR4	初期化	初期化	保持	保持	保持	
IMR5	初期化	初期化	保持	保持	保持	
IMR6	初期化	初期化	保持	保持	保持	
IMR9	初期化	初期化	保持	保持	保持	
IMCR0	初期化	初期化	保持	保持	保持	
IMCR1	初期化	初期化	保持	保持	保持	
IMCR4	初期化	初期化	保持	保持	保持	
IMCR5	初期化	初期化	保持	保持	保持	
IMCR6	初期化	初期化	保持	保持	保持	
IMCR9	初期化	初期化	保持	保持	保持	
ICR0	初期化	初期化	保持	保持	保持	
ICR1	初期化	初期化	保持	保持	保持	
ICR2	初期化	初期化	保持	保持	保持	
IRR0	初期化	初期化	保持	保持	保持	
CMNCR	初期化	保持	保持	保持	保持	BSC
CS0BCR	初期化	保持	保持	保持	保持	
CS3BCR	初期化	保持	保持	保持	保持	

レジスタ略称	パワーオン リセット* ¹	マニュアル リセット* ¹	ソフトウェア スタンバイ* ¹	モジュール スタンバイ* ¹	スリープ	モジュール
CS4BCR	初期化	保持	保持	保持	保持	BSC
CS0WCR	初期化	保持	保持	保持	保持	
CS3WCR	初期化	保持	保持	保持	保持	
CS4WCR	初期化	保持	保持	保持	保持	
SDCR	初期化	保持	保持	保持	保持	
RTCSR	初期化	保持	保持	保持	保持	
RTCNT	初期化	保持	保持	保持	保持	
RTCOR	初期化	保持	保持	保持	保持	
SDMR3						DMAC
SAR_0	不定	不定	保持	保持	保持	
DAR_0	不定	不定	保持	保持	保持	
DMATCR_0	不定	不定	保持	保持	保持	
CHCR_0	初期化	初期化	保持	保持	保持	
IAR_0	不定	不定	保持	保持	保持	
SAR_1	不定	不定	保持	保持	保持	
DAR_1	不定	不定	保持	保持	保持	
DMATCR_1	不定	不定	保持	保持	保持	
CHCR_1	初期化	初期化	保持	保持	保持	
IAR_1	不定	不定	保持	保持	保持	
SAR_2	不定	不定	保持	保持	保持	
DAR_2	不定	不定	保持	保持	保持	
DMATCR_2	不定	不定	保持	保持	保持	
CHCR_2	初期化	初期化	保持	保持	保持	
IAR_2	不定	不定	保持	保持	保持	
SAR_3	不定	不定	保持	保持	保持	
DAR_3	不定	不定	保持	保持	保持	
DMATCR_3	不定	不定	保持	保持	保持	
CHCR_3	初期化	初期化	保持	保持	保持	
IAR_3	不定	不定	保持	保持	保持	
DMAOR	初期化	初期化	保持	保持	保持	CPG
DMARS0	初期化	初期化	保持	保持	保持	
DMARS1	初期化	初期化	保持	保持	保持	
DMARS2	初期化	初期化	保持	保持	保持	
FRQCR	初期化* ²	保持	保持	保持	保持	WDT
WTCNT	初期化* ²	保持	保持	保持	保持	
WTCSR	初期化* ²	保持	保持	保持	保持	

27. レジスタ一覧

レジスタ略称	パワーオン リセット* ¹	マニュアル リセット* ¹	ソフトウェア スタンバイ* ¹	モジュール スタンバイ* ¹	スリープ	モジュール	
STBCR	初期化	保持	保持	保持	保持	低消費電力	
STBCR2	初期化	保持	保持	保持	保持		
STBCR3	初期化	保持	保持	保持	保持		
STBCR4	初期化	保持	保持	保持	保持		
TSTR	初期化	初期化	初期化	初期化	保持	TMU	
TCOR_0	初期化	初期化	保持	保持	保持		
TCNT_0	初期化	初期化	保持	保持	保持		
TCR_0	初期化	初期化	保持	保持	保持		
TCOR_1	初期化	初期化	保持	保持	保持		
TCNT_1	初期化	初期化	保持	保持	保持		
TCR_1	初期化	初期化	保持	保持	保持		
TCOR_2	初期化	初期化	保持	保持	保持		
TCNT_2	初期化	初期化	保持	保持	保持		
TCR_2	初期化	初期化	保持	保持	保持		
SIMDR	初期化	初期化	保持	保持	保持		SIOF
SISCR	初期化	初期化	保持	保持	保持		
SITDAR	初期化	初期化	保持	保持	保持		
SIRDAR	初期化	初期化	保持	保持	保持		
SICDAR	初期化	初期化	保持	保持	保持		
SICTR	初期化	初期化	保持	保持* ³	保持		
SIFCTR	初期化	初期化	保持	保持	保持		
SISTR	初期化	初期化	保持	初期化	保持		
SIER	初期化	初期化	保持	保持	保持		
SITDR	初期化	初期化	保持	初期化	保持		
SIRDR	初期化	初期化	保持	保持	保持		
SITCR	初期化	初期化	保持	初期化	保持		
SIRCR	初期化	初期化	保持	保持	保持		
SPICR	初期化	初期化	保持	保持	保持		
SCSMR_0	初期化	初期化	保持	保持	保持	SCIF_0	
SCBRR_0	初期化	初期化	保持	保持	保持		
SCSCR_0	初期化	初期化	保持	保持	保持		
SCTDSR_0	初期化	初期化	保持	保持	保持		
SCFER_0	初期化	初期化	保持	保持	保持		
SCSSR_0	初期化	初期化	保持	保持	保持		
SCFCR_0	初期化	初期化	保持	保持	保持		

レジスタ略称	パワーオン リセット* ¹	マニュアル リセット* ¹	ソフトウェア スタンバイ* ¹	モジュール スタンバイ* ¹	スリープ	モジュール	
SCFDR_0	初期化	初期化	保持	保持	保持	SCIF_0	
SCFTDR_0	初期化	初期化	保持	保持	保持		
SCFRDR_0	初期化	初期化	保持	保持	保持		
SCSMR_1	初期化	初期化	保持	保持	保持	SCIF_1	
SCBRR_1	初期化	初期化	保持	保持	保持		
SCSCR_1	初期化	初期化	保持	保持	保持		
SCTDSR_1	初期化	初期化	保持	保持	保持		
SCFER_1	初期化	初期化	保持	保持	保持		
SCSSR_1	初期化	初期化	保持	保持	保持		
SCFCR_1	初期化	初期化	保持	保持	保持		
SCFDR_1	初期化	初期化	保持	保持	保持		
SCFTDR_1	初期化	初期化	保持	保持	保持		
SCFRDR_1	初期化	初期化	保持	保持	保持		
EXCPGCR	初期化	初期化	保持	保持	保持		USBPM
HREVR	初期化	初期化	保持	保持	保持		USBH
HCTLR	初期化	初期化	保持	保持	保持		
HCSR	初期化	初期化	保持	保持	保持		
HISR	初期化	初期化	保持	保持	保持		
HIER	初期化	初期化	保持	保持	保持		
HIDR	初期化	初期化	保持	保持	保持		
HHCCAR	初期化	初期化	保持	保持	保持		
HPCEDR	初期化	初期化	保持	保持	保持		
HCHEDR	初期化	初期化	保持	保持	保持		
HCCEDR	初期化	初期化	保持	保持	保持		
HBHEDR	初期化	初期化	保持	保持	保持		
HBCEDR	初期化	初期化	保持	保持	保持		
HDHEDR	初期化	初期化	保持	保持	保持		
HFIR	初期化	初期化	保持	保持	保持		
HFRR	初期化	初期化	保持	保持	保持		
HFNR	初期化	初期化	保持	保持	保持		
HPSR	初期化	初期化	保持	保持	保持		
HLSTR	初期化	初期化	保持	保持	保持		
HRDRA	初期化	初期化	保持	保持	保持		
HRDRB	初期化	初期化	保持	保持	保持		
HRSR	初期化	初期化	保持	保持	保持		
HRRSR	初期化	初期化	保持	保持	保持		

27. レジスタ一覧

レジスタ略称	パワーオン リセット* ¹	マニュアル リセット* ¹	ソフトウェア スタンバイ* ¹	モジュール スタンバイ* ¹	スリープ	モジュール
IFR0	初期化	初期化	保持	保持	保持	USBF
ISR0	初期化	初期化	保持	保持	保持	
IER0	初期化	初期化	保持	保持	保持	
EPDR0i	初期化	初期化	保持	保持	保持	
EPDR0o	初期化	初期化	保持	保持	保持	
EPDR0s	初期化	初期化	保持	保持	保持	
EPDR1	初期化	初期化	保持	保持	保持	
EPDR2i	初期化	初期化	保持	保持	保持	
EPDR2o	初期化	初期化	保持	保持	保持	
EPDR3i	初期化	初期化	保持	保持	保持	
EPDR3o	初期化	初期化	保持	保持	保持	
EPDR4	初期化	初期化	保持	保持	保持	
EPDR5	初期化	初期化	保持	保持	保持	
EPDR6	初期化	初期化	保持	保持	保持	
EPSZ0o	初期化	初期化	保持	保持	保持	
EPSZ2o	初期化	初期化	保持	保持	保持	
EPSZ3o	初期化	初期化	保持	保持	保持	
EPSZ6	初期化	初期化	保持	保持	保持	
TRG	初期化	初期化	保持	保持	保持	
DASTS	初期化	初期化	保持	保持	保持	
FCLR	初期化	初期化	保持	保持	保持	
DMA	初期化	初期化	保持	保持	保持	
EPSTL	初期化	初期化	保持	保持	保持	
CVR	初期化	初期化	保持	保持	保持	
TSR	初期化	初期化	保持	保持	保持	
CTLR	初期化	初期化	保持	保持	保持	
EPIR	初期化	初期化	保持	保持	保持	
全レジスタ	初期化 / 保持* ⁴	保持	保持	保持	保持	BT
DADR_0	初期化	保持	保持	保持	保持	DAC
DADR_1	初期化	保持	保持	保持	保持	
DACR	初期化	保持	保持	保持	保持	
PADR	初期化	保持	保持	保持	保持	I/O ポート
PBDR	初期化	保持	保持	保持	保持	
PDDR	初期化	保持	保持	保持	保持	
PEDR	初期化	保持	保持	保持	保持	

レジスタ略称	パワーオン リセット*1	マニュアル リセット*1	ソフトウェア スタンバイ*1	モジュール スタンバイ*1	スリープ	モジュール	
PGDR	初期化	保持	保持	保持	保持	I/O ポート	
PHDR	初期化	保持	保持	保持	保持		
PACR	初期化	保持	保持	保持	保持	PFC	
PBCR	初期化	保持	保持	保持	保持		
PDCR	初期化	保持	保持	保持	保持		
PECR	初期化	保持	保持	保持	保持		
PGCR	初期化	保持	保持	保持	保持		
PHCR	初期化	保持	保持	保持	保持		
PSELA	初期化	保持	保持	保持	保持		
HIZCRA	初期化	保持	保持	保持	保持		
NCCR	初期化	保持	保持	保持	保持		
BDRB	初期化	保持	保持	保持	保持		UBC
BDMRB	初期化	保持	保持	保持	保持		
BRCR	初期化	保持	保持	保持	保持		
BETR	初期化	保持	保持	保持	保持		
BARB	初期化	保持	保持	保持	保持		
BAMRB	初期化	保持	保持	保持	保持		
BBRB	初期化	保持	保持	保持	保持		
BRSR	初期化	保持	保持	保持	保持		
BARA	初期化	保持	保持	保持	保持		
BAMRA	初期化	保持	保持	保持	保持		
BBRA	初期化	保持	保持	保持	保持		
BRDR	初期化	保持	保持	保持	保持		
SDIR	保持*6	保持*6	保持	保持	保持	H-UDI	
SDID/SDIDH*5	-	-	-	-	-		
SDID/SDIDL*5	-	-	-	-	-		
U メモリ	不定	不定	保持	保持	保持	U メモリ	
X/Y メモリ	不定	不定	保持	保持	保持	X/Y メモリ	

【注】 *1 レジスタの初期値は、各モジュールの章を参照してください。また、初期値が不定のレジスタについても値が保持されないため、初期化と表現しています。

*2 RESETP 端子からのリセット時に初期化します。その他のリセットでは、値を保持します。

*3 初期化されるビットがあります。

*4 端子から入力されるパワーオンリセットでのみ初期化されます。WDT から発生するパワーオンリセットでは初期化されません

*5 固定値です。

*6 SDIR は、TRST で初期化されます。

28. 電気的特性

28.1 絶対最大定格

絶対最大定格を表 28.1 に示します。

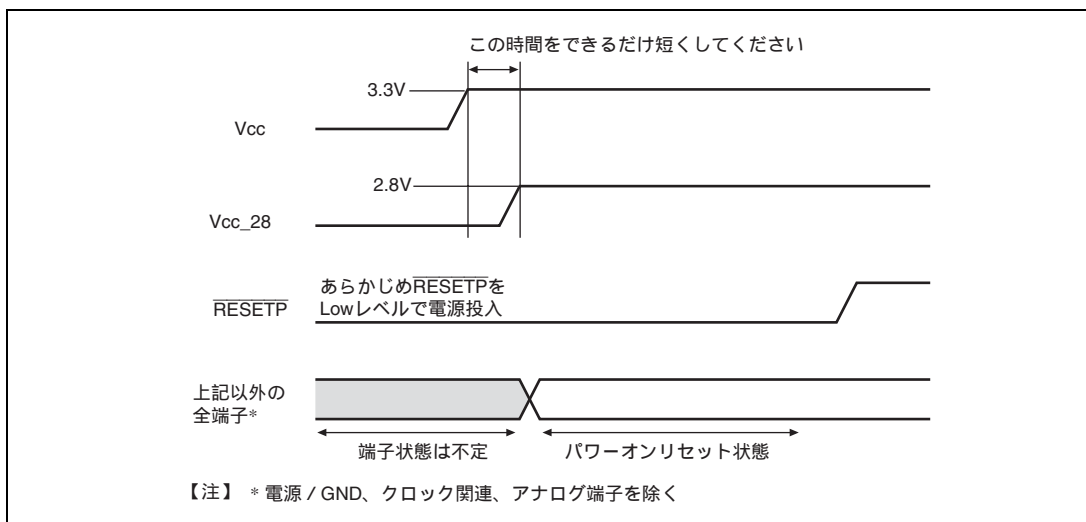
表 28.1 絶対最大定格

項目	記号	定格値	単位
電源電圧 (Vcc_28 以外)	V _{cc} (I/O)、V _{cc} (内部)、V _{cc} (PLL)、 V _{cc} (DLL)、V _{cc} (SREG)	- 0.3 ~ 4.2	V
電源電圧 (Vcc_28)	Vcc_28	- 0.3 ~ 4.2	V
電源電圧 (VDD) : 外部電圧印加モード時	VDD	- 0.3 ~ 2.1	V
入力電圧 (Vcc_28 以外)	V _{in}	- 0.3 ~ V _{cc} + 0.3	V
入力電圧 (Vcc_28)	V _{in}	- 0.3 ~ V _{cc_28} + 0.3	V
アナログ電源電圧 (DAC)	AV _{cc} (DAC)	- 0.3 ~ 4.2	V
アナログ電源電圧 (USB)	AV _{cc} (USB)	-0.3 ~ 4.2	V
アナログ入力電圧 (USB)	V _{in}	-0.3 ~ AV _{cc} (USB)+0.3	V
動作温度	T _{opr}	-40 ~ 85	
保存温度	T _{stg}	- 55 ~ 125	

【使用上の注意】

1. 絶対最大定格を越えて LSI を使用した場合、LSI の永久破壊となることがあります。
2. Vcc_28 と他の 3.3V 系電源 (V_{cc}、AV_{cc}) の投入順序について特に規定はありませんが、一方の電源を遮断すると動作が不安定になりますので、以下を推奨します。
 - (1) Vcc_28 以外の 3.3V 系電源を先に投入し、この後、Vcc_28 電源を投入してください。この間隔をできるだけ短くすることを推奨します。
 - (2) すべての電源に電圧が印加され、RESETP 端子にローレベルが入力されるまでの間、内部回路は不確定なので、各端子の状態も不定となります。この不定状態によってシステムが誤動作を起こさないようにシステム設計を行ってください。以下に電源投入時の波形を示します。
 - (3) Vcc_28 の電源電圧は、上記絶対最大定格の範囲内であつ、接続する RF IC の電源電圧と同じにしてください。

28. 電气的特性



電源投入シーケンス

3. 電源の切断順序について特に規定はありませんが、一方の電源を遮断すると動作が不安定になりますので、以下を推奨します。
 - (1) 電源投入時と逆に Vcc_28 電源を先に切断し、この後、3.3V 系電源を切ってください。この間隔をできるだけ短くすることを推奨します。
 - (2) Vcc_28 電源だけを切断した状態では、端子状態は不定となります。この状態によってシステムが誤動作を起こさないようにシステム設計を行ってください。
4. 内蔵レギュレータ使用時、各 V_{DD} 端子には電源雑音平滑化のためコンデンサを外付けしてください。
5. 内蔵レギュレータ使用時、各 V_{DD} 端子には外部から電圧を印加しないでください。また、他のデバイスへの電流供給はできません。

28.2 DC 特性

DC 特性を表 28.2、表 28.3 に示します。

表 28.2 DC 特性 (1) 【共通項目】

条件 : Ta = -40 ~ 85

項 目	記号	Min.	Typ.	Max.	単位	測定条件	
消費電流	通常動作時	I_{CC}	-	-	350	mA	V_{CC} 、 AV_{CC} (DAC)、 AV_{CC} (USB) = 3.3V $I = 120\text{MHz}$ $B = 60\text{MHz}$
V_{CC} -28 消費電流		$I_{VCC,28}$	-	-	6	mA	弊社 RF IC 接続時*1
入力リーク電流	全入力端子	$ I_{in} $	-	-	1.0	μA	$V_{in} =$ $0.5 \sim V_{CC} - 0.5\text{V}$
スリープ状態 リーク電流	入出力、全出力 端子 (オフ状態)	$ I_{STI} $	-	-	1.0	μA	$V_{in} =$ $0.5 \sim V_{CC} - 0.5\text{V}$
プルアップ抵抗	ポート端子	P_{pull}	30	60	120	k	
端子容量	全デジタル端子*2	C	-	-	10	pF	
アナログ電源電圧 (DAC)		AV_{CC} (DAC)	2.7	3.3	3.6	V	
内部電源電圧 (V_{DD})		V_{DD}	1.4	1.5	1.6	V	外部印加モード時

【注】 *1 接続可能な RF IC は、HD157102NP です。

*2 電源、アナログ、 V_{BB} 、USB_N、USB_P、RDI_CTRL4、DA0、DA1、EXTAL、XTAL、EXTAL2、XTAL2、RREF、NC は除く。

*3 AV_{CC} (DAC)は、 $V_{CC} - 0.2\text{V}$ AV_{CC} (DAC) $V_{CC} + 0.2\text{V}$ の条件を満足しなければなりません。
D/A コンバータを使用しない場合でも AV_{CC} (DAC)端子を開放しないで、 AV_{CC} (DAC)は V_{CC} に AV_{SS} (DAC)は V_{SS} に接続してください。

*4 消費電流値は、 $V_{Imin} = V_{CC} - 0.5\text{V}$ 、 $V_{Imax} = 0.5\text{V}$ の条件で、すべての出力端子を無負荷状態にした場合の値です。

28. 電気的特性

表 28.2 DC 特性 (2-a) 【DAC、USB 関連端子を除く】

条件 : Ta = -40 ~ 85

項目		記号	Min.	Typ.	Max.	単位	測定条件
電源電圧		V_{CC}	2.7	3.3	3.6	V	
		V_{CC-28}	2.7	2.8	3.0		* ¹
			2.7	3.3	3.6		* ²
入力 High レベル電圧	NMI、MD1、MD2、 MD5、ASEMD0、 IRQ0、IRQ2 ~ 4	V_{IH}	$V_{CC} \times 0.9$	-	$V_{CC} + 0.3$	V	
	\overline{TRST}		$V_{CC} \times 0.75$	-	$V_{CC} + 0.3$		
	EXTAL		$V_{CC} - 0.3$	-	$V_{CC} + 0.3$		
			2.0	-	$V_{CC} + 0.3$		HD157102NP の BRCLK 接続時
	EXTAL2		$V_{CC} - 0.3$	-	$V_{CC} + 0.3$		
	\overline{RESETP}		$V_{CC-28} \times 0.9$	-	$V_{CC-28} + 0.3$		
	RDI_TXTRDATA、 RDI_REFCLK_IN、 RCI_SPI_TXRX		2.0	-	$V_{CC-28} + 0.3$		
	その他の入力端子		2.0	-	$V_{CC} + 0.3$		
入力 Low レベル電圧	NMI、IRQ0、 IRQ2 ~ 4、MD1、 MD2、MD5、 $\overline{ASEMD0}$ 、 \overline{TRST}	V_{IL}	- 0.3	-	$V_{CC} \times 0.1$	V	
	EXTAL、EXTAL2		- 0.3	-	$V_{CC} \times 0.2$		
	\overline{RESETP} 、 RDI_TXTRDATA、 RDI_REFCLK_IN、 RCI_SPI_TXRX		- 0.3	-	$V_{CC-28} \times 0.1$		
	その他の入力端子		- 0.3	-	$V_{CC} \times 0.2$		
出力 High レベル電圧	RDI_CTRL4	V_{OH}	2.25	-	-	V	V_{CC-28} 、 $V_{CC} = 2.7V$ 、 $I_{OH} = - 3mA$
	その他の出力端子		2.4	-	-		V_{CC-28} 、 $V_{CC} = 2.7V$ 、 $I_{OH} = - 200 \mu A$
			2.0	-	-		V_{CC-28} 、 $V_{CC} = 2.7V$ 、 $I_{OH} = - 2mA$
出力 Low レベル電圧	RDI_TXTRDATA、 RDI_RXBDW_OUT、 RCI_SPI_CLK、 RCI_SPI_ENB、 RCI_SPI_TXRX、 RDI_CTRL3	V_{OL}	-	-	0.55	V	$V_{CC-28} = 3.0V$ 、 $I_{OL} = 2.0mA$

項目	記号	Min.	Typ.	Max.	単位	測定条件	
出力 Low レベル電圧	RDI_CTRL4	V_{OL}	-	-	0.25	V	$V_{CC-28} = 3.0V$ 、 $I_{OL} = 0.5mA$
	その他の出力端子	V_{OL}	-	-	0.55	V	$V_{CC} = 3.6V$ 、 $I_{OL} = 2.0mA$

【注】 *1 ルネサス テクノロジ製 RF IC (HD157102NP) を接続した場合

*2 RF IC を接続しない場合は、 V_{CC} に接続してください。

表 28.2 DC 特性 (2-b) 【USB 関連端子*】

条件: $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	測定条件
入力 High レベル電圧	V_{IH}	2.0	-	$V_{CC} + 0.3$	V	
入力 Low レベル電圧	V_{IL}	-0.3	-	$V_{CC} \times 0.2$	V	
入力 High レベル電圧 (UCLK)	$V_{IH}(UCLK)$	$V_{CC} - 0.3$	-	$V_{CC} + 0.3$	V	
入力 Low レベル電圧 (UCLK)	$V_{IL}(UCLK)$	-0.3	-	$V_{CC} \times 0.2$	V	
出力 High レベル電圧	V_{OH}	2.4	-	-	V	$V_{CC} = 2.7V$ 、 $I_{OH} = -200 \mu A$
		2.0	-	-		$V_{CC} = 2.7V$ 、 $I_{OH} = -2mA$
出力 Low レベル電圧	V_{OL}	-	-	0.55	V	$V_{CC} = 3.6V$ 、 $I_{OL} = 2.0mA$

【注】 * UCLK、 $\overline{USB_PWR_EN/USB_PULLUP}$ 、 $\overline{USB_OVR_CRNT/USB_VBUS}$ の各端子

表 28.2 DC 特性 (2-c) 【USB トランシーバ関連端子*】

条件: $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	測定条件
電源電圧	$AV_{CC}(USB)$	3.0	3.3	3.6	V	USB を使用する 場合
		2.7	3.0	3.6		USB を使用し ない場合
差動入力感度	V_{DI}	0.2	-	-	V	$ (USB_P)-(USB_N) $
差動コモンモード範囲	V_{CM}	0.8	-	2.5	V	
シングルエンディッドレシーバ スレッシュホールド電圧	V_{SE}	0.8	-	2.0	V	
出力 High レベル電圧	V_{OH}	2.8	-	$AV_{CC}(USB)$	V	USB を使用する 場合
出力 Low レベル電圧	V_{OL}	-	-	0.3	V	
トライステートリーク電流	I_{LO}	-10	-	10	μA	$0V < V_{IN} < 3.3V$

【注】 * USB_P、USB_N 端子

28. 電気的特性

表 28.3 出力許容電流値

条件： $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $AV_{CC} = 2.7 \sim 3.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位
出力 Low レベル許容電流 (1 端子あたり)	I_{OL}	-	-	2	mA
出力 Low レベル許容電流 (総和)	I_{OL}	-	-	50	mA
出力 High レベル許容電流 (1 端子あたり)	$-I_{OH}$	-	-	2	mA
出力 High レベル許容電流 (総和)	$(-I_{OH})$	-	-	50	mA

【注】 LSI の信頼性を確保するため、出力電流値は表 28.3 の値を越えないようにしてください。

28.3 AC 特性

本 LSI の入力は原則としてクロック同期入力です。特にことわりがない限り、各入力信号のセットアップ・ホールド時間は必ず守ってください。

表 28.4 動作周波数

条件： $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $AV_{CC} = 2.7 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	備考
動作周波数	CPU、 キャッシュ (I)	f	20	-	120	MHz
	外部バス (B)	20	-	60		
	周辺モジュール (P)	4	-	30		

28.3.1 クロックタイミング

表 28.5 クロックタイミング

条件： $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $AV_{CC} = 2.7 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

外部バス最大動作周波数：60MHz

項目	記号	Min.	Max.	単位	参照図
EXTAL クロック入力周波数	f_{EX}	10	30	MHz	28.1
EXTAL クロック入力サイクル時間	t_{EXCYC}	33.3	100	ns	
EXTAL クロック入力 Low レベルパルス幅	t_{EXL}	15		ns	
EXTAL クロック入力 High レベルパルス幅	t_{EXH}	15		ns	
EXTAL クロック入力立ち上がり時間	t_{EXr}		4	ns	
EXTAL クロック入力立ち下がり時間	t_{EXf}		4	ns	
CKIO クロック入力周波数	f_{CKI}	20	60	MHz	28.2
CKIO クロック入力サイクル時間	t_{CKICYC}	16.6	50	ns	
CKIO クロック入力 Low レベルパルス幅	t_{CKIL}	4	-	ns	

項目	記号	Min.	Max.	単位	参照図
CKIO クロック入力 High レベルパルス幅	t_{CKIH}	4	-	ns	28.2
CKIO クロック入力立ち上がり時間	t_{CKIR}	-	2	ns	
CKIO クロック入力立ち下がり時間	t_{CKIF}	-	2	ns	
CKIO クロック出力周波数	f_{OP}	20	60	MHz	28.3
CKIO クロック出力サイクル時間	t_{cyc}	16.7	50	ns	
CKIO クロック出力 Low レベルパルス幅	t_{CKOL}	5	-	ns	
CKIO クロック出力 High レベルパルス幅	t_{CKOH}	5	-	ns	
CKIO クロック出力立ち上がり時間	t_{CKOr}	-	6	ns	
CKIO クロック出力立ち下がり時間	t_{CKOf}	-	6	ns	
RESETP アサート時間	t_{RESPW}	100	-	μs	
スタンバイ復帰発振安定時間 1	t_{SOC2}	10	-	ms	28.4
スタンバイ復帰発振安定時間 2	t_{SOC3}	-	10	ms	28.5
TRST アサート時間	t_{TRSTW}	100	-	μs	28.6、28.8
TRST アサート時間 2	t_{TRSTW2}	10	-	ms	28.10
RESETP アサート時間 1	t_{RESPW1}	100 ^{*1}	-	μs	28.6、 28.8、 28.10
RESETP アサート時間 2	t_{RESPW2}	100	-	μs	28.6 ~ 28.9
RESETP アサート時間 3	t_{RESPW3}	100 ^{*1}	-	μs	28.7、 28.9、 28.11
パワーオン発振安定時間	t_{SOC}	200	-	μs	28.10
PLL 同期安定化時間	t_{PLL}	100 ^{*2}	-	μs	28.6 ~ 28.11

- 【注】 *1 図 28.6 ~ 図 28.9 において、EXTAL/XTAL 端子に水晶発振子を接続する場合は、Min.10msec.が必要です。
- *2 図 28.6、28.7 の場合、RESETP 信号入力ネグート直後から RDI_REFCLK_IN に入力されるクロック 1312 サイクル分の時間がこの規定をクリアできるようにしてください。
- *3 図 28.6 ~ 図 28.11 の EXTAL 入力に対する CKIO 出力および内部クロックは同じ周波数ではなく、リセット時の初期設定値としては、
EXTAL:CKIO:内部クロック (Iclk:Bclk:Pclk) = 1:2:(2:2:1/2)
となります。ただし、クロックモード 7 は CKIO 入力モードとなり、
CKIO:内部クロック (Iclk:Bclk:Pclk) = 1:(1:1:1/4)
となります。詳しくは「第 11 章 クロックパルス発振器 (CPG)」を参照してください。
- *4 図 28.8 ~ 図 28.11 に記載されている「RF 起動時間 (T_{xtl})」(HD157100NP および HD157102NP の内蔵発振回路の安定時間)は、HD157102NP のデータシートでいずれも 6msec. (Typical) となっています。

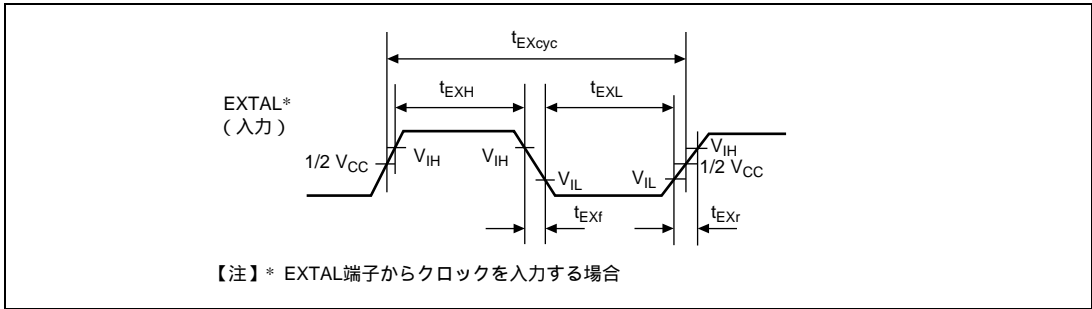


図 28.1 EXTAL クロック入力タイミング

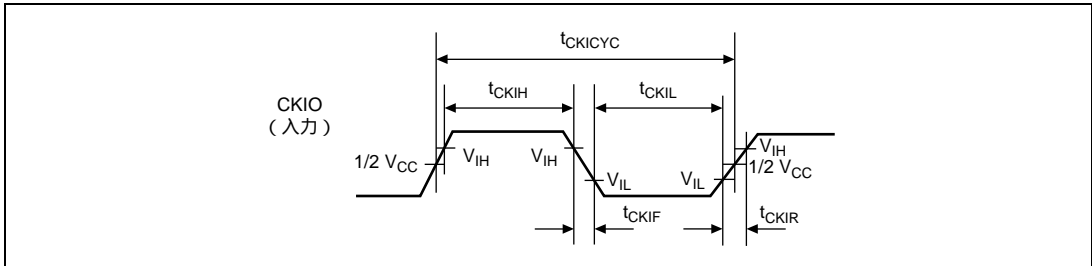


図 28.2 CKIO クロック入力タイミング

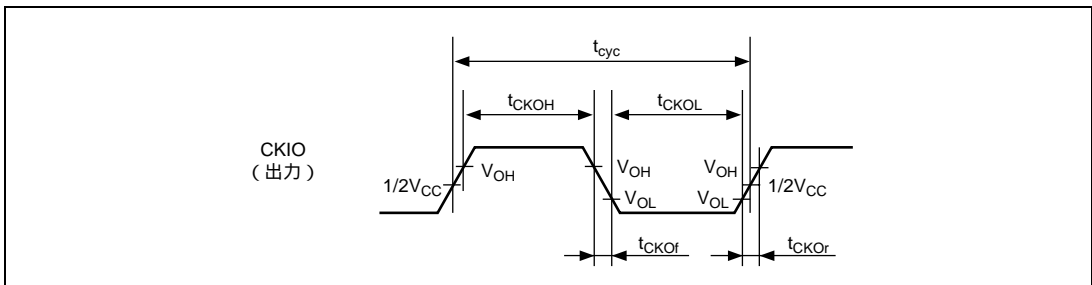


図 28.3 CKIO クロック出力タイミング

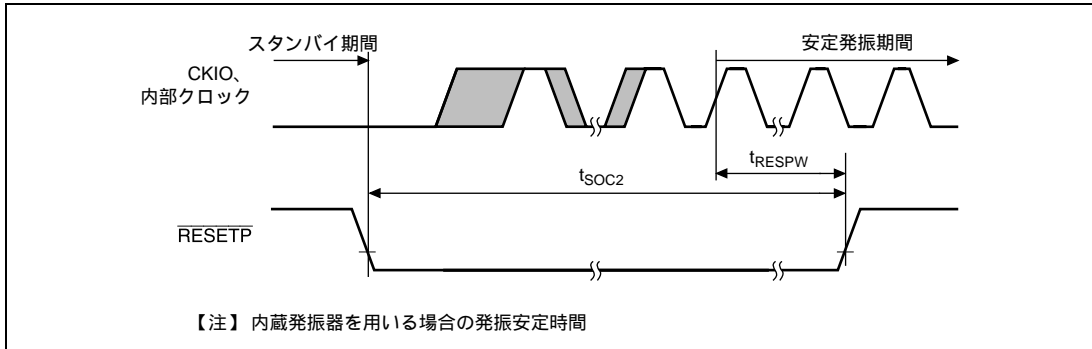


図 28.4 スタンバイ復帰時発振安定時間（リセットによる復帰）

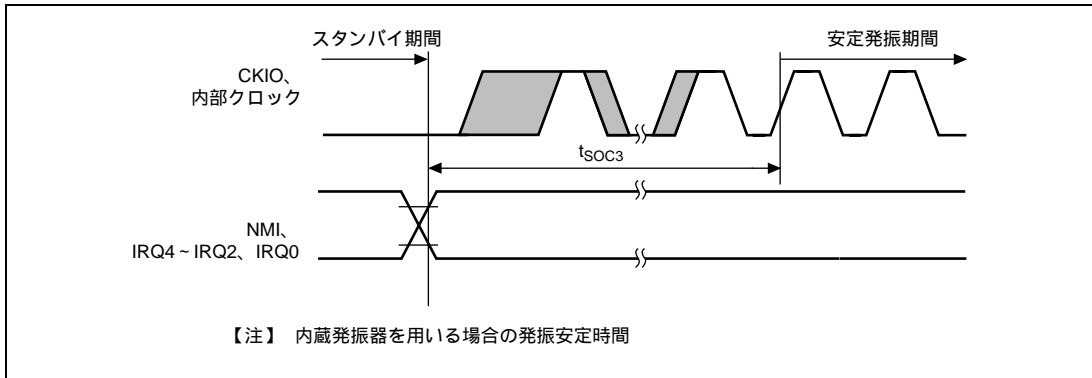


図 28.5 スタンバイ復帰時発振安定時間（NMI、IRQ による復帰）

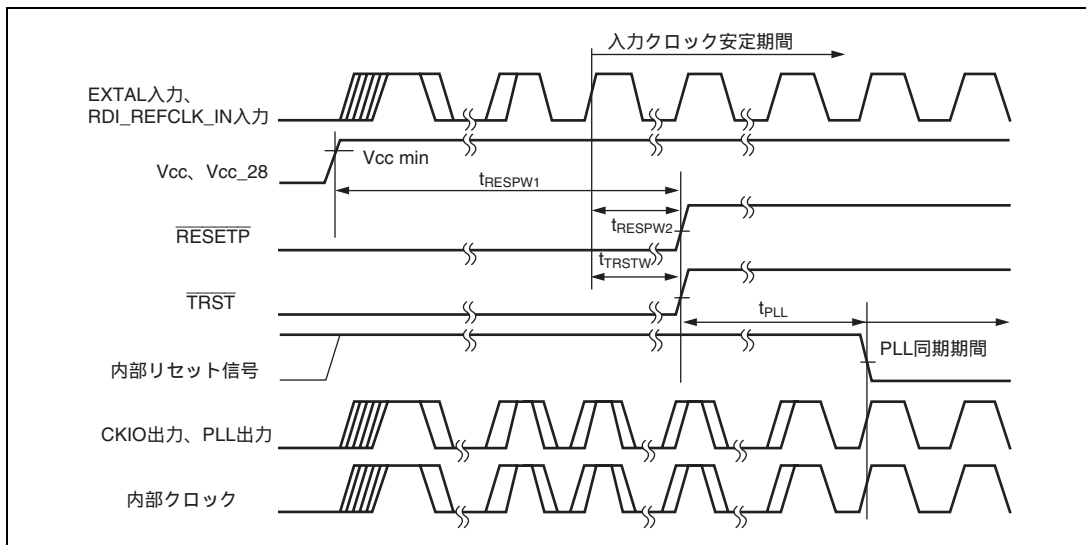


図 28.6 パワーオンリセット時のタイミングチャート
 (電源 ON から発振するクロックを RDI_REFCLK_IN 端子および EXTAL 端子に供給した場合)

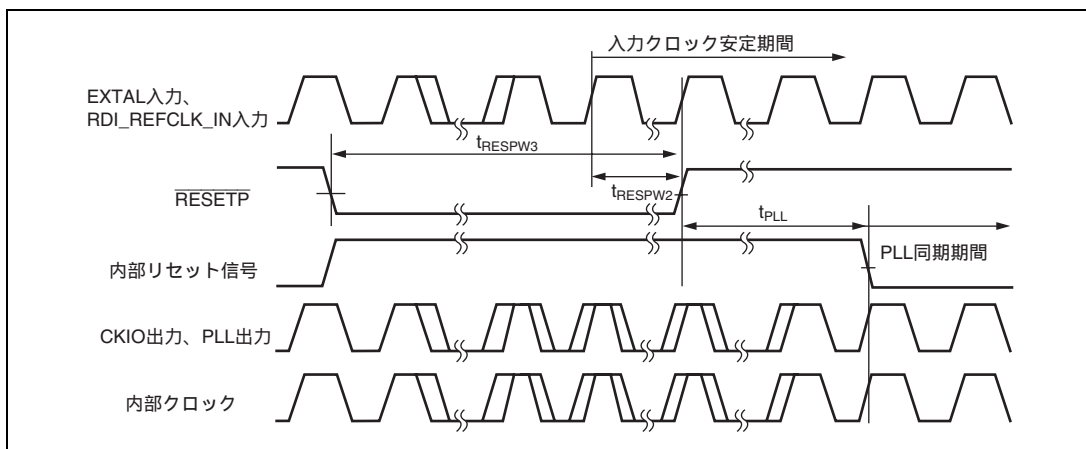


図 28.7 通常のリセット時のタイミングチャート
 (電源 ON から発振するクロックを RDI_REFCLK_IN 端子および EXTAL 端子に供給した場合)

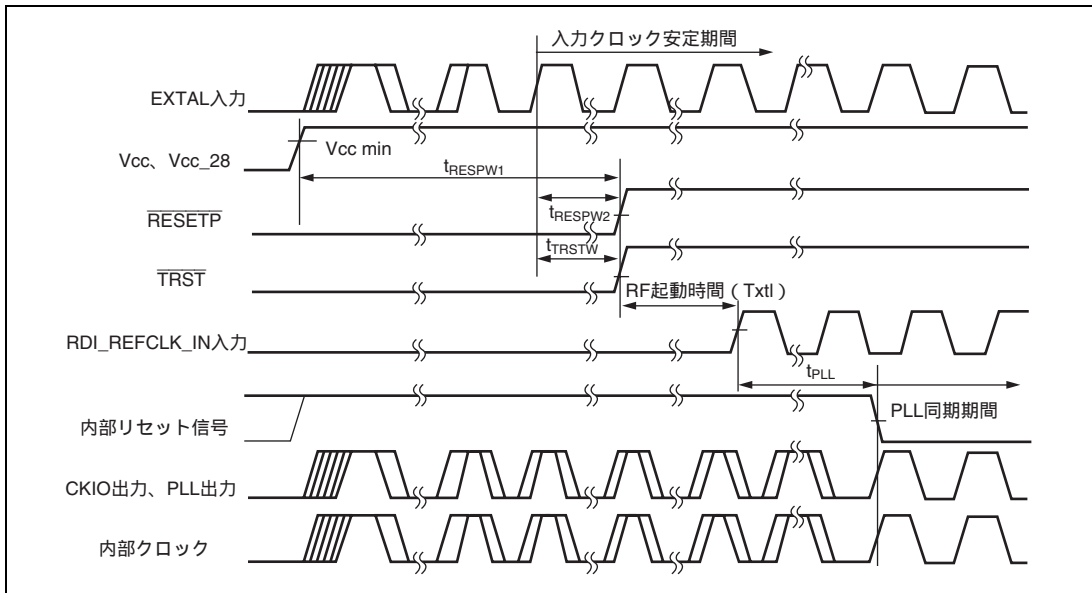


図 28.8 パワーオンリセット時のタイミングチャート

(弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子のみに接続し、EXTAL 端子には電源 ON から発振する、BRCLK とは別のクロックを供給した場合)

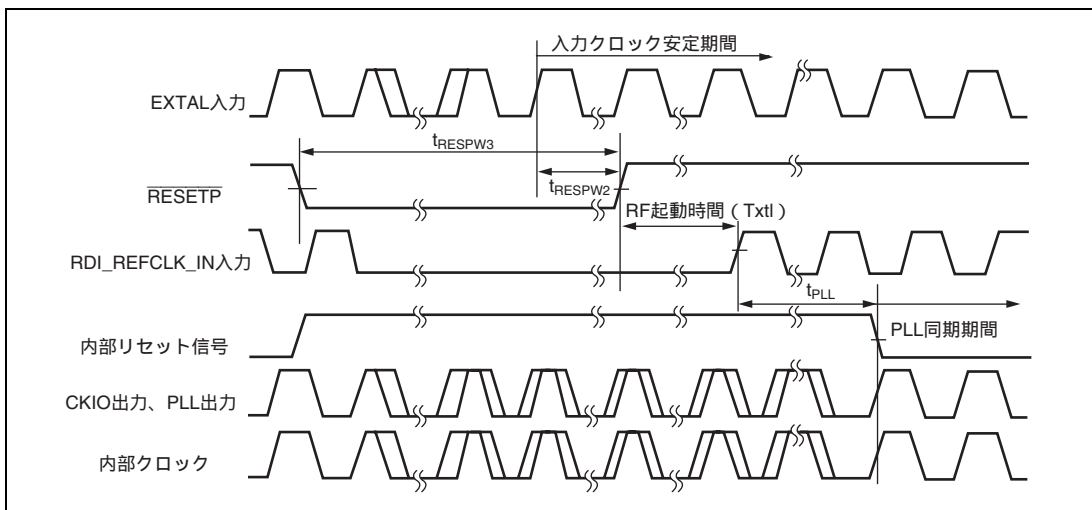


図 28.9 通常のリセット時のタイミングチャート

(弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子のみに接続し、EXTAL 端子には電源 ON から発振する、BRCLK とは別のクロックを供給した場合)

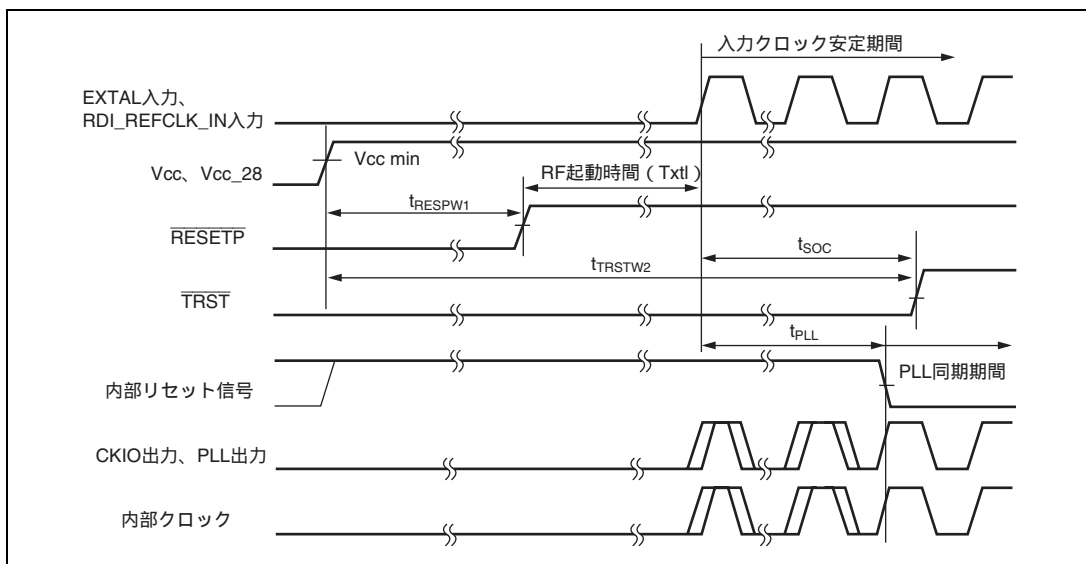


図 28.10 パワーオンリセット時のタイミングチャート

(弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子および EXTAL 端子に接続し、電源投入時の TRST 端子および RESETP 端子に供給する信号を個別に供給した場合)

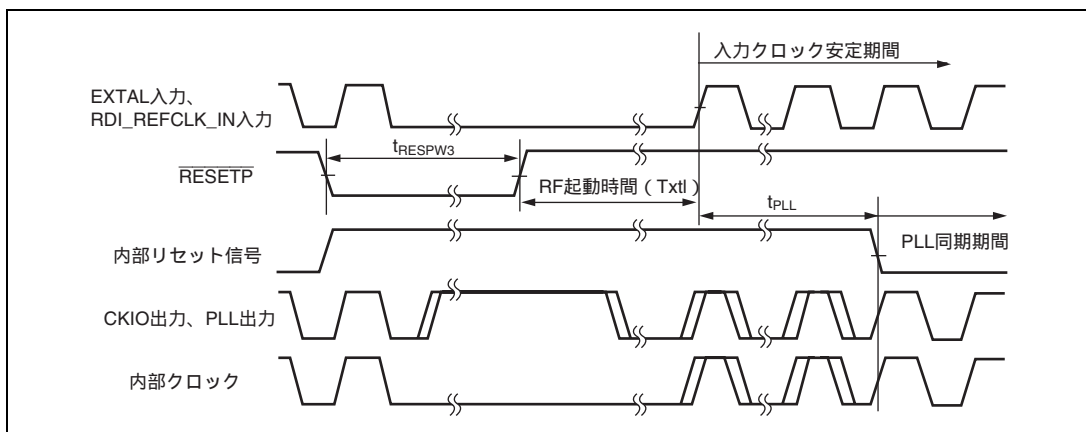


図 28.11 通常のリセット時のタイミングチャート

(弊社 RF IC (HD157102NP) の BRCLK を RDI_REFCLK_IN 端子および EXTAL 端子に接続し、電源投入時の TRST 端子および RESETP 端子に供給する信号を個別に供給した場合)

28.3.2 制御信号タイミング

表 28.6 制御信号タイミング

条件：V_{CC} = 2.7 ~ 3.6V、V_{CC-28} = 2.7 ~ 3.0V、AV_{CC} = 2.7 ~ 3.6V、V_{DD} = 1.4 ~ 1.6V、Ta = -40 ~ 85 *²

項目	記号	Min.	Max.	単位	参照図
RESETP パルス幅	t _{RESPW}	100	-	μs	28.12、 28.13
RESETP セットアップ時間* ¹	t _{RESPS}	30	-	ns	
RESETP ホールド時間	t _{RESPH}	4	-	ns	
BOOT_E セットアップ時間* ⁴	t _{BOOTS}	50	-	μs	28.12
BREQ セットアップ時間	t _{BREQS}	1/2t _{cyt} * ³ + 7	-	ns	28.16
BREQ ホールド時間	t _{BREQH}	1/2t _{cyt} + 3	-	ns	
NMI セットアップ時間* ¹	t _{NMIS}	20	-	ns	28.13
NMI ホールド時間	t _{NMIH}	4	-	ns	
IRQ4 ~ IRQ2、IRQ0 セットアップ時間* ¹	t _{IRQS}	20	-	ns	
IRQ4 ~ IRQ2、IRQ0 ホールド時間	t _{IRQH}	4	-	ns	
IRQOUT 遅延時間	t _{IRQOD}	-	13	ns	28.14
REFOUT 遅延時間	t _{REFOD}	-	13	ns	28.15
BACK 遅延時間	t _{BACKD}	1/2t _{cyt}	1/2t _{cyt} + 13	ns	28.16、 28.17
バスタイステート遅延時間 1	t _{BOFF1}	0	30	ns	
バスタイステート遅延時間 2	t _{BOFF2}	0	30	ns	28.16、 28.17
バスバッファオンタイム 1	t _{BON1}	0	30	ns	
バスバッファオンタイム 2	t _{BON2}	0	30	ns	

【注】 *1 RESETP、NMI、および IRQ4 ~ IRQ2、IRQ0 は非同期信号です。ここに示されたセットアップ時間が守られた場合、クロックの立ち上がりで変化が検出されます。セットアップ時間が守られない場合、次のクロックの立ち上がりエッジまで検出が遅れることがあります。

また、NMI およびエッジ検出の場合の IRQ4 ~ IRQ2 および IRQ0 は、周辺モジュールクロック(P)で 2 サイクル以上のパルス幅が必要です。

*2 外部バスクロックの上限周波数は 60MHz です。

*3 t_{cyt} は外部バスクロック (B) の 1 サイクル時間を示します。

*4 BOOT-E は、RESETP がローアサートされている期間中 (リセット期間中) 以外では変化させないでください。リセット期間以外で入力を変化させた場合、動作の保証はできません。

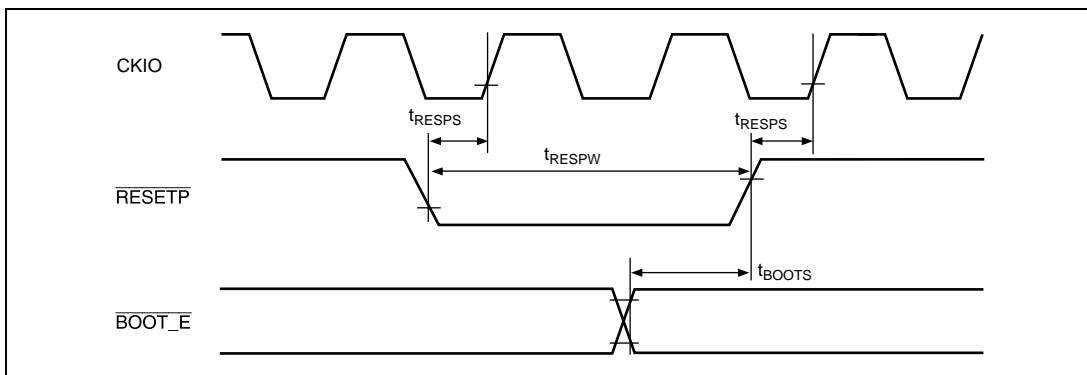


図 28.12 リセットおよび $\overline{\text{BOOT_E}}$ 入力タイミング

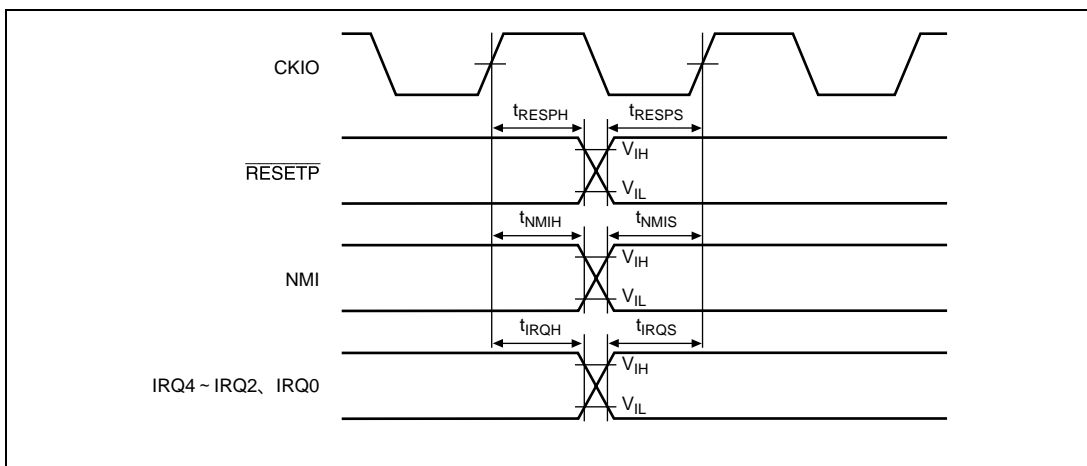


図 28.13 割り込み信号入力タイミング

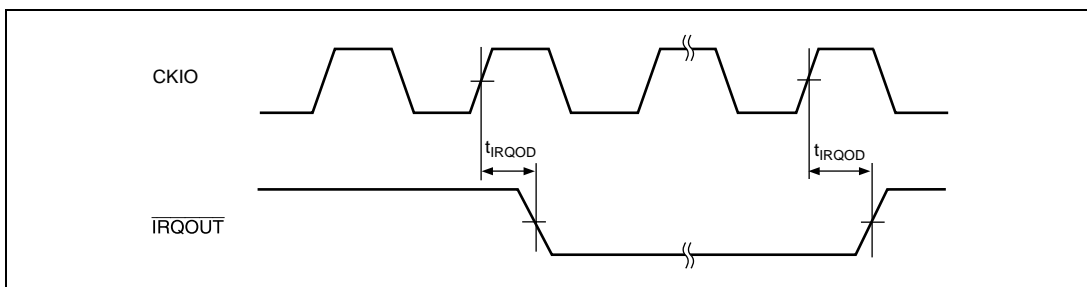


図 28.14 $\overline{\text{IRQOUT}}$ タイミング

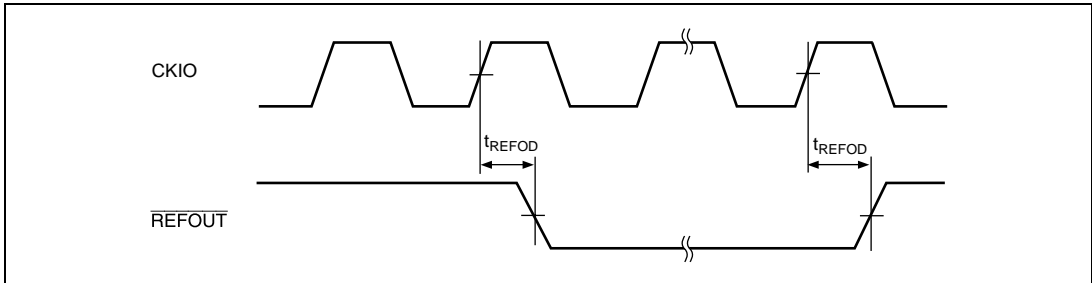


図 28.15 REFOUT タイミング

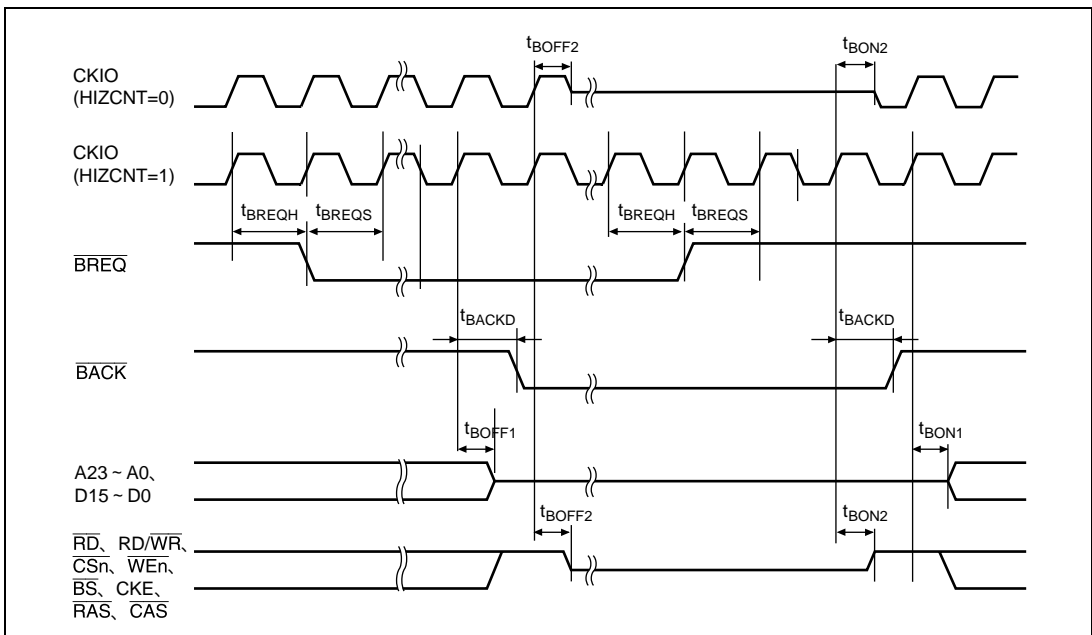


図 28.16 バス権解放タイミング

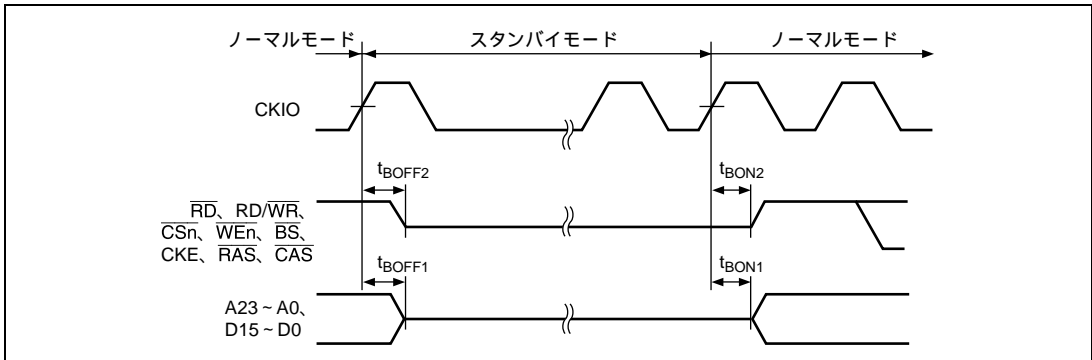


図 28.17 スタンバイ時の端子ドライブタイミング

28. 電気的特性

28.3.3 AC バスタイミング仕様

表 28.7 バスタイミング

条件：クロック動作モード 5、6、7*1、 $V_{CC}=2.7\sim 3.6V$ 、 $V_{CC-28}=2.7\sim 3.0V$ 、 $AV_{CC}=2.7\sim 3.6V$ 、 $V_{DD}=1.4\sim 1.6V$ 、 $T_a=-40\sim 85$

項目	記号	60MHz*2		単位	参照図
		Min.	Max.		
アドレス遅延時間 1	t_{AD1}	1	13	ns	28.18 ~ 28.42、28.47、28.48
アドレス遅延時間 2	t_{AD2}	$1/2t_{cyc}$	$1/2t_{cyc} + 13$	ns	28.25
アドレス遅延時間 3	t_{AD3}	$1/2t_{cyc}$	$1/2t_{cyc} + 13$	ns	28.43 ~ 28.46
アドレスセットアップ時間	t_{AS}	0	-	ns	28.18 ~ 28.21、28.25
アドレスホールド時間	t_{AH}	0	-	ns	28.18 ~ 28.24
BS 遅延時間	t_{BSD}	-	13	ns	28.18 ~ 28.43、28.47、28.48
CS 遅延時間 1	t_{CSD1}	1	13	ns	28.18 ~ 28.42、28.47、28.48
CS 遅延時間 2	t_{CSD2}	$1/2t_{cyc}$	$1/2t_{cyc} + 13$	ns	28.43 ~ 28.46
リードライト遅延時間 1	t_{RWD1}	1	13	ns	28.18 ~ 28.42、28.47、28.48
リードライト遅延時間 2	t_{RWD2}	$1/2t_{cyc}$	$1/2t_{cyc} + 13$	ns	28.43 ~ 28.46
リードストロブ遅延時間	t_{RSD}	$1/2t_{cyc}$	$1/2t_{cyc} + 13$	ns	28.18 ~ 28.25
リードデータセットアップ時間 1	t_{RDS1}	$1/2t_{cyc} + 10$	-	ns	28.18 ~ 28.24
リードデータセットアップ時間 2	t_{RDS2}	7	-	ns	28.26 ~ 28.29、28.34 ~ 28.36、28.47、28.48
リードデータセットアップ時間 3	t_{RDS3}	$1/2t_{cyc} + 10$	-	ns	28.25
リードデータセットアップ時間 4	t_{RDS4}	$1/2t_{cyc} + 10$	-	ns	28.43
リードデータホールド時間 1	t_{RDH1}	0	-	ns	28.18 ~ 28.24
リードデータホールド時間 2	t_{RDH2}	2	-	ns	28.26 ~ 28.29、28.34 ~ 28.36、28.47、28.48
リードデータホールド時間 3	t_{RDH3}	0	-	ns	28.25
リードデータホールド時間 4	t_{RDH4}	0	-	ns	28.43
ライトイネーブル遅延時間 1	t_{WED1}	$1/2t_{cyc}$	$1/2t_{cyc} + 13$	ns	28.18 ~ 28.23
ライトイネーブル遅延時間 2	t_{WED2}	-	13	ns	28.24
ライトデータ遅延時間 1	t_{WDD1}	-	13	ns	28.18 ~ 28.24
ライトデータ遅延時間 2	t_{WDD2}	-	13	ns	28.30 ~ 28.33、28.37 ~ 28.39、28.47、28.48
ライトデータ遅延時間 3	t_{WDD3}	-	$1/2t_{cyc} + 13$	ns	28.43
ライトデータホールド時間 1	t_{WDH1}	1	-	ns	28.18 ~ 28.24
ライトデータホールド時間 2	t_{WDH2}	1	-	ns	28.30 ~ 28.33、28.37 ~ 28.39、28.47、28.48
ライトデータホールド時間 3	t_{WDH3}	$1/2t_{cyc}$	-	ns	28.43
WAIT セットアップ時間	t_{WTS1}	$1/2t_{cyc} + 7$	-	ns	28.18 ~ 28.25

項目	記号	60MHz* ²		単位	参照図
		Min.	Max.		
WAIT ホールド時間	t_{WITH1}	$1/2t_{\text{cyc}} + 3$	-	ns	28.18 ~ 28.25
$\overline{\text{RAS}}$ 遅延時間 1	t_{RASD1}	1	13	ns	28.26 ~ 28.42、28.47、28.48
$\overline{\text{RAS}}$ 遅延時間 2	t_{RASD2}	$1/2t_{\text{cyc}}$	$1/2t_{\text{cyc}} + 13$	ns	28.43 ~ 28.46
$\overline{\text{CAS}}$ 遅延時間 1	t_{CASD1}	1	13	ns	28.26 ~ 28.42、28.47、28.48
$\overline{\text{CAS}}$ 遅延時間 2	t_{CASD2}	$1/2t_{\text{cyc}}$	$1/2t_{\text{cyc}} + 13$	ns	28.43 ~ 28.46
DQM 遅延時間 1	t_{DQMD1}	1	13	ns	28.26 ~ 28.42、28.47、28.48
DQM 遅延時間 2	t_{DQMD2}	$1/2t_{\text{cyc}}$	$1/2t_{\text{cyc}} + 13$	ns	28.43 ~ 28.46
CKE 遅延時間 1	t_{CKED1}	1	13	ns	28.41、28.42、28.47、28.48
CKE 遅延時間 2	t_{CKED2}	$1/2t_{\text{cyc}}$	$1/2t_{\text{cyc}} + 13$	ns	28.45

【注】 *1 SDRAM インタフェースのタイミングは、クロック動作モード 7 での値です。SDRAM インタフェースは、クロック動作モード 7 での使用を推奨します。

*2 外部バスクロックは 60MHz が上限です。

28.3.4 基本タイミング

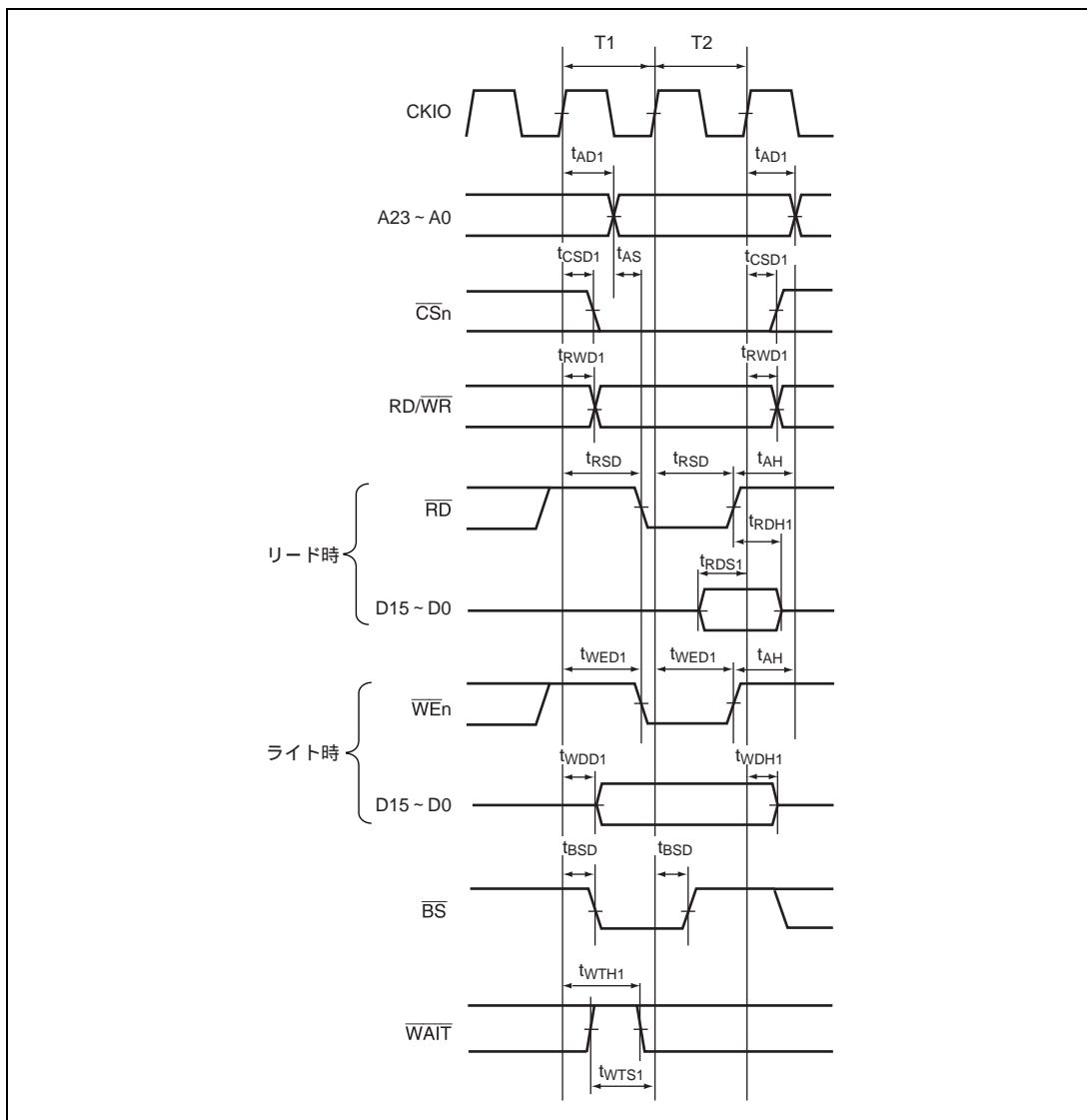


図 28.18 通常空間基本バスサイクル (ノーウェイト)

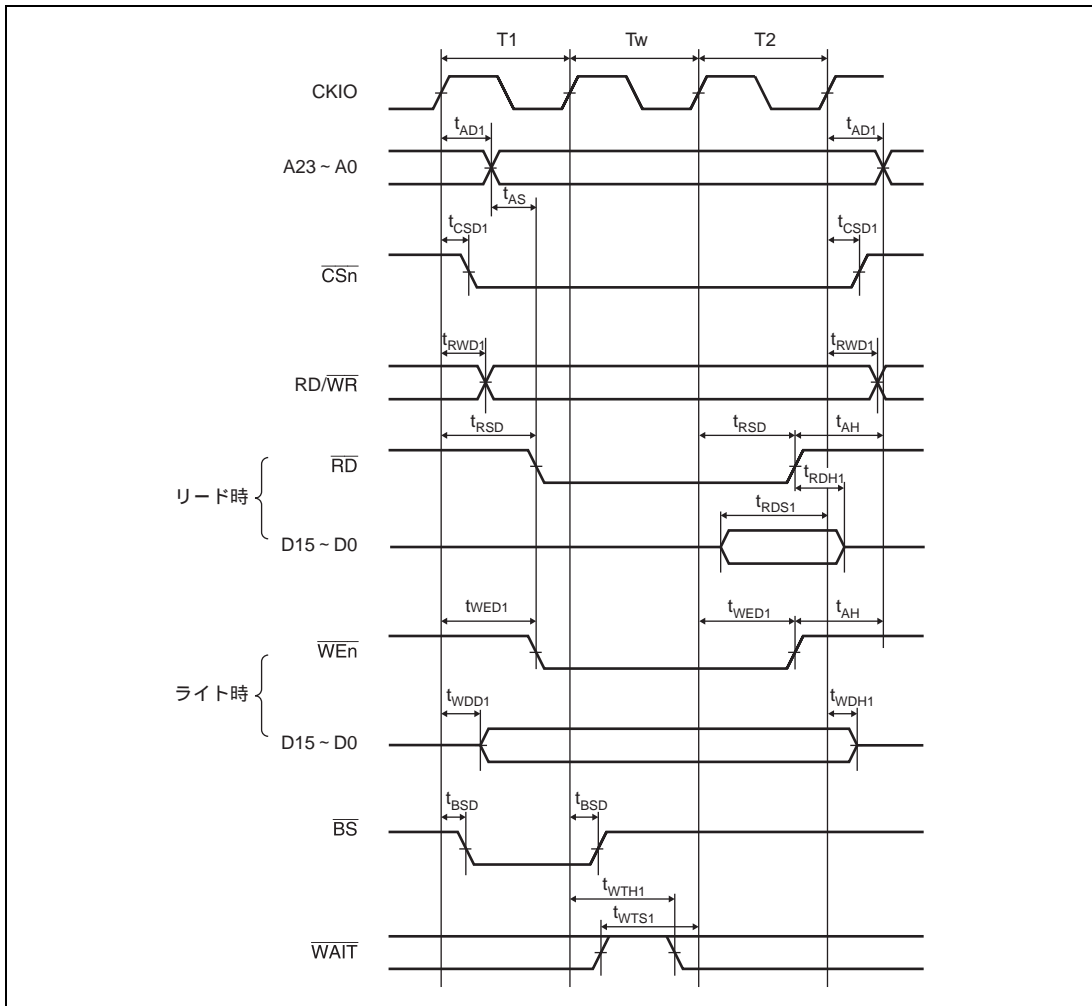


図 28.19 通常空間基本バスサイクル (ソフトウェアウェイト 1)

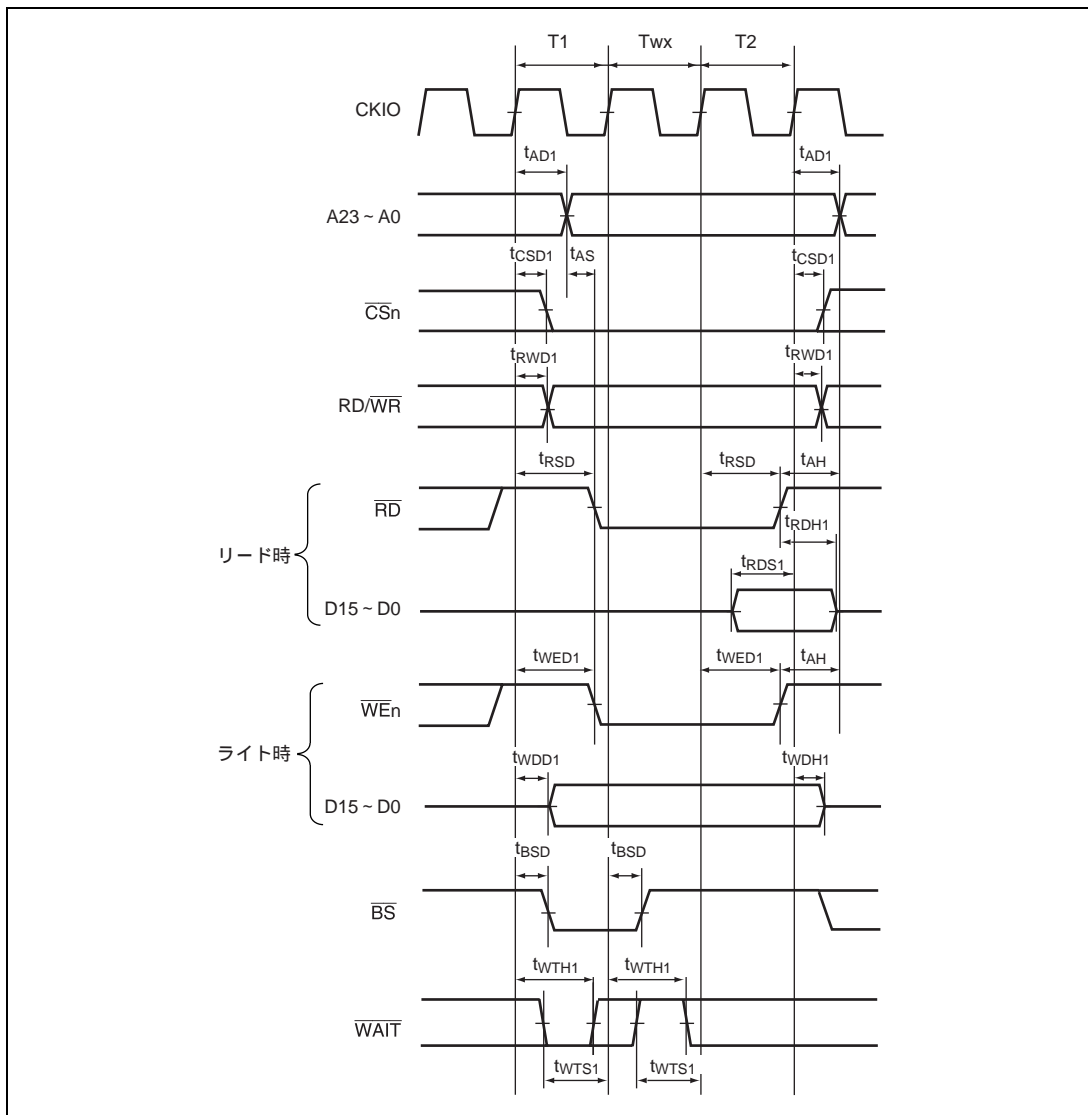


図 28.20 通常空間基本バスサイクル (非同期外部ウェイト 1 挿入)

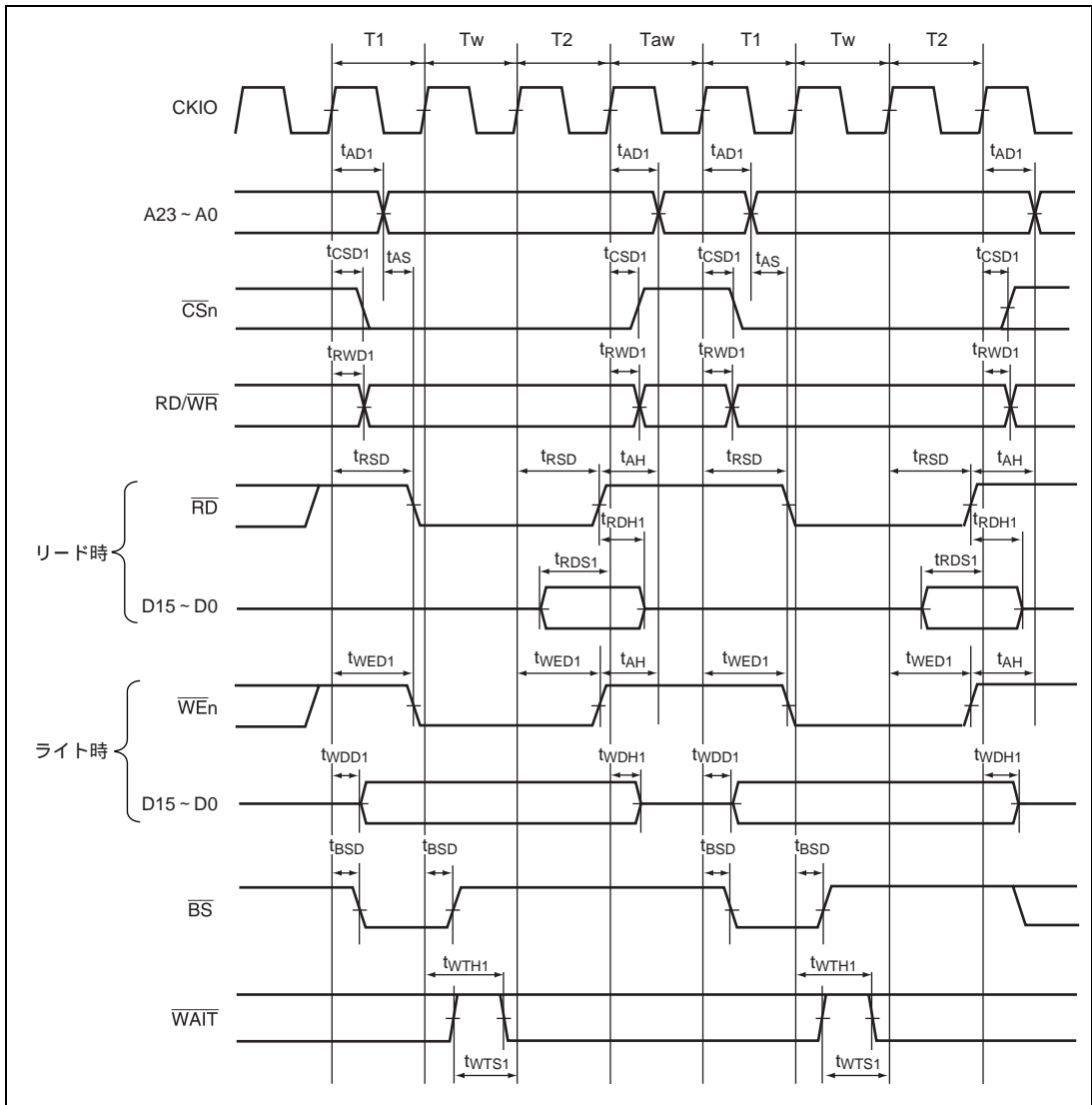


図 28.21 通常空間基本バスサイクル
 (ソフトウェアウェイト1、非同期外部ウェイト有効(WMビット=0)、アイドルサイクルなし設定)

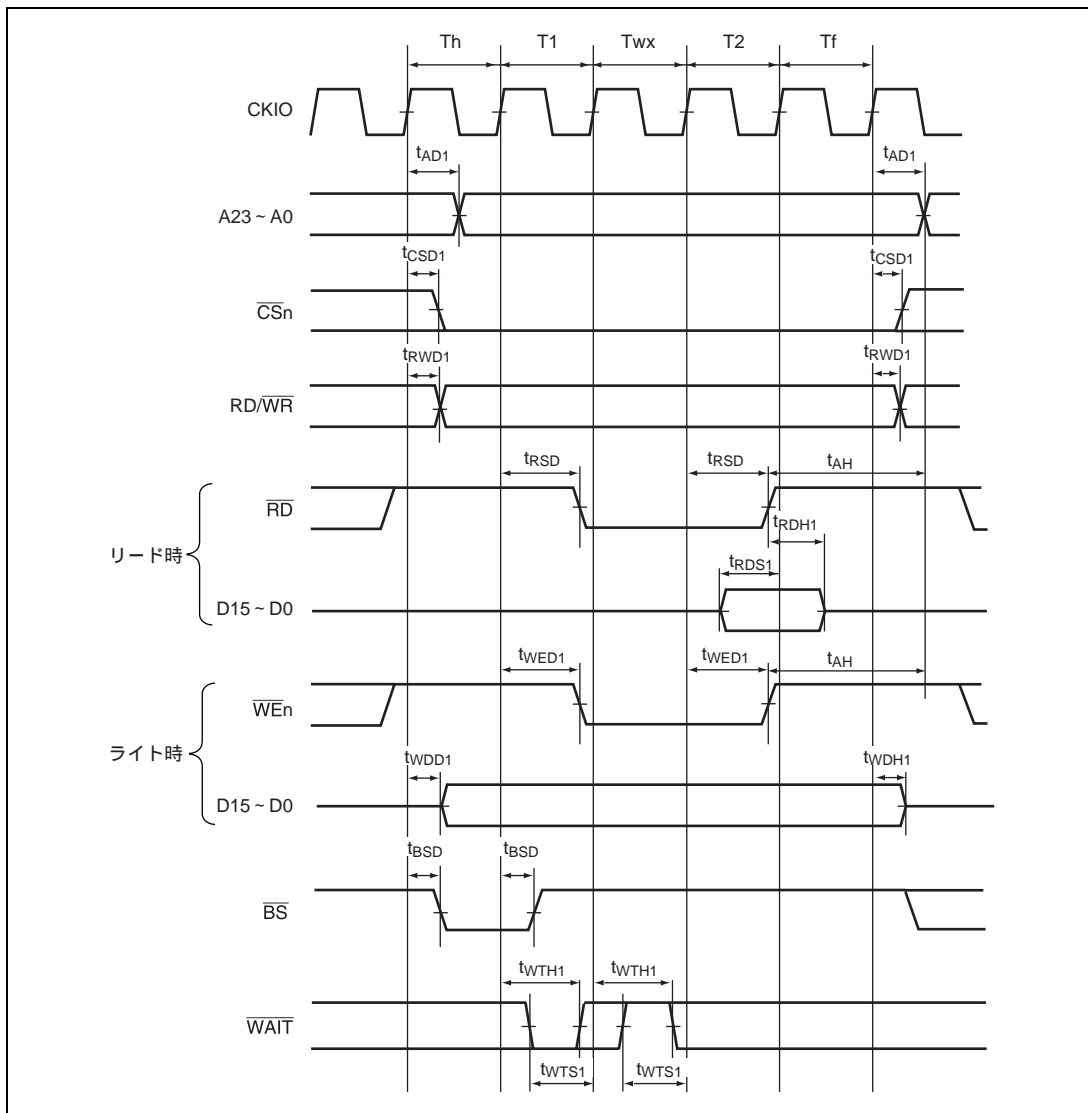


図 28.22 通常空間 CS 拡張バスサイクル
 (SW=1 サイクル、HW=1 サイクル、非同期外部ウェイト 1 挿入)

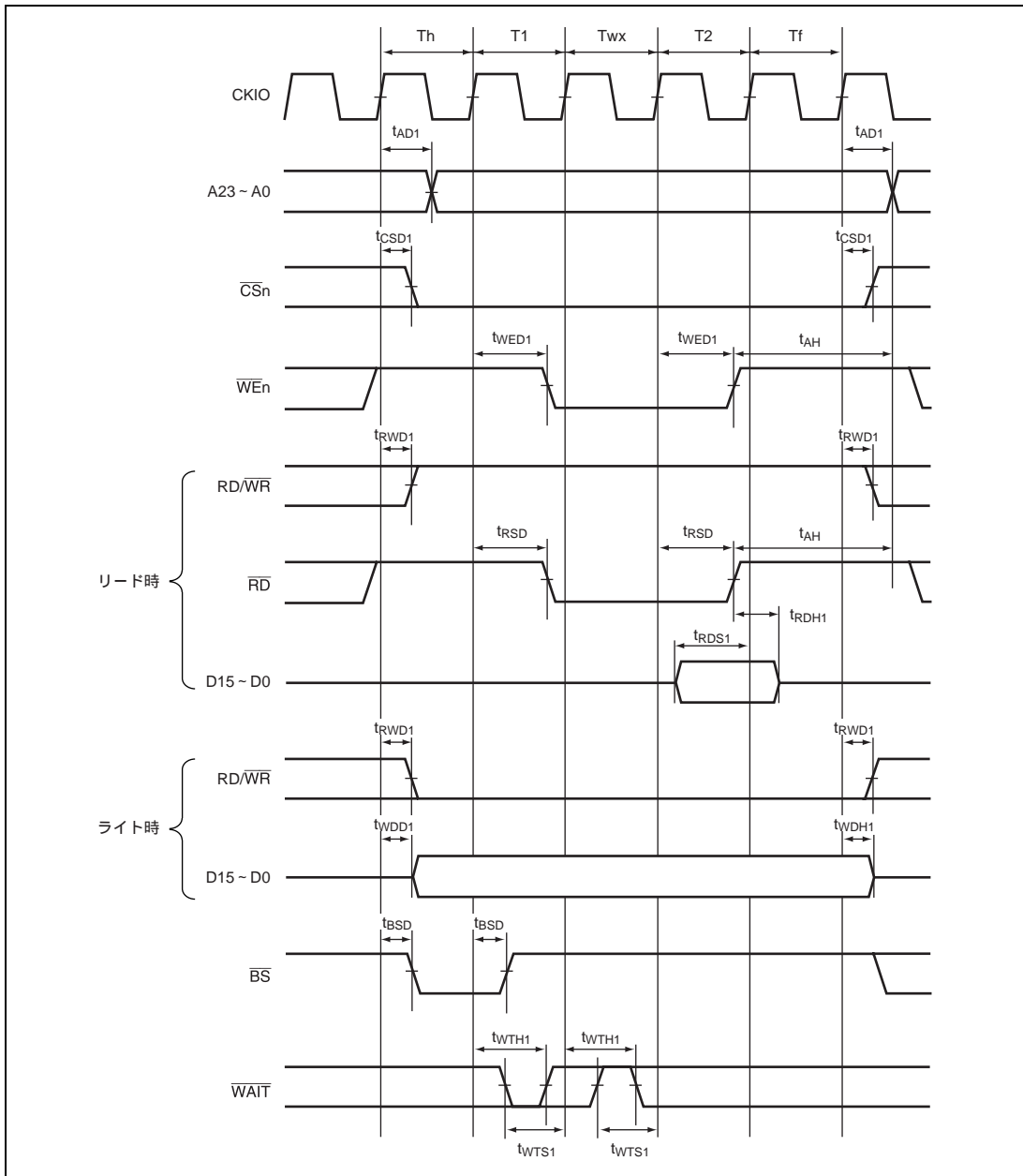


図 28.23 バイト選択付き SRAM バスサイクル
 (SW = 1 サイクル、HW = 1 サイクル、非同期外部ウェイト 1 挿入、
 BAS = 0 (ライトサイクル UB、LB コントロール))

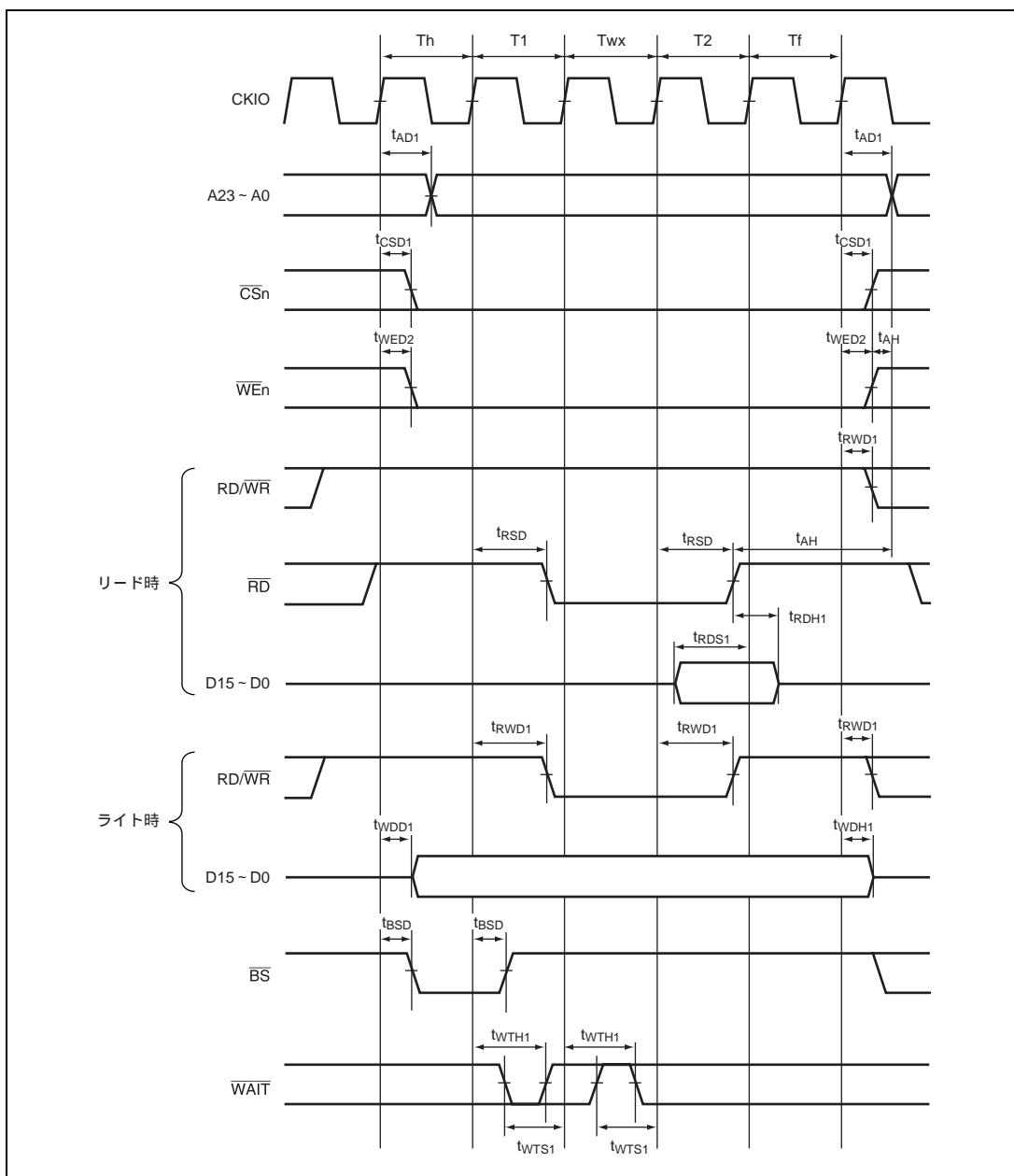


図 28.24 バイト選択付き SRAM バスサイクル
 (SW = 1 サイクル、HW = 1 サイクル、非同期外部ウェイト 1 挿入、
 BAS = 1 (ライトサイクル WE コントロール))

28.3.5 バースト ROM タイミング

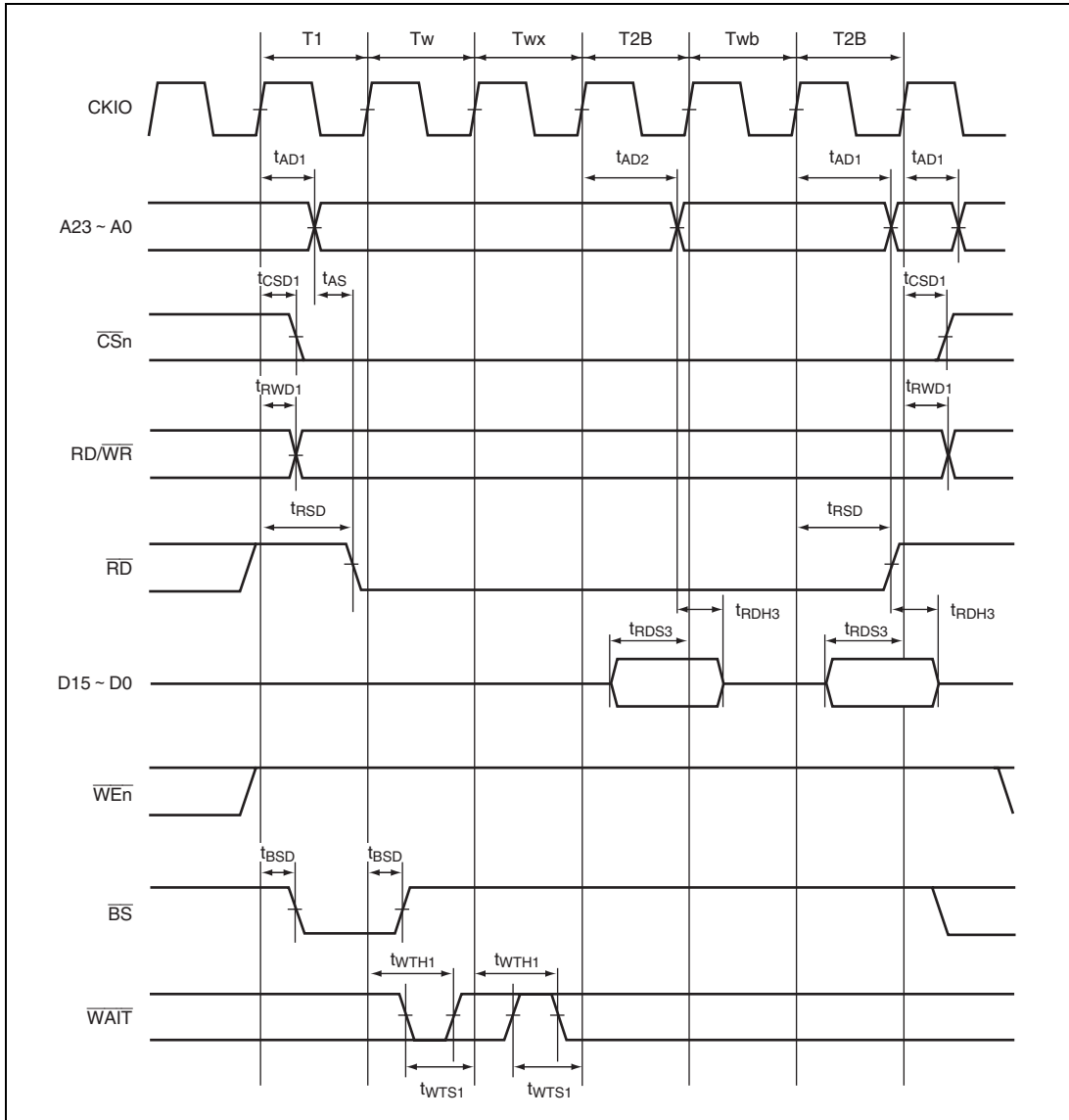


図 28.25 バースト ROM リードバスサイクル
(ソフトウェアウェイト 1、非同期外部ウェイト 1 挿入、バーストウェイト 1、2 バースト)

28.3.6 シンクロナス DRAM タイミング

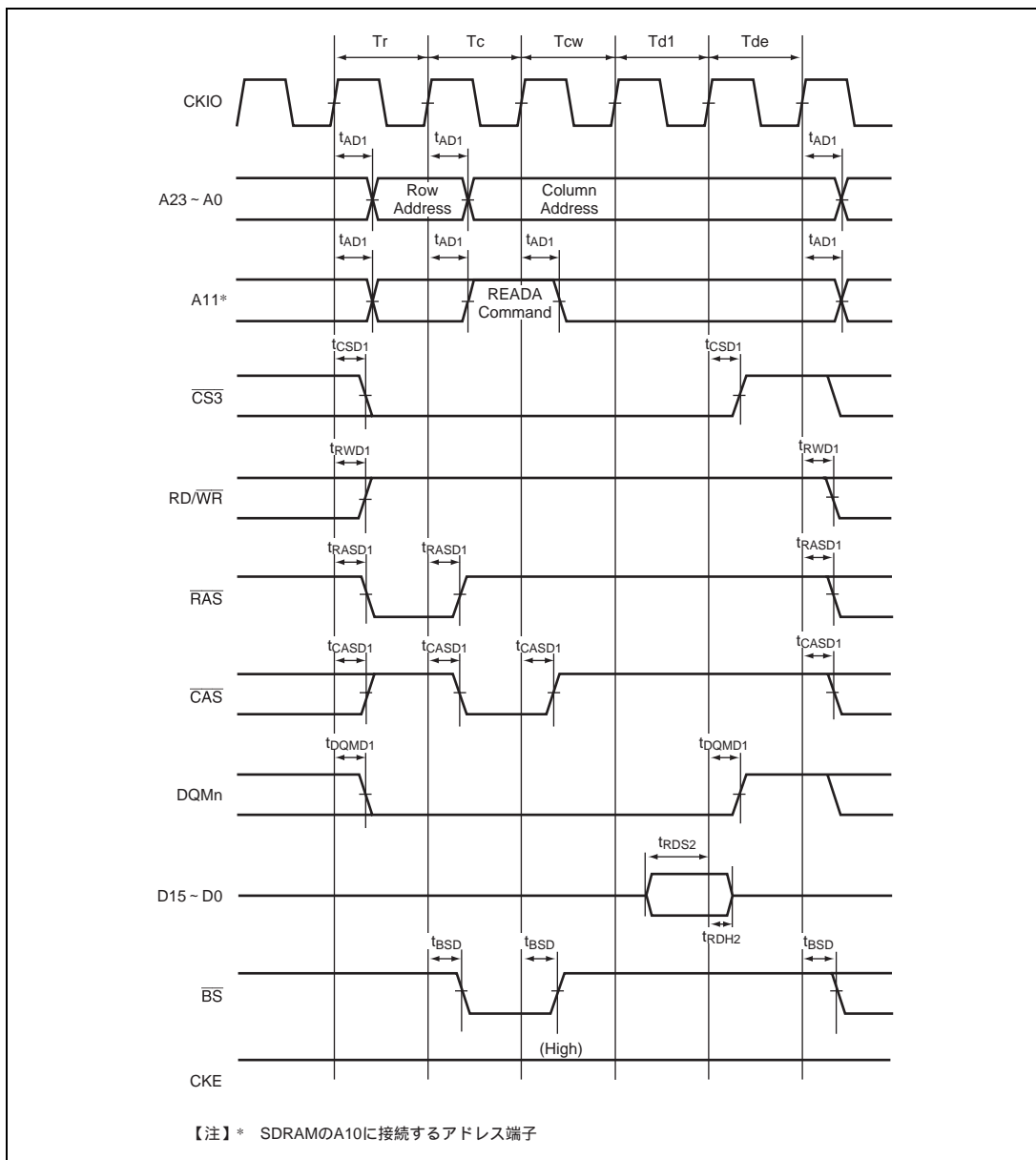


図 28.26 シンクロナス DRAM シングルリードバスサイクル
(オートプリチャージモード、CAS レイテンシ 2、TRCD=1 サイクル、TRP=1 サイクル)

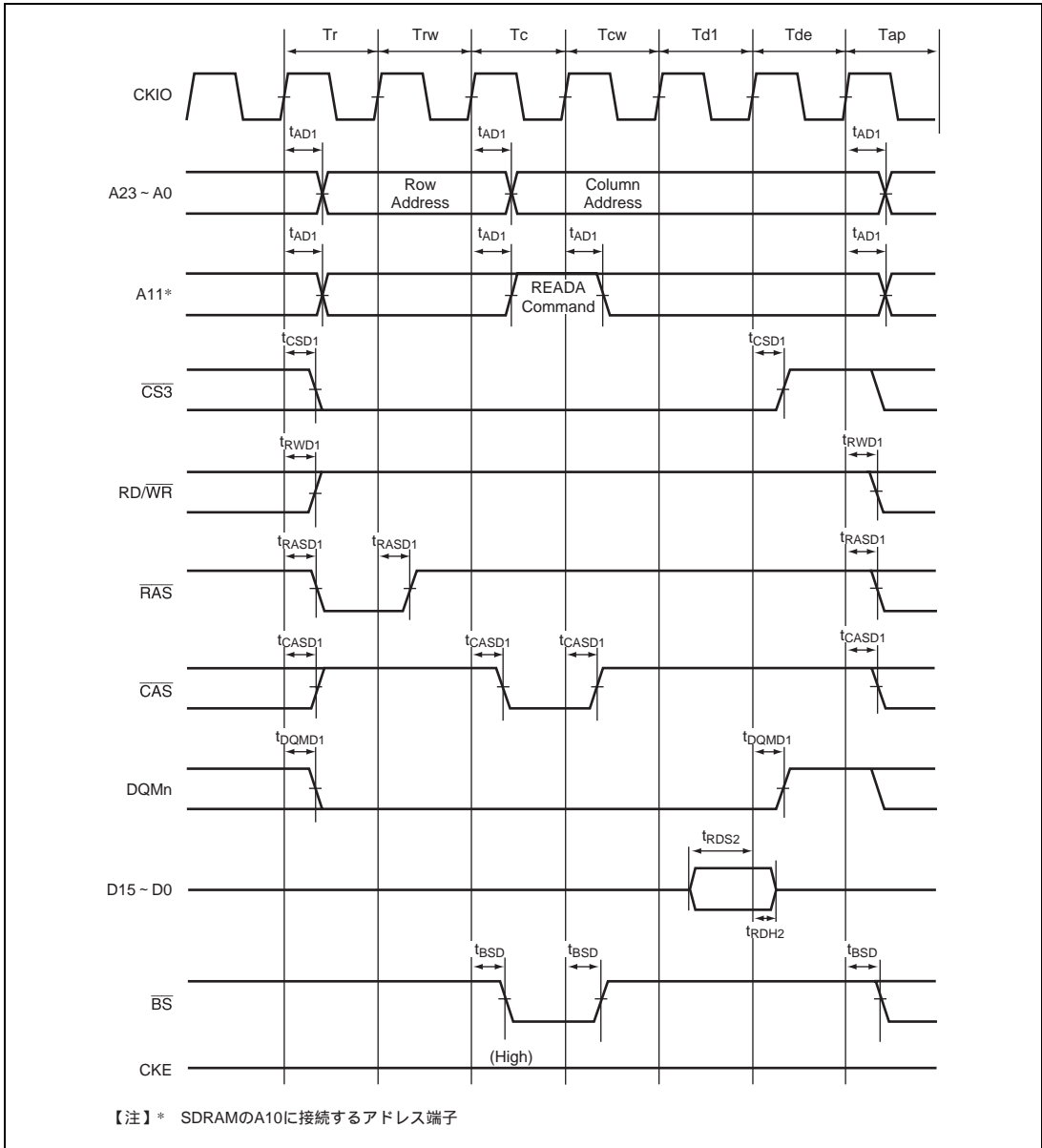


図 28.27 シンクロナス DRAM シングルリードバスサイクル
 (オートプリチャージモード、CAS レイテンシ 2、TRCD = 2 サイクル、TRP = 2 サイクル)

28. 電気的特性

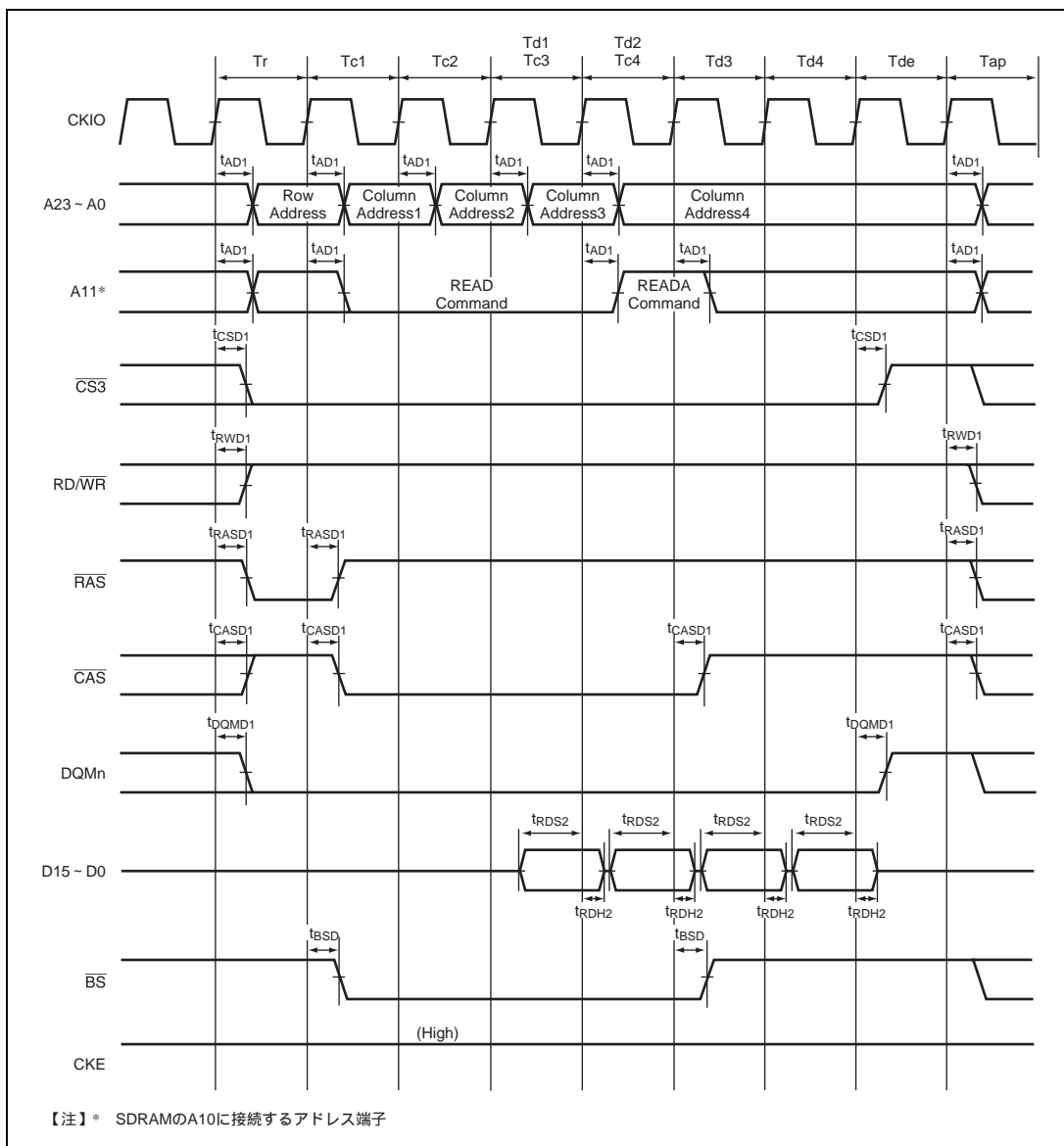


図 28.28 シンクロナス DRAM バーストリードバスサイクル (シングルリード×4)
(オートプリチャージモード、CAS レイテンシ 2、TRCD = 1 サイクル、TRP = 2 サイクル)

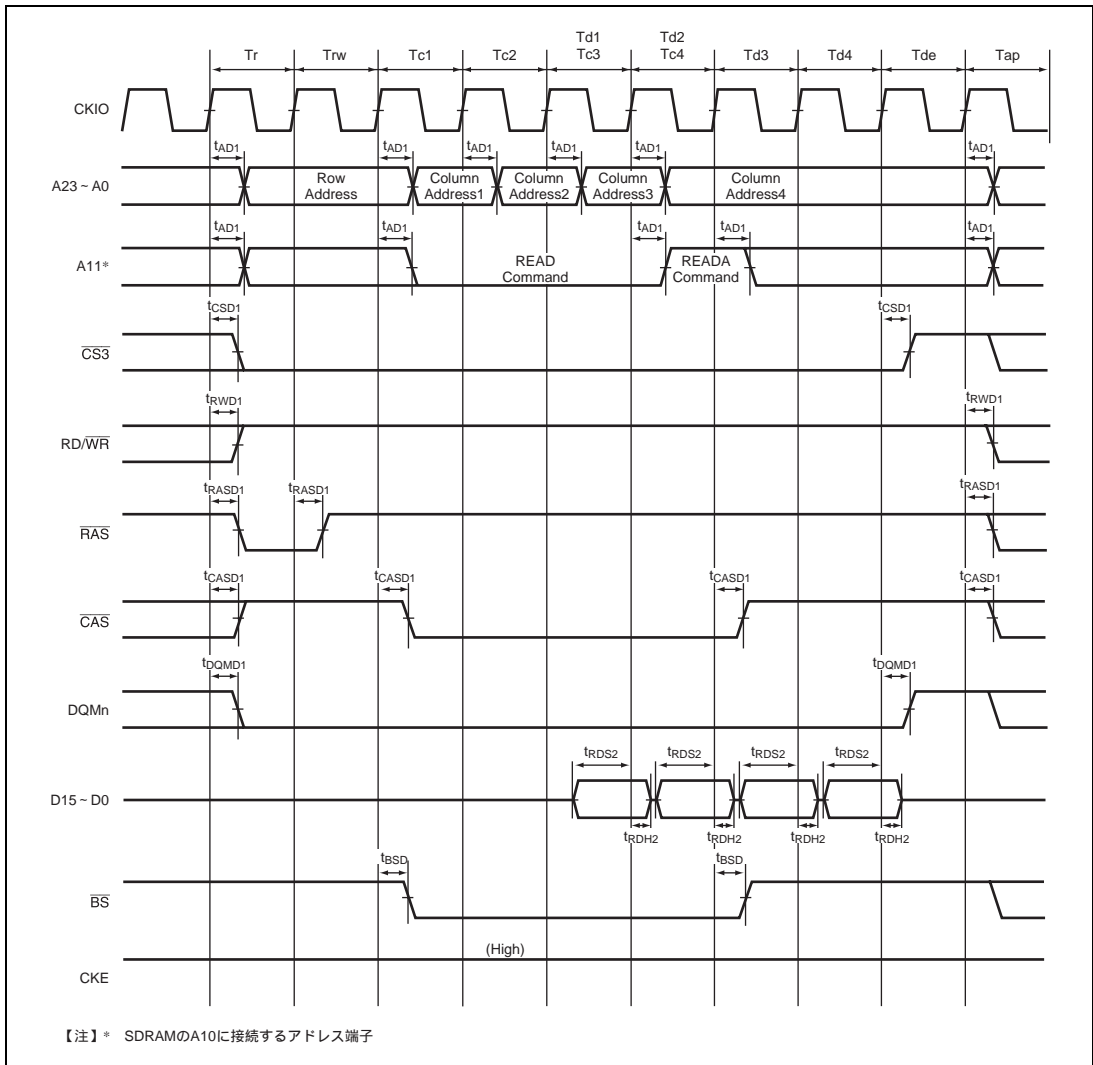


図 28.29 シンクロナス DRAM バーストリードバスサイクル (シングルリード×4)
 (オートプリチャージモード、CAS レイテンシ 2、TRCD = 2 サイクル、TRP = 1 サイクル)

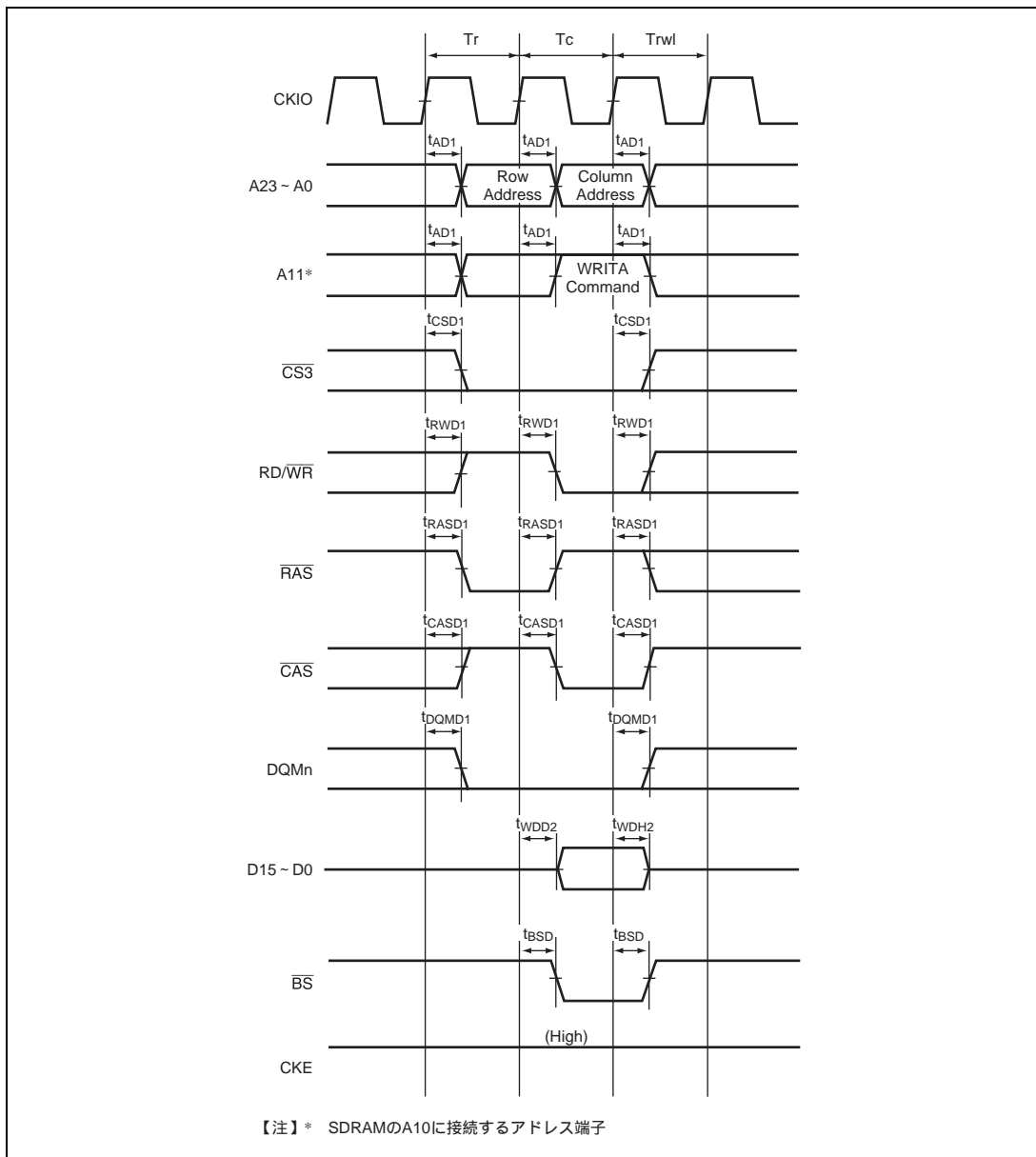


図 28.30 シンクロナス DRAM シングルライトバスサイクル
(オートプリチャージモード、TRWL = 1 サイクル)

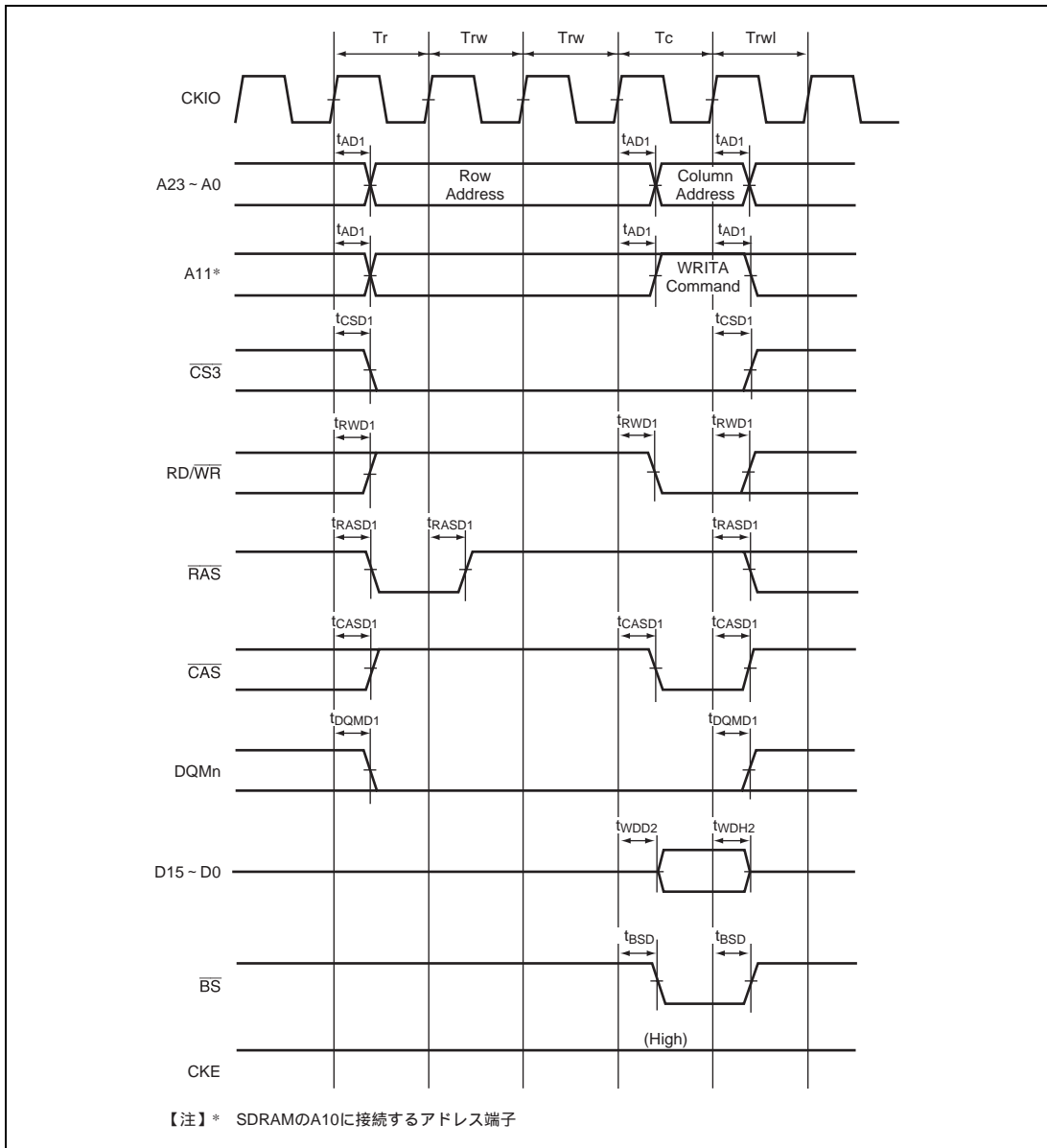


図 28.31 シンクロナス DRAM シングルライトバスサイクル
(オートプリチャージモード、TRCD = 3 サイクル、TRWL = 1 サイクル)

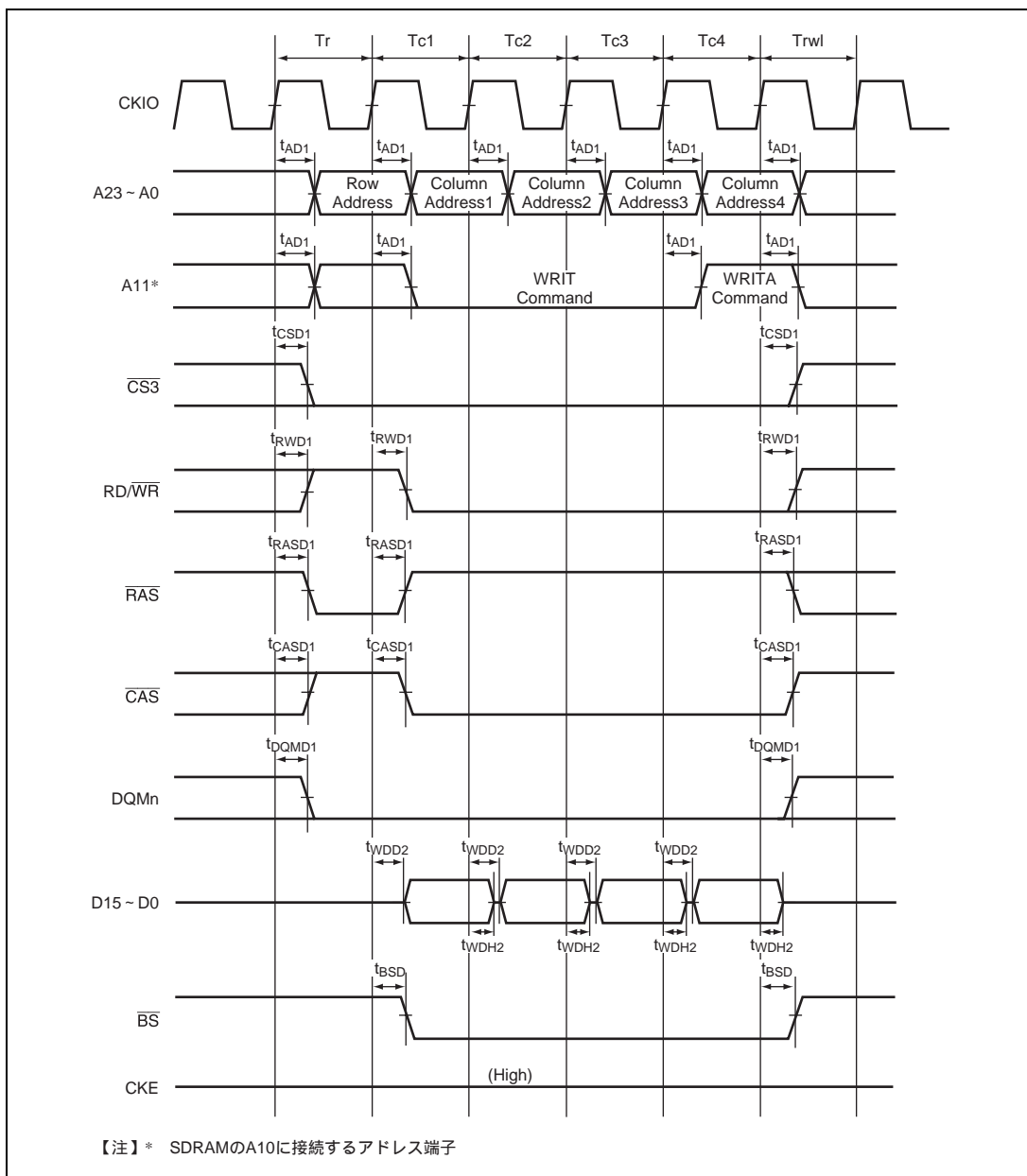


図 28.32 シンクロナス DRAM バーストライトバスサイクル (シングルライト×4)
(オートプリチャージモード、TRCD=1 サイクル、TRWL=1 サイクル)

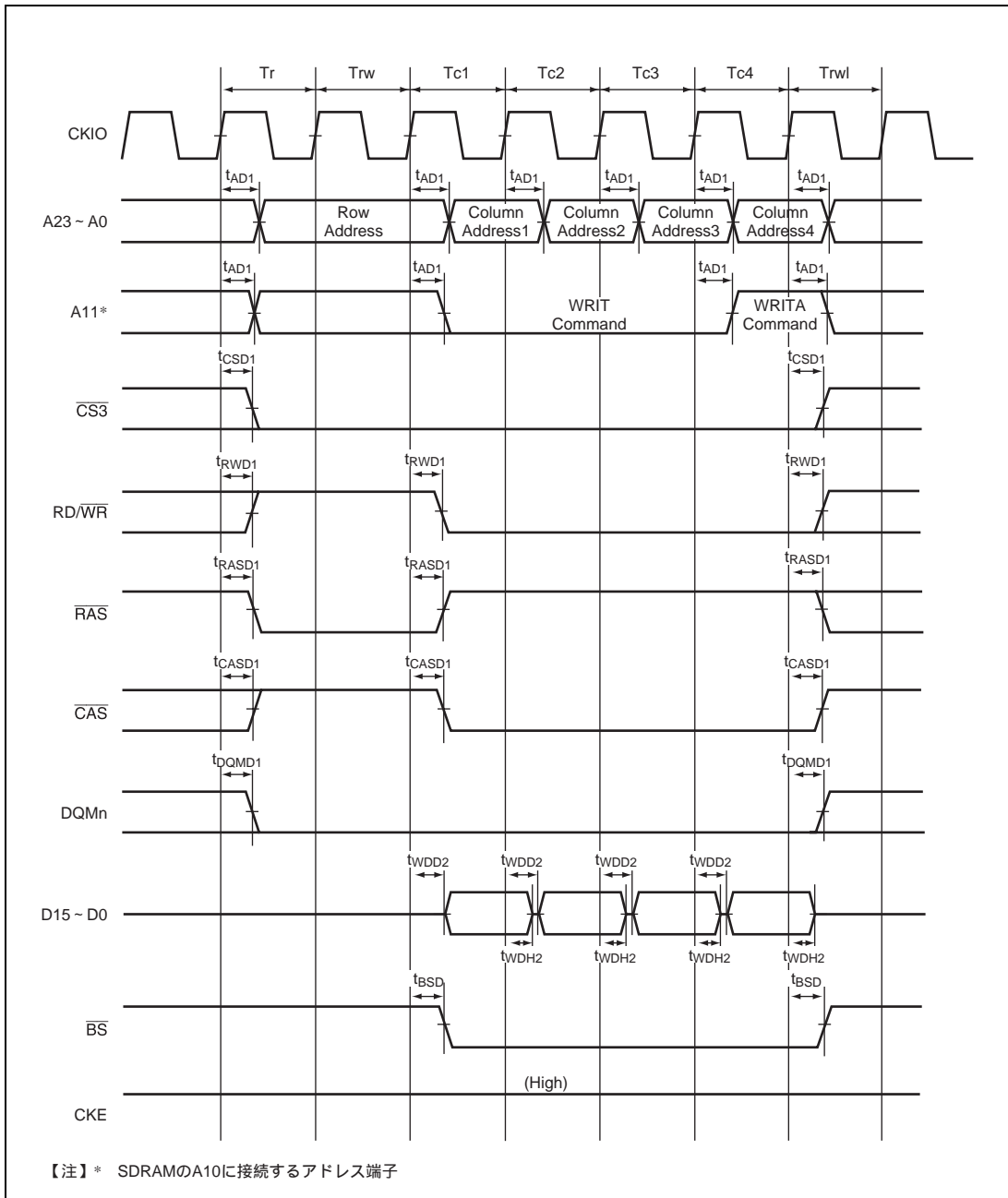


図 28.33 シンクロナス DRAM バーストライトパスサイクル (シングルライト×4)
(オートプリチャージモード、TRCD = 2 サイクル、TRWL = 1 サイクル)

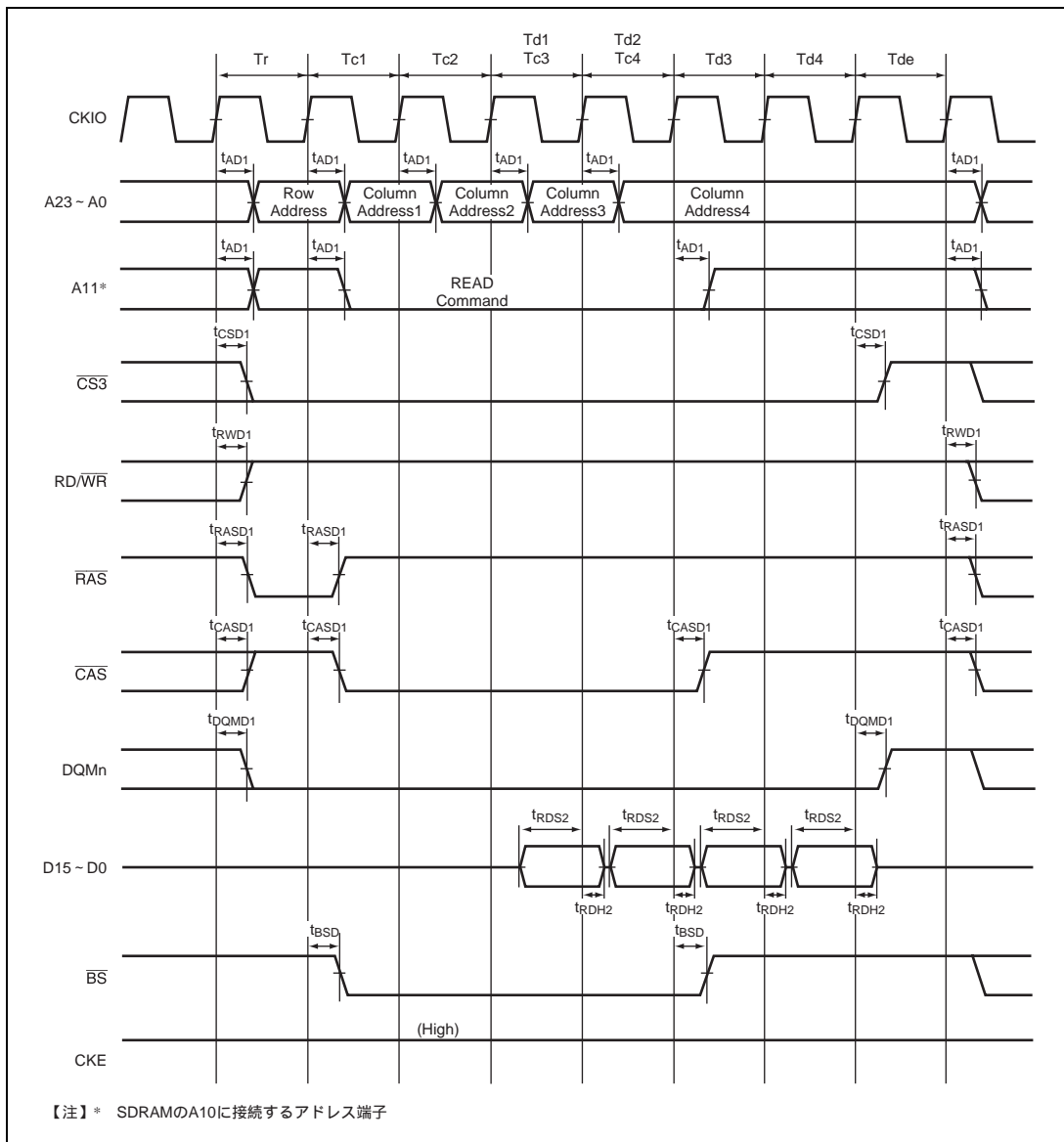


図 28.34 シンクロナス DRAM バーストリードバスサイクル (シングルリード×4)
 (バンクアクティブモード : ACTV+READ コマンド、CAS レイテンシ 2、TRCD = 1 サイクル)

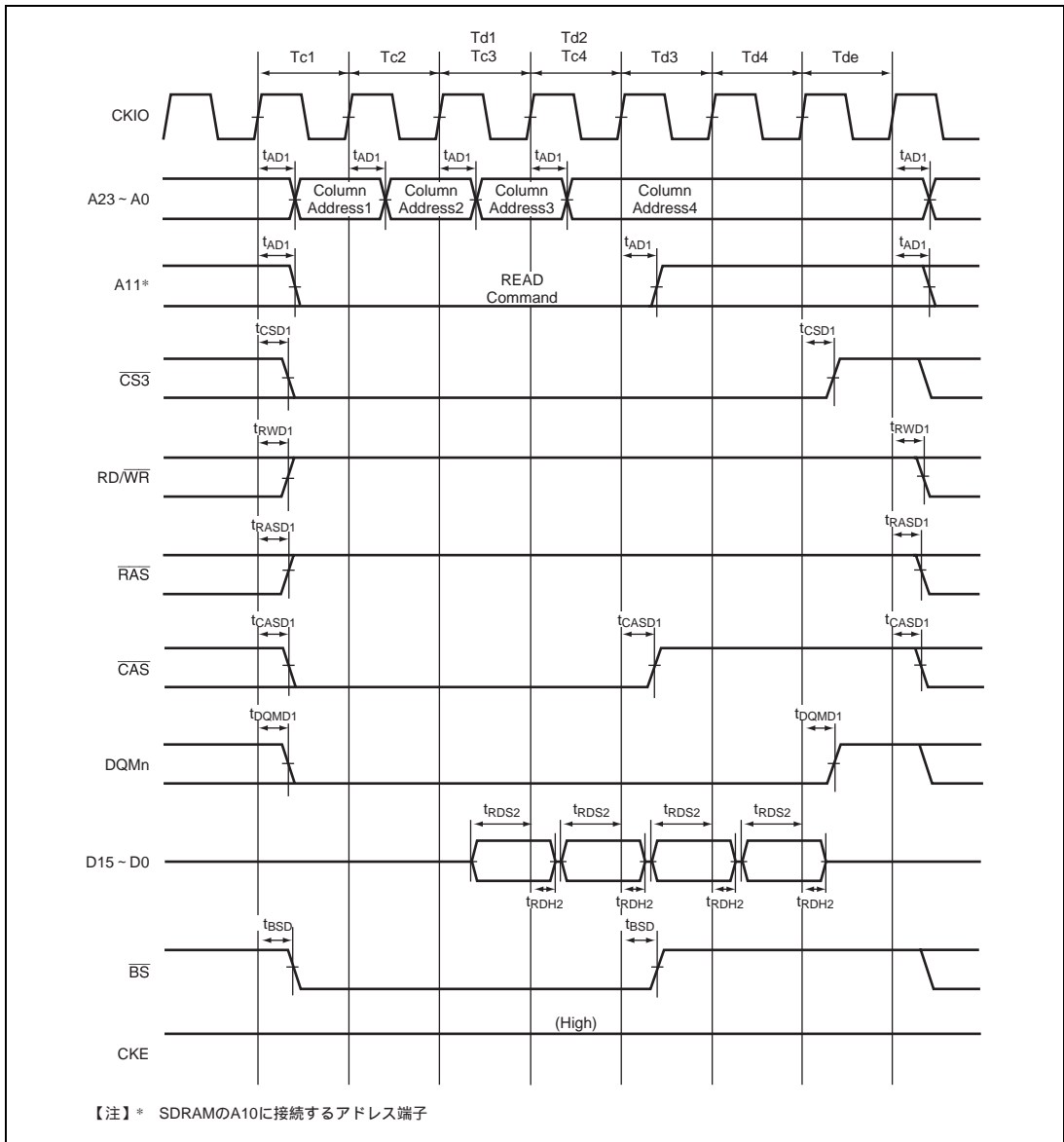


図 28.35 シンクロナス DRAM バーストリードバスサイクル (シングルリード×4)
 (バンクアクティブモード : READ コマンド、同一ロウアドレス、CAS レイテンシ 2、TRCD = 1 サイクル)

28. 電気的特性

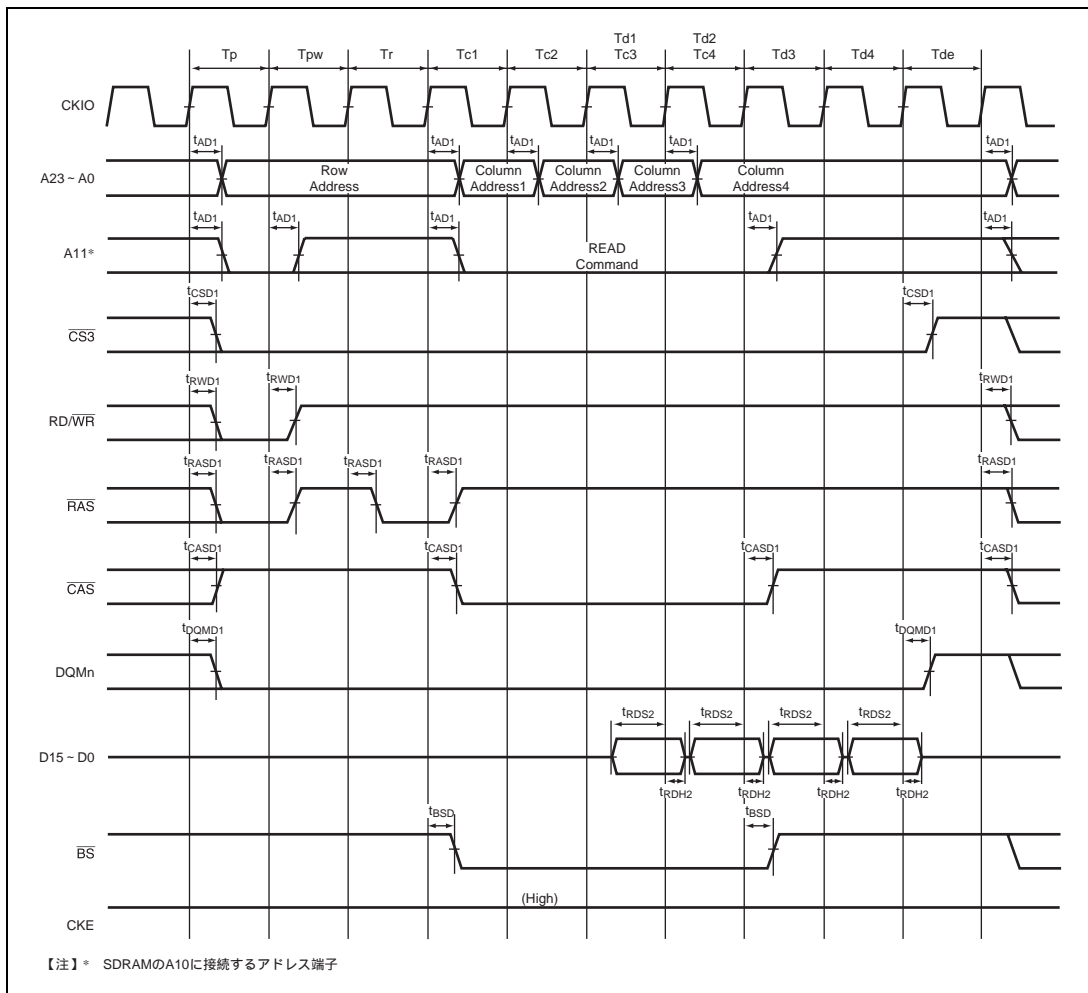


図 28.36 シンクロナス DRAM バーストリードバスサイクル (シングルリード×4)
 (バンクアクティブモード : PRE+ACTV+READ コマンド、異なるロウアドレス、CAS レイテンシ 2、
 TRCD = 1 サイクル)

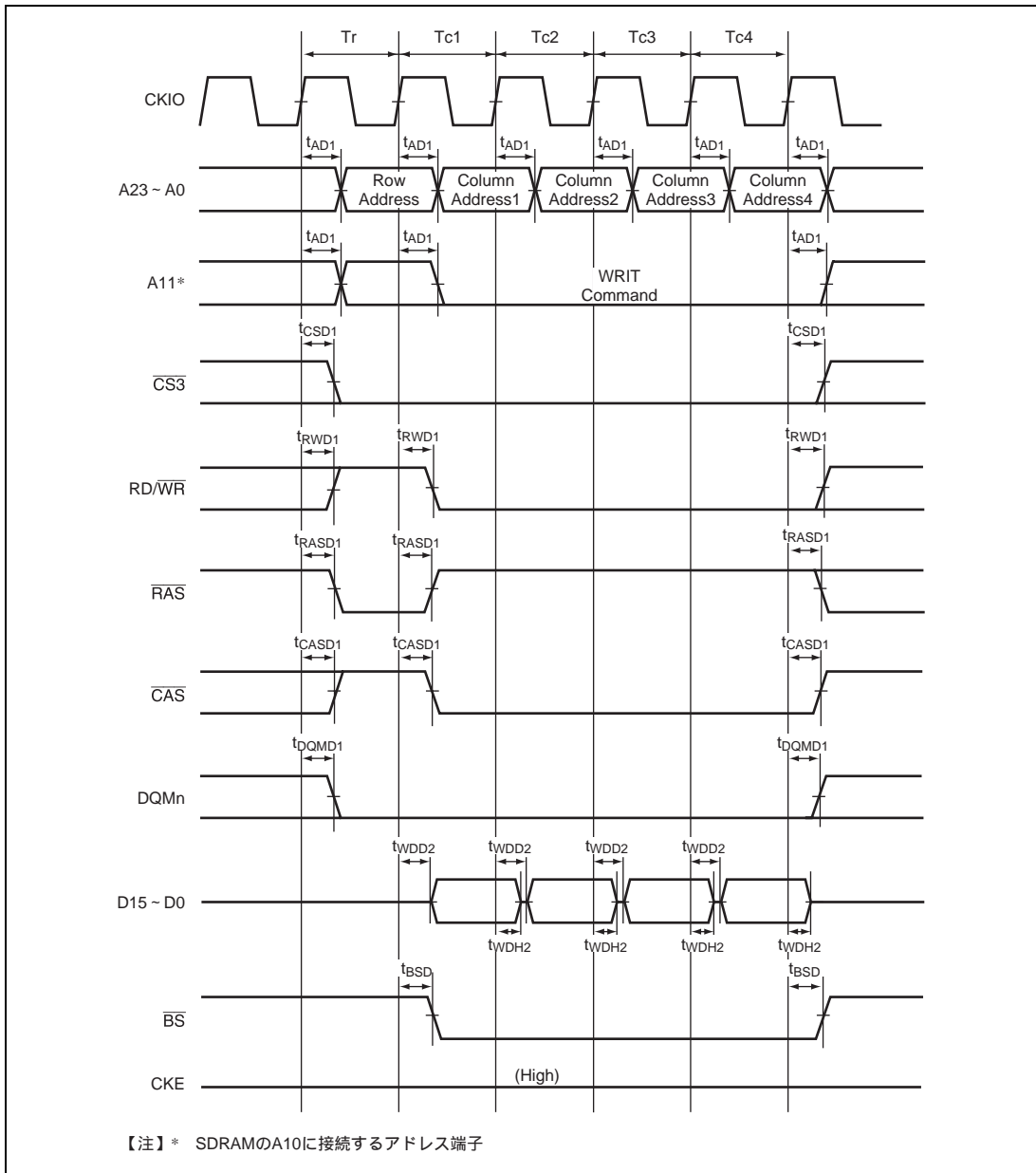


図 28.37 シンクロナス DRAM パーストライトバスサイクル (シングルライト×4)
(バンクアクティブモード、ACTV+WRIT コマンド、TRCD=1 サイクル)

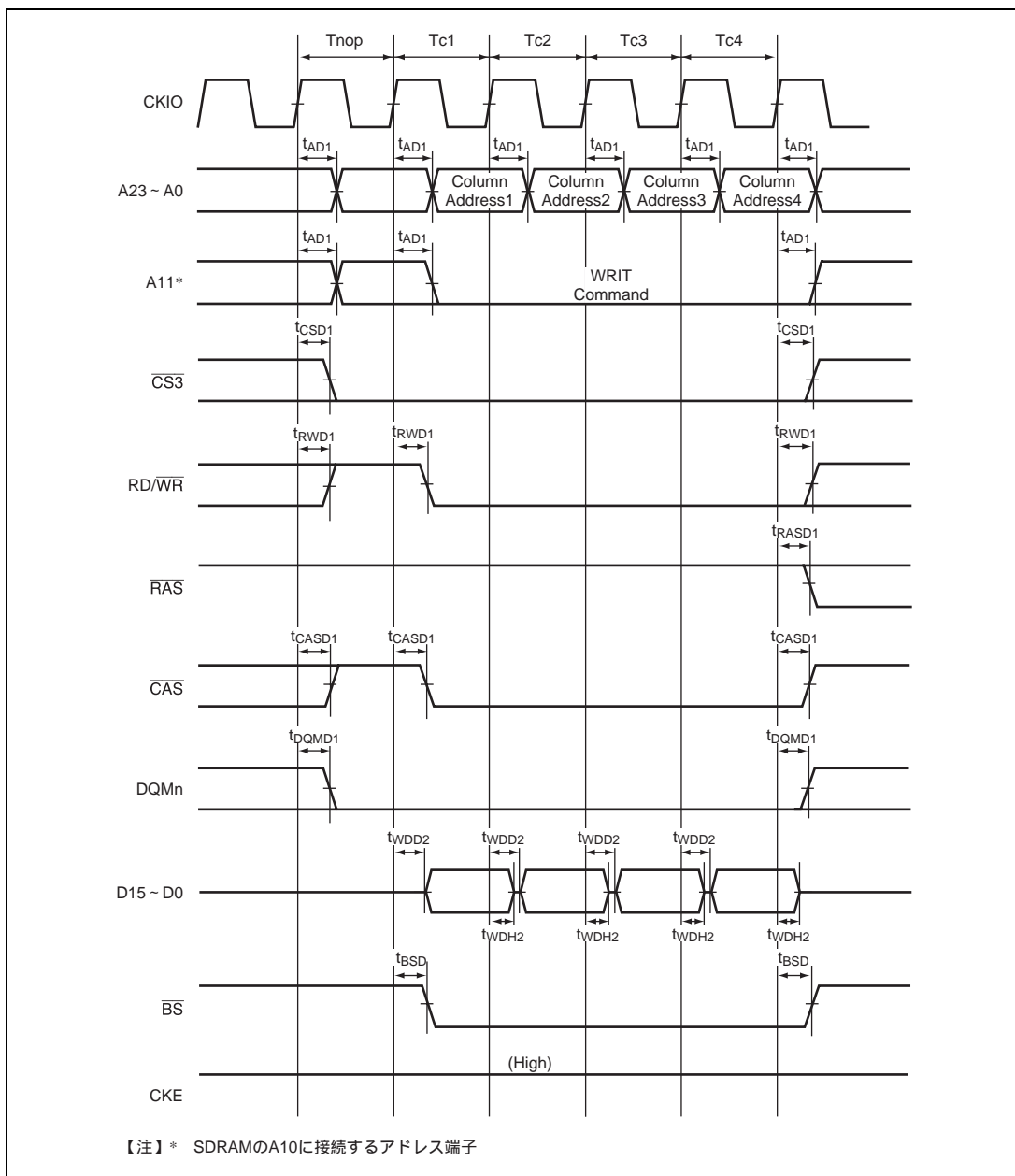


図 28.38 シンクロナス DRAM パーストライトバスサイクル (シングルライト×4)
(バンクアクティブモード、ACTV+WRIT コマンド、TRCD=1 サイクル)

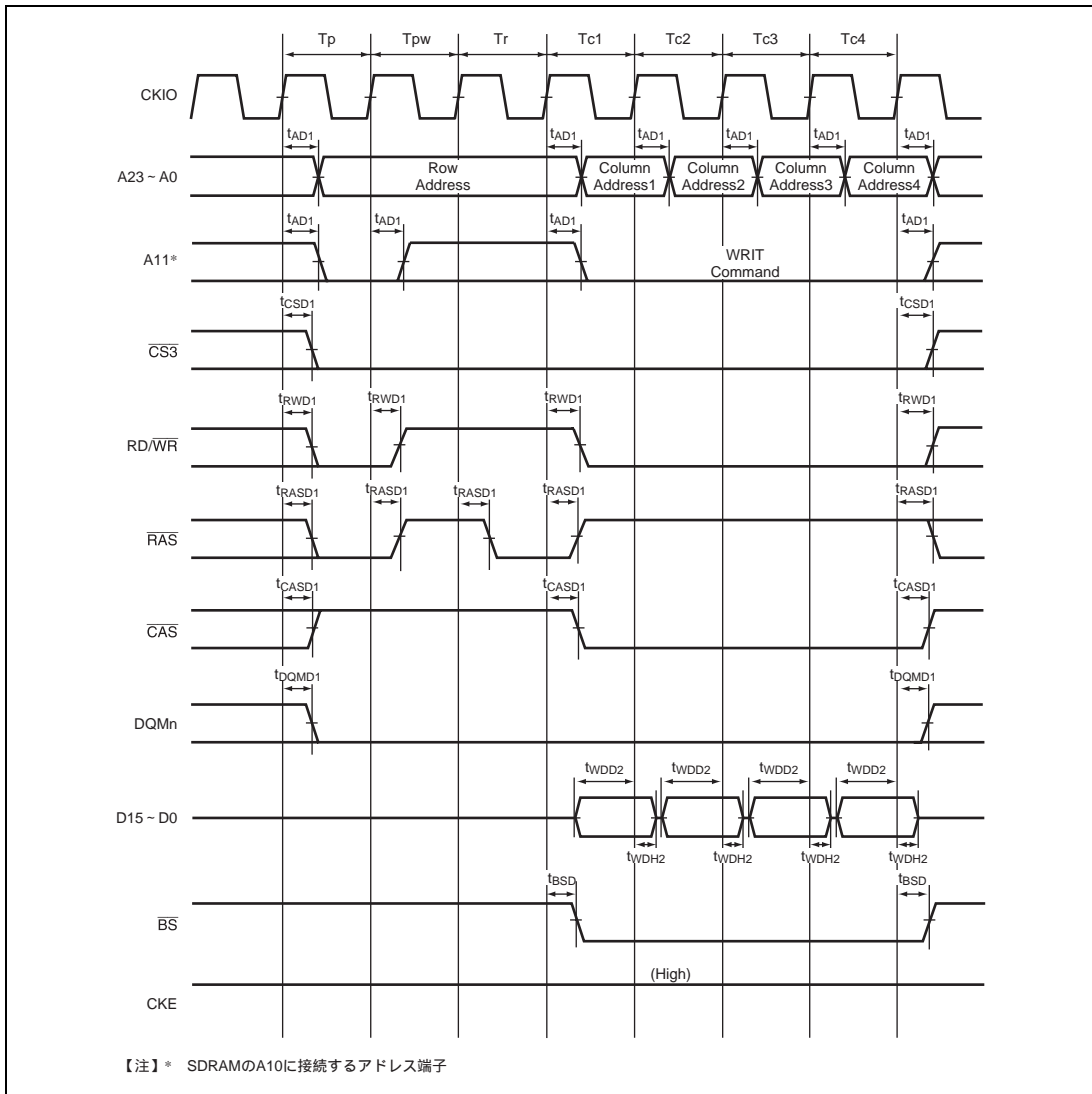


図 28.39 シンクロナス DRAM パーストライトバスサイクル (シングルライト×4)
(バンクアクティブモード、PRE+ACTV+WRIT コマンド、TRCD = 1 サイクル)

28. 電気的特性

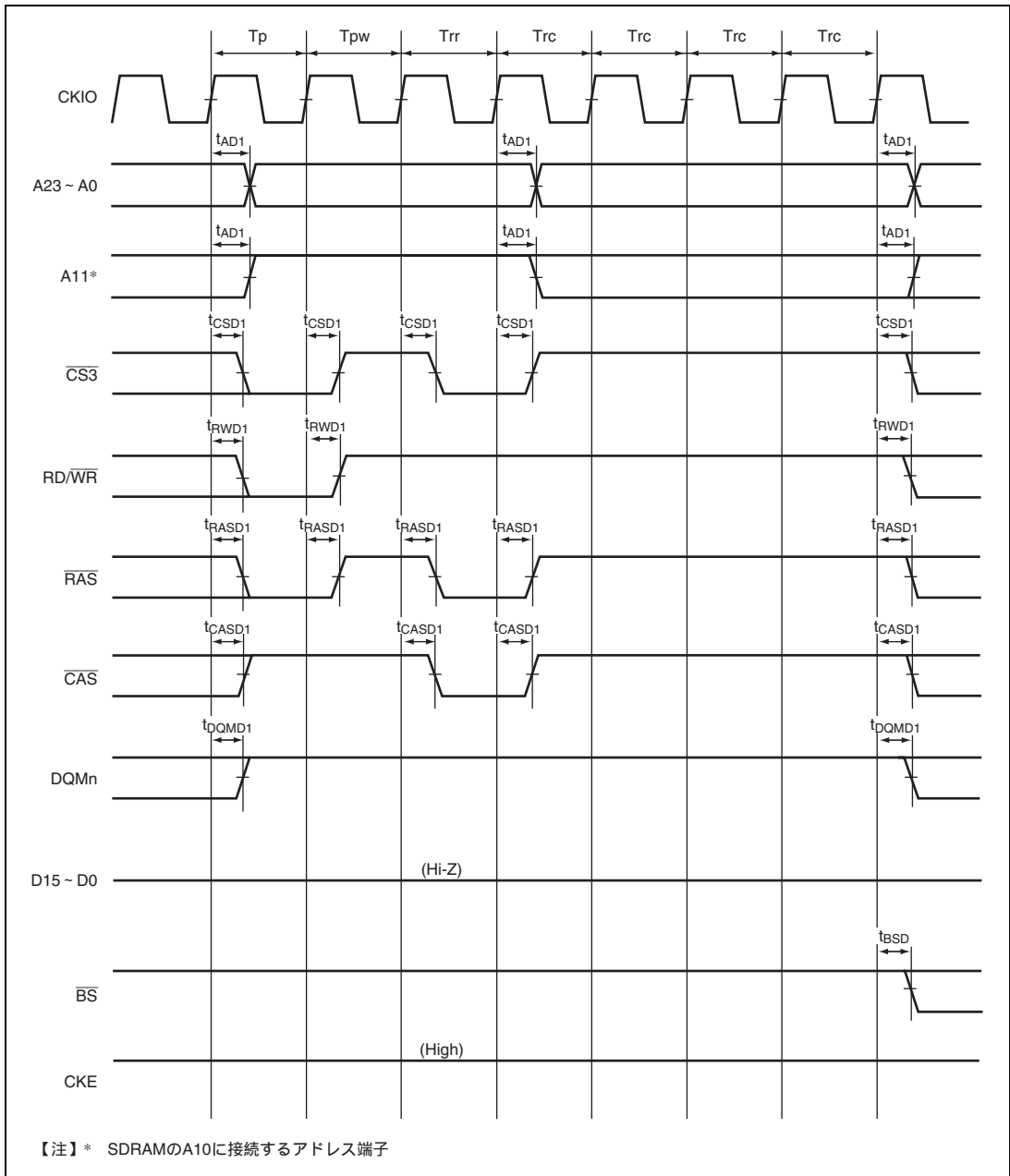


図 28.40 シンクロナス DRAM オートリフレッシュタイミング (TRP=2 サイクル)

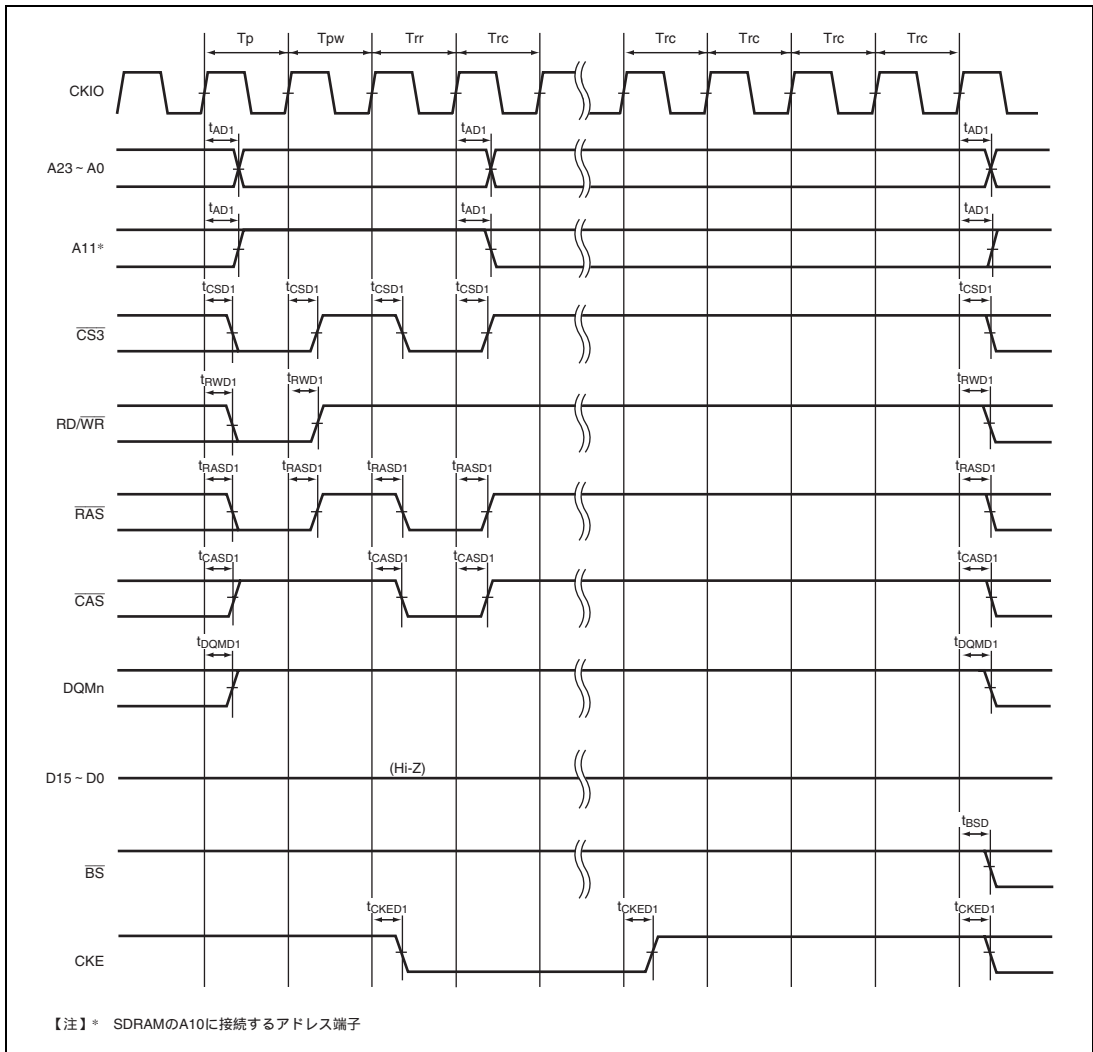


図 28.41 シンクロナス DRAM セルフリフレッシュタイミング (TRP=2 サイクル)

28. 電氣的特性

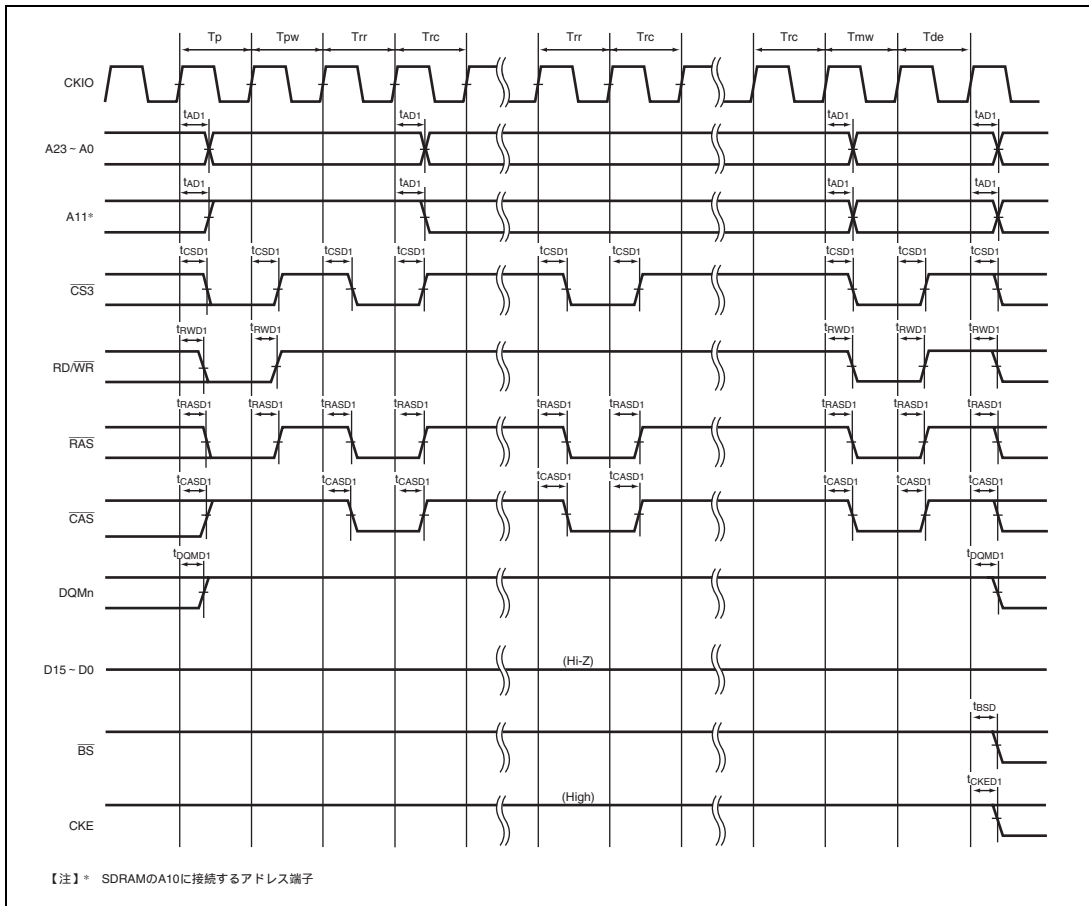


図 28.42 シンクロナス DRAM パワーオンシーケンス (モードライトタイミング、TRP = 2 サイクル)

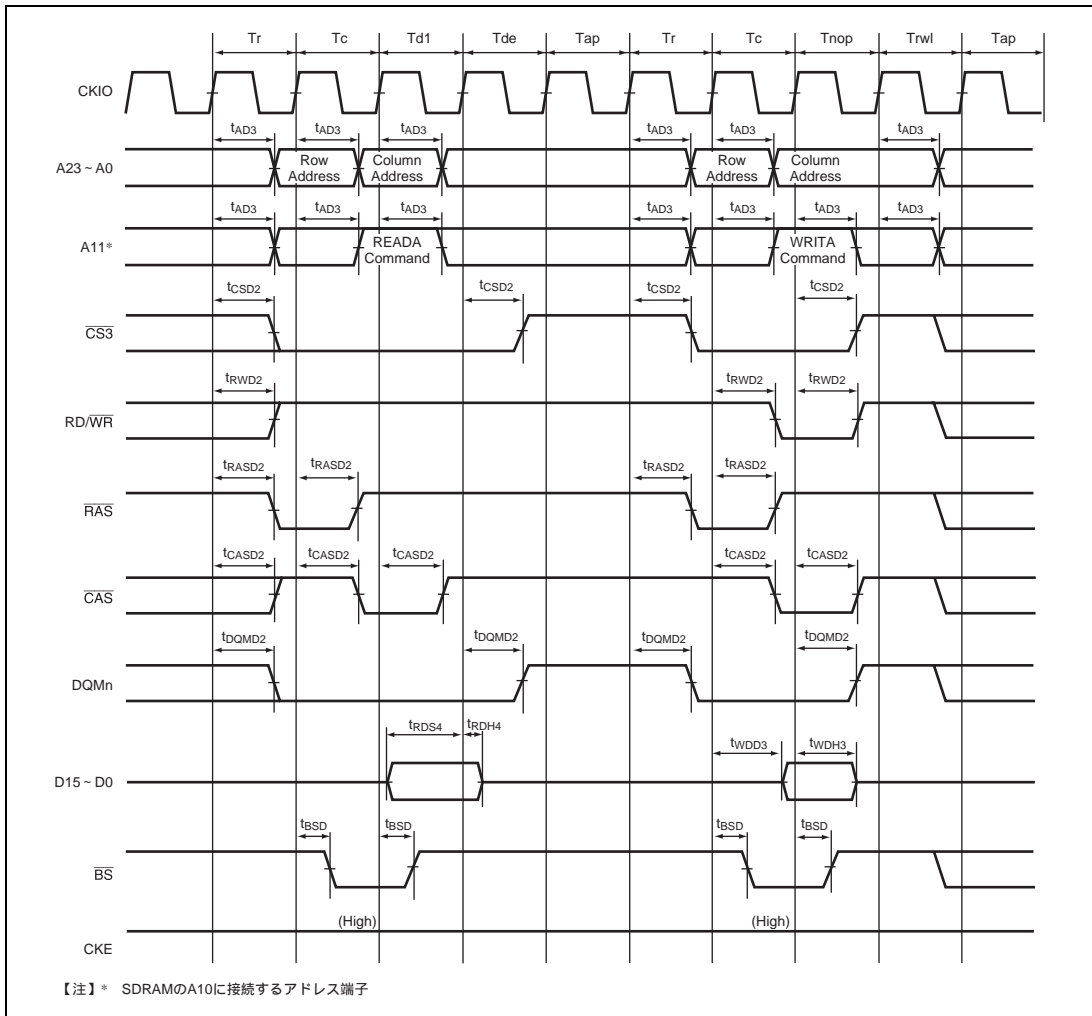


図 28.43 シンクロナス DRAM 低周波モードでのアクセスタイミング
(オートプリチャージモード、TRWL = 1 サイクル)

28. 電気的特性

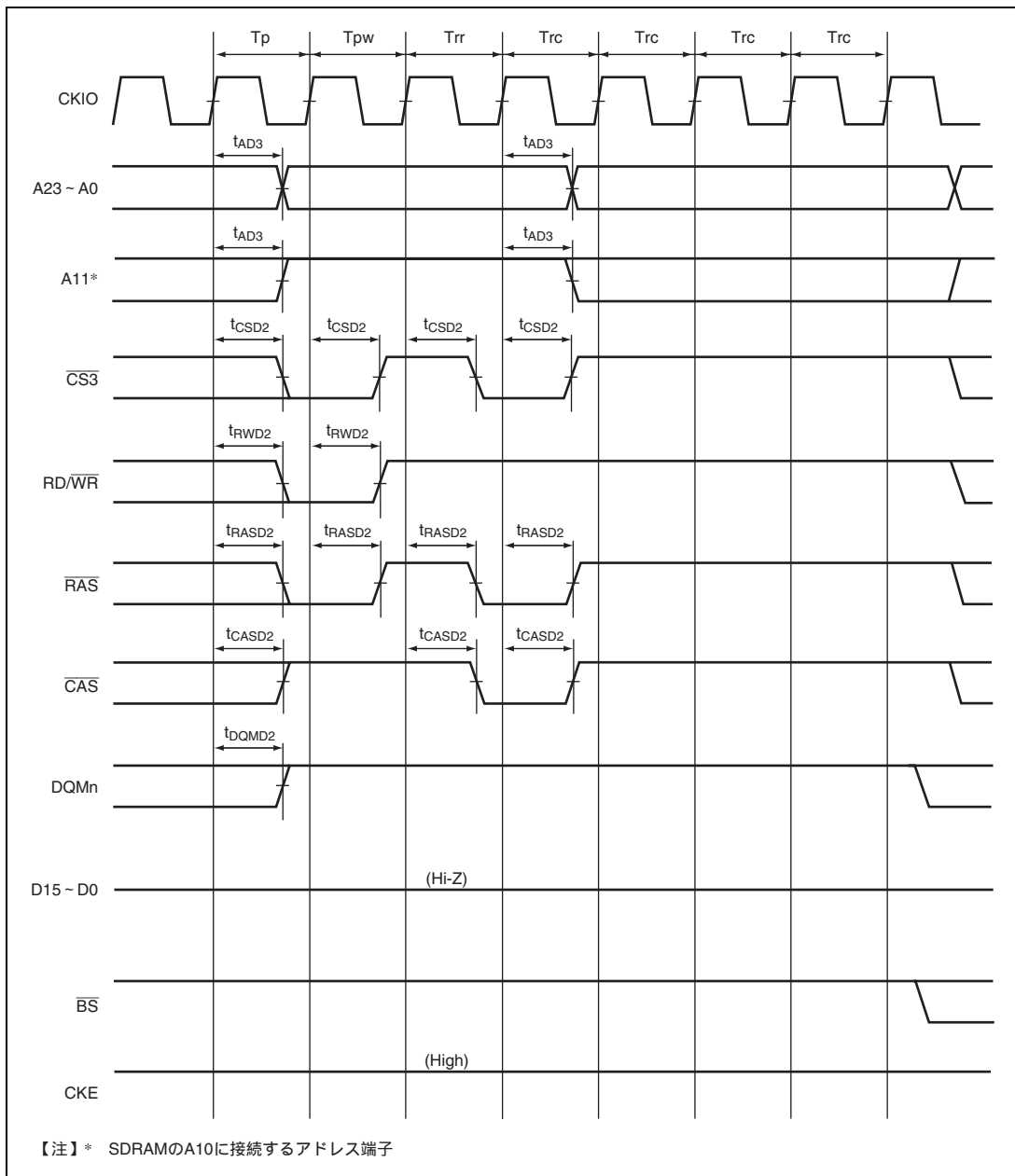


図 28.44 シンクロナス DRAM 低周波モードオートリフレッシュタイミング (TRP=2 サイクル)

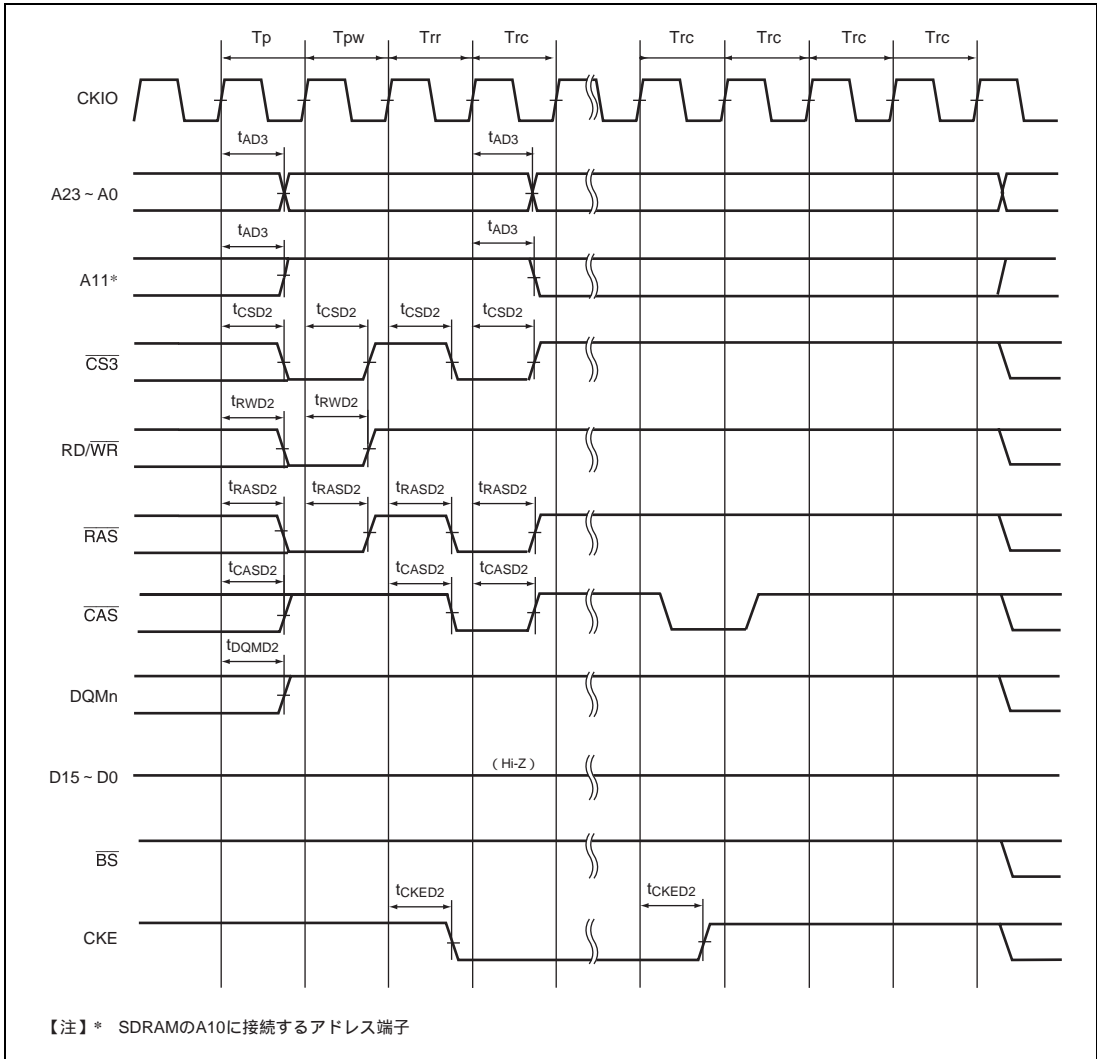


図 28.45 シンクロナス DRAM 低周波モードセルフリフレッシュタイミング (TRP=2 サイクル)

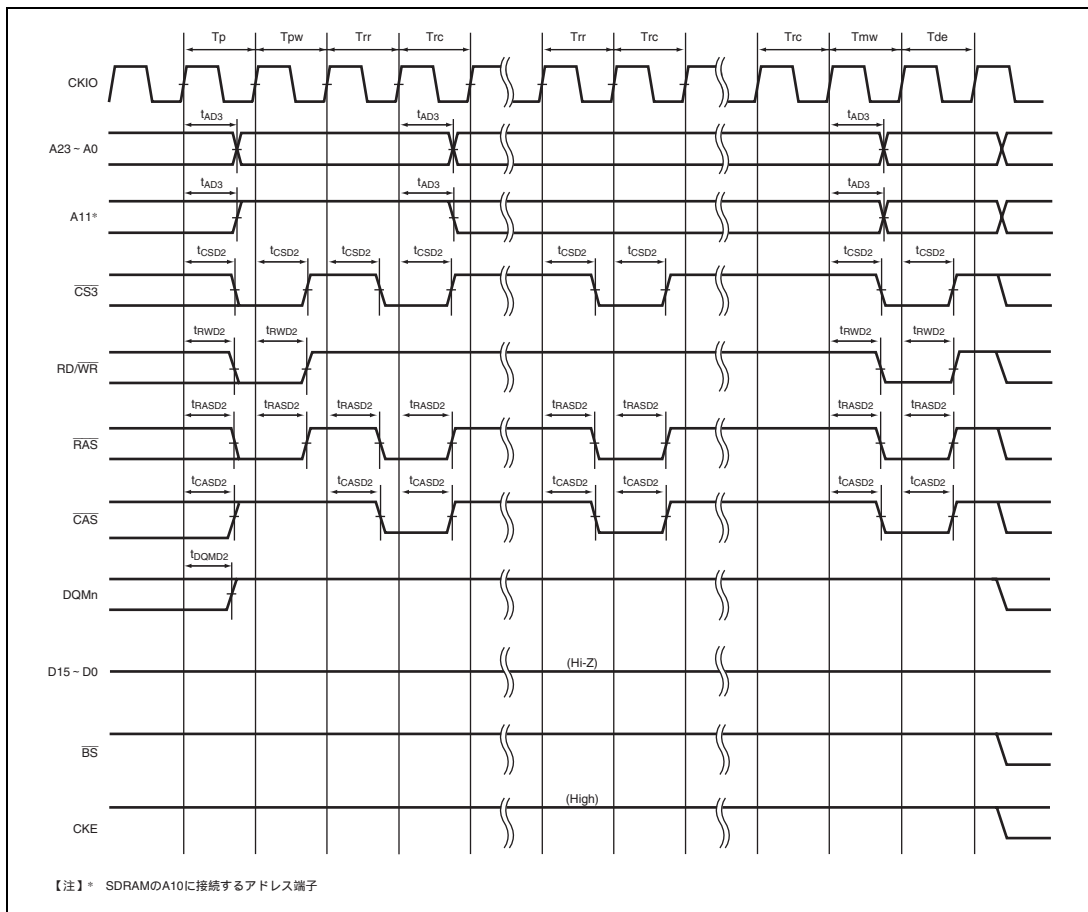


図 28.46 シンクロナス DRAM 低周波モードパワーオンシーケンス
(モードライトタイミング、TRP = 2 サイクル)

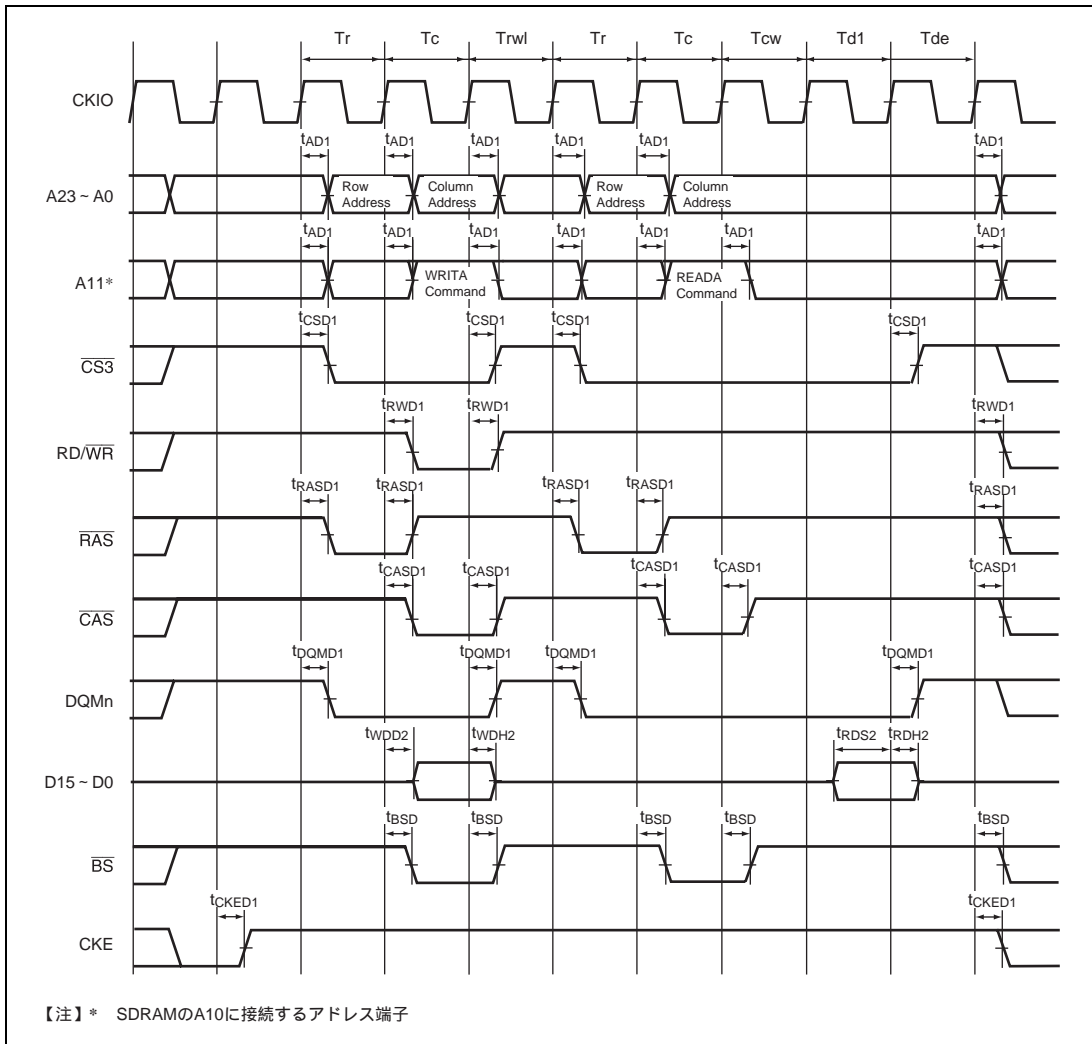


図 28.47 シンクロナス DRAM パワーダウンモードライトリードバスサイクル
 (オートプリチャージモード、TRCD=1 サイクル、TRP=1 サイクル、TRWL=1 サイクル)

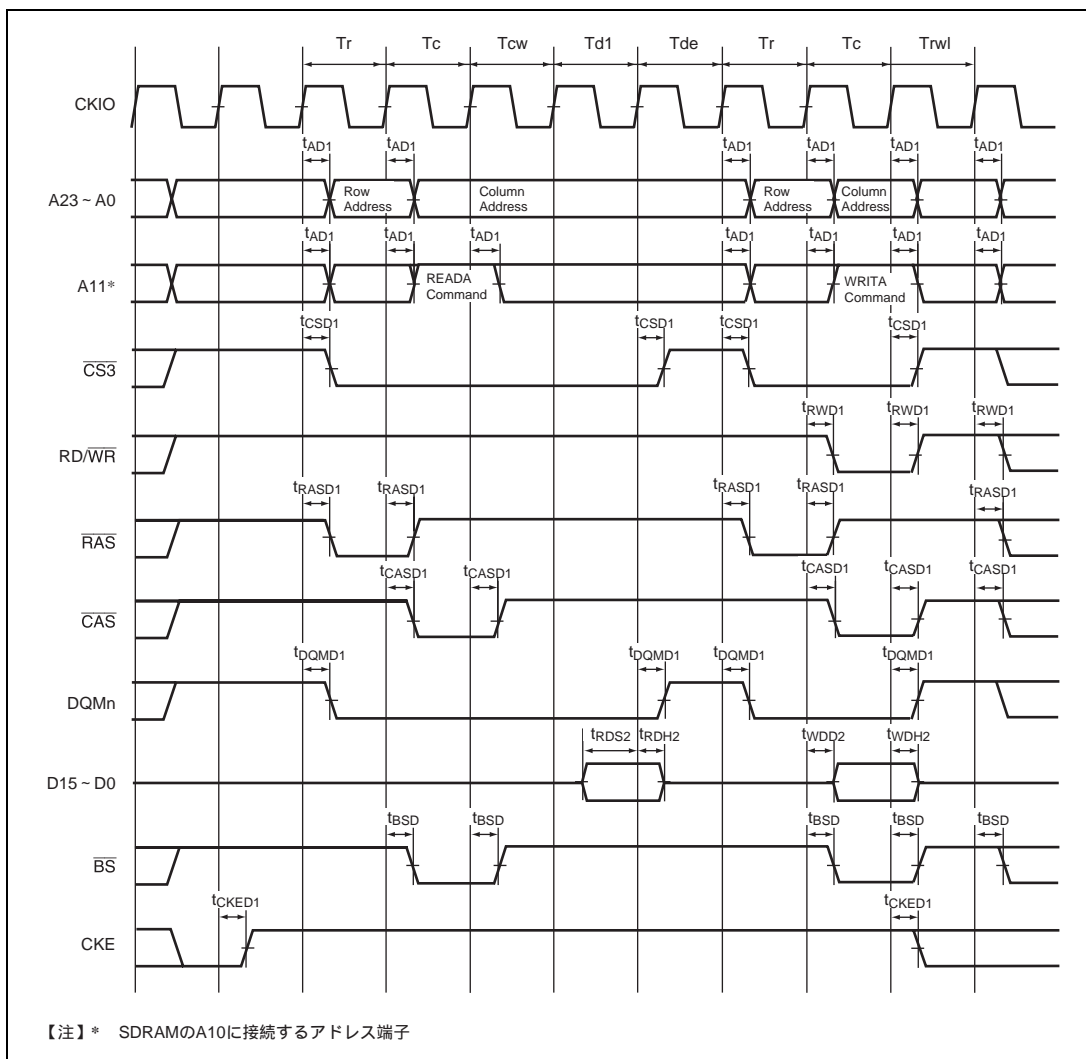


図 28.48 シンクロナス DRAM パワーダウンモードリードライトバスサイクル
 (オートプリチャージモード、TRCD=1 サイクル、TRP=1 サイクル、TRWL=1 サイクル)

28.3.7 周辺モジュール信号タイミング

表 28.8 周辺モジュール信号タイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $AV_{CC} = 2.7 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

モジュール	項目	記号	Min.	Max.	単位	参照図
I/O ポート	出力データ遅延時間	t_{PORTD}	-	17	ns	28.49
	入力データセットアップ時間	t_{PORTS}	17	-		
	入力データホールド時間	t_{PORTH}	10	-		
DMAC	DREQ セットアップ時間	t_{DREQS}	8	-	ns	28.50
	DREQ ホールド時間	t_{DREQH}	8	-		
	DACK 遅延時間	t_{DACKD}	-	14	28.51	
	TEND 遅延時間	t_{TENDD}	-	14		

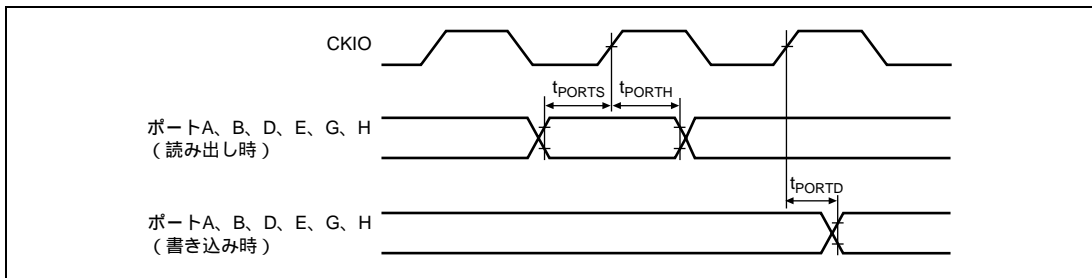


図 28.49 I/O ポートタイミング

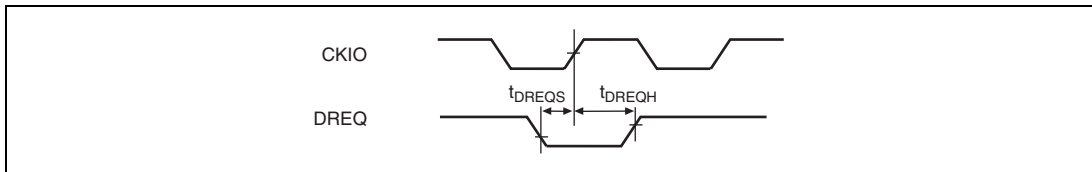


図 28.50 DREQ 入力タイミング (DREQ ローレベル検出)

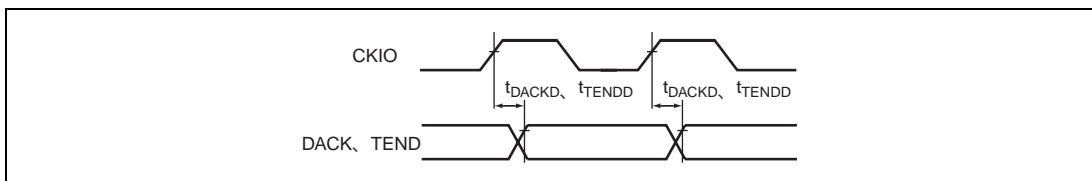


図 28.51 DACK、TEND 出力タイミング

28.3.8 SIOF モジュール信号タイミング

表 28.9 SIOF モジュール信号タイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $AV_{CC} = 2.7 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Max.	単位	参照図
SIOF_MCLK クロック入力サイクル時間	t_{MCYC}	t_{poyc}		ns	28.52
SIOF_MCLK 入力 High レベル幅	t_{MWH}	$0.4 \times t_{MCYC}$		ns	28.52
SIOF_MCLK 入力 Low レベル幅	t_{MWL}	$0.4 \times t_{MCYC}$		ns	28.52
SIOF_SCK クロックサイクル時間	t_{SICYC}	t_{poyc}		ns	28.53 ~ 28.57
SIOF_SCK 出力 High レベル幅	t_{SWHO}	$0.4 \times t_{SICYC}$		ns	28.53 ~ 28.56
SIOF_SCK 出力 Low レベル幅	t_{SWLO}	$0.4 \times t_{SICYC}$		ns	28.53 ~ 28.56
SIOF_SYNC 出力遅延時間	t_{FSD}		20	ns	28.53 ~ 28.56
SIOF_SCK 入力 High レベル幅	t_{SWHI}	$0.4 \times t_{SICYC}$		ns	28.57
SIOF_SCK 入力 Low レベル幅	t_{SWLI}	$0.4 \times t_{SICYC}$		ns	28.57
SIOF_SYNC 入力セットアップ時間	t_{FSS}	20		ns	28.57
SIOF_SYNC 入力ホールド時間	t_{FSH}	20		ns	28.57
SIOF_TXD 出力遅延時間	t_{STDD}		20	ns	28.53 ~ 28.59
SIOF_RXD 入力セットアップ時間	t_{SRDS}	20		ns	28.53 ~ 28.59
SIOF_RXD 入力ホールド時間	t_{SRDH}	10		ns	28.53 ~ 28.59
スレープセレクトセットアップ時間	t_{SSS}	$t_{SICYC}/2 \times n^{*2} - 20$		ns	28.58, 28.59
スレープセレクトホールド時間	t_{SSH}	$t_{SICYC}/2 \times n^{*2}$		ns	28.58, 28.59

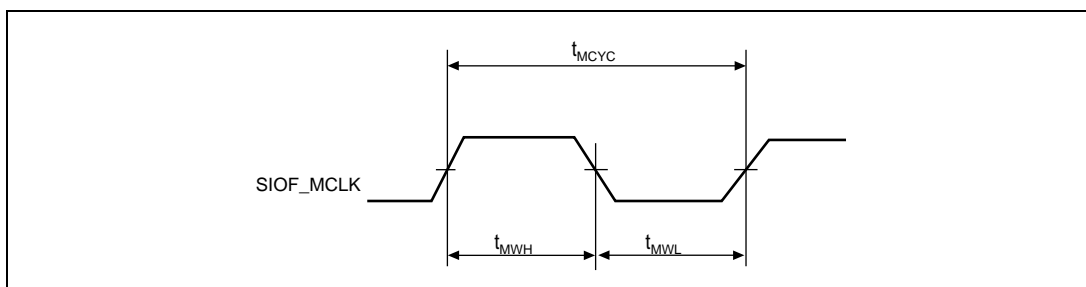
【注】 *1 t_{poyc} は周辺クロック (P) の 1 サイクル時間を示します。*2 $t_{SICYC}/2 \times n$ は SPICR の SSAST[1 : 0]ビットによって決定されます。

図 28.52 SIOF_MCLK 入力タイミング

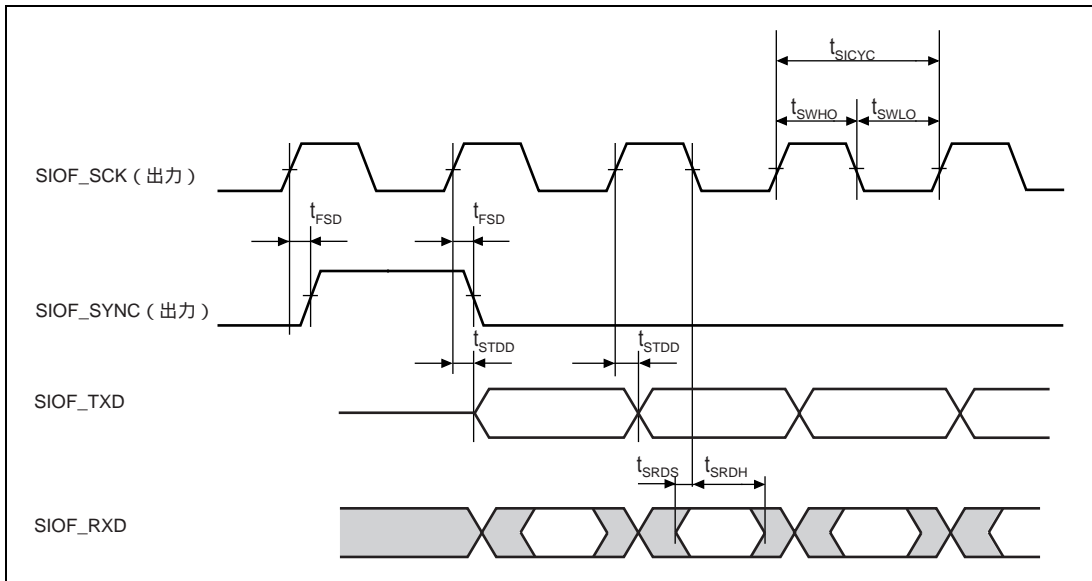


図 28.53 SIOF 送受信タイミング (マスタモード 1・立ち下がりサンプリング時)

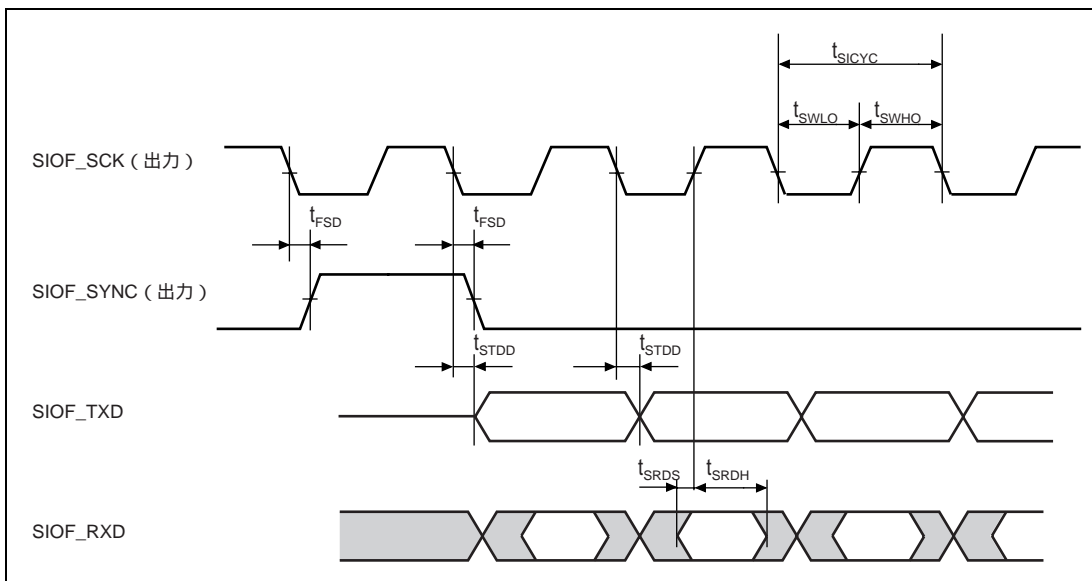


図 28.54 SIOF 送受信タイミング (マスタモード 1・立ち上がりサンプリング時)

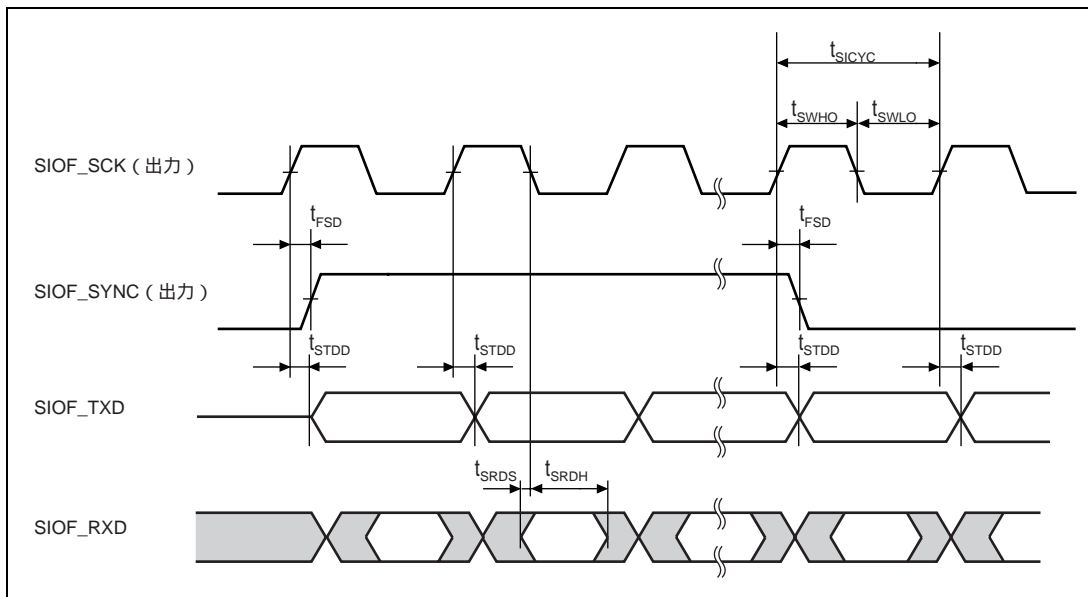


図 28.55 SIOF 送受信タイミング (マスタモード 2・立ち下がりサンプリング時)

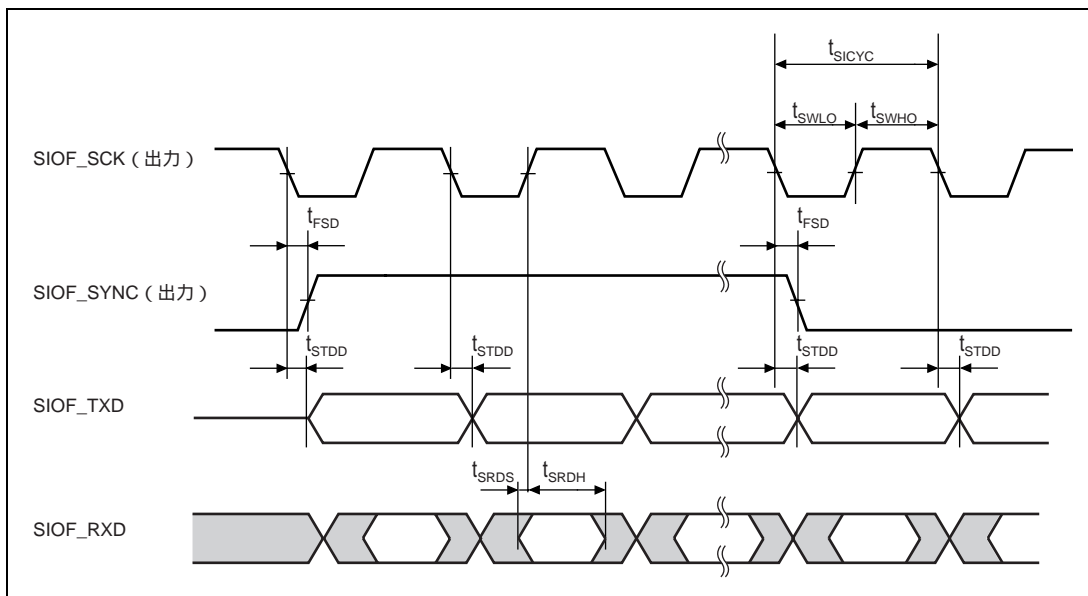


図 28.56 SIOF 送受信タイミング (マスタモード 2・立ち上がりサンプリング時)

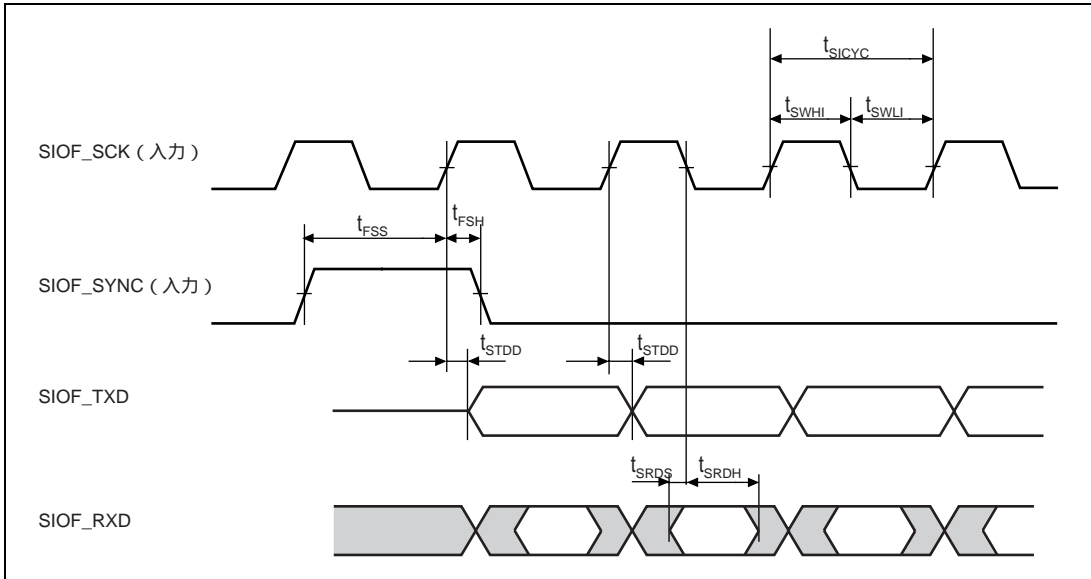


図 28.57 SIOF 送受信タイミング (スレープモード 1・スレープモード 2 時)

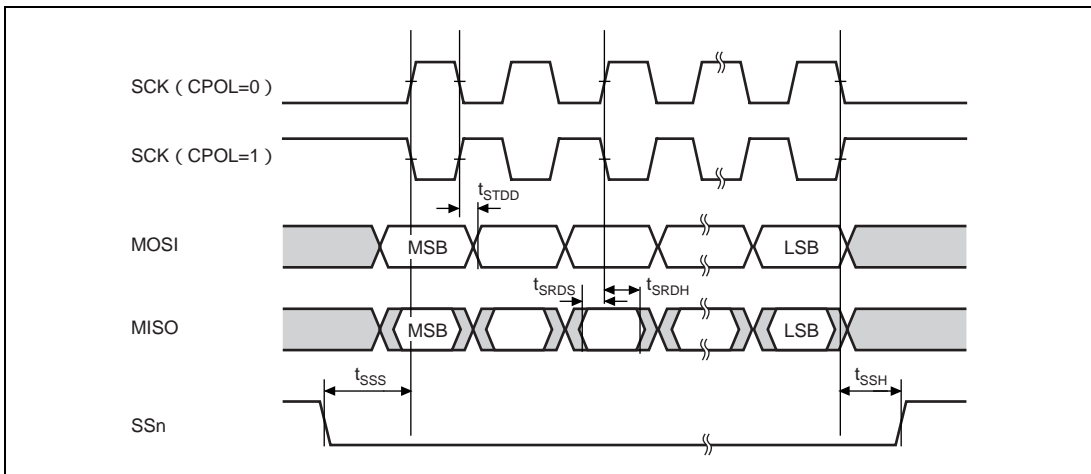


図 28.58 SIOF 送受信タイミング (SPI モード、CPHA = 0、SSAST[1 : 0] = 01)

28. 電気的特性

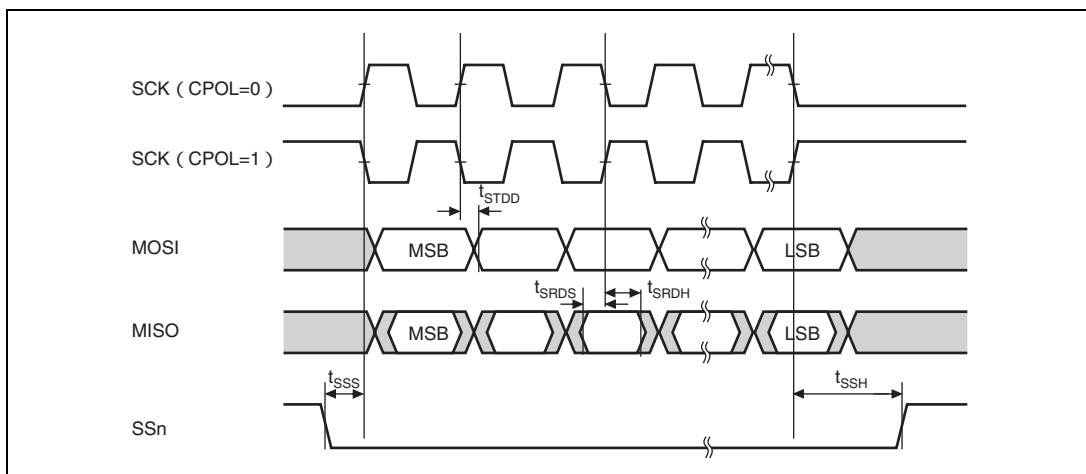


図 28.59 SIOF 送受信タイミング (SPI モード、CPHA = 1、SSAST[1 : 0] = 01)

28.3.9 SCIF モジュール信号タイミング

表 28.10 SCIF モジュール信号タイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC28} = 2.7 \sim 3.0V$ 、 $AV_{CC} = 2.7 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Max.	単位	参照図
SCIF 入力クロックサイクル	調歩同期	t_{SCYC}	$4 \times t_{PCYC}$	ns	28.60、 28.61
	クロック同期		$12 \times t_{PCYC}$	ns	
SCIF 入力クロックハイレベル幅	t_{SCWH}	$0.4 \times t_{PCYC}$		ns	28.60
SCIF 入力クロックローレベル幅	t_{SCWL}	$0.4 \times t_{PCYC}$		ns	
SCIF 送信データ遅延時間	t_{TXD}		$3t_{PCYC} + 50$	ns	28.61
SCIF 受信データセットアップ時間 (クロック同期)	t_{RXS}	$2 \times t_{PCYC}$		ns	
SCIF 受信データホールド時間 (クロック同期)	t_{RXH}	$2 \times t_{PCYC}$		ns	

【注】 t_{pcyc} は周辺クロック (P) の 1 サイクル時間を示します。

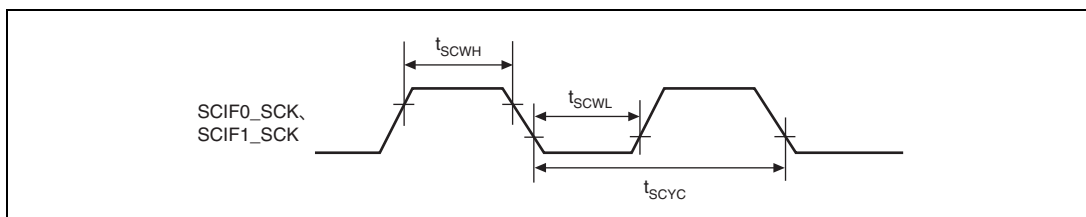


図 28.60 SCIF 入力クロックサイクル

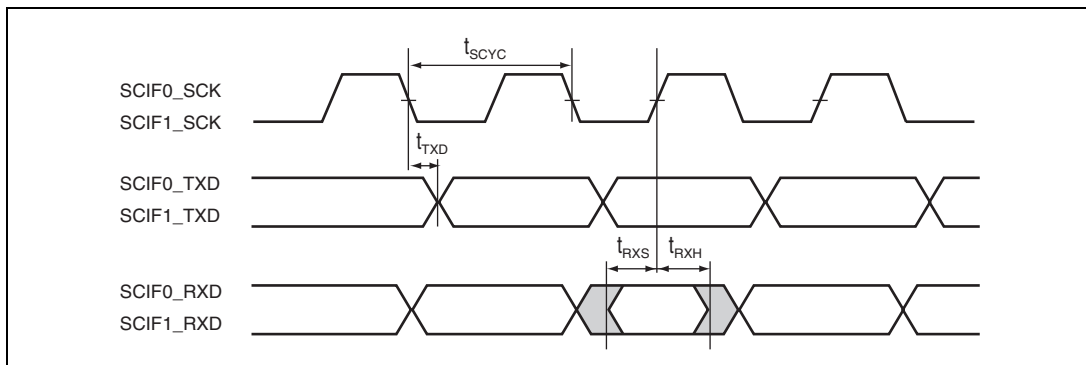


図 28.61 SCIF クロック同期式モード時の入出力タイミング

28.3.10 USB モジュール信号タイミング

表 28.11 USB モジュールクロックタイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $AV_{CC}(USB) = 3.0 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Max.	単位	参照図
クロック立ち上がり時間	t_{R48}		4	ns	28.62
クロック立ち下がり時間	t_{F48}		4	ns	
デューティ	t_{HIGH} / t_{LOW}	90	110	%	

表 28.12 USB モジュールクロックタイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $AV_{CC}(USB) = 3.0 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	周波数	単位	条件
周波数	$1/t_{FREQ}$	48.0 ($\pm 500ppm$)	MHz	USB ホスト使用時
		48.0 ($\pm 2500ppm$)	MHz	USB ファンクション使用時

【注】 USB は内蔵遅倍回路を使用しないで外部から UCLK 端子にクロック供給させて動作させる場合は、USB 規格に準拠するクロックを入力してください。

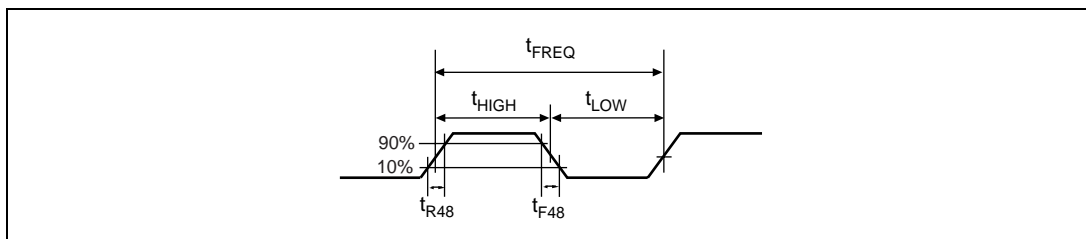


図 28.62 USB クロックタイミング

28.3.11 USB トランシーバタイミング

表 28.13 USB トランシーバタイミング (Full Speed 動作時)

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $AV_{CC}(USB) = 3.0 \sim 3.6V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	測定条件
立ち上がり時間	t_r	4	-	20	ns	$C_L = 50pF$
立ち下がり時間	t_f	4	-	20	ns	$C_L = 50pF$
立ち上がり / 立ち下がり時間比	t_r/t_f	80	-	120	%	
出力信号クロスオーバー電圧	V_{CRS}	1.3	-	2.0	V	-
出力ドライバ抵抗*	Z_{DRU}	28	-	44		

【注】 * 外付け直列抵抗 $R_S = 27 \pm 1\%$ を含みます。

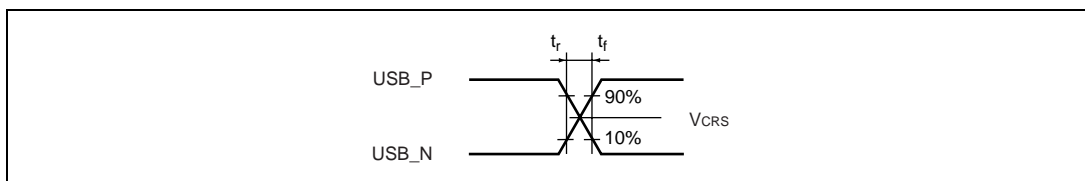


図 28.63 USB トランシーバタイミング

• 測定回路

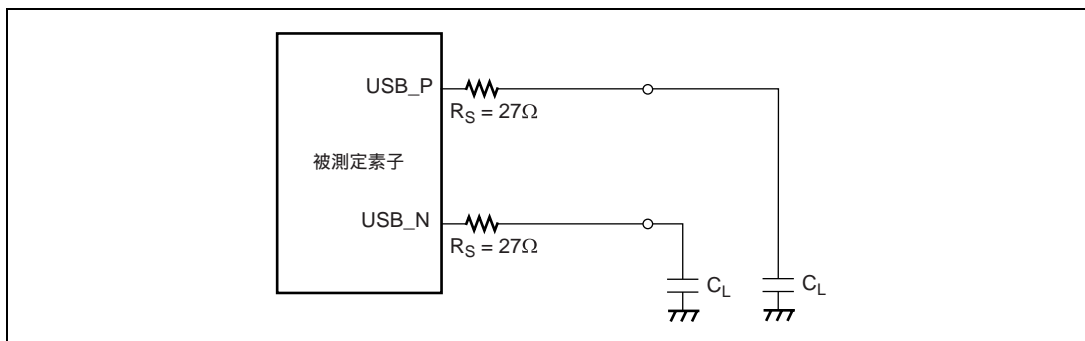


図 28.64 USB トランシーバ特性評価回路 (Full Speed 時)

28.3.12 ブルートゥースインタフェース (BT) タイミング

表 28.14 ブルートゥースインタフェースモジュールクロックタイミング

条件: $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	参照図
クロック立ち上がり時間	t_r			4	ns	28.65
クロック立ち下がり時間	t_f			4	ns	
デューティー	t_{HIGH} / t_{LOW}	90	100	110	%	

表 28.15 ブルートゥースインタフェースモジュールクロックタイミング

条件: $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	周波数	単位	条件
周波数	$1/t_{FREQ}$	13.0 ($\pm 20ppm$)	MHz	弊社 RF IC を接続する場合* ¹ 、* ² Bluetooth 機能を使用しない場合 (本 LSI 単独使用時)* ²
		8.0 ~ 13.0		

【注】 *1 接続可能な RF IC は HD157102NP です。

*2 内蔵通信回路を使用して RDI_REFCLK_IN (13MHz) から USB クロックを生成し、USB を使用する場合には、クロック精度は USB 規格の値を保証できません。クロック精度を保証したい場合には、UCLK 端子から直接入力を使用してください。

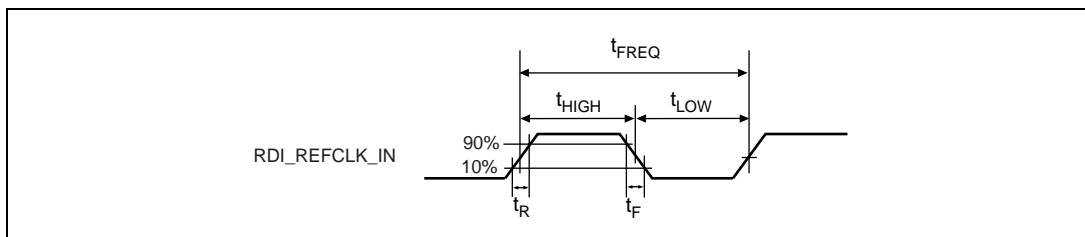


図 28.65 ブルートゥースインタフェースモジュールクロックタイミング

表 28.16 ブルートゥースインタフェースモジュールローパワークロックタイミング

条件: $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	参照図
クロック立ち上がり時間	t_r			4	ns	28.66
クロック立ち下がり時間	t_f			4	ns	
デューティー	t_{HIGH} / t_{LOW}	90	100	110	%	

28. 電気的特性

表 28.17 ブルートゥースインタフェースモジュールローパワークロックタイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	周波数	単位	条件
周波数	$1/t_{FREQ}$	32.0 ($\pm 250ppm$)	kHz	周波数変換回路バイパス時
		32.768 ($\pm 250ppm$)	kHz	周波数変換回路使用時 (ただし最大誤差は $30 \mu s$ 程度発生)

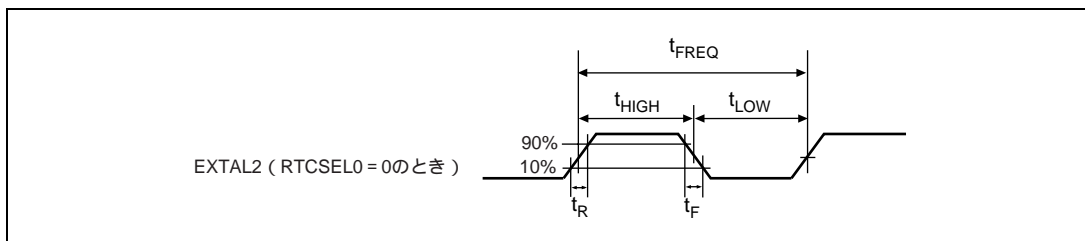


図 28.66 ブルートゥースインタフェースモジュールローパワークロックタイミング

表 28.18 ブルートゥースインタフェース (BT) Voice Codec インタフェース信号のタイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$ 、 $V_{CC-28} = 2.7 \sim 3.0V$ 、 $V_{DD} = 1.4 \sim 1.6V$ 、 $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	参照図
VCI_SCO_CLK_OUT クロック出力周波数	f_{VCI_CLK}	0.2	-	1	MHz	28.67、28.68
VCI_SCO_SYNC_OUT 出力遅延時間	t_{SYNCD}	-	-	200	ns	28.67、28.68
VCI_SCO_RX 出力遅延時間	t_{RXD}	-	-	200	ns	28.67、28.68
VCI_SCO_TX セットアップ時間	t_{TXS}	100	-	-	ns	28.67、28.68
VCI_SCO_TX ホールド時間	t_{TXH}	100	-	-	ns	28.67、28.68
VCI_HWC 出力遅延時間*	T_{HWCD}	-	-	200	ns	28.67

【注】 * 外付け Voice Codec IC として STLC7550 を接続する場合。

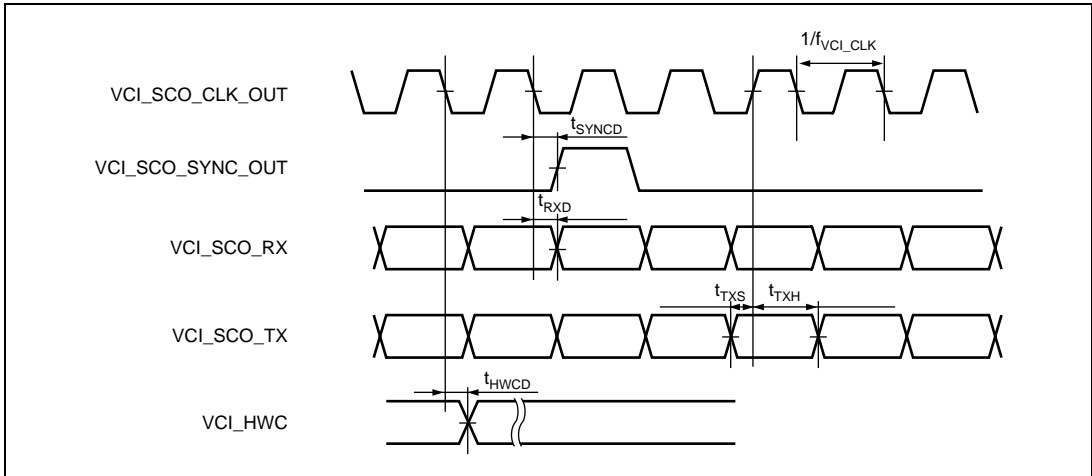


図 28.67 ブルートゥースインタフェース (BT) Voice Codec(STLC7550)インタフェース信号のタイミング

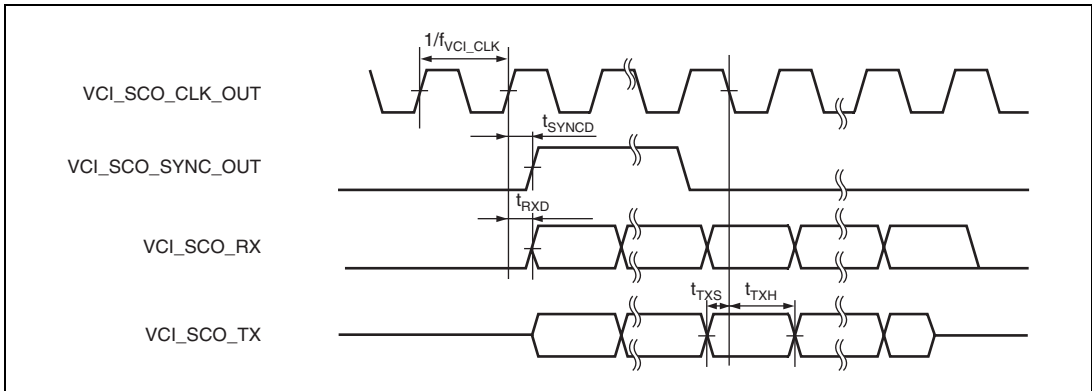


図 28.68 ブルートゥースインタフェース (BT) Voice Codec(MC145483)インタフェース信号のタイミング

表 28.19 ブルートゥースインタフェース (BT) RF 用 SPI インタフェース信号のタイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$, $V_{CC-28} = 2.7 \sim 3.0V$, $V_{DD} = 1.4 \sim 1.6V$, $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	参照図
RCL_SPI_CLK クロック出力周波数*	f_{SPI_CLK}	-	6.5	-	MHz	28.69
RCL_SPI_CLK duty		45	50	55	%	28.69
RCL_SPI_TXRX 出力遅延時間	t_{DELAY}	-	-	30	ns	28.69
RCL_SPI_TXRX セットアップ時間	t_{SETUP}	40	-	-	ns	28.69
RCL_SPI_TXRX ホールド時間	t_{HOLD}	45	-	-	ns	28.69

【注】 * ルネサス テクノロジ製 RF IC (HD157102NP) を接続する場合。

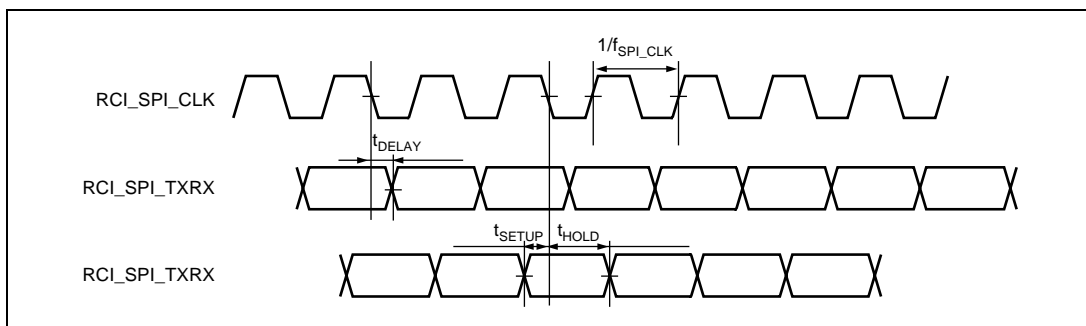


図 28.69 ブルートゥースインタフェース (BT) RF 用 SPI インタフェース信号のタイミング

表 28.20 受信データタイミング

条件 : $V_{CC} = 2.7 \sim 3.6V$, $V_{CC-28} = 2.7 \sim 3.0V$, $V_{DD} = 1.4 \sim 1.6V$, $T_a = -40 \sim 85$

項目	記号	Min.	Typ.	Max.	単位	参照図
RDI_TXTRDATA	$t_{TXTRDATA}$	0.7	1	1.3	μs	28.70

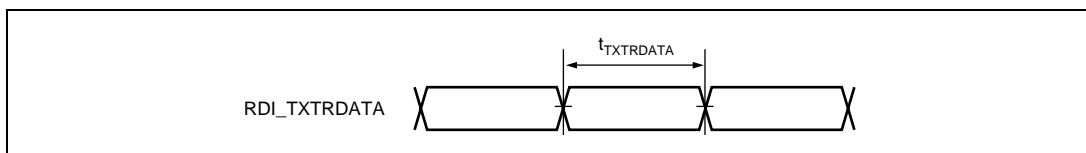


図 28.70 ブルートゥースインタフェース (BT) 受信データタイミング

28.3.13 AC 特性測定条件

- 入出力信号参照レベル : 1.5V ($V_{CC} = 2.7 \sim 3.6V$, $V_{CC-28} = 2.7 \sim 3.0V$)
- 入力パルスレベル : $V_{SS} \sim 3.3V$ (ただし、 \overline{RESETP} 、RDI_TXTRDATA、RDI_REFCLK_IN、RCI_SPL_TXRXは、 $V_{SS} \sim 2.8V$)
- 入力立ち上がり、立ち下がり時間 : 1ns

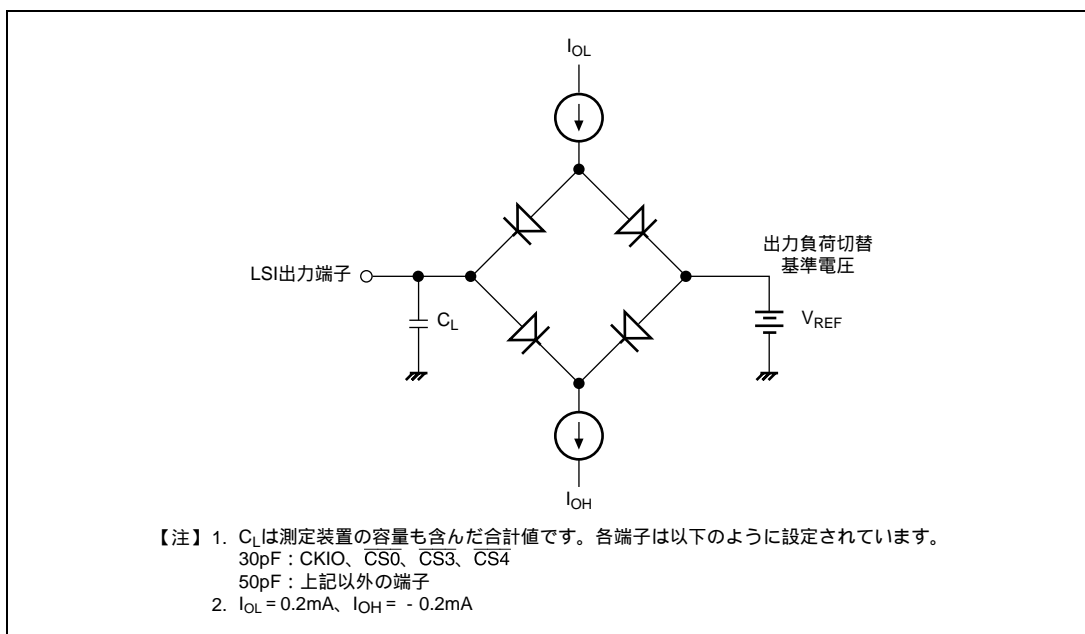


図 28.71 出力負荷回路

28.4 D/A コンバータ特性

D/A コンバータ特性を表 28.21 に示します。

表 28.21 D/A コンバータ特性

条件 : $V_{CC} = 2.7 \sim 3.6\text{V}$ 、 $V_{CC-28} = 2.7 \sim 3.0\text{V}$ 、 $AV_{CC}(\text{DAC}) = 2.7 \sim 3.6\text{V}$ 、 $V_{DD} = 1.4 \sim 1.6\text{V}$ 、 $T_a = -40 \sim 85$

項目	Min.	Typ.	Max.	単位
分解能	8	8	8	bits
変換時間 (20pF 容量負荷)	-	-	10	μs
絶対精度 (2M 抵抗負荷)	-	-	± 4.0	LSB

付録

A. 端子状態表

本 LSI の各端子の状態は、動作モード（リセット、低消費電力モード、バス権解放状態）により下表のようになります。表中の記号の意味は、次の通りです。

【記号の意味】

- O: 出力（不定）
- I: 入力
- IP: 入力（プルアップ MOS オン）
- H: ハイレベル出力
- L: ローレベル出力
- Z1: ハイインピーダンス（入力/出力バッファオフ・プルアップ MOS オフ）
- Z2: ハイインピーダンス（出力バッファオフ）
- P: レジスタ設定によって「I」か「O」になります。
- K: 入力端子はハイインピーダンス、出力端子は状態の保持
- V: 入力/出力バッファオフ・プルアップ MOS オン

表 A.1 端子状態

分類	端子名	リセット		低消費電力モード			バス権解放 状態
		パワーオン リセット	マニュアル リセット	ソフトウェア スタンバイ	モジュール スタンバイ	スリープ	
クロック	EXTAL	I	I	I	I	I	I
	XTAL	O	O	O	O	O	O
	CKIO	I、O* ¹	I、O* ¹	Z、I、O* ^{1&2}	I、O* ¹	I、O* ¹	Z、I、O* ^{1&2}
動作モード コントロール	MD5、MD2、MD1	I	I	I	I	I	I
システム制御	RESETP	I	I	I	I	I	I
	BOOT_E	IP	IP	IP	IP	IP	IP
	BREQ	I	I	I	I	I	I
	BACK	Z1	O	O	O	O	O
割り込み	NMI	I	I	I	I	I	I
	IRQ4 ~ IRQ2、IRQ0	V	I	I	I	I	I
	IRQOUT	V	I	O	O	O	O

付録

分類	端子名	リセット		低消費電力モード			バス権解放 状態
		パワーオン リセット	マニュアル リセット	ソフトウェア スタンバイ	モジュール スタンバイ	スリープ	
アドレスバス	A23 - A19, A0	Z1	O	Z, O ^{*3}	O		Z1
	A18 - A1	O	O	Z, O ^{*3}	O		Z2
データバス	D15 - D0	Z1	Z1	Z1	I, O		Z1
バス制御	CS0, CS3, CS4	H	O	Z, O ^{*3}	O		Z2
	\overline{RD}	H	O	Z, O ^{*3}	O		Z2
	RD/ \overline{WR}	H	O	Z, O ^{*3}	O		Z2
	\overline{BS}	H	O	Z, O ^{*3}	O		Z2
	$\overline{WE1/DQM1}$	H	O	Z, O ^{*3}	O		Z2
	$\overline{WE0/DQM0}$	H	O	Z, O ^{*3}	O		Z2
	\overline{CAS}	H	O	Z, O ^{*3}	O		Z, O ^{*4}
	\overline{RAS}	H	O	Z, O ^{*3}	O		Z, O ^{*4}
	CKE	H	O	Z, O ^{*3}	O		Z, O ^{*4}
	\overline{REFOUT}	O	O	O	O		O
	WAIT	IP	IP	IP	IP		IP
ダイレクトメモリ アクセスコントローラ (DMAC)	DREQ	V	I	Z1	I		I
FIFO 付きシリアル インタフェース (SIOF)	SIOF_TXD(MOSI)	V	O	Z1	O		O
	SIOF_RXD(MISO)	V	I	Z1	I		I
	SIOF_SCK(SCK)	V	O	Z1	I, O		I, O
	SIOF_MCLK	V	I	Z1	I		I
	SIOF_SYNC(SIOF_SS0)	V	O	Z1	I, O		I, O
	SIOF_SS1	V	O	Z1	O		O
	SIOF_SS2	V	O	Z1	O		O
FIFO 内蔵シリアル コミュニケーション インタフェース (SCIF0, 1)	SCIF0_TXD, SCIF1_TXD	Z1	Z1	Z1	O		O
	SCIF0_RXD, SCIF1_RXD	V	Z1	Z1	I		I
	SCIF0_SCK, SCIF1_SCK	V	Z1	Z1	I, O		I, O
	SCIF0_RTS, SCIF1_RTS	V	O	Z1	O		O
	SCIF0_CTS, SCIF1_CTS	V	I	Z1	I		I

分類	端子名	リセット		低消費電力モード			バス権解放 状態
		パワーオン リセット	マニュアル リセット	ソフトウェア スタンバイ	モジュール スタンバイ	スリープ	
USB	UCLK	I	I	I	I	I	I
	USB_PWR_EN/ USB_PULLUP	O	O	O	O	O	O
	USB_OVR_CRNT/ USB_VBUS	I	I	I	I	I	I
	USB_N	I, O	I, O	I, O	I, O	I, O	I, O
	USB_P	I, O	I, O	I, O	I, O	I, O	I, O
ブルートゥース インタフェース (BT)	RDI_TXTRDATA	O	O	O	O	O	O
	RDI_RXBDW_OUT	O	O	O	O	O	O
	RDI_REFCLK_IN	I	I	I	I	I	I
	RDI_CTRL2	O	O	O	O	O	O
	RDI_CTRL3	O	O	O	O	O	O
	RDI_CTRL4	O	O	O	O	O	O
	RCI_SPI_CLK	O	O	O	O	O	O
	RCI_SPI_TXRX	O	O	O	O	O	O
	RCI_SPI_ENB	O	O	O	O	O	O
	VCI_SCO_CLK_OUT	Z1	O	O	O	O	O
	VCI_SCO_SYNC_OUT	Z1	O	O	O	O	O
	VCI_SCO_TX	V	I	I	I	I	I
	VCI_SCO_RX	V	O	O	O	O	O
	VCI_HWC	V	O	O	O	O	O
	VCI_CODEC_PWRDWN	V	O	O	O	O	O
	RTCSEL0	I	I	I	I	I	I
	RREF	I	I	I	I	I	I
EXTAL2	I	I	I	I	I	I	
XTAL2	O	O	O	O	O	O	
D/A コンバータ (DAC)	DA1, DA0	Z2	O	O	O	O	O
I/O ポート	PTA6 ~ PTA0	V	P	K	P	P	P
	PTB7 ~ PTB0 ^{*5}	V	P	K	P	P	P
	PTD5 ~ PTD0 ^{*5}	V	P	K	P	P	P
	PTE6 ~ PTE0 ^{*5}	V	P	K	P	P	P
	PTG4 ~ PTG2, PTG0	V	P	K	P	P	P
	PTG1	Z1	P	K	P	P	P
	PTH4 ~ PTH2, PTH0	V	P	K	P	P	P
	PTH1	Z1	P	K	P	P	P

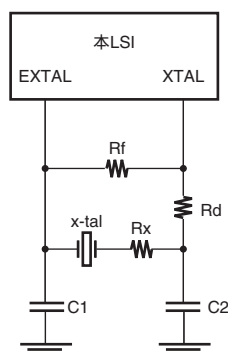
分類	端子名	リセット		低消費電力モード			バス権解放 状態
		パワーオン リセット	マニュアル リセット	ソフトウェア スタンバイ	モジュール スタンバイ	スリープ	
ユーザデバッグ インタフェース (H-UDI)	TCK	I	I	I	I	I	
	TMS	IP	IP	IP	IP	IP	
	TDI	IP	IP	IP	IP	IP	
	TDO	Z2	Z2	Z2	Z2	Z2	
	$\overline{\text{TRST}}$	IP	IP	IP	IP	IP	
アドバンスト ユーザデバッグ (AUD)	AUDATA3 ~ AUDATA0	O	O	O	O	O	
	AUDCK	O	O	O	O	O	
	AUDSYNC	H	H	H	H	H	
E10A インタフェース	$\overline{\text{ASEBRKAK}}$	O	O	O	O	O	
	$\overline{\text{ASEMD0}}$	I	I	I	I	I	

- 【注】 *1 クロック動作モードに依存します。
- *2 「Z」もしくは「I、O」は、レジスタ設定に依存します。
「I」もしくは「O」は、クロック動作モードに依存します。
- *3 ソフトウェアスタンバイモード時、レジスタ設定に従って「Z」もしくは「O」になります。
- *4 バス権解放時、レジスタ設定に従って「Z」もしくは「O」になります。
- *5 BSC 機能とマルチプレクスされており、初期状態でバス制御機能になる端子があります。詳しくは、表 24.1 を参照してください。

B. 外部水晶発振子使用上の注意

外部水晶発振子周辺の設計では、水晶発振子のメーカーと評価検討の上、回路定数を決めてください。以下に、キンセキ株式会社で実施した回路検討例を示します。

B.1 推奨回路図



x-tal仕様

型名	CX-53G
サイズ	5.0 × 3.2 × 1.3mm

検討結果：

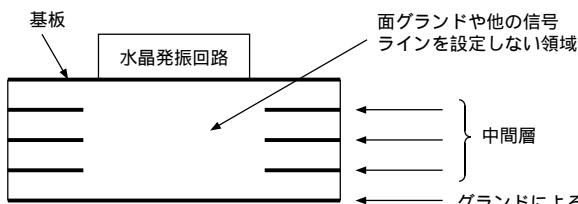
水晶発振子型名	CX-53G	CX-53G	CX-53G	
周波数 (kHz)	11289.6	18432	30000	
オーバートーン次数	基本波	基本波	基本波	
製作負荷容量 (pF)	8.0	8.0	8.0	
推奨回路定数	Rf	No connect.	No connect.	No connect.
	Rd	0	0	0
	Rx	0	0	0
	C1	7pF	7pF	8pF
	C2	7pF	7pF	8pF
推奨定数での特性	負性抵抗 ()	-5442	-2826	-943
	回路負荷容量 (pF)	7.93	7.91	8.07
	周波数偏差 (× 10 ⁻⁶)	+1.1	+1.5	-1.7
	励振電力 (μW)	83	125	65

【注】 水晶発振回路の特性は回路定数や実装状態などにより変化致します。
 水晶振動子と水晶発振回路のマッチングに関しましては十分ご確認の上、ご使用下さるよう、お願い申し上げます。
 又、推奨定数や特性はお客様が製作された製品の特性を保証するものではありませんことを予め御了承願います。
 又、基板の設計に関する部分については「B2. 基板設計時のご注意」を参照ください。

B.2 基板設計時のご注意

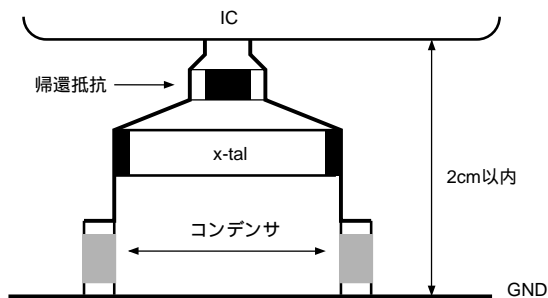
発振回路を設計する場合は次の項目にご注意ください。

- 発振回路はできるだけ発振用ICの近くに配置する。
- 水晶発振回路を形成するパーツの下部のグラントは抜く。
- 配線は最短にする（2cm以内）
- 水晶発振動子の両端に接続されるパターン同士は接近させない。
- 発振回路のシールドが必要な場合は必要最小限の面積にする。



【注】 太い線は発振回路以外の信号ラインや面グラント

基板断面図（多層基板の場合）



基板平面図

本付録に関するお問い合わせは、キンセキ株式会社の下記窓口にお問い合わせください。

- | | |
|-----------------|--|
| キンセキ株式会社（本社事務所） | 〒201-8648 東京都狛江市和泉本町1-8-1
http://www.kinseki.co.jp/ E-mail: kanriks@kinseki.co.jp |
| 国内営業部 | TEL (03) 5497-3111 FAX (03) 5497-3209 |
| 海外営業部 | TEL (03) 5497-3111 FAX (03) 5497-3209 |
| キンセキ株式会社関西営業所 | 〒532-0003 大阪市淀川区宮原3-5-36 新大阪MTビル2号館12階
TEL (06) 4807-4351 FAX (06) 4807-4352 |
| キンセキ株式会社中部営業所 | 〒464-0026 愛知県名古屋市千種区井上町66星が丘ISビル3階
TEL (052) 782-2011 FAX (052) 782-8066 |
| キンセキ株式会社東北営業所 | 〒981-3133 宮城県仙台市泉区泉中央1-23-6 トラストセンタービル5階
TEL (022) 373-8844 FAX (022) 373-7551 |

C. 外形寸法図

外形寸法図については、「ルネサス半導体パッケージ」に掲載されている寸法図を優先します。

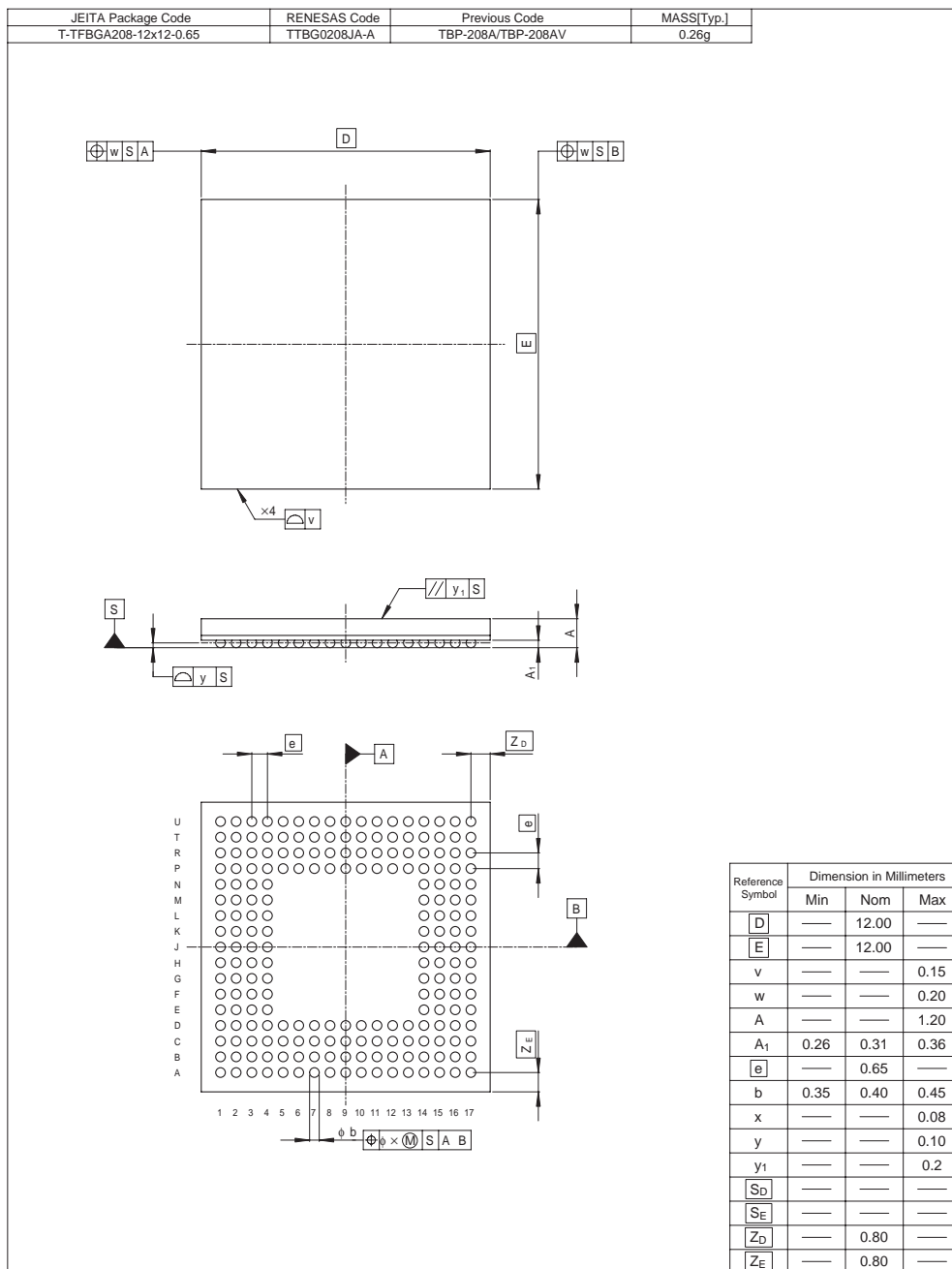


図 B.1 TBP-208A 外形寸法図

索引

16 ビットと 32 ビット ディスプレースメント	2-16	T ビット	2-16
API	21-10	U0 領域	2-4
AUD	26-1	USB ピンマルチプレクスコントローラ	18-1
Bluetooth HCI/TCI コマンド	21-10	USB ファンクションコントローラ	20-1
CPU	2-1	USB ホストコントローラ	19-1
CPU 拡張命令	3-7	USB 標準コマンド	20-49
D/A コンバータ	22-1	Uxy 領域	2-5
DSP ステータスレジスタ	3-29	U メモリ	7-1
DSP データ演算命令	3-28	Voice Codec IC	21-5
DSP データ転送命令	3-18	X/Y メモリ	6-1
DSP レジスタ	3-6, 3-28	アイソクロナスイン転送	20-45
DSP 演算ユニット	3-1	アクセスウェイト制御	9-34
DSP 命令	3-31	アドレスアレイ	5-2, 5-9
FIFO 内蔵シリアルコミュニケーション	16-1	アドレスアレイライト (連想あり)	5-10
FIFO 付きシリアルインタフェース	15-1	アドレスアレイライト (連想なし)	5-10
I/O ポート	23-1	アドレスアレイリード	5-9
ICR	8-6, 8-7	アドレッシングモード	2-17
IMCR	8-10	インターバルタイマモード	12-7
IMR	8-9	インターミットtentモード	10-27
IPRA ~ IPRH	8-4	インタラプトイン転送	20-42
IRQ 割り込み	8-11	ウォッチドッグタイマ	12-1
IRR	8-9	ウォッチドッグタイマモード	12-7
JTAG	26-1	オートリクエストモード	10-17
LRU	5-2	オートリロードカウンタ動作	14-7
NMI 割り込み	8-11	キャッシュ	5-1
P0 領域	2-4	グローバルベースレジスタ	2-13
P1 領域	2-4	クロックパルス発振器	11-1
P2 領域	2-4	クロックリジューム制御機能	21-9
P3 領域	2-4	クロック動作モード	11-5
P4 領域	2-4	クロック同期式モード	16-32
PC トレース	25-20	コントロールレジスタ	2-7, 2-11
RF IC	21-4	コントロール転送	20-36
SDRAM インタフェース	9-37	サイクルスチールモード	10-27
SPI モード	15-43	シーケンシャルブレイク	25-18
TAP コントローラ	26-9	システムレジスタ	2-7, 2-10
TCR	14-5	シャドウ空間	9-5
TSTR	14-4	シリアルクロック	15-24

シングルアドレスモード	10-25	ライトミス	5-8
ステータスレジスタ	2-7, 2-11, 3-4	ラウンドロビンモード	10-20
ストール動作	20-50	リードヒット	5-7
スリープモード	13-8	リードミス	5-7
セルフリフレッシュ	9-53	リセット	4-7
ソフトウェアスタンバイモード	13-8	リセット状態	2-1
ソフトウェアスタンバイ解除	12-6	リテラル定数	2-16
タイマユニット	14-1	リトルエンディアン	2-14, 9-26
ダイレクトメモリアクセスコントローラ	10-1	リビートエンドレジスタ	3-5
ダブルデータ転送命令	3-19	リビートスタートレジスタ	3-5
データアライメント	9-26	リビートモード転送	10-17
データアレイ	5-2, 5-10	リビート制御命令	3-7
データアレイライト	5-11	リフレッシュ	9-52
データアレイリード	5-10	レジスタ	
デュアルアドレスモード	10-23	BAMRA	25-4, 27-7, 27-23, 27-29
バースト ROM インタフェース	9-59	BAMRB	25-6, 27-7, 27-23, 27-29
バーストモード	10-28	BARA	25-3, 27-7, 27-23, 27-29
バイト選択付き SRAM インタフェース	9-60	BARB	25-5, 27-7, 27-23, 27-29
バスアービトラーション	9-65	BBRA	25-4, 27-7, 27-23, 27-29
バスクロック	11-1	BBRB	25-8, 27-7, 27-23, 27-29
バスステートコントローラ	9-1	BDMRB	25-7, 27-7, 27-22, 27-29
バルクアウト転送	20-44	BDRB	25-6, 27-7, 27-22, 27-29
バルクイン転送	20-43	BETR	25-11, 27-7, 27-22, 27-29
ビッグエンディアン	2-14, 9-26, 17-7	BRCR	25-9, 27-7, 27-22, 27-29
ピンファンクションコントローラ	24-1	BRDR	25-14, 27-7, 27-23, 27-29
ブート機能	17-1	BRSR	25-13, 27-7, 27-23, 27-29
プリフェッチヒット	5-8	CCR1	5-3, 27-2, 27-8, 27-24
プリフェッチミス	5-8	CCR2	5-4, 27-2, 27-8, 27-24
ブルートゥースインタフェース	21-1	CHCR	10-5, 27-3, 27-12, 27-25
プログラムカウンタ	2-7, 2-10	CMNCR	9-8, 27-2, 27-9, 27-24
プロシジャーレジスタ	2-10	CS0BCR	9-9, 27-3, 27-10, 27-24
ベクタベースレジスタ	2-13	CS0WCR	9-12, 27-3, 27-10, 27-25
ボード設計	11-11	CTLR	20-27, 27-6, 27-21, 27-28
ボーレートジェネレータ	15-24	CVR	20-26, 27-6, 27-21, 27-28
マニュアルリセット	4-7	DACR	22-3, 27-6, 27-21, 27-28
モジュールスタンバイ	13-10	DADR	22-2, 27-6, 27-21, 27-28
モジュロアドレッシング	3-24	DAR	10-4, 27-3, 27-11, 27-25
モジュロレジスタ	3-5	DASTS	20-22, 27-6, 27-21, 27-28
ユーザデバッグインタフェース	26-1	DMA	20-24, 27-6, 27-21, 27-28
ユーザブレークコントローラ	25-1	DMAOR	10-11, 27-3, 27-14, 27-25
ユーザモード	2-2	DMARS	10-13, 27-4, 27-14, 27-25
ライトヒット	5-8	DMATCR	10-4, 27-3, 27-11, 27-25

EPDR0i.....	20-16, 27-6, 27-20, 27-28	HRPSR.....	19-26, 27-6, 27-20, 27-27
EPDR0o.....	20-17, 27-6, 27-20, 27-28	HRSR	19-25, 27-6, 27-20, 27-27
EPDR0s.....	20-17, 27-6, 27-20, 27-28	IAR	10-10, 27-3, 27-12, 27-25
EPDR1.....	20-17, 27-6, 27-20, 27-28	ICR0	8-6, 27-2, 27-9, 27-24
EPDR2i.....	20-18, 27-6, 27-21, 27-28	ICR1	8-7, 27-2, 27-9, 27-24
EPDR2o.....	20-18, 27-6, 27-21, 27-28	ICR2	8-8, 27-2, 27-9, 27-24
EPDR3i.....	20-18, 27-6, 27-21, 27-28	IER0	20-14, 27-6, 27-20, 27-28
EPDR3o.....	20-19, 27-6, 27-21, 27-28	IFR0.....	20-5, 27-6, 27-20, 27-28
EPDR4.....	20-19, 27-6, 27-21, 27-28	IMCR	8-10, 27-2, 27-9, 27-24
EPDR5.....	20-19, 27-6, 27-21, 27-28	IMR.....	8-9, 27-2, 27-9, 27-24
EPDR6.....	20-19, 27-6, 27-21, 27-28	INTEVT2	4-3, 27-2, 27-8, 27-24
EPIR	20-28, 27-6, 27-21, 27-28	IPR	8-4, 27-2, 27-8, 27-24
EPSTL.....	20-25, 27-6, 27-21, 27-28	IRR0	8-9, 27-2, 27-9, 27-24
EPSZ0o.....	20-20, 27-6, 27-21, 27-28	ISR0	20-11, 27-6, 27-20, 27-28
EPSZ2o.....	20-20, 27-6, 27-21, 27-28	NCCR	24-13, 27-7, 27-22, 27-29
EPSZ3o.....	20-20, 27-6, 27-21, 27-28	PACR	24-3, 27-7, 27-22, 27-29
EPSZ6	20-20, 27-6, 27-21, 27-28	PADR	23-2, 27-6, 27-21, 27-28
EXCPGCR.....	18-3, 27-5, 27-17, 27-27	PBCR	24-4, 27-7, 27-22, 27-29
EXPEVT	4-2, 27-2, 27-8, 27-24	PBDR	23-3, 27-6, 27-21, 27-28
FCLR	20-23, 27-6, 27-21, 27-28	PDCR	24-6, 27-7, 27-22, 27-29
FRQCR.....	11-8, 27-4, 27-14, 27-25	PDDR	23-5, 27-6, 27-21, 27-28
HBCEDR	19-15, 27-5, 27-19, 27-27	PECR	24-7, 27-7, 27-22, 27-29
HBHEDR	19-15, 27-5, 27-18, 27-27	PEDR	23-7, 27-7, 27-21, 27-28
HCCEDR	19-15, 27-5, 27-18, 27-27	PGCR	24-8, 27-7, 27-22, 27-29
HCHEDR	19-15, 27-5, 27-18, 27-27	PGDR	23-9, 27-7, 27-21, 27-29
HCSR	19-6, 27-5, 27-17, 27-27	PHCR	24-9, 27-7, 27-22, 27-29
HCTLR.....	19-3, 27-5, 27-17, 27-27	PHDR	23-11, 27-7, 27-22, 27-29
HDHEDR	19-16, 27-5, 27-19, 27-27	PSELA.....	24-10, 27-7, 27-22, 27-29
HFIR	19-16, 27-5, 27-19, 27-27	RTCNT	9-25, 27-3, 27-11, 27-25
HFNR.....	19-19, 27-5, 27-19, 27-27	RTCOR.....	9-25, 27-3, 27-11, 27-25
HFRR.....	19-18, 27-5, 27-19, 27-27	RTCSR.....	9-24, 27-3, 27-11, 27-25
HHCCAR	19-14, 27-5, 27-18, 27-27	RWTCNT.....	9-26
HIDR.....	19-12, 27-5, 27-18, 27-27	SAR	10-4, 27-3, 27-11, 27-25
HIER	19-10, 27-5, 27-18, 27-27	SCBRR.....	16-17, 27-4, 27-16, 27-26
HISR	19-8, 27-5, 27-18, 27-27	SCFCR.....	16-19, 27-5, 27-16, 27-26
HIZCRA	24-11, 27-7, 27-22, 27-29	SCFDR.....	16-21, 27-5, 27-17, 27-27
HLSTR.....	19-21, 27-5, 27-19, 27-27	SCFER	16-12, 27-5, 27-16, 27-26
HPCEDR	19-14, 27-5, 27-18, 27-27	SCFRDR	16-5, 27-5, 27-17, 27-27
HPSR.....	19-20, 27-5, 27-19, 27-27	SCFTDR.....	16-6, 27-5, 27-17, 27-27
HRDRA.....	19-22, 27-6, 27-19, 27-27	SCRSR.....	16-5
HRDRB.....	19-24, 27-6, 27-20, 27-27	SCSCR.....	16-8, 27-5, 27-16, 27-26
HREVR.....	19-3, 27-5, 27-17, 27-27	SCSMR	16-6, 27-4, 27-16, 27-26

SCSSR	16-13, 27-5, 27-16, 27-26	WTCNT	12-3, 27-4, 27-14, 27-25
SCTDSR	16-22, 27-5, 27-16, 27-26	WTCSR	12-3, 27-4, 27-14, 27-25
SCTSR	16-5	ロードストアアーキテクチャ	2-15
SDBPR	26-3	一般例外	4-8
SDBSR	26-4	外部アドレス空間	2-6
SDCR	9-21, 27-3, 27-11, 27-25	外部リクエストモード	10-17
SDID	26-8, 27-7, 27-23, 27-29	割り込みコントローラ	8-1
SDIR	26-3, 27-7, 27-23, 27-29	割り込み要因	8-12
SDMR3	27-3, 27-11, 27-25	空間分割	9-5
SICDAR	15-21, 27-4, 27-15, 27-26	実効アドレス	2-17
SICTR	15-6, 27-4, 27-15, 27-26	受信マージン	16-45
SIFCTR	15-16, 27-4, 27-15, 27-26	周辺クロック	11-1
SIIER	15-15, 27-4, 27-16, 27-26	処理モード	2-2
SIMDR	15-4, 27-4, 27-15, 27-26	状態遷移	13-11
SIRCR	15-10, 27-4, 27-16, 27-26	積和レジスタ	2-10
SIRDAR	15-20, 27-4, 27-15, 27-26	絶対アドレス	2-16
SIRDRE	15-8, 27-4, 27-16, 27-26	多重割り込み	8-16
SISCR	15-18, 27-4, 27-15, 27-26	退避プログラムカウンタ	2-12
SISTR	15-10, 27-4, 27-16, 27-26	遅延分岐	2-16
SITCR	15-9, 27-4, 27-16, 27-26	調歩同期式モード	16-22
SITDAR	15-19, 27-4, 27-15, 27-26	低消費電力モード	13-1
SITDR	15-8, 27-4, 27-16, 27-26	低消費電力状態	2-1
SPICR	15-22, 27-4, 27-16, 27-26	特権モード	2-2
STBCR	13-4, 27-4, 27-14, 27-26	内蔵周辺モジュールリクエストモード	10-18
STBCR2	13-5, 27-4, 27-14, 27-26	内蔵周辺モジュール割り込み	8-12
STBCR3	13-6, 27-4, 27-14, 27-26	内部クロック	11-1
STBCR4	13-6, 27-4, 27-14, 27-26	汎用レジスタ	2-7, 2-9
TCNT	14-6, 27-4, 27-14, 27-26	物理アドレス空間	2-5
TCOR	14-6, 27-4, 27-14, 27-26	命令形式	2-20
TCR	14-5, 27-4, 27-14, 27-26	例外コード	4-4
TEA	4-3, 27-2, 27-8, 27-24	例外処理	4-1
TRA	4-2, 27-2, 27-8, 27-24	例外処理ベクタアドレス	4-4
TRG	20-21, 27-6, 27-21, 27-28	例外処理状態	2-1
TSR	20-26, 27-6, 27-21, 27-28	論理アドレス空間	2-3
TSTR	14-4, 27-4, 27-14, 27-26		

ルネサス32ビットRISCマイクロコンピュータ
ハードウェアマニュアル
SH7660

発行年月日 2004年1月26日 Rev.1.00
2007年9月20日 Rev.3.00

発行 株式会社ルネサス テクノロジ 営業統括部
〒100-0004 東京都千代田区大手町 2-6-2

編集 株式会社ルネサスソリューションズ
グローバルストラテジックコミュニケーション本部
カスタマサポート部

株式会社ルネサス テクノロジ 営業統括部 〒100-0004 東京都千代田区大手町2-6-2 日本ビル

営業お問合せ窓口
株式会社ルネサス販売

RENESAS

<http://www.renesas.com>

本			社	〒100-0004	千代田区大手町2-6-2 (日本ビル)	(03) 5201-5350
京	浜	支	社	〒212-0058	川崎市幸区鹿島田890-12 (新川崎三井ビル)	(044) 549-1662
西	東	京	支	〒190-0023	立川市柴崎町2-2-23 (第二高島ビル2F)	(042) 524-8701
東	北	支	社	〒980-0013	仙台市青葉区花京院1-1-20 (花京院スクエア13F)	(022) 221-1351
い	わ	き	支	〒970-8026	いわき市平小太郎町4-9 (平小太郎ビル)	(0246) 22-3222
茨	城	支	店	〒312-0034	ひたちなか市堀口832-2 (日立システムプラザ勝田1F)	(029) 271-9411
新	潟	支	店	〒950-0087	新潟市東大通1-4-2 (新潟三井物産ビル3F)	(025) 241-4361
松	本	支	社	〒390-0815	松本市深志1-2-11 (昭和ビル7F)	(0263) 33-6622
中	部	支	社	〒460-0008	名古屋市中区栄4-2-29 (名古屋広小路プレイス)	(052) 249-3330
関	西	支	社	〒541-0044	大阪市中央区伏見町4-1-1 (明治安田生命大阪御堂筋ビル)	(06) 6233-9500
北	陸	支	社	〒920-0031	金沢市広岡3-1-1 (金沢パークビル8F)	(076) 233-5980
広	島	支	店	〒730-0036	広島市中区袋町5-25 (広島袋町ビルディング8F)	(082) 244-2570
島	取	支	店	〒680-0822	鳥取市今町2-251 (日本生命鳥取駅前ビル)	(0857) 21-1915
九	州	支	社	〒812-0011	福岡市博多区博多駅前2-17-1 (博多プレステージ5F)	(092) 481-7695

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

総合お問合せ窓口：コンタクトセンター E-Mail: csc@renesas.com

SH7660
ハードウェアマニュアル



ルネサス エレクトロニクス株式会社
神奈川県川崎市中原区下沼部1753 〒211-8668

RJJ09B0120-0300