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い申し上げます。

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ものですが、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て、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がその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

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

RENESAS

ユーザーズ・マニュアル

保守／廃止

V55PI™

16ビット・マイクロプロセッサ

ハードウエア編

μPD70433

資料番号　U10514JJ5V1UM00（第5版）
発行年月　September 1997　N

© NEC Corporation 1991, 1992, 1994

保守／廃止

〔メ　モ〕

保守／廃止

目　次　要　約

第１章　概　　説 … 1

第２章　端子機能 … 11

第３章　CPU機能 … 37

第４章　バス制御機能 … 85

第５章　割り込み機能 … 121

第６章　DMA機能（DMAコントローラ） … 199

第７章　シリアル・インタフェース機能 … 265

第８章　パラレル・インタフェース機能 … 321

第９章　タイマ機能 … 353

第10章　リアルタイム出力ポート機能 … 375

第11章　PWM出力機能 … 389

第12章　ウォッチドッグ・タイマ機能 … 395

第13章　A/Dコンバータ機能 … 399

第14章　ポート機能 … 411

第15章　スタンバイ機能 … 429

第16章　クロック発生機能 … 435

第17章　ソフトウエア・インターバル・タイマ機能 … 441

第18章　コーデック機能 … 443

第19章　リセット機能 … 513

付録A　ニモニック一覧 … 517

付録B　注意事項 … 551

付録C　レジスタ索引 … 561

保守／廃止

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，V30，V20HL，V30HL，V33A，V25，V35，V25+，V40，V50，V40HL，V50HL，V53A，V55PI，Vシリーズは日本電気株式会社の商標です。

保守／廃止

本資料に掲載の応用回路および回路定数は，例示的に示したものであり，量産設計を対象とするものではありません。

○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。

保守／廃止

本版で改訂された主な箇所

箇　所	内　容
p. 4, 5	**1.3　オーダ情報で**μPD70433GD-5BBを削除
p. 35	**2.4(1) ポートの**P60-P63/ANI0-ANI3端子の推奨接続方法を変更
p. 106	**4.8　バス・ホールド機能に**注意を追加
p. 207	**図6-5　ディマンド・リリース・モード転送タイミングの**データのタイミングを変更
p. 395	**図12-1　RESET端子の接続例**を追加
p. 429	**表15-1　HALT/STOPモードの動作状態に**ポートの項目を追加, $\overline{\text{IORD}}$, $\overline{\text{IOWR}}$の項目を変更
p. 493	**18.6.3　MH符号化命令に**注意を追加
p. 495	**18.6.4　MR符号化命令に**注意を追加
p. 554	**B.2(7) シングルステップ・モード使用上の注意事項**を追加
p. 556	**B.2(8) インテリジェントDMAモード-1使用上の注意事項**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。

保守／廃止

はじめに

対象者　このマニュアルは，μPD70433（別名称V55PI）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μPD70433の持つハードウエ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　μPD70433のユーザーズ・マニュアルは，ハードウエア編（このマニュアル）と命令編の２冊に分かれています。

ハードウエア編	命令編
●概説	●概説
●端子機能	●命令の説明
●CPU機能	●V20，V30またはV25，V35に対する追加命令
●内蔵周辺ハードウエア	●キュー操作命令
●スタンバイ機能	●コーデック命令
●リセット機能	

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。なお，このマニュアルでは，μPD70433という製品名をV55PIの名称に統一して説明してあります。

μPD70108，70116（別名称V20™，V30™）をすでに経験しているユーザ

→V55PI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**V55PI　ユーザーズ・マニュアル　命令編の第３章　V20，V30またはV25，V35に対する追加命令**を参照してください。

μPD70320，70330（別名称V25™，V35™）をすでに経験しているユーザ

→**1.2　V55PIとV25との機能比較**でV55PI特有の内蔵周辺ハードウエアを確認してください。

ソフトウエアの違いについては，別冊の**V55PI　ユーザーズ・マニュアル　命令編の第３章　V20，V30またはV25，V35に対する追加命令**を参照してください。

命令機能の詳細を知りたいとき

→別冊の**V55PI　ユーザーズ・マニュアル　命令編**を参照してください。

保守／廃止

一通りV55PIのハードウエア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。

電気的特性を知りたいとき

→別冊の**V55PI　データ・シート**を参照してください。

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…××××または××××B
	10進数…××××
	16進数…××××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　関連資料は暫定版の場合がありますが，この資料では「暫定」の表示をしておりません。あらかじめご了承ください。

- パンフレット（U10244J）
- データ・シート（U11775J）
- ユーザーズ・マニュアル　ハードウエア編（このマニュアル）
- 〃　命令編（U10231J）
- アプリケーション・ノート　ソフトウエア編（IEA-743）
- 〃　ファクシミリ編（IEA-727）
- 〃　ハードウエア編（IEA-750）
- 特殊機能レジスタ活用表（IEM-5592）

保守／廃止

開発ツール関連資料一覧

	資料名		資料番号
ハードウエア	IE-70433-BX　ユーザーズ・マニュアル		U10608J
ソフトウエア	RA70116-I, SP70116-I		
	ユーザーズ・マニュアル	アセンブリ言語編	EEU-861
		操作編	EEU-869

保守／廃止

〔メ　モ〕

保守／廃止

目　　次

第１章　概　　説 … 1

1.1　特　　徴 … 2
1.2　V55PIとV25との機能比較 … 3
1.3　オーダ情報 … 4
1.4　端子接続図（Top View） … 5
1.5　内部ブロック … 8
1.5.1　内部ブロック図 … 8
1.5.2　内部ブロック構成 … 9

第２章　端子機能 … 11

2.1　端子機能一覧 … 11
2.1.1　ポート機能 … 11
2.1.2　ポート以外の機能 … 13
2.1.3　端子状態 … 16
2.2　端子機能の説明 … 19
2.2.1　P00-P07 (Port 0) … 19
2.2.2　P10-P16 (Port 1) … 19
2.2.3　P20-P25 (Port 2) … 20
2.2.4　P30-P36 (Port 3) … 21
2.2.5　P40-P47 (Port 4) … 23
2.2.6　P50-P52 (Port 5) … 24
2.2.7　P60-P63 (Port 6) … 26
2.2.8　P70-P77 (Port 7) … 26
2.2.9　P80, P81 (Port 8) … 27
2.2.10　GND (Ground) … 27
2.2.11　V_{DD} (Power Supply) … 28
2.2.12　AV_{SS} (Analog V_{SS}) … 28
2.2.13　AV_{DD} (Analog V_{DD}) … 28
2.2.14　AV_{REF} (Analog Reference Voltage) … 28
2.2.15　$\overline{RESET}$ (Reset) … 28
2.2.16　X1, X2 (Crystal) … 28
2.2.17　CLKOUT (Clock Out) … 29
2.2.18　$\overline{ASTB}$ (Address Strobe) … 29
2.2.19　$\overline{RD}$ (Memory Read Strobe) … 29
2.2.20　$\overline{WRL}$ (Memory Write Low Strobe) … 29
2.2.21　$\overline{WRH}$ (Memory Write High Strobe) … 29
2.2.22　READY (Ready) … 30
2.2.23　$\overline{DEX}$ (Data Exchange) … 30
2.2.24　$\overline{RAS}$ (Row Address Strobe) … 30
2.2.25　D8/$\overline{D16}$ (Data 8/16) … 31
2.2.26　$\overline{BUSLOCK}$ (Buslock) … 31

保守／廃止

2.2.27 $\overline{\text{POLL}}$ (Poll) … 31
2.2.28 HLDRQ (Hold Request) … 31
2.2.29 $\overline{\text{HLDAK}}$ (Hold Acknowledge) … 31
2.2.30 AD0-AD15 (Address/Data 0-15) … 31
2.2.31 A16-A23 (Address 16-23) … 31
2.2.32 $\overline{\text{IORD}}$ (I/O Read) … 31
2.2.33 $\overline{\text{IOWR}}$ (I/O Write) … 32
2.2.34 $\overline{\text{WDTOUT}}$ (Watchdog Timer Output) … 32
2.2.35 $\overline{\text{DMAAK0}}$, $\overline{\text{DMAAK1}}$ (DMA Acknowledge 0, 1) … 32
2.2.36 $\overline{\text{TCE0}}$, $\overline{\text{TCE1}}$ (Terminal Count End 0, 1) … 32
2.2.37 $\overline{\text{REFRQ}}$ (Refresh Request) … 32
2.3 端子の入出力回路 … 33
2.3.1 入出力回路のタイプ … 33
2.3.2 入出力回路 … 34
2.4 未使用端子の処理 … 35

第3章 CPU機能 … 37

3.1 特 徴 … 37
3.2 レジスタ … 38
3.2.1 レジスタ・バンク … 38
3.2.2 汎用レジスタ (AW, BW, CW, DW) … 40
3.2.3 ポインタ (SP, BP) とインデクス・レジスタ (IX, IY) … 40
3.2.4 セグメント・レジスタ (PS, SS, DS0, DS1) … 41
3.2.5 拡張セグメント・レジスタ (DS2, DS3) … 42
3.2.6 特殊機能レジスタ (SFR) … 44
3.3 プログラム・カウンタ (PC) … 45
3.4 プログラム・ステータス・ワード (PSW) … 45
3.4.1 CY (Carry Flag) … 46
3.4.2 P (Parity Flag) … 47
3.4.3 AC (Auxiliary Flag) … 47
3.4.4 Z (Zero Flag) … 47
3.4.5 S (Sign Flag) … 47
3.4.6 V (Overflow Flag) … 48
3.4.7 $\overline{\text{IBRK}}$ (I/O Break Flag) … 49
3.4.8 BRK (Break Flag) … 49
3.4.9 IE (Interrupt Enable Flag) … 49
3.4.10 DIR (Direction Flag) … 49
3.4.11 RB0-RB3 (Register Bank0-3 Flag) … 49
3.5 メモリ空間 … 50
3.5.1 基本メモリ空間 … 50
3.5.2 拡張メモリ空間 … 51
3.5.3 特殊機能レジスタ領域 … 53
3.5.4 ベクタ・テーブル領域 … 62
3.6 レジスタ・ファイル空間 … 64
3.7 I/O空間 … 66
3.8 論理アドレスと物理アドレス … 67

保守／廃止

3.8.1　セグメント方式　…　67
3.8.2　セグメントの構成　…　69
3.8.3　ダイナミック・リロケーション　…　73
3.9　実効アドレス　…　74
3.10　アドレシング・モード　…　75
3.10.1　命令アドレス　…　75
3.10.2　メモリ・オペランド・アドレス　…　78

第4章　バス制御機能　…　85

4.1　外部バス制御端子　…　85
4.2　外部データ・バス幅選択機能　…　85
4.3　基本バス・サイクル　…　85
4.4　メモリ・ブロック制御機能　…　90
4.4.1　メモリ・ブロック・コントロール・レジスタ　…　90
4.5　ウエイト機能　…　94
4.5.1　アドレス・ウエイトとデータ・ウエイト　…　94
4.5.2　プログラマブル・ウエイト機能　…　94
4.5.3　READY端子によるウエイト制御　…　95
4.5.4　DMA転送サイクルのウエイト　…　98
4.5.5　SFRアクセス時のウエイト　…　98
4.6　$\overline{\text{RAS}}$端子機能　…　99
4.7　リフレッシュ機能　…　101
4.7.1　リフレッシュ・モード・レジスタ（RFM）　…　101
4.7.2　リフレッシュ・サイクルのウエイト制御　…　103
4.7.3　リフレッシュ・アドレス　…　103
4.7.4　リフレッシュ・タイミング　…　103
4.7.5　疑似SRAMの接続　…　105
4.8　バス・ホールド機能　…　106
4.9　バス使用権　…　107
4.10　バス・タイミング　…　107

第5章　割り込み機能　…　121

5.1　特　　徴　…　121
5.2　割り込み要因　…　122
5.2.1　ノンマスカブル割り込み　…　125
5.2.2　マスカブル割り込み　…　125
5.3　優先順位制御　…　128
5.3.1　多重割り込みに関する優先順位制御　…　128
5.3.2　同時に割り込みが発生したときの優先順位制御　…　129
5.3.3　多重割り込みの制御モード選択　…　130
5.4　割り込み応答方式　…　131
5.4.1　ベクタ割り込み　…　132
5.4.2　レジスタ・バンク切り替え機能　…　133
5.4.3　マクロ・サービス機能　…　138
5.5　制御レジスタ　…　172

保守／廃止

5.5.1　割り込みモード・コントロール・レジスタ（IMC）　… 172
5.5.2　割り込みマスク・フラグ・レジスタ（MK0L, MK0H, MK1L, MK1H）　… 174
5.5.3　割り込み要求制御レジスタ（IC9-14, 16-32, 36, 37）　… 175
5.5.4　インサービス・プライオリティ・レジスタ（ISPR）　… 178
5.6　外部割り込み　… 179
5.6.1　外部割り込みモード・レジスタ（INTM0, INTM1）　… 179
5.7　ソフトウエア割り込み　… 180
5.7.1　一般のソフトウエア割り込み　… 180
5.7.2　入出力命令割り込み　… 181
5.7.3　FPO命令割り込み　… 182
5.8　例外トラップ　… 183
5.9　ハードウエア割り込み要求が受け付けられないタイミング　… 183
5.10　ブロック処理命令実行中の割り込み処理　… 184
5.11　コーデック命令実行中の割り込み処理　… 184
5.12　割り込み要求の処理動作例　… 186
5.13　割り込み応答時間　… 194
5.14　注意事項　… 197
5.14.1　マクロ・サービスでSFRアクセスを行う場合　… 197
5.14.2　割り込みモード・コントロール・レジスタの操作　… 197
5.14.3　マクロ・サービス終了時の割り込み要求発生　… 197
5.14.4　割り込みが発生しない場合　… 197
5.14.5　FINT命令についての注意事項　… 198

第6章　DMA機能（DMAコントローラ）　… 199

6.1　特　　徴　… 199
6.2　基本動作　… 200
6.3　構　　成　… 203
6.3.1　チャネル構成　… 203
6.3.2　転送対象　… 203
6.3.3　起動要因　… 204
6.3.4　転送モード　… 207
6.3.5　動作モード　… 209
6.4　動　　作　… 230
6.4.1　起　　動　… 230
6.4.2　停　　止　… 231
6.4.3　中　　断　… 231
6.5　優先順位　… 232
6.5.1　チャネル間の優先順位　… 232
6.5.2　リフレッシュ・サイクルとの優先順位　… 232
6.5.3　他の制御動作との優先順位　… 232
6.6　制御レジスタ　… 233
6.6.1　DMAモード・レジスタ（DMAM0, DMAM1）　… 233
6.6.2　DMAコントロール・レジスタ（DMAC0, DMAC1）　… 237
6.6.3　DMAステータス・レジスタ（DMAS）　… 240
6.7　DMAサービス・チャネル　… 242
6.7.1　ターミナル・カウンタ（TC0, TC1）　… 242

保守／廃止

6.7.2　ターミナル・カウンタ・モジュロ・レジスタ（TCM0, TCM1）　…　243
6.7.3　DMAアップ・ダウン・カウンタ（UDC0, UDC1）　…　243
6.7.4　DMAリード・ライト・ポインタ（DPTC0, DPTC1）　…　243
6.7.5　DMAメモリ・アドレス・レジスタ（MAR0, MAR1）　…　243
6.7.6　DMAコンペア・レジスタ（DCM0, DCM1）　…　244
6.8　割り込み要求　…　245
6.9　DMA要求に対する応答時間　…　246
6.10　DMA転送タイミング　…　247
6.11　注意事項　…　260
6.11.1　メイン・チャネル側DMAコントローラにおける転送対象指定の制限　…　260
6.11.2　２チャネル動作モード使用時の制限　…　262
6.11.3　リフレッシュ・サイクル自動挿入許可時の転送モード　…　263
6.11.4　UART受信割り込み要求をDMAの転送要求ソースとしている場合　…　263

第７章　シリアル・インタフェース機能　…　265

7.1　特　　徴　…　265
7.2　通信プロトコル　…　265
7.2.1　UART　…　266
7.2.2　CSI　…　266
7.2.3　制御レジスタ　…　267
7.3　ボー・レート・ジェネレータ　…　270
7.3.1　構　　成　…　270
7.3.2　動　　作　…　271
7.3.3　制御レジスタ　…　273
7.4　UART動作　…　275
7.4.1　構　　成　…　275
7.4.2　送信動作　…　276
7.4.3　受信動作　…　277
7.4.4　ウエーク・アップ機能　…　278
7.4.5　制御レジスタ　…　280
7.4.6　割り込み要求　…　288
7.4.7　DMA要求　…　291
7.4.8　注意事項　…　292
7.5　CSI動作　…　294
7.5.1　構　　成　…　294
7.5.2　３線式シリアルI/Oモード（IOEモード）の動作　…　299
7.5.3　シリアル・バス・インタフェース・モード（SBIモード）の動作　…　302
7.5.4　制御レジスタ　…　309
7.5.5　割り込み要求　…　320
7.5.6　注意事項　…　320

第８章　パラレル・インタフェース機能　…　321

8.1　特　　徴　…　321
8.2　構　　成　…　321

保守／廃止

8.3 動　作 … 324
8.3.1 入力モード … 324
8.3.2 出力モード … 325
8.4 制御レジスタ … 328
8.4.1 パラレル・インタフェース・コントロール・レジスタ0（PAC0） … 328
8.4.2 パラレル・インタフェース・コントロール・レジスタ1（PAC1） … 333
8.4.3 パラレル・インタフェース・ステータス・レジスタ（PAS） … 336
8.4.4 パラレル・インタフェース・アクノリッジ・インターバル・レジスタ（PAI1, PAI2） … 338
8.4.5 パラレル・インタフェース・バッファ・レジスタ（PAB） … 339
8.5 割り込み要求 … 339
8.5.1 入力モード時 … 339
8.5.2 出力モード時 … 339
8.6 動作タイミング … 340

第9章　タイマ機能 … 353

9.1 特　徴 … 353
9.2 構　成 … 353
9.3 動　作 … 356
9.3.1 カウント機能 … 356
9.3.2 キャプチャ機能 … 360
9.3.3 コンペア機能 … 361
9.3.4 タイマ出力機能 … 364
9.3.5 カスケード接続 … 368
9.4 制御レジスタ … 369
9.4.1 タイマ・コントロール・レジスタ（TMC0, TMC1） … 369
9.4.2 タイマ出力制御レジスタ（TOC0, TOC1） … 371
9.5 割り込み要求 … 372
9.5.1 タイマ・ユニットからの割り込み要求 … 372
9.5.2 キャプチャ・トリガ兼用外部割り込み … 372
9.6 注意事項 … 373

第10章　リアルタイム出力ポート機能 … 375

10.1 特　徴 … 375
10.2 構　成 … 375
10.3 動　作 … 378
10.3.1 基本動作 … 378
10.3.2 ディレイ機能 … 380
10.3.3 マクロ・サービスとの組み合わせ … 381
10.3.4 応 用 例 … 382
10.3.5 ポート・アクセス時の動作 … 383
10.4 制御レジスタ … 384
10.4.1 ポート7モード・コントロール・レジスタ（PMC7） … 384
10.4.2 リアルタイム出力ポート・コントロール・レジスタ（RTPC） … 385
10.4.3 リアルタイム出力ポート・ディレイ指定レジスタ（RTPD） … 387

保守／廃止

10.4.4 ディレイ・カウンタ … 387

第11章 PWM出力機能 … 389

11.1 特 徴 … 389
11.2 構 成 … 389
11.3 動 作 … 391
11.3.1 設 定 … 391
11.3.2 基本動作 … 391
11.3.3 PWM出力中の動作 … 392
11.4 制御レジスタ … 394
11.4.1 PWMコントロール・レジスタ（PWMC） … 394
11.4.2 PWMレジスタ（PWM） … 394

第12章 ウォッチドッグ・タイマ機能 … 395

12.1 特 徴 … 395
12.2 構成と動作 … 395
12.3 制御レジスタ … 397
12.3.1 ウォッチドッグ・タイマ・モード・レジスタ（WDM） … 397

第13章 A/Dコンバータ機能 … 399

13.1 特 徴 … 399
13.2 構 成 … 400
13.3 動 作 … 403
13.3.1 A/Dコンバータの基本動作 … 403
13.3.2 セレクト・モード … 405
13.3.3 スキャン・モード … 406
13.4 制御レジスタ … 408
13.4.1 A/Dコンバータ・モード・レジスタ（ADM） … 408
13.4.2 A/Dコンバージョン・リザルト・レジスタ（ADCR0-ADCR3） … 409
13.5 割り込み要求 … 409
13.6 注意事項 … 409
13.6.1 STOP命令を実行する場合について … 409
13.6.2 A/D変換動作シーケンスの初期化について … 409
13.6.3 変換結果の取りこぼしについて … 410

第14章 ポート機能 … 411

14.1 特 徴 … 411
14.2 構 成 … 411
14.2.1 出力ポートに指定された場合（PMXn＝0） … 413
14.2.2 入力ポートに指定された場合（PMXn＝1） … 415
14.2.3 コントロール指定の場合（PMXn＝1） … 416
14.3 ポートの各端子機能 … 417
14.3.1 ポート0（P00-P07） … 417

保守／廃止

14.3.2 ポート1（P10-P16） … 418
14.3.3 ポート2（P20-P25） … 419
14.3.4 ポート3（P30-P36） … 420
14.3.5 ポート4（P40-P47） … 421
14.3.6 ポート5（P50-P52） … 423
14.3.7 ポート6（P60-P63） … 424
14.3.8 ポート7（P70-P77） … 425
14.3.9 ポート8（P80, P81） … 426

第15章 スタンバイ機能 … 429

15.1 特 徴 … 429
15.2 スタンバイ・モード … 429
15.3 制御レジスタ … 430
15.3.1 スタンバイ・コントロール・レジスタ（STBC） … 430
15.4 HALTモード … 430
15.4.1 HALTモードの設定および動作 … 430
15.4.2 HALTモードの解除 … 430
15.5 STOPモード … 432
15.5.1 STOPモードの設定および動作 … 432
15.5.2 STOPモードの解除 … 432

第16章 クロック発生機能 … 435

16.1 特 徴 … 435
16.2 構 成 … 435
16.3 動 作 … 436
16.3.1 クロック発振回路 … 436
16.3.2 タイム・ベース・カウンタ（TBC） … 436
16.4 プロセッサ・コントロール・レジスタ（PRC） … 438
16.5 クロック切り替えタイミング … 440

第17章 ソフトウエア・インターバル・タイマ機能 … 441

17.1 構 成 … 441
17.2 制御レジスタ … 441
17.3 割り込み要求 … 442

第18章 コーデック機能 … 443

18.1 特 徴 … 443
18.2 2値画情報の符号化／復合化方式 … 446
18.2.1 ラン・レングス符号化 … 446
18.2.2 MH方式 … 449
18.2.3 MR方式 … 453
18.2.4 MMR方式 … 460
18.3 メモリ・マップ … 461

保守／廃止

18.4 符号化／復号化変換テーブル … 464
18.4.1 符号化変換テーブル … 464
18.4.2 復号化変換テーブル … 466
18.4.3 復号化変換テーブル（MH復号化命令用） … 468
18.4.4 符号化変換テーブル（MH符号化命令用） … 478
18.5 処理フロー … 485
18.6 各命令の機能説明 … 487
18.6.1 データ送出命令 … 487
18.6.2 変化点テーブル作成命令 … 490
18.6.3 MH符号化命令 … 492
18.6.4 MR符号化命令 … 494
18.6.5 EOL検出命令 … 496
18.6.6 1ビット検出命令 … 500
18.6.7 MH復号化変化点テーブル作成命令 … 502
18.6.8 MR復号化変化点テーブル作成命令 … 505
18.6.9 画素データ作成命令 … 508
18.7 処理時間 … 510

第19章 リセット機能 … 513

付録A ニモニック一覧 … 517

付録B 注意事項 … 551

B.1 割り込み機能 … 551
B.2 DMA機能 … 552
B.3 シリアル・インタフェース機能（UART） … 556
B.4 シリアル・インタフェース機能（CSI） … 558
B.5 タイマ機能 … 558
B.6 リアルタイム出力ポート機能 … 559
B.7 A/Dコンバータ機能 … 559
B.8 コーデック機能 … 560

付録C レジスタ索引 … 561

C.1 50音順 … 561
C.2 アルファベット順 … 564

保守／廃止

〔メ　モ〕

保守／廃止

図の目次（1/6）

図番号	タイトル，ページ
1-1	V55PIの位置づけ … 1
2-1	タイプ別入出力回路 … 34
3-1	レジスタ・バンクの構成 … 39
3-2	PSWのビット構成 … 46
3-3	基本メモリ空間 … 51
3-4	拡張メモリ空間 … 52
3-5	レジスタ・ファイル空間 … 65
3-6	I/Oマップ（64 Kバイト） … 66
3-7	セグメント方式の概念図 … 67
3-8	セグメント・レジスタ，オフセット・アドレスと物理アドレスの関係 … 68
3-9	拡張セグメント・レジスタ，オフセット・アドレスと物理アドレスの関係 … 68
3-10	セグメント・レジスタとセグメント，およびメモリ空間の関係 … 70
3-11	ダイナミック・リロケーション … 73
3-12	メモリ・アドレスの計算 … 74
4-1	メモリの分割制御 … 92
4-2	リセット後のメモリ・ブロックの状態 … 93
4-3	メモリ・リード・サイクル READY端子によるウエイト挿入時 … 96
4-4	メモリ・ライト・サイクル READY端子によるウエイト挿入時 … 97
4-5	DRAM接続回路例（アドレス・ウエイト挿入時） … 100
4-6	リフレッシュ・サイクルのタイミング・チャート … 104
4-7	疑似SRAM接続回路例（アドレス・ウエイト挿入時） … 105
4-8	メモリ・リード・サイクル … 108
4-9	メモリ・リード・サイクル データ・ウエイト（1ウエイト）挿入時 … 109
4-10	メモリ・リード・サイクル アドレス・ウエイト挿入時 … 110
4-11	メモリ・リード・サイクル データ・ウエイト（1ウエイト）アドレス・ウエイト挿入時 … 111
4-12	メモリ・ライト・サイクル … 112
4-13	メモリ・ライト・サイクル データ・ウエイト（1ウエイト）挿入時 … 113
4-14	メモリ・ライト・サイクル アドレス・ウエイト挿入時 … 114
4-15	メモリ・ライト・サイクル データ・ウエイト（1ウエイト）アドレス・ウエイト挿入時 … 115
4-16	I/Oリード・サイクル … 116
4-17	I/Oリード・サイクル データ・ウエイト（1ウエイト）挿入時 … 117

保守／廃止

図の目次（2/6）

図番号	タイトル，ページ
4-18	I/Oライト・サイクル … 118
4-19	I/Oライト・サイクル　データ・ウエイト（1ウエイト）挿入時 … 119
5-1	マスカブル割り込みの処理形態 … 127
5-2	割り込み受け付け動作（①→④の順に行われる） … 132
5-3	ベクタ・テーブルの構成 … 132
5-4	ベクタ・テーブルとレジスタ・バンク番号 … 133
5-5	レジスタ・バンク切り替えのシーケンス（割り込みによるレジスタ・バンク切り替え時） … 135
5-6	レジスタ・バンク復帰シーケンス … 136
5-7	TSKSW命令によるレジスタ・バンク切り替えのシーケンス … 138
5-8	マクロ・サービスによる割り込み処理効率 … 139
5-9	マクロ・サービス・コントロール・ワードの基本構成 … 142
6-1	$\overline{\text{DMAAK}}$出力の接続例（シングルトランスファ・モード） … 201
6-2	DMA転送のタイミング例（シングルトランスファ・モード） … 202
6-3	チャネル構成図 … 203
6-4	シングルトランスファ・モード転送タイミング … 207
6-5	ディマンド・リリース・モード転送タイミング … 207
6-6	シングルステップ・モード転送タイミング … 208
6-7	バースト・モード転送タイミング … 208
6-8	インテリジェントDMAモード-1（リング・バッファ方式） … 210
6-9	デクリメント・モード時のバッファ・メモリ（インテリジェントDMAモード-1） … 212
6-10	インテリジェントDMAモード-2（カウンタ制御方式） … 215
6-11	デクリメント・モード時のバッファ・メモリ（インテリジェントDMAモード-2） … 217
6-12	ネクスト・アドレス指定モード … 219
6-13	2チャネル動作モード（停止モード） … 222
6-14	2チャネル動作モード（繰り返しモード） … 224
6-15	メモリ-メモリ転送モード（停止モード） … 226
6-16	メモリ-メモリ転送モード（繰り返しモード） … 228
6-17	メモリ→外部I/O … 248
6-18	外部I/O→メモリ … 250
6-19	メモリ→内部I/O（SFR） … 252
6-20	内部I/O（SFR）→メモリ … 253
6-21	メモリ↔メモリ … 254

保守／廃止

図の目次（3/6）

図番号	タイトル，ページ
6-22	メモリ→SFR（メモリ・アクセス） … 256
6-23	SFR（メモリ・アクセス）→メモリ … 258
7-1	ボー・レート・ジェネレータのブロック図 … 270
7-2	UARTのブロック図 … 275
7-3	基本的なデータ・フォーマット … 277
7-4	複数のシステムを接続した例 … 279
7-5	ウエーク・アップ機能を選択したときのデータ・フォーマット … 280
7-6	UART受信エラー割り込み発生タイミング … 289
7-7	UART受信割り込み発生タイミング … 289
7-8	UART送信割り込み発生タイミング … 290
7-9	クロックト・シリアル・インタフェースのブロック図 … 296
7-10	IOEモードのシステム構成例 … 299
7-11	IOEモードの転送タイミング … 300
7-12	SBIモードのシステム構成例 … 303
7-13	マスタ・デバイスからスレーブ・デバイスへのアドレス転送動作 … 304
7-14	マスタ・デバイスからスレーブ・デバイスへのコマンド転送動作 … 304
7-15	マスタ・デバイスからスレーブ・デバイスへのデータ転送動作 … 305
7-16	スレーブ・デバイスからマスタ・デバイスへのデータ転送動作 … 305
7-17	$\overline{\text{SCK0}}$端子とバス・ライン … 314
7-18	$\overline{\text{SCK0}}$端子とバス・ライン（コマンド条件検出） … 314
7-19	RELT，CMDT，RELD，CMDDの動作 … 316
7-20	ACKTの動作 … 316
7-21	ACKEの動作 … 317
7-22	ACKDの動作 … 318
7-23	BSYEの動作 … 319
8-1	パラレル・インタフェースのブロック図 … 322
8-2	タイミング・チャート（出力モード時） … 327
8-3	タイミング・チャート（STBT=0，BSYT=0） … 341
8-4	タイミング・チャート（STBT=1，BSYT=0） … 341
8-5	タイミング・チャート（STBT=1，BSYT=1） … 341
8-6	タイミング・チャート（STBT=0，BSYT=1） … 342
8-7	タイミング・チャート（ACKT1，ACKT0=00） … 343

保守／廃止

図の目次（4/6）

図番号	タイトル，ページ
8-8	タイミング・チャート（ACKT1，ACKT0＝01） … 343
8-9	タイミング・チャート（ACKT1，ACKT0＝10） … 343
8-10	タイミング・チャート（ACKT1，ACKT0＝00） … 344
8-11	タイミング・チャート（ACKT1，ACKT0＝01） … 344
8-12	タイミング・チャート（ACKT1，ACKT0＝10） … 344
8-13	タイミング・チャート（ACKT1，ACKT0＝00） … 345
8-14	タイミング・チャート（ACKT1，ACKT0＝01） … 345
8-15	タイミング・チャート（ACKT1，ACKT0＝10） … 345
8-16	タイミング・チャート（ACKT1，ACKT0＝00） … 346
8-17	タイミング・チャート（ACKT1，ACKT0＝01） … 346
8-18	タイミング・チャート（ACKT1，ACKT0＝10） … 346
8-19	タイミング・チャート（ACKT1，ACKT0＝00） … 347
8-20	タイミング・チャート（ACKT1，ACKT0＝01） … 347
8-21	タイミング・チャート（ACKT1，ACKT0＝10） … 347
8-22	タイミング・チャート（ACKT1，ACKT0＝00） … 348
8-23	タイミング・チャート（ACKT1，ACKT0＝01） … 348
8-24	タイミング・チャート（ACKT1，ACKT0＝10） … 348
8-25	タイミング・チャート（ACKT1，ACKT0＝00） … 349
8-26	タイミング・チャート（ACKT1，ACKT0＝01） … 349
8-27	タイミング・チャート（ACKT1，ACKT0＝10） … 349
8-28	タイミング・チャート（ACKT1，ACKT0＝00，ACKM＝0） … 350
8-29	タイミング・チャート（ACKT1，ACKT0＝01，ACKM＝0） … 350
8-30	タイミング・チャート（ACKT1，ACKT0＝10，ACKM＝0） … 350
8-31	タイミング・チャート（ACKT1，ACKT0＝00，ACKM＝0） … 351
8-32	タイミング・チャート（ACKT1，ACKT0＝01，ACKM＝0） … 351
8-33	タイミング・チャート（ACKT1，ACKT0＝10，ACKM＝0） … 351
9-1	タイマ・ユニットのブロック図 … 355
9-2	フリー・ランニング・タイマ動作（タイマ０の場合） … 357
9-3	インターバル・タイマ動作（CM10をコンペア・レジスタとして使用する場合） … 358
9-4	キャプチャ動作（タイマ０の場合） … 360
9-5	コンペア動作（タイマ０の場合） … 362
9-6	タイマ０をインターバル・タイマとして使用する場合 … 363
9-7	タイマ０によるトグル出力 … 366

保守／廃止

図の目次（5/6）

図番号	タイトル，ページ
9-8	タイマ2によるセット／リセット出力 … 367
9-9	カスケード接続によるTO30出力波形 … 368
10-1	リアルタイム出力ポートのブロック図 … 377
10-2	リアルタイム出力ポートの動作タイミング … 379
10-3	リアルタイム出力ポートの出力動作（ディレイ・モード） … 381
10-4	リアルタイム出力ポートによるステッピング・モータ制御 … 382
11-1	PWMユニットのブロック図 … 390
11-2	PWM出力動作タイミング … 391
11-3	PWM出力動作タイミング（CE＝0→1にしたとき） … 392
11-4	PWM出力動作タイミング（CE＝1→0にしたとき） … 393
11-5	PWM出力動作タイミング（ALVビットを変化させたとき） … 393
12-1	$\overline{\text{RESET}}$端子の接続例 … 395
12-2	ウォッチドッグ・タイマの構成図 … 397
13-1	A/Dコンバータのブロック図 … 402
13-2	A/Dコンバータの基本動作 … 404
13-3	セレクト・モードの動作タイミング … 406
13-4	スキャン・モードの動作タイミング … 407
14-1	ポート0，2-5，7，8の構成 … 412
14-2	出力ポート指定のポート … 413
14-3	入力ポート指定のポート … 415
14-4	コントロール指定のポート … 416
14-5	ポート1のブロック図 … 418
14-6	ポート6のブロック図 … 425
15-1	NMI端子入力によるSTOPモードの解除動作 … 432
16-1	クロック発生回路 … 435
16-2	クロック発生回路の外付け回路 … 436
16-3	分周器（タイム・ベース・カウンタTBC）の構成 … 437

保守／廃止

図の目次（6/6）

図番号	タイトル，ページ
16－4	システム・クロックの切り替えタイミング … 440
17－1	ソフトウエア・インターバル・タイマの構成 … 441
18－1	MH，MR符号化処理フロー … 444
18－2	MH，MR復号化処理フロー … 445
18－3	サンプル図 … 447
18－4	ビット・マップ・データ … 447
18－5	ラン・レングス符号列 … 448
18－6	1ライン分の符号化例 … 451
18－7	1ページ分のMH符号 … 452
18－8	画像の垂直方向の相関 … 453
18－9	符号化の基準となる点 … 454
18－10	パス・モードの例 … 454
18－11	垂直モードの例（a1がb1の右側2画素目にある場合） … 455
18－12	水平モードの例 … 456
18－13	ライン始端の処理例 … 457
18－14	ライン終端の処理例 … 457
18－15	Kパラメータ（K＝4の場合） … 458
18－16	1ページ分のMR符号 … 459
18－17	1ページ分のMMR符号 … 460
18－18	符号化命令と各メモリ上のデータ … 463
18－19	復号化命令と各メモリ上のデータ … 463
18－20	1ライン分の符号化処理の流れ … 485
18－21	1ライン分の復号化処理の流れ … 486
18－22	MH符号化／復号化命令の処理時間 … 510
18－23	MR符号化／復号化命令の処理時間 … 511

保守／廃止

表の目次（1/3）

表番号	タイトル，ページ
1-1	V55PIとV25の機能比較一覧 … 3
2-1	ポート端子およびコントロール端子の状態 … 16
2-2	ポート以外の端子の状態 … 18
2-3	ポート0の動作（n=0-7） … 19
2-4	ポート1の動作 … 19
2-5	ポート2の動作（n=0-5） … 20
2-6	ポート3の動作（n=0-6） … 21
2-7	ポート4の動作（n=0-7） … 23
2-8	ポート5の動作（n=0-2） … 24
2-9	ポート6の動作 … 26
2-10	ポート7の動作（n=0-7） … 26
2-11	ポート8の動作（n=0, 1） … 27
2-12	メモリ・ライト・サイクルにおける$\overline{WRL}$, $\overline{WRH}$端子の動作(外部データ・バス幅16ビット時) … 30
3-1	汎用レジスタのオフセット … 40
3-2	ポインタとインデクス・レジスタのオフセット … 41
3-3	セグメント・レジスタのオフセット … 42
3-4	拡張セグメント・レジスタのオフセット … 44
3-5	特殊機能レジスター覧 … 54
3-6	セグメント・レジスタとオフセット・アドレス指定の対応 … 69
4-1	SFRアクセス時のウエイト（1ウエイト以上） … 98
5-1	割り込み要因一覧 … 123
5-2	割り込み要求とマクロ・サービス・コントロール・ワード・アドレス … 143
5-3	マクロ・サービス動作モードの分類 … 145
5-4	EVENTモードの動作指定 … 149
5-5	データのバイト／ワード指定 … 157
5-6	BLKTRSモードの動作指定 … 160
5-7	メモリ空間のアドレス・モード指定 … 160
5-8	BLKTRS-Cモードの動作指定 … 163
5-9	メモリ空間のアドレス・モード指定 … 163
5-10	タイマ・コンペア・レジスタ設定データ・テーブルのアドレス更新設定 … 165

保守／廃止

表の目次（2/3）

表番号	タイトル，ページ
5-11	タイマ・コンペア・レジスタ設定データへの操作指定 … 165
5-12	割り込み要求信号とSFRの対応 … 168
5-13	リアルタイム出力ポートへの書き込み … 168
5-14	ソフトウエアによる割り込み … 180
5-15	マクロ・サービス処理時間 … 195
6-1	DMA転送対象と転送方法 … 200
6-2	設定可能な転送対象と起動要因の組み合わせ … 205
6-3	転送モード … 208
6-4	動作モード一覧 … 209
6-5	IOPnとDPTCnの対応 … 214
6-6	動作モードとレジスタの対応 … 242
6-7	外部データ・バス幅とDMA転送対象 … 261
7-1	サポート・プロトコル … 265
7-2	制御レジスター覧 … 268
7-3	制御端子一覧 … 269
7-4	BRGレジスタへの設定値 … 272
7-5	受信エラーの要因 … 288
7-6	DMA要求と接続できるUART割り込み要求 … 291
7-7	クロックト・シリアル・インタフェース用端子 … 295
7-8	制御レジスタによる機能指定（SO/SB0端子）…ch0 … 297
7-9	制御レジスタによる機能指定（SI/SB1端子）…ch0 … 297
7-10	制御レジスタによる機能指定（SO端子）…ch1 … 298
7-11	制御レジスタによる機能指定（SI端子）…ch1 … 298
7-12	制御レジスタによる機能指定（$\overline{\text{SCK}}$端子） … 298
7-13	SBIモードにおける各種の信号 … 307
8-1	割り込み要求とDMA要求発生のトリガ端子指定（出力モード） … 326
8-2	$\overline{\text{ACK}}$信号の出力指定（MBビットのリセット時） … 330
8-3	ACKM，ACKT1，ACKT0の設定とBUSY信号および$\overline{\text{ACK}}$信号のタイミング … 330
8-4	代表的なPAI設定値と$\overline{\text{ACK}}$出力幅 … 338
9-1	タイマ・ユニットの種類 … 353

保守／廃止

表の目次（3/3）

表番号	タイトル，ページ
9-2	各タイマの機能 … 356
9-3	タイマのカウント可能時間 … 357
10-1	RTPD設定値とディレイ時間 … 380
10-2	ポート7のアクセス対象 … 383
10-3	リアルタイム出力ポート（RTP）へのデータ転送タイミング … 386
11-1	PWMレジスタの値とデューティ比 … 394
13-1	A/D変換時間 … 405
14-1	ポート0の動作（n=0-7） … 417
14-2	ポート1の動作 … 418
14-3	ポート2の動作（n=0-5） … 419
14-4	ポート3の動作（n=0-6） … 420
14-5	ポート4の動作（n=0-7） … 422
14-6	ポート5の動作（n=0-2） … 423
14-7	ポート6の動作 … 424
14-8	ポート7の動作（n=0-7） … 425
14-9	ポート8の動作（n=0, 1） … 427
15-1	HALT/STOPモードの動作状態 … 429
15-2	マクロ・サービス要求によるHALTモードの解除 … 431
18-1	ラン・レングス符号 … 450
18-2	垂直モードの符号 … 455
18-3	拡張セグメント・オーバライド・プリフィクスの付加可能なコーデック命令 … 462
18-4	MHDEC命令の終了状態 … 504
18-5	MRDEC命令の終了状態 … 507
19-1	ハードウエアのリセット後の状態 … 513
A-1	ニモニック一覧 … 518

保守／廃止

〔メ　モ〕

保守／廃止

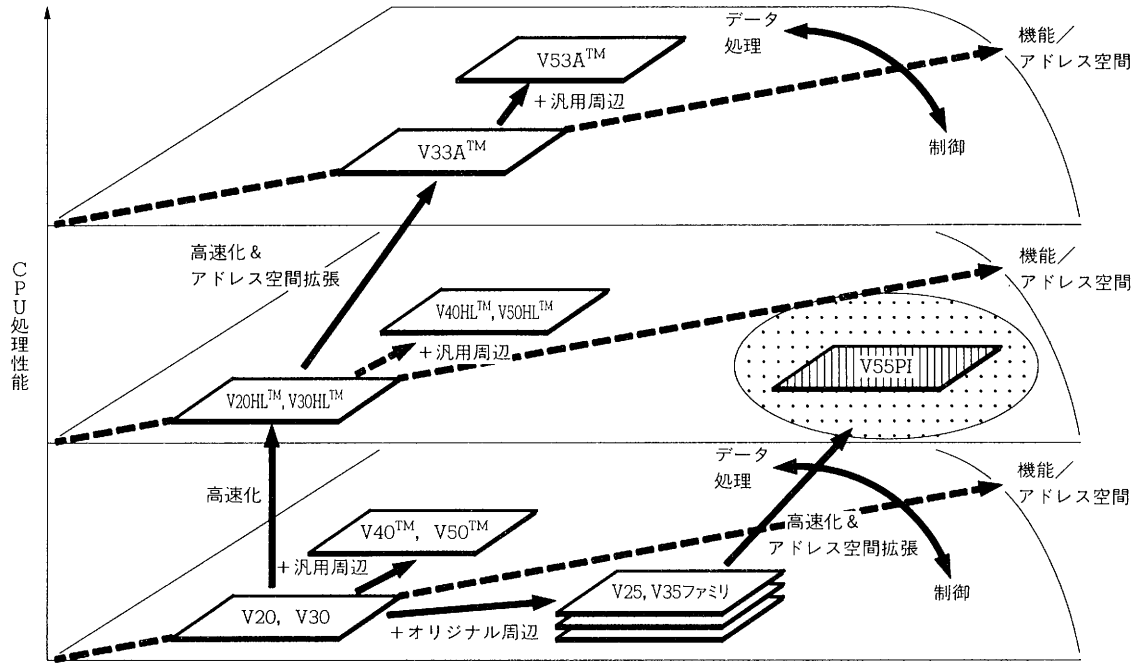
第1章　概　　説

V55PIは，16ビットCPU，RAM，シリアル・インタフェース，パラレル・インタフェース，A/Dコンバータ，タイマ，DMAコントローラ，割り込みコントローラなどを1チップに集積したマイクロプロセッサです。

V55PIは，マイクロプロセッサV20，V30および，シングルチップ・マイクロコンピュータV25，V35とソフトウエア・アップワード・コンパチブルです。V25の上位機種として，より高機能，高性能化を図っており，特にプリンタ，ファクシミリをはじめとする，メカ制御を伴うデータ処理システムの制御に最適です。

★

図1－1　V55PIの位置づけ



16ビットVシリーズの製品展開

保守／廃止

1.1 特　徴

V55PIは，次のような機能を備えています。

○内部16ビット・アーキテクチャ　外部16/8ビット・データ・バス幅選択可能

○V20，V30（ネイティブ・モード）およびV25，V35とソフトウエア・コンパチブル（追加命令あり）

○最小インストラクション・サイクル：160 ns/12.5 MHz（外部25 MHz時）
125 ns/16 MHz（外部32 MHz時）

○アドレス空間：16 Mバイト
- 基本メモリ空間1 Mバイト
- 拡張メモリ空間16 Mバイト

○レジスタ・ファイル空間（内蔵RAM上）：512バイト／16レジスタ・バンク

○I/O空間：64 Kバイト

○メモリ空間を可変サイズ（最大６ブロック）で分割して自動ウエイト制御

○I/Oライン（入力ポート：11ビット，入出力ポート：42ビット）

○DMAコントローラ（DMAC）：最大４チャネル構成可能

● ４種類のDMA転送モード（シングルトランスファ，ディマンド・リリース，シングルステップ，バースト）

●インテリジェントDMAモード－1，－2

○シリアル・インタフェース：２チャネル

●調歩同期モード（UART）またはクロック同期モード（CSI）のいずれかを選択可能

○パラレル・インタフェース：８ビット

●セントロニクス・データ入出力および汎用データ入出力

○A/Dコンバータ（８ビット分解能）：４チャネル

○リアルタイム出力ポート：４ビット×２チャネル，または８ビット×１チャネル

○PWM（Pulse Width Modulation）信号出力機能（８ビット）

○割り込みコントローラ

●プログラマブル・プライオリティ（４レベル）による多重割り込み処理制御

● 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

○16ビット・タイマ・カウンタ：４チャネル

○ウォッチドッグ・タイマ機能

○ソフトウエア・インターバル・タイマ（16ビット）

○アドレス部ウエイト挿入機能とRAS/CAS切り替えタイミング発生機能

○DRAM，疑似SRAMリフレッシュ機能

○スタンバイ機能（STOPモード，HALTモード）

○クロック発生回路内蔵

保守／廃止

1.2 V55PIとV25との機能比較

V55PIとV25との機能比較を表1－1に示します。

表1－1 V55PIとV25の機能比較一覧

比較項目			V25	V55PI
CPU	アドレス空間	メモリ	1Mバイト	16Mバイト
		I/O	64Kバイト	64Kバイト
		レジスタ・ファイル	なし	512バイト
	レジスタ・バンク（内蔵RAM）		8バンク／256バイト	16バンク／512バイト
	バス構成（内部／外部：ビット）		16/8	16/16 （外部16/8選択可能）
	ソフトウエア継承性		V30上位互換	V25，V30上位互換
周辺ハードウエア	シリアル・インタフェース	チャネル数	2ch	2ch
		インタフェース・モード	アシンクロナス・モード（UART） I/Oインタフェース・モード	UART クロック同期式（SBI/IOEモード）
	パラレル・インタフェース	チャネル数	なし	1ch（8ビット）
		仕様		セントロニクス対応可
	DMA機能	チャネル数	2ch	4ch
		転送モード	4モード	4モード
		動作モード		5モード
	割り込み機能	レベル	8レベル	4レベル
		応答方式	ベクタ割り込み レジスタ・バンク切り替え マクロ・サービス （2モード）	ベクタ割り込み レジスタ・バンク切り替え マクロ・サービス （7モード）
	汎用ポート		4（入力） 20（入出力）	11（入力） 42（入出力）
	リアルタイム出力ポート		なし	2ch（4ビット）
	タイマ機能		2ch（16ビット）	5ch（16ビット） （ソフトウエア・インターバル・タイマ含む）
	PWM出力機能		なし	1ch（8ビット）
	A/Dコンバータ		なし	4ch（8ビット）
	プログラマブル・ウエイト機能		メモリ7分割（アドレス固定）	メモリ6分割（アドレス可変）
	リフレッシュ機能（リフレッシュ・アドレス）		あり（9ビット）	あり（20ビット）
	ウォッチドッグ・タイマ機能		なし	あり
	スタンバイ機能		STOP，HALT	STOP，HALT

保守／廃止

★ 1.3 オーダ情報

オーダ名称	パッケージ	最大動作周波数(MHz)
μPD70433GD-12-5BB	120ピン・プラスチックQFP(□28 mm)	12.5
μPD70433GD-16-5BB	〃	16
μPD70433R-12	132ピン・セラミックPGA	12.5
μPD70433R-16	〃	16
μPD70433GJ-12-3EB	120ピン・プラスチックQFP(ファインピッチ)(□20 mm)	12.5
μPD70433GJ-16-3EB	〃	16

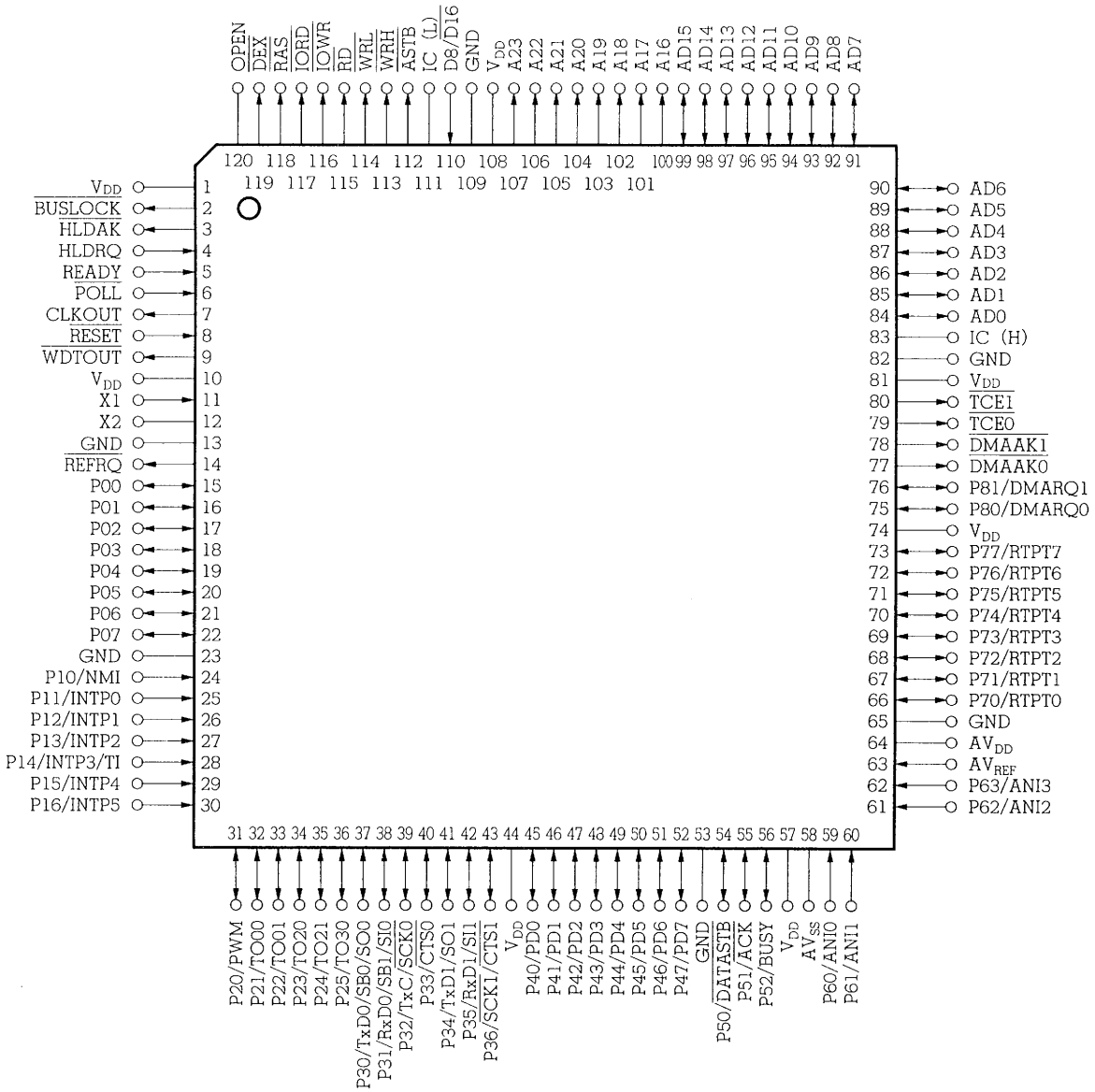
保守／廃止

1.4 端子接続図（Top View）

（1）120ピン・プラスチックQFP（□28 mm），
120ピン・プラスチックQFP（ファインピッチ）（□20 mm）

μPD70433GD-xx-5BB

μPD70433GJ-xx-3EB



備考 IC：Internally Connected

注意1. IC（H）端子は外部で抵抗（1-10 kΩ）を介してV_{DD}に接続してください。

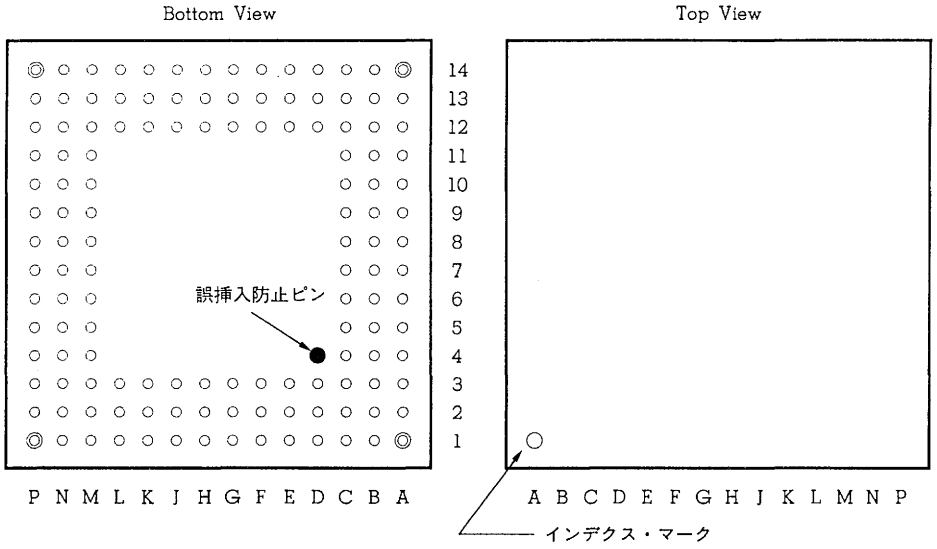
2. IC（L）端子は外部で抵抗（1-10 kΩ）を介してGNDに接続してください。

3. OPEN端子は何も接続しないでください。

保守／廃止

(2) 132ピン・セラミックPGA

μPD70433R-xx



備考　誤挿入防止ピンは，ピン数に含みません。

番号	信号名	ポート	番号	信号名	ポート	番号	信号名	ポート
A1	ANI1	P61	B5	PD7	P47	C9	$\overline{\text{CTS0}}$	P33
A2	AV_{SS}	—	B6	PD5	P45	C10	TO30	P25
A3	$\overline{\text{ACK}}$	P51	B7	PD2	P42	C11	TO00	P21
A4	$\overline{\text{DATASTB}}$	P50	B8	PD0	P40	C12	NC	—
A5	PD6	P46	B9	RxD1/SI1	P35	C13	INTP4	P15
A6	PD4	P44	B10	RxD0/SB1/SI0	P31	C14	INTP0	P11
A7	PD1	P41	B11	TO21	P24	D1	RTPT2	P72
A8	NC	—	B12	TO01	P22	D2	GND	—
A9	$\overline{\text{SCK1}}/\overline{\text{CTS1}}$	P36	B13	NC	—	D3	ANI3	P63
A10	TxD1/SO1	P34	B14	INTP3/TI	P14	D12	INTP5	P16
A11	TxC/$\overline{\text{SCK0}}$	P32	C1	RTPT1	P71	D13	INTP2	P13
A12	TxD0/SB0/SO0	P30	C2	AV_{REF}	—	D14	NMI	P10
A13	TO20	P23	C3	NC	—	E1	RTPT5	P75
A14	PWM	P20	C4	NC	—	E2	RTPT3	P73
B1	AV_{DD}	—	C5	V_{DD}	—	E3	RTPT0	P70
B2	ANI2	P62	C6	GND	—	E12	INTP1	P12
B3	ANI0	P60	C7	PD3	P43	E13	GND	—
B4	BUSY	P52	C8	V_{DD}	—	E14	—	P06

保守／廃止

番号	信号名	ポート	番号	信号名	ポート	番号	信号名	ポート
F1	RTPT7	P77	K3	AD2	—	N3	AD9	—
F2	RTPT6	P76	K12	$\overline{\text{POLL}}$	—	N4	AD11	—
F3	RTPT4	P74	K13	$\overline{\text{WDTOUT}}$	—	N5	AD14	—
F12	—	P07	K14	X1	—	N6	A18	—
F13	—	P05	L1	AD0	—	N7	A21	—
F14	—	P04	L2	AD3	—	N8	A23	—
G1	NC	—	L3	AD6	—	N9	D8/$\overline{\text{D16}}$	—
G2	DMARQ0	P80	L12	$\overline{\text{BUSLOCK}}$	—	N10	$\overline{\text{ASTB}}$	—
G3	V_{DD}	—	L13	READY	—	N11	$\overline{\text{IOWR}}$	—
G12	—	P03	L14	$\overline{\text{RESET}}$	—	N12	$\overline{\text{DEX}}$	—
G13	—	P02	M1	AD1	—	N13	V_{DD}	—
G14	—	P01	M2	AD5	—	N14	HLDRQ	—
H1	DMARQ1	P81	M3	NC	—	P1	AD7	—
H2	$\overline{\text{DMAAK0}}$	—	M4	AD8	—	P2	AD10	—
H3	$\overline{\text{DMAAK1}}$	—	M5	AD12	—	P3	AD13	—
H12	$\overline{\text{REFRQ}}$	—	M6	A16	—	P4	AD15	—
H13	—	P00	M7	A20	—	P5	A17	—
H14	NC	—	M8	V_{DD}	—	P6	A19	—
J1	$\overline{\text{TCE0}}$	—	M9	$\overline{\text{WRH}}$	—	P7	NC	—
J2	$\overline{\text{TCE1}}$	—	M10	$\overline{\text{IORD}}$	—	P8	A22	—
J3	GND	—	M11	NC	—	P9	GND	—
J12	V_{DD}	—	M12	NC	—	P10	IC (L)	—
J13	X2	—	M13	$\overline{\text{HLDAK}}$	—	P11	$\overline{\text{WRL}}$	—
J14	GND	—	M14	CLKOUT	—	P12	$\overline{\text{RD}}$	—
K1	V_{DD}	—	N1	AD4	—	P13	$\overline{\text{RAS}}$	—
K2	IC (H)	—	N2	NC	—	P14	OPEN	—

備考　IC：Internally Connected

NC：Non-Connection

注意1．IC（H）端子は外部で抵抗（1-10 kΩ）を介してV_{DD}に接続してください。

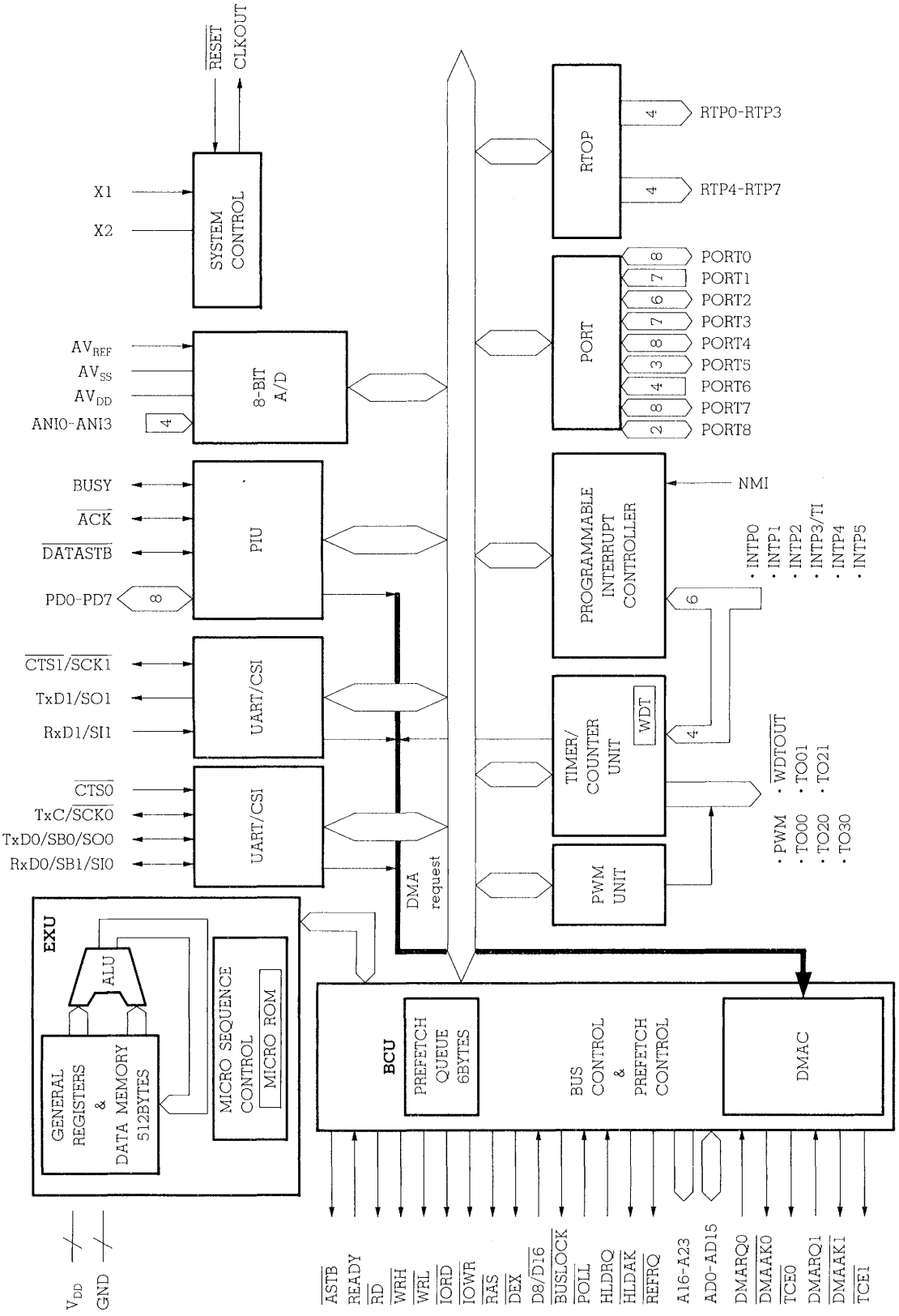
2．IC（L）端子は外部で抵抗（1-10 kΩ）を介してGNDに接続してください。

3．OPEN端子は何も接続しないでください。

保守／廃止

1.5 内部ブロック

1.5.1 内部ブロック図



保守／廃止

1.5.2 内部ブロック構成

○バス・コントロール・ユニット（BCU）

バスの制御を行います。BCUではエクセキューション・ユニット（EXU）で得られた物理アドレスに基づいて，必要な内部／外部のバス・サイクルを起動します。

○エクセキューション・ユニット（EXU）

アドレス計算，算術論理計算，データ転送などをマイクロプログラム（オペコードのデコード結果に基づいて，マイクロシーケンサを制御するためのファームウエア）で制御します。

EXUの内部には512バイトのRAM（レジスタ・ファイル空間に対応）を内蔵しています。

○割り込みコントローラ（INTC）

内蔵周辺ハードウエアおよび外部からのハードウエア割り込み要求を，ベクタ割り込み，レジスタ・バンク切り替えまたはマクロ・サービスのいずれかによって処理します。さらに，プログラマブルな4レベルの割り込み優先順位を制御可能であり，割り込み要因に対し多重処理制御を行うこともできます。

○DMAコントローラ（DMAC）

汎用DMAコントローラで，メモリ空間16Mバイトをリニアに扱うことができます。動作モードにはメモリーメモリ転送モードのほか，インテリジェントDMA（リング・バッファ方式およびカウンタ制御方式）モード，ネクスト・アドレス指定モード，2チャネル動作モードがあります。

○UART/クロックト・シリアル・インタフェース（UART/CSI）

スタート／ストップ・ビットによりデータ同期をとる調歩同期式シリアル・インタフェース（UART）とクロック同期式シリアル・インタフェース（CSI）をサポートしており，選択して使用可能です。

クロックト・シリアル・インタフェースは，さらにシリアル・バス・インタフェース・モード（SBI）と3線式シリアルI/Oモードを選択して使用できます。

○パラレル・インタフェース・ユニット（PIU）

ストローブ信号同期で8ビット単位の入出力を行い，セントロニクス・インタフェースおよび汎用パラレル・データ通信機能をサポートします。

○A/Dコンバータ・ユニット（8ビットA/D）

4本のアナログ入力を持つ逐次変換方式の8ビットA/Dコンバータで，A/D変換結果レジスタを4本備えています。

保守／廃止

○タイマ／カウンタ・ユニット（TCU）

タイマ／カウンタ・ユニットは16ビット・タイマ／カウンタを内蔵し，インターバル・タイマ，フリー・ランニング・タイマ，イベント・カウンタとして使用できます。

○PWM（Pulse Width Modulation）ユニット（PWM）

８ビット精度のPWM（パルス幅変調）信号出力機能を備えています。

○ウォッチドッグ・タイマ（WDT）

プログラムの暴走，システムの異常などを検出するための８ビットのウォッチドッグ・タイマを内蔵しています。ウォッチドッグ・タイマ割り込みの発生を外部に知らせるため$\overline{\text{WDTOUT}}$端子を備えています。

○ポート（PORT）

53本のポート端子を備え，ポート端子とコントロール端子の機能を選択して使用することができます。

○リアルタイム出力ポート（RTOP）

タイマ０からの割り込みをトリガとしたリアルタイム出力ポートで，プログラマブルなインターバルで８ビット・バッファ・レジスタの内容を４ビット単位または８ビット単位で出力できます。

○クロック・ジェネレータ（CG）

X1，X2端子に接続されたクリスタル振動子／セラミック発振子の固有振動数の1/2，1/4，1/8，1/16の周波数のクロックを発生し，CPU動作クロックとして供給します。

○ソフトウエア・インターバル・タイマ（SIT）

ソフトウエア・タイマ機能用または時計機能用のタイマとして16ビット・ソフトウエア・インターバル・タイマを内蔵しています。入力クロック（カウント・クロック）の選択，ソフトウエア・タイマ・カウンタ・コンペア・レジスタの設定により，インターバル割り込みを設定できます。

保守／廃止

第２章　端子機能

V55PIの端子は，機能別にポート端子とそれ以外の端子に分けることができます。

2.1　端子機能一覧

2.1.1　ポート機能

端子名称	入出力	機　　能	兼用端子
P00-P07	入出力	ポート０ ビット単位で入出力の指定可能 ８ビット入出力ポート	—
P10[注]	入　力	ポート１ ７ビット入力ポート P10は有効エッジが入力されるとNMI入力として動作します。また，P1レジスタのビット０でNMI入力の状態を示します。	NMI
P11			INTP0
P12			INTP1
P13			INTP2
P14			INTP3/TI
P15			INTP4
P16			INTP5
P20	入出力	ポート２ ビット単位で入出力の指定可能 ６ビット入出力ポート	PWM
P21			TO00
P22			TO01
P23			TO20
P24			TO21
P25			TO30
P30		ポート３ ビット単位で入出力の指定可能 ７ビット入出力ポート	TxD0/SB0/SO0
P31			RxD0/SB1/SI0
P32			TxC/$\overline{\text{SCK0}}$
P33			$\overline{\text{CTS0}}$
P34			TxD1/SO1
P35			RxD1/SI1
P36			$\overline{\text{CTS1}}$/$\overline{\text{SCK1}}$
P40-P47		ポート４ ビット単位で入出力の指定可能 ８ビット入出力ポート	PD0-PD7

注　汎用ポートとしては使用不可（ノンマスカブル割り込み）

保守／廃止

端子名称	入出力	機　　能	兼用端子
P50	入出力	ポート5 ビット単位で入出力の指定可能 3ビット入出力ポート	$\overline{\text{DATASTB}}$
P51			$\overline{\text{ACK}}$
P52			BUSY
P60-P63	入　力	ポート6 ビット単位で入出力の指定可能 4ビット入力ポート	ANI0-ANI3
P70-P77	入出力	ポート7 ビット単位で入出力の指定可能 8ビット入出力ポート	RTP0-RTP7
P80		ポート8 ビット単位で入出力の指定可能 2ビット入出力ポート	DMARQ0
P81			DMARQ1

保守／廃止

2.1.2　ポート以外の機能

（1）バス制御のための端子機能

端子名称	入出力	機　　能	兼用端子
$\overline{\text{ASTB}}$	出　力	外部バスの外部バス・サイクルのアドレス・ストローブ信号出力	—
$\overline{\text{RD}}$		外部バスの外部メモリ・サイクルのデータ・リード・ストローブ信号出力	
$\overline{\text{WRL}}$		外部バスの外部メモリ・サイクルの下位バイト・データ・ライト・ストローブ信号出力	
$\overline{\text{WRH}}$		外部バスの外部メモリ・サイクルの上位バイト・データ・ライト・ストローブ信号出力	
READY	入　力	外部バスの外部バス・サイクルのレディ信号入力	
$\overline{\text{DEX}}$	出　力	外部バス・サイクルの上位バイト・データ・イネーブル信号出力	
$\overline{\text{RAS}}$		DRAMロウ・アドレス・ラッチ・タイミング信号出力	
D8/$\overline{\text{D16}}$	入　力	外部バスのデータ・バス幅選択信号入力	
$\overline{\text{BUSLOCK}}$	出　力	外部バスのバス・ロック信号出力	
$\overline{\text{POLL}}$	入　力	$\overline{\text{POLL}}$信号（POLL命令実行時にサンプリング）入力	
HLDRQ		外部バスのホールド・リクエスト信号入力	
$\overline{\text{HLDAK}}$	出　力	外部バスのホールド・アクノリッジ信号出力	
$\overline{\text{REFRQ}}$		リフレッシュ・ステータス信号出力	
AD0-AD15	3ステート入出力	外部バスの外部バス・サイクルのアドレス／データ・マルチプレクス信号入出力	
A16-A23	3ステート出　力	外部バスの外部バス・サイクルのアドレス信号出力	
$\overline{\text{IORD}}$	出　力	外部I/Oサイクルのデータ・リード・ストローブ信号出力	
$\overline{\text{IOWR}}$		外部I/Oサイクルのデータ・ライト・ストローブ信号出力	
DMARQ0	入　力	DMA要求信号入力（チャネル０）	P80
DMARQ1		DMA要求信号入力（チャネル１）	P81
$\overline{\text{DMAAK0}}$	出　力	DMAアクノリッジ信号出力（チャネル０）	—
$\overline{\text{DMAAK1}}$		DMAアクノリッジ信号出力（チャネル１）	
$\overline{\text{TCE0}}$		DMA終了信号出力（チャネル０）	
$\overline{\text{TCE1}}$		DMA終了信号出力（チャネル１）	

保守／廃止

（2）それ以外の端子機能

端子名称	入出力	機　　能	兼用端子
GND	—	GND電位	—
V_{DD}		正電源供給	
AV_{SS}		A/Dコンバータ のGND電位	
AV_{DD}		A/Dコンバータのアナログ電源供給	
AV_{REF}	入力	A/Dコンバータの基準電圧入力	
$\overline{RESET}$		システム・リセット信号入力	
X1		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。外部クロックを供給する場合はX1に入力し，X2はオープンにしてください。	
X2	—		
CLKOUT	出　力	内部システム・クロックϕの出力	
$\overline{WDTOUT}$		ウォッチドッグ・タイマのオーバフロー信号出力	
NMI	入　力	ノンマスカブル割り込み要求入力[注1]	P10
INTP0		外部割り込み要求入力[注2]	P11
INTP1			P12
INTP2			P13
INTP3			P14/TI
INTP4			P15
INTP5			P16
TI		外部イベント・クロック入力	P14/INTP3
PWM	出　力	PWM出力	P20
TO00, TO01, TO20, TO21, TO30		タイマ・ユニット出力	P21-P25
TxD0		UART送信データ出力	P30/SB0/SO0
TxD1			P34/SO1
RxD0	入　力	UART受信データ入力	P31/SB1/SI0
RxD1			P35/SI1
TxC	出　力	UART送信クロック出力	P32/$\overline{SCK0}$
$\overline{CTS0}$	入　力	UART送信許可信号入力	P33
$\overline{CTS1}$			P36/$\overline{SCK1}$
SB0	入出力	SBI送受信データ入出力	P30/TxD0/SO0
SB1			P31/RxD0/SI0

注1．NMI割り込みはマスクできないため，有効エッジ検出により必ずNMI割り込みが起動されます（ポート1に対するリード動作時には，端子レベルが読み出されます）。

2．各割り込みをマスクまたは禁止（IE＝0）することにより，汎用入力ポートとして使用することができます。

保守／廃止

端子名称	入出力	機　　能	兼用端子
SO0	出　力	CSI送信データ出力	P30/TxD0/SB0
SO1			P34/TxD1
SI0	入　力	CSI受信データ入力	P31/RxD0/SB1
SI1			P35/RxD1
$\overline{SCK0}$	入出力	CSIシリアル・クロック入出力	P32/TxC
$\overline{SCK1}$			P36/$\overline{CTS1}$
PD0-PD7		パラレル・インタフェース　データ入出力	P40-P47
$\overline{DATASTB}$		パラレル・インタフェース　データ・ストローブ信号	P50
$\overline{ACK}$		パラレル・インタフェース　アクノリッジ信号	P51
BUSY		パラレル・インタフェース　ビジィ信号	P52
ANI0-ANI3	入　力	A/Dコンバータへのアナログ入力信号	P60-P63
RTP0-RTP7	出　力	リアルタイム出力ポート	P70-P77

保守／廃止

2.1.3　端子状態

各端子のリセット時，HOLD，およびスタンバイ・モードにおける端子状態を示します。

ポートとコントロール入出力の兼用端子は，リセット時にポートに切り替わります。

表2－1　ポート端子およびコントロール端子の状態（1/2）

端子名称	リセット時	コントロール・モードが指定された場合		
		HOLD	スタンバイ・モード	
			HALT	STOP
P00-P07	Hi-Z	（コントロール・モードの指定なし）出力は常に保持，入力はHi-Z		
P10/NMI		入力専用端子のため常にHi-Z		
P11/INTP0				
P12/INTP1				
P13/INTP2				
P14/INTP3/TI				
P15/INTP4				
P16/INTP5				
P20/PWM		継続	保持	
P21/TO00				
P22/TO01				
P23/TO20				
P24/TO21				
P25/TO30				
P30/TxD0/SB0/SO0		Type A	Type B	
P31/RxD0/SB1/SI0		Type A	Type B	
P32/TxC/$\overline{\text{SCK0}}$		Type A	Type B	
P33/$\overline{\text{CTS0}}$		Hi-Z	Hi-Z	
P34/TxD1/SO1		継続	保持	
P35/RxD1/SI1		Hi-Z	Hi-Z	
P36/$\overline{\text{CTS1}}$/$\overline{\text{SCK1}}$		Type A	Type B	

備考　継続　　：指定された動作を継続します。

保持　　：そのモードに移る直前の値を保持します。

Hi-Z　　：ハイ・インピーダンスを示します。

Type A：コントロール信号入力モードを選択したときはHi-Z，コントロール信号出力モードを選択したときは動作を継続します。

Type B：コントロール信号入力モードを選択したときはHi-Z，コントロール信号出力モードを選択したときは直前の値を保持します。

保守／廃止

表２－１　ポート端子およびコントロール端子の状態（2/2）

端　子　名　称	リセット時	コントロール・モードが指定された場合		
		HOLD	スタンバイ・モード	
			HALT	STOP
P40-P47/PD0-PD7	Hi-Z	Type A		Type B
P50/$\overline{\text{DATASTB}}$		Type A		Type B
P51/$\overline{\text{ACK}}$				
P52/BUSY				
P60-P63/ANI0-ANI3		Hi-Z		
P70-P77/RTP0-RTP7		継続	継続	保持
P80/DMARQ0		Hi-Z		
P81/DMARQ1				

備考　継続　　：指定された動作を継続します。

保持　　：そのモードに移る直前の値を保持します。

Hi-Z　　：ハイ・インピーダンスを示します。

Type A：コントロール信号入力モードを選択したときはHi-Z, コントロール信号出力モードを選択したときは動作を継続します。

Type B：コントロール信号入力モードを選択したときはHi-Z, コントロール信号出力モードを選択したときは直前の値を保持します。

保守／廃止

表２－２　ポート以外の端子の状態

端　子　名　称	リセット時	HOLD	HALT	STOP
$\overline{\text{RESET}}$	Hi-Z			
CLKOUT	継続	継続	継続	保持
$\overline{\text{ASTB}}$	Hi-Z	Hi-Z	継続	H
$\overline{\text{RD}}$				
$\overline{\text{WRL}}$				
$\overline{\text{WRH}}$				
READY	Hi-Z			
$\overline{\text{DEX}}$	Hi-Z	Hi-Z	継続	H
$\overline{\text{RAS}}$	Hi-Z	Hi-Z	継続	H
$\overline{\text{REFRQ}}$	H		継続	H
D8/$\overline{\text{D16}}$	Hi-Z			
$\overline{\text{BUSLOCK}}$	Hi-Z	Hi-Z	H	H
$\overline{\text{POLL}}$	Hi-Z			
HLDRQ	Hi-Z			
$\overline{\text{HLDAK}}$	Hi-Z	継続	継続	H
AD0-AD15	Hi-Z	Hi-Z	継続	保持
A16-A23	Hi-Z	Hi-Z	継続	注意
$\overline{\text{IORD}}$	Hi-Z	Hi-Z	H	H
$\overline{\text{IOWR}}$				
$\overline{\text{WDTOUT}}$	継続	継続	継続	H
$\overline{\text{DMAAK0}}$	Hi-Z	H	継続	保持
$\overline{\text{DMAAK1}}$				
$\overline{\text{TCE0}}$				
$\overline{\text{TCE1}}$				

備考　継続：指定された動作を継続します。

保持：そのモードに移る直前の値を保持します。

H　：ハイ・レベルを示します。

Hi-Z：ハイ・インピーダンスを示します。

注意　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。

保守／廃止

2.2　端子機能の説明

2.2.1　P00-P07（Port 0）……３ステート入出力

８ビットの汎用入出力ポートです。すべて入出力ポートとして使用できます。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に設定できます。

表２-３　ポート０の動作（n=0-7）

	ポート・モード	
	PM0n=1	PM0n=0
P00	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	入力ポート	出力ポート

2.2.2　P10-P16（Port 1）……入力

ポート１および割り込み入力端子として働く７ビットの特殊入力端子です。P10-P16は入力専用ポートとして機能するほか，外部割り込み入力または，タイマ・ユニットへのキャプチャ・トリガ入力端子，A/Dコンバータ・トリガ入力端子として機能します。

このポートは割り込み入力端子と兼用となっていますが，割り込み機能と入力ポートの機能を切り替えることはできません。

入力ポート機能のみを使用したいときは該当する割り込みおよびキャプチャ動作をマスクしてください（システム・リセットによりNMI以外，自動的にマスク状態，およびキャプチャ動作禁止状態になります）。

表２-４　ポート１の動作

	機能	
P10	NMI入力	兼　入力専用ポート
P11	INTP0入力	兼　入力専用ポート
P12	INTP1入力	兼　入力専用ポート
P13	INTP2入力	兼　入力専用ポート
P14	INTP3入力／TI入力	兼　入力専用ポート
P15	INTP4入力	兼　入力専用ポート
P16	INTP5入力	兼　入力専用ポート

保守／廃止

(a) NMI (Non-Maskable Interrupt) ……P10兼用入力

ソフトウエアによってマスク不可能なノンマスカブル割り込み要求入力端子です。ノンマスカブル割り込みはCPUがいつでも受け付ける割り込みであり，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，有効エッジは割り込みモード・レジスタ（INTM）で指定します。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のベクタ割り込みが発生します。このNMI入力はスタンバイ・モードの解除にも使用されます。なお，NMI入力状態はP10を読み出すことによりモニタできます。

(b) INTP0-INTP5 (Interrupt from Peripheral 0-5)

……P11-P16兼用入力，P14/TI兼用入力

ソフトウエアによってマスク可能なマスカブル割り込み要求入力端子です。INTPn(n＝0-5)入力はエッジで検出され，有効エッジは割り込みモード・レジスタ（INTM）で指定します。このINTPn入力は，タイマ・ユニットのキャプチャ・レジスタのラッチ・タイミング入力となるほか，HALTモードの解除にも使用されます。

また，INTP5はパラレル・インタフェース機能においてセントロニクス仕様のインタフェースとして使用する際のINIT信号入力などに使用できます。

(c) TI (Timer Event Clock) ……P14/INTP3兼用入力

TI端子はタイマ１の外部イベント・クロック入力端子です。

2.2.3　P20-P25 (Port 2) …… ３ステート入出力

ポート２および各種コントロール信号として働く６ビット入出力の兼用端子です。

P20-P25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）によって，ビット単位でポート／コントロール・モードの選択ができます。

表２-５　ポート２の動作（n＝0-5）

	コントロール・モード	ポート・モード	
	PMC2n＝1	PMC2n＝0	
		PM2n＝1	PM2n＝0
P20	PWM出力	入力ポート	出力ポート
P21	TO00出力	入力ポート	出力ポート
P22	TO01出力	入力ポート	出力ポート
P23	TO20出力	入力ポート	出力ポート
P24	TO21出力	入力ポート	出力ポート
P25	TO30出力	入力ポート	出力ポート

保守／廃止

(1) ポート・モード

P20-P25端子は，ポート２モード・レジスタ（PM2）により，ビット単位で入力または出力ポートに設定できます。

(2) コントロール・モード

P20-P25端子はPMC2レジスタにより，ビット単位にコントロール・モードに設定することができます（PMC2n＝1；n＝0-5)。

(a) PWM（Pulse Width Modulation）……P20兼用出力

PWMユニットからのパルス信号を出力する端子です。

(b) TO00，TO01，TO20，TO21，TO30（Timer Output）……P21-P25兼用出力

３つのタイマ（TM0，TM2，TM3）それぞれからのタイマ出力用の端子です。

2.2.4　P30-P36（Port 3）……３ステート入出力

ポート３および各種コントロール信号として働く７ビット入出力の兼用端子です。

P30-P36端子の動作モードは，ポート３モード・コントロール・レジスタ（PMC3）によりビット単位でポート／コントロール・モードの選択ができます。

表２-６　ポート３の動作（n＝0-6）

	コントロール・モード	ポート・モード	
	PMC3n＝1	PMC3n＝0	
		PM3n＝1	PM3n＝0
P30	TxD0出力/SB0入出力/SO0出力	入力ポート	出力ポート
P31	RxD0入力/SB1入出力/SI0入力	入力ポート	出力ポート
P32	TxC出力/$\overline{\text{SCK0}}$入出力	入力ポート	出力ポート
P33	$\overline{\text{CTS0}}$入力	入力ポート	出力ポート
P34	TxD1出力/SO1出力	入力ポート	出力ポート
P35	RxD1入力/SI1入力	入力ポート	出力ポート
P36	$\overline{\text{CTS1}}$入力/$\overline{\text{SCK1}}$入出力	入力ポート	出力ポート

(1) ポート・モード

P30-P36端子は，ポート３モード・レジスタ（PM3）により，ビット単位で入力または出力ポートに設定できます。

保守／廃止

(2) コントロール・モード

P30-P36端子はPMC3レジスタにより，ビット単位にコントロール・モードに設定することができます（PMC3n＝1；n＝0-6）。

また，コントロール・モードにおいての兼用の機能については，各シリアル・インタフェース・プロトコルを選択した際，端子機能が決定します。

(a) **TxD0，TxD1（Transmit Data 0，1）……P30，P34兼用出力**

UARTのch0，1の送信データ出力端子です。

(b) **RxD0，RxD1（Receive Data 0，1）……P31，P35兼用入力**

UARTのch0，1の受信データ入力端子です。

(c) **SB0，SB1（Serial Bus 0，1）……P30，P31兼用入出力**

SBIモードでのch0，1の送受信データ入出力端子です。

(d) **SI0，SI1入力（Serial Input 0，1）……P31，P35兼用入力**

3線式シリアルI/Oインタフェース・モードでのch0，1の受信データ入力端子です。

(e) **SO0，SO1出力（Serial Output 0，1）……P30，P34兼用出力**

3線式シリアルI/Oインタフェース・モードでのch0，1の送信データ出力端子です。

(f) **TxC（Transmit Clock）……P32兼用出力**

UARTの送信クロックの出力端子です。

(g) **$\overline{\text{SCK0}}$，$\overline{\text{SCK1}}$（Serial Clock 0，1）……P32，P36兼用入出力**

クロック同期式シリアル・インタフェース（SBIモード，3線式I/Oインタフェース・モード）のch0，1のシリアル・クロック入出力端子です。

(h) **$\overline{\text{CTS0}}$，$\overline{\text{CTS1}}$（Clear to Send 0，1）……P33，P36兼用入力**

UARTでのch0，1のCTS端子です。送信許可状態にする入力端子です。

保守／廃止

2.2.5　P40-P47（Port 4）……3ステート入出力

ポート4およびパラレル・インタフェースのデータ入出力端子として働く8ビット入出力の兼用端子です。

P40-P47端子の動作モードは，ポート4モード・コントロール・レジスタ（PMC4）により8ビット単位でポート／コントロール・モードの選択ができます。

表2-7　ポート4の動作（n=0-7）

	コントロール・モード	ポート・モード	
	$PMC4_7=1$	$PMC4_7=0$	
		PM4n=1	PM4n=0
P40	PD0入出力	入力ポート	出力ポート
P41	PD1入出力	入力ポート	出力ポート
P42	PD2入出力	入力ポート	出力ポート
P43	PD3入出力	入力ポート	出力ポート
P44	PD4入出力	入力ポート	出力ポート
P45	PD5入出力	入力ポート	出力ポート
P46	PD6入出力	入力ポート	出力ポート
P47	PD7入出力	入力ポート	出力ポート

(1) ポート・モード

P40-P47端子は，ポート4モード・レジスタ（PM4）により，ビット単位で入力または出力ポートに設定できます。

(2) コントロール・モード

P40-P47端子は，PMC4レジスタにより，8ビット単位でコントロール・モードに設定することができます（$PMC4_7=1$）。

(a) PD0-PD7（Parallel Data 0-7）……P40-P47兼用入出力

パラレル・インタフェースの送受信データ入出力端子です。

保守／廃止

2.2.6　P50-P52（Port 5）……３ステート入出力

ポート５およびパラレル・インタフェース制御信号として働く３ビット入出力の兼用端子です。

P50-P52端子の動作モードは，ポート５モード・コントロール・レジスタ（PMC5）によりビット単位でポート／コントロール・モードの選択ができます。

表２-８　ポート５の動作（n=0-2）

	コントロール・モード	ポート・モード	
	PMC5n＝1	PMC5n＝0	
		PM5n＝1	PM5n＝0
P50	$\overline{\text{DATASTB}}$入出力	入力ポート	出力ポート
P51	$\overline{\text{ACK}}$入出力	入力ポート	出力ポート
P52	BUSY入出力	入力ポート	出力ポート

(1) ポート・モード

P50-P52端子は，ポート５モード・レジスタ（PM5）により，ビット単位で入力または出力ポートに設定できます。

(2) コントロール・モード

P50-P52端子はPMC5レジスタにより，ビット単位でコントロール・モードに設定することができます（PMC5n＝1；n=0-2）。

(a) $\overline{\text{DATASTB}}$（Data Strobe）……P50兼用入出力

パラレル・インタフェース機能におけるストローブ信号入出力端子です。

入力モード時

受信データをパラレル・インタフェース・バッファ（PAB）に書き込むためのストローブ信号入力端子です。

この信号の立ち上がりエッジまたは立ち下がりエッジに同期して，PD0-PD7上のデータがPABレジスタに取り込まれ，BUSY信号がハイ・レベルになります。エッジは，パラレル・インタフェース・コントロール・レジスタ０（PAC0）の設定により指定します。

出力モード時

送信データを外部バッファ・レジスタなどにラッチさせるためのストローブ信号出力端子です。

保守／廃止

(b) $\overline{\text{ACK}}$ (Acknowledge) ……P51兼用入出力

パラレル・インタフェース機能において，PABレジスタへの送受信完了を示すアクノリッジ信号の入出力端子です。

入力モード時

PABレジスタ中の受信データを，DMAまたは命令により読み出したことを外部に知らせるアクノリッジ信号の出力端子です。

BUSY信号とのタイミング関係により，出力タイミングを３種類から選択することができます。

出力モード時

受信完了を示すアクノリッジ信号の入力端子です。

$\overline{\text{ACK}}$端子＝“ハイ・レベル→ロウ・レベル”の変化を検出します。

(c) BUSY (Busy) ……P52兼用入出力

パラレル・インタフェース機能において，送受信不許可状態であることを示すBUSY信号の入出力端子です。

入力モード時

受信不許可状態であることを示すBUSY信号の出力端子です。

入力モードでの端子レベルでのセット／リセット条件を次に示します。

セット条件：次の条件のどれかを満たしたとき，端子レベルがハイ・レベルになります。

- $\overline{\text{DATASTB}}$信号入力（立ち上がりまたは立ち下がり同期）
- ソフトウエアでパラレル・インタフェース・コントロール・レジスタ（PAC1）のMB0，MB1ビットを“1×”に設定したとき
- PAC1レジスタのINITビット＝1のときに，外部割り込みINTP5が入力されたとき

リセット条件：次の条件をすべて満たしたとき，端子レベルがロウ・レベルになります。

- PABレジスタから受信データが読み出され（ステータス・レジスタ：IBF＝0）一定時間経過したとき（モードによって異なります）
- ソフトウエアでPAC1レジスタのMB0，MB1ビットを“00”に設定したとき

出力モード時

送信不許可状態であることを示すBUSY信号の入力端子です。

BUSY端子の“ハイ・レベル→ロウ・レベル”の変化を検出します。

保守／廃止

2.2.7　P60-P63（Port 6）……3ステート入力

ポート6およびA/Dコンバータのアナログ入力端子として働く4ビット入力の兼用端子です。

このポートはアナログ入力端子と兼用になっていますが，入力ポートとアナログ入力端子との切り替えはできません。

表2-9　ポート6の動作

	機能	
P60	ANI0入力	兼　入力専用ポート
P61	ANI1入力	兼　入力専用ポート
P62	ANI2入力	兼　入力専用ポート
P63	ANI3入力	兼　入力専用ポート

2.2.8　P70-P77（Port 7）……3ステート入出力

ポート7およびリアルタイム出力ポートとして働く8ビット入出力の兼用端子です。

P70-P77端子の動作モードは，ポート7モード・コントロール・レジスタ（PMC7）によりビット単位でポート／コントロール・モードの選択ができます。

表2-10　ポート7の動作（n=0-7）

	コントロール・モード	ポート・モード	
	PMC7n=1	PMC7n=0	
		PM7n=1	PM7n=0
P70	RTP0出力	入力ポート	出力ポート
P71	RTP1出力	入力ポート	出力ポート
P72	RTP2出力	入力ポート	出力ポート
P73	RTP3出力	入力ポート	出力ポート
P74	RTP4出力	入力ポート	出力ポート
P75	RTP5出力	入力ポート	出力ポート
P76	RTP6出力	入力ポート	出力ポート
P77	RTP7出力	入力ポート	出力ポート

(1) ポート・モード

P70-P77端子は，ポート7モード・レジスタ（PM7）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。

保守／廃止

(2) コントロール・モード

P70-P77端子はPMC7レジスタにより，ビット単位でコントロール・モードに設定することができます（PMC7n＝1；n＝0-7）。

(a) RTP0-RTP7 (Realtime Output Port 0-7) ……P70-P77兼用出力

リアルタイム出力ポートの出力端子です。RTP0-RTP3がリアルタイム出力ポートのバッファであるP7Lレジスタに，RTP4-RTP7がP7Hレジスタに対応し，4ビット単位での出力も可能です。

2.2.9 P80, P81 (Port 8) ……3ステート入出力

ポート8およびDMAリクエスト信号入力端子として働く2ビット入出力の兼用端子です。

P80，P81端子の動作モードは，ポート8モード・コントロール・レジスタ（PMC8）によりビット単位でポート／コントロール・モードの選択ができます。

表2-11　ポート8の動作（n＝0，1）

	コントロール・モード	ポート・モード	
	PMC8n＝1	PMC8n＝0	
		PM8n＝1	PM8n＝0
P80	DMARQ0入力	入力ポート	出力ポート
P81	DMARQ1入力	入力ポート	出力ポート

(1) ポート・モード

P80，P81端子は，ポート8モード・レジスタ（PM8）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。

(2) コントロール・モード

P80，P81端子はPMC8レジスタにより，ビット単位にコントロール・モードに設定することができます（PMC8n＝1；n＝0，1）。

(a) DMARQ0, DMARQ1 (DMA Request 0, 1) ……P80, P81兼用入力

DMAコントローラ（チャネル0，1）の外部DMA要求入力端子です。

ディマンド・リリース・モードではレベルでセンスされますが，そのほかの転送モードでは立ち上がりエッジ・トリガとなります。

2.2.10 GND (Ground)

全GND端子をGND電位に固定してください。

保守／廃止

2.2.11　V_{DD} (Power Supply)

正電源供給端子です。全V_{DD}端子をV_{DD}に接続してください。

2.2.12　AV_{SS} (Analog V_{SS})

A/Dコンバータの GND端子です。GND電位に固定してください。

2.2.13　AV_{DD} (Analog V_{DD})

A/Dコンバータの正電源供給端子です。

2.2.14　AV_{REF} (Analog Reference Voltage)

A/Dコンバータの基準電圧入力端子です。

2.2.15　$\overline{\text{RESET}}$ (Reset) ……入力

$\overline{\text{RESET}}$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・レベル幅を持つ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(HALT/STOP)モードの解除にも使用されます。

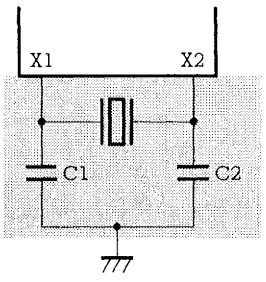
$\overline{\text{RESET}}$はアクティブ・ロウの入力端子です。

2.2.16　X1，X2 (Crystal)

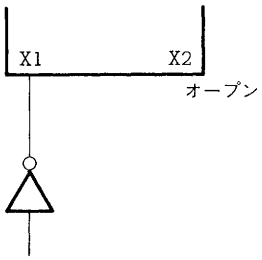
システム・クロック発生用クリスタル振動子／セラミック発振子の接続端子です。外部クロックを入力することも可能です。

外部クロックを入力する場合は，X1端子に入力し，X2端子はオープンにしてください。

(a) クリスタル／セラミック注発振



(b) 外部クロック入力



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．▨の範囲にほかの信号線を通さないでください。

注　クリスタル振動子／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ください。

保守／廃止

2.2.17　CLKOUT（Clock Out）……出力

クロック発生回路で設定した周波数で，CPUおよび周辺ハードウエアへの各種クロックを供給する内部システム・クロックの出力端子です。

2.2.18　$\overline{\text{ASTB}}$（Address Strobe）……出力

$\overline{\text{ASTB}}$端子は，外部バスのアドレス情報を外部ラッチするためのストローブ信号を出力する端子です。

2.2.19　$\overline{\text{RD}}$（Memory Read Strobe）……出力

$\overline{\text{RD}}$端子は，外部バスに接続されているメモリからの読み出しサイクルにおいてアクティブとなる，リード・ストローブ信号出力端子です。

2.2.20　$\overline{\text{WRL}}$（Memory Write Low Strobe）……出力

$\overline{\text{WRL}}$端子は，外部バスに接続されているメモリへの書き込みサイクルでアクティブとなる，下位８ビット・データのライト・ストローブ信号を出力する端子です。

この信号がアクティブとなるバス・サイクルは次のとおりです。

(1) 外部データ・バス幅16ビット時

- 偶数アドレスに対するバイト・データのアクセス，または奇数アドレスに対するワード・データのための２回目のバイト・アクセスによるバス・サイクル
- 偶数アドレスに対するワード・データのアクセスによるバス・サイクル

(2) 外部データ・バス幅８ビット時

- 外部メモリへのすべてのライト・サイクル

2.2.21　$\overline{\text{WRH}}$（Memory Write High Strobe）……出力

$\overline{\text{WRH}}$端子は，外部バスに接続されているメモリへの書き込みサイクルでアクティブとなる，上位８ビット・データのライト・ストローブ信号を出力する端子です。

外部データ・バス幅が16ビットのときのみ有効です（８ビット時には常にインアクティブ状態となります）。

保守／廃止

表2-12　メモリ・ライト・サイクルにおける$\overline{WRL}$, $\overline{WRH}$端子の動作

（外部データ・バス幅16ビット時）

		$\overline{WRL}$	$\overline{WRH}$
バイト・アクセス	偶数アドレス	アクティブ	インアクティブ
	奇数アドレス	インアクティブ	アクティブ
ワード・アクセス	偶数アドレス	アクティブ	アクティブ
	奇数アドレス[注]	2回目のバス・サイクルでだけアクティブ	1回目のバス・サイクルでだけアクティブ

注　2回のバス・サイクルを起動します。

2.2.22　READY（Ready）……入力

外部から外部バス・サイクルに対するウエイト・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　PWCレジスタによるウエイト設定が1以上のときのみ有効となります。

2.2.23　$\overline{DEX}$（Data Exchange）……出力

外部データ・バスの上位8ビット（D15-D8）に対するアクセスであるかどうかを示す信号端子です。T1サイクルのCLKOUT信号の立ち下がりで変化します。外部データ・バスの下位8ビット（D7-D0）に対してはA0により制御します。

アクセス		$\overline{DEX}$	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

注意　8ビット・バス幅のときは不定となりますので$\overline{DEX}$端子には何も接続しないでください。

2.2.24　$\overline{RAS}$（Row Address Strobe）……出力

外部バスに接続するDRAMにロウ・アドレス・ラッチ・タイミングを出力する端子です。

保守／廃止

2.2.25　D8/$\overline{\text{D16}}$（Data 8/16）……入力

外部データ・バス幅を８ビット，16ビットのどちらにするかを制御する入力端子です。

ロウ・レベルまたはハイ・レベルのどちらかに固定して使用します。

2.2.26　$\overline{\text{BUSLOCK}}$（Buslock）……出力

CPU以外のバス使用者（たとえばDMAコントローラやほかのプロセッサ）に対するバスの使用禁止を示します。BUSLOCKプリフィクス命令に続く１命令を実行中にアクティブとなります。

この信号がアクティブであるバス・ロック期間中は，バスに対してのCPU以外のいかなるバス要求（DMA転送要求，リフレッシュ要求を含む）も受け付けられません。

2.2.27　$\overline{\text{POLL}}$（Poll）……入力

$\overline{\text{POLL}}$入力信号は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。

この機能は，CPUのプログラムと外部デバイスとの同期をとるために利用されます。

2.2.28　HLDRQ（Hold Request）……入力

外部デバイスがV55PIに対して，外部バス（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。

HLDRQは，アクティブ・ハイ信号入力端子です。

2.2.29　$\overline{\text{HLDAK}}$（Hold Acknowledge）……出力

CPUがホールド要求信号（HLDRQ）を受けて，外部バスをハイ・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，外部バス（アドレス・バス，データ・バス，制御バス）は，ハイ・インピーダンス状態になります。

2.2.30　AD0-AD15（Address/Data 0-15）……入出力

外部バスのアドレス／データのマルチプレクス信号入出力端子です。

2.2.31　A16-A23（Address 16-23）……出力

外部バスのアドレス信号出力端子です。

2.2.32　$\overline{\text{IORD}}$（I/O Read）……出力

外部I/Oからの読み出しサイクルにおいてアクティブとなるリード・ストローブ信号出力端子です。

この信号はCPUによるI/Oリード・サイクルのほか，外部I/OからのDMA転送時にもアクティブになります。

$\overline{\text{IORD}}$は，アクティブ・ロウの出力端子です。

保守／廃止

2.2.33　$\overline{\text{IOWR}}$（I/O Write）……出力

外部I/Oへの書き込みサイクルにおいてアクティブとなるライト・ストローブ信号出力端子です。

この信号はCPUによるI/Oライト・サイクルのほか，外部I/Oに対するDMA転送時にもアクティブになります。

$\overline{\text{IOWR}}$は，アクティブ・ロウの出力端子です。

2.2.34　$\overline{\text{WDTOUT}}$（Watchdog Timer Output）……出力

ウォッチドッグ・タイマがノンマスカブル割り込みを発生したことを示す信号の出力端子です。

2.2.35　$\overline{\text{DMAAK0}}$, $\overline{\text{DMAAK1}}$（DMA Acknowledge 0, 1）……出力

DMAコントローラ（チャネル0，1）のDMAアクノリッジ信号出力端子です。外部DMA要求DMARQ0, DMARQ1に対して出力されるほか，内蔵タイマによりI/O-メモリ間のDMA転送が行われたときにもこれらの信号が出力されます。ただし，メモリ-メモリ間，メモリ-SFR（特殊機能レジスタ）間のDMA転送時には出力されません。これらはアクティブ・ロウ信号の出力端子です。

2.2.36　$\overline{\text{TCE0}}$, $\overline{\text{TCE1}}$（Terminal Count End 0, 1）……出力

DMAコントローラ（チャネル0，1）のDMA完了信号を出力する端子です。

転送対象に外部I/Oを指定しているときで，インテリジェントDMAモード-2，2チャネル動作モード（終了時停止），およびメモリ-メモリ転送モード（終了時停止）のときに有効です。それぞれの動作モードで，ターミナル・カウンタ（TC0，TC1）のダウン・カウントによりボローが発生したときに出力されます。その他の転送対象・動作モードのときには常にインアクティブです。これらはアクティブ・ロウ信号の出力端子です。

2.2.37　$\overline{\text{REFRQ}}$（Refresh Request）……出力

リフレッシュ・サイクルにおいてアクティブとなり，リフレッシュ・サイクルであることを外部に知らせる出力端子です。

$\overline{\text{REFRQ}}$は，アクティブ・ロウの出力端子です。

保守／廃止

2.3　端子の入出力回路

2.3.1　入出力回路のタイプ

(1) ポ ー ト

端子名称	タイプ	端子名称	タイプ
P00-P07	5	P32/TxC/$\overline{SCK0}$	2
P10/NMI	2	P33/$\overline{CTS0}$	
P11/INTP0		P34/TxD1/SO1	5
P12/INTP1		P35/RxD1/SI1	2
P13/INTP2		P36/$\overline{CTS1}$/$\overline{SCK1}$	
P14/INTP3/TI		P40-P47/PD0-PD7	5
P15/INTP4		P50/$\overline{DATASTB}$	
P16/INTP5		P51/$\overline{ACK}$	
P20/PWM	5	P52/BUSY	
P21/TO00		P60-P63/ANI0-ANI3	9
P22/TO01		P70-P77/RTP0-RTP7	5
P23/TO20		P80/DMARQ0	
P24/TO21		P81/DMARQ1	
P25/TO30			
P30/TxD0/SB0/SO0	2		
P31/RxD0/SB1/SI0			

(2) ポート以外

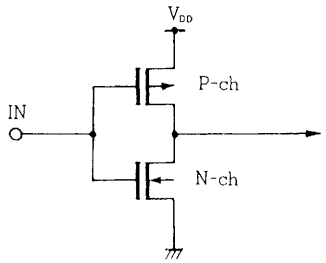
端子名称	タイプ	端子名称	タイプ
$\overline{RESET}$	2	AD0-AD15	5
READY	1	A16-A23	4
CLKOUT	3	$\overline{RAS}$	
$\overline{ASTB}$	4	$\overline{IORD}$	
$\overline{RD}$		$\overline{IOWR}$	
$\overline{WRL}$		D8/$\overline{D16}$	1
$\overline{WRH}$		$\overline{DEX}$	4
$\overline{BUSLOCK}$		$\overline{DMAAK0}$	
$\overline{HLDAK}$		$\overline{DMAAK1}$	
$\overline{WDTOUT}$		$\overline{TCE0}$	
HLDRQ	1	$\overline{TCE1}$	
$\overline{POLL}$		$\overline{REFRQ}$	

保守／廃止

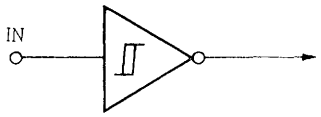
2.3.2　入出力回路

図2－1　タイプ別入出力回路

タイプ1

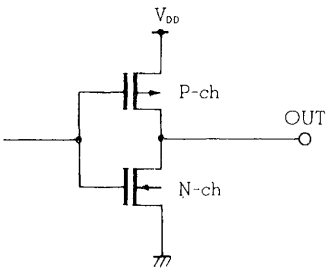


タイプ2

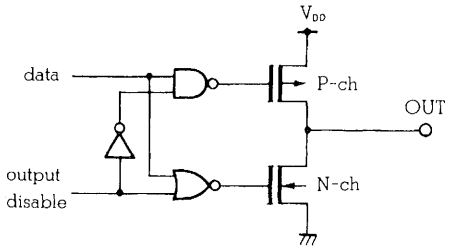


ヒステリシス特性を有するシュミット・トリガ入力となっています。

タイプ3

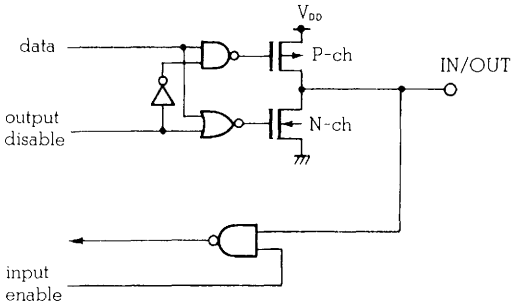


タイプ4

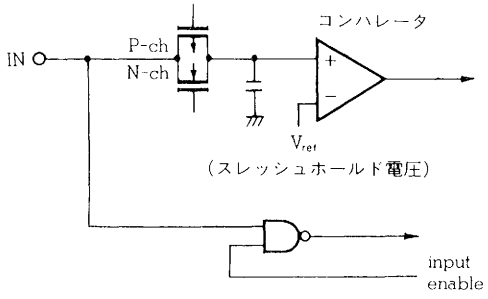


出力ハイ・インピーダンス（P-ch，N-chともにオフ）とすることができるプッシュプル出力となっています。

タイプ5



タイプ9



保守／廃止

2.4　未使用端子の処理

次に未使用端子の推奨接続方法を示します。

抵抗を介してV_{DD}またはGNDに接続する場合，1-10 kΩの抵抗を使用することをお奨めします。

(1) ポ ー ト

端子名称	推　奨　接　続　方　法
P00-P07	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
P10/NMI	抵抗を介してGNDに接続
P11/INTP0	抵抗を介してV_{DD}またはGNDに接続
P12/INTP1	
P13/INTP2	
P14/INTP3/TI	
P15/INTP4	
P16/INTP5	
P20/PWM	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
P21/TO00	
P22/TO01	
P23/TO20	
P24/TO21	
P25/TO30	
P30/TxD0/SB0/SO0	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
P31/RxD0/SB1/SI0	
P32/TxC/$\overline{SCK0}$	
P33/$\overline{CTS0}$	
P34/TxD1/SO1	
P35/RxD1/SI1	
P36/$\overline{CTS1}$/$\overline{SCK1}$	
P40-P47/PD0-PD7	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
P50/$\overline{DATASTB}$	
P51/$\overline{ACK}$	
P52/BUSY	
★ P60-P63/ANI0-ANI3	抵抗を介してGNDに接続
P70-P77/RTP0-RTP7	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
P80/DMARQ0	
P81/DMARQ1	

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。

保守／廃止

(2) ポート以外

端子名称	推　奨　接　続　方　法
CLKOUT	オープン
$\overline{\text{ASTB}}$	
$\overline{\text{RD}}$	
$\overline{\text{WRH}}$	
$\overline{\text{WRL}}$	
A16-A23	
$\overline{\text{DEX}}$	
$\overline{\text{RAS}}$	
$\overline{\text{HLDAK}}$	
$\overline{\text{BUSLOCK}}$	
$\overline{\text{WDTOUT}}$	
$\overline{\text{DMAAK0}}$	
$\overline{\text{DMAAK1}}$	
$\overline{\text{TCE0}}$	
$\overline{\text{TCE1}}$	
$\overline{\text{IORD}}$	
$\overline{\text{IOWR}}$	
$\overline{\text{REFRQ}}$	
HLDRQ	抵抗を介してGNDに接続
$\overline{\text{POLL}}$	
READY	抵抗を介してV_{DD}に接続
AD0-AD15	個別に抵抗を介してV_{DD}に接続
AV_{SS}	抵抗を介してGNDに接続
AV_{REF}	
AV_{DD}	抵抗を介してV_{DD}に接続

(3) その他

端子名称	推　奨　接　続　方　法
IC (H)	抵抗を介してV_{DD}に接続
IC (L)	抵抗を介してGNDに接続
OPEN	オープン

保守／廃止

第３章　CPU機能

V55PIは，V20，V30（ネイティブ・モード）およびV25，V35とオブジェクト・コード・レベルでソフトウエア・アップワード・コンパチブルなCPUを持っています。

3.1 特　　徴

○V20，V30（ネイティブ・モード）およびV25，V35とソフトウエア・アップワード・コンパチブル（追加命令あり）

○最小インストラクション・サイクル：160 ns/12.5 MHz（外部25 MHz時）
125 ns/16 MHz（外部32 MHz時）

○アドレス空間：16 Mバイト ― 基本メモリ(プログラム／データ／スタック)空間　1 Mバイト
└ 拡張メモリ(データ)空間　16 Mバイト

○レジスタ・ファイル空間（内蔵RAM上）：512バイト/16レジスタ・バンク

○I/O空間：64 Kバイト

○レジスタ構成（V20，V30およびV25，V35との比較）

項　　目		V20，V30	V25，V35	**V55PI**
拡張セグメント・レジスタ		な　し	な　し	DS2，DS3
レジスタ・バンク		な　し	8バンク（メモリ空間上）	16バンク（レジスタ・ファイル空間上）
PSW	モード・フラグ	MD	な　し	な　し
	レジスタ・バンク用フラグ	な　し	RB0-RB2	RB0-RB3
	入出力命令トラップ用フラグ	な　し	$\overline{\text{IBRK}}$	$\overline{\text{IBRK}}$
	ユーザ・フラグ	な　し	F0，F1	な　し
特殊機能レジスタ領域		な　し	240バイト（FFF00H-FFFEFHにメモリ・マッピング）	496バイト（FFE00H-FFFEFHにメモリ・マッピング）

○内部16ビット・アーキテクチャ，外部16/8ビット・データ・バス幅選択可能

○メモリを可変サイズ（最大６ブロック）で分割して自動ウエイト制御

●プログラマブル・ウエイト機能

●READY端子によるウエイト機能

○リフレッシュ機能

リフレッシュ・サイクルを自動生成（$\overline{\text{REFRQ}}$出力，$\overline{\text{RAS}}$出力）

保守／廃止

3.2　レジスタ

V55PIのCPUは，V20，V30（ネイティブ・モード）およびV25，V35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。汎用のレジスタ・セットは，レジスタ・ファイル空間にマッピングされています。この汎用のレジスタ・セットは内蔵RAMと兼用になっており，内蔵RAM上に最大16のレジスタ・セットをバンク形式で持つことができます。

また，そのほかに内蔵周辺ハードウエアの制御用に種々の特殊機能レジスタを持っています。この特殊機能レジスタはメモリ空間のOFFE00H-OFFFEFHにマッピングされています。

3.2.1　レジスタ・バンク

汎用レジスタ・セットは，レジスタ・ファイル空間（内蔵RAM上）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，1バンクあたり32バイト使用し16バンクまで設定可能です。

通常CPUは，レジスタ・バンク15を用いてプログラムを実行し，マスカブルなハードウエア割り込みやソフトウエア割り込み（BRKCS命令）によってほかのレジスタ・バンクへ自動的に切り替えることができます。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）によって行えます。

レジスタ・バンクの構成は，図3-1のようになっています。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。レジスタ・バンク中の先頭からのワード領域は，拡張セグメント・レジスタ（DS2）の領域となります。ベクタPC/DS3領域は，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値をセットするための領域です。この領域は拡張セグメント・レジスタ（DS3）の領域としても使用されます。PSW退避領域は，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。PC退避領域は，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。

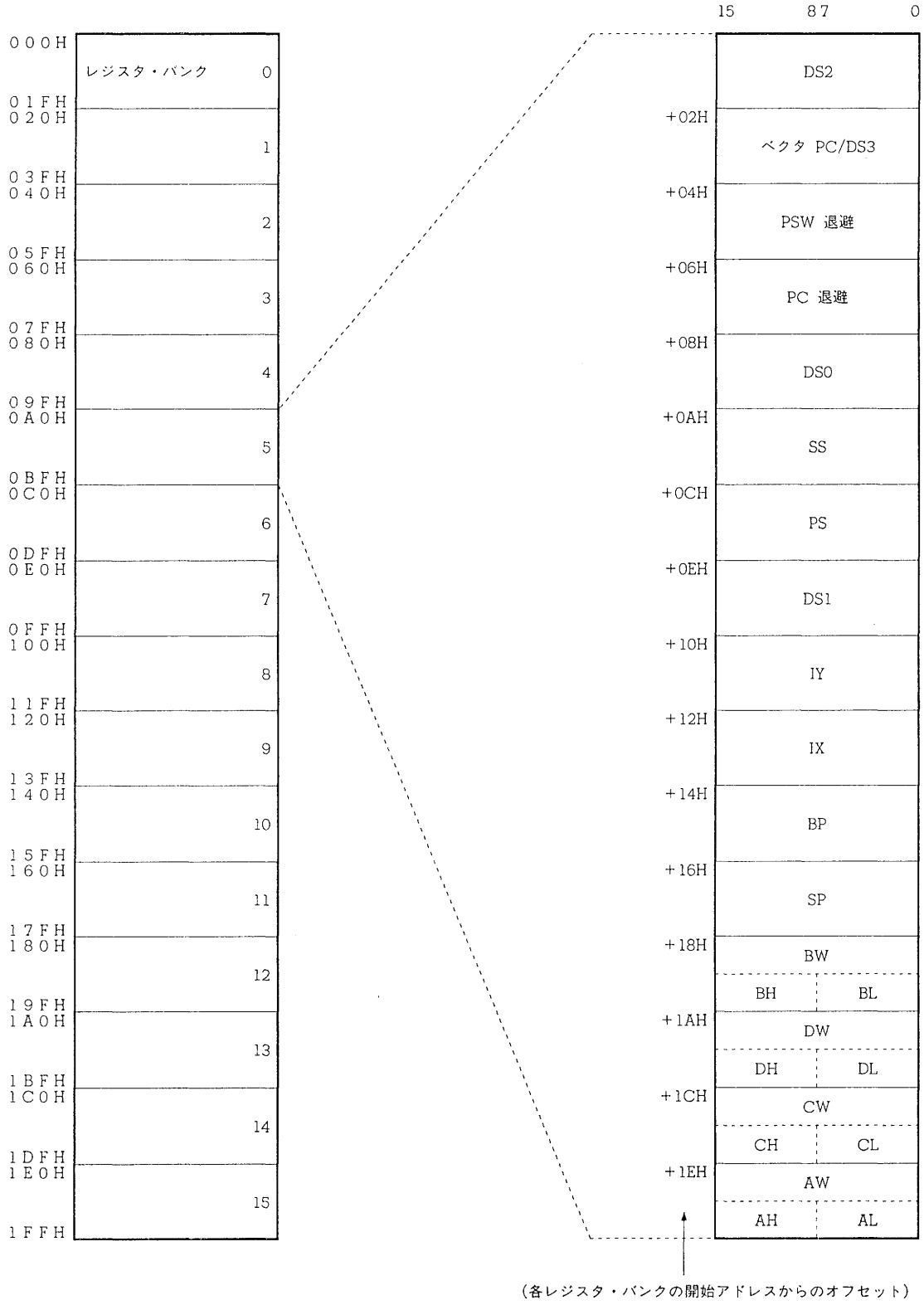
リセット後には，レジスタ・バンク15が自動的に選択されます。また，リセット後のセグメント・レジスタの初期化はレジスタ・バンク15のみ行います。

また，この汎用レジスタ・セットのマッピングされているレジスタ・ファイル空間は，メモリ操作命令に専用プリフィクス命令（IRAM：）を付加してデータ・メモリとしてもアクセスできます。

そのほか16セットのレジスタ・バンクのうち，バンク0，バンク1はマクロ・サービス・チャネル（マクロ・サービスのためのパラメータおよびワーク領域）が重複して割り付けられています。

保守／廃止

図３－１　レジスタ・バンクの構成



保守／廃止

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な命令に対して，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ィフォールト・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，BCDローテート，データ変換
AH ：バイト乗算／除算
BW ：データ変換
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，ローテート命令，BCD演算
DW：ワード乗算／除算，間接アドレシング入力／出力

これらのレジスタは，レジスタ・ファイル空間（内蔵RAM上）にマッピングされています。そのアドレスは，（レジスタ・バンクの番号×32）にレジスタごとのオフセットを加えた値となります。

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）

ベースト・アドレシング（BP），インデクスト・アドレシング（IX，IY），ベースト・インデクスト・アドレシング（BP，IX，IY）などによるメモリ・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

保守／廃止

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おいて，固定的なアドレス・ポインタとして使用されます。

SP：スタック操作
IX ：プリミティブ・ブロック転送命令，BCD演算命令のソース側アドレス指定
IY ：プリミティブ・ブロック転送命令，BCD演算命令のデスティネーション側アドレス指定

これらのレジスタは，レジスタ・ファイル空間（内蔵RAM上）にマッピングされています。そのアドレスは，（レジスタ・バンクの番号×32）にレジスタごとのオフセットを加えた値となります。

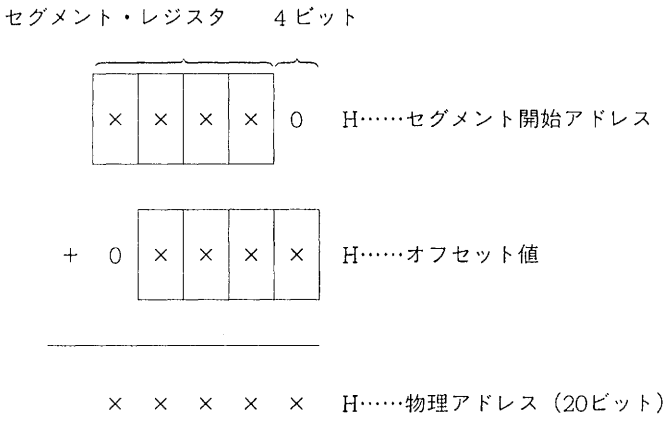
表3－2　ポインタとインデクス・レジスタのオフセット

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，1Mバイトの基本メモリ空間を64Kバイト単位の論理セグメントに分割して管理します。CPUは，各セグメントの開始アドレスをセグメント・レジスタで指示し，開始アドレスからの相対アドレスをオフセットとし，別のレジスタまたは実効アドレスで指示します。

物理アドレスは次に示すように作成されます。



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，次の場合に使用されます。

保守／廃止

PS　：プログラム・コード・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シング

DS0：一般変数のアクセス，プリミティブ・ブロック転送命令，BCD演算命令のソース・ブロック・データのアクセス

DS1：プリミティブ・ブロック転送命令，BCD演算命令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，一般変数のアクセスは，セグメント・オーバライド・プリフィクスを用いることにより，DS0からほかのセグメント・レジスタに変更して使用することが可能です。また，BPをベース・レジスタとしたアドレシングにおいても，同様にSSの代わりにほかのセグメント・レジスタを使用することが可能です。

例

```
MOV  AW, 1000H
MOV  DS1, AW
MOV  BL, DS1：BYTE PTR [IX]    ; DS1：IXからのバイト・データ・リード
```

リセット時には，レジスタ・バンク15のPSはFFFFHに，SS，DS0，DS1は0000Hに初期化されます。

これらのレジスタは，レジスタ・ファイル空間（内蔵RAM上）にマッピングされています。そのアドレスは，（レジスタ・バンクの番号×32）にレジスタごとのオフセットを加えた値となります。

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　拡張セグメント・レジスタ（DS2，DS3）

V55PIには，1Mバイトの基本メモリ空間をアクセスするためのセグメント・レジスタのほかに，16Mバイトの拡張メモリ空間を64Kバイト単位のセグメントに分割し，各セグメントの開始アドレスを指定する拡張セグメント・レジスタが設けられています。拡張セグメント・レジスタには，DS2（Data Segment2），DS3（Data Segment3）があり，以下の場合に使用されます。

保守／廃止

DS2：拡張メモリ空間の一般変数のアクセス（セグメント・オーバライド・プリフィクスによる），拡張メモリ空間に対するプリミティブ・ブロック転送命令，BCD演算命令のソース・ブロック・データのアクセス

DS3：拡張メモリ空間の一般変数のアクセス（セグメント・オーバライド・プリフィクスによる），拡張メモリ空間に対するプリミティブ・ブロック転送命令，BCD演算命令のデスティネーション・ブロック・データのアクセス

拡張セグメント・レジスタを使用したデータ・アクセスは，セグメント・オーバライド・プリフィクスを用いることにより行います。また，特にプリミティブ・ブロック転送命令，BCD演算命令においては，DS2とDS3を二重にセグメント・オーバライド・プリフィクスで指定することができます（この場合，DS2とDS3の指定順序は任意です）。

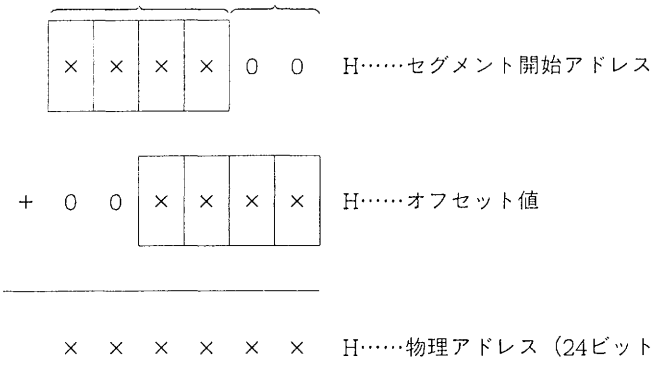
例

```
REP
DS2:
DS3:        MOVBKW ; DS2 : IXからDS3 : IYへのワード・メモリ・ブロック転送
```

各セグメントの開始アドレスを拡張セグメント・レジスタで指示し，開始アドレスからの相対アドレスをオフセットとし，別のレジスタまたはイミーディエト・データで指示することによりアクセスします。

物理アドレスは次に示すように作成されます。

拡張セグメント・レジスタ　８ビット



リセット時には，レジスタ・バンク15のDS2，DS3は0000Hに初期化されます。

これらのレジスタは，レジスタ・ファイル空間（内蔵RAM上）にマッピングされています。そのアドレスは，（レジスタ・バンクの番号×32）にレジスタごとのオフセットを加えた値となります。

保守／廃止

表３－４　拡張セグメント・レジスタのオフセット

レジスタ	オフセット
DS2	00H
DS3	02H（ベクタPCと兼用）

3.2.6　特殊機能レジスタ（SFR）

V55PIには，内蔵周辺ハードウエアの制御機能を持ったレジスタ群があります。

レジスタは周辺ハードウエアごとに制御内容によっていくつか用意されており，レジスタ内の各ビットによって具体的な動作が設定できます。これらのレジスタ群はメモリ空間上に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　特殊機能レジスタ領域**参照）。

例

```
MOV  AW, 0FFE0H
MOV  DS1, AW
MOV  BL, DS1：BYTE PTR [1EFH]    ；0FFE0H：1EFH（PRCレジスタ）の読み出し
```

また，特に特殊機能レジスタにのみ有効な命令としてBTCLR命令とBTCLRL命令があります。このうちBTCLRL命令はV25，V35に対して追加された命令です。

BTCLR命令は，特殊機能レジスタ領域の上位240バイト（0FFF00H-0FFFEFH）内のレジスタに対して有効であり，BTCLRL命令は下位256バイト（0FFE00H-0FFEFFH）内のレジスタに対して有効な命令です。

保守／廃止

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値を保持する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コードをフェッチするご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，ブレーク命令実行時などには，新たなロケーションのアドレス値がロードされます。

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は物理アドレス0FFFF0H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・フラグで構成されます。

○ステータス・フラグ

V（Overflow）	…オーバフローの検出フラグ
S（Sign）	…サイン・ビットの検出フラグ
Z（Zero）	…オール・ゼロの検出フラグ
AC（Auxiliary Carry）	…４ビット単位のキャリー，ボローの検出フラグ
P（Parity）	…パリティの検出フラグ
CY（Carry）	…キャリー，ボローの検出フラグ

○コントロール・フラグ

RB0-RB3（Register Bank0-3）	…レジスタ・バンクの指示フラグ
DIR（Direction）	…ブロック転送／入出力系命令の方向制御フラグ
IE（Interrupt Enable）	…割り込み許可状態の制御フラグ
BRK（Break）	…シングルステップ割り込みの制御フラグ
$\overline{\text{IBRK}}$（I/O Break）	…入出力命令トラップの制御フラグ

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，命令によって直接セット，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。

PSWの内容は，PUSH/POP命令によって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12-15（RB0-RB3）はPSWに戻りません。

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。

保守／廃止

図３－２　PSWのビット構成

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RB3	RB2	RB1	RB0	V	DIR	IE	BRK	S	Z	0	AC	0	P	$\overline{\text{IBRK}}$	CY

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ならリセットされ，０以外ならセットされます。

符号つ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ならリセットされ，０以外ならセットされます。

符号つ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・イミーディエト演算の場合は，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ならセットされ，０ならリセットされます。

保守／廃止

3.4.2　P（Parity Flag）

（1）２進加減算，論理演算，シフト

演算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ビットへのボ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進演算，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，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

保守／廃止

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ならリセットされ，０以外ならセットされます。

符号つ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ならリセットされ，０以外ならセットされます。

符号つ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・イミーディエト演算の場合は，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

演算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，演算結果により次のようになります。

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，演算結果により次のようになります。

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。

保守／廃止

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

入出力命令時のソフトウエ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とソフトウエア割り込み(割り込みベクタ６)が発生します。これにより，ソフトウエ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エ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命令実行するとソフトウエア割り込み(割り込みベクタ１)が発生します。これにより，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　RB0-RB3 (Register Bank0-3 Flag)

RB0-RB3で，内蔵RAM領域内に設定された16個のレジスタ・バンクの中の，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RB0-RB3の値を変えないでください。

保守／廃止

3.5　メモリ空間

V55PIは，16Mバイトのメモリ空間を持っています。このうち下位1Mバイト（000000H-0FFFFFH）を，基本メモリ空間として，また基本メモリ空間を含む16Mバイト（000000H-FFFFFFH）を拡張メモリ空間としてアクセスできます。基本メモリ空間は，V25，V35と同様にセグメント・レジスタ（PS，DS0，DS1，SS）を使用してアクセスすることができる空間です。拡張メモリ空間は拡張セグメント・レジスタ（DS2，DS3）を使用してアクセスすることができる空間で，最下位の1Mバイトには基本メモリ空間がマッピングされています。物理アドレスについては，**3.2.4　セグメント・レジスタ（PS，SS，DS0，DS1）**，**3.2.5　拡張セグメント・レジスタ（DS2，DS3）**を参照してください。

0FFE00H-0FFFEFHの496バイト空間内に内蔵周辺ハードウエアのモード・レジスタ，制御レジスタなどの機能を割り付けられたレジスタ群がマッピングされており，メモリ・アクセスによって操作します。

また，これらとは独立して，512バイトのレジスタ・ファイル空間（内蔵RAM上）があります。レジスタ・ファイル空間はV25，V35と同様にレジスタ操作命令を使用してアクセス可能なほか，メモリ操作命令に専用のプリフィクス命令（IRAM：）を付加してデータ・メモリとしてもアクセスすることができます。

セグメント内オフセットがFFFFH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，FFFFH＋1番地となります（V20，V30では，0000H番地が参照されます）。

3.5.1　基本メモリ空間

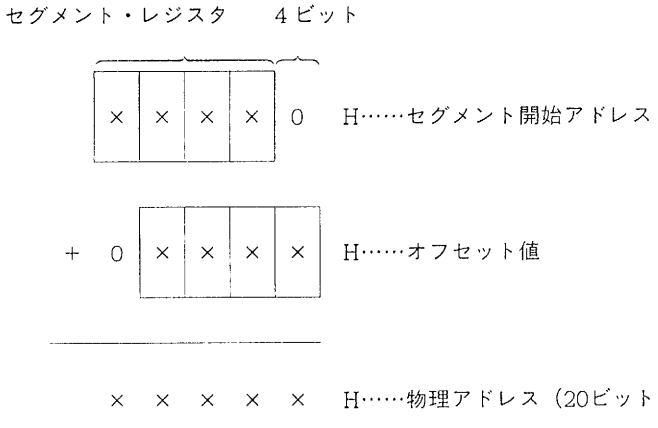
メモリ空間には1Mバイトの基本メモリ空間と16Mバイトの拡張メモリ空間があります。拡張メモリ空間の下位1Mバイト（000000H-0FFFFFH）に基本メモリ空間がマッピングされています。

図３-３に1Mバイトの基本メモリ空間を示します。

ソフトウエアによる基本メモリ空間へのアクセス条件はV20，V30およびV25，V35と同様です。

基本メモリ空間の物理アドレスは，セグメント・レジスタPS，SS，DS0，DS1で示されるセグメント開始アドレスと，ほかのレジスタまたはイミーディエト・データによって示されるセグメント開始位置からのオフセット値によって指定されます。

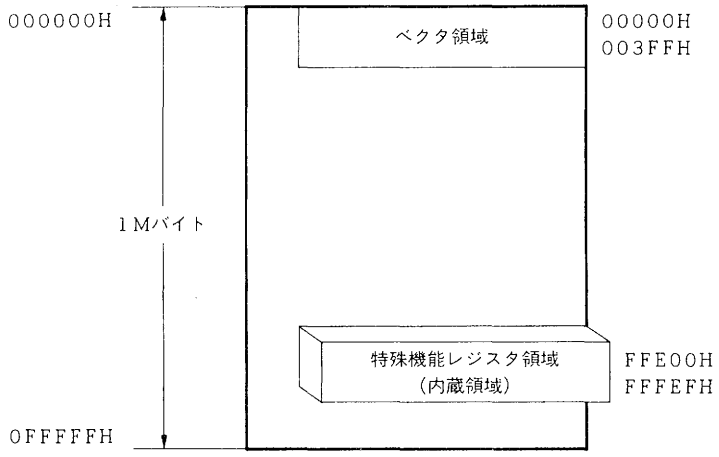
1Mバイトの基本メモリ空間の物理アドレスは次に示すように生成されます。



保守／廃止

また，基本メモリ空間内には，ベクタ割り込み用のベクタ領域と特殊機能レジスタ領域がマッピングされています。特殊機能レジスタがマッピングされている領域と重複する外部メモリ領域に対するメモリ・アクセスはできません。

図3－3　基本メモリ空間



なお，0FFFF0H-0FFFFFHはシステム・ブート用プログラム領域となっており，リセット解除後PSは0FFFFH，PCは0Hとなるため，0FFFF0H番地からプログラム実行を開始します。

3.5.2　拡張メモリ空間

図3－4に16Mバイトの拡張メモリ空間を示します。

ソフトウエアによる拡張メモリ空間へのアクセスは，データ・アクセスのみ可能です。

ただし，最下位1Mバイト（000000H-0FFFFFH）は基本メモリ空間がマッピングされており，セグメント・レジスタPS，SS，DS0，DS1を使用してアクセスすることができます。

拡張メモリ空間では，拡張セグメント・レジスタDS2，DS3を用いてデータ・アクセスができます。DS2，DS3は拡張セグメント・オーバライド・プリフィクスとして，メモリ操作命令に付加して指定できます。

拡張メモリ空間の物理アドレスは，拡張セグメント・レジスタで示されるセグメントの開始アドレスとほかのレジスタまたはイミーディエト・データによって示されるセグメント開始位置からのオフセット値によって指定されます。生成されたアドレスが最下位1Mバイト（000000H-0FFFFFH）を指す場合は，基本メモリ空間をアクセスします。

16Mバイトの拡張メモリ空間の物理アドレスは次に示すように生成されます。

保守／廃止

拡張セグメント・レジスタ　８ビット

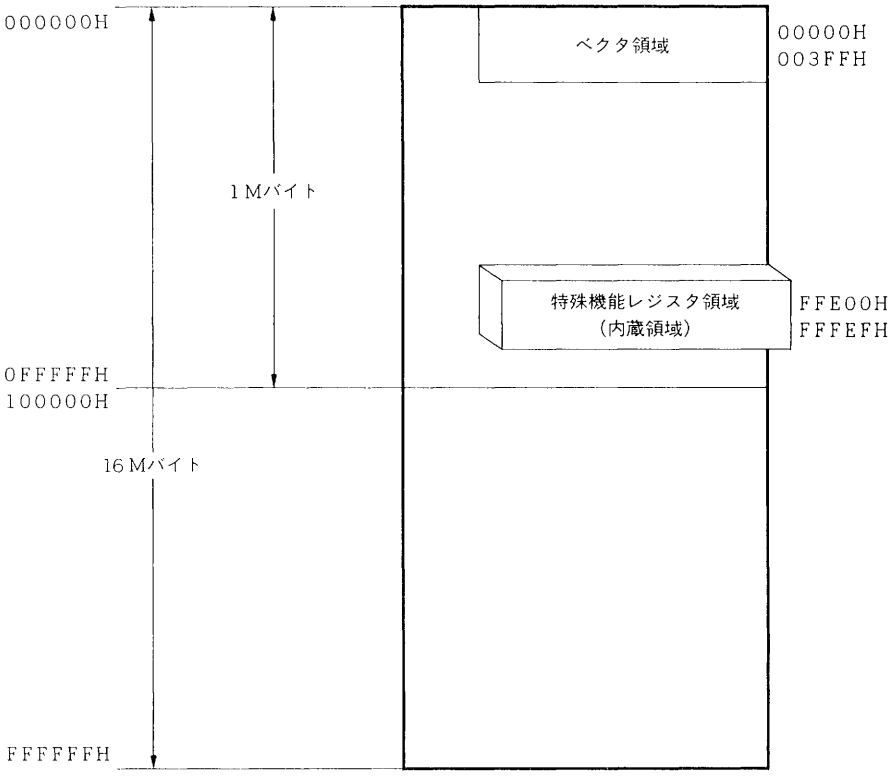
×	×	×	×	0	0	H……セグメント開始アドレス

＋

0	0	×	×	×	×	H……オフセット値

×	×	×	×	×	×	H……物理アドレス（24ビット）

図３－４　拡張メモリ空間



保守／廃止

3.5.3　特殊機能レジスタ領域

0FFE00H-0FFFEFHの496バイト空間内に，内蔵周辺ハードウエアの動作モード指定，状態モニタリングなどの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。

特殊機能レジスタの操作はメモリ操作命令によるアクセスで行います。

特殊機能レジスタ領域をアクセスしたときは，アドレス・バスにアクセス中のアドレスが出力されます。特殊機能レジスタへのライト・アクセス時は，アドレス出力後にライト・データが出力されます。リード・アクセス時はアドレスのみ出力され，その後ハイ・インピーダンス状態になります。

特殊機能レジスタ領域をアクセスしたときは，$\overline{RD}$，$\overline{WRH}$，$\overline{WRL}$，$\overline{IORD}$，$\overline{IOWR}$などの制御信号はアクティブになりません。

表３-５に特殊機能レジスタの一覧を示します。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したものです。メモリ操作命令のオペランド記述形式（シンボル名）に相当します。

R/W ……その特殊機能レジスタが，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，１ビット操作，８ビット操作，16ビット操作，32ビット操作のうち可能なものを示します。

RESET時……$\overline{RESET}$入力時の各レジスタの状態を示します。

注意　記載されていないアドレスの部分は予約領域ですので，ユーザ・プログラムではアクセスしないでください。

備考　21ビット(24ビット)特殊機能レジスタの下位16ビットは，ワード・アクセスによるリード／ライトのみ可能です。上位５ビット（８ビット）は，バイト・アクセス，ワード・アクセスのどちらも可能です（ワード・アクセス時，上位８ビットは不定値となります)。

保守／廃止

表３－５ 特殊機能レジスター覧（1/8）

アドレス	特殊機能レジスタ名称	略 号		R/W	操作可能ビット単位				リセット時
					1ビット	8ビット	16ビット	32ビット	
0FFE00H	A/Dコンバージョン・リザルト・レジスタ0	ADCR0		R		○			不定
0FFE02H	A/Dコンバージョン・リザルト・レジスタ1	ADCR1		R		○			不定
0FFE04H	A/Dコンバージョン・リザルト・レジスタ2	ADCR2		R		○			不定
0FFE06H	A/Dコンバージョン・リザルト・レジスタ3	ADCR3		R		○			不定
0FFE10H	パラレル・インタフェース・バッファ	PAB		R/W注1		○			不定
0FFE18H	パラレル・インタフェース・コントロール・レジスタ0	PAC0		R/W	○	○			90H
0FFE19H	パラレル・インタフェース・コントロール・レジスタ1	PAC1		R/W	○	○			03H
0FFE1AH	パラレル・インタフェース・ステータス・レジスタ	PAS		R/W注2		○			40H
0FFE1CH	パラレル・インタフェース・アクノリッジ・インターバル・レジスタ1	PAI1		W		○			不定
0FFE1DH	パラレル・インタフェース・アクノリッジ・インターバル・レジスタ2	PAI2		W		○			不定
0FFE20H	A/Dコンバータ・モード・レジスタ	ADM		R/W	○	○			00H
0FFEC0H	割り込みマスク・フラグ・レジスタ0（下位）	MK0	MK0L	R/W	○	○	○		FFH
0FFEC1H	割り込みマスク・フラグ・レジスタ0（上位）		MK0H	R/W	○	○			FFH
0FFEC2H	割り込みマスク・フラグ・レジスタ1（下位）	MK1	MK1L	R/W	○	○	○		FFH
0FFEC3H	割り込みマスク・フラグ・レジスタ1（上位）		MK1H	R/W	○	○			FFH
0FFEC4H	インサービス・プライオリティ・レジスタ	ISPR		R	○	○			00H
0FFEC5H	割り込みモード・コントロール・レジスタ	IMC		R/W		○			80H
0FFEC9H	割り込み要求制御レジスタ09	IC09		R/W	○	○			43H
0FFECAH	割り込み要求制御レジスタ10	IC10		R/W	○	○			43H
0FFECBH	割り込み要求制御レジスタ11	IC11		R/W	○	○			43H

注1. 入力／出力モードにより異なります。

2. 一部のビットがR，他のビットはR/W可能です。

保守／廃止

表３－５　特殊機能レジスター覧（2/8）

アドレス	特殊機能レジスタ名称	略　号	R/W	操作可能ビット単位				リセット時
				1ビット	8ビット	16ビット	32ビット	
0FFECCH	割り込み要求制御レジスタ12	IC12	R/W	○	○			43H
0FFECDH	割り込み要求制御レジスタ13	IC13	R/W	○	○			43H
0FFECEH	割り込み要求制御レジスタ14	IC14	R/W	○	○			43H
0FFED0H	割り込み要求制御レジスタ16	IC16	R/W	○	○			43H
0FFED1H	割り込み要求制御レジスタ17	IC17	R/W	○	○			43H
0FFED2H	割り込み要求制御レジスタ18	IC18	R/W	○	○			43H
0FFED3H	割り込み要求制御レジスタ19	IC19	R/W	○	○			43H
0FFED4H	割り込み要求制御レジスタ20	IC20	R/W	○	○			43H
0FFED5H	割り込み要求制御レジスタ21	IC21	R/W	○	○			43H
0FFED6H	割り込み要求制御レジスタ22	IC22	R/W	○	○			43H
0FFED7H	割り込み要求制御レジスタ23	IC23	R/W	○	○			43H
0FFED8H	割り込み要求制御レジスタ24	IC24	R/W	○	○			43H
0FFED9H	割り込み要求制御レジスタ25	IC25	R/W	○	○			43H
0FFEDAH	割り込み要求制御レジスタ26	IC26	R/W	○	○			43H
0FFEDBH	割り込み要求制御レジスタ27	IC27	R/W	○	○			43H
0FFEDCH	割り込み要求制御レジスタ28	IC28	R/W	○	○			43H
0FFEDDH	割り込み要求制御レジスタ29	IC29	R/W	○	○			43H
0FFEDEH	割り込み要求制御レジスタ30	IC30	R/W	○	○			43H
0FFEDFH	割り込み要求制御レジスタ31	IC31	R/W	○	○			43H
0FFEE0H	割り込み要求制御レジスタ32	IC32	R/W	○	○			43H
0FFEE4H	割り込み要求制御レジスタ36	IC36	R/W	○	○			43H
0FFEE5H	割り込み要求制御レジスタ37	IC37	R/W	○	○			43H

保守／廃止

表３－５ 特殊機能レジスター覧（3/8）

アドレス	特殊機能レジスタ名称	略 号	R/W	操作可能ビット単位				リセット時
				1ビット	8ビット	16ビット	32ビット	
0FFF00H	ポート0	P0	R/W	○	○			不定
0FFF01H	ポート1	P1	R	○	○			不定
0FFF02H	ポート2	P2	R/W	○	○			不定
0FFF03H	ポート3	P3	R/W	○	○			不定
0FFF04H	ポート4	P4	R/W	○	○			不定
0FFF05H	ポート5	P5	R/W	○	○			不定
0FFF06H	ポート6	P6	R	○	○			不定
0FFF07H	ポート7	P7	R/W	○	○			不定
0FFF08H	ポート8	P8	R/W	○	○			不定
0FFF0CH	ポート・リード・コントロール・レジスタ	PRDC	R/W	○	○			00H
0FFF0EH	リアルタイム出力ポート	RTP	R/W	○	○			不定
0FFF10H	ポート0モード・レジスタ	PM0	R/W	○	○			FFH
0FFF12H	ポート2モード・レジスタ	PM2	R/W	○	○			FFH
0FFF13H	ポート3モード・レジスタ	PM3	R/W	○	○			FFH
0FFF14H	ポート4モード・レジスタ	PM4	R/W	○	○			FFH
0FFF15H	ポート5モード・レジスタ	PM5	R/W	○	○			FFH
0FFF17H	ポート7モード・レジスタ	PM7	R/W	○	○			FFH
0FFF18H	ポート8モード・レジスタ	PM8	R/W	○	○			FFH
0FFF22H	ポート2モード・コントロール・レジスタ	PMC2	R/W	○	○			00H
0FFF23H	ポート3モード・コントロール・レジスタ	PMC3	R/W	○	○			00H
0FFF24H	ポート4モード・コントロール・レジスタ	PMC4	R/W	○	○			00H
0FFF25H	ポート5モード・コントロール・レジスタ	PMC5	R/W	○	○			00H

保守／廃止

表３－５　特殊機能レジスター覧（4/8）

アドレス	特殊機能レジスタ名称	略　号		R/W	操作可能ビット単位				リセット時
					1ビット	8ビット	16ビット	32ビット	
0FFF27H	ポート７モード・コントロール・レジスタ	PMC7		R/W	○	○			00H
0FFF28H	ポート８モード・コントロール・レジスタ	PMC8		R/W	○	○			00H
0FFF2CH	リアルタイム出力ポート・コントロール・レジスタ	RTPC		R/W	○	○			40H
0FFF2DH	リアルタイム出力ポート・ディレイ指定レジスタ	RTPD		R/W	○	○			不定
0FFF2EH	ポート７バッファ（下位）	P7L		R/W	○	○			不定
0FFF2FH	ポート７バッファ（上位）	P7H		R/W	○	○			不定
0FFF30H	タイマ・コントロール・レジスタ０	TMC	TMC0	R/W	○	○	○		00H
0FFF31H	タイマ・コントロール・レジスタ１		TMC1	R/W	○	○			00H
0FFF32H	タイマ出力制御レジスタ０	TOC	TOC0	R/W	○	○	○		00H
0FFF33H	タイマ出力制御レジスタ１		TOC1	R/W	○	○			00H
0FFF34H	外部割り込みモード・レジスタ０	INTM	INTM0	R/W	○	○	○		00H
0FFF35H	外部割り込みモード・レジスタ１		INTM1	R/W	○	○			00H
0FFF40H	タイマ・レジスタ０	TM0		R/W			○		00H
0FFF42H	タイマ・レジスタ１	TM1		R/W			○		00H
0FFF44H	タイマ・レジスタ２	TM2		R/W			○		00H
0FFF46H	タイマ・レジスタ３	TM3		R/W			○		00H
0FFF48H	タイマ・キャプチャ・レジスタ00	CT00		R/W			○		不定
0FFF4AH	タイマ・キャプチャ・レジスタ01	CT01		R/W			○		不定
0FFF4CH	タイマ・コンペア・レジスタ00	CM00		R/W			○		不定
0FFF4EH	タイマ・コンペア・レジスタ01	CM01		R/W			○		不定
0FFF50H	タイマ・キャプチャ・レジスタ10	CT10		R/W			○		不定
0FFF52H	タイマ・コンペア・レジスタ10	CM10		R/W			○		不定

保守／廃止

表３－５　特殊機能レジスター一覧（5/8）

アドレス	特殊機能レジスタ名称	略　号	R/W	操作可能ビット単位				リセット時
				1ビット	8ビット	16ビット	32ビット	
0FFF54H	タイマ・コンペア・レジスタ11	CM11	R/W			○		不定
0FFF58H	タイマ・コンペア・レジスタ20	CM20	R/W			○		不定
0FFF5AH	タイマ・コンペア・レジスタ21	CM21	R/W			○		不定
0FFF5CH	タイマ・コンペア・レジスタ22	CM22	R/W			○		不定
0FFF5EH	タイマ・コンペア・レジスタ23	CM23	R/W			○		不定
0FFF60H	ウォッチドッグ・タイマ・モード・レジスタ	WDM	R/W注1	○	○			00H
0FFF64H	タイマ・コンペア・レジスタ30	CM30	R/W			○		不定
0FFF66H	タイマ・コンペア・レジスタ31	CM31	R/W			○		不定
0FFF6CH	PWMレジスタ	PWM	R/W	○	○			不定
0FFF6DH	PWMコントロール・レジスタ	PWMC	R/W	○	○			00H
0FFF70H	送信ボー・レート・ジェネレータ・レジスタ０	TxBRG0	R/W	○	○			不定
0FFF71H	受信ボー・レート・ジェネレータ・レジスタ０	RxBRG0	R/W	○	○			不定
0FFF72H	プリスケーラ・レジスタ０	PRS0	R/W	○	○			00H
0FFF73H	UARTモード・レジスタ0/クロックト・シリアル・インタフェース・モード・レジスタ０	UARTM0/CSIM0	R/W	(○)	○			00H
0FFF74H	UARTステータス・レジスタ0/SBIコントロール・レジスタ０	UARTS0/SBIC0	注2/注3	(○)	○			20H/00H
0FFF75H	UART送信バッファ0/クロックト・シリアルI/Oシフト・レジスタ０	TxB0/SIO0	W		○			不定
0FFF76H	受信バッファ０	RxB0	R		○			不定

注1．WDMレジスタに対する書き込みは，RSTWDT命令によってのみ可能です（８ビット単位のみ）。

2．一部のビットはR，他のビットはR/W

3．ビット単位にRまたはW

備考　（　）：モードによって異なります。

保守／廃止

表３－５　特殊機能レジスター覧（6/8）

アドレス	特殊機能レジスタ名称	略　号		R/W	操作可能ビット単位				リセット時
					1ビット	8ビット	16ビット	32ビット	
0FFF78H	送信ボー・レート・ジェネレータ・レジスタ1	TxBRG1		R/W	○	○			不定
0FFF79H	受信ボー・レート・ジェネレータ・レジスタ1	RxBRG1		R/W	○	○			不定
0FFF7AH	プリスケーラ・レジスタ1	PRS1		R/W	○	○			00H
0FFF7BH	UARTモード・レジスタ1/クロックト・シリアル・インタフェース・モード・レジスタ1	UARTM1/CSIM1		R/W	(○)	○			00H
0FFF7CH	UARTステータス・レジスタ1	UARTS1		注	(○)	○			20H
0FFF7DH	UART送信バッファ1/クロックト・シリアルI/Oシフト・レジスタ1	TxB1/SIO1		W		○			不定
0FFF7EH	受信バッファ1	RxB1		R		○			不定
0FFF7FH	プロトコル選択レジスタ	ASP		R/W	○	○			00H
0FFF80H	ターミナル・カウンタ0（下位）	TC0	TC0L	R/W			○	○	不定
0FFF82H	ターミナル・カウンタ0（上位）		TC0H	R/W		○	○		不定
0FFF84H	ターミナル・カウンタ・モジュロ・レジスタ0（下位）	TCM0	TCM0L	R/W			○	○	不定
0FFF86H	ターミナル・カウンタ・モジュロ・レジスタ0（上位）		TCM0H	R/W		○	○		不定
0FFF88H	DMAアップ・ダウン・カウンタ0（下位）	UDC0	UDC0L	R/W			○	○	不定
0FFF8AH	DMAアップ・ダウン・カウンタ0（上位）		UDC0H	R/W		○	○		不定
0FFF8CH	DMAコンペア・レジスタ0（下位）	DCM0	DCM0L	R/W			○	○	不定
0FFF8EH	DMAコンペア・レジスタ0（上位）		DCM0H	R/W		○	○		不定
0FFF90H	DMAメモリ・アドレス・レジスタ0（下位）	MAR0	MAR0L	R/W			○	○	不定
0FFF92H	DMAメモリ・アドレス・レジスタ0（上位）		MAR0H	R/W		○	○		不定

注　一部のビットはR，他のビットはR/W

備考　（　）：モードによって異なります。

保守／廃止

表３－５　特殊機能レジスター覧（7/8）

アドレス	特殊機能レジスタ名称	略　号		R/W	操作可能ビット単位				リセット時
					1ビット	8ビット	16ビット	32ビット	
0FFF94H	DMAリード・ライト・ポインタ０（下位）	DPTC0	DPTC0L	R/W			○	○	不定
0FFF96H	DMAリード・ライト・ポインタ０（上位）		DPTC0H	R/W		○	○		不定
0FFF9CH	DMAモード・レジスタ０	DMAM0		R/W	○	○			E0H
0FFF9DH	DMAコントロール・レジスタ０	DMAC0		R/W	○	○			00H
0FFF9EH	DMAステータス・レジスタ	DMAS		R/W	○注	○			00H
0FFFA0H	ターミナル・カウンタ１（下位）	TC1	TC1L	R/W			○	○	不定
0FFFA2H	ターミナル・カウンタ１（上位）		TC1H	R/W		○	○		不定
0FFFA4H	ターミナル・カウンタ・モジュロ・レジスタ１（下位）	TCM1	TCM1L	R/W			○	○	不定
0FFFA6H	ターミナル・カウンタ・モジュロ・レジスタ１（上位）		TCM1H	R/W		○	○		不定
0FFFA8H	DMAアップ・ダウン・カウンタ１（下位）	UDC1	UDC1L	R/W			○	○	不定
0FFFAAH	DMAアップ・ダウン・カウンタ１（上位）		UDC1H	R/W		○	○		不定
0FFFACH	DMAコンペア・レジスタ１（下位）	DCM1	DCM1L	R/W			○	○	不定
0FFFAEH	DMAコンペア・レジスタ１（上位）		DCM1H	R/W		○	○		不定
0FFFB0H	DMAメモリ・アドレス・レジスタ１（下位）	MAR1	MAR1L	R/W			○	○	不定
0FFFB2H	DMAメモリ・アドレス・レジスタ１（上位）		MAR1H	R/W		○	○		不定
0FFFB4H	DMAリード・ライト・ポインタ１（下位）	DPTC1	DPTC1L	R/W			○	○	不定
0FFFB6H	DMAリード・ライト・ポインタ１（上位）		DPTC1H	R/W		○	○		不定
0FFFBCH	DMAモード・レジスタ１	DMAM1		R/W	○	○			E0H
0FFFBDH	DMAコントロール・レジスタ１	DMAC1		R/W	○	○			00H
0FFFE0H	ソフトウエア・タイマ・カウンタ	STC		R			○		不定
0FFFE2H	ソフトウエア・タイマ・カウンタ・コンペア・レジスタ	STMC		R/W			○		FFFFH

注　ビット・クリアは可能です。

保守／廃止

表３－５　特殊機能レジスター覧（8/8）

アドレス	特殊機能レジスタ名称	略　号	R/W	操作可能ビット単位				リセット時
				1ビット	8ビット	16ビット	32ビット	
0FFFE8H	プログラマブル・ウエイト・コントロール・レジスタ0	PWC0	R/W	○	○			EAH
0FFFE9H	プログラマブル・ウエイト・コントロール・レジスタ1	PWC1	R/W	○	○			AAH
0FFFEAH	メモリ・ブロック・コントロール・レジスタ	MBC	R/W	○	○			FCH
0FFFECH	リフレッシュ・モード・レジスタ	RFM	R/W	○	○			77H
0FFFEEH	スタンバイ・コントロール・レジスタ	STBC	R/W注1	○	○			不定注2
0FFFEFH	プロセッサ・コントロール・レジスタ	PRC	R/W	○	○			EEH

注1．スタンバイ・コントロール・レジスタのSBFビット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（書き込みは“1”だけ可）。

2．パワー・オン・リセット時：00H，その他：変化せず

保守／廃止

3.5.4　ベクタ・テーブル領域

メモリ空間の00000H-003FFHの１Kバイトの領域は，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256ベクタ分（１ベクタ当り４バイトを使用）保有します。

V55PI専用の内蔵周辺およびソフトウエア割り込みのベクタとして，初期状態ではベクタ0-47が予約されています。また，ベクタ8-ベクタ47は，割り込みモード・コントロール・レジスタ（IMC）のV0，V1ビットでNMIを除くハードウエア割り込みのベクタ・アドレスを変更可能です。

ベクタ・テーブルとして使用しない領域は，ほかの用途で使用できます。

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命令
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）：入出力命令
ベクタ 7　（0001CH）：FPO命令／例外トラップ

V1＝V0＝0のとき

ベクタ 8　（00020H）：INTWDT
ベクタ 9　（00024H）：INTP0
ベクタ10　（00028H）：INTP1
ベクタ11　（0002CH）：INTP2
ベクタ12　（00030H）：INTP3
ベクタ13　（00034H）：INTP4
ベクタ14　（00038H）：INTP5
ベクタ15　（0003CH）：システム予約
ベクタ16　（00040H）：INTCM00
ベクタ17　（00044H）：INTCM01
ベクタ18　（00048H）：INTCM10
ベクタ19　（0004CH）：INTCM11
ベクタ20　（00050H）：INTCM21
ベクタ21　（00054H）：INTCM31
ベクタ22　（00058H）：INTD0　DMA＃0_MAIN
ベクタ23　（0005CH）：INTD0S　DMA＃0_SUB
ベクタ24　（00060H）：INTD1　DMA＃1_MAIN
ベクタ25　（00064H）：INTD1S　DMA＃1_SUB
ベクタ26　（00068H）：INTSER0
ベクタ27　（0006CH）：INTSER1

保守／廃止

ベクタ28　(00070H)　：INTSR0/INTCSI0
ベクタ29　(00074H)　：INTSR1/INTCSI1
ベクタ30　(00078H)　：INTST0
ベクタ31　(0007CH)　：INTST1
ベクタ32　(00080H)　：INTSIT
ベクタ33　(00084H)　：システム予約
ベクタ34　(00088H)　：システム予約
ベクタ35　(0008CH)　：システム予約
ベクタ36　(00090H)　：INTPAI
ベクタ37　(00094H)　：INTAD
ベクタ38　(00098H)　：システム予約
ベクタ39　(0009CH)　：システム予約
ベクタ40　(000A0H)　：システム予約
ベクタ41　(000A4H)　：システム予約
ベクタ42　(000A8H)　：システム予約
ベクタ43　(000ACH)　：システム予約
ベクタ44　(000B0H)　：システム予約
ベクタ45　(000B4H)　：システム予約
ベクタ46　(000B8H)　：システム予約
ベクタ47　(000BCH)　：システム予約

V1＝0，V0＝1のとき
ベクタ72　(00120H)　：INTWDT
ベクタ73　(00124H)　：INTP0
　｜　　｜　　｜
ベクタ110　(001B8H)　：システム予約
ベクタ111　(001BCH)　：システム予約

V1＝1，V0＝0のとき
ベクタ136　(00220H)　：INTWDT
ベクタ137　(00224H)　：INTP0
　｜　　｜　　｜
ベクタ174　(002B8H)　：システム予約
ベクタ175　(002BCH)　：システム予約

V1＝V0＝1のとき
ベクタ200　(00320H)　：INTWDT

保守／廃止

ベクタ201　（00324H）：INTP0
　|　　|　　|
ベクタ238　（003B8H）：システム予約
ベクタ239　（003BCH）：システム予約

備考　システム予約の領域は，将来の品種展開での機能拡張のために予約している領域です。これらの領域に対しても，ほかのベクタ・テーブルに対するアクセスと同様にアクセスすることができます。

3.6　レジスタ・ファイル空間

図3-5に，レジスタ・ファイル空間を示します。

レジスタ・ファイル空間のサイズは512バイトで，最大16バンクのレジスタ・セットを設定することができます。

レジスタ・ファイル空間はメモリ空間とは分離されており，レジスタ・ファイル空間へのアクセスは，V25，V35と同様にレジスタ操作命令を使用して可能なほか，メモリ操作命令に専用のプリフィクス命令（IRAM：）を付加してデータ・メモリとしてアクセスすることも可能です。

IRAM：プリフィクス命令をメモリ操作命令に付加すると，CPUはメモリ・アドレスのオフセット値の下位の9ビットをレジスタ・ファイルのアドレスとしてデータ・アクセスを行います。このとき，セグメント・レジスタとの物理アドレス加算は行わず，また外部バス・サイクルも起動しません（外部バス・サイクルに対しては非同期に独立してアクセスされます）。

例

```
Label1：　MOV　IRAM：〔0024H〕，AW……①
　　　　　MOV　〔0056H〕，BW　　……②
```

①　IRAM：プリフィクスを用いてレジスタ・ファイル空間へのデータを転送する場合を示します。AWレジスタの値がレジスタ・ファイルの24H番地に格納されます。

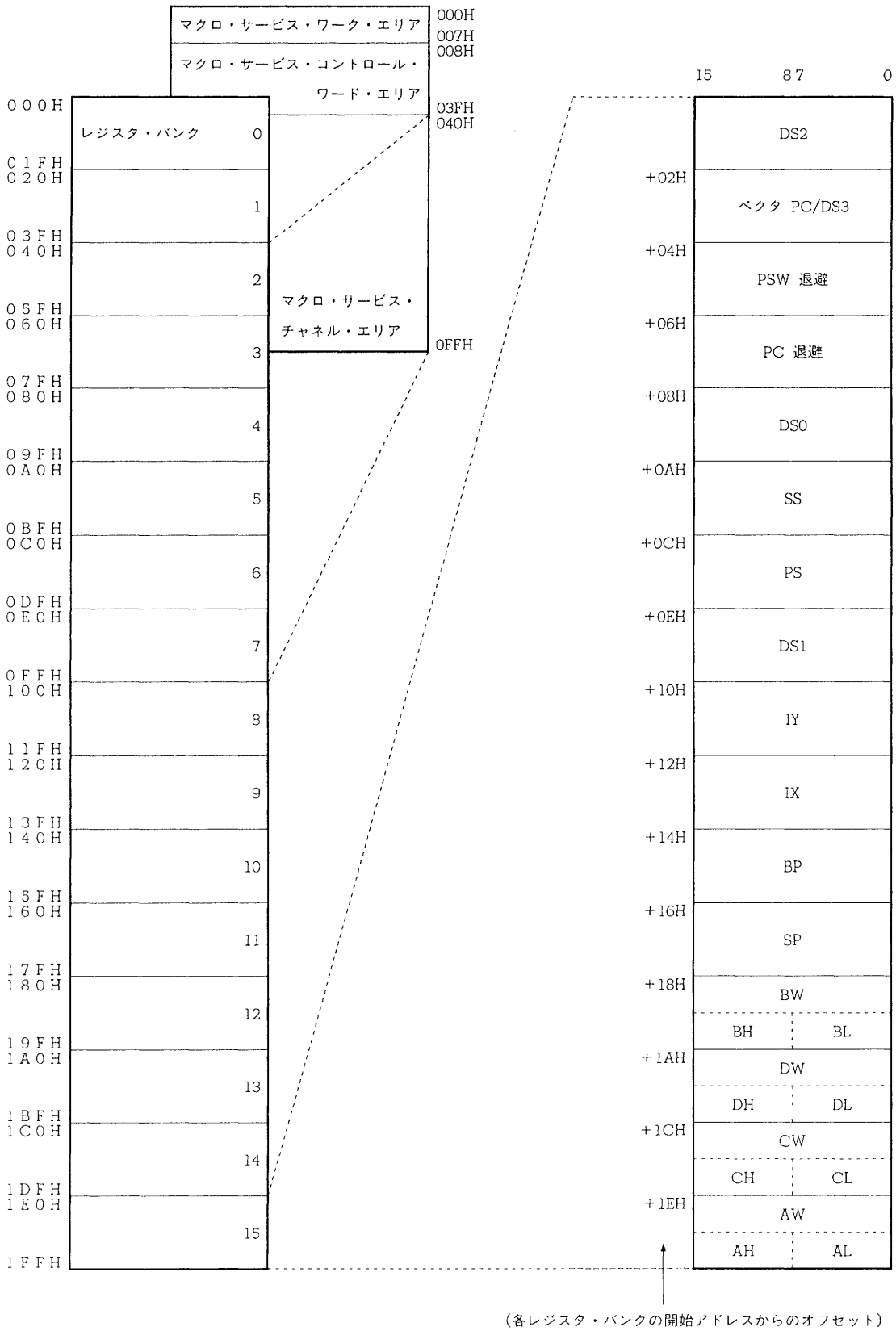
②　メモリ空間へのデータを転送する場合を示します。

ソース・ブロックとデスティネーション・ブロックの指定を伴うプリミティブ・ブロック転送命令，BCD演算命令にIRAMプリフィクス命令を付加した場合，デスティネーション・ブロックに対して有効になります。

また，レジスタ・ファイル空間には，マクロ・サービスのコントロール・ワード・エリア（008H-03FH），マクロ・サービス・ワーク・エリア（000H-007H），およびマクロ・サービス・チャネルでの使用領域（008H-0FFH）が重複して割り付けられています。このワーク・エリアは，ワークを必要とする特定のマクロ・サービス（RTOPTRN）を使用しなければ任意のデータ空間としても使用可能です。

保守／廃止

図3-5　レジスタ・ファイル空間



保守／廃止

3.7　I/O空間

V55PIは，64 Kバイトの I/O空間を持っています。

図３－６にI/O空間のマップを示します。

I/O空間は，アドレス・バス／データ・バス，コントロール信号（$\overline{\text{IORD}}$，$\overline{\text{IOWR}}$など）を用いてアクセスします。

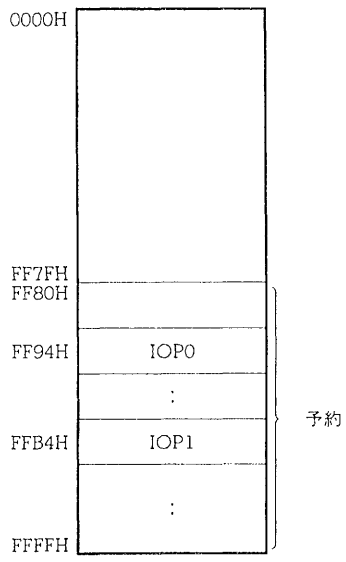
使用しないアドレス・バスの上位８ビットからは０が出力されます。

また，ソフトウエアおよびREADY端子によって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。

I/O空間のFF80H-FFFFHは予約領域になっており，V55PIが内蔵している周辺DMAの入出力リード・ライト・ポインタ（IOP）２つが割り付けられています。アドレスはIOP0がFF94H，IOP1がFFB4Hです。

CPUがIOPのアドレスをオペランドとする入出力命令を実行すると，DMAコントローラはIOPの内容をアドレス値として，DMAコントローラの転送バッファ上のデータの読み出し／書き込みを行い，DMAコントロール・レジスタの内容に従って自動的にIOPの値をインクリメント（または，デクリメント）します。したがってDMAコントローラが書き込んだデータを入出力命令で参照したり，逆に入出力命令で書き込んだデータをDMAコントローラで転送することができます。

図３－６　I/Oマップ（64 Kバイト）



備考　IOPnは，DMAリード・ライト・ポインタn（DPTCn）に対応しています。

保守／廃止

3.8　論理アドレスと物理アドレス

メモリ空間のアドレスには論理アドレスと物理アドレスの２種類があります。

物理アドレスはハードウエアに直接対応したアドレスを意味します。V55PIは16 Mバイトのメモリ空間を持っています。このうち下位１Mバイト（000000H-0FFFFFH）を基本メモリ空間として，また基本メモリ空間を含む16 Mバイト（000000-FFFFFFH）を拡張メモリ空間としてアクセスします。したがって，物理アドレスの値の範囲は000000H-FFFFFFHです。基本メモリ空間，拡張メモリ空間の詳細については，**3.5　メモリ空間**を参照してください。

物理アドレスの生成は，命令をフェッチしたりデータを転送するバス・コントロール・ユニット（BCU）が起動するたびに行われます。

論理アドレスは，セグメント方式のアドレス指定の際に用いるアドレスを意味します。

3.8.1　セグメント方式

セグメントとは，プログラム作成の際に直接依存しない小単位（最大64 Kバイト）のアドレス空間のことです。各セグメントは連続したメモリから構成され，それぞれ独立して指定することができます。

マシン語のプログラム作成では，物理アドレスを直接管理することはできません。V55PIでは，メモリのアドレスをセグメント方式で指定します。

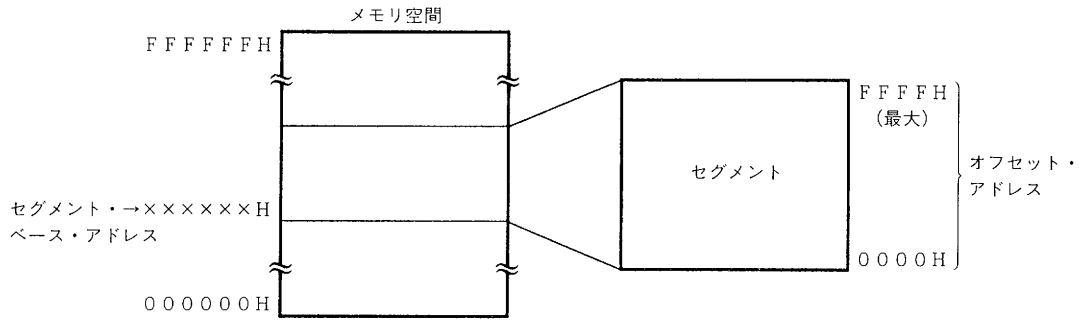
セグメント方式のアドレス指定には，次の２種類のアドレスを使用します。

セグメント・ベース・アドレス：セグメントの先頭アドレス（16Mバイト・メモリ空間内でのアドレス）

オフセット・アドレス　　：各セグメントごとに付けられたアドレス

セグメント方式ではセグメント・ベース・アドレスを基準点として固定し，各セグメント内での処理はオフセット・アドレスのみをアドレスとして扱います。

図３－７　セグメント方式の概念図



セグメント・ベース・アドレスはセグメント・レジスタにより指定されます。

保守／廃止

物理アドレスはセグメント・ベース・アドレスにオフセット・アドレスを加えた値となります。

図３－８にセグメント・レジスタ，オフセット・アドレスと物理アドレスの関係を示します。

図３－８　セグメント・レジスタ，オフセット・アドレスと物理アドレスの関係

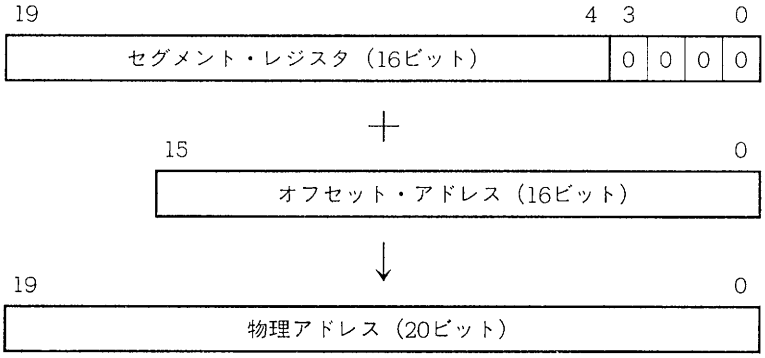
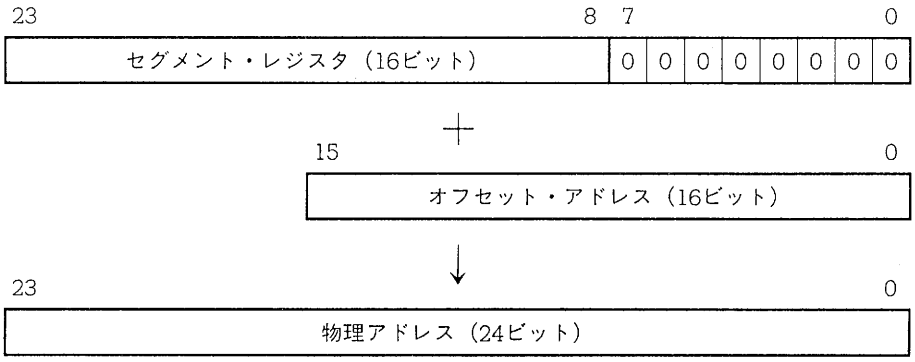


図３－８に示すように，セグメント・レジスタ（PS，SS，DS0，DS1）の内容を16倍（４ビット左へシフト）したものにオフセット値を加えたものが物理アドレス（20ビット）になります。このとき，セグメント・レジスタの内容およびオフセット値は符号なしデータとして扱います。

なお，拡張セグメント・レジスタ（DS2，DS3）の場合は，拡張セグメント・レジスタの内容を32倍（８ビット左へシフト）したものにオフセット値を加えたものが物理アドレス（24ビット）になります。

図３－９　拡張セグメント・レジスタ，オフセット・アドレスと物理アドレスの関係



論理アドレスにより配置アドレスが指定される複数のセグメントの集合として作るプログラムで，各セグメントは個々にコンパイル，アセンブルされて複数のオブジェクト・モジュールになります。個々のオブジェクト・モジュールはセグメント名，サイズ，内容区分，制御情報などを持っており，リンク処理実行時のパラメータになります。

複数のオブジェクト・モジュールはリンクされ，物理アドレスに対応したセグメント・ベース・アドレスが指定されて実際のメモリにロードできる状態になります。

保守／廃止

3.8.2　セグメントの構成

V55PIでは，６種類のセグメント（プログラム，スタック，データ０，データ１，データ２，データ３）を区別して定義することができ，各セグメントは，それぞれ次の４つのセグメント・レジスタと２つの拡張セグメント・レジスタで，開始アドレスが指定されます（**3.2.4　セグメント・レジスタ（PS, SS, DS0, DS1）**，**3.2.5　拡張セグメント・レジスタ（DS2, DS3）**参照）。

BCUは，メモリ・バス・サイクルの種類により，物理アドレス生成にそれぞれ異なるセグメント・レジスタを使用します。

セグメント・レジスタ

- プログラム・セグメント・レジスタ（PS）
- スタック・セグメント・レジスタ（SS）
- データ・セグメント０レジスタ（DS0）
- データ・セグメント１レジスタ（DS1）

拡張セグメント・レジスタ

- 拡張データ・セグメント２レジスタ（DS2）
- 拡張データ・セグメント３レジスタ（DS3）

各セグメント内のオフセット・アドレスは，特定のレジスタまたは実効アドレスによって指定されます。セグメント・レジスタとオフセット・アドレス指定の対応を表３-６に示します。

表３-６　セグメント・レジスタとオフセット・アドレス指定の対応

セグメント・レジスタ ／ オフセット	ディフォールト	オーバライド
PFP	PS	不可
SP	SS	不可
実効アドレス（BPベース）		PS, DS0, DS1, DS2, DS3
実効アドレス（非BPベース）	DS0	PS, SS, DS1, DS2
命令群A[注]でのIX		
命令群B[注]でのIY	DS1	DS3

注　命令群A：プリミティブ・ブロック転送命令，プリミティブ出力命令，BCD演算命令，EXT命令

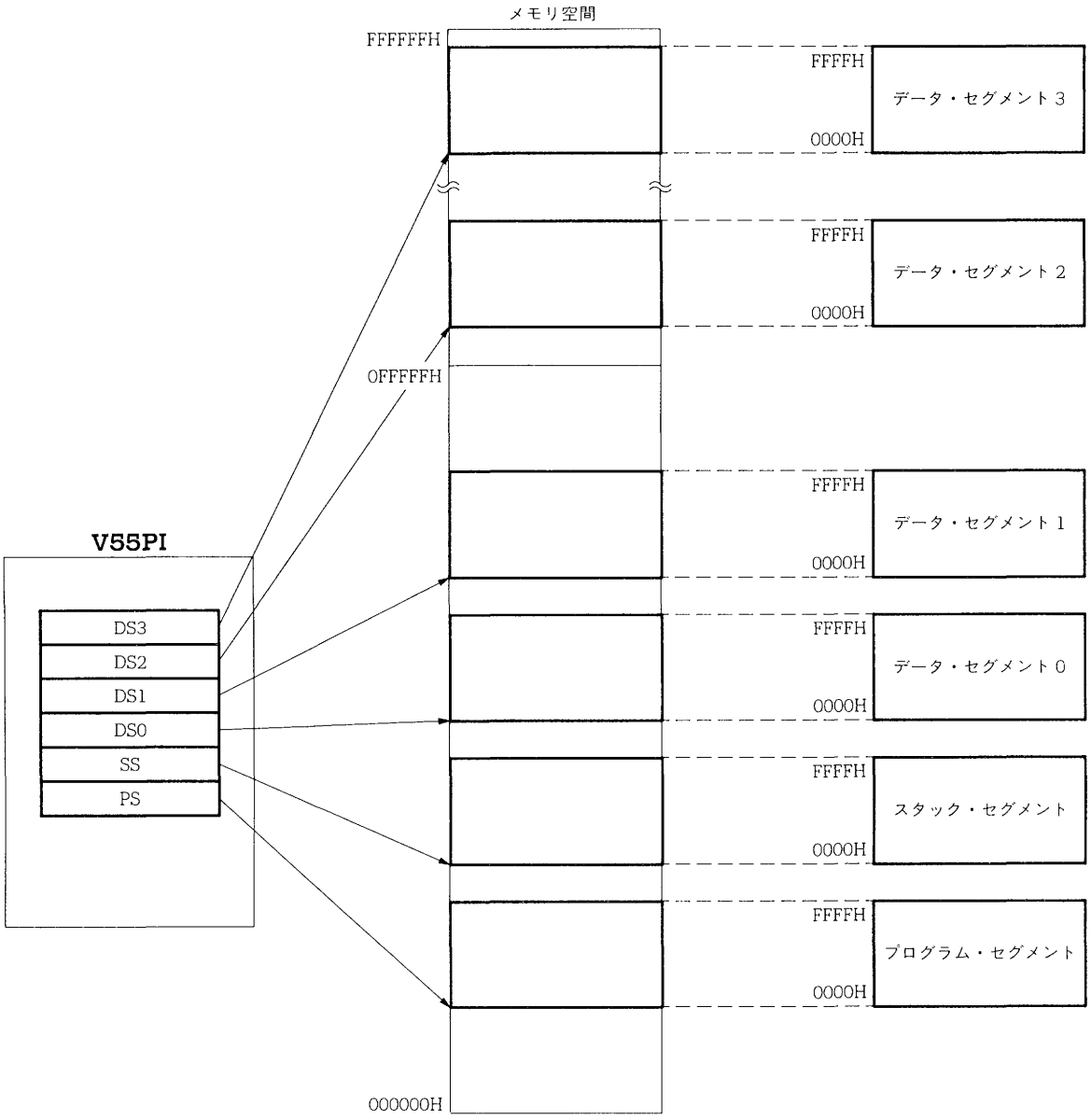
命令群B：プリミティブ・ブロック転送命令，プリミティブ入力命令，BCD演算命令，INS命令

保守／廃止

ディフォールト・オフセットがプリフェッチ・ポインタ（PFP），スタック・ポインタ（SP）の場合は，組み合わせることのできるセグメント・レジスタはそれぞれPS，SSに固定されており，ほかのセグメント・レジスタは使用できません。

その他のディフォールト・オフセットでは，ディフォールトのセグメント・レジスタ以外に指定のセグメント・レジスタまたは拡張セグメント・レジスタをセグメント・オーバライド・プリフィクスによって指定することができます。

図3－10　セグメント・レジスタとセグメント，およびメモリ空間の関係



保守／廃止

各セグメントは次のような意味を持っています。

(1) プログラム・セグメント

このセグメントの先頭アドレスはプログラム・セグメント・レジスタ(PS)で決定され，プリフェッチ・ポインタ（PFP）によって先頭アドレスからのオフセットが指定されます。

このセグメントには，命令コード，テーブル・データなどが配置されます。

セグメント・オーバライド・プリフィクス（PS：）を用いれば，一般変数領域や，命令群A実行時のソース・データ領域としてプログラム・セグメントを利用することができます。

(2) スタック・セグメント

このセグメントの先頭アドレスはスタック・セグメント・レジスタ（SS）で決定され，スタック・ポインタ（SP），およびベース・アドレスとしてベース・ポインタ（BP）を用いた場合の実効アドレスによって先頭アドレスからのオフセットが指定されます。

割り込みやサブルーチン処理の際に，リターン・アドレス(PS，PCの内容)やプログラム・ステータス・ワード（PSW），汎用レジスタなどの内容の退避領域，パラメータの受け渡し領域，ローカル変数領域として利用されます。

セグメント・オーバライド・プリフィクス（SS：）を用いれば，一般変数領域や，命令群A実行時のソース・データ領域としてスタック・セグメントを利用することができます。

(3) データ・セグメント０

このセグメントの先頭アドレスはデータ・セグメント０レジスタ（DS0）によって決定され，ベース・アドレスとしてBPを用いない場合の実効アドレスによって先頭アドレスからのオフセットが指定されます。

このセグメントは，一般変数の格納エリアに利用されます。

命令群A実行の際には，ソース・データ領域として利用されます。ただし，この場合，インデクス・レジスタ（IX）の内容がオフセットとなります。

ベース・アドレスとしてBPを用いた場合の実効アドレスでは，ディフォールトとしてスタック・セグメントが使用されますが，セグメント・オーバライド・プリフィクス(DS0：)を用いれば，データ・セグメント０を利用することができます。

(4) データ・セグメント１

このセグメントの先頭アドレスはデータ・セグメント１レジスタ（DS1）によって決定されます。

命令群B実行時での，デスティネーション・データ領域として利用されます。この場合，インデクス・レジスタ（IY）の内容がオフセットとなります。

セグメント・オーバライド・プリフィクス（DS1：）を用いれば，一般変数領域や，命令群A実行時のソース・データ領域としてデータ・セグメント１を利用することができます。

保守／廃止

(5) データ・セグメント２

このセグメントの先頭アドレスは拡張データ・セグメント２レジスタ（DS2）によって決定され，ベース・アドレスとしてBPを用いない場合の実効アドレスによって先頭アドレスからのオフセットが指定されます。

このセグメントは，拡張メモリ空間の一般変数の格納エリアに利用されます。

命令群A実行の際には，ソース・データ領域として利用されます。ただし，この場合，インデクス・レジスタ（IX）の内容がオフセットとなります。

ベース・アドレスとしてBPを用いた場合の実効アドレスでは，ディフォールトとしてスタック・セグメントが使用されますが，拡張セグメント・オーバライド・プリフィクス（DS2：）を用いれば，データ・セグメント２を利用することができます。

(6) データ・セグメント３

このセグメントの先頭アドレスは拡張データ・セグメント３レジスタ（DS3）によって決定されます。

命令群B実行時での，デスティネーション・データ領域として使用されます。この場合，インデクス・レジスタ（IY）の内容がオフセットとなります。

拡張セグメント・オーバライド・プリフィクス（DS3：）を用いれば，一般変数領域や，命令群B実行時のソース・データ領域としてデータ・セグメント３を利用することができます。

保守／廃止

3.8.3　ダイナミック・リロケーション

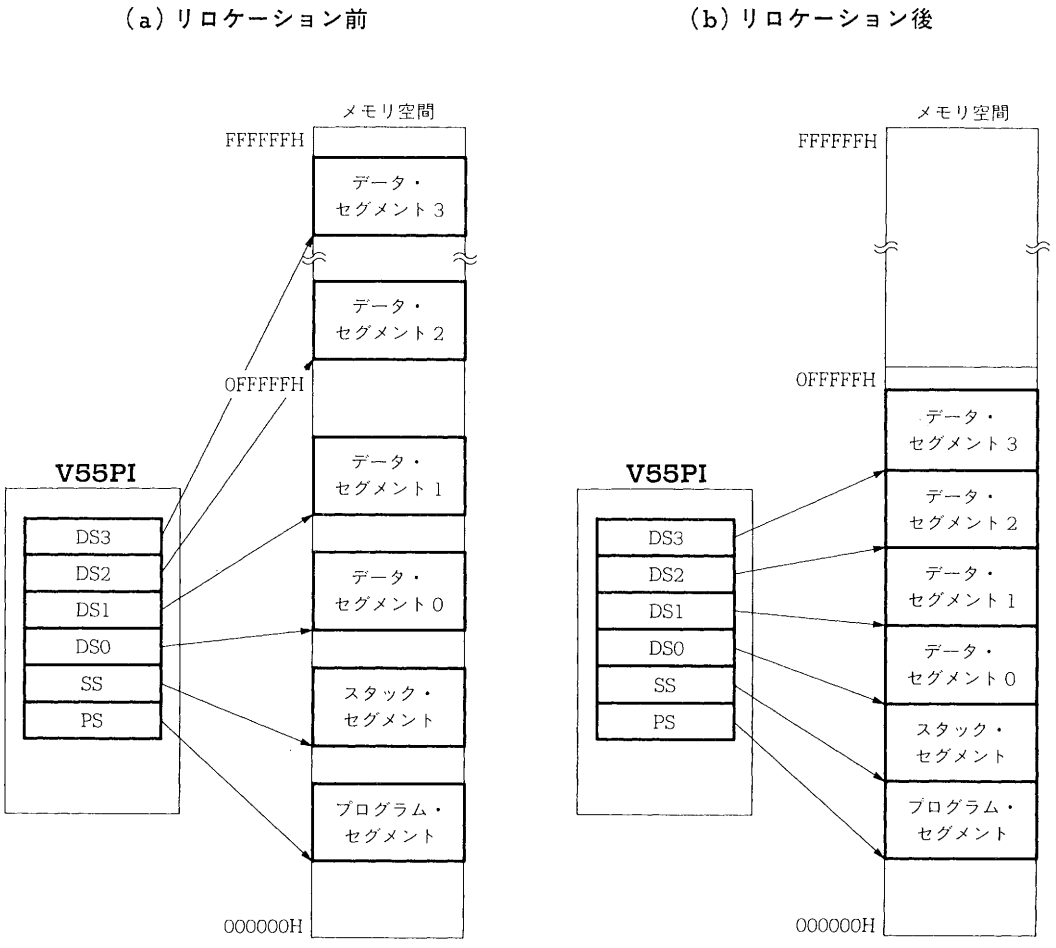
複数のファイルにばらばらに格納されているプログラムを，実行するごとに空いているメモリ空間に再配置することをダイナミック・リロケーションといいます。

図３－11にダイナミック・リロケーションの概念図を示します。

V55PIでは，プログラムのメモリ・アドレス指定は，すべて各セグメントのベース・アドレス（各セグメント・レジスタ，拡張セグメント・レジスタにより指定）に対するオフセット値だけで決定できます。したがって，プログラムを配置する際，セグメント・レジスタ，拡張セグメント・レジスタの内容を，そのプログラムを配置しようとしているメモリの物理アドレスに合わせるだけでメモリの任意空間に配置することができます（ただし，プログラムの中で各セグメントのベース・アドレスを変更するような処理を行わない場合にかぎります）。このため，プログラムのメモリへの配置の自由度が高く（16バイト単位でのアドレス指定が可能），メモリのより効果的な使用が可能になり，さらに複数のジョブやタスクを実行するシステムを容易に実現できます。

このことを利用すれば，フロッピィ・ディスクやハード・ディスクなどの外部記憶媒体上のファイルにあるプログラムを実行させるのに，OSがメモリの配置領域や種類およびセグメント・レジスタ，拡張セグメント・レジスタの管理を行い，空いている任意のメモリ領域にプログラムをロードして実行させることができます。

図３－11　ダイナミック・リロケーション



保守／廃止

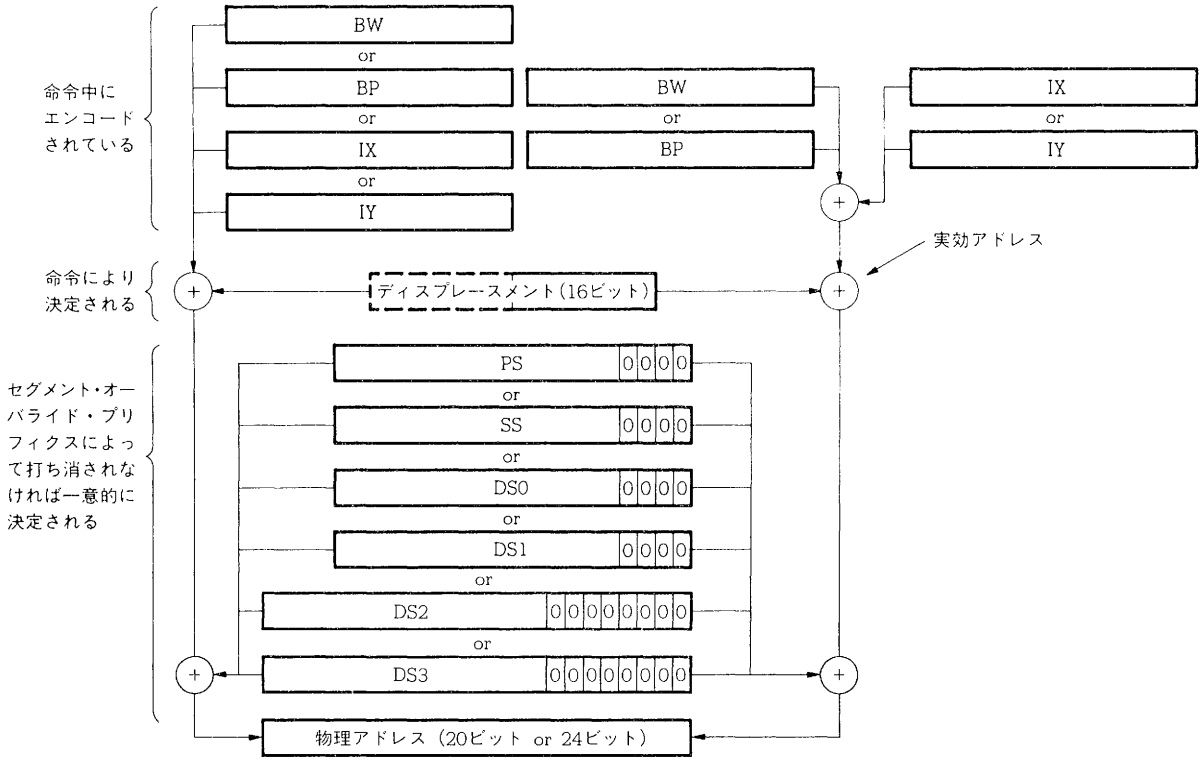
3.9　実効アドレス

実効アドレス（EA）は無符号の16ビット数で，命令処理の対象となるメモリ・アドレスを該当セグメントのベース・アドレスに対するオフセット値で表したものです。命令実行時に，エクセキューション・ユニット（EXU）によって，命令オペランドの指定に従って計算されます。

EXUはEAを数種の異なる方法（アドレシング・モード）で計算しますが，方法の選択は命令の２バイト目のオペランドで指定します。命令の２バイト目にエンコードされた情報は，オペランドで示されるメモリの実効アドレスをEXUがどのように計算するかを表します。このオペランド・コードは，コンパイラまたはアセンブラによりプログラムのステートメントまたは命令記述から自動的に生成されます。アセンブリ言語では，すべてのアドレシング・モードを使用できます（**3.10　アドレシング・モード**参照）。

EAの計算方法を次に示します。図３－12は，EXUがディスプレースメント，ベース・レジスタの内容，およびインデクス・レジスタの内容を加算することによってEAを計算することを示しています。どの命令でも，この３つの要素を任意に組み合わせることができます。ディスプレースメントは，オペランドで示される８ビットまたは16ビットのイミーディエト数です。

図３－12　メモリ・アドレスの計算



保守／廃止

3.10　アドレシング・モード

3.10.1　命令アドレス

命令アドレスはPSおよびPCの内容によって決定され，通常，命令を実行するごとにフェッチする命令のバイト数に応じて自動的にインクリメントされます。分岐を伴う命令を実行する際には，次に示すようなアドレシングにより，分岐先アドレス情報がPCまたはPSとPCにロードされ分岐します。

プリミティブ命令以外のメモリ・アドレシングにおいてBPを使用する場合，ディフォールトのセグメント・レジスタはDS0になります。

プリミティブ命令のメモリ・アドレシングにおいて，デスティネーション・ブロックのディフォールトのセグメント・レジスタはDS1になります。また，メモリ・アドレシングにおいてソース・ブロックのディフォールトのセグメント・レジスタはDS0になります。

(1) ダイレクト・アドレシング

命令バイト中の２バイトまたは４バイト・イミーディエト・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   memptr16
BR   memptr32
```

(2) レラティブ・アドレシング

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・データとなりPCに加算されます。

加算されるときの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
```

保守／廃止

(3) レジスタ・アドレシング

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。

```
CALL regptr16
BR   regptr16
```

(4) レジスタ・インダイレクト・アドレシング

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）により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，また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。

(5) インデクスト・アドレシング

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）に加算されます。その結果により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。

保守／廃止

記述例

```
CALL memptr16          CALL var [IX] [2]
CALL memptr32          CALL var [IY]
BR   memptr16          BR   var [BW]
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，またダブル・ワード属性を持つ場合はmemptr32の命令コードが，それぞれアセンブラによって生成されます。

(6) ベースト・アドレシング

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，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）に加算されます。その結果により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，またダブル・ワード属性を持つ場合はmemptr32の命令コードが，それぞれアセンブラによって生成されます。

(7) ベースト・インデクスト・アドレシング

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つが加算されます。その結果により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。

保守／廃止

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がワード属性を持つ場合はmemptr16の命令コードが，またダブル・ワード属性を持つ場合はmemptr32の命令コードが，それぞれアセンブラによって生成されます。

3.10.2　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。

(1) レジスタ・アドレシング

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令する１ビット（W）と組になって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。

xsregの場合は２種の拡張セグメント・レジスタ（DS2，DS3）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もあります。

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，reg′	AW，BW，CW，DW，SP，BP，IX，IY， AL，AH，BL，BH，CL，CH，DL，DH
reg16，reg16′	AW，BW，CW，DW，SP，BP，IX，IY
reg8，reg8′	AL，AH，BL，BH，CL，CH，DL，DH
sreg	PS，SS，DS0，DS1
xsreg	DS2，DS3
acc	AW，AL

記述例

MOV reg，regの場合

MOV BP，SP

MOV AL，CL

保守／廃止

(2) イミーディエト・アドレシング

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別して決定し，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
```

(3) ダイレクト・アドレシング

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	8/16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_BAR，2000H

MOV acc，dmemの場合
MOV AL，BYTE_BAR
```

保守／廃止

(4) レジスタ・インダイレクト・アドレシング

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

mem	[IX]，[IY]，[BW]

記述例

SUB mem，regの場合

SUB　[IX]，AW

(5) オートインクリメント／オートデクリメント・アドレシング

レジスタ・インダイレクト・アドレシングに属するものですが，ディフォールト・レジスタの内容で操作対象をアドレスしたあと，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。

このアドレシングは，すべてディフォールト・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	ディフォールト・レジスタ
dst-block	IY
src-block	IX

このアドレシングとバイト／ワード・オペランドの繰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。

保守／廃止

(6) インデクスト・アドレシング

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，インデクスト・レジスタとして働く16ビット・レジスタ（IXまたはIY）に加算されます。その結果により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。

(7) ベースト・アドレシング

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，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，その結果により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す。ベース・レジスタは各構造の開始アドレスを指し，ディスプレースメントは各構造内の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	〃

保守／廃止

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ます。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。

(8) ベースト・インデクスト・アドレシング

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，その結果により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。つまり，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・データの何番目かをインデクス・レジスタが指すというような使い方ができます。

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	記述方式
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]
```

保守／廃止

(9) ビット・アドレシング

命令バイト中の3/4ビット・イミーディエト・データ，またはCLレジスタの下位3/4ビットにより操作対象の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の場合	TEST1 AL，CL
NOT1	reg8，imm3の場合	NOT1 CL，5
CLR1	mem16，CLの場合	CLR1 WORD_VAR [IX]，CL
SET1	mem16，imm4の場合	SET1 WORD_VAR [BP]，9

(10) 特殊機能レジスタ・アドレシング

命令バイト中の１バイト・イミーディエト・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。

このアドレシング・モードはBTCLR命令およびBTCLRL命令に適用されます。

記述形式	記述方法
sfr	８ビット特殊機能レジスタ（アドレス：FFF00H-FFFEFH）を指定する８ビット変数
sfr1	８ビット特殊機能レジスタ（アドレス：FFE00H-FFEFFH）を指定する８ビット変数

記述例

BTCLR	DMAS，3，45

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，FFE00H-FFFEFHの領域にあります。

保守／廃止

〔メ　モ〕

保守／廃止

第４章　バス制御機能

V55PIは，外部バスを介して外部メモリ，外部I/Oとのインタフェースを行います。

4.1　外部バス制御端子

外部バス制御のための端子については**2.1.2　ポート以外の機能**を参照してください。

なお，ポート端子との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（PMCn）によって相当する機能を選択する必要があります。

4.2　外部データ・バス幅選択機能

外部バスのデータ入出力を８ビットで行うか16ビットで行うかを指定することができます。

データ・バス幅選択機能の指定は，D8/$\overline{\text{D16}}$端子の操作によって行います。ハイ・レベルに固定すれば外部データ・バス幅は８ビットとなり，ロウ・レベルに固定すれば16ビットとなります。リセット解除時にD8/$\overline{\text{D16}}$端子のレベルがモニタリングされ，外部データ・バス幅が決定されます。

D8/$\overline{\text{D16}}$端子	外部データ・バス幅
ロウ・レベル	16ビット
ハイ・レベル	８ビット

注意　８ビット・バス幅のとき，AD8-AD15端子は１つのバス・サイクルの間，アドレス出力を保持します（ホールド・モード時を除く）。また，$\overline{\text{WRL}}$端子はメモリ・ライト・ストローブ信号出力となり，$\overline{\text{WRH}}$端子は，常にハイ・レベルとなります。

4.3　基本バス・サイクル

V55PIのバス・サイクルは，基本的に（ウエイト・サイクルを挿入しない場合）３クロックで動作します。アドレス出力信号の一部とデータ入出力信号はマルチプレクスされており，時分割でアドレス出力とデータ入出力を行います。

アドレス・サイクルであるT1サイクル（最初の１クロック）でアドレス出力を，データ・サイクルのT2，T3サイクル（残りの２クロック）でデータ入出力を行います。

V55PIが外部バスに対して起動するバス・サイクルは次のとおりです。

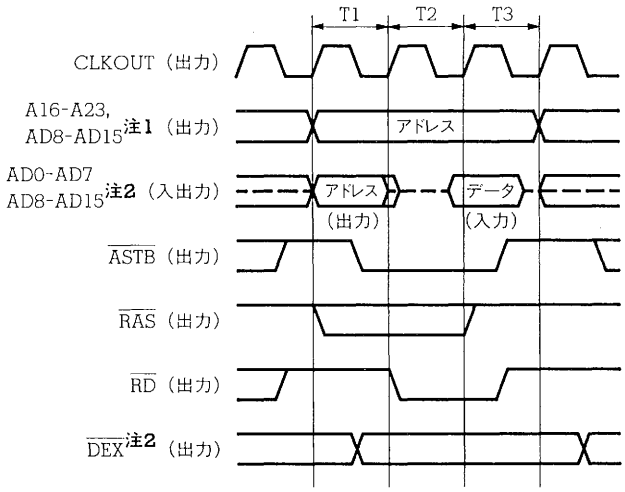
保守／廃止

- プログラム・コード・フェッチ・サイクル
- メモリ・リード・サイクル
- メモリ・ライト・サイクル
- I/Oリード・サイクル
- I/Oライト・サイクル
- リフレッシュ・サイクル
- DMA転送サイクル（外部I/O-外部メモリ）
- SFRリード・サイクル
- SFRライト・サイクル

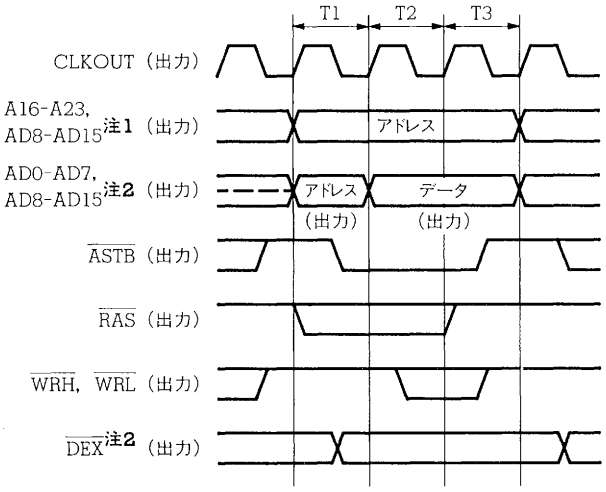
次の(1)～(6)に各サイクルのタイミング・チャートの概略を示します。

また，特殊機能レジスタ（SFR）に対してアクセスした場合の外部バスの状態を(7)，(8)に示します。

(1) メモリ・リード・サイクル，プログラム・コード・フェッチ・サイクル



(2) メモリ・ライト・サイクル



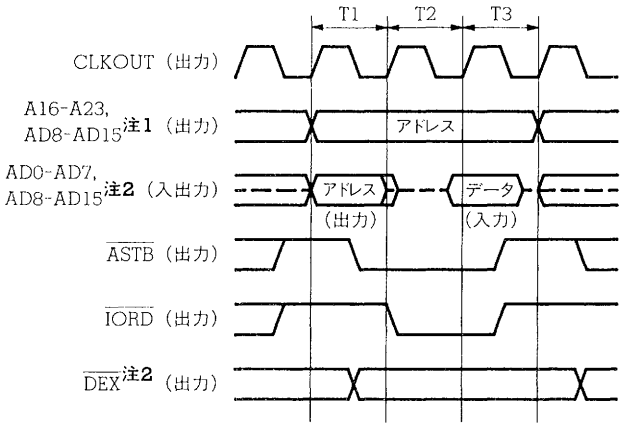
注1. 外部バス幅８ビットのとき

2. 外部バス幅16ビットのとき

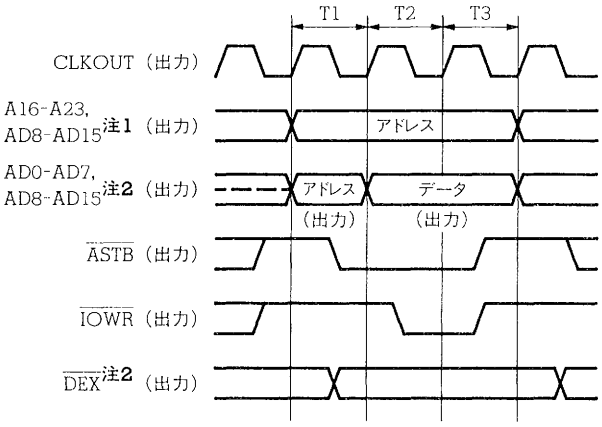
備考　破線はハイ・インピーダンスを示します。

保守／廃止

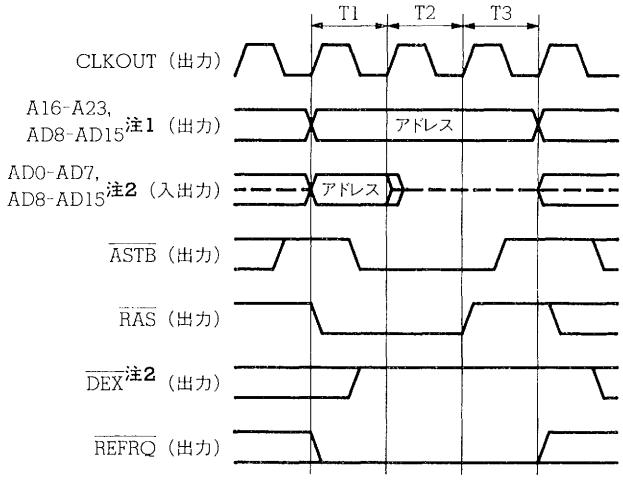
(3) I/Oリード・サイクル



(4) I/Oライト・サイクル



(5) リフレッシュ・サイクル



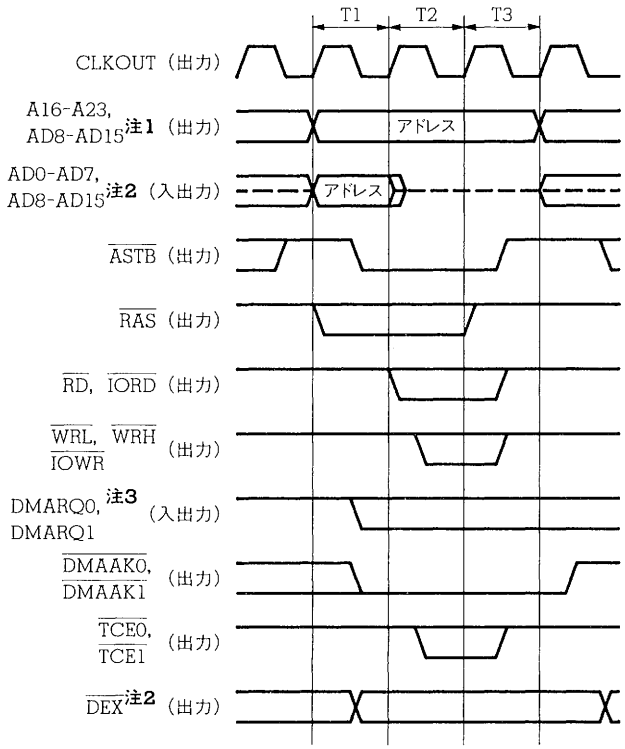
注1．外部バス幅８ビットのとき

2．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

(6) DMA転送サイクル（外部I/O-外部メモリ）



注1. 外部バス幅８ビットのとき

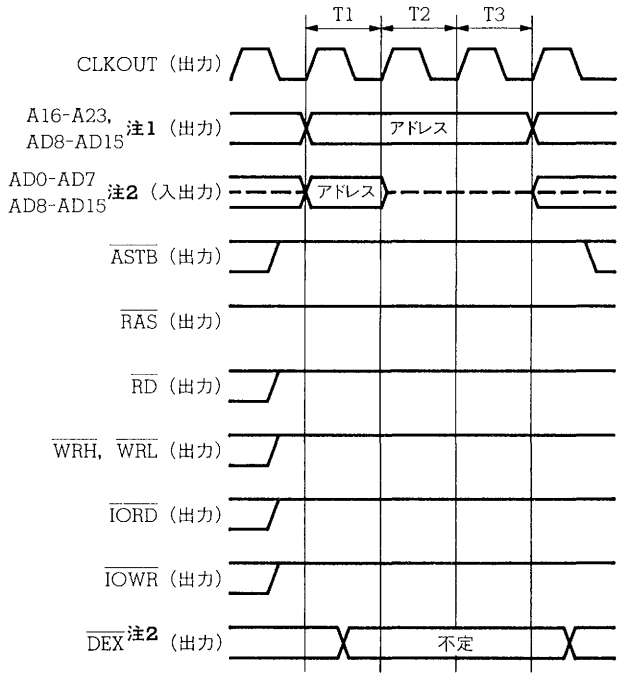
2. 外部バス幅16ビットのとき

3. ディマンド・リリース・モードのとき

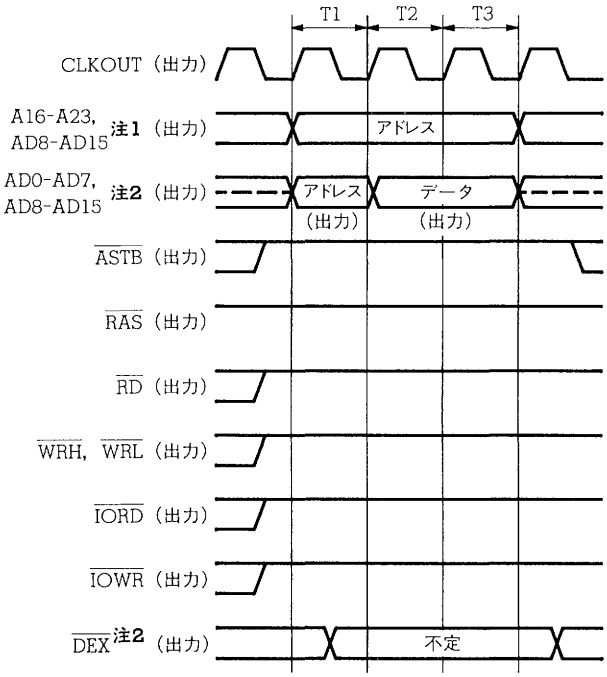
備考　破線はハイ・インピーダンスを示します。

保守／廃止

(7) SFRリード・サイクル



(8) SFRライト・サイクル



注1. 外部バス幅８ビットのとき

2. 外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

4.4　メモリ・ブロック制御機能

V55PIは，１Mバイトの基本メモリ空間と16Mバイトの拡張メモリ空間を持っています。このようなメモリ空間に対応した外部メモリを接続するために，V55PIではメモリ空間を複数のメモリ・ブロックに分割して制御します。たとえば，基本メモリ空間は高速で小容量なROM，SRAMなどに適応するためのサイズを小刻みに指定可能なメモリ・ブロックとして，また，拡張メモリ空間は低速で大容量のROM，DRAMなどに適応するためのメモリ・ブロックとして使用することができます。

4.4.1　メモリ・ブロック・コントロール・レジスタ

V55PIでは，基本メモリ空間（000000H-0FFFFFH）をメモリ・サイズ可変の最大４ブロックに分割し，高位の拡張メモリ空間部分（100000H-FFFFFFH）をメモリ・サイズ可変に２分割し，ブロックごとにウエイト制御を行います。基本メモリ空間の各ブロックのメモリ・サイズはメモリ・ブロック・コントロール・レジスタ（MBC）で指定します。

MBCレジスタは基本メモリ空間を最大４ブロック（ブロック０-ブロック３）に，拡張メモリ空間を２ブロック（ブロック４-ブロック５）に分割指定する８ビットのレジスタです。

MBCレジスタはブロック０-２およびブロック４の上限値をそれぞれ２ビットで設定します。ブロック３の上限アドレスは0FFFFFHに固定されています。

ブロック上限アドレスの指定値が下位ブロックの上限アドレスの指定値より小さい場合は，ブロック範囲は０となり，そのブロックは構成されません。

保守／廃止

	7	6	5	4	3	2	1	0
MBC	MB31	MB30	MB21	MB20	MB11	MB10	MB01	MB00

ブロック０の上限アドレス

MB01	MB00	上限アドレス
0	0	000000H（0 K）
0	1	01FFFFH（128 K）
1	0	03FFFFH（256 K）
1	1	07FFFFH（512 K）

ブロック１の上限アドレス

MB11	MB10	上限アドレス
0	0	03FFFFH（256 K）
0	1	07FFFFH（512 K）
1	0	0BFFFFH（768 K）
1	1	0FFFFFH（1 M）

ブロック２の上限アドレス

MB21	MB20	上限アドレス
0	0	09FFFFH（640 K）
0	1	0BFFFFH（768 K）
1	0	0DFFFFH（896 K）
1	1	0FFFFFH（1 M）

ブロック４の上限アドレス

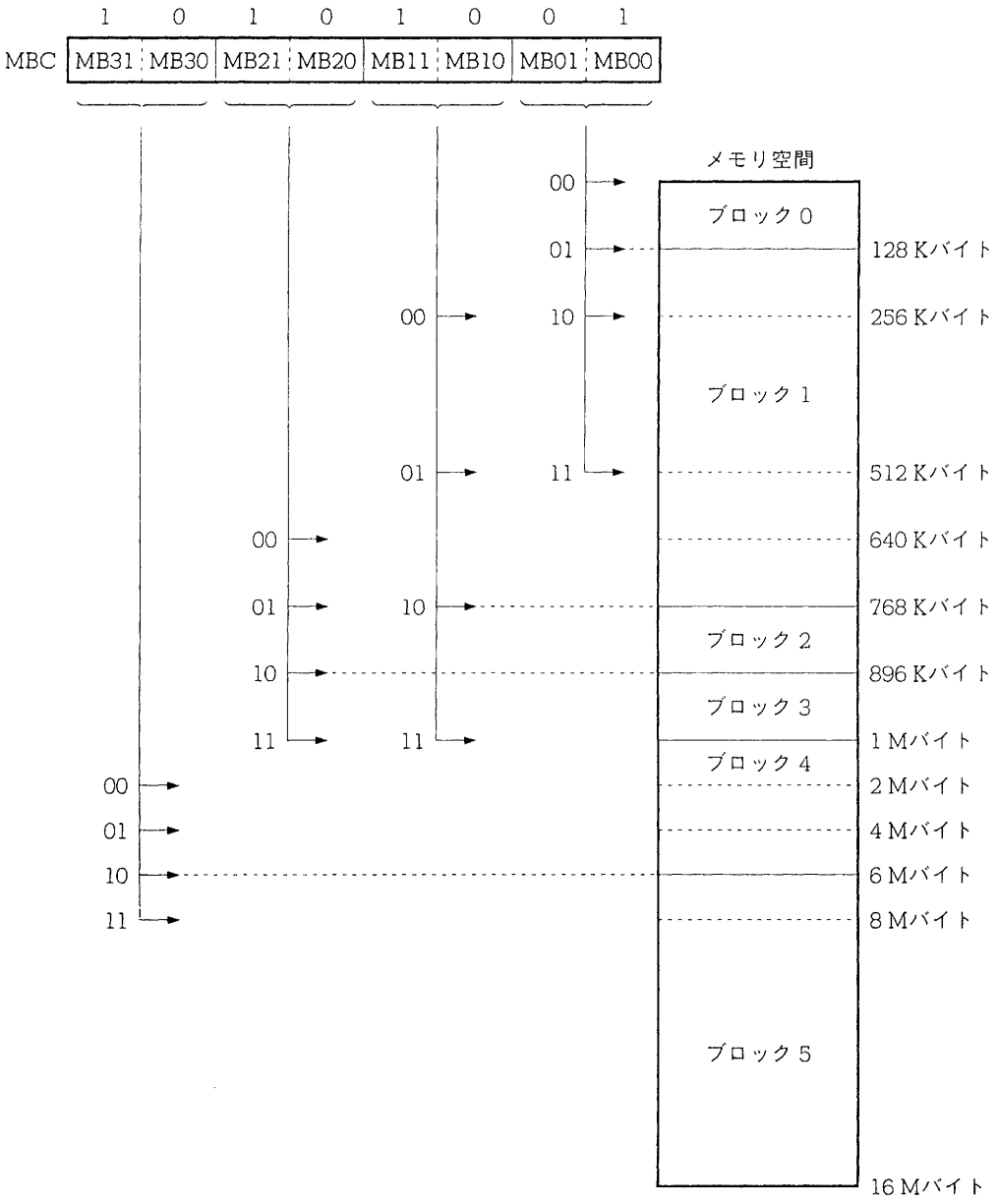
MB31	MB30	上限アドレス
0	0	1FFFFFH（2 M）
0	1	3FFFFFH（4 M）
1	0	5FFFFFH（6 M）
1	1	7FFFFFH（8 M）

備考　　はリセット後の状態を示します。

保守／廃止

図４－１に，MBCレジスタの値をA9Hに設定した場合のメモリ・ブロック構成を示します。

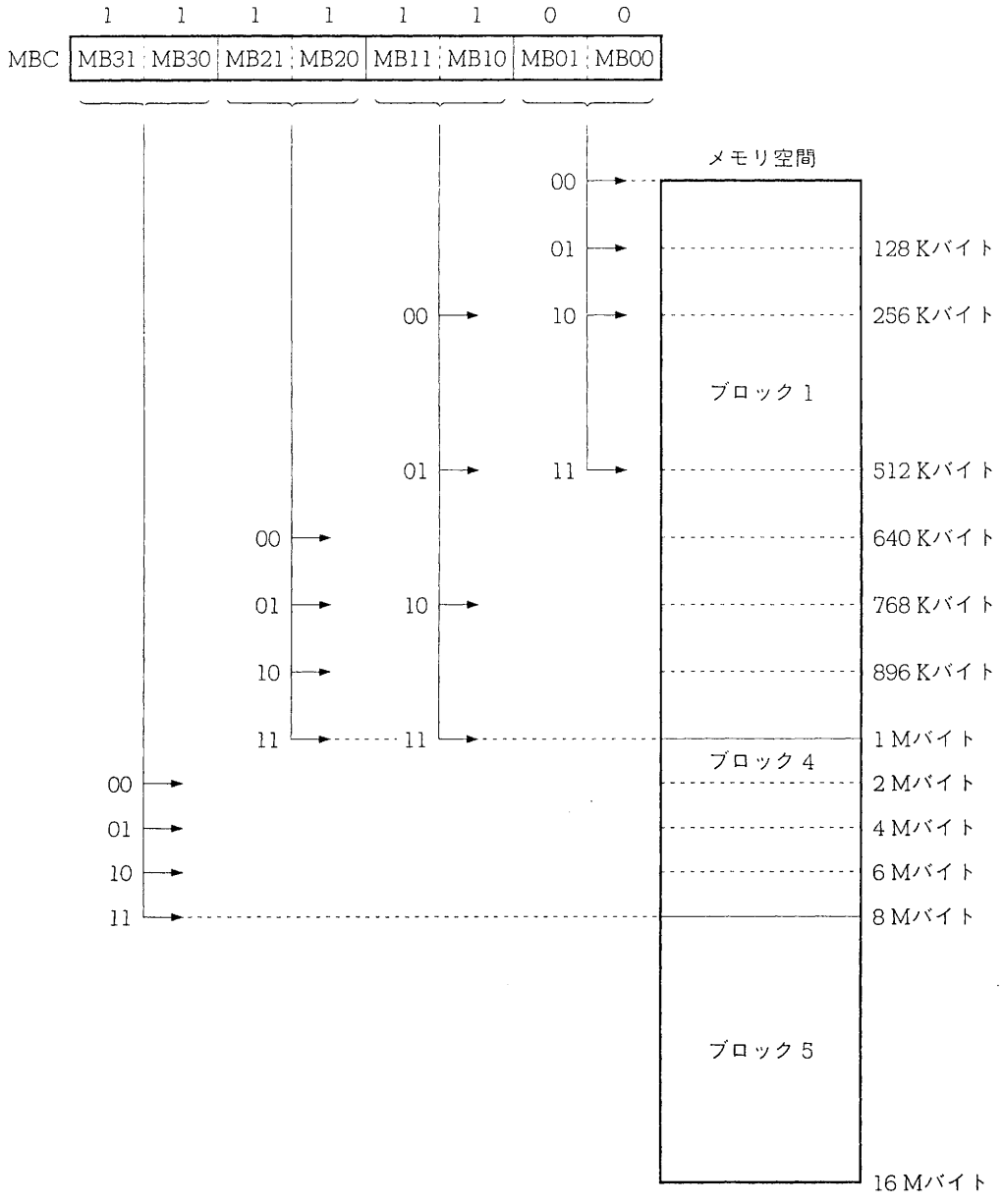
図４－１　メモリの分割制御



保守／廃止

MBCレジスタのリセット後の値は，FCHになります。リセット後のメモリ・ブロックの状態を図４－２に示します。

図４－２　リセット後のメモリ・ブロックの状態



保守／廃止

4.5　ウエイト機能

メモリ空間において，メモリ・ブロック単位で外部バス・サイクルにウエイト・ステートを挿入することができます。また，I/O空間においても，ウエイト・ステートの挿入が可能です。

リフレッシュ・サイクルについては，メモリ・ブロック4に対して指定したステート数のウエイト・サイクルが挿入されます。

4.5.1　アドレス・ウエイトとデータ・ウエイト

外部バス・サイクルに挿入できるウエイト・ステートには，アドレス・ウエイトとデータ・ウエイトの2種類があります。

(1) アドレス・ウエイト

バス・サイクルにおいてアドレス・ウエイトを挿入することにより，アドレス出力の期間（T1サイクル）が1クロック分延長されます（図4－10参照）。DRAMのプリチャージ時間を確保するために用います。

(2) データ・ウエイト

通常のウエイト・サイクルとして，T2サイクルとT3サイクルの間に挿入されます。バス・サイクルにおけるデータ入出力の期間を1クロック単位で延長することができます（図4－9参照）。外部メモリ，I/Oに対するアクセス時間を確保するために用います。

4.5.2　プログラマブル・ウエイト機能

V55PIでは，ソフトウエアで外部バス・サイクルにウエイト・ステートを挿入することができます。ただし，アドレス・ウエイトは，メモリ・ブロック1とメモリ・ブロック4に対してのみ挿入することができます。データ・ウエイトのウエイト・ステートの指定は，0，1，2，3ウエイトの中から選択できるほか，READY端子による追加制御が可能です。ブロックごとのウエイト・ステート数の指定は，PWC0レジスタ，PWC1レジスタで行うことができます。

(1) プログラマブル・ウエイト・コントロール・レジスタ（PWC0，PWC1）

バス・サイクルに挿入するウエイト・ステート数を制御するレジスタです。このレジスタは8/1ビット・アクセスが可能です。

PWC1レジスタのリセット後の値は，AAH，PWC0レジスタはEAHになります。

表中の □ は，リセット後の状態を示します。

	7	6	5	4	3	2	1	0	リセット時
PWC1	（ブロック3）		（ブロック2）		（ブロック1）		（ブロック0）		(AAH)
	DW31	DW30	DW21	DW20	DW11	DW10	DW01	DW00	

保守／廃止

	7	6	5	4	3	2	1	0	リセット時
PWC0	ブロック4	ブロック1	(I/O空間)		(ブロック5)		(ブロック4)		(EAH)
	AW1	AW0	IOW1	IOW0	DW51	DW50	DW41	DW40	

データ・ウエイト（DW，IOW）

DWn1/IOW1	DWn0/IOW0	ウエイト・ステート
0	0	0注1
0	1	1注2
1	0	2注2
1	1	3注2

注1. READY端子は無視されます。

2. READY端子による追加制御も可能です。

アドレス・ウエイト（AW）

AWn		ウエイト・ステート
AW0	0	挿入しない（ブロック1）
	1	挿入する（ブロック1）
AW1	0	挿入しない（ブロック4）
	1	挿入する（ブロック4）

4.5.3　READY端子によるウエイト制御

READY端子によるウエイト・ステート挿入は，プログラマブル・ウエイト・コントロール・レジスタ（PWCn）で設定のウエイト・ステート挿入とは独立に行います。

PWCnレジスタに指定したウエイト・ステート挿入数よりもREADY端子によるウエイト・ステート挿入数の方が多い場合は，READY端子によるウエイト・ステートの挿入を行います。逆に，PWCnレジスタに指定したウエイト・ステート数の方がREADY端子によるウエイト・ステート数よりも多い場合には，PWCnレジスタに指定したウエイト・ステート挿入を行います。

注意　**READY端子によるウエイト・ステート挿入を行う場合は，PWCnレジスタで1以上のウエイト・ステートを設定してください。PWCnレジスタで0ウエイトを設定すると，READY信号を無視し，ウエイト・ステートは挿入しません。**

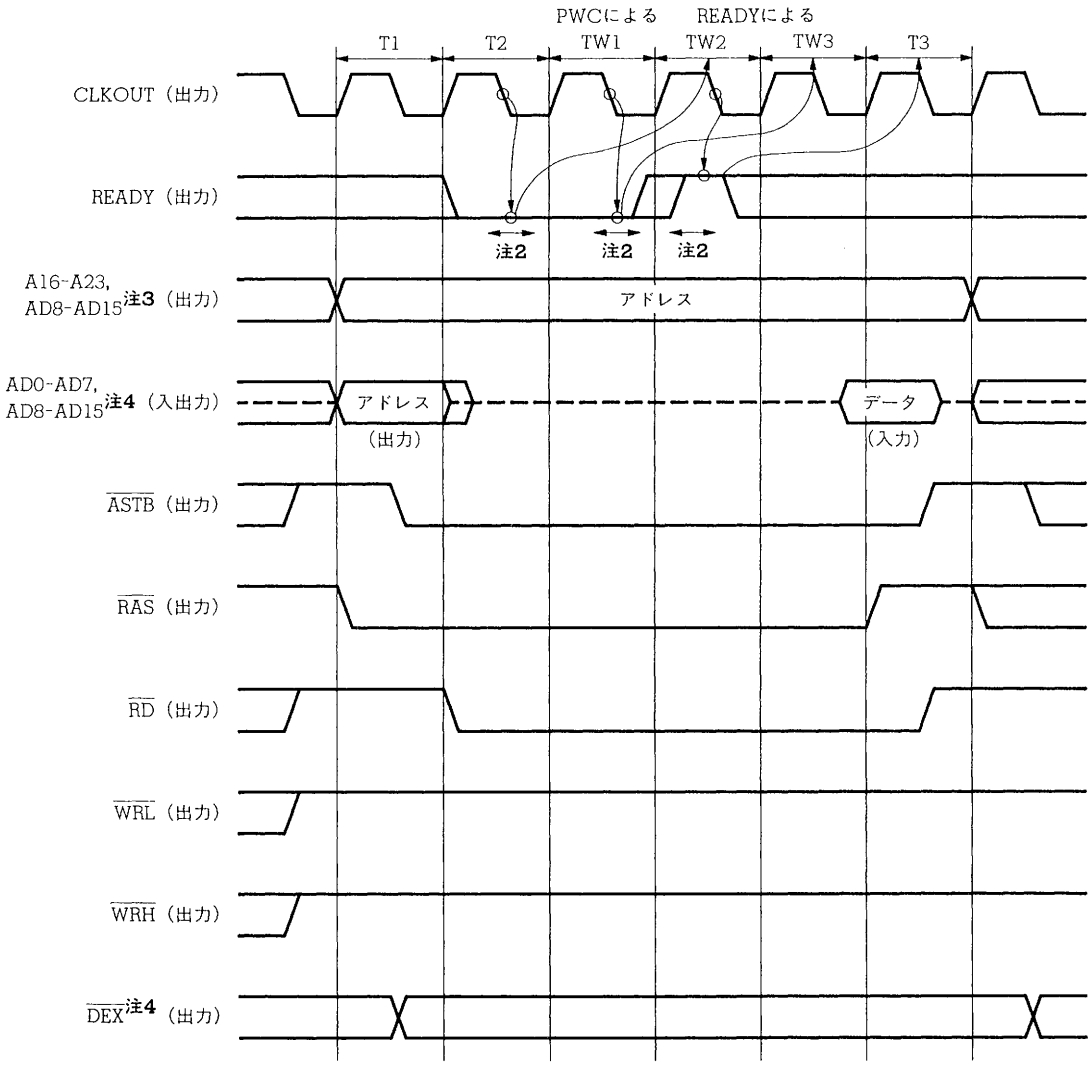
READY端子のサンプリングは，READY信号により挿入されるウエイト・ステートに対し2つ前のバス・サイクルで，CLKOUTの立ち下がりタイミングに同期して行われます。図4-3，4-4にサンプリングのタイミングを示します。

保守／廃止

図４－３　メモリ・リード・サイクル

READY端子によるウエイト挿入時

（アドレス・ウエイトなし，PWCレジスタによるデータ・ウエイト１のとき[注1]）



注1. PWCレジスタでデータ・ウエイトを２に設定した場合，T2サイクルのREADY信号はサンプリングの対象になりません。T2サイクルの状態によらずTW1サイクルのREADY状態だけで，TW3サイクルの挿入の有無が決定されます。

また，PWCレジスタでデータ・ウエイトを３に設定した場合も同様に，T2サイクル，TW1サイクルのREADY信号はサンプリングの対象になりません。

2. READY端子のサンプリング

3. 外部バス幅８ビットのとき

4. 外部バス幅16ビットのとき

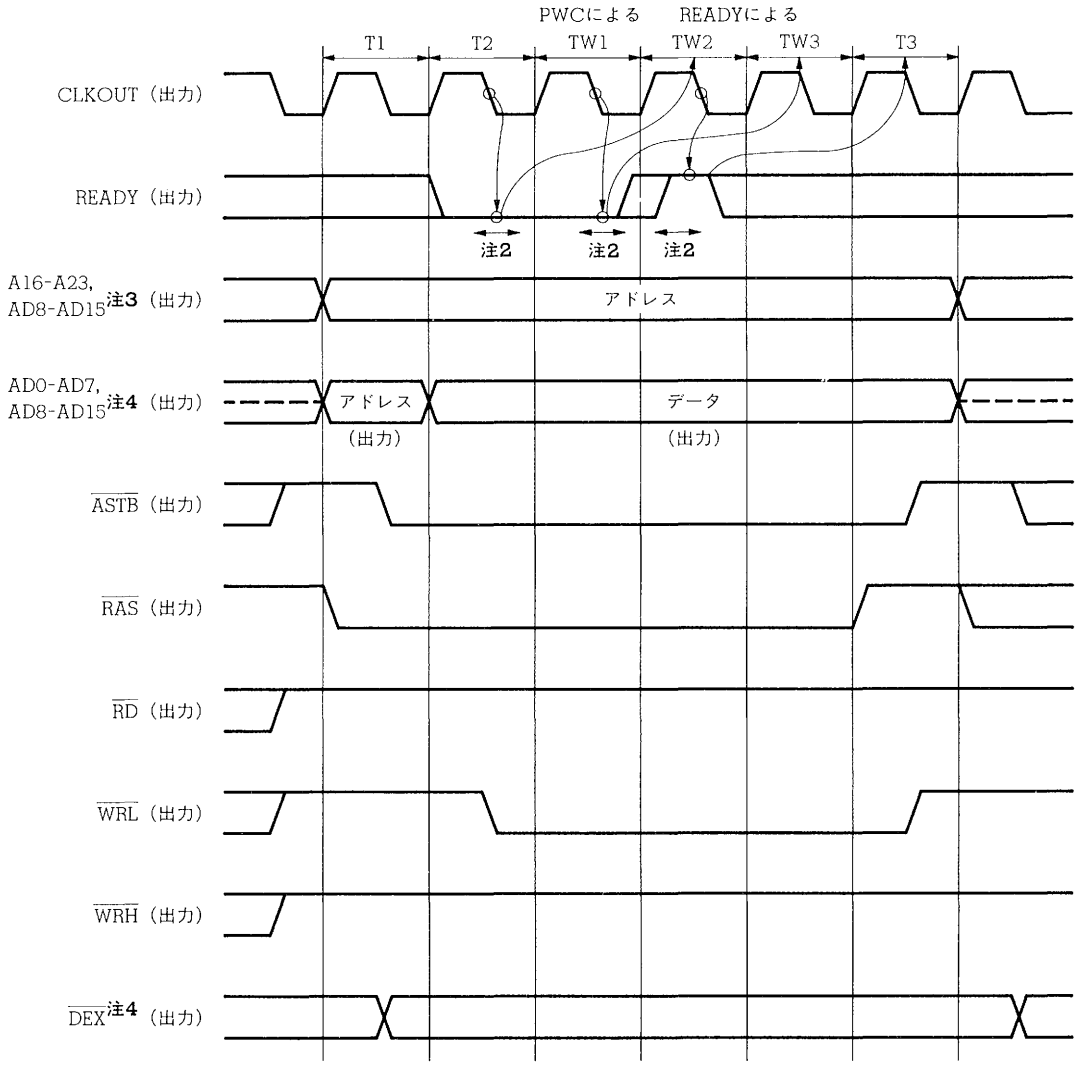
備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－４　メモリ・ライト・サイクル

READY端子によるウエイト挿入時

（アドレス・ウエイトなし，PWCレジスタによるデータ・ウエイト１のとき[注1]）



注1．PWCレジスタでデータ・ウエイトを２に設定した場合，T2サイクルのREADY信号はサンプリングの対象になりません。T2サイクルの状態によらずTW1サイクルのREADY状態だけで，TW3サイクルの挿入の有無が決定されます。

また，PWCレジスタでデータ・ウエイトを３に設定した場合も同様に，T2サイクル，TW1サイクルのREADY信号はサンプリングの対象になりません。

2．READY端子のサンプリング

3．外部バス幅８ビットのとき

4．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

4.5.4　DMA転送サイクルのウエイト

外部I/O-外部メモリのDMA転送におけるウエイト制御は，アドレス・ウエイト，データ・ウエイトに関してそれぞれ転送先，転送元のうち，挿入サイクル数の設定の大きいものに従います。

例　PWCレジスタにより，メモリ，I/Oのウエイト数を次のように設定した場合のDMA転送におけるウエイト数

- ●アドレス・ウエイト（メモリ）：1
- ●データ・ウエイト（メモリ）　：1
- ●データ・ウエイト（I/O）　　：2

→

- ●アドレス・ウエイト：1
- ●データ・ウエイト：2

4.5.5　SFRアクセス時のウエイト

SFRへアクセスする際，通常，リード時，ライト時ともに1ウエイトが挿入されます。ただし，次に示すSFRレジスタの場合は，1以上のウエイトが挿入されます。

表4-1　SFRアクセス時のウエイト（1ウエイト以上）

機　能	レ　ジ　ス　タ　名	ウエイト数	
		ライト時	リード時
割り込み	割り込みマスク・フラグ・レジスタ（MK0，MK1）	1	2
	割り込み要求制御レジスタ（IC09-IC37）		
	インサービス・プライオリティ・レジスタ（ISPR）		
	割り込みモード・コントロール・レジスタ（IMC）		
	外部割り込みモード・レジスタ（INTM0，INTM1）		
タイマ	タイマ・コントロール・レジスタ（TMC0，TMC1）	1，2注	2
	タイマ出力制御レジスタ（TOC0，TOC1）	1	2
	タイマ・レジスタ（TM0-TM3）	1-9注	2-9注
	タイマ・コンペア・レジスタ（CM00，CM01，CM10，CM11，CM20，CM21，CM22，CM23，CM30，CM31） タイマ・キャプチャ・レジスタ（CT00，CT01，CT10）	1，2注	2，3注
シリアル・インタフェース	送信ボー・レート・ジェネレータ・レジスタ（TxBRG0，TxBRG1） 受信ボー・レート・ジェネレータ・レジスタ（RxBRG0，RxBRG1）	1，2注	3，4注

注　アクセスするタイミングにより，挿入されるウエイト数が異なります。

保守／廃止

4.6　$\overline{\text{RAS}}$端子機能

$\overline{\text{RAS}}$端子機能は，ブロック１とブロック４に対して$\overline{\text{RAS}}$端子によりDRAMのRASタイミングを与える機能です。$\overline{\text{RD}}$，$\overline{\text{WRH}}$，$\overline{\text{WRL}}$端子はDRAMのCASタイミングを与え，$\overline{\text{ASTB}}$端子はDRAMのロウ／カラム・アドレス切り替えタイミングを与えます。

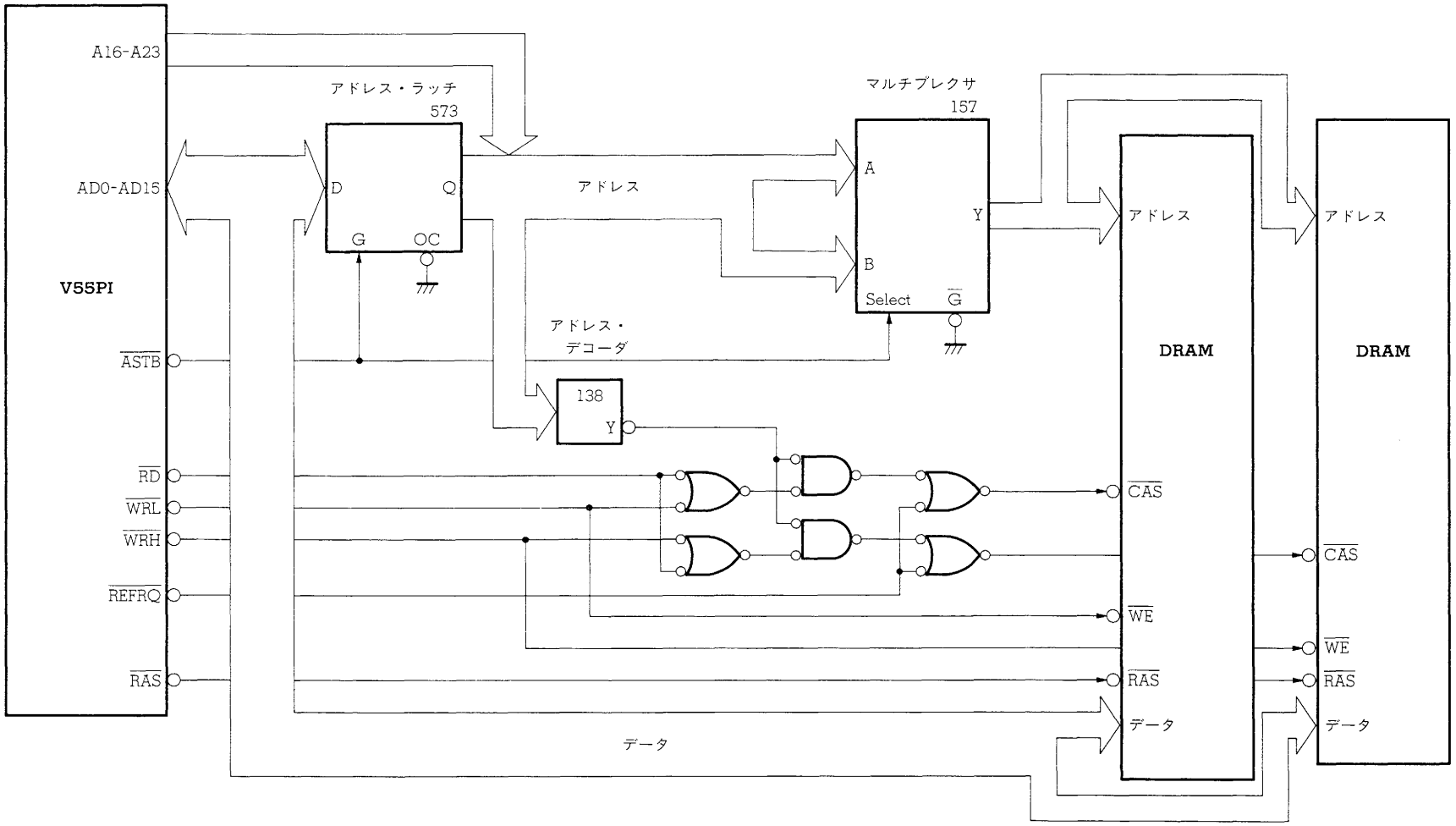
$\overline{\text{RAS}}$端子は，MBCレジスタにより指定されるブロック１とブロック４に対するバス・サイクル（リフレッシュ・サイクルを含む）でのみアクティブとなります。

なお，$\overline{\text{RAS}}$端子を直結してDRAMのロウ・アドレス・ストローブ信号として使用する場合，アドレス・ウエイトの挿入が不可欠となります。

$\overline{\text{RAS}}$端子を使用したDRAMの接続回路例を図４－５に示します。

保守／廃止

図4－5　DRAM接続回路例（アドレス・ウエイト挿入時）



備考　アドレス／データ・バスの信号線，およびアドレス・ラッチ，アドレス・デコーダ，マルチプレクサの数は，外付けのメモリのサイズにより異なります。

保守／廃止

4.7　リフレッシュ機能

V55PIは，DRAM，疑似SRAMのリフレッシュを行うために次のような機能を持っています。

○一連のバス・サイクル中に定期的にリフレッシュ・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・サイクル中にウエイト・ステートを挿入する機能

4.7.1　リフレッシュ・モード・レジスタ（RFM）

RFMレジスタは，リフレッシュ動作を制御する８ビットのレジスタです。

リフレッシュの周期は，タイム・ベース・カウンタの出力タップから選択できます。

なお，ほかのバス・サイクルによりリフレッシュ要求が保留されている状態で次のリフレッシュ要求が発生した場合，最後のリフレッシュ要求のみが有効となります。

RFMレジスタのリセット後の値は77Hになります。

保守／廃止

	7	6	5	4	3	2	1	0
RFM	0	HLDRF	HLTRF	RFEN	0	RFT2	RFT1	RFT0

リフレッシュの周期の指定

RFT2	RFT1	RFT0	リフレッシュ周期		
				ϕ = 12.5 MHz時	ϕ = 16 MHz時
0	0	0	64/ϕ	5.12 μs	4.0 μs
0	0	1	128/ϕ	10.24 μs	8.0 μs
0	1	0	192/ϕ	15.36 μs	12.0 μs
0	1	1	256/ϕ	20.48 μs	16.0 μs
1	0	0	8/ϕ	0.64 μs	0.5 μs
1	0	1	16/ϕ	1.28 μs	1.0 μs
1	1	0	24/ϕ	1.92 μs	1.5 μs
1	1	1	32/ϕ	2.56 μs	2.0 μs

（ϕ：システム・クロック）

リフレッシュ・サイクルの自動挿入の指定

RFEN	リフレッシュ・サイクル自動挿入
0	自動挿入を禁止
1	自動挿入を許可

HALTモード時の

リフレッシュ・サイクルの自動挿入の指定

HLTRF	リフレッシュ・サイクル自動挿入
0	自動挿入を禁止
1	自動挿入を許可[注1]

ホールド・モード時の

リフレッシュ・サイクルの自動挿入の指定

HLDRF	リフレッシュ・サイクル自動挿入
0	自動挿入を禁止
1	自動挿入を許可[注1,2]

注1. RFENビット＝0の場合はHLTRF，HLDRFビットの内容にかかわらず禁止となります。

2. 許可状態時には，リフレッシュ・タイミングごとに$\overline{\text{HLDAK}}$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。

備考 ▨ はリセット後の状態を示します。

保守／廃止

4.7.2　リフレッシュ・サイクルのウエイト制御

リフレッシュ・サイクル中にウエイト・ステートを挿入することができます。

プログラマブル・ウエイト・コントロール・レジスタ（PWC0）により，メモリ・ブロック４に対して，指定したウエイト・ステート数およびREADY端子によるウエイト・ステート数が挿入されます。

指定についての詳細は，**4.5　ウエイト機能**を参照してください。

4.7.3　リフレッシュ・アドレス

リフレッシュ・サイクルでアクティブになるバスは，AD0-AD15およびA16-A19です。

リフレッシュ・サイクル１回ごとに，外部８ビット・バス幅の場合は１ずつ×00000-×FFFFFHまで，外部16ビット・バス幅の場合は２ずつ×00001H-×FFFFFHまでカウント・アップされます。

リセットによる初期化後，外部８ビット・バス幅の場合は×00000Hから，外部16ビット・バス幅の場合は×00001Hから開始します。

また，外部16ビット・バス幅の場合，リフレッシュ・アドレスの最下位ビット（A0）は“1”固定となり，$\overline{\text{DEX}}$端子出力も“1”固定となります。

A20-A23は，リフレッシュ・サイクル中不定となります。

4.7.4　リフレッシュ・タイミング

DRAMのRASオンリー・リフレッシュ・サイクルとCASビフォアRASリフレッシュ・サイクルをサポートします。

$\overline{\text{REFRQ}}$端子がアクティブになることにより，リフレッシュ・サイクルであることを外部に知らせます。

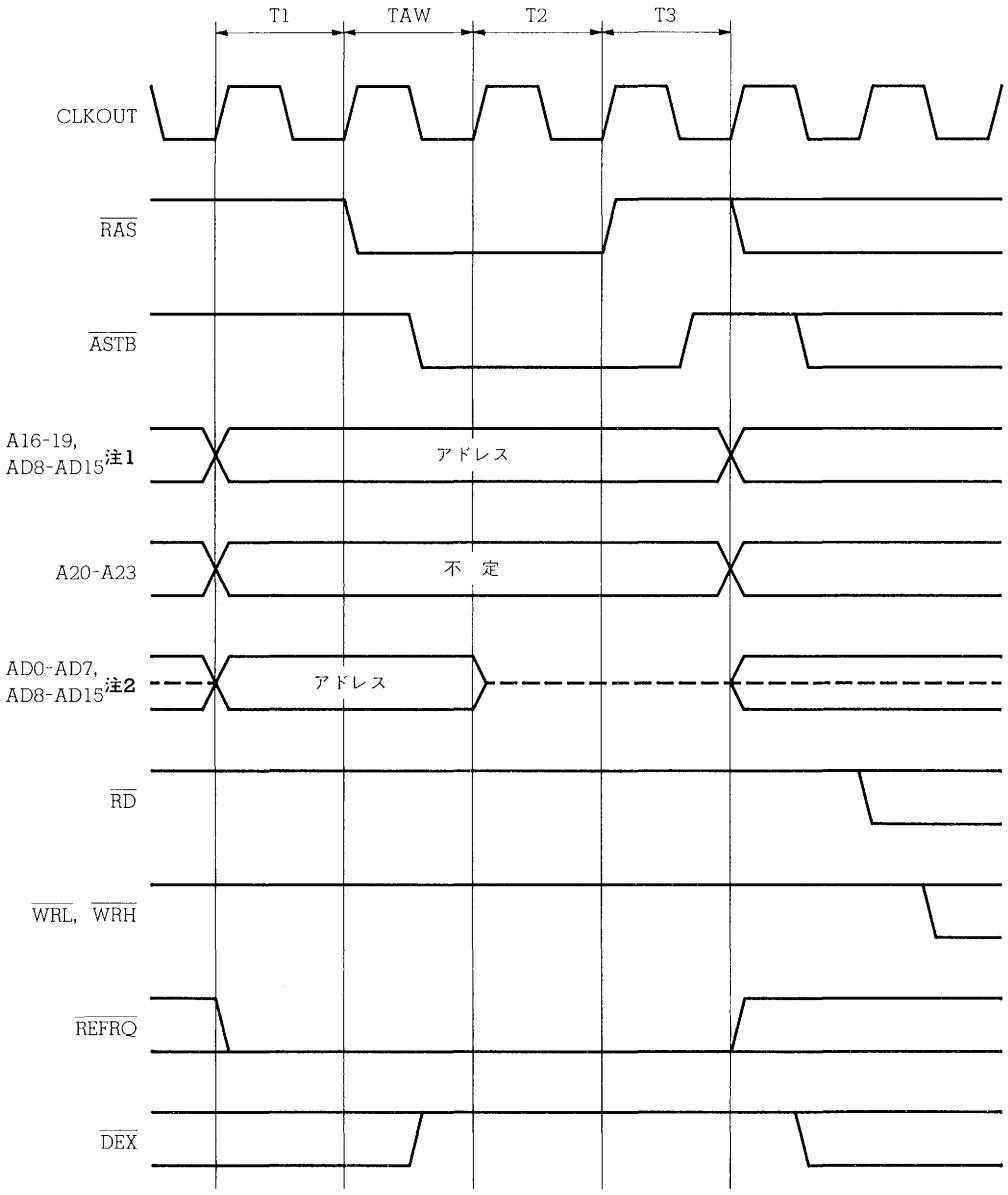
リセットによる初期化後リフレッシュ・サイクルが挿入されるまでの時間（$\overline{\text{RESET}}$↑→$\overline{\text{REFRQ}}$）の最大値は，１μs＋13.5T（T：$\phi=f_{XX}/8$モード時のサイクル回数）となります。

図４-６にリフレッシュ・サイクルのタイミングを示します。

保守／廃止

図４－６　リフレッシュ・サイクルのタイミング・チャート

（TAWはアドレス・ウエイトの挿入時のみ）



注1．外部８ビット・バス幅のとき

2．外部16ビット・バス幅のとき

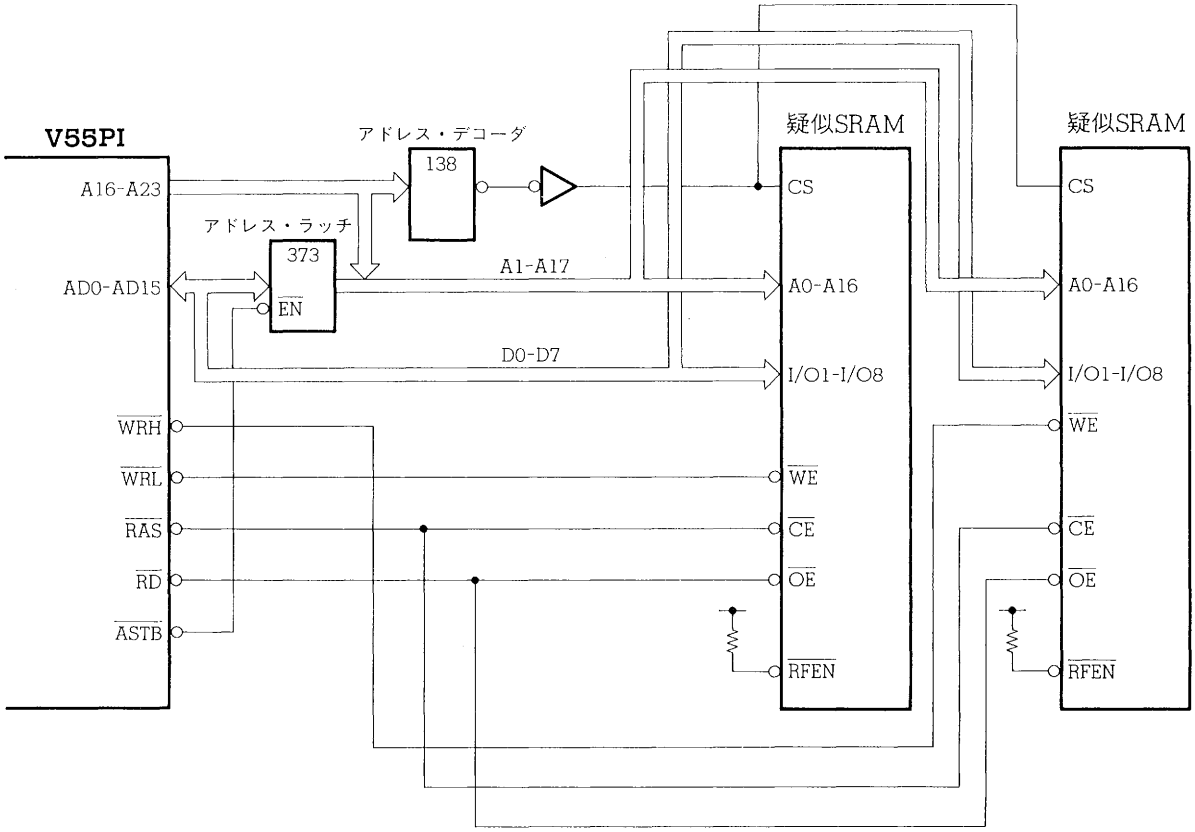
備考　破線はハイ・インピーダンスを示します。

保守／廃止

4.7.5　疑似SRAMの接続

疑似SRAMを接続する場合の回路例を図４－７に示します。

図４－７　疑似SRAM接続回路例（アドレス・ウエイト挿入時）



備考　アドレス・ラッチ，アドレス・デコーダの数は，外付けのメモリのサイズにより異なります。

保守／廃止

4.8　バス・ホールド機能

V55PIはバス・ホールド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，外部のデバイスが外部バスを使用したいことを知らせます。HLDRQ信号のハイ・レベルを検出すると，次のような動作が行われます。

(1) $\overline{\text{HLDAK}}$信号をロウ・レベルにする（外部デバイスにバスを解放したことを示す）。
(2) ホールド・モードに移行する。
　ホールド・モード中は，常にHLDRQ信号をチェックします。
(3) HLDRQ信号のロウ・レベルを検出する。
(4) $\overline{\text{HLDAK}}$信号をハイ・レベルにする（外部デバイスにバスを解放していないことを示す）。
(5) ホールド・モードを解除する。
(6) 命令の実行を再開する。

HALTモード中でも，バス・ホールドの要求を受け付けます。HLDRQ信号がロウ・レベルになり，ホールド・モードが解除されるとHALTモードに戻ります。なお，次の場合は，バス・ホールド要求を受け付けません。

- STOPモード中
- BUSLOCKプリフィクス命令に続く1命令の実行中

ホールド・モード時のリフレッシュ・サイクルの挿入は，リフレッシュ・モード・レジスタ（RFM）で指定することにより行うことができます。

まず，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・レベルにします。HLDRQ信号がロウ・レベルになったことを確認したのち，リフレッシュ・サイクルを発生します。このあと$\overline{\text{HLDAK}}$信号を再びロウ・レベルにしたときの動作は，次のとおりです。

- HLDRQ信号がハイ・レベルになったとき：ホールド・モードに再度移行する。
- HLDRQ信号がロウ・レベルのまま　　：ホールド・モードが解除され，命令の実行を再開する。

★　**注意　リセット中にHLDRQ信号がアクティブになる場合，リセット解除後のリセット・ベクタ・リードは行われず，ホールド・モードになります。**

保守／廃止

4.9　バス使用権

V55PIのバス使用の優先順位は，次のようになります。

ただし，割り込みアクノリッジ・サイクル注の実行中はリフレッシュ・サイクル，ホールド・モードへの移行，DMA転送サイクルは一時的に保留されます。なお，STOPモード時は，バス・サイクルは発生しません。

注　CPUが割り込み要求を受け付けたとき，割り込み処理ルーチンへ分岐するまでの前処理の期間

(1) ホールド・モード

BUSLOCKプリフィクスに続く１命令の実行中，および割り込みアクノリッジ・サイクル中以外では，ホールド・モードに移行します。

(2) リフレッシュ・サイクル

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（ホールド・モード中のリフレッシュ・サイクルの自動挿入が許可されている場合）。

(3) DMA転送サイクル

内蔵のDMAコントローラによるDMA転送のためのバス・サイクルです。

(4) 他のバス・サイクル

V55PIのメモリ操作命令の実行，およびプログラム・コードのフェッチ動作によるバス・サイクルです。

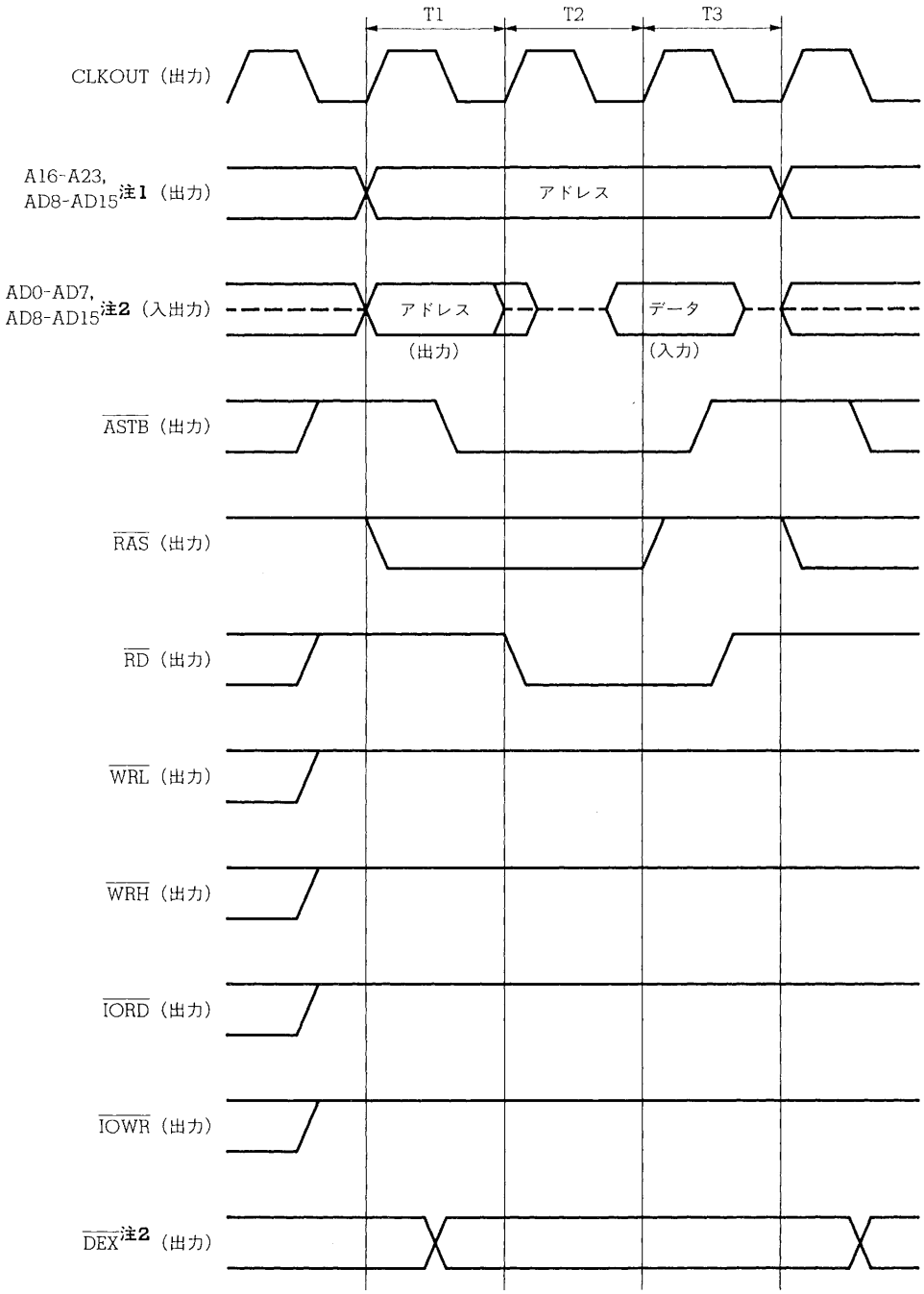
4.10　バス・タイミング

図４－８から図４－19に主なバス・タイミングを示します（リフレッシュ，DMA転送のタイミングを除く）。

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，アドレス／データ・バスの出力は不定です。

保守／廃止

図４－８　メモリ・リード・サイクル



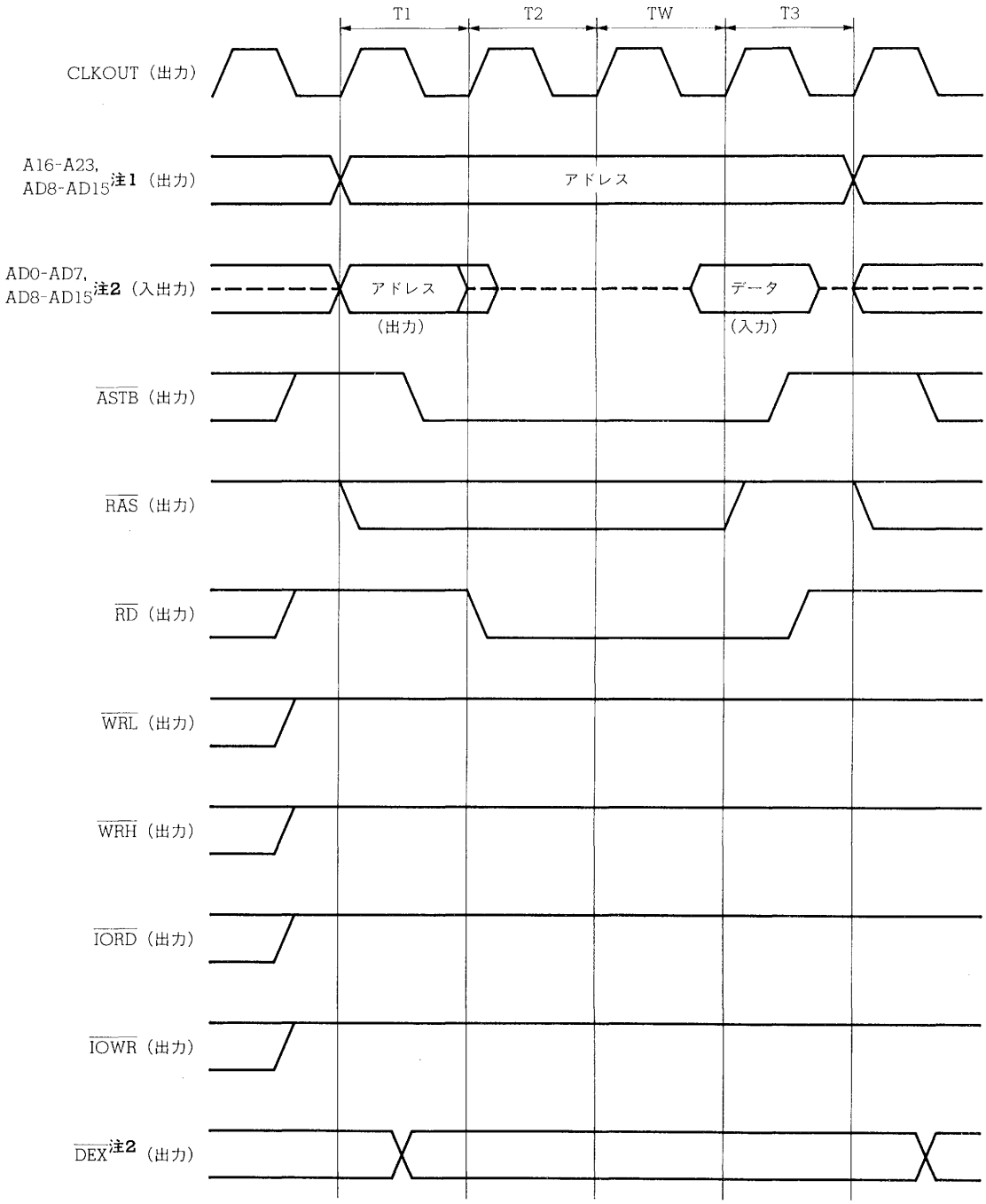
注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－９　メモリ・リード・サイクル
データ・ウエイト（１ウエイト）挿入時



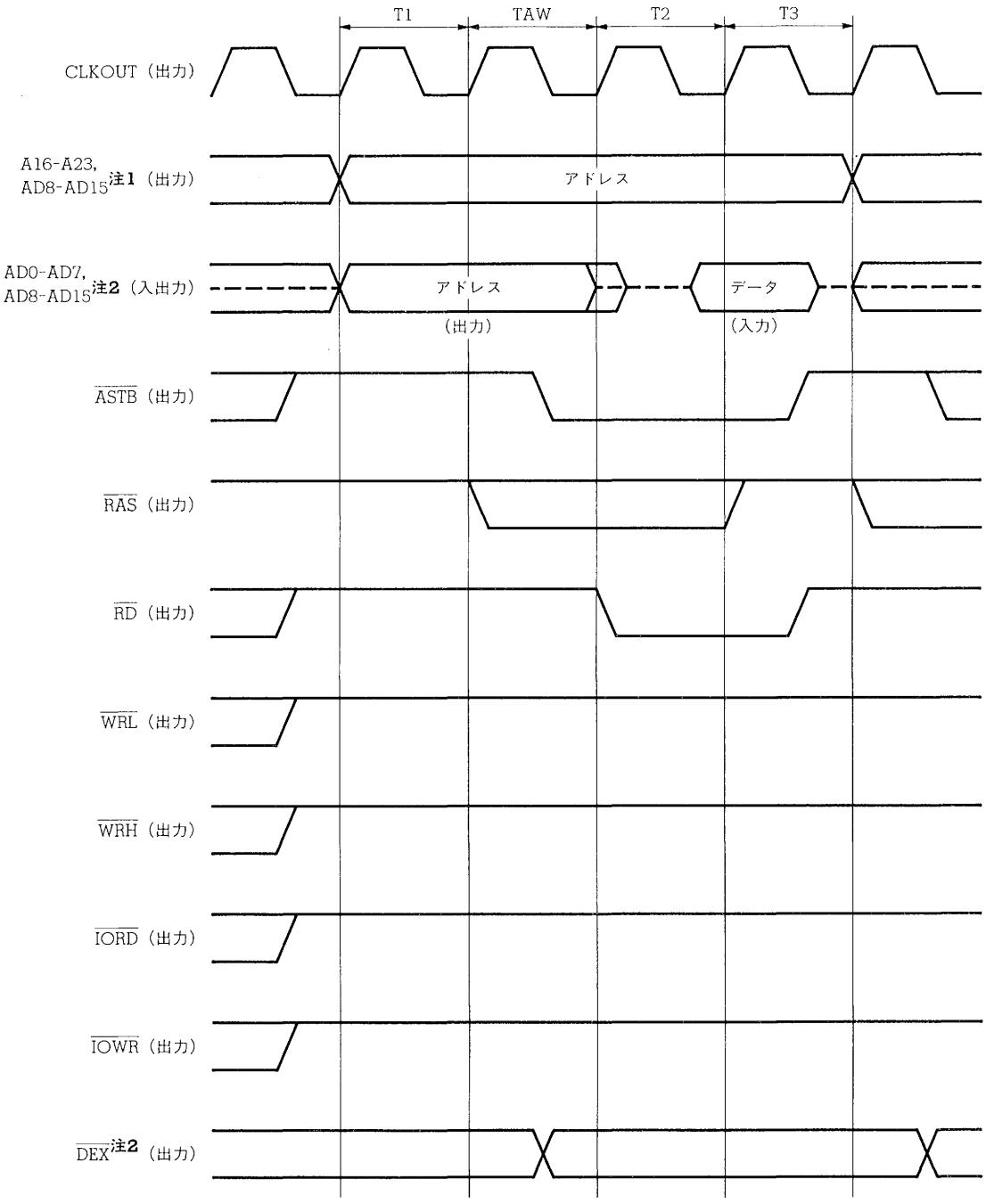
注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－10　メモリ・リード・サイクル
アドレス・ウエイト挿入時



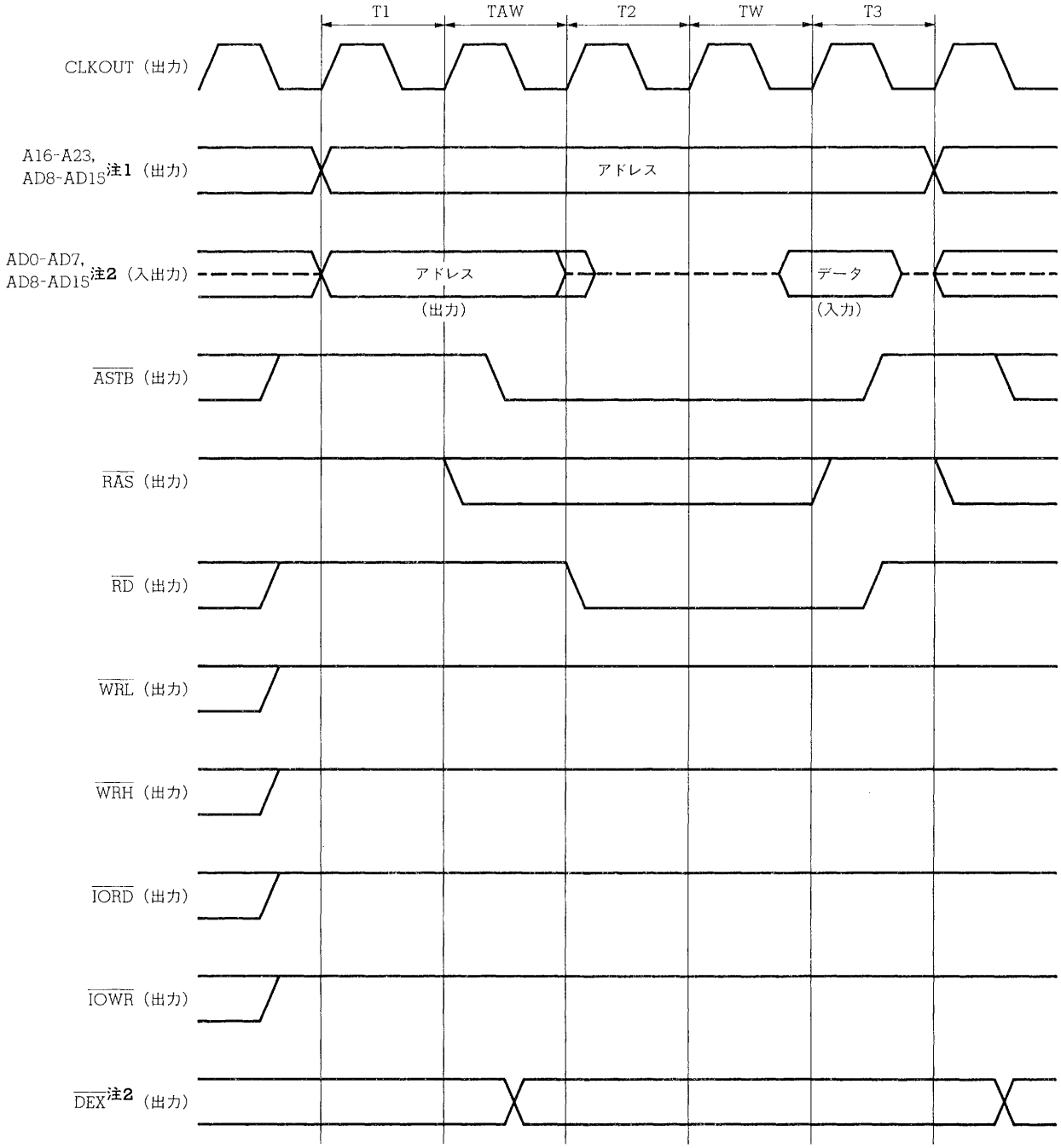
注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－１１　メモリ・リード・サイクル
データ・ウエイト（１ウエイト）
アドレス・ウエイト挿入時



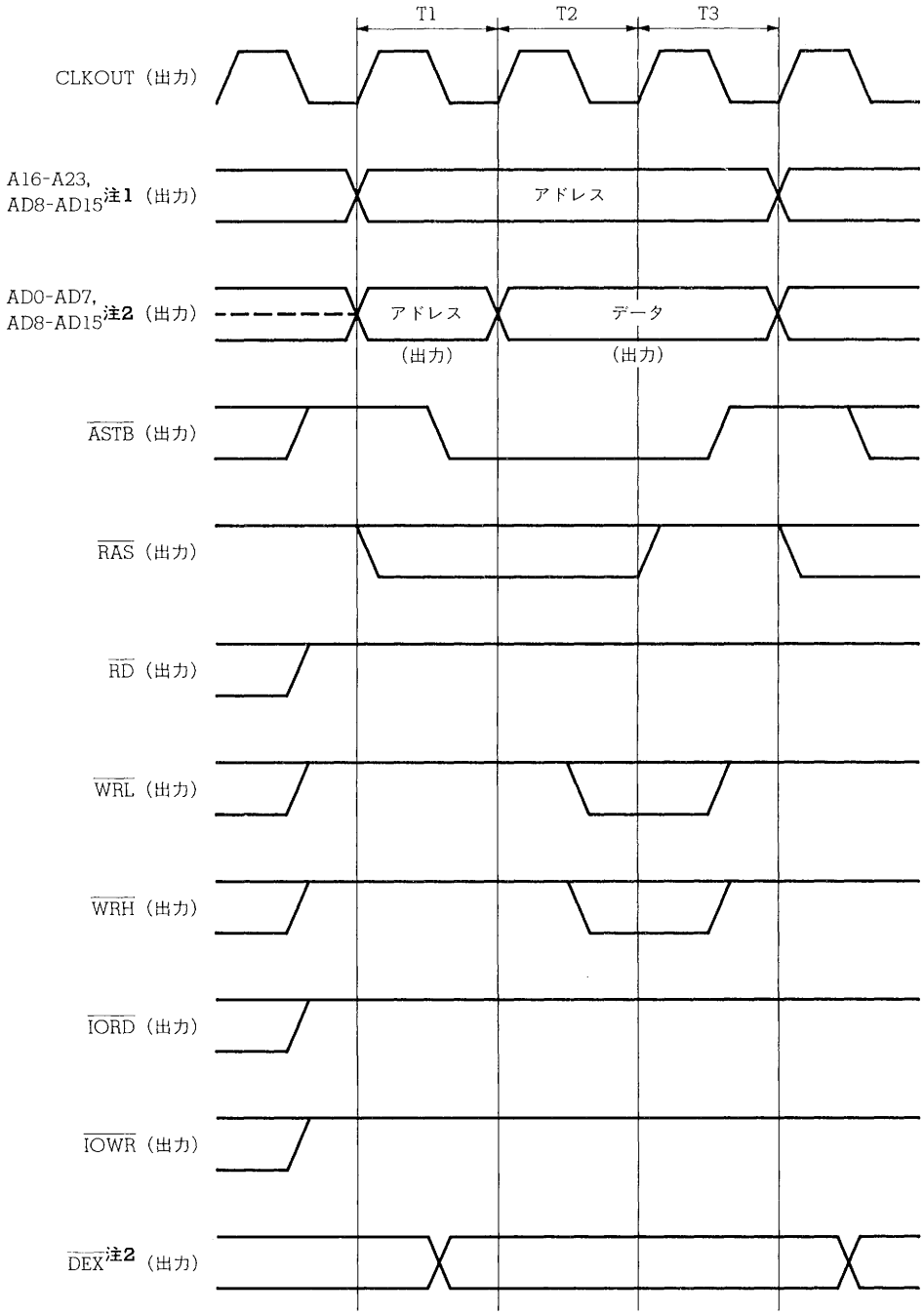
注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－12　メモリ・ライト・サイクル



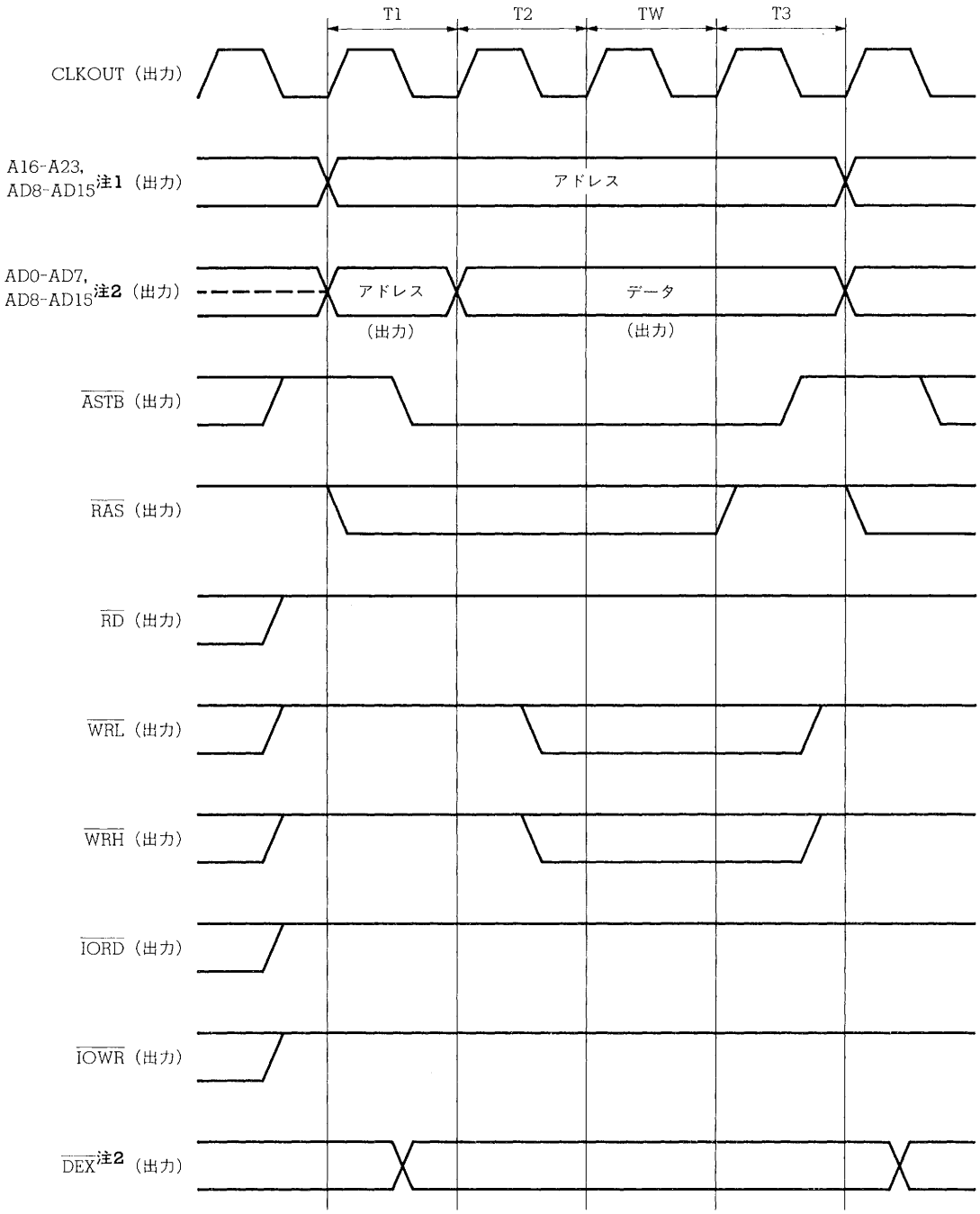
注1．外部バス幅８ビットのとき

2．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－13　メモリ・ライト・サイクル
データ・ウエイト（１ウエイト）挿入時



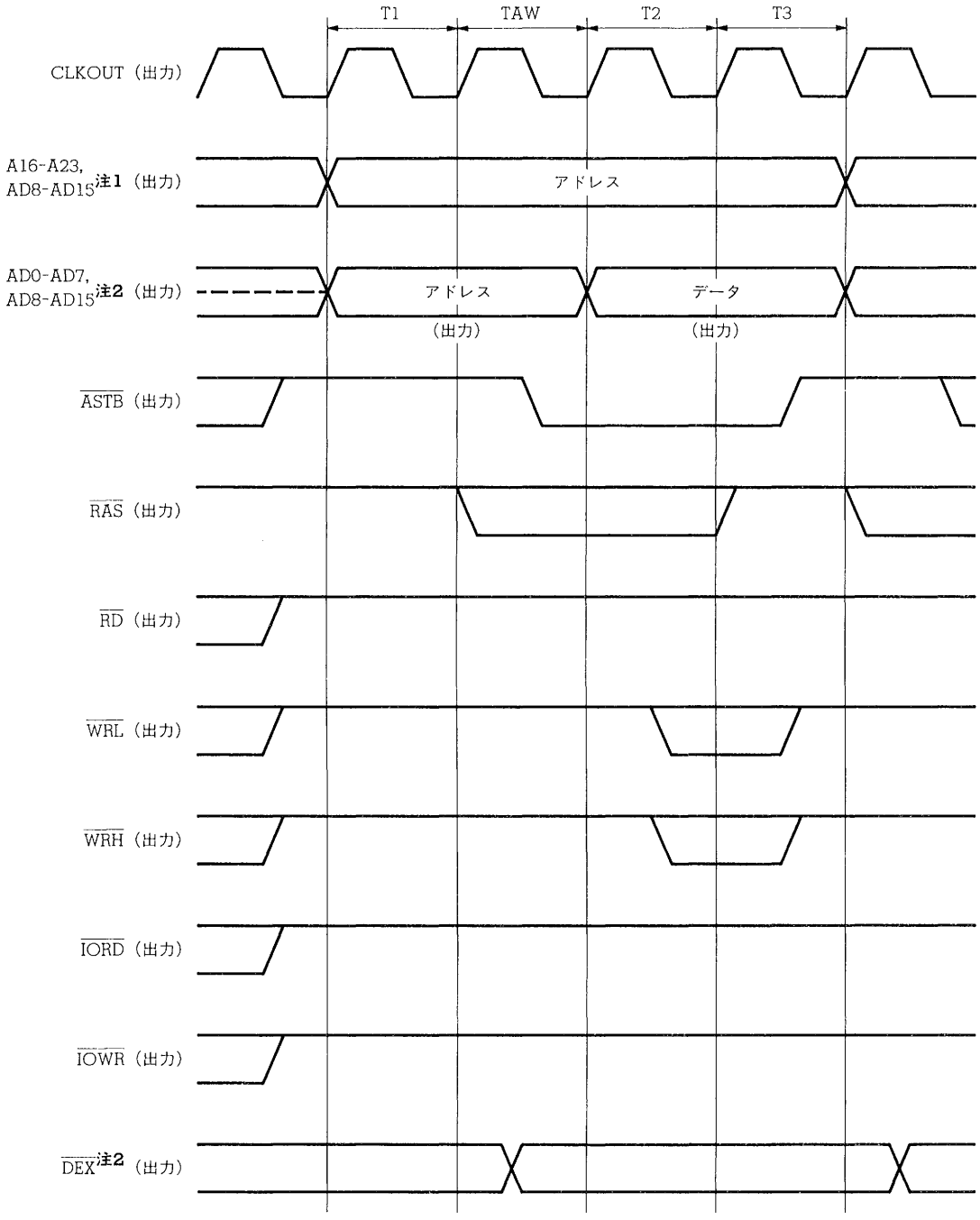
注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－14　メモリ・ライト・サイクル
アドレス・ウエイト挿入時



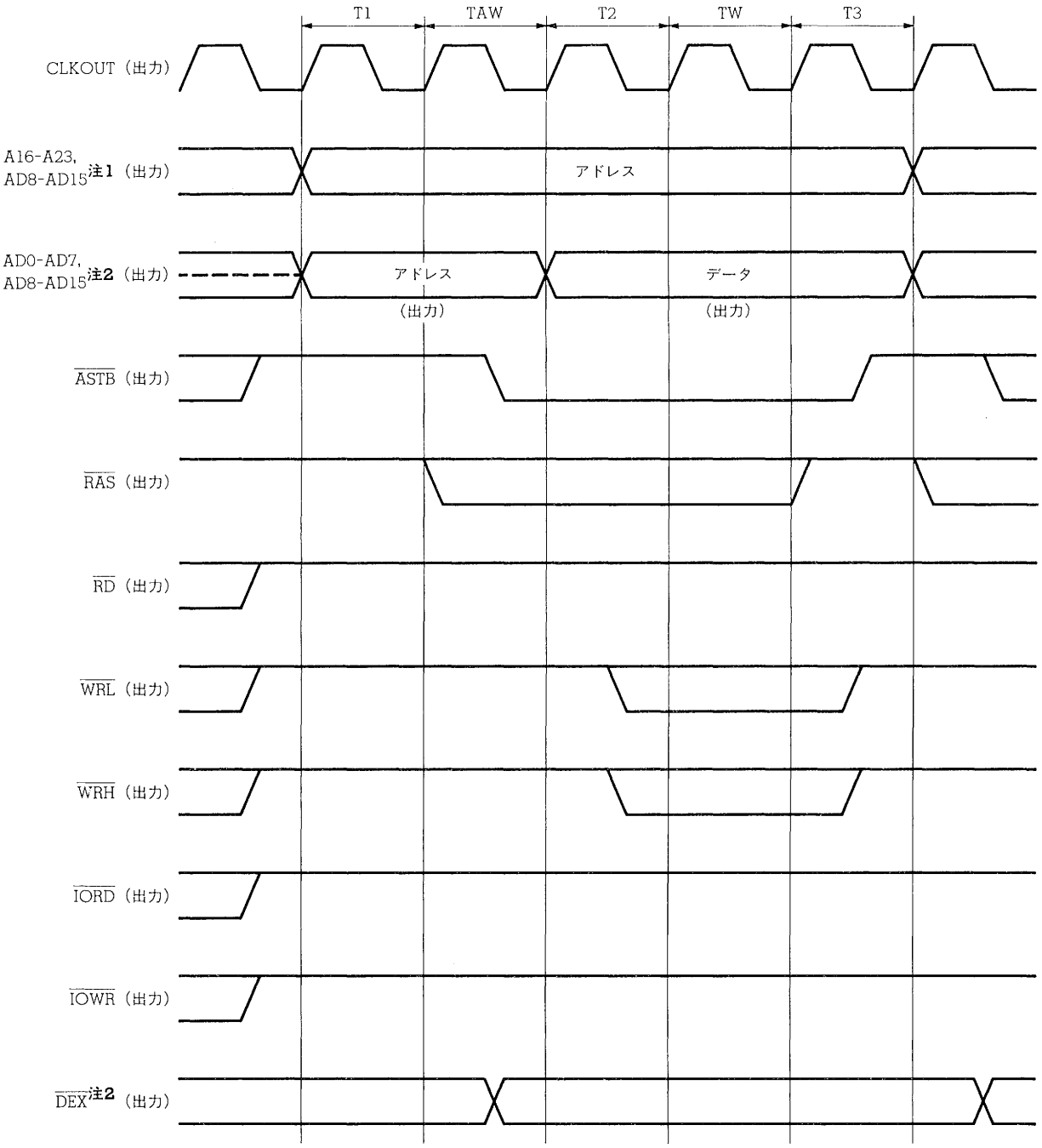
注1．外部バス幅８ビットのとき

2．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－15　メモリ・ライト・サイクル
データ・ウエイト（１ウエイト）
アドレス・ウエイト挿入時



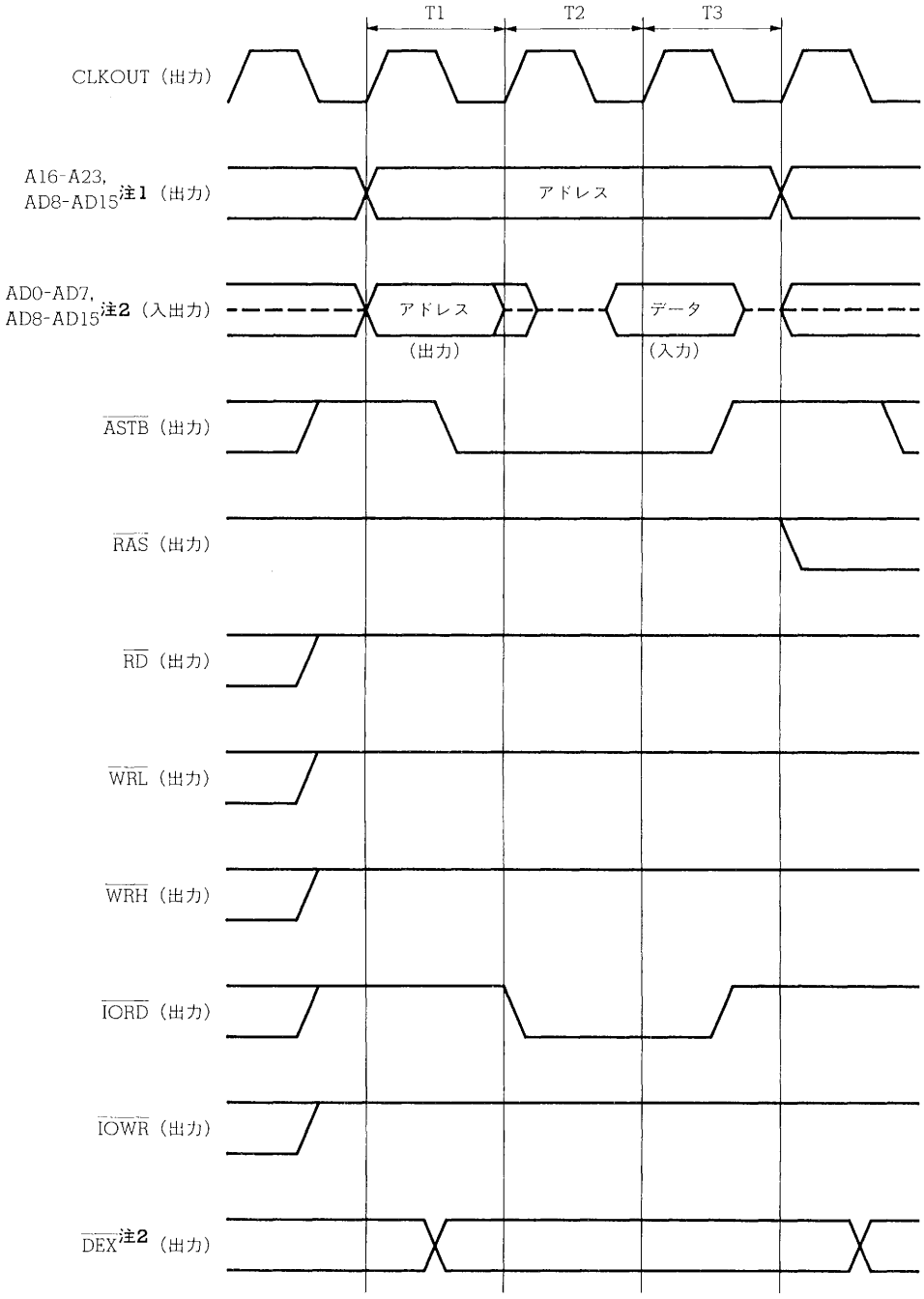
注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－16　I/Oリード・サイクル



注1．外部バス幅８ビットのとき

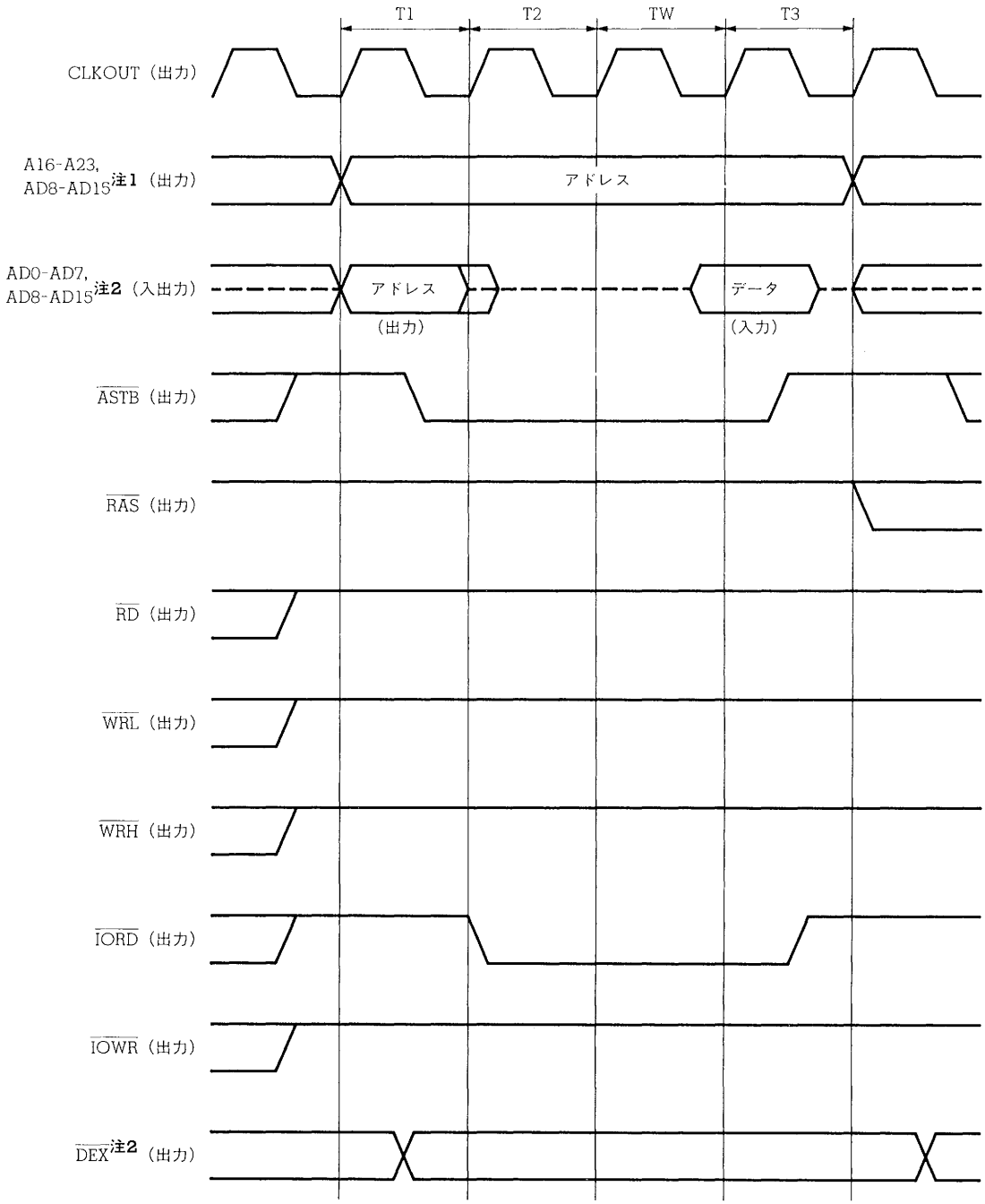
2．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－17　I/Oリード・サイクル

データ・ウエイト（１ウエイト）挿入時



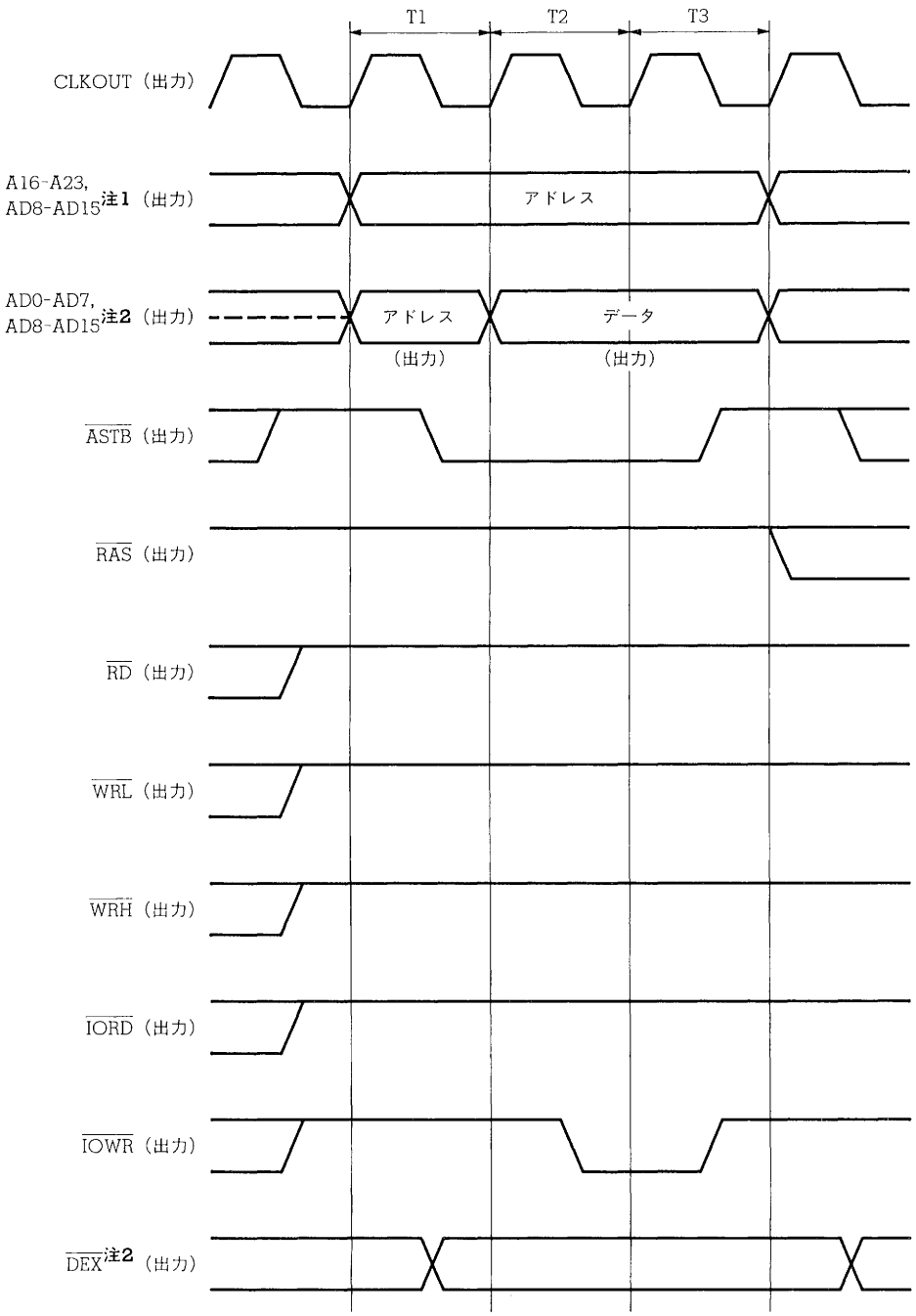
注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－18　I/Oライト・サイクル



注1．外部バス幅８ビットのとき

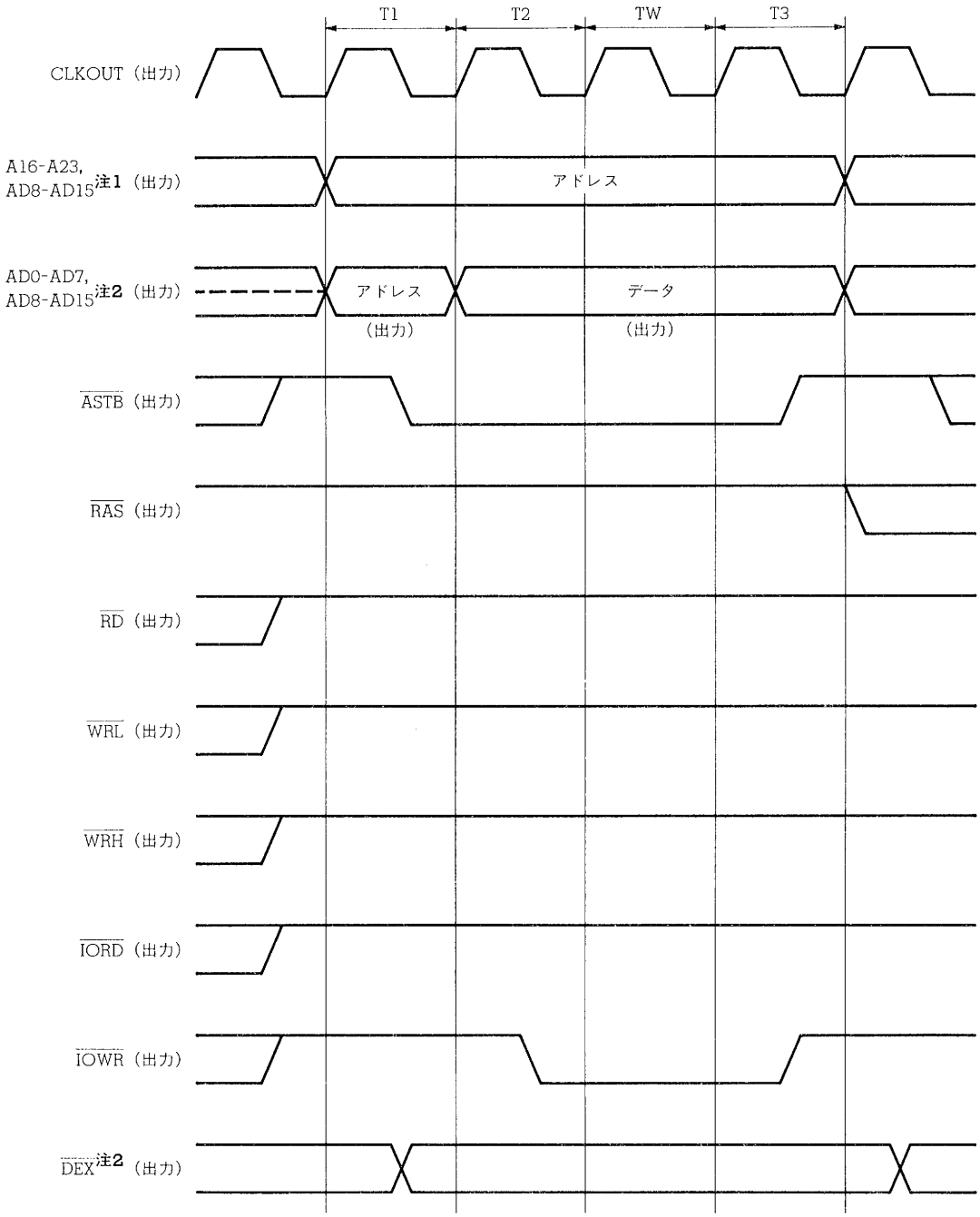
2．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図４－19　I/Oライト・サイクル

データ・ウエイト（１ウエイト）挿入時



注１．外部バス幅８ビットのとき

２．外部バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

〔メ　モ〕

保守／廃止

第５章　割り込み機能

V55PIは，内部19要因，外部６本，合計25要因のマスカブルなハードウエア割り込み要求を制御する強力な割り込みコントローラ（INTC）を内蔵しています。割り込みコントローラは，プログラマブルな優先順位に基づき，多重割り込み処理を制御します。また，割り込みの処理形態として，ベクタ割り込み機能，マクロ・サービス機能，レジスタ・バンク切り替え機能を用意しています。

5.1　特　　徴

V55PIの割り込み機能は，次のような特徴を持っています。

○割り込み要求に対する処理形態の充実

- ●ベクタ割り込み機能：ベクタ・テーブルにより指定される割り込み処理ルーチンへ分岐
- ●レジスタ・バンク切り替え機能：レジスタ・バンク自動切り替えによる高速割り込み応答
- ●マクロ・サービス機能：マイクロプログラム（ファームウエア）による高速割り込み処理

○プログラマブルな４レベルの優先順位制御

○優先順位に従った割り込み多重処理制御

○V55PIの内蔵周辺ハードウエアに密着した豊富なマクロ・サービス機能（次の７モード）

EVTCNT　：イベント・カウント処理
BLKTRS　：特殊機能レジスタと外部メモリ・バッファとの間のデータ転送
BLKTRS-C　：　〃　（転送データ検出機能付き）
DTACMP　：特殊機能レジスタの状態検出
DTADIF　：タイマ・キャプチャ機能による時間計測
RTOPTRN　：リアルタイム出力ポートの自動制御
DTACMP-M：外部I/Oとメモリ間のデータ転送

○外部割り込み要求入力：７本（NMI，INTP0-INTP5）

○マスカブル割り込み要求は個々にマスク可能

このような特徴を持つ割り込みコントローラの動作制御は，割り込みモード・コントロール・レジスタ（IMC），割り込みマスク・フラグ・レジスタ（MK0，MK1），外部割り込みモード・レジスタ（INTM0，

保守／廃止

INTM1），およびマスカブル割り込み要因ごとに用意されている割り込み要求制御レジスタ（IC××）で行います。

また，割り込み制御命令には，許可／禁止を制御するEI，DI命令，割り込みからの復帰を指示するRETI，RETRBI命令，内蔵の割り込みコントローラに割り込み処理の終了を知らせるFINT命令があります。

5.2　割り込み要因

V55PIが扱う割り込み要求は次の４種類に分類されます。

（1）ノンマスカブル割り込み要求（NMI，ウォッチドッグ・タイマ）
（2）マスカブル割り込み要求　　（内蔵周辺回路，外部端子からの割り込み要求）
（3）ソフトウエア割り込み要求　（BRK，CHKIND命令，除算エラーほか）
（4）例外トラップ　　（特殊書き込みエラー）

表５－１に割り込み要因の一覧を示します。

このうち，割り込みコントローラの制御する割り込み要因は，ノンマスカブル割り込みとマスカブル割り込みの計27要因です。命令実行により割り込みを発生するソフトウエア割り込みと例外トラップは割り込みコントローラの制御の対象になりません。

保守／廃止

表5－1 割り込み要因一覧（1/2）

割り込みの種類	ディフォールトの優先順位	割り込み要求信号	割り込み要因			ディフォールトのベクタ・テーブル番号	ベクタ・アドレス	マクロ・サービス	レジスタ・バンク切り替え	マクロ・サービス制御ワード・アドレス
			割り込み要求制御レジスタ	発生ソース	発生ユニット					
ノンマスカブル	1	NMI	—	NMI端子入力	—	2	00008H	なし	なし	—
	2	INTWDT		ウォッチドッグ・タイマのオーバフロー	WDT	8	00×20H	なし	なし	
マスカブル	3	INTP0	IC9	INTP0端子入力	外部	9	00×24H	あり	あり	012H
	4	INTP1	IC10	INTP1端子入力		10	00×28H	あり	あり	014H
	5	INTP2	IC11	INTP2端子入力		11	00×2CH	あり	あり	016H
	6	INTP3	IC12	INTP3端子入力		12	00×30H	あり	あり	018H
	7	INTP4	IC13	INTP4端子入力		13	00×34H	あり	あり	01AH
	8	INTP5	IC14	INTP5端子入力		14	00×38H	あり	あり	01CH
	9	INTCM00	IC16	CM00の一致検出	タイマ	16	00×40H	あり	あり	020H
	10	INTCM01	IC17	CM01の一致検出		17	00×44H	あり	あり	022H
	11	INTCM10	IC18	CM10の一致検出		18	00×48H	あり	あり	024H
	12	INTCM11	IC19	CM11の一致検出		19	00×4CH	あり	あり	026H
	13	INTCM21	IC20	CM21の一致検出		20	00×50H	あり	あり	028H
	14	INTCM31	IC21	CM31の一致検出		21	00×54H	あり	あり	02AH
	15	INTD0	IC22	DMAチャネル0_メイン	DMA	22	00×58H	あり	あり	02CH
	16	INTD0S	IC23	DMAチャネル0_サブ		23	00×5CH	あり	あり	02EH
	17	INTD1	IC24	DMAチャネル1_メイン		24	00×60H	あり	あり	030H
	18	INTD1S	IC25	DMAチャネル1_サブ		25	00×64H	あり	あり	032H

保守／廃止

表5－1　割り込み要因一覧（2/2）

割り込みの種類	ディフォールトの優先順位	割り込み要求信号	割り込み要因			ディフォールトのベクタ・テーブル番号	ベクタ・アドレス	マクロ・サービス	レジスタ・バンク切り替え	マクロ・サービス制御ワード・アドレス
			割り込み要求制御レジスタ	発生ソース	発生ユニット					
マスカブル	19	INTSER0	IC26	UART受信エラー（ch0）	シリアルI/F	26	00×68H	なし	あり	—
	20	INTSER1	IC27	UART受信エラー（ch1）		27	00×6CH	なし	あり	—
	21	INTSR0/INTCSI0	IC28	UART受信（ch0）/シリアル送受信（ch0）		28	00×70H	あり	あり	038H
								あり	あり	
	22	INTSR1/INTCSI1	IC29	UART受信（ch1）/シリアル送受信（ch1）		29	00×74H	あり	あり	03AH
								あり	あり	
	23	INTST0	IC30	UART送信（ch0）		30	00×78H	あり	あり	03CH
	24	INTST1	IC31	UART送信（ch1）		31	00×7CH	あり	あり	03EH
	25	INTSIT	IC32	STMの一致検出	SIT	32	00×80H	なし	あり	—
	26	INTPAI	IC36	パラレルI/F	パラレルI/F	36	00×90H	あり	あり	008H
	27	INTAD	IC37	A/Dコンバータ	A/Dコンバータ	37	00×94H	あり	あり	00AH
ソフトウエア	—	—	—	ディバイド・エラー	—	0	00000H	なし	なし	—
				BRKフラグ（シングルステップ）		1	00004H	なし	なし	
				BRK3命令		3	0000CH	なし	なし	
				BRKV命令		4	00010H	なし	なし	
				CHKIND命令		5	00014H	なし	なし	
				入出力命令（IBRKフラグ）		6	00018H	なし	なし	
				BRK imm8		注	00×注H	なし	なし	
				BRKCS命令		—	—	なし	あり	
例外トラップ				FPO命令／例外トラップ		7	0001CH	なし	なし	

注　0-255（0-FFH）の範囲で値を変更することが可能であることを示しています。

備考　“×”は，IMCレジスタのV0，V1ビットの値で決定されます。

保守／廃止

5.2.1　ノンマスカブル割り込み

ノンマスカブル割り込みはソフトウエアによる優先順位指定には無関係で，割り込み禁止状態(IE＝0)でも常に最優先で受け付けられる割り込みです。P10端子への外部有効エッジ入力検出(NMI)と内部ウォッチドッグ・タイマのオーバフロー信号出力（INTWDT）の２要因があります。

マスカブル割り込み要求と異なり，プログラマブル優先順位制御の対象にはなりません。NMIとINTWDTは，多重割り込み処理制御を受けませんので何レベルでもネスティングが可能です。

ノンマスカブル割り込み要求は通常のベクタ割り込みで処理されます。なお，ディフォールトの優先順位により，NMIの方がINTWDTより高い優先順位に決められており，NMIとINTWDTの要求が同時に発生したときにはNMIが先に受け付けられます。

(1) NMI

NMIはエッジで検出されます。エッジの方向は，割り込み制御レジスタの１つである外部割り込みモード・レジスタ０(INTM0)のES0(ビット０)で選択できます。ES0＝0にすると立ち下がりエッジ，ES0＝1にすると立ち上がりエッジで割り込みを発生します。

この割り込みはベクタ応答しかできず，ベクタ番号は２に固定されています。

NMIが受け付けられると割り込み禁止状態(IE＝0)となり，マスカブル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は受け付けます。**5.3.2　同時に割り込みが発生したときの優先順位制御，5.12　割り込み要求の処理動作例**を参照してください)。

NMI処理中に再度NMIに有効エッジが入力された場合は，実行しているNMI処理を中断し，その有効エッジに対応するNMI処理に移行します。

注意　以下，NMIと記述した場合は，ノンマスカブル割り込み要求の総称ではなく，P10端子への有効エッジ入力検出を示します。

(2) INTWDT

INTWDTはプログラムの暴走，システムの異常などを検出するための割り込みです。この割り込みはベクタ割り込みのみで処理され，初期値のベクタ番号は８に固定されています（ただし割り込みモード・コントロール・レジスタ（IMC）のV0，V1ビットでベクタ・アドレスを変更可能です)。

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，ほかの割り込みを禁止します(ただし，マクロ・サービスおよびNMIは受け付けられます。**5.3.2　同時に割り込みが発生したときの優先順位制御**を参照してください)。

詳細は，**第12章　ウォッチドッグ・タイマ機能**を参照してください。

5.2.2　マスカブル割り込み

V55PIには，外部６本，内部19要因の合計25要因のマスカブル割り込みがあり，割り込みコントローラにより多重割り込み処理制御を行います。

割り込みマスク・フラグ・レジスタ(MK0，MK1)内，また要因ごとの割り込み要求制御レジスタ(IC××)

保守／廃止

内にマスク・ビットがありますので，必要に応じ，割り込み要求をマスクすることができます。

また，マスカブル割り込みは割り込み要求制御レジスタ（IC××）で次の割り込み処理形態のどれかを指定することが可能です。

① ベクタ割り込み
② マクロ・サービス処理
③ レジスタ・バンク切り替え機能

また，PSW内のビット９割り込み許可フラグ（IEフラグ）は，すべてのマスカブル割り込み処理の許可／禁止を制御します。

IE＝0　すべてのマスカブル割り込みを禁止します。

IE＝1　マスカブル割り込み要求を受け付け可能です。

IEフラグは，割り込み要求を受け付けるとリセット(0)されます（なお，マクロ・サービスでは変化しません)。

マスカブル割り込みの処理形態は図５－１に示すとおりです。

保守／廃止

図５－１　マスカブル割り込みの処理形態

マスカブル割り込み処理形態

- 割り込みマスク・フラグ・レジスタ（MK0，MK1），または割り込み要求制御レジスタ（IC××）の割り込みマスク・フラグ（MKビット）＝1（割り込み要求マスク状態）：ベクタ割り込み処理，マクロ・サービス処理，レジスタ・バンク切り替え処理の保留
- 割り込みマスク・フラグ・レジスタ（MK0，MK1），または割り込み要求制御レジスタ（IC××）の割り込みマスク・フラグ（MKビット）＝0（割り込み要求マスク解除）
  - 割り込み要求制御レジスタ（IC××）のMS/$\overline{\text{INT}}$ビット＝0：ベクタ割り込み処理，レジスタ・バンク切り替え処理モード
    - 割り込み禁止状態（IE＝0）：ベクタ割り込み処理，レジスタ・バンク切り替え処理の禁止
    - 割り込み許可状態（IE＝1）
      - 割り込み要求制御レジスタ（IC××）のENCSビット＝0：ベクタ割り込み処理実行
      - 割り込み要求制御レジスタ（IC××）のENCSビット＝1：レジスタ・バンク切り替え処理実行
  - 割り込み要求制御レジスタ（IC××）のMS/$\overline{\text{INT}}$ビット＝1：マクロ・サービス処理実行

保守／廃止

5.3　優先順位制御

多重割り込みに関する優先順位制御はマスカブル割り込みに対して行われますが，各割り込み要因には，4レベルの優先順位を任意に設定することが可能です。

V55PIの内蔵割り込みコントローラは，この情報を基に優先順位制御を行います。

多重割り込み処理制御に対する基本的な制御の概念はV25，V35と同じです。

5.3.1　多重割り込みに関する優先順位制御

多重割り込み処理制御は，現在処理中の割り込みより優先順位レベルの高い割り込み要求があった場合，現在処理中の割り込みを中断して，優先順位の高い割り込み要求を受け付け処理を行う機能です。

現在処理中の割り込みの優先順位レベル以下の割り込み要求であった場合は，その割り込み要求は保留されます。

割り込みの多重処理制御は，割り込み許可状態（IE＝1）のときに行われます。したがって，割り込み処理ルーチンでも割り込み許可状態（IE＝1）にする必要があります。

多重割り込み処理制御のための優先順位は，各マスカブル割り込み要求ごとに0-3まで（0が最優先）の4レベルが，ソフトウエアにより任意に設定可能です。優先順位レベルの設定は，マスカブル割り込み要求ごとに用意されている割り込み要求制御レジスタ（IC××）のPR00-PR01ビットで行います。リセット時には，MKビットにより割り込み要求はマスクされ，PR00-PR01ビットにより優先順位はレベル3に設定されます。

マスカブル割り込みの優先順位

（高）　レベル0＞レベル1＞レベル2＞レベル3　（低）

割り込み受け付け後，マスカブル割り込み応答では，割り込み処理ルーチンの最後で処理が終了したことを割り込みコントローラに示すため，割り込み処理からの復帰命令（RETI命令またはRETRBI命令）の直前でFINT命令を実行する必要があります。FINT命令を実行しないと，以後正しく優先順位制御ができなくなります。

多重処理制御により中断された割り込み処理は，優先順位レベルの高い割り込み処理が終了し，FINT命令，RETI命令またはRETRBI命令が実行されたあと，再開します。

保留された割り込み要求は，実行中の割り込み処理が終了したあと，FINT命令，RETI命令またはRETRBI命令の実行後に受け付けられます。

注意1．NMIとINTWDTは多重処理制御を受けませんので，常時受け付けられます。
またNMI，INTWDTは，割り込み要求を受け付けたあと割り込み禁止状態（IE＝0）になりますが，割り込み処理ルーチン内で割り込み許可状態（IE＝1）にすれば，マスカブル割り込みの多重処理が実行されます。

保守／廃止

注意2.　マクロ・サービスでの応答を設定した割り込み要求については，上記優先順位制御の影響を受けず，ディフォールト優先順位のみ有効となります。ただし，マクロ・サービス完了割り込みについては優先順位制御を受けます。

マクロ・サービスの１回の処理は１つの命令と同様に実行されるため，途中でほかの割り込み（マクロ・サービスを含む）を受け付けることはありません（5.4.3(2) マクロ・サービス完了割り込み参照）。

5.3.2　同時に割り込みが発生したときの優先順位制御

複数の割り込み要求が同時に発生した場合の優先順位は次のようになります。

同時に発生した割り込み要求の優先順位

（高）NMI＞INTWDT＞マクロ・サービスで処理されるマスカブル割り込み

＞ベクタ割り込み，またはレジスタ・バンク切り替え機能で処理されるマスカブル割り込み（低）

また，ベクタ割り込みまたはレジスタ・バンク切り替え機能で処理される複数のマスカブル割り込みの間では，各割り込み要求制御レジスタ（TC××）のPR00-PR01ビットで指定される優先順位に従って，順次割り込み要求が受け付けられます。ある優先順位に設定された割り込みが複数ある場合，表５-１で示されるディフォールト優先順位に従って割り込み要求を受け付けます。

具体例については，**5.12　割り込み要求の処理動作例**も参照してください。

(1) マクロ・サービスとNMIの実行順序について

マクロ・サービスはEI，DI状態にかかわらず受け付けられるため，NMIあるいはINTWDTの割り込み処理ルーチン実行中でも実行されます。

NMI要求とマクロ・サービス要求が同時に発生した場合，割り込みコントローラにおける処理制御は，NMI要求が優先されて受け付けられます。ただし，NMI処理ルーチンへの分岐直後にマクロ・サービスが実行されるため，割り込み処理ルーチンを１つの単位として考えた場合にはプログラム上の実行順序は次のようになるので注意してください。

マクロ・サービス → NMI

NMI（INTWDT）の処理ルーチンでマクロ・サービスの実行を禁止するには，NMI処理ルーチンの１番目の命令で割り込みマスク・フラグ・レジスタを操作し，マクロ・サービス起動の割り込み要求をマスクすることが必要です。

(2) NMIとINTWDTの実行順序について

NMI要求とINTWDT要求はEI，DI状態にかかわらず受け付けられます。また，NMI要求とINTWDT

保守／廃止

要求が同時に発生した場合，割り込みコントローラにおける処理制御はディフォールトで次のように行われます。

NMI → INTWDT

ただし，NMI処理ルーチンへの分岐直後にINTWDT要求が受け付けられるため，割り込み処理ルーチンを１つの単位として考えた場合にはプログラム上の実行順序は次のようになるので注意してください。

INTWDT → NMI

NMIの処理ルーチンでINTWDTの実行を禁止するには，NMI処理ルーチンの１番目の命令でウォッチドッグ・タイマのカウンタ・クリア命令（RSTWDT）を実行してください。

NMI要求と同時にINTWDT要求が発生していてもウォッチドッグ・タイマ割り込みがキャンセルされ，割り込みは実行されません（逆にウォッチドッグ・タイマ割り込みの処理ルーチン内でNMIの発生を禁止することはできません）。

5.3.3　多重割り込みの制御モード選択

割り込みモード・コントロール・レジスタ（IMC）のビット７（PRSLビット）の設定により，最下位の優先順位レベル（レベル３）のマスカブル割り込みについて多重割り込み処理制御の動作モード選択を行うことができます（**5.5.1　割り込みモード・コントロール・レジスタ（IMC）**参照）。

PRSLビット＝1：

レベル３のマスカブル割り込みについても多重割り込み処理制御を行います。

レベル３の割り込み処理中の多重割り込み処理制御については，上位レベル（レベル２以上）の割り込み要求のみ受け付け可能で，同一レベル（レベル３）の割り込みは多重割り込み禁止となり，割り込みは保留されます。

割り込みの優先順位制御（割り込み要求制御レジスタPR00，PR01ビット）により，最下位レベル（レベル３）の多重割り込み処理制御を行う場合にこのモードを選択します。

PRSLビット＝0：

レベル３のマスカブル割り込みについて，多重処理制御を行いません。

このモードでは，レベル３のマスカブル割り込みを処理中に，割り込み許可状態（IE＝1）にすれば同一レベル（レベル３）の割り込み要求も受け付けることができます（現在処理中の割り込みが再度発生した場合は，同じ割り込みがもう一度受け付けられることになります）。

ただし，レベル２以上のマスカブル割り込みを処理中の場合は，レベル３の割り込み要求は保留されます。このモードでは上位レベルの割り込み要求が受け付け可能です。レベル２以上のマスカブル割り込みを処理中の場合，同一レベルの割り込みは禁止となり，割り込み要求は保留されます。

保守／廃止

このモードでは，インサービス・プライオリティ・レジスタ（ISPR）のビット３の値は不定です。

割り込みモード・コントロール・レジスタ（IMC）については，**5.5　制御レジスタ**を参照してください。

5.4　割り込み応答方式

V55PI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種類の割り込み応答方法を持っています。マスカブル割り込み要求では，これらの機能を割り込みの使用目的にあわせて，割り込み要因ごとの割り込み要求制御レジスタ（IC××）で選択することが可能です。内蔵割り込みコントローラは，設定された応答方式に応じて割り込みの要求に対応します。

特にV55PIとV25，V35との割り込み応答方式との比較を次に示します。

(1) ベクタ割り込み機能

- V25，V35と同様の機能
- EI状態でのみ受け付け
- DI状態では割り込み要求は保留
- 優先順位はディフォールト優先順位および優先順位指定フラグで指定
- より高い優先順位に指定した割り込みへのネスティングあり

(2) レジスタ・バンク切り替え機能

- レジスタ・バンクの拡張（V25：８バンク→V55PI：16バンク）
- V25，V35からレジスタ・バンク指定方法を変更
- 割り込みで切り替えるレジスタ・バンクは割り込み要求ごとのベクタ・テーブル（メモリ）のオフセット・アドレス（下位ビット０-３）上に設定
  （V25，V35は各要求の優先順位レベル＝レジスタ・バンク）
- ほかの機能についてはベクタ割り込み機能と同様

(3) マクロ・サービス機能

- マクロ・サービスの種類がV25，V35から大きく拡張
- 機能拡張に伴いマクロ・サービス制御ワードがレジスタ・ファイル空間（内蔵RAM）にマッピング（V25，V35では特殊機能レジスタ領域にマッピング）
- EI状態／DI状態のどちらでも受け付け可能(NMI，INTWDTの処理ルーチンからでも受け付け可能)
- 優先順位はディフォールト優先順位のみ有効
- ネスティングなし（命令でなくマイクロプログラムでの処理のため）

保守／廃止

5.4.1　ベクタ割り込み

ベクタ割り込みでは割り込み許可状態(EI状態)でのみ割り込み要求を受け付けます。ベクタ割り込み要求を受け付けるとCPUは割り込み禁止状態（DI状態）になり，現在のPSWの内容とPCとPSの内容をスタックに退避させます。その後，ベクタ・テーブルから対応するベクタを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を開始します。ベクタ番号は割り込み要因ごとに固定されています。また，DI状態では，割り込み要求は保留され，EI状態になった時点で受け付けられます。

割り込みからの復帰はRETI命令で行いますが，ノンマスカブル割り込みを除くハードウエア割り込みの場合は，FINT命令を復帰命令の前に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。

図５－２　割り込み受け付け動作（① → ④の順に行われる）

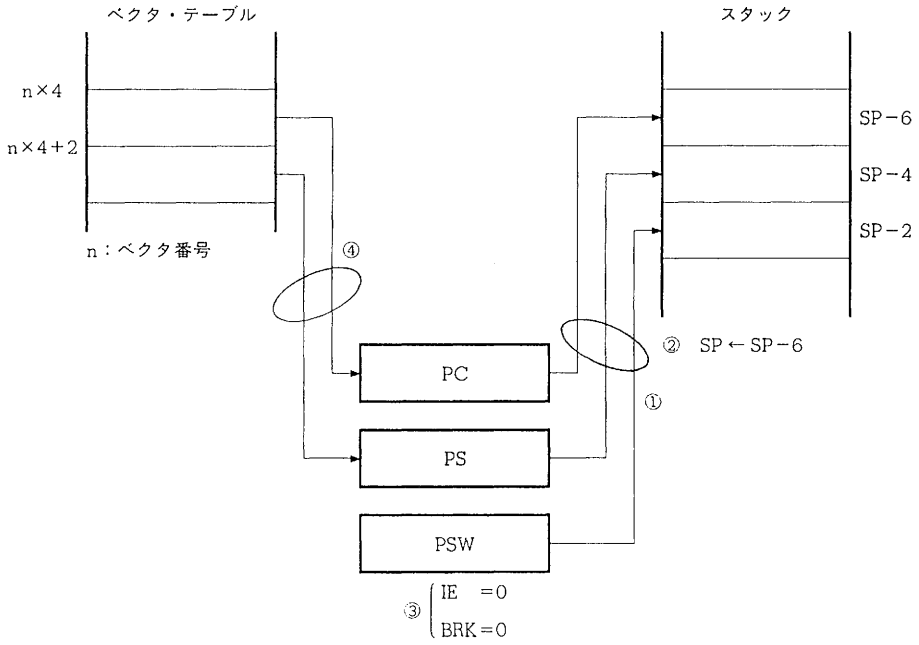
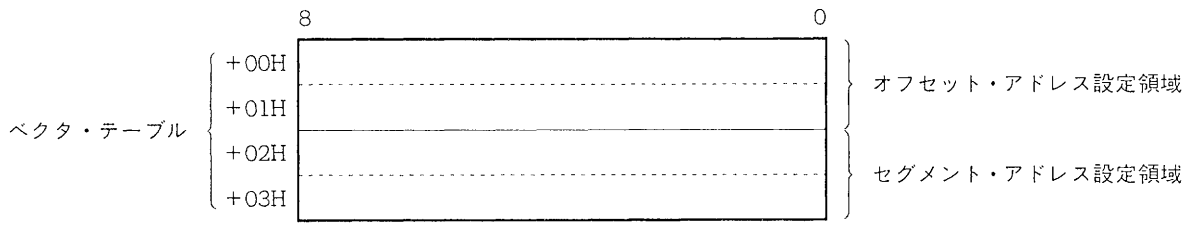


図５－３　ベクタ・テーブルの構成



保守／廃止

5.4.2　レジスタ・バンク切り替え機能

V55PIは，内蔵RAM領域に汎用のレジスタ・セットをマッピングしており最大16バンクまでのレジスタ・セットを持つことができます。このレジスタ・バンクを割り込みの応答時に自動的に切り替えることで，割り込み処理を行います。従来ソフトウエ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すみますので，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。

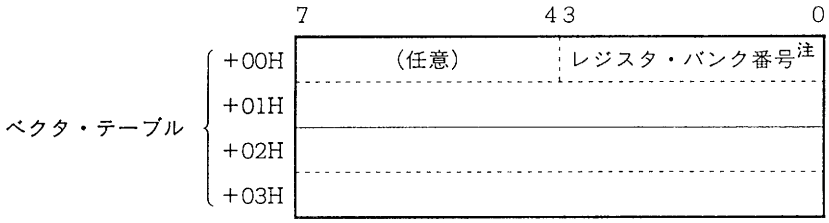
また，BRKCS，TSKSW命令の実行により，レジスタ・バンク切り替えを行うこともできます。

なお，レジスタ・バンクはベクタPCとDS3の領域を兼用しているため，割り込みでレジスタ・バンクを使用し，かつ割り込み処理中にDS3を使用する場合，ベクタPCをスタックへ退避する必要があります。

(1) 割り込みによるレジスタ・バンク切り替え機能

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，各割り込み要求に対応する割り込み要求制御レジスタ内のビット４（ENCSビット）を“1”に設定し，レジスタ・バンク切り替えの起動を許可にします。割り込みで切り替わるレジスタ・バンク番号は，割り込み要因ごとのベクタ・テーブルの先頭アドレスの下位４ビット上に設定します。割り込み許可状態(IE＝1)，かつ割り込みマスク解除の状態で指定した割り込み要求が発生すると，レジスタ・バンクが切り替わります。

図５－４　ベクタ・テーブルとレジスタ・バンク番号



注　割り込みベクタ・アドレスの下位４ビットが使用されるため，同一の割り込み要求に対し，ベクタ割り込みとレジスタ・バンク切り替えを併用する場合には，切り替え先のレジスタ・バンク番号を次のようにしてください。

（割り込み処理ルーチンの先頭アドレスの下位４ビット）＝（切り替え先のレジスタ・バンク番号）

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５－５参照）。

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

保守／廃止

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の実行を開始します。

このシーケンスからも分かるように，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および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。ただし，割り込み処理ルーチンの先頭でMOVSPA命令を使用する場合はSS, SPのイニシャライズは不要です。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。PSを変更してそのまま割り込み処理を終了すると，次にその割り込みが受け付けられたときに，変更したPSの値となってしまい，期待した番地に処理が移行しなくなります。

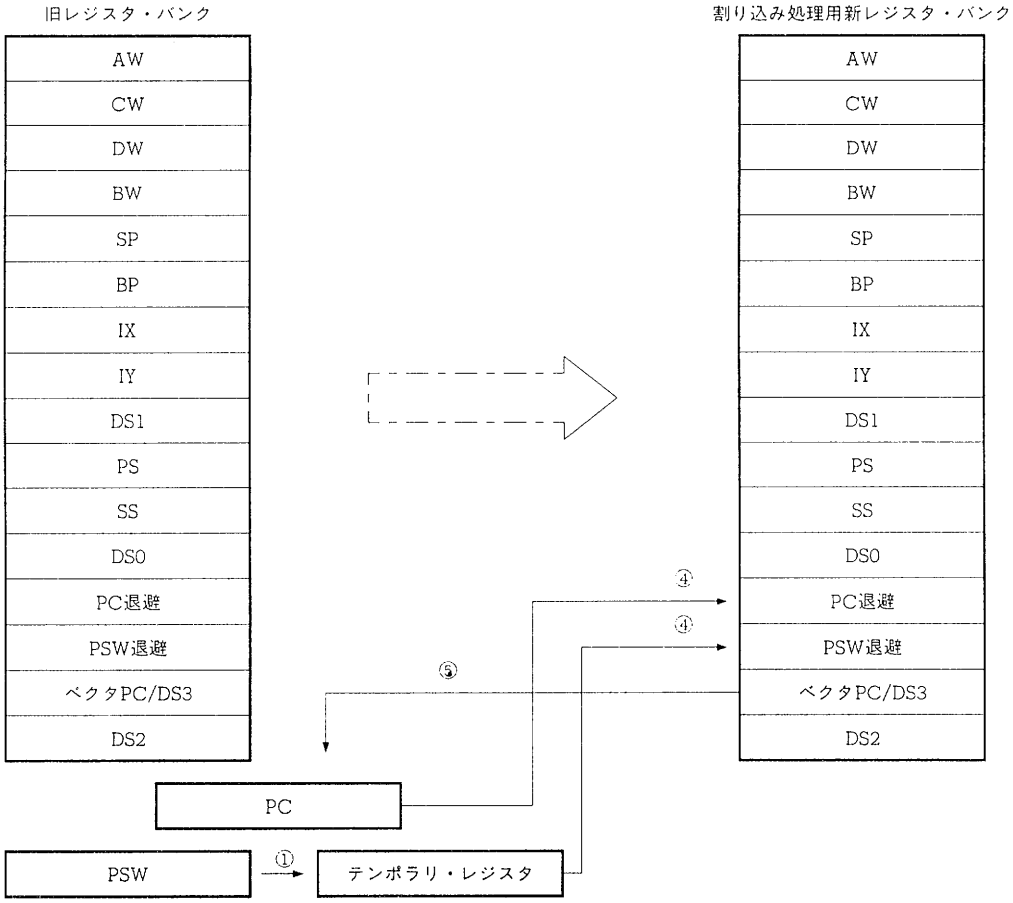
割り込みによるレジスタ・バンク切り替えからの復帰は，切り替えの要因となった割り込みが多重割り込み処理制御を受けるものであるため，FINT命令実行後，RETRBI命令を実行することにより行われます。このとき，FINT命令はRETRBI命令の直前で実行してください。

RETRBI命令を実行すると図５－６に示すようにレジスタ・バンク中の退避領域からPCとPSWをそれぞれ復帰させます。RETI命令では，レジスタ・バンクの復帰を行いませんので，正常にメイン・ルーチンへ復帰できません。

保守／廃止

図５－５　レジスタ・バンク切り替えのシーケンス

（割り込みによるレジスタ・バンク切り替え時）



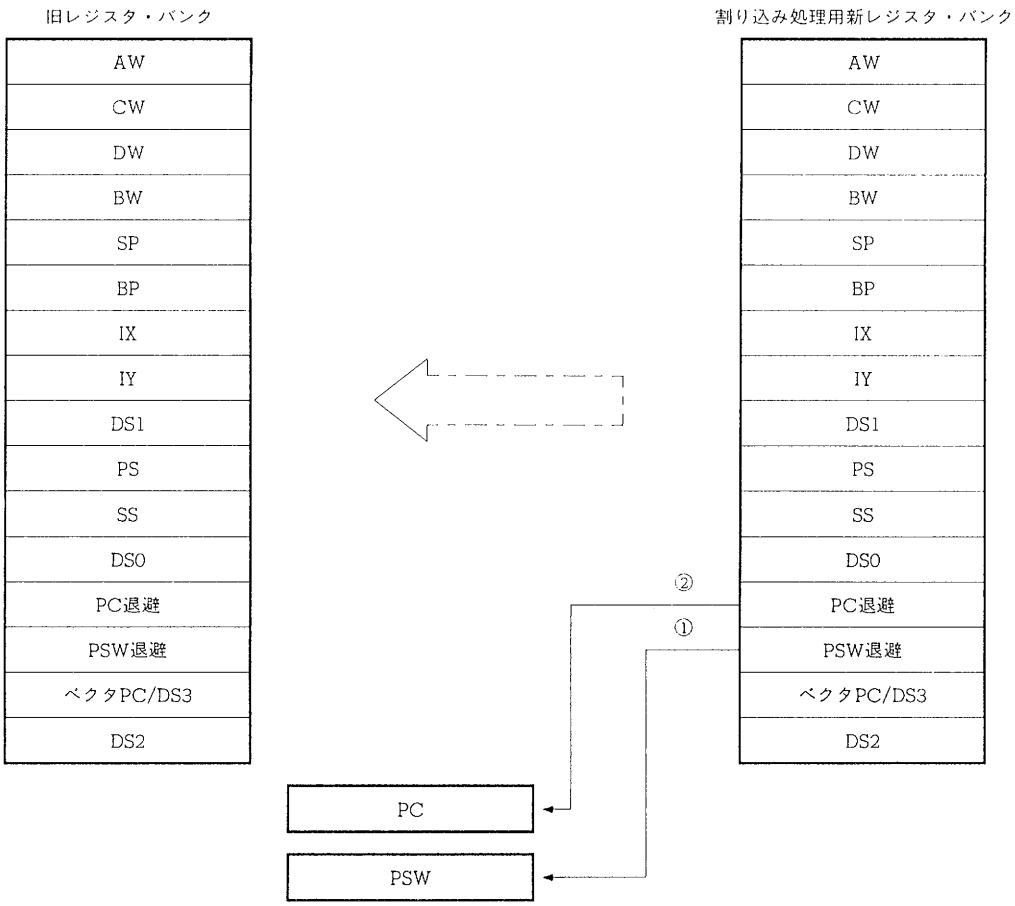
② レジスタ・バンク切り替え

③ IE＝0，BRK＝0

保守／廃止

図５－６　レジスタ・バンク復帰シーケンス

RETRBI命令の実行により，次のシーケンスでレジスタ復帰処理を行います。



(2) ソフトウエアによるレジスタ・バンク切り替え機能

(a) BRKCS命令による切り替え（BRKCS reg16）

BRKCS命令の実行により，割り込みによるレジスタ・バンク切り替えと同じシーケンスでレジスタ・バンク切り替えができます。したがって，高速なサブルーチン・コールに使用できます。

オペランドに記述される16ビット・レジスタの下位４ビットでレジスタ・バンクを選択します。

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をセットすることができますので，切り替え前後でスタックを連続させて使用することができます。

保守／廃止

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) TSKSW命令による切り替え（TSKSW reg16）

TSKSW命令の実行によっても，レジスタ・バンク切り替えができます。マルチタスク制御を行う場合，各レジスタ・バンクにそれぞれのタスクのプログラムの実行情報（コンテキスト）を割り当てて，高速にタスク切り替えを行うことができます。

オペランドに記述される16ビット・レジスタの下位４ビットでレジスタ・バンクを選択します。

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。

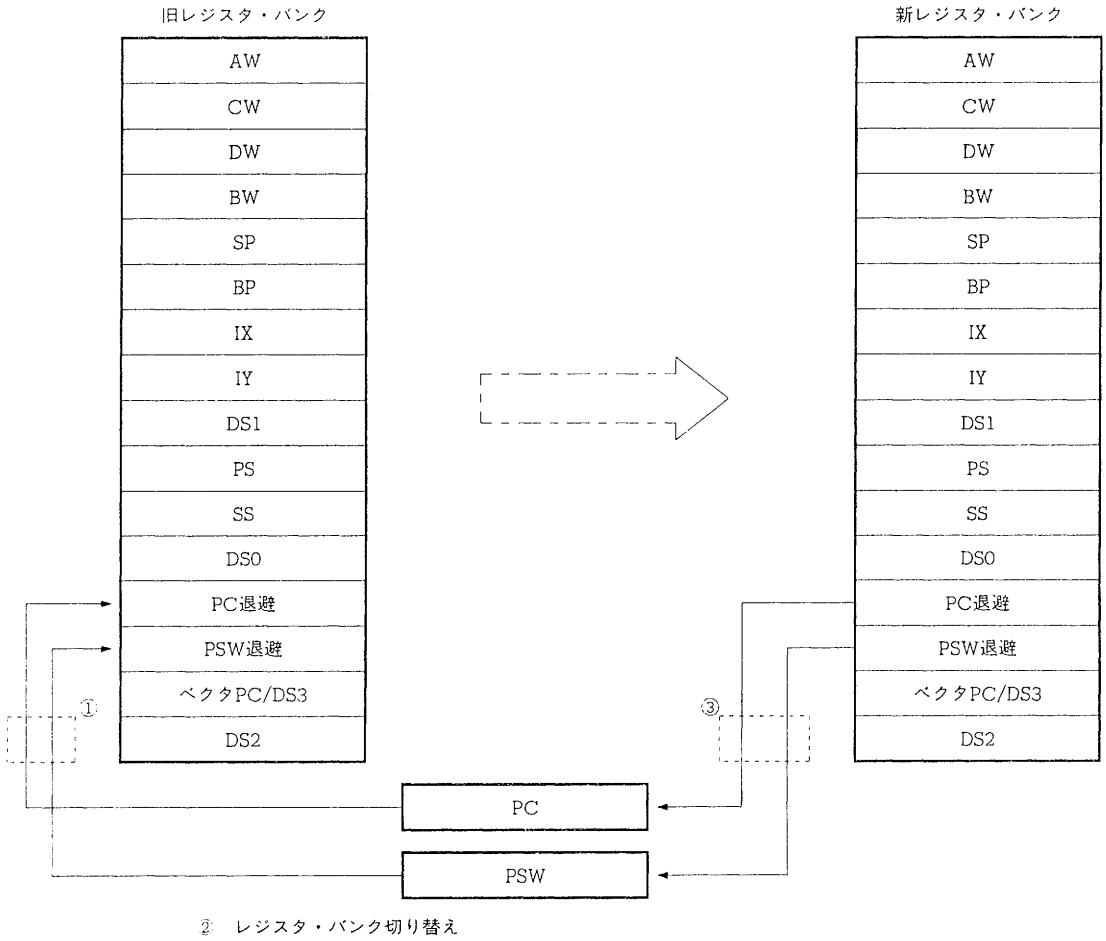
① PC，PSWを現在（切り替え前）のレジスタ・バンク内のPC退避エリア，PSW退避エリアに退避します。
② レジスタ・バンクを切り替えます。
③ 新しいレジスタ・バンク内のPSW退避エリアの値をPSWにロードし，PC退避エリアの値をPCにロードし分岐します。

TSKSW命令を実行する場合は，最初の切り替え前に選択するレジスタ・バンク内のPSとPC退避エリア，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。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はそれぞれ切り替え前の値をセットすることができます。

レジスタ・バンク切り替えのシーケンスは図５－７のように行われます。

保守／廃止

図5－7　TSKSW命令によるレジスタ・バンク切り替えのシーケンス



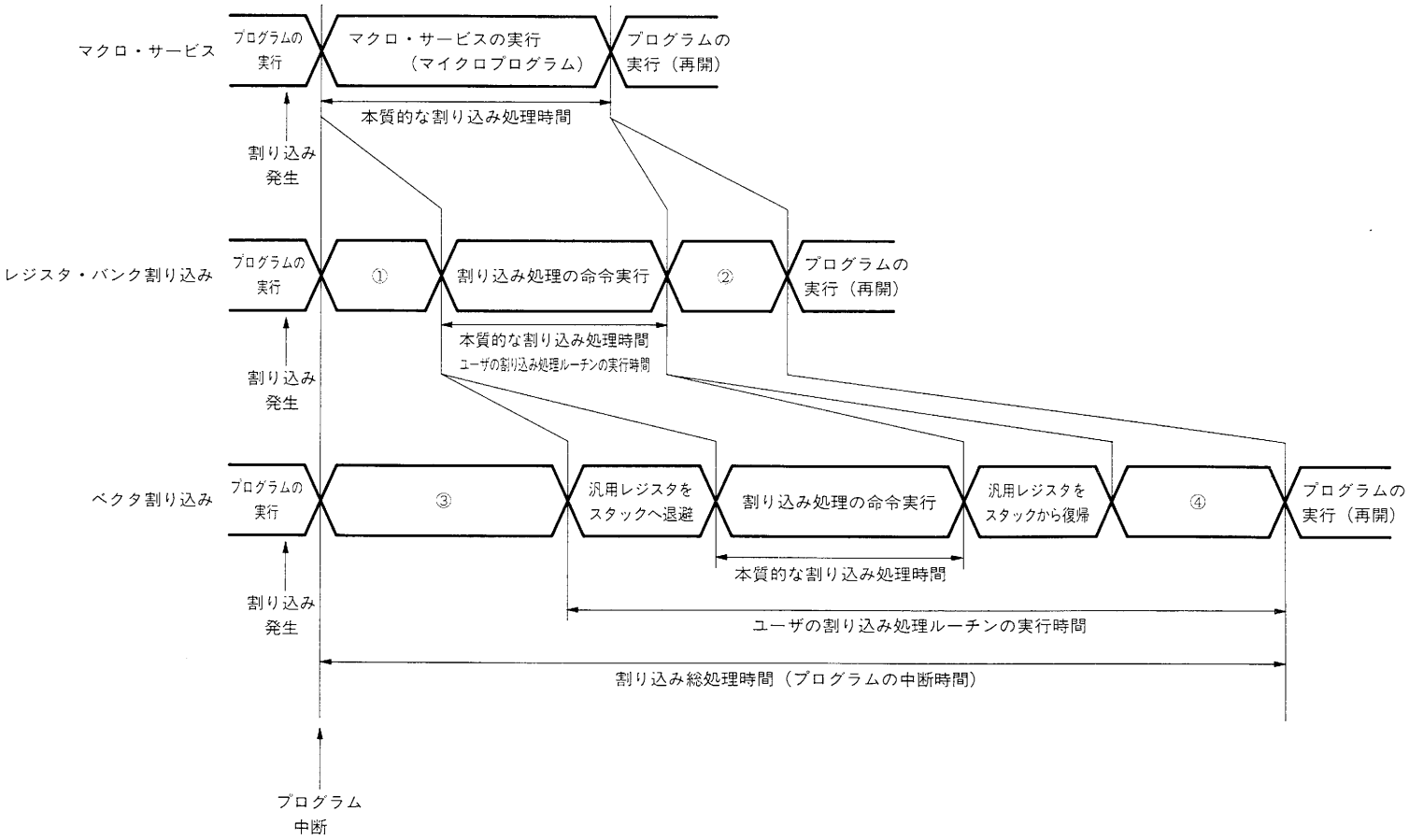
5.4.3　マクロ・サービス機能

マクロ・サービス機能は単純なデータ転送などを，割り込み要求で起動されるマイクロプログラム（CPU内部の専用ファームウエア）で処理する機能です。従来，ユーザ・プログラムでコーディング，実行していた単純かつ定型的な割り込み処理を自動処理します。

マクロ・サービス処理は，割り込み要求に起因し処理されます。ソフトウエア処理が主体となるユーザ・プログラムの割り込み処理ルーチンへの分岐の発生頻度をできるだけ少なくし，レジスタ退避，初期化，レジスタ復帰，割り込みルーチンからの復帰といった一連の処理によるソフトウエアのオーバヘッドを抑え，CPUの使用効率を向上させることを目的としています（図5－8参照）。

保守／廃止

図５－８　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。

保守／廃止

また，マクロ・サービスで行われる処理はソフトウエア的には意識する必要がなく，従来，バイト単位でソフトウエアによって処理していたデータを一固まりのデータとして処理することができ，効率よくプログラミングすることが可能です。

V55PIのマクロ・サービスでは，V25，V35におけるような単なるデータ転送だけでなく，データ比較，データの差分計算など，V55PIの内蔵周辺ハードウエアに密着した数々の動作モードをサポートしています。

(a) EVTCNT (EVENT COUNTER)

マクロ・サービスの発生ごとにカウンタを更新し，カウンタが0になると対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

(b) DTACMP (DATA COMPARE)

割り込み要因固有のSFRとあらかじめ設定したバイト・データと比較し，一致すれば対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

(c) DTADIF (DATA DIFFERENCE)

タイマ／カウンタ・ユニットのキャプチャ・レジスタを用いた差分の計算をします。タイマ割り込みによって起動され，今回ラッチしたキャプチャ・レジスタの値から前回ラッチしたキャプチャ・レジスタの値を減算し，あらかじめ指定したメモリ・バッファに結果をストアします。

あらかじめ設定した回数の処理を行うと，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

(d) BLKTRS (BLOCK TRANSFER)

あらかじめ指定したメモリ・バッファとSFR間のデータ転送を行います。

あらかじめ設定した回数のデータ転送を行うと，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

(e) BLKTRS-C (BLOCK TRANSFER WITH CHARACTER SEARCH)

あらかじめ指定されたメモリ・バッファとSFR間のデータ転送を行います。あらかじめ設定した回数のデータ転送を終了するか，または転送データがあらかじめ設定したキャラクタ・データと一致した場合は，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

(f) RTOPTRN (RTOP TRANSFER)

リアルタイム出力ポートに出力するデータをポート7バッファ(P7H，P7L)に転送するとともに，リアルタイム出力ポートに出力するインターバルを指定するデータをタイマ・コンペア・

保守／廃止

レジスタ（CM00，CM01）に転送します。

(g) DTACMP-M (DATA COMPARE WITH CHARACTER MASK)

外部I/Oから読み込んだステータス・データと，あらかじめ設定したマスク・データとの論理積をとります。その結果と，あらかじめ設定したバイト・データを比較します。一致すれば外部I/Oとメモリ間のデータ転送を行います。一致しない場合，またはあらかじめ設定した回数のデータ転送を行うと，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

(1) マクロ・サービス処理

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によって制御されます。

割り込み要求制御レジスタ（IC××）内のMS/$\overline{\text{INT}}$ビットがマクロ・サービス指定状態（＝1）のときに起動され，割り込み要求が発生したら，プログラム・ステータス情報（PC，PSW）を保持したまま内蔵RAM（レジスタ・ファイル）内のマクロ・サービス・モード・レジスタで指定される処理を起動し，その処理完了後に続く命令の実行またはほかの割り込み処理を実行します。マクロ・サービスは，割り込み許可（IE＝1），割り込み禁止（IE＝0）状態にかかわらず処理されます。

(2) マクロ・サービス完了割り込み

マクロ・サービス・カウンタ（MSC）＝0またはマクロ・サービスが終了する条件が成立すると，対応する割り込み要求制御レジスタのIFビットはセットされたままでMS/$\overline{\text{INT}}$ビットがクリアされ，ベクタ割り込みまたはレジスタ・バンク切り替え（マクロ・サービス完了割り込み）が発生します。割り込み要求が受け付けられると，通常の割り込みと同様にIFビットはクリアされます。

続けてマクロ・サービスを行う場合は，この割り込み処理ルーチン内で必要パラメータの再設定をし，MS/$\overline{\text{INT}}$ビットをセットしてください。

マクロ・サービス完了割り込みは，優先順位と割り込み禁止（DI）状態によって制御されます。したがって，マクロ・サービス完了割り込み保留中に再度同じ割り込みが発生すると，2回目に発生した割り込みは無視されます（UART送信割り込み，UART受信割り込みは除く。**7.4.8(6)送受信割り込みについて**参照）。

マクロ・サービス完了割り込み処理中に発生した同一割り込み要求は，次のように受け付けられます。

- 割り込み要求発生後，MS/$\overline{\text{INT}}$ビットをセットした場合
  IFビット＝1となり，割り込み要求は保留されます。MS/$\overline{\text{INT}}$ビットをセットするとマクロ・サービスを実行します。
- MS/$\overline{\text{INT}}$ビットをセットしたあとに，割り込み要求が発生した場合
  割り込み要求が発生すると，マクロ・サービスを実行します。

保守／廃止

●MS/$\overline{\text{INT}}$ビット＝0のまま割り込み処理を終了する場合

IFビット＝1となり，実行している割り込み処理が終了するまで保留されます。

割り込み処理終了後，保留されていた割り込みが受け付けられます。

(3) マクロ・サービス制御レジスタ

マクロ・サービス動作の指定やその他の制御には，割り込み要因ごとに指定するマクロ・サービス・コントロール・ワードと，動作モードごとに指定するマクロ・サービス・チャネルを使用します。

(a) マクロ・サービス・コントロール・ワード

マクロ・サービス・コントロール・ワードは内蔵RAM領域（レジスタ・ファイル空間）内の下位64バイト（000H-03FH）に，レジスタ・バンク0，1と重なるように配置されています。

V55PIは25要因のマスカブル割り込み要求のうち，表5-2に示す22要因に対してマクロ・サービスをサポートしていますが，各割り込み要求に対して使用されるマクロ・サービス・コントロール・ワードのアドレスが固定されています。

マクロ・サービス・コントロール・ワードは，マクロ・サービス機能を制御するマクロ・サービス・モード・レジスタとマクロ・サービス・チャネル・ポインタで構成されています。

マクロ・サービス・コントロール・ワードの基本構成を図5-9に示します。

図5-9　マクロ・サービス・コントロール・ワードの基本構成

アドレス	
0××H	マクロ・サービス・モード・レジスタ
(0××+1) H	マクロ・サービス・チャネル・ポインタ

マクロ・サービス・モード・レジスタによって実行されるマクロ・サービス動作モードを設定し，マクロ・サービス・チャネル・ポインタによって，動作モードごとに使用されるマクロ・サービス・チャネルのアドレスを指定します。

マクロ・サービス処理を行うときは，前もってマクロ・サービス処理を指定できる割り込み要求に対応するマクロ・サービス・モード・レジスタとチャネル・ポインタを初期設定しておく必要があります。

マクロ・サービス機能を利用しない割り込み要因がある場合は，マクロ・サービス・コントロール・ワードは通常の内蔵RAMとして任意に利用可能です。

保守／廃止

表5－2　割り込み要求とマクロ・サービス・コントロール・ワード・アドレス

コントロール・ワード・アドレス	割り込み要求	割り込み発生要因
008H	INTPAI	パラレル・インタフェース
00AH	INTAD	A/Dコンバータ
012H	INTP0	INTP0端子入力
014H	INTP1	INTP1端子入力
016H	INTP2	INTP2端子入力
018H	INTP3	INTP3端子入力
01AH	INTP4	INTP4端子入力
01CH	INTP5	INTP5端子入力
020H	INTCM00	CM00の一致検出
022H	INTCM01	CM01の一致検出
024H	INTCM10	CM10の一致検出
026H	INTCM11	CM11の一致検出
028H	INTCM21	CM21の一致検出
02AH	INTCM31	CM31の一致検出
02CH	INTD0	DMAチャネル0メイン
02EH	INTD0S	DMAチャネル0サブ
030H	INTD1	DMAチャネル1メイン
032H	INTD1S	DMAチャネル1サブ
038H	INTSR0/ INTCSI0	UARTチャネル0受信／ シリアル・チャネル0送受信
03AH	INTSR1/ INTCSI1	UARTチャネル1受信／ シリアル・チャネル1送受信
03CH	INTST0	UARTチャネル0送信
03EH	INTST1	UARTチャネル1送信

注意　000H-007Hはマクロ・サービス（RTOPTRN）の実行時にワーク・エリアとして使用されます。

(b) マクロ・サービス・モード・レジスタ

マクロ・サービス・モード・レジスタはマクロ・サービスの動作を指定する8ビット・レジスタです。マクロ・サービス・コントロール・ワードの一部としてレジスタ・ファイル空間にマッピングされています（図5－9参照）。

(c) マクロ・サービス・チャネル・ポインタ

マクロ・サービス・チャネル・ポインタは，レジスタ・ファイル空間の下位256バイト内のマ

保守／廃止

クロ・サービス・チャネルの先頭アドレスを指定します。

マクロ・サービス・コントロール・ワードの一部としてレジスタ・ファイル空間にマッピングされています。

(d) マクロ・サービス・チャネル

マクロ・サービス・チャネルはマクロ・サービス・カウンタ，バッファなどを含む構造体でマクロ・サービス・モードごとに異なった構造をしています。マクロ・サービス・チャネルはレジスタ・ファイル空間の下位256バイト（000H-0FFH）に配置する必要があります。マクロ・サービス・チャネルは必ず偶数アドレスから始まるように指定してください。

保守／廃止

(4) マクロ・サービス動作モード

マクロ・サービス・モード・レジスタの設定により，マクロ・サービスの動作を指定します。

表５－３　マクロ・サービス動作モードの分類

マクロ・サービス・モード・レジスタへの設定値	機　　能	チャネル指定
０００００　００Ｃ０	EVTCNT (EVENT COUNTER)	なし
０００００　１０００	DTACMP (DATA COMPARE)	あり
０００１　００Ｗ０	DTADIF (DATA DIFFERENCE)	
０１０Ｍ　Ｄ０Ｗ０	BLKTRS (BLOCK TRANSFER)	
０１１Ｍ　Ｄ０Ｗ０	BLKTRS-C (BLKTRS WITH CHARACTER SEARCH)	
１０００　０ＵＡ０	RTOPTRN (REAL-TIME OUTPUT PORT TRANSFER)	
００１Ｍ　００ｄ０	DTACMP-M (DATA COMPARE WITH CHARACTER MASK)	

注意　上記以外のコードを設定した場合，マクロ・サービス機能および以後の動作を保証できませんので注意してください。

マクロ・サービス動作モードの設定値において，次のようなサブコマンドを指定して動作を決定します。

C：マクロ・サービス・カウンタ（MSC）のインクリメント（＋1）／デクリメント（－1）指定

C	動　　作
0	インクリメント
1	デクリメント

W：データのバイト／ワード指定

W	バイト／ワード
0	バイト
1	ワード

保守／廃止

D, d：データの転送方向指定

D	転送方向
0	メモリ ← SFR
1	SFR ← メモリ

d	転送方向
0	メモリ←I/O
1	I/O←メモリ

M：メモリ空間のアドレス・モード指定

M	アドレス・モード
0	1Mバイトの基本メモリ空間のアクセス・モード
1	16Mバイトの拡張メモリ空間のアクセス・モード

U：コンペア・レジスタ設定データ・テーブルのアドレス更新設定

U	コンペア・レジスタ設定データ参照アドレス
0	MEM. PTR（セグメント：MTS, オフセット：MTP）を更新しない
1	MEM. PTR-MSC×2+2（セグメント：MTS, オフセット：MTP）でアドレス更新する

A：コンペア・レジスタ設定データへの操作指定

A	コンペア・レジスタ設定データ
0	テーブル上の設定データをそのままコンペア・レジスタへ転送
1	テーブル上の設定データ＋現在のコンペア・レジスタ設定データを転送

1Mバイトの基本メモリ空間のアクセス・モード

おのおの4ビット

MSS　○ ○ ○ ○ 0 0　H…セグメント値

+）MSP　　△ △ △ △　H…オフセット値

×　×　×　×　×　H…物理アドレス

16Mバイトの拡張メモリ空間のアクセス・モード

おのおの4ビット

MSS　○ ○ ○ ○ 0 0　H…拡張セグメント値

+）MSP　　△ △ △ △　H…オフセット値

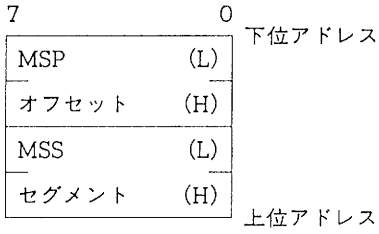
×　×　×　×　×　×　H…物理アドレス

保守／廃止

(5) アドレシング方法

マクロ・サービス・チャネルにおけるデータ・アドレシング方法を次に示します。

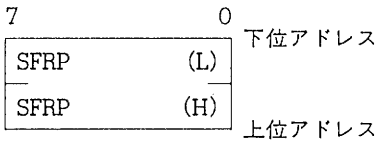
(a) 外部メモリ・アクセス（＝MEM1.PTR）



SFRをアクセス対象とすることも可能です。

(b) SFRアクセス（＝SFRP）

SFRアドレス指定ポインタは２バイト構成です。転送対象となる特殊機能レジスタ（SFR）のアドレスの下位16ビットを設定してください。上位７ビットにはすべて１を設定してください。



例　RxB0（UART受信バッファ：0FFF76H）を指定する場合

SFRP(L)……76H

SFRP(H)……0FFH

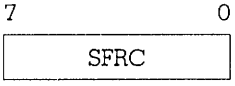
PAB（パラレル・インタフェース・バッファ：0FFE10H）を指定する場合

SFRP……FE10H

(c) タイマ・コンペア・レジスタ・アクセス（＝SFRC）

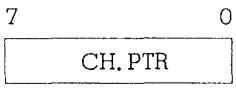
RTOPTRNモードで使用します。

コンペア・レジスタのアドレスは，下位８ビットのみ指定しCM0またはCM1を選択します。



保守／廃止

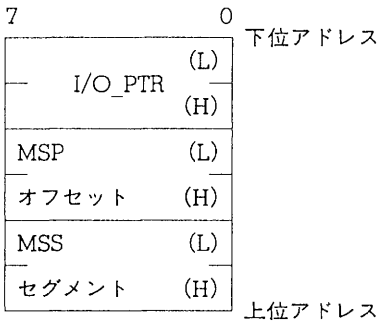
(d) 内蔵RAMアクセス



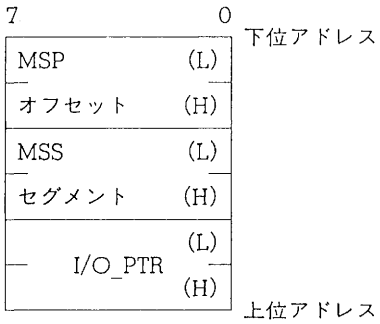
レジスタ・ファイル空間のアドレスは下位256バイト（000H-0FFH）に限定されます。

(e) 外部I/Oアクセス（=I/O_PTR）

メモリ←I/O転送の場合



I/O←メモリ転送の場合



保守／廃止

(6) 各マクロ・サービスの動作

(a) EVTCNT (EVENT COUNTER)

あらかじめ設定した回数のマクロ・サービス発生後，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

〔マクロ・サービス・コントロール・ワード〕

内部レジスタ・ファイル

7	0 オフセット・アドレス
000000C0	00H
MSC	01H

〔動　　作〕

① 割り込み要求で，マクロ・サービスが発生するごとに，マクロ・サービス・カウンタ(MSC)をインクリメント（+1）またはデクリメント（−1）します。

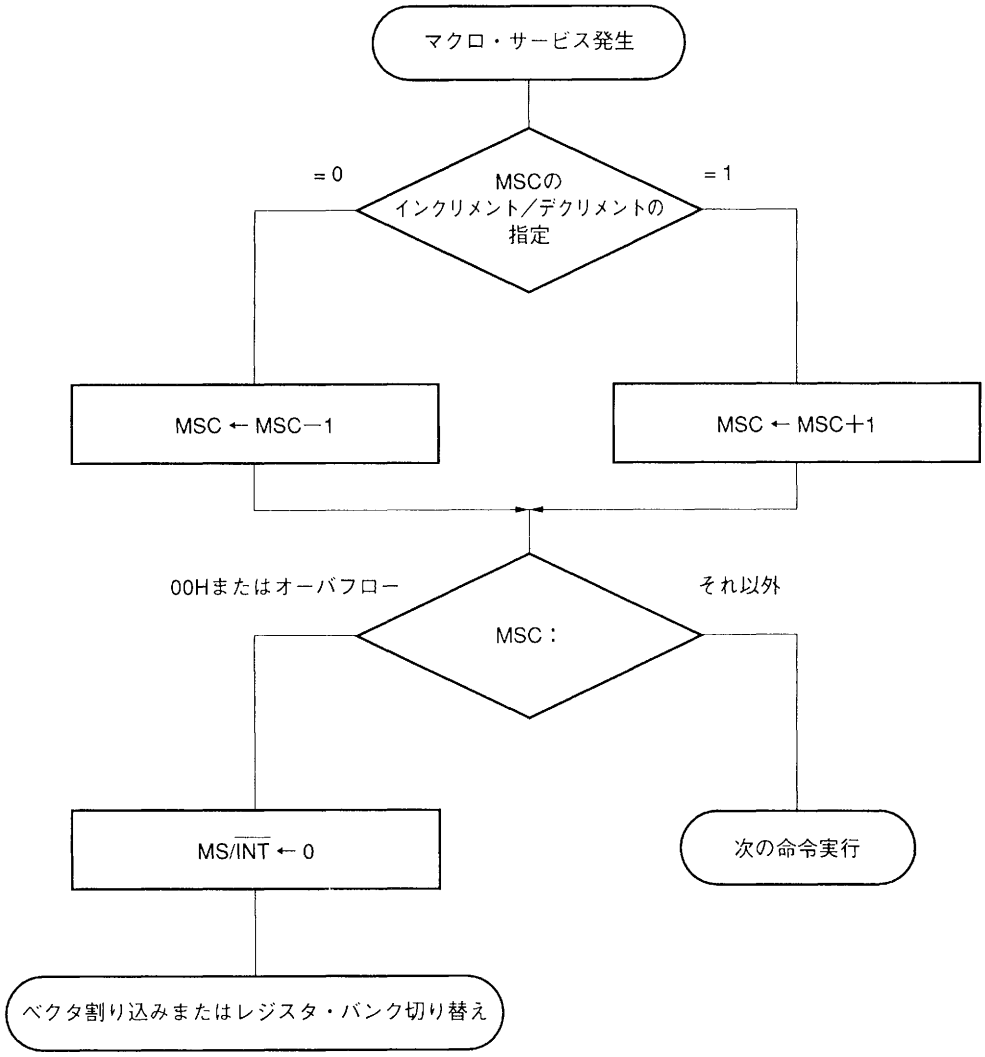
② MSCが00Hまたはオーバフローになると，対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$を0にし，ベクタ割り込み要求またはレジスタ・バンク切り替え要求を発生します。デクリメントのとき，MSCの初期値を0に設定すると256回の処理を行います。

MSCをインクリメントするかデクリメントするかはマクロ・サービス・モード・レジスタでのサブコマンドCの指定に従います。

表5－4　EVENTモードの動作指定

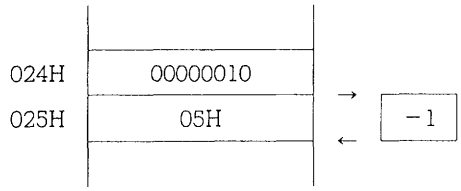
C	動　　作
0	インクリメント
1	デクリメント

保守／廃止

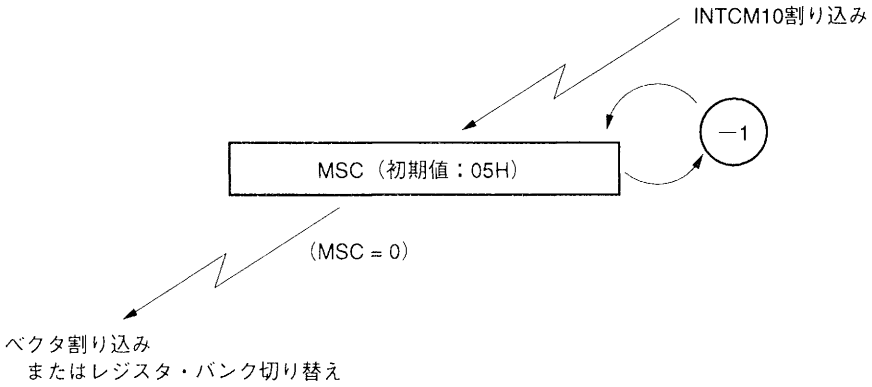


このモードでは，マクロ・サービス機能は割り込みの発生回数を分周するカウンタとして機能します。

例　INTCM10割り込み要求の発生回数をマクロ・サービスで５分周します。



保守／廃止



〔用　　途〕

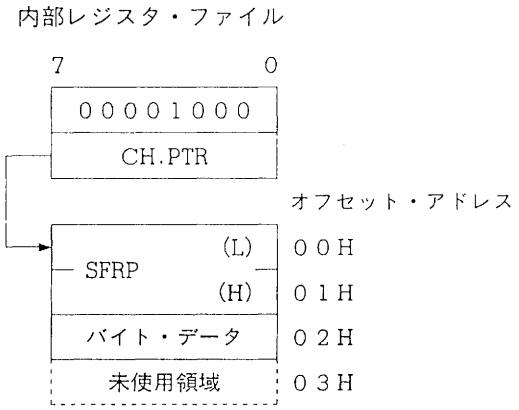
イベント・カウンタ，キャプチャ回数の測定など

保守／廃止

(b) DTACMP (DATA COMPARE)

割り込み要因固有のSFRの内容とあらかじめ設定したバイト・データとを比較し，一致すれば対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

〔マクロ・サービス・コントロール・ワード〕



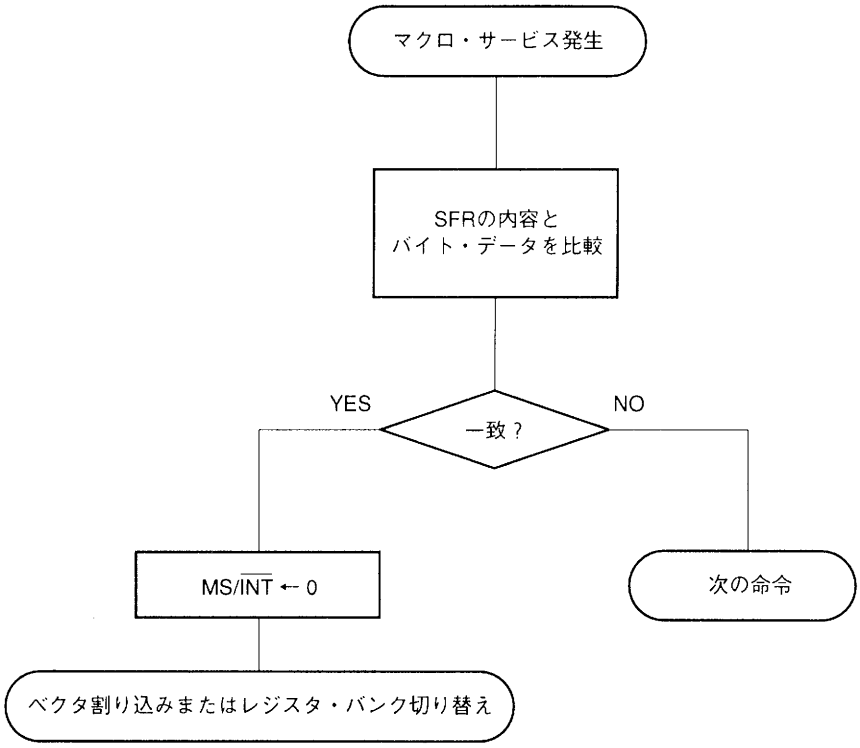
〔動　　作〕

① チャネル・ポインタ（CH.PTR）でマクロ・サービス・チャネルの先頭のアドレスを指定します。

② マクロ・サービスが発生すると，SFRポインタ(SFRP)で指定されるSFRの内容とバイト・データを比較して，一致する場合は，対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$を０にし，ベクタ割り込み要求またはレジスタ・バンク切り替え要求を発生します。一致しない場合は，そのままマクロ・サービスから抜け，処理を継続します。

注意　チャネル開始アドレスは偶数バイトからのため，未使用領域の位置から続けてほかのマクロ・サービス・チャネルを記述することはできません。

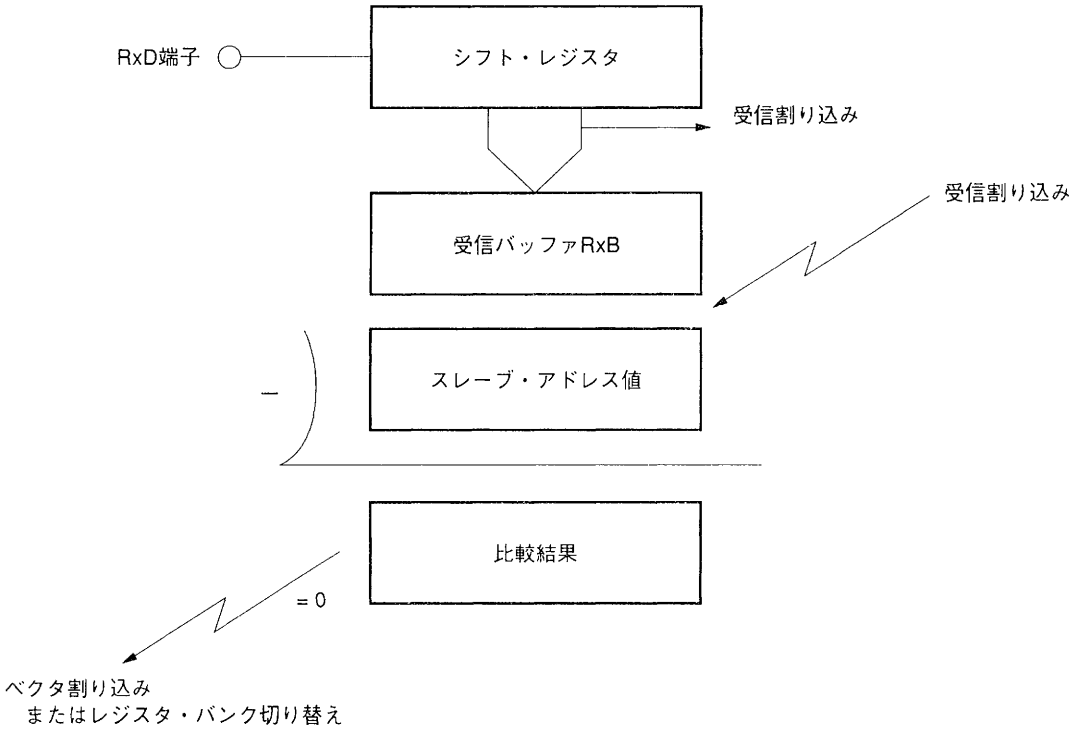
保守／廃止

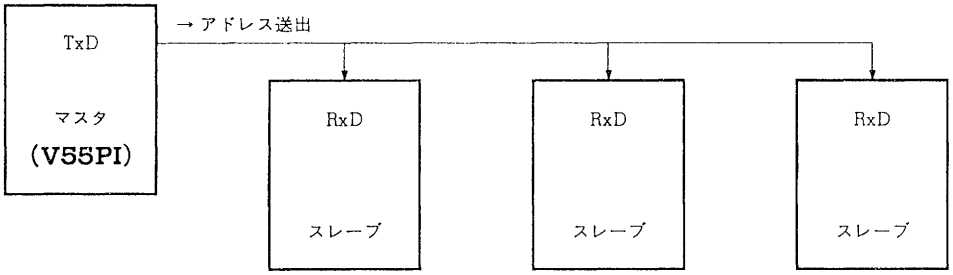


例　シリアル通信機能を用い，アドレス検出を行うシステム上でのアドレスの一致検出に使用できます。

特に，HALTモード中では，スレーブ・アドレスの受信でマクロ・サービスが起動され，受信バッファ内のスレーブ・アドレスとバイト・データが一致した場合は，ベクタ割り込み要求またはレジスタ・バンク切り替え要求が発生します。一致しない場合は，再度HALTモードになります。

保守／廃止





〔用　　途〕

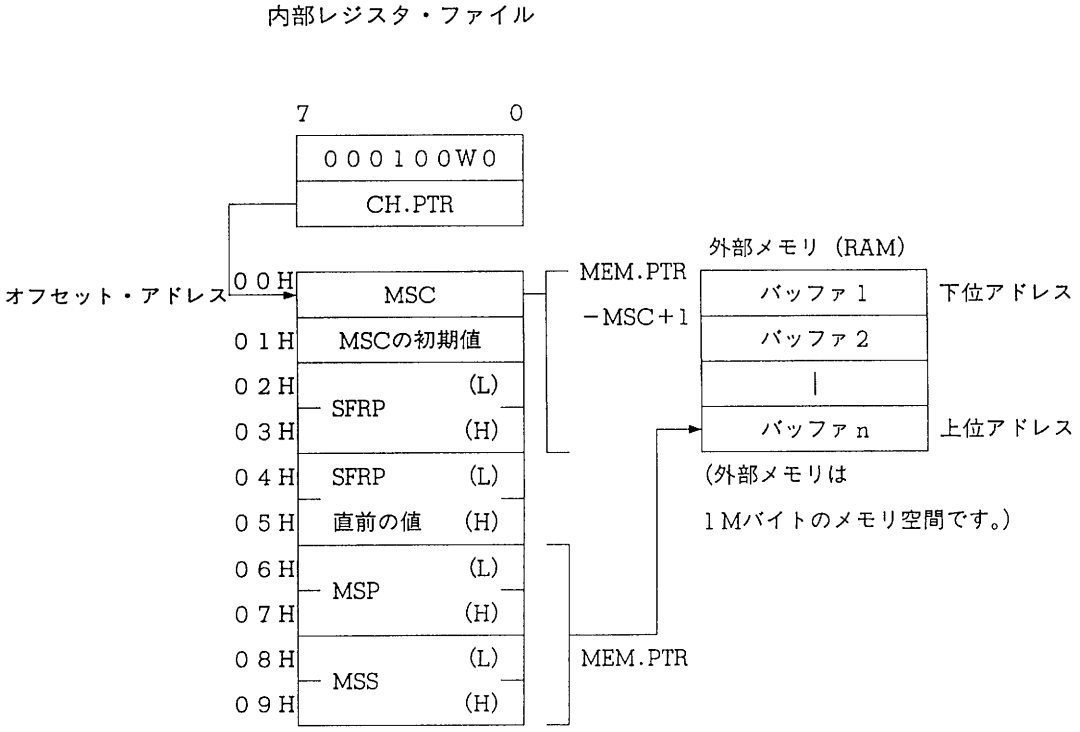
シリアル通信におけるスレーブ・アドレス検出，ポート入力状態のポーリングなど

保守／廃止

(c) DTADIF (DATA DIFFERENCE)

タイマ／カウンタ・ユニットのキャプチャ・レジスタを用いた差分の計算などに使用します。

〔マクロ・サービス・コントロール・ワード〕



〔動　作〕

① チャネル・ポインタ(CH.PTR)でマクロ・サービス・チャネルの先頭アドレスを指定します。

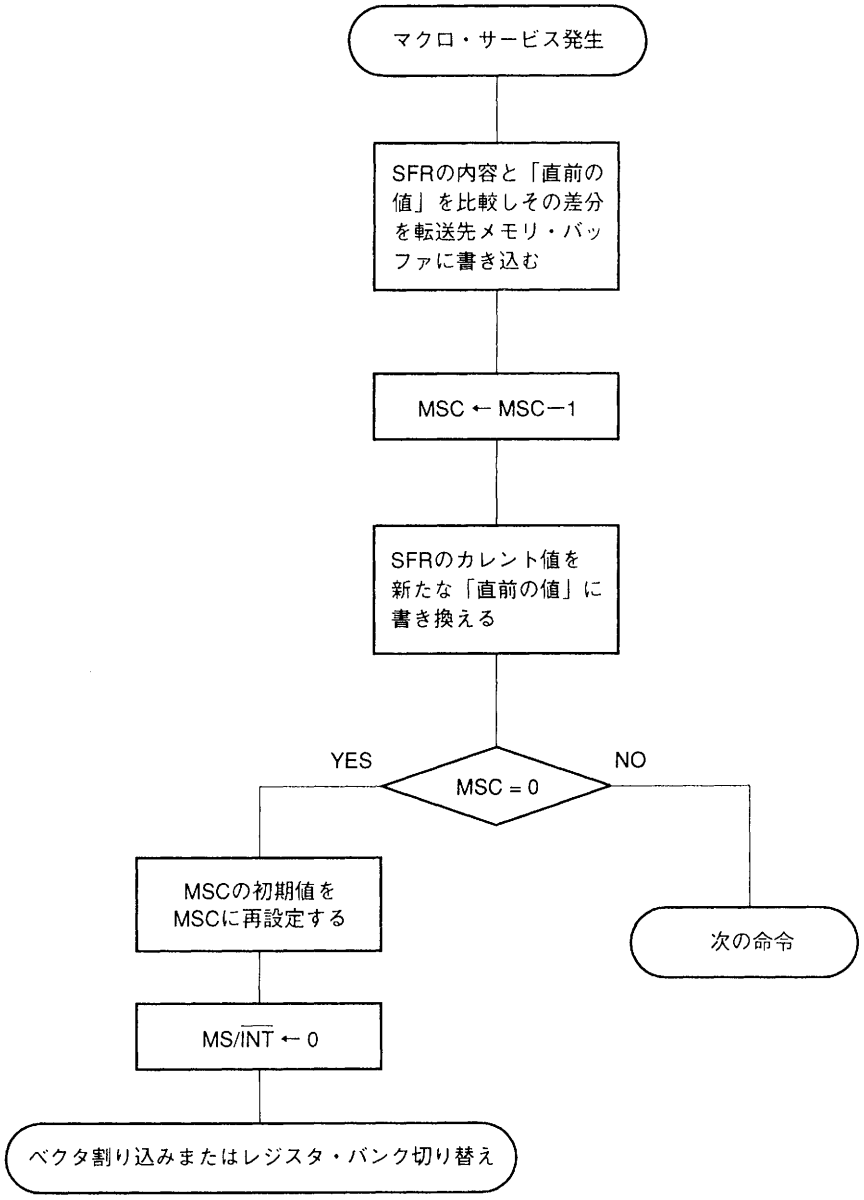
② MEM.PTR〔メモリ・ポインタ（MSP）とメモリ・セグメント（MSS)〕により，転送先の外部メモリ領域（バッファn）の最高位アドレスを指定します。MEM.PTRは，1Mバイトの基本メモリ空間をアクセスすることができます。

③ SFRPで指定されるSFR（たとえばキャプチャ・レジスタ）のカレント値と「直前の値」の差分を転送先バッファ・メモリに書き込みます。このとき，データの書き込みはバッファ1から開始します。書き込みが行われるごと（バイト／ワード）にMSCはデクリメントされますので，以後バイト転送の場合，MEM.PTR−MSC+1，ワード転送の場合MEM.PTR−MSC×2+2で示されるバッファ・メモリに結果が書き込まれます。

④ SFRのカレント値を新たな「直前の値」とし，領域に格納します。「直前の値」はバイト・モードのときは「SFRP直前の値(L)」に格納します。

⑤ MSC＝0になるとMSCの初期値をMSCに再設定し，対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$を0にし，ベクタ割り込み要求またはレジスタ・バンク切り替え要求を発生します。MSCの初期値＝0とすると256回の処理を行います。

保守／廃止



注意　転送対象となる外部メモリについて，MSS，MSPで示される転送アドレスがセグメント境界にかかる場合は，セグメントをまたいでの転送はできませんので注意してください。

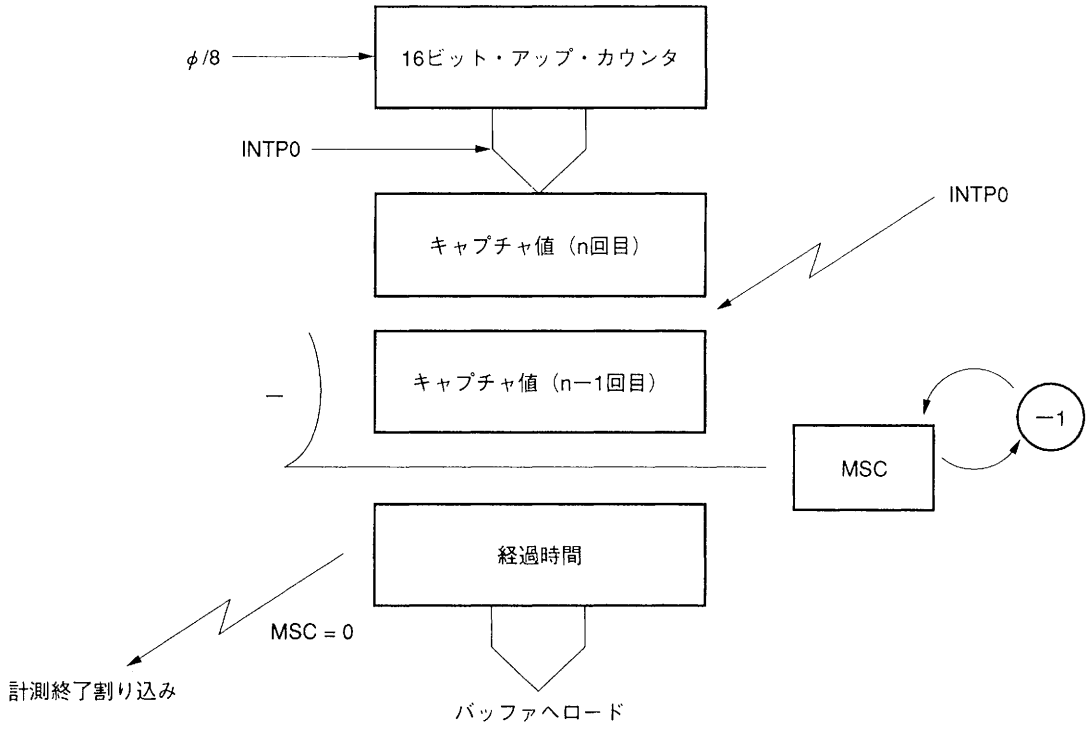
保守／廃止

バイト・データを扱うかワード・データを扱うかの指定は，マクロ・サービス・モード・レジスタでのサブコマンドWの指定に従います。

表5－5　データのバイト／ワード指定

W	バイト／ワード
0	バイト
1	ワード

例　INTP0による時間計測に使用できます。



〔用　　途〕

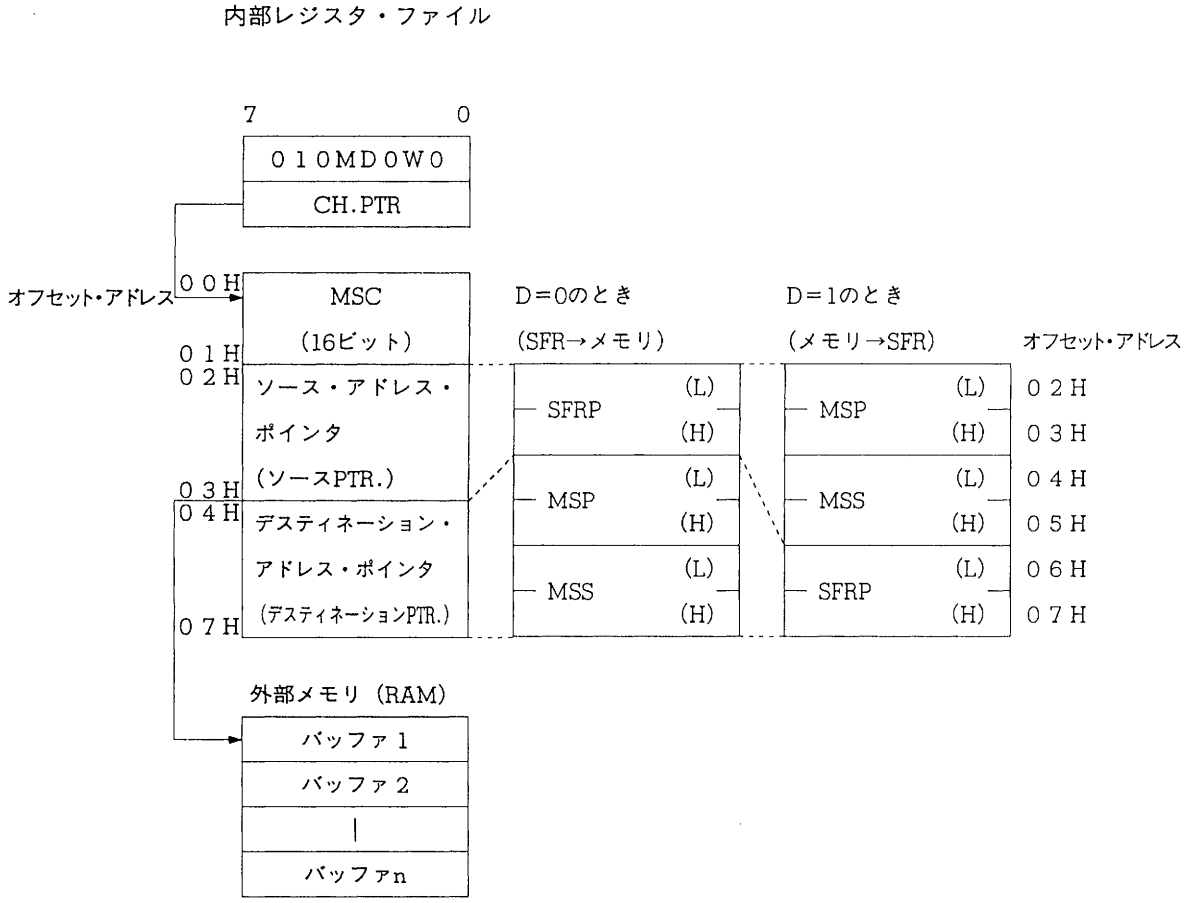
タイマ／カウンタ・ユニットのキャプチャ機能を利用したタイミング計測，A/Dコンバータによる変換値の変化量測定など

保守／廃止

(d) BLKTRS (BLOCK TRANSFER)

あらかじめ指定されたメモリ・バッファとSFR間のデータの転送を行います。あらかじめ設定した回数のデータ転送を行うと，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

〔マクロ・サービス・コントロール・ワード〕



〔動　作〕

① チャネル・ポインタ(CH.PTR)でマクロ・サービス・チャネルの先頭アドレスを指定します。

② SFRPで指定されるSFRとMEM.PTR〔メモリ・ポインタ（MSP）とメモリ・セグメント（MSS)〕でアドレスされる外部メモリ・バッファとの間でデータ転送を行います。データの転送は，バッファ1から開始します。

③ バイト／ワード・データのデータ転送終了後，MEM.PTRをインクリメント（+1/+2）します。

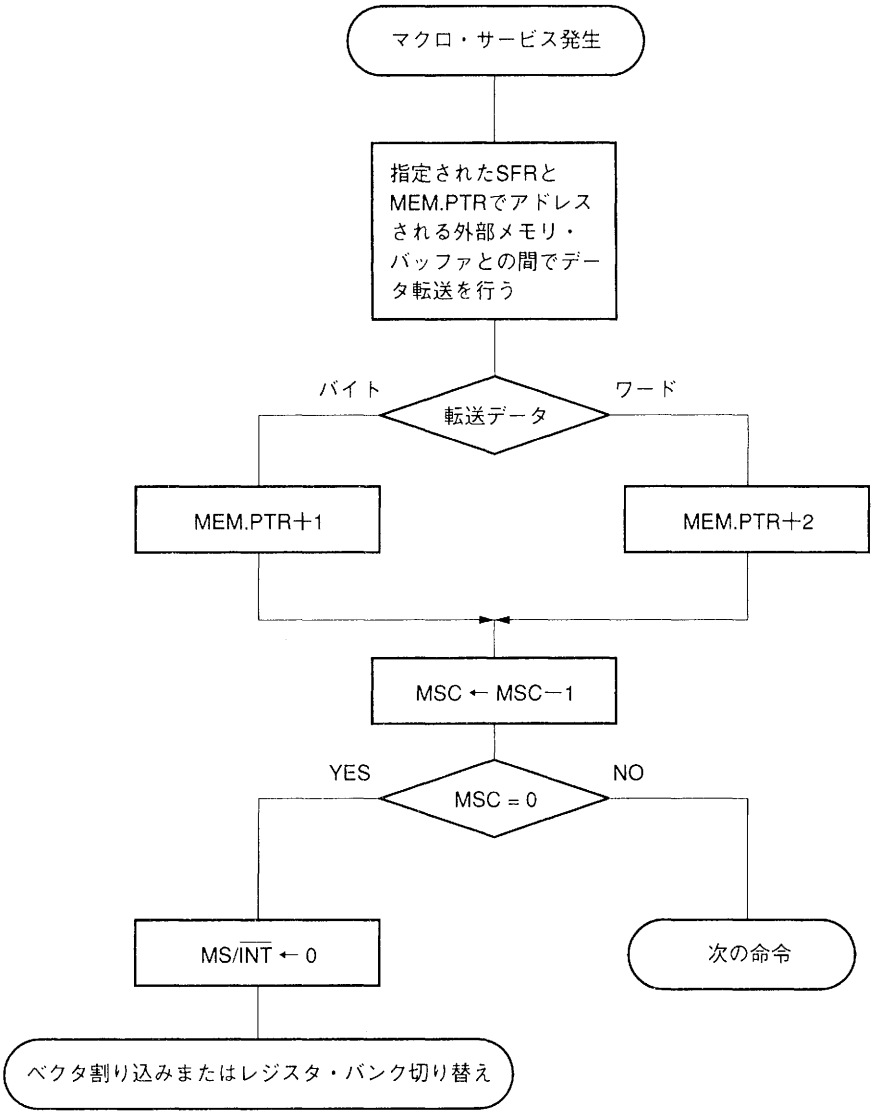
④ 転送するごとにマクロ・サービス・カウンタ（MSC）は，デクリメント（−1）されます。

⑤ MSC=0になると対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$を0にし，ベクタ割り込

保守／廃止

み要求またはレジスタ・バンク切り替え要求を発生します。なお，MSC＝0を設定すると65536回の処理を行います。

注意　転送対象となる外部メモリについて，MSS，MSPで示される転送アドレスがセグメント境界にかかる場合は転送できません。



保守／廃止

表５－６　BLKTRSモードの動作指定

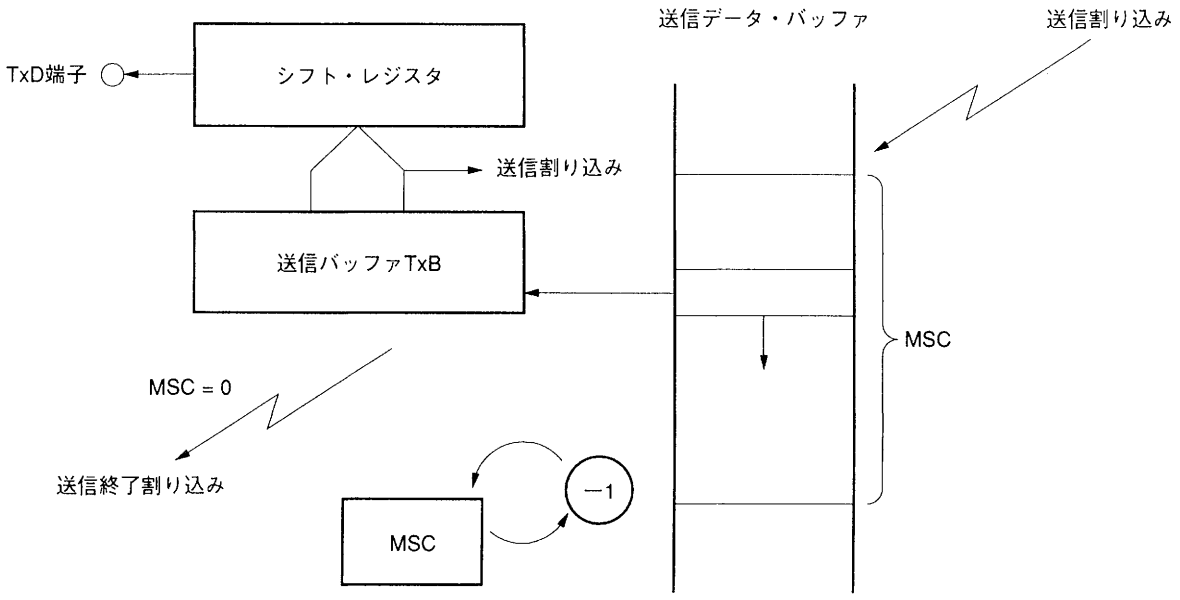
D	W	動　　作	転送データ
0	0	メモリ ← SFR	バイト
0	1	メモリ ← SFR	ワード
1	0	SFR ← メモリ	バイト
1	1	SFR ← メモリ	ワード

表５－７　メモリ空間のアドレス・モード指定

M	アドレス・モード
0	1 Mバイトの基本メモリ空間のアクセス・モード
1	16 Mバイトの拡張メモリ空間のアクセス・モード[注]

注　拡張セグメント・レジスタ（DS2，DS3）の内容には関係ありません。

例　UARTによるデータ送信に使用できます。



〔用　　途〕

UARTのデータ送受信など

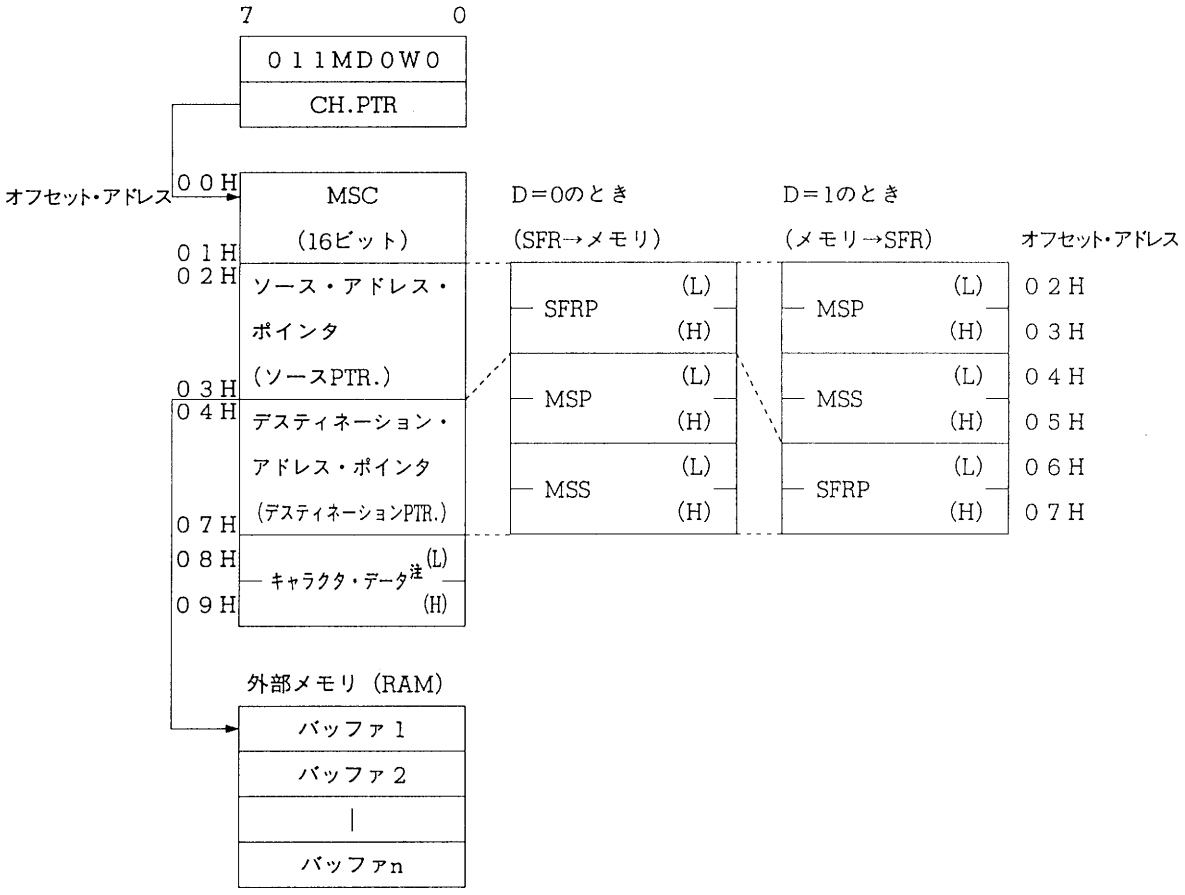
保守／廃止

(e) BLKTRS-C (BLKTRS WITH CHARACTER SEARCH)

あらかじめ指定されたメモリ・バッファとSFR間のデータ転送を行います。あらかじめ設定した回数のデータ転送を終了するか，または転送データがあらかじめ設定したキャラクタ・データと一致した場合は，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

〔マクロ・サービス・コントロール・ワード〕

内部レジスタ・ファイル



注　バイト転送モード時には，1バイト・データで指定します。

〔動　　作〕

① チャネル・ポインタ（CH.PTR）でマクロ・サービス・チャネルの先頭アドレスを指定します。

② SFRPで指定されるSFRとMEM.PTR〔メモリ・ポインタ（MSP）とメモリ・セグメント（MSS)〕でアドレスされる外部メモリ・バッファとの間でマクロ・サービスが発生するた

保守／廃止

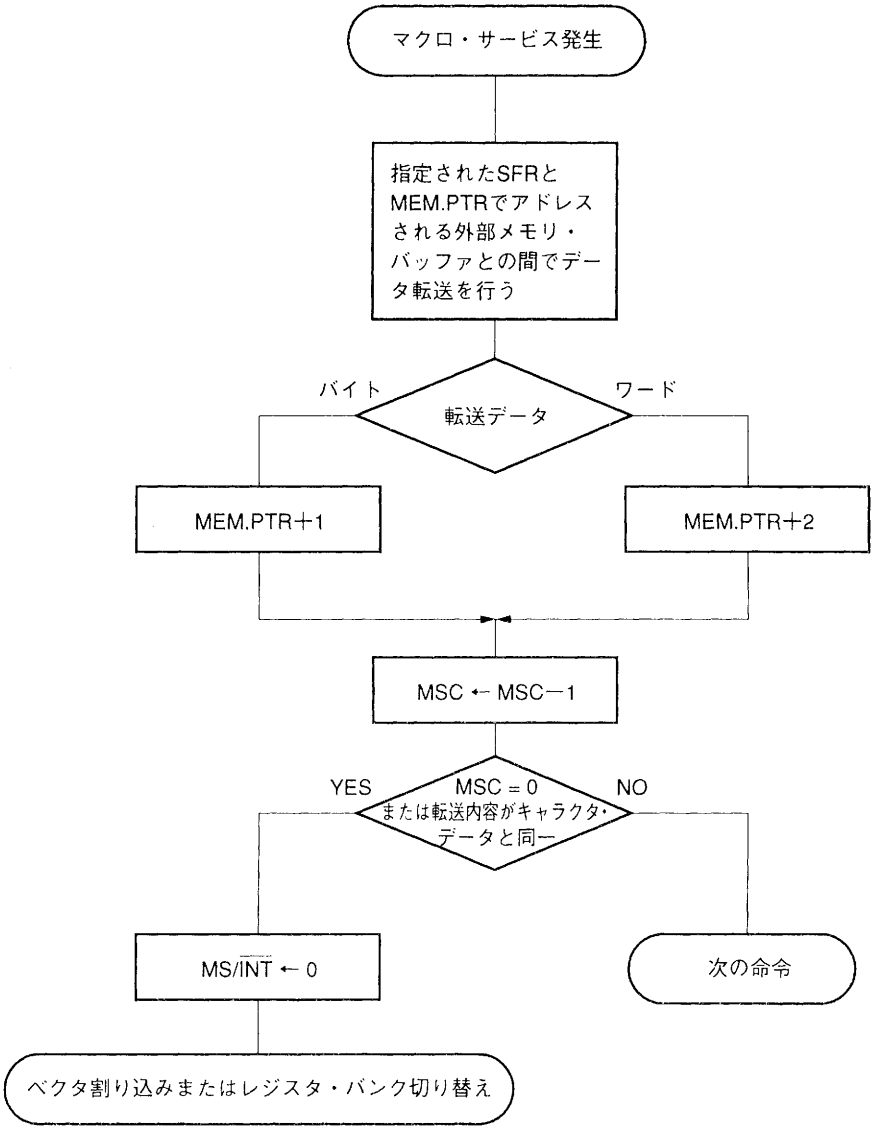
びにデータ転送を行います。データの転送は，バッファ1から開始します。

③　バイト／ワード・データのデータ転送終了後，MEM.PTRをインクリメント（+1/+2）します。

④　転送するごとにマクロ・サービス・カウンタ（MSC）は，デクリメント（−1）されます。

⑤　MSC＝0または，キャラクタ・データと同一の内容が転送されるまでマクロ・サービス処理を続けます。

⑥　MSC＝0または，キャラクタ・データと同一の内容が転送されると，対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$を0にしベクタ割り込み要求またはレジスタ・バンク切り替え要求を発生します。なお，MSC＝0を設定すると，65536回の処理を行います。



保守／廃止

表５－８　BLKTRS-Cモードの動作指定

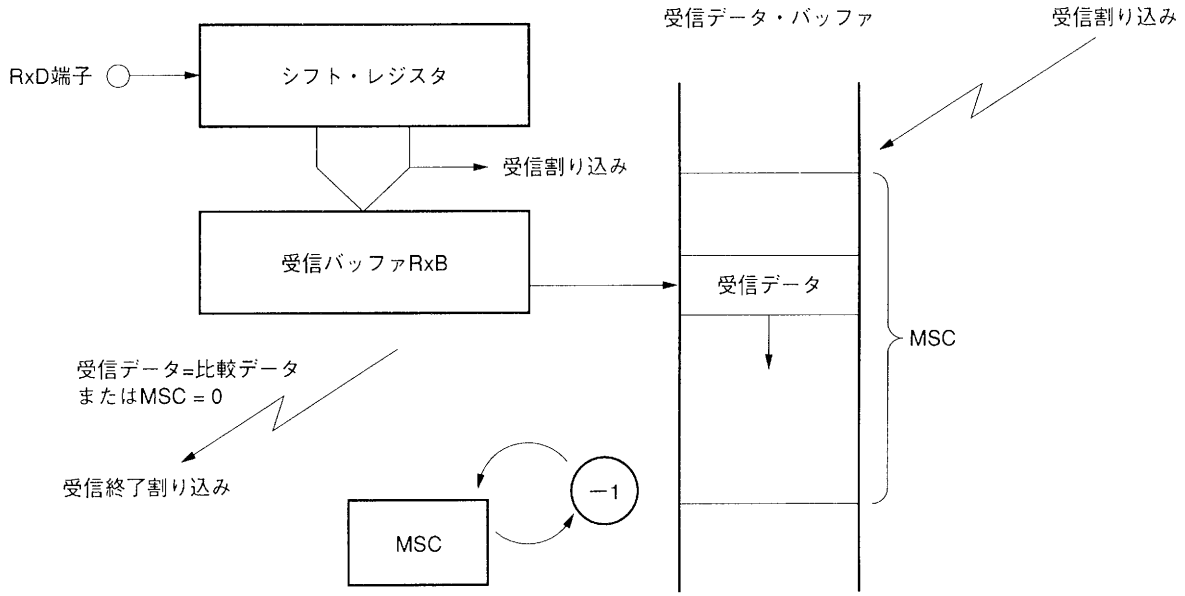
D	W	動　　作	転送データ
0	0	メモリ ← SFR	バイト
0	1	メモリ ← SFR	ワード
1	0	SFR ← メモリ	バイト
1	1	SFR ← メモリ	ワード

表５－９　メモリ空間のアドレス・モード指定

M	アドレス・モード
0	1 Mバイトのメイン・メモリ空間のアクセス・モード
1	16 Mバイトのメイン・メモリ空間のアクセス・モード[注]

注　拡張セグメント・レジスタ（DS2，DS3）の内容には関係ありません。

例　UARTによるデータ受信（終了コード付き）に使用できます。



〔用　　途〕

UARTのデータ送受信（受信終了コードの検出機能付き）など

保守／廃止

(f) RTOPTRN (REAL TIME OUTPUT PORT TRANSFER)

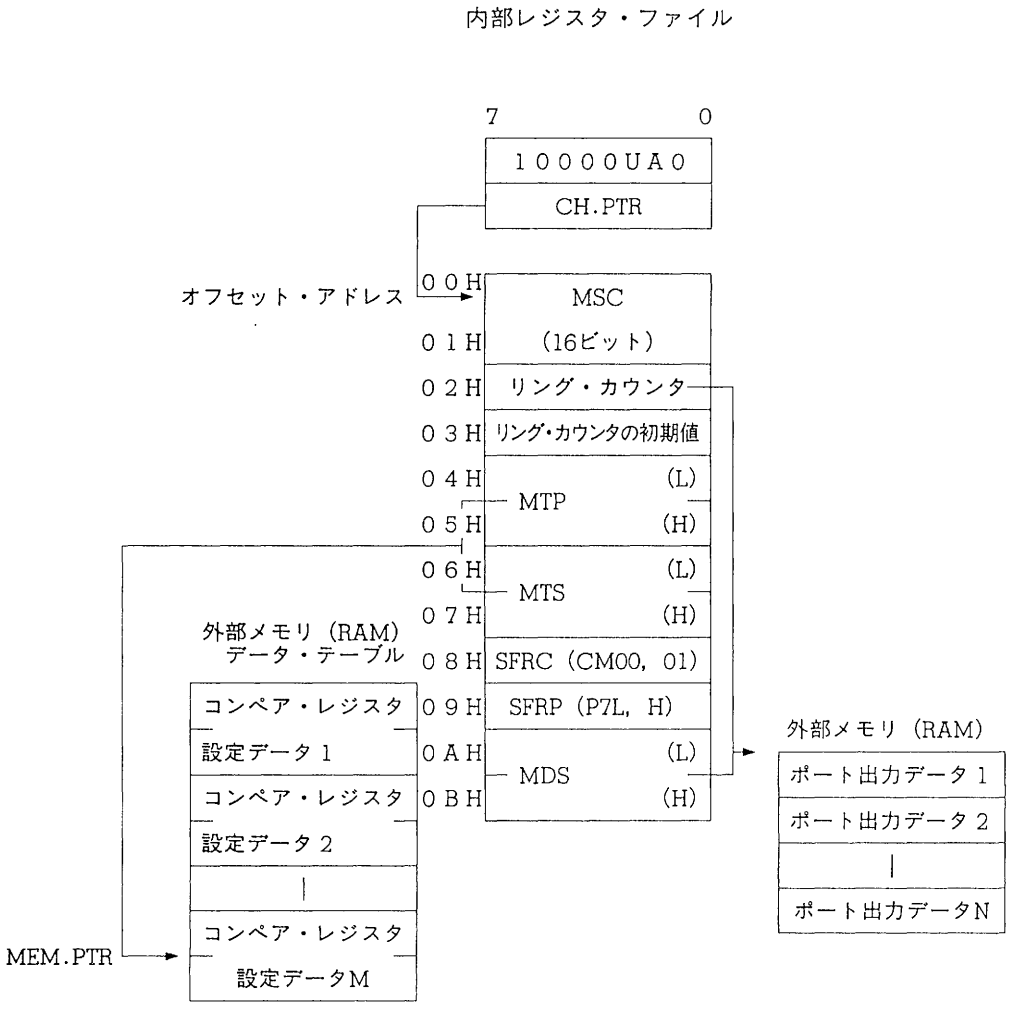
リアルタイム出力ポートに出力するデータ転送を行います。

１回のマクロ・サービス要求で次の２種類の転送を行います。

① リアルタイム出力ポートへの出力インターバル値を，メモリからSFR(タイマ・コンペア・レジスタ）に転送します。

② リアルタイム出力ポートへの出力データをメモリからSFR（P7L, P7H）に転送します。

〔マクロ・サービス・コントロール・ワード〕



備考　外部メモリは１Mバイトの基本メモリ空間です。

〔動　　作〕

チャネル・ポインタ（CH.PTR）でマクロ・サービス・チャネルの先頭アドレスを指定します。

保守／廃止

１回のマクロ・サービス転送要求に対して次の動作が行われます。

① SFRCで指定されるタイマ・コンペア・レジスタ（CM00，CM01）へMTS，MTPでアドレスされるデータ・テーブルからワード・データを転送します。

参照されるデータ・テーブル，およびテーブル上のデータの操作については表５－10，表５－11に示すように，マクロ・サービス・モード・レジスタへの設定で指定できます。

表５－10　タイマ・コンペア・レジスタ設定データ・テーブルのアドレス更新設定

U	コンペア・レジスタ設定データ参照アドレス
0	MEM. PTR（セグメント：MTS，オフセット：MTP）を更新しない
1	MEM. PTR-MSC×2+2(セグメント：MTS，オフセット：MTP)で，アドレス更新する

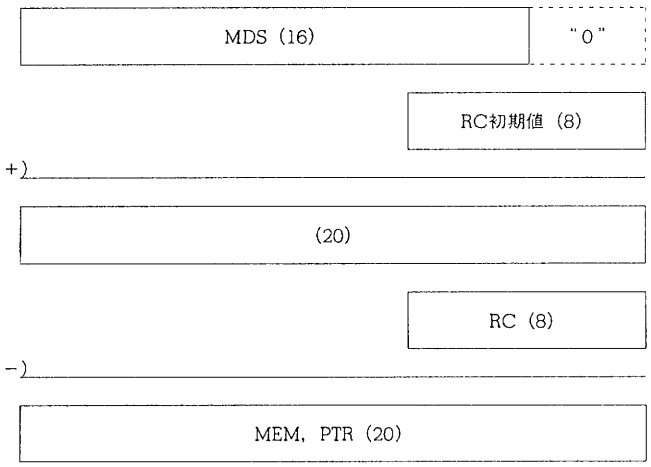
表５－11　タイマ・コンペア・レジスタ設定データへの操作指定

A	コンペア・レジスタ設定データ
0	テーブル上の設定データをそのままコンペア・レジスタへ転送
1	テーブル上設定データ＋現在のコンペア・レジスタ設定データを転送

② リアルタイム出力ポートへ出力データ（バイト・データ）を転送します。

MEM. PTR〔メモリ・セグメント（MDS）とメモリ・ポインタ（リング・カウンタの初期値－リング・カウンタ)〕でアドレスされる外部メモリからSFRPで指定されるリアルタイム出力ポートなどへバイト単位でデータを転送し，リング・カウンタの値をデクリメント（－1）します（リアルタイム出力ポート以外に，汎用ポートにデータ転送することも可能です)。

転送対象アドレス（MEM. PTR）は次のように求められます。() 内はビット数です。



保守／廃止

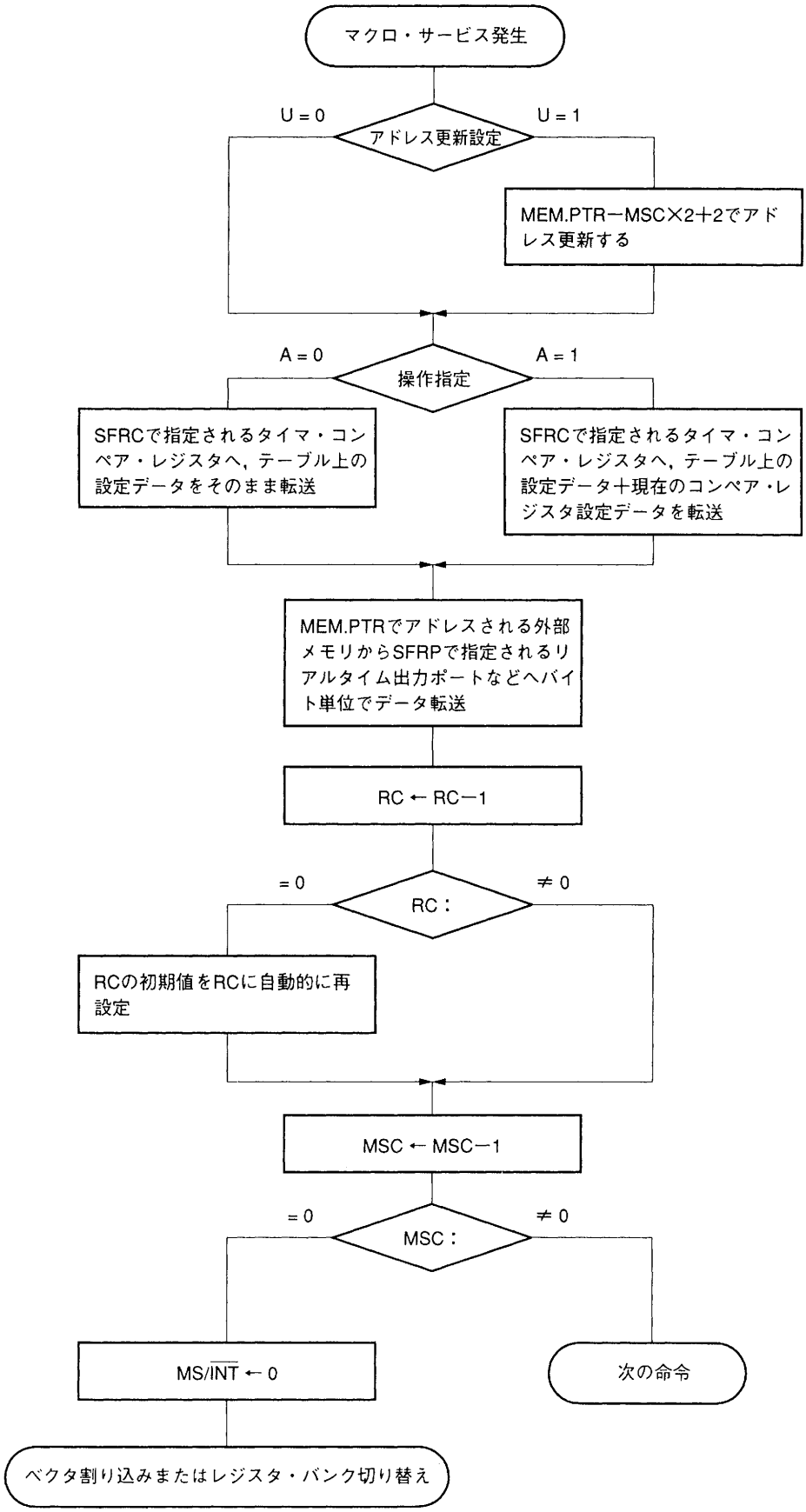
③　リング・カウンタ（RC）の値が0になったとき，RCの初期値をRCに自動的に再設定します。

④　転送するごとにマクロ・サービス・カウンタはデクリメントされます。MSC＝0になったとき，対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$を0にし，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

なお，MSCの初期値としてMSC＝0を設定すると，65536回転送処理を行います。

SFRC，SFRPにはそれぞれSFRアドレスの下位8ビットを設定します。

保守／廃止


マクロ・サービス発生
U = 0
アドレス更新設定
U = 1
MEM.PTR−MSC×2+2でアドレス更新する
A = 0
操作指定
A = 1
SFRCで指定されるタイマ・コンペア・レジスタへ，テーブル上の設定データをそのまま転送
SFRCで指定されるタイマ・コンペア・レジスタへ，テーブル上の設定データ＋現在のコンペア・レジスタ設定データを転送
MEM.PTRでアドレスされる外部メモリからSFRPで指定されるリアルタイム出力ポートなどへバイト単位でデータ転送
RC ← RC−1
= 0
RC：
≠ 0
RCの初期値をRCに自動的に再設定
MSC ← MSC−1
= 0
MSC：
≠ 0
MS/INT ← 0
次の命令
ベクタ割り込みまたはレジスタ・バンク切り替え

保守／廃止

注意1．各割り込み要求（INTCM00，INTCM01）に対して，マクロ・サービス転送対象となるタイマ・コンペア・レジスタおよびリアルタイム出力ポートはそれぞれ次のように決まっています。SFRC，SFRPを設定する際は，注意してください。

表5－12　割り込み要求信号とSFRの対応

割り込み要求信号	タイマ・コンペア・レジスタ	ポート7バッファ
INTCM00	CM00	P7L
INTCM01	CM01	P7H

2．リアルタイム出力ポートへの書き込みは，リアルタイム出力ポート・コントロール・レジスタ（RTPC）のBYTEビットの指定により次のようになります（第10章　リアルタイム出力ポート機能参照）。

表5－13　リアルタイム出力ポートへの書き込み

BYTE	割り込み要求信号	リアルタイム出力ポートへの書き込み
0	INTCM00	下位4ビットがP7Lに書き込まれる
	INTCM01	上位4ビットがP7Hに書き込まれる
1	INTCM01	8ビット単位でP7H，P7Lに書き込まれる

〔用　途〕

リアルタイム出力用ポート7バッファ，およびタイマ・コンペア・レジスタへのデータ転送（ステッピング・モータ制御）など（図10－4　リアルタイム出力ポートによるステッピング・モータ制御参照）

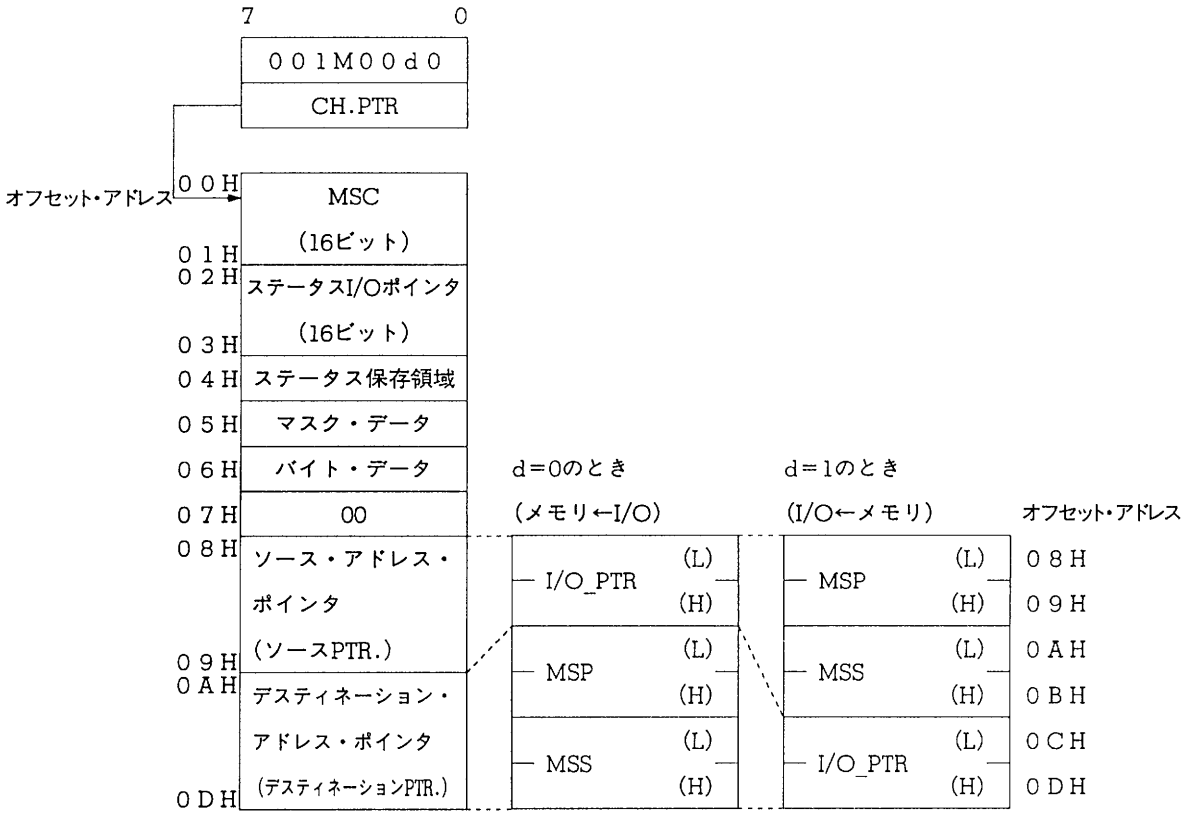
保守／廃止

(g) DTACMP-M (DATA COMPARE WITH CHARACTER MASK)

外部I/Oから読み込んだステータス・データと，あらかじめ設定したマスク・データとの論理積をとります。その結果と，あらかじめ設定したバイト・データを比較します。一致すれば外部I/Oとメモリ間のデータ転送を行います。一致しない場合，またはあらかじめ設定した回数のデータ転送を行うと，対応する割り込み要因のマクロ・サービスを終了し，ベクタ割り込みまたはレジスタ・バンク切り替えを発生します。

〔マクロ・サービス・コントロール・ワード〕

内部レジスタ・ファイル



〔動　作〕

① ステータスI/Oポインタで示されるアドレスの外部I/Oのバイト・データを読み込み，ステータス保存領域に格納します。

② ステータス保存領域に格納したデータとマスク・データ領域のデータとの論理積をとります。その結果とバイト・データ領域のデータを比較します。

③ 比較した結果，データが一致した場合は，I/O_PTRでアドレスされる外部I/OとMSP，MSSでアドレスされる外部メモリ・バッファとの間でデータ転送を行います。一致しない場合は，対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$ビットをクリア（0）し，ベクタ割り込

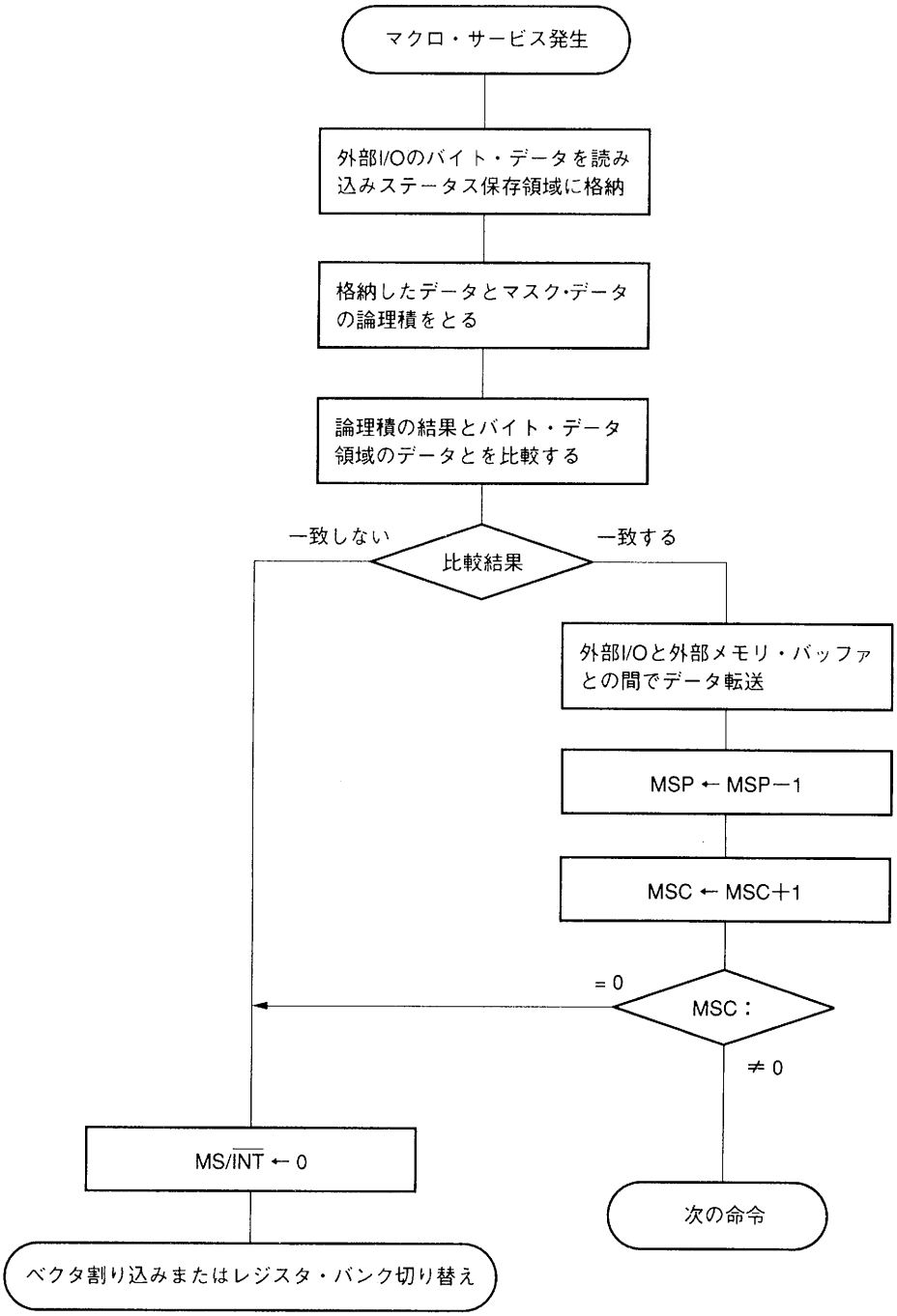
保守／廃止

み要求またはレジスタ・バンク切り替え要求を発生させます。

④　1回のデータ転送ごとにMSPをインクリメント，MSCをデクリメントします。MSCをデクリメントした結果，MSC＝0となった場合は，対応する割り込み要求制御レジスタのMS/$\overline{\text{INT}}$ビットをクリア（0）し，ベクタ割り込み要求またはレジスタ・バンク切り替え要求を発生させます。

備考　このマクロ・サービス動作モードは，外部I/O-メモリ間の転送を対象とします。SFR-メモリ間の転送については，**5.4.3(6)(d) BLKTRS（BLOCK TRANSFER），(e)BLKTRS-C（BLKTRS WITH CHARACTER SEARCH）**を参照してください。

保守／廃止



〔用　　途〕

モデムとの接続など

保守／廃止

5.5　制御レジスタ

割り込み制御レジスタには，割り込み優先順位モードの切り替えとベクタ・テーブル指定を行う割り込みモード・コントロール・レジスタ（IMC）と，マスカブル割り込みを制御するレジスタとして，割り込みマスク・フラグ・レジスタ（MK0，MK1），割り込み要求制御レジスタ（INT××），インサービス・プライオリティ・レジスタ（ISPR）があります。

5.5.1　割り込みモード・コントロール・レジスタ（IMC）

優先順位制御モードの切り替え，およびベクタ・テーブルの指定を行います。

NMIを含むすべての割り込み機能を使用する前に，データ設定を行う必要があります。

IMCレジスタはリセット後に一度だけ書き込みが可能です。その後書き換え操作を行っても書き込みデータを無視します。

ベクタ・テーブルは，00000H-0003FFHのベクタ・テーブル領域内で，システム（V55PIの内蔵機能）の使用するベクタ領域のタイプをV0，V1ビットによって指定することができます。ベクタ・テーブル領域の詳細は**3.5.4　ベクタ・テーブル領域**を参照してください。

また，PRSLビットによる優先順位制御モードについては，**5.3.3　多重割り込みの制御モード選択**を，具体的な処理動作については，**5.12　割り込み要求の処理動作例**を参照してください。

注意　リセット後の初期状態（リセット時の値＝80H）から変更が不要な場合でも，必ず一度レジスタ設定を行ってください。これは，その後の誤った書き込み動作によるシステムの異常動作を防ぐためです。

保守／廃止

	7	6	5	4	3	2	1	0
IMC	PRSL	0	V1	V0	0	0	0	0

ベクタ・テーブルの指定

V1	V0	システムの使用するベクタ・テーブル	アドレス
0	0	8-47（初期値）	00020H-000BCH
0	1	72-111	00120H-001BCH
1	0	136-175	00220H-002BCH
1	1	200-239	00320H-003BCH

優先順位制御モード選択

（レベル３のマスカブル割り込みについて）

PRSL	優先順位制御
0	優先順位制御を行わない
1	優先順位制御を行う

注意　割り込みシーケンスの誤動作を防止するために，このレジスタの書き換え操作は，割り込み禁止状態のときに行う必要があります。したがって，操作前に必ずDI命令を実行してください。

備考1.　　はリセット後の状態を示します。

2. V55PIでは，リセット後，ベクタ８-ベクタ47は内蔵周辺ハードウエア用のシステム領域となっていますが，IMCレジスタのV0，V1ビットで，ベクタ８以降のベクタ・テーブル領域の指定を容易に変更することができます。このようにして，ほかのVシリーズ応用製品のプログラムを簡単に移植できます。

例　V20，V30：ベクタ０-５（CPU専用），ベクタ６-31（予約領域）
　　V25，V35：ベクタ０-７（CPU専用），ベクタ８-31（予約領域）

保守／廃止

5.5.2　割り込みマスク・フラグ・レジスタ(MK0L, MK0H, MK1L, MK1H)

割り込みマスク・フラグ・レジスタは，レジスタ中の各ビットに対応したマスカブル割り込み要求について処理の禁止／許可を指定するレジスタです。各マスク・フラグの番号と割り込み要求との関係は，表5－1に示したディフォールト・ベクタ・テーブルの番号と対応しています。ビット操作命令により各フラグをビット単位で操作可能です。リセット後の値は，FFHになります。

割り込み要求制御レジスタ(IC××)のMKビットと機能的，物理的に同一のものです。つまり，割り込み要求制御レジスタまたは割り込みマスク・フラグ・レジスタの値を変更すると他方のレジスタの値が自動的に変化します（16/8，または1ビット単位のメモリ・アクセス命令でも操作可能です）。

	7	6	5	4	3	2	1	0
MK0L	1	MK14	MK13	MK12	MK11	MK10	MK09	1
MK0H	MK23	MK22	MK21	MK20	MK19	MK18	MK17	MK16
MK1L	MK31	MK30	MK29	MK28	MK27	MK26	MK25	MK24
MK1H	1	1	MK37	MK36	1	1	1	MK32

MK××	割り込み処理の許可／禁止の指定
0	割り込み処理を許可
1	割り込み処理を禁止

備考　□ はリセット後の状態を示します。

保守／廃止

5.5.3　割り込み要求制御レジスタ（IC9-14，16-32，36，37）

マスカブル割り込みに対する要因ごとの割り込み制御を設定します。

リセット後の値は，43Hになります。

	7	6	5	4	3	2	1	0
IC××	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0

優先順位の指定

PR1	PR0	指定する優先順位
0	0	レベル0
0	1	レベル1
1	0	レベル2
1	1	レベル3

割り込み処理形態の指定

MS/$\overline{\text{INT}}$	ENCS	動　作
0	0	ベクタ割り込み処理
0	1	レジスタ・バンク切り替え処理
1	0	マクロ・サービス処理＋終了後ベクタ割り込み
1	1	マクロ・サービス処理＋終了後レジスタ・バンク切り替え

MS/$\overline{\text{INT}}$：マクロ・サービス処理フラグ

ENCS：レジスタ・バンク切り替え処理フラグ

割り込みのマスク指定

MK	マスクの指定
0	マスクしない
1	マスクする

割り込みの応答表示

IF	割り込み要求の有無
0	割り込み要求なし
1	割り込み要求あり

備考　　はリセット後の状態を示します。

保守／廃止

IF　割り込み要求フラグ

IFフラグは対応する割り込み要求が発生した時点でセット(1)され，受け付けられると自動的にクリア(0)されます。

割り込み処理の起動に利用されます。IFフラグの操作はソフトウエアでもセット／リセット可能です。したがって，ソフトウエアによってマスカブル割り込みを発生させることができます。リセット後は“0”に初期化されます。

MK　割り込みマスク・フラグ

割り込みマスク・フラグ・レジスタ（MK0，MK1）にあるフラグと，機能的に同一のものです（どちらか一方の操作で他方の該当フラグも同内容に書き換わります）。

リセット後は“1”に初期化され，割り込み禁止状態になります。

MS/INT　マクロ・サービス処理ビット

割り込み要求をマクロ・サービスで処理することを指定します。

リセット後は“0”に初期化されます（ベクタ割り込み処理指定となります）。

ENCS　レジスタ・バンク切り替え処理ビット

割り込み要求をレジスタ・バンク切り替えで処理することを指定します。

リセット後は“0”に初期化されます（ベクタ割り込み処理指定となります）。

PR1　**PR0**　優先順位指定ビット

多重割り込み処理で使用され，マスカブル割り込みの優先順位をレベル0からレベル3までの4段階に設定可能です（レベル0の割り込みが最上位です）。

リセット後は“11”に初期化され，最下位の優先順位レベルを指定します。

備考　PR1，PR0の内容の影響を受けるのは，ベクタ割り込み，レジスタ・バンク切り替えで処理されるマスカブル割り込みに限ります。マクロ・サービスは，このビットの内容に影響を受けません。常に最高レベルの割り込みとして処理されます(**5.12　割り込み要求の処理動作例**参照)。

保守／廃止

割り込み要求制御レジスタ（IC××）と割り込み要因との対応を示します。

IC09	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTP0)
IC10	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTP1)
IC11	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTP2)
IC12	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTP3)
IC13	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTP4)
IC14	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTP5)
IC16	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTCM00)
IC17	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTCM01)
IC18	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTCM10)
IC19	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTCM11)
IC20	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTCM21)
IC21	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTCM31)
IC22	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTD0 DMA≠0_MAIN)
IC23	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTD0S DMA≠0_SUB)
IC24	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTD1 DMA≠1_MAIN)
IC25	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTD1S DMA≠1_SUB)
IC26[注]	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTSER0)
IC27[注]	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTSER1)
IC28	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTSR0/INTCSI0)
IC29	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTSR1/INTCSI1)
IC30	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTST0)
IC31	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTST1)
IC32[注]	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTSIT)
IC36	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTPAI)
IC37	IF	MK	MS/$\overline{\text{INT}}$	ENCS	0	0	PR1	PR0	(INTAD)

注　MS/$\overline{\text{INT}}$には必ず0を設定してください。

保守／廃止

5.5.4　インサービス・プライオリティ・レジスタ（ISPR）

現在処理中のマスカブル割り込み処理の優先順位レベルを記憶，保持します。割り込みコントローラが優先順位の制御を行う際に使用します。

割り込み要求を受け付けると優先順位レベルに対応するビットを順にセットします。以降，多重割り込みを受け付けるとそのたびに該当するビットをセット(1)します。

ビットのリセット動作は，FINT命令の実行により，ISPRレジスタ内で最も優先順位の高い割り込みに対応するビットが自動的にクリアされます。

	7	6	5	4	3	2	1	0
ISPR	0	0	0	0	レベル3	レベル2	レベル1	レベル0

各優先順位の割り込み処理状態

0	割り込み要求を受け付けていない
1	割り込み要求を受け付け中

注意1．マスカブル割り込みの処理ルーチンからRETI命令，RETRBI命令で復帰する場合，必ず復帰命令の直前でFINT命令を実行してください。

2．ノンマスカブル割り込み，ソフトウエア割り込みの処理ルーチンから復帰する場合は，FINT命令は実行しないでください。

備考　□ はリセット後の状態を示します。

保守／廃止

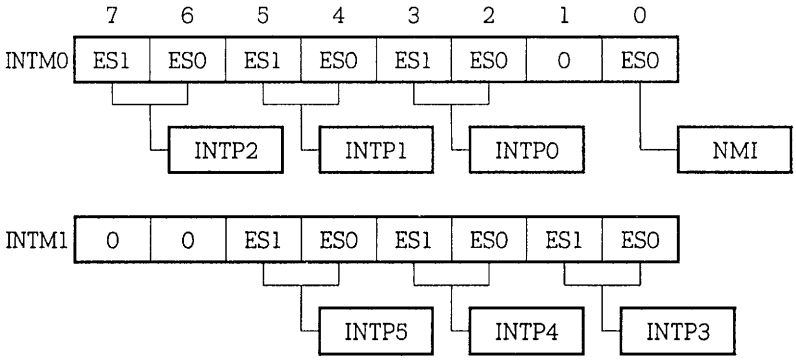
5.6　外部割り込み

外部からの割り込み要求としてNMI，INTP0-INTP5の7本があります。

5.6.1　外部割り込みモード・レジスタ（INTM0，INTM1）

外部割り込みの受け付けは，有効エッジを選択可能であり，特殊機能レジスタの外部割り込みモード・レジスタ（INTM0，INTM1）によって指定することが可能です。

INTM0，INTM1の各レジスタのリセット後の値は，00Hになります。



有効エッジの指定

ES1	ES0	有効エッジ
0	0	立ち下がりエッジ
0	1	立ち上がりエッジ
1	0	設定禁止
1	1	立ち上がりおよび立ち下がりエッジ

注意　**NMIでは，“立ち上がりおよび立ち下がりエッジ”は選択できません。**

備考　▒ はリセット後の状態を示します。

外部割り込み端子（INTP0-INTP5）は，ほかの内蔵周辺ハードウエア・ユニットのトリガ信号発生機能を兼用しています（タイマ・ユニット，A/Dコンバータ，パラレル・インタフェース・コントローラ）。

外部割り込み端子として使用しない場合には，割り込みマスク・フラグ・レジスタでマスクしておく必要があります。特にINTP3，INTP4，INTP5端子については，兼用トリガ機能の禁止／許可制御が必要となります。

保守／廃止

5.7　ソフトウエア割り込み

V55PIには合計11種類のソフトウエアによる割り込みがあります（表5－14参照）。

V25，V35のソフトウエアによる割り込みとコンパチブルです。

ソフトウエア割り込みは，内蔵の割り込みコントローラの制御は受けず，また，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フラグによる割り込み(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(ハードウエアによる割り込みより優先されます)。また，NMIとソフトウエア割り込みが競合した場合も，ソフトウエア割り込みの方が優先されます。

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，ハードウエア，ソフトウエアの別なく，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，ほかの割り込み(ハードウエア，ソフトウエアいずれのも)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5－14　ソフトウエアによる割り込み

割り込みソース	ベクタ	優先順位
DIVUのディバイド・エラー DIVのディバイド・エラー	0	1
BRKフラグ（シングルステップ）	1	2
BRK3	3	1
BRKV	4	
CHKIND境界オーバ	5	
入出力命令（$\overline{\text{IBRK}}$フラグ）	6	
BRK imm8	注	
BRKCS命令	—	
FPO命令 例外トラップ	7	

注　0-255（0-FFH）の任意の値（imm8で指定）

5.7.1　一般のソフトウエア割り込み

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エアによる割り込みの受け付けのシーケンスは，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。

次に各ソフトウエア割り込みについて説明します。

保守／廃止

(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(1)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(1)されていると，1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。

5.7.2　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割り込みを発生します。**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，一般のソフトウエア割り込み（5.7.1　一般のソフトウエア割り込み参照）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。このほかの動作は一般のソフトウエ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の復帰時には，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，ソフトウエ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V20，V30で使用していた外部I/Oを使用するプログラムを簡単に移植できます。

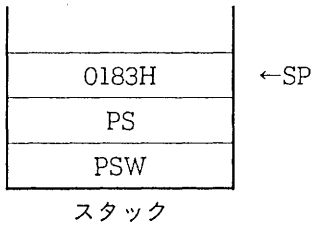
また，PSWについては割り込みが発生する直前の内容がスタックに退避され，その後，IE＝BRK＝0，$\overline{\text{IBRK}}$＝1に自動的に設定されます。$\overline{\text{IBRK}}$＝1にすることにより，割り込み処理中の命令は入出力命令として実行され，割り込みからの復帰により自動的に元の状態（$\overline{\text{IBRK}}$＝0）に戻ります。

保守／廃止

例1. プリフィクスなし

アドレス　命令

0183H IN　　AW, DW

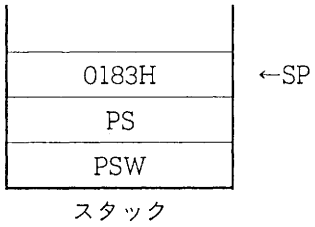


例2. プリフィクスあり

アドレス　命令

0183H REP

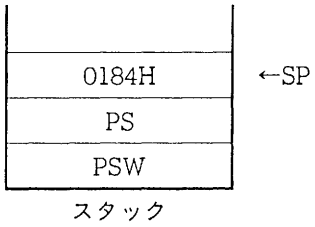
0814H INM



参考　一般のソフトウエア割り込み

アドレス命令

0183H BRK 3



5.7.3　FPO命令割り込み

V55PIは，V20，V30と外部バスの構成が異なるため，V20，V30用の浮動小数点演算用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が発生します。**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の値が，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(5.7.2　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，ソフトウエ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。

保守／廃止

5.8　例外トラップ

例外トラップはウォッチドッグ・タイマ・モード・レジスタ（WDM）の不正書き込みにより発生します。

この割り込みはV25，V35にはなく，V55PIでウォッチドッグ・タイマを備えるにあたり，新たに追加された機能です。

例外トラップ割り込みはベクタ7で処理され，FPO割り込みとベクタ番号が共通です。

WDMレジスタへの書き込みの命令については，**12.2　構成と動作**を参照してください。

5.9　ハードウエア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ハードウエア割り込み要求は受け付けられません。

(1) セグメント・レジスタsreg，および拡張セグメント・レジスタxsregに対する操作命令

MOV sreg，reg16 ；MOV sreg，mem16 ；POP sreg

MOV xsreg，reg16；MOV xsreg，mem16；POP xsreg

(2) プログラム・ステータス・ワード（PSW）に対する操作命令

POP PSW

(3) プリフィクス命令

PS：，SS：，DS0：，DS1：，DS2：，DS3：，IRAM：，

REPC，REPNC，REP，REPE，REPZ，REPNE，REPNZ，

BUSLOCK

(4) EI，DI，RETI，RETRBI

(5) FINT

(6) BRKCS，BRK3，BRKimm8，

BRKV（V＝1のとき），

CHKIND（mem32＞reg16または（mem32＋2）＜reg16のとき）

FPO1 fp-op，FPO1 fp-op，mem，

FPO2 fp-op，FPO2 fp-op，mem，

TSKSW，MOVSPB，MOVSPA

DIV，DIVU（ディバイド・エラー時）

(7) RSTWDT

受け付けられないタイミングで発生したハードウエア割り込み要求は，続く命令の実行終了時に受け付け可能状態であれば受け付けられます。

また，ハードウエア割り込み要求が受け付けられた場合，ほかの優先順位の高いハードウエア割り込み要求は，割り込み処理ルーチン内でEI命令が実行されたあとに受け付けられます。

保守／廃止

5.10　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要求を受け付けます。

割り込み要求は1回のデータ転送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最大4種のプリフィクスを付加することができます。V55PI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断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に退避します。

例　REP
　　MOVBK SS：SRC_BLK，DES_BLOCK

5.11　コーデック命令実行中の割り込み処理

9つのコーデック命令のうち，次の6命令では命令実行中でも割り込み要求を受け付けます。

命　　令	オペランド	命令の処理単位
変化点テーブル作成命令	COLTRP	変化点テーブルの1ワード分の画素データ処理
MH符号化命令	MHENC	変化点テーブルの1ワード分の符号化処理
MR符号化命令	MRENC	MR符号化の1モード単位（パス・モード，垂直モード，水平モードのどれか）の符号化処理
MH復号化変化点テーブル作成命令	MHDEC	変化点テーブルの1ワード分の復号化処理
MR復号化変化点テーブル作成命令	MRDEC	MR復号化の1モード単位（パス・モード，垂直モード，水平モードのどれか）の復号化処理
画素データ作成命令	CNVTRP	変化点テーブルの1ワード分の画素データ変換処理

割り込み要求は，1回の命令処理単位の実行終了ごとにサンプリングされます。マスク指定，優先順位制御により割り込み要求が受け付けられる状態であれば，コーデック命令の実行を中断し，割り込み処理へ分岐します。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，実行中のコーデック命令（付加されたプリフィクスを含む）の先頭アドレスを退避します。

中断されたコーデック命令で使用中のパラメータ（レジスタまたはレジスタ・バンク上）は，割り込み処理からの復帰時に継続して使用されます。ただし，割り込み処理からの復帰時には元の状態であることが必要です。したがって，割り込み処理の中でデータの操作をしないようにしてください。

保守／廃止

外部データ・バスが16ビットでノー・ウエイト時のA4サイズ（1728ビット／ライン）の処理実行時における，各命令の割り込み要求の非サンプリング時間（最大値）を次に示します。

（単位：μs）

処理	方式	COLTRP	MHENC	MRENC	MHDEC	MRDEC	CNVTRP
符号化	MH	4.6	19	—	—	—	—
	MR	4.6	19	24	—	—	—
復号化	MH	—	—	—	23.52	—	58.16
	MR	—	—	—	23.52	33	58.16

なお，割り込みのサンプリングを行わない命令（ALBIT，SCHEOL，GETBIT）の場合，命令実行中に割り込み要求が発生すると，割り込み要因ごとに最大1つ保留されます。したがって，コーデック命令終了後に割り込み許可（EI）状態であれば，保留されていた割り込みが受け付けられます。

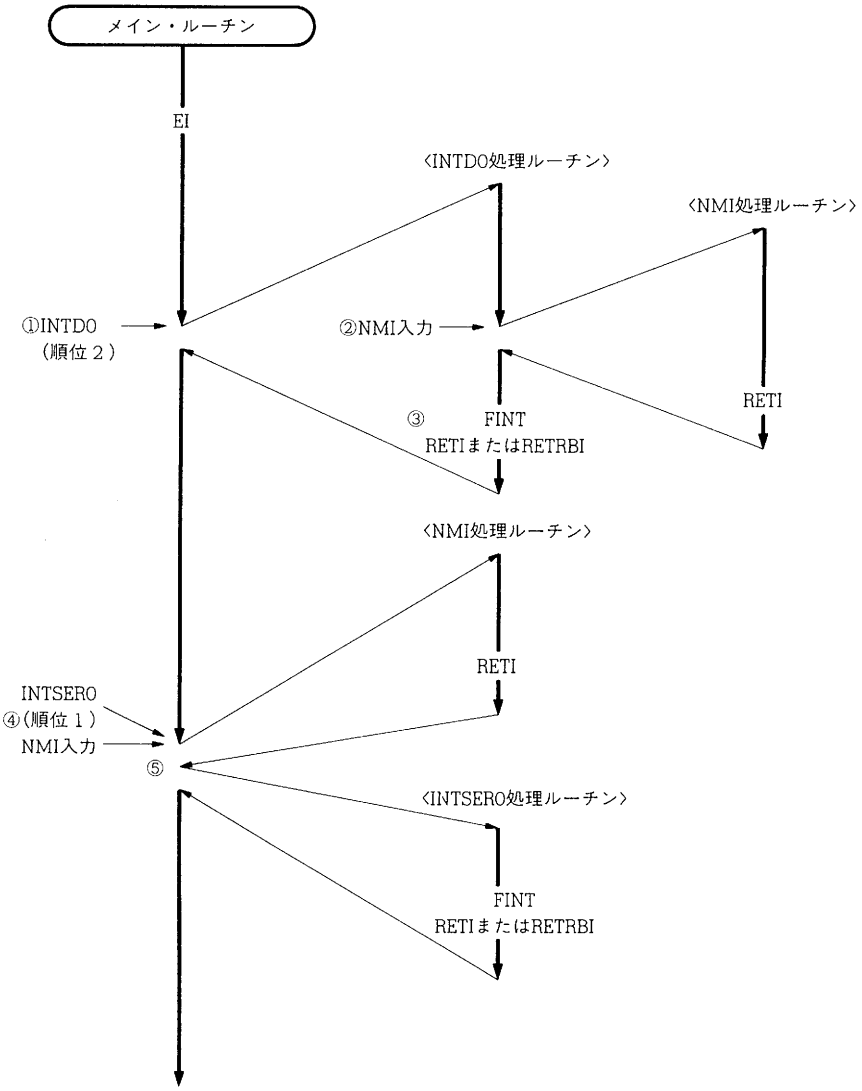
保守／廃止

5.12　割り込み要求の処理動作例

(1) NMI

○ソフトウエアによってマスクされません。

○すべての割り込みに対して最優先に受け付けられます。



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処理を開始します。

②　INTDO処理中は割り込み禁止状態(DI)になりますが，NMIの発生により，INTDO処理を中断しNMI処理を開始します。

③　NMI処理が終了すると中断されていたINTDOが処理され，復帰命令を実行して復帰します。

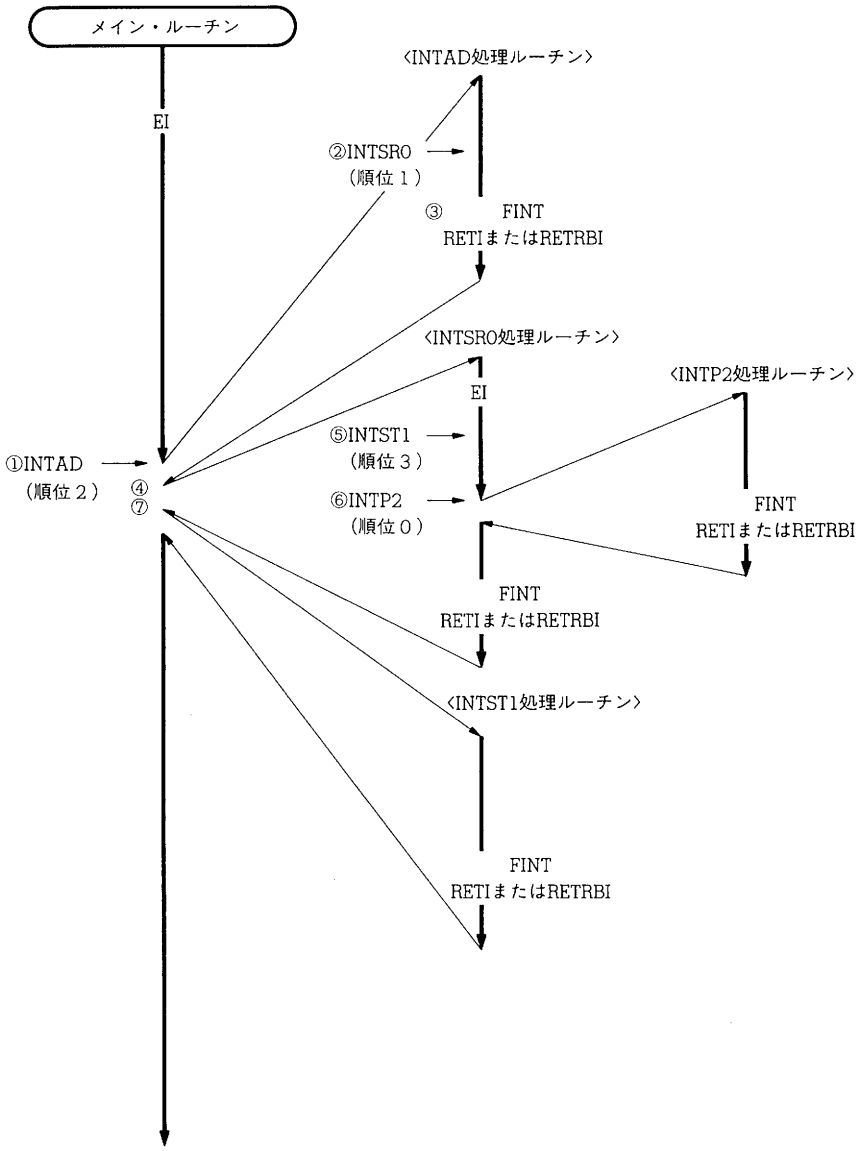
④　優先順位１に設定したINTSEROとNMIが同時に発生すると，すべての割り込みに対して最優先のNMIが受け付けられ，NMI処理を開始します。

⑤　NMI処理が終了すると保留されていたINTSEROが処理されます。

保守／廃止

（2）多重割り込み処理制御を受ける割り込み……A

○優先順位に従って多重処理を行います。



①　割り込み許可状態（EI）で優先順位２に設定したINTADが発生し，INTAD処理を開始します。

②　INTADが受け付けられると割り込み禁止状態（DI）になりますので，INTADより高位の優先順位１に設定したINTSR0が発生しますが受け付けられません。

③　優先順位制御を受ける割り込みは，復帰時にFINT命令を実行しなければなりません。

④　INTAD処理が終了し割り込み許可状態になると，保留されていたINTSR0処理を開始します。

⑤　割り込み許可状態（EI）で優先順位３に設定したINTST1が発生しますが，INTSR0の割り込み処理中でありINTSR0より優先順位が低位ですので，受け付けられません。

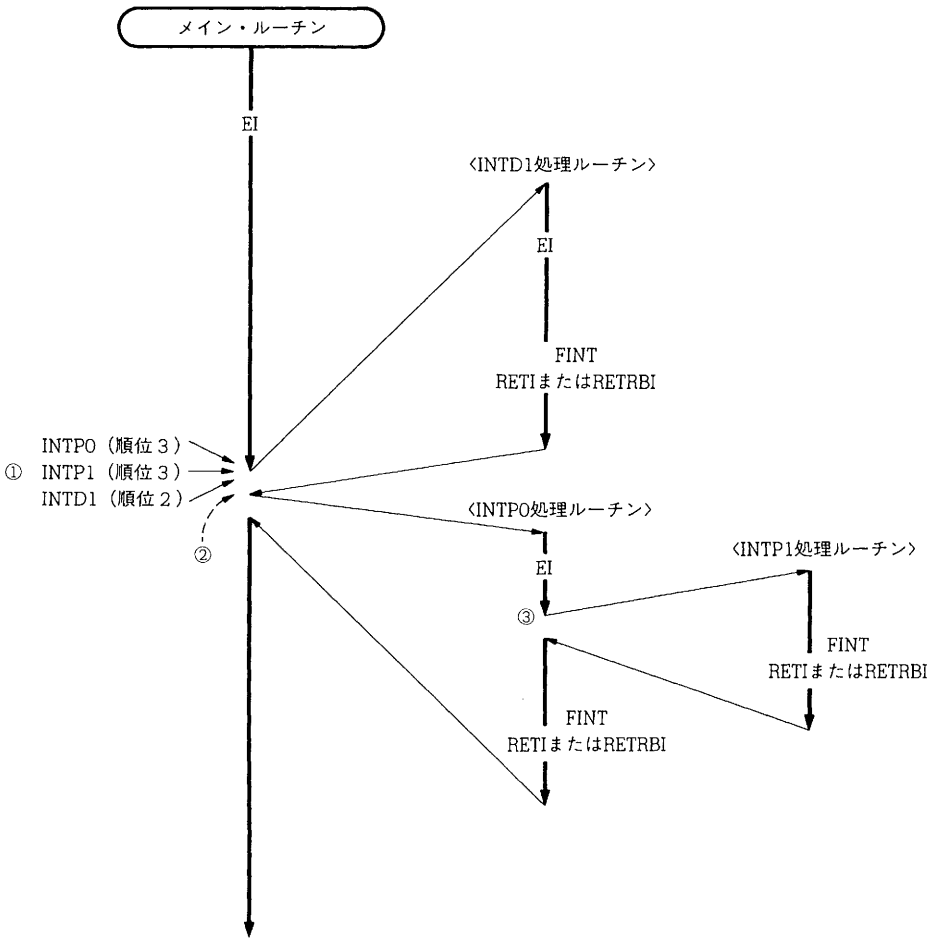
⑥　続けて発生した優先順位０に設定したINTP2は，INTSR0より高位のため受け付けられ，INTP2の処理を開始します。

⑦　INTP2処理が終了し，中断されていたINTSR0の処理も終了し復帰すると，INTST1が受け付けられます。

保守／廃止

(3) 多重割り込み処理制御を受ける割り込み……B (PRSL＝0の場合)

○PRSLビット＝0のとき，優先順位３のマスカブル割り込みは優先順位制御を行わず，同一レベルのネスティングが可能となります。



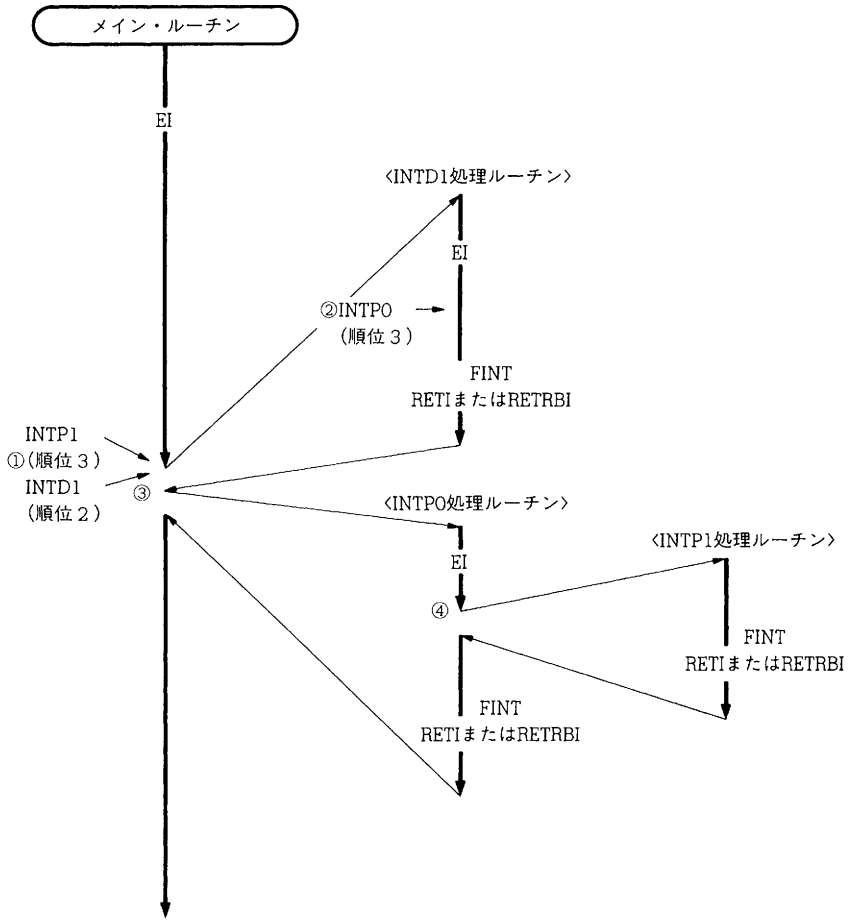
① 割り込み許可状態（EI）で優先順位３に設定したINTP0とINTP1，および優先順位２に設定したINTD1が同時に発生すると，プログラムで上位の優先順位に設定したINTD1が受け付けられINTD1処理を開始します。

② INTD1処理が終了すると，保留されていたINTP0とINTP1が同一優先順位に設定されているため，ディフォールト優先順位の高いINTP0が受け付けられます。

③ 割り込み許可状態（EI）にすると，保留されていたINTP1の処理を開始します。

保守／廃止

（4）多重割り込み処理制御を受ける割り込み……C（PRSL＝0の場合）



① 割り込み許可状態（EI）で優先順位３に設定したINTP1と優先順位２に設定したINTD1が同時に発生すると，プログラムで上位の優先順位に設定したINTD1が受け付けられ，INTD1処理を開始します。

② 割り込み許可状態（EI）で優先順位３に設定したINTP0が発生しますが，INTD1割り込みの処理中であり，INTD1より優先順位が低位なため受け付けられません。

③ INTD1処理が終了すると，保留されていたINTP0とINTP1が同一優先順位に設定されていますので，ディフォールト優先順位の高いINTP0が受け付けられます。

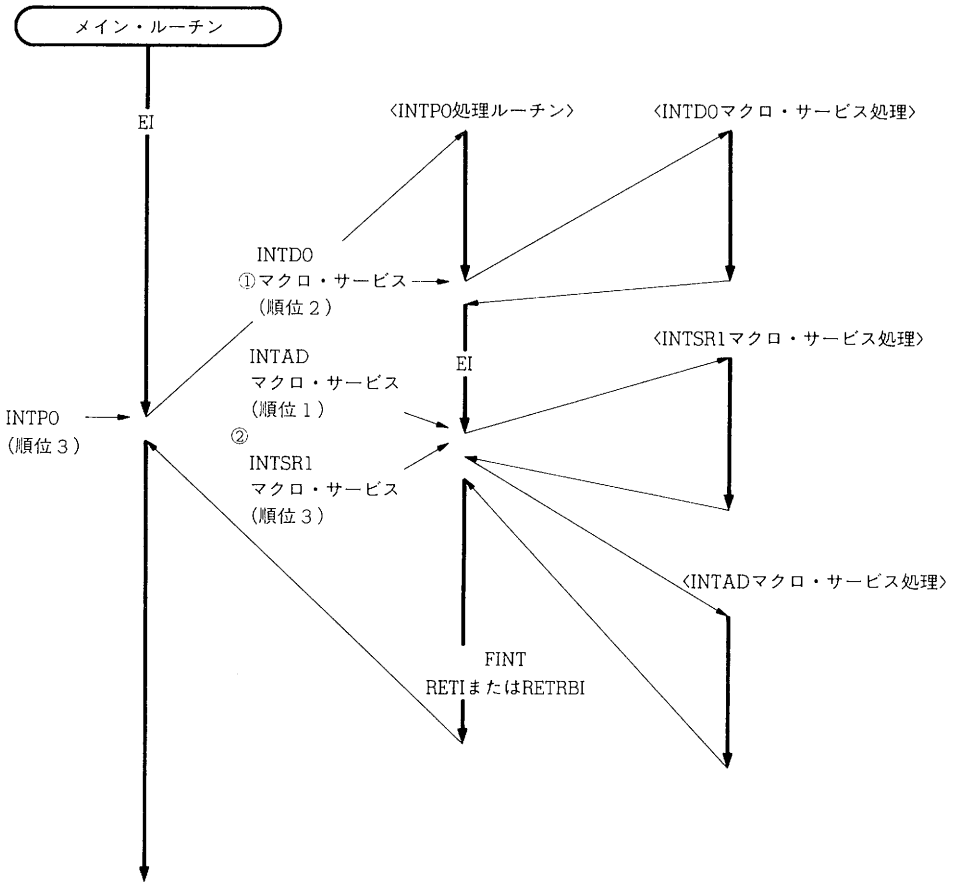
④ 割り込み許可状態（EI）にすると，保留されていたINTP1の処理を開始します。

保守／廃止

(5) マクロ・サービスによる割り込み……A

○割り込み許可（EI），割り込み禁止（DI）状態にかかわらず受け付けられます。

○PRフラグによる優先順位指定の制御を受けません。

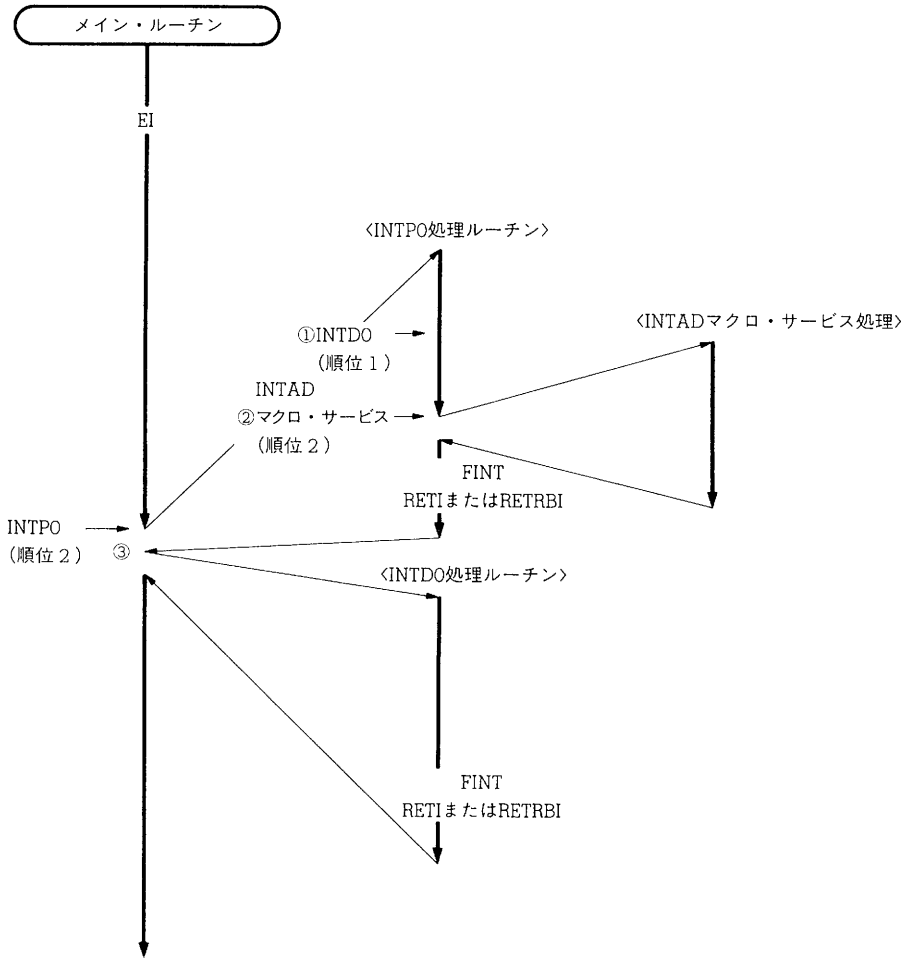


① 優先順位３に設定したINTP0処理中に，優先順位２に設定したINTD0のマクロ・サービスによる割り込みが発生すると，割り込み禁止状態（DI）であってもマクロ・サービス処理を行います。

② 続いて，割り込み許可状態（EI）にしたINTP0の処理中に，優先順位１と３にそれぞれ設定したINTADとINTSR1のマクロ・サービスによる割り込みが同時に発生すると，ディフォールト優先順位の高いINTSR1のマクロ・サービス処理の方から順番に受け付けられます。

保守／廃止

(6) マクロ・サービスによる割り込み……B



① 優先順位２に設定したINTP0処理中に，優先順位１に設定したINTD0の割り込み要求が発生しますが，割り込み禁止状態（DI）ですので保留されます。

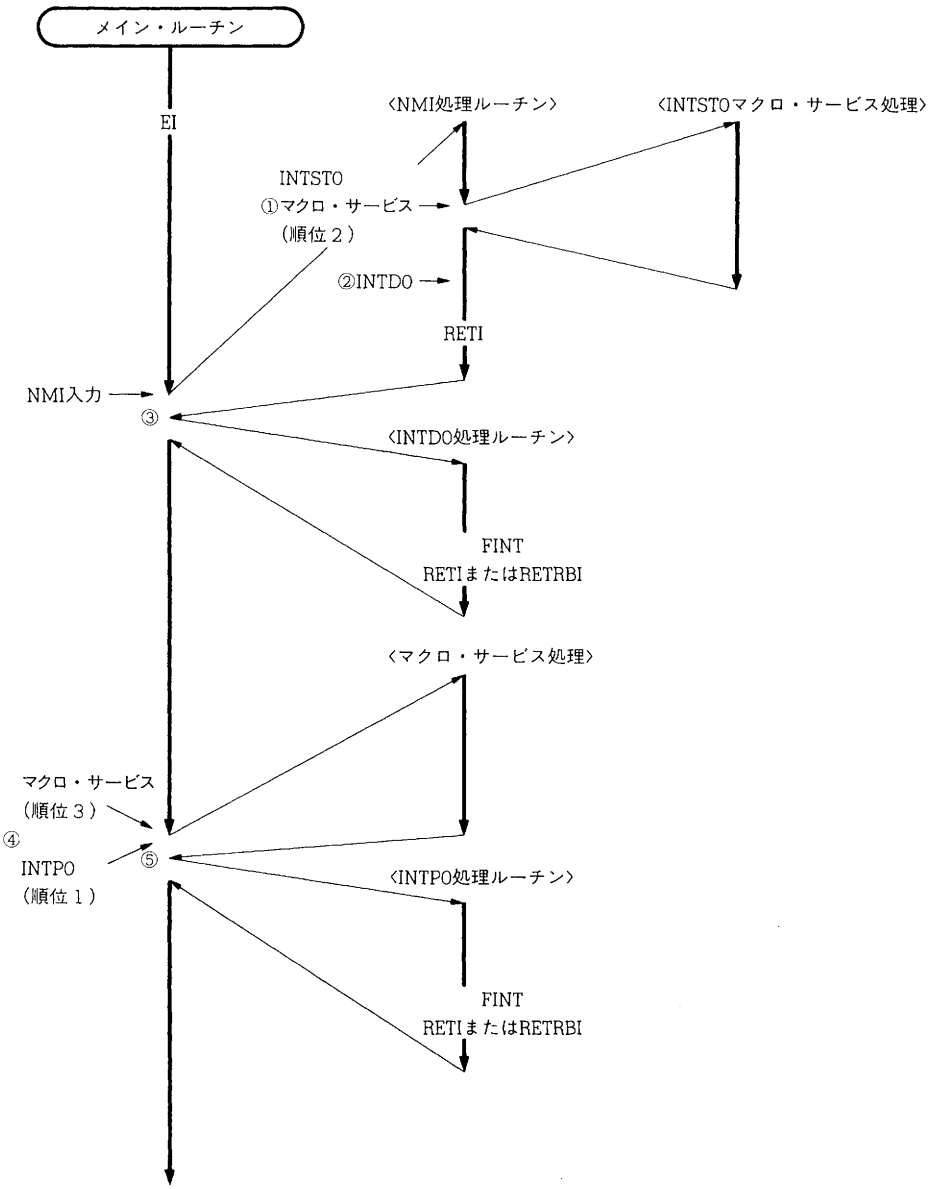
② 優先順位２に設定したINTADのマクロ・サービスによる割り込み要求が発生すると，処理中の割り込みおよび保留されている割り込みの優先順位，割り込み禁止状態（DI）にかかわらずマクロ・サービス処理を行います。

③ INTP0の処理が終了すると，保留されていたINTD0が受け付けられ，INTD0の処理を開始します。

保守／廃止

(7) マクロ・サービスによる割り込みのほか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ます。



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・サービス処理は行います。

② 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）が発生しても受け付けられません。

③ NMI処理が終了すると保留されていたINTDOが受け付けられます。

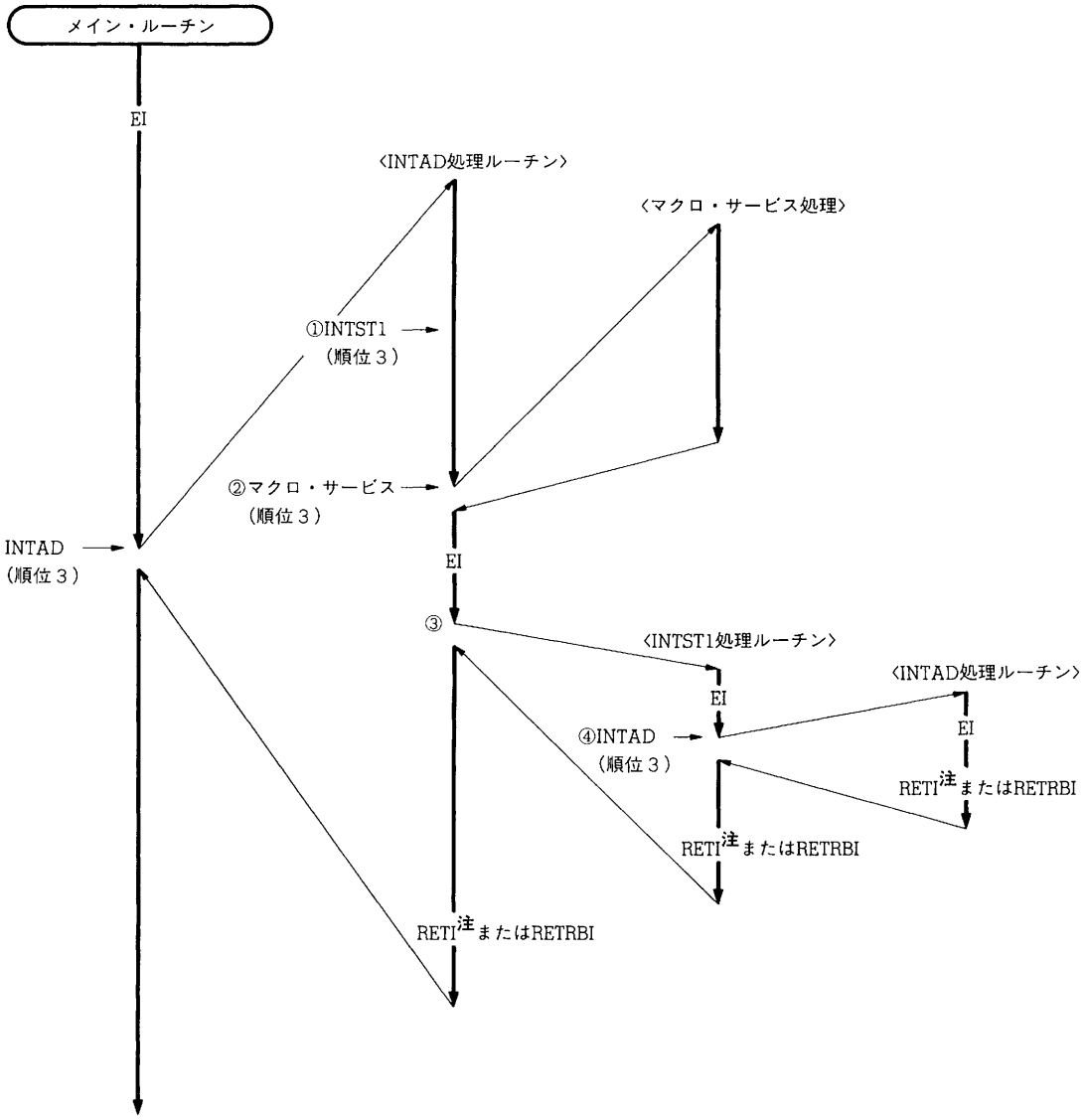
④ 割り込み許可状態（EI）で，マクロ・サービスによる割り込みと多重処理制御を受ける割り込み（INTPO）が同時に発生すると，マクロ・サービスによる割り込みが優先されて受け付けられ，INTPOは保留されます。

⑤ マクロ・サービス処理終了後，保留されていたINTPOが受け付けられます。

保守／廃止

(8) 各種割り込みの多重割り込み処理（PRSL＝0の場合）

○多重割り込み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



① 優先順位３に設定したINTADが受け付けられ，割り込み禁止状態（DI）になっていますので，優先順位３に設定したINTST1が発生しても受け付けられません。

② 続いて，優先順位３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います。

③ 割り込み許可状態（EI）にすると，保留されていたINTST1の処理を開始します。

④ INTST1処理中，割り込み許可状態（EI）にあれば，INTADの発生によってINTADの処理を開始します。同一割り込みの多重処理を行いたくない場合は，割り込み処理ルーチンで割り込み許可状態にする前にマスク・フラグをセット(1)します。

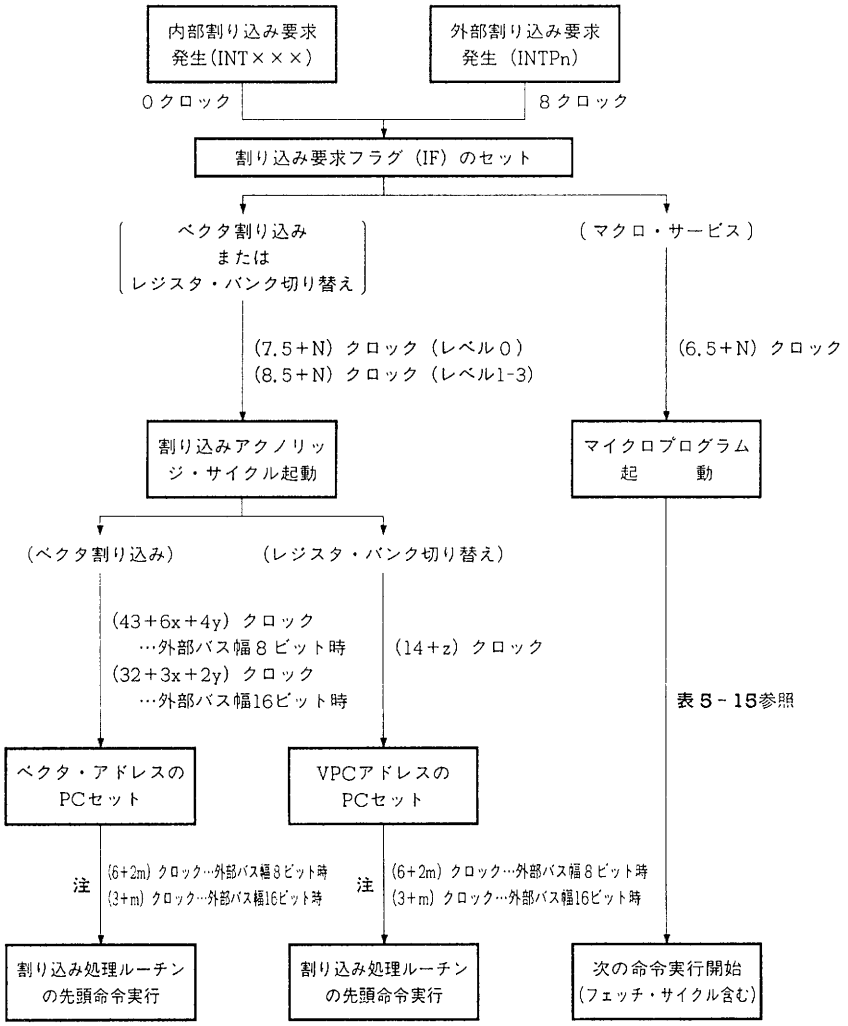
注 PRSL＝0の場合，優先順位３の割り込み発生は，割り込みコントローラのISPRレジスタに記憶されないため，FINT命令を実行する必要はありません（また，FINT命令の発行による動作上の問題も特にありません）。

保守／廃止

5.13　割り込み応答時間

ホールド要求，DMA転送要求，リフレッシュ要求などを考慮しない場合の割り込み応答時間を次に示します。

(1) マスカブル割り込み



N：この時点でCPUが実行中の命令の残り実行クロック数

x：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

y：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

z：ベクタ・テーブルからレジスタ・バンク番号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開始）

注　このクロック数は，割り込み処理ルーチンの先頭命令をフェッチするためのものです。

保守／廃止

表5－15　マクロ・サービス処理時間（単位：クロック）

マクロ・サービス機能			8ビット・バス幅	16ビット・バス幅
EVTCNT	INCモード		15	15
	DECモード		15	15
DTACMP			22	22
DTADIF	バイト処理		29＋t	29＋t
	ワード処理		33＋t	30＋t
BLKTRS	メモリ←SFR	バイト処理	29＋2t	29＋2t
		ワード処理	35＋4t	31＋2t
	SFR←メモリ	バイト処理	30＋2t	30＋2t
		ワード処理	36＋4t	31＋2t
BLKTRS-C（MSC≠0）	メモリ←SFR	バイト処理	40＋t	40＋t
		ワード処理	42＋t	40＋t
	SFR←メモリ	バイト処理	41＋t	41＋t
		ワード処理	49＋t	46＋t
BLKTRS-C（MSC＝0）	メモリ←SFR	バイト処理	30＋t	30＋t
		ワード処理	33＋t	30＋t
	SFR←メモリ	バイト処理	33＋t	33＋t
		ワード処理	36＋t	34＋t
RTOPTRN	A＝0		75＋4t	72＋2t
	A＝1		79＋4t	76＋2t
DTACMP-M	ステータス判定によりアボートした場合		27＋m	27＋m
	外部I/O→メモリの転送を行う場合		51＋m＋n	51＋m＋n
	メモリ→外部I/Oの転送を行う場合		49＋m＋p	49＋m＋p

t：転送対象となるメモリに対するウエイト数

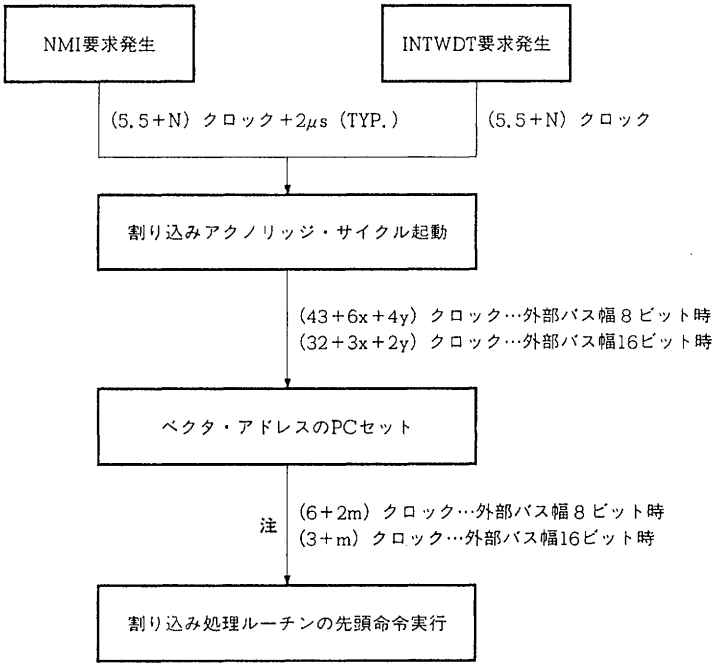
m：ステータス・データ（I/O）のリード・サイクルのウエイト数

n：転送データ（I/O）のリード・サイクルのウエイト数

P：転送データ（メモリ）のリード・サイクルのウエイト数

保守／廃止

(2) ノンマスカブル割り込み



N：この時点でCPUが実行中の命令の残り実行クロック数

x：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

y：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開始）

注　このクロック数は，割り込み処理ルーチンの先頭命令をフェッチするためのものです。

保守／廃止

5.14　注意事項

5.14.1　マクロ・サービスでSFRアクセスを行う場合

マクロ・サービス・チャネルのSFRアドレス指定ポインタには，SFRアドレスの下位16ビットを設定してください。また，上位7ビットにはすべて1を設定してください。

5.14.2　割り込みモード・コントロール・レジスタの操作

割り込みモード・コントロール・レジスタは，リセット後の初期状態から変更が不要な場合でも，必ず一度設定してください。これは，その後の誤った書き込み動作によるシステムの異常動作を防ぐためです。

なお，このレジスタの操作は，必ず割り込み禁止の状態（DI状態）で行ってください。

5.14.3　マクロ・サービス終了時の割り込み要求発生

マクロ・サービスが終了したときに発生するベクタ割り込み（またはレジスタ・バンク切り替え）処理中に再度同じ割り込み要求が発生すると，割り込み要求制御レジスタのIFフラグ＝1となります。この状態で割り込み要求制御レジスタのMS/$\overline{\text{INT}}$ビットを0→1に変更すると，ただちにマクロ・サービスが実行されます。マクロ・サービス終了後，再び割り込み処理の続きを実行します。

5.14.4　割り込みが発生しない場合

割り込みが発生するはずのタイミングで発生しない場合は，次に示す内容をまず確認してください。

(1) EI状態になっているか

EI命令を実行したか確認してください。PSWレジスタのIEフラグがセット(1)されていればEI状態です。

(2) 割り込みがマスクされてないか

割り込み要求制御レジスタ（IC××）のビット6（MSK）をリセット(0)して，マスクを解除してください。

(3) 割り込み要求が発生しているか

割り込み要求制御レジスタ（IC××）のビット7（IF）がセット(1)されていれば，割り込み要求は発生しています。このビットが0のときは割り込み要求が発生していません。ソフトウエアで割り込み要求をクリアしていないか確認してください。

(4) 優先順位が高い割り込みを処理中でないか

インサービス・プライオリティ・レジスタ（ISPR）に高い割り込み優先順位に相当するビットがセットされている可能性があります。この場合，V55PIはセットされているビットに対応する優先順位の割り込み処理を現在処理中であると認識し，それより低い優先順位の割り込みを受け付けません。

保守／廃止

ISPRレジスタのビットをリセットできるのは，FINT命令だけです。したがって，セットされている優先順位の割り込み処理ルーチン内でFINT命令を実行しているか確認してください。

ISPRレジスタの詳細については，**5.5.4　インサービス・プライオリティ・レジスタ（ISPR）**を参照してください。

5.14.5　FINT命令についての注意事項

FINT命令は，割り込み処理の最後のRETI命令の直前で必ず実行してください。

FINT命令は，ISPRレジスタでセットされているビットの最高優先順位に対応するビットをクリアする命令です。割り込み処理ルーチン内でFINT命令を実行しないと，その割り込みより低い優先順位の割り込みを受け付けられなくなります。

また，割り込み処理の途中でFINT命令を実行するとV55PIは優先順位の管理ができなくなり，処理中の割り込みより低い優先順位の割り込みも受け付けるようになります。したがって，割り込み処理の途中でFINT命令を実行する場合は，優先順位の管理をアプリケーションにより行ってください。

FINT命令の詳細については，**V55PI UM　命令編**を参照してください。

保守／廃止

第６章　DMA機能（DMAコントローラ）

V55PIは，内蔵周辺ハードウエア（シリアル・インタフェース，パラレル・インタフェース，タイマ），外部DMARQ端子，またはソフトウエア・トリガによるDMA（Direct Memory Access）要求に基づいて，メモリ-I/O間，またはメモリ-メモリ間でDMA転送を実行制御する２チャネルのDMAコントローラを内蔵しています。

DMAコントローラの各チャネルはさらにメイン・チャネルとサブチャネルから構成され，動作モードによってメイン・チャネルとサブチャネルを１つのチャネルとして使用するか，別々のチャネルとして使用するかが決定されます。別々のチャネルとして使用した場合，最大４チャネル分のDMA機能を構成することができます。

6.1　特　　徴

○２組の独立なDMAチャネル（最大４チャネル構成可能）

○４種類の転送モード

- シングルトランスファ・モード…１回のDMA要求に対して，１回のDMA転送サイクルを実行
- ディマンド・リリース・モード…DMA要求がアクティブの期間，連続してDMA転送サイクルを実行
- シングルステップ・モード…DMA要求の発生後，DMA転送サイクルとCPUバス・サイクルを交互に実行
- バースト・モード…１回のDMA要求に対して，指定回数のDMA転送サイクルを連続して実行

○５種類の動作モード

- インテリジェントDMAモード-1（リング・バッファ方式）…リング・バッファに対するDMA転送を制御
- インテリジェントDMAモード-2（カウンタ制御方式）…転送データを任意のバイト数ごとに分けて連続転送
- ネクスト・アドレス指定モード…異なる転送バッファの間での連続転送が可能
- ２チャネル動作モード…メイン・チャネルとサブチャネルをそれぞれ独立したチャネルとして使用
- メモリ-メモリ転送モード…１回のDMA転送サイクルにつき２回のバス・サイクルを起動し，メモリ-メモリ間の転送を実行

○３クロック／１バス・サイクル（ノー・ウエイト時）

○転送対象

- 外部I/O↔メモリ…1DMA転送サイクル／１バス・サイクル
- SFR（内部I/O）↔メモリ…1DMA転送サイクル／１バス・サイクル
- メモリ↔メモリ（メモリにはSFRを含む）…1DMA転送サイクル／２バス・サイクル

○バイト転送／ワード転送選択可能

保守／廃止

○転送アドレスのインクリメント／デクリメント／未更新を選択可能

○24ビットのDMAメモリ・アドレス・レジスタ（MAR0，MAR1）

○21ビットのターミナル・カウンタ（TC0，TC1）

○DMAコントローラの制御用端子（**2.2　端子機能の説明**参照）

- ●外部DMAリクエスト信号入力端子（DMARQ0，DMARQ1：ポートP80，P81端子兼用）
- ●DMAアクノリッジ信号出力端子（$\overline{\text{DMAAK0}}$，$\overline{\text{DMAAK1}}$）
- ●DMA転送終了信号出力端子（$\overline{\text{TCE0}}$，$\overline{\text{TCE1}}$）

6.2　基本動作

DMA機能により，CPUが介在することなくメモリ-メモリ，I/O（特殊機能レジスタ（SFR）を含む）-メモリのデータ転送を実現します。DMA機能を使用することにより，次の点で効果が得られます。

- ●ハードウエア制御により，データの転送レートが飛躍的に向上。
- ●割り込みによるソフトウエア処理に比べ，外部事象に対する応答性が向上。
- ●プログラム処理に対するオーバ・ヘッドは基本的にバス動作だけとなり，データ転送によるCPU処理性能の低下を最低限に抑えることが可能。
- ●単純なデータ転送処理にユーザ・プログラムが介在する必要がなくなり，効率的なプログラムのコーディングが可能。

転送動作としては，I/O（一部SFRを含む）-メモリ転送とメモリ（一部SFRを含む）-メモリ転送の2通りがあり，次のように転送方法（バス動作）が異なります。

表6－1　DMA転送対象と転送方法

転送動作	転送対象	転送方法	転送元選択方法	転送先選択方法
I/O-メモリ転送	外部I/O→外部メモリ	1回のバス・サイクルで転送元へのライト動作と転送先へのリード動作を実行	$\overline{\text{DMAAK}}$出力	メモリ・アドレス
	外部I/O←外部メモリ		メモリ・アドレス	$\overline{\text{DMAAK}}$出力
	SFR[注1]→外部メモリ		内部専用信号	メモリ・アドレス
	SFR[注1]←外部メモリ		メモリ・アドレス	内部専用信号
メモリ-メモリ転送[注2]	外部メモリ→外部メモリ	1回目のバス・サイクルで転送元へのリード動作を，2回目のバス・サイクルで転送先へのライト動作を実行	メモリ・アドレス	メモリ・アドレス
	SFR→外部メモリ		SFRアドレス	メモリ・アドレス
	SFR←外部メモリ		メモリ・アドレス	SFRアドレス

注1.　PAB，RxB0，RxB1，TxB0，TxB1だけ指定可能

2.　SFRを対象とするメモリ-メモリ転送を行う場合，ディマンド・リリース・モードは使用できません。これは，SFRアクセス時はバス・サイクルが発生せず転送数を制御できないためです。

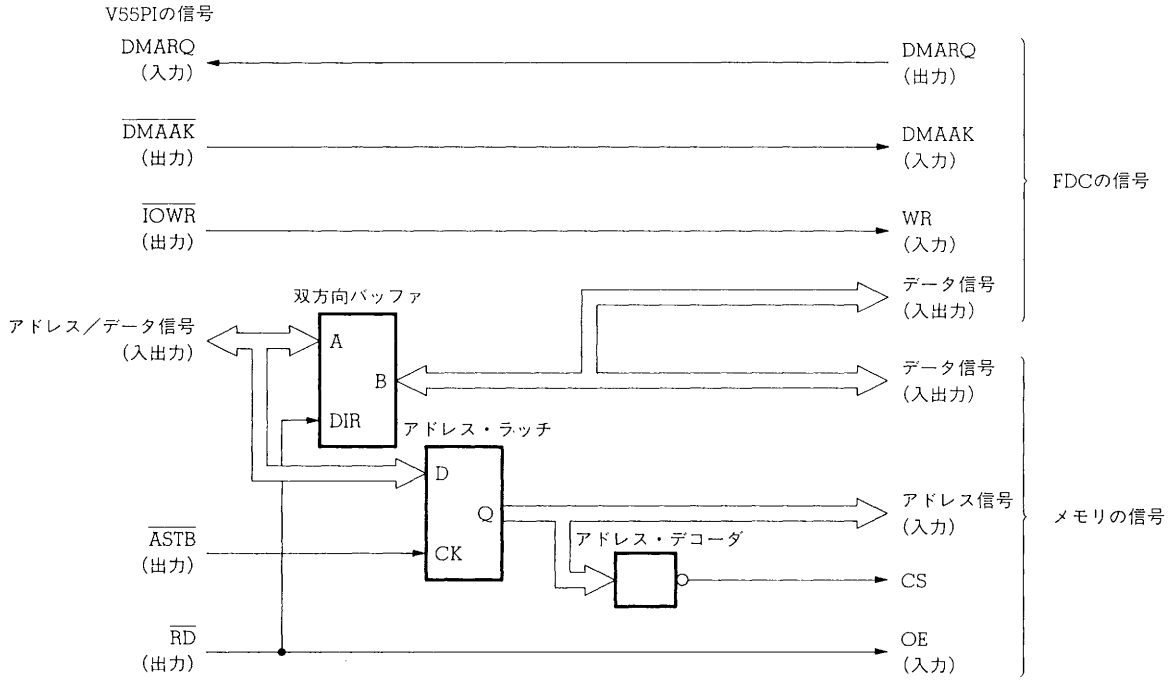
保守／廃止

外部データ・バス幅が16ビットのときに８ビットSFR-メモリ間のDMA転送を行う場合，有効となるデータ・バスは次のとおりです。

- 外部メモリのアドレスが偶数の場合：下位８ビット（D0-D7）
- 外部メモリのアドレスが奇数の場合：上位８ビット（D8-D15）

図６－１にμPD72070相当のFDC（フロッピィ・ディスク・コントローラ）を接続した場合の$\overline{\text{DMAAK}}$出力の接続例を，図６－２にそのタイミング例を示します。それぞれ外部メモリ-外部I/Oの転送でシングルトランスファ・モードの場合です。

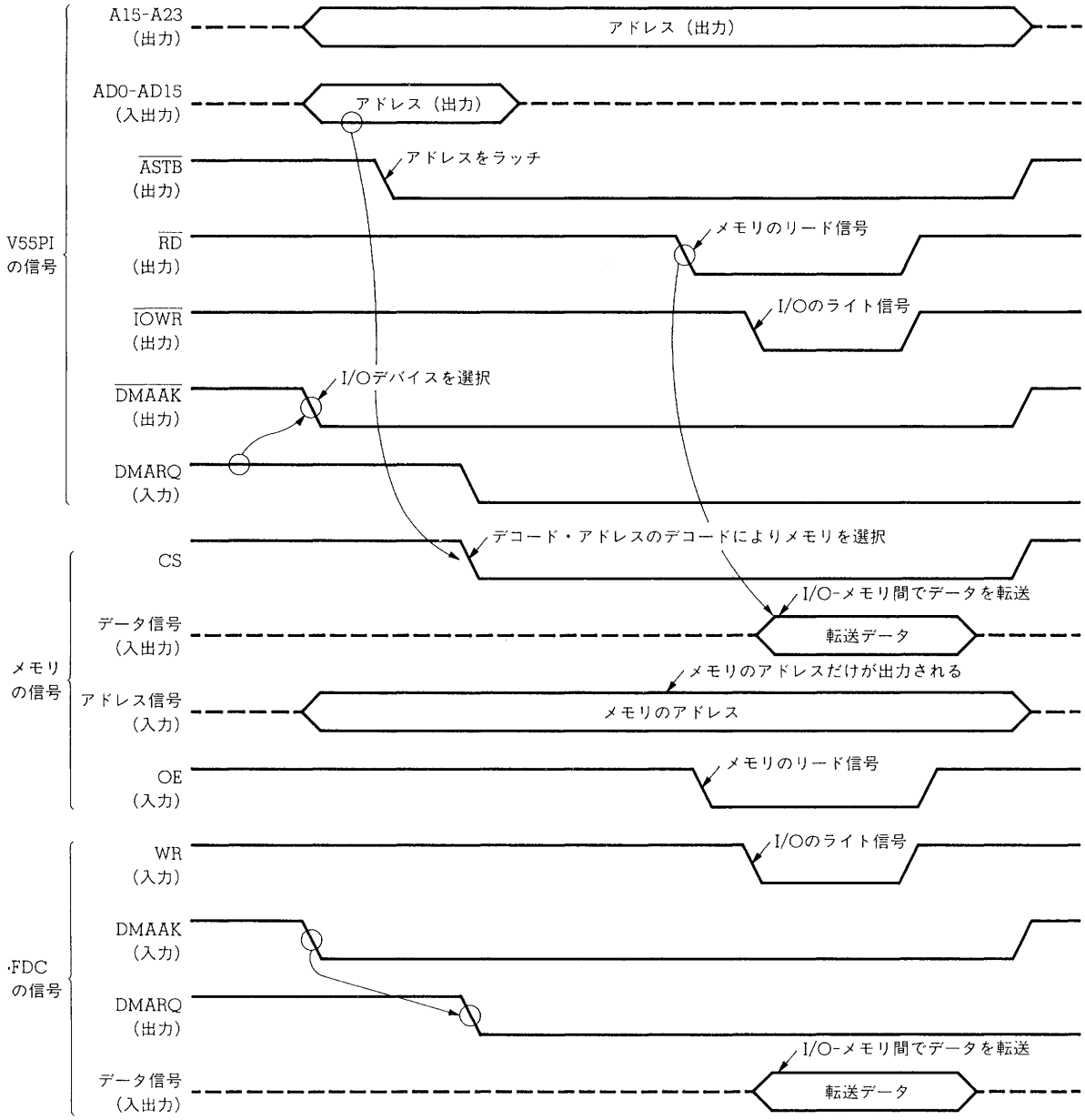
図６－１　$\overline{\text{DMAAK}}$出力の接続例（シングルトランスファ・モード）
（メモリ→I/O転送）



備考　この回路例では，DMA転送サイクルで使用する信号だけを示しています。
なお，実際の設計の際はタイミングなどを十分考慮してから行ってください。

保守／廃止

図６－２　DMA転送のタイミング例（シングルトランスファ・モード）
（メモリ→I/O転送）



備考　破線はハイ・インピーダンスを示します。

保守／廃止

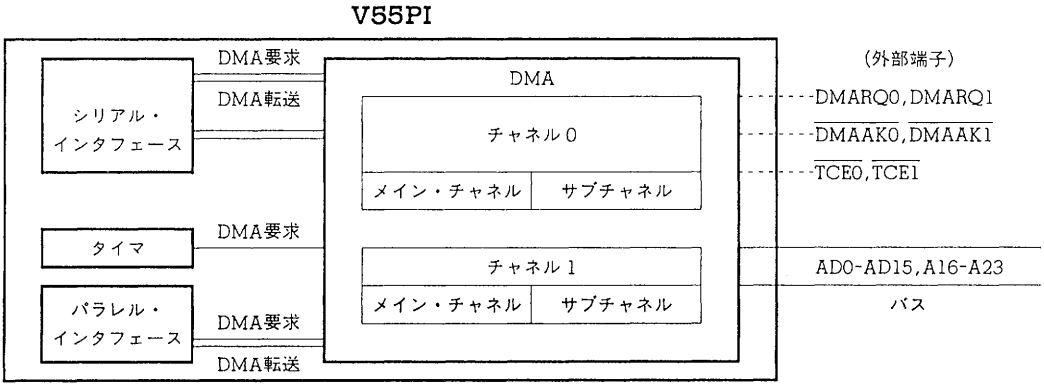
6.3　構　　成

V55PIのDMAコントローラは，２組の独立なDMAチャネルを持っています。従来のDMAコントローラで制御可能な外部メモリまたは外部I/Oと，外部メモリとのDMA転送のほか，内蔵周辺ハードウエアのSFRと外部メモリとのDMA転送が可能です。また，単なるデータ転送だけでなく，転送対象のメモリ・バッファに対するポインタ管理を自動的に行うインテリジェントDMAモードを備えています。

6.3.1　チャネル構成

DMAコントローラのチャネル構成を図６－３に示します。２組のDMAチャネルはそれぞれメイン・チャネルとサブチャネルから構成されています。２チャネル動作モード指定時には独立して動作可能なため，最大４つのDMAチャネルを使用することができます。ただしこの場合，転送対象と起動要因の組み合わせは限定されます（**表６－２　設定可能な転送対象と起動要因の組み合わせ**参照）。

図６－３　チャネル構成図



6.3.2　転送対象

DMAによるデータ転送の転送対象（転送元，転送先）として，次のものが選択できます。

DMAモード・レジスタ（DMAM0，DMAM1）のMDビットとTSビットにより，動作モードと起動要因の組み合わせで指定します。また，転送方向はDMACレジスタのDIRビットで指定します。

（1）外部メモリ（16Mバイト空間のうち，連続した２Mバイト（最大）を指定可能）

（2）外部I/O

（3）SFR

- パラレル・インタフェース・バッファ（PAB）
- UART送信バッファ（TxB0，TxB1）
- 受信バッファ（RxB0，RxB1）
- ポート（P0-P8），ポート７バッファ（P7L，P7H）

保守／廃止

- リアルタイム出力ポート（RTP）
- タイマ・コンペア・レジスタ（CM00，CM01，CM10，CM11，CM20，CM21，CM22，CM23，CM30，CM31）
- PWMレジスタ（PWM）
- A/Dコンバージョン・リザルト・レジスタ（ADCR0-ADCR3）

注意　**DMA機能をマクロ・サービスと同時に使用する場合，双方の処理で同一のメモリ・ブロックまたは同一のSFRを転送対象に指定しないでください。**

6.3.3　起動要因

DMA転送サイクルを起動する要因として，次のものが選択できます。これらからのDMA転送要求に応じてDMA転送が実行されます。DMAモード・レジスタ（DMAM0，DMAM1）のTSビットにより転送対象との組み合わせで指定します。

（1）外部端子入力（DMARQ0，DMARQ1）

（2）内部割り込み要求

- パラレル・インタフェース割り込み（INTPAI）
- UART送信割り込み（INTST0，INTST1）
- UART受信割り込み（INTSR0，INTSR1）

（3）タイマ／カウンタのコンペア・レジスタ（CM23）の一致信号

また，DMAMレジスタのTDMAビットを操作することにより，ソフトウエアによってDMA転送サイクルを起動することもできます。

保守／廃止

表６－２　設定可能な転送対象と起動要因の組み合わせ（1/2）

（チャネル０）

対応転送モード（転送対象）	TS1	TS0	メイン・チャネル		サブチャネル[注1]	
			起動要因	転送対象I/O	起動要因	転送対象I/O
I/O-メモリ転送	0	0	パラレル・インタフェース INTPAI	データ入出力バッファ (PAB)	シリアル送信完了 INTST0	送信バッファ (TxB0)
			I/O ↔ メモリ		メモリ → I/O	
			DIR＝0，または１[注2]		メイン・チャネルと逆方向	
	0	1	DMARQ0	外部I/O ($\overline{DMAAK0}$)	DMARQ1	外部I/O ($\overline{DMAAK1}$)
			I/O ↔ メモリ		I/O ↔ メモリ	
			DIR＝0，または１		メイン・チャネルと同方向	
	1	0	内蔵タイマ CM23一致信号	外部I/O ($\overline{DMAAK0}$)	DMARQ1	外部I/O ($\overline{DMAAK1}$)
			I/O ↔ メモリ		I/O ↔ メモリ	
			DIR＝0，または１		メイン・チャネルと逆方向	
	1	1	シリアル送信完了 INTST0	送信バッファ (TxB0)	シリアル送信完了 INTST1	送信バッファ (TxB1)
			メモリ → I/O		メモリ → I/O	
			DIR＝１（固定）		メイン・チャネルと同方向	
メモリ-メモリ転送	0	0	設定禁止			
	0	1	DMARQ0	—		
	1	0	内蔵タイマ CM23一致信号	—		
	1	1	設定禁止			

注1．サブチャネルは，２チャネル動作モード時のみ有効です。

2．DIRへの設定値を１とし（メモリ → I/O），パラレル・インタフェースを起動要因に選択した場合，サブチャネル側は使用できません。

保守／廃止

表６－２　設定可能な転送対象と起動要因の組み合わせ（2/2）

（チャネル１）

対応転送モード（転送対象）	TS1	TS0	メイン・チャネル		サブチャネル[注1]	
			起動要因	転送対象I/O	起動要因	転送対象I/O
I/O-メモリ転送	0	0	シリアル受信完了 INTSR1	受信バッファ (RxB1)	シリアル送信完了 INTST0	送信バッファ (TxB0)
			I/O → メモリ		メモリ → I/O	
			DIR＝0		メイン・チャネルと逆方向	
	0	1	内蔵タイマ CM23一致信号	外部I/O ($\overline{\text{DMAAK1}}$)	DMARQ0	外部I/O ($\overline{\text{DMAAK0}}$)
			I/O ↔ メモリ		I/O ↔ メモリ	
			DIR＝0，または１		メイン・チャネルと同方向	
	1	0	DMARQ1	外部I/O ($\overline{\text{DMAAK1}}$)	DMARQ0	外部I/O ($\overline{\text{DMAAK0}}$)
			I/O ↔ メモリ		I/O ↔ メモリ	
			DIR＝0，または１		メイン・チャネルと逆方向	
	1	1	パラレル・インタフェース INTPAI	データ入出力バッファ (PAB)	シリアル受信完了 INTSR0	受信バッファ (RxB0)
			I/O ↔ メモリ		I/O → メモリ	
			DIR＝0，または１[注2]		メイン・チャネルと同方向	
メモリ-メモリ転送	0	0	設定禁止			
	0	1	内蔵タイマ CM23一致信号	—		
	1	0	DMARQ1	—		
	1	1	設定禁止			

注1. サブチャネルは，２チャネル動作モード時のみ有効です。

2. DIRへの設定値を１とし（メモリ → I/O），パラレル・インタフェースを起動要因に選択した場合，サブチャネル側は使用できません。

保守／廃止

6.3.4　転送モード

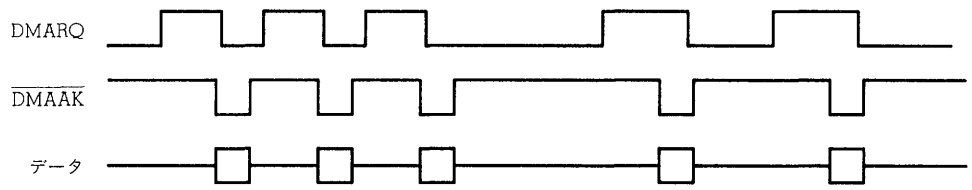
各DMAチャネルに対して，４種類の転送モードが選択できます。転送モードごとに転送サイクルの起動方法が異なります。転送モードはDMACレジスタのTMDビットで指定します。

(1) シングルトランスファ・モード

DMA転送要求が発生するごとに，１回のDMA転送サイクルを実行します。DMA転送サイクル終了後，ただちにバスの使用権を解放するため，CPUは命令実行を継続することができます。DMA転送要求として，DMARQ入力の立ち上がりエッジまたは内蔵周辺ハードウエアからの要求信号が選択できます。

注意　複数のチャネルから同時にDMA転送要求が発生した場合，すべてのチャネルの要求が処理されるまで，バス使用権はDMAコントローラが持っているままとなります。ただし，ホールド・モードとリフレッシュ・サイクルによるバスの解放を除きます。

図６－４　シングルトランスファ・モード転送タイミング

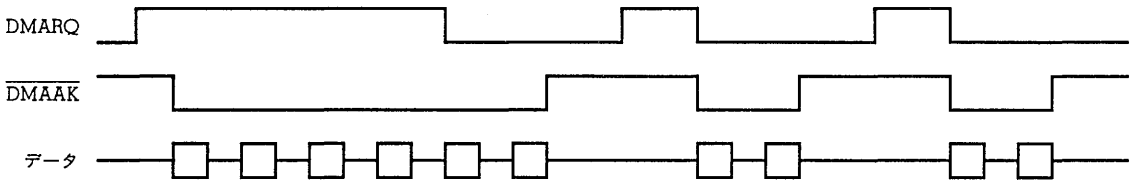


(2) ディマンド・リリース・モード

DMARQ入力がハイ・レベルの期間中，連続的にDMA転送サイクルを実行します。その間，CPUバス・サイクルは保留されるため，命令実行が停止します。

★

図６－５　ディマンド・リリース・モード転送タイミング

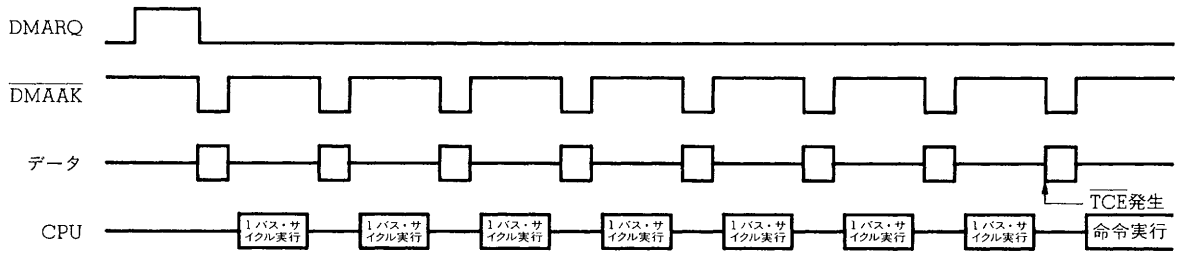


(3) シングルステップ・モード

DMA転送要求発生後，１回のDMA転送サイクルと１回のCPUバス・サイクルを指定回数だけ交互に繰り返します。DMA転送要求として，DMARQ入力の立ち上がりエッジまたは，内蔵周辺ハードウエアからの要求信号が選択できます。

保守／廃止

図６－６　シングルステップ・モード転送タイミング



(4) バースト・モード

DMA転送要求発生後，指定回数のDMA転送サイクルを連続して実行します。その間，CPUバス・サイクルは保留されるため，命令実行が停止します。DMA転送要求として，DMARQ入力の立ち上がりエッジまたは，内蔵周辺ハードウエアからの要求信号が選択できます。

図６－７　バースト・モード転送タイミング

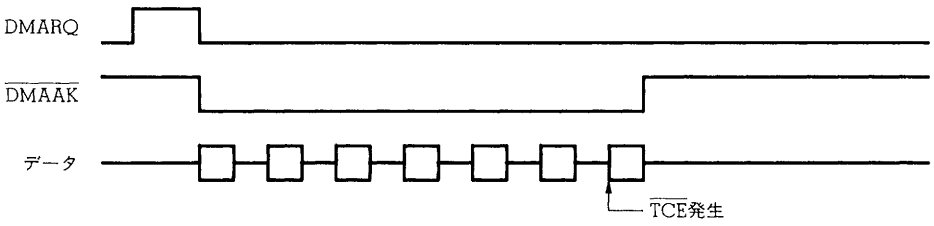


表６－３　転送モード

転送モード	DMA起動要因			停止方法	割り込み
	内蔵周辺	ソフトウエア・トリガ	DMARQ端子		
シングルトランスファ・モード	○	○	○	DMAMnレジスタのEDMAビットのリセット	受け付ける
ディマンド・リリース・モード	×	×	○	転送中はDMARQ端子をロウ・レベルにすると停止。DMAMnレジスタのEDMAビットのリセット	転送中は受け付けない ほかの場合は受け付ける
シングルステップ・モード	○注	○	○	DMAMnレジスタのEDMAビットのリセット	受け付ける
バースト・モード	○注	○	○	なし（転送中は停止不可）	受け付けない

注　DMA起動要因が内蔵タイマ割り込みで，転送I/Oの指定が外部の場合のみ転送可能となります。

保守／廃止

6.3.5　動作モード

転送対象のメモリ・ブロックに対するポインタ制御の違いにより，次の5種類の動作モードがあります。DMAMレジスタのMDビットで指定します。

- インテリジェントDMAモード-1
- インテリジェントDMAモード-2
- ネクスト・アドレス指定モード
- 2チャネル動作モード
- メモリ-メモリ転送モード

各動作モードにおける使用可能な転送モードとDMA割り込み（INTD）発生タイミングを表6-4に示します。

表6-4　動作モード一覧

動作モード		転送対象	使用可能な転送モード[注1]				DPTC	INTD
			①	②	③	④	(IOP)	発生条件
インテリジェントDMAモード-1（リング・バッファ方式）		I/O(SFR[注2]) → メモリ	可	可	不可	不可	使用可	A
インテリジェントDMAモード-2（カウンタ制御方式）		メモリ → I/O(SFR[注2])	不可	不可	可	可	使用可	B
ネクスト・アドレス指定モード		I/O(SFR[注2]) - メモリ	可	可	不可	不可	不可	B
2チャネル動作モード	(終了時停止)	I/O(SFR[注2]) - メモリ	可	可	可	可	不可	B
	(繰り返し)	I/O(SFR[注2]) - メモリ	可	可	可	不可	不可	B
メモリ-メモリ転送[注3]モード	(終了時停止)	メモリ(SFR) - メモリ(SFR)	可	不可	可	可	不可	B
	(繰り返し)	メモリ(SFR) - メモリ(SFR)	可	不可	可	不可	不可	B

注1. 転送モード（それぞれの動作については，6.3.4　転送モード参照）

① シングルトランスファ・モード

② ディマンド・リリース・モード

③ シングルステップ・モード

④ バースト・モード

2. PAB，RxB0，RxB1，TxB0，TxB1だけ指定可能

3. SFRを対象とするメモリ-メモリ転送を行う場合，ディマンド・リリース・モードは使用できません。これは，SFRアクセス時はバス・サイクルが発生せず転送数を制御できないためです。

備考　INTD発生条件

A：DMA転送実行により，DMAアップ・ダウン・カウンタ（UDC）とDMAコンペア・レジスタ（DCM）の値が一致したとき。およびUDCとDCMの値が一致した状態から，IN命令またはINM命令により入出力リード・ライト・ポインタ（IOP）を利用してデータを読み出したとき。

B：TCのデクリメントによりボローが発生したとき（$\overline{\text{TCE}}$信号の発生）。

保守／廃止

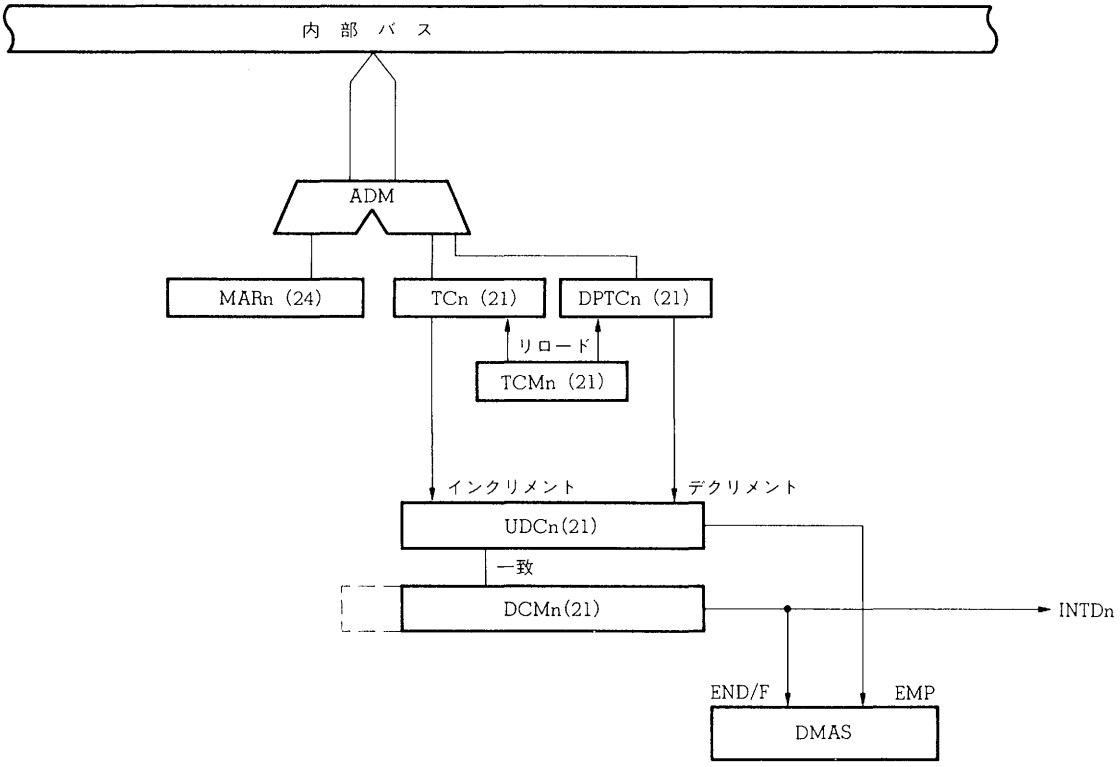
各動作モードにおけるレジスタの設定と動作の説明を以降に示します。レジスタの詳細については，**6.6　制御レジスタ，6.7　DMAサービス・チャネル**を参照してください。

（1）インテリジェントDMAモード－1（リング・バッファ方式）

DMA転送先メモリの指定領域が論理的にリング構造（リング・バッファ）になるよう，繰り返しI/OからメモリへDMA転送を行うモードです。

図６－８にインテリジェントDMAモード－1の機能ブロック図を示します。

図６－８　インテリジェントDMAモード－1（リング・バッファ方式）



保守／廃止

【レジスタの設定】

レ　ジ　ス　タ　名	設　定　内　容
MARn（DMAメモリ・アドレス・レジスタ）	DMA転送バッファ・メモリの終了アドレス（ベース・アドレス）
TCMn（ターミナル・カウンタ・モジュロ・レジスタ） TCn（ターミナル・カウンタ） DPTCn（DMAリード・ライト・ポインタ）	バッファ・サイズ－1（バイト転送のとき） バッファ・サイズ－2（ワード転送のとき）注1
UDCn（DMAアップ・ダウン・カウンタ）	有効データ数を示します
DCMn（DMAコンペア・レジスタ）	バッファ・サイズ－バッファ・マージン注2
DMAC（DMAコントロール・レジスタ）のPDビット	アドレスの更新方向（6.6.2　参照）

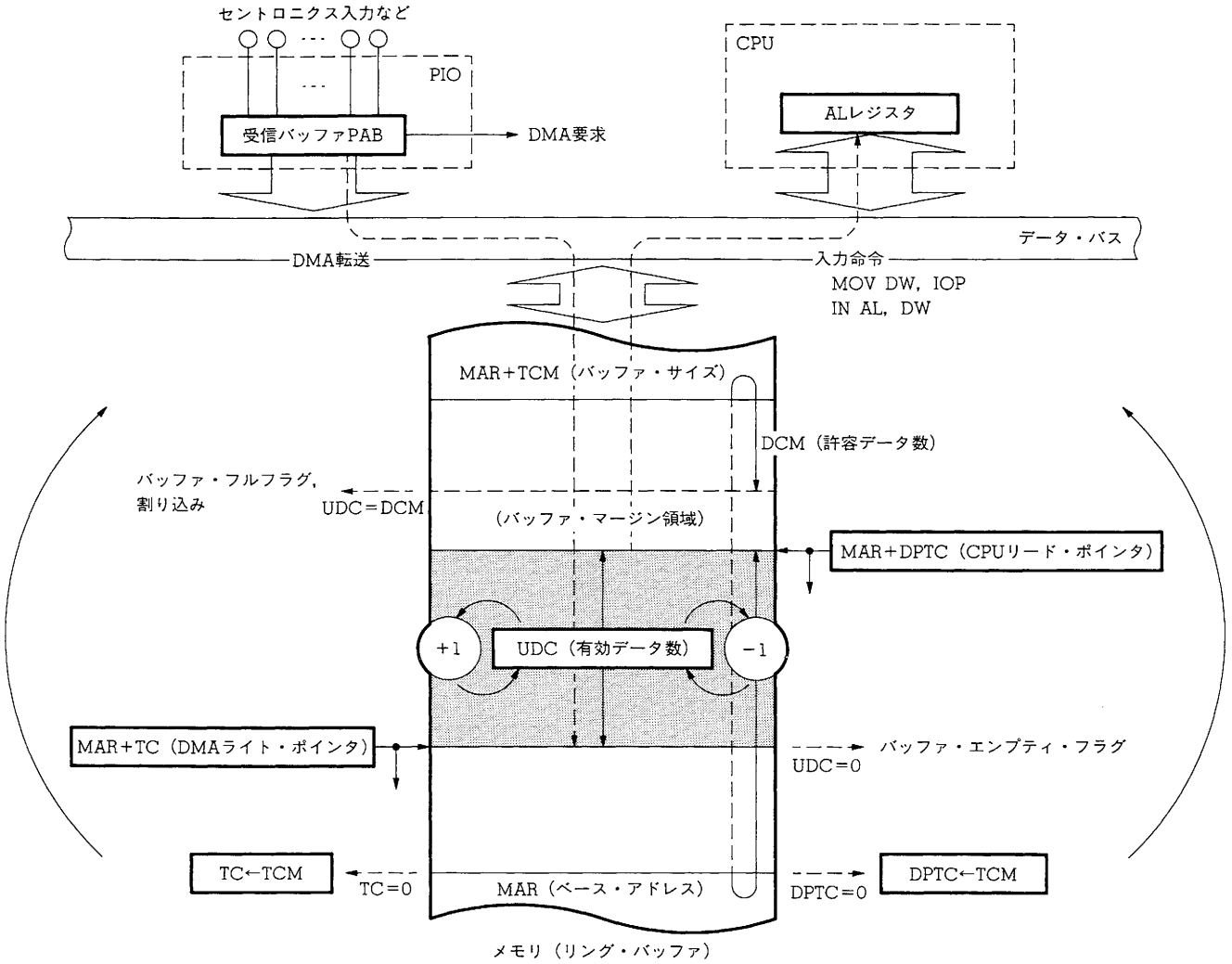
注1．ワード転送ではバッファ・サイズを偶数に設定してください。

2．バッファ・マージンは，DMA割り込み要求（INTD）が発生してからDMAを停止させる（ソフトウエアでDMAモード・レジスタ（DMAM）のEDMA＝0に設定）までの間に転送されるデータを格納するためのマージン領域です。

図６－９にデクリメント・モード時のバッファ・メモリのイメージを示します。

保守／廃止

図6－9 デクリメント・モード時のバッファ・メモリ（インテリジェントDMAモード－1）



備考 ▒ は，データが格納されている領域を示します。

保守／廃止

【動作の説明】

（a）DMA転送

① DMA要求の発生

② I/Oからメモリへのデータ転送

転送先のメモリ・アドレスは次の値になります。

- インクリメント・モードのとき：MAR－TC
- デクリメント・モードのとき　：MAR＋TC

③ TCを－1（ワード転送の場合は－2）する。

UDCを＋1（ワード転送の場合は＋2）する。

④ DMA転送終了

〈TCボロー発生時（リング・バッファの境界を越えた場合）〉

TCにTCMの値が自動的にリロードされます。その結果アドレスは，MAR－TC（インクリメント・モード時）またはMAR＋TC（デクリメント・モード時）が指すリング・バッファの先頭アドレスとなります。

（b）転送データ読み出し

① 入出力リード・ライト・ポインタ（IOPn）からのデータ読み出し（IN命令実行）

IN命令は，DMA転送単位と同じ単位で行ってください。転送先のメモリ・アドレスは次の値になります。

- インクリメント・モードのとき：MAR－DPTC
- デクリメント・モードのとき　：MAR＋DPTC

② DPTC，UDCの値をそれぞれ－1（ワード転送の場合は－2）する。

IOPとDPTCの対応表を表６－５に示します。外部８ビット・バス使用時は必ずバイト単位で読み出してください。

〈DPTCボロー発生時（リング・バッファの境界を越えた場合）〉

DPTCにTCMの値が自動的にリロードされます。その結果アドレスは，MAR－DPTC（インクリメント・モード時）またはMAR＋DPTC（デクリメント・モード時）が指すリング・バッファの先頭アドレスとなります。

保守／廃止

表６－５　IOPnとDPTCnの対応

入出力リード・ライト・ポインタ（I/Oアドレス）	DMAリード・ライト・ポインタ（SFRアドレス）
IOP0（FF94H）注	DPTC0（0FFF94H）
IOP1（FFB4H）注	DPTC1（0FFFB4H）

注　IOPnおよび対応するDPTCnを使用して，メモリに対するアクセスを行うことができます。IOP0，IOP1はデータ・メモリ空間（16 Mバイト）に対するI/O参照用のポインタとして機能します。

［使用例］　23000H番地のメモリからデータを読み出す場合の命令シーケンス

```
MOV     MAR0, 2000H
MOV     MAR0H, 02H
MOV     DPTC0, 1000H
IN      AL, IOP0
```

（c）有効データ数の管理

UDCの値をチェックすることにより，まだ読み出されていない有効データ数を知ることができます。

UDCは，１回のDMA転送が実行されるごとに，＋1（ワード転送の場合は＋2）されます。また，IN命令によるIOPを使用したデータの読み出しごとに，－1（ワード転送の場合は－2）されます。

〈UDCとDCMの値が一致した（リング・バッファがフル状態になった）とき〉

バッファ・フル・フラグ（END/F）をセット(1)します。同時にDMA割り込み要求（INTD）を発生します。バッファ・フルの状態からIN命令によるIOP領域に対するデータの読み出しが行われた場合，UDCは－1され，END/Fフラグをリセット（0）し，同時にDMA割り込み要求（INTD）を発生します。

〈UDCからボローが発生した（リング・バッファ内に有効データがなくなった）とき〉

バッファ・エンプティ・フラグ（EMP）をセット(1)します。割り込み要求（INTD）は発生しません。

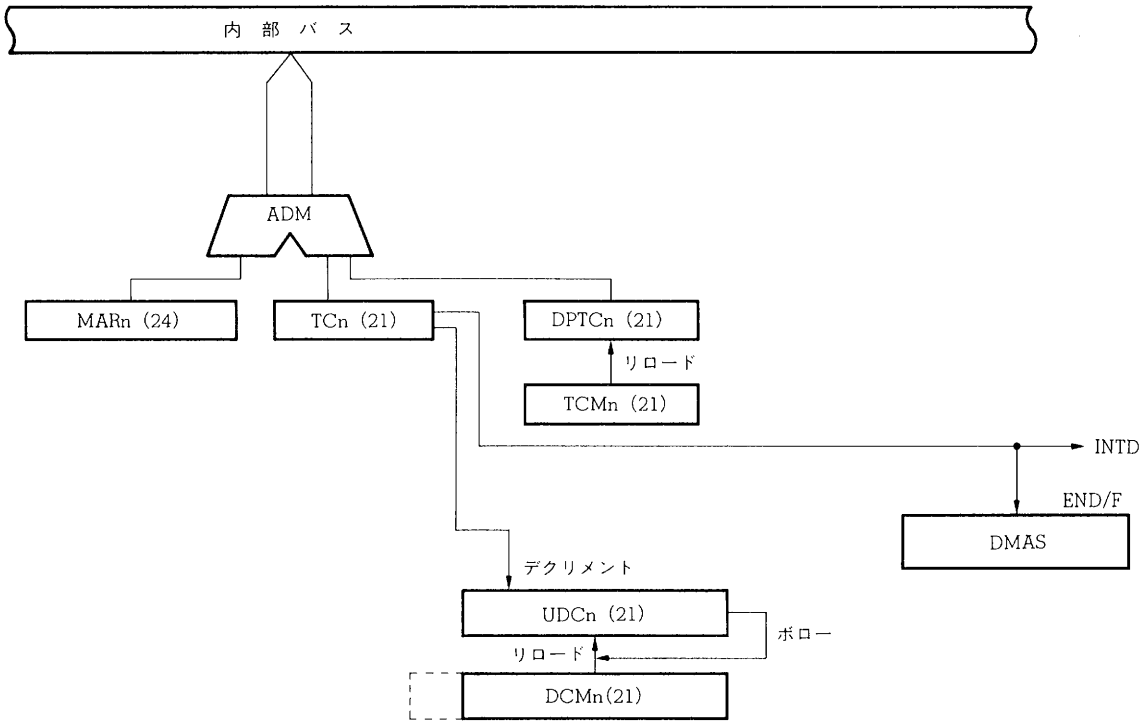
保守／廃止

(2) インテリジェントDMAモード－2（カウンタ制御方式）

DMA転送元メモリの指定領域に展開されたデータを，１回のDMA起動要求ごとに複数回のDMA転送を行い転送するモードです。

図６－10にインテリジェントDMAモード－2の機能ブロック図を示します。

図６－10　インテリジェントDMAモード－2（カウンタ制御方式）



保守／廃止

【レジスタの設定】

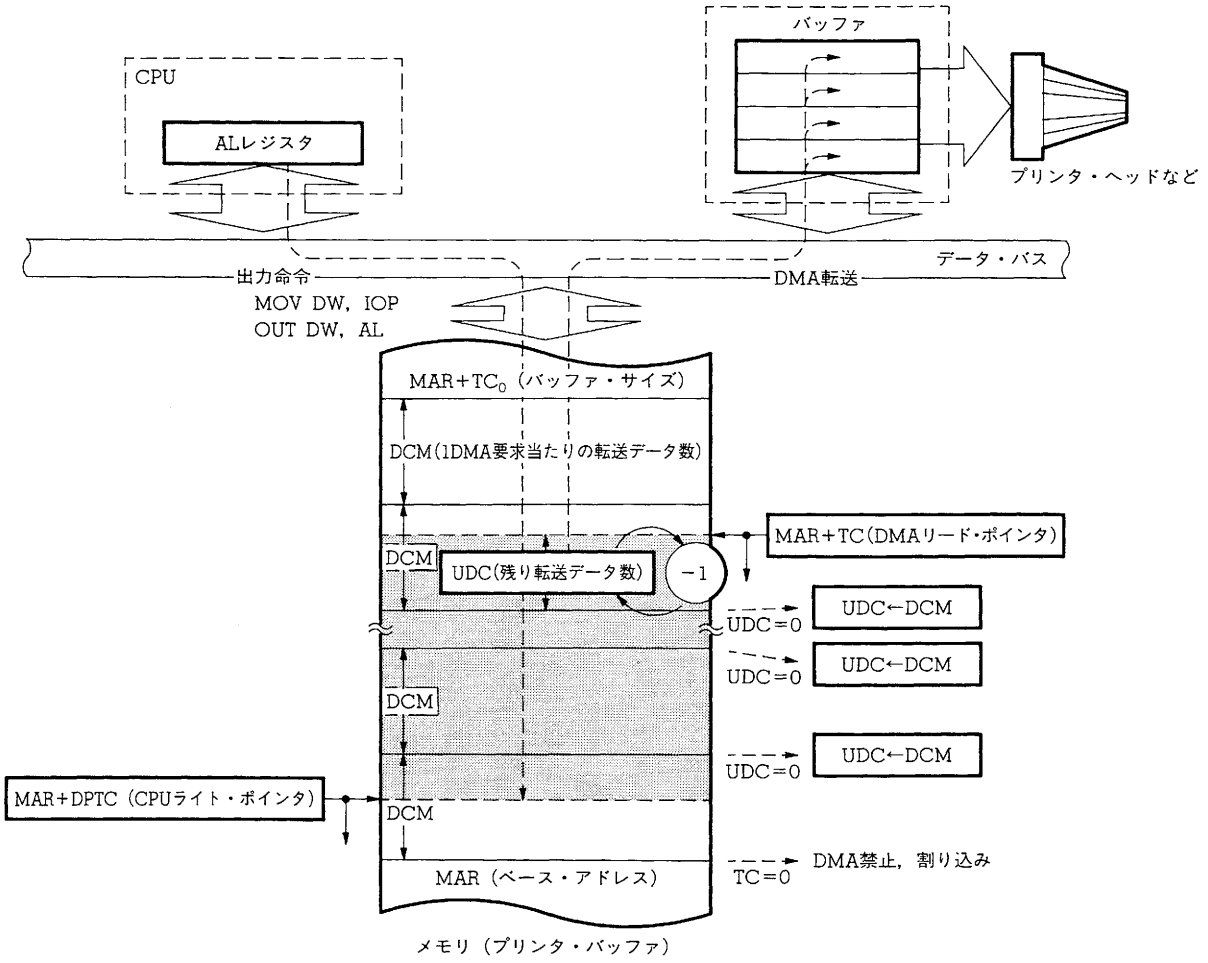
レ　ジ　ス　タ　名	設　定　内　容
MARn（DMAメモリ・アドレス・レジスタ）	DMA転送バッファ・メモリの終了アドレス（ベース・アドレス）
TCMn（ターミナル・カウンタ・モジュロ・レジスタ） TCn（ターミナル・カウンタ） DPTCn（DMAリード・ライト・ポインタ）	バッファ・サイズ－1（バイト転送のとき） バッファ・サイズ－2（ワード転送のとき）注
DCMn（DMAコンペア・レジスタ） UDCn（DMAアップ・ダウン・カウンタ）	1回のDMA起動要求により実行するDMA転送回数－1（バイト転送のとき） 1回のDMA起動要求により実行するDMA転送回数－2（ワード転送のとき）
DMAC（DMAコントロール・レジスタ）のPDビット	アドレスの更新方向（**6.6.2**参照）

注　ワード転送ではバッファ・サイズを偶数に設定してください。

図６－11にデクリメント・モード時のバッファア・メモリのイメージを示します。

保守／廃止

図６－１１　デクリメント・モード時のバッファ・メモリ（インテリジェントDMAモード－２）



備考　　は，データが格納されている領域を示します。

保守／廃止

【動作の説明】

(a) 転送バッファ・メモリへの転送データの書き込み

① I/O空間に割り付けられた入出力リード・ライト・ポインタ（IOPn）に対するデータの書き込み（OUT命令実行）

OUT命令は，DMA転送単位と同じ単位で行ってください。転送先のメモリ・アドレスは次の値になります。

- インクリメント・モードのとき：MAR－DPTC
- デクリメント・モードのとき　：MAR＋DPTC

② DPTCの値を－1（ワード転送の場合は－2）する。

IOPとDPTCの対応表を表６-５に示します。外部８ビット・バス使用時は必ずバイト単位でアクセスしてください。

〈DPTCボロー発生時（バッファの境界を越えた場合）〉

DPTCにTCMの値が自動的にリロードされます。その結果アドレスは，MAR－DPTC（インクリメント・モード時）またはMAR＋DPTC（デクリメント・モード時）が指すバッファの先頭アドレスとなります。

(b) DMA転送

① DMA要求の発生

② メモリからI/Oへのデータ転送

転送元のメモリ・アドレスは次の値になります。

- インクリメント・モードのとき：MAR－TC
- デクリメント・モードのとき　：MAR＋TC

③ TC，UDCをそれぞれ－1（ワード転送の場合は－2）する。

④ UDCからボローが発生するまで，②，③を繰り返す。

⑤ UDCからボローが発生したらDMA転送を終了し，UDCにDCM（下位21ビット）の値がリロードされる。

〈TCボロー発生時（バッファの境界を越えた場合）〉

DMA割り込み要求（INTD）が発生し，同時にEND/Fフラグがセット(1)されます（DMA禁止状態になる）。引き続きDMA転送を行う場合は，DMA割り込み処理内でTCの値を再設定します(TCには自動的にTCMの値はリロードされません)。このとき，UDCの値にかかわらずDMA禁止状態になるため，TCの値は次のように設定する必要があります。

バイト転送：（N×（UDC＋1）－1）×1

ワード転送：（N×（UDC＋1）－1）×2　　Nは整数

保守／廃止

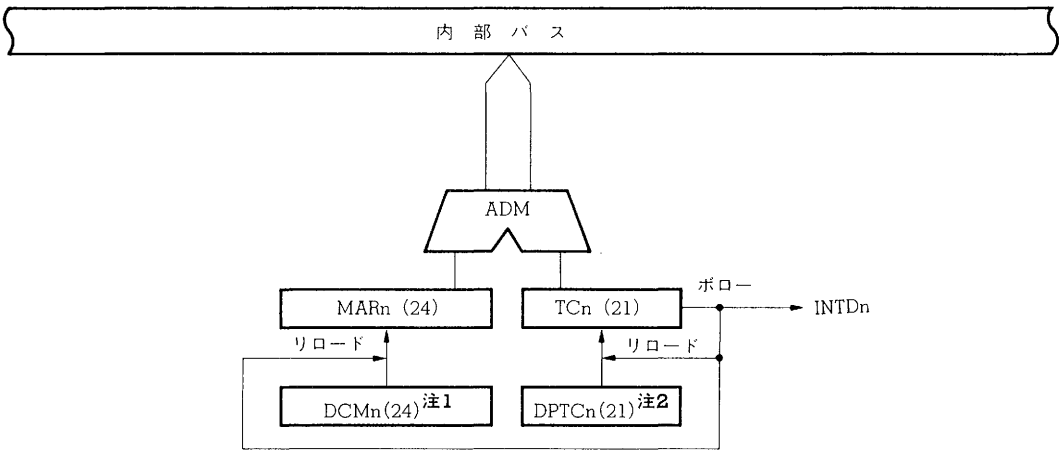
注意　このモードでDMA転送を行う場合，PWCレジスタでノー・ウエイトを指定しても自動的に１ウエイト挿入されます。これは，TCボロー発生時に正常にボローを発生できるようにするためです。

(3) ネクスト・アドレス指定モード

DMA転送対象メモリの指定領域でのDMA転送が終了後，連続してネクストDMAメモリ・アドレス・レジスタ（NMARn＝DCMn：DMAコンペア・レジスタ）の指定するメモリ・アドレスを基にDMA転送を続けて実行するモードです。このモードではDMAアップ・ダウン・カウンタ（UDCn）は使用しません。

図６－12にネクスト・アドレス指定モードの機能ブロック図を示します。

図６－12　ネクスト・アドレス指定モード



注1．DCMnは，NMARnと共用

2．DPTCnは，NTCnと共用

保守／廃止

【レジスタの設定】

レ　ジ　ス　タ　名	設　定　内　容
MARn（DMAメモリ・アドレス・レジスタ）	DMA転送バッファ・メモリの終了アドレス
NMARn（ネクストDMAメモリ・アドレス・レジスタ）	次に実行するDMA転送バッファ・メモリの終了アドレス
TCn（ターミナル・カウンタ）	開始アドレスと終了アドレスとの差に相当するメモリ・サイズ－1（バイト転送の場合） 開始アドレスと終了アドレスとの差に相当するメモリ・サイズ－2（ワード転送の場合）
DPTCn（DMAリード・ライト・ポインタ）注	次に実行するDMA転送の開始アドレスと終了アドレスとの差に相当するメモリ・サイズ－1（バイト転送の場合） 次に実行するDMA転送の開始アドレスと終了アドレスとの差に相当するメモリ・サイズ－2（ワード転送の場合）
DMAC（DMAコントロール・レジスタ）のPDビット	アドレスの更新方向（**6.6.2**参照）

注　このモードでは，DPTCをNTC（ネクスト・ターミナル・カウンタ）として用います。

【動作の説明】

（a）1回のDMA転送動作

① DMA要求の発生

② I/O－メモリ間のデータ転送

転送対象となるメモリ・アドレスは次の値になります。

- インクリメント・モードのとき：MAR－TC
- デクリメント・モードのとき　：MAR＋TC

③ TCを－1（ワード転送の場合は－2）する。

〈TCボロー発生時（バッファの境界を越えた場合）〉

NMARレジスタの内容をMARレジスタに，NTCレジスタの内容をTCにそれぞれリロードし注，同時に割り込み要求（INTD）を発生します。

このとき，END/Fフラグは変化しません。つまりDMA許可状態のままのため，次にDMA要求が発生すると，新しく設定されたTC，MARに従ってDMA転送を続けます。

注　ボロー発生時，NTCレジスタの内容はいったんTCMレジスタにロードされます。TCMレジスタからTCレジスタにロードされるタイミングは，次のDMAサイクルを実行する直前です。

したがって，ボロー発生直後にTCレジスタを読み出すとそのままの値が読み出されます。

保守／廃止

(b) メモリからの転送データの読み出し，書き込み

メモリへは，通常のメモリ操作命令で書き込み，読み出しを行います。入出力リード・ライト・ポインタ（IOPn）による操作は行いません。

注意1．DMA割り込み処理内でTCの値を変更する場合は，DMA転送を禁止してから変更してください。また，そのとき必ず次のどちらかの操作を行ってください。

- TCMレジスタに新しく設定するTCの値を書き込む
- DMASレジスタのEND/Fフラグに“1”を書き込む（初期状態に戻すことにより，TCの値がそのままダウンカウントされる）

ただし，DMA転送を禁止する前に次のDMA要求が発生すると，TCMの値がTCにロードされ実行されてしまいます。

2．ネクスト・アドレス指定モードのDMAでは，TCおよびDPTC（NTC）への設定値をバイト転送では１以上，ワード転送では２以上に設定してください。

保守／廃止

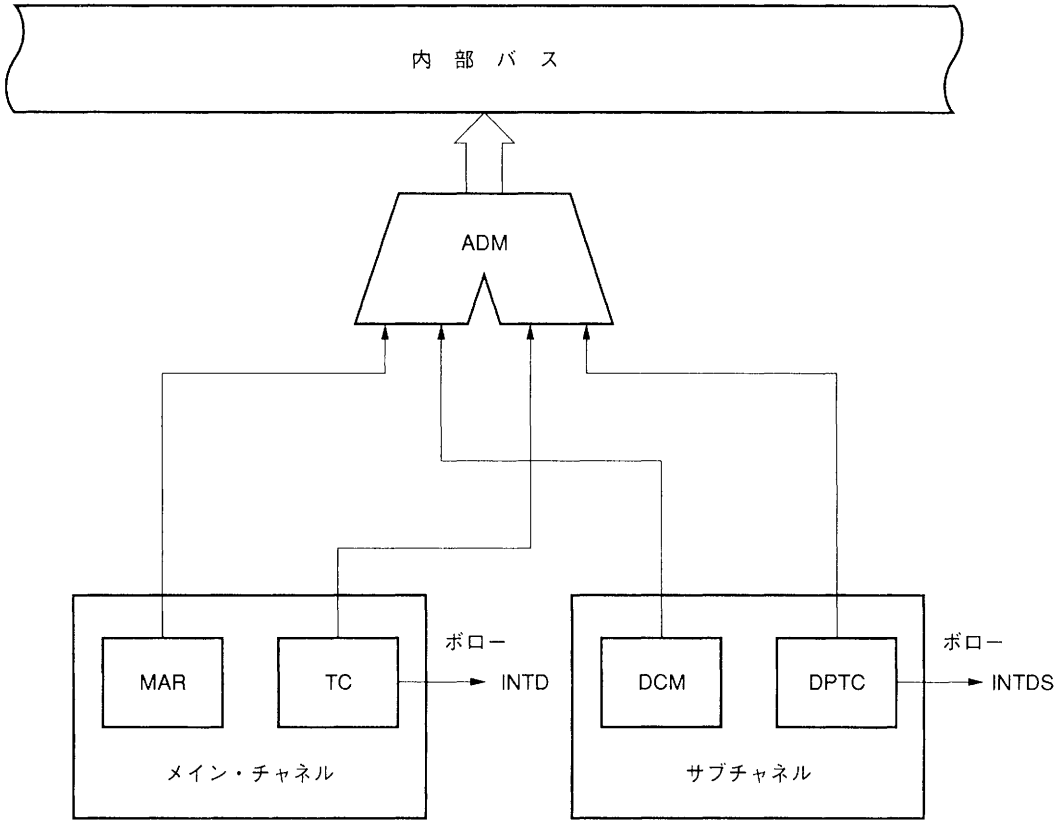
（4）２チャネル動作モード

チャネル０およびチャネル１が，おのおのメイン・チャネルとサブチャネルの２チャネルの制御により，それぞれ２種類のDMA転送要求ソースに対するDMA転送を実行するモードです。メイン・チャネル，サブチャネルとも，指定回数のDMA転送を終了したときにDMA転送を停止するモードと，継続して同一の転送領域からDMA転送を繰り返し行うモードの２つのモードを持っています。

各モードの機能ブロック図を図６－13，６－14に示します。

（a）DMA転送終了時に停止するモード（DMAMnレジスタ　MD2＝1，MD1＝MD0＝0）

図６－13　２チャネル動作モード（停止モード）



保守／廃止

【レジスタの設定】

●メイン・チャネル側

レ　ジ　ス　タ　名	設　定　内　容
MARn（DMAメモリ・アドレス・レジスタ）	DMA転送バッファ・メモリの終了アドレス（ベース・アドレス）
TCn（ターミナル・カウンタ）	転送バイト数－1（バイト転送のとき） 転送バイト数－2（ワード転送のとき）
DMAC（DMAコントロール・レジスタ）のPDビット	アドレスの更新方向（**6.6.2**参照）

●サブチャネル側

レ　ジ　ス　タ　名	設　定　内　容
DCMn（DMAコンペア・レジスタ） ＝サブMARn	DMA転送バッファ・メモリの終了アドレス（ベース・アドレス）
DPTCn（DMAリード・ライト・ポインタ） ＝サブTCn	転送バイト数－1（バイト転送のとき） 転送バイト数－2（ワード転送のとき）
DMAC（DMAコントロール・レジスタ）のPSビット	アドレスの更新方向（**6.6.2**参照）

【動作の説明】

① DMA要求の発生

② I/O－メモリ間のデータ転送

転送対象となるメモリ・アドレスは次の値になります。

●インクリメント・モードのとき：MAR－TC

●デクリメント・モードのとき　：MAR＋TC

③ メイン・チャネル側：TCを－1（ワード転送の場合は－2）する。

サブチャネル側　　：サブTCを－1（ワード転送の場合は－2）する。

〈TC（サブTC）ボロー発生時（バッファの境界を越えた場合）〉

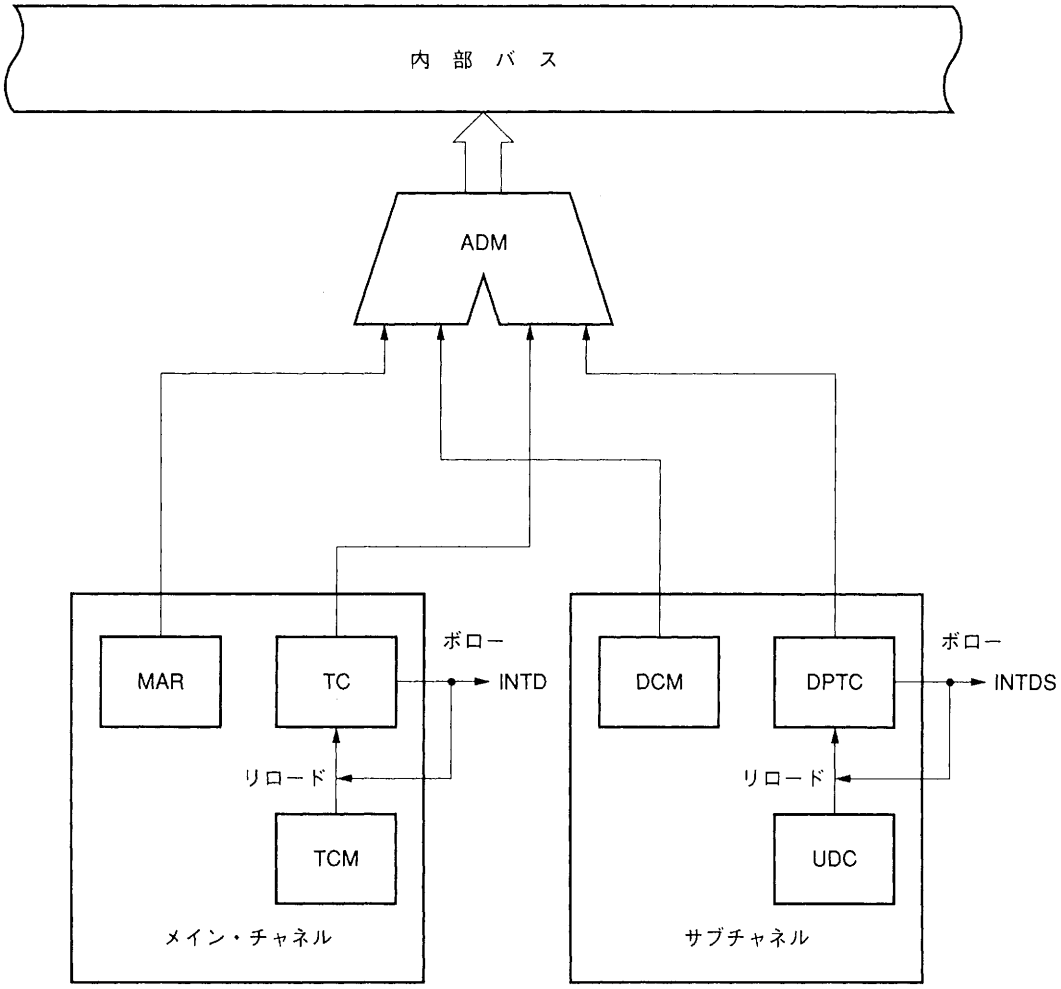
メイン・チャネル側：DMA割り込み要求(INTD)を発生し，同時にEND/Fフラグがセット(1)され，DMA転送禁止状態になります。

サブチャネル側　　：DMA割り込み要求（INTDS）を発生し，同時にEMPフラグがセット(1)され，DMA転送禁止状態になります。

保守／廃止

（b）DMA転送終了時に繰り返し転送を続行するモード（DMAMnレジスタ　MD2＝MD0＝1，MD1＝0）

図6－14　2チャネル動作モード（繰り返しモード）



【レジスタの設定】

●メイン・チャネル側

レジスタ名	設定内容
MARn（DMAメモリ・アドレス・レジスタ）	DMA転送バッファ・メモリの終了アドレス（ベース・アドレス）
TCn（ターミナル・カウンタ） TCMn（ターミナル・カウンタ・モジュロ・レジスタ）	転送バイト数－1（バイト転送のとき） 転送バイト数－2（ワード転送のとき）
DMAC（DMAコントロール・レジスタ）のPDビット	アドレスの更新方向（6.6.2参照）

保守／廃止

●サブチャネル側

レ　ジ　ス　タ　名	設　定　内　容
DCMn（DMAコンペア・レジスタ） ＝サブMARn	DMA転送バッファ・メモリの終了アドレス（ベース・アドレス）
DPTCn（DMAリード・ライト・ポインタ） ＝サブTCn UDCn（DMAアップ・ダウン・カウンタ） ＝サブTCMn	転送バイト数－1（バイト転送のとき） 転送バイト数－2（ワード転送のとき）
DMAC（DMAコントロール・レジスタ）のPSビット	アドレスの更新方向（**6.6.2**参照）

【動作の説明】

① DMA要求の発生

② I/O－メモリ間のデータ転送

転送対象となるメモリ・アドレスは次の値になります。

- ●インクリメント・モードのとき：MAR－TC
- ●デクリメント・モードのとき　：MAR＋TC

③ メイン・チャネル側：TCを－1（ワード転送の場合は－2）する。

サブチャネル側　　：サブTCを－1（ワード転送の場合は－2）する。

〈TC（サブTC）ボロー発生時（バッファの境界を越えた場合）〉

メイン・チャネル側：DMA割り込み要求（INTD）を発生し，同時にTCMレジスタの内容をTCへ自動的にリロードします。このとき，END/Fフラグは変化せず，DMAは禁止状態になりません。したがって，繰り返し同一のDMA転送領域でのDMA転送を実行します。

サブチャネル側　　：DMA割り込み要求（INTDS）を発生し，同時にサブTCMの内容をサブTCへ自動的にリロードします。このとき，EMPフラグは変化せずDMAは禁止状態になりません。したがって，繰り返し同一のDMA転送領域でのDMA転送を実行します。

備考　動作中のDMA転送は，DMA割り込み処理などでソフトウエアによりDMA許可フラグ（EDMA）をクリア(0)すると停止します。

保守／廃止

(5) メモリ－メモリ転送モード

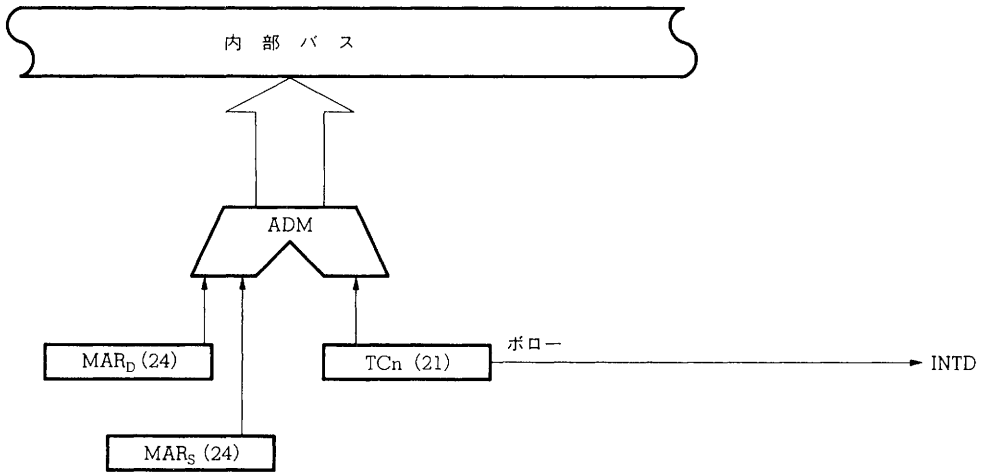
２回のバス・サイクルを連続して起動し，メモリとメモリとの間でDMA転送を実行するモードです。このモードではDMAアップ・ダウン・カウンタ（UDCn）およびDMAリード・ライト・ポインタ（DPTCn）は使用しません。

メモリ－メモリ転送モードには，指定回数のDMA転送を終了したときにDMA転送を停止するモードと，継続して同一の転送領域からDMA転送を繰り返し行うモードの２種類があります。

各モードの機能ブロック図を図６－15，６－16に示します。

(a) DMA転送終了時に停止するモード（DMAMnレジスタ　MD2＝MD1＝1，MD0＝0）

図６－15　メモリ-メモリ転送モード
（停止モード）



保守／廃止

【レジスタの設定】

レ　ジ　ス　タ　名	設　定　内　容
DCMn（DMAコンペア・レジスタ） ＝MAR_S	ソース側DMA転送バッファ・メモリの終了アドレス
MARn（DMAメモリ・アドレス・レジスタ） ＝MAR_D	デスティネーション側DMA転送バッファ・メモリの終了アドレス
TCn（ターミナル・カウンタ）	転送バイト数－1（バイト転送のとき） 転送バイト数－2（ワード転送のとき）
DMAC（DMAコントロール・レジスタ）のPSビット	ソース側のアドレスの更新方向（6.6.2参照）
DMAC（DMAコントロール・レジスタ）のPDビット	デスティネーション側のアドレスの更新方向（6.6.2参照）

【動作の説明】

① DMA要求の発生

② メモリ-メモリ間のデータ転送

転送対象となるメモリ・アドレスは次の値になります。

	ソース側	デスティネーション側
インクリメント・モードのとき	MAR_S－TC	MAR_D－TC
デクリメント・モードのとき	MAR_S＋TC	MAR_D＋TC

ただし，アドレス未更新時はMAR_S（ソース側）またはMAR_D（デスティネーション側）のみで指定されます。

③ TCを－1（ワード転送の場合は－2）する。

〈TCボロー発生時（バッファの境界を越えた場合）〉

DMA割り込み要求(INTD)を発生し，同時にEND/Fフラグがセット(1)されDMA禁止状態になります。また，TCMの内容はTCへリロードされます[注]。

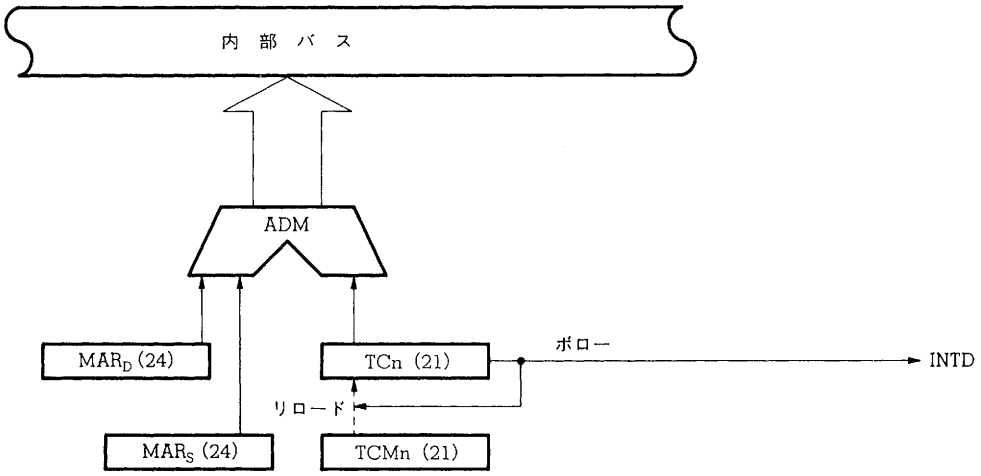
注　TCMレジスタの内容がTCレジスタにロードされるタイミングは，次のDMAサイクルを実行する直前です。

したがって，ボロー発生直後にTCレジスタを読み出すとそのままの値が読み出されます。

保守／廃止

(b) DMA転送終了時に繰り返し転送を続行するモード（DMAMnレジスタ　MD2＝MD0＝MD1＝1）

図６－16　メモリ-メモリ転送モード
（繰り返しモード）



【レジスタの設定】

レ　ジ　ス　タ　名	設　定　内　容
DCMn（DMAコンペア・レジスタ） ＝MAR_S	ソース側DMA転送バッファ・メモリの終了アドレス
MARn（DMAメモリ・アドレス・レジスタ） ＝MAR_D	デスティネーション側DMA転送バッファ・メモリの終了アドレス
TCn（ターミナル・カウンタ） TCMn（ターミナル・カウンタ・モジュロ・レジスタ）	転送バイト数－1（バイト転送のとき） 転送バイト数－2（ワード転送のとき）
DMAC（DMAコントロール・レジスタ）のPSビット	ソース側のアドレスの更新方向（6.6.2参照）
DMAC（DMAコントロール・レジスタ）のPDビット	デスティネーション側のアドレスの更新方向（6.6.2参照）

【動作の説明】

① DMA要求の発生

② メモリ-メモリ間のデータ転送

転送対象となるメモリ・アドレスは次の値になります。

保守／廃止

	ソース側	デスティネーション側
インクリメント・モードのとき	MAR_S－TC	MAR_D－TC
デクリメント・モードのとき	MAR_S＋TC	MAR_D＋TC

ただし，アドレス未更新時はMAR_S（ソース側）またはMAR_D（デスティネーション側）のみで指定されます。

③ TCを－1（ワード転送の場合は－2）する。

〈TCボロー発生時（バッファの境界を越えた場合）〉

DMA割り込み要求(INTD)を発生し，同時にTCMの内容をTCへ自動的にリロードします[注]。このときEND/Fフラグは変化せず，繰り返し同一のDMA転送領域でのDMA転送を実行します。

注　TCMレジスタの内容がTCレジスタにロードされるタイミングは，次のDMAサイクルを実行する直前です。
したがって，ボロー発生直後にTCレジスタを読み出すとそのままの値が読み出されます。

備考　動作中のDMA転送は，DMA割り込み処理などでソフトウエアによりDMA許可フラグ（EDMA）をクリア（0）すると停止します。

保守／廃止

6.4　動　　作

一連のDMA転送は，複数回のDMA転送サイクルに分けて実行されます。次に，DMA転送サイクルの起動，停止，中断について説明します。

6.4.1　起　　動

DMA転送許可状態（DMAMレジスタのEDMA，EDMASビットが“1”）において，次のどちらかの方法によりDMA転送サイクルが起動されます。DMAサービス・チャネルの設定に従って実行されます。

(1) 起動要因からのDMA要求

起動要因として次のものが選択できます。各起動要因は，DMAモード・レジスタ（DMAMn）のTS0，TS1ビットにより指定します（**6.6.1　DMAモード・レジスタ（DMAM0，DMAM1）**参照）。

(a) 内蔵周辺ハードウエアからの割り込み要求

- シリアル送信割り込み（INTST0，INTST1）
- シリアル受信割り込み（INTSR0，INTSR1）
- 内蔵タイマ割り込み（タイマ2：CM23一致信号）
- パラレル・インタフェース入出力割り込み（INTPAI）

(b) 外部端子入力（DMARQ0，DMARQ1）

(2) ソフトウエアによるDMA要求

DMAモード・レジスタ（DMAMn）のTDMAビットをソフトウエアによりセット(1)するとDMAを起動できます。ただし，ディマンド・リリース・モードではこの方法で起動することはできません。

バースト・モード，シングルステップ・モード，ディマンド・リリース・モードでは，停止条件が成立するまでDMA転送サイクルを実行し続けます。

HALTモード中にDMA要求が発生した場合，内部的にはHALTモードを解除したあとDMA転送を実行します。DMA転送実行後は再びHALTモードに戻ります。このときプログラム・カウンタの内容は変化しません。

STOPモード中にDMA要求が外部から入力された場合，DMAコントローラはDMA転送を実行しません。また要求の発生も記憶されません。

保守／廃止

6.4.2　停　　止

一連のDMA転送は，次のどれかの停止条件により停止します。

なお，実行中のDMA転送サイクルは最後まで正常に実行されます。STOP，HALT命令とDMA転送が競合した場合でも，DMAサイクル実行中はスタンバイ・モードに入りません。DMA転送終了後，スタンバイ・モードに移り停止します。

(1) DMA転送終了条件成立

- ２チャネル動作モードでDMAステータス・レジスタ（DMAS）のEMPフラグがセット(1)された状態のとき
- DMAステータス・レジスタ（DMAS）のEND/Fフラグがセット(1)された状態のとき

(2) EDMA，EDMASビットの操作

- DMAモード・レジスタ（DMAMn）のEDMAビットをクリア(0)したとき
- DMAモード・レジスタ(DMAMn)のEDMASビットをクリア(0)したとき（２チャネル動作モード時のサブチャネルのDMA動作だけ停止）

(3) DMARQ入力のロウ・レベル検出

ディマンド・リリース・モードのときだけ，DMA転送が停止します。

6.4.3　中　　断

DMA転送は次の要因が起きると中断されます。

(1) バス・ホールド

(2) リフレッシュ・サイクル

DMA転送が中断される要因がなくなると，引き続きDMA転送を再開します。なお，DMA転送中に優先順位の高い別のDMA要求が発生しても，実行中のDMA転送は中断されません。

保守／廃止

6.5　優先順位

6.5.1　チャネル間の優先順位

同時に複数のチャネルへのDMA転送要求がある場合のチャネル間の優先順位は次のとおり固定されています。

（高）　チャネル0（メイン・チャネル）　>　チャネル0（サブチャネル）　>　チャネル1（メイン・チャネル）　>　チャネル1（サブチャネル）　（低）

注意　チャネル間の優先順位の制御にネスティングの概念はなく，一度受け付けたDMA要求はDMA転送終了条件が成立するまで転送し続けます。したがって，バースト・モードで使用する場合は注意してください。

6.5.2　リフレッシュ・サイクルとの優先順位

リフレッシュ要求とDMA要求が同時に発生した場合，DMA転送は保留され，リフレッシュ・サイクルが優先します。

6.5.3　他の制御動作との優先順位

命令によりDMA制御レジスタ操作を行うときには，自動的に両チャネルともDMA実行禁止（要求保留）状態となります。

DMAによるバスの占有により，通常のCPUによるデータ・アクセス，命令フェッチなどの競合以外に次のような条件も考慮する必要があります。

① メモリ・リフレッシュ動作
② スタンバイ動作
③ NMI入力
④ ウォッチドッグ・タイマのオーバフロー割り込み（INTWDT）
⑤ 外部ホールド要求

これらの条件とDMAとの優先順位は次のようになります。

外部ホールド要求　>　メモリ・リフレッシュ動作　>　DMA　>　スタンバイ動作

③，④とDMAの競合については，バス・コントロール・ユニットの動作にはまったく影響しません。つまり，DMA動作中はバス制御権がDMAコントローラにあるため，NMI，INTWDTの処理プログラムでバス要求が発生してもDMAが優先されます（バス要求が発生しない処理は並列に実行されます）。

CPUによる外部メモリへのアクセスとDMAまたはリフレッシュが競合した場合には，あとからアクセスを要求した側のバス・サイクル起動を保留します。ただし，メモリ・アクセスとDMAの要求が同時の場合は，メモリ・アクセスの方が優先されます。

保守／廃止

6.6　制御レジスタ

DMA動作モード，転送モードの指定を行うために，DMAチャネルごとにDMAモード・レジスタ（DMAMn），DMAコントロール・レジスタ（DMACn）があります。また，DMA転送終了状態などを示すDMAステータス・レジスタ（DMAS）があります。

6.6.1　DMAモード・レジスタ（DMAM0，DMAM1）

DMAの動作モード，DMA転送の禁止／許可などを指定する８ビットのレジスタです。

８ビットまたは１ビットのメモリ操作命令によって書き込み／読み出しが可能です。

リセット後の値は，E0Hになります。以降，表中の網がけの部分はリセット後の状態を示します。

DMAM0
DMAM1

7	6	5	4	3	2	1	0
MD2	MD1	MD0	EDMAS	EDMA	TDMA	TS1	TS0

MD2，**MD1**，**MD0** 動作モードの指定

動作モードを指定します。

MD2	MD1	MD0	動作モード	
0	0	0	I/O(SFR)→メモリ転送	インテリジェントDMAモード－１（リング・バッファ方式）
0	0	1	I/O(SFR)←メモリ転送	インテリジェントDMAモード－２（カウンタ制御方式）
0	1	0	I/O(SFR)↔メモリ転送	ネクスト・アドレス指定モード
0	1	1	設定禁止	
1	0	0	I/O↔メモリ転送	２チャネル動作モード（終了時停止）
1	0	1	I/O↔メモリ転送	２チャネル動作モード（繰り返し）
1	1	0	メモリ↔メモリ転送	メモリ－メモリ転送モード（終了時停止）
1	1	1	メモリ↔メモリ転送	メモリ－メモリ転送モード（繰り返し）

EDMAS **DMA転送許可／禁止制御ビット（サブチャネル）**

２チャネル動作モード時のサブチャネルのDMA転送の許可，禁止を指定するビットです。

EDMAS	DMA転送の許可／禁止
0	DMA転送禁止状態
1	DMA転送許可状態

DMAステータス・レジスタ(DMAS)のEMPフラグがセット(1)されているときには，EDMASビットをセット(1)してもDMA転送は許可状態になりません。

保守／廃止

EDMA　DMA転送許可／禁止制御ビット

DMA転送の許可，禁止を指定するビットです。

EDMA	DMA転送の許可／禁止
0	DMA転送禁止状態
1	DMA転送許可状態

DMAステータス・レジスタ（DMAS）のEND/Fフラグがセット(1)されているときには，EDMAビットをセット(1)してもDMA転送は許可状態になりません。

TDMA　DMA転送のソフトウエア起動ビット

DMA転送をソフトウエアで起動するためのトリガ・ビットです。

EDMA	TDMA	転送スタート
0	×	DMA転送禁止状態
1	0	状態維持
1	1	DMA転送スタート（ソフトウエア・トリガ）

DMAステータス・レジスタ（DMAS）のEND/Fフラグがセット(1)されているときには，DMA転送は起動できません。

このビットの読み出しレベルは常に(0)です。

ディマンド・リリース・モードではソフトウエア・トリガで起動することはできません。

また，２チャネル動作モードではTDMAビットによりメイン・チャネルのみソフトウエア・トリガで起動することができます。

TS1，TS0　DMA起動要求，転送I/Oの指定

TS1，TS0の２ビットにより最大４ソースの中からDMA起動要求を選択することが可能です。

次に示す表において，２チャネル動作モードのときは，メイン・チャネルおよびサブチャネルで指定されたDMA要求が選択され，２チャネル動作モード以外のときは，メイン・チャネルで指定されたDMA要求が選択されます。

保守／廃止

（チャネル０）

対応転送モード（転送対象）	TS1	TS0	メイン・チャネル		サブチャネル	
			DMA起動トリガ	転送 I/O	DMA起動トリガ	転送 I/O
I/O－メモリ転送	0	0	パラレル・インタフェース INTPAI	データ入出力バッファ (PAB)	シリアル送信完了 INTST0	送信バッファ (TxB0)
			転送方向（I/O ↔ メモリ）注		転送方向（メモリ → I/O）	
	0	1	DMARQ0	外部 I/O ($\overline{\text{DMAAK0}}$)	DMARQ1	外部 I/O ($\overline{\text{DMAAK1}}$)
			転送方向（I/O ↔ メモリ）		転送方向（メイン・チャネルと同方向）	
	1	0	内蔵タイマ CM23一致信号	外部 I/O ($\overline{\text{DMAAK0}}$)	DMARQ1	外部 I/O ($\overline{\text{DMAAK1}}$)
			転送方向（I/O ↔ メモリ）		転送方向（メイン・チャネルと逆方向）	
	1	1	シリアル送信完了 INTST0	送信バッファ (TxB0)	シリアル送信完了 INTST1	送信バッファ (TxB1)
			転送方向（メモリ → I/O）		転送方向（メモリ → I/O）	
メモリ－メモリ転　送	0	0	設定禁止			
	0	1	DMARQ0	—		
	1	0	CM23一致信号	—		
	1	1	設定禁止			

注　メイン・チャネル側の転送方向にメモリ → I/O（DIR＝1）を選択した場合，サブチャネル側は使用できません。

注意　メモリ－メモリ転送では，$\overline{\text{DMAAK}n}$および内蔵タイマ，シリアル制御回路に対するアクノリッジ信号は出力しません。

保守／廃止

（チャネル１）

対応転送モード（転送対象）	TS1	TS0	メイン・チャネル		サブチャネル	
			DMA起動トリガ	転送 I/O	DMA起動トリガ	転送 I/O
I/O－メモリ転送	0	0	シリアル受信完了 INTSR1	受信バッファ (RxB1)	シリアル送信完了 INTST0	送信バッファ (TxB0)
			転送方向（I/O → メモリ）		転送方向（メモリ → I/O）	
	0	1	内蔵タイマ CM23一致信号	外部 I/O ($\overline{\text{DMAAK1}}$)	DMARQ0	外部 I/O ($\overline{\text{DMAAK0}}$)
			転送方向（I/O ↔ メモリ）		転送方向(メイン・チャネルと同方向)	
	1	0	DMARQ1	外部 I/O ($\overline{\text{DMAAK1}}$)	DMARQ0	外部 I/O ($\overline{\text{DMAAK0}}$)
			転送方向（I/O ↔ メモリ）		転送方向(メイン・チャネルと逆方向)	
	1	1	パラレル・インタフェース INTPAI	データ入出力バッファ (PAB)	シリアル受信完了 INTSR0	受信バッファ (RxB0)
			転送方向（メモリ ↔ I/O）注		転送方向（I/O → メモリ）	
メモリ－メモリ転　送	0	0	設定禁止			
	0	1	CM23一致信号	—		
	1	0	DMARQ1	—		
	1	1	設定禁止			

注　メイン・チャネル側の転送方向にメモリ → I/O（DIR＝1）を選択した場合，サブチャネル側は使用できません。

注意　メモリ－メモリ転送では，$\overline{\text{DMAAKn}}$および内蔵タイマ，シリアル制御回路に対するアクノリッジ信号は出力しません。

保守／廃止

6.6.2　DMAコントロール・レジスタ（DMAC0，DMAC1）

DMAの転送モード，メモリのソース・アドレス，デスティネーション・アドレスの更新方法などを指定する８ビットのレジスタです。8ビットまたは１ビットのメモリ操作命令により，書き込み／読み出しができます。

リセット後の値は，00Hになります。

	7	6	5	4	3	2	1	0
DMAC0 DMAC1	TMD1	TMD0	DIR	W	PD1	PD0	PS1	PS0

アドレスの更新指定

PD1/PS1	PD0/PS0	アドレスの更新モード
0	0	更新せず
0	1	インクリメント
1	0	デクリメント
1	1	更新せず

転送処理方法の指定

W	転送処理方法
0	バイト転送
1	ワード転送

転送方向の指定

DIR	転送方向
0	I/O → メモリ転送
1	メモリ → I/O転送

転送モードの指定

TMD1	TMD0	転送モード
0	0	シングルトランスファ・モード
0	1	ディマンド・リリース・モード
1	0	バースト・モード
1	1	シングルステップ・モード

備考　　は，リセット後の状態を示します。

保守／廃止

DIR 転送方向の指定ビット

インテリジェントDMAモード，ネクスト・アドレス指定モード，２チャネル動作モードを指定した場合のみ有効となるビットです。

インテリジェントDMAモード－1では，I/O → メモリ転送のみ可能ですので，必ずDIRビットをクリア(0)してください。

同様にインテリジェントDMAモード－2では，メモリ → I/O転送のみ可能ですので，必ずDIRビットをセット(1)してください。

また，メモリ－メモリ転送ではDIRビットを必ず“0”に設定してください。

PD1，**PD0**／**PS1**，**PS0** アドレス更新の指定ビット

PD1，PD0ビットの指定による，各動作モードのメモリ・アドレスの生成内容は次のようになります。

転送対象　：I/O ↔ メモリ

動作モード：インテリジェントDMAモード－1，インテリジェントDMAモード－2
ネクスト・アドレス指定モード，２チャネル動作モード

PD1	PD0	アドレス更新モード	メモリ・アドレスの生成方法
0	0	設定禁止	—
0	1	インクリメント	MAR−TC
1	0	デクリメント	MAR+TC
1	1	設定禁止	—

転送対象　：メモリ ↔ メモリ

動作モード：メモリ－メモリ転送モード

PD1	PD0	アドレス更新モード	メモリ・アドレスの生成方法
0	0	未更新	MAR
0	1	インクリメント	MAR−TC
1	0	デクリメント	MAR+TC
1	1	未更新	MAR

アドレスの未更新モードはメモリ－メモリ転送モード時のみ可能です。そのほかの動作モードでは，設定禁止になります。

アドレスの未更新モードを利用して，DMA転送を通常のメモリに対してではなく，メモリ・マップドI/O（FIFOなど）を対象に行うことも可能です。

また，アドレスの未更新モード時もTCの減算は実行され，DMA終了信号出力およびDMA割り込

保守／廃止

み要求（INTD）が発生します。

PS1，PS0ビットの指定による，各動作モードのメモリ・アドレスの生成内容は次のようになります。

転送対象　：メモリ↔メモリ

動作モード：メモリ－メモリ転送モード

PS1	PS0	アドレス更新モード	メモリ・アドレスの生成方法
0	0	未更新	DCM
0	1	インクリメント	DCM－TC
1	0	デクリメント	DCM＋TC
1	1	未更新	DCM

転送対象　：メモリ↔I/O

動作モード：インテリジェントDMAモード－1，インテリジェントDMAモード－2

PS1	PS0	アドレス更新モード	メモリ・アドレスの生成方法
0	0	設定禁止	—
0	1	インクリメント	MAR－DPTC
1	0	デクリメント	MAR＋DPTC
1	1	設定禁止	—

DPTCは，１回のIOPnによる読み出し／書き込み命令実行ごとにデクリメントされます。

ネクスト・アドレス指定モードのときは，PS1＝0，PS0＝0を設定してください。

保守／廃止

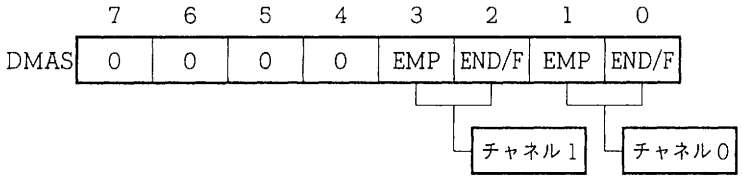
6.6.3　DMAステータス・レジスタ（DMAS）

DMA転送終了状態，インテリジェントDMAモード－1におけるリング・バッファ内のデータのバッファ・フル／バッファ・エンプティ状態を示す，リード／ライト可能な８ビットのレジスタです。

８ビットのメモリ操作命令によりレジスタの読み出し／書き込み（リセットのみ）ができます。レジスタの内容を書き換える際には，ビット7－4には必ず“0”を設定してください。

注意1．レジスタの内容はDMA転送モードを切り替えても初期化されないため，モードの切り替え後必要に応じて初期化してください。

2．１ビット操作命令は使用できません。リセットするビットだけを１にしたデータを８ビット操作命令で書き込んでください。ただし，ビット・テスト命令を使用することはできます。



END/F　転送終了／バッファ・フル・フラグ

インテリジェントDMAモード－1，インテリジェントDMAモード－2，２チャネル動作モード（終了時停止モード），メモリ－メモリ転送モード（終了時停止モード）時に有効なフラグです。

ネクスト・アドレス指定モード，２チャネル動作モード（繰り返し），メモリ－メモリ転送モード（繰り返し）ではこのフラグは変化しません。このフラグがセット（1）されているときは，DMAMレジスタのEDMAビットがセット（1）されていてもDMA転送は禁止状態となります。

（1）インテリジェントDMAモード－1

リング・バッファ内のデータ・フル状態検出フラグとして機能します。

DMA転送によりDMAアップ・ダウン・カウンタ（UDC）がインクリメントされ，DMAコンペア・レジスタ（DCM）の設定値に一致するとセット（1）されます。

また，UDCとDCMレジスタの値が一致した状態から，入出力命令でバッファ内のデータが読み出され，UDCがデクリメントされたときにリセット（0）されます。また，命令によって“1”を書き込む操作でもリセット（0）されます。

（2）インテリジェントDMAモード－2，２チャネル動作モード（終了時停止），メモリ－メモリ転送モード（終了時停止）

ターミナル・カウンタ（TC）の設定回数のDMA転送が終了したことを示すフラグとして機能します。

保守／廃止

TCのデクリメント時にボローが発生するとセット(1)されます。

また，命令によって“1”を書き込む操作でリセット(0)されます。

EMP バッファ・エンプティ・フラグ

インテリジェントDMAモード－1，2チャネル動作モード時に有効なフラグです。ほかのモードではこのフラグは変化しません。このフラグがセット(1)されているときは，DMAMレジスタのEDMASビットがセット(1)されていてもDMA転送は禁止状態となります。

(1) インテリジェントDMAモード－1

リング・バッファ内のデータ・エンプティ状態検出フラグとして機能します。

入出力リード・ライト・ポインタ（IOP）を介してバッファ内のデータを読み出すことによりUDCがデクリメントされ，UDCからボローが発生するとセット(1)されます。DMAが実行され，リング・バッファ内に有効データが格納されるとリセット(0)されます。また，命令によって“1”を書き込む操作でもリセット(0)されます。

(2) 2チャネル動作モード

サブチャネル側のターミナル・カウンタ（UDC）に設定した転送回数分のDMA転送が終了したことを示すフラグです。

UDCのデクリメント時にボローが発生するとセット(1)されます。また，命令によって“1”を書き込む操作でリセット(0)されます。

保守／廃止

6.7　DMAサービス・チャネル

DMA転送の転送元，転送先，転送回数などを指定するために，次に示すようなレジスタが特殊機能レジスタ領域内に，チャネルごとに用意されています（チャネル構成は図６－３　チャネル構成図参照）。

各動作モードで使用されるレジスタを表６－６に示します。

各レジスタの下位16ビット（ビット15－0）は，命令によるワード単位での読み書きが可能です。バイト単位でのアクセスはできません。

表６－６　動作モードとレジスタの対応

レジスタ名称（ビット数）	動作モード				
	インテリジェントDMAモード－１	インテリジェントDMAモード－２	ネクスト・アドレス指定モード	２チャネル動作モード	メモリ－メモリ転送モード
TC（21ビット）	TC	TC	TC	メインTC	TC
TCM（21ビット）	TCM	TCM	未使用	メインTCM	TCM
UDC（21ビット）	UDC	UDC	未使用	サブTCM	未使用
DPTC（21ビット）	DPTC	DPTC	NTC	サブTC	未使用
MAR（24ビット）	MAR	MAR	MAR	メインMAR	MAR_D（デスティネーション）
DCM（24ビット）	DCM	DCM	NMAR	サブMAR	MAR_S（ソース）

6.7.1　ターミナル・カウンタ（TC0，TC1）

DMAの転送バイト数を設定することにより転送回数を制御する21ビット・レジスタです。

１回のDMA転送実行ごとに内容がデクリメントされ，ボローの発生によりDMA転送が終了します。バイト転送／ワード転送に応じて，次のように転送回数を設定してください。

バイト転送：（転送回数－1）×1

ワード転送：（転送回数－1）×2

最大転送バイト数は2097152バイトで，ワード転送では最大転送回数は1048576回となります。

保守／廃止

6.7.2　ターミナル・カウンタ・モジュロ・レジスタ（TCM0，TCM1）

DMA転送回数の情報をTCnにリロードする21ビットのモジュロ・レジスタです。インテリジェントDMAモード－1，およびインテリジェントDMAモード－2では，DPTCにもリロードされます。

6.7.3　DMAアップ・ダウン・カウンタ（UDC0，UDC1）

UDCnは，インテリジェントDMAモードのとき使用される21ビット・レジスタです。インテリジェントDMAモード－1ではDMA転送バイト（ワード）数とCPUによるメモリの読み出し／書き込みデータのバイト（ワード）数との差を示し，１回のDMA転送が実行されるとインクリメントされ，I/O操作命令により入出力リード・ライト・ポインタ（IOP）に読み出しまたは書き込みされるとデクリメントされます。

インテリジェントDMAモード－2では，１回のDMA起動要求により実行するDMAの転送回数を設定してください。

また２チャネル動作モードでは，サブチャネル側のTCMレジスタとして使用されます。

6.7.4　DMAリード・ライト・ポインタ（DPTC0，DPTC1）

DPTCnは，インテリジェントDMAモードのとき使用され，CPUによるメモリのリード／ライト・ポインタを示すための21ビット・レジスタです。リード／ライト・アドレス（MARn±DPTCn）は，入出力リード・ライト・ポインタ（IOP）により示されます。

ネクスト・アドレス指定モードでのネクスト・ターミナル・カウンタ（NTCn）と兼用です。

また２チャネル動作モードでは，サブチャネル側のTCとして使用されます。

6.7.5　DMAメモリ・アドレス・レジスタ（MAR0，MAR1）

DMA転送対象のメモリ・アドレスを示すための24ビット・レジスタです。

DMA転送の終了メモリ・アドレスを指定します。DMA転送の対象のメモリ・アドレスはMARn±TCnで示されます。

２チャネル動作モードでは，メイン・チャネル側のメモリ・アドレス・レジスタとして使用されます。

メモリ－メモリ転送の場合，メモリ・アドレス・レジスタ（デスティネーション：MAR_Dn）として使用します。

保守／廃止

6.7.6　DMAコンペア・レジスタ（DCM0，DCM1）

DCMnレジスタは，インテリジェントDMAモードのとき使用される24ビット・レジスタです。

インテリジェントDMAモード－1では，UDCnと比較することでバッファ・フル状態を判断するために使用されます。

インテリジェントDMAモード－2では，UDCのデクリメントの結果，ボローが発生したときにUDCにリロードされるDMA転送回数の初期値を保持するレジスタとして機能します。

また２チャネル動作モードでは，サブチャネル側のMARレジスタとして使用されます。

ネクスト・アドレス指定モードでは，ターミナル・カウント終了時（TCのデクリメント結果，ボローが発生した時点）に，MARレジスタにリロードする次の転送先（元）メモリ・アドレスを設定するネクスト・メモリ・アドレス・レジスタ（NMARn）として，メモリ－メモリ転送モードではDMAメモリ・アドレス・レジスタ（ソース：MAR_Sn）として使用されます。

保守／廃止

6.8　割り込み要求

DMAコントローラの各チャネルからの割り込み要求信号を割り込みコントローラに対して伝えます。DMAのサービス・チャネルとDMA割り込み要求信号との対応を次に示します。

〔２チャネル動作モード以外のモード〕

DMAチャネル	割り込み要求信号
チャネル０	INTD0
チャネル１	INTD1

〔２チャネル動作モード〕

DMAチャネル	割り込み要求信号
チャネル０（メイン・チャネル）	INTD0
チャネル１（メイン・チャネル）	INTD1
チャネル０（サブチャネル）	INTDS0
チャネル１（サブチャネル）	INTDS1

(1) インテリジェントDMAモード－1の場合

UDCnがインクリメントされてDCMnレジスタの値と一致したとき(END/Fフラグがセット(1)されるとき)，またはUDCとDCMレジスタの値が一致した状態から，入出力リード・ライト・ポインタ（IOP）を介した入出力命令によりバッファ内のデータが読み出され，UDCがデクリメントされたとき（END/Fフラグがリセット(0)されたとき)，DMA割り込みが発生します。

(2) インテリジェントDMAモード－2

ネクスト・アドレス指定モード，

２チャネル動作モード，

メモリ－メモリ転送モードの場合

DMA転送ごとのTCnのデクリメントによりボローが発生すると，割り込み要求が発生します。

保守／廃止

6.9　DMA要求に対する応答時間

DMA要求に対する応答時間は，次の計算式で求めることができます。

6～(9＋W＋α)（単位：クロック）

W：DMA転送サイクル直前のバス・サイクルのウエイト数

α：各種命令の実行に伴うバス・サイクルとの競合による遅延クロック数

命令によって次の表のように異なります。

命　令	クロック数
SFR（デスティネーション）に対する次の命令 論理演算，増減，補数演算，ビット操作，XCH命令	11＋2a＋b
SFR（デスティネーション）に対する次の命令 シフト，ローテート命令	14＋2a＋b＋n
SFR（デスティネーション）に対する次の命令 比較命令	4＋a
INS命令	24＋2a＋b＋l＋m
EXT命令	15＋a＋b
BTCLR，BTCLRL命令	14＋a＋b
その他の命令	0

a：TMレジスタに対する操作の場合：8

SFR（TxBRG，RxBRG，IMC，ISPR，MK，IC，TMC，TOC，INTM，CT，CM）に対する操作の場合：2

SFR（上記以外）に対する操作の場合：0

b：メモリに対するウエイト数

l：reg8設定値

m：reg8'設定値

n：シフト，ローテート回数

保守／廃止

6.10　DMA転送タイミング

次に，DMA転送モードにおけるタイミング・チャートを示します。DMAコントローラによるDMA転送は，READY端子およびPWC0レジスタ，PWC1レジスタによるウエイト動作を通常のバス・サイクル同様に行います。

タイミング・チャートは，次に挙げる転送対象ごとに示します。

転送対象	モード	図番号	転送対象	モード	図番号
メモリ→外部I/O	ST，SS	6-17	メモリ↔メモリ	ST，SS	6-21
	DR，BS			BS	
外部I/O→メモリ	ST，SS	6-18	メモリ→SFR（メモリ・アクセス）	ST，SS	6-22
	DR，BS			BS	
メモリ→内部I/O（SFR）	ST	6-19	SFR（メモリ・アクセス）→メモリ	ST，SS	6-23
				BS	
内部I/O（SFR）→メモリ	ST	6-20			

備考　ST：シングルトランスファ・モード

SS：シングルステップ・モード

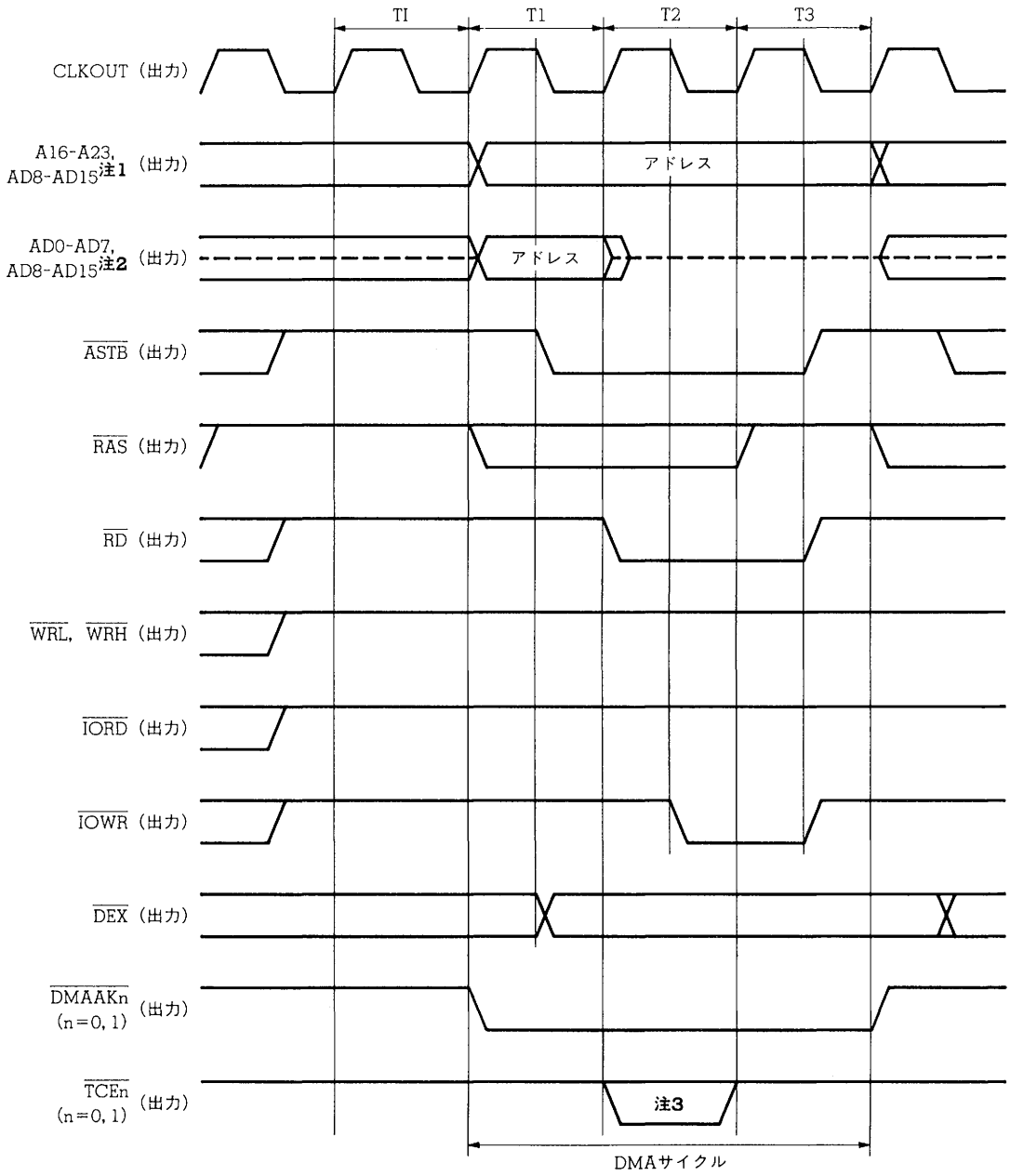
DR：ディマンド・リリース・モード

BS：バースト・モード

保守／廃止

図６－17　メモリ → 外部I/O（1/2）

（a）シングルトランスファ・モード，シングルステップ・モード



注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

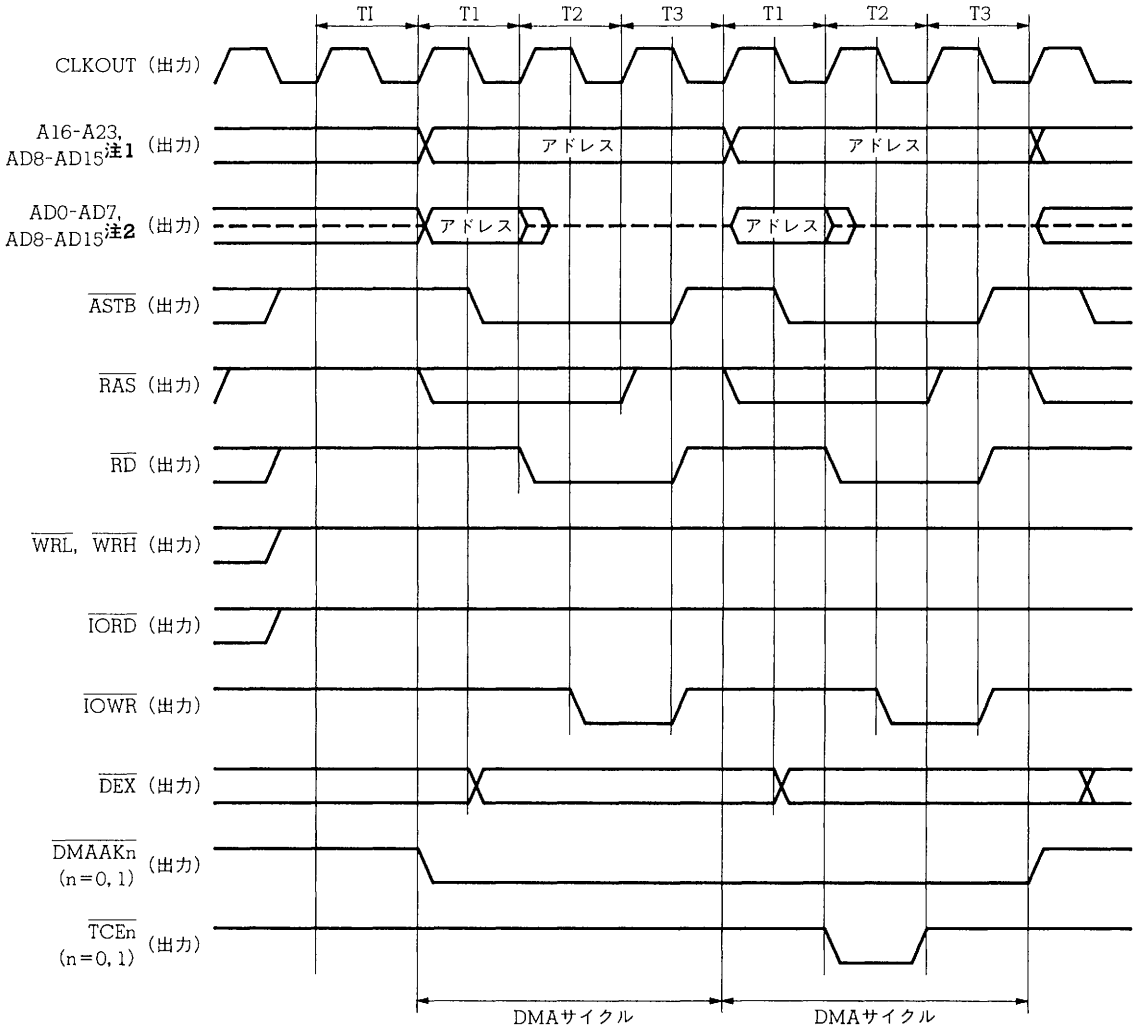
3．TCボローが発生したとき，ロウ・レベルを出力します。

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－17　メモリ → 外部I/O（2/2）

（b）ディマンド・リリース・モード，バースト・モード（２回転送）



注１．外部データ・バス幅８ビットのとき

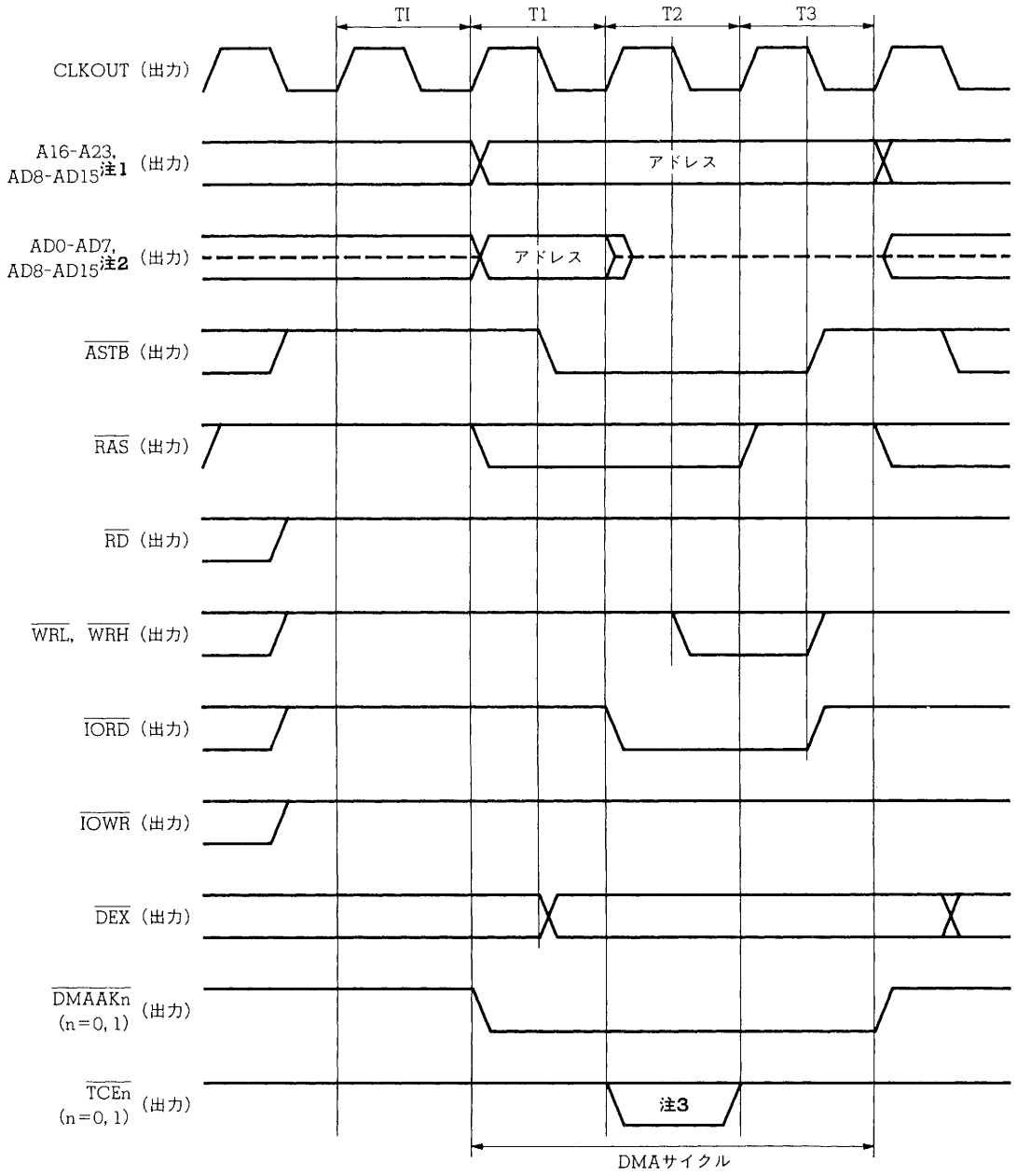
２．外部データ・バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－18　外部I/O → メモリ（1/2）

（a）シングルトランスファ・モード，シングルステップ・モード



注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

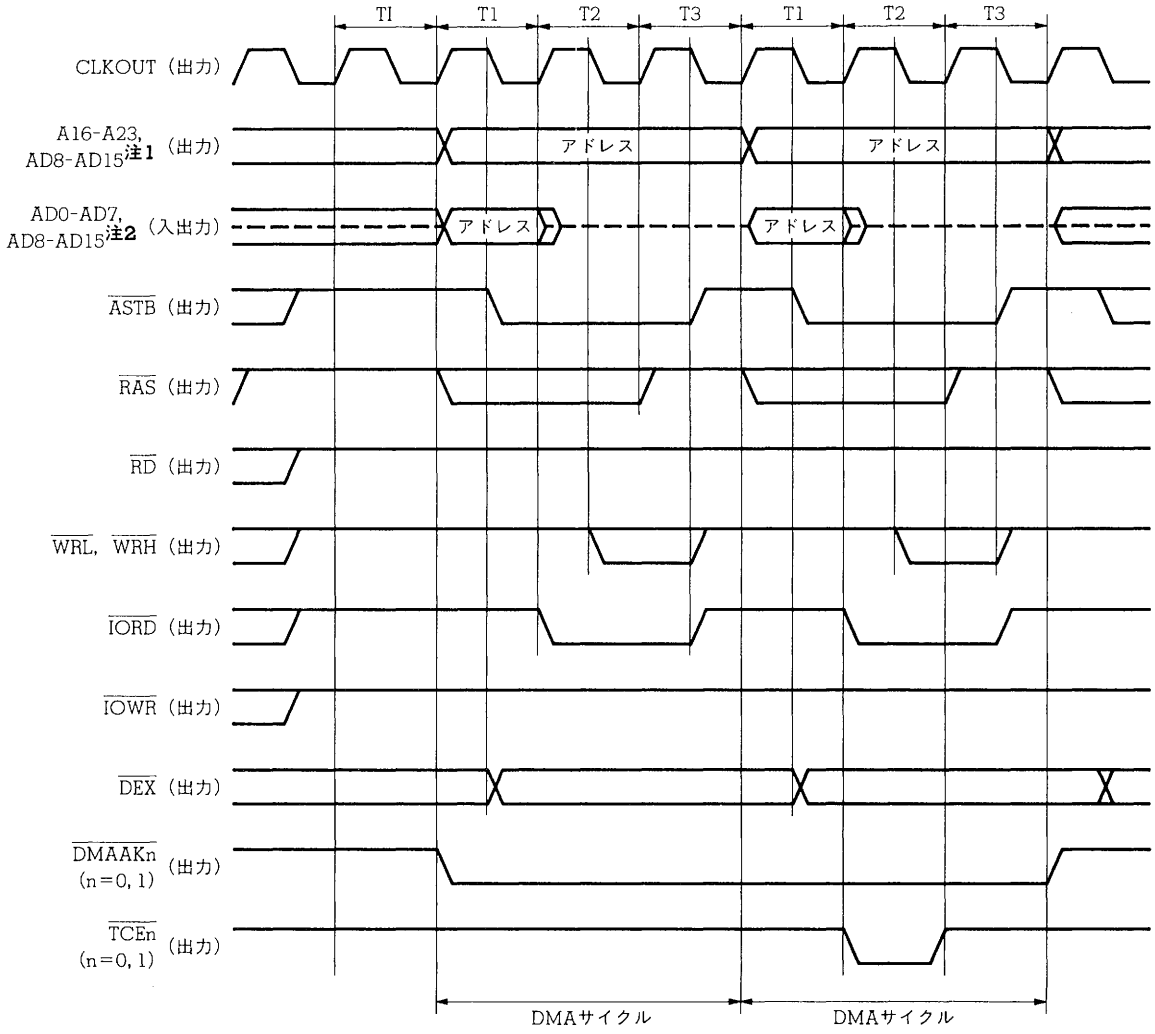
3．TCボローが発生したとき，ロウ・レベルを出力します。

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－18　外部I/O → メモリ（2/2）

（b）ディマンド・リリース・モード，バースト・モード（２回転送）



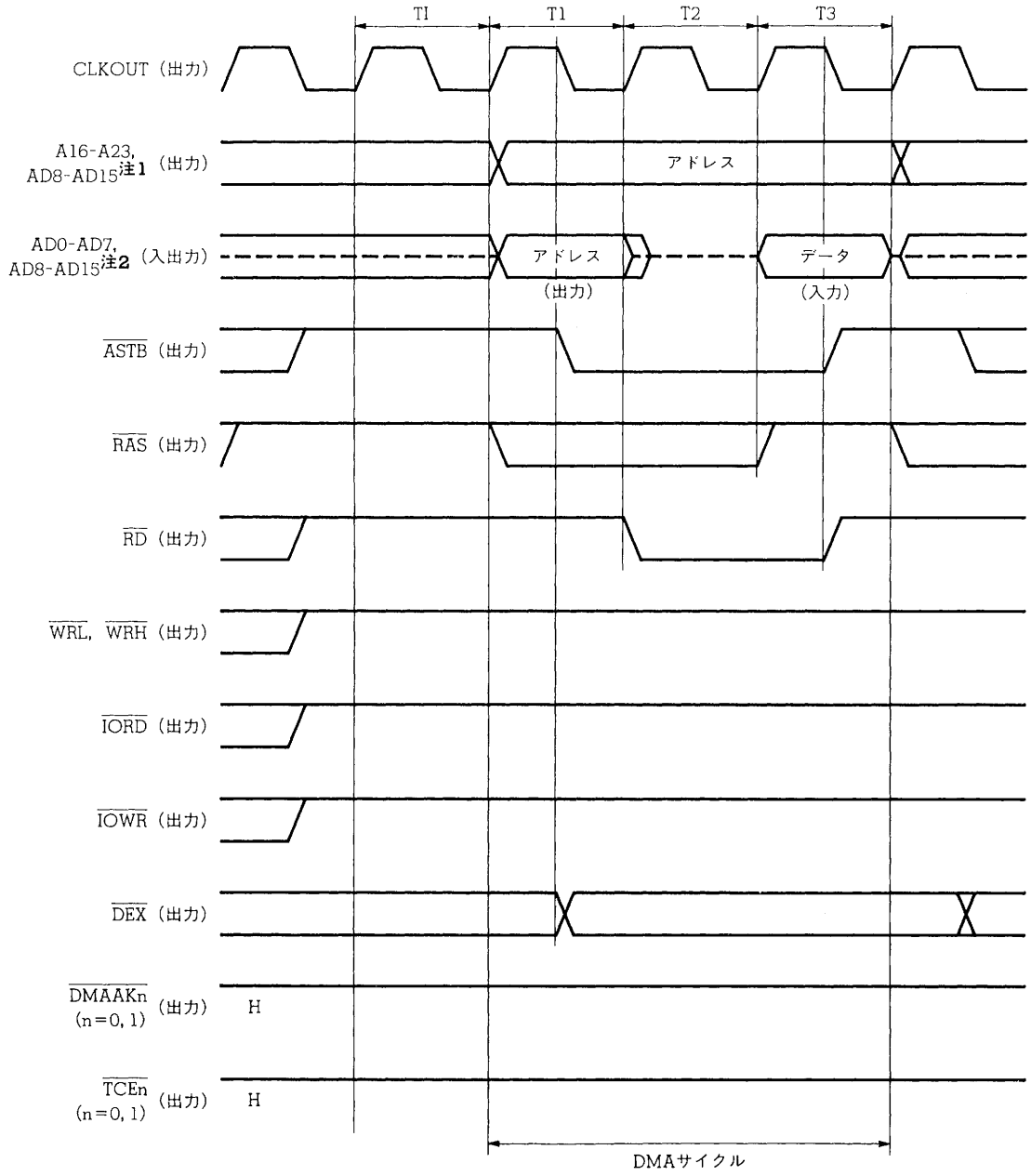
注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－19　メモリ → 内部I/O（SFR）



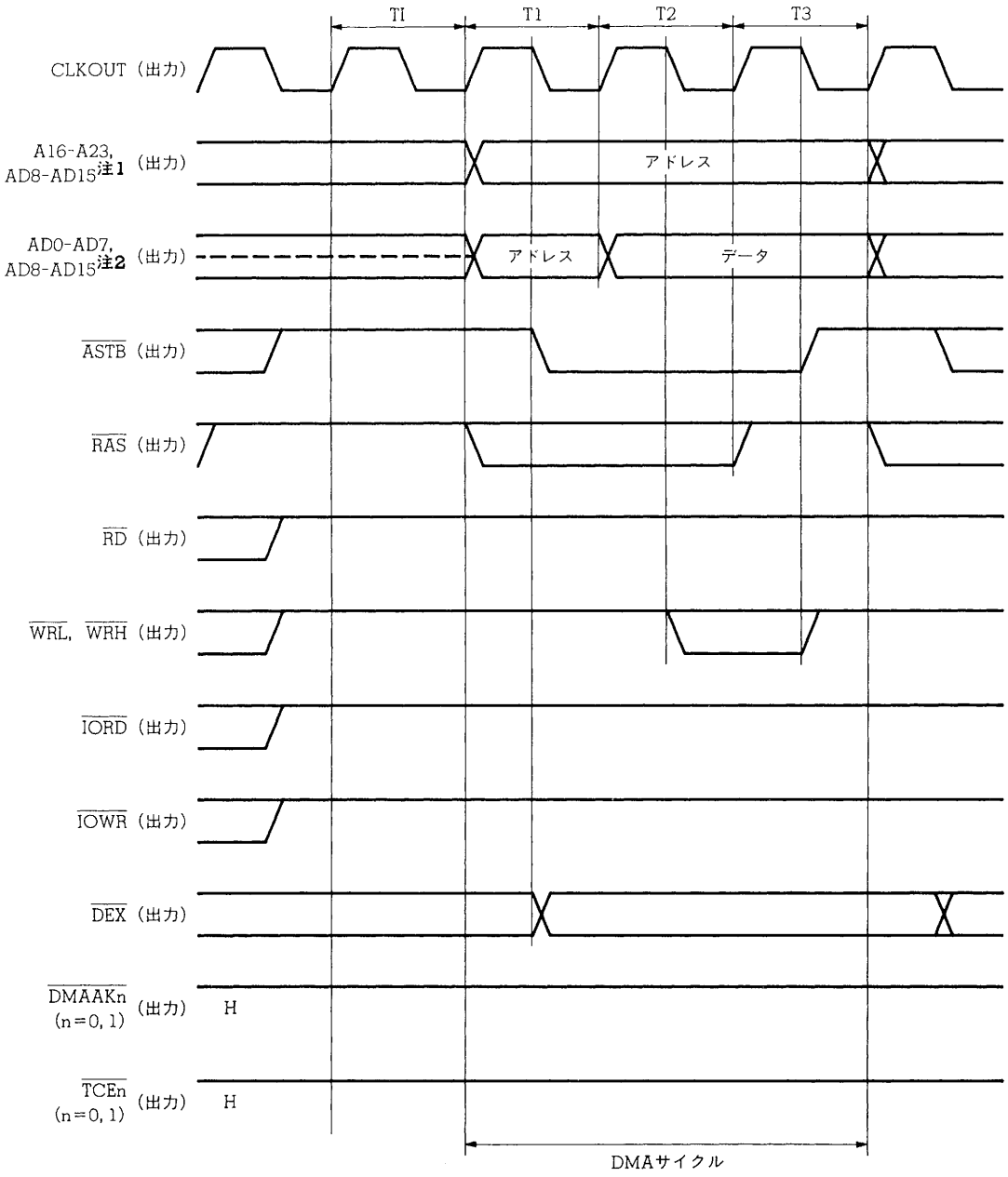
注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－20　内部I/O（SFR）→メモリ



注1．外部データ・バス幅８ビットのとき

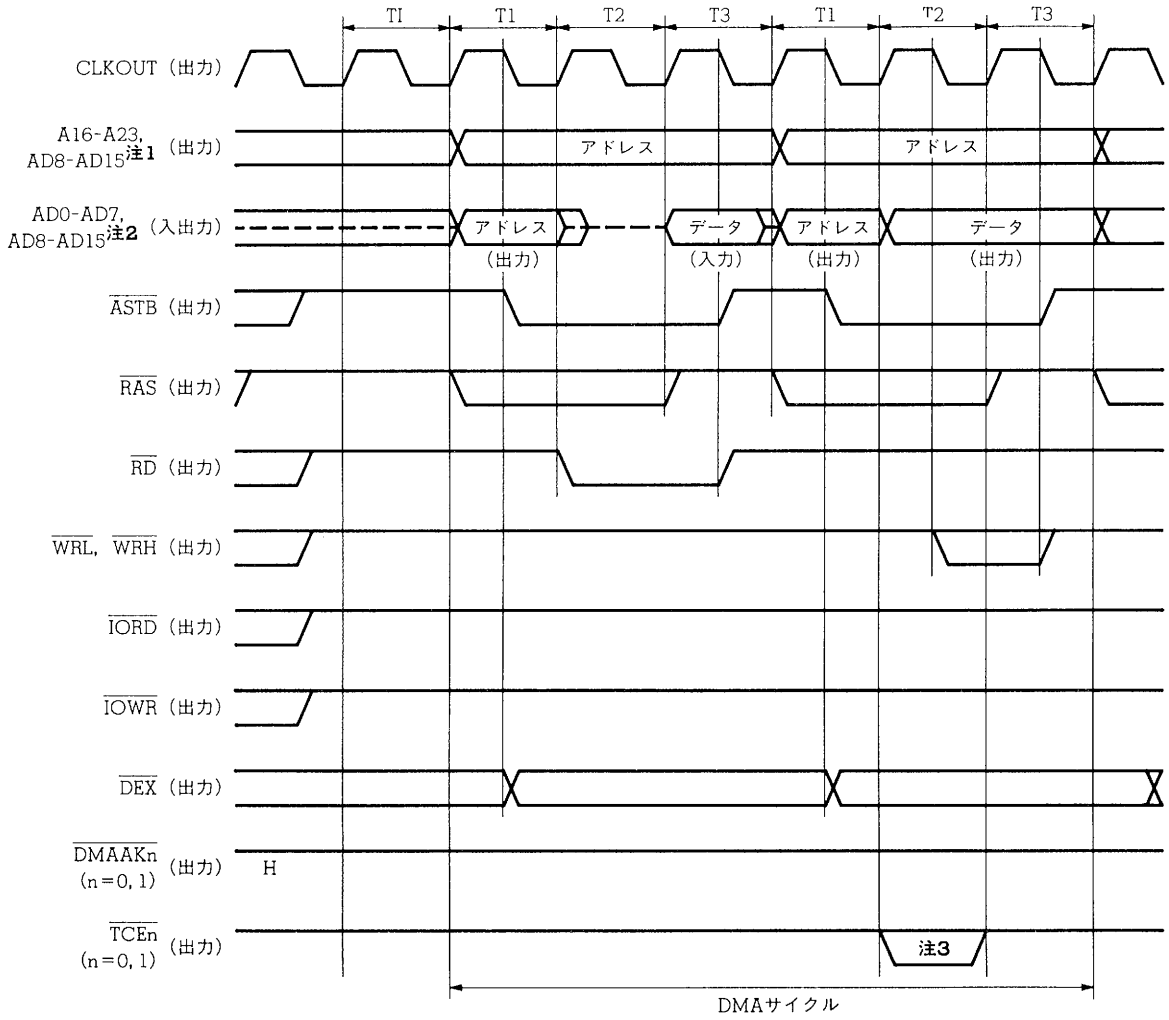
2．外部データ・バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－21　メモリ↔メモリ（1/2）

（a）シングルトランスファ・モード，シングルステップ・モード



注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

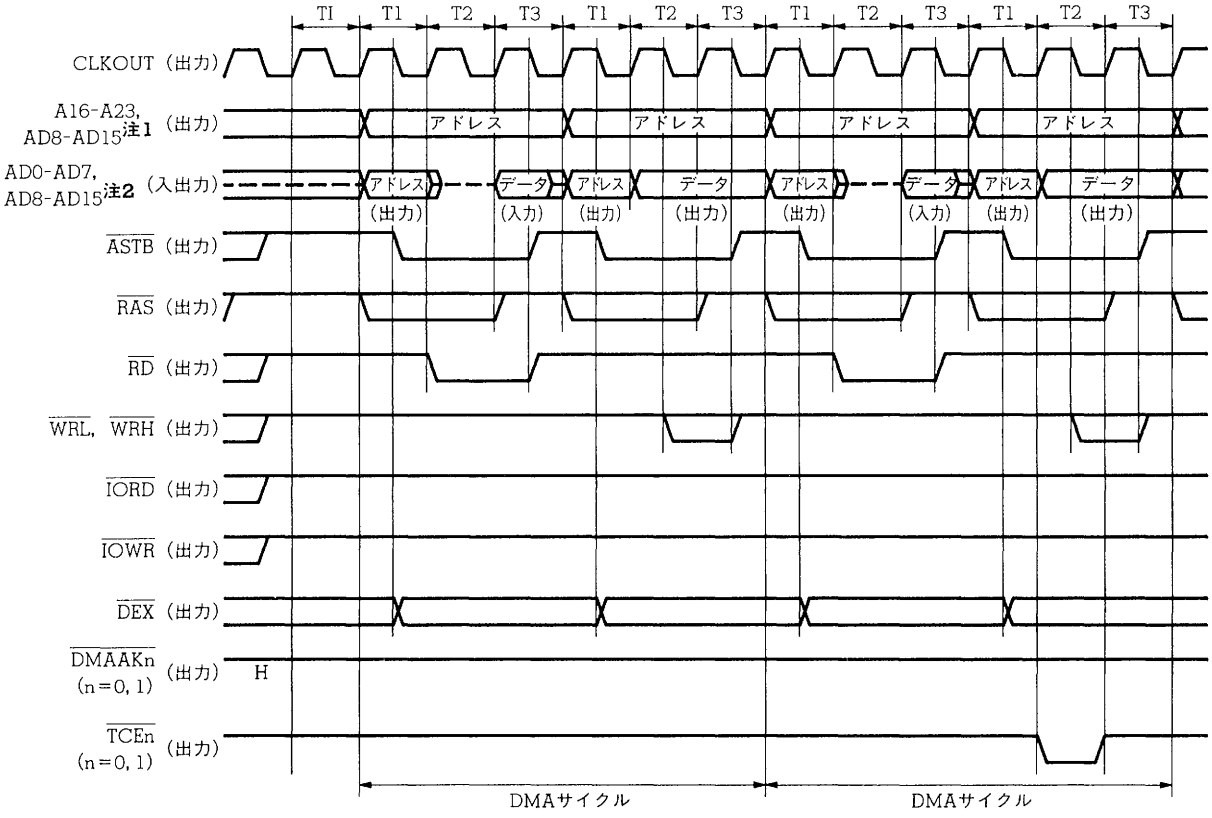
3．TCボローが発生したとき，ロウ・レベルを出力します。

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－21　メモリ↔メモリ（2/2）

（b）バースト・モード（２回転送）



注1．外部データ・バス幅８ビットのとき

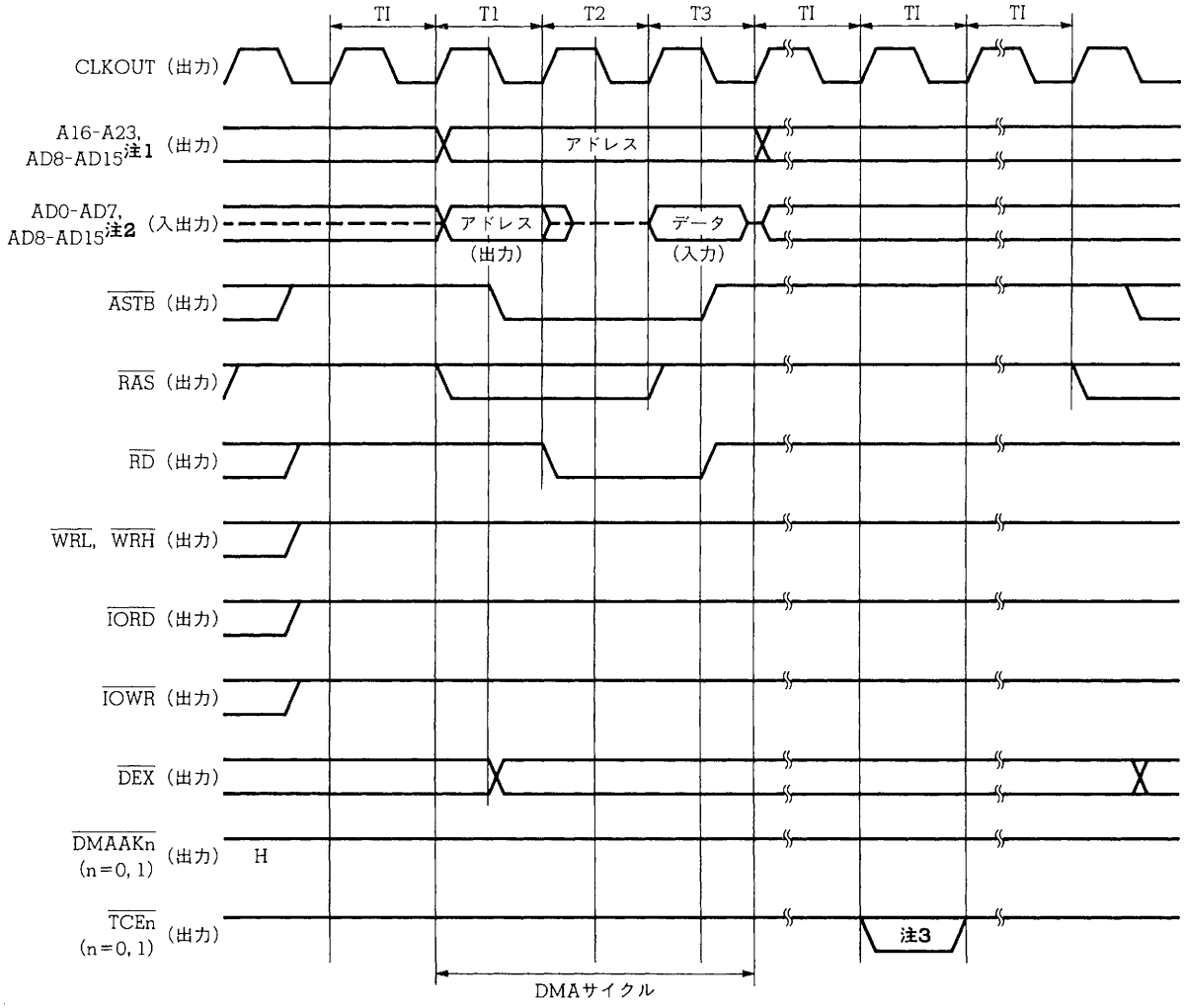
2．外部データ・バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－22　メモリ → SFR（メモリ・アクセス）（1/2）

（a）シングルトランスファ・モード，シングルステップ・モード



注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

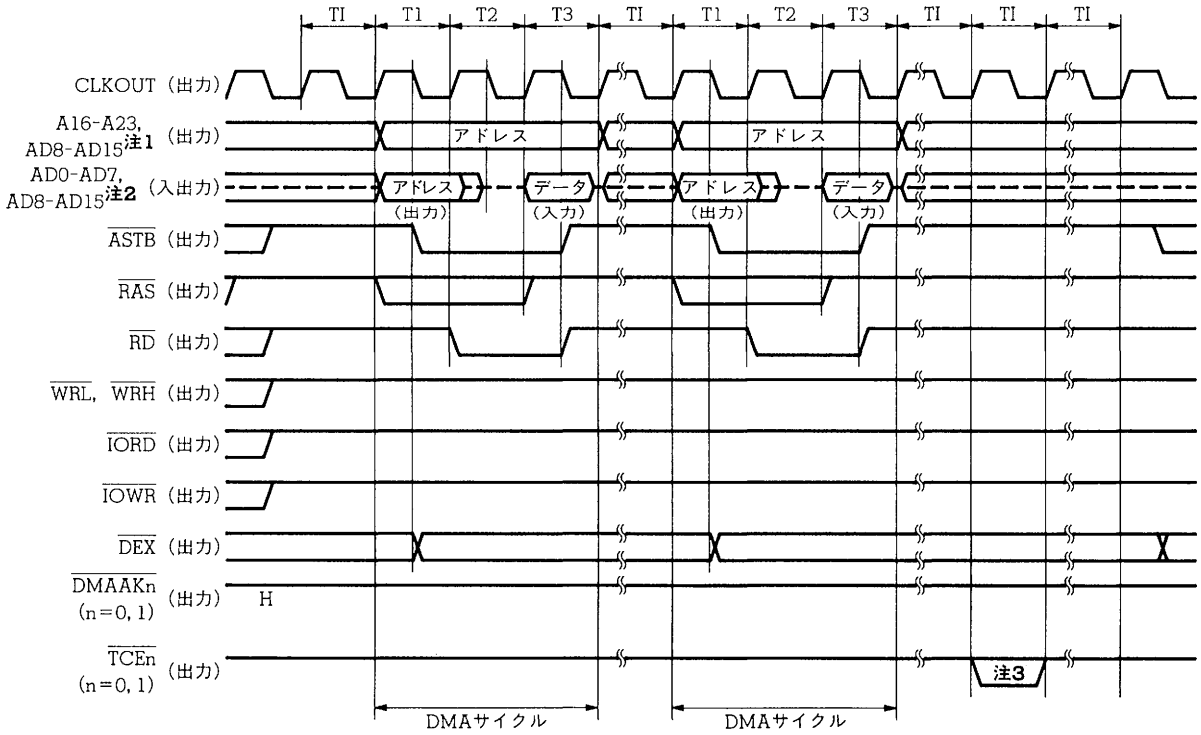
3．TCボローが発生したとき，ロウ・レベルを出力します。

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－22　メモリ→SFR（メモリ・アクセス）（2/2）

（b）バースト・モード（２回転送）



注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

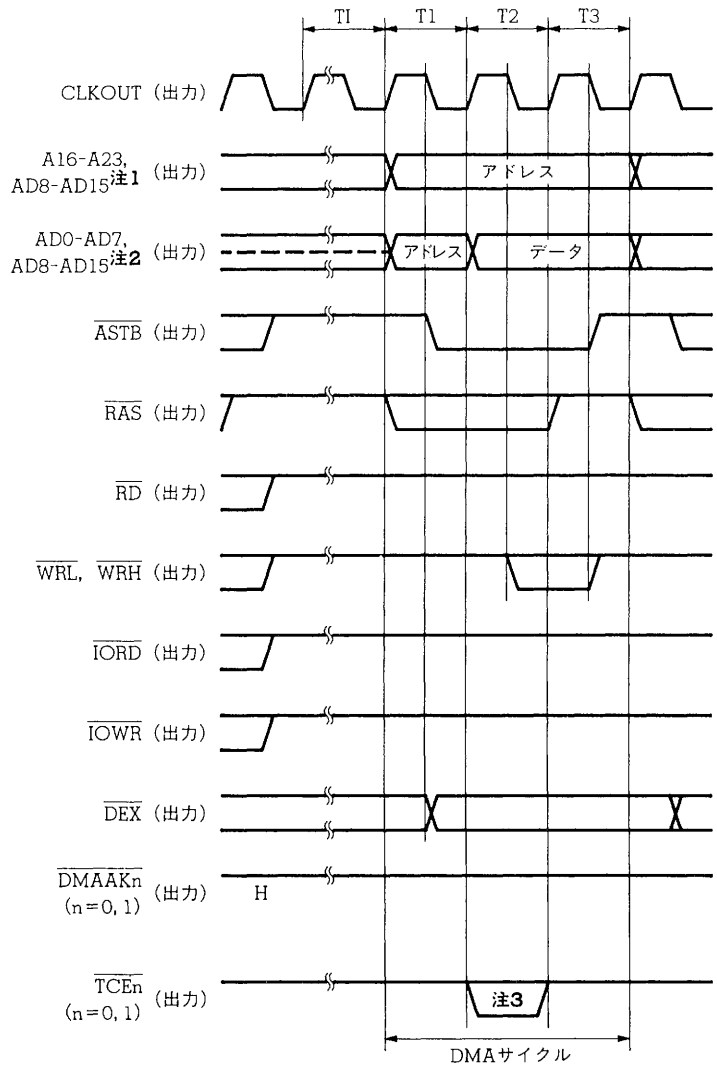
3．TCボローが発生したとき，ロウ・レベルを出力します。

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－23　SFR（メモリ・アクセス）→メモリ（1/2）

（a）シングルトランスファ・モード，シングルステップ・モード



注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

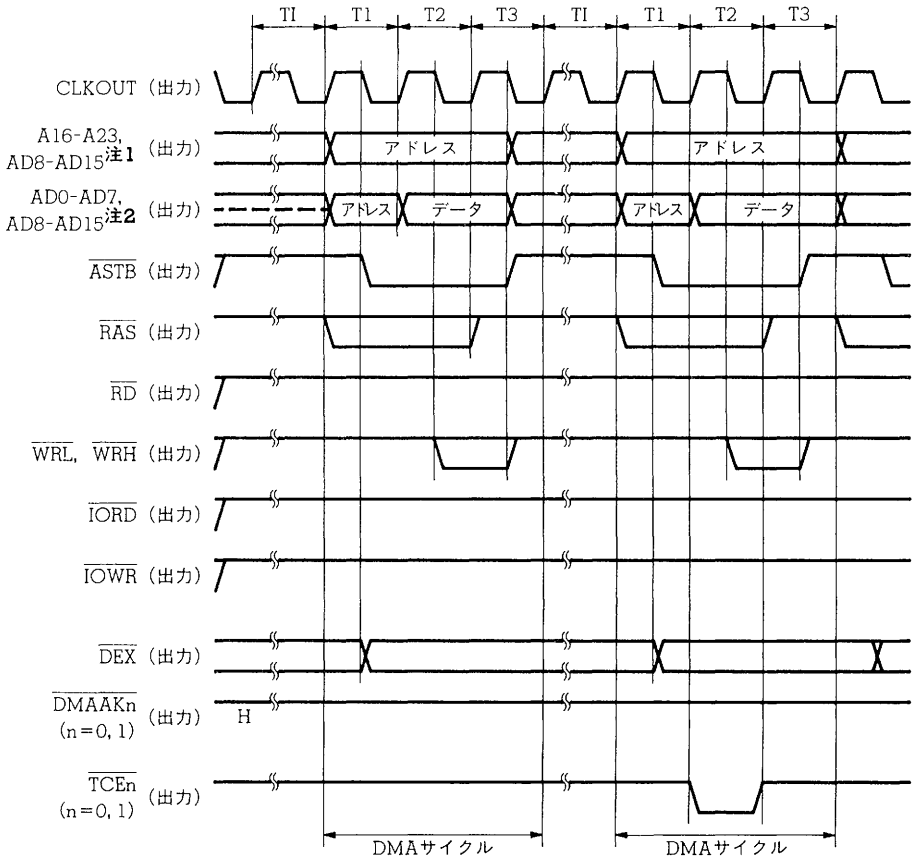
3．TCボローが発生したとき，ロウ・レベルを出力します。

備考　破線はハイ・インピーダンスを示します。

保守／廃止

図６－23　SFR（メモリ・アクセス）→メモリ（2/2）

（b）バースト・モード（２回転送）



注1．外部データ・バス幅８ビットのとき

2．外部データ・バス幅16ビットのとき

備考　破線はハイ・インピーダンスを示します。

保守／廃止

6.11　注意事項

6.11.1　メイン・チャネル側DMAコントローラにおける転送対象指定の制限

メイン・チャネルを使用する以下のDMA動作モードでは，転送対象にSFRおよびメモリを指定する場合，次のような制限事項がありますので注意してください。

- インテリジェントDMAモード－1
- インテリジェントDMAモード－2
- ネクスト・アドレス指定モード
- メモリ－メモリ転送モード

（1）転送対象指定可能SFRについて

DMA転送対象に指定可能なSFRは，一部のレジスタにかぎられています（**6.3.2　転送対象（3）SFR**参照）。それ以外のSFR（割り込み要求制御レジスタ，割り込みマスク・フラグ・レジスタ，インサービス・プライオリティ・レジスタ，DMA関係の制御レジスタなど）をDMAの転送対象とした場合は，その後の正常動作は保証されません。

（2）マクロ・サービスでの指定との関係について

マクロ・サービスとDMA転送で，同時に同一のSFRまたはメモリを転送対象とすることはできません。

（3）外部８ビット・バス・モード使用の際の指定について

外部８ビット・バス・モードで使用する際には，16ビットSFR（**表３－５　特殊機能レジスター覧**参照）と外部メモリとの間でDMA転送を実行することはできません。

外部データ・バス幅とDMA転送対象との関係を表６－７に示します。

保守／廃止

表６－７　外部データ・バス幅とDMA転送対象

転送対象	バイト／ワード	偶アドレス／奇アドレス	外部バスのデータ幅	
			８ビット	16ビット
メモリ↔I/O	バイト		使用可	使用可
	ワード	偶アドレス	使用不可	使用可
		奇アドレス	使用不可	使用不可
メモリ↔SFR	バイト		使用可	使用可
	ワード	偶アドレス	使用不可	使用可
		奇アドレス	使用不可	使用不可
メモリ↔メモリ	バイト		使用可	使用可
	ワード	偶↔偶	使用不可 （バイト×２回を使用する）	使用可
		偶↔奇	使用不可 （バイト×２回を使用する）	使用不可 （外部メモリは偶アドレスで使用）
		奇↔偶	使用不可 （バイト×２回を使用する）	使用不可 （外部メモリは偶アドレスで使用）
		奇↔奇	使用不可 （バイト×２回を使用する）	使用不可 （外部メモリは偶アドレスで使用）

保守／廃止

6.11.2　２チャネル動作モード使用時の制限

(1) DMAコントロール・レジスタ（DMAC0，DMAC1）の指定について

DMAC0，DMAC1は，メイン・チャネルとサブチャネルで共通のレジスタとなります。したがって，次のような制限があります。

TMD1，TMD0ビット（7，6ビット）：転送モードの指定ビットです。
メイン・チャネルとサブチャネルとで同一の転送モードとなります。

Wビット（４ビット）：転送処理方法指定ビットです。
メイン・チャネルとサブチャネルとで同一の処理方法となります。

DIRビット（５ビット）：転送方向指定ビットです。
このビットでの指定は，メイン・チャネル側の転送方向の指定となります。サブチャネル側の転送方向は，DMAMレジスタのTS1，TS0ビット（0，1ビット）でメイン・チャネルとサブチャネルの転送方向が同方向であるか逆方向であるかを指定することにより決まります。

(2) DMA制御端子について

DMA制御端子は，各チャネルのメイン側が優先して使用されます。もし，一方のチャネルのメイン側のDMA要求ソースが端子を使用しないものであれば，他チャネルのサブ側の制御端子として使用することができます。

例1. チャネル0：TS1，TS0ビット＝01指定
→ メイン・チャネルが外部DMARQ0端子からのDMA要求
サブチャネルが外部DMARQ1端子からのDMA要求
チャネル1：TS1，TS0ビット＝10指定
→ メイン・チャネルが外部DMARQ1端子からのDMA要求
サブチャネルが外部DMARQ0端子からのDMA要求

このとき，チャネル０，チャネル１の各メイン・チャネルがDMA制御端子を使用するため，チャネル０，チャネル１の各サブチャネルはDMA制御端子を使用できません。

保守／廃止

例2. チャネル0：TS1，TS0ビット＝01指定

→ メイン・チャネルが外部DMARQ0端子からのDMA要求

サブチャネルが外部DMARQ1端子からのDMA要求

チャネル1：TS1，TS0ビット＝00指定

→ メイン・チャネルがシリアル・チャネル1受信割り込み（INTSR1）でのDMA要求

サブチャネルがシリアル・チャネル0送信割り込み（INTST0）でのDMA要求

このとき，チャネル1のメイン・チャネルはDMA制御端子を使用していませんので，チャネル0のサブチャネルはメイン・チャネルと同様にDMA制御端子を使用することができます。

6.11.3　リフレッシュ・サイクル自動挿入許可時の転送モード

リフレッシュ・サイクルの自動挿入を許可した状態（RFMレジスタのRFENビット＝1）でI/O-メモリ間のDMA転送を行う場合，シングルトランスファ・モードを使用してください。ディマンド・リリース・モードを指定すると，DMARQ入力の立ち下がりに続く最後のDMA転送サイクルが正常に発生しないことがあります。

6.11.4　UART受信割り込み要求をDMAの転送要求ソースとしている場合

DMA機能を使用してUART受信を行う場合，受信エラーが発生しても，受信割り込み要求に従って受信バッファ（RxB）から外部メモリへDMA転送が実行されてしまいます。したがってこの場合，DMA転送後に発生する受信エラー割り込み内で対処してください。

受信エラーを発生したデータは，外部メモリ領域のMAR＋TC＋1（デクリメント時）またはMAR－TC－1（インクリメント時）で示されるアドレスに格納されています。

なお，受信エラー割り込み内で，エラーの発生した受信データを操作するときは，次のDMA転送が発生しないようDMA禁止状態にしてから行ってください。

〔メ　モ〕

保守／廃止

第7章　シリアル・インタフェース機能

V55PIは２チャネルのシリアル・インタフェース・ユニット（ch0，ch1）を備えています。
V55PIがサポートする２つの通信プロトコルを次に示します。

(1) 調歩同期式　　UART
(2) クロック同期式　CSI ―― SBI：２線式シリアル・バス・インタフェース
　　IOE：I/O拡張用３線式シリアル・インタフェース

7.1 特　　徴

○２つの通信プロトコルをサポート
○シリアル・チャネル：２チャネル
○ウエーク・アップ機能
○専用ボー・レート・ジェネレータ内蔵
○送受信完了によるDMA要求発生（送受信データのDMA転送可能）

7.2 通信プロトコル

UARTはスタート／ストップ・ビットによりデータ同期をとる調歩同期式のシリアル・インタフェースで，従来のシングルチップ・マイコンのUART機能を強化したものです。

CSI（クロックト・シリアル・インタフェース）はクロックを送受して同期をとるクロック同期式のシリアル・インタフェースです。CSIは，NECシングルチップ・マイクロコンピュータにおける標準シリアル・バス・インタフェース仕様のサブセットとなっており，I^2C機能はサポートしていません。また，ウエーク・アップ解除動作はマクロ・サービスを用いて実現する方式になっています。

表7－1　サポート・プロトコル

シリアル・インタフェース・ユニット	プロトコル		
	クロック周期式（CSI）		調歩同期式
	SBI	IOE	(UART)
チャネル0	○	○	○
チャネル1	―	○	○

各チャネルは，UART機能，CSI機能のどちらかをプログラマブルに選択可能です。プロトコル選択は，

保守／廃止

プロトコル選択レジスタ（ASP）で行います。

7.2.1　UART

(1) 基本構成

○制御端子

●シリアル・データ入力端子（RxD）

●シリアル・データ出力端子（TxD）

●送信許可制御端子　　（$\overline{CTS}$）

●送信クロック出力端子　　（TxC）…チャネル0のみ

○転送制御部，送信用，受信用の各8ビット・シフト・レジスタ

○UART送信バッファ（TxB）

○受信バッファ（RxB）

○送受信独立のBRGレジスタ（TxBRG，RxBRG）

(2) 特　　徴

○転送速度　95-390 Kbps（システム・クロックϕ＝12.5 MHz時）

123-500 Kbps（システム・クロックϕ＝16 MHz時）

○全二重動作可能

○専用（送信用，受信用）ボー・レート・ジェネレータ内蔵

○ウエーク・アップ機能

○0パリティ機能

○パリティ・エラー検出

○フレーミング・エラー検出

○オーバラン・エラー検出

○UART割り込みソース3種

●UART受信エラー割り込み（INTSER0，INTSER1）

●UART受信割り込み（INTSR0，INTSR1）

●UART送信割り込み（INTST0，INTST1）

○マクロ・サービス機能

●UART受信割り込み（INTSR0，INTSR1）

●UART送信割り込み（INTST0，INTST1）

7.2.2　CSI

(1) 基本構成

○制御端子

保守／廃止

●シリアル・データ入力端子（SI/SB0）…SB0はチャネル0のみ

●シリアル・データ出力端子（SO/SB1）…SB1はチャネル0のみ

●シリアル・クロック端子（$\overline{SCK}$）

○転送制御部，送受信共用の8ビット・シフト・レジスタ（SIO）

○受信バッファ（RxB）

○送受信共用のBRGレジスタ（TxBRG）

(2)特　　徴

○転送速度　最大3.125 Mbps（システム・クロックϕ＝12.5 MHz時）

最大4.0 Mbps（システム・クロックϕ＝16 MHz時）

○半二重通信

○データ長は8ビット単位

○外部，内部クロック選択可能

○データのMSBファースト／LSBファースト選択可能

○SBIモード（2線式のNEC方式シリアル・バス）…ch0のみ

●アドレス／コマンド／データ判断機能

●アドレスによるチップ・セレクト機能

●ウエーク・アップ機能

●アクノリッジ信号（$\overline{ACK}$）制御機能

●ビジィ信号（$\overline{BUSY}$）制御機能

7.2.3　制御レジスタ

(1) プロトコル選択レジスタ（ASP）

シリアル・インタフェース・ユニット各チャネルの動作プロトコルを指定するレジスタです。

このレジスタで選択したプロトコルに従って，各チャネルの端子機能およびレジスタ機能が切り替わります。

8/1ビットのメモリ・アクセスによってリード／ライトができます。

リセット後の値は，00Hになります。

保守／廃止

	7	6	5	4	3	2	1	0
ASP	0	0	0	0	0	0	P(ch1)	P(ch0)

動作プロトコルの指定

P	動作プロトコル
1	調歩同期式（UART）
0	クロック同期式（CSI）

備考　▒ はリセット後の状態を示します。

なお，チャネル0（ch0）のクロック同期式動作時のモード選択（SBIモードとIOEモードの切り替え）は，クロック・シリアル・インタフェース・モード・レジスタ（CSIM）で行います。

(2) 各プロトコルの制御レジスタ

UARTとCSIの制御レジスタは同一のメモリ・アドレスにマッピングされています。選択したプロトコルに応じて機能が切り替わります。

表7-2　制御レジスター覧

レジスタ・アドレス	選択したプロトコル	
	調歩同期式P＝1	クロック同期式P＝0
0FFF73H	UARTM0（動作モード）	CSIM0（動作モード）
0FFF7BH	UARTM1（動作モード）	CSIM1（動作モード）
0FFF74H	UARTS0（ステータス）	SBIC0（制御）
0FFF7CH	UARTS1（ステータス）	未使用[注1]
0FFF75H	TxB0（送信バッファ）	SIO0（シフト・レジスタ）[注2]
0FFF7DH	TxB1（送信バッファ）	SIO1（シフト・レジスタ）[注2]
0FFF76H	RxB0（受信バッファ）	RxB0（受信バッファ）
0FFF7EH	RxB1（受信バッファ）	RxB1（受信バッファ）
0FFF72H	PRS0（プリスケーラ）	PRS0（プリスケーラ）
0FFF7AH	PRS1（プリスケーラ）	PRS1（プリスケーラ）
0FFF70H	TxBRG0（送信BRG）	TxBRG0（送信BRG）
0FFF78H	TxBRG1（送信BRG）	TxBRG1（送信BRG）
0FFF71H	RxBRG0（受信BRG）	未使用[注3]
0FFF79H	RxBRG1（受信BRG）	

注1. ch1の3線式シリアルI/Oモード（IOEモード）ではSBICレジスタは使用しません。

2. クロック同期式モードでは送信シフト・レジスタ（SIO）に直接送信データを書き込みます。

3. クロック同期式モードでは送受信ともデータの転送速度は同一です。送信側のボー・レート・ジェネレータ（BRG）のみを使用します。

保守／廃止

(3) 各プロトコルの制御端子

UARTとCSIの制御端子は同一のポート端子に割り当てられています。選択したプロトコルに応じて機能が切り替わります（ただし，ポートをコントロール機能（PMC3n＝1）に設定した場合）。

表7－3　制御端子一覧

制　御　端　子	選択したプロトコル	
	調歩同期式P＝1	クロック同期式P＝0
送信データ出力端子	TxD0	SO0/SB0 (ch0)
	TxD1	SO1 (ch1)
受信データ出力端子	RxD0	SI0/SB1 (ch0)
	RxD1	SI1 (ch1)
クロック端子	TxC	$\overline{SCK0}$ (ch0)
送信制御端子	$\overline{CTS0}$	未使用 (ch0)
クロック／制御端子	$\overline{CTS1}$	$\overline{SCK1}$ (ch1)

保守／廃止

7.3　ボー・レート・ジェネレータ

V55PIは，送信専用および受信専用のシリアル・クロックを発生するボー・レート・ジェネレータ（BRG）を内蔵しています。

このためUARTでは，送信と受信おのおの任意のボー・レートを設定して動作することが可能です（CSIでは送受信のボー・レート・ジェネレータは同一です）。

7.3.1　構　　成

ボー・レート・ジェネレータは，送受信のシリアル・クロックを発生する専用の８ビット・タイマです。

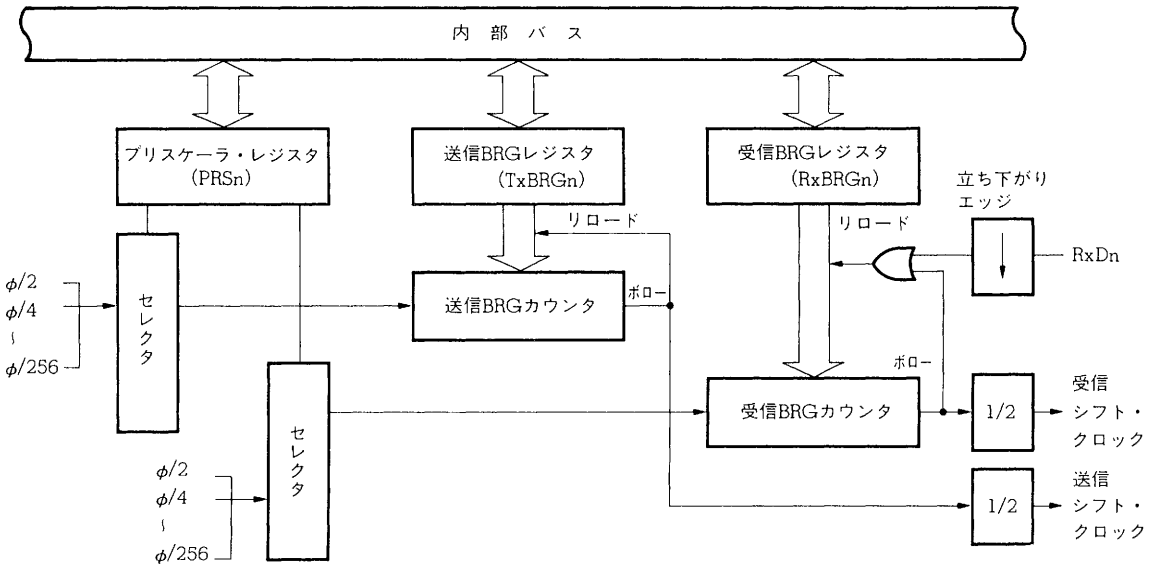
送受信各BRGカウンタのボロー出力を２分周してシリアル・クロックとして用います。

各BRGカウンタの１周期は，BRGレジスタ（TxBRG，RxBRG）にプログラムでカウンタ値（0-255）を設定することで指定します。

各BRGカウンタの入力クロックに，プリスケーラ・レジスタ（PRS）により内部システム・クロックの分周出力を指定します。

ボー・レート・ジェネレータのブロック図を図７-１に示します。

図７-１　ボー・レート・ジェネレータのブロック図



備考　n＝0 または 1

保守／廃止

7.3.2　動　　作

ボー・レート・ジェネレータは選択したプロトコルにより次のように動作が異なります。

(1) UARTでの動作

受信BRGはスタート・ビットを検出後にBRGレジスタ（RxBRG）から受信BRGカウンタに値をプリセットし，シリアル・クロックと受信データの同期をとります。

送信BRGは上記受信動作には無関係にシリアル・クロックを発生します。

なお，UARTにおけるシリアル受信データRxDのサンプリング（スタート・ビット検出）には，システム・クロックϕを使用しています。

注意　UARTでは，シリアル・クロックと受信データの位相ずれを抑えるため，受信BRGレジスタには7以上の値を設定して使用してください。

(2) CSIでの動作

受信BRGは使用しません。

シリアル・クロックに内部クロックを選択している場合には，送信BRGの出力クロックが送受信のシリアル・クロックになります。

ボー・レートの計算式は，次のようになります。

〔ボー・レートの計算式〕

$$\frac{\phi}{2^n}\times\frac{1}{(M+1)}\times\frac{1}{2}=\text{シリアル・クロック周波数}$$

$$\text{ボー・レート}=\frac{\phi}{2^{n+1}\ (M+1)}\ \text{〔bps〕}$$

ϕ　：システム・クロック周波数〔Hz〕

M　：8ビットBRGレジスタ設定値

（0 ≦ M ≦ 255，ただしUARTでは7 ≦ M ≦ 255）

$\phi/2^n$：プリスケーラ・レジスタ（PRS）で選択したBRGへの入力クロック

（1 ≦ n ≦ 8；nはPRSレジスタの該当ビットへの設定値＋1）

内部クロック12.5 MHz，16 MHzそれぞれの場合の各標準ボー・レートに対応するBRGレジスタ（TxBRG，RxBRG）への設定値と誤差（%）を表7-4に示します。

保守／廃止

表7-4　BRGレジスタへの設定値

(1) ϕ＝12.5 MHzの場合

ボー・レート（bps）	n	BRGレジスタ設定値(M)	誤差（%）
150	8	162	−0.15
300	7	162	−0.15
600	6	162	−0.15
1200	5	162	−0.15
2400	4	162	−0.15
4800	3	162	−0.15
9600	2	162	−0.15
19200	1	162	−0.15
38400	1	80	+0.47
76800	1	40	−0.76
153600	1	19	+1.72
390625	1	7	0
1.5625 M	1	1注	0
3.125 M	1	0注	0

注　UARTで設定できるのは，M ≧ 7のみです。したがって，M＝0，1は設定できません。

(2) ϕ＝16 MHzの場合

ボー・レート（bps）	n	BRGレジスタ設定値(M)	誤差（%）
150	8	207	−0.16
300	7	207	−0.16
600	6	207	−0.16
1200	5	207	−0.16
2400	4	207	−0.16
4800	3	207	−0.16
9600	2	207	−0.16
19200	1	207	−0.16
38400	1	103	−0.16
76800	1	51	−0.16
153600	1	25	−0.16
500000	1	7	0
1.0 M	1	3注	0
4.0 M	1	0注	0

注　UARTで設定できるのは，M ≧ 7のみです。したがって，M＝0，3は設定できません。

保守／廃止

7.3.3　制御レジスタ

(1) プリスケーラ・レジスタ（PRS0, PRS1）

ボー・レート・ジェネレータへの入力クロックを指定する8ビットのレジスタです。

8/1ビットのメモリ操作命令で，このレジスタのリード／ライトが可能です(PRS0はch0に，PRS1はch1に対応します)。

リセット後の値は，00Hになります。

	7	6	5	4	3	2	1	0
PRS0 PRS1	0	0	TSK2	TSK1	TSK0	RSK2	RSK1	RSK0

受信シリアル・クロック発生回路への入力クロックの指定

RSK2	RSK1	RSK0	入力クロック周波数
0	0	0	$\phi/2$
0	0	1	$\phi/2^2$
0	1	0	$\phi/2^3$
0	1	1	$\phi/2^4$
1	0	0	$\phi/2^5$
1	0	1	$\phi/2^6$
1	1	0	$\phi/2^7$
1	1	1	$\phi/2^8$

送信シリアル・クロック発生回路への入力クロックの指定

TSK2	TSK1	TSK0	入力クロック周波数
0	0	0	$\phi/2$
0	0	1	$\phi/2^2$
0	1	0	$\phi/2^3$
0	1	1	$\phi/2^4$
1	0	0	$\phi/2^5$
1	0	1	$\phi/2^6$
1	1	0	$\phi/2^7$
1	1	1	$\phi/2^8$

ϕ：内部システム・クロック周波数

備考　□はリセット後の状態を示します。

保守／廃止

(2) 送信ボー・レート・ジェネレータ・レジスタ（TxBRG0, TxBRG1）

送信ボー・レートのカウント値を設定する８ビットのレジスタです。8/1ビットのメモリ操作命令でこのレジスタの書き込み／読み出しが可能です。

TxBRG0はチャネル０に，TxBRG1はチャネル１に対応し，クロックト・シリアル・インタフェース・モードでは送受信両方のボー・レート指定を行います。

リセット後の値は，不定となります。

(3) 受信ボー・レート・ジェネレータ・レジスタ（RxBRG0, RxBRG1）

受信ボー・レートのカウント値を設定する８ビットのレジスタです。8/1ビットのメモリ操作命令で，このレジスタの書き込み／読み出しが可能です。

RxBRG0はチャネル０に，RxBRG1はチャネル１に対応し，クロックト・シリアル・インタフェース・モードではこのレジスタは使用しません。

リセット後の値は，不定となります。

注意　送信，受信中には，送信／受信BRGレジスタを読みださないでください。シフト・クロックの周期が一時的に狂ってしまいます。

保守／廃止

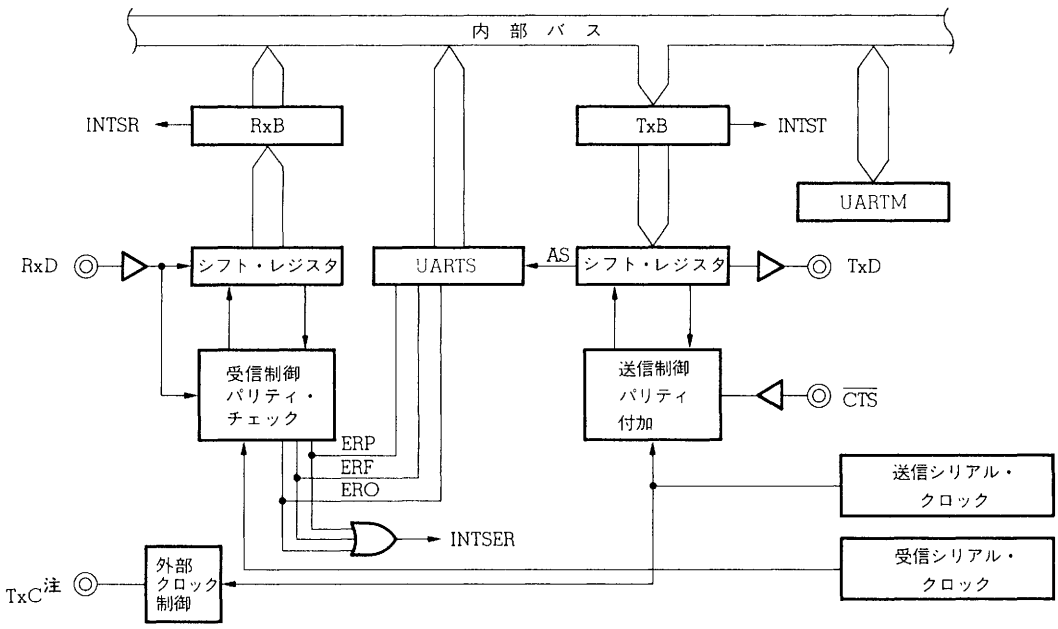
7.4　UART動作

7.4.1　構　　成

V55PIは，専用のボー・レート・ジェネレータを内蔵するUARTを備えています。図7-2にUARTのブロック図を示します。

UARTは，シリアル・データ入力端子（RxD），シリアル・データ出力端子（TxD），シリアル・クロック出力端子[注]（TxC），送信許可状態の制御入力端子（$\overline{CTS}$）の各端子と転送制御部，送信用および受信用の8ビット・シフト・レジスタ，UARTモード・レジスタ（UARTM），UARTステータス・レジスタ（UARTS），UART送信バッファ（TxB），受信バッファ（RxB）および送信，受信用ボー・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フト・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）。

図7-2　UARTのブロック図



注　チャネル0のみ

保守／廃止

7.4.2　送信動作

(1) 許可状態

設定：UARTモード・レジスタ（UARTM）のTxEビット＝1，かつ$\overline{\text{CTS}}$端子＝0

動作：①　シフト・レジスタからTxD端子を通してデータが送られます。

②　1フレーム分のデータをすべて送信すると，TxBバッファのデータがシフト・レジスタへ転送され，送信割り込みが発生します。

③　引き続き，シフト・レジスタの内容がTxD端子から転送されます。

(2) 禁止状態

設定：$\overline{\text{CTS}}$端子＝1

または，UARTMレジスタのTxEビット＝0

動作：TxBバッファからシフト・レジスタへのデータ転送が禁止されます（シフト・レジスタが空の場合でも同様）。

注意　送信中に送信禁止操作を行った場合，送信中のデータは最後まで送られます。

(3) 送信の起動

送信の起動方法は次の2通りがあります。

- UART送信バッファが空の状態で送信許可状態に設定する方法
  UART送信割り込み要求が発生します。割り込み処理内で，送信バッファに送信データの書き込みを行います。
- 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設定する方法
  送信バッファ内で保持されていたデータが送信されます。

すでに送信許可状態のときは，送信バッファに送信データの書き込みを行います。

なお，$\overline{\text{RESET}}$入力により，送信バッファは空の状態になります。このとき，送信許可状態にするとUART送信割り込み要求が発生します。

(4) ポート制御との相違

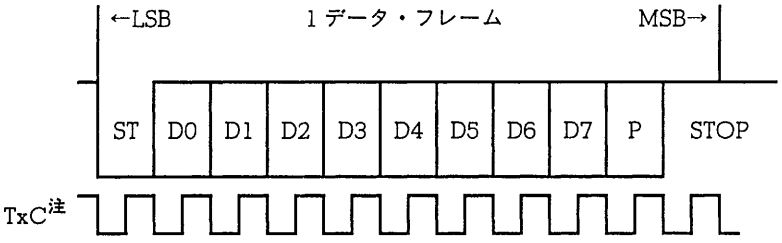
$\overline{\text{CTS}}$端子を，ポート3モード・コントロール・レジスタ(PMC3)で，ポート・モードに設定(PMC33＝0またはPMC36＝0)すると，$\overline{\text{CTS}}$端子の値にかかわらずUARTMレジスタのTxEビットの設定だけで送信動作の許可／禁止を制御できます。

保守／廃止

(5) 基本的なデータ・フォーマット

送信データの基本的なデータ・フォーマットは図7-3に示すようにスタート・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7-3　基本的なデータ・フォーマット



スタート・ビット　　(ST)　：1ビット
キャラクタ・ビット　　(D0-D7)　：7/8ビット
パリティ・ビット　　(P)　：奇数/偶数/0/付加しない
ストップ・ビット　　(STOP)　：1/2ビット

注　チャネル0だけ

7.4.3　受信動作

(1) 許可状態

設定：UARTモード・レジスタ（UARTM）のRxEビット＝1

注意1.　ただし，UARTMレジスタのWUPビット＝1のときは，RxEビットの値にかかわらず受信許可状態になります（7.4.4　ウエーク・アップ機能参照）。

2.　RxD端子入力がロウ・レベルの状態で受信を許可すると，そのロウ・レベルをスタート・ビットと判断し，受信を開始することがあります。したがって，必ずRxD端子入力がハイ・レベルの状態で受信を許可してください。

動作：①　RxD端子入力の立ち下がり信号エッジを検出すると，受信BRGカウンタをリセットしてダウン・カウントを開始し，受信を開始します。

②　1フレーム分のデータをすべて受信すると，シフト・レジスタ内のデータがRxBバッファへ転送され，受信割り込みが発生します。

保守／廃止

(2) 禁止状態

設定：UARTMレジスタのRxEビット＝0

動作：受信データはシフト・レジスタへ入力されません。

注意　データ受信中に受信禁止操作を行った場合，受信中のデータは正常に受信できません。

(3) パリティ機能

UARTMレジスタのPRTY0，PRTY1ビットにより，パリティ機能の1つである“0パリティ”を選択すると，シリアル・データの送信時，送信データにパリティ・ビットとして無条件に“0”を付加します。シリアル・データ受信時には，受信データに付加されるパリティの状態に関係なくデータを受信し，パリティ・エラーは発生しません。“0パリティ機能”は，システムの電源立ち上がり時など，シリアル・データ送受信のフォーマットが確定していない状態での通信を行う場合に有効です。

7.4.4　ウエーク・アップ機能

シリアル・データのキャラクタ・ビットのあとに送られてくるウエーク・アップ・ビット（WUPT）の値により，UART受信割り込みの発生を制御することができます。

(1) 設　定

UARTモード・レジスタ（UARTM）のWUPビット＝1にすることにより，ウエーク・アップ機能を選択します。

(2) 動　作

ウエーク・アップ機能を選択した状態でデータの送信を行うと，受信側は受信禁止状態でもそのデータを受信できます。

次に，受信禁止状態のとき（RxEビット＝0）と許可状態のとき（RxEビット＝1）のそれぞれの動作を説明します。

●受信禁止状態のとき（RxEビット＝0）

① スタート・ビットの検出に同期して受信シフト動作を開始します。

② 1データ・フレーム分の受信を終了すると，受信データのウエーク・アップ・ビット・データが“1”であることをチェックします。

③ “1”であれば，受信シフト・レジスタの内容を受信バッファへ転送し，受信割り込みまたは受信エラー割り込みを発生します。

●受信許可状態のとき（RxEビット＝1）

① スタート・ビットの検出に同期して受信シフト動作を開始します。

保守／廃止

② 1データ・フレーム分の受信を終了すると，受信シフト・レジスタの内容を受信バッファへ転送し，受信割り込みまたは受信エラー割り込みを発生します。

(3) 特　長

ウエーク・アップ機能を用いることにより，図7-4のように複数のシステムをUARTでお互いに接続して，効率よくデータの送受信を行うことができます。

WUPTビット＝0のデータを受信する際にスレーブ・デバイスをウエーク・アップ状態にしておくと，ウエーク・アップ状態を解除したスレーブ・デバイス以外はシリアル受信割り込みを発生しません。したがって，ある1つのスレーブ・デバイスと送受信を行うとき，その他のスレーブ・デバイスはシリアル通信に関係なく動作させることができます。

ウエーク・アップ機能がないと，データを受信するごとに割り込みが発生するため，すべてのスレーブ・デバイスがシリアル通信ごとに影響を受けることになります。

(4) 使 用 例

V55PIがマスタとして特定のアドレスの相手にデータを送信する場合を説明します。

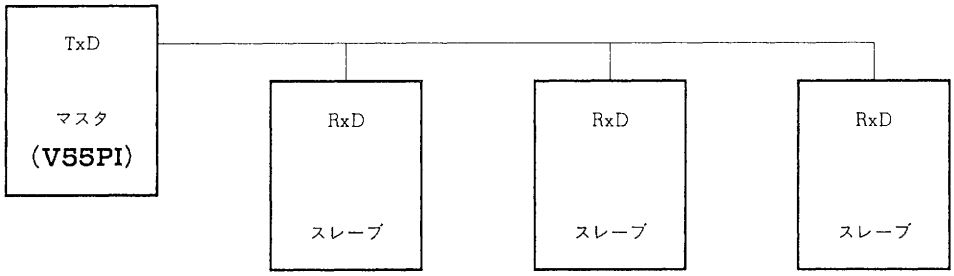
① マスタ側，複数のスレーブ側ともUARTMレジスタのWUPビット＝1，RxEビット＝0にします。

② マスタ側：WUPTビット＝1にしてアドレス・データを送出します。
　複数のスレーブ側：ウエーク・アップ機能を選択しているため受信割り込みを発生し，送出されたアドレス・データをソフトウエアでチェックします。

③ 特定のスレーブ：アドレスが自分自身であることを検出すると，UARTMレジスタのRxEビット＝1にします。
　マスタ側：データのWUPTビット＝0にして送信します。

④ 送信データのWUPTビット＝0なので，特定のアドレス以外のスレーブに割り込みが起こりません。

図7-4　複数のシステムを接続した例

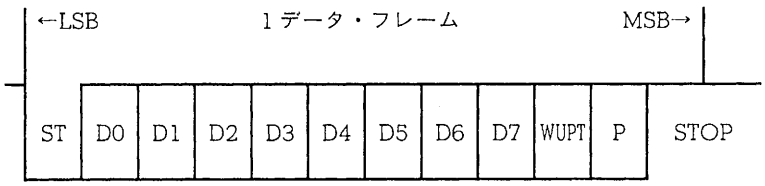


保守／廃止

(5) ウエーク・アップ機能を選択したときのデータ・フォーマット

ウエーク・アップ機能を選択したときのデータ・フォーマットは図7-5に示すようにスタート・ビット，キャラクタ・ビット，ウエーク・アップ・ビット（送信付加ビット・データ），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7-5　ウエーク・アップ機能を選択したときのデータ・フォーマット



スタート・ビット	(ST)	：1ビット
キャラクタ・ビット	(D0-D7)	：8ビット
ウエーク・アップ・ビット	(WUPT)	：1ビット
パリティ・ビット	(P)	：奇数/偶数/0/付加しない
ストップ・ビット	(STOP)	：1/2ビット

注意　ウエーク・アップ・ビット・データはパリティの計算データに含まれます。

7.4.5　制御レジスタ

UARTの制御レジスタとして，UARTモード・レジスタ（UARTM0，UARTM1），UARTステータス・レジスタ（UARTS0，UARTS1），UART送信バッファ（TxB），受信バッファ（RxB）があります。

(1) UARTモード・レジスタ（UARTM0，UARTM1）

UARTモード・レジスタの各ビットは次のような動作モードを指定します。

リセット後の値は，00Hになります。

保守／廃止

	7	6	5	4	3	2	1	0
UARTM0 UARTM1	TxE	RxE	PRTY1	PRTY0	CL/WUPT	SL	ASIE	WUP

ウエーク・アップ動作許可指定

WUP	指　定
0	使用しない
1	使用する

AS割り込み[注]機能の動作許可指定

ASIE	指　定
0	割り込み禁止
1	割り込み許可

ストップ・ビット長の指定

SL	ストップ・ビット長
0	1
1	2

キャラクタ長の指定

WUP	CL/WUPT	キャラクタ長	ウエーク・アップ・ビット
0	0	7ビット	使用しない
0	1	8ビット	使用しない
1	0	8ビット	0送出
1	1	8ビット	1送出

パリティ・ビットの指定

PRTY1	PRTY0	動　作
0	0	パリティなし
0	1	0パリティ
1	0	奇数パリティ
1	1	偶数パリティ

受信許可の指定

RxE	動　作
0	受信禁止状態
1	受信許可状態

送信許可の指定

TxE	$\overline{\text{CTS0}}$	動　作
0	×	送信禁止状態
1	0	送信許可状態
1	1	送信禁止状態

注　AS（All Sent）割り込み：オール・セント状態検出割り込み（**7.4.6 (3)** 参照）

注意　UARTが送受信中に，ASIEフラグを除くUARTMレジスタのビット・データの変更はできません。

備考　□ はリセット後の状態を示します。

保守／廃止

CL/WUPT キャラクタ長またはウエーク・アップ・ビット・データの指定

CL/WUPTビットはWUPビット（ウエーク・アップ機能選択ビット）の内容により機能が変化します。送信ウエーク・アップ・ビット・データはパリティ機能を使用する場合には，送信パリティの計算データに含まれます。

(a) ウエーク・アップ機能を使用しない場合（WUP＝0）

CLが有効になり，1データ・フレームのキャラクタ長を指定します。

CL	キャラクタ長
0	7ビット
1	8ビット

(b) ウエーク・アップ機能を使用する場合（WUP＝1）

WUPTが有効になり，送信するウエーク・アップ・ビット・データを指定します。

ウエーク・アップ・モードではキャラクタ長は，8ビットに固定です。

WUPT	ウエーク・アップ・ビット・データ
0	送信データのウエーク・アップ・ビット・データ位置に0を送出
1	送信データのウエーク・アップ・ビット・データ位置に1を送出

PRTY1 **PRTY0** パリティ・ビットの指定

0パリティの場合，送信側はパリティ・ビット0で送信します。受信側は受信時のパリティ・エラーは発生しません。

RxE 受信許可制御の指定

受信許可状態／禁止状態の指定をします。WUPビット（ウエーク・アップ動作の許可選択ビット）の内容により機能が変化します。

(a) ウエーク・アップ機能を使用しない場合（WUP＝0）

RxE	動作
0	受信禁止状態
1	受信許可状態

(i) 受信禁止状態に設定した場合（RxE＝0）

受信データはシフト・レジスタに入力されません。

保守／廃止

(ii) 受信許可状態に設定した場合（RxE＝1）

スタート・ビットの検出に同期して受信データのシフト・レジスタへの入力を開始し，1データ・フレーム分の受信の終了により，受信シフト・レジスタの内容を受信バッファ(RxB)に転送します。また，受信シフト・レジスタから受信バッファへの転送に同期して，UART受信割り込み/UART受信エラー割り込みを発生します。

(b) ウエーク・アップ機能を使用する場合（WUP＝1）

(i) 受信禁止状態に設定した場合（RxE＝0）

スタート・ビットの検出に同期して受信シフト動作を開始し，1データ・フレーム分の受信を終了し，受信データのウエーク・アップ・ビット・データが“1”であることをチェックします。“1”であれば，(a)の(ii)と同様，受信シフト・レジスタの内容を受信バッファに転送し，UART受信割り込み／UART受信エラー割り込みを発生します。

(ii) 受信許可状態に設定した場合（RxE＝1）

受信データのウエーク・アップ・ビット・データがUARTステータス・レジスタ（UARTS）のWUPRフラグにロードされる機能を除いて(a)の(ii)の動作と同じです。

TxE 送信許可制御の指定

$\overline{\text{CTS}}$端子と兼用のポートを，ポート3モード・コントロール・レジスタ（PMC3）で，ポート・モードに設定（ビット3，6＝0）すると，TxEビットのみで送信動作の許可／禁止を制御できます。

保守／廃止

(2) UARTステータス・レジスタ（UARTS0, UARTS1）

UARTSレジスタは，1キャラクタ分の受信終了時の受信エラー・ステータスを示します。エラー・フラグの内容は，データの受信ごとに更新されます。

リセット後のUARTSレジスタの値は20Hになります。

UARTS0
UARTS1

7	6	5	4	3	2	1	0
AS	WUPR	TxBE	RxBF	0	ERP	ERF	ERO

オーバラン・エラー・フラグ

ERO	動　　作
0	受信バッファからのデータを読み出し後，次のデータ受信が完了したとき
1	受信バッファから前のデータを引き取る前に，次のデータ受信が完了したとき

フレーミング・エラー・フラグ

ERF	動　　作
0	次の受信完了でストップ・ビットが検出されたとき
1	ストップ・ビットが検出されないとき

パリティ・エラー・フラグ

ERP	動　　作
0	次のパリティで送信パリティと受信パリティが一致したとき
1	送信パリティと受信パリティが一致しないとき

受信バッファ・フル・フラグ

RxBF	動　　作
0	受信バッファから受信データを読み出したことにより受信バッファに有効なデータがなくなったとき
1	シフト・レジスタが受信完了し，受信データが受信バッファに転送されたとき

次頁参照

保守／廃止

UARTS0
UARTS1

7	6	5	4	3	2	1	0
AS	WUPR	TxBE	RxBF	0	ERP	ERF	ERO

前頁参照

送信バッファ・エンプティ・フラグ

TxBE	動　　作
0	送信バッファへの送信データ書き込みにより，送信バッファがフルになったとき
1	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，送信バッファに空き状態ができたとき

受信ウエーク・アップ・ビット検出フラグ

WUPR	動　　作
0	受信したデータのウエーク・アップ・ビット＝0
1	受信したデータのウエーク・アップ・ビット＝1

全送信データ送出終了検出フラグ

AS	動　　作
0	送信バッファへの送信データ書き込み時
1	送信許可状態かつ送信バッファが空で送信端子よりデータの最終ビット送出が終了したとき

注意　**ERP, ERF, EROは命令により読み出し／書き込みが可能です。0 を書き込むことによりリセット(0)されます。ほかのステータス・フラグは読み出しのみ可能です。**

備考　□ はリセット後の状態を示します。

保守／廃止

ERO オーバラン・エラー・フラグ

オーバラン・エラーが発生したときにも，シフト・レジスタからバッファ（RxB）への受信データが転送されます。したがって，RxBには，最後に転送された受信データが書き込まれています。

なお，EROフラグをクリアしても，RxBに受信データが残っているかぎり，次フレームのデータ受信によりオーバランが再度発生します。オーバラン・エラー発生時は必ずRxBの読み出しも行ってください。

ERF フレーミング・エラー・フラグ

ERFフラグは1データ・フレームの受信ごとに内容が更新されます。

ERP パリティ・エラー・フラグ

ERPフラグは1データ・フレームの受信ごとに内容が更新されます。

"0パリティ機能"を選択している場合にはパリティ・エラーによるUART受信エラー割り込み(INTSER)は発生しません。パリティなしまたは"0パリティ"を指定した場合，ERPフラグは不定となります。ただし，パリティの演算結果をERPフラグにより検出することはできます。

設定	ERP	演算結果	ERP＝1のときの割り込み発生
パリティなし	0	データ中の1の数が奇数	発生しない
	1	データ中の1の数が偶数	
0パリティ	0	データ中の1の数が偶数	発生しない
	1	データ中の1の数が奇数	
奇数パリティ	0	エラーなし（データ中の1の数が奇数）	発生する
	1	エラーあり（データ中の1の数が偶数）	
偶数パリティ	0	エラーなし（データ中の1の数が偶数）	発生する
	1	エラーあり（データ中の1の数が奇数）	

RxBF 受信バッファ・フル・フラグ

受信バッファに有効な受信データが格納されたことを示すステータス・フラグです。

このフラグをポーリングすることで受信データの読み出しが可能であることを検出できます。

TxBE 送信バッファ・エンプティ・フラグ

送信バッファが空であることを示すステータス・フラグです。

このフラグをポーリングすることで送信データの書き込みが可能であることを検出できます。

WUPR ウエーク・アップ・ビット・データ・モニタ・フラグ

受信したウエーク・アップ・ビット・データがフラグの状態と対応するため，ウエーク・アップ・

保守／廃止

ビット・データをモニタすることができます。

UARTMレジスタでウエーク・アップ機能を指定したときに有効です。

AS　全送信データの送出終了検出フラグ

このフラグを利用してTxD端子よりデータ送出が完了したかどうかをモニタできます。モデムなどの応用でTxBに書き込んだデータがTxD端子よりすべて回路上に送出されたことを検出し，回路を切断するといった応用が可能です。

(3) 受信バッファ（RxB0，RxB1）

受信バッファは，受信データを保持する８ビットのバッファです。

７ビット・キャラクタの受信状態では最上位ビットに“0”が自動的に格納されます。

受信許可状態では，受信データは１データ・フレーム分の処理終了後，シフト・レジスタからバッファに転送されます。

また，同時にUART受信割り込み要求（INTSR0，INTSR1）が発生します。

(4) UART送信バッファ（TxB0，TxB1）

送信バッファは，送信データを保持する８ビットのバッファです。

送信許可状態では，送信バッファへのデータ書き込みにより，送信クロックに同期してバッファからシフト・レジスタにデータが転送され，送信動作が開始されます。

送信禁止状態で，送信バッファに送信データを書き込んでもシリアル送信は行われず，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によりバッファ内のデータがシフト・レジスタに伝えられて送信が開始されます。

送信バッファから送信シフト・レジスタへのデータ送出時に，UART送信割り込み要求（INTST0，INTST1）が発生します。

保守／廃止

7.4.6　割り込み要求

UARTからは，次の３種類の割り込み要求を発生します。

UART受信エラー割り込み	: INTSER0, INTSER1
UART受信割り込み	: INTSR0, INTSR1
UART送信割り込み（AS状態検出割り込み兼用）	: INTST0, INTST1

これら３種類の割り込み要求の優先順位は，次のようになります。

INTSERn＞INTSRn＞INTSTn（n＝0，1）

(1) UART受信エラー割り込み（INTSER0，INTSER1）

受信許可状態中で，受信エラーが発生した場合，UART受信エラー割り込みを発生します。

UART受信エラー割り込みの発生タイミングは，受信データの最初のストップ・ビットを検出後，受信シフト・レジスタ内のデータを受信バッファに転送するのに同期して発生します（受信割り込みINTSRnと同一タイミングで発生します）。

受信禁止状態中は，UART受信エラー割り込みは発生しません。また，０パリティ機能を選択している場合（PRTY1＝0，PRTY0＝1）には，パリティ・エラーによるUART受信エラー割り込みの発生はありません。

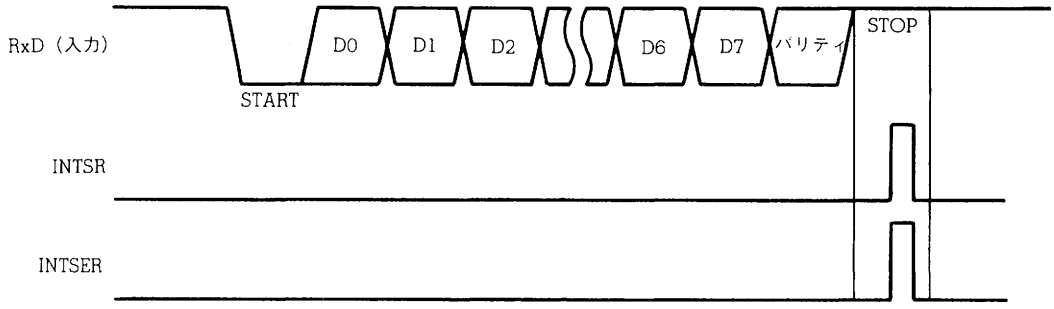
表７－５　受信エラーの要因

受信エラー	要　　因
パリティ・エラー	送信時のパリティ指定と受信データのパリティが一致しない
フレーミング・エラー	ストップ・ビットが検出されない[注]
オーバラン・エラー	受信バッファからデータを読み出す前に次のデータ受信完了

注　受信時には，ストップ・ビットは１ビットとしてのみ動作します。したがって，ストップ・ビットを２ビットとして指定している場合で，ストップ・ビットの２ビット目がロウ・レベルの場合は，エラーを発生しません（次のデータのスタート・ビットとして認識します）。

保守／廃止

図7-6　UART受信エラー割り込み発生タイミング



(2) UART受信割り込み（INTSRO，INTSR1）

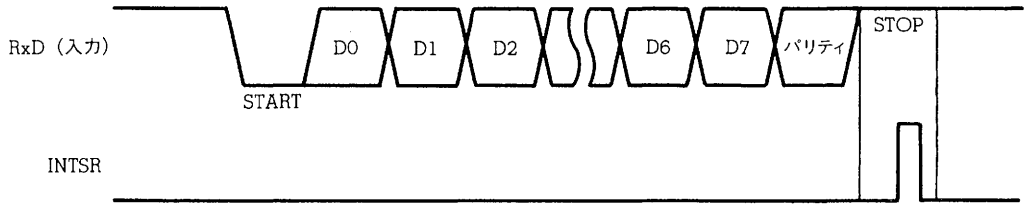
UART受信割り込みの発生タイミングは，受信許可状態中で受信データの最初のストップ・ビットを検出後，受信シフト・レジスタ内のデータを受信バッファに転送するのに同期して発生します。

UART受信割り込みは，受信エラーが起こった場合にも発生しますが，受信エラー割り込みの方が処理自体の優先順位は高くなります。受信エラー割り込みの処理終了後に，受信割り込み要求が受け付けられます。

受信禁止状態およびウエーク・アップ機能選択時（WUP＝1，RxE＝0）にウエーク・アップ条件が成立しないときには，UART受信割り込みの発生はありません。

UART受信割り込みが発生しないときには，受信バッファ（RxB）の内容は保持されます。

図7-7　UART受信割り込み発生タイミング



注意1．WUP＝1のときには，RxE＝1または受信ウエーク・アップ・ビット・データ＝1のときのみ受信状態となり，UART受信割り込み／受信エラー割り込みが発生します。

2．UART受信割り込みは，条件が成立する最初のタイミングでのみ割り込みが発生します。

3．受信バッファ（RxB）からの受信データ引き取りが遅れ，オーバラン・エラーが発生したときには，あとから受信したデータがオーバ・ライトする形で受信バッファに書き込まれます。この場合，再度受信割り込みが発生します。

4．UART受信割り込みは，割り込み制御レジスタ，ベクタなどの割り込み応答制御をCSI動作のシリアル送受信割り込み(INTCSIO，INTCSI1)と兼用するようになっていま

保守／廃止

す。プロトコル選択によってどちらの割り込み制御で用いるかが決定されます。

(3) UART送信割り込み（INTSTO, INTST1）

通常，送信動作時と，AS状態（オール・セント状態）[注]検出モード許可時に発生します。

注　送信バッファに次のデータを設定する前にシフト・レジスタ内のデータ送信も終了し，送信するデータがなくなった状態（送信バッファ，シフト・レジスタがともに空の状態）を指します。

(i) 送信動作時

送信許可状態(TxE＝1)かつ送信シフト・レジスタより1データ・フレームの最終データ(STOPビット)の送出が終了し，シフト・レジスタが空の状態になったことを検出して，送信バッファ（TxB）よりシフト・レジスタへのデータ転送が行われます。この結果，バッファが空になるとUART送信割り込み要求を発生します。

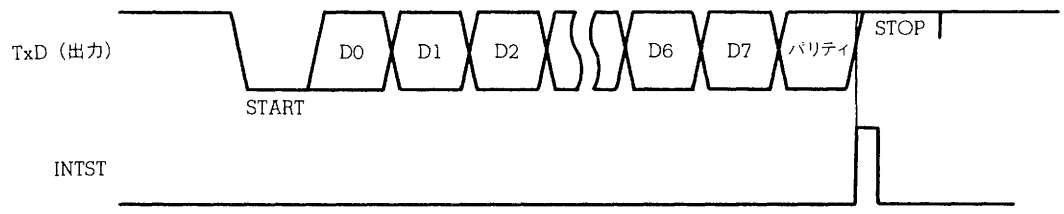
(ii) オール・セント状態検出モード許可のとき（UARTMレジスタ：ASIE＝1）

上記条件に加えて，送信許可状態かつ送信バッファが空のときに，送信端子TxDより送信データの最終データ（STOPビット）の送出が終了したときにもUART送信割り込みが発生します。また，このときUARTステータス・レジスタ（UARTS）のASフラグがセット(1)されます。

注意　送信許可状態のときにUART送信バッファ（TxB）が空の状態であることは，UART送信割り込み発生の要因となります。したがって，リセット解除後，TxEフラグを“0→1”にすることで送信割り込み要求を発生させ，これを利用して割り込み処理による送信バッファへのデータ転送を行うことが可能です。

図7-8　UART送信割り込み発生タイミング（1/2）

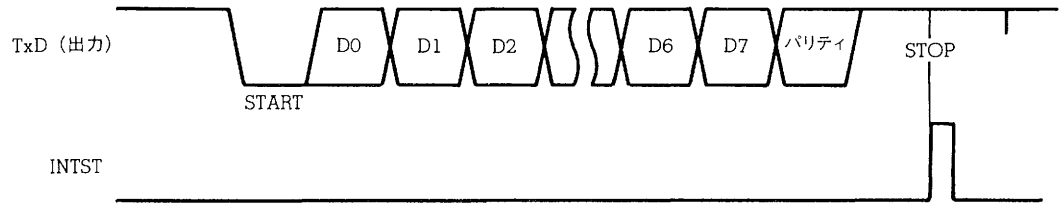
(a) ストップ・ビット長：1



保守／廃止

図7-8　UART送信割り込み発生タイミング（2/2）

（b）ストップ・ビット長：2



7.4.7　DMA要求

送信割り込み要求，受信割り込み要求はそのままDMA要求に接続することができます。

外部メモリ→受信バッファ（RxB），送信バッファ（TxB）→外部メモリのデータ転送をDMAで行うことにより，送信データの設定や受信データの受け取りに関するCPUの負荷を軽減できます。また，データを高速に転送できるため，高速なシリアル通信にも対応できます。

表7-6に，対象となる割り込み要求，転送対象，DMAチャネルを示します。DMAの選択については**6.6.1　DMAモード・レジスタ（DMAM0，DMAM1）**を参照してください。

注意1．送受信割り込み要求をDMA要求として設定する場合，必ず対応する割り込みをマスク（割り込み要求制御レジスタのMKビット＝1）してください。マスクしない場合，誤動作する可能性があります。

2．CSIモード時は，送受信データのDMA転送はできません。

表7-6　DMA要求と接続できるUART割り込み要求

シリアル・インタフェース・チャネル		割り込み要求（DMA要求ソース）	転送対象I/O	TS1	TS0	対象DMAチャネル
ch0	受信	INTSR0	RxB0	1	1	ch1：サブチャネル
	送信	INTST0	TxB0	0	0	ch0：サブチャネル ch1：サブチャネル
				1	1	ch0：メイン・チャネル
ch1	受信	INTSR1	RxB1	0	0	ch1：メイン・チャネル
	送信	INTST1	TxB1	1	1	ch0：サブチャネル

保守／廃止

7.4.8　注意事項

(1) 送信動作中の送信禁止操作

送信動作中に$\overline{CTS}$=1，またはTxE=0により送信禁止操作を行った場合，送信中のデータは最後まで送られ，送信シフト・レジスタのデータを送信完了した時点で送信を停止します。

送信バッファ(TxB)にデータがある場合，そのデータは送出されず次に送信許可になるまで送信バッファに保持されます。

(2) 送信割り込みの設定

送信バッファに送信データを書き込む前に送信を許可すると，送信割り込み要求がただちに発生します。送信を許可する前に，必ず割り込みの設定を行ってください。

(3) 受信エラー割り込み発生に伴う受信割り込み

受信エラー割り込みが発生したときは，続いて受信割り込み要求も発生します。受信エラー割り込みのあとの受信割り込み要求を受け付けたくない場合は，UARTMレジスタのRxEビットをソフトウエアでクリアしてください。エラーが発生しても，受信バッファ(RxB)へ受信したデータは転送されます。RxEビットをソフトウエアでクリアする際は，オーバラン・エラーの発生を防ぐため必ず受信バッファの値を読み出してください。

(4) 受信BRGレジスタへの値設定

UARTでは，シリアル・クロックと受信データの位相ずれを抑えるため，受信BRGレジスタに7以上の値を設定して使用してください。

(5) 送受信BRGレジスタの読み出し

送信，受信中には，送信／受信BRGレジスタを読み出さないでください。シフト・クロックの周期が一時的に狂ってしまいます。

(6) 送受信割り込みについて

V55PIでは，送受信割り込み要求の発生を2回分割り込みコントローラ側で記憶します。ただし，ソフトウエアにより割り込みフラグをクリアし，割り込み要求を取り消した場合，割り込み要求は2回分とも取り消されます。

(7) 途中で送信が止まった場合について

途中で送信が止まるという現象が発生した場合は，次に示す内容をまず確認してください。

① 送信割り込み内で送信バッファにデータがセットされているか

送信割り込みは，送信バッファ（TxB）からシフト・レジスタへ送信データが転送された結果

保守／廃止

TxBが空になったときに発生します。したがって，送信割り込みで送信データを設定する場合は，送信割り込み処理内で必ずTxBにデータを書き込んでください。

② ソフトウエアで割り込みフラグ（IF）をクリアしてないか

送信許可に設定するとすぐに割り込み要求が発生します。したがって，送信許可に設定したあとで割り込み要求制御レジスタ（IC××）を設定する場合，誤って割り込みフラグをクリアしてしまう可能性があります。

5.14.4 割り込みが発生しない場合についても参照してください。

保守／廃止

7.5　CSI動作

V55PIのCSI（クロックト・シリアル・インタフェース）には次の２つの動作モードがあります。

（1）３線式シリアルI/Oモード（IOEモード）

シリアル・クロック（$\overline{SCK}$）とシリアル・データ入出力（SI，SO）の３本のラインによって８ビット長のデータ転送を行います。

V25，V25＋™に対して機能が強化されており，データのMSBファースト／LSBファースト選択が可能となっています。

（2）シリアル・バス・インタフェース・モード（SBIモード）

シリアル・クロック（$\overline{SCK}$）と，シリアル・バス・インタフェース（SB0またはSB1）の２本のラインにより複数のデバイスと通信可能なモードです。

NECシリアル・バス・フォーマットに準拠します。

SBIモードでは，シリアル・データ・バス上にシリアル通信の対象デバイス選択のための“アドレス”，対象デバイスに対して指令を与える“コマンド”，および実際の“データ”を出力することができます。したがって，ハンドシェークのためのラインを必要とせず，入出力ポートの有効活用ができます。

また，ウエーク・アップ解除動作は，マクロ・サービスを用いて実現する方式をとっています。

注意　SBIモードでは，受信動作の許可（CRxEビット＝1），送信動作の許可（CTxEビット＝1）を行う前に必ず次の設定を行ってください。

- **シリアル・クロックの選択**
- **SBIモードでのシリアル・データ入出力端子の選択**

7.5.1　構　　成

V55PIのクロックト・シリアル・インタフェースは，図7-9のように構成されます。

クロックト・シリアル・インタフェースは，SO0/SB0（SO1），SI0/SB1（SI1），シリアル・クロック入出力（$\overline{SCK0}$，$\overline{SCK1}$）の各端子と，クロックト・シリアル・インタフェース・モード・レジスタ（CSIM0，CSIM1），シリアル・インタフェース・コントロール・レジスタ（SBIC0），クロックト・シリアルI/Oシフト・レジスタ（SIO0，SIO1）および受信バッファ（RxB0，RxB1）で構成されます。

SO0/SB0端子は，IOEモードでは，シリアル・データ出力端子（SO），SBIモードでは，シリアル・データ入出力端子SB0として機能します。

また，SI0/SB1端子は，IOEモードでは，シリアル・データ入力端子（SI），SBIモードではシリアル・データ入出力端子SB1として機能します。

端子の一覧を表7-7に示します。

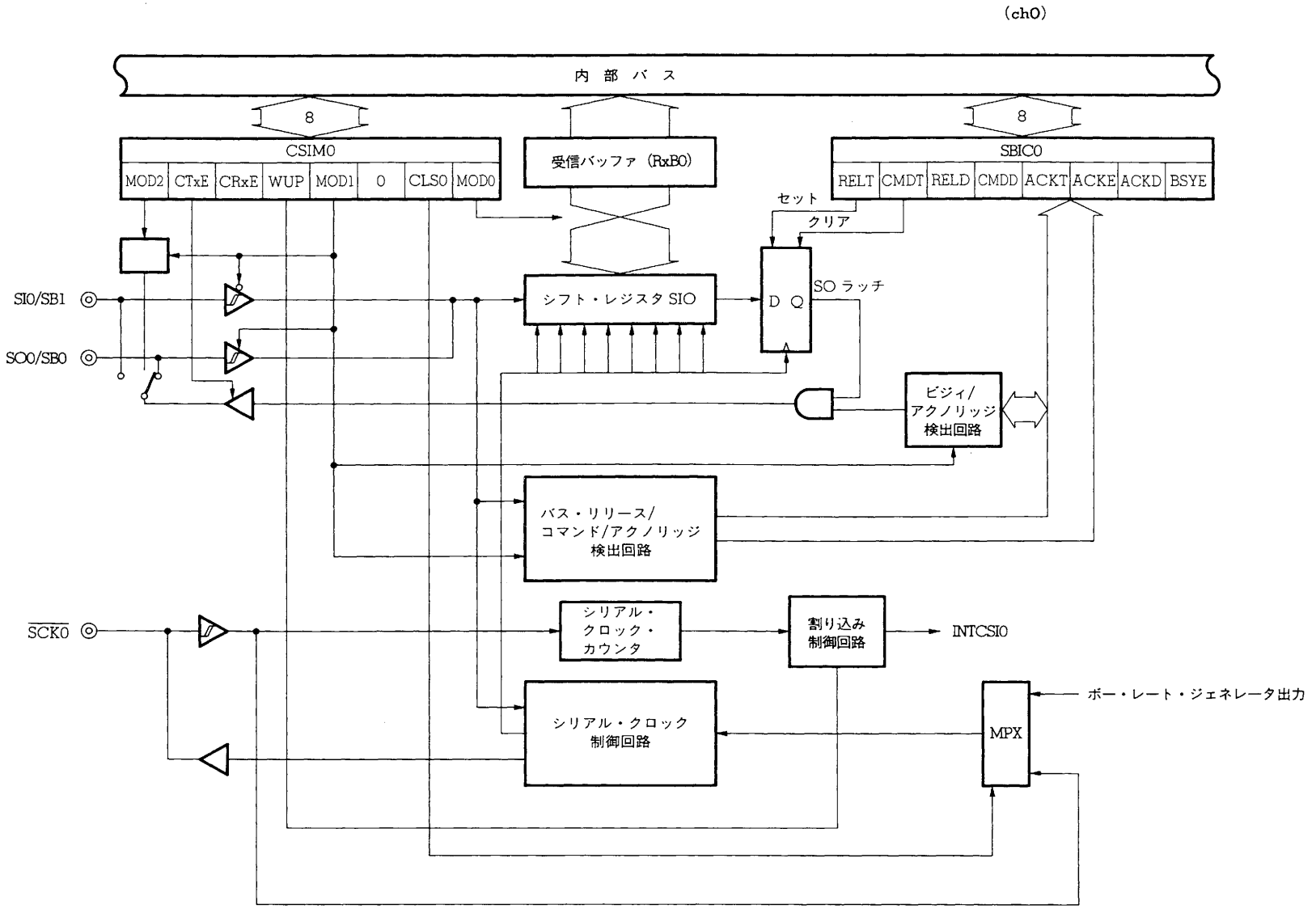
保守／廃止

表7-7　クロックト・シリアル・インタフェース用端子

端子名称	IOEモード	SBIモード
SO0/SB0	シリアル・データ出力（SO0）	シリアル・データ入出力(SB0)
SI0/SB1	シリアル・データ入力（SI0）	シリアル・データ入出力(SB1)
SO1	シリアル・データ出力（SO1）	—
SI1	シリアル・データ入力（SI1）	—
$\overline{SCK}$	シリアル・クロック入出力	

保守／廃止

図7-9　クロックト・シリアル・インタフェースのブロック図



保守／廃止

SO, SI, $\overline{SCK}$の各端子はポート3の各端子との兼用になっており，プロトコルをUARTモードに，ポートをコントロール機能に設定している場合（PMCn＝1），クロックト・シリアル・インタフェース・モード・レジスタ（CSIM）の各制御ビットにより次のように機能します。

表7－8　制御レジスタによる機能指定（SO/SB0端子）…ch0

PMC30	PM30	CSIM0		動作モード	P30リード動作	P30ライト動作
		MOD1	MOD2			
1	0	1	0	・シリアル・データ入出力（SB0） ・N-chオープン・ドレーン注	出力バッファ前のコントロール信号をリード	出力ラッチにデータをライト
	1				端子レベルをリード	
	0	1	1	ハイ・インピーダンス	出力バッファ前のコントロール信号をリード	
	1				端子レベルをリード	
	0	0	1/0	・シリアル・データ出力（SO） ・CMOS出力	出力バッファ前のコントロール信号をリード	
	1				端子レベルをリード	

表7－9　制御レジスタによる機能指定（SI/SB1端子）…ch0

PMC31	PM31	CSIM0		動作モード	P31リード動作	P31ライト動作
		MOD1	MOD2			
1	0	1	0	ハイ・インピーダンス	出力バッファ前のコントロール信号をリード	出力ラッチにデータをライト
	1				端子レベルをリード	
	0	1	1	・シリアル・データ入出力（SB1） ・N-chオープン・ドレーン注	出力バッファ前のコントロール信号をリード	
	1				端子レベルをリード	
	0	0	1/0	・シリアル・データ入力（SI） ・ハイ・インピーダンス	出力バッファ前のコントロール信号をリード	
	1				端子レベルをリード	

注　外部にプルアップ抵抗を接続してください。

保守／廃止

表7－10　制御レジスタによる機能指定（SO端子）…ch1

PMC34	PM34	動作モード	P34リード動作	P34ライト動作
1	0	・シリアル・データ出力（SO） ・CMOS出力	出力バッファ前のコントロール信号をリード	出力ラッチにデータをライト
	1		端子レベルをリード	

表7－11　制御レジスタによる機能指定（SI端子）…ch1

PMC35	PM35	動作モード	P35リード動作	P35ライト動作
1	0	・シリアル・データ入力（SI） ・ハイ・インピーダンス	出力バッファ前のコントロール信号をリード	出力ラッチにデータをライト
	1		端子レベルをリード	

表7－12　制御レジスタによる機能指定（$\overline{\text{SCK}}$端子）

PMC32/PMC36	PM32/PM36	CSIM CLS0	動作モード	P32/P36 リード動作	P32/P36 ライト動作
1	0	0	・外部$\overline{\text{SCK}}$入力 ・ハイ・インピーダンス	出力バッファ前のコントロール信号をリード	出力ラッチにデータをライト
		1	・$\overline{\text{SCK}}$出力 ・CMOS出力		
	1	0	・外部$\overline{\text{SCK}}$入力 ・ハイ・インピーダンス	端子レベルをリード	
		1	・$\overline{\text{SCK}}$出力 ・CMOS出力		

備考　$\overline{\text{SCK}}$端子は，MOD1，MOD2ビットの設定により影響を受けません。

保守／廃止

7.5.2　3線式シリアルI/Oモード（IOEモード）の動作

3線式シリアルI/Oモード（IOEモード）は，従来のクロック同期式シリアル・インタフェースを内蔵するデバイスとの通信を行うためのモードです。

基本的には，シリアル・クロック（$\overline{SCK}$），シリアル・データ出力（SO），シリアル・データ入力（SI）の3本のラインで通信を行います。複数のデバイスと接続する場合は，ハンドシェーク用のラインが必要です。

図7－10　IOEモードのシステム構成例

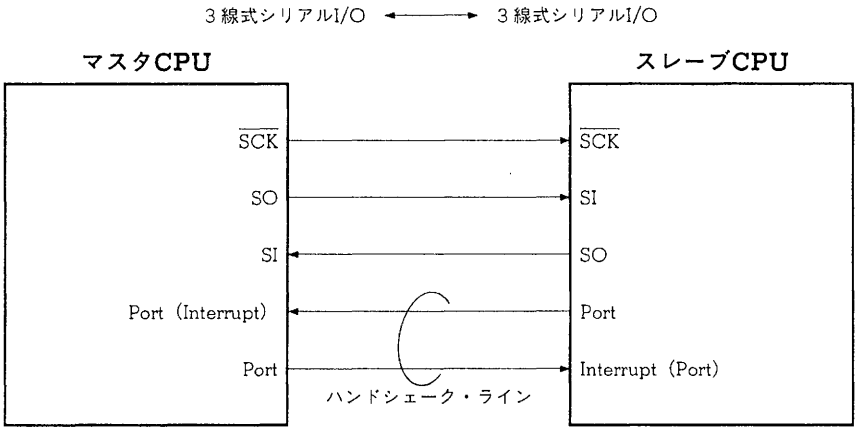
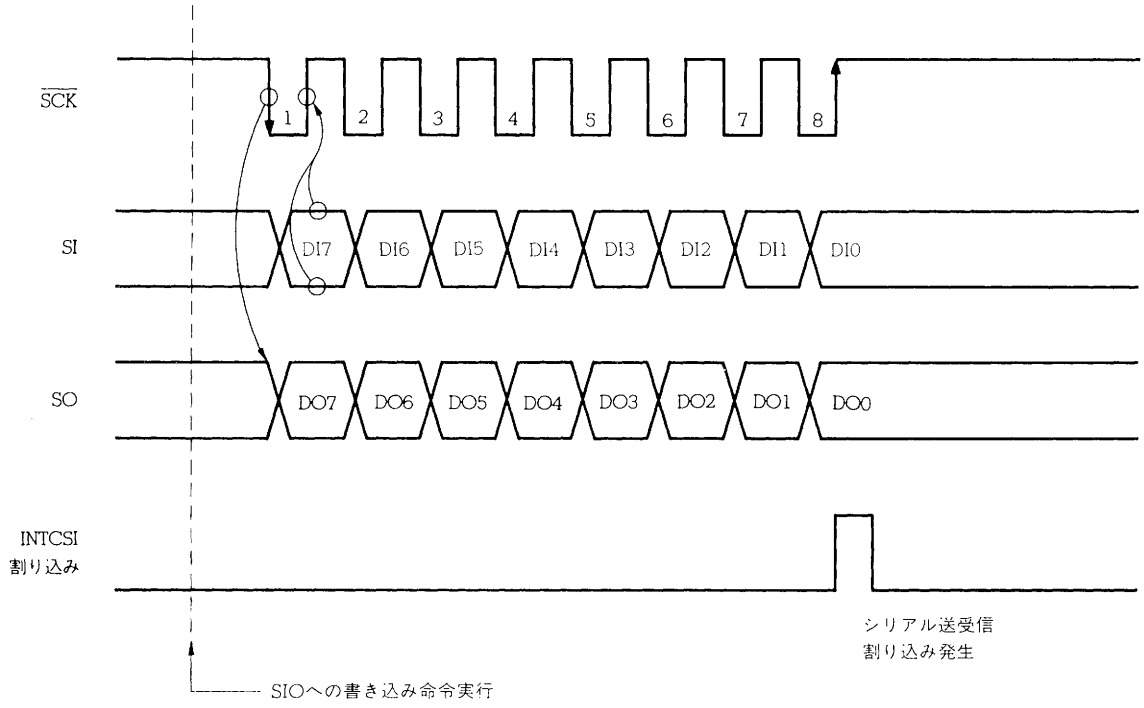


図7－11にIOEモードの転送タイミングを示します。

シリアル転送の送信／受信機能は，クロックト・シリアル・インタフェース・モード・レジスタ（CSIM）の送信許可ビット（CTxE）と受信許可ビット（CRxE），シリアルI/Oシフト・レジスタ（SIO），シリアル・クロック（$\overline{SCK}$）により制御します。

保守／廃止

図7－11　IOEモードの転送タイミング



備考　送信時：$\overline{SCK}$の立ち下がりに同期して転送をスタート

受信時：$\overline{SCK}$の立ち上がりで受信データをラッチ

(1) 送信動作

$\overline{SCK}$信号の立ち下がりに同期して，SIOレジスタのデータをSO端子へMSBを先頭に順次出力します。

(a) 許可状態

設定：クロック・シリアル・インタフェース・モード・レジスタ（CSIM）のCTxEビット＝1，CRxEビット＝0

起動：●内部クロックを選択した場合

① 送信許可状態に設定する。

② SIOレジスタに送信データを書き込む。

③ $\overline{SCK}$端子からシフト・クロックが出力されて送信開始。

●外部クロックを選択した場合

① 送信許可状態に設定する。

② SIOレジスタに送信データを書き込む。

③ $\overline{SCK}$端子にシフト・クロックが入力されて送信開始。

保守／廃止

(b) 禁止状態

設定：CSIMレジスタのCTxEビット＝0

各端子の状態：SO端子はハイ・インピーダンス出力です。

$\overline{SCK}$端子にシフト・クロックを入力してもSO端子からデータは出力されません。

(c) 送信割り込みの発生タイミング

SIOレジスタの内容を8ビット出力するごとにシリアル送受信割り込み（INTCSI）が発生します。

(2) 受信動作

$\overline{SCK}$信号の立ち上がりに同期して，SI端子へ入力されたデータをSIOレジスタへ順次入力します。受信データが8ビットそろったところで，SIOレジスタ内の受信データを受信バッファ（RxB）へ転送します。

(a) 許可状態

設定：CSIMレジスタのCRxEビット＝1，CTxEビット＝0

起動：●内部クロックを選択した場合

① 受信許可状態に設定する。

② 受信バッファ（RxB）を読み出す。

③ $\overline{SCK}$端子からシフト・クロックが出力されて受信開始。

●外部クロックを選択した場合

① 受信許可状態に設定する。

② 受信バッファ（RxB）を読み出す。

③ $\overline{SCK}$端子にシフト・クロックが入力されて受信開始。

(b) 禁止状態

設定：CSIMレジスタのCRxEビット＝0

動作：$\overline{SCK}$端子へシフト・クロックを入力してもデータの受信ができません。ただし，受信禁止時も受信バッファのデータを読み出すことはできます。

(c) 受信割り込みの発生タイミング

受信データが8ビット入力されるごとにシリアル送受信割り込み（INTCSI）が発生します。

保守／廃止

(3) 送受信動作

(a) 許可状態

設定：CSIMレジスタのCTxEビット＝CRxEビット＝1

起動：●内部クロックを選択した場合

① 送受信許可状態に設定する。

② SIOレジスタに送信データを書き込む。

③ $\overline{SCK}$端子からシフト・クロックが出力されて送受信開始。

●外部クロックを選択した場合

① 送受信許可状態に設定する。

② SIOレジスタに送信データを書き込む。

③ $\overline{SCK}$端子にシフト・クロックが入力されて送受信開始。

(b) 禁止状態

設定：CSIMレジスタのCTxEビット＝CRxEビット＝0

各端子の状態：SO端子はハイ・インピーダンス出力です。

$\overline{SCK}$端子にシフト・クロックを入力してもSO端子からデータは出力されません。

また，データの受信はされません。ただし受信バッファ(RxB)のデータを読み出すことはできます。

(c) 送受信割り込みの発生タイミング

8ビット・データの転送ごとにシリアル送受信割り込み（INTCSI）が発生します。

7.5.3 シリアル・バス・インタフェース・モード（SBIモード）の動作

SBIモードは，シリアル・クロック（$\overline{SCK0}$）とシリアル・バス（SB0またはSB1）の2本のラインで複数のデバイスと通信を行うモードです。SBIモードでのシステム構成図を図7-12に示します。

シリアル・バスの選択は，CSIMレジスタで行います。ここではSB0（SO/SB0）を用いて説明を行いますが，SB1（SI/SB1）を選択した場合も動作はまったく同じです。

シリアル・バス上にアドレスを出力することによって複数のデバイスの中から通信対象とするスレーブ・デバイスを選択します（**図7-13**参照）。

通信対象が決定したのち，マスタ・デバイスとスレーブ・デバイスとの間でコマンド，データの送受信を行い，シリアル通信を実現します（**図7-14，図7-15**参照）。

(1) シリアル転送の送信機能

SBIモードにおいてシリアル送信は，クロクト・シリアル・インタフェース・モード・レジスタ（CSIM）の送信許可ビット（CTxE）をセット(1)した状態でクロクト・シリアルI/Oシフト・レ

保守／廃止

ジスタ（SIO）へ送信データを書き込むことによって行います。送信後のシフト・レジスタには“0”が書き込まれます。

シリアル・クロック（$\overline{SCK0}$）の立ち下がりに同期して，シフト・レジスタのシフトが行われ，送信データがSOラッチに保持されます。SOラッチに保持されたデータが，SB0端子からMSBを先頭にして出力されます。

(2) シリアル転送の受信機能

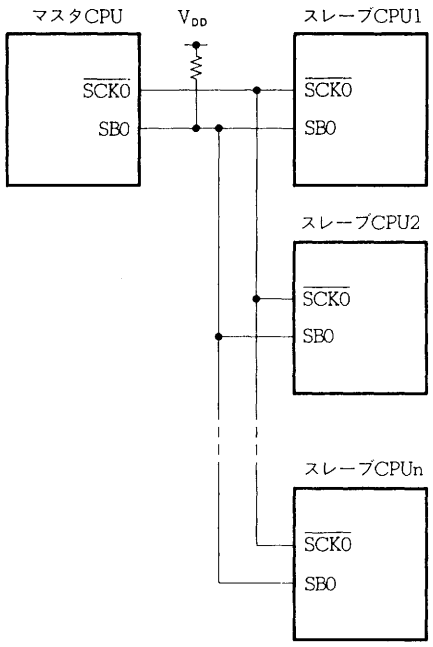
シリアル受信は，CSIMレジスタの受信許可ビット（CRxE）をセット(1)するか，あるいは，このビットがセットされている状態で受信バッファ（RxB）から受信データを読み出すことによって行います。

シリアル・クロック（$\overline{SCK0}$）の立ち上がりでSB0端子に入力されたデータがMSBを先頭にSIOレジスタにラッチされます。SB0端子は，シリアル・データ・バスに指定された場合，外部で抵抗によるプルアップが必要となります。

(3) シリアル転送の送受信動作

シリアル送受信は，CRxEビット＝CTxE＝1の状態でSIOレジスタに送信データを書き込むことで行います。SIOレジスタにはシリアル・バス上のデータがそのまま入力されます。SIOレジスタに書き込んだ送信データとシリアル・バスから入力されたデータを比較することで，バス衝突の有無などを確認できます。

図7－12　SBIモードのシステム構成例



保守／廃止

図7－13　マスタ・デバイスからスレーブ・デバイスへのアドレス転送動作

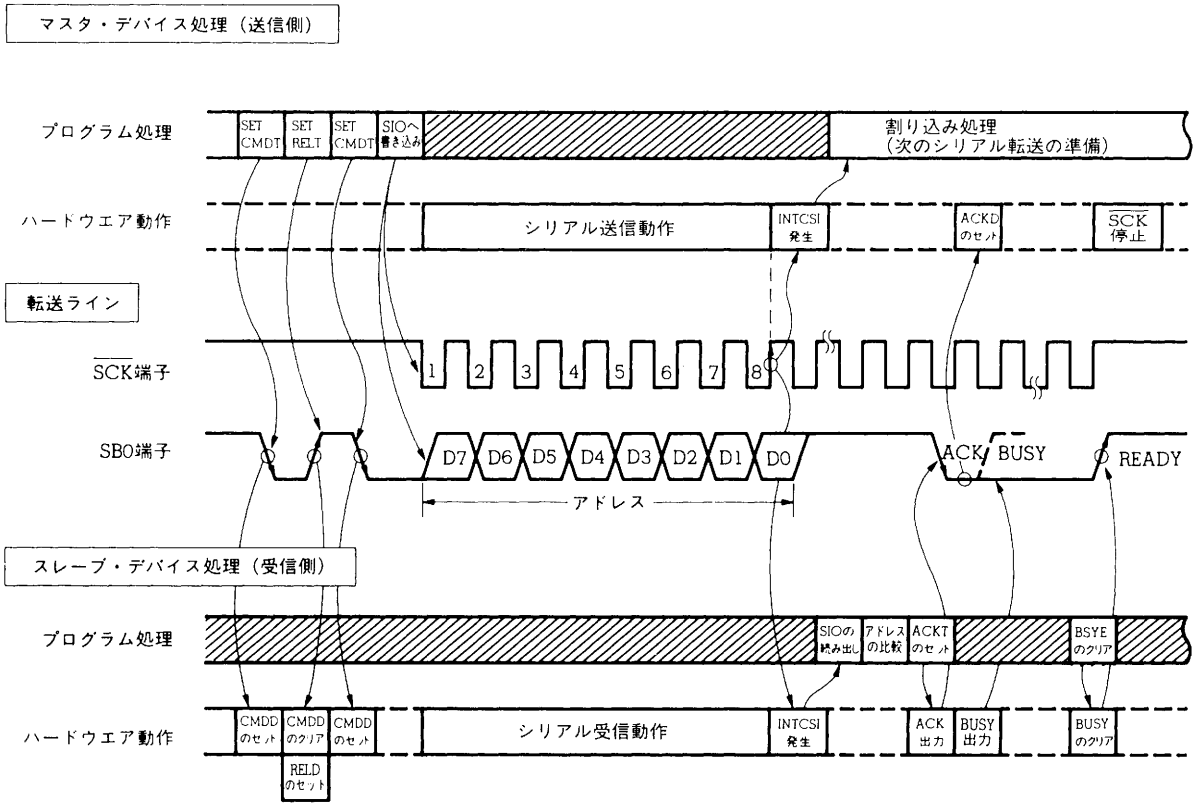
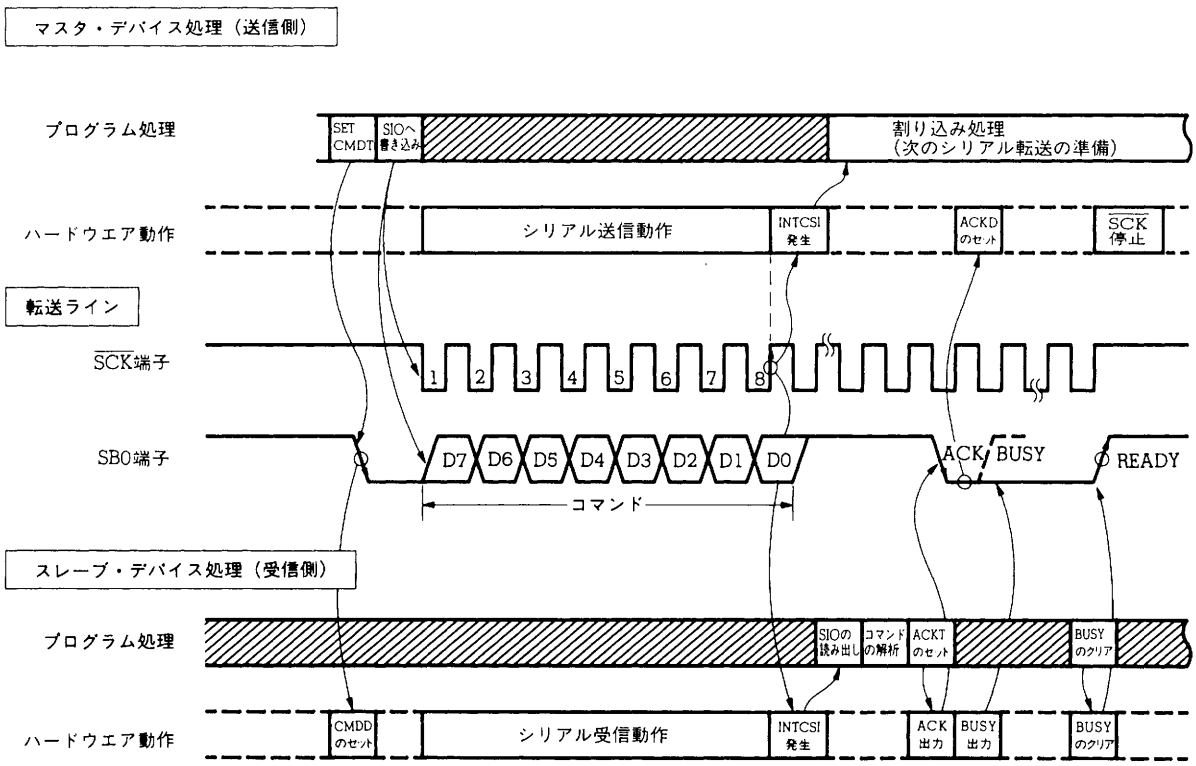


図7－14　マスタ・デバイスからスレーブ・デバイスへのコマンド転送動作



保守／廃止

図7－15　マスタ・デバイスからスレーブ・デバイスへのデータ転送動作

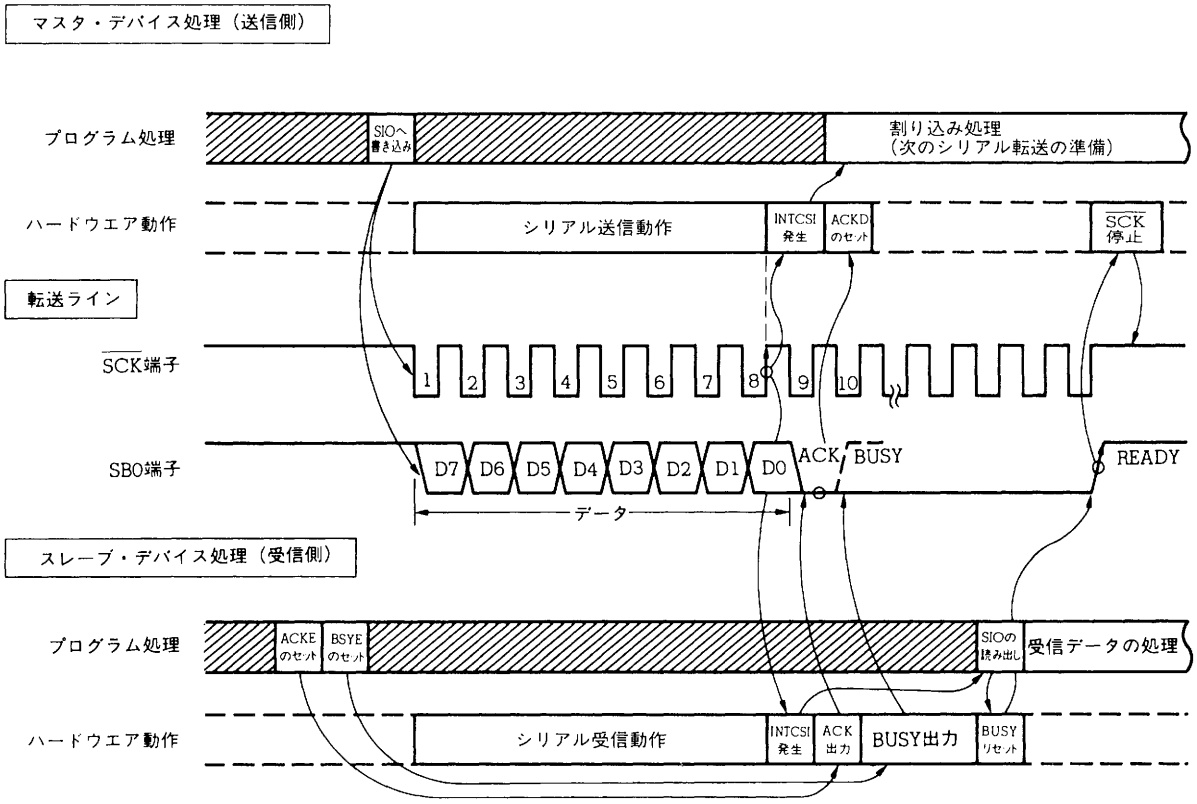
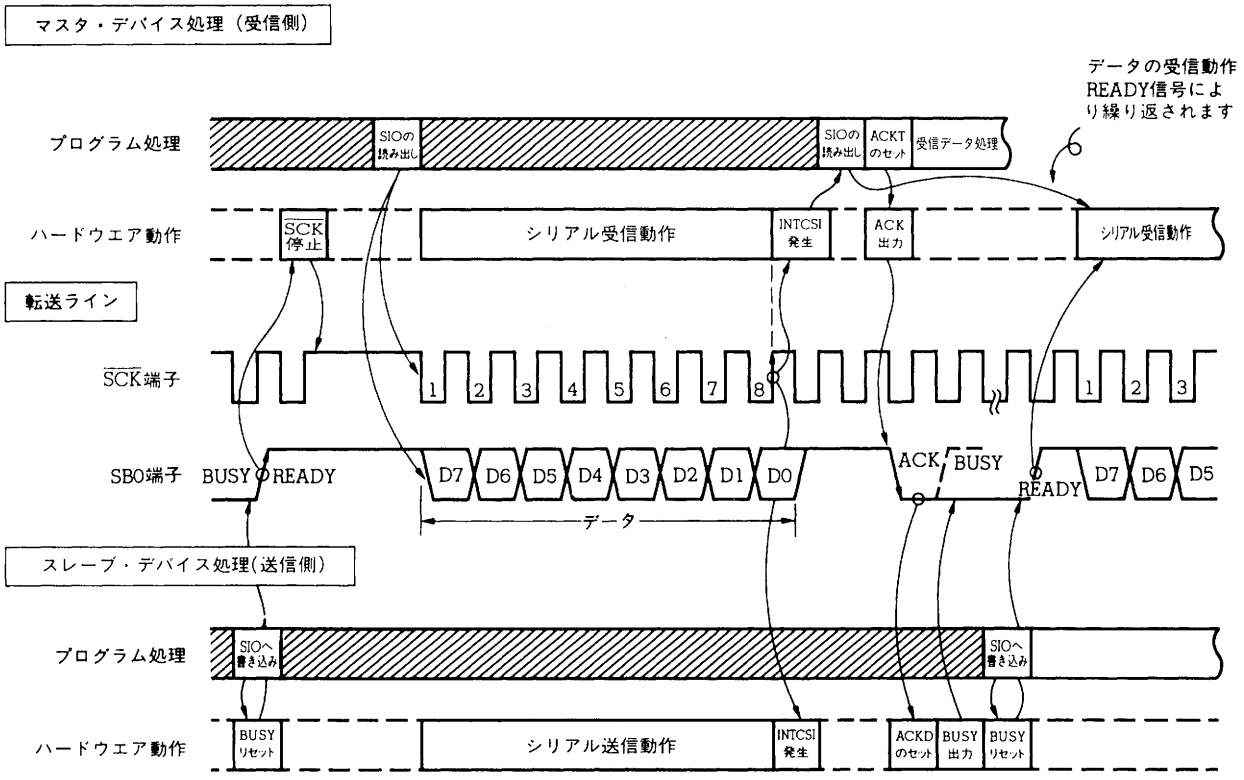


図7－16　スレーブ・デバイスからマスタ・デバイスへのデータ転送動作



保守／廃止

(4) ウエーク・アップ機能

SBIモードでは，アドレスを受信したときにだけ，割り込み要求信号(INTCSI)を発生するウエーク・アップ機能があります。

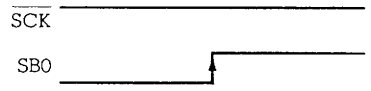
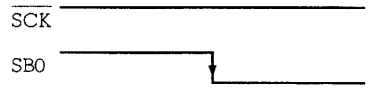
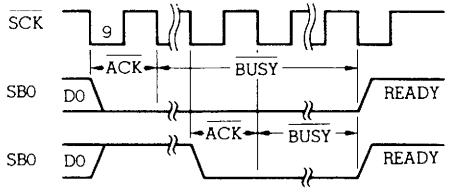
ウエーク・アップ機能の設定は，CSIMレジスタのビット5（WUP）をセット(1)することによって行います。

シリアル・インタフェースが図7-12のようなシステム構成で，ウエーク・アップ機能が設定されている場合，マスタ・デバイスからのアドレス受信時のみ全スレーブ・デバイスのシリアル送受信割り込み（INTCSI）が発生します。スレーブ・デバイスのアドレス一致検出は，ソフトウエアで行い，一致アドレス受信時はウエーク・アップ状態を解除（WUP＝0）し，以後の“コマンド”，“データ”受信に備えます。その後の“コマンド”，“データ”転送時は，ウエーク・アップ状態を解除したスレーブ・デバイス以外はシリアル送受信割り込みを発生しませんので，ある1つのスレーブCPUと送受信を行うとき，その他のスレーブCPUはシリアル通信に関係なく動作させることができます。ウエーク・アップ機能がない場合は，データを受信するごとに割り込みを発生しますので，すべてのスレーブCPUがシリアル通信ごとに影響を受けます。ウエーク・アップ機能を用いることによって効率のよい処理を行うことができます。

SBIモードにおける，マスタ・デバイスおよびスレーブ・デバイスから出力される各種の信号について，その動作，タイミングなどを表7-13に示します

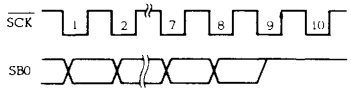
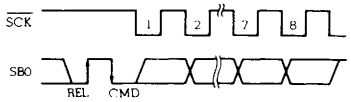
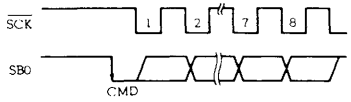
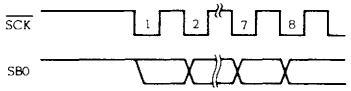
保守／廃止

表 7－13　SBIモードにおける各種の信号（1/2）

信号名称	出力するデバイス	定　　義	タイミング・チャート	出力される条件	フラグへの影響	信号の意味
バス・リリース信号（REL）	マスタ	$\overline{SCK}$＝1のときのSB0の立ち上がりエッジ		●RELTのセット	●RELDをセット ●CMDDをクリア	今まで選択していたスレーブに対する一連の転送処理の終了を示す。
コマンド信号（CMD）	マスタ	$\overline{SCK}$＝1のとき，SB0の立ち下がりエッジ		●CMDTのセット	●CMDDをセット	i）REL信号出力後，送信データはアドレス。 ii）REL信号出力なし。送信データはコマンド。
アクノリッジ信号（$\overline{ACK}$）	スレーブ	シリアル受信完了後，$\overline{SCK}$の１クロックの期間SB0に出力されるロウ・レベルの信号	〔同期ビジィ出力〕	●ACKE＝1 ●ACKTのセット	●ACKDをセット	受信完了
ビジィ信号（$\overline{BUSY}$）	スレーブ	〔同期ビジィ信号〕 アクノリッジ信号に続いてSB0に出力されるロウ・レベルの信号		●BSYE＝1	—	処理中のため，シリアル受信不可状態。
レディ信号（READY）	スレーブ	シリアル転送開始前，完了後SB0に出力されるハイ・レベルの信号		●BSYE＝0 ●SIOへのデータ書き込み命令実行（転送開始指示）	—	シリアル受信可能状態。

保守／廃止

表7－13 SBIモードにおける各種の信号（2/2）

信号名称	出力するデバイス	定　　義	タイミング・チャート	出力される条件	フラグへの影響	信号の意味
シリアル・クロック（$\overline{SCK}$）	マスタ	アドレス／コマンド／データ，ACK信号，同期BUSY信号などの出力のための同期クロック。最初の8個でアドレス／コマンド／データを転送する。		CTxE＝1のとき，SIOへのデータ書き込み命令実行（シリアル転送のスタート指示）注3	CSIIF注1をセット（9クロック目の立ち上がり）注2	シリアル・データ・バスへの信号出力のタイミング
アドレス（A0-A7）	マスタ	REL信号，CMD信号出力後に，$\overline{SCK}$に同期して転送される8ビット・データ。			注2	シリアル・バス上のスレーブ・デバイスのアドレス値
コマンド（D0-D7）	マスタ	REL信号は出力されず，CMD信号のみ出力されたのち，$\overline{SCK}$に同期して転送される8ビット・データ。			なし	スレーブ・デバイスへの指示メッセージ
データ（D0-D7）	マスタ／スレーブ	REL信号，CMD信号ともに出力されず，$\overline{SCK}$に同期して転送される8ビット・データ。			なし	スレーブ，またはマスタで処理する実際のデータ

注1. シリアル送受信割り込み（INTCSI）に対する割り込み要求フラグです。

2. WUP＝0のとき，常に9クロック目の$\overline{SCK}$の立ち上がりでCSIIFをセットする。

   WUP＝1のとき，アドレスを受信したときのみ，CSIIFをセットする。

3. $\overline{BUSY}$状態のときは，READY状態になったのち，転送スタートする。

保守／廃止

7.5.4　制御レジスタ

(1) クロックト・シリアル・インタフェース・モード・レジスタ（CSIM0，CSIM1）

クロックト・シリアル・インタフェースの基本動作モードを指定する８ビットのレジスタです。

CSIM0レジスタはch0に，CSIM1レジスタはch1に対応します。

８ビット操作のメモリ・アクセスによってリード／ライト可能です。

リセット後のCSIMレジスタの値は，00Hになります。

保守／廃止

	7	6	5	4	3	2	1	0
CSIM1	CTxE	CRxE	0	0	0	MOD0	0	CLS0

	7	6	5	4	3	2	1	0
CSIM0	CTxE	CRxE	WUP	MOD2	MOD1	MOD0	0	CLS0

動作クロックの指定（ch0，ch1）

CLS0	クロックの指定	$\overline{SCK}$端子
0	外部クロック・スレーブ	入力
1	内部クロック・マスタ注1	出力

各種動作モードの指定（ch0）

MOD1	MOD0	動作モード	転送方向
0	0	IOEモード	MSBファースト
0	1	IOEモード	LSBファースト
1	0	SBIモード	MSBファースト
1	1	設定禁止	

各種動作モードの指定（ch1）

MOD0	動作モード	転送方向
0	IOEモード	MSBファースト
1	IOEモード	LSBファースト

使用端子の指定（ch0）

MOD2	使用端子と動作
0	SO0/SB0端子で送受信動作
1	SI0/SB1端子で送受信動作

ウエーク・アップ動作の指定注2（ch0）

WUP	動　作
0	転送ごとに割り込み発生
1	アドレス受信時割り込み発生

受信許可／禁止の指定（ch0，ch1）

CRxE	動　作
0	受信動作を禁止
1	受信動作を許可

送信許可／禁止の指定（ch0，ch1）

CTxE	動　作
0	送信動作を禁止
1	送信動作を許可

注1. BRG出力

2. SBIモードで有効

備考 ▨ はリセット後の状態を示します。

保守／廃止

MOD1 **MOD0** 動作モード選択ビット

シリアル転送およびプロトコルの各種モードの指定を行います。

[ch0の場合]

MOD1	MOD0	動作モード	転送方向	使用端子数	使用端子
0	0	IOEモード	MSBファースト	3	SO0/SB0 SI0/SB1 $\overline{SCK0}$
0	1	IOEモード	LSBファースト		
1	0	SBIモード	MSBファースト	2	SO0/SB0（SI0/SB1） $\overline{SCK0}$
1	1	設　定　禁　止			

[ch1の場合]

MOD0	動作モード	転送方向	使用端子数	使用端子
0	IOEモード	MSBファースト	3	SO1 SI1 $\overline{SCK1}$
1	IOEモード	LSBファースト		

MOD2 端子指定ビット（ch0のみ有効）

シリアル・バス・インタフェース・モード（SBIモード）で使用するシリアル・データの入出力端子を指定します。

シリアル・データの入出力端子としてSO0/SB0端子を選択した場合でも，SI0/SB1端子を選択した場合でも同様のSBI機能が実現できます。

IOEモード選択時はMOD2ビットの設定値は意味を持ちません。

MOD0，MOD1で3線式IOEモードを選択した場合，MOD2による端子指定は無効となります。

WUP ウエーク・アップ動作指定ビット

ウエーク・アップ条件の動作を制御します。SBIモード選択時に有効なビットです。IOEモードの場合は必ず“0”に設定してください。

マスタ・デバイスとスレーブ・デバイスが通信しているときにセット(1)し，無用な割り込みが発生しないようにすることができます。

注意　IOEモードの場合，WUPビット＝1にしていると一切の転送動作を行わず，割り込み要求（INTCSI）は発生しません。

保守／廃止

(2) SBIコントロール・レジスタ（SBIC0）

SBIモードでのみ使用します。

シリアル・バスの動作を制御するビットと，ステータスを示すフラグから構成される8ビットのレジスタです。8/1ビット操作のメモリ・アクセスによってセット／クリア／テストなどの操作が可能です。リセット後の値は，00Hになります。

	7	6	5	4	3	2	1	0
SBIC0	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT

バス・リリース・トリガ・ビット

RELT	処　理
0	バス・ライン変化なし
1	バス・ライン（SB1/SB0）0→1

コマンド・トリガ・ビット

CMDT	処　理
0	バス・ライン変化なし
1	バス・ライン（SB1/SB0）1→0

ウエーク・アップ条件検出フラグ

コマンド条件検出フラグ

アクノリッジ・トリガ・ビット

ACKT	処　理
0	$\overline{ACK}$信号を出力しない
1	$\overline{ACK}$信号を出力

アクノリッジ許可ビット

ACKE	処　理
0	$\overline{ACK}$信号の出力を禁止
1	$\overline{ACK}$信号の出力を許可

アクノリッジ検出フラグ

ACKD	動　作
0	$\overline{ACK}$信号を未検出
1	$\overline{ACK}$信号を検出

ビジィ許可ビット

BSYE	処　理
0	$\overline{BUSY}$信号の出力を禁止
1	$\overline{BUSY}$信号の出力を許可

注意　**ch0をIOEモードで使用する際には，SBIC0レジスタには00Hを書き込んでください。**

備考　▨ はリセット後の状態を示します。

保守／廃止

RELT　バス・リリース・トリガ・ビット（ライト・オンリー）

シリアル出力ラッチのセット制御ビットです。

このビットは，一連の転送作業を終了し，バスをアイドル状態にするためのトリガ・ビットです。

ソフトウエアでこのビットをセット(1)すると，シリアル出力ラッチがセット(1)されます。

$\overline{SCK0}$端子がハイ・レベルのときに（シリアル・クロックが出力されていないとき），このビットをセット(1)することにより，バス・ライン（SB0またはSB1）を0 → 1と変化させ，シリアル・バスの回線を切ることができます。

注意　シリアル出力ラッチの操作は，$\overline{SCK0}$端子がハイ・レベルの期間（シリアル・クロックが出力されていないとき）に行ってください。シリアル転送処理中に操作を行った場合は正しく書き込まれません。

CMDT　コマンド・トリガ・ビット（ライト・オンリー）

シリアル出力ラッチのクリア制御ビットです。

ソフトウエアでこのビットをセット(1)すると，シリアル出力ラッチがリセット(0)されます。

$\overline{SCK0}$端子がハイ・レベルのときに（シリアル・クロックが出力されていないとき），このビットをセット(1)することにより，バス・ライン（SB0またはSB1）を1 → 0と変化させ，アドレス／コマンドの送信を知らせることができます。

注意　シリアル出力ラッチの操作は，$\overline{SCK0}$端子がハイ・レベルの期間（シリアル・クロックが出力されていないとき）に行ってください。シリアル転送処理中に操作を行った場合は正しく書き込まれません。

シリアル出力ラッチの書き換え条件を次に示します。

セット条件1.　RELTビットへの“1”の書き込み

2.　SBIモードでの転送終了

リセット条件1.　CMDTビットへの“1”の書き込み

2.　システム・リセット

RELD　ウエーク・アップ条件検出フラグ（リード・オンリー）

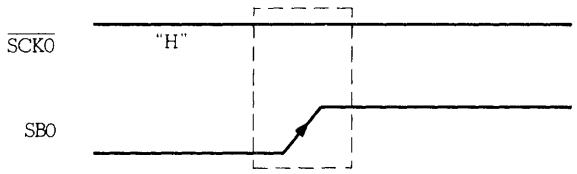
$\overline{SCK0}$端子がハイ・レベル期間にバス・ライン（SB1またはSB0）が0 → 1に変化したことを示すフラグです。

このフラグがセット(1)されて，かつCMDDフラグがセット(1)されている場合は受信データがアドレスであることが分かります。

保守／廃止

図7－17　$\overline{SCK0}$端子とバス・ライン

セット条件　$\overline{SCK0}$がハイ・レベルの間に（シリアル・クロックが出力されていないとき）0→1に変化した。



クリア条件1.　シリアル・スタート命令実行時

CTxE＝1でシフト・レジスタ・ライト

CTxE＝0＊CRxE＝1でシフト・レジスタ・リード

CRxE＝0→1でCSIMレジスタ書き換え

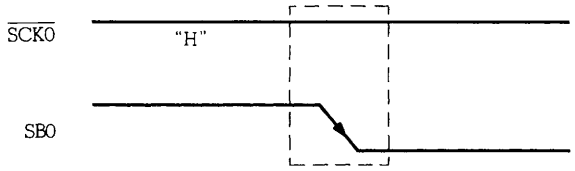
2.　システム・リセット

CMDD　コマンド条件検出フラグ（リード・オンリー）

$\overline{SCK0}$端子がハイ・レベル期間にバス・ライン（SB1またはSB0）が1→0に変化したこと示すフラグです。

図7－18　$\overline{SCK0}$端子とバス・ライン（コマンド条件検出）

セット条件　$\overline{SCK0}$がハイ・レベルの間にバス・ラインが1→0に変化した。



クリア条件1.　RELDフラグがセットされた場合

2.　シリアル・スタート命令実行時

3.　システム・リセット

ACKT　アクノリッジ・トリガ・ビット（ライト・オンリー）

アクノリッジ（$\overline{ACK}$）信号を出力するトリガ・ビットです。

データ転送後にこのビットに1を書き込むと，次の$\overline{SCK0}$の立ち下がりに同期してアクノリッジ（$\overline{ACK}$）信号を出力します。

このビットは，転送終了後に受信データや送信データが正しいことを確認してから，または，受

保守／廃止

信アドレスと自分のアドレスが一致したことを確認してからアクノリッジ（$\overline{ACK}$）信号を出力するといった応用操作で使用します。

注意　このビットを一度セットしたあとで，$\overline{ACK}$出力を取り消すことはできません。

ACKE　アクノリッジ許可ビット

アクノリッジ（$\overline{ACK}$）信号の出力許可ビットです。

このビットを1にしておくと，以降9クロック目にアクノリッジ（$\overline{ACK}$）信号を出力します。

データ転送終了後にこのビットを0→1にすると，次の$\overline{SCK0}$の立ち下がりに同期して$\overline{ACK}$信号を出力します。常時$\overline{ACK}$信号を出力したい場合に，1をセットしておき，毎回$\overline{ACK}$信号を出力するような応用操作で使用します。この場合には，送受信をスタートする前にビットをセットしておきます。

このビットを0にしておくと，ACKは出力しません。

ACKD　アクノリッジ検出フラグ（リード・オンリー）

アクノリッジ（$\overline{ACK}$）信号を検出したかどうかを示すフラグです。

セット条件　　アクノリッジを検出したとき

クリア条件1.　シリアル・スタート時（実際のスタート時）

2.　システム・リセット

BSYE　ビジィ許可ビット

$\overline{BUSY}$信号の出力制御を行うビットです。

このビットをセット(1)することにより，$\overline{ACK}$信号を検出したときに自動的に$\overline{BUSY}$信号を出力します。データ転送開始前にこのビットをセットしておけば，転送が終了し，デバイスのどれかが$\overline{ACK}$信号を出力したあとに$\overline{BUSY}$信号としてSIOラインをロウ・レベルにすることができます。

$\overline{BUSY}$は，シリアル・スタート指示があったときに自動的に解除するほか，BSYEビットをプログラムでクリア(0)したときにも解除します。

図7－19から図7－23に，SBIC0レジスタの各ビット動作と転送のタイミング関係を示します。

保守／廃止

図7－19　RELT, CMDT, RELD, CMDDの動作

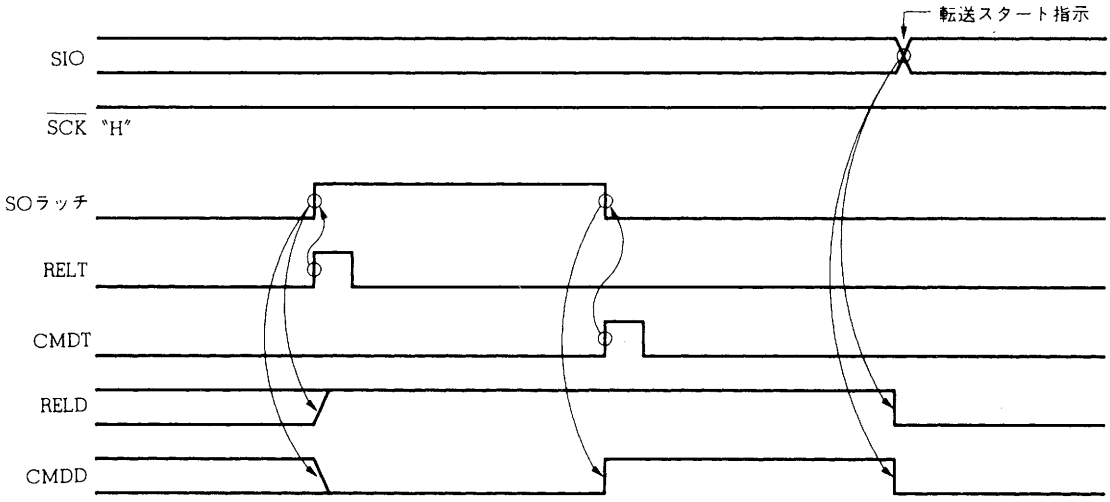
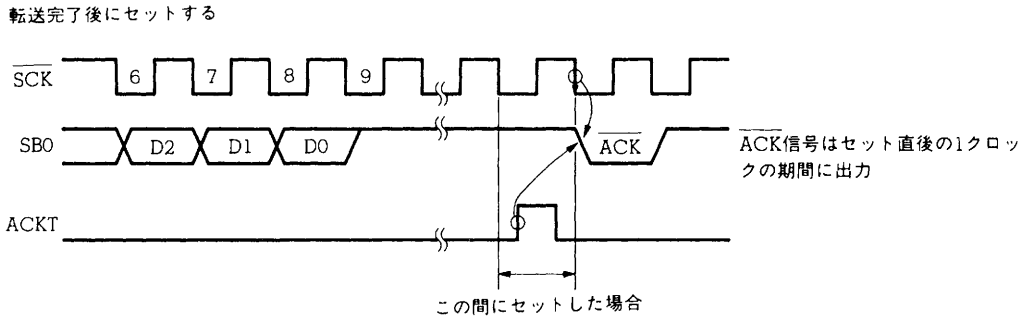


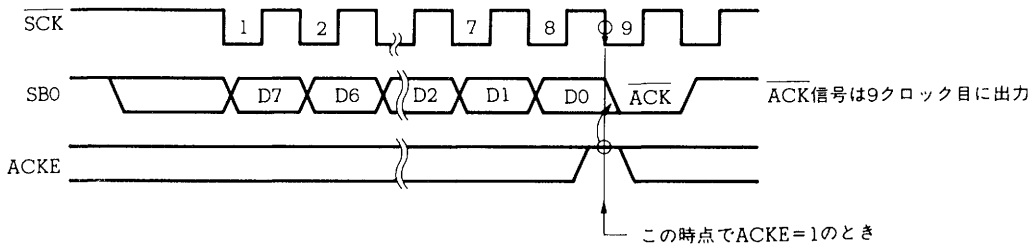
図7－20　ACKTの動作



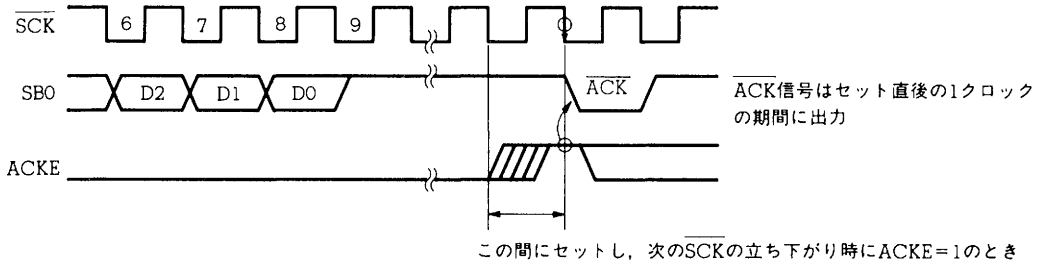
保守／廃止

図 7－21　ACKEの動作

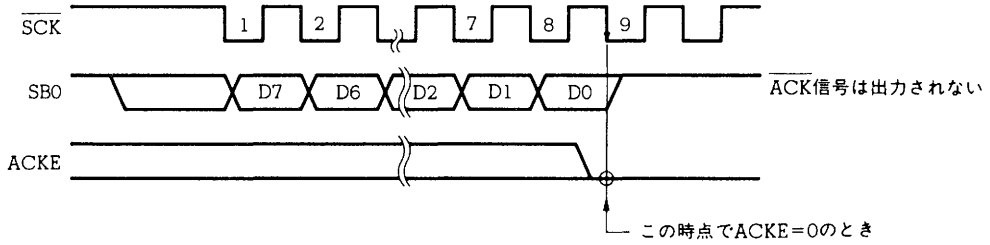
（a）転送時にACKE＝1の場合



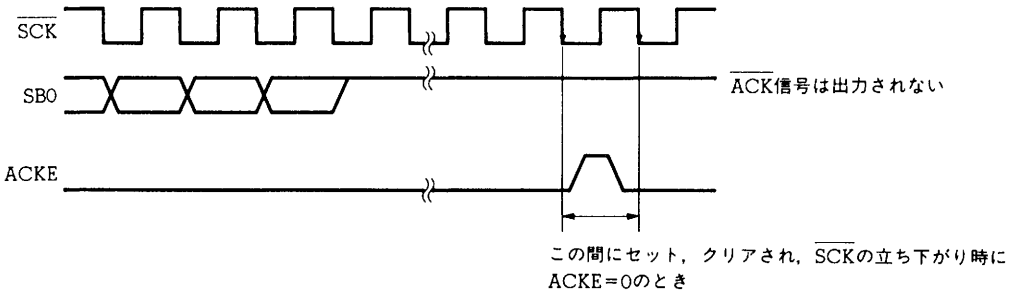
（b）転送完了後にセットした場合



（c）転送完了時にACKE＝0の場合



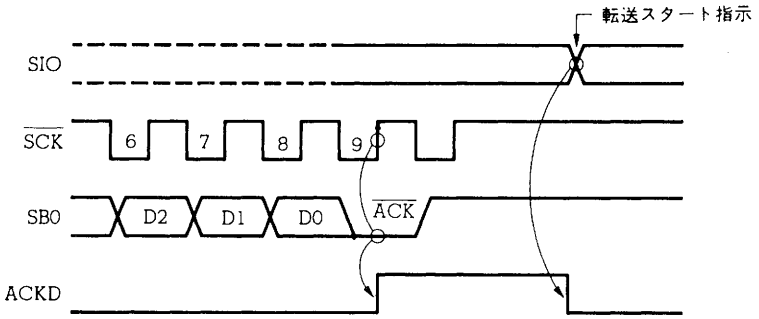
（d）ACKE＝1の期間が短い場合



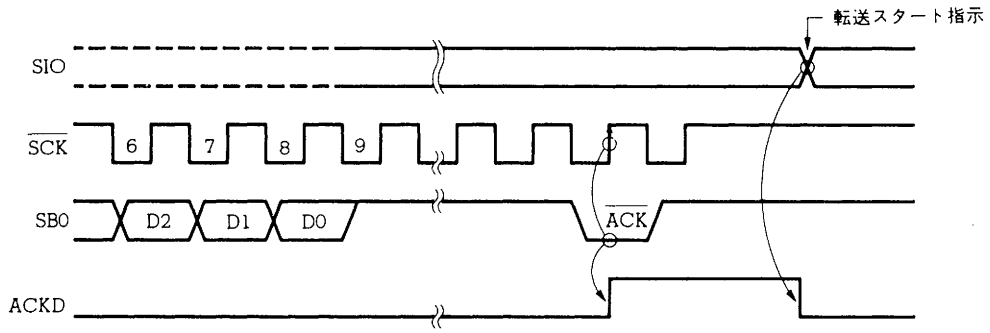
保守／廃止

図7－22　ACKDの動作

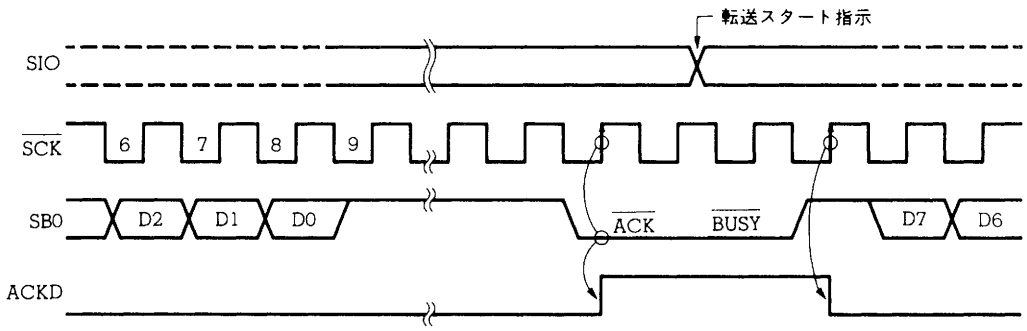
(a) $\overline{SCK}$の9クロック目の期間に$\overline{ACK}$信号が出力された場合



(b) $\overline{SCK}$の9クロック目以降に$\overline{ACK}$信号が出力された場合

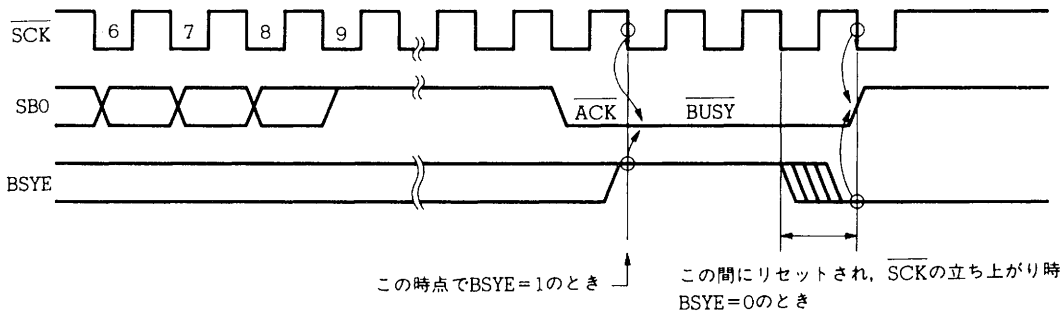


(c) BUSY中に転送スタート指示した場合のクリアのタイミング



保守／廃止

図 7－23　BSYEの動作



(3) クロックト・シリアルI/Oシフト・レジスタ（SIO0，SIO1）

パラレルーシリアルの変換を行う８ビットのシフト・レジスタです。CPUから送信データを８ビット操作のメモリ・アクセスで書き込みのみが可能です。

シフト動作は，CSIMレジスタのMD0，MD1ビットにより，LSB先頭またはMSB先頭のどちらかを選択します。

(4) 受信バッファ（RxB0，RxB1）

受信データを保持する８ビットのレジスタです。CPUから８ビット操作のメモリ・アクセスで読み出しのみが可能です。

保守／廃止

7.5.5　割り込み要求

クロックト・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，シリアル送受信割り込み要求（INTCSI0，INTCSI1）があります。

（1）シリアル送受信割り込み要求（INTCSI0，INTCSI1）

IOEモードの場合は，シリアル転送ごとにシリアル送受信割り込み要求が発生します。

SBIモードの場合は，CSIMレジスタのWUPビットによるウエーク・アップ動作の指定によって，割り込み要求の発生条件が異なります。

WUPビットがクリア(0)されている場合は，シリアル転送後ごとに，シリアル送受信完了割り込み要求が発生しますが，WUPビットがセット(1)されている場合は，アドレスを受信した場合にのみ割り込み要求が発生します。

注意　INTCSI割り込みは，UARTのINTSR割り込みと，割り込み要求制御レジスタおよびベクタなどの割り込み応答機能を兼用しています。プロトコルの選択により自動的に対応の割り込み制御レジスタに設定されます。

7.5.6　注意事項

（1）SBIモードでの設定順序について

SBIモードでは，受信動作の許可（CSIMレジスタ　CRxEビット＝1），送信動作の許可（CSIMレジスタ　CTxEビット＝1）の設定を行う前に必ず次の設定を行ってください。

- シリアル・クロックの選択
- SBIモードでのシリアル・データ入出力端子の選択

（2）IOEモードでの受信データのキャンセル

IOEモードでは，受信動作を途中で中止した場合，CSIMレジスタのCTxEビットとCRxEビットをともに0にすることにより，シフト・カウンタがリセットされシフト・レジスタ（SIO）は初期状態に戻ります。

保守／廃止

第8章　パラレル・インタフェース機能

V55PIは，セントロニクス仕様のインタフェースにおけるデータ入力および汎用のデータ入出力のためのパラレル・インタフェース・ユニットを内蔵しています。

8.1　特　　徴

○セントロニクス仕様のインタフェースに対応
○ソフトウエアによる入出力モード切り替え可能
○ソフトウエアによるBUSY信号操作可能
○BUSY信号および$\overline{\text{ACK}}$信号の出力タイミング設定可能
○外部割り込みによるイニシャライズ機能
○パラレル・インタフェース専用割り込みソース
　●パラレル・インタフェース割り込み（INTPAI）
○パラレル送受信時のDMA要求信号発生機能
　●INTPAIをDMA起動トリガとします。
○信号端子の入力特性はTTLレベル（セントロニクス仕様のインタフェース）

8.2　構　　成

V55PIには，制御レジスタとして次のようなレジスタがあります(各レジスタの詳細は，**8.4　制御レジスタ**を参照してください。

(1) パラレル・インタフェース・コントロール・レジスタ 0（PAC0）

入力モードと出力モードの切り替えを制御します。

(2) パラレル・インタフェース・コントロール・レジスタ 1（PAC1）

入力モード時のBUSY端子を制御します。

(3) パラレル・インタフェース・ステータス・レジスタ（PAS）

(4) パラレル・インタフェース・アクノリッジ・インターバル・レジスタ（PAI）

(5) パラレル・インタフェース・バッファ（PAB）

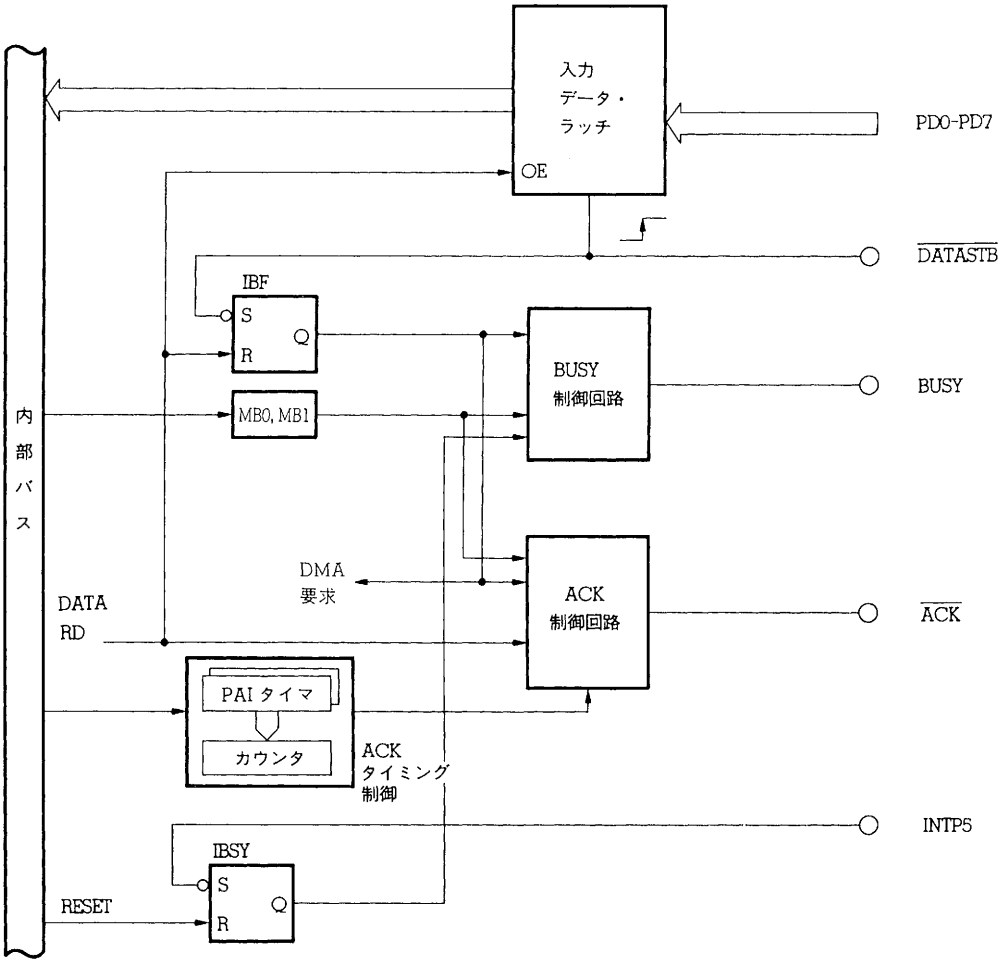
入出力データをラッチしておきます。

保守／廃止

図８－１に各モードでのパラレル・インタフェースの構成を示します。

図８－１　パラレル・インタフェースのブロック図（1/2）

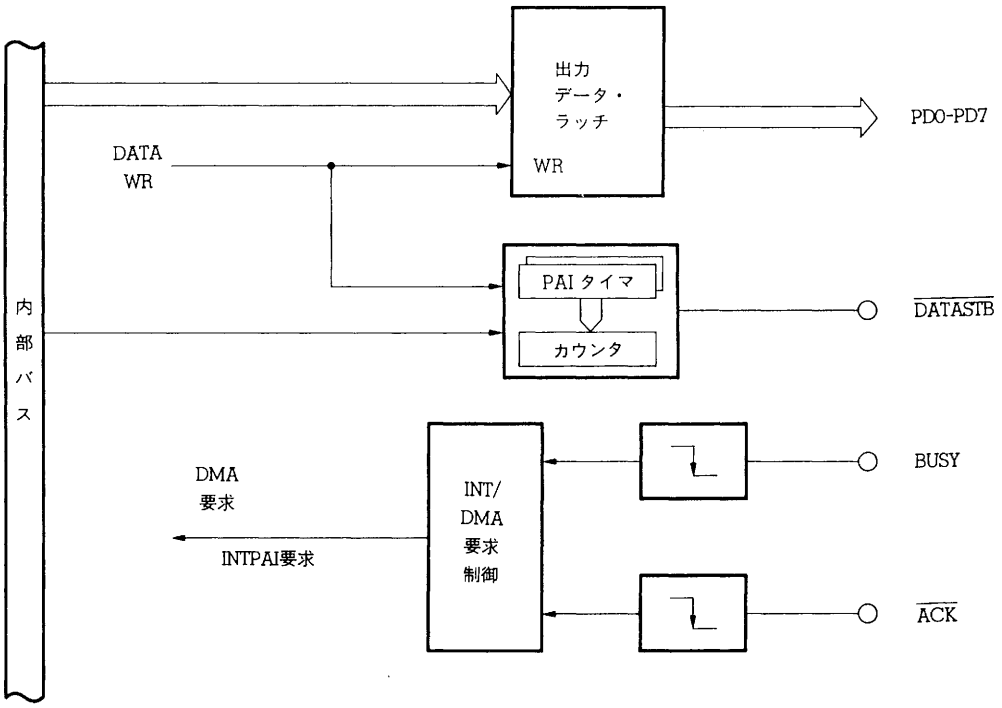
（a）入力モード



保守／廃止

図8-1　パラレル・インタフェースのブロック図（2/2）

（b）出力モード



保守／廃止

8.3 動　　作

入力，出力各モードでのインタフェース動作について説明します。

8.3.1 入力モード

PAC0レジスタのビット7（MOD）＝1にすることにより，入力モードとして動作します。

（1）基本動作

データの入力は次の順で行われます。

① $\overline{\text{DATASTB}}$信号の入力

$\overline{\text{DATASTB}}$信号の入力により，PD0-PD7端子のデータをPABレジスタにラッチします。PASレジスタのビット7がセット(1)されます。

② 割り込み要求，DMA要求の発生

PABレジスタにデータをラッチするタイミングでBUSY信号を出力し，DMA要求および割り込み要求（INTPAI）を発生します。

③ PABレジスタからのデータの読み出し

DMA転送，マクロ・サービスまたは命令操作により，CPU側でPABレジスタのデータを読み込みます。PASレジスタのビット7がクリアされます。

④ $\overline{\text{ACK}}$端子からの応答信号出力

PABレジスタのデータを読み込むタイミングで，$\overline{\text{ACK}}$信号を出力します。このとき内部PAIタイマがクリアされ，カウント動作を開始します。

⑤ $\overline{\text{ACK}}$出力の停止

（2）転送タイミング，制御信号出力タイミング

BUSY，$\overline{\text{ACK}}$信号の出力タイミングなどは，次に示すレジスタにより制御されます。詳細は，それぞれの参考箇所で説明してあります。

動　　作	レジスタ	ビット	参照箇所
BUSY信号出力タイミング	PAC0	ビット2（BSYT）	8.4.1参照
PABレジスタへのデータ取り込みタイミング	PAC0	ビット3（STBT）	8.4.1参照
$\overline{\text{ACK}}$，BUSY信号の制御	PAC1	ビット0，1（MB0，MB1）	表8-3参照
	PAC0	ビット0，1（ACKT0，ACKT1）	表8-2参照

保守／廃止

(3) PAIタイマ

内部PAIタイマは，PAI1，PAI2レジスタと常に比較を行い，BUSY信号と$\overline{\text{ACK}}$信号のタイミングを制御します。PAI1，PAI2レジスタには，$\overline{\text{ACK}}$信号のアクティブ幅に対応した値を設定してください（**8.4.4　パラレル・インタフェース・アクノリッジ・インターバル・レジスタ（PAI1，PAI2）**参照）。

8.3.2　出力モード

PAC0レジスタのビット7（MOD）=0にすることにより，出力モードとして動作します。

(1) 基本動作

データの出力は次の順で行われます。

①　PABレジスタへのデータの書き込み

DMA転送，マクロ・サービスまたは命令操作により，PABレジスタにデータを書き込みます。

②　PABレジスタのデータ出力

PABレジスタ内のデータをPD0-PD7端子から出力します。PASレジスタのビット6がセット(1)されます。

③　$\overline{\text{DATASTB}}$信号の出力

データ出力からT1時間経過後，$\overline{\text{DATASTB}}$信号を出力します。

なお，T1の長さはPAI1レジスタで，T2の長さ（$\overline{\text{DATASTB}}$信号のロウ・レベル幅）はPAI2レジスタで指定します。

④　割り込み要求，DMA要求の発生

$\overline{\text{DATASTB}}$信号出力後，BUSY端子または$\overline{\text{ACK}}$端子がハイ・レベルからロウ・レベルに変化すると，割り込み要求（INTPAI）およびDMA要求が発生します（**図8-1(b)**参照）。

各要求の発生タイミングにBUSY端子，$\overline{\text{ACK}}$端子のどちらを使用するか，または両方使用するのかを選択することができます。ポート5モード・コントロール・レジスタ（PMC5）により，表8-1のように設定します。

保守／廃止

表8－1　割り込み要求とDMA要求発生のトリガ端子指定（出力モード）

PMC51	PMC52	内　　容
1	0	$\overline{\text{ACK}}$端子だけを使用
0	1	BUSY端子だけを使用
1	1	$\overline{\text{ACK}}$端子，BUSY端子の両方を使用
0	0	$\overline{\text{ACK}}$端子，BUSY端子を使用しない（$\overline{\text{DATASTB}}$信号の出力直後に割り込み要求，DMA要求が発生

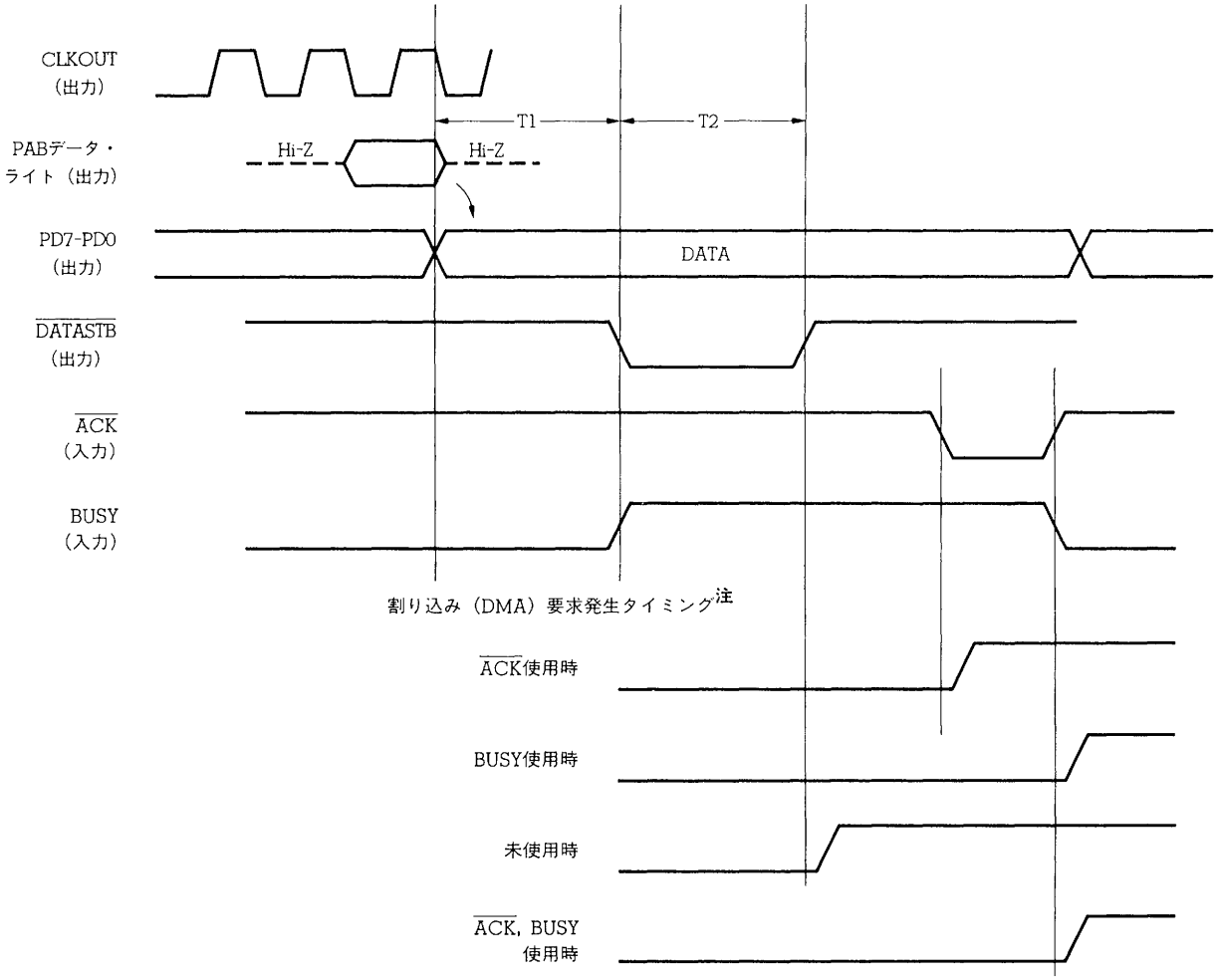
（2）ポート制御との相違

BUSY端子を兼用の汎用出力ポートに設定することによって(PMC52＝0, PM52＝0)，P52(BUSY端子）の出力レベルをパラレル・インタフェースの動作（PAC0，PAC1レジスタの設定）にかかわらず，P52に対する命令操作で任意に指定することができます。この場合，PAC1レジスタによるBUSY端子制御の場合と異なり，$\overline{\text{ACK}}$信号の自動出力，および割り込み要求，DMA要求の保留などの制御は行われません。

図8－2に出力モード時のタイミングを示します。

保守／廃止

図８－２　タイミング・チャート（出力モード時）



注　使用するトリガ端子により，タイミングが異なります。

保守／廃止

8.4　制御レジスタ

8.4.1　パラレル・インタフェース・コントロール・レジスタ0（PAC0）

入出力モード制御，$\overline{\text{DATASTB}}$信号，BUSY信号および$\overline{\text{ACK}}$信号の出力タイミング制御，データ取り込みタイミング制御，入力データの有効／無効制御を指定する8ビット・レジスタです。8/1ビット操作のメモリ・アクセスによって，書き込み／読み出しが可能です。

リセット後の値は，90Hになります。

	7	6	5	4	3	2	1	0
PAC0	MOD	EIN	ACKIN	ACKM	STBT	BSYT	ACKT1	ACKT0

表8-2，8-3参照

BUSY信号のセット・タイミングの指定[注]

BSYT	タイミング
1	$\overline{\text{DATASTB}}$信号の立ち上がり
0	$\overline{\text{DATASTB}}$信号の立ち下がり

入力データの取り込みタイミングの指定[注]

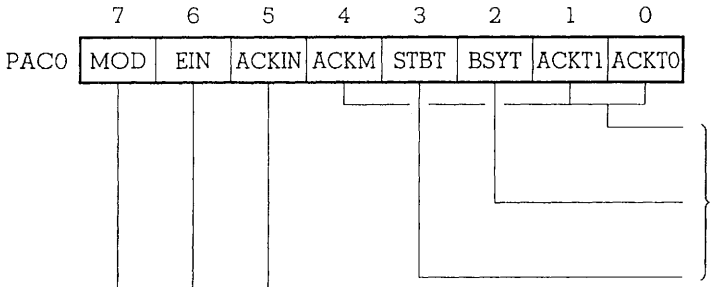
STBT	タイミング
1	$\overline{\text{DATASTB}}$信号の立ち上がり
0	$\overline{\text{DATASTB}}$信号の立ち下がり

次頁参照

注　入力モードでのみ有効です。

備考　□ はリセット後の状態を示します。

保守／廃止



$\overline{ACK}$信号出力中の$\overline{DATASTB}$入力に対する$\overline{ACK}$端子状態の指定[注]

ACKIN	状　態
1	$\overline{DATASTB}$で強制的にハイ・レベル
0	$\overline{ACK}$信号出力動作を継続

受信モード時のPABレジスタのオーバライト許可／禁止指定[注]

EIN	動　作
1	オーバライト許可
0	オーバライト禁止

パラレル・インタフェースの動作モード指定

MOD	入出力モード
0	出力モード
1	入力モード

注　入力モードでのみ有効です。

備考　□ はリセット後の状態を示します。

保守／廃止

表8－2　$\overline{ACK}$信号の出力指定（MBビットのリセット時）

ACKM	$\overline{ACK}$信号の出力
1	出力する注
0	出力しない

注　ただし，次の場合は$\overline{ACK}$信号を出力しません。

- PABレジスタに読み出されていない受信データがある場合（図8－13～図8－15参照）。
- $\overline{ACK}$信号の自動出力中にMB1, MB0ビットをリセットした場合（図8－22～図8－24参照）。出力中の$\overline{ACK}$信号が優先されます。

表8－3　ACKM，ACKT1，ACKT0の設定とBUSY信号および$\overline{ACK}$信号のタイミング

ACKM	ACKT1	ACKT0	BUSY信号リセット（L）・タイミング		$\overline{ACK}$信号出力タイミング	
			PABレジスタのデータ読み出し後	MB1, MB0ビットのリセット後	PABレジスタのデータ読み出し後	MB1, MB0ビットのリセット後
1	0	0	T1経過後リセット		直後にアクティブ，T1＋T2経過後インアクティブ	
	0	1	直後にリセット		直後にアクティブ，T1経過後インアクティブ	
	1	0	T1経過後リセット			
	1	1	設定禁止			
0	0	0	T1経過後リセット	直後にリセット	$\overline{ACK}$信号は出力されない	
	0	1	直後にリセット			
	1	0	T1経過後リセット			
	1	1	設定禁止			

備考　T1，T2の周期は，パラレル・インタフェース・アクノリッジ・インターバル・レジスタ（PAI1，PAI2）で設定します。

保守／廃止

MOD　パラレル・インタフェースの動作モード指定

パラレル・インタフェースを入力モードで使用する場合“1”，出力モードで使用する場合“0”に指定します。設定内容に応じて，ほかのフラグ機能およびパラレル・インタフェース・バッファ（PAB）の機能が切り替わります。設定内容を変更する場合は，必ずパラレル・インタフェース・コントロール・レジスタ1（PAC1）にもデータの再設定を行ってください。

MODビットを変更すると，次のようにユニットの動作は中断され初期化されます。なお，MODビットを変更してから内部回路が切り替わるまで，1.5クロックかかります。

- パラレル・インタフェース・ステータス・レジスタ（PAS）の内容が初期化されます。
- PAIタイマの動作（T1，T2のカウント）が停止します。
- パラレル・インタフェース・バッファ（PAB）の内容が不定となります。
- 各出力端子の信号（$\overline{ACK}$，$\overline{DATASTB}$）はイン・アクティブに戻ります。BUSY信号はハイ・レベルになります（PAC0，PAC1およびPAI1，PAI2は変化しません）。

EIN　受信モード時のPABレジスタのオーバライト許可／禁止指定

入力モードでのみ有効です。出力モードでは“0”を書き込んでください。

パラレル・インタフェース・ステータス・レジスタ（PAS）のバッファ・フル・フラグ（IBF）がセット(1)されている状態（PABレジスタ内に引き取られていない受信データが存在するとき）で，書き込みデータ・ストローブ信号（$\overline{DATASTB}$）が入力された場合のPABレジスタの動作を指定します。BUSY出力信号および割り込みとDMAの要求信号の発生には影響しません。

EIN＝0：PABレジスタの書き換え禁止モード

PABレジスタに前の受信データが引き取られていないまま残っている場合，受信データを命令（DMA）などで読み出すまではデータが保持され，データを書き込もうとしても無効になります。

EIN＝1：PABレジスタの書き換え許可モード

PABレジスタに前の受信データが残っている場合でも，$\overline{DATASTB}$信号入力があればPABレジスタの内容が更新されます。

パラレル・インタフェース・ステータス・レジスタ（PAS）のバッファ・フル・フラグ（IBF）がリセット(0)されている状態では，EINビットの内容に無関係に$\overline{DATASTB}$信号入力によってPABにデータが書き込まれます。

保守／廃止

ACKIN　**$\overline{\text{ACK}}$信号出力中の$\overline{\text{DATASTB}}$信号入力に対する$\overline{\text{ACK}}$端子状態の指定**

入力モードでのみ有効です。出力モードでは“0”を書き込んでください。

ACKIN＝1：$\overline{\text{ACK}}$信号出力動作中に$\overline{\text{DATASTB}}$信号入力があると，$\overline{\text{ACK}}$端子を強制的にハイ・レベル(インアクティブ)に戻します。また同時に内部PAIタイマによるT1，T2のカウント動作も停止，終了させます($\overline{\text{ACK}}$信号の出力，T1，T2のカウントを強制的に打ち切るモードです)。

ACKIN＝0：$\overline{\text{DATASTB}}$信号入力があっても，$\overline{\text{ACK}}$信号出力動作を継続して実行します。また，内部PAIタイマの動作にも影響を与えません。ただし，データは，PABレジスタに格納されます(**図8－28，図8－29**参照)。

ACKM　**MBビットのリセット時の$\overline{\text{ACK}}$出力指定**

入力モードでのみ有効です。出力モードでは“0”を書き込んでください。

(1) MBビットのリセット時の$\overline{\text{ACK}}$出力指定

命令，DMA，マクロ・サービスによりPABレジスタから受信データが引き取られるたびに，自動的に$\overline{\text{ACK}}$信号を1回出力します($\overline{\text{ACK}}$信号の自動出力)。これは，ACKMビットの影響を受けません。

ACKMビットは，PAC1レジスタのMB1ビットとMB0ビットをともにリセット(0)した場合にも，$\overline{\text{ACK}}$信号の出力を行うかどうかを指定します(**表8－2，図8－10～図8－12**参照)。

ACKM＝1にすることにより，初期設定時にBUSY信号を1→0にしたあとに続けて$\overline{\text{ACK}}$信号を出力させることなどができます。

(2) BUSY信号のリセット・タイミングの指定

ACKMビットの設定により，BUSY信号のリセット・タイミングが異なります(**表8－3**参照)。

STBT　**入力データの取り込みタイミングの指定**

入力モードでのみ有効です。出力モードでは“0”を書き込んでください。

入力データのPABレジスタへの取り込みを$\overline{\text{DATASTB}}$信号の立ち上がりエッジで行うか立ち下がりエッジで行うかを指定するビットです(**図8－3～図8－6**参照)。

保守／廃止

BSYT　BUSY信号のセット・タイミングの指定

入力モードでのみ有効です。出力モードでは“0”を書き込んでください。

$\overline{\text{DATASTB}}$信号入力により，BUSY端子から自動的にハイ・レベルを出力させるタイミングを$\overline{\text{DATASTB}}$信号の立ち上がりに同期させるか立ち下がりに同期させるかを指定するビットです。

BUSY信号のセット・タイミングは，STBTビットによるPABレジスタの入力データ取り込みタイミングとは独立に設定できます（図8-3～図8-6参照）。

ACKT1，ACKT0　BUSY信号リセット・タイミングと$\overline{\text{ACK}}$信号出力タイミングの指定

入力モードでのみ有効です。出力モードではどちらのビットも“0”を書き込んでください。

これらのビットで$\overline{\text{ACK}}$信号の出力タイミングを指定します。また，これらのビットとACKMビットの設定により，BUSY信号のリセット・タイミングを指定します（表8-3参照）。

8.4.2　パラレル・インタフェース・コントロール・レジスタ1（PAC1）

BUSY端子の制御を行う8ビットのレジスタです。

8/1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。

リセット後の値は03Hになります。

保守／廃止

	7	6	5	4	3	2	1	0
PAC1	0	0	0	0	0	INIT	MB1	MB0

BUSY端子の操作指定[注1]

MB1	MB0	状　　態
0	0	BUSY端子をロウ・レベルにする（ただしPABレジスタに受信データがある場合，設定直後はハイ・レベル，一定期間[注2]経過後ロウ・レベル）。
0	1	設定禁止
1	0	BUSY端子を一定期間[注3]経過後ハイ・レベルにする（設定直後はロウ・レベル）。
1	1	BUSY端子を強制的にハイ・レベルにする。

INTP5入力によるBUSY端子の動作指定[注1]

INIT	動　　作
1	INTP5入力に同期しBUSY端子を強制的にハイ・レベルにします。
0	動作は変化しません。

注1. 入力モードでのみ有効です。

2. 一定期間：PAC0レジスタのACKT1，ACKT0ビットの設定により次のようになります。

- ACKT1＝0，ACKT0＝1のとき

  PABレジスタに受信データが読み出されず残っている期間。

- ACKT1＝ACKT0＝0またはACKT1＝1，ACKT0＝0のとき

  PABレジスタに受信データが読み出されず残っている期間＋PABレジスタの読み出し後PAI1レジスタで指定した時間（T1）を経過するまでの期間。

3. 一定期間：PAC0レジスタのACKM＝1のときで，ACKT1，ACKT0ビットの設定により次のようになります。

- ACKT1＝0，ACKT0＝1のとき

  PABレジスタの読み出し後PAI1レジスタで指定した時間(T1)を経過するまでの期間。

- ACKT1＝ACKT0＝0またはACKT1＝1，ACKT0＝0のとき

  PAI2レジスタで指定した時間（T2）を経過するまでの期間。

備考 □ はリセット後の状態を示します。

保守／廃止

MB1，**MB0**　**BUSY端子の操作指定**

入力モードでのみ有効です。出力モードでは“00”を書き込んでください。

BUSY端子をハイ・レベルまたはロウ・レベルに指定するためのビットです。

$\overline{\text{DATASTB}}$信号が入力されたときの動作を次に示します。

●MB1＝MB0＝0のとき

入力データをPABレジスタへ格納します。

その後BUSY端子は，PAC0レジスタのBSYTビットで指定したタイミングで自動的にハイ・レベルになり，割り込み要求（INTPAI），DMA要求を発生します。

●MB1＝1，MB0＝0またはMB1＝MB0＝1のとき

入力データをPABレジスタへ格納します。

割り込み要求（INTPAI），DMA要求は，MB1＝MB0＝0になるまで保留されます。

INIT　**INTP5入力によるBUSY端子の動作指定**

入力モードでのみ有効です。出力モードでは“0”を書き込んでください。

外部割り込みINTP5の入力によりBUSY端子を自動的にハイ・レベルにすることを許可／禁止するビットです。

INIT＝0：INTP5入力があっても，このユニットの動作は変化しません。INTP5は単に外部割り込み入力端子としてのみ機能します。

INIT＝1：INTP5入力（有効エッジ入力）に同期してBUSY端子を強制的にハイ・レベルにします。INTP5入力があってもMB1，MB0ビットおよびPABレジスタの状態には影響を与えません。また$\overline{\text{ACK}}$信号出力およびPAIタイマのカウント動作（T1，T2のカウント）も継続して実行します。

INTP5入力によりBUSY端子がハイ・レベルになったことは，INTP5割り込みの発生およびパラレル・インタフェース・ステータス・レジスタ（PAS）のIBSYフラグを読み出すことで知ることができます。

保守／廃止

8.4.3　パラレル・インタフェース・ステータス・レジスタ（PAS）

パラレル・インタフェース・ユニットの内部状態を示すステータス・フラグが割り付けられています。

8ビットのメモリ・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。

リセットおよび入出力モードの切り替え（PAC0レジスタのMODビットの操作）後の値は40Hになります。

	7	6	5	4	3	2	1	0
PAS	IBF	OBE	0	0	IBSY	0	0	ERR

エラー・フラグ[注1]

ERR	状　態
1	IBF＝1のとき$\overline{\text{DATASTB}}$入力でセット
0	IBF＝0のとき$\overline{\text{DATASTB}}$入力でリセット

INTP入力によるBUSY端子の状態[注1]

IBSY	BUSY端子の状態
1	INTP5入力でBUSY端子がハイ・レベル
0	BUSY端子はロウ・レベル

バッファ・エンプティ・フラグ[注2]

OBE	PABレジスタの状態
1	データを送信した状態
0	データが書き込まれた状態

バッファ・フル・フラグ[注1]

IBF	PABレジスタの状態
1	データを受信した状態
0	データがない状態

注1.　入力モードでのみ有効です。

2.　出力モードでのみ有効です。

備考　□　はリセット後の状態を示します。

IBF　**バッファ・フル・フラグ**

入力モードでのみ有効です。出力モードでは不定です。

パラレル・インタフェース・バッファ（PAB）内に受信データがあるか否かを示すフラグです。

$\overline{\text{DATASTB}}$信号によりPABレジスタへデータが入力されたときセット(1)，PABレジスタを読み出したときリセット(0)されます。

保守／廃止

OBE バッファ・エンプティ・フラグ

出力モードでのみ有効です。入力モードでは不定です。

PABレジスタの送信データをポートPD0 - PD7より出力したか否かを示すフラグです。命令でPABレジスタに出力データを書き込んだときにリセット(0)され，PABレジスタのデータをポートPD0 - PD7に出力し，PABレジスタが空になるとセット(1)されます。

IBSY ビジィ・フラグ

入力モードでのみ有効です。出力モードでは不定です。

外部割り込み端子INTP5入力（有効エッジ入力）により，BUSY端子がハイ・レベルとなっていることを示すステータス・フラグです。コントロール・レジスタ1（PAC1）でINIT＝1に設定すると，外部割り込み端子INTP5の割り込み信号入力に同期しIBSYフラグがセット(1)され，同時にBUSY端子がハイ・レベルになります。

このフラグに“1”を書き込む命令を実行することにより，リセット(0)することができます。ただし“0”を書き込む命令を実行してもフラグ内容は変化しません。

このフラグをリセット(0)しないと，BUSY端子はロウ・レベルに戻りません。このフラグがセット(1)されているときには，受信データの読み出しおよびコントロール・レジスタ1（PAC1）のMB1，MB0ビットのリセット(00)を行っても，BUSY端子はロウ・レベルになりません。

ERR エラー・フラグ

入力モードでのみ有効です。出力モードでは不定です。

PABレジスタから受信データを読み出す以前に（IBF＝1のとき）次の$\overline{\text{DATASTB}}$信号入力があるとセット(1)されるステータス・フラグです。コントロール・レジスタ0（PAC0）のEINビットの指定にかかわらず上記の動作を行います。

IBF＝0のときの$\overline{\text{DATASTB}}$信号入力によって自動的にリセット(0)されます。

このフラグに“1”を書き込む命令を実行することにより，リセット(0)することができます。ただし“0”を書き込む命令を実行してもフラグ内容は変化しません。

保守／廃止

8.4.4　パラレル・インタフェース・アクノリッジ・インターバル・レジスタ（PAI1，PAI2）

8ビット操作のメモリ・アクセスによってレジスタの書き込みだけができます。

リセット後のPAI1，PAI2レジスタの値は不定になります。

これらのレジスタは，入力モード時と出力モード時で，指定する対象が次のように異なります。

(1) 入力モード

$\overline{ACK}$信号のアクティブ幅を次のように指定します。アクティブ幅は，PAC0レジスタのACKTビットの値によりT1またはT1＋T2で表します。内蔵PAIタイマのカウント時間T1はPAI1レジスタで，T2はPAI2レジスタで指定します。

●ACKT0＝0，ACKT1＝1またはACKT0＝1，ACKT1＝0のとき

$T1 = 2/\phi \times$PAI1設定値

●ACKT0＝0，ACKT1＝0のとき

$T1 + T2 =$（$2/\phi \times$PAI1設定値）＋（$2/\phi \times$PAI2設定値）

表8－4に，ϕ＝12.5 MHz，16 MHzのそれぞれの場合の代表的なPAI設定値と$\overline{ACK}$出力幅の関係を示します。

表8－4　代表的なPAI設定値と$\overline{ACK}$出力幅

(1) ϕ＝12.5 MHzの場合（単位：μs）

PAIn	$\overline{ACK}$出力幅
0	40.96
1	0.16
3	0.48
6	0.96
13	2.08
25	4.0
31	4.96
50	8.0
125	20.0

(n＝1，2)

(2) ϕ＝16 MHzの場合（単位：μs）

PAIn	$\overline{ACK}$出力幅
0	32.0
1	0.125
4	0.5
8	1.0
16	2.0
32	4.0
40	5.0
64	8.0
128	16.0

(n＝1，2)

保守／廃止

(2) 出力モード

$\overline{\text{DATASTB}}$信号の出力遅延時間（T1）と$\overline{\text{DATASTB}}$信号のアクティブ幅（T2）を次のように指定します。内蔵PAIタイマのカウント時間T1はPAI1レジスタで，T2はPAI2レジスタで指定します。

T1＝2/ϕ×PAI1設定値

T2＝2/ϕ×PAI2設定値

8.4.5 パラレル・インタフェース・バッファ・レジスタ（PAB）

8ビットのメモリ操作命令により読み出し／書き込みを行うことができます。

(1) 入力モード時

読み出しのみ可能です。書き込みはできません。

$\overline{\text{DATASTB}}$信号入力と同期してデータ入出力端子PD0 - PD7上のデータをラッチする入力バッファ・レジスタとして機能します。

このレジスタにデータが書き込まれたときIBFフラグはセット(1)され，データが読み出されたときIBFフラグはリセット(0)されます。

(2) 出力モード時

書き込みのみ可能です。読み出すと不定となります。

$\overline{\text{DATASTB}}$信号と同期してデータ出力端子PD0 - PD7上に出力するデータをラッチする出力バッファ・レジスタとして機能します。

このレジスタに命令でデータを書き込むとOBEフラグはリセット(0)され，本レジスタからPD0 - PD7上にデータが出力されたときOBEフラグはセット(1)されます。

8.5 割り込み要求

8.5.1 入力モード時

$\overline{\text{DATASTB}}$信号の入力によりデータをパラレル・インタフェース・バッファ（PAB）に取り込むと同時に，割り込み要求信号INTPAIとDMA要求信号を発生します。

ただし，PABレジスタへのデータ取り込み時にMB1，MB0ビット＝1×となっており，BUSY端子がハイ・レベルになっていれば，MB1，MB0ビットをリセット(1× → 00)するまで，割り込み要求信号INTPAIおよびDMA要求信号の発生を保留します。

8.5.2 出力モード時

PABレジスタに書き込んだデータを$\overline{\text{DATASTB}}$信号とともに出力後，外部よりBUSY端子および$\overline{\text{ACK}}$端子に有効な応答信号が入力されると，割り込み要求信号INTPAIおよびDMA要求信号を発生します。

割り込み要求信号INTPAIの発生タイミングとDMA要求信号の発生タイミングは同時です。

保守／廃止

8.6　動作タイミング

パラレル・インタフェース・ユニットの動作タイミングを次に示します。

図番号	モード	ACKT1, 0	ACKM	動作条件：MB1，MB0ビットの設定条件他
図8-3	入力	00	—	STBT，BSYTビットの設定によるBUSY信号の出力およびDMA転送要求，割り込み要求の発生タイミング指定
図8-4	入力	00	—	
図8-5	入力	00	—	
図8-6	入力	00	—	
図8-7	入力	00	—	ACKT1，ACKT0ビットの指定による$\overline{\text{ACK}}$信号出力
図8-8	入力	01	—	
図8-9	入力	10	—	
図8-10	入力	00	1/0	MB1，MB0ビットのリセットによる$\overline{\text{ACK}}$信号出力 MB1，MB0：1× → 00
図8-11	入力	01	1/0	
図8-12	入力	10	1/0	
図8-13	入力	00	—	MB1，MB0ビットのセット後に$\overline{\text{DATASTB}}$信号入力があった場合，割り込み（DMA）要求の保留動作
図8-14	入力	01	—	
図8-15	入力	10	—	
図8-16	入力	00	—	
図8-17	入力	01	—	
図8-18	入力	10	—	
図8-19	入力	00	—	MB1，MB0ビットのセット前に$\overline{\text{DATASTB}}$信号入力があった場合
図8-20	入力	01	—	
図8-21	入力	10	—	
図8-22	入力	00	—	
図8-23	入力	01	—	
図8-24	入力	10	—	
図8-25	入力	00	1/0	MB1，MB0ビットのリセットによる$\overline{\text{ACK}}$信号出力 MB1，MB0ビットのセット前に$\overline{\text{DATASTB}}$信号入力があった場合
図8-26	入力	01	1/0	
図8-27	入力	10	1/0	
図8-28	入力	00	0	連続して$\overline{\text{DATASTB}}$信号入力があった場合（ACKIN＝0）
図8-29	入力	01	0	
図8-30	入力	10	0	
図8-31	入力	00	0	連続して$\overline{\text{DATASTB}}$信号入力があった場合（ACKIN＝0） MB1，MB0の指定との競合
図8-32	入力	01	0	
図8-33	入力	10	0	

保守／廃止

（BUSY端子▽，PAB▼）

図８－３　タイミング・チャート（STBT＝0，BSYT＝0）

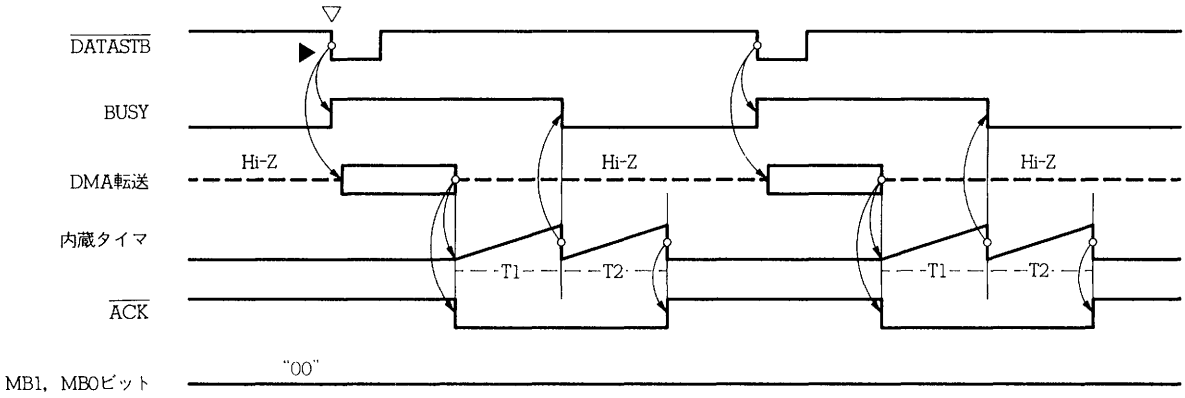


図８－４　タイミング・チャート（STBT＝1，BSYT＝0）

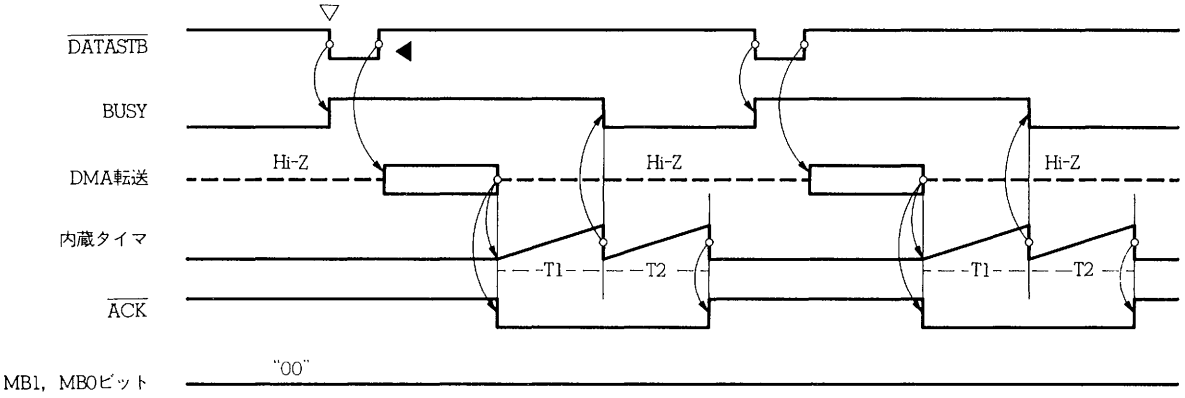
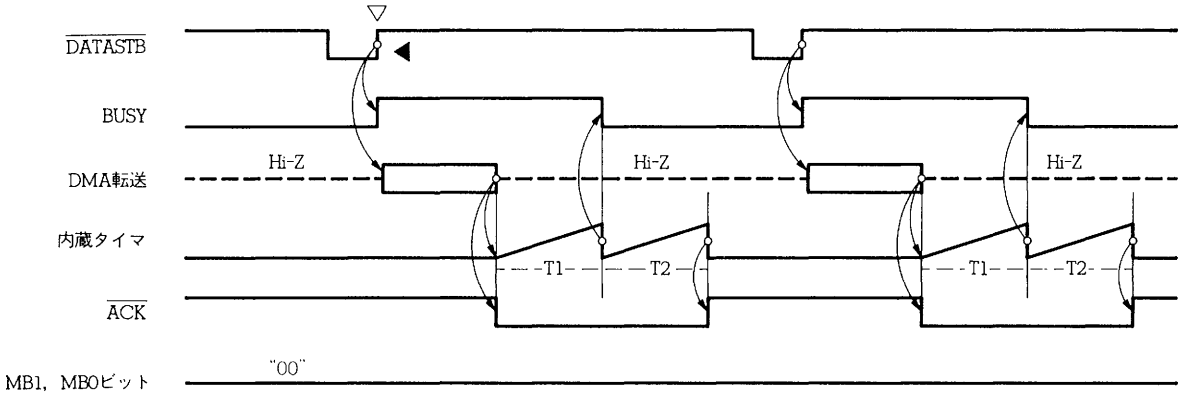
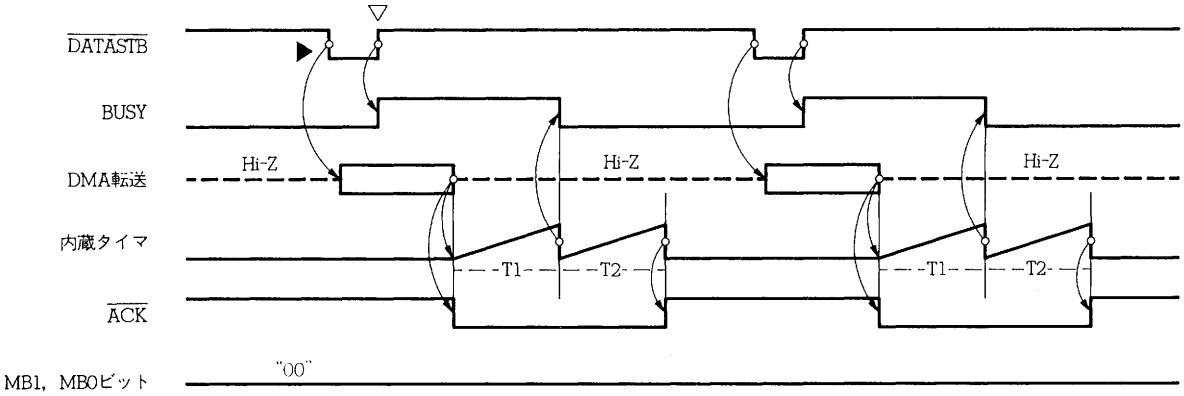


図８－５　タイミング・チャート（STBT＝1，BSYT＝1）



保守／廃止

図８－６　タイミング・チャート（STBT＝0，BSYT＝1）



保守／廃止

図８－７　タイミング・チャート（ACKT1，ACKT0＝00）

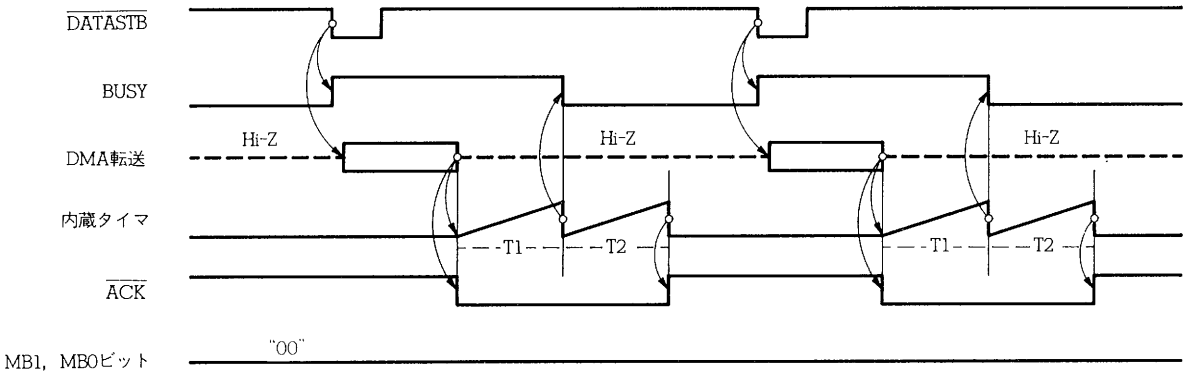


図８－８　タイミング・チャート（ACKT1，ACKT0＝01）

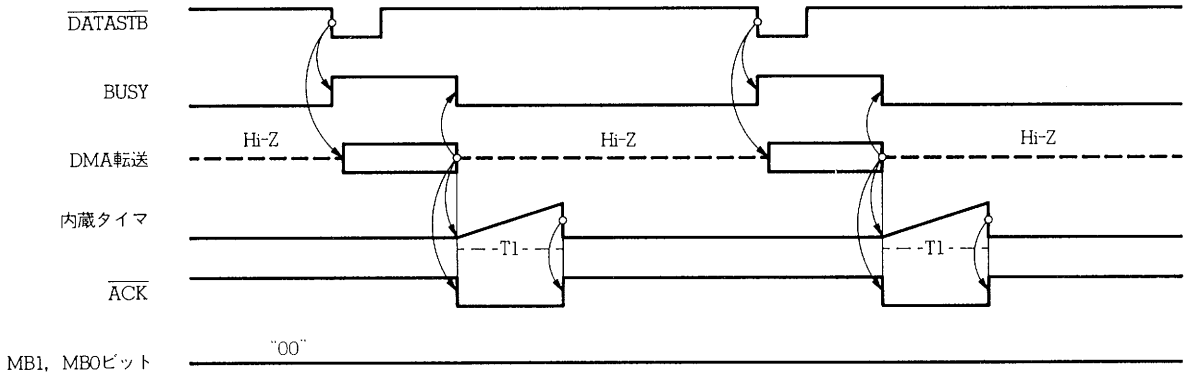
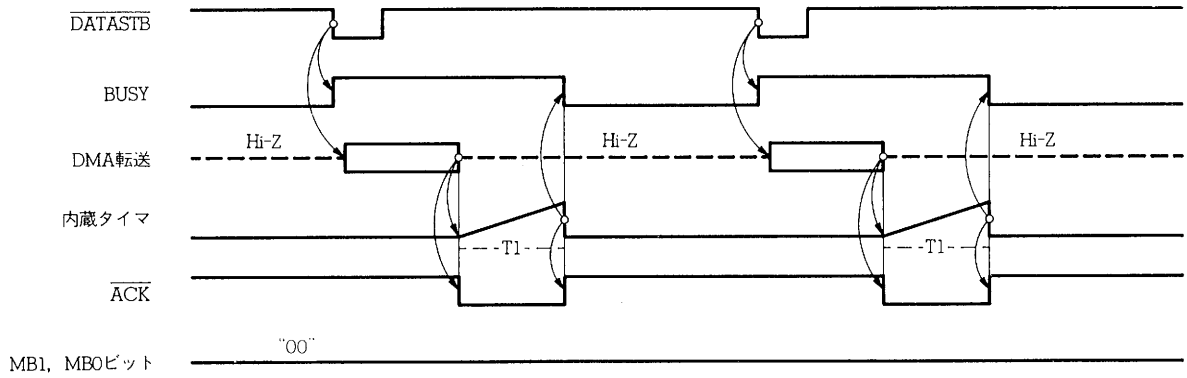


図８－９　タイミング・チャート（ACKT1，ACKT0＝10）



保守／廃止

図8-10　タイミング・チャート（ACKT1，ACKT0=00）

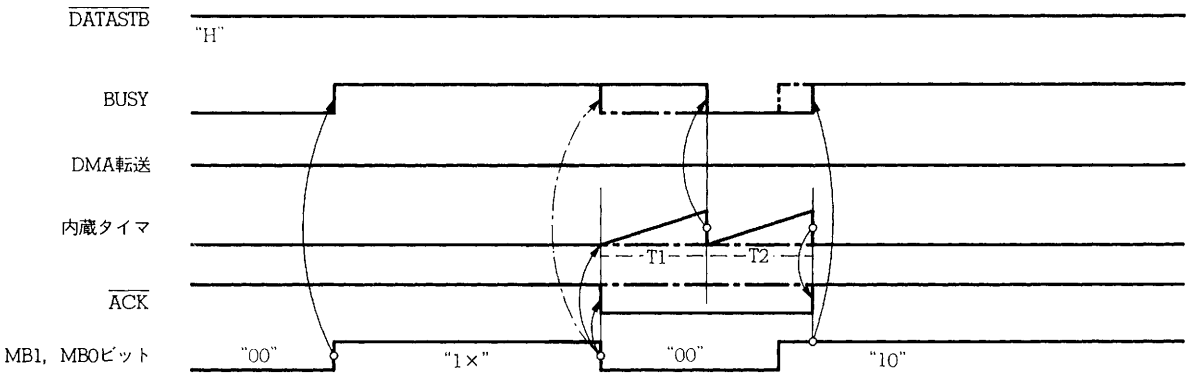


図8-11　タイミング・チャート（ACKT1，ACKT0=01）

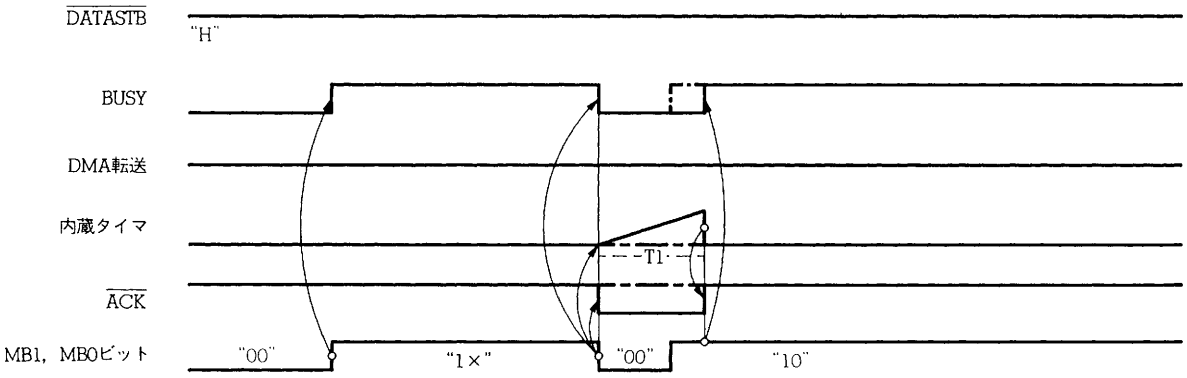
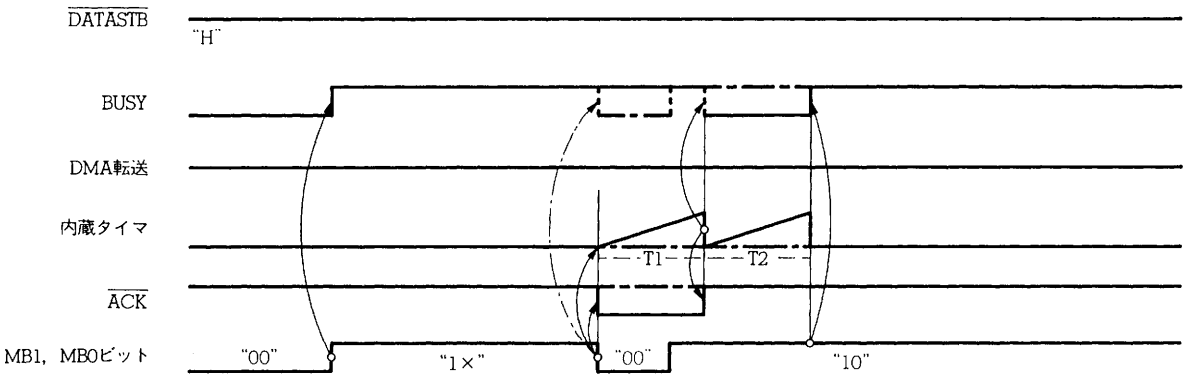


図8-12　タイミング・チャート（ACKT1，ACKT0=10）



備考　－－－：ACKM=0

——：ACKM=1

保守／廃止

図8－13　タイミング・チャート（ACKT1，ACKT0＝00）

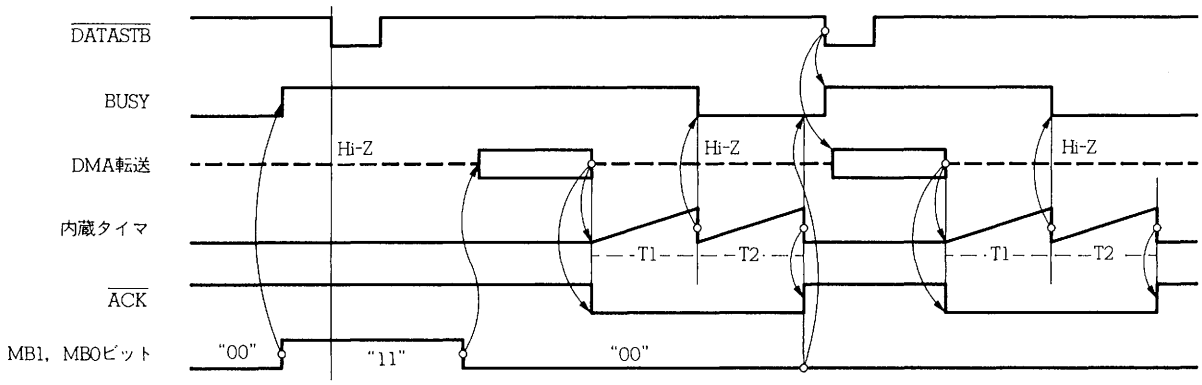


図8－14　タイミング・チャート（ACKT1，ACKT0＝01）

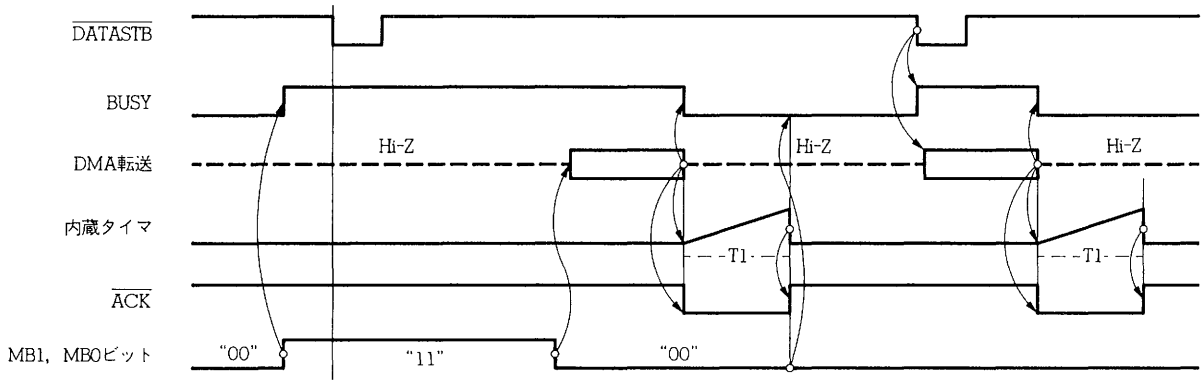
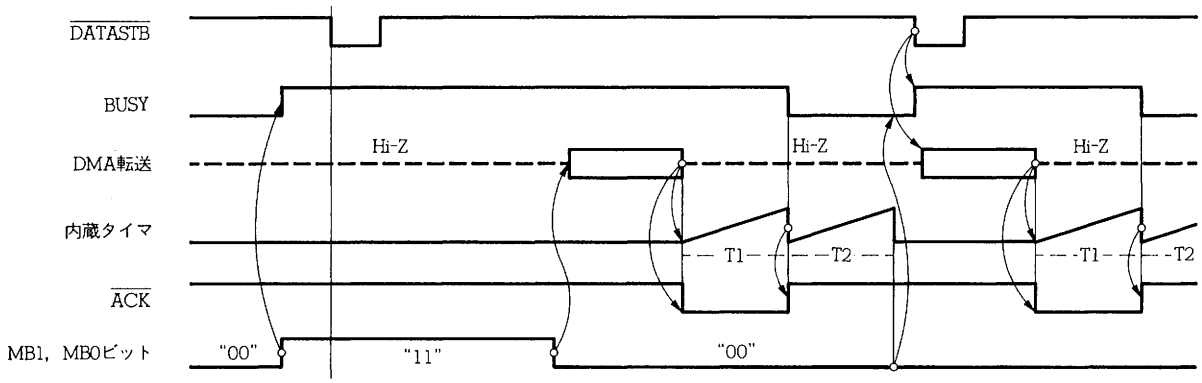


図8－15　タイミング・チャート（ACKT1，ACKT0＝10）



保守／廃止

図8-16　タイミング・チャート（ACKT1，ACKT0＝00）

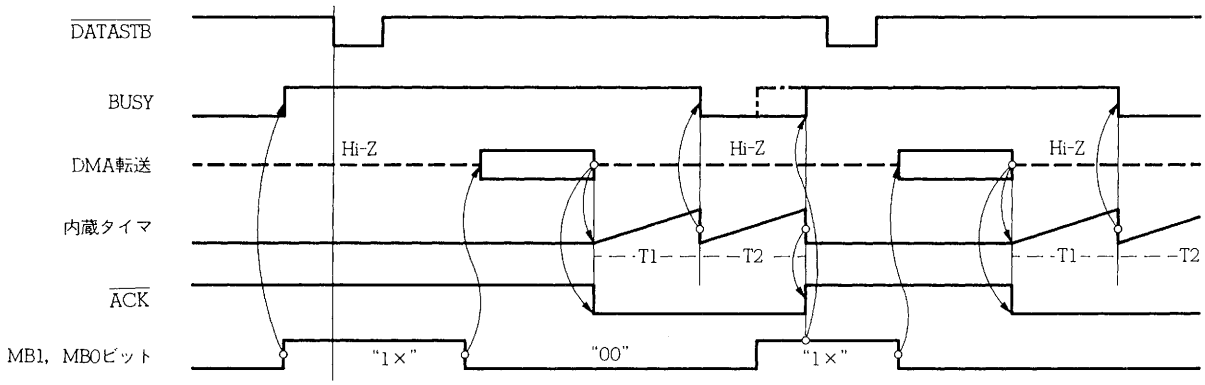


図8-17　タイミング・チャート（ACKT1，ACKT0＝01）

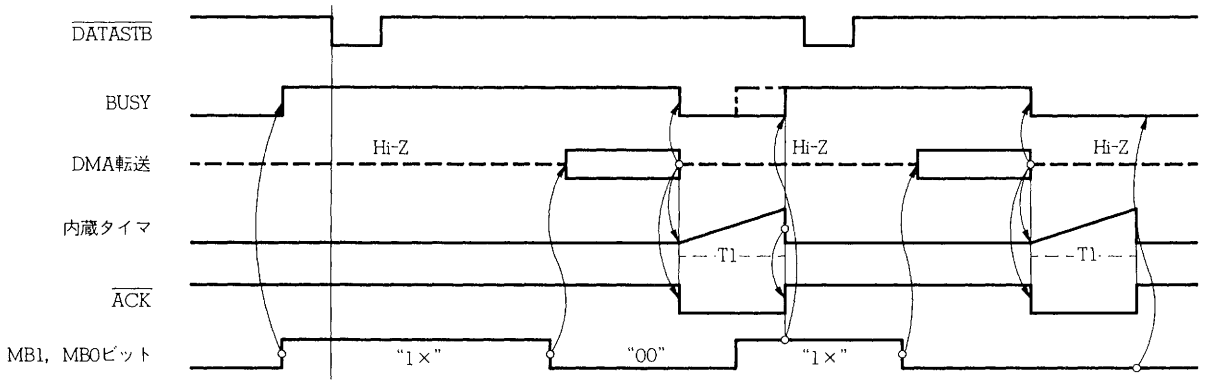
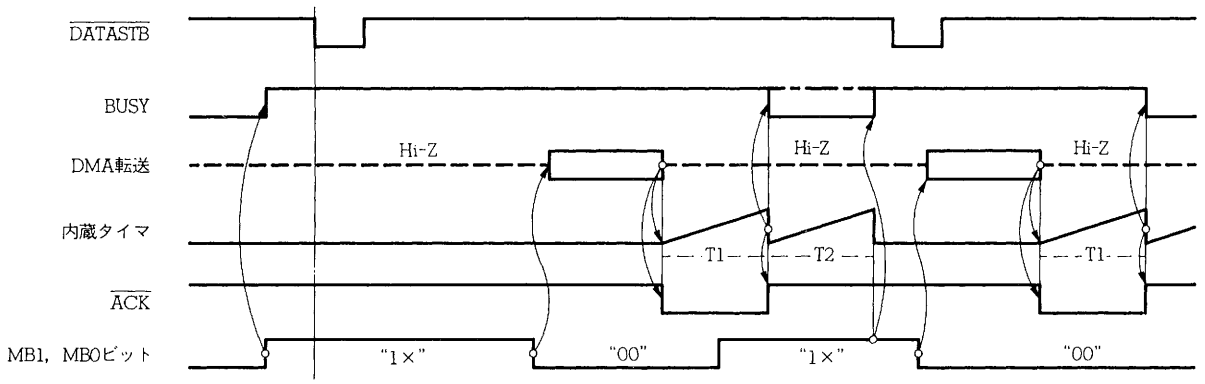


図8-18　タイミング・チャート（ACKT1，ACKT0＝10）



備考　——　MB1，MB0：“10”に設定したとき

　　－－－　MB1，MB0：“11”に設定したとき

保守／廃止

図８－19　タイミング・チャート（ACKT1，ACKT0＝00）

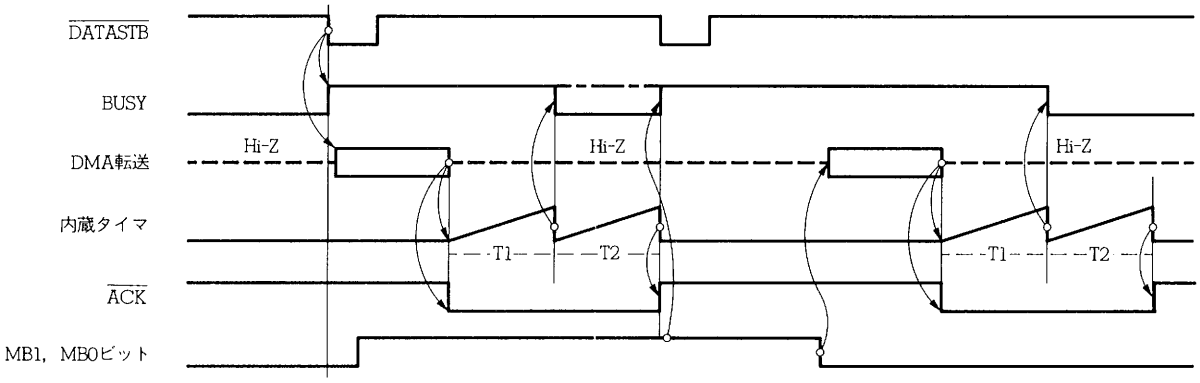


図８－20　タイミング・チャート（ACKT1，ACKT0＝01）

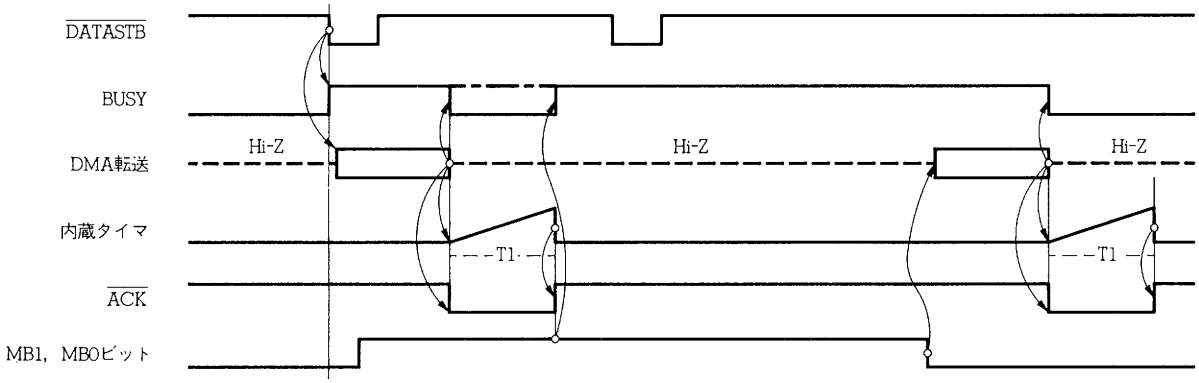
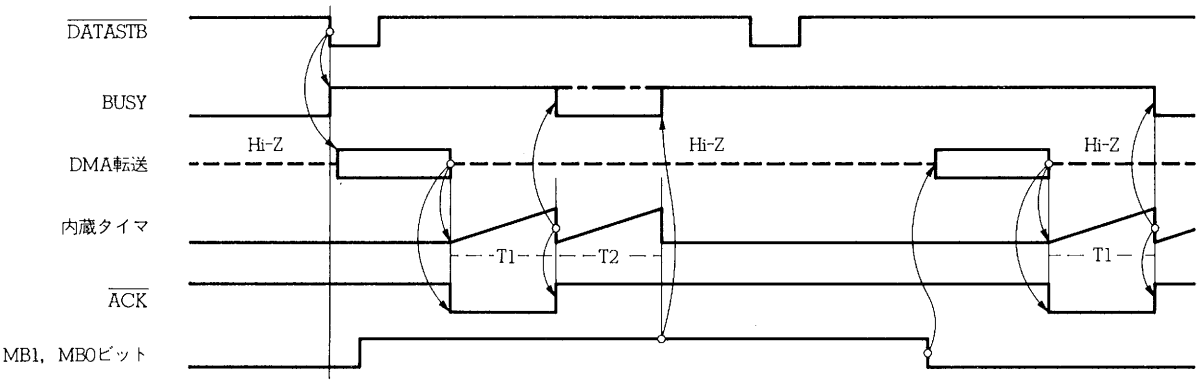


図８－21　タイミング・チャート（ACKT1，ACKT0＝10）



備考　――― MB1，MB0：“10”に設定したとき

　　－－－ MB1，MB0：“11”に設定したとき

保守／廃止

図8-22　タイミング・チャート（ACKT1，ACKT0=00）

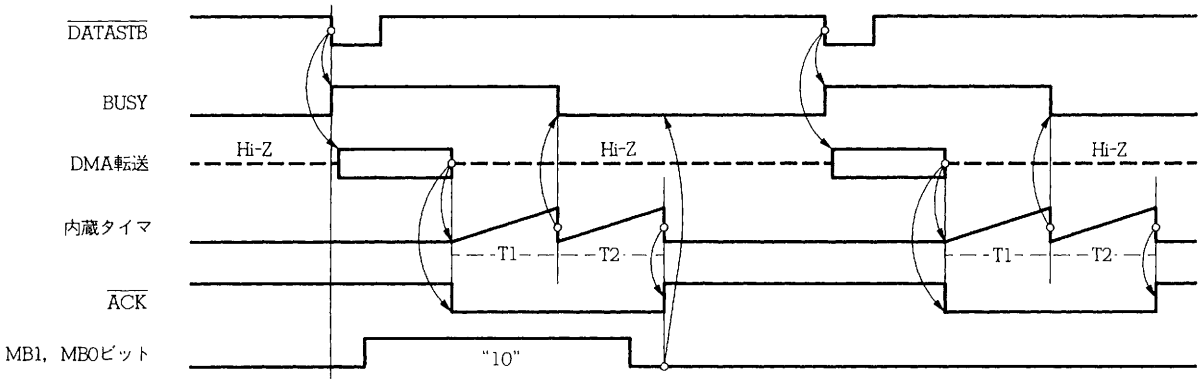


図8-23　タイミング・チャート（ACKT1，ACKT0=01）

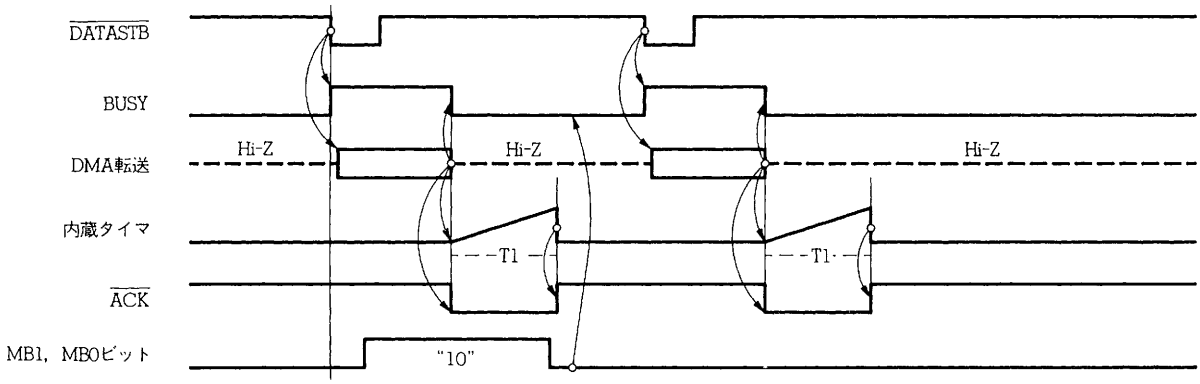
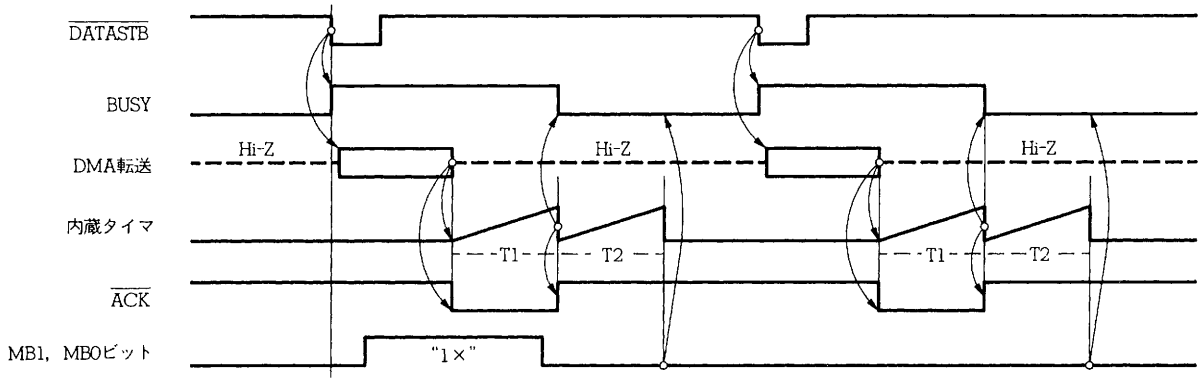


図8-24　タイミング・チャート（ACKT1，ACKT0=10）



保守／廃止

図8－25　タイミング・チャート（ACKT1，ACKT0＝00）

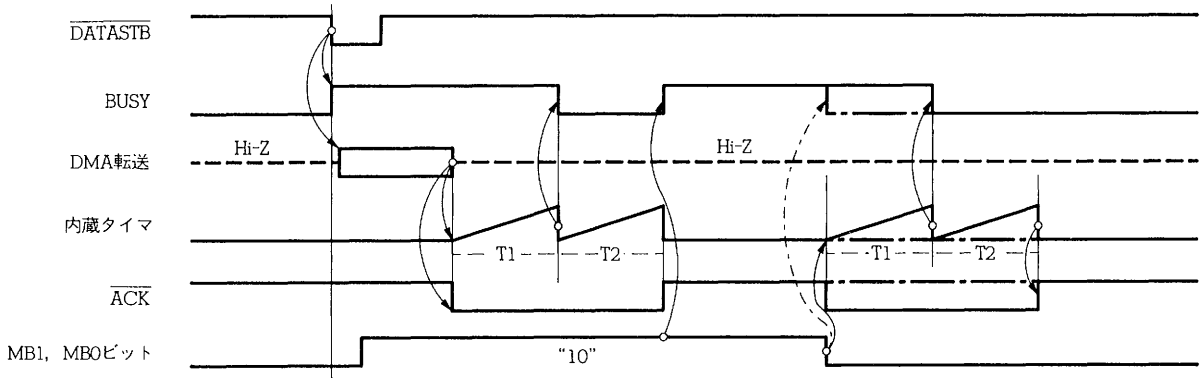


図8－26　タイミング・チャート（ACKT1，ACKT0＝01）

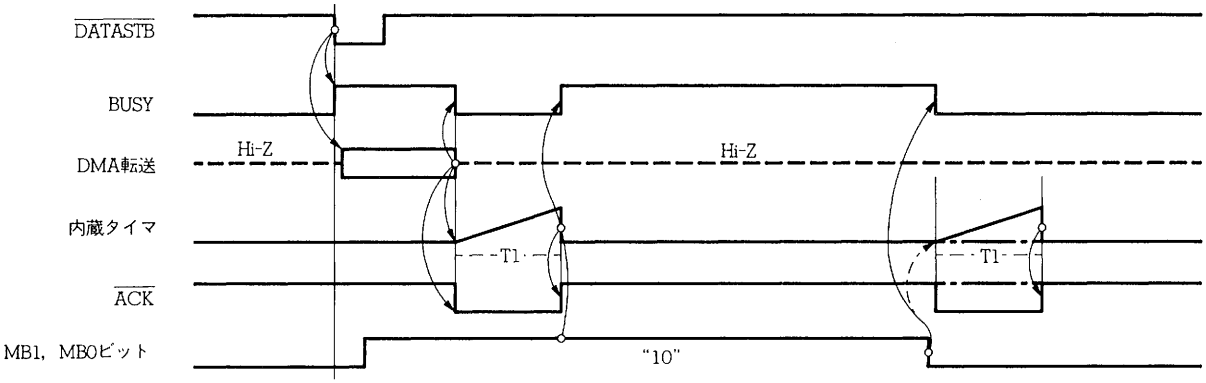
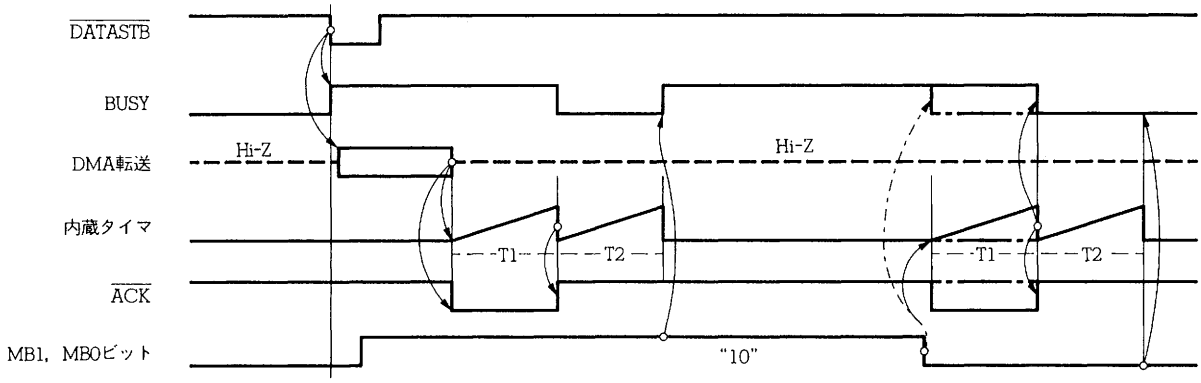


図8－27　タイミング・チャート（ACKT1，ACKT0＝10）



備考　－－－　ACKM＝0

　　——　ACKM＝1

保守／廃止

図8－28　タイミング・チャート（ACKT1，ACKT0＝00，ACKM＝0）

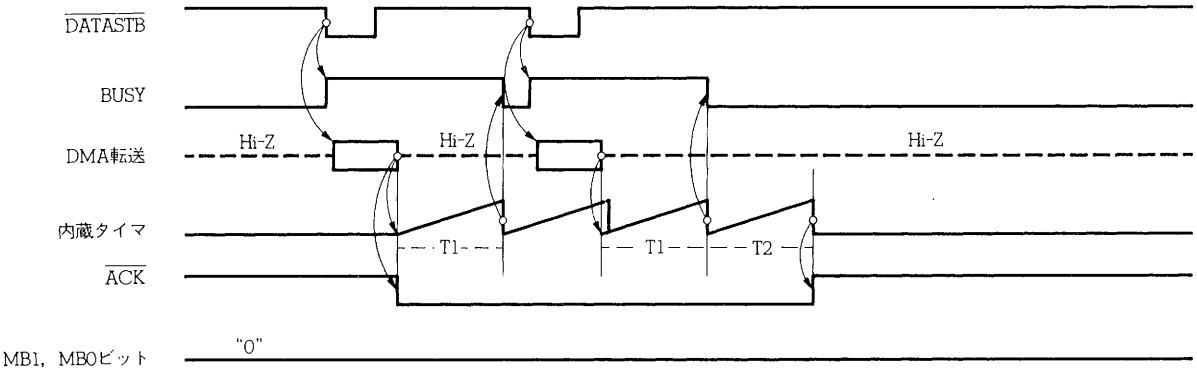


図8－29　タイミング・チャート（ACKT1，ACKT0＝01，ACKM＝0）

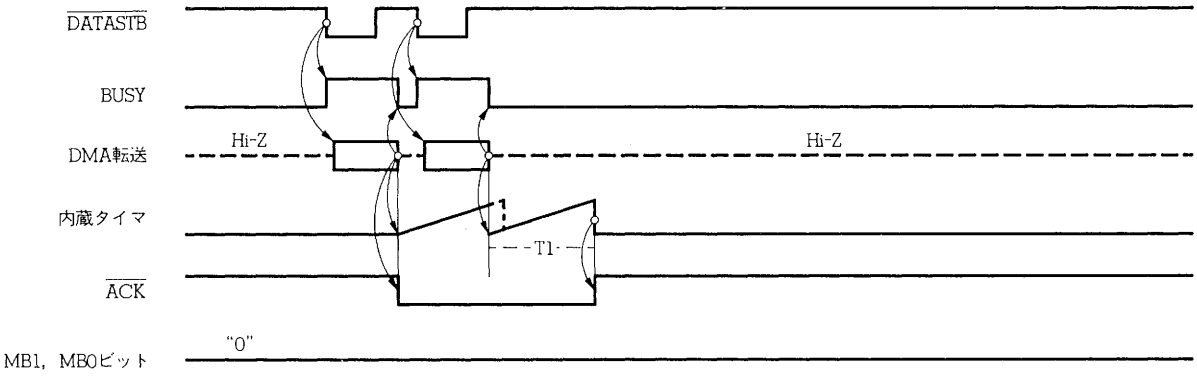
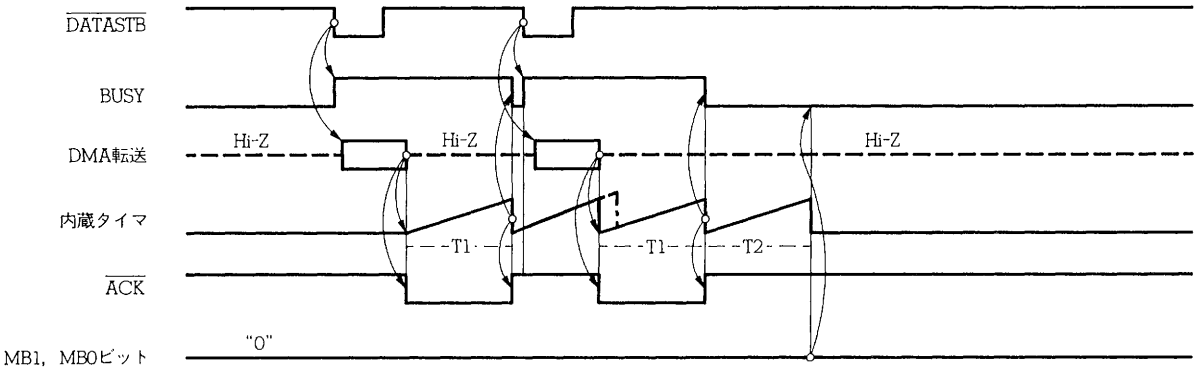


図8－30　タイミング・チャート（ACKT1，ACKT0＝10，ACKM＝0）



保守／廃止

図８－31　タイミング・チャート（ACKT1，ACKT0＝00，ACKM＝0）

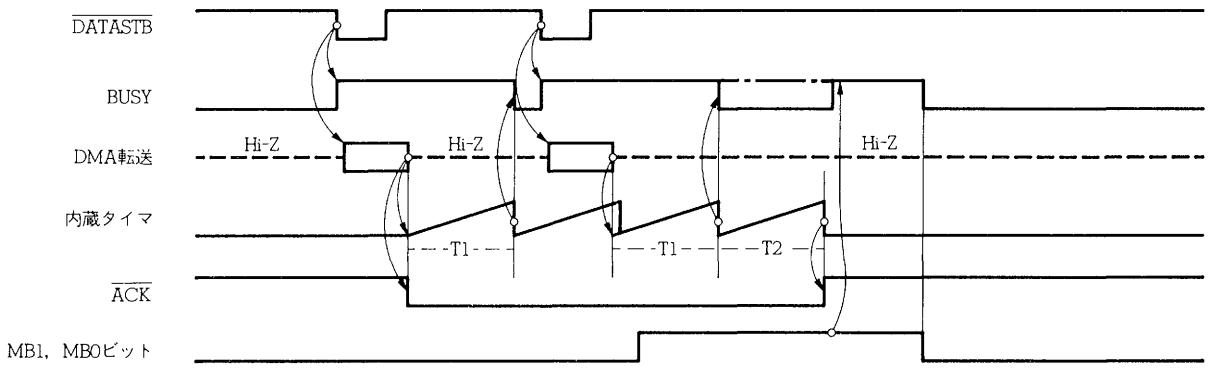


図８－32　タイミング・チャート（ACKT1，ACKT0＝01，ACKM＝0）

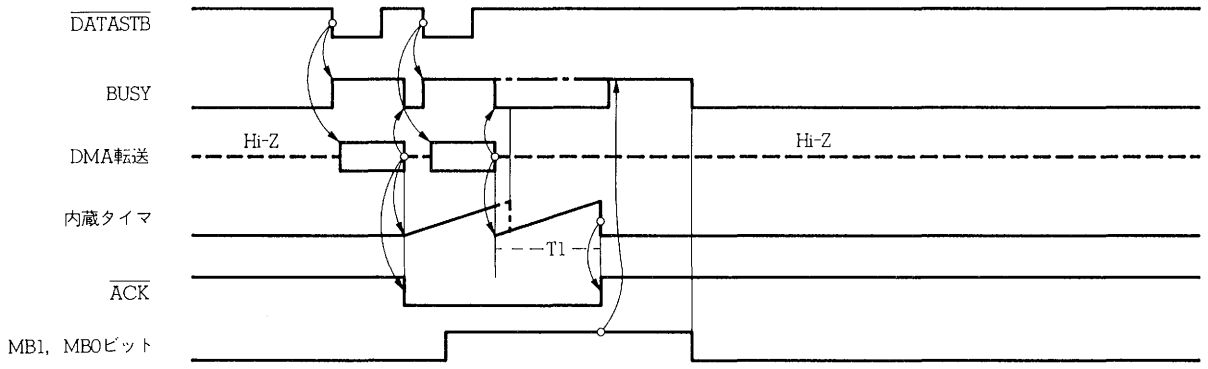
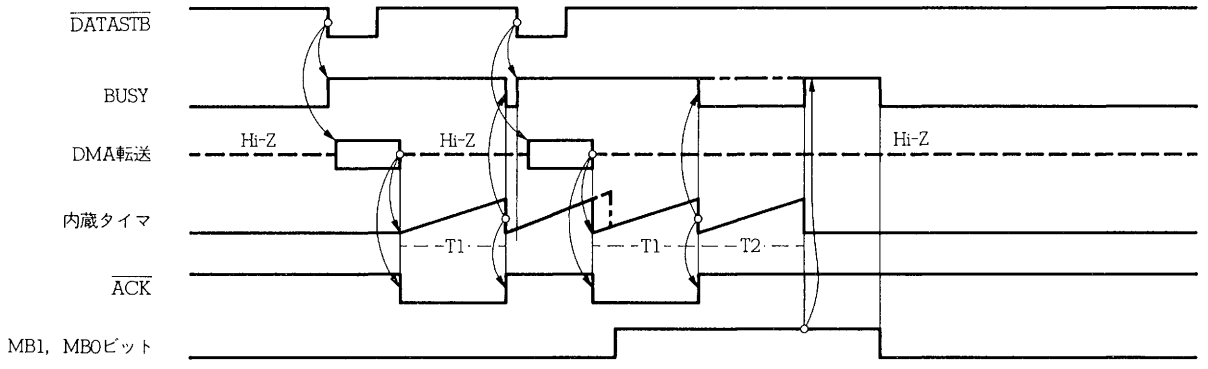


図８－33　タイミング・チャート（ACKT1，ACKT0＝10，ACKM＝0）



備考　――― MB1，MB0：“10”に設定したとき

　　－－－ MB1，MB0：“11”に設定したとき

〔メ　モ〕

保守／廃止

第９章　タイマ機能

V55PIのタイマ・ユニットはインターバル・タイマ，フリー・ランニング・タイマおよびイベント・カウンタとして用いることができます。また，タイマから発生する割り込み要求に同期させ，P7をリアルタイム出力ポートとして操作することもできます。ここでは，通常のタイマ機能について説明します。

9.1　特　　徴

○16ビット・タイマ×4

○カウント・クロックのソース２種

●システム・クロックの分周出力（ϕ/8，ϕ/32：システム・クロックϕ）を選択可能

●TI端子からの外部入力パルスを選択可能

○外部カウント出力信号（TOn出力）

○16ビット・キャプチャ・レジスタを３本内蔵（外部割り込み入力信号INTP0-INTP2をトリガ）

○タイマ・ユニット専用割り込みソース６種

9.2　構　　成

タイマ・ユニットの構成を図９－１に示します。タイマ・ユニットは大きく４つの独立した16ビット・タイマで構成されます。各タイマは，次のように使用することができます。

表９－１　タイマ・ユニットの種類

		タイマ０	タイマ１	タイマ２	タイマ３
種類	フリー・ランニング・タイマ	○	○	—	—
	イベント・カウンタ	—	○	—	—
	インターバル・タイマ	—	○	○	○
出力	割り込み要求数	2	2	1	1
	リアルタイム出力トリガ	○	—	—	—
	外部出力端子数	2	—	2	1

保守／廃止

(1) タイマ０

タイマ０は，16ビットのタイマ・レジスタ０（TM0），16ビットのコンペア・レジスタ（CM00，CM01），16ビットのキャプチャ・レジスタ（CT00，CT01）から構成され，16ビットのフリー・ランニング・タイマとして機能します。

(2) タイマ１

タイマ１は，16ビットのタイマ・レジスタ１（TM1），16ビットのコンペア・レジスタ（CM10，CM11），16ビットのキャプチャ・レジスタ（CT10）から構成され，16ビットのフリー・ランニング・タイマ，イベント・カウンタおよびインターバル・タイマとして機能します。

CM10一致信号でのカウンタ・クリア動作を禁止することによって，フリー・ランニング・タイマとして使用可能です。

CM10一致信号でのカウンタ・クリア動作により，16ビット・インターバル・タイマ機能を持ちます。

(3) タイマ２

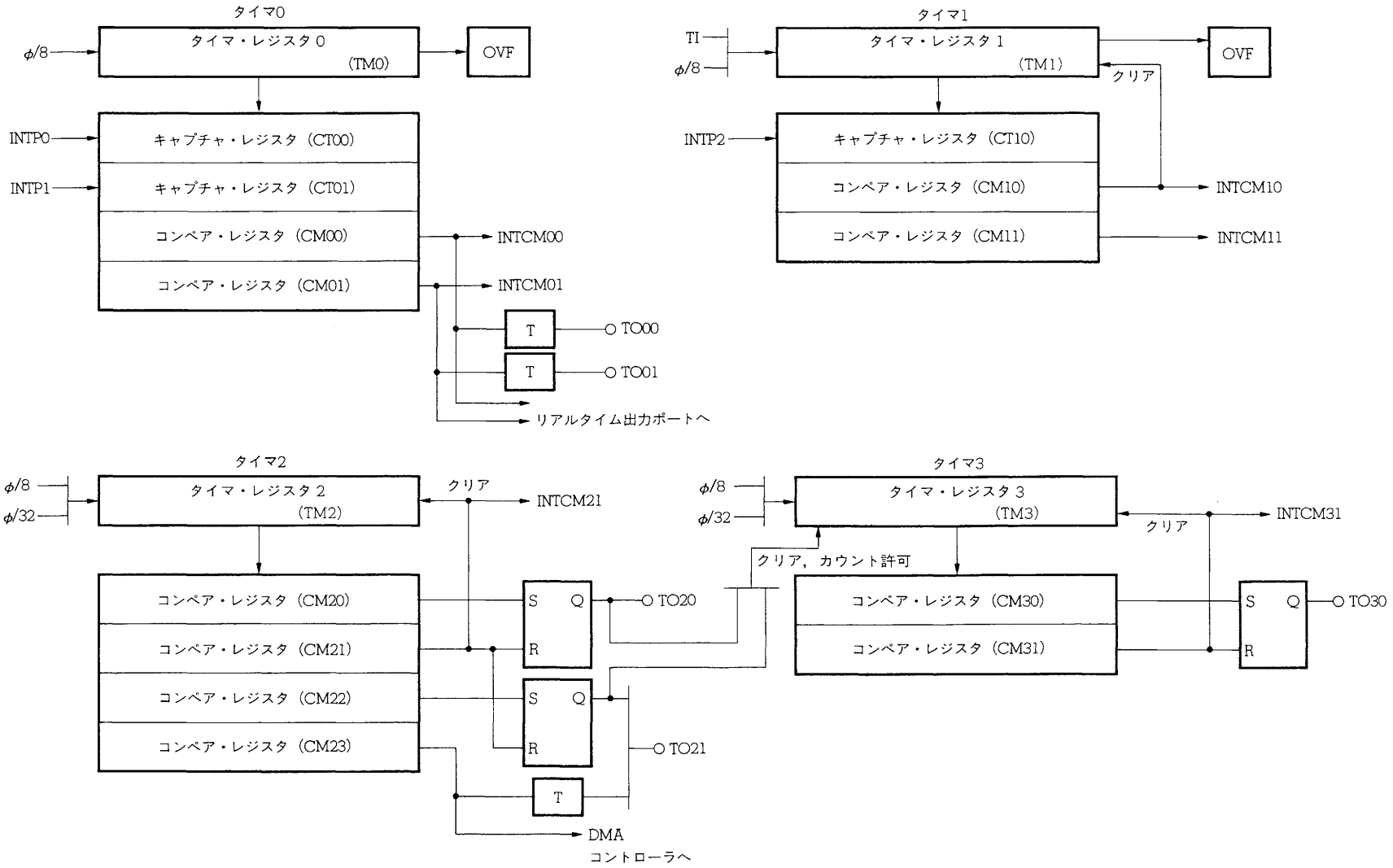
タイマ２は，16ビットのタイマ・レジスタ２（TM2），16ビットのコンペア・レジスタ（CM20，CM21，CM22，CM23）から構成され，16ビットのインターバル・タイマとして機能します。

(4) タイマ３

タイマ３は，16ビットのタイマ・レジスタ３（TM3），16ビットのコンペア・レジスタ（CM30，CM31）から構成され，16ビットのインターバル・タイマとして機能します。

保守／廃止

図９－１　タイマ・ユニットのブロック図



保守／廃止

9.3 動　　作

各タイマの機能を表９－２に示します。

表９－２　各タイマの機能

			タイマ０	タイマ１	タイマ２	タイマ３
機能	カウント機能		○	○	○	○
	キャプチャ機能		○	○	—	—
	コンペア機能		○	○	○	○
	タイマ出力機能	トグル出力	○	—	○	—
		セット/リセット出力	—	—	○	○
	カスケード接続		—	—	○	

9.3.1 カウント機能

カウント動作には，次の２種類の動作方法があります。

●フリー・ランニング・タイマ（図９－２参照）

16ビットのタイマ・レジスタ（TM）でソース・クロックをカウントします。

カウントの開始は，TMCレジスタのCEビットをセットすることにより行います。

CEビットがセットされている状態でさらにCEビットをセット(1)した場合，TMレジスタはクリアされないでカウント動作を続けます。

FFFFHまでカウントするとTMレジスタからのボローが発生し，TMCレジスタのOVFフラグをセットします。ボロー発生時，TMレジスタはクリアされカウント動作を継続します。

●インターバル・タイマ（図９－３参照）

あらかじめ設定したカウント時間をインターバルとして，割り込み要求を繰り返し発生します。

16ビットのタイマ・レジスタ（TM）でソース・クロックをカウントします。

カウントの開始は，TMCレジスタのCEビットをセットすることにより行います。

TMレジスタがカウント・アップした値とCMレジスタの値が一致すると，TMレジスタをクリアしたあとカウント動作を継続します。

カウント・クロックとシステム・クロック周波数ごとのカウント可能時間を表９－３に示します。

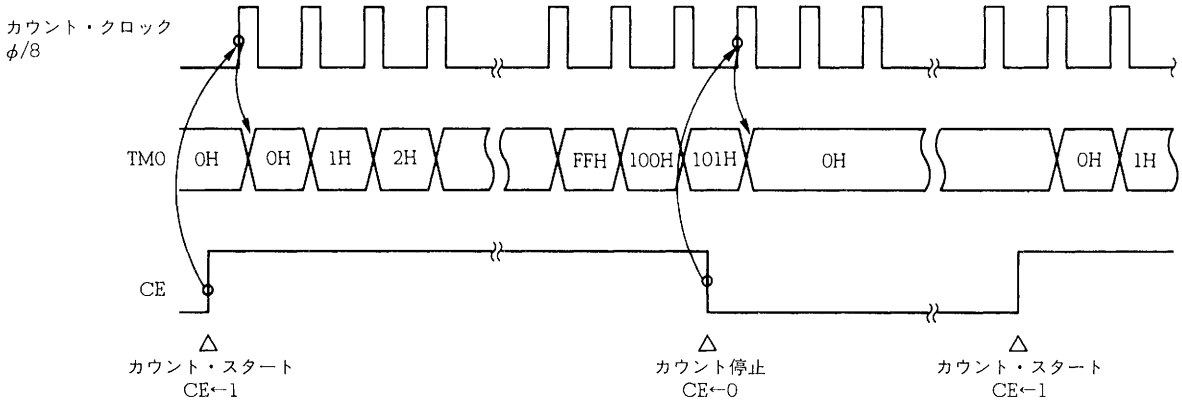
保守／廃止

表９－３　タイマのカウント可能時間

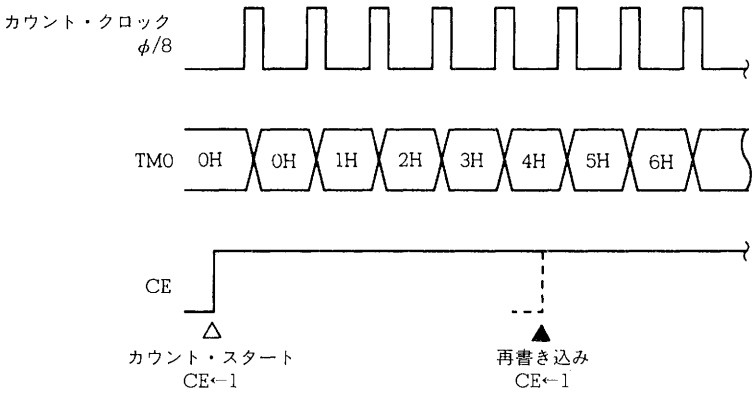
カウント・クロック	システム・クロック（ϕ）				備　　考
	8 MHz	10 MHz	12.5 MHz	16 MHz	
ϕ/8	1 μs～65.4 ms	0.8 μs～52.4 ms	0.6 μs～42.0 ms	0.5 μs～32.7 ms	カウント時間＝［カウント・クロック／（コンペア・レジスタ値＋1）］$^{-1}$
ϕ/32	4 μs～262 ms	3.2 μs～209.6 ms	2.6 μs～167.8 ms	2 μs～131 ms	

図９－２　フリー・ランニング・タイマ動作（タイマ０の場合）（1/2）

（a）カウント・スタート→カウント停止→カウント・スタートの場合



（b）カウント・スタート後に再度CEビット“1”を書き込んだ場合



保守／廃止

図９－２　フリー・ランニング・タイマ動作（タイマ０の場合）（2/2）

（c）TMOがFFFFHのときの動作

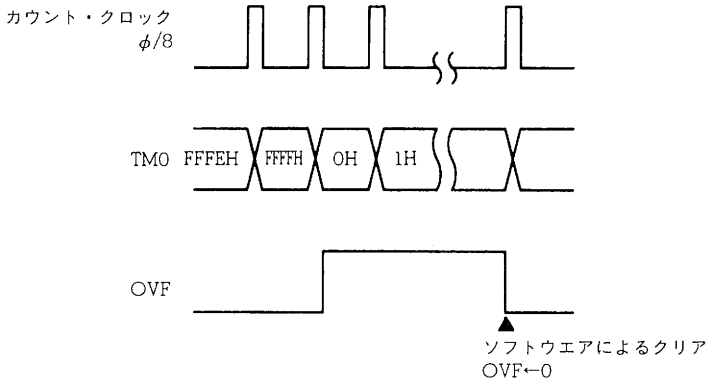
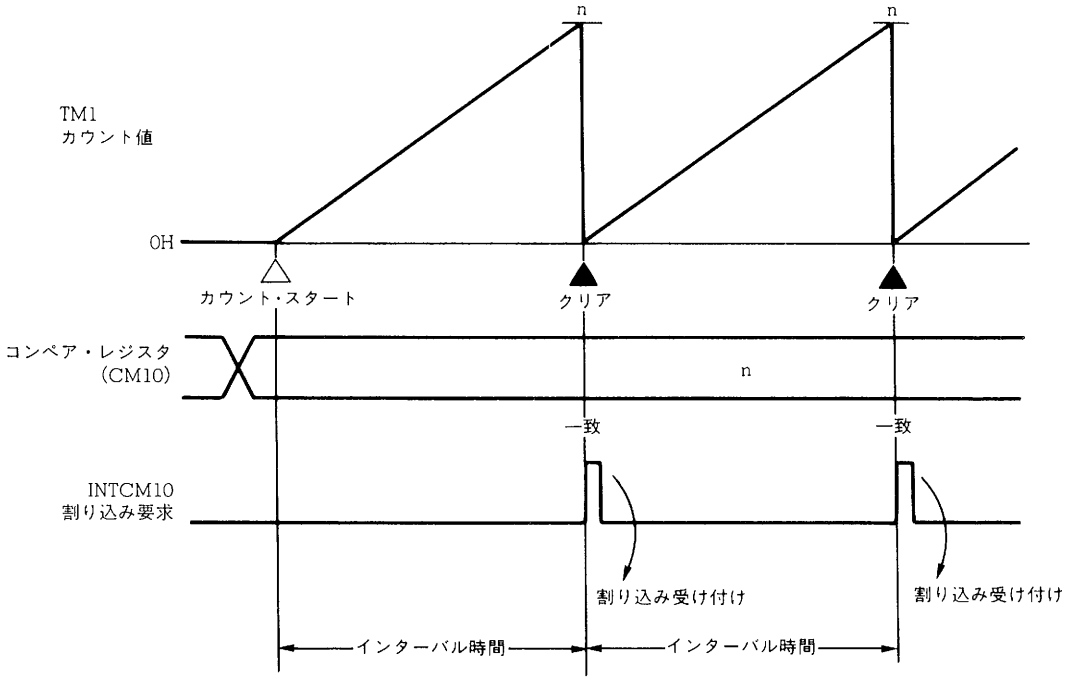


図９－３　インターバル・タイマ動作（CM10をコンペア・レジスタとして使用する場合）



備考　インターバル時間＝(n＋1)×x/φ

0≦n≦FFFFH

x＝8，32

保守／廃止

(1) タイマ０動作

フリー・ランニング・タイマとしてのみ動作します。

ソース・クロックは内部クロック（ϕ/8）を使用します。

使用するレジスタは次のとおりです。

- タイマ・レジスタ：TM0

(2) タイマ1動作

フリー・ランニング・タイマとインターバル・タイマの両方の動作を実行できます。

- フリー・ランニング・タイマ：TMC0レジスタのCLRビット＝1
- インターバル・タイマ：TMC0レジスタのCLRビット＝0

ソース･クロックは，内部クロック（ϕ/8）と外部クロック（TI端子入力）のどちらかを選択することができます。

使用するレジスタは次のとおりです。

- タイマ・レジスタ：TM1
- コンペア・レジスタ：CM10

(3) タイマ2動作

インターバル・タイマとしてのみ動作します。

ソース・クロックは，内部クロックのうちϕ/8とϕ/32が選択できます。

使用するレジスタは次のとおりです。

- タイマ・レジスタ：TM2
- コンペア・レジスタ：CM21

(4) タイマ3動作

タイマ３はタイマ２とのカスケード接続が可能です。

インターバル・タイマとしてのみ動作します。

ソース・クロックは，内部クロックのうちϕ/8とϕ/32が選択できます。

使用するレジスタは次のとおりです。

- タイマ・レジスタ：TM3
- コンペア・レジスタ：CM31

なお，タイマ２とのカスケード･モードに接続されている場合（TMC1レジスタのCCLRビット＝1），タイマ２のセット／リセット出力でカウント開始または停止の制御ができます(**9.3.5　カスケード接続**参照)。

保守／廃止

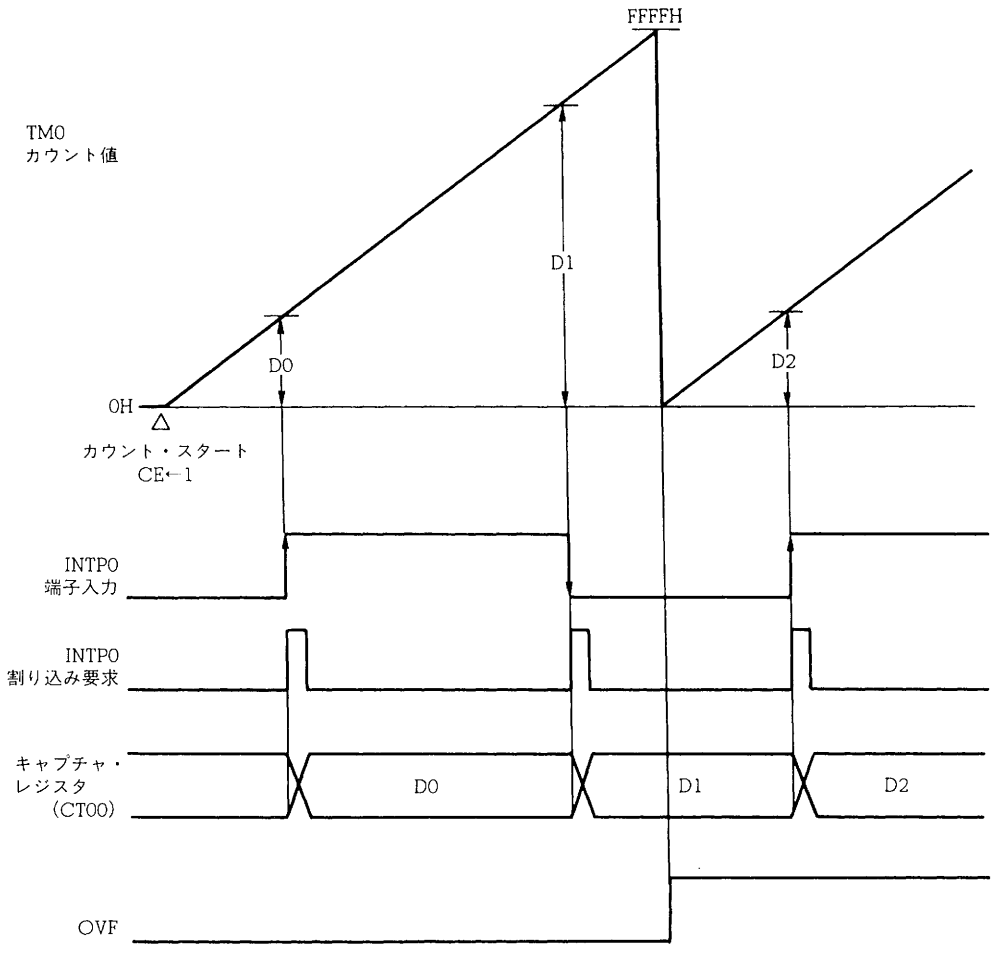
9.3.2　キャプチャ機能

この機能を用いることにより，入力信号のパルス幅と入力パルスの周期を求めることができます。

外部からのキャプチャ・トリガ信号（外部割り込み兼用端子INTP）が発生すると，そのときのTMレジスタの値をキャプチャ・レジスタ（CT）にラッチ（キャプチャ動作）します。ラッチした値は，次のキャプチャ動作が実行されるまでキャプチャ・レジスタ（CT）に保持されます。

キャプチャ・トリガの有効エッジは，外部割り込みモード・レジスタ１（INTM1）により設定します（**5.6　外部割り込み**参照）。立ち上がり，立ち下がり両エッジがキャプチャ・トリガとなるように設定すれば，外部からの入力パルス幅を測定することができます。また，片側エッジでキャプチャ・トリガを発生させた場合，入力パルスの周期を測定することができます。

図９－４　キャプチャ動作（タイマ０の場合）



備考　Dn：TM0のカウント値（n＝0，1，2，…）

保守／廃止

(1) タイマ０動作

２つのキャプチャ・レジスタを備えています。

キャプチャ・レジスタ	トリガ信号	タイマ・レジスタ
CT00	INTP0	TM0
CT01	INTP1	

(2) タイマ１動作

１つのキャプチャ・レジスタを備えています。

キャプチャ・レジスタ	トリガ信号	タイマ・レジスタ
CT10	INTP2	TM1

9.3.3　コンペア機能

あらかじめ設定されたコンペア・レジスタ（CM）の値とタイマ・レジスタ（TM）のカウント値を比較します。比較した結果CMレジスタとTMレジスタの値が一致した場合の動作は次の３種類です。

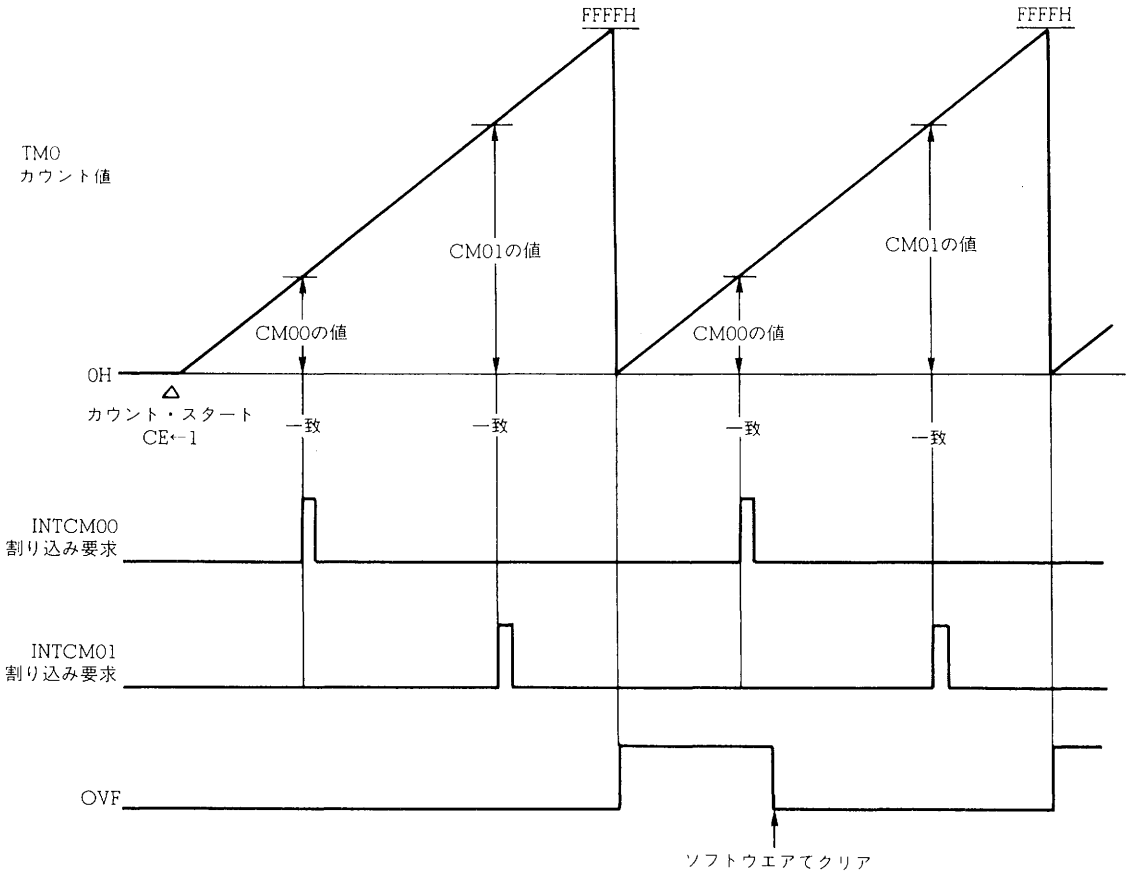
- 割り込み要求（INTCM）が発生
- タイマ出力要求が発生
- TMレジスタをクリアしたあとカウント動作を継続

なおTMレジスタのクリア動作は，次のコンペア・レジスタを使用するときにかぎり行われます。

コンペア・レジスタ	クリアするTMレジスタ
CM10	TM1
CM21	TM2
CM31	TM3

保守／廃止

図９－５　コンペア動作（タイマ０の場合）



(1) タイマ０動作

割り込み要求の機能があります。

コンペア・レジスタ	割り込み要求	タイマ・レジスタ
CM00	INTCM00	TM0
CM01	INTCM01	

(2) タイマ１動作

割り込み要求の機能があります。

コンペア・レジスタ	割り込み要求	タイマ・レジスタ
CM10	INTCM10	TM1
CM11	INTCM11	

保守／廃止

(3) タイマ２動作

割り込み要求とタイマ出力要求の機能があります。

コンペア・レジスタ	割り込み要求	タイマ出力	タイマ・レジスタ
CM20	なし	TO20（セット）	TM2
CM21	INTCM21	TO21/TO21（リセット）	
CM22	なし	TO21（セット）	
CM23	なし	TO21（トグル）	

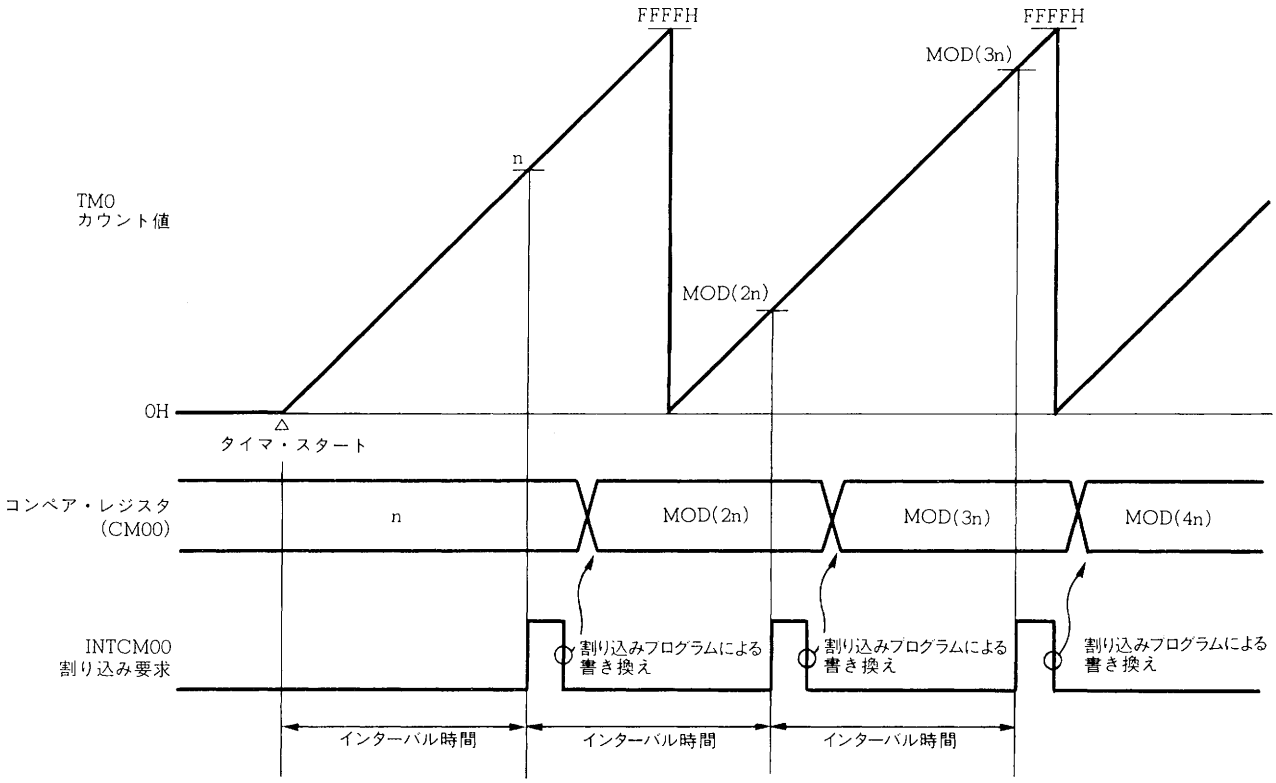
さらに，タイマ出力TO20，TO21はタイマ３とのカスケード接続が可能です。

(4) タイマ３動作

割り込み要求とタイマ出力要求の機能があります。

コンペア・レジスタ	割り込み要求	タイマ出力	タイマ・レジスタ
CM30	なし	TO30（セット）	TM3
CM31	INTCM31	TO30（リセット）	

図９-６　タイマ０をインターバル・タイマとして使用する場合



備考　インターバル時間＝n×8/ϕ，1≦n≦FFFFH

保守／廃止

9.3.4　タイマ出力機能

ポート2モード・コントロール・レジスタ（PMC2）で対応ビットをコントロール・モードに設定することにより，タイマ・ユニットの出力端子として使用できます。

タイマ出力機能には，トグル出力とセット／リセット出力があります。どちらの機能を使用するかは，タイマ出力制御レジスタ1（TOC1）のMODビットで指定します（**9.4.2　タイマ出力制御レジスタ（TOC0, TOC1）** 参照）。

●トグル出力

コンペア・レジスタ（CM）とタイマ・レジスタ（TM）の値が一致したときに発生するタイマ出力に同期して，タイマ出力端子（TO）のレベルを反転します。

●セット／リセット出力

コンペア・レジスタ（CM20，CM22，CM30）とタイマ・レジスタ（TM2，TM3）の値が一致したときに発生する信号で，TO端子（TO20，TO21，TO30）をセット(1)します。コンペア・レジスタ（CM21，CM31）とタイマ・レジスタ（TM2，TM3）の値が一致したときに発生する信号で，TO端子（TO20，TO21，TO30）をリセット(0)します。

(1) タイマ0動作

トグル出力の機能があります。

また，この一致検出は，リアルタイム出力ポートのトリガとして使用されます。

コンペア・レジスタ	タイマ出力端子	タイマ・レジスタ
CM00	TO00	TM0
CM01	TO01	

(2) タイマ2動作

トグル出力とセット／リセット出力の機能があります。

セット／リセット出力として使用する場合，TM2レジスタはCM21との一致検出でクリアされ，これによりTO20，TO21端子より2相のパルスが出力されます。

コンペア・レジスタ	タイマ出力端子	タイマ・レジスタ
CM23	TO21（トグル）	TM2
CM20	TO20（セット）	
CM21	TO20/TO21（リセット）	
CM22	TO21（セット）	

保守／廃止

(3) タイマ３動作

セット／リセット出力の機能があります。

なお，これはカスケード接続モードでない場合の動作です。カスケード接続モードの場合は，**9.3.5**を参照してください。

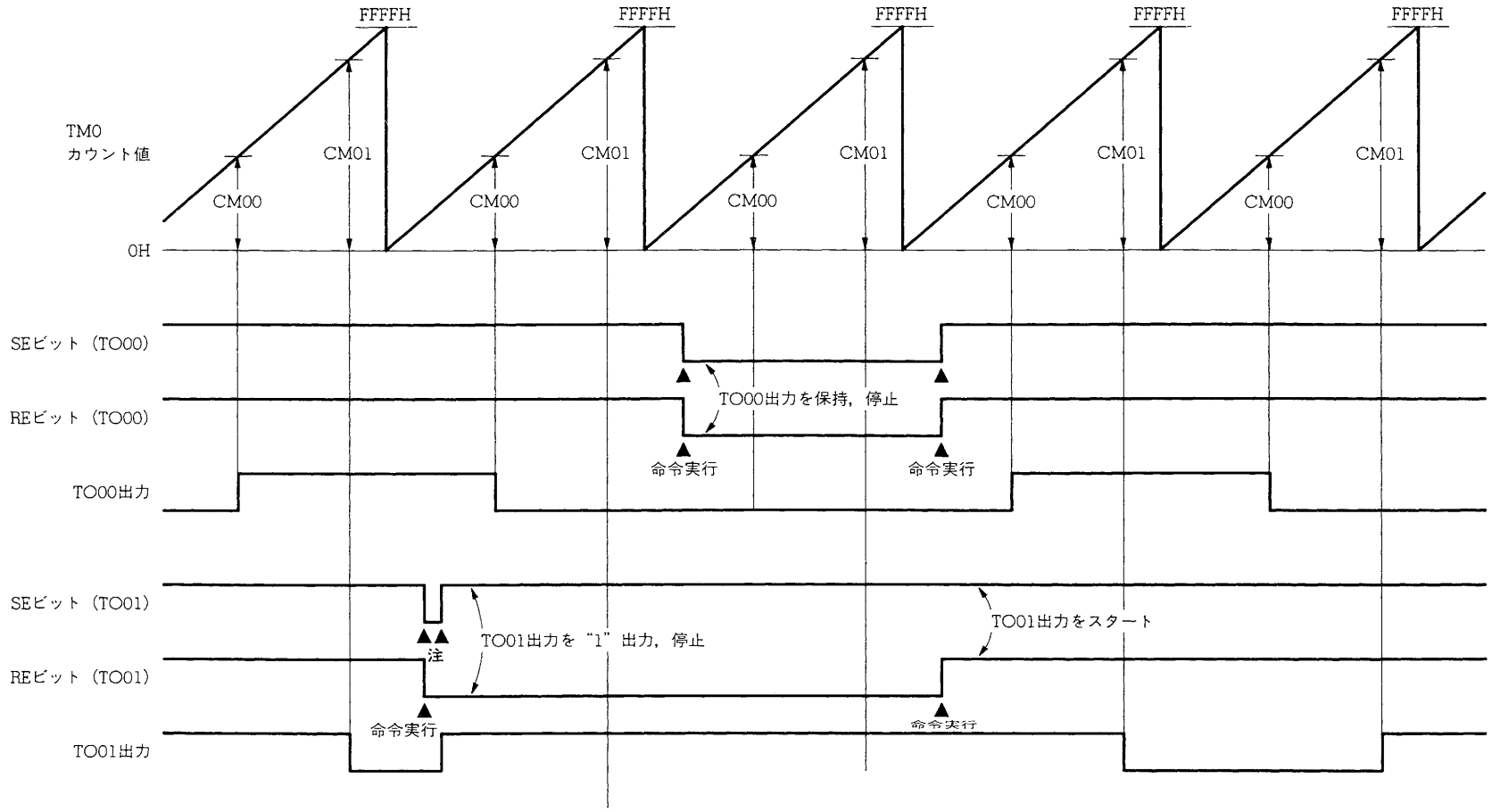
コンペア・レジスタ	タイマ出力端子	タイマ・レジスタ
CM30	TO30（セット）	TM3
CM31	TO30（リセット）	

トグル出力の動作を図９－7に，セット／リセット出力の動作を図９－８に示します。

なお，タイマ出力端子は，タイマ出力制御レジスタのSEビットとREビットの組み合わせにより制御されます。

保守／廃止

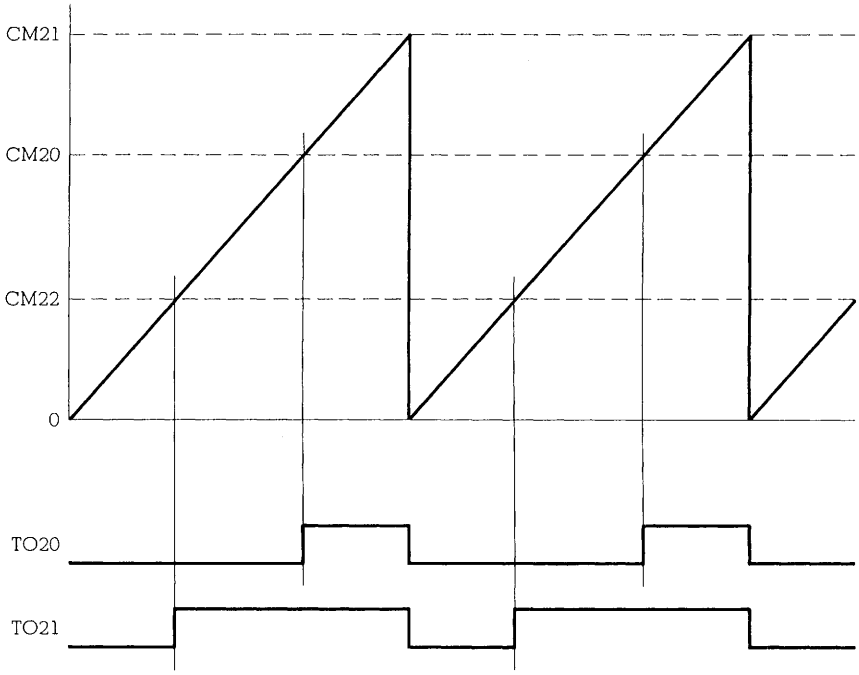
図9-7　タイマ0によるトグル出力



注　TO出力の信号レベルの変更と，タイマ出力の動作の変更，許可制御は同時には実行できません。したがって，タイマ動作の停止，TO出力レベルの変更を二段階に分けて行っています。

保守／廃止

図９－８　タイマ２によるセット／リセット出力



保守／廃止

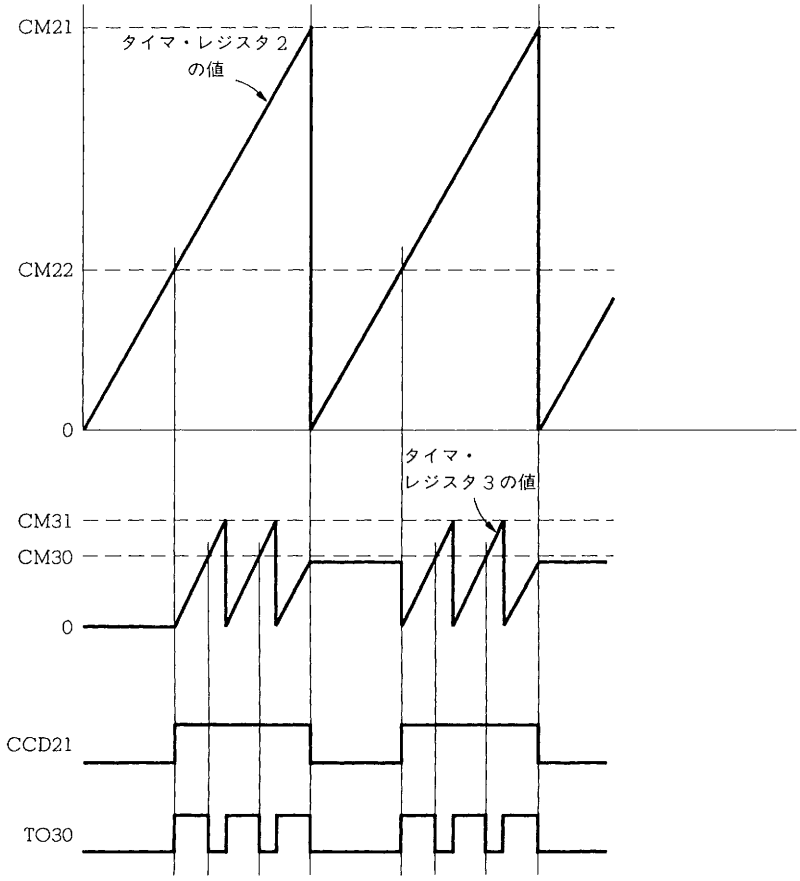
9.3.5　カスケード接続

タイマ２とタイマ３は，タイマ・コントロール・レジスタ（TMC1）のCCLRビットをセット(1)することによって，カスケードに接続することができます。

タイマ２のセット／リセット出力CCD20（TO20端子と対応）がロウ・レベルの期間中，タイマ３カウント動作は停止し，TO30端子からの出力はマスクされます。CCD20出力の立ち上がりでタイマ３はカウント動作を開始し，ハイ・レベルの期間中TO30端子からタイマ３のセット／リセット出力が行われます。CCD20の立ち上がりでタイマ３はクリアされます。同様にタイマ２からのセット／リセット出力CCD21（TO21端子と対応）を使用できます。

CCD21信号によるカスケード接続を行った場合のTO30の出力波形を図９-９に示します。

図９-９　カスケード接続によるTO30出力波形



注意1．カスケード接続モードでは，TO端子の出力レベルが通常と逆になります。CM30レジスタとの一致検出でリセット(0)され，CM31レジスタとの一致検出でセット(1)されます。

2．カスケード接続モードでは，タイマ２のCEビットを１→０にした場合，タイマ３もカウントを停止します。これは，CCD20またはCCD21の出力がロウ・レベルの期間中は，タイマ３がカウントを停止するためです。

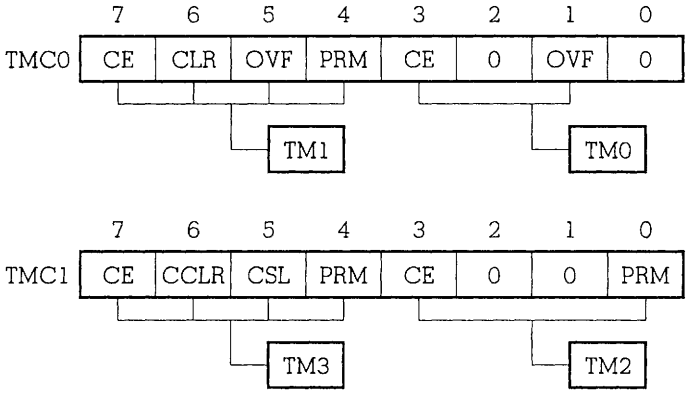
保守／廃止

９.４　制御レジスタ

９.４.１　タイマ・コントロール・レジスタ（TMC0，TMC1）

TMC0レジスタは，タイマ０，タイマ１の動作指定とカウント・クロックの選択を行う８ビットのレジスタです。同様にTMC1レジスタは，タイマ２，タイマ３の動作指定とカウント・クロックの選択を行います。TMC0レジスタとTMC1レジスタをワード・アクセスすることによって，タイマ０-３の動作開始／停止を同時に行うことが可能です。

リセット後のTMC0，TMC1レジスタの値は，00Hになります。以降，表中の網がけの部分はリセット後の状態を示します。



PRM　**TMnのカウント・クロックのソース選択（ϕ：システム・クロック）**

タイマ	PRM	カウント・クロック
TM0	0	ϕ/8（固定）
TM1	0	ϕ/8
	1	外部クロック（TI）
TM2	0	ϕ/8
	1	ϕ/32
TM3	0	ϕ/8
	1	ϕ/32

OVF　**タイマのオーバフロー・ステータス・フラグ**

タイマ	OVF	状　　態
TM0	0	TM0からキャリー信号が発生していない
	1	TM0からキャリー信号が発生している
TM1	0	TM1からキャリー信号が発生していない
	1	TM1からキャリー信号が発生している

保守／廃止

CLR　タイマ1（TM1）の動作モードの指定

CLR	モ　ー　ド	動　　作
0	インターバル・タイマ	CM10レジスタの一致信号でカウンタをクリア
1	フリー・ランニング・タイマ	CM10レジスタの一致信号にかかわらず，カウント動作を継続

CE　タイマnの動作制御フラグ[注]

CE	動　　作
0	TMnはクリアされ，停止状態
1	TMnはカウント動作を開始

注　CE＝0のとき，命令によるタイマ・カウンタへの書き込みを許可します。
CE＝1のときには値は書き込まれません。

CCLR　**CSL**　TO30端子出力動作の指定

CCLR	CSL	タイマ3のセット／リセット出力制御
0	0	CCD20，CCD21出力の制御を受けずセット／リセット出力
0	1	設定禁止
1	0	TO30端子のセット／リセット出力制御信号としてCCD20を選択
1	1	TO30端子のセット／リセット出力制御信号としてCCD21を選択

CCLR＝1のモードにおける動作

CCD20（またはCCD21）がロウ・レベルの期間中，タイマ3はカウント動作を停止します。またこの間，TO30端子はマスクされロウ・レベルを出力します。

CCD20（またはCCD21）の立ち上がりでタイマ3はカウントを開始します。CCD20（CCD21）がハイ・レベルの期間中，TO30からCM31，CM30の一致信号によるセット／リセット信号が出力されます。

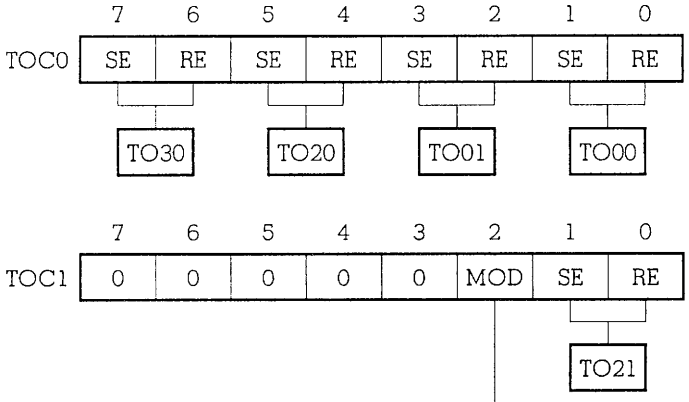
CCD20（またはCCD21）の立ち下がりでタイマ3はクリアされます（なおCCLR＝1のモードにおいても，タイマ3のCEビットのクリアによってタイマ3はクリアされます）。

保守／廃止

9.4.2　タイマ出力制御レジスタ（TOC0，TOC1）

TOC0，TOC1レジスタは，外部出力端子TO00，TO01，TO20，TO21，TO30の出力レベルを制御します。

リセット後，TOC0，TOC1レジスタの値は，00Hになります。またリセット後，ポート2モード・コントロール・レジスタ（PMC2）をコントロール・モードに設定することにより，各端子の出力レベルは1になります。



TO21出力信号の出力モード選択フラグ

MOD	モード	動　　作
0	トグル出力モード	CM23レジスタの一致信号でTO21出力レベルを反転
1	セット／リセット出力モード	CM22レジスタの一致信号でTO21の出力レベルをセット(1) CM21レジスタの一致信号でTO21の出力レベルをリセット(0)

TO00，TO01，TO20，TO21，TO30端子制御フラグ

SE	RE	出　力　状　態
0	0	保持，停止
0	1	0出力，停止
1	0	1出力，停止
1	1	スタート

注意　出力端子TO00，TO01，TO20，TO21，TO30の信号レベルの変更と，タイマの出力動作の変更，許可制御は，同時には実行できません。

備考　□ はリセット後の状態を示します。

保守／廃止

９.５　割り込み要求

９.５.１　タイマ・ユニットからの割り込み要求

タイマ・ユニットは各タイマとコンペア・レジスタとの一致検出時に割り込み要求を発生し，割り込み制御ユニットに伝えます。

タイマ・ユニットから発生する割り込みソースは，次の６種類です。

- INTCM00（CM00一致検出割り込み）
- INTCM01（CM01一致検出割り込み）
- INTCM10（CM10一致検出割り込み）
- INTCM11（CM11一致検出割り込み）
- INTCM21（CM21一致検出割り込み）
- INTCM31（CM31一致検出割り込み）

９.５.２　キャプチャ・トリガ兼用外部割り込み

次の外部割り込み信号は，タイマ０，タイマ１のキャプチャ・トリガとなります。

外部割り込み制御および設定制御レジスタについては，**５.６　外部割り込み**を参照してください。

- INTP0（キャプチャ・レジスタCT00のキャプチャ・トリガ）
- INTP1（キャプチャ・レジスタCT01のキャプチャ・トリガ）
- INTP2（キャプチャ・レジスタCT10のキャプチャ・トリガ）

保守／廃止

9.6　注意事項

○TI入力イベントのカウント・タイミング

イベント・カウント入力端子（TI）のサンプリングはϕ/8ごとに行われます。

カウント精度は次のようになります。

システム・クロックϕの分周比	サンプリング時間	
	f_{XX}＝25 MHz時	f_{XX}＝32 MHz時
$\phi=f_{XX}/2$	0.64 μs	0.50 μs
$\phi=f_{XX}/4$	1.28 μs	1.00 μs
$\phi=f_{XX}/8$	2.56 μs	2.00 μs
$\phi=f_{XX}/16$	5.12 μs	4.00 μs

f_{XX}：クリスタル発振周波数

① 外部イベント入力端子TIにシリアル・クロックを入力し，通信速度を計算する応用回路の場合，通信速度を64 kbpsとすれば最低でも15.6 μsの精度が必要となります。

② サンプリングの精度を考えると，入力クロックの最大周波数は，$\phi=f_{XX}/2$モードで156 kHz程度までとなります。

ただし，回路上，TI＜ϕ/8未満であれば，誤計算は起こしません（外部イベント入力に対し，高速かつリアルタイムに応答が必要な応用回路にはあまり適していません）。

〔メ　モ〕

保守／廃止

第10章　リアルタイム出力ポート機能

V55PIのポート７にはリアルタイム出力ポート機能が内蔵されており，タイマ０からのプログラマブルなインターバルでポート７バッファ（P7H，P7L）の内容を１ビット単位で出力できます。

10.1　特　　徴

○８ビット×1chまたは４ビット×2ch
○出力制御タイマ，出力バッファにマクロ・サービスを使用可能
○出力ディレイ機能内蔵

10.2　構　　成

リアルタイム出力ポートの構成を図10－１に示します。次のようなバッファ・レジスタ，出力ポート，制御レジスタで構成されます。

(1) ポート７バッファ（P7H，P7L）

ポート７バッファはポート７をリアルタイム出力ポート・モードに設定したときに，次に出力するデータを保持しておくレジスタです。表10－３に示すタイミングで内容をリアルタイム出力ポート（RTP）へ転送し，端子へ出力します。

ポート７バッファの内容はリセット入力により影響を受けません。

(2) リアルタイム出力ポート（RTP）

リアルタイム出力ポートの出力データはポート７バッファからこの出力ポートに保持され，端子より出力されます。

8/1ビットの命令操作によりリード／ライトすることができます(ポート７の出力ポートとは異なります)。

(3) リアルタイム出力ポート・ディレイ指定レジスタ（RTPD），ディレイ・カウンタ

リアルタイム出力ポート(RTP)から出力端子への出力タイミングに遅延時間を入れるモードを使用する際，設定および使用するレジスタです。

P7Lのビットが“0”に設定されている場合，転送トリガの発生タイミングからリアルタイム出力ポート・ディレイ指定レジスタに設定されているカウント・クロック時間分の遅延時間が経過したのち，出力端子の対応ビットに“0”が出力されます。このときの遅延時間は，ディレイ・カウンタでカウントされます。

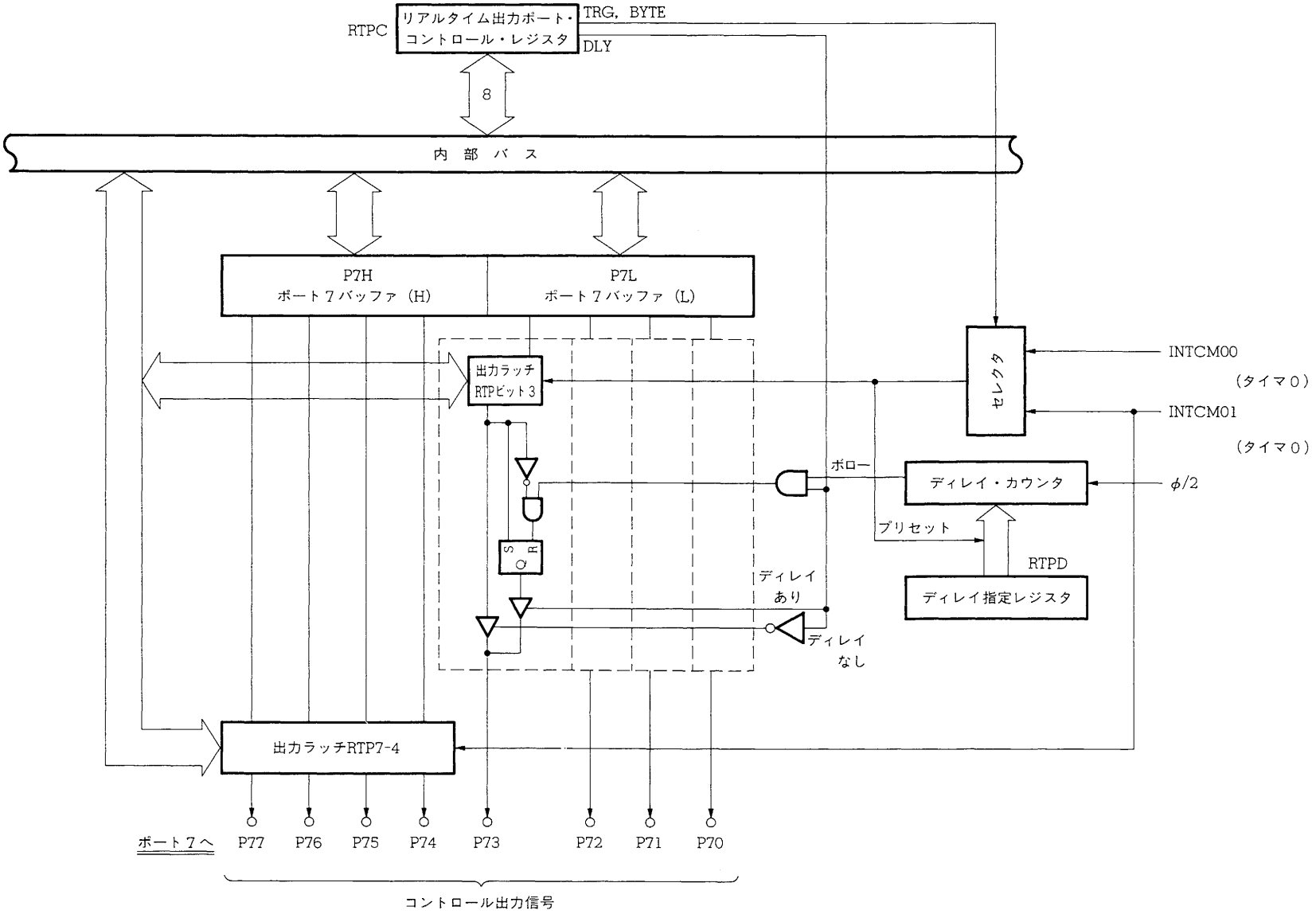
保守／廃止

(4) リアルタイム出力ポート・コントロール・レジスタ（RTPC）

リアルタイム出力ポートの動作モードを指定します。データ出力時のディレイ挿入の有無，ポート7バッファへのデータ転送タイミング，また，転送タイミングのトリガなどを指定することができます。

保守／廃止

図10－1　リアルタイム出力ポートのブロック図



保守／廃止

10.3　動　　作

ポート７モード・コントロール・レジスタ（PMC7）で対応ビットをコントロール・モードに設定することにより，リアルタイム出力ポート（RTP）の出力端子として使用できます。ポート７（P7），ポート７バッファ（P7H，P7L），リアルタイム出力ポートがアクセスできます。

10.3.1　基本動作

データの出力は次の順で行われます。

① ポート７バッファ（P7H，P7L）にデータを書き込みます。

② タイマ割り込み要求（INTCM00，INTCM01），またはリアルタイム出力ポート・コントロール・レジスタ（RTPC）のTRGビットへの書き込みタイミングで，ポート７バッファの内容がRTPに転送されます。

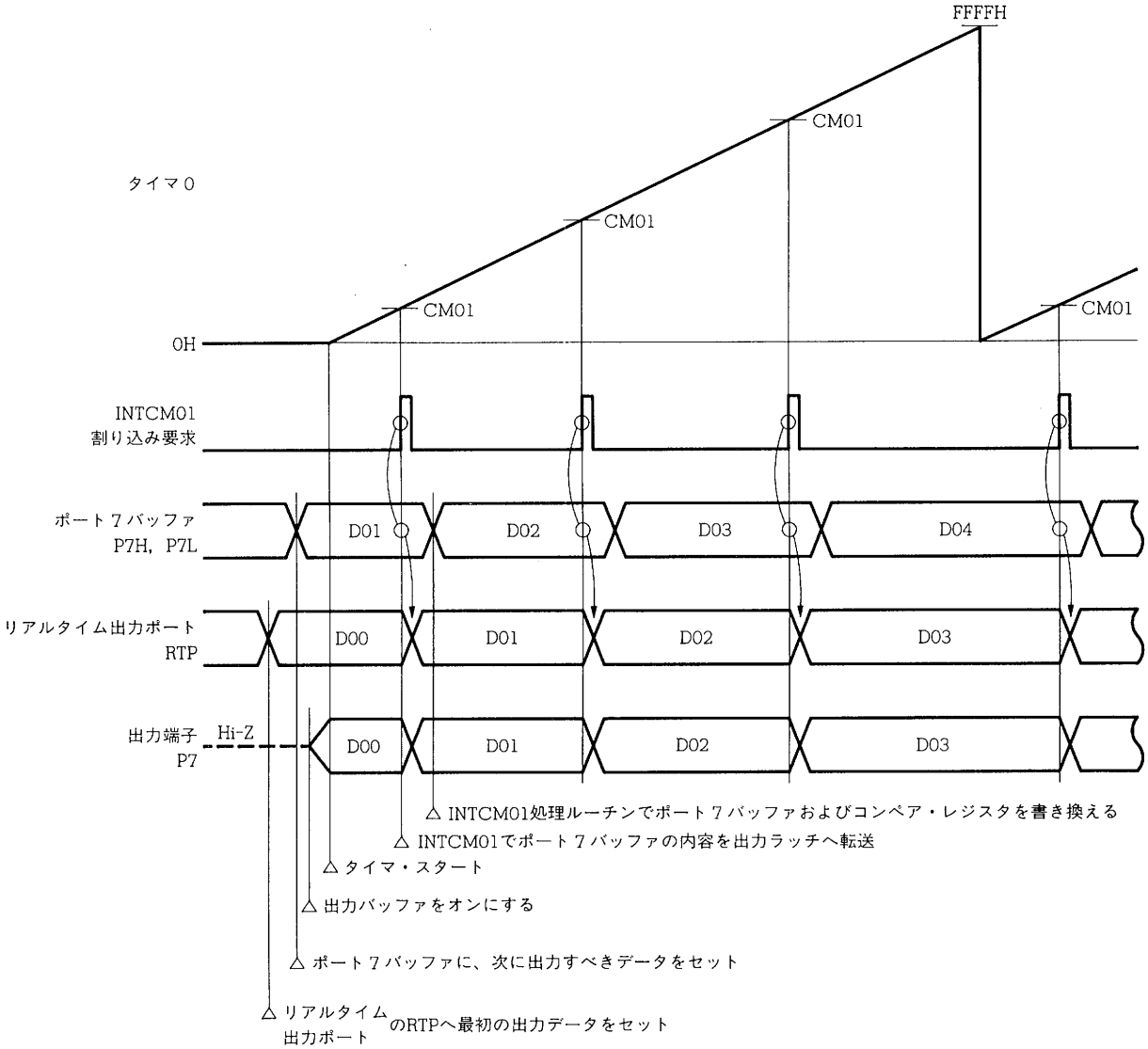
③ RTPの内容が端子へ出力されます。

②のデータ転送タイミング（転送トリガ）は，RTPCレジスタのBYTEビットとTRGビットで指定します。タイマ割り込み要求を転送トリガとして選択した場合は，割り込み要求のマスク，保留にかかわらず，割り込み要求が発生するとバッファ・レジスタから出力ラッチへの転送が行われます。

図10－２に，タイマ０の割り込み要求（INTCM01）を転送トリガとした場合（BYTEビット＝1，TRGビット＝0）のリアルタイム出力ポートの動作タイミングを示します。

保守／廃止

図10－2　リアルタイム出力ポートの動作タイミング



保守／廃止

10.3.2　ディレイ機能

リアルタイム出力ポート・ディレイ指定レジスタ（RTPD）の設定により，端子への出力タイミングにディレイ（遅延）時間を入れることができます。ただし，使用できるのはP7Lバッファだけです。

ポート出力が1から0に変化するときにかぎり，出力データの変化タイミングを遅延させることができます。ディレイ時間は，ディレイ・カウンタによって決まります。

(1) 設　定

RTPCレジスタのDLYビット＝1にすることにより，RTPDレジスタとディレイ・カウンタが有効になります。ディレイ時間は，次のように計算されます（RTPD＝0のときはRTPD設定値に256を入れてください）。

$2/\phi \times$（RTPD設定値）

表10－1に，RTPD設定値とディレイ時間の関係を示します。

表10－1　RTPD設定値とディレイ時間

RTPD	ディレイ時間（μs）	
	ϕ＝12.5 MHz時	ϕ＝16 MHz時
0	40.96	32.0
1	0.16	0.125
2	0.32	0.25
⋮	⋮	⋮
255	40.8	31.875

（ϕ：システム・クロック）

(2) 転送トリガ

ディレイ指定の転送トリガは，RTPCレジスタのBYTEビットの設定により次のようになります。

(a) 4ビット単位指定のとき（BYTE＝0）

- 割り込み要求の発生（INTCM00）
- RTPCレジスタのTRGビットのセット

(b) 8ビット単位指定のとき（BYTE＝1）

- 割り込み要求の発生（INTCM01）
- RTPCレジスタのTRGビットのセット

保守／廃止

(3) 動　　作

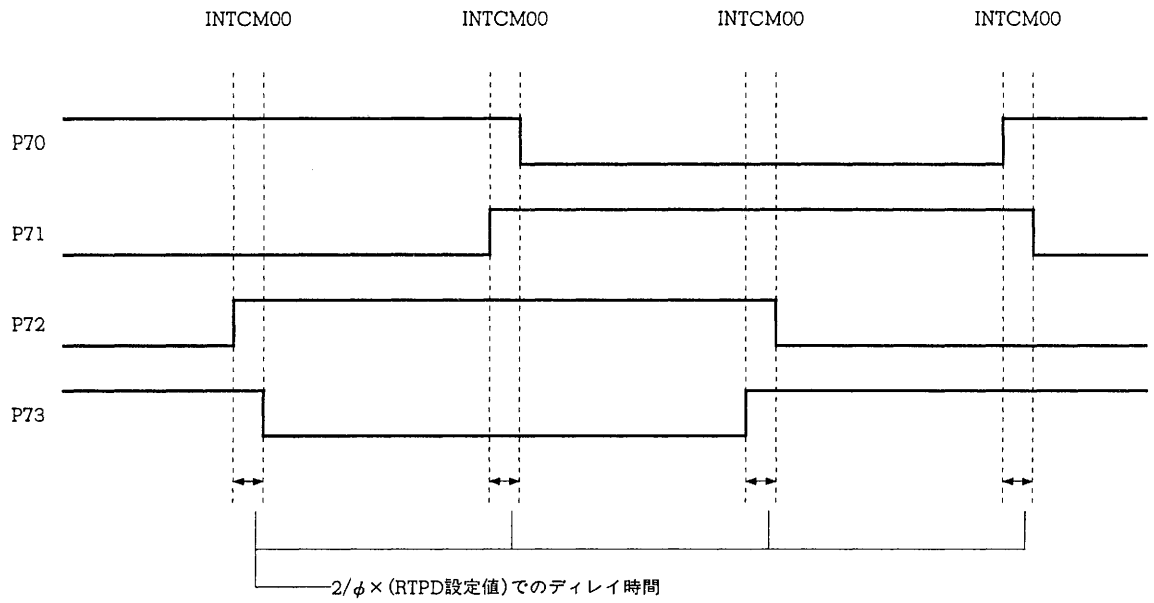
① 転送トリガ（INTCM要求またはTRGビットのセット）の発生により，RTPDレジスタの値をディレイ・カウンタへセットします。

② ダウン・カウントを開始します（2/φのタイミングでデクリメント）。

③ P7Lバッファに保持されているデータにより，次のタイミングで端子へ出力します。

- データが１のとき：そのまま端子へ出力
- データが０のとき：ディレイ・カウンタのボローが発生したタイミングで端子へ出力

図10－3　リアルタイム出力ポートの出力動作（ディレイ・モード）



10.3.3　マクロ・サービスとの組み合わせ

(1) 設　　定

割り込み要求制御レジスタのMS/$\overline{\text{INT}}$ビットをセット(1)することにより，マクロ・サービスを選択します。

マクロ・サービス・モード・レジスタによって，動作モードをRTOPTRNモードに設定します。

(2) 動　　作

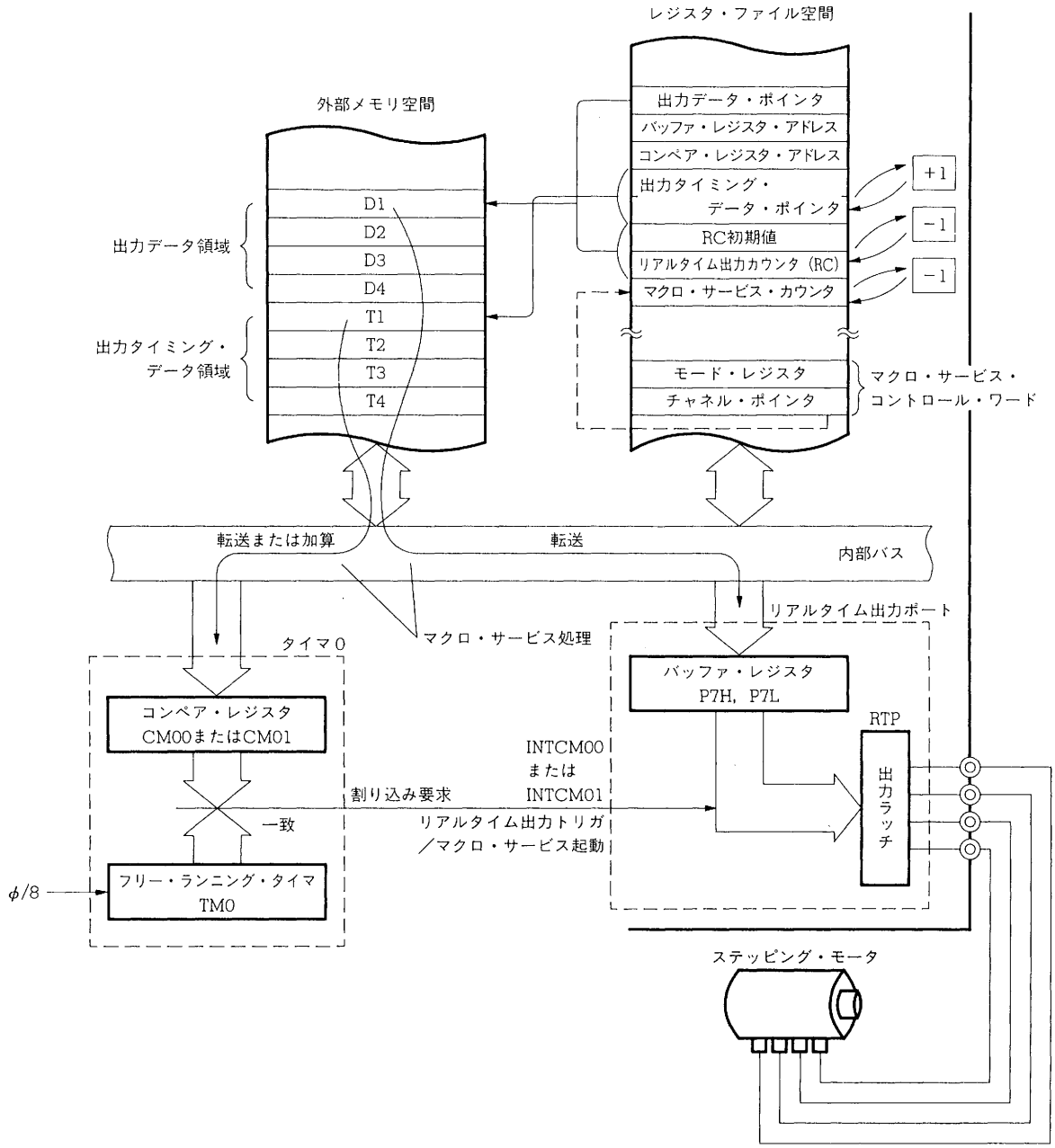
外部メモリ空間にあらかじめ設定してある２つのデータ格納領域より，リアルタイム出力ポート機能のバッファ・レジスタ（P7H，P7L）およびコンペア・レジスタ（CM00，CM01）に更新データを転送します（**5.4.3(6)(f)RTOPTRN(REAL TIME OUTPUT PORT TRANSFER)**参照）。

保守／廃止

10.3.4 応 用 例

リアルタイム出力ポートへの出力パターンと出力インターバルを直接制御する例を図10－4に示します。

図10－4　リアルタイム出力ポートによるステッピング・モータ制御



保守／廃止

10.3.5　ポート・アクセス時の動作

ポート７をPMC7レジスタ，RTPCレジスタにより各動作モードに設定してリード／ライト動作を行った場合にアクセス対象となるSFRを表10－２に示します。

８ビットのリアルタイム出力ポートとして指定している場合(BYTEビット＝1)，P7H，P7Lどちらのアドレスを指定して書き込みを行っても，８ビット・データをP7バッファに一度に書き込みます。

また，どちらのアドレスを指定して読み込みを行っても，P7H，P7Lを同時に読み出すことができます。

表10－２　ポート７のアクセス対象

BYTE (RTPCレジスタ)	PMC7n	動作モード	アクセスSFR	リード・アクセス対象		ライト・アクセス対象	
				上位４ビット	下位４ビット	上位４ビット	下位４ビット
1/0	0	ポート・モード	P7	注		ポート７出力ラッチ	
			P7H	P7H	P7L	P7H	—
			P7L	P7H	P7L	—	P7L
			RTP	RTP		RTP	
0	1	４ビット・セパレート・リアルタイム出力ポート	P7	RTP		P7	
			P7H	P7H	P7L	P7H	—
			P7L	P7H	P7L	—	P7L
			RTP	RTP		RTP	
1	1	８ビット・リアルタイム出力ポート	P7	RTP		P7	
			P7H	P7H	P7L	P7H	P7L
			P7L	P7H	P7L	P7H	P7L
			RTP	RTP		RTP	

注　ポート・リード・コントロール・レジスタ（PRDC）の設定により，ポート７モード・レジスタ（PM7）のビットごとにリード・アクセス対象が次のように異なります。

- PRDC＝0のとき

  PM7n＝0（出力モード）のビット…ポート７出力ラッチの値

  PM7n＝1（入力モード）のビット…ポート７の該当端子のレベル

- PRDC＝1のとき

  PM7n＝0（出力モード）のビット…ポート７の該当端子のレベル

  PM7n＝1（入力モード）のビット…“0”

備考　—：ライト・アクセスは行われず，RTP，P7H，P7Lは影響を受けません。

保守／廃止

10.4　制御レジスタ

10.4.1　ポート7モード・コントロール・レジスタ（PMC7）

ポート7をビット単位でリアルタイム出力ポートに設定する8ビットのレジスタです。8/1ビットのメモリ・アクセスでリード／ライトが可能です。リアルタイム出力ポートとして使用する場合，該当ビットを“1”に設定します。リアルタイム出力ポートとして使用するビット以外は汎用ポートとして使用することができます。

リセット後のPMC7レジスタの値は00Hとなり，すべてのビットがポート・モード指定となります。

	7	6	5	4	3	2	1	0
PMC7	PMC77	PMC76	PMC75	PMC74	PMC73	PMC72	PMC71	PMC70

P7の各ビットのモード指定

PMC7n	モード指定
0	入出力ポート
1	リアルタイム出力ポート

備考　　はリセット後の状態を示します。

保守／廃止

10.4.2　リアルタイム出力ポート・コントロール・レジスタ（RTPC）

リアルタイム出力ポートの動作モードを指定する８ビットのレジスタです。8/1ビットのメモリ・アクセスでリード／ライトが可能です。

リセット後のRTPCレジスタの値は，40Hになります。

	7	6	5	4	3	2	1	0
RTPC	TRG	BYTE	DLY	0	0	0	0	0

データ出力時のディレイの指定

DLY	モード指定
0	ディレイなしモード
1	ディレイありモード

転送タイミングの指定

BYTE		転送タイミング
0	P7L	INTCM00のタイミング
	P7H	INTCM01のタイミング
1	P7L	INTCM01のタイミング
	P7H	

転送タイミングのトリガ指定

TRG	トリガ
0	タイマ０の割り込み要求
1	このビットへの書き込み

備考　　はリセット後の状態を示します。

TRG　転送タイミングのトリガ指定

ポート７バッファ（P7H，P7L）の値をリアルタイム出力ポート（RTP）へ転送するトリガ・ソースを指定します。

TRG	リアルタイム出力ポートの転送タイミング指定
0	タイマ０から発生する割り込み要求[注]がアクティブとなるタイミングで，レジスタの内容を転送
1	TRGビットへの書き込み動作タイミングで，レジスタの内容を転送（転送後１を保持）

注　P7L：INTCM00（BYTEビット＝0のとき），
INTCM01（BYTEビット＝1のとき）
P7H：INTCM01

保守／廃止

BYTE 転送タイミングの指定

ポート7バッファ（P7H，P7L）から出力ポート（RTP）へのデータ転送タイミングを設定します（表10－3参照）。

TRGビット＝0のときのみ有効となります。

表10－3　リアルタイム出力ポート（RTP）へのデータ転送タイミング

BYTE	TRG	P7H	P7L
0	0	INTCM01がアクティブとなるタイミング	INTCM00がアクティブとなるタイミング
0	1	TRGビットへの“1”の書き込みタイミング（データ転送後も1を保持）	
1	0	INTCM01がアクティブとなるタイミング	
1	1	TRGビットへの“1”の書き込みタイミング（データ転送後も1を保持）	

DLY データ出力時のディレイの指定

転送トリガが発生したとき，P7Lの“0”データを出力端子に出力するまでに遅延時間を挿入するモードを指定します。

転送データが“1”のときには出力タイミングに遅延は入りません。また，ポート7バッファの下位側P7Lについてのみ有効です。上位側は常にP7Hのデータ転送と同時に端子出力が変化します。

DLY	モード	端子からのデータ出力タイミング	
0	ディレイなし	出力ポート（RTP）への転送トリガ発生タイミングに同期してP7Lの各ビット・データを端子に出力	
1	ディレイあり	出力ビットデータ＝0	出力ポート（RTP）への転送トリガ発生タイミングから，RTPDレジスタで設定のディレイを挿入後に端子出力
		出力ビットデータ＝1	出力ポート（RTP）への転送トリガ発生タイミングに同期して端子へ出力

保守／廃止

10.4.3　リアルタイム出力ポート・ディレイ指定レジスタ（RTPD）

RTPDレジスタは，転送トリガが発生したときにポート7バッファ（P7L）の“0”データを端子に出力するまでの遅延時間を指定する8ビットのレジスタです。8/1ビットのメモリ・アクセスでリード／ライトが可能です。

リセット入力によってRTPDレジスタの値は，不定となります。

このレジスタの設定は，リアルタイム出力ポート・コントロール・レジスタ（RTPC）のDLYビットが1のときのみ有効です。

10.4.4　ディレイ・カウンタ

転送トリガが発生してからポート7バッファ（P7L）の“0”データを端子に出力するまでのディレイ時間をカウントする8ビットのカウンタです。

RTPCレジスタのDLYビット（ビット5）＝1のときに有効です。

このカウンタは命令によるリード／ライトはできません。

注意　ディレイ・カウンタ動作中にディレイありモードからディレイなしモードに切り替えると，次にP7Lバッファの値をリアルタイム出力ポートへ転送するタイミングが発生するまで，P70-P73端子の出力状態は変化しません。
RTP0-RTP3の値をP70-P73へ出力するには，RTPCレジスタのTRGビットの操作が必要です。

〔メ　モ〕

保守／廃止

第11章　PWM出力機能

V55PIは，8ビット精度のPWM（パルス幅変調）信号出力機能を備えています。

PWM出力は，外部にロウ・パス・フィルタなどを接続することによって，ディジタル－アナログ変換出力として使用することができます。モータなどのアクチュエータ制御信号に最適です。

11.1　特　　徴

○PWM出力パルスのアクティブ・レベルを選択可能

○周波数：25 MHz（システム・クロックϕ＝12.5 MHz時）

→ PWM周期：40.96 μs

：32 MHz（システム・クロックϕ＝16 MHz時）

→ PWM周期：32.00 μs

○出力パルス幅（デューティ）：0，1/256，……，255/256

→ 分解能：160 ns（システム・クロックϕ＝12.5 MHz時）

125 ns（システム・クロックϕ＝16 MHz時）

11.2　構　　成

各部の機能は次のとおりです。PWMユニットの構成を図11－1に示します。

(1) PWMレジスタ

PWM出力のデューティ比（アクティブ・レベル幅）を設定します。任意のタイミングで書き換えることができます。

リセット入力により影響を受けません。

(2) PWMコントロール・レジスタ

PWM出力のアクティブ・レベルの設定，出力の制御を行います。

(3) 8ビット・カウンタ

ϕ/2のクロックでカウント・アップします。リセット入力により00Hになります。

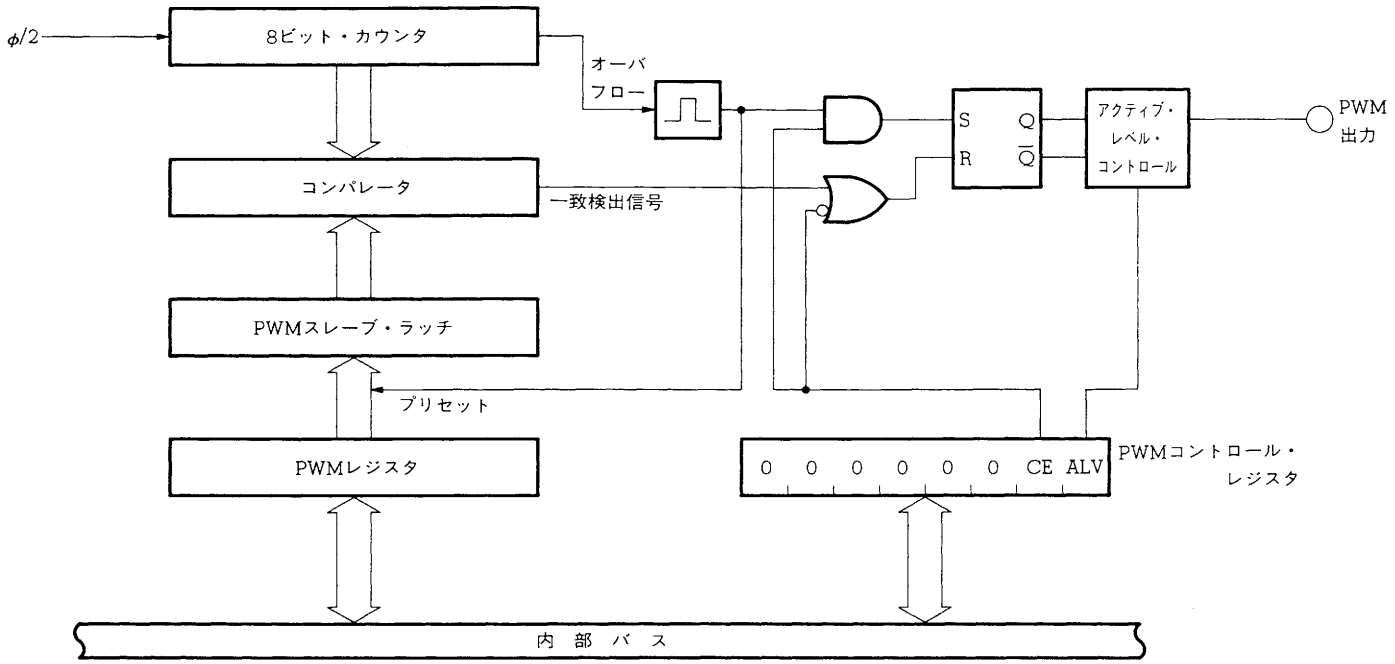
(4) PWMスレーブ・ラッチ

そのときのアクティブ・レベル幅を決定するレジスタです。

8ビット・カウンタのオーバフローで，PWMレジスタからダウンロードされます。

保守／廃止

図11－1　PWMユニットのブロック図



保守／廃止

11.3 動　　作

11.3.1 設　　定

ポート２モード・コントロール・レジスタ（PMC2）の対応ビットをセット(1)し，コントロール・モードに指定します。

その後の動作は，PWMコントロール・レジスタのCEビットにより制御されます。CEビット＝1にすると，次のタイム・ベース・カウンタ（TBC）の2^9からのタップ出力によりPWM端子のレベルがアクティブになり，PWMレジスタの値をPWMスレーブ・ラッチへ転送します。８ビット・カウンタは，ϕ/2のカウント・クロックでカウント・アップを続けます（**図11－3**参照）。

11.3.2 基本動作

以下の基本動作をCE＝0となるまで繰り返します。

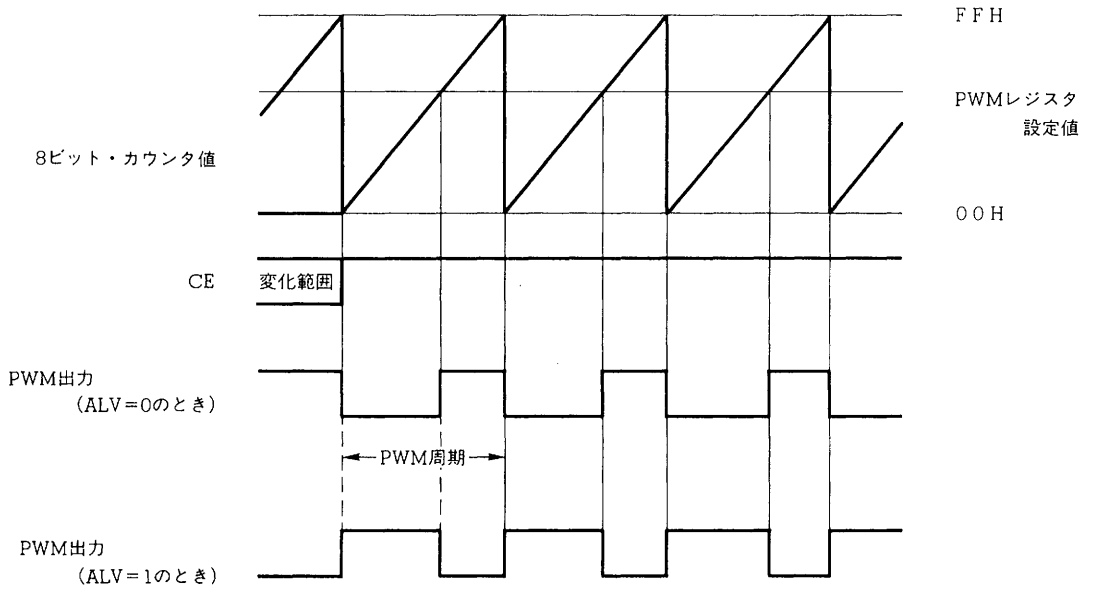
ただし８ビット・カウンタは，CEビットの値にかかわらず常にアップ・カウントを続けます。

① PWMレジスタへPWM出力のアクティブ・レベル幅を設定します。

② ８ビット・カウンタからのオーバフローにより，PWMスレーブ・ラッチへデータを転送します。PWM出力をセットし，アクティブ・レベルとします。

③ ８ビット・カウンタの値とPWMスレーブ・ラッチの値が一致したとき，PWM出力をインアクティブ・レベルとします。

図11－2　PWM出力動作タイミング



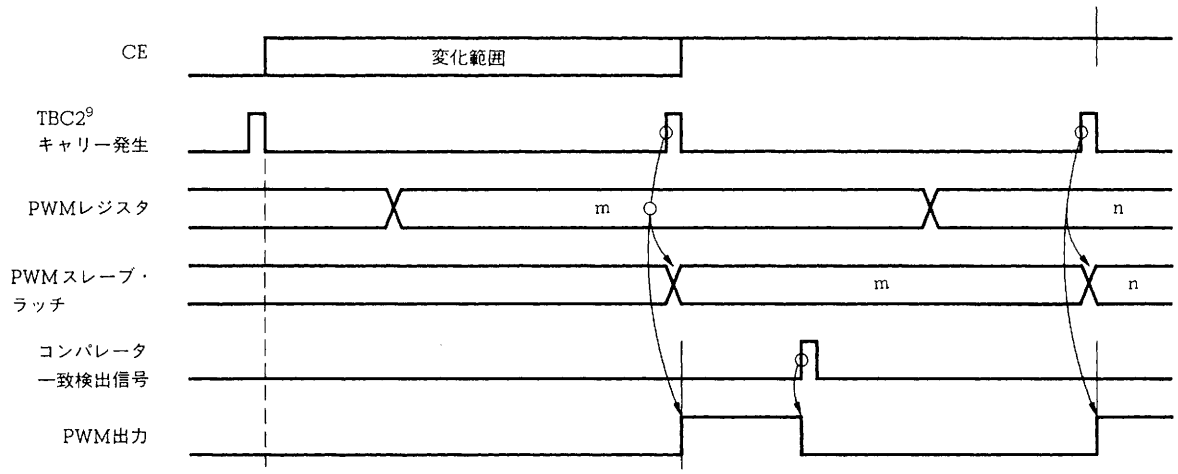
保守／廃止

11.3.3　PWM出力中の動作

次の動作は，PWM出力中でも有効となります。

- PWMレジスタの内容を任意のタイミングで書き換えることにより，PWM出力のアクティブ・レベル幅を変化させることができます（**11.4.2　PWMレジスタ（PWM）**参照）。
  ただしPWMレジスタの値は，8ビット・カウンタのオーバフローでPWMスレーブ・ラッチへプリセットされるため，現在出力中のアクティブ・レベル幅に影響は与えません。
- PWMコントロール・レジスタのCEビットをリセット(0)すると，PWM出力中でもただちにPWM出力をインアクティブにします（図**11－4**参照）。
- PWM出力のアクティブ・レベルは，PWMコントロール・レジスタのALVビットにより切り替えることができます。その場合，PWM周期の途中でもPWM出力を切り替えます（図**11－5**参照）。

図11－3　PWM出力動作タイミング（CE＝0→1にしたとき）



保守／廃止

図11－4　PWM出力動作タイミング（CE＝1→0にしたとき）

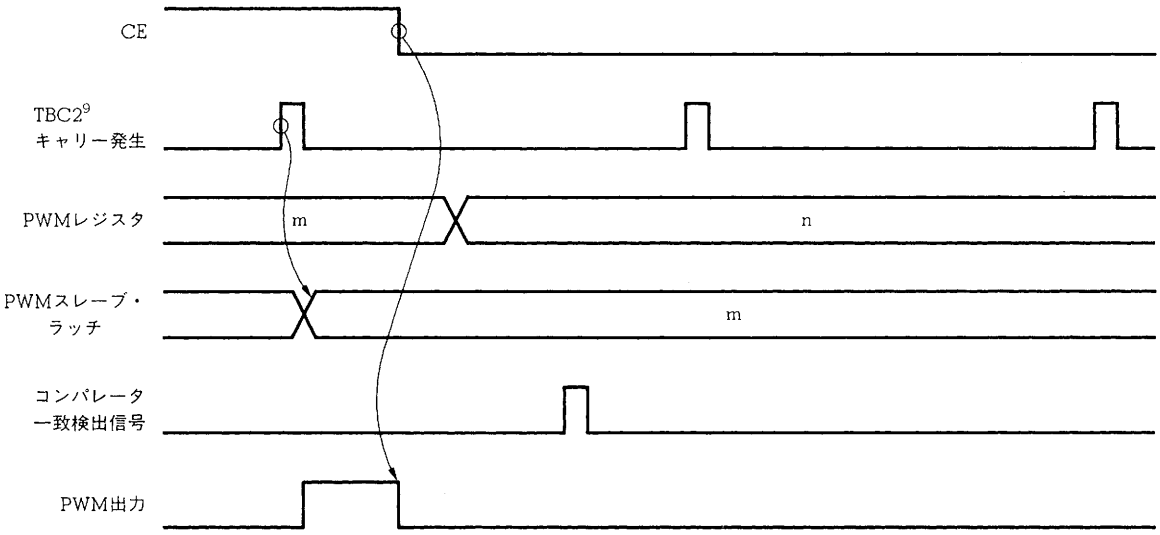
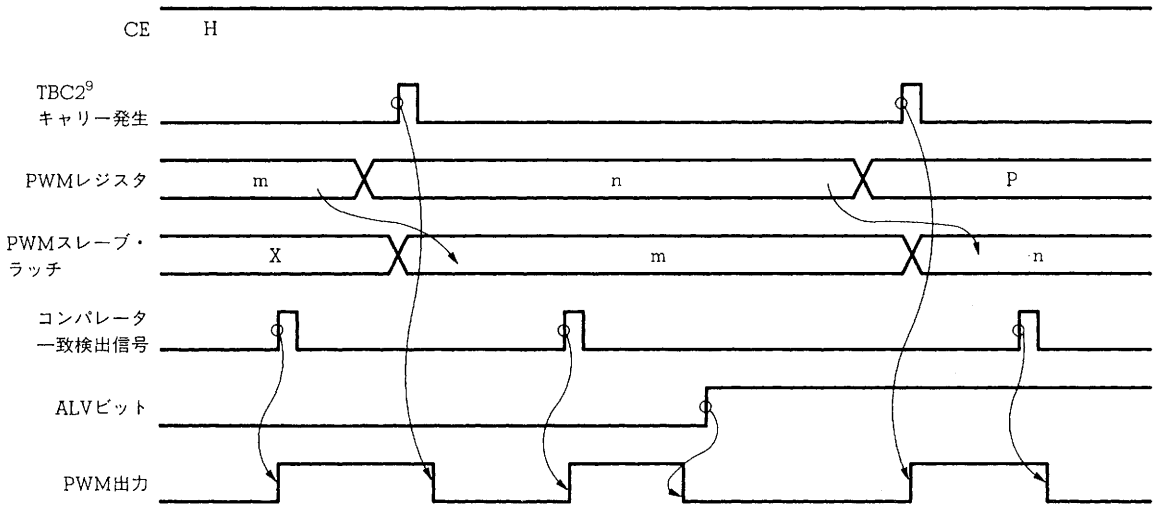


図11－5　PWM出力動作タイミング（ALVビットを変化させたとき）



保守／廃止

11.4　制御レジスタ

11.4.1　PWMコントロール・レジスタ（PWMC）

PWMレジスタ（PWM）の動作モードおよびPWM出力を制御する８ビットのレジスタです。

PWMCレジスタのリセット後の値は，00Hになります。

	7	6	5	4	3	2	1	0
PWMC	0	0	0	0	0	0	CE	ALV

PWM出力のアクティブ・レベル指定

ALV[注]	アクティブ・レベル
0	ロウ・レベル
1	ハイ・レベル

PWMの動作の指定

CE	PWM動作
0	動作停止（PWM出力は$\overline{\text{ALV}}$レベルで変化なし）
1	動作イネーブル（PWM出力は変化する）

注　ALVを切り替えたときは，切り替え直後に出力値が変わります。

備考　□ はリセット後の状態を示します。

11.4.2　PWMレジスタ（PWM）

PWM出力のアクティブ・レベル幅を設定する８ビットのレジスタです。PWMレジスタへの設定により，PWM出力のアクティブ・レベル幅 t_W は次式のように決まります。

t_W＝PWMレジスタの値 × t_{CYC}

t_{CYC}は，内部システム・クロック・サイクル・タイムで，$2/\phi$です。

表11－1　PWMレジスタの値とデューティ比

PWMレジスタの値	デューティ
00H	0[注]
01H	1/256
⋮	⋮
FFH	255/256

注　ALVの指定を反転し，CEをリセット(0)することにより，デューティを100％にすることができます。

保守／廃止

第12章　ウォッチドッグ・タイマ機能

ウォッチドック・タイマは，プログラムの暴走やデッドロックを防ぐための機能です。

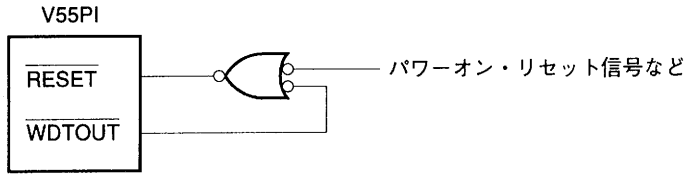
12.1　特　　徴

○３種のオーバフロー時間を設定可能(10.4，41.9，167.7[ms]：システム・クロックϕ＝12.5 MHz時)

(8.1，32.7，131.0 [ms]：システム・クロックϕ＝16 MHz時)

○$\overline{\text{RESET}}$端子に接続可能な出力端子装備（$\overline{\text{WDTOUT}}$端子）

★　図12－1　$\overline{\text{RESET}}$端子の接続例



12.2　構成と動作

ウォッチドッグ・タイマ割り込みが発生しないことで，プログラムまたはシステムが正常に動作していることを確認できます。ウォッチドッグ・タイマ機能の使用には，プログラム実行の一定間隔，またはサブルーチンの先頭などにウォッチドッグ・タイマをクリア（カウント開始）する命令（RSTWDT命令）を入れておく必要があります。

ウォッチドッグ・タイマをクリアする命令が設定した時間内に実行されずにウォッチドッグ・タイマがオーバフローすると，ウォッチドッグ・タイマ割り込み（INTWDT）を発生するとともに，$\overline{\text{WDTOUT}}$端子にロウ・レベルを出力し，プログラムの異常を知らせます。

図12－1にウォッチドッグ・タイマの構成を示します。

保守／廃止

(1) ウォッチドッグ・タイマ

ウォッチドッグ・タイマは，メイン・クロック発振回路の発振安定時間確保用タイマを兼ねており，システム・リセット入力およびSTOPモードに入るとカウンタ値がクリアされ，00Hになります。

いったんウォッチドッグ・タイマをスタートすると，ウォッチドッグ・タイマ・モード・レジスタ（WDM）の書き込みがハードウエアで禁止されるため，ソフトウエアによるオーバフローまでのインターバル・タイミングの変更，ウォッチドッグ・タイマの停止などはできません（ただし，ウォッチドッグ・タイマのクリアと，WDMレジスタの読み出しはいつでも可能です）。

ウォッチドッグ・タイマは，スタンバイ・モードにおいて，次のように動作します。

（STOPモード）

カウントが停止されます。NMI入力によるSTOPモード解除時，NMI端子への有効レベル入力終了後カウントが開始されます。ウォッチドッグ・タイマのオーバフローが発生すると内部クロック供給を再開します。このときINTWDTは発生しません（STOPモード解除後，ウォッチドッグ・タイマのオーバフロー発生までの時間内にメイン・クロック発振回路の発振が安定しないとシステム・クロックが不安定となり，誤動作する恐れがあります）。

（HALTモード）

カウントを継続します。したがって，INTWDTでHALTモードが解除されます。

（HOLDモード）

外部からのリクエストでCPUがホールド状態にあるときは，カウンタの内容を保持したまま，カウント動作を停止します。ホールド解除後，自動的にカウントを再開します。内部DMAによるホールド遷移はカウント停止しません。

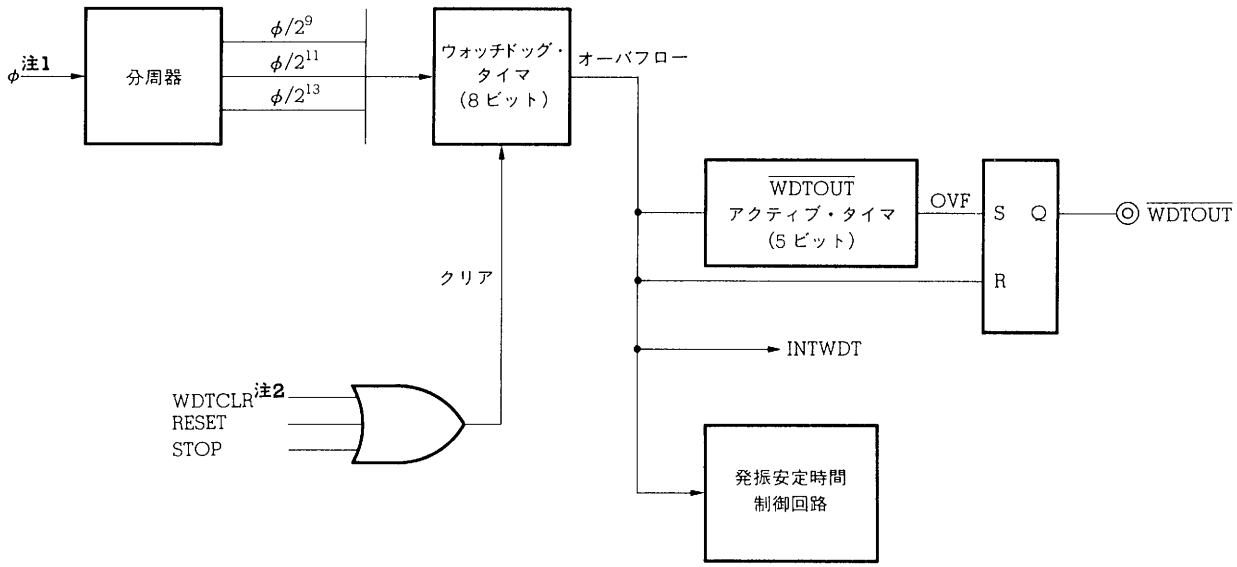
(2) $\overline{\text{WDTOUT}}$アクティブ・タイマ

ウォッチドッグ・タイマ割り込み要求発生時に出力される信号について，32クロック期間アクティブ・ロウを保つためのタイマです。$\overline{\text{WDTOUT}}$端子と$\overline{\text{RESET}}$端子を外部で接続することにより，ウォッチドッグ・タイマのオーバフローでシステム・リセットをかけることができます。

注意　電源投入時のリセット直後は，$\overline{\text{WDTOUT}}$端子の出力は不定となります。このときリセット入力から最大32クロック内にハイ・レベルとなります。

保守／廃止

図12－2　ウォッチドッグ・タイマの構成図



注1．φ：システム・クロック

2．WDTCLR：命令によるウォッチドッグ・タイマのクリア

ウォッチドッグ・タイマを用いた応用

プログラム開発の初期段階では，ウォッチドッグ・タイマは使用せず，プログラムの概略のディバグが終了したあと，ウォッチドッグ・タイマを含めたディバグを行います。

応用システムが異常になってからの応答時間が重要な場合には，オーバフロー時間が小さいモードを選択します。また，ウォッチドッグ・タイマとして使用しない場合は，ノンマスカブルのタイム・ベース割り込みとして使用することも可能です。

12.3　制御レジスタ

12.3.1　ウォッチドッグ・タイマ・モード・レジスタ（WDM）

WDMレジスタは，ウォッチドッグ・タイマの動作を制御する８ビットのレジスタです。

プログラムの暴走によってWDMレジスタの内容が誤って書き換えられないように，専用命令によってのみ書き込むことができます。

RSTWDT imm8，imm8′

この専用命令は，特殊なコード構成（４バイト）になっています。imm8とimm8′のオペコードが互いに補数でなければ，書き込みが行われません。

上記オペランドが補数でない場合には，例外トラップ割り込みが発生します（**5.8　例外トラップ**参照）。

システム・リセット（$\overline{\text{RESET}}$入力）後，WDMレジスタのRUNビット＝1に設定し，いったんウォッチ

保守／廃止

ドッグ・タイマをスタートさせると，ウォッチドッグ・タイマを停止させることができるのはシステム・リセットだけです。

RUNビット＝0に書き換えようとしてもその設定は無効となり，WDMレジスタの内容は変更しません。ただし，WDMレジスタの読み出しは，データ転送命令によりいつでもできます。

ウォッチドッグ・タイマは，専用命令（RSTWDT）の実行により常にクリアできます。なお，この命令ではオペコードimm8とimm8'の値が互いに補数の関係になるようにしてください（**V55PI　ユーザーズ・マニュアル　命令編**参照）。

リセット後の値は，00Hになります。

	7	6	5	4	3	2	1	0
WDM	RUN	0	0	0	0	WDI2	WDI1	0

オーバフロー時間の設定

WDI2	WDI1	カウント・クロック	オーバフロー時間	
			ϕ=12.5 MHz	ϕ=16 MHz
0	0	$\phi/2^9$	10.4 ms注	8.1 ms注
0	1	$\phi/2^{11}$	41.9 ms	32.7 ms
1	0	$\phi/2^{13}$	167.7 ms	131.0 ms
1	1	予約	設定禁止	設定禁止

（ϕ：システム・クロック）

ウォッチドッグ・タイマの指定

RUN	動作の指定
0	ウォッチドッグ・タイマの停止
1	ウォッチドッグ・タイマをクリアし，カウントを開始

注　STOPモード解除時の発振安定時間としてのカウント時間

備考　▨ はリセット後の状態を示します。

保守／廃止

第13章　A/Dコンバータ機能

V55PIは，４本のアナログ入力（ANI0-ANI3）を持つ高速，高精度の８ビット・アナログ／ディジタル（A/D)コンバータを内蔵しています。A/Dコンバータは，逐次比較方式を採用し，変換結果を保持する４本のA/Dコンバージョン・リザルト・レジスタ（ADCR0-ADCR3）を備えています。

13.1　特　　徴

○８ビットのA/D変換結果レジスタを４本内蔵

○アナログ入力端子（ANI0-ANI3）：４本

○A/Dコンバータの変換動作モード：２種類

- ●スキャン・モード：複数のアナログ入力を順次選択し，変換する。
- ●セレクト・モード：アナログ入力を１端子に固定し，連続的に変換する。

○変換起動方法：２種類

- ●ハードウエア・スタート：トリガ入力（INTP4）による起動
- ●ソフトウエア・スタート：A/Dコンバータ・モード・レジスタ（ADM）のビット設定による起動

○変換終了割り込み要求（INTAD）発生

保守／廃止

13.2　構　　成

アナログ入力（ANI0-ANI3）は４本あり，A/Dコンバータ・モード・レジスタ（ADM）により選択します。選択したアナログ入力は，電圧コンパレータの一方の入力になります。

各部の機能は次のとおりです。

（1）入力回路

A/Dコンバータ・モード・レジスタ（ADM）の指定によりアナログ入力を選択し，動作モードに従ってアナログ入力をサンプリング＆ホールド回路に送ります。

（2）サンプリング＆ホールド回路

順次送られてくるアナログ入力を１つ１つサンプリングし，A/D変換中のアナログ入力を保持します。

（3）電圧コンパレータ

アナログ入力の電圧と直列抵抗ストリングの電圧タップの電圧を比較します。

（4）直列抵抗ストリング

アナログ入力の電圧と比較する電圧を発生させるためのものです。

直列抵抗ストリングは，A/Dコンバータ用の基準電圧端子（AV_{REF}）とA/Dコンバータ用のGND端子（AV_{SS}）間に接続されています。２端子間を256の等価な電圧ステップにするため，等価な抵抗255個と，その1/2の抵抗値の抵抗２個で構成されています。

直列抵抗ストリングの電圧タップは，SARレジスタで制御されるタップ・デコーダによって選択します。

（5）８ビットSARレジスタ（逐次変換レジスタ）

SARレジスタは，直列抵抗ストリングの電圧タップの値がアナログ入力の電圧値と一致するように，最上位ビット（MSB）から１ビットずつ設定されます。

最下位ビット（LSB）まで設定すると（A/D変換終了），SARレジスタの内容（変換結果）がA/Dコンバージョン・リザルト・レジスタに保持されます。

（6）A/Dコンバージョン・リザルト・レジスタ（ADCR0-ADCR3）

A/D変換結果を保持する８ビット・レジスタです。A/D変換が終了するたびに，SARレジスタから変換結果がロードされます。データがロードされると変換終了割り込み要求（INTAD）が発生します。

リセットにより不定となります。

４本のアナログ入力（ANI0-ANI3）に，４つのレジスタ（ADCR0-ADCR3）が１対１で次のよう

保守／廃止

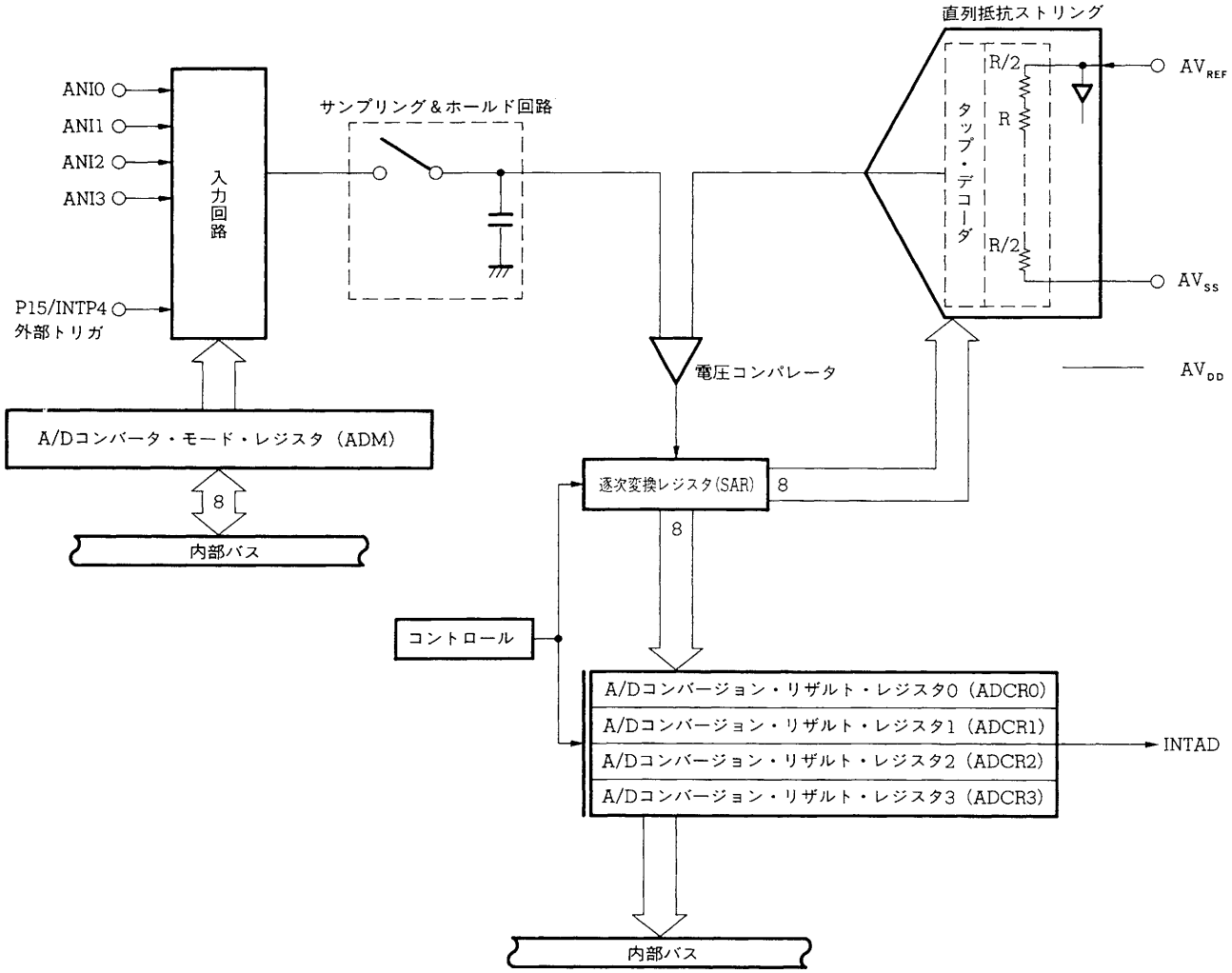
に対応しています。

アナログ入力	ADCRレジスタ
ANI0	ADCR0
ANI1	ADCR1
ANI2	ADCR2
ANI3	ADCR3

注意　**ANI0-ANI3**端子には，**A/D**変換使用時，未使用時どちらの場合もAV_{SS}-AV_{REF}の範囲外の電圧が加わらないようにしてください。

保守／廃止

図13－1　A/Dコンバータのブロック図



保守／廃止

13.3 動　　作

A/Dコンバータは，セレクト・モードとスキャン・モードの２つの動作モードから一方を選択できます。次に各モードに共通する基本動作内容について説明し，続けてそれぞれのモードについて説明します。

13.3.1 A/Dコンバータの基本動作

(1) A/D変換動作手順

A/D変換は，次のような手順で行います。

(a) アナログ入力の選択と動作モードの指定をA/Dコンバータ・モード・レジスタ（ADM）によって設定します。

(b) ADMレジスタのビット７（CS）をセット(1)し，A/D変換を開始させます。

(c) 変換開始とともにSARレジスタの最上位ビット（ビット７）が自動的にセット(1)されます。

(d) SARレジスタのビット７がセットされるとタップ・デコーダは，直列抵抗ストリングの電圧タップを$\frac{255}{512}AV_{REF}$（$\fallingdotseq\frac{1}{2}AV_{REF}$）にします。

(e) 直列抵抗ストリングの電圧タップとアナログ入力との電圧差を電圧コンパレータで比較します。もし，アナログ入力が (1/2) AV_{REF}よりも大きければ，SARレジスタのMSBをセットしたままです。

また，もし (1/2) AV_{REF}よりも小さければ，MSBをリセットします。

(f) 次にSARレジスタのビット６が自動的にセット(1)され，次の比較に移ります。ここではすでに結果がセットされているビット７の値によって，次に示すように直列抵抗ストリングの電圧タップが選択されます。

・ビット７＝1……$\frac{383}{512}AV_{REF}\fallingdotseq\frac{3}{4}AV_{REF}$

・ビット７＝0……$\frac{127}{512}AV_{REF}\fallingdotseq\frac{1}{4}AV_{REF}$

この電圧タップとアナログ入力電圧を比較し，その結果でSARレジスタのビット６が次のように操作されます。

・アナログ入力電圧≧電圧タップ：ビット６=1

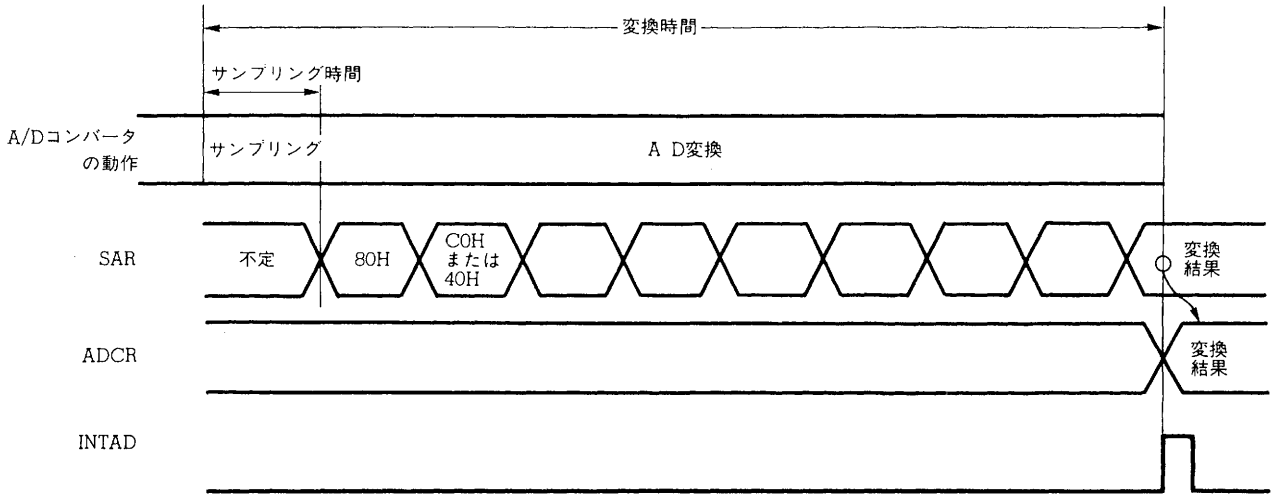
・アナログ入力電圧＜電圧タップ：ビット６=0

(g) このような比較をSARレジスタの最下位ビット（ビット０）まで続けます（バイナリ・サーチ法）。

保守／廃止

(h) 8ビットの比較が終了したとき，SARレジスタには有効なディジタルの結果が残り，その値がA/Dコンバージョン・リザルト・レジスタ（ADCR）に転送され，ラッチされます。
同時に，A/D変換終了割り込み要求（INTAD）が発生します。

図13-2　A/Dコンバータの基本動作



(2) A/D変換動作の制御

A/D変換動作の起動方法には次の2つがあります。
ADMレジスタのTRGビットの指定によりどちらかを選択できます。

●ADMレジスタのCSビットによる起動（TRG=0のとき）

CSビット=1にすることにより変換動作を開始します。
ADMレジスタのビット0-2（MS，AN0，AN1）で指定したチャネルのA/D変換を繰り返し実行します。1回の変換動作が終了すると，次の変換動作を順に実行します。

●外部端子トリガのINTP4入力による起動（TRG=1のとき）

INTP4への有効エッジ入力待ち状態で停止し，入力されると変換動作を開始します。
一度外部トリガ入力により変換動作を開始したあとは，TRGビットの再セット(1)またはINTP4への有効エッジ入力を行っても変換動作は停止しません。変換動作シーケンスが初期化されるだけです。したがって，動作を停止し再び外部トリガ入力による変換動作の開始待ち状態にするためには，次の手順で設定してください。

① CS←0，TRG←1（変換動作停止）
② CS←1，TRG←1

A/D変換動作の停止は，CSビットをリセット(0)することにより行います。変換動作はそれまで連続して行われ，CSビットをリセットすると変換動作中でも動作を中止します。

保守／廃止

A/D変換動作中に，ADMレジスタに対する書き込み操作を行うと変換動作シーケンスは初期化されたあと継続されます。

変換動作の時間は，ADMレジスタのFRビットで設定できます。この時間は，変換動作を開始してから変換結果がADCRレジスタにラッチされるまでの時間です。

ADCRレジスタは，リセットによる影響を受けません。

(3) A/D変換時間

A/D変換に要する時間は，内部システム・クロック周波数(ϕ)とADMレジスタのFRビットによって決定されます。

A/D変換時間は，1回のA/D変換に要するすべての時間を含んでおり，サンプリング時間も含みます。

表13－1にその値を示します。

表13－1　A/D変換時間

システム・クロック（ϕ）の範囲	FRビット	変換時間	サンプリング時間
8 MHz＜ϕ＜16 MHz	0	160/ϕ (10 μs-20 μs)	32/ϕ (3 μs-6 μs)
4 MHz＜ϕ≦8 MHz	1	120/ϕ (15 μs-30 μs)	24/ϕ (2 μs-4 μs)

13.3.2　セレクト・モード

(1) 設　　定

ADMレジスタのビット0（MS）をセット(1)します。

(2) 動　　作

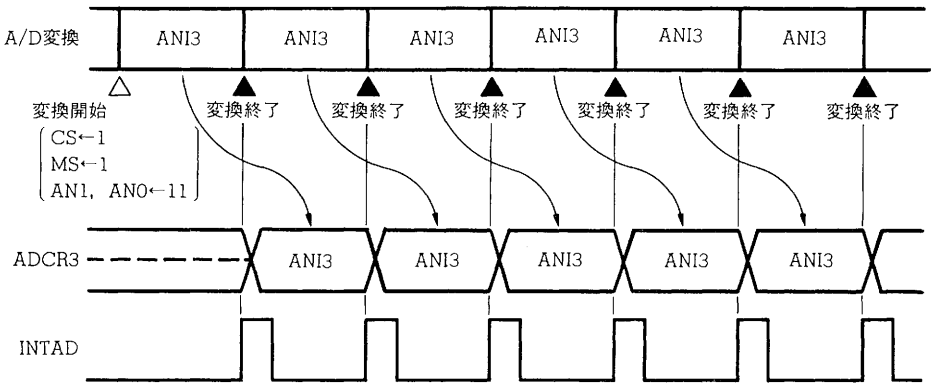
ADMレジスタのビット1，2（AN0，AN1）により1つのアナログ入力を指定し，指定されたアナログ入力端子のA/D変換を開始します。

選択されたアナログ入力の変換動作が終了すると，変換結果はアナログ入力に1対1に対応したA/Dコンバージョン・リザルト・レジスタ（ADCR）に格納され，割り込み（INTAD）を発生します。割り込み要求が受け付けられたかどうかに関係なく，再び同じアナログ入力の変換動作を開始します。この動作はADMレジスタが書き換えられるまで続きます。

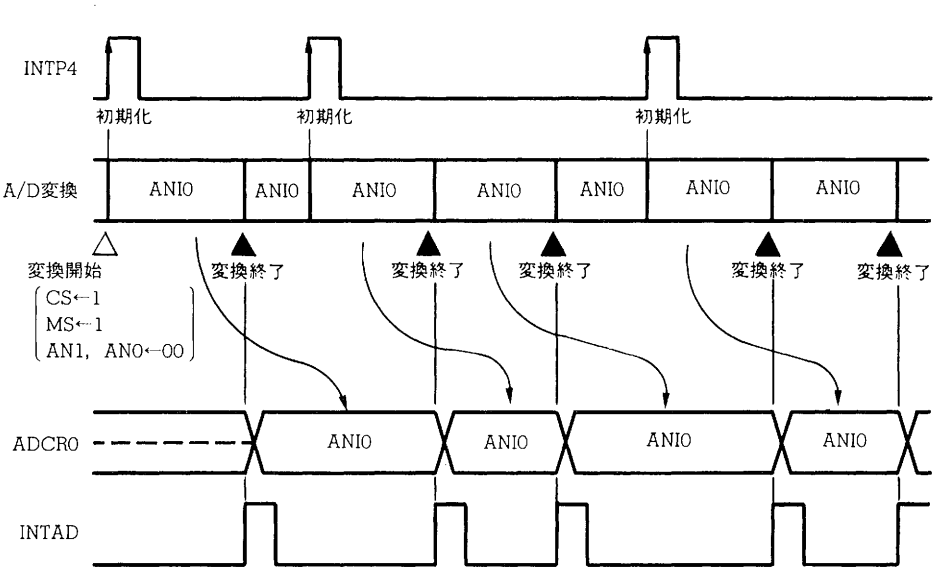
保守／廃止

図13-3　セレクト・モードの動作タイミング

(a) TRGビット←0



(b) TRGビット←1



13.3.3　スキャン・モード

(1) 設　　定

ADMレジスタのビット0（MS）をリセット（0）します。

(2) 動　　作

ADMレジスタのビット1，2（AN0，AN1）により指定されたアナログ入力端子からの入力を順に選択し変換します。

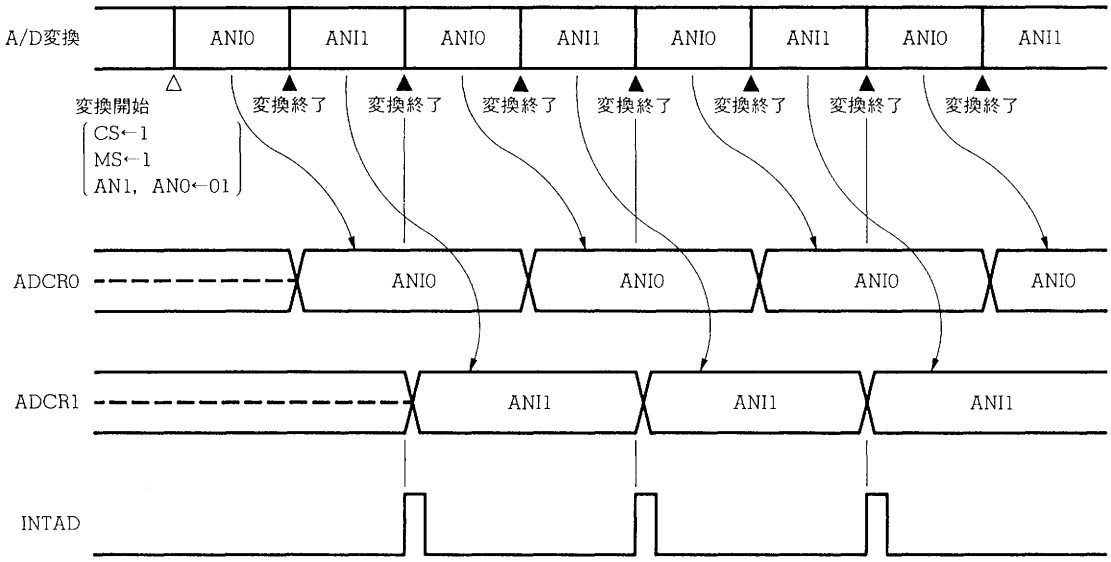
たとえば，AN1，AN0＝01のときはANI0とANI1が繰り返しスキャン（ANI0→ANI1→ANI0→ANI1

保守／廃止

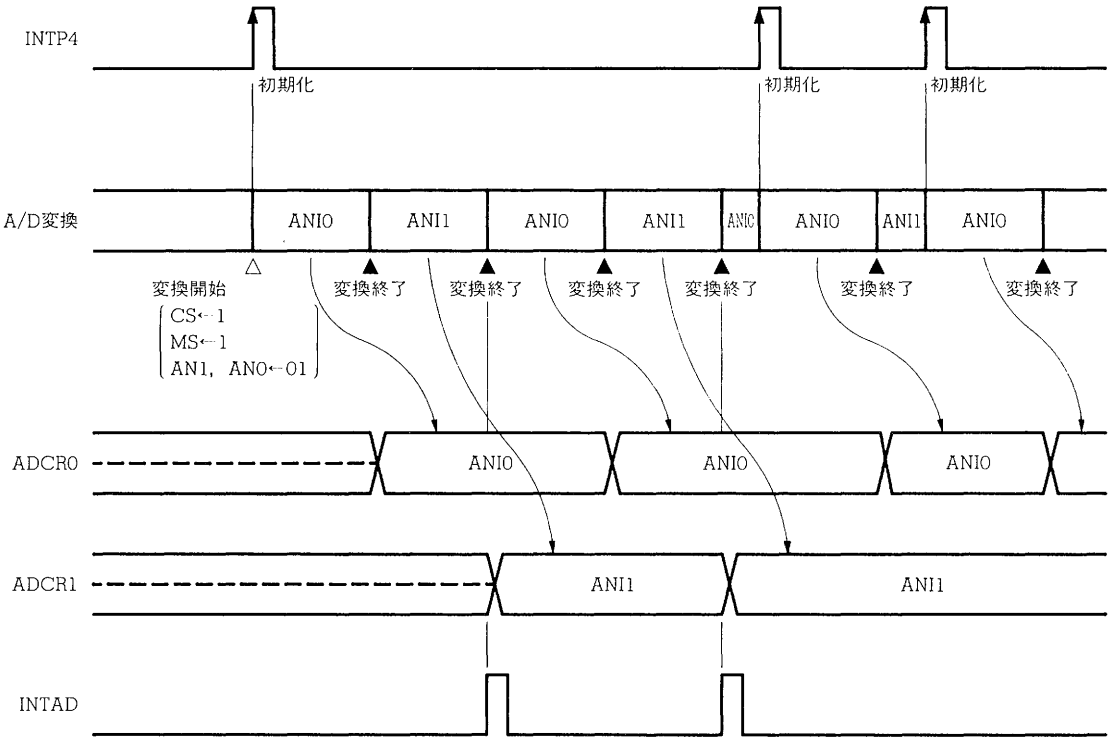
…)されます。変換結果は，アナログ入力に1対1に対応したADCRレジスタに格納されます。チャネル0から順次変換していき，ADMレジスタで指定された入力チャネルの変換をすべて終了すると割り込み要求（INTAD）を発生し，再びチャネル0から変換を開始します。

図13－4　スキャン・モードの動作タイミング

(a) TRGビット←0



(b) TRGビット←1



保守／廃止

13.4　制御レジスタ

13.4.1　A/Dコンバータ・モード・レジスタ（ADM）

A/Dコンバータ・モード・レジスタは，A/Dコンバータの動作を制御するためのレジスタです。

ADMレジスタのリセット後の値は，00Hになります。

	7	6	5	4	3	2	1	0
ADM	CS	TRG	0	FR	0	AN1	AN0	MS

変換動作モードの指定

MS	変換動作モード
0	スキャン・モード
1	セレクト・モード

アナログ入力の指定

AN1	AN0	アナログ入力	
		スキャン・モード時	セレクト・モード時
0	0	ANI0入力をスキャン	ANI0入力をセレクト
0	1	ANI0，ANI1入力をスキャン	ANI1入力をセレクト
1	0	ANI0, ANI1, ANI2入力をスキャン	ANI2入力をセレクト
1	1	ANI0, ANI1, ANI2, ANI3入力をスキャン	ANI3入力をセレクト

変換動作時間の切り替え

FR	変換動作時間
0	160クロック（8 MHz＜ϕ＜16 MHzのとき）
1	120クロック（4 MHz＜ϕ≦8 MHzのとき）

外部端子トリガの制御

TRG	トリガ制御
0	外部端子トリガ禁止
1	外部端子トリガ許可

A/D変換動作制御

CS	変　換　動　作
0	A/D変換動作停止
1	A/D変換動作開始

備考　□ はリセット後の状態を示します。

保守／廃止

FR 変換動作時間の切り替え

発振周波数を変化させた場合でも，A/D変換時間を大きく変化させないようにするための制御ビットです。内部システム・クロックϕとFRビットにより，1回の変換速度が決まります。

4 MHz＜ϕ≦8 MHzのときには，FR＝1にしてください。

CS A/D変換動作制御

CS＝0にすると，変換中であってもすべての変換動作を中止します。このとき，ADCRレジスタの更新は行われません。また割り込み要求（INTAD）も発生しません。

13.4.2　A/Dコンバージョン・リザルト・レジスタ（ADCR0-ADCR3）

A/Dコンバージョン・リザルト・レジスタはA/D変換結果を保持する8ビットのレジスタです。

13.5　割り込み要求

A/Dコンバータ割り込み要求（INTAD）は，選択されたアナログ入力の変換結果がADCRレジスタに格納されたときに発生します。割り込み形態は割り込み要求制御レジスタにより制御されます。

割り込み要求制御レジスタについては，**5.5.3　割り込み要求制御レジスタ（IC9-14，16-32，36，37）**を参照してください。

13.6　注意事項

13.6.1　STOP命令を実行する場合について

A/Dコンバータの動作中にSTOPモードにする場合は，必ずA/Dコンバータの変換動作を停止しておいてください。これは，A/Dコンバータがサンプリング動作中にSTOP状態になると，サンプリング・ゲートが開いたままとなるためです。A/Dコンバータ・モード・レジスタのCSビット＝0にすると，変換中であってもすべての変換動作を中止します。

13.6.2　A/D変換動作シーケンスの初期化について

外部トリガ端子への有効エッジ入力やA/Dコンバータ・モード・レジスタのTRGビットの再セット（1→1）によって，A/D変換動作シーケンスは初期化されます。ただし，外部端子への有効エッジ入力の周期は，必ず変換動作の1周期よりも長くしてください。繰り返し外部端子へ有効エッジ入力を行ってA/D変換を行わせる場合などで，有効エッジ入力の周期が変換動作シーケンスの1周期より短いと，変換が行われない可能性があります。

保守／廃止

13.6.3　変換結果の取りこぼしについて

A/Dコンバータのセレクト・モード使用時に，A/D変換結果の読み出しをベクタ割り込みで行っている場合，他の割り込み処理などによって，A/D変換終了割り込みが長時間(FRビットが0の場合160クロック，FRビットが1の場合120クロック以上)保留されると変換結果を取りこぼしてしまいます。この変換結果の取りこぼしをソフトウエア的に検出することは困難です。したがって，変換結果の取りこぼしが発生しないよう，次のような対策をとってください。

・他の割り込み処理時間をA/D変換の時間に対して十分短くする。
・多重割り込みを使用して，他の割り込み処理中でもA/D変換終了割り込みを受け付けるようにする。
・A/D変換終了割り込みをマクロ・サービスで行うようにする。

保守／廃止

第14章　ポート機能

V55PIの端子は，一部を除いてポート端子，コントロール端子が兼用となっています。各端子の詳細については，**2.1　端子機能一覧**を参照してください。

14.1　特　　徴

- ○１ビットごとに入力ポート，出力ポート，コントロール機能の選択が可能
- ○入出力ポート専用端子：8
- ○外部割り込み要求入力兼用入力ポート端子：7
- ○アナログ入力兼用入力ポート端子：4
- ○その他コントロール機能兼用入出力ポート端子：34

14.2　構　　成

V55PIのポート０，２-５，７，８は基本的に図14-１に示す３ステート双方向ポートで構成されています。

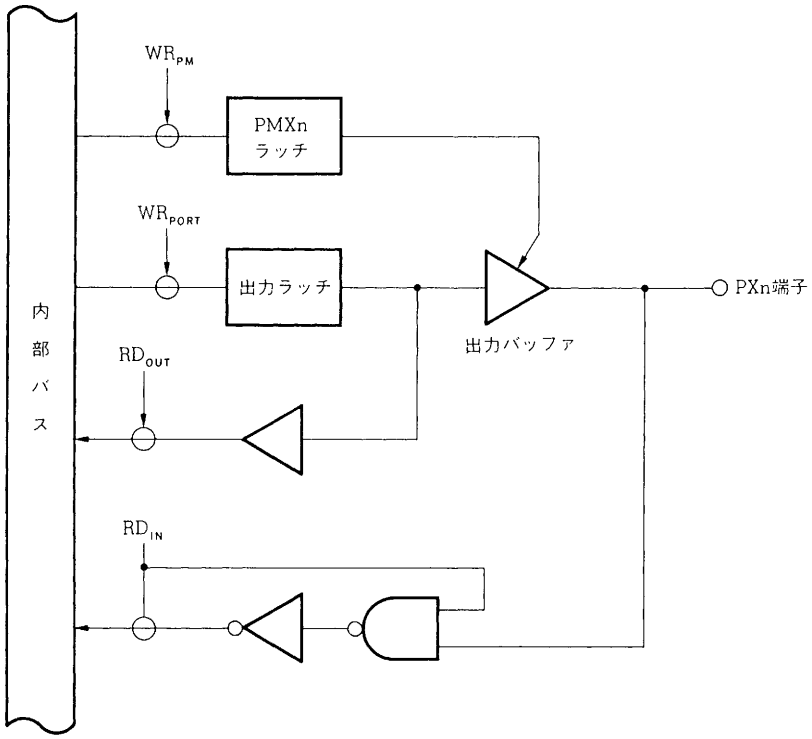
各ポートのポート・モード・レジスタ（PMn）で，ビットごとに入力／出力ポートに指定することができます。

$\overline{\text{RESET}}$入力によりポート・モード・レジスタの各ビット（PMXn）はセット(1)され，入力ポートに指定されます。このとき，すべてのポート端子はハイ・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。

以降，表中の  の部分は，リセット後の状態を示します。

保守／廃止

図14－1　ポート0，2－5，7，8の構成



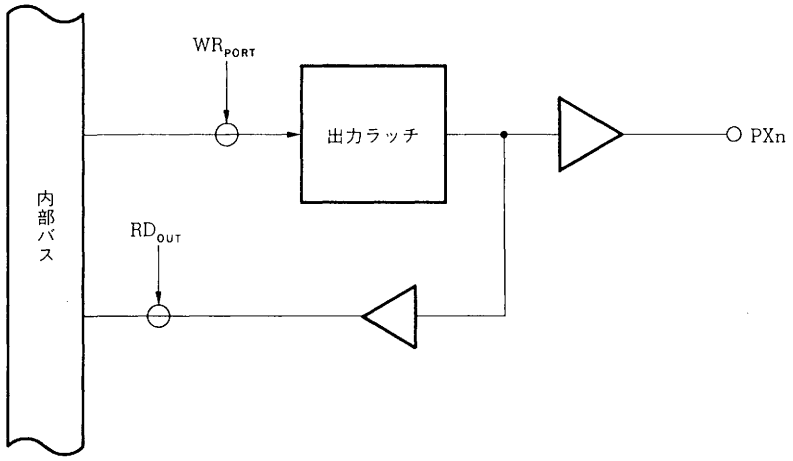
備考　Xはポート番号を，nはポート・モード・レジスタ内のビット番号を示します。

保守／廃止

14.2.1　出力ポートに指定された場合（PMXn＝0）

出力ラッチが有効となり，転送命令などで出力ラッチへデータ転送することができます。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。

図14－2　出力ポート指定のポート



備考　Xはポート番号を，nはポート・モード・レジスタ内のビット番号を示します。

出力ポートの出力状態をチェックする機能として出力ポートを読み出す際に，出力ラッチの値を読み出すか，あるいは端子のレベルを読み出すかをソフトウエアで指定することができます。

このための制御レジスタとしてポート・リード・コントロール・レジスタ（PRDC）があります。

	7	6	5	4	3	2	1	0
PRDC	0	0	0	0	0	0	0	PRDC0

ポート	動作	PRDC0	データ
入力ポート	リード	0	端子レベル
		1	0を読み込む
出力ポート	リード	0	出力ラッチの値
		1	端子レベル

CPUのSFRアクセスにより，出力ポートは次のように動作します。

リード動作　：①　PRDC＝0のとき……出力ラッチの値を読み込みます。

②　PRDC＝1のとき……端子レベルを読み込みます。

ライト動作　：ポートの出力ラッチに書き込みを行うと同時に，端子に出力します。

保守／廃止

PRDCレジスタの操作により，ユーザ・プログラムで端子チェックを行う際に次の点に注意します。

(1) PRDC＝1でのポートのビット操作命令

端子アクセス・モード(PRDC＝1)では，ポートへのビット操作命令は正常に動作しません。ポートのチェック終了後は，必ず通常モード（PRDC＝0）に戻します。

(2) PRDC＝1での割り込みの発生

端子アクセス・モードにおいて割り込みが発生すると，割り込み処理ルーチン内でポート操作(ビットのセット，ポートへの書き込みなど）を行う可能性があり，誤動作の原因となります。チェック動作を開始する前に割り込み禁止状態（DI状態）にしておく必要があります。また，ポートを操作するマクロ・サービスは使用しないでください。

(3) ノンマスカブル割り込みの発生

ノンマスカブル割り込みの発生は避けられませんので，システムに応じて，プログラム上で次のどちらかの処置をするようにしてください。

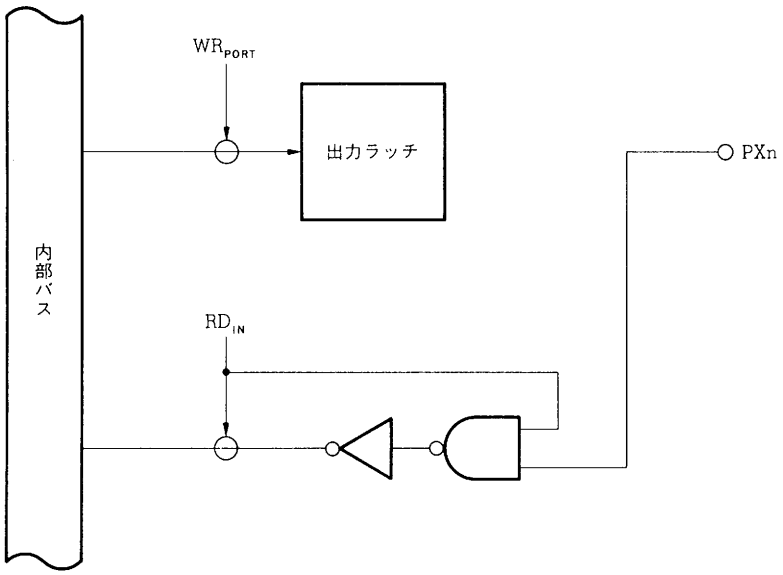
- ノンマスカブル割り込みルーチンではポート操作は行わない。
- ノンマスカブル割り込み処理ルーチンの最初でPRDCビットを退避して，メイン・ルーチンにリターンする際に復帰させる。

保守／廃止

14.2.2　入力ポートに指定された場合（PMXn＝1）

ポート端子のレベルを転送命令などで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，転送命令によりデータは，ポートの入出力指定に関係なく，すべての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読み出すことはできません。

図14－3　入力ポート指定のポート



備考　Xはポート番号を，nはポート・モード・レジスタ内のビット番号を示します。

CPUのSFRアクセスにより，入力ポートは次のように動作します。

リード動作：①　PRDC＝0のとき　……端子レベルを読み込みます。
②　PRDC＝1のとき　……“0”を読み込みます。

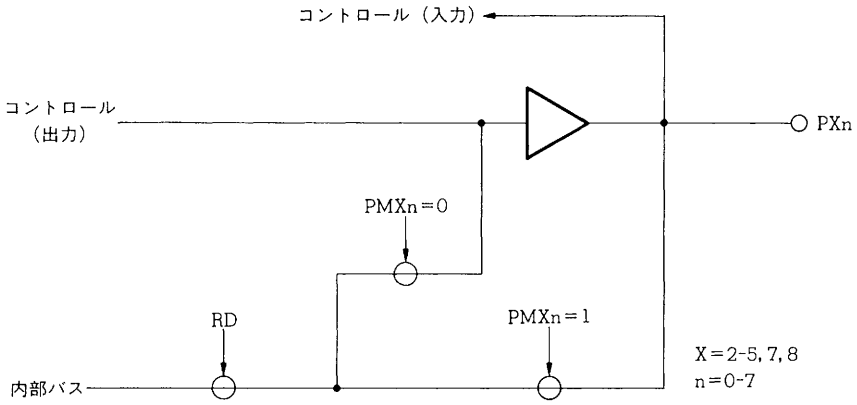
ライト動作：①　入力専用ポートの場合……ポートに対する影響はありません。
CPUはSFRアクセス・サイクルを実行します。
②　入出力ポートの場合　……ポートの出力ラッチに書き込みを行いますが端子には出力しません（書き込まれたデータは保持され，出力ポートに設定された時点でデータを端子から出力します）。

保守／廃止

14.2.3　コントロール指定の場合（PMXn＝1）

ポートはポート・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，またはバイト単位にコントロール信号入力または出力端子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14－4　コントロール指定のポート



備考　Xはポート番号を，nはポート・モード・レジスタ内のビット番号を示します。

(1) ポートがコントロール信号出力の場合

ポート・モード・レジスタの各ビット（PMXn)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。

(2) ポートがコントロール信号入力の場合

ポート・モード・レジスタの各ビット（PMXn）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。

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出力ラッチの状態が読み込まれます。

保守／廃止

14.3　ポートの各端子機能

各ポートには対応した制御レジスタが用意されており，これらレジスタのビット指定でポート各端子を汎用ポートとして使用するか，コントロール信号入出力端子として使用するか，また特にポート・モードのとき入出力どちらで使用するかを指定できます。

14.3.1　ポート0（P00-P07）

8ビットの汎用入出力ポートです。ポート0モード・レジスタの指定により，ビット単位に入力／出力の切り替えができます。

表14-1　ポート0の動作（n=0-7）

	ポート・モード	
	PM0n=1	PM0n=0
P00	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	入力ポート	出力ポート

(1) ポート0モード・レジスタ（PM0）

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

$\overline{\text{RESET}}$入力によって，すべてのビットがセット(1)され，入力ポートになります。

	7	6	5	4	3	2	1	0
PM0	PM07	PM06	PM05	PM04	PM03	PM02	PM01	PM00

PM0n	入出力指定
0	出力
1	入力

保守／廃止

14.3.2　ポート１（P10-P16）

７ビットの特殊入力ポートです。P10-P16は入力専用ポートとして機能するほか，外部割り込み入力，タイマ・ユニットへのキャプチャ・トリガ入力端子，またはA/Dコンバータ・トリガ入力端子として機能します。

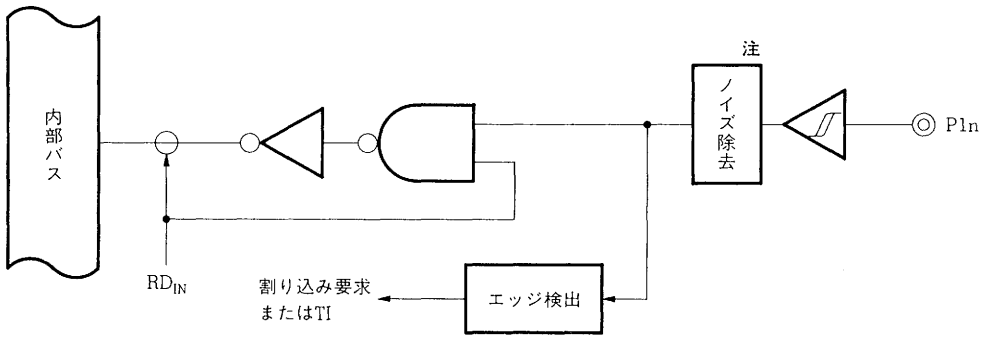
このポートは割り込み入力端子と兼用となっていますが，割り込み機能と入力ポートの機能を切り替えることはできません。

入力ポート機能のみを使用したいときは該当する割り込みおよびキャプチャ動作をマスクしてください（システム・リセットによりNMI以外，自動的にマスク状態，およびキャプチャ動作禁止状態になります）。

表14－2　ポート１の動作

	機　　能
P10	NMI入力　　兼　入力専用ポート
P11	INTP0入力　　兼　入力専用ポート
P12	INTP1入力　　兼　入力専用ポート
P13	INTP2入力　　兼　入力専用ポート
P14	INTP3入力／TI入力　　兼　入力専用ポート
P15	INTP4入力　　兼　入力専用ポート
P16	INTP5入力　　兼　入力専用ポート

図14－5　ポート１のブロック図



注　P10/NMI端子のみ。

備考　n＝0-6

保守／廃止

14.3.3　ポート２（P20-P25）

６ビットの汎用入出力ポートです。ポート２モード・レジスタ（PM2）の設定により，ビット単位に入力／出力の指定ができます。

ポート２モード・コントロール・レジスタ（PMC2）の設定により，PWM出力，タイマ・ユニットの出力端子としても機能します。

表14－３　ポート２の動作（n=0-5）

	コントロール・モード	ポート・モード	
	PMC2n＝1	PMC2n＝0	
		PM2n＝1	PM2n＝0
P20	PWM出力	入力ポート	出力ポート
P21	TO00出力	入力ポート	出力ポート
P22	TO01出力	入力ポート	出力ポート
P23	TO20出力	入力ポート	出力ポート
P24	TO21出力	入力ポート	出力ポート
P25	TO30出力	入力ポート	出力ポート

(1) 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・モードの設定を，ビット単位で行う８ビットのレジスタです。

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

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レジスタはすべてのビットがリセット(0)され，ポート・モードになります。

	7	6	5	4	3	2	1	0
PMC2	0	0	PMC25	PMC24	PMC23	PMC22	PMC21	PMC20

PMC2n	モード指定
0	ポート・モード
1	コントロール・モード

保守／廃止

(2) ポート２モード・レジスタ（PM2）

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。

PMC2レジスタの対応ビットがリセット(0)されているとき（ポート・モードのとき）PM2レジスタの各ビットは有効になります。8/1ビット操作のメモリ・アクセスによって，このレジスタの読み出し／書き込みができます。

$\overline{\text{RESET}}$入力によって，すべてのビットがセット(1)され，入力ポートになります。

	7	6	5	4	3	2	1	0
PM2	1	1	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	入出力指定
0	出力
1	入力

14.3.4　ポート３（P30-P36）

７ビットの汎用入出力ポートです。ポート３モード・レジスタ（PM3）の設定により，ビット単位に入力／出力の指定ができます。

ポート３モード・コントロール・レジスタ（PMC3）の設定により，シリアル用入出力端子としても機能します。

表14－4　ポート３の動作（n＝0-6）

	コントロール・モード	ポート・モード	
	PMC3n＝1	PMC3n＝0	
		PM3n＝1	PM3n＝0
P30	TxD0出力/SB0入出力/SO0出力	入力ポート	出力ポート
P31	RxD0入力/SB1入出力/SI0入力	入力ポート	出力ポート
P32	TxC出力/$\overline{\text{SCK0}}$入出力	入力ポート	出力ポート
P33	$\overline{\text{CTS0}}$入力	入力ポート	出力ポート
P34	TxD1出力/SO1出力	入力ポート	出力ポート
P35	RxD1入力/SI1入力	入力ポート	出力ポート
P36	$\overline{\text{CTS1}}$入力/$\overline{\text{SCK1}}$入出力	入力ポート	出力ポート

保守／廃止

(1) ポート３モード・コントロール・レジスタ（PMC3）

ポート３のポート／コントロール・モードの設定を，ビット単位で行う８ビットのレジスタです。

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

PMC3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，PMC3レジスタはすべてのビットがリセット(0)され，ポート・モードになります。

	7	6	5	4	3	2	1	0
PMC3	0	PMC3 6	PMC3 5	PMC3 4	PMC3 3	PMC3 2	PMC3 1	PMC3 0

PMC3n	モード指定
0	ポート・モード
1	コントロール・モード

(2) ポート３モード・レジスタ（PM3）

ポート３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。

PMC3レジスタの対応ビットがリセット(0)されているとき（ポート・モードのとき）PM3レジスタの各ビットは有効になります。8/1ビット操作のメモリ・アクセスによって，このレジスタの読み出し／書き込みができます。

$\overline{\text{RESET}}$入力によって，すべてのビットがセット(1)され，入力ポートになります。

	7	6	5	4	3	2	1	0
PM3	1	PM3 6	PM3 5	PM3 4	PM3 3	PM3 2	PM3 1	PM3 0

PM3n	入出力指定
0	出力
1	入力

14.3.5　ポート４（P40-P47）

８ビットの汎用入出力ポートです。ポート４モード・レジスタ（PM4）の設定により，ビット単位に入力／出力の指定ができます。

ポート４モード・コントロール・レジスタ（PMC4）の設定によりパラレル・インタフェースのデータ入出力端子（PD0-PD7）としても機能します。

保守／廃止

表14－5　ポート4の動作（n＝0-7）

	コントロール・モード	ポート・モード	
	$PMC4_7$＝1	$PMC4_7$＝0	
		PM4n＝1	PM4n＝0
P40	PD0入出力	入力ポート	出力ポート
P41	PD1入出力	入力ポート	出力ポート
P42	PD2入出力	入力ポート	出力ポート
P43	PD3入出力	入力ポート	出力ポート
P44	PD4入出力	入力ポート	出力ポート
P45	PD5入出力	入力ポート	出力ポート
P46	PD6入出力	入力ポート	出力ポート
P47	PD7入出力	入力ポート	出力ポート

(1) ポート4モード・コントロール・レジスタ（PMC4）

ポート4のポート／コントロール・モードの設定を，バイト単位で行う8ビットのレジスタです。

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

PMC4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。$\overline{RESET}$入力時，PMC4ビットはリセット(0)され，ポート・モードになります。

	7	6	5	4	3	2	1	0
PMC4	$PMC4_7$	0	0	0	0	0	0	0

$PMC4_7$	モード指定
0	ポート・モード
1	コントロール・モード

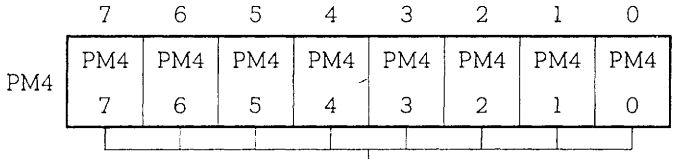
(2) ポート4モード・レジスタ（PM4）

ポート4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。

PMC4レジスタの対応ビットがリセット(0)されているとき（ポート・モードのとき）PM4レジスタの各ビットは有効になります。8/1ビット操作のメモリ・アクセスによって，このレジスタの読み出し／書き込みができます。

$\overline{RESET}$入力によって，すべてのビットがセット(1)され，入力ポートになります。

保守／廃止



PM4n	入出力指定
0	出力
1	入力

14.3.6　ポート５（P50-P52）

３ビットの汎用入出力ポートです。ポート５モード・レジスタ（PM5）の設定により，ビット単位に入力／出力の指定ができます。

ポート５モード・コントロール・レジスタ（PMC5）の設定によりパラレル・インタフェース制御信号としても機能します。

表14－６　ポート５の動作（n＝0-2）

	コントロール・モード	ポート・モード	
	PMC5n＝1	PMC5n＝0	
		PM5n＝1	PM5n＝0
P50	$\overline{\text{DATASTB}}$入出力	入力ポート	出力ポート
P51	$\overline{\text{ACK}}$入出力	入力ポート	出力ポート
P52	BUSY入出力	入力ポート	出力ポート

(1) ポート５モード・コントロール・レジスタ（PMC5）

ポート５のポート／コントロール・モードの設定を，ビット単位で行う８ビットのレジスタです。

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

PMC5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，PMC5レジスタはすべてのビットがリセット(0)され，ポート・モードになります。

保守／廃止

	7	6	5	4	3	2	1	0
PMC5	0	0	0	0	0	PMC52	PMC51	PMC50

PMC5n	モード指定
0	ポート・モード
1	コントロール・モード

(2) ポート5モード・レジスタ（PM5）

ポート5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。

PMC5レジスタの対応ビットがリセット(0)されているとき（ポート・モードのとき）PM5レジスタの各ビットは有効になります。8/1ビット操作のメモリ・アクセスによって，このレジスタの読み出し／書き込みができます。

$\overline{\text{RESET}}$入力によって，すべてのビットがセット(1)され，入力ポートになります。

	7	6	5	4	3	2	1	0
PM5	1	1	1	1	1	PM52	PM51	PM50

PM5n	入出力指定
0	出力
1	入力

14.3.7　ポート6（P60-P63）

4ビット汎用入力ポートです。A/Dコンバータのアナログ入力としても機能します。

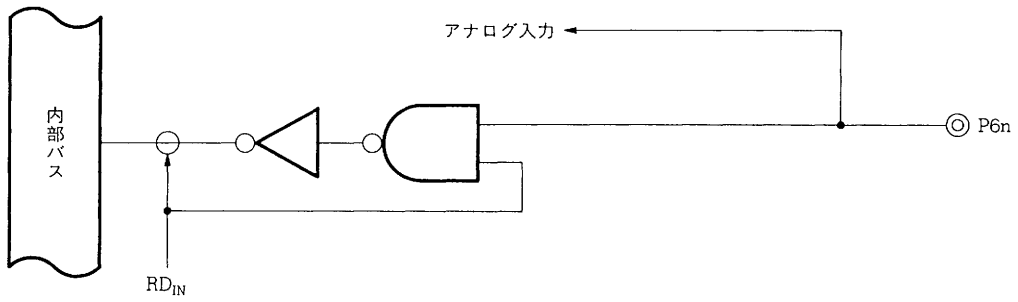
このポートはアナログ入力端子と兼用になっていますが，入力ポートとアナログ入力端子との切り替えはできません。

表14-7　ポート6の動作

	機能	
P60	ANI0入力	兼　入力専用ポート
P61	ANI1入力	兼　入力専用ポート
P62	ANI2入力	兼　入力専用ポート
P63	ANI3入力	兼　入力専用ポート

保守／廃止

図14－6　ポート６のブロック図



備考　n＝0-3

14.3.8　ポート７（P70-P77）

８ビットの汎用入出力ポートです。ポート７モード・レジスタ（PM7）の設定により，ビット単位に入力／出力の指定ができます。

ポート７モード・コントロール・レジスタ（PMC7）の設定により，リアルタイム出力ポートとしても機能します。リアルタイム出力ポートの詳細は**第10章　リアルタイム出力ポート機能**を参照してください。

表14－8　ポート７の動作（n＝0-7）

	コントロール・モード	ポート・モード	
	PMC7n＝1	PMC7n＝0	
		PM7n＝1	PM7n＝0
P70	RTP0出力	入力ポート	出力ポート
P71	RTP1出力	入力ポート	出力ポート
P72	RTP2出力	入力ポート	出力ポート
P73	RTP3出力	入力ポート	出力ポート
P74	RTP4出力	入力ポート	出力ポート
P75	RTP5出力	入力ポート	出力ポート
P76	RTP6出力	入力ポート	出力ポート
P77	RTP7出力	入力ポート	出力ポート

保守／廃止

(1) ポート７モード・コントロール・レジスタ（PMC7）

ポート７のポート／コントロール・モードの設定を，ビット単位で行う８ビットのレジスタです。

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

PMC7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，PMC7レジスタはすべてのビットがリセット(0)され，ポート・モードになります。

	7	6	5	4	3	2	1	0
PMC7	PMC7 7	PMC7 6	PMC7 5	PMC7 4	PMC7 3	PMC7 2	PMC7 1	PMC7 0

PMC7n	モード指定
0	ポート・モード
1	コントロール・モード

(2) ポート７モード・レジスタ（PM7）

ポート７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。

PMC7レジスタの対応ビットがリセット(0)されているとき（ポート・モードのとき）PM7レジスタの各ビットは有効になります。8/1ビット操作のメモリ・アクセスによって，このレジスタの読み出し／書き込みができます。

$\overline{\text{RESET}}$入力によって，すべてのビットがセット(1)され，入力ポートになります。

	7	6	5	4	3	2	1	0
PM7	PM7 7	PM7 6	PM7 5	PM7 4	PM7 3	PM7 2	PM7 1	PM7 0

PM7n	入出力指定
0	出力
1	入力

14.3.9　ポート８（P80，P81）

２ビットの汎用入出力ポートです。ポート８モード・レジスタ（PM8）の設定により，ビット単位に入力／出力の指定ができます。

ポート８モード・コントロール・レジスタ（PMC8）の設定により，DMA制御端子としても機能します。

保守／廃止

表14－9　ポート８の動作（n＝0，1）

	コントロール・モード	ポート・モード	
	PMC8n＝1	PMC8n＝0	
		PM8n＝1	PM8n＝0
P80	DMARQ0入力	入力ポート	出力ポート
P81	DMARQ1入力	入力ポート	出力ポート

(1) ポート８モード・コントロール・レジスタ（PMC8）

ポート８のポート／コントロール・モードの設定を，ビット単位で行う８ビットのレジスタです。

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

PMC8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，PMC8レジスタはすべてのビットがリセット(0)され，ポート・モードになります。

	7	6	5	4	3	2	1	0
PMC8	0	0	0	0	0	0	PMC81	PMC80

PMC8n	モード指定
0	ポート・モード
1	コントロール・モード

(2) ポート８モード・レジスタ（PM8）

ポート８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。

PMC8レジスタの対応ビットがリセット(0)されているとき（ポート・モードのとき）PM8レジスタの各ビットは有効になります。８ビット操作のメモリ・アクセスによって，このレジスタの読み出し／書き込みができます。

$\overline{\text{RESET}}$入力によって，すべてのビットがセット(1)され，入力ポートになります。

保守／廃止

	7	6	5	4	3	2	1	0
PM8	1	1	1	1	1	1	PM81	PM80

PM8n	入出力指定
0	出力
1	入力

保守／廃止

第15章　スタンバイ機能

V55PIには，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。いずれのスタンバイ状態へも専用命令で遷移することができます。

15.1　特　　徴

○ストップ状態からの復帰判定可能
○HALTモード（CPUの動作を停止）
○STOPモード（クロック発振を停止）

15.2　スタンバイ・モード

スタンバイ・モードとしてHALTモード，STOPモードがあります。各モードの動作状態を表15－1に示します。

表15－1　HALT/STOPモードの動作状態

項　　目		HALTモード	STOPモード
クロック発振回路		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	
ウォッチドッグ・タイマ			
ホールド回路			
シリアル・インタフェース			
パラレル・インタフェース			
A/Dコンバータ			
割り込み要求コントローラ			
DMAコントローラ			
★ ポート		入力：Hi-Z，出力：保持	入力：Hi-Z，出力：保持
★ $\overline{IORD}$，$\overline{IOWR}$		ハイ・レベル	ハイ・レベル
バス・ライン	AD0-AD15	DMACの動作状態に応じて変化	保　持
	A16-A23		
R/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
データ保持		CPUのステータス，RAMの内容など，内部データはすべて保持	CPUのステータス，RAMの内容など，内部データはすべて保持
解除方法		●NMI ●INTWDT ●マスカブル割り込み要求 ●$\overline{RESET}$入力	●NMI ●$\overline{RESET}$入力

保守／廃止

15.3　制御レジスタ

15.3.1　スタンバイ・コントロール・レジスタ（STBC）

ストップ状態からの復帰判定用に使用します。SBFフラグは命令によるセット(1)のみ可能です。電源投入時，電圧が0 Vから立ち上がったときのみリセット(0)されます。$\overline{\text{RESET}}$信号およびソフトウエアでは，以前の状態のままを保ち，リセット(0)されません。

このため，STOPモードに入る前にあらかじめSBFフラグをセット(1)しておけば，システムの初期化ルーチンで，電源投入によるスタートかSTOPモードからの復帰かを判別することができます。

	7	6	5	4	3	2	1	0
STBC	0	0	0	0	0	0	0	SBF

SBF	動　　作
0	電源投入時　V_{DD}=0〔V〕
1	命令による書き込み

15.4　HALTモード

15.4.1　HALTモードの設定および動作

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モードとなります。

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。各ハードウエアの状態は表15－1のようになります。

なお，DMA転送中にHALT命令を実行した場合には，1回のDMA要求に対応した転送バス・サイクルが終了するまでHALTモードへの遷移は保留されます。

15.4.2　HALTモードの解除

(1) 割り込み要求による解除

優先順位に関係なく，ノンマスカブル割り込み(NMI，INTWDT)要求，マスクされていないマスカブル割り込み要求により解除されます。

ノンマスカブル割り込みによるHALTモード解除後は，必ずベクタ・アドレスに分岐します。

割り込み処理ルーチン中でHALTモードに入った場合，処理中の割り込みとHALTモード解除ソースとなった割り込みの優先順位によって解除後の動作が異なります。

保守／廃止

(a) 優先順位が，“解除ソース割り込み＞処理中の割り込み”の場合

HALTモード解除後，ベクタ割り込みに分岐します。または，相当のレジスタ・バンクに切り替わります（ただし，EI状態の場合）。

(b) 優先順位が上記以外の場合，またはDI状態の場合

HALTモード解除後，HALT命令の次の命令から実行します。

(c) マクロ・サービスによる解除

マクロ・サービス終了後，表15－2のようになります。

表15－2　マクロ・サービス要求によるHALTモードの解除

マクロ・サービス終了後の割り込み発生条件	HALTモードの解除	HALTモード解除後の動作	
		EI状態	DI状態
成立する場合	解除する	割り込み処理ルーチンに分岐または次の命令を実行	次の命令を実行
成立しない場合	マクロ・サービス終了後再びHALTモードに戻る	—	

(2) リセットによる解除

$\overline{\text{RESET}}$入力によって解除されます。通常のリセット動作を行います。

(3) DMA要求による解除

HALTモード中にDMA要求が発生すると，内部的にはHALTモードを解除したあとにDMA転送を実行します。DMA転送実行後は再びHALTモードに戻ります。このときPCの内容は変化しません。したがって，外見上はHALTモードを解除したようには見えません。

保守／廃止

15.5　STOPモード

15.5.1　STOPモードの設定および動作

クロック発振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モードとなります。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し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。各ハードウエアの状態は表15-1のようになります。

DMA転送中にSTOP命令を実行した場合には，1回のDMA要求に対応した転送バス・サイクルが終了するまでSTOPモードへの遷移は保留されます。また，リフレッシュ・サイクルとSTOP命令の実行が競合した場合も，リフッレッシュ・サイクルの終了まで保留されます。

注意　STOP命令を実行する場合，必ずADMレジスタのCSビット＝0にしてA/Dコンバータの変換動作を停止しておいてください。

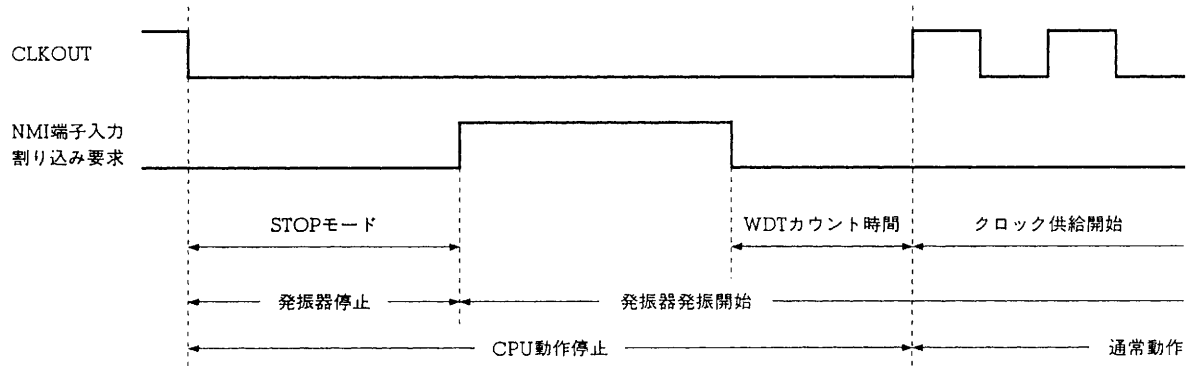
15.5.2　STOPモードの解除

(1) NMI要求による解除

「NMIのエッジ検出後のハイまたはロウ・レベル幅＋ウォッチドッグ・タイマのオーバフロー時間[注]」分だけ発振安定時間としてウエイトし，その後内部システム・クロックの供給が開始されます。

ノンマスカブル割り込みによるSTOPモード解除後は，必ずベクタ・アドレスに分岐します。

図15-1　NMI端子入力によるSTOPモードの解除動作



注　STOPモード解除時のウォッチドッグ・タイマのオーバフロー時間は，システム・クロックが12.5 MHzのとき10.4 ms，16 MHzのとき8.1 msとなります。

このときウォッチドッグ・タイマ割り込み（INTWDT）は発生しません。

保守／廃止

(2) リセットによる解除

$\overline{\text{RESET}}$入力により解除されます。$\overline{\text{RESET}}$端子にロウ・レベル信号が入力されるとクロック発振を再開します。発振が安定するまでロウ・レベルを保持する必要があります。

〔メ　モ〕

保守／廃止

第16章　クロック発生機能

クロック発生回路は，CPUおよび周辺ハードウエアに供給する各種クロックを発生し，これらの動作タイミングを制御します。

16.1　特　　徴

○クロック発生回路内蔵

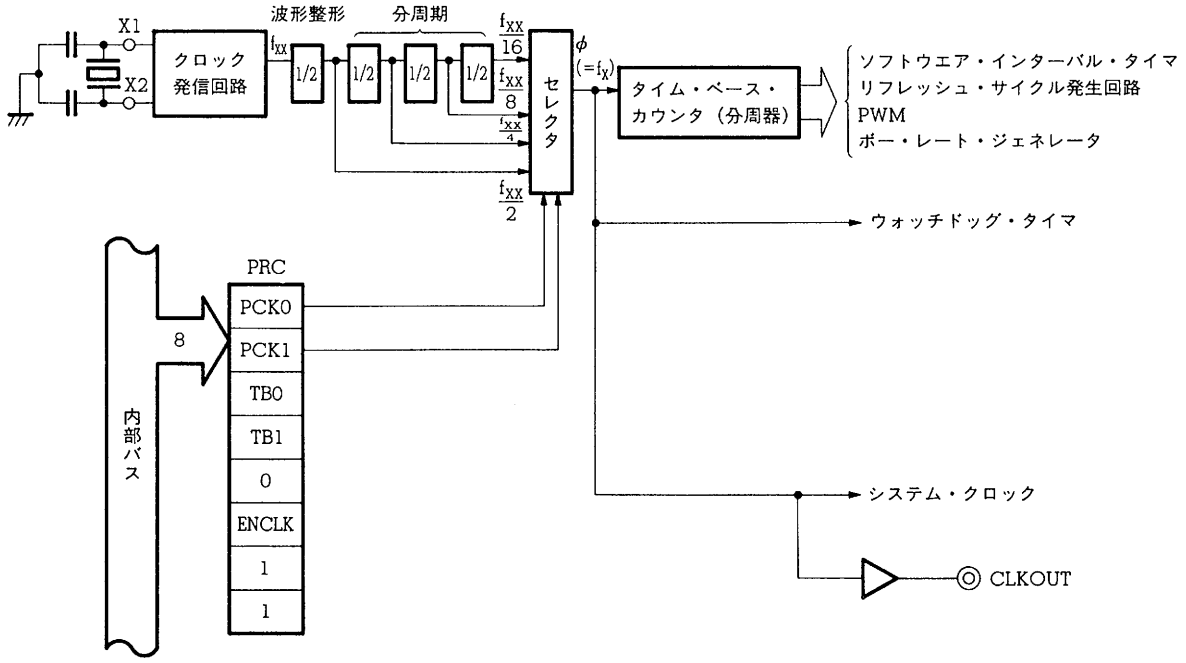
○外部クロック入力（X1入力）

○クリスタル／セラミック発振（X1，X2入力）

16.2　構　　成

クロック発生回路は図16－1のように構成されています。

図16－1　クロック発生回路



f_{XX}　：発振周波数

ϕ　：システム・クロック

PRC：プロセッサ・コントロール・レジスタ

保守／廃止

16.3 動　　作

16.3.1 クロック発振回路

クロック発振回路では，X1，X2端子に接続されたクリスタル振動子またはセラミック発振子によって基準クロックを発生します。クロック発振回路の出力は，波形整形（1/2分周）され，プロセッサ・コントロール・レジスタ（PRC）により分周比を選択してシステム・クロックφとして使用されます。

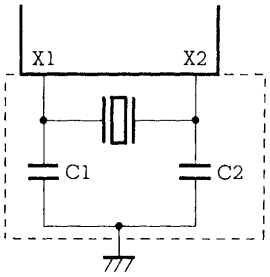
システム・クロックφの分周比はPRCレジスタのPCK1とPCK0ビットの指定により，発振周波数（f_{XX}）の1/2，1/4，1/8，1/16を選択することができます。

システム・クロックφを低速にすることによって，バッテリ駆動システムで電圧が低下してもより長時間の動作が可能となります。

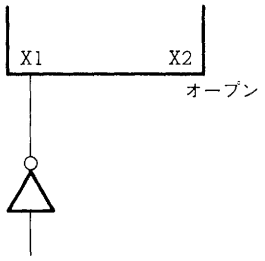
また，外部クロックを入力することもできます。その場合X1端子にクロック信号を入力し，X2端子はオープンにしてください。

図16－2　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



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．破線の範囲にほかの信号線を通さないでください。

16.3.2 タイム・ベース・カウンタ（TBC）

V55PIは，内部システム・クロックφを分周する分周器（タイム・ベース・カウンタTBC）を各タイマ・ユニットで共有しています。

TBCは命令による読み出し，書き込みともできません。

TBCのタップ出力（2^n分周クロック）は下記ユニットへカウント・クロックとして供給されています。

(1) リフレッシュ・サイクル発生回路

(2) ソフトウエア・インターバル・タイマ

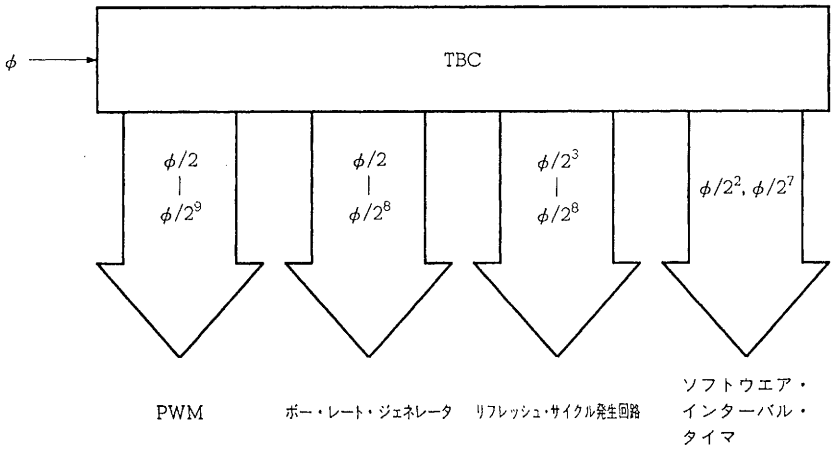
(3) PWMユニット

(4) ボー・レート・ジェネレータ

保守／廃止

TBCはリセット入力によってのみ00Hにクリアされ，その後は常にインクリメントされます。TBCはSTOPモードでは動作を停止します。TBCの構成を図16－3に示します。

図16－3　分周器（タイム・ベース・カウンタTBC）の構成



保守／廃止

16.4　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，ソフトウエア・インターバル・タイマの入力クロックの指定，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ビットのレジスタです。

リセット後PRCレジスタの値は，EEHになります。

なお，ビット４は“0”に，ビット６，７は“1”に必ず設定してください。

	7	6	5	4	3	2	1	0
PRC	1	1	ENCLK	0	TB1	TB0	PCK1	PCK0

システム・クロックφの分周比指定

PCK1	PCK0	指　定
0	0	φ＝発振周波数（f_{XX}）×（1/2）
0	1	φ＝発振周波数（f_{XX}）×（1/4）
1	0	φ＝発振周波数（f_{XX}）×（1/8）
1	1	φ＝発振周波数（f_{XX}）×（1/16）

ソフトウエア・インターバル・タイマの入力クロックの指定

TB1	TB0	入力クロック周波数	最大インターバル時間	
			φ＝12.5 MHz時	φ＝16 MHz時
0	0	$\phi/2^2$	21.0 ms	16.4 ms
0	1	$\phi/2^7$	671 ms	524.2 ms
1	0	動作停止	∞	∞
1	1			

CLKOUT端子からのシステム・クロック出力制御指定

ENCLK	指　定
0	出力禁止（ロウ・レベルで停止）
1	出力許可

（スタンバイ・モードにおけるCLKOUT端子出力レベル）

HALTモード：出力継続

STOPモード：出力停止（直前レベル保持）

備考　▨ はリセット後の状態を示します。

保守／廃止

PCK1　**PCK0**　**システム・クロックの分周比指定**

発振器の出力をこのビットで指定した値で分周後，システム・クロックφとして使用します。

外付けクリスタルの発振周波数f_X＝32.0 MHzの場合，リセット直後には，システム・クロックφは，次のような値になります。

$\phi = (f_X) \times 1/8 = 4.0$ MHz

V55PIの性能を引き出すためには，PCK1＝PCK0＝0に設定してφ＝16 MHzで動作させるようにしてください。

注意　このビットを書き換えることにより，システム・クロックから生成されるクロックすべての周波数が変化します。

TB1　**TB0**　**ソフトウエア・インターバル・タイマの入力カウント・クロック周波数の指定および動作制御**

TB1ビットにより，ソフトウエア・インターバル・タイマの停止／動作制御を行うことができます。TB1ビットを1→0に設定してからカウント動作を再開するとき，ソフトウエア・タイマ・カウンタ（STC）は0クリアされたあと，インクリメントを開始します。

保守／廃止

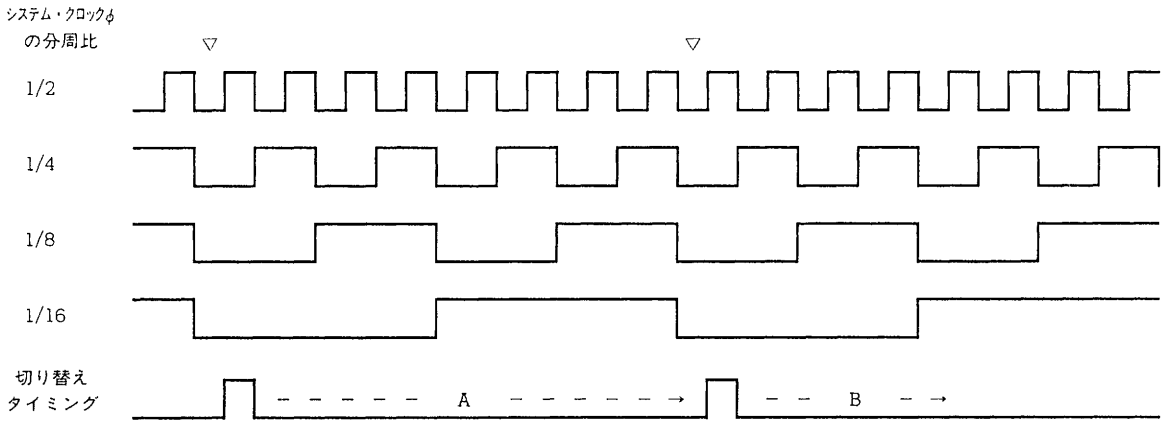
16.5　クロック切り替えタイミング

システム・クロックの切り替えは，PRCレジスタの書き換え直後には行われず，一定のマシン・サイクルの間は切り替え前のクロックで動作します。

したがって，システム・クロック切り替え後にスタンバイ命令を実行する場合，およびタイマ，ボー・レート・ジェネレータ，リフレッシュ回路などの動作クロックとして，切り替え後のシステム・クロックを用いる必要がある場合には，この切り替え時間経過後，スタンバイ命令の実行，または該当回路を動作許可状態とする必要があります。

次に切り替えタイミングについての一例を示します。

図16－4　システム・クロックの切り替えタイミング



1/2，1/4，1/8，1/16のすべてのクロックがロウ・レベルとなった次のクロックからシステム・クロックの周波数が切り替わります。

したがって，Aの期間内（▽-▽間）にMOV命令などによるPRCレジスタのPCK1，PCK0ビット書き込みサイクル(内部バスの書き込みサイクル)が実行されると，Bのタイミングからは新しいシステム・クロック周波数で動作します。

保守／廃止

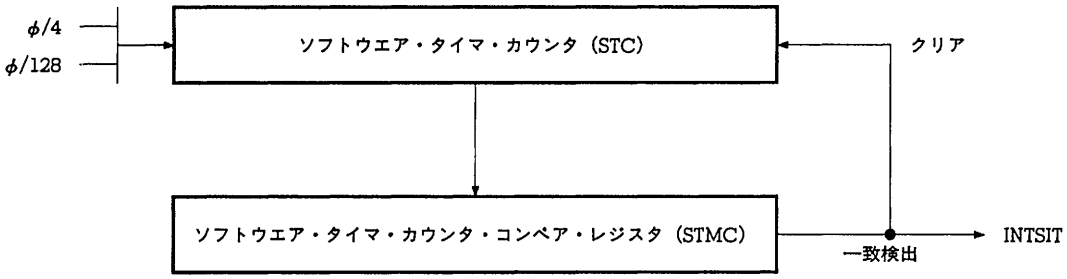
第17章　ソフトウエア・インターバル・タイマ機能

V55PIは，ソフトウエア・タイマ機能用および時計機能用のタイマとして，16ビット・ソフトウエア・インターバル・タイマを内蔵しています。

17.1　構　　成

ソフトウエア・インターバル・タイマの構成は図17－1のとおりです。

図17－1　ソフトウエア・インターバル・タイマの構成



17.2　制御レジスタ

(1) プロセッサ・コントロール・レジスタ（PRC）

ソフトウエア・インターバル・タイマへの入力クロック（カウント・クロック）は，プロセッサ・コントロール・レジスタ（PRC）のTB0ビットによって選択することができます。また，PRCレジスタのTB1ビットにより，タイマの停止／動作を制御することができます。

	7	6	5	4	3	2	1	0
PRC	1	1	ENCLK	0	TB1	TB0	PCK1	PCK0

ソフトウエア・インターバル・タイマの入力クロック指定

TB1	TB0	入力クロック周波数	最大インターバル時間	
			ϕ＝12.5 MHz時	ϕ＝16 MHz時
0	0	$\phi/2^2$	21.0 ms	16.4 ms
0	1	$\phi/2^7$	671 ms	524.2 ms
1	0	動作停止	∞	∞
1	1			

備考　▒▒ はリセット後の状態を示します。

保守／廃止

(2) ソフトウエア・タイマ・カウンタ（STC）

PRCレジスタにより指定された入力クロックのタイミングで，カウント・アップを行う16ビットのカウンタです。ソフトウエア・タイマ・カウンタ・コンペア・レジスタの値と比較を行い，一致を検出すると割り込み要求が発生します。一致検出後，STCの内容はクリア(0)されます。

16ビット操作命令による読み出しのみ可能です。

(3) ソフトウエア・タイマ・カウンタ・コンペア・レジスタ（STMC）

ソフトウエア・インターバル・タイマから発生する割り込み要求のインターバル時間は，16ビットのソフトウエア・タイマ・カウンタ・コンペア・レジスタ（STMC）の設定値によって指定します。

STMCレジスタのリセット後の値は，FFFFHとなります。

また，STMCレジスタは16ビット操作命令による読み出し／書き込みが可能です。

17.3　割り込み要求

ソフトウエア・タイマ・カウンタ（STC）は，PRCレジスタで指定された入力クロックでカウント・アップされ，ソフトウエア・タイマ・カウンタ・コンペア・レジスタ（STMC）の設定値と一致した時点でインターバル割り込み（INTSIT）を発生します。

インターバル割り込みの発生周期は次のようになります。

発生周期＝STMCレジスタ設定値/入力クロック周波数

STMCレジスタに“0”を設定すると入力クロックの選択内容によらず，2クロックごとに割り込みが発生します。

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。

INTSIT割り込みは，割り込み要求制御レジスタIC32でマスク可能です。

保守／廃止

第18章　コーデック機能

V55PIは，９種類のコーデック命令を持っています。

V55PIではこれらの命令を用いることで，２値画情報のMH[注1]符号化だけでなく，従来ACEE（アドバンスト・コンプレッション・エクスパンション・エンジン）などの専用デバイスを用いる必要のあったMR[注2]符号化を，小規模かつ高速なソフトウエア・コーデックにより実現できます。

注1. MH：Modified Huffman

2. MR：Modified Relative element designate

18.1 特　　徴

V55PIはコーデック命令として次に示す９命令（圧縮系：４，伸長系：５）を持っています。

○圧縮系命令

(1) 変化点テーブル作成命令：COLTRP

(2) データ送出命令（EOL[注1]，FILL，RTC[注2]などを送出）：ALBIT

(3) MH符号化命令：MHENC

(4) MR符号化命令：MRENC

○伸長系命令

(5) EOL検出命令：SCHEOL

(6) １ビット（タグ）検出命令：GETBIT

(7) MH復号化変化点テーブル作成命令：MHDEC

(8) MR復号化変化点テーブル作成命令：MRDEC

(9) 画素データ作成命令：CNVTRP

このような命令を使用し，MH，MR符号化は図18－１のように，MH，MR復号化は図18－２のように行われます。

注1. EOL：End of Line

2. RTC：Return To Control

注意　V55PIのコーデック命令を用いて圧縮／伸長処理を行う場合，次のことを前提に指定してください。

保守／廃止

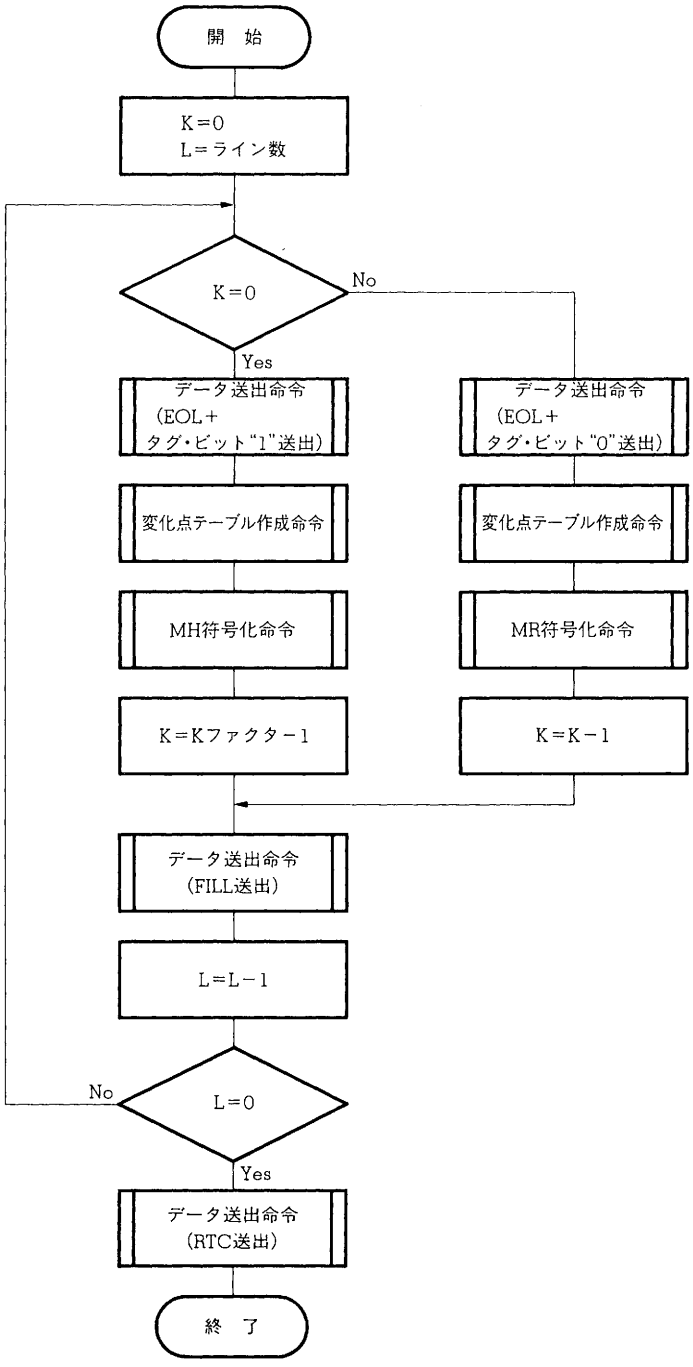
● 1ライン単位ごとに圧縮／伸長を行う。

●圧縮処理中にタスク切り替えおよび割り込みが発生することを考慮する。

● 1ライン処理ビット数が1ページ処理中変わらないようにする。

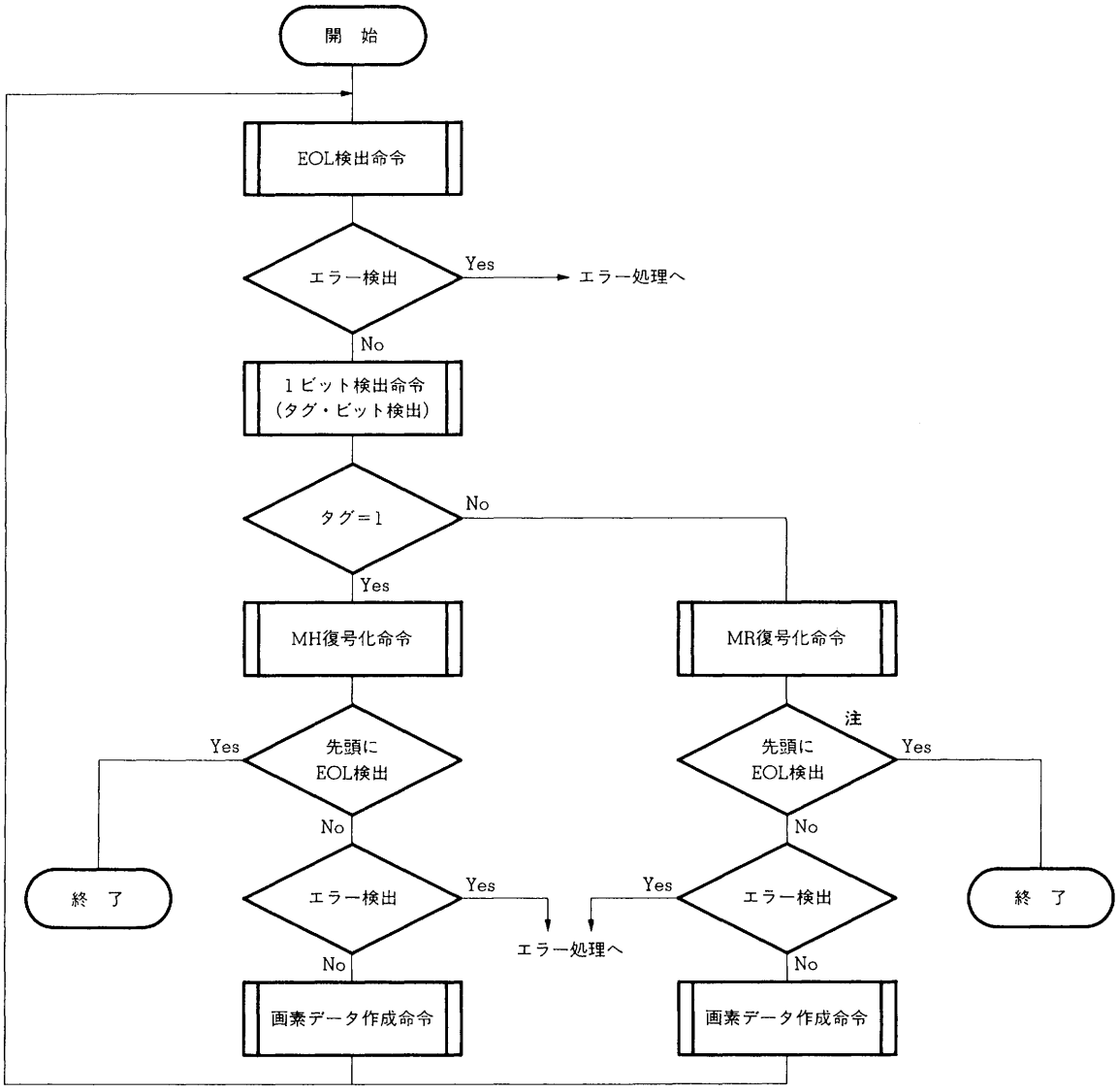
●セグメントをまたぐ64 Kバイト以上のデータに対しては，途中でセグメント値を変更する。

図18－1　MH，MR符号化処理フロー



保守／廃止

図18－2　MH，MR復号化処理フロー



注　RTCは2個のEOLで検出しています。

保守／廃止

18.2　2値画情報の符号化／復号化方式

ここでは，2値画情報の符号化(圧縮)方式と復号化(伸長)方式について説明します。

2値画情報の符号化(圧縮)方式としては，ITU-T（国際電気通信連合）勧告に基づいて，MH方式，MR方式とMMR[注]方式が標準的に採用されています。復号化（伸長）は符号化と逆の手順で実現します。

ITU-T勧告についての詳細は「ITU-T　ホワイト・ブック　G3/G4ファクシミリ関連Tシリーズ勧告」（財団法人　新日本ITU協会刊）を参照してください。

注　MMR：Modified MR

各方式の説明に入る前に，その基本的な考え方であるラン・レングス符号化について説明します。

18.2.1　ラン・レングス符号化

いま，図18－3のような絵をスキャナなどで2値画情報として取り込んだ場合，メモリ上には図18－4のようなビット・マップ・データとして取り込まれます。つまり，白の画素は0で，黒の画素は1で表されます。

保守／廃止

図18－3　サンプル図

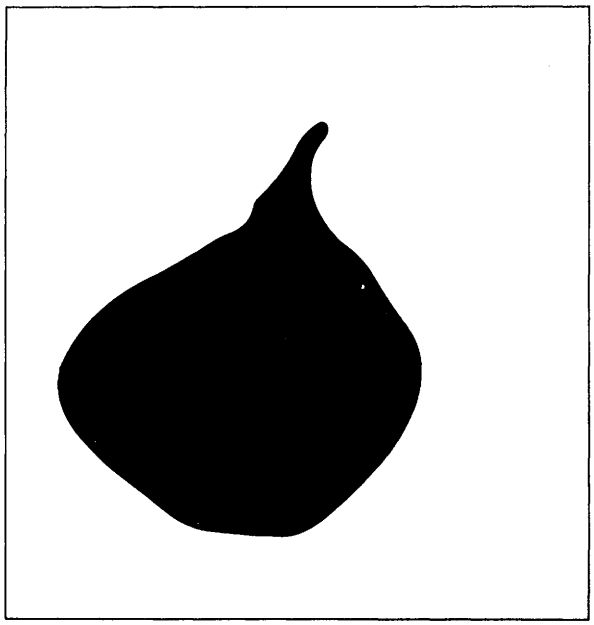
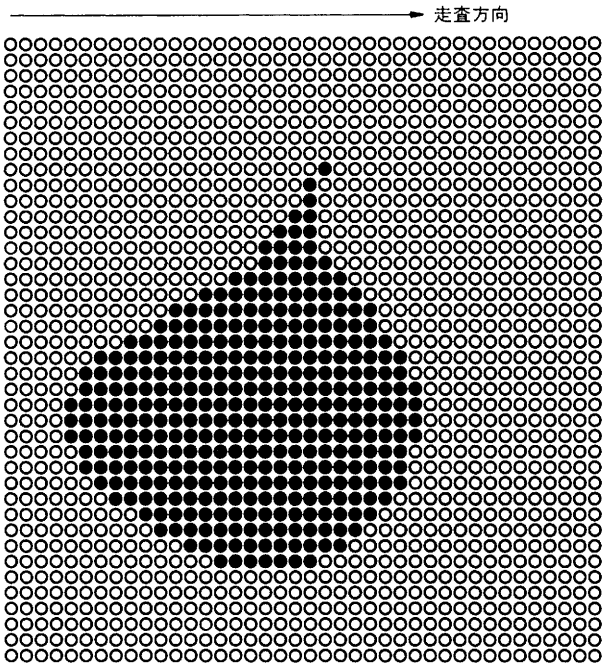


図18－4　ビット・マップ・データ



備考　○：ビット０

●：ビット１

保守／廃止

これを図18－4の矢印の方向に走査していくと，初めは0のビットが連続していますが，あるところで0→1へ変化する点（変化点）にぶつかります。さらに，走査を進めると今度は1→0への変化点にぶつかります。ラン・レングス符号化とは，ひとつの変化点から次の変化点までの連続する同じ色（白か黒）の画素数（これをラン・レングスといいます）を何らかの方法で符号に置き換えるものです。

たとえば，符号としてラン・レングスそのものを使えば，図18－4のビット・マップ・データは図18－5のような数値列に置き換わります。元のデータ量は40×40＝1600ビット＝200バイトですが，符号化後は数十バイトに圧縮できます。スキャナの解像度を上げれば，この差はさらに大きくなります。

図18－5　ラン・レングス符号列

341 1 38 1 39 1 38 2 37 3 36 4 36 5 37 8 30 11 27 13 25 15 23 18 20 21 18
22 18 23 16 24 16 24 16 24 17 22 18 22 20 19 23 16 25 14 28 11 31 7 259

符号を復号して元の画像を再生するには，走査の開始点の色と画像の横サイズが分かれば符号化の逆の手順によって容易に実現できます。しかも，この再生画像は元の画像と完全に同じものです。

以上がラン・レングス符号化の考え方です。次項より説明するITU-T勧告の符号化方式は，このラン・レングス符号化が基礎となっています。

保守／廃止

18.2.2　MH方式

MH方式は一次元符号化方式とも呼ばれます。

この方式では，ラン・レングスを画像データの各水平ラインごとに計数します。各ラン・レングスに対する符号としては，表18－1のものを採用しています。符号にはメークアップ符号とターミネート符号の２種類があります。64個未満のラン・レングスに対してはターミネート符号が割り当てられ，64個以上のラン・レングスについてはメークアップ符号とターミネート符号の組み合わせが割り当てられます（**例１，例２**　参照）。

例1．ラン・レングスが231の場合

231＝64×3＋39＝192＋39

192：メークアップ符号
39：ターミネート符号

2．ラン・レングスが128の場合

128＝64×2＋0＝128＋0

128：メークアップ符号
0：ターミネート符号

保守／廃止

表18－1　ラン・レングス符号

(a)　ターミネート符号

ラン・レングス	白ラン用符号語	黒ラン用符号語	ラン・レングス	白ラン用符号語	黒ラン用符号語
0	00110101	0000110111	32	00011011	0000011010100
1	000111	010	33	00010010	0000011010101
2	0111	11	34	00010011	0000110100010
3	1000	10	35	00010100	0000110100011
4	1011	011	36	00010101	0000110101000
5	1100	0011	37	00010110	0000110101001
6	1110	0010	38	00010111	0000110101010
7	1111	00011	39	00101000	0000110101011
8	10011	000101	40	00101001	0000011011000
9	10100	000100	41	00101010	0000011011001
10	00111	0000100	42	00101011	0000110110100
11	01000	0000101	43	00101100	0000110110101
12	001000	0000111	44	00101101	0000010101000
13	000011	00000100	45	00000100	0000010101001
14	110100	00000111	46	00000101	0000010101010
15	110101	000011000	47	00001010	0000010101011
16	101010	0000010111	48	00001011	0000011001000
17	101011	0000011000	49	01010010	0000011001001
18	0100111	0000001000	50	01010011	0000010100010
19	0001100	00001100111	51	01010100	0000010100011
20	0001000	00001101000	52	01010101	0000001001000
21	0010111	00001101100	53	00100100	0000001101111
22	0000011	00000110111	54	00100101	0000001110000
23	0000100	00000101000	55	01011000	0000001001111
24	0101000	00000010111	56	01011001	0000001010000
25	0101011	00000011000	57	01011010	0000010110000
26	0010011	0000110010100	58	01011011	0000010110001
27	0100100	0000110010111	59	01001010	0000001010111
28	0011000	0000110011000	60	01001011	0000001011000
29	00000010	0000110011101	61	00110010	0000010110100
30	00000011	0000011010000	62	00110011	0000011001100
31	00011010	0000011010011	63	00110100	0000011001111

(b)　メークアップ符号 (1/2)

ラン・レングス	白ラン用符号語	黒ラン用符号語	ラン・レングス	白ラン用符号語	黒ラン用符号語
64	11011	0000001111	960	011010100	0000001110011
128	10010	000011001000	1024	011010101	0000001110100
192	010111	000011001001	1088	011010110	0000001110101
256	0110111	000001011011	1152	011010111	0000001110110
320	00110110	000000110011	1216	011011000	0000001110111
384	00110111	000000110100	1280	011011001	0000001010010
448	01100100	000000110101	1344	011011010	0000001010011
512	01100101	0000001101100	1408	011011011	0000001010100
576	01101000	0000001101101	1472	010011000	0000001010101
640	01100111	0000001001010	1536	010011001	0000001011010
704	011001100	0000001001011	1600	010011010	0000001011011
768	011001101	0000001001100	1664	011000	0000001100100
832	011010010	0000001001101	1728	010011011	0000001100101
896	011010011	0000001110010	EOL	000000000001	000000000001

保守／廃止

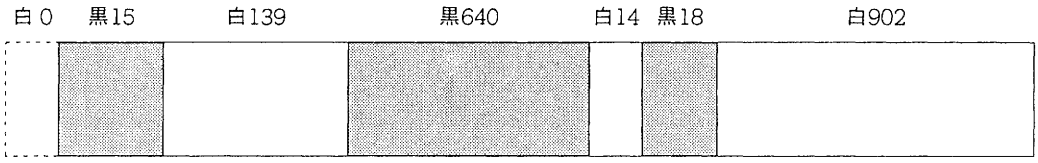
(b)　メークアップ符号 (2/2)

ラン・レングス	符号語（白ラン，黒ラン共通）	ラン・レングス	符号語（白ラン，黒ラン共通）
1792	00000001000	2240	000000010110
1856	00000001100	2304	000000010111
1920	00000001101	2368	000000011100
1984	000000010010	2432	000000011101
2048	000000010011	2496	000000011110
2112	000000010100	2560	000000011111
2176	000000010101		

図18－6に1ライン分の符号化の例を示します。ラインの先頭画素が黒で始まる場合には，その直前に0個の仮想的な白画素があるものとして符号化します。

図18－6　1ライン分の符号化例

画像データ



計 1728ビット

符号データ

白0	黒15	白128	白11	黒640	黒0	白14	黒18	白896	白6
00110101	000011000	10010	01000	0000001001010	0000110111	110100	0000001000	011010011	1110

計 79ビット

1ページ分の符号化は次の手順で行います。

① 最初に符号表中のEOL（End Of Line）符号を出力します。

② 1ラインを符号化します。

③ 必要ならば，最小伝送時間[注]を保証するために複数個の0から成るフィル・ビットを付加します。

④ EOL符号を付加します。

⑤ ①から④を繰り返します。

⑥ 1ページの最終ラインの符号（またはそれに続くフィル・ビット）の後ろにRTC（Return To Control）符号を付加してページの終端を示します。ただし，

RTC＝EOL×6

とします。

保守／廃止

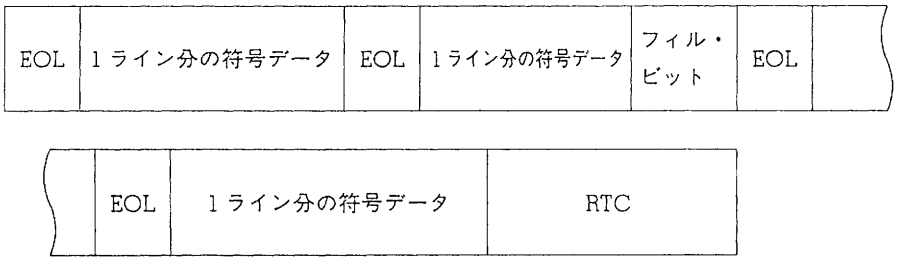
注　最小伝送時間

ファクシミリなどにおいては，受信側の受信許容速度がまちまちです。受信許容速度は主として，受信バッファの大きさやプリント・アウトの速度に依存して決まります。符号データの量は画像データのパターンに依存し，送信速度は符号データの量に依存します。したがって，この受信許容速度が遅いにもかかわらず短いライン符号が次々と送られてきた場合には，1ライン分の画像を出力し終わらないうちに次のライン符号を受信しなければならなくなり，データの取りこぼしが生じることになります。

このため，1ライン分の符号データを受信できる最小の時間（最小伝送時間）が機種ごとに決まっています。送信側が1ライン分の符号データを送信する場合に，この時間を守れないときはフィル・ビットを挿入して調整してください。また，送り先の最小伝送時間は符号データの送信を開始する前に，それぞれの機器間で連絡をとることになっています。

この手順で符号化することにより，1ページ分の画像データは図18-7に示すようになります。

図18-7　1ページ分のMH符号



備考　RTC＝EOL×6

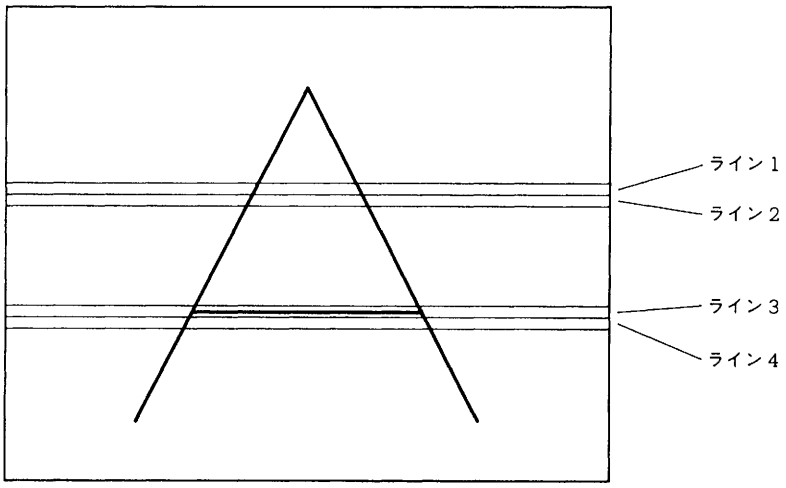
保守／廃止

18.2.3　MR方式

MR方式は，画像データの垂直方向の相関に着目して，符号化するラインを１つ手前のラインと比較しながら符号化を進める方式です。

文字や図形などが描かれた文書画像では基本的に縦，横，斜めの線で画像が構成されています。したがって，図18－８のライン１とライン２のように，あるラインについて特定の画像パターンが現れたならば，その上下のラインについても同じようなパターンが現れる頻度が高いと言えます。同様に，図18－８のライン３とライン４のように，上下のラインでまったく違う画像パターンが現れる頻度も高いと言えます。そこで，このような２つの場合に対して短い符号を割り当てることによって，MH方式以上の圧縮率が期待できます。

図18－８　画像の垂直方向の相関



この方式の説明をする前に説明で使用する用語と記号の定義をします。

符号化するラインの１つ手前のラインを参照ラインと呼びます。符号化するライン上と参照ライン上の変化点に対して，図18－９のようにa1，a2，b1，b2のレーベルを付けます。この４個の画素の位置関係から３つのモードを考えて符号化します。a0は基準点と呼ばれる走査を開始する画素です。5個の画素は次のように定義されています。

① a0：各ラインを符号化する際に，ラインの最初の画素の直前に仮想的な白画素があるものとして，ここに置かれます。a0を開始点として１ラインのある範囲の符号化が終了すると，符号化された領域の後ろに新たにa0の位置が決定されます。

② a1：符号化するライン上で，a0の次に現れる変化点に置かれます。

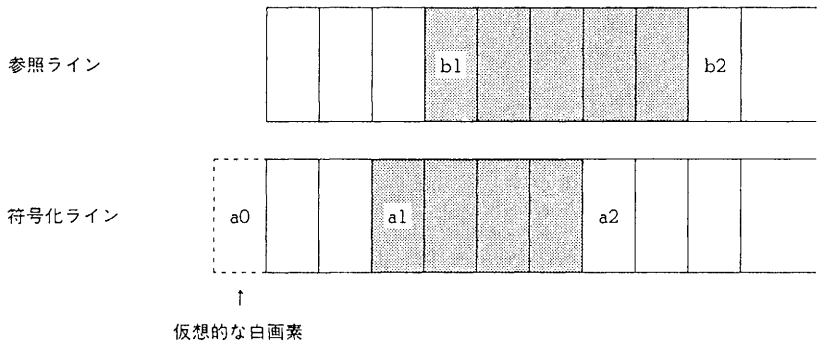
③ a2：符号化するライン上で，a1の次に現れる変化点に置かれます。

④ b1：参照ライン上で，a0の右に現れる，a0の画素の色と反対色の最初の画素に置かれます。

⑤ b2：参照ライン上で，b1の次に現れる変化点に置かれます。

保守／廃止

図18－9　符号化の基準となる点



符号化のモードとしては，次の３つがあります。

(1)　パス・モード

(2)　垂直モード

(3)　水平モード

それぞれのモードの詳細は次のとおりです。

(1)　パス・モード

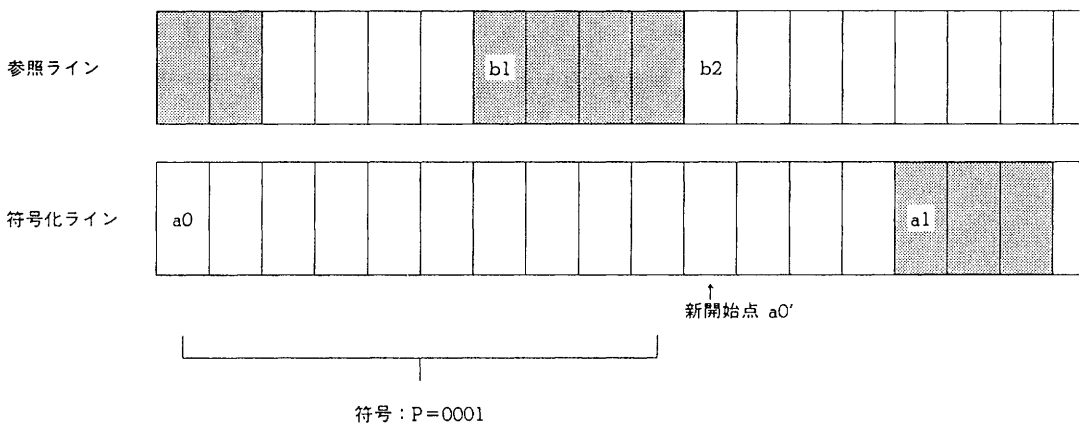
パス・モードは，b2がa1の左側にある場合に対応します。このことは，符号化するライン上でのa0の次の変化点が，b2に対応する位置までは存在しないことを意味します(図18－10)。図18－８のライン３とライン４のような組み合わせで起こります。

具体的な符号化は，次の手順で行います。

①　符号としてパス・モード符号P（＝0001）を出力します。

②　b2に対応する位置に新基準点a0'を設定します。a0'は以後の符号化において，a0となります。

図18－10　パス・モードの例



保守／廃止

(2) 垂直モード

垂直モードは，a1がb1の左右計７画素以内の範囲にある場合に対応します(図18－11)。図18－８のライン１とライン２のような組み合わせで起こります。

具体的な符号化は，次の手順で行います。

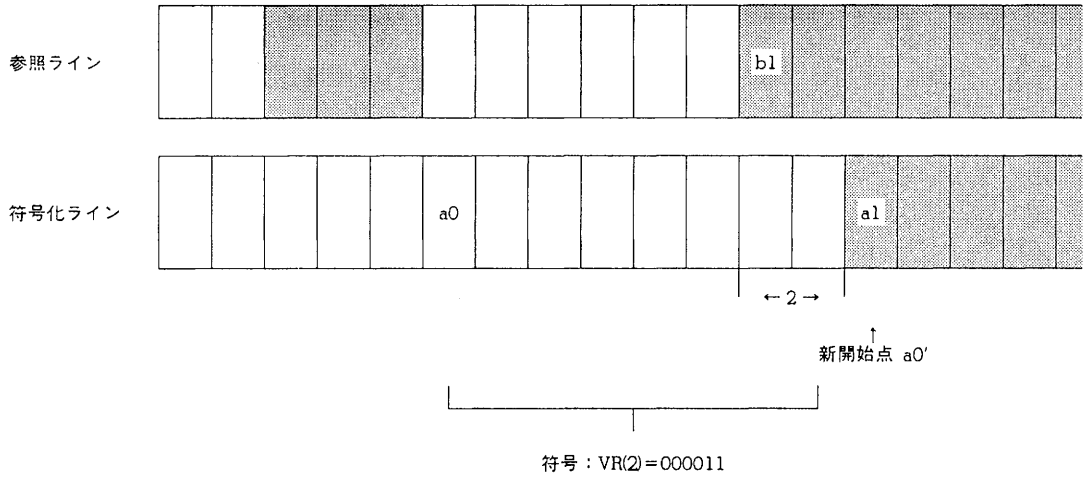
① 符号として，b1に対するa1の相対的な位置（右側か左側）および距離（画素数）に着目して表に従った符号を出力します（表18－２)。

② a1の位置に新基準点a0'を設定します。a0'は以後の符号化において，a0となります。

表18－２　垂直モードの符号

種類	a1とb1との位置関係	符号
V(0)	a1がb1の真下にある	1
VL(1)	a1がb1から１画素分左にある	010
VL(2)	a1がb1から２画素分左にある	000010
VL(3)	a1がb1から３画素分左にある	0000010
VR(1)	a1がb1から１画素分右にある	011
VR(2)	a1がb1から２画素分右にある	000011
VR(3)	a1がb1から３画素分右にある	0000011

図18－11　垂直モードの例（a1がb1の右側２画素目にある場合）



保守／廃止

(3) 水平モード

パス・モードまたは垂直モード以外の場合，つまりb2がa1の右側にあり，かつa1とb1の距離が3画素を越える場合に対応します（図18－12）。

具体的な符号化は，次の手順で行います。

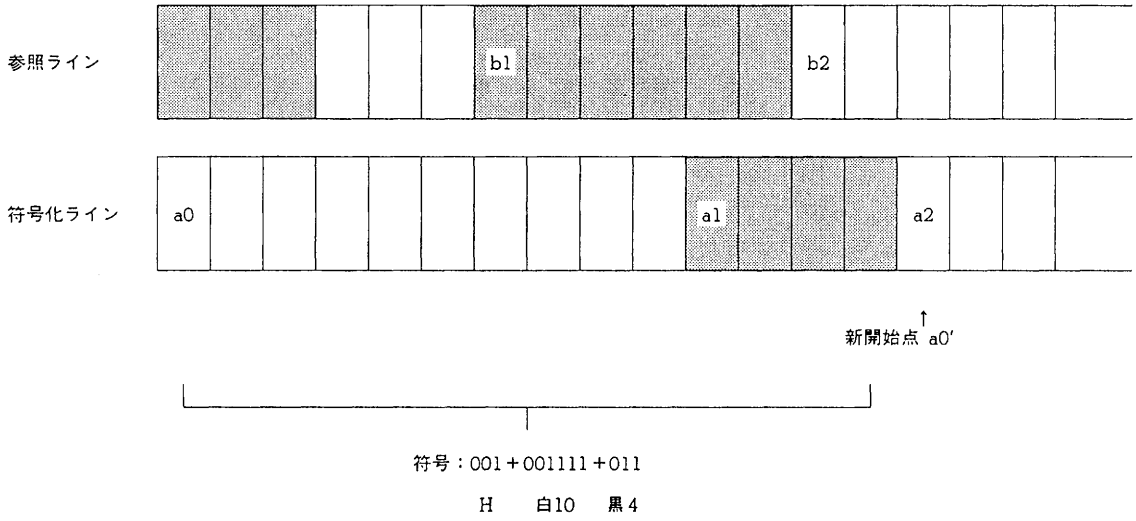
① 符号として，まず水平モード符号H（＝001）を出力します。

② a0からa1までのラン・レングスをMH方式の符号表（表18－1）に従って符号化します。

③ a1からa2までのラン・レングスをMH方式の符号表（表18－1）に従って符号化します。

④ a2の位置に新基準点a0′を設定します。a0′は以後の符号化において，a0となります。

図18－12　水平モードの例

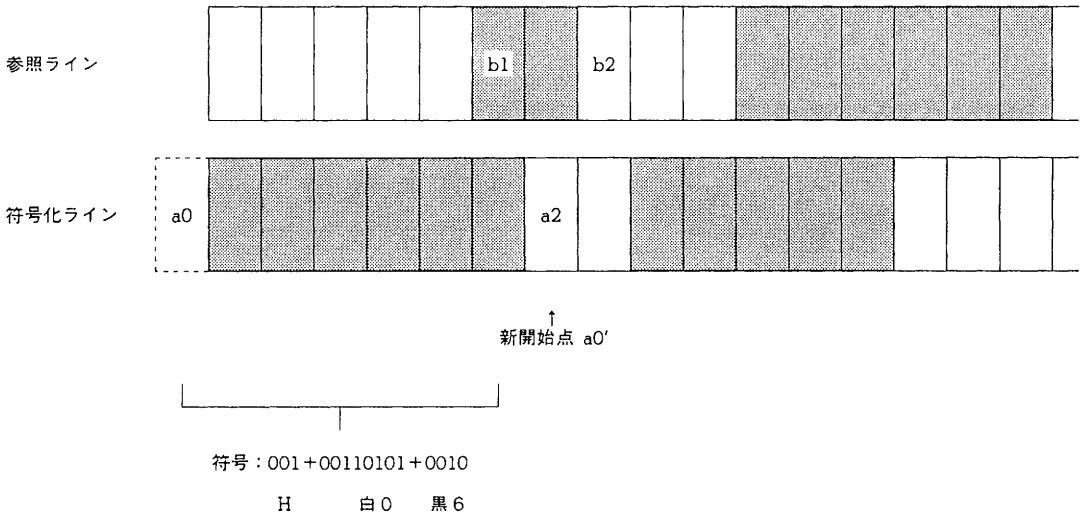


保守／廃止

各ラインの始端と終端ではa0-a2，b1-b2の関係が特殊なものとなるため特別な処理が必要です。

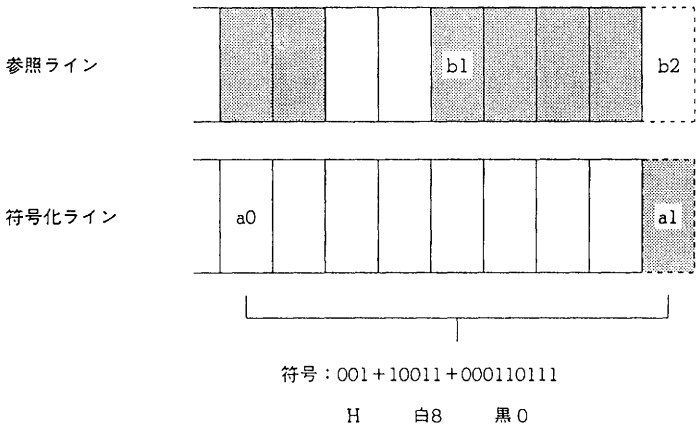
ライン始端では，a0はラインの最初の画素の直前に０個の仮想的な白画素があるものとして，ここに置かれます。したがって，図18－13のようにラインが黒画素で始まっているような場合，符号はHに続いてラン・レングス０の符号が来ることになります。

図18－13　ライン始端の処理例



ライン終端では，最後の画素の直後にそれと反対色の仮想的な画素が０個続くものとして，仮想画素までを符号化します（図18－14）。

図18－14　ライン終端の処理例



以上の手順を二次元符号化方式と呼びます。

１ページ分の符号化はKというパラメータを使って一次元符号化と二次元符号化の組み合わせで行います（図18－15）。

保守／廃止

具体的な符号化は，次の手順で行います。

① 第１ライン目を一次元符号化します。

② 第２ライン目以降を二次元符号化します。

③ K＋１ライン目を一次元符号化します。

④ ①から③の手順を繰り返します。

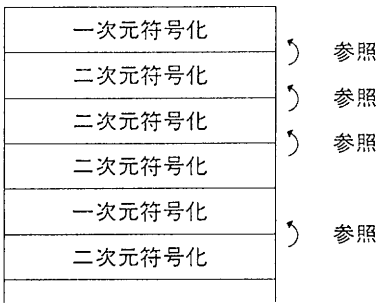
つまり，Kラインごとに一次元符号化方式で符号化します。これは次の理由によります。二次元符号化方式は直前のラインを参照して符号化します。このため，あるラインの符号が伝送誤りなどで正常な符号でなくなった場合，このライン以降のすべての復元画像が正常なものでなくなります。これを防ぐために，Kラインごとに一次元符号化を挿入します。

Kの値は，垂直方向の解像度により，一般的に次のようになります。

・100LPI（Lines Per Inch）の場合，K＝2

・200LPI（Lines Per Inch）の場合，K＝4

図18－15　Kパラメータ（K＝4の場合）



一次元符号化したラインと二次元符号化したラインとを区別するため，ライン符号の区切り符号を付加します。

・一次元符号の直前：EOL＋1

・二次元符号の直前：EOL＋0

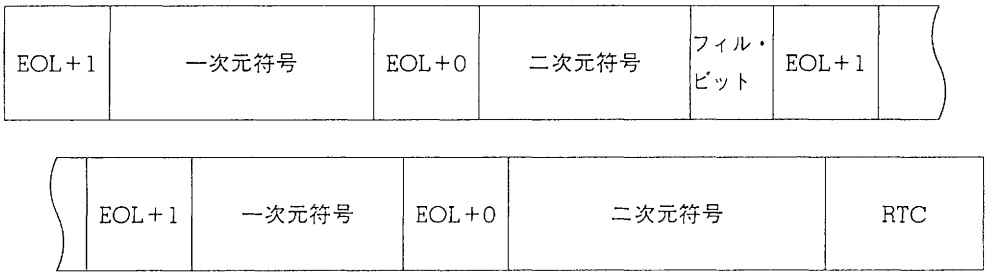
また，ページ終端には次の符号を付加します。

RTC＝(EOL＋1)×6

この手順で符号化することにより，１ページ分の画像データは図18－16に示すようになります。

保守／廃止

図18－16　1ページ分のMR符号



備考　RTC＝(EOL＋1)×6

保守／廃止

18.2.4　MMR方式

MMR方式では，すべてのラインを二次元符号化します。ページの第１ライン目はその直前に仮想的な全白のラインがあるものとして，二次元符号化されます。

MR方式では，符号データを通常の電話回線で伝送することを前提として規定されているため，伝送誤りの伝播を制限するためにKパラメータが用いられています。これに対して，MMR方式はデジタル回線を用いた誤りのない伝送を前提としているため，Kパラメータを無限大に設定しています。

１ページ分の符号化は次の手順で行います。

① ページの第１ライン目はその直前に仮想的な全白のラインがあるものとして，第１ライン目からすべてのラインを二次元符号化します。各ライン符号間を区切る特別な符号は用いません。

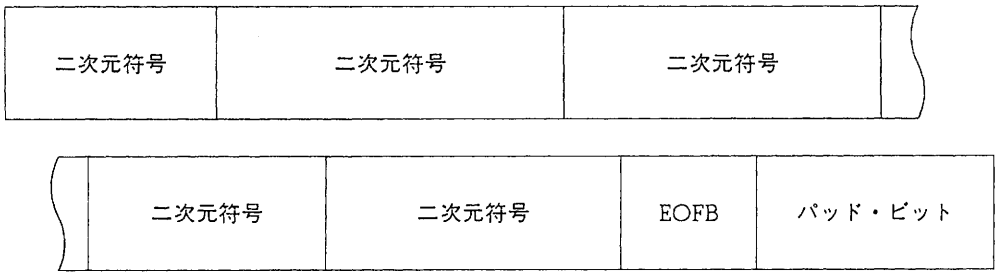
② ページ終端を示す符号として，最終ライン符号の直後にEOFB（End Of Facsimile Block）符号を付加します。ただし，EOFB符号は次の値とします。

EOFB＝EOL×2

③ １ページ分の符号データ量をバイト単位にそろえる必要がある場合，またはある大きさ以上にそろえる必要がある場合には，EOFB符号の後ろにパッド・ビットとして０のビット列を付加します。

この手順で符号化することにより，１ページ分の画像データは図18－17に示すようになります。

図18－17　１ページ分のMMR符号



備考　EOFB＝EOL×2

保守／廃止

18.3　メモリ・マップ

V55PIのコーデック命令が必要とするデータ用メモリ領域は次のとおりです。

(1) レジスタ・ファイル空間

パラメータ設定用レジスタ・バンクです。

パラメータの設定についての詳細は，**18.6　各命令の機能説明**を参照してください。

(2) ユーザRAM

符号化ライン変化点テーブル：符号化を行うために必要な変化点情報格納領域
（nビット／ラインの場合，最大2n＋4バイトの領域が必要）

参照ライン・テーブル　：参照ラインの変化点情報格納領域

画素データ・バッファ　：符号化の場合には，スキャナから読み込まれた画素データ，復号化の場合には，モデムから受信された符号化データの格納領域

送信／受信バッファ　：モデム／スキャナに，符号化されたデータを引き渡すためのバッファ

プリント・バッファ　：記録系に，復号化された画素データを引き渡すためのバッファ

(3) ユーザROM

符号化変換テーブル　：MH，MR符号化のための変換テーブル

復号化変換テーブル　：MH，MR復号化のための変換テーブル

(4) 拡張メモリ空間へのアクセス

拡張セグメント・オーバライド・プリフィクス命令（DS2：またはDS3：）を使用して16 Mバイト拡張メモリ空間をアクセスできます。

ただし命令の実行時に使用するセグメント・レジスタDS2，DS3は各命令ごとのパラメータ設定用レジスタ・バンク内のDS2，DS3を使用して行います。

保守／廃止

表18－3　拡張セグメント・オーバライド・プリフィクスの付加可能なコーデック命令

DS2：	DS3：	コーデック命令
可	可	COLTRP
可	不可	MHENC
可	可	MHDEC
可	不可	MRENC
可	可	MRDEC
可	不可	SCHEOL
可	不可	GETBIT
可	可	CNVTRP

記述例

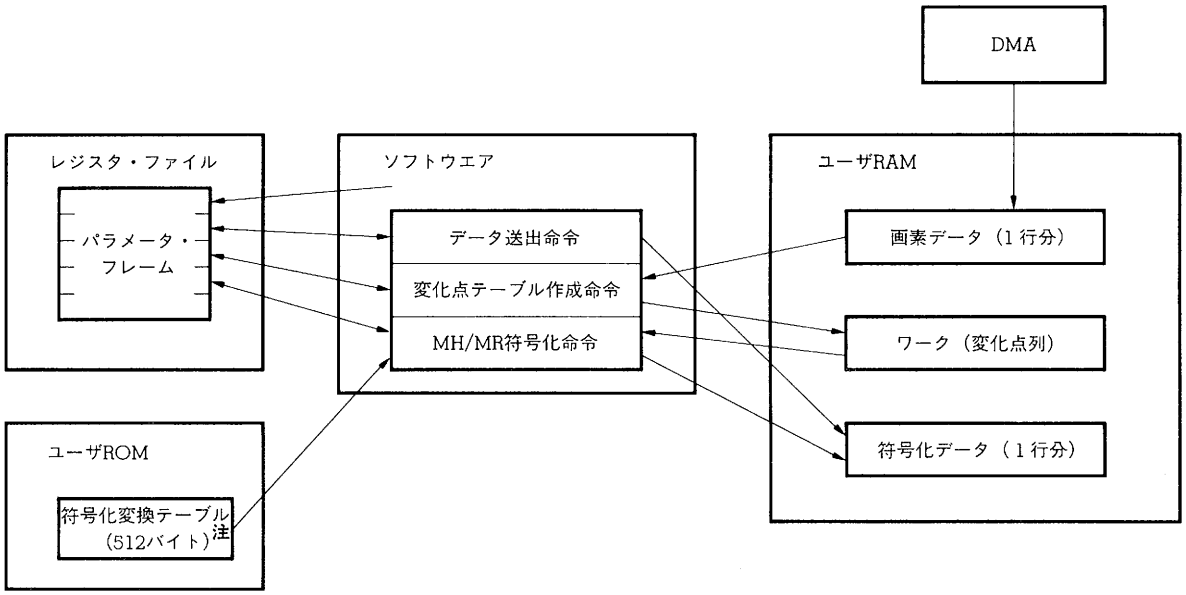
DS2：DS3：COLTRP

DS2：SCHEOL

図18－18に符号化命令と各メモリ上のデータの関係を，図18－19に復号化命令と各メモリ上のデータの関係を示します。

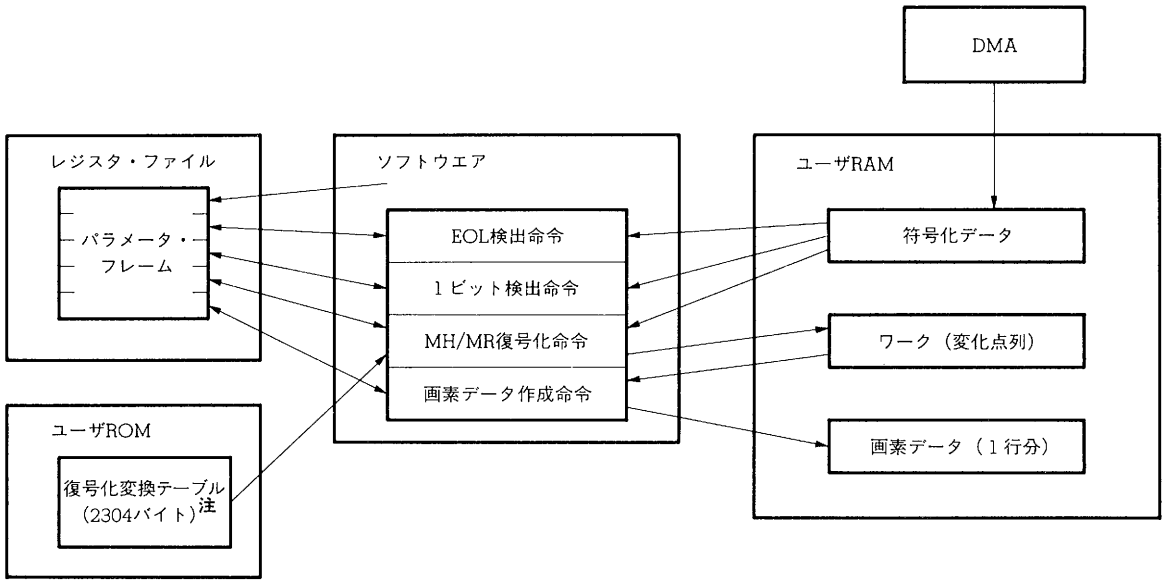
保守／廃止

図18－18　符号化命令と各メモリ上のデータ



注　MH，MR符号化命令の場合です。

図18－19　復号化命令と各メモリ上のデータ



注　MH，MR復号化命令の場合です。

保守／廃止

18.4　符号化／復号化変換テーブル

符号化変換テーブルは，画素データ（ラン・レングス）をMH符号に変換するためのテーブルです。MH符号化命令“MHENC”，MR符号化命令“MRENC”を使用する場合には，**18.4.1　符号化変換テーブル**で規定されるテーブルを指定してください（**18.4.4　符号化変換テーブル（MH符号化命令用）**参照）。

復号化変換テーブルは，符号を画素データ（ラン・レングス）に変換するためのテーブルです。MH復号化命令“MHDEC”，MR復号化命令“MRDEC”を使用する場合には，**18.4.2　復号化変換テーブル**で規定されるテーブルを指定してください（**18.4.3　復号化変換テーブル（MH復号化命令用）**参照）。

符号化／復号化変換テーブルの開始オフセットは，それぞれ“0000H”番地に配置してください。

18.4.1　符号化変換テーブル

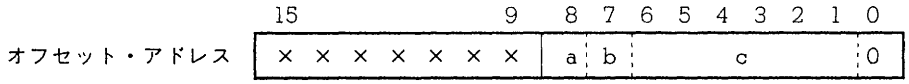
符号化変換テーブルは，次のような規則で構成されます。

ラン・レングス＝Rの画素データ符号（MH）は，符号化変換テーブルの，次に示すようなオフセット・アドレスの指定先に格納されています。

(1) オフセット・アドレス

オフセット・アドレスは，次のように生成されます。

例　R＝64M＋Tの場合



a：0＝白，1＝黒

b：0＝ターミネート符号，1＝メークアップ符号

c：ターミネート符号の場合（b＝0）

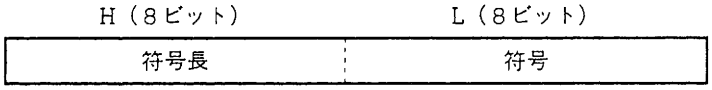
T（2進数）の値

メークアップ符号の場合（b＝1）

M（2進数）の値

(2) オフセット・アドレスの指定先

オフセット・アドレスが指定するワード領域には，次のような符号が格納されます。



保守／廃止

●符号が８ビット未満の場合，符号は右詰めにされます。８ビットに満たない分は“0”となります。

●符号が８ビット以上の場合，符号は左詰めにされます。

●符号はLSB先頭で示されます。

例　白ラン・レングスR＝708の場合

R＝64×11＋4　→　M＝11，T＝4

●メークアップ符号

符号は“０**１１００１１００**”です。

オフセット・アドレス＝××××××××010010110の示すワード領域に次のように格納されます。この領域に格納されない分についてはすべて“0”です。

00001001	**00110011**
符号長	符号

●ターミネート符号

符号は“**１０１１**”です。

オフセット・アドレス＝××××××××000001000の示すワード領域に次のように格納されます。８ビットに満たない分についてはすべて“0”です。

00000100	0000**1101**
符号長	符号

保守／廃止

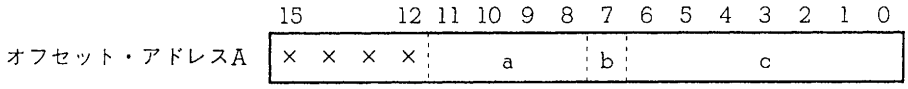
18.4.2　復号化変換テーブル

復号化変換テーブルは，次のような規則で生成されます。

変換の対象となる符号データの，連続する“0”の数をカウントします。

(1) オフセット・アドレスA

オフセット・アドレスAは，次のように生成されます。

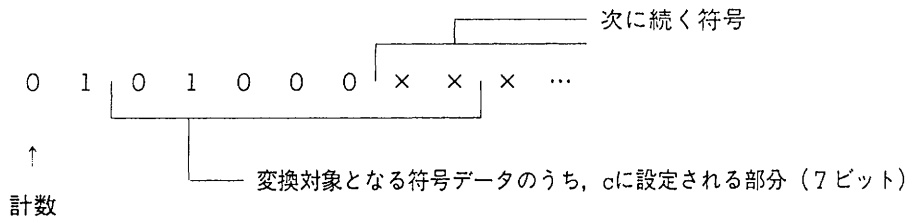


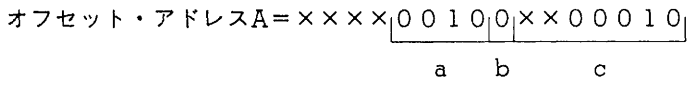
a：（カウントした“0”の数）＋1

b：0＝白，1＝黒

c：変換対象となる符号データ

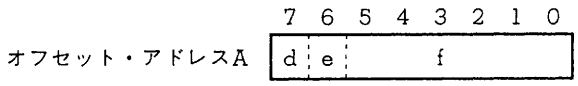
例　変換対象となる符号＝0101000（白ラン・レングス＝24）の場合





(2) オフセット・アドレスAの指定先

オフセット・アドレスAが指定するバイト領域には，次のようなデータが格納されます。



d：0＝白，1＝黒

e：0＝ターミネート符号，1＝メークアップ符号

f：ラン・レングス

保守／廃止

(3) オフセット・アドレスB

(2)で格納された８ビットのデータに“000H”を付加し，12ビットにしたものをオフセット・アドレスBとします。オフセット・アドレスBが指定するバイト領域には，変換対象となる符号の“符号長－（カウントした“0”の数＋1)”が格納されます。

例　オフセット・アドレスB＝××××０００００００１１０００（×０１８H）
（変換対象となる符号が“０１０１０００”の場合）

	7	6	5	4	3	2	1	0	
オフセット・アドレスB	0	0	0	0	0	1	0	1	7－(1＋1)＝5

保守／廃止

18.4.3　復号化変換テーブル（MH復号化命令用）

```
;*******************************************
;*      MH decode (expansion) table        *
;*         compression code -> data        *
;*******************************************
;
;      * EOL
;         Number of '0' = 11
;
;      * EOL + '1'
;       -----4bit---+-1-+-------------7bit--------------
;       +---+---+---+---+---+---+---+---+---+---+---+---+
;       |num of'0'+1|col|           search data         | ---+
;       +---+---+---+---+---+---+---+---+---+---+---+---+    |
;           num of '0'+1: number of '0'+1                    |
;           col: color                                       |
;           search data: 7 bits                              |
;           ex) 0001 1011xxxx                                |
;               num of '0' = 3                               |
;               search data = 1011xxx                        |
;                                                            |
;       --1-+-1-+---------6bit-----------                    |
;       +---+---+---+---+---+---+---+---+                    |
;       |col|t/m|       run length      | <------------------+
;       +---+---+---+---+---+---+---+---+ ------------+
;           col: color                                 |
;           t/m: 0 = terminate code                    |
;                1 = makeup code                       |
;           run length: 6 bits                         |
;                                                      |
;       ------4bit------+-----4bit-------              |
;       +---+---+---+---+---+---+---+---+              |
;       | 0   0   0   0 |sch data length| <------------+
;       +---+---+---+---+---+---+---+---+
;        data length
;           terminate:  number of '0'+1 + seach data length
;           makeup    : (number of '0'+1 + seach data length)*64
```

保守／廃止

```
;
;+++++++++++++ Decode Table +++++++++++++
code    segment at      xxxxh                   ;     offset
        ORG     0h
;------ White Terminal Data Table -------
        db      05h,02h,02h,03h,03h,03h,03h,03h   ; W.T=000h-007h
        db      04h,04h,02h,03h,03h,01h,05h,05h   ; W.T=008h-00Fh
        db      05h,05h,05h,03h,03h,04h,01h,02h   ; W.T=010h-017h
        db      05h,05h,04h,05h,04h,01h,01h,04h   ; W.T=018h-01Fh
        db      04h,04h,04h,04h,04h,04h,04h,05h   ; W.T=020h-
        db      05h,05h,05h,05h,05h,02h,02h,03h
        db      03h,06h,06h,06h,06h,05h,05h,06h   ; W.T=030h-
        db      06h,06h,06h,06h,06h,05h,05h,05h
;------- White Makeup Data Table --------
        db      00h,04h,04h,04h,05h,05h,05h,06h   ; W.M=040h-
        db      06h,06h,06h,07h,07h,07h,07h,07h
        db      07h,07h,07h,07h,07h,07h,07h,07h   ; W.M=050h-057h
        db      07h,07h,04h,07h                   ; W.M=058h-05Bh
;---- White/Black Addition Data Table ---
        db      03h,03h,03h,04h                   ; W/B.M=05Ch-05Fh
        db      04h,04h,04h,04h,04h,04h,04h,04h   ; W/B.M=060h-067h
        db      04h,00h,00h,00h,00h,00h,00h,00h   ; W/B.M=068h dummy=069h-
        db      00h,00h,00h,00h,00h,00h,00h,00h   ; dummy
        db      00h,00h,00h,00h,00h,00h,00h,00h
;------ Black Terminate Data Table ------
        db      05h,01h,01h,01h,01h,01h,01h,01h   ; B.T=080h-087h
        db      02h,02h,02h,02h,02h,02h,02h,04h   ; B.T=088h-08Fh
        db      04h,04h,03h,06h,06h,06h,05h,05h   ; B.T=090h-
        db      04h,04h,07h,07h,07h,07h,06h,06h
        db      06h,06h,07h,07h,07h,07h,07h,07h   ; B.T=0A0h-
        db      06h,06h,07h,07h,06h,06h,06h,06h
        db      06h,06h,06h,06h,05h,05h,05h,05h   ; B.T=0B0h-
        db      05h,06h,06h,05h,05h,06h,06h,06h
;------- Black Makeup Data Table --------
        db      00h,03h,07h,07h,06h,05h,05h,05h   ; B.M=0C0h-
        db      06h,06h,06h,06h,06h,06h,06h,06h
        db      06h,06h,06h,06h,06h,06h,06h,06h   ; B.M=0D0h-0D7h
        db      06h,06h,06h,06h                   ; B.M=0D8h-0DBh
```

保守／廃止

```
        ORG     0100h
;++++++++++++++ Decode Table ++++++++++++++
        db      003h,005h,009h,006h,042h,00Eh,004h,007h   ; address=0100h-
        db      003h,005h,010h,006h,008h,041h,004h,007h
        db      003h,005h,009h,006h,042h,00Fh,004h,007h   ; address=0110h-
        db      003h,005h,011h,006h,008h,041h,004h,007h
        db      003h,005h,009h,006h,042h,00Eh,004h,007h   ; address=0120h-
        db      003h,005h,010h,006h,008h,041h,004h,007h
        db      003h,005h,009h,006h,042h,00Fh,004h,007h   ; address=0130h-
        db      003h,005h,011h,006h,008h,041h,004h,007h
        db      003h,005h,009h,006h,042h,00Eh,004h,007h   ; address=0140h-
        db      003h,005h,010h,006h,008h,041h,004h,007h
        db      003h,005h,009h,006h,042h,00Fh,004h,007h   ; address=0150h-
        db      003h,005h,011h,006h,008h,041h,004h,007h
        db      003h,005h,009h,006h,042h,00Eh,004h,007h   ; address=0160h-
        db      003h,005h,010h,006h,008h,041h,004h,007h
        db      003h,005h,009h,006h,042h,00Fh,004h,007h   ; address=0170h-
        db      003h,005h,011h,006h,008h,041h,004h,007h
;
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=0180h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=0190h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=01A0h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=01B0h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=01C0h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=01D0h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=01E0h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
        db      083h,082h,083h,082h,083h,082h,083h,082h   ; address=01F0h-
        db      083h,082h,083h,082h,083h,082h,083h,082h
```

保守／廃止

```
        ORG     0200h
;++++++++++++++ Decode Table ++++++++++++++
        db      00Bh,05Ah,018h,002h,01Bh,049h,037h,002h   ; address=0200h-
        db      00Bh,047h,033h,002h,057h,053h,043h,002h
        db      00Bh,05Ah,031h,002h,03Bh,04Fh,039h,002h   ; address=0210h-
        db      00Bh,04Bh,019h,002h,012h,044h,043h,002h
        db      00Bh,05Ah,018h,002h,01Bh,04Dh,038h,002h   ; address=0220h-
        db      00Bh,048h,034h,002h,059h,055h,043h,002h
        db      00Bh,05Ah,032h,002h,03Ch,051h,03Ah,002h   ; address=0230h-
        db      00Bh,04Ah,019h,002h,012h,044h,043h,002h
        db      00Bh,05Ah,018h,002h,01Bh,049h,037h,002h   ; address=0240h-
        db      00Bh,047h,033h,002h,058h,054h,043h,002h
        db      00Bh,05Ah,031h,002h,03Bh,050h,039h,002h   ; address=0250h-
        db      00Bh,04Ch,019h,002h,012h,044h,043h,002h
        db      00Bh,05Ah,018h,002h,01Bh,04Eh,038h,002h   ; address=0260h-
        db      00Bh,048h,034h,002h,05Bh,056h,043h,002h
        db      00Bh,05Ah,032h,002h,03Ch,052h,03Ah,002h   ; address=0270h-
        db      00Bh,04Ah,019h,002h,012h,044h,043h,002h
;
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=0280h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=0290h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=02A0h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=02B0h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=02C0h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=02D0h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=02E0h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
        db      081h,084h,081h,084h,081h,084h,081h,084h   ; address=02F0h-
        db      081h,084h,081h,084h,081h,084h,081h,084h
```

保守／廃止

```
        ORG     0300h
;++++++++++++++ Decode Table ++++++++++++++
        db      00Ch,01Ch,027h,00Ah,035h,03Fh,02Bh,00Ah   ; address=0300h-
        db      00Ch,03Dh,029h,00Ah,01Ah,045h,015h,00Ah
        db      00Ch,01Ch,028h,00Ah,036h,000h,02Ch,00Ah   ; address=0310h-
        db      00Ch,03Eh,02Ah,00Ah,01Ah,046h,015h,00Ah
        db      00Ch,01Ch,027h,00Ah,035h,03Fh,02Bh,00Ah   ; address=0320h-
        db      00Ch,03Dh,029h,00Ah,01Ah,045h,015h,00Ah
        db      00Ch,01Ch,028h,00Ah,036h,000h,02Ch,00Ah   ; address=0330h-
        db      00Ch,03Eh,02Ah,00Ah,01Ah,046h,015h,00Ah
        db      00Ch,01Ch,027h,00Ah,035h,03Fh,02Bh,00Ah   ; address=0340h-
        db      00Ch,03Dh,029h,00Ah,01Ah,045h,015h,00Ah
        db      00Ch,01Ch,028h,00Ah,036h,000h,02Ch,00Ah   ; address=0350h-
        db      00Ch,03Eh,02Ah,00Ah,01Ah,046h,015h,00Ah
        db      00Ch,01Ch,027h,00Ah,035h,03Fh,02Bh,00Ah   ; address=0360h-
        db      00Ch,03Dh,029h,00Ah,01Ah,045h,015h,00Ah
        db      00Ch,01Ch,028h,00Ah,036h,000h,02Ch,00Ah   ; address=0370h-
        db      00Ch,03Eh,02Ah,00Ah,01Ah,046h,015h,00Ah

        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=0380h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=0390h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=03A0h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=03B0h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=03C0h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=03D0h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=03E0h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
        db      086h,085h,086h,085h,086h,085h,086h,085h   ; address=03F0h-
        db      086h,085h,086h,085h,086h,085h,086h,085h
```

保守／廃止

```
        ORG     0400h
;++++++++++++++ Decode Table ++++++++++++++
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0400h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0410h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0420h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0430h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0440h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0450h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0460h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
        db      014h,013h,023h,001h,021h,01Fh,025h,001h   ; address=0470h-
        db      014h,013h,024h,001h,022h,020h,026h,001h
;
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=0480h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=0490h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=04A0h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=04B0h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=04C0h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=04D0h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=04E0h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
        db      089h,087h,088h,087h,089h,087h,088h,087h   ; address=04F0h-
        db      089h,087h,088h,087h,089h,087h,088h,087h
```

保守／廃止

```
        ORG     0500h
;+++++++++++++ Decode Table +++++++++++++
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0500h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0510h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0520h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0530h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0540h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0550h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0560h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh   ; address=0570h-
        db      017h,00Dh,02Fh,00Dh,017h,00Dh,030h,00Dh
;
        db      08Ah,08Fh,08Bh,08Ch,08Ah,094h,08Bh,08Ch   ; address=0580h-
        db      08Ah,0C2h,08Bh,08Ch,08Ah,095h,08Bh,08Ch
        db      08Ah,08Fh,08Bh,08Ch,08Ah,0A4h,08Bh,08Ch   ; address=0590h-
        db      08Ah,09Ch,08Bh,08Ch,08Ah,080h,08Bh,08Ch
        db      08Ah,08Fh,08Bh,08Ch,08Ah,0A2h,08Bh,08Ch   ; address=05A0h-
        db      08Ah,09Ah,08Bh,08Ch,08Ah,0AAh,08Bh,08Ch
        db      08Ah,08Fh,08Bh,08Ch,08Ah,0A6h,08Bh,08Ch   ; address=05B0h-
        db      08Ah,093h,08Bh,08Ch,08Ah,080h,08Bh,08Ch
        db      08Ah,08Fh,08Bh,08Ch,08Ah,094h,08Bh,08Ch   ; address=05C0h-
        db      08Ah,0C3h,08Bh,08Ch,08Ah,095h,08Bh,08Ch
        db      08Ah,08Fh,08Bh,08Ch,08Ah,0A5h,08Bh,08Ch   ; address=05D0h-
        db      08Ah,09Dh,08Bh,08Ch,08Ah,080h,08Bh,08Ch
        db      08Ah,08Fh,08Bh,08Ch,08Ah,0A3h,08Bh,08Ch   ; address=05E0h-
        db      08Ah,09Bh,08Bh,08Ch,08Ah,0ABh,08Bh,08Ch
        db      08Ah,08Fh,08Bh,08Ch,08Ah,0A7h,08Bh,08Ch   ; address=05F0h-
        db      08Ah,093h,08Bh,08Ch,08Ah,080h,08Bh,08Ch
```

保守／廃止

```
        ORG     0600h
;++++++++++++++ Decode Table ++++++++++++++
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=0600h-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=0610h-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=0620h-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=062Eh-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=0640h-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=0650h-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=0660h-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h   ; address=0670h-
        db      02Dh,016h,02Eh,016h,02Dh,016h,02Eh,016h
;
        db      08Dh,091h,097h,08Eh,08Dh,09Eh,0B9h,08Eh   ; address=0680h-
        db      08Dh,0B0h,0ACh,08Eh,08Dh,0A8h,090h,08Eh
        db      08Dh,091h,0B2h,08Eh,08Dh,0A0h,0BDh,08Eh   ; address=0690h-
        db      08Dh,0BEh,0AEh,08Eh,08Dh,096h,090h,08Eh
        db      08Dh,091h,097h,08Eh,08Dh,09Fh,0BAh,08Eh   ; address=06A0h-
        db      08Dh,0B1h,0ADh,08Eh,08Dh,0A9h,090h,08Eh
        db      08Dh,091h,0B3h,08Eh,08Dh,0A1h,0C4h,08Eh   ; address=06B0h-
        db      08Dh,0BFh,0AFh,08Eh,08Dh,096h,090h,08Eh
        db      08Dh,091h,097h,08Eh,08Dh,09Eh,0B9h,08Eh   ; address=06C0h-
        db      08Dh,0B0h,0ACh,08Eh,08Dh,0A8h,090h,08Eh
        db      08Dh,091h,0B2h,08Eh,08Dh,0A0h,0BDh,08Eh   ; address=06D0h-
        db      08Dh,0BEh,0AEh,08Eh,08Dh,096h,090h,08Eh
        db      08Dh,091h,097h,08Eh,08Dh,09Fh,0BAh,08Eh   ; address=06E0h-
        db      08Dh,0B1h,0ADh,08Eh,08Dh,0A9h,090h,08Eh
        db      08Dh,091h,0B3h,08Eh,08Dh,0A1h,0C4h,08Eh   ; address=06F0h-
        db      08Dh,0BFh,0AFh,08Eh,08Dh,096h,090h,08Eh
```

保守／廃止

```
        ORG     0700h
;++++++++++++++ Decode Table ++++++++++++++
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=0700h-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=0710h-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=0720h-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=072Eh-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=0740h-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=0750h-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=0760h-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh   ; address=0770h-
        db      01Dh,01Eh,01Dh,01Eh,01Dh,01Eh,01Dh,01Eh
;
        db      092h,099h,0B8h,0B6h,0B4h,0C6h,0BCh,0C1h   ; address=0780h-
        db      092h,0DAh,0D6h,0D0h,0CCh,0C8h,098h,0C1h
        db      092h,099h,0D4h,0CEh,0CAh,0C7h,0D8h,0C1h   ; address=0790h-
        db      092h,0C5h,0BBh,0D2h,0B7h,0B5h,098h,0C1h
        db      092h,099h,0B8h,0B6h,0B4h,0C6h,0BCh,0C1h   ; address=07A0h-
        db      092h,0DBh,0D7h,0D1h,0CDh,0C9h,098h,0C1h
        db      092h,099h,0D5h,0CFh,0CBh,0C7h,0D9h,0C1h   ; address=07B0h-
        db      092h,0C5h,0BBh,0D3h,0B7h,0B5h,098h,0C1h
        db      092h,099h,0B8h,0B6h,0B4h,0C6h,0BCh,0C1h   ; address=07C0h-
        db      092h,0DAh,0D6h,0D0h,0CCh,0C8h,098h,0C1h
        db      092h,099h,0D4h,0CEh,0CAh,0C7h,0D8h,0C1h   ; address=07D0h-
        db      092h,0C5h,0BBh,0D2h,0B7h,0B5h,098h,0C1h
        db      092h,099h,0B8h,0B6h,0B4h,0C6h,0BCh,0C1h   ; address=07E0h-
        db      092h,0DBh,0D7h,0D1h,0CDh,0C9h,098h,0C1h
        db      092h,099h,0D5h,0CFh,0CBh,0C7h,0D9h,0C1h   ; address=07F0h-
        db      092h,0C5h,0BBh,0D3h,0B7h,0B5h,098h,0C1h
```

保守／廃止

```
        ORG     0800h
;+++++++++++++ Decode Table +++++++++++++
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0800h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0810h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0820h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0830h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0840h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0850h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0860h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0870h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
;
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0880h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=0890h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=08A0h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=08B0h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=08C0h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=08D0h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=08E0h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
        db      05Ch,05Dh,061h,065h,05Fh,05Eh,063h,067h   ; address=08F0h-
        db      05Ch,05Dh,062h,066h,060h,05Eh,064h,068h
code    ends

        end
```

保守／廃止

18.4.4　符号化変換テーブル（MH符号化命令用）

```
;*****************************************
;*      MH encode (compression) table      *
;*         data -> compression code        *
;*****************************************
;
;       address= +XXXXh
;
;       -1+1+----6bit---+1-      -----8bit(H)----+----8bit(L)-----
;       +-+-+-+-+-+-+-+-+-+-+      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
;       |a|b|     c     |0| ===> |  code length  |               |
;       +-+-+-+-+-+-+-+-+-+-+      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
;
;       * Run Length >= 64
;        Run_Length = 64T +T
;        a: 0 = white
;           1 = black
;        b: 0 = terminate --- c = T
;           1 = makeup    --- c = M
;
;       * Run Length < 64
;        Only change the terminate code
;
code    segment at      xxxxh
        ORG     0h
;++++++++++ White Encode Table ++++++++++
;-------------- Terminate ---------------
        db      0ACh, 008h              ; address= +0000h
        db      038h, 006h
        db      00Eh, 004h
        db      001h, 004h
        db      00Dh, 004h
        db      003h, 004h
        db      007h, 004h
        db      00Fh, 004h
        db      019h, 005h              ; address= +0010h
        db      005h, 005h
        db      01Ch, 005h
        db      002h, 005h
        db      004h, 006h
        db      030h, 006h
```

保守／廃止

```
db      00Bh, 006h
db      02Bh, 006h
db      015h, 006h              ; address= +0020h
db      035h, 006h
db      072h, 007h
db      018h, 007h
db      008h, 007h
db      074h, 007h
db      060h, 007h
db      010h, 007h
db      00Ah, 007h              ; address= +0030h
db      06Ah, 007h
db      064h, 007h
db      012h, 007h
db      00Ch, 007h
db      040h, 008h
db      0C0h, 008h
db      058h, 008h
db      0D8h, 008h              ; address= +0040h
db      048h, 008h
db      0C8h, 008h
db      028h, 008h
db      0A8h, 008h
db      068h, 008h
db      0E8h, 008h
db      014h, 008h
db      094h, 008h              ; address= +0050h
db      054h, 008h
db      0D4h, 008h
db      034h, 008h
db      0B4h, 008h
db      020h, 008h
db      0A0h, 008h
db      050h, 008h
db      0D0h, 008h              ; address= +0060h
db      04Ah, 008h
db      0CAh, 008h
db      02Ah, 008h
db      0AAh, 008h
```

保守／廃止

```
        db      024h, 008h
        db      0A4h, 008h
        db      01Ah, 008h
        db      09Ah, 008h              ; address= +0070h
        db      05Ah, 008h
        db      0DAh, 008h
        db      052h, 008h
        db      0D2h, 008h
        db      04Ch, 008h
        db      0CCh, 008h
        db      02Ch, 008h
;-------------- Terminate ---------------
        db      000h, 000h              ; address= +0080h / no data
        db      01Bh, 005h
        db      009h, 005h
        db      03Ah, 006h
        db      076h, 007h
        db      06Ch, 008h
        db      0ECh, 008h
        db      026h, 008h
        db      0A6h, 008h              ; address= +0090h
        db      016h, 008h
        db      0E6h, 008h
        db      033h, 009h
        db      0B3h, 009h
        db      04Bh, 009h
        db      0CBh, 009h
        db      02Bh, 009h
        db      0ABh, 009h              ; address= +00A0h
        db      06Bh, 009h
        db      0EBh, 009h
        db      01Bh, 009h
        db      09Bh, 009h
        db      05Bh, 009h
        db      0DBh, 009h
        db      019h, 009h
        db      099h, 009h              ; address= +00B0h
        db      059h, 009h
        db      006h, 006h
```

保守／廃止

```
        db      0D9h, 009h
        db      010h, 00Bh
        db      030h, 00Bh
        db      0B0h, 00Bh
        db      048h, 00Ch
        db      0C8h, 00Ch              ; address= +00C0h
        db      028h, 00Ch
        db      0A8h, 00Ch
        db      068h, 00Ch
        db      0E8h, 00Ch
        db      038h, 00Ch
        db      0B8h, 00Ch
        db      078h, 00Ch
        db      0F8h, 00Ch              ; address= +00D0h

;++++++++++ Black Encode Table ++++++++++
        ORG     0100h
;-------------- Terminate ---------------
        db      0ECh, 00Ah              ; address= +0100h
        db      002h, 003h
        db      003h, 002h
        db      001h, 002h
        db      006h, 003h
        db      00Ch, 004h
        db      004h, 004h
        db      018h, 005h
        db      028h, 006h              ; address= +0110h
        db      008h, 006h
        db      010h, 007h
        db      050h, 007h
        db      070h, 007h
        db      020h, 008h
        db      0E0h, 008h
        db      018h, 009h
        db      0E8h, 00Ah              ; address= +0120h
        db      018h, 00Ah
        db      010h, 00Ah
        db      0E6h, 00Bh
        db      016h, 00Bh
```

保守／廃止

```
db      036h, 00Bh
db      0ECh, 00Bh
db      014h, 00Bh
db      0E8h, 00Bh              ; address= +0130h
db      018h, 00Bh
db      053h, 00Ch
db      0D3h, 00Ch
db      033h, 00Ch
db      0B3h, 00Ch
db      016h, 00Ch
db      096h, 00Ch
db      056h, 00Ch              ; address= +0140h
db      0D6h, 00Ch
db      04Bh, 00Ch
db      0CBh, 00Ch
db      02Bh, 00Ch
db      0ABh, 00Ch
db      06Bh, 00Ch
db      0EBh, 00Ch
db      036h, 00Ch              ; address= +0150h
db      0B6h, 00Ch
db      05Bh, 00Ch
db      0DBh, 00Ch
db      02Ah, 00Ch
db      0AAh, 00Ch
db      06Ah, 00Ch
db      0EAh, 00Ch
db      026h, 00Ch              ; address= +0160h
db      0A6h, 00Ch
db      04Ah, 00Ch
db      0CAh, 00Ch
db      024h, 00Ch
db      0ECh, 00Ch
db      01Ch, 00Ch
db      0E4h, 00Ch
db      014h, 00Ch              ; address= +0170h
db      01Ah, 00Ch
db      09Ah, 00Ch
db      0D4h, 00Ch
```

保守／廃止

```
        db      034h, 00Ch
        db      05Ah, 00Ch
        db      066h, 00Ch
        db      0E6h, 00Ch
;-------------- Makeup ------------------
        db      000h, 000h              ; address= +0180h / no data
        db      0F0h, 00Ah
        db      013h, 00Ch
        db      093h, 00Ch
        db      0DAh, 00Ch
        db      0CCh, 00Ch
        db      02Ch, 00Ch
        db      0ACh, 00Ch
        db      036h, 00Dh              ; address= +0190h
        db      0B6h, 00Dh
        db      052h, 00Dh
        db      0D2h, 00Dh
        db      032h, 00Dh
        db      0B2h, 00Dh
        db      04Eh, 00Dh
        db      0CEh, 00Dh
        db      02Eh, 00Dh              ; address= +01A0h
        db      0AEh, 00Dh
        db      06Eh, 00Dh
        db      0EEh, 00Dh
        db      04Ah, 00Dh
        db      0CAh, 00Dh
        db      02Ah, 00Dh
        db      0AAh, 00Dh
        db      05Ah, 00Dh              ; address= +01B0h
        db      0DAh, 00Dh
        db      026h, 00Dh
        db      0A6h, 00Dh
        db      010h, 00Bh
        db      030h, 00Bh
        db      0B0h, 00Bh
        db      048h, 00Ch
        db      0C8h, 00Ch              ; address= +01C0h
        db      028h, 00Ch
```

保守／廃止

```
        db      0A8h, 00Ch
        db      068h, 00Ch
        db      0E8h, 00Ch
        db      038h, 00Ch
        db      0B8h, 00Ch
        db      078h, 00Ch
        db      0F8h, 00Ch              ; address= +01D0h

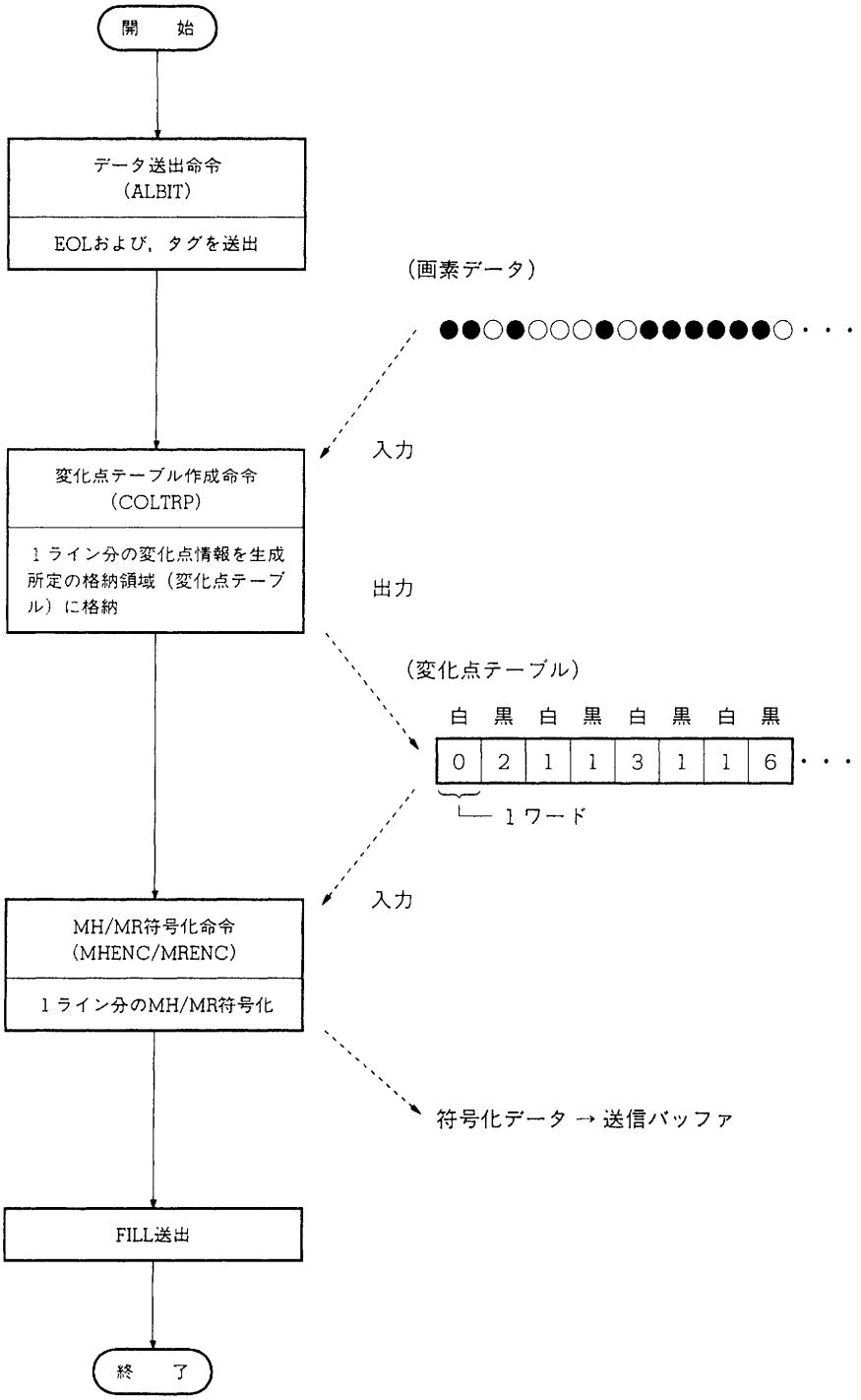
code    ends
        end
```

保守／廃止

18.5　処理フロー

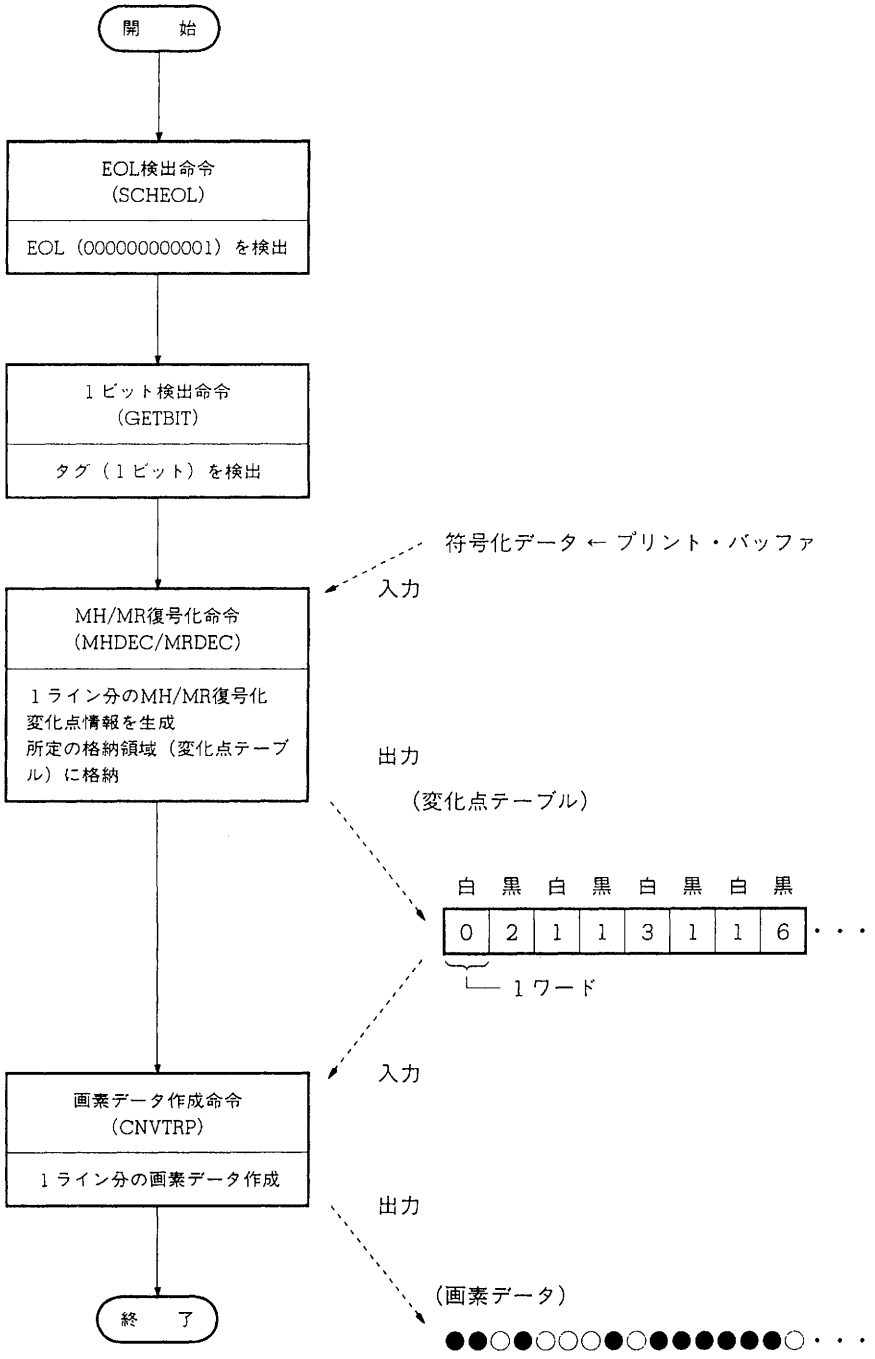
符号化／復号化の処理において，**18.1**　特徴に示した各専用命令は図18－20，図18－21に示すような順に使用されます。

図18－20　1ライン分の符号化処理の流れ



保守／廃止

図18－21　1ライン分の復号化処理の流れ



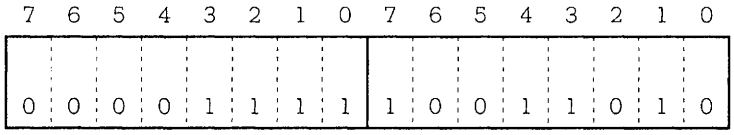
保守／廃止

18.6　各命令の機能説明

18.6.1　データ送出命令

＜記　述　形　式＞　ALBIT（Add Last BIT）

＜命 令 語 形 式＞



＜バ　イ　ト　数＞　2

＜ク ロ ッ ク 数＞

外部バス幅	条　件	クロック数
8ビット・バス幅	CL＋CH＜16のとき	14＋n
	CL＋CH＝16 かつCH＝0のとき	18
	上記以外	43－m＋n
16ビット・バス幅	CL＋CH＜16のとき	14＋n
	CL＋CH＝16 かつCH＝0のとき	18
	上記以外	43－m＋n

m：CHレジスタの設定値

n：CLレジスタの設定値

＜機　　能＞

符号化処理においてEOL，FILLなど16ビット以下のデータを送信バッファへ出力するための命令です。データの先頭はLSBに入ります。

指定された送出したいデータが，処理中の16ビットに満たない半端符号と合わせて16ビット以上にまとまったところで指定された送信バッファへ出力します。16ビットを越える分，あるいは合わせて16ビットに満たないデータはパラメータとしてBW（処理中半端符号データ）に，そのデータのビット数はパラメータとしてCH（処理中半端符号ビット数）に保持します。

割り込み要求のサンプリングを行いません。

保守／廃止

＜入力パラメータ＞

BW ：処理中半端符号データ

CH ：処理中半端符号ビット数

DW：送出符号データ

CL ：送出符号データ・ビット数

BP ：バッファ・サイズ（送信バッファの残りバイト・サイズ）

DSl：送信バッファ・セグメント

IY ：送信バッファ・オフセット

＜出力パラメータ＞

BW ：処理中半端符号データ

CH ：処理中半端符号ビット数

BP ：バッファ・サイズ（送信バッファの残りバイト・サイズ）

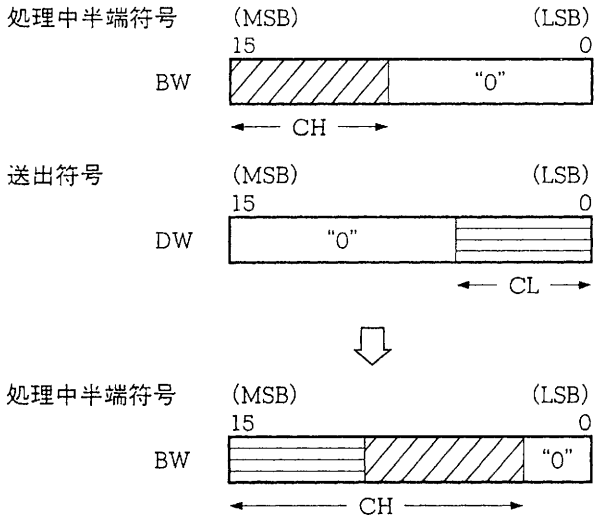
DSl：送信バッファ・セグメント

IY ：送信バッファ・オフセット

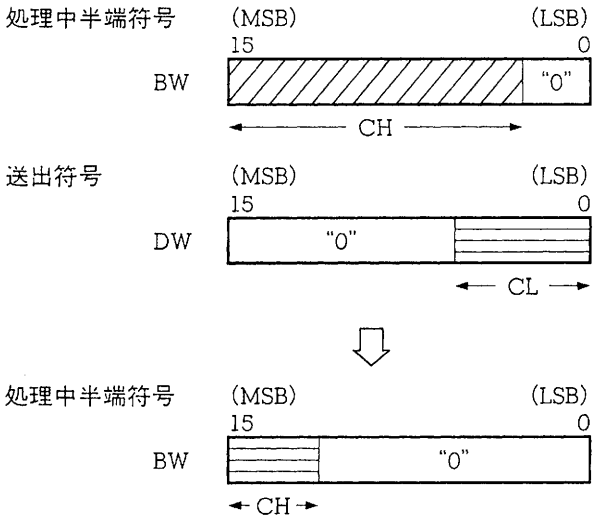
Zフラグ	CYフラグ	送信バッファ残り	データ残り
0	不定	あり	—
1	0	なし	なし
1	1	なし	あり

保守／廃止

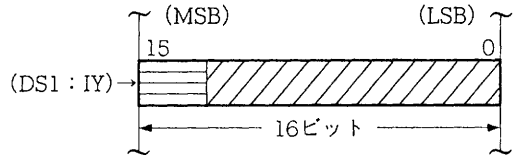
例1．処理中半端符号と送出符号が合わせて16ビットに満たない場合



2．処理中半端符号と送出符号が合わせて16ビット以上の場合



送信バッファ



IY ← IY＋2

BP ← BP－2

保守／廃止

18.6.2　変化点テーブル作成命令

<記　述　形　式>　COLTRP（Collect Turning Point）

<命 令 語 形 式>

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	1	0	0	1	1	0	1	1

<バ　イ　ト　数>　2

<ク ロ ッ ク 数>

COLTRP命令実行中に割り込みによる中断がない場合のクロック数を示します。

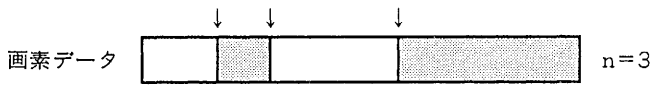
なお，割り込みによる中断があった場合は，どの画素データまで処理が終了しているかによってクロック数が異なってくるため，数値は算出していません。

外部バス幅	条　件	クロック数
8ビット・バス幅	画素データが白から始まる場合	17＋60m＋7n＋2mT
	画素データが黒から始まる場合	24＋60m＋7n＋2mT
16ビット・バス幅	画素データが白から始まる場合	17＋57m＋7n＋mT
	画素データが黒から始まる場合	24＋57m＋7n＋mT

m：画素データ・バッファ・サイズ（バイト）

n ：画素データの色が白→黒へ，または黒→白へ変化する回数

例



T：画素データ・バッファが置かれている領域のウエイト数

<機　　能>

符号化処理において，１ライン分の画素データの変化点情報（白／黒のラン・レングス）を指定されたメモリ（＝変化点テーブル）上に生成する命令です。データの先頭はLSBに入ります。

変化点テーブルは，白のラン・レングスを先頭に１ワードごとに各色（白／黒）のラン・レングスを順次格納します。変化点テーブルの最終ワードには，１ライン分の終了情報として“FFFFH”が格納されます。

保守／廃止

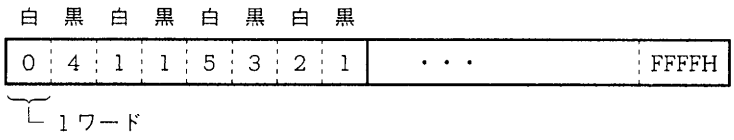
中断処理はありません。１ワード分の画素データ処理ごとに割り込み要求のサンプリングを行います。

（画素データ）

●●●●○●○○○○○●●●○○●○…

↓　COLTRP命令発行

（変化点テーブル）



＜入力パラメータ＞

AL：レジスタ・バンク番号

ALで指定されるレジスタ・バンク

00H	（画素データ・バッファ・セグメント）	（DS2）	拡張セグメント・オーバライド・プリフィクス注1
02H	（変化点テーブル・セグメント）	（DS3）	
04H	ワーク・エリア		
06H	ワーク・エリア		
08H	画素データ・バッファ・セグメント	（DS0）	入力データ処理情報
0AH	画素データ・バッファ・オフセット	（ポインタ）	
0CH	画素データ・バッファ・サイズ注2	（カウンタ）	
0EH	変化点テーブル・セグメント	（DS1）	出力データ処理情報
10H	変化点テーブル・オフセット	（ポインタ）	
12H	ワーク・エリア		
14H	ワーク・エリア		
16H	ワーク・エリア		
18H	ワーク・エリア		
1AH	ワーク・エリア		
1CH	０クリア注3/ワーク・エリア		０クリア必要パラメータ
1EH	０クリア注3/ワーク・エリア		

注1. DS2：，DS3：を付加した場合，オフセット値は08H，0EHの代わりにそれぞれ00H，02Hが参照されます。

2. バイト単位。偶数で指定してください。

3. 新規命令処理時に０クリアを行います。

注意　各命令で使用されるパラメータ用のレジスタ・バンク内のポインタ，カウンタ値は，命令実行によって更新されます。

保守／廃止

18.6.3　MH符号化命令

＜記　述　形　式＞　MHENC（MH Encode）

＜命 令 語 形 式＞

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	1	0	0	1	0	0	1	1

＜バ　イ　ト　数＞　2

＜ク ロ ッ ク 数＞

不定（処理パターンにより，数値が異なるため）

＜機　　能＞

１ライン分の画素データの変化点情報を入力し，所定の符号化変換テーブルによりMH符号化を行い，その符号を指定されたメモリ（送信バッファ）上に送出する命令です。データの先頭はLSBに入ります。

各色（白または黒）の変化点情報（変化点テーブルの１ワード）の符号化処理ごとに割り込み要求のサンプリングを行います。

MHENC命令において，連続した色（白または黒）で符号化できる最大画素数は合計5183画素です。

＜入力パラメータ＞

AL　：レジスタ・バンク番号指定

BW　：処理中半端符号データ

CH　：処理中半端符号ビット数

BP　：バッファ・サイズ（送信バッファの残りバイト・サイズ）

DS1：送信バッファ・セグメント

IY　：送信バッファ・オフセット

保守／廃止

ALで指定されるレジスタ・バンク

00H	（変化点テーブル・セグメント）　（DS2）	拡張セグメント・オーバライド・プリフィクス[注1]
02H	ワーク・エリア	
04H	ワーク・エリア	
06H	ワーク・エリア	
08H	変化点テーブル・セグメント　（DS0）	入力データ処理情報
0AH	符号化ライン変化点テーブル・オフセット　（ポインタ）	
0CH	ワーク・エリア	
0EH	符号化変換テーブル・セグメント　（DS1）	出力データ処理情報
10H	ワーク・エリア	
12H	ワーク・エリア	
14H	ワーク・エリア	
16H	ワーク・エリア	
18H	ワーク・エリア	
1AH	ワーク・エリア	
1CH	0クリア[注2]ワーク・エリア	0クリア必要パラメータ
1EH	0クリア[注2]ワーク・エリア	

注1. DS2：を付加した場合，オフセット値は08Hの代わりに00Hが参照されます。

2. 新規命令処理時に0クリアを行います。
中断した処理を再開する場合，0クリアが必要なパラメータは中断時の状態のままにしておいてください。

＜出力パラメータ＞

BW ：処理中半端符号データ

CH ：処理中半端符号ビット数

BP ：バッファ・サイズ（送信バッファの残りバイト・サイズ）

DS1：送信バッファ・セグメント

IY ：送信バッファ・オフセット

Zフラグ	CYフラグ	送信バッファ残り	データ残り	処理
0	0	あり	CHの値	終了
0	1	—	—	—
1	0	なし	なし	終了
1	1	なし	あり	中断

★ **注意　最後の符号化データを送信バッファに送出した際に，送信バッファの残りバイト・サイズがちょうど"0"となった場合，MHENC命令実行後のフラグ内容は，Z＝1，CY＝1（送信バッファ残りなし，データ残りあり，処理中断）となります。したがって，MHENC命令を再実行してください。**

保守／廃止

18.6.4　MR符号化命令

＜記　述　形　式＞　MRENC（MR Encode）

＜命 令 語 形 式＞

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	1

＜バ　イ　ト　数＞　2

＜ク ロ ッ ク 数＞

不定（処理パターンにより，数値が異なるため）

＜機　　能＞

参照ラインの変化点情報と符号化ラインの変化点情報とを入力し，１ライン分のMR符号化を行い，その符号を指定されたメモリ（送信バッファ）上に送出する命令です。データの先頭はLSBに入ります。

MR符号化の１モード単位（パス・モード，垂直モード，水平モードのいずれか）の符号化処理ごとに割り込み要求のサンプリングを行います。

MRENC命令の水平モードのMH符号化において，連続した色（白または黒）で符号化できる最大画素数は合計5183画素です。

＜入力パラメータ＞

AL　：レジスタ・バンク番号指定

BW　：処理中半端符号データ

CH　：処理中半端符号ビット数

BP　：バッファ・サイズ（送信バッファの残りバイト・サイズ）

DS1：送信バッファ・セグメント

IY　：送信バッファ・オフセット

保守／廃止

ALで指定されるレジスタ・バンク

00H	（変化点テーブル・セグメント）　(DS2)	拡張セグメント・オーバライド・プリフィクス[注1]
02H	ワーク・エリア	
04H	ワーク・エリア	
06H	ワーク・エリア	
08H	変化点テーブル・セグメント　(DS0)	
0AH	符号化ライン変化点テーブル・オフセット　（ポインタ）	
0CH	参照ライン変化点テーブル・オフセット（ポインタ）	
0EH	符号化変換テーブル・セグメント　(DS1)	
10H	ワーク・エリア	
12H	ワーク・エリア	
14H	ワーク・エリア	
16H	ワーク・エリア	
18H	ワーク・エリア	
1AH	ワーク・エリア	
1CH	0クリア[注2]/ワーク・エリア	0クリア必要パラメータ
1EH	0クリア[注2]/ワーク・エリア	

注1. DS2：を付加した場合，オフセット値は08Hの代わりに00Hが参照されます。

2. 新規命令処理時に0クリアを行います。
中断した処理を再開する場合，0クリアが必要なパラメータは中断時の状態のままにしておいてください。

＜出力パラメータ＞

BW ：処理中半端符号データ

CH ：処理中半端符号ビット数

BP ：バッファ・サイズ（送信バッファの残りバイト・サイズ）

DS1：送信バッファ・セグメント

IY ：送信バッファ・オフセット

Zフラグ	CYフラグ	送信バッファ残り	データ残り	処理
0	0	あり	CHの値	終了
0	1	—	—	—
1	0	なし	なし	終了
1	1	なし	あり	中断

★ **注意　最後の符号化データを送信バッファに送出した際に，送信バッファの残りバイト・サイズがちょうど"0"となった場合，MRENC命令実行後のフラグ内容は，Z＝1，CY＝1（送信バッファ残りなし，データ残りあり，処理中断）となります。したがって，MRENC命令を再実行してください。**

保守／廃止

18.6.5　EOL検出命令

<記　述　形　式>　SCHEOL (Search EOL)

<命 令 語 形 式>

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	0	1	1	1	1	0	0	0

<バ　イ　ト　数>　2

<ク ロ ッ ク 数>

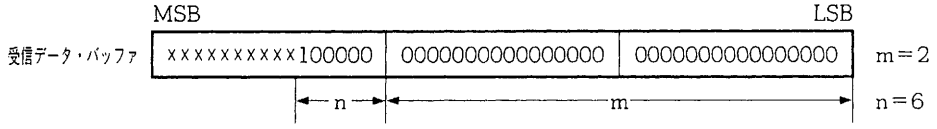
次の表の条件を満たさない場合，つまりFILL以外のコードを検出したあとにEOLを検出した場合については，FILL以外のコードの数値によってクロック数が異なってくるため，数値は算出していません。

外部バス幅	条　件	クロック数
8ビット・バス幅	先頭にEOLを検出した場合	24＋59m＋3n＋2mT
	先頭にFILL付きのEOLを検出した場合	
16ビット・バス幅	先頭にEOLを検出した場合	24＋56m＋3n＋mT
	先頭にFILL付きのEOLを検出した場合	

m：受信データ・バッファの指定された位置から最初に“1”を検出するまでのデータが，すべて“0”であるワード数

n：受信データ・バッファの指定された位置から最初に“1”を検出したワードの，“0”の連続ビット数＋1

例



T：受信データ・バッファが置かれている領域のウエイト数

保守／廃止

＜機　　能＞

復号化処理において，“EOL”(000000000001)を検出するための命令です。メモリ(受信バッファ)上の指定された位置から最初のEOLを検出します。データの先頭はLSBに入ります。

割り込み要求のサンプリングを行いません。したがって，割り込み要求のサンプリングを確実に行うには，SCHEOL命令が一定時間ごとに中断されるように，受信バッファ・サイズを小さく設定してください。これにより，長時間にわたる割り込み要求の保留を回避することができます。SCHEOL命令の実行時間（最長となる場合）の算出式を次に示します（リフレッシュ・サイクル，DMAサイクル，バス・ホールド要求は考慮していません)。

- 16ビット・バス幅のとき：（214＋T）N/16＋74＋T（クロック）
- ８ビット・バス幅のとき：（110＋T）N/8＋74＋T（クロック）

N：EOL検出までのビット数

T：受信バッファのウエイト数

復号化処理のEOLを検出する処理において，次のような場合が考えられます。

① 指定された受信バッファ内が，すべて“0”の場合（EOL検出処理の中断)。
② 一番先頭にEOLを検出する場合。
③ FILL付きEOLを先頭に検出する場合。
④ FILL以外の符号を検出したあと，EOLを検出する場合。

＜入力パラメータ＞

AL：レジスタ・バンク番号指定

保守／廃止

ALで指定されるレジスタ・バンク

00H	（受信バッファ・セグメント）　（DS2）	拡張セグメント・オーバライド・プリフィクス[注1]
02H	ワーク・エリア	
04H	処理中半端符号データ	
06H	処理中半端符号ビット数	
08H	受信バッファ・セグメント　（DS0）	
0AH	受信バッファ・オフセット　（ポインタ）	
0CH	受信バッファ・サイズ[注2]　（カウンタ）	
0EH	ワーク・エリア	
10H	ワーク・エリア	
12H	ワーク・エリア	
14H	ワーク・エリア	
16H	ワーク・エリア	
18H	ワーク・エリア	
1AH	ワーク・エリア	
1CH	ワーク・エリア	
1EH	0クリア[注3]/ワーク・エリア	…0クリア必要パラメータ

注1．DS2：を付加した場合，オフセット値は08Hの代わりに00Hが参照されます。

2．バイト単位。0以外の偶数値を設定してください。

3．新規命令処理時に0クリアを行います。
中断した処理を再開する場合，0クリアが必要なパラメータは中断時の状態のままにしておいてください。

＜出力パラメータ＞

Zフラグ	CYフラグ	受信バッファ残り	終了状態	AHステータス
0	0	あり	正常終了	01，02
0	1	あり	終了エラー	03
1	0	なし	正常終了	01，02
1	1	なし	終了エラー	00，03

AH内の数値によって終了状態をチェックすることができます。数値と終了状態を次に示します。

保守／廃止

AH	終　了　状　態
00H	中断
01H	先頭にEOLを検出
02H	FILL付きのEOLを先頭に検出
03H	EOL以外の符号を検出したあと，EOLを検出または受信バッファの残り容量がなくなったあと，EOLを検出

保守／廃止

18.6.6　1ビット検出命令

＜記　述　形　式＞　GETBIT (GET BIT)

＜命 令 語 形 式＞

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	0	1	1	1	1	0	0	1

＜バ　イ　ト　数＞　2

＜ク ロ ッ ク 数＞

外部バス幅	条　件	クロック数
8ビット・バス幅	処理中の半端符号ビット数≠0のとき	13
	処理中の半端符号ビット数＝0のとき	18＋2T
16ビット・バス幅	処理中の半端符号ビット数≠0のとき	13
	処理中の半端符号ビット数＝0のとき	15＋T

T：受信データ・バッファが置かれている領域のウエイト数

＜機　　能＞

復号化処理において，タグなどの1ビットを検出するための命令です。データの先頭と処理中の半端符号はLSBから入ります。

メモリ（受信バッファ）上の指定された位置から1ビットを取り出し，CYフラグに設定します。

割り込み要求のサンプリングを行いません。

保守／廃止

＜入力パラメータ＞

AL：レジスタ・バンク番号指定

ALで指定されるレジスタ・バンク

00H	（受信バッファ・セグメント）	（DS2）	拡張セグメント・オーバライド・プリフィクス注1
02H	ワーク・エリア		
04H	処理中半端符号データ		
06H	処理中半端符号ビット数		
08H	受信バッファ・セグメント	（DS0）	
0AH	受信バッファ・オフセット	（ポインタ）	
0CH	受信バッファ・サイズ注2	（カウンタ）	
0EH	ワーク・エリア		
10H	ワーク・エリア		
12H	ワーク・エリア		
14H	ワーク・エリア		
16H	ワーク・エリア		
18H	ワーク・エリア		
1AH	ワーク・エリア		
1CH	ワーク・エリア		
1EH	ワーク・エリア		

注1．DS2：を付加した場合，オフセット値は08Hの代わりに00Hが参照されます。

2．バイト単位。0以外の偶数値を設定してください。

＜出力パラメータ＞

CYフラグ：検出した1ビット・データ

Zフラグ	受信バッファ残り	
0	あり	…受信データ・バッファ・サイズ>0
1	なし	…受信データ・バッファ・サイズ=0

注意　半端符号データの有無にかかわらず，受信データ・バッファ・サイズのカウンタ（レジスタ・バンクの0CHの内容）が“0”のとき，Zフラグ＝1となります。

保守／廃止

18.6.7　MH復号化変化点テーブル作成命令

＜記　述　形　式＞　MHDEC（MH Decode）

＜命 令 語 形 式＞

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	0	1	1	1	1	1	0	0

＜バ　イ　ト　数＞　2

＜ク ロ ッ ク 数＞

不定（処理パターンにより，数値が異なるため）

＜機　　能＞

指定されたメモリ（受信バッファ）上の符号データを入力し，復号化変換テーブルによりMH復号化を行い，生成された変化点情報を指定されたメモリ（変化点テーブル）上に送出する命令です。データの先頭と処理中の半端符号はLSBから入ります。

変化点情報を生成することにより，１ライン分の画素データ・ビット数に達したことを認識し命令を終了とします。

MH復号化の符号データから変化点テーブルを作成する処理において，次のような場合が考えられます。

① 指定された受信バッファ内に１ライン分の符号データがあり，変化点情報への変換を終了する場合（正常終了）。

② 指定された受信バッファ内の符号が１ライン分に満たなくて，変化点情報への変換が途中で終了する場合（中断）。

③ １番先頭にEOLを検出する場合。

④ FILL付きのEOLを先頭に検出する場合。

⑤ １ライン分の画素数に満たないデータのあと，EOLを検出する場合。

⑥ １ライン分の画素数以上に復号化するデータがある場合。

⑦ 異常なコードを検出する場合。

正常終了のときだけ，変化点テーブルに“FFFFH”（１ライン分の終了コード）を出力します。

各色（白または黒）の変化点を生成する復号化処理ごとに，割り込み要求のサンプリングを行い

保守／廃止

ます。

MHDEC命令において，復号化できる1ラインの最大画素数は合計65535画素です。

MHDEC命令実行中に“0”の連続する異常符号を検出すると，次に“1”を検出するまで割り込み要求が保留されます。この場合，受信バッファ・サイズを小さく設定してMHDEC命令が一定の時間で中断されるようにすることにより，長時間にわたる割り込み要求の保留を回避することができます。MHDEC命令の実行時間（最長となる場合）の算出式を次に示します（リフレッシュ・サイクル，DMAサイクル，バス・ホールド要求は考慮していません）。

- 16ビット・バス幅のとき：40+2T+（59+T）N/16（クロック）
- 8ビット・バス幅のとき：44+4T+（37+2T）N/8（クロック）

N：“0”の連続するビット数

T：変化点テーブルのウエイト数

＜入力パラメータ＞

AL：レジスタ・バンク番号指定

ALで指定されるレジスタ・バンク

00H	（受信バッファ・セグメント）　（DS2）	拡張セグメント・オーバライド・プリフィクス注1
02H	（変化点テーブル・セグメント）　（DS3）	
04H	処理中半端符号データ	
06H	処理中半端符号ビット数	
08H	受信バッファ・セグメント　（DS0）	
0AH	受信バッファ・オフセット　（ポインタ）	
0CH	受信バッファ・サイズ注2　（カウンタ）	
0EH	変化点テーブル・セグメント　（DS1）	
10H	復号化ライン変化点テーブル・オフセット	
12H	1ライン分の総画素ビット数	
14H	復号化変換テーブル・セグメント	
16H	ワーク・エリア	
18H	ワーク・エリア	
1AH	ワーク・エリア	
1CH	0クリア注3/ワーク・エリア	0クリア必要パラメータ
1EH	0クリア注3/ワーク・エリア	

注1. DS2：，DS3：を付加した場合，オフセット値は08H，0EHの代わりにそれぞれ00H，02Hが参照されます。

2. バイト単位。0以外の偶数値を設定してください。

保守／廃止

注3. 新規命令処理時に０クリアを行います。
中断した処理を再開する場合，０クリアが必要なパラメータは中断時の状態のままにしておいてください。

＜出力パラメータ＞

Zフラグ	CYフラグ	受信バッファ残り	終了状態	AHステータス
0	0	あり	正常終了	変化しない
0	1	あり	終了エラー	**表18－4参照**
1	0	なし	正常終了	変化しない
1	1	なし	終了エラー	**表18－4参照**

AH内の数値によって終了状態をチェックすることができます。数値と終了状態を次に示します。

なお，AH＝01H，02Hの場合は，ALレジスタで指定されるレジスタ・バンクのオフセット・アドレス1AH番地のワーク・エリアの値により，終了状態がそれぞれ２通りあります。

表18－4　MHDEC命令の終了状態

AH	終了状態	
00H	中断	
01H	ワーク・エリア＝0	先頭にEOLを検出
	ワーク・エリア≠0	先頭にメークアップ符号＋EOLを検出
02H	ワーク・エリア＝0	先頭にFILL付きEOLを検出
	ワーク・エリア≠0	先頭にメークアップ符号＋FILL付きEOLを検出
03H	１ライン分の画素数に満たないデータのあと，EOLを検出	
04H	１ライン分の画素数以上に復号化するデータが存在	
05H	異常なコードを検出	

備考　異常コードを検出して終了した場合，復号化ライン変化点テーブルのオフセットは，最後に正常出力された変化点テーブル・データの次のワード領域のアドレスを指しています。

保守／廃止

18.6.8　MR復号化変化点テーブル作成命令

<記　述　形　式>　MRDEC（MR Decode）

<命 令 語 形 式>

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	0	1	1	1	1	1	0	1

<バ　イ　ト　数>　2

<ク ロ ッ ク 数>

不定（処理パターンにより，数値が異なるため）

<機　　能>

指定されたメモリ（受信バッファ）上の復号ラインの符号データと参照ラインの変化点情報を入力し，復号化変換テーブルによりMR復号化を行い，生成された復号化ラインの変化点情報を指定されたメモリ（変化点テーブル）上に送出する命令です。データの先頭と処理中の半端符号はLSBから入ります。

変化点情報を生成することにより，１ライン分の画素データ・ビット数に達したことを認識し命令を終了とします。

MR復号化の符号データから変化点テーブルを作成する処理において，次のような場合が考えられます。

① 指定された受信バッファ内に１ライン分の符号データがあり，変化点情報への変換を終了する場合（正常終了）。
② 指定された受信バッファ内の符号が１ライン分に満たなくて，変化点情報への変換が途中で終了する場合（中断）。
③ １番先頭にEOLを検出する場合。
④ FILL付きのEOLを先頭に検出する場合。
⑤ １ライン分の画素数に満たないデータのあと，EOLを検出する場合。
⑥ １ライン分の画素数以上に復号化するデータがある場合。
⑦ 異常なコードを検出する場合。

正常終了のときだけ，変化点テーブルに“FFFFH”（１ライン分の終了コード）を出力します。

保守／廃止

各モード（パス・モード，垂直モード，水平モード）単位の復号化処理ごとに，割り込み要求のサンプリングを行います。

MRDEC命令において，MR復号化できる最大画素数は合計32767画素です。

MRDEC命令実行中に“0”の連続する異常符号を検出すると，次に“1”を検出するまで割り込み要求が保留されます。この場合，受信バッファ・サイズを小さく設定してMRDEC命令が一定の時間で中断されるようにすることにより，長時間にわたる割り込み要求の保留を回避することができます。MRDEC命令の実行時間（最長となる場合）の算出式を次に示します（リフレッシュ・サイクル，DMAサイクル，バス・ホールド要求は考慮していません）。

- 16ビット・バス幅のとき：211＋6T＋（59＋T）N/16（クロック）
- 8ビット・バス幅のとき：227＋12T＋（37＋2T）N/8（クロック）

N：“0”の連続するビット数

T：変化点テーブルのウエイト数

＜入力パラメータ＞

AL：レジスタ・バンク番号指定

ALで指定されるレジスタ・バンク

00H	（受信バッファ・セグメント）	（DS2）	拡張セグメント・オーバライド・プリフィクス注1
02H	（変化点テーブル・セグメント）	（DS3）	
04H	処理中半端符号データ		
06H	処理中半端符号ビット数		
08H	受信バッファ・セグメント	（DS0）	
0AH	受信バッファ・オフセット	（ポインタ）	
0CH	受信バッファ・サイズ注2	（カウンタ）	
0EH	変化点テーブル・セグメント	（DS1）	
10H	復号化ライン変化点テーブル・オフセット		
12H	参照ライン変化点テーブル・オフセット		
14H	復号化変換テーブル・セグメント		
16H	ワーク・エリア		
18H	ワーク・エリア		
1AH	ワーク・エリア		
1CH	0クリア注3/ワーク・エリア		0クリア必要パラメータ
1EH	0クリア注3/ワーク・エリア		

注1. DS2：，DS3：を付加した場合，オフセット値は08H，0EHの代わりにそれぞれ00H，02Hが参照されます。

2. バイト単位。0以外の偶数値を設定してください。

保守／廃止

注3. 新規命令処理時に0クリアを行います。
中断した処理を再開する場合,0クリアが必要なパラメータは中断時の状態のままにしておいてください。

＜出力パラメータ＞

Zフラグ	CYフラグ	受信バッファ残り	終了状態	AHステータス
0	0	あり	正常終了	変化しない
0	1	あり	終了エラー	**表18－5参照**
1	0	なし	正常終了	変化しない
1	1	なし	終了エラー	**表18－5参照**

AH内の数値によって終了状態をチェックすることができます。数値と終了状態を次に示します。

表18－5　MRDEC命令の終了状態

AH	終　了　状　態
00H	中断
01H	先頭にEOLを検出
02H	FILL付きのEOLを先頭に検出
03H	1ライン分の画素数に満たないデータのあと，EOLを検出
04H	1ライン分の画素数以上に復号するデータが存在
05H	異常なコードを検出

備考　異常コードを検出して終了した場合,復号化ライン変化点テーブルのオフセットは，最後に正常出力された変化点テーブル・データの次のワード領域のアドレスを指しています。

保守／廃止

18.6.9　画素データ作成命令

<記　述　形　式>　CNVTRP（Convert Turning Point）

<命 令 語 形 式>

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	1	0	1	1	1	1	0	1	0

<バ　イ　ト　数>　2

<ク ロ ッ ク 数>

CNVTRP命令実行中に割り込みによる中断がない場合のクロック数を示します。

なお，割り込みによる中断があった場合は，どの変化点情報まで画素データに変換する処理を終了したかによってクロック数が異なってくるため，数値は算出していません。

外部バス幅	条　件	クロック数
8ビット・バス幅	変化点情報の個数（n）が偶数の場合 （変化点が黒で終わる場合）	(34+53n)/2+14m+8k+ 2（m+k）T+2（n+1）t
	変化点情報の個数（n）が奇数の場合 （変化点が白で終わる場合）	(33+53n)/2+14m+8k+ 2（m+k）T+2（n+1）t
16ビット・バス幅	変化点情報の個数（n）が偶数の場合 （変化点が黒で終わる場合）	(28+47n)/2+11m+5k+（m+ k）T+（n+1）t
	変化点情報の個数（n）が奇数の場合 （変化点が白で終わる場合）	(27+47n)/2+11m+5k+（m+ k）T+（n+1）t

k：全白／全黒の画素データを送出する回数

m：全白／全黒以外の画素データを送出する回数

n：変化点情報の個数（白の変化点情報の個数と黒の変化点情報の個数の和）

T：画素データ・バッファが置かれている領域のウエイト数

t：変化点テーブルが置かれている領域のウエイト数

保守／廃止

＜機　　能＞

指定されたメモリ（変化点テーブル）上の１ライン分の変化点情報を入力し，画素データに変換し，指定されたメモリ（プリント・バッファ）上に１ワード（16ビット）単位で送出する命令です。データの先頭はLSBに入ります。

各色（白または黒）の変化点情報（変化点テーブルの１ワード）を画素データに変換する処理ごとに割り込み要求のサンプリングを行います。

なお，16の倍数以外の画素データ数を示す変化点テーブルを使用した場合，最後の16ビットに満たない画素データは送出されません。

＜入力パラメータ＞

AL：レジスタ・バンク番号指定

ALで指定されるレジスタ・バンク

00H	（変化点テーブル・セグメント）	（DS2）	拡張セグメント・オーバライド・プリフィクス注1
02H	（プリント・バッファ・セグメント）	（DS3）	
04H	ワーク・エリア		
06H	ワーク・エリア		
08H	変化点テーブル・セグメント	（DS0）	
0AH	変化点テーブル・オフセット	（ポインタ）	
0CH	ワーク・エリア		
0EH	プリント・バッファ・セグメント	（DS1）	
10H	プリント・バッファ・オフセット	（ポインタ）	
12H	ワーク・エリア		
14H	ワーク・エリア		
16H	ワーク・エリア		
18H	ワーク・エリア		
1AH	処理中の半端画素データ		
1CH	０クリア注2/処理中の半端画素ビット数		０クリア必要パラメータ
1EH	０クリア注2/ワーク・エリア		

注1．DS2：，DS3：を付加した場合，オフセット値は08H，0EHの代わりにそれぞれ00H，02Hが参照されます。

2．新規命令処理時に０クリアを行います。

注意　中断処理はありません。

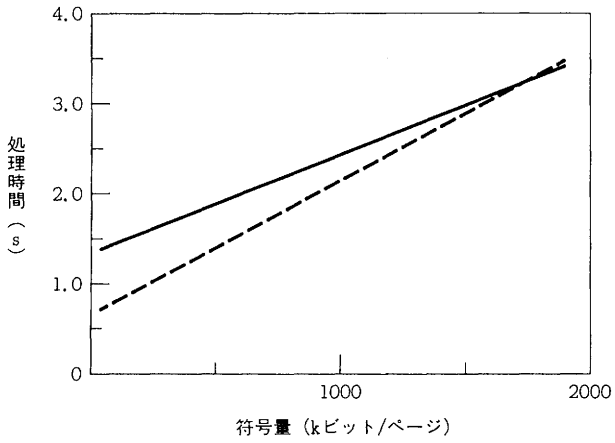
保守／廃止

18.7　処理時間

コーデック命令を使用した場合の，符号化／復号化命令の符号量当たりの処理時間を次に示します。

図18－22　MH符号化／復号化命令の処理時間

(1) 8ビット・バス幅，1ウエイトの場合

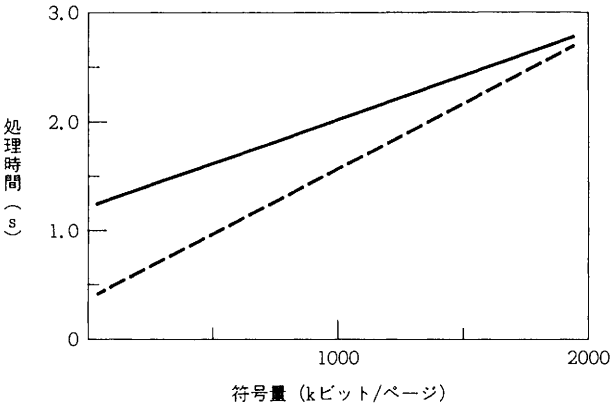


備考1. ——— 符号化命令

----- 復号化命令

2. 測定条件は12.5 MHzです。

(2) 16ビット・バス幅，ノー・ウエイトの場合



備考1. ——— 符号化命令

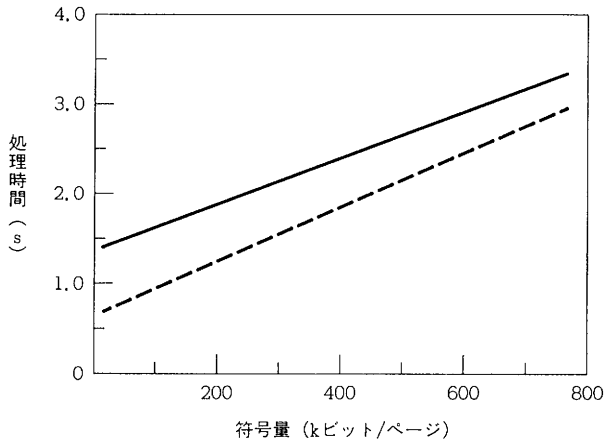
----- 復号化命令

2. 測定条件は12.5 MHzです。

保守／廃止

図18－23　MR符号化／復号化命令の処理時間

（1）8ビット・バス幅，1ウエイトの場合

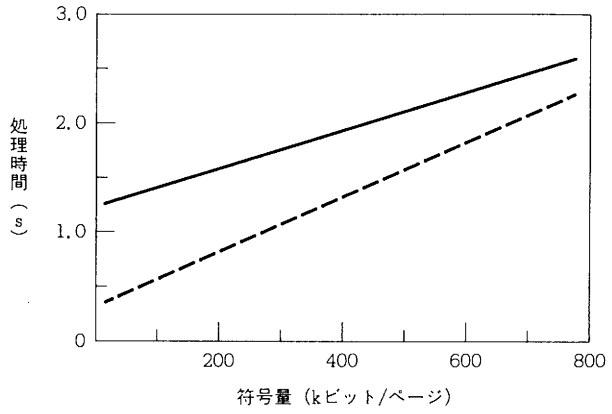


備考1. ——— 符号化命令

----- 復号化命令

2. 測定条件は12.5 MHz，Kファクタ＝4 です。

（2）16ビット・バス幅，ノー・ウエイトの場合



備考1. ——— 符号化命令

----- 復号化命令

2. 測定条件は12.5 MHz，Kファクタ＝4 です。

〔メ　モ〕

保守／廃止

第19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・リセットがかかります。リセットがかかると，そのとき実行されている動作の終了を待たずにリセット状態へ移行します。$\overline{\text{RESET}}$端子にロウ・レベルが入力されてから各端子の状態がリセット中の状態に移行するまでの時間は1 μs(MAX.)です。各ハードウエア関係のレジスタは表19－1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，プログラム中で必要に応じてイニシャライズしてください。V55PIの性能を引き出すためには，最低限PRCレジスタとRFMレジスタ，MBCレジスタおよびPWCレジスタをそれぞれアプリケーションに応じた値に設定する必要があります。

表19－1　ハードウエアのリセット後の状態（1/4）

ハードウエア（略号）			アドレス	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	—	不定
	セグメント・レジスタ	SS, DS0-DS3	—	0000H
		PS	—	FFFFH
ポート	ポート・レジスタ	P0-P8	FFF00H-FFF08H	不定
	ポート・モード・レジスタ	PM0 PM2-PM5 PM7, PM8	FFF10H FFF12H-FFF15H FFF17H, FFF18H	FFH
	ポート・モード・ コントロール・レジスタ	PMC2-PMC5 PMC7, PMC8	FFF22H-FFF25H FFF27H, FFF28H	00H
	ポート・リード・ コントロール・レジスタ	PRDC	FFF0CH	00H
リアルタイム出力ポート	リアルタイム出力ポート・ コントロール・レジスタ	RTPC	FFF2CH	40H
	リアルタイム出力ポート・ ディレイ指定レジスタ	RTPD	FFF2DH	不定
	ポート7バッファ	P7H, P7L	FFF2EH-FFF2FH	不定
	リアルタイム出力ポート	RTP	FFF0EH	不定

保守／廃止

表19－1　ハードウエアのリセット後の状態（2/4）

ハードウエア（略号）				アドレス	リセット後の状態
タイマ・ユニット	タイマ・レジスタ	TM0-TM3		FFF40H-FFF47H	00H
	タイマ・キャプチャ・レジスタ	CT00, CT01 CT10		FFF48H-FFF4BH FFF50H-FFF51H	不定
	タイマ・コンペア・レジスタ	CM00, CM01 CM10, CM11 CM20, CM21 CM22, CM23 CM30, CM31		FFF4CH-FFF4FH FFF52H-FFF55H FFF58H-FFF5BH FFF5CH-FFF5FH FFF64H-FFF67H	不定
	タイマ・コントロール・レジスタ	TMC	TMC0 TMC1	FFF30H FFF31H	00H
	タイマ出力制御レジスタ	TOC	TOC0 TOC1	FFF32H FFF33H	00H
ソフトウエア・インターバル・タイマ	ソフトウエア・タイマ・カウンタ	STC		FFFE0H-FFFE1H	不定
	ソフトウエア・タイマ・カウンタ・コンペア・レジスタ	STMC		FFFE2H-FFFE3H	FFFFH
PWMユニット	PWMレジスタ	PWM		FFF6CH	不定
	PWMコントロール・レジスタ	PWMC		FFF6DH	00H
DMAコントローラ	DMAモード・レジスタ	DMAM0, DMAM1		FFF9CH, FFFBCH	E0H
	DMAコントロール・レジスタ	DMAC0, DMAC1		FFF9DH, FFFBDH	00H
	ターミナル・カウンタ	TC0	TC0L TC0H	FFF80H, FFF81H FFF82H	不定
		TC1	TC1L TC1H	FFFA0H, FFFA1H FFFA2H	
	ターミナル・カウンタ・モジュロ・レジスタ	TCM0	TCM0L TCM0H	FFF84H, FFF85H FFF86H	不定
		TCM1	TCM1L TCM1H	FFFA4H, FFFA5H FFFA6H	
	DMAメモリ・アドレス・レジスタ	MAR0	MAR0L MAR0H	FFF90H, FFF91H FFF92H	不定
		MAR1	MAR1L MAR1H	FFFB0H, FFFB1H FFFB2H	
	DMAアップ・ダウン・カウンタ	UDC0	UDC0L UDC0H	FFF88H, FFF89H FFF8AH	不定
		UDC1	UDC1L UDC1H	FFFA8H, FFFA9H FFFAAH	

保守／廃止

表19－1　ハードウエアのリセット後の状態（3/4）

ハードウエア（略号）				アドレス	リセット後の状態
DMA コントローラ	DMAコンペア・レジスタ	DCM0	DCM0L	FFF8CH，FFF8DH	不定
			DCM0H	FFF8EH	
		DCM1	DCM1L	FFFACH，FFFADH	
			DCM1H	FFFAEH	
	DMAリード・ライト・ポインタ	DPTC0	DPTC0L	FFF94H，FFF95H	不定
			DPTC0H	FFF96H	
		DPTC1	DPTC1L	FFFB4H，FFFB5H	
			DPTC1H	FFFB6H	
	DMAステータス・レジスタ	DMAS		FFF9EH	00H
A/D コンバータ	A/Dコンバータ・モード・レジスタ	ADM		FFE20H	00H
	A/Dコンバージョン・リザルト・レジスタ	ADCR0		FFE00H	不定
		ADCR1		FFE02H	
		ADCR2		FFE04H	
		ADCR3		FFE06H	
パラレル・インタフェース	パラレル・インタフェース・バッファ	PAB		FFE10H	不定
	パラレル・インタフェース・コントロール・レジスタ	PAC0		FFE18H	90H
		PAC1		FFE19H	03H
	パラレル・インタフェース・ステータス・レジスタ	PAS		FFE1AH	40H
	パラレル・インタフェース・アクノリッジ・インターバル・レジスタ	PAI1		FFE1CH	不定
		PAI2		FFE1DH	
UART/CSI	プロトコル選択レジスタ	ASP		FFF7FH	00H
	UARTモード・レジスタ／CSIモード・レジスタ	UARTM0/CSIM0		FFF73H	00H
		UARTM1/CSIM1		FFF7BH	
	UARTステータス・レジスタ／SBIコントロール・レジスタ	UARTS0/SBIC0		FFF74H	20H/00H
		UARTS1		FFF7CH	20H
	受信バッファ	RxB0		FFF76H	不定
		RxB1		FFF7EH	
	UART送信バッファ／CSIシフト・レジスタ	TxB0/SIO0		FFF75H	不定
		TxB1/SIO1		FFF7DH	
	受信ボー・レート・ジェネレータ・レジスタ	RxBRG0		FFF71H	不定
		RxBRG1		FFF79H	
	送信ボー・レート・ジェネレータ・レジスタ	TxBRG0		FFF70H	不定
		TxBRG1		FFF78H	

保守／廃止

表19－1　ハードウエアのリセット後の状態（4/4）

ハードウエア（略号）				アドレス	リセット後の状態
UART/CSI	プリスケーラ・レジスタ	PRS0		FFF72H	00H
		PRS1		FFF7AH	
割り込み	割り込みモード・コントロール・レジスタ	IMC		FFEC5H	80H
	割り込みマスク・フラグ・レジスタ	MK0	MK0L	FFEC0H	FFH
			MK0H	FFEC1H	
		MK1	MK1L	FFEC2H	
			MK1H	FFEC3H	
	割り込み要求制御レジスタ	IC09-IC14		FFEC9H-FFECEH	43H
		IC16-IC32		FFED0H-FFEE1H	
		IC36-IC37		FFEE4H-FFEE5H	
	インサービス・プライオリティ・レジスタ	ISPR		FFEC4H	00H
	外部割り込みモード・レジスタ	INTM	INTM0	FFF34H	00H
			INTM1	FFF35H	
スタンバイ・コントロール・レジスタ		STBC		FFFEEH	不定[注]
プロセッサ・コントロール・レジスタ		PRC		FFFEFH	EEH
プログラマブル・ウエイト・コントロール・レジスタ		PWC0		FFFE8H	EAH
		PWC1		FFFE9H	AAH
リフレッシュ・モード・レジスタ		RFM		FFFECH	77H
メモリ・ブロック・コントロール・レジスタ		MBC		FFFEAH	FCH
ウォッチドッグ・タイマ・モード・レジスタ		WDM		FFF60H	00H

注　パワーオン・リセット時：00H，その他：変化しません。

保守／廃止

付録A　ニモニック一覧

ニモニック一覧を表A-1に示します。

保守／廃止

表A－1 ニモニック一覧（1/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
データ転送命令	MOV	reg, reg′	1	0	0	0	1	0	1	W	1	1	reg				reg′		2	reg ←→ reg′						
		mem, reg	1	0	0	0	1	0	0	W	mod		reg			mem			2-4	(mem) ←→ reg						
		reg, mem	1	0	0	0	1	0	1	W	mod		reg			mem			2-4	reg ← (mem)						
		mem, imm	1	1	0	0	0	1	1	W	mod	0	0	0		mem			3-6	(mem) ← imm						
		reg, imm	1	0	1	1	W		reg										2-3	reg ← imm						
		acc, dmem	1	0	1	0	0	0	0	W									3	W = 0のとき AL ← (dmem) W = 1のとき AH ← (dmem + 1), AL ← (dmem)						
		dmem, acc	1	0	1	0	0	0	1	W									3	W = 0のとき (dmem) ← AL W = 1のとき (dmem + 1) ← AH, (dmem) ← AL						
		sreg, reg16	1	0	0	0	1	1	1	0	1	1	0	sreg			reg		2	sreg ← reg16　　sreg：SS, DS0, DS1						
		sreg, mem16	1	0	0	0	1	1	1	0	mod	0	sreg			mem			2-4	sreg ← (mem16)　　sreg：SS, DS0, DS1						
		reg16, sreg	1	0	0	0	1	1	0	0	1	1	0	sreg			reg		2	reg16 ← sreg						
		mem16, sreg	1	0	0	0	1	1	0	0	mod	0	sreg			mem			2-4	(mem16) ← sreg						
		DS0, reg16, mem32	1	1	0	0	0	1	0	1	mod		reg			mem			2-4	reg16 ← (mem32) DS0 ← (mem32 + 2)						
		DS1, reg16, mem32	1	1	0	0	0	1	0	0	mod		reg			mem			2-4	reg16 ← (mem32) DS1 ← (mem32 + 2)						
		AH, PSW	1	0	0	1	1	1	1	1									1	AH ← S, Z, 0, AC, 0, P, $\overline{\text{IBRK}}$, CY						
		PSW, AH	1	0	0	1	1	1	1	0									1	S, Z, 0, AC, 0, P, $\overline{\text{IBRK}}$, CY ← AH	×	×		×	×	×

保守／廃止

表A-1 ニモニック一覧 (2/33)

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
データ転送命令	LDEA	reg16, mem16	1	0	0	0	1	1	0	1	mod		reg			mem			2-4	reg16 ← mem16						
	TRANS TRANSB[注1]	(src-table)	1	1	0	1	0	1	1	1									1	AL ← (BW + AL)						
	XCH	reg, reg'	1	0	0	0	0	1	1	W	1	1	reg			reg'			2	reg ↔ reg'						
		mem, reg reg, mem	1	0	0	0	0	1	1	W	mod		reg			mem			2-4	(mem) ↔ reg						
		AW, reg16 reg16, AW	1	0	0	1	0	reg											1	AW ↔ reg16						
	MOVSPA[注2]		0	0	0	0	1	1	1	1	0	0	1	0	0	1	0	1	2	新レジスタ・バンクSS, SP ← 旧レジスタ・バンクのSS, SP						
	MOVSPB[注2]	reg16	0	0	0	0	1	1	1	1	1	0	0	1	0	1	0	1	3	reg16で示されるレジスタ・バンクのSS, SP ← 現在のレジスタ・バンクのSS, SP						
			1	1	1	1	1	reg																		

注1. TRANS命令の場合は，オペランド省略可。TRANSB命令の場合は，オペランドなし。

2. V20, V30に対して，追加した命令

保守／廃止

表A－1 ニモニック一覧（3/33）

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
リピート・プリフィクス	REPC		0 1 1 0 0 1 0 1		1	CW ≠ 0の間，続くバイトのプリミティブ・ブロック転送命令を実行し，CWをデクリメント(－1)する。 保留割り込みがあれば処理する。 CY ≠ 1のときループを抜け出る。						
	REPNC		0 1 1 0 0 1 0 0		1	同上 CY ≠ 0のときループを抜け出る。						
	REP REPE REPZ		1 1 1 1 0 0 1 1		1	CW ≠ 0の間，続くバイトのプリミティブ・ブロック転送命令を実行し，CWをデクリメント(－1)する。 保留割り込みがあれば処理する。 プリミティブ・ブロック転送命令がCMPBKまたはCMPMでかつ，Z ≠ 1のときループを抜け出る。						
	REPNE REPNZ		1 1 1 1 0 0 1 0		1	同上 Z ≠ 0のときループを抜け出る。						
プリミティブ・ブロック転送命令	MOVBK MOVBKB MOVBKW	dst-block, src-block	1 0 1 0 0 1 0 W		1	W = 0のとき (IY) ← (IX) DIR = 0 : IX ← IX + 1, IY ← IY + 1 DIR = 1 : IX ← IX － 1, IY ← IY － 1 W = 1のとき (IY + 1, IY) ← (IX + 1, IX) DIR = 0 : IX ← IX + 2, IY ← IY + 2 DIR = 1 : IX ← IX － 2, IY ← IY － 2						
	CMPBK CMPBKB CMPBKW	src-block, dst-block	1 0 1 0 0 1 1 W		1	W = 0のとき (IX) － (IY) DIR = 0 : IX ← IX + 1, IY ← IY + 1 DIR = 1 : IX ← IX － 1, IY ← IY － 1 W = 1のとき (IX + 1, IX) － (IY + 1, IY) DIR = 0 : IX ← IX + 2, IY ← IY + 2 DIR = 1 : IX ← IX － 2, IY ← IY － 2	×	×	×	×	×	×

保守／廃止

表A-1 ニモニック一覧 (4/33)

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
プリミティブ・ブロック転送命令	CMPM CMPMB CMPMW	dst-block	1 0 1 0 1 1 1 W		1	W = 0のとき AL − (IY) DIR = 0 : IY ← IY + 1, DIR = 1 : IY ← IY − 1 W = 1のとき AW − (IY + 1, IY) DIR = 0 : IY ← IY + 2, DIR = 1 : IY ← IY − 2	×	×	×	×	×	×
	LDM LDMB LDMW	src-block	1 0 1 0 1 1 0 W		1	W = 0のとき AL ← (IX) DIR = 0 : IX ← IX + 1, DIR = 1 : IX ← IX − 1 W = 1のとき AW ← (IX + 1, IX) DIR = 0 : IX ← IX + 2, DIR = 1 : IX ← IX − 2						
	STM STMB STMW	dst-block	1 0 1 0 1 0 1 W		1	W = 0のとき (IY) ← AL DIR = 0 : IY ← IY + 1, DIR = 1 : IY ← IY − 1 W = 1のとき (IY + 1, IY) ← AW DIR = 0 : IY ← IY + 2, DIR = 1 : IY ← IY − 2						
ビット・フィールド操作命令	INS	reg8, reg8′	0 0 0 0 1 1 1 1 1 1 reg′ reg	0 0 1 1 0 0 0 1	3	16ビット・フィールド ← AW						
		reg8, imm4	0 0 0 0 1 1 1 1 1 1 0 0 0 reg	0 0 1 1 1 0 0 1	4	16ビット・フィールド ← AW						
	EXT	reg8, reg8′	0 0 0 0 1 1 1 1 1 1 reg′ reg	0 0 1 1 0 0 1 1	3	AW ← 16ビット・フィールド						
		reg8, imm4	0 0 0 0 1 1 1 1 1 1 0 0 0 reg	0 0 1 1 1 0 1 1	4	AW ← 16ビット・フィールド						

保守／廃止

表A－1 ニモニック一覧（5/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
入出力命令	IN注	acc, imm8	1	1	1	0	0	1	0	W									2	W＝0のとき AL←(imm8) W＝1のとき AH←(imm8＋1), AL←(imm8)						
		acc, DW	1	1	1	0	1	1	0	W									1	W＝0のとき AL←(DW) W＝1のとき AH←(DW＋1), AL←(DW)						
	OUT注	imm8, acc	1	1	1	0	0	1	1	W									2	W＝0のとき (imm8)←AL W＝1のとき (imm8＋1)←AH, (imm8)←AL						
		DW, acc	1	1	1	0	1	1	1	W									1	W＝0のとき (DW)←AL W＝1のとき (DW＋1)←AH, (DW)←AL						
プリミティブ入出力命令	INM注	dst-block, DW	0	1	1	0	1	1	0	W									1	W＝0のとき (IY)←(DW) DIR＝0：IY←IY＋1, DIR＝1：IY←IY－1 W＝1のとき (IY＋1, IY)←(DW＋1, DW) DIR＝0：IY←IY＋2, DIR＝1：IY←IY－2						
	OUTM注	DW, src-block	0	1	1	0	1	1	1	W									1	W＝0のとき (DW)←(IX) DIR＝0：IX←IX＋1, DIR＝1：IX←IX－1 W＝1のとき (DW＋1, DW)←(IX＋1, IX) DIR＝0：IX←IX＋2, DIR＝1：IX←IX－2						

注 $\overline{IBRK}$ ＝0のとき，ソフトウエア割り込みが自動的に発生し，命令は実行されません。

保守／廃止

表A－1　ニモニック一覧（6/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
加減算命令	ADD	reg, reg′	0	0	0	0	0	0	1	W	1	1	reg				reg′		2	reg ← reg + reg′	×	×	×	×	×	×
		mem, reg	0	0	0	0	0	0	0	W	mod		reg				mem		2-4	(mem) ← (mem) + reg	×	×	×	×	×	×
		reg, mem	0	0	0	0	0	0	1	W	mod		reg				mem		2-4	reg ← reg + (mem)	×	×	×	×	×	×
		reg, imm	1	0	0	0	0	0	S	W	1	1	0	0	0		reg		3-4	reg ← reg + imm	×	×	×	×	×	×
		mem, imm	1	0	0	0	0	0	S	W	mod		0	0	0		mem		3-6	(mem) ← (mem) + imm	×	×	×	×	×	×
		acc, imm	0	0	0	0	0	1	0	W									2-3	W = 0のとき　AL ← AL + imm	×	×	×	×	×	×
	ADDC	reg, reg′	0	0	0	1	0	0	1	W	1	1	reg				reg′		2	reg ← reg + reg′ + CY	×	×	×	×	×	×
		mem, reg	0	0	0	1	0	0	0	W	mod		reg				mem		2-4	(mem) ← (mem) + reg + CY	×	×	×	×	×	×
		reg, mem	0	0	0	1	0	0	1	W	mod		reg				mem		2-4	reg ← reg + (mem) + CY	×	×	×	×	×	×
		reg, imm	1	0	0	0	0	0	S	W	1	1	0	1	0		reg		3-4	reg ← reg + imm + CY	×	×	×	×	×	×
		mem, imm	1	0	0	0	0	0	S	W	mod		0	1	0		mem		3-6	(mem) ← (mem) + imm + CY	×	×	×	×	×	×
		acc, imm	0	0	0	1	0	1	0	W									2-3	W = 0のとき　AL ← AL + imm + CY W = 1のとき　AW ← AW + imm + CY	×	×	×	×	×	×
	SUB	reg, reg′	0	0	1	0	1	0	1	W	1	1	reg				reg′		2	reg ← reg − reg′	×	×	×	×	×	×
		mem, reg	0	0	1	0	1	0	0	W	mod		reg				mem		2-4	(mem) ← (mem) − reg	×	×	×	×	×	×
		reg, mem	0	0	1	0	1	0	1	W	mod		reg				mem		2-4	reg ← reg − (mem)	×	×	×	×	×	×
		reg, imm	1	0	0	0	0	0	S	W	1	1	1	0	1		reg		3-4	reg ← reg − imm	×	×	×	×	×	×
		mem, imm	1	0	0	0	0	0	S	W	mod		1	0	1		mem		3-6	(mem) ← (mem) − imm	×	×	×	×	×	×
		acc, imm	0	0	1	0	1	1	0	W									2-3	W = 0のとき　AL ← AL − imm W = 1のとき　AW ← AW − imm	×	×	×	×	×	×

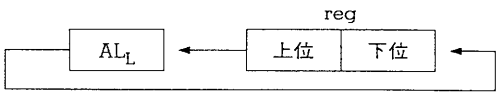
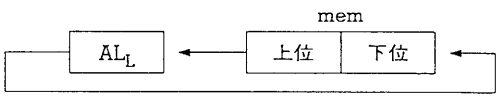
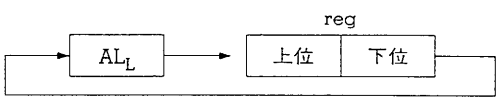
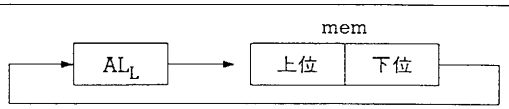
保守／廃止

表A-1　ニモニック一覧（7/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
加減算命令	SUBC	reg, reg'	0	0	0	1	1	0	1	W	1	1	reg				reg'		2	reg ← reg − reg' − CY	×	×	×	×	×	×
		mem, reg	0	0	0	1	1	0	0	W	mod		reg			mem			2-4	(mem) ← (mem) − reg − CY	×	×	×	×	×	×
		reg, mem	0	0	0	1	1	0	1	W	mod		reg			mem			2-4	reg ← reg − (mem) − CY	×	×	×	×	×	×
		reg, imm	1	0	0	0	0	0	S	W	1	1	0	1	1	reg			3-4	reg ← reg − imm − CY	×	×	×	×	×	×
		mem, imm	1	0	0	0	0	0	S	W	mod		0	1	1	mem			3-6	(mem) ← (mem) − imm − CY	×	×	×	×	×	×
		acc, imm	0	0	0	1	1	1	0	W									2-3	W = 0のとき　AL ← AL − imm − CY W = 1のとき　AW ← AW − imm − CY	×	×	×	×	×	×

保守／廃止

表A－1 ニモニック一覧（8/33）

命令群	ニモニック	オペランド	オペレーション・コード 7 6 5 4 3 2 1 0	オペレーション・コード 7 6 5 4 3 2 1 0	バイト数	オペレーション	フラグ AC	CY	V	P	S	Z
BCD演算命令	ADD4S[注1]	(dst-string, src-string)	0 0 0 0 1 1 1 1	0 0 1 0 0 0 0 0	2	dst BCD string ← dst BCD string + src BCD string[注2]	U	×	U	U	U	×
	SUB4S[注1]	(dst-string, src-string)	0 0 0 0 1 1 1 1	0 0 1 0 0 0 1 0	2	dst BCD string ← dst BCD string − src BCD string[注2]	U	×	U	U	U	×
	CMP4S[注1]	(dst-string, src-string)	0 0 0 0 1 1 1 1	0 0 1 0 0 1 1 0	2	dst BCD string − src BCD string[注2]	U	×	U	U	U	×
	ROL4	reg8	0 0 0 0 1 1 1 1 1 1 0 0 0 reg	0 0 1 0 1 0 0 0	3							
		mem8	0 0 0 0 1 1 1 1 mod 0 0 0 mem	0 0 1 0 1 0 0 0	3-5							
	ROR4	reg8	0 0 0 0 1 1 1 1 1 1 0 0 0 reg	0 0 1 0 1 0 1 0	3							
		mem8	0 0 0 0 1 1 1 1 mod 0 0 0 mem	0 0 1 0 1 0 1 0	3-5							

備考 nはBCD桁数の1/2です。

注1. オペランドの省略可能

2. BCD桁数はCLレジスタで与えられ，1-254の値が設定可能です。

保守／廃止

表A－1 ニモニック一覧（9/33）

命令群	ニモニック	オペランド	オペレーション・コード																バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
増減命令	INC	reg8	1	1	1	1	1	1	1	0	1	1	0	0	0	reg			2	reg8 ← reg8 + 1	×		×	×	×	×
		mem	1	1	1	1	1	1	1	W	mod		0	0	0	mem			2-4	(mem) ← (mem) + 1	×		×	×	×	×
		reg16	0	1	0	0	0	reg											1	reg16 ← reg16 + 1	×		×	×	×	×
	DEC	reg8	1	1	1	1	1	1	1	0	1	1	0	0	1	reg			2	reg8 ← reg8 − 1	×		×	×	×	×
		mem	1	1	1	1	1	1	1	W	mod		0	0	1	mem			2-4	(mem) ← (mem) − 1	×		×	×	×	×
		reg16	0	1	0	0	1	reg											1	reg16 ← reg16 − 1	×		×	×	×	×
乗算命令	MULU	reg8	1	1	1	1	0	1	1	0	1	1	1	0	0	reg			2	AW ← AL × reg8 AH = 0 : CY ← 0, V ← 0 AH ≠ 0 : CY ← 1, V ← 1	U	×	×	U	U	U
		mem8	1	1	1	1	0	1	1	0	mod		1	0	0	mem			2-4	AW ← AL × (mem8) AH = 0 : CY ← 0, V ← 0 AH ≠ 0 : CY ← 1, V ← 1	U	×	×	U	U	U
		reg16	1	1	1	1	0	1	1	1	1	1	1	0	0	reg			2	DW, AW ← AW × reg16 DW = 0 : CY ← 0, V ← 0 DW = 1 : CY ← 1, V ← 1	U	×	×	U	U	U
		mem16	1	1	1	1	0	1	1	1	mod		1	0	0	mem			2-4	DW, AW ← AW × (mem16) DW = 0 : CY ← 0, V ← 0 DW = 1 : CY ← 1, V ← 1	U	×	×	U	U	U

保守／廃止

表A－1 ニモニック一覧（10/33）

命令群	ニモニック	オペランド	オペレーション・コード 7 6 5 4 3 2 1 0	オペレーション・コード 7 6 5 4 3 2 1 0	バイト数	オペレーション	フラグ AC	CY	V	P	S	Z
乗算命令	MUL	reg8	1 1 1 1 0 1 1 0	1 1 1 0 1 reg	2	AW ← AL × reg8 AH = ALのサイン拡張：CY ← 0, V ← 0 AH ≠ ALのサイン拡張：CY ← 1, V ← 1	U	×	×	U	U	U
		mem8	1 1 1 1 0 1 1 0	mod 1 0 1 mem	2-4	AW ← AL × (mem8) AH = ALのサイン拡張：CY ← 0, V ← 0 AH ≠ ALのサイン拡張：CY ← 1, V ← 1	U	×	×	U	U	U
		reg16	1 1 1 1 0 1 1 1	1 1 1 0 1 reg	2	DW, AW ← AW × reg16 DW = AWのサイン拡張：CY ← 0, V ← 0 DW ≠ AWのサイン拡張：CY ← 1, V ← 1	U	×	×	U	U	U
		mem16	1 1 1 1 0 1 1 1	mod 1 0 1 mem	2-4	DW, AW ← AW × (mem16) DW = AWのサイン拡張：CY ← 0, V ← 0 DW ≠ AWのサイン拡張：CY ← 1, V ← 1	U	×	×	U	U	U
		reg16, (reg16,)注 imm8	0 1 1 0 1 0 1 1	1 1 reg reg	3	reg16 ← reg16 × imm8 積 ≦ 16ビット：CY ← 0, V ← 0 積 > 16ビット：CY ← 1, V ← 1	U	×	×	U	U	U
		reg16, mem16, imm8	0 1 1 0 1 0 1 1	mod reg mem	3-5	reg16 ← (mem16) × imm8 積 ≦ 16ビット：CY ← 0, V ← 0 積 > 16ビット：CY ← 1, V ← 1	U	×	×	U	U	U
		reg16, (reg16,)注 imm16	0 1 1 0 1 0 0 1	1 1 reg reg	4	reg16 ← reg16 × imm16 積 ≦ 16ビット：CY ← 0, V ← 0 積 > 16ビット：CY ← 1, V ← 1	U	×	×	U	U	U
		reg16, mem16, imm16	0 1 1 0 1 0 0 1	mod reg mem	4-6	reg16 ← (mem16) × imm16 積 ≦ 16ビット：CY ← 0, V ← 0 積 > 16ビット：CY ← 1, V ← 1	U	×	×	U	U	U

注 第2オペランドは省略可。省略した場合は第1オペランドと同じレジスタを指定したことになります。

保守／廃止

表A-1 ニモニック一覧 (11/33)

命令群	ニモニック	オペランド	オペレーション・コード 7 6 5 4 3 2 1 0	オペレーション・コード 7 6 5 4 3 2 1 0	バイト数	オペレーション	フラグ AC	CY	V	P	S	Z
符号なし除算命令	DIVU	reg8	1 1 1 1 0 1 1 0	1 1 1 1 0 reg	2	temp ← AW temp ÷ reg8 ≦ FFHのとき AH ← temp%reg8, AL ← temp ÷ reg8 temp ÷ reg8 > FFHのとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U
		mem8	1 1 1 1 0 1 1 0	mod 1 1 0 mem	2-4	temp ← AW temp ÷ (mem8) ≦ FFHのとき AH ← temp % (mem8), AL ← temp ÷ (mem8) temp ÷ (mem8) > FFHのとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U
		reg16	1 1 1 1 0 1 1 1	1 1 1 1 0 reg	2	temp ← DW, AW temp ÷ reg16 ≦ FFFFHのとき DW ← temp % reg16, AW ← temp ÷ reg16 temp ÷ reg16 > FFFFHのとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U
		mem16	1 1 1 1 0 1 1 1	mod 1 1 0 mem	2-4	temp ← DW, AW temp ÷ (mem16) ≦ FFFFHのとき DW ← temp % (mem16), AW ← temp ÷ (mem16) temp ÷ (mem16) > FFFFHのとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U

保守／廃止

表A-1 ニモニック一覧 (12/33)

命令群	ニモニック	オペランド	オペレーション・コード 7 6 5 4 3 2 1 0	オペレーション・コード 7 6 5 4 3 2 1 0	バイト数	オペレーション	フラグ AC	CY	V	P	S	Z
符号付き除算命令	DIV	reg8	1 1 1 1 0 1 1 0	1 1 1 1 1 reg	2	temp ← AW temp ÷ reg8 > 0でtemp ÷ reg8 ≦ 7FHまたは temp ÷ reg8 < 0でtemp ÷ reg8 > 0 − 7FH − 1のとき AH ← temp % reg8, AL ← temp ÷ reg8 temp ÷ reg8 > 0でtemp ÷ reg8 > 7FHまたは temp ÷ reg8 > 0でtemp ÷ reg8 < 0 − 7FH − 1のとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U
		mem8	1 1 1 1 0 1 1 0	mod 1 1 1 mem	2-4	temp ← AW temp ÷ (mem8) > 0でtemp ÷ (mem8) ≦ 7FHまたは temp ÷ (mem8) < 0 で temp ÷ (mem8) > 0 − 7FH − 1のとき AH ← temp % (mem8), AL ← temp ÷ (mem8) temp ÷ (mem8) > 0でtemp ÷ (mem8) > 7FHまたは temp ÷ (mem8) > 0 で temp ÷ (mem8) < 0 − 7FH − 1のとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U
		reg16	1 1 1 1 0 1 1 1	1 1 1 1 1 reg	2	temp ← DW, AW temp ÷ reg16 > 0でtemp ÷ reg16 ≦ 7FFFHまたは temp ÷ reg16 < 0 で temp ÷ reg16 > 0 − 7FFFH − 1のとき DW ← temp % reg16, AW ← temp ÷ reg16 temp ÷ reg16 > 0でtemp ÷ reg16 > 7FFFHまたは temp ÷ reg16 > 0 で temp ÷ reg16 < 0 − 7FFFH − 1のとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U

保守／廃止

表A-1 ニモニック一覧（13/33）

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
符号付き除算命令	DIV	mem16	1 1 1 1 0 1 1 1	mod 1 1 1 mem	2-4	temp ← DW, AW temp ÷ (mem16) > 0 でtemp ÷ (mem16) ≦ 7FFFHまたは temp ÷ (mem16) < 0 でtemp ÷ (mem16) > 0 − 7FFFH − 1のとき DW ← temp % (mem16), AW ← temp ÷ (mem16) temp ÷ (mem16) > 0 でtemp ÷ (mem16) > 7FFFHまたは temp ÷ (mem16) > 0 でtemp ÷ (mem16) < 0 − 7FFFH − 1のとき (SP − 1, SP − 2) ← PSW, (SP − 3, SP − 4) ← PS (SP − 5, SP − 6) ← PC, SP ← SP − 6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U
BCD補正命令	ADJBA		0 0 1 1 0 1 1 1		1	AL ∧ 0FH > 9 or AC = 1のときAL ← AL + 6, AH ← AH + 1, AC ← 1, CY ← AC, AL ← AL ∧ 0FH	×	×	U	U	U	U
	ADJ4A		0 0 1 0 0 1 1 1		1	AL ∧ 0FH > 9 or AC = 1のとき AL ← AL + 6, CY ← CY ∨ AC, AC ← 1 AL > 9FH or CY = 1のとき AL ← AL + 60H, CY ← 1	×	×	U	×	×	×
	ADJBS		0 0 1 1 1 1 1 1		1	AL ∧ 0FH > 9 or AC = 1のときAL ← AL − 6, AH ← AH − 1, AC ← 1, CY ← AC, AL ← AL ∧ 0FH	×	×	U	U	U	U
	ADJ4S		0 0 1 0 1 1 1 1		1	AL ∧ 0FH > 9 or AC = 1のときAL ← AL − 6, CY ← CY ∨ AC, AC ← 1 AL > 9FH or CY = 1のとき AL ← AL − 60H, CY ← 1	×	×	U	×	×	×
データ変換命令	CVTBD		1 1 0 1 0 1 0 0	0 0 0 0 1 0 1 0	2	AH ← AL ÷ 0AH, AL ← AL % 0AH	U	U	U	×	×	×
	CVTDB		1 1 0 1 0 1 0 1	0 0 0 0 1 0 1 0	2	AH ← 0, AL ← AH × 0AH + AL	U	U	U	×	×	×
	CVTBW		1 0 0 1 1 0 0 0		1	AL < 80HのときAH ← 0, それ以外のときAH ← FFH						
	CVTWL		1 0 0 1 1 0 0 1		1	AW < 8000HのときDW ← 0, それ以外のときDW ← FFFFH						

保守／廃止

表A－1 ニモニック一覧（14/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
比較命令	CMP	reg, reg′	0	0	1	1	1	0	1	W	1	1	reg				reg′		2	reg − reg′	×	×	×	×	×	×
		mem, reg	0	0	1	1	1	0	0	W	mod		reg			mem			2-4	(mem) − reg	×	×	×	×	×	×
		reg, mem	0	0	1	1	1	0	1	W	mod		reg			mem			2-4	reg − (mem)	×	×	×	×	×	×
		reg, imm	1	0	0	0	0	0	S	W	1	1	1	1	1	reg			3-4	reg − imm	×	×	×	×	×	×
		mem, imm	1	0	0	0	0	0	S	W	mod		1	1	1	mem			3-6	(mem) − imm	×	×	×	×	×	×
		acc, imm	0	0	1	1	1	1	0	W									2-3	W = 0のとき AL − imm W = 1のとき AW − imm	×	×	×	×	×	×
補数演算命令	NOT	reg	1	1	1	1	0	1	1	W	1	1	0	1	0	reg			2	reg ← $\overline{\text{reg}}$						
		mem	1	1	1	1	0	1	1	W	mod		0	1	0	mem			2-4	(mem) ← $\overline{(\text{mem})}$						
	NEG	reg	1	1	1	1	0	1	1	W	1	1	0	1	1	reg			2	reg ← $\overline{\text{reg}}$ + 1	×	×	×	×	×	×
		mem	1	1	1	1	0	1	1	W	mod		0	1	1	mem			2-4	(mem) ← $\overline{(\text{mem})}$ + 1	×	×	×	×	×	×
論理演算命令	TEST	reg, reg′	1	0	0	0	0	1	0	W	1	1	reg				reg′		2	reg ∧ reg′	U	0	0	×	×	×
		mem, reg reg, mem	1	0	0	0	0	1	0	W	mod		reg			mem			2-4	(mem) ∧ reg	U	0	0	×	×	×
		reg, imm	1	1	1	1	0	1	1	W	1	1	0	0	0	reg			3-4	reg ∧ imm	U	0	0	×	×	×
		mem, imm	1	1	1	1	0	1	1	W	mod		0	0	0	mem			3-6	(mem) ∧ imm	U	0	0	×	×	×
		acc, imm	1	0	1	0	1	0	0	W									2-3	W = 0のとき AL ∧ imm8 W = 1のとき AW ∧ imm16	U	0	0	×	×	×

保守／廃止

表A－1　ニモニック一覧（15/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
論理演算命令	AND	reg, reg'	0	0	1	0	0	0	1	W	1	1	reg				reg'		2	reg ← reg ∧ reg'	U	0	0	×	×	×
		mem, reg	0	0	1	0	0	0	0	W	mod		reg			mem			2-4	(mem) ← (mem) ∧ reg	U	0	0	×	×	×
		reg, mem	0	0	1	0	0	0	1	W	mod		reg			mem			2-4	reg ← reg ∧ (mem)	U	0	0	×	×	×
		reg, imm	1	0	0	0	0	0	0	W	1	1	1	0	0		reg		3-4	reg ← reg ∧ imm	U	0	0	×	×	×
		mem, imm	1	0	0	0	0	0	0	W	mod	1	0	0		mem			3-6	(mem) ← (mem) ∧ imm	U	0	0	×	×	×
		acc, imm	0	0	1	0	0	1	0	W									2-3	W = 0のとき　AL ← AL ∧ imm8 W = 1のとき　AW ← AW ∧ imm16	U	0	0	×	×	×
	OR	reg, reg'	0	0	0	0	1	0	1	W	1	1	reg				reg'		2	reg ← reg ∨ reg'	U	0	0	×	×	×
		mem, reg	0	0	0	0	1	0	0	W	mod		reg			mem			2-4	(mem) ← (mem) ∨ reg	U	0	0	×	×	×
		reg, mem	0	0	0	0	1	0	1	W	mod		reg			mem			2-4	reg ← reg ∨ (mem)	U	0	0	×	×	×
		reg, imm	1	0	0	0	0	0	0	W	1	1	0	0	1		reg		3-4	reg ← reg ∨ imm	U	0	0	×	×	×
		mem, imm	1	0	0	0	0	0	0	W	mod	0	0	1		mem			3-6	(mem) ← (mem) ∨ imm	U	0	0	×	×	×
		acc, imm	0	0	0	0	1	1	0	W									2-3	W = 0のとき　AL ← AL ∨ imm8 W = 1のとき　AW ∨ imm16	U	0	0	×	×	×
	XOR	reg, reg'	0	0	1	1	0	0	1	W	1	1	reg				reg'		2	reg ← reg ⊻ reg'	U	0	0	×	×	×
		mem, reg	0	0	1	1	0	0	0	W	mod		reg			mem			2-4	(mem) ← (mem) ⊻ reg	U	0	0	×	×	×
		reg, mem	0	0	1	1	0	0	1	W	mod		reg			mem			2-4	reg ← reg ⊻ (mem)	U	0	0	×	×	×
		reg, imm	1	0	0	0	0	0	0	W	1	1	1	1	0		reg		3-4	reg ← reg ⊻ imm	U	0	0	×	×	×
		mem, imm	1	0	0	0	0	0	0	W	mod	1	1	0		mem			3-6	(mem) ← (mem) ⊻ imm	U	0	0	×	×	×
		acc, imm	0	0	1	1	0	1	0	W									2-3	W = 0のとき　AL ← AL ⊻ imm8 W = 1のとき　AW ← AW ⊻ imm16	U	0	0	×	×	×

保守／廃止

表A－1　ニモニック一覧（16/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
ビット操作命令	TEST1	reg8, CL	0	0	0	1	0	0	0	0	1	1	0	0	0	reg			3	reg8のビットNO.CL＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
		mem8, CL					0	0	0	0	mod	0	0	0	mem				3-5	(mem8)のビットNO.CL＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
		reg16, CL					0	0	0	1	1	1	0	0	0	reg			3	reg16のビットNO.CL＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
		mem16, CL					0	0	0	1	mod	0	0	0	mem				3-5	(mem16)のビットNO.CL＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
		reg8, imm3					1	0	0	0	1	1	0	0	0	reg			4	reg8のビットNO.imm3＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
		mem8, imm3					1	0	0	0	mod	0	0	0	mem				4-6	(mem8)のビットNO.imm3＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
		reg16, imm4					1	0	0	1	1	1	0	0	0	reg			4	reg16のビットNO.imm4＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
		mem16, imm4					1	0	0	1	mod	0	0	0	mem				4-6	(mem16)のビットNO.imm4＝0：Z←1 ＝1：Z←0	U	0	0	U	U	×
	NOT1	reg8, CL					0	1	1	0	1	1	0	0	0	reg			3	reg8のビットNO.CL←$\overline{\text{reg8のビットNO.CL}}$						
		mem8, CL					0	1	1	0	mod	0	0	0	mem				3-5	(mem8)のビットNO.CL←$\overline{\text{(mem8)ビットNO.CL}}$						
		reg16, CL					0	1	1	1	1	1	0	0	0	reg			3	reg16のビットNO.CL←$\overline{\text{reg16のビットNO.CL}}$						
		mem16, CL					0	1	1	1	mod	0	0	0	mem				3-5	(mem16)のビットNO.CL←$\overline{\text{(mem16)のビットNO.CL}}$						
		reg8, imm3					1	1	1	0	1	1	0	0	0	reg			4	reg8のビットNO.imm3←$\overline{\text{reg8のビットNO.imm3}}$						
		mem8, imm3					1	1	1	0	mod	0	0	0	mem				4-6	(mem8)のビットNO.imm3 ←$\overline{\text{(mem8) のビットNO.imm3}}$						
		reg16, imm4					1	1	1	1	1	1	0	0	0	reg			4	reg16のビットNO.imm4←$\overline{\text{reg16のビットNO.imm4}}$						
		mem16, imm4					1	1	1	1	mod	0	0	0	mem				4-6	(mem16)のビットNO.imm4 ←$\overline{\text{(mem16)のビットNO.imm4}}$						
			2バイト目[注]								3バイト目[注]									注　1バイト目＝0FH						
	NOT1	CY	1	1	1	1	0	1	0	1									1	CY←$\overline{\text{CY}}$		×				

保守／廃止

表A-1 ニモニック一覧 (17/33)

命令群	ニモニック	オペランド	オペレーション・コード																バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
ビット操作命令	CLR1	reg8, CL	0	0	0	1	0	0	1	0	1	1	0	0	0	reg			3	reg8のビットNO. CL ← 0						
		mem8, CL					0	0	1	0	mod		0	0	0	mem			3-5	(mem8)のビットNO. CL ← 0						
		reg16, CL					0	0	1	1	1	1	0	0	0	reg			3	reg16のビットNO. CL ← 0						
		mem16, CL					0	0	1	1	mod		0	0	0	mem			3-5	(mem16)のビットNO. CL ← 0						
		reg8, imm3					1	0	1	0	1	1	0	0	0	reg			4	reg8のビットNO. imm3 ← 0						
		mem8, imm3					1	0	1	0	mod		0	0	0	mem			4-6	(mem8)のビットNO. imm3 ← 0						
		reg16, imm4					1	0	1	1	1	1	0	0	0	reg			4	reg16のビットNO. imm4 ← 0						
		mem16, imm4					1	0	1	1	mod		0	0	0	mem			4-6	(mem16)のビットNO. imm4 ← 0						
			2バイト目[注]								3バイト目[注]									注 1バイト目 = 0FH						
		CY	1	1	1	1	1	0	0	0									1	CY ← 0		0				
		DIR	1	1	1	1	1	1	0	0									1	DIR ← 0						
	SET1	reg8, CL	0	0	0	1	0	1	0	0	1	1	0	0	0	reg			3	reg8のビットNO. CL ← 1						
		mem8, CL					0	1	0	0	mod		0	0	0	mem			3-5	(mem8)のビットNO. CL ← 1						
		reg16, CL					0	1	0	1	1	1	0	0	0	reg			3	reg16のビットNO. CL ← 1						
		mem16, CL					0	1	0	1	mod		0	0	0	mem			3-5	(mem16)のビットNO. CL ← 1						
		reg8, imm3					1	1	0	0	1	1	0	0	0	reg			4	reg8のビットNO. imm3 ← 1						
		mem8, imm3					1	1	0	0	mod		0	0	0	mem			4-6	(mem8)のビットNO. imm3 ← 1						
		reg16, imm4					1	1	0	1	1	1	0	0	0	reg			4	reg16のビットNO. imm4 ← 1						
		mem16, imm4					1	1	0	1	mod		0	0	0	mem			4-6	(mem16)のビットNO. imm4 ← 1						
			2バイト目[注]								3バイト目[注]									注 1バイト目 = 0FH						
		CY	1	1	1	1	1	0	0	1									1	CY ← 1		1				
		DIR	1	1	1	1	1	1	0	1									1	DIR ← 1						

保守／廃止

表A－1 ニモニック一覧（18/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
ビット操作命令	BSCH	reg8	0	0	0	0	1	1	1	1	0	0	1	1	1	1	0	0	3	①CL←0 ②［reg8のビットNo.CL＝0のとき］CL＜7のときCL←CL＋1，②から再実行 CL＝7のときZ←1 ［reg8のビットNo.CL＝1のとき］Z←0	U	U	U	U	U	×
			1	1	0	0	0	reg																		
		mem8	0	0	0	0	1	1	1	1	0	0	1	1	1	1	0	0	3－5	①CL←0 ②［mem8のビットNo.CL＝0のとき］CL＜7のときCL←CL＋1，②から再実行 CL＝7のときZ←1 ［mem8のビットNo.CL＝1のとき］Z←0	U	U	U	U	U	×
			mod		0	0	0	mem																		
		reg16	0	0	0	0	1	1	1	1	0	0	1	1	1	1	0	1	3	①CL←0 ②［reg16のビットNo.CL＝0のとき］CL＜15のときCL←CL＋1，②から再実行 CL＝15のときZ←1 ［reg16のビットNo.CL＝1のとき］Z←0	U	U	U	U	U	×
			1	1	0	0	0	reg																		
		mem16	0	0	0	0	1	1	1	1	0	0	1	1	1	1	0	1	3－5	①CL←0 ②［mem16のビットNo.CL＝0のとき］CL＜15のときCL←CL＋1，②から再実行 CL＝15のときZ←1 ［mem16のビットNo.CL＝1のとき］Z←0	U	U	U	U	U	×
			mod		0	0	0	mem																		

保守／廃止

表A-1 ニモニック一覧（19/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
シフト命令	SHL	reg, 1	1	1	0	1	0	0	0	W	1	1	1	0	0	reg			2	CY ← regのMSB, reg ← reg × 2 regのMSB ≠ CYのときV ← 1 regのMSB = CYのときV ← 0	U	×	×	×	×	×
		mem, 1	1	1	0	1	0	0	0	W	mod	1	0	0	mem				2-4	CY ← (mem)のMSB, (mem) ← (mem) × 2 (mem)のMSB ≠ CYのときV ← 1 (mem)のMSB = CYのときV ← 0	U	×	×	×	×	×
		reg, CL	1	1	0	1	0	0	1	W	1	1	1	0	0	reg			2	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← regのMSB, reg ← reg × 2 temp ← temp − 1	U	×	U	×	×	×
		mem, CL	1	1	0	1	0	0	1	W	mod	1	0	0	mem				2-4	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のMSB, (mem) ← (mem) × 2 temp ← temp − 1	U	×	U	×	×	×
		reg, imm8	1	1	0	0	0	0	0	W	1	1	1	0	0	reg			3	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← regのMSB, reg ← reg × 2 temp ← temp − 1	U	×	U	×	×	×
		mem, imm8	1	1	0	0	0	0	0	W	mod	1	0	0	mem				3-5	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のMSB, (mem) ← (mem) × 2 temp ← temp − 1	U	×	U	×	×	×

保守／廃止

表A－1 ニモニック一覧（20/33）

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
シフト命令	SHR	reg, 1	1 1 0 1 0 0 0 W	1 1 1 0 1 reg	2	CY ← regのLSB, reg ← reg ÷ 2 regのMSB ≠ regのMSBの次のビット：V ← 1 regのMSB = regのMSBの次のビット：V ← 0	U	×	×	×	×	×
		mem, 1	1 1 0 1 0 0 0 W	mod 1 0 1 mem	2-4	CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 (mem)のMSB ≠ (mem)のMSBの次のビット：V ← 1 (mem)のMSB = (mem)のMSBの次のビット：V ← 0	U	×	×	×	×	×
		reg, CL	1 1 0 1 0 0 1 W	1 1 1 0 1 reg	2	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← regのLSB, reg ← reg ÷ 2 temp ← temp − 1	U	×	U	×	×	×
		mem, CL	1 1 0 1 0 0 1 W	mod 1 0 1 mem	2-4	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 temp ← temp − 1	U	×	U	×	×	×
		reg, imm8	1 1 0 0 0 0 0 W	1 1 1 0 1 reg	3	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← regのLSB, reg ← reg ÷ 2 temp ← temp − 1	U	×	U	×	×	×
		mem, imm8	1 1 0 0 0 0 0 W	mod 1 0 1 mem	3-5	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 temp ← temp − 1	U	×	U	×	×	×

保守／廃止

表A-1 ニモニック一覧 (21/33)

命令群	ニモニック	オペランド	オペレーション・コード															バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
シフト命令	SHRA	reg, 1	1	1	0	1	0	0	0	W	1	1	1	1	1		reg		2	CY ← regのLSB, reg ← reg ÷ 2, V ← 0 オペランドのMSBは変化しません。	U	×	0	×	×	×
		mem, 1	1	1	0	1	0	0	0	W	mod		1	1	1		reg		2-4	CY ← (mem)のLSB, (mem) ← (mem) ÷ 2, V ← 0 オペランドのMSBは変化しません。	U	×	0	×	×	×
		reg, CL	1	1	0	1	0	0	1	W	1	1	1	1	1		reg		2	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← regのLSB, reg ← reg ÷ 2 temp ← temp − 1 オペランドのMSBは変化しません。	U	×	U	×	×	×
		mem, CL	1	1	0	1	0	0	1	W	mod		1	1	1		mem		2-4	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 temp ← temp − 1 オペランドのMSBは変化しません。	U	×	U	×	×	×
		reg, imm8	1	1	0	0	0	0	0	W	1	1	1	1	1		reg		3	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← regのLSB, reg ← reg ÷ 2 temp ← temp − 1 オペランドのMSBは変化しません。	U	×	U	×	×	×
		mem, imm8	1	1	0	0	0	0	0	W	mod		1	1	1		mem		3-5	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 temp ← temp − 1 オペランドのMSBは変化しません。	U	×	U	×	×	×

保守／廃止

表A－1 ニモニック一覧（22/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
ローテート命令	ROL	reg, 1	1	1	0	1	0	0	0	W	1	1	0	0	0	reg			2	CY ← regのMSB, reg ← reg × 2 + CY regのMSB ≠ CY : V ← 1 regのMSB = CY : V ← 0			×	×			
		mem, 1	1	1	0	1	0	0	0	W	mod	0	0	0	mem				2－4	CY ← (mem)のMSB, (mem) ← (mem) × 2 + CY (mem)のMSB ≠ CY : V ← 1 (mem)のMSB = CY : V ← 0			×	×			
		reg, CL	1	1	0	1	0	0	1	W	1	1	0	0	0	reg			2	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← regのMSB, reg ← reg × 2 + CY temp ← temp － 1			×	U			
		mem, CL	1	1	0	1	0	0	1	W	mod	0	0	0	mem				2－4	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のMSB, (mem) ← (mem) × 2 + CY temp ← temp － 1			×	U			
		reg, imm8	1	1	0	0	0	0	0	W	1	1	0	0	0	reg			3	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← regのMSB, reg ← reg × 2 + CY temp ← temp － 1			×	U			
		mem, imm8	1	1	0	0	0	0	0	W	mod	0	0	0	mem				3－5	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のMSB, (mem) ← (mem) × 2 + CY temp ← temp － 1			×	U			

保守／廃止

表A-1 ニモニック一覧 (23/33)

命令群	ニモニック	オペランド	オペレーション・コード																バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
ローテート命令	ROR	reg, 1	1	1	0	1	0	0	0	W	1	1	0	0	1	reg			2	CY ← regのLSB, reg ← reg ÷ 2 regのMSB ← CY regのMSB ≠ regのMSBの次のビット：V ← 1 regのMSB = regのMSBの次のビット：V ← 0		×	×			
		mem, 1	1	1	0	1	0	0	0	W	mod	0	0	1	mem				2-4	CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 (mem)のMSB ← CY (mem)のMSB ≠ (mem)のMSBの次のビット：V ← 1 (mem)のMSB = (mem)のMSBの次のビット：V ← 0		×	×			
		reg, CL	1	1	0	1	0	0	1	W	1	1	0	0	1	reg			2	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← regのLSB, reg ← reg ÷ 2 regのMSB ← CY temp ← temp − 1		×	U			
		mem, CL	1	1	0	1	0	0	1	W	mod	0	0	1	mem				2-4	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 (mem)のMSB ← CY temp ← temp − 1		×	U			
		reg, imm8	1	1	0	0	0	0	0	W	1	1	0	0	1	reg			3	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← regのLSB, reg ← reg ÷ 2 regのMSB ← CY temp ← temp − 1		×	U			
		mem, imm8	1	1	0	0	0	0	0	W	mod	0	0	1	mem				3-5	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す CY ← (mem)のLSB, (mem) ← (mem) ÷ 2 (mem)のMSB ← CY temp ← temp − 1		×	U			

保守／廃止

表A-1 ニモニック一覧 (24/33)

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
ローテート命令	ROLC	reg, 1	1 1 0 1 0 0 0 W	1 1 0 1 0 reg	2	tmpcy ← CY, CY ← regのMSB reg ← reg × 2 + tmpcy regのMSB ≠ CY : V ← 1 regのMSB = CY : V ← 0		×	×			
		mem, 1	1 1 0 1 0 0 0 W	mod 0 1 0 mem	2-4	tmpcy ← CY, CY ← (mem)のMSB (mem) ← (mem) × 2 + tmpcy (mem)のMSB ≠ CY : V ← 1 (mem)のMSB = CY : V ← 0		×	×			
		reg, CL	1 1 0 1 0 0 1 W	1 1 0 1 0 reg	2	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← regのMSB reg ← reg × 2 + tmpcy temp ← temp − 1		×	U			
		mem, CL	1 1 0 1 0 0 1 W	mod 0 1 0 mem	2-4	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← (mem)のMSB (mem) ← (mem) × 2 + tmpcy temp ← temp − 1		×	U			
		reg, imm8	1 1 0 0 0 0 0 W	1 1 0 1 0 reg	3	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← regのMSB reg ← reg × 2 + tmpcy temp ← temp − 1		×	U			
		mem, imm8	1 1 0 0 0 0 0 W	mod 0 1 0 mem	3-5	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← (mem)のMSB (mem) ← (mem) × 2 + tmpcy temp ← temp − 1		×	U			

保守／廃止

表A－1　ニモニック一覧（25/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
ロテート命令	RORC	reg, 1	1	1	0	1	0	0	0	W	1	1	0	1	1	reg			2	tmpcy ← CY, CY ‹ regのLSB reg ← reg ÷ 2 regのMSB ← tmpcy regのMSB ≠ regのMSBの次のビット：V ‹ 1 regのMSB = regのMSBの次のビット：V ‹ 0			×	×			
		mem, 1	1	1	0	1	0	0	0	W	mod	0	1	1	mem				2-4	tmpcy ← CY, CY ← (mem)のLSB (mem) ← (mem) ÷ 2 (mem)のMSB ← tmpcy (mem)のMSB ≠ (mem)のMSBの次のビット：V ← 1 (mem)のMSB = (mem)のMSBの次のビット：V ← 0			×	×			
		reg, CL	1	1	0	1	0	0	1	W	1	1	0	1	1	reg			2	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← regのLSB reg ← reg ÷ 2 regのMSB ← tmpcy temp ← temp − 1			×	U			
		mem, CL	1	1	0	1	0	0	1	W	mod	0	1	1	mem				2-4	temp ← CL, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← (mem)のLSB (mem) ← (mem) ÷ 2 (mem)のMSB ← tmpcy temp ← temp − 1			×	U			
		reg, imm8	1	1	0	0	0	0	0	W	1	1	0	1	1	reg			3	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← regのLSB reg ← reg ÷ 2 regのMSB ← tmpcy temp ← temp − 1			×	U			
		mem, imm8	1	1	0	0	0	0	0	W	mod	0	1	1	mem				3-5	temp ← imm8, temp ≠ 0の間, 次の動作を繰り返す tmpcy ← CY, CY ← (mem)のLSB (mem) ← (mem) ÷ 2 (mem)のMSB ← tmpcy temp ← temp − 1			×	U			

保守／廃止

表A－1　ニモニック一覧（26/33）

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
サブルーチン制御命令	CALL	near－proc	1 1 1 0 1 0 0 0		3	(SP － 1, SP － 2) ← PC, SP ← SP － 2 PC ← PC + disp						
		regptr16	1 1 1 1 1 1 1 1	1 1 0 1 0 reg	2	(SP － 1, SP － 2) ← PC, PC ← regptr16 SP ← SP － 2						
		memptr16	1 1 1 1 1 1 1 1	mod 0 1 0 mem	2－4	(SP － 1, SP － 2) ← PC, SP ← SP － 2 PC ← (memptr16)						
		far－proc	1 0 0 1 1 0 1 0		5	(SP － 1, SP － 2) ← PS, (SP － 3, SP － 4) ← PC SP ← SP － 4 PS ← seg, PC ← offset						
		memptr32	1 1 1 1 1 1 1 1	mod 0 1 1 mem	2－4	(SP － 1, SP － 2) ← PS, (SP － 3, SP － 4) ← PC SP ← SP － 4 PS ← (memptr32 + 2), PC ← (memptr32)						
	RET		1 1 0 0 0 0 1 1		1	PC ← (SP + 1, SP) SP ← SP + 2						
		pop－value	1 1 0 0 0 0 1 0		3	PC ← (SP + 1, SP) SP ← SP + 2, SP ← SP + pop－value						
			1 1 0 0 1 0 1 1		1	PC ← (SP + 1, SP) PS ← (SP + 3, SP + 2) SP ← SP + 4						
		pop－value	1 1 0 0 1 0 1 0		3	PC ← (SP + 1, SP) PS ← (SP + 3, SP + 2) SP ← SP + 4, SP ← SP + pop－value						

保守／廃止

表A-1 ニモニック一覧 (27/33)

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
スタック操作命令	PUSH	mem16	1	1	1	1	1	1	1	1	mod	1	1	0	mem				2-4	(SP − 1, SP − 2) ← (mem16) SP ← SP − 2						
		reg16	0	1	0	1	0	reg											1	(SP − 1, SP − 2) ← reg16 SP ← SP − 2						
		sreg	0	0	0	sreg		1	1	0									1	(SP − 1, SP − 2) ← sreg SP ← SP − 2						
		PSW	1	0	0	1	1	1	0	0									1	(SP − 1, SP − 2) ← PSW SP ← SP − 2						
		R	0	1	1	0	0	0	0	0									1	Push registers on the stack						
		imm	0	1	1	0	1	0	S	0									2-3	(SP − 1, SP − 2) ← imm SP ← SP − 2, S = 1のときサイン拡張						
	POP	mem16	1	0	0	0	1	1	1	1	mod	0	0	0	mem				2-4	SP ← SP + 2 (mem16) ← (SP − 1, SP − 2)						
		reg16	0	1	0	1	1	reg											1	SP ← SP + 2 reg16 ← (SP − 1, SP − 2)						
		sreg	0	0	0	sreg		1	1	1									1	SP ← SP + 2 sreg ← (SP − 1, SP − 2) sreg : SS, DS0, DS1						
		PSW	1	0	0	1	1	1	0	1									1	SP ← SP + 2 PSW ← (SP − 1, SP − 2)	R	R	R	R	R	R
		R	0	1	1	0	0	0	0	1									1	Pop registers from the stack						
	PREPARE	imm16, imm8	1	1	0	0	1	0	0	0									4	Prepare New Stack Frame						
	DISPOSE		1	1	0	0	1	0	0	1									1	Dispose of Stack Frame						

保守／廃止

表A－1 ニモニック一覧（28/33）

命令群	ニモニック	オペランド	オペレーション・コード 7 6 5 4 3 2 1 0	オペレーション・コード 7 6 5 4 3 2 1 0	バイト数	オペレーション	フラグ AC	CY	V	P	S	Z
ブランチ命令	BR	near－label	1 1 1 0 1 0 0 1		3	PC ← PC + disp						
		short－label	1 1 1 0 1 0 1 1		2	PC ← PC + ext－disp8						
		regptr16	1 1 1 1 1 1 1 1	1 1 1 0 0 reg	2	PC ← regptr16						
		memptr16	1 1 1 1 1 1 1 1	mod 1 0 0 mem	2－4	PC ← (memptr16)						
		far－label	1 1 1 0 1 0 1 0		5	PS ← Seg PC ← offset						
		memptr32	1 1 1 1 1 1 1 1	mod 1 0 1 mem	2－4	PS ← (memptr32 + 2) PC ← (memptr32)						
条件付きブランチ命令	BV	short－label	0 1 1 1 0 0 0 0		2	if V = 1 PC ← PC + ext − disp8						
	BNV	short－label	0 0 0 1		2	if V = 0 PC ← PC + ext − disp8						
	BC BL	short－label	0 0 1 0		2	if CY = 1 PC ← PC + ext − disp8						
	BNC BNL	short－label	0 0 1 1		2	if CY = 0 PC ← PC + ext − disp8						
	BE BZ	short－label	0 1 0 0		2	if Z = 1 PC ← PC + ext − disp8						
	BNE BNZ	short－label	0 1 0 1		2	if Z = 0 PC ← PC + ext − disp8						
	BNH	short－label	0 1 1 0		2	if CY ∧ Z = 1 PC ← PC + ext − disp8						
	BH	short－label	0 1 1 1		2	if CY ∧ Z = 0 PC ← PC + ext − disp8						
	BN	short－label	1 0 0 0		2	if S = 1 PC ← PC + ext − disp8						
	BP	short－label	1 0 0 1		2	if S = 0 PC ← PC + ext − disp8						
	BPE	short－label	1 0 1 0		2	if P = 1 PC ← PC + ext − disp8						
	BPO	short－label	1 0 1 1		2	if P = 0 PC ← PC + ext − disp8						
	BLT	short－label	1 1 0 0		2	if S ∀ V = 1 PC ← PC + ext − disp8						

保守／廃止

表A－1 ニモニック一覧（29/33）

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
条件付きブランチ命令	BGE	short－label	0	1	1	1	1	1	0	1									2	if S ∀ V = 0 PC ← PC + ext − disp8						
	BLE	short－label					1	1	1	0									2	if (S ∀ V) ∨ Z = 1 PC ← PC + ext − disp8						
	BGT	short－label					1	1	1	1									2	if (S ∀ V) ∨ Z = 0 PC ← PC + ext − disp8						
	DBNZNE	short－label	1	1	1	0	0	0	0	0									2	CW = CW − 1 if Z = 0 and CW ≠ 0 PC ← PC + ext − disp8						
	DBNZE	short－label					0	0	0	1									2	CW = CW − 1 if Z = 1 and CW ≠ 0 PC ← PC + ext − disp8						
	DBNZ	short－label					0	0	1	0									2	CW = CW − 1 if CW ≠ 0 PC ← PC + ext − disp8						
	BCWZ	short－label					0	0	1	1									2	if CW = 0 PC ← PC + ext − disp8						
	BTCLR[注]	sfr, imm3, short－label	0	0	0	0	1	1	1	1	1	0	0	1	1	1	0	0	5	(sfr)のビットNo. imm3 = 1のとき PC ← PC + ext − disp8, (sfr)のビットNo. imm3 ← 0						
			sfr								imm3															
			disp8																							

注 V20，V30に対して，追加した命令です。

保守／廃止

表A－1 ニモニック一覧（30/33）

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
割り込み命令	BRK	3	1 1 0 0 1 1 0 0		1	(SP－1, SP－2)←PSW, (SP－3, SP－4)←PS, (SP－5, SP－6)←PC, SP←SP－6 IE←0, BRK←0 PS←(15, 14), PC←(13, 12)						
		imm8 (≠3)	1 1 0 0 1 1 0 1		2	(SP－1, SP－2)←PSW, (SP－3, SP－4)←PS, (SP－5, SP－6)←PC, SP←SP－6 IE←0, BRK←0 PS←(n×4＋3, n×4＋2), PC←(n×4＋1, n×4) n＝imm8						
	BRKV		1 1 0 0 1 1 1 0		1	V＝1のとき (SP－1, SP－2)←PSW, (SP－3, SP－4)←PS, (SP－5, SP－6)←PC, SP←SP－6 IE←0, BRK←0 PS←(19, 18), PC←(17, 16)						
	RETI		1 1 0 0 1 1 1 1		1	PC←(SP＋1, SP), PS←(SP＋3, SP＋2), PSW←(SP＋5, SP＋4), SP←SP＋6	R	R	R	R	R	R
	RETRBI注		0 0 0 0 1 1 1 1	1 0 0 1 0 0 0 1	2	PC←現在選択されているレジスタ・バンク上のPC退避領域の値 PSW←現在選択されているレジスタ・バンク上のPSW退避領域の値	R	R	R	R	R	R
	FINT注		0 0 0 0 1 1 1 1	1 0 0 1 0 0 1 0	2	CPUに内蔵される割り込みコントローラへ割り込み処理ルーチンが終了したことを示す						
	CHKIND	reg16, mem32	0 1 1 0 0 0 1 0	mod reg mem	2－4	(mem32)＞reg16または (mem32＋2)＜reg16のとき (SP－1, SP－2)←PSW, (SP－3, SP－4)←PS, (SP－5, SP－6)←PC, SP←SP－6 IE←0, BRK←0 PS←(23, 22), PC←(21, 20)						

注 V20, V30に対して，追加した命令です。

保守／廃止

表A－1 ニモニック一覧（31/33）

命令群	ニモニック	オペランド	オペレーション・コード		バイト数	オペレーション	フラグ					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z
バンク切り替え	BRKCS[注1]	reg16	0 0 0 0 1 1 1 1	0 0 1 0 1 1 0 1	3	temp ← PSW RB3－0 ← reg16の下位４ビット，IE ← 0，BRK ← 0， 新たに選択されたレジスタ・バンク上のPSW退避領域 ← temp 新たに選択されたレジスタ・バンク上のPC退避領域 ← PC PC ← 新たに選択されたレジスタ・バンク上のベクタPC						
			1 1 0 0 0 reg									
	TSKSW[注1]	reg16	0 0 0 0 1 1 1 1	1 0 0 1 0 1 0 0	3	現在選択されているレジスタ・バンク上のPSW退避領域 ← PSW 現在選択されているレジスタ・バンク上のPC退避領域 ← PC RB3－0 ← reg16の下位４ビット， PSW ← 新たに選択されたレジスタ・バンク上のPSW退避領域の値 PC ← 新たに選択されたレジスタ・バンク上のPC退避領域の値	×	×	×	×	×	×
			1 1 1 1 1 reg									
CPU制御命令	HALT		1 1 1 1 0 1 0 0		1	CPU Halt						
	STOP[注1]		0 0 0 0 1 1 1 1	1 0 0 1 1 1 1 0	2	CPU Stop						
	POLL		1 0 0 1 1 0 1 1		1	Poll and wait						
	DI		1 1 1 1 1 0 1 0		1	IE ← 0						
	EI		1 1 1 1 1 0 1 1		1	IE ← 1						
	BUSLOCK		1 1 1 1 0 0 0 0		1	Bus Lock Prefix						
	FPO1	fp－op	1 1 0 1 1 X X X	1 1 Y Y Y Z Z Z	2	(SP － 1，SP － 2) ← PSW， (SP － 3，SP － 4) ← PS， (SP － 5，SP － 6) ← PC － x[注3]，SP ← SP － 6 IE ← 0，BRK ← 0 PC ← (01DH，01CH)，PS ← (01FH，01EH)						
		fp－op，mem	1 1 0 1 1 X X X	mod Y Y Y mem	2－4							
	FPO2	fp－op	0 1 1 0 0 1 1 X	1 1 Y Y Y Z Z Z	2							
		fp－op，mem	0 1 1 0 0 1 1 X	mod Y Y Y mem	2－4							
	NOP		1 0 0 1 0 0 0 0		1	No Operation						
↑	注2		0 0 1 sreg 1 1 0		1	セグメント・オーバライド・プリフィクス						

└セグメント・オーバライド・プリフィクス

注1．V20，V30に対して，追加した命令

2．DS0：，DS1：，PS：，SS：の４種類

3．x：命令のバイト数＋プリフィクスの数

保守／廃止

表A－1 ニモニック一覧（32/33）

V25，V35に対する追加命令

命令群	ニモニック	オペランド	オペレーション・コード														バイト数	オペレーション	フラグ					
			7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z
追加命令	PUSH	DS2	0	0	0	0	1	1	1	1	0	1	1	1	1	1	1	0	2	(SP－1，SP－2)←DS2，SP←SP－2						
		DS3／VPC	0	0	0	0	1	1	1	1	0	1	1	1	0	1	1	0	2	(SP－1，SP－2)←DS3／VPC，SP←SP－2						
	POP	DS2	0	0	0	0	1	1	1	1	0	1	1	1	1	1	1	1	2	SP←SP＋2，DS2←(SP－1，SP－2)						
		DS3／VPC	0	0	0	0	1	1	1	1	0	1	1	1	0	1	1	1	2	SP←SP＋2，DS3/VPC←(SP－1，SP－2)						
	DS2：		0	1	1	0	0	0	1	1									1	拡張セグメント・オーバライド・プリフィクス						
	DS3：		1	1	0	1	0	1	1	0									1	拡張セグメント・オーバライド・プリフィクス						
	MOV	xsreg，reg16	1	0	0	0	1	1	1	0	1	1	1	xsreg			reg		2	xsreg←reg16　xsreg：DS2，DS3／VPC						
		xsreg，mem16	1	0	0	0	1	1	1	0	mod		1	xsreg			mem		2－4	xsreg←(mem16)　xsreg：DS2，DS3／VPC						
		reg16，xsreg	1	0	0	0	1	1	0	0	1	1	1	xsreg			reg		2	reg16←xsreg　xsreg：DS2，DS3／VPC						
		mem16，xsreg	1	0	0	0	1	1	0	0	mod		1	xsreg			mem		2－4	(mem16)←xsreg　xsreg：DS2，DS3／VPC						
		DS2，reg16，mem32	0	0	0	0	1	1	1	1	0	0	1	1	1	1	1	0	3－5	reg16←(mem32) DS2←(mem32＋2)						
			mod			reg			mem																		
		DS3，reg16，mem32	0	0	0	0	1	1	1	1	0	0	1	1	0	1	1	0	3－5	reg16←(mem32) DS3←(mem32＋2)						
			mod			reg			mem																		
	IRAM：		1	1	1	1	0	0	0	1									1	レジスタ・ファイル用オーバライド・プリフィクス						
	RSTWDT	imm8，$\overline{\text{imm8'}}$	0	0	0	0	1	1	1	1	1	0	0	1	0	1	1	0	4	imm8＝$\overline{\text{imm8'}}$のとき　WDMレジスタ←imm8 imm8≠$\overline{\text{imm8'}}$のとき　(SP－1，SP－2)←PSW，(SP－3，SP－4)←PS， (SP－5，SP－6)←PC－x[注]，SP←SP－6 IE←0，BRK←0 PC←(20H，21H)，PS←(22H，23H)						
			imm8								$\overline{\text{imm8'}}$															
	BTCLRL	sfrl，imm3，short－label	0	0	0	0	1	1	1	1	1	0	0	1	1	1	0	1	5	(sfrl)のビットNo．imm3＝1のとき PC←PC＋ext－dep8． (sfrl)のビットNo．imm3←0						
			sfrl								imm3															
			disp8																							

注　x：命令のバイト数＋プリフィクスの数

保守／廃止

表A－1 ニモニック一覧（33/33）

キュー操作命令

命令群	ニモニック	オペレーション・コード 7 6 5 4 3 2 1 0	オペレーション・コード 7 6 5 4 3 2 1 0	バイト数	オペレーション	フラグ AC	CY	V	P	S	Z
キュー操作命令	QHOUT	0 0 0 0 1 1 1 1	0 1 1 1 0 0 0 0	4	キューの先頭ブロックを外します。	U	U	U	U	U	×
	QOUT	0 0 0 0 1 1 1 1	0 1 1 1 0 0 0 1	4	キューの任意のブロックを外します。	U	U	U	U	U	×
	QTIN	0 0 0 0 1 1 1 1	0 1 1 1 0 0 1 0	4	キューにブロックをキューイングします。						

FAX専用命令

命令群	ニモニック	オペレーション・コード 7 6 5 4 3 2 1 0	オペレーション・コード 7 6 5 4 3 2 1 0	バイト数	オペレーション	フラグ AC	CY	V	P	S	Z
FAX専用命令	ALBIT	0 0 0 0 1 1 1 1	1 0 0 1 1 0 1 0	2	CH + CL ≧ 16のときBW, DW → DS1 : IYの送信バッファへ出力 CH + CL < 16のときCH + CL → CH, BW, DW → BW 16ビットを越える分，CH + CL − 16 → CH, BW, DW → BW	U	×	U	U	U	×
	COLTRP	0 0 0 0 1 1 1 1	1 0 0 1 1 0 1 1	2	1ライン分の画素データの変化点情報を変化点テーブルに格納(先頭白ラン・レングス)						
	MHENC	0 0 0 0 1 1 1 1	1 0 0 1 0 0 1 1	2	変化点テーブルからMH符号を生成	U	×	U	U	U	×
	MRENC	0 0 0 0 1 1 1 1	1 0 0 1 0 1 1 1	2	変化点テーブルからMR符号を生成	U	×	U	U	U	×
	SCHEOL	0 0 0 0 1 1 1 1	0 1 1 1 1 0 0 0	2	MH/MR符号中の“EOL”を検出	U	×	U	U	U	×
	GETBIT	0 0 0 0 1 1 1 1	0 1 1 1 1 0 0 1	2	画素データの先頭ビットを取り出し，CYフラグに設定	U	×	U	U	U	×
	MHDEC	0 0 0 0 1 1 1 1	0 1 1 1 1 1 0 0	2	MH符号から変化点テーブルを生成	U	×	U	U	U	×
	MRDEC	0 0 0 0 1 1 1 1	0 1 1 1 1 1 0 1	2	MR符号から変化点テーブルを生成	U	×	U	U	U	×
	CNVTRP	0 0 0 0 1 1 1 1	0 1 1 1 1 0 1 0	2	変化点テーブルの1ライン分の変化点情報を画素データに変換						

保守／廃止

付録B　注意事項

B.1　割り込み機能

(1) マクロ・サービスでSFRアクセスを行う場合

マクロ・サービス・チャネルのSFRアドレス指定ポインタには，SFRアドレスの下位16ビットを設定してください。また，上位7ビットにはすべて1を設定してください。

(2) 割り込みモード・コントロール・レジスタの操作

割り込みモード・コントロール・レジスタは，リセット後の初期状態から変更が不要な場合でも，必ず一度設定してください。これは，その後の誤った書き込み動作によるシステムの異常動作を防ぐためです。

なお，このレジスタの操作は，必ず割り込み禁止の状態（DI状態）で行ってください。

(3) マクロ・サービス終了時の割り込み要求発生

マクロ・サービスが終了したときに発生するベクタ割り込み（またはレジスタ・バンク切り替え）処理中に再度同じ割り込み要求が発生すると，割り込み要求制御レジスタのIFフラグ＝1となります。この状態で割り込み要求制御レジスタのMS/$\overline{\text{INT}}$ビットを0→1に変更すると，ただちにマクロ・サービスが実行されます。マクロ・サービス終了後，再び割り込み処理の続きを実行します。

(4) 割り込みが発生しない場合

割り込みが発生するはずのタイミングで発生しない場合は，次に示す内容をまず確認してください。

(a) EI状態になっているか

EI命令を実行したか確認してください。PSWレジスタのIEフラグがセット(1)されていればEI状態です。

(b) 割り込みがマスクされていないか

割り込み要求制御レジスタ（IC××）のビット6（MSK）をリセット(0)して，マスクを解除してください。

(c) 割り込み要求が発生しているか

割り込み要求制御レジスタ（IC××）のビット7（IF）がセット(1)されていれば，割り込み要求は発生しています。このビットが0のときは割り込み要求が発生していません。ソフトウ

保守／廃止

エアで割り込み要求をクリアしていないか確認してください。

(d) 優先順位が高い割り込みを処理中でないか

インサービス・プライオリティ・レジスタ（ISPR）に高い割り込み優先順位に相当するビットがセットされている可能性があります。この場合，V55PIはセットされているビットに対応する優先順位の割り込み処理を現在処理中であると認識し，それより低い優先順位の割り込みを受け付けません。

ISPRレジスタのビットをリセットできるのは，FINT命令だけです。したがって，セットされている優先順位の割り込み処理ルーチン内でFINT命令を実行しているか確認してください。

ISPRレジスタの詳細については，**5.5.4　インサービス・プライオリティ・レジスタ（ISPR）**を参照してください。

(5) FINT命令についての注意事項

FINT命令は，割り込み処理の最後のRETI命令の直前で必ず実行してください。

FINT命令は，ISPRレジスタでセットされているビットの最高優先順位に対応するビットをクリアする命令です。割り込み処理ルーチン内でFINT命令を実行しないと，その割り込みより低い優先順位の割り込みを受け付けられなくなります。

また，割り込み処理の途中でFINT命令を実行するとV55PIは優先順位の管理ができなくなり，処理中の割り込みより低い優先順位の割り込みも受け付けるようになります。したがって，割り込み処理の途中でFINT命令を実行する場合は，優先順位の管理をアプリケーションにより行ってください。

FINT命令の詳細については，**V55PI UM　命令編**を参照してください。

B.2　DMA機能

(1) メイン・チャネル側DMAコントローラにおける転送対象指定の制限

メイン・チャネルを使用する以下のDMA動作モードでは，転送対象にSFRおよびメモリを指定する場合，次のような制限事項がありますので注意してください。

インテリジェントDMAモード-1
インテリジェントDMAモード-2
ネクスト・アドレス指定モード
メモリ-メモリ転送モード

(a) 転送対象指定可能SFRについて

DMA転送対象に指定可能なSFRは，次に示すレジスタに限られます。

それ以外のSFR（割り込み要求制御レジスタ，割り込みマスク・フラグ・レジスタ，インサー

保守／廃止

ビス・プライオリティ・レジスタ，DMA関係の制御レジスタなど）をDMAの転送対象とした場合は，その後の正常動作は保証しません。

特殊機能レジスタの名称	特殊機能レジスタの略称
ポート，ポート7バッファ	P0-P8，P7L，P7H
リアルタイム出力ポート	RTP
タイマ・コンペア・レジスタ	CM00，CM01，CM10，CM11，CM20，CM21，CM22，CM23，CM30，CM31
PWMレジスタ	PWM
UART送信／受信バッファ	TxB0，TxB1，RxB0，RxB1
パラレル・インタフェース・バッファ	PAB
A/Dコンバージョン・リザルト・レジスタ	ADCR0-ADCR3

(b) DMA機能をマクロ・サービスと同時に使用する場合

DMA機能をマクロ・サービスと同時に使用する場合，双方の処理で同一のメモリ・ブロックまたは同一のSFRを転送対象に指定しないでください。

(c) 外部8ビット・バス・モード使用時の注意事項

外部8ビット・バス・モードで使用する場合，16ビットSFR（**表3-5　特殊機能レジスター覧**参照）と外部メモリとの間でDMA転送を実行することはできません（**表6-7　外部データ・バス幅とDMA転送対象**参照）。

(2) DMA転送中のレジスタの読み出しについて

ネクスト・アドレス指定モード，メモリ-メモリ転送モードでは，TCMレジスタの内容がTCレジスタにロードされるタイミングは，次のDMAサイクルを実行する直前です。したがって，ボロー発生直後にTCレジスタを読み出すと，そのままの値が読み出されてしまいます。

(3) DMA割り込み処理内でTCレジスタの値を変更する場合

ネクスト・アドレス指定モードにおいて，DMA割り込み処理内でTCの値を変更する場合は，DMA転送を禁止してから変更してください。また，そのとき必ず次のどちらかの操作を行ってください。

- TCMレジスタに新しく設定するTCの値を書き込む
- DMASレジスタのEND/Fフラグに“1”を書き込む（初期状態に戻すことにより，TCの値がそのままダウンカウントされる）

保守／廃止

ただし，DMA転送を禁止する前に次のDMA要求が発生すると，TCMの値がTCにロードされ実行されてしまいます。

(4) DMAステータス・レジスタ（DMAS）に関する注意事項

DMA転送モードを切り替えても，DMASレジスタの内容は初期化されません。モード切り替え後，必要に応じて初期化してください。

また，このレジスタは1ビット操作命令は使用できません。リセットするビットだけを1にしたデータを8ビット操作命令で書き込んでください。ただし，ビット・テスト命令を使用することはできます。

(5) リフレッシュ・サイクル自動挿入許可時の転送モード

リフレッシュ・サイクルの自動挿入を許可した状態（RFMレジスタのRFENビット＝1）でI/O-メモリ間のDMA転送を行う場合，シングルトランスファ・モードを使用してください。ディマンド・リリース・モードを指定すると，DMARQ入力の立ち下がりに続く最後のDMA転送サイクルが正常に発生しないことがあります。

(6) UART受信割り込み要求をDMAの転送要求ソースとしている場合

DMA機能を使用してUART受信を行う場合，受信エラーが発生しても，受信割り込み要求に従って受信バッファ（RxB）から外部メモリへDMA転送が実行されてしまいます。したがってこの場合，DMA転送後に発生する受信エラー割り込み内で対処してください。

受信エラーを発生したデータは，外部メモリ領域のMAR＋TC＋1(デクリメント時)またはMAR－TC－1（インクリメント時）で示されるアドレスに格納されています。

なお，受信エラー割り込み内で，エラーの発生した受信データを操作するときは，次のDMA転送が発生しないようDMA禁止状態にしてから行ってください。

★

(7) シングルステップ・モード使用上の注意事項

次のような設定内容で，チャネル0のDMA転送中にチャネル1の入出力ポインタ(IOP1)からのデータ読み出し（IN命令またはINM命令）を行う場合，チャネル0の各動作モードにより，いくつかの注意事項があります。

ただし，チャネル0のDMA転送を行っていないときは，この注意事項は発生しません。

(a) 設定内容

- チャネル0の転送モードをシングルステップ・モードに設定し，同時に動作モードを2チャネル動作モード，メモリ-メモリ転送モードまたはインテリジェントDMAモード-2に設定する。
- チャネル1の動作モードをインテリジェントDMAモード-1に設定する。

保守／廃止

(b) チャネル0の各動作モードの注意事項

チャネル0動作モード	チャネル0DCM0値	該当注意事項
2チャネル動作モード（メイン・チャネル）	000000H	①，②
2チャネル動作モード（メイン・チャネル）	000000H以外	②，③
2チャネル動作モード（サブチャネル）	値に依存しない	②，③，④
メモリ-メモリ転送モード	1FFFFFH	①，②
メモリ-メモリ転送モード	1FFFFFH以外	②，③
インテリジェントDMAモード-2	値に依存しない	②，③，⑤[注]

注　チャネル0をインテリジェントDMAモード-2（シングルステップ・モード），チャネル1をインテリジェントDMAモード-1の組み合わせでは，使用しないでください。

① チャネル1のIOP1読み出し時に，不正なチャネル0のINTD0割り込みが，毎回発生します。チャネル0が2チャネル動作モードの場合は，DCM0に000000H以外の値を設定してください。また，チャネル0がメモリ-メモリ転送モードの場合は，DCM0に1FFFFFH以外の値を設定してください。

② チャネル1側のインテリジェントDMAモード-1のDMA転送で，DCM1＝UDC1となってINTD1割り込みが発生したあと，バッファ・フル状態からIN命令によりIOP1領域に対するデータの読み出しを行った場合，UDC1はデクリメント（−1）されますが，そのときに発生すべきINTD1割り込みが発生しません。したがって，チャネル1側のインテリジェントDMAモード-1のDMA転送で，DCM1＝UDC1となって，INTD1バッファ・フル割り込みが発生したあと，本来バッファ・フル状態からIOP1の読み出しを実行するときに発生するINTD1割り込みの処理も実行してください（このUDC1がデクリメント（−1）されたときに発生するINTD1割り込み自体は，あまり使用する用途はないと考えられます）。

③ チャネル1側のインテリジェントDMAモード-1のDMA転送で，DCM1＝UDC1となったときにバッファ・フル・フラグEND/F（DMASのビット2）がセット（1）されます。次にバッファ・フル状態からIN命令によりIOP1領域に対するデータの読み出しを行った場合，UDC1はデクリメント（−1）されますが，本来リセット（0）されるべきEND/Fフラグが，リセットされません。したがって，②と同様のタイミングにおけるチャネル1側のインテリジェントDMAモード-1のDMA転送で，DCM1＝UDC1となったときにバッファ・フル・フラグEND/F(DMASのビット2）をセット（1）し，次に，バッファ・フル状態からIOP1の内容を読み出すときにEND/Fフラグをソフトウエアでリセット（0）してください。

④ チャネル1のIOP1の内容を読み出すと，チャネル1のUDC1がデクリメント（−1）されます。デクリメント（−1）されることにより，UDC1からボローが発生したときに，本来セット（1）されるべきバッファ・エンプティ・フラグEMP（DMASのビット3）が，セットされません。したがって，バッファ・エンプティ・フラグEMPは，使用しないでください（UDC1の値をチェックし1FFFFFH値を判定することでバッファ・エンプティを検出します）。

保守／廃止

⑤　④と同様のタイミングでUDC1からボローが発生したとき，本来チャネル0には影響しないはずですが，チャネル0のUDC0にDCM0値がリロードされ，UDC0の値が変化してしまいます。回避方法はありません。

★ **(8) インテリジェントDMAモード-1使用上の注意事項**

ターミナル・カウンタのボロー発生(TC＝1FFFFFH)直後に，バッファ・フル・フラグEND/F(DMASのビット2）を命令（"1" を書き込む操作実行）によりリセット（0）すると，ボロー情報が消えてしまい，次のIOPリードまたはDMA転送時にDCM値をTCにリロードできなくなります。このため，続いてMAR＋TCで指し示すアドレスにDMA転送をしたときに，MAR＋1FFFFFH番地に転送されてしまい，転送アドレスを誤ってしまいます。したがって，END/Fフラグを命令によりリセット（0）しないでください（"1" を書き込む操作をしないでください)。

B.3　シリアル・インタフェース機能（UART）

(1) 送信動作中の送信禁止操作

送信動作中に$\overline{CTS}$＝1，またはTxE＝0により送信禁止操作を行った場合，送信中のデータは最後まで送られ，送信シフト・レジスタのデータを送信完了した時点で送信を停止します。

送信バッファ(TxB)にデータがある場合，そのデータは送出されず，次に送信許可になるまで送信バッファに保持されます。

(2) 送信割り込みの設定

送信バッファに送信データを書き込む前に送信を許可すると，送信割り込みがただちに発生します。送信許可にする前に，必ず送信割り込みの設定を行ってください。

(3) 受信エラー割り込み発生に伴う受信割り込み

受信エラー割り込みが発生したときは，続いて受信割り込みも発生します。受信エラー割り込みのあとの受信割り込みを受け付けたくない場合は，UARTMレジスタのRxEビットをソフトウエアでクリアしてください。エラーが発生しても，受信バッファ（RxB）へ受信したデータは転送されます。RxEビットをソフトウエアでクリアする際は，オーバラン・エラーの発生を防ぐため必ず受信バッファの値を読み出してください。

(4) 受信BRGレジスタへの値設定

UARTでは，シリアル・クロックと受信データの位相ずれを抑えるため，受信BRGレジスタに7以上の値を設定して使用してください。

保守／廃止

(5) 送受信BRGレジスタの読み出し

送信，受信中には，送信／受信BRGレジスタを読み出さないでください。シフト・クロックの周期が一時的に狂ってしまいます。

(6) 送受信割り込みについて

V55PIでは，送受信割り込みの発生を２回分，割り込みコントローラ側で記憶します。

ただし，ソフトウエアにより割り込みフラグをクリアし，割り込み要求を取り消した場合，割り込み要求は２回分とも取り消されます。

(7) 途中で送信が止まった場合について

途中で送信が止まるという現象が発生した場合は，次に示す内容をまず確認してください。

① 送信割り込み内で送信バッファにデータがセットされているか

送信割り込みは，送信バッファ（TxB）からシフト・レジスタへ送信データが転送された結果TxBが空になったときに発生します。したがって，送信割り込みで送信データを設定する場合は，送信割り込み処理内で必ずTxBにデータを書き込んでください。

② ソフトウエアで割り込みフラグ（IF）をクリアしてないか

送信許可に設定するとすぐに割り込み要求が発生します。したがって，送信許可に設定したあとで割り込み要求制御レジスタ（IC××）を設定する場合，誤って割り込みフラグをクリアしてしまう可能性があります。

B.1(4) 割り込みが発生しない場合についても参照してください。

保守／廃止

B.4　シリアル・インタフェース機能（CSI）

(1) SBIモードでの設定順序

SBIモードでは，受信動作の許可（CSIMレジスタのCRxEビット＝1），送信動作の許可（CSIMレジスタのCTxEビット＝1）の設定を行う前に必ず次の設定を行ってください。

- シリアル・クロックの選択
- SBIモードでのシリアル・データ入出力端子の選択

(2) IOEモードでの受信データのキャンセル

IOEモードでは，受信動作を途中で中止した場合，CSIMレジスタのCTxEビットとCRxEビットをともに0にすることにより，シフト・カウンタがリセットされシフト・レジスタ（SIO）は初期状態に戻ります。

B.5　タイマ機能

○TI入力イベントのカウント・タイミング

イベント・カウント入力端子（TI）のサンプリングはϕ/8ごとに行われます。

カウント精度は次のようになります。

システム・クロックϕの分周比	サンプリング時間	
	f_{XX}＝25 MHz時	f_{XX}＝32 MHz時
$\phi=f_{XX}/2$	0.64 μs	0.50 μs
$\phi=f_{XX}/4$	1.28 μs	1.00 μs
$\phi=f_{XX}/8$	2.56 μs	2.00 μs
$\phi=f_{XX}/16$	5.12 μs	4.00 μs

f_{XX}：クリスタル発振周波数

① 外部イベント入力端子TIにシリアル・クロックを入力し，通信速度を計算する応用回路の場合，通信速度を64 Kbpsとすれば最低でも15.6 μsの精度が必要となります。

② サンプリングの精度を考えると，入力クロックの最大周波数は，$\phi=f_{XX}/2$モードで156 kHz程度までとなります。

ただし，回路上，TI＜ϕ/8未満であれば，誤計算は起こしません(外部イベント入力に対し，高速かつリアルタイムに応答が必要な応用回路にはあまり適していません)。

保守／廃止

B.6　リアルタイム出力ポート機能

ディレイ・カウンタの動作中にディレイありモードからディレイなしモードに切り替えると，次にP7Lバッファの値をリアルタイム出力ポートへ転送するタイミングが発生するまで，P70-P73端子の出力状態は変化しません。

RTP0-RTP3の値をP70-P73へ出力するには，RTPCレジスタのTRGビットの操作が必要です(**10.4.2　リアルタイム出力ポート・コントロール・レジスタ（RTPC）** 参照)。

B.7　A/Dコンバータ機能

(1) STOP命令を実行する場合について

A/Dコンバータの動作中にSTOPモードにする場合は，必ずA/Dコンバータの変換動作を停止しておいてください。これは，A/Dコンバータがサンプリング動作中にSTOP状態になると，サンプリング・ゲートが開いたままとなるためです。A/Dコンバータ・モード・レジスタのCSビット＝0にすると，変換中であってもすべての変換動作を中止します。

(2) A/D変換動作シーケンスの初期化について

外部トリガ端子への有効エッジ入力やA/Dコンバータ・モード・レジスタのTRGビットの再セット(1→1)によって，A/D変換動作シーケンスは初期化されます。ただし，外部端子への有効エッジ入力の周期は，必ず変換動作の1周期よりも長くしてください。繰り返し外部端子へ有効エッジ入力を行ってA/D変換を行わせる場合などで，有効エッジ入力の周期が変換動作シーケンスの1周期より短いと，変換が行われない可能性があります。

(3) 変換結果の取りこぼしについて

A/Dコンバータのセレクト・モード使用時に，A/D変換結果の読み出しをベクタ割り込みで行っている場合，他の割り込み処理などによって，A/D変換終了割り込みが長時間（FRビットが0の場合160クロック，FRビットが1の場合120クロック以上)保留されると変換結果を取りこぼしてしまいます。この変換結果の取りこぼしをソフトウエア的に検出することは困難です。したがって，変換結果の取りこぼしが発生しないよう，次のような対策をとってください。

- 他の割り込み処理時間をA/D変換の時間に対して十分短くする。
- 多重割り込みを使用して，他の割り込み処理中でもA/D変換終了割り込みを受け付けるようにする。
- A/D変換終了割り込みをマクロ・サービスで行うようにする。

保守／廃止

B.8　コーデック機能

(1) データの転送方向

コーデック命令では，画素データ，符号データなどのシリアル・データは必ず先頭ビットがLSB側として扱われます。

(2) 命令実行中断後の再実行について

コーデック命令（SCHEOL，MHENC，MRENC，MHDEC，MRDEC）では，命令実行中断後に再実行する場合，0クリアが必要なパラメータは中断時の状態のままにしておいてください。

(3) データのバッファ・サイズ指定について

コーデック命令（SCHEOL，GETBIT，MHDEC，MRDEC）の入力データである受信データ・バッファ・サイズは0以外の偶数値を設定してください。また，COLTRP命令の画素データ・バッファ・サイズも偶数値を設定してください。

(4) 拡張セグメント・オーバライド・プリフィクスを付加した場合

コーデック命令（COLTRP，SCHEOL，GETBIT，MHDEC，MRDEC，MHENC，MRENC，C-NVTRP）に拡張セグメント・オーバライド・プリフィクス（DS2：，DS3：）を付加した場合，入力パラメータとしてALレジスタで指定されるレジスタ・バンクのオフセット値は，08H，0EHの代わりにそれぞれ00H，02Hが参照されます。

保守／廃止

付録C　レジスタ索引

C.1　50音順

【あ行】

インサービス・プライオリティ・レジスタ（ISPR）　…　178

ウォッチドッグ・タイマ・モード・レジスタ（WDM）　…　397

【か行】

外部割り込みモード・レジスタ（INTM0，INTM1）　…　179

クロックト・シリアル・インタフェース・モード・レジスタ（CSIM0，CSIM1）　…　309

クロックト・シリアルI/Oシフト・レジスタ（SIO0，SIO1）　…　319

キャプチャ・レジスタ（CT00，CT01，CT10）　…　355

コンペア・レジスタ（CM00，CM01，CM10，CM11，CM20，CM21，CM22，CM23，CM30，CM31）　…　355

【さ行】

受信バッファ（RxB0，RxB1）　…　287，319

受信ボー・レート・ジェネレータ・レジスタ（RxBRG0，RxBRG1）　…　274

スタンバイ・コントロール・レジスタ（STBC）　…　430

送信ボー・レート・ジェネレータ・レジスタ（TxBRG0，TxBRG1）　…　274

ソフトウエア・タイマ・カウンタ（STC）　…　442

ソフトウエア・タイマ・カウンタ・コンペア・レジスタ（STMC）　…　442

【た行】

タイマ・コントロール・レジスタ（TMC0，TMC1）　…　369

タイマ出力制御レジスタ（TOC0，TOC1）　…　371

タイマ・レジスタ（TM0-TM3）　…　355

ターミナル・カウンタ（TC0，TC1）　…　242

ターミナル・カウンタ・モジュロ・レジスタ（TCM0，TCM1）　…　243

ディレイ・カウンタ　…　387

【は行】

パラレル・インタフェース・アクノリッジ・インターバル・レジスタ（PAI1，PAI2）　…　338

パラレル・インタフェース・コントロール・レジスタ0（PAC0）　…　328

保守／廃止

パラレル・インタフェース・コントロール・レジスタ1（PAC1）　…　333
パラレル・インタフェース・ステータス・レジスタ（PAS）　…　336
パラレル・インタフェース・バッファ・レジスタ（PAB）　…　339
プリスケーラ・レジスタ（PRS0，PRS1）　…　273
プログラマブル・ウエイト・コントロール・レジスタ（PWC0，PWC1）　…　94
プロセッサ・コントロール・レジスタ（PRC）　…　438，441
プロトコル選択レジスタ（ASP）　…　267
ポート0（P0）　…　417
ポート1（P1）　…　418
ポート2（P2）　…　419
ポート3（P3）　…　420
ポート4（P4）　…　421
ポート5（P5）　…　423
ポート6（P6）　…　424
ポート7（P7）　…　425
ポート8（P8）　…　426
ポート・リード・コントロール・レジスタ（PRDC）　…　413
ポート0モード・レジスタ（PM0）　…　417
ポート2モード・コントロール・レジスタ（PMC2）　…　419
ポート2モード・レジスタ（PM2）　…　420
ポート3モード・コントロール・レジスタ（PMC3）　…　421
ポート3モード・レジスタ（PM3）　…　421
ポート4モード・コントロール・レジスタ（PMC4）　…　422
ポート4モード・レジスタ（PM4）　…　422
ポート5モード・コントロール・レジスタ（PMC5）　…　423
ポート5モード・レジスタ（PM5）　…　424
ポート7バッファ（上位）（P7H）　…　375
ポート7バッファ（下位）（P7L）　…　375
ポート7モード・コントロール・レジスタ（PMC7）　…　384，426
ポート7モード・レジスタ（PM7）　…　426
ポート8モード・コントロール・レジスタ（PMC8）　…　427
ポート8モード・レジスタ（PM8）　…　427

【ま行】

メモリ・ブロック・コントロール・レジスタ（MBC）　…　90

保守／廃止

【ら行】

リフレッシュ・モード・レジスタ（RFM）　…　101
リアルタイム出力ポート（RTP）　…　375
リアルタイム出力ポート・コントロール・レジスタ（RTPC）　…　385
リアルタイム出力ポート・ディレイ指定レジスタ（RTPD）　…　387

【わ行】

割り込みマスク・フラグ・レジスタ（MK0L，MK0H，MK1L，MK1H）　…　174
割り込みモード・コントロール・レジスタ（IMC）　…　172
割り込み要求制御レジスタ（IC9-14，16-32，36，37）　…175

保守／廃止

C.2　アルファベット順

【A】

A/Dコンバージョン・リザルト・レジスタ（ADCR0-ADCR3）　…　409

A/Dコンバータ・モード・レジスタ（ADM）　…　408

ASP（プロトコル選択レジスタ）　…　267

【C】

CM00，CM01，CM10，CM11，CM20，CM21，CM22，CM23，CM30，CM31（コンペア・レジスタ）　…　355

CSIM0，CSIM1（クロックト・シリアル・インタフェース・モード・レジスタ）　…　309

CT00，CT01，CT10（キャプチャ・レジスタ）　…　355

【D】

DMAアップ・ダウン・カウンタ（UDC0，UDC1）　…　243

DMAコントロール・レジスタ（DMAC0，DMAC1）　…　237

DMAコンペア・レジスタ（DCM0，DCM1）　…　244

DMAステータス・レジスタ（DMAS）　…　240

DMAメモリ・アドレス・レジスタ（MAR0，MAR1）　…　243

DMAモード・レジスタ（DMAM0，DMAM1）　…　233

DMAリード・ライト・ポインタ（DPTC0，DPTC1）　…　243

【I】

IC9-14，16-32，36，37（割り込み要求制御レジスタ）　…　175

IMC（割り込みモード・コントロール・レジスタ）　…　172

INTM0，INTM1（外部割り込みモード・レジスタ）　…　179

ISPR（インサービス・プライオリティ・レジスタ）　…　178

【M】

MAR0，MAR1（DMAメモリ・アドレス・レジスタ）　…　243

MBC（メモリ・ブロック・コントロール・レジスタ）　…　90

MK0L，MK0H，MK1L，MK1H（割り込みマスク・フラグ・レジスタ）　…　174

【P】

P0（ポート0）　…　417

P1（ポート1）　…　418

P2（ポート2）　…　419

保守／廃止

P3（ポート3）　…　420
P4（ポート4）　…　421
P5（ポート5）　…　423
P6（ポート6）　…　424
P7（ポート7）　…　425
P8（ポート8）　…　426
PAB（パラレル・インタフェース・バッファ・レジスタ）　…　339
PAC0（パラレル・インタフェース・コントロール・レジスタ0）　…　328
PAC1（パラレル・インタフェース・コントロール・レジスタ1）　…　333
PAI1, PAI2（パラレル・インタフェース・アクノリッジ・インターバル・レジスタ）　…　338
PAS（パラレル・インタフェース・ステータス・レジスタ）　…　336
P7H（ポート7バッファ（上位））　…　375
P7L（ポート7バッファ（下位））　…　375
PM0（ポート0モード・レジスタ）　…　417
PM2（ポート2モード・レジスタ）　…　420
PM3（ポート3モード・レジスタ）　…　421
PM4（ポート4モード・レジスタ）　…　422
PM5（ポート5モード・レジスタ）　…　424
PM7（ポート7モード・レジスタ）　…　426
PM8（ポート8モード・レジスタ）　…　427
PMC2（ポート2モード・コントロール・レジスタ）　…　419
PMC3（ポート3モード・コントロール・レジスタ）　…　421
PMC4（ポート4モード・コントロール・レジスタ）　…　422
PMC5（ポート5モード・コントロール・レジスタ）　…　423
PMC7（ポート7モード・コントロール・レジスタ）　…　384, 426
PMC8（ポート8モード・コントロール・レジスタ）　…　427
PRC（プロセッサ・コントロール・レジスタ）　…　438, 441
PRDC（ポート・リード・コントロール・レジスタ）　…　413
PRS0, PRS1（プリスケーラ・レジスタ）　…　273
PWC0, PWC1（プログラマブル・ウエイト・コントロール・レジスタ）　…　94
PWMコントロール・レジスタ（PWMC）　…　394
PWMレジスタ（PWM）　…　394

【R】

RFM（リフレッシュ・モード・レジスタ）　…　101
RTP（リアルタイム出力ポート）　…　375
RTPC（リアルタイム出力ポート・コントロール・レジスタ）　…　385

RTPD（リアルタイム出力ポート・ディレイ指定レジスタ）　… 387
RxB0, RxB1（受信バッファ）　… 287, 319
RxBRG0, RxBRG1（受信ボー・レート・ジェネレータ・レジスタ）　… 274

【S】
SBIコントロール・レジスタ（SBIC0）　… 312
SIO0, SIO1（クロックト・シリアルI/Oシフト・レジスタ）　… 319
STBC（スタンバイ・コントロール・レジスタ）　… 430
STC（ソフトウエア・タイマ・カウンタ）　… 442
STMC（ソフトウエア・タイマ・カウンタ・コンペア・レジスタ）　… 442

【T】
TC0, TC1（ターミナル・カウンタ）　… 242
TCM0, TCM1（ターミナル・カウンタ・モジュロ・レジスタ）　… 243
TM0-TM3（タイマ・レジスタ）　… 355
TMC0, TMC1（タイマ・コントロール・レジスタ）　… 369
TOC0, TOC1（タイマ出力制御レジスタ）　… 371
TxB0, TxB1（UART送信バッファ）　… 287
TxBRG0, TxBRG1（送信ボー・レート・ジェネレータ・レジスタ）　… 274

【U】
UART送信バッファ（TxB0, TxB1）　… 287
UARTステータス・レジスタ（UARTS0, UARTS1）　… 284
UARTモード・レジスタ（UARTM0, UARTM1）　… 280
UDC0, UDC1（DMAアップ・ダウン・カウンタ）　… 243

【W】
WDM（ウォッチドッグ・タイマ・モード・レジスタ）　…397

〔メ　モ〕

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 半導体第一販売部 半導体第二販売部	〒460 名古屋市中区錦一丁目17番1号（NEC中部ビル）	名古屋 名古屋	(052)222-2170 (052)222-2190
関西支社 半導体第一販売部 半導体第二販売部 半導体第三販売部	〒540 大阪市中央区城見一丁目4番24号（NEC関西ビル）	大阪 大阪 大阪	(06) 945-3178 (06) 945-3200 (06) 945-3208

北海道支社 札幌	(011)231-0161	太田支店 太田	(0276)46-4011	富山支店 富山	(0764)31-8461
東北支社 仙台	(022)267-8740	宇都宮支店 宇都宮	(028)621-2281	三重支店 津	(0592)25-7341
岩手支店 盛岡	(019)651-4344	小山支店 小山	(0285)24-5011	京都支社 京都	(075)344-7824
山形支店 山形	(0236)23-5511	長野支社 松本	(0263)35-1662	神戸支社 神戸	(078)333-3854
郡山支店 郡山	(0249)23-5511	甲府支店 甲府	(0552)24-4141	中国支社 広島	(082)242-5504
いわき支店 いわき	(0246)21-5511	埼玉支社 大宮	(048)641-1411	鳥取支店 鳥取	(0857)27-5311
長岡支店 長岡	(0258)36-2155	立川支社 立川	(0425)26-5981	岡山支店 岡山	(086)225-4455
土浦支店 土浦	(0298)23-6161	千葉支社 千葉	(043)238-8116	四国支社 高松	(0878)36-1200
水戸支店 水戸	(029)226-1717	静岡支社 静岡	(054)255-2211	新居浜支店 新居浜	(0897)32-5001
神奈川支社 横浜	(045)324-5524	北陸支社 金沢	(0762)23-1621	松山支店 松山	(089)945-4149
群馬支店 高崎	(0273)26-1255	福井支店 福井	(0776)22-1866	九州支社 福岡	(092)271-7700

【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)548-7924
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 名古屋市中区錦一丁目17番1号（NEC中部ビル）	名古屋	(052)222-2125
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

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)

保守／廃止

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］V55PI　ユーザーズ・マニュアル　ハードウエア編
（U10514JJ5V1UM00（第５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

キリトリ