

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用途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、海底中継機器、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。

RENESAS

ユーザーズ・マニュアル

保守/廃止

V831TM

32ビット・マイクロプロセッサ

ハードウェア編

μPD705101

資料番号 U12273JJ4V0UM00 (第4版)

発行年月 November 1998 N CP(K)

© NEC Corporation 1997

(メ モ)

目次要約

第1章	概 説	...	19
第2章	端子機能	...	27
第3章	CPU機能	...	37
第4章	割り込み／例外処理機能	...	39
第5章	バス制御機能	...	53
第6章	ウェイト制御機能	...	87
第7章	メモリ・アクセス制御機能	...	95
第8章	DMA機能	...	101
第9章	シリアル・インタフェース機能	...	127
第10章	タイマ／カウンタ機能	...	149
第11章	ポート機能	...	167
第12章	クロック発生機能	...	171
第13章	スタンバイ機能	...	175
第14章	リセット／NMI制御機能	...	181
第15章	ディバグ／トレース機能	...	185
付録A	レジスタ索引	...	187
付録B	総合索引	...	191

CMOSデバイスの一般的注意事項

静電気対策（MOS全般）

注意 MOSデバイス取り扱いの際は静電気防止を心がけてください。

MOSデバイスは強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、NECが出荷梱包に使用している導電性のトレーやマガジン・ケース、または導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。

また、MOSデバイスを実装したボードについても同様の扱いをしてください。

未使用入力の処理（CMOS特有）

注意 CMOSデバイスの入力レベルは固定してください。

バイポーラやNMOSのデバイスと異なり、CMOSデバイスの入力に何も接続しない状態で動作させると、ノイズなどに起因する中間レベル入力が生じ、内部で貫通電流が流れて誤動作を引き起こす恐れがあります。プルアップかプルダウンによって入力レベルを固定してください。また、未使用端子が出力となる可能性（タイミングは規定しません）を考慮すると、個別に抵抗を介して V_{DD} またはGNDに接続することが有効です。

資料中に「未使用端子の処理」について記載のある製品については、その内容を守ってください。

初期化以前の状態（MOS全般）

注意 電源投入時、MOSデバイスの初期状態は不定です。

分子レベルのイオン注入量等で特性が決定するため、初期状態は製造工程の管理外です。電源投入時の端子の出力状態や入出力設定、レジスタ内容などは保証しておりません。ただし、リセット動作やモード設定で定義している項目については、これらの動作ののちに保証の対象となります。

リセット機能を持つデバイスの電源投入後は、まずリセット動作を実行してください。

V830, V831, V830ファミリは、日本電気株式会社の商標です。

Windowsは米国Microsoft Corp.の米国およびその他の国における登録商標または商標です。

UNIXはX/Openカンパニーリミテッドがライセンスしている米国ならびに他の国における登録商標です。

本資料の内容は、後日変更する場合があります。

文書による当社の承諾なしに本資料の転載複製を禁じます。

本資料に記載された製品の使用もしくは本資料に記載の情報の使用に際して、当社は当社もしくは第三者の知的所有権その他の権利に対する保証または実施権の許諾を行うものではありません。上記使用に起因する第三者所有の権利にかかわる問題が発生した場合、当社はその責を負うものではありませんのでご了承ください。

当社は品質、信頼性の向上に努めていますが、半導体製品はある確率で故障が発生します。当社半導体製品の故障により結果として、人身事故、火災事故、社会的な損害等を生じさせない冗長設計、延焼対策設計、誤動作防止設計等安全設計に十分ご注意願います。

当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定して頂く「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の上ご使用願います。

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

特別水準：輸送機器（自動車、列車、船舶等）、交通用信号機器、防災／防犯装置、各種安全装置、生命維持を直接の目的としない医療機器

特定水準：航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器、生命維持のための装置またはシステム等

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。当社製品を上記の「標準水準」の用途以外でご使用をお考えのお客様は、必ず事前に当社販売窓口までご相談頂きますようお願い致します。

この製品は耐放射線設計をしておりません。

本版で改訂された主な箇所

箇 所	内 容
p. 37	3. 1 内蔵周辺I/O空間の記述を修正
p. 52	4. 7 外部入力端子による割り込み要求を追加
p. 53	5. 2 外部I/Oサイクルの記述を修正
p. 65	5. 5 (4) データ・バス幅による分類の記述を修正
p. 88	6. 2 アドレス空間とブロックの記述を修正
p. 94	6. 4 $\overline{\text{READY}}$ 端子によるウェイト制御を追加
p. 120	8. 7. 3 内蔵周辺ハードウェアからの要求の記述を修正
p. 125	図8 - 20 16-32ビット・データ・バス幅 (32ビット転送バス・サイジング) の修正
p. 126	図8 - 21 16-16ビット・データ・バス幅 (32ビット転送バス・サイジング) の修正
p. 143	9. 2. 5 (2) (b) 送受信動作の起動に注意を追加
p. 178	13. 2. 2 (3) $\overline{\text{RESET}}$ 端子入力による解除の記述を修正
p. 179	13. 3. 2 (2) $\overline{\text{RESET}}$ 端子入力による解除の記述を修正
p. 181	14. 3 リセットの記述を修正

本文欄外の★印は、本版で改訂された主な箇所を示しています。

巻末にアンケート・コーナを設けております。このドキュメントに対するご意見をお気軽にお寄せください。

はじめに

- 対象者** このマニュアルは、V831 (μ PD705101) の機能を理解し、それをういた応用システムを設計するユーザを対象としています。
- 目的** このマニュアルは、次に示すV831の持つハードウェア機能をユーザに理解していただくことを目的としています。
- 構成** V831のユーザズ・マニュアルは、ハードウェア編(このマニュアル)とアーキテクチャ編(V830ファミリ™ ユーザズ・マニュアル アーキテクチャ編)の2冊に分かれています。

ハードウェア編	アーキテクチャ編
・概 説	・レジスタ・セット
・端子機能	・データ・セット
・CPU機能	・アドレス空間
・内蔵周辺機能	・命 令
・付 録	・割り込みと例外

読み方 このマニュアルの読者には、電気、論理回路、およびマイクロコンピュータに関する一般知識を必要とします。

命令機能の詳細を理解しようとするとき

V830ファミリ ユーザズ・マニュアル アーキテクチャ編を参照してください。

レジスタ名が分かっている、レジスタの詳細を確認するとき

付録A レジスタ索引を利用してください。

一通りV831の機能を理解しようとするとき

目次に従ってお読みください。

- 凡 例** データ表記の重み：左が上位桁、右が下位桁
アクティブ・ロウの表記： $\overline{\text{xxx}}$ (端子、信号名称に上線)
メモリ・マップのアドレス：上部-上位、下部-下位
注：本文中につけた注の説明
注意：気をつけて読んでいただきたい内容
備考：本文の補足説明
数の表記：2進数 ... xxx または xxxxB
10進数 ... xxx
16進数 ... xxxH
2のべき数を示す接頭語(アドレス空間、メモリ容量)：
K(キロ)： $2^{10} = 1024$
M(メガ)： $2^{20} = 1024^2$

G (ギガ) : $2^{30} = 1024^3$

関連資料 関連資料は暫定版の場合がありますが、この資料では「暫定」の表示をしておりません。あらかじめご了承ください。

V831に関する資料

資料名	資料番号
V831 ユーザーズ・マニュアル ハードウェア編	このマニュアル
V830 ファミリー ユーザーズ・マニュアル アーキテクチャ編	U12496J
V831 データ・シート	U12979J

開発ツールに関する資料 (ユーザーズ・マニュアル)

資料名		資料番号	
IE-70000-MC-NW (V831用インサーキット・エミュレータ)		U12476J	
CA830 (Cコンパイラ)	操作編 (UNIX™ベース)	U11013J	
	操作編 (Windows™ベース)	U11068J	
	アセンブリ言語編	U11014J	
	C言語編	U11010J	
	プロジェクト・マネージャ編	U11991J	
RX830 (リアルタイムOS)	ITRON1	基礎編	U11730J
		インストレーション編	U11731J
		テクニカル編	U11713J
	μITRON Ver.3.0	基礎編	U13152J
		インストレーション編	U13151J
		テクニカル編	U13150J

目 次

第1章 概 説 ... 19

- 1.1 特 徴 ... 19
- 1.2 応用分野 ... 20
- 1.3 オーダ情報 ... 21
- 1.4 端子接続図 (Top View) ... 21
- 1.5 内部ブロック構成 ... 23
- 1.6 内部ユニット ... 24

第2章 端子機能 ... 27

- 2.1 端子機能一覧 ... 27
- 2.2 端子状態 ... 29
- 2.3 端子機能説明 ... 30
 - 2.3.1 アドレス・バス ... 30
 - 2.3.2 データ・バス ... 30
 - 2.3.3 バス制御信号 ... 30
 - 2.3.4 システム制御信号 ... 32
 - 2.3.5 割り込み制御信号 ... 32
 - 2.3.6 DRAM制御信号 ... 33
 - 2.3.7 DMA制御信号 ... 34
 - 2.3.8 リアルタイム・パルス制御信号 ... 34
 - 2.3.9 シリアル制御信号 ... 35
 - 2.3.10 ポート制御信号 ... 35
 - 2.3.11 デバッグ制御信号 ... 35

第3章 CPU機能 ... 37

- 3.1 内蔵周辺I/O空間 ... 37
 - 3.1.1 注意事項 ... 38
- 3.2 CPUコア・システム・レジスタ ... 38

第4章 割り込み／例外処理機能 ... 39

- 4.1 割り込み／例外処理 ... 39
 - 4.1.1 割り込み／例外処理の種類 ... 39
- 4.2 ノンマスカブル割り込み ... 41
 - 4.2.1 ノンマスカブル割り込みの処理形態 ... 41
- 4.3 マスカブル割り込み ... 42
 - 4.3.1 マスカブル割り込みの処理形態 ... 42
 - 4.3.2 マスカブル割り込みの優先順位 ... 44
- 4.4 例外処理 ... 45
- 4.5 例外／割り込みからの復帰 ... 46

- 4.5.1 例外/割り込みからの復帰 ... 46
- 4.5.2 致命的例外ルーチンからの復帰 ... 46
- 4.6 **割り込み制御レジスタ** ... 47
 - 4.6.1 割り込みグループ優先順位レジスタ (IGP) ... 47
 - 4.6.2 割り込みクリア・レジスタ (ICR) ... 48
 - 4.6.3 割り込み要求レジスタ (IRR) ... 49
 - 4.6.4 割り込み要求マスク・レジスタ (IMR) ... 49
 - 4.6.5 ICUモード・レジスタ (IMOD) ... 50
- 4.7 **外部入力端子による割り込み要求** ... 52

第5章 バス制御機能 ... 53

- 5.1 **特 徴** ... 53
- 5.2 **外部I/Oサイクル** ... 53
 - 5.2.1 バイト・アクセス制御 ... 54
- 5.3 **SRAM (ROM) サイクル** ... 55
 - 5.3.1 SRAM (ROM) シングル・サイクル ... 57
 - 5.3.2 SRAM (ROM) バースト・サイクル ... 59
 - 5.3.3 バイト・アクセス制御 ... 61
- 5.4 **Page-ROMサイクル** ... 62
 - 5.4.1 Page-ROMシングル・サイクル ... 62
 - 5.4.2 Page-ROMバースト・サイクル ... 62
- 5.5 **DRAMサイクル** ... 65
 - 5.5.1 DRAMシングル・サイクル ... 67
 - 5.5.2 DRAMバースト・サイクル ... 70
 - 5.5.3 タイミング制御 ... 75
 - 5.5.4 バイト・アクセス制御 ... 76
 - 5.5.5 リフレッシュ制御 ... 77
- 5.6 **アイドル・ステート** ... 79
- 5.7 **バス・サイジング** ... 80
- 5.8 **バス・ホールド・サイクル** ... 84
- 5.9 **バス・アービトレーション** ... 85
- 5.10 **ライト・バッファ動作** ... 85
- 5.11 **メモリ・マップトI/O** ... 86

第6章 ウェイト制御機能 ... 87

- 6.1 **特 徴** ... 87
- 6.2 **アドレス空間とブロック** ... 88
- 6.3 **ウェイト制御レジスタ** ... 89
 - 6.3.1 バス・サイクル・タイプ・コントロール・レジスタ (BCTC) ... 89
 - 6.3.2 データ・バス幅コントロール・レジスタ (DBC) ... 90
 - 6.3.3 プログラマブル・ウェイト・コントロール・レジスタ0 (PWC0) ... 91
 - 6.3.4 プログラマブル・ウェイト・コントロール・レジスタ1 (PWC1) ... 92
 - 6.3.5 プログラマブル・アイドル・コントロール・レジスタ (PIC) ... 93
- 6.4 **READY端子によるウェイト制御** ... 94
 - 6.4.1 $\overline{\text{READY}}$ 端子入力のサンプリング・タイミング ... 94
 - 6.4.2 $\overline{\text{READY}}$ 端子とプログラマブル・ウェイト両方を使用する場合 ... 94

第7章 メモリ・アクセス制御機能 ... 95

- 7.1 特 徴 ... 95
- 7.2 DRAM制御機能 ... 95
 - 7.2.1 アドレス・マルチプレクス機能 ... 96
 - 7.2.2 on-page/off-pageの判断 ... 96
 - 7.2.3 DRAMコンフィギュレーション・レジスタ (DRC) ... 96
 - 7.2.4 リフレッシュ機能 ... 98
- 7.3 Page-ROM制御機能 ... 100
 - 7.3.1 Page-ROMコンフィギュレーション・レジスタ (PRC) ... 100

第8章 DMA機能 ... 101

- 8.1 特 徴 ... 101
- 8.2 構 成 ... 102
- 8.3 DMA制御レジスタ ... 103
 - 8.3.1 DMAソース・アドレス・レジスタ0-3 (DSA0-DSA3) ... 103
 - 8.3.2 DMAデスティネーション・アドレス・レジスタ0-3 (DDA0-DDA3) ... 105
 - 8.3.3 DMAバイト・カウント・レジスタ0-3 (DBC0-DBC3) ... 106
 - 8.3.4 DMAチャンネル・コントロール・レジスタ0-3 (DCHC0-DCHC3) ... 108
 - 8.3.5 DMAコントロール・レジスタ (DC) ... 111
- 8.4 転送モード ... 112
 - 8.4.1 シングル転送モード ... 112
 - 8.4.2 デイモンド転送モード ... 113
- 8.5 DMA転送タイプと転送対象 ... 113
 - 8.5.1 2サイクル転送 ... 113
 - 8.5.2 転送対象 ... 114
- 8.6 DMAチャンネルの優先順位 ... 115
- 8.7 DMA転送要求 ... 115
 - 8.7.1 DMARQ端子からの要求 ... 115
 - 8.7.2 ソフトウェアからの要求 ... 120
 - 8.7.3 内蔵周辺ハードウェアからの要求 ... 120
- 8.8 DMA転送終了割り込み ... 121
 - 8.8.1 TCnビット参照とDMA転送終了割り込みの動作 ... 121
- 8.9 DMA転送終了出力 ... 122
- 8.10 強制中断 ... 124
 - 8.10.1 NMI信号による中断 ... 124
 - 8.10.2 HLDRQ信号, リフレッシュによる一時中断 ... 124
- 8.11 DMA転送のバス・サイジング ... 125

第9章 シリアル・インタフェース機能 ... 127

- 9.1 アシクロナス・シリアル・インタフェース (UART) ... 127
 - 9.1.1 概 要 ... 127
 - 9.1.2 特 徴 ... 128
 - 9.1.3 構 成 ... 129
 - 9.1.4 モード・レジスタとコントロール・レジスタ ... 130
 - 9.1.5 割り込み要求 ... 136

9.1.6	基本動作	...	137
9.2	クロック同期式シリアル・インタフェース (CSI)	...	139
9.2.1	特徴	...	139
9.2.2	構成	...	139
9.2.3	モード・レジスタとコントロール・レジスタ	...	140
9.2.4	端子機能	...	142
9.2.5	基本動作	...	142
9.3	ポー・レート・ジェネレータ	...	144
9.3.1	構成と機能	...	144
9.3.2	ポー・レート・ジェネレータ・コンペア・レジスタ (BRG0)	...	147
9.3.3	ポー・レート・ジェネレータ・プリスケラ・モード・レジスタ (BPRM0)	...	147
第10章 タイマ/カウンタ機能 ... 149			
10.1	特徴	...	149
10.2	構成	...	150
10.2.1	タイマ1	...	151
10.2.2	タイマ4	...	153
10.2.3	キャプチャ/コンペア・レジスタ (CC10-CC13)	...	154
10.3	タイマ/カウンタ制御レジスタ	...	156
10.3.1	タイマ・ユニット・モード・レジスタ (TUM1)	...	156
10.3.2	タイマ・コントロール・レジスタ1 (TMC1)	...	158
10.3.3	タイマ・コントロール・レジスタ4 (TMC4)	...	159
10.3.4	タイマ出力コントロール・レジスタ (TOC1)	...	160
10.3.5	ICUモード・レジスタ (IMOD)	...	161
10.3.6	タイマ・オーバフロー・ステータス・レジスタ (TOVS)	...	161
10.4	動作	...	162
10.4.1	タイマ1	...	162
10.4.2	タイマ4	...	163
10.5	注意事項	...	164
第11章 ポート機能 ... 167			
11.1	構成	...	167
11.2	ポート制御レジスタ	...	169
11.2.1	入出力ポート・レジスタ (PORT)	...	169
11.2.2	入出力モード・レジスタ (PM)	...	169
11.2.3	ポート・コントロール・モード・レジスタ (PC)	...	170
第12章 クロック発生機能 ... 171			
12.1	構成	...	171
12.2	入力クロック選択	...	172
12.2.1	ロックアップ時間	...	172
12.3	クロック出力制御	...	172
12.3.1	クロック出力禁止モード	...	172
12.4	クロック制御レジスタ	...	173
12.4.1	クロック・コントロール・レジスタ (CGC)	...	173

12.4.2 PLLコントロール・レジスタ (PLLCR) ... 174

第13章 スタンバイ機能 ... 175

- 13.1 スタンバイ・モード ... 175
- 13.2 HALTモード ... 176
 - 13.2.1 HALTモードの設定と動作状態 ... 176
 - 13.2.2 HALTモードの解除 ... 177
- 13.3 STOPモード ... 178
 - 13.3.1 STOPモードの設定と動作状態 ... 178
 - 13.3.2 STOPモードの解除 ... 179
- 13.4 発振安定時間の確保 ... 179

第14章 リセット/NMI制御機能 ... 181

- 14.1 特 徴 ... 181
- 14.2 ノンマスカブル割り込み (NMI) ... 181
- 14.3 リセット ... 181
 - 14.3.1 端子機能 ... 181
 - 14.3.2 イニシャライズ ... 183

第15章 ディバグ/トレース機能 ... 185

- 15.1 特 徴 ... 185

付録A レジスタ索引 ... 187

- A.1 50音順 ... 187
- A.2 アルファベット順 ... 189

付録B 総合索引 ... 191

- B.1 50音で始まる語句の索引 ... 191
- B.2 アルファベットで始まる語句の索引 ... 193

図の目次 (1/4)

図番号	タイトル, ページ
3 - 1	内蔵周辺I/Oマップ ... 37
4 - 1	ノンマスカブル割り込み要求の処理フロー ... 41
4 - 2	マスカブル割り込み要求の処理フロー ... 43
4 - 3	例外処理フロー ... 45
4 - 4	例外 / 割り込みからの復帰フロー ... 46
4 - 5	致命的例外ルーチンからの復帰フロー ... 46
4 - 6	割り込みグループ優先順位レジスタ (IGP) ... 47
4 - 7	割り込みクリア・レジスタ (ICR) ... 48
4 - 8	割り込み要求レジスタ (IRR) ... 49
4 - 9	割り込み要求マスク・レジスタ (IMR) ... 49
4 - 10	ICUモード・レジスタ (IMOD) ... 51
5 - 1	外部I/Oサイクル (32ビット・バス・モード) ... 54
5 - 2	16M ROM (1M×16) との接続例 (32ビット・バス・モード時) ... 55
5 - 3	1M SRAM (128K×8) との接続例 (32ビット・バス・モード時) ... 56
5 - 4	SRAM (ROM) シングル・サイクル (32ビット・バス・モード) ... 58
5 - 5	SRAM (ROM) バースト・サイクル ... 60
5 - 6	Page-ROMバースト・サイクル (32ビット・バス・モード) ... 63
5 - 7	Page-ROMバースト・サイクル (16ビット・バス・モード, 8バイト・ページ・サイズ) ... 64
5 - 8	16M EDO DRAM (1M×16) との接続例 (16ビット・バス・モード時) ... 66
5 - 9	16M EDO DRAM (1M×16) との接続例 (32ビット・バス・モード時) ... 66
5 - 10	DRAMシングル1クロックCAS on-page/off-pageサイクル (32ビット・バス・モード) ... 68
5 - 11	DRAMシングル2クロックCAS on-page/off-pageサイクル (32ビット・バス・モード) ... 69
5 - 12	DRAMバースト1クロックCAS off-pageサイクル (32ビット・バス・モード) ... 71
5 - 13	DRAMバースト1クロックCAS on-pageサイクル (32ビット・バス・モード) ... 72
5 - 14	DRAMバースト2クロックCAS off-pageサイクル (32ビット・バス・モード) ... 73
5 - 15	DRAMバースト2クロックCAS on-pageサイクル (32ビット・バス・モード) ... 74
5 - 16	DRAMアクセス・タイミング (バーストoff-pageサイクル) ... 75
5 - 17	CBRリフレッシュ・サイクル (RFW1, 0 = 01, RP = 1のとき) ... 77
5 - 18	CBRリフレッシュ・サイクル (RFW1, 0 = 00, RP = 0のとき) ... 78
5 - 19	CBRセルフ・リフレッシュ・サイクル (NMIによる解除, RP = 0のとき) ... 78
5 - 20	CBRセルフ・リフレッシュ・サイクル (RESETによる解除, RP = 0のとき) ... 79
5 - 21	DRAMシングル・サイクルでバス・サイジングによって追加されたアクセス ... 80
5 - 22	DRAMバースト・サイクルでバス・サイジングによって追加されたアクセス ... 81
5 - 23	I/Oサイクルでバス・サイジングによって追加されたアクセス ... 82
5 - 24	SRAMシングル・サイクルでバス・サイジングによって追加されたアクセス ... 83
5 - 25	バス・ホールド・サイクル ... 84

図の目次 (2/4)

図番号	タイトル, ページ
6 - 1	アドレス空間 ... 88
6 - 2	バス・サイクル・タイプ・コントロール・レジスタ (BCTC) ... 89
6 - 3	データ・バス幅コントロール・レジスタ (DBC) ... 90
6 - 4	プログラマブル・ウェイト・コントロール・レジスタ0 (PWC0) ... 91
6 - 5	プログラマブル・ウェイト・コントロール・レジスタ1 (PWC1) ... 92
6 - 6	プログラマブル・アイドル・コントロール・レジスタ (PIC) ... 93
7 - 1	ロウ・アドレス, カラム・アドレスの出力 ... 96
7 - 2	DRAMコンフィギュレーション・レジスタ (DRC) ... 97
7 - 3	リフレッシュ・コントロール・レジスタ (RFC) ... 99
7 - 4	Page-ROMコンフィギュレーション・レジスタ (PRC) ... 100
8 - 1	DMACブロック図 ... 102
8 - 2	DMAソース・アドレス・レジスタ0H-3H (DSA0H-DSA3H) ... 103
8 - 3	DMAソース・アドレス・レジスタ0L-3L (DSA0L-DSA3L) ... 104
8 - 4	DMAデスティネーション・アドレス・レジスタ0H-3H (DDA0H-DDA3H) ... 105
8 - 5	DMAデスティネーション・アドレス・レジスタ0L-3L (DDA0L-DDA3L) ... 106
8 - 6	DMAバイト・カウント・レジスタ0-3 (DBC0-DBC3) ... 107
8 - 7	DMAチャンネル・コントロール・レジスタ0-3 (DCHC0-DCHC3) ... 108
8 - 8	DMAコントロール・レジスタ (DC) ... 111
8 - 9	シングル転送の例1 ... 112
8 - 10	シングル転送の例2 ... 112
8 - 11	ダイヤモンド転送の例 ... 113
8 - 12	2サイクル・ダイヤモンド転送 (外部I/O DRAM (on-page)) ... 114
8 - 13	2サイクル・ダイヤモンド転送 (16ビットSRAM 32ビットDRAM, 32ビット転送時) ... 116
8 - 14	2サイクル・ダイヤモンド転送 (32ビットDRAM 16ビットSRAM, 32ビット転送時) ... 117
8 - 15	シングル転送 (16ビットSRAM 32ビットDRAM, 32ビット転送時) ... 118
8 - 16	シングル転送 (8ビットI/O 32ビットSRAM, 8ビット転送時) ... 119
8 - 17	内蔵周辺ハードウェアからの要求による転送例 ... 121
8 - 18	チャンネル0とチャンネル1の転送終了処理 ... 122
8 - 19	DMA転送終了出力タイミング ... 123
8 - 20	16-32ビット・データ・バス幅 (32ビット転送バス・サイジング) ... 125
8 - 21	16-16ビット・データ・バス幅 (32ビット転送バス・サイジング) ... 126
9 - 1	UARTのブロック図 ... 129
9 - 2	アシンクロナス・シリアル・インタフェース・モード・レジスタ00 (ASIM00) ... 130
9 - 3	アシンクロナス・シリアル・インタフェース・モード・レジスタ01 (ASIM01) ... 132
9 - 4	アシンクロナス・シリアル・インタフェース・ステータス・レジスタ (ASIS0) ... 133
9 - 5	受信バッファ (RXB0, RXB0L) ... 134

図の目次 (3/4)

図番号	タイトル, ページ
9 - 6	送信シフト・レジスタ (TXS0, TXS0L) ... 135
9 - 7	UARTの送信データ・フォーマット ... 137
9 - 8	CSIのブロック図 ... 139
9 - 9	クロック同期式シリアル・インタフェース・モード・レジスタ0 (CSIM0) ... 140
9 - 10	シリアルI/Oシフト・レジスタ0 (SIO0) ... 141
9 - 11	CSIの転送タイミング ... 142
9 - 12	ポー・レート・ジェネレータ (BRG) のブロック構成 ... 144
9 - 13	ポー・レート・ジェネレータ・コンペア・レジスタ (BRG0) ... 147
9 - 14	ポー・レート・ジェネレータ・プリスケラ・モード・レジスタ (BPRM0) ... 147
10 - 1	タイマ1のブロック構成 ... 150
10 - 2	タイマ4のブロック構成 ... 151
10 - 3	タイマ1 (TM1) ... 151
10 - 4	タイマ4 (TM4) ... 153
10 - 5	コンペア・レジスタ (CM4) ... 153
10 - 6	キャプチャ/コンペア・レジスタ (CC10-CC13) ... 154
10 - 7	タイマ・ユニット・モード・レジスタ (TUM1) ... 156
10 - 8	タイマ・コントロール・レジスタ1 (TMC1) ... 158
10 - 9	タイマ・コントロール・レジスタ4 (TMC4) ... 159
10 - 10	タイマ出力コントロール・レジスタ (TOC1) ... 160
10 - 11	タイマ・オーバフロー・ステータス・レジスタ (TOVS) ... 161
10 - 12	タイマ1の基本動作 ... 162
10 - 13	キャプチャ動作例 ... 163
10 - 14	タイマ4の基本動作 ... 163
10 - 15	コンペア動作例 ... 164
11 - 1	ポート0のブロック図 ... 167
11 - 2	ポート1のブロック図 ... 168
11 - 3	ポート2のブロック図 ... 168
11 - 4	入出力ポート・レジスタ (PORT) ... 169
11 - 5	入出力モード・レジスタ (PM) ... 169
11 - 6	ポート・コントロール・モード・レジスタ (PC) ... 170
12 - 1	クロック発生機能のブロック図 ... 171
12 - 2	クロック出力禁止モード ... 172
12 - 3	クロック・コントロール・レジスタ (CGC) ... 173
12 - 4	PLLコントロール・レジスタ (PLLCR) ... 174
13 - 1	状態遷移図 ... 176

図の目次 (4/4)

図番号	タイトル, ページ
13 - 2	STOPモード解除タイミング (NMI信号入力時) ... 179
13 - 3	STOPモード解除タイミング (RESET信号入力時) ... 180
14 - 1	リセット信号の受け付け ... 182

表の目次

表番号	タイトル, ページ
2 - 1	各端子の状態 ... 29
4 - 1	割り込み一覧 ... 39
4 - 2	割り込み一覧 (マスカブル割り込み) ... 40
4 - 3	優先順位と例外コード, ハンドラ・アドレス, 割り込み優先の関係 ... 48
4 - 4	割り込み制御レジスタの各ビットと割り込み要求信号の対応 ... 50
5 - 1	32ビット・データ・バス ($\overline{\text{xxMWR}}$) ... 61
5 - 2	16ビット・データ・バス ($\overline{\text{xxMWR}}$) ... 61
5 - 3	32ビット・データ・バス ($\overline{\text{xxCAS}}$) ... 76
5 - 4	16ビット・データ・バス ($\overline{\text{xxCAS}}$) ... 76
5 - 5	$\overline{\text{RAS}}$ アクティブ期間 ... 77
5 - 6	アイドル期間中のバス制御信号の値 ... 79
7 - 1	on-page/off-page判断で比較するアドレス ... 96
9 - 1	スタート条件 ... 143
9 - 2	BRG設定データ ... 146
10 - 1	キャプチャ/コンペア・レジスタ ... 155
11 - 1	コントロール・モード時の動作 ... 167
12 - 1	PLLシンセサイザによる逡倍機能 ... 171
13 - 1	スタンバイ制御によるクロック・ジェネレータの動作 ... 175
13 - 2	HALTモード時の動作状態 ... 177
13 - 3	割り込み要求によるHALTモード解除後の動作 ... 177
13 - 4	STOPモード時の動作状態 ... 178
14 - 1	リセット直後の出力端子の状態 ... 182
14 - 2	各レジスタのリセット後の初期値 ... 183

第1章 概 説

V831は、組み込み制御向け高性能32ビット・マイクロプロセッサV830™ (μ PD705100) をプロセッサ・コアに使用し、DRAM/ROMコントローラ、4チャンネルのDMAコントローラ、リアルタイム・パルス・ユニット、シリアル・インタフェース、割り込みコントローラなどの周辺機能を内蔵した32ビットRISCマイクロプロセッサです。

V831は、高い割り込み応答性、最適化されたパイプライン構造に加え、マルチメディア機能を実現するために積和演算命令、連結シフト命令、高速分岐命令などを持ち、インターネット/イントラネット機器、カー・ナビゲーション、高性能テレビ、カラーFAXなどのマルチメディア機器分野できわめて高いパフォーマンスを実現できます。

1.1 特 徴

CPU機能

V830命令互換

命令キャッシュ	: 4Kバイト
命令RAM	: 4Kバイト
データ・キャッシュ	: 4Kバイト
データRAM	: 4Kバイト
最小命令実行サイクル数	: 1サイクル
メモリ空間, I/O空間	: 各4Gバイト
汎用レジスタ数	: 32ビット×32本

割り込み / 例外処理機能

ノンマスカブル	: 外部入力 1本
マスカブル	: 外部入力 8本 (4本は内部と兼用)
内部要因	: 11種類

4つのグループごとの優先順位を指定可能

バス制御機能

チップ・セレクト出力: 8ブロック ($\overline{\text{RAS}} + \overline{\text{CS}}7$ 本)

4つの $\overline{\text{CS}}$ に対しメモリとI/O空間の選択が可能

各ブロックのリニア・アドレス空間: 16Mバイト

ウェイト制御機能

DRAM空間	: 0, 1ウェイトのソフトウェア制御
その他メモリ空間	: 0~7ウェイトのソフトウェア制御
I/O空間	: 0~15ウェイトのソフトウェア制御
アイドル・ステート	: 0~3ステート挿入可能

メモリ・アクセス制御機能

DRAMのハイパー・ページ・モード対応

Page-ROMのページ・モード対応

DMA機能

4チャンネル

最大転送回数 : 16, 777, 216 (2^{24})

転送タイプ : 2サイクル転送

転送モード : シングル転送, デイモンド転送

プログラマブル・ウエイト機能

シリアル・インタフェース機能

非同期式シリアル・インタフェース (UART) : 1チャンネル

同期式シリアル・インタフェース (CSI) : 1チャンネル

専用ポー・レート・ジェネレータ内蔵 : 1チャンネル

タイマ・カウンタ機能

16ビット・タイマ/イベント・カウンタ : 1チャンネル

タイマ出力 : 2本

16ビット・キャプチャ/コンペア・レジスタ : 4本

16ビット・インターバル・タイマ : 1チャンネル

ポート機能

3本の入出力ポート

クロック発生機能

PLLクロック・シンセサイザ

スタンバイ機能

HALT, STOPモード

デバッグ機能

デバッグ専用同期式シリアル・インタフェース : 1チャンネル

トレース専用インタフェース : 1チャンネル

各種トレース機能 : 分岐PCトレース, データ・トレース

パッケージ

160ピン・プラスチックLQFP (24mm)

1.2 応用分野

- ・インターネット/イントラネット機器
- ・カー・ナビゲーション
- ・高性能テレビ
- ・カラーFAX

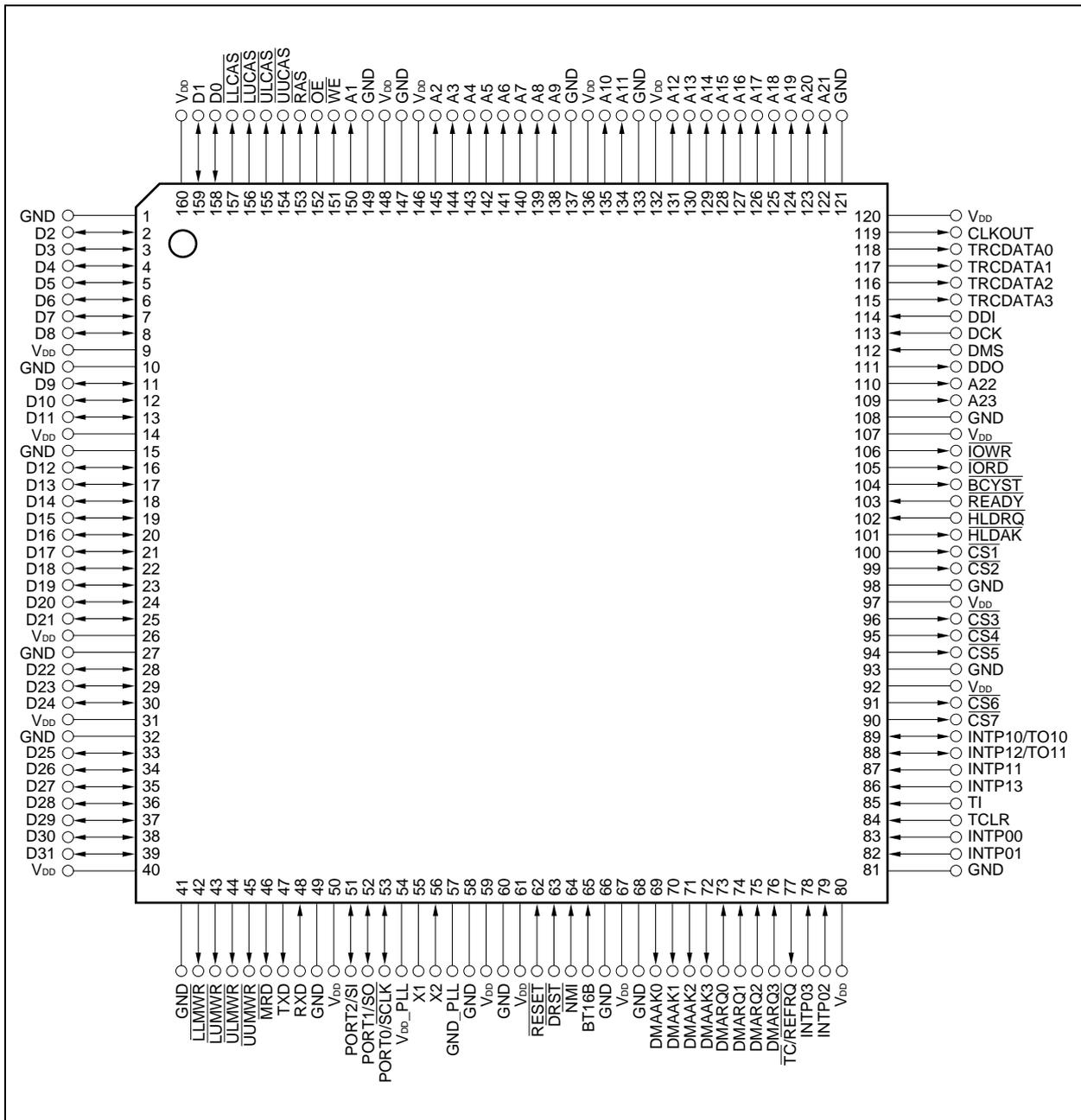
1.3 オーダ情報

オーダ名称	パッケージ
μ PD705101GM-100-8ED	160ピン・プラスチックLQFP(ファインピッチ)(24 mm)

1.4 端子接続図 (Top View)

・ 160ピン・プラスチックLQFP (ファインピッチ) (24 mm)

μ PD705101GM-100-8ED

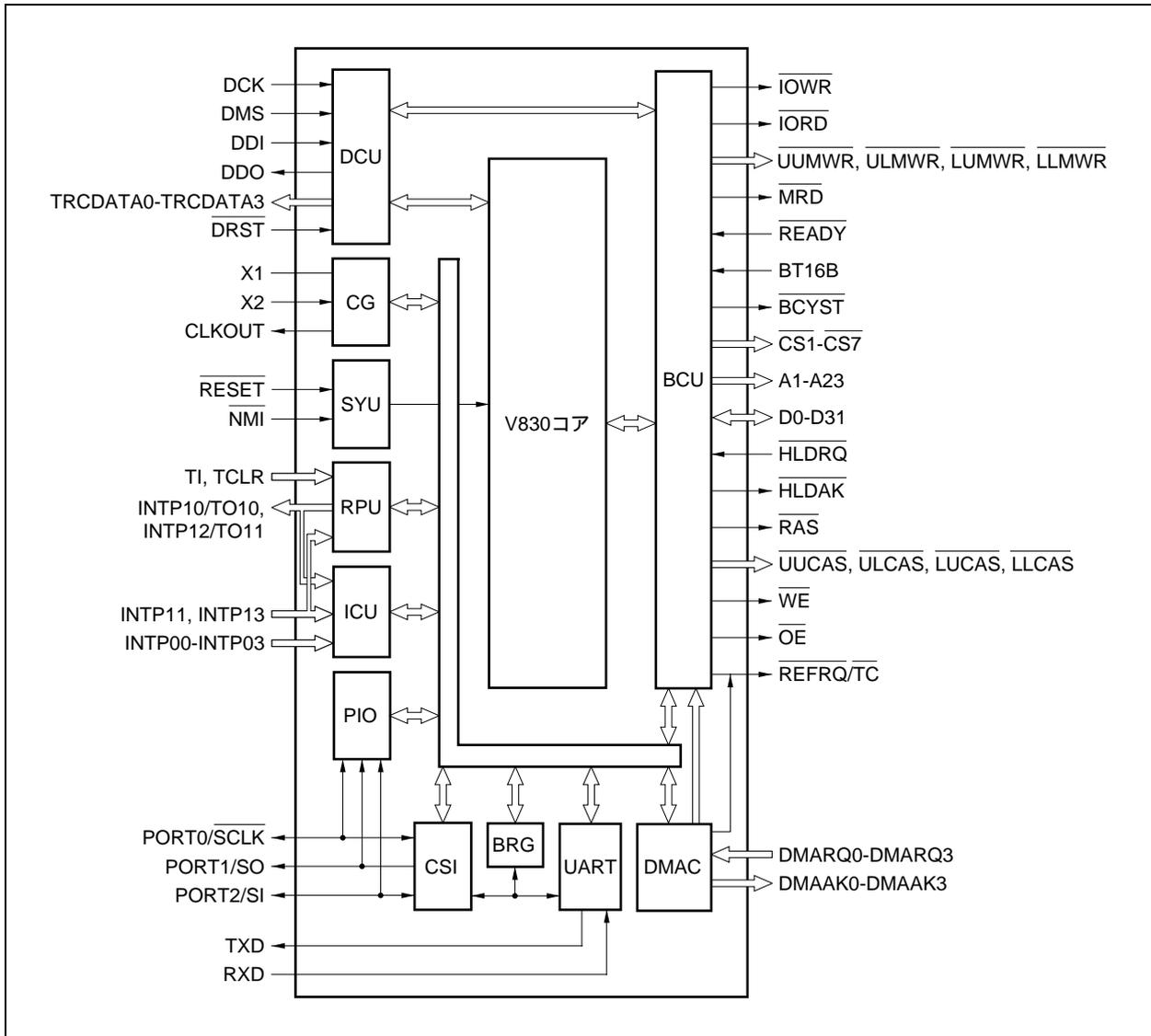


端子名称

A1-A23 : Address Bus	$\overline{\text{NMI}}$: Non-Maskable Interrupt Request
$\overline{\text{BCYST}}$: Bus Cycle Start	$\overline{\text{OE}}$: Output Enable
BT16B : Boot Bus Size 16-bit	PORT0-PORT2 : Port
CLKOUT : Clock Out	$\overline{\text{RAS}}$: Row Address Strobe
$\overline{\text{CS1}}-\overline{\text{CS7}}$: Chip Select	$\overline{\text{READY}}$: Ready
D0-D31 : Data Bus	$\overline{\text{REFRQ}}$: Refresh Request
DCK : Debug Clock	$\overline{\text{RESET}}$: Reset
DDI : Debug Data Input	RXD : Receive Data
DDO : Debug Data Output	$\overline{\text{SCLK}}$: Serial Clock
DMAAK0-DMAAK3 : DMA Acknowledge	SI : Serial Input
DMARQ0-DMARQ3 : DMA Request	SO : Serial Output
DMS : Debug Mode Select	$\overline{\text{TC}}$: Terminal Count
$\overline{\text{DRST}}$: Debug Reset	TCLR : Timer Clear
GND : Ground	TI : Timer Input
GND_PLL : PLL Ground	TO10,TO11 : Timer Output
$\overline{\text{HLD\AA K}}$: Hold Acknowledge	TRCDATA0-TRCDATA3 : Trace Data
$\overline{\text{HLDRQ}}$: Hold Request	TXD : transmit Data
INTP00-INTP03,INTP10-INTP13 : Interrupt Request From Peripheral	$\overline{\text{ULCAS}}$: Upper Lower Column Address Strobe
$\overline{\text{IORD}}$: I/O Read	$\overline{\text{ULMWR}}$: Upper Lower Memory Write
$\overline{\text{IOWR}}$: I/O Write	$\overline{\text{UUCAS}}$: Upper Upper Column Address Strobe
$\overline{\text{LLCAS}}$: Lower Lower Column Address Strobe	$\overline{\text{UUMWR}}$: Upper Upper Memory Write
$\overline{\text{LLMWR}}$: Lower Lower Memory Write	V _{DD} : Power Supply
$\overline{\text{LUCAS}}$: Lower Upper Column Address Strobe	V _{DD_PLL} : PLL Power Supply
$\overline{\text{LUMWR}}$: Lower Upper Memory Write	$\overline{\text{WE}}$: Write Enable
$\overline{\text{MRD}}$: Memory Read	X1,X2 : Crystal Oscillator

1.5 内部ブロック構成

V831の内部ブロック構成を次に示します。



1.6 内部ユニット

V831の内部ユニット構成と機能概要を次に示します。

(1) バス・コントロール・ユニット (BCU)

アドレス・バス, データ・バス, コントロール・バスの端子を制御します。次にBCUの機能を示します。

(a) バス・アビトレーション

各バス・マスタ (CPU, DRAMC, DMAC, 外部バス・マスタ) 間で, バス使用権の調停を行います。バス使用権は, 実行中のバス・サイクル終了後およびアイドル・ステートにおいて切り替えることができます。

(b) ウェイト・コントロール

$\overline{\text{RAS}}$ 信号と7本のチップ・セレクト信号 ($\overline{\text{CS1}}\text{-}\overline{\text{CS7}}$) に対応するリニア16Mバイト空間の8個にエリアを管理します。チップ・セレクト信号の生成, ウェイト制御およびバス・サイクルの種類を選択します。

(c) DRAMコントローラ

$\overline{\text{RAS}}$ 信号と4本の $\overline{\text{CAS}}$ 信号の生成とDRAMへのアクセス制御を行います。DRAMのハイパー・ページ・モードに対応します。DRAMへのアクセスには, 通常アクセス (off-page) とハイパー・ページ・アクセス (on-page) の2種類のサイクルがあります。

(d) ROMコントローラ

ページ・アクセス機能付きROMへのアクセスに対応します。直前のバス・サイクルとのアドレス比較を行い, 通常アクセス (off-page) / ページ・アクセス (on-page) のウェイト制御を行います。8バイトから16バイトまでのページ幅に対応します。

(2) 割り込みコントローラ (ICU)

内蔵周辺ハードウェアおよび外部からのマスカブル割り込み要求 (INTP00-INTP03, INTP10-INTP13) を処理します。これらの割り込み要求を, 4つのグループ単位ごとに優先順位を指定でき, エッジまたはレベルによる割り込み要因の多重処理を行います。

(3) DMAコントローラ (DMAC)

CPUの代わりにメモリとI/Oの間でデータ転送を行います。転送タイプは, 2サイクル転送になります。転送モードには, シングル転送とダイヤモンド転送の2種類があります。

(4) シリアル・インタフェース (UART/CSI/BRG)

アシンクロナス・シリアル・インタフェース (UART) とクロック同期式シリアル・インタフェース (CSI) を1チャンネル備えています。シリアル・クロック・ソースは, ボー・レート・ジェネレータ (BRG) 出力とバス・クロックから選択できます。

(5) リアルタイム・パルス・ユニット (RPU)

タイマ/カウンタ機能を実現します。16ビットのタイマ/イベント・カウンタと16ビットのインターバル・タイマを内蔵して、パルス間隔や周波数の計算、プログラマブルなパルスの出力ができます。

(6) クロック・ジェネレータ (CG)

X1, X2端子に接続された発振子の3倍の周波数を、CPUの動作クロックとして供給します。また、バス・クロック (入力クロック同周期) を周辺ユニットの動作クロックとしても供給します。発振子を接続する代わりに外部クロックを入力することもできます。

(7) ポート (PIO)

ポート機能を実現しています。3本の入出力ポートを備えます。ポート端子とシリアル・コントロール端子の機能を選択して使用できます。

(8) システム・コントロール・ユニット (SYU)

$\overline{\text{RESET}}$ 信号 (入力) / $\overline{\text{NMI}}$ 信号 (入力) のノイズを除去する回路を内蔵しています。

(9) デバッグ・コントロール・ユニット (DCU)

基本的なデバッグ機能を実現するため、マッピングとトレースができる回路を内蔵しています。

(メ モ)

第2章 端子機能

2.1 端子機能一覧

(1/2)

端子名称	入出力	機能	兼用端子	
D0-D31	3ステート入出力	データ・バス		
A1-A23	3ステート出力	アドレス・バス		
$\overline{\text{UUCAS}}$		カラム・アドレス・ストロープ(最上位バイト)		
$\overline{\text{ULCAS}}$		カラム・アドレス・ストロープ(第2バイト)		
$\overline{\text{LUCAS}}$		カラム・アドレス・ストロープ(第3バイト)		
$\overline{\text{LLCAS}}$		カラム・アドレス・ストロープ(最下位バイト)		
$\overline{\text{RAS}}$		ロウ・アドレス・ストロープ/チップ・セレクト		
$\overline{\text{UUMWR}}$		メモリ・ライト・ストロープ(最上位バイト)		
$\overline{\text{ULMWR}}$		メモリ・ライト・ストロープ(第2バイト)		
$\overline{\text{LUMWR}}$		メモリ・ライト・ストロープ(第3バイト)		
$\overline{\text{LLMWR}}$		メモリ・ライト・ストロープ(最下位バイト)		
$\overline{\text{MRD}}$		メモリ・リード・ストロープ		
$\overline{\text{WE}}$		DRAM用ライト・ストロープ		
$\overline{\text{OE}}$		DRAM用リード・ストロープ		
$\overline{\text{IORD}}$		I/Oリード・ストロープ		
$\overline{\text{IOWR}}$		I/Oライト・ストロープ		
$\overline{\text{REFRQ}}$		DRAMリフレッシュ要求	TC	
$\overline{\text{CS1, CS2, CS7}}$		メモリ・チップ・セレクト		
$\overline{\text{CS3-CS6}}$		メモリ・チップ・セレクト/I/Oチップ・セレクト		
$\overline{\text{BCYST}}$		バス・サイクル・スタート		
BT16B		入力	ブート時のバス・サイズ指定	
$\overline{\text{READY}}$			バス・サイクル終結許可	
$\overline{\text{DMARQ0-DMARQ3}}$			DMA要求(CH0-CH3)	
$\overline{\text{DMAAK0-DMAAK3}}$		出力	DMA許可(CH0-CH3)	
TC	DMA転送終了		$\overline{\text{REFRQ}}$	
RXD	入力	UARTデータ入力		
TXD	出力	UARTデータ出力		
SI	入力	CSIデータ入力	PORT2	
SO	出力	CSIデータ出力	PORT1	
SCLK	入出力	CSIクロック入出力	PORT0	
TI	入力	タイマ1カウント・クロック入力		
TCLR		タイマ1クリア, スタート		
TO10	出力	RPUパルス出力	INTP10	
TO11			INTP12	

(2/2)

端子名称	入出力	機能	兼用端子
INTP10	入力	割り込み要求	TO10
INTP11			
INTP12			TO11
INTP13			
INTP00-INTP03			
HLDQR			バス使用要求
HLDAR	出力	バス使用許可	
NMI	入力	ノンマスクブル割り込み要求	
RESET		システム・リセット	
PORT0	入出力	ポート	SCLK
PORT1			SO
PORT2			SI
X1		水晶振動子接続（外部クロック入力時オープン）	
X2	入力	水晶振動子接続 / 外部クロック入力	
CLKOUT	出力	バス・クロック出力	
DCK	入力	ディバグ・クロック入力	
DDI		ディバグ・データ入力	
DDO	3ステート出力	ディバグ・データ出力	
DMS	入力	ディバグ・モード・セレクト	
DRST		リセット入力（ディバグ・モジュール）	
TRCDATA0- TRCDATA3	出力	トレース・データ出力	
V _{DD}		正電源供給	
GND		グランド電位	
V _{DD_PLL}		PLL（内部クロック発生器）用正電源供給	
GND_PLL		PLL（内部クロック発生器）用グランド電位	

2.2 端子状態

動作状態における各端子の端子状態を表2 - 1に示します。

表2 - 1 各端子の状態

端子 \ 動作状態	リセット	バス・ホールド	HALTモード	STOPモード
CLKOUT	クロック出力	クロック出力	クロック出力	0
CS1-CS7	1	Hi-Z	注1	1
A1-A23	不 定	Hi-Z	注1	不 定
D0-D31	Hi-Z	Hi-Z	注1	Hi-Z
BCYST	1	Hi-Z	注1	1
MRD	1	Hi-Z	注1	1
OE	1	Hi-Z	注1	1
WE	1	Hi-Z	注1	1
LLMWR, LUMWR, ULMWR, UUMWR	1	Hi-Z	注1	1
IORD	1	Hi-Z	注1	1
IOWR	1	Hi-Z	注1	1
HLDK	1	0	注1	1
RAS	1	Hi-Z	注1	0 ^{注2}
LLCAS, LUCAS, ULCAS, UUCS	1	Hi-Z	注1	0 ^{注3}
TC/REFRQ	1	Hi-Z	注1	0 ^{注4}

注1 . HALTモード中はDMA動作ができません。

2 . CBRセルフ・リフレッシュが禁止の場合は、STOPモードの前の状態を保持します。

3 . CBRセルフ・リフレッシュが禁止の場合は1になります。

4 . CBRセルフ・リフレッシュが禁止の場合は1になります。ただし、TCを選択している場合はTC信号を出力します。

備考 0 : ロウ・レベル出力

1 : ハイ・レベル出力

Hi-Z : ハイ・インピーダンス

2.3 端子機能説明

2.3.1 アドレス・バス

(1) A1-A23 (Address Bus) ...3ステート出力

V831が、外部の主記憶や入出力装置をアクセスする際のアドレス信号を出力します。2²⁴バイトのアドレス空間をアクセスできます。バス・クロックの立ち上がりに同期して変化します。

2.3.2 データ・バス

(1) D0-D31 (Data Bus) ...3ステート入出力

V831が、外部の主記憶や入出力装置をアクセスする際の書き込みデータおよび読み出しデータの入出力を行う信号です。バス・クロックの立ち上がりに同期して変化します。

2.3.3 バス制御信号

(1) $\overline{\text{READY}}$ (Ready) ...入力

メモリやI/Oのアクセス時間に合うようにバス・サイクルを延長させるための信号です。リード信号/ライト信号直後のバス・クロックの立ち上がりでサンプリングされます。なお、 $\overline{\text{READY}}$ 入力のセットアップ/ホールド時間は必ず守ってください。守らない場合の動作については保証できません。

(2) $\overline{\text{HLDRQ}}$ (Hold Request) ...入力

CPUに対してバス使用权を要求する入力端子です。バス・クロックの立ち上がりでサンプリングされません。

(3) $\overline{\text{HLDACK}}$ (Hold Acknowledge) ...出力

$\overline{\text{HLDRQ}}$ 入力に対するアクノリッジ信号です。

CPUが $\overline{\text{HLDRQ}}$ 信号を受け付けると $\overline{\text{HLDACK}}$ 信号をアクティブにします。 $\overline{\text{HLDRQ}}$ 入力がインアクティブになると、CPUは $\overline{\text{HLDACK}}$ 信号をインアクティブにして、バスの制御権を取り戻します。

(4) $\overline{\text{MRD}}$ (Memory Read) ...3ステート出力

実行中のバス・サイクルが外部メモリに対するリード・サイクルであることを示すストロープ信号です。バス・クロックの立ち下がりに同期して変化します。

ただし、Page-ROMサイクルが連続している間は、常にアクティブです。また、リフレッシュ・サイクル中は常にインアクティブです。

(5) $\overline{\text{LLMWR}}$ (Lower Lower Memory Write) ...3ステート出力

外部メモリに対するライト・データのストロープ信号です。データ・バスは、最下位バイトが有効になります。バス・クロックの立ち下がりに同期して変化します。

(6) $\overline{\text{LUMWR}}$ (Lower Upper Memory Write) ...3ステート出力

外部メモリに対するライト・データのストロープ信号です。データ・バスは、第3バイトが有効になります。バス・クロックの立ち下がりに同期して変化します。

(7) $\overline{\text{ULMWR}}$ (Upper Lower Memory Write) ...3ステート出力

外部メモリに対するライト・データのストロープ信号です。データ・バスは、第2バイトが有効になります。バス・クロックの立ち下がりに同期して変化します。

(8) $\overline{\text{UUMWR}}$ (Upper Upper Memory Write) ...3ステート出力

外部メモリに対するライト・データのストロープ信号です。データ・バスは、最上位バイトが有効になります。バス・クロックの立ち下がりに同期して変化します。

(9) $\overline{\text{IORD}}$ (I/O Read) ...3ステート出力

実行中のバス・サイクルが外部I/Oに対するリード・サイクルであることを示すストロープ信号です。バス・クロックの立ち下がりに同期して変化します。

(10) $\overline{\text{IOWR}}$ (I/O Write) ...3ステート出力

外部のI/Oに対するライト・データのストロープ信号です。バス・クロックの立ち下がりに同期して変化します。

(11) BT16B (Boot Bus Size 16-bit) ...入力

CPU初期化時に $\overline{\text{CS7}}$ で指定されるエリアの外部データ・バス幅を16ビットに固定する信号です。BT16B信号をアクティブにすると16ビット・データ・バス・システムに対応したモードになります。 $\overline{\text{RESET}}$ 信号がハイ・レベルになった次のバス・クロックの立ち上がりでサンプリングします。

BT16Bは、リセット時だけ変更できます。リセット時以外での変更はCPU動作を保証しません。

(12) $\overline{\text{BCYST}}$ (Bus Cycle Start) ...出力

バス・サイクルの最初の1サイクルを示す信号です。ただし、DRAMアクセス時には、 $\overline{\text{CAS}}$ サイクルのタイミングで発生します。この信号はバス・クロックの立ち上がりに同期して変化します。

2.3.4 システム制御信号

(1) $\overline{\text{RESET}}$ (Reset) ...入力

V831を初期化します。アクティブ期間は、必ず25クロック分以上保持してください。電源投入時に $\overline{\text{RESET}}$ 端子に入力するロウ・レベル幅は、これに発振子の発振安定時間を合わせた時間以上を確保してください。発振安定時間は、使用する発振子の規格に従った安定時間を確保してください。

PLLの安定時間は、10 ms以上を確保してください。 $\overline{\text{RESET}}$ 信号が受け付けられ、 $\overline{\text{RESET}}$ 信号がインアクティブになると、V831は各信号と内部レジスタを初期化し、アドレスのFFFFFFF0Hから命令の実行を開始します。

(2) X1, X2 (Crystal Oscillator) ...入力

内部クロック・ジェネレータを使用する場合に、水晶振動子(クリスタル)を接続します。外部クロックを使用する場合は、X2端子に外部クロックを入力して、X1端子をオープンにしてください。

(3) CLKOUT (Clock Out) ...出力

内部で生成したバス・クロックを出力します。

(4) $\overline{\text{CS1}}$, $\overline{\text{CS2}}$, $\overline{\text{CS7}}$ (Chip Select) ...3ステート出力

メモリ・アドレス空間に対して、チップ・セレクト信号を出力します。信号を出力するアドレス・ブロックは、チップ・セレクト信号ごとに固定されています。バス・クロックの立ち上がりに同期して変化します。

(5) $\overline{\text{CS3}}$ - $\overline{\text{CS6}}$ (Chip Select) ...3ステート出力

メモリ・アドレス空間に対するチップ・セレクト信号か、I/Oアドレス空間に対するチップ・セレクト信号のいずれかを出力します。どちらを出力するかは、レジスタ設定により決定されます。信号を出力するアドレス・ブロックは、チップ・セレクト信号ごとに固定されています。バス・クロックの立ち上がりに同期して変化します。

2.3.5 割り込み制御信号

(1) INT10-INT13 (Interrupt Request From Peripheral) ...入力

割り込み制御ユニット(ICU)に対する非同期割り込み要求信号です。エッジトリガまたはレベルトリガ(ハイ・レベル)を選択できます。ただし、RPUで使用する場合はレベルトリガは選択できません。

(2) INT00-INT03 (Interrupt Request From Peripheral) ...入力

割り込み制御ユニット(ICU)に対する非同期割り込み要求信号です。エッジトリガまたはレベルトリガ(ハイ・レベル)を選択できます。

(3) $\overline{\text{NMI}}$ (Non-maskable Interrupt Request) ...入力

CPUに対するマスクできない割り込み要求信号です。クロックの立ち上がりでサンプリングして、5クロック以下のノイズを除去します。

$\overline{\text{NMI}}$ 要求の受け付けは、ノイズを除去したあとの $\overline{\text{NMI}}$ 信号の立ち下がりエッジにより受け付けられます。

2.3.6 DRAM制御信号

(1) $\overline{\text{REFRQ}}$ (Refresh Request) ...3ステート出力

DRAMに対するリフレッシュ要求信号です。バス・クロックの立ち上がりに同期して変化します。外部回路でアドレスをデコードして、接続するDRAMを増やす場合に、リフレッシュ・サイクル時の $\overline{\text{RAS}}$ 制御に使用します。

(2) $\overline{\text{OE}}$ (Output Enable) ...3ステート出力

DRAMに対するリード・イネーブル信号です。バス・クロックの立ち上がりに同期して変化します。

(3) $\overline{\text{RAS}}$ (Row Address Strobe) ...3ステート出力

DRAMに対するロウ・アドレスのストロブ信号です。リフレッシュ・サイクル時は、タイミングが異なります。バス・クロックの立ち上がりに同期して変化します。

(4) $\overline{\text{LLCAS}}$ (Lower Lower Column Address Strobe) ...3ステート出力

DRAMに対するカラム・アドレスのストロブ信号です。データ・バスは最下位バイトが有効になります。リフレッシュ・サイクル時は、タイミングが異なります。バス・クロックの立ち下がりが立ち上がりに同期して変化します。

(5) $\overline{\text{LUCAS}}$ (Lower Upper Column Address Strobe) ...3ステート出力

DRAMに対するカラム・アドレスのストロブ信号です。データ・バスは第3バイトが有効になります。リフレッシュ・サイクル時は、タイミングが異なります。バス・クロックの立ち下がりが立ち上がりに同期して変化します。

(6) $\overline{\text{ULCAS}}$ (Upper Lower Column Address Strobe) ...3ステート出力

DRAMに対するカラム・アドレスのストロブ信号です。データ・バスは第2バイトが有効になります。リフレッシュ・サイクル時は、タイミングが異なります。バス・クロックの立ち下がりが立ち上がりに同期して変化します。

(7) $\overline{\text{UUCAS}}$ (Upper Upper Column Address Strobe) ...3ステート出力

DRAMに対するカラム・アドレスのストロブ信号です。データ・バスは最上位バイトが有効になります。リフレッシュ・サイクル時は、タイミングが異なります。バス・クロックの立ち下がりが立ち上がりに同期して変化します。

(8) $\overline{\text{WE}}$ (Write Enable) ...3ステート出力

実行中のバス・サイクルが、DRAMに対するライト・サイクルであることを示す信号です。バス・クロックの立ち上がりに同期して変化します。

2.3.7 DMA制御信号

(1) DMARQ0-DMARQ3 (DMA Request) ...入力

DMAサービスの要求信号です。それぞれDMAチャンネル0-3に対応して互いに独立しています。優先順位は固定で、DMARQ0 > DMARQ1 > DMARQ2 > DMARQ3になります。DMARQ0-DMARQ3信号は、バス・クロックの立ち上がりでサンプリングされます。DMA要求が受け付けられるまで、アクティブ・レベルを保持してください。DMARQ0-DMARQ3信号を使用しない場合は、端子をインアクティブ・レベルにしてください (DMACのレジスタでアクティブ・レベルを変更できます)。

(2) DMAAK0-DMAAK3 (DMA Acknowledge) ...出力

DMAサービス要求を許可したことを示す信号です。それぞれDMAチャンネル0-3に対応して互いに独立しています。バス・クロックの立ち上がりでアクティブになり、DMA転送中はアクティブ・レベルを保持します。

(3) \overline{TC} (Terminal Count) ...出力

DMAコントローラによるDMA転送が終了したことを示す信号です。 \overline{TC} 信号は、バス・クロックの立ち上がりでアクティブになります。また、チャンネル0-3の \overline{TC} の論理和をとって出力するため、外部回路でそれぞれDMAAK0-DMAAK3信号と論理積をとることにより、それぞれのチャンネルの \overline{TC} 信号を生成してください。

2.3.8 リアルタイム・パルス制御信号

(1) TO10, TO11 (Timer Output) ...出力

リアルタイム・パルス・ユニット (RPU) のタイマ1 (TM1) のキャプチャ/コンペア・レジスタ (CC0) のコンペア一致信号です。TO10, TO11信号は、それぞれCC10, CC12レジスタの一致検出でセットされ、CC11, CC13レジスタの一致検出でリセットされます。RPUのモード設定によって、出力を反転できます。

(2) TCLR (Timer Clear) ...入力

RPUのTM1のカウント・クリア・スタート信号です。

(3) TI (Timer Input) ...入力

RPUのTM1が使用する外部クロック信号です。RPUのTM1が外部クロック信号を使用するか、内部のバス・クロックを分周したクロックを使用するかは、イニシャライズ時に設定してください。

2.3.9 シリアル制御信号

(1) TXD (Transmit Data) ...出力

UARTのシリアル送信データ出力端子です。TXD信号は、内部シリアル・クロックに同期して変化します。非送信時はハイ・レベルを保持します。

(2) RXD (Receive Data) ...入力

UARTのシリアル受信データ入力端子です。

(3) SCLK (Serial Clock) ...入出力

CSIのシリアル・クロック入出力端子です。レジスタの設定により入出力を設定します。

(4) SO (Serial Output) ...出力

CSIのシリアル送信データ出力端子です。SO信号は、SCLK信号の立ち下がりに同期して変化します。非送信時はハイ・インピーダンス状態になります。

(5) SI (Serial Input) ...入力

CSIのシリアル受信データ入力端子です。SI信号はSCLK信号の立ち上がりでサンプリングされます。

2.3.10 ポート制御信号

(1) PORT0-PORT2 (Port) ...3ステート入出力

ポートの信号です。レジスタで入出力を選択できます。

2.3.11 デバッグ制御信号

(1) DCK (Debug Clock) ...入力

デバッグ・クロック入力端子です。DCK信号の立ち上がりに同期してDMS信号、DDI信号がサンプリングされ、DCK信号の立ち下がりに同期して、DDO端子よりデータが出力されます。デバッグ機能を使用しない場合は、ハイ・レベルにしてください。

(2) DDI (Debug Data Input) ...入力

デバッグ・データ入力端子です。デバッグ・シリアル・インタフェースが、ShiftステートのときにDCK信号の立ち上がりに同期してサンプリングされます。データは、LSBファーストです。デバッグ機能を使用しない場合は、ハイ・レベルにしてください。

(3) DDO (Debug Data Output) ...3ステート出力

デバッグ・データ出力端子です。デバッグ・シリアル・インタフェースが、ShiftステートのときにDCK信号の立ち下がりに同期してデータが出力されます。データは、LSBファーストです。

(4) DMS (Debug Mode Select) ...入力

ディバグ・モード・セレクト入力端子です。DMS信号のレベルにより、ディバグ・シリアル・インタフェースのステート・マシンが変化します。DMS信号は、DCK信号の立ち上がりに同期してサンプリングされます。ディバグ機能を使用しない場合は、ハイ・レベルにしてください。

(5) DRST (Debug Reset) ...入力

ディバグ・リセット入力端子です。DCUを非同期に初期化する負論理の信号です。ロウ・レベルにするとDCUをリセット／無効にします。ディバグ機能を使用しない場合は、ロウ・レベルにしてください。

(6) TRCDATA0-TRCDATA3 (Trace Data) ...出力

トレース・データ出力端子です。CLKOUT信号の立ち上がりに同期して、パケット化されたトレース・データをLSB側から4ビットごとに出力します。

第3章 CPU機能

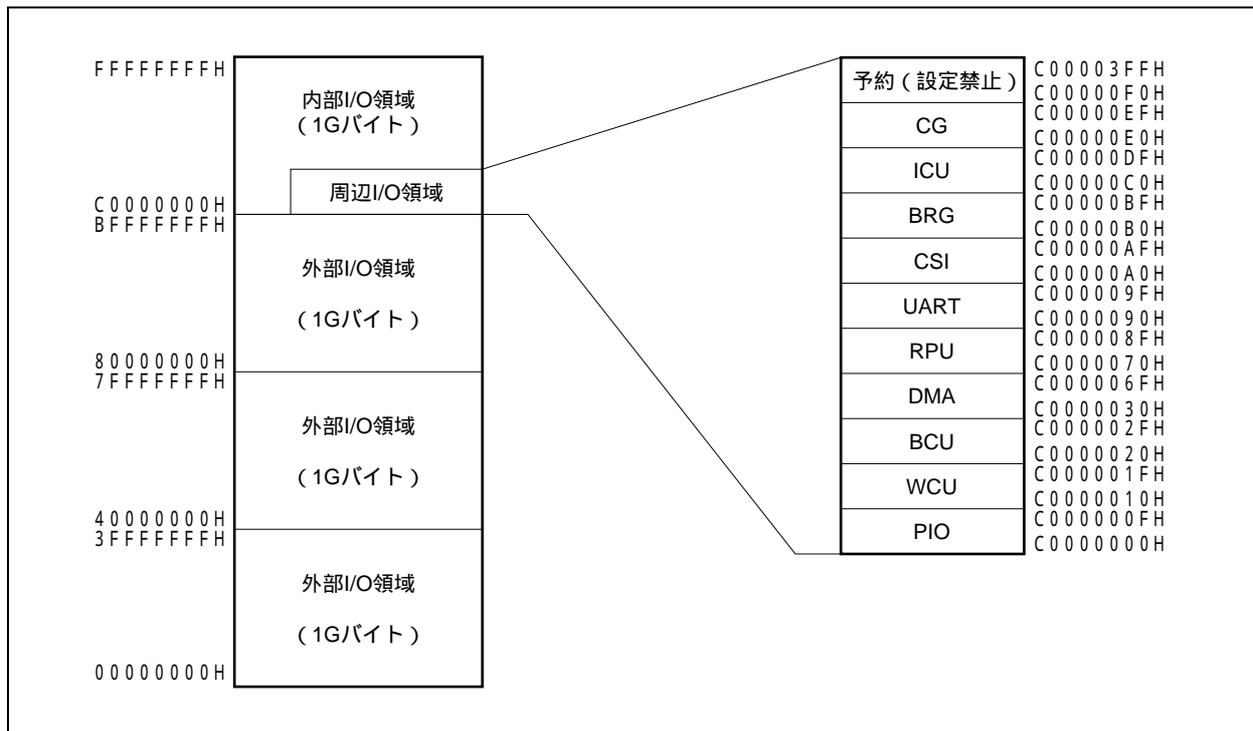
この章では、V831の内蔵周辺I/O空間、CPUコア・システム・レジスタの変更について説明します。命令セット、レジスタ構成については、V830ファミリ ユーザーズ・マニュアル アーキテクチャ編を参照してください。内蔵周辺I/Oレジスタについては、周辺ユニットの機能説明に示している各レジスタを参照してください。

3.1 内蔵周辺I/O空間

内蔵の周辺I/O空間は、上位1Gバイトの内蔵I/O領域のC0000000H-C00003FFFH (1Kバイト) に予約されています。内部I/Oへのアクセスは、IN.H/OUT.H命令 (ハーフワード単位) またはIN.B/OUT.B命令 (バイト単位) を使用してください。なお、内部I/O領域をアクセスした場合は、外部へのバス・サイクルは起動されず、アイドル・ステートとなります。アイドル・ステート中の各信号の状態は5.6 アイドル・ステートを参照してください。

★

図3-1 内蔵周辺I/Oマップ



3.1.1 注意事項

次に内蔵周辺I/O空間の注意事項について示します。

- ・ハードウェア的には、レジスタ・アドレスのデコードに32ビット・アドレスの下位10ビットだけを使用して、1Kバイトの領域をレジスタの割り付けに使用しています。
- ・アドレスの下位ビット（A0）は、デコードしていません。したがって、奇数アドレス（ $2n+1$ 番地）のレジスタにバイト・アクセスした場合は、ハードウェア上、偶数アドレス（ $2n$ 番地）のレジスタにアクセスします。
- ・V831では、ワード・アクセスできるレジスタはありません。内蔵周辺I/Oレジスタにワード・アクセスした場合は、ハードウェア上、強制的にハーフワード・アクセスに置き換わります。
- ・バイト・アクセスができるレジスタにハーフワード・アクセスをした場合は、リード時は上位8ビット・データが不定になり、ライト時は下位8ビット・データがレジスタに書き込まれます。
- ・レジスタを示していないアドレスは将来の拡張用に予約されており、アクセスした場合の動作は不定になり、動作保証はしません。

3.2 CPUコア・システム・レジスタ

プロセッサIDレジスタ（PIR）は、V831用に変更されています。PIRは、CPUタイプ番号を識別するためのレジスタです。PIRの内容は、00008301H固定です（32ビット単位でリードだけできます）。

各レジスタのリセット後の初期値は、表14-2 各レジスタのリセット後の初期値を参照してください。

第4章 割り込み / 例外処理機能

4.1 割り込み / 例外処理

V831は、専用の割り込みコントローラ（ICU）を内蔵して、合計15要因の割り込み要求を処理できます。

割り込みは、プログラム実行とは別に独立して発生する事象です。例外は、プログラム実行に依存して発生する事象です。一般には、割り込みは例外より優先的に処理されます。

V831では、オンチップの周辺機能ユニットおよび外部からの各割り込み要求を処理できます。さらに、命令による例外処理の起動（TRAP命令）や、例外の発生（不正命令コードなど（アドレス・トラップ例外は除きます））による例外処理の起動ができます。例外要因は、ECR（Exception Cause Register）に格納されている例外コードで確認できます。

各割り込み要求は、ソフトウェアにより4レベルの優先順位を指定できます。外部割り込みは、要求発生から最小6バス・クロック後に割り込み処理が起動されます。なお、ノイズ除去する時間（3バス・クロック）を含みます。内部割り込みの場合は、要求発生から最小3バス・クロック後に起動されます。

4.1.1 割り込み / 例外処理の種類

V831の割り込み / 例外処理は、次に示す4種類に分類されます。

- ・ ノンマスカブル割り込み : 1要因
- ・ マスカブル割り込み : 15要因
- ・ ソフトウェア例外 : 32要因
- ・ 例外トラップ : 4要因

表4-1 割り込み一覧

種 類	分 類	割り込み / 例外要因		例外コード (ECR)	ハンドラ・ アドレス	復帰PC ^{注2}
		名称 ^{注1}	発生要因			
リセット	割り込み	RESET	リセット入力	FFF0H	FFFFFFF0H	不定
ノンマスカブル	割り込み	NMI	NMI入力	FFD0H	FFFFFFD0H	next PC ^{注3}
ソフトウェア例外	例外	TRAP 1nH	TRAP命令	FFBnH	FFFFFFB0H	next PC
		TRAP 0nH	TRAP命令	FFAnH	FFFFFFA0H	
例外トラップ	例外	NMI	二重例外	注4	FFFFFFD0H	current PC
		FAULT	致命的例外	変化しない	FFFFFFE0H	
		I-OPC	不正命令コード	FF90H	FFFFFF90H	
		DIV0	ゼロ除算	FF80H	FFFFFF80H	

注1. 開発ツールまたはソフトウェアでのハンドラの名称です。

2. 割り込み / 例外処理起動時に、EIPC/FEPC/DPCにセーブされるPC値のことです。

3. すべての命令は、割り込みで命令実効を中断できません。

4. 二重例外の要因となった例外の例外コードです。

備考 n = 0H-FH

表4-2 割り込み一覧 (マスクابل割り込み)

種類	分類	グループ	グループ内 優先順位	割り込み要因			例外 コード	ハンドラ・アドレス ^{注3}		復帰PC ^{注1}
				名称	発生要因	発生ユニット		HCCW.IHA = 0	HCCW.IHA = 1	
マスク アブル	割 り 込 み	GR3	3	RESERVED	予約	-	FEF0H	FFFFFFEF0H	FE0000F0H	next PC ^{注2}
			2	INTOV1	タイマ1オーバフロー	RPU	FEE0H	FFFFFFEE0H	FE0000E0H	
			1	INTSER	UART受信エラー	UART	FED0H	FFFFFFED0H	FE0000D0H	
			0	INTP03	INTP03端子入力	外部	FEC0H	FFFFFFEC0H	FE0000C0H	
		GR2	3	INTSR	UART受信終了	UART	FEB0H	FFFFFFEB0H	FE0000B0H	
			2	INTST	UART送信終了	UART	FEA0H	FFFFFFEA0H	FE0000A0H	
			1	INTCSI	CSI送受信終了	CSI	FE90H	FFFFFFE90H	FE000090H	
			0	INT02	INTP02端子入力	外部	FE80H	FFFFFFE80H	FE000080H	
		GR1	3	INTDMA	DMA転送終了	DMAC	FE70H	FFFFFFE70H	FE000070H	
			2	INTP10/ INTCC10	INTP10端子入力 / CC10の一致	外部 / RPU	FE60H	FFFFFFE60H	FE000060H	
			1	INTP11/ INTCC11	INTP11端子入力 / CC11の一致	外部 / RPU	FE50H	FFFFFFE50H	FE000050H	
			0	INTP01	INTP01端子入力	外部	FE40H	FFFFFFE40H	FE000040H	
		GR0	3	INTCM4	CM4の一致	RPU	FE30H	FFFFFFE30H	FE000030H	
			2	INTP12/ INTCC12	INTP12端子入力 / CC12の一致	外部 / RPU	FE20H	FFFFFFE20H	FE000020H	
			1	INTP13/ INTCC13	INTP13端子入力 / CC13の一致	外部 / RPU	FE10H	FFFFFFE10H	FE000010H	
			0	INTP00	INTP00の端子入力	外部	FE00H	FFFFFFE00H	FE000000H	

注1. 割り込み時に、EIPCにセーブされるPC値のことです。

2. すべての命令は、割り込みで命令実効を中断できません。

3. ハンドラ・アドレスは、HCCW.IHA = 0のときFFFFFFEn0H, HCCW.IHA = 1のときFE0000n0Hを選択できます (n = 0H-FH)。

注意 マスクアブル割り込みの例外コードとハンドラ・アドレスは、デフォルト優先順位での値です。優先順位を変更した場合は、表4-3 優先順位と例外コード, ハンドラ・アドレス, 割り込み優先の関係を参照してください。

4.2 ノンマスクابل割り込み

ノンマスクابل割り込み要求は、割り込み禁止状態であっても無条件に受け付けられます。また、割り込み優先順位の対象にならず、すべての割り込みに対して最優先の割り込み要求です。

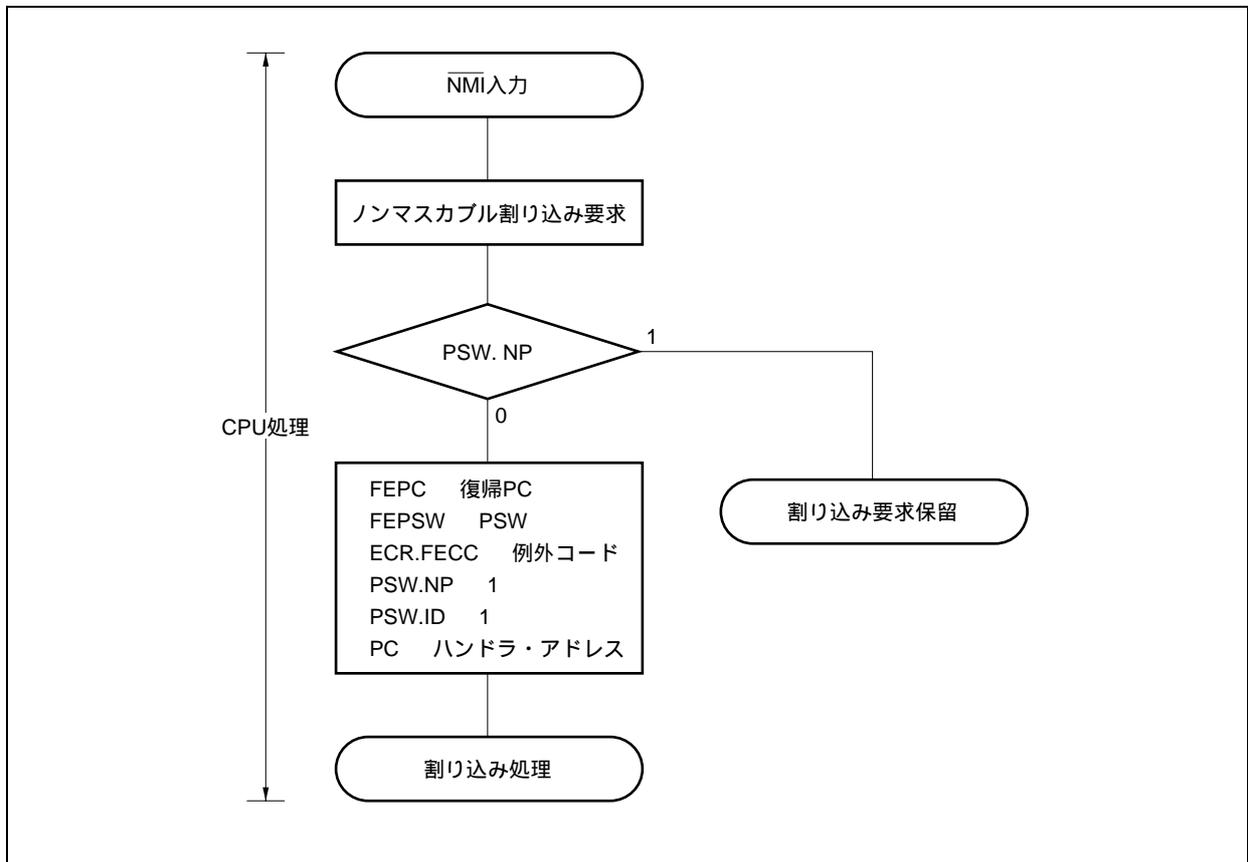
ノンマスクابل割り込み要求は、 $\overline{\text{NMI}}$ 端子によって行います。

4.2.1 ノンマスクابل割り込みの処理形態

$\overline{\text{NMI}}$ 信号入力によりノンマスクابل割り込みが発生した場合は、図4-1に示す処理を行い、ハンドラ・ルーチンに制御を移します。状態退避レジスタには、FEPC、FEPSWを使用します。

ノンマスクابل割り込みの処理中（PSW.NP = 1）に発生した $\overline{\text{NMI}}$ 信号入力は、CPU内部で保留されます。この場合、RETI命令、LDSR命令を使用してPSW.NP = 0にすると、保留していたノンマスクابل割り込み要求により新たなノンマスクابل割り込み処理を開始します。

図4-1 ノンマスクابل割り込み要求の処理フロー



4.3 マスカブル割り込み

マスカブル割り込み要求は、制御レジスタにより割り込み受け付けをマスクできる割り込み要求です。V832には、15種類のマスカブル割り込み要因があります。

複数のマスカブル割り込みが同時に発生した場合は、ICUの制御により、あらかじめ設定されているデフォルト優先順位によりその受け付け優先順位が決定します。また、デフォルト優先順位とは別に、ソフトウェアにより4レベルの優先順位を設定できます（グループ内の優先順位は固定です）。

マスカブル割り込みは、PSWのNP, EP, IDビットの論理和でマスクされます。さらにCPUに渡す割り込みレベル n がPSWの割り込み許可レベル（PSWのI0-I3ビット）より低い場合は割り込みは受け付けられません。したがって、最高位の割り込みレベル（ $n = 15$ ）を割り込み許可レベルで禁止することはできません。

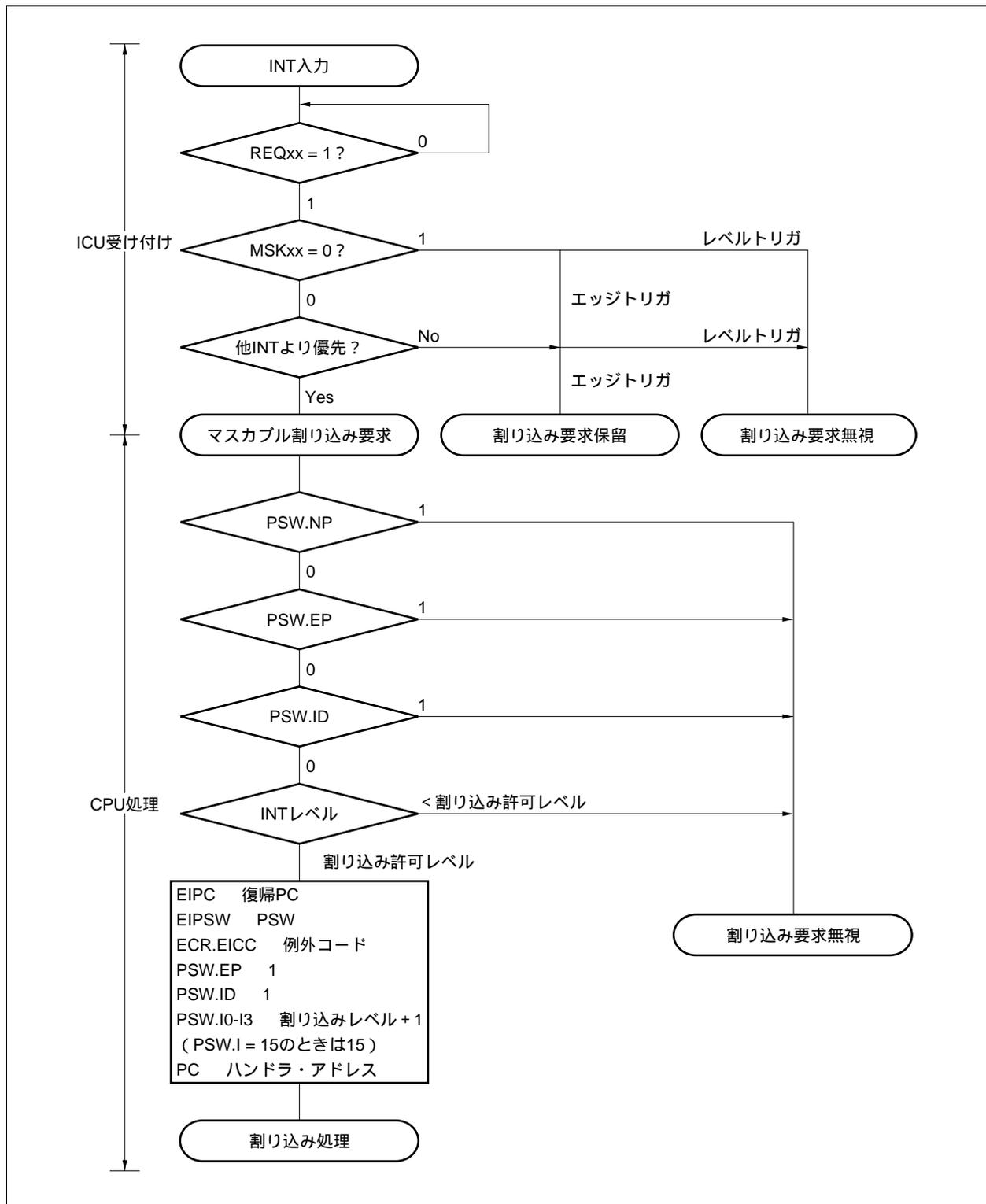
割り込み要求が受け付けられると割り込み禁止状態（PSW.ID = 1）になり、以後のマスカブル割り込み要求の受け付けを禁止します。また、PSWのI0-I3ビットに受け付けた割り込みレベル n に1を加算した値（ $n + 1$ ）を設定します。

多重割り込みを許可するときは、まずEIPC, EIPSWをメモリまたはレジスタに退避してから割り込み許可状態（PSW.ID = 0, EP = 0）にしてください。またRETI命令は、割り込み禁止状態（PSW.ID = 1）にしてから実行し、EIPC, EIPSWを元の値に復帰させます。

4.3.1 マスカブル割り込みの処理形態

割り込み要求信号（INT）入力によりマスカブル割り込みが発生した場合は、図4-2に示す処理を行い、ハンドラ・ルーチンに制御を移します。状態待避レジスタにはEIPC, EIPSWを使用します。なお、ICUにおいてマスクされているINT入力、ほかの割り込み処理中（PSW.NP = 1またはPSW.ID = 1）に発生したINT入力は、ICU内部で保留されます。この場合、マスクを解除し、またはRETI命令、LDSR命令を使用してPSW.NP = 0, PSW.ID = 0にすると、保留していたINT入力により新たなマスカブル割り込み処理を開始します。

図4-2 マスカブル割り込み要求の処理フロー



4.3.2 マスカブル割り込みの優先順位

V831は、割り込み処理中にさらに別の割り込みを受け付ける多重割り込みの処理を行います。多重割り込みは、優先順位によって制御できます。

優先順位制御には、デフォルト優先順位による制御と、割り込みグループ優先順位レジスタ設定によるプログラム優先順位制御があります。

デフォルト優先順位による制御は、複数の割り込みが同時に発生している場合、割り込み要求の各グループ（4つの割り込み要求で1グループ）にあらかじめ割り付けてある優先順位（デフォルト優先順位）に従って割り込み処理を行います（表4-2 割り込み一覧（マスカブル割り込み）参照）。

プログラマブル優先順位制御は、各割り込み要求を割り込みグループ優先順位制御レジスタによってグループごとに4レベルに分けます。ただし、グループ内の各割り込み要求の優先順位は固定です。

なお、割り込み要求を受け付けるとPSWのIDフラグ、EPフラグが自動的にセットされるので、多重割り込みを使用する場合は、割り込み処理プログラム中で割り込み許可状態（PSW.ID = 0, PSW.EP = 0）にしてください。

（マスカブル割り込み要求または例外の処理プログラム）

EIPCをメモリまたはレジスタへ退避

EIPSWをメモリまたはレジスタへ退避

割り込み受け付け許可（PSW.ID = 0, PSW.EP = 0）

...

...

...

← INT入力などの割り込み受け付け

割り込み受け付け禁止（PSW.ID = 1, PSW.EP = 1）

退避していた値をEIPSWに復帰

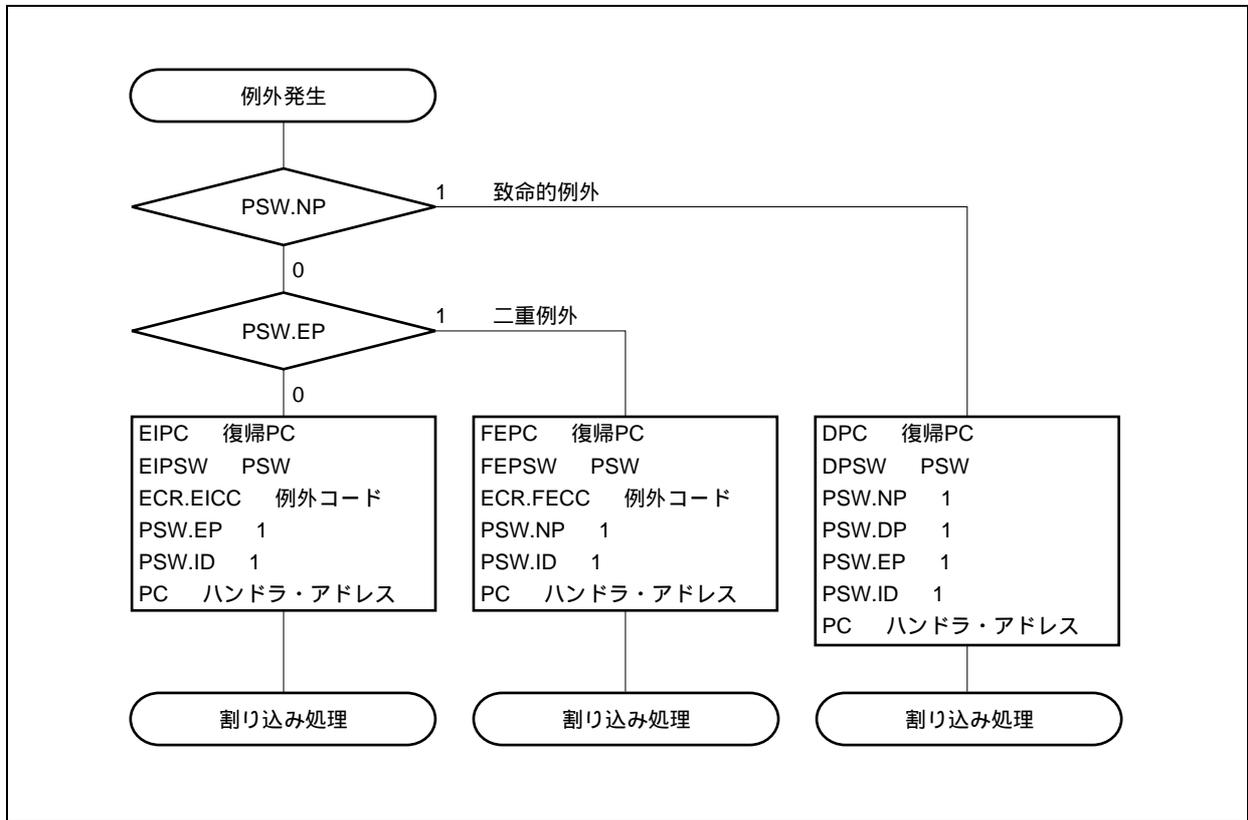
退避していた値をEIPCに復帰

RETI命令の実行

4.4 例外処理

例外が発生した場合、次の処理を行いハンドラ・ルーチンへ制御を移します。

図4-3 例外処理フロー

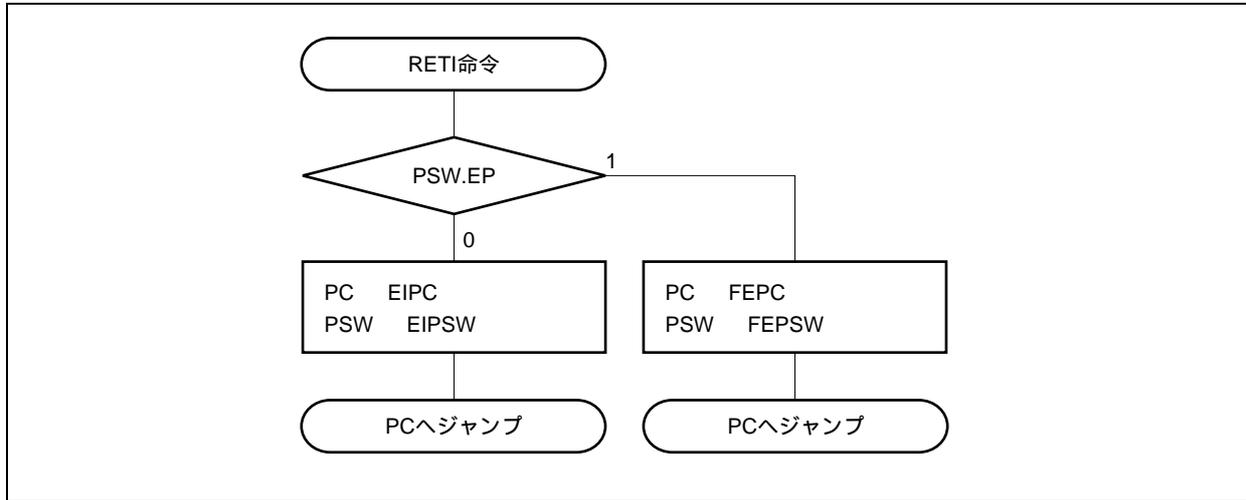


4.5 例外 / 割り込みからの復帰

4.5.1 例外 / 割り込みからの復帰

致命的例外以外の例外事象および割り込み事象からの復帰は、すべてRETI命令により行います。

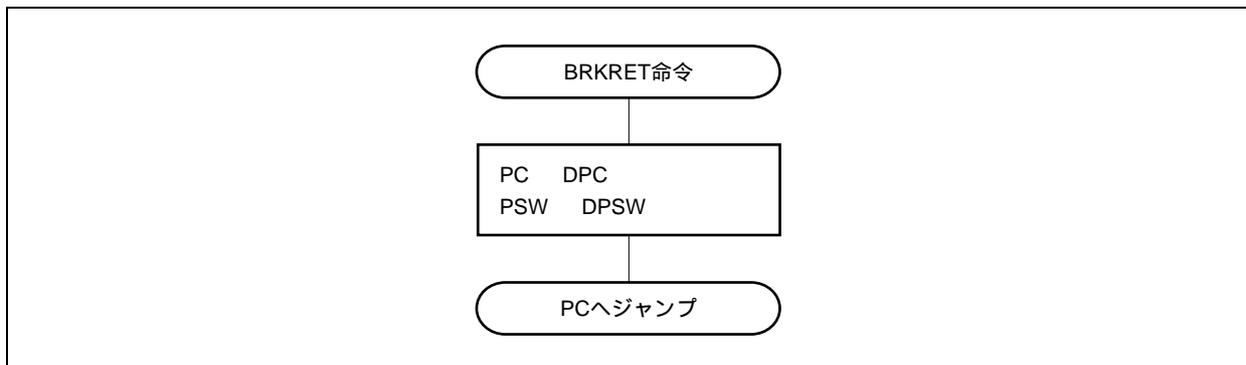
図4 - 4 例外 / 割り込みからの復帰フロー



4.5.2 致命的例外ルーチンからの復帰

致命的例外処理からの復帰は、BRKRET命令により行います。

図4 - 5 致命的例外ルーチンからの復帰フロー



4.6 割り込み制御レジスタ

割り込みは、レベルトリガとエッジトリガの選択ができます。

V831内部ユニット内からの割り込みは、すべてエッジトリガになります。レベルトリガとエッジトリガの設定方法は、4.6.5 ICUモード・レジスタ (IMOD) を参照してください。

4.6.1 割り込みグループ優先順位レジスタ (IGP)

割り込みグループ優先順位レジスタ (IGP) は、グループ間の優先順位を設定します。優先順位は割り込みマスク状態で設定 / 変更してください。ハンドラ・アドレスの関係を表4-3に示します。

図4-6 割り込みグループ優先順位レジスタ (IGP)

アドレス : C00000C0H												
	7	6	5	4	3	2	1	0				
IGP	PR3	PR2	PR1	PR0								
初期値 :	1	1	1	0	0	1	0	0				
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W				
ビット	ビット名	説明										
7-0	PR3-PR0	<p>Group Priority</p> <p>4分割される割り込みグループn (PRn) の優先順位を指定します (n = 0-3)。 優先順位は0が最低で、3が最高です。</p> <table border="1"> <thead> <tr> <th>PRn</th> <th>優先順位</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td>0 (最低優先)</td> </tr> <tr> <td>0 1</td> <td>1</td> </tr> <tr> <td>1 0</td> <td>2</td> </tr> <tr> <td>1 1</td> <td>3 (最高優先)</td> </tr> </tbody> </table>	PRn	優先順位	0 0	0 (最低優先)	0 1	1	1 0	2	1 1	3 (最高優先)
PRn	優先順位											
0 0	0 (最低優先)											
0 1	1											
1 0	2											
1 1	3 (最高優先)											
<p>注意 グループ間の優先順位を同一に設定しないでください。設定した場合の動作は保証しません。</p>												

4.6.3 割り込み要求レジスタ (IRR)

割り込み要求を保持します。

図4-8 割り込み要求レジスタ (IRR)

アドレス : C00000C4H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRR	0	REQ14	REQ13	REQ12	REQ11	REQ10	REQ9	REQ8	REQ7	REQ6	REQ5	REQ4	REQ3	REQ2	REQ1	REQ0

初期値 : 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

R/W : R R R R R R R R R R R R R R R R

ビット	ビット名	説明
14-0	REQ14-REQ0	Interrupt Request 割り込み要求の発生でセットされます。マスク・レジスタの影響は受けません。エッジ割り込みの場合、ICRレジスタの操作によってリセットされます。読み出しだけです。 IRRレジスタへの書き込みはできません。 0 : 割り込み要求はありません。 1 : 割り込み要求が発生しています。

4.6.4 割り込み要求マスク・レジスタ (IMR)

割り込み要求をマスクします。

図4-9 割り込み要求マスク・レジスタ (IMR)

アドレス : C00000C6H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR	1	MSK14	MSK13	MSK12	MSK11	MSK10	MSK9	MSK8	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0

初期値 : 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

R/W : R R/W R/W

ビット	ビット名	説明
14-0	MSK14-MSK0	Mask Interrupt Request 対応する割り込み要求をマスクします。ただし、マスクしたい割り込み要求が発生していないときにマスクしてください。 0 : マスクしません。 1 : マスクします。

表4 - 4 割り込み制御レジスタの各ビットと割り込み要求信号の対応

ICRレジスタのビット	IRRレジスタのビット	IMRレジスタのビット	割り込み要求信号
CLR14	REQ14	MSK14	INTOV1
CLR13	REQ13	MSK13	INTSER
CLR12	REQ12	MSK12	INTP03
CLR11	REQ11	MSK11	INTSR
CLR10	REQ10	MSK10	INTST
CLR9	REQ9	MSK9	INTCSI
CLR8	REQ8	MSK8	INTP02
CLR7	REQ7	MSK7	INTDMA
CLR6	REQ6	MSK6	INTP10/INTCC10
CLR5	REQ5	MSK5	INTP11/INTCC11
CLR4	REQ4	MSK4	INTP01
CLR3	REQ3	MSK3	INTCM4
CLR2	REQ2	MSK2	INTP12/INTCC12
CLR1	REQ1	MSK1	INTP13/INTCC13
CLR0	REQ0	MSK0	INTP00

注意 グループの優先順位には依存しません。固定の割り込みに対応します。

4.6.5 ICUモード・レジスタ (IMOD)

外部端子による外部割り込み要求 (INTP00-INTP03, INTP10-INTP13) のトリガ・モードを設定します。トリガ・モードには、レベルトリガとエッジトリガの2種類があります。

(1) レベルトリガ

クロックごとにサンプリングします。割り込み要求が発生した場合、CPUの割り込みハンドラ内で割り込み要求を認識するまでアクティブ・レベル (ハイ・レベル) を保持してください。認識される前に割り込み要求を取り下げると、分岐先ベクタが不定になるなどの誤動作が発生します。また、割り込みクリア・レジスタ (ICR) によって、割り込み要求レジスタ (IRR) をクリアできません。

(2) エッジトリガ

クロックの立ち上がりでサンプリングします。クロックの立ち上がりで、割り込み要求信号がIMODレジスタで指定したエッジで変化した場合、割り込み要求が受け付けられます。割り込み要求は、連続して入力しても1回しかカウントされません。内部割り込み要求もエッジ入力のため、割り込み処理ルーチンにおいて割り込み要求レジスタ (IRR) の対応するビットをクリアしてください。

注意 エッジトリガ時は、内部のエッジ検出回路の制約により、3バス・クロック分以上の割り込み要求レベルを保持をしてください。

図4 - 10 ICUモード・レジスタ (IMOD)

アドレス : C00000C8H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMOD	ITM13		ITM12		ITM11		ITM10		ITM03		ITM02		ITM01		ITM00	

初期値 :

1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W : R/W R/W R/W R/W R/W R/W R/W R/W

ビット	ビット名	説 明																														
15-8	ITM13-ITM10	<p>Interrupt Trigger Mode1</p> <p>INTP1n端子のトリガ・モードを設定します。ただし、トリガ・モードの設定を変更する場合は、設定したい割り込み要求が発生していないときに変更してください (n = 0-3)。</p> <p>また、INTP1n端子は、RPUのタイマ1のキャプチャ・モード用の割り込みと兼用になります。したがって、RPUのTUM1レジスタの設定により有効になるトリガ・モードが次のように異なります。</p> <p>(1) TUM1レジスタにCMS1n = 1, IMS1n = 1を設定した場合 (n = 0-3)</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <thead> <tr> <th>ITM1n</th> <th>トリガ・モード</th> </tr> </thead> <tbody> <tr><td>0 0</td><td>レベルトリガ (ハイ・アクティブ)</td></tr> <tr><td>0 1</td><td>予約 (設定禁止)</td></tr> <tr><td>1 0</td><td>立ち上がりエッジトリガ</td></tr> <tr><td>1 1</td><td>予約 (設定禁止)</td></tr> </tbody> </table> <p>(2) TUM1レジスタにCMS1n = 1, IMS1n = 0を設定した場合 (n = 0-3)</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <thead> <tr> <th>ITM1n</th> <th>トリガ・モード</th> </tr> </thead> <tbody> <tr><td>0 0</td><td>予約 (設定禁止)</td></tr> <tr><td>0 1</td><td>予約 (設定禁止)</td></tr> <tr><td>1 0</td><td>立ち上がりエッジトリガ</td></tr> <tr><td>1 1</td><td>予約 (設定禁止)</td></tr> </tbody> </table> <p>(3) TUM1レジスタにCMS1n = 0を設定した場合 (n = 0-3)</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <thead> <tr> <th>ITM1n</th> <th>トリガ・モード</th> </tr> </thead> <tbody> <tr><td>0 0</td><td>予約 (設定禁止)</td></tr> <tr><td>0 1</td><td>予約 (設定禁止)</td></tr> <tr><td>1 0</td><td>立ち上がりエッジトリガ</td></tr> <tr><td>1 1</td><td>立ち上がりと立ち下りの両エッジトリガ</td></tr> </tbody> </table> <p>注意 RPUのタイマ1の動作中は、ITM13-ITM10の設定を変更しないでください。</p>	ITM1n	トリガ・モード	0 0	レベルトリガ (ハイ・アクティブ)	0 1	予約 (設定禁止)	1 0	立ち上がりエッジトリガ	1 1	予約 (設定禁止)	ITM1n	トリガ・モード	0 0	予約 (設定禁止)	0 1	予約 (設定禁止)	1 0	立ち上がりエッジトリガ	1 1	予約 (設定禁止)	ITM1n	トリガ・モード	0 0	予約 (設定禁止)	0 1	予約 (設定禁止)	1 0	立ち上がりエッジトリガ	1 1	立ち上がりと立ち下りの両エッジトリガ
ITM1n	トリガ・モード																															
0 0	レベルトリガ (ハイ・アクティブ)																															
0 1	予約 (設定禁止)																															
1 0	立ち上がりエッジトリガ																															
1 1	予約 (設定禁止)																															
ITM1n	トリガ・モード																															
0 0	予約 (設定禁止)																															
0 1	予約 (設定禁止)																															
1 0	立ち上がりエッジトリガ																															
1 1	予約 (設定禁止)																															
ITM1n	トリガ・モード																															
0 0	予約 (設定禁止)																															
0 1	予約 (設定禁止)																															
1 0	立ち上がりエッジトリガ																															
1 1	立ち上がりと立ち下りの両エッジトリガ																															
7-0	ITM03-ITM00	<p>Interrupt Trigger Mode0</p> <p>INTP0n端子のトリガ・モードを設定します。ただし、トリガ・モードの設定を変更する場合は、設定したい割り込み要求が発生していないときに変更してください (n = 0-3)。</p> <table border="1" style="width: 100%; border-collapse: collapse; margin-bottom: 10px;"> <thead> <tr> <th>ITM0n</th> <th>トリガ・モード</th> </tr> </thead> <tbody> <tr><td>0 0</td><td>レベルトリガ (ハイ・アクティブ)</td></tr> <tr><td>0 1</td><td>予約 (設定禁止)</td></tr> <tr><td>1 0</td><td>立ち上がりエッジトリガ</td></tr> <tr><td>1 1</td><td>予約 (設定禁止)</td></tr> </tbody> </table>	ITM0n	トリガ・モード	0 0	レベルトリガ (ハイ・アクティブ)	0 1	予約 (設定禁止)	1 0	立ち上がりエッジトリガ	1 1	予約 (設定禁止)																				
ITM0n	トリガ・モード																															
0 0	レベルトリガ (ハイ・アクティブ)																															
0 1	予約 (設定禁止)																															
1 0	立ち上がりエッジトリガ																															
1 1	予約 (設定禁止)																															

★ 4.7 外部入力端子による割り込み要求

外部入力端子による要求には、INTP0n端子もしくはINTP1n端子を使用します。

外部入力端子による要求には、次に示すレジスタで設定してください。

- ・割り込み要求マスク・レジスタ (IMR) (4.6.4参照)
- ・ICUモード・レジスタ (IMOD) (4.6.5参照)
- ・タイマ・ユニット・モード・レジスタ (TUM1) (10.3.1参照)

(1) INTP0n端子を使用する場合

- ・ICUモード・レジスタ (IMOD) にてトリガ・モードを設定
- ・割り込み要求マスク・レジスタで割り込み要求のマスクを解除

(2) INTP1n端子を外部入力割り込み端子として使用する場合

- ・ICUモード・レジスタ (IMOD) にてトリガ・モードを設定
- ・割り込み要求マスク・レジスタで割り込み要求のマスクを解除
- ・タイマ1の制御レジスタ, タイマ・ユニット・モード・レジスタ (TUM1) のIMSビットを1に設定 (外部からの入力信号を割り込み要求信号にする)

INTP1n入力は、タイマ1のキャプチャ機能と兼用になっています。タイマ1のキャプチャ機能を使用する場合は、使用するキャプチャ・レジスタに対応するINTP1n端子入力はキャプチャ・トリガ (キャプチャ割り込み) となります。また、INTP1nの要求信号およびベクタ・アドレスは、コンペア・レジスタの一致割り込みINTCC1nと兼用です。したがって、コンペア・レジスタの一致割り込みを使用する場合は、兼用となっているINTP1n端子は使用できません。

詳細は第10章 **タイマ/カウンタ機能**を参照してください。

備考 n = 0-3

第5章 バス制御機能

V831のBCUIは、EDO DRAM (Extended Data Output DRAM) , Page-ROM, SRAM (ROM) , I/Oをダイレクトに接続できます。

EDO DRAMへのアクセスは、アドレス・バス、データ・バス、4本の $\overline{\text{xxCAS}}$ と、 $\overline{\text{RAS}}$ 、 $\overline{\text{OE}}$ 、 $\overline{\text{WE}}$ 信号を使用します。DRAMへのアドレスは、ロウ・アドレス、カラム・アドレスをマルチプレクスしてアドレス端子から出力します。Page-ROM, SRAM (ROM) へのアクセスは、アドレス・バス、データ・バス、4本の $\overline{\text{xxMWR}}$ と、 $\overline{\text{MRD}}$ 、 $\overline{\text{CS}}$ 信号を使用します。

I/Oへのアクセスは、アドレス・バス、データ・バスおよび $\overline{\text{IOWR}}$ 、 $\overline{\text{IORD}}$ 、 $\overline{\text{CS}}$ 信号を使用します。

Page-ROM, SRAM (ROM) , I/Oへのアクセスは、 $\overline{\text{READY}}$ 信号によるウェイト制御もできます。

外部バス・マスタとのアクセス調停には、 $\overline{\text{HLDRQ}}$ 、 $\overline{\text{HLDAK}}$ 信号を使用します。

- 備考1 . $\overline{\text{xxCAS}}$: $\overline{\text{UUCAS}}$, $\overline{\text{ULCAS}}$, $\overline{\text{LUCAS}}$, $\overline{\text{LLCAS}}$
 2 . $\overline{\text{xxMWR}}$: $\overline{\text{UUMWR}}$, $\overline{\text{ULMWR}}$, $\overline{\text{LUMWR}}$, $\overline{\text{LLMWR}}$

5.1 特 徴

EDO DRAM, Page-ROM, SRAM (ROM) , I/Oをダイレクトに接続可能

最小1バス・クロックの $\overline{\text{CAS}}$ アクセス

4本の $\overline{\text{CAS}}$ 信号によるDRAMへのバイト・アクセス制御

$\overline{\text{READY}}$ 信号によるウェイト制御

5.2 外部I/Oサイクル

- ★ I/O空間のブロック3-6領域にIN/OUT命令によるアクセスがあり、かつ、BCTCレジスタによってI/Oサイクルが選択されていると、I/Oサイクルを行います。I/Oサイクルはシングル・サイクルだけで、基本サイクルは T_a 、 T_s ステートの2バス・クロック・サイクルです (図5 - 1参照) 。

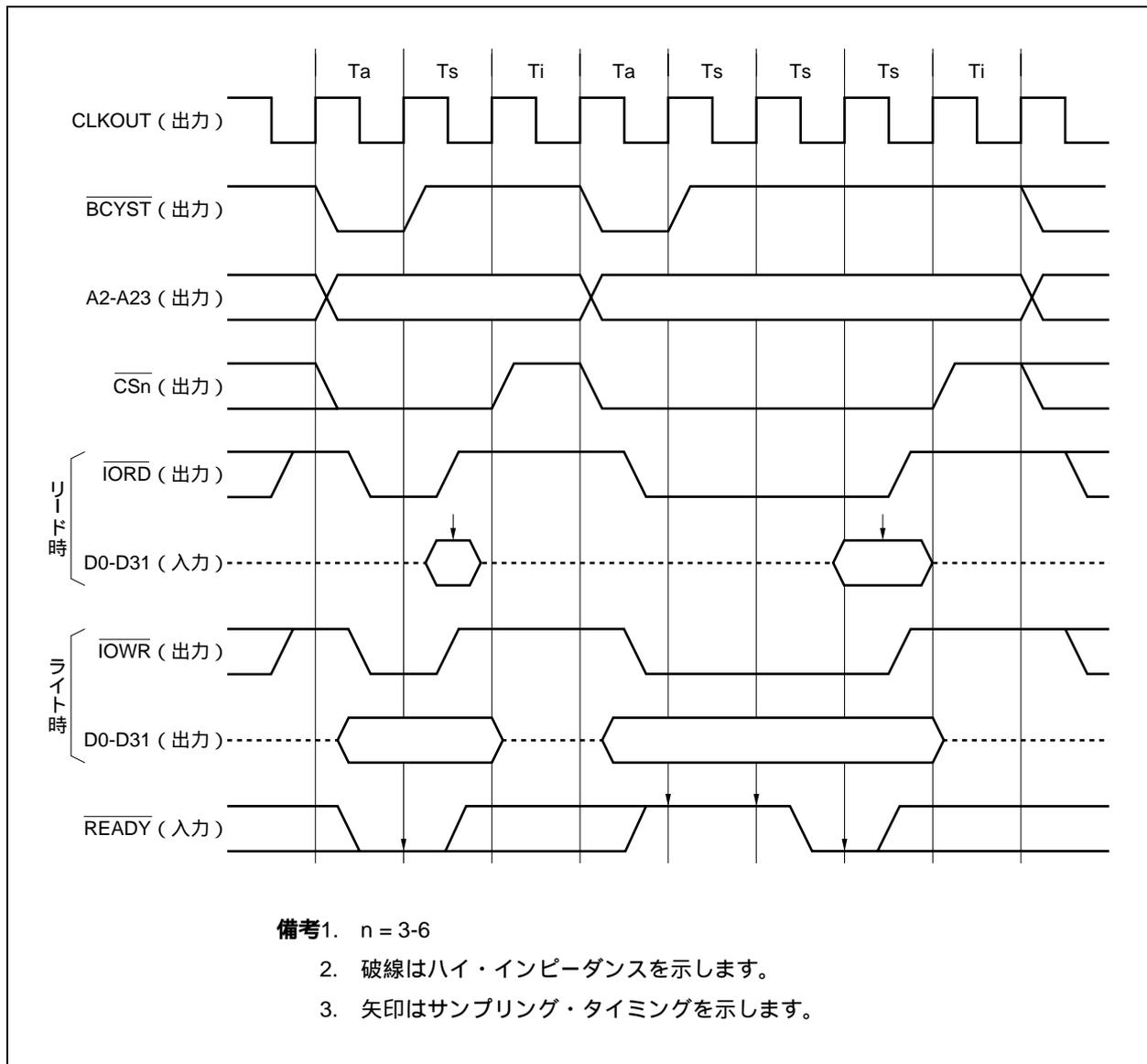
T_a ステートは、外部デバイスの制御信号の出力を開始するステートです。バス・クロックの立ち上がりでアドレスを出力して、 $\overline{\text{CS}}$ 信号がアクティブになります。 $\overline{\text{BCYST}}$ 信号は、 T_a ステート期間 (1バス・クロック) アクティブになります。リード/ライト時は、それぞれ $\overline{\text{IORD}}$ 信号と $\overline{\text{IOWR}}$ 信号がバス・クロックの立ち下がりでアクティブになります。ライト時は、バス・クロックの立ち上がりでライト・データを出力します。

T_s ステートは、外部デバイスのデータをリード/ライトするステートです。リード時は、バス・クロックの立ち下がりデータをサンプリングします。ライト時は、 T_s ステート期間中、ライト・データを出力し続けます。 $\overline{\text{READY}}$ 信号は、 T_s ステートのバス・クロックの立ち上がりでサンプリングします。 $\overline{\text{READY}}$ 信号がアクティブなら、 $\overline{\text{IORD}}$ または $\overline{\text{IOWR}}$ 信号をインアクティブにして、リード/ライト・サイクルを終了します。 $\overline{\text{READY}}$ 信号がインアクティブなら、もう一度 T_s ステートを実行します。

また、CPUが行うI/Oリード・サイクル後は、必ず T_i ステートが1バス・クロック強制挿入されます。

ウェイト制御は、 $\overline{\text{READY}}$ 信号のほかに、PWC0/PWC1レジスタで制御できます。PWC0/PWC1レジスタで設定したウェイト数と $\overline{\text{READY}}$ 入力によるウェイト数は論理和の形をとるため、どちらか多い方のウェイト数が挿入されます。

図5-1 外部I/Oサイクル (32ビット・バス・モード)



5.2.1 バイト・アクセス制御

I/Oライト・ストローブ信号がIOWR信号だけのために、外部I/Oはバイト・アクセス制御ができません。したがって、32ビット・データ・バス時はワード（32ビット）ごとに、16ビット・データ・バス時はハーフワード（16ビット）ごとにI/Oを接続してください。

32ビット・データ・バスでのハーフワード・アクセス時には、A1が変化します。

5.3 SRAM (ROM) サイクル

メモリ空間のブロック1-7領域にアクセスがあり、BCTCレジスタによってSRAM (ROM) サイクルを選択すると、SRAM (ROM) サイクルを行います。SRAM (ROM) サイクルには、連続アクセスするデータ長、データ・バス幅の違いにより次の種類に分類されます。

(1) 連続アクセスするデータ長による分類

シングル・サイクル (アクセス単位は4/2/1バイト)

- ・ ST命令実行によるSRAM (ROM) アクセス
- ・ アンキャッシュャブル領域へのLD命令実行または命令フェッチによるSRAM (ROM) アクセス
- ・ DMAの2サイクル転送によるSRAM (ROM) アクセス

バースト・サイクル (アクセス単位は16バイト)

- ・ 命令/データ・キャッシュのリフィルによるSRAM (ROM) アクセス
- ・ 内蔵RAMとのブロック転送命令実行によるSRAM (ROM) アクセス

(2) データ・バス幅による分類

DBCレジスタのBWnビットでデータ・バス幅を設定します (n = 1-6)。

- ・ BWnビット = 1のとき : 16ビット・バス・モード
- ・ BWnビット = 0のとき : 32ビット・バス・モード

図5-2 16 M ROM (1 M × 16) との接続例 (32ビット・バス・モード時)

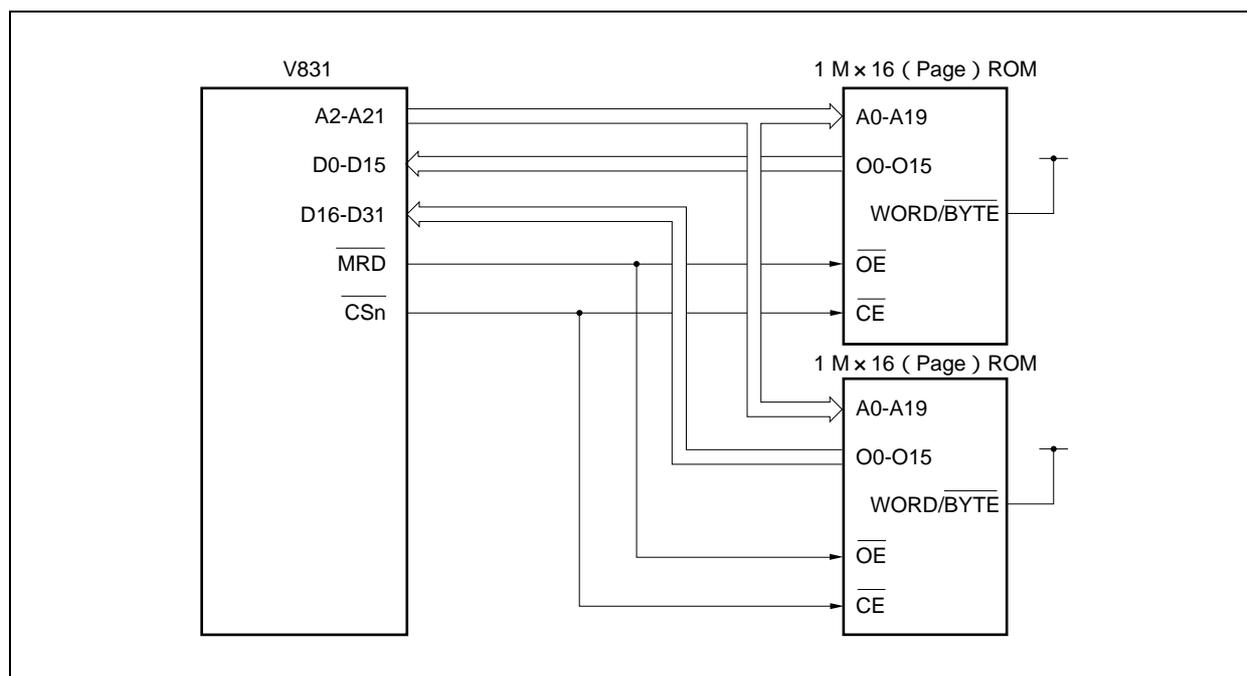
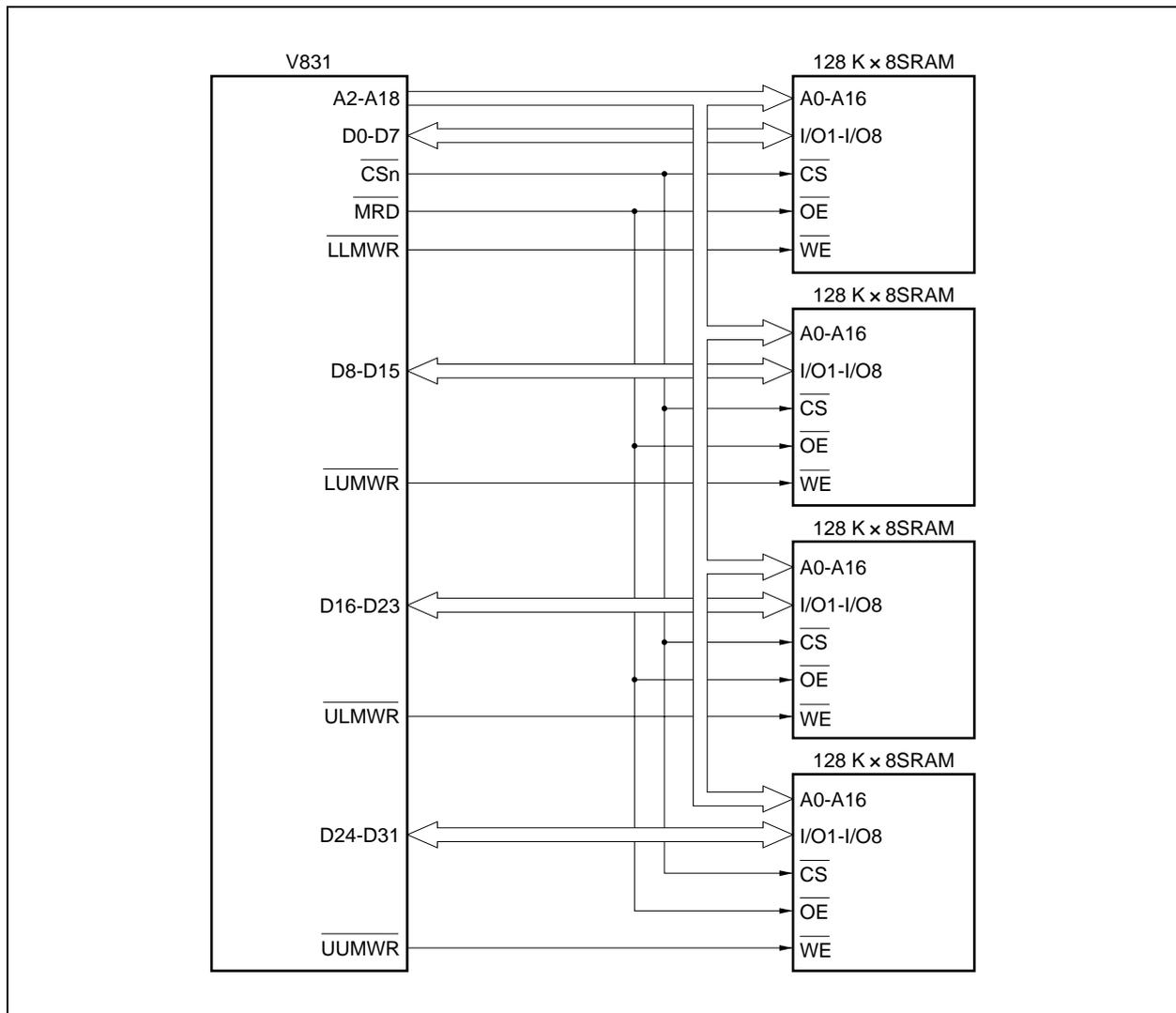


図5-3 1 M SRAM (128 K × 8) との接続例 (32ビット・バス・モード時)



5.3.1 SRAM (ROM) シングル・サイクル

SRAM (ROM) シングル・サイクルは、ST命令実行、アンキャッシュブルLD命令実行または命令フェッチ、DMA2サイクル転送で、SRAM (ROM) サイクルを設定しているブロックにアクセスすると起動します。基本サイクルはTa、Tsステートの2バス・クロック・サイクルです（図5 - 4参照）。

Taステートは、外部デバイスの制御信号の出力を開始するステートです。バス・クロックの立ち上がりでアドレスを出力して、 \overline{CS} 信号がアクティブになります。 \overline{BCYST} 信号は、Taステート期間（1バス・クロック）アクティブになります。リード/ライト時には、それぞれMRD信号とxxMWR信号がバス・クロックの立ち下がりでアクティブになります。ライト時には、バス・クロックの立ち上がりでライト・データを出力します。

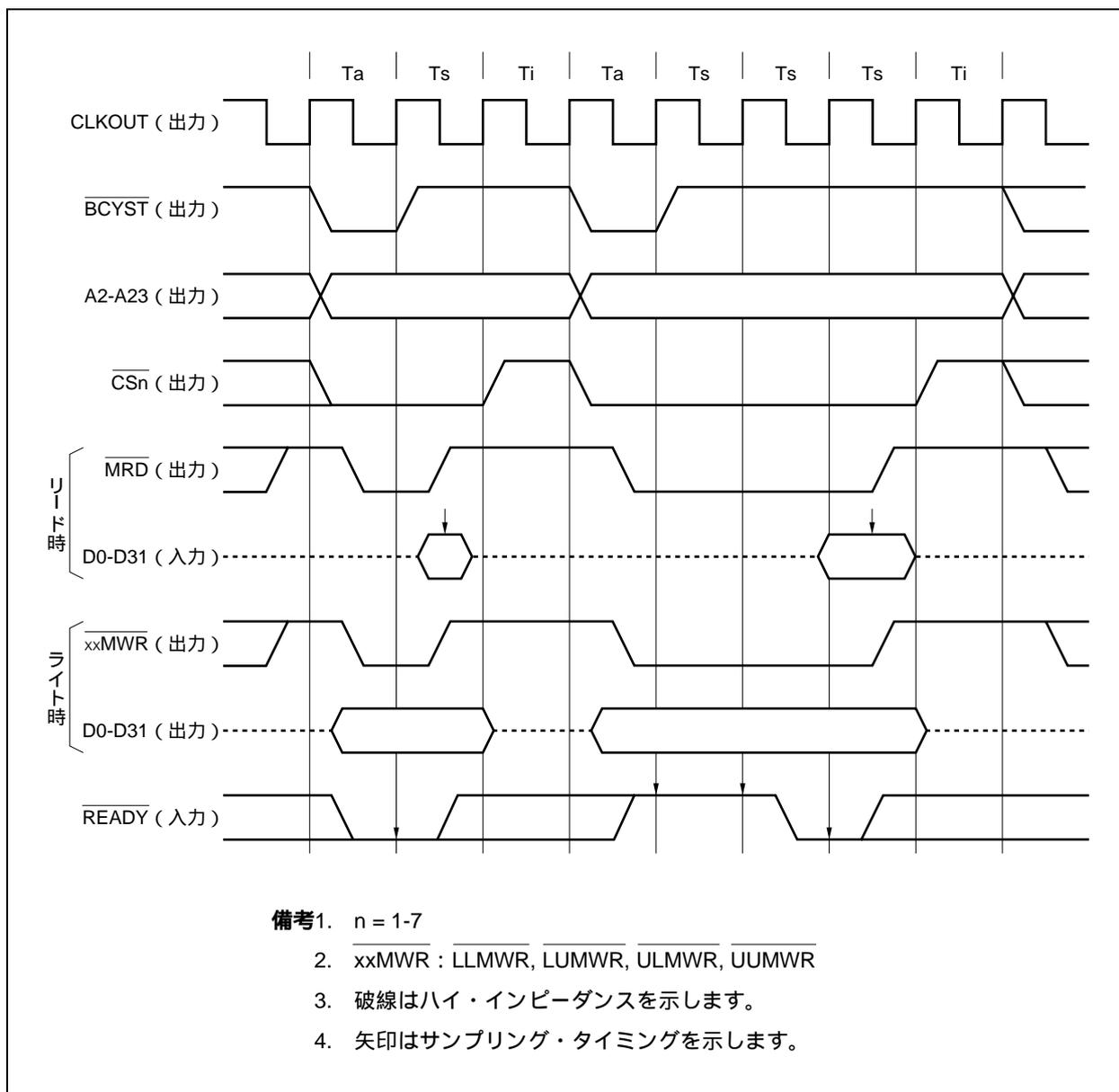
Tsステートは、外部デバイスのデータをリード/ライトするステートです。リード時には、バス・クロックの立ち下がりですべてデータをサンプルします。ライト時は、Tsステート期間中、ライト・データを出力し続けます。 \overline{READY} 信号は、Tsステートのバス・クロックの立ち上がりでサンプルします。 \overline{READY} 信号がアクティブになると、MRDまたはxxMWR信号をインアクティブにしてサイクルを終了します。インアクティブの場合は、もう一度Tsステートを実行します。

またCPUが行うSRAM (ROM) リード・サイクル後は、必ずTiステートが1バス・クロック強制挿入されます。

ウェイト制御は、PWC0/PWC1レジスタによる制御と \overline{READY} 端子による制御ができます（図5 - 4参照）。PWC0/PWC1レジスタで設定したウェイト数と \overline{READY} 入力によるウェイト数は論理和の形を取るため、どちらが多い方のウェイト数が挿入されます。

備考 xxMWR : UUMWR, ULMWR, LUMWR, LLMWR

図5-4 SRAM (ROM) シングル・サイクル (32ビット・バス・モード)



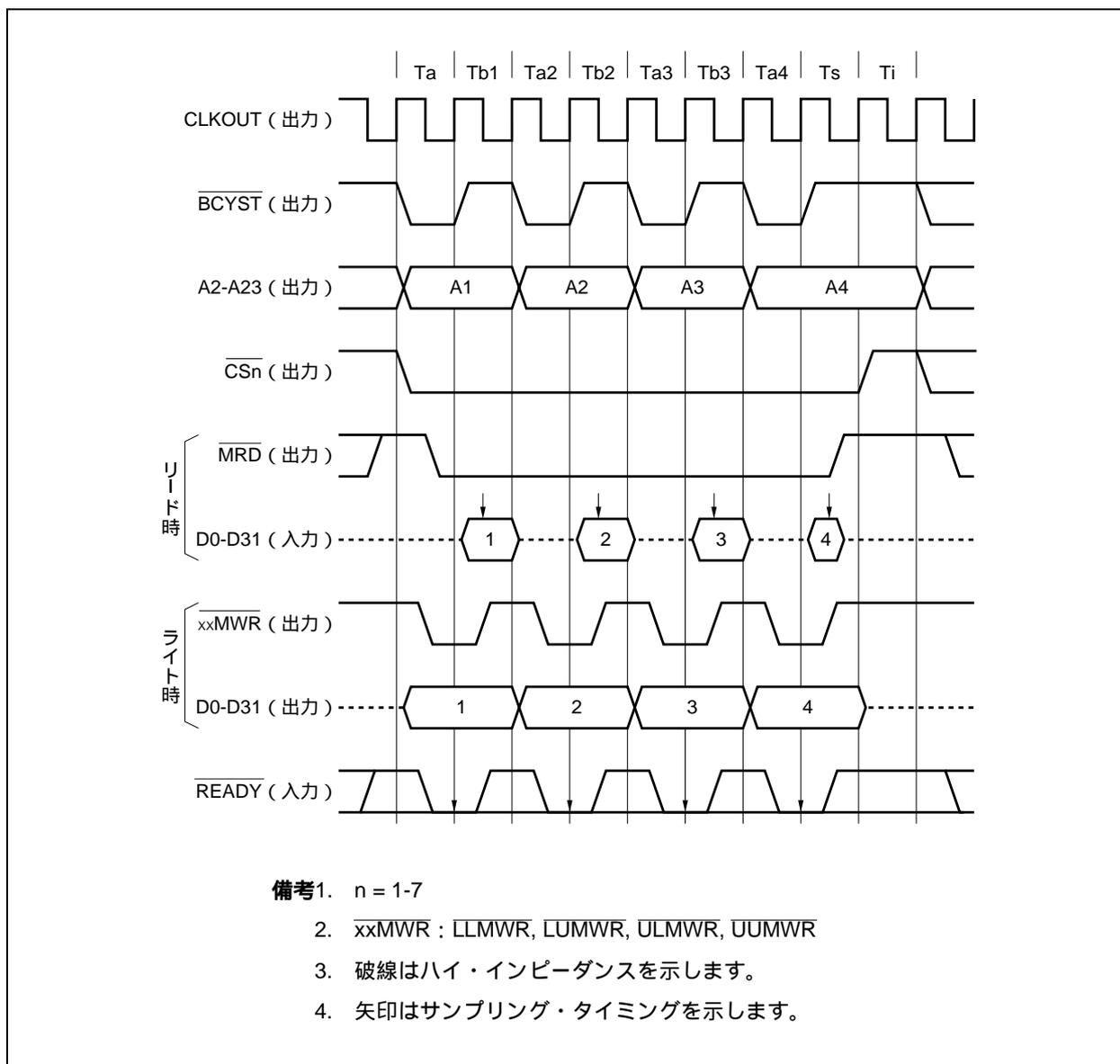
5.3.2 SRAM (ROM) パースト・サイクル

SRAM (ROM) パースト・サイクルは、命令/データ・キャッシュ・リフィル、内蔵RAMとのブロック転送命令実行でSRAM (ROM) サイクルを設定しているブロックにアクセスすると起動します。基本アクセスは、2バス・クロック・アクセスです。32ビット・バス・モードの場合、1回のパースト・サイクルで、4回の連続したアクセスが発生します（図5 - 5参照）。

1回のアクセスでのSRAM (ROM) の制御信号のタイミングは、 $\overline{\text{MRD}}$ 信号を除いて、シングル・サイクルと同じです。 $\overline{\text{MRD}}$ 信号は、バス・クロックの立ち上がり同期して、パースト・サイクルの期間はアクティブの状態になります。 $\overline{\text{BCYST}}$ 信号は、各アクセスのTa-Ta4ステート期間アクティブになります。 $\overline{\text{READY}}$ 信号は、Tb1-Tb3, Tsステートのバス・クロックの立ち上がりでサンプリングします。

ウェイト制御は、PWC0/PWC1レジスタによる制御と $\overline{\text{READY}}$ 端子による制御ができます。PWC0/PWC1レジスタで設定したウェイト数と $\overline{\text{READY}}$ 入力によるウェイト数は論理和の形をとるため、どちらか多い方のウェイト数が挿入されます。

図5-5 SRAM (ROM) パースト・サイクル



5.3.3 バイト・アクセス制御

4本の $\overline{\text{xxMWR}}$ 信号 ($\overline{\text{UUMWR}}$, $\overline{\text{ULMWR}}$, $\overline{\text{LUMWR}}$, $\overline{\text{LLMWR}}$) によりバイト・アクセスを制御します。データ・バス幅が32ビットのときは4本の $\overline{\text{xxMWR}}$ 信号を使用して、データ・バス幅が16ビットのときは2本の $\overline{\text{xxMWR}}$ 信号 ($\overline{\text{LUMWR}}$, $\overline{\text{LLMWR}}$) を使用します。

次に $\overline{\text{xxMWR}}$ 信号とアクセス・アドレスとの関係を示します。

表5-1 32ビット・データ・バス ($\overline{\text{xxMWR}}$)

データ・サイズ	アドレス		$\overline{\text{xxMWR}}$			
	A1	A0	$\overline{\text{UUMWR}}$	$\overline{\text{ULMWR}}$	$\overline{\text{LUMWR}}$	$\overline{\text{LLMWR}}$
バイト	0	0	1	1	1	0
	0	1	1	1	0	1
	1	0	1	0	1	1
	1	1	0	1	1	1
ハーフワード (16ビット)	0	0	1	1	0	0
	1	0	0	0	1	1
ワード (32ビット)	0	0	0	0	0	0

備考 1: ハイ・レベル出力

0: ロウ・レベル出力

表5-2 16ビット・データ・バス ($\overline{\text{xxMWR}}$)

データ・サイズ		アドレス	$\overline{\text{xxMWR}}$			
		A0	$\overline{\text{UUMWR}}$	$\overline{\text{ULMWR}}$	$\overline{\text{LUMWR}}$	$\overline{\text{LLMWR}}$
バイト		0	1	1	1	0
		1	1	1	0	1
ハーフワード (16ビット)		0	1	1	0	0
ワード (32ビット)	1回目	0	1	1	0	0
	2回目	0	1	1	0	0

備考 1: ハイ・レベル出力

0: ロウ・レベル出力

5.4 Page-ROMサイクル

メモリ空間のブロック7領域にアクセスがあり、BCTCレジスタのCT7ビットによってPage-ROMサイクルを選択していると、Page-ROMサイクルが行われます。Page-ROMサイクルには、連続アクセスするデータ長、データ・バスの幅の違いにより次の種類に分類されます。

(1) アクセスするデータ長による分類

シングル・サイクル（アクセス単位は4/2/1バイト）

- ・ アンキャッシュブル領域へのLD命令実行または命令フェッチによるPage-ROMアクセス
- ・ DMAの2サイクル転送によるPage-ROMアクセス

バースト・サイクル（アクセス単位は16バイト）

- ・ 命令キャッシュのリフィルによるPage-ROMアクセス
- ・ 内蔵RAMとのブロック転送命令実行によるPage-ROMアクセス

(2) データ・バス幅による分類

BT16B端子でデータ・バス幅を設定します。バースト・サイクルで32ビット・バス・モードの場合は4回連続したアクセスを行い、16ビット・バス・モードの場合は8回連続したアクセスを行います。

- ・ BT16B端子 = 1のとき：16ビット・バス・モード
- ・ BT16B端子 = 0のとき：32ビット・バス・モード

5.4.1 Page-ROMシングル・サイクル

アンキャッシュブルLD命令実行または命令フェッチ、DMA2サイクル転送でPage-ROMサイクルが選択されているブロック7のアクセス時に起動するサイクルです。ウエイト制御は、 $\overline{\text{READY}}$ 端子制御および内部レジスタ制御ができます。PWC1レジスタのWS7ビットによってウエイトを制御します。バス・タイミングは、SRAM（ROM）シングル・サイクルと同じです。

5.4.2 Page-ROMバースト・サイクル

命令キャッシュ・リフィル、内蔵RAMとのブロック転送命令実行で、Page-ROMサイクルが選択されているブロック7のアクセス時に起動するサイクルです。バス・タイミングは、SRAM（ROM）バースト・サイクルと同じですが、ウエイト数の設定が異なります。

32ビット・バス・モード時、または16ビット・バス・モード時にPRCレジスタのPSビットが0のとき（ページ・サイズが16バイト）、1回目のアクセスは通常のアクセス（off-page）を行い、2回目以降は、on-pageアクセスを行います。1アクセスの基本サイクルは2バス・クロック・サイクルです（図5-6参照）。1回目のoff-pageアクセス時は、PWC1レジスタのWS7ビットによってウエイトを制御します。2回目以降のon-pageアクセス時は、PRCレジスタのPWSビットによってウエイトを制御します。また、 $\overline{\text{READY}}$ 端子によるウエイト制御もできます。

16ビット・バス・モード時でPRCレジスタのPSビットが1のとき（ページ・サイズが8バイト）は、1回目と5回目のアクセスは通常のアクセス（off-page）を行い、2回目～4回目と6回目～8回目は、on-pageアクセスを行います。各アクセスのウエイト数は、32ビット・バス・モード時と同じです。

図5-6 Page-ROMバースト・サイクル (32ビット・バス・モード)

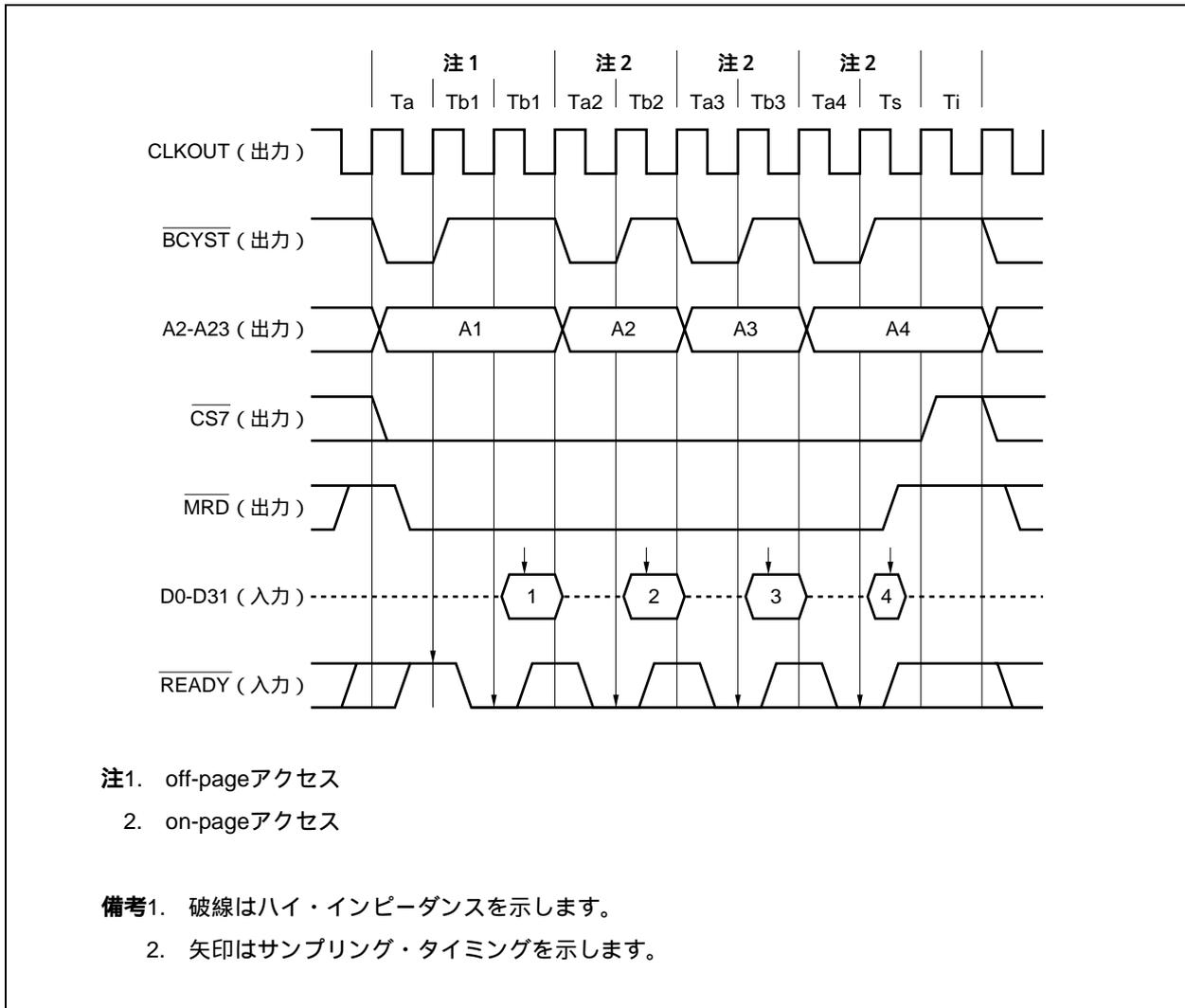
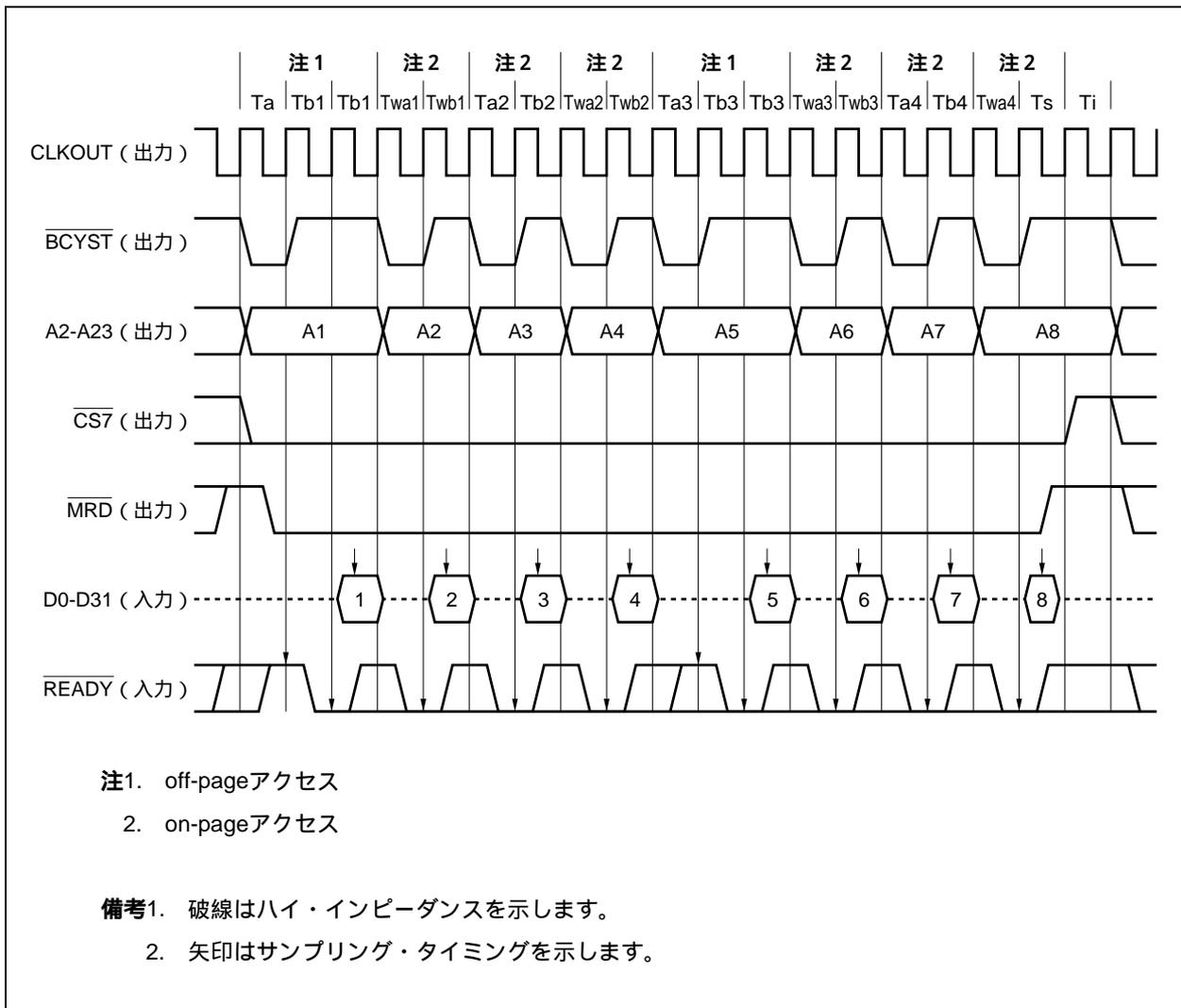


図5-7 Page-ROMバースト・サイクル(16ビット・バス・モード, 8バイト・ページ・サイズ)



5.5 DRAMサイクル

メモリ空間のブロック0領域にアクセスすると、EDO DRAMサイクルを行います。EDO DRAMサイクルには、連続アクセスするデータ長、on-page/off-page、 $\overline{\text{CAS}}$ サイクル期間、データ・バス幅の違いにより次の種類に分類されます。

(1) 連続アクセスするデータ長による分類

シングル・サイクル（アクセス単位は4/2/1バイト）

- ・ ST命令実行によるDRAMアクセス
- ・ アンキャッシュャブル領域へのLD命令実行または命令フェッチによるDRAMアクセス
- ・ DMAの2サイクル転送によるDRAMアクセス

バースト・サイクル（アクセス単位は16バイト）

- ・ 命令/データ・キャッシュのリフィルによるDRAMアクセス
- ・ 内蔵RAMとのブロック転送命令実行によるDRAMアクセス

(2) on-page/off-pageによる分類

on-pageサイクル

- ・ $\overline{\text{RAS}}$ アクティブ、かつ前回のDRAMサイクルと同一ロウ・アドレス

off-pageサイクル

- ・ $\overline{\text{RAS}}$ インアクティブ
- ・ $\overline{\text{RAS}}$ アクティブ、かつ前回のDRAMサイクルと異なるロウ・アドレス

(3) $\overline{\text{CAS}}$ サイクル期間による分類

DRCレジスタのCRWTビットでリード/ライト時の $\overline{\text{CAS}}$ サイクル期間を設定します。

1クロック $\overline{\text{CAS}}$ サイクル

- ・ CRWTビット= 00
- ・ CRWTビット= 10（ライト・サイクルだけ）

2クロック $\overline{\text{CAS}}$ サイクル

- ・ CRWTビット= 11
- ・ CRWTビット= 10（リード・サイクルだけ）

★ (4) データ・バス幅による分類

DBCレジスタのBW0ビットでデータ・バス幅を設定します。

- ・ BW0ビット= 0のとき：32ビット・バス・モード
- ・ BW0ビット= 1のとき：16ビット・バス・モード

図5 - 8 16 M EDO DRAM (1 M×16) との接続例 (16ビット・バス・モード時)

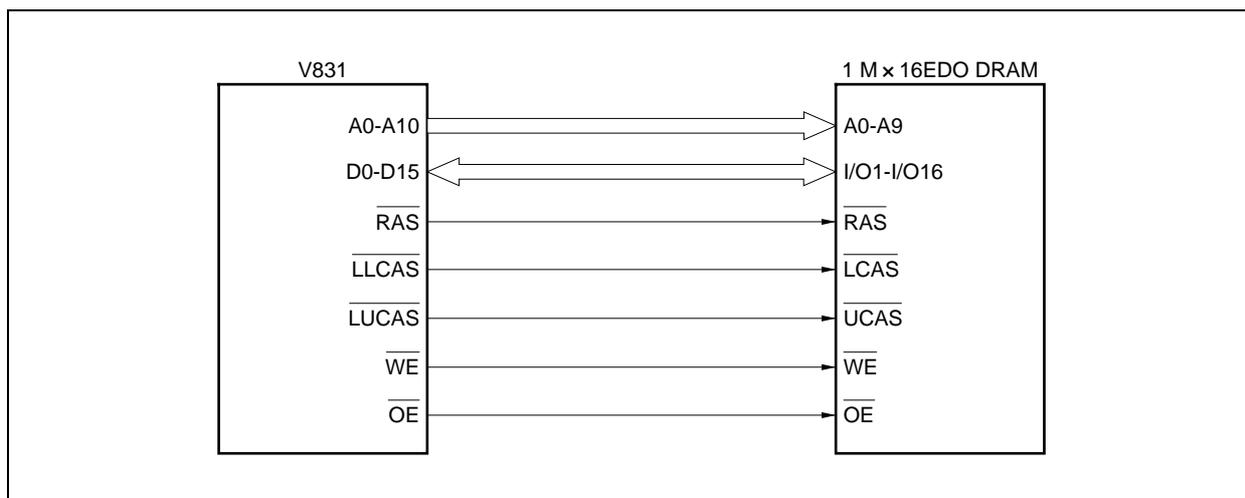
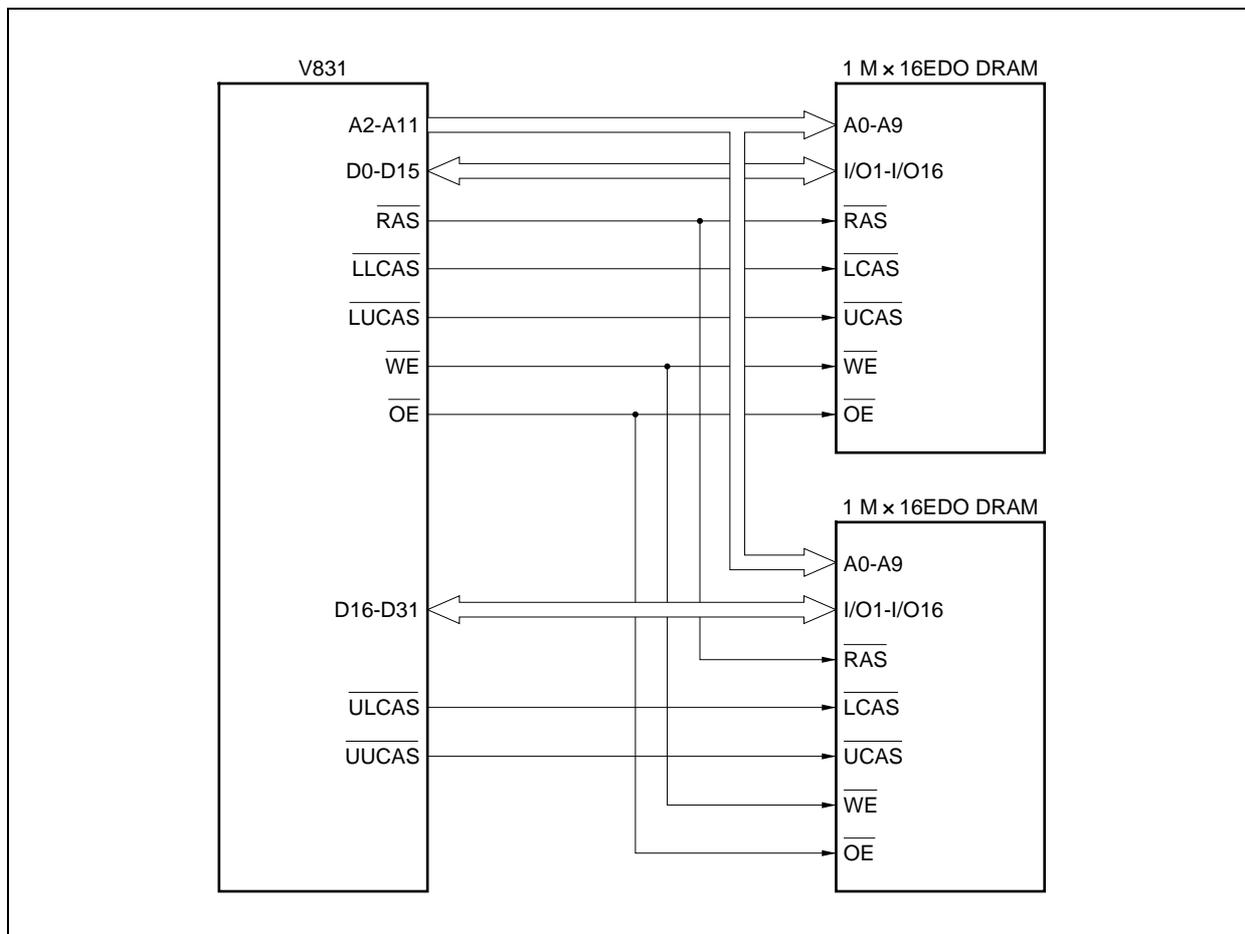


図5 - 9 16 M EDO DRAM (1 M×16) との接続例 (32ビット・バス・モード時)



5.5.1 DRAMシングル・サイクル

ST命令実行，アンキャッシュブルLD命令実行または命令フェッチ，DMA2サイクル転送でブロック0のアクセス時に起動するサイクルです。図5 - 10にシングル1クロックCAS off-page/on-pageサイクル，図5 - 11にシングル2クロックCAS off-page/on-pageサイクルのタイミングを示します。

1クロック $\overline{\text{CAS}}$ サイクルの場合，on-pageアクセス時のサイクルは T_c , T_{ce} の2バス・クロック・サイクルです。 T_{cl} は $\overline{\text{CAS}}$ アサート・ステート， T_{ce} は $\overline{\text{CAS}}$ サイクル終了ステートです。off-pageアクセス時は $\overline{\text{RAS}}$ 信号のプリチャージを行うため，on-pageアクセスの前に T_{rm} (ROWミス・ステート)， T_{rp} ($\overline{\text{RAS}}$ プリチャージ・ステート)， T_{rc} ($\overline{\text{RAS-CAS}}$ ステート)の最低3つのステートが入ります。

2クロック $\overline{\text{CAS}}$ サイクルの場合，on-pageアクセス時のサイクルは T_{ca} , T_{cn} , T_{ce} の3バス・クロック・サイクルです。 T_{cal} は $\overline{\text{CAS}}$ アサート・ステート， T_{cn} は $\overline{\text{CAS}}$ ネグート・ステート， T_{ce} は $\overline{\text{CAS}}$ サイクル終了ステートです。off-pageアクセス時は $\overline{\text{RAS}}$ のプリチャージを行うため，on-pageアクセスの前に T_{rm} (ROWミス・ステート)， T_{rp} ($\overline{\text{RAS}}$ プリチャージ・ステート)， T_{rc} ($\overline{\text{RAS-CAS}}$ ステート)の最低3つのステートが入ります。

CPUから起動されたリード・サイクルのあとには，必ず1ステートの T_i サイクルが強制的に挿入されます。

図5 - 10 DRAMシングルクロックCAS on-page/off-pageサイクル (32ビット・バス・モード)

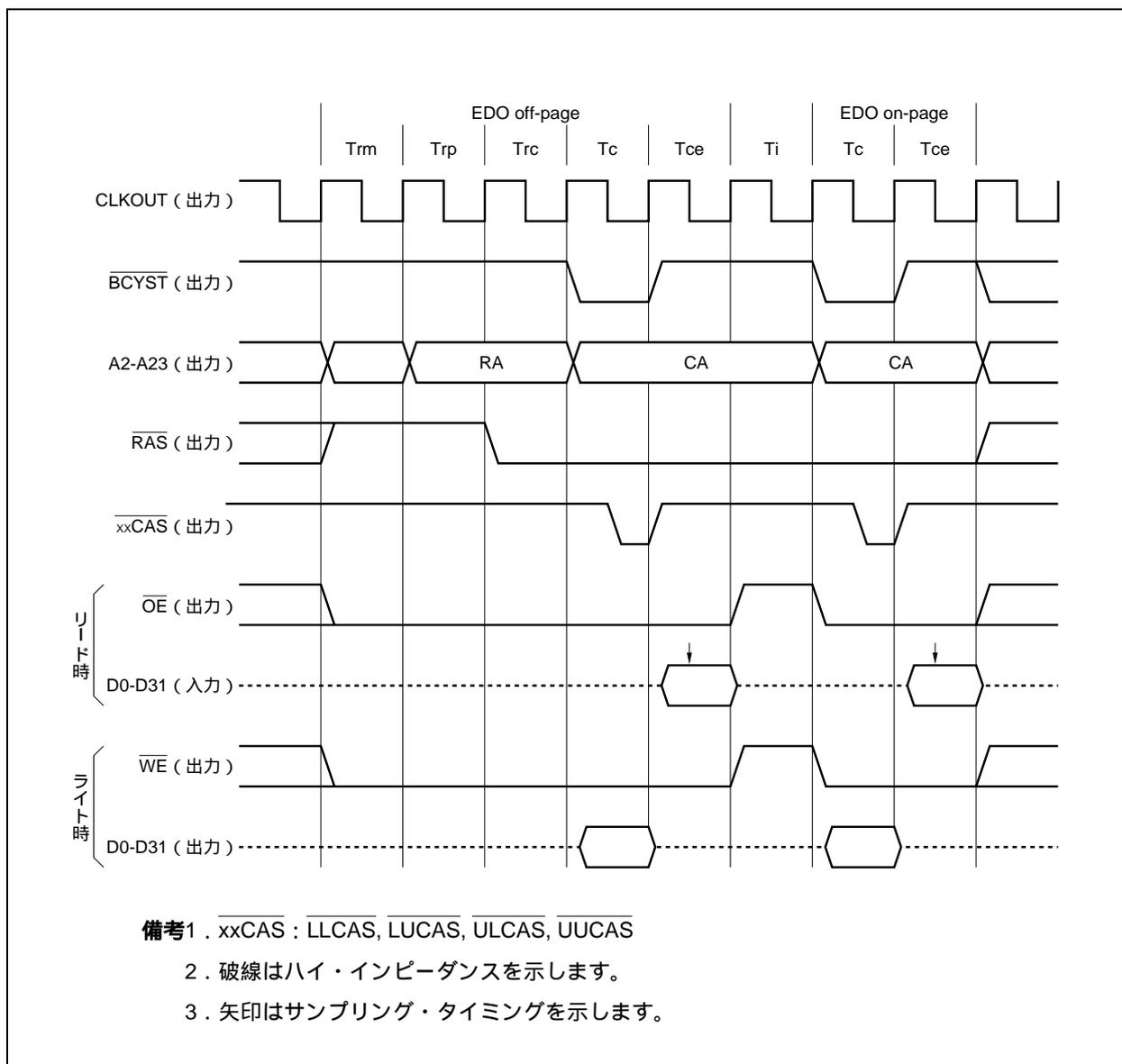
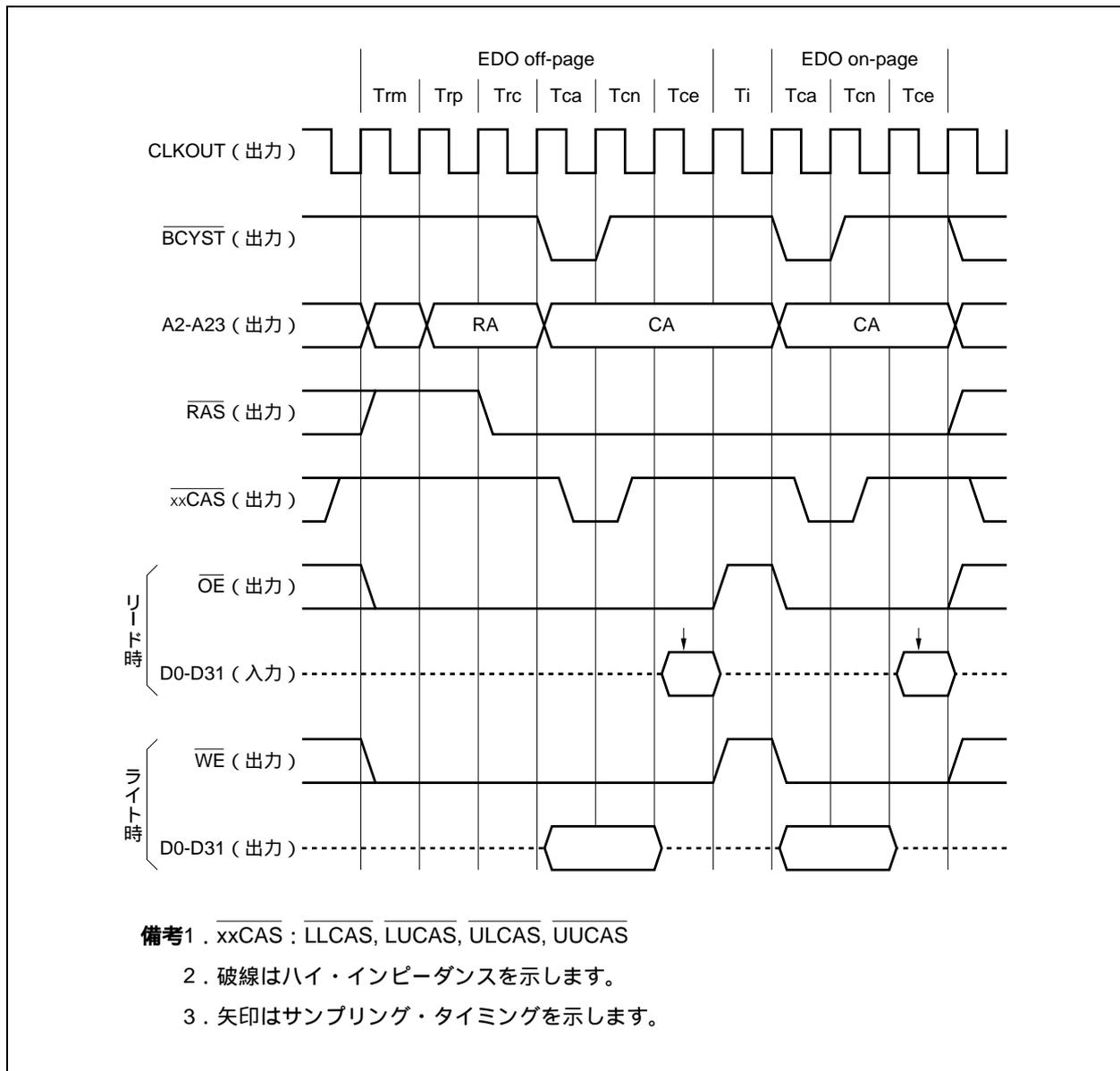


図5 - 11 DRAMシングル2クロックCAS on-page/off-pageサイクル (32ビット・バス・モード)



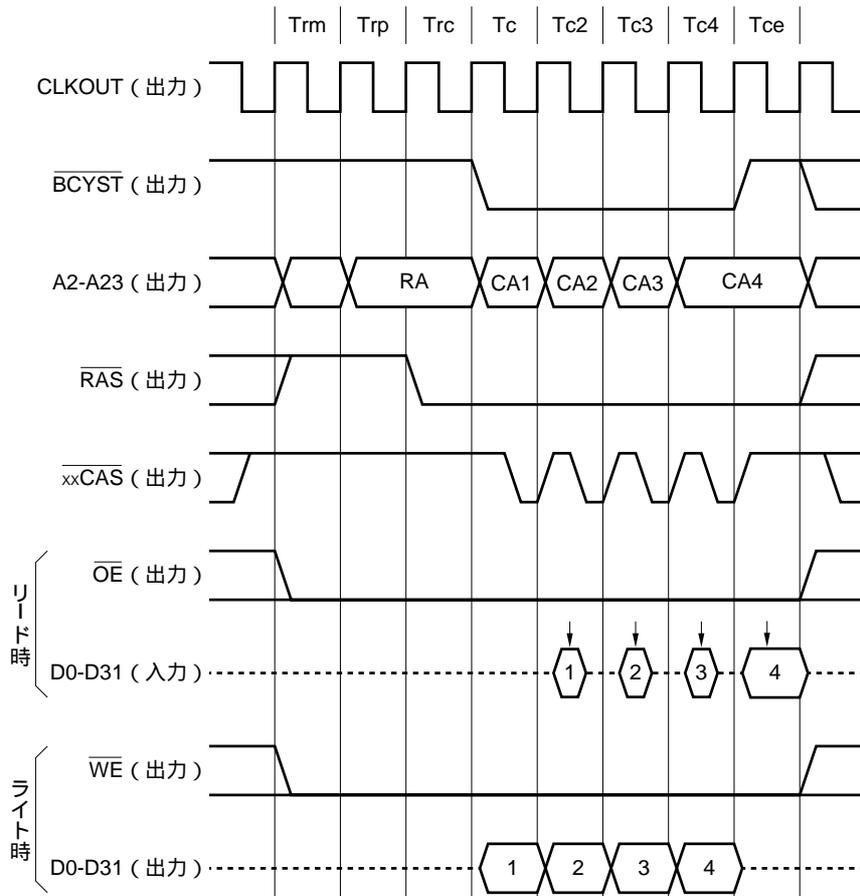
5.5.2 DRAMバースト・サイクル

DRAMバースト・サイクルは、命令/データ・キャッシュ・リフィル、内蔵RAMとのブロック転送命令実行でブロック0にアクセスすると起動します。16バイト連続のバス・サイクルです。2回目以降は、ページ・アクセスを行います。図5 - 12にバースト1クロック $\overline{\text{CAS}}$ off-pageサイクル、図5 - 13にバースト1クロック $\overline{\text{CAS}}$ on-pageサイクルを示します。図5 - 14にバースト2クロック $\overline{\text{CAS}}$ off-pageサイクル、図5 - 15にバースト2クロック $\overline{\text{CAS}}$ on-pageサイクルのタイミングを示します。

1クロック $\overline{\text{CAS}}$ サイクルの場合、on-pageアクセス時のサイクルは5バス・クロック・サイクル(T_c - T_{c4} , T_{ce})です。 T_c - T_{c4} はそれぞれ4回のアクセスの $\overline{\text{CAS}}$ アサート・ステート、 T_{ce} は $\overline{\text{CAS}}$ サイクル終了ステートです。off-pageアクセス時は $\overline{\text{RAS}}$ プリチャージを行うため、on-pageアクセスの前に最低3つのステート(T_{rm} , T_{rp} , T_{rc})が入ります。

2クロック $\overline{\text{CAS}}$ サイクルの場合、on-pageアクセス時のサイクルは9バス・クロック・サイクル(T_{ca} - T_{cn4} , T_{ce})です。 T_{ca} - T_{ca4} はそれぞれ4回のアクセスの $\overline{\text{CAS}}$ アサート・ステート、 T_{cn1} - T_{cn4} は $\overline{\text{CAS}}$ ネゲート・ステート、 T_{ce} は $\overline{\text{CAS}}$ サイクル終了ステートです。off-pageアクセス時は $\overline{\text{RAS}}$ プリチャージを行うため、on-pageアクセスの前に最低3つのステート(T_{rm} , T_{rp} , T_{rc})が入ります。

図5 - 12 DRAMバースト1クロックCAS off-pageサイクル (32ビット・バス・モード)



備考1 . xxCAS : LLCAS, LUCAS, ULCAS, UUCAS

2 . 破線はハイ・インピーダンスを示します。

3 . 矢印はサンプリング・タイミングを示します。

図5 - 13 DRAMバースト1クロックCAS on-pageサイクル (32ビット・バス・モード)

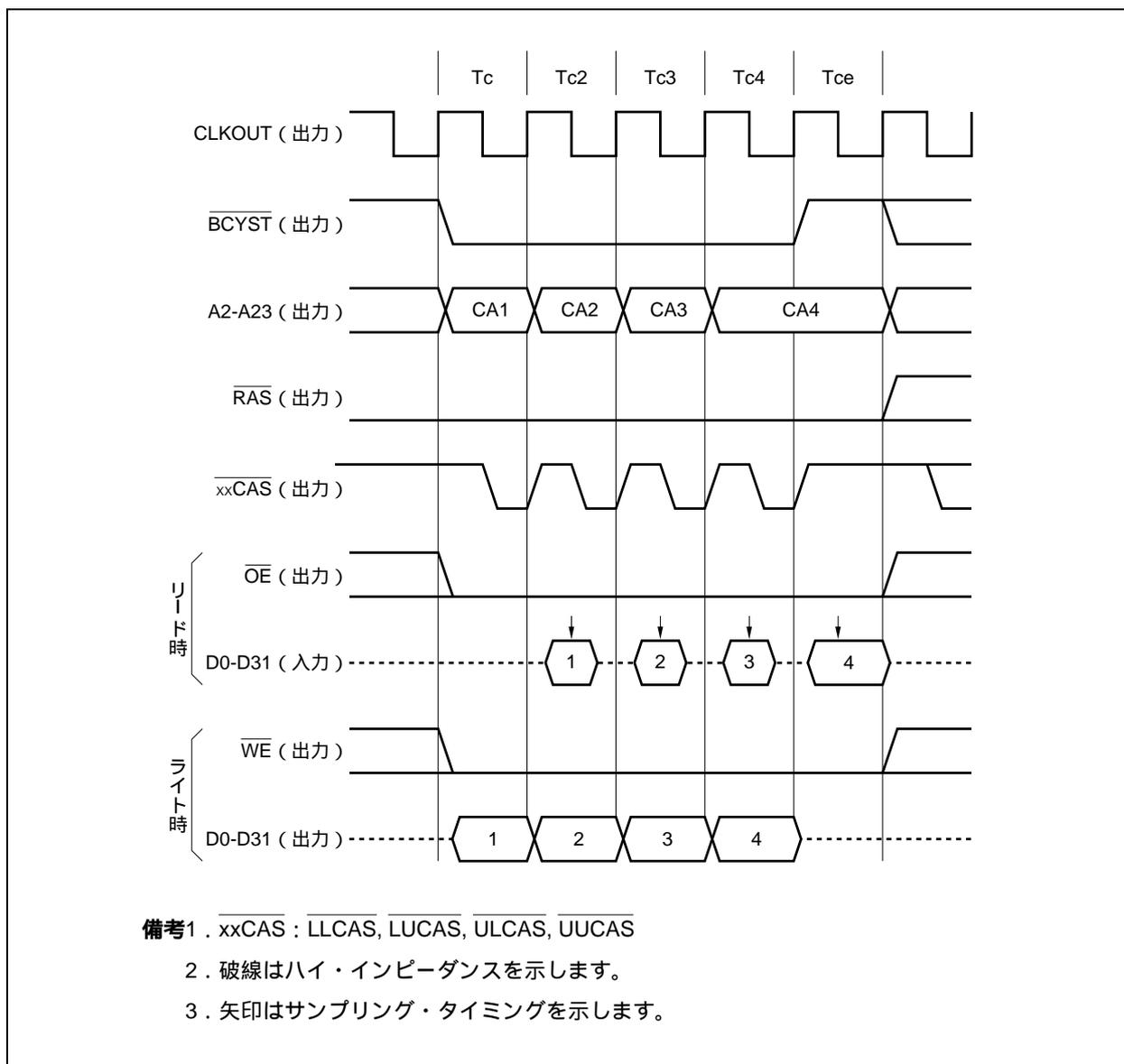


図5 - 14 DRAMバースト2クロックCAS off-pageサイクル (32ビット・バス・モード)

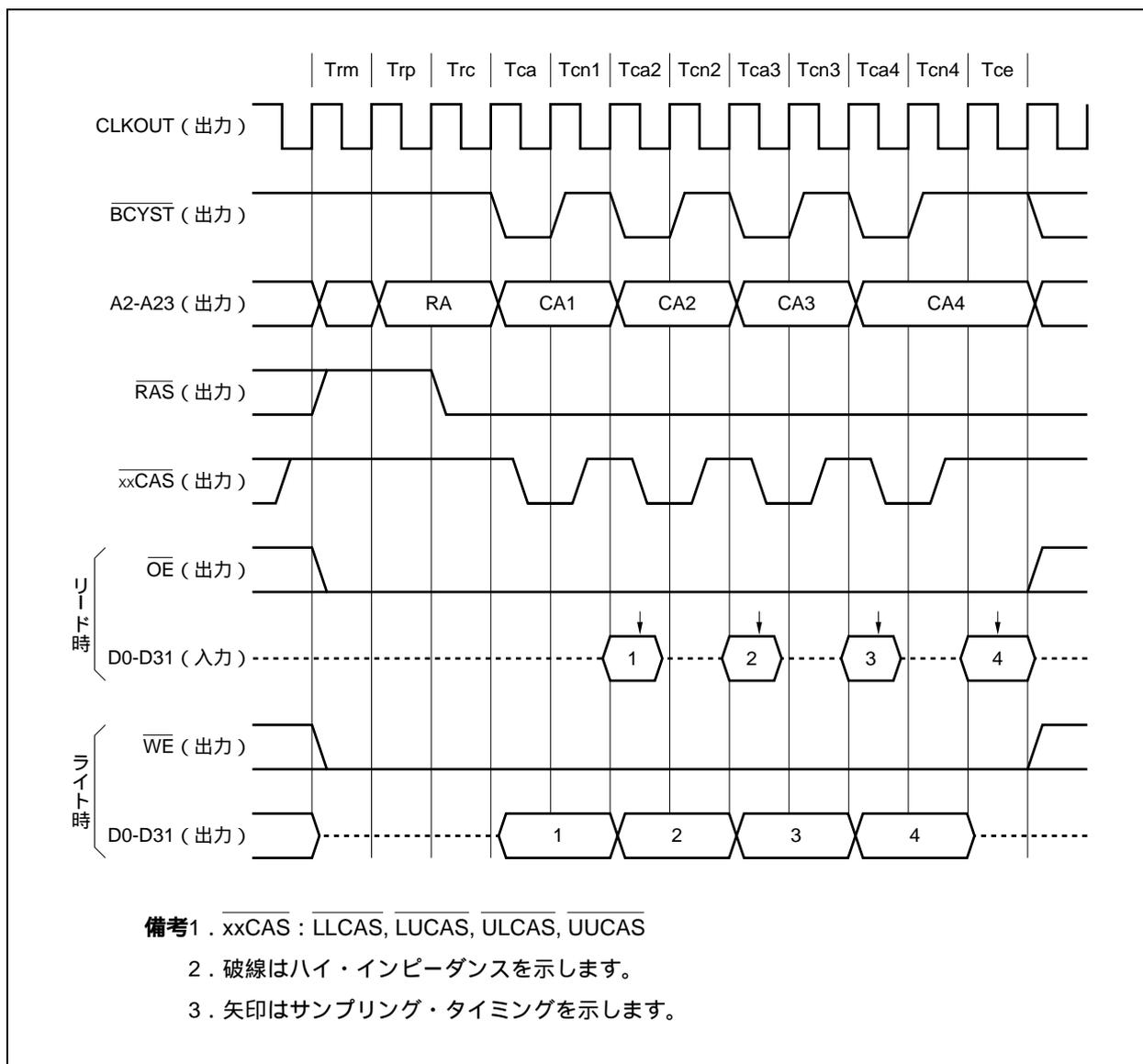
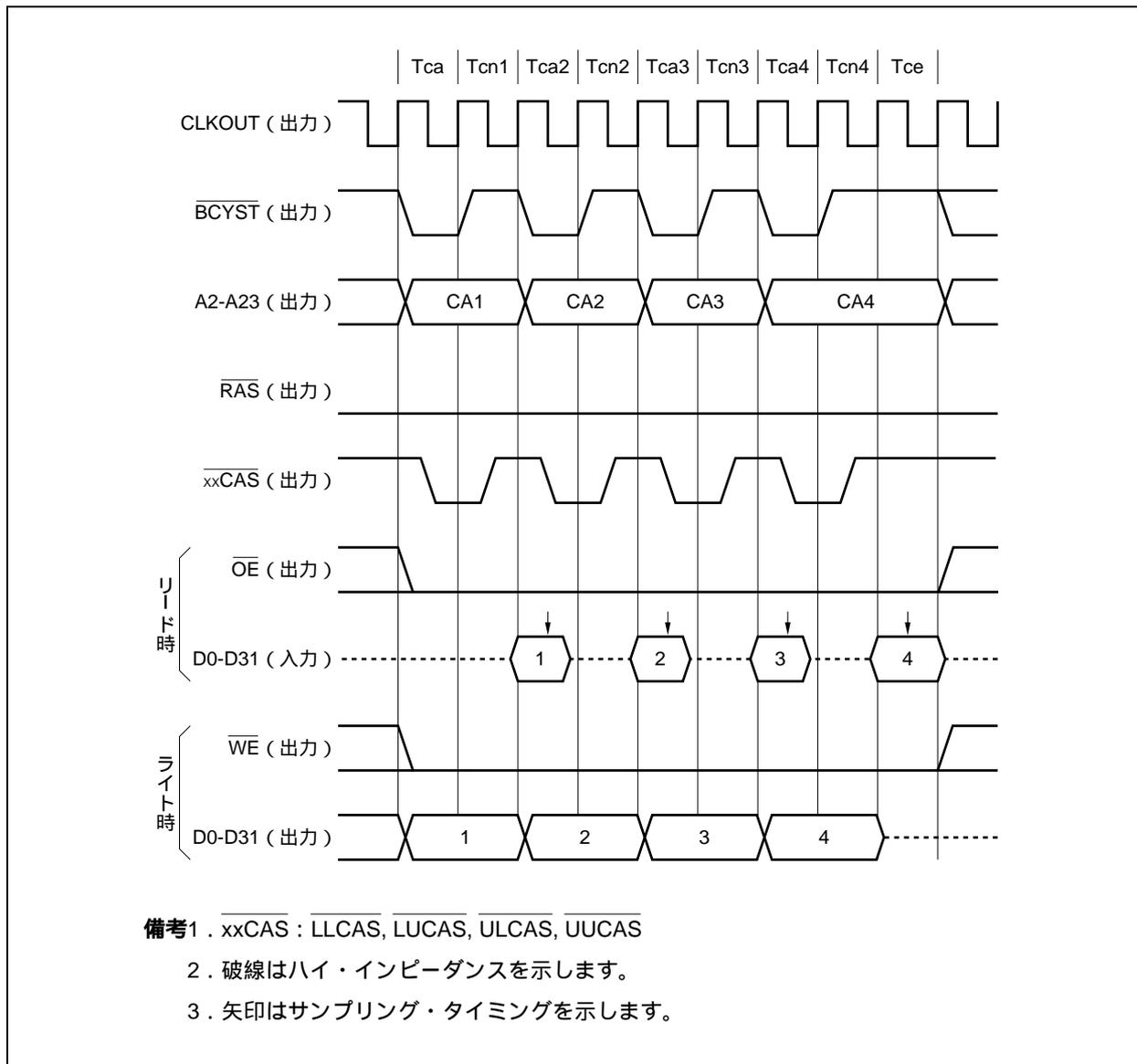


図5 - 15 DRAMバースト2クロックCAS on-pageサイクル (32ビット・バス・モード)



- 備考1 . xxCAS : LLCAS, LUCAS, ULCAS, UUCAS
 2 . 破線はハイ・インピーダンスを示します。
 3 . 矢印はサンプリング・タイミングを示します。

5.5.3 タイミング制御

V831のDRAMアクセス・タイミング制御は、 $\overline{\text{CAS}}$ サイクル期間以外に、 $\overline{\text{RAS}}$ プリチャージ期間、 $\overline{\text{RAS-CAS}}$ 遅延期間を設定できます。ただし、DRAMサイクル時は、 $\overline{\text{READY}}$ 端子によるウェイト制御はできません。

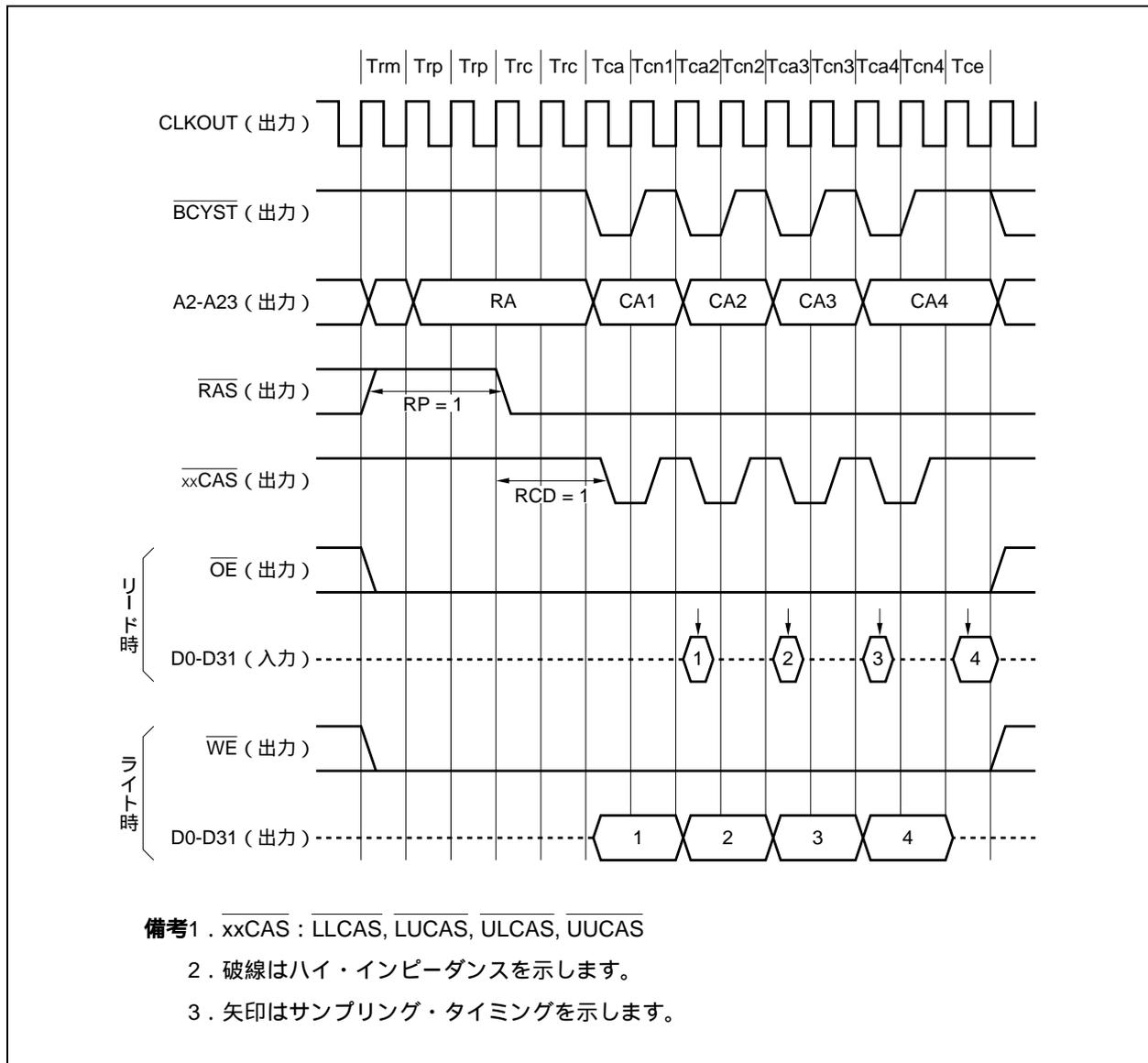
(1) $\overline{\text{RAS}}$ プリチャージ期間の制御

DRCレジスタのRPビットで $\overline{\text{RAS}}$ のプリチャージ期間を設定します。RPビットを0にすると $\overline{\text{RAS}}$ プリチャージ期間は2バス・クロック、1にすると3バス・クロックになります。図5-16はRPビットが1の場合を示しています。

(2) $\overline{\text{RAS-CAS}}$ 遅延時間

DRCレジスタのRCDビットでリード時の $\overline{\text{RAS-CAS}}$ 遅延期間を設定します。RCDビットを0にすると $\overline{\text{RAS-CAS}}$ 遅延期間は1.5バス・クロック、1にすると2.5バス・クロックになります。図5-16はRCDビットが1の場合を示しています。

図5-16 DRAMアクセス・タイミング (バーストoff-pageサイクル)



5.5.4 バイト・アクセス制御

4本の $\overline{\text{xxCAS}}$ 信号 ($\overline{\text{UUCAS}}$, $\overline{\text{ULCAS}}$, $\overline{\text{LUCAS}}$, $\overline{\text{LLCAS}}$) によりバイト・アクセス制御を行います。データ・バス幅が32ビットのときは、4本の $\overline{\text{xxCAS}}$ 信号を使い、データ・バス幅が16ビットのときは、2本の $\overline{\text{xxCAS}}$ 信号 ($\overline{\text{LUCAS}}$, $\overline{\text{LLCAS}}$) を使います。次に $\overline{\text{xxCAS}}$ 信号とアクセス・アドレスとの関係を示します。

表5-3 32ビット・データ・バス ($\overline{\text{xxCAS}}$)

データ・サイズ	アドレス		$\overline{\text{xxCAS}}$			
	A1	A0	$\overline{\text{UUCAS}}$	$\overline{\text{ULCAS}}$	$\overline{\text{LUCAS}}$	$\overline{\text{LLCAS}}$
バイト	0	0	1	1	1	0
	0	1	1	1	0	1
	1	0	1	0	1	1
	1	1	0	1	1	1
ハーフワード (16ビット)	0	0	1	1	0	0
	1	0	0	0	1	1
ワード (32ビット)	0	0	0	0	0	0

表5-4 16ビット・データ・バス ($\overline{\text{xxCAS}}$)

データ・サイズ		アドレス	$\overline{\text{xxCAS}}$			
		A0	$\overline{\text{UUCAS}}$	$\overline{\text{ULCAS}}$	$\overline{\text{LUCAS}}$	$\overline{\text{LLCAS}}$
バイト		0	1	1	1	0
		1	1	1	0	1
ハーフワード (16ビット)		0	1	1	0	0
ワード (32ビット)	1回目	0	1	1	0	0
	2回目	0	1	1	0	0

備考 1: ハイ・レベル出力
0: ロウ・レベル出力

5.5.5 リフレッシュ制御

CBRリフレッシュ・サイクル, CBRセルフ・リフレッシュ・サイクルを自動的に発生できます。

(1) CBRリフレッシュ・サイクル

CBRリフレッシュ・サイクルのタイミングを図5 - 17に示します。RFCレジスタのRFWビットでCBRリフレッシュ・サイクルの $\overline{\text{RAS}}$ アクティブ期間を設定できます。

$\overline{\text{RAS}}$ プリチャージ期間は, DRCレジスタのRPビットで設定します。 $\overline{\text{READY}}$ 端子によるウェイト制御はできません。

表5 - 5 $\overline{\text{RAS}}$ アクティブ期間

RFCレジスタ		$\overline{\text{RAS}}$ アクティブ期間
RFW1	RFW0	
0	0	3バス・クロック
0	1	4バス・クロック
1	0	5バス・クロック

図5 - 17 CBRリフレッシュ・サイクル (RFW1,0 = 01, RP = 1のとき)

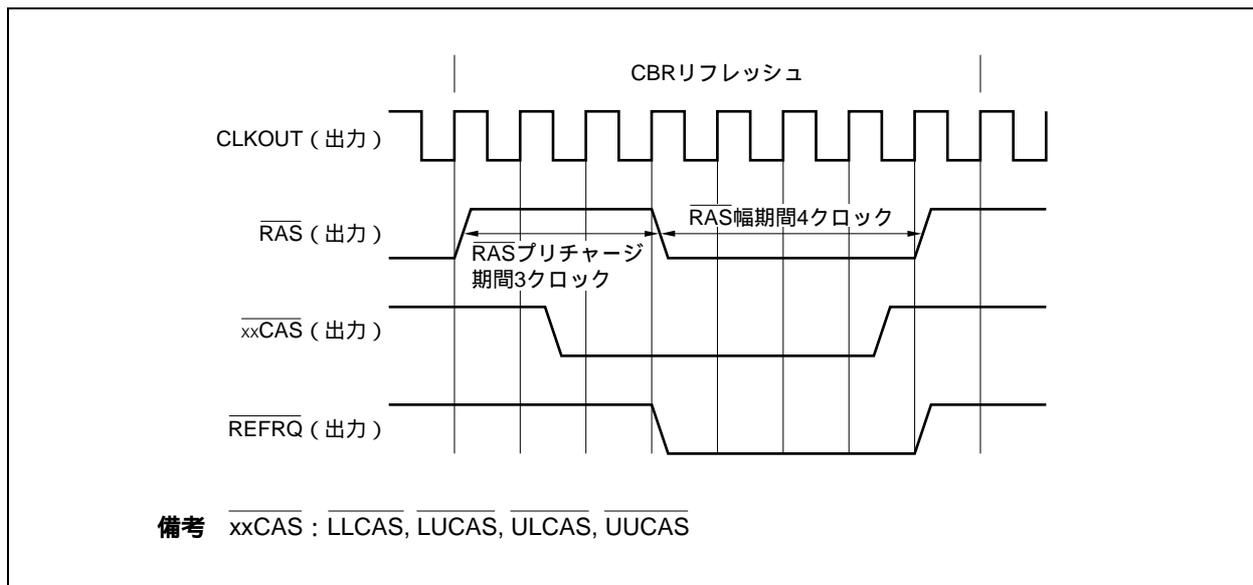
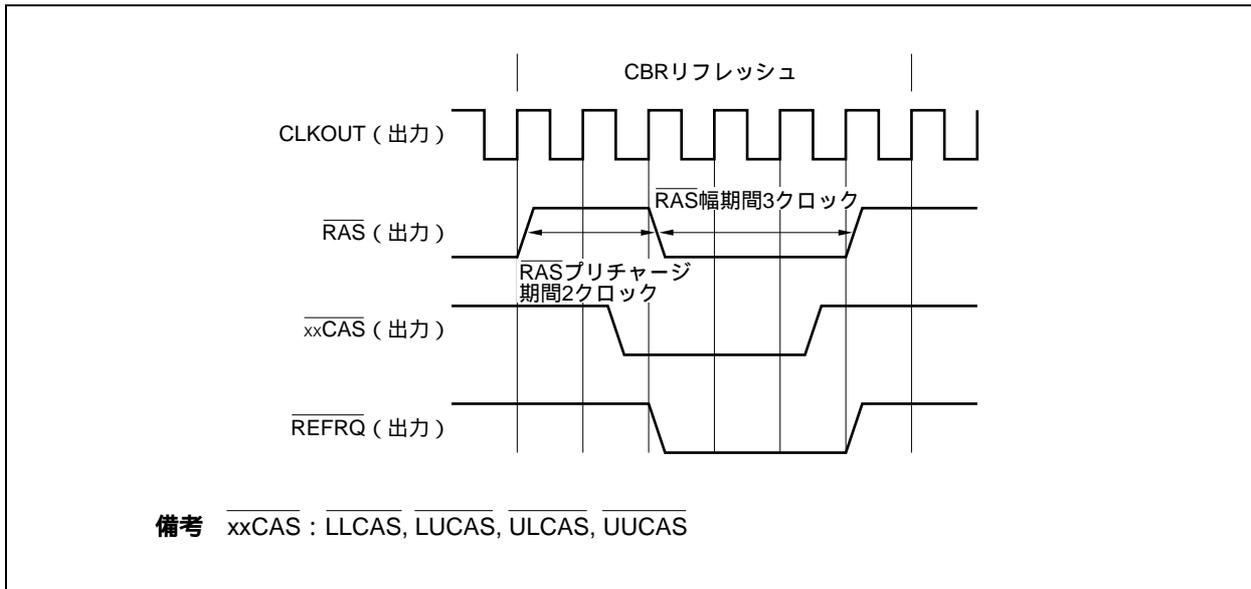


図5 - 18 CBRリフレッシュ・サイクル (RFW1,0 = 00, RP = 0のとき)



(2) CBRセルフ・リフレッシュ・サイクル

STOPモード時にCBRセルフ・リフレッシュ・サイクルを発生します。セルフ・リフレッシュ・サイクルは、RFCレジスタのRENビットが1のときSTBY命令を実行すると起動します。セルフ・リフレッシュの解除はRESET端子またはNMI端子で行います。

図5 - 19にCBRセルフ・リフレッシュのタイミングを示します。CBRセルフ・リフレッシュのあとのRASプリチャージ期間 (trps) は、最小5パス・クロックです。

図5 - 19 CBRリフレッシュ・サイクル (NMIによる解除, RP = 0のとき)

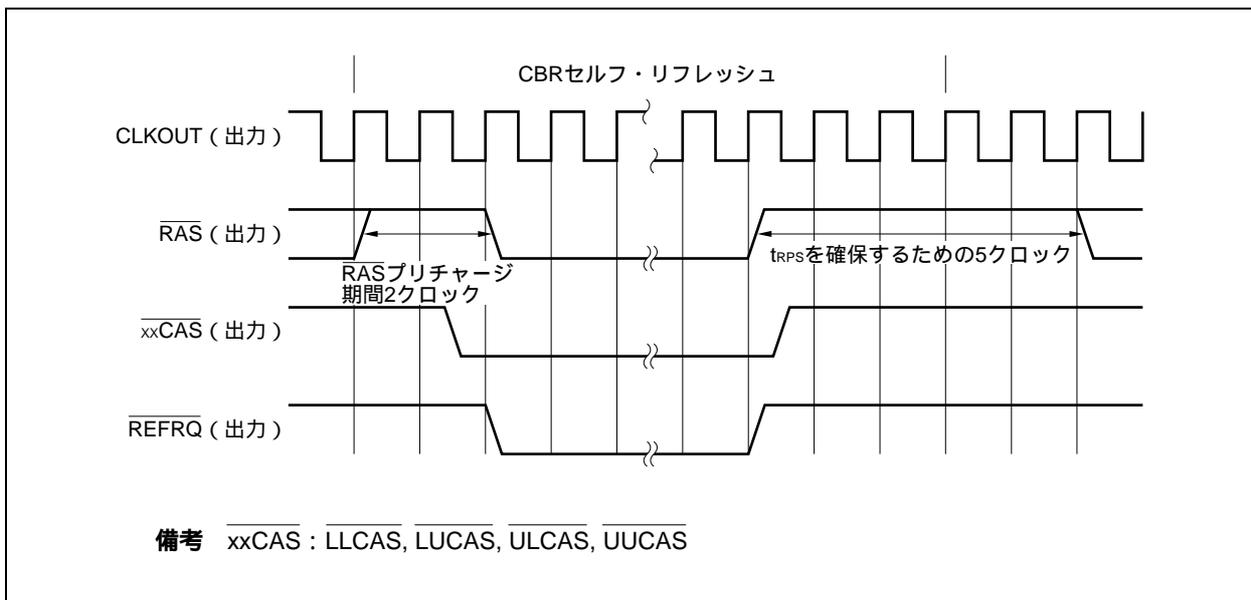
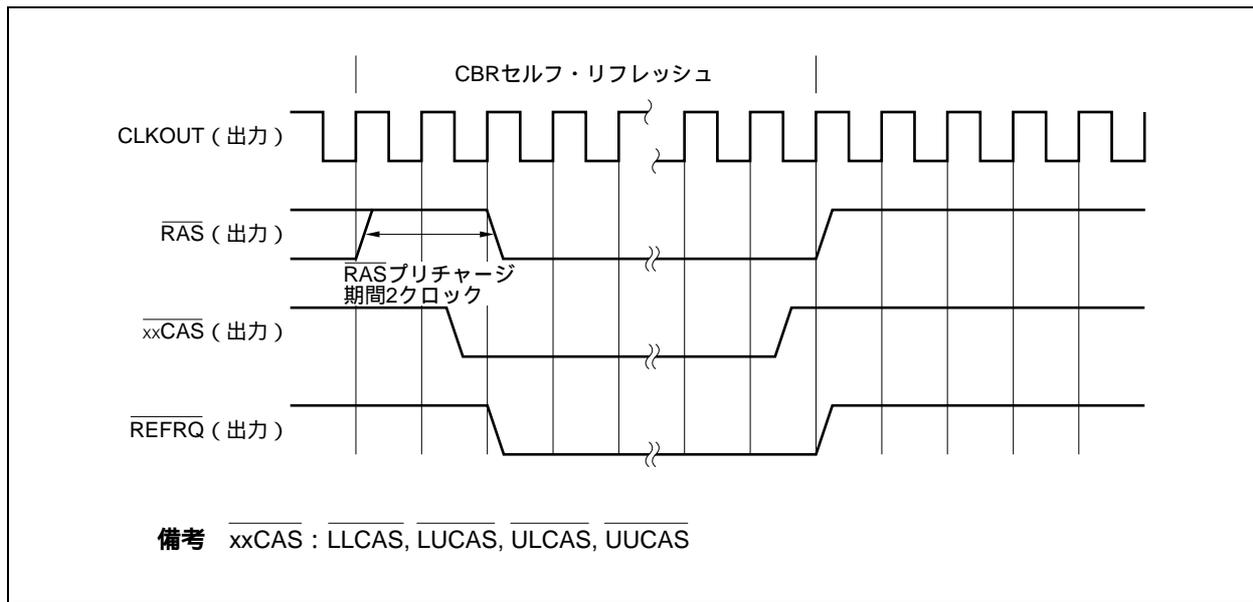


図5-20 CBRセルフ・リフレッシュ・サイクル (RESETによる解除, RP = 0のとき)



5.6 アイドル・ステート

PICレジスタのISnビットでnブロックに対するリード・サイクル後のアイドル・ステート数を設定します。(n = 0-7)。アイドル期間中のバス制御信号の値を次に示します。

表5-6 アイドル期間中のバス制御信号の値

信号名	信号値
A1-A23	前回のサイクルの値を保持
D0-D31	ハイ・インピーダンス
RAS	前回のサイクルの値を保持
$\overline{\text{UUMWR}}$, $\overline{\text{ULMWR}}$, $\overline{\text{LUMWR}}$, $\overline{\text{LLMWR}}$, $\overline{\text{MRD}}$, $\overline{\text{CS1-CS7}}$, $\overline{\text{IORD}}$, $\overline{\text{IOWR}}$, $\overline{\text{BCYST}}$, $\overline{\text{UUCAS}}$, $\overline{\text{ULCAS}}$, $\overline{\text{LUCAS}}$, $\overline{\text{LLCAS}}$, $\overline{\text{WE}}$, $\overline{\text{OE}}$	インアクティブ

5.7 バス・サイジング

V831は、32ビットから16ビットへのバス・サイジング機能があります。シングル・サイクルで32ビット・データを16ビット・データ・バスを介してアクセスする場合は、2回のアクセスを行います。バースト・サイクルで32ビット・データを16ビット・データ・バスを介してアクセスする場合は、8回連続してアクセスを行います。

次にデータ・バス幅16ビットでDRAMにシングル・サイクルでワード・アクセスした場合のタイミング（図5-21参照）、バースト・サイクルでアクセスした場合のタイミング（図5-22参照）、I/Oにワード・アクセスした場合のタイミング（図5-23参照）、SRAMにシングル・サイクルでワード・アクセスした場合のタイミング（図5-24参照）を示します。

図5-21 DRAMシングル・サイクルでバス・サイジングによって追加されたアクセス

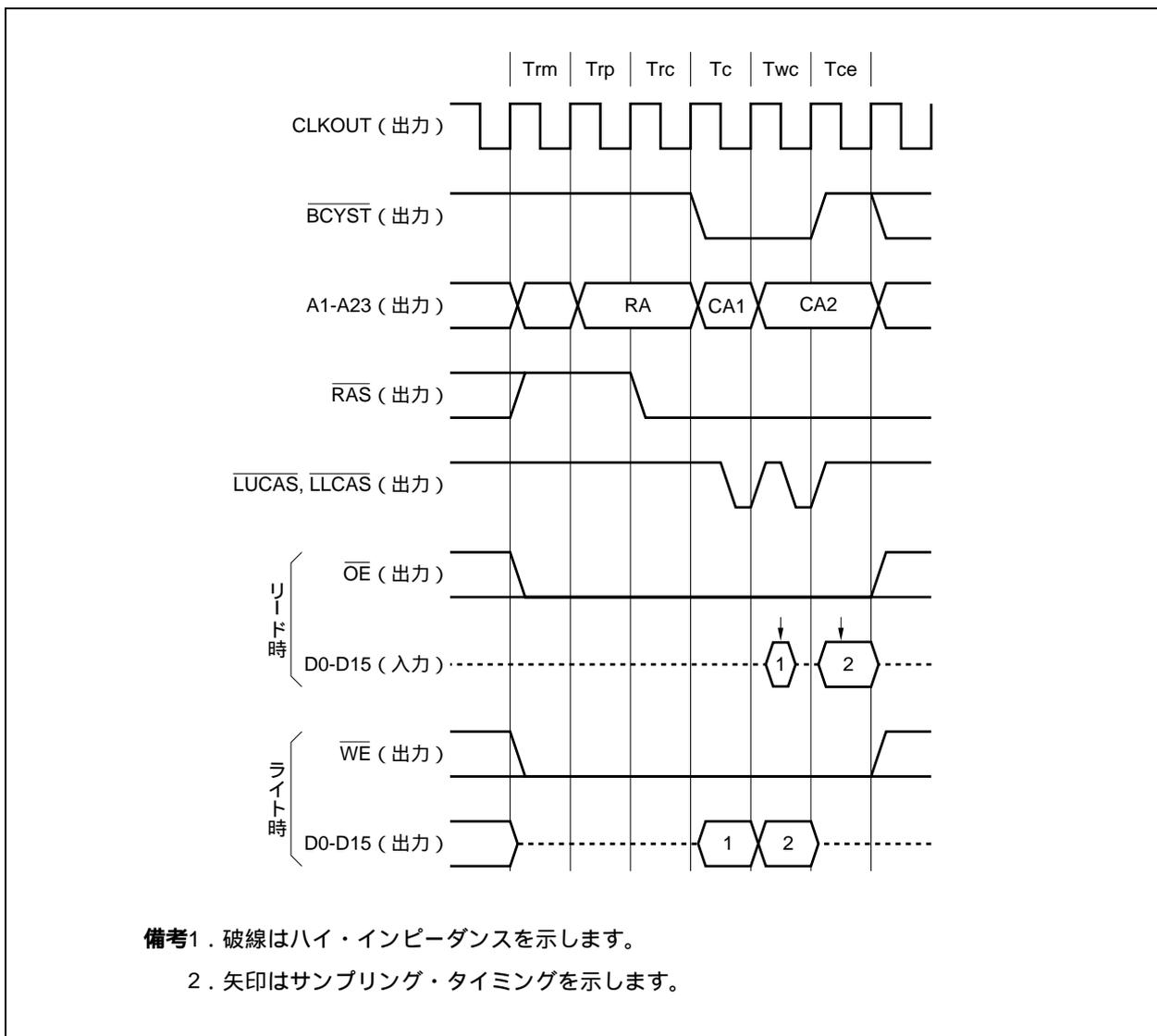


図5-22 DRAMバースト・サイクルでバス・サイジングによって追加されたアクセス

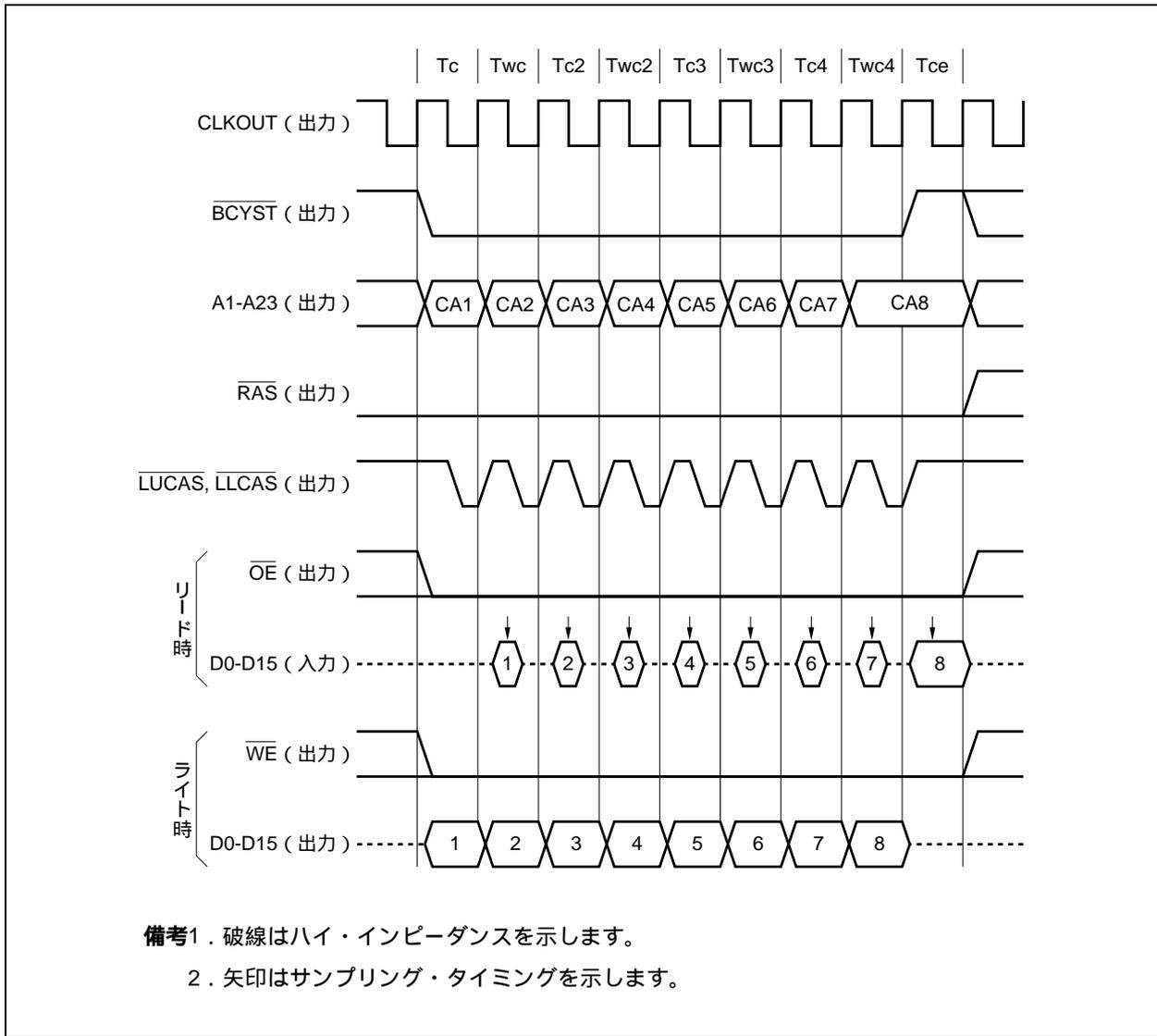


図5-23 I/Oサイクルでバス・サイジングによって追加されたアクセス

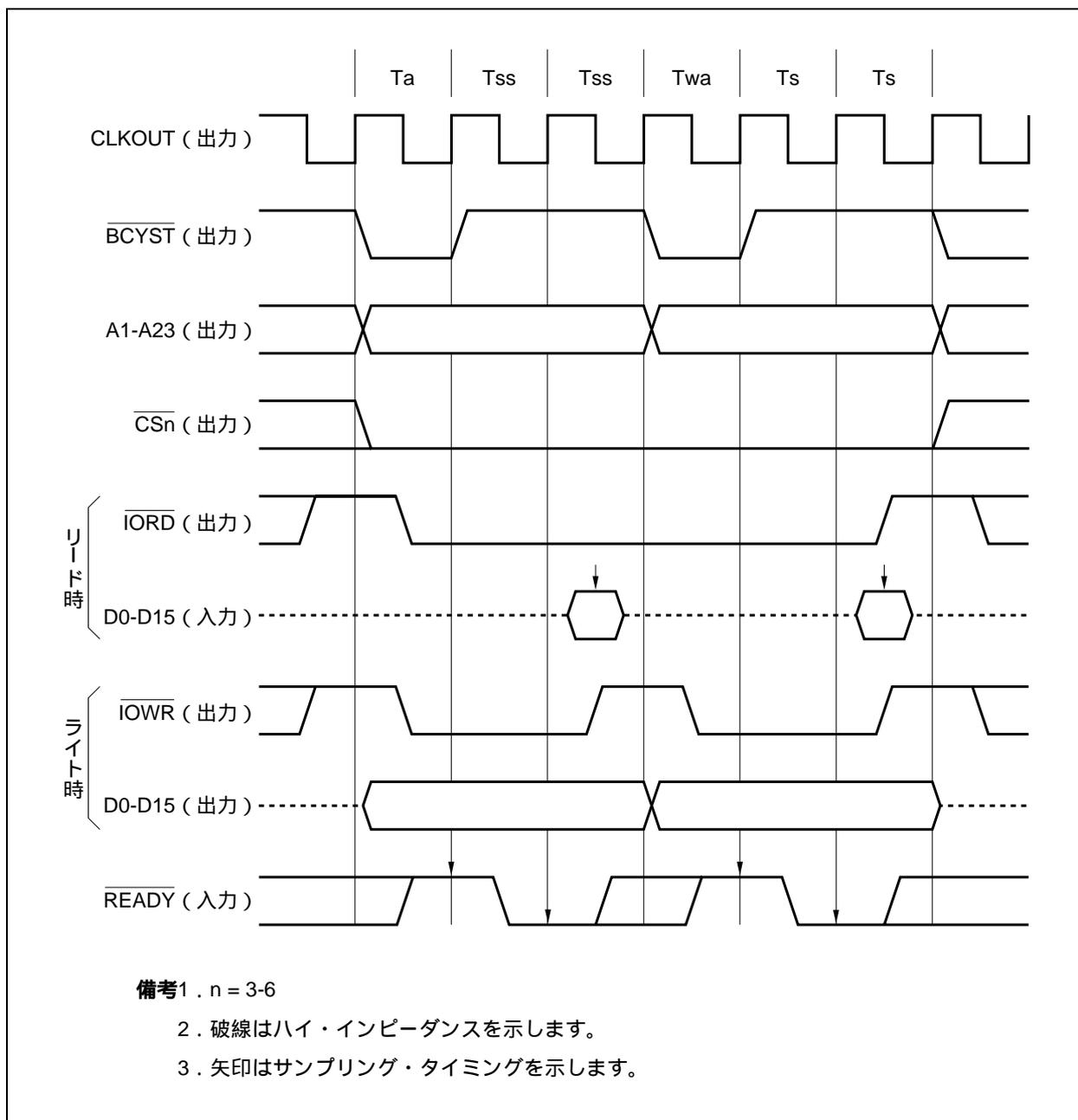
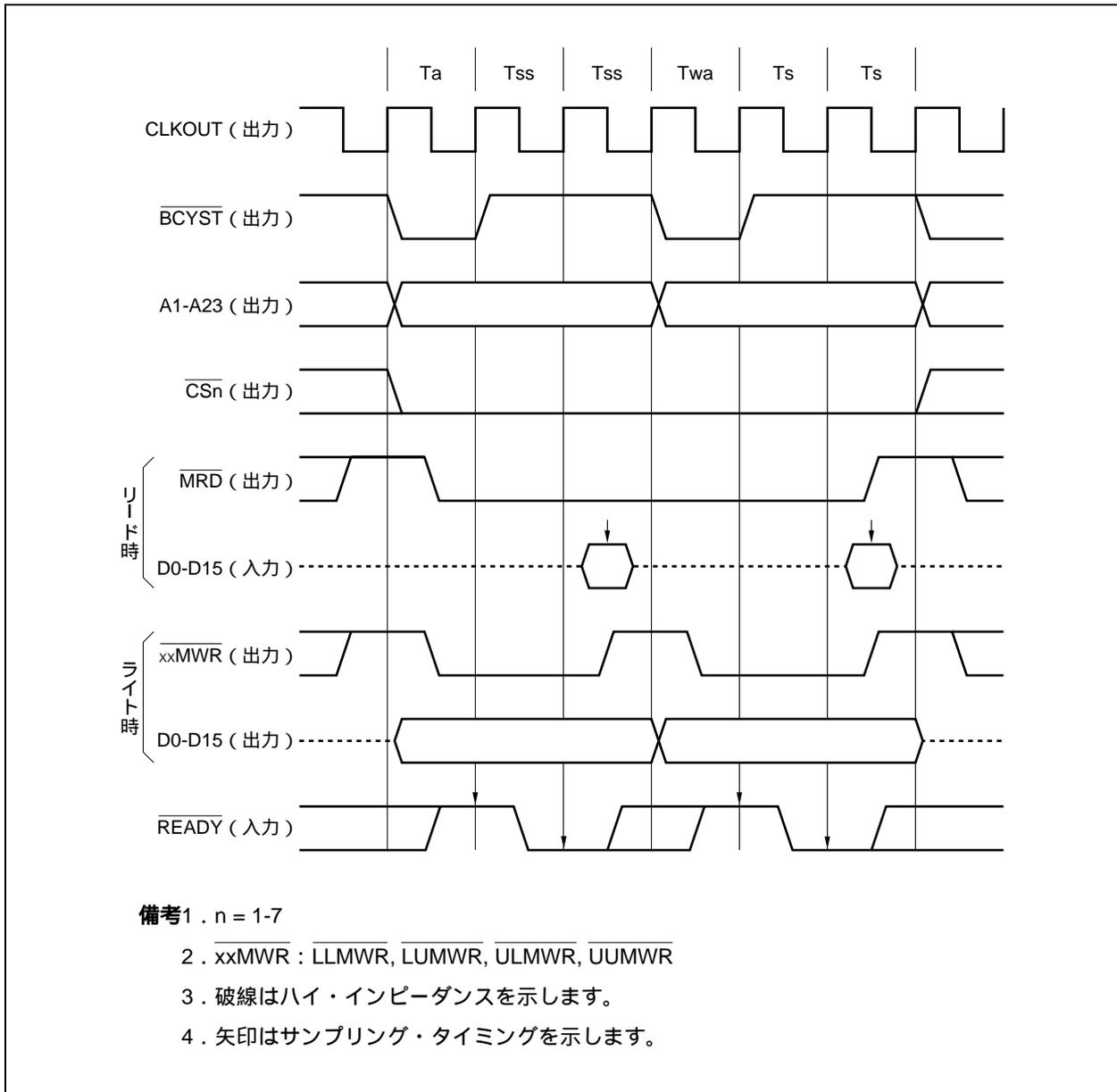


図5-24 SRAMシングル・サイクルでバス・サイジングによって追加されたアクセス

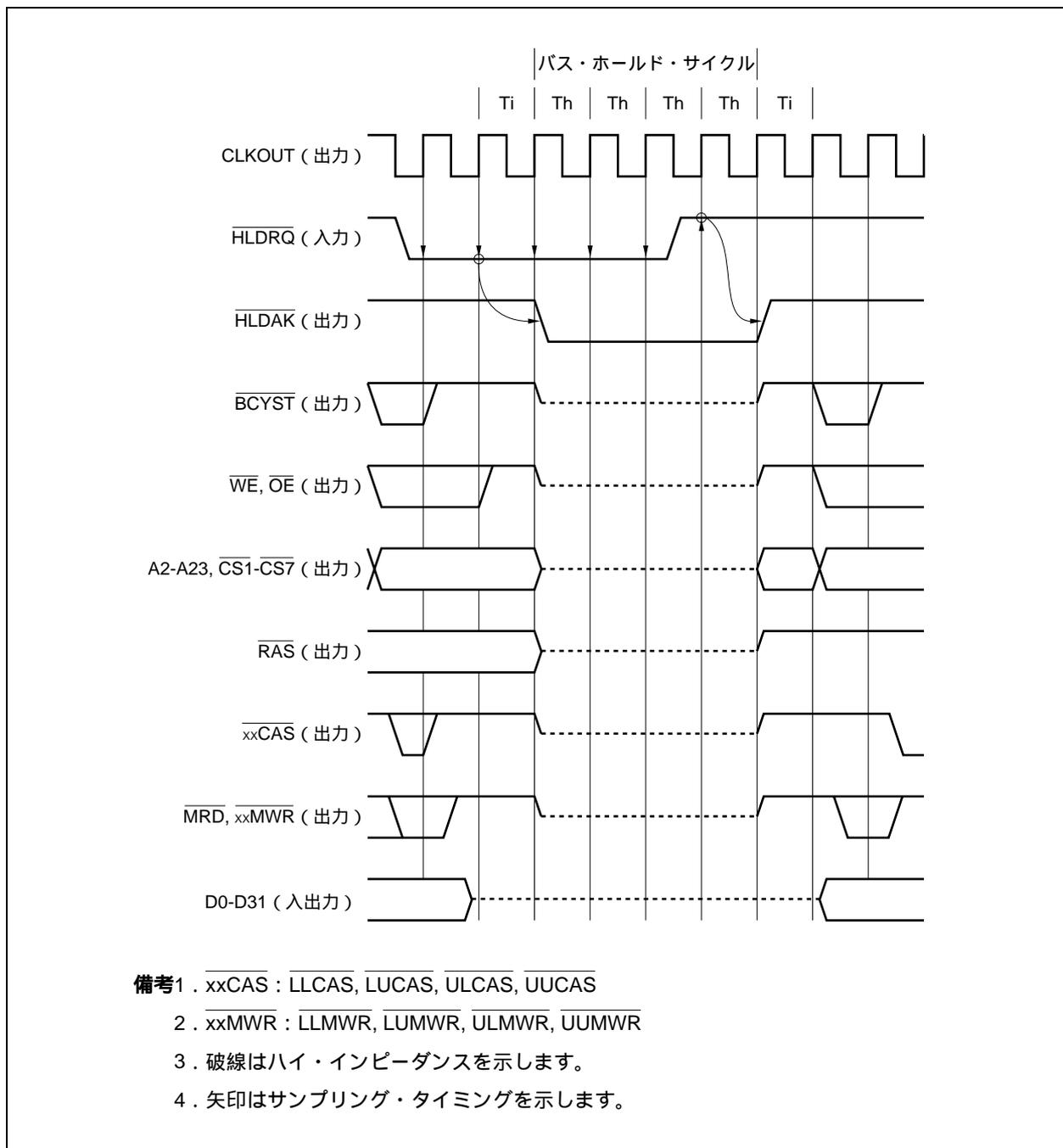


5.8 バス・ホールド・サイクル

外部バス・マスタは、 $\overline{\text{HLDRQ}}$ 信号をアクティブにすると、バス使用权をV831に要求できます。V831は、バスのアービトレーション後、 $\overline{\text{HLDAK}}$ 信号をアクティブにしてバスを開放します。 $\overline{\text{HLDRQ}}$ 信号がアクティブになると1バス・クロック・サイクル後にバス・ホールドを受け付け、アイドル・サイクル (T_i) になります。アイドル・サイクル (T_i) を1バス・クロック・サイクル実行後、バス・ホールド・サイクル (T_h) になり、バスをハイ・インピーダンスにします (図5 - 25参照)。

なお、バス・ホールド・サイクル後は、 $\overline{\text{RAS}}$ 信号はインアクティブになります。 $\overline{\text{NMI}}$ 信号によるバス・ホールドの解除は行いません。

図5 - 25 バス・ホールド・サイクル



5.9 バス・アービトレーション

V831は、内部の2つのバス・マスタ（CPUとDMA）およびDRAMのリフレッシュ、外部のバス・マスタとのバス・アービトレーションを行います。優先順位を次に示します。

バス・ロック > DRAMリフレッシュ=外部バス・マスタ > DMA > CPU

(1) バス・ロック > 外部バス・マスタ

CAXI命令によって起動されたバス・ロック・リード・サイクルとバス・ロック・ライト・サイクルの間は、外部バス・マスタはバス使用権をとることができません。

バス・ロック中であっても、DRAMリフレッシュは行われます。

(2) 外部バス・マスタ > DMA

DMA転送中は外部バス・マスタがバス使用権をとれます。外部バス・マスタがバスを使用中は、DMAはバス使用権をとれません。

一回のDMA転送のリード・サイクルとライト・サイクルの間は、外部バス・マスタはバスの使用権をとれません。

(3) DRAMリフレッシュ > DMA

DMAがデマンド転送中にDRAMのリフレッシュ要求が発生した場合、DRAMCのリフレッシュ要求が優先され、リフレッシュを実行します。

(4) DRAMリフレッシュ、外部バス・マスタ

外部バス・マスタがバスを使用中にDRAMのリフレッシュ要求が発生した場合、リフレッシュ要求を保留します。保留したリフレッシュ要求は、BCU内部のリフレッシュ要求キューに最大7回まで蓄えられます。外部バス・マスタがバスを開放したら、リフレッシュ要求キューに蓄えられた回数分のリフレッシュを行います。

5.10 ライト・バッファ動作

V831は、ライト動作を高速化するために内部にライト・バッファを4段内蔵しています。このためCPUはストア命令によるバス・サイクルの終了を待たずに次の命令を実行できます。すべてのライト・バッファが使用中の場合、ストア命令はライト・バッファが空くまで待ちます。また、内蔵RAMに対するストア命令はライト・バッファを使用しないため、待ち時間は発生しません。メモリ・アクセスの順序関係を保証するため、次の場合はライト・バッファの内容をすべて外部メモリに出力してから各処理を行います。

- ・ロード命令でデータ・キャッシュ・ミスした場合
- ・ロード命令で非キャッシュ領域にアクセスした場合

同様に、次に示す命令ではライト・バッファの内容をすべて外部メモリに出力されるまで待ち、それから命令実行を行います。

- ・ I/Oアクセス命令
- ・ ブロック転送命令 (BILD, BDLD, BIST, BDST)
- ・ HALT, STBY, CAXI命令

バス・ホールド動作はライト・バッファの動作とは関係なく機能します。したがって、外部メモリにストアしたデータなどをDMA転送した場合は、最後のストア命令実行後にI/Oライトなどを行い、ライト・バッファの内容をすべて出力してからDMA転送が実行されるようにしてください。

5.11 メモリ・マップトI/O

V831ではストア命令の実行タイミングとそのバス・ライト動作が時間的にずれる場合があるので、メモリ・マップトI/Oをクリティカルなタイミングで使用する場合は注意してください。

INT命令はゼロ拡張を行うことに対して、LD命令は符号拡張を行うという違いがあるので注意してください。

第6章 ウェイト制御機能

バス・コントロール・ユニット (BCU) は、7本の \overline{CS} 信号と1本のRAS信号に対応した8つのブロックを管理して、バス・サイクルの種類を選択、 \overline{CS} 信号の生成、データ・バス幅の選択、ウェイト制御およびアイドル・ステートの挿入を行います。

6.1 特 徴

I/O空間とメモリ空間をあわせて8ブロック制御可能

各ブロックのリニア・アドレス空間：16 Mバイト

バス・サイクルの選択機能

ブロック0 : EDO DRAM

ブロック1, 2 : SRAM (ROM)

ブロック3-6 : I/O, SRAM (ROM) 選択可能

ブロック7 : Page-ROM, SRAM (ROM) 選択可能

データ・バス幅選択機能

各ブロックごとにデータ・バス幅を32ビット/16ビット選択可能

ウェイト制御機能

ブロック0 : EDO DRAMアクセス・タイミング制御可能

ブロック1-4, 7 : 0-7ウェイト

ブロック5-6 : 0-15ウェイト

アイドル・ステート挿入機能

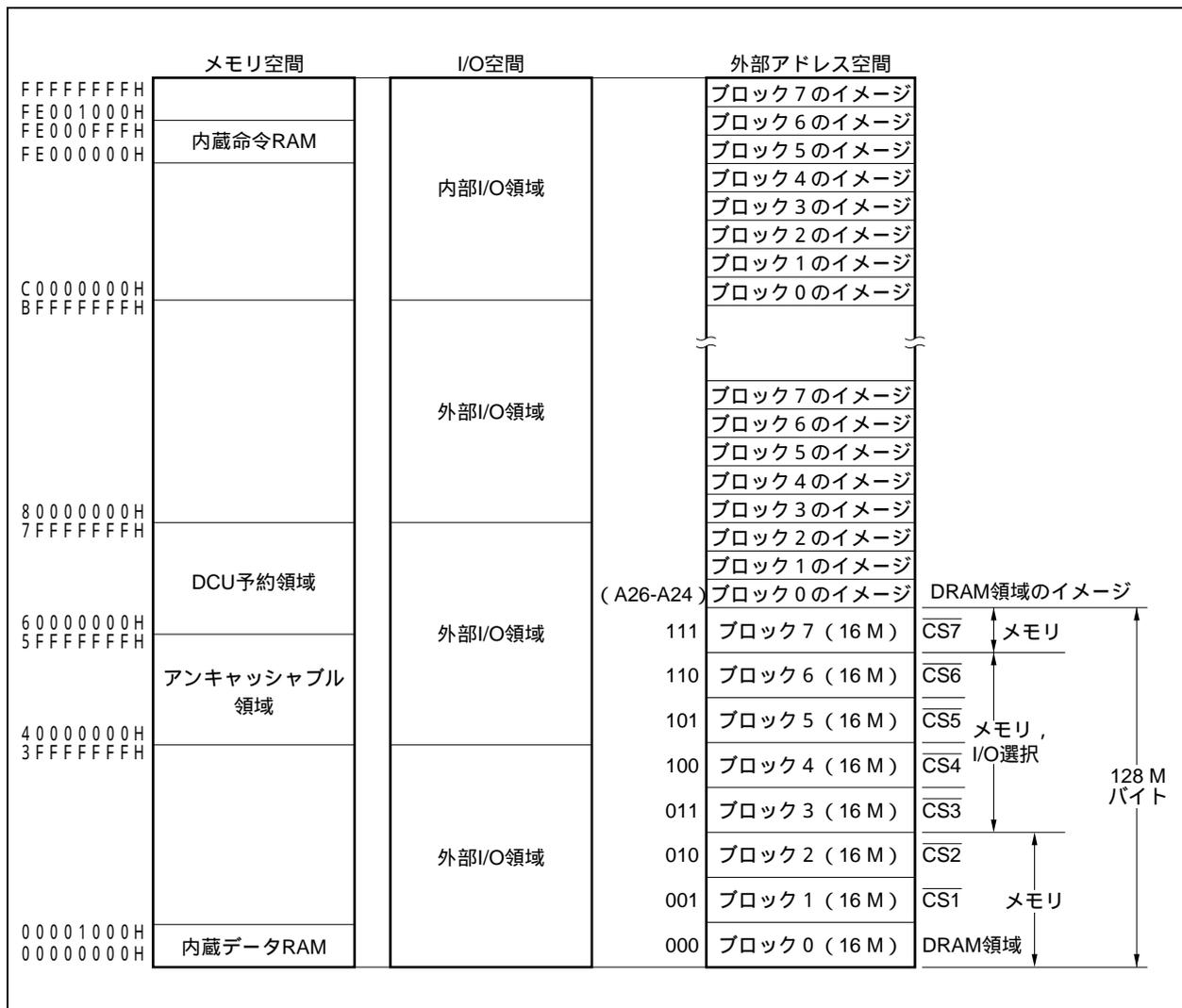
各ブロックごとに0-3ステート (バス・クロック)

6.2 アドレス空間とブロック

内部4 Gバイトのメモリ，I/O空間は，16 Mバイトごとのブロックに分割されます。各ブロックのリニア・アドレス空間は16 Mバイトです。なお，アンキャッシュابل領域（60000000H-7FFFFFFFH）は，DCUのディバグ・モニタ空間として使用するので外部メモリを接続しないでください。

- ★ 各ブロックのイメージはDCU予約領域と重なりますが，動作に問題はありません。ただし，DCU予約領域のアドレス（60000000-7FFFFFFF）には外部デバイスをマッピングしないでください。

図6-1 アドレス空間



6.3 ウェイト制御レジスタ

BCUは、バス・サイクル・タイプ・コントロール・レジスタ (BCTC) およびデータ・バス幅コントロール・レジスタ (DBC) と、ウェイト制御を行うプログラマブル・ウェイト・コントロール・レジスタ (PWC0, PWC1) を備えています。

6.3.1 バス・サイクル・タイプ・コントロール・レジスタ (BCTC)

ブロック3-7に対するバス・サイクルの種類を設定します。ただし、ブロック0はDRAMサイクル、ブロック1-2はSRAM/ROMサイクル固定です。8ビット単位でリード/ライトできます。

図6-2 バス・サイクル・タイプ・コントロール・レジスタ (BCTC)

アドレス : C0000010H		
		7 6 5 4 3 2 1 0
BCTC		CT7 CT6 CT5 CT4 CT3 0 0 0
初期値 :		0 0 0 0 0 0 0 0
R/W :		R/W R/W R/W R/W R/W R R R
ビット	ビット名	説明
7	CT7	Cycle Type7 メモリ・ブロック7にアクセスがあると、 $\overline{CS7}$ 信号を出力します。そのときに起動するサイクルを指定します。 0 : SRAM (ROM) サイクル 1 : Page-ROMサイクル
6-3	CT6-CT3	Cycle Type6-3 $\overline{CS6}$ - $\overline{CS3}$ 信号をメモリとI/Oのどちらかに出力すると、起動するサイクルの指定を行います。 0 : SRAM (ROM) サイクル 1 : I/Oサイクル

6.3.2 データ・バス幅コントロール・レジスタ (DBC)

ブロック0-6に対するデータ・バス幅を設定します。8ビット単位でリード/ライトできます。ただし、通常ブートROMが置かれるブロック7 (CS7) のバス幅は、BT16B端子の値で設定します。

図6-3 データ・バス幅コントロール・レジスタ (DBC)

アドレス : C0000012H

	7	6	5	4	3	2	1	0
DBC	BW7	BW6	BW5	BW4	BW3	BW2	BW1	BW0

初期値 :

BT16B	0	0	0	0	0	0	0	0
-------	---	---	---	---	---	---	---	---

R/W : R R/W R/W R/W R/W R/W R/W R/W

ビット	ビット名	説明
7	BW7	Bus Width BT16B端子の値を読み出せます。なお、BT16B端子の値で設定されるブロック7 (CS7) のデータ・バス幅を示します。 0 : 32ビット・バス幅 1 : 16ビット・バス幅
6-0	BW6-BW0	Bus Width ブロック6-0のデータ・バス幅を指定します。 0 : 32ビット・バス幅 1 : 16ビット・バス幅

6.3.3 プログラマブル・ウェイト・コントロール・レジスタ0 (PWC0)

ブロック1-3に対するアクセスのウェイト数を設定します。16ビット単位でリード/ライトできます。それぞれ、7ステートまでのウェイト・ステートを挿入できます。

ブロック0はDRAM領域固定です。DRAMアクセスのウェイト数は、DRAMコンフィギュレーション・レジスタ (DRC) で指定します。

図6-4 プログラマブル・ウェイト・コントロール・レジスタ0 (PWC0)

アドレス : C0000014H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWC0	0	WS3			0	WS2			0	WS1			0	0	0	0

初期値 :

0	1	1	1	0	1	1	1	0	1	1	1	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W : R R/W R/W R/W R R/W R/W R/W R R/W R/W R/W R R R R

ビット	ビット名	説 明																				
14-12	WS3	Wait States3 ブロック3 ($\overline{CS3}$) にアクセス時のウェイト数を設定します。0ウェイトから7ウェイトまで設定できます。 <table border="1" style="margin: 10px auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;">WS3</th> <th style="width: 15%;">ウェイト数</th> <th style="width: 10%;">WS3</th> <th style="width: 15%;">ウェイト数</th> </tr> </thead> <tbody> <tr> <td>0 0 0</td> <td>0</td> <td>1 0 0</td> <td>4</td> </tr> <tr> <td>0 0 1</td> <td>1</td> <td>1 0 1</td> <td>5</td> </tr> <tr> <td>0 1 0</td> <td>2</td> <td>1 1 0</td> <td>6</td> </tr> <tr> <td>0 1 1</td> <td>3</td> <td>1 1 1</td> <td>7</td> </tr> </tbody> </table>	WS3	ウェイト数	WS3	ウェイト数	0 0 0	0	1 0 0	4	0 0 1	1	1 0 1	5	0 1 0	2	1 1 0	6	0 1 1	3	1 1 1	7
WS3	ウェイト数	WS3	ウェイト数																			
0 0 0	0	1 0 0	4																			
0 0 1	1	1 0 1	5																			
0 1 0	2	1 1 0	6																			
0 1 1	3	1 1 1	7																			
10-8	WS2	Wait States2 ブロック2 ($\overline{CS2}$) にアクセス時のウェイト数を設定します。0ウェイトから7ウェイトまで設定できます。ウェイト数の設定方法は、WS3と同じです。																				
6-4	WS1	Wait States1 ブロック1 ($\overline{CS1}$) にアクセス時のウェイト数を設定します。0ウェイトから7ウェイトまで設定できます。ウェイト数の設定方法は、WS3と同じです。																				

6.3.4 プログラマブル・ウェイト・コントロール・レジスタ1 (PWC1)

ブロック4-7に対するアクセスのウェイト数を設定します。16ビット単位でリード/ライトできます。ブロック4, 7は, 7ステートまでのウェイト・ステートを挿入できます。ブロック5, 6は, 15ステートまでのウェイト・ステートを挿入できます。

図6-5 プログラマブル・ウェイト・コントロール・レジスタ1 (PWC1)

アドレス : C0000016H																	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PWC1	0	WS7			WS6				WS5				0	WS4			
初期値 :	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	
R/W :	R	R/W	R	R/W	R/W	R/W											

ビット	ビット名	説明																																				
14-12	WS7	Wait States7 ブロック7 ($\overline{CS7}$) にアクセス時のウェイト数を設定します。0ウェイトから7ウェイトまで設定できます。 <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">WS7</th> <th style="width: 15%;">ウェイト数</th> <th style="width: 10%;">WS7</th> <th style="width: 15%;">ウェイト数</th> </tr> </thead> <tbody> <tr> <td>0 0 0</td> <td style="text-align: center;">0</td> <td>1 0 0</td> <td style="text-align: center;">4</td> </tr> <tr> <td>0 0 1</td> <td style="text-align: center;">1</td> <td>1 0 1</td> <td style="text-align: center;">5</td> </tr> <tr> <td>0 1 0</td> <td style="text-align: center;">2</td> <td>1 1 0</td> <td style="text-align: center;">6</td> </tr> <tr> <td>0 1 1</td> <td style="text-align: center;">3</td> <td>1 1 1</td> <td style="text-align: center;">7</td> </tr> </tbody> </table> BCTCのCT7ビットでPage-ROMサイクルを選択した場合は, シングル・サイクルのウェイト数およびバースト・サイクルのoff-pageアクセス時のウェイト数になります。 on-pageアクセス時のウェイト数はPRCレジスタのPWSビットで設定できます。	WS7	ウェイト数	WS7	ウェイト数	0 0 0	0	1 0 0	4	0 0 1	1	1 0 1	5	0 1 0	2	1 1 0	6	0 1 1	3	1 1 1	7																
WS7	ウェイト数	WS7	ウェイト数																																			
0 0 0	0	1 0 0	4																																			
0 0 1	1	1 0 1	5																																			
0 1 0	2	1 1 0	6																																			
0 1 1	3	1 1 1	7																																			
11-8	WS6	Wait States6 ブロック6 ($\overline{CS6}$) にアクセス時のウェイト数を設定します。0ウェイトから15ウェイトまで設定できます。 <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">WS6</th> <th style="width: 15%;">ウェイト数</th> <th style="width: 10%;">WS6</th> <th style="width: 15%;">ウェイト数</th> </tr> </thead> <tbody> <tr> <td>0 0 0 0</td> <td style="text-align: center;">0</td> <td>1 0 0 0</td> <td style="text-align: center;">8</td> </tr> <tr> <td>0 0 0 1</td> <td style="text-align: center;">1</td> <td>1 0 0 1</td> <td style="text-align: center;">9</td> </tr> <tr> <td>0 0 1 0</td> <td style="text-align: center;">2</td> <td>1 0 1 0</td> <td style="text-align: center;">10</td> </tr> <tr> <td>0 0 1 1</td> <td style="text-align: center;">3</td> <td>1 0 1 1</td> <td style="text-align: center;">11</td> </tr> <tr> <td>0 1 0 0</td> <td style="text-align: center;">4</td> <td>1 1 0 0</td> <td style="text-align: center;">12</td> </tr> <tr> <td>0 1 0 1</td> <td style="text-align: center;">5</td> <td>1 1 0 1</td> <td style="text-align: center;">13</td> </tr> <tr> <td>0 1 1 0</td> <td style="text-align: center;">6</td> <td>1 1 1 0</td> <td style="text-align: center;">14</td> </tr> <tr> <td>0 1 1 1</td> <td style="text-align: center;">7</td> <td>1 1 1 1</td> <td style="text-align: center;">15</td> </tr> </tbody> </table>	WS6	ウェイト数	WS6	ウェイト数	0 0 0 0	0	1 0 0 0	8	0 0 0 1	1	1 0 0 1	9	0 0 1 0	2	1 0 1 0	10	0 0 1 1	3	1 0 1 1	11	0 1 0 0	4	1 1 0 0	12	0 1 0 1	5	1 1 0 1	13	0 1 1 0	6	1 1 1 0	14	0 1 1 1	7	1 1 1 1	15
WS6	ウェイト数	WS6	ウェイト数																																			
0 0 0 0	0	1 0 0 0	8																																			
0 0 0 1	1	1 0 0 1	9																																			
0 0 1 0	2	1 0 1 0	10																																			
0 0 1 1	3	1 0 1 1	11																																			
0 1 0 0	4	1 1 0 0	12																																			
0 1 0 1	5	1 1 0 1	13																																			
0 1 1 0	6	1 1 1 0	14																																			
0 1 1 1	7	1 1 1 1	15																																			
7-4	WS5	Wait States5 ブロック5 ($\overline{CS5}$) にアクセス時のウェイト数を設定します。0ウェイトから15ウェイトまで設定できます。ウェイト数の設定方法は, WS6と同じです。																																				
2-0	WS4	Wait States4 ブロック4 ($\overline{CS4}$) にアクセス時のウェイト数を設定します。0ウェイトから7ウェイトまで設定できます。ウェイト数の設定方法は, WS7と同じです。																																				

6.3.5 プログラマブル・アイドル・コントロール・レジスタ (PIC)

ブロック0-7に対するリード・アクセス後に挿入するアイドル・ステート数を設定します。16ビット単位でリード/ライトできます。それぞれ3ステートまでのアイドル・ステートを挿入できます。

V830CPUコアからのアクセスでは、リード・サイクルのあと必ず1ステートのアイドル・ステートが挿入されます。このアイドル・ステートは、PICレジスタのISnの設定値にカウントされます。つまり、ISn (n = 0-7) が0または1のとき、1ステートのアイドル・ステートが挿入されます。

DMAの2サイクル転送時も、PICレジスタの設定は有効になります。

図6-6 プログラマブル・アイドル・コントロール・レジスタ (PIC)

アドレス: C0000018H																		
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
PIC	IS7		IS6		IS5		IS4		IS3		IS2		IS1		IS0			
初期値:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W		
ビット	ビット名		説明															
15-0	IS7-IS0		Idle State7-0 アイドル・ステート数を指定します。IS7-IS0は、ブロック7-0に対応しています。CSn信号に対応するブロックにリード・アクセスがあると、リード・アクセス後にISn (n = 0-7) で設定される数のアイドル・ステートを自動挿入します。															
			ISn		アイドル・ステート挿入数													
			0	0	0													
			0	1	1													
			1	0	2													
			1	1	3													

★ 6.4 READY端子によるウェイト制御

V831は、I/O, SRAM, ROM領域に対し、PWC0, PWC1レジスタとREADY端子により、ウェイトを制御できません。

6.4.1 READY端子入力のサンプリング・タイミング

- ・ I/O領域：Ts状態の立ち上がりでサンプリングします。プログラマブル・ウェイト回数が終了していない、または、 $\overline{\text{READY}}$ 端子がアクティブになっていない場合（ウェイトが挿入される場合）は、Tsサイクルを繰り返します。
- ・ SRAM, ROMのシングル・サイクル：Ts状態の立ち上がりでサンプリングします。プログラマブル・ウェイト回数が終了していない、または、 $\overline{\text{READY}}$ 端子がアクティブになっていない場合は、Tsサイクルを繰り返します。
- ・ SRAM, ROMのバースト・サイクル：各Tb状態の立ち上がりとTs状態の立ち上がりで、 $\overline{\text{READY}}$ 端子のサンプリングを行います。ウェイト挿入時はTbサイクルまたはTsサイクルを繰り返します。
- ・ Page-ROMサイクル：Tb状態の立ち上がりと、Ts状態の立ち上がりで $\overline{\text{READY}}$ 端子信号のサンプリングをします。ウェイト挿入時はTbサイクルまたはTsサイクルを繰り返します。プログラマブル・ウェイトの設定はon-page時と、off-page時で異なる挿入ウェイト数を設定することが可能です。

6.4.2 READY端子とプログラマブル・ウェイト両方を使用する場合

PWC0, PWC1レジスタで設定したウェイト数と $\overline{\text{READY}}$ 端子入力によるウェイト数は論理和の形を取るため、どちらが多い方のウェイト数が挿入されます。

したがって、 $\overline{\text{READY}}$ 端子によりウェイト制御を行う場合、上述のサンプリング・タイミングに $\overline{\text{READY}}$ 端子をインアクティブに制御していても、プログラマブル・ウェイトによるウェイト挿入回数が終了するまで、ウェイトは解除されません。また、プログラマブル・ウェイトが解除される前に、 $\overline{\text{READY}}$ 端子をインアクティブに制御しても、プログラマブル・ウェイト後のTs状態で $\overline{\text{READY}}$ 端子がインアクティブになっていなければウェイトは解除されません。したがって、誤使用を避けるため、 $\overline{\text{READY}}$ 端子入力によるウェイトを使用する場合は、PWC0, PWC1レジスタの対応するブロックのウェイト数を0に設定して使用することを推奨します。

- ・ DRAMサイクル： $\overline{\text{READY}}$ 端子入力によるウェイト制御を行うことはできません。プログラマブルにプリチャージ期間、RAS-CAS遅延時間、CASサイクル期間を設定できます。詳細は5.5 DRAMサイクルおよび第7章 メモリ・アクセス制御機能を参照してください。

第7章 メモリ・アクセス制御機能

この章では、DRAM制御機能とPage-ROM制御機能を説明します。

V831のBCUIは、EDO DRAM, Page-ROM, SRAM (ROM) をダイレクトに接続できます。

EDO DRAMへのアクセスは、アドレス・バス、データ・バス、4本の $\overline{\text{xxCAS}}$, $\overline{\text{RAS}}$, $\overline{\text{OE}}$, $\overline{\text{WE}}$ 信号を用いて行います。DRAMへのアドレスは、ロウ・アドレス、カラム・アドレスをマルチプレクスしてアドレス端子から出力します。

Page-ROMへのページ・アクセスは、バースト・アクセス時に有効になります。ページ・サイズは、8バイトまたは16バイトから選択できます。

SRAM (ROM) へのアクセスは、アドレス・バス、データ・バス、4本の $\overline{\text{xxMWR}}$, $\overline{\text{MRD}}$, $\overline{\text{CS}}$ 信号を用いて行います。

- 備考1. $\overline{\text{xxCAS}}$: $\overline{\text{UUCAS}}$, $\overline{\text{ULCAS}}$, $\overline{\text{LUCAS}}$, $\overline{\text{LLCAS}}$
 2. $\overline{\text{xxMWR}}$: $\overline{\text{UUMWR}}$, $\overline{\text{ULMWR}}$, $\overline{\text{LUMWR}}$, $\overline{\text{LLMWR}}$

7.1 特 徴

DRAM制御機能

- ・ $\overline{\text{RAS}}$, $\overline{\text{LLCAS}}$, $\overline{\text{LUCAS}}$, $\overline{\text{ULCAS}}$, $\overline{\text{UUCAS}}$, $\overline{\text{REFRQ}}$, $\overline{\text{OE}}$, $\overline{\text{WE}}$ 信号の生成
- ・ アドレス・マルチプレクス : 8, 9, 10ビット
- ・ DRAMアクセスのタイミング制御

$\overline{\text{CAS}}$ アクセス期間 : 1バス・クロック / 2バス・クロック選択可能

$\overline{\text{RAS}}$ - $\overline{\text{CAS}}$ 遅延期間 : 1.5バス・クロック / 2.5バス・クロック選択可能

$\overline{\text{RAS}}$ プリチャージ期間 : 2バス・クロック / 3バス・クロック選択可能

- ・ CBRリフレッシュ, CBRセルフ・リフレッシュ機能

Page-ROM制御機能

- ・ ページ・サイズ : 8バイト / 16バイト
- ・ ページ・アクセス時のウエイト制御 : 0ウエイト / 1ウエイト

7.2 DRAM制御機能

BCUIは、 $\overline{\text{RAS}}$, $\overline{\text{LLCAS}}$, $\overline{\text{LUCAS}}$, $\overline{\text{ULCAS}}$, $\overline{\text{UUCAS}}$, $\overline{\text{REFRQ}}$, $\overline{\text{OE}}$, $\overline{\text{WE}}$ 信号の生成とDRAMへのアクセスを制御します。DRAMへのアドレスは、DRAMのロウ・アドレス、カラム・アドレスをマルチプレクスしてアドレス端子から出力します。

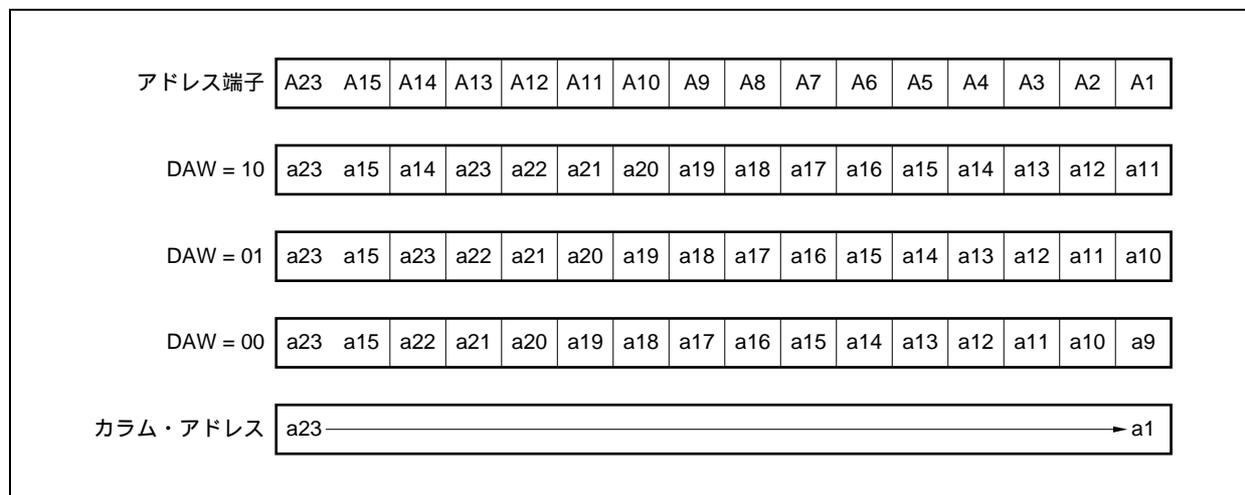
接続するDRAMは、 $\times 8$ ビット以上でハイパー・ページ・モード (EDO) を備えていることを前提とします。

リフレッシュは、 $\overline{\text{CAS}}$ ビフォー $\overline{\text{RAS}}$ (CBR) 方式で、リフレッシュ周期を任意に設定できます。また、STOPモード時には、CBRセルフ・リフレッシュを行います。

7.2.1 アドレス・マルチプレクス機能

DRAMコンフィギュレーション・レジスタ（DRC）のDAWビットの値により，DRAMサイクルでのロウ・アドレス，カラム・アドレス出力は，図7-1のようにアドレスがマルチプレクスします。図7-1のa1-a23はCPUから出力されたアドレスを，A1-A23はV831のアドレス端子を示しています。

図7-1 ロウ・アドレス，カラム・アドレスの出力



7.2.2 on-page/off-pageの判断

DRAMコンフィギュレーション・レジスタ（DRC）のHPAEビットが1でページ・アクセスが許可されている状態で $\overline{\text{RAS}}$ 信号がアクティブの場合，起動するDRAMアクセスが前回のDRAMアクセスと同一ページ内かどうかの判断を行います。表7-1に比較するアドレスとアドレス・シフト量の関係を示します。

表7-1 on-page/off-page判断で比較するアドレス

アドレス・シフト量	データ・バス幅	
	16ビット	32ビット
8	a23-a9	a23-a10
9	a23-a10	a23-a11
10	a23-a11	a23-a12

7.2.3 DRAMコンフィギュレーション・レジスタ（DRC）

DRAMアクセス時のアドレス・マルチプレクス幅と， $\overline{\text{RAS}}$ ， $\overline{\text{CAS}}$ 信号の出力タイミングを設定します。8ビット単位でリード/ライトできます。

図7-2 DRAMコンフィギュレーション・レジスタ (DRC)

アドレス : C0000020H

	7	6	5	4	3	2	1	0
DRC	HPAE	RP	RCD	CRWT	0	0	0	DAW

初期値 :	0	1	1	1	1	0	0	1
-------	---	---	---	---	---	---	---	---

R/W : R/W R/W R/W R/W R/W R R/W R/W

ビット	ビット名	説明																	
7	HPAE	Hyper Page Mode Enable DRAMのハイパー・ページ・モードに対応したページ・アクセスの起動を制御します。ただし、パースト・サイクル時の2回目以降のDRAMアクセスは、ページ・アクセスを禁止できません。 0 : 起動禁止 (常にoff-pageアクセスを行います。) 1 : 起動許可 (on-pageアクセスの起動を許可します。)																	
6	RP	RAS Precharge RASのプリチャージ期間を設定します。 0 : 2バス・クロック 1 : 3バス・クロック																	
5	RCD	RAS-CAS Delay RAS-CAS遅延期間を設定します。 0 : 1.5バス・クロック 1 : 2.5バス・クロック																	
4,3	CRWT	CAS Read Write Timing DRAMリード/ライト時のCASサイクル期間を設定します。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th rowspan="2">CRWT</th> <th colspan="2">CASサイクル期間</th> </tr> <tr> <th>リード・サイクル時</th> <th>ライト・サイクル時</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td colspan="2">1バス・クロック</td> </tr> <tr> <td>0 1</td> <td colspan="2">設定禁止</td> </tr> <tr> <td>1 0</td> <td>2バス・クロック</td> <td>1バス・クロック</td> </tr> <tr> <td>1 1</td> <td colspan="2">2バス・クロック</td> </tr> </tbody> </table>	CRWT	CASサイクル期間		リード・サイクル時	ライト・サイクル時	0 0	1バス・クロック		0 1	設定禁止		1 0	2バス・クロック	1バス・クロック	1 1	2バス・クロック	
CRWT	CASサイクル期間																		
	リード・サイクル時	ライト・サイクル時																	
0 0	1バス・クロック																		
0 1	設定禁止																		
1 0	2バス・クロック	1バス・クロック																	
1 1	2バス・クロック																		
1,0	DAW	DRAM Address Width DRAMサイクル中のカラム・アドレス時のアドレス幅を設定します。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>DAW</th> <th>アドレス幅</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td>8ビット</td> </tr> <tr> <td>0 1</td> <td>9ビット</td> </tr> <tr> <td>1 0</td> <td>10ビット</td> </tr> <tr> <td>1 1</td> <td>設定禁止</td> </tr> </tbody> </table>	DAW	アドレス幅	0 0	8ビット	0 1	9ビット	1 0	10ビット	1 1	設定禁止							
DAW	アドレス幅																		
0 0	8ビット																		
0 1	9ビット																		
1 0	10ビット																		
1 1	設定禁止																		

7.2.4 リフレッシュ機能

BCUは、外部DRAMのリフレッシュ動作に必要な分散型のCBRリフレッシュ・サイクルを自動的に発生できます。リフレッシュの許可/禁止とリフレッシュ間隔は、リフレッシュ・コントロール・レジスタ (RFC) で設定します。

BCUは、リフレッシュ要求を最大7回まで蓄えるリフレッシュ要求キューを備えています。

(1) リフレッシュ要求キュー

BCUは最大7回までのリフレッシュ要求を蓄えるリフレッシュ要求キューを備えています。バスが解放されると、リフレッシュ要求キューの内容が“0”になるまでリフレッシュ・サイクルを連続して発生します。

リフレッシュ要求キューの内容が“7”で、さらにリフレッシュ要求が発生した場合、リフレッシュ要求キューの内容は“7”のまま変化しません。

(2) リフレッシュ・コントロール・レジスタ (RFC)

リフレッシュ・コントロール・レジスタ (RFC) は、リフレッシュの許可/禁止、リフレッシュ・サイクルの長さ、リフレッシュの間隔を設定します。16ビット単位でリード/ライトできます。リフレッシュの間隔は、次の計算式により求められます。

$$\begin{aligned} \text{リフレッシュ間隔} (\mu\text{s}) &= \\ &\text{リフレッシュ・カウント・クロック} (t_{\text{RCY}}) \times \text{インターバル・ファクタ} (RI) \end{aligned}$$

リフレッシュ・カウント・クロック、インターバル・ファクタは、それぞれRFCレジスタのRCCビット、RIビットにより求めます。

ただし、BCUはDRAMアクセス後は常に $\overline{\text{RAS}}$ 信号をアクティブにするので、リフレッシュ周期が $\overline{\text{RAS}}$ パルス幅 (t_{RAS}) の最大値より長い場合は、リフレッシュ周期を t_{RAS} の最大値より短くしてください。

注意 RFCレジスタのRIビットの設定を変更する場合は、必ずRENビットでリフレッシュを禁止したあとにRIビットの内容を変更してください。

図7-3 リフレッシュ・コントロール・レジスタ (RFC)

アドレス : C0000022H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RFC	REN	0	0	RFTC	RFW	RCC	0								RI	

初期値 :

1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W : R/W R R R/W R/W R/W R/W R/W R R/W R/W R/W R/W R/W R/W R/W

ビット	ビット名	説明															
15	REN	Refresh Enable CBRリフレッシュ, CBRセルフ・リフレッシュの許可/禁止を設定します。 0 : CBRリフレッシュ, CBRセルフ・リフレッシュ禁止 1 : CBRリフレッシュ, CBRセルフ・リフレッシュ許可															
12	RFTC	Refresh Request Terminal Count REFRQ/TC の出力を切り替えます。 0 : \overline{TC} 出力 1 : \overline{REFRQ} 出力															
11, 10	RFW	Refresh Wait CBR リフレッシュ・サイクルの \overline{RAS} アクティブ期間を設定します。 <table border="1" style="margin: 5px auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;">RFW</th> <th style="width: 10%;"></th> <th style="width: 80%;">RAS アクティブ期間</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>3バス・クロック</td></tr> <tr><td>0</td><td>1</td><td>4バス・クロック</td></tr> <tr><td>1</td><td>0</td><td>5バス・クロック</td></tr> <tr><td>1</td><td>1</td><td>設定禁止</td></tr> </tbody> </table>	RFW		RAS アクティブ期間	0	0	3バス・クロック	0	1	4バス・クロック	1	0	5バス・クロック	1	1	設定禁止
RFW		RAS アクティブ期間															
0	0	3バス・クロック															
0	1	4バス・クロック															
1	0	5バス・クロック															
1	1	設定禁止															
9, 8	RCC	Refresh Count Clock リフレッシュ・カウント・クロック (t_{RCY}) を指定します。 は内部バス・クロックの周波数を示します。 <table border="1" style="margin: 5px auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;">RCC</th> <th style="width: 10%;"></th> <th style="width: 80%;">リフレッシュ・カウント・クロック (t_{RCY})</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>32/</td></tr> <tr><td>0</td><td>1</td><td>128/</td></tr> <tr><td>1</td><td></td><td>設定禁止</td></tr> </tbody> </table>	RCC		リフレッシュ・カウント・クロック (t_{RCY})	0	0	32/	0	1	128/	1		設定禁止			
RCC		リフレッシュ・カウント・クロック (t_{RCY})															
0	0	32/															
0	1	128/															
1		設定禁止															
6-0	RI	Refresh Interval リフレッシュ・タイミング生成用インターバル・タイマのインターバル・ファクタを設定します。 <table border="1" style="margin: 5px auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;"></th> <th style="width: 10%;">RI</th> <th style="width: 80%;">インターバル・ファクタ</th> </tr> </thead> <tbody> <tr><td></td><td>0 0 0 0 0 0 0</td><td>1</td></tr> <tr><td></td><td>0 0 0 0 0 0 1</td><td>2</td></tr> <tr><td></td><td>: : : : : :</td><td>:</td></tr> <tr><td></td><td>1 1 1 1 1 1 1</td><td>128</td></tr> </tbody> </table>		RI	インターバル・ファクタ		0 0 0 0 0 0 0	1		0 0 0 0 0 0 1	2		: : : : : :	:		1 1 1 1 1 1 1	128
	RI	インターバル・ファクタ															
	0 0 0 0 0 0 0	1															
	0 0 0 0 0 0 1	2															
	: : : : : :	:															
	1 1 1 1 1 1 1	128															

7.3 Page-ROM制御機能

BCUは、Page-ROMへのページ・アクセスを制御します。ページ・サイズの設定とページ・アクセス時のウェイト数の設定ができます。

7.3.1 Page-ROMコンフィギュレーション・レジスタ (PRC)

Page-ROMのon-pageアクセス時のウェイト数およびページ・サイズを設定します。ブロック7にPage-ROMサイクルが設定されているときだけ有効です。

図7-4 Page-ROMコンフィギュレーション・レジスタ (PRC)

アドレス : C0000024H

	7	6	5	4	3	2	1	0
PRC	0	0	0	0	0	0	PWS	PS

初期値 :	0	0	0	0	0	0	1	1
-------	---	---	---	---	---	---	---	---

R/W :	R	R	R	R	R	R	R/W	R/W
-------	---	---	---	---	---	---	-----	-----

ビット	ビット名	説明						
1	PWS	Page-ROM Wait States Page-ROMのon-pageアクセス時のウェイト数を設定します。 0 : 0ウェイト 1 : 1ウェイト BCTCレジスタのCT7ビットでPage-ROMサイクルを選択している場合だけ有効です。 off-pageアクセス時のウェイト数は、PWC1レジスタのWS7ビットの設定により従います。						
0	PS	Page Size Page-ROMのページ・サイズを設定します。 <table border="1" style="margin-left: 20px; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">PS</th> <th style="width: 95%;">ページ・サイズ</th> </tr> </thead> <tbody> <tr> <td>0</td> <td> 16バイト 16/32ビット・バス・モードで有効です。両モードとも、バースト・サイクルの1回目のアクセスがoff-pageアクセスになります。 </td> </tr> <tr> <td>1</td> <td> 8バイト 16ビット・バス・モード時だけ有効です。8回連続アクセスのバースト・サイクルでは、1回目と5回目のアクセスがoff-pageアクセスになります。 32ビット・バス・モード時はこの設定は無視されます。 4回連続アクセスのバースト・サイクルでは、1回目だけoff-pageアクセスになります。 </td> </tr> </tbody> </table>	PS	ページ・サイズ	0	16バイト 16/32ビット・バス・モードで有効です。両モードとも、バースト・サイクルの1回目のアクセスがoff-pageアクセスになります。	1	8バイト 16ビット・バス・モード時だけ有効です。8回連続アクセスのバースト・サイクルでは、1回目と5回目のアクセスがoff-pageアクセスになります。 32ビット・バス・モード時はこの設定は無視されます。 4回連続アクセスのバースト・サイクルでは、1回目だけoff-pageアクセスになります。
PS	ページ・サイズ							
0	16バイト 16/32ビット・バス・モードで有効です。両モードとも、バースト・サイクルの1回目のアクセスがoff-pageアクセスになります。							
1	8バイト 16ビット・バス・モード時だけ有効です。8回連続アクセスのバースト・サイクルでは、1回目と5回目のアクセスがoff-pageアクセスになります。 32ビット・バス・モード時はこの設定は無視されます。 4回連続アクセスのバースト・サイクルでは、1回目だけoff-pageアクセスになります。							

第8章 DMA機能

V831は、DMA転送を制御するDMA (Direct Memory Access) コントローラがあります。

DMAC (DMAコントローラ) は、内蔵周辺ハードウェア (シリアル・インタフェース, タイマ), 外部DMARQ端子, またはソフトウェア・トリガによるDMA要求に基づいて, メモリ-I/O間またはメモリ-メモリ間でデータを転送します。

8.1 特 徴

4組の独立したDMAチャンネル

転送単位: バイト, ハーフワード (2バイト), ワード (4バイト) 単位

最大転送回数: 16,777,216 (2^{24}) 回

転送タイプ: 2サイクル転送

2種類の転送モード

- ・シングル転送モード
- ・ディマンド転送モード

転送要求

- ・外部DMARQ端子 (4本)
- ・内蔵周辺ハードウェア (シリアル・インタフェース (3本), タイマ) からの要求
- ・ソフトウェアによる要求

転送対象

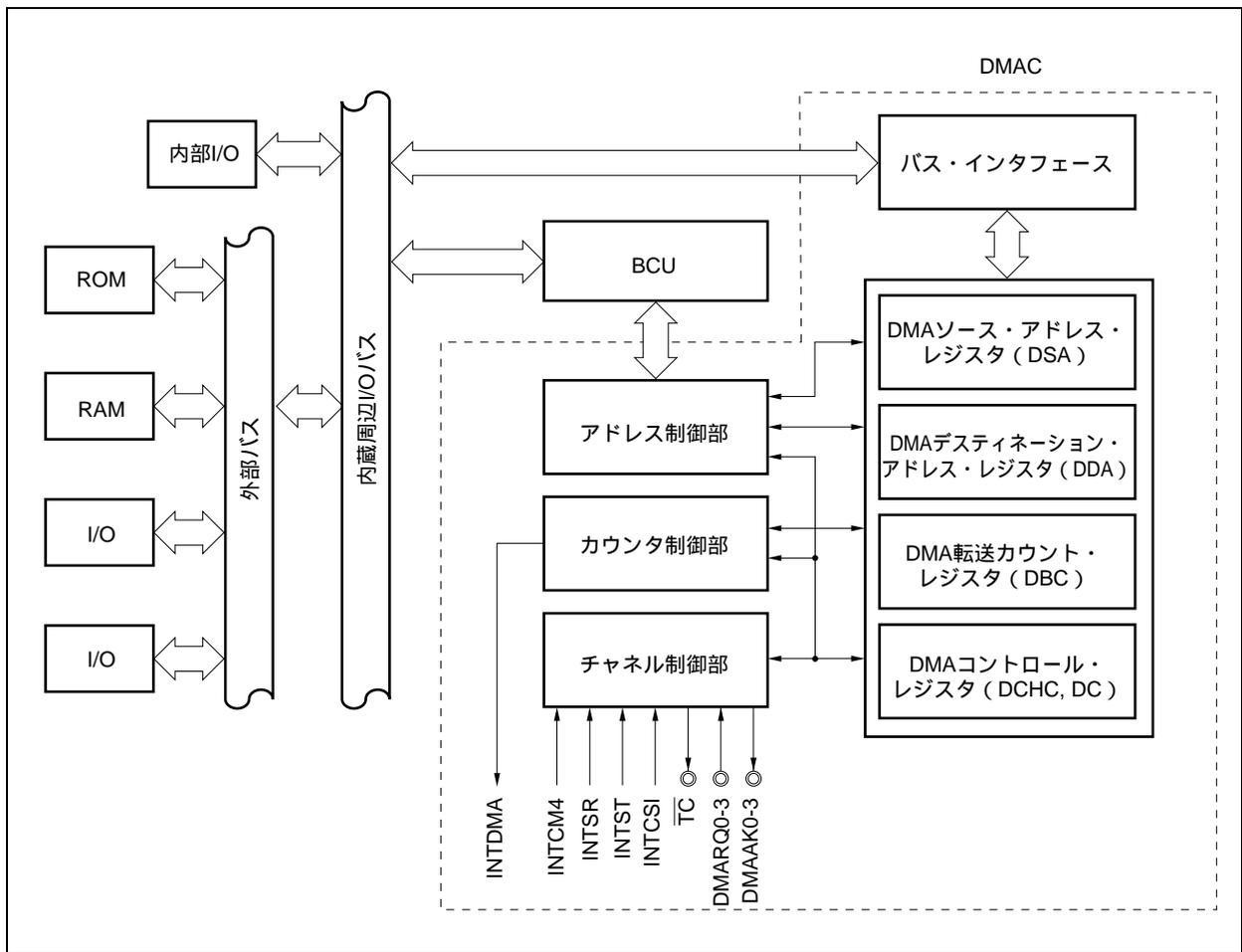
- ・メモリ-I/O
- ・メモリ-メモリ

プログラマブル・ウェイト機能

DMA転送終了信号出力 (\overline{TC})

8.2 構成

図8 - 1 DMACブロック図



8.3 DMA制御レジスタ

8.3.1 DMAソース・アドレス・レジスタ0-3 (DSA0-DSA3)

DMAチャンネル0-3のDMA転送元アドレス（32ビット）を設定します。このレジスタは、DSAnH、DSAnLの2つの16ビット・レジスタに分かれます（ $n=0-3$ ）。設定するアドレス値は、DMA転送データ・サイズ（DCHCレジスタで設定）にあわせて設定してください（ハーフワード転送：2の倍数、ワード転送：4の倍数）。

DMA転送中は、次のDMA転送元アドレスを保持します。

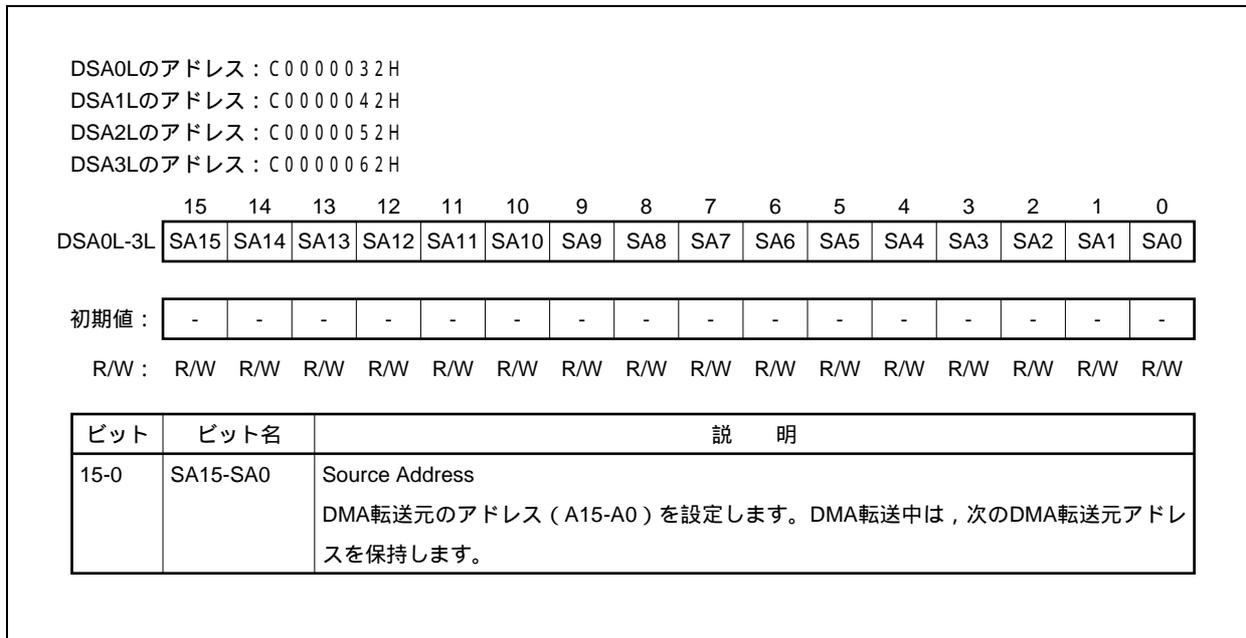
(1) DMAソース・アドレス・レジスタ0H-3H (DSA0H-DSA3H)

図8-2 DMAソース・アドレス・レジスタ0H-3H (DSA0H-DSA3H)

DSA0Hのアドレス：C0000030H DSA1Hのアドレス：C0000040H DSA2Hのアドレス：C0000050H DSA3Hのアドレス：C0000060H																
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DSA0H-3H	SA31	SA30	SA29	SA28	SA27	SA26	SA25	SA24	SA23	SA22	SA21	SA20	SA19	SA18	SA17	SA16
初期値：	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W：	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット	ビット名	説明														
15-8	SA31-SA24	Source Address DMA転送元のアドレス（A31-A24）を設定します。DMA転送中は、次のDMA転送元アドレスを保持します。 注意 設定するアドレスは32ビットですが、外部に出力されるアドレスはA23-A1ビットのため、カウンタは24ビット幅としてSA31-SA24の8ビットは固定になります。したがって、チップ・セレクトに対応したブロックにまたがる転送はできません。														
7-0	SA23-SA16	Source Address DMA転送元のアドレス（A23-A16）を設定します。DMA転送中は、次のDMA転送元アドレスを保持します。														

(2) DMAソース・アドレス・レジスタ0L-3L (DSA0L-DSA3L)

図8 - 3 DMAソース・アドレス・レジスタ0L-3L (DSA0L-DSA3L)



8.3.2 DMAデスティネーション・アドレス・レジスタ0-3 (DDA0-DDA3)

DMAチャネル0-3のDMA転送元アドレス (32ビット) を設定します。このレジスタは、DDAnH, DDAnLの2つの16ビット・レジスタに分かれます (n = 0-3)。設定するアドレス値は、DMA転送データ・サイズ (DCHCレジスタで設定) にあわせて設定してください (ハーフワード転送: 2の倍数, ワード転送: 4の倍数)。

DMA転送中は、次のDMA転送先アドレスを保持します。

(1) DMAデスティネーション・アドレス・レジスタ0H-3H (DDA0H-DDA3H)

図8-4 DMAデスティネーション・アドレス・レジスタ0H-3H (DDA0H-DDA3H)

DDA0Hのアドレス : C0000034H
 DDA1Hのアドレス : C0000044H
 DDA2Hのアドレス : C0000054H
 DDA3Hのアドレス : C0000064H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DDA0H-3H	DA31	DA30	DA29	DA28	DA27	DA26	DA25	DA24	DA23	DA22	DA21	DA20	DA19	DA18	DA17	DA16

初期値 : - - - - - - - - - - - - - - -

R/W : R/W R/W

ビット	ビット名	説明
15-8	DA31-DA24	Destination Address DMA転送先のアドレス (A31-A24) を設定します。DMA転送中は、次のDMA転送先アドレスを保持します。 注意 設定するアドレスは32ビットですが、外部に出力されるアドレスはA23-A1ビットのため、カウンタは24ビット幅としてDA31-DA24の8ビットは固定になります。したがって、チップ・セレクトに対応したブロックにまたがる転送はできません。
7-0	DA23-DA16	Destination Address DMA転送先のアドレス (A23-A16) を設定します。DMA転送中は、次のDMA転送先アドレスを保持します。

(2) DMAデスティネーション・アドレス・レジスタ0L-3L (DDA0L-DDA3L)

図8 - 5 DMAデスティネーション・アドレス・レジスタ0L-3L (DDA0L-DDA3L)

DDA0Lのアドレス : C0000036H																
DDA1Lのアドレス : C0000046H																
DDA2Lのアドレス : C0000056H																
DDA3Lのアドレス : C0000066H																
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DDA0L-3L	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
初期値 :	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ビット	ビット名		説明													
15-0	DA15-DA0		Destination Address DMA転送先のアドレス (A15-A0) を設定します。DMA転送中は、次のDMA転送先アドレスを保持します。													

8.3.3 DMAバイト・カウント・レジスタ0-3 (DBC0-DBC3)

DMAチャンネル0-3のバイト転送数 (24ビット) を設定します。このレジスタは、DBCnH, DBCnLの2つの16ビット・レジスタに分かれます (n=0-3)。

DMA転送中は、残りのバイト転送数を保持します。

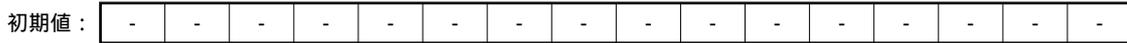
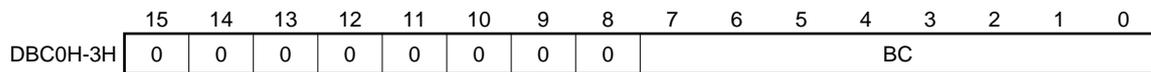
次にバイト転送、ハーフワード転送、ワード転送のデクリメントと設定を示します。ポローが発生すると転送が終了します。

- バイト転送 : 1つずつデクリメント 【設定】 転送回数 - 1
 ハーフワード転送 : 2つずつデクリメント 【設定】 (転送回数 - 1) × 2
 ワード転送 : 4つずつデクリメント 【設定】 (転送回数 - 1) × 4

図8-6 DMAバイト・カウント・レジスタ0-3 (DBC0-DBC3)

(1) DMAバイト・カウント・レジスタ0H-3H (DBC0H-DSC3H)

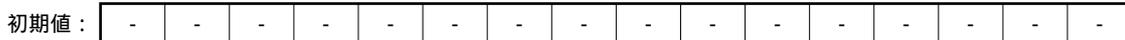
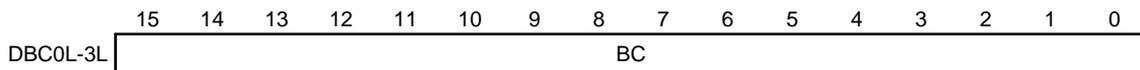
DBC0Hのアドレス : C0000038H
 DBC1Hのアドレス : C0000048H
 DBC2Hのアドレス : C0000058H
 DBC3Hのアドレス : C0000068H



R/W : R R R R R R R R R/W R/W R/W R/W R/W R/W R/W R/W

(2) DMAバイト・カウント・レジスタ0L-3L (DBC0L-DBC3L)

DBC0Lのアドレス : C000003AH
 DBC1Lのアドレス : C000004AH
 DBC2Lのアドレス : C000005AH
 DBC3Lのアドレス : C000006AH



R/W : R/W R/W

ビット	ビット名	説明										
7-0	BC	Byte Count										
: DBCnH		バイト転送数を設定します。DMA転送中は、残りのバイト転送数を保持します。										
15-0		DBCnHレジスタのビット7がMSB, DBCnLレジスタのビット0がLSBになります (n=0-3)。										
: DBCnL		<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>BC</th> <th>状態</th> </tr> </thead> <tbody> <tr> <td>000000H</td> <td>1回のバイト転送または残りバイト転送数</td> </tr> <tr> <td>000001H</td> <td>2回のバイト転送または残りバイト転送数</td> </tr> <tr> <td>:</td> <td>:</td> </tr> <tr> <td>FFFFFFH</td> <td>2²⁴回のバイト転送または残りバイト転送数</td> </tr> </tbody> </table>	BC	状態	000000H	1回のバイト転送または残りバイト転送数	000001H	2回のバイト転送または残りバイト転送数	:	:	FFFFFFH	2 ²⁴ 回のバイト転送または残りバイト転送数
BC	状態											
000000H	1回のバイト転送または残りバイト転送数											
000001H	2回のバイト転送または残りバイト転送数											
:	:											
FFFFFFH	2 ²⁴ 回のバイト転送または残りバイト転送数											

8.3.4 DMAチャンネル・コントロール・レジスタ0-3 (DCHC0-DCHC3)

DMAチャンネル0-3のDMA転送動作モードを制御する16ビット・レジスタです。

図8-7 DMAチャンネル・コントロール・レジスタ0-3 (DCHC0-DCHC3) (1/3)

DCHC0のアドレス : C000003CH
 DCHC1のアドレス : C000004CH
 DCHC2のアドレス : C000005CH
 DCHC3のアドレス : C000006CH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCHC0-3	0	TTYP		TBT	SAD	DAD	DAL	DRL	TM	DS	EN					

初期値 :

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W : R R/W R/W

ビット	ビット名	説明																																				
14-12	TTYP	Transfer Type DMA転送の起動要因を設定します (n = 0-3)。 <table border="1" style="margin: 10px auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="3">TTYP</th> <th>起動要因</th> </tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td> <td>DMARQn信号で起動 : 2サイクル転送</td> </tr> <tr> <td>0</td><td>0</td><td>1</td> <td>ソフトウェアで起動 : 2サイクル転送</td> </tr> <tr> <td>0</td><td>1</td><td>0</td> <td>設定禁止</td> </tr> <tr> <td>0</td><td>1</td><td>1</td> <td>設定禁止</td> </tr> <tr> <td>1</td><td>0</td><td>0</td> <td>INTST信号で起動 : 2サイクル転送</td> </tr> <tr> <td>1</td><td>0</td><td>1</td> <td>INTSR信号で起動 : 2サイクル転送</td> </tr> <tr> <td>1</td><td>1</td><td>0</td> <td>INTCSI信号で起動 : 2サイクル転送</td> </tr> <tr> <td>1</td><td>1</td><td>1</td> <td>INTCM4信号で起動 : 2サイクル転送</td> </tr> </tbody> </table>	TTYP			起動要因	0	0	0	DMARQn信号で起動 : 2サイクル転送	0	0	1	ソフトウェアで起動 : 2サイクル転送	0	1	0	設定禁止	0	1	1	設定禁止	1	0	0	INTST信号で起動 : 2サイクル転送	1	0	1	INTSR信号で起動 : 2サイクル転送	1	1	0	INTCSI信号で起動 : 2サイクル転送	1	1	1	INTCM4信号で起動 : 2サイクル転送
TTYP			起動要因																																			
0	0	0	DMARQn信号で起動 : 2サイクル転送																																			
0	0	1	ソフトウェアで起動 : 2サイクル転送																																			
0	1	0	設定禁止																																			
0	1	1	設定禁止																																			
1	0	0	INTST信号で起動 : 2サイクル転送																																			
1	0	1	INTSR信号で起動 : 2サイクル転送																																			
1	1	0	INTCSI信号で起動 : 2サイクル転送																																			
1	1	1	INTCM4信号で起動 : 2サイクル転送																																			
11, 10	TBT	Transfer Block Type 転送元, 転送先のアドレス・ブロックがメモリかI/Oかを設定します。 <table border="1" style="margin: 10px auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="2">TBT</th> <th colspan="2">転送ブロック</th> </tr> </thead> <tbody> <tr> <td>0</td><td>0</td> <td>メモリ</td><td>メモリ</td> </tr> <tr> <td>0</td><td>1</td> <td>メモリ</td><td>I/O</td> </tr> <tr> <td>1</td><td>0</td> <td>I/O</td><td>メモリ</td> </tr> <tr> <td>1</td><td>1</td> <td colspan="2">設定禁止</td> </tr> </tbody> </table>	TBT		転送ブロック		0	0	メモリ	メモリ	0	1	メモリ	I/O	1	0	I/O	メモリ	1	1	設定禁止																	
TBT		転送ブロック																																				
0	0	メモリ	メモリ																																			
0	1	メモリ	I/O																																			
1	0	I/O	メモリ																																			
1	1	設定禁止																																				
9, 8	SAD	Source Address count Direction DMAチャンネルnの転送元アドレスのカウンタ方向を設定します (n = 0-3)。 <table border="1" style="margin: 10px auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th colspan="2">SAD</th> <th>カウンタ方向</th> </tr> </thead> <tbody> <tr> <td>0</td><td>0</td> <td>インクリメント</td> </tr> <tr> <td>0</td><td>1</td> <td>デクリメント</td> </tr> <tr> <td>1</td><td>0</td> <td>固定</td> </tr> <tr> <td>1</td><td>1</td> <td>設定禁止</td> </tr> </tbody> </table>	SAD		カウンタ方向	0	0	インクリメント	0	1	デクリメント	1	0	固定	1	1	設定禁止																					
SAD		カウンタ方向																																				
0	0	インクリメント																																				
0	1	デクリメント																																				
1	0	固定																																				
1	1	設定禁止																																				

図8-7 DMAチャンネル・コントロール・レジスタ0-3 (DCHC0-DCHC3) (2/3)

DCHC0のアドレス : C000003CH
 DCHC1のアドレス : C000004CH
 DCHC2のアドレス : C000005CH
 DCHC3のアドレス : C000006CH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCHC0-3	0	TTYP			TBT	SAD	DAD	DAL	DRL	TM	DS	EN				

初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W : R R/W R/W

ビット	ビット名	説明										
7, 6	DAD	Destination Address Count Direction DMAチャンネルnの転送先アドレスのカウンタ方向を設定します (n = 0-3)。 <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr> <th>DAD</th> <th>カウンタ方向</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td>インクリメント</td> </tr> <tr> <td>0 1</td> <td>デクリメント</td> </tr> <tr> <td>1 0</td> <td>固定</td> </tr> <tr> <td>1 1</td> <td>設定禁止</td> </tr> </tbody> </table>	DAD	カウンタ方向	0 0	インクリメント	0 1	デクリメント	1 0	固定	1 1	設定禁止
DAD	カウンタ方向											
0 0	インクリメント											
0 1	デクリメント											
1 0	固定											
1 1	設定禁止											
5	DAL	DMAAK Level DMAAKn信号のアクティブ・レベルを設定します (n = 0-3)。 0 : アクティブ・ロウ 1 : アクティブ・ハイ 注意 リセットからレジスタ設定の間, DMAAKn信号はハイ・レベル出力になります。										
4	DRL	DMARQ Level DMARQn信号の検出レベルを設定します (n = 0-3)。 0 : ロウ・レベル検出 1 : ハイ・レベル検出										
3	TM	Transfer Mode DMA転送時の転送モードを設定します (n = 0-3)。 0 : シングル転送モード 1 : デイマンド転送モード 注意 DMA転送の起動要因を内蔵周辺ハードウェアからの要求とする場合は, シングル転送モードにしてください。										
2, 1	DS	Data Size DMA転送での転送サイズを設定します。転送先がI/O空間の場合, 転送データ・サイズの設定は, DBCレジスタに設定したデータ・バス幅 (転送先) 以上にしてください (アドレス・カウンタ固定 (DADビット = 10) の場合を除きます)。 <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr> <th>DS</th> <th>転送データ・サイズ</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td>バイト単位</td> </tr> <tr> <td>0 1</td> <td>ハーフワード (2バイト) 単位</td> </tr> <tr> <td>1 0</td> <td>ワード (4バイト) 単位</td> </tr> <tr> <td>1 1</td> <td>設定禁止</td> </tr> </tbody> </table>	DS	転送データ・サイズ	0 0	バイト単位	0 1	ハーフワード (2バイト) 単位	1 0	ワード (4バイト) 単位	1 1	設定禁止
DS	転送データ・サイズ											
0 0	バイト単位											
0 1	ハーフワード (2バイト) 単位											
1 0	ワード (4バイト) 単位											
1 1	設定禁止											

図8-7 DMAチャンネル・コントロール・レジスタ0-3 (DCHC0-DCHC3) (3/3)

DCHC0のアドレス : C000003CH
 DCHC1のアドレス : C000004CH
 DCHC2のアドレス : C000005CH
 DCHC3のアドレス : C000006CH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DCHC0-3	0	TTYP			TBT	SAD		DAD	DAL	DRL	TM	DS		EN		

初期値 :

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W : R R/W R/W

ビット	ビット名	説明
0	EN	Enable DMAチャンネルnのDMA転送の許可/禁止を設定します (n = 0-3)。ターミナル・カウントによりDMA転送が終了するとリセットされます。DMA転送を開始するには、ENビット = 1, DCレジスタのTCnビット = 0, MENビット = 1に設定してください (n = 0-3)。 0 : DMA転送禁止 (リセット) 1 : DMA転送許可

8.3.5 DMAコントロール・レジスタ (DC)

DMA転送動作モードを制御する16ビット・レジスタです。

図8-8 DMAコントロール・レジスタ (DC)

アドレス : C000006EH																	
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
DC	0	0	0	0	0	0	0	0	0	TC3	TC2	TC1	TC0	0	0	0	MEN
初期値 :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
ビット	ビット名		説明														
7-4	TC3-TC0		Terminal Count Channel3-0 DMAチャンネルnのDMA転送の終了/未終了を示すステータス・ビットです (n = 0-3)。読み出しだけです。ターミナル・カウントによりDMA転送が終了するとセットされ、読み出しによりリセットされます。 0 : DMA転送未終了 (リセット) 1 : DMA転送終了 (セット)														
0	MEN		Master Enable DMA全チャンネルのDMA転送の許可/禁止を設定します (n = 0-3)。また、 $\overline{\text{NMI}}$ 信号入力によってリセットされます。DMA転送を開始するには、MENビット = 1, TCnビット = 0, DCHCnレジスタのENビット = 1に設定してください (n = 0-3)。各チャンネルの転送許可/禁止は、DCHC0-DCHC3レジスタのENビットを使用してください。 例 DMA転送が許可されたときMENビット = 1として、 $\overline{\text{NMI}}$ 信号によりMENビット = 0とされたときだけ、NMI処理の中でMENビット = 1とする。 0 : DMA転送禁止 (リセット) 1 : DMA転送許可														

8.4 転送モード

8.4.1 シングル転送モード

シングル転送では、DMACは1回の転送ごとにバスを解放します。そのあとDMA転送要求があると再度1回の転送を行います。この動作をターミナル・カウントが発生するまで続けます。

DMACがバスを解放している間にほかの優先順位の高いDMA転送要求が発生した場合、常に優先順位の高いDMA要求を優先させます。

図8 - 9、8 - 10にシングル転送の例を示します。図8 - 10は優先順位の高いDMA要求が発生した場合のシングル転送例で、DMAチャンネル0はディマンド転送、チャンネル1はシングル転送です。

図8 - 9 シングル転送の例1

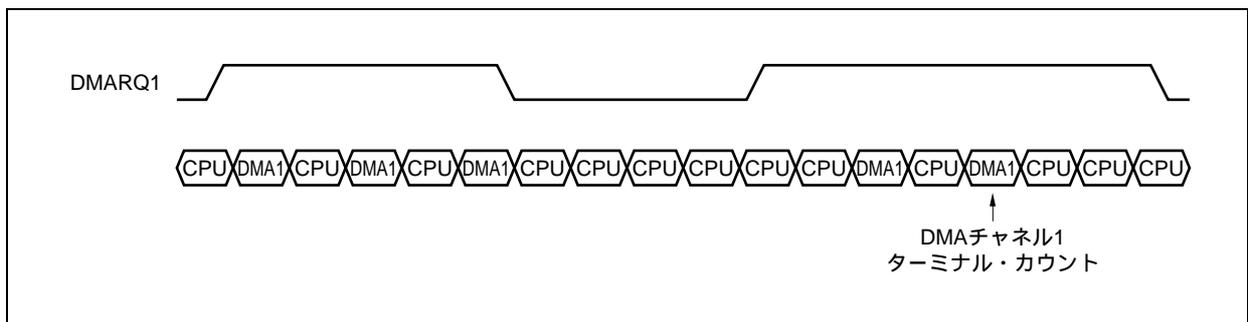
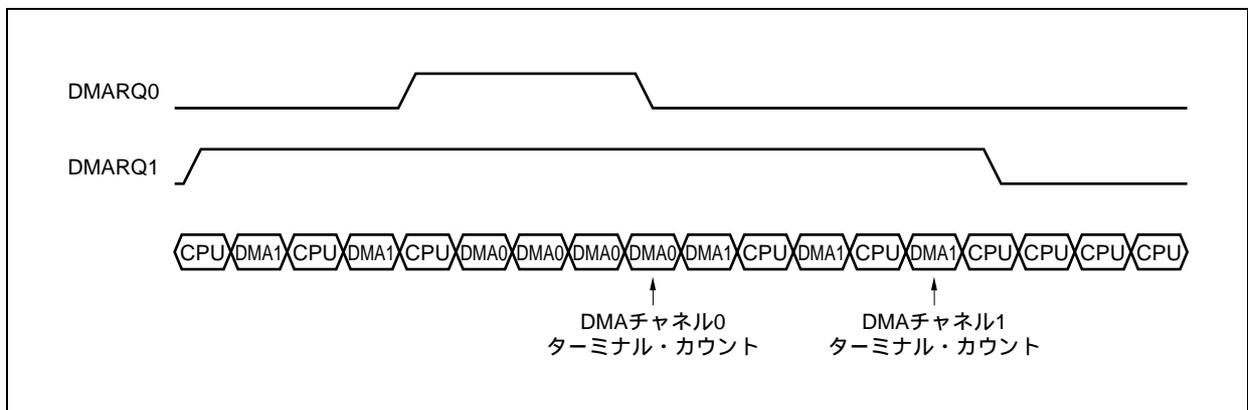


図8 - 10 シングル転送の例2



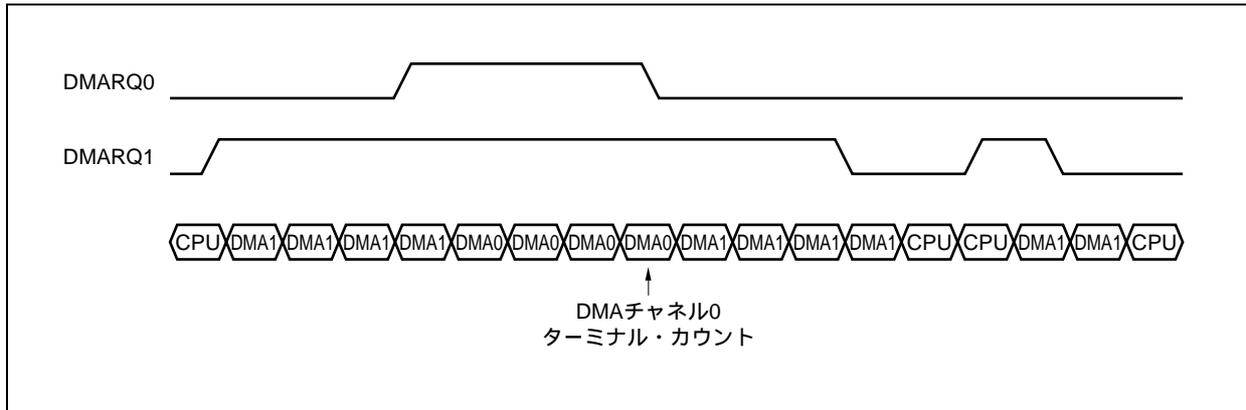
8.4.2 デイマンド転送モード

デイマンド転送では、DMA転送要求が発生している間バスを解放しないで転送を続けます。DMA転送要求が発生していれば、ターミナル・カウントが発生するまで転送を続けます。

DMA転送要求が中断して再度DMA転送要求が発生した場合は、転送を再開できます。

図8 - 11にデイマンド転送の例を示します。優先順位の高いDMA要求が発生した場合のデイマンド転送の例で、DMAチャンネル0, 1はデイマンド転送です。

図8 - 11 デイマンド転送の例



8.5 DMA転送タイプと転送対象

8.5.1 2サイクル転送

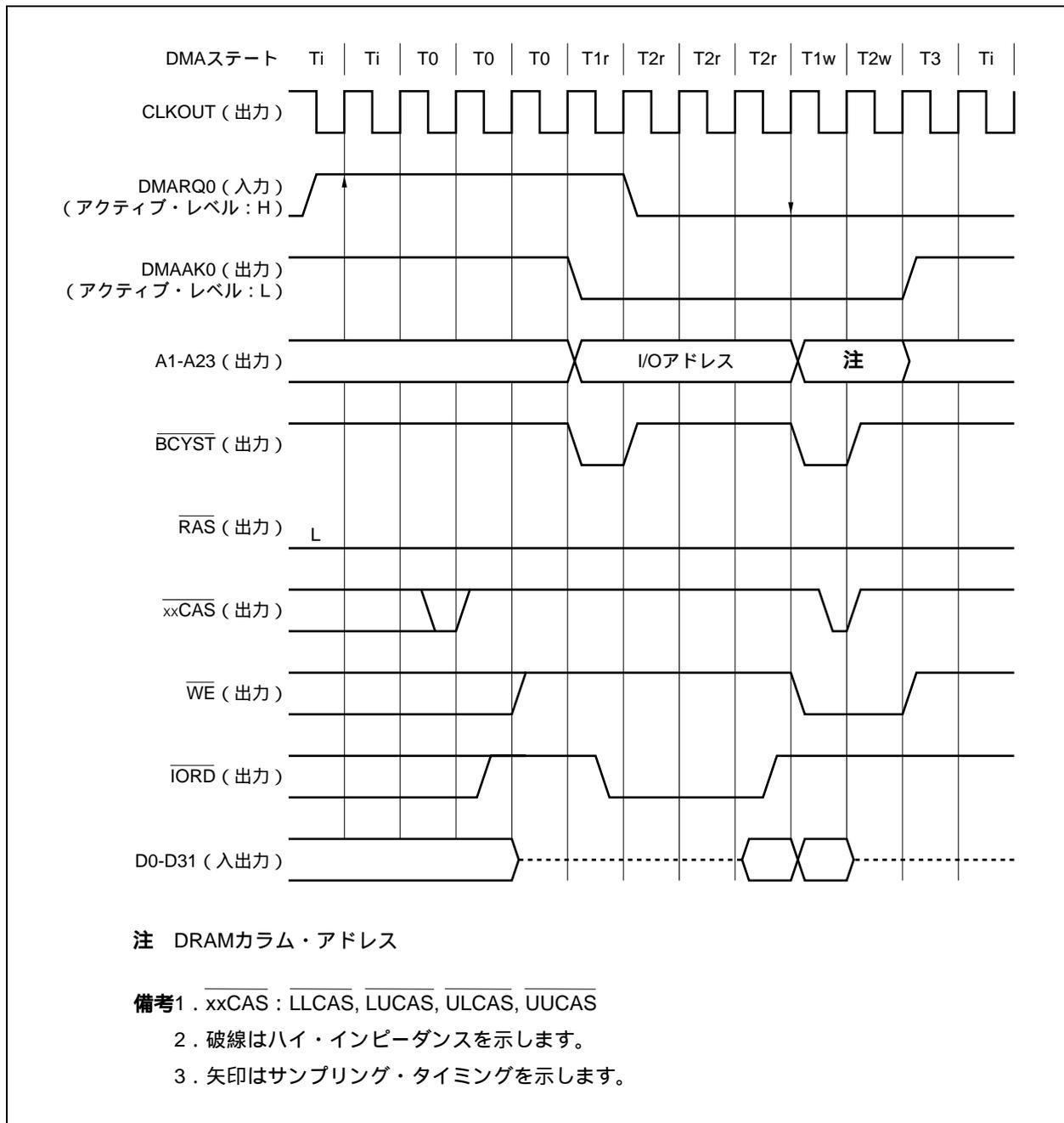
2サイクル転送は、転送元 DMAC, DMAC 転送先と2回のサイクルでデータを転送します。

1回目のサイクル：転送元のアドレスを出力して転送元からDMACへ読み込みます。

2回目のサイクル：転送先アドレスを出力してDMACから転送先へ書き込みます。

図8 - 12に2サイクル転送の例を示します。

図8 - 12 2サイクル・ディマンド転送（外部I/O DRAM (on-page)）



8.5.2 転送対象

転送の種類と対象の関係を次に示します。

DMA転送対象	転送種類
I/O（外部，内部）-メモリ	2サイクル
メモリ-メモリ	2サイクル

- 注意1. DMACは内蔵RAMにアクセスできません。
 2. DMACはCPUコア内の内部I/Oにアクセスできません。IN.W/OUT.W命令を使用してください。
 3. DMA転送中にDMACレジスタへの書き込みは行わないでください。
 4. キャッシュ領域へDMA転送（書き込み）を行った場合は、メモリとキャッシュの値が異なるので、必要に応じてキャッシュのインバリッドを行ってください。

8.6 DMAチャンネルの優先順位

DMAチャンネルの優先順位は固定で、次のようになります。

チャンネル0 > チャンネル1 > チャンネル2 > チャンネル3

DMA転送要求をサンプリングするのは、DMAステートのTiステートのクロックの立ち上がり時と、ライト・サイクルの $\overline{\text{BCYST}}$ 信号がアクティブとなるクロックの立ち上がり時だけです（ライト・サイクルのサンプリングはダイヤモンド転送中だけ行われます）。このとき優先順位が有効になり、次の転送から優先順位が高いDMA転送が行われます。

8.7 DMA転送要求

DMA転送要求には、外部DMARQ端子からの要求、ソフトウェアからの要求、内蔵周辺ハードウェアからの要求の3種類があります。それぞれDMAチャンネル制御レジスタ（DCHC）で設定します。3種類のどの転送要求であってもDMAAK信号は出力します。

8.7.1 DMARQ端子からの要求

DMARQ端子による要求は、DMAステートのTiステートのクロックの立ち上がりでサンプリングされますが、対応するDMAAK信号がアクティブになるまで行ってください。

DMACがTiステートのときにDMARQ端子がアクティブになると、T0ステートに移行してDMA転送を開始します。

ダイヤモンド転送中のDMARQ端子による要求のサンプリングは、ライト・サイクルの $\overline{\text{BCYST}}$ 信号がアクティブとなるクロックの立ち上がりで行います。

ダイヤモンド転送時に次の転送をしない場合は、T1rステートの $\overline{\text{BCYST}}$ 信号を確認したあと、DMARQ信号をインアクティブにしてください（ライト・サイクルの $\overline{\text{BCYST}}$ 信号がアクティブとなるクロックの立ち上がりでサンプリングするため、リード・サイクルで処理してください）。

DMARQ端子をアクティブにするとDMA転送を開始しますが、DMARQ端子をアクティブの状態のままにしておくと次のDMA転送を開始します。システムを設計する際は、DMARQ端子をアクティブにする期間を十分考慮してください。転送モードによる次のDMA転送に対するDMARQ端子のサンプリング・タイミングを次に示します。

(1) 2サイクル・ダイヤモンド転送

DMARQ端子は、ダイヤモンド転送時に図8-13に示すのタイミングでサンプリングされます。のタイミングで、2回目のDMA転送サイクルのDMARQ端子がサンプリングされます。

ダイヤモンド転送時に、2回目のDMA転送サイクルを発生させたくない場合は、からまでの間にDMARQ端子をインアクティブにしてください。

図8-13のタイミングは、DMA転送のアイドル・ステートが設定されていません。アイドル・ステートを設定した場合、SRAMリードとDRAMライト（TsとTc）の間にアイドル・ステートが挿入されます。

図8-13 2サイクル・ディマンド転送(16ビットSRAM 32ビットDRAM, 32ビット転送時)

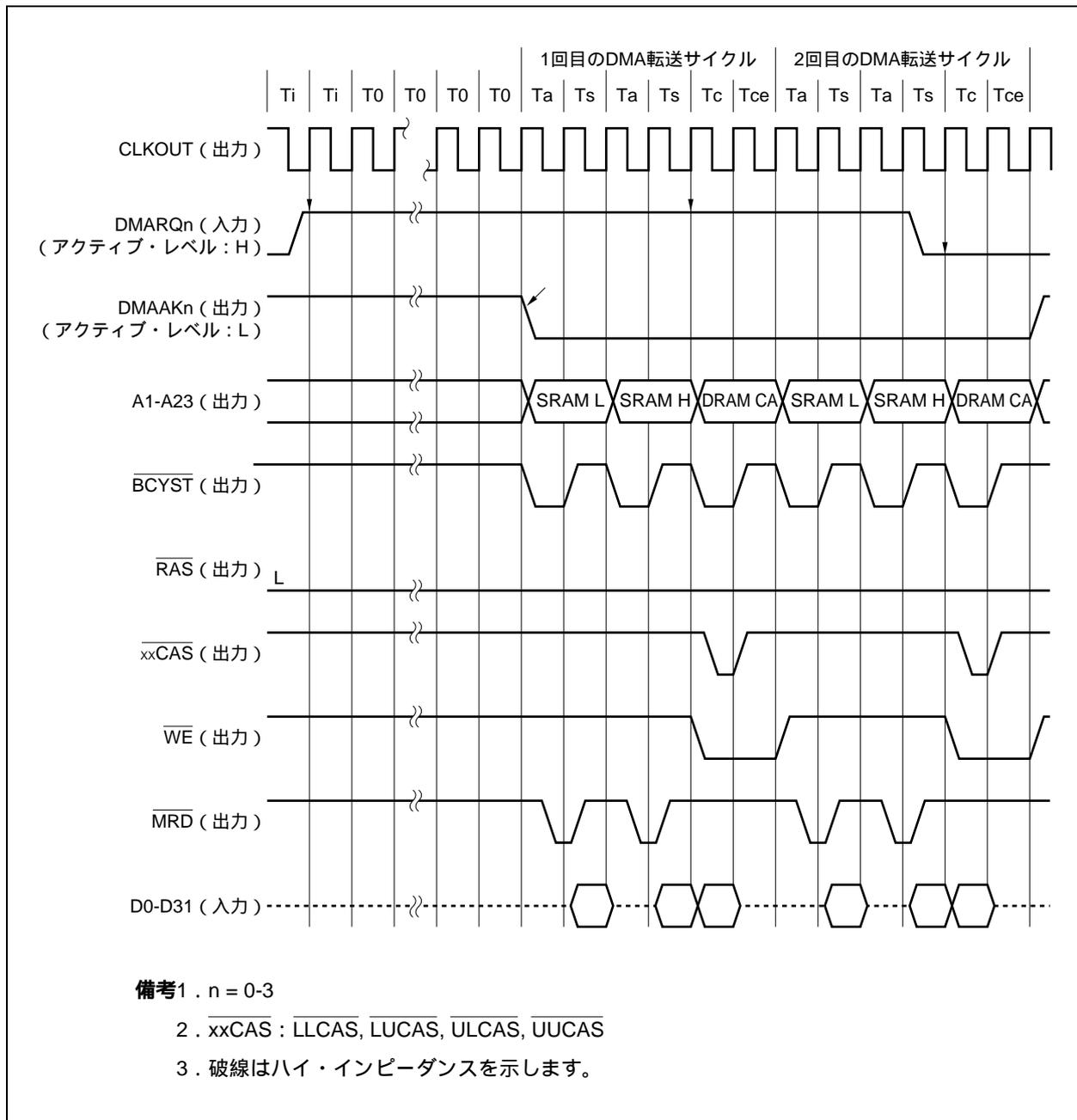
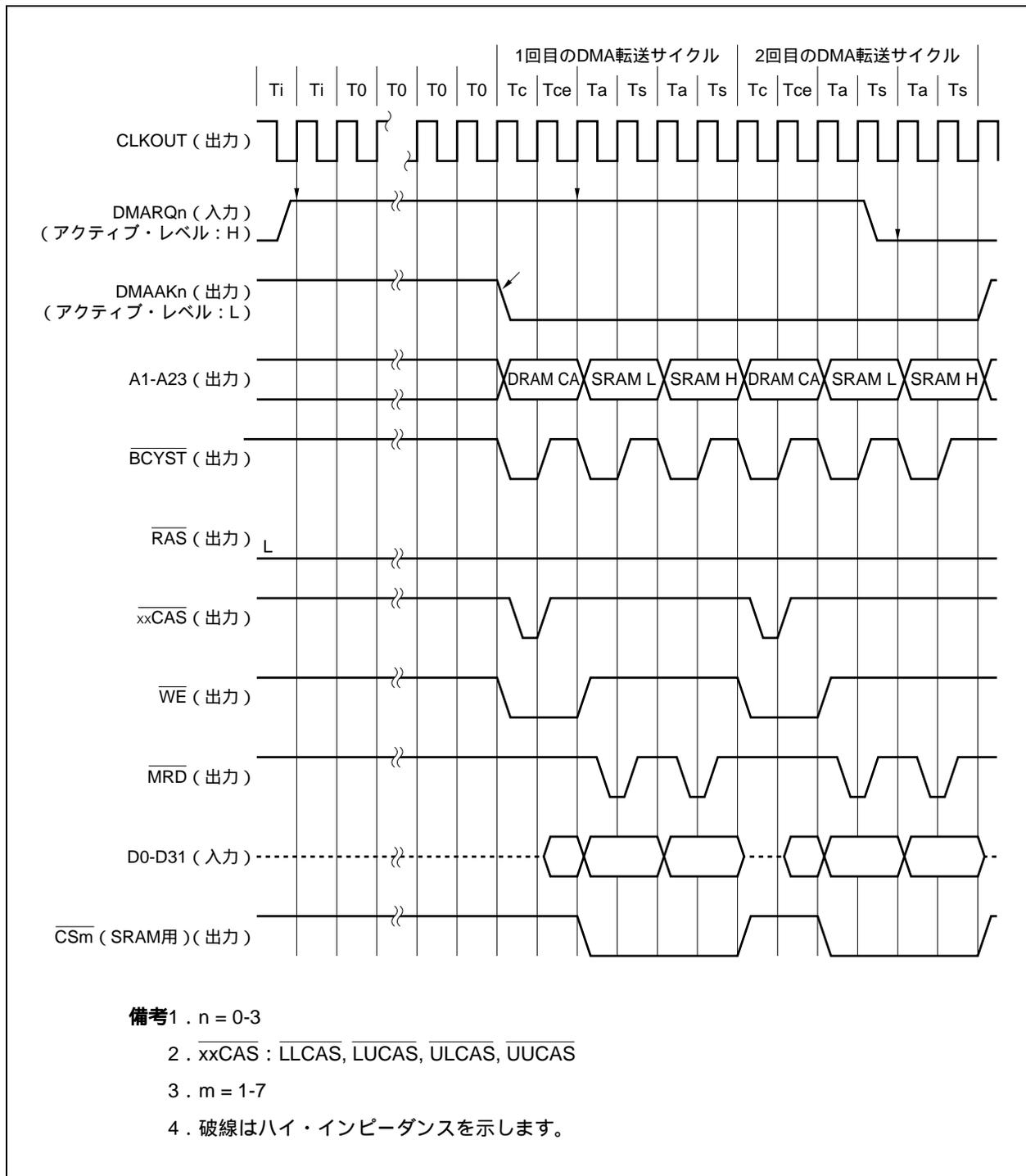


図8 - 14 2サイクル・ディマンド転送 (32ビットDRAM 16ビットSRAM, 32ビット転送時)



(2) シングル転送

DMARQ端子は、シングル転送時に図8 - 15に示す のタイミングでサンプリングされ、 のタイミン
 グまで保持します。 のタイミングで、2回目のDMA転送サイクルのDMARQ端子がサンプリングされま
 す。シングル転送時に、2回目のDMA転送サイクルを発生させたくない場合は、 から までの間にDMARQ
 端子をインアクティブにしてください。

図8 - 15 シングル転送 (16ビットSRAM 32ビットDRAM, 32ビット転送時)

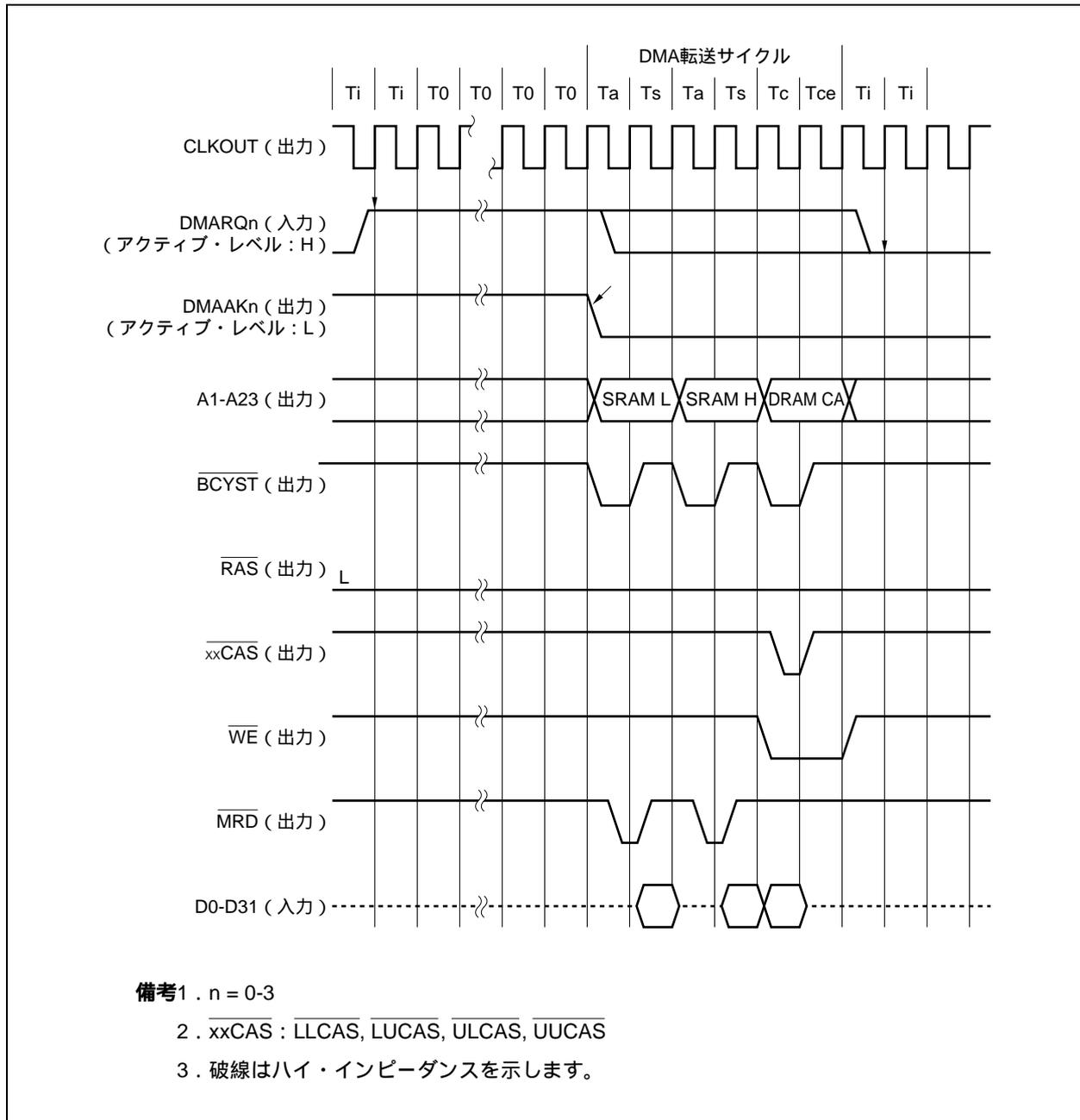
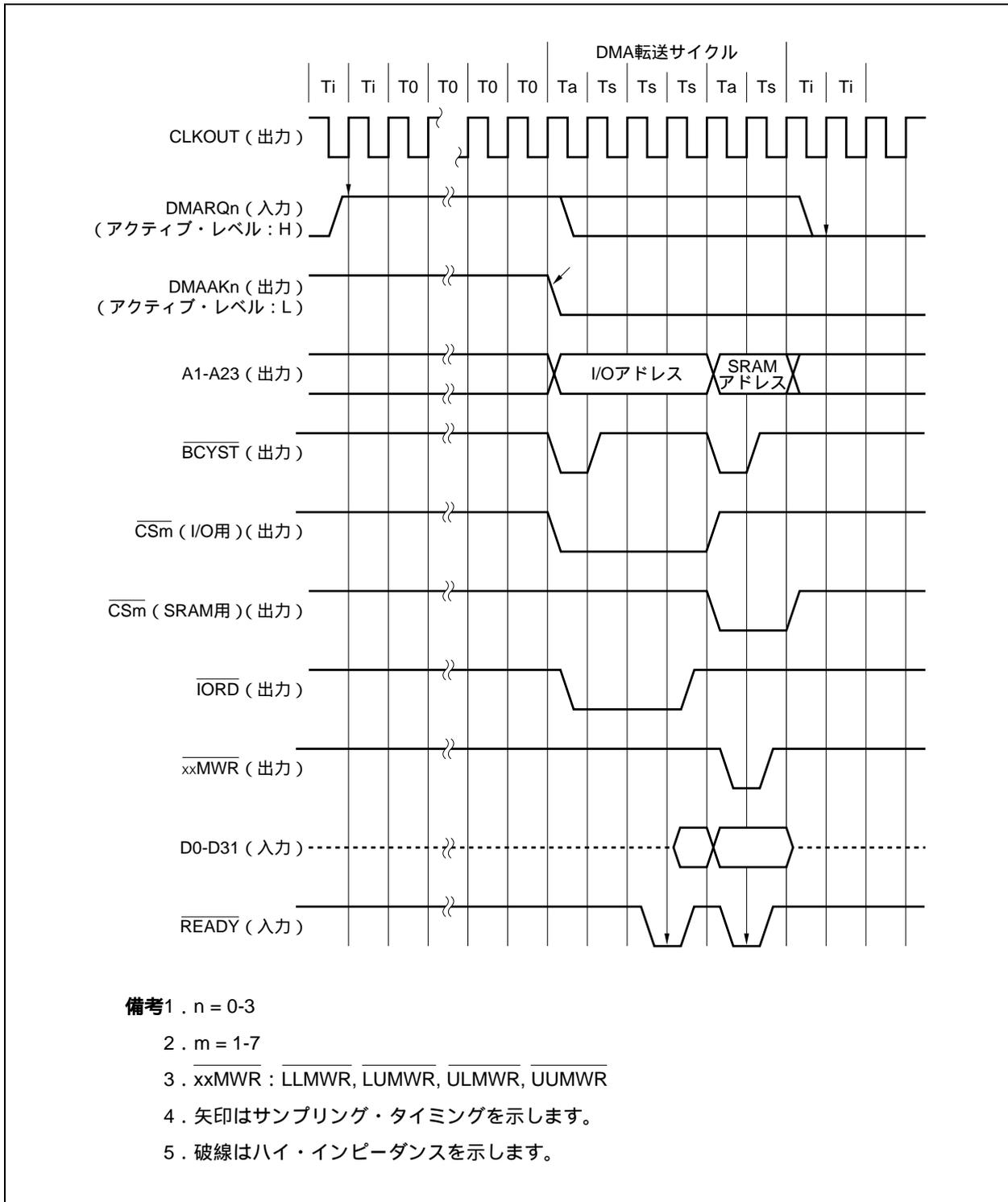


図8 - 16 シングル転送 (8ビットI/O 32ビットSRAM , 8ビット転送時)



8.7.2 ソフトウェアからの要求

ソフトウェアによる要求は、DCHCレジスタのENビットを1にするとDMA転送を開始します。

8.7.3 内蔵周辺ハードウェアからの要求

内蔵周辺ハードウェアからの転送要求信号（割り込み要求信号）には、次の4種類があります。

- ・UARTからの送信終了割り込み要求（INTST）
- ・UARTからの受信終了割り込み要求（INTSR）
- ・CSIからの送受信終了割り込み要求（INTCSI）
- ・RPUからのコンペア・レジスタ4（CM4）の一致割り込み要求（INTCM4）

内蔵周辺ハードウェアからの転送要求により1転送行われます。転送が終了する前に次の転送要求が発生しても、その要求は無視されます。1つの内蔵周辺ハードウェアからの転送要求を複数のチャンネルで使用することはできません。

シングル転送で使用してください。

★ 内蔵周辺ハードウェアからの転送要求は、割り込み要求マスク・レジスタ（IMR）でマスクしていても発生します。内蔵周辺ハードウェアの割り込み要求をDMAの転送要求信号として使用する場合も、割り込み要求は発生します。内蔵周辺ハードウェアからの転送要求を使用時、同一内蔵周辺からの割り込み発生を望まないときは、割り込み要求マスク・レジスタで対応する割り込み要求をマスクしてください。

注意 内蔵周辺ハードウェアから転送要求する場合は、DMA転送の設定（転送許可設定を含む）が完了してから内蔵周辺ハードウェアからの転送要求を発生させてください。DMA転送を設定する前に内蔵周辺ハードウェアからの転送要求が発生している場合は、DMA転送設定直後に転送を行ってしまいます。次に例を2つ示します。

例1. 2つのタスク（タスク1とタスク2）でCSIのデータを処理する場合

タスク1：CSIからCPUの割り込み処理によるソフトウェア転送

タスク2：CSIからDMA転送

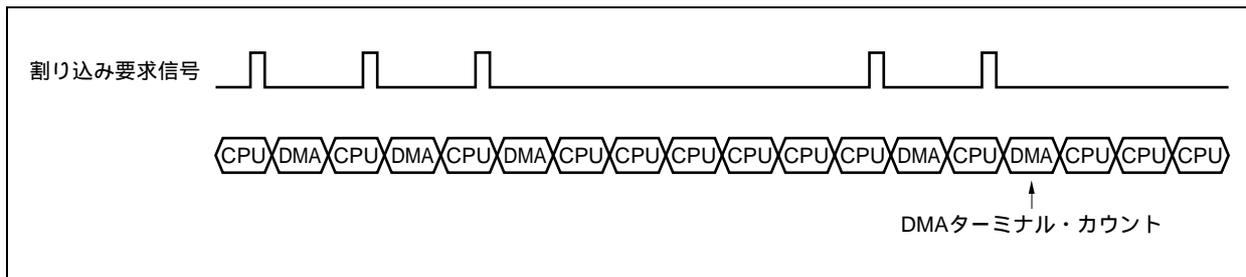
CSIにデータを受信してタスク1（ソフトウェア転送）でCSIのデータを受信したあとに、タスク2のDMA転送を設定すると、タスク1の割り込みによるDMA転送要求が保持された状態になっているため、タスク2のDMA転送を設定した途端にCSIにデータを受信していないのにDMA転送を行ってしまいます。

また、最初にDMARQ端子からDMA転送を起動する設定をしてから、内蔵周辺ハードウェアからDMA転送を起動する設定に変更した場合は、その内蔵周辺ハードウェアから1回でも割り込みが発生すると、内蔵周辺ハードウェアからの起動に切り換えた瞬間にDMA転送が起動されます。

例2. タイマ（CM4）の動作中に、ある期間だけINTCM4によるDMA転送する場合

割り込み要求レジスタ（IRR）にタイマ割り込みによる転送要求が保持されているため、DMA転送を許可した時点でDMA転送が開始されます。

図8 - 17 内蔵周辺ハードウェアからの要求による転送例



8.8 DMA転送終了割り込み

DMA転送が終了してDCレジスタのTCnビットが1になると、割り込みコントローラに対してDMA転送終了割り込み要求を発生します（表4 - 2 割り込み一覧（マスカブル割り込み）参照）。

8.8.1 TCnビット参照とDMA転送終了割り込みの動作

V831のDMACは、DMA転送が終了するとDMA転送終了割り込み（INTDMA）が発生するとともに、DMAコントロール・レジスタ（DC）のTCnビットの該当するチャンネルのビットがセット（1）されます。複数のDMAチャンネルを使用するシステムの場合は、INTDMAハンドラ内のTCnビットを読み出すタイミングと、INTDMAの割り込みラッチをクリアするタイミングに注意してください（n = 0-3）。

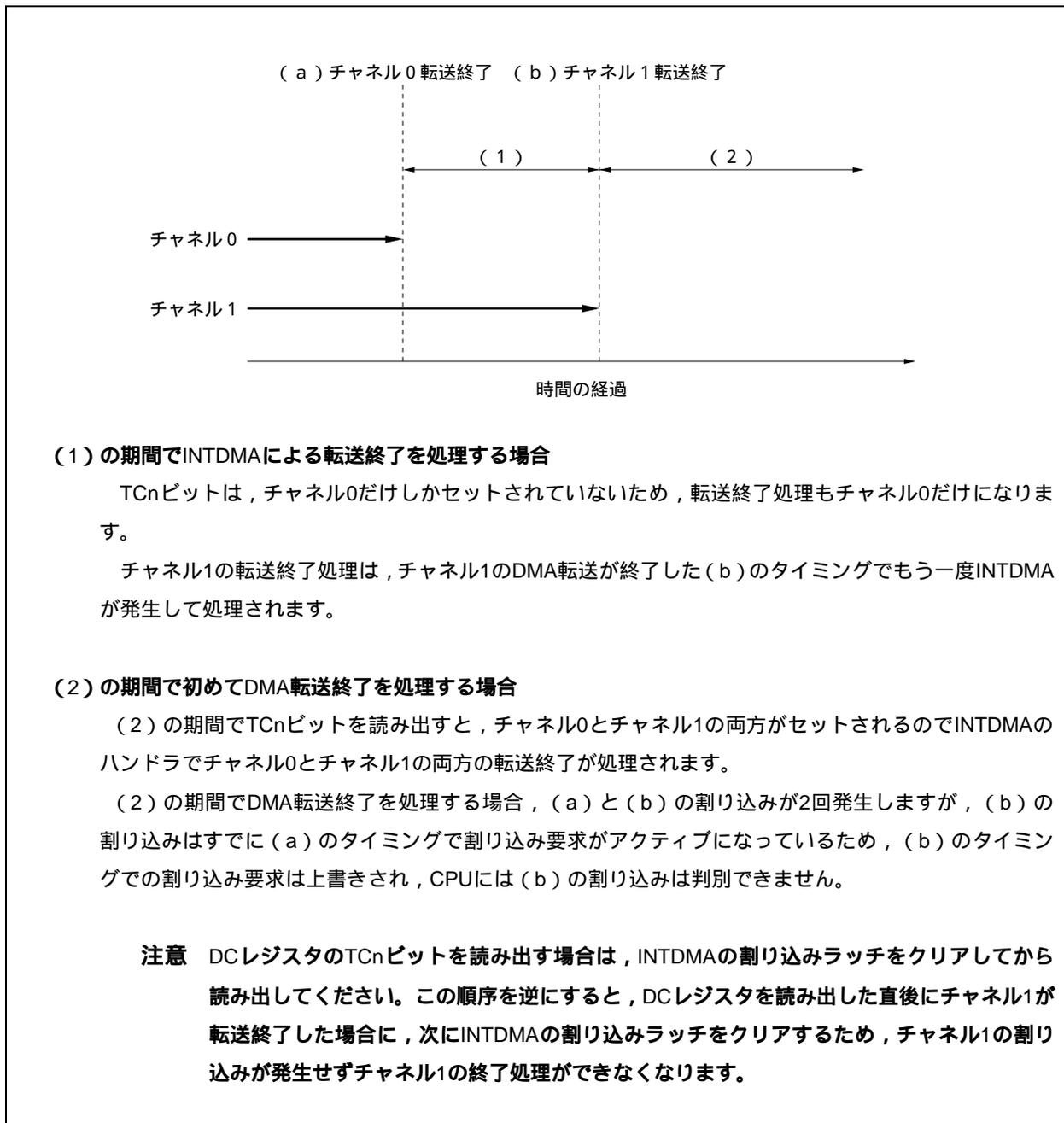
たとえば、2チャンネル（チャンネル0とチャンネル1）を使用してDMA転送している場合、INTDMAの割り込みハンドラは次に示す手順で処理してください。

DMA転送終了割り込み（INTDMA）の割り込みラッチをクリアします。

DMAコントロール・レジスタ（DC）のTCnビットを読み出します。

TCnがセットされているビットに該当するすべてのDMAチャンネルの転送終了を処理します。

図8 - 18 チャンネル0とチャンネル1の転送終了処理



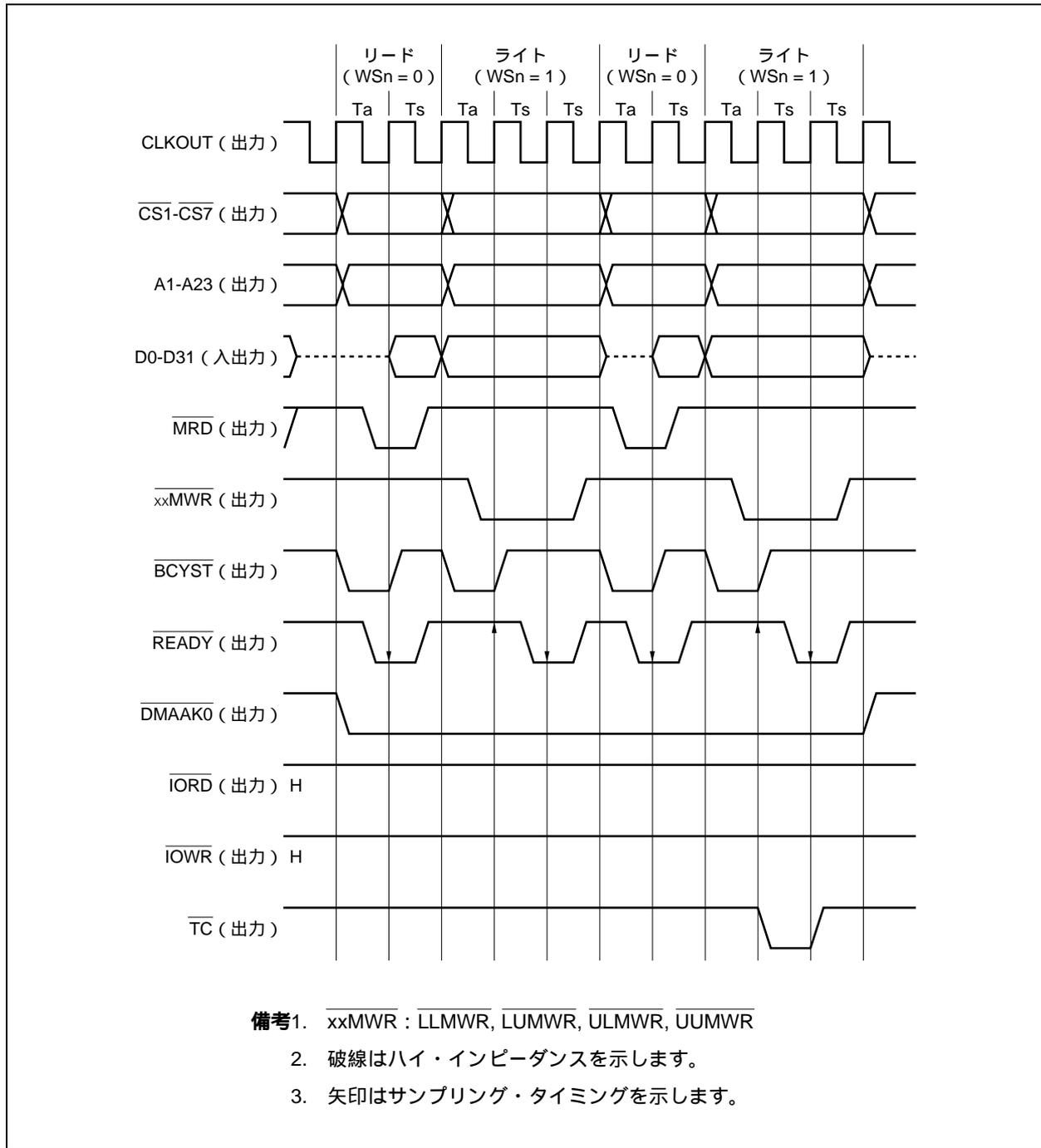
8.9 DMA転送終了出力

DMA転送が終了するライト・サイクルの $\overline{\text{BCYST}}$ 信号がアクティブになった次のクロックで、 $\overline{\text{TC}}$ 信号が1クロックの間アクティブになります(内蔵周辺I/Oライト時は、ライト・サイクルの2クロック目で出力)。図8 - 19にSRAM SRAM転送時の $\overline{\text{TC}}$ 信号の出力タイミングを示します。

$\overline{\text{TC}}$ 信号は、チャンネル0-3のDMA転送終了出力の論理和をとって出力します。外部回路でDMAAK0-DMAAK3信号と論理積をとることにより、チャンネル0-3のDMA転送終了出力を生成できます。

$\overline{\text{TC}}/\overline{\text{REFRQ}}$ 兼用端子の端子機能選択は、RFCレジスタのRFTCビットで設定します。初期値は $\overline{\text{REFRQ}}$ 端子が選択(RETC ビット = 1)されているため、 $\overline{\text{REFRQ}}$ 信号を出力します。

図8 - 19 DMA転送終了出力タイミング



8. 10 強制中断

8. 10. 1 $\overline{\text{NMI}}$ 信号による中断

DMA転送中に $\overline{\text{NMI}}$ 信号入力により、DMA転送を強制的に中断（ライト・サイクル終了後中断）できます。このときDMACは、DCレジスタのMENビットを0にして、DMA転送を禁止します。また、この状態でMENビットを1にするとDMA転送を中断したところから再開します。

ノンマスカブル割り込み処理中に $\overline{\text{NMI}}$ 信号入力があった場合、CPUコアはこの割り込みを保留しますが、DMA転送の強制中断要求は保留されません。したがって、MENビット = 0のときに $\overline{\text{NMI}}$ 信号入力があってもDMA転送の強制中断要求は無視されます。

STOPモード中（バス・クロック停止）に $\overline{\text{NMI}}$ 信号入力があっても、MENビットは0になりません。バス・クロック出力が開始されても、 $\overline{\text{NMI}}$ 信号がアクティブであればMENビットは0になります。

8. 10. 2 $\overline{\text{HLDRQ}}$ 信号、リフレッシュによる一時中断

DMA転送中に $\overline{\text{HLDRQ}}$ 信号入力、またはDRAMリフレッシュ要求により、DMA転送を一時中断（ライト・サイクル終了後中断）できます（5. 9 バス・アービトレーション参照）。DMAより優先順位の高いバス・マスタがバスを解放したときに、DMA転送要求がアクティブであれば、DMA転送を再開します。一時中断しているとき、DMAAK信号はインアクティブになります。

8.11 DMA転送のバス・サイジング

V831は、32ビットから16ビットへのバス・サイジング機能があります。32ビット・データを16ビット・データ・バスを介してアクセスする場合は、2回アクセスします。この連続した2回のアクセスは、バースト・サイクルになります。

図8 - 20に転送元16ビット (I/O) , 転送先32ビット・データ・バス幅 (DRAM) の32ビット (ワード) デマンド転送の例を示します。

★ 図8 - 20 16-32ビット・データ・バス幅 (32ビット転送バス・サイジング)

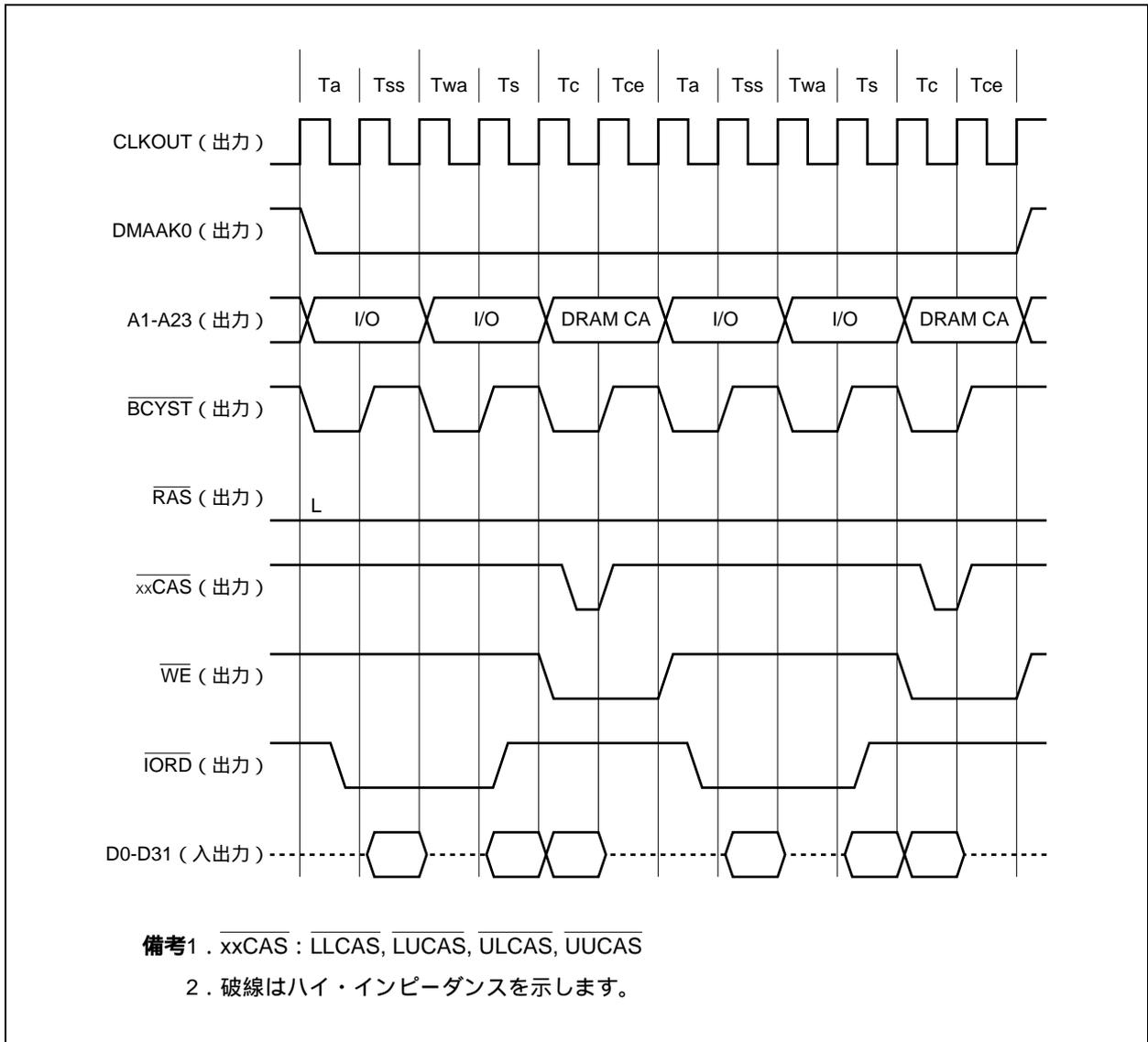
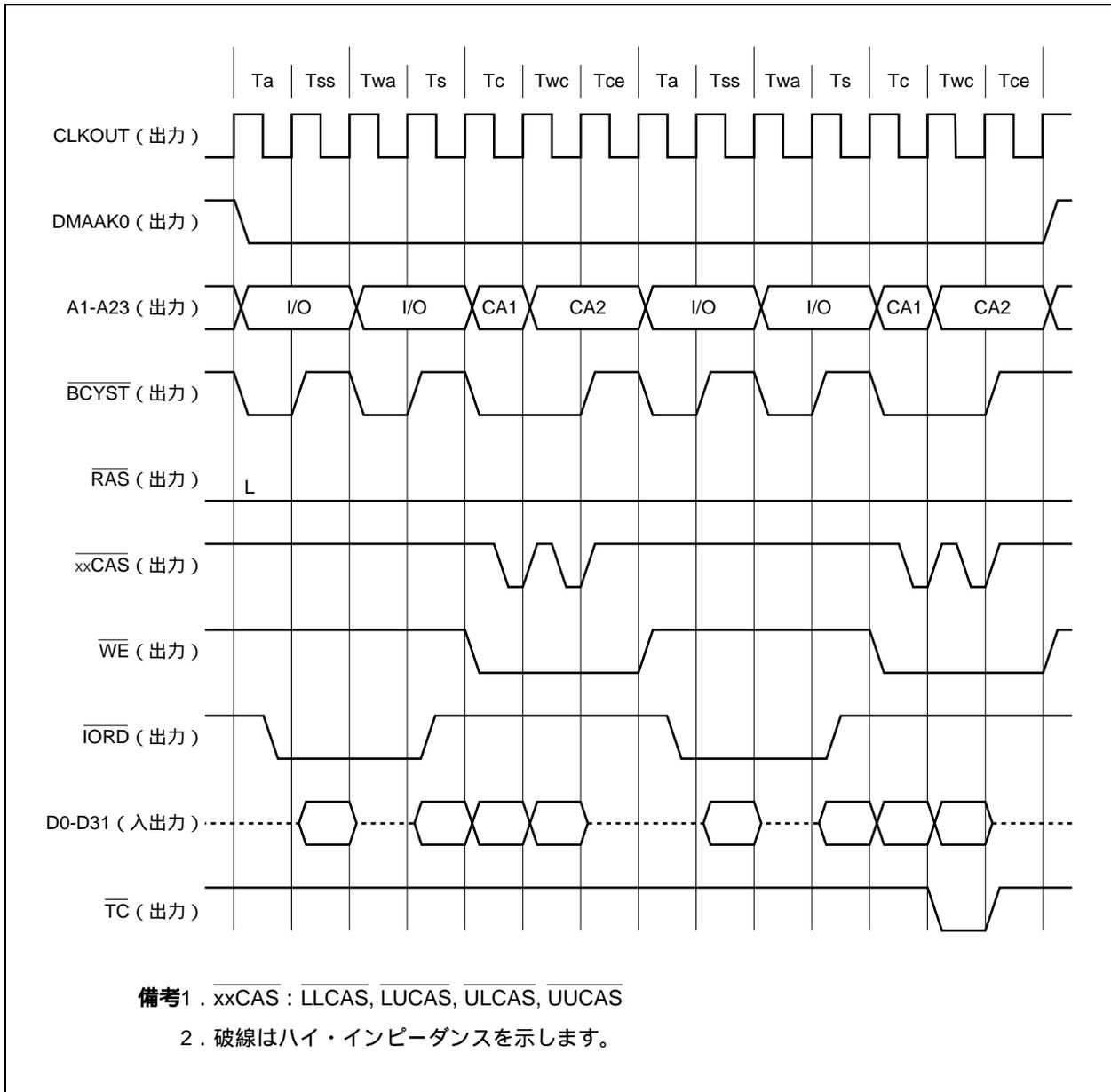


図8 - 21に転送元16ビット (I/O) , 転送先16ビット・データ・バス幅 (DRAM) の32ビット (ワード) デマンド転送の例を示します。

DRAMライト・サイクルがバス・サイジングによってバースト・サイクルになる場合、 \overline{TC} 信号の出力タイミングはTwcステートになります。DRAM以外の場合は、1回目のライト・サイクルで \overline{TC} 信号を出力します。

★

図8 - 21 16-16ビット・データ・バス幅 (32ビット転送バス・サイジング)



第9章 シリアル・インタフェース機能

V831は、シリアル・インタフェース機能として2チャンネルの送受信チャンネルがあります。インタフェース形態としては次の2種類があり、それぞれ1チャンネルずつ備えています。

- ・ アシクロナス・シリアル・インタフェース：UART (Universal Asynchronous Receiver/Transmitter)
- ・ クロック同期式シリアル・インタフェース：CSI (Clocked Serial Interface)

BRG (ボー・レート・ジェネレータ) は、1チャンネルを備えUART/CSIに排他的に使用できる構成になっています。

9.1 アシクロナス・シリアル・インタフェース (UART)

9.1.1 概要

V831のUARTには次に示す特徴があります。

- ・ 送信バッファ・レジスタを持っていません。
- ・ 専用ボー・レート・ジェネレータを内蔵して、ボー・レートを任意に設定できます。

(1) 送信バッファの削除

従来のUARTは送信側と受信側にそれぞれ送信バッファと受信バッファが設定されていますが、V831のUARTはハードウェアを軽減するために送信バッファを削除しています。送信シフト・レジスタへのデータ転送により送信処理を開始します。したがって、送信バッファに対する送信許可制御機能やCTS (シリアル送信制御) 端子入力による送信処理制御機能も削除しているので、割り込み制御のソフトウェア処理で対応してください。

なお、受信バッファについては従来どおり設定されています。

(2) 専用ボー・レート・ジェネレータの内蔵

V831は、シリアル・クロックを発生する専用ボー・レート・ジェネレータを1チャンネル内蔵しているので、精度の良いシリアル転送レートを設定できます。

9.1.2 特 徴

全二重通信 受信バッファ (RXB) 内蔵 (送信バッファ (TXB) は内蔵していません)

2端子構成 (V831のUARTはSCLK端子, CTS端子がありません)

- ・TXD 送信デ - タの出力端子
- ・RXD 受信デ - タの入力端子

転送速度: 150 bps-76800 bps (バス・クロック: 33 MHz, BRG使用時)

ボ - ・レ - ト・ジェネレ - タを内蔵

シリアル・クロック・ソースは, ボ - ・レ - ト・ジェネレ - タ出力とバス・クロック (ϕ) から選択可能

受信エラー - 検出機能

- ・パリティ・エラー -
- ・フレ - ミング・エラー -
- ・オ - バラン・エラー -

割り込みソ - ス (3種類)

- ・受信エラー - 割り込み (INTSER)

3種類の受信エラー - の論理和で割り込み要求を発生します (エラー内容の詳細は, 9. 1. 4 (3) **アシンクロナス・シリアル・インタフェース・ステータス・レジスタ (ASIS0)** を参照)。

- ・受信終了割り込み (INTSR)

受信許可状態のとき, シフト・レジスタから受信バッファへ受信データの転送が終了すると受信終了割り込み要求が発生します。

- ・送信終了割り込み (INTST)

シリアル送信を行って, シフト・レジスタから送信デ - タ (9/8/7ビット) をシリアル送信し終わると送信終了割り込み要求を発生します。送受信デ - タのキャラクタ長は, ASIM00, ASIM01レジスタで指定します。

キャラクタ長: 7, 8ビット

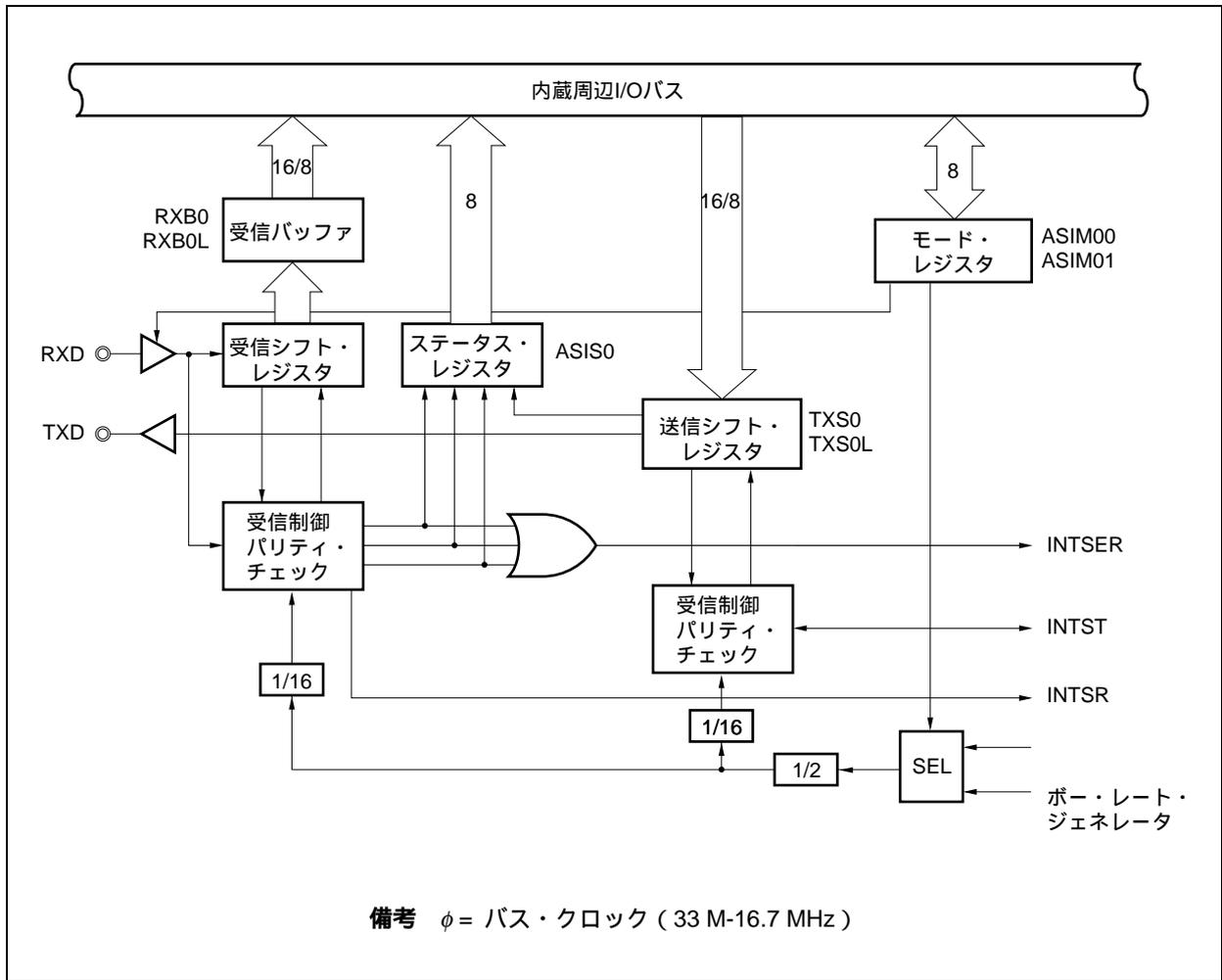
 : 9ビット (拡張ビット付加時)

パリティ機能: 奇数, 偶数, 0, なし

送信ストップ・ビット: 1, 2ビット

9.1.3 構成

図9-1 UARTのブロック図



9.1.4 モード・レジスタとコントロール・レジスタ

(1) アシクロナス・シリアル・インタフェース・モード・レジスタ00 (ASIM00)

UARTの転送モードを指定します。8ビット単位でリード/ライトできます。

注意 UARTの送受信中にASIM00の値を変更した場合の動作は保証できません。

図9-2 アシクロナス・シリアル・インタフェース・モード・レジスタ00 (ASIM00) (1/2)

アドレス : C0000090H

	7	6	5	4	3	2	1	0
ASIM00	1	RXE0	PS01	PS00	CL0	SL0	0	SCLS0

初期値 :

1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

R/W : R R/W R/W R/W R/W R/W R R/W

ビット	ビット名	説明															
6	RXE0	Receive Enable 受信許可状態 / 禁止状態を指定します。 0 : 受信禁止状態 1 : 受信許可状態 受信禁止時に受信シフト・レジスタはスタート・ビットを検出しません。シフト・イン処理，受信バッファへの転送処理は行わず，受信バッファの内容は保持されます。受信許可状態中はスタート・ビットの検出に同期して受信シフト動作を開始し，1フレーム分の受信を終了すると受信シフト・レジスタの内容を受信バッファに転送します。また，受信バッファへの転送に同期して，受信終了割り込み (INTSR0) を発生します。															
5, 4	PS01, PS00	Parity Select パリティ・ビットを指定します。 <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">PS01</th> <th style="width: 5%;">PS00</th> <th style="width: 90%;">動作</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>パリティなし</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td>0パリティ指定 送信側 パリティ・ビットを0にして送信 受信側 受信時にパリティ・エラーを発生しない</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>奇数パリティ指定</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td>偶数パリティ指定</td> </tr> </tbody> </table>	PS01	PS00	動作	0	0	パリティなし	0	1	0パリティ指定 送信側 パリティ・ビットを0にして送信 受信側 受信時にパリティ・エラーを発生しない	1	0	奇数パリティ指定	1	1	偶数パリティ指定
PS01	PS00	動作															
0	0	パリティなし															
0	1	0パリティ指定 送信側 パリティ・ビットを0にして送信 受信側 受信時にパリティ・エラーを発生しない															
1	0	奇数パリティ指定															
1	1	偶数パリティ指定															
3	CL0	Character Length 1フレームのキャラクタ長を指定します。 0 : 7ビット 1 : 8ビット															
2	SL0	Stop Bit Length ストップ・ビット数を指定します。 0 : 1ビット 1 : 2ビット															

図9-2 アシクロナス・シリアル・インタフェース・モードレジスタ00 (ASIM00) (2/2)

アドレス : C0000090H

ASIM00	7	6	5	4	3	2	1	0
	1	RXE0	PS01	PS00	CL0	SL0	0	SCLS0

初期値 :

1	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---

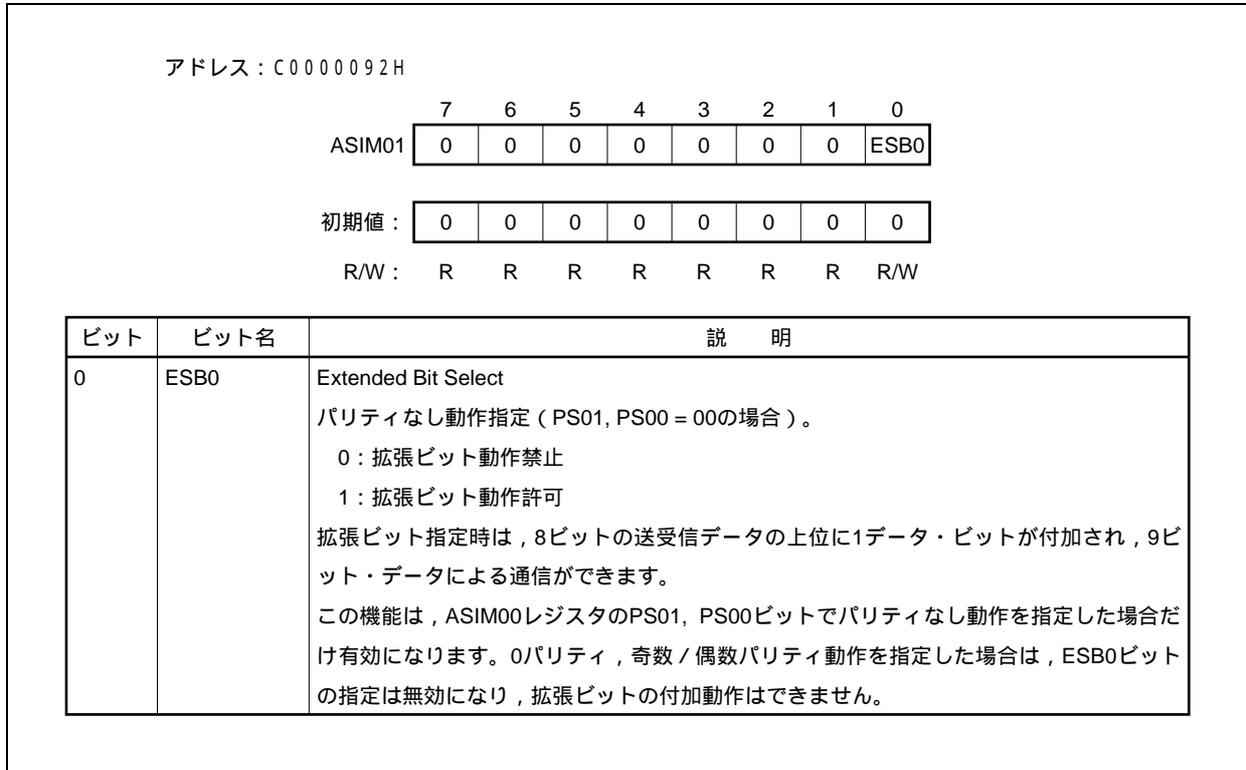
R/W : R R/W R/W R/W R/W R/W R R/W

ビット	ビット名	説明										
0	SCLS0	<p>Serial Clock Source</p> <p>シリアル・クロック・ソースを指定します。</p> <p style="margin-left: 20px;">0 : ポー・レート・ジェネレータ出力</p> <p style="margin-left: 20px;">1 : (バス・クロック)</p> <p>シリアル・クロックは、ASIM00レジスタのSCLS0ビットで指定されたシリアル・クロック・ソースを2分周したクロックです。シリアル・クロックの16分周したクロックがUARTのポー・レート・クロックになります。</p> <p>・ SCLS0 = 1の場合</p> <p style="margin-left: 20px;">シリアル・クロック・ソースに (バス・クロック) が選択されます。×16のサンプリング・レートを使用しているため、ポー・レートは次の式で表せます。</p> <p style="margin-left: 20px;">ポー・レート = $\text{クロック} / 2 / 16$ (bps)</p> <p style="margin-left: 20px;">上記の式に基づき代表的クロックを使用した場合のポー・レートの値を次に示します。</p> <table border="1" style="margin-left: 40px; border-collapse: collapse;"> <tr> <td style="width: 30%;"></td> <td style="width: 10%;">33 MHz</td> <td style="width: 10%;">25 MHz</td> <td style="width: 10%;">20 MHz</td> <td style="width: 10%;">16 MHz</td> </tr> <tr> <td>ポー・レート (bps)</td> <td style="text-align: center;">1031K</td> <td style="text-align: center;">781K</td> <td style="text-align: center;">625K</td> <td style="text-align: center;">500K</td> </tr> </table> <p>・ SCLS0 = 0の場合</p> <p style="margin-left: 20px;">シリアル・クロック・ソースとして、ポー・レート・ジェネレータ出力が選択されます。ポー・レート・ジェネレータの詳細は、9.3 ポー・レート・ジェネレータを参照してください。</p>		33 MHz	25 MHz	20 MHz	16 MHz	ポー・レート (bps)	1031K	781K	625K	500K
	33 MHz	25 MHz	20 MHz	16 MHz								
ポー・レート (bps)	1031K	781K	625K	500K								

(2) アシクロナス・シリアル・インタフェース・モードレジスタ01 (ASIM01)

UARTの転送モードを指定します。8ビット単位でリード/ライトできます。

図9-3 アシクロナス・シリアル・インタフェース・モードレジスタ01 (ASIM01)



(3) アシクロナス・シリアル・インタフェース・ステータス・レジスタ (ASIS0)

UARTの受信終了時のエラー・ステータスを示す3ビットのエラー・フラグと、送信ステータス・フラグで構成されるレジスタです。8ビット単位でリードだけです。

受信エラーが発生した場合は、このレジスタの内容を読み出したあと、受信バッファRXB0またはRXB0Lの内容を読み出し、エラー・フラグを0にクリアしてください。

受信エラーを示すステータス・フラグは、常に最も新しく発生したエラーの状態を示しています。つまり、受信データの読み出し前に複数回の同一エラーが発生した場合、最後に発生したエラーの状態だけを保持します。

図9-4 アシクロナス・シリアル・インタフェース・ステータス・レジスタ (ASIS0)

アドレス: C0000094H								
	7	6	5	4	3	2	1	0
ASIS0	SOT0	0	0	0	0	PE0	FE0	OVE0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

ビット	ビット名	説明
7	SOT0	Status Of Transmission 送信動作状態を示すステータス・フラグです。 1: 送信開始タイミング 0: 送信終了タイミング
2	PE0	Parity Error パリティ・エラーを示すステータス・フラグです。 1: 送信パリティと受信パリティが一致しないとき 0: 受信バッファからのデータ読み出し処理
1	FE0	Framing Error フレーミング・エラーを示すステータス・フラグです。 1: ストップ・ビットが検出されないとき 0: 受信バッファからのデータ読み出し処理
0	OVE0	Overrun Error オーバラン・エラーを示すステータス・フラグです。 1: 受信バッファから受信データを引き取る前にUARTが次の受信処理を終了したとき 0: 受信バッファからの受信データ読み出し処理 なお、1フレームの受信ごとに受信シフト・レジスタの内容が受信バッファに転送される構成のため、オーバラン・エラーが発生したときには、受信バッファに次のデータが上書きされ、1回前の受信データは保持されません。

(4) 受信バッファ (RXB0, RXB0L)

RXB0は、受信データを保持する9ビットのバッファ・レジスタです。7, 8ビット/キャラクタの受信では上位に0が格納されます。

このレジスタへのハーフワード (16ビット) アクセス時はRXB0を、バイト・アクセス時はRXB0Lを指定します。リードだけです。

受信許可状態中は、受信データは1フレーム分のシフト・イン処理終了に同期して受信シフト・レジスタから受信バッファに転送されます。また、受信バッファへの転送により、受信終了割り込み要求 (INTSR) が発生します。

受信禁止状態中は、1フレーム分のシフト・イン処理が終了しても受信バッファへの転送処理は行われず、受信バッファの内容は保持されます。また、受信終了割り込み要求 (INTSR) も発生しません。

RXE0は、拡張ビットです。ASIM01レジスタで拡張ビット動作許可を指定したときは、RXE0ビットに拡張ビットが格納されます。また、拡張ビット動作禁止を指定したときは、0が格納されます。

図9-5 受信バッファ (RXB0, RXB0L)



(5) 送信シフト・レジスタ (TXS0, TXS0L)

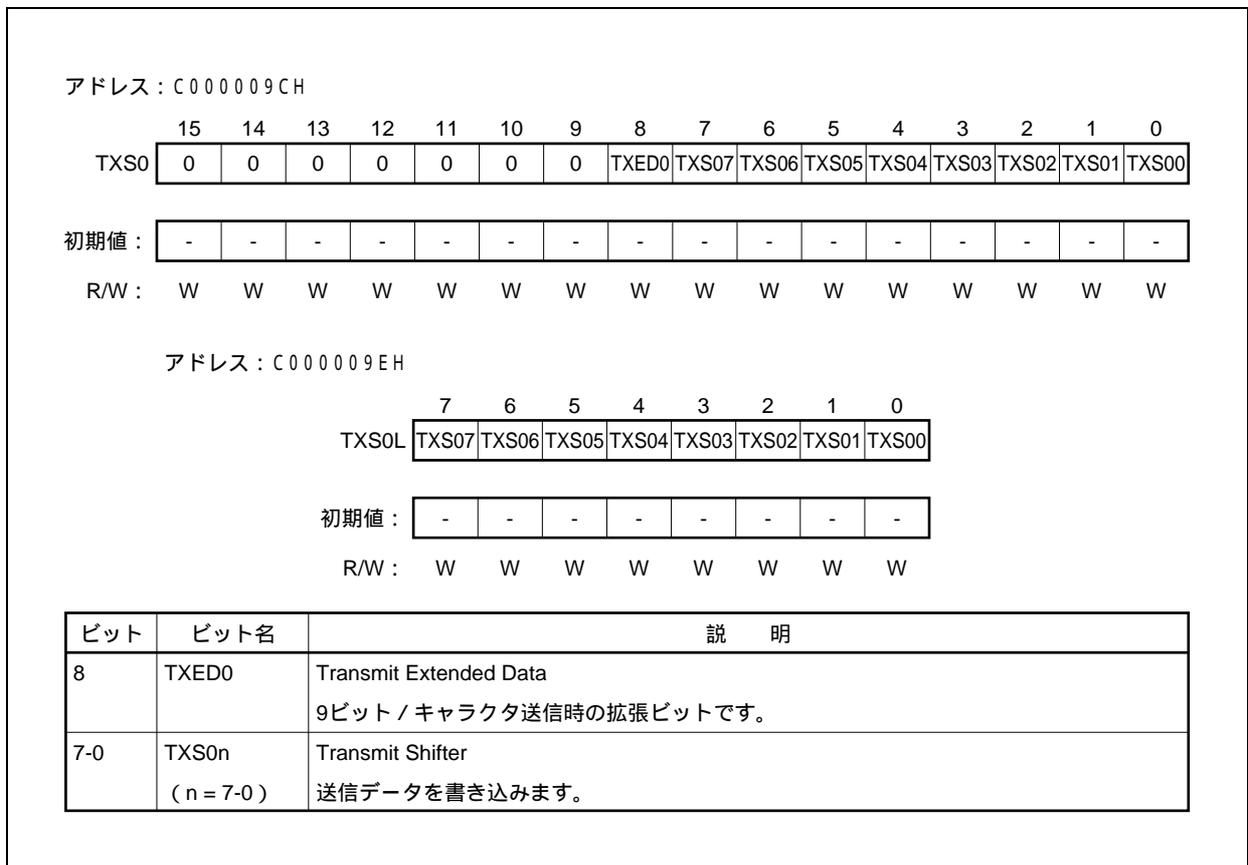
TXS0は、9ビットの送信処理用シフト・レジスタです。このレジスタへのデータ書き込みにより、送信動作を開始します。

V831のUARTは送信バッファがないため、送信終了（バッファへの転送終了）による割り込み要求ではなく、TXS0のデータを含む1フレームの送出終了に同期して送信終了割り込み要求（INTST）が発生します。

このレジスタへのハーフワード（16ビット）アクセス時はTXS0を、バイト・アクセス時はTXS0Lを指定します。

TXED0は、拡張ビットです。ASIM01レジスタで拡張ビット動作許可を指定したときは、TXED0ビットに拡張ビットが格納されます。また、拡張ビット動作禁止を指定したときは、0が格納されます。

図9 - 6 送信シフト・レジスタ (TXS0, TXS0L)



9.1.5 割り込み要求

UARTからは次の3種類の割り込み要求が発生します。

(1) 受信エラー割り込み (INTSER)

受信許可状態中で3種類の受信エラーの論理和で受信エラー割り込み要求が発生します (図9-4 アシ
ンクロナス・シリアル・インタフェース・ステータス・レジスタ (ASIS0) 参照)。

受信禁止状態中は、受信エラー割り込み要求は発生しません。

(2) 受信終了割り込み (INTSR)

受信許可状態中で、受信シフト・レジスタにデータがシフト・インされ受信バッファに転送されると受
信終了割り込み要求が発生します。受信終了割り込み要求は、受信エラーが起こった場合にも発生します。

受信禁止状態中は、受信終了割り込み要求は発生しません。

(3) 送信終了割り込み (INTST)

V831のUARTは送信バッファがないため、送信シフト・レジスタから7ビット/8ビット/9ビットのキ
ャラクタを含む1フレーム分の送信データがシフト・アウトされると、送信終了割り込み要求が発生しま
す。送信終了割り込み要求は、送信データの最終ビットの送信開始時に出力されます。

送信終了割り込みと受信終了割り込みを使用して、DMA転送ができます (第8章 DMA機能を参照)。

9.1.6 基本動作

(1) 送信

(a) 送信許可状態

UARTは、常に送信許可状態になっています。また、V831のUARTはCTS（シリアル送信制御）入力端子がないので、相手側が受信状態であるかを確認する場合は、汎用入力ポートを使用してください。

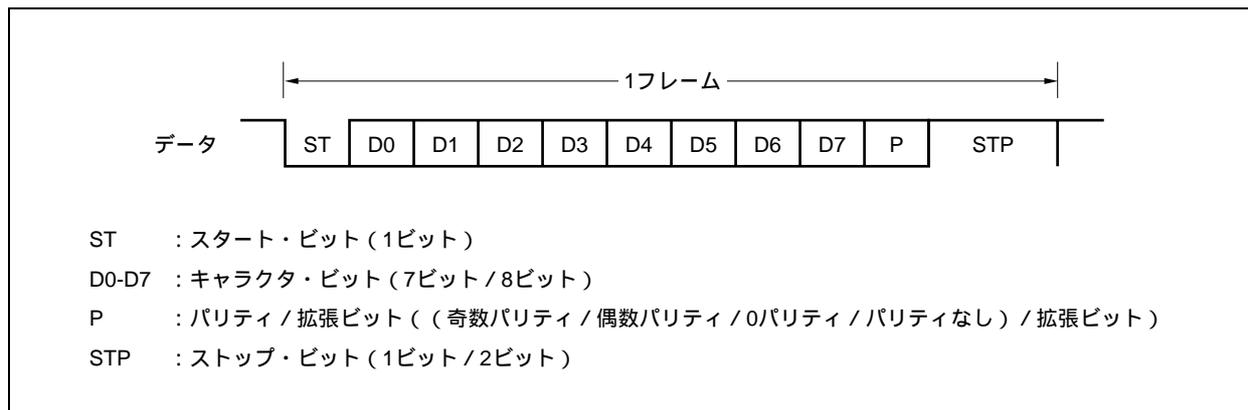
(b) 送信動作の起動

送信シフト・レジスタ（TXS0, TXS0L）にデータを書き込むと送信動作を起動します。

(c) 送信データのフォーマット

送信データ・フォーマットを図9-7に示します。スタート・ビット、キャラクタ・ビット、パリティ・ビット、ストップ・ビットで1フレームを構成します

図9-7 UARTの送信データ・フォーマット



(d) 送信割り込み要求

1フレーム分のデータの送出が終了すると送信終了割り込み要求が発生します。

注意 TXS0が空の状態は、送信終了割り込みの要因ではありません。1フレーム分の送出終了が送信終了割り込みを発生する要因となります。したがって、リセット解除後にTXS0が空であるために送信終了割り込みが発生することはありません。

(2) 受信

(a) 受信許可状態

受信動作は、ASIM00レジスタのRXE0ビットをセット（1）すると許可状態になります。

- ・ RXE0 = 1 (受信許可状態)
- ・ RXE0 = 0 (受信禁止状態)

受信禁止状態では、受信ハードウェアは初期状態で待機します。このとき、受信終了割り込み要求 / 受信エラー割り込み要求は発生しないで、受信バッファの内容は保持されます。

(b) 受信動作の起動

受信動作は、スタート・ビットの検出により起動されます。

ボ・・レ・ト・ジェネレ・タまたはバス・クロックからのシリアル・クロックで、RXD端子をサンプリングします。RXD端子の立ち下がりエッジを検出してから8シリアル・クロック後、再びRXD端子をサンプリングし、ロウ・レベルを確認するとスタート・ビットと認識して受信処理動作に移り、以降16シリアル・クロック単位にRXD端子入力をサンプリングします。

RXD端子の立ち下がりエッジを検出してから8シリアル・クロック後のサンプリングで、ハイ・レベルを確認すると、この立ち下がりエッジをスタート・ビットと認識せず、サンプル・タイミング発生用のシリアル・クロック・カウンタは初期化されて動作を停止し、次の立ち下がりエッジ入力を待ちます。

(c) 受信終了割り込み要求

受信許可状態 ($RXE0 = 1$) のとき、1フレーム分のデータの受信が終了すると、シフト・レジスタ内の受信データがRXB0に転送され、受信終了割り込み要求が発生します。

受信禁止状態 ($RXE0 = 0$) では、受信終了割り込み要求は発生しません。

(d) 受信エラー・フラグ

受信動作に同期して、パリティ・エラー、フレーミング・エラー、オーバラン・エラーの3種類のエラー・フラグ (ASIS0レジスタの0-2ビット) が影響を受けます。この3つの各エラー・フラグの論理和で、受信エラー割り込み要求が発生します。

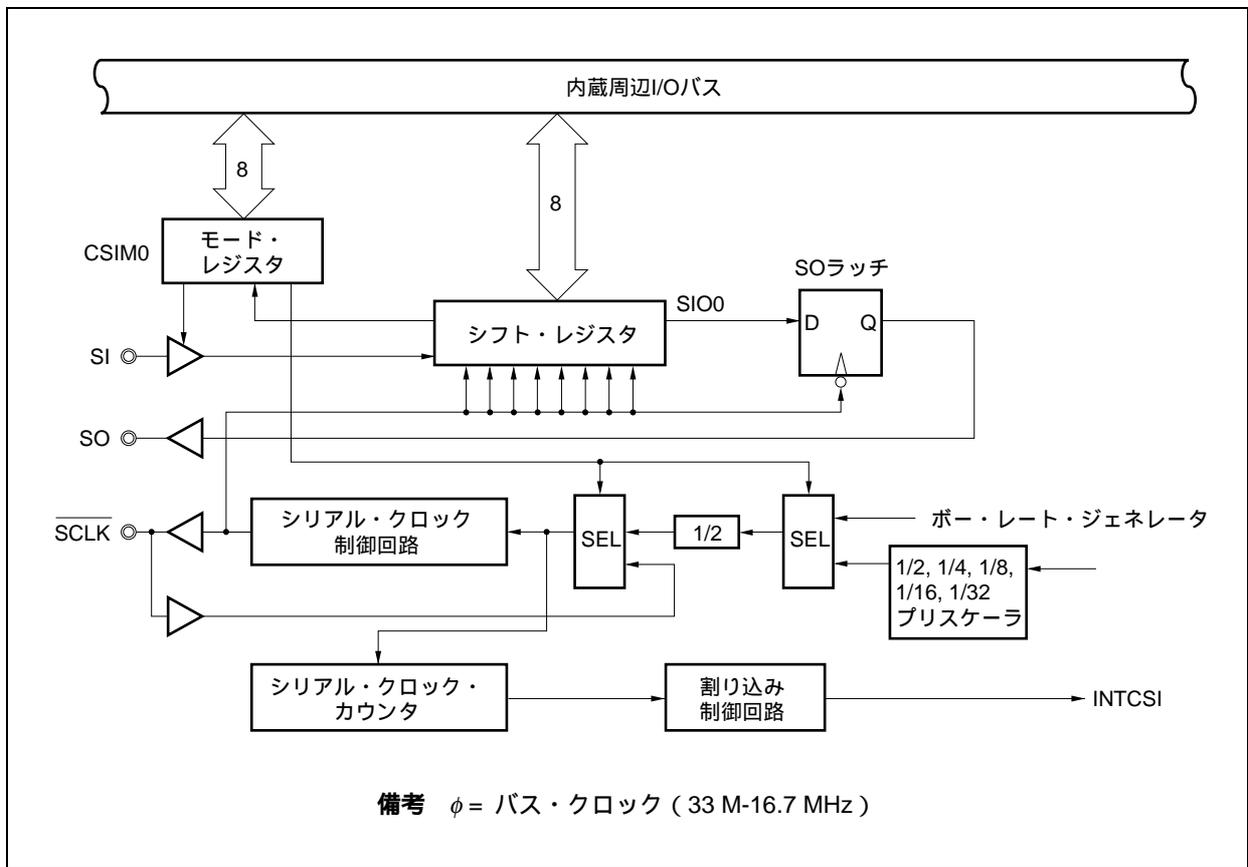
9.2 クロック同期式シリアル・インタフェース (CSI)

9.2.1 特徴

- 高速転送 最大8.25 Mbps (バス・クロック : 33 MHz)
- 送受信は同時で半二重通信可能 (バッファは備えていません)
- キャラクタ長 : 8ビット
- 外部/内部シリアル・クロック選択

9.2.2 構成

図9-8 CSIのブロック図



9.2.3 モード・レジスタとコントロール・レジスタ

(1) クロック同期式シリアル・インタフェース・モード・レジスタ0 (CSIM0)

CSIの基本動作モードを指定します。8ビット単位でリード/ライトできます(ただし、CSOT0(ビット5)はリードだけ)。

図9-9 クロック同期式シリアル・インタフェース・モード・レジスタ0 (CSIM0) (1/2)



図9 - 9 クロック同期式シリアル・インタフェース・モードレジスタ0 (CSIM0) (2/2)

アドレス : C00000A0H

	7	6	5	4	3	2	1	0
CSIM0	CTXE0	CRXE0	CSOT0	MOD0	0	CLS02	CLS01	CLS00

初期値 :

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

R/W : R/W R/W R R/W R R/W R/W R/W

ビット	ビット名	説明																																																				
2-0	CLS02-CLS00	<p>Clock Source</p> <p>シリアル・クロックを指定します。</p> <table border="1" style="margin-left: 20px; border-collapse: collapse;"> <thead> <tr> <th>CLS02</th> <th>CLS01</th> <th>CLS00</th> <th>シリアル・クロックの指定</th> <th>SCLK端子</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>外部クロック</td> <td>入力</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>内部クロック</td> <td>BPRM0レジスタで指定^{注1}</td> <td>出力</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td></td> <td>予約 (設定禁止)</td> <td>-</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td></td> <td>/4^{注2}</td> <td>出力</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td></td> <td>/8^{注2}</td> <td>出力</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td></td> <td>/16^{注2}</td> <td>出力</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td></td> <td>/32^{注2}</td> <td>出力</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td></td> <td>/64^{注2}</td> <td>出力</td> </tr> </tbody> </table> <p>注1. BPRM0レジスタの設定については、9.3 ボー・レート・ジェネレータを参照してください。</p> <p>2. /4, /8, /16, /32, /64は分周信号です。 =バス・クロック (16-33 MHz)</p>	CLS02	CLS01	CLS00	シリアル・クロックの指定	SCLK端子	0	0	0	外部クロック	入力	0	0	1	内部クロック	BPRM0レジスタで指定 ^{注1}	出力	0	1	0		予約 (設定禁止)	-	0	1	1		/4 ^{注2}	出力	1	0	0		/8 ^{注2}	出力	1	0	1		/16 ^{注2}	出力	1	1	0		/32 ^{注2}	出力	1	1	1		/64 ^{注2}	出力
CLS02	CLS01	CLS00	シリアル・クロックの指定	SCLK端子																																																		
0	0	0	外部クロック	入力																																																		
0	0	1	内部クロック	BPRM0レジスタで指定 ^{注1}	出力																																																	
0	1	0		予約 (設定禁止)	-																																																	
0	1	1		/4 ^{注2}	出力																																																	
1	0	0		/8 ^{注2}	出力																																																	
1	0	1		/16 ^{注2}	出力																																																	
1	1	0		/32 ^{注2}	出力																																																	
1	1	1		/64 ^{注2}	出力																																																	

(2) シリアルI/Oシフト・レジスタ0 (SIO0)

パラレル・データをシリアル・データに、シリアル・データをパラレル・データに変換するレジスタです。CTXE0 = 1またはCRXE0 = 1のときにシフト動作を行います。8ビット単位でリード/ライトできません。

図9 - 10 シリアルI/Oシフト・レジスタ0 (SIO0)

アドレス : C00000A2H

	7	6	5	4	3	2	1	0
SIO0	SIO07	SIO06	SIO05	SIO04	SIO03	SIO02	SIO01	SIO00

初期値 :

-	-	-	-	-	-	-	-
---	---	---	---	---	---	---	---

R/W : R/W R/W R/W R/W R/W R/W R/W R/W

ビット	ビット名	説明
7-0	SIO07-SIO00	<p>Serial I/O</p> <p>データはMSBまたはLSB側からシフト・イン (受信) またはシフト・アウト (送信) します。</p>

9.2.4 端子機能

クロック同期式シリアル・インタフェース (CSI) は、次に示す端子を使用します。これらの端子は、入出力ポートと兼用端子になっています。初期状態では入出力ポートが選択されているため、CSIを使用する場合は、ポート・コントロール・モード・レジスタ (PC) のPC2-PC0ビットをCSIインタフェースに設定してください (11.2.3 ポート・コントロール・モード・レジスタ (PC) 参照)。

- ・ SO : シリアル・データ出力端子
- ・ SI : シリアル・データ入力端子
- ・ SCLK : シリアル・クロック入出力端子 (モード切り換えは、CSIM0レジスタで指定してください。)

9.2.5 基本動作

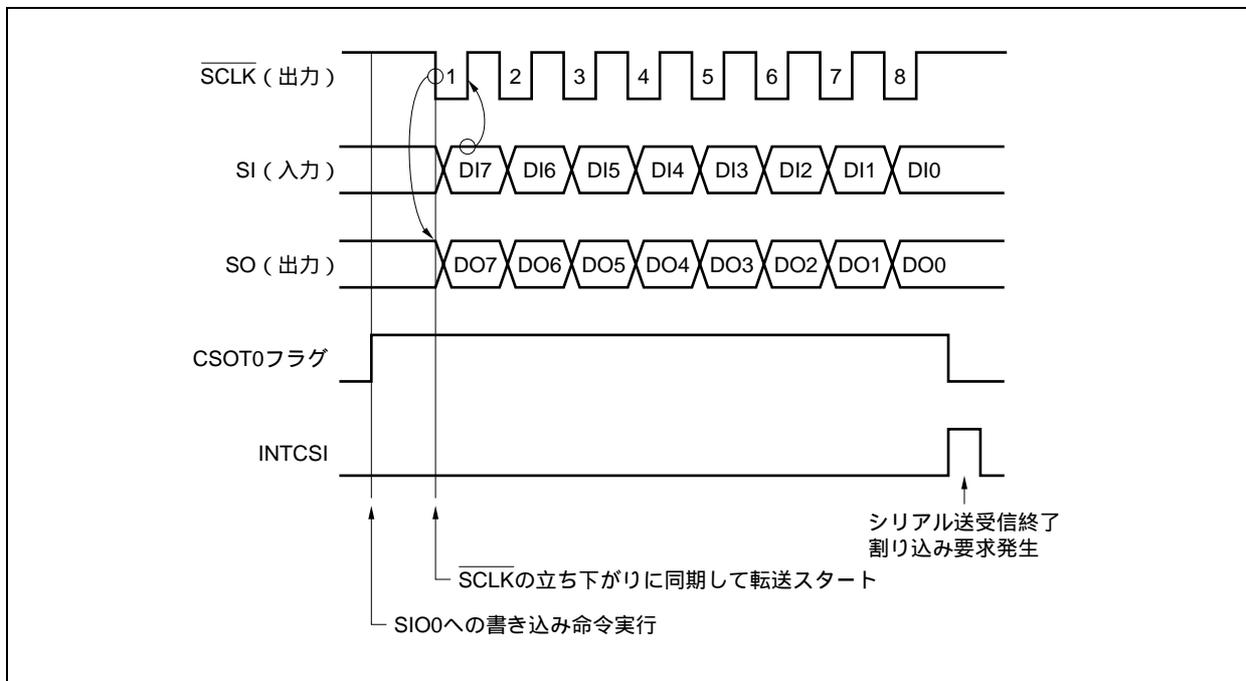
(1) 転送フォーマット

V831のCSIでは、1本のクロック・ラインと2本のデータ・ラインの3線でインタフェースします。シリアル転送は、SIO0レジスタに対する転送データの書き込み命令を実行するとスタートします。

送信の場合は、SCLK信号の立ち下がりに同期してSO端子からデータを出力します。受信の場合は、SCLK信号の立ち上がりに同期してSI端子の入力データをラッチします。

シリアル・クロック・カウンタのオーバフロー (8カウント目のSCLK信号の立ち上がり) で、SCLK信号は停止して、割り込み要求信号 (INTCSI) が発生します。

図9 - 11 CSIの転送タイミング



(2) 送受信許可

V831のCSIは、8ビット・シフト・レジスタが1つだけあり、バッファがないため、送信と受信は同時に行います。

(a) 送受信許可条件

CTXE0 = 1の場合、送信を許可します。

CRXE0 = 1の場合、受信を許可します。

CTXE0 = CRXE0 = 1の場合、送受信を許可します。

(i) CTXE0によるSIO出力の禁止

CTXE0 = 0の場合、シリアル出力はハイ・インピーダンスになります。

CTXE0 = 1の場合、シフト・レジスタのデータを出力します。

(ii) CRXE0によるSIO入力禁止

CRXE0 = 0の場合、シフト・レジスタ入力は0になります。

CRXE0 = 1の場合、シリアル入力がシフト・レジスタに入力されます。

(iii) 送信データのチェックを行う場合

SI信号とSO信号を接続し2線式で使用する場合、送信データを自分自身で受信し、バスの競合が発生していないかチェックをするために、CTXE0 = CRXE0 = 1にします。

(b) 送受信動作の起動

送受信動作の起動（スタート）は、シフト・レジスタに対するリード/ライトにより行います。送信許可ビット（CTXE0）、受信許可ビット（CRXE0）を次に示すように設定すると送信、受信のスタート制御を行います。

表9 - 1 スタート条件

CTXE0	CRXE0	スタート条件
0	0	スタートしない
0	1	シフト・レジスタ読み出し
1	0	シフト・レジスタ書き込み
1	1	シフト・レジスタ書き込み
0	0 1	CRXE0ビット書き換え

CTXE0 = 0のとき、シフト・レジスタをライトしたあとCRXE0ビットを1にしても転送はスタートしません。CTXE0が0のとき、CRXE0ビット = 0からCRXE0ビット = 1にするとシリアル・クロックが発生し、受信動作を開始します。

CSIM0レジスタのCSOT0ビット = 1（転送動作中）のときは、スタート条件を受け付けません。

★

注意 シリアル・クロックとして外部クロック入力を使用時、一つのデータの送受信終了前に外部から入力しているシリアル・クロックが停止した場合、V831のCSIは次のシリアル・クロック入力を中断データの続きとして判断します（シリアル・クロック・カウンタはクリアされません）。

そのデータの送受信を中断し、再度やり直したい場合、次の手順でCSIの初期化を行ってください。

- ・ CTXE, CRXEビットをクリアします（送受信を一旦禁止）。
- ・ CTXE, CRXEビットをセットします（送受信を再び許可）。
- ・ 送信のみ、受信のみの場合は対応する禁止/許可ビットのみクリア セットしてください。

9.3 ボー・レート・ジェネレータ

9.3.1 構成と機能

シリアル・インタフェースでは、シリアル・クロックをボー・レート・ジェネレータ出力または ϕ （バス・クロック）の分周値から選択したものをボー・レートとして使用できます。

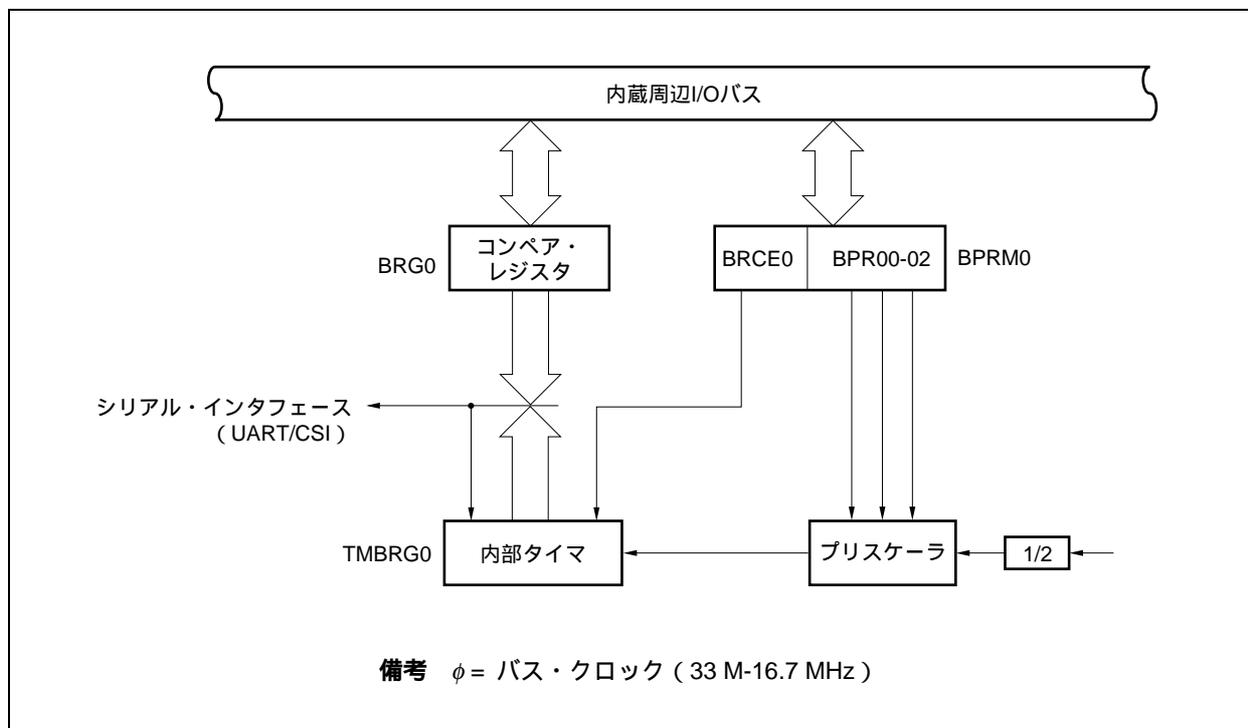
シリアル・クロック・ソースは、次に示すレジスタで指定してください。

- ・ UARTの場合、ASIM00レジスタのSCLS0ビットで指定（図9-2参照）
- ・ CSIの場合、CSIM0レジスタのCLS02-CLS00ビットで指定（図9-9参照）

ボー・レート・ジェネレータ出力を指定した場合は、クロック・ソースとしてボー・レート・ジェネレータ（BRG）が選択されます。

ボー・レート・ジェネレータは、UARTとCSIで共用しています。

図9-12 ボー・レート・ジェネレータ（BRG）のブロック構成



ボー・レート・ジェネレータは、それぞれ送受信のシフト・クロックを発生する専用の8ビット・タイマ (TMBRG0)、コンペア・レジスタ (BRG0)、モード・レジスタ (BPRM0) とプリスケラで構成しています。

(1) 入力クロック

BRGへは、バス・クロック (ϕ) が入力されます。

(2) BRGへの設定

(a) UART

UARTにてボー・レート・ジェネレータを指定した場合、 $\times 16$ のサンプリング・レートを使用的るので、実際のボー・レートは次に示す式で表されます。

$$\text{ボー・レート} = \frac{\phi}{2 \times m \times 2^n \times 16 \times 2} \text{ [bps]}$$

= バス・クロック周波数 [Hz]

m = BRG0レジスタ設定値 (1 m 256[#])

n = プリスケラ設定値 (BPRM0レジスタのBPR00-BPR02) (n = 0, 1, 2, 3, 4)

注 m = 256の設定値は、BRG0レジスタへの0ライトで行います。

(b) CSI

CSIにてボー・レート・ジェネレータを指定した場合、実際のボー・レートは次に示す式で表されます。

$$\text{ボー・レート} = \frac{\phi}{2 \times m \times 2^n \times 2} \text{ [bps]}$$

= バス・クロック周波数 [Hz]

m = BRG0レジスタ設定値 (1 m 256[#])

n = プリスケラ設定値 (BPRM0レジスタのBPR00-BPR02) (n = 0, 1, 2, 3, 4)

注 m = 256の設定値は、BRG0レジスタへの0ライトで行います。

代表的クロックを使用したときのボー・レート・ジェネレータの設定値を表9 - 2に示します。

表9-2 BRG設定データ

ボー・レート [bps]		φ = 33 MHz			φ = 25 MHz			φ = 20 MHz			φ = 16 MHz		
UART	CSI	BPR	BRG0	誤差	BPR	BRG0	誤差	BPR	BRG0	誤差	BPR	BRG0	誤差
110	1760	-	-	-	4	222	0.02 %	4	178	0.25 %	4	142	0.03 %
150	2400	4	215	0.07 %	4	163	0.15 %	4	130	0.16 %	3	208	0.16 %
300	4800	3	215	0.07 %	3	163	0.15 %	3	130	0.16 %	2	208	0.16 %
600	9600	2	215	0.07 %	2	163	0.15 %	2	130	0.16 %	1	208	0.16 %
1200	19200	1	215	0.07 %	1	163	0.15 %	1	130	0.16 %	0	208	0.16 %
2400	38400	0	215	0.07 %	0	163	0.15 %	0	130	0.16 %	0	104	0.16 %
4800	76800	0	107	0.39 %	0	81	0.47 %	0	65	0.16 %	0	52	0.16 %
9600	153600	0	54	0.54 %	0	41	0.76 %	0	33	1.36 %	0	26	0.16 %
10400	166400	0	50	0.84 %	0	38	1.16 %	0	30	0.16 %	0	24	0.16 %
19200	307200	0	27	0.54 %	0	20	1.73 %	0	16	1.73 %	0	13	0.16 %
38400	614400	0	13	3.29 %	0	10	1.73 %	0	8	1.73 %	0	7	6.99 % ^注
76800	1228800	0	7	4.09 %	0	5	1.73 %	0	4	1.73 %	-	-	-
153600	2457600	0	3	11.9 % ^注	0	2	27.2 % ^注	0	2	1.73 %	-	-	-

注 誤差が大きくて使用できません。

備考 BPR : BPRM0レジスタのBPR00-BPR02ビット

(3) ボー・レート・ジェネレータの誤差

ボー・レート・ジェネレータの誤差は次のように表されます。

$$\text{誤差} [\%] = \left(\frac{\text{実際のボー・レート (誤差のあるボー・レート)}}{\text{希望するボー・レート (正常なボー・レート)}} - 1 \right) \times 100$$

$$\text{例} \quad \left(\frac{9520}{9600} - 1 \right) \times 100 = -0.833 [\%]$$

$$\left(\frac{5000}{4800} - 1 \right) \times 100 = +4.167 [\%]$$

(4) ボー・レート・ジェネレータの誤差許容範囲

ボー・レート・ジェネレータの許容範囲は、1フレームのビット数に依存します。

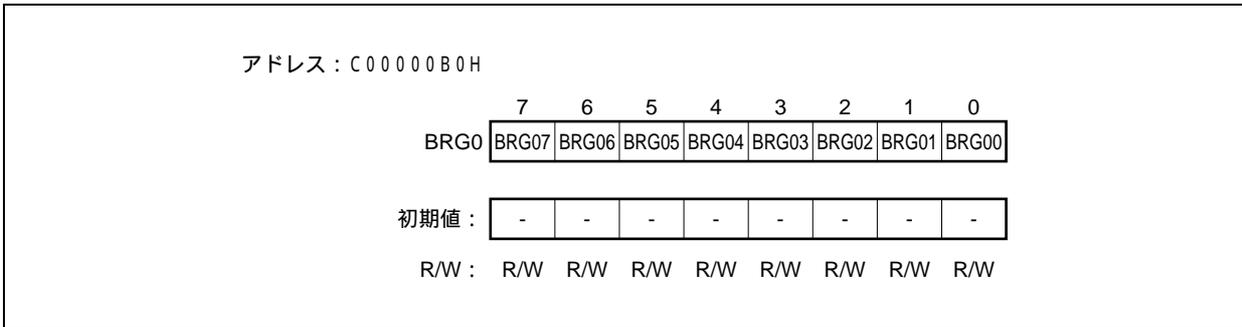
16ビットでのボー・レートの誤差：±5 %，サンプル・タイミング：±4.5 %を基本的な許容限度としています。ただし、実用上の許容限度は、送信側、受信側がともに誤差を含んでいる場合を想定して、ボー・レート誤差：±2.3 %になります。

9.3.2 ポー・レート・ジェネレータ・コンペア・レジスタ (BRG0)

ポー・レート・ジェネレータにおけるタイマ・カウント値を設定する8ビットのコンペア・レジスタです。8ビット単位でリード/ライトできます。

BRG0レジスタへの書き込みによって、内部タイマ (TMBRG0) がクリアされます。したがって、送受信動作中にソフトウェアで、このレジスタを書き換えることはできません。

図9-13 ポー・レート・ジェネレータ・コンペア・レジスタ (BRG0)



9.3.3 ポー・レート・ジェネレータ・プリスケアラ・モード・レジスタ (BPRM0)

専用ポー・レート・ジェネレータのタイマ・カウント動作制御とカウント・クロックの選択を行います。8ビット単位でリード/ライトできます。

図9-14 ポー・レート・ジェネレータ・プリスケアラ・モード・レジスタ (BPRM0)

アドレス : C00000B2H

	7	6	5	4	3	2	1	0
BPRM0	BRCE0	0	0	0	0	BPR02	BPR01	BPR00

初期値 :

0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---

R/W : R/W R R R R R/W R/W R/W

ビット	ビット名	説明																								
7	BRCE0	Baud Rate Generator Count Enable BRGのカウント動作を制御します。 0 : クリアされたままカウント動作を停止します。 1 : カウント動作を許可します。																								
2-0	BPR02-BPR00	Baud Rate Generator Prescaler TMBRG0へ入力するカウント・クロックを指定します。 <table border="1" style="margin: 5px auto; border-collapse: collapse;"> <thead> <tr> <th>BPR02</th> <th>BPR01</th> <th>BPR00</th> <th>動作</th> </tr> </thead> <tbody> <tr><td style="text-align: center;">0</td><td style="text-align: center;">0</td><td style="text-align: center;">0</td><td>/2 (n=0)</td></tr> <tr><td style="text-align: center;">0</td><td style="text-align: center;">0</td><td style="text-align: center;">1</td><td>/4 (n=1)</td></tr> <tr><td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td><td>/8 (n=2)</td></tr> <tr><td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td>/16 (n=3)</td></tr> <tr><td style="text-align: center;">1</td><td style="text-align: center;">x</td><td style="text-align: center;">x</td><td>/32 (n=4)</td></tr> </tbody> </table>	BPR02	BPR01	BPR00	動作	0	0	0	/2 (n=0)	0	0	1	/4 (n=1)	0	1	0	/8 (n=2)	0	1	1	/16 (n=3)	1	x	x	/32 (n=4)
BPR02	BPR01	BPR00	動作																							
0	0	0	/2 (n=0)																							
0	0	1	/4 (n=1)																							
0	1	0	/8 (n=2)																							
0	1	1	/16 (n=3)																							
1	x	x	/32 (n=4)																							

備考 n : プリスケアラ設定値, : バス・クロック

注意 送受信動作中はカウント・クロックを変更できません。

(メ モ)

第10章 タイマ/カウンタ機能

10.1 特 徴

パルス間隔や周波数の計測および、プログラマブルなパルスを出力

- ・ 16ビット計測可能
- ・ 多彩な形状のパルスを発生可能 (インターバル・パルス, ワンショット・パルス)

タイマ1

- ・ 16ビット・タイマ/イベント・カウンタ
- ・ カウント・クロックのソース: 2種 (システム・クロックの分周を選択, 外部パルス入力)
- ・ キャプチャ/コンペア共用レジスタ: 4本
- ・ カウント・クリア端子: TCLR
- ・ 割り込みソース: 5種
- ・ 外部パルス出力: 2本

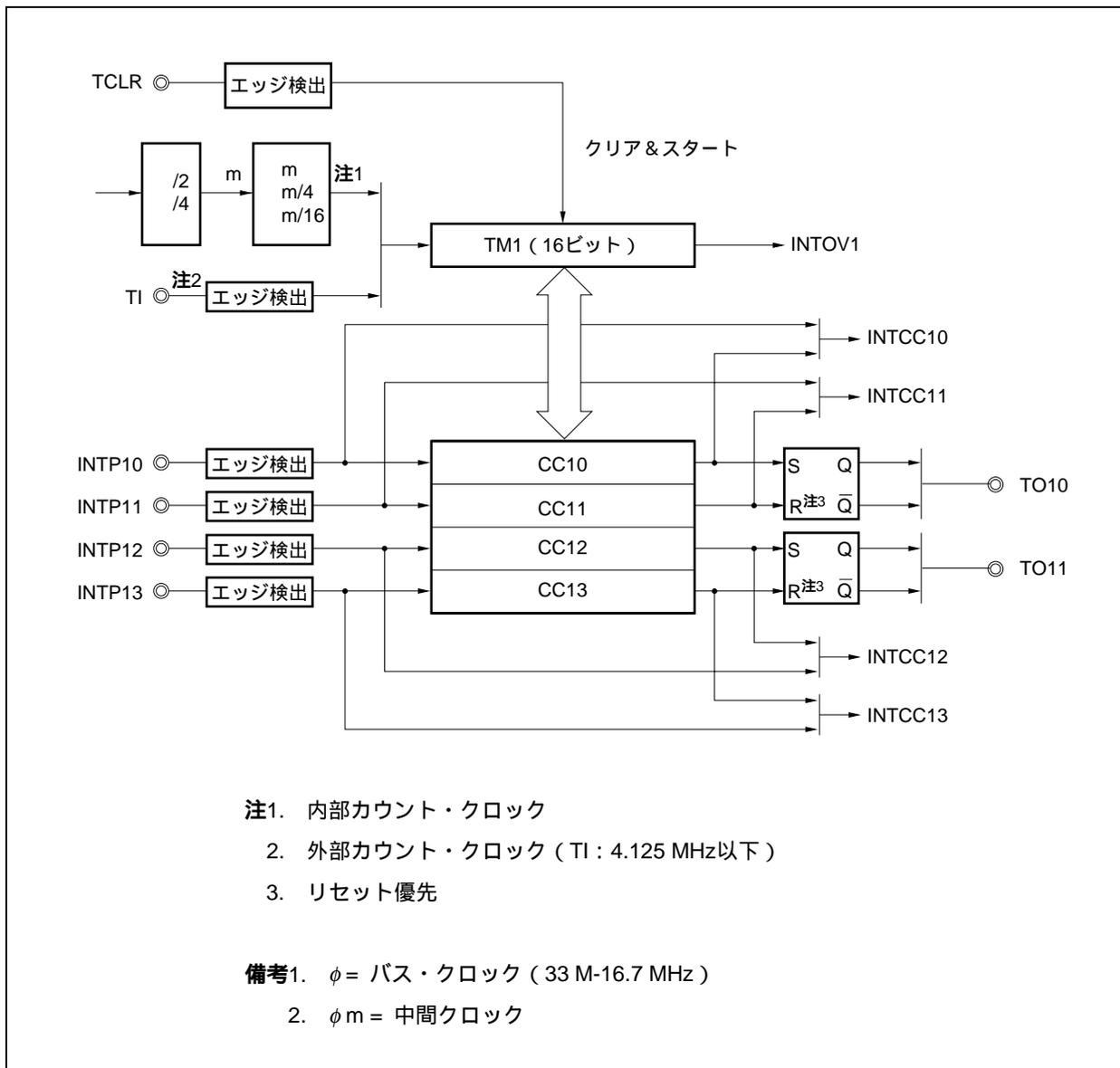
タイマ4

- ・ 16ビット・インターバル・タイマ
- ・ カウント・クロックは, システム・クロックの分周から選択
- ・ コンペア・レジスタ: 1本
- ・ 割り込みソース: 1種

10.2 構成

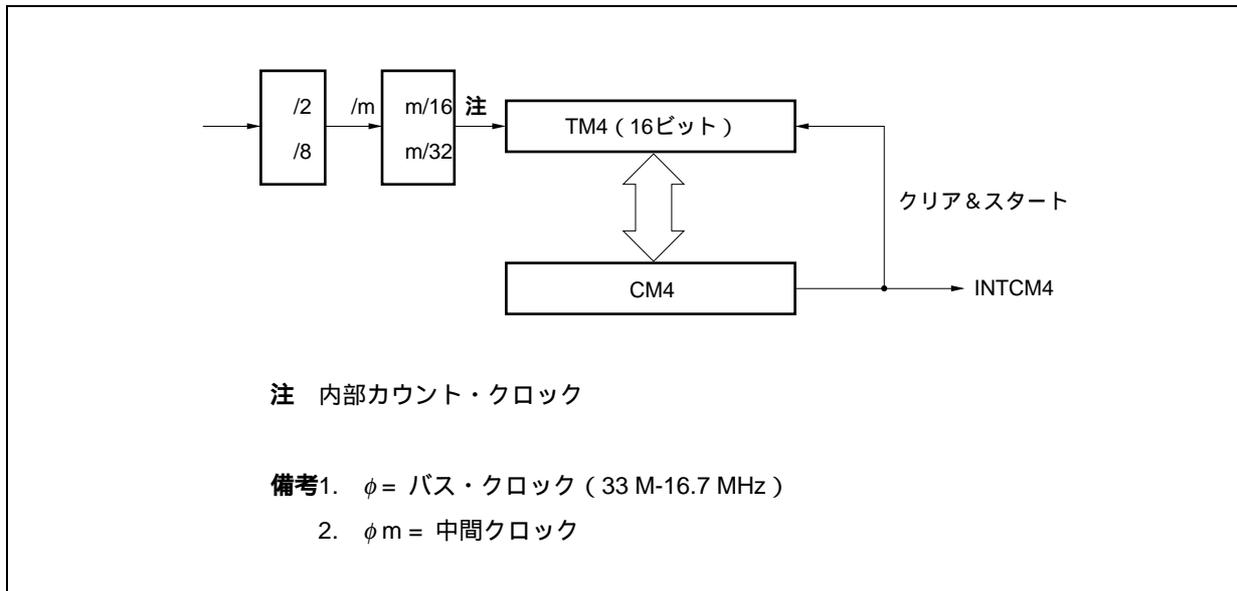
(1) タイマ1 (16ビット・タイマ/イベント・カウンタ)

図10-1 タイマ1のブロック構成



(2) タイマ4 (16ビット・インターバル・タイマ)

図10-2 タイマ4のブロック構成

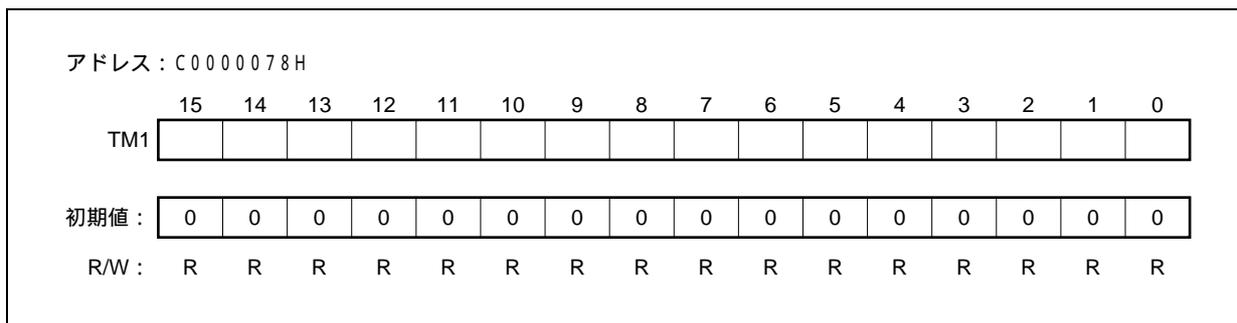


10.2.1 タイマ1

(1) タイマ1 (TM1)

タイマ1は、16ビット・タイマ/イベント・カウンタとして機能します。おもに、周期計測、または周波数計測のほか、パルス出力としても利用できます。TM1レジスタは、16ビット単位でリードだけです。

図10-3 タイマ1 (TM1)



タイマのスタート/ストップは、タイマ・コントロール・レジスタ1 (TMC1) によって制御します (図10-8 タイマ・コントロール・レジスタ1 (TMC1) 参照)。

カウント・クロックの内部/外部の選択は、TMC1レジスタで行います。

(a) 外部カウント・クロックを選択

TM1はイベント・カウンタとして動作します。タイマ・ユニット・モード・レジスタ (TUM1) によって有効エッジを指定し、TI端子入力によりTM1をカウント・アップします。

(b) 内部カウント・クロックを選択

TM1はフリー・ランニング・タイマとして動作します。TMC1レジスタで指定される内部クロック ($\phi/2 \sim \phi/64$) によりTM1をカウント・アップします。

内部クロックを選択するときは、前段のプリスケアラによって ϕm (中間クロック) が $\phi/2$, $\phi/4$ の2種類から選択され、次に後段のプリスケアラによって最終的なカウント・クロックが ϕm , $\phi m/4$, $\phi m/16$ の3種類から選択され、合計6通りのカウント・クロックを指定できます。

タイマがオーバーフローするとオーバーフロー割り込み (INTOV1) を発生できます。また、TUM1レジスタの指定により、オーバーフロー後タイマを停止できます。

TUM1レジスタの設定により、外部入力信号 (TCLR) によってタイマをクリアし、スタートできます。このとき、プリスケアラも同時にクリアされるので、外部入力信号 (TCLR) からの最初のタイマ・カウント・アップまでの時間は、プリスケアラの分周比に応じて一定となります。

注意 タイマ動作中はカウント・クロックを変更できません。

(c) キャプチャ/コンペア・レジスタの設定**キャプチャ・レジスタに設定した場合**

外部入力信号 (INTP10-INTP13) で割り込み信号 (INTCC10-INTCC13) を発生します。また、INTP10-INTP13の有効エッジは外部割り込みモード・レジスタ (IMOD) の設定により、立ち上がりエッジ、立ち上がりと立ち下りの両エッジから選択できます (図4-10 ICUモード・レジスタ (IMOD) 参照)。

コンペア・レジスタに設定した場合

TUM1レジスタの設定により、コンペア・レジスタの一致信号で割り込み信号 (INTCC10-INTCC13) を発生できます (図10-7 タイマ・ユニット・モード・レジスタ (TUM1) 参照)。

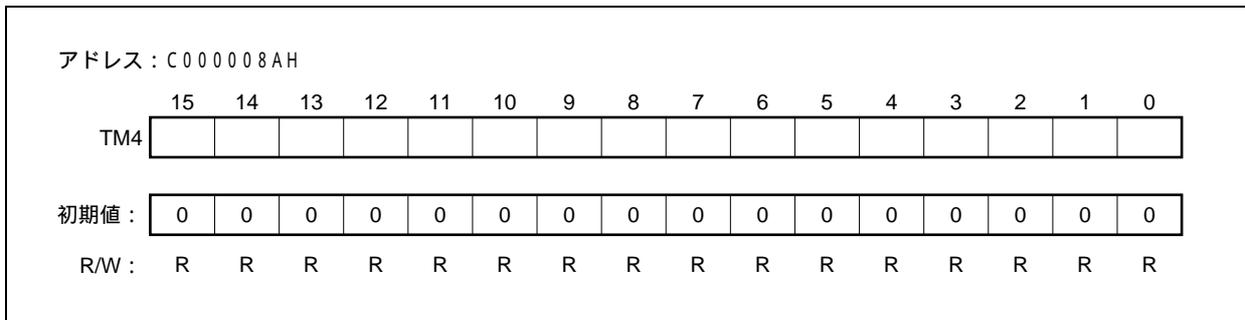
注意 タイマ動作中にモードを変更しないでください。

10.2.2 タイマ4

(1) タイマ4 (TM4)

タイマ4は、16ビット・タイマとして機能します。おもに、ソフトウェアのためのインターバル・タイマとして利用できます。TM4レジスタは、16ビット単位でリードだけできます。

図10-4 タイマ4 (TM4)



タイマのスタートおよびストップは、タイマ・コントロール・レジスタ (TMC4) で制御します。TMC4レジスタの指定により次に示すカウント・クロックに選択できます。

まず前段のプリスケラにより、 ϕ_m (中間クロック) を $\phi/2$, $\phi/8$ の2種類から選択します。次に後段のプリスケラによりカウント・クロックを $\phi_m/16$, $\phi_m/32$ の2種類から選択します。合計で4通りのカウント・クロックを指定できます。

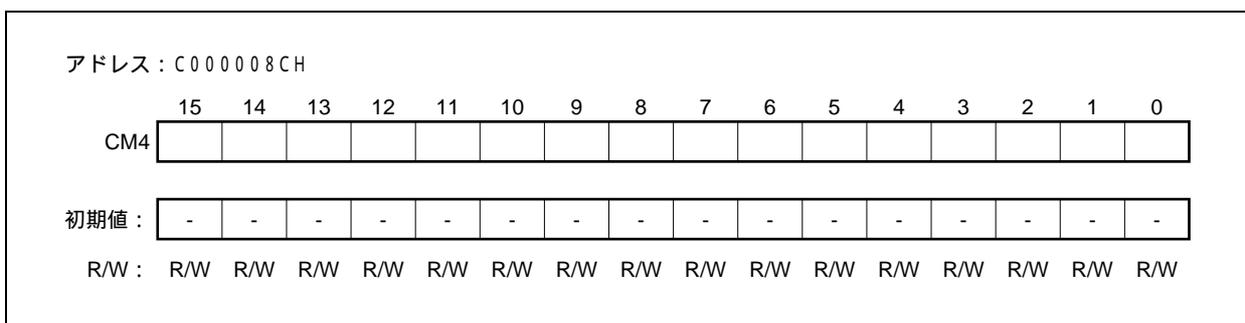
コンペア・レジスタの一致信号で、TM4をクリアして0からのカウント・アップを再開します。

- 注意1. TM4をインターバル動作で使用する場合、コンペア一致が発生したあと、タイマを次のカウント・クロックでクリアするため、分周比が大きい場合は一致割り込み要求発生直後にタイマの値を読み出すとタイマの値が0でないことがあります。
- 2. タイマ動作中はカウント・クロックを変更できません。

(2) コンペア・レジスタ (CM4)

CM4は16ビットのレジスタで、TM4に接続しています。16ビット単位でリード/ライトできます。

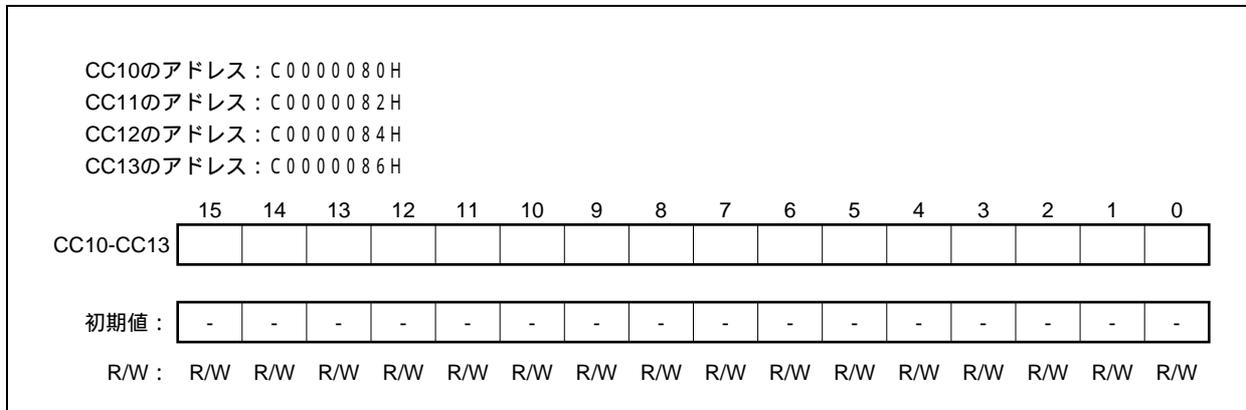
図10-5 コンペア・レジスタ (CM4)



10.2.3 キャプチャ/コンペア・レジスタ (CC10-CC13)

キャプチャ/コンペア・レジスタ (CC10-CC13) は、TUM1レジスタの指定によりキャプチャ・レジスタまたはコンペア・レジスタとして使用できます。CC10-CC13は、16ビット単位でリード/ライトできます。

図10-6 キャプチャ/コンペア・レジスタ (CC10-CC13)



コンペア・レジスタは、接続したタイマのカウント・クロックごとにタイマのカウント値と比較するために使用します。一致すると一致信号が発生します。一致信号によって、割り込み要求が発生できます。

各コンペア・レジスタはセット/リセット出力機能があります。一致信号の発生に同期して、対応するタイマ出力をセットまたはリセットします。

キャプチャ・レジスタは、対応する割り込み要求信号入力端子 (INTP10-INTP13) において有効エッジを検出すると、これをキャプチャ・トリガとして接続したタイマの値をラッチします (カウント・クロックとは非同期)。

(1) キャプチャ・レジスタを指定したとき

キャプチャ・レジスタを指定したときは、入力端子 (INTP1n) の有効エッジ検出で割り込み要求が発生します。このとき、コンペア・レジスタの一致信号であるINTCC1nによって割り込み要求を発生させることはできません (n=0-3)。

キャプチャ・レジスタへのキャプチャ (ラッチ) タイミングとレジスタへの命令による書き込み動作が競合した場合は、レジスタへの命令による書き込み動作を優先して、キャプチャ動作を無視します。

(2) コンペア・レジスタを指定したとき

コンペア・レジスタを指定した場合、TUM1レジスタの指定により、一致信号であるINTCC1nまたは入力端子 (INTP1n) の有効エッジ検出のいずれかを選択して割り込み要求信号にできます。INTP1nを選択した場合、タイマ出力の指定と並行して、外部割り込み要求の受け付けができます (n=0-3)。

指定したレジスタに設定している値に対してTM1のカウント値との比較を行います。コンペア・レジスタに指定したCC1nレジスタにはすべて任意の値を設定してください。4つのCC10-CC13レジスタすべてをコンペア・レジスタに設定して、そのなかの2つのレジスタの比較結果で使用する場合でも、必ず残りの2つのレジスタにも任意の値を設定して使用してください。任意の値を設定しない場合の動作は保証できません。

キャプチャ/コンペア・レジスタ、コンペア・レジスタの機能一覧を次に示します。

表10-1 キャプチャ/コンペア・レジスタ

タイマ	レジスタ	タイマ 再スタート	発生する 割り込み信号	キャプチャ・ トリガ	タイマ出力 (セット/リセット)	その他の機能
TM1	CC10	-	INTCC10	INTP10	TO10 (セット)	-
	CC11	-	INTCC11	INTP11	TO10 (リセット)	-
	CC12	-	INTCC12	INTP12	TO11 (セット)	-
	CC13	-	INTCC13	INTP13	TO11 (リセット)	-
	TM1	-	INTOV1	-	-	外部クリア
TM4	CM4		INTCM4	-	-	-

10.3 タイマ/カウンタ制御レジスタ

10.3.1 タイマ・ユニット・モード・レジスタ (TUM1)

TUM1レジスタは、キャプチャ/コンペア・レジスタの動作モードを指定します。16ビット単位でリード/ライトできます。

図10-7 タイマ・ユニット・モード・レジスタ (TUM1) (1/2)

アドレス：C0000072H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TUM1	0	0	OST	ECLR1	TES11	TES10	CES11	CES10	CMS13	CMS12	CMS11	CMS10	IMS13	IMS12	IMS11	IMS10

初期値：

0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W: R R R/W R/W

ビット	ビット名	説明															
13	OST	Overflow Stop タイマのオーバーフロー後の動作を指定します。このフラグはTM1だけ有効です。 0：タイマのオーバーフロー後、タイマはカウント・アップを続けます。 1：タイマのオーバーフロー後、タイマは0000Hを保持して停止状態になります。 このとき、TMC1レジスタのCE1ビットは1のままです。 次の動作によりカウント・アップを再開します。 ・CE1ビットへ1をライト (ECLR1 = 0のとき) ・タイマ・クリア端子 (TCLR1) のトリガ入力 (ECLR1 = 1のとき)															
12	ECLR1	External Input Timer Clear TM1の外部クリア入力 (TCLR) によるタイマのクリアを許可します。 0：外部入力によるクリアはしません。 1：外部入力によりTM1をクリアします。クリア後、カウント・アップを開始します。															
11, 10	TES11, TES10	TI1 Edge Select 外部クロック入力 (TI) の有効エッジを指定します。 <table border="1" style="margin: 5px auto; border-collapse: collapse;"> <thead> <tr> <th>TES11</th> <th>TES10</th> <th>有効エッジ</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>RFU (予約)</td> </tr> <tr> <td>0</td> <td>1</td> <td>RFU (予約)</td> </tr> <tr> <td>1</td> <td>0</td> <td>立ち上がりエッジ (初期値)</td> </tr> <tr> <td>1</td> <td>1</td> <td>立ち上がりと立ち下がりの両エッジ</td> </tr> </tbody> </table>	TES11	TES10	有効エッジ	0	0	RFU (予約)	0	1	RFU (予約)	1	0	立ち上がりエッジ (初期値)	1	1	立ち上がりと立ち下がりの両エッジ
TES11	TES10	有効エッジ															
0	0	RFU (予約)															
0	1	RFU (予約)															
1	0	立ち上がりエッジ (初期値)															
1	1	立ち上がりと立ち下がりの両エッジ															
9, 8	CES11, CES10	TCLR1 Edge Select 外部クリア入力 (TCLR) の有効エッジを指定します。 <table border="1" style="margin: 5px auto; border-collapse: collapse;"> <thead> <tr> <th>CES11</th> <th>CES10</th> <th>有効エッジ</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>RFU (予約)</td> </tr> <tr> <td>0</td> <td>1</td> <td>RFU (予約)</td> </tr> <tr> <td>1</td> <td>0</td> <td>立ち上がりエッジ (初期値)</td> </tr> <tr> <td>1</td> <td>1</td> <td>立ち上がりと立ち下がりの両エッジ</td> </tr> </tbody> </table>	CES11	CES10	有効エッジ	0	0	RFU (予約)	0	1	RFU (予約)	1	0	立ち上がりエッジ (初期値)	1	1	立ち上がりと立ち下がりの両エッジ
CES11	CES10	有効エッジ															
0	0	RFU (予約)															
0	1	RFU (予約)															
1	0	立ち上がりエッジ (初期値)															
1	1	立ち上がりと立ち下がりの両エッジ															

図10-7 タイマ・ユニット・モード・レジスタ (TUM1) (2/2)

アドレス : C0000072H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TUM1	0	0	OST	ECLR1	TES11	TES10	CES11	CES10	CMS13	CMS12	CMS11	CMS10	IMS13	IMS12	IMS11	IMS10

初期値 :

0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

R/W : R R R/W R/W

ビット	ビット名	説明
7-4	CMS13-CMS10	Capture/Compare Mode Select キャプチャ/コンペア・レジスタ (CC13-CC10) の動作モードを選択します。 0 : キャプチャ・レジスタとして動作します。ただし、キャプチャ・レジスタ指定時のキャプチャ動作は、TMC1レジスタのCEビット = 1のときだけ行います。 1 : コンペア・レジスタとして動作します。
3-0	IMS13-IMS10	Interrupt Mode Select 割り込みソースとして、INTP1nかINTCC1nかを選択します (n = 3-0)。 0 : コンペア・レジスタの一致信号 (INTCC1n) を割り込み要求信号にします。 1 : 外部からの入力信号 (INTP1n) を割り込み要求信号にします。

注意 タイマ動作中にCMS1n, IMS1nを変更した場合の動作は保証できません (n = 3-0)。

10.3.2 タイマ・コントロール・レジスタ1 (TMC1)

TMC1は、タイマ1 (TM1) の動作を制御します。8ビット単位でリード/ライトできます。

図10-8 タイマ・コントロール・レジスタ1 (TMC1)

アドレス : C0000074H

	7	6	5	4	3	2	1	0
TMC1	CE1	0	0	ETI	PRS11	PRS10	PRM11	0

初期値 :

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

R/W : R/W R R R/W R/W R/W R/W R

ビット	ビット名	説明															
7	CE1	Count Enable タイマの動作を制御します。 0 : タイマは0000H状態にクリアされ、動作しません。 1 : タイマはカウント動作を行います。ただし、TUM1のECLR1ビット = 1のときは、タイマ・クリア信号 (TCLR) の入力があるまでカウント・アップしません。 TUM1のECLR1ビット = 0のときは、CE1ビット = 1によるタイマのカウント・スタートは、CE1ビットへのセット動作 (1) がスタート・トリガになります。したがって、ECLR1ビット = 1の状態ではCE1ビットをセットしたあと、ECLR1ビット = 0にしても、タイマはスタートしません。															
4	ETI	External TI Input カウント・クロックの外部と内部の切り替えを指定します。 0 : 系 (内部) を指定します。 1 : TI (外部) を指定します。															
3, 2	PRS11, PRS10	Prescaler Clock Select 外部カウント・クロックを選択します (mは中間クロック)。 <table border="1" style="margin-left: 20px; width: 100%;"> <thead> <tr> <th>PRS11</th> <th>PRS10</th> <th>カウント・クロック</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>m</td> </tr> <tr> <td>0</td> <td>1</td> <td>m/4</td> </tr> <tr> <td>1</td> <td>0</td> <td>RFU (予約)</td> </tr> <tr> <td>1</td> <td>1</td> <td>m/16</td> </tr> </tbody> </table>	PRS11	PRS10	カウント・クロック	0	0	m	0	1	m/4	1	0	RFU (予約)	1	1	m/16
PRS11	PRS10	カウント・クロック															
0	0	m															
0	1	m/4															
1	0	RFU (予約)															
1	1	m/16															
1	PRM11	Prescaler Clock Mode カウント・クロックの中間クロック mを指定します (はバス・クロック)。 0 : /2 1 : /4															

注意 カウント動作中にクロックを変更した場合の動作は保証できません。クロックを切り替える場合はカウント動作を停止してから切り替えてください。

10.3.3 タイマ・コントロール・レジスタ4 (TMC4)

TMC4は、タイマ4 (TM4) の動作を制御します。8ビット単位でリード/ライトできます。

図10-9 タイマ・コントロール・レジスタ4 (TMC4)

アドレス : C0000088H

7	6	5	4	3	2	1	0
TMC4	CE4	0	0	0	0	PRS40	PRM41 PRM40

初期値 :

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

R/W : R/W R R R R R/W R/W R/W

ビット	ビット名	説明															
7	CE4	Count Enable タイマの動作を制御します。 0 : タイマは0000H状態にクリアされ、動作しません。 1 : タイマはカウント動作を行います。															
2	PRS40	Prescaler Clock Select 内部カウント・クロックを選択します (mは中間クロック)。 0 : m/16 1 : m/32															
1, 0	PRM41, PRM40	Prescaler Clock Mode カウント・クロックの中間クロック mを指定します (はバス・クロック)。 <table border="1" style="margin-left: 20px; border-collapse: collapse;"> <thead> <tr> <th>PRM41</th> <th>PRM40</th> <th>m</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>/2</td> </tr> <tr> <td>0</td> <td>1</td> <td>RFU (予約)</td> </tr> <tr> <td>1</td> <td>0</td> <td>/8</td> </tr> <tr> <td>1</td> <td>1</td> <td>RFU (予約)</td> </tr> </tbody> </table>	PRM41	PRM40	m	0	0	/2	0	1	RFU (予約)	1	0	/8	1	1	RFU (予約)
PRM41	PRM40	m															
0	0	/2															
0	1	RFU (予約)															
1	0	/8															
1	1	RFU (予約)															

注意 カウント動作中にクロックを変更した場合の動作は保証できません。クロックを切り替える場合はカウント動作を停止してから切り替えてください。

10.3.4 タイマ出力コントロール・レジスタ (TOC1)

TOC1は、タイマ出力モードを指定します。8ビット単位でリード/ライトできます。

図10-10 タイマ出力コントロール・レジスタ (TOC1)

アドレス : C0000076H

	7	6	5	4	3	2	1	0
TOC1	ENTO11	ALV11	ENTO10	ALV10	0	0	TOPC11	TOPC10

初期値 :

0	0	0	0	0	0	1	1
---	---	---	---	---	---	---	---

R/W : R/W R/W R/W R/W R R R/W R/W

ビット	ビット名	説明
7, 5	ENTO11, ENTO10	Enable TOxx pin 対応するタイマ出力 (TO11, TO10) を許可します。 0 : タイマ出力機能は禁止状態です。対応するTO11, TO10端子からはALVビットの逆相のレベル (インアクティブ・レベル) を出力します。対応するコンペア・レジスタから一致信号が発生してもTO11, TO10端子のレベルは変化しません。 1 : タイマ出力は許可状態です。対応するコンペア・レジスタから一致信号が発生するとタイマ出力が変化します。タイマ出力を許可してから最初に一致信号が発生するまでは、ALVビットの逆相レベル (インアクティブ・レベル) が出力されます。
6, 4	ALV11, ALV10	Active Level TOxx pin タイマ出力のアクティブ・レベルを指定します。 0 : アクティブ・レベルはロウ・レベル (0) 1 : アクティブ・レベルはハイ・レベル (1)
1	TOPC11	Prescaler Clock Mode 兼用端子TO11/INTP12の端子機能を選択します。 0 : TO11出力 1 : INTP12入力 (初期値)
0	TOPC10	Prescaler Clock Mode 兼用端子TO10/INTP10の端子機能を選択します。 0 : TO10出力 1 : INTP10入力 (初期値)

注意 TO10, TO11出力は外部割り込み信号 (INTP1n) では変化しません。TO10, TO11を使用するときは、キャプチャ/コンペア・レジスタをコンペア・レジスタ (CMS1n = 1) に指定してください (n = 0-3)。

備考 TO10, TO11出力のF/Fはリセット優先です。

10.3.5 ICUモード・レジスタ (IMOD)

外部割り込み信号の有効エッジを指定する制御レジスタです。

TM1のCC1nをキャプチャ・レジスタとして使用する場合、外部割り込み (INTP1n) の有効エッジをキャプチャ・トリガとして検出します (n = 0-3)。この有効エッジは、ICUモード・レジスタ (IMOD) で指定できます (詳細は4.6.5 ICUモード・レジスタ (IMOD) を参照)。

10.3.6 タイマ・オーバーフロー・ステータス・レジスタ (TOVS)

タイマ1, 4 (TM1, TM4) からオーバーフロー・フラグを格納します。8ビット単位でリード/ライトできます。TOVSレジスタをソフトウェアでテスト&リセットすることで、オーバーフロー発生をポーリングできます。

図10-11 タイマ・オーバーフロー・ステータス・レジスタ (TOVS)

アドレス: C0000070H								
	7	6	5	4	3	2	1	0
TOVS	0	0	0	OVF4	0	0	OVF1	0
初期値:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R

ビット	ビット名	説明
4, 1	OVF4, OVF1	Overflow Flag TMn (n = 4, 1) オーバフロー・フラグ。 0: TMnのオーバーフロー発生なし 1: TMnのオーバーフロー発生 TOVSレジスタはマスタ/スレーブ構成になっているため、CPUからのアクセス期間中はスレーブ側への転送 (TOVSレジスタへの転送) はできません。したがって、TOVSレジスタの読み出し中にオーバーフローが発生しても、フラグの値は変化せず、次の読み出し時に反映されます。

注意 TM1からはオーバーフローに同期して、割り込みコントローラに対し、割り込み要求信号 (INTOV1) が発生しますが、割り込みの動作とTOVSとはまったく独立しており、OVF1に1をライトしてもINTOV1は発生しません。また、OVF1に0をライトすることによりフラグをクリアできます。

10.4 動作

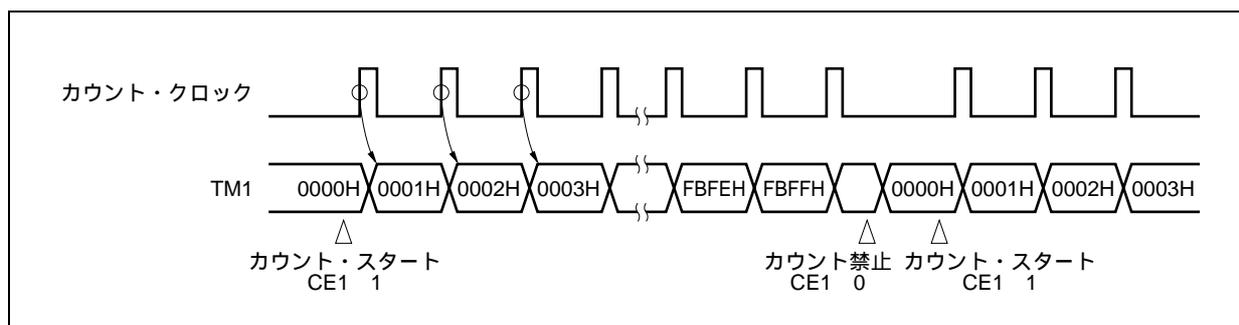
10.4.1 タイマ1

タイマ1は、16ビットのフリー・ランニング・タイマまたは、外部信号のイベント・カウンタとして機能します。動作の設定は、タイマ・コントロール・レジスタ1 (TMC1) で指定します。

タイマ1のカウンタ動作は、TMC1レジスタのPRS11, PRS10, PRM11ビットで指定する内部クロック ($\phi/2 \sim \phi/64$)、または外部クロック入力 (TI) をカウント・アップします。このときカウント・クロックを外部クロックに指定すると、TM1はイベント・カウンタとして動作します。カウントの結果、タイマがオーバーフローするとオーバーフロー割り込み要求 (INTOV1) が発生します。

フリー・ランニング・タイマとして動作する場合、CC10-CC13レジスタとTM1のカウント値が一致すると、割り込み要求信号 (INTCC10-INTCC13) を発生させるとともに、タイマ出力信号 (TO10, TO11) のセット/リセットができます。また、外部トリガとしての外部割り込み要求入力端子 (INTP10-INTP13) から検出された有効エッジに同期して、TM1のカウント値をCC10-CC13レジスタに保持するキャプチャ動作を行います。キャプチャの値は、次のキャプチャ・トリガが発生するまで保持されます。

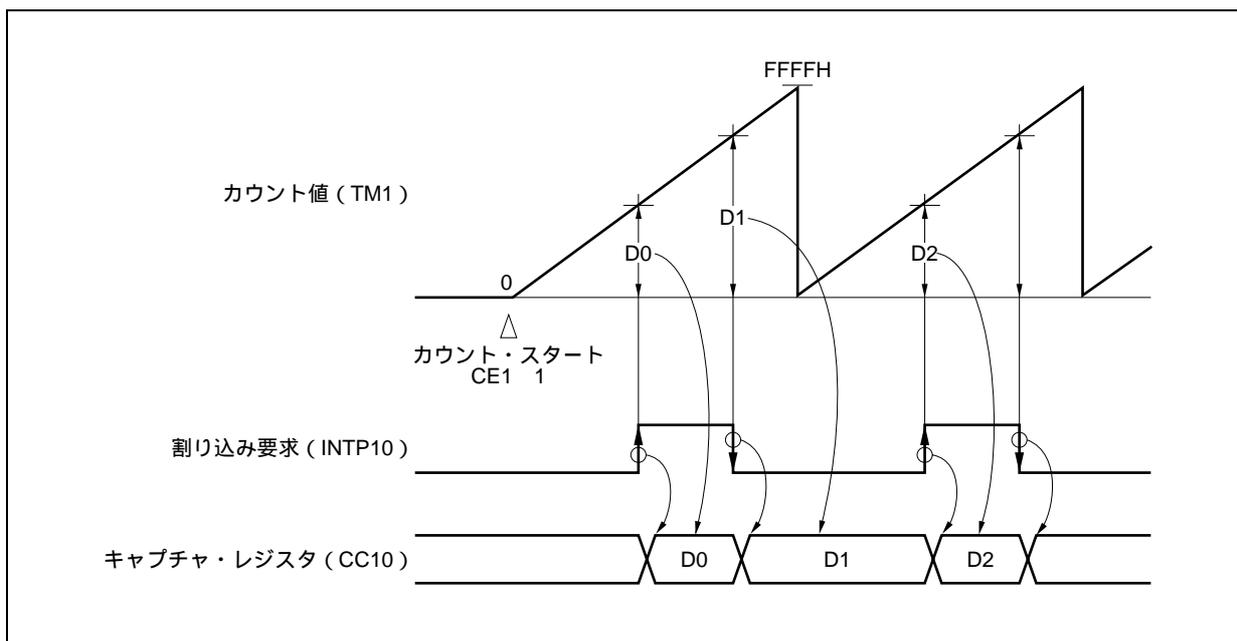
図10 - 12 タイマ1の基本動作



立ち上がり、立ち下がりの両エッジをキャプチャ・トリガとした場合は、外部からの入力パルスの幅を測定できます。

立ち上がりエッジをキャプチャ・トリガとした場合は、入力パルスの周期を測定できます。

図10 - 13 キャプチャ動作例



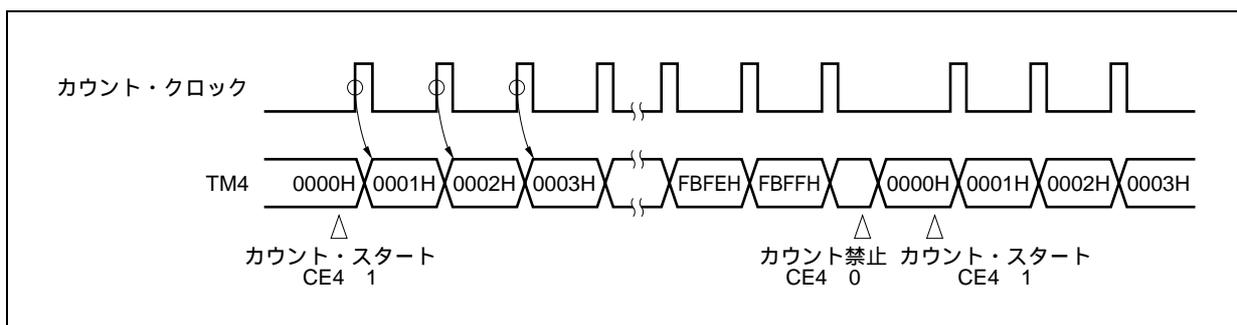
10.4.2 タイマ4

タイマ4は、16ビットのインターバル・タイマとして機能します。動作の設定は、タイマ・コントロール・レジスタ4 (TMC4) で指定します。

タイマ4のカウンタ動作は、TMC4レジスタのPRS40, PRM41, PRM40ビットで指定される内部クロック ($\phi/32 \sim \phi/256$) をカウンタ・アップします。カウンタの結果、TM4の値がCM4と一致すると、TM4をクリアします。同時に一致割り込み要求 (INTCM4) が発生します。

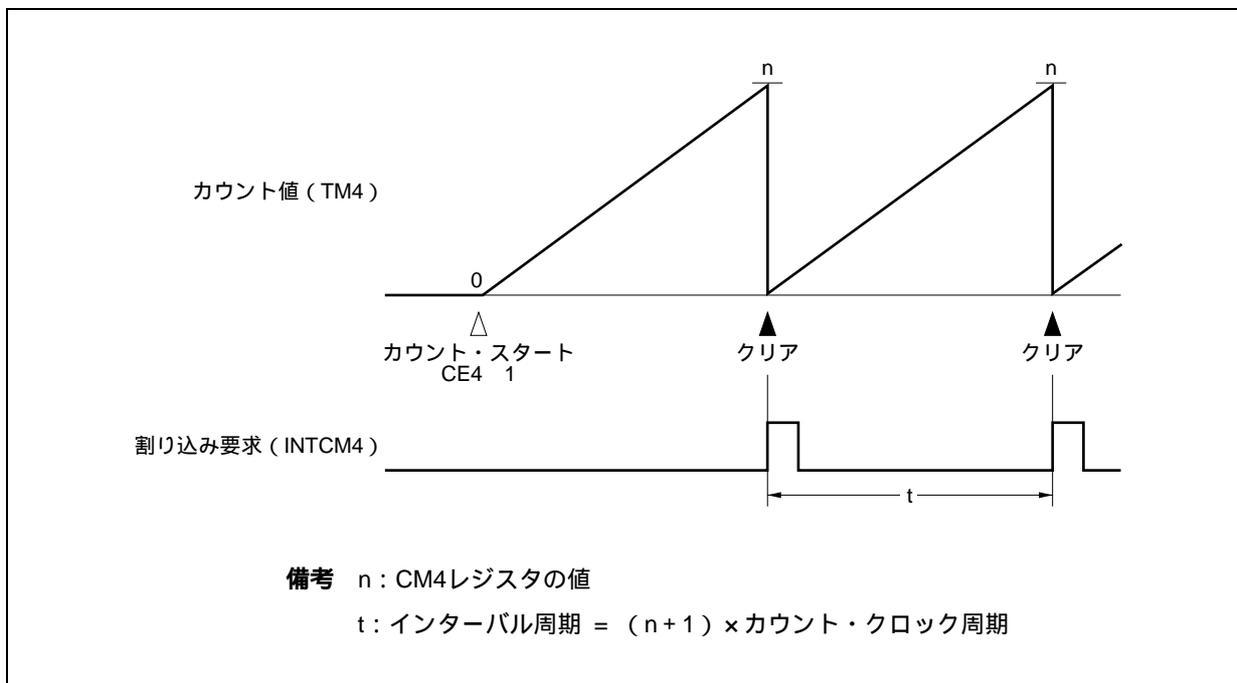
一致割り込み (INTCM4) を使用して、DMA転送ができます (第8章 DMA機能を参照)。

図10 - 14 タイマ4の基本動作



コンペア動作で一致を検出すると、次のカウント・クロック入力でTM4を0にクリアします。この機能により、CM4に設定した値 + 1 のカウント・クロック周期のインターバル・タイマとして動作します。

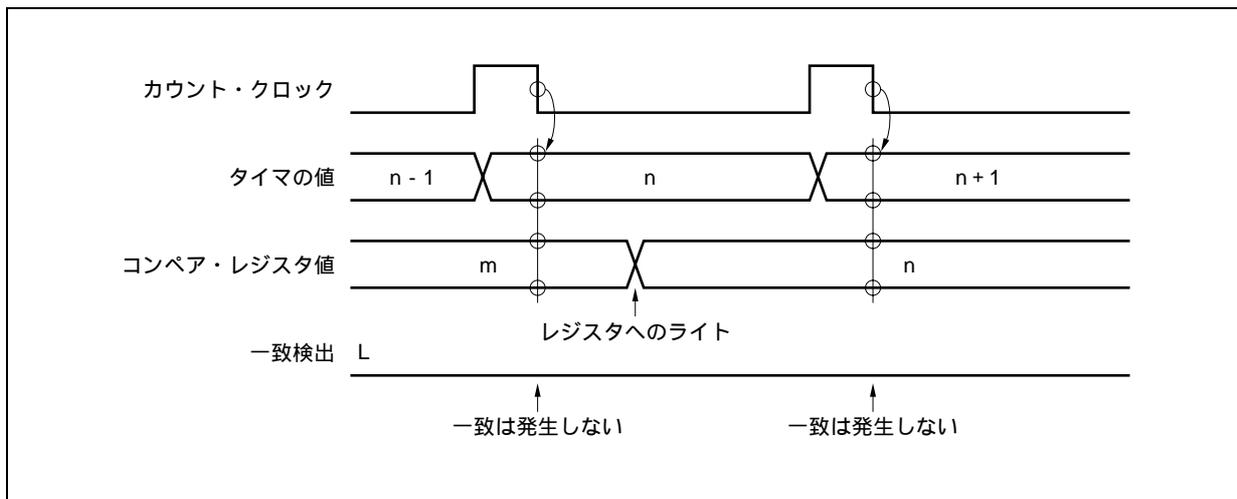
図10 - 15 コンペア動作例



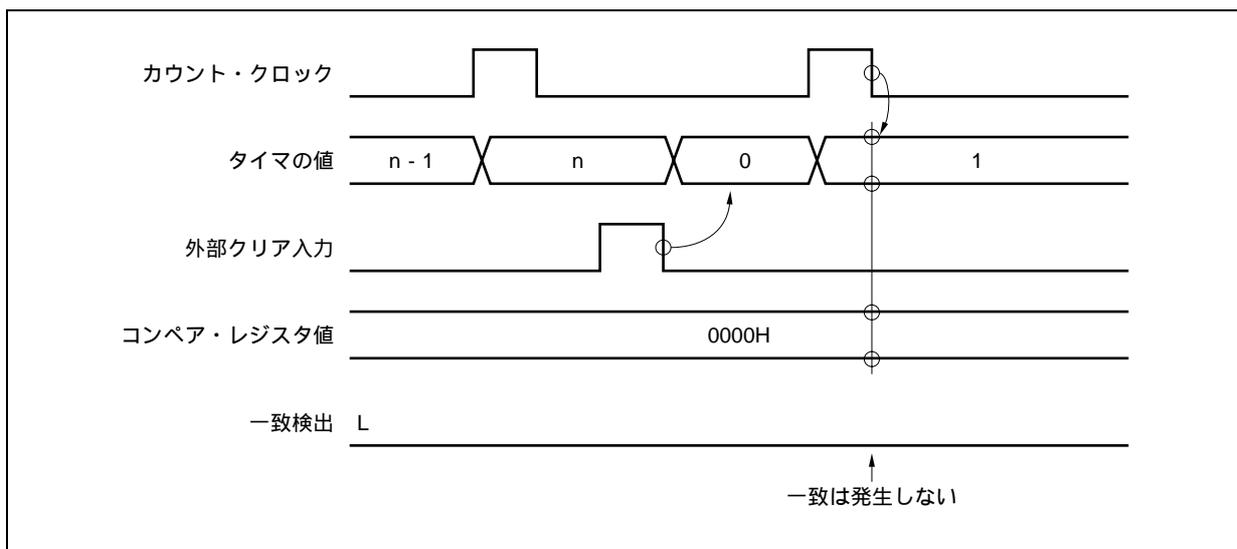
10.5 注意事項

コンペア・レジスタによる一致検出は、常にタイマのカウント・アップ直後のタイミングで行います。次に示す場合、一致は発生しません。

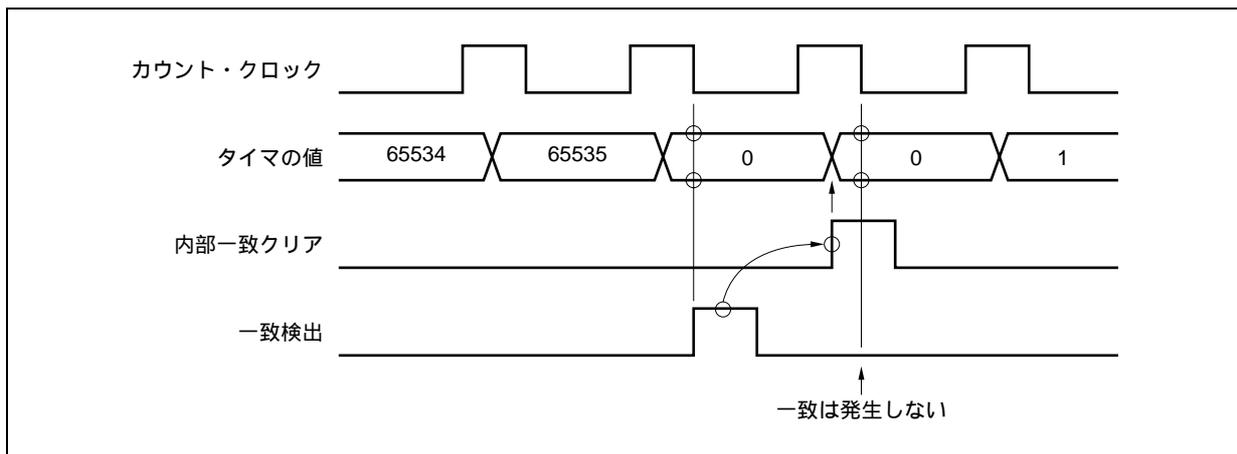
(1) コンペア・レジスタの書き換え時 (TM1, TM4)



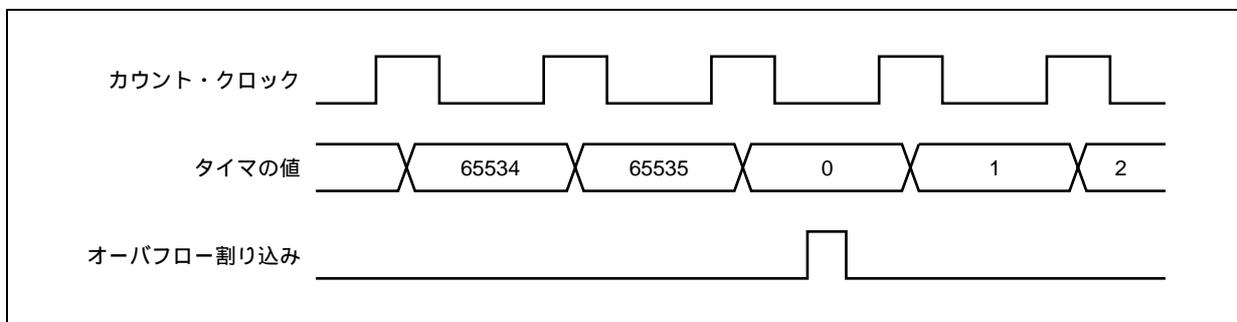
(2) 外部クリア時 (TM1)



(3) タイマ・クリア時 (TM4) (コンペア・レジスタ値=0000Hのとき)



注意 フリー・ランニングでタイマ動作させるときは、タイマのオーバーフローでタイマの値は0になります。



(メ モ)

第11章 ポート機能

V831のポートは、1ビット単位で入出力を指定できる3ビット入出力ポートです。

ポートとしての機能のほかにコントロール・モードでは、シリアル・インタフェース（CSI）の入出力として動作します。

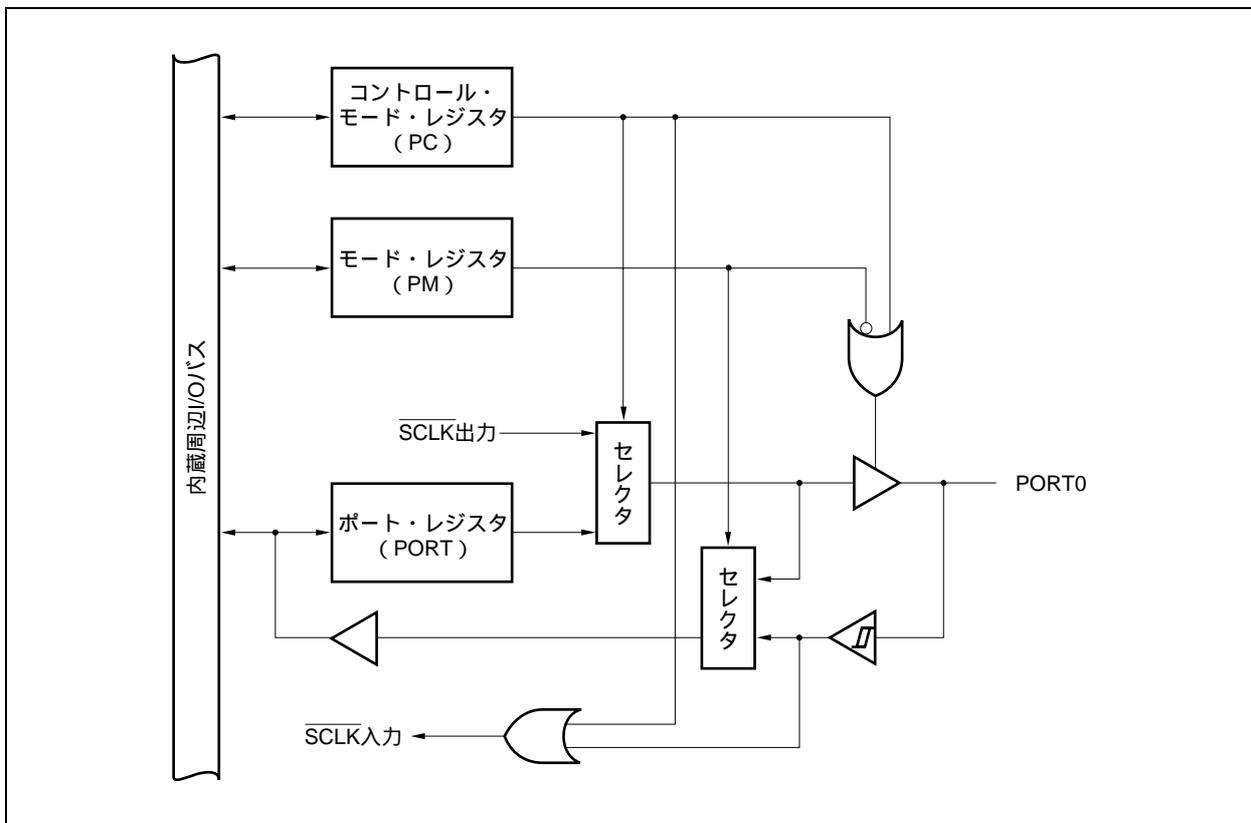
表11-1 コントロール・モード時の動作

ポート	コントロール・モード	備考
ポート0	SCLK	シリアル・インタフェース（CSI）用入出力と兼用
ポート1	SO	
ポート2	SI	

11.1 構成

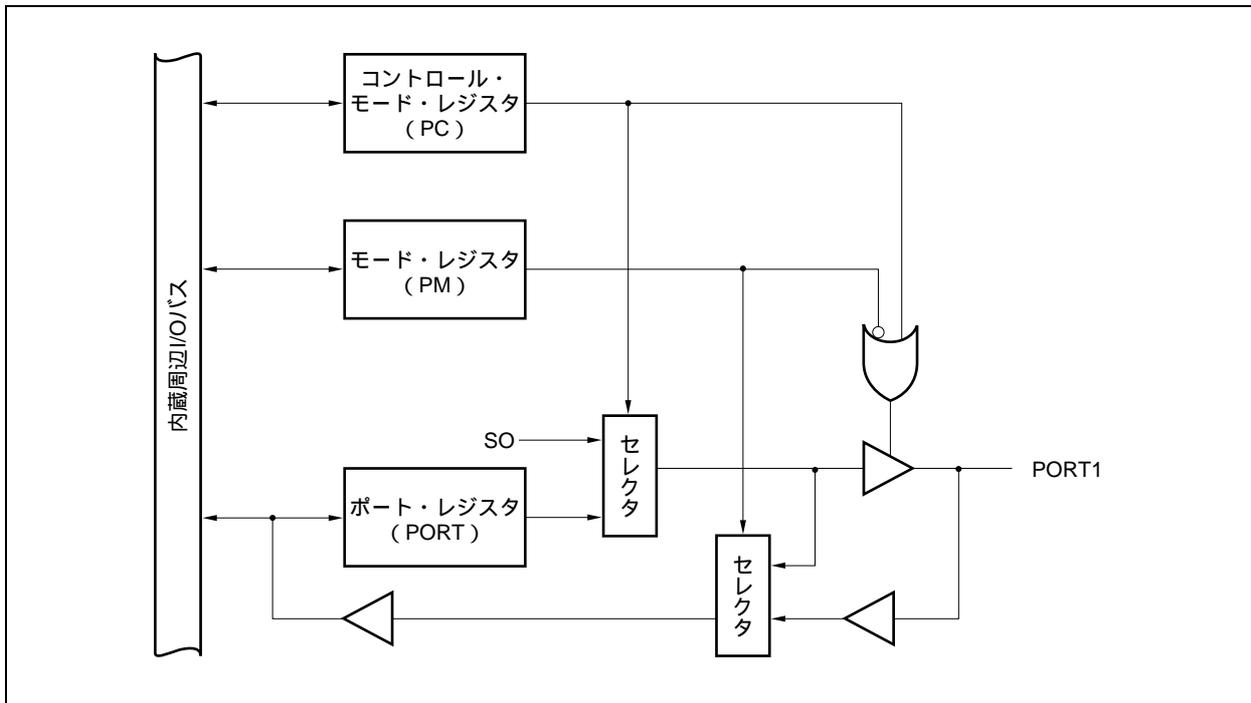
(1) ポート0の構成

図11-1 ポート0のブロック図



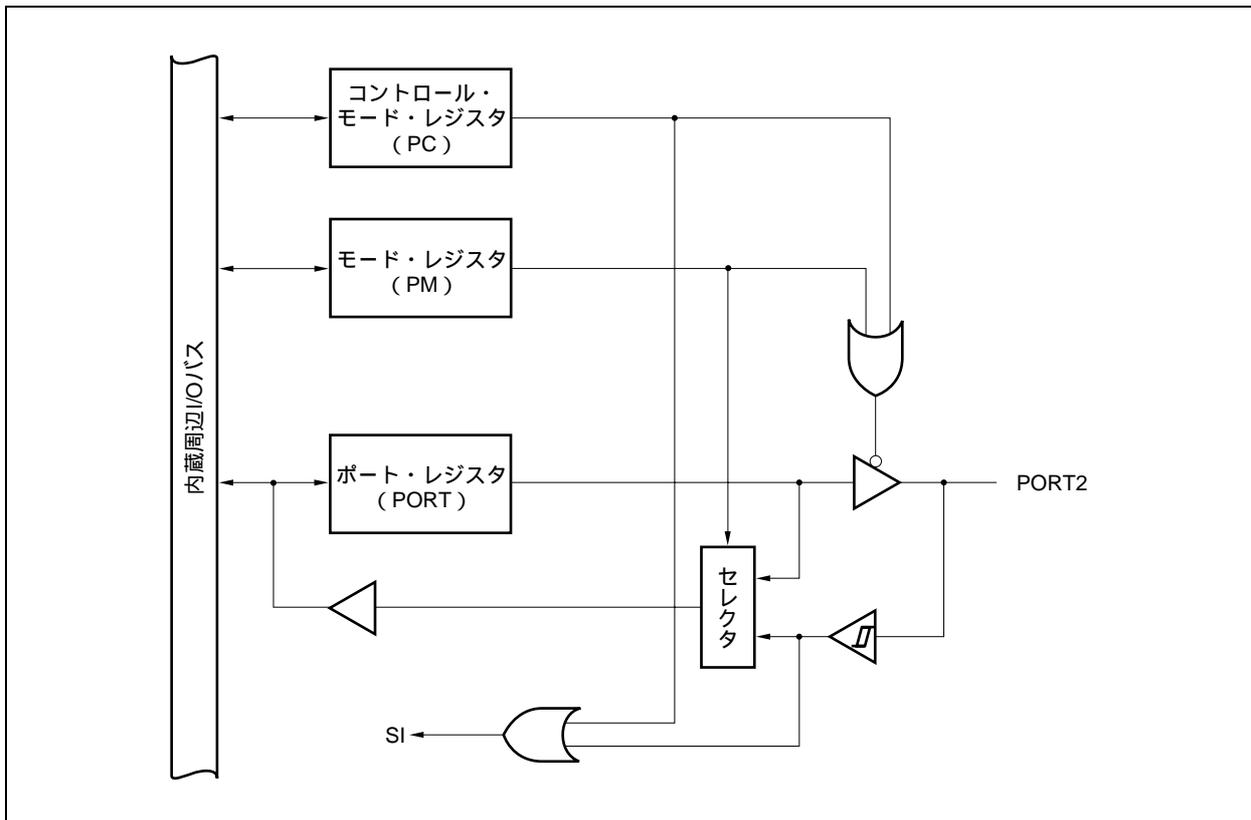
(2) ポート1の構成

図11-2 ポート1のブロック図



(3) ポート2の構成

図11-3 ポート2のブロック図



11.2 ポート制御レジスタ

11.2.1 入出力ポート・レジスタ (PORT)

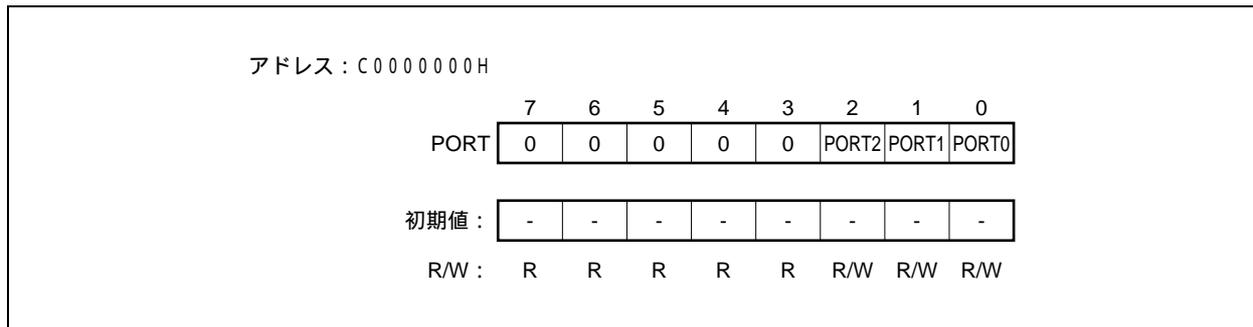
3ビット入出力ポートです。

8ビット単位のリード/ライトが可能です。

入力モード時は、ポート端子を直接読むことができます。ただし、本レジスタは値を保持しません。

出力モード時は、本レジスタに書き込まれた値をポート端子に出力します。

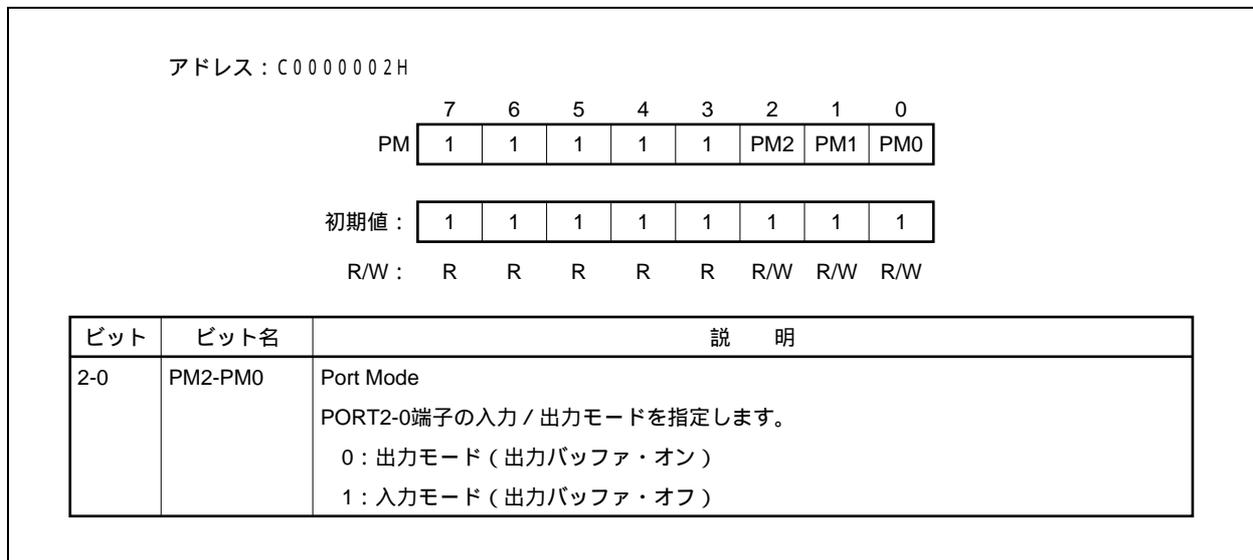
図11-4 入出力ポート・レジスタ (PORT)



11.2.2 入出力モード・レジスタ (PM)

1ビット単位でポートの入力/出力モードを設定します。8ビット単位のリード/ライトが可能です。

図11-5 入出力モード・レジスタ (PM)



11.2.3 ポート・コントロール・モード・レジスタ (PC)

ポート/CSI動作切り替えのコントロール・モードを設定します。8ビット単位のリード/ライトが可能です。

図11-6 ポート・コントロール・モード・レジスタ (PC)

アドレス : C0000004H

	7	6	5	4	3	2	1	0
PC	0	0	0	0	0	PC2	PC1	PC0

初期値 :

0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---

R/W : R R R R R R/W R/W R/W

ビット	ビット名	説 明																
2-0	PC2-PC0	Port Control PORT2-PORT0端子の動作モードを指定します。 <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr> <th>PC2</th> <th>PC1</th> <th>PC0</th> <th>動作モード</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>入出力ポート・モード</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>CSIインタフェース</td> </tr> <tr> <td colspan="3">上記以外</td> <td>予約 (設定禁止)</td> </tr> </tbody> </table>	PC2	PC1	PC0	動作モード	0	0	0	入出力ポート・モード	1	1	1	CSIインタフェース	上記以外			予約 (設定禁止)
PC2	PC1	PC0	動作モード															
0	0	0	入出力ポート・モード															
1	1	1	CSIインタフェース															
上記以外			予約 (設定禁止)															

注意 ポート・コントロール・モード・レジスタ (PC) を設定する前にCSI (9. 2参照) のクロック同期式シリアル・インタフェース・モード・レジスタ (CSIM0) のCLS02-CLS00ビットを設定してください。

第12章 クロック発生機能

クロック・ジェネレータは、CPUをはじめとする内蔵の各ハードウェア・ユニットに供給されるCPUクロックとバス・クロック (ϕ) を発生、制御します。

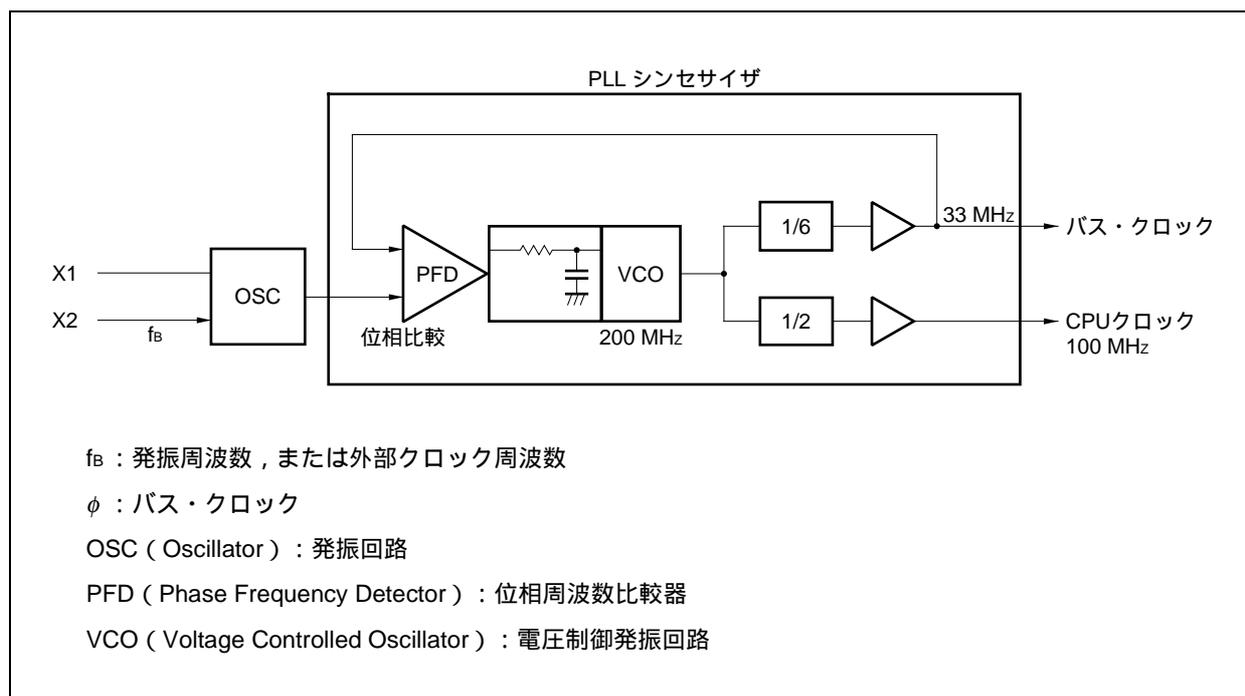
表12 - 1 PLLシンセサイザによる逡倍機能

	規 格
バス・クロック (ϕ)	16.7-33 MHz (f_B)
CPUクロック	50-100 MHz ($3 \times f_B$)

備考 PLL : Phase Locked Loop

12.1 構 成

図12 - 1 クロック発生機能のブロック図



12.2 入力クロック選択

クロック・ジェネレータは、クロック発振回路とPLLシンセサイザで構成しています。たとえば、33 MHzのクリスタル振動子またはセラミック発振子をX1, X2端子に接続すると、33 MHzのバス・クロックと100 MHzのCPUクロックを生成できます。

また、外部クロックを発振回路に直接入力できます。この場合、X2端子だけクロック信号を入力して、X1端子をオープンにしてください。

12.2.1 ロックアップ時間

電源投入直後またはSTOPモード解除直後は、PLLが所定の周波数においてフェーズ・ロック状態で、安定するまでの周波数安定時間（ロックアップ時間）が発生します。この安定するまでの状態をアンロック状態と呼び、安定している状態をロック状態と呼びます。

電源投入直後、またはSTOPモードから $\overline{\text{RESET}}$ 信号で復帰した場合は、 $\overline{\text{RESET}}$ 信号による安定時間を確保してください（13.4 発振安定時間の確保を参照）。

STOPモードからのNMI信号による復帰については自動的に安定時間を確保します。

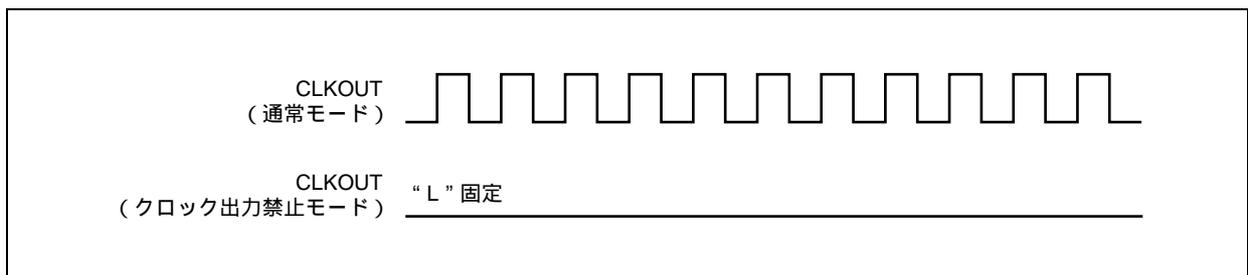
12.3 クロック出力制御

クロック・コントロール・レジスタ（CGC）のCOEビットで、CLKOUT端子の動作を選択できます。HALT/STOPの各スタンバイ・モードと組み合わせると、より効果的にパワー・セービングが行えます。

12.3.1 クロック出力禁止モード

CLKOUT端子からのクロック出力を禁止します。CLKOUTの動作が完全に停止するため、一層の低消費電力化およびCLKOUT端子からの輻射ノイズ抑止を行えます。

図12-2 クロック出力禁止モード



12.4 クロック制御レジスタ

12.4.1 クロック・コントロール・レジスタ (CGC)

CLKOUTの出力を制御します。8ビット単位でリード/ライトができます。

図12-3 クロック・コントロール・レジスタ (CGC)

アドレス : C00000E0H

	7	6	5	4	3	2	1	0
CGC	KEY				CESEL	0	0	COE

初期値 :

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

R/W : R/W R/W R/W R/W R/W R R R/W

ビット	ビット名	説明
7-4	KEY	KEY Data データを識別します。値は必ず“0111”にしてください。 CESEL, COEビットに書き込むときに、KEYデータの値が“0111”以外の場合には書き込みません。 リード時は常に“0000”が読み出されます。
3	CESEL	Crystal External Select X1, X2端子の機能を指定します。 0 : X1, X2端子に発振子を接続してください。 1 : X2端子に外部クロックを接続, X1端子はオープンにしてください。
0	COE	Clock Out Enable CLKOUTの出力を許可/禁止します。 0 : 出力禁止 (CLKOUT端子はロウ・レベル固定) 1 : 出力許可

第13章 スタンバイ機能

V831には、低消費電力化を図るためのスタンバイ機能として、動作クロックを制御するモードが2つあります。

HALTモード（CPUクロックだけ停止）

STOPモード（システム全体を停止、クロック・ジェネレータも停止）

HALT, STBY命令で各モードに移行します。

13.1 スタンバイ・モード

スタンバイ・モードには、次に示すモードがあります。

(1) HALTモード

クロック・ジェネレータ（発振回路およびPLLシンセサイザ）は動作を継続しますが、CPUの動作クロックを停止させるモードです。その他の内蔵周辺機能へのクロック供給は継続され、動作を継続します。通常モードと組み合わせる動作により、システム全体の消費電力を低減できます。

(2) STOPモード

クロック・ジェネレータ（PLLシンセサイザ）を停止させ、システム全体を停止させるモードです。PLLシンセサイザが停止し、周辺内蔵機能も停止するのでHALTモードより、さらに消費電力を低減できます。

PLLシンセサイザのクロック出力が停止するので、STOPモードの解除後はCPUクロックとバス・クロックが安定するまでは、発振回路の発振安定時間を確保してください。また、プログラムによってはPLL回路のロックアップ時間が必要な場合があります。

HALT, STOPの各モードにおけるクロック・ジェネレータの動作を表13 - 1に示します。各モードを用途により切り替えて使用すると、効果的な低消費電力システムを実現できます。

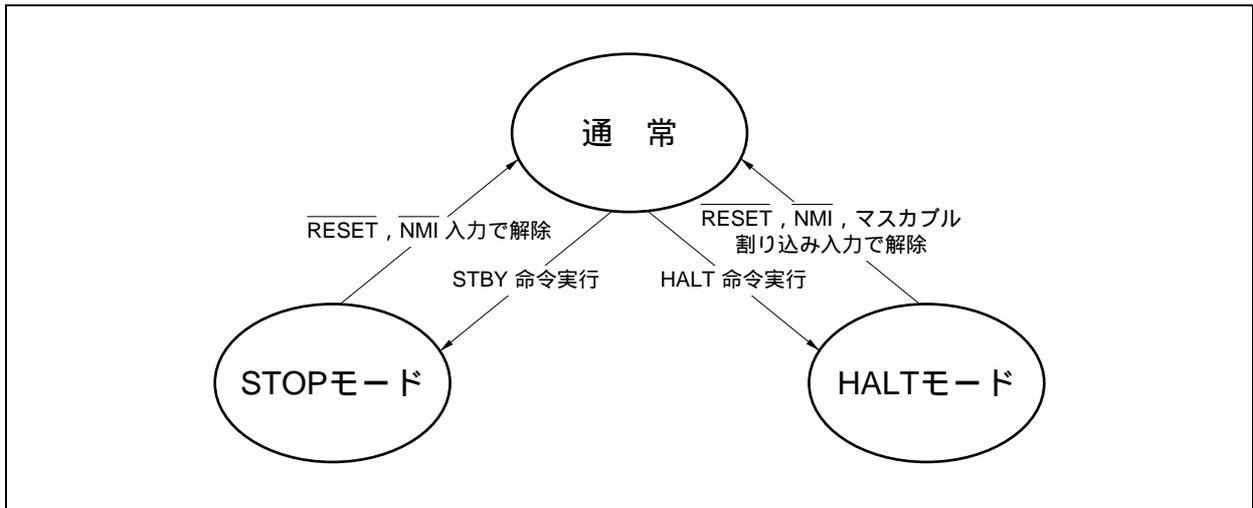
表13 - 1 スタンバイ制御によるクロック・ジェネレータの動作

スタンバイ・モード	発振回路（OSC）	PLLシンセサイザ	周辺I/Oへの クロック供給	CPUへの クロック供給
通常モード				
HALTモード				×
STOPモード		×	×	×

備考 : 動作

 × : 停止

図13-1 状態遷移図



13.2 HALTモード

13.2.1 HALTモードの設定と動作状態

クロック・ジェネレータ（発振回路およびPLLシンセサイザ）は動作を継続しますが、CPUの動作クロックを停止させるモードです。その他の周辺内蔵機能へのクロック供給は継続され、動作を継続します。CPUの空き時間にHALTモードに設定すると、システム全体の消費電力を低減できます。

HALT命令を実行するとHALTモードへ移行します。HALTモードでは、プログラムの実行は停止しますが、その直前のすべてのレジスタの内容は保持されます。また、CPUの命令処理に依存しないオンチップの周辺機能は動作を継続します。HALTモード中の $\overline{\text{HLDRQ}}$ 信号は受け付けます。

ノンマスカブル割り込み処理ルーチン内でHALTモードに設定する場合は、HALT命令を実行する前に、 $\text{PSW.NP} = 1$ （NMI処理中の状態）から $\text{PSW.NP} = 0$ （NMI処理中でない状態）にして、新たなノンマスカブル割り込み要求を受け付ける状態にしてください。 $\text{PSW.NP} = 0$ にしない場合、 $\overline{\text{NMI}}$ によってHALTモードからの復帰ができなくなります。

HALTモード時の各ハードウェアの状態を表13-2に示します。

表13-2 HALTモード時の動作状態

機 能	動作状態 ^{注1}	
発振回路	動 作	
PLLシンセサイザ	動 作	
バス・クロック	動 作	
CPU	停 止	
ポート出力	保 持	
周辺機能	動 作	
内部データ	CPUのレジスタなど内部データはすべてHALTモード設定前の状態を保持	
A1-A23	不 定	$\overline{\text{HLD}}\text{AK} = 0$ のときはハイ・インピーダンス
D0-D31	ハイ・インピーダンス	
$\overline{\text{BCYST}}$	1	$\overline{\text{HLD}}\text{AK} = 0$ のときはハイ・インピーダンス
$\overline{\text{CS1-CS7}}$		
$\overline{\text{IORD}}, \overline{\text{IOWR}}$		
$\overline{\text{MRD}}, \overline{\text{WE}}, \overline{\text{OE}}, \overline{\text{LLMWR}}, \overline{\text{LUMWR}}, \overline{\text{ULMWR}}, \overline{\text{UUMWR}}$		
$\overline{\text{REFRQ}}, \overline{\text{LLCAS}}, \overline{\text{LUCAS}}, \overline{\text{ULCAS}}, \overline{\text{UUCAS}}$		
RAS	注3	
$\overline{\text{HLDRQ}}$	動 作	
CLKOUT	クロック出力（クロック出力禁止でないとき）	

注1. DMA転送中の場合，各端子は動作状態になります。

2. CBRリフレッシュ時以外。

3. CBRリフレッシュが入るまでは前の状態を保持します。CBRリフレッシュ後は“1”になります。

13.2.2 HALTモードの解除

HALTモードは，ノンマスクابل割り込み要求，マスクされていないマスクابل割り込み要求，およびRESET端子入力により解除されます。

(1) ノンマスクابل割り込み要求による解除

$\overline{\text{NMI}}$ 信号を検出すると，CPUクロックの供給を再開します。

(2) マスクابل割り込み要求による解除

マスクされていないマスクابل割り込み要求により，CPUクロックの供給を再開します。

割り込み処理ルーチン内でHALTモードに設定する場合は，HALT命令を実行する前に，HALT状態を解除する割り込み要求を受け付ける状態に設定してください。割り込み許可（PSW.ID = 0, EP = 0）と割り込み許可レベルを設定してください。

表13-3 割り込み要求によるHALTモード解除後の動作

解除要因	EI状態（PSW.ID = 0）	DI状態（PSW.ID = 1）
ノンマスクابل割り込み要求	ハンドラ・アドレス分岐	
マスクابل割り込み要求	ハンドラ・アドレス分岐	解除しません

(3) RESET端子入力による解除

- ★ 通常のリセット動作と同じです。したがって、スタンバイ・モードに入る前のレジスタ値等の状態は保持しません。

13.3 STOPモード

13.3.1 STOPモードの設定と動作状態

PLLシンセサイザを停止させるモードです（発振回路は停止しません）。

PLLシンセサイザが停止し、内蔵周辺回路も停止することにより低消費電力を実現します。STBY命令を実行すると、CBRセルフ・リフレッシュ・サイクルが起動してSTOPモードに移行します。

STOPモード解除後は、発振安定時間を確保してください。

STOPモードでは、プログラムの実行は停止しますが、その直前のすべてのレジスタの内容は保持されます。また、内蔵周辺機能の動作も停止します。

STOPモードに入ると内蔵周辺およびCLKOUTのクロック供給が停止するので、STOPモードに入る前に内蔵周辺および外部周辺の動作が終了したことを確認してからSTBY命令を実行してください。

STBY命令実行前にはDMACのDCHC0-DCHC3のENビットを0にしてDMA転送を禁止してください。

ノンマスクابل割り込み処理ルーチン内でSTOPモードに設定する場合は、STBY命令を実行する前に、PSW.NP = 1（NMI処理中の状態）からPSW.NP = 0（NMI処理中でない状態）にして、新たなノンマスクابل割り込み要求を受け付ける状態にしてください。PSW.NP = 0にしない場合、NMIによってSTOPモードからの復帰ができなくなります。

STOPモード時の各ハードウェアの状態を表13-4に示します。

表13-4 STOPモード時の動作状態

機 能	動作状態
発振回路	動 作
PLLシンセサイザ	停 止
パス・クロック	停 止
CPU	停 止
ポート出力	保 持
周辺機能	停 止
内部データ	CPUのレジスタなど内部データはすべてSTOPモード設定前の状態を保持
A1-A23	不 定
D0-D31	ハイ・インピーダンス
$\overline{\text{BCYST}}$	1
$\overline{\text{CS1-CS7}}$	
$\overline{\text{IORD}}, \overline{\text{IOWR}}$	
$\overline{\text{MRD}}, \overline{\text{WE}}, \overline{\text{OE}}, \overline{\text{LLMWR}}, \overline{\text{LUMWR}}, \overline{\text{ULMWR}}, \overline{\text{UUMWR}}$	
$\overline{\text{REFRQ}}, \overline{\text{RAS}}, \overline{\text{LLCAS}}, \overline{\text{LUCAS}}, \overline{\text{ULCAS}}, \overline{\text{UUCAS}}$	CBRセルフ・リフレッシュ ^注
$\overline{\text{HLDRQ}}$	受け付けない
$\overline{\text{CLKOUT}}$	0

注．CBRリフレッシュが禁止されている場合，CBRセルフ・リフレッシュは行いません。

その場合，この端子の状態はSTOPモードに入る前の状態を保持します。

13.3.2 STOPモードの解除

STOPモードは、ノンマスクابل割り込み要求、および $\overline{\text{RESET}}$ 端子入力で解除します。STOPモード解除時は、発振回路の発振安定時間を確保してください。

(1) ノンマスクابل割り込み要求 (NMI) による解除

$\overline{\text{NMI}}$ 信号を検出すると、PLLシンセサイザの動作が再開します。そのあと、発振安定時間分だけウェイトして、CPUクロックとバス・クロックの供給を開始します。

$\overline{\text{NMI}}$ 信号によるSTOPモード解除時に起動される割り込み処理は、通常のノンマスクابل割り込み処理と同等に扱われます。したがって、プログラムで両者を区別する場合は、ソフトウェア・ステータスをあらかじめ用意して、STBY命令実行前にステータスを設定してください。ノンマスクابل割り込み処理でこのステータスをチェックすると、通常の $\overline{\text{NMI}}$ 信号との区別ができます。

(2) $\overline{\text{RESET}}$ 端子入力による解除

- ★ 通常のリセット動作と同じです。したがって、スタンバイ・モードに入る前のレジスタ値等の状態は保持しません。

13.4 発振安定時間の確保

STOPモードを解除したあと、停止状態のPLL回路が安定するまでの時間確保が必要になります。

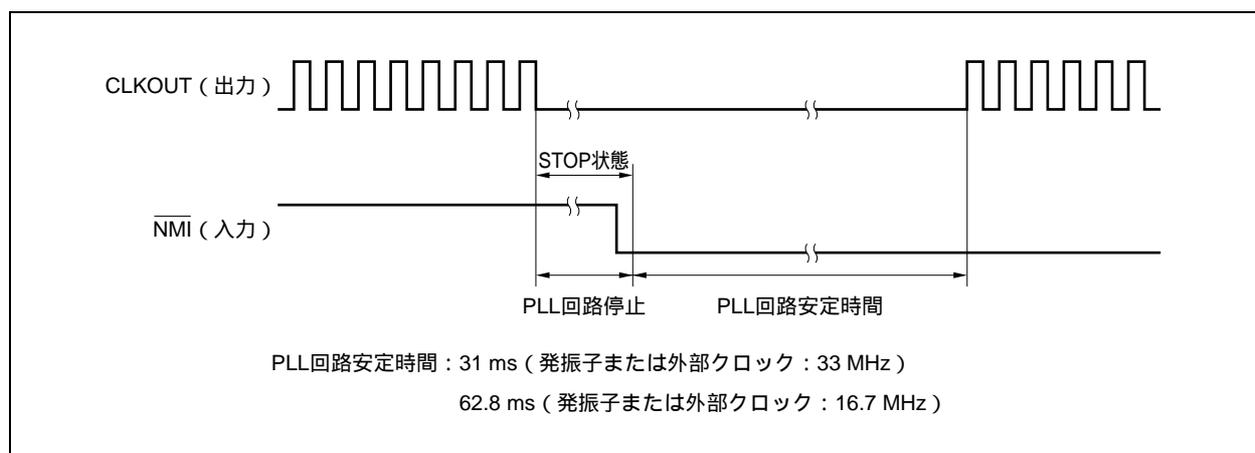
(1) 発振安定時間確保タイマで時間を確保する場合 ($\overline{\text{NMI}}$ 信号入力)

$\overline{\text{NMI}}$ 端子に有効エッジを入力すると、STOPモードが解除されます。

その場合、発振安定時間は内蔵の発振安定時間確保タイマにより自動的にクロック出力が安定するまでの時間を確保します。

所定時間後に、クロック出力が開始し、NMI処理のハンドラ・アドレスに分岐します。

図13-2 STOPモード解除タイミング ($\overline{\text{NMI}}$ 信号入力時)

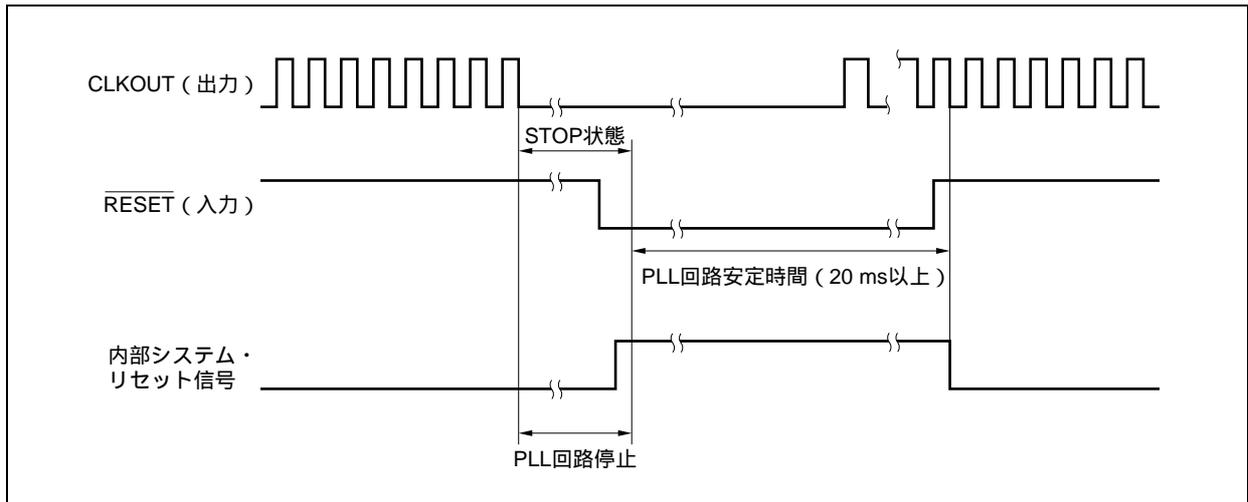


(2) 信号レベル幅で時間を確保する場合 (RESET信号入力)

RESET端子への立ち下がりエッジ入力により、STOPモードを解除します。RESET端子に入力する信号のロウ・レベル幅は、発振回路からのクロック出力が安定するまで時間を確保してください。PLLの安定時間を確保するためには、ロウ・レベル幅を20 ms以上入力してください。

RESET端子への立ち上がりエッジ入力後、クロックの供給を開始して、システム・リセット時のハンドラ・アドレスに分岐します。

図13 - 3 STOPモード解除タイミング (RESET信号入力時)



第14章 リセット/NMI制御機能

リセット/NMI制御機能は、システム・コントロール・ユニット (SYU) にて実現します。システム・コントロール・ユニットは、 $\overline{\text{RESET}}$ 信号と $\overline{\text{NMI}}$ 信号をコントロールする回路です。

14.1 特 徴

$\overline{\text{RESET}}$ 端子、 $\overline{\text{NMI}}$ 端子に外部入力クロック・サンプリングによるノイズ除去回路を内蔵しています。

ディバグ・コントロール・ユニットからの強制リセット、リセット・マスク、NMIマスク処理を行います。

14.2 ノンマスカブル割り込み (NMI)

外部入力クロック (STOPモードでも停止しないクロックであり、バス・クロックと同周期のクロック) の立ち上がりで $\overline{\text{NMI}}$ 信号をサンプリングします。

5外部入力クロック未満のノイズを除去してからエッジを検出します。したがって、 $\overline{\text{NMI}}$ 信号は5外部入力クロック分以上のロウ・レベル幅が必要です。

$\overline{\text{NMI}}$ 信号がハイ・レベルからロウ・レベルに変化したとき、割り込みを検出します。 $\overline{\text{NMI}}$ 信号は、立ち下がりエッジ検出であるため、一度ノンマスカブル割り込み要求を検出したあとはいつでもインアクティブにできます。検出した割り込みは要求は、CPUが割り込み処理を開始するまでCPU内部で保持されます。

14.3 リセット

$\overline{\text{RESET}}$ 端子にロウ・レベルが入力されるとシステム・リセットがかかり、オンチップの各ハードウェアは初期状態にイニシャライズされます。

$\overline{\text{RESET}}$ 端子がロウ・レベルからハイ・レベルになると、リセット状態が解除され、CPUはプログラムの実行を開始します。各種レジスタの内容は、プログラムの中で必要に応じてイニシャライズしてください。

$\overline{\text{RESET}}$ 信号は、5外部入力クロック分未満のノイズ幅を除去してからエッジを検出します。したがって、5外部入力クロック分以上のロウ・レベル幅が必要です。そのため、CPUコアに対するリセット・イネーブルの最低幅 (20クロック分) を満足させるには、外部リセット信号幅は25クロック分以上必要になります。

★ リセット解除後から、最初の $\overline{\text{BCYST}}$ 信号がアクティブになるまでの最小クロック数は10バス・クロックとなります。

14.3.1 端子機能

システム・リセット期間中とリセット直後の出力端子の状態を表14-1に示します。リセット期間中は、この状態を保持します。

$\overline{\text{RESET}}$ 端子を25外部入力クロック以上のロウ・レベルを保持したあとインアクティブにして、 $\overline{\text{HLDRQ}}$ 信号がインアクティブであれば、命令フェッチのためのメモリ・リード・サイクルを起動します。

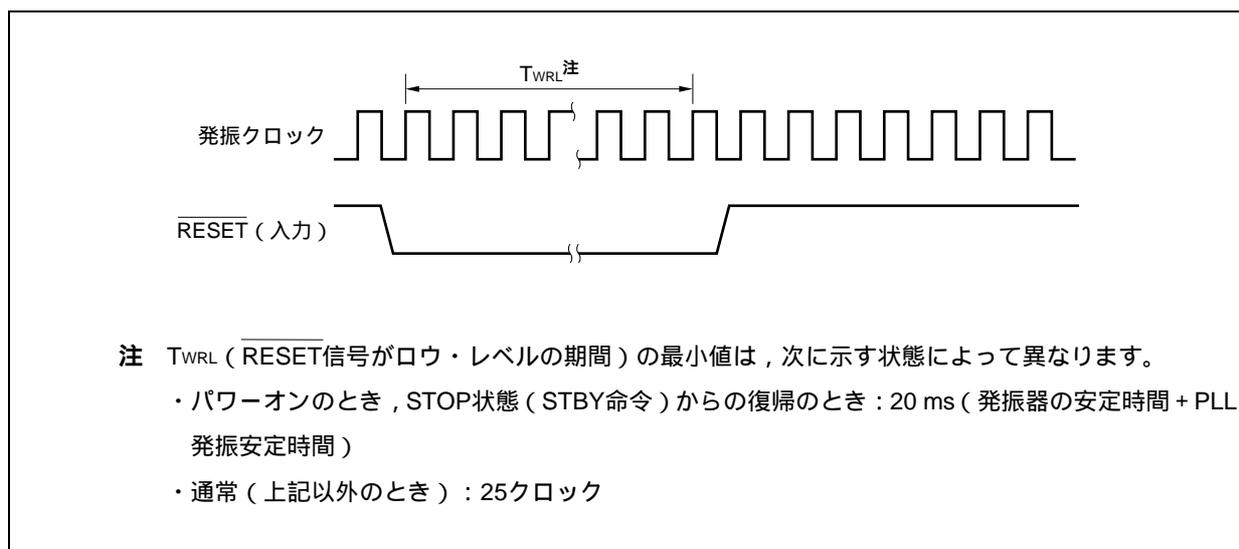
パワーオン・リセット時は、 $\overline{\text{HLDRQ}}$ 信号をインアクティブにしてください。リセットによりハイ・インピーダンスになる端子については、プルアップまたはプルダウン抵抗を付けてください。これらの処理をしない場合、端子がハイ・インピーダンスになるとメモリなどの内容を破壊するおそれがあります。

また、リセット期間中でもCLKOUT端子からのクロックは出力されます。

表14 - 1 リセット直後の出力端子の状態

機 能	動作状態
A1-A23	不定
D0-D31	ハイ・インピーダンス
CS1-CS7	1
BCYST	1
IORD, LOWR	1
WE, OE	1
LLMWR, LUMWR, ULMWR, UUMWR	1
LLCAS, LUCAS, ULCAS, UUCAS	1
RAS	1
CLKOUT	クロック出力
HLDAK	1
DMAAK0-DMAAK3	1
PORT2/SI	ハイ・インピーダンス
PORT1/SO	ハイ・インピーダンス
PORT0/SCLK	ハイ・インピーダンス
TXD	1
DDO	不定
TRCDATA0-TRCDATA3	不定
TC/REFRQ	1
TO10/INTP10, TO11/INTP12	ハイ・インピーダンス

図14 - 1 リセット信号の受け付け



14.3.2 イニシャライズ

各レジスタのリセット後の初期値を表14-2に示します。

各レジスタの内容はプログラム中で必要に応じてイニシャライズしてください。特にクロック・コントロール・レジスタ (CGC) は、システム設定 (X1, X2端子機能, CLKOUT端子動作など) に関するレジスタなので注意してください。

表14-2 各レジスタのリセット後の初期値

レジスタ		リセット後の初期値	
システム・レジスタ	プログラム・カウンタ	PC	FFFFFFF0H
	例外 / 割り込み時状態退避レジスタ	EIPC	不 定
		EIPSW	不 定
	NMI / 二重例外時状態退避レジスタ	FEPC	不 定
		FEPS	不 定
	例外要因レジスタ	ECR	0000FFF0H
	プログラム・ステータス・ワード	PSW	00008000H
	プロセッサIDレジスタ ^注	PIR	00008301H
	タスク・コントロール・ワード	TKCW	000000E0H
	デバッグ例外時状態退避レジスタ	DPC	不 定
DPSW		不 定	
ハードウェア・コンフィギュレーション・コントロール・レジスタ	HCCW	00000000H	
内部レジスタ	PLLコントロール・レジスタ ^注	PLLCR	00000002H
	キャッシュ・メモリ・コントロール・レジスタ	CMCR	00000000H
	命令キャッシュ・タグ・レジスタ	ICTR	xxxxx000H
	データ・キャッシュ・タグ・レジスタ	DCTR	xxxxx000H
	命令RAMレジスタ	IRAMR	不 定

注 V831では、初期値のまま固定になります。

(メ モ)

第15章 ディバグ/トレース機能

V831は、ディバグ/トレース機能を実現するディバグ・コントロール・ユニット（DCU）を備えています。

15.1 特徴

ディバグ用信号装備：10本（専用： $\overline{\text{DRST}}$, DCK, DMS, DDI, DDO, TRCDATA0-TRCDATA3）

（兼用：CLKOUT）

- ・ユーザ基板にディバグ用の配線，コネクタを付けると，オンチップでのディバグが可能（ただし，致命的例外ハンドラ（FFFFFFE0H-FFFFFFEFH）は使用できなくなります。）
- ・ディバグ装置を介してホスト・マシンとのインタフェースを行うディバグ・インタフェースを装備
- ・ユーザのプログラムの実行状況を監視するトレース・インタフェースを装備

ディバグ機能

- ・強制リセット機能（CPUコアおよび周辺機能を強制的にリセット可能）
- ・強制ブレーク機能（ユーザ・プログラムの実行を強制的に中断可能）
- ・ユーザ・プログラムの実行を任意のアドレスで中断可能
- ・ユーザ・プログラムを任意のアドレスから実行開始可能
- ・ユーザ・プログラム中断中にユーザ・リソース（メモリ，I/Oなど）のリード/ライト可能
- ・ユーザ・プログラムのダウンロード可能
- ・マスク機能（外部入力信号（ $\overline{\text{RESET}}$, $\overline{\text{HLDRQ}}$, $\overline{\text{NMI}}$, INTP00-INTP03, INTP10-INTP13）をマスク可能）

トレース機能

- ・PCトレース（分岐トレース）
 - ユーザ・プログラム実行中に発生したすべての分岐（処理の遷移）をトレース可能
 - 機能別に分別された9種類の分岐からトレース要因を選択可能
- ・データ・トレース
 - 外部バス，内蔵周辺I/Oバスを経由するすべてのデータ・アクセスをトレース可能（CPUコア内蔵のデータRAM，データ・キャッシュ，システムI/Oはトレースできません。）
 - ライト・アクセスではライト・データをトレース可能
- ・リアルタイムでのトレースが可能
- ・トレースの強制スタート/強制ストップ，任意の実行PCからのトレース・スタートが可能
- ・トレース・バッファを装備（PCトレース，データ・トレース兼用）
 - 最小8要因から最大40要因までのトレース・データを格納可能

(メ モ)

付録A レジスタ索引

A.1 50音順

【あ行】

ICUモード・レジスタ (IMOD) ...	50, 161
アシンクロナス・シリアル・インタフェース・ステータス・レジスタ (ASIS0) ...	133
アシンクロナス・シリアル・インタフェース・モード・レジスタ00 (ASIM00) ...	130
アシンクロナス・シリアル・インタフェース・モード・レジスタ01 (ASIM01) ...	132

【か行】

キャプチャ/コンペア・レジスタ (CC10-CC13) ...	154
クロック・コントロール・レジスタ (CGC) ...	173
クロック同期式シリアル・インタフェース・モード・レジスタ0 (CSIM0) ...	140
コンペア・レジスタ (CM4) ...	153

【さ行】

受信バッファ (RXB0, RXB0L) ...	134
シリアル/I/Oシフト・レジスタ0 (SIO0) ...	141
送信シフト・レジスタ (TXS0, TXS0L) ...	135

【た行】

タイマ1 (TM1) ...	151
タイマ4 (TM4) ...	153
タイマ・オーバフロー・ステータス・レジスタ (TOVS) ...	161
タイマ・コントロール・レジスタ1 (TMC1) ...	158
タイマ・コントロール・レジスタ4 (TMC4) ...	159
タイマ出力コントロール・レジスタ (TOC1) ...	160
タイマ・ユニット・モード・レジスタ (TUM1) ...	156
DMAコントロール・レジスタ (DC) ...	111
DMAソース・アドレス・レジスタ0H-3H (DSA0H-DSA3H) ...	103
DMAソース・アドレス・レジスタ0L-3L (DSA0L-DSA3L) ...	104
DMAチャンネル・コントロール・レジスタ0-3 (DCHC0-DCHC3) ...	108
DMAデスティネーション・アドレス・レジスタ0H-3H (DDA0H-DDA3H) ...	105
DMAデスティネーション・アドレス・レジスタ0L-3L (DDA0L-DDA3L) ...	106
DMAバイト・カウント・レジスタ0-3 (DBC0-DBC3) ...	106
DRAMコンフィギュレーション・レジスタ (DRC) ...	96
データ・バス幅コントロール・レジスタ (DBC) ...	90

【な行】

入出力ポート・レジスタ (PORT) ... 169

入出力モード・レジスタ (PM) ... 169

【は行】

バス・サイクル・タイプ・コントロール・レジスタ (BCTC) ... 89

PLLコントロール・レジスタ (PLLCR) ... 174

プログラマブル・アイドル・コントロール・レジスタ (PIC) ... 93

プログラマブル・ウェイト・コントロール・レジスタ0 (PWC0) ... 91

プログラマブル・ウェイト・コントロール・レジスタ1 (PWC1) ... 92

Page-ROMコンフィギュレーション・レジスタ (PRC) ... 100

ポート・コントロール・モード・レジスタ (PC) ... 170

ポー・レート・ジェネレータ・コンペア・レジスタ (BRG0) ... 147

ポー・レート・ジェネレータ・プリスケアラ・モード・レジスタ (BPRM0) ... 147

【ら行】

リフレッシュ・コントロール・レジスタ (RFC) ... 98

【わ行】

割り込みクリア・レジスタ (ICR) ... 48

割り込みグループ優先順位レジスタ (IGP) ... 47

割り込み要求マスク・レジスタ (IMR) ... 49

割り込み要求レジスタ (IRR) ... 49

A.2 アルファベット順

【A】

ASIM00 (アシンクロナス・シリアル・インタフェース・モード・レジスタ00)	...	130
ASIM01 (アシンクロナス・シリアル・インタフェース・モード・レジスタ01)	...	132
ASIS0 (アシンクロナス・シリアル・インタフェース・ステータス・レジスタ)	...	133

【B】

BCTC (バス・サイクル・タイプ・コントロール・レジスタ)	...	89
BPRM0 (ポー・レート・ジェネレータ・プリスケラ・モード・レジスタ)	...	147
BRG0 (ポー・レート・ジェネレータ・コンペア・レジスタ)	...	147

【C】

CC10-CC13 (キャプチャ/コンペア・レジスタ)	...	154
CGC (クロック・コントロール・レジスタ)	...	173
CM4 (コンペア・レジスタ)	...	153
CSIM0 (クロック同期式シリアル・インタフェース・モード・レジスタ0)	...	140

【D】

DBC (データ・バス幅コントロール・レジスタ)	...	90
DBC0-DBC3 (DMAバイト・カウント・レジスタ0-3)	...	106
DC (DMAコントロール・レジスタ)	...	111
DCHC0-DCHC3 (DMAチャンネル・コントロール・レジスタ0-3)	...	108
DDA0H-DDA3H (DMAデスティネーション・アドレス・レジスタ0H-3H)	...	105
DDA0L-DDA3L (DMAデスティネーション・アドレス・レジスタ0L-3L)	...	106
DRC (DRAMコンフィギュレーション・レジスタ)	...	96
DSA0H-DSA3H (DMAソース・アドレス・レジスタ0H-3H)	...	103
DSA0L-DSA3L (DMAソース・アドレス・レジスタ0L-3L)	...	104

【I】

ICR (割り込みクリア・レジスタ)	...	48
IGP (割り込みグループ優先順位レジスタ)	...	47
IMOD (ICUモード・レジスタ)	...	50, 161
IMR (割り込み要求マスク・レジスタ)	...	49
IRR (割り込み要求レジスタ)	...	49

【P】

PC (ポート・コントロール・モード・レジスタ)	...	170
PIC (プログラマブル・アイドル・コントロール・レジスタ)	...	93
PLLCR (PLLコントロール・レジスタ)	...	174
PM (入出力モード・レジスタ)	...	169
PORT (入出力ポート・レジスタ)	...	169
PRC (Page-ROMコンフィギュレーション・レジスタ)	...	100

PWC0 (プログラマブル・ウェイト・コントロール・レジスタ0) ...	91
PWC1 (プログラマブル・ウェイト・コントロール・レジスタ1) ...	92

【R】

RFC (リフレッシュ・コントロール・レジスタ) ...	98
RXB0, RXB0L (受信バッファ) ...	134

【S】

SIO0 (シリアルI/Oシフト・レジスタ0) ...	141
-----------------------------	-----

【T】

TM1 (タイマ1) ...	151
TM4 (タイマ4) ...	153
TMC1 (タイマ・コントロール・レジスタ1) ...	158
TMC4 (タイマ・コントロール・レジスタ4) ...	159
TOC1 (タイマ出力コントロール・レジスタ) ...	160
TOVS (タイマ・オーバフロー・ステータス・レジスタ) ...	161
TUM1 (タイマ・ユニット・モード・レジスタ) ...	156
TXS0, TXS0L (送信シフト・レジスタ) ...	135

付録B 総合索引

B.1 50音で始まる語句の索引

【あ】

アイドル・ステート ... 79
 アシクロナス・シリアル・インタフェース
 ... 127
 アドレス空間とブロック ... 88
 アドレス・バス ... 30
 アドレス・マルチプレクス機能 ... 96

【い】

イニシャライズ ... 183

【う】

ウェイト制御レジスタ ... 89

【か】

外部I/Oサイクル ... 53
 外部入力端子による割り込み要求 ... 52

【き】

キャプチャ/コンペア・レジスタ ... 154
 強制中断 ... 124

【く】

クロック・コントロール・レジスタ ... 173
 クロック出力禁止モード ... 172
 クロック出力制御 ... 172
 クロック制御レジスタ ... 173
 クロック同期式シリアル・インタフェース ... 139

【し】

システム制御信号 ... 32
 シリアル制御信号 ... 35
 シングル転送モード ... 112

【す】

スタンバイ・モード ... 175

【そ】

ソフトウェアからの要求 ... 120

【た】

タイマ1 ... 151
 タイマ4 ... 153
 タイマ・オーバフロー・ステータス・レジスタ
 ... 161
 タイマ/カウンタ制御レジスタ ... 156
 タイマ・コントロール・レジスタ1 ... 158
 タイマ・コントロール・レジスタ4 ... 159
 タイマ出力コントロール・レジスタ ... 160
 タイマ・ユニット・モード・レジスタ ... 156
 端子接続図 (Top View) ... 21

【ち】

致命的例外ルーチンからの復帰 ... 46

【て】

ディバグ制御信号 ... 35
 ディバグ/トレース機能 ... 185
 デiamond転送モード ... 113
 データ・バス ... 30
 データ・バス幅コントロール・レジスタ ... 90
 転送モード ... 112

【な】

内蔵周辺I/O空間 ... 37
 内蔵周辺ハードウェアからの要求 ... 120
 内部ブロック構成 ... 23
 内部ユニット ... 24

【に】

入出力ポート・レジスタ ... 169
 入出力モード・レジスタ ... 169
 入力クロック選択 ... 172

【の】

ノンマスクابل割り込み ... 41, 181

【は】

バス・アービトレーション ... 85
 バス・サイクル・タイプ・コントロール・レジスタ
 ... 89
 バス・サイジング ... 80
 バス制御信号 ... 30
 バス・ホールド・サイクル ... 84
 発振安定時間の確保 ... 179

【ふ】

プログラマブル・アイドル・コントロール・レジスタ
 ... 93
 プログラマブル・ウェイト・コントロール・レジスタ0
 ... 91
 プログラマブル・ウェイト・コントロール・レジスタ1
 ... 92

【ほ】

ポート・コントロール・モード・レジスタ ... 170
 ポート制御信号 ... 35
 ポート制御レジスタ ... 169
 ボー・レート・ジェネレータ ... 144
 ボー・レート・ジェネレータ・コンペア・レジスタ
 ... 147
 ボー・レート・ジェネレータ・プリスケラ・モード・レジスタ
 ... 147

【ま】

マスクابل割り込み ... 42
 マスクابل割り込みの優先順位 ... 44

【り】

リアルタイム・パルス制御信号 ... 34
 リセット ... 181
 リフレッシュ機能 ... 98

【れ】

例外処理 ... 45
 例外/割り込みからの復帰 ... 46

【ろ】

ロックアップ時間 ... 172

【わ】

割り込みクリア・レジスタ ... 48
 割り込みグループ優先順位レジスタ ... 47
 割り込み制御信号 ... 32
 割り込み制御レジスタ ... 47
 割り込み要求マスク・レジスタ ... 49
 割り込み要求レジスタ ... 49
 割り込み/例外処理 ... 39

B.2 アルファベットで始まる語句の索引

【A】

A1-A23 ... 30

【B】

BCTC ... 89

 $\overline{\text{BCYST}}$... 31

BPRM0 ... 147

BRG0 ... 147

BT16B ... 31

【C】

CC10-CC13 ... 154

CGC ... 173

CLKOUT ... 32

CPUコア・システム・レジスタ ... 38

CS1-CS7 ... 32

CSI ... 139

【D】

D0-D31 ... 30

DBC ... 90

DBC0-DBC3 ... 106

DC ... 111

DCHC0-DCHC3 ... 108

DCK ... 35

DDA0-DDA3 ... 105

DDI ... 35

DDO ... 35

DMAAK0-DMAAK3 ... 34

DMARQ0-DMARQ3 ... 34

DMARQ端子からの要求 ... 115

DMAコントロール・レジスタ ... 111

DMA制御信号 ... 34

DMA制御レジスタ ... 103

DMAソース・アドレス・レジスタ0-3 ... 103

DMAチャンネル・コントロール・レジスタ0-3 ... 108

DMAチャンネルの優先順位 ... 115

DMAデスティネーション・アドレス・レジスタ0-3
... 105

DMA転送終了出力 ... 122

DMA転送終了割り込み ... 121

DMA転送タイプと転送対象 ... 113

DMA転送のバス・サイジング ... 125

DMA転送要求 ... 115

DMAバイト・カウント・レジスタ0-3 ... 106

DMS ... 36

DRAMコンフィギュレーション・レジスタ ... 96

DRAMサイクル ... 65

DRAM制御機能 ... 95

DRAM制御信号 ... 33

DRC ... 96

 $\overline{\text{DRST}}$... 36

DSA0-DSA3 ... 103

【G】

GND ... 28

GND_PLL ... 28

【H】

HALTモード ... 176

HALTモードの解除 ... 177

HALTモードの設定と動作状態 ... 176

 $\overline{\text{HLDK}}$... 30 $\overline{\text{HLDRQ}}$... 30 $\overline{\text{HLDRQ}}$ 信号, リフレッシュによる一時中断
... 124

【I】

ICR ... 48

ICUモード・レジスタ ... 50, 161

IGP ... 47

IMOD ... 50, 161

IMR ... 49

INTP00-INTP03 ... 32

INTP10-INTP13 ... 32

 $\overline{\text{IORD}}$... 31

IOWR ... 31

IRR ... 49

【L】

 $\overline{\text{LLCAS}}$... 33 $\overline{\text{LLMWR}}$... 30

- $\overline{\text{LUCAS}}$... 33
 $\overline{\text{LUMWR}}$... 31
- 【M】**
 $\overline{\text{MRD}}$... 30
- 【N】**
 $\overline{\text{NMI}}$... 32
 $\overline{\text{NMI}}$ 信号による中断 ... 124
- 【O】**
 $\overline{\text{OE}}$... 33
on-page/off-pageの判断 ... 96
- 【P】**
Page-ROMコンフィギュレーション・レジスタ
... 100
Page-ROMサイクル ... 62
Page-ROM制御機能 ... 100
PC ... 170
PIC ... 93
PLLCR ... 174
PLLコントロール・レジスタ ... 174
PM ... 169
PORT ... 169
PORT0-PORT2 ... 35
PRC ... 100
PWC0 ... 91
PWC1 ... 92
- 【R】**
 $\overline{\text{RAS}}$... 33
 $\overline{\text{READY}}$... 30
 $\overline{\text{READY}}$ 端子によるウエイト制御 ... 94
 $\overline{\text{REFRQ}}$... 33
 $\overline{\text{RESET}}$... 32
RXD ... 35
- 【S】**
 $\overline{\text{SCLK}}$... 35
SI ... 35
SO ... 35
SRAM (ROM) サイクル ... 55
- STOPモード ... 178
STOPモードの解除 ... 179
STOPモードの設定と動作状態 ... 178
- 【T】**
 $\overline{\text{TC}}$... 34
TCLR ... 34
TI ... 34
TMC1 ... 158
TMC4 ... 159
TO10 ... 34
TO11 ... 34
TOC1 ... 160
TOVS ... 161
TRCDATA0-TRCDATA3 ... 36
TUM1 ... 156
TXD ... 35
- 【U】**
UART ... 127
 $\overline{\text{ULCAS}}$... 33
 $\overline{\text{ULMWR}}$... 31
 $\overline{\text{UUCAS}}$... 33
 $\overline{\text{UUMWR}}$... 31
- 【V】**
 V_{DD} ... 28
 $V_{\text{DD_PLL}}$... 28
- 【W】**
 $\overline{\text{WE}}$... 33
- 【X】**
X1, X2 ... 32

(メ モ)

— お問い合わせ先 —

【技術的なお問い合わせ先】

NEC 半導体テクニカルホットライン (インフォメーションセンター)
 (電話: 午前 9:00 ~ 12:00, 午後 1:00 ~ 5:00)

電話 : 044-548-8899
 FAX : 044-548-7900
 E-mail : s-info@saed.tmg.nec.co.jp

【営業関係お問い合わせ先】

半導体第一販売事業部 半導体第二販売事業部 半導体第三販売事業部	〒108-8001	東京都港区芝5-7-1 (日本電気本社ビル)			(03)3454-1111			
中部支社 半導体第一販売部 半導体第二販売部	〒460-8525	愛知県名古屋市中区錦1-17-1 (日本電気中部ビル)			(052)222-2170 (052)222-2190			
関西支社 半導体第一販売部 半導体第二販売部 半導体第三販売部	〒540-8551	大阪府大阪市中央区城見1-4-24 (日本電気関西ビル)			(06) 945-3178 (06) 945-3200 (06) 945-3208			
北海道支社 東北支社 岩手支店 郡山支店 いわき支店 長岡支店 水戸支店 土浦支店 群馬支店 太田支店	札幌 仙台 盛岡 郡山 いわき 長岡 水戸 土浦 高崎 太田	(011)251-5599 (022)267-8740 (019)651-4344 (0249)23-5511 (0246)21-5511 (0258)36-2155 (029)226-1717 (0298)23-6161 (027)326-1255 (0276)46-4011	宇都宮支店 小山支店 甲府支店 長野支社 静岡支社 立川支社 埼玉支社 千葉支社 神奈川支社 三重支店	宇都宮 小山 甲府 松本 静岡 立川 大宮 千葉 横浜 津	(028)621-2281 (0285)24-5011 (0552)24-4141 (0263)35-1662 (054)254-4794 (042)526-5981,6167 (048)649-1415 (043)238-8116 (045)682-4524 (059)225-7341	北陸支社 京都支社 神戸支社 中国支社 鳥取支店 岡山支店 松山支店 九州支社	金沢 京都 神戸 広島 鳥取 岡山 松山 福岡	(076)232-7303 (075)344-7824 (078)333-3854 (082)242-5504 (0857)27-5311 (086)225-4455 (089)945-4149 (092)261-2806

アンケート記入のお願い

お手数ですが、このドキュメントに対するご意見をお寄せください。今後のドキュメント作成の参考にさせていただきます。

[ドキュメント名] V831 ユーザーズ・マニュアル ハードウェア編
(U12273JJ4V0UM00 (第4版))

[お名前など] (さしつかえのない範囲で)
御社名(学校名, その他) ()
ご住所 ()
お電話番号 ()
お仕事の内容 ()
お名前 ()

1. ご評価 (各欄に をご記入ください)

項 目	大変良い	良 い	普 通	悪 い	大変悪い
全体の構成					
説明内容					
用語解説					
調べやすさ					
デザイン, 字の大きさなど					
その他 ()					
()					

2. わかりやすい所 (第 章, 第 章, 第 章, 第 章, その他)
理由 []

3. わかりにくい所 (第 章, 第 章, 第 章, 第 章, その他)
理由 []

4. ご意見, ご要望
[]

5. このドキュメントをお届けしたのは
NEC販売員, 特約店販売員, NEC半導体ソリューション技術本部員,
その他 ()

ご協力ありがとうございました。
下記あてにFAXで送信いただくか、最寄りの販売員にコピーをお渡しください。