

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。

保守/廃止

V25+™, V35+™

16/8, 16ビット・シングルチップ・マイクロコンピュータ

ハードウェア編

μPD70325
μPD70325(A)
μPD70335
μPD70335(A)

保守／廃止

概 説	1
端子機能	2
内部CPU機能	3
割り込み機能	4
バス制御	5
DMAコントローラ	6
ポート機能	7
クロック発生回路	8
タイマ・ユニット	9
タイム・ベース・カウンタ	10
シリアル・インタフェース	11
スタンバイ機能	12
リセット機能	13
アドレス生成	14
動作状態の遷移	15
付 録	付

CMOSデバイスの一般的注意事項**① 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。

② 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。

③ 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。

V20, V25, V25+, V30, V35, V35+は日本電気株式会社の商標です。

インターツールは米国Intermetrics Microsystems Software, Inc.の商標です。

- 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
旧版p. 235-582	旧版の第15章 命令の説明をV25, V35ファミリー ユーザーズ・マニュアル 命令編へ分冊
p. 6-9	1.4 端子接続図の注意を修正
p. 32	2.3.2 入出力回路に注を追加
p. 33	2.4 未使用端子の処理に説明を追加
p. 38-41	3.1.4 CPUパイプラインを追加
p. 70, 71	4.4 割り込み要求を追加
p. 73-78	4.5.2 レジスタ・バンク切り替え機能に説明を追加
p. 80, 82, 84	4.5.3 マクロ・サービス機能に図4-7, 図4-9, 図4-11, (3) マクロ・サービス完了割り込みを追加
p. 94, 95	4.10 割り込みプライオリティ・レジスタ (ISPR) に説明, 注意, 図4-18を追加
p. 101	4.13 割り込み要求が受け付けられない場合を追加
p. 109, 110	4.16 (7) 外部割り込みの優先順位を追加
p. 145	図5-6 RFMを追加
p. 159	6.2 DMA動作に説明を追加
p. 180	図6-15 IORD信号, IOWR信号作成回路例を修正
p. 190	7.1.2 (4) 注意事項を追加
p. 193	7.2.3 ポートT受信を追加
p. 201, 202	9.2 タイマ・コントロール・レジスタ (TMCO, TMC1) に説明, 図9-3を追加
p. 216, 218	11.2.2 送信完了割り込み, 11.3.2 送信完了割り込みに説明を追加
p. 219, 220	11.4 割り込みを使用しない場合の送信の起動を追加
p. 224, 225	11.6 ボー・レート・ジェネレータに説明を追加
p. 259, 260	A.6 マクロ・サービスのマスクに関する制限事項を追加
p. 261, 262	A.7 CVTBD/CVTDB命令に関する制限事項を追加
p. 263-290	付録B 設定例を追加

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・コーナーを設けております。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、 μ PD70325, 70335 (別名称V25+, V35+) の機能を理解し、それを用いたアプリケーション・システムを設計するユーザのエンジニアを対象とします。
- 目的** このマニュアルは、次の構成に示す μ PD70325, 70335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ます。
- 構成** μ PD70325, 70335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) と命令編の2冊に分かれています。

ハードウェア編

- ・概 説
- ・端子機能
- ・内部CPU機能
- ・内部周辺デバイス
- ・スタンバイ機能
- ・リセット機能
- ・付録 (制限事項, Q&A)

命 令 編

- ・概 説
- ・命令の説明
- ・V20, V30に対する追加命令
- ・付 録

読み方 このマニュアルの読者には、電気、論理回路、マイクロコンピュータの一般知識を必要とします。

μ PD70108, 70116 (別名称V20TM, V30TM) をすでに経験しているユーザ

→ V25+, V35+は、V20, V30のネイティブ・モードの命令と互換性があります。

別冊の**V25, V35ファミリ ユーザズ・マニュアル 命令編の第3章 V20, V30**に対する追加命令を参照してください。

μ PD70320, 70330 (別名称V25TM, V35TM) をすでに経験しているユーザ

→ V25+, V35+は、V25, V35のROMレス版と同じ機能を持っていますので、**1.2 V25, V35との違い**を中心にお読みください。

命令機能の詳細を知りたいとき

→ 別冊の**V25, V35ファミリ ユーザズ・マニュアル 命令編**を参照してください。

一通りV25+, V35+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。

V25+, V35+の動作について何か不明な点が出たとき

→ 付録**C Q&A**をお読みください。

V25+, V35+の電気的特性を知りたいとき

→ 別冊の**データ・シート**を参照してください。

V25+, V35+の各種機能の応用例を知りたいとき

→ 別冊のアプリケーション・ノートを参照してください。

- 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 : $\overline{\times\times\times}$ (端子, 信号名称に上線)
 - メモリ・マップのアドレス : 上部-下位, 下部-上位
 - 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 $\dots\times\times\times\times$ または $\times\times\times\times B$

10進数 $\dots\times\times\times\times$

16進数 $\dots\times\times\times\times H$

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): $2^{20}=1024^2$

関連資料

V25+, V35+に関する資料

品名	資料名 データ・シート	ユーザズ・マニュアル		アプリケーション・ ノート	レジスタ 活用表	インフォメーション Q&A集
		ハードウェア編	命令編			
V25+	IC-7889	このマニュアル	IEM-847	—	IEM-5578	IEI-646
V35+	IC-7890			IEA-709		

開発ツールに関する資料

資料名	資料番号
IE-70325-BX, IE-70335-BX (インサーキット・エミュレータ)	EEU-728
RA70116- I (インターツール™・アセンブラ), SP70116- I (インターツール・ソフトウェア・パッケージ) アセンブリ言語編	EEU-861
RA70116- I (インターツール・アセンブラ), SP70116- I (インターツール・ソフトウェア・パッケージ) 操作編	EEU-869
LC70116 (ロケータ)	EEU-834
TD325-BX, TD335-BX (ターボ・ディバッガ)	EEU-851

目 次

第1章	概 説	… 1
1.1	特 徴	… 2
1.2	V25, V35との違い	… 4
1.3	オーダ情報と品質水準	… 5
1.4	端子接続図 (Top View)	… 6
1.5	内部ブロック図	… 11
1.5.1	μ PD70325	… 11
1.5.2	μ PD70335	… 12
第2章	端子機能	… 13
2.1	端子機能一覧	… 13
2.1.1	μ PD70325	… 13
2.1.2	μ PD70335	… 16
2.1.3	各モードにおける端子状態	… 19
2.2	各端子の機能説明	… 20
2.2.1	P00-P07 (Port 0)	… 20
2.2.2	P10-P17 (Port 1)	… 21
2.2.3	P20-P27 (Port 2)	… 23
2.2.4	PT0-PT7 (Port with Comparator 0-7)	… 25
2.2.5	V_{TH} (Threshold Voltage)	… 25
2.2.6	TxD0, TxD1 (Transmit Data 0, 1)	… 25
2.2.7	RxD0, RxD1 (Receive Data 0, 1)	… 25
2.2.8	$\overline{CTS0}$ (Clear to Send 0)	… 26
2.2.9	$\overline{CTS1}$ (Clear to Send 1)	… 26
2.2.10	\overline{RESET} (Reset)	… 26
2.2.11	\overline{EA} (External Access)	… 26
2.2.12	X1, X2 (Crystal)	… 27
2.2.13	D0-D15 (Data Bus 0-15)	… 27
2.2.14	A0-A19 (Address Bus 0-19)	… 27
2.2.15	\overline{MREQ} (Memory Request)	… 28
2.2.16	\overline{MSTB} (Memory Strobe)	… 28
2.2.17	R/\overline{W} (Read/Write Strobe)	… 28
2.2.18	\overline{REFRQ} (Refresh Request)	… 28
2.2.19	\overline{IOSTB} (I/O Strobe)	… 28
2.2.20	V_{DD} (Power Supply)	… 28
2.2.21	GND (Ground)	… 29
2.2.22	\overline{UBE} (Upper Byte Enable)	… 29
2.3	端子の入出力回路	… 30
2.3.1	入出力回路タイプ	… 30
2.3.2	入出力回路	… 32
2.4	未使用端子の処理	… 33

第3章 内部CPU機能 … 37

- 3.1 ハードウェア構成 … 37
 - 3.1.1 PAU (アドレス演算ユニット) … 37
 - 3.1.2 EXU (実行ユニット) … 37
 - 3.1.3 BCU (バス・コントロール・ユニット) … 37
 - 3.1.4 CPUパイプライン … 38
- ★ 3.2 レジスタ … 41
 - 3.2.1 レジスタ・バンク … 41
 - 3.2.2 汎用レジスタ (AW, BW, CW, DW) … 43
 - 3.2.3 ポインタ (SP, BP) とインデクス・レジスタ (IX, IY) … 43
 - 3.2.4 セグメント・レジスタ (PS, SS, DS0, DS1) … 44
 - 3.2.5 内部データ領域ベース・レジスタ (IDB) … 45
 - 3.2.6 特殊機能レジスタ … 45
- 3.3 プログラム・カウンタ (PC) … 46
- 3.4 プログラム・ステータス・ワード (PSW) … 46
 - 3.4.1 CY (Carry Flag) … 47
 - 3.4.2 P (Parity Flag) … 48
 - 3.4.3 AC (Auxiliary Flag) … 48
 - 3.4.4 Z (Zero Flag) … 48
 - 3.4.5 S (Sign Flag) … 49
 - 3.4.6 V (Overflow Flag) … 49
 - 3.4.7 $\overline{\text{IBRK}}$ (I/O Break Flag) … 50
 - 3.4.8 BRK (Break Flag) … 50
 - 3.4.9 IE (Interrupt Enable Flag) … 50
 - 3.4.10 DIR (Direction Flag) … 50
 - 3.4.11 RB0-RB2 (Register Bank 0-2 Flag) … 50
 - 3.4.12 FO, F1 (User Flag 0, 1 Flag) … 50
- 3.5 メモリ空間 … 52
 - 3.5.1 内部データ領域 … 54
 - 3.5.2 内部データ領域ベース・レジスタ (IDB) … 57
 - 3.5.3 特殊機能レジスタ領域 … 57
 - 3.5.4 内蔵RAM領域 … 61
 - 3.5.5 ベクタ・テーブル領域 … 62
 - 3.5.6 外部メモリ領域 … 63
- 3.6 I/O空間 … 64

第4章 割り込み機能 … 65

- 4.1 割り込みコントローラ … 65
- 4.2 割り込み要因 … 65
- 4.3 優先順位制御 … 68
 - 4.3.1 多重割り込みの優先順位制御 … 68
 - 4.3.2 同時に割り込みが発生したときの優先順位制御 … 69
- ★ 4.4 割り込み要求 … 70
- 4.5 割り込み応答方式 … 72
 - 4.5.1 ベクタ割り込み … 72

4.5.2	レジスタ・バンク切り替え機能	… 73	
4.5.3	マクロ・サービス機能	… 79	
4.5.4	マクロ・サービス制御レジスタ	… 85	
4.6	NMI (Non-Maskable Interrupt)	… 88	
4.7	INT (Interrupt)	… 89	
4.8	割り込み要求制御レジスタ	… 90	
4.9	割り込み要因レジスタ (IRQS)	… 92	
4.10	割り込みプライオリティ・レジスタ (ISPR)	… 94	
4.11	外部割り込み	… 96	
4.11.1	外部割り込みモード・レジスタ (INTM)	… 96	
4.11.2	外部割り込み要求制御レジスタ (EXICO, EXIC1, EXIC2)	… 97	
4.11.3	外部割り込みマクロ・サービス制御レジスタ (EMS0, EMS1, EMS2)	… 97	
4.12	ソフトウェアによる割り込み	… 99	
4.12.1	一般のソフトウェア割り込み	… 99	
4.12.2	入出力命令割り込み	… 100	
4.12.3	FPO命令割り込み	… 101	
4.13	割り込み要求が受け付けられない場合	… 101	★
4.14	割り込み要求が受け付けられないタイミング	… 102	
4.15	ブロック処理命令実行中の割り込み処理	… 102	
4.16	割り込み要求の受け付け方	… 103	
4.17	ハードウェア割り込みの応答時間	… 111	
4.17.1	V25+の割り込み応答時間 (システム・クロック数)	… 111	
4.17.2	V25+のNMI応答時間 (システム・クロック数)	… 113	
4.17.3	V25+のINT応答時間 (システム・クロック数)	… 114	
4.17.4	V35+の割り込み応答時間 (システム・クロック数)	… 116	
4.17.5	V35+のNMI応答時間 (システム・クロック数)	… 117	
4.17.6	V35+のINT応答時間 (システム・クロック数)	… 118	
第5章	バス制御	… 119	
5.1	プログラマブル・ウェイト機能	… 123	
5.2	バス・ホールド機能	… 141	
5.2.1	HLDRQ→ <u>HLDA</u> K応答時間 (単位:クロック)	… 142	
5.3	リフレッシュ機能	… 142	
5.3.1	リフレッシュ・モード・レジスタ (RFM)	… 142	
5.3.2	疑似SRAMとの接続	… 146	
5.3.3	DRAMとの接続	… 147	
5.4	バス使用权	… 148	
5.5	バス・タイミング	… 149	
5.5.1	μPD70325	… 149	
5.5.2	μPD70335	… 152	
第6章	DMAコントローラ	… 157	
6.1	端子機能	… 157	
6.2	DMA動作	… 157	

- 6.3 DMAの制御のためのレジスタ … 165
 - 6.3.1 DMAモード・レジスタ (DMAM0, DMAM1) … 165
 - 6.3.2 DMAコントロール・レジスタ (DMAC0, DMAC1) … 167
 - 6.3.3 DMA割り込み要求制御レジスタ (DICO, DIC1) … 168
- 6.4 DMAサービス・チャンネル … 169
 - 6.4.1 ソース・アドレス・ポインタ (SAR0, SAR1) … 169
 - 6.4.2 デスティネーション・アドレス・ポインタ (DAR0, DAR1) … 171
 - 6.4.3 ターミナル・カウンタ (TC0, TC1) … 173
- 6.5 DMA転送タイミング … 174
 - 6.5.1 μ PD70325 … 174
 - 6.5.2 μ PD70335 … 176
- 6.6 DMA実行時間 … 180
 - 6.6.1 V25+のDMA実行時間 (システム・クロック数) … 180
 - 6.6.2 V35+のDMA実行時間 (システム・クロック数) … 181

第7章 ポート機能 … 183

- 7.1 ポートO-2 … 183
 - 7.1.1 ハードウェア構成 … 183
 - 7.1.2 各ポートの機能 … 186
- 7.2 ポートT (PTO-PT7) … 191
 - 7.2.1 ハードウェア構成 … 191
 - 7.2.2 ポートTモード・レジスタ (PMT) … 192
 - 7.2.3 ポートT受信 … 193

★

第8章 クロック発生回路 … 195

- 8.1 クロック発生回路の構成 … 195
- 8.2 プロセッサ・コントロール・レジスタ (PRC) … 198

第9章 タイマ・ユニット … 199

- 9.1 タイマ・ユニットの構成と動作 … 199
- 9.2 タイマ・コントロール・レジスタ (TMCO, TMC1) … 201
- 9.3 タイマ・ユニット割り込み要求 … 207
 - 9.3.1 タイマ・ユニット割り込み要求制御レジスタ (TMICO, TMIC1, TMIC2) … 208
 - 9.3.2 タイマ・ユニット・マクロ・サービス制御レジスタ (TMMS0, TMMS1, TMMS2) … 208

第10章 タイム・ベース・カウンタ … 211

- 10.1 タイム・ベース・カウンタの構成 … 211
- 10.2 タイム・ベース・インターバルの指定 … 211
- 10.3 タイム・ベース割り込み要求制御レジスタ (TBIC) … 212

第11章 シリアル・インタフェース … 213

- 11.1 シリアル・インタフェースの構成 … 213
- 11.2 アシクロナス・モード … 215
 - 11.2.1 送信 … 215
 - 11.2.2 送信完了割り込み … 216
 - 11.2.3 受信 … 216
 - 11.2.4 受信完了割り込み … 217
- 11.3 I/Oインタフェース・モード … 218
 - 11.3.1 送信 … 218
 - 11.3.2 送信完了割り込み … 218
 - 11.3.3 受信 … 219
 - 11.3.4 受信エラー割り込み … 219
 - 11.3.5 シリアル・レジスタのクリア … 219
- 11.4 割り込みを使用しない場合の送信の起動 … 219 ★
- 11.5 シリアル・モード・レジスタ (SCMO, SCM1) … 220
- 11.6 ボー・レート・ジェネレータ … 224
 - 11.6.1 シリアル制御レジスタ (SCC0, SCC1) … 225
- 11.7 シリアル・エラーと送受信状態の検出 … 226
 - 11.7.1 シリアル・ステータス・レジスタ (SCS0, SCS1) … 227
- 11.8 ブレーク検出機能 … 229
- 11.9 シリアル・インタフェース割り込み要求 … 229
 - 11.9.1 割り込み要求制御レジスタ (SEICn, SRICn, STICn : n=0, 1) … 229
 - 11.9.2 マクロ・サービス制御レジスタ (SRMSn, STMSn : n=0, 1) … 230

第12章 スタンバイ機能 … 231

- 12.1 スタンバイ・コントロール・レジスタ (STBC) … 231
- 12.2 HALTモード … 232
 - 12.2.1 HALTモードの解除 … 232
- 12.3 STOPモード … 234
 - 12.3.1 STOPモードの解除 … 234

第13章 リセット機能 … 237

第14章 アドレス生成 … 239

- 14.1 命令アドレス … 239
 - 14.1.1 ダイレクト・アドレッシング … 239
 - 14.1.2 レラティブ・アドレッシング … 239
 - 14.1.3 レジスタ・アドレッシング … 240
 - 14.1.4 レジスタ・インダイレクト・アドレッシング … 240
 - 14.1.5 インデクスト・アドレッシング … 240
 - 14.1.6 ベースト・アドレッシング … 241
 - 14.1.7 ベースト・インデクスト・アドレッシング … 241
- 14.2 メモリ・オペランド・アドレス … 242
 - 14.2.1 レジスタ・アドレッシング … 242

14.2.2	イミューディエト・アドレッシング …	242
14.2.3	ダイレクト・アドレッシング …	243
14.2.4	レジスタ・インダイレクト・アドレッシング …	243
14.2.5	オートインクリメント/デクリメント・アドレッシング …	244
14.2.6	インデクスト・アドレッシング …	244
14.2.7	ベースト・アドレッシング …	245
14.2.8	ベースト・インデクスト・アドレッシング …	246
14.2.9	ビット・アドレッシング …	246
14.2.10	特殊機能レジスタ・アドレッシング …	247

第15章 動作状態の遷移 … 249

付録A 制限事項 … 251

A.1	マクロ・サービスとINTの競合に関する制限事項 …	251
A.1.1	制限事項内容 …	251
A.1.2	回避方法 …	251
A.2	MOVSPA命令の使用上の注意事項 …	252
A.2.1	デバイスの動作説明（現象） …	252
A.2.2	回避方法 …	254
A.3	シリアル送信中の送信禁止操作による送信データ抜けに関する制限事項 …	255
A.3.1	デバイスの動作説明（現象） …	255
A.3.2	回避方法 …	255
A.4	割り込みの優先順位と処理順序に関する注意事項 …	256
A.5	タイマのダウン・カウント開始に関する注意事項 …	257
A.5.1	デバイスの動作説明（現象） …	257
A.5.2	回避方法 …	257
★ A.6	マクロ・サービスのマスクに関する制限事項 …	259
A.6.1	デバイスの動作説明（現象） …	259
A.6.2	回避方法 …	259
★ A.7	CVTBD/CVTDB命令に関する制限事項 …	261
A.7.1	デバイスの動作説明（現象） …	261
A.7.2	回避方法 …	261

★ **付録B 設定例 … 263**

B.1	ポート …	263
B.2	プログラマブル・ウェイト，プロセッサ・コントロール，リフレッシュ機能 …	264
B.3	レジスタ・バンクの切り替え …	267
B.3.1	割り込み要求によるレジスタ・バンクの切り替え …	267
B.3.2	命令によるレジスタ・バンクの切り替え（BRKCS命令とMOVSPA命令） …	269
B.3.3	命令によるレジスタ・バンクの切り替え（MOVSPB命令とTSKSW命令） …	269
B.4	内部データ領域のアクセス …	272
B.5	タイマ・ユニット …	273
B.6	I/Oインタフェース・モード …	277
B.7	マクロ・サービス …	280
B.7.1	ノーマル・モード（シリアル・インタフェースUART送信） …	280

B.7.2	キャラクタ・サーチ・モード (シリアル・インタフェースUART送信)	…	283
B.8	DMAコントローラ	…	287
B.8.1	ダイヤモンド・リリース・モード	…	287
B.8.2	シングルステップ・モード	…	288
B.8.3	バースト・モード	…	289
付録C	Q&A	…	291
C.1	内部CPU機能	…	296
C.2	割り込み機能	…	298
C.3	バス制御	…	305
C.4	DMAコントローラ	…	311
C.5	クロック発生回路	…	319
C.6	タイマ・ユニット	…	320
C.7	シリアル・インタフェース	…	321
C.8	スタンバイ機能	…	325
C.9	リセット機能	…	326
C.10	その他	…	328
付録D	レジスタ索引 (アルファベット順)	…	331

{ × 毛 }

図 の 目 次 (1/4)

図番号	タイトル, ページ
3-1	CPUパイプラインの動作例 … 39
3-2	レジスタ・バンクの構成 … 42
3-3	FLAG … 51
3-4	メモリ・マップ … 53
3-5	メモリ空間のアクセス条件 … 55
3-6	内蔵RAM領域のマップ … 61
3-7	I/Oマップ (64 Kバイト) … 64
4-1	多重処理制御を受ける割り込みの処理形態 … 69
4-2	同一割り込みの多重処理 … 71
4-3	割り込み受け付けの動作 … 73
4-4	レジスタ・バンク切り替えのシーケンス … 75
4-5	レジスタ・バンク復帰のシーケンス … 76
4-6	TSKSW命令によるレジスタ・バンク切り替えのシーケンス … 78
4-7	マクロ・サービスによる割り込み処理効率 … 80
4-8	ノーマル・モードの動作フロー … 81
4-9	シリアル・インタフェースの送信例 … 82
4-10	キャラクタ・サーチ・モードの動作フロー … 83
4-11	シリアル・インタフェースの受信例 … 84
4-12	マクロ・サービス制御レジスタ … 86
4-13	マクロ・サービス・チャンネルの構成 … 86
4-14	INT割り込みアクノリッジ・タイミング … 89
4-15	割り込み要求制御レジスタ … 91
4-16	IRQS … 92
4-17	ISPR … 94
4-18	ISPRの状態 … 95
4-19	INTM … 96
4-20	EXIC0, EXIC1, EXIC2 … 97
4-21	EMSO, EMS1, EMS2 … 98
5-1	メモリ・バンク構成 … 121
5-2	WTC … 123
5-3	READY端子によるウェイト (μ PD70325) … 125
5-4	READY端子によるウェイト (μ PD70335) … 133
5-5	RFLVビットによる制御回路 … 144
5-6	RFM … 145

図 の 目 次 (2/4)

図番号	タイトル, ページ
5-7	μPD42832接続回路例 … 146
5-8	μPD41256接続回路例 … 147
5-9	メモリ・リード・サイクル … 149
5-10	メモリ・ライト・サイクル … 149
5-11	I/Oリード・サイクル … 149
5-12	I/Oライト・サイクル … 149
5-13	メモリ・リード・サイクル (1 ウェイト・ステート挿入時) … 149
5-14	メモリ・ライト・サイクル (2 ウェイト・ステート挿入時) … 149
5-15	メモリ・ライト・サイクル (READY端子による操作時) … 150
5-16	リフレッシュ・サイクル (1 ウェイト・ステート挿入時) … 150
5-17	バス・ホールド受け付け解除タイミング … 150
5-18	ホールド・モード時のリフレッシュ・サイクル (ウェイト・ステート数0) … 150
5-19	HALT中のバス・ホールド受け付け解除タイミング … 151
5-20	メモリ・リード・サイクル … 152
5-21	メモリ・ライト・サイクル … 152
5-22	I/Oリード・サイクル … 152
5-23	I/Oライト・サイクル … 152
5-24	メモリ・リード・サイクル (1 ウェイト・ステート挿入時) … 153
5-25	メモリ・ライト・サイクル (2 ウェイト・ステート挿入時) … 153
5-26	メモリ・ライト・サイクル (READY端子による操作時) … 154
5-27	リフレッシュ・サイクル (1 ウェイト・ステート挿入時) … 154
5-28	バス・ホールド受け付け解除タイミング … 155
5-29	ホールド・モード時のリフレッシュ・サイクル … 155
5-30	HALT中のバス・ホールド受け付け解除タイミング … 156
6-1	DMAM0, DMAM1 … 166
6-2	DMAC0, DMAC1 … 167
6-3	DIC0, DIC1 … 168
6-4	SAR0 (SAR0H, SAR0M, SAR0L), SAR1 (SAR1H, SAR1M, SAR1L) … 170
6-5	DAR0 (DAR0H, DAR0M, DAR0L), DAR1 (DAR1H, DAR1M, DAR1L) … 172
6-6	TC0 (TC0H, TC0L), TC1 (TC1H, TC1L) … 173
6-7	バースト・モード時 … 174
6-8	シングルステップ・モード時 (μPD70325の場合) … 175
6-9	μPD70325での1転送モード (メモリ→I/O, ノー・ウェイト時) … 175
6-10	ダイヤモンド・リリース・モード (I/O→メモリ, I/O: 2 ウェイト, メモリ: ノー・ウェイト時) … 175

図 の 目 次 (3/4)

図番号	タイトル, ページ
6-11	バースト・モード時 (バイト転送) … 176
6-12	シングルステップ・モード時 (μ PD70335の場合) … 177
6-13	μ PD70335での1転送モード (メモリ \rightarrow I/O, ノー・ウエイト時) … 178
6-14	ダイヤモンド・リリース・モード (I/O \rightarrow メモリ, I/O: 2ウエイト, メモリ: ノー・ウエイト) … 179
6-15	$\overline{\text{IORD}}$ 信号, $\overline{\text{IOWR}}$ 信号作成回路例 … 180
7-1	ポート0-2構成 … 183
7-2	出力ポート指定のポート … 184
7-3	入力ポート指定のポート … 185
7-4	コントロール指定のポート … 185
7-5	PMC0 … 187
7-6	PM0 … 187
7-7	PMC1 … 188
7-8	PM1 … 189
7-9	PMC2 … 190
7-10	PM2 … 190
7-11	ポートTのブロック図 … 191
7-12	PMT … 192
8-1	クロック発生回路のブロック図 … 195
8-2	クロック発生回路の外付け回路 … 196
8-3	PRC … 198
9-1	インターバル・タイマ・モード時のタイマ・ユニットの構成 … 199
9-2	ワンショット・タイマ・モード時のタイマ・ユニットの構成 … 200
9-3	TOUT端子の出力状態 (ワンショット・タイマ・モード時) … 202
9-4	TMC0 … 205
9-5	TMC1 … 206
9-6	タイマ・ユニットからの割り込み要求 … 207
9-7	TMIC0, TMIC1, TMIC2 … 208
9-8	TMMS0, TMMS1, TMMS2 … 209
10-1	タイム・ベース・カウンタの構成 … 211
10-2	PRC … 212
10-3	TBIC … 212
11-1	シリアル・インタフェースの機能 … 214
11-2	送信データのフォーマット … 215
11-3	受信データのサンプル・タイミング … 217

図 の 目 次 (4/4)

図番号	タイトル, ページ
11-4	SCM0, SCM1 (アシンクロナス・モード設定時) … 222
11-5	SCM0 (I/Oインタフェース・モード設定時) … 223
11-6	SCC0, SCC1 … 226
11-7	SCS0, SCS1 … 228
11-8	SEIC _n , SRIC _n , STIC _n (n=0, 1) … 229
11-9	SRMS _n , STMS _n (n=0, 1) … 230
12-1	STBC … 231
12-2	割り込み要求によるHALTモードの解除 … 232
12-3	HALTモード中のマクロ・サービス/DMAの起動 … 233
12-4	NMI端子入力によるSTOPモードの解除 … 234
15-1	動作状態遷移図 … 250
A-1	正常動作時 … 252
A-2	異常動作時 … 253

表 の 目 次

表番号	タイトル, ページ
2-1	ポート0の動作 (n=0-7) … 20
2-2	ポート1の動作 (n=0-7) … 21
2-3	ポート2の動作 (n=0-7) … 24
3-1	汎用レジスタのオフセット … 43
3-2	ポインタとインデクス・レジスタのオフセット … 44
3-3	セグメント・レジスタのオフセット … 45
3-4	特殊機能レジスタ一覧 … 58
4-1	割り込み要因の一覧表 … 66
4-2	ソフトウェアによる割り込み … 99
5-1	バス制御のための端子機能 … 119
5-2	μPD70335のデータ・アクセス … 121
5-3	アドレス時分割出力 … 122
5-4	ウェイト・ステートの設定 … 124
5-5	リフレッシュの周期 … 143
5-6	リフレッシュ・サイクル中のウェイト・ステート … 143
5-7	$\overline{\text{REFRQ}}$ 信号の出力レベル … 144
6-1	各転送モードの機能 … 158
7-1	ポート0の動作 (n=0-7) … 186
7-2	ポート1の動作 (n=0-7) … 188
7-3	ポート2の動作 (n=0-7) … 189
8-1	推奨するセラミック発振子 … 197
8-2	推奨するクリスタル振動子 … 197
9-1	インターバル・タイマ・モード時のTMnのカウント時間 (n=0, 1) … 203
9-2	ワンショット・タイマ・モード時のTM0とMD0のカウント時間 … 204
11-1	ポーレート・ジェネレータの設定値 (参考) … 225
12-1	割り込み要求によるHALTモード解除後の動作 … 233
12-2	HALTモード / STOPモード … 235
13-1	各ハードウェアのリセット後の状態 … 237
A-1	CVTBD/CVTDB命令に関する制限事項対象命令 … 262

[× 毛]

第1章 概 説

μ PD70325, 70335 (別名称V25+, V35+) は, μ PD70108, 70116 (別名称V20, V30) とソフトウェアの互換性を持つシングルチップ・マイクロコンピュータ μ PD70320, 70330 (別名称V25, V35) の一部周辺機能を強化したマイクロコンピュータです。

μ PD70325, 70335 (別名称V25+, V35+) はともに16ビットCPUを内蔵し, 外部データ・バスがそれぞれ8ビット, 16ビットとなっています。

μ PD70325, 70335 (別名称V25+, V35+) は, μ PD70320, 70330 (別名称V25, V35) と同様に, 制御用マイコンとして重要となる強力な割り込み処理機能のほか, インターバル・タイマ, シリアル・インタフェース, 高速DMAコントローラなどの周辺機能を内蔵しています。

また, ソフトウェアの互換性を保ちながらも, メモリ・マッピングされたレジスタ・バンク, 内蔵ハードウェアなど使いやすさを追求した独自のアーキテクチャを数多く採用しています。

これらの特徴により, 多量のデータを処理し, 複数の機器を制御する必要があるシステムのメイン・コントローラとして最適です。特に, プリンタ, ハンディ・ワープロ, 各種端末などデータを大量に扱うシステム制御に適しています。

1.1 特 徴

- 内部16ビット・アーキテクチャ
- 外部データ・バス幅
 - ・ μ PD70325：8ビット
 - ・ μ PD70335：16ビット
- μ PD70108, 70116（ネイティブ・モード時）とソフトウェア・コンパチブル（追加命令あり）
- 3段パイプライン方式
- 最小インストラクション・サイクル
 - ・ μ PD70325-8, 70335-8：250 ns（外部クロック16 MHz）
 - ・ μ PD70325(A)-9, 70335(A)-9：220 ns（外部クロック18 MHz）
 - ・ μ PD70325-10, 70335-10：200 ns（外部クロック20 MHz）
- 内部メモリ RAM：256バイト
- メモリ空間 1 Mバイト
- I/O空間 64 Kバイト
- レジスタ・バンク（メモリ・マップト方式）：8バンク
- オン・チップ周辺ハードウェアのメモリ・マッピング（特殊機能レジスタ）
- コンパレータ付き入力ポート（ポートT）：8ビット
- I/Oライン 入力ポート：4ビット
出力ポート：20ビット
- シリアル・インタフェース：2チャンネル
 - ・専用ポー・レート・ジェネレータ内蔵
 - ・アシンクロナス・モード, I/Oインタフェース・モード
- 割り込みコントローラ
 - ・プログラマブル・プライオリティ（8レベル）
 - ・3種類の割り込み応答方式
 - ベクタ割り込み機能
 - レジスタ・バンク切り替え機能
 - マクロ・サービス機能
- DRAM, 疑似SRAMリフレッシュ機能
- DMAコントローラ：2チャンネル
 - ・4種類のDMA転送モード
 - ・転送レート
 - μ PD70325-8：最大4 Mバイト/秒^{注1}または最大2 Mバイト/秒^{注2}
 - μ PD70325-10：最大5 Mバイト/秒^{注1}または最大2.5 Mバイト/秒^{注2}
 - μ PD70335-8：最大5.3 Mバイト/秒^{注1}または最大2.7 Mバイト/秒^{注2}
 - μ PD70335-10：最大6.7 Mバイト/秒^{注1}または最大3.3 Mバイト/秒^{注2}

- 16ビット・タイマ：2チャンネル
- タイム・ベース・カウンタ（20ビット）：1チャンネル
- プログラマブル・ウェイト機能
- スタンバイ機能（STOP/HALT）
- CMOS
- パッケージ
 - ・84ピン・プラスチックQFJ（Quad Flat J-leaded Package）
 - ・94ピン・プラスチックQFP（Quad Flat Package）
- μ PD70325(A), 70335(A)は μ PD70325, 70335にくらべて動作温度範囲が広い。

- 注1. デイマンド・リリース・モード（DMARQ端子による停止制御をしない場合）時
2. デイマンド・リリース・モード（DMARQ端子による停止制御をする場合）時またはバースト・モード時



1.2 V25, V35との違い

		V25	V35	V25+	V35+
		μPD70320	μPD70330	μPD70325	μPD70335
D M A 機 能	転送処理方法	マイクロプログラムによる		専用ハードウェアによる	
	最大転送レート (8 MHz動作時)	0.6 Mバイト/秒	0.8 Mバイト/秒	4 Mバイト/秒	5.3 Mバイト/秒
	DMA要求のサンプリング・タイミング	命令実行サイクル間		バス・サイクル間	
	DMAサービス・チャンネル	内蔵RAM領域内		特殊機能レジスタ内	
	転送アドレスの指定方法	セグメント方式		リニア方式	
	シングルステップ・モードでの実行形式	1DMA転送 / 1命令実行		1DMA転送 / 1バス・サイクル	
	DMA転送中の割り込み要求 (ダイヤモンド・リリース・モード)	NMIのみ受け付ける		受け付けない	
	DMARQで停止制御するときの必要ウェイト数 (ダイヤモンド・リリース・モード)	必要なし		2 ウェイト	
	転送処理単位	バイト / ワード	バイト / ワード	バイト	バイト / ワード
	TC (ターミナル・カウンタ) の設定値	DMA転送回数		(DMA転送回数) - 1	
	ターミナル・カウンタの発生タイミング	TC = 0		TC = FFFFH	
	TC出力ロウ・レベル幅	固 定		ウェイト挿入により伸長	
シ リ ア ル ・ イ ン タ フ ェ ー ス	アシンクロナス・モード時の送信クロック出力 (チャンネル 0)	不 可		可 (SCK0端子)	
	シリアル・エラー・レジスタ	あ り		シリアル・ステータス・レジスタ	
	受信バッファ・フル・フラグ	な し		シリアル・ステータス・レジスタ内	
	送信バッファ・エンプティ・フラグ	な し		シリアル・ステータス・レジスタ内	
	オール・セント・フラグ	な し		シリアル・ステータス・レジスタ内	
割り込み機能	割り込み要因レジスタ	な し		あ り	
最大動作周波数		8 MHz		10 MHz	

1.3 オーダ情報と品質水準

(1) オーダ情報

オーダ名称	外部データ・バス (ビット)	パッケージ	最大動作周波数 (MHz)
μ PD70325L-8	8	84ピン・プラスチックQFJ	8
μ PD70325L-10	8	//	10
μ PD70325GJ-8-5BG	8	94ピン・プラスチックQFP	8
μ PD70325GJ-10-5BG	8	//	10
μ PD70335L-8	16	84ピン・プラスチックQFJ	8
μ PD70335L-10	16	//	10
μ PD70335GJ-8-5BG	16	94ピン・プラスチックQFP	8
μ PD70335GJ-10-5BG	16	//	10
μ PD70325GJ(A)-9-5BG	8	//	9
μ PD70335GJ(A)-9-5BG	16	//	9

QFJ : Quad Flat J-leaded Package

QFP : Quad Flat Package

(2) 品質水準

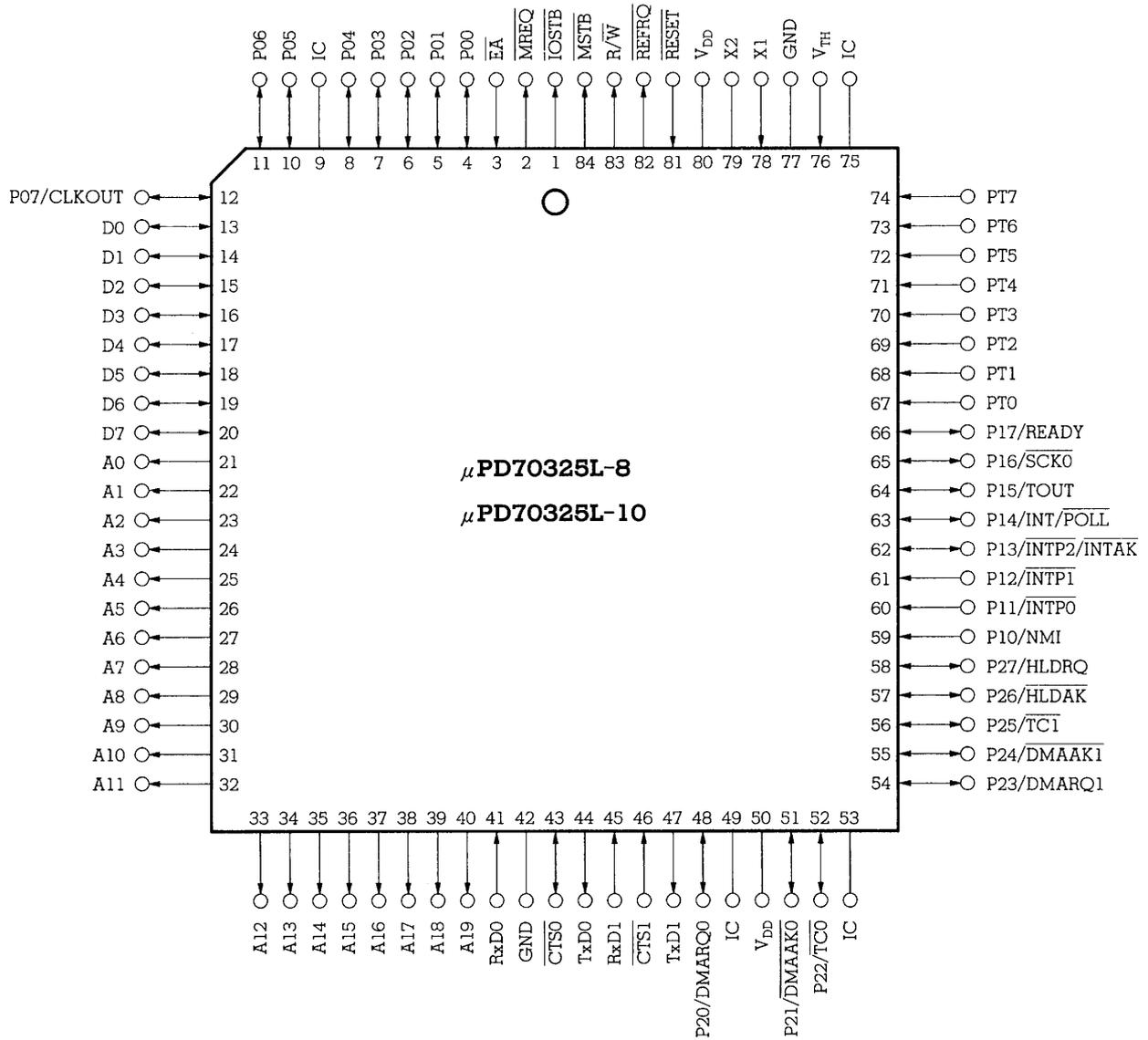
オーダ名称	パッケージ	品質水準
μ PD70325L-8	84ピン・プラスチックQFJ	標準 (一般電子機器用)
μ PD70325L-10	//	//
μ PD70325GJ-8-5BG	94ピン・プラスチックQFP	//
μ PD70325GJ-10-5BG	//	//
μ PD70335L-8	84ピン・プラスチックQFJ	//
μ PD70335L-10	//	//
μ PD70335GJ-8-5BG	94ピン・プラスチックQFP	//
μ PD70335GJ-10-5BG	//	//
μ PD70325GJ(A)-9-5BG	//	特別 (高信頼度電子機器用)
μ PD70335GJ(A)-9-5BG	//	//

品質水準とその応用分野の詳細については当社発行の資料「NEC 半導体デバイスの品質水準」(IEI-620)をご覧ください。

1.4 端子接続図 (Top View)

(1) 84ピン・プラスチックQFJ

(a) μ PD70325

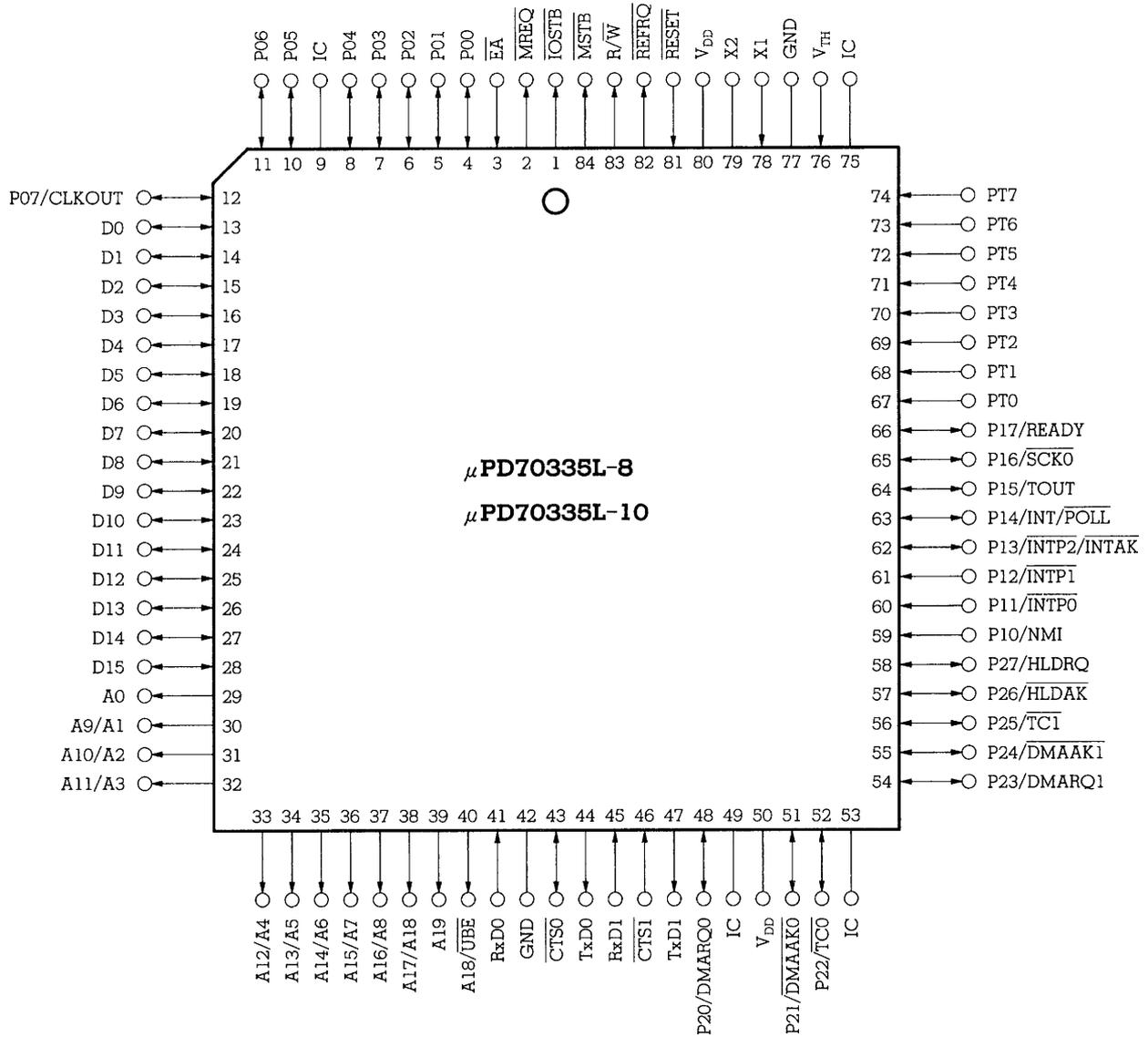


IC : Internally Connected

- ★ 注意1. IC端子は個別に抵抗 (3-10 kΩ) を介して, V_{DD} に接続してください。
- ★ 2. \overline{EA} 端子は抵抗 (3-10 kΩ) を介して, GNDに接続してください。



(b) μ PD70335

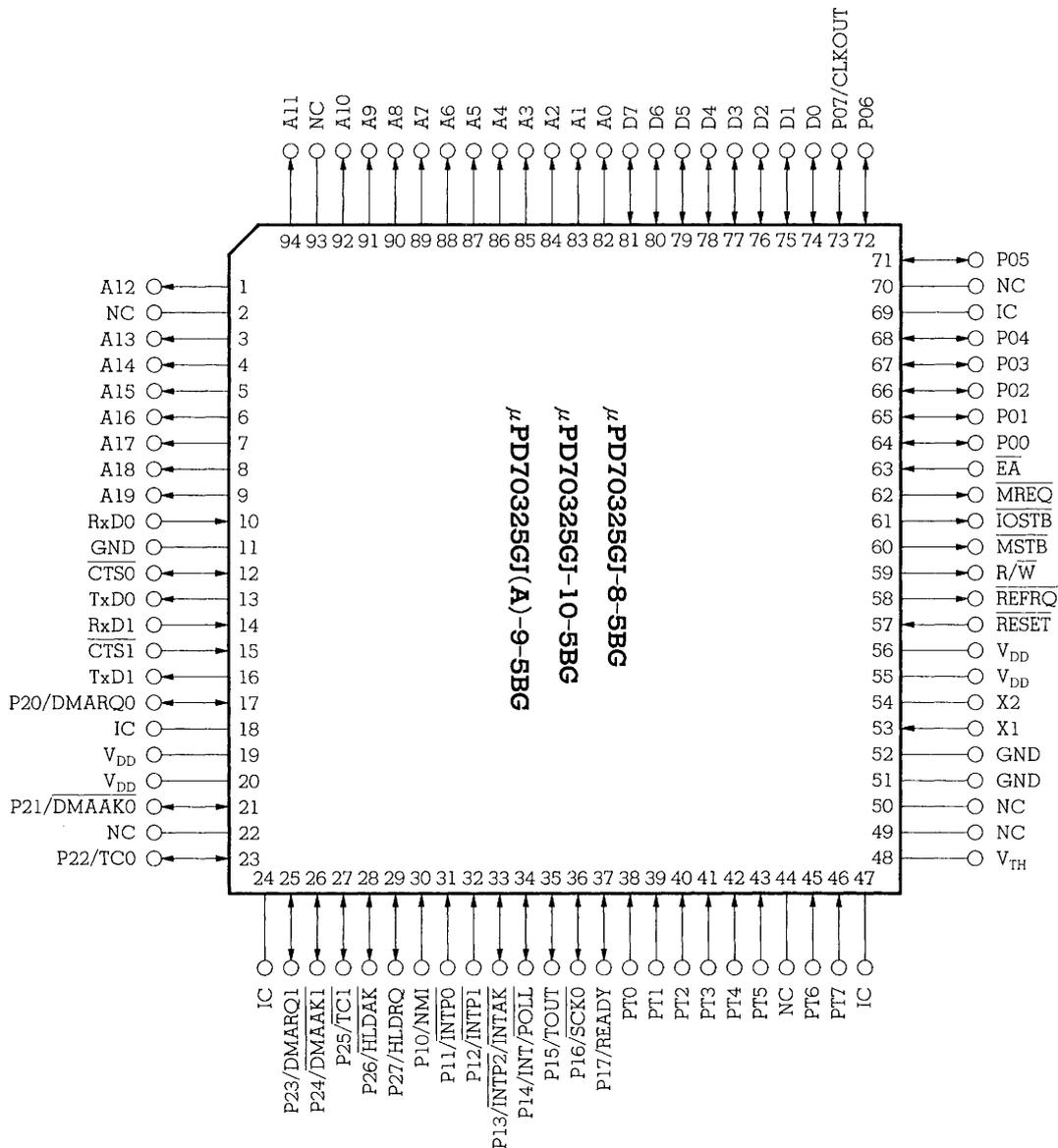


IC : Internally Connected

- 注意1. IC端子は個別に抵抗(3-10 k Ω)を介して, V_{DD} に接続してください。ただし, 9番ピンは抵抗(3-10 k Ω)を介して, GNDに接続してください。★
- 2. EA端子は抵抗(3-10 k Ω)を介して, GNDに接続してください。★

(2) 94ピン・プラスチックQFP

(a) μ PD70325

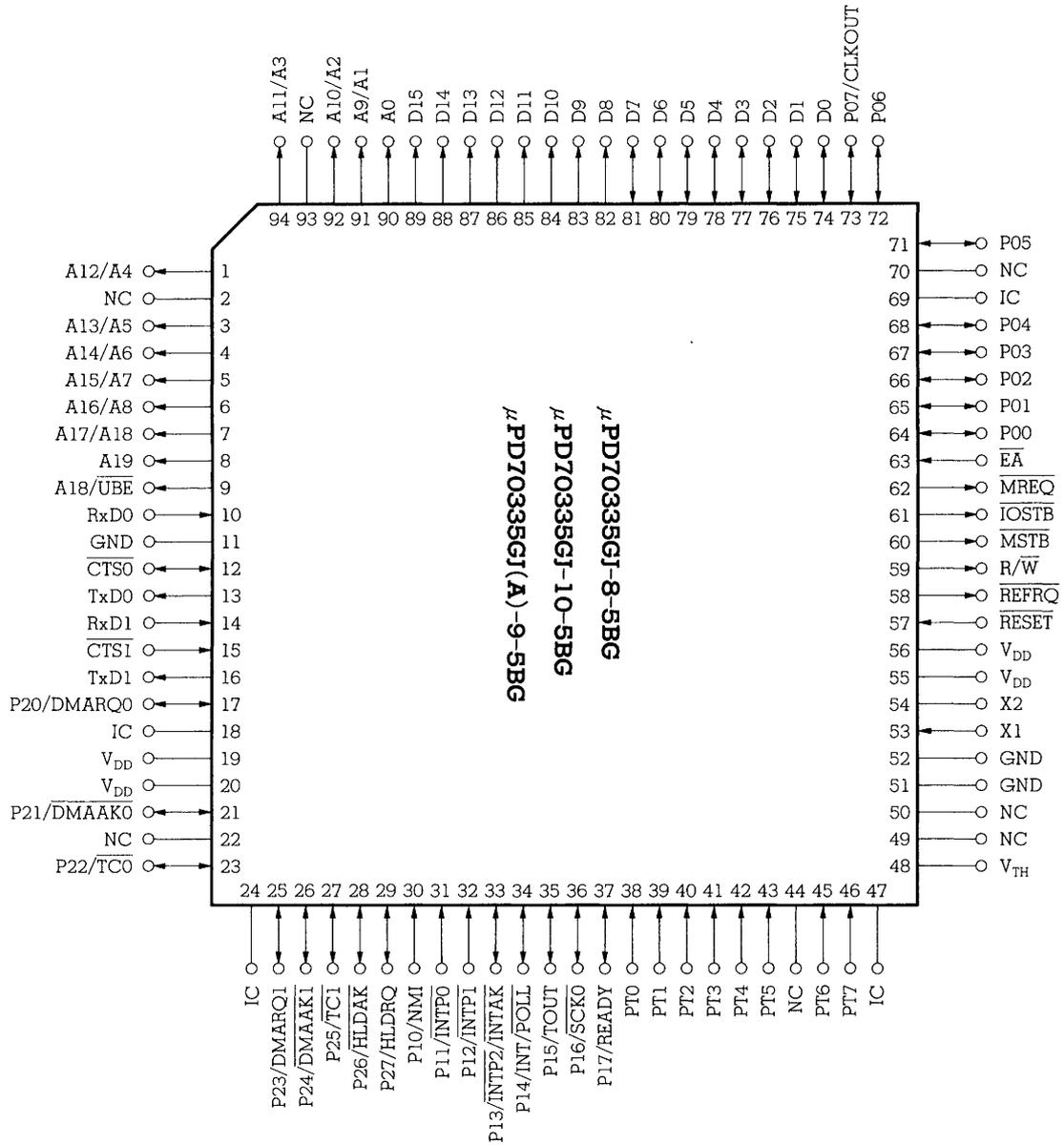


IC : Internally Connected
NC : Non-Connection

- ★ 注意1. IC端子は個別に抵抗 (3-10 k Ω) を介して, V_{DD}に接続してください。
- ★ 2. EA端子は抵抗 (3-10 k Ω) を介して, GNDに接続してください。



(b) μ PD70335



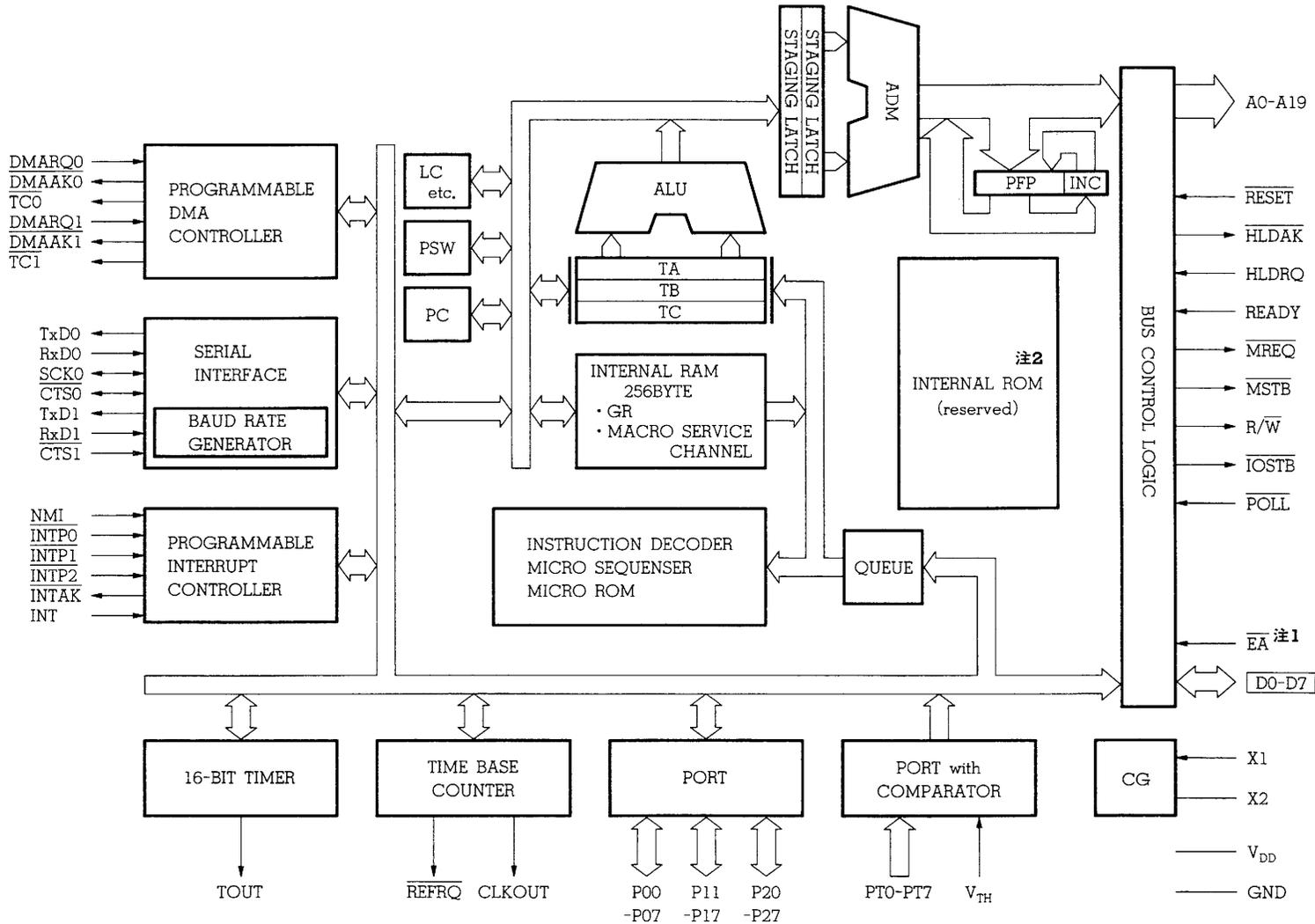
IC : Internally Connected
 NC : Non-Connection

- 注意1. IC端子は個別に抵抗 (3-10k Ω) を介して, V_{DD}に接続してください。★
- 2. EA端子は抵抗 (3-10k Ω) を介して, GNDに接続してください。★

$\overline{\text{IOSTB}}$: I/O Strobe	HLDRQ	: Hold Request
$\overline{\text{MREQ}}$: Memory Request	NMI	: Non-Maskable Interrupt Request
P00-P07	: Port 0	$\overline{\text{INTP0-INTP2}}$: Interrupt From Peripherals
P11-P17	: Port 1	$\overline{\text{INTAK}}$: Interrupt Acknowledge
P20-P27	: Port 2	INT	: Interrupt Request
CLKOUT	: Clock Out	$\overline{\text{POLL}}$: Polling
D0-D15	: Data Bus	TOUT	: Timer Output
A0-A19	: Address Bus	$\overline{\text{SCK0}}$: Serial Clock
RxD0, RxD1	: Receive Data	READY	: Ready
$\overline{\text{CTS0}}, \overline{\text{CTS1}}$: Clear To Send	PT0-PT7	: Port T
TxD0, TxD1	: Transfer Data	X1, X2	: Crystal
DMARQ0, DMARQ1	: DMA Request	$\overline{\text{RESET}}$: Reset
$\overline{\text{DMAAK0}}, \overline{\text{DMAAK1}}$: DMA Acknowledge	$\overline{\text{REFRQ}}$: Refresh Request
$\overline{\text{TC0}}, \overline{\text{TC1}}$: Terminal Count	$\text{R}/\overline{\text{W}}$: Read/Write
$\overline{\text{HLDAK}}$: Hold Acknowledge	$\overline{\text{MSTB}}$: Memory Strobe
		$\overline{\text{EA}}$: External Access

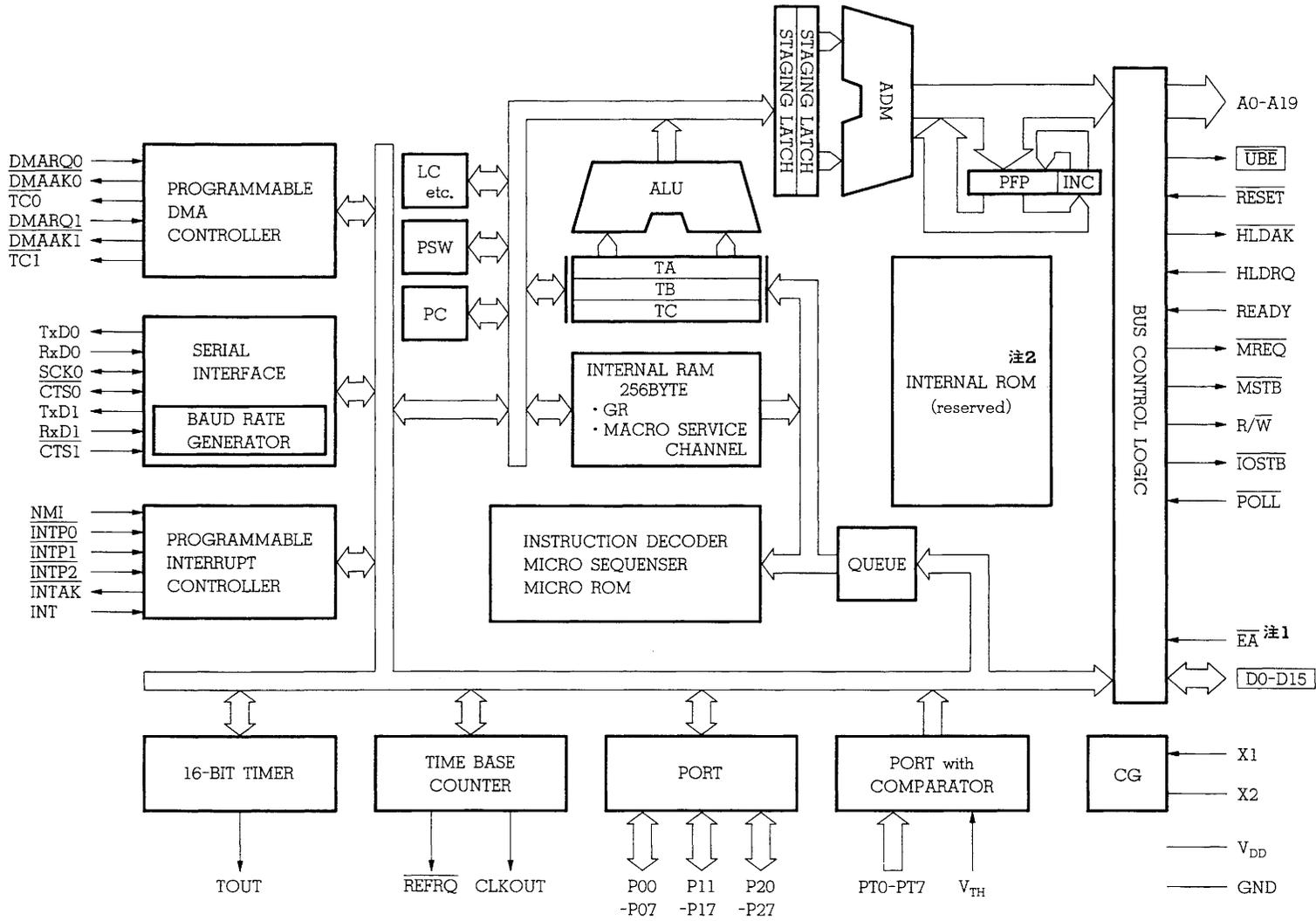
1.5 内部ブロック図

1.5.1 μ PD70325



- 注1. 外部でロウ・レベルに固定します。
 2. ユーザは、使用できません。

1.5.2 μ PD70335



- 注1. 外部でロウ・レベルに固定します。
- 2. ユーザは、使用できません。

第 2 章 端子機能

2.1 端子機能一覧

2.1.1 μ PD70325

(1) ポート

端子名称	入出力	兼用端子	機能
P00-P06	入出力	—	ビット単位で入出力の指定ができる 8 ビット入出力ポート
P07		CLKOUT	
P10	入力	NMI	汎用ポートとしては使用不可 (ノンマスクブル割り込み要求入力)
P11		$\overline{\text{INTP0}}$	
P12		$\overline{\text{INTP1}}$	
P13		$\overline{\text{INTP2/INTAK}}$	
P14	入出力	$\overline{\text{POLL/INT}}$	ビット単位で入出力の指定ができる入出力ポート
P15		TOUT	
P16		$\overline{\text{SCK0}}$	
P17		READY	
P20	入出力	$\overline{\text{DMARQ0}}$	ビット単位で入出力の指定ができる 8 ビット入出力ポート
P21		$\overline{\text{DMAAK0}}$	
P22		$\overline{\text{TC0}}$	
P23		$\overline{\text{DMARQ1}}$	
P24		$\overline{\text{DMAAK1}}$	
P25		$\overline{\text{TC1}}$	
P26		$\overline{\text{HLDK}}$	
P27		$\overline{\text{HLDRQ}}$	
PT0-PT7	入力	—	8 ビット・コンパレータ付き入力ポート

備考 リセット解除後、各ポート端子は入力ポートとなります。P13/ $\overline{\text{INTP2/INTAK}}$ を $\overline{\text{INTAK}}$ 端子として使用するときは、リセット解除後の外部割り込みコントローラの誤動作を回避するため必ずプルアップしてください。

(2) ポート以外

端子名称	入出力	機 能	兼用端子	
I $\overline{O}STB$	出 力	I/OリードまたはI/Oライトのストロープ出力	—	
M \overline{REQ}		メモリ・バス・サイクルが起動されたことを示す出力		
CLKOUT		システム・クロック出力		P07
D0-D7	入出力	8ビット・データ・バス	—	
A0-A19	出 力	20ビット・アドレス出力		
RxD0	入 力	シリアル・データ入力		
RxD1				
C $\overline{T}S0$	入出力	アシンクロナス・モード時：CTS入力		
		I/Oインタフェース・モード時：受信クロック入出力		
C $\overline{T}S1$	入 力	CTS入力		
TxD0	出 力	シリアル・データ出力		
TxD1				
DMARQ0	入 力	DMAリクエスト入力 (CH0)		P20
DMARQ1		DMAリクエスト入力 (CH1)	P23	
D $\overline{M}AAK0$	出 力	DMAアクノリッジ出力 (CH0)	P21	
D $\overline{M}AAK1$		DMAアクノリッジ出力 (CH1)	P24	
T $\overline{C}0$		DMA終了出力 (CH0)	P22	
T $\overline{C}1$		DMA終了出力 (CH1)	P25	
H $\overline{L}DAK$		HOLDアクノリッジ出力	P26	
H $\overline{L}DRQ$		入 力	HOLDリクエスト入力	P27
NMI			ノンマスカブル割り込み要求入力	P10
I $\overline{N}TP0$	外部割り込み要求入力		P11	
I $\overline{N}TP1$			P12	
I $\overline{N}TP2$			P13/I $\overline{N}TAK$	
I $\overline{N}TAK$	出 力	I $\overline{N}T$ のアクノリッジ信号出力	P13/I $\overline{N}TP2$	
POLL	入 力	POLL入力	P14/I $\overline{N}T$	
I $\overline{N}T$		外部割り込み要求入力	P14/POLL	
TOUT	出 力	タイマ出力	P15	
SCK $\overline{0}$		シリアル・クロック出力	P16	
READY	入 力	READY入力	P17	
X1	入 力	システム・クロック発振用のクリスタル振動子/セラミック発振子接続端子 (外部クロック入力は、X1端子とX2端子に逆相のクロックを入力可能)	—	
X2	—			
RESET	入 力	リセット信号入力		

端子名称	入出力	機 能	兼用端子
REFRQ	出 力	DRAMリフレッシュ・パルス出力	—
$\overline{R/W}$		リード・サイクル, ライト・サイクルの識別信号出力	
\overline{MSTB}		メモリ・リードまたはメモリ・ライトのストロブ出力	
V_{TH}	入 力	コンパレータの基準電圧入力	
V_{DD}	—	正電源供給端子	
GND		GND端子	
IC		インターナル・コネクタ(抵抗を介して, V_{DD} に接続)	
\overline{EA}		外部メモリ・アクセス(抵抗を介して, GNDに接続)	

2.1.2 μ PD70335

(1) ポート

端子名称	入出力	兼用端子	機能
P00-P06	入出力	—	ビット単位で入出力の指定ができる8ビット入出力ポート
P07		CLKOUT	
P10	入力	NMI	汎用ポートとしては使用不可（ノンマスカブル割り込み要求入力）
P11		$\overline{\text{INTP0}}$	
P12		$\overline{\text{INTP1}}$	
P13		$\overline{\text{INTP2/INTAK}}$	
P14	入出力	$\overline{\text{POLL/INT}}$	ビット単位で入出力の指定ができる入出力ポート
P15		TOUT	
P16		$\overline{\text{SCK0}}$	
P17		READY	
P20	入出力	DMARQ0	ビット単位で入出力の指定ができる8ビット入出力ポート
P21		$\overline{\text{DMAAK0}}$	
P22		$\overline{\text{TC0}}$	
P23		DMARQ1	
P24		$\overline{\text{DMAAK1}}$	
P25		$\overline{\text{TC1}}$	
P26		$\overline{\text{HLDAK}}$	
P27		HLDRQ	
PT0-PT7	入力	—	8ビット・コンパレータ付き入力ポート

備考 リセット解除後、各ポート端子は入力ポートとなります。P13/ $\overline{\text{INTP2/INTAK}}$ を $\overline{\text{INTAK}}$ 端子として使用するときは、リセット解除後の外部割り込みコントローラの誤動作を回避するため必ずプルアップしてください。

(2) ポート以外

端子名称	入出力	機能	兼用端子
$\overline{\text{IOSTB}}$	出力	I/OリードまたはI/Oライトのストロープ出力、および下位アドレスのストロープ出力	—
$\overline{\text{MREQ}}$		メモリ・バス・サイクルまたはI/Oバス・サイクルを起動したことを示す出力、および上位アドレスのストロープ出力	
CLKOUT		システム・クロック出力	P07
D0-D15	入出力	16ビット・データ・バス	—
A0	出力	アドレスのLSB出力で下位メモリ・バンクを選択	
$\text{A9/A1-A16/A8, A17/A18, A19}$		19ビット・アドレスを時分割で出力	
$\overline{\text{UBE}}$	出力	アドレスのビット18の出力と上位メモリ・バンクの選択に用いる。	A18
RxD0	入力	シリアル・データ入力	
RxD1			
$\overline{\text{CTS0}}$	入出力	アシンクロナス・モード時：CTS入力 I/Oインタフェース・モード時：受信クロック入出力	
$\overline{\text{CTS1}}$	入力	CTS入力	
TxD0	出力	シリアル・データ出力	
TxD1			
DMARQ0	入力	DMAリクエスト入力 (CH0)	P20
DMARQ1		DMAリクエスト入力 (CH1)	P23
$\overline{\text{DMAAK0}}$	出力	DMAアクノリッジ出力 (CH0)	P21
$\overline{\text{DMAAK1}}$		DMAアクノリッジ出力 (CH1)	P24
$\overline{\text{TC0}}$		DMA終了出力 (CH0)	P22
$\overline{\text{TC1}}$		DMA終了出力 (CH1)	P25
$\overline{\text{HLDK}}$		HOLDアクノリッジ出力	P26
HLDRQ	入力	HOLDリクエスト入力	P27
NMI		ノンマスクブル割り込み要求入力	P10
$\overline{\text{INTP0}}$		外部割り込み要求入力	P11
$\overline{\text{INTP1}}$			P12
$\overline{\text{INTP2}}$			P13/ $\overline{\text{INTAK}}$
$\overline{\text{INTAK}}$	出力	INTのアクノリッジ信号出力	P13/ $\overline{\text{INTP2}}$
$\overline{\text{POLL}}$	入力	$\overline{\text{POLL}}$ 入力	P14/ $\overline{\text{INT}}$
INT		外部割り込み要求入力	P14/ $\overline{\text{POLL}}$

端子名称	入出力	機能	兼用端子
TOUT	出力	タイマ出力	P15
$\overline{\text{SCK0}}$		シリアル・クロック出力	P16
READY	入力	READY入力	P17
X1	入力	システム・クロック発振用のクリスタル振動子／セラミック発振子接続端子（外部クロック入力は、X1端子とX2端子に逆相のクロックを入力可能）	—
X2			
$\overline{\text{RESET}}$	入力	リセット信号入力	
$\overline{\text{REFRQ}}$	出力	DRAMリフレッシュ・パルス出力	
R/ $\overline{\text{W}}$		リード・サイクル、ライト・サイクルの識別信号出力	
$\overline{\text{MSTB}}$		メモリ・リードまたはメモリ・ライトのストロープ出力、および下位アドレスのストロープ出力	
V_{TH}	入力	コンパレータの基準電圧入力	
V_{DD}	—	正電源供給端子	
GND		GND端子	
IC		インターナル・コネクタ（抵抗を介して、 V_{DD} に接続）	
$\overline{\text{EA}}$		外部メモリ・アクセス（抵抗を介して、GNDに接続）	

2.1.3 各モードにおける端子状態

端 子		動作状態	ホールド	HALT	STOP	RESET 期 間 中	その他
ポート	ポート・モードの出力時	保 持	保 持	保 持	保 持	Hi-Z	継 続
	コントロール・モードの出力時	継 続	継 続	保 持	保 持	Hi-Z	継 続
TxDO, TxD1		継 続	継 続	保 持	保 持	Hi-Z	継 続
$\overline{\text{CTS0}}$		継 続	継 続	保 持	保 持	Hi-Z	継 続
DO-D15		Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	継 続
AO-A19, $\overline{\text{UBE}}$		Hi-Z	保持 ^注	保持 ^注	保持 ^注	Hi-Z	継 続
$\overline{\text{MREQ}}$		Hi-Z	ハイ・レベル	ハイ・レベル	ハイ・レベル	Hi-Z	継 続
$\overline{\text{MSTB}}$		Hi-Z	ハイ・レベル	ハイ・レベル	ハイ・レベル	Hi-Z	継 続
R/ $\overline{\text{W}}$		Hi-Z	ハイ・レベル	ハイ・レベル	ハイ・レベル	Hi-Z	継 続
$\overline{\text{REFRQ}}$		Hi-Z	継 続	保 持	保 持	Hi-Z	継 続
$\overline{\text{IOSTB}}$		Hi-Z	ハイ・レベル	ハイ・レベル	ハイ・レベル	Hi-Z	継 続

継 続：指定された動作を継続

保 持：そのモードに移る直前の状態を保持

Hi-Z ：ハイ・インピーダンス

注 アドレス・データが保持されますが、出力されるアドレスは不定です。

2.2 各端子の機能説明

μPD70325, 70335には専用の制御端子とポートが兼用になっている端子があります。ポートと兼用の端子の設定方法、ハードウェア構成については第7章 ポート機能を参照してください。

2.2.1 P00-P07 (Port0) …………… 3 ステート入出力

ポート0 (出力ラッチ付き8ビット入出力ポート) の8ビット入出力端子です。P07端子はCLKOUT出力としても働きます。

各端子の動作モードについて、P00-P06端子はポート・モード専用で、P07端子はポート0モード・コントロール・レジスタ (PMCO) によってポート/コントロール・モードの選択ができます (表2-1 参照)。

表2-1 ポート0の動作 (n=0-7)

	PMCO _n =1	PMCO _n =0	
		PM0 _n =1	PM0 _n =0
P00	X	入力ポート	出力ポート
P01		入力ポート	出力ポート
P02		入力ポート	出力ポート
P03		入力ポート	出力ポート
P04		入力ポート	出力ポート
P05		入力ポート	出力ポート
P06		入力ポート	出力ポート
P07	CLKOUT出力	入力ポート	出力ポート

(1) ポート・モード

P00-P07端子は、ポート0モード・レジスタ (PM0) により、ビット単位に入力または出力ポートの設定ができます。ただし、P07端子はPMCOレジスタによりポート・モードに設定しておきます (PMCOレジスタのビット7=0)。

(2) コントロール・モード

P07端子は、PMCOレジスタによりコントロール・モードに設定した場合 (PMCOレジスタのビット7=1)、CLKOUT出力として働きます。

(a) CLKOUT (Clock Out) …………… P07兼用出力

クロック発生回路で設定された周波数で、CPUおよび周辺ハードウェアに各種クロックを供給するシステム・クロック (CLK) の出力端子です。



出力ポートに設定した端子は、ホールド・モード、スタンバイ・モードでは直前のデータを保持します。P07端子をCLKOUTに設定したときは、ホールド・モード、HALTモードではクロック出力を継続し、STOPモードでは直前のデータを保持します。

$\overline{\text{RESET}}$ 入力時、P00-P07端子は入力ポートになります（出力ハイ・インピーダンス）。

2.2.2 P10-P17 (Port1) 3 ステート入出力

ポート 1 (出力ラッチ付き 8 ビット入出力ポート) および、各種コントロール信号としても働く 8 ビット入出力の兼用端子です。

P10-P17端子の動作モードは、ポート 1 モード・コントロール・レジスタ (PMCl) により、ビット単位にポート/コントロール・モードの設定ができます (表 2-2 参照)。ただし、P11-P12端子はポート・モード専用で、割り込み要求入力としても働きます。また、P10はNMI専用入力です。

表 2-2 ポート 1 の動作 (n=0-7)

	PMCl _n =1	PMCl _n =0	
		PM1 _n =1	PM1 _n =0
P10	X	NMI入力	X
P11		$\overline{\text{INTPO}}$ 入力	
P12		$\overline{\text{INTPI}}$ 入力	
P13		$\overline{\text{INTAK}}$ 出力	
P14	INT入力	入力ポート ($\overline{\text{POLL}}$ 入力)	出力ポート
P15	TOUT出力	入力ポート	出力ポート
P16	$\overline{\text{SCKO}}$ 出力	入力ポート	出力ポート
P17	READY入力	入力ポート	出力ポート

(1) ポート・モード

P10-P17端子は、PMClレジスタによりポート・モードに設定した場合 (PMCl_n=0; n=0-7), ポート 1 モード・レジスタ (PM1) により、ビット単位に入力または出力ポートに設定できます。

P10-P13端子は入力ポート専用で、割り込み要求入力としても働きます。また、P14端子は入力ポートに設定時、 $\overline{\text{POLL}}$ 入力としても働きます。

(a) NMI (Non-Maskable Interrupt)P10入力

ソフトウェアによってマスク不可能なノンマスクابل割り込み要求入力端子です。

ノンマスクابل割り込みは、CPUがいつでも受け付ける割り込みのため、どの割り込みに対しても優先します。

NMI入力はエッジで検出され、有効エッジは外部割り込みモード・レジスタ (INTM) で指定

します。この入力はいくつクロック・サイクルでサンプリングされ、イン・アクティブ・レベルからアクティブ・レベルに変化した後、一定のアクティブ・レベルを入力することでNMIが受け付けられます。NMIが受け付けられると、実行中の命令終了後にベクタ番号2の割り込みが発生します。

このNMI入力はCPUのスタンバイ・モードの解除にも使用されます。

なお、NMI入力の状態は、P10を読み出すことによりモニタできます。

(b) $\overline{\text{INTPO}}\text{-}\overline{\text{INTP2}}$ (Interrupt from Peripheral0-2)……………P11-P13兼用入力

ソフトウェアによってマスク可能な外部割り込み要求入力端子です。

$\overline{\text{INTPn}}$ (n=0-2)入力はエッジで検出され、有効エッジは外部割り込みモード・レジスタ (INTM) で指定します。この入力はいくつクロック・サイクルでサンプリングされ、イン・アクティブ・レベルからアクティブ・レベルに変化したあと、一定以上のアクティブ・レベルを入力することで $\overline{\text{INTPn}}$ (n=0-2) が受け付けられます。

この $\overline{\text{INTPn}}$ 入力はHALTモードの解除にも使用されます。

(c) $\overline{\text{POLL}}$ (Poll) ……………P14/INT兼用入力

$\overline{\text{POLL}}$ 入力はPOLL命令によってチェックされ、ロウ・レベルであれば次の命令に移ります。ハイ・レベルであればロウ・レベルになるまで続く5クロック・サイクルごとに $\overline{\text{POLL}}$ 入力をチェックします。

これらの機能はCPUのプログラムと外部デバイス動作との同期をとるために利用されます。

注意 $\overline{\text{POLL}}$ 端子はP14が入力モード(入力ポート、INT入力)と指定されたときに機能します。

その他の場合は、POLL命令実行時にロウ・レベルとみなされます。

(2) コントロール・モード

P13-P17端子は、PMC1レジスタにより、ビット単位にコントロール端子として設定することができます。

(a) $\overline{\text{INTAK}}$ (Interrupt Acknowledge) ……………P13/ $\overline{\text{INTP2}}$ 兼用出力

ソフトウェアによってマスク可能な割り込み要求入力 (INT) のアックノリッジ信号出力端子です。

$\overline{\text{INTAK}}$ 信号はCPUがINT信号を受け付けたときにロウ・レベルを出力し、外部デバイスはこの信号に同期させてデータ・バス (D0-D15) を介して割り込みベクタをCPUに入力します。

INT端子とともに、 μ PD71059または相当品の割り込みコントローラを接続するのに使用します。

(b) INT (Interrupt) ……P14/POLL兼用入力

ソフトウェアによってマスク可能な割り込み要求入力端子です。

この入力はハイ・レベル・アクティブで、命令の最終クロック・サイクルで検出されます。外部デバイスはCPUから出力される $\overline{\text{INTAK}}$ 信号によって、INT割り込み要求が受け付けられたことを確認します。INT信号は、少なくとも最初の $\overline{\text{INTAK}}$ 信号が出力されるまでハイ・レベルを保持する必要があります。

INT端子は、 $\overline{\text{INTAK}}$ 端子とともに $\mu\text{PD71059}$ または相当品の割り込みコントローラを接続するのに使用します。

このINT入力はHALTモードの解除にも使用します。

(c) TOUT (Timer Output) ……P15兼用出力

タイマ・ユニット (タイマ0) からの出力端子です。

(d) $\overline{\text{SCKO}}$ (Serial Clock0) ……P16兼用出力

シリアル・インタフェース (チャンネル0) の送信クロック出力端子です。

(e) READY (Ready) ……P17兼用出力

外部からバス・サイクル時 (メモリ・リフレッシュ・サイクルは除く) にウエイト・ステートの挿入を制御する入力端子です。

READY端子は、レベルでセンスされ、ロウ・レベルの期間中ウエイト・ステートが挿入されます。

注意 P17を選択すると、バス・サイクルは常にレディとなります。

出力ポートに設定した端子は、ホールド・モード、スタンバイ・モードでは直前のデータを保持します。コントロール・モードに設定したコントロール信号出力端子は、ホールド・モード、HALTモードでは動作を継続し、STOPモードでは直前のデータを保持します。

$\overline{\text{RESET}}$ 入力時、P00-P17端子は入力ポートになります (出力ハイ・インピーダンス)。

2.2.3 P20-P27 (Port2) ……3 ステート入出力

ポート2 (出カラッチ付き8ビット入出力ポート) および、各種コントロール信号としても働く8ビット入出力の兼用端子です。

P20-P27端子の動作モードは、ポート2モード・コントロール・レジスタ (PMC2) により、ビット単位にポート/コントロール・モードの設定ができます (表2-3参照)。

表 2-3 ポート 2 の動作 (n=0-7)

	PMC2n=1	PMC2n=0	
		PM2n=1	PM2n=0
P20	DMARQ0入力	入力ポート	出力ポート
P21	DMAAK0出力	入力ポート	出力ポート
P22	TC0出力	入力ポート	出力ポート
P23	DMARQ1入力	入力ポート	出力ポート
P24	DMAAK1出力	入力ポート	出力ポート
P25	TC1出力	入力ポート	出力ポート
P26	HLDAK出力	入力ポート	出力ポート
P27	HLDRQ入力	入力ポート	出力ポート

(1) ポート・モード

P20-P27端子は、PMC2レジスタによりポート・モードに設定した場合 (PMC2n=0; n=0-7), ポート 2 モード・レジスタ (PM2) により、ビット単位に入力または出力ポートに設定できません。

(2) コントロール・モード

P20-P27端子は、PMC2レジスタにより、ビット単位にコントロール端子に設定 (PMC2n=1; n=0-7) することができます。

(a) DMARQ0, DMARQ1 (DMA Request0, 1) ……………P20, P23兼用入力

DMAコントローラ (チャンネル0, 1) のDMA要求入力端子です。これらはアクティブ・ハイ信号です。

(b) DMAAK0, DMAAK1 (DMA Acknowledge0, 1) ……………P21, P24兼用出力

DMAコントローラ (チャンネル0, 1) のDMAアクノリッジ出力端子です。ただし、メモリ-メモリ間のDMA転送 (バースト・モード, シングルステップ・モード) 時には出力されません。これらはアクティブ・ロウ信号です。

(c) TC0, TC1 (Terminal Count0, 1) ……………P22, P25兼用出力

DMAコントローラ (チャンネル0, 1) のDMA完了信号出力端子です。
DMAサービス・チャンネル中のTC0, TC1がそれぞれ0になったときに出力されます。これらはアクティブ・ロウ信号です。

(d) $\overline{\text{HLDAK}}$ (Hold Acknowledge) ……P26兼用出力

$\mu\text{PD70325}$, 70335 がホールド要求信号 ($\overline{\text{HLDRQ}}$) を受け付けて、バスをハイ・インピーダンス状態にしたことを示すアクノリッジ信号出力端子です。

この信号がアクティブ (ロウ・レベル) の間、アドレス・バス、データ・バス、制御バスは、ハイ・インピーダンスになります。

(e) $\overline{\text{HLDRQ}}$ (Hold Request) ……P27兼用入力

外部デバイスが $\mu\text{PD70325}$, 70335 に対してアドレス・バス、データ・バス、制御バスの解放を要求するための信号入力端子です。

$\overline{\text{HLDRQ}}$ 入力は、アクティブ・ハイ信号です。

出力ポートに設定した端子は、ホールド・モード、スタンバイ・モードでは直前のデータを保持します。コントロール・モードに設定したコントロール信号出力端子は、ホールド・モード、HALTモードでは動作を継続し、STOPモードでは直前のデータを保持します。

$\overline{\text{RESET}}$ 入力により、入力ポートになります (出力ハイ・インピーダンス)。

2.2.4 PTO-PT7 (Port with Comparator0-7) ……入力

ポートT (PT0-PT7) は、スレッシュホールド電圧 (基準電圧) を16段階に変えられるコンパレータ付きの8ビット入力端子です。

2.2.5 V_{TH} (Threshold Voltage) ……入力

ポートTの基準電圧入力端子です。

2.2.6 TxD0 , TxD1 (Transmit Data0, 1) ……出力

シリアル・インタフェース (チャンネル0, 1) のシリアル・データ出力端子です。

アシンクロナス・モード時、送信データはスタート・ビット、キャラクタ・ビット、パリティ・ビット、ストップ・ビットの1データ・フレームで、最下位 (LSB) から送信されます。 TxD0 , TxD1 端子は送信禁止のとき、あるいはシリアル・レジスタに送信するデータを持たないとき、マーク状態(1)になります。

I/Oインタフェース・モード時 (TxD0 端子のみ) は、送信データは8ビット固定で最上位ビット (MSB) から送信されます。

TxD0 , TxD1 端子は、ホールド・モード/HALTモードでは動作を継続しますが、STOPモードでは直前のデータを保持します。また、 $\overline{\text{RESET}}$ 入力時はハイ・インピーダンスになります。

2.2.7 RxDO , RxD1 (Receive Data0, 1) ……入力

シリアル・インタフェース (チャンネル0, 1) のシリアル・データ入力端子です。

アシンクロナス・モード時、受信許可状態で RxDO , RxD1 端子入力のロウ・レベルを検出するとスター

ト・ビットとして認識して受信動作を行います。

I/Oインタフェース・モード時 (RxDO端子のみ) は、受信データは受信クロックの立ち上がりエッジでシリアル・レジスタに入力されます。

2.2.8 $\overline{\text{CTS0}}$ (Clear to Send0) ……入出力

シリアル・インタフェース (チャンネル0) のCTS端子です。

$\overline{\text{CTS0}}$ 端子は、アシンクロナス・モード時はロウ・レベル・アクティブで、送信許可状態にする入力端子です。I/Oインタフェース・モード時は受信クロックの入出力端子となります。

$\overline{\text{CTS0}}$ 端子は、ホールド・モード/HALTモードでは動作を継続しますが、STOPモードでは直前のデータを保持します。また、 $\overline{\text{RESET}}$ 入力時、I/Oインタフェース・モードの入力端子となります (出力ハイ・インピーダンス)。

2.2.9 $\overline{\text{CTS1}}$ (Clear to Send1) ……入力

シリアル・インタフェース (チャンネル1) のCTS端子です。

$\overline{\text{CTS1}}$ 端子は、ロウ・レベル・アクティブで送信許可状態にする入力端子です (アシンクロナス・モード)。

2.2.10 $\overline{\text{RESET}}$ (Reset) ……入力

ロウ・レベル・アクティブのリセット入力端子です。

$\overline{\text{RESET}}$ 入力は非同期入力であり、動作クロックに無関係に一定のロウ・レベル幅を有する信号が入力されると、すべての動作に優先してシステム・リセットがかかります。

通常のイニシャライズ/スタートのほかに、スタンバイ (STOP/HALT) モードの解除にも使用されます。

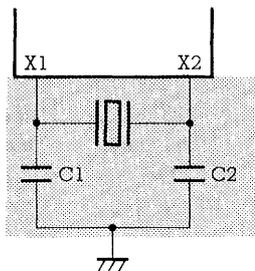
2.2.11 $\overline{\text{EA}}$ (External Access) ……入力

各部メモリをアクセスするための入力端子です。抵抗 (3-10 k Ω) を介して、GNDに接続してください。

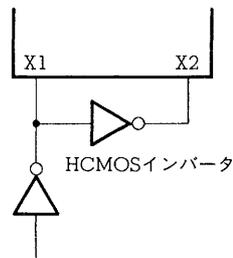
2.2.12 X1, X2 (Crystal)

システム・クロック発生用クリスタル振動子／セラミック発振子接続端子です。外部クロックを入力することも可能です。

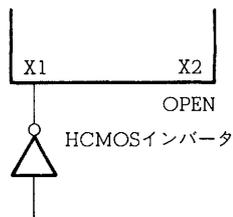
(a) クリスタル／セラミック^注発振



(b) 外部クロック入力



(c) 外部クロック入力 (X2: オープン)



- 注意1. 発振回路はX1, X2端子にできるかぎり近づけてください。
 2. の範囲に他の信号線を通さないでください。

注 セラミック発振子および外付け容量は、デバイスとのマッチングを考慮して、当社推奨の製品を使用してください。

2.2.13 DO-D15 (Data Bus0-15) ……………入出力

データ・バス入出力端子です。

ホールド・モード／スタンバイ・モード／ $\overline{\text{RESET}}$ 入力時、すべてハイ・インピーダンスになります。

2.2.14 AO-A19 (Address Bus0-19) ……………出力

20ビット・アドレス・バス出力端子です。

スタンバイ・モードではアドレス・データは保有されますが、出力されるアドレスは不定です。ホールド・モード／ $\overline{\text{RESET}}$ 入力時ではハイ・インピーダンスになります。

2.2.15 $\overline{\text{MREQ}}$ (Memory Request) ……出力

メモリ・バス・サイクルが起動され、アドレス・バス上のメモリ・アドレスが有効であることを示すロウ・レベル・アクティブの出力端子です。

スタンバイ・モードでは、ハイ・レベル固定となります。ホールド・モード/ $\overline{\text{RESET}}$ 入力時は、ハイ・インピーダンスとなります。

2.2.16 $\overline{\text{MSTB}}$ (Memory Strobe) ……出力

$\overline{\text{MSTB}}$ 端子は、 $\overline{\text{MREQ}}$ 、 $\text{R}/\overline{\text{W}}$ 信号と組み合わせてメモリ・アクセスの制御を行う出力端子です。

メモリ・ライト時、データ・バスに出力しているデータが有効であることを示します。 $\overline{\text{MSTB}}$ 出力は、 $\overline{\text{MREQ}}$ とは立ち下がりタイミングだけが異なる信号です。

スタンバイ・モードでは、ハイ・レベル固定となります。ホールド・モード/ $\overline{\text{RESET}}$ 入力時は、ハイ・インピーダンスとなります。

2.2.17 $\text{R}/\overline{\text{W}}$ (Read/Write Strobe) ……出力

メモリ・バス・サイクル起動時、メモリ・リード・サイクル、またはメモリ・ライト・サイクルの識別用信号出力端子です。

$\text{R}/\overline{\text{W}}$ 出力のハイ・レベルがメモリ・リード・サイクルを示し、ロウ・レベルがメモリ・ライト・サイクルを示します。

スタンバイ・モードでは、ハイ・レベル固定となります。ホールド・モード/ $\overline{\text{RESET}}$ 入力時は、ハイ・インピーダンスとなります。

2.2.18 $\overline{\text{REFRQ}}$ (Refresh Request) ……出力

リフレッシュ・パルスの出力端子です。

$\overline{\text{REFRQ}}$ 端子の出力は、リフレッシュ・モード・レジスタ (RFM) 内容により制御されます。HALTモードでは動作を継続しますが、STOPモードでは直前のデータを保持します。また、ホールド・モード/ $\overline{\text{RESET}}$ 入力時はハイ・インピーダンスとなります。

2.2.19 $\overline{\text{IOSTB}}$ (I/O Strobe) ……出力

I/Oバス・サイクルが起動されたことを示すロウ・レベル・アクティブの出力端子です。

$\overline{\text{IOSTB}}$ 端子は、ロウ・レベル時にA0-A15に出力されているI/Oアドレスが有効であることを示します。

スタンバイ・モードでは、ハイ・レベル固定となります。ホールド・モード/ $\overline{\text{RESET}}$ 入力時は、ハイ・インピーダンスとなります。

2.2.20 V_{DD} (Power Supply)

正電源供給端子です。

全 V_{DD} 端子に、 V_{DD} 電圧を供給してください。

2.2.21 GND (Ground)

GND電位の端子です。

全GND端子を，GND電位に固定してください。

2.2.22 $\overline{\text{UBE}}$ (Upper Byte Enable) ……………入力

μ PD70335で上位のメモリ・バンクを選択する調子です。A18と兼用しています。

この信号がアクティブとなるのは，次のバス・サイクルです。

- ・奇数アドレスに対するバイト・アクセス，または奇数アドレスに対するワード・データのための1回目のバイト・アクセス
- ・偶数アドレスに対するワード・データのアクセス

ア ク セ ス		$\overline{\text{UBE}}$	A0
偶数アドレスのワード		0	0
奇数アドレスのワード	1回目	0	1
	2回目	1	0
偶数アドレスのバイト		1	0
奇数アドレスのバイト		0	1

上述の動作は，I/Oサイクルにおいても有効です。

2.3 端子の入出力回路

2.3.1に入出力回路タイプ、2.3.2にタイプごとの回路を示します。

2.3.1 入出力回路タイプ

(1) ポート

端子名	タイプ	端子名	タイプ
P00-P06	5	P20/DMARQ0	5
P07/CLKOUT	5	P21/DMAAK0	5
P10/NMI	2	P22/TC0	5
P11/INTP0	1	P23/DMARQ1	5
P12/INTP1	1	P24/DMAAK1	5
P13/INTP2/INTAK	5	P25/TC1	5
P14/INT/POLL	5	P26/HLDAK	5
P15/TOUT	5	P27/HLDRQ	5
P16/SCK0	5	PT0-PT7	7
P17/READY	5		

(2) ポート以外

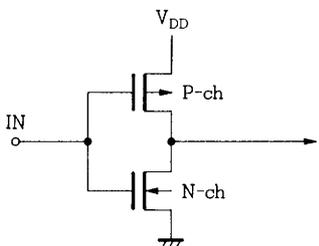
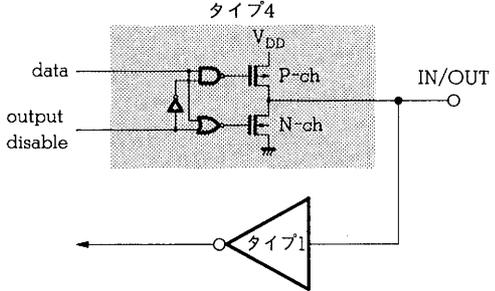
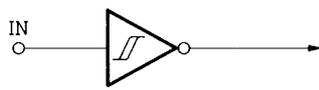
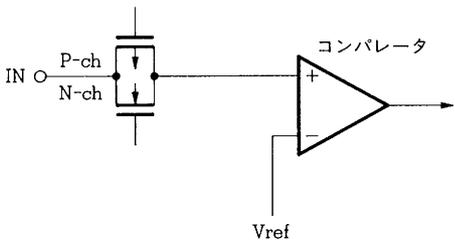
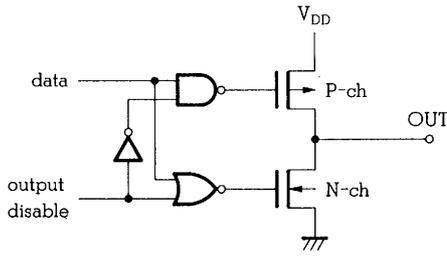
(a) μ PD70325

端子名	タイプ	端子名	タイプ
TxD0	4	EA	1
TxD1	4	D0-D7	5
RxD0	1	A0-A19	4
RxD1	1	MREQ	4
CTS0	5	MSTB	4
CTS1	1	R/W	4
REFRQ	4	IOSTB	4
RESET	2		

(b) μ PD70335

端子名	タイプ	端子名	タイプ
TxD0	4	A0	4
TxD1	4	A9/A1-A16/A8	4
RxD0	1	A17/A18	4
RxD1	1	A19	4
$\overline{\text{CTS0}}$	5	A18/ $\overline{\text{UBE}}$	4
$\overline{\text{CTS1}}$	1	$\overline{\text{MREQ}}$	4
$\overline{\text{REFRQ}}$	4	$\overline{\text{MSTB}}$	4
$\overline{\text{RESET}}$	2	R/ $\overline{\text{W}}$	4
$\overline{\text{EA}}$	1	$\overline{\text{IOSTB}}$	4
DO-D15	5		

2.3.2 入出力回路

<p>タイプ 1</p> 	<p>タイプ 5</p>  <p>タイプ4のプッシュプル出力とタイプ1の入力バッファで構成される入出力回路となっています。</p>
<p>タイプ 2</p>  <p>ヒステリシス特性^注を有するシュミット・トリガ入力となっています。</p>	<p>タイプ 7</p>  <p>(スレッシュホールド電圧)</p>
<p>タイプ 4</p>  <p>出力ハイ・インピーダンス (P-ch, N-chともにオフ) とすることができるプッシュプル出力となっています。</p>	

★ 注 軽いヒステリシス特性を持つように設計しています。ただし、この端子の入力に対する特性を満足できない場合は、必ずヒステリシス特性を持つデバイスを外部に接続してください。

2.4 未使用端子の処理

抵抗を介して V_{DD} またはGNDに接続する場合、3-10 k Ω の抵抗を使用することを推奨します。次に未使用端子の推奨接続方法を示します。★

(1) ポート

端子名	推奨接続方法	
P00-P06	入力状態：個別に抵抗を介して V_{DD} に接続	
P07/CLKOUT	出力状態：オープン	
P10/NMI	抵抗を介してGNDに接続	
P11/ $\overline{\text{INTP0}}$	抵抗を介して V_{DD} またはGNDに接続	
P12/ $\overline{\text{INTP1}}$		
P13/ $\overline{\text{INTP2}}$ / $\overline{\text{INTAK}}$	入力状態：個別に抵抗を介して V_{DD} に接続	
P14/ $\overline{\text{INT}}$ / $\overline{\text{POLL}}$	出力状態：オープン	
P15/TOOUT		
P16/ $\overline{\text{SCK0}}$		
P17/READY		
P20/DMARQ0		
P21/DMAAK0		
P22/ $\overline{\text{TC0}}$		
P23/DMARQ1		
P24/ $\overline{\text{DMAAK1}}$		
P25/ $\overline{\text{TC1}}$		
P26/ $\overline{\text{HLDAK}}$		
P27/HLDRQ		
PT0-PT7		抵抗を介してGNDに接続

備考 リセット解除後、各ポート端子は入力ポートとなります。

(2) ポート以外

(a) μ PD70325

端子名	推奨接続方法
TxD0	オープン
TxD1	
RxD0	抵抗を介して V_{DD} またはGNDに接続
RxD1	
$\overline{CTS0}$	入力状態：個別に抵抗を介して V_{DD} に接続 出力状態：オープン
$\overline{CTS1}$	抵抗を介して V_{DD} またはGNDに接続
\overline{REFRQ}	オープン
V_{TH}	抵抗を介してGNDに接続
D0-D7	入力状態：個別に抵抗を介して V_{DD} に接続 出力状態：オープン
A0-A19	オープン
\overline{MREQ}	
\overline{MSTB}	
R/\overline{W}	
\overline{IOSTB}	

(b) μ PD70335

端子名	推奨接続方法
TxD0	オープン
TxD1	
RxD0	抵抗を介して V_{DD} またはGNDに接続
RxD1	
$\overline{CTS0}$	入力状態：個別に抵抗を介して V_{DD} に接続 出力状態：オープン
$\overline{CTS1}$	抵抗を介して V_{DD} またはGNDに接続
\overline{REFRQ}	オープン
V_{TH}	抵抗を介してGNDに接続
D0-D15	入力状態：個別に抵抗を介して V_{DD} に接続 出力状態：オープン
A0	オープン
A9/A1-A16/A8	
A17-A18	
A19	
A18/ \overline{UBE}	
\overline{MREQ}	
\overline{MSTB}	
R/ \overline{W}	
\overline{IOSTB}	

保守 / 廃止

[× 氏]

第3章 内部CPU機能

μ PD70325, 70335は, μ PD70116, 70108のネイティブ・モードとソフトウェア・コンパチブルな16ビットCPUを持っています。

3.1 ハードウェア構成

μ PD70325, 70335の内部CPUは次に示す3つの機能ユニットに大別でき, それぞれのユニットがパイプライン式に連動して処理を行うため, バスの使用効率が向上し, 高速な命令実行が可能です。

- PAU (アドレス演算ユニット)
- EXU (実行ユニット)
- BCU (バス・コントロール・ユニット)

3.1.1 PAU (アドレス演算ユニット)

EXUから得られる情報から20ビットの物理アドレスを生成し, BCUの起動を要求します。

3.1.2 EXU (実行ユニット)

算術論理演算, データ転送など命令実行に関する基本的な処理を行います。EXUの1パイプライン・ステージは, 2-4クロックで, 1つまたは複数のパイプライン・ステージで1つの命令を構成します。

3.1.3 BCU (バス・コントロール・ユニット)

PAUで得られた20ビット物理アドレスに基づいて, 必要なバス・サイクルを起動します。同時にREADY, HOLD, リフレッシュなどのバス制御も行います。

PAUからバス・サイクルの起動要求がないときは, 命令コード先取りアドレスを発生し, 命令の先行フェッチを行います。先取りした命令コードはプリフェッチ・キューに格納されます。

BCUの1パイプライン・ステージは, 1回のバス・サイクルとなります。

プリフェッチ・キュー

μ PD70325, 70335は6バイトの命令キュー (FIFO) を持っており, BCUがプリフェッチする命令コードを最大6バイト分までストアすることができます。

キューにストアされた命令コードは, EXUによってフェッチされ実行されます。

ブランチ, コール, リターン, ブレーク命令実行時や外部割り込み処理時には, キューの内容はクリアされ, 新たなロケーションの命令がプリフェッチされます。

通常, キューに1バイト以上の空きがあるとプリフェッチを行います。

連続して実行される命令の平均実行時間が, 1つ1つの命令の命令コードのプリフェッチに必要な

クロック数をある程度越えていれば、EXUが1つの命令を実行終了したときに、続いてEXUが実行できる命令コードがキューに用意されているようになり、メモリからのフェッチ時間を命令実行時間から省けるようになります。この結果、1命令ごとにフェッチ、実行を行うほかのCPUに比べて処理速度が向上します。

ただし次の場合、キューの効果は減少します。

- ブランチ命令実行時のようにキューがクリアされる命令が多いとき
- 実行時間の短い命令が続くとき

★ 3.1.4 CPUパイプライン

μ PD70325, 70335のCPUは同期式パイプライン構造を採用しています。したがって、同時に起動されるPAU, EXU, BCUの各ユニットの1パイプライン・ステージの処理がすべてのユニットで終了したあと、次の処理を起動します。次にパイプラインの動作例を図3-1に示します。

例 次に示す命令を実行した場合のパイプラインの動きを示します。

MOV	mem, reg	...	①		
MOV	reg, reg'	...	②		
MOV	reg, mem	...	③		
ADD	reg, reg'	...	④		

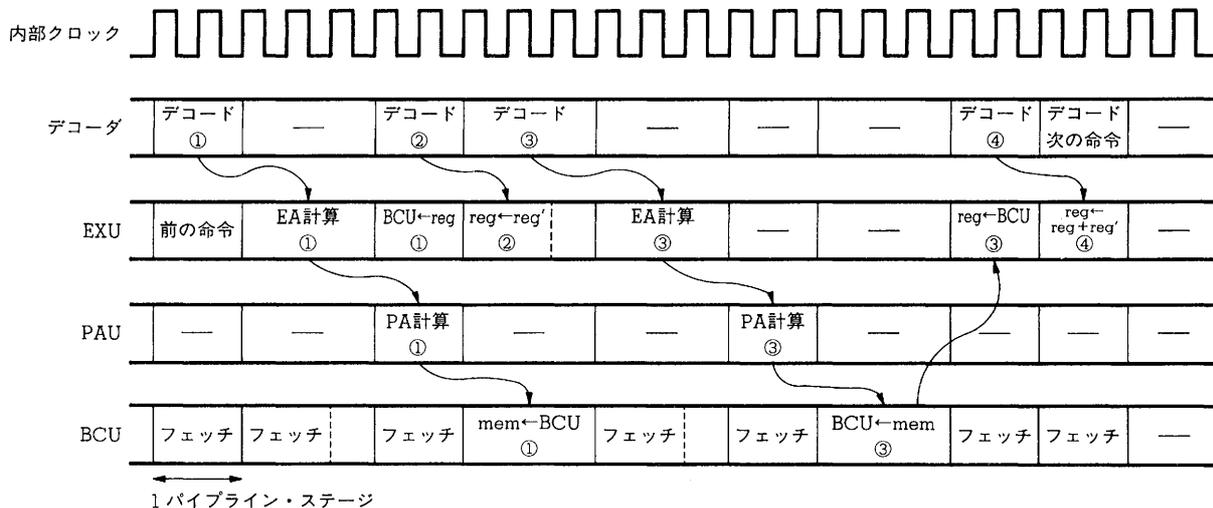
例は次の条件を想定しています。

- プリフェッチ・サイクル：2クロック（0ウエイト）
- メモリ・リード／ライト：3クロック（1ウエイト）

図 3-1 CPUパイプラインの動作例 (1/2)

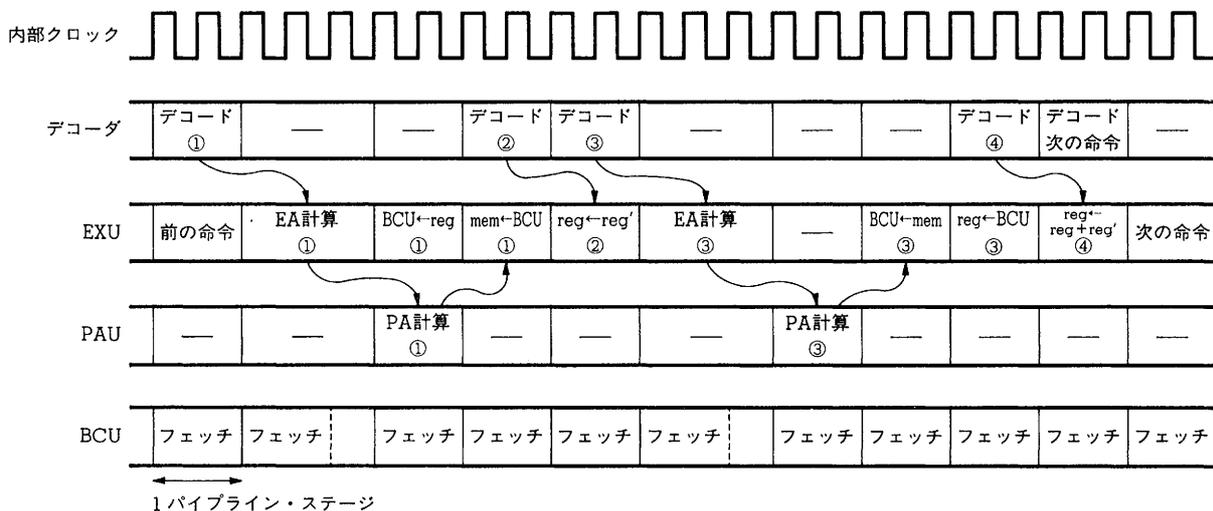
★

(a) 内部RAMアクセス禁止の場合



備考 BCUでのメモリ・ライト動作中も次の命令を実行します。

(b) 内部RAMアクセス許可の場合 (内部RAMへアクセス)

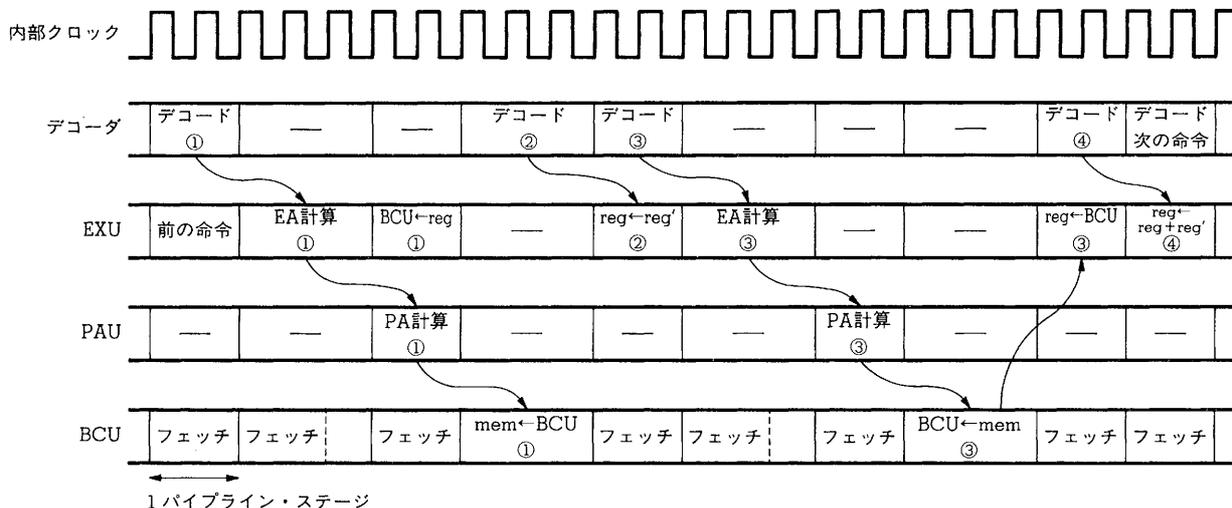


備考 内部RAMへのメモリ・ライト、メモリ・リードはEXUで行います。その間もBCUはプログラム・フェッチをします。

★

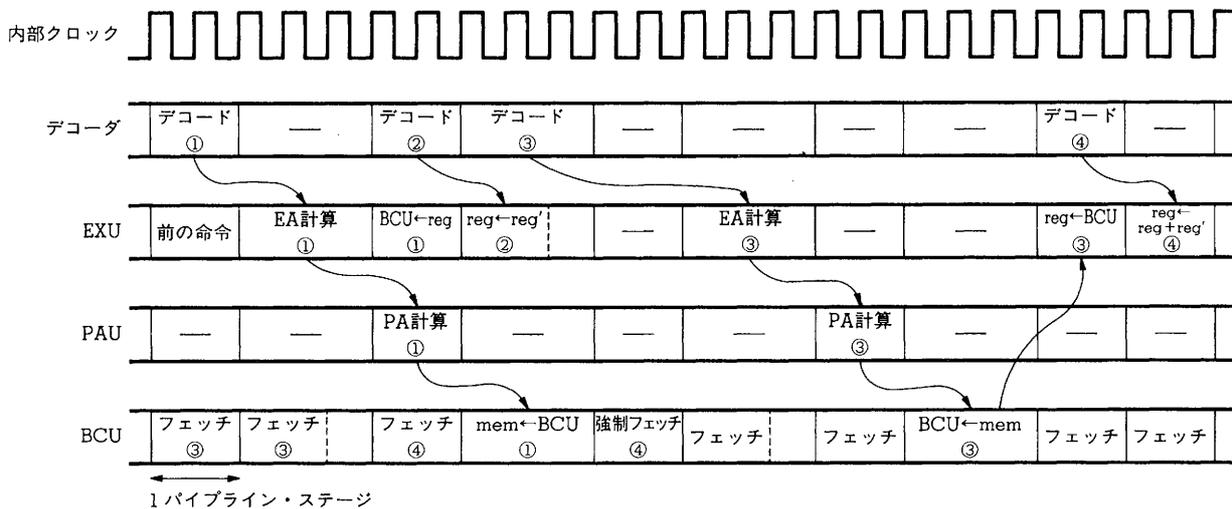
図3-1 CPUパイプラインの動作例(2/2)

(c) 内部RAMアクセス許可の場合(外部RAMへアクセス)



備考 内部RAMアクセス許可の場合、BCUがメモリ・ライト中も、EXUは次の命令へ処理を移しません。

(d) 強制プリフェッチ・サイクルが入る場合



備考 プリフェッチ・キュー内に格納されているコードが2バイト未満になると、強制プリフェッチ・サイクルが起動します。

強制プリフェッチ・サイクル

μPD70325, 70335には6バイトのプリフェッチ・キューがあります。通常、キューに1バイト以上の空きがあるとプリフェッチを行います。

キュー内の命令コードが1バイト以下になったとき、次のパイプライン・ステージは命令の実行を中止し、プリフェッチ・キュー内に2バイト以上の命令コードが格納されるまで強制的にプリフェッチ・サイクルが起動されます。

また、分岐命令、CALL命令などアドレスが連続にならない場合はプリフェッチ・キューの内容はクリアされ、新しいロケーションの命令コードを2バイト、フェッチしてから命令の実行を開始します。

3.2 レジスタ

μ PD70325, 70335のCPUは μ PD70116, 70108とコンパチブルな汎用のレジスタ・セットを持っています。また、オンチップ周辺ハードウェアの制御用に種々の特殊機能のレジスタを持っています。これらのレジスタは、すべてメモリ空間にマッピングされています。特に汎用のレジスタ・セットは、内蔵RAMと兼用になっており、内蔵RAM上に最大8バンクのレジスタ・セットを持つことができます。

また、これらのレジスタのアドレスは、4Kバイト単位でリロケート可能で、このアドレスは特殊機能レジスタの1つである内部データ領域ベース・レジスタ (IDB) によって指定されます (3.5.2参照)。

3.2.1 レジスタ・バンク

汎用のレジスタ・セットは、内蔵RAM領域にマッピングされています。汎用のレジスタ・セットはバンク形式になっており、8バンクまで設定可能で、1バンクあたり32バイト使用します。この8バンクのうちバンク0とバンク1は、マクロ・サービス・チャンネル (4.5.4参照) 用にも使用されます。また、データ・メモリとしてもアクセスできます (3.5.4参照)。

通常CPUはレジスタ・バンク7を用いてプログラムを実行し、割り込みによって他のレジスタ・バンクへ自動的に切り替えられます。割り込みによって切り替えられたレジスタ・バンクから元のレジスタ・バンクへ戻るには、割り込みからの復帰命令RETRBI (μ PD70108, 70116に対する追加命令) によるのみ行えます (4.5.2参照)。

レジスタ・バンクの構成は、図3-2のようになっています。レジスタ・バンク中の(+00H), (+01H)は、レジスタ・バンクを使用する場合に予約領域となります。汎用のレジスタ・セットは、各レジスタ・バンクの開始アドレスからのオフセットで(+08H)–(+1FH)の領域にマッピングされています。なお、(+02H)–(+07H)の領域は、レジスタ・バンクの切り替え時に使用されるため一般用途には使用できません。

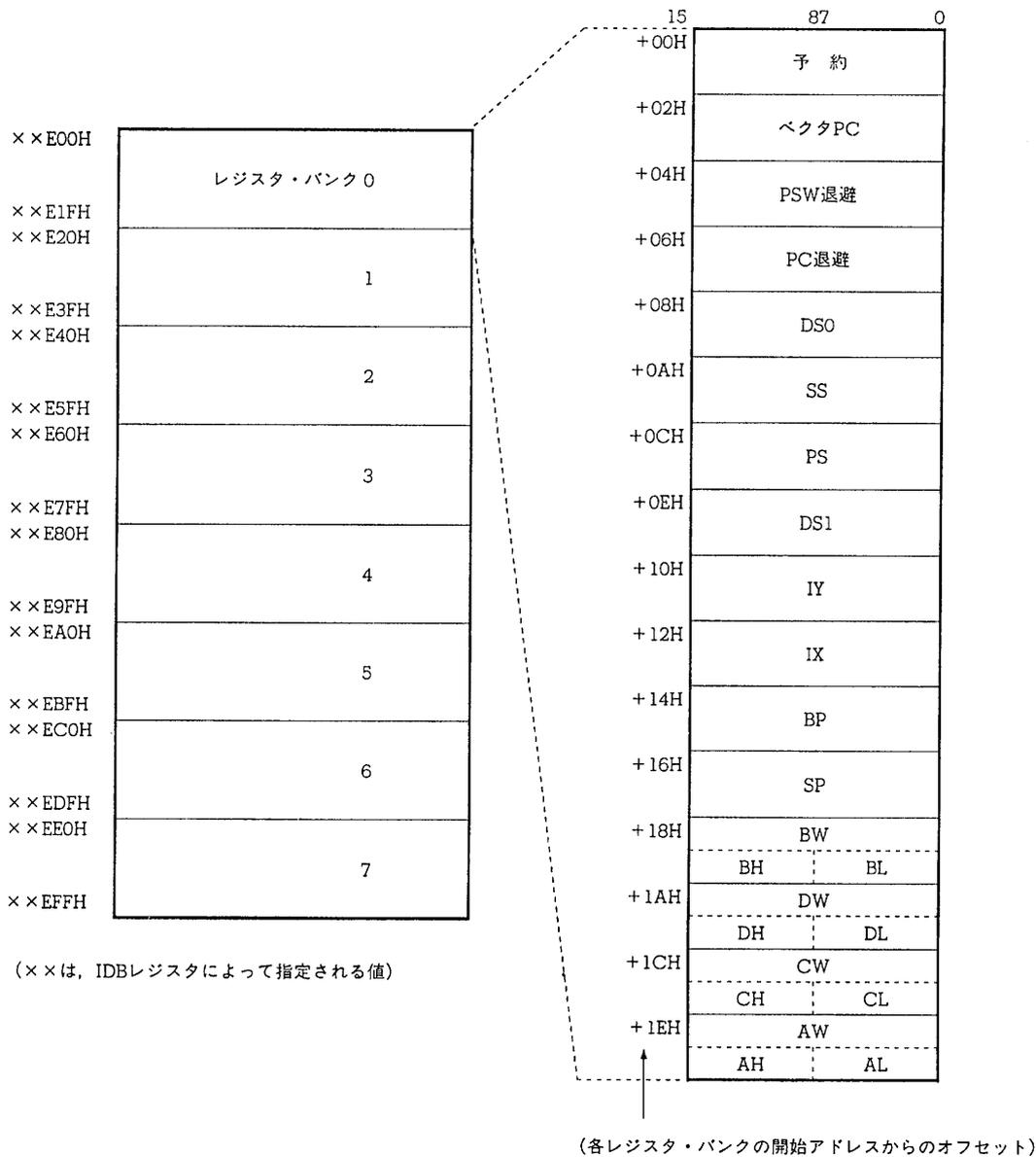
(+02H)の領域へは、レジスタ・バンクが切り替わったときにPCへロードされる値、すなわち割り込み処理ルーチンの先頭アドレスのオフセットをセットします。

(+04H)は、レジスタ・バンクが切り替わったときにPSWを退避するための領域です。

(+06H)は、レジスタ・バンクが切り替わったときにPCを退避するための領域です (4.5.2参照)。

リセット後には、レジスタ・バンク7が自動的に選択されます。また、リセット後のセグメント・レジスタ (3.2.4参照)の初期化は、レジスタ・バンク7のレジスタにのみ行います。

図3-2 レジスタ・バンクの構成



3.2.2 汎用レジスタ (AW, BW, CW, DW)

汎用レジスタには4つの16ビット・レジスタがあります。これらのレジスタは、16ビット・レジスタとしてのアクセスはもちろん、各レジスタを上位、下位の8ビットに分けて、8ビット・レジスタ (AH, AL, BH, BL, CH, CL, DH, DL) としてもアクセス可能です。

これらのレジスタは転送命令、算術命令、論理演算命令などの広範の命令に対して、8ビット・レジスタまたは16ビット・レジスタとして利用されます。

また、各レジスタは、次に示すような特定の命令処理にデフォルト・レジスタとして使用されます。

AW：ワード乗算/除算，ワード入力/出力，データ変換

AL：バイト乗算/除算，バイト入力/出力，翻訳，BCDローテーション，データ変換

AH：バイト乗算/除算

BW：翻訳

CW：ループ制御ブランチ，リピート・プリフィクス

CL：シフト命令，ローテーション命令，BCD演算

DW：ワード乗算/除算，間接アドレッシング入力/出力

これらのレジスタは、内蔵RAMにマッピングされています。そのアドレスは次の値になります。

(IDBレジスタ^注の値×4096) + (0E00H) + (レジスタ・バンクの番号×32) + (レジスタごとのオフセット)

注 IDBレジスタは3.5.2を参照してください。

表 3-1 汎用レジスタのオフセット

レジスタ	オフセット	レジスタ	オフセット
AW	1EH	AL	1EH
		AH	1FH
BW	18H	BL	18H
		BH	19H
CW	1CH	CL	1CH
		CH	1DH
DW	1AH	DL	1AH
		DH	1BH

3.2.3 ポインタ (SP, BP) とインデクス・レジスタ (IX, IY)

ベース・アドレッシング (BP)、インデクス・アドレッシング (IX, IY)、ベース・インデクス・アドレッシング (BP, IX, IY) 等によるメモリ・アクセス時に、ベース・ポインタまたはインデクス・レジスタとして使用される16ビットのレジスタです。また、SPはスタック操作時のポインタとして使用されます。さらに、汎用レジスタと同様、転送、算術演算などの命令に対して使用されますが、この場合



8ビット・レジスタとしては使用できません。さらに、各レジスタは、次に示すような特定の処理にディフォルト・レジスタとして使用されます。

SP：スタック操作

IX：ブロック転送, BCDストリング演算のソース側

IY：ブロック転送, BCDストリング演算のデスティネーション側

これらのレジスタは、内蔵RAMにマッピングされており、そのアドレスは次の値になります。

(IDBレジスタ^注の値×4096) + (0E00H) + (レジスタ・バンクの番号×32) + (レジスタごとのオフセット)。

注 IDBレジスタは3.5.2を参照してください。

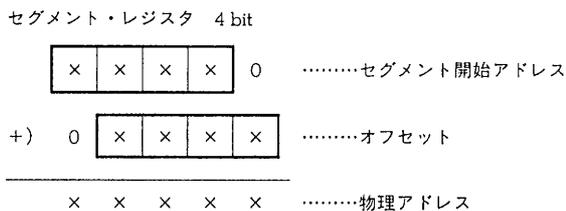
表3-2 ポインタとインデクス・レジスタのオフセット

レジスタ	オフセット
SP	16H
BP	14H
IX	12H
IY	10H

3.2.4 セグメント・レジスタ (PS, SS, DS0, DS1)

CPUは、64Kバイト単位の論理セグメントにメモリ空間を分割します。セグメント・レジスタは、各セグメントの開始アドレスを指定します。開始アドレスからのオフセットは、別のレジスタまたは実効アドレスで指定されます。

したがって、物理アドレスは次に示すように生成されます。



セグメント・レジスタには、PS(Program Segment), SS(Stack Segment), DS0 (Data Segment0), DS1 (Data Segment1) の4種があります。それぞれのセグメントは、以下の場合に使用されます。

PS：プログラム・フェッチ

SS：スタック操作命令, BPをベース・レジスタとしたアドレッシング

DS0：一般変数のアクセス, ブロック転送命令等のソース・ブロック・データのアクセス

DS1：ブロック転送命令等のデスティネーション・ブロック・データのアクセス

ただし、セグメント・オーバーライド・プリフィクスを用いることにより、DS0の代わりに他のセグメントを使用することができます。またBPをベース・レジスタとしたアドレッシングにおいても、同様にSSの代わりに他のセグメントを使用することが可能です。

リセット時には、レジスタ・バンク7のPSはFFFFHに、SS、DS0、DS1は0000Hにイニシャライズされます。

これらのレジスタは内蔵RAMにマッピングされており、そのアドレスは次の値になります。
 $(\text{IDBレジスタ注の値} \times 4096) + (0E00H) + (\text{レジスタ・バンクの番号} \times 32) + (\text{レジスタごとのオフセット})$ 。

注 IDBレジスタは**3.5.2**を参照してください。

表 3-3 セグメント・レジスタのオフセット

レジスタ	オフセット
DS0	08H
DS1	0EH
SS	0AH
PS	0CH

3.2.5 内部データ領域ベース・レジスタ (IDB)

IDBレジスタは、内蔵RAM（汎用レジスタと兼用）とオンチップ周辺ハードウェアの制御用などの特殊機能レジスタ（**3.5.3**参照）の領域である内部データ領域（**3.5.1**参照）のアドレスを決定するための8ビットのレジスタです。このレジスタは、FFFFHと、(自分自身の値×4096+FFFH)の2か所のアドレスで参照することができます（**3.5.2**参照）。

3.2.6 特殊機能レジスタ

μPD70325, 70335は、オンチップ周辺ハードウェアの制御用に特殊な機能を持ったレジスタ群を持っています。これらのレジスタ群は、内部データ領域内の特殊機能レジスタ領域にメモリ・マッピングされており、通常のメモリと同様の方法で読み書きを行います（**3.5.3**参照）。

また、追加されたBTCLR命令は、この特殊機能レジスタに対してのみ有効です。

3.3 プログラム・カウンタ (PC)

CPUが実行しようとしているプログラム・メモリ・アドレスのオフセットを保持する16ビット・バイナリ・カウンタです。

PCは、命令キューから命令バイトをフェッチすることにインクリメントされます。また、ブランチ、コール、リターン、ブレイク命令実行時などには、新たなロケーションがロードされます。

また、リセット時には0000HがPCにロードされます。なお、PSはリセット時にFFFFHに初期化されますので、リセット後CPUはFFFF0H番地から実行を開始します。

3.4 プログラム・ステータス・ワード (PSW)

PSW (プログラム・ステータス・ワード) は、6種類のステータス・フラグと5種類のコントロール・フラグおよびユーザ・フラグで構成されます。

○ステータス・フラグ

- ・V (Overflow)
- ・S (Sign)
- ・Z (Zero)
- ・AC (Auxiliary Carry)
- ・P (Parity)
- ・CY (Carry)

○コントロール・フラグ

- ・RB0-RB2 (Register Bank0-2)
- ・DIR (Direction)
- ・IE (Interrupt Enable)
- ・BRK (Break)
- ・ $\overline{\text{IBRK}}$ (I/O Break)

○ユーザ・フラグ

- FO (user Flag0)
- F1 (user Flag1)

ステータス・フラグは、各種命令実行の結果 (データ値) に従って自動的にセット (1)、リセット (0) されます。さらにCYフラグは、命令によっても直接にセット、リセット、反転が可能です。

コントロール・フラグは、命令によってセット、リセットされ、CPUの動作を制御します。IEとBRKフラグは、割り込み処理が起動されると必ずリセットされます。

ユーザ・フラグはユーザが使用できるフラグで、命令でセット、リセットおよびテストが可能です。

PSWのビット構成を次に示します。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	RB2	RB1	RBO	V	DIR	IE	BRK	S	Z	F1	AC	FO	P	$\overline{\text{IBRK}}$	CY

PUSH/POP命令によって、PSWの内容をスタックに退避させたり、復帰させたりすることができます。ただし、POP PSW命令で復帰する際には、ビット12-14 (RBO-2) はPSWにもどりません。

また、PSWの下位8ビットをMOV命令によってAHレジスタに退避させたり、復帰させたりすることができます。

割り込みの発生により、IE、BRKがリセットされる前に、PSWの内容をスタックへ自動的に退避させます。

RESET入力によって、RBO-RB2、 $\overline{\text{IBRK}}$ およびビット15はセット(1)され、その他はリセット(0)されます。

注意 PSWのビット15は、必ず(1)にしてください。

3.4.1 CY (Carry Flag)

(1) 2進加減算

バイト演算の場合には、演算結果のビット7からのキャリーまたはボローがあったときセットされ、なければリセットされます。

ワード演算の場合には、演算結果のビット15からのキャリーまたはボローがあったときセットされ、なければリセットされます。

インクリメント、デクリメント命令では変化しません。

(2) 論理演算

演算結果にかかわらずリセットされます。

(3) 2進乗算

符号なしバイト演算の結果、AHが0ならばリセットされ、0以外ならばセットされます。

符号付きバイト演算の結果、AHがALのサイン拡張になっていれば、リセットされ、それ以外ではセットされます。

符号なしワード演算の結果、DWが0ならばリセットされ、0以外ならばセットされます。

符号付きワード演算の結果、DWがAWのサイン拡張になっていればリセットされ、それ以外ではセットされます。

8ビット・イミディエイト演算の場合は、積が16ビット以内のときにはリセットされ、16ビットを越える場合にはセットされます。

(4) 2進除算

不 定

(5) シフト／ローテート

CYを含むシフトおよびローテートの場合は、CYへシフトされるビットが1ならばセットされ、0ならばリセットされます。

3.4.2 P (Parity Flag)**(1) 2進加減算，論理演算，シフト**

演算の結果を構成するビットの下位8ビットのうち、1のビットの数が偶数のときセットされ、奇数のときリセットされます。

結果がオール0のときにはセットされます。

(2) 2進乗除算

不 定

3.4.3 AC (Auxiliary Flag)**(1) 2進加減算**

バイト演算の場合、下位4ビットから上位4ビットへのキャリーまたは上位4ビットから下位4ビットへのポローがあるときセットされ、それ以外のときリセットされます。

ワード演算では、下位バイトについてバイト演算の場合と同様の動作を行います。

(2) 論理演算，2進乗除算，シフト／ローテート

不 定

3.4.4 Z (Zero Flag)**(1) 2進加減算，論理演算，シフト／ローテート**

バイト演算の場合には結果の8ビットが、ワード演算の場合には結果の16ビットが、すべて0の場合にセットされ、それ以外のときにはリセットされます。

(2) 2進乗除算

不 定

3.4.5 S (Sign Flag)

(1) 2進加減算, 論理演算, シフト/ローテート

バイト演算の場合, 結果のビット7が1のときセットされ, 0のときリセットされます。

ワード演算の場合, 結果のビット15が1のときセットされ, 0のときリセットされます。

(2) 2進乗除算

不 定

3.4.6 V (Overflow Flag)

(1) 2進加減算

バイト演算の場合, ビット7とビット6からのキャリーが違っていればセットされ, 同じであればリセットされます。

ワード演算の場合, ビット15とビット14からのキャリーが違っていればセットされ, 同じであればリセットされます。

(2) 論理演算

演算結果にかかわらずリセットされます。

(3) 2進乗算

符号なしバイト演算の結果, AHが0ならばリセットされ, 0以外ならばセットされます。

符号付きバイト演算の結果, AHがALのサイン拡張になっていればリセットされ, それ以外ではセットされません。

符号なしワード演算の結果, DWが0ならばリセットされ, 0以外ならばセットされます。

符号付きワード演算の結果, DWがAWのサイン拡張になっていればリセットされ, それ以外ではセットされません。

8ビット・イミディエイト演算の場合は, 積が16ビット以内のときにはリセットされ, 16ビットを越えるときにはセットされます。

(4) 2進除算

演算の結果にかかわらずリセットされます。

(5) シフト/ローテート

左1ビット・シフト/ローテートの場合, 演算結果において,

CY=最上位ビットのとき:リセット

CY≠最上位ビットのとき：セット

右1ビット・シフト／ローテートの場合、演算結果において、

最上位ビット＝最上位の次の下位ビットのとき：リセット

最上位ビット≠最上位の次の下位ビットのとき：セット

多ビット・シフト／ローテートの場合は不定となります。

3.4.7 $\overline{\text{IBRK}}$ (I/O Break Flag)

入出力命令時のソフトウェア割り込みの発生を制御します。

$\overline{\text{IBRK}}=0$ のとき、入出力命令を実行しようとするソフトウェア割り込み(割り込みベクタ19)が発生します。このため、ソフトウェアによる入出力命令のエミュレートができます。

$\overline{\text{IBRK}}=1$ のとき、入出力命令を実行してもソフトウェア割り込みは発生しません。

3.4.8 BRK (Break Flag)

BRKフラグがセットされていれば、1命令を実行するとソフトウェア割り込み(割り込みベクタ1)が発生します。このため、1命令ずつのトレースが可能となります。

PSWの一部としてスタックに退避している状態でのみ、メモリ操作命令によってセットされます。セット後にPSWへ復帰すると有効になります。

3.4.9 IE (Interrupt Enable Flag)

EI命令でセットされて割り込みを許可状態にし、DI命令でリセットされて割り込みを禁止状態にします。

3.4.10 DIR (Direction Flag)

SET1 DIR命令でセットされ、CLR1 DIR命令でリセットされます。

DIRフラグがセットされていると、ブロック転送／入出力系命令において上位アドレスから下位アドレスへ向かって処理を行い、リセットされていると、下位アドレスから上位アドレスへ向かって処理を行います。

3.4.11 RBO-RB2 (Register Bank0-2 Flag)

RBO-RB2で、内部RAM内に設定された8個のレジスタ・バンクの中から現在使用中のレジスタ・バンクを示します。

POP PSW命令によるスタックからの復帰はしません。

注意 割り込み処理ルーチンの中で、スタック、PSW退避レジスタに退避しているRBO～RB2の値を変えないでください。

3.4.12 FO, F1 (User Flag0, 1 Flag)

ユーザが自由に使用できるフラグです。

PSWに関する命令でこのフラグのセット、リセットができます。また、FLAGレジスタ(特殊機能レジスタ)によってもセット、リセット、テストができます。

FLAGレジスタのビット構成を図3-3に示します。

図3-3 FLAG

7	6	5	4	3	2	1	0
—	—	F1	—	F0	—	—	—

3.5 メモリ空間

μPD70325, 70335は、1 Mバイトのメモリ空間を持っています。図3-4にメモリ・マップを示します。

00000H-003FFH : ベクタ領域 (ベクタとして使用しない場合、ほかの目的に利用できます。)

××E00H-××FFFH : 内部データ領域 (4 Kバイト単位でロケーションを変えられます。)

FFFFCH-FFFFEH : 予約領域

FFFFFFH : IDBレジスタ

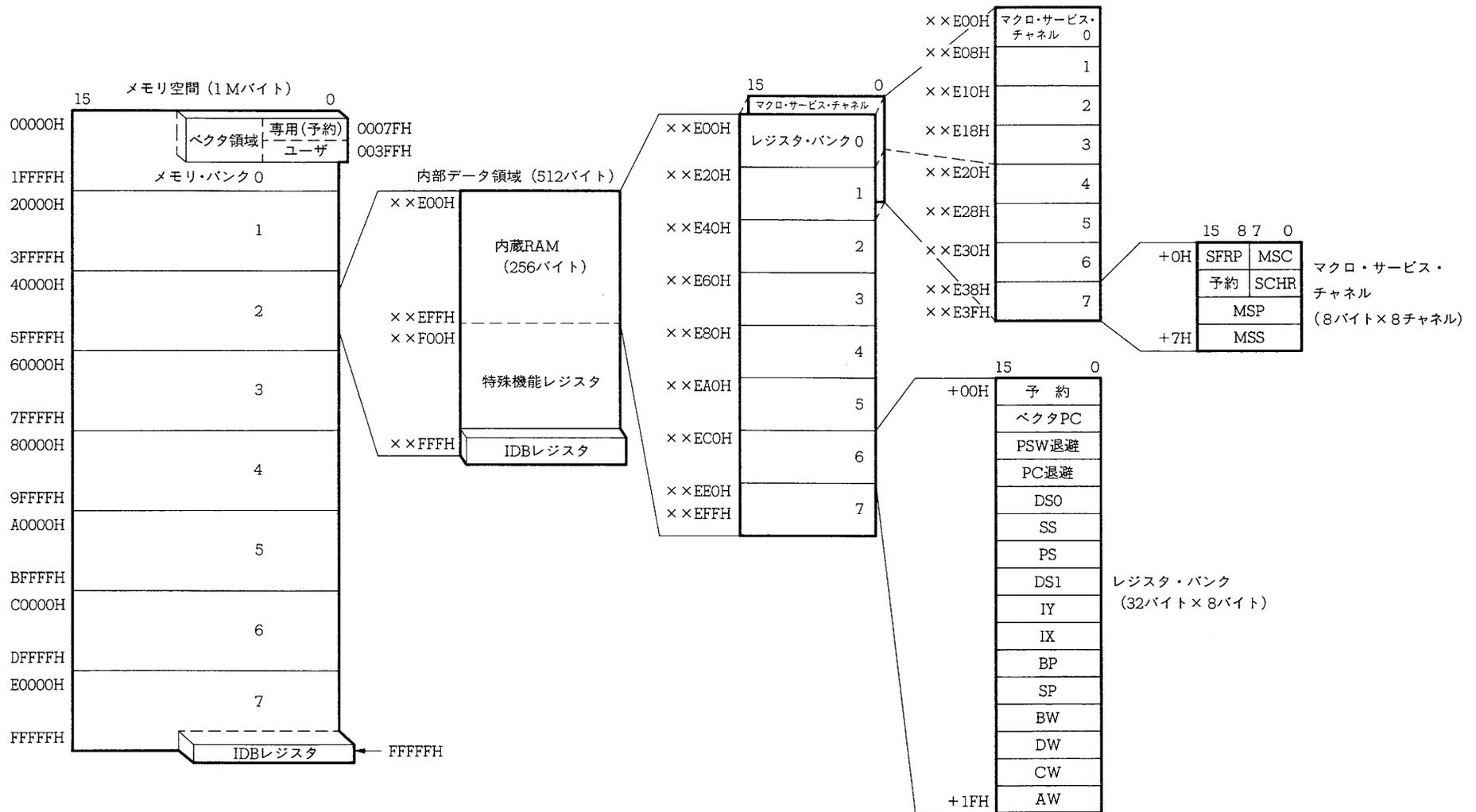
備考 ×× : IDBレジスタの値

メモリ空間のアクセスでは、128 Kバイトごとにウェイト・サイクルをプログラマブルに挿入できます。メモリ・マップされていない空間に対してはアクセスしないでください。

物理アドレスについては、3.2.4を参照してください。

- 注意1. セグメント内オフセットがFFFFFFH番地でワード参照を行うと、2 バイト目の参照番地は同じセグメント内の0000H番地ではなく、FFFFFFH+1番地となります(V20, V30では0000H番地)。
2. セグメント値とオフセット値の組み合わせが、物理アドレスFFFFFFH番地を超えた場合、次に生成するアドレスは00000H番地になります。

図 3-4 メモリ・マップ



3.5.1 内部データ領域

内部データ領域は、内蔵RAM領域と特殊機能レジスタ領域を併せた512バイトの領域で、1 Mバイトのメモリ空間内を4 Kバイト単位でリロケート可能です。内部データ領域のベース・アドレスの設定は、IDBレジスタ（内部データ領域ベース・レジスタ）によって行います。20ビットある内部データ領域ベース・アドレスの上位8ビットをIDBレジスタで指定し、下位12ビットはE00H固定です。

$\overline{\text{RESET}}$ 入力時、IDBレジスタがFFHに初期化されるため、内部データ領域はFFE00H-FFFFFFHにリロケートされます。

(1) 内部データ領域のアクセス

内部データ領域は、図3-4に示すように外部メモリ領域と重なって割り付けてあります。重なっている各メモリ領域は、メモリ・アクセスの条件によって対象となる領域が区別されます。メモリ・アクセスの条件とアクセスの対象となるメモリ領域を図3-5(1)、(2)に示します。

内部データ領域からのデータ・フェッチは2クロックで行います。

内部データ領域および内蔵ROMをアクセスしたときは、 $\overline{\text{MREQ}}$ 、 $\overline{\text{MSTB}}$ 、 $\overline{\text{IOSTB}}$ などの制御信号は外部へ出力されません。

(2) 内蔵RAM領域

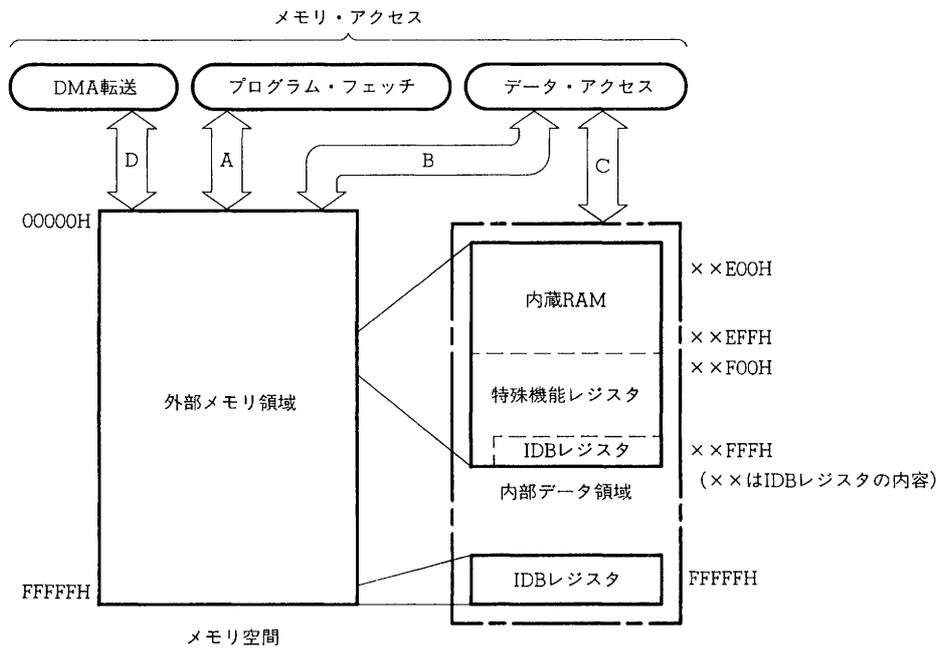
内部データ領域の下位256バイト（ $\times\times\text{E00H}-\times\times\text{FFFH}$ ： $\times\times$ はIDBレジスタの内容）が内蔵RAM領域です。内蔵RAM領域には、普通のRAMとしての用途のほかにレジスタ・バンク、マクロ・サービス・チャンネルが機能的に割り付けてあります（3.5.4参照）。特殊機能レジスタであるプロセッサ・コントロール・レジスタ（PRC）のビット6（RAMEN）をリセット(0)することによって、内蔵RAMを通常のRAMとしてアクセスすることを禁止できます。

RAMENビットをリセット(0)していても、マクロ・サービス機能、DMA転送は、正常に動作します。ただし、このときマクロ・サービス、DMAの再設定を行うことはできません。RAMENビットがセット(1)されているときに行ってください。

(3) 特殊機能レジスタ領域

内部データ領域の上位256バイト（ $\times\times\text{F00H}-\times\times\text{FFFH}$ 、 $\times\times$ はIDBレジスタの内容）が特殊機能レジスタ領域となります。特殊機能レジスタには、オンチップ周辺ハードウェアのモード・レジスタ、制御レジスタなどの特別な機能を割り付けたレジスタ群が割り付けてあります（3.5.3参照）。

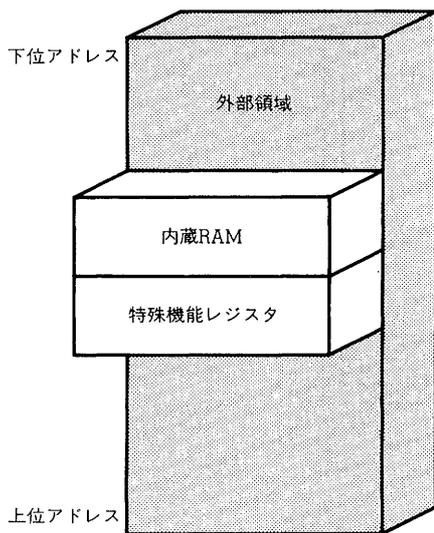
図 3-5 メモリ空間のアクセス条件



- A: プログラム・フェッチは内部データ領域以外がアクセスの対象
- B: 内部データ領域外のデータ・アクセス, または内蔵RAMアクセス禁止時の内蔵RAM領域に相当するアドレスのデータ・アクセス
- C: データ・アクセスは, Bの条件に相当しない場合, 内部データ領域がアクセスの対象として優先
- D: DMA転送は, 外部領域がアクセスの対象

(1) プログラム・フェッチ

アクセス対象：外部領域，内部ROM領域

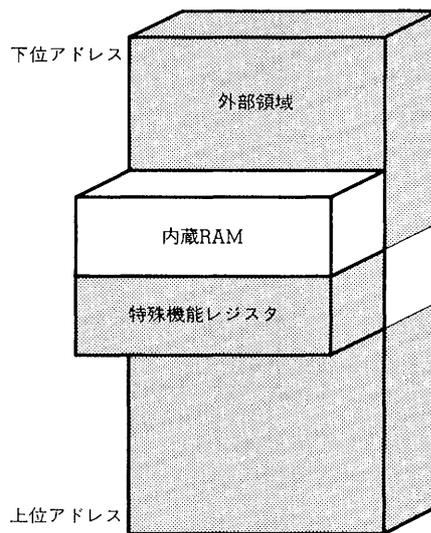


(2) データ・アクセス

(a) 内蔵RAMアクセス禁止時

RAMEN=0 (PRC)

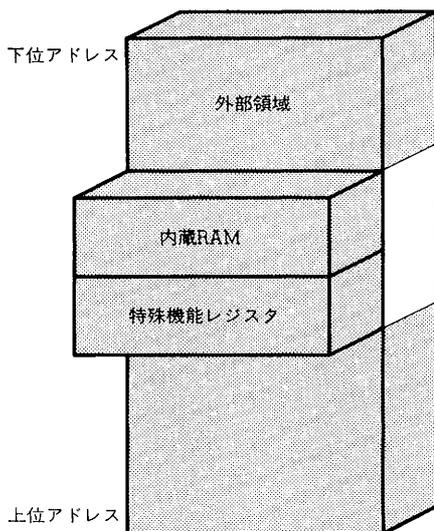
アクセス対象：特殊機能レジスタ領域が優先される



(b) 内蔵RAMアクセス許可時

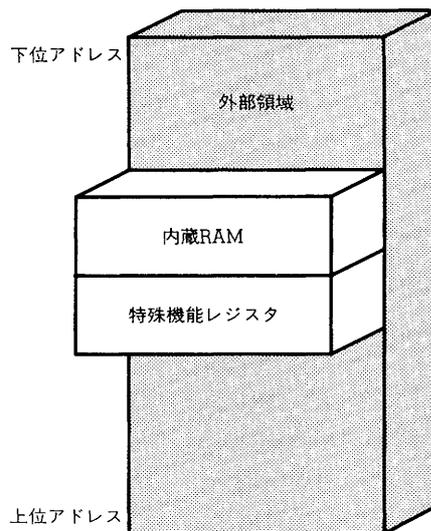
RAMEN=1 (PRC)

アクセス対象：内部データ領域が優先される



(c) DMA転送時

アクセス対象：外部領域だけ



 はアクセス対象

3.5.2 内部データ領域ベース・レジスタ (IDB)

内部データ領域（後述する内蔵RAMと特殊機能レジスタの領域）の物理アドレスを決定するレジスタです。内部データ領域ベース・アドレスの上位8ビットを指定します。下位12ビットはE00H固定となります。IDBとしてのアクセスだけ行ってください。

IDBには、特殊機能レジスタ領域内の××FFFH（××はIDBレジスタの内容）と固定アドレスFFFFHの2つのアドレスが割り付けてあります。この2つのアドレスのどちらか一方に対するメモリ・アクセスで、IDBの変更や参照を行うことができます。

IDBには、リセット時にFFHがセットされます。したがって、内部データ領域のベース・アドレスはFFE00Hとなります。

3.5.3 特殊機能レジスタ領域

××FO0H-××FFFH（××はIDBレジスタの内容）にオンチップ周辺ハードウェアのモード・レジスタ、制御レジスタなどの特別な機能を割り付けられたレジスタ群がマッピングされています。

これらの領域からはプログラム・フェッチを行うことができません。

メモリ・アクセスによって、特殊機能レジスタを操作します。

表3-4に特殊機能レジスタの一覧表を示します。表中の項目の意味は次のとおりです。

- ・略号……………特殊機能レジスタの名称を記号で表わしたものです。命令のオペランドに相当します。
- ・R/W ……………該当する機能レジスタが、読み出し／書き込み可能かどうかを示します。
R/W：読み出し（Read）／書き込み（Write）可能
R : 読み出し（Read）のみ可能
W : 書き込み（Write）のみ可能
- ・操作方法……………各レジスタが、16ビット操作、8ビット操作、1ビット操作のうち可能なものを示します。
- ・RESET時…………… $\overline{\text{RESET}}$ 入力時の各レジスタの状態を示します。

なお、アドレス上位8ビットの××はIDBレジスタの内容です。

また、記載されていないアドレスの部分は予約されています。読み出し時の内容は不定となり、書き込み動作は意味をもちません。

表3-4 特殊機能レジスタ一覧(1/3)

分類	アドレス	特殊機能レジスタ名称	略号	R/W	操作方法 (ビット)	RESET時			
ポート	××F00H	ポート0	P0	R/W	8/1	不定			
	××F01H	ポート0モード・レジスタ	PM0	W	8	FFH			
	××F02H	ポート0モード・コントロール・レジスタ	PMCO			00H			
	××F08H	ポート1	P1	R/W	8/1	不定			
	××F09H	ポート1モード・レジスタ	PM1	W	8	FFH			
	××F0AH	ポート1モード・コントロール・レジスタ	PMC1			00H			
	××F10H	ポート2	P2	P/W	8/1	不定			
	××F11H	ポート2モード・レジスタ	PM2	W	8	FFH			
	××F12H	ポート2モード・コントロール・レジスタ	PMC2			00H			
	××F38H	ポートT	PT	R	8	不定			
××F3BH	ポートTモード・レジスタ	PMT	R/W	8/1	00H				
割り込み制御	××F40H	外部割り込みモード・レジスタ	INTM	R/W	8/1	00H			
	××F44H	外部割り込みマクロ・サービス制御レジスタ0 ^{注4}	EMSO			不定			
	××F45H	外部割り込みマクロ・サービス制御レジスタ1 ^{注4}	EMS1						
	××F46H	外部割り込みマクロ・サービス制御レジスタ2 ^{注4}	EMS2						
	××F4CH	外部割り込み要求制御レジスタ0 ^{注4}	EXICO			47H			
	××F4DH	外部割り込み要求制御レジスタ1 ^{注4}	EXIC1						
	××F4EH	外部割り込み要求制御レジスタ2 ^{注4}	EXIC2						
	××FEFH	割り込み要因レジスタ ^{注4}	IRQS	R	8	不定			
××FFCH	割り込みプライオリティ・レジスタ ^{注4}	ISPR	R	8	00H				
シリアル・インタフェース・チャンネル1	××F60H	受信バッファ・レジスタ0	RxB0	R	8	不定			
	××F62H	送信バッファ・レジスタ0	TxB0	W	8	不定			
	××F65H	シリアル受信マクロ・サービス制御レジスタ0 ^{注4}	SRMSO	R/W	8/1	不定			
	××F66H	シリアル送信マクロ・サービス制御レジスタ0 ^{注4}	STMSO						
	××F68H	シリアル・モード・レジスタ0	SCMO			00H			
	××F69H	シリアル制御レジスタ0	SCCO						
	××F6AH	ボー・レート・ジェネレータ・レジスタ0	BRG0	R	8	60H			
	××F6BH	シリアル・ステータス・レジスタ0	SCSO						
	××F6CH	シリアル・エラー割り込み要求制御レジスタ0 ^{注4}	SEICO				R/W	8/1	47H
	××F6DH	シリアル受信割り込み要求制御レジスタ0 ^{注4}	SRICO						
××F6EH	シリアル送信割り込み要求制御レジスタ0 ^{注4}	STICO							



表 3-4 特殊機能レジスタ一覧 (2/3)

分類	アドレス	特殊機能レジスタ名称	略号	R/W	操作方法 (ビット)	RESET時
シリアル・インタフェース・チャネル2	××F70H	受信バッファ・レジスタ1	RxB1	R	8	不定
	××F72H	送信バッファ・レジスタ1	TxB1	W		
	××F75H	シリアル受信マクロ・サービス制御レジスタ1 ^{注4}	SRMS1	R/W	8/1	不定
	××F76H	シリアル送信マクロ・サービス制御レジスタ1 ^{注4}	STMS1			不定
	××F78H	シリアル・モード・レジスタ1	SCM1			00H
	××F79H	シリアル制御レジスタ1	SCC1			
	××F7AH	ポーレート・ジェネレータ・レジスタ1	BRG1			
	××F7BH	シリアル・ステータス・レジスタ1	SCS1	R	8	60H
	××F7CH	シリアル・エラー割り込み要求制御レジスタ1 ^{注4}	SEIC1	R/W	8/1	47H
	××F7DH	シリアル受信割り込み要求制御レジスタ1 ^{注4}	SRIC1			
	××F7EH	シリアル送信割り込み要求制御レジスタ1 ^{注4}	STIC1			
タイマ・ユニット	××F80H	タイマ・レジスタ0 ^{注5}	TM0	R/W	16	不定
	××F82H	モジュロ/タイマ・レジスタ0 ^{注5}	MDO			
	××F88H	タイマ・レジスタ1 ^{注5}	TM1			
	××F8AH	モジュロ/タイマ・レジスタ1 ^{注5}	MD1	R/W	8/1	00H
	××F90H	タイマ・コントロール・レジスタ0 ^{注5}	TMC0			
	××F91H	タイマ・コントロール・レジスタ1 ^{注5}	TMC1	R/W	8/1	不定
	××F94H	タイマ・ユニット・マクロ・サービス制御レジスタ0 ^{注4}	TMMS0			
	××F95H	タイマ・ユニット・マクロ・サービス制御レジスタ1 ^{注4}	TMMS1			
	××F96H	タイマ・ユニット・マクロ・サービス制御レジスタ2 ^{注4}	TMMS2			
	××F9CH	タイマ・ユニット割り込み要求制御レジスタ0 ^{注4}	TMIC0			47H
	××F9DH	タイマ・ユニット割り込み要求制御レジスタ1 ^{注4}	TMIC1			
××F9EH	タイマ・ユニット割り込み要求制御レジスタ2 ^{注4}	TMIC2				

表 3-4 特殊機能レジスタ一覧 (3/3)

分類	アドレス	特殊機能レジスタ名称	略号	R/W	操作方法 (ビット)	RESET時		
DMA コントローラ	××FA0H	DMAコントロール・レジスタ 0	DMAC0	R/W	8/1	保持 ^{注6}		
	××FA1H	DMAモード・レジスタ 0	DMAM0			00H		
	××FA2H	DMAコントロール・レジスタ 1	DMAC1			保持 ^{注6}		
	××FA3H	DMAモード・レジスタ 1	DMAM1			00H		
	××FACH	DMA割り込み要求制御レジスタ 0 ^{注4}	DIC0	R/W	8/1	47H		
	××FADH	DMA割り込み要求制御レジスタ 1 ^{注4}	DIC1					
	××FC0H	ソース・アドレス・ポインタ 0 (下位)	SAR0L	R/W	16/8	不 定		
	××FC1H	ソース・アドレス・ポインタ 0 (中位)	SAR0M		8			
	××FC2H	ソース・アドレス・ポインタ 0 (上位)	SAR0H		16/8			
	××FC4H	デスティネーション・アドレス・ポインタ 0 (下位)	DAR0L		8			
	××FC5H	デスティネーション・アドレス・ポインタ 0 (中位)	DAR0M		16/8			
	××FC6H	デスティネーション・アドレス・ポインタ 0 (上位)	DAR0H		8			
	××FC8H	ターミナル・カウンタ 0 (下位)	TC0L		16/8			
	××FC9H	ターミナル・カウンタ 0 (上位)	TC0H		16/8			
	××FD0H	ソース・アドレス・ポインタ 1 (下位)	SAR1L		8			
	××FD1H	ソース・アドレス・ポインタ 1 (中位)	SAR1M		16/8			
	××FD2H	ソース・アドレス・ポインタ 1 (上位)	SAR1H		8			
	××FD4H	デスティネーション・アドレス・ポインタ 1 (下位)	DAR1L		16/8			
	××FD5H	デスティネーション・アドレス・ポインタ 1 (中位)	DAR1M		8			
	××FD6H	デスティネーション・アドレス・ポインタ 1 (上位)	DAR1H		16/8			
××FD8H	ターミナル・カウンタ 1 (下位)	TC1L	8					
××FD9H	ターミナル・カウンタ 1 (上位)	TC1H	16/8					
システム・ コントローラ	××FE0H	スタンバイ・コントロール・レジスタ	STBC		R/W ^{注1}		8/1	保持 ^{注2}
	××FE1H	リフレッシュ・モード・レジスタ	RFM		R/W		8/1	FCH
	××FE8H	ウェイト・コントロール・レジスタ	WTC		R/W		16/8	FFFFH
	××FEAH	ユーザ・フラグ・レジスタ ^{注3}	FLAG		R/W		8/1	00H
	××FEBH	プロセッサ・コントロール・レジスタ	PRC	R/W	8/1	4EH		
	××FECH	タイム・ベース割り込み要求制御レジスタ ^{注4}	TBIC			47H		
	××FFFH	内部データ領域ベース・レジスタ ^{注4}	IDB	R/W	8/1	FFH		

注1. スタンバイ・コントロール・レジスタは命令によってセット(1)することはできますが、クリア(0)することはできません (Wは“1”のみ可)。

2. パワーオン・リセット時：00H

3. ユーザ・フラグ・レジスタ (FLAG) のビット 3, 5 以外のビット操作は意味をもちません。また、FLAG レジスタ中のユーザ・フラグ 0, 1 (F0, F1) の内容は、PSWのF0, F1の操作によっても変わります (3.4.12参照)。

- 注4. これらのレジスタのアクセス・サイクルには、1ウエイト・サイクルが挿入されます。
5. これらのレジスタのアクセス・サイクルには、最大6ウエイト・サイクルが挿入されます。
6. パワーオン・リセット時：不定

3.5.4 内蔵RAM領域

××E00H-××EFFH (××はIDBレジスタの内容) に、256バイトのRAMが内蔵されています。

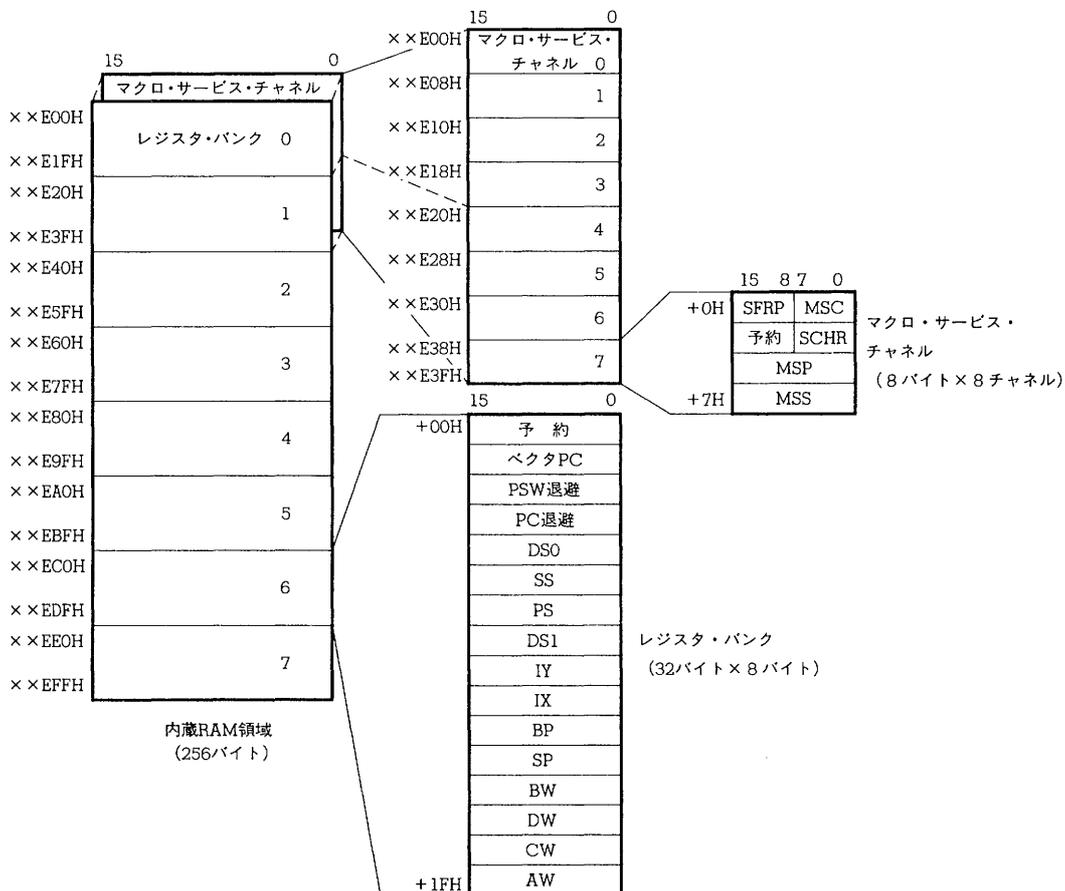
また、内蔵RAMには8バンクのレジスタ・バンクが割り付けられています。さらに、マクロ・サービス・チャンネルのレジスタも重複して割り付けてあります。

内蔵RAMのアクセス中、アドレス・バス、データ・バスには、現在アクセスしているアドレス、データがそのまま出力されます。R/ \overline{W} 信号は前の状態を保持します。 \overline{MREQ} 信号、 \overline{MSTB} 信号はインアクティブ状態となります。

プロセッサ・コントロール・レジスタ (PRC) のビット6 (RAMEN) をリセット (0) することにより、内蔵RAMがメモリ・アクセスの対象となることを禁止できます。また、内蔵RAMからは、プログラム・フェッチを行うことはできません。さらに、メモリ・アクセスを禁止した場合は、レジスタとしてのみアクセスされます。

内蔵RAMに対してアクセスする場合のクロック数は、ノー・ウエイトのメモリ・アクセスと同じ2クロックとなります。★

図 3-6 内蔵RAM領域のマップ



3.5.5 ベクタ・テーブル領域

00000H-003FFHの1Kバイトの領域は、ベクタ割り込みの256ベクタ分(1ベクタ当り4バイトを使用)が割り付けてあります。ベクタ・テーブルには、割り込み要求、ブレーク命令などによる該当割り込みルーチン開始アドレスを設定します。

ベクタ 0	(00000H)	: ディバイド・エラー
ベクタ 1	(00004H)	: シングルステップ
ベクタ 2	(00008H)	: NMI入力
ベクタ 3	(0000CH)	: BRK 3 命令
ベクタ 4	(00010H)	: BRKV命令
ベクタ 5	(00014H)	: CHKIND命令
ベクタ 6	(00018H)	: 予約
ベクタ 7	(0001CH)	: FPO命令
ベクタ 8	(00020H)	: 予約
ベクタ 11	(0002CH)	: 予約
ベクタ 12	(00030H)	: INTSERO
ベクタ 13	(00034H)	: INTSRO
ベクタ 14	(00038H)	: INTSTO
ベクタ 15	(0003CH)	: 予約
ベクタ 16	(00040H)	: INTSER1
ベクタ 17	(00044H)	: INTSR1
ベクタ 18	(00048H)	: INTST1
ベクタ 19	(0004CH)	: 入出力命令
ベクタ 20	(00050H)	: INTDO
ベクタ 21	(00054H)	: INTD1
ベクタ 22	(00058H)	: 予約
ベクタ 23	(0005CH)	: 予約
ベクタ 24	(00060H)	: INTPO
ベクタ 25	(00064H)	: INTP1
ベクタ 26	(00068H)	: INTP2
ベクタ 27	(0006CH)	: 予約
ベクタ 28	(00070H)	: INTTU0
ベクタ 29	(00074H)	: INTTU1
ベクタ 30	(00078H)	: INTTU2
ベクタ 31	(0007CH)	: INTTB
ベクタ 32	(00080H)	: } ユーザ領域
		} ・BRK imm8 命令
ベクタ 255	(003FCH)	: } ・INT入力

ベクタ0-31は使用要因が指定されており（一部予約領域）、一般用途には使用できません。

ベクタ32-255は一般用途として、2バイト・ブレーク命令、INT入力の2種類で使用可能です。使用しない部分についてはベクタ以外の用途にも使用できます。

ベクタは4バイトで構成され、割り込み受け付け時に、上位2バイトがプログラム・セグメントPSへ、下位2バイトがプログラム・カウンタへロードされます。

例 ベクタ 0	000H	001H	PC ← (001H, 000H)
	002H	003H	PS ← (003H, 002H)

3.5.6 外部メモリ領域

μ PD70325, 70335は、00000H-FFFFEHの領域に外部メモリ（ROM, RAM等）を接続します。ただし、FFFO0H-FFFEFHとFFFFCH-FFFFEHの領域は予約されています。

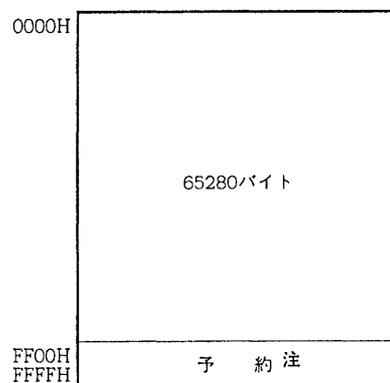
外部メモリをアクセスするには、アドレス・バス（A0-A19）、データ・バス（D0-D15）、そして $\overline{\text{MREQ}}$, $\overline{\text{MSTB}}$, $\text{R}/\overline{\text{W}}$, $\overline{\text{UBE}}$ 信号を用います。また、疑似SRAMのリフレッシュ用にリフレッシュ・パルス出力端子（ $\overline{\text{REFRQ}}$ ）を備えているので、疑似SRAMを容易に接続できます。DRAMのリフレッシュ用にリフレッシュ・アドレスを自動的に出力する機能もあるので、DRAMも容易に接続できます（5.3参照）。さらに、ソフトウェアによってメモリ・サイクル中に128 Kバイト単位でウェイト・サイクルを挿入することができます（5.1参照）。

3.6 I/O空間

μ PD70325, 70335は、1 Mバイトのメモリ空間の他に64 KバイトのI/O空間を持っています。図3-7にI/O空間のマップを示します。

I/O空間をアクセスするには、アドレス・バス (A0-A15)、データ・バス (D0-D7)、そして $\overline{\text{IOSTB}}$, $\text{R}/\overline{\text{W}}$, $\overline{\text{DMAAK0}}$, $\overline{\text{DMAAK1}}$ 信号を用います。使用しないアドレス・バスの上位4ビット (A16-A19) からは、0が出力されます。また、ソフトウェアによってI/Oサイクル中にウエイト・サイクルを挿入することができます (5.1参照)。

図3-7 I/Oマップ (64 Kバイト)



注 予約領域をアクセスするときも、アドレス、データおよび制御信号を出力します。

第4章 割り込み機能

4.1 割り込みコントローラ

μ PD70325, 70335は、17要因の多重割り込みを制御できる割り込みコントローラを内蔵しています。この割り込みコントローラは、外部5、内部12の合計17要因の割り込みをグループに分けて管理し、グループ単位でプログラマブルな多重処理制御を行うことができます。さらに、割り込み要因の性質に応じて、ベクタ割り込み機能、レジスタ・バンク切り替え機能、マクロ・サービス機能の3種の割り込み処理機能を選択できます。

また、 μ PD71059または相当品の割り込みコントローラを外部接続することにより、外部割り込み要因数を容易に拡張することができます。

割り込みコントローラを設定するには、割り込み要求制御レジスタとマクロ・サービス制御レジスタを用います。これらは、各割り込み要因ごとに用意されています。

割り込み制御命令を次に示します。

DI命令 : 割り込みを禁止します。

EI命令 : 割り込みを許可します。

RETI命令 : 割り込みからの復帰を指示します。

RETRBI命令 : BRKCS命令からの復帰を指示します。

FINT命令 : 内蔵割り込みコントローラに割り込み処理の終了を知らせます。

このほか、多重割り込みの受け付け状態を知るために、割り込み要因レジスタと割り込みプライオリティ・レジスタがあります。

4.2 割り込み要因

μ PD70325, 70335の割り込み要因は、外部5、内部12の合計17あります。17要因の割り込みは、8グループに分けて割り込みコントローラで管理されます。このグループの構成はハードウェアで固定されています。8グループの割り込みのうち、NMIとINT、INTTBを除く5グループの割り込みについては、0-7（0が最高位）の優先順位をソフトウェアで任意に設定できます。また、割り込みコントローラがサポートしている機能は、割り込み要因によって異なります。表4-1に割り込み要因の一覧を示します。

表 4-1 割り込み要因の一覧表 (1/2)

グループ	割り込み要因	外部/内部	ベクタ	マクロ・サービス	バンク切り替え	優先順位 ^注			多重処理制御
						設定	グループ間	グループ内	
NMI	NMI (Non Maskable Interrupt)	外部	2	なし	なし	不可	0	—	受けない
外部INT	INT (INTerrupt)	外部	外部入力	なし	なし	不可	7	—	受けない
タイマ割り込み	INTTU0 (INTerrupt from Timer Unit0)	内部	28	あり	あり	可	1	1	受ける
	INTTU1 (INTerrupt from Timer Unit1)		29					2	
	INTTU2 (INTerrupt from Timer Unit2)		30					3	
DMA	INTD0 (INTerrupt from DMA channel0)	内部	20	なし	あり	可	2	1	受ける
	INTD1 (INTerrupt from DMA channel1)		21					2	
外部	INTPO (INTerrupt from Peripheral #0)	外部	24	あり	あり	可	3	1	受ける
	INTP1 (INTerrupt from Peripheral #1)		25					2	
	INTP2 (INTerrupt from Peripheral #2)		26					3	

注 グループ間優先順位，グループ内優先順位は，同じ優先順位に設定された割り込み要求が同時に発生した場合に受け付けられる順番を示します。

表 4-1 割り込み要因の一覧表 (2/2)

グループ	割り込み要因	外部/内部	ベクタ	マクロ・サービス	バンク切り替え	優先順位 ^注			多重処理制御
						設定	グループ間	グループ内	
シリアル・インタフェース・チャンネル0	INTSERO (INTerrupt from Serial Error of channel0)	内部	12	なし	あり	可	4	1	受ける
	INTSRO (INTerrupt from Serial Receiver of channel0)		13	あり				2	
	INTSTO (INTerrupt from Serial Transmitter of channel0)		14	あり				3	
シリアル・インタフェース・チャンネル1	INTSER1 (INTerrupt from Serial Error of channel1)	内部	16	なし	あり	可	5	1	受ける
	INTSR1 (INTerrupt from Serial Receiver of channel1)		17	あり				2	
	INTST1 (INTerrupt from Serial Transmitter of channel1)		18	あり				3	
タイム・ベース	INTTB (INTerrupt from Time Base counter)	内部	31	なし	なし	不可 (7固定)	6	—	受ける

注 グループ間優先順位，グループ内優先順位は，同じ優先順位に設定された割り込み要求が同時に発生した場合に受け付けられる順番を示します。

4.3 優先順位制御

4.3.1 多重割り込みの優先順位制御

多重割り込みの優先順位制御は、NMIとINTを除いた割り込みに対してグループ単位で行われます。

割り込み多重処理の制御はEI状態のときに行われます。したがって、多重処理を行うときは割り込み処理ルーチン中でEI状態にしてください。ただし、マクロ・サービスによる割り込み応答は、DI状態でも多重処理制御を受けます。

(1) 多重処理制御

多重処理制御では、現在処理中の割り込みより優先順位の高い割り込み要求を受け付けて、現在処理中の割り込みを中断し、優先順位の高い割り込みの処理を行います。また、現在処理中の割り込みの優先順位以下であれば、その割り込みは保留されます。保留された割り込みは、現在実行中の割り込み処理ルーチン中で割り込み制御レジスタ（各割り込み要因ごとに用意されている）の中の割り込みマスク・ビットをセットせず、なおかつ割り込み要求フラグをリセットしなければ、現在実行中の割り込みが終了した時点で受け付けられます。

同じ優先順位の割り込み要因や同一グループ内の割り込み要因での多重割り込みはできません。

(2) NMIとINTおよびソフトウェア割り込みを除く割り込み応答

NMIとINTおよびソフトウェア割り込みを除く割り込み応答では、割り込み処理ルーチンの最後で割り込み処理ルーチンが終了したことを割り込みコントローラに示すためにFINT命令を実行しなければいけません。この命令を実行しないと、それ以後の割り込みは、FINT命令を実行しなかった割り込みより優先順位が高くないと受け付けられなくなります（4.10 割り込みプライオリティ・レジスタ (ISPR) 参照）。

多重処理制御を受ける割り込みの処理形態を図4-1に示します。

(3) NMIとINTによる割り込み応答

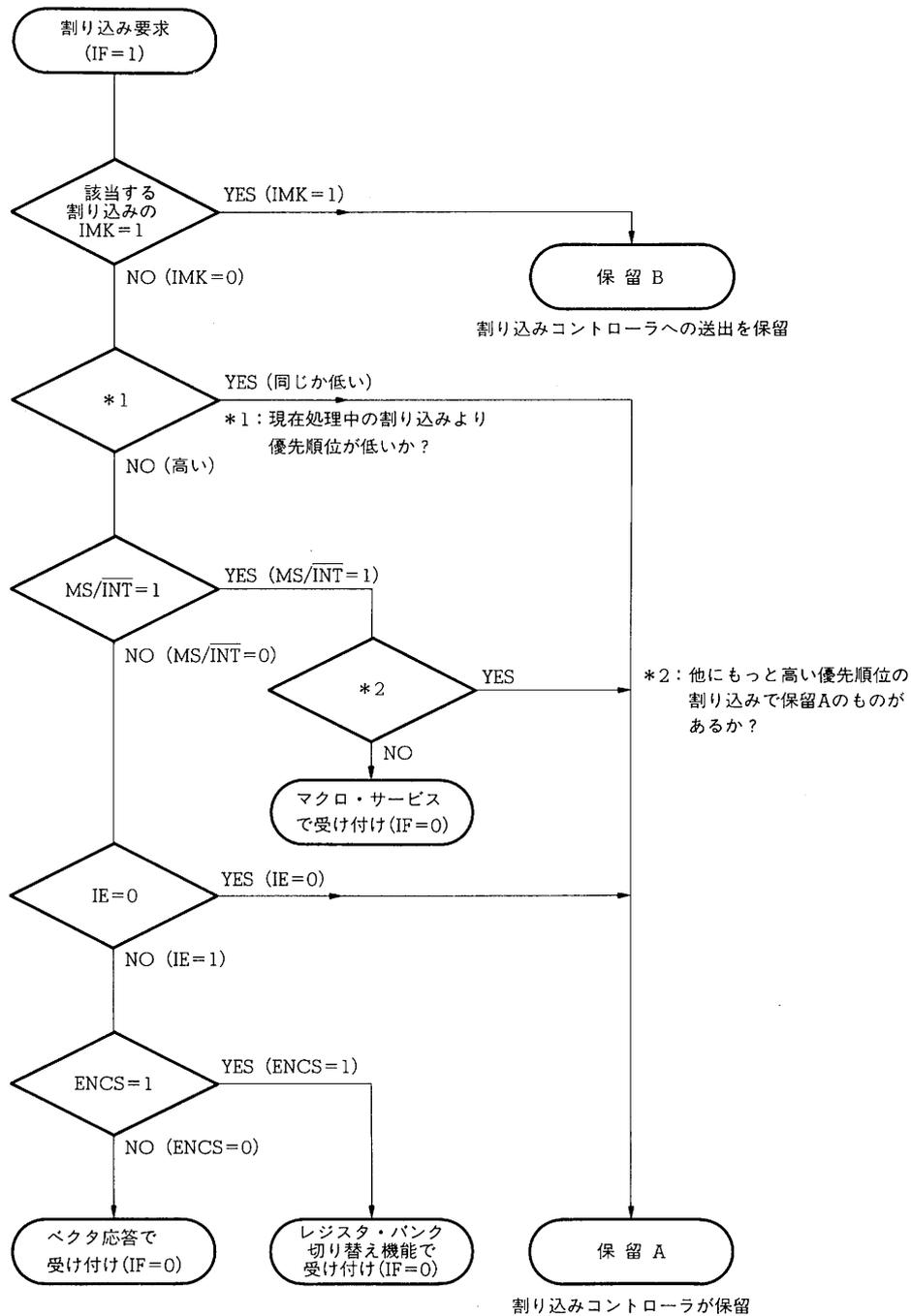
NMIとINTによる割り込み応答は、多重処理制御を受けないので、許可状態にあれば(NMIは常時)受け付けられます。

(4) 優先順位の設定

優先順位は、各割り込みグループごとに0-7まで(0が最優先)の8レベルが任意に設定可能です。この優先順位は、後述するレジスタ・バンク切り替え機能を利用した場合の切り替え先のレジスタ・バンクの番号を同時に示します。優先順位の設定は、各割り込み要因ごとに用意されている割り込み制御レジスタのPRO-2の3ビットで行います(設定した優先順位は、PRO-2を書き換えないかぎり変化しません)。ただし、この設定は、割り込みグループの中で最優先の優先順位を持っている割り込み要因の割り込み制御レジスタでのみ有効で、その他の割り込み制御レジスタでは書き込んでも無視され、読み出し時には7固定となっています。また、リセットにより、優先順位はすべ

てレベル7に初期化されます。**A.4** 割り込みの優先順位と処理順序に関する注意事項を参照してください。

図 4-1 多重処理制御を受ける割り込みの処理形態



4.3.2 同時に割り込みが発生したときの優先順位制御

割り込みが同時に発生した場合の受け付けの優先順位は、NMIが最高で、INTが最低です。NMIとINT以外の優先順位は、多重割り込みの優先順位と同じになります。同一順位に指定されたグループ間では

ハードウェアで固定された優先順位に従います。また、同一グループ内でも同様にグループ内の優先順位に従います。

以下に優先順位制御の例を示します。

- 例1. 優先順位を3と指定したINTSROと優先順位を6と指定したINTTU2が同時に発生した場合、INTSROが先に受け付けられます。
2. INTPOとINTP1が同時に発生した場合、INTPOが先に受け付けられます。
3. NMIとINTD1が同時に発生した場合、NMIが先に受け付けられます。
4. INTTBとINTが同時に発生した場合、INTTBが先に受け付けられます。
5. 優先順位を4と指定したINTTU1およびINTSER1が同時に発生した場合、INTTU1が先に受け付けられます。

★ 4.4 割り込み要求

(1) 割り込み要求の発生

割り込み要求が発生すると、対応する割り込み要求制御レジスタのIF(ビット7)がセット(1)されます。IFがセット(1)されている状態は、割り込み要求のあることを示します。

IFをソフトウェアでセットすることにより、割り込み要求を発生させることもできます。

(2) 割り込み要求の受け付け

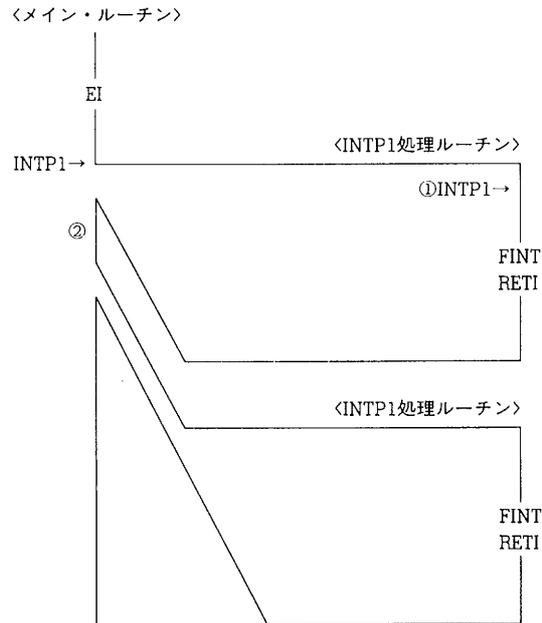
割り込み要求が受け付けられると、対応する割り込み要求制御レジスタのIF(ビット7)がリセット(0)されます。したがって、割り込み処理中に再び同じ割り込み要求が発生した場合、IFがセット(1)され、その割り込みは保留されます。

(3) 割り込み要求の保留

同じ要因の割り込み要求は1つだけ保留できます(図4-2参照)。したがって、IF(ビット7)がセット(1)されている状態で、同じ要因の割り込み要求が発生した場合は、その割り込み要求は無視されます。

割り込み要求はIFがセット(1)されている状態で保留されます。割り込み要求の保留中に、ソフトウェアでIFをリセット(0)すると、その割り込み要求はキャンセルされます。

図4-2 同一割り込みの多重処理



- ① INTP1処理中またはFINT命令実行中に同一割り込みのINTP1が発生するとINTP1は保留します。
ただし、保留できるのは1つだけです。
- ② 最初のINTP1の処理が終了すると、保留されていたINTP1の処理を開始します。

(4) 割り込み要求のポーリング

割り込み要求の発生するタイミングを知りたいが、割り込みを受け付けたくない場合は、割り込み制御レジスタのIMKビットをセット(1)することにより、割り込み要求をマスクし、IFビットをポーリングすることにより、割り込み要求の発生を検出できます。ただし、割り込み要求の発生を検出後、IFビットをソフトウェアでリセット(0)する必要があります。

4.5 割り込み応答方式

μ PD70325, 70335は、ベクタ割り込み機能、レジスタ・バンク切り替え機能、マクロ・サービス機能の3種の割り込み応答方法を持っています。これらの機能は、割り込みの使用目的に合わせて選択することが可能です。割り込みコントローラは、割り込み制御レジスタで設定された応答方法に応じて割り込みの要求に対応します。

ベクタ割り込み機能、レジスタ・バンク切り替え機能で割り込みを受け付けた場合は、その機能に応じた方法でPC, PS, PSWの内容を退避させます。また、PSWを退避させたあと、IE, BRKの各フラグをリセットし、DI状態とします。したがって、NMIとマクロ・サービスによる割り込み応答を除く割り込みとシングルステップ割り込みは禁止されます（シングルステップ割り込みを除くソフトウェア割り込みは発生します）（4.11参照）。

4.5.1 ベクタ割り込み

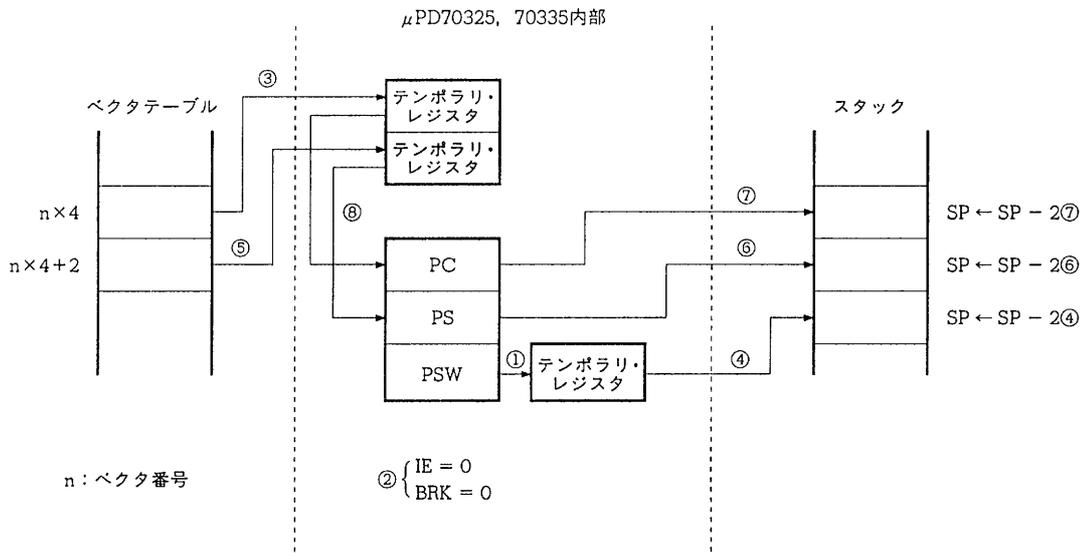
ベクタ割り込みでは割り込みを受け付けると、現在のPSWの内容とPCとPSの内容をスタックに退避させ、ベクタ・テーブルからベクタを1つ選択し、そのベクタで示されるアドレスから割り込み処理ルーチンとして実行します。ベクタは、INTによる割り込み以外はベクタはすべて固定となります。INTによる割り込みの場合、割り込みアクノリッジ・サイクルを発生しデータ・バスから割り込みベクタを取り込みます（4.7 INT参照）。INT以外の割り込みのベクタは、表4-1に示したとおりです。

割り込みからの復帰は、RETI命令で行いますが、NMIとINTを除く割り込みからの復帰の場合は、FINT命令を実行する必要があります。割り込みからの復帰時には、スタックからPC, PS, PSWを復帰させます。

ベクタ割り込みのシーケンスは次のように行われます（図4-3）。

- ① PSWの値をテンポラリ・レジスタに書き込む。
- ② IE, BRKフラグをクリアする。
- ③ ベクタ・アドレスのPCの値を内部のテンポラリ・レジスタに読み込む。
- ④ $SP \leftarrow SP - 2$ として、割り込みが発生したときのPSWの値をスタックに書き込む。
- ⑤ ベクタ・アドレスのPSの値を内部のテンポラリ・レジスタに読み込む。
- ⑥ $SP \leftarrow SP - 2$ として、割り込みが発生したときのPSの値にスタックに書き込む。
- ⑦ $SP \leftarrow SP - 2$ として、割り込みが発生したときのPCの値にスタックに書き込む。
- ⑧ ベクタ・アドレスから読み込んだ値をそれぞれPC, PSに書き込み分岐する。

図4-3 割り込み受け付けの動作



4.5.2 レジスタ・バンク切り替え機能

★

μPD70325, 70335は、内部RAMに汎用のレジスタ・セットをマッピングしており、最大8バンクまでのレジスタ・バンクを持つことができます。このレジスタ・バンクをBRKCS, TSKSW命令の実行、または割り込みの応答時に自動的に切り替えることで、従来ソフトウェアで行っていたレジスタ群のスタックへの退避処理を行わずに済みますので、プログラム実行環境の高速切り替えが可能となっています。

(1) 割り込み要求発生による切り替え

(a) 設定

割り込み応答時にレジスタ・バンク切り替え機能を使用する場合、各割り込み要因ごとに用意されている割り込み制御レジスタのENCSビットをセット(1)します。レジスタ・バンクは、割り込みグループごとに1つ指定することができ、多重割り込みの優先順位と同じ数字になり、割り込み制御レジスタのPRO-2で指定します。レジスタ・バンク番号は、割り込み優先順位と一致します。

レジスタ・バンク切り替え機能を利用する場合は、切り替え先のレジスタ・バンク中のPSとベクタPCはあらかじめイニシャライズしておく必要があります。SS, SPは切り替えの前後いずれかでイニシャライズしてください。切り替え後のイニシャライズにMOVSPA命令を実行すると、SS, SPにはそれぞれ切り替え前の値がセットされ、スタックをレジスタ・バンク切り替え前から連続的に使用できます。その他のレジスタについては、必要に応じてイニシャライズしてください。ただし、PSについては割り込み処理ルーチン中で変更しないようにしてください。

(b) 切り替えシーケンス

レジスタ・バンク切り替えのシーケンスは次のように行われます (図4-4)。

- ① PSWの内容をテンポラリ・レジスタに退避させます。
- ② レジスタ・バンクを切り替えます。
- ③ IE=0, BRK=0にします。
- ④ PCおよびテンポラリ・レジスタに退避させたPSWの内容をレジスタ・バンク中のそれぞれの退避領域へ退避します。
- ⑤ レジスタ・バンク中のベクタPC領域からPCへ割り込み処理ルーチンの開始アドレスのオフセットをロードします。

以上でレジスタ・バンクは切り替えられ、割り込み処理ルーチンを実行します。

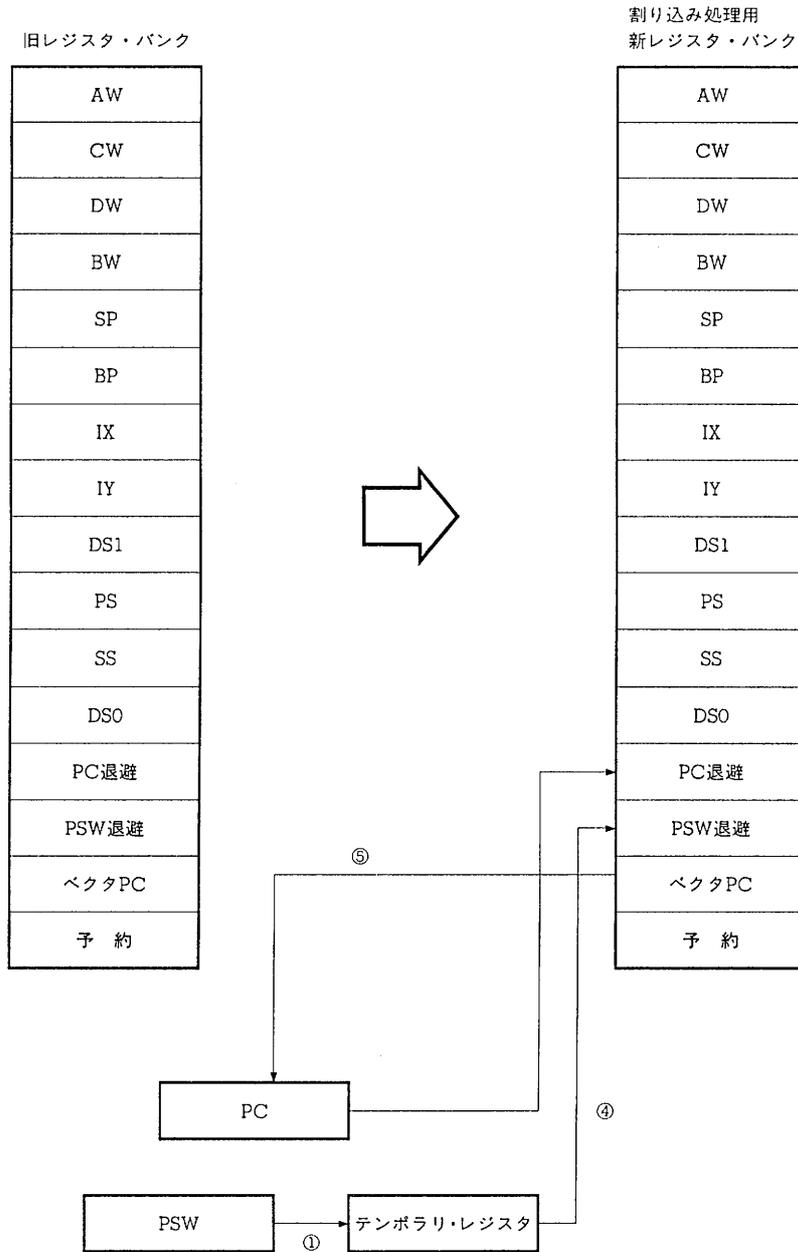
(c) レジスタ・バンク切り替えからの復帰

レジスタ・バンク切り替え割り込みからの復帰は、FINT命令 (レジスタ・バンク切り替え機能が利用できるのは、多重割り込みの制御を受けるものに限られる) 実行後、RETRBI命令を実行することにより行われます。RETRBI命令を実行すると、図4-5に示すようにレジスタ・バンクの中の退避領域からPCとPSWをそれぞれ復帰させます (RETI命令では、レジスタ・バンクの復旧を行いませんので、正常にメイン・ルーチンへ復帰できません)。

なお、レジスタ・バンク切り替え機能は、同一優先順位に設定した割り込みグループ内の1つの割り込みにのみ利用できます (4.8参照)。

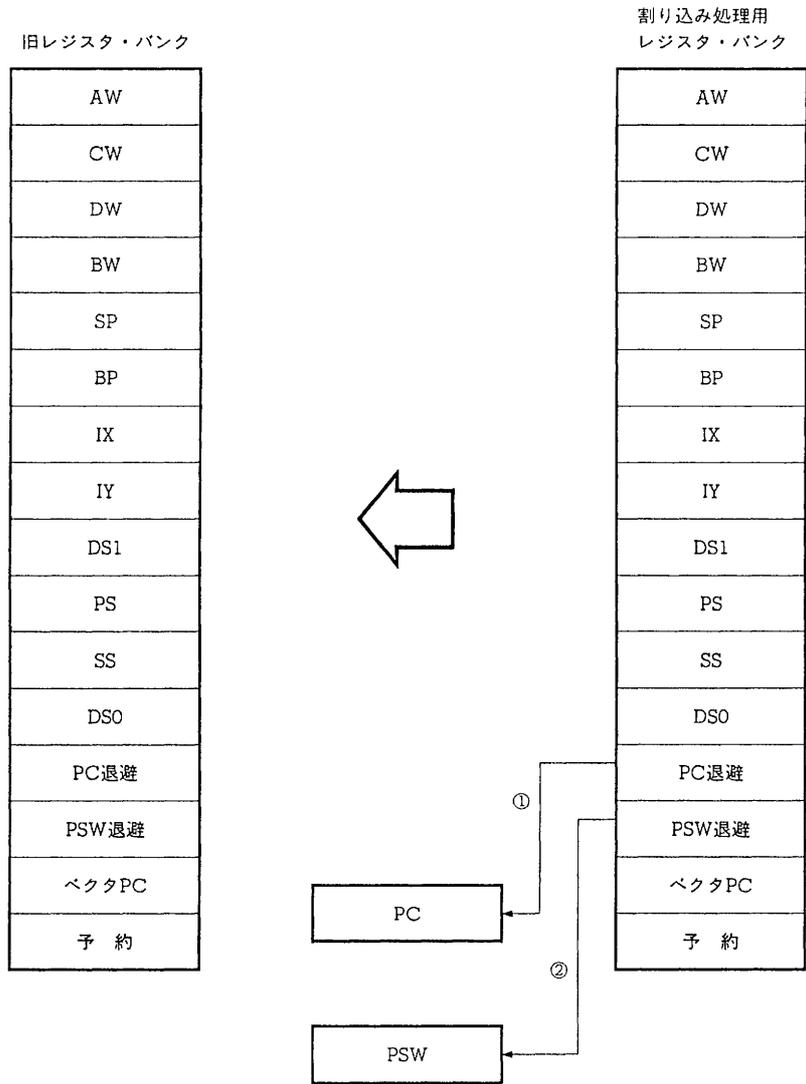
同一優先順位に設定した割り込みグループ内で、複数の割り込みの応答方式をレジスタ・バンク切り替えとした場合、すべて同一のレジスタ・バンクへ切り替わります。

図 4-4 レジスタ・バンク切り替えのシーケンス



②レジスタ・バンク切り替え
③IE=0, BRK=0

図 4-5 レジスタ・バンク復帰のシーケンス



(2) ソフトウェアによる切り替え (BRKCS命令)

BRKCS命令の実行により、レジスタ・バンク切り替えができます。BRKCS命令は高速なサブルーチン・コールとしても利用できます。

切り替え先のレジスタ番号は、まず、オペランドに記述される16ビット・レジスタの下位3ビットでレジスタ・バンクを指定します。次に、PC、PSWの内容を新しいレジスタ・バンク内のPC退避エリア、PSW退避エリアにそれぞれ退避させます。その後、レジスタ・バンク内にあらかじめストアしておいたベクタPCをPCにロードして分岐します。

(a) 設定

BRKCS命令を実行する場合は、選択するレジスタ・バンク内のPSとベクタPCはあらかじめイニシャライズしておく必要があります。SS、SPは切り替えの前後いずれかでイニシャライズしてください。イニシャライズにMOVSPA/MOVSPB命令を実行すると、切り替え先のSS、SPにはそれぞれ切り替え前の値がセットされ、スタックを連続的に使用できます。

新しいレジスタ・バンクからの復帰は、RETRBI命令を実行することにより行われます。その際、FINT命令を実行する必要はありません。

(b) 切り替えおよび復帰のシーケンス

レジスタ・バンク切り替えおよび復帰のシーケンスは、割り込み要求発生によるレジスタ・バンク切り替えと同じです。ただし、復帰の際にFINT命令を実行する必要はありません。RETRBI命令だけ実行すると復帰します。

(3) ソフトウェアによる切り替え (TSKSW命令)

TSKSW命令の実行により、レジスタ・バンク切り替えができます。TSKSW命令により、高速なタスク切り替えができます。

TSKSW命令は、まず、PC、PSWの内容を現在（レジスタ・バンク切り替え前）のレジスタ・バンク内のPC退避エリア、PSW退避エリアにそれぞれ退避させます。次に、オペランドに記述される16ビット・レジスタの下位3ビットでレジスタ・バンクを指定します。その後、新しいレジスタ・バンク内にあらかじめストアしておいたPC退避エリアの内容をPCにロードして分岐します。

(a) 設定

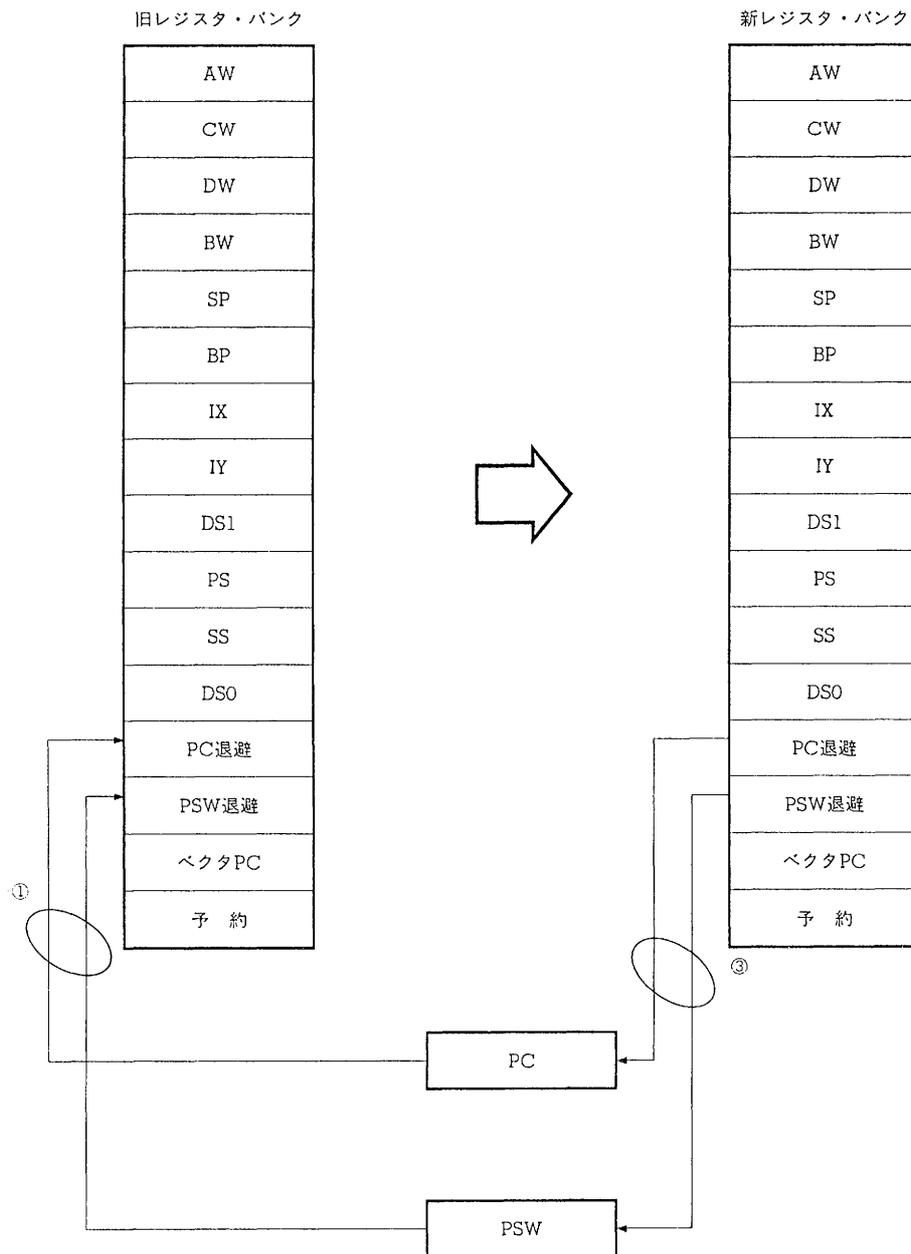
TSKSW命令を実行する場合は、選択するレジスタ・バンク内のPSとPC退避エリア、SS、SP、PSW退避エリアはあらかじめイニシャライズしておく必要があります。SS、SPのイニシャライズにMOVSPB命令を実行すると、切り替え先のSS、SPにはそれぞれ切り替え前の値がセットされ、スタックを連続的に使用できます。

(b) 切り替えシーケンス

レジスタ・バンク切り替えのシーケンスは次のように行われます（図4-6）。

- ① PC, PSWを現在（切り替え前）のレジスタ・バンク内のPC退避エリア, PSW退避エリアに退避します。
- ② レジスタ・バンクを切り替えます。
- ③ 新しいレジスタ・バンク内のPSW退避エリアの値をPSWにロードし, PC退避エリアの値をPCにロードし分岐します。

図 4-6 TSKSW命令によるレジスタ・バンク切り替えのシーケンス



②レジスタ・バンク切り替え

4.5.3 マクロ・サービス機能

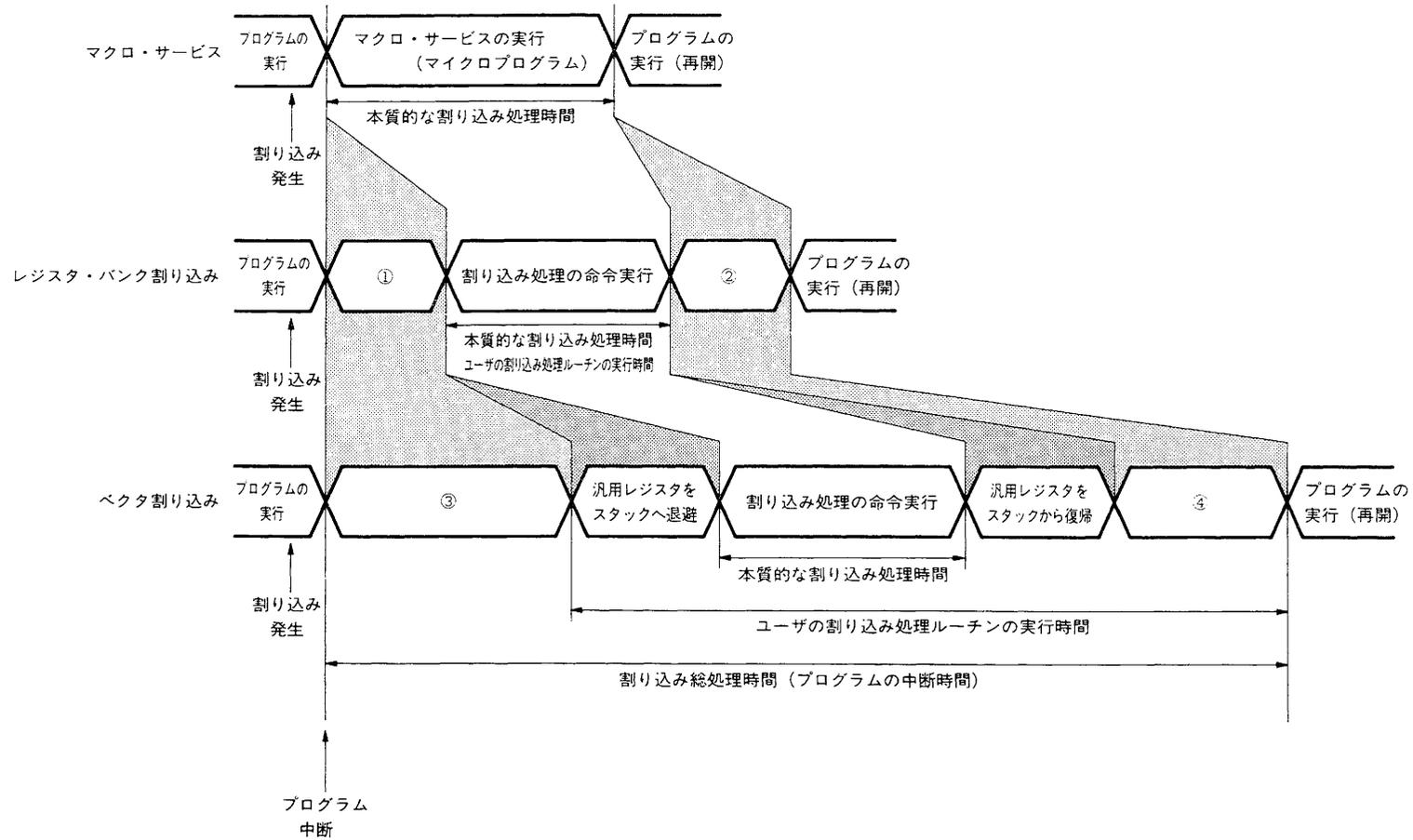
マクロ・サービス機能は、割り込み要求によって特殊機能レジスタ領域と外部メモリ空間の間でデータ転送を1バイトまたは1ワード単位で行う機能です。この機能により、単純なデータ転送などの簡単な処理をソフトウェアによる割り込み処理で行わずに済み、割り込み処理にかかるオーバーヘッド（レジスタの退避、初期化、復帰などの動作）を削減することができます。また、マクロ・サービス機能で行われる処理はソフトウェア的には意識する必要がなく、従来、1バイト単位でソフトウェアによって処理していたものを一固まりのデータとして処理することができ、効率良くプログラミングすることが可能です。

マクロ・サービス機能は他の割り込み応答方式と異なり、各割り込み要因ごとに用意されている割り込み制御レジスタのIMKビット（割り込みマスク・ビット）がリセットされていて、MS/INTビット（マクロ・サービス許可ビット）がセットされていればEI、DIの状態にかかわらず動作します（4.8参照）。ただし、優先順位の制御は受けません。

MSCで指定された回数のマクロ・サービスを終了したとき、またはSFRPの値と転送したデータが一致したときに、MS/INTビットがリセットされ、マクロ・サービス完了割り込みが発生します。このマクロ・サービス完了割り込みは、DI状態では保留されます。

図 4-7 マクロ・サービスによる割り込み処理効率

★



- ① PC, PSWをレジスタ・ファイルへ退避。ベクタ・テーブルからレジスタ番号をリード, 分岐。
- ② RETRBI命令の実行。PC, PSWをレジスタ・ファイルから復帰, 分岐。
- ③ PC, PS, PSWをスタックへ退避。ベクタ・テーブルからPC, PSをリード, 分岐。
- ④ RETI命令の実行。PC, PS, PSWをスタックから復帰, 分岐。

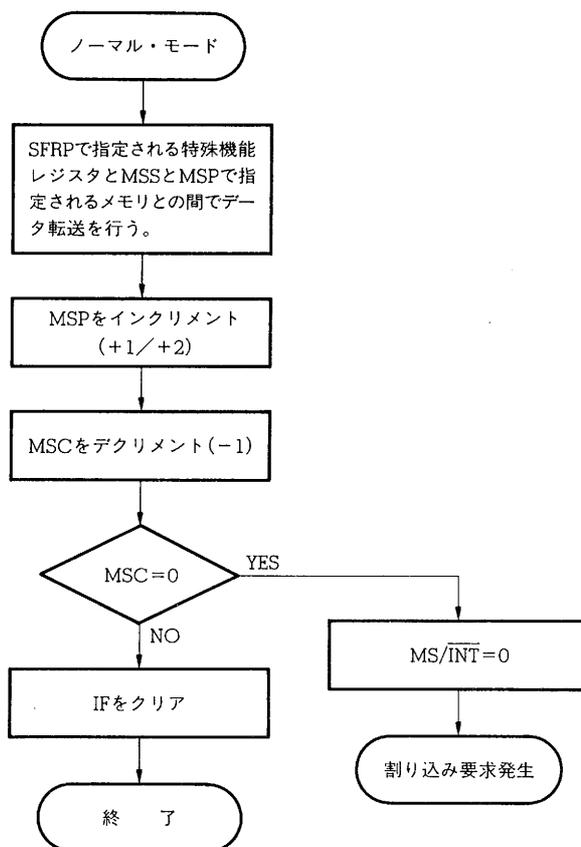
マクロ・サービス機能には次に示す2種類の動作モードがあります。

(1) ノーマル・モード

あらかじめマクロ・サービス・カウンタ (MSC) に設定した回数のデータ転送を、割り込み要求の発生ごとに1バイトまたは1ワードずつ行います。

図4-8にノーマル・モードの動作フローを示します。

図4-8 ノーマル・モードの動作フロー



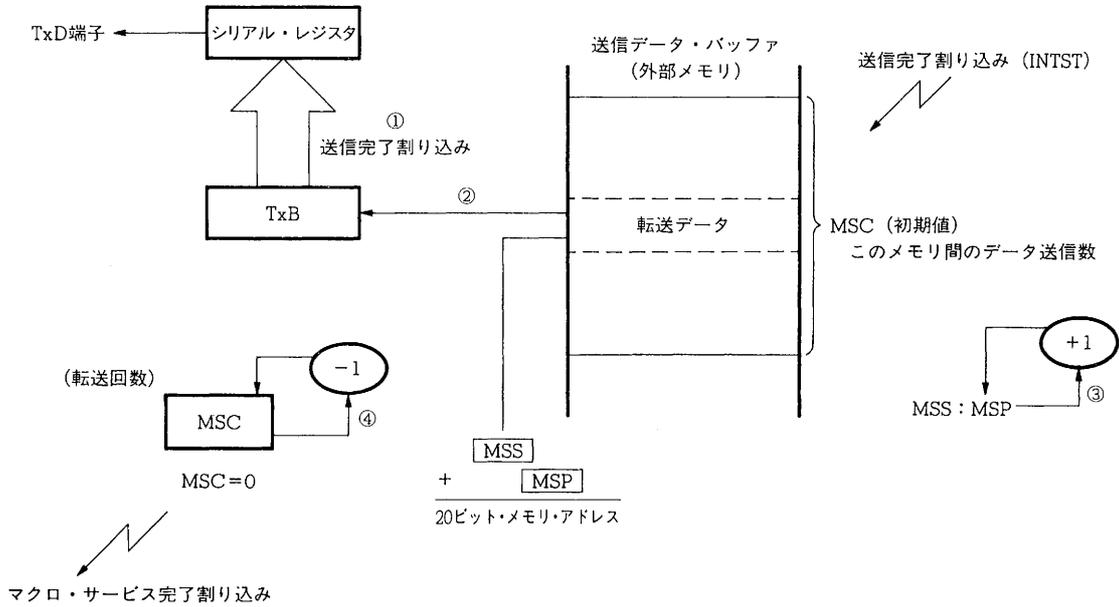
備考 SFRP, MSS, MSP, MSCについては, 4.5.4参照。

MS/INTは, 4.8参照。

シリアル・インタフェースの送信の例を次に示します。

★

図4-9 シリアル・インタフェースの送信例



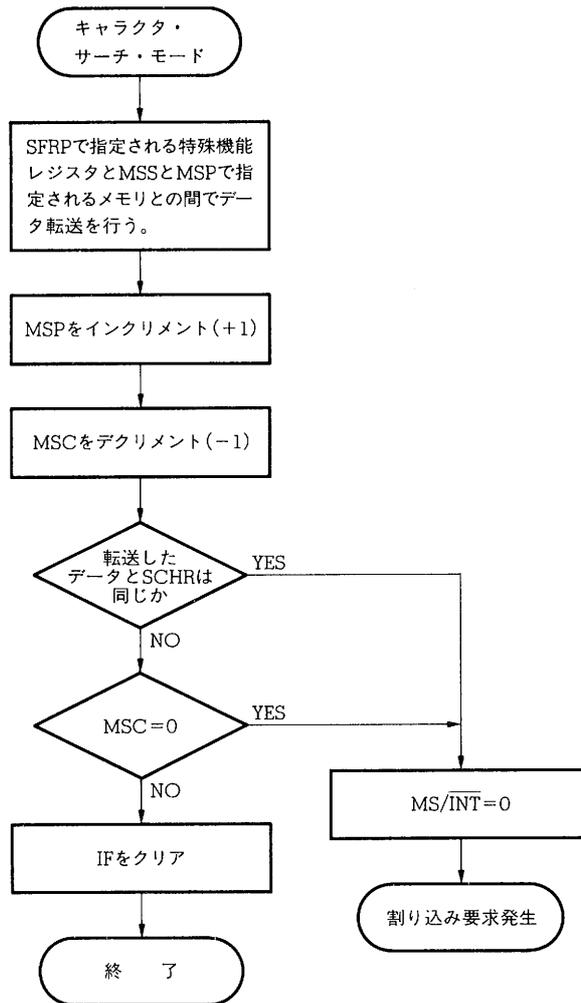
- ① TxBレジスタの内容がシリアル・レジスタへ転送され、送信完了割り込みが発生します。
- ② MSS : MSPで示されるアドレスよりデータをTxBへ転送します。
- ③ MSPをインクリメントします。
- ④ MSCをデクリメントします。MSC=0になるとマクロ・サービス完了割り込みが発生します。

(2) キャラクタ・サーチ・モード

1バイトのデータ転送を割り込み要求の発生ごとに、あらかじめマクロ・サービス・カウンタ(MSC)に設定したバイト数または転送を行ったデータが、あらかじめSCHRキャラクタ・データに設定した8ビット・データと一致するまで行います。

図4-10にキャラクタ・サーチ・モードの動作フローを示します。

図4-10 キャラクタ・サーチ・モードの動作フロー

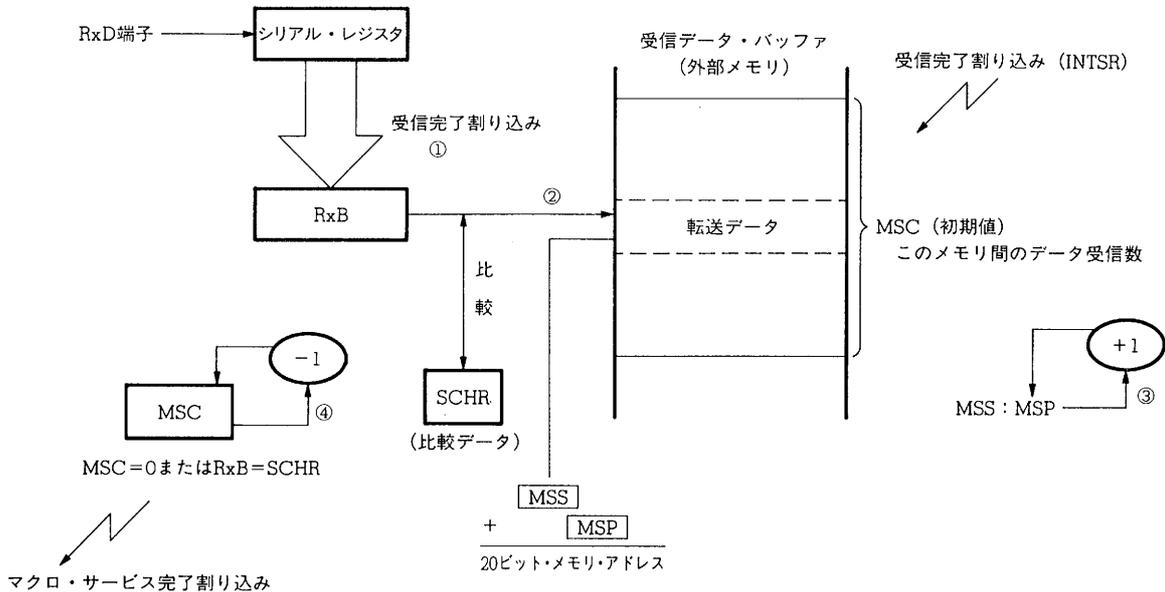


備考 SFRP, MSS, MSP, MSC, SCHRは, 4.5.4参照。

MS/INTは, 4.8参照。

シリアル・インタフェースの受信（終了コード付き）の例を次に示します。

★ 図4-11 シリアル・インタフェースの受信例



- ① シリアル・レジスタの内容がRxBレジスタへ転送され、受信完了割り込みが発生します。
- ② RxBの内容をMSS:MSPで示されるアドレスへ転送します。
- ③ MSPをインクリメントします。
- ④ MSCをデクリメントします。MSC=0またはRxB=SCHRになるとマクロ・サービス完了割り込みが発生します。

マクロ・サービス機能は、マクロ・サービスが可能な割り込み要因ごとに用意されているマクロ・サービス制御レジスタとマクロ・サービス制御レジスタで指定されるマクロ・サービス・チャンネルによって制御されます。

★ (3) マクロ・サービス完了割り込み

MSC=0となったとき、または転送データが比較データと一致したとき(キャラクタ・サーチ・モード時だけ)は、割り込み制御レジスタのMS/INTビットだけをリセット(0)し、IFビットはリセット(0)されません。その結果、EI状態であれば、マクロ・サービスの終了に引き続きENC5ビットで指定される割り込みが発生します。DI状態時にマクロ・サービスが完了したときは、この割り込みは保留されます。

4.5.4 マクロ・サービス制御レジスタ

マクロ・サービス機能を制御する8ビット・レジスタです。マクロ・サービス制御レジスタは下図に示す構成になっており、各ビットの機能を以下に説明します。

7	6	5	4	3	2	1	0
MSM2	MSM1	MSM0	DIR	0	CH2	CH1	CH0

CH0 - **CH2** マクロ・サービス・チャンネルを指定するビット

0-7まで指定できます。

DIR データ転送の方向を指定するビット

0でメモリから特殊機能レジスタに転送し、1で特殊機能レジスタからメモリに転送します。

MSM0 - **MSM2** マクロ・サービス・モードを指定するビット

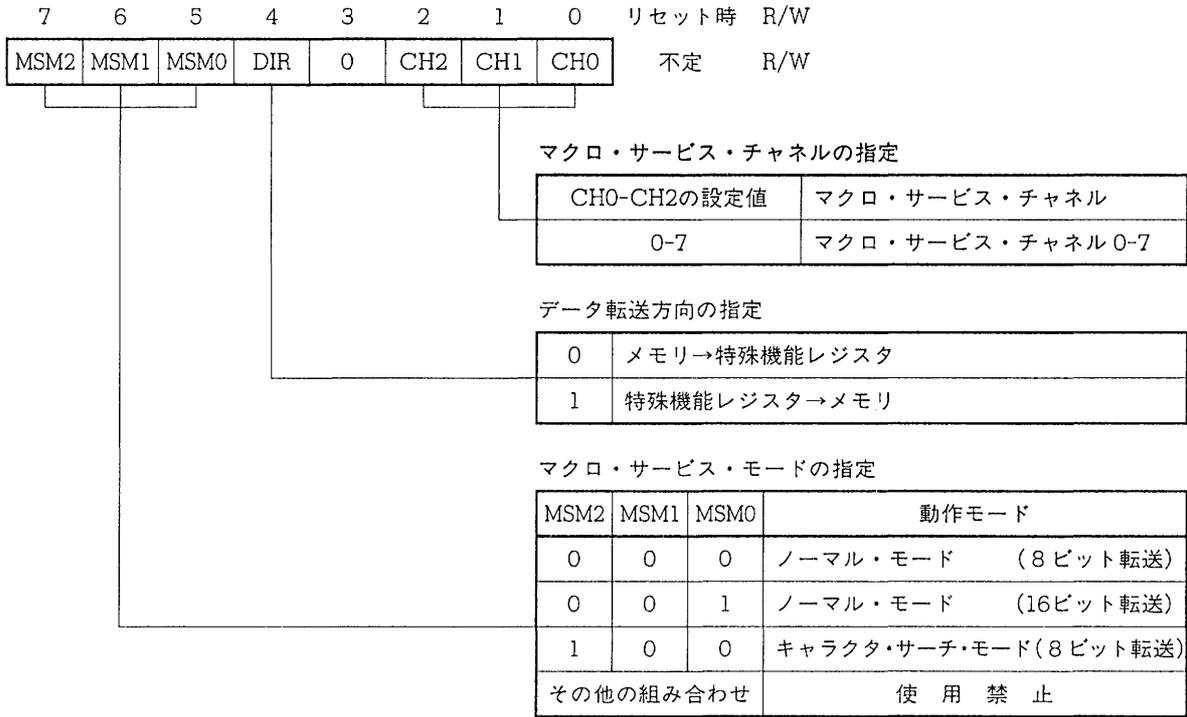
MSM0 - **MSM2** ビットの組み合わせによって、ノーマル・モード、キャラクタ・サーチ・モードの動作モードと、ノーマル・モードの転送データ・ビット数（8ビット/16ビット）を指定します。

MSM2	MSM1	MSM0	動作モード
0	0	0	ノーマル・モード（8ビット転送）
0	0	1	ノーマル・モード（16ビット転送）
1	0	0	キャラクタ・サーチ・モード（8ビット転送）
その他の組み合わせ			使用禁止

マクロ・サービス制御レジスタは、特殊機能レジスタ内にあります。8ビット操作または1ビット操作のメモリ・アクセスによって、このレジスタの書き込み/読み出しができます。

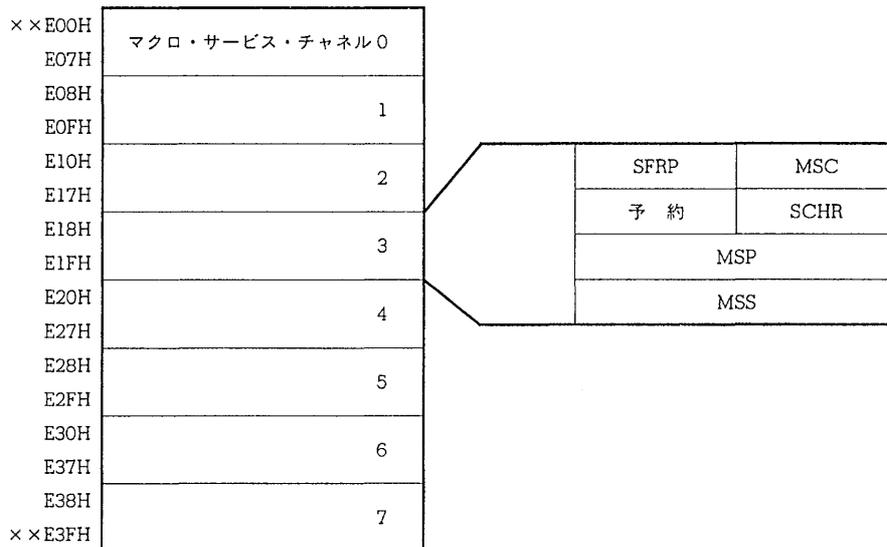
マクロ・サービス制御レジスタは、マクロ・サービスが可能な割り込み要因ごとに用意されています。マクロ・サービスが可能な割り込み要因は、タイマ割り込み（INTTU0-2）、外部割り込み（INTPO-2）、シリアル受信/送信割り込み（INTSRO, 1/INTSTO, 1）です。割り込み要因ごとのマクロ・サービス制御レジスタのロケーションは、該当する項目を参照してください。

図4-12 マクロ・サービス制御レジスタ



マクロ・サービス・チャンネルは、図4-13に示すように内蔵RAMの××E00H-××E3FH(××はIDBレジスタの内容)に割り付けてあります。マクロ・サービス・チャンネルによって、マクロ・サービスのデータの転送先、転送元、転送数、比較キャラクタを設定し、最大8チャンネルの使用が可能です。

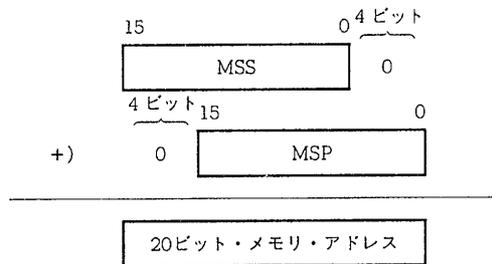
図4-13 マクロ・サービス・チャンネルの構成



- ・MSC (+0H) : マクロ・サービスで行う転送数
- ・SFRP (+1H) : 特殊機能レジスタのアドレスのオフセット値, $\times\times\text{FOOH} + \text{SFRP}$ ($\times\times$ はIDBの内容) が特殊機能レジスタのアドレス
- ・SCHR (+2H) : キャラクタ・サーチ・モード時に比較する8ビット・データ
- ・MSP (+4H) : マクロ・サービスでデータ転送の対象となるメモリ・アドレスのオフセット値
- ・MSS (+6H) : マクロ・サービスでデータ転送の対象となるメモリ・アドレスのセグメント値
データ転送の対象となるメモリ・アドレスは $\text{MSS} \times 16 + \text{MSP}$ となる。

() 内は各マクロ・サービス・チャンネルの開始アドレスからのオフセット。

マクロ・サービスでの転送の対象となるメモリ・アドレスは, MSSで示されるセグメントとMSPで示されるセグメントからのオフセット値で示されます。



マクロ・サービス・チャンネル中のMSC, MSPは1データ転送(8ビット/16ビット)ごとにMSCではデクリメント(-1), MSPではインクリメント(+1/+2)します。この後, 割り込み要求フラグをクリアします。MSCに0を書くと, 256回転送を行います。

注意 レジスタ・バンクとマクロ・サービス・チャンネルは, 同じ内蔵RAMに割り付けてあるために, マクロ・サービスを使用するときはレジスタ・バンクは使用できません。★

4.6 NMI (Non-Maskable Interrupt)

NMIは最優先の禁止することのできない割り込みです。この割り込みはエッジで検出されます。エッジの方向は特殊機能レジスタのINTMレジスタのビット0, ESNMIビットで選択できます。ESNMIビットを0にすると立ち下がりエッジ, 1にすると立ち上がりエッジで割り込みが発生します。この割り込みはベクタ応答しかできず, ベクタ・タイプは2に固定されています。また, この入力はP10端子と兼用されており, P10を読み込むことによりレベルを調べることができます。NMIが受け付けられると割り込み禁止状態 (IE=0) となり, 他の割り込みを禁止します (ただし, マクロサービスによる割り込み応答は, 受け付けられます)。

なお, NMI処理中のNMI要求は受け付けられません。

4.7 INT (Interrupt)

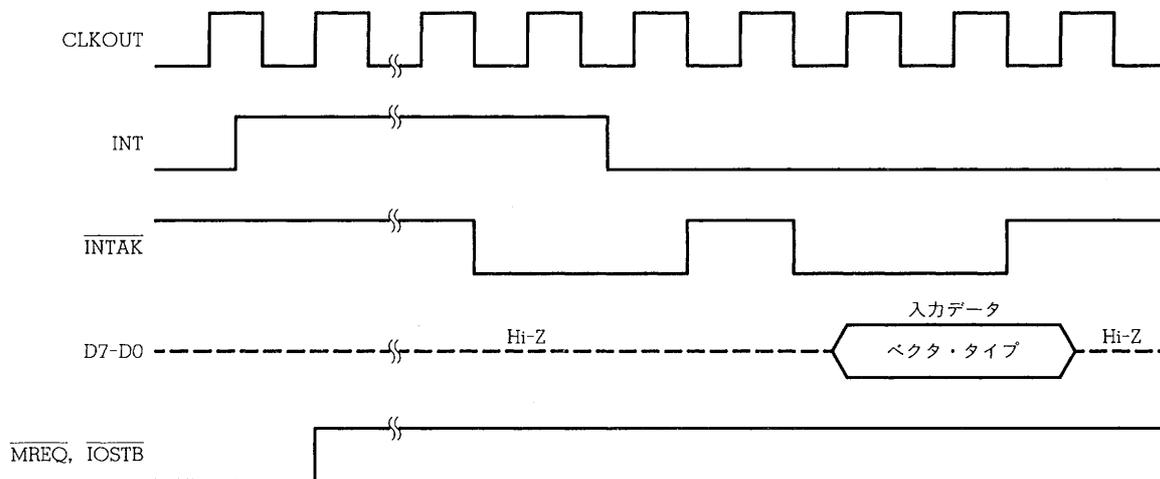
INTはマスク可能な割り込みです。この割り込みはレベル（アクティブ・ハイ）で検出されます。INTは割り込みコントローラによって多重処理の制御を受けず、割り込み許可状態（IE=1）であればいつでも受け付けられます。ただし、複数の割り込みが同時に発生したときの優先順位は最下位となります。また、INTはベクタ応答しかできず、ベクタ・タイプは割り込みアクノリッジ・サイクルでデータ・バスから取り込みます。割り込みアクノリッジ・サイクルは $\overline{\text{INTAK}}$ 出力によって確認できます。INT端子はP14、 $\overline{\text{POLL}}$ と兼用になっており、特殊機能レジスタのポート1モード・コントロール・レジスタ（PMC1）のビット4で選択します。したがって、INTの機能が選択されていないときは割り込み許可状態（IE=1）でも割り込みは発生しません。 $\overline{\text{INTAK}}$ はP13、 $\overline{\text{INTP2}}$ と兼用になっており、PMC1のビット3により機能を選択します（ $\overline{\text{INTAK}}$ 機能の選択を行わないと、割り込みアクノリッジ・サイクルの発生を外部に知らせることができません）。割り込みアクノリッジ・サイクルのタイミングを図4-14に示します。

INT信号は、少なくとも最初の $\overline{\text{INTAK}}$ 信号が出力されるまで、ハイ・レベルを保持してください。

また、 $\mu\text{PD71059}$ または相当品の割り込みコントローラを接続して外部割り込みの入力を最大64まで拡張することができます。

割り込みが受け付けられると、割り込み禁止状態（IE=0）となります。

図4-14 INT割り込みアクノリッジ・タイミング



4.8 割り込み要求制御レジスタ

INT, NMI以外の割り込みを制御する8ビット・レジスタです。

割り込み要求制御レジスタは下図の構成になっています。各ビットの機能を以下に説明します。

7	6	5	4	3	2	1	0
IF	IMK	MS/ $\overline{\text{INT}}$	ENCS	0	PR2	PR1	PRO

PRO - **PR2** 割り込みグループの優先順位を指定するビット

0-7まで指定できます。この指定はグループ内で最高優先順位を持った割り込みの制御レジスタのみ可能で、他の割り込み要求制御レジスタでは指定は無効となります(読み出し時は7固定)。他の割り込み要求制御レジスタの優先順位は、グループ内で最高優先順位を持った割り込み要求制御レジスタの優先順位に従います。

また、この指定はレジスタ・バンク切り替え機能における切り替え先のレジスタ・バンクを指定します。

ENCS レジスタ・バンク切り替え機能を使用するか否かを指定するビット

1でレジスタ・バンク切り替え機能を使用し、0でベクタ割り込みが使用されます。

MS/ $\overline{\text{INT}}$ 割り込み応答方法を選択するビット

1でマクロ・サービス機能を使用し、0でベクタ割り込み、またはレジスタ・バンク切り替え機能を使用します。

IMK 割り込みのマスクを設定するビット

1で割り込みをマスクし、0でマスクを解除します。

IF 割り込み要求のあることを示すビット

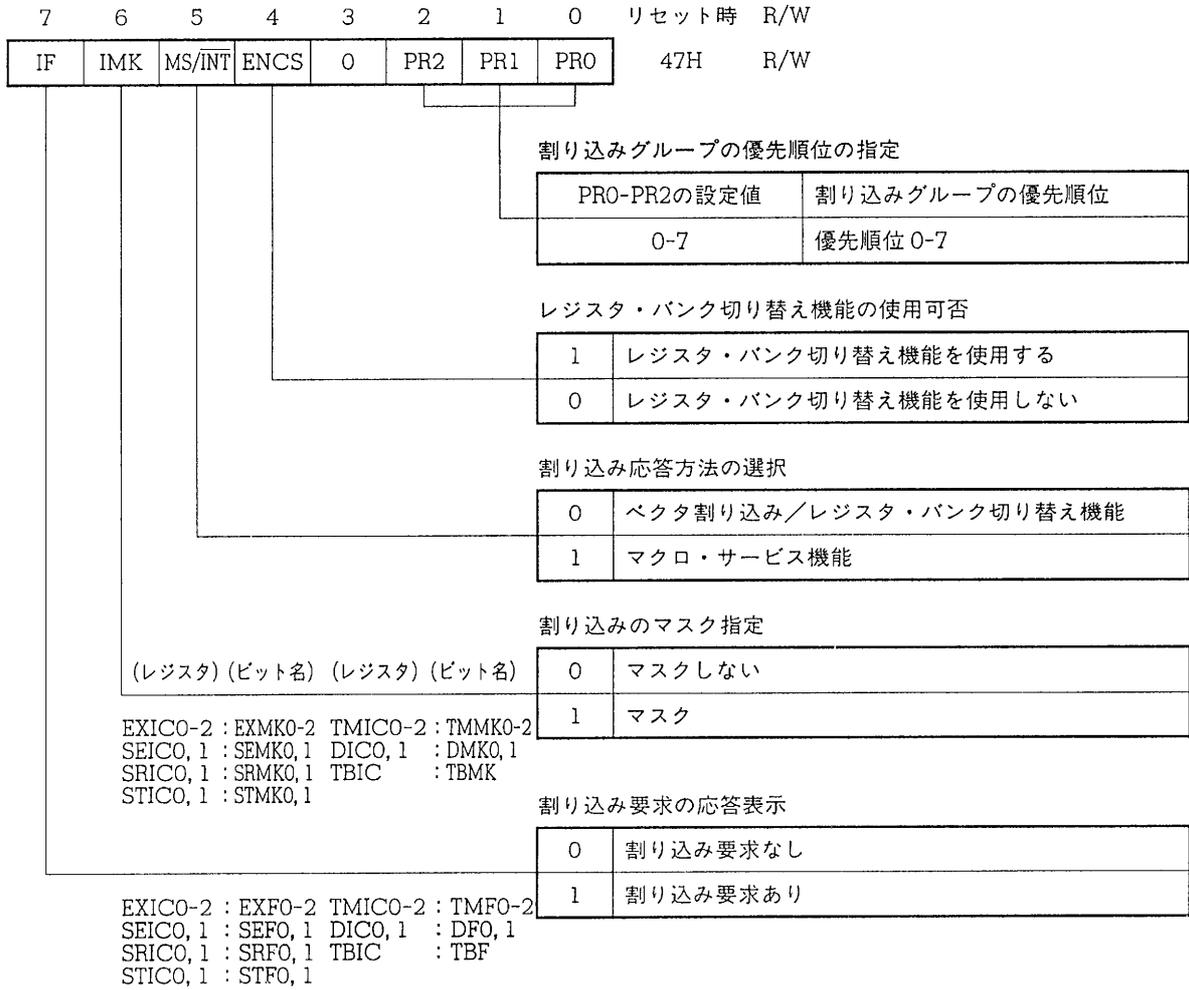
1で割り込み要求のあることを示し、0でないことを示します。このビットは、割り込み要求の発生でセット(1)され、割り込みの受け付け、またはBTCLR命令(μ PD70108, 70116からの追加命令)等の命令によってリセット(0)されます。IMKビットにより割り込みがマスクされていても、割り込み要求が発生すればセット(1)します。

注意 割り込み処理中にプログラムでIFビットを0に書き換えてしまうと、割り込みが発生しなくなります。

割り込み要求制御レジスタは特殊機能レジスタ内にあります。8ビット操作または1ビット操作のメモリ・アクセスによって、このレジスタの書き込み/読み出しができません。

割り込み要求制御レジスタは、割り込み要因(INT, NMI以外)ごとに用意されています。割り込み要因ごとの割り込み要求制御レジスタのロケーションは、該当する項目を参照してください。

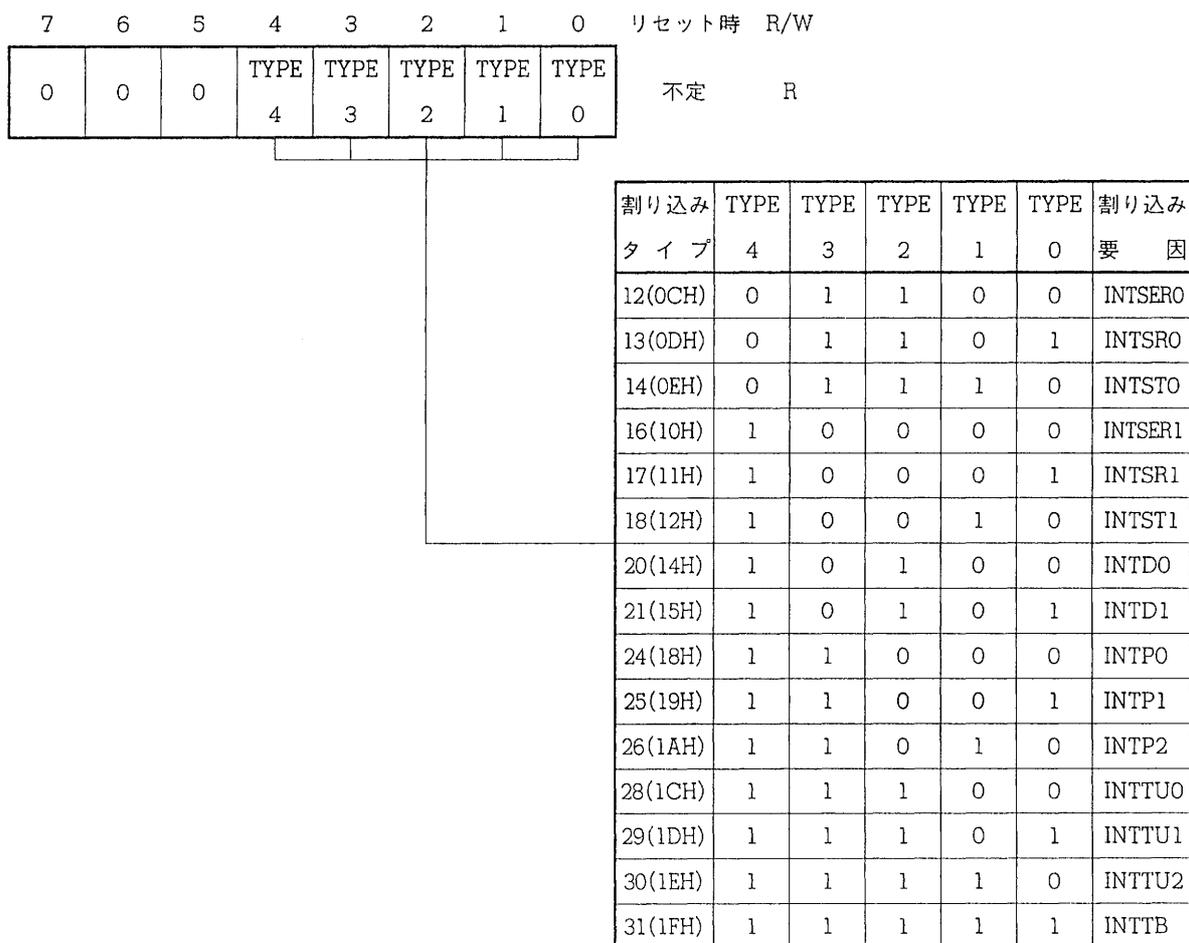
図 4-15 割り込み要求制御レジスタ



4.9 割り込み要因レジスタ (IRQS)

受け付けられた割り込みの割り込み要因を識別するための割り込みタイプを格納する8ビット・レジスタです。IRQSレジスタは図4-16のような構成となっており、上位3ビットは固定です。

図4-16 IRQS



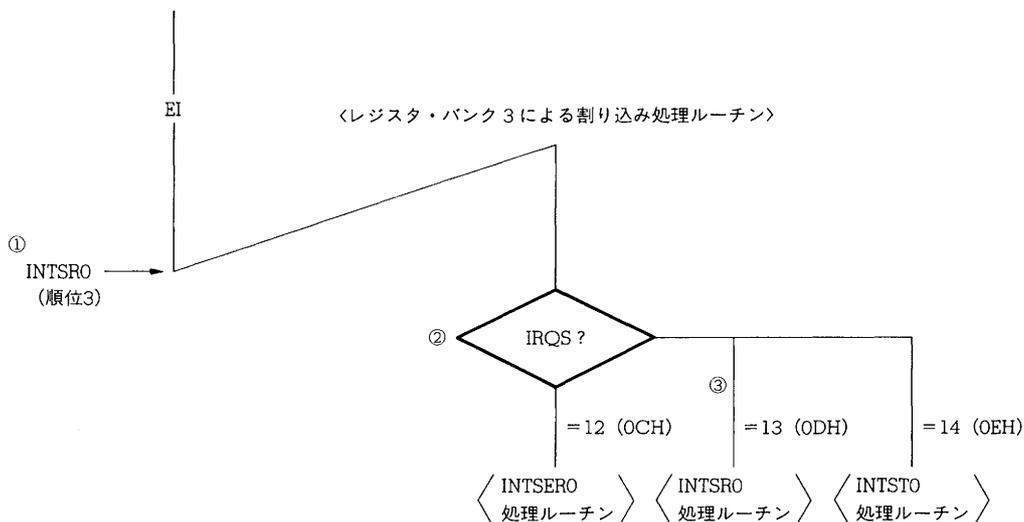
割り込みタイプは優先順位制御を受ける割り込み要因 (NMI, INTを除く) ごとに決まっており、ベクタ番号と同じ値となっています。

IRQSレジスタに格納される割り込みタイプは、割り込みタイプをもつ割り込み要因 (図4-16参照) からの割り込み要求 (マクロ・サービスによる応答時を除く) を受け付けることにより更新され、次の割り込み要求が受け付けられるまで保持されます。これ以外の割り込み (NMI, INTなど) によっては変化しません。したがって、レジスタ・バンク切り替えによる割り込み処理において、IRQSレジスタの読み出しにより割り込み要求ソースを判別する場合、EI命令等により次の割り込み要求が受け付けられるようになる前に、このレジスタの読み出しを行ってください。

IRQSレジスタは8ビット操作のメモリ・アクセスによって読み出しが可能です。

次に、IRQSを利用したレジスタ・バンク切り替えによる割り込み処理の例を示します。

〈メイン・ルーチン〉

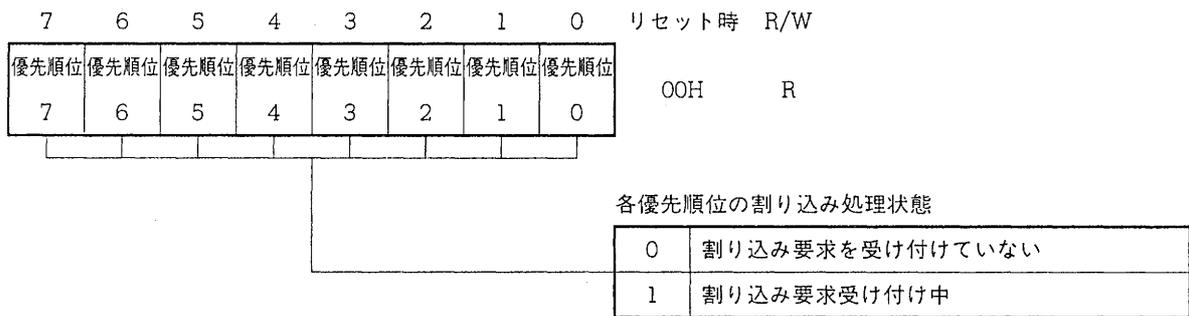


- ① 割り込み許可状態 (EI) で優先順位 3 に設定したINTSROが発生し、レジスタ・バンク切り替えにより、割り込み処理 (レジスタ・バンク 3 を使用) を開始する。
- ② この処理開始直後のDI状態で、IRQSを読み出す (他の割り込みによりIRQSの値が変化しないように、必ずDI状態のままでIRQSの読み出しを行う)。
- ③ 読み出した値 (この場合、13となる) に従って、INTSROの処理ルーチンを選択し実行する。

4.10 割り込みプライオリティ・レジスタ (ISPR)

割り込みコントローラの優先順位制御による多重割り込みの処理状態を示す8ビット・レジスタです。ISPRレジスタへの書き込みはできません。

図4-17 ISPR



ビット0-7はそれぞれ優先順位0-7に対応し、その優先順位の割り込み要求の受け付けによってセット(1)されます。

1回のFINT命令の実行により、セットされているビットのうち最下位(最高優先順位)の1ビットをリセット(0)します。

ビット0-7のいずれかのビットがセットされている場合、そのビットに対応する優先順位以下の割り込み要求は受け付けられず保留されます。

また、そのビットに対応する優先順位より高い要求は受け付けられ、今まで処理していた割り込みは、中断して保留されます。

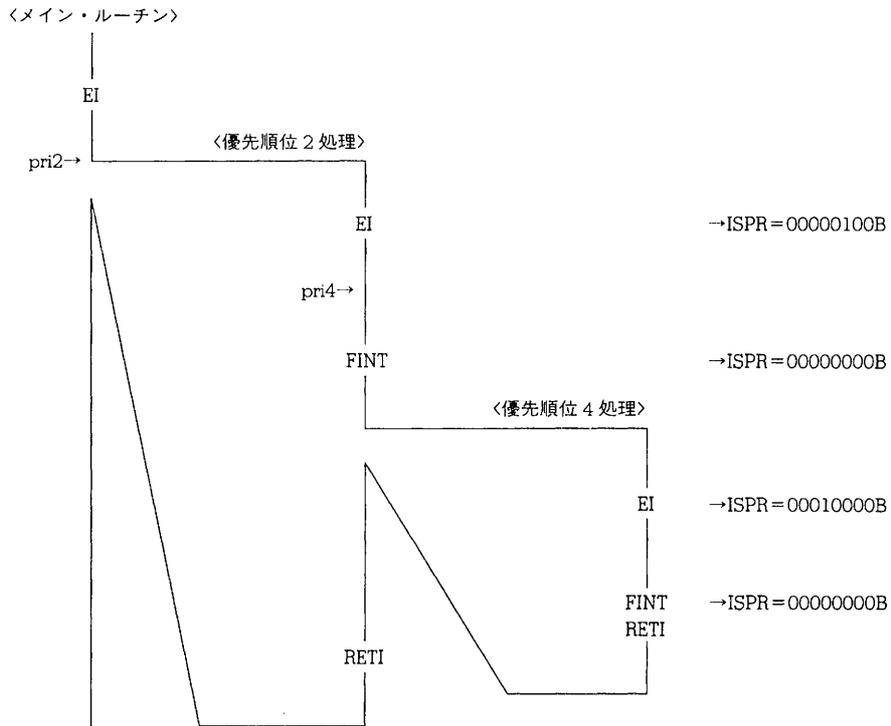
ISPRレジスタは特殊機能レジスタ内にあり、8ビット操作のメモリ・アクセスによって読み出しができません。

$\overline{\text{RESET}}$ 入力時、ISPRの内容はOOHに初期化されます。

- ★ 注意1. FINT命令を実行しないと、その処理を終了しても該当するビットがリセット(0)しないため、その優先順位以下の割り込みを受け付けることができません。
- ★ 2. FINT命令をRETI命令またはRETRBI命令の直前以外に使用した場合、優先順位の管理ができなくなります。その場合はアプリケーション上で管理してください。

図 4-18 ISPRの状態

★



4.1.1 外部割り込み

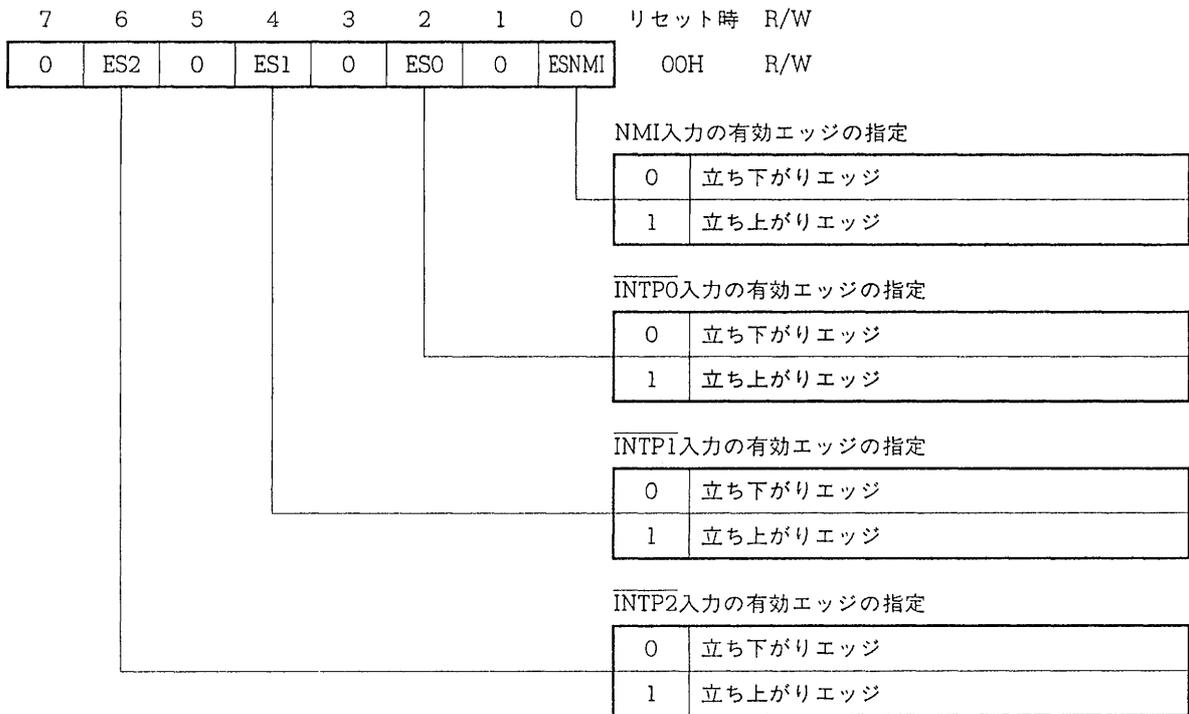
外部からの割り込み要求として5要因あります。このうちINTはレベルで検出され、INT以外 (INTP0-2, NMI)はエッジで検出されます。INT以外のエッジで検出する割り込みは、それぞれ有効エッジを特殊機能レジスタの外部割り込みモード・レジスタ (INTM) によって指定することが可能です。

4.1.1.1 外部割り込みモード・レジスタ (INTM)

外部割り込み要求の有効エッジを指定する8ビット・レジスタです。エッジで検出する割り込みは、NMI, INTP0-2があります。これらの割り込みの有効エッジが、INTMレジスタで指定されます。

INTMレジスタは、図4-19のような構成と機能を持っています。

図4-19 INTM



INTMレジスタは特殊機能レジスタ内にあります。8ビット操作または1ビット操作のメモリ・アクセスによって、このレジスタの書き込み／読み出しができます。

$\overline{\text{RESET}}$ 入力時、INTMレジスタの内容はOOHに初期化されます。

4.11.2 外部割り込み要求制御レジスタ (EXICO, EXIC1, EXIC2)

EXIC_nレジスタ (n=0-2) は、3つの外部割り込み要求端子 ($\overline{\text{INTP0}}-\overline{\text{INTP2}}$) から発生する割り込み要求 (EXF0-EXF2) を制御する8ビット・レジスタです。

この3つの割り込み要求が外部割り込み要求として1つのグループを構成し、優先順位がプログラマブルです。グループ内では、以下のように優先順位が固定されています。

EXF0 > EXF1 > EXF2

図 4-20 EXICO, EXIC1, EXIC2

	7	6	5	4	3	2	1	0
EXICO	EXF0	EXMK0	MS/ $\overline{\text{INT}}$	ENCS	0	PR2	PR1	PRO
EXIC1	EXF1	EXMK1	MS/ $\overline{\text{INT}}$	ENCS	0	1	1	1
EXIC2	EXF2	EXMK2	MS/ $\overline{\text{INT}}$	ENCS	0	1	1	1

注意 EXIC1, EXIC2レジスタのビット2-0は“1”に固定されます。

EXIC1, EXIC2レジスタの割り込み要求の優先順位は、EXICOレジスタのPR2-PROビットの設定に従います。

EXIC_nレジスタの各ビットについては、4.8 割り込み要求制御レジスタを参照してください。

EXIC_nレジスタは、メモリの8ビット操作命令または1ビット操作命令で読み出し/書き込みができます。この場合、1サイクルのウエイト・サイクルが自動的に挿入されます。

$\overline{\text{RESET}}$ 入力で内容が47Hになります。

4.11.3 外部割り込みマクロ・サービス制御レジスタ (EMSO, EMS1, EMS2)

EMS_nレジスタ (n=0-2) は、外部割り込み要求端子 ($\overline{\text{INTP0}}-\overline{\text{INTP2}}$) より発生する3つの割り込み要求によって起動されるマクロ・サービスを制御する8ビット・レジスタです。

EMSOレジスタは、EXF0フラグによって起動されるマクロ・サービスを制御します。

EMS1レジスタはEXF1フラグ, EMS2レジスタはEXF2フラグによって起動されるマクロ・サービスを制御します。

EMS_nレジスタは、メモリの8ビット操作命令または1ビット命令で読み出し/書き込みができます。この場合、1ウエイトが挿入されます。

図4-21 EMS0, EMS1, EMS2

7	6	5	4	3	2	1	0
MSM2	MSM1	MSM0	DIR	0	CH2	CH1	CH0

EMS_nレジスタの各ビットについては、4.5.4 マクロ・サービス制御レジスタを参照してください。

4.12 ソフトウェアによる割り込み

μ PD70325, 70335は、合計9種類のソフトウェアによる割り込みがあります(表4-2)。6種は、 μ PD70108, 70116のソフトウェアによる割り込みとコンパチブル(ただし、エミュレーション・モードに関する割り込みはありません)です。他の3種は、特有の機能です。

これらの割り込みのベクタはあらかじめ決まっています。

なお、BRKによる割り込み(シングルステップ割り込み)以外は、その割り込みが発生する条件が成立すれば常時受け付けられます(ハードウェアによる割り込みより優先されます)。ただし、BRKフラグによる割り込みは、BRK=1のときに発生しますが(ハードウェア、ソフトウェアの区別なく)、割り込みが受け付けられると自動的にBRKフラグはリセットされますので、他の割り込み(ハードウェア、ソフトウェアいずれも)より優先順位は低くなり、その割り込み処理中にBRKフラグによる割り込みは発生しません。

表4-2 ソフトウェアによる割り込み

割り込みソース	ベクタ	優先順位
DIVUのディバイド・エラー	0	1
DIVのディバイド・エラー		
CHKIND境界オーバ	5	
BRKV	4	
BRK 3	3	
BRK imm8	32-255	
BRKフラグ(シングルステップ)	1	2
入出力命令($\overline{\text{IBRK}}$ フラグ)	19	1
FPO命令	7	
BRKCS命令	—	

4.12.1 一般のソフトウェア割り込み

入出力命令による割り込みとFPO命令による割り込み以外のソフトウェアによる割り込みの受け付けの実行シーケンスは、ベクタ割り込みと同一になります。すなわち、次の命令のアドレス情報とPSWをスタックに退避し、IE=BRK=0にし、ベクタの内容をPSとPCへロードします。

次に各ソフトウェア割り込みについて説明します。

(1) DIVUのディバイド・エラー、DIVのディバイド・エラー

除算命令の実行により商がオーバフローした場合に必ず発生します。

(2) CHKIND境界オーバ

あらかじめ定義された配列の境界をインデクス値が越えているか否かをチェックする命令(CHKIND)の実行により、境界をインデクス値が越えていると判断された場合に発生します。

(3) BRKV

BRKV命令の実行時にV（オーバフロー・フラグ）がセットされていると発生します。

(4) BRK 3

BRK 3命令の実行により発生します。

(5) BRK imm8

BRK imm8命令の実行により発生します。

(6) BRKフラグ（シングルステップ）

BRK=1にセットされていると、一命令実行するたびに発生します。

リピート・プリフィクスが付加された場合、ループを抜けるまでは発生しません。

4.12.2 入出力命令割り込み

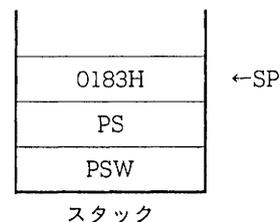
$\overline{\text{IBRK}}=0$ のときに入出力命令を実行しようとするとき割り込みが発生します。この割り込み受け付けのときにスタックに退避されるアドレス情報は、一般のソフトウェア割り込み（4.12.1参照）と異なり、その入出力命令が置かれているアドレスとなります。なお、この入出力命令にプリフィクスが付加されていた場合には、そのプリフィクスが置かれているアドレスとなります。この他の動作は一般のソフトウェア割り込みと同じになります。なお入出力命令割り込みから復帰時には、正常に復帰するためにスタック上のPCの値を調整する必要があります。

スタックに退避するアドレス情報を命令の先頭アドレスとすることにより、ソフトウェアで具体的にどの命令を実行しようとして割り込みが発生したのかを知ることができます。この機能により、従来の μ PD70108, 70116で使用していたプログラムを移植することが容易に行えます。

また、PSWについては割り込みが発生する直前の内容がスタックに退避され、その後、 $\text{IE}=\text{BRK}=0$, $\overline{\text{IBRK}}=1$ に自動的に設定されます。 $\overline{\text{IBRK}}=1$ にすることにより、割り込み処理中の入出力命令は入出力命令として実行され、割り込みからの復帰により自動的に元の状態（ $\overline{\text{IBRK}}=0$ ）に戻ります。

例 1. プリフィクスなし

アドレス	命令
⋮	⋮
0183H	IN AW, DW
⋮	⋮



例 2. プリフィクスあり

アドレス	命令
⋮	⋮
0183H	REP
0184H	INM
⋮	⋮

	←SP
0183H	
PS	
PSW	
スタック	

参考 一般のソフトウェア割り込み

アドレス	命令
⋮	⋮
0183H	BRK 3
⋮	⋮

	←SP
0184H	
PS	
PSW	
スタック	

4.12.3 FPO命令割り込み

μ PD70325, 70335は、 μ PD70108, 70116と外部バスの構成が異なるため、 μ PD70108, 70116用の浮動小数点演算用のコプロセッサは接続できません。したがって、このコプロセッサ用のFPO命令を実行しようとしたときにその命令の動作をエミュレートするために割り込みを発生させます。この割り込みは、入出力命令割り込みと同様にスタック上に退避させるPCの値は、この命令の先頭アドレス(プリフィクスが付けられたときは、プリフィクスの先頭アドレス)となります(4.12.2 入出力命令割り込み参照)。したがって、そのFPO命令をデコードし、ソフトウェアでエミュレーションすることが可能となります。FPO命令割り込みからの復帰時には、入出力命令割り込みと同様に、スタック上に退避させたPCの値を調整する必要があります。

4.13 割り込み要求が受け付けられない場合

★

割り込み要求が受け付けられない場合、次の項目を確認してください。

- ・割り込み要求制御レジスタのIFビットがセット(1)されている。また、リセット(0)されている場合は、ソフトウェアで書き換えていないか確認してください。
- ・割り込み要求制御レジスタのIMKビットがセット(1)されている。
- ・ISPRレジスタの優先順位の高いビットがセット(1)されている。
- ・PSWのIEフラグがDI状態になっている。

4.14 割り込み要求が受け付けられないタイミング

次の各命令とその次の命令の間ではすべての割り込み要求は受け付けられません。

(i) **sreg**に対する操作命令

```
MOV sreg, reg16; MOV sreg, mem16; POP sreg; POP PSW; MOVSPB
```

(ii) プリフィクス命令

```
PS:, SS:, DS0:, DS1:, REPC, REPNC, REP, REPE, REPZ, REPNE, REPNZ, BUSLOCK
```

(iii) **EI, RETI, DI**

(iv) **FINT**

受け付けられないタイミングで発生した割り込みは、INTを除けば、続く命令の実行終了後に受け付けられる状態であれば受け付けられます。

4.15 ブロック処理命令実行中の割り込み処理

ブロック処理（転送，比較，検索，入出力）命令実行中に割り込みを受け付けます。

割り込みは1回の処理を終了した時点で受け付けられます。このとき、スタックまたはレジスタ・バンクのPC退避領域に退避されるアドレスは、自動的にプリフィクスを含めた命令の先頭となります。割り込みからの復帰時にプリフィクスから再実行することにより、実行途中であったブロック処理を再開することができます。

ブロック処理の命令には、理論的にリピート・プリフィクスを含めて最大3種のプリフィクスを付加することができます。 μ PD70325, 70335では、割り込みを受け付けた時点で何種のプリフィクスが付加されているブロック処理命令を実行していたのか判別できるようになっており、その付加状態に応じてPCの値を自動的に減じてスタック等に退避します。

例 REP

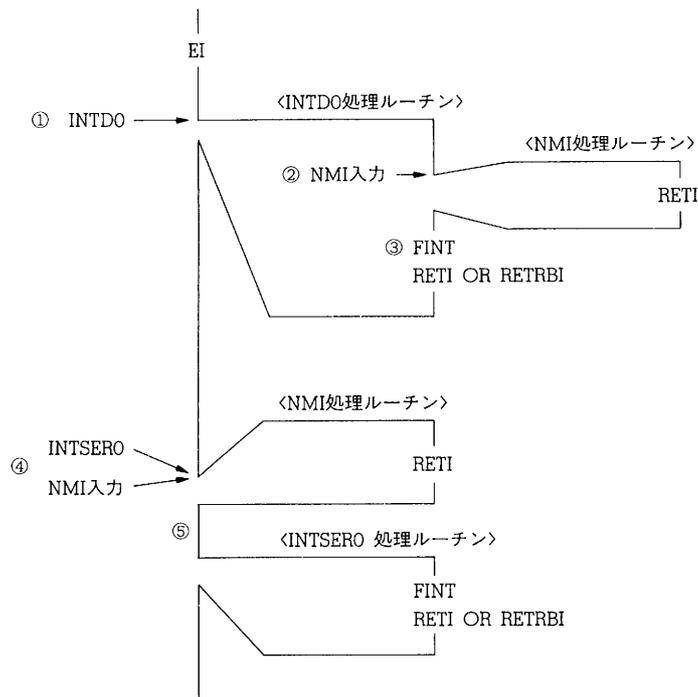
```
MOV BK SS:SRC BLK, DES BLOCK
```

4.16 割り込み要求の受け付け方

(1) NMI

- ソフトウェアによってマスクされない
- すべての割り込みに対して最優先

<メイン・ルーチン>

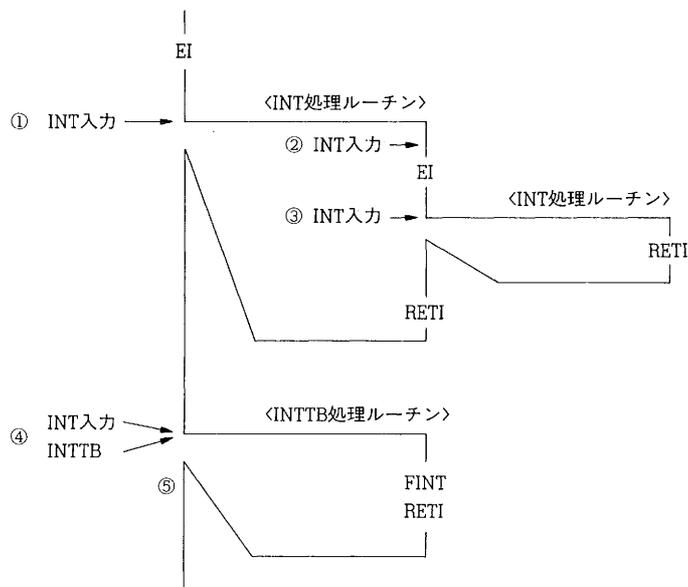


- ① 割り込み許可状態 (EI) で優先順位 2 に設定したINTDOが発生し、INTDO処理を開始する。
- ② INTDO処理中は割り込み禁止状態(DI)になるが、NMIの発生により、INTDO処理を禁止しNMI処理を開始する。
- ③ NMI処理が終了すると禁止されていたINTDOが処理され、復帰する。
- ④ 優先順位 4 に設定したINTSEROとNMIが同時に発生すると、すべての割り込みに対して最優先のNMIが受け付けられ、NMI処理を開始する。
- ⑤ NMI処理が終了すると保留されていたINTSEROが処理される。

(2) INT

- 割り込み許可状態 (EI=1) であれば、いつでも受け付けられる
- 複数の割り込みが発生したときの優先順位は最下位
- 多重処理コントローラの制御は受けない

〈メイン・ルーチン〉

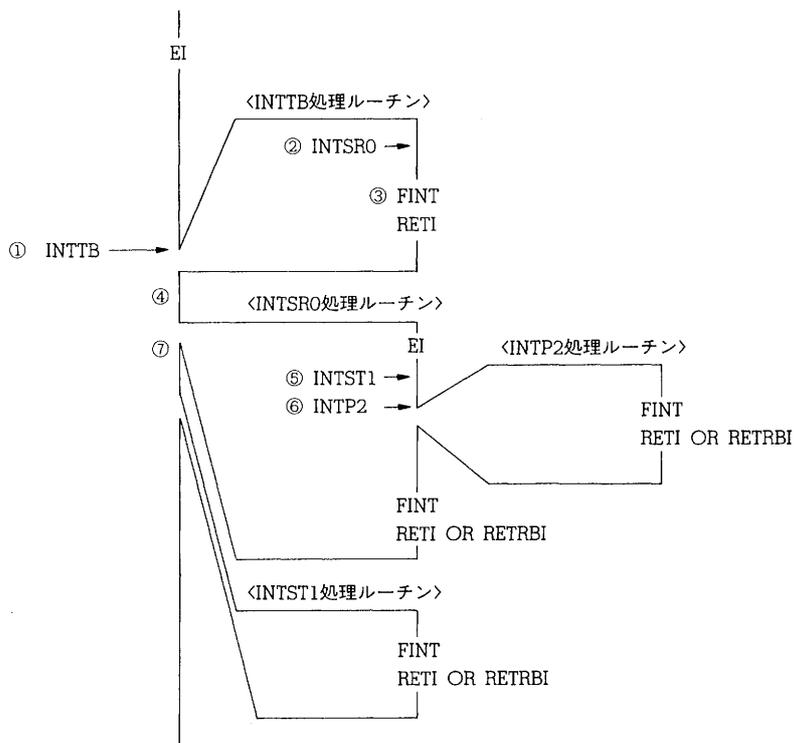


- ① 割り込み許可状態 (EI) でINTが発生すると、INT処理を開始する。
- ② INTが受け付けられると自動的に割り込み禁止状態 (DI) になるので、INTが再び発生しても受け付けない。
- ③ INT処理中でも割り込み許可状態 (EI) にすると、再びINTの発生があれば、INTの多重処理を行う。
- ④ INTと優先順位6に設定したINTTBが同時に発生すると、INTは優先順位が最下位のため受け付けられず、INTTBの処理を開始する。
- ⑤ INTTBの処理終了後、INTがイン・アクティブであれば、INTの割り込みは受け付けない。

(3) 多重処理制御を受ける割り込み

○優先順位に従って多重処理を行う

<メイン・ルーチン>

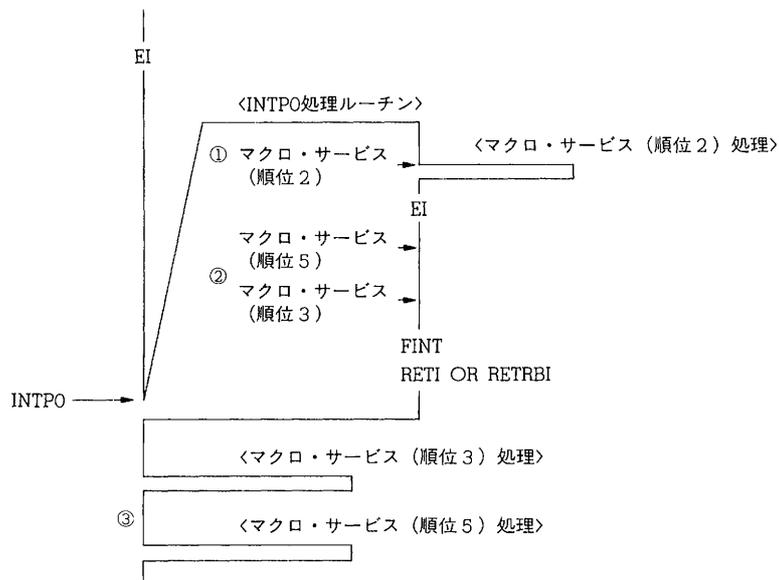


- ① 割り込み許可状態 (EI) で優先順位 6 に設定したINTTBが発生し、INTTB処理を開始する。
- ② INTTBが受け付けられると割り込み禁止状態 (DI) になるため、INTTBより高位の優先順位 4 に設定したINTSROが発生するが受け付けられない。
- ③ 優先順位制御を受ける割り込みは、復帰時、FINT命令を実行しなければならない。
- ④ INTTB処理が終了し割り込み許可状態 (EI) になると、保留されていたINTSRO処理を開始する。
- ⑤ 割り込み許可状態 (EI) で優先順位 5 に設定したINTST1が発生するが、INTSROの割り込み処理中でありINTSROより優先順位が低位なため、受け付けられない。
- ⑥ 続けて発生した優先順位 3 に設定したINTP2は、INTSROより高位のため受け付けられ、INTP2の処理を開始する。
- ⑦ INTP2処理が終了し、禁止されていたINTSROの処理も終了し復帰すると、INTST1が受け付けられる。

(4) マクロ・サービスによる割り込み

- 割り込み許可 (EI), 禁止 (DI) 状態にかかわらず受け付けられる
- 多重処理を受ける割り込みと同じく, 優先順位に従って多重処理を行う

〈メイン・ルーチン〉

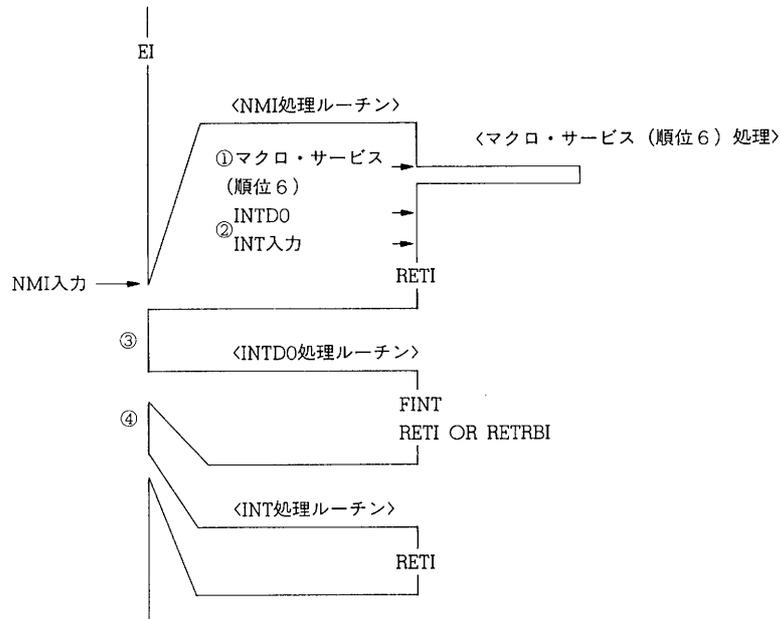


- ① 優先順位 3 に設定した INTPO 処理中に優先順位 2 に設定したマクロ・サービスによる割り込みが発生すると, 割り込み禁止状態 (DI) であってもマクロ・サービス処理を行う。
- ② 続いて割り込み許可状態 (EI) にした INTPO の処理中に優先順位 5 と 3 にそれぞれ設定したマクロ・サービスによる割り込みが発生するが, INTPO 以下の優先順位により保留される。
- ③ INTPO 処理が終了すると保留されていたマクロ・サービスによる割り込みが優先順位の高い順番に受け付けられる。

(5) マクロ・サービスによる割り込みの他の割り込みに対する優先順位

○NMI処理中でもマクロ・サービスによる割り込みは受け付けられる

<メイン・ルーチン>

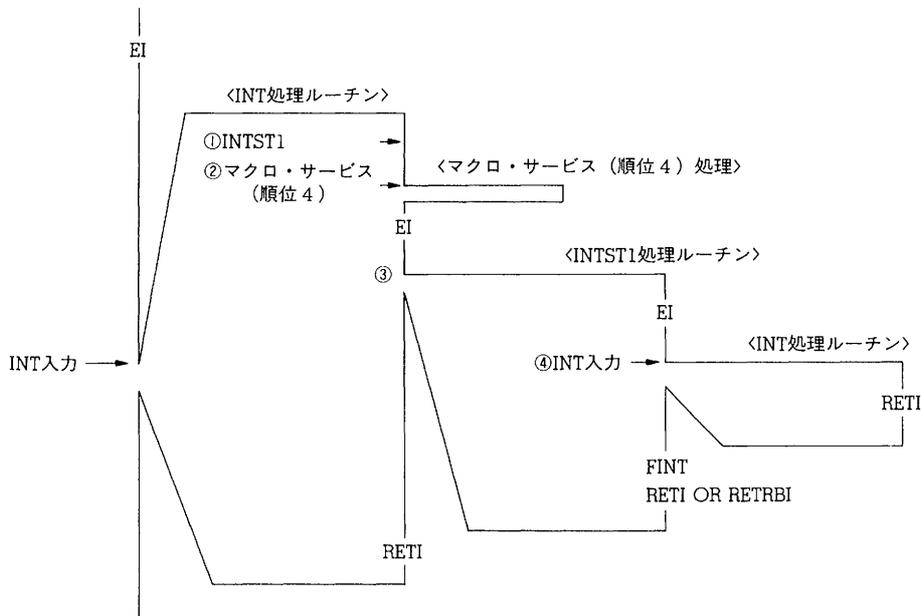


- ① NMI処理中で割り込み禁止状態 (DI) であっても、マクロ・サービスによる割り込みが発生するとマクロ・サービス処理は行う。
- ② 同じNMI処理中で割り込み禁止状態 (DI) にあるとき、多重処理制御を受ける割り込み (INTDO) やINTが発生しても受け付けられない。
- ③ NMI処理が終了すると保留されていたINTDOが受け付けられる。
- ④ INTDOの処理が終了した時点でINTがアクティブであれば、INTが受け付けられる。

(6) 各種割り込みの多重処理

○INT, 多重処理制御を受ける割り込み, マクロ・サービスによる割り込み

〈メイン・ルーチン〉



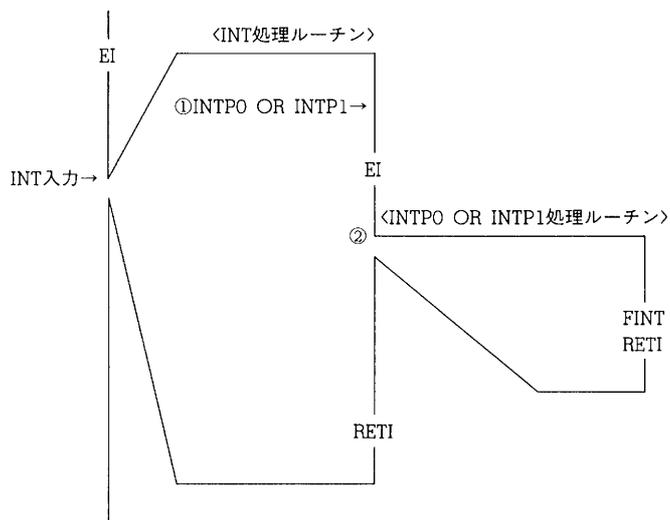
- ① INTが受け付けられ割り込み禁止状態(DI)になっているため, 優先順位5に設定したINTST1が発生しても受け付けられない。
- ② 続いて, 優先順位4に設定したマクロ・サービスによる割り込みが発生すると, マクロ・サービスの処理を行う。
- ③ 割り込み許可状態 (EI) にすると保留されていたINTST1の処理を開始する。
- ④ INTST1処理中, 割り込み許可状態 (EI) にあれば, INT発生によってINT処理を開始する。

(7) 外部割り込みの優先順位

★

- INT割り込み処理ルーチンを実行中に、INTP0またはINTP1入力の外部割り込みがあった場合、割り込みフラグがEI状態であれば受け付けます。

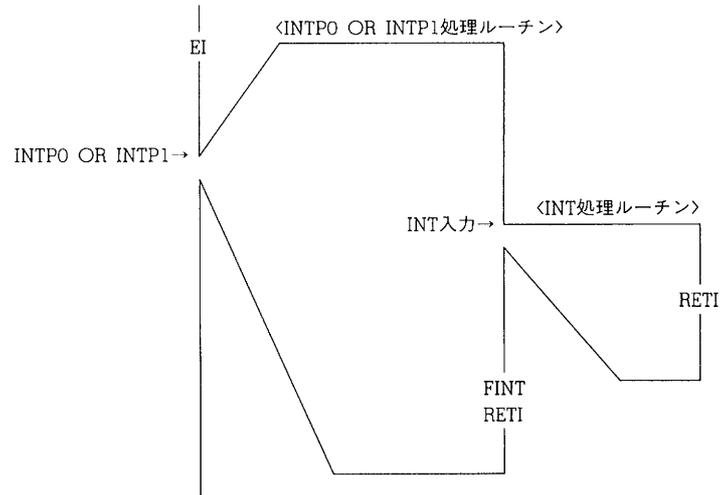
〈メイン・ルーチン〉



- ① INTが受け付けられて割り込み禁止状態 (DI) になっているため、INTP0またはINTP1の割り込み要求が発生しても受け付けられない。
- ② 割り込みフラグを許可状態 (EI) にすると、保留されていたINTP0またはINTP1の割り込み処理を開始する。

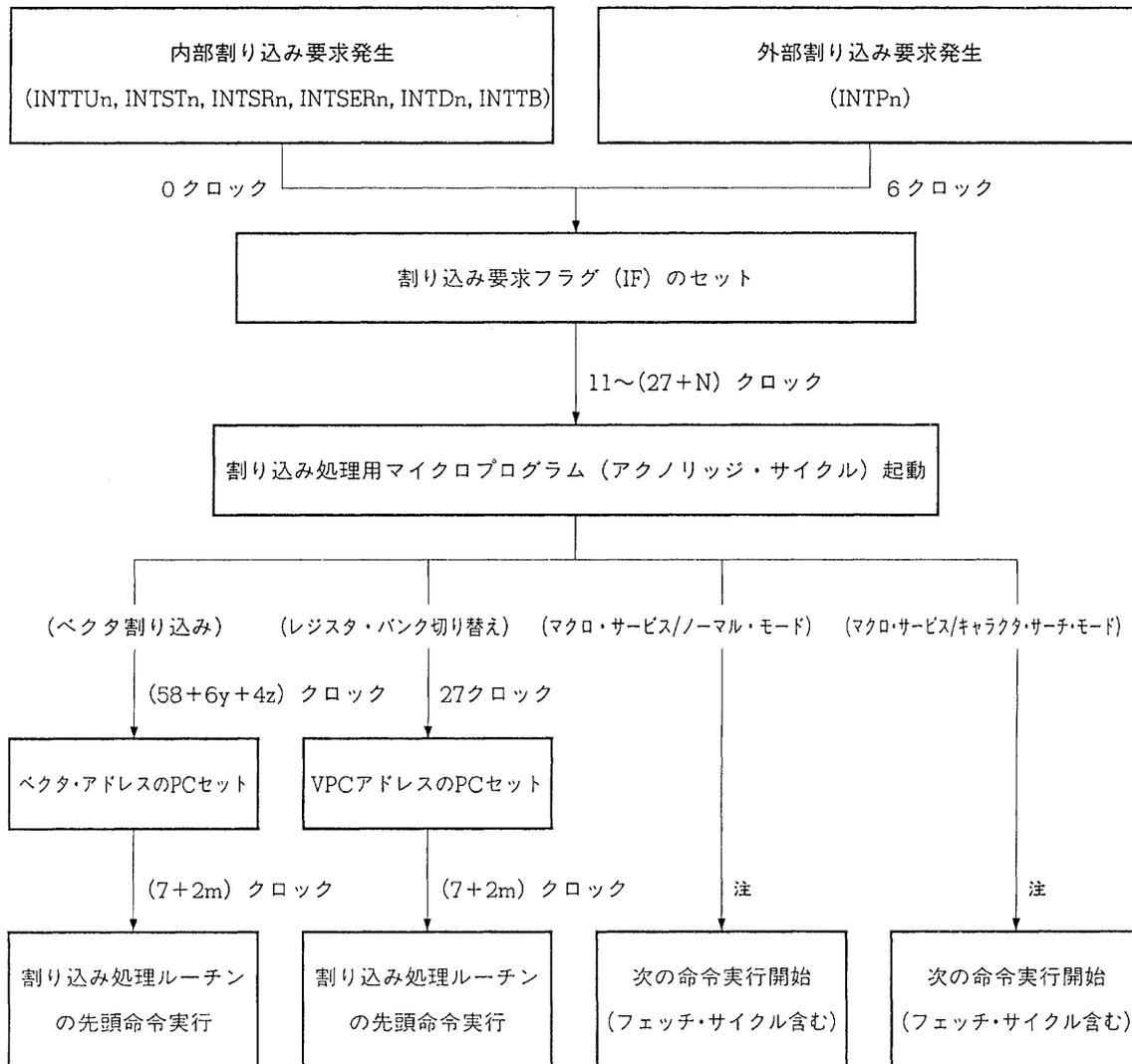
○INTPOまたはINTP1の割り込みを実行中にINT入力があった場合、割り込みフラグがEI状態であればいつでも受け付けます。

<メイン・ルーチン>



4.17 ハードウェア割り込みの応答時間

4.17.1 V25+の割り込み応答時間 (システム・クロック数)



注 マクロ・サービス転送処理時間 (システム・クロック数)

N : この時点でCPUが実行中の命令の残り実行クロック数

y : PC, PS, PSWをスタックに退避するときのメモリに対するウェイト数

z : ベクタPC, ベクタPSをリードするときのメモリに対するウェイト数

m : 割り込み処理ルーチンの先頭命令をフェッチするときのメモリに対するウェイト数 (2バイト分フェッチした時点で命令実行を開始します)。

注意 リフレッシュ・サイクル, ホールド要求, DMA要求, 他の割り込み要求などは考慮しておりません。

ノーマル・モード (単位: クロック)

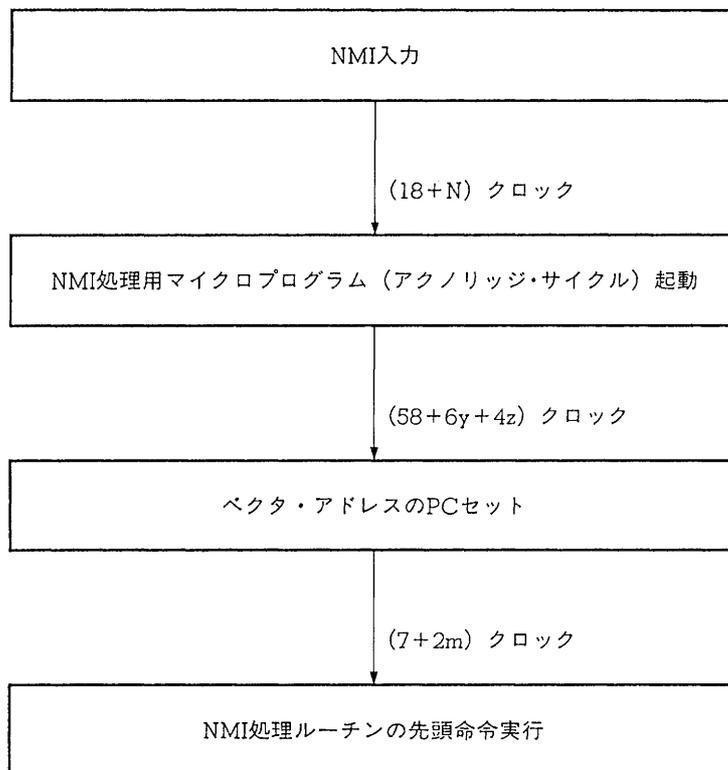
	バイト転送		ワード転送	
	内蔵RAM アクセス許可	内蔵RAM アクセス禁止	内蔵RAM アクセス許可	内蔵RAM アクセス禁止
	メモリ→SFR	24+t	19+t	26+2t
SFR→メモリ	22+t	20+t	22+2t	22+2t

キャラクタ・サーチ・モード (単位: クロック)

	バイト転送	
	内蔵RAM アクセス許可	内蔵RAM アクセス禁止
	メモリ→SFR	27+t
SFR→メモリ	37+t	34+t

t : 転送対象となるメモリに対するウェイト数

4.17.2 V25+のNMI応答時間（システム・クロック数）



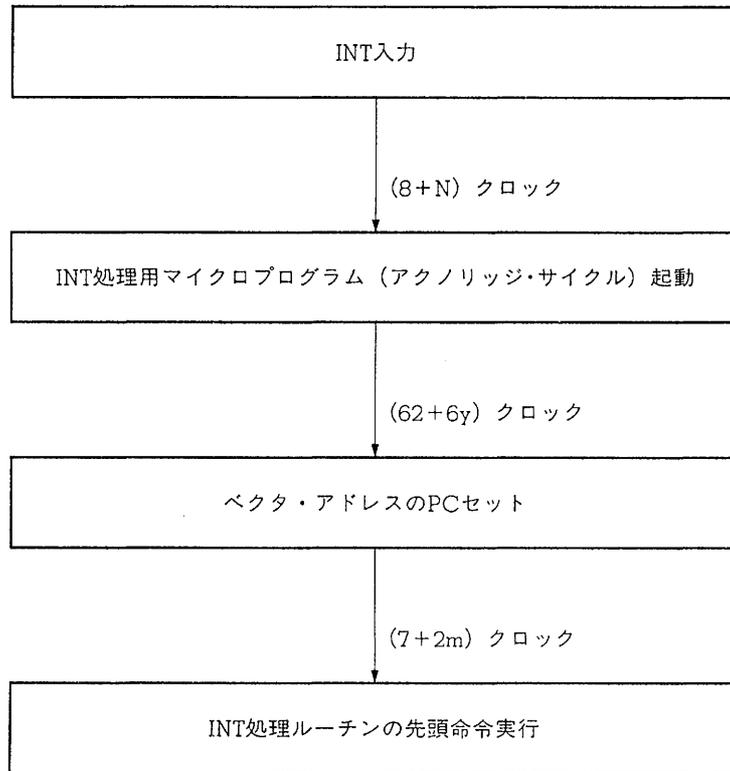
N : この時点で実行中の命令の残り実行クロック数

y : PC, PS, PSWをスタックに退避するときのメモリに対するウェイト数

z : ベクタPC, ベクタPSをリードするときのメモリに対するウェイト数

m : 割り込み処理ルーチンの先頭命令をフェッチするときのメモリに対するウェイト数（2バイト分フェッチした時点で命令実行を開始します）。

4.17.3 V25+のINT応答時間（システム・クロック数）



N : この時点で実行中の命令の残り実行クロック数

y : PC, PS, PSWをスタックに退避するときのメモリに対するウェイト数

z : ベクタPC, ベクタPSをリードするときのメモリに対するウェイト数

m : 割り込み処理ルーチンの先頭命令をフェッチするときのメモリに対するウェイト数（2バイト分フェッチした時点で命令実行を開始します）。

ノーマル・モード (単位: クロック)

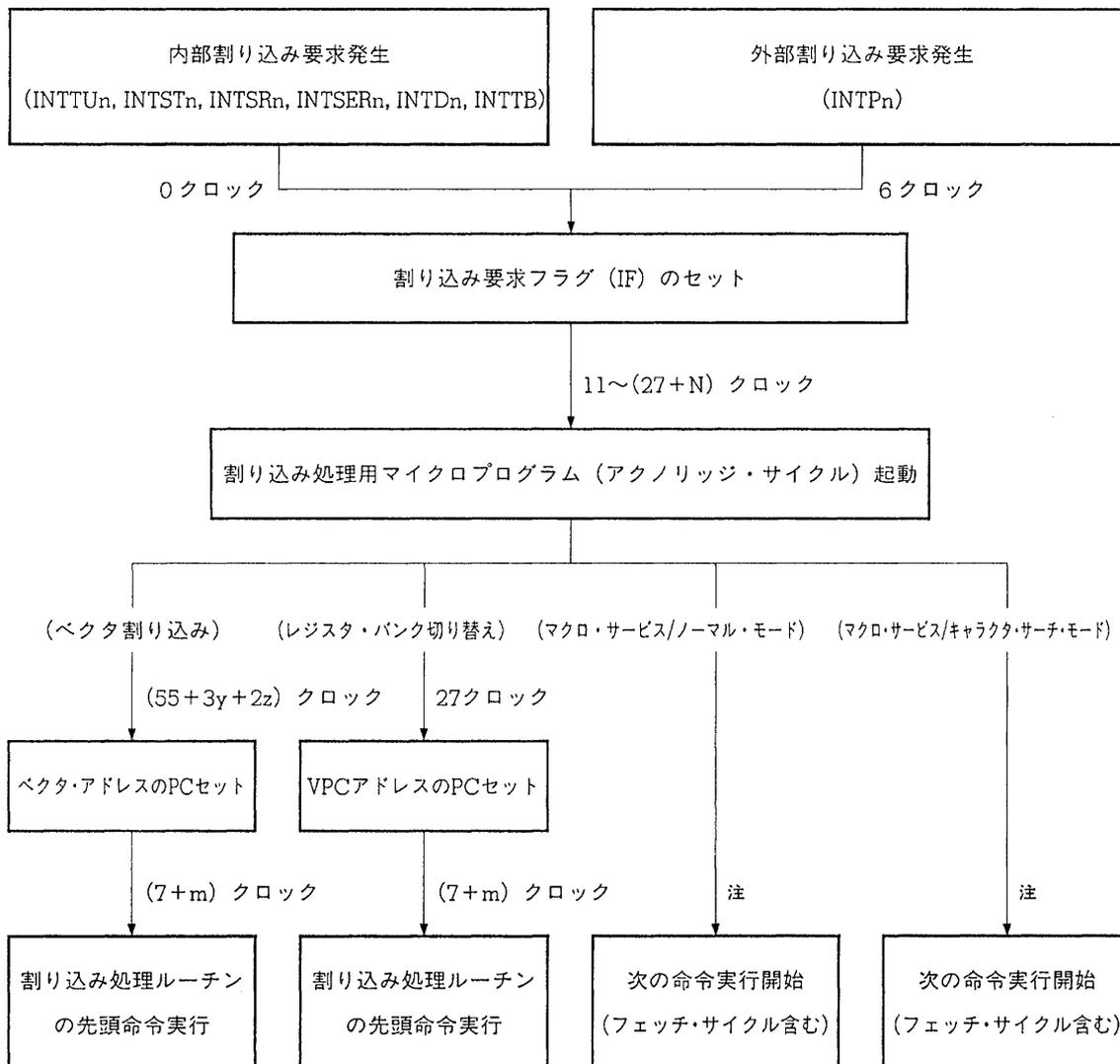
	バイト転送		ワード転送	
	内蔵RAM アクセス許可	内蔵RAM アクセス禁止	内蔵RAM アクセス許可	内蔵RAM アクセス禁止
メモリ→SFR	25+t	20+t	25+t	20+t
SFR→メモリ	22+t	21+t	22+t	21+t

キャラクタ・サーチ・モード (単位: クロック)

	バイト転送	
	内蔵RAM アクセス許可	内蔵RAM アクセス禁止
メモリ→SFR	28+t	28+t
SFR→メモリ	38+t	35+t

t : 転送対象となるメモリに対するウェイト数

4.17.4 V35+の割り込み応答時間（システム・クロック数）



注 マクロ・サービス転送処理時間（システム・クロック数）

N：この時点でCPUが実行中の命令の残り実行クロック数

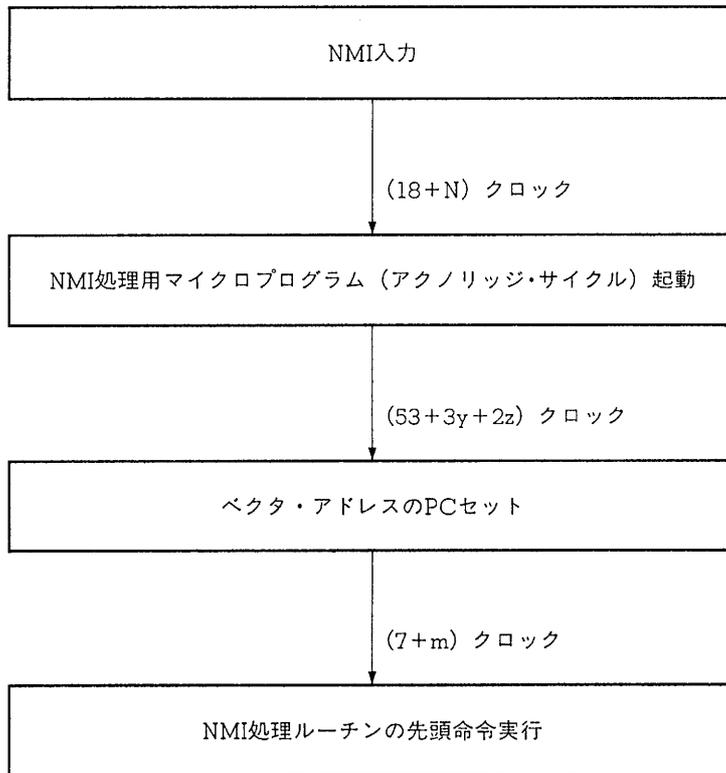
y：PC, PS, PSWをスタックに退避するときのメモリに対するウェイト数

z：ベクタPC, ベクタPSをリードするときのメモリに対するウェイト数

m：割り込み処理ルーチンの先頭命令をフェッチするときのメモリに対するウェイト数（2バイト分フェッチした時点で命令実行を開始します）。

注意 リフレッシュ・サイクル, ホールド要求, DMA要求, 他の割り込み要求などは考慮していません。

4.17.5 V35+のNMI応答時間（システム・クロック数）



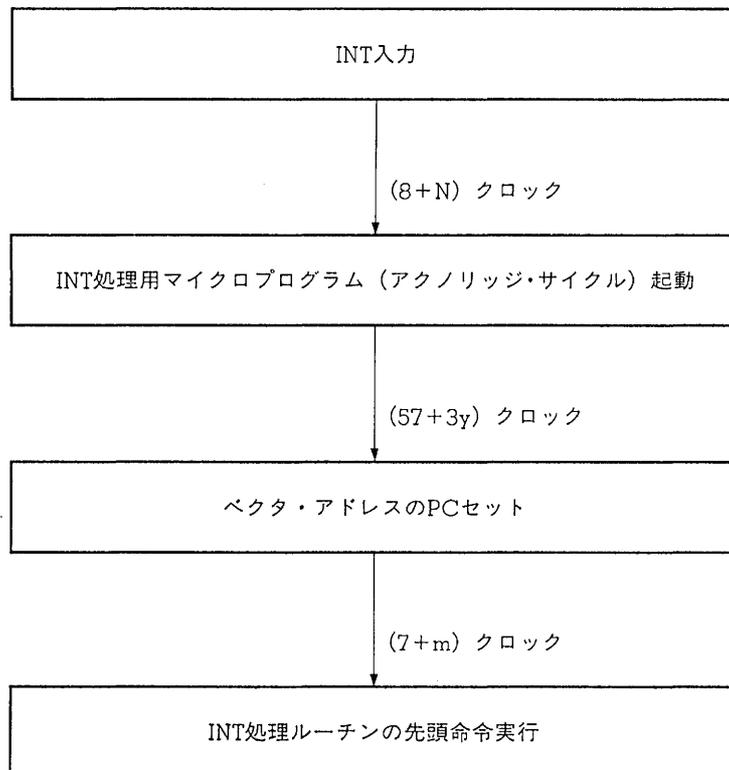
N : この時点で実行中の命令の残り実行クロック数

y : PC, PS, PSWをスタックに退避するときのメモリに対するウエイト数

z : ベクタPC, ベクタPSをリードするときのメモリに対するウエイト数

m : 割り込み処理ルーチンの先頭命令をフェッチするときのメモリに対するウエイト数（2バイト分フェッチした時点で命令実行を開始します）。

4.17.6 V35+のINT応答時間（システム・クロック数）



N：この時点で実行中の命令の残り実行クロック数

y：PC, PS, PSWをスタックに退避するときのメモリに対するウェイト数

z：ベクタPC, ベクタPSをリードするときのメモリに対するウェイト数

m：割り込み処理ルーチンの先頭命令をフェッチするときのメモリに対するウェイト数（2バイト分フェッチした時点で命令実行を開始します）。

第5章 バス制御

μ PD70325, 70335は、バスの制御のために表5-1に示す端子を持っています。

なお兼用端子についてはその機能を使用する場合、ポート・モード・コントロール・レジスタ (PM_{Cn}) によって相当する機能を選択する必要があります。

5

表5-1 バス制御のための端子機能

(a) μ PD70325

端子名	入出力	機 能	備 考
AO-A19	出	アドレス・バス	
D0-D7	入/出	データ・バス	
R/ \overline{W}	出	リード/ライトの識別	
\overline{MREQ}	出	メモリ・サイクルを示す	
\overline{MSTB}	出	メモリ・リード/メモリ・ライトのストロープ信号	
\overline{IOSTB}	出	I/Oサイクルのストロープ信号	
\overline{REFRQ}	出	メモリ・リフレッシュ・サイクルを示す	
HLDRQ	入	バス・ホールド要求信号	P27と兼用
HLDAK	出	バス・ホールド・アクノリッジ信号	P26と兼用
DMARQ0	入	DMA要求信号	P20と兼用
DMARQ1	入	DMA要求信号	P23と兼用
$\overline{DMAAK0}$	出	DMAアクノリッジ・サイクルを示す	P21と兼用
$\overline{DMAAK1}$	出	DMAアクノリッジ・サイクルを示す	P24と兼用
READY	入	外部からバス・サイクルにウエイトを挿入する	P17と兼用
\overline{INTAK}	出	割り込みアクノリッジ・サイクルを示す	P13, $\overline{INTP2}$ と兼用

μ PD70325の \overline{MSTB} 信号は、 \overline{MREQ} 信号より1/2クロック遅れてアクティブとなり、 \overline{MREQ} 信号と同時にインアクティブになります。メモリのアクセス信号を生成する際は、 \overline{MSTB} 信号より \overline{MREQ} 信号を用いるとメモリのアクセス信号が1/2クロック早くアクティブとなります。

★

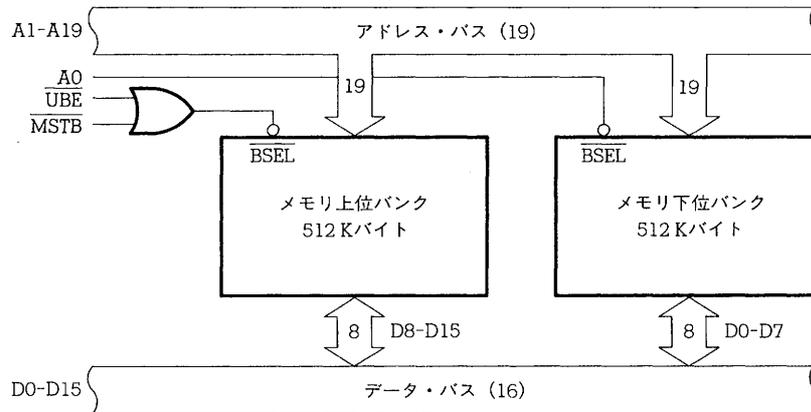
(b) μ PD70335

端子名	入出力	機能	備考
A0	出	アドレスのLSB出力。下位メモリ・バンクの選択に用いる。	
A9/A1-A16/A8, A17/A18, A19	出	19ビット・アドレスを時分割で出力	
A18/ $\overline{\text{UBE}}$	出	アドレスのビット18の出力と上位メモリ・バンクの選択信号を時分割で出力	
D0-D15	入/出	データ・バス	
$\overline{\text{R}}/\overline{\text{W}}$	出	リード/ライトの識別	
$\overline{\text{MREQ}}$	出	バス・サイクルが起動されたことを示す。上位アドレスのストロープ信号。	
$\overline{\text{MSTB}}$	出	メモリ・リード/メモリ・ライトのストロープ信号。下位アドレスのストロープ信号。	
$\overline{\text{IOSTB}}$	出	I/Oサイクルのストロープ信号。下位アドレスのストロープ信号。	
$\overline{\text{REFRQ}}$	出	メモリ・リフレッシュ・サイクルを示す。	
$\overline{\text{HLDRQ}}$	入	バス・ホールド要求信号	P27と兼用
$\overline{\text{HLDAK}}$	出	バス・ホールド・アクノリッジ信号	P26と兼用
$\overline{\text{DMARQ0}}$	入	DMA要求信号	P20と兼用
$\overline{\text{DMARQ1}}$	入	DMA要求信号	P23と兼用
$\overline{\text{DMAAK0}}$	出	DMAアクノリッジ・サイクルを示す。	P21と兼用
$\overline{\text{DMAAK1}}$	出	DMAアクノリッジ・サイクルを示す。	P24と兼用
READY	入	外部からバス・サイクルにウェイトを挿入する。	P17と兼用
$\overline{\text{INTAK}}$	出	割り込みアクノリッジ・サイクルを示す。	P13, $\overline{\text{INTP2}}$ と兼用

μ PD70335は1バイトごとにメモリ番地を管理し、さらに16ビットの外部データ・バスをもっているため、メモリを上位バンクと下位バンクに分けて接続します。図5-1にその概略を示します。

上位バンク、下位バンクのそれぞれのメモリのアドレス端子には、物理アドレスのA0を除く上位19ビットを入力します。A0信号とA18/ $\overline{\text{UBE}}$ 信号は、それぞれ下位、上位のメモリ・バンクの選択に用います。

図 5-1 メモリ・バンク構成



μPD70335のメモリ・サイクルは、T1, T2, T3の3ステートからなります (5.5 バス・タイミング参照)。

T1ステートでは20ビット・アドレスの内の第1のアドレス(上位アドレス)を、T2ステートでは20ビット・アドレスの内の第2のアドレス(下位アドレス)を外部アドレス・バス上に出力します。T3ステートではデータの読み出し、書き込みを行います。

ウェイト・ステートTWは、リード・サイクルではT2とT3ステートの間に挿入されます。ライト・サイクルでは、T1とT2ステートの間に挿入されます。

A0端子は、T1ステートからT3ステートまで物理アドレスの最下位ビットを出力します。A18/ $\overline{\text{UBE}}$ 端子は、T1ステートで物理アドレスの第18ビットを出力し、T1ステートの次のステートで $\overline{\text{UBE}}$ 信号を出力します。

A0信号と $\overline{\text{UBE}}$ 信号の関係は、表 5-2 のようになります。

表 5-2 μPD70335のデータ・アクセス

アクセス	$\overline{\text{UBE}}$	A0	バス・サイクル数
偶数アドレスのワード	0	0	1
奇数アドレスのワード	0	1	2
	1	0	
偶数アドレスのバイト	1	0	1
奇数アドレスのバイト	0	1	1

I/Oリード/ライト・サイクルのタイミングは、メモリ・リード/ライト・サイクルと同じタイミングです。ただし、 $\overline{\text{MSTB}}$ の代わりに $\overline{\text{IOSTB}}$ がアクティブになります。

メモリ・リフレッシュ・サイクルでは、 $\overline{\text{MREQ}}$ 信号と $\overline{\text{MSTB}}$ 信号はインアクティブです。

μPD70335は、外部メモリとI/Oをアクセスする場合、11本のアドレス端子とA18/ $\overline{\text{UBE}}$ 端子の計12本の端子から、表 5-3 に示すように20ビットの物理アドレスを時分割して出力します。

表 5-3 アドレス時分割出力

端子名	メモリ・サイクル		I/Oサイクル		リフレッシュ・サイクル
	1回目	2回目	1回目	2回目	
A0	A0	A0	A0	A0	"0"
A9/A1	A9	A1	A9	A1	A0
A10/A2	A10	A2	A10	A2	A1
A11/A3	A11	A3	A11	A3	A2
A12/A4	A12	A4	A12	A4	A3
A13/A5	A13	A5	A13	A5	A4
A14/A6	A14	A6	A14	A6	A5
A15/A7	A15	A7	A15	A7	A6
A16/A8	A16	A8	"0"	A8	A7
A17/A18	A17	A18	"0"	"0"	A8
A19	A19	A19	"0"	"0"	"0"
A18/ $\overline{\text{UBE}}$	A18	$\overline{\text{UBE}}$	"0"	$\overline{\text{UBE}}$	"0"
$\overline{\text{REFRQ}}$	"1"	"1"	"1"	"1"	"0"

★

5.1 プログラマブル・ウェイト機能

μPD70325, 70335は、バス・サイクル時（メモリ・リフレッシュ・サイクルを除く）にウェイト・ステートを挿入することがソフトウェアによって指定できます。これは1 Mバイトのメモリ空間を128 Kバイト単位の8ブロックとI/O空間に対して図5-2に示すようにウェイト・コントロール・レジスタ（WTC）によって指定します。ただし、メモリ空間のブロック6（C0000H-DFFFFH）とブロック7（E0000H-FFFFFFH）は同一の設定となります。

ウェイト・ステートの指定は、表5-4に示すように0, 1, 2, および2以上でREADY端子による制御の4つから、他のブロックとは独立に任意に指定できます。なお、READY端子による制御を使用する場合、READY端子はP17と兼用となっているためポート1モード・コントロール・レジスタ（PMC1）のビット7を1にする必要があります。PMC1のビット7が0の場合は、常にREADY状態すなわちウェイト・ステートは2ステートになります。READY端子による制御を選んだ場合は、READY端子の状態にかかわらず2ステートのウェイト・ステートTAWが挿入されます。READY端子はレベルでセンスされ、ロウ・レベルの期間中ウェイト・ステートが挿入されます。

内部データ領域に対するアクセスは、プログラマブル・ウェイト機能の影響を受けません。この設定は、リフレッシュ時を除いた外部領域に対するすべてのアクセスに適用されます。

RESET入力でWTCレジスタの内容はFFFFFFHに初期化されます。

図5-2 WTC

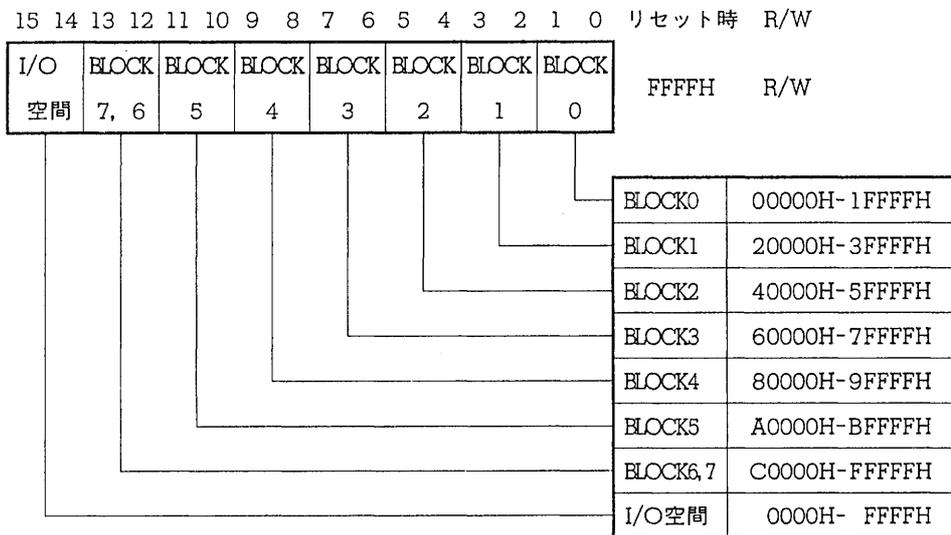


表 5-4 ウェイト・ステートの設定

BLOCKn/I/O空間	ウェイト・ステート
00	0 ステート
01	1 ステート
10	2 ステート
11	2 ステート+READY端子による挿入ステート

ウェイト・コントロール・レジスタ (WTC) でREADY端子によるウェイト制御を選ぶと、 μ PD70325, 70335は次に示すように2ウェイト (TAW) を自動的に挿入します。 μ PD70325, 70335のREADY端子によるウェイトを図5-3, 図5-4に示します。

μ PD70325 : T1ステートとT2ステートの間

μ PD70335 : T2ステートとT3ステートの間

μ PD70325, 70335は、READY端子の状態を最初のTAWでサンプリングします。このとき、READY端子の状態によって、ウェイト・ステート (TW) 挿入の許可/禁止を行います。

(1) READY端子がハイ・レベルの場合

自動挿入するTAW以上のウェイト・ステート挿入 (TW) を禁止します。READY端子はウェイト・ステート挿入を禁止するときだけ、ハイ・レベルにしてください。それ以外はロウ・レベルにしてください。

(2) READY端子がロウ・レベルの場合

自動挿入するTAWのあとにウェイト・ステート (TW) を追加します。TWは、TAWおよびTWでREADY端子のロウ・レベルをサンプリングした回数分が挿入されます。なお、最初のTAWでREADY端子がハイ・レベルの場合、次のTAWがロウ・レベルであってもREADYによるウェイト・ステート (TW) は挿入されません。

$\overline{\text{MREQ}}$, $\overline{\text{MSTB}}$, $\overline{\text{IOSTB}}$ の各制御信号は μ PD70325と μ PD70335では出力タイミングが異なります。READY信号のタイミング規定は次のようになります。

μ PD70325 : メモリ・リード/ライト・サイクルのとき、 $\overline{\text{MREQ}}$ により規定します。

I/Oリード/ライト・サイクルのとき、 $\overline{\text{IOSTB}}$ により規定します。

μ PD70335 : メモリ・リード/ライト・サイクル^注のとき、 $\overline{\text{MREQ}}$ により規定します。

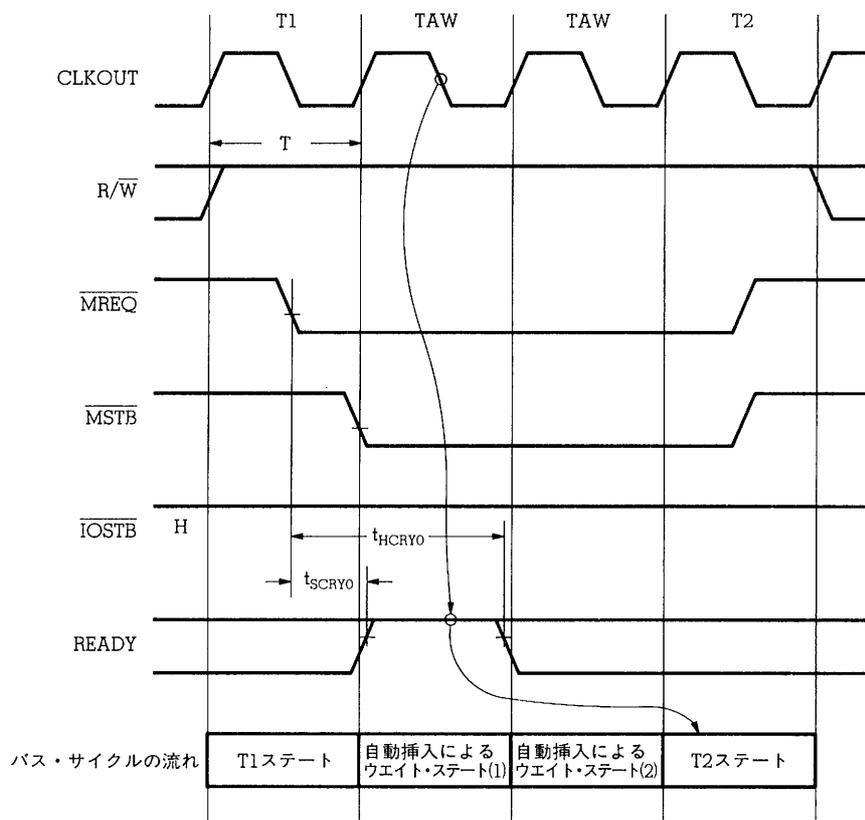
I/Oリード/ライト・サイクルのとき、 $\overline{\text{MREQ}}$ および $\overline{\text{IOSTB}}$ により規定します。

注 μ PD70335はメモリ・リード・サイクルとメモリ・ライト・サイクルでは $\overline{\text{MSTB}}$ の出力タイミングが異なります。

注意 ウェイト・ステートにはリフレッシュ・サイクルが挿入されません。したがって、リフレッシュ機能を使用してDRAMのリフレッシュを行う場合、ウェイト・ステートが長くなると、その間はリフレッシュが行われなくなり、DRAMの内容を保持しないことがあります。

図 5-3 READY端子によるウェイト (μ PD70325) (1/8)

(a) メモリ・リード・サイクル ウェイト・ステート追加なし



動作周波数10 MHzの場合

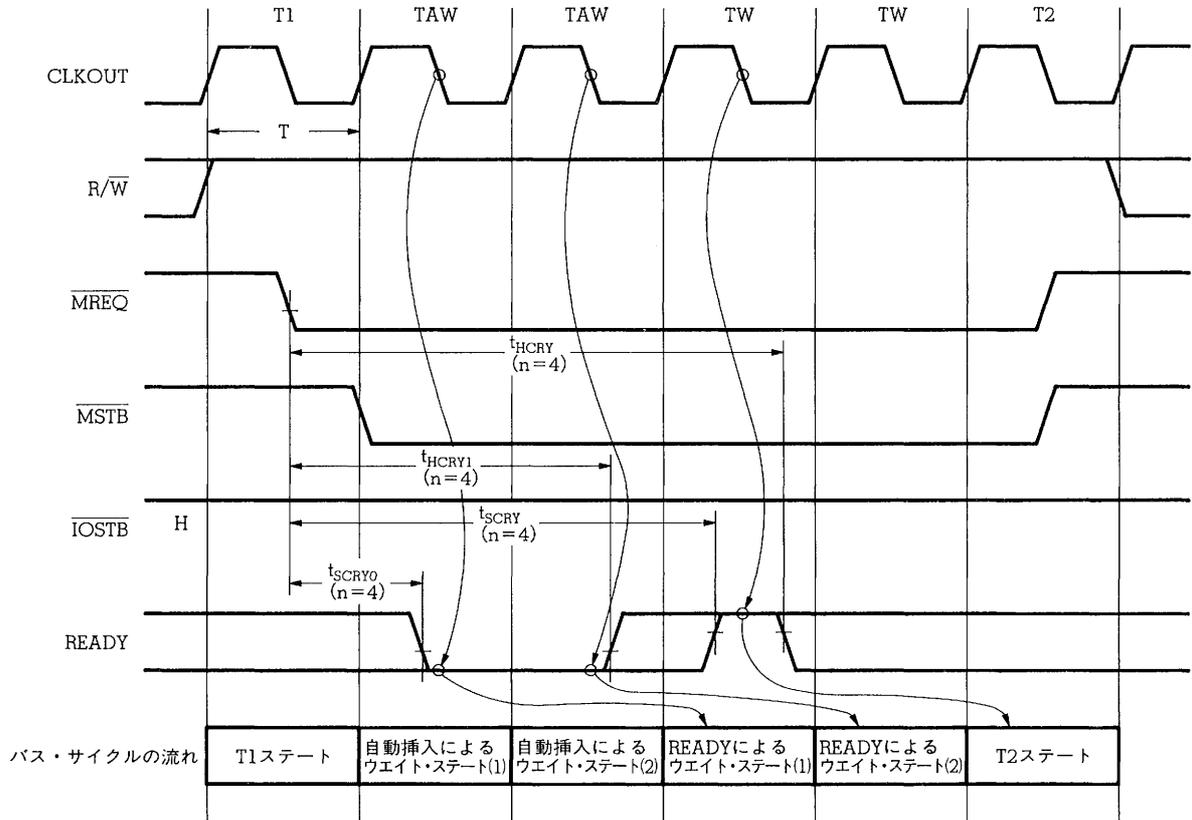
$$T = 100 \text{ ns}$$

$$t_{\text{SCRYO}} = T - 80 (\text{Max.}) = 20 \text{ ns}$$

$$t_{\text{HCRYO}} = T (\text{Min.}) = 100 \text{ ns}$$

図 5-3 READY端子によるウェイト (μ PD70325) (2/8)

(b) メモリ・リード・サイクル 2 ウェイト追加



動作周波数10 MHzの場合

$$T = 100 \text{ ns}$$

合計のウェイト数は4ウェイトのため、nには4を代入します。

$$t_{\text{SCRY0}} = T - 80 (\text{Max.}) = 20 \text{ ns}$$

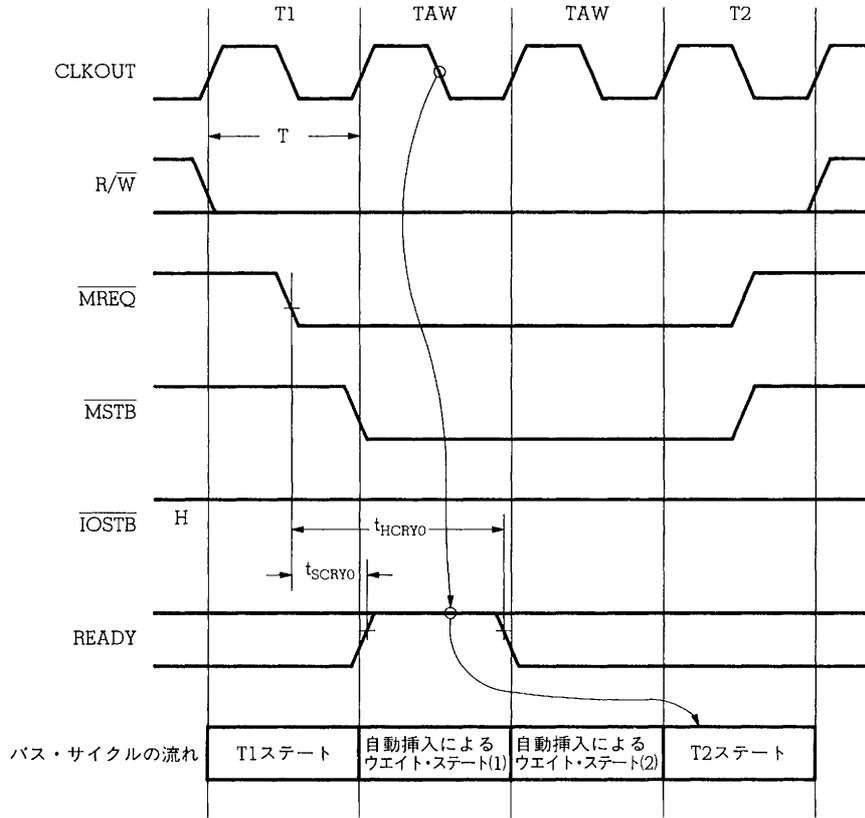
$$t_{\text{HCRY1}} = (n - 2)T (\text{Min.}) = 200 \text{ ns}$$

$$t_{\text{SCRY}} = (n - 1)T - 80 (\text{Max.}) = 220 \text{ ns}$$

$$t_{\text{HCRY}} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$$

図 5-3 READY端子によるウェイト (μ PD70325) (3/8)

(c) メモリ・ライト・サイクル ウェイト・ステート追加なし



動作周波数10 MHzの場合

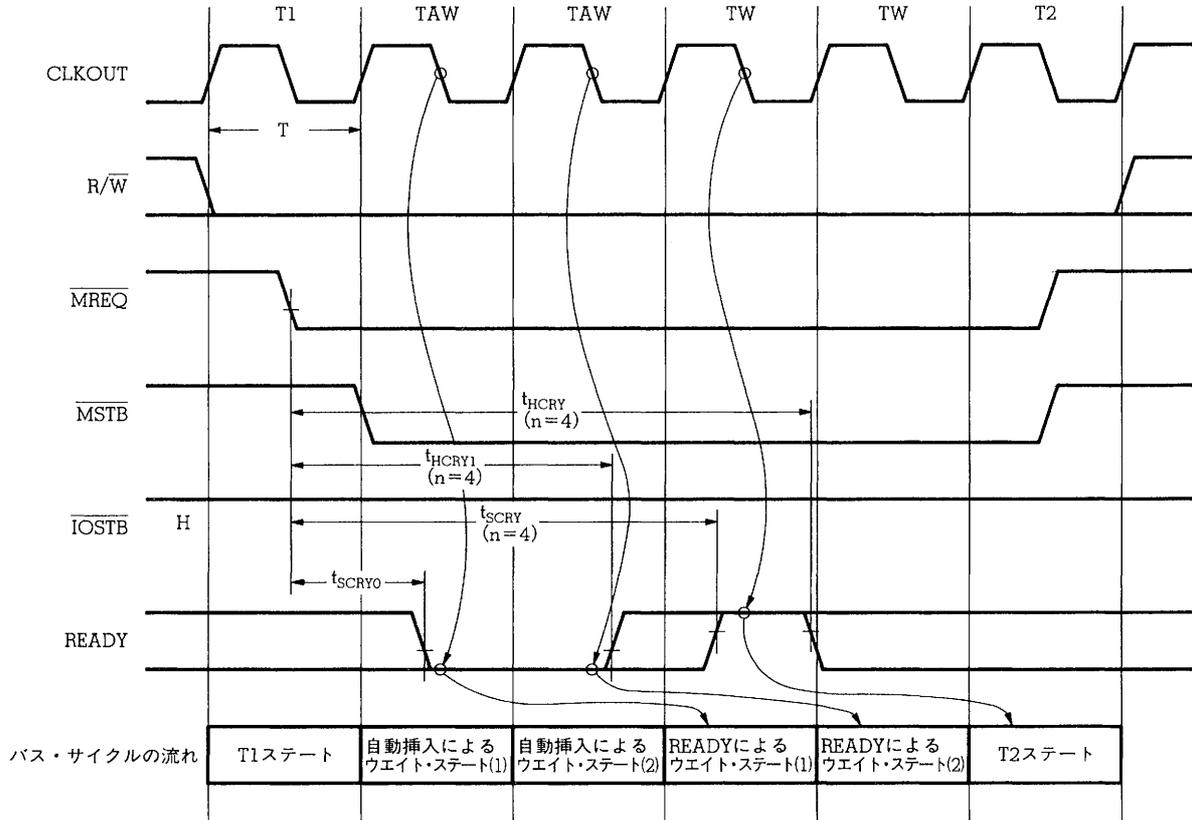
$$T = 100 \text{ ns}$$

$$t_{\text{SCRYO}} = T - 80 (\text{Max.}) = 20 \text{ ns}$$

$$t_{\text{HCRYO}} = T (\text{Min.}) = 100 \text{ ns}$$

図5-3 READY端子によるウェイト (μ PD70325) (4/8)

(d) メモリ・ライト・サイクル 2ウェイト追加



動作周波数10 MHzの場合

$$T = 100 \text{ ns}$$

合計のウェイト数は4ウェイトのため、nには4を代入します。

$$t_{\text{SCRY0}} = T - 80 (\text{Max.}) = 20 \text{ ns}$$

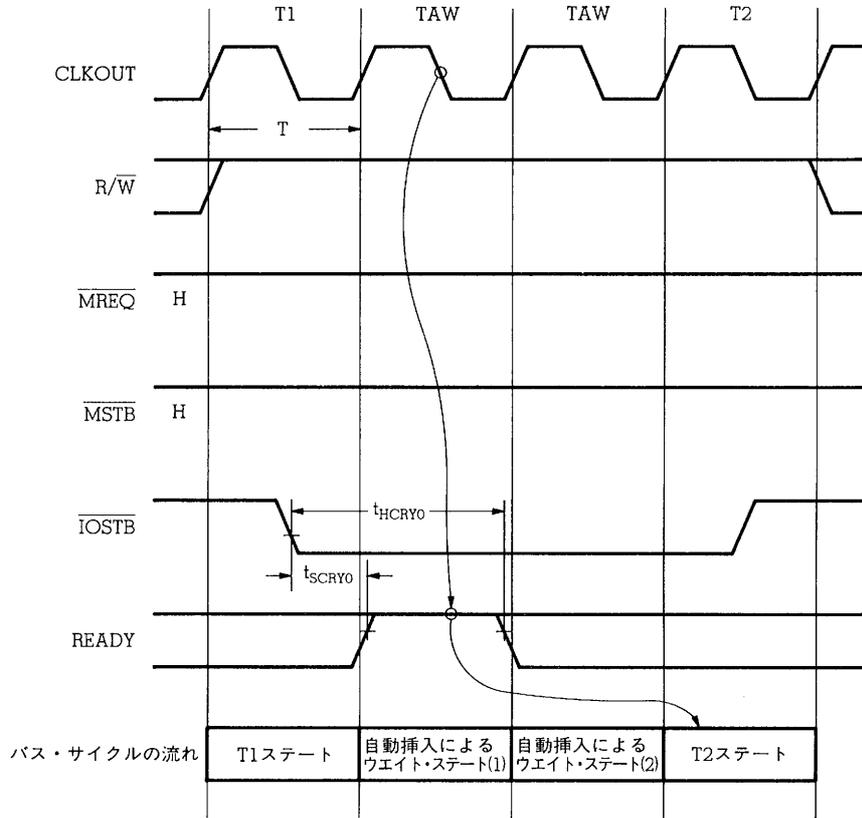
$$t_{\text{HCRY1}} = (n - 2)T (\text{Min.}) = 200 \text{ ns}$$

$$t_{\text{SCRY}} = (n - 1)T - 80 (\text{Max.}) = 220 \text{ ns}$$

$$t_{\text{HCRY}} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$$

図5-3 READY端子によるウェイト (μPD70325) (5/8)

(e) I/Oリード・サイクル ウェイト・ステート追加なし



動作周波数10 MHzの場合

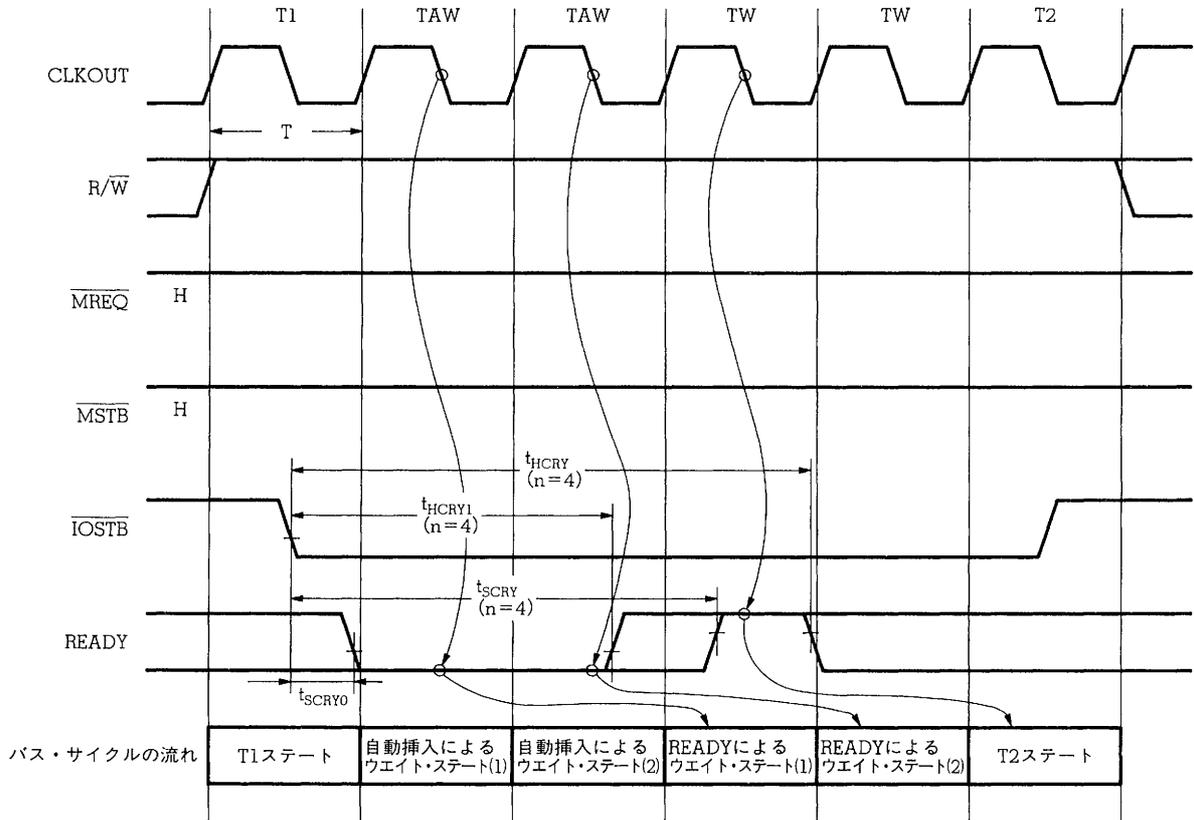
$$T = 100 \text{ ns}$$

$$t_{\text{SCRYO}} = T - 80 (\text{Max.}) = 20 \text{ ns}$$

$$t_{\text{HCRYO}} = T (\text{Min.}) = 100 \text{ ns}$$

図5-3 READY端子によるウェイト (μPD70325) (6/8)

(f) I/Oリード・サイクル 2ウェイト追加



動作周波数10 MHzの場合

$T = 100 \text{ ns}$

合計のウェイト数は4ウェイトのため、nには4を代入します。

$t_{\text{SCRY0}} = T - 80 (\text{Max.}) = 20 \text{ ns}$

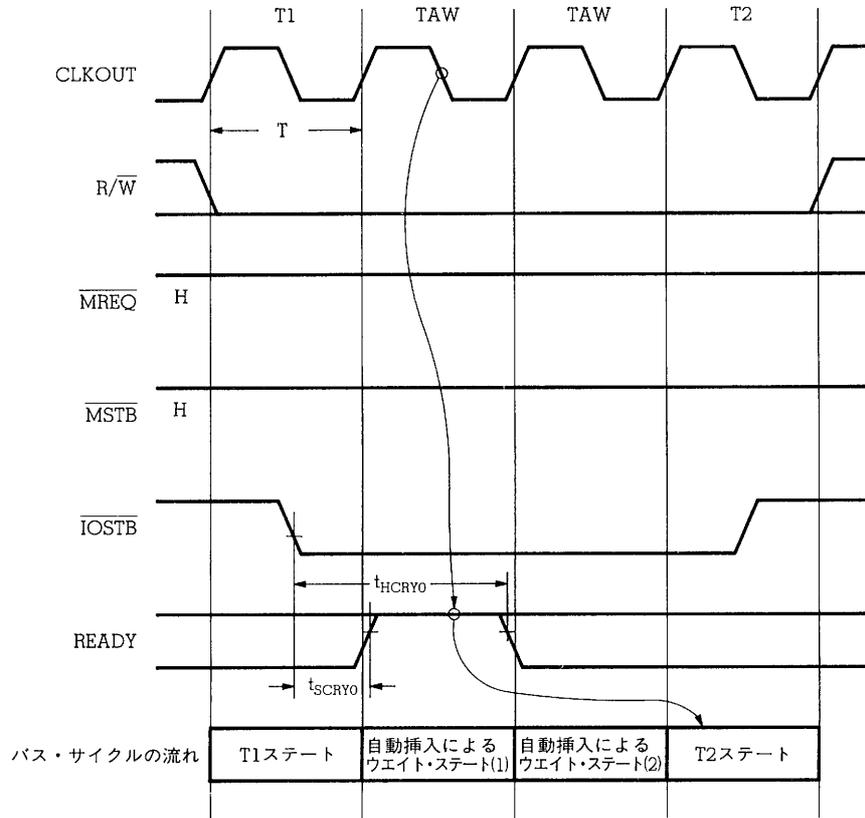
$t_{\text{HCRY1}} = (n - 2)T (\text{Min.}) = 200 \text{ ns}$

$t_{\text{SCRY}} = (n - 1)T - 80 (\text{Max.}) = 220 \text{ ns}$

$t_{\text{HCRY}} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$

図 5-3 READY端子によるウェイト (μPD70325) (7/8)

(g) I/Oライト・サイクル ウェイト・ステート追加なし



動作周波数10 MHzの場合

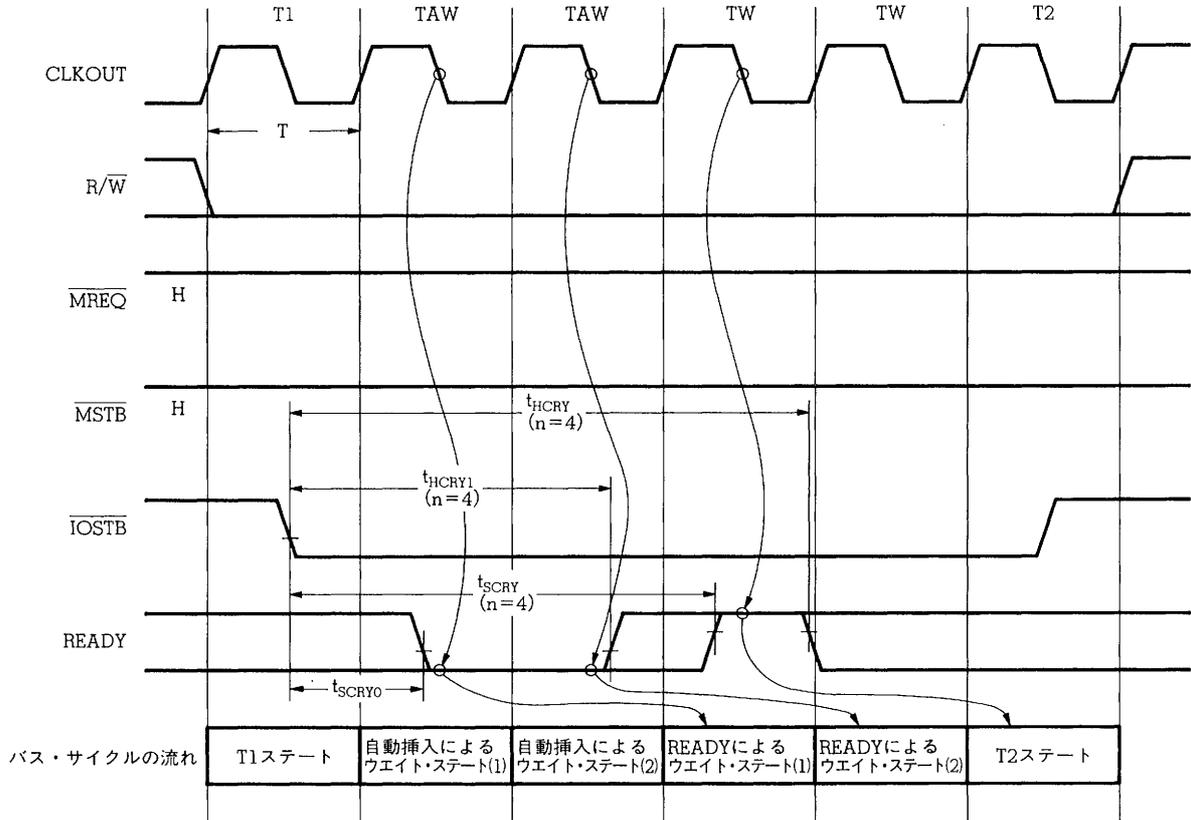
$$T = 100 \text{ ns}$$

$$t_{\text{SCRYO}} = T - 80 (\text{Max.}) = 20 \text{ ns}$$

$$t_{\text{HCRYO}} = T (\text{Min.}) = 100 \text{ ns}$$

図5-3 READY端子によるウェイト (μPD70325) (8/8)

(h) I/Oライト・サイクル 2ウェイト追加



動作周波数10 MHzの場合

$T = 100 \text{ ns}$

合計のウェイト数は4ウェイトのため、nには4を代入します。

$t_{SCRY0} = T - 80(\text{Max.}) = 20 \text{ ns}$

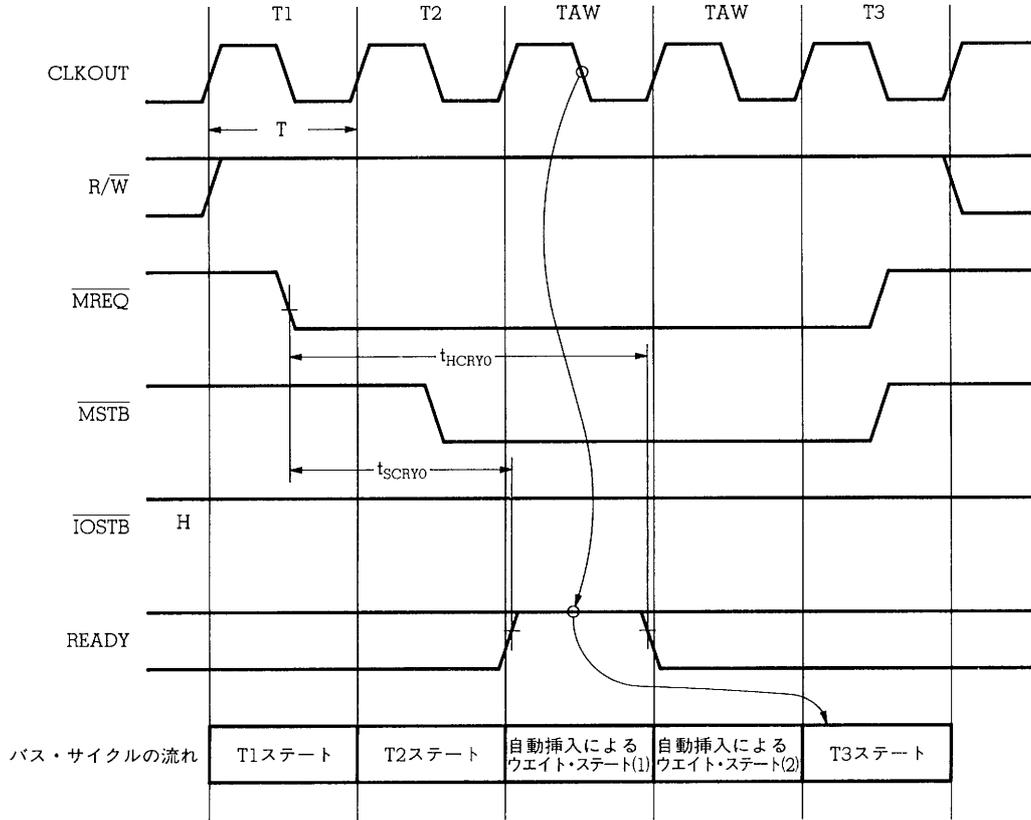
$t_{HCRY1} = (n - 2)T(\text{Min.}) = 200 \text{ ns}$

$t_{SCRY} = (n - 1)T - 80(\text{Max.}) = 220 \text{ ns}$

$t_{HCRY} = (n - 1)T(\text{Min.}) = 300 \text{ ns}$

図5-4 READY端子によるウェイト (μPD70335) (1/8)

(a) メモリ・リード・サイクル ウェイト・ステート追加なし



動作周波数10 MHzの場合

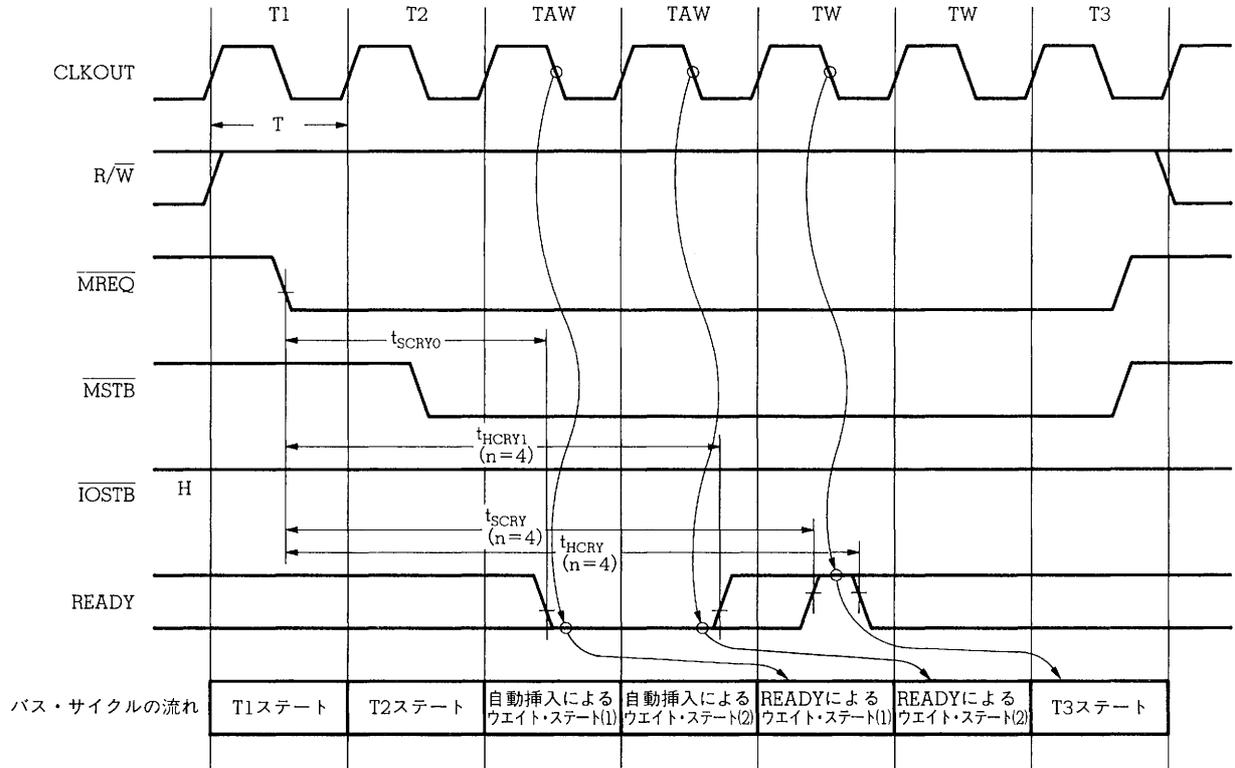
$T = 100 \text{ ns}$

$t_{SCRYO} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$

$t_{HCRYO} = 2T (\text{Min.}) = 200 \text{ ns}$

図5-4 READY端子によるウェイト (μ PD70335) (2/8)

(b) メモリ・リード・サイクル 2ウェイト追加



動作周波数10 MHzの場合

$$T = 100 \text{ ns}$$

合計のウェイト数は4ウェイトのため、nには4を代入します。

$$t_{SCRY0} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$$

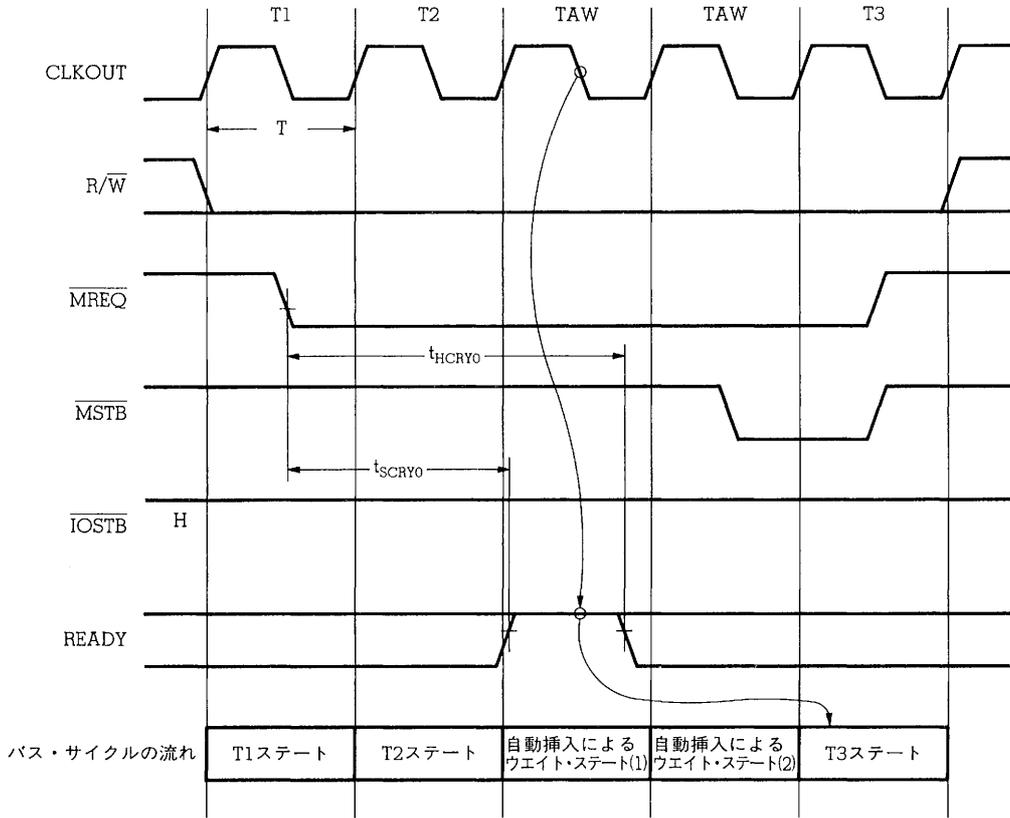
$$t_{HCRY1} = (n-1)T (\text{Min.}) = 300 \text{ ns}$$

$$t_{SCRY} = nT - 80 (\text{Max.}) = 320 \text{ ns}$$

$$t_{HCRY} = nT (\text{Min.}) = 400 \text{ ns}$$

図 5-4 READY端子によるウェイト (μPD70335) (3/8)

(c) メモリ・ライト・サイクル ウェイト・ステート追加なし



動作周波数10 MHzの場合

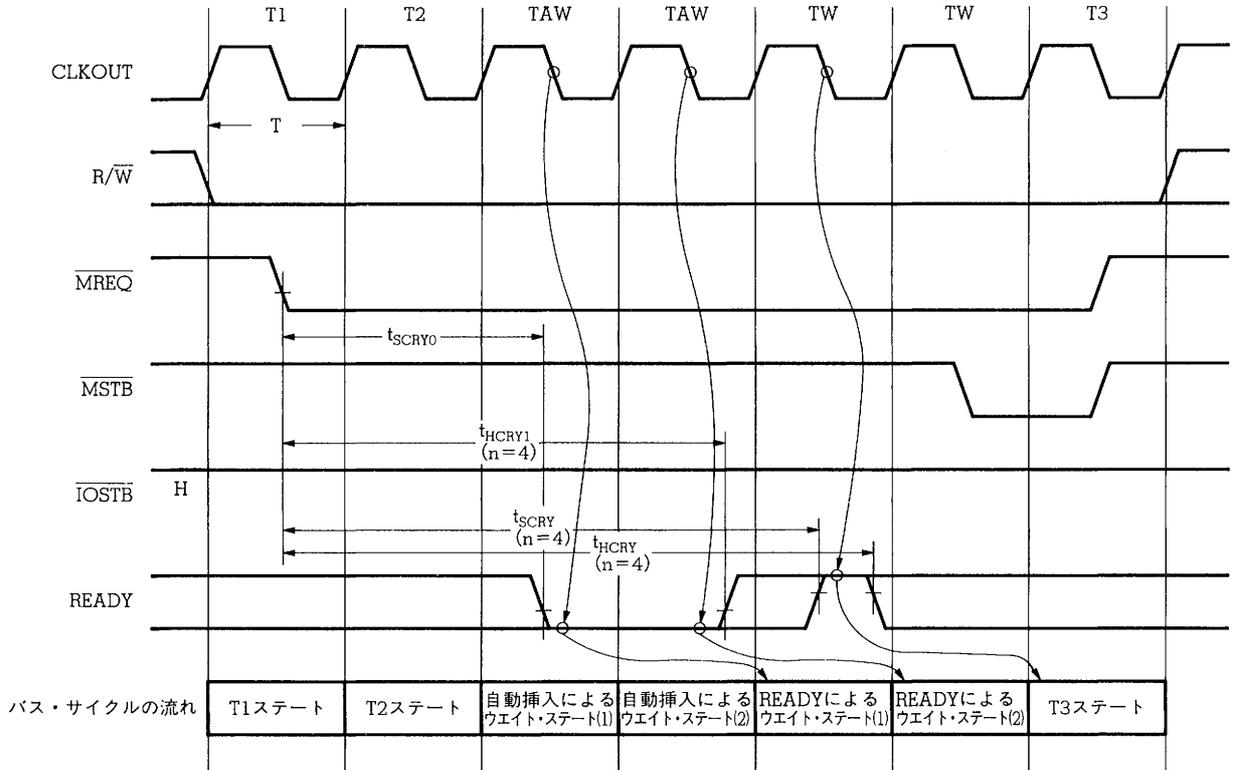
$T = 100 \text{ ns}$

$t_{SCRYO} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$

$t_{HCRYO} = 2T (\text{Min.}) = 200 \text{ ns}$

図 5-4 READY端子によるウェイト (μPD70335) (4/8)

(d) メモリ・ライト・サイクル 2 ウェイト追加



動作周波数10 MHzの場合

$T = 100 \text{ ns}$

合計のウェイト数は 4 ウェイトのため、nには 4 を代入します。

$t_{SCRY0} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$

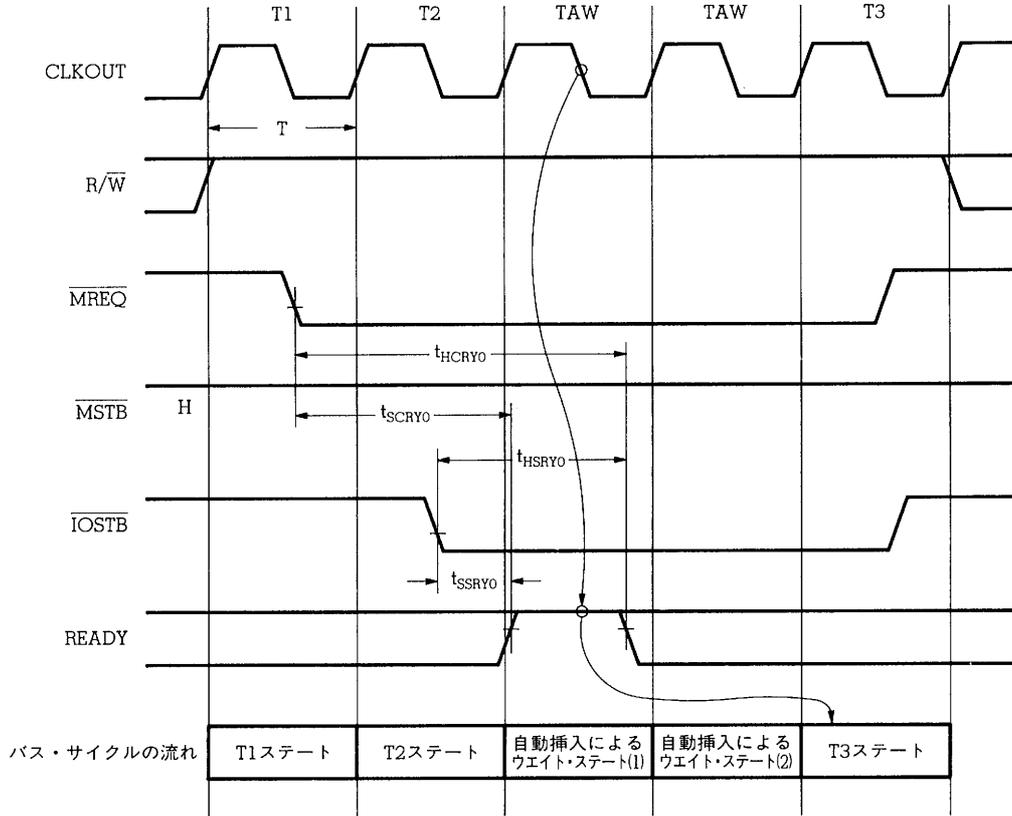
$t_{HCRY1} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$

$t_{SCRY} = nT - 80 (\text{Max.}) = 320 \text{ ns}$

$t_{HCRY} = nT (\text{Min.}) = 400 \text{ ns}$

図5-4 READY端子によるウェイト (μPD70335) (5/8)

(e) I/Oリード・サイクル ウェイト・ステート追加なし



動作周波数10 MHzの場合

$T = 100 \text{ ns}$

$t_{SCRYO} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$

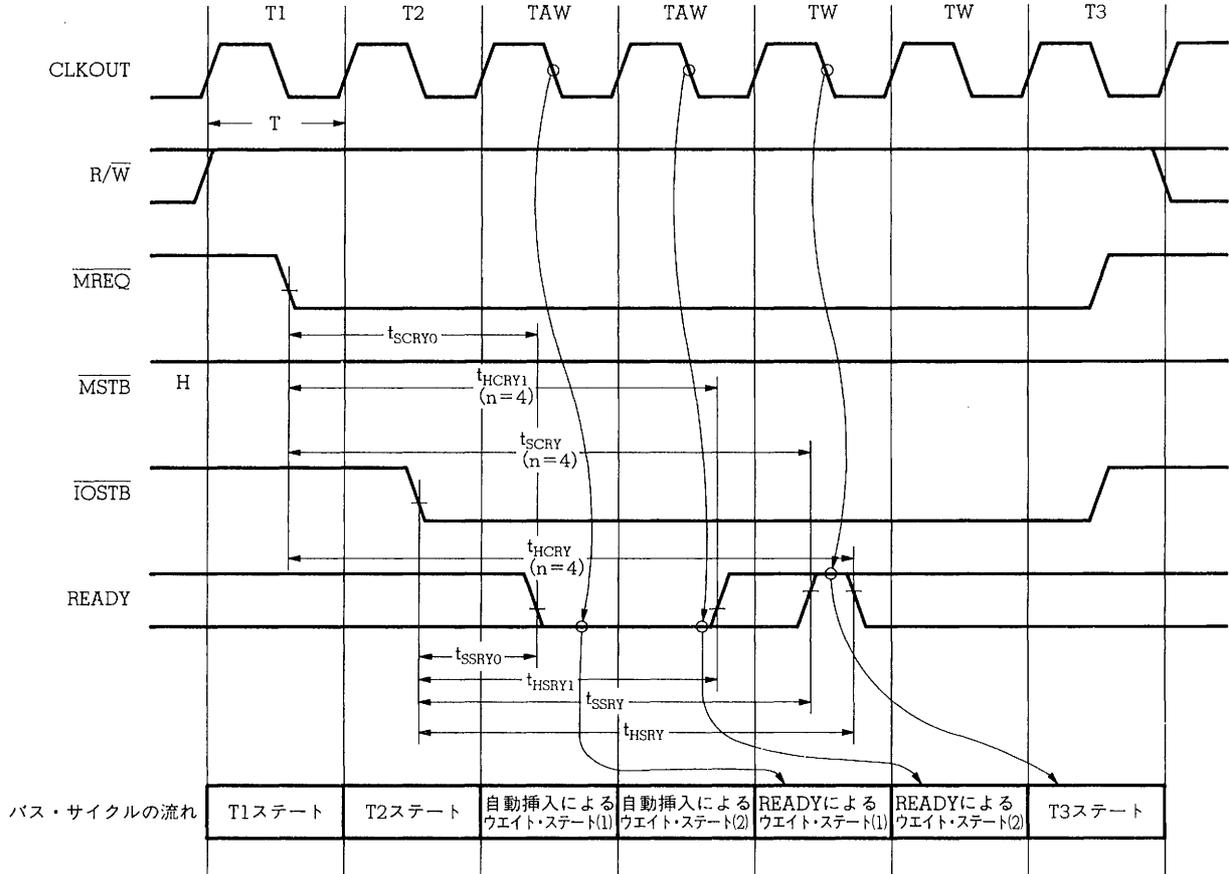
$t_{HCRYO} = 2T (\text{Min.}) = 200 \text{ ns}$

$t_{SSRYO} = T - 80 (\text{Max.}) = 20 \text{ ns}$

$t_{HSRYO} = T (\text{Min.}) = 100 \text{ ns}$

図5-4 READY端子によるウェイト (μPD70335) (6/8)

(f) I/Oリード・サイクル 2ウェイト追加



動作周波数10 MHzの場合

$T = 100 \text{ ns}$

合計のウェイト数は4ウェイトのため、nには4を代入します。

$t_{SCRY0} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$

$t_{HCRY1} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$

$t_{SCRY} = nT - 80 (\text{Max.}) = 320 \text{ ns}$

$t_{HCRY} = nT (\text{Min.}) = 400 \text{ ns}$

$t_{SSRY0} = T - 80 (\text{Max.}) = 20 \text{ ns}$

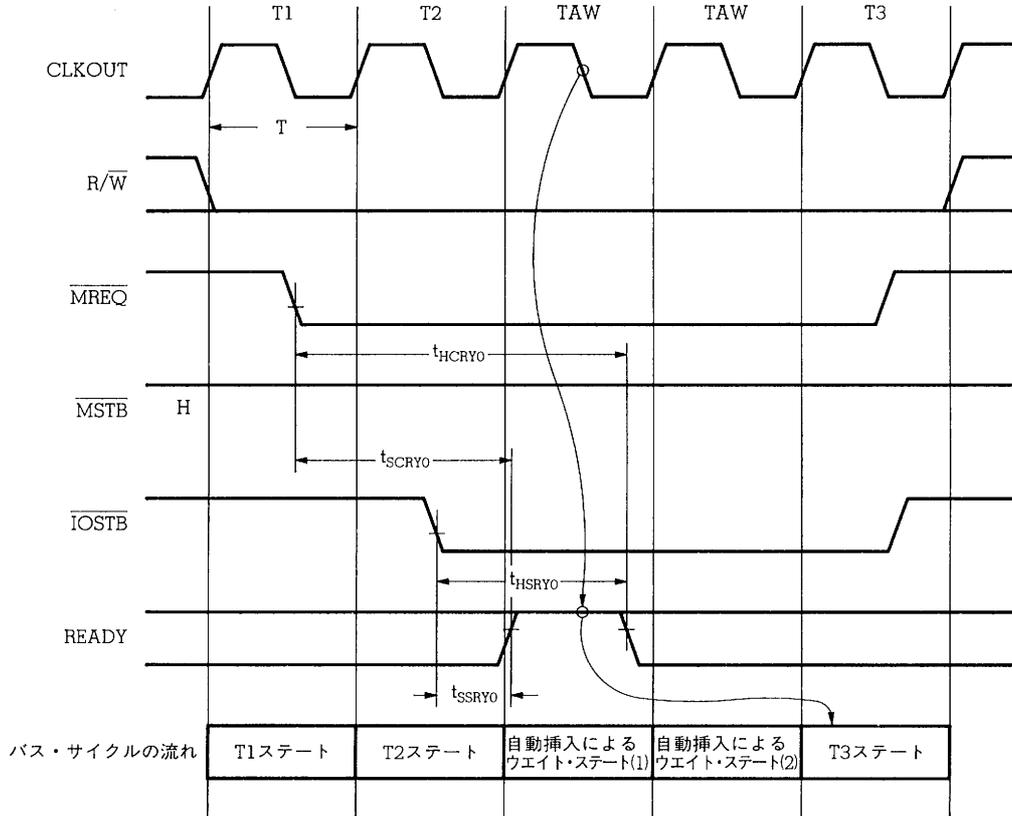
$t_{HSRY1} = (n - 2)T (\text{Min.}) = 200 \text{ ns}$

$t_{SSRY} = (n - 1)T - 80 (\text{Max.}) = 220 \text{ ns}$

$t_{HSRY} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$

図5-4 READY端子によるウェイト (μPD70335) (7/8)

(g) I/Oライト・サイクル・ウェイト・ステート追加なし



動作周波数10 MHzの場合

$T = 100 \text{ ns}$

$t_{SCRYO} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$

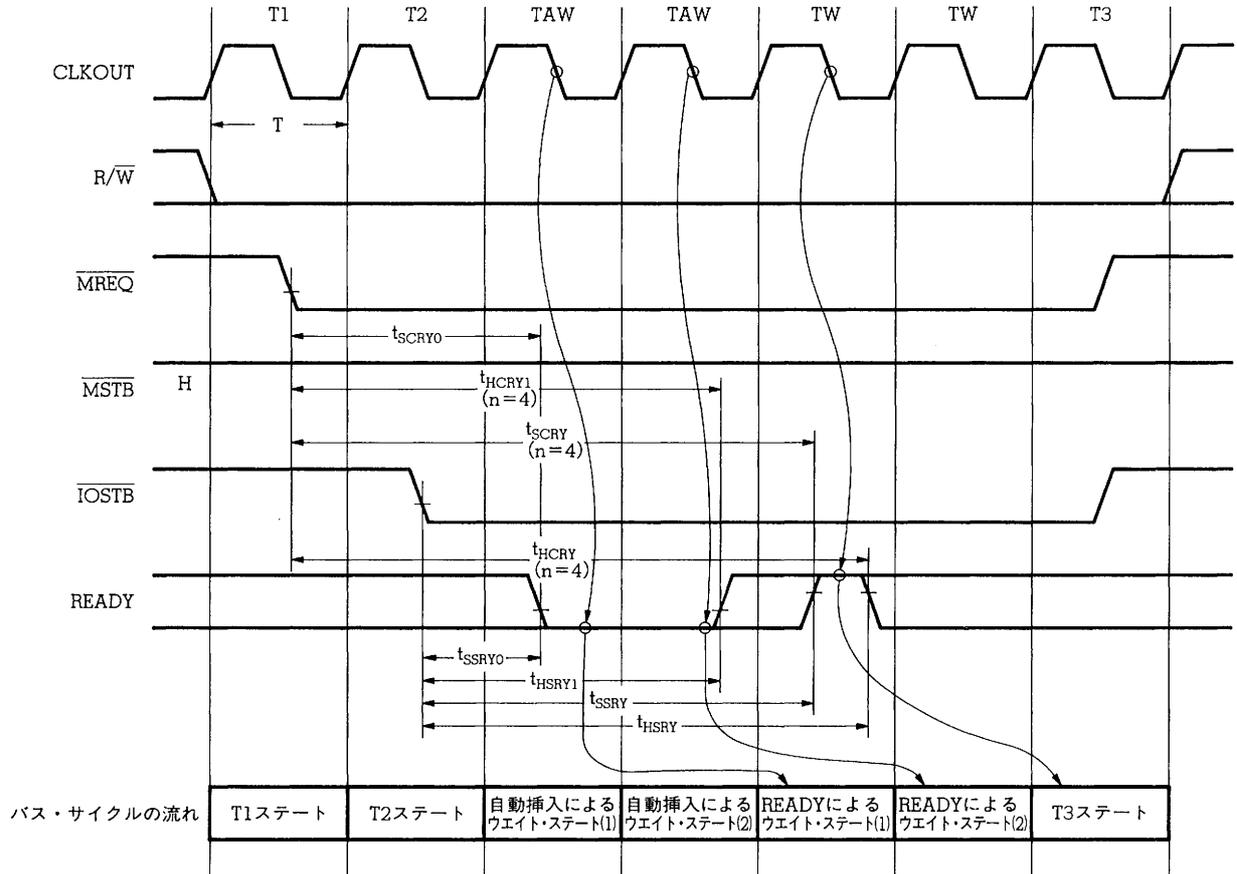
$t_{HCRYO} = 2T (\text{Min.}) = 200 \text{ ns}$

$t_{SSRYO} = T - 80 (\text{Max.}) = 20 \text{ ns}$

$t_{HSRYO} = T (\text{Min.}) = 100 \text{ ns}$

図5-4 READY端子によるウェイト (μPD70335) (8/8)

(h) I/Oライト・サイクル 2ウェイト追加



動作周波数10 MHzの場合

$T = 100 \text{ ns}$

合計のウェイト数は4ウェイトのため、nには4を代入します。

$t_{SCRY0} = 2T - 80 (\text{Max.}) = 120 \text{ ns}$

$t_{HCRY1} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$

$t_{SCRY} = nT - 80 (\text{Max.}) = 320 \text{ ns}$

$t_{HCRY} = nT (\text{Min.}) = 400 \text{ ns}$

$t_{SSRY0} = T - 80 (\text{Max.}) = 20 \text{ ns}$

$t_{HSRY1} = (n - 2)T (\text{Min.}) = 200 \text{ ns}$

$t_{SSRY} = (n - 1)T - 80 (\text{Max.}) = 220 \text{ ns}$

$t_{HSRY} = (n - 1)T (\text{Min.}) = 300 \text{ ns}$

5.2 バス・ホールド機能

μ PD70325, 70335は、バス・ホールドの機能を持っています。HLDRQ端子に外部からハイ・レベルを入力することにより、外部の素子がバスを使用したことを知らせます。 μ PD70325, 70335は、HLDRQ端子がハイ・レベルであることを検出すると、A0-A19, D0-D7 (D0-D15), $\overline{\text{REFRQ}}$, $\overline{\text{MREQ}}$, $\overline{\text{MSTB}}$, $\overline{\text{IOSTB}}$, $\overline{\text{R/W}}$ の各出力をハイ・インピーダンスにし、 $\overline{\text{HLDAK}}$ 端子をロウ・レベルにして外部の素子にバスを解放したことを示し、ホールド・モードに移行します。ホールド・モード中は、命令の実行、プリフェッチ割り込み受け付け等を停止し、バスを使用しないオンチップ周辺ハードウェアのみを動作させます。ホールド・モード中はHLDRQ端子をチェックし、ロウ・レベルを検出すると $\overline{\text{HLDAK}}$ 信号をハイ・レベルにし、外部の素子に対しバスを開放していないことを示し、1クロックおいてから実行を再開します。

HALTモード（12.2参照）中でも、バス・ホールドの要求は受け付け可能で、ホールド・モードが解除されると（HLDRQ信号がロウ・レベルであれば）HALTモードに戻ります。

リピート・プリフィクス付きのブロック転送命令実行中のバス・ホールド要求は、1回のバス・サイクルごとに受け付けることができます。★

また、BUSLOCK^注プリフィクスに続く1命令の実行中および割り込み受け付け動作中は、バス・ホールドの要求は受け付けません。

μ PD70325, 70335は、ホールド・モード時にメモリ・リフレッシュ・サイクルの挿入を行うことができます。リフレッシュ・モード・レジスタ（RFM）のHLDRF（ビット6）をセットすることにより行われます。リフレッシュ・タイミングごとに $\overline{\text{HLDAK}}$ 信号を強制的にハイ・レベルにし、HLDRQがロウ・レベルになったことを確認したのち、リフレッシュ・サイクルを行います。このあと、HLDRQ信号がハイ・レベルになっていれば、ホールド・モードへ再度移行します。また、このときHLDRQ信号がロウ・レベルのままなら、ホールド・モードが解除され命令の実行を再開します。

HLDRQ端子はP27と、 $\overline{\text{HLDAK}}$ 端子はP26とそれぞれ兼用となっていますので、バス・ホールド機能を使用するときはポート2モード・コントロール・レジスタ（PMC2）のビット6, 7を1にしてください。

注 BUSLOCK

REP

MOV BK

このようなプログラムでブロック処理命令実行中、バス・ホールドの要求を受け付けません。

5.2.1 HLD \overline{RQ} →HLD \overline{AK} 応答時間（単位：クロック）

次にHLD \overline{RQ} →HLD \overline{AK} 応答時間を示します。ただし、以下に示す場合を除きます。

- 外付けの割り込みコントローラによる割り込みアクノリッジ・サイクルが発生しているとき
- BUSLOCK命令が実行されているとき
- STOPモードのとき

	MIN.	MAX.
μ PD70325	3	7+2W
μ PD70335	3	6+W

W：ウェイト数

5.3 リフレッシュ機能

μ PD70325, 70335は、DRAM、疑似SRAMのリフレッシュを行うための各種の機能を持っています。一連のバス・サイクル中に定期的リフレッシュ・サイクルを挿入する機能、DRAM、疑似SRAMをリフレッシュするためのリフレッシュ・アドレスの出力、リフレッシュ・パルスの出力機能、疑似SRAMのパワー・ダウン・セルフ・リフレッシュ・モードをサポートする機能、ホールド・モード中、HALTモード中にリフレッシュ・サイクルを発生する機能、リフレッシュ・サイクル中にウェイト・ステートを挿入する機能があります。

5.3.1 リフレッシュ・モード・レジスタ（RFM）

RFMレジスタは、リフレッシュ機能の制御を行う8ビット・レジスタです。8ビット操作または1ビット操作のメモリ・アクセスによって、このレジスタの書き込み/読み出しができます。

\overline{RESET} 入力でRFMレジスタの内容はFCHに初期化されます。

RFMレジスタは、下図のように構成されます。以下に各ビットの機能を説明します。

7	6	5	4	3	2	1	0
RFLV	HLD \overline{RF}	HL \overline{TRF}	RFEN	RFW1	RFW0	RFT1	RFT0

RFT0 , **RFT1** リフレッシュの周期を指定するビット

リフレッシュの周期は、タイム・ベース・カウンタ（第10章参照）の出力タップ3-6から選択できます。リフレッシュ・サイクルは表5-5に示す周期の間隔で発生します。

表 5-5 リフレッシュの周期

 $f_{CLK}=8\text{ MHz}$ のとき

RFT	RFT	リフレッシュの周期
1	0	
0	0	$2^4/f_{CLK}$ (2.0 μs)
0	1	$2^5/f_{CLK}$ (4.0 μs)
1	0	$2^6/f_{CLK}$ (8.0 μs)
1	1	$2^7/f_{CLK}$ (16.0 μs)

RFW0 , **RFW1** リフレッシュ・サイクル中に挿入するウェイト・ステート数を指定するビット
リフレッシュ・サイクル中のウェイト・ステート数の指定は、前述したプログラマブル・ウェイトの機能 (5.1参照) によらず、表 5-6 に示すようにRFW0, 1によって行われます。

表 5-6 リフレッシュ・サイクル中のウェイト・ステート

RFW	RFW	ウェイト・ステート
1	0	
0	0	0 ステート
0	1	1 ステート
1	0	2 ステート
1	1	2 ステート

RFEN リフレッシュ・サイクルの自動挿入を許可するビット

1 のとき、リフレッシュ・サイクルの自動挿入を許可します。0 のときは、リフレッシュ・サイクルの自動挿入を禁止します。 $\overline{\text{REFRQ}}$ 端子の出力はRFLVビットの内容により制御されます(詳細はRFLVビットの説明を参照)。

HLTRF HALTモード中のリフレッシュ・サイクルの自動挿入を許可するビット

1 で許可, 0 で禁止となります。ただし, RFENビット=0の場合はHLTRFビットの内容にかかわらず禁止となります。

HLDRF ホールド・モード中のリフレッシュ・サイクルの自動挿入を許可するビット

1 で許可, 0 で禁止となります。許可状態(1)のときには、リフレッシュ・タイミングごとに $\overline{\text{HLDK}}$ 出力を強制的にハイ・レベルに立ち上げ、リフレッシュ・サイクルを自動挿入します。

RFLV $\overline{\text{REFRQ}}$ 信号の出力レベルを指定するビット

図 5-5 のような回路構成になっており、表 5-7 に示す論理で出力が決定されます。

RFLVビットは、読み出し時はマスターRFLVの出力となり、書き込み時はスレーブRFLVに書き込まれます。また、マスターRFLVへの書き込みはリフレッシュ・タイミングの発生時に行われます。

RFLVビットを使用することにより、疑似SRAMのパワー・ダウン・セルフ・リフレッシュ・モードのサポートが可能です。

図 5-5 RFLVビットによる制御回路

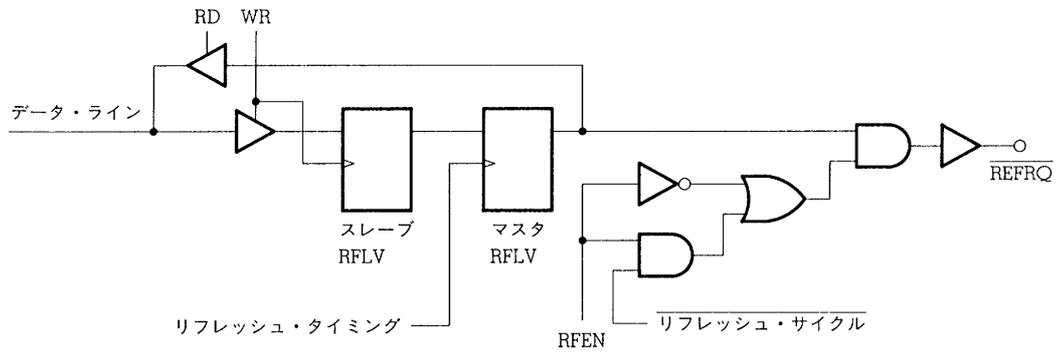


表 5-7 $\overline{\text{REFRQ}}$ 信号の出力レベル

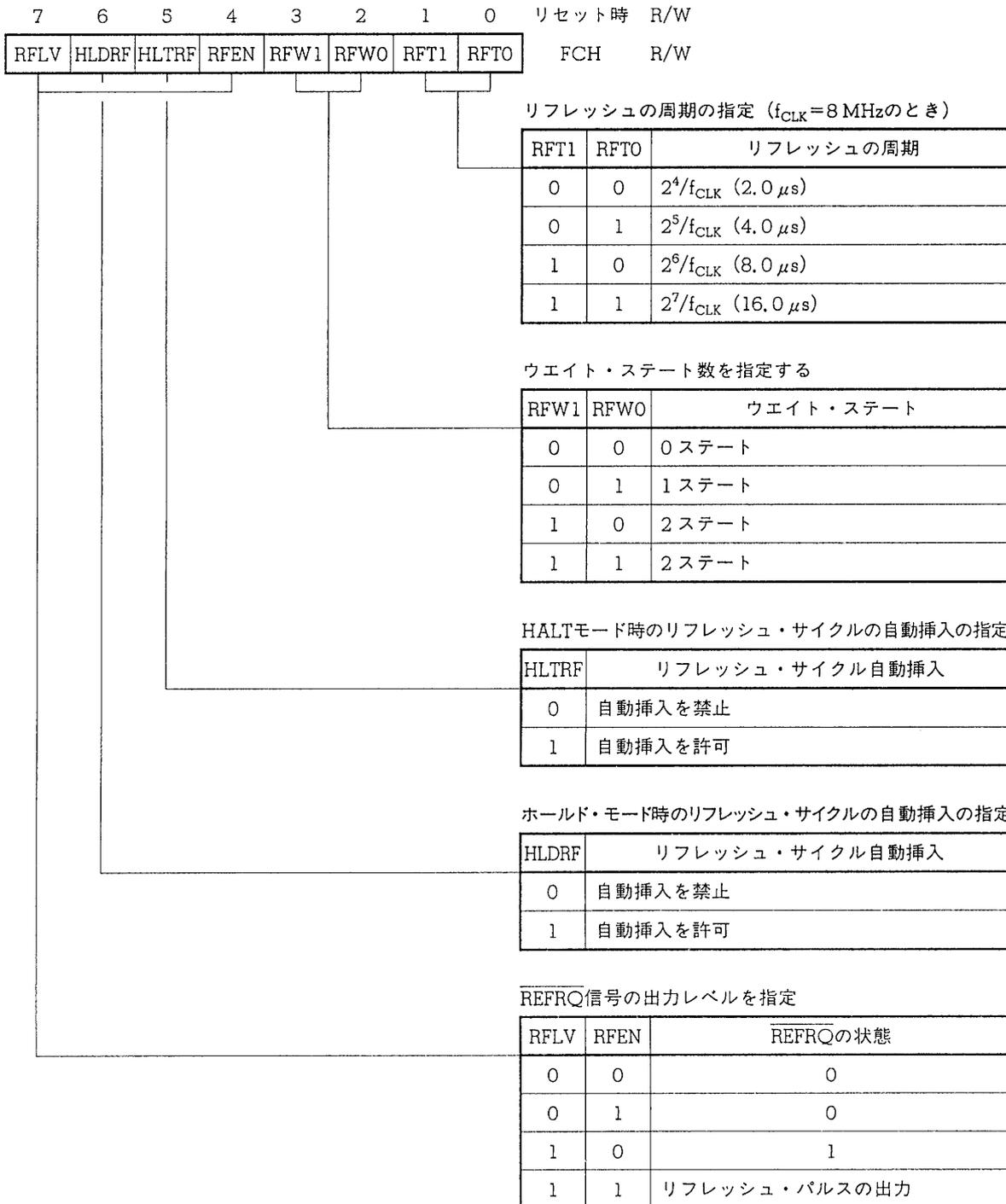
RFLV	RFEN	$\overline{\text{REFRQ}}$ の状態
0	0	0
0	1	0
1	0	1
1	1	リフレッシュ・パルス出力

リフレッシュ・サイクルの挿入は、RFENビットが1のときにリフレッシュのタイミングになると行われます。このとき $\overline{\text{MREQ}}$ 、 $\overline{\text{MSTB}}$ 、 $\overline{\text{IOSTB}}$ はハイ・レベルとなり、A0-A8にリフレッシュ・アドレスをA9-A18にロウ・レベルを出力し($\mu\text{PD70325}$ の場合です。 $\mu\text{PD70335}$ については表 5-3 参照。), リフレッシュ・パルスを $\overline{\text{REFRQ}}$ 端子から出力します。

なお、RFLVビットは書き込んでも、次のリフレッシュ・タイミングまで読み出しデータとはなりません。

図5-6 RFM

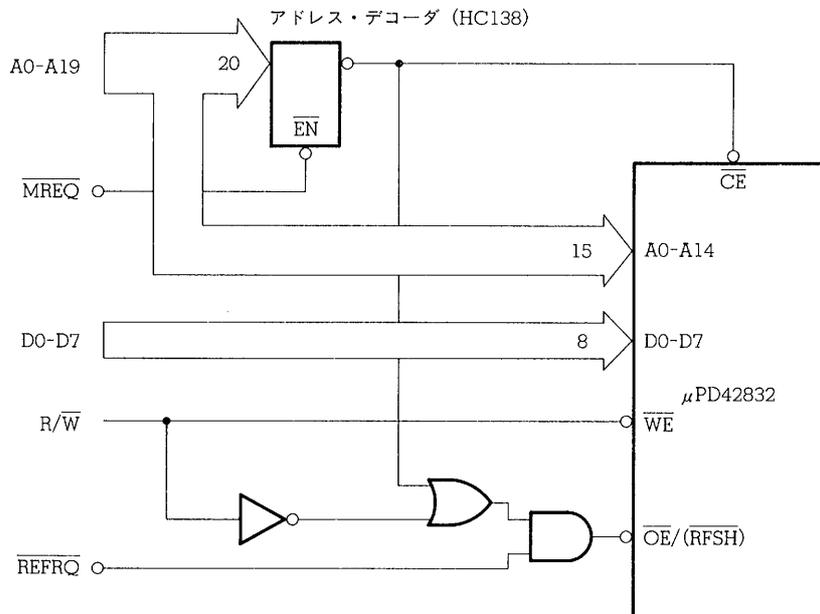
★



5.3.2 疑似SRAMとの接続

μ PD42832相当の疑似SRAMを接続する回路例を図5-7に示します。

図5-7 μ PD42832接続回路例



この接続例では、 \overline{CE} 端子が“ハイ・レベル”のときに $\overline{OE}/(\overline{RFSH})$ 端子に \overline{REFRQ} 端子からパルスを与えるパルス・リフレッシュ・モードと、ソフトウェア上でリフレッシュ・モード・レジスタ (RFM) のビット7 (RFLV) をリセット(0)することにより \overline{REFRQ} 端子を“ロウ・レベル”にするパワー・ダウン・セルフ・リフレッシュ・モードが使用できます。

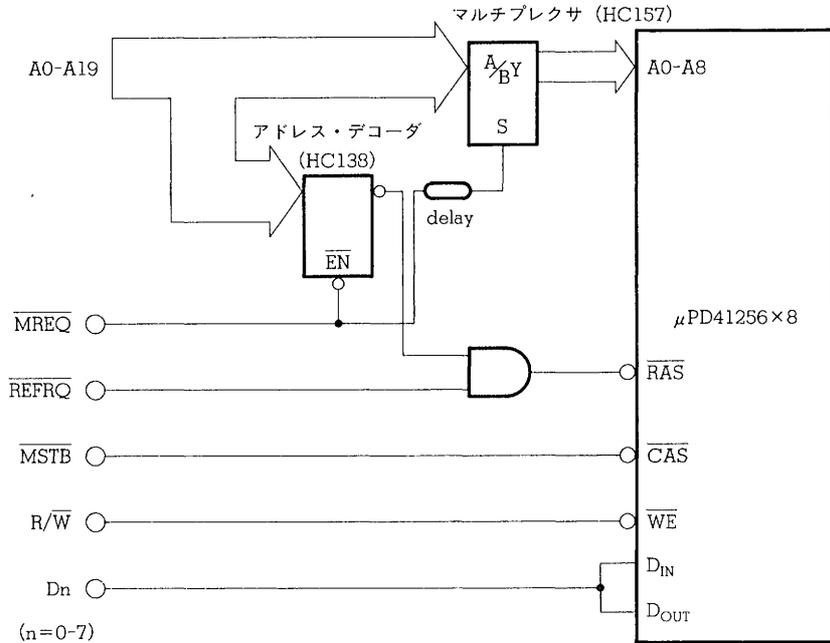
パワー・ダウン・セルフ・リフレッシュは、CPUがスタンバイ (STOP) 状態でパルス・リフレッシュ動作をできないときに使用します。したがって、スタンバイ状態に入る直前にRFLVビットをリセット(0)します。また、スタンバイ状態から復帰した場合は、パルス・リフレッシュ動作ができるようにRFLVビットをセット(1)します。

注意 リセット中 (\overline{RESET} 端子 = “0”) は \overline{REFRQ} 端子がハイ・インピーダンス状態になります。

5.3.3 DRAMとの接続

256 K×1ビット構成のμPD41256を接続する回路例を図5-8に示します。

図5-8 μPD41256接続回路例



この接続例では、 $\overline{\text{REFRQ}}$ 端子から出力されるパルスに同期して、アドレス・バスに出力される9ビットのリフレッシュ・アドレスを使用する $\overline{\text{RAS}}$ オンリー・リフレッシュにより、リフレッシュ動作を行っています。

5.4 バス使用权

μPD70325, 70335のバスの使用权の優先順位は次の順になります。

(1) リフレッシュ・サイクル (5.3参照)

リフレッシュ・サイクルの挿入が許可されていると、リフレッシュ・サイクルが必ず発生します。ただし、ホールド・モード中は、 $\overline{\text{HLDAK}}$ 信号を強制的にハイ・レベルにし、 HLDRQ 信号がロウ・レベルになるのを待ってリフレッシュ・サイクルを実行します。なお、READY端子によるウェイト・サイクルが挿入されている間は、リフレッシュ・サイクルは起動しません。

(2) ホールド・モード (5.2参照)

BUSLOCKプリフィクスに続く一命令の実行中、および割り込みアクノリッジ・サイクル中以外は、ホールド・モードに移行します。

(3) DMAサイクル (第6章参照)

(4) 他のバス・サイクル

ただし、 $\overline{\text{INTAK}}$ サイクルを実行中は、リフレッシュ・サイクル、ホールド・モード、DMAサイクルは一時的に保留されます。DMAについては、内部割り込みに対する割り込み受け付け動作中も保留されます。割り込み受け付けに要する時間は、4.17 ハードウェア割り込みの応答時間を参照してください。

なお、ストップ・モード時はバスは動作しません (バスの状態については表12-2参照)。

ブロック転送、DMA転送、マクロ・サービスなどのデータ転送は、同一プログラム上で同時に使用できますが、単一のバス上でデータ転送が行われるため、同時に実行されることはありません。

5.5 バス・タイミング

図5-9～図5-30までに主なバスのタイミングを示します (DMAを除く)。

バス・アクセスを行っていないときは、制御端子はインアクティブになり、データ・バス、アドレス・バスの出力は不定です。

5.5.1 μ PD70325

図5-9 メモリ・リード・サイクル

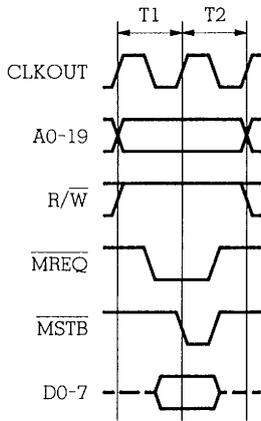


図5-10 メモリ・ライト・サイクル

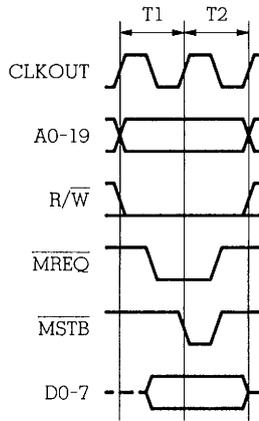


図5-11 I/Oリード・サイクル

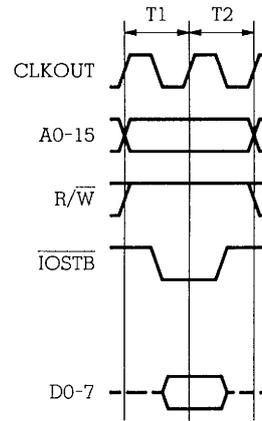


図5-12 I/Oライト・サイクル

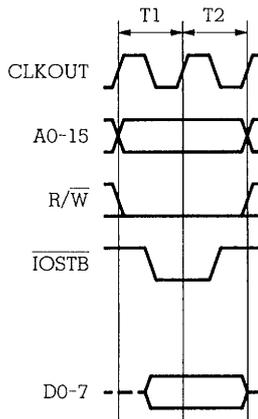


図5-13 メモリ・リード・サイクル

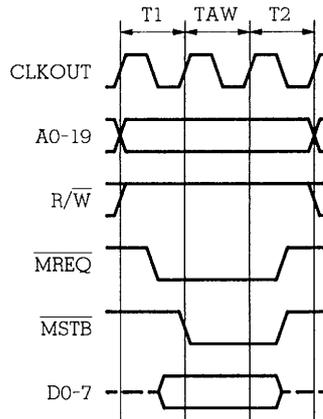
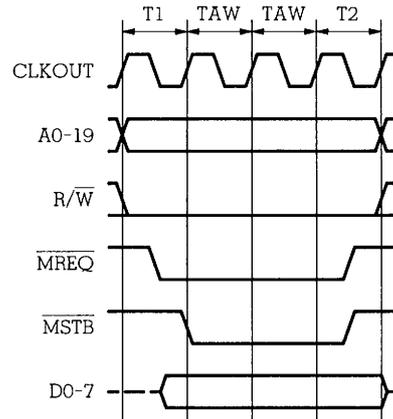


図5-14 メモリ・ライト・サイクル

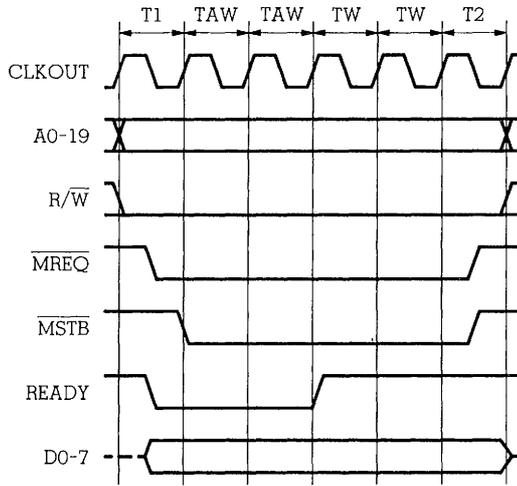


(1ウェイト・ステート挿入時)

(2ウェイト・ステート挿入時)

備考 破線はハイ・インピーダンスを示します。

図 5-15 メモリ・ライト・サイクル
(READY端子による操作時)



備考 破線はハイ・インピーダンスを示します。

図 5-16 リフレッシュ・サイクル
(1 ウェイト・ステート挿入時)

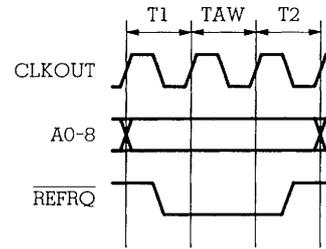
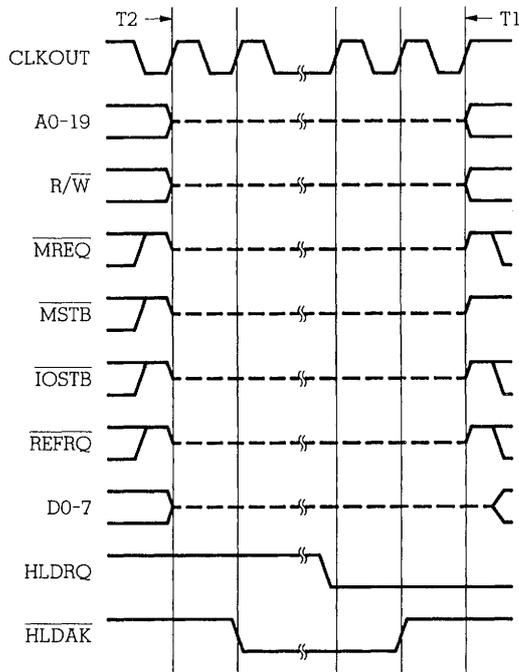
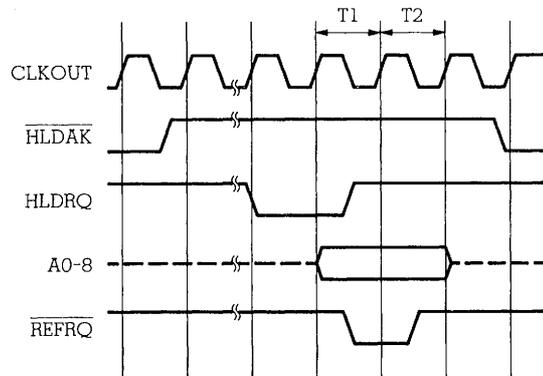


図 5-17 バス・ホールド受け付け解除タイミング



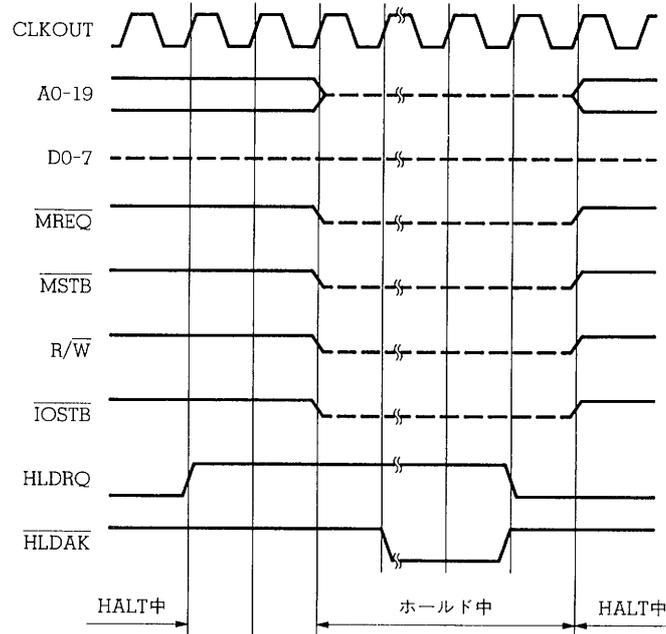
備考 破線はハイ・インピーダンスを示します。

図 5-18 ホールド・モード時のリフレッシュ・サイクル
(ウェイト・ステート数 0)



備考 破線はハイ・インピーダンスを示します。

図5-19 HALT中のバス・ホールド受け付け解除タイミング



備考 破線はハイ・インピーダンスを示します。

5.5.2 μ PD70335

図 5-20 メモリ・リード・サイクル

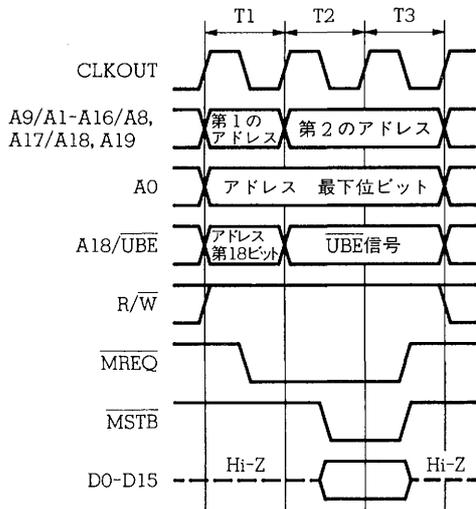


図 5-21 メモリ・ライト・サイクル

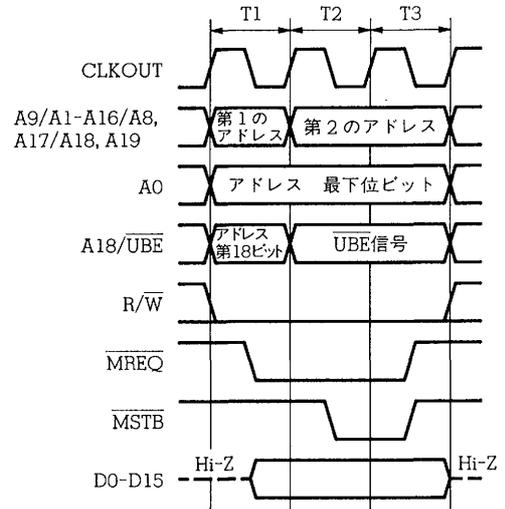


図 5-22 I/Oリード・サイクル

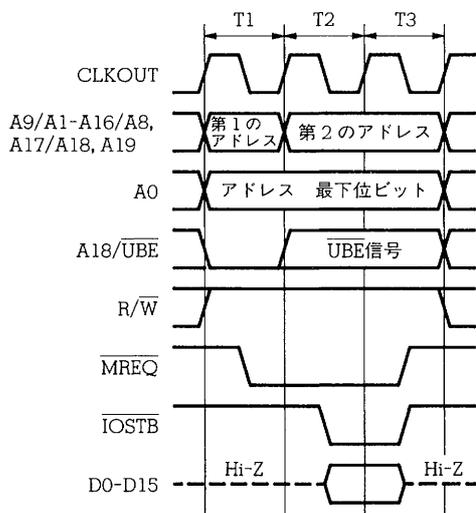


図 5-23 I/Oライト・サイクル

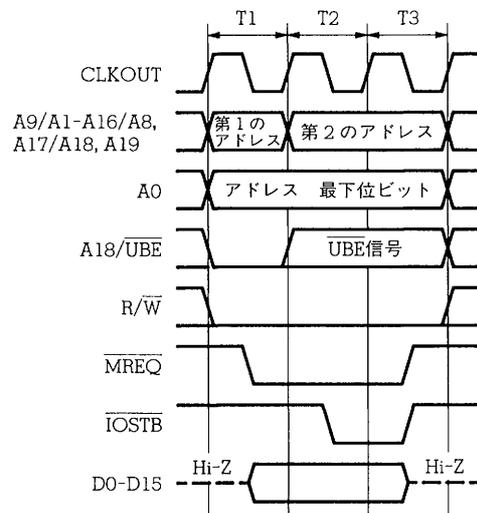


図 5-24 メモリ・リード・サイクル (1 ウェイト・ステート挿入時)

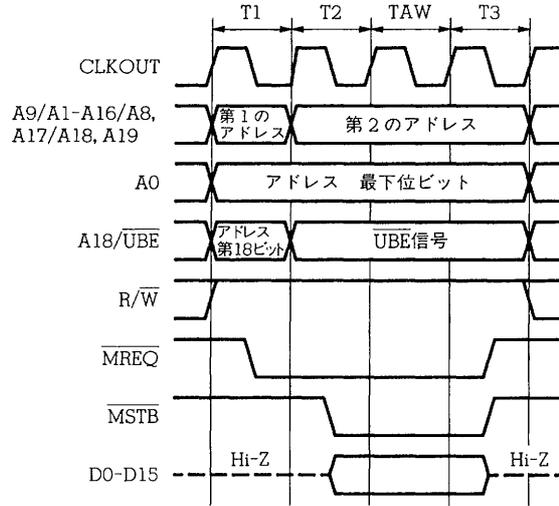


図 5-25 メモリ・ライト・サイクル (2 ウェイト・ステート挿入時)

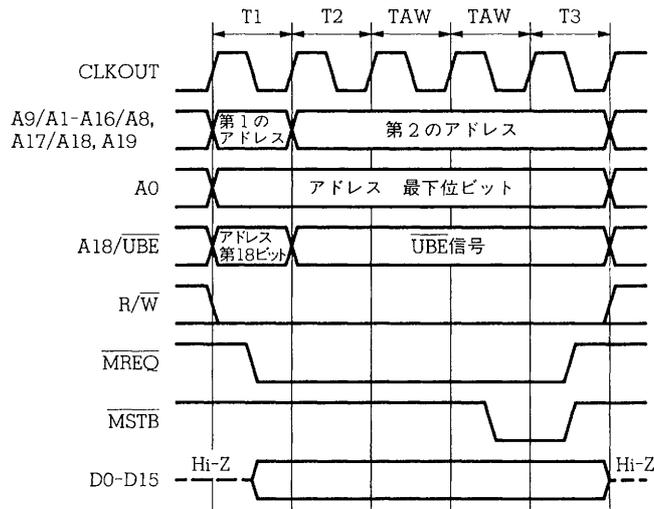


図5-26 メモリ・ライト・サイクル (READY端子による操作時)

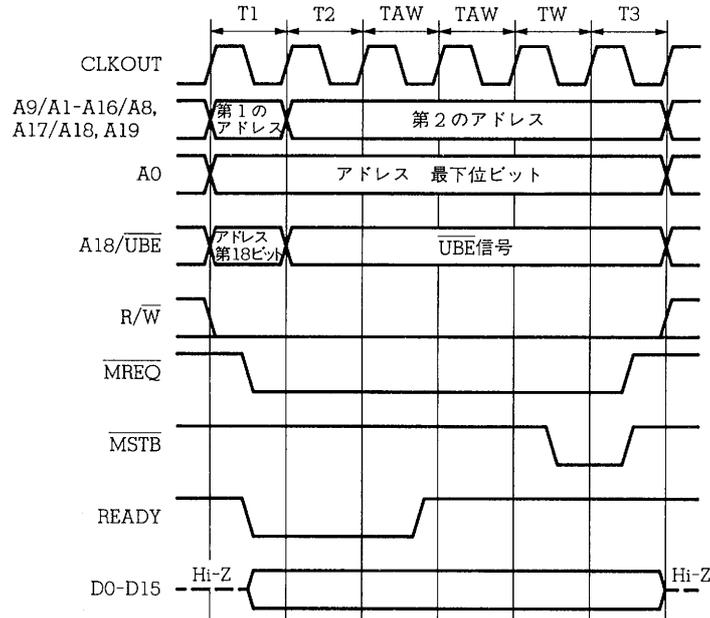
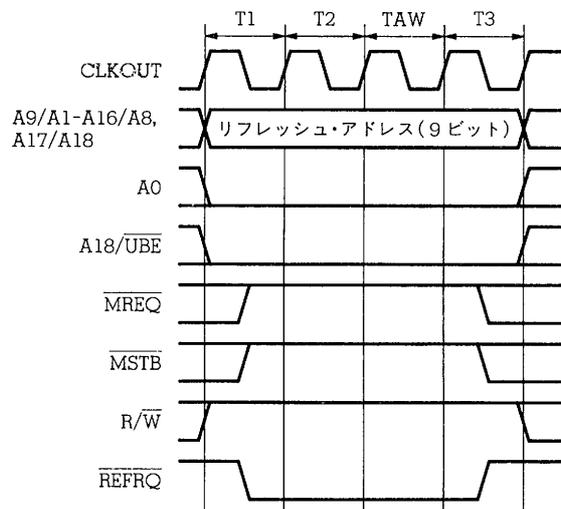


図5-27 リフレッシュ・サイクル (1ウエイト・ステート挿入時)



★

図 5-28 バス・ホールド受け付け解除タイミング

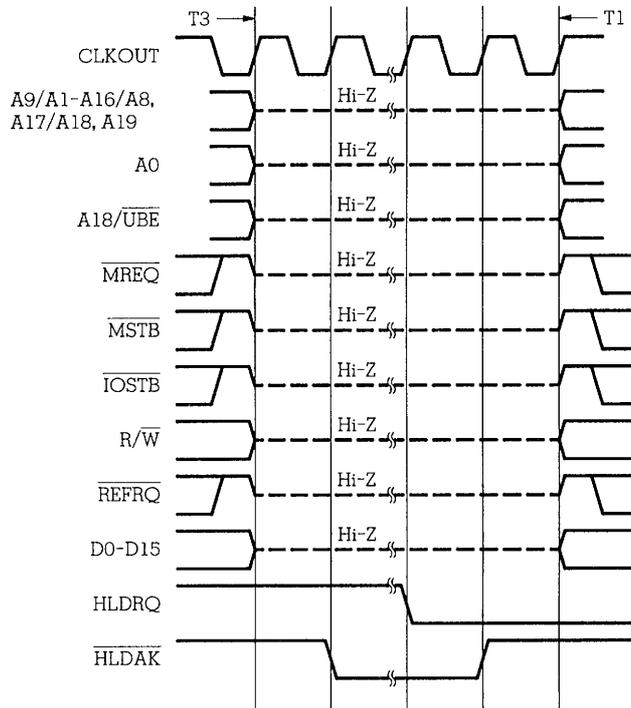
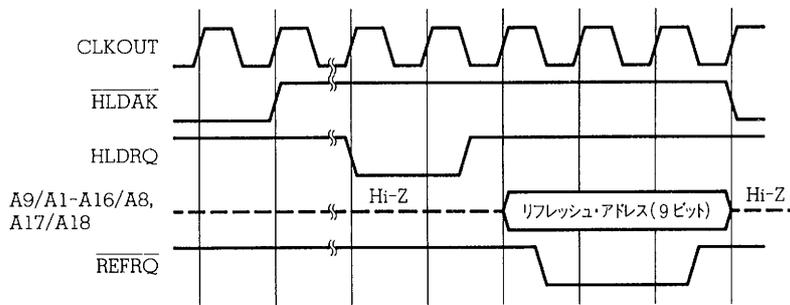
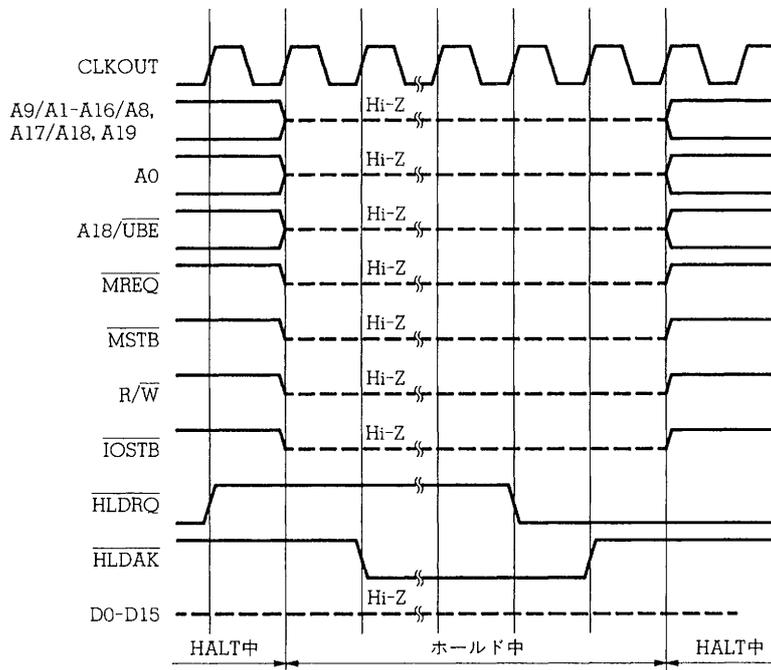


図 5-29 ホールド・モード時のリフレッシュ・サイクル



★

図 5-30 HALT中のバス・ホールド受け付け解除タイミング



第6章 DMAコントローラ

μ PD70325, 70335は、1 Mバイトのメモリ空間を直接指定することのできる2チャンネルのDMAコントローラを内蔵しています。

6.1 端子機能

DMAコントローラのために次に示す端子が用意されています。これらの端子はすべてポートと兼用になっているため、使用する場合は該当するポート2モード・コントロール・レジスタ (PMC2) のビットを1にする必要があります。

(1) $\overline{\text{DMARQ0}}$, $\overline{\text{DMARQ1}}$ (P20, P23)

アクティブ・ハイのDMA要求入力端子。

ディマンド・リリース・モードで、 $\overline{\text{DMARQn}}$ 端子によって、DMA転送中の停止制御をするためには、DMA転送の対象となる空間に対して2ステートのウェイト・ステートが挿入されるように設定してください。また、 $\overline{\text{DMAAKn}}$ の立ち下がりから1クロック以内に $\overline{\text{DMARQn}}$ をロウ・レベルにしてください。

(2) $\overline{\text{DMAAK0}}$, $\overline{\text{DMAAK1}}$ (P21, P24)

アクティブ・ロウのDMA応答出力端子。

ただし、メモリ-メモリ間のDMA転送 (バースト・モード, シングルステップ・モード) 時には出力しません。

(3) $\overline{\text{TC0}}$, $\overline{\text{TC1}}$ (P22, P25)

アクティブ・ロウのDMA完了出力端子。

$\overline{\text{TC0}}$, $\overline{\text{TC1}}$ のデクリメントの際にポローが発生したときに出力します。

6.2 DMA動作

μ PD70325, 70335には4種のDMA転送モードを持っています。各転送モードの機能等を表6-1に示します。

DMA転送では、内部データ領域はアクセスできません。内部データ領域に相当するアドレスをアクセスしたときは、同じアドレスの外部メモリがアクセスの対象となります。

表 6-1 各転送モードの機能

モード名	転送対象	機 能	DMA起動	停止方法	割り込み	HALT中	DMA動作中のDMA要求
シングル ステップ	メモリ -メモリ	1回のDMA要求で命令の1バス・サイクルの実行と1DMA転送を指定回数だけ交互に繰り返す。	<ul style="list-style-type: none"> DMARQの立ち上がり DMA制御レジスタのTDMAビットのセット 	ソフトウェアによる(EDMAビットのクリア)。	すべて受け付ける。	指定回数のDMA転送を連続して行う。	チャンネル1のDMAは保留または中断され、チャンネル0のDMAを行う。
バースト	メモリ -メモリ	指定回数のDMA転送を1回のDMA要求で連続して行う。	<ul style="list-style-type: none"> DMARQの立ち上がり DMA制御レジスタのTDMAビットのセット 	なし	<ul style="list-style-type: none"> DMA転送中は受け付けない。 	指定回数のDMA転送を連続して行う。	DMAの転送が終了するまで他のDMAは保留される。
1転送	メモリ -I/O	DMA要求の発生する度に1回のDMA転送を行う。	<ul style="list-style-type: none"> DMARQの立ち上がり 	ソフトウェアによる(EDMAビットのクリア)。	すべて受け付ける。	通常と同じ	1DMA転送後、要求のあったDMAを行う。
ダイヤモンド・リリース	メモリ -I/O	DMARQ端子がハイ・レベルの期間中DMA転送を行う。	<ul style="list-style-type: none"> DMARQのハイ・レベル 	<ul style="list-style-type: none"> DMA転送中はロウ・レベルで停止 その他の場合はソフトウェアによる(EDMAビットのクリア)。 	<ul style="list-style-type: none"> DMA転送中は受け付けない。 DMA転送中以外はすべて受け付ける。 	通常と同じ	チャンネル1のDMAは保留または中断され、チャンネル0のDMAを行う。

メモリ-メモリ間のDMA転送では、 $\overline{\text{DMAAK}}$ 信号は出力されません。メモリ-I/O間のDMA転送では、1DMAサイクルごとに $\overline{\text{DMAAK}}$ 信号が出力されます。DMA転送中にI/Oをアクセスする場合は、 $\overline{\text{IOSTB}}$ 信号の代わりに $\overline{\text{DMAAK}}$ 信号を使用してください。★

DMA転送中でもプログラマブル・ウェイト機能(5.1参照)は有効です。メモリ-メモリ間では、転送先、転送元ごとに指定したウェイト・ステートが挿入されます。メモリ-I/O間では、1回のバス・サイクルで1転送を済ませるため、メモリとI/Oとで遅い方のウェイト・ステートが挿入されます。

また、DMA転送中でもバス・ホールド機能とリフレッシュ機能は有効で、DMA転送はこれらによって一時的に中断されます。

リピート・プリフィクス付きブロック処理(転送、比較、検索入出力)命令実行中でもDMA転送は、その要求があれば行われます。このとき、ブロック処理命令の実行は一時的に中断されます。

さらに、BUSLOCKプリフィクスを付加した場合も同様にDMA転送が行われます。

DMA転送中の割り込みは受け付けられず、すべて保留されます。

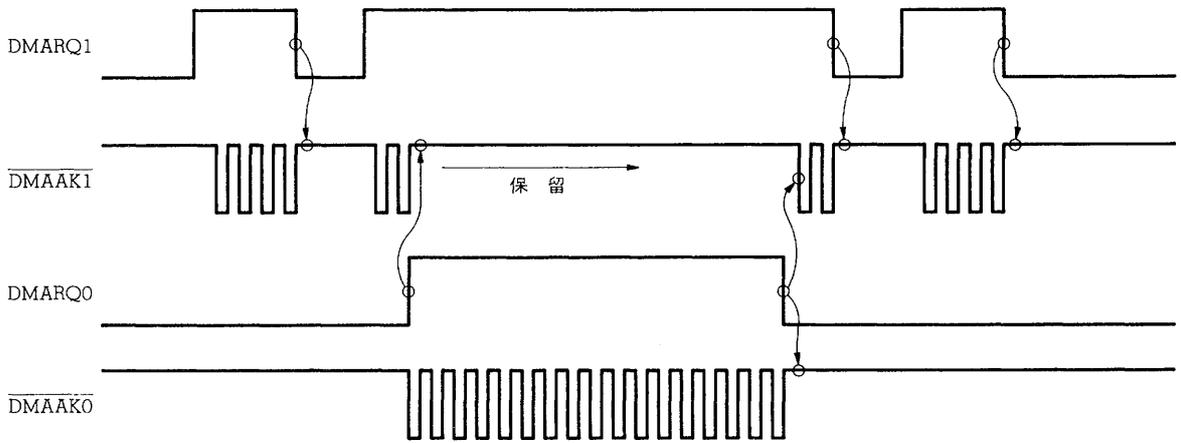
HALTモード中のDMA転送はその要求があれば行われます。DMA転送が終了するとHALTモードに戻ります。HALTモードに戻った時点でDMA転送完了割り込みが発生していれば、HALTモードは解除されます。

同時にDMAの要求が発生した場合にはチャンネル0が優先されます。

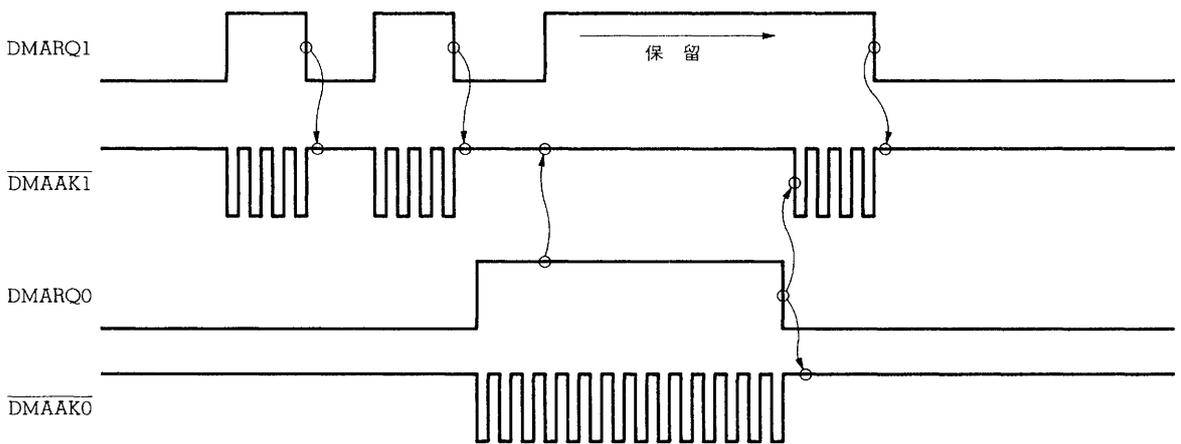
指定回数分のDMA転送が終了したときの動作を次に示します。★

- ・TC端子にロウ・レベルが出力されます。
- ・EDMAビットがクリア(0)されます。
- ・DMA完了割り込み要求が発生します。

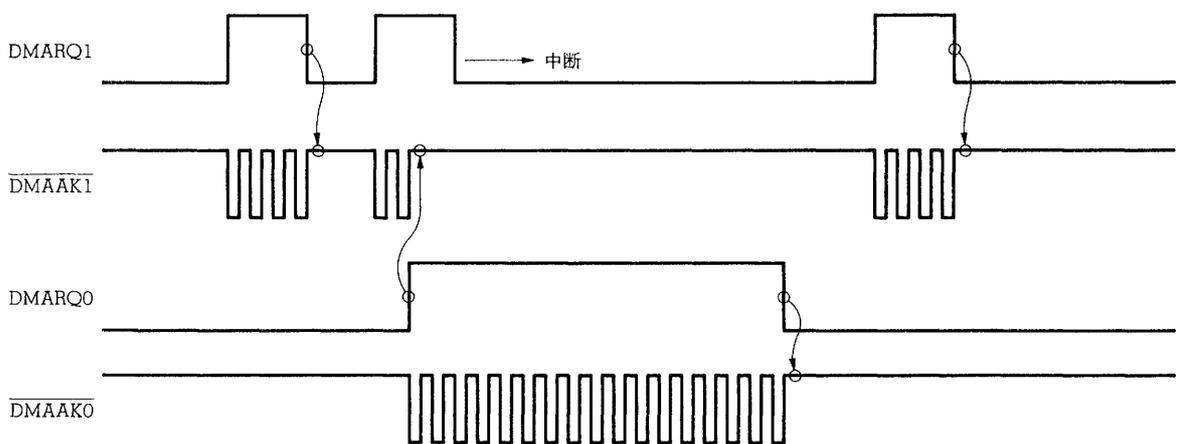
ダイヤモンド・リリース・モード時：DMA要求による保留の例 1



ダイヤモンド・リリース・モード時：DMA要求による保留の例 2



ダイヤモンド・リリース・モード時：DMA要求による中断の例



μ PD70325のダイヤモンド・リリース・モードでのDMA転送レート t は、次のようになります。

$$t = f_{\text{CLK}} / (2 + w) \quad [\text{バイト/秒}] \quad f_{\text{CLK}} : \text{システム・クロック [Hz]}$$

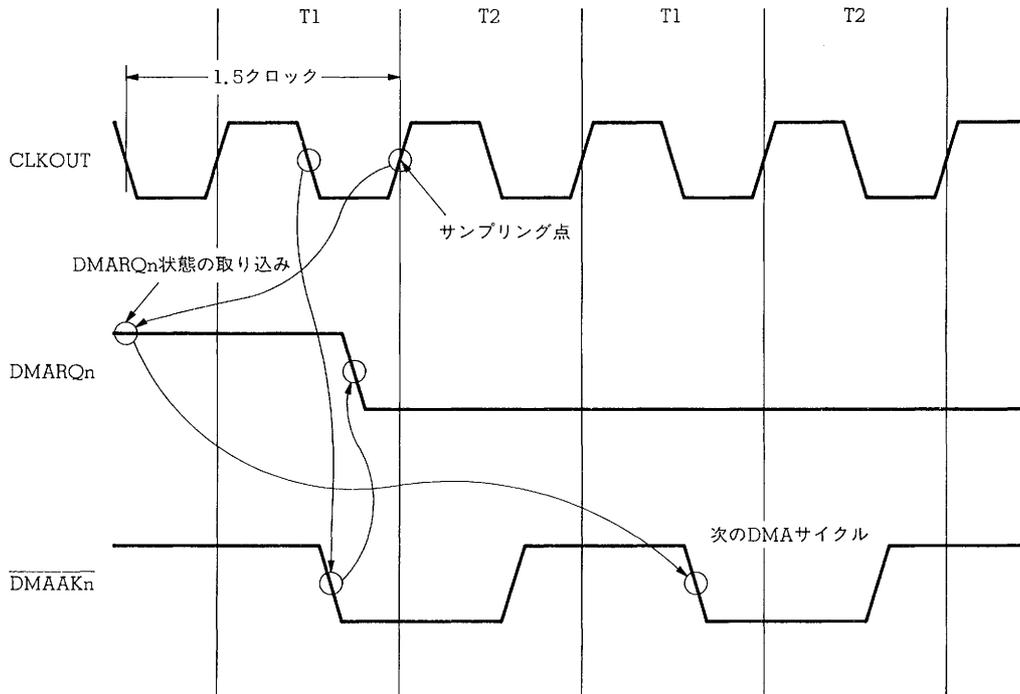
w : 1DMAサイクル当たりのウェイト数

また、 μ PD70335では、次のようになります。

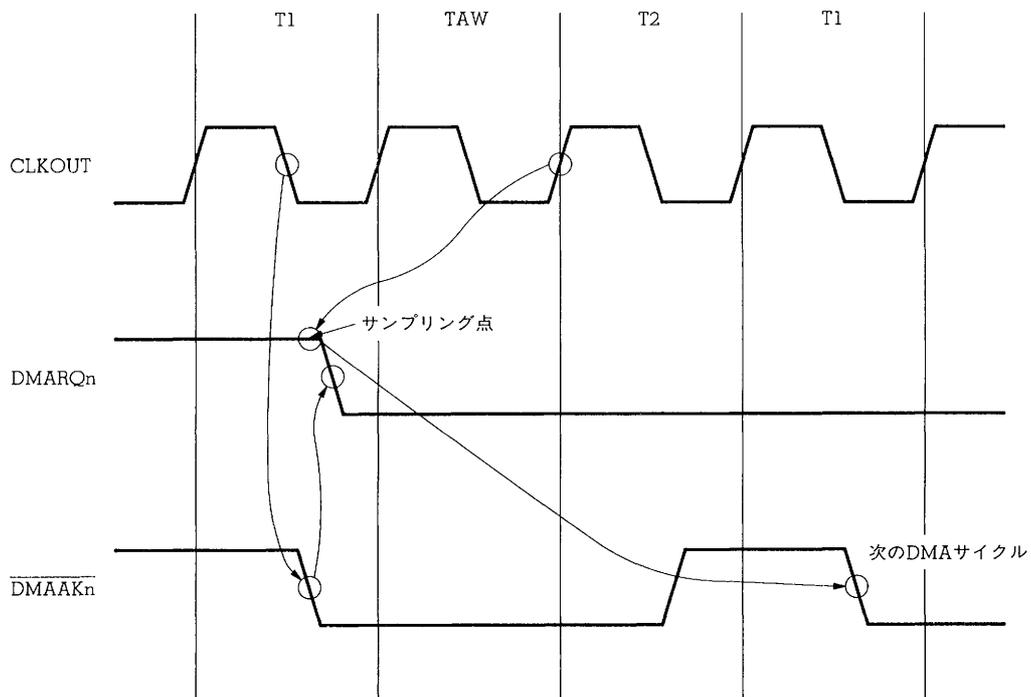
$$t = f_{\text{CLK}} / (3 + w) \quad [\text{バイト/秒}]$$

次に、このモードの動作タイミングを示します。

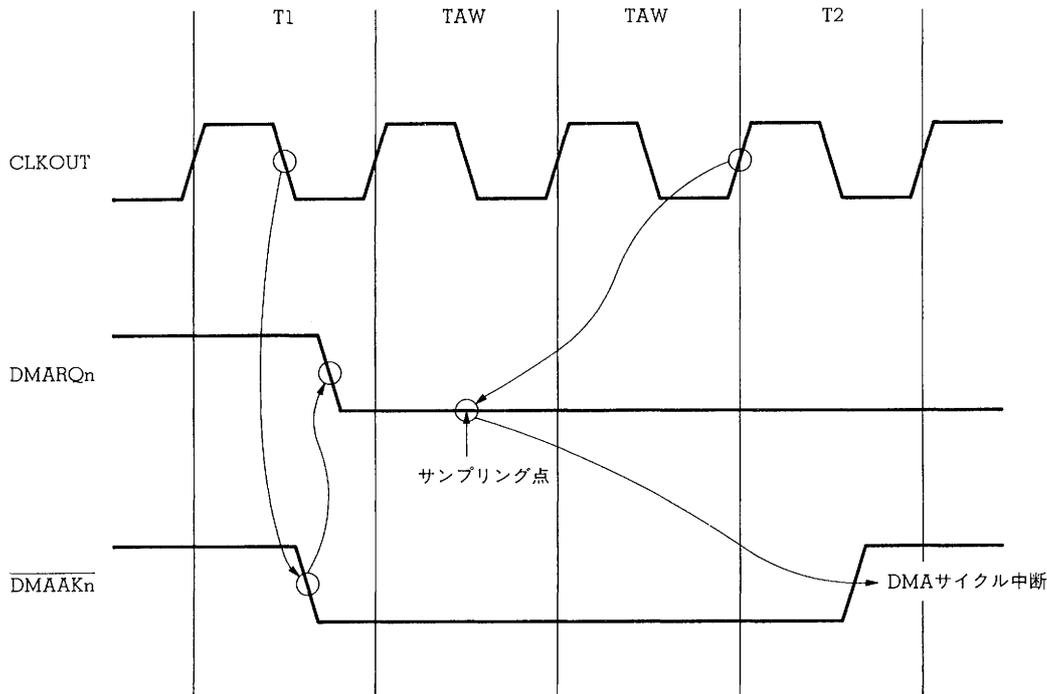
(1) ノー・ウェイトの場合 (μ PD70325)



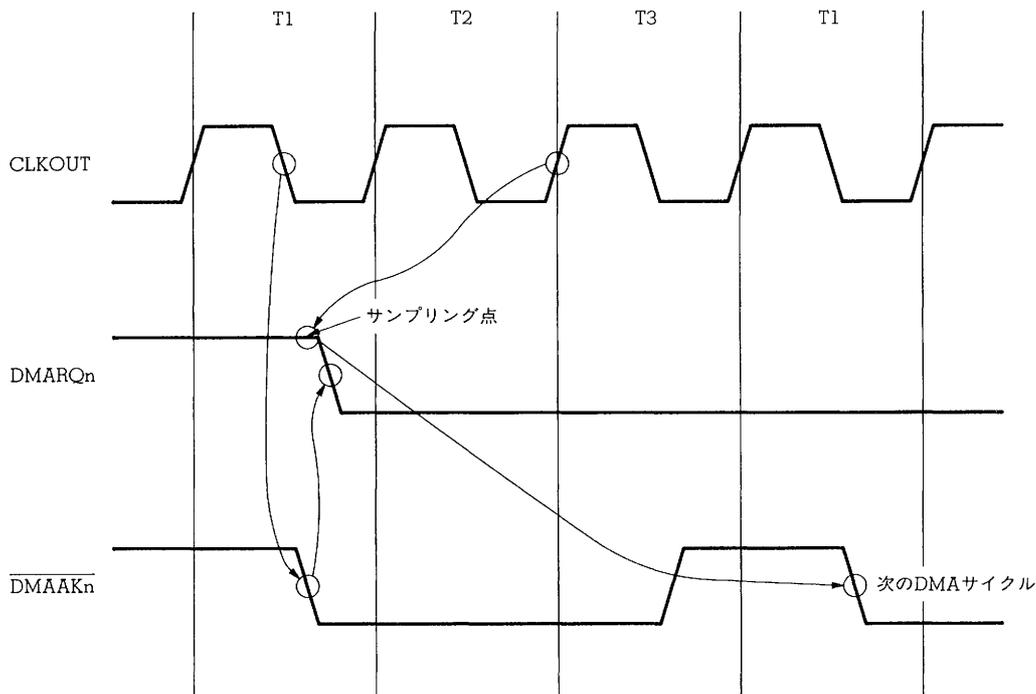
(2) 1ウェイトの場合 (μ PD70325)

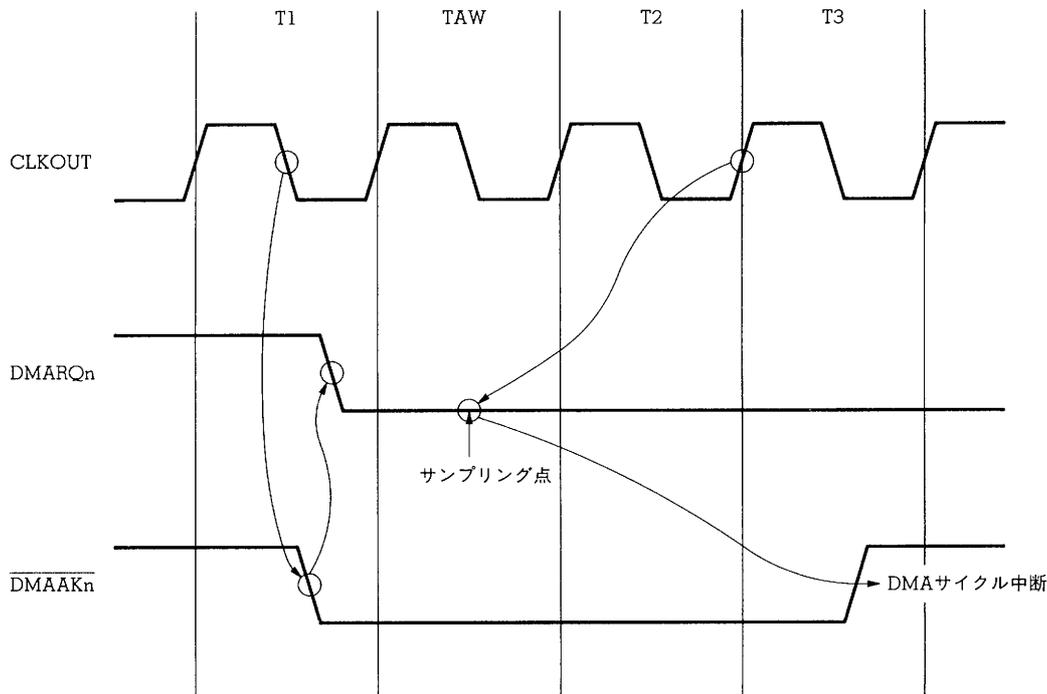


(3) 2ウェイトの場合 (μ PD70325)



(4) ノー・ウェイトの場合 (μ PD70335)



(5) 2 ウェイトの場合 (μ PD70335)

ディマンド・リリース・モードにおいて、DMAコントローラはDMARQn入力をバス・サイクルごとにサンプリングし、その状態によって、続いてDMAサイクルを起動するか、または、DMAサイクルを中断するかを決定します。

また、DMARQn入力はレベル・センスであり、DMARQnによるDMA要求の受け付けはDMAAKn出力によって外部に示されるため、DMARQn入力はDMAAKnの立ち下がりまで保持する必要があります。

ところが、 μ PD70325の場合、DMARQnの状態の取り込みはT2サイクルの1.5クロック前で行われるため、ノー・ウェイト、1ウェイトでは、いったん立ち上げたDMARQnをその次のDMAサイクルのためのサンプリング以前に確実に立ち下げることができません ((1), (2)参照)。

したがって、通常、ディマンド・リリース・モードを使用する場合、転送対象となるメモリまたはI/Oのいずれかに2状態以上のウェイト・サイクルを挿入する必要があります ((3)参照)。

同様に、 μ PD70335の場合、T3サイクルの1.5クロック前でDMARQnの状態の取り込みが行われるため、転送対象のメモリ、あるいは、I/Oに1ウェイト以上の挿入が必要です ((4), (5)参照)。

ただし、いったん立ち上げたDMARQnを一連のDMA転送の途中で操作せず、TCn出力(ターミナル・カウント発生)まで保持することによりTCnレジスタにあらかじめ設定した回数分のDMA転送を行わせる場合、ウェイト数1または0のより高速なDMA転送を設定することができます(メモリ-I/O間のバースト・モードに相当)。

6.3 DMAの制御のためのレジスタ

DMAの転送モードの指定などを行うためにDMAモード・レジスタ、DMAコントロール・レジスタがあります。そして、割り込みを制御するためのレジスタがあります。これらのレジスタはすべて各チャンネルごとに用意されています。

6.3.1 DMAモード・レジスタ (DMAMO, DMAM1)

DMAの転送モードなどを指定する8ビット・レジスタです。8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み/読み出しができます。DMAMOはチャンネル0に、DMAM1はチャンネル1にそれぞれ対応します。

RESET入力でDMAM_nレジスタの内容は00Hに初期化されます。

7	6	5	4	3	2	1	0
MD2	MD1	MD0	W	EDMA	TDMA	0	0

MD2 , **MD1** , **MD0** 転送モードを指定するビット

MD2	MD1	MD0	転送モード
0	0	0	シングルステップ・モード
0	0	1	ダイヤモンド・リリース・モード (I/O→メモリ)
0	1	0	ダイヤモンド・リリース・モード (メモリ→I/O)
0	1	1	使用禁止
1	0	0	バースト・モード
1	0	1	1転送モード (I/O→メモリ)
1	1	0	1転送モード (メモリ→I/O)
1	1	1	使用禁止

W 転送処理をバイトで行うか、ワードで行うかを指定するビット

0でバイト転送, 1でワード転送となります。

μPD70325では必ずバイト転送 (W=0) を指定してください。

μPD70335のワード転送では、ソースとデスティネーションの初期アドレスはどちらも偶数を設定してください。

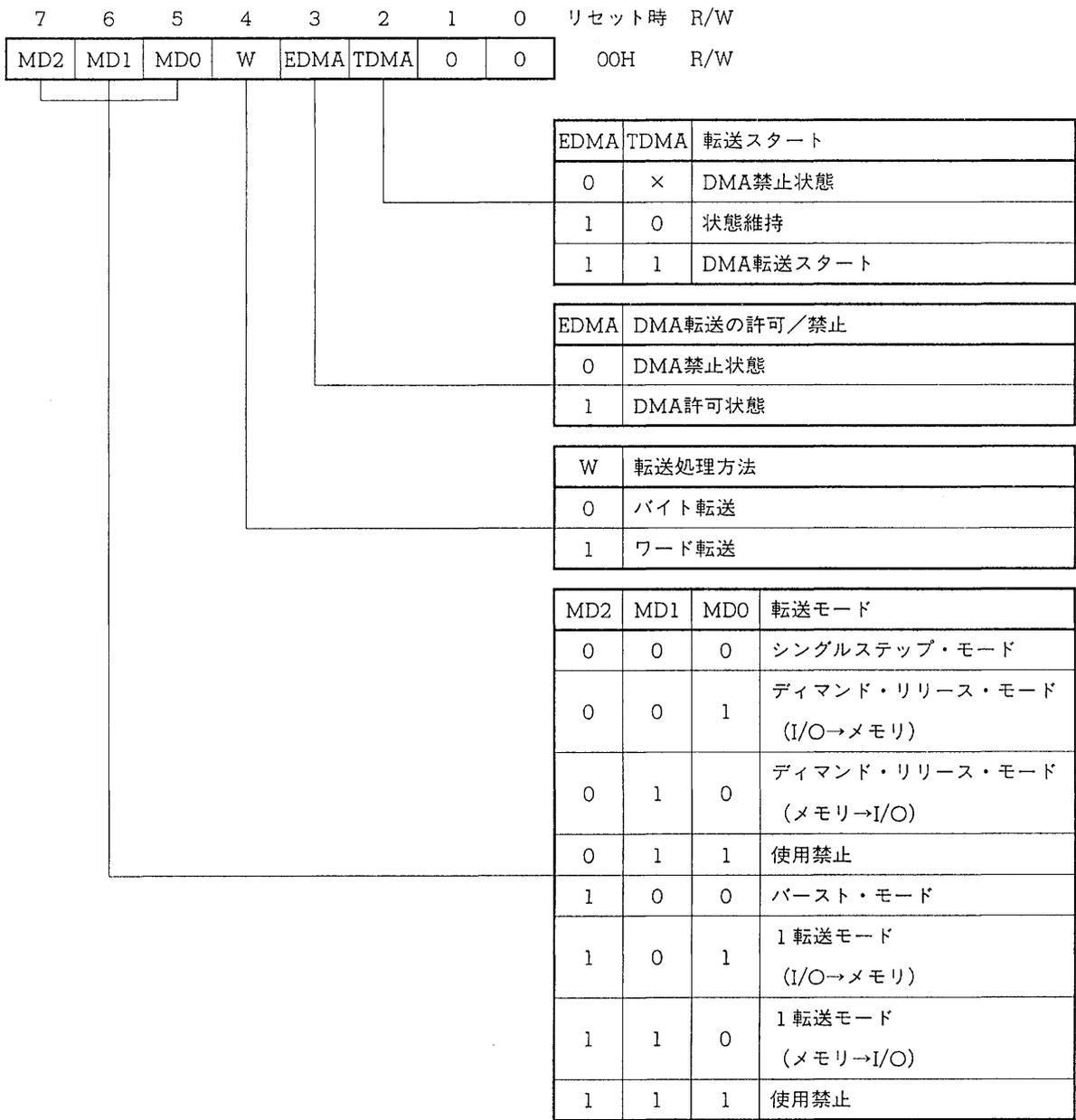
EDMA DMA転送の許可, 禁止を指定するビット

1で許可, 0で禁止となります。0 (禁止) の場合のDMA要求は保留されず無視されます。なお、このビットはDMAサービス・チャンネルのターミナル・カウンタ (TC) がFFFFHになると自動的にクリア(0)されます。

TDMA 転送スタート・ビット

シングルステップ・モードまたはバースト・モードの場合にのみ有効で、このビットに1を書き込むことによりDMAが起動されます(ただし、EDMAがセット(1)されている場合)。このビットの読み出しレベルは常に0です。また、ダイヤモンド・リリース・モード、1転送モードでは意味を持ちません。

図 6-1 DMAMO, DMAM1



6.3.2 DMAコントロール・レジスタ (DMACO, DMAC1)

DMA転送におけるソース・アドレス、デスティネーション・アドレスの更新方法を指定する8ビット・レジスタです。8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み／読み出しができます。

$\overline{\text{RESET}}$ 入力してもDMACnレジスタの内容は保持されます。

図6-2に示すようにDMACnレジスタのビット1, 0(PS1, PSO)は、ソース側のアドレスのオフセット値の更新モードを指定します。ビット5, 4(PD1, PDO)は、デスティネーション側のアドレスのオフセット値の更新モードを指定します。

図6-2 DMACO, DMAC1



注 パワーオン・リセット時：不定

6.3.3 DMA割り込み要求制御レジスタ (DICO, DIC1)

DMA転送の完了によって発生する割り込みの制御のための8ビット・レジスタです。割り込みはターミナル・カウンタ (TC)=FFFFHとなったときに発生します。

8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み/読み出しができます。この場合、1ウエイトが挿入されます。

$\overline{\text{RESET}}$ 入力でDIC_nレジスタの内容は47Hに初期化されます。

これらの割り込みは、マクロ・サービス機能をサポートしません。また、チャンネル0 (INTD0) とチャンネル1 (INTD1) のDMA転送完了の割り込みは1つのグループを形成し、割り込みの優先順位はチャンネル0の方が高くなっています。INTD0の制御はDICOレジスタで行い、ベクタは20になります。また、INTD1の制御はDIC1で行い、ベクタは21になります (3.5.5参照)。

(割り込みの優先順位) DF0>DF1

図 6-3 DICO, DIC1

7	6	5	4	3	2	1	0
DF0	DMK0	0	ENCS	0	PR2	PR1	PRO
DF1	DMK1	0	ENCS	0	1	1	1

注意 DIC1レジスタのビット2-0は“1”に固定されます。

DIC1レジスタの割り込み要求の優先順位は、DICOレジスタのPR2-0ビットの設定に従います。

DF0/DF1ビットは、DMA転送完了の割り込み要求フラグで、DMK0/DMK1ビットは、DMA転送完了割り込みのマスク・ビットです。

その他のビット・フィールドについては4.7項を参照してください。

6.4 DMAサービス・チャンネル

μ PD70325, 70335では、DMA転送の転送元、転送先、転送回数を指定するために、ソース・アドレス・ポインタ、デスティネーション・アドレス・ポインタおよびターミナル・カウンタを特殊機能レジスタ内に、各チャンネルごとに用意しています。これにより一連のDMA処理において最大64 Kバイト分のデータ転送が可能です。

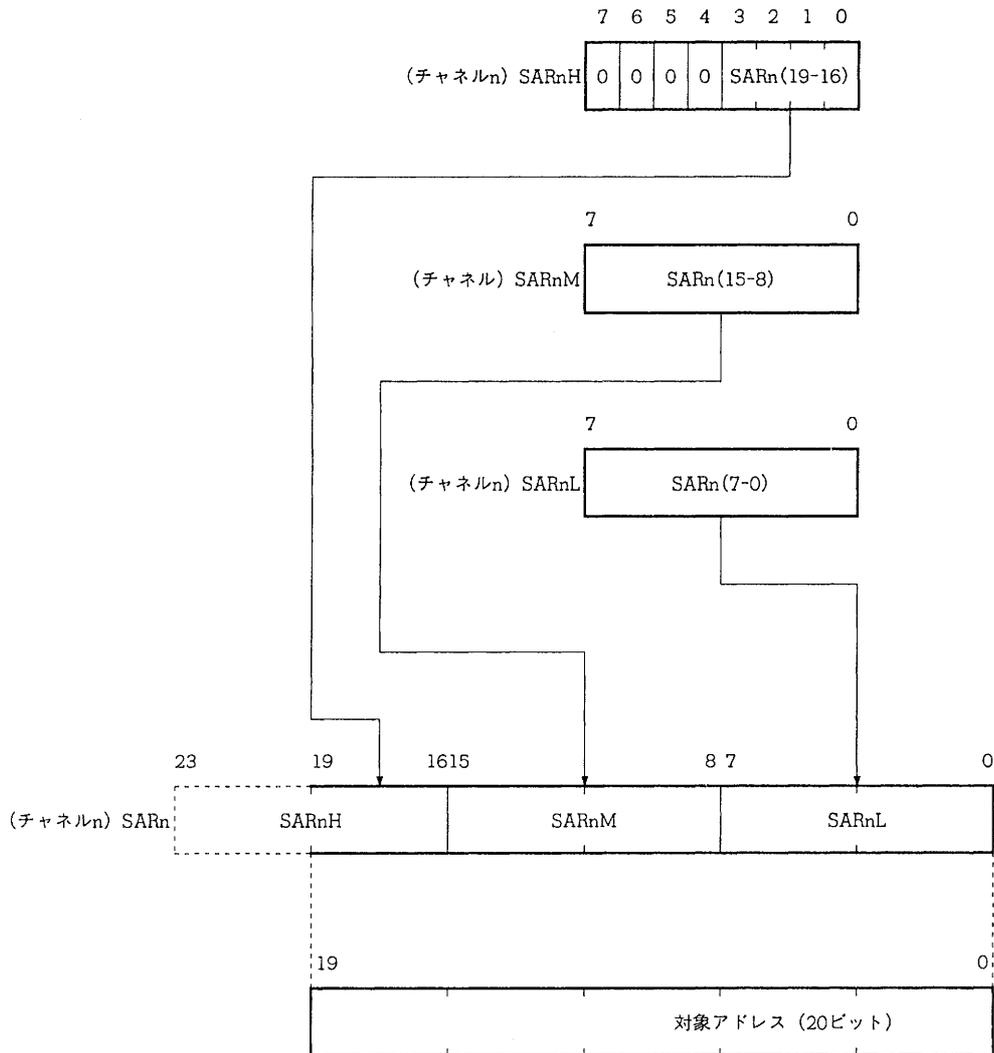
6.4.1 ソース・アドレス・ポインタ (SAR0, SAR1)

DMA転送における転送元アドレスを指定する20ビットのポインタです。20ビットのアドレスの下位8ビット、その上位8ビット、さらに最上位4ビットをそれぞれSARnLレジスタ, SARnMレジスタ, SARnHレジスタ (n=0, 1) により直接指定します。SARnHレジスタ (n=0, 1) の上位4ビットは0固定です。

SARnLレジスタ, SARnMレジスタ, SARnHレジスタ (n=0, 1) は16/8ビット操作 (SARnHレジスタは8ビット操作のみ) のメモリ・アクセスによって読み出し/書き込みを行います。

ポインタの値はDMA動作時にDMAコントロール・レジスタ (DMACn) によって指定されたモードに従って自動的に更新されます。バイト・データの場合は±1またはそのまま, ワード・データの場合は±2またはそのままとなります (μ PD70325の場合, バイト転送のみ可)。また, 更新20ビットに対してリニアに行われます。

図 6-4 SAR0 (SAR0H, SAR0M, SAR0L), SAR1 (SAR1H, SAR1M, SAR1L)



備考 n=0,1

6.4.2 デスティネーション・アドレス・ポインタ (DAR0, DAR1)

DMA転送における転送先アドレスを指定する20ビットのポインタです。20ビットのアドレスは、各レジスタにより次のように指定します。

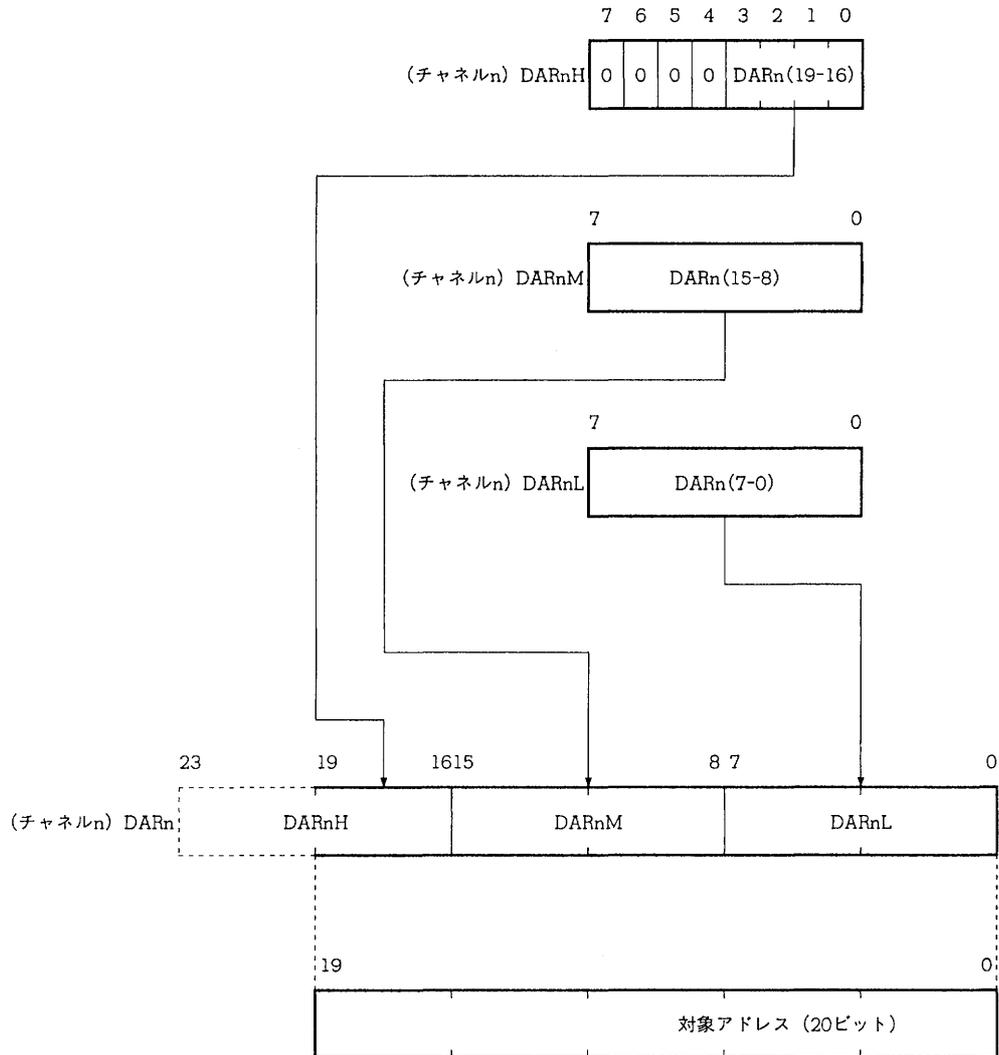
- DARnL：ビット7-ビット0
- DARnM：ビット15-ビット8
- DARnH：ビット19-ビット16

DARnHレジスタ (n=0, 1) の上位4ビットは0固定です。

DARnLレジスタ, DARnMレジスタ, DARnHレジスタ (n=0, 1) は16/8ビット操作 (DARnHレジスタは8ビット操作のみ) のメモリ・アクセスによって読み出し/書き込みを行います。

ポインタの値はDMA動作時にDMAコントロール・レジスタ (DMACn) によって指定されたモードに従って自動的に更新されます。バイト・データの場合は±1またはそのまま, ワード・データの場合は±2またはそのままとなります (μ PD70325の場合, バイト転送のみ可)。また, 更新は20ビットに対してリニアに行われます。

図6-5 DARO (DAROH, DAROM, DAROL), DARI (DARIH, DARIM, DARI L)



備考 n=0, 1

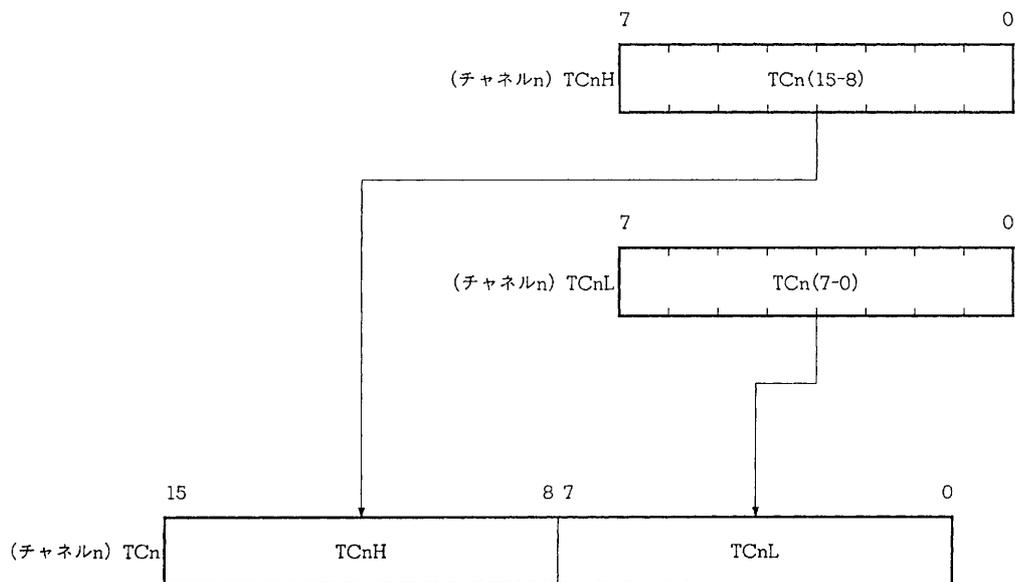
6.4.3 ターミナル・カウンタ (TC0, TC1)

DMA転送における転送回数を指定する16ビットのカウンタです。下位8ビット、上位8ビットをそれぞれTCnLレジスタ、TCnHレジスタ ($n=0, 1$) により指定します。転送回数-1の値を設定してください。

TCnLレジスタ、TCnHレジスタ ($n=0, 1$) は16/8ビット操作のメモリ・アクセスによって読み出し/書き込みが可能です。

カウンタの値はDMA動作により自動的に更新され、1回のDMA転送ごとに(-1)されます。

図 6-6 TCO (TCOH, TCOL), TC1 (TC1H, TC1L)



備考 $n=0, 1$

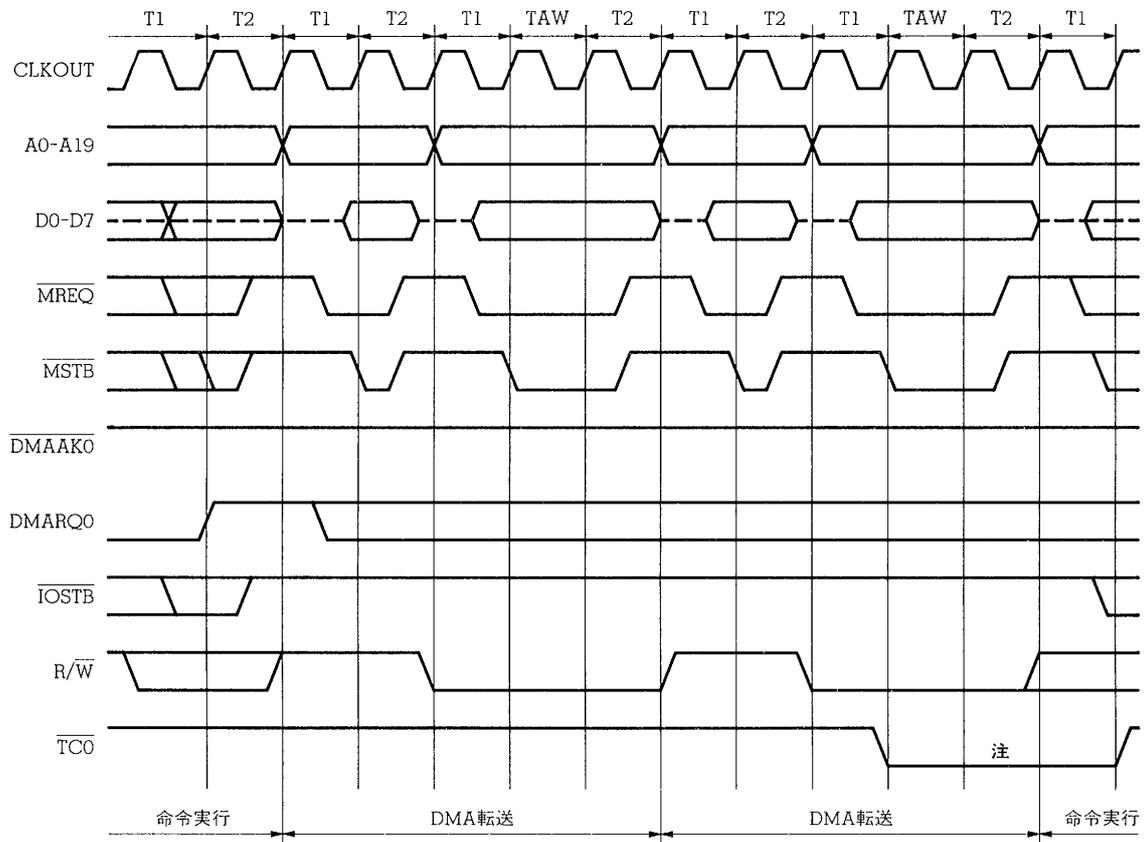
6.5 DMA転送タイミング

図6-7～図6-15におもなDMA転送のタイミングについて示します。

6.5.1 μ PD70325

TC=1のときにDMARQ信号によってDMAを起動した場合で転送元のメモリ・バンクはノー・ウェイ
ト、転送先は1ウェイト・ステートの挿入が行われているときのバースト・モードのタイミングを示し
ます。

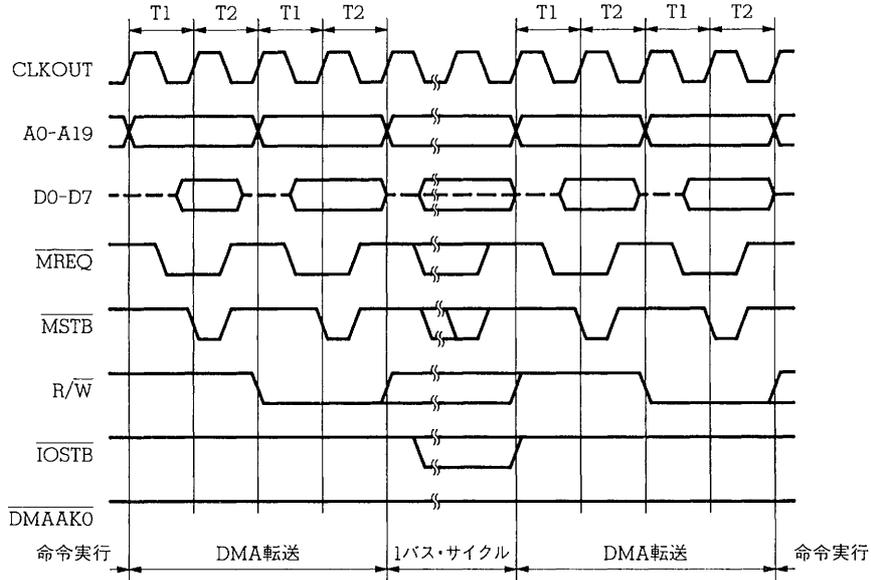
図6-7 バースト・モード時



注 TCOのアクティブ期間は2+Wクロック (W:ウェイト数)

備考 破線はハイ・インピーダンスを示します。

図 6-8 シングルステップ・モード時 (μ PD70325の場合)



備考 破線はハイ・インピーダンスを示します。

図 6-9 μ PD70325での1転送モード(メモリ→I/O, ノー・ウエイト時)

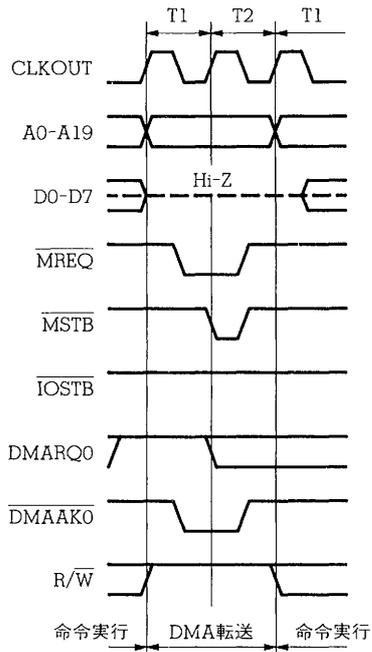
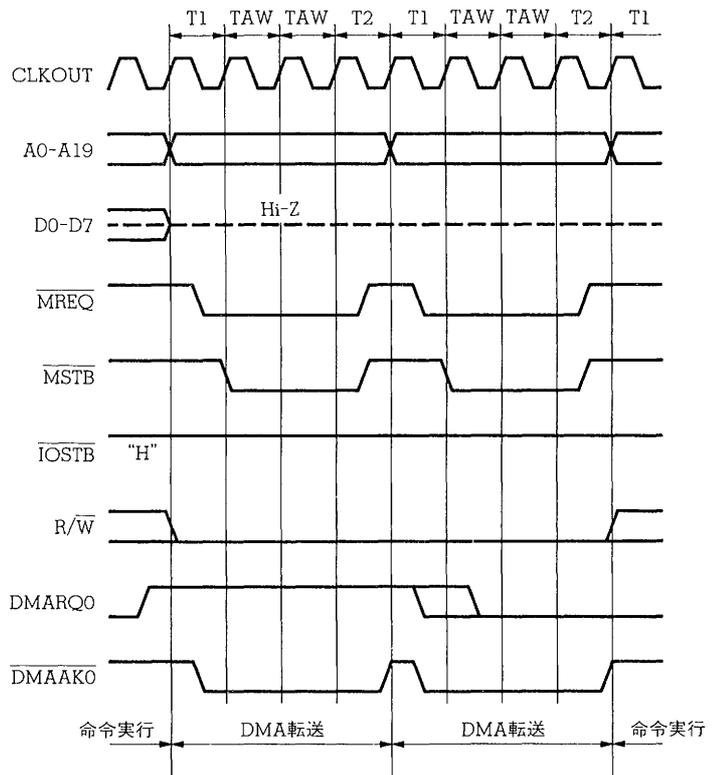


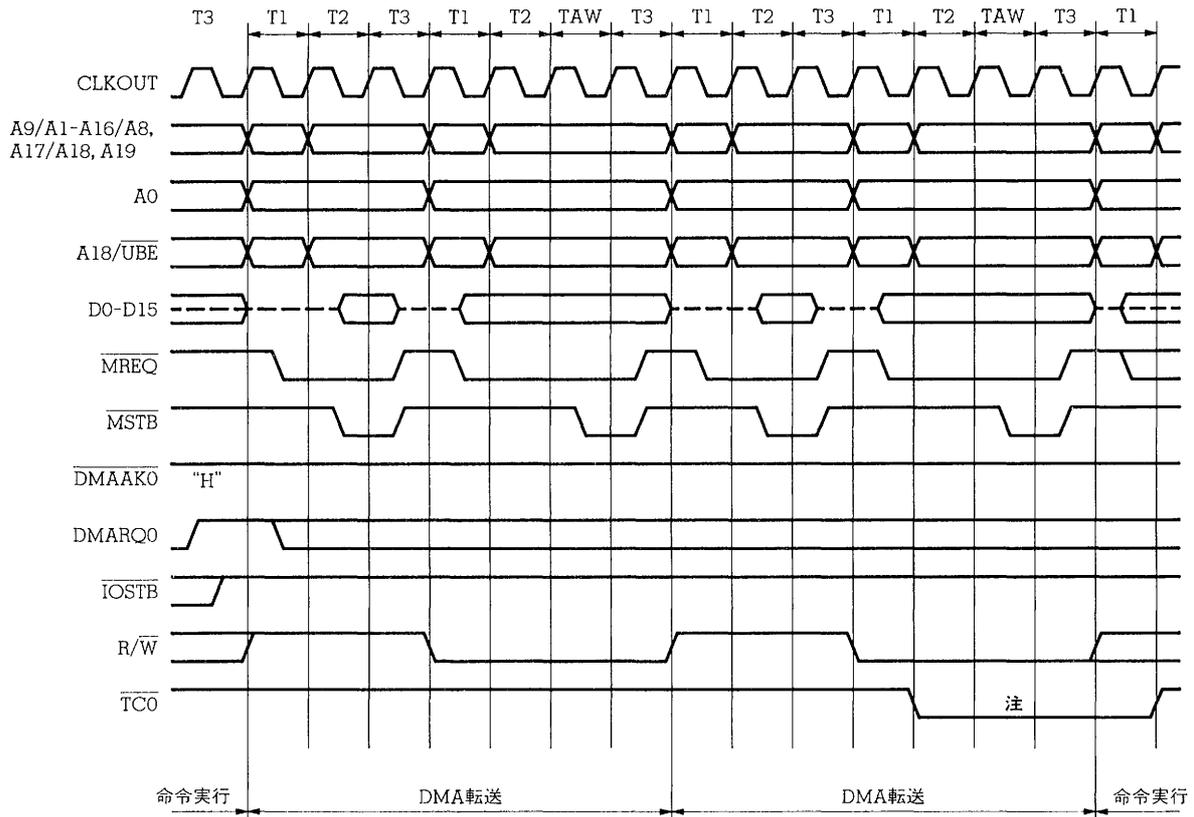
図 6-10 デイマンド・リリース・モード (I/O→メモリ, I/O: 2ウエイト, メモリ: ノー・ウエイト時)



6.5.2 μ PD70335

TC=1のときにDMARQ信号によってDMAを起動した場合で、転送元のメモリ・バンクはノー・ウエイト、転送先は1ウエイト・ステートの挿入が行われているときのバースト・モード（バイト転送）のタイミングを示します。

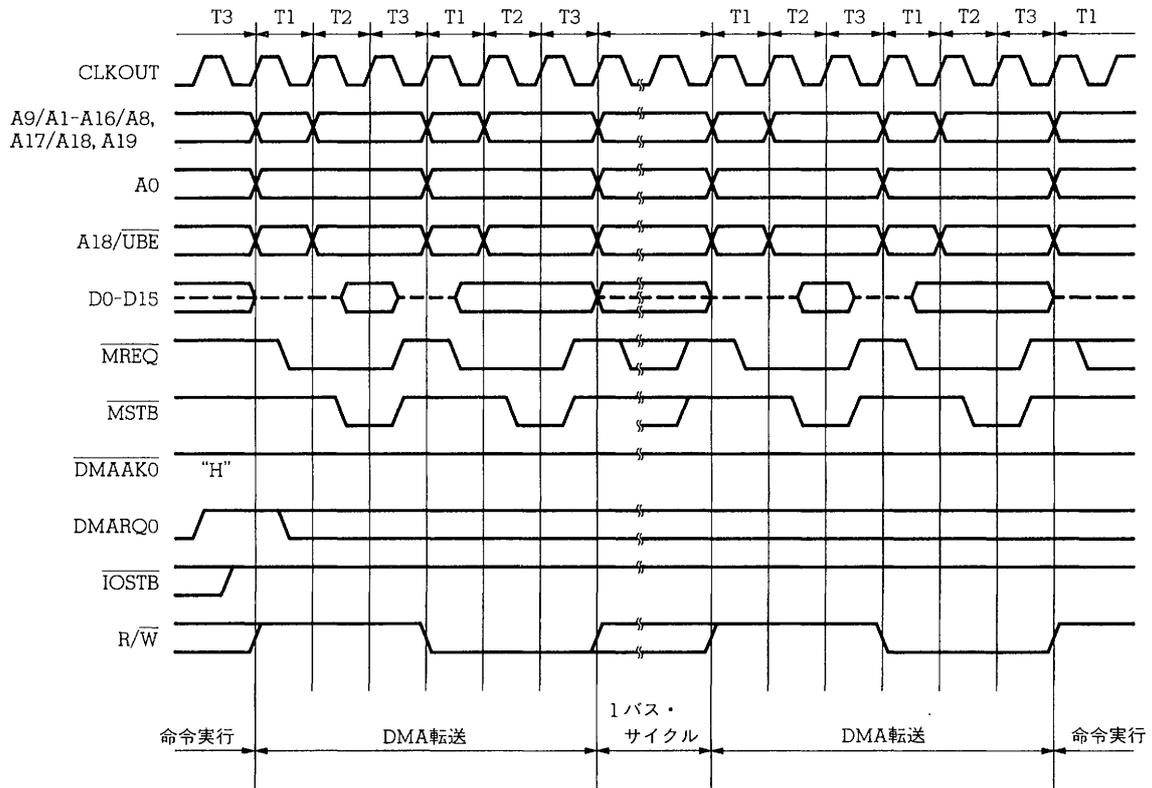
図6-11 バースト・モード時（バイト転送）



注 \overline{TCO} のアクティブ期間は3+Wクロック

備考 破線はハイ・インピーダンスを示します。

図 6-12 シングルステップ・モード時 (μ PD70335の場合)



備考 破線はハイ・インピーダンスを示します。

図 6-13 μ PD70335での1転送モード (メモリ→I/O, ノー・ウエイト時)

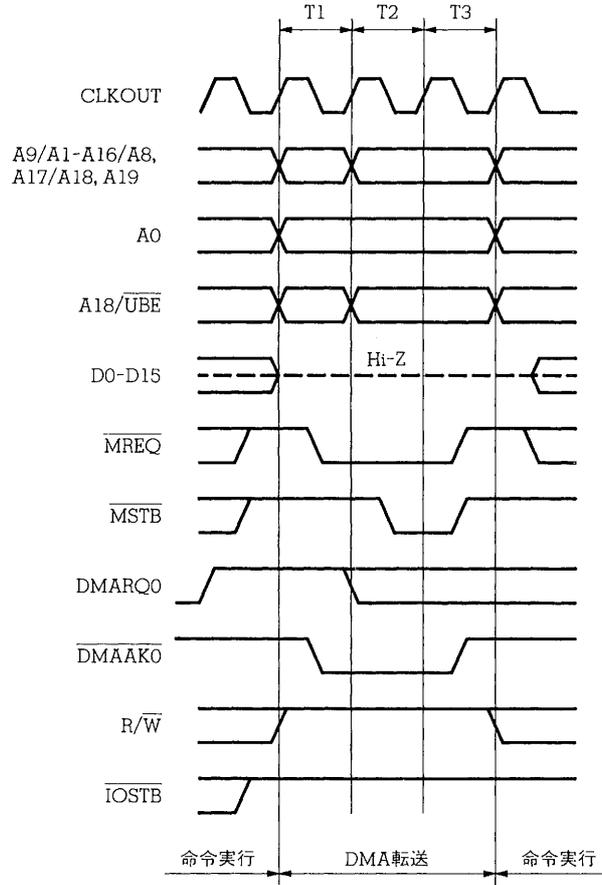
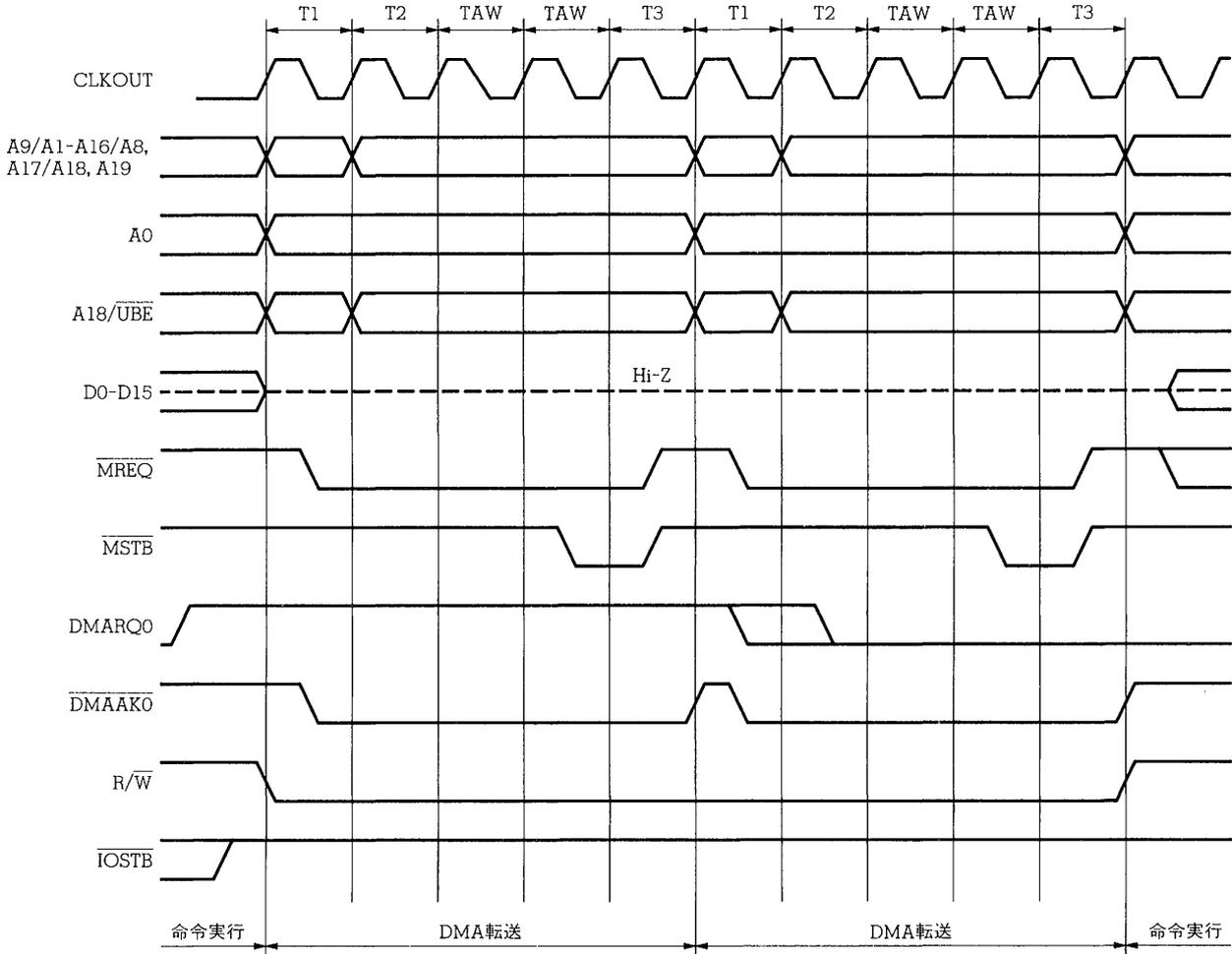


図6-14 ディマンド・リリース・モード (I/O→メモリ, I/O: 2ウエイト, メモリ: ノー・ウエイト)



I/Oとメモリの間でのDMA転送では、 \overline{IOSTB} 信号は出力されません (“H” レベル)。 \overline{DMAAK} 信号が出力されます (“L” レベル)。

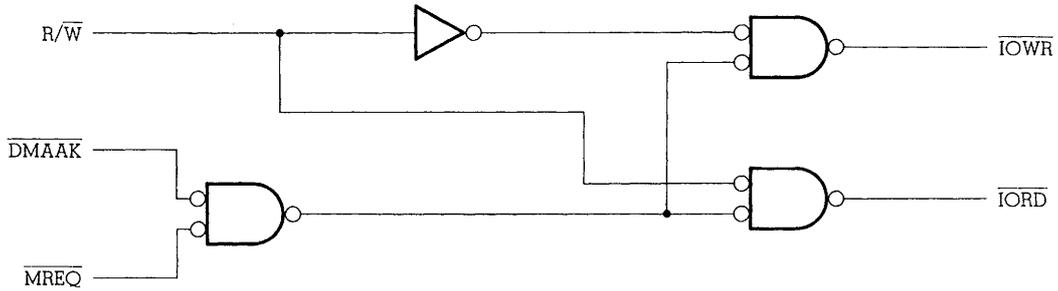
したがって、I/Oに対するアクセスの判定には \overline{DMAAK} 信号を使用します。メモリとメモリの間でのDMA転送で、 \overline{DMAAK} 信号は出力されません。

メモリ→I/O時の \overline{MREQ} ↑と \overline{DMAAK} ↑の遅延時間は規定できません。 \overline{MREQ} ↑に対して \overline{DMAAK} ↑が遅れた場合も考慮して回路を設計してください。

R/W信号、 \overline{IOSTB} 信号、 \overline{DMAAK} 信号、 \overline{MREQ} 信号から \overline{IORD} 信号、 \overline{IOWR} 信号を作成する回路例を図6-15に示します。

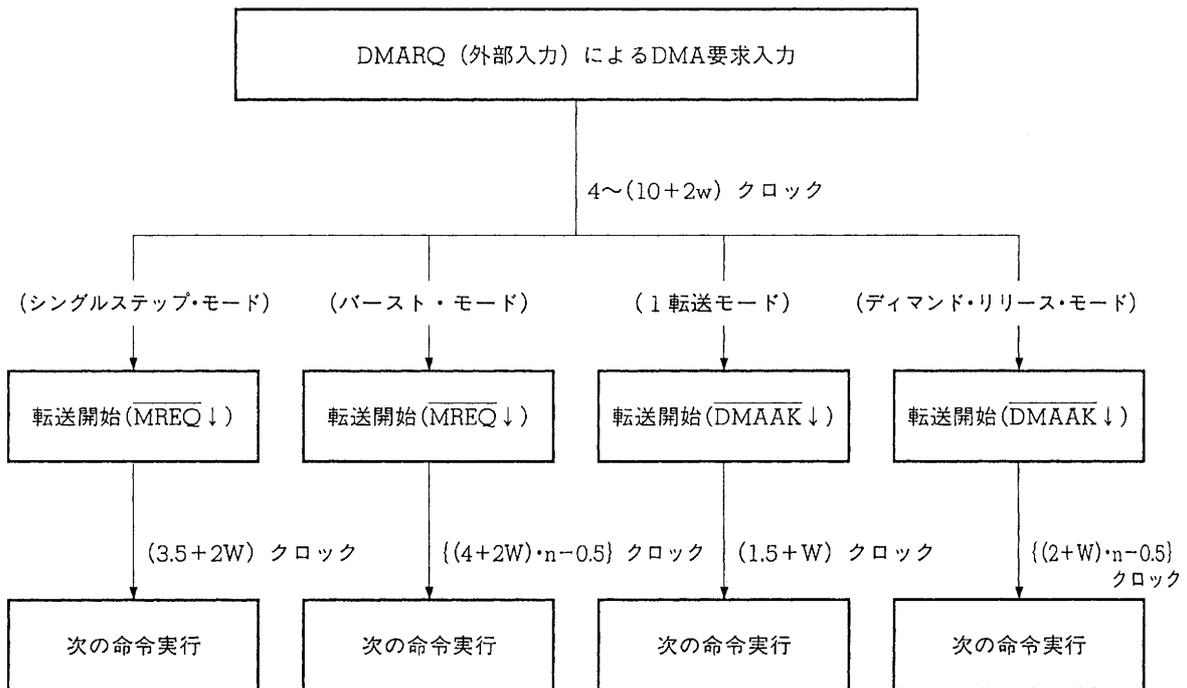
★

図 6-15 IORD信号, IOWR信号作成回路例



6.6 DMA実行時間

6.6.1 V25+のDMA実行時間 (システム・クロック数)



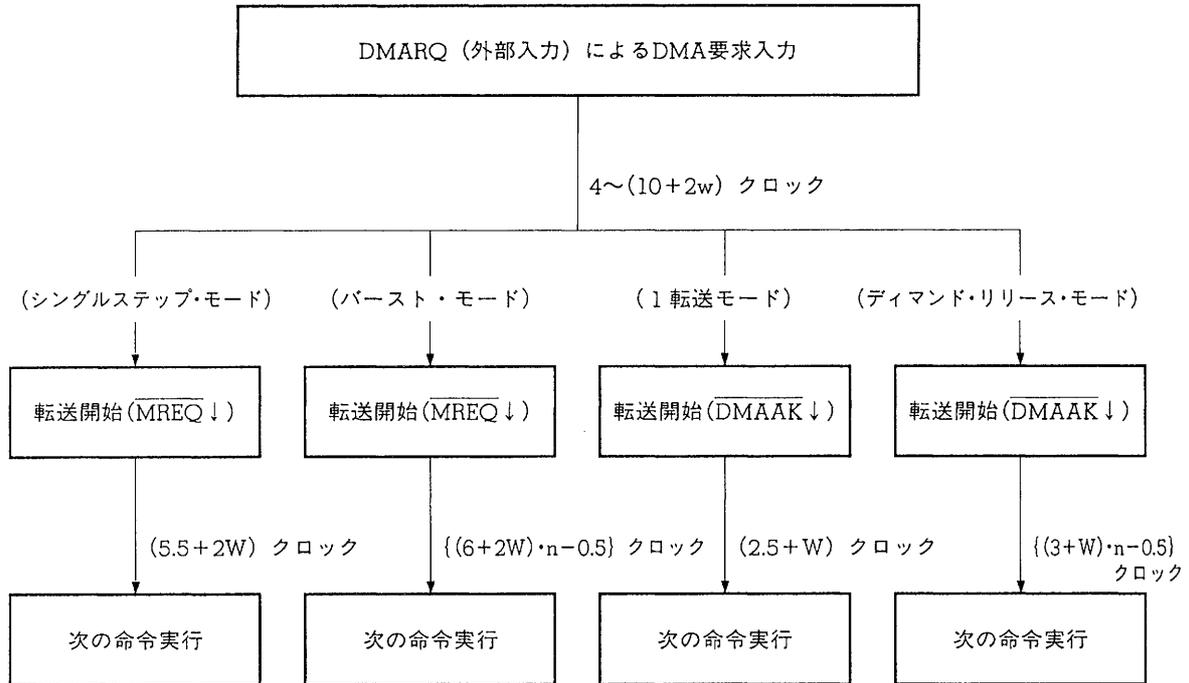
w : その時点のバス・サイクルのウェイト数

W : DMA転送のバス・サイクル当たりのウェイト数

n : DMA転送回数

注意 リフレッシュ・サイクル, ホールド要求, 割り込み要求, 他のDMA要求などは考慮しておりません。

6.6.2 V35+のDMA実行時間 (システム・クロック数)



w : その時点のバス・サイクルのウェイト数

W : DMA転送のバス・サイクル当たりのウェイト数

n : DMA転送回数

注意 リフレッシュ・サイクル, ホールド要求, 割り込み要求, 他のDMA要求などは考慮しておりません。

(メ モ)

第 7 章 ポート機能

7.1 ポート 0-2

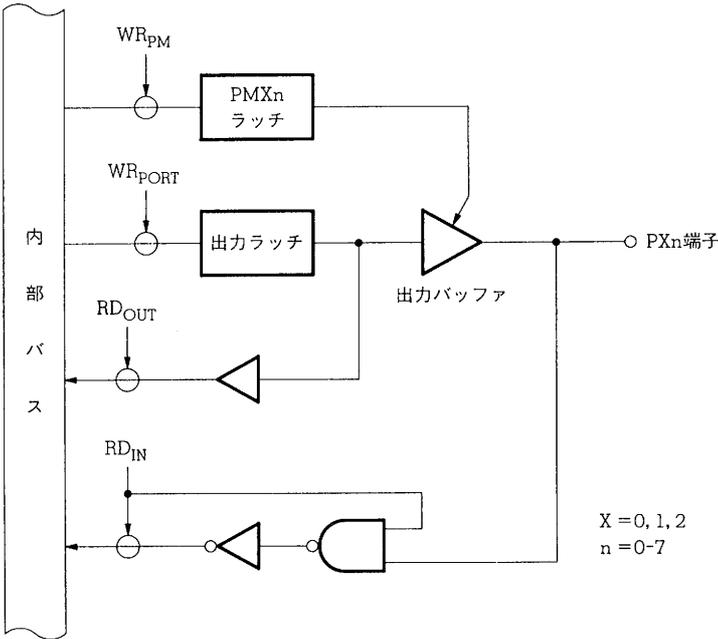
7.1.1 ハードウェア構成

μ PD70325, 70335のポート 0-2 は基本的に図 7-1 に示す 3 ステート双方向ポートで構成されています。

$\overline{\text{RESET}}$ 入力によりポート・モード・レジスタの各ビットはセット(1)され、入力ポートに指定されます。全ポート端子はハイ・インピーダンス状態となります。また、出力ラッチの内容は $\overline{\text{RESET}}$ 入力の影響を受けません。

7

図 7-1 ポート 0-2 構成



備考 PMXnラッチ：ポート・モード・レジスタ
PMXのビットn

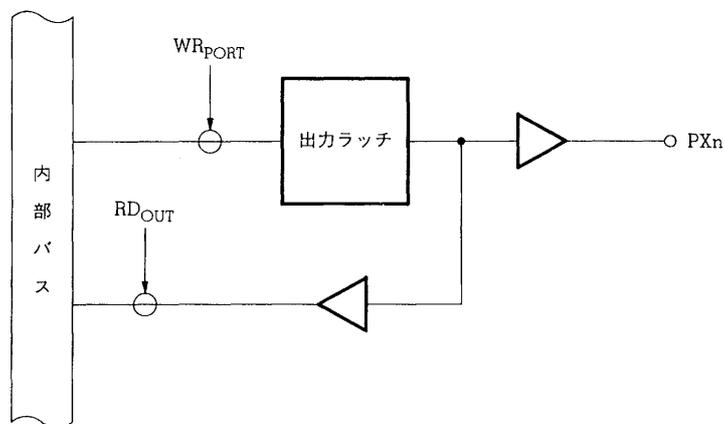
(1) 出力ポートに指定された場合 ($PMX_n=0$)

出力ラッチが有効となり、転送命令により出力ラッチと汎用レジスタとの間でデータのやりとりが行えます。

また出力ラッチの内容は論理演算命令により自由にセットできます。一度出力ラッチに書き込まれたデータは、次にポートを操作する命令が実行されるまで保持されます。

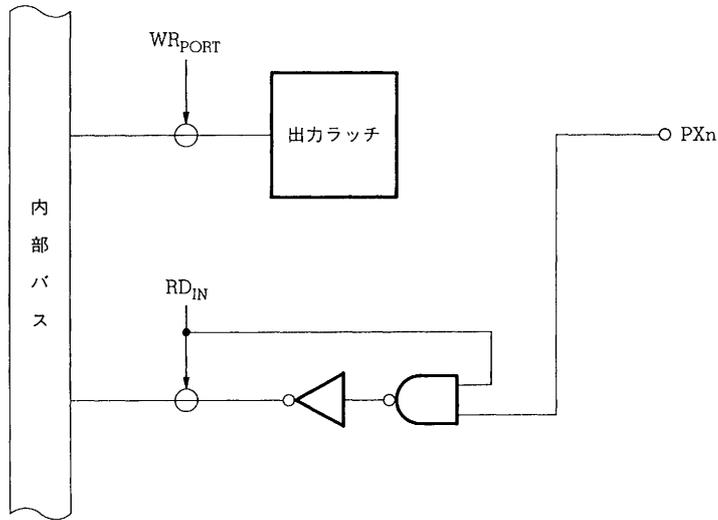
ポートのデータを読み込む操作を行った場合、出力端子の状態ではなく、出力ラッチの値が読み込まれます。

図 7-2 出力ポート指定のポート

(2) 入力ポートに指定された場合 ($PMX_n=1$)

ポート端子のレベルを転送命令で汎用レジスタにロードすることができます。この場合でも出力ラッチへの書き込みは可能で、転送命令により汎用レジスタから転送されたデータは、ポートの入出力指定に関係なく、全出力ラッチにストアされます。ただし、入力ポートに指定されているビットの出力バッファはハイ・インピーダンス状態になっているため、ポート端子には出力されません（入力指定のビットが出力ポートに切り替えられたとき、出力ラッチの内容がポート端子に出力されます）。また、入力ポートに指定されているビットの出力ラッチの内容を汎用レジスタにロードすることはできません。

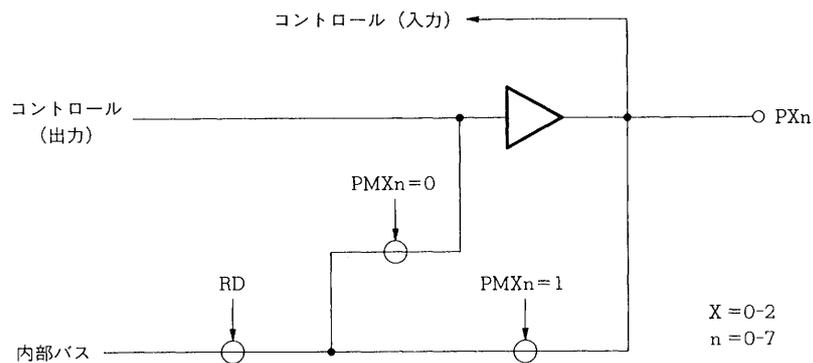
図 7-3 入力ポート指定のポート



(3) コントロール指定の場合 (PMCX_n=1)

ポート0-2はポート・モード・コントロール・レジスタ (PMCX) のビットをセット(1)することにより、ポート・モード・レジスタ (PMX) の設定にかかわらず、ビット単位にコントロール信号の入力あるいは出力として使用することができます。各端子をコントロール信号として用いる場合、ポートのアクセス命令を実行することにより、コントロール信号の状態をみることができます。

図 7-4 コントロール指定のポート



(a) ポートがコントロール信号出力の場合

ポート・モード・レジスタ (PMX_n) がセット(1)されているときに、ポートの読み込み命令を実行すると、コントロール信号の端子状態を読み込むことができます。

ポート・モード・レジスタがリセット(0)されているときにポートの読み込み命令を実行すると内部のコントロール信号の状態を読み込むことができます。

(b) ポートがコントロール信号入力の場合

ポート・モード・レジスタがセット(1)されている場合のみ、ポートの読み込み命令を実行するとコントロール信号の端子状態を読み込むことができます。

7.1.2 各ポートの機能

(1) P00-07 (ポート 0) …… 3 ステート入出力

8ビットの特殊入出力ポートです。ビット単位に入出力の指定ができる汎用入出力ポートとして機能するほか、システム・クロック出力端子 (P07と兼用) としても機能します。それらの切り替えは、ポート 0 モード・レジスタ (PM0) およびポート 0 モード・レジスタ (PMCO) の指定により、ビット単位に行うことができます。

表 7-1 ポート 0 の動作 (n=0-7)

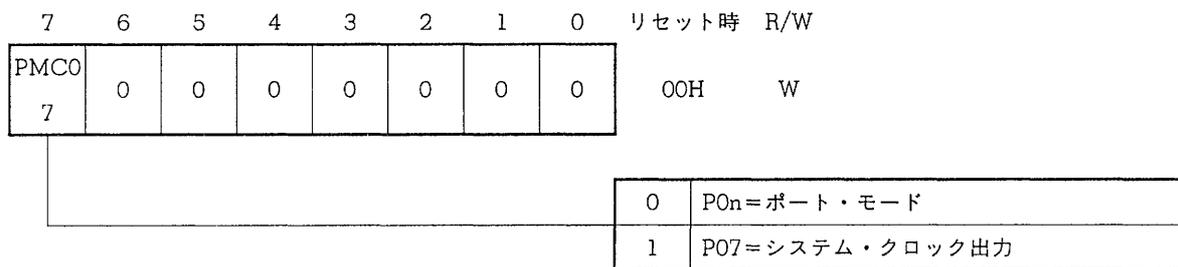
	PMCO _n =1	PMCO _n =0	
		PM0 _n =1	PM0 _n =0
P00	X	入力ポート	出力ポート
P01		入力ポート	出力ポート
P02		入力ポート	出力ポート
P03		入力ポート	出力ポート
P04		入力ポート	出力ポート
P05		入力ポート	出力ポート
P06		入力ポート	出力ポート
P07	CLKOUT出力	入力ポート	出力ポート

(a) ポート 0 モード・コントロール・レジスタ (PMCO)

ポート 0 のポート / システム・クロック出力の指定をビット単位で行う 8ビットのレジスタです。8ビット操作のメモリ・アクセスによって、このレジスタの書き込みのみができます。PMCOレジスタの対応ビットがセット(1)されていればシステム・クロック出力モード(P07), リセット(0)されていればポート・モードになります。

RESET入力時PMCOレジスタは全ビットがリセット(0)され、ポート・モードになります。

図 7-5 PMCO



(b) ポート 0 モード・レジスタ (PMO)

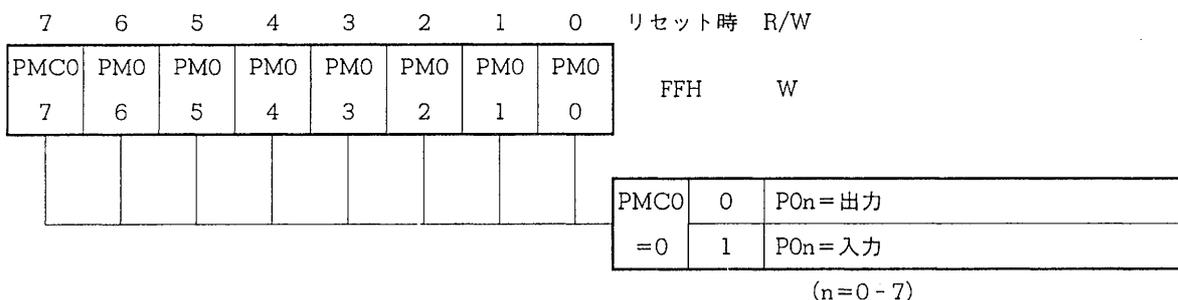
PMOは、ポート 0 の入力/出力をビット単位で指定する 8 ビットのレジスタです。

8 ビット操作のメモリ・アクセスによって、このレジスタの書き込みのみができます。

PMCOの対応するビットが“0” のときPMOは有効になります。

$\overline{\text{RESET}}$ 入力により全ビットがセット(1)され、入力ポートとなります。

図 7-6 PMO



(2) P10-17 (ポート 1) …… 3 ステート入出力

8 ビットの特異入出力ポートです。ポート 0 と同様、ビット単位に入出力の指定ができる汎用入出力ポートとして機能する他、各種コントロール端子としても機能します。それらの切り替えは、ポート 1 モード・コントロール・レジスタ (PM1) およびポート 1 モード・コントロール・レジスタ (PMC1) の指定により、ビット単位に行うことができます。

P10-P13端子はポート 1 (P1) を直接リード・アクセスすることによって、端子レベルを読み込むことができます。

P10端子は汎用の入力ポートとしては使用できません。



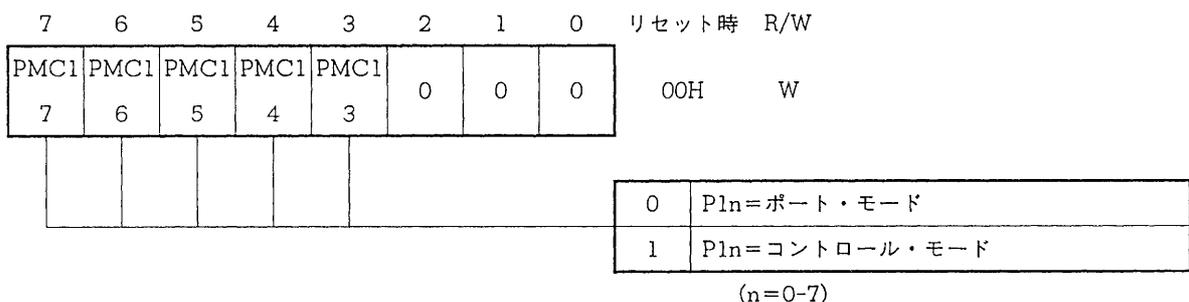
表 7-2 ポート 1 の動作 (n=0-7)

	PMCl _n =1	PMCl _n =0	
		PMl _n =1	PMl _n =0
P10	X	NMI入力	X
P11		INTP0入力	
P12		INTP1入力	
P13		INTAK出力 INTP2入力	
P14	INT入力	入力ポート (POLL入力)	出力ポート
P15	TOUT出力	入力ポート	出力ポート
P16	SCKO出力	入力ポート	出力ポート
P17	READY入力	入力ポート	出力ポート

(a) ポート 1 モード・コントロール・レジスタ (PMCl)

ポート 1 のポート/コントロール信号入出力の指定をビット単位で行う 8 ビットのレジスタです。8 ビット操作のメモリ・アクセスによって、このレジスタの書き込みのみができます。PMClレジスタの対応ビットがセット(1)されていればコントロール信号入出力モード、リセット(0)されていればポート・モードになります。 $\overline{\text{RESET}}$ 入力時PMClレジスタは全ビットがリセット(0)され、ポート・モードになります。ただし、P10-P12はポート・モードに固定です。

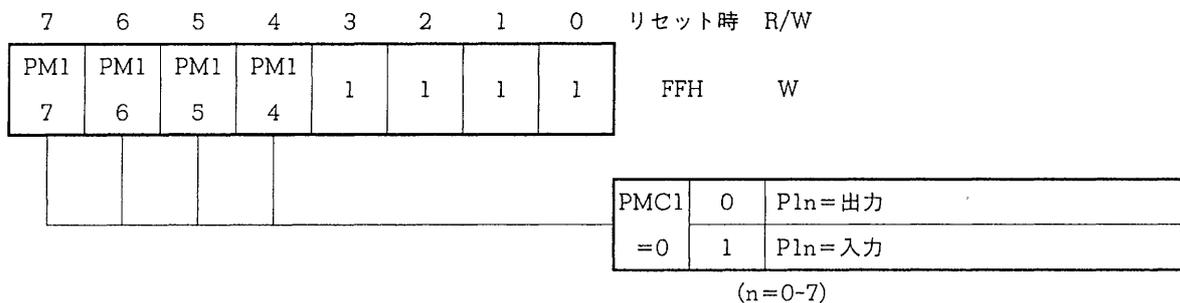
図 7-7 PMCl



(b) ポート 1 モード・レジスタ (PMl)

PMlは、ポート 1 の入力/出力をビット単位で指定する 8 ビットのレジスタです。8 ビット操作のメモリ・アクセスによって、このレジスタの書き込みのみができます。PMClの対応するビットが“0”のときPMlは有効になります。 $\overline{\text{RESET}}$ 入力により全ビットがセット(1)され、入力ポートとなります。

図 7-8 PM1



(3) P20-27 (ポート 2) …… 3 ステート入出力

8ビットの特殊入出力ポートです。ポート0と同様、ビット単位に入出力の指定ができる汎用入出力ポートとして機能する他、各種コントロール端子としても機能します。それらの切り替えは、ポート 2 モード・レジスタ (PM2) およびポート 2 モード・コントロール・レジスタ (PMC2) の指定により、ビット単位に行うことができます。

表 7-3 ポート 2 の動作 (n=0-7)

	PMC2n=1	PMC2n=0	
		PM2n=1	PM2n=0
P20	DMARQ0入力	入力ポート	出力ポート
P21	$\overline{\text{DMAAK0}}$ 入力	入力ポート	出力ポート
P22	$\overline{\text{TC0}}$ 出力	入力ポート	出力ポート
P23	DMARQ1入力	入力ポート	出力ポート
P24	$\overline{\text{DMAAK1}}$ 出力	入力ポート	出力ポート
P25	$\overline{\text{TC1}}$ 出力	入力ポート	出力ポート
P26	$\overline{\text{HLDK}}$ 出力	入力ポート	出力ポート
P27	HLDK入力	入力ポート	出力ポート

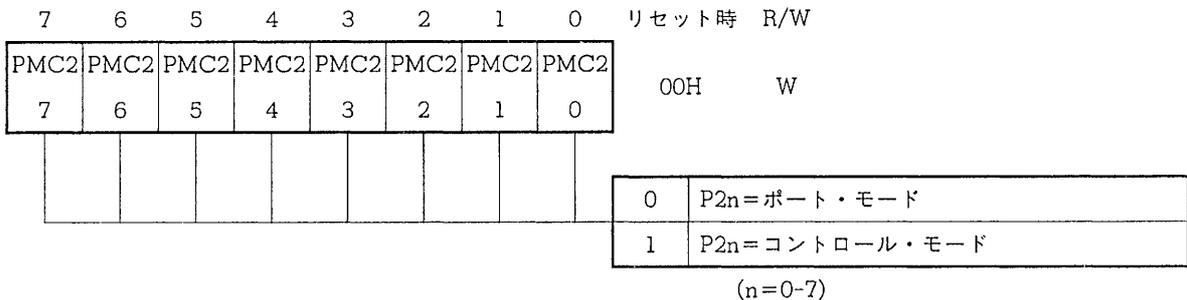
(a) ポート 2 モード・コントロール・レジスタ (PMC2)

ポート 2 のポート / コントロール信号入出力の指定をビット単位で行う 8 ビットのレジスタです。

8 ビット操作のメモリ・アクセスによって、このレジスタの書き込みのみができます。

PMC2レジスタの対応ビットがセット(1)されていればコントロール信号入出力モード、リセット(0)されていればポート・モードになります。 $\overline{\text{RESET}}$ 入力時PMC2レジスタは全ビットがリセット(0)され、ポート・モードになります。

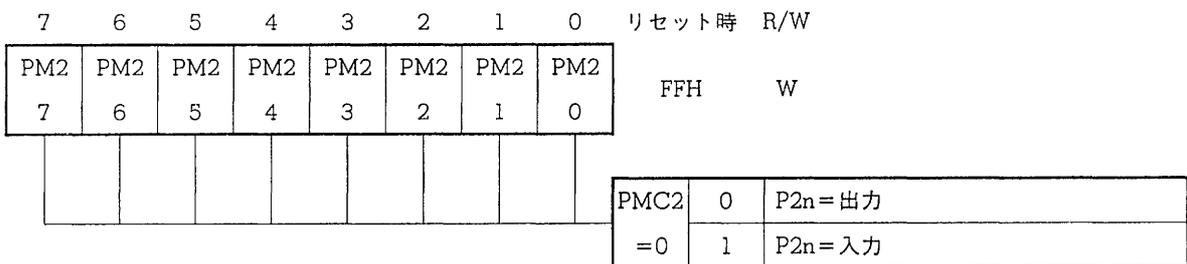
図 7-9 PMC2



(b) ポート 2 モード・レジスタ (PM2)

PM2は、ポート 2 の入力／出力をビット単位で指定する 8 ビットのレジスタです。
 8 ビット操作のメモリ・アクセスによって、このレジスタの書き込みのみができます。
 PMC2の対応するビットが0のとき、PM2の各ビットは有効になります。
 $\overline{\text{RESET}}$ 入力により全ビットがセット(1)され、入力ポートとなります。

図 7-10 PM2



★ (4) 注意事項

リセットしたあとに出力ポートとして使用するときは以下に示すことに注意してください。

・パワーオン・リセット時

出力ラッチの内容は不定になっているため、出力ポートとして使用する場合は、出力したい値をポートに書き込んでから、ポート・モード・レジスタを入力から出力に切り替えてください。

ポートを設定する前に切り替えると、ポート端子より不定値が出力されます。

・システム・リセット時

リセット前の出力ラッチの値が保持されているため、ポート・モード・レジスタを入力から出力に切り替えるとリセット前の出力ラッチの値がポート端子より出力されます。

7.2 ポートT (PT0-PT7)

ポートTは、スレッシュホールド電圧 (基準電圧) を16段階に替えられる8ビットの入力ポートです。このアナログ入力によってコンパレータ動作を行います。

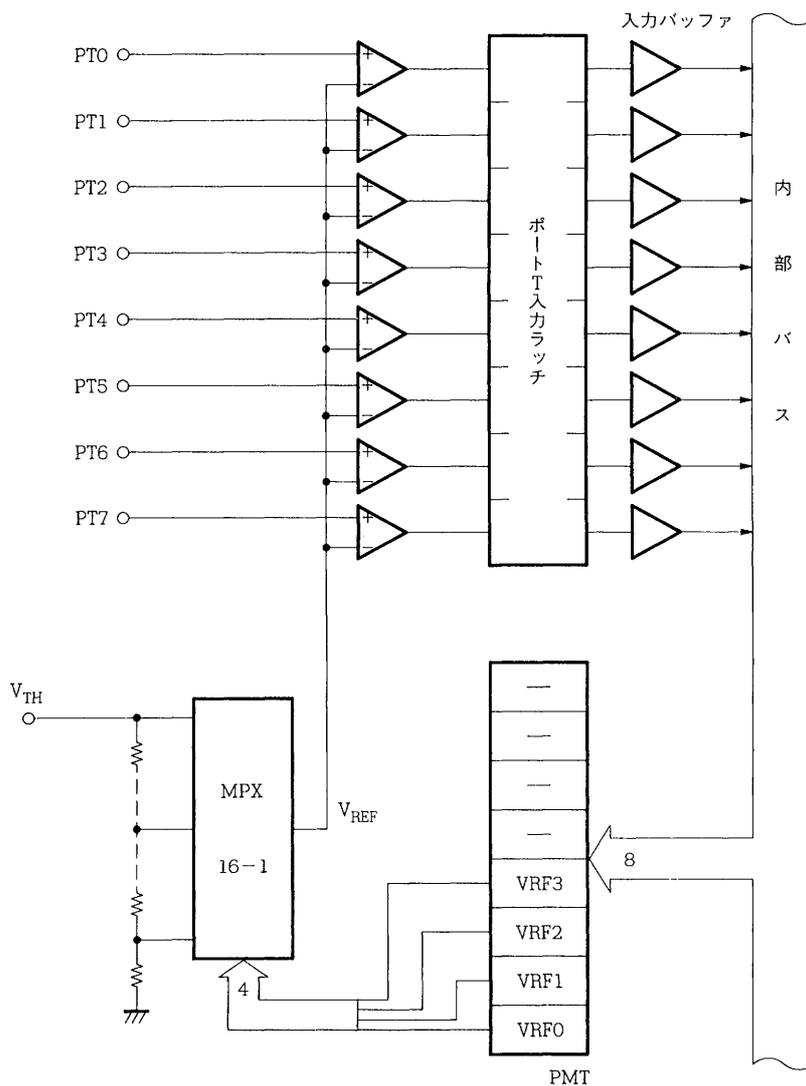
7.2.1 ハードウェア構成

ポートTは、図7-11のようにPT0-PT7のコンパレータ入力, 基準電源入力の V_{TH} 端子, 比較電圧(V_{REF})を $1/16 \times V_{TH} \sim 16/16 \times V_{TH}$ の16通りに選択するマルチプレクス回路 (MPX), MPXを制御するポート・モードTレジスタ (PMT), および8個のラッチで構成されています。

PMTの設定により選択された V_{REF} とPT0-PT7入力が, コンパレータにより比較され, その結果がポートT入力ラッチにラッチされます。

$$\begin{cases} V_{REF} > PTn \rightarrow 0 \\ V_{REF} < PTn \rightarrow 1 \end{cases}$$

図7-11 ポートTのブロック図





注意 V_{TH} 端子は、高抵抗でGND端子と接続されています。したがって、スタンバイ・モード時に V_{TH} 端子に電圧が印加されていると、消費電流が増加します。

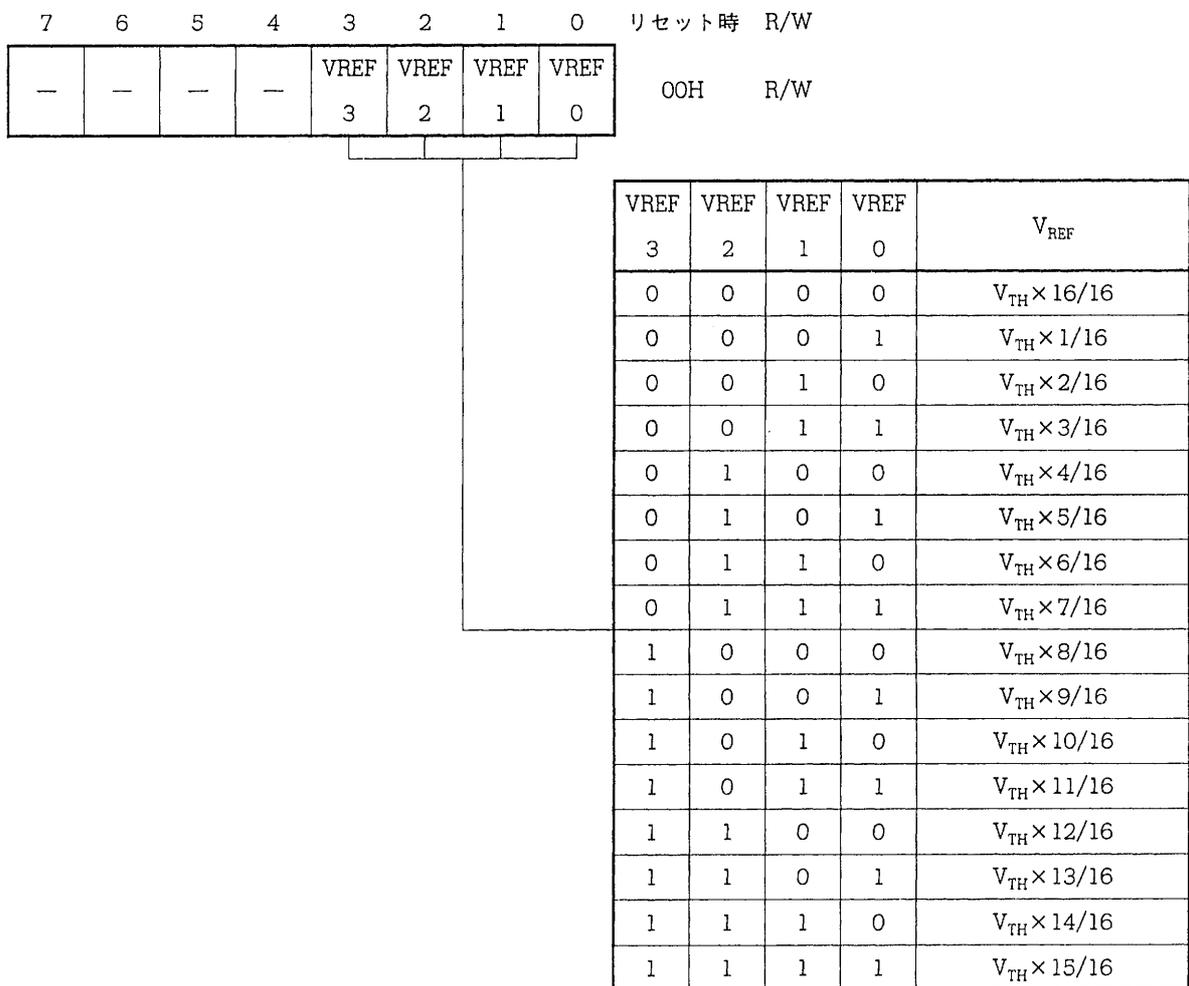
7.2.2 ポートTモード・レジスタ (PMT)

PMTは、コンパレータの比較電圧 (V_{REF}) を図7-12のように16通りのうちから設定します。

8ビット操作または1ビット操作のメモリ・アクセスによって、このレジスタの書き込み/読み出しができます。

\overline{RESET} 入力でPMTは全ビット、リセット(0)されます。

図7-12 PMT



7.2.3 ポートT受信

★

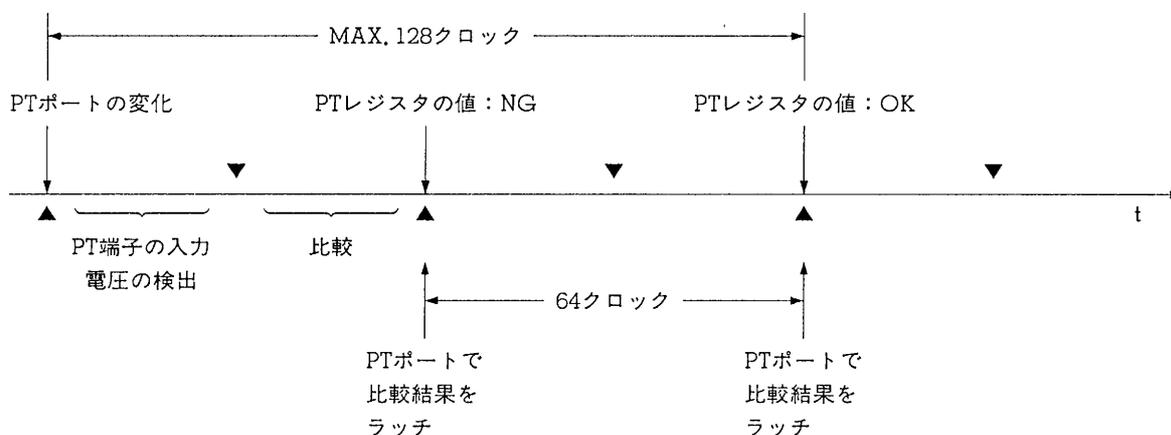
PT0-7端子の比較動作は次の2つのステージに分割して実行しています。

第1ステージ：PT端子の電圧を検出します。

第2ステージ：第1ステージで検出した電圧とPMTレジスタにより設定される電圧を比較します。

第1ステージと第2ステージの実行時間を合わせた比較時間は、64クロックになります。

第1ステージと第2ステージは、常に交互に実行されます。その開始タイミングをソフトウェアまたはハードウェアで制御することはできません。したがって、正しい比較結果を得るにはPT端子の入力電圧または比較電圧を変更したときは、128クロック以上経過したあとに、PTレジスタの値を読み出してください。ただし、PT端子の入力電圧はその間安定させてください。



保守 / 廃止

(× 毛)

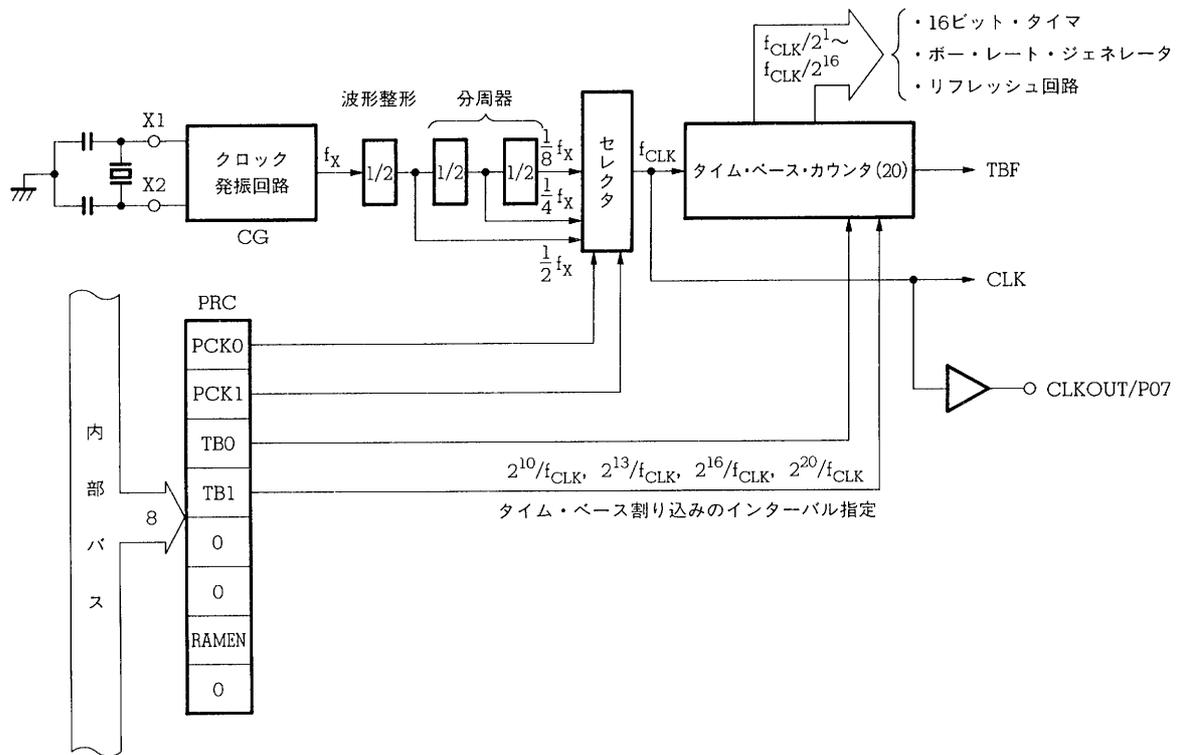
第 8 章 クロック発生回路

クロック発生回路は、CPUおよび周辺ハードウェアに各種クロックを供給し、CPUの動作モードを制御する回路です。

8.1 クロック発生回路の構成

クロック発生回路は図 8-1 のように構成されています。

図 8-1 クロック発生回路のブロック図



f_x : 発振周波数

f_{CLK} : システム・クロック周波数

PRC : プロセッサ・コントロール・レジスタ

TBF : タイム・ベース割り込み要求フラグ

CLKOUT : システム・クロック出力端子

クロック発生回路のクロックは、X1, X2端子に接続されたクリスタル発振またはセラミック発振子によって発振します。クロック発振回路の出力は波形整形（1/2分周）され、分周比を選択してシステム・クロック（CLK）として使用します。

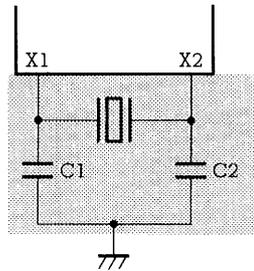
CLKの分周比は、プロセッサ・コントロール・レジスタ（PRC）のビット0, 1（PCK0, PCK1）の指定により、発振周波数（ f_x ）の1/2, 1/4または1/8に選択することができます。

CLKを低速にすることによってシステム・クロック周波数が低くなり、バッテリー駆動システムで電圧が低下しても長時間の動作が可能となります。

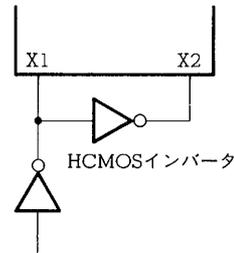
また、外部クロックを入力することもできます。

図 8-2 クロック発生回路の外付け回路

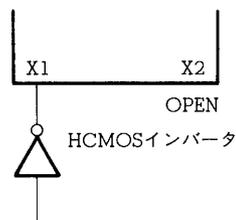
(a) クリスタル/セラミック発振



(b) 外部クロック入力



(c) 外部クロック入力 (X2: オープン)



- 注意1. 発振回路はX1, X2端子にできるかぎり近づけてください。
 2. 図8-2(a)の範囲に他の信号線を通さないでください。

表 8-1 推奨するセラミック発振子

メーカー	品 名	推奨定数	
		C1 [pF]	C2 [pF]
村田製作所	CSA16,00MX040	30	30
	CSA20,00MX040	10	10
TDK	FCR16,0M2G	30	30

表 8-2 推奨するクリスタル振動子

メーカー	品 名	推奨定数	
		C1 [pF]	C2 [pF]
キンセキ	HC-49/U (KR-100)	22	22
	HC-49/U (KR-160)	22	22
	HC-49/U (KR-200)	22	22

備考 各発振子の特性等につきましては、各発振子メーカーへご確認ください。

8.2 プロセッサ・コントロール・レジスタ (PRC)

PRCレジスタは、CPUの動作クロック、タイム・ベース割り込みの周期、内部RAMのメモリ参照許可など、CPUや内部システム制御に関連する項目を集中的に制御する8ビット・レジスタです。

8ビット操作または1ビット操作のメモリ・アクセスによって、このレジスタの書き込み／読み出しができます。

$\overline{\text{RESET}}$ 入力によってPRCの内容は4EHに初期化されます。

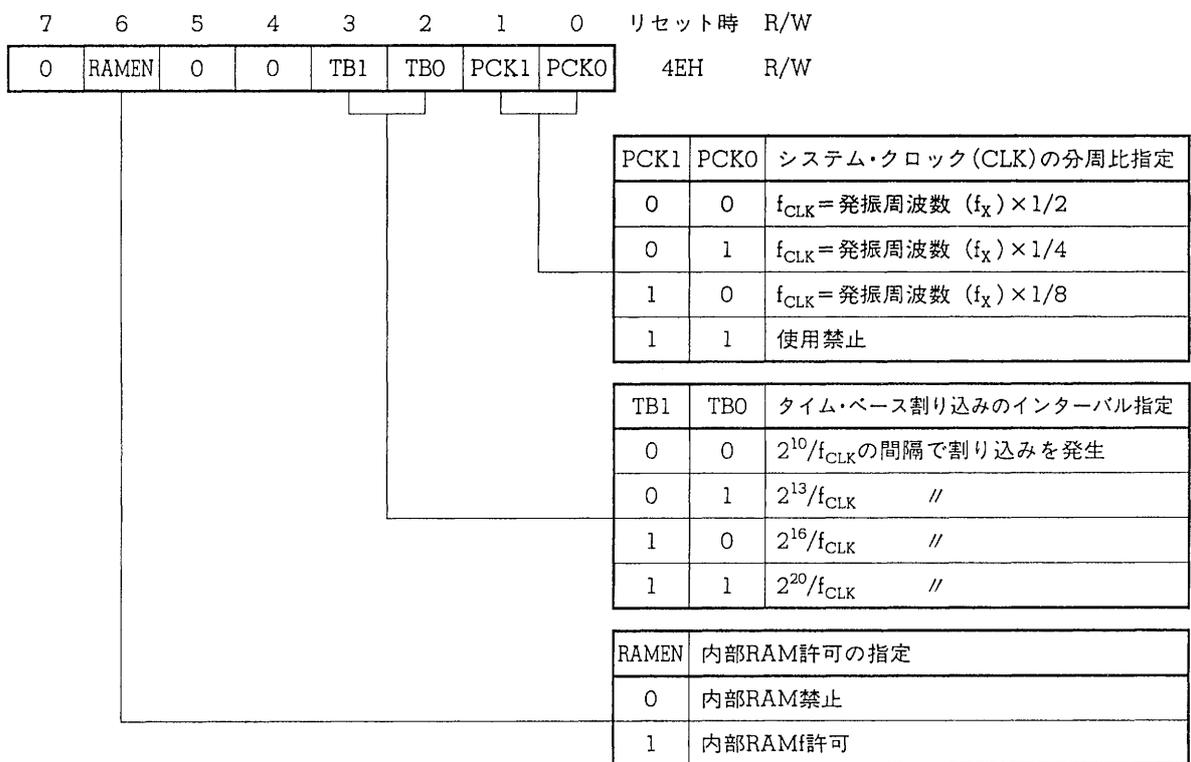
PCK0, 1ビットは、システム・クロックの分周比を指定します。発振器の周波数をPCK0, 1で指定した値で分周後、システム・クロック (CLK) として使用します。

TBO, 1ビットは、タイム・ベース割り込みのインターバルを指定します。TBO, 1ビットによって、4種類のロング・インターバル時間を選択できます。

RAMENビットは、内部RAMのメモリ参照の許可を制御します。禁止状態 (RAMENビット "0") では、内部RAMのアドレス判別を行わず、外部メモリが常にアクセスの対象になります。

なおレジスタとしてRAMを参照するときには、内部RAMが常にアクセスの対象になります。

図 8-3 PRC



第9章 タイマ・ユニット

μ PD70325, 70335のタイマ・ユニットは、インターバル・タイマ、ワンショット・タイマおよび矩形波出力として用いることができます。

9.1 タイマ・ユニットの構成と動作

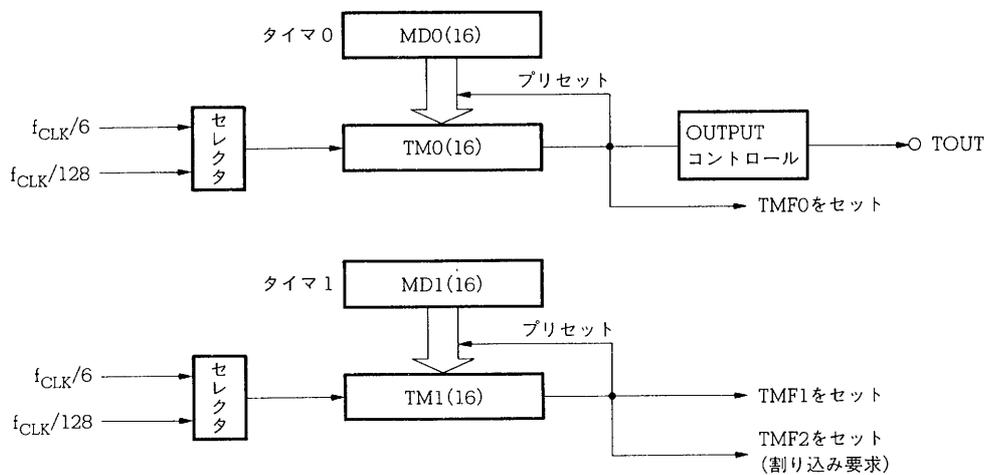
タイマ・ユニットは、2つの16ビット・タイマ・レジスタ、2つの16ビット・モジュロ／タイマ・レジスタおよび1つの8ビット・タイマ・コントロール・レジスタで構成されます。以下、各動作モードについて、構成と動作を説明します。

(1) インターバル・タイマ・モード

タイマ・ユニットをインターバル・タイマ・モードに設定した場合、図9-1のようにタイマ0, 1の2つを用いることができます。

9

図9-1 インターバル・タイマ・モード時のタイマ・ユニットの構成



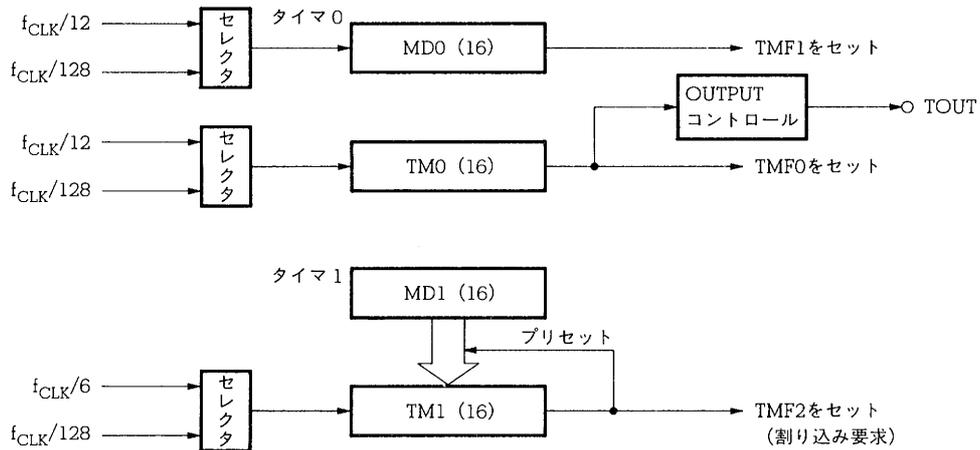
タイマ・コントロール・レジスタ (TMC0) でインターバル・タイマ・モードを指定し、TS0ビットをセット(1)すると、TM0レジスタにMD0レジスタ値をセットし、TCLK0ビットで指定されたクロックをダウン・カウントします。ダウン・カウント中、タイマの値が“0”になると次のクロックで再びTM0レジスタにMD0レジスタ値をセットし、再びダウン・カウントを繰り返します(カウント値=設定値+1)。

タイマ1のレジスタについても同様のカウント動作をします。

(2) ワンショット・タイマ・モード

タイマ・ユニットをワンショット・タイマ・モードに設定した場合、図9-2のようにタイマ0を用います。ただし、タイマ1をインターバル・タイマとして同時に動作させることもできます。

図9-2 ワンショット・タイマ・モード時のタイマ・ユニットの構成



タイマ・コントロール・レジスタ (TMC0) でワンショット・タイマ・モードを指定し、TS0/MS0ビットをセット(1)すると、TM0/MD0レジスタはTCLK0/MCLK0ビットで指定されたクロックをダウン・カウントします。カウント中、タイマの値が“0”になるとカウント動作が停止します。このとき、TM0/MD0レジスタは0000Hを保持します。



9.2 タイマ・コントロール・レジスタ (TMC0, TMC1)

TMC0レジスタはTM0, MD0レジスタの動作を制御し、TMC1レジスタはTM1, MD1レジスタの動作を制御する8ビット・レジスタです。

8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み／読み出しができます。この場合、最大6ウェイトが挿入されます。

$\overline{\text{RESET}}$ 入力により、各レジスタの内容は00Hに初期化されます。TMC0とTMC1では下図のようにフォーマットが異なります。

(インターバル・タイマ・モード, ワンショット・タイマ・モード)

	7	6	5	4	3	2	1	0
TMC0	TS0	TCLK0	MS0	MCLK0	ENTO	ALV	MOD1	MOD0

(インターバル・タイマ・モード)

	7	6	5	4	3	2	1	0
TMC1	TS1	TCLK1	0	0	0	0	0	0

TMC0, TMC1レジスタのビット0, 1 (MOD0, 1) によって、TM0とMD0, TM1とMD1で構成されるタイマ0, タイマ1の動作モードを指定します。

MOD0, **MOD1** タイマ0, タイマ1の動作モードを指定するビット

MOD0, 1ビットが、MOD0=0, MOD1=0でインターバル・タイマ動作モードになり、MOD0=1, MOD1=0でワンショット・タイマ動作モードとなります。インターバル・タイマ動作モードでは、TM0, TM1は設定値をダウン・カウントするタイマ・レジスタ、MD0, MD1はインターバルの設定値を保持するモジュロ・レジスタとなります。ワンショット・タイマ動作モードでは、TM0とMD0がともに設定値をダウン・カウントするタイマ・レジスタとなります。なおタイマ1は、TMC1のビット0, 1が“0”固定となり、インターバル・タイマ動作のみです。

したがって、タイマ0はTM0とMD0で構成され、16ビット・インターバル・タイマ、または16ビット・ワンショット・タイマとして動作します。タイマ1はTM1とMD1で構成され、16ビット・インターバル・タイマとして動作します。

また、タイマ0はTOUT端子への矩形波を出力することができます。TMC0レジスタによって、TOUT端子への矩形波出力の制御を行います。ただし、TOUT端子はP15と兼用なので、TOUT端子に矩形波を出力するには、ポート1モード・コントロール・レジスタのビット5 (PMC15) をセット(1)し、コントロール・モードにしてください。このときのTOUT端子への矩形波出力は、CPU内部のタイミ

★

ALV TOUT端子出力のアクティブ・レベルを指定するビット

ENTOビットがリセット(0)時, TOUT端子出力のアクティブ・レベルは, ALVビットがリセット(0)されるとロウ・アクティブ, セット(1)されるとハイ・アクティブとなります。

ENTO TOUT端子への矩形波出力の動作を指定するビット

ENTOビットがリセット(0)されると, インターバル・タイマ・モード時では, TOUT端子レベルはALVビット指定のインアクティブ状態となります。

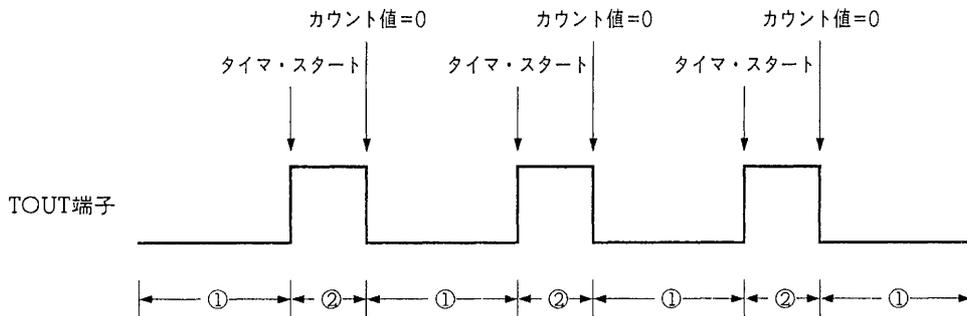
ワンショット・タイマ・モード時はALVビットの設定値により次のように変化します。

★

ALV	変化のタイミング	TOUT端子
0	TSO=1	1
	TMFO=1	0
1	TSO=1	0
	TMFO=1	1

ENTOビットがセット(1)されると, タイマ・ユニット割り込み要求フラグ (TMFO) がセット(1)されるタイミングで, TOUT端子レベルが反転します。

図 9-3 TOUT端子の出力状態 (ワンショット・タイマ・モード時)



① : ALV=1, ENTO=0

② : ALV=0, ENTO=1

タイマ・カウンタをスタートさせるときに, ②に設定します。

ダウン・カウント終了後 (TOUT出力の反転後), ①のように設定してください。

以下, TMC0, TMC1レジスタその他のビットの説明を動作モード別に行います。

(1) インターバル・タイマ・モード (MOD0=0, MOD1=0) : タイマ0, 1

TCLK0, **TCLK1** TM0, TM1レジスタのカウント・クロックを指定するビット

表9-1 にシステム・クロック周波数 (f_{CLK}) が 8 MHz の場合の参考値を示します。



TS0 , **TS1** タイマ0, タイマ1の動作を制御するビット

TS0/TS1ビットをセット(1)すると, TM0/TM1レジスタにMD0/MD1レジスタの値がセットされ, TM0/TM1レジスタはダウン・カウントを開始します。TS0/TS1ビットをクリア(0)するとTM0/TM1レジスタは, TM0/TM1, MD0/MD1レジスタの内容を保持したままダウン・カウントを停止します。

ダウン・カウント中, タイマの値が“0”あるいは, もう一度TS0/TS1ビットをセット(1)すると, 再びTM0/TM1レジスタにMD0/MD1レジスタの値をセットし, ダウン・カウント動作を再度開始します。

表9-1 インターバル・タイマ・モード時のTMnのカウント時間 (n=0, 1)

$f_{CLK} = 8 \text{ MHz}$ のとき

TCLKn	カウント・クロック	分解能	フル・カウント
0	$f_{CLK}/6$	$0.75 \mu\text{s}$	49.1 ms
1	$f_{CLK}/128$	$16 \mu\text{s}$	1.04 s

(2) ワンショット・タイマ・モード (MOD0=1, MOD1=0) : タイマ0のみ

TCLK0 TM0レジスタのカウント・クロックを指定するビット

表9-2にシステム・クロック周波数 (f_{CLK}) が8 MHzの場合の参考値を示します。

TS0 TM0レジスタの動作を制御するビット

TS0ビットをセット(1)すると, そのとき保持されていたTM0レジスタの値からダウン・カウントし, タイマの値が“0”になるとTS0ビットはクリア(0)され, カウント動作を停止します。TS0ビットをクリア(0)すると, TM0レジスタの値を保持したままカウントを停止します。

MCLK0 MD0レジスタのカウント・クロックを指定するビット

表9-2にシステム・クロック周波数 (f_{CLK}) が8 MHzの場合の参考値を示します。インターバル・タイマ・モードに指定されている場合は, MCLK0ビットはカウント動作に影響を与えません。

MS0 MD0レジスタのカウント動作を制御するビット

MS0ビットをセット(1)すると, そのとき保持されていたMD0レジスタの値からダウン・カウントし, タイマの値が“0”になるとMS0ビットはクリア(0)され, カウント動作を停止します。MS0ビットをクリア(0)すると, MD0レジスタの値を保持したままカウントを停止します。

MS0ビットは, インターバル・タイマ動作時にはカウント動作に影響を与えません。

表 9-2 ワンショット・タイマ・モード時のTMOとMDOのカウント時間

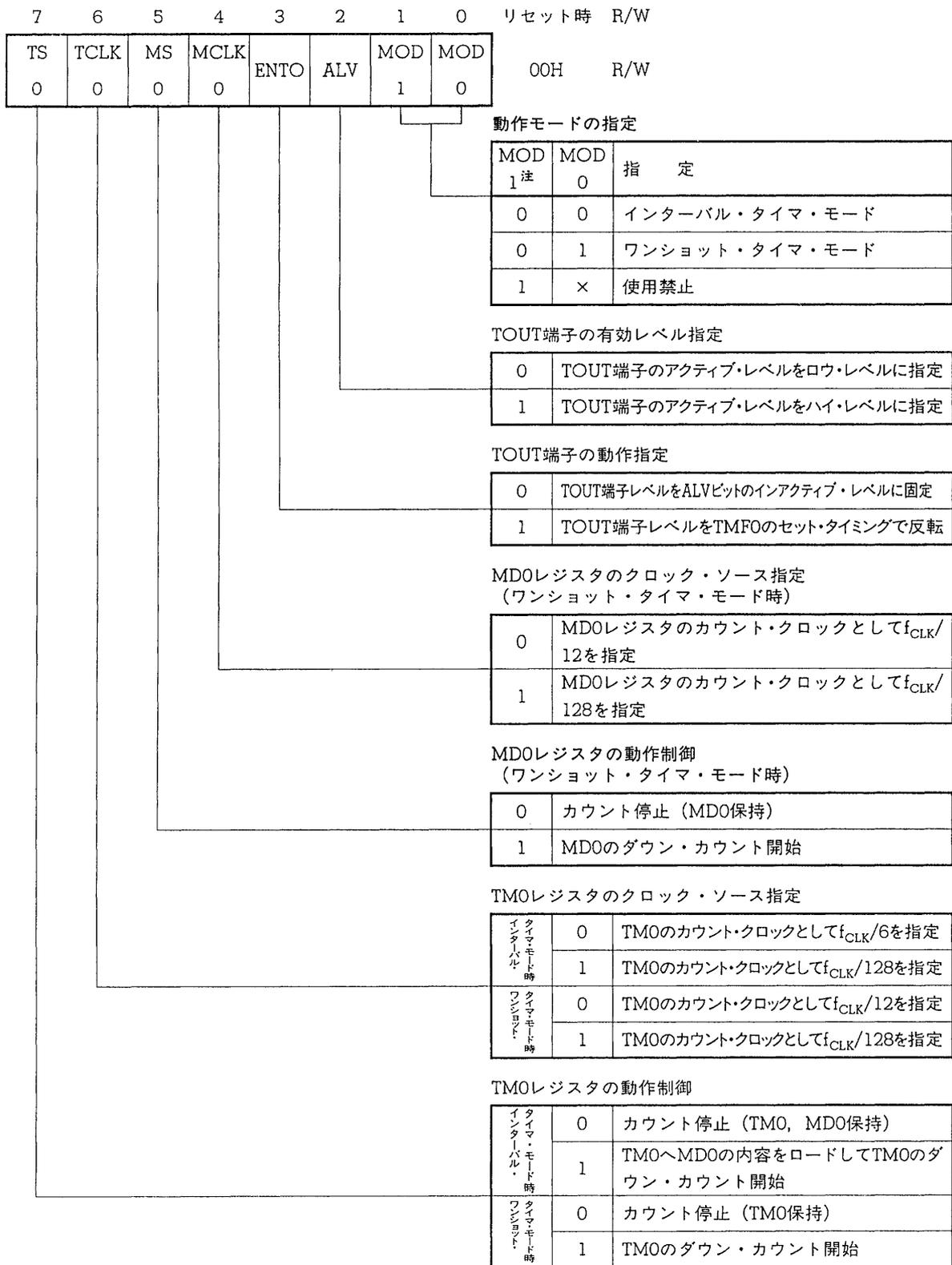
 $f_{\text{CLK}} = 8 \text{ MHz}$ のとき

TCLK0/ MCLK0	カウント・クロック	分解能	フル・カウント
0	$f_{\text{CLK}}/12$	$1.5 \mu\text{s}$	98,3 ms
1	$f_{\text{CLK}}/128$	$16 \mu\text{s}$	1.04 s

注意 TMOレジスタは、インターバル・タイマ・モードに指定された場合とワンショット・タイマ・モードに指定された場合とでは、カウント・クロックが異なります。



図9-4 TMCO



注 MOD1には必ず“0”を書き込んでください。

図 9-5 TMC1



9.3 タイマ・ユニット割り込み要求

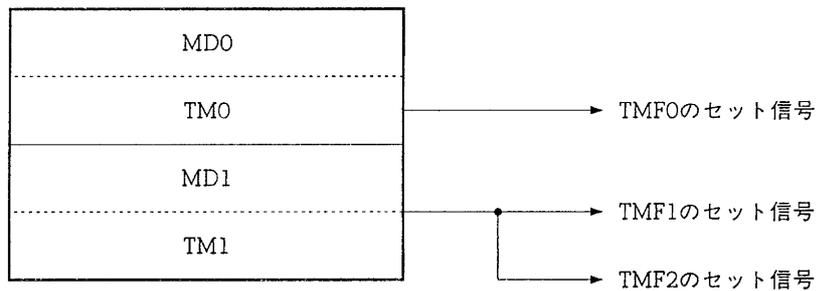
タイマ・ユニットからは3系統の割り込み要求 (TMFO-2) が発生します。これらの発生条件は、タイマの動作モード指定によって異なります。

インターバル・タイマ・モードに設定されている場合、TM0レジスタの値がカウント・ダウンにより“0”となるタイミングでTMFOがセット(1)され、TM1レジスタの値がカウント・ダウンにより“0”となるタイミングでTMF1, TMF2がセット(1)されます (図9-6(a))。

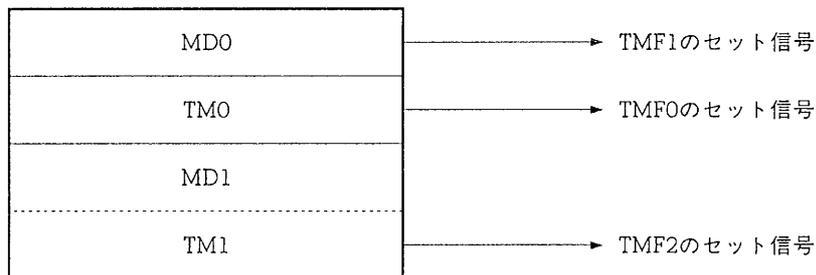
TM0, MD0レジスタがワンショット・タイマ・モードに設定されている場合、TM0レジスタの値がカウント・ダウンにより“0”となるタイミングでTMFOを、MD0レジスタの値がカウント・ダウンにより“0”となるタイミングでTMF1をそれぞれセット(1)します。この場合、TMF2はインターバル・タイマとして動作するTM1レジスタのカウント・ダウンにより“0”に変化するタイミングでセット(1)されます。

図9-6 タイマ・ユニットからの割り込み要求

(a) TM0, MD0をインターバル・タイマ・モードに指定した場合



(b) TM0, MD0をワンショット・タイマ・モードに指定した場合



TMFO-2: タイマ・ユニット割り込み要求フラグ0-2

9.3.1 タイマ・ユニット割り込み要求制御レジスタ (TMICO, TMIC1, TMIC2)

TMIC_nレジスタ (n=0-2) は、タイマ・ユニットから発生する3つの割り込み要求を制御する8ビット・レジスタです。この3つの割り込み要求が1つのグループを構成し、タイマ・ユニット割り込み要求として優先順位がプログラマブルに指定されます。グループ内では、以下のように優先順位が固定されています。

TMF0 > TMF1 > TMF2

図 9-7 TMICO, TMIC1, TMIC2

	7	6	5	4	3	2	1	0
TMICO	TMF0	TMMK0	MS/INT	ENCS	0	PR2	PR1	PRO
TMIC1	TMF1	TMMK1	MS/INT	ENCS	0	1	1	1
TMIC2	TMF2	TMMK2	MS/INT	ENCS	0	1	1	1

注意 TMIC1とTMIC2のビット2-0は“1”に固定されます。

TMIC1とTMIC2の割り込み要求の優先順位は、TMICOのPR2-0の設定に従います。

TMIC_nレジスタの各ビットについては、4.8 割り込み要求制御レジスタを参照してください。

8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み/読み出しができます。この場合、1ウェイトが挿入されます。

TMIC_nレジスタの内容は、RESET入力で47Hに初期化されます。

9.3.2 タイマ・ユニット・マクロ・サービス制御レジスタ (TMMS0, TMMS1, TMMS2)

タイマ・ユニットより発生する3種類の割り込み要求によって起動されるマクロ・サービスを制御する8ビット・レジスタです。

TMMS0レジスタは、TMF0フラグによって起動されるマクロ・サービスを制御します。

TMMS1はTMF1フラグ、TMMS2はTMF2フラグによって起動されるマクロ・サービスを制御します。

8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み/読み出しができます。この場合、1ウェイトが挿入されます。

図9-8 TMMS0, TMMS1, TMMS2

7	6	5	4	3	2	1	0
MSM	MSM	MSM	DIR	0	CH	CH	CH
2	1	0			2	1	0

TMMSnレジスタの各ビットについては、4.5.4 マクロ・サービス制御レジスタを参照してください。

(メ モ)

第10章 タイム・ベース・カウンタ

μ PD70325, 70335は、時計機能用のロング・インターバル・タイマとしての機能を内蔵します。

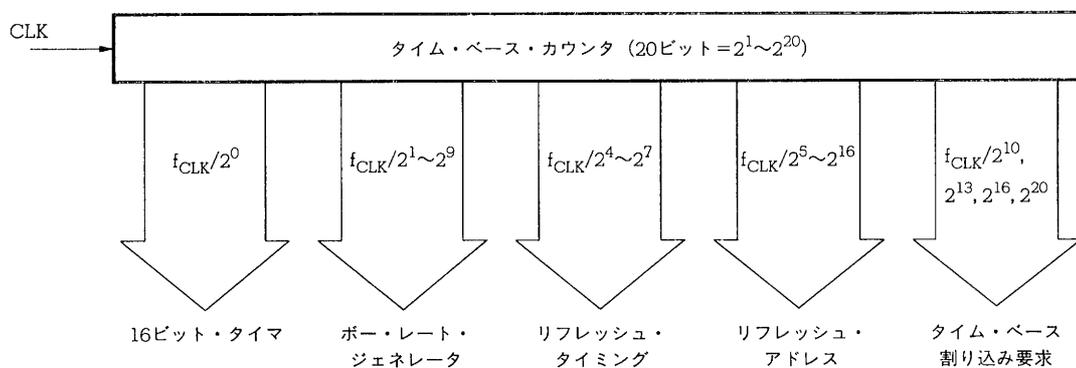
10.1 タイム・ベース・カウンタの構成

タイム・ベース・カウンタの構成図を図10-1に示します。

タイム・ベース・カウンタは、システム・クロック（CLK）を分周する20個の分周器から構成されます。分周期のタップ出力の下位側は、16ビット・タイマのカウンタ・クロック、ポーレート・ジェネレータの入力クロック、リフレッシュ・タイミング発生、リフレッシュ・アドレス生成にそれぞれ使用されます。また20本のタップ出力のうち、10, 13, 16, 20ビット目の出力タップがタイム・ベース割り込みに使用されます。

タイム・ベース・カウンタの内容は、 $\overline{\text{RESET}}$ 入力によってのみ00000Hにクリアされ、その後は常にインクリメントし続けます。

図 10-1 タイム・ベース・カウンタの構成

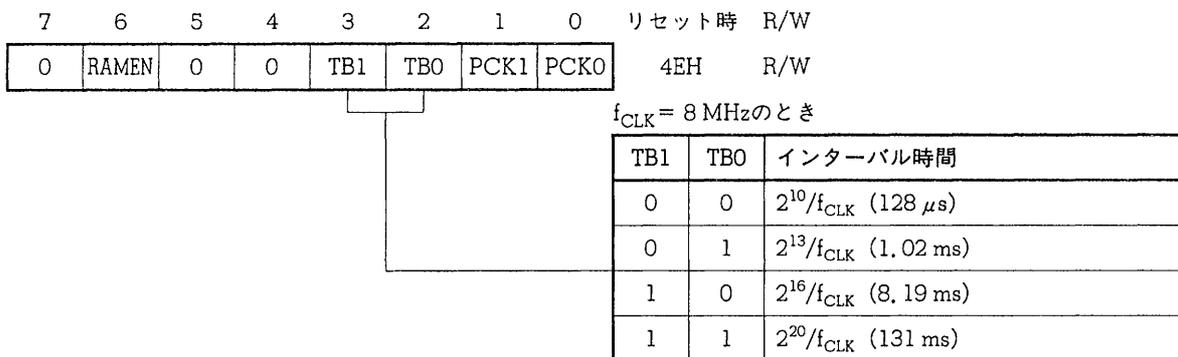


10.2 タイム・ベース・インターバルの指定

タイム・ベース・カウンタから発生する割り込み要求のインターバル時間は、プロセッサ・コントロール・レジスタ（PRC）のビット2, 3（TBO, 1）によって図10-2の4種類から選択することができます。



図 10-2 PRC



注意 TB0, 1ビット設定直後から最初の割り込み要求発生までの時間は不定です。

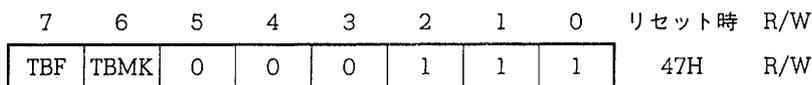
10.3 タイム・ベース割り込み要求制御レジスタ (TBIC)

TBICはタイム・ベース・カウンタから発生する割り込み要求のマスク制御を行うための8ビット・レジスタです。

8ビット操作または1ビット操作のメモリ・アクセスによって、TBICの書き込み/読み出しができません。この場合、1ウエイトが挿入されます。

TBICの内容は、 $\overline{\text{RESET}}$ 入力で47Hに初期化されます。

図 10-3 TBIC



プロセッサ・コントロール・レジスタ (PRC) によって指定されたタイム・ベース・カウンタの出力タップが、ハイ・レベルになるタイミングで、割り込み要求フラグ (TBF) がセット (1) され、割り込み要求が発生します。

TBICのビット4, 5は、“0”固定で、タイム・ベース・カウンタによる割り込みではレジスタ・バンク切り替え機能、マクロ・サービス機能はありません。またTBICのビット0-2は“1”固定によりタイム・ベース割り込み (INTTB) の優先順位は“7”固定であり、優先順位7である他の割り込みがあっても最下位に固定されます。ただし、多重処理制御は受けます。

第11章 シリアル・インタフェース

11.1 シリアル・インタフェースの構成

μ PD70325, 70335は、専用のボー・レート・ジェネレータを内蔵するシリアル・インタフェースを2チャンネル備えています。

シリアル・インタフェースは、スタート/ストップ・ビットによる送受信方式です。スタート・ビットによってデータのビット同期とキャラクタ同期をとるアシンクロナス(調歩同期)・モード、および μ PD7810系などのシリアル・データ転送の方式と同様に制御されたシリアル・クロックに同期してデータの転送を行うI/Oインタフェース・モードの2種類の動作モードを備えています。

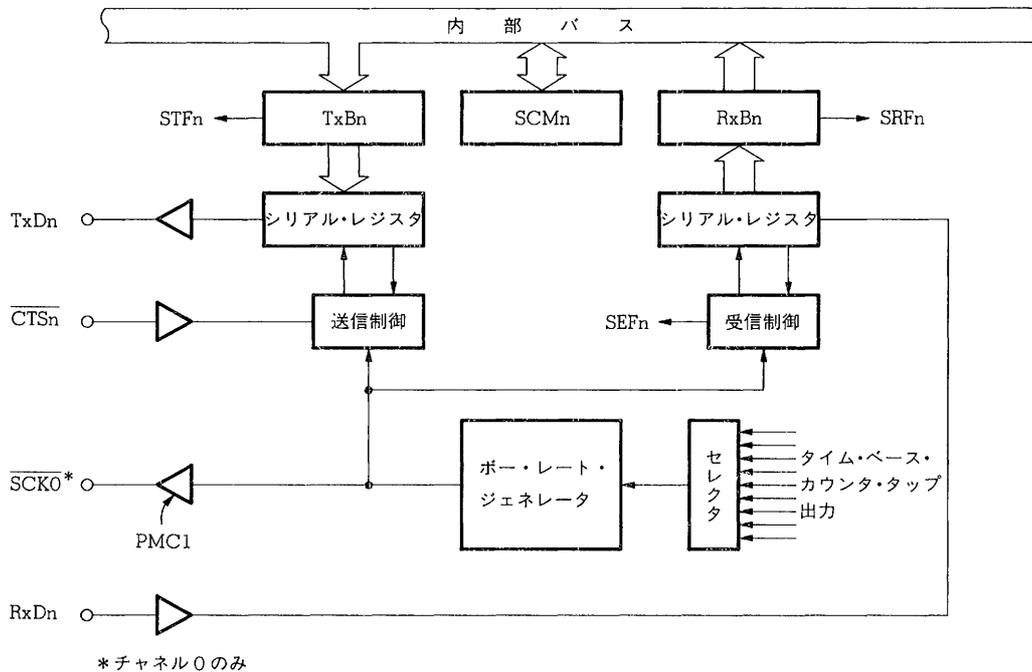
図11-1にシリアル・インタフェースのアシンクロナス・モード設定時と、I/Oインタフェース設定時の構成図を示します。

シリアル・インタフェース部は、シリアル・データ入力 (RxDn)、シリアル・データ出力 (TxDn)、シリアル・クロック出力 ($\overline{\text{SCK0}}$)、送信許可状態の制御入力 ($\overline{\text{CTS}}_n$) の端子^注と、転送制御部、送信、受信用の8ビット・シリアル・レジスタ、送信バッファ (TxBn)、受信バッファ (RxBn) およびボー・レート・ジェネレータで構成されます。送信、受信用にそれぞれシリアル・レジスタとバッファを備えているので、送信、受信が独立に行えます (全二重動作可能)。また、I/Oインタフェース・モードの場合、 $\overline{\text{CTS}}_n$ 端子が、受信クロックの入出力端子の機能を持つので、I/Oインタフェース・モードにおいても、全二重でシリアル動作が可能です。

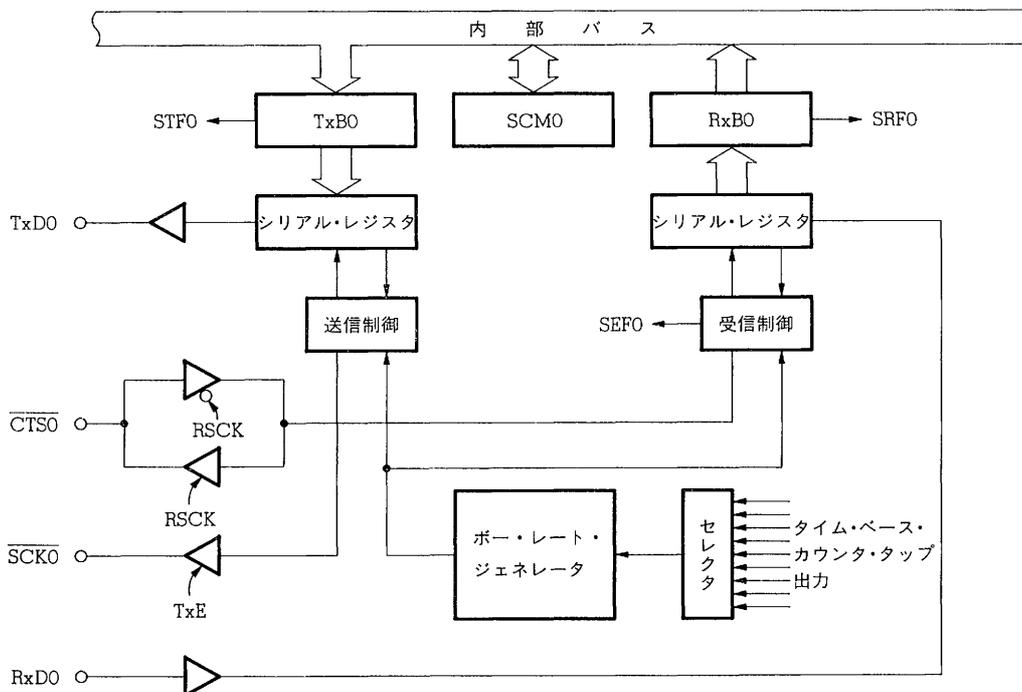
注 A.3 シリアル送信中の送信禁止操作による送信データ抜けに関する制限事項を参照してください。

図 11-1 シリアル・インタフェースの機能

(a) アシンクロナス・モード設定時 (n=0, 1)



(b) I/Oインタフェース・モード設定時 (チャンネル0のみ)



11.2 アシクロナス・モード

アシクロナス・モードの場合、キャラクタ長、ストップ・ビット数、パリティ許可、奇偶数パリティの指定をシリアル・モード・レジスタ (SCMn) で制御できます。

11.2.1 送信

シリアル・モード・レジスタ (SCMn) のビット 7 (TxRDY) がセット (1) され、かつ $\overline{\text{CTS}}_n$ 端子がアクティブ (0) のとき、送信許可状態^注となります。

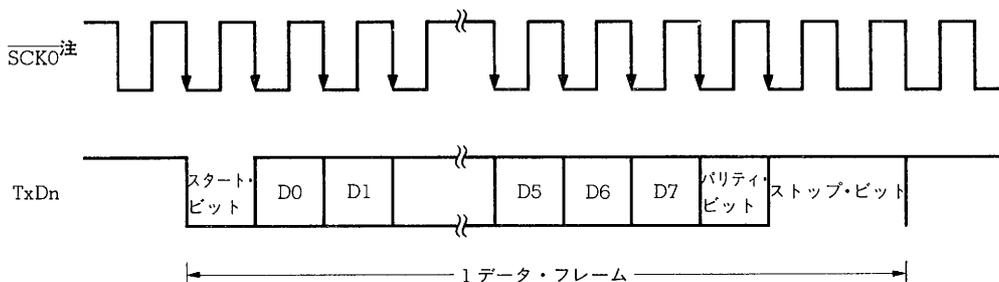
送信の起動方法には、次の 3 通りがあります。

- (a) 送信バッファ (TxB) が空の状態に送信許可状態に設定することにより、送信完了割り込み要求を発生させ、割り込み処理内で送信バッファへ送信データの書き込み動作を行います。
- (b) 送信許可状態で送信バッファに送信データを転送すると、1 つ前の送信動作が終了後、この送信データの送信が連続して行われます。
- (c) 送信禁止状態で送信バッファにあらかじめ送信データを書き込んでおいて、その後、送信許可状態にすると、送信バッファ内に保持されていたデータが送信されます。

注 送信許可状態に設定する手順には特別の制限はなく、TxRDY = "1" にしておいて $\overline{\text{CTS}}_n$ 端子をアクティブとすることも、 $\overline{\text{CTS}}_n$ 端子をアクティブにしておいて TxRDY = "1" に設定することも可能です。

送信データのフォーマットは、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 データ・フレームを構成します (図 11-2 参照)。送信データは TxDn 端子より最下位ビット (LSB) から送信クロックの立ち下がりに同期して送出されます。送信クロックはチャンネル 0 のみ $\overline{\text{SCKO}}$ 端子から出力されます (チャンネル 1 については出力されません)。この送信クロックは、チャンネル 0 のボーレートがそのままクロック数として出力されます。TxDn 端子は、送信禁止状態のとき、あるいはシリアル・レジスタに送信するデータを持たないとき、マーク状態 (1) になります。

図 11-2 送信データのフォーマット



注 チャンネル0のみ

スタート・ビット……………1ビット

キャラクタ・ビット……7/8ビット

パリティ・ビット……………奇/偶/0/付加しない

ストップ・ビット……………1/2ビット

11.2.2 送信完了割り込み

送信完了割り込み要求は、送信バッファ (TxBn) が空になるとただちに発生します。

RESET入力により、送信バッファ (TxBn) は空の状態になります。このとき、送信許可状態に設定すると送信完了割り込み要求が発生します。また、送信動作の開始により送信バッファ内の送信データがシフト・レジスタに転送されると送信バッファは空の状態となり、送信完了割り込み要求が発生します。

★ 送信完了割り込み要求の発生ごとに送信データを送信バッファに書き込むことによりマーク状態(1)が挿入されることなく、連続的なデータ送信が可能です。送信禁止状態にすると、送信バッファからシフト・レジスタへの転送が禁止されます。

送信動作が行われている間に送信禁止状態に切り替えられた場合、送信中のデータは1フレーム分最後まで送信されます。ただし、すでに新しい送信データが送信バッファに書き込まれていると、送信バッファからシフト・レジスタへの転送は禁止され、送信バッファの内容はそのまま保持されます。再び送信許可状態に設定されると、このタイミングに同期して送信バッファの内容がシフト・レジスタに転送され、送信が開始されると同時に送信完了割り込み要求が発生します。

★ 注意 送信完了割り込みをマクロ・サービスで使用している場合、MSC=0となるマクロ・サービスが長い時間保留されて、シリアル・レジスタ、送信バッファとともに空になった状態でマクロ・サービス、マクロ・サービス完了割り込みが受け付けられたとき、マクロ・サービス完了割り込み処理ルーチン内で、送信許可→送信禁止→送信許可の操作を行ってください。この操作をしないと以降の送信完了割り込みが発生しないため、送信動作が停止します（たとえば、送信割り込みより優先順位の高い割り込み処理の時間が長い場合）。ただし、それ以外の動作は正常に動作します。

11.2.3 受信

シリアル・モード・レジスタ (SCM2) のビット6 (RxE) がセット(1)されたとき受信許可状態となります。受信禁止状態 (RxE=0) では、受信ハードウェアは初期状態で待機します。

RxDn端子入力をポー・レート・ジェネレータへの入力クロックでサンプリングし、立ち下がりエッジを検出すると受信動作が起動し、受信用ポー・レート・ジェネレータがカウントを開始します。受信用ポー・レート・ジェネレータからの最初のタイミング信号でRxDn端子入力のロウ・レベルを検知するとスタート・ビットとして認識して以降の受信動作を行います。最初のタイミング信号でハイ・レベルを検出すると、スタート・ビットとして認めず、ポー・レート・ジェネレータは初期化されて動作を停止します。

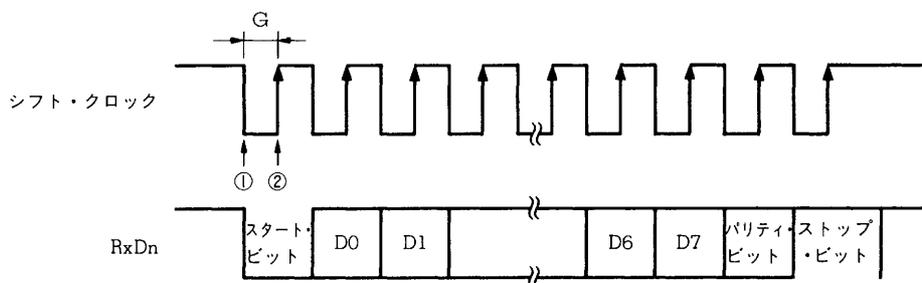
受信データのサンプリングは、スタート・ビットの検出後シフト・クロックの立ち上がりにより同期して行われます（図11-3参照）。

スタート・ビットの検出は、ポー・レート・ジェネレータへの入力クロック（SCCレジスタにより選択される）により行われます。ポー・レートは、このクロックを2分周したものをBRGレジスタの設定値（G）だけカウントすることにより生成されます。

つまり、ポー・レート・ジェネレータへの入力クロックによるサンプリングで、受信端子（RxD）のロウ・レベルを検出した後、ポー・レート・ジェネレータへの入力クロックのエッジによりGをカウントした時点で、スタート・ビットの検出を行います。ここでスタート・ビットを検出すると、それ以降ポー・レート・ジェネレータへの入力クロックのエッジで2Gをカウントすることにより、受信データの取り込みを行います。

図11-3 受信データのサンプル・タイミング

★



- ① RxD端子入力をポー・レート・ジェネレータへの入力クロックでサンプリングして、立ち下がりエッジを検出すると受信動作が起動します。
- ② 受信用ポー・レート・ジェネレータが、カウントを開始します。

11.2.4 受信完了割り込み

受信割り込み要求は、シリアル・モード・レジスタのビット3（CL）で指定されたキャラクタ長の受信が完了すると、シフト・レジスタ内の受信データが受信バッファ（RxBn）に転送され、受信完了割り込み要求が発生します。

受信時に、奇偶数パリティのチェックを行い（PRTY1ビット^注=1の場合）不一致のとき（パリティ・エラー）、ストップ・ビットがロウ・レベルのとき（フレーミング・エラー）、または、受信バッファがフルで次のデータが受信バッファに転送されたとき（オーバラン・エラー）に、受信エラー・フラグがセットされ、受信エラー割り込み要求が発生します（11.7参照）。

注 PRTY1ビットはシリアル・モード・レジスタのビット5です。

注意 受信時にRxEビットをクリアし、受信禁止状態とした場合、受信中のキャラクタは保証されません。

11.3 I/Oインタフェース・モード

I/Oインタフェース・モードは μ PD7810などのシリアル・インタフェースと同一のモードで、外部にI/Oを拡張する場合、あるいはI/Oコントローラ(A/Dコンバータ、液晶コントローラなど)を接続する場合に有効です。

I/Oインタフェース・モードの場合には、キャラクタ長8ビット固定、パリティ・ビットなしで、最上位ビット(MSB)よりデータ転送を行います。

I/Oインタフェース・モードはチャンネル0でのみ使用できます。

11.3.1 送信

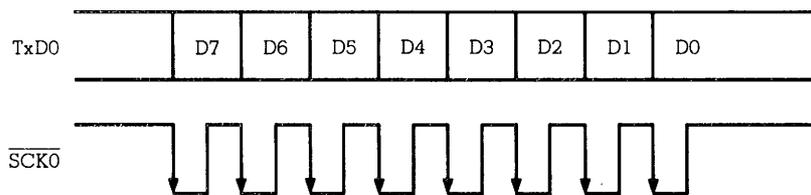
シリアル・モード・レジスタのビット7(TxE)がセットされたとき、送信許可状態となります。I/Oインタフェース・モード時、 $\overline{\text{SCK0}}$ 端子が送信クロック出力端子となります。

送信クロックを外部から入力することはできません。

送信動作の起動には、アシンクロナス・モードと同様に次の3通りがあります。

- (a) 送信バッファ(TxB0)が空の状態を送信許可状態に設定することにより、送信完了割り込み要求が発生させ、割り込み処理内で送信バッファへ送信データの書き込み動作を行います。
- (b) 送信許可状態を送信バッファ(TxB0)に送信データを転送すると、1つ前の送信動作が終了後、この送信データの送信が連続して行われます。
- (c) 送信禁止状態を送信バッファ(TxB0)にあらかじめ送信データを書き込んでおいて、その後、送信許可状態にすると、送信バッファ(TxB0)内に保持されていたデータが送信されます。

送信データのフォーマットは下図に示すとおりです。8ビット固定で最上位ビット(MSB)から送信されます。



11.3.2 送信完了割り込み

送信完了割り込み要求は、送信バッファ(TxB0)が空になるとただちに発生します。

$\overline{\text{RESET}}$ 入力により送信バッファ(TxB0)は空の状態になります。このとき、送信許可状態に設定すると送信完了割り込み要求が発生します。また、送信動作の開始により送信バッファ(TxB0)内の送信データがシフト・レジスタに転送されると送信バッファは空の状態となり、送信完了割り込み要求が発生します。

★ 注意 送信完了割り込みをマクロ・サービスで使用している場合、MSC=0となるマクロ・サービス

スが長い時間保留されて、シリアル・レジスタ、送信バッファとともに空になった状態でマクロ・サービス、マクロ・サービス完了割り込みが受け付けられたとき、マクロ・サービス完了割り込み処理ルーチン内で、送信許可→送信禁止→送信許可の操作を行ってください。この操作をしないと以降の送信完了割り込みが発生しないため、送信動作が停止します（たとえば、送信割り込みより優先順位の高い割り込み処理の時間が長い場合）。ただし、それ以外の動作は正常に動作します。

11.3.3 受信

シリアル・モード・レジスタ (SCM0) のビット 6 (RxE) がセット (1) されたとき受信許可状態となります。受信データは受信クロックの立ち上がりエッジでシリアル・レジスタに入力されます。シリアル・レジスタが 8 ビットのデータを受信するとシリアル・レジスタから受信バッファ (RxB0) へデータを転送し、受信完了割り込み要求が発生します。

I/Oインタフェース・モードにおける受信クロックは、シリアル・モード・レジスタ (SCM0) のビット 2 (RSCK) の指定で外部受信クロックと内部受信クロックの両方から選択できます。

内部受信クロックを使用する場合は、SCM0のTSKビットに 1 を書き込むことによって起動します。★
したがって、マクロ・サービスは使用できません。

また、I/Oインタフェース・モード時は、 $\overline{\text{CTS0}}$ 端子が受信クロックの入出力端子として機能します。

注意 受信時にRxEビットをクリアし、受信禁止状態とした場合、受信中のキャラクタは保証されません。

11.3.4 受信エラー割り込み

受信時に、受信バッファ (RxB0) がフルで次のデータが受信バッファに転送されたとき (オーバラン・エラー)、受信エラー・フラグがセットされ、受信エラー割り込み要求が発生します。

11.3.5 シリアル・レジスタのクリア

シリアル・レジスタをクリアするには、次のどちらかの操作を行います。

- SCC0レジスタへの書き込み
- SCM0レジスタのMD0ビットの変更

(I/Oインタフェース・モード→アシンクロナス・モード→I/Oインタフェース・モード)

11.4 割り込みを使用しない場合の送信の起動

割り込みを使用しない場合の送信の起動には次の 2 通りあります。

- (1) 送信バッファ (TxB) が空の状態を送信許可状態に設定すると、割り込み制御レジスタ (STICn) のビット 7 (STFn) がセット (1) されるので、一度クリア (0) します。次に送信バッファへ送信デー

タの書き込みを行ったあと、STICnのビットをポーリングすることにより、送信を行います。
 (2) 送信禁止状態で送信バッファ(TxB)に1バイト目のデータを書き込んだあと、送信許可状態に設定すると、送信バッファに保持されていたデータがシリアル・レジスタへダウンロードされ、データが送出されます。

備考 STICnレジスタのビット7(STFn)は、送信完了割り込みが受け付けられるまでクリア(0)されません。したがって、割り込みを使用しない場合には、送信バッファにデータを書き込むときにSTFnフラグをソフトウェアでクリア(0)してください。

11.5 シリアル・モード・レジスタ (SCM0, SCM1)

SCMnレジスタ (n=0, 1) は、シリアル・インタフェースの転送モードを指定する8ビット・レジスタで、チャンネル0 (SCM0)、チャンネル1 (SCM1) の両方にそれぞれ設定できます。SCMnのビット2からビット7の意味付けは、ビット1, 0 (MD1, MD0) の指定によって異なります。

7	6	5	4	3	2	1	0
						MD	MD
						1	0

MD1, MD0=0, 1 (アシンクロナス・モード)

7	6	5	4	3	2	1	0
TxRDY	RxE	PRTY	PRTY	CL	SL	0	1
1		0					

MD1, MD0=0, 0 (I/Oインタフェース・モード)

7	6	5	4	3	2	1	0
TxE	RxE	0	0	TSK	RSCK	0	0

MD1, MD0ビットは、シリアル・インタフェースの転送モードを指定するビット・フィールドです。MD1, MD0=0, 1に設定するとアシンクロナス・モードとなり、MD1, MD0=0, 0に設定すると、I/Oインタフェース・モードとなります。ただし、I/Oインタフェース・モードは、SCM0のみ設定できます。

8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み／読み出しができます。

$\overline{\text{RESET}}$ 入力では内容はOOHにクリアされます。

シリアル送信の場合、送信データがTxBからシリアル・レジスタへ転送されることにより、割り込み要求フラグ (STF_n) がセットされたあと、TxD端子へ1ビットずつデータが送出されます。

そのため、もしSTF_nがセットされた直後に、シリアル送信動作中の状態で、シリアル・モード・レジスタ (SCM_n) を操作すると操作以降の送信動作が正常にできないことがあります。

したがって、シリアル・モード・レジスタ (SCM_n) はデータをセットしたあとに送信動作が完了するのに十分な時間をとってから操作してください。

(1) アシンクロナス・モード設定時

RxE 受信許可制御を行うビット

受信動作中に受信禁止状態 (RxE=0) にすると受信処理は中断され、受信完了の割り込みは発生しません。

SL ストップ・ビット数を指定するビット

SLビットが、リセット(0)されるとストップ・ビットは1ビット、セット(1)されると2ビットとなります。

CL キャラクタ長を指定するビット

CLビットが、リセット(0)されると7ビット長、セット(1)されると8ビット長となります。

PRTY0, **PRTY1** パリティの付加を指定するビット

PRTY0, 1ビットは、パリティなし、奇数パリティ、偶数パリティ、0パリティの指定をします。

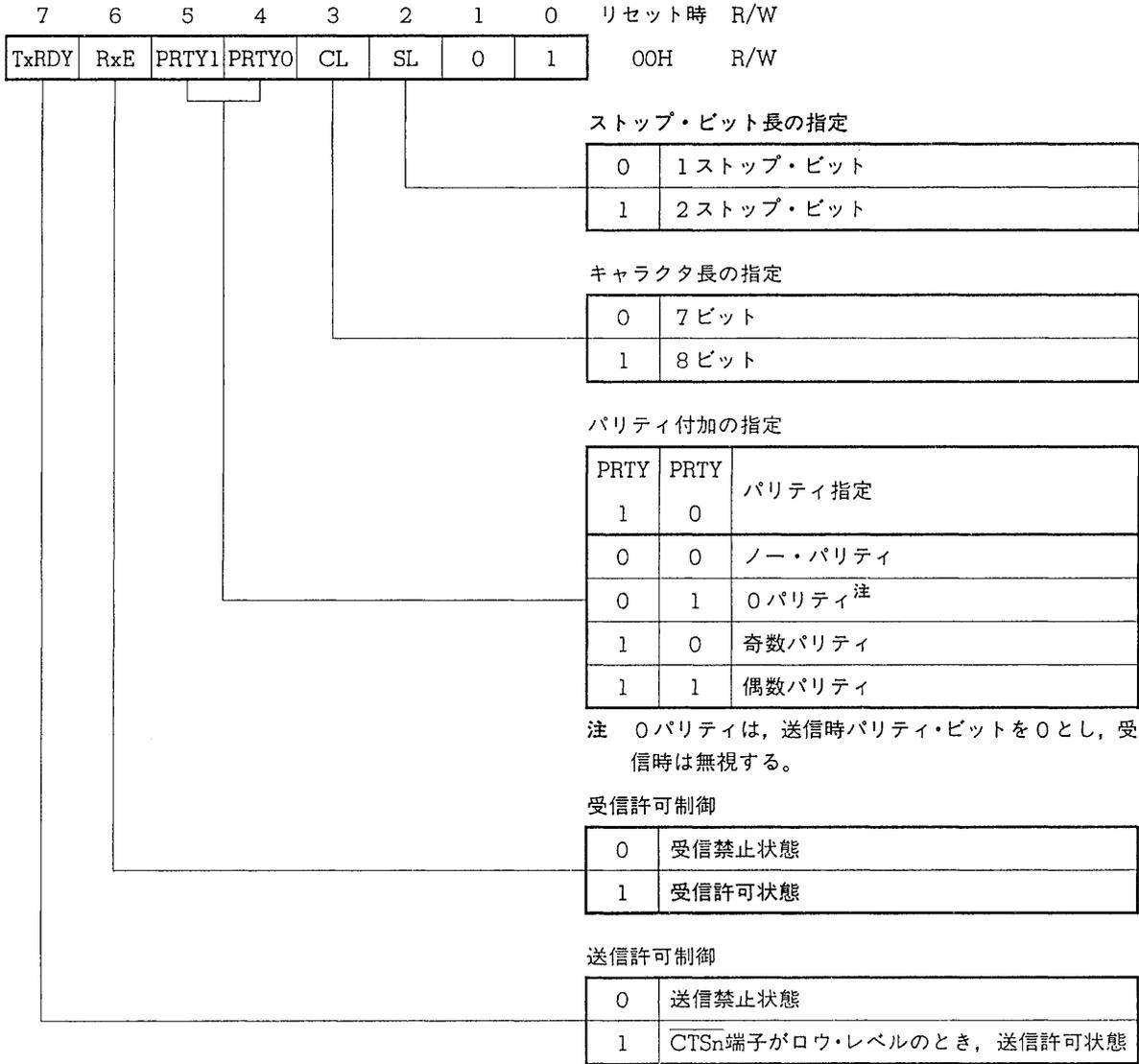
0パリティは、パリティ・ビットを送信時“0”とし、受信時には無視します。

TxRDY 送信許可状態を制御するためのビット

$\overline{\text{CTS}}_n$ 端子がロウ・レベルでかつTxRDY=1のとき、送信許可状態となります。



図 11-4 SCMO, SCM1 (アシンクロナス・モード設定時)



(2) I/Oインタフェース・モード設定時

RSCK シリアル受信クロックのソースを指定するビット

RSCKビットがリセット(0)されると外部受信クロックで受信動作を行い、RSCKビットがセット(1)されると内部受信クロックで受信動作を行います。受信クロックの入出力は、CTS₀端子によって行われます。

TSK 受信クロックの出力トリガ・ビット

RSCKビットがセット(1)されているときのみ有効となり、TSKビットに1を書き込むと8発の受信用シフト・クロックがCTS₀端子から出力されます。

シリアル・クロックが出力されると、このビットは自動的にリセット(0)されます。

RxE 受信許可制御を行うビット

RxEビットがセット(1)されると受信許可状態となり、リセット(0)されると受信禁止状態となります。受信動作中に受信禁止状態にすると受信処理はその時点で中止され、受信完了割り込み要求は発生しません。

TxE 送信許可制御を行うビット

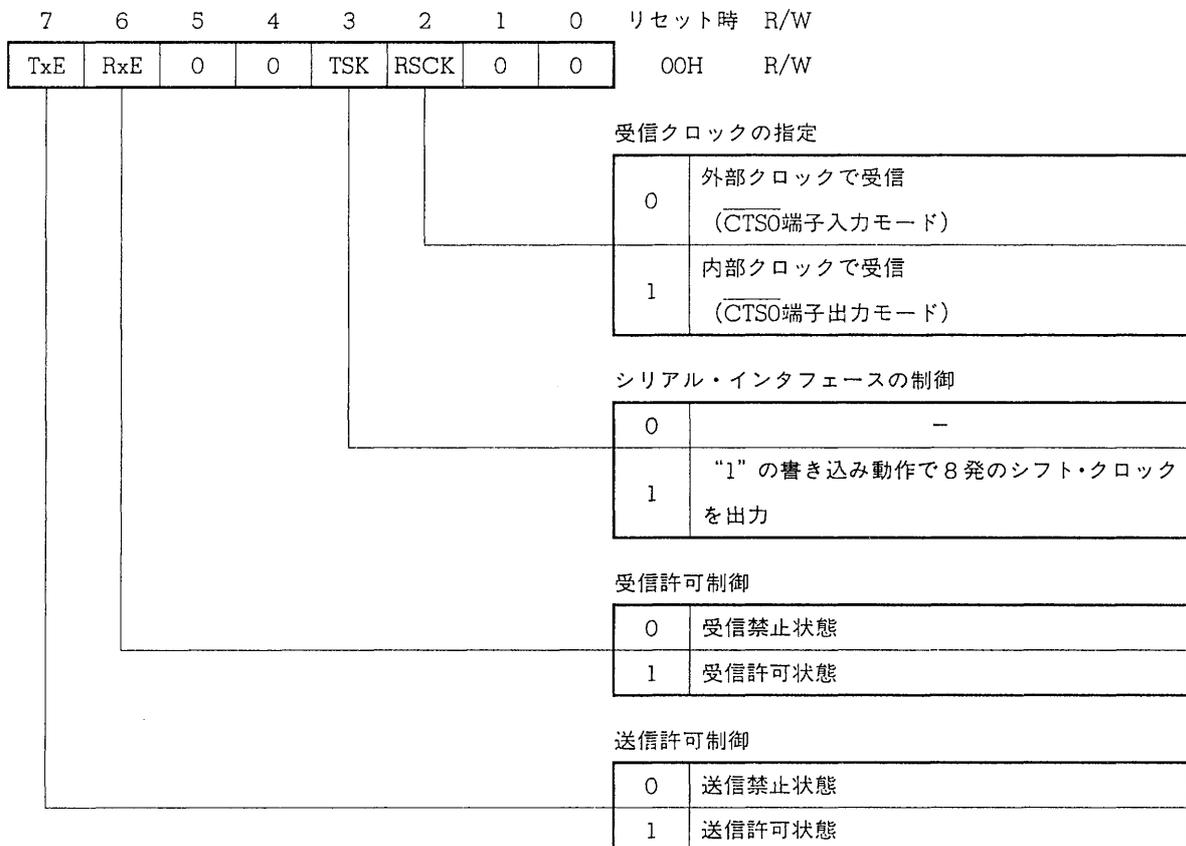
TxEビットがセット(1)されると送信許可状態となり、リセット(0)されると、送信禁止状態となります。

送信許可状態 (TxE=1) で送信バッファに転送データを書き込むと、送信中であれば送信処理を完了してから、送信が行われていなければただちに、対応するシリアル送信を開始します。

また、送信禁止状態 (TxE=0) で送信バッファに送信データを書き込むと、シリアル送信は行われず、送信バッファ内のデータはそのまま保持されます。その後、送信許可状態への切り替えと同時にバッファ内に保持されている送信データの送信処理が開始されます。

送信動作中にTxEビットをリセット(0) (送信禁止状態) しても送信動作は最後まで行われま
す。ただし、禁止状態に設定した時点で、すでに送信バッファ内に次の送信データが格納されて
いるときには、この送信データに対する次の送信が保留され、そのままバッファ内に保持されま
す。

図 11-5 SCMO (I/Oインタフェース・モード設定時)



11.6 ボー・レート・ジェネレータ

ボー・レート・ジェネレータは、送受信のシフト・クロックを発生するシリアル・インタフェース専用の8ビット・タイマです。各チャンネルごとに、送信用ボー・レート・ジェネレータと受信用ボー・レート・ジェネレータを備えています。ボー・レートは送受信とも同じものを使用し、ボー・レート・ジェネレータ・レジスタ (BRGn) へ値を書き込んでボー・レートが決定します。ただし、ボー・レートは最大750 kbpsです。750 kbps以上のデータ送信/受信は、データ間に1シフト・クロック以上のアイドル時間を挿入することによって可能となります。

ボー・レート・ジェネレータへの入力クロックの指定は、シリアル制御レジスタ (SCCn) のPRS3-0でタイム・ベース・カウンタ (10.1参照) の出力タップを選択することによって行います。シリアル・インタフェースのシフト・クロックには、ボー・レート・ジェネレータの出力信号を用いています。ボー・レートと各パラメータの関係は次のとおりです。

$$B \cdot G = 10^6 \times \frac{f_{\text{CLK}}}{2^{n+1}}$$

各パラメータの定義を次に示します。

B : 転送ボー・レート [bps]

B = 110, 150, …… , 9600, 19200, ……

G : ボー・レート・ジェネレータ・レジスタ (BRGn) への設定値 ($2 \leq G \leq 255$)

n : シリアル制御レジスタで指定するボー・レート・ジェネレータへの入力クロック指定番号 ($0 \leq n \leq 8$)

CLK : システム・クロック周波数 [MHz]

★ 2種類のモードについて説明します。

・I/Oインタフェース・モード時 (クロック同期)

ボー・レートの誤差は許されません。送受信クロックの立ち上がりに対するセットアップ時間とホールド時間でデータ・タイミングが規定されます。

・アシンクロナス・モード時 (非同期)

ボー・レート = $(1/G) \times 10^6 \times f_{\text{CLK}} / 2^{n+1}$ であり、ボー・レートの誤差は f_{CLK} に依存します。

Gが小さければ、 f_{CLK} の周波数偏差の影響が大きくなります。このため、Gを小さくするとボー・レートの許容誤差は小さくなります。

16 MHzのクリスタルを外付け使用した場合の、各標準ボー・レートに対するボー・レート・ジェネレータの設定値を表 11-1 に示します ($f_{\text{CLK}} = f \times 12$ のとき)。

表 11-1 ポー・レート・ジェネレータの設定値 (参考)

 $f_{CLK}=8\text{ MHz}$ のとき

ポー・レート	n	BRGnレジスタ 設定値G	誤差 (%)
110	8	142	0.03
150	7	208	0.16
300	6	208	0.16
600	5	208	0.16
1200	4	208	0.16
2400	3	208	0.16
4800	2	208	0.16
9600	1	208	0.16
19200	0	208	0.16
38400	0	104	0.16

n: ポー・レート・ジェネレータへの入力クロックの指定番号

ポー・レートの誤差の算出方法を以下に示します。

★

$$\sqrt{\left\{ \frac{\left[\left(\begin{array}{l} \text{各パラメータを基に} \\ \text{算出するポー・レート} \end{array} \right) - \left(\begin{array}{l} \text{要求の} \\ \text{ポー・レート値} \end{array} \right) \right]^2}{\text{要求値のポー・レート値}} \right\}} \times 100 (\%)$$

表11-1にあるポー・レートの誤差とは、パラメータに従って設定を行ったときの要求値に対する誤差のことをいいます。

11.6.1 シリアル制御レジスタ (SCC0, SCC1)

SCCnレジスタ (n=0, 1) は、シリアル・インタフェースの転送レートを制御するレジスタです。

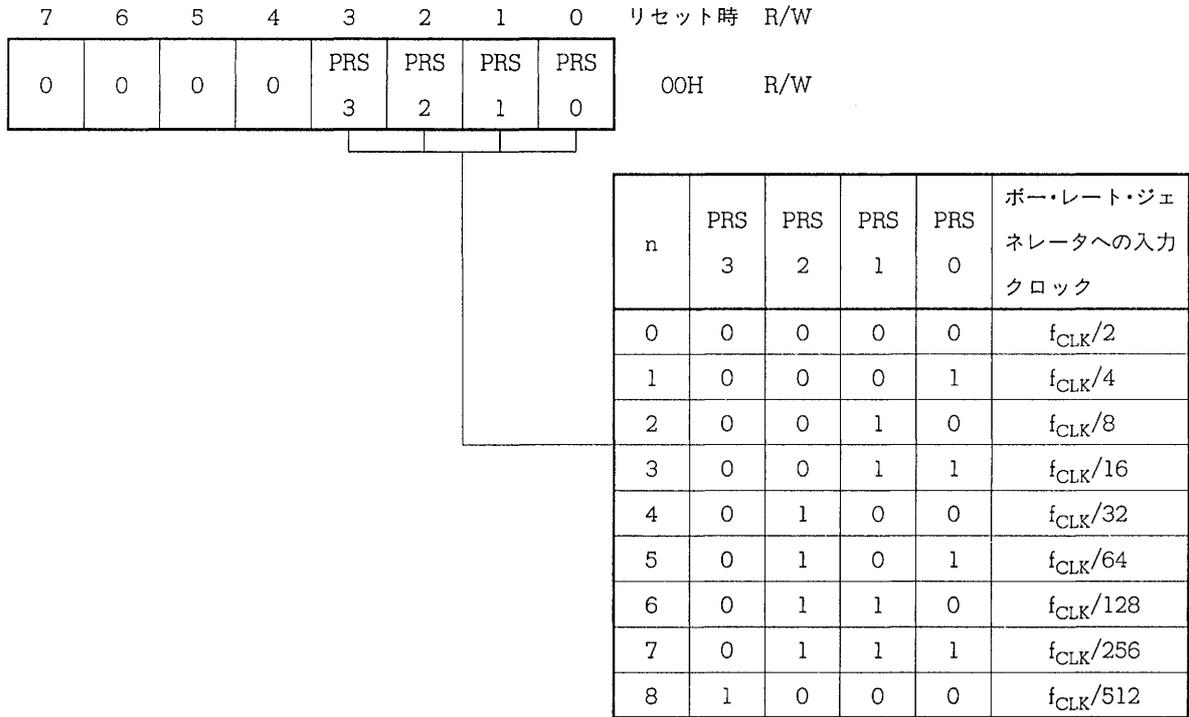
8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み/読み出しができます。

$\overline{\text{RESET}}$ 入力により、内容は00Hに初期化されます。

PRS3-0のビット・フィールドで、ポー・レート・ジェネレータに入力するタイム・ベース・カウンタの出力タップを指定します。



図 11-6 SCC0, SCC1



f_{CLK} : システム・クロック周波数

n : ポー・レート・ジェネレータへの入力クロックの指
定番号

11.7 シリアル・エラーと送受信状態の検出

シリアル・インタフェースの受信エラーを、以下の3種類、検出することができます。

- (a) パリティ・エラー (アシンクロナス・モード)
パリティ演算結果と受信パリティが一致しない。
- (b) フレーミング・エラー (アシンクロナス・モード)
ストップ・ビットが検出されない。
- (c) オーバラン・エラー (アシンクロナス・モード, I/Oインタフェース・モード)
RxBから前の受信データを引き取る前に、次の受信が完了した。

また、送受信バッファについて、以下の2種類の状態を検出することができます。

- (a) 送信バッファ・エンプティ状態
- (b) 受信バッファ・フル状態

11.7.1 シリアル・ステータス・レジスタ (SCSO, SCS1)

受信エラー状態、受信端子状態、送受信バッファのデータ格納状態を示す8ビット・レジスタです。チャンネル0、1の両方にそれぞれ用意されています。

8ビット操作のメモリ・アクセスによって、これらのレジスタの読み出しのみができます。

各エラー・フラグは次のデータ受信終了時に更新され、それまでは以前の内容が保持されます。

$\overline{\text{RESET}}$ 入力で、SCSnの内容は60Hに初期化されます。

ERPn パリティ・エラー・フラグ

ERPフラグは、送信パリティと受信パリティが一致しないときにセット(1)され、次のデータ受信時に送信パリティと受信パリティが一致すればリセット(0)されます。

ERFn フレーミング・エラー・フラグ

ERFフラグは、ストップ・ビットが検出されないときにセット(1)され、次のデータ受信時にストップ・ビットが検出されるとリセット(0)されます。

EROn オーバラン・エラー・フラグ

EROフラグは、RxBから前の受信データを引き取る前に、次の受信が完了したときにセット(1)され、受信バッファからの受信データ読み出し後のデータ受信終了時にリセット(0)されます。

TxBEn 送信バッファ・エンプティ・フラグ

送信バッファ空であることを示すフラグ。

送信動作により送信バッファ内の送信データがシフト・レジスタに転送され、送信バッファが空状態になったとき、または、ポー・レート・ジェネレータ・レジスタ (BRGn)、シリアル制御レジスタ (SCCn) の値を書き換えたときセット(1)されます。また、送信バッファに送信データを書き込んだときリセット(0)されます。

RxBFn 受信バッファ・フル・フラグ

受信バッファに受信データが格納されていることを示すフラグ。

受信動作によりシフト・レジスタから受信データが受信バッファに転送されたとき、受信エラー状態に関係なくセット(1)され、受信バッファから受信データを読み出したとき、または、ポー・レート・ジェネレータ・レジスタ (BRGn)、シリアル制御レジスタ (SCCn) の値を書き換えたときリセット(0)されます。

ASn オール・セント・フラグ

送信バッファおよび送信シフト・レジスタが空であり、どちらにもデータが残っていないことを示すフラグ。

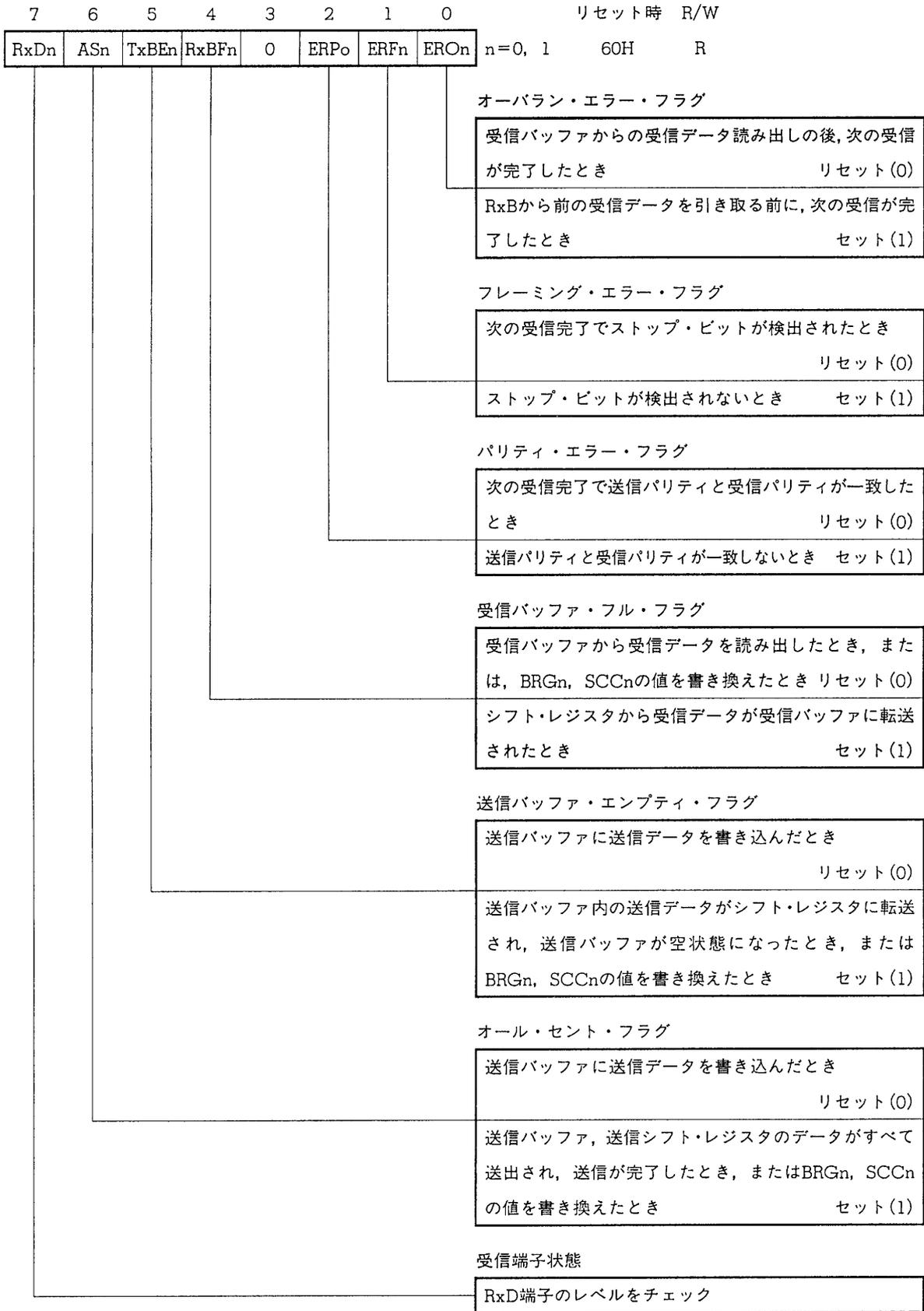
送信動作により送信バッファおよび送信シフト・レジスタのデータがすべて送出され送信が完了したとき、または、ポー・レート・ジェネレータ・レジスタ (BRGn)、シリアル制御レジスタ (SCCn) の値を書き換えたときセット(1)されます。また、送信バッファに送信データを書き込んだときリセット(0)されます。

RxDn RxD端子の受信状態をチェックするビット

このビットには、RxD端子のレベルがセットされます。



図 11-7 SCS0, SCS1



11.8 ブレーク検出機能

μ PD70325, 70335では、ソフトウェア処理で回線のブレーク状態を検出することができます(アシンクロナス・モードのみ)。以下にブレーク状態検出の手順を示します。

(1) 1回目のフレーミング・エラーによる受信エラー割り込みの発生

受信エラー処理ルーチン内で受信データをチェックし、00Hであることを確認します。

同時に受信エラー・フラグをチェックし、フレーミング・エラーであることを確認します。

(2) 2回目のフレーミング・エラーによる受信エラー割り込みの発生

ブレーク状態のときは、再びフレーミング・エラーが発生します。

受信データが再び00Hであり、連続して00Hのデータを、フレーミング・エラーを伴いながら受信したということ、およびシリアル・ステータス・レジスタ (SCSn) のビット7 (RxDn) を使用して直接端子状態を確認することで、回線がブレーク状態であると判断します。

11.9 シリアル・インタフェース割り込み要求

シリアル・インタフェースから発生する割り込み要求には、送信完了割り込み要求、受信完了割り込み要求、受信エラー割り込み要求の3種類が2チャンネルそれぞれに対応してあります。

11.9.1 割り込み要求制御レジスタ (SEICn, SRICn, STICn : n=0, 1)

シリアル・インタフェースから発生する受信エラー割り込み要求 (SEFn), 受信完了割り込み要求 (SRFn), 送信完了割り込み要求 (STFn) の3本の割り込み要求を制御するレジスタです。

3つの割り込み要求制御レジスタが1グループを形成し、シリアル・インタフェース割り込み要求として優先順位指定を行うことができます。グループ内では、優先順位が次のように決まっています。

$$SEFn > SRFn > STFn$$

受信エラーが生じると、受信エラー割り込み要求が優先的に受け付けられます。これは、受信完了割り込みより優先順位が高いためです。その割り込み処理を終えると、受信完了割り込みを受け付けます。

図 11-8 SEICn, SRICn, STICn (n=0, 1)

	7	6	5	4	3	2	1	0
SEIC0/SEIC1	SEFn	SEMKn	MS/INT	ENCS	0	PR2	PR1	PRO
SRIC0/SRIC1	SRFn	SRMKn	MS/INT	ENCS	0	1	1	1
STIC0/STIC1	STFn	STMKn	MS/INT	ENCS	0	1	1	1

注意 $SRICn$ と $STICn$ のビット2-0は，“1”に固定されています。

$SRICn$ と $STICn$ の割り込み要求の優先順位は、 $SEICn$ のPR2-0の設定に従います。

SEF, $SRFn$, $STFn$ ビットは割り込み要求フラグで、それぞれ受信エラーの発生、受信完了、送信完了によってセット(1)され、割り込み要求の受け付け、またはソフトウェアによってリセット(0)されません。

その他のビット・フィールドについては、4.8 割り込み要求制御レジスタを参照してください。

8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み／読み出しができます。この場合、1ウエイトが挿入されます。

$SEICn$, $SRICn$, $STICn$ の内容は、RESET入力により、内容は47Hに初期化されます。

11.9.2 マクロ・サービス制御レジスタ ($SRMSn$, $STMSn$: $n=0, 1$)

$SRMSn$ は、シリアル・インタフェースの受信完了に伴うマクロ・サービス処理モードとチャンネルを指定する8ビット・レジスタです。 $STMSn$ は、シリアル・インタフェースの送信完了に伴うマクロ・サービス処理モードとチャンネルを指定する8ビット・レジスタです。 $SRMSn$, $STMSn$ は、シリアル・インタフェース2チャンネルにそれぞれ対応しています。

8ビット操作または1ビット操作のメモリ・アクセスによって、これらのレジスタの書き込み／読み出しができます。この場合、1ウエイトが挿入されます。

マクロ・サービス制御レジスタの各ビットについては、4.5.4 マクロ・サービス制御レジスタを参照してください。

図 11-9 $SRMSn$, $STMSn$ ($n=0, 1$)

7	6	5	4	3	2	1	0
MSM2	MSM1	MSM0	DIR	0	CH2	CH1	CH0

第12章 スタンバイ機能

μ PD70325, 70335は、低消費電力化を図るスタンバイ機能として、動作クロックを制御する2つのモードがあります。

- ・HALTモード……CPUの動作クロックだけ停止させるモードです。ただし、CPUの各種ステータスとデータ、内部RAMの内容はすべて保持されます。周辺ハードウェアは動作を継続します。
 - ・STOPモード……発振器を停止させ、内部回路全体を停止させるモードです。
- 超低消費電力で、しかも内部RAMの内容とポート上の出力データを保持します。

各モードの設定は、HALT命令、STOP命令によって行います。

12.1 スタンバイ・コントロール・レジスタ (STBC)

STBCレジスタは、スタンバイ・フラグ (SBF) を含む8ビット・レジスタです。上位7ビットは“0”に固定されています。

SBFフラグは、STOP状態からの復帰判定用に使用します。

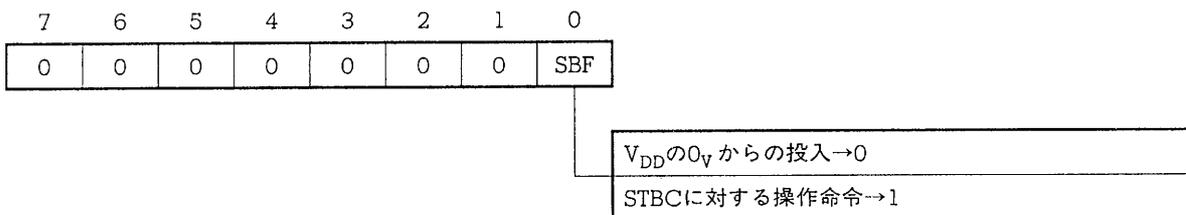
SBFフラグは命令でのみセット(1)されます。また、SBFフラグは、電源電圧(V_{DD})が0Vから立ち上がったときのみクリア(0)され、命令によってはクリア(0)されません。

そのため、SBFフラグをテストすることにより、電源投入時のリセット・スタート(SBF=0)かSTOPモードからの復帰(SBF=1^注)かを判別できます。

STBCレジスタは、 $\overline{\text{RESET}}$ 信号入力によっては初期化されません。

注 SBFフラグは、STOPモードに入る前にあらかじめセット(1)しておいてください。

図 12-1 STBC



12.2 HALTモード

CPUの動作クロックを停止させるモードです。

CPUの空き時間にHALTモードに設定することにより、システム全体の消費電力を低減させることができます。HALT命令を実行することにより、HALT状態となります。

HALTモードでは、CPUクロックが停止しプログラムの実行は停止されますが、その直前のすべてのレジスタ、内蔵RAMの内容は保持されます。各ハードウェアの状態は表 12-2 のようになります。

12.2.1 HALTモードの解除

HALTモードは、ノンマスクابل割り込み (NMI) 要求、マスクされていないマスクابل割り込み要求および $\overline{\text{RESET}}$ 入力によって解除されます (図 12-2)。

なお、HALT命令の実行によりHALT状態に移った直後、マスクされていない割り込み要求やマクロ・サービス要求あるいはDMA要求があった場合、HALT命令の直後にある1命令を実行します。したがって、HALT命令の直後にはNOP命令を1つ以上置いてください。

さらにマクロ・サービス要求、またはDMA処理要求によって、HALTモードからマクロ・サービス、DMAの処理に入ります (図 12-3)。マクロ・サービス、DMAの処理を終了すると、再びHALTモードに戻ります。ただし、マクロ・サービス、DMAの処理中、表 12-1 の条件になるとHALTモードを解除します。

(1) 割り込み要求による解除

(a) 割り込み処理ルーチンで、HALTモードにセットしたとき

処理中の割り込みよりも優先順位の高いマスクされていないマスクابل割り込み要求、またはノンマスクابل割り込み要求の発生により解除されます。

(b) (a) 以外の場合

優先順位に関係なくノンマスクابل割り込み要求または、マスクされていないマスクابل割り込み要求の発生によって解除されます。

(2) $\overline{\text{RESET}}$ 入力による解除

通常のリセット動作と同じです。

図 12-2 割り込み要求によるHALTモードの解除

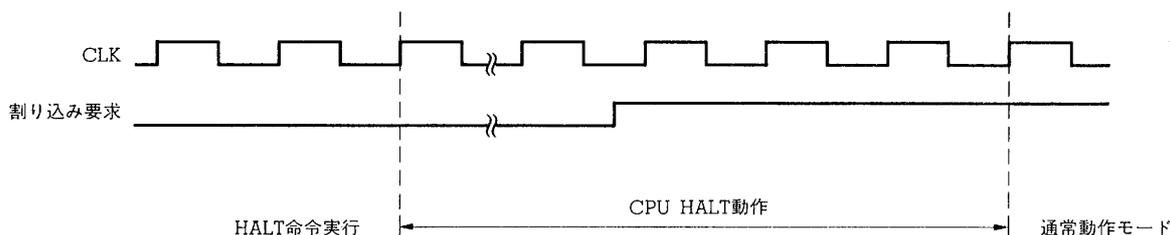


図 12-3 HALTモード中のマクロ・サービス/DMAの起動

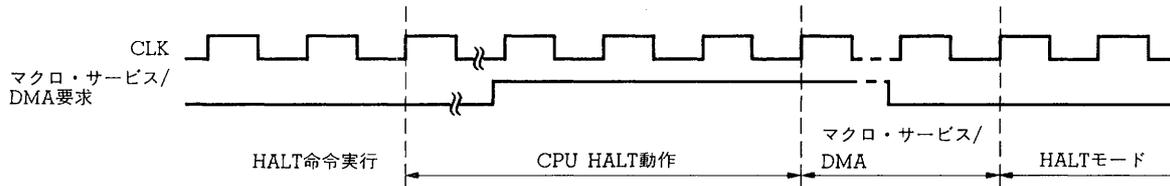


表 12-1 割り込み要求によるHALTモード解除後の動作

解除ソース	E I 状態	D I 状態
ノンマスクابل 割り込み要求	解除後、ベクタ・アドレスに分岐します。	解除後、ベクタ・アドレスに分岐します。
マスクابل割り込み要求	解除後、ベクタ・アドレスに分岐します。	解除後、次の命令を実行します。
マクロ・サービス要求	マクロ・サービスが起動し、マクロ・サービス・カウンタがOHになるか、またはキャラクタ・サーチ・モードで転送データが一致した場合、ベクタ・アドレスに分岐します。 マクロ・サービス・カウンタがOHとならない場合、またはキャラクタ・サーチ・モードで転送データが一致しない場合、再びHALT状態になります。	マクロ・サービスが起動し、マクロ・サービス・カウンタがOHになるか、またはキャラクタ・サーチ・モードで転送データが一致した場合、HALTモードを解除し、次の命令を実行します。
DMA要求	DMAが起動し、ターミナル・カウンタがFFFFHになるとベクタ・アドレスに分岐します。 ターミナル・カウンタがFFFFHとならない場合、再びHALT状態になります。	DMAが起動し、ターミナル・カウンタがFFFFHになるとHALTモードを解除し、次の命令を実行します。

12.3 STOPモード

発振器を停止させるモードです。

応用システム全体の停止時に有効で、超低消費電力を実現します。STOP命令を実行することにより、STOP状態となります。STOPモードでは、すべてのクロックが停止します。プログラムの実行は停止されますが、その直前のすべてのレジスタ、内蔵RAMの内容は保持されます。各ハードウェアの状態は表 12-2 のようになります。

注意 STOPモードでは、X1、X2端子は固定レベルになります。したがって、外部クロックを使用する場合、STOPモードを使用しないでください。STOPモードを使用する場合、クリスタル振動子またはセラミック発振子を使用してください。

12.3.1 STOPモードの解除

STOPモードは、NMI要求および、 $\overline{\text{RESET}}$ 入力により解除されます。

(1) NMI要求による解除 (図 12-4)

NMI端子に有効エッジが入力されると、発振器が発振を再開します。また、タイム・ベース・カウンタ (TBC) も動作しはじめます。STOPモードを解除したのち、ただちにはクロックの供給を行わず、TBCによる発信安定時間の計数後、クロック供給を開始します。TBCによる発信安定時間の計数は、プロセッサ・コントロール・レジスタ (PRC) のビット 2, 3 (TBO, 1) で指定されたインターバル時間の1/2になります (ただし、このときTBCから発生する割り込み要求は禁止されません)。

このインターバル時間の1/2は30 ms以上になるように設定してください。

(2) $\overline{\text{RESET}}$ 入力による解除

通常のリセット動作と同じです。

図 12-4 NMI端子入力によるSTOPモードの解除

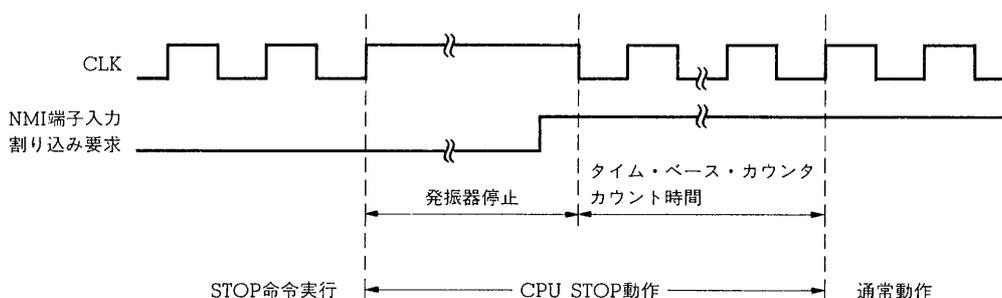


表 12-2 HALTモード/STOPモード

項 目		HALTモード	STOPモード
発振器		動作	停止
内部システム・クロック		停止	
16ビット・タイマ		動作	
タイム・ベース・カウンタ			
HOLD回路			
シリアル・インタフェース			
割り込み要求コントローラ			
DMAコントローラ			
I/Oライン		保持	保持
バス・ライン	A0-A19	保持	保持
	D0-D15	ハイ・インピーダンス	ハイ・インピーダンス
R/ \overline{W} 出力		ハイ・レベル	ハイ・レベル
リフレッシュ動作		動作/停止	停止
データ保持		CPUのステータス, RAMの内容等, 内部データはすべて保持。	CPUのステータス, RAMの内容等, 内部データはすべて保持。
解除方法		<ul style="list-style-type: none"> ・ノンマスクابل割り込み要求 ・マスクابل割り込み要求 ・$\overline{\text{RESET}}$入力 ・マクロ・サービス要求^注 ・DMA^注 	<ul style="list-style-type: none"> ・ノンマスクابل割り込み要求 ・$\overline{\text{RESET}}$入力

注 マクロ・サービス, DMAの処理後はHALTモードに再び戻る。

(メ モ)

第13章 リセット機能

$\overline{\text{RESET}}$ 入力端子にロウ・レベルが入力されると、システム・リセットがかかり各ハードウェアは表 13-1 に示すような状態になります。 $\overline{\text{RESET}}$ 入力がハイ・レベルになるとリセット状態が解除され、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。

表 13-1 各ハードウェアのリセット後の状態 (1/2)

ハードウェア (略号)		アドレス ^注 (下位12ビット: ××□□□H)	リセット後の状態
プログラム・カウンタ		PC	0000H
プログラム・ステータス・ワード		PSW	F002H
内蔵RAM	データ・メモリ		不 定
	汎用レジスタ	AW, CW, DW, BW, SP, BP, IX, IY	EFEH-EFOH
	セグメント・レジスタ (バンク 7 のみ)	DS1, SS, DS0 PS	EEEH, EEAH, EE8H EECH
ポート	ポート・レジスタ	P0, P1, P2	F00H, F08H, F10H
		PT	F38H
	ポート・モード・レジスタ	PM0, PM1, PM2	F01H, F09H, F11H
		PMT	F3BH
ポート・モード・コントロール・レジスタ	PMC0, PMC1, PMC2	F02H, FOAH, F12H	
タイマ・ユニット	タイマ・レジスタ	TM0, TM1	F80H, F88H
	モジュロ/タイマ・レジスタ	MD0, MD1	F82H, F8AH
	タイマ・コントロール・レジスタ	TMC0, TMC1	F90H, F91H
	割り込み要求制御レジスタ	TMIC0-TMIC2	F9CH-F9EH
	マクロ・サービス制御レジスタ	TMMS0-TMMS2	F94H-F96H
DMA コントローラ	DMAモード・レジスタ	DMAM0, DMAM1	FA1H, FA3H
	DMAコントロール・レジスタ	DMAC0, DMAC1	FA0H, FA2H
	割り込み要求制御レジスタ	DIC0, DIC1	FACH, FADH
ソース・アドレス・ポインタ		SAR0L, SAR0M, SAR0H, SAR1L, SAR1M, SAR1H,	FC0H, FC1H, FC2H, FD0H, FD1H, FD2H,
デスティネーション・アドレス・ポインタ		DAR0L, DAR0M, DAR0H, DAR1L, DAR1M, DAR1H,	FC4H, FC5H, FC6H, FD4H, FD5H, FD6H,
ターミナル・カウンタ		TC0L, TC0H, TC1L, TC1H,	FC8H, FC9H, FD8H, FD9H,

注 アドレス上位 8 ビットの××はIDBレジスタで指定される値

表 13-1 各ハードウェアのリセット後の状態 (2/2)

ハードウェア (略号)		アドレス ^{注1} (下位12ビット: ××□□□H)	リセット後の状態	
シリアル・インタフェース	シリアル・モード・レジスタ	SCM0, SCM1	F68H, F78H	00H
	シリアル制御レジスタ	SCC0, SCC1	F69H, F79H	00H
	ボー・レート・ジェネレータ・レジスタ	BRG0, BRG1	F6AH, F7AH	00H
	受信バッファ・レジスタ	RxB0, RxB1	F60H, F70H	不定
	送信バッファ・レジスタ	TxB0, TxB1	F62H, F72H	不定
	シリアル・ステータス・レジスタ	SCS0, SCS1	F6BH, F7BH	60H
	割り込み要求制御レジスタ	(エラー) SEIC0, SEIC1	F6CH, F7CH	47H
		(受信) SRIC0, SRIC1	F6DH, F7DH	
		(送信) STIC0, STIC1	F6EH, F7EH	
	マクロ・サービス制御レジスタ	(受信) SRMS0, SRMS1	F65H, F75H	不定
(送信) STMS0, STMS1		F66H, F76H		
タイマ・ベース割り込み要求制御レジスタ		TBIC	FECH	47H
ユーザ・フラグ・レジスタ		FLAG	FEAH	00H
内部データ領域ベース・レジスタ		IDB	FFFH	FFH
プロセッサ・コントロール・レジスタ		PRC	FEBH	4EH
ウェイト・コントロール・レジスタ		WTC	FE8H	FFFFH
リフレッシュ・モード・レジスタ		RFM	FE1H	FCH
スタンバイ・コントロール・レジスタ		STBC	FE0H	保持 ^{注2}
外部割り込み	外部割り込みモード・レジスタ	INTM	F40H	00H
	割り込み要求制御レジスタ	EXIC0-EXIC2	F4CH-F4EH	47H
	マクロ・サービス制御レジスタ	EMS0-EMS2	F44H-F46H	不定
割り込み要因レジスタ		IRQS	FEFH	不定
割り込みプライオリティ・レジスタ		ISPR	FFCH	00H

注1. アドレス上位8ビットの××はIDBレジスタで指定される値

2. パワーオン・リセット時: 00H

第14章 アドレス生成

14.1 命令アドレス

命令が実行されるごとに自動的にインクリメントされる以外に、命令の実行順を制御するいくつかの方法があります。それらを以下に示します。

14.1.1 ダイレクト・アドレッシング

命令バイト中の2バイトまたは4バイト・イミューディエト・データが直接PCまたはPSとPCの両方にロードされ、ブランチ・アドレスとなります。

次の命令を実行する際に用いられます。

```
CALL far-proc
CALL memptr16
CALL memptr32
BR far-label
BR memptr16
BR memptr32
```

14.1.2 レラティブ・アドレッシング

命令バイト中の1バイトまたは2バイト・イミューディエト・データが符号付きディスプレースメント値となってPCに加算され、ブランチ・アドレスとなります。

8ビット・ディスプレースメントのときは、符号拡張されて16ビット・データとなってPCに加算されます。

加算されるときPCの内容は、次の命令の先頭アドレスを示しています。

次の命令を実行する際に用いられます。

```
CALL near-proc
BR near-label
BR short-label
条件付きブランチ命令 short-label
```

14.1.3 レジスタ・アドレッシング

命令バイト中のレジスタ指定フィールド（3ビット）で指定される任意の16ビット・レジスタの内容が、ブランチ・アドレスとしてPCにロードされます。

データの場合と異なり、8個のすべての16ビット・レジスタ（AW, BW, CW, DW, IX, IY, SP, BP）を使用することができます。

次の命令を実行する際に用いられます。

記述例

CALL regptr16	CALL AW
BR regptr16	BR BW

14.1.4 レジスタ・インダイレクト・アドレッシング

命令バイト中のレジスタ指定フィールドで指定される16ビット・レジスタ（IX, IY, BW）でアドレスされるメモリの内容（ワードまたはダブルワード）がブランチ・アドレスとしてPC（またはPCとPSの両方）にロードされます。

記述例

CALL memptr16	CALL WORD PTR {IX}
CALL memptr32	CALL DWORD PTR {IY}
BR memptr16	BR WORD PTR {BW}
BR memptr32	BR DWORD PTR {IX}

備考 WORD PTRと指定されたものはmemptr16の命令コードが、DWORD PTRと指定されたものはmemptr32の命令コードが、アセンブラによって生成されます。

14.1.5 インデクスト・アドレッシング

命令バイト中の1バイトまたは2バイト・イミューディエト・データが符号付きディスプレイメント値となって、インデクス・レジスタとして働く16ビット・レジスタ（IXまたはIY）に加算され、その結果がアドレスするメモリの内容（ワードまたはダブルワード）がブランチ・アドレスとしてPCにロードされます。

次の命令を実行する際に用いられます。

記述例

CALL memptr16	CALL var {IX} {2}
CALL memptr32	CALL var {IY}
BR memptr16	BR var {IY}
BR memptr32	BR var {IX+4}

備考 変数varがワード属性を持つ場合はmemptr16の命令コードが、ダブルワード属性を持つ場合はmemptr32の命令コードがアセンブラによって生成されます。

14.1.6 ベースト・アドレッシング

命令バイト中の1バイトまたは2バイト・イミューディエト・データが符号付きディスプレースメント値となって、ベース・レジスタとして働く16ビット・レジスタ（BPまたはBW）に加算され、その結果がアドレスするメモリの内容（ワードまたはダブルワード）がブランチ・アドレスとしてPCにロードされます。

次の命令を実行する際に用いられます。

		記 述 例
CALL	memptr16	CALL var [BP+2]
CALL	memptr32	CALL var [BP]
BR	memptr16	BR var [BW] [2]
BR	memptr32	BR var [BP]

備考 変数varがワード属性を持つ場合はmemptr16の命令コードが、ダブルワードの属性を持つ場合はmemptr32の命令コードがアセンブラによって生成されます。

14.1.7 ベースト・インデクスト・アドレッシング

命令バイト中の1バイトまたは2バイト・イミューディエト・データが符号付きディスプレースメント値となり、この値とベース・レジスタとして働く16ビット・レジスタ（BPまたはBW）およびインデクス・レジスタとして働く16ビット・レジスタ（IXまたはIY）の3つが加算され、その結果がアドレスするメモリの内容（ワードまたはダブルワード）が、ブランチ・アドレスとしてPCにロードされます。

次の命令を実行する際に用いられます。

		記 述 例
CALL	memptr16	CALL var [BP] [IX]
CALL	memptr32	CALL var [BW+2] [IY]
BR	memptr16	BR var [BW] [2] [IX]
BR	memptr32	BR var [BP+4] [IY]

備考 変数varがワードの属性を持つ場合はmemptr16の命令コードが、ダブルワードの属性を持つ場合はmemptr32の命令コードがアセンブラによって生成されます。

14.2 メモリ・オペランド・アドレス

命令を実行する際に操作対象となるレジスタやメモリなどをアドレスする方法として、次に示すいくつかの方法があります。

14.2.1 レジスタ・アドレッシング

命令バイト中のレジスタ指定フィールド (reg= 3 ビット・フィールド, sreg= 2 ビット・フィールド) の内容が操作対象となるレジスタをアドレスします。

regの場合は、同じく命令バイト中のワードかバイトかを指定する 1 ビット (W) と組みになって 8 種のワード・レジスタ (AW, BW, CW, DW, BP, SP, IX, IY) と 8 種のバイト・レジスタ (AL, AH, BL, BH, CL, CH, DL, DH) を指定します。

sregの場合は、4 種のセグメント・レジスタ (PS, SS, DS0, DS1) を指定します。

また、命令のオペレーション・コードが特定のレジスタを指定する場合があります。

次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
reg	AW, BW, CW, DW, SP, BP, IX, IY, AL, AH, BL, BH, CL, CH, DL, DH
reg16	AW, BW, CW, DW, SP, BP, IX, IY
reg8	AL, AH, BL, BH, CL, CH, DL, DH
sreg	PS, SS, DS0, DS1
acc	AW, AL

記述例

MOV	reg, reg'の場合
MOV	BP, SP
MOV	AL, CL

14.2.2 イミディエト・アドレッシング

命令バイト中の 1 バイトまたは 2 バイト・イミディエト・データがそのまま操作対象となります。

次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
imm	8/16ビット・イミーディエト・データ
imm16	16ビット //
imm8	8ビット //
pop-value	16ビット //

immの場合、8ビットか16ビットかの区別は、オペランドに記述されたimmの値または同時に記述される別のオペランドの属性をアセンブラが判断して決定し、ワード/バイト指定ビットWを決定します。

記述例

```
MOV reg, immの場合
MOV AL, 5; バイト
```

```
MUL reg16, reg16, imm16の場合
MUL AW, BW, 1000H
```

14.2.3 ダイレクト・アドレッシング

命令バイト中のイミーディエト・データが、操作対象となるメモリをアドレスします。

次のオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
mem	8ビットまたは16ビット・メモリ・データを指定する16ビット変数
dmem	//
imm4	ビット・フィールド・データのビット長を示す4ビット変数

記述例

```
MOV mem, immの場合
MOV WORD_VAR, 2000H
```

```
MOV acc, dmemの場合
MOV AL, BYTE_VAR
```

14.2.4 レジスタ・インダイレクト・アドレッシング

命令バイト中のメモリ指定フィールド (mod, mem) によって指定される16ビット・レジスタ (IX, IY, BW) が、操作対象となるメモリをアドレスします。

次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
mem	{IX}, {IY}, {BW}

記述例

SUB mem, regの場合

SUB {IX}, AW

14.2.5 オートインクリメント / デクリメント・アドレッシング

レジスタ・インダイレクト・アドレッシングに属するものですが、デフォルト・レジスタの内容で操作対象をアドレスした後、そのレジスタの内容を自動的にインクリメント / デクリメント（バイト処理なら+1/-1、ワード処理なら+2/-2）します。

つまり、このアドレッシング機能を用いれば、次のバイト / ワード・オペランド処理のためのアドレス更新が自動的に行われるわけです。

インクリメントとデクリメントの区別は、方向フラグ(DIR)によって行われ、DIR=0ならインクリメント、1ならデクリメントです。

このアドレッシングは、すべてデフォルト・レジスタに対して行われ、次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	デフォルト・レジスタ
dst-block	IY
src-block	IX

このアドレッシングとバイト / ワード・オペランドのくり返し処理回数をカウントするカウンタ (CW) が組み合わされて、ブロック・データ処理の制御に用いられます。

14.2.6 インデクスト・アドレッシング

命令バイト中の1バイトまたは2バイト・イミーディエト・データが符号付きディスプレースメント値となって、インデクス・レジスタとして働く16ビット・レジスタ (IXまたはIY) に加算され、その結果が操作対象となるメモリ・オペランドをアドレスします。

このアドレッシングは、アレイ・タイプのデータをアクセスするのに有効で、ディスプレースメントがアレイの開始アドレスを指し、インデクス・レジスタの内容がそこから何番目のアレイかを決定します。

次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
mem	var {IX}, var {IY}
mem16	//
mem8	//

記述例

```

TEST mem, immの場合
TEST BYTE_VAR (IX), 7FH
TEST BYTE_VAR (IX+8), 7FH
TEST WORD_VAR (IX) (8), 7FFFH

```

備考 変数varがバイト属性を持つ場合はバイト・オペランドが指定され、ワード属性を持つ場合はワード・オペランドが指定され、それぞれに該当する命令コードがアセンブラによって生成されます。

14.2.7 ベースト・アドレッシング

命令バイト中の1バイトまたは2バイト・イミディエト・データが符号付きディスプレースメント値となって、ベース・レジスタとして働く16ビット・レジスタ (BPまたはBW) に加算され、その結果が操作対象となるメモリ・オペランドをアドレスします。

このアドレッシングは、メモリの複数箇所に置かれる構造タイプのデータをアクセスするのに有効で、ベース・レジスタが各構造の開始アドレスを指し、ディスプレースメントが各構造内の1要素を選択します。

次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
mem	var (BP), var (BW)
mem16	//
mem8	//

記述例

```

SHL mem, 1の場合
SHL BYTE_VAR (BP), 1
SHL WORD_VAR (BP+2), 1
SHL BYTE_VAR (BP) (4), 1

```

備考 変数varがバイト属性を持つ場合はバイト・オペランドが指定され、ワード属性を持つ場合はワード・オペランドが指定され、それぞれに該当する命令コードがアセンブラによって生成されます。

14.2.8 ベース・インデクスト・アドレッシング

命令バイト中の1バイトまたは2バイト・イミディエト・データが符号付きディスプレースメント値となり、この値とベース・レジスタとして働く16ビット・レジスタ (BPまたはBW) およびインデクス・レジスタとして働く16ビット・レジスタ (IXまたはIY) が加算され、その結果が操作対象となるメモリ・オペランドをアドレスします。

このアドレッシングは、ベース・レジスタの内容とインデクス・レジスタの内容を両方変化させて1つのデータを指すことができるため、アレイ・タイプを含んだ構造タイプのデータをアクセスするのに非常に有効です。すなわち、ベース・レジスタで各構造の先頭アドレスを指し、ディスプレースメント値がそこからアレイ・データの先頭アドレスまでのオフセット分を示し、アレイ・データの何番目をインデクス・レジスタが指すというようなことができます。

次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
mem	var {ベース・レジスタ} {インデクス・レジスタ}
mem16	//
mem8	//

記述例

```
PUSH    mem16の場合
PUSH    WORD_VAR {BP} {IX}
PUSH    WORD_VAR {BP+2} {IX+6}
PUSH    WORD_VAR {BP} {4} {IX} {8}
```

14.2.9 ビット・アドレッシング

命令バイト中の3/4ビット・イミディエト・データまたはCLレジスタの下位3/4ビットが操作対象の8/16ビット・レジスタまたはメモリの1つのビットを指定します。

このアドレッシングを用いた命令によれば、レジスタ/メモリの特定の1ビットだけを、ほかのビット内容を意識せずにテスト(0, 1の判定)/セット/クリア/反転することができます。つまり、AND命令やOR命令を使用してセット、リセットするときのように、1ビットを操作するのにバイト/ワード・データを用意しなければならないということは起こりません。

次に示すオペランド記述形式を持つ命令を実行する際に用いられます。

記述形式	記述方法
imm4	ワード・オペランドのビット番号
imm3	バイト・オペランドのビット番号
CL	CL

記述例

```
TEST1 reg8, CL
TEST1 AL, CL
NOT1 reg8, imm3
NOT1 CL, 5
CLR1 mem16, CL
CLR1 WORD_VAR (IX), CL
SET1 mem16, imm4
SET1 WORD_VAR (BP), 9
```

14.2.10 特殊機能レジスタ・アドレッシング

命令バイト中の1バイト・イミディエト・データが特殊機能レジスタ領域の先頭からのオフセット(符号なし)となって、操作対象となる特殊機能レジスタをアドレスします。

このアドレッシング・モードはBTCLR命令にのみ適用されます。

記述形式	記述方法
sfr	8ビット特殊機能レジスタを指定する8ビット変数

記述例

```
BTCLR EXIC, 7, 45
```

備考 特殊機能レジスタはメモリ空間にマッピングされており、 $\times\times F00H \sim \times\times FFFH$ までの領域にあります($\times\times$ はIDBレジスタの値)。詳細は**3.5.3 特殊機能レジスタ領域の項**参照。

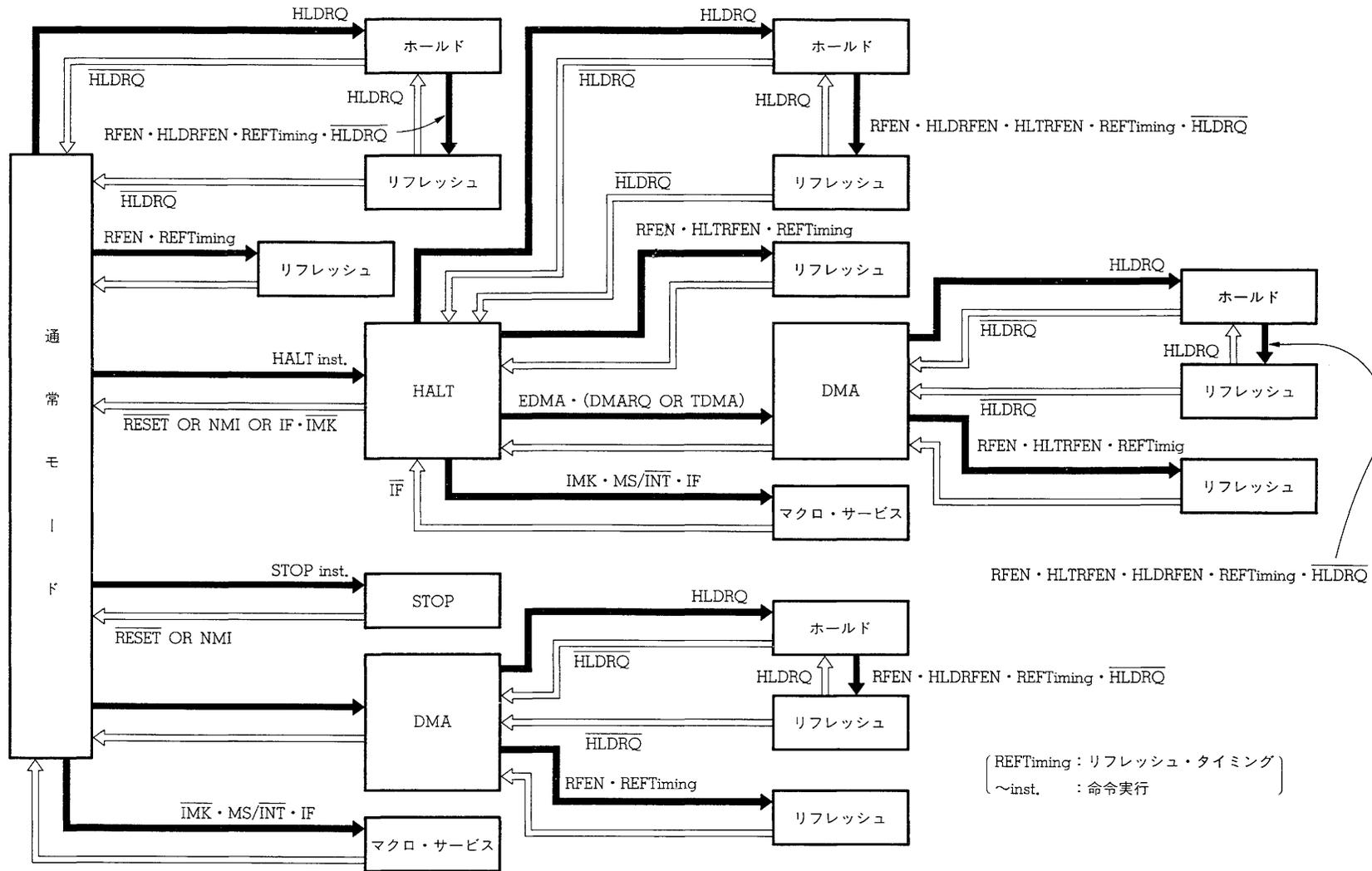
保守 / 廃止

[メ 毛]

第15章 動作状態の遷移

μ PD70325, 70335は、各動作モード(マクロ・サービス, DMA, リフレッシュ, ホールド, HALT, STOP)間を図 15-1 に示す条件と流れで、遷移します。

図 15-1 動作状態遷移図



付録A 制限事項

A.1 マクロ・サービスとINTの競合に関する制限事項

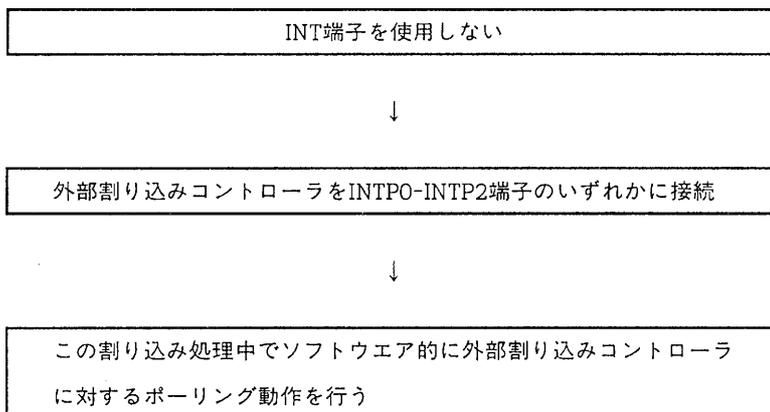
A.1.1 制限事項内容

マクロ・サービス要求が、割り込み優先順位制御により保留されている状態でINT要求があった場合、保留されるべきマクロ・サービスが保留されずに先に実行されることがあります。

A.1.2 回避方法

マクロ・サービス機能を使用するとき、この現象が貴社のシステムの動作に悪影響を及ぼす場合、INT入力を同時に使用しないでください。

外部割り込みコントローラ (μ PD71059) を外付けし、かつ、マクロ・サービス機能を使用するとき、この現象が貴社システムの動作に悪影響を及ぼす場合、次のような手順で割り込み処理を行ってください。

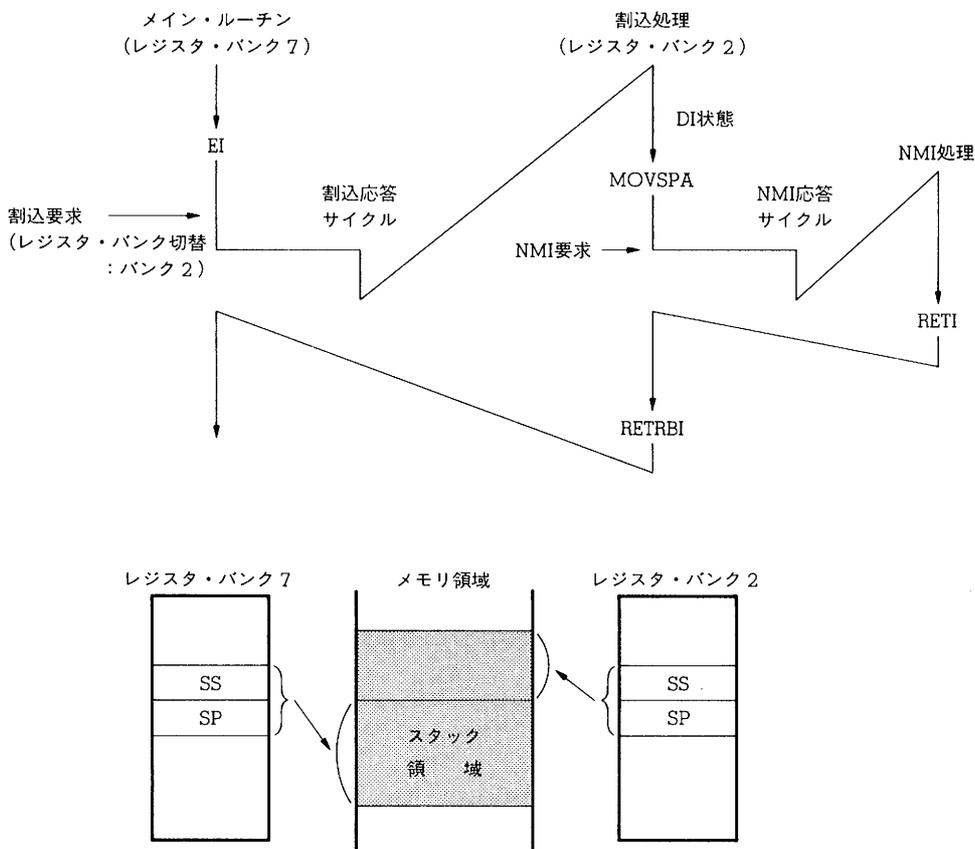


A.2 MOVSPA命令の使用上の注意事項

A.2.1 デバイスの動作説明（現象）

「非同期に発生する割り込みでレジスタ・バンク切り替えを行う場合、切り替え前後のレジスタ・バンクが連続のスタック領域を使用する」ような応用では、切り替え前後のレジスタ・バンク間でSS, SPをコピーするMOVSPA命令を利用することができます。この場合、切り替え後のレジスタ・バンクのプログラム処理の先頭でMOVSPA命令を実行することにより、レジスタ・バンク切り替え前後で使用されるスタック領域を連続させることができます（図A-1参照）。

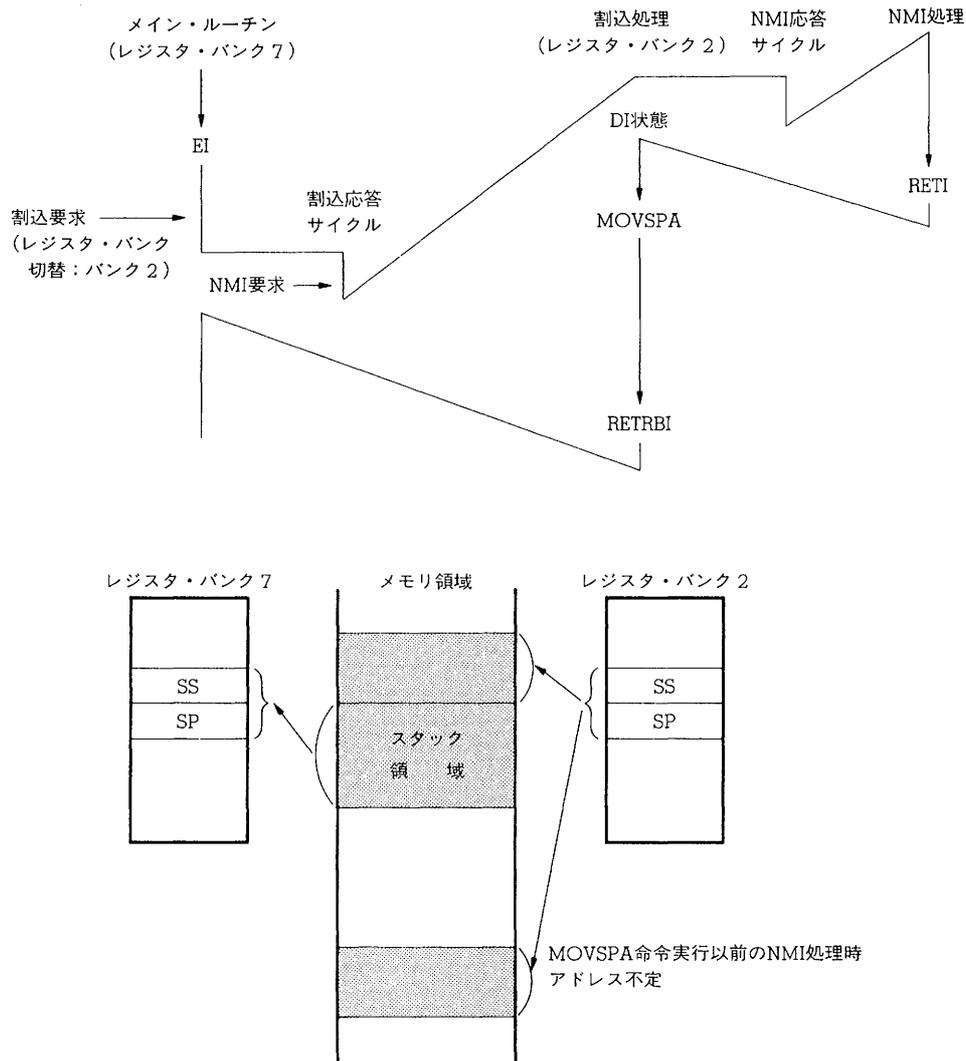
図A-1 正常動作時



ところが、レジスタ・バンク切り替えにより、応答する割り込みINT××が受け付けられた直後にNMI要求があった場合、INT××の処理ルーチンを実行する前にNMIが受け付けられるため、新しく切り替わったレジスタ・バンク中のSS, SPで示されるスタック領域に対し、NMI受け付け時のPS, PC, PSWの退避が行われます。

このため、「MOVSPA命令により、切り替え前後のレジスタ・バンクのスタック領域を連続させる」ような応用では、レジスタ・バンク切り替えにより応答する割り込みが受け付けられた直後にNMI要求があった場合、SS, SPのコピーの前にNMIが受け付けられ、不定なメモリ領域をスタック領域としてアクセス（書き込み）することになります（図A-2参照）。

図A-2 異常動作時



A.2.2 回避方法

レジスタ・バンクの切り替えにより、応答する割り込みとNMIを同時に使用する場合、(1)、(2)のどちらかの方法により上記の現象を回避してください。

(1) MOVSPA命令を使用せず、割り込み発生により切り替わるレジスタ・バンクごとに、個別にスタック領域を設定する。

(2) 以下の(a)、(b)の手順により、上記の現象に対応するための予備用スタックを用意し、スタック領域が不定とならないようにする。

(a) あらかじめ、予備用スタックとして使用するための領域を確保しておき、初期化ルーチンにおいて、割り込み発生により切り替わるレジスタ・バンクのSS, SPをその領域にセットしておく。^注

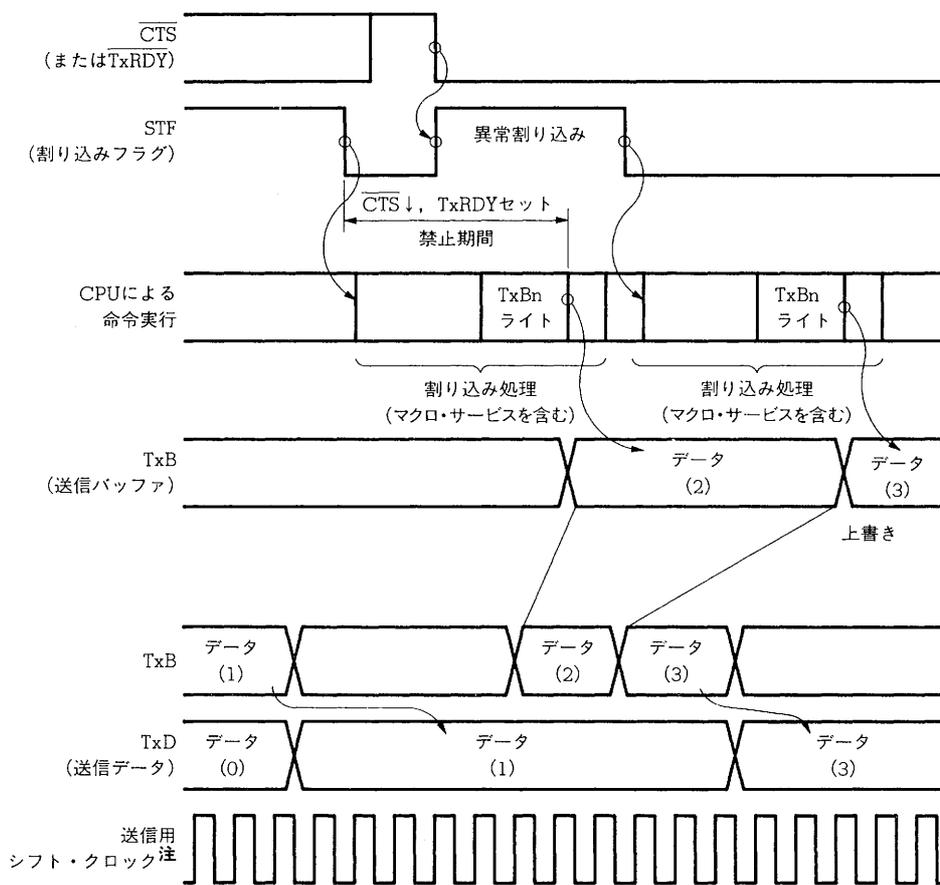
(b) 切り替え後のレジスタ・バンクからもとのレジスタ・バンクへ復帰する直前に、(a)で設定した値をセットしておく。

注 (b)を行わない場合、2回目以降の割り込みの発生時に、再びスタック領域が不定となります。

A.3 シリアル送信中の送信禁止操作による送信データ抜けに関する制限事項

A.3.1 デバイスの動作説明（現象）

内蔵のシリアル・インタフェースにおいて、送信完了割り込み要求を受け付けてから送信バッファTxBへ次の書き込みを行うまでの間に $\overline{\text{CTS}}$ 入力を立ち下げる、あるいは、TxRDYフラグをセットすることにより送信許可状態に変化させたとき、余分な送信完了割り込み要求が発生します（マクロ・サービスを使用する場合も同様です）。その結果、その割り込み処理（またはマクロ・サービス）によるTxBへの上書きによって直前に書き込まれた送信データ（1キャラクタ）が抜けることがあります。



注 11ビット構成/1キャラクタ転送の場合（たとえば、データ長8、ストップ・ビット2、ノー・パリティ）

A.3.2 回避方法

TxBへの送信データの書き込みを行う場合、その直前でシリアル・ステータス・レジスタ（SCS）内の送信バッファ・エンpty・フラグ（TxBE）を読み出し、TxBが空であることを確認してから行ってください。V25, V35は送信バッファ・エンpty・フラグがないため、この回避方法はできません。

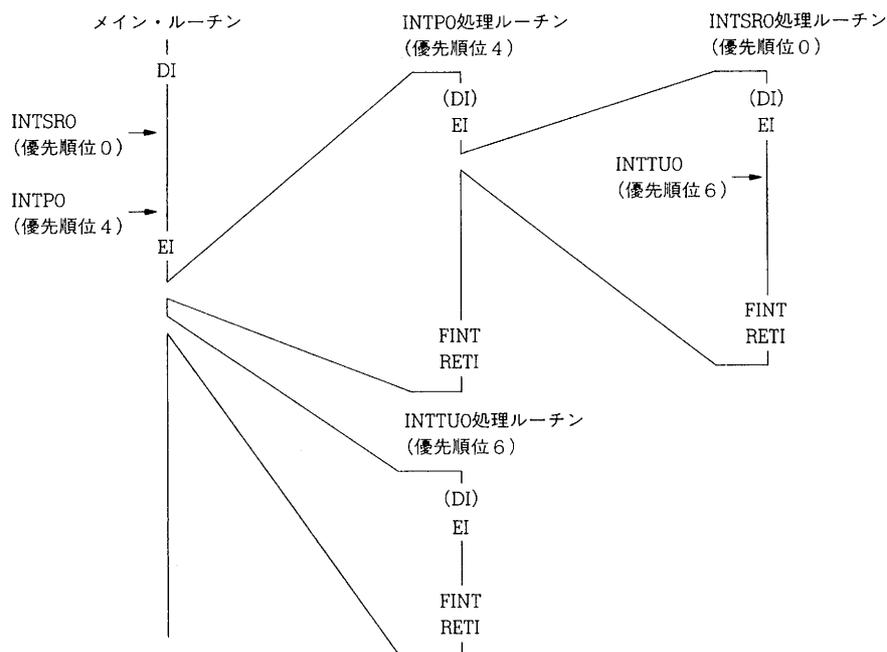
A.4 割り込みの優先順位と処理順序に関する注意事項

CPUがDI状態からEI状態に移したとき、DI状態で保留されていた複数の割り込み要求のうち、ソフトウェアによってプログラマブルに指定される優先順位の低い割り込み要求が優先順位の高い割り込み要求より先に受け付けられる場合があります（発生タイミングは内部タイミングのみに依存するため、外部的に発生タイミングを限定できません）。

しかし、優先順位の低い割り込み処理中でEI命令を実行すると、優先順位に従った多重処理制御により、優先順位の高い割り込みも引き続き受け付けられます。したがって、優先順位の低い割り込み処理については、その先頭でEI命令を実行することにより処理中はEI状態となるようにしてください。

本来、優先順位の低い割り込み処理中はEI状態であるべきであり、優先順位の低い割り込み処理中でDI状態が長く続くと、その間に発生した高い優先順位の割り込み要求はその期間保留されるため、優先順位の設定が意味を持たなくなります。優先順位の低い割り込み処理中ではできるだけ早めにEI命令を実行することが必要です。

多重処理制御による正常処理の例



A.5 タイマのダウン・カウント開始に関する注意事項

A.5.1 デバイスの動作説明（現象）

タイマ・レジスタTMの値が0の状態でのダウン・カウントを開始すると、カウント開始直後に次の現象が起きます。

- タイマ割り込み要求の発生
- TOUT端子のレベルの反転（ENTOビットが“1”のとき）

インターバル・タイマを使用していてこのようなことが起こった場合、TMCレジスタのTSビットをクリア(0)することによって、ダウン・カウント中のタイマ・レジスタTMを停止させることができます。このとき、タイマ停止のタイミングがTM=0となるタイミングと重なると、次にダウン・カウントを開始（TMCレジスタのTSビットをセット(1)）させた直後に、同様の現象が発生してしまいます。

A.5.2 回避方法

タイマ・レジスタTMのダウン・カウントを停止（TSビットのクリア）させたあと、TMレジスタに0000H以外のワード・データを書き込んでください。カウント開始直後に発生する余分なTOUT端子の反転、タイマ割り込み要求はなくなり、正常に動作します。

例

```

MOV    TMC0, 00H
MOV    TMIC0, 00H
MOV    MDO, 0FE0H

MOV    TMC0, 88H    ダウン・カウント開始

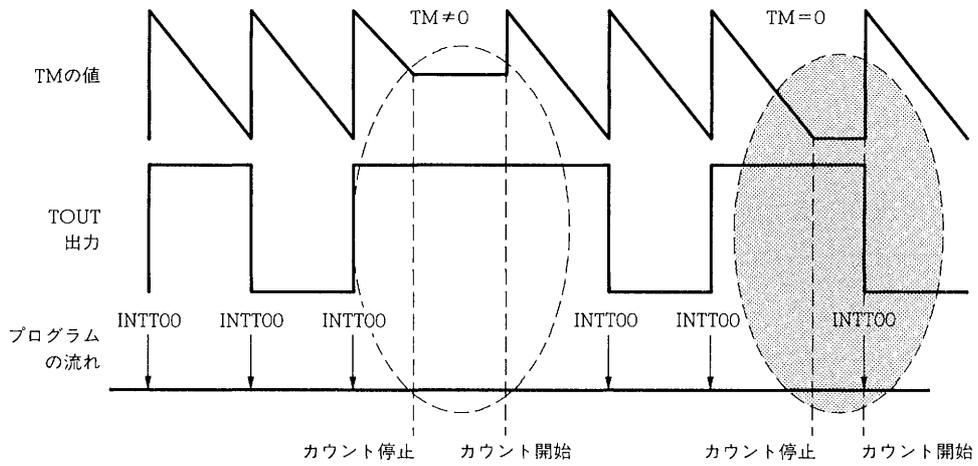
MOV    TMC0, 00H    ダウン・カウント停止

MOV    TM0, 0001H   TMレジスタに0000H以外の値を書く

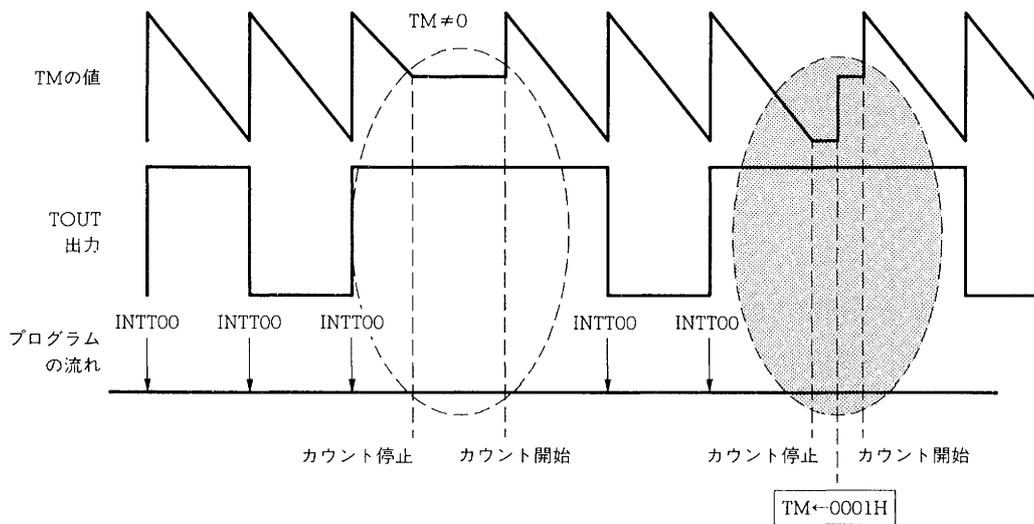
MOV    TMC0, 88H    ダウン・カウント開始

```

回避方法を実施しなかった場合



回避方法を実施した場合



A.6 マクロ・サービスのマスクに関する制限事項

★

A.6.1 デバイスの動作説明（現象）

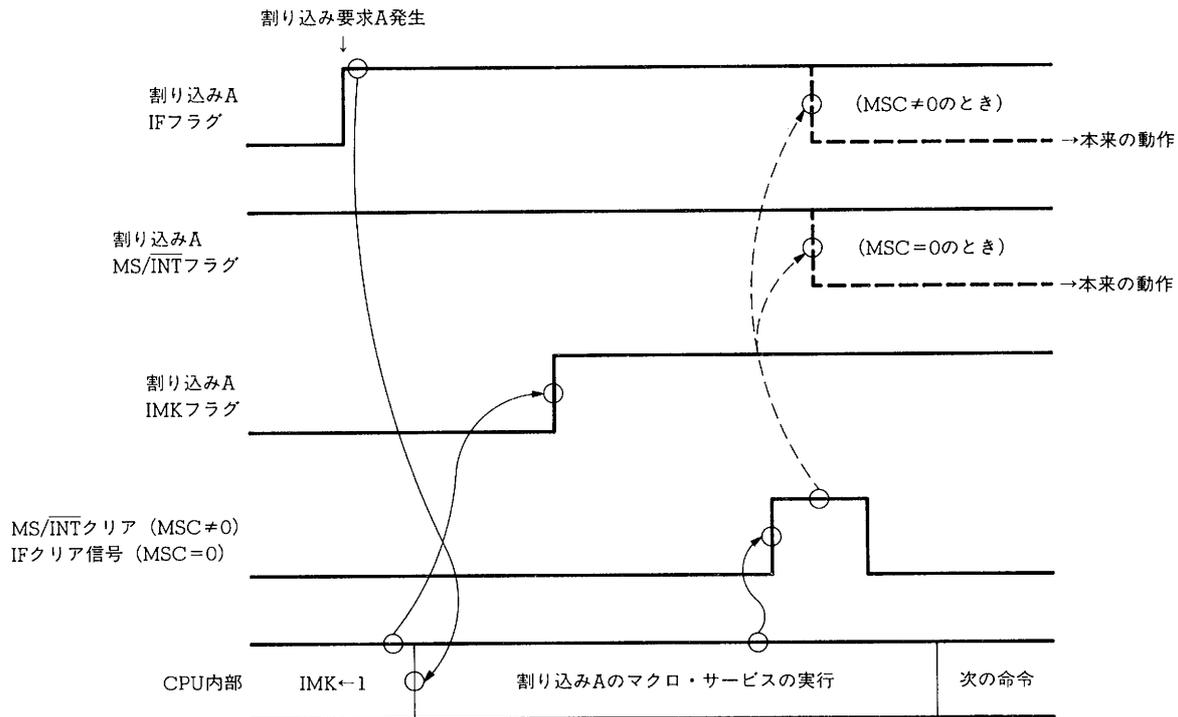
マクロ・サービス応答に設定された割り込みA（マクロ・サービスA）の実行直前に、マクロ・サービスAのIMKフラグがセットされる命令が実行されると、続くマクロ・サービスA実行中にIMKフラグがセットされます。そのときにマクロ・サービスAは正常に実行されずに以下に示す現象が発生します。

● **MSC (マクロ・サービス・カウンタ) ≠ 0 のとき**

マクロ・サービスA実行後にIFフラグがクリアされないために、次の割り込み要求が発生していても、マスク解除したあと再びマクロ・サービスAが実行されます。

● **MSC (マクロ・サービス・カウンタ) = 0 のとき**

マクロ・サービスA実行後にMS/ $\overline{\text{INT}}$ フラグがクリアされないために、マクロ・サービスAの完了割り込みが発生せず、マスク解除したあと再びマクロ・サービスAが実行されます。その結果、MSCは0からOFFHへデクリメントされ、もう一度MSC=00Hとなるまで割り込みが発生することにマクロ・サービスを実行します。



A.6.2 回避方法

マクロ・サービスに設定された割り込みのマスク・フラグをセット / リセットする場合、以下に示す手順で行ってください。

A.7 CVTBD/CVTDB命令に関する制限事項

★

A.7.1 デバイスの動作説明（現象）

表A-1 CVTBD/CVTDB命令に関する制限事項対象命令に示す命令の実行直後にCVTBD命令またはCVTDB命令を実行すると、制限事項対象命令の最後のメモリ・ライト動作時に正常なデータが書き込まれずに、不定なデータが書き込まれます。このとき、メモリへの書き込みデータが変化するだけでフラグは正常に動作します。

ただし、内部RAMアクセス許可 (RAMEN=1) の場合、または制限事項対象命令でアクセスするメモリのウェイト数が μ PD70325のときは2クロック未満、 μ PD70335のときは1クロック未満の場合はこの現象は発生しません。

A.7.2 回避方法

制限事項対象命令の直後にCVTBD/CVTDB命令を実行する場合は、CVTBD/CVTDB命令の直前にNOP命令（1命令以上）を実行してください。

例 OR mem, reg
NOP ←必ずNOPを1つ以上入れてください。
CVTBD

表 A-1 CVTBD/CVTDB命令に関する制限事項対象命令

命令群	ニモニック	オペランド	命令群	ニモニック	オペランド	
データ転送命令	MOV	mem, reg	ビット操作命令	CLR1	mem16, CL	
		mem, imm			mem8, imm3	
		mem16, sreg			mem16, imm4	
	XCH	mem, reg		SET1	mem8, CL	
		reg, mem			mem16, CL	
プリミティブ・ブロック転送命令	MOVBK	dst_block, src_block			mem8, imm3	
	STM	dst_block			mem16, imm4	
入出力命令	OUT	imm8, acc	シフト命令	SHL	mem, 1	
		DW, acc			mem, CL	
プリミティブ入出力命令	OUTM	DW, src_block				
加減算命令	ADD	mem, reg		SHR	mem, 1	
		mem, imm			mem, CL	
	ADDC	mem, reg			mem, imm8	
		mem, imm	SHRA	mem, 1		
	SUB	mem, reg		mem, CL		
		mem, imm	mem, imm8			
BCD演算命令	ROL4	mem8	ローテート命令	ROL	mem, 1	
		mem8			mem, CL	
増減命令	INC	mem		ROR	mem, 1	
		mem			mem, CL	
補数演算命令	NOT	mem				mem, imm8
		mem	ROLC	mem, 1		
論理演算命令	AND	mem, reg			mem, CL	
		mem, imm	mem, imm8			
	OR	mem, reg	RORC	mem, 1		
		mem, imm		mem, CL		
	XOR	mem, reg			mem, imm8	
		mem, imm	スタック操作命令	PUSH	mem16	
ビット操作命令	NOT1	mem8, CL			reg16	
		mem16, CL			sreg	
		mem8, imm3			PSW	
		mem16, imm4			R	
	CLR1	mem8, CL			imm	
			POP		mem16	

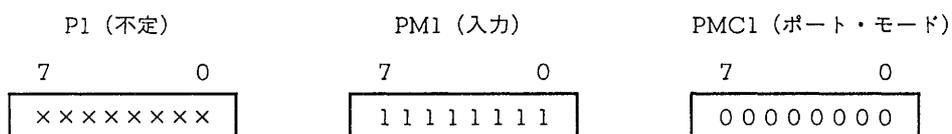
付録B 設定例

★

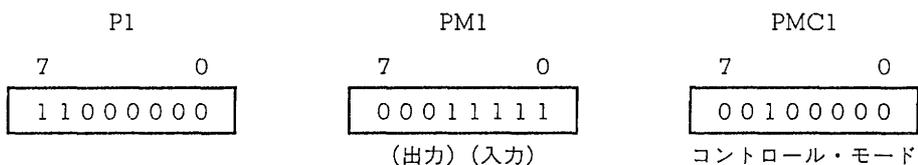
B.1 ポート

パワーオン・リセット後に、ポート1のイニシャライズを行う例を示します。

リセット時



設定後



```

;*****
;***                特殊機能レジスタの設定                ***
;*****
;

```

```

      ⋮
MOV   P1,  11000000B
MOV   PM1, 00011111B
MOV   PMC1,00100000B      ;TOUT出力
      ⋮

```

付

- P10-P14 : 入力ポートより、データの入力で変化します。
- P15 : コントロール・モードにより、TOUT出力します。
- P16, P17 : 出力ポートより、“1”を出力します。

B.2 プログラマブル・ウエイト, プロセッサ・コントロール, リフレッシュ機能

ウエイト挿入, リフレッシュなどのモード設定方法

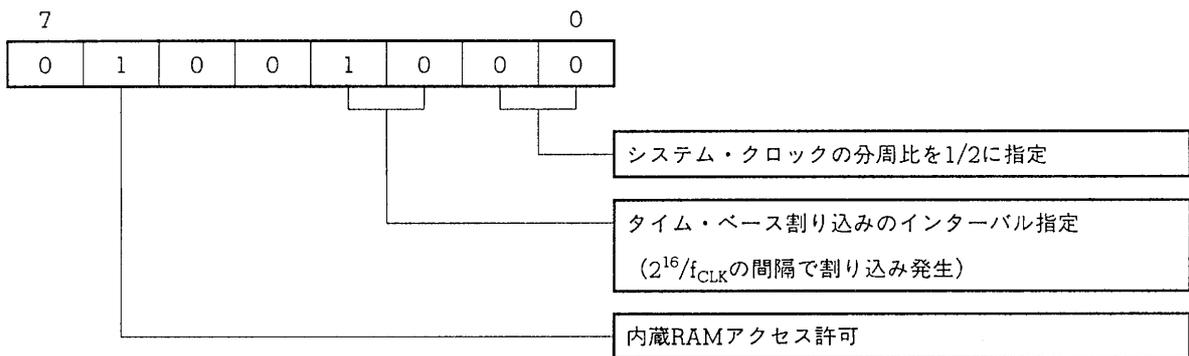
```

;*****
;***                      特殊機能レジスタの初期化                      ***
;*****
;
;
;
MOV     PRC, 01001000B      ;①
MOV     WTCH, 00011010B    ;②
MOV     WTCL, 01010101B   ;②
MOV     RFM, 11110101B    ;③
;
MOV     STBC, 00000001B   ;④
;
HALT

;*****
;
;
MOV     RFM, 00000000B    ;⑤
STOP
;

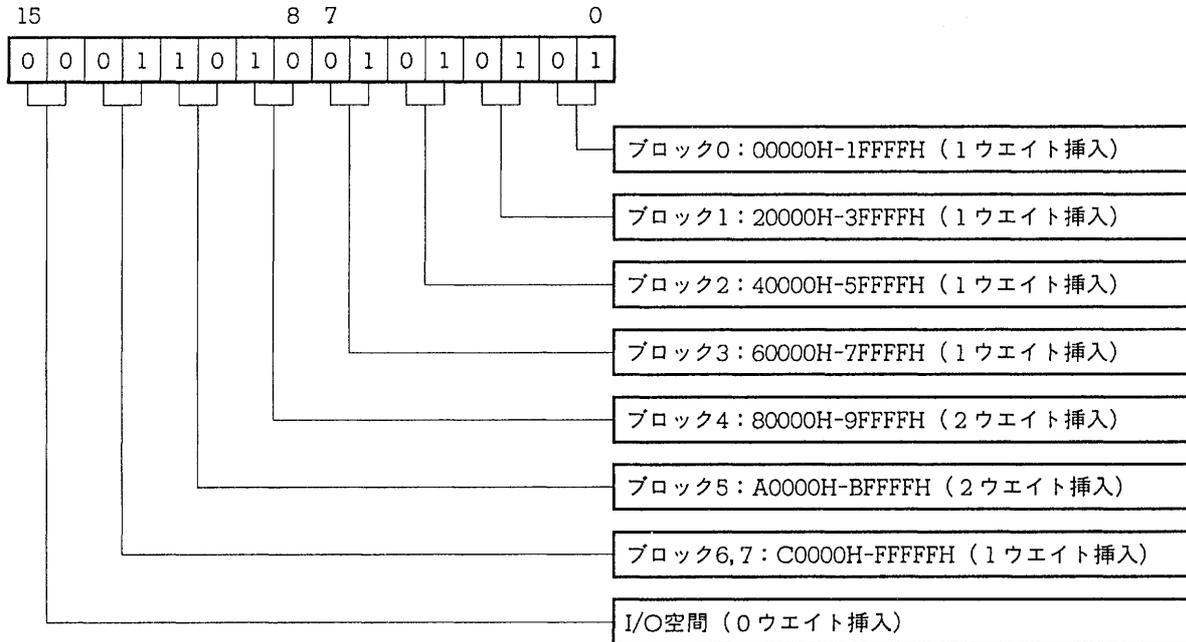
```

① PRC (プロセッサ・コントロール・レジスタ)



② WTC (ウェイト・コントロール・レジスタ)

8ブロックのメモリ空間とI/O空間に対して個別にウェイト挿入



次のように1つにまとめることができます。

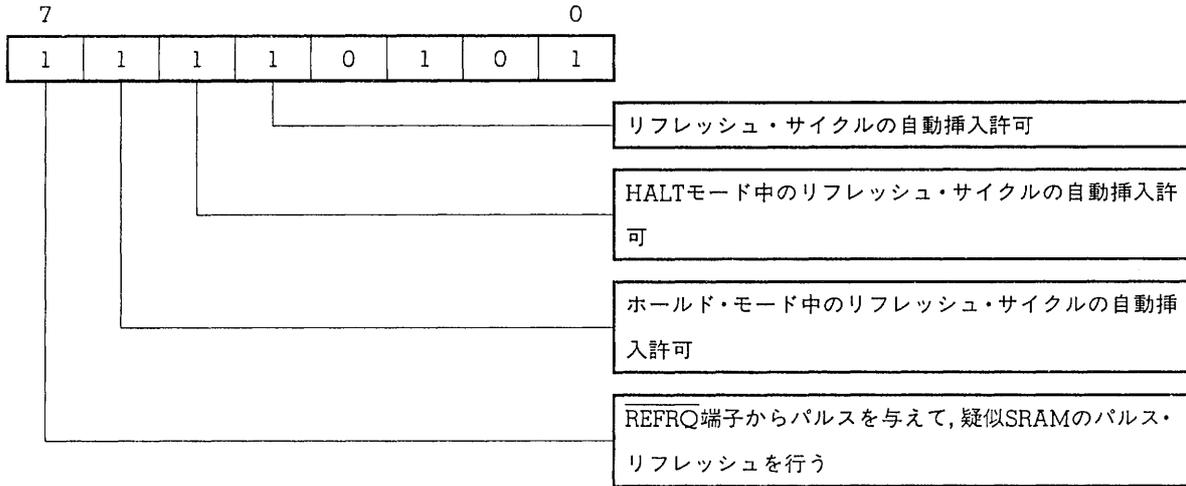
```
MOV WTC, 1A55H
```

メモリとウェイトの関係

メモリの配置	容量	ブロック	ウェイト
00000H-07FFFFH	32 Kバイト	ブロック 0	1 ステート
40000H-47FFFFH	32 Kバイト	ブロック 2	1 ステート
80000H-BFFFFFH	256 Kバイト	ブロック 4, 5	2 ステート
F8000H-FFFFFH	32 Kバイト	ブロック 6, 7	1 ステート

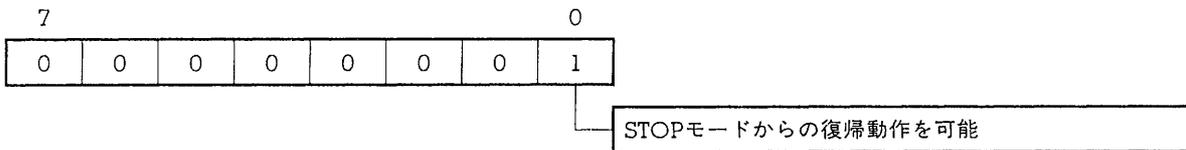
③ RFM (リフレッシュ・モード・レジスタ)

一連のバス・サイクル中にリフレッシュ・サイクルを挿入



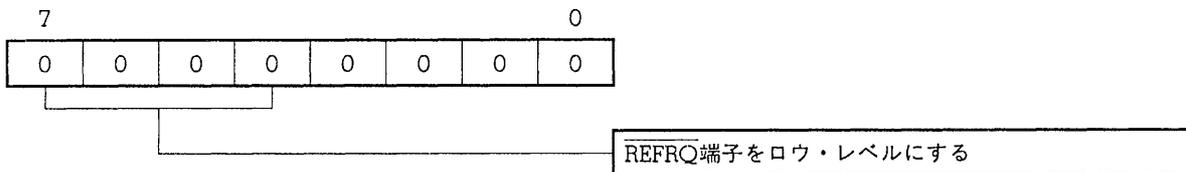
④ STBC (スタンバイ・コントロール・レジスタ)

STOPモードからの復帰動作を制御



⑤ RFM (リフレッシュ・モード・レジスタ)

$\overline{\text{REFRQ}}$ 端子をロウ・レベルにして、疑似SRAMのパワーダウン・セルフ・リフレッシュを使用



B.3 レジスタ・バンクの切り替え

B.3.1 割り込み要求によるレジスタ・バンクの切り替え

INTTU0 (タイマ・ユニット0の割り込み)の応答時にレジスタ・バンク7からレジスタ・バンク6に切り替えます。

```

*****
***          タイマ割り込みの設定 (レジスタ・バンクの切り替え)          ***
*****
;
;
;
MOV          REGBANK.BK6.BVPC, OFFSET INTTU0 注          ;①ベクタPCの初期化
MOV          REGBANK.BK6.BPS,  SEG INTTU0              ;①P S
MOV          REGBANK.BK6.BDS0, 0                       ;①D S 0
MOV          REGBANK.BK6.BDS1, 0                       ;①D S 1
;
MOV          TMICO, 00010110B                          ;②
MOV          MDO,   OFFH                               ;
MOV          TMC0,  80H                               ;③
;
EI
;
;
=====
;===          レジスタ・バンク6の処理          ===
=====
INTTU0:
MOVSPA      ;④
;
FINT        ;⑤レジスタ・バンクからの復帰
RETRBI     ;⑤

```

注 アセンブラ (RA70116-I) の構造体フィールドの記述形式で、内蔵RAM領域のレジスタ・バンク6内のベクタPCの位置を示します。

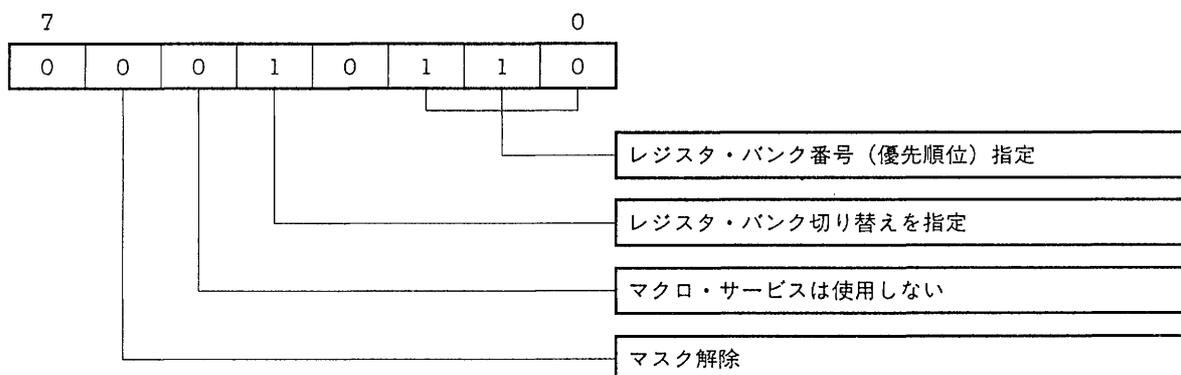
この形式を使用しない場合は、内蔵RAM領域 (レジスタ・バンク0) の先頭アドレス (××E00H) とレジスタ・バンク6のオフセット・アドレス (00C0H)、ベクタPCのオフセット・アドレス (0002H) を加算して、レジスタ・バンク6のベクタPCの位置 (××ECH) を決定してください。

備考 ××：内部データ領域ベース・アドレスの上位8ビット

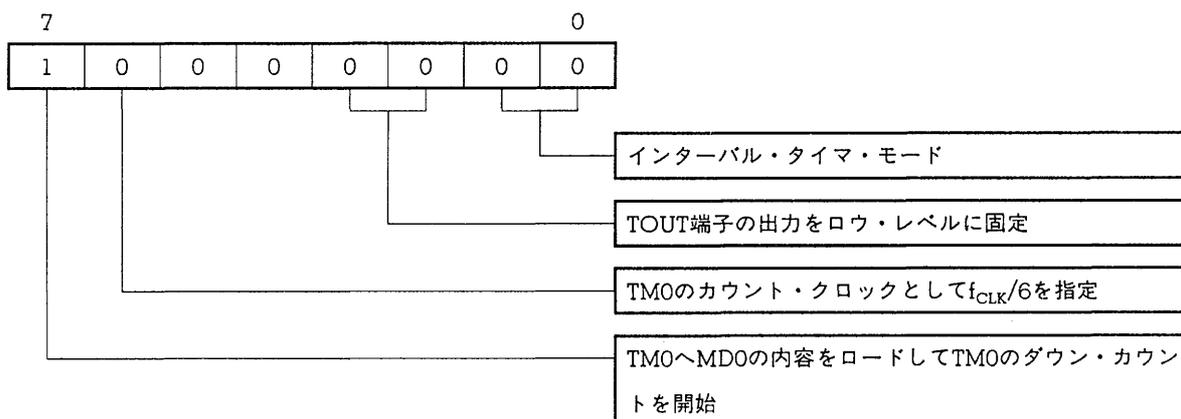
- ① 切り替え先のレジスタ・バンク 6 のPSとベクタPCは、あらかじめイニシャライズする必要があります。

DS0, DS1などの上記以外のレジスタは、必要に応じてイニシャライズを行います。

- ② TMIC0 (タイマ・ユニット 0 の割り込み要求制御レジスタ)



- ③ TMC0 (タイマ・コントロール・レジスタ 0)



- ④ 切り替え先のレジスタ・バンク 6 のSS, SPへ切り替え前のレジスタ・バンク 7 のSS, SPの内容を転送します。

- ⑤ 周辺ハードウェアからの割り込み処理を終了する場合はRETI命令またはRETRBI命令の直前にFINT命令の実行が必要です。また、レジスタ・バンク切り替えから復帰するときはRETRBI命令を使用してください。

カウンタ・ダウンの結果、カウンタ値=0により割り込み要求 (INTTU0) が発生します。

B.3.2 命令によるレジスタ・バンクの切り替え (BRKCS命令とMOVSPA命令)

レジスタ・バンク 7 からレジスタ・バンク 5 へ切り替えます。

```

*****
***                BRKCS reg16 & MOVSPAを使用した切り替え                ***
*****
;
;
;
MOV      REGBANK.BK5.BVPC, OFFSET BANK5      ;①ベクタPCの初期化
MOV      REGBANK.BK5.BPS,  SEG BANK5         ;①P S
MOV      REGBANK.BK5.BDS0, 0                  ;①D S 0
MOV      REGBANK.BK5.BDS1, 0                  ;①D S 1
;
MOV      AW, 5                                ;
BRKCS    AW                                    ;②
;
EI
;
;
=====
;===                レジスタ・バンク 5 の処理                ===
=====
;
;
BANK5:
MOVSPA   ;③
;
RETRBI   ;④

```

- ① レジスタ・バンク 5 のPS, ベクタPCに割り込みルーチンのPS, PCを設定します。
DS0, DS1以外のレジスタは必要に応じて設定します。
- ② BRKCS命令の実行により (reg16) に設定した値のレジスタ・バンクへ切り替わります。
ここではレジスタ・バンク 5 に切り替えるので, 5 をレジスタに設定します。
- ③ 切り替え先のレジスタ・バンク 5 のSS, SPへ切り替え前のレジスタ・バンク 7 のSS, SPの内容を
転送します。
- ④ 新しいレジスタ・バンクからの復帰を行います。この場合, FINT命令を実行する必要はありません。

B.3.3 命令によるレジスタ・バンクの切り替え (MOVSPB命令とTSKSW命令)

レジスタ・バンク 7 からレジスタ・バンク 6 へ切り替えます。

```

;*****
;***          MOVSPB reg16 & TSKSW reg16を使用した切り替え          ***
;*****
;
;
;
MOV      REGBANK.BK6.BPC,  OFFSET BANK6          ;① P C 退避の初期化
MOV      REGBANK.BK6.BPSW,  0E002H              ;① P S W 退避
MOV      REGBANK.BK6.BPS,   SEG BANK6          ;① P S
MOV      REGBANK.BK6.BDS0,  0                   ;① D S 0
MOV      REGBANK.BK6.BDS1,  0                   ;① D S 1
;
;
MOV      AW, 6                                  ;
MOVSPB   AW                                     ;②
TSKSW    AW                                     ;③
;
;
;=====
;===          レジスタ・バンク 6 の処理          ===
;=====
;
;
BANK6:
;
MOV      AW, 7                                  ;④
TSKSW    AW
;

```

① 選択するレジスタ・バンク内のPSとPC退避領域、PSW退避領域、SS、SPは、あらかじめイニシャライズしておく必要があります (B.3.3ではSS、SPはMOVSPB命令によって設定しています)。また、TSKSW命令を使用した場合、選択したレジスタ・バンク内のPSW退避領域、PC退避領域の値がロードされます。

② MOVSPB命令の実行により、切り替え先のSS、SPにはそれぞれ切り替え前の値がセットされます。ここではレジスタ・バンク6に切り替えるので、6をAWレジスタに設定します。

③ TSKSW命令により、切り替え先のレジスタ・バンク6を選択して、あらかじめストアしておいたPC退避領域の内容をPCにロードして分岐します。

MOVSPB命令、TSKSW命令を実行することによって、レジスタ・バンク6に切り替わります。

④ ①のようにレジスタ・バンクの切り替えに必要なイニシャライズを行えば、TSKSW命令の実行によりレジスタ・バンク7を選択して分岐します。ただし、ここではMOVSPB命令を実行していないので、スタックの連続使用はできません。

注意1. TSKSW命令をBRKCS命令や割り込み要求発生によるレジスタ・バンクの切り替え処理中に使用するとPSW退避領域の内容が破壊され、元のバンクに復帰できなくなります。ただし、TSKSW命令によるレジスタ・バンクの切り替え処理中に使用しても問題ありません。

注意2. 切り替え先のPSW退避領域のRBO, RB1, RB2の値と切り替え先のレジスタ・バンク番号は、一致させる必要があります。

B.4 内部データ領域のアクセス

内部データ領域（内蔵RAMと特殊機能レジスタ）が0FE00H-0FFFFHにある場合のアクセスを示します。ただし、ASSUME、ASGNSFRなどのアセンブラ（RA70116-I）疑似命令は設定が終わっている状態です。

```

;*****
;***                      レジスタの初期化                      ***
;*****
;
START:
    SETIDB    0FH 注          ;IDBレジスタで物理アドレスのベースアドレス(0F00H)を設定

    MOV      AW, DATA        ;DATA(0000H)をセグメント・レジスタに設定
    MOV      DS0, AW          ;
;
;=====
;===                      特殊機能レジスタの初期化                      ===
;=====
;
    MOV      P0, 00001111B    ;ポート0(P00-P03)から1を出力して,(P04-P07)から0を出力
    MOV      PM0, 00000000B   ;するように設定
    MOV      PMCO, 00000000B  ;
;
    MOV      PRC, 01000100B   ;内蔵RAMアクセス許可(ビット6→1)
    ;
    MOV      AL, [FE00]       ;内蔵RAMの内容を1バイト取り込む
    ;                          ;AL←[FE00]
    MOV      AL, P0           ;P0の内容をALへ取り込む
    ;

```

注 内部データ領域のアドレスをアセンブラ（RA70116-I）に指示する疑似命令です。この疑似命令の記述に対してアセンブラは、次の命令を生成します。

```

PUSH    DS0
PUSH    0FFFFH
POP     DS0
MOV     DS0:BYTE PTR [0FH],xx
POP     DS0

```

xxの値はオペランドに記述した式の値（0FH）が設定されます。

B.5 タイマ・ユニット

インターバル・タイマ（タイマ1）とワンショット・タイマ（タイマ0）を用いて、それぞれのタイマ・ユニット割り込みを発生させます。

(1) インターバル・タイマ・モード

```

:
;*****
;***          インターバル・タイマ・モードの設定 (約4.9ms)          ***
;*****
;
:
MOV          TMIC2, 00000111B      ;①
MOV          TMC1,  00000000B      ;②
MOV          MD1,   0FF0H          ;③
:

```

(2) ワンショット・タイマ・モード

```

:
;*****
;***          ワンショット・タイマ・モードの設定 (約9.8ms)          ***
;*****
;
:
MOV          TMIC0, 00000001B      ;④
MOV          TMC0,  00001001B      ;⑤
MOV          MD0,   0FF0H          ;⑥
:
;
:
;=====
;===          ベクタ・アドレス No. 28 (INTTU0) の設定          ===
;=====
;
MOV          IY, 28*4
MOV          WORD PTR[IY], OFFSET INTTU0
MOV          WORD PTR[IY+2], SEG INTTU0
:

```

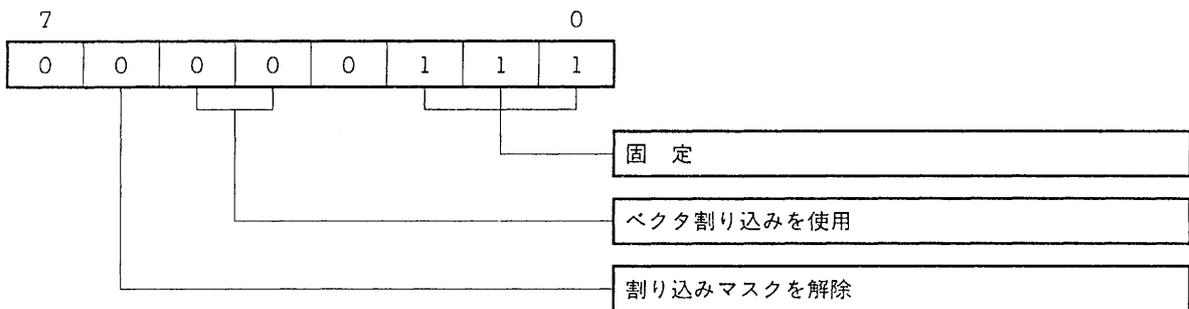
```

;
;=====
;===          ベクタ・アドレス No. 30 (INTTU1) の設定          ===
;=====
;
MOV      IY, 30*4
MOV      WORD PTR[IY],   OFFSET INTTU1
MOV      WORD PTR[IY+2], SEG INTTU1
;
EI
SETI     TMC1, 7          ;⑦
        :
;=====
;===          タイマ・スタート          ===
;=====
;
SETI     TMC0, 5          ;⑧
        :
;=====
;===          タイマ・ユニット割り込み処理          ===
;=====
;
INTTU0:
        :
        FINT
        RETI              ;割り込みからの復帰
;
INTTU1:
        :
        FINT
        RETI              ;割り込みからの復帰

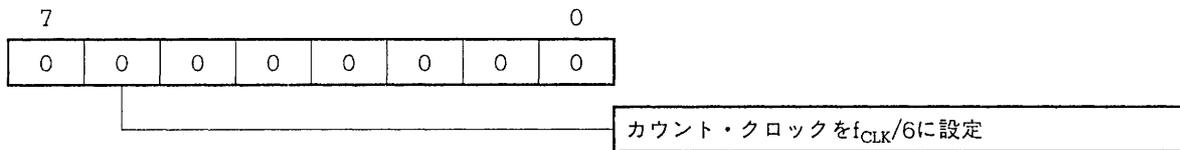
```

(1) インターバル・タイマ・モード

① TMIC2 (タイマ・ユニット割り込み要求制御レジスタ 2)



② TMC1 (タイマ・コントロール・レジスタ1)

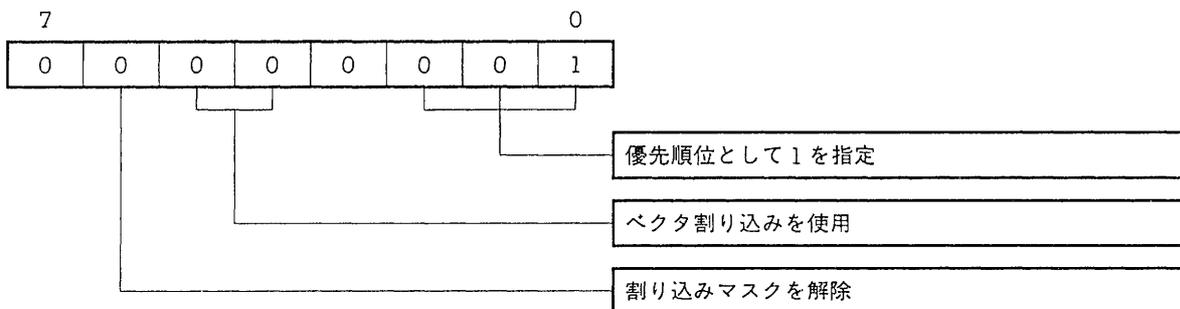


③ MD1

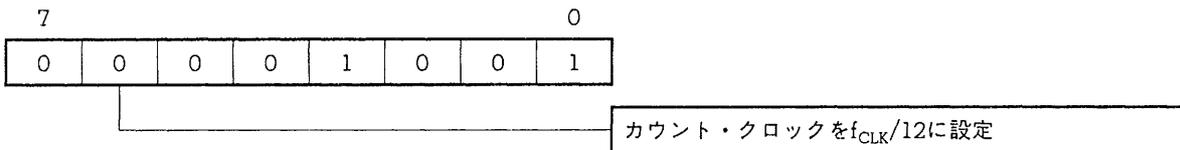
カウント値 (FFOH) を設定して、約4.9 ms間隔でインターバル割り込みが発生します。

(2) ワンショット・タイマ・モード

④ TMIC0 (タイマ・ユニット割り込み要求制御レジスタ0)



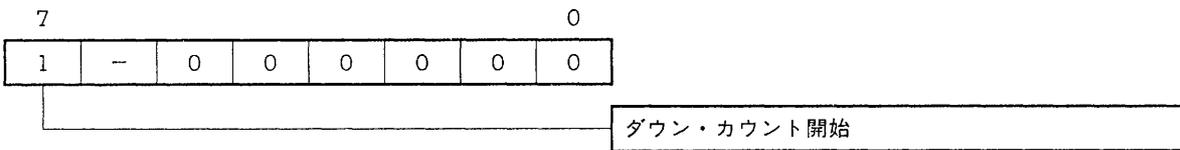
⑤ TMC0 (タイマ・コントロール・レジスタ0)



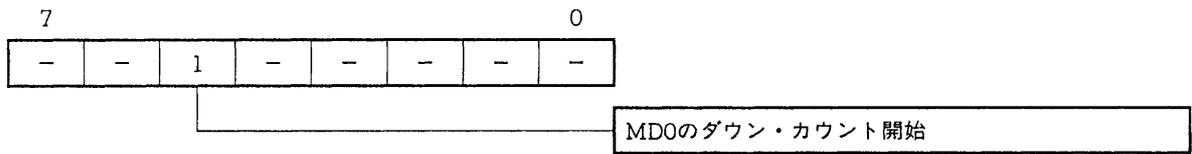
⑥ MD0

カウント値 (FFOH) を設定して、約9.8 ms後に1回だけ割り込み要求が発生します。

⑦ TMC1 (タイマ・コントロール・レジスタ1)



③ TMC0 (タイマ・コントロール・レジスタ0)



B.6 I/Oインタフェース・モード

I/Oインタフェース・モード（受信）をベクタ割り込みで使します。

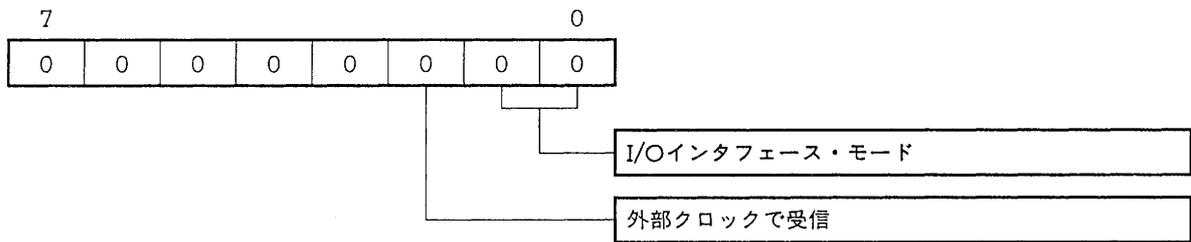
```

;+++++
;+++                                     特殊機能レジスタの設定                                     +++
;+++++
;
MOV     SCM0, 00000000B      ;①
MOV     SCC0, 00000011B      ;②
MOV     BRG0, 130           ;②
;
MOV     SEIC0, 01000011B     ;③
MOV     SRIC0, 10000111B     ;③
;
MOV     P1, 11111111B        ;ポート1(P16)をコントロール・モードに設定し, SCK0出力を行う
MOV     PM1, 01000000B       ;
MOV     PMC1, 00000000B       ;
;
SETI    SCM0, 6              ;④
      :
EI      ;SRIC0(ビット7→1)により, 割り込み許可になり, 受信完了
      :                      ;割り込みを発生します。

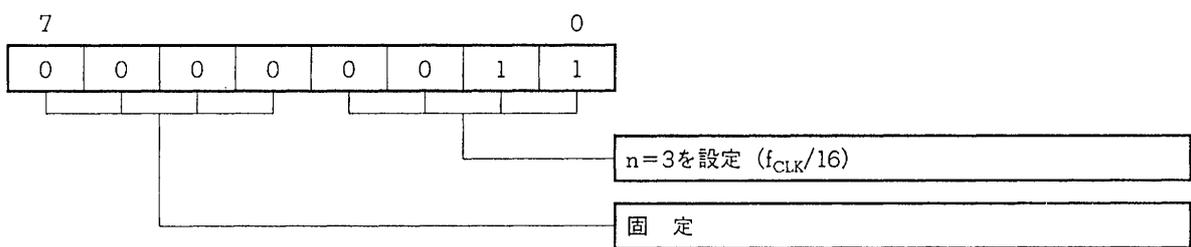
;*****
;***                                     シリアル・データ受信処理（受信完了割り込み処理中）                                     ***
;*****
;
INTSR0:
      :
MOV     AL, RXB0             ;受信データを受信バッファ(RxB0)からALに取り込む
      :
FINT
RETI

```

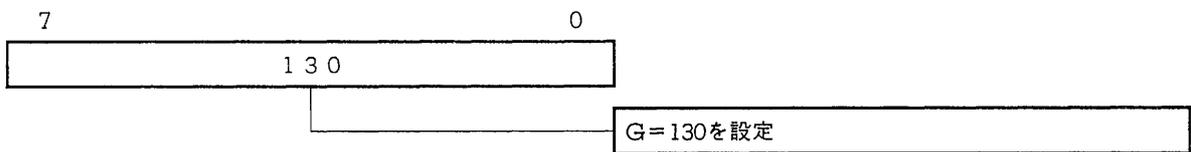
① SCMO (シリアル・モード・レジスタ・チャンネル0)



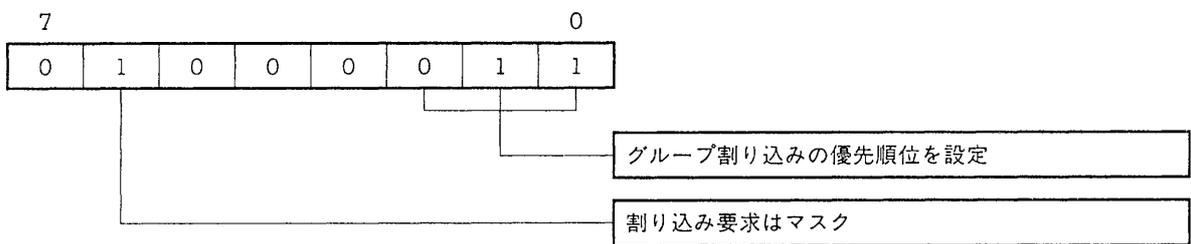
② SCC0 (シリアル制御レジスタ0)



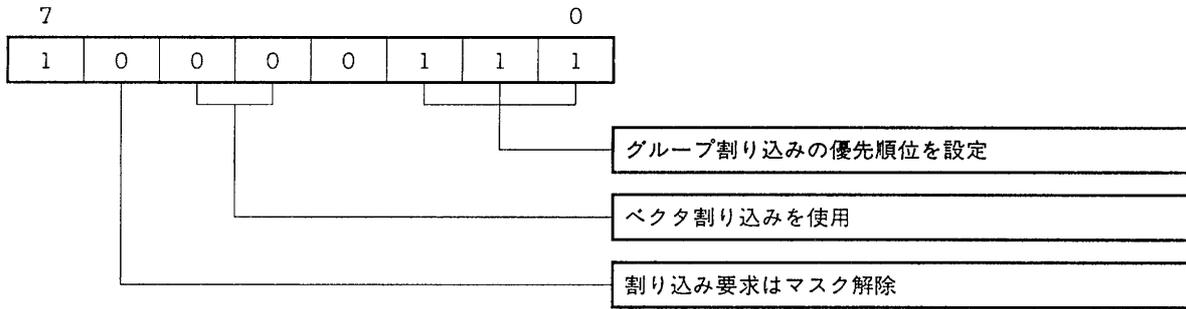
BRG0 (ポーレート・ジェネレータ・レジスタ0)



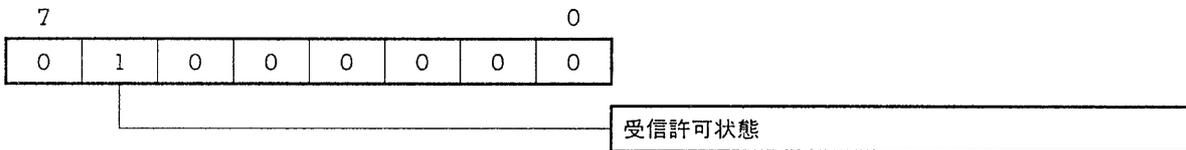
③ SEIC0 (シリアル・エラー割り込み要求制御レジスタ0)



SRIC0 (シリアル受信割り込み要求制御レジスタ0)



④ SCM0 (シリアル・モード・レジスタ・チャンネル0)



B.7 マクロ・サービス

送信データが格納されているメモリの先頭アドレスを(BAR_CODE), マクロ・サービスのチャンネル0の先頭アドレスを(REGBANK)とした場合を例に示します。

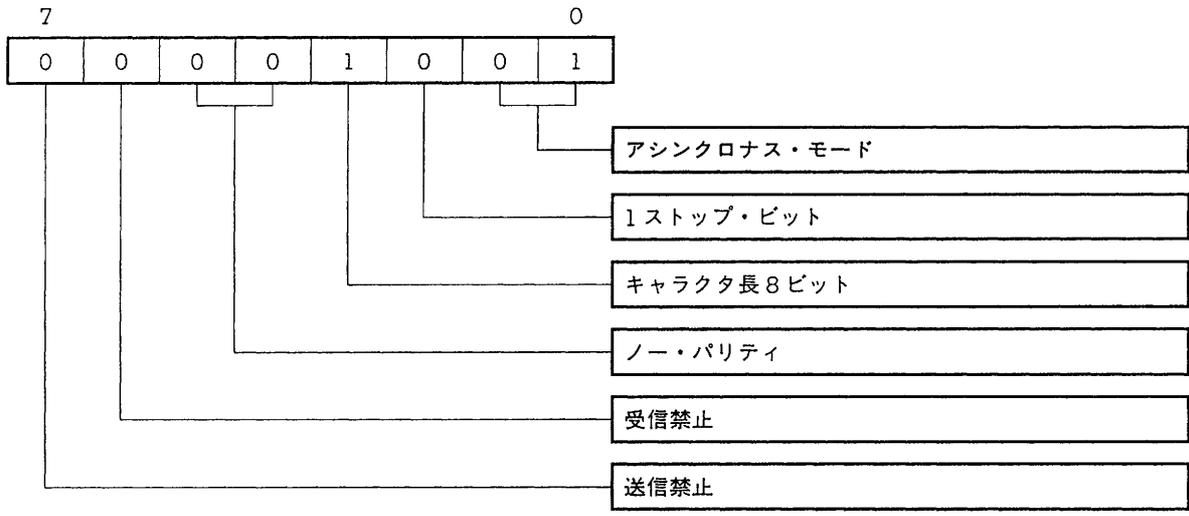
B.7.1 ノーマル・モード (シリアル・インタフェースUART送信)

```

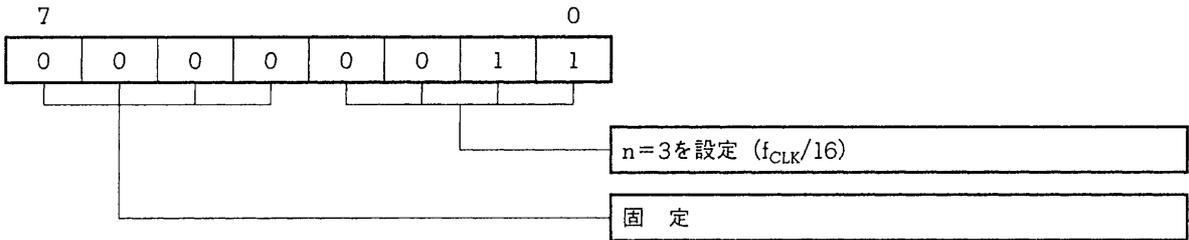
;
;+++++
;+++          特殊機能レジスタの設定          +++++
;+++++
;
MOV          SCMO, 00001001B          ;①
MOV          SCC0, 00000011B          ;②
MOV          BRG0, 130                 ;②
;
MOV          SEIC0, 01000011B         ;③
MOV          STIC0, 00110111B         ;④
;
;*****
;***          マクロ・サービス (チャンネル2) ノーマル・モードの初期化          ***
;*****
;
MOV          STMS0, 00000010B          ;⑤
;
MOV          IY, OFFSET REGBANK+8*2    ;⑥
MOV          BYTE PTR [IY], 15         ;⑦
MOV          BYTE PTR [IY+1], LOW TXB0 ;⑧
MOV          WORD PTR [IY+4], OFFSET BAR_CODE ;⑨
MOV          WORD PTR [IY+6], SEG BAR_CODE ;⑩
;
;*****
;***          レジスタ・バンク3の設定          ***
;*****
;
;          ; B. 3を参照してください
;
;*****
;
SETI         SCMO, 7                   ;⑪
;
;

```

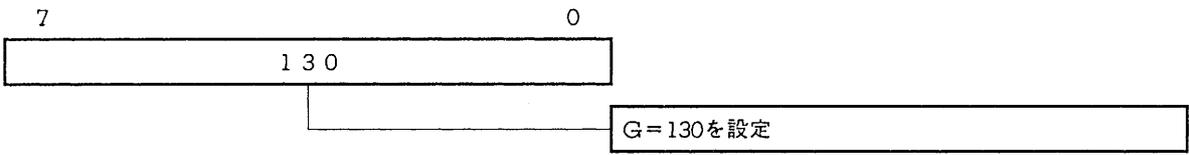
① SCMO (シリアル・モード・レジスタ・チャンネル0)



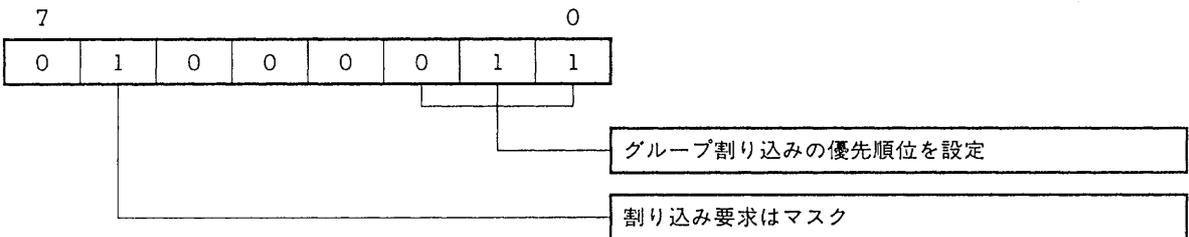
② SCC0 (シリアル制御レジスタ0)



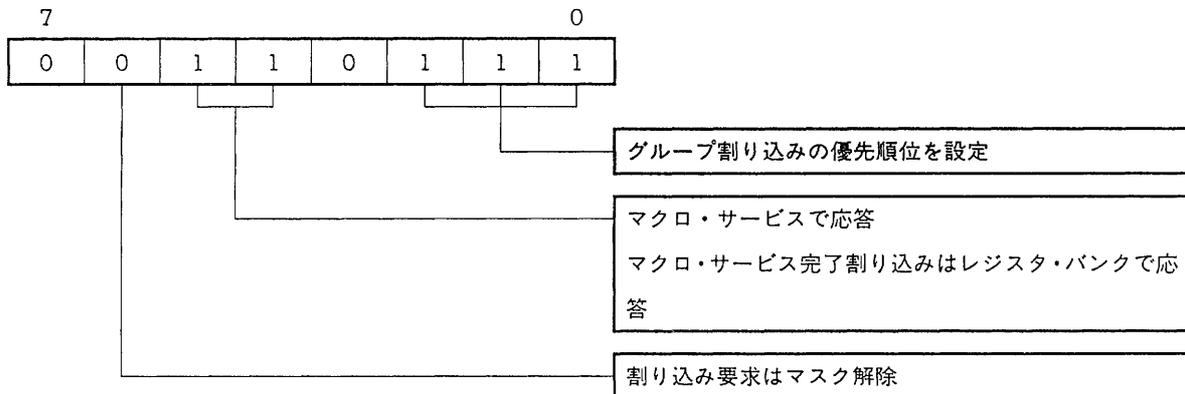
BRG0 (ポー・レート・ジェネレータ・レジスタ0)



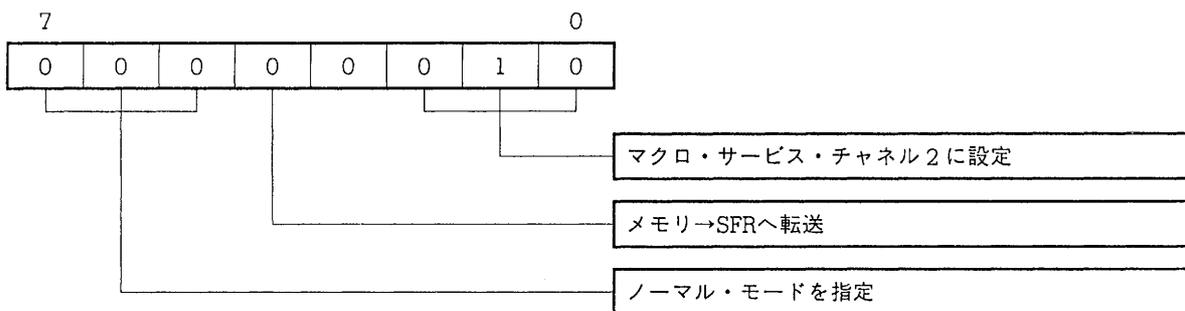
③ SEIC0 (シリアル・エラー割り込み要求制御レジスタ0)



④ STICO (シリアル送信割り込み要求制御レジスタ 0)



⑤ SRMSO (マクロ・サービス制御レジスタ 0)



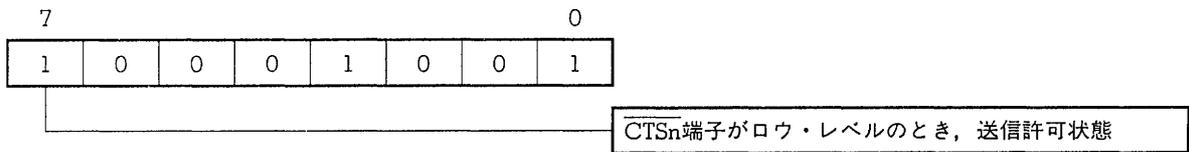
- ⑥ インデクス・レジスタ (IY) に、マクロ・サービス・チャンネル2の先頭オフセット・アドレスを設定
- ⑦ MSC [IY+0]: マクロ・サービスで行う転送回数 (15回) を設定
- ⑧ SFRP [IY+1]: 特殊機能レジスタ (TxBO) のアドレスの下位1バイトを設定
LOW: アセンブラ (RA70116-I) のバイト分離演算子で、式の下位1バイトの値を返します。
- ⑨ MSP [IY+4]: マクロ・サービスでデータ転送の対象となるメモリ・アドレスのオフセット値を設定
- ⑩ MSS [IY+6]: マクロ・サービスでデータ転送の対象となるメモリ・アドレスのセグメント値を設定

データ転送の対象となるメモリ・アドレスは、 $MSS \times 16 + MSP$

[] 内の+nは、各マクロ・サービス・チャンネルの開始アドレスからのオフセットを示します。



① SCMO (シリアル・モード・レジスタ・チャンネル0)



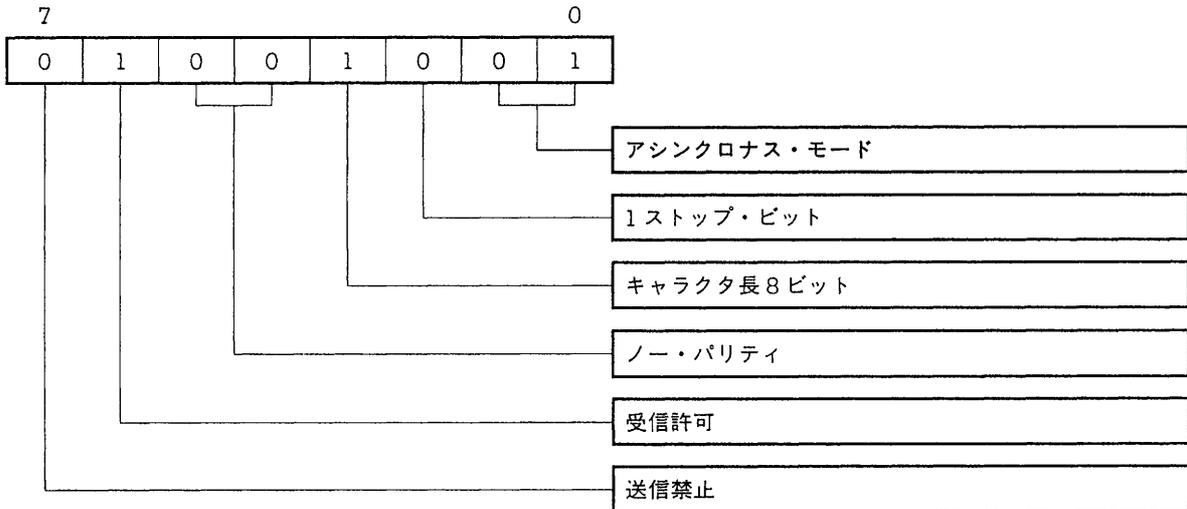
B.7.2 キャラクタ・サーチ・モード (シリアル・インタフェースUART受信)

```

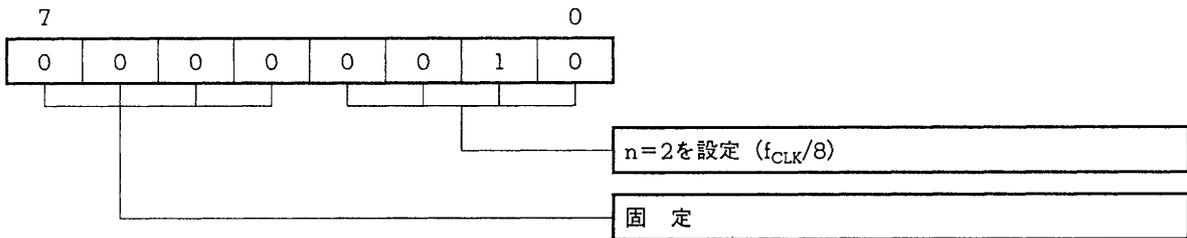
:
;+++++
;+++                特殊機能レジスタの設定                +++
;+++++
;
MOV      SCMO, 01001001B      ;①
MOV      SCC0, 00000010B     ;②
MOV      BRG0, 130           ;
;
MOV      SEIC0, 01000011B    ;③
MOV      SRIC0, 00110111B    ;③
MOV      STIC0, 01000111B    ;③
;
;*****
;***                マクロ・サービス (チャンネル2)   キャラクタ・サーチ・モードの初期化                ***
;*****
;
MOV      SRMS0, 10000010B     ;④
;
MOV      IY, OFFSET REGBANK+8*2 ;⑤
MOV      BYTE PTR [IY], 15    ;⑥
MOV      BYTE PTR [IY+1], LOW RXB0 ;⑦
MOV      BYTE PTR [IY+2], 0AH  ;⑧
MOV      WORD PTR [IY+4], OFFSET BAR_CODE ;⑨
MOV      WORD PTR [IY+6], SEG BAR_CODE ;⑩

```

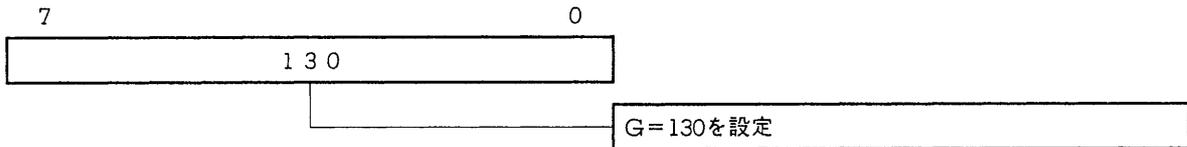
① SCMO (シリアル・モード・レジスタ・チャンネル0)



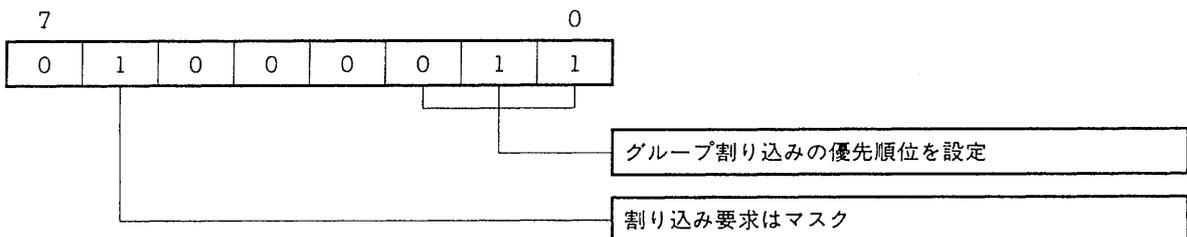
② SCC0 (シリアル制御レジスタ0)



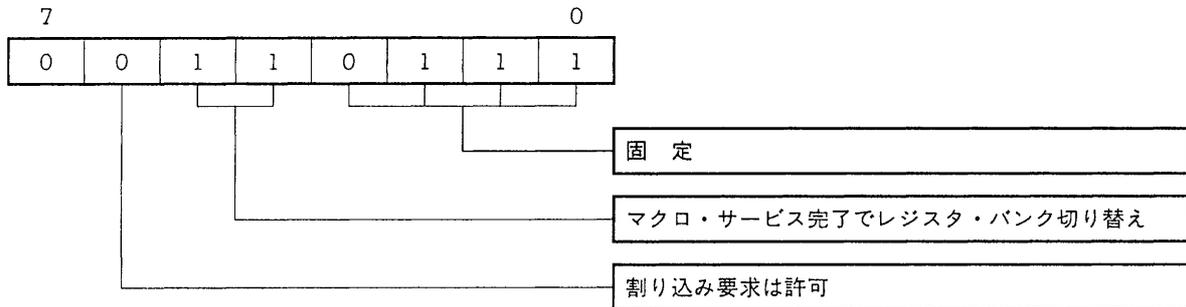
BRG0 (ポー・レート・ジェネレータ・レジスタ0)



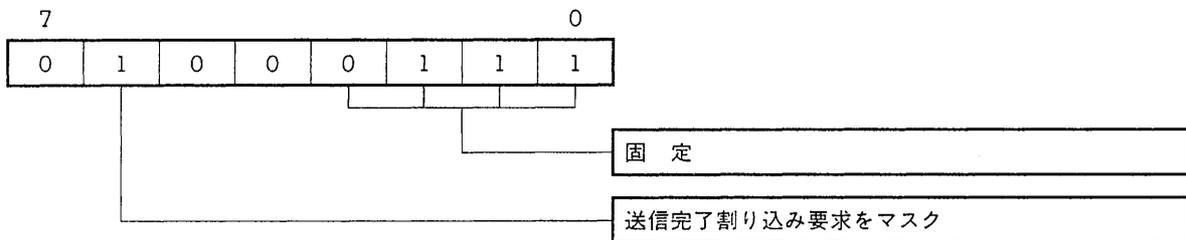
③ SEIC0 (シリアル・エラー・割り込み要求制御レジスタ0)



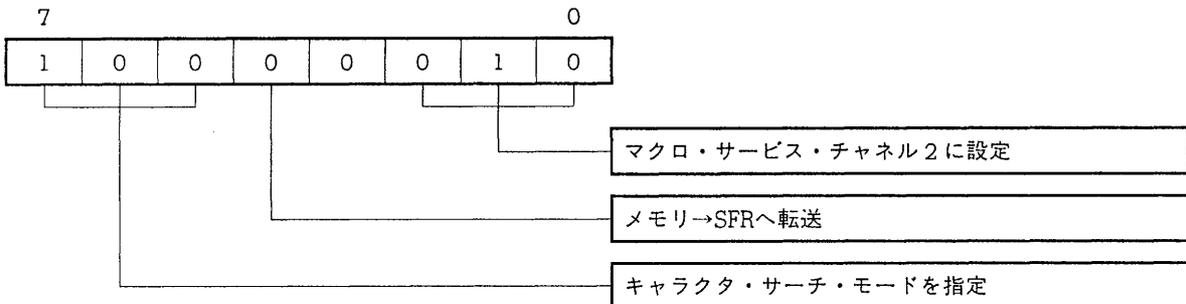
SRIC0 (シリアル受信割り込み要求制御レジスタ0)



STIC0 (シリアル送信割り込み要求制御レジスタ0)



④ SRMS0 (マクロ・サービス制御レジスタ0)



- ⑤ インデクス・レジスタ (IY) に、マクロ・サービス・チャンネル2の先頭オフセット・アドレスを設定
- ⑥ MSC [IY+0]: マクロ・サービスで行う転送回数 (15回) を設定
- ⑦ SFRP [IY+1]: 特殊機能レジスタ (RxBO) のアドレスの下位1バイトを設定
LOW: アセンブラ (RA70116-I) のバイト分離演算子で、式の下位1バイトの値を返します。
- ⑧ SCHR [IY+2]: キャラクタ・サーチ・モード時に比較する8ビットのデータ (OAH) を設定
- ⑨ MSP [IY+4]: マクロ・サービスでデータ転送の対象となるメモリ・アドレスのオフセット値を設定

- ⑩ MSS [IY+6] : マクロ・サービスでデータ転送の対象となるメモリ・アドレスのセグメント値を設定

データ転送の対象となるメモリ・アドレスは、 $MSS \times 16 + MSP$

[] 内の+nは、各マクロ・サービス・チャンネルの開始アドレスからのオフセットを示します。

B.8 DMAコントローラ

転送元のソース・データが格納されているメモリの先頭アドレス (MSG_TBL_SAR) を、転送先 (データ格納メモリ) の先頭アドレス (MSG_TBL_DAR) とした場合を例に示します。

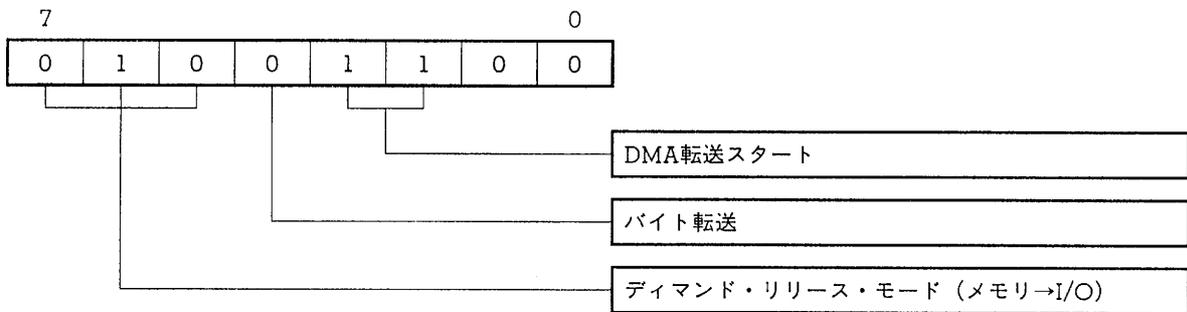
B.8.1 ディマンド・リリース・モード

```

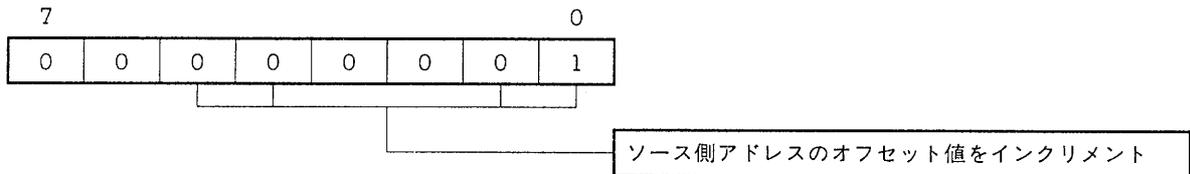
;
;*****
;***          DMAコントローラ (チャンネル0) の初期化          ***
;*****
;
MOV     DMAM0, 01001100B           ;①
MOV     DMAC0, 00000001B         ;②
;
MOV     SAR0L, } MSG_TBL_SAR      ;③
MOV     SAR0M, }
MOV     SAR0H, }
MOV     TCO, 3599                 ;④
;
;
;

```

① DMAM0 (DMAモード・レジスタ0)



② DMAC0 (DMAコントロール・レジスタ0)



③ DMA転送のソース側のアドレスを設定

④ DMA転送回数 (3600回) の設定

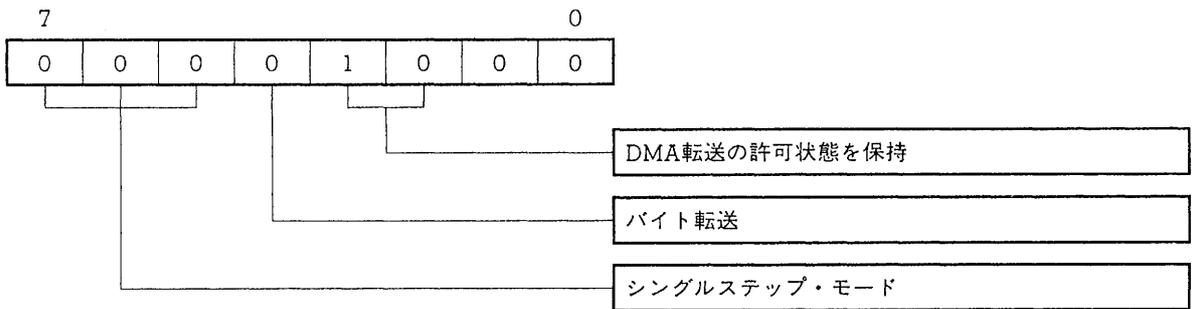
バイト転送なので、データ数は3600バイト必要です。

B.8.2 シングルステップ・モード

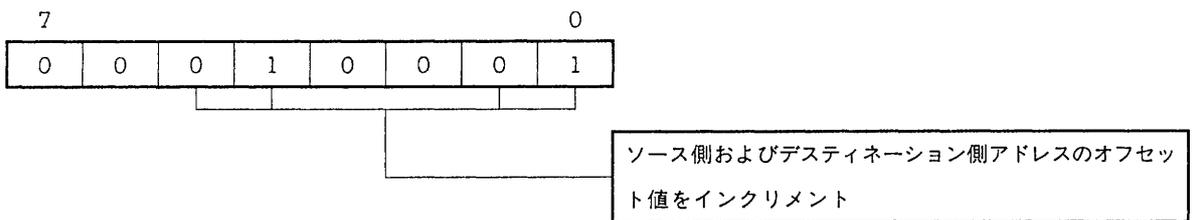
```

;
;*****
;***          DMAコントローラ (チャンネル0) の初期化          ***
;*****
;
MOV          DMAM0, 00001000B          ;①
MOV          DMAC0, 00010001B          ;②
;
MOV          SAR0L, } MSG_TBL_SAR      ;③
MOV          SAR0M, }
MOV          SAR0H, }
MOV          DAR0L, } MSG_TBL_DAR      ;④
MOV          DAR0M, }
MOV          DAR0H, }
MOV          TC0, 3599                  ;⑤
;
          :
SETI        DMAM0, 2                    ;DMA転送スタート
          :
    
```

① DMAM0 (DMAモード・レジスタ0)



② DMAC0 (DMAコントロール・レジスタ0)



- ③ DMA転送のソース側のアドレスを設定
- ④ DMA転送のデスティネーション側を設定
- ⑤ DMA転送回数（3600回）の設定

バイト転送なので、データ数は3600バイト必要です。

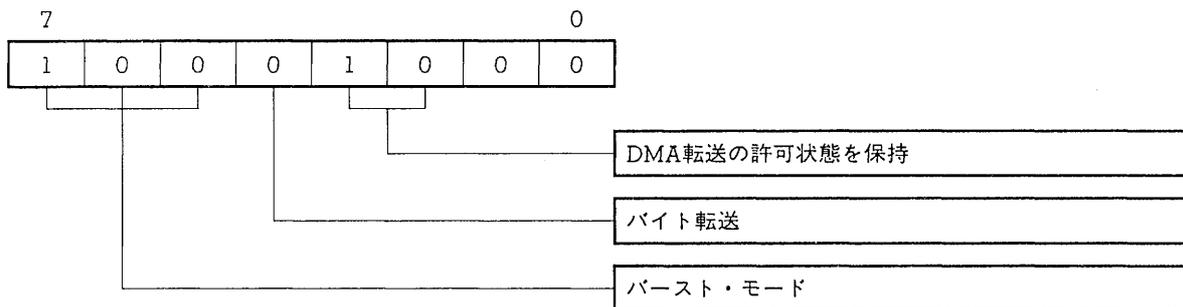
B.8.3 バースト・モード

```

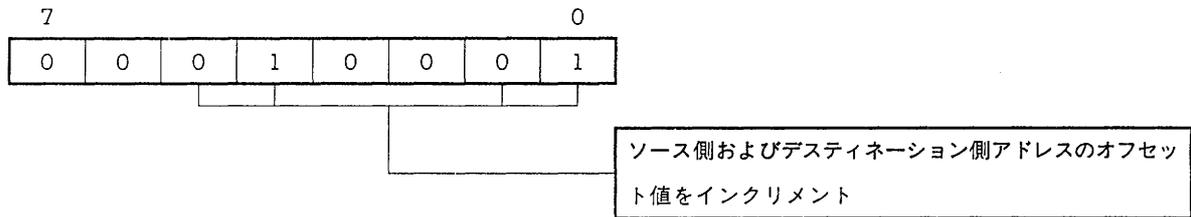
;
;*****
;***          DMAコントローラ（チャンネル0）の初期化          ***
;*****
;
MOV     DMAM0, 10001000B           ;①
MOV     DMAC0, 00010001B         ;②
;
MOV     SAR0L, } MSG_TBL_SAR      ;③
MOV     SAR0M, }
MOV     SAR0H, }
MOV     DAR0L, } MSG_TBL_DAR      ;④
MOV     DAR0M, }
MOV     DAR0H, }
MOV     TC0, 3599                ;⑤
;
;
SETI    DMAM0, 2                  ;DMA転送スタート
;

```

① DMAM0 (DMAモード・レジスタ0)



② DMAC0 (DMAコントロール・レジスタ0)



- ③ DMA転送のソース側のアドレスを設定
 - ④ DMA転送のデスティネーション側のアドレスを設定
 - ⑤ DMA転送回数 (3600回) の設定
- バイト転送なので、データ数は3600バイト必要です。

付録C Q&A

C.1 内部CPU機能 … 296

Q.1.1 メモリ空間において

- (1) セグメント内オフセット・アドレスがFFFFH番地にワード・アクセスを行うと、実アドレスはどうか？ … 296
- (2) V25, V35ファミリをリセットすると、内部データ領域はFFE00H-FFFFFHにロケートされるが、内部データ領域の特殊機能レジスタ領域とブート・ストラップ・アドレスと重ならないか？ … 296

Q.1.2 V25ファミリとメモリをインタフェースするときのメモリ・アクセス時間の考え方は？ … 296

Q.1.3 I/O空間において

- (1) FFO0H-FFFFHは予約領域となっているが、どういうときに使用する？ … 297
- (2) I/O空間のアドレス・デコーダを上位8ビット (A8-A15) に対して行わなかった場合、使用上何か問題となる？ … 297

Q.1.4 メモリ・マップトI/Oを使用する場合、

- (1) CPUの内蔵メモリのアクセスをするときに、IDBレジスタの書き換えはユーザが行わずにCPUが行う？ … 297
- (2) ハードウェア割り込みに対してレジスタ・バンクを使用したい場合、レジスタ・バンク内の領域に直接書き込む？ … 297

C.2 割り込み機能 … 298

Q.2.1 同一グループ内の多重割り込みは可能？ … 298

Q.2.2 外部割り込み優先順位について

- (1) INT割り込み実行中に、INTPOもしくはINTP1入力の割り込み要求があった場合、その割り込みを受け付ける？ … 299
- (2) INT割り込み実行中に、INTPOとINTP1に同時に割り込み要求入力があるとどうなる？ … 299
- (3) INTPO, もしくはINTP1の割り込みを実行中にINT入力があると、INT割り込みを受け付ける？ … 299

Q.2.3 レジスタ・バンク切り替え機能について

- (1) 同一グループ内の複数の割り込みに対しては、同一のレジスタ・バンクを使用する？ またその場合、多重割り込み処理をできる？ … 301
- (2) レジスタ・バンクが使用できない割り込みはある？ … 301
- (3) レジスタ・バンクの切り替えは、CPUが自動的に行う？ もし、CPUが自動的に行うなら切り替え前のレジスタ・バンクの内容を退避したり、復帰できる？ … 301
- (4) 割り込み要求レジスタのPRO-PR2は、レジスタ・バンク切り替え先のレジスタ・バンクを指定する。また、同一グループ内では最高優先以外は、この値は7 (PRO-PR2=1, 1, 1) 固定となっている。最高優先以外は、レジスタ・バンク7に固定されてしまう？ … 301

Q.2.4 「NMIとINT, およびソフトウェア割り込みを除く割り込み処理が終了する直前にFINT命令を実行する」とユーザーズ・マニュアルの説明にあるが、

- (1) 入出力命令割り込みとFPO命令割り込みにFINT命令は必要？ … 302
- (2) FINT命令を割り込みが終了する直前以外の場合に実行してもよい？ … 302

Q.2.5 割り込み要求制御レジスタで、ビット7(IF)をソフトウェアによって1にセットしたときのV25, V35ファミリの動作(EI状態)はどうか？ … 302

- Q.2.6** マクロ・サービス機能は、割り込み要求によって特殊機能レジスタ領域とメモリ空間の間でデータを行う機能だが、
- (1) マクロ・サービス機能でブロック転送はできる？ … 303
 - (2) マクロ・サービス機能で、特殊機能レジスタを用いずI/O ↔ メモリ間の転送をできる？ … 303
 - (3) 特殊機能レジスタ ↔ メモリ間のデータ転送の際、アドレスおよびバス制御信号は外部に出力される？ … 303
 - (4) マクロ・サービス機能の応用例はどのようなものがある？ … 303
 - (5) 多くのタスク処理をする場合の効率のよいマクロ・サービス・チャンネルの使用法はある？ … 303
 - (6) MSC (マクロ・サービスで行う転送数) が0のとき、割り込み発生はどここのベクタに行く？ … 303
- Q.2.7** INTAKサイクルにウエイト・ステート挿入はできる？ … 304
- Q.2.8** V25ファミリで1ワードを読み出ししたり、書き込んだりする場合、下位バイトと上位バイトに分けて実行するが、下位バイトの読み出しと上位バイトの読み出しの間で割り込み要求がきたときはその割り込みを受け付ける？ … 304
- C.3** バス制御 … 305
- Q.3.1** V35ファミリのメモリ・バンク構成について
- (1) メモリ・バンクの選択にUBE端子とMSTB端子でチップ・セレクトするのはなぜ？MSTB端子が下位アドレスの読み出しをするのに、UBE端子が必要な理由は？ … 305
 - (2) リード・サイクルでは、A0信号およびUBE信号は必要か？ … 305
 - (3) メモリ・サイクルのアドレス時分割出力の2回目のバス・サイクルで、A18ビットの物理アドレスが出力されるのはなぜ？ … 305
 - (4) アドレスは上位ビット、下位ビットのどちらが先に出力される？ … 305
 - (5) プログラム・フェッチは、バイト単位でもできる？ … 305
- Q.3.2** 内蔵RAM領域、内蔵ROM領域のアクセスについて
- (1) 内蔵RAM領域のアクセス時、バス・サイクル中にウエイト・コントロール・レジスタ(WTC)によって、ウエイト・ステートを挿入することはできる？ … 306
 - (2) V25, V35の内蔵ROM領域に対して、メモリ・アクセスを行った場合、外部にバス・アドレス制御信号は出力される？ … 306
- Q.3.3** READY信号の入力について
- (1) ウエイト・ステートが0ステート、1ステート、2ステート固定のとき、READY信号が“L”になると、CPUは異常動作する？ … 307
 - (2) V25で、TAWを2ステート挿入し、READY端子を“L”にしてTWを1ステート挿入する場合のタイミングの規定は？ … 307
 - (3) READY信号の入力を非同期 (t_{SCRY} , t_{HCRY} の条件を満たさない場合)とした場合、CPUは暴走する？ … 307
 - (4) 最初、READY信号をインアクティブ (“L”)とし、バス・サイクルを抜けるときのみREADY信号をアクティブ (“H”)とすることはできる？ … 307
- Q.3.4** メモリ・サイクルにおいて
- (1) V25ファミリでの、MREQ信号とMSTB信号の違いは？ … 308
 - (2) CPUが内部のメモリをアクセスしたとき、MREQ信号、MSTB信号はインアクティブ状態となる？ … 308
 - (3) V35ファミリで、メモリ・リード・タイミングではMSTB信号はウエイト挿入により伸びるが、メモリ・ライト・サイクルではMSTB信号がウエイトで伸びないのはなぜ？ … 308

- Q.3.5 DMAサイクルからCPUのバス・サイクル（フェッチやデータ・アクセス）へ移るとき、アイドル・サイクルは入る？ … 308
- Q.3.6 V25ファミリのユーザーズ・マニュアルでは、I/Oリード/ライト・サイクルの図においてCLKOUT信号の立ち下がりでデータの読み込みや書き込みを行っているが、設計時にユーザーズ・マニュアルの図と同じにする必要がある？ … 309
- Q.3.7 V25ファミリのメモリ・リード・タイミングにおいて
- (1) t_{DADR} , t_{DMRD} , t_{DMSD} はともに満たす必要がある？ … 309
 - (2) \overline{MREQ} 信号の立ち下がりにデータ遅延時間 t_{DMRD} 以内のスピードが必要？ … 309
 - (3) \overline{MREQ} 信号の立ち上がり (t_{WMRL}) でデータを読み込む？ … 309
- Q.3.8 V25ファミリのメモリおよびI/Oのリード/ライト・サイクルのタイミングで、 t_{DAIS} , t_{DAMR} はウエイトを挿入すると伸びる？ … 310

C.4 DMAコントローラ … 311

- Q.4.1 V25, V25SのDMA転送タイミングで、DMARQ信号が入力されてからDMAAK信号が出力されるまでの所要時間のMIN.値は？ … 311
- Q.4.2 DMAモード・レジスタについて
- (1) DMARQ信号の入力時にEDMAビットが0であった場合、DMA要求は受け付けられる？ … 312
 - (2) レジスタ設定に対するDMARQ信号の入力に規定はある？ … 312
 - (3) DMA転送を一時停止させる制御は、EDMAビットで可能？ … 312
- Q.4.3 V25ファミリにおいて、メモリ→I/O間のDMA転送では、I/Oに対するアクセスはどのようにして決定する？ … 312
- Q.4.4 V25, V35とV25+, V35+の違いについて
- (1) DMA転送アドレスの指定方法の違いはある？ … 313
 - (2) V25+, V35+でのDMA転送において、転送レートの向上の直接の要因はある？ … 313
- Q.4.5 DMAの1回の転送バイト数は指定できる？ … 313
- Q.4.6 DMA要求を取り消したときのCPUの動作について、以下のタイミングの場合はどのように動作する（ディマンド・リリース・モード以外）？ … 314
- Q.4.7 V25において、DMA転送が終了してからDMA割り込みが発生するまでの時間はどのくらい？ … 315
- Q.4.8 バースト・モードのDMA転送において、V25+, V35+のDMA転送中にリフレッシュ要求があるとDMAタイミングはどうなる？ … 315
- Q.4.9 シングルステップ・モードとバースト・モードでは、DMARQ信号を常に“H”とし、DMA処理をDMAモード・レジスタのEDMAビットとTDMAビットで制御するといった、非同期なDMA転送処理は可能？ … 316
- Q.4.10 1転送モードにおいて
- (1) アドレス・バス上にメモリのアドレスが出力された場合、I/Oアドレスはどこで指定する？ … 317
 - (2) DMAモード・レジスタのEDMAビットを0から1に変更する前に、DMARQ信号がアクティブであった場合、DMA要求は受け付けられる？ … 317
 - (3) ユーザーズ・マニュアルでは、1転送モードの転送対象はメモリ→I/O間の転送しか記載されていないが、I/O→メモリ間の転送の場合は R/\overline{W} 信号のレベルだけが異なると考えていい？ また、I/O→メモリ間の転送では、DMAサービス・チャンネルのアドレス指定は必要？ … 317
 - (4) V25ファミリでDMA要求を受け付けるためのDMARQ信号のパルス幅のMIN.値は？ … 317

Q.4.11 V25+, V35+のダイヤモンド・リリース・モードについて

- (1) DMARQ信号による制御で1バイトずつ連続転送するとき、DMA転送サイクル以外の期間は、CPUのバス・サイクルは動作する？ … 318
- (2) DMARQ信号による制御を用いて、DMA転送を行うためには、どのような制御を行う？ … 318

C.5 クロック発生回路 … 319**Q.5.1 プロセッサ・コントロール・レジスタ (PRC) について**

- (1) PCK0, PCK1の各ビットにより発振周波数の分周比を変える場合、CPUの動作に影響があるか？ … 319
- (2) PRCのRAMENビットによって内部RAMのアクセスを禁止した場合、同一アドレスの外部メモリにアクセスはできる？また、このときレジスタ・バンクは使用できる？ … 319

C.6 タイマ・ユニット … 320**Q.6.1 タイマ・ユニットをインターバル・タイマ・モードに設定した場合、**

- (1) タイマ0, タイマ1は16ビット・フル・カウント固定？ … 320
- (2) タイマ0のTM0レジスタに設定した周期を方形波にして、TOUT端子に出力することは可能？ … 320

Q.6.2 V25で、インターバル・タイマ・モードでシステム・クロック周波数 (f_{CLK}) が8 MHzの場合、タイマ・コントロール・レジスタのTCLKnビット=0を使用するとき、20 msのインターバル・タイマ値を設定するためにMDnレジスタにセットする値は？ … 320**C.7 シリアル・インタフェース … 321****Q.7.1 ボー・レートが750 kbps以上のデータ送信/受信で、データ間に1クロック以上のアイドル時間を挿入するということは、1ビットずつに1クロックを入れるということ？ … 321****Q.7.2 V25, V35ファミリと別のマイクロコンピュータをシリアル・インタフェースする場合、シリアル・ラインへのノイズの混入などにより、データが1ビットずれて復帰しない現象があるときの対処法はある？ … 321****Q.7.3 V25ファミリのシリアル・インタフェースで、アシンクロナス・モードを使用して送信を行う場合、**

- (1) プログラムによるマクロ・サービス起動後、実際にスタート・ビットが出力されるまでの時間はどのくらい？ … 322
- (2) ボー・レートを4800 bpsとしたとき、プログラムで送信バッファにデータをセットしてから、実際にスタート・ビットが出力されるまでの時間はどのくらい？ … 322

Q.7.4 I/Oインタフェース・モードの送信で、送信クロックを外部から入力することはできる？ … 323**Q.7.5 V35ファミリにおいて、I/Oインタフェース・モードでマクロ・サービスの応答時間（最大転送レート）は？ … 323****Q.7.6 シリアル・インタフェース割り込み要求において**

- (1) 割り込み要求制御レジスタのビット7(SEFn, SRFn, STFn)を0にクリアする条件は？ … 324
- (2) SRFnビットが立つタイミングは？ … 324
- (3) シリアル・インタフェース割り込みはレジスタ・バンク切り替え応答が可能？ … 324
- (4) マクロ・サービスで受信完了割り込み処理中に受信エラー割り込みが発生した場合、それまで実行中の割り込み処理は保留になる？ … 324

C.8 スタンバイ機能 … 325

- Q.8.1 STOPモードをNMIにより解除した場合の発振安定時間は？ … 325
- Q.8.2 STOPモードをNMIにより解除した場合のCPUが動作するまでの V_{DD} 消費電流は？ … 325

C.9 リセット機能 … 326

- Q.9.1 システム・リセットがかかったときの V_{DD} 消費電流は？ … 326
- Q.9.2 V35ファミリにおいて、 $\overline{\text{RESET}}$ 信号のレベルを“L”から“H”に立ち上げて、システム・リセットを解除するとき、
- (1) CPUが動作するまでの時間は？ … 327
 - (2) $\overline{\text{REFRQ}}$ 信号が出力されるまでの時間は？ … 327

C.10 その他 … 328

- Q.10.1 V25とV25+, またはV35とV35+をプログラムで区別する方法はある？ … 328
- Q.10.2 V25とV25S, またはV35とV35Sの違いは？ … 328
- Q.10.3 IC端子を外部でプルアップ抵抗を介して“H”に固定するのはなぜ？ … 329
- Q.10.4 命令実行時間（クロック数）の“EA+6+T”のEAとTは何を表している？ … 329
- Q.10.5 転送命令MOVについて、「MOV mem reg」の場合に内蔵RAMアクセス禁止のクロック数が“EA+2”でウエイトが入らないのはなぜ？ … 330

C.1 内部CPU機能

Q.1.1

メモリ空間において

- (1) セグメント内オフセット・アドレスがFFFFH番地にワード・アクセスを行うと、実アドレスはどうなる？
- (2) V25, V35ファミリをリセットすると、内部データ領域はFFE00H-FFFFFFHにロケートされるが、内部データ領域の特殊機能レジスタ領域とブート・ストラップ・アドレスと重ならないか？

A.1.1

- (1) V25, V35ファミリでは、セグメントの境界（オフセット・アドレス=FFFFHの次の番地）にまたがるようにワード・アクセスを行うと、セグメント内の1バイト（オフセット・アドレス=FFFFH番地）とセグメント外の1バイト（次のセグメントの先頭番地）をアクセスします。V20, V30, V40, V50では、同じセグメント内の0000H番地をアクセスしますのでV25, V35ファミリとは異なっています。
- (2) アドレスは重なりますが、内部データ領域はプログラム・フェッチの対象になりません。プログラム・フェッチのためのメモリ・アクセス時は、必ず外部にバス制御信号（ $\overline{\text{MREQ}}$, $\text{R}/\overline{\text{W}}$, $\overline{\text{MSTB}}$ など）が出力されますので、外部メモリが対象となります。ただしROM内蔵版($\mu\text{PD70P322}$)は制御信号を出力しません。

特殊機能レジスタ領域は、データ・アクセスのみ対象となります。

Q.1.2

V25ファミリとメモリをインタフェースするときのメモリ・アクセス時間の考え方は？

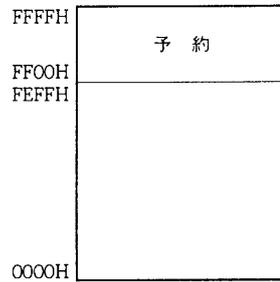
A.1.2

V25ファミリとメモリをインタフェースする場合、V25ファミリのスペックの t_{DADR} をメモリのアドレス・アクセス時間、 t_{DMRD} をメモリの $\overline{\text{OE}}$ アクセス時間に対応させてください。

Q. 1.3

I/O空間において

- (1) FFO0H-FFFFHは予約領域となっているが、どういうときに使用する？



- (2) I/O空間のアドレス・デコードを上位8ビット (A8-A15) に対して行わなかった場合、使用上何か問題となる？

A. 1.3

- (1) 予約領域とは将来の品種展開において使用する可能性のある領域で、特に使用していません。
 (2) 使用上問題とはなりません。

Q. 1.4

メモリ・マップトI/Oを使用する場合、

- (1) CPUの内蔵メモリのアクセスをするときに、IDBレジスタの書き換えはユーザが行わずにCPUが行う？
 (2) ハードウェア割り込みに対してレジスタ・バンクを使用したい場合、レジスタ・バンク内の領域に直接書き込む？

A. 1.4

- (1) ユーザがIDBレジスタをセットする必要があります。特殊機能レジスタのアクセスは、通常のメモリ・アクセスと同様にプログラムします。
 (2) レジスタ・バンクは、IDBレジスタに従ってメモリ上にリロケートされます。したがって、レジスタ・バンク領域へ直接書き込みを行ってください。

C.2 割り込み機能

Q.2.1

同一グループ内の多重割り込みは可能？

A.2.1

同一グループ内での多重割り込みはできません。同一グループ内での割り込みフラグが同時に立っている場合、優先順位が最高位の割り込みが受け付けられて、次にFINT命令が実行されるまで同一グループの他の割り込みは受け付けられません。

したがって、たとえば外部割り込みのINTP1を実行中にINTPOの割り込み要求が発生しても、INTPOとINTP1の多重処理はできません。

Q.2.2

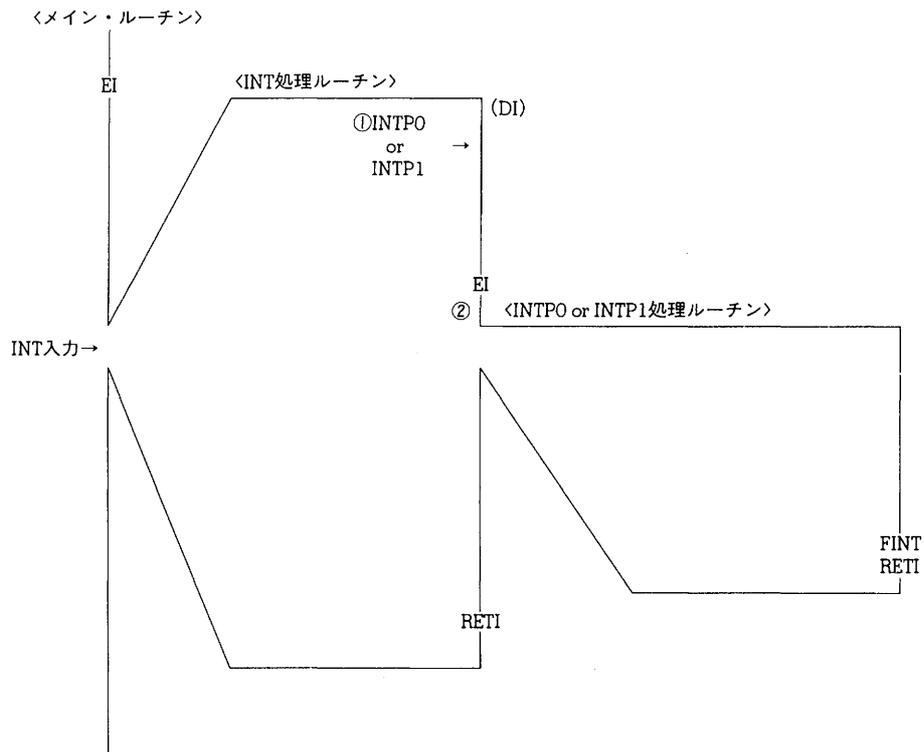
外部割り込みの優先順位について

- (1) INT割り込み実行中に、INTPOもしくはINTP1入力の外部割り込み要求があった場合、その割り込みを受け付ける？
- (2) INT割り込み実行中に、INTPOとINTP1に同時に割り込み要求入力があるとうなる？
- (3) INTPO、もしくはINTP1の割り込みを実行中にINT入力があると、INT割り込みを受け付ける？

A.2.2

- (1) INT割り込み処理ルーチンを実行中に、INTPOもしくはINTP1入力があった場合、割り込みフラグがEI状態であれば受け付けます。

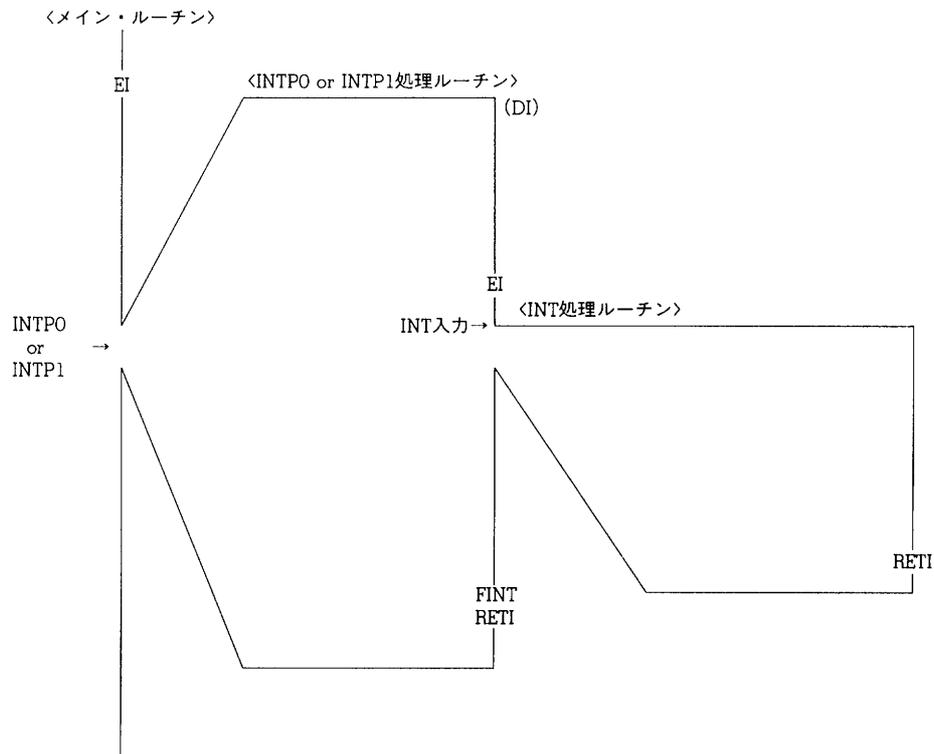
以下に、INTと、INTPOまたはINTP1による割り込みルーチン例を示します。



- ① INTが受け付けられて割り込み禁止状態 (DI) になっているため、INTPOまたはINTP1の割り込み要求が発生しても受け付けられない。
- ② 割り込みフラグを許可状態 (EI) にすると、保留されていたINTPOまたはINTP1の割り込み処理を開始する。

- (2) INT割り込み処理ルーチンを実行中に、INTPOとINTP1が同時に入力された場合、同一グループ内での多重割り込みは行いませんので、INTPOが受け付けられてINTP1は保留されます。
- (3) INT割り込みは多重処理制御を受けません。このため、INT割り込みは、割り込みフラグがEI状態であればいつでも受け付けられます。

以下にINTPOまたはINTP1と、INTの割り込み処理ルーチン例を示します。



Q.2.3

レジスタ・バンク切り替え機能について

- (1) 同一グループ内の複数の割り込みに対しては、同一のレジスタ・バンクを使用する？またその場合、多重割り込み処理をできる？
- (2) レジスタ・バンクが使用できない割り込みはある？
- (3) レジスタ・バンクの切り替えは、CPUが自動的に行う？もし、CPUが自動的に行うなら切り替え前のレジスタ・バンクの内容を退避したり、復帰できる？
- (4) 割り込み要求レジスタのPRO-PR2は、レジスタ・バンク切り替え先のレジスタ・バンクを指定する。また、同一グループ内では最高優先以外は、この値は7 (PRO-PR2=1, 1, 1) 固定となっている。最高優先以外は、レジスタ・バンク7に固定されてしまう？

A 2.3

- (1) 同一グループ内の複数の割り込みに対して、レジスタ・バンク切り替えによる応答を設定した場合、それらの割り込みは同一のレジスタ・バンクを使って応答します。
しかし、V25, V35では、どの割り込みの発生によってそのレジスタ・バンク使用の処理ルーチンに分岐したかを調べる方法がなく、同一グループ内の2つ以上の割り込みをレジスタ・バンク切り替えによって応答させることはできません。ただし、同一の割り込み処理を行うことはできます。
一方、V25+, V35+ではレジスタ・バンク切り替え応答後、割り込み要因レジスタ (IRQS) を利用して、どの割り込みが受け付けられたか調べることができます。
- (2) NMI, INT, INTTB, ソフトウェア割り込み (BRKCS命令を除く) は、レジスタ・バンク切り替え割り込みができません。
- (3) レジスタ・バンク切り替えはCPUが自動的に行います。この際、PC, PSWの退避を切り替え後のバンクに対して行いますが、その他のレジスタはそのままとなり、切り替え前のレジスタの中に保持されません。復帰もPC, PSWのみ行います。
- (4) バンク7に固定されません。
同一グループを形成する割り込みソースのレジスタ・バンク切り替え時のバンク番号は、優先順位と同様にグループ内の最高優先順位をもつ割り込みソースの割り込み要求制御レジスタ (PRO-PR2) によって決定します。

Q.2.4

「NMIとINT, およびソフトウェア割り込みを除く割り込み処理が終了する直前にFINT命令を実行する」とユーザーズ・マニュアルの説明にあるが、

- (1) 入出力命令割り込みとFPO命令割り込みにFINT命令は必要?
- (2) FINT命令を割り込み処理が終了する直前以外の場合に実行してもよい?

A.2.4

- (1) FINT命令は不要です。FINT命令は内部の割り込みコントローラに対する割り込み処理終了の合図となる命令ですので、割り込みコントローラより優先順位制御を受ける割り込み処理からの復帰の際に実行する必要があります。
- (2) FINT命令は必ず規定のRETI (RETRBI) 命令の直前で実行するようにしてください。

V25, V35ファミリには、割り込み優先順位を管理するための特殊機能レジスタがあり、FINT命令は割り込みコントローラ内の割り込みプライオリティ・レジスタ (ISPR) のセットされているビットのうち最下位 (最高優先順位) の1ビットをリセットします。したがって、FINT命令が実行されると、処理中の割り込みと同レベルまたはより低レベルの割り込みが受け付けられるようになります (ただし、FINT命令とその次の命令の間では、割り込みは受け付けられません)。

FINT命令を実行しないと、優先順位制御が正確に行えません。

Q.2.5

割り込み要求制御レジスタで、ビット7 (IF) をソフトウェアによって1にセットしたときのV25, V35ファミリの動作 (EI状態) はどうなる?

A.2.5

ハードウェアによる割り込み要求発生時と同様に割り込み (あるいはマクロ・サービス) が発生します。

Q.2.6

マクロ・サービス機能は、割り込み要求によって特殊機能レジスタ領域とメモリ空間の間でデータ転送を行う機能だが、

- (1) マクロ・サービス機能でブロック転送はできる？
- (2) マクロ・サービス機能で、特殊機能レジスタを用いずI/O←メモリ間の転送をできる？
- (3) 特殊機能レジスタ←メモリ間のデータ転送の際、アドレスおよびバス制御信号は外部に出力される？
- (4) マクロ・サービス機能の応用例はどのようなものがある？
- (5) 多くのタスク処理をする場合の効率のよいマクロ・サービス・チャンネルの使用法はある？
- (6) MSC（マクロ・サービスで行う転送数）が0のとき、割り込み発生はどこへ行く？

A.2.6

- (1) マクロ・サービス機能は、1回の割り込み要求により1回のデータ転送（演算処理を含む）を行うもので、ブロック転送はできません。ただし、複数回の要求により、ブロック・データの処理することは可能です。
- (2) I/O←メモリ間の転送はDMAを使用します。マクロ・サービスは特殊機能レジスタとメモリ間のデータ転送に用います。
- (3) メモリに対するアドレスおよびバス制御信号は、外部メモリをアクセスする場合に出力されます。ただし、特殊機能レジスタに対するアドレス、バス制御信号は出力されません。
- (4) 応用例を以下に示します。
 - ・一定時間間隔でポートからのデータを取り込み、そのデータをサーチする。
 - ・一定時間間隔でポートへデータを出力する。
 - ・外部からの要求によって、上記のインターバル時間を変更する。
- (5) マクロ・サービスは優先順位制御により保留されている必要性がほとんどないため、マクロ・サービス・チャンネルは0から使用し、レジスタ・バンクは7から使用するのが一般に効率的です。
- (6) MSCが0の場合は、割り込み要求制御レジスタにある MS/\overline{INT} ビットが0となります。このビットが0であると、ENCSCビットによりベクタ割り込みまたはレジスタ・バンク切り替えによる割り込みが発生します。このとき、ENCSCビットが0でベクタ割り込みを使用し、1でレジスタ・バンク切り替え機能を使用します。

Q.2.7

INTAK サイクルにウエイト・ステート挿入はできる？

A.2.7

INTAK サイクルにウエイト・ステートを挿入することはできません。

V25, V35ファミリでは、外付け割り込みコントローラはあくまで付加機能（拡張用）として位置付けており、内蔵のウエイト・コントローラには割り込みアクノリッジ・サイクルへのウエイト・ステート挿入の機能はありません。

Q.2.8

V25ファミリで1ワードを読み出したり、書き込んだりする場合、下位バイトと上位バイトに分けて実行するが、下位バイトの読み出しと上位バイトの読み出しの間で割り込み要求がきたときはその割り込みを受け付ける？

A.2.8

1命令の実行終了までは、割り込み要求を受け付けません。

C.3 バス制御

Q.3.1

V35ファミリのメモリ・バンク構成について

- (1) メモリ・バンクの選択に \overline{UBE} 端子と \overline{MSTB} 端子でチップ・セレクトするのはなぜ？ \overline{MSTB} 端子が下位アドレスの読み出しをするのに、 \overline{UBE} 端子が必要な理由は？
- (2) リード・サイクルでは、A0信号および \overline{UBE} 信号は必要か？
- (3) メモリ・サイクルのアドレス時分割出力の2回目のバス・サイクルで、A18ビットの物理アドレスが出力されるのはなぜ？
- (4) アドレスは上位ビット、下位ビットのどちらが先に出力される？
- (5) プログラム・フェッチは、バイト単位でもできる？

A.3.1

- (1) \overline{UBE} 端子は、A18端子とマルチプレクスされており、 \overline{MSTB} 端子がアクティブとなるタイミングでアクティブとなります。このため、上位メモリ・バンクのチップ・セレクトは、 \overline{UBE} 端子と \overline{MSTB} 端子をANDで作成します。

\overline{UBE} 端子は、奇数アドレスに対するワード単位での書き込み動作（バイト単位の読み出し動作が2回起こります）およびバイト単位での書き込み動作時に、メモリ上位バンクあるいはメモリ下位バンクの一方のみをアクセスする場合に必要となります。

- (2) リード・サイクルでは、A0信号および \overline{UBE} 信号は特に必要ではありません。
- (3) A18ビットを出力するのは、ビット数制限の中でできるだけアドレスを早く出力するためです。
- (4) A9-A19ビット、A0ビットが先に出力されます。
- (5) プログラム・フェッチは、信号的には必ずワード単位でアクセスします。奇数アドレス時にはそのうちの上位の1バイトのみを取り込みます（下位バイトは無効となります）。

Q.3.2

内蔵RAM領域、内蔵ROM領域のアクセスについて

- (1) 内蔵RAM領域のアクセス時、バス・サイクル中にウェイト・コントロール・レジスタ (WTC) によって、ウェイト・ステートを挿入することはできる？
- (2) V25, V35の内蔵ROM領域に対して、メモリ・アクセスを行った場合、外部にバス・アドレス制御信号は出力される？

A.3.2

- (1) 内蔵RAM領域に対するアクセスにおいては、プログラマブル・ウェイトの設定に影響を受けず常にノー・ウェイトでアクセスします。V25, V35の内蔵ROMも同様です。
- (2) 内蔵ROM領域に対してプログラム・フェッチあるいはデータ・アクセスを行った場合、外部に対してバス・アドレス制御信号を出力しません。内蔵ROM領域をフェッチするときは内部の専用バスを使用しますので、外部バスのアドレスは前にアクセスしていたサイクルのアドレスを保持したままとなります。

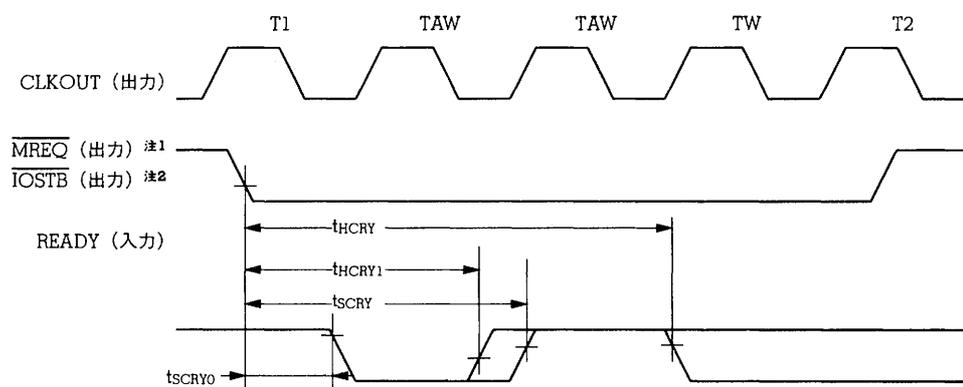
Q.3.3

READY信号の入力について

- (1) ウェイト・ステートが0ステート、1ステート、2ステート固定のとき、READY信号が“L”になると、CPUは異常動作する？
- (2) V25で、TAWを2ステート挿入し、READY端子を“L”にしてTWを1ステート挿入する場合のタイミングの規定は？
- (3) READY信号の入力を非同期(t_{SCRY} , t_{HCRY} の条件を満たさない場合)とした場合、CPUは暴走する？
- (4) 最初、READY信号をインアクティブ(“L”)とし、バス・サイクルを抜けるときだけにREADY信号をアクティブ(“H”)とすることはできる？

A.3.3

- (1) 0ステート、1ステート、2ステート固定のとき、READY信号の入力は無視します。よって、CPUは、正常動作します。
- (2) V25のREADY信号の入力については、 t_{SCRY} と t_{HCRY} の規定を満たしてください。



注1. メモリ・サイクルの場合

注2. I/Oサイクルの場合

- (3) 非同期入力とした場合は、ウェイト・ステートが入るか入らないかのいずれかの状態となり、暴走などは起こりません。
- (4) できます。最初、READY信号をインアクティブ(“L”)とし、バス・サイクルを抜けるときだけにREADY信号をアクティブ(“H”)としても問題は起こりません。

Q.3.4

メモリ・サイクルにおいて

- (1) V25ファミリでの、 $\overline{\text{MREQ}}$ 信号と $\overline{\text{MSTB}}$ 信号の違いは？
- (2) CPUが内部のメモリをアクセスしたとき、 $\overline{\text{MREQ}}$ 信号、 $\overline{\text{MSTB}}$ 信号はインアクティブ状態となる？
- (3) V35ファミリで、メモリ・リード・タイミングでは $\overline{\text{MSTB}}$ 信号はウエイト挿入により伸びるが、メモリ・ライト・サイクルでは $\overline{\text{MSTB}}$ 信号がウエイトで伸びないのはなぜ？

A.3.4

- (1) V25ファミリとメモリをインタフェースする場合、アドレスをデコードしてメモリのチップ・セレクト信号を作ります。このとき、 $\overline{\text{MREQ}}$ 信号はメモリの $\overline{\text{OE}}$ を制御すると考えてください。
 $\overline{\text{MSTB}}$ 信号は、DRAMとのインタフェースに使用します。 $\overline{\text{MSTB}}$ 信号でDRAMの $\overline{\text{CAS}}$ 信号を制御します。DRAMインタフェース以外には特に使用する必要はありません。
- (2) CPUが内部のメモリをアクセスした場合、 $\overline{\text{MREQ}}$ 信号、 $\overline{\text{MSTB}}$ 信号はインアクティブ状態("H")となります。
- (3) $\overline{\text{MSTB}}$ 信号はライト・サイクルではウエイトの挿入によらず、その幅は一定です。V35ファミリではDRAMをインタフェースする場合、 $\overline{\text{MSTB}}$ 信号でメモリの入力信号である $\overline{\text{CAS}}$ 信号を作ります。出力の遅いI/OからメモリへDMA転送する場合、DRAMは $\overline{\text{CAS}}$ 信号の立ち下がり($\overline{\text{MSTB}}$ 信号の立ち下がり)でデータを書き込むために $\overline{\text{MSTB}}$ 信号を遅らせています。

Q.3.5

DMAサイクルからCPUのバス・サイクル(フェッチやデータ・アクセス)へ移るとき、アイドル・サイクルは入る？

A.3.5

アイドル・サイクルは入りません。

DMAサイクルが終了した時点でプリフェッチがすでにあった場合、アイドル・サイクルなしで次のバス・サイクル(プリフェッチ)に移ります。

Q.3.6

V25ファミリのユーザーズ・マニュアルでは、I/Oリード/ライト・サイクルの図においてCLKOUT信号の立ち下がりデータを読み込みや書き込みを行っているが、設計時にユーザーズ・マニュアルの図と同じにする必要はあるか？

A 3 6

I/Oリード/ライト・サイクルのタイミングについては、データ・シートにAC特性を記載していますので参照してください。設計時にCLKOUT信号を使用する必要はありません。I/Oリード/ライト・サイクルのタイミングは、 $\overline{\text{IOSTB}}$ 信号によって生成します。

Q.3.7

V25ファミリのメモリ・リード・タイミングにおいて

- (1) t_{DADR} , t_{DMRD} , t_{DMSD} はともに満たす必要があるか？
- (2) $\overline{\text{MREQ}}$ 信号の立ち下がりからデータ遅延時間 t_{DMRD} 以内のスピードが必要か？
- (3) $\overline{\text{MREQ}}$ 信号の立ち上がり (t_{WMRL}) でデータを読み込むか？

A.3.7

- (1) t_{DADR} , t_{DMRD} , t_{DMSD} はともに満たす必要があります (Q.1.2参照)。
- (2) t_{DMRD} 以内のスピードは必要です。
- (3) $\overline{\text{MREQ}}$ 信号の立ち上がりでデータを読み込みます。

Q.3.8

V25ファミリのメモリおよびI/Oのリード/ライト・サイクルのタイミングで、 t_{DAIS} 、 t_{DAMR} はウェイトを挿入すると伸びる？

A.3.8

t_{DAIS} 、 t_{DAMR} は、ウェイトを挿入しても伸びません。

備考 ウェイトによりAC特性が変化する場合、規定はウェイト数 n の関数で表現されます。

C.4 DMAコントローラ

Q.4.1

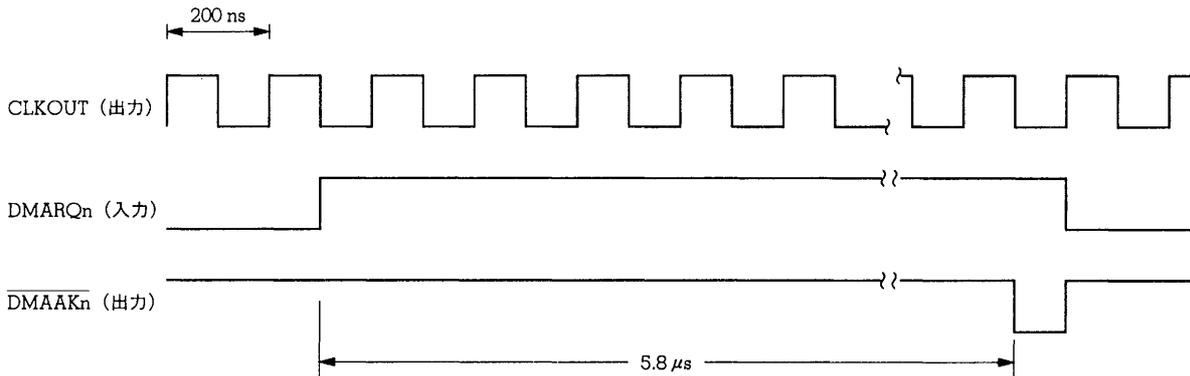
V25, V25SのDMA転送タイミングで、DMARQ信号が入力されてから $\overline{\text{DMAAK}}$ 信号が出力されるまでの所要時間のMIN. 値は？

A.4.1

V25, V25Sでは、DMARQ信号が入力されてから $\overline{\text{DMAAK}}$ 信号が出力されるまでのMIN. 値は以下のとおりです。Nは実行中の命令の所要クロック数を表します。

- ・ 1転送モード 29+Nクロック
- ・ その他のモード 27+Nクロック

下図のタイミングでは、1転送モード（ノー・ウエイト時）で1クロック200 nsの場合、 $\overline{\text{DMAAK}}$ 信号が出力されるまで $29 \times 200 \text{ ns} = 5.8 \mu\text{s}$ が必要となっています。



(バス・ホールド機能, リフレッシュ機能は使用しない状態)

Q.4.2

DMAモード・レジスタについて

- (1) DMARQ信号の入力時にEDMAビットが0であった場合、DMA要求は受け付けられる？
- (2) レジスタ設定に対するDMARQ信号の入力に規定はある？
- (3) DMA転送を一時停止させる制御は、EDMAビットで可能？

A.4.2

- (1) DMAモード・レジスタのEDMAビットが0であった場合、DMA要求（DMARQ信号の入力）は無視されます。
- (2) レジスタ設定に対するDMARQ信号の入力遅延時間は規定していません。
- (3) 可能です。EDMAビットを0にすることで、DMA転送を一時停止させて、その後EDMAビットを1にすれば、DMA転送を再開できます。

Q.4.3

V25ファミリにおいて、メモリ→I/O間のDMA転送では、I/Oに対するアクセスはどのようにして決定する？

A.4.3

I/Oのアクセスは $\overline{\text{DMAAK}}$ 信号によって行います。

I/O、メモリ間のDMA転送において、その転送先あるいは転送元となるI/Oは、ハードウェアによって固定されている必要があります。I/Oは $\overline{\text{DMAAK}}$ 信号によってDMA応答が来たことを認知します。

Q.4.4

V25, V35とV25+, V35+の違いについて

- (1) DMA転送アドレスの指定方法の違いはある？
- (2) V25+, V35+でのDMA転送において、転送レートの向上の直接の要因はある？

A 4.4

- (1) V25, V35はセグメント指定方式ですが、V25+, V35+はリニア・アドレス指定方式です。
- (2) 転送レートの向上の直接の要因として、タイミング・チャート上のダミー・サイクルの有無があります。

V25, V35ではDMA転送サイクルをマイクロプログラム（CPU内部の基本ソフトウェア）により起動しており、要求受け付けおよび転送アドレスの制御をソフトウェアにより行っています。したがって、DMAのバス・サイクルの前後にダミー・サイクルを挿入しています。

一方、V25+, V35+では転送処理を専用ハードウェアにより行っているため、V25, V35に比べて処理が瞬時に行われますので、ダミー・サイクルは挿入されません。

以上の違いにより、V25+, V35+ではDMAの応答性、転送レートが飛躍的に向上しています。

Q.4.5

DMAの1回の転送バイト数は指定できる？

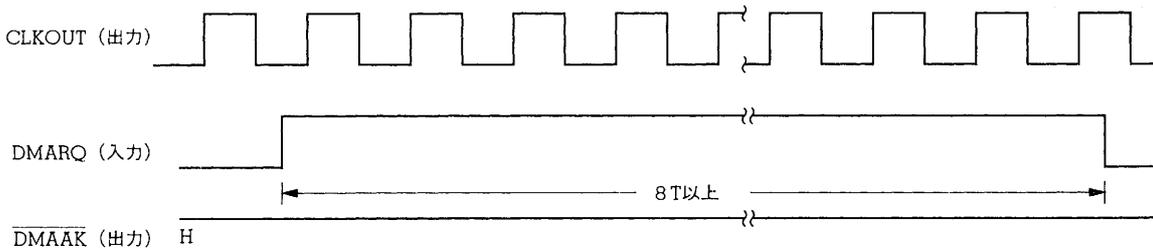
A 4.5

ターミナル・カウンタ (TC) の設定により、DMA転送のバイト転送数およびワード転送数を指定できます。バイト転送時に64 Kバイト、ワード転送時に128 Kバイトの連続転送が可能です。

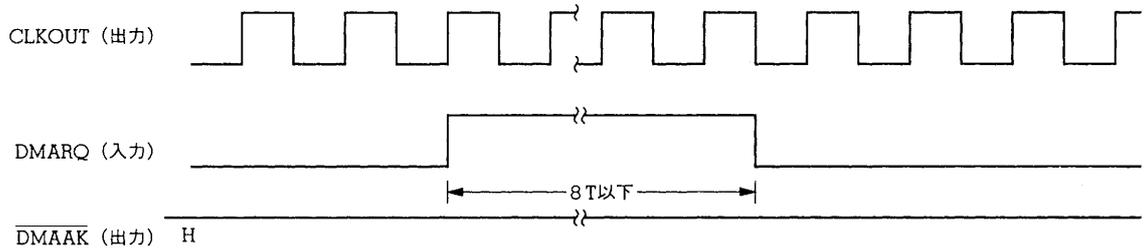
Q.4.6

DMA要求を取り消したときのCPUの動作について、以下のタイミングの場合はどのように動作する(ダイヤモンド・リリース・モード以外)？

- ① DMARQ信号が8 T以上アクティブで、 $\overline{\text{DMAAK}}$ 信号が出力される前にDMA要求を取り消した場合



- ② DMARQ信号が8 T以内アクティブ(ショート・パルスを含む)で、 $\overline{\text{DMAAK}}$ 信号が出力される前にDMA要求を取り消した場合



A.4.6

①の場合、DMARQ信号の立ち上がりでDMARQを受けるので(シングルステップ、バースト、1転送)、DMA要求を受け付けます。

②の場合、CPUは特に異常な動作はしませんが、DMA要求を受け付けないことがあります。

Q.4.7

V25において、DMA転送が終了してからDMA割り込みが発生するまでの時間はどのくらい？

A.4.7

V25は、DMA割り込み要求制御レジスタのDF0、DF1をセットしたのち、ベクタ割り込みでは76クロック (MIN.)、レジスタ・バンク切り替え割り込みでは45クロック (MIN.) 後に、割り込み処理ルーチンの先頭命令を実行します。

ただし、TC=0の最終DMA転送時のバス状態によっては、DMA転送直後で割り込みアクリッジ・サイクルが起動する前にプログラムの命令フェッチを挿入することもあります。このときは、その命令フェッチのクロック分だけ割り込み処理が遅れます。

Q.4.8

バースト・モードのDMA転送において、V25+、V35+のDMA転送中にリフレッシュ要求があるとDMAタイミングはどうなる？

A.4.8

バースト・モードのDMA転送中にリフレッシュ要求があった場合、リフレッシュ・サイクルを挿入します。1回 (1バイト) の転送サイクルを単位として、その間にリフレッシュ・サイクルを挿入します。このとき、前後にアイドル・サイクルは入りません。

Q.4.9

シングルステップ・モードとバースト・モードでは、DMARQ信号を常に“H”とし、DMA処理をDMAモード・レジスタのEDMAビットとTDMAビットで制御するといった、非同期なDMA転送処理は可能？

A 4 9

シングルステップ・モードの場合、EDMAビットで制御可能（停止可能）です。

バースト・モードの場合、DMA転送が一度起動するとEDMAビット、TDMAビットの操作を含めて、いかなる命令処理も行われないため、DMA転送を中断することはできません。

なお、これらのモードの起動は、DMARQ信号入力の立ち上がりエッジあるいはTDMAビットの操作により行います。このときは、EDMAビットは許可状態となっている必要があります。

また、非同期に動作させる方法としては、DMARQ信号を“H”に固定し、PMC2レジスタをポート・モードからコントロール・モードに操作することにより、DMA転送の起動を行う方法もありますが、TDMAビットによる制御を推奨します。

Q.4.10

1 転送モードにおいて

- (1) アドレス・バス上にメモリのアドレスが出力された場合、I/Oアドレスはどこで指定する？
- (2) DMAモード・レジスタのEDMAビットを0から1に変更する間に、DMARQ信号がアクティブであった場合、DMA要求は受け付けられる？
- (3) ユーザーズ・マニュアルでは、1 転送モードの転送対象はメモリ→I/O間の転送しか記載されていないが、I/O→メモリ間の転送の場合はR/ \overline{W} 信号のレベルだけが異なると考えていい？また、I/O→メモリ間の転送では、DMAサービス・チャンネルのアドレス指定は必要？
- (4) V25ファミリでDMA要求を受け付けるためのDMARQ信号のパルス幅のMIN. 値は？

A.4.10

- (1) 1 転送モードのメモリ→I/O間転送の場合、I/Oアドレスは出力されません。
DMA転送を行うとき、DMA要求を行っているI/Oはハードウェアで決定しています。このため、 \overline{DMAAK} 信号をI/Oのチップ・セレクト信号として使用すれば、I/Oを選択できます。
- (2) DMA要求は受け付けられません。1 転送モードでは、EDMAビットが0のときは、DMA要求を無視します。
- (3) 1 転送モードで、メモリ→I/O間の転送とI/O→メモリ間の転送ではR/ \overline{W} 信号の出力レベルが異なります。
また、I/Oとメモリ間の転送においては、I/Oのアドレス指定はソフトウェア的にはできませんので、DMAサービス・チャンネルの設定は意味を持ちません。
- (4) AC特性の t_{WIQH} で規定しています。

Q.4.11

V25+, V35+のダイヤモンド・リリース・モードについて

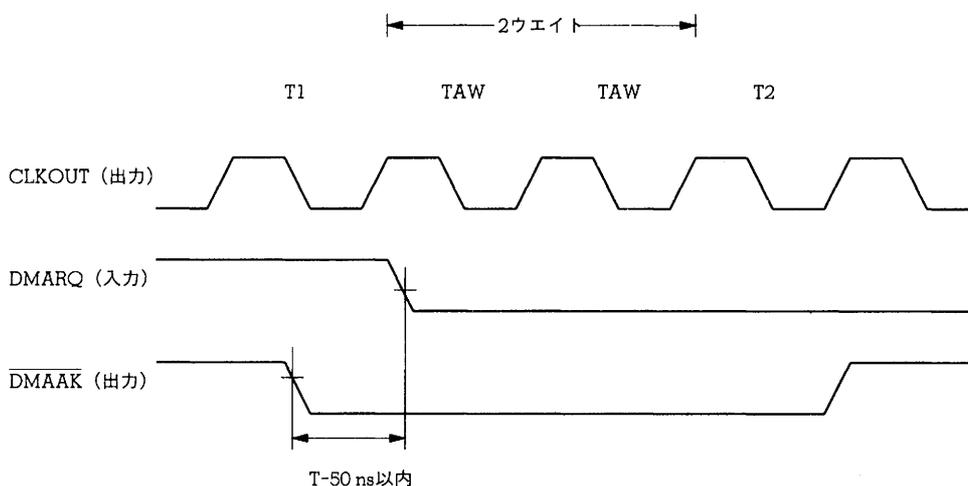
- (1) DMARQ信号による制御で1バイトずつ連続転送するとき、DMA転送サイクル以外の期間は、CPUのバス・サイクルは動作する？
- (2) DMARQ信号による制御を用いて、DMA転送を行うためには、どのような制御を行う？

A.4.11

- (1) V25+, V35+ではDMA転送を専用ハードウェアで処理しますので、DMA転送サイクル以外の期間は、CPUのバス・サイクルが動作します。
- (2) デイモンド・リリース・モードでDMA転送を行うためには、DMARQ信号による停止制御を行うことが必要です。停止制御では、DMA転送サイクルのウェイト数が2ステート以上であることと、転送停止前の最終サイクルの $\overline{\text{DMAAK}}$ 信号の立ち下がり後、 $(T-50)$ ns (t_{SDADQ}) 以内にDMARQ信号を立ち下げなければなりません（下図を参照）。

したがって、最終DMA転送サイクルで確実に転送停止させるために、 $\overline{\text{DMAAK}}$ 信号の立ち下がり後、 $(T-50)$ nsを越えるDMARQ信号をマスクします。

また、確実に1回以上のDMAを起動するには、 $\overline{\text{DMAAK}}$ 信号の立ち下がりまでDMARQ信号を“H”の状態に維持してください。



C.5 クロック発生回路

Q.5.1

プロセッサ・コントロール・レジスタ (PRC) について

- (1) PCK0, PCK1の各ビットにより発振周波数の分周比を変える場合, CPUの動作に影響があるか?
- (2) PRCのRAMENビットによって内部RAMのアクセスを禁止した場合, 同一アドレスの外部メモリにアクセスはできる? また, このときレジスタ・バンクは使用できる?

A.5.1

- (1) 発振周波数の分周比をPCK0, PCK1の各ビットにより変更しても, CPUの動作に問題は起こりません。ソフトウェアで周波数の切り替えをすることはできます。
- (2) RAMENビットを0にセットして内部RAMのアクセスを禁止した場合, 外部メモリがアクセスされません。同一アドレスでも構いません。
レジスタ・バンクは使用できます。メモリとしてのデータ・アクセスはできませんが, レジスタとして参照することは可能です。

C.6 タイマ・ユニット

Q.6.1

タイマ・ユニットをインターバル・タイマ・モードに設定した場合

- (1) タイマ0, タイマ1は16ビット・フル・カウント固定?
- (2) タイマ0のTM0レジスタに設定した周期を方形波にして, TOUT端子に出力することは可能?

A.6.1

- (1) タイマ0, タイマ1は, フル・カウント固定ではありません。カウント値をMD0, MD1の各レジスタにセットすることにより, 任意のカウント動作が可能です。
- (2) インターバル・タイマ・モード時のタイマ0では, TM0レジスタはタイマ (カウンタ) として働き, 周期はMD0ビットに設定されます。この周期で反転する方形波をTOUT端子に出力できます。なお, デューティ50%以外の方角波は出力できません。

Q.6.2

V25で, インターバル・タイマ・モードでシステム・クロック周波数 (f_{CLK}) が8 MHzの場合, タイマ・コントロール・レジスタのTCLKnビット=0を使用するとき, 20 msのインターバル・タイマ値を設定するためにMDnレジスタにセットする値は?

A.6.2

$f_{CLK}=8\text{ MHz}$, TCLKnビット=0のとき, 20 msのインターバル・タイマ値を設定するには, MDnレジスタの値を682 BH (26667) としてください。

計算例

$$TCLKn = 0 : \text{カウント} \cdot \text{クロック} f_{CLK} / 6$$

$$(8\text{ MHzで使用する } 8\text{ MHz} / 6 \approx 1.33\text{ MHz (クロック周期: } 0.75\ \mu\text{s}))$$

20 msのインターバル・タイマは,

$$\frac{20\text{ ms}}{0.75\ \mu\text{s}} \approx 26667$$

カウントします。このため, MDn=682 BH (26667) を設定します。

C.7 シリアル・インタフェース

Q.7.1

ボー・レートが750 kbps以上のデータ送信／受信で、データ間に1クロック以上のアイドル時間を挿入するということは、1ビットずつに1クロックを入れるということ？

A.7.1

データ間に1クロック以上のアイドル時間を挿入するということは、1バイトのデータ転送ごとに1クロック以上（シフト・クロック）のアイドル時間が必要ということです。アシンクロナス・モードの場合、レベルが“H”のアイドル・サイクルが1クロック以上必要です。アシンクロナス・モードのとき、V25, V35ファミリの受信をストップ・ビット=1とし、送信側のシリアル・コントローラをストップ・ビット=2にしてインタフェースしてください。

Q.7.2

V25, V35ファミリと別のマイクロコンピュータをシリアル・インタフェースする場合、シリアル・ライジへのノイズの混入などにより、データが1ビットずれて復帰しない現象があるときの対処法はある？

A.7.2

ノイズによるデータのビットのずれが起こった場合、シリアル・レジスタをクリアし、送受信を再びやり直す必要があります。

シリアル・レジスタをクリアするには、次のどちらかの操作を行います。

①SCC0レジスタへの書き込み

②SCM0レジスタのMD0ビットの変更

(I/Oインタフェース・モード→アシンクロナス・モード→I/Oインタフェース・モード)

Q.7.3

V25ファミリのシリアル・インタフェースで、アシンクロナス・モードを使用して送信を行う場合、

- (1) プログラムによるマクロ・サービス起動後、実際にスタート・ビットが出力されるまでの時間はどのくらい？
- (2) ボー・レートを4800 bpsとしたとき、プログラムで送信バッファにデータをセットしてから、実際にスタート・ビットが出力されるまでの時間はどのくらい？

A.7.3

- (1) 内蔵RAM許可、バイト転送の場合の概略時間の求め方を以下に示します。
 - ①まず、IFフラグは、IFフラグをセットする命令の次の命令実行開始のタイミングでセットされます。
 - ②マクロ・サービス（ノーマル・モード）の場合
11クロック後（MIN.）にマクロ・サービスが実行されます。さらに、メモリ→送信バッファ（TxBn）へデータ転送を実行するとき、 $(24+W)$ クロックかかります（Wはウエイト数）。
 - ③マクロ・サービス（キャラクタ・サーチ・モード）の場合
11クロック後にマクロ・サービスが実行されます。②の場合と同じようにメモリ→送信バッファ（TxBn）のデータ転送の実行に $(27+W)$ クロックかかります。
 - ④マクロ・サービスが終了し、次の命令実行開始のタイミングで、実際に②または③で行った送信バッファ（TxBn）へのデータ転送が実行されます。
 - ⑤スタート・ビットの出力は、④の終了後の次のシフト・クロックの立ち上がりで行われます。概略時間は、①と④が無視されるとすれば、②+⑤または③+⑤の時間で求められます。
- (2) 送信バッファ（TxBn）への書き込み命令を実行すると、次の命令開始のタイミングで、送信バッファへデータを転送します。シリアル・インタフェースは、命令実行（システム・クロック）と独立に動作しており、スタート・ビットは、 $1/f_{CLK}$ (s) ~ $1/4800$ (s) 後に出力されます。

Q.7.4

I/Oインタフェース・モードの送信で、送信クロックを外部から入力することはできる？

A.7.4

送信クロックを外部から入力することはできません。

Q.7.5

V35ファミリにおいて、I/Oインタフェース・モードでマクロ・サービスの応答時間（最大転送レート）は？

A.7.5

シリアル・インタフェースに使用するマクロ・サービスの応答時間（要求発生→転送実行開始）は次のとおりです。Nは要求受け付け時に実行中の命令の残りクロック数です。

送信、内蔵RAM許可の場合

- ・ノーマル・モード… $36+N \sim 52+N$ クロック
- ・キャラクタ・サーチ・モード… $39+N \sim 55+N$ クロック

したがって、送信バッファとのデータ交換のスピードは、この応答時間により制限されます。

受信の場合も同様です（ただし、クロック数が異なります）。

ただし、バス・ホールド、リフレッシュ・サイクル等が含まれる場合、この範囲ではありません。

Q.7.6

シリアル・インタフェース割り込み要求において

- (1) 割り込み要求制御レジスタのビット7 (SEFn, SRFn, STFn) を0にクリアする条件は？
- (2) SRFnビットが立つタイミングは？
- (3) シリアル・インタフェース割り込みはレジスタ・バンク切り替え応答が可能？
- (4) マクロ・サービスで受信完了割り込み処理中に受信エラー割り込みが発生した場合、それまで実行中の割り込み処理は保留になる？

A.7.6

- (1) SEFn, SRFn, STFnビットは、各々の割り込みアクノリッジ・サイクル中に自動的に0にクリアされます。
- (2) SRFnビットが立つタイミングは、受信バッファ (RxBn) に、シフト・レジスタからデータがセットされた時点です。
- (3) シリアル・インタフェース割り込みは、レジスタ・バンク切り替え応答が可能です。
- (4) 受信エラー割り込みは、受信完了割り込みや送信完了割り込みよりも優先順位が高く、受信エラー割り込み処理中は、受信完了割り込みは保留となります。

したがって、受信割り込みをマクロ・サービスで処理する場合、この保留状態を解除 (FINT命令の実行) したのち、マクロ・サービスが実行され、その時点で受信バンクにあるデータを転送します。

C.8 スタンバイ機能

Q.8.1

STOPモードをNMIにより解除した場合の発振安定時間は？

A.8.1

発振安定時間は30 msです。

Q.8.2

STOPモードをNMIにより解除した場合のCPUが動作するまでの V_{DD} 消費電流は？

A.8.2

V_{DD} 消費電流はHALTモードと同じになります。

C.9 リセット機能

Q.9.1

システム・リセットがかかったときの V_{DD} 消費電流は？

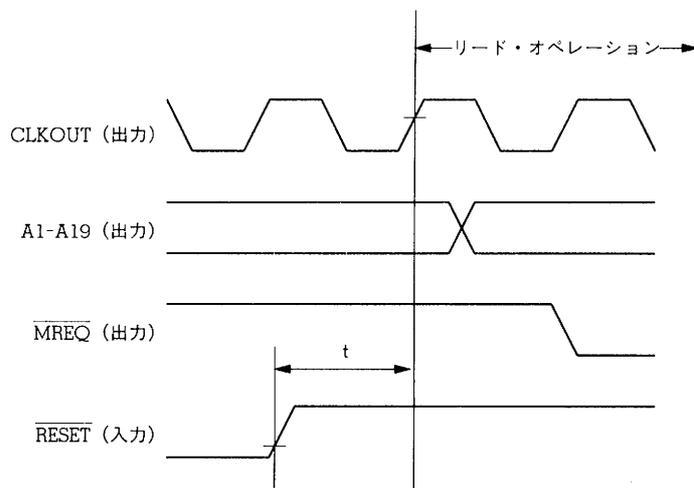
A.9.1

リセット中の V_{DD} 消費電流は、通常動作モードと同じです。

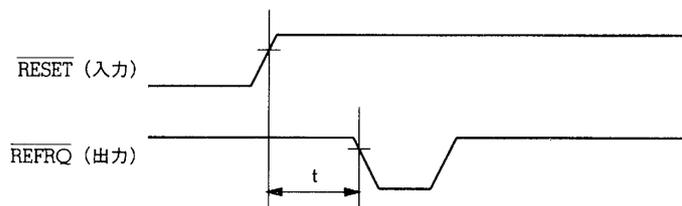
Q.9.2

V35ファミリにおいて、 $\overline{\text{RESET}}$ 信号のレベルを“L”から“H”に立ち上げて、システム・リセットを解除するとき

- (1) CPUが動作するまでの時間は？



- (2) $\overline{\text{REFRQ}}$ 信号が出力されるまでの時間は？



A.9.2

- (1) $\overline{\text{RESET}}$ 信号の立ち上げから、21クロック後にスタート・アドレス (FFFF0H) へ分岐します。
- (2) リフレッシュ・モード・レジスタ (RFM) が連続的に $\overline{\text{RESET}}$ 信号入力の状態にあるとき、システム・リセット解除後、 $2^4/f_{\text{CLK}}$ (秒) 後に最初の $\overline{\text{REFRQ}}$ 信号が出力されます。

C.10 その他

Q.10.1

V25とV25+, またV35とV35+をプログラムで区別する方法はある？

A.10.1

特殊機能レジスタの××F6BH番地、または××F7BH番地の内容をリセット直後に読み出せば判別できます。下表を参照してください。

	F6BH	F7BH
V25, V35	00H (SCE0)	00H (SCE1)
V25+, V35+	60H (SCS0)	60H (SCS1)

ただし、先頭ビット（ビット7）は、RxD0、1端子の状態をそのままモニタしているため、実際に読み出されている値はRxD0、1端子の状態によります。

V25とV25+, V35とV35+の機能上の違いについては、V25+, V35+のユーザーズ・マニュアルを参照してください。

Q.10.2

V25とV25S, またはV35とV35Sの違いは？

A.10.2

V25S, V35Sには命令コード変換機能のための、NモードとSモードという2種類の動作モードがあります。

Nモードは、V25, V35の命令コードを実行するモードです。Sモードは、ユーザ定義の命令コードを実行するモードです。

その他の違いについては、V25S, V35Sのユーザーズ・マニュアルを参照してください。

Q.10.3

IC端子を外部でプルアップ抵抗を介して“H”に固定するのはなぜ？

A.10.3

IC端子は内部的に接続されており、プルアップ抵抗はサージなどの外部入力に対して端子を保護するために推奨しているものです。

Q.10.4

命令実行時間（クロック数）の“EA+6+T”のEAとTは何を表している？

A.10.4

EAは、メモリのアドレッシング・モードにより変化するクロック数です。Tはウェイト・ステート数です。

Q. 10.5

転送命令MOVについて、「MOV mem reg」の場合に内蔵RAMアクセス禁止のクロック数が“EA+2”でウェイトが入らないのはなぜ？

A. 10.5

V25, V35ファミリは3段パイプライン処理をして、以下のようにになっています。

PAU → EXU → BCU
(実効アドレス計算) (演算実行) (バス駆動)

「MOV mem reg」命令の実行クロック数はこのうちPAU, EXUでの所要クロック数のみで、演算処理が終了するとデータをBCUに送って次の命令実行に移るからです。

付録D レジスタ索引 (アルファベット順)

略号	名称	ページ
BRG0, 1	ポーレート・ジェネレータ・レジスタ 0, 1	58, 59, 224
DAR0H	デスティネーション・アドレス・ポインタ 0 (上位)	60, 171
DAR1H	デスティネーション・アドレス・ポインタ 1 (上位)	60, 171
DAR0L	デスティネーション・アドレス・ポインタ 0 (下位)	60, 171
DAR1L	デスティネーション・アドレス・ポインタ 1 (下位)	60, 171
DAR0M	デスティネーション・アドレス・ポインタ 0 (中位)	60, 171
DAR1M	デスティネーション・アドレス・ポインタ 1 (中位)	60, 171
DIC0, 1	DMA割り込み要求制御レジスタ 0, 1	60, 168
DMAC0, 1	DMAコントロール・レジスタ 0, 1	60, 167
DMAM0, 1	DMAモード・レジスタ 0, 1	60, 165
EMSO-2	外部割り込みマクロ・サービス制御レジスタ 0-2	58, 97
EXIC0-2	外部割り込み要求制御レジスタ 0-2	58, 97
FLAG	ユーザ・フラグ・レジスタ	50, 60
IDB	内部データ領域ベース・レジスタ	57, 60
INTM	外部割り込みモード・レジスタ	58, 96
IRQS	割り込み要因レジスタ	58, 92
ISPR	割り込みプライオリティ・レジスタ	58, 94
MDO, 1	モジュロ/タイマ・レジスタ 0, 1	59, 199
MSC	マクロ・サービス・カウンタ	86
MSP	マクロ・サービス・ポインタ	86
MSS	マクロ・サービス・セグメント	86
PO-2	ポート 0-2	58, 183
PC	プログラム・カウンタ	46
PM0-2	ポート 0-2 モード・レジスタ	58, 186
PMCO-2	ポート 0-2 モード・コントロール・レジスタ	58, 186
PMT	ポート T モード・レジスタ	58, 192
PRC	プロセッサ・コントロール・レジスタ	60, 198
PSW	プログラム・ステータス・ワード	46
PT	ポート T	58, 191
RFM	リフレッシュ・モード・レジスタ	60, 142
RxB0, 1	受信バッファ・レジスタ 0, 1	58, 59, 213
SAR0H	ソース・アドレス・ポインタ 0 (上位)	60, 169
SAR1H	ソース・アドレス・ポインタ 1 (上位)	60, 169

付

略号	名称	ページ
SAR0L	ソース・アドレス・ポインタ 0 (下位)	60, 169
SAR1L	ソース・アドレス・ポインタ 1 (下位)	60, 169
SAR0M	ソース・アドレス・ポインタ 0 (中位)	60, 169
SAR1M	ソース・アドレス・ポインタ 1 (中位)	60, 169
SCC0, 1	シリアル制御レジスタ 0, 1	58, 59, 225
SCHR	サーチ・キャラクタ	86
SCM0, 1	シリアル・モード・レジスタ 0, 1	58, 59, 220
SCSO, 1	シリアル・ステータス・レジスタ 0, 1	58, 59, 227
SEIC0, 1	シリアル・エラー割り込み要求制御レジスタ 0, 1	58, 59, 229
SFRP	特殊機能レジスタ・ポインタ	86
SRIC0, 1	シリアル受信割り込み要求制御レジスタ 0, 1	58, 59, 229
SRMS0, 1	シリアル受信マクロ・サービス制御レジスタ 0, 1	58, 59, 230
STBC	スタンバイ・コントロール・レジスタ	60, 231
STIC0, 1	シリアル送信割り込み要求制御レジスタ 0, 1	58, 59, 229
STMS0, 1	シリアル送信マクロ・サービス制御レジスタ 0, 1	58, 59, 230
TBIC	タイム・ベース割り込み要求制御レジスタ	60, 212
TC0H	ターミナル・カウンタ 0 (上位)	60, 173
TC1H	ターミナル・カウンタ 1 (上位)	60, 173
TC0L	ターミナル・カウンタ 0 (下位)	60, 173
TC1L	ターミナル・カウンタ 1 (下位)	60, 173
TM0, 1	タイマ・レジスタ 0, 1	59, 199
TMC0, 1	タイマ・コントロール・レジスタ 0, 1	59, 201
TMIC0-2	タイマ・ユニット割り込み要求制御レジスタ 0-2	59, 208
TMMS0-2	タイマ・ユニット・マクロ・サービス制御レジスタ 0-2	59, 208
TxB0, 1	送信バッファ・レジスタ 0, 1	58, 59, 213
WTC	ウェイト・コントロール・レジスタ	60, 123

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] V25+, V35+ ユーザーズ・マニュアル ハードウェア編
(IEU-706E (第6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。

NEC 半導体インフォメーションセンター

FAX : (044)548-7900

保守 / 廃止

— お問い合わせは、最寄りのNECへ —

【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	〒108-01 東京都港区芝五丁目7番1号 (NEC本社ビル)	東京 (03)3454-1111 (大代表)
中部支社 半導体販売部	〒460 名古屋市中区栄四丁目14番5号 (松下中日ビル)	名古屋 (052)242-2755
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 大阪市中央区城見一丁目4番24号 (NEC関西ビル)	大阪 (06) 945-3178 大阪 (06) 945-3200 大阪 (06) 945-3208
北海道支社 東北支社 岩手支店 山形支店 郡山支店 いわき支店 長岡支店 土浦支店 水戸支店 神奈川支社 群馬支店 太田支店 宇都宮支店	札幌 (011)231-0161 仙台 (022)261-5511 盛岡 (0196)51-4344 山形 (0236)23-5511 郡山 (0249)23-5511 いわき (0246)21-5511 長岡 (0258)36-2155 土浦 (0298)23-6161 水戸 (0292)26-1717 横浜 (045)324-5511 高崎 (0273)26-1255 太田 (0276)46-4011 宇都宮 (0286)21-2281	小山支店 (0285)24-5011 長野支社 (0262)35-1444 松本支店 (0263)35-1666 諏訪支店 (0266)53-5350 甲府支店 (0552)24-4141 埼玉支社 (048)641-1411 立川支社 (0425)26-5981 千葉支社 (043)238-8116 静岡支社 (054)255-2211 浜松支店 (0559)63-4455 浜松支店 (053)452-2711 北陸支店 (0762)23-1621 福井支店 (0776)22-1866
富山支店 三浦支店 京都支社 神戸支社 中国支社 鳥取支店 岡山支店 四国支社 新居浜支店 松山支店 九州支社 北九州支店	富山 (0764)31-8461 津 (0592)25-7341 京都 (075)344-7824 神戸 (078)333-3854 広島 (082)242-5504 鳥取 (0857)27-5311 岡山 (086)225-4455 高松 (0878)36-1200 新居浜 (0897)32-5001 松山 (0899)45-4111 福岡 (092)271-7700 北九州 (093)541-2887	

【本資料に関する技術お問い合わせ先】

半導体ソリューション技術本部 マイクロコンピュータ技術部	〒210 川崎市幸区塚越三丁目484番地	川崎 (044)548-8890	半導体 インフォメーションセンター FAX(044)548-7900 (FAXにてお願い致します)
半導体販売技術本部 東日本販売技術部	〒108-01 東京都港区芝五丁目7番1号 (NEC本社ビル)	東京 (03)3798-9619	
半導体販売技術本部 中部販売技術部	〒460 名古屋市中区栄四丁目14番5号 (松下中日ビル)	名古屋 (052)242-2762	
半導体販売技術本部 西日本販売技術部	〒540 大阪市中央区城見一丁目4番24号 (NEC関西ビル)	大阪 (06) 945-3383	