

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

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

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

2010年4月1日  
ルネサスエレクトロニクス株式会社

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

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

## ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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

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



ユーザーズ・マニュアル（暫定）

# μPD784976 サブシリーズ

16 ビット・シングルチップ・マイクロコンピュータ

ハードウェア編

---

μPD784975

μPD78F4976

資料番号 U14119JJ1V0UM00（第1版）  
発行年月 November 1999 N CP(K)

© NEC Corporation 1999

[メモ]

# 目 次 要 約

第1章	概 説	...	25
第2章	端子機能	...	35
第3章	CPUアーキテクチャ	...	47
第4章	ポート機能	...	81
第5章	クロック発生回路	...	103
第6章	タイマ/カウンタの概説	...	113
第7章	16ビット・タイマ/イベント・カウンタ	...	115
第8章	8ビットPWMタイマ	...	139
第9章	ウォッチドッグ・タイマ	...	159
第10章	A/Dコンバータ	...	165
第11章	シリアル・インタフェース	...	179
第12章	FIPコントローラ/ドライバ	...	185
第13章	エッジ検出機能	...	201
第14章	割り込み機能	...	203
第15章	スタンバイ機能	...	271
第16章	リセット機能	...	299
第17章	$\mu$ PD78F4976のプログラミング	...	301
第18章	命令のオペレーション	...	305
付録A	開発ツール	...	339
付録B	組み込み用ソフトウェア	...	351
付録C	レジスタ索引	...	353

## CMOSデバイスの一般的注意事項

### 静電気対策（MOS全般）

**注意** MOSデバイス取り扱いの際は静電気防止を心がけてください。

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

また、MOSデバイスを実装したボードについても同様の扱いをしてください。

### 未使用入力の処理（CMOS特有）

**注意** CMOSデバイスの入力レベルは固定してください。

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

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

### 初期化以前の状態（MOS全般）

**注意** 電源投入時、MOSデバイスの初期状態は不定です。

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

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

FIPは、日本電気株式会社の登録商標です。

IEBusは、日本電気株式会社の商標です。

MS-DOSおよびWindowsは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。

IBM DOS, PC/AT, PC DOSは、米国IBM社の商標です。

HP9000シリーズ300, HP9000シリーズ700, HP-UXは、米国ヒューレット・パカード社の商標です。

SPARCstationは、米国SPARC International, Inc.の商標です。

Solaris, SunOSは、米国サン・マイクロシステムズ社の商標です。

NEWS, NEWS-OSは、ソニー株式会社の商標です。

OSF/Motifは、Open Software Foundation, Inc.の商標です。

イーサネットは、米国ゼロックス社の商標です。

TRONは、The Realtime Operating system Nucleusの略称です。

ITRONは、Industrial TRONの略称です。

本製品のうち、外国為替および外国貿易管理法の規定により規制貨物等（または役務）に該当するものについては、日本国外に輸出する際に、同法に基づき日本国政府の輸出許可が必要です。

非該当品：μPD78F4976GF-3BA

ユーザ判定品：μPD784975GF-x x x-3BA

- 本資料は、この製品の企画段階で作成していますので、予告なしに内容を変更することがあります。また本資料で扱う製品の製品化を中止することがあります。
  - 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財産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。
  - 本資料に記載された回路、ソフトウェア、及びこれらに付随する情報は、半導体製品の動作例、応用例を説明するためのものです。従って、これら回路・ソフトウェア・情報をお客様の機器に使用される場合には、お客様の責任において機器設計をしてください。これらの使用に起因するお客様もしくは第三者の損害に対して、当社は一切その責を負いません。
  - 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。

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



# はじめに

- 対象者** このマニュアルは、 $\mu$ PD784976サブシリーズの機能を理解し、その応用システムやアプリケーションを設計、開発するユーザのエンジニアを対象としています。
- 目的** このマニュアルは、次の構成に示す機能をユーザに理解していただくことを目的としています。
- 構成**  $\mu$ PD784976サブシリーズのマニュアルは、このマニュアルと命令編（78K/ シリーズ共通）の2冊に分かれています。

$\mu$ PD784976サブシリーズ ユーザズ・マニュアル (このマニュアル)	78K/ シリーズ ユーザズ・マニュアル 命令編
端子機能	CPU機能
内部ブロック機能	命令セット
割り込み	命令の説明
その他の内蔵周辺機能	

- 読み方** このマニュアルを読むにあたっては、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。

一通りの機能を理解しようとするとき

目次に従って読んでください。

レジスタ・フォーマットの見方

ビット番号を で囲んでいるものは、そのビット名称がRA78K4では予約語に、CC78K4ではsfrbit.hというヘッダ・ファイルで定義済みとなっているものです。

レジスタ名が分かっている、レジスタの詳細を確認するとき

**付録C レジスタ索引**を利用してください。

$\mu$ PD784976サブシリーズの命令機能の詳細を知りたいとき

別冊の78K/ シリーズ ユーザズ・マニュアル 命令編(U10905J)を参照してください。

$\mu$ PD784976サブシリーズの各種機能の応用例を知りたいとき

別冊の78K/ シリーズ アプリケーション・ノート ソフトウェア基礎編(U10095J)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	: $\overline{x \times x}$ (端子, 信号名称に上線)
	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... $x \times x \times x$ または $x \times x \times x B$
		10進数... $x \times x \times x$
		16進数... $x \times x \times x H$

## 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。

### デバイスの関連資料

資料名	資料番号	
	和文	英文
μPD784976サブシリーズ ユーザーズ・マニュアル	このマニュアル	作成予定
μPD784975 ペーパー・マシン	U14116J	U14116E
μPD78F4976 ペーパー・マシン	作成予定	作成予定
78K/ シリーズ インストラクション活用表	U10594J	-
78K/ シリーズ インストラクション・セット	U10595J	-
78K/ シリーズ ユーザーズ・マニュアル 命令編	U10905J	-
78K/ シリーズ アプリケーション・ノート ソフトウェア基礎編	U10095J	-

注意 上記関連資料は予告なしに内容を変更することがあります。設計などには必ず最新の資料をご使用ください。

### 開発ツールの資料（ユーザズ・マニュアル）

資料名		資料番号	
		和文	英文
RA78K4 アセンブラ・パッケージ	言語編	U11162J	U11162E
	操作編	U11334J	U11334E
RA78K 構造化アセンブラ・プリプロセッサ		U11743J	U11743E
CC78K4 Cコンパイラ	言語編	U11571J	U11571E
	操作編	U11572J	U11572E
IE-78K4-NS		U13356J	U13356E
IE-784000-R		U12903J	U12903E
IE-784976-NS-EM1		作成予定	-
EP-78064		EEU-934	EEU-1469
SM78K4 システム・シミュレータ Windowsベース	レファレンス編	U10093J	U10093E
SM78Kシリーズ システム・シミュレータ	外部部品ユーザオープン インタフェース仕様編	U10092J	U10092E
ID78K4-NS 統合ディバッガ PCベース	レファレンス編	U12796J	U12796E
ID78K4 統合ディバッガ Windowsベース	レファレンス編	U10440J	U10440E
ID78K4 統合ディバッガ HP-UX™, SunOS™, NEWS-OS™ベース	レファレンス編	U11960J	U11960E

### 組み込み用ソフトウェアの資料（ユーザズ・マニュアル）

資料名		資料番号	
		和文	英文
78K/ シリーズ リアルタイムOS	基礎編	U10603J	U10603E
	インストール編	U10604J	U10604E
	ディバッガ編	U10364J	-
78K/ シリーズ用OS MX78K4	基礎編	U11779J	-

### その他の資料

資料名		資料番号	
		和文	英文
SEMICONDUCTORS SELECTION GUIDE Products & Packages ( CD-ROM )		X13769X	
半導体デバイス 実装マニュアル		C10535J	C10535E
NEC半導体デバイスの品質水準		C11531J	C11531E
NEC半導体デバイスの信頼性品質管理		C10983J	C10983E
静電気放電（ESD）破壊対策ガイド		C11892J	C11892E
半導体 品質／信頼性ハンドブック		C12769J	-
マイクロコンピュータ関連製品ガイド 社外メーカ編		U11416J	-

**注意** 上記関連資料は、予告なしに内容を変更することがあります。設計などには、必ず最新の資料をご使用ください。

[メモ]

# 目 次

<b>第1章 概 説</b> ...	25
1.1 特 徴 ...	27
1.2 応用分野 ...	27
1.3 オーダ情報 ...	28
1.4 端子接続図 (Top View) ...	29
1.5 ブロック図 ...	31
1.6 機能概要 ...	32
1.7 マスク・オプションについて ...	34
<b>第2章 端子機能</b> ...	35
2.1 端子機能一覧 ...	35
2.2 端子機能の説明 ...	38
2.2.1 P00-P03 (Port0) ...	38
2.2.2 P10-P17 (Port1) ...	38
2.2.3 P20, P25-P27 (Port2) ...	38
2.2.4 P40-P47 (Port4) ...	39
2.2.5 P50-P57 (Port5) ...	39
2.2.6 P60-P67 (Port6) ...	39
2.2.7 P70-P77 (Port7) ...	40
2.2.8 P80-P87 (Port8) ...	40
2.2.9 P90-P97 (Port9) ...	41
2.2.10 P100-P107 (Port10) ...	41
2.2.11 FIP0-FIP15 ...	41
2.2.12 V <sub>LOAD</sub> ...	41
2.2.13 AV <sub>DD</sub> ...	41
2.2.14 AV <sub>SS</sub> ...	42
2.2.15 $\overline{\text{RESET}}$ ...	42
2.2.16 X1, X2 ...	42
2.2.17 V <sub>DD0</sub> -V <sub>DD2</sub> ...	42
2.2.18 V <sub>SS0</sub> , V <sub>SS1</sub> ...	42
2.2.19 V <sub>PP</sub> ( $\mu$ PD78F4976のみ ) ...	42
2.2.20 IC ( マスクROM製品のみ ) ...	42
2.3 端子の入出力回路と未使用端子の処理 ...	43
<b>第3章 CPUアーキテクチャ</b> ...	47
3.1 メモリ空間 ...	47
3.2 内部ROM領域 ...	51
3.3 ベース領域 ...	52
3.3.1 ベクタ・テーブル領域 ...	53
3.3.2 CALLT命令テーブル領域 ...	54
3.3.3 CALLF命令エントリ領域 ...	54
3.4 内部データ領域 ...	55
3.4.1 内部RAM領域 ...	56
3.4.2 特殊機能レジスタ (SFR) 領域 ...	59

3.5	$\mu$ PD78F4976のメモリ・マッピング	...	60
3.6	制御レジスタ	...	61
3.6.1	プログラム・カウンタ (PC)	...	61
3.6.2	プログラム・ステータス・ワード (PSW)	...	61
3.6.3	RSSビットの使用法	...	65
3.6.4	スタック・ポインタ (SP)	...	68
3.7	汎用レジスタ	...	72
3.7.1	構成	...	72
3.7.2	機能	...	74
3.8	特殊機能レジスタ (SFR)	...	77
3.9	注意事項	...	80
<b>第4章 ポート機能</b> ... 81			
4.1	ポートの機能	...	81
4.2	ポートの構成	...	84
4.2.1	ポート0	...	84
4.2.2	ポート1	...	85
4.2.3	ポート2	...	86
4.2.4	ポート4	...	89
4.2.5	ポート5	...	90
4.2.6	ポート6	...	91
4.2.7	ポート7	...	94
4.2.8	ポート8	...	95
4.2.9	ポート9	...	96
4.2.10	ポート10	...	97
4.3	ポート機能を制御するレジスタ	...	98
4.4	ポート機能の動作	...	101
4.4.1	入出力ポートへの書き込み	...	101
4.4.2	入出力ポートからの読み出し	...	101
4.4.3	入出力ポートでの演算	...	102
4.5	マスク・オプションの選択	...	102
<b>第5章 クロック発生回路</b> ... 103			
5.1	クロック発生回路の機能	...	103
5.2	クロック発生回路の構成	...	103
5.3	制御レジスタ	...	104
5.4	メイン・システム・クロック発振回路	...	108
5.4.1	分周回路	...	110
5.5	クロック発生回路の動作	...	111
5.6	CPUクロックの切り替え	...	111
<b>第6章 タイマ/カウンタの概説</b> ... 113			
<b>第7章 16ビット・タイマ/イベント・カウンタ</b> ... 115			
7.1	機能	...	115
7.2	構成	...	116
7.3	制御レジスタ	...	120

7.4	動作	...	124
7.4.1	インターバル・タイマ(16ビット)としての動作	...	124
7.4.2	パルス幅測定としての動作	...	126
7.4.3	外部イベント・カウンタとしての動作	...	133
7.5	注意事項	...	136
<b>第8章 8ビットPWMタイマ</b> ... 139			
8.1	8ビットPWMタイマの機能	...	139
8.2	8ビットPWMタイマの構成	...	140
8.3	8ビットPWMタイマを制御するレジスタ	...	143
8.4	8ビットPWMタイマの動作	...	146
8.4.1	インターバル・タイマ(8ビット動作)としての動作	...	146
8.4.2	外部イベント・カウンタとしての動作	...	149
8.4.3	方形波出力(8ビット分解能)としての動作	...	150
8.4.4	8ビットPWM出力としての動作	...	151
8.5	8ビットPWMタイマの注意事項	...	156
<b>第9章 ウォッチドッグ・タイマ</b> ... 159			
9.1	構成	...	159
9.2	制御レジスタ	...	160
9.3	動作	...	162
9.4	注意事項	...	162
9.4.1	ウォッチドッグ・タイマ使用時の一般的な注意事項	...	162
9.4.2	μPD784976サブシリーズのウォッチドッグ・タイマに関する注意事項	...	163
<b>第10章 A/Dコンバータ</b> ... 165			
10.1	A/Dコンバータの機能	...	165
10.2	A/Dコンバータの構成	...	165
10.3	A/Dコンバータを制御するレジスタ	...	168
10.4	A/Dコンバータの動作	...	170
10.4.1	A/Dコンバータの基本動作	...	170
10.4.2	入力電圧と変換結果	...	172
10.4.3	A/Dコンバータの動作モード	...	173
10.5	A/Dコンバータの注意事項	...	174
<b>第11章 シリアル・インタフェース</b> ... 179			
11.1	シリアル・インタフェースの機能	...	179
11.2	シリアル・インタフェースの構成	...	179
11.3	シリアル・インタフェースを制御するレジスタ	...	181
11.4	シリアル・インタフェースの動作	...	182
11.4.1	動作停止モード	...	182
11.4.2	3線式シリアルI/Oモード	...	183
<b>第12章 FIPコントローラ/ドライバ</b> ... 185			
12.1	FIPコントローラ/ドライバの機能	...	185
12.2	FIPコントローラ/ドライバの構成	...	186

- 12.3 FIPコントローラ/ドライバを制御するレジスタ ... 187
  - 12.3.1 制御レジスタ ... 187
  - 12.3.2 1表示期間とブランキング幅 ... 192
- 12.4 表示データ・メモリ ... 193
- 12.5 キー・スキャン・フラグとキー・スキャン・データ ... 194
  - 12.5.1 キー・スキャン・フラグ ... 194
  - 12.5.2 キー・スキャン・データ ... 194
- 12.6 蛍光表示管のもれ発光について ... 195
- 12.7 全損失の計算方法 ... 197

## 第13章 エッジ検出機能 ... 201

- 13.1 制御レジスタ ... 201
- 13.2 P64, P65, P67端子のエッジ検出 ... 202

## 第14章 割り込み機能 ... 203

- 14.1 割り込み要求ソース ... 204
  - 14.1.1 ソフトウエア割り込み ... 205
  - 14.1.2 オペランド・エラー割り込み ... 205
  - 14.1.3 ノンマスカブル割り込み ... 205
  - 14.1.4 マスカブル割り込み ... 205
- 14.2 割り込み処理モード ... 206
  - 14.2.1 ベクタ割り込み処理 ... 206
  - 14.2.2 マクロ・サービス ... 206
  - 14.2.3 コンテキスト・スイッチング ... 206
- 14.3 割り込み処理制御レジスタ ... 207
  - 14.3.1 割り込み制御レジスタ ... 209
  - 14.3.2 割り込みマスク・レジスタ (MK0) ... 213
  - 14.3.3 インサービス・プライオリティ・レジスタ (ISPR) ... 215
  - 14.3.4 割り込みモード・コントロール・レジスタ (IMC) ... 216
  - 14.3.5 ウォッチドッグ・タイマ・モード・レジスタ (WDM) ... 217
  - 14.3.6 割り込み選択コントロール・レジスタ (SNMI) ... 218
  - 14.3.7 プログラム・ステータス・ワード (PSW) ... 219
- 14.4 ソフトウエア割り込みの受け付け動作 ... 219
  - 14.4.1 BRK命令によるソフトウエア割り込みの受け付け動作 ... 219
  - 14.4.2 BRKCS命令によるソフトウエア割り込み (ソフトウエア・コンテキスト・スイッチング)の受け付け動作 ... 220
- 14.5 オペランド・エラー割り込みの受け付け動作 ... 221
- 14.6 ノンマスカブル割り込みの受け付け動作 ... 222
- 14.7 マスカブル割り込みの受け付け動作 ... 225
  - 14.7.1 ベクタ割り込み ... 227
  - 14.7.2 コンテキスト・スイッチング ... 227
  - 14.7.3 マスカブル割り込みの優先順位 ... 229
- 14.8 マクロ・サービス機能 ... 235
  - 14.8.1 マクロ・サービスの概要 ... 235
  - 14.8.2 マクロ・サービスの種類 ... 235
  - 14.8.3 マクロ・サービスの基本動作 ... 238
  - 14.8.4 マクロ・サービス終了時の動作 ... 239
  - 14.8.5 マクロ・サービス制御レジスタ ... 242
  - 14.8.6 マクロ・サービス・タイプA ... 245
  - 14.8.7 マクロ・サービス・タイプB ... 249



- 14.8.8 マクロ・サービス・タイプC ... 254
- 14.8.9 カウンタ・モード ... 261
- 14.9 割り込み要求およびマクロ・サービスが一時的に保留される場合 ... 263
- 14.10 割り込みおよびマクロ・サービスで一時的に実行が中断される命令 ... 264
- 14.11 割り込みおよびマクロ・サービスの動作タイミング ... 264
  - 14.11.1 割り込みの受け付け処理時間 ... 265
  - 14.11.2 マクロ・サービスの処理時間 ... 266
- 14.12 割り込み機能を初期状態に戻す方法 ... 267
- 14.13 注意事項 ... 268

## 第15章 スタンバイ機能 ... 271

- 15.1 構成と機能 ... 271
- 15.2 制御レジスタ ... 272
  - 15.2.1 スタンバイ・コントロール・レジスタ (STBC) ... 272
  - 15.2.2 発振安定時間指定レジスタ (OSTS) ... 274
- 15.3 HALTモード ... 276
  - 15.3.1 HALTモードの設定および動作状態 ... 276
  - 15.3.2 HALTモードの解除 ... 276
- 15.4 STOPモード ... 284
  - 15.4.1 STOPモードの設定および動作状態 ... 284
  - 15.4.2 STOPモードの解除 ... 285
- 15.5 IDLEモード ... 290
  - 15.5.1 IDLEモードの設定および動作状態 ... 290
  - 15.5.2 IDLEモードの解除 ... 291
- 15.6 STOP/IDLEモード使用時のチェック項目 ... 296
- 15.7 注意事項 ... 298

## 第16章 リセット機能 ... 299

## 第17章 $\mu$ PD78F4976のプログラミング ... 301

- 17.1 通信方式の選択 ... 301
- 17.2 フラッシュ・メモリ・プログラミングの機能 ... 302
- 17.3 Flashpro の接続 ... 303

## 第18章 命令のオペレーション ... 305

- 18.1 凡 例 ... 305
- 18.2 オペレーション一覧 ... 309
- 18.3 アドレッシング別命令一覧 ... 334

## 付録A 開発ツール ... 339

- A.1 言語処理用ソフトウェア ... 340
- A.2 フラッシュ・メモリ書き込み用ツール ... 343
- A.3 デバッグ用ツール ... 344
  - A.3.1 ハードウェア ... 344
  - A.3.2 ソフトウェア ... 346
- A.4 変換ソケット (EV-9200GF-100) ... 348

## 付録B 組み込み用ソフトウェア ... 351

付録C レジスタ索引 ... 353

C.1 レジスタ索引 (50音順) ... 353

C.2 レジスタ索引 (アルファベット順) ... 355

## 図の目次 (1/5)

図番号	タイトル, ページ
2 - 1	端子の入出力回路一覧 ... 44
3 - 1	メモリ拡張モード・レジスタ (MM) のフォーマット ... 47
3 - 2	μPD784975のメモリ・マップ ... 49
3 - 3	μPD78F4976のメモリ・マップ ... 50
3 - 4	内部RAMのメモリ・マップ ... 57
3 - 5	内部メモリ・サイズ切り替えレジスタ (IMS) のフォーマット ... 60
3 - 6	プログラム・カウンタ (PC) のフォーマット ... 61
3 - 7	プログラム・ステータス・ワード (PSW) のフォーマット ... 62
3 - 8	スタック・ポインタ (SP) のフォーマット ... 68
3 - 9	スタック領域へ退避されるデータ ... 69
3 - 10	スタック領域から復帰されるデータ ... 70
3 - 11	汎用レジスタのフォーマット ... 72
3 - 12	汎用レジスタのアドレス ... 73
4 - 1	ポートの種類 ... 81
4 - 2	P00-P03のブロック図 ... 84
4 - 3	P10-P17のブロック図 ... 85
4 - 4	P20, P25のブロック図 ... 86
4 - 5	P26のブロック図 ... 87
4 - 6	P27のブロック図 ... 88
4 - 7	P40-P47のブロック図 ... 89
4 - 8	P50-P57のブロック図 ... 90
4 - 9	P60, P64, P65, P67のブロック図 ... 91
4 - 10	P61のブロック図 ... 92
4 - 11	P62, P63, P66のブロック図 ... 93
4 - 12	P70-P77のブロック図 ... 94
4 - 13	P80-P87のブロック図 ... 95
4 - 14	P90-P97のブロック図 ... 96
4 - 15	P100-P107のブロック図 ... 97
4 - 16	ポート・モード・レジスタのフォーマット ... 99
4 - 17	プルアップ抵抗オプション・レジスタ 2 (PU2) のフォーマット ... 99
4 - 18	プルアップ抵抗オプション・レジスタ (PUO) のフォーマット ... 100
5 - 1	クロック発生回路のブロック図 ... 103
5 - 2	スタンバイ・コントロール・レジスタ (STBC) のフォーマット ... 105
5 - 3	発振モード選択レジスタ (CC) のフォーマット ... 106
5 - 4	発振安定時間指定レジスタ (OSTS) のフォーマット ... 107
5 - 5	メイン・システム・クロック発振回路の外付け回路 ... 108
5 - 6	発振子の接続の悪い例 ... 109

## 図の目次 (2/5)

図番号	タイトル, ページ
5 - 7	CPUクロックの切り替え ... 112
6 - 1	タイマ/カウンタのブロック図 ... 113
7 - 1	16ビット・タイマ/イベント・カウンタ (TM0) のブロック図 ... 116
7 - 2	16ビット・タイマ・モード・コントロール・レジスタ0 (TMC0) のフォーマット ... 121
7 - 3	キャプチャ/コンペア・コントロール・レジスタ0 (CRC0) のフォーマット ... 122
7 - 4	プリスケラ・モード・レジスタ0 (PRM0) のフォーマット ... 123
7 - 5	インターバル・タイマ動作時の制御レジスタ設定内容 ... 124
7 - 6	インターバル・タイマの構成図 ... 125
7 - 7	インターバル・タイマ動作のタイミング ... 125
7 - 8	フリーランニング・カウンタとキャプチャ・レジスタ1本によるパルス幅測定時の制御レジスタ設定内容 ... 126
7 - 9	フリーランニング・カウンタによるパルス幅測定の構成図 ... 127
7 - 10	フリーランニング・カウンタとキャプチャ・レジスタ1本によるパルス幅測定動作のタイミング (両エッジ 指定時) ... 127
7 - 11	フリーランニング・カウンタによる2つのパルス幅測定時の制御レジスタ設定内容 ... 128
7 - 12	立ち上がりエッジ指定時のCR01キャプチャ動作 ... 129
7 - 13	フリーランニング・カウンタによるパルス幅測定動作のタイミング (両エッジ指定時) ... 129
7 - 14	フリーランニング・カウンタとキャプチャ・レジスタ2本によるパルス幅測定時の制御レジスタ設定内容 ... 130
7 - 15	フリーランニング・カウンタとキャプチャ・レジスタ2本によるパルス幅測定動作のタイミング (立ち上が りエッジ指定時) ... 131
7 - 16	リスタートによるパルス幅測定時の制御レジスタ設定内容 ... 132
7 - 17	リスタートによるパルス幅測定動作のタイミング (立ち上がりエッジ指定時) ... 133
7 - 18	外部イベント・カウンタ・モード時の制御レジスタ設定内容 ... 134
7 - 19	外部イベント・カウンタの構成図 ... 134
7 - 20	外部イベント・カウンタ動作のタイミング (立ち上がりエッジ指定時) ... 135
7 - 21	16ビット・タイマ・カウンタ0のスタート・タイミング ... 136
7 - 22	タイマ・カウント動作中のコンペア・レジスタの変更後のタイミング ... 136
7 - 23	キャプチャ・レジスタのデータ保持タイミング ... 137
7 - 24	OVF0フラグの動作のタイミング ... 138
8 - 1	8ビットPWMタイマ50 (TM50) のブロック図 ... 140
8 - 2	8ビットPWMタイマ51 (TM51) のブロック図 ... 141
8 - 3	タイマ・クロック選択レジスタ5n (TCL5n) のフォーマット ... 143
8 - 4	8ビット・タイマ・コントロール・レジスタ5n (TMC5n) のフォーマット ... 145
8 - 5	インターバル・タイマ動作のタイミング ... 147
8 - 6	外部イベント・カウンタ動作のタイミング (立ち上がりエッジ指定時) ... 149
8 - 7	PWM出力の動作タイミング ... 152

## 図の目次 (3/5)

図番号	タイトル, ページ
8 - 8	CR5n変更による動作のタイミング ... 153
8 - 9	16ビット分解能カスケード接続モード ... 155
8 - 10	8ビット・カウンタ5n (TM5n) のスタート・タイミング ... 156
8 - 11	タイマ・カウント動作中のコンペア・レジスタの変更後のタイミング ... 156
9 - 1	ウォッチドッグ・タイマのブロック図 ... 159
9 - 2	ウォッチドッグ・タイマ・モード・レジスタ (WDM) のフォーマット ... 161
10 - 1	A/Dコンバータのブロック図 ... 166
10 - 2	A/Dコンバータ・モード・レジスタのフォーマット ... 168
10 - 3	A/Dコンバータ入力選択レジスタのフォーマット ... 169
10 - 4	A/Dコンバータの基本動作 ... 171
10 - 5	アナログ入力電圧とA/D変換結果の関係 ... 172
10 - 6	ソフトウェア・スタートによるA/D変換動作 ... 173
10 - 7	スタンバイ・モード時の消費電流を低減させる方法例 ... 174
10 - 8	アナログ入力端子の処理 ... 175
10 - 9	A/D変換終了割り込み要求発生タイミング ... 176
10 - 10	AVDD端子の処理 ... 176
11 - 1	シリアル・インタフェースのブロック図 ... 180
11 - 2	シリアル動作モード・レジスタnのフォーマット ... 181
11 - 3	3線式シリアルI/Oモードのタイミング ... 184
12 - 1	FIPコントローラ/ドライバのブロック図 ... 186
12 - 2	表示モード・レジスタ0のフォーマット ... 188
12 - 3	表示モード・レジスタ1のフォーマット ... 190
12 - 4	表示モード・レジスタ2のフォーマット ... 191
12 - 5	FIP出力信号のブランキング幅 ... 192
12 - 6	表示データ・メモリのアドレス配置とFIP出力の関係 (FIP出力48本, 16パターンの場合) ... 193
12 - 7	ブランキング時間が短いことによるもれ発光の概念図 ... 195
12 - 8	Cseによるもれ発光の説明図 ... 196
12 - 9	Cseによるもれ発光の概念図 ... 197
12 - 10	許容全損失 $P_T$ ( $T_A = -40 \sim +85$ ) ... 197
12 - 11	10セグメント-11桁表示時の表示データ・メモリの内容とFIP出力の関係 ... 199
13 - 1	外部割り込み立ち上がりエッジ許可レジスタ (EGP0), 外部割り込み立ち下がりエッジ許可レジスタ (EGN0) のフォーマット ... 201
13 - 2	P64, P65, P67端子のエッジ検出 ... 202

## 図の目次 (4/5)

図番号	タイトル, ページ
14 - 1	割り込み制御レジスタ ( x x ICn ) ... 211
14 - 2	割り込みマスク・レジスタ ( MK0 ) のフォーマット ... 214
14 - 3	インサース・プライオリティ・レジスタ ( ISPR ) のフォーマット ... 215
14 - 4	割り込みモード・コントロール・レジスタ ( IMC ) のフォーマット ... 216
14 - 5	ウォッチドッグ・タイマ・モード・レジスタ ( WDM ) のフォーマット ... 217
14 - 6	割り込み選択コントロール・レジスタ ( SNMI ) のフォーマット ... 218
14 - 7	プログラム・ステータス・ワード ( PSWL ) のフォーマット ... 219
14 - 8	BRKCS命令の実行によるコンテキスト・スイッチング動作 ... 220
14 - 9	BRKCS命令によるソフトウェア割り込みからの復帰動作 ( RETCSB命令の動作 ) ... 221
14 - 10	ノンマスカブル割り込み要求の受け付け動作 ... 223
14 - 11	割り込み要求受け付け処理アルゴリズム ... 226
14 - 12	割り込み要求の発生によるコンテキスト・スイッチング動作 ... 227
14 - 13	RETCS命令によるコンテキスト・スイッチング機能を使用した割り込みからの復帰動作 ... 228
14 - 14	割り込み処理中に他の割り込み要求が発生した場合の処理例 ... 230
14 - 15	同時発生した割り込み要求の処理例 ... 233
14 - 16	IMCの設定によるレベル3の割り込みの受け付け動作の違い ... 234
14 - 17	ベクタ割り込みとマクロ・サービスの処理の違い ... 235
14 - 18	マクロ・サービス処理シーケンス ... 238
14 - 19	VCIE = 0のときのマクロ・サービス終了時の動作 ... 240
14 - 20	VCIE = 1のときのマクロ・サービス終了時の動作 ... 241
14 - 21	マクロ・サービス・コントロール・ワードのフォーマット ... 242
14 - 22	マクロ・サービス・モード・レジスタのフォーマット ... 243
14 - 23	マクロ・サービス・データ転送処理フロー ( タイプA ) ... 246
14 - 24	タイプAのマクロ・サービス・チャンネル ... 248
14 - 25	マクロ・サービス・データ転送処理フロー ( タイプB ) ... 250
14 - 26	タイプBのマクロ・サービス・チャンネル ... 251
14 - 27	外部割り込みに同期したパラレル・データ入力 ... 252
14 - 28	パラレル・データ入力のタイミング ... 253
14 - 29	マクロ・サービス・データ転送処理フロー ( タイプC ) ... 255
14 - 30	タイプCのマクロ・サービス・チャンネル ... 258
14 - 31	マクロ・サービス・データ転送処理フロー ( カウンタ・モード ) ... 261
14 - 32	カウンタ・モード ... 262
14 - 33	エッジの回数のカウント ... 262
14 - 34	割り込み要求の発生と受け付け ( 単位 : クロック = 1/fCLK ) ... 264
15 - 1	スタンバイ・モードの遷移図 ... 271
15 - 2	スタンバイ・コントロール・レジスタ ( STBC ) のフォーマット ... 273

## 図の目次 (5/5)

図番号	タイトル, ページ
15 - 3	発振安定時間指定レジスタ (OSTS) のフォーマット ... 275
15 - 4	HALTモード解除後の動作 ... 278
15 - 5	STOPモード解除後の動作 ... 286
15 - 6	INTP0-INTP2入力によるSTOPモードの解除 ... 289
15 - 7	IDLEモード解除後の動作 ... 292
16 - 1	リセット期間中のメイン・システム・クロックの発振 ... 299
16 - 2	リセット信号の受け付け ... 300
17 - 1	通信方式選択フォーマット ... 302
17 - 2	3線式シリアルI/O方式でのFlashpro の接続 (3線式シリアルI/O0を使用する場合) ... 303
A - 1	開発ツール構成 ... 341
A - 2	EV-9200GF-100 外形図 (参考) (単位: mm) ... 348
A - 3	EV-9200GF-100 基板取り付け推奨パターン (参考) (単位: mm) ... 349

## 表の目次 (1/2)

表番号	タイトル, ページ
1 - 1	マスクROM製品のマスク・オプション一覧 ... 34
2 - 1	各端子の入出力回路タイプ ... 43
3 - 1	ベクタ・テーブル・アドレス ... 53
3 - 2	内部RAM領域一覧 ... 56
3 - 3	内部メモリ・サイズ切り替えレジスタ (IMS) の設定値 ... 61
3 - 4	レジスタ・バンクの選択 ... 64
3 - 5	機能名称-絶対名称の対応 ... 76
3 - 6	特殊機能レジスタ (SFR) 一覧 ... 78
4 - 1	ポートの機能 ... 82
4 - 2	ポートの構成 ... 84
4 - 3	兼用機能使用時のポート・モード・レジスタ, 出力ラッチの設定 ... 98
4 - 4	マスクROM製品のマスク・オプションと $\mu$ PD78F4976との比較 ... 102
5 - 1	クロック発生回路の構成 ... 103
6 - 1	タイマ・カウンタの動作 ... 113
7 - 1	16ビット・タイマ/イベント・カウンタ (TM0) の構成 ... 116
7 - 2	TI00端子の有効エッジとCR00のキャプチャ・トリガ ... 118
7 - 3	TI051端子の有効エッジとCR00のキャプチャ・トリガ ... 118
7 - 4	TI00端子の有効エッジとCR01のキャプチャ・トリガ ... 119
8 - 1	8ビットPWMタイマの構成 ... 140
10 - 1	A/Dコンバータの構成 ... 165
11 - 1	シリアル・インタフェースの構成 ... 179
12 - 1	FIP出力端子とポートの兼用端子対応表 ... 185
12 - 2	FIPコントローラ/ドライバの構成 ... 186
14 - 1	割り込み要求の処理形態 ... 203
14 - 2	割り込み要求ソースの種類 ... 204
14 - 3	制御レジスタ一覧 ... 207



## 表の目次 (2/2)

表番号	タイトル, ページ
14 - 4	割り込み要求ソースに対する割り込み制御レジスタのフラグ一覧 ... 208
14 - 5	多重割り込み処理 ... 229
14 - 6	マクロ・サービスが使用可能な割り込み ... 236
14 - 7	割り込み受け付け処理時間 ... 265
14 - 8	マクロ・サービス処理時間 ... 266
15 - 1	HALTモード時の動作状態 ... 276
15 - 2	HALTモードの解除と解除後の動作 ... 277
15 - 3	マスカブル割り込み要求によるHALTモードの解除 ... 283
15 - 4	STOPモード時の動作状態 ... 284
15 - 5	STOPモードの解除と解除後の動作 ... 285
15 - 6	IDLEモード時の動作状態 ... 290
15 - 7	IDLEモードの解除と解除後の動作 ... 291
16 - 1	各ハードウェアのリセット中, リセット後の状態 ... 300
17 - 1	通信方式一覧 ... 301
17 - 2	フラッシュ・メモリ・プログラミングの主な機能 ... 302
18 - 1	8ビット・アドレッシング別命令一覧表 ... 334
18 - 2	16ビット・アドレッシング別命令一覧表 ... 335
18 - 3	24ビット・アドレッシング別命令一覧表 ... 336
18 - 4	ビット操作命令アドレッシング別命令一覧表 ... 336
18 - 5	コール・リターン命令/分岐命令アドレッシング別命令一覧表 ... 337

〔メ モ〕

# 第 1 章 概 説

μPD784976サブシリーズは、78K/ シリーズのひとつで、100ピン版のASSP用途向け製品です。78K/ シリーズは、16ビット・シングルチップ・マイクロコンピュータで、1 Mバイトのメモリ空間に対するアクセス機能などを持った高性能CPUを備えた製品です。

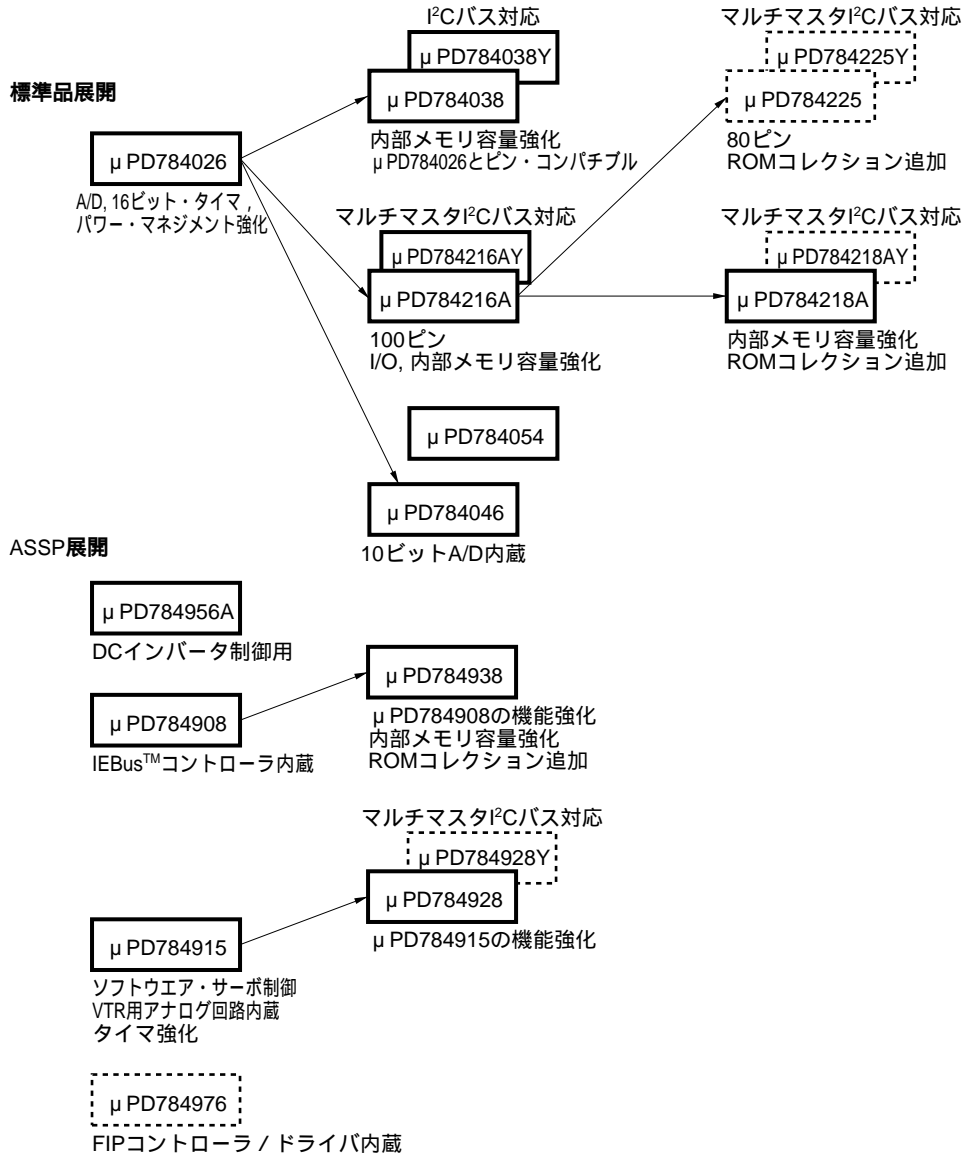
μPD784975は、96 KバイトのマスクROMと3584バイトのRAMを内蔵しています。また、FIPコントローラ・ドライバ、高機能タイマ/イベント・カウンタ、2チャンネル独立のシリアル・インタフェースを内蔵しています。

μPD78F4976は、128 Kバイトのフラッシュ・メモリと5120バイトのRAMを内蔵しています。

78K/IVシリーズ製品展開図

□ : 量産中

□ (点線) : 開発中



## 1.1 特 徴

大容量ROM, RAM

項目 品名	プログラム・メモリ		データ・メモリ		
	マスクROM	フラッシュ・メモリ	周辺RAM	高速RAM	FIP <sup>®</sup> 表示用RAM
μPD784975	96 Kバイト	-	3072バイト	512バイト	96バイト
μPD78F4976	-	128 Kバイト注	4608バイト		

注 メモリ・サイズ切り替えレジスタ (IMS) により, 96 Kバイトの選択可能。

最小命令実行時間

- ・ 160 ns/320 ns/640 ns/1280 ns/2560 ns (  $f_{xx} = 12.5$  MHz動作時 )

I/Oポート : 72本

FIPコントローラ / ドライバ : 表示出力合計48本 (ユニバーサル・グリッド対応)

- ・ 表示電流10 mA : 16本
- ・ 表示電流3 mA : 32本

8ビット分解能A/Dコンバータ : 12チャンネル

- ・ 電源電圧 (  $AV_{DD} = 4.5 \sim 5.5$  V )

シリアル・インタフェース : 2チャンネル

- ・ 3線式シリアルI/Oモード : 2チャンネル

タイマ : 4チャンネル

- ・ 16ビット・タイマ / イベント・カウンタ : 1チャンネル
- ・ 8ビットPWMタイマ : 2チャンネル
- ・ ウォッチドック・タイマ : 1チャンネル

ベクタ割り込み要因 : 15

電源電圧 :  $V_{DD} = 4.5 \sim 5.5$  V

## 1.2 応用分野

一体型ミニコンポ, セパレート型ミニコンポ, チューナ, カセット・デッキ, CDプレーヤ, オーディオ・アンプなど

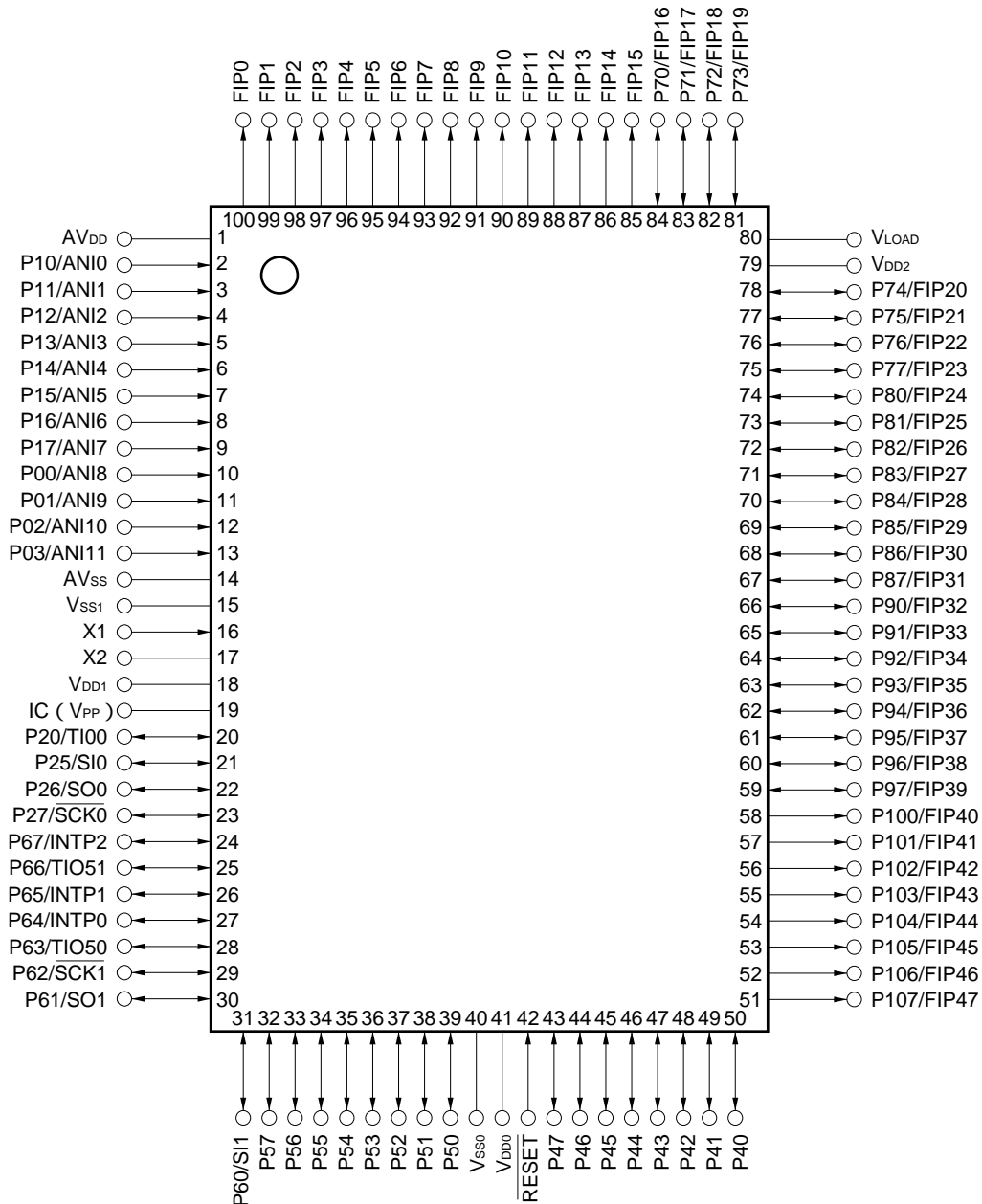
## 1.3 オーダ情報

オーダ名称	パッケージ	内部ROM
μPD784975GF-×××-3BA	100ピン・プラスチックQFP (14×20 mm)	マスクROM
μPD78F4976GF-3BA	"	フラッシュ・メモリ

**備考** ×××はROMコード番号です。

## 1.4 端子接続図 (Top View)

- ・ 100ピン・プラスチックQFP (14 × 20 mm)
- μ PD784975GF- x × x -3BA, 78F4976GF-3BA



- 注意 1.** 通常動作モード時は、IC (Internally Connected) 端子はV<sub>SS1</sub>に直接接続してください。
2. AV<sub>DD</sub>端子はV<sub>DD1</sub>に接続してください。
  3. AV<sub>SS</sub>端子はV<sub>SS1</sub>に接続してください。

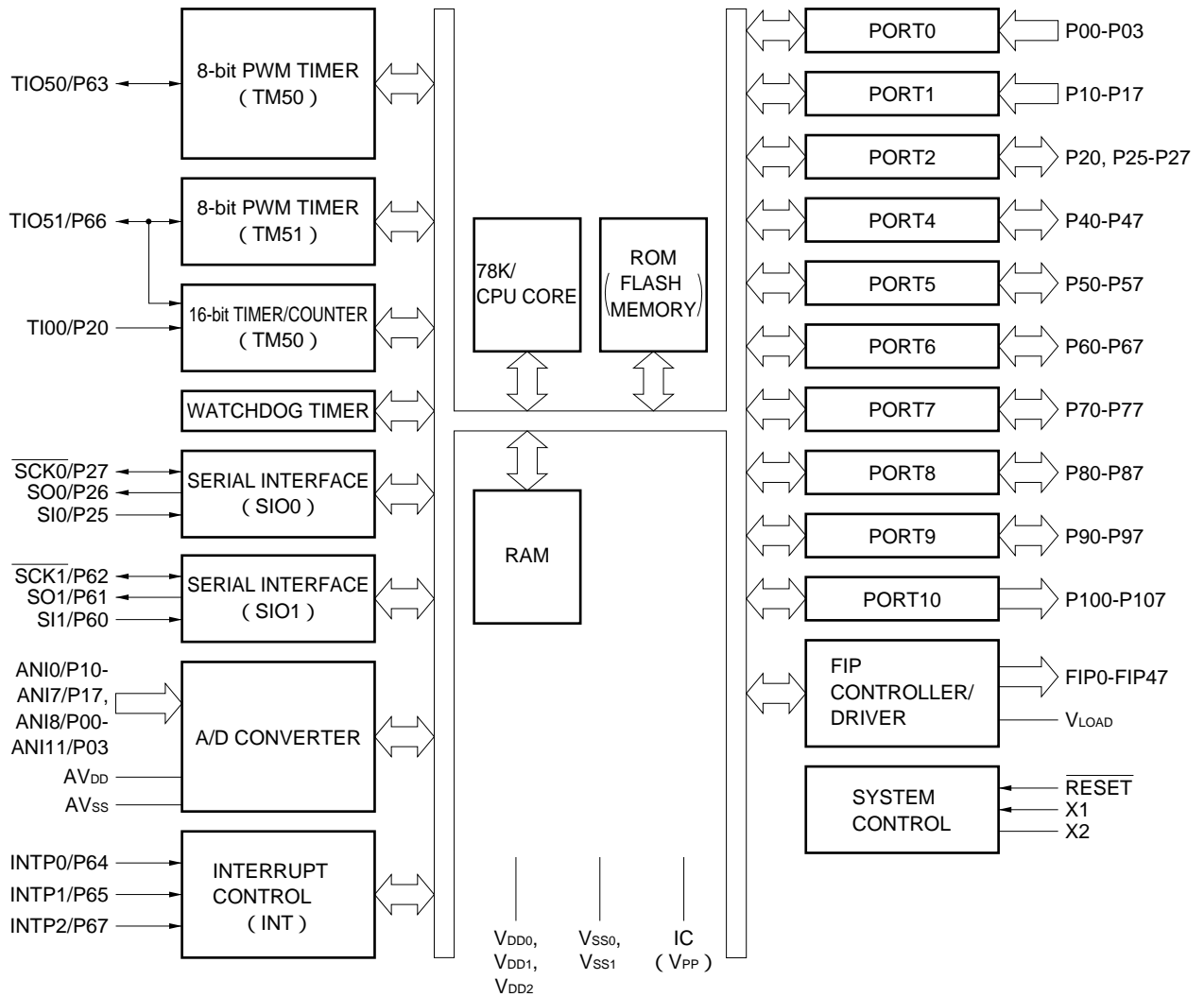
**備考 1.** マイコン内部から発生するノイズを低減する必要がある応用分野で使用する場合、V<sub>DD0</sub>とV<sub>DD1</sub>に個別の電源を供給し、V<sub>SS0</sub>とV<sub>SS1</sub>を別々のグランド・ラインに接続するなどのノイズ対策を行うことを推奨します。

2. ( )内はμ PD78F4976のとき。

ANI0-ANI11	: Analog Input	P90-P97	: Port9
AV <sub>DD</sub>	: Analog Power Supply	P100-P107	: Port10
AV <sub>SS</sub>	: Analog Ground	$\overline{\text{RESET}}$	: Reset
FIP0-FIP47	: Fluorescent Indicator Panel	$\overline{\text{SCK0}}, \overline{\text{SCK1}}$	: Serial Clock
IC	: Internally Connected	SI0, SI1	: Serial Input
INTP0-INTP2	: External Interrupt Input	SO0, SO1	: Serial Output
P00-P03	: Port0	TI00	: Timer Input
P10-P17	: Port1	TIO50, TIO51	: Timer Input/Output
P20, P25-P27	: Port2	V <sub>DD0</sub> -V <sub>DD2</sub>	: Power Supply
P40-P47	: Port4	V <sub>LOAD</sub>	: Negative Power Supply
P50-P57	: Port5	V <sub>PP</sub>	: Programming Power Supply
P60-P67	: Port6	V <sub>SS0</sub> , V <sub>SS1</sub>	: Ground
P70-P77	: Port7	X1, X2	: Crystal
P80-P87	: Port8		



## 1.5 ブロック図



備考1 . 内部ROM容量は製品によって異なります。

2 . FLASH MEMORYおよびV<sub>PP</sub>端子は , μ PD78F4976のときのみ。

## 1.6 機能概要

品 名		μ PD784975	μ PD78F4976
内部メモリ	ROM	マスクROM	フラッシュ・メモリ
		96 Kバイト	128 Kバイト <sup>注</sup>
	周辺RAM	3072バイト	4608バイト
	高速RAM	512バイト	
	FIP表示用RAM	96バイト	
汎用レジスタ		8 ビット × 8 × 4 バンク	
最小命令実行時間		160 ns/320 ns/640 ns/1280 ns/2560 ns ( f <sub>xx</sub> = 12.5 MHz動作時 )	
命令セット		<ul style="list-style-type: none"> <li>・ 16ビット演算</li> <li>・ 乗除算 ( 8 ビット × 8 ビット, 16ビット ÷ 8 ビット )</li> <li>・ ビット操作 ( セット, リセット, テスト, プール演算 )</li> <li>・ BCD補正 など</li> </ul>	
I/Oポート ( FIP兼用端子を含む )		<ul style="list-style-type: none"> <li>・ 合計 : 72本</li> <li>・ CMOS入力 : 12本</li> <li>・ CMOS入出力 : 20本</li> <li>・ N-chオープン・ドレイン入出力 : 8本</li> <li>・ P-chオープン・ドレイン入出力 : 24本</li> <li>・ P-chオープン・ドレイン出力 : 8本</li> </ul>	
FIPコントローラ/ドライバ		<ul style="list-style-type: none"> <li>・ 表示出力合計 : 48本</li> <li>・ 表示電流10 mA : 16本</li> <li>・ 表示電流3 mA : 32本</li> </ul>	
A/Dコンバータ		<ul style="list-style-type: none"> <li>・ 8 ビット分解能 × 12チャンネル</li> <li>・ 電源電圧 : AV<sub>DD</sub> = 4.5 ~ 5.5 V</li> </ul>	
シリアル・インタフェース		3 線式シリアルI/Oモード : 2チャンネル	
タイマ		<ul style="list-style-type: none"> <li>・ 16ビット・タイマ/イベント・カウンタ : 1チャンネル</li> <li>・ 8 ビットPWMタイマ : 2チャンネル</li> <li>・ ウォッチドッグ・タイマ : 1チャンネル</li> </ul>	
タイマ出力		2本 ( 8 ビットPWM出力可能 )	
ベクタ割り込み 要因	マスカブル	内部 : 7, 外部 : 3, 内外部兼用 : 2	
	ノンマスカブル	内部 : 1	
	ソフトウェア	BRK命令, BRKCS命令, オペランド・エラー	
電源電圧		V <sub>DD</sub> = 4.5 ~ 5.5 V	
パッケージ		100ピン・プラスチックQFP ( 14 × 20 mm )	

注 メモリ・サイズ切り替えレジスタ ( IMS ) により, 96 Kバイトの選択可能。

タイマの概要（詳細は、第7章 16ビット・タイマ/イベント・カウンタ，第8章 8ビットPWMタイマを参照）を次に示します。

項 目		名 称	16ビット・タイマ/ イベント・カウンタ	8ビットPWMタイマ (TM50)	8ビットPWMタイマ (TM51)
カウント幅	8ビット		-		
	16ビット				
動作モード	インターバル・タイマ		1ch	1ch	1ch
	外部イベント・カウンタ				
機能	タイマ出力		-		
	PWM出力		-		
	方形波出力		-		
	パルス幅測定		2入力	-	-
	割り込み要求数		2	1	1

シリアル概要（詳細は、第11章 シリアル・インタフェースを参照）を以下に示します。

機 能	SI0	SI1
3線式シリアル/Oモード	(MSB固定)	(MSB固定)

## 1.7 マスク・オプションについて

マスクROM製品（ $\mu$ PD784975）には、マスク・オプションがあります。オーダの際にマスク・オプションを指定することにより、表1 - 1に示すプルアップ抵抗やプルダウン抵抗を内蔵することができます。プルアップ抵抗やプルダウン抵抗が必要なとき、これらのマスク・オプションを利用すると、部品点数の削減と実装面積の縮小を図ることができます。

$\mu$ PD784976サブシリーズで用意されているマスク・オプションを表1 - 1に示します。

表1 - 1 マスクROM製品のマスク・オプション一覧

端子名	マスク・オプション
P50-P57	1ビット単位でプルアップ抵抗を内蔵できます。
P70/FIP16-P77/FIP23, P80/FIP24-P87/FIP31, P90/FIP32-P97/FIP39, P100/FIP40-P107/FIP47	1ビット単位でプルダウン抵抗を内蔵できます。

## 第2章 端子機能

### 2.1 端子機能一覧

#### (1) ポート端子 (1/2)

端子名称	入出力	機能	リセット時	兼用端子
P00-P03	入力	ポート 0。 4 ビットの入力専用ポート。	-	ANI8-ANI11
P10-P17	入力	ポート 1。 8 ビットの入力専用ポート。	-	ANI0-ANI7
P20	入出力	ポート 2。 4 ビットの入出力ポート。 1 ビット単位で入力 / 出力の指定可能。 ソフトウェアにより、1 ビット単位または 8 ビット単位で内蔵プルアップ抵抗を使用可能。	入力	TI00
P25				SI0
P26				SO0
P27				SCK0
P40-P47	入出力	ポート 4。 8 ビットの入出力ポート。 1 ビット単位で入力 / 出力の指定可能。 ソフトウェアにより、ポート単位で内蔵プルアップ抵抗を使用可能。 LEDを直接駆動可能。	入力	-
P50-P57	入出力	ポート 5。 8 ビットN-chオープン・ドレイン中耐圧入出力ポート。 1 ビット単位で入力 / 出力の指定可能。 マスクROM製品のみマスク・オプションにより、1 ビット単位で内蔵プルアップ抵抗を使用可能。μPD78F4976は、プルアップ抵抗を内蔵していません。 LEDを直接駆動可能。	入力	-
P60	入出力	ポート 6。 8 ビットの入出力ポート。 1 ビット単位で入力 / 出力の指定可能。 ソフトウェアにより、ポート単位で内蔵プルアップ抵抗を使用可能。	入力	SI1
P61				SO1
P62				SCK1
P63				TIO50
P64				INTP0
P65				INTP1
P66				TIO51
P67				INTP2
P70-P77	入出力	ポート 7。 8 ビットP-chオープン・ドレイン高耐圧入出力ポート。 1 ビット単位で入力 / 出力の指定可能。 マスクROM製品のみマスク・オプションにより、1 ビット単位で内蔵プルダウン抵抗を使用可能。μPD78F4976は、プルダウン抵抗を内蔵していません。	入力	FIP16-FIP23

## (1) ポート端子 (2/2)

端子名称	入出力	機能	リセット時	兼用端子
P80-P87	入出力	<p>ポート8。</p> <p>P-chオープン・ドレイン8ビット高耐圧入出力ポート。</p> <p>1ビット単位で入力/出力の指定可能。</p> <p>マスクROM製品のみマスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵が可能。μPD78F4976は、プルダウン抵抗を内蔵していません。</p>	入力	FIP24-FIP31
P90-P97	入出力	<p>ポート9。</p> <p>P-chオープン・ドレイン8ビット高耐圧入出力ポート。</p> <p>1ビット単位で入力/出力の指定可能。</p> <p>マスクROM製品のみマスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵が可能。μPD78F4976は、プルダウン抵抗を内蔵していません。</p>	入力	FIP32-FIP39
P100-P107	出力	<p>ポート10。</p> <p>P-chオープン・ドレイン8ビット高耐圧出力ポート。</p> <p>マスクROM製品のみマスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵が可能。μPD78F4976は、プルダウン抵抗を内蔵していません。</p>	-	FIP40-FIP47

(2) ポート以外の端子

端子名称	入出力	機能	リセット時	兼用端子
INTP0	入力	有効エッジ（立ち上がりエッジ，立ち下がりエッジ，立ち上がりおよび立ち下がりの両エッジ）指定可能な外部割り込み要求入力	入力	P64
INTP1				P65
INTP2				P67
SI0	入力	シリアル・データ入力（3線式シリアルI/O0）		P25
SO0	出力	シリアル・データ出力（3線式シリアルI/O0）		P26
$\overline{\text{SCK0}}$	入出力	シリアル・クロック入力/出力（3線式シリアルI/O0）		P27
SI1	入力	シリアル・データ入力（3線式シリアルI/O1）		P60
SO1	出力	シリアル・データ出力（3線式シリアルI/O1）		P61
$\overline{\text{SCK1}}$	入出力	シリアル・クロック入力/出力（3線式シリアルI/O1）		P62
TI00	入力	16ビット・タイマ/カウンタ0（TM0）への外部カウント・クロック入力，または16ビット・キャプチャ/コンペア・レジスタ（CR00/CR01）へのキャプチャ・トリガ信号入力		P20
TI050	入出力	8ビットPWMタイマ（TM50）への外部カウント・クロック入力/タイマ出力		P63
TI051		8ビットPWMタイマ（TM51）への外部カウント・クロック入力/タイマ出力，16ビット・キャプチャ/コンペア・レジスタ（CR00）へのキャプチャ・トリガ信号入力		P66
ANI0-ANI7	入力	A/Dコンバータ用のアナログ電圧入力		P10-P17
ANI8-ANI11				P00-P03
AV <sub>DD</sub>	-	A/Dコンバータ用のアナログ電源。V <sub>DD1</sub> と同電位にしてください。	-	-
AV <sub>SS</sub>		A/Dコンバータ用のグラウンド電位。V <sub>SS1</sub> と同電位にしてください。		
FIP0-FIP15	出力	FIPコントローラ/ドライバの高耐圧大電流出力	出力	
FIP16-FIP23			入力	P70-P77
FIP24-FIP31				P80-P87
FIP32-FIP39				P90-P97
FIP40-FIP47			-	P100-P107
V <sub>LORD</sub>	-	FIPコントローラ/ドライバのブルダウン抵抗接続		-
$\overline{\text{RESET}}$	入力	システム・リセット入力。		
X1		メイン・システム・クロック発振用クリスタル接続		
X2		-		
V <sub>DD0</sub>		ポート部の正電源		
V <sub>DD1</sub>		正電源（ポート部，アナログ，FIPコントローラ/ドライバ部を除く）		
V <sub>DD2</sub>		FIPコントローラ/ドライバ部の正電源		
V <sub>SS0</sub>		ポート部のグラウンド電位		
V <sub>SS1</sub>		グラウンド電位（ポート部，アナログ部を除く）		
V <sub>PP</sub> <sup>注</sup>		プログラム書き込み/ベリファイ時の高電圧印加。通常動作モード時は，V <sub>SS1</sub> に直接接続してください。		
IC		内部接続されています。V <sub>SS1</sub> に直接接続してください。		

注 V<sub>PP</sub>はμPD78F4976のみ。

## 2.2 端子機能の説明

### 2.2.1 P00-P03 (Port0)

4ビットの入力専用ポートです。入力ポートのほかに、A/Dコンバータのアナログ入力機能があります。

1ビット単位で次のような動作モードを指定できます。

#### (1) ポート・モード

4ビットの入力専用ポートとして機能します。

#### (2) コントロール・モード

A/Dコンバータのアナログ入力端子 (ANI8-ANI11) として機能します。

### 2.2.2 P10-P17 (Port1)

8ビットの入力専用ポートです。入力ポートのほかにA/Dコンバータのアナログ入力機能があります。

1ビット単位で次のような動作モードを指定できます。

#### (1) ポート・モード

8ビットの入力専用ポートとして機能します。

#### (2) コントロール・モード

A/Dコンバータのアナログ入力端子 (ANI0-ANI7) として機能します。

### 2.2.3 P20, P25-P27 (Port2)

4ビットの入出力ポートです。入出力ポートのほかにシリアル・インタフェース0のデータ入出力、クロック入出力、タイマの入力機能があります。

1ビット単位で次のような動作モードを指定できます。

#### (1) ポート・モード

4ビットの入出力ポートとして機能します。ポート2モード・レジスタ (PM2) により、1ビット単位で入力ポートまたは出力ポートに指定できます。プルアップ抵抗オプション・レジスタ2 (PU2) により、1ビット単位で内蔵プルアップ抵抗を使用できます。

#### (2) コントロール・モード

シリアル・インタフェース0のデータ入出力、クロック入出力として機能します。

##### (a) SI0, SO0

シリアル・インタフェース0のシリアル・データの入出力端子です。



**(b)  $\overline{\text{SCK0}}$** 

シリアル・インタフェース0のシリアル・クロックの入出力端子です。

**(c) TI00**

16ビット・タイマ・カウンタ0 (TM0) のタイマ入力端子です。

**2.2.4 P40-P47 (Port4)**

8ビットの入出力ポートです。ポート4モード・レジスタ (PM4) により、1ビット単位で入力ポートまたは出力ポートに指定できます。入力ポートとして使用する時のみ、プルアップ抵抗オプション・レジスタ (PUO) のビット4 (PUO4) により、8ビット単位で内蔵プルアップ抵抗を使用できます。

LEDを直接駆動可能です。

**2.2.5 P50-P57 (Port5)**

8ビットの入出力ポートです。ポート5モード・レジスタ (PM5) により、1ビット単位で入力ポートまたは出力ポートに指定できます。

P50-P57はN-chオープン・ドレインになっています。マスクROM製品は、マスク・オプションにより、1ビット単位でプルアップ抵抗を内蔵できます。μPD78F4976は、プルアップ抵抗を内蔵していません。

LEDを直接駆動可能です。

**2.2.6 P60-P67 (Port6)**

8ビットの入出力ポートです。入出力ポートのほかにシリアル・インタフェース1のデータ入出力、クロック入出力、タイマ入出力、外部割り込み要求入力機能などがあります。

1ビット単位で次のような動作モードを指定できます。

**(1) ポート・モード**

8ビットの入出力ポートとして機能します。ポート6モード・レジスタ (PM6) により、1ビット単位で入力/出力の指定ができます。

入力ポートとして使用する時のみ、プルアップ抵抗オプション・レジスタ (PUO) のビット6 (PUO6) により、8ビット単位で内蔵プルアップ抵抗を使用できます。

**(2) コントロール・モード**

シリアル・インタフェース1のデータ入出力、クロック入出力、タイマの入出力、タイマのキャプチャ・トリガ信号入力、外部割り込み要求入力機能などがあります。

**(a) SI1, SO1**

シリアル・インタフェース1のシリアル・データの入出力端子です。

**(b)  $\overline{\text{SCK1}}$** 

シリアル・インタフェース1のシリアル・クロックの入出力端子です。

**(c) TIO50, TIO51**

TIO50: 8ビットPWMタイマへの外部カウント・クロック入力端子および出力端子です。

TIO51: 8ビットPWMタイマへの外部カウント・クロック入力端子およびタイマ出力端子, 16ビット・キャプチャ/コンペア・レジスタ00 (CR00) のキャプチャ・トリガ信号入力端子です。

**(d) INTP0, INTP1, INTP2**

有効エッジ (立ち上がりエッジ, 立ち下がりエッジ, 立ち上がりおよび立ち下がりの両エッジ) 指定可能な外部割り込み要求入力端子として機能します。

**2.2.7 P70-P77 (Port7)**

8ビットの入出力ポートです。入出力ポートのほかにFIPコントローラ/ドライバの出力機能があります。

1ビット単位で次のような動作モードを指定できます。

**(1) ポート・モード**

8ビットの入出力ポートとして機能します。

P70-P77はP-chオープン・ドレインになっています。マスクROM製品は, マスク・オプションにより, 1ビット単位でプルダウン抵抗を内蔵できます。μPD78F4976は, プルダウン抵抗を内蔵していません。

**(2) コントロール・モード**

FIPコントローラ/ドライバの出力端子 (FIP16-FIP23) として機能します。

**2.2.8 P80-P87 (Port8)**

8ビットの入出力ポートです。入出力ポートのほかにFIPコントローラ/ドライバの出力機能があります。

1ビット単位で次のような動作モードを指定できます。

**(1) ポート・モード**

8ビットの入出力ポートとして機能します。

P80-P87はP-chオープン・ドレインになっています。マスクROM製品は, マスク・オプションにより, 1ビット単位でプルダウン抵抗を内蔵できます。μPD78F4976は, プルダウン抵抗を内蔵していません。

**(2) コントロール・モード**

FIPコントローラ/ドライバの出力端子 (FIP24-FIP31) として機能します。

### 2.2.9 P90-P97 (Port9)

8ビットの入出力ポートです。入出力ポートのほかにFIPコントローラ/ドライバの出力機能があります。

1ビット単位で次のような動作モードを指定できます。

#### (1) ポート・モード

8ビットの入出力ポートとして機能します。

P90-P97はP-chオープン・ドレインになっています。マスクROM製品は、マスク・オプションにより、1ビット単位でプルダウン抵抗を内蔵できます。μPD78F4976は、プルダウン抵抗を内蔵していません。

#### (2) コントロール・モード

FIPコントローラ/ドライバの出力端子 (FIP32-FIP39) として機能します。

### 2.2.10 P100-P107 (Port10)

8ビットの出力専用ポートです。出力ポートのほかにFIPコントローラ/ドライバの出力機能があります。

1ビット単位で次のような動作モードを指定できます。

#### (1) ポート・モード

8ビットの出力専用ポートとして機能します。

P100-P107はP-chオープン・ドレインになっています。マスクROM製品は、マスク・オプションにより、1ビット単位でプルダウン抵抗を内蔵できます。μPD78F4976は、プルダウン抵抗を内蔵していません。

#### (2) コントロール・モード

FIPコントローラ/ドライバの出力端子 (FIP40-FIP47) として機能します。

### 2.2.11 FIP0-FIP15

FIPコントローラ/ドライバの出力端子です。

### 2.2.12 VLOAD

FIPコントローラ/ドライバのプルダウン抵抗接続端子です。

### 2.2.13 AVDD

A/Dコンバータのアナログ電源端子です。

A/Dコンバータを使用しないときでも、常にV<sub>DD1</sub>端子と同電位で使用してください。

### 2.2.14 AVss

A/Dコンバータのグランド電位端子です。

A/Dコンバータを使用しないときでも、常にV<sub>SS1</sub>端子と同電位で使用してください。

### 2.2.15 $\overline{\text{RESET}}$

ロウ・レベル・アクティブのシステム・リセット入力端子です。

### 2.2.16 X1, X2

メイン・システム・クロック発振用クリスタル振動子接続端子です。

外部クロックを供給するときは、X1に入力し、X2にその反転信号を入力してください。

### 2.2.17 VDD0-VDD2

V<sub>DD0</sub>は、ポート部の正電源供給端子です。

V<sub>DD1</sub>は、ポート、アナログ、FIPコントローラ/ドライバ部以外の正電源供給端子です。

V<sub>DD2</sub>は、FIPコントローラ/ドライバ部の正電源供給端子です。

### 2.2.18 VSS0, VSS1

V<sub>SS0</sub>は、ポート部のグランド電位端子です。

V<sub>SS1</sub>は、ポート、アナログ部以外のグランド電位端子です。

### 2.2.19 VPP ( $\mu$ PD78F4976のみ )

フラッシュ・メモリ・プログラミング・モード設定およびプログラム書き込み/ベリファイ時の高電圧印加端子です。

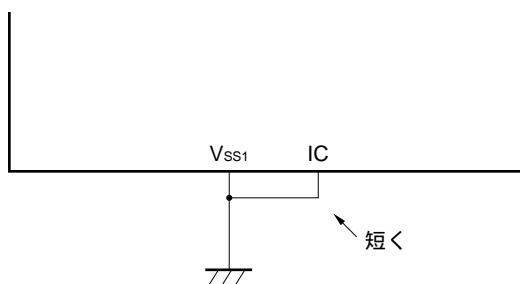
通常動作モード時は、V<sub>SS1</sub>に直接接続してください。

### 2.2.20 IC ( マスクROM製品のみ )

IC ( Internally Connected ) 端子は、当社出荷時に  $\mu$ PD784975を検査するためのテスト・モードに設定するための端子です。通常動作モード時には、IC端子をV<sub>SS1</sub>端子に直接接続し、その配線長を極力短くしてください。

IC端子とV<sub>SS1</sub>端子間の配線の引き回しが長い場合や、IC端子に外来ノイズが加わった場合などで、IC端子とV<sub>SS1</sub>端子間に電位差が生じたときには、お客様のプログラムが正常に動作しないことがあります。

IC端子をV<sub>SS1</sub>端子に直接接続してください



## 2.3 端子の入出力回路と未使用端子の処理

各端子の入出力回路タイプと、未使用端子の処理を表2-1に示します。

また、各タイプの入出力回路の構成は、図2-1を参照してください。

表2-1 各端子の入出力回路タイプ

端子名	入出力回路タイプ	入出力	未使用時の推奨接続方法
P00/ANI8-P03/ANI11	9	入力	V <sub>DD0</sub> またはV <sub>SS0</sub> に接続してください。
P10/ANI0-P17/ANI7			
P20/TI00	8-C	入出力	入力時：個別に抵抗を介して、V <sub>SS0</sub> に接続してください。 出力時：オープンにしてください。
P25/SI0			
P26/SO0	5-H		
P27/ $\overline{\text{SCK0}}$	8-C		
P40-P47	5-H		
マスクROM製品			
P50-P57	13-J	入出力	入力時：個別に抵抗を介して、V <sub>DD0</sub> に接続してください。 出力時：オープンにしてください。
P70/FIP16-P77/FIP23	15-F	入出力	入力時：個別に抵抗を介して、V <sub>SS0</sub> に接続してください。 出力時：オープンにしてください。
P80/FIP24-P87/FIP31			
P90/FIP32-P97/FIP39			
P100/FIP40-P107/FIP47	14-F	出力	オープンにしてください。
IC	-	-	V <sub>SS1</sub> に直接接続してください。
μ PD78F4976			
P50-P57	13-K	入出力	入力時：個別に抵抗を介して、V <sub>DD0</sub> に接続してください。 出力時：オープンにしてください。
P70/FIP16-P77/FIP23	15-E	入出力	入力時：個別に抵抗を介して、V <sub>SS0</sub> に接続してください。 出力時：オープンにしてください。
P80/FIP24-P87/FIP31			
P90/FIP32-P97/FIP39			
P100/FIP40-P107/FIP47	14-E	出力	オープンにしてください。
V <sub>PP</sub>	-	-	V <sub>SS1</sub> に直接接続してください。
FIP0-FIP15	14-C	出力	オープンにしてください。
RESET	2	入力	-
AV <sub>DD</sub>	-	-	V <sub>DD1</sub> に接続してください。
AV <sub>SS</sub>			V <sub>SS1</sub> に接続してください。
V <sub>LOAD</sub>			

図2-1 端子の入出力回路一覧(1/2)

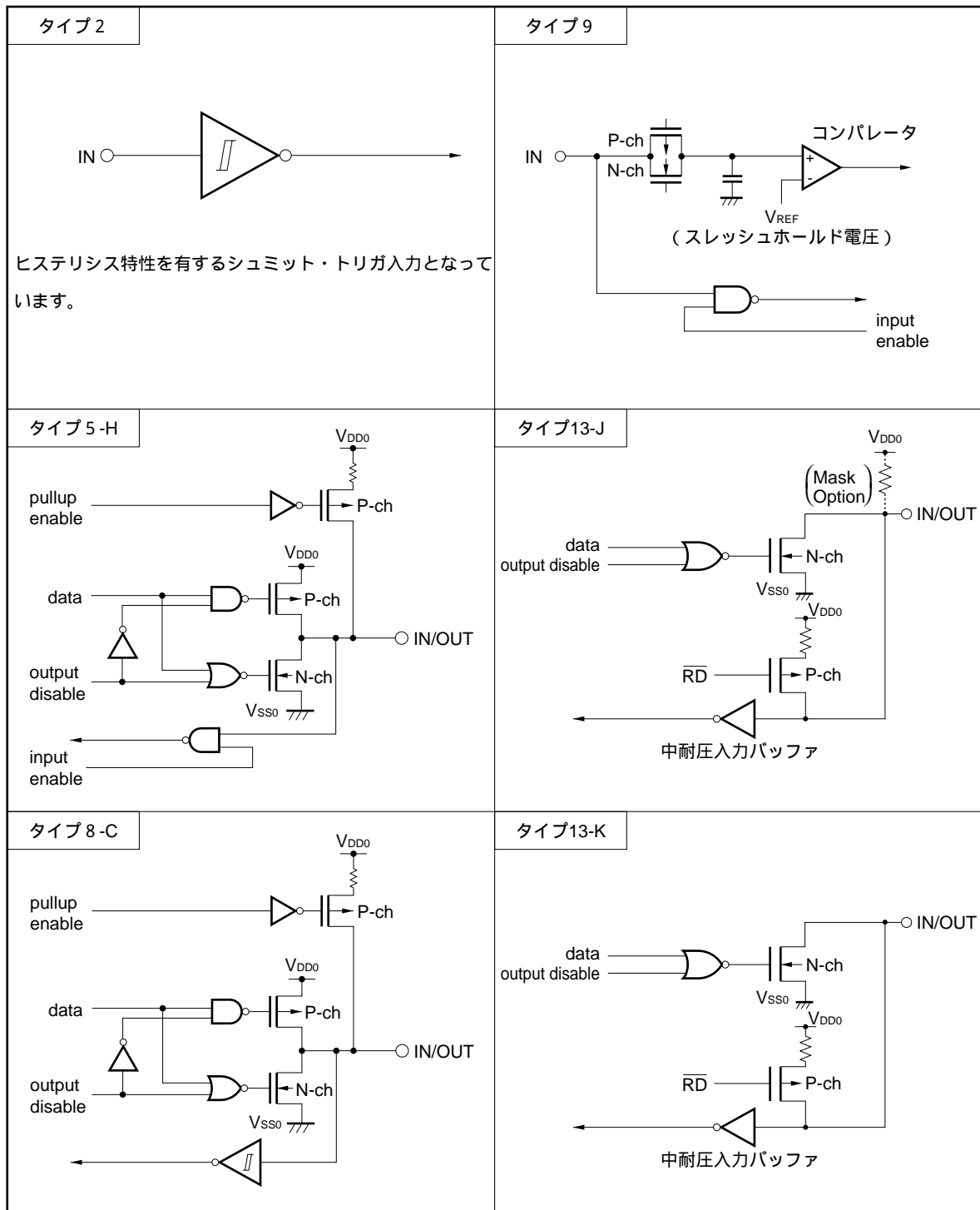
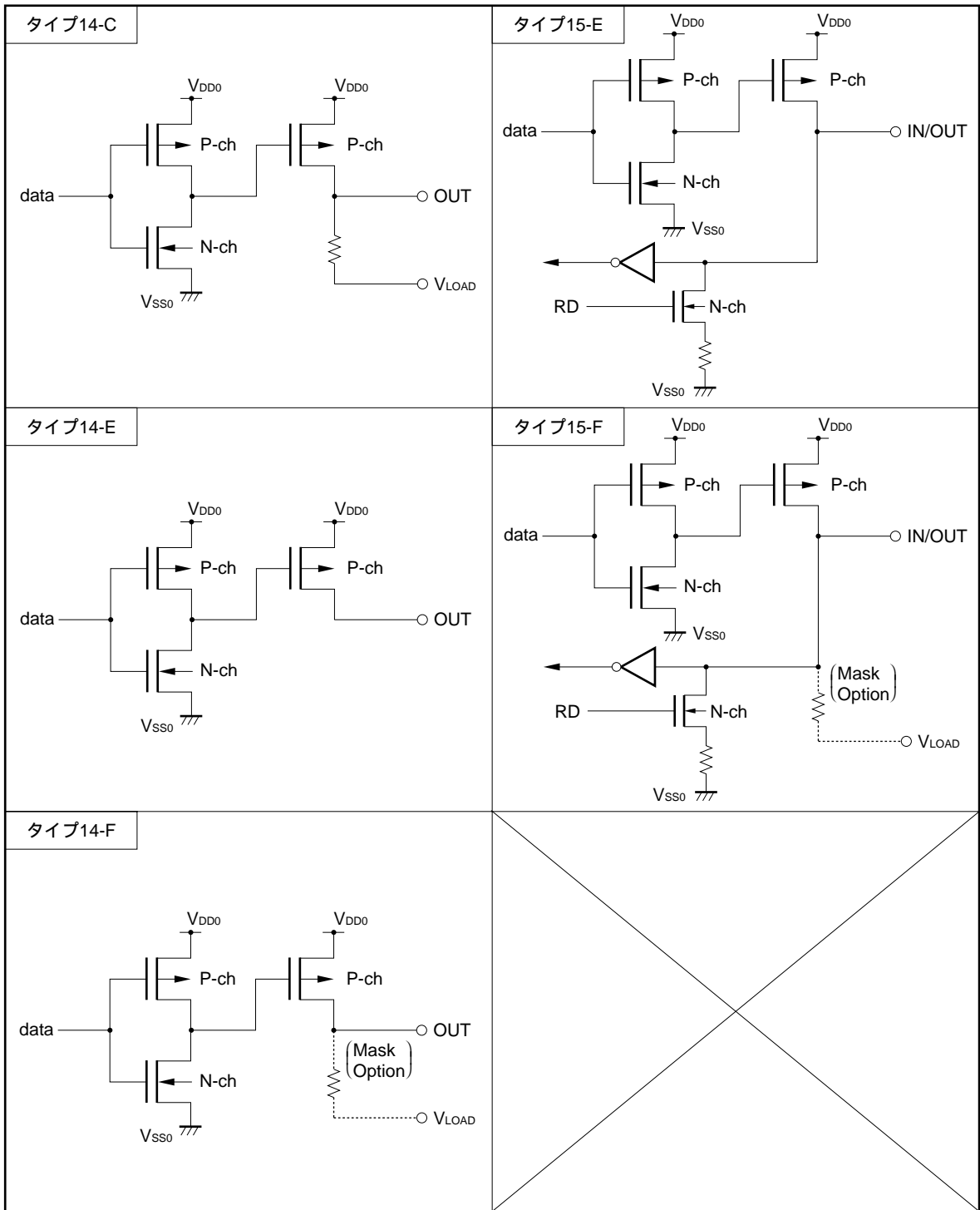


図2-1 端子の入出力回路一覧(2/2)



(メモ)



## 第3章 CPUアーキテクチャ

### 3.1 メモリ空間

μPD784975は、1Mバイトの空間をアクセスできます。LOCATION命令によって、内部データ領域（特殊機能レジスタおよび内部RAM）のマッピングが異なります。LOCATION命令は、リセット解除後に必ず実行する必要があり、2回以上使用することはできません。

リセット解除後のプログラムは、次のようになっている必要があります。

```
RSTVCT  CSEG  AT 0
        DW    RSTSTRT
        {
INITSEG  CSEG  BASE
RSTSTRT : LOCATION 0H ; or LOCATION 0FH
        MOVG  SP, #STKBGN
        MOV   MM, #80H
```

リセット解除後は、メモリ拡張モード・レジスタ（MM）に必ず80Hを設定してください。

MMIは、内部ROMからの命令フェッチ速度を選択するレジスタです。

MMIは、8ビット・メモリ操作命令で読み出し/書き込みを行います。

図3 - 1にMMのフォーマットを示します。

$\overline{\text{RESET}}$ 入力により、20Hになります。

図3 - 1 メモリ拡張モード・レジスタ（MM）のフォーマット

アドレス：0FFC4H リセット時：20H R/W

略号	7	6	5	4	3	2	1	0
MM	IFCH	0	AW	0	0	0	0	0

IFCH	AW	内部ROMからの命令フェッチ速度の選択
0	1	低速モード（6サイクル=1バイトの速度でフェッチを行う）
1	0	通常モード（2サイクル=2バイトの速度でフェッチを行う）
上記以外		設定禁止

**注意** リセット後は、必ず80Hを設定してください。

(1) LOCATION 0命令を実行した場合

・内部メモリ

内部データ領域と内部ROM領域は、次のようになります。

品名	内部データ領域	内部ROM領域
μ PD784975	0F100H-0FFFFH	00000H-0E9FFH
	0EA00H-0EA5FH	10000H-17FFFH
μ PD78F4976	0EB00H-0FFFFH	00000H-0E9FFH
	0EA00H-0EA5FH	10000H-1FFFFH

**備考** 内蔵しているROMのうち、内部データ領域と重なる次の領域は、LOCATION 0命令実行時には使用できません。

品名	使用不可領域
μ PD784975	0EA00H-0FFFFH (5632バイト)
μ PD78F4976	

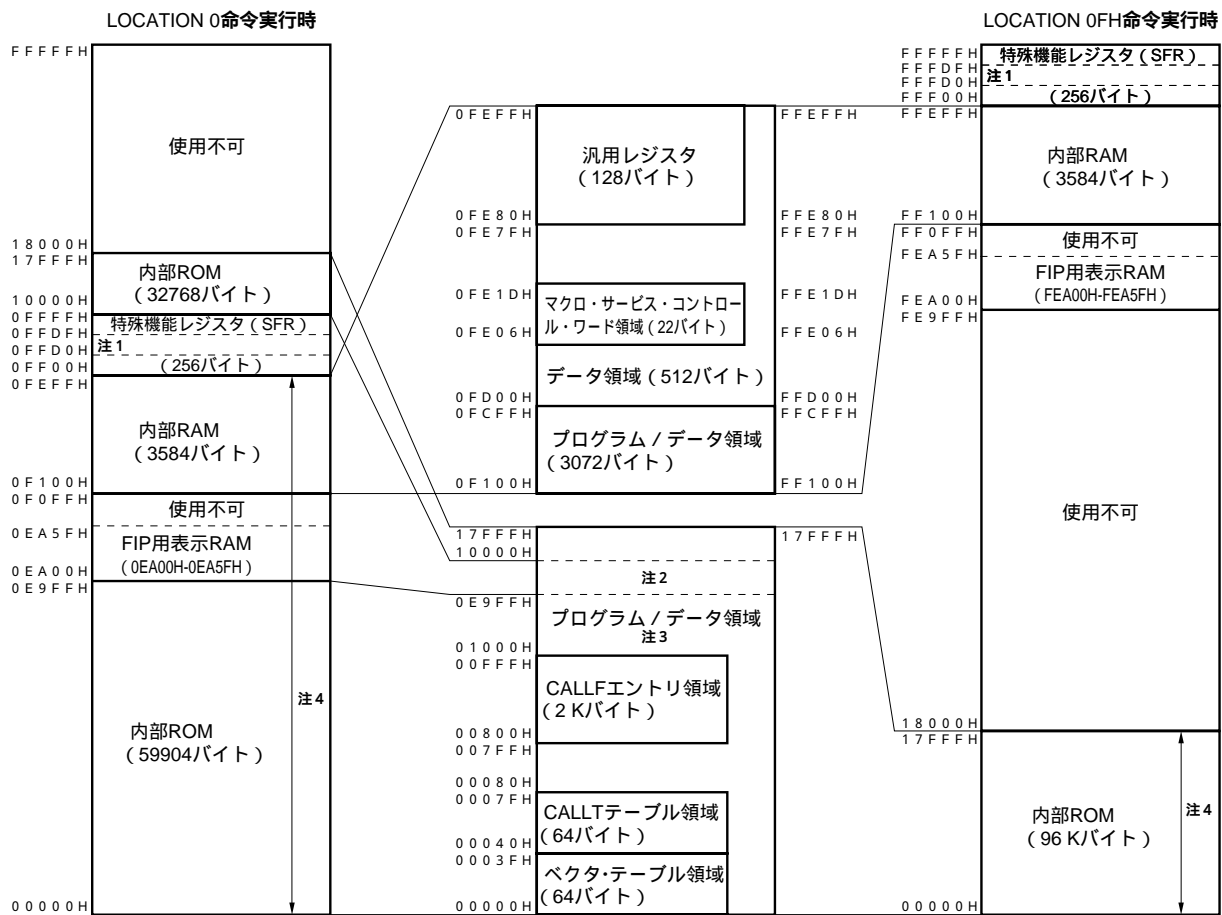
(2) LOCATION 0FH命令を実行した場合

・内部メモリ

内部データ領域と内部ROM領域は、次のようになります。

品名	内部データ領域	内部ROM領域
μ PD784975	FF100H-FFFFFFH	00000H-17FFFH
	FEA00H-FEA5FH	
μ PD78F4976	FEB00H-FFFFFFH	00000H-1FFFFH
	FEA00H-FEA5FH	

図3 - 2 μPD784975のメモリ・マップ



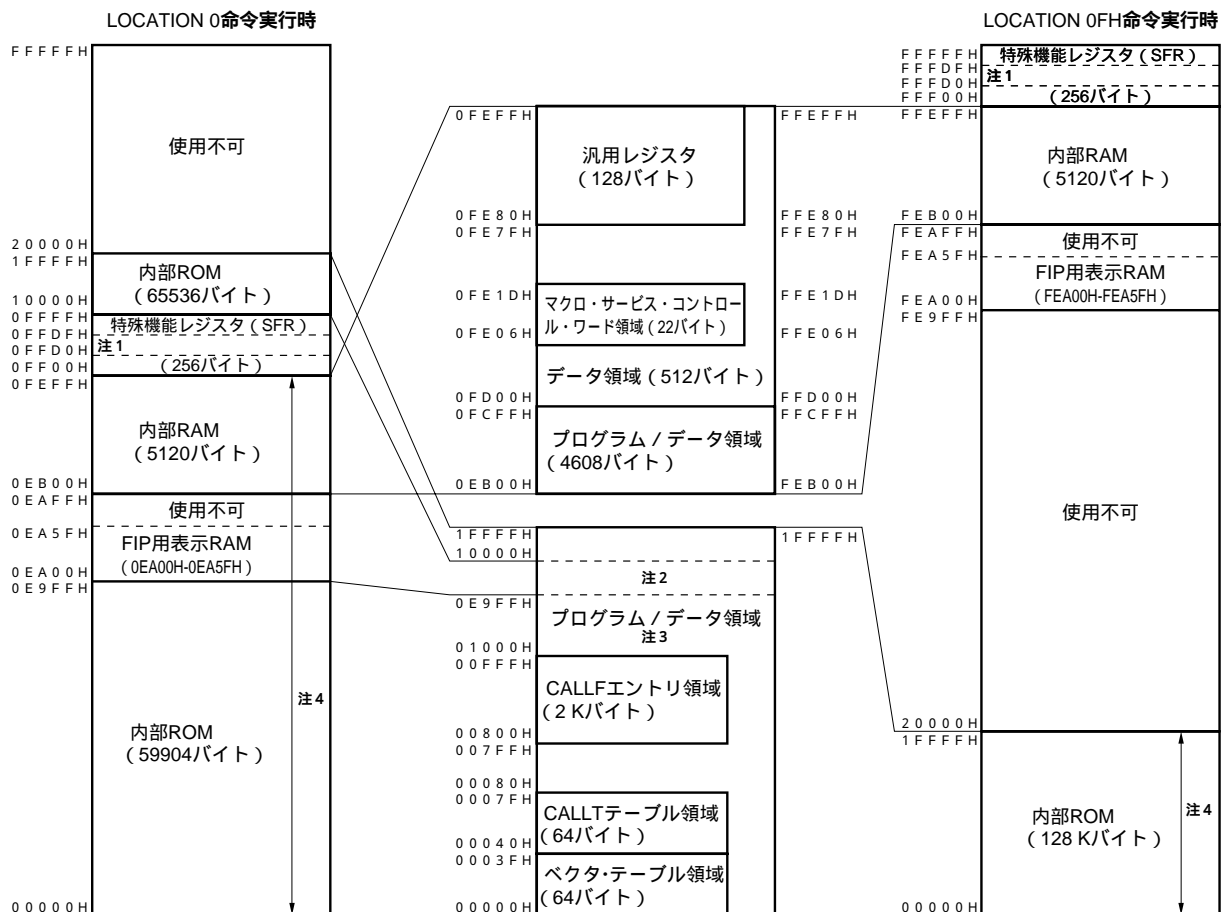
注1 . 使用不可領域

2 . この領域の5632バイトは、LOCATION 0FH命令実行時のみ内部ROMとして使用できます

3 . LOCATION 0命令実行時：92672バイト，LOCATION 0FH命令実行時：98304バイト

4 . ベース領域，リセットまたは割り込みによるエントリ領域。ただし，リセットについては内部RAMを除く

図3 - 3 μPD78F4976のメモリ・マップ



## 注1 . 使用不可領域

2 . この領域の5632バイトは、LOCATION 0FH命令実行時のみ内部ROMとして使用できます

3 . LOCATION 0 命令実行時：92672バイト，LOCATION 0FH命令実行時：98304バイト

4 . ベース領域，リセットまたは割り込みによるエントリ領域。ただし，リセットについては内部RAMを除く

## 3.2 内部ROM領域

μPD784976サブシリーズの次に示す製品は、ROMを内蔵しており、プログラムやテーブル・データなどを格納できます。

なお、LOCATION 0 命令実行時に内部ROM領域と内部データ領域が重なった場合は、内部データ領域がアクセスの対象となり、重なった部分の内部ROM領域はアクセスできません。

品名	内部ROM	アドレス空間	
		LOCATION 0 命令	LOCATION 0FH命令
μPD784975	96 K × 8 ビット	00000H-0E9FFH 10000H-17FFFH	00000H-17FFFH
μPD78F4976	128 K × 8 ビット	00000H-0E9FFH 10000H-1FFFFH	00000H-1FFFFH

内部ROMは、高速にアクセスすることが可能です。通常は、外部ROMと同等のスピードでフェッチするようになっており、メモリ拡張モード・レジスタ(MM)のIFCHビットをセット(1)することにより、高速フェッチ機能が使用され、内部ROMのフェッチを高速(2バイト単位に2システム・クロックでフェッチする)に行うようになります。

なお、外部ROMフェッチと同等な命令実行サイクルを選択した場合は、ウエイト機能によるウエイトの挿入が行われませんが、高速フェッチ使用時には、内部ROMに対してはウエイトは挿入されません。ただし、内部ROM領域に対して、外部ウエイトを設定しないでください。内部ROM領域に対して外部ウエイトが設定されると、CPUがデッドロック状態となります。デッドロック状態は、リセット入力によってのみ解除されず。

$\overline{\text{RESET}}$ 入力により、外部ROMフェッチ・サイクルと同等な命令実行サイクルとなります。

### 3.3 ベース領域

0-FFFFHの空間は、ベース領域となっています。次の用途については、ベース領域が対象となります。

- ・リセットのエントリ・アドレス
- ・割り込みのエントリ・アドレス
- ・CALLT命令のエントリ・アドレス
- ・16ビット・イミディエト・アドレッシング・モード（命令アドレスのアドレッシング時）
- ・16ビット・ダイレクト・アドレッシング・モード
- ・16ビット・レジスタ・アドレッシング・モード（命令アドレスのアドレッシング時）
- ・16ビット・レジスタ・インダイレクト・アドレッシング・モード
- ・ショート・ダイレクト16ビット・メモリ・インダイレクト・アドレッシング・モード

また、ベース領域には、ベクタ・テーブル領域、CALLT命令テーブル領域、CALLF命令エントリ領域が割り付けられています。

なお、LOCATION 0 命令実行時には、内部データ領域がベース領域内に配置されます。内部データ領域のうち、内部高速RAM領域および特殊機能レジスタ（SFR）領域からは、プログラムのフェッチは行えませんので注意が必要です。また、内部RAM領域のデータは、初期化を行ってから使用してください。

### 3.3.1 ベクタ・テーブル領域

00000H-0003FHの64バイト領域はベクタ・テーブル領域として予約されています。ベクタ・テーブル領域に $\overline{\text{RESET}}$ 入力、各割り込み要求発生により分岐する場合のプログラム・スタート・アドレスを格納しておきます。また、各割り込みでコンテキスト・スイッチングを使用する場合に、切り替え先のレジスタ・バンクの番号を格納します。

なお、ベクタ・テーブルとして使用していない部分は、プログラム・メモリまたはデータ・メモリとして使用できます。

ベクタ・テーブルに書ける値は、16ビットの値です。したがって、分岐できるのはベース領域のみです。

表3 - 1 ベクタ・テーブル・アドレス

割り込み要因	ベクタ・テーブル・アドレス
BRK命令	0 0 3 E H
オペランド・エラー	0 0 3 C H
INTWDT(ノンマスクابل)	0 0 0 4 H
INTWDT(マスクابل)	0 0 0 6 H
INTP0	0 0 0 8 H
INTP1	0 0 0 A H
INTP2	0 0 0 C H
INTTM00	0 0 0 E H
INTTM01	0 0 1 0 H
INTKS	0 0 1 2 H
INTCSI0	0 0 1 4 H
INTCSI1	0 0 1 6 H
INTTM50	0 0 1 8 H
INTTM51	0 0 1 A H
INTAD	0 0 1 C H

### 3.3.2 CALLT命令テーブル領域

00040H-0007FHの64バイトの領域には、1バイト・コール命令（CALLT）のサブルーチン・エン트리・アドレスを格納できます。

CALLT命令では、このテーブルを参照し、テーブル中に書かれているベース領域のアドレスへサブルーチンとして分岐します。CALLT命令は1バイトであるため、プログラム中で記述回数の多いサブルーチン・コールをCALLT命令とすることで、プログラムのオブジェクト・サイズを圧縮できます。なお、テーブルには最大32個のサブルーチン・エン트리・アドレスを記述できますので、記述頻度の多い順に登録することをお勧めします。

また、CALLT命令のテーブルとして使用しない場合は、通常のプログラム・メモリまたはデータ・メモリとして使用可能です。

### 3.3.3 CALLF命令エン트리領域

00800H-00FFFHの領域は、2バイト・コール命令（CALLF）で直接サブルーチン・コールできます。

CALLF命令は、2バイトのコール命令であるため、直接サブルーチン・コールのCALL命令（3バイトまたは4バイト）を使用した場合に比べ、オブジェクト・サイズを圧縮できます。

高速性を活かしたいときには、この領域に直接サブルーチンを記述することが有効です。

オブジェクト・サイズを小さくしたい場合には、この領域に無条件分岐（BR）命令を記述し、サブルーチン本体はこの領域外に置くことで、5カ所以上からコールされているサブルーチンについて、オブジェクト・サイズの圧縮が図れます。この場合、CALLFエン트리領域中ではBR命令の4バイトしか場所をとらないため、多くのサブルーチンでオブジェクト・サイズの圧縮ができます。



## 3.4 内部データ領域

内部データ領域は、内部RAM領域、特殊機能レジスタ領域で構成される領域です（図3 - 1，図3 - 2参照）。

内部データ領域は、LOCATION命令によって、内部データ領域の最終アドレスを0FFFFH（LOCATION 0命令実行時）にするか、FFFFFFH（LOCATION 0FH命令実行時）にするかを選択できます。このLOCATION命令による内部データ領域のアドレスの選択は、リセット解除直後に必ず1回実行する必要があるため、一度選択したあとは、変更することはできません。リセット解除後のプログラムは、例のようにする必要があります。内部データ領域と他の領域が同一アドレスに割り当てられた場合は、内部データ領域がアクセスの対象となり、他の領域にはアクセスできません。

```
例 RSTVCT  CSEG  AT 0
      DW    RSTSTRT
      {
INITSEG  CSEG  BASE
RSTSTRT : LOCATION 0H ; or LOCATION 0FH
      MOVG  SP, #STKBGN
      MOV   MM, #80H
```

**注意** LOCATION 0命令実行時には、リセット解除後のプログラムが内部データ領域に重ならないようにする必要があります。また、マスク不可能な割り込み処理ルーチンのエントリ・アドレスも、内部データ領域と重ならないようにしてください。なお、マスク可能割り込みのエントリ領域などについては、内部データ領域を参照する前に初期化を行う必要があります。

### 3.4.1 内部RAM領域

μPD784975は、汎用スタティックRAMを内蔵しています。

この領域は、次のように構成されています。

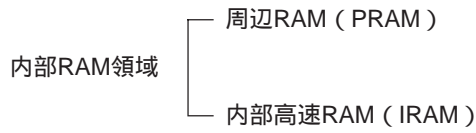


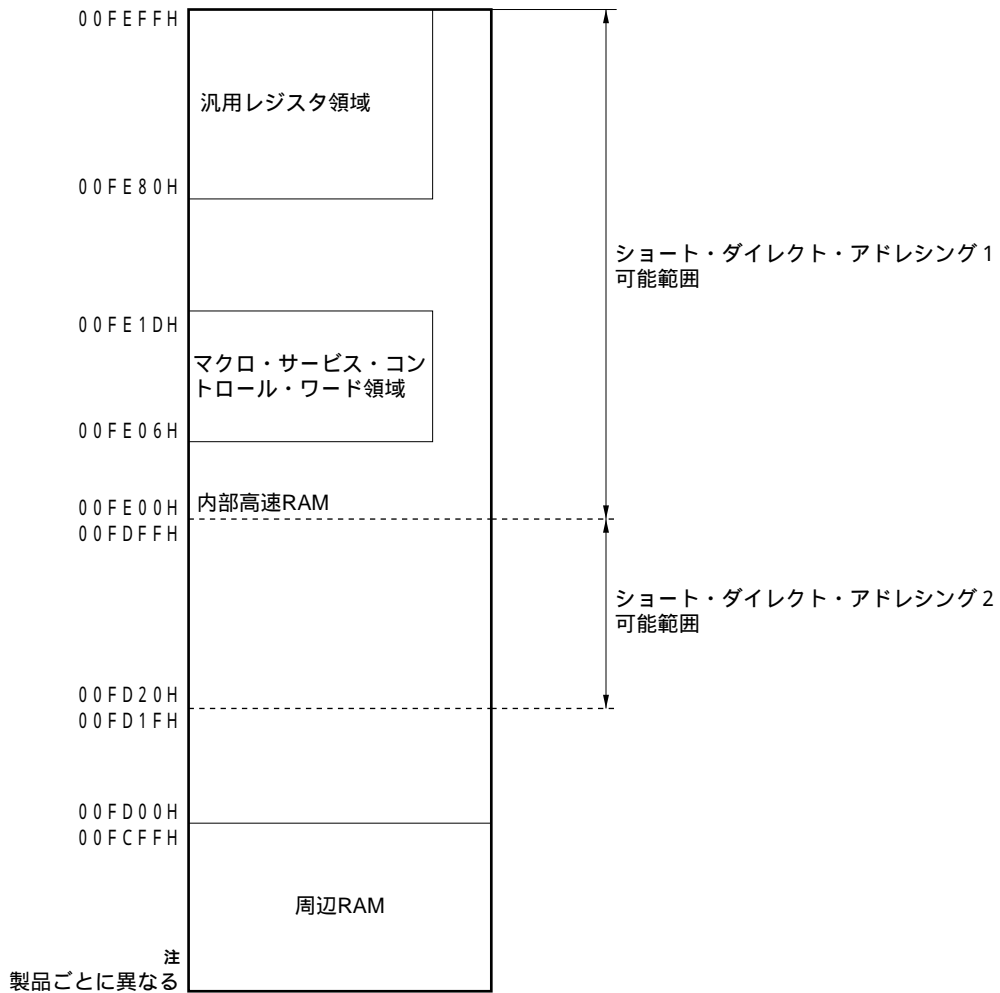
表3 - 2 内部RAM領域一覧

品名	内部RAM領域		
	内部RAM	周辺RAM : PRAM	内部高速RAM : IRAM
μPD784975	3584バイト (0F100H-0FEFFH)	3072バイト (0F100H-0FCFFH)	512バイト (0FD00H-0FEFFH)
μPD78F4956	5120バイト (0EB00H-0FEFFH)	4608バイト (0EB00H-0FCFFH)	

**備考** 表中のアドレスは、LOCATION 0 命令実行時の値です。LOCATION 0FH命令実行時には、上記の値に0F0000Hを加えてください。

内部RAMのメモリ・マップを図3 - 4に示します。

図3 - 4 内部RAMのメモリ・マップ



注   μ PD784975               : 00F100H  
      μ PD78F4976           : 00EB00H

備考  図中のアドレスは、LOCATION 0 命令実行時の値です。LOCATION 0FH 命令実行時には、上記の値に0F0000Hを加えてください。

### (1) 内部高速RAM (IRAM)

内部高速RAM (IRAM) は、高速アクセスが可能です。このうち、FD20H-FEFFFHは、高速アクセス用のショート・ダイレクト・アドレッシング・モードが使用できます。ショート・ダイレクト・アドレッシング・モードには、その対象となるアドレスにより、ショート・ダイレクト・アドレッシング1とショート・ダイレクト・アドレッシング2の2種類があります。いずれのアドレッシング・モードも、その機能は同一です。一部の命令では、ショート・ダイレクト・アドレッシング2は、ショート・ダイレクト・アドレッシング1より語長が短くなっています。詳細は、78 K/ シリーズ ユーザーズ・マニュアル 命令編 (U10905J) を参照してください。

IRAMから、プログラムのフェッチを行うことはできません。IRAMが、マッピングされているアドレスからプログラムのフェッチを行うと、CPUは暴走します。

また、IRAMには、次の領域が予約されています。

- ・汎用レジスタ領域 : FE80H-FEFFFH
- ・マクロ・サービス・コントロール・ワード領域 : FE06H-FE1DH
- ・マクロ・サービス・チャンネル領域 : FE00H-FEFFFH (アドレスは、マクロ・サービス・コントロール・ワードで指定します)

これらの領域で、予約されている機能を使用していない場合は、通常のデータ・メモリとして使用できます。

**備考** 本文中のアドレスは、LOCATION 0 命令実行時のアドレスです。LOCATION 0FH命令実行時には、本文中の値に0F0000Hを加えてください。

### (2) 周辺RAM (PRAM)

周辺RAM (PRAM) は、普通のプログラム・メモリまたはデータ・メモリとして使用します。プログラム・メモリとして使用する場合は、事前に、プログラムによって周辺RAMにプログラムを書き込んでおく必要があります。

なお、周辺RAMからのプログラム・フェッチは、2バイト単位に2クロックで行うことができるため高速です。

### 3.4.2 特殊機能レジスタ（SFR）領域

0FF00H-0FFFFHの領域には、オンチップ周辺ハードウェアの特殊機能レジスタ（SFR）がマッピングされています（図3 - 2，図3 - 3参照）。

**注意** この領域内で、SFRのマッピングされていないアドレスをアクセスしないでください。誤ってアクセスすると、CPUがデッドロック状態になることがあります。デッドロック状態は、リセット入力によってのみ解除されます。

**備考** 本文中のアドレスは、LOCATION 0 命令実行時のアドレスです。LOCATION 0FH命令実行時には、本文中の値に0F0000Hを加えてください。

### 3.5 $\mu$ PD78F4976のメモリ・マッピング

$\mu$ PD78F4976は、フラッシュ・メモリを128 Kバイト、内部RAMを5120バイト内蔵しています。

$\mu$ PD78F4976は、ソフトウェアにより内部メモリの一部を使用しないようにするための機能（メモリ・サイズの切り替え機能）を内蔵しています。

メモリ・サイズの切り替えは、内部メモリ・サイズ切り替えレジスタ（IMS）によって行います。

IMSを設定することにより、内部メモリ（ROM, RAM）容量の異なるマスクROM製品のメモリ・マッピングと同一のメモリ・マッピングにすることができます。

IMSは、8ビット・メモリ操作命令で書き込みのみ可能です。

$\overline{\text{RESET}}$ 入力により、FFHになります。

図3-5 内部メモリ・サイズ切り替えレジスタ（IMS）のフォーマット

アドレス：0FFFCH リセット時：FFH W

略号	7	6	5	4	3	2	1	0
IMS	1	1	ROM1	ROM0	1	1	RAM1	RAM0

ROM1	ROM0	内部ROM容量の選択
0	0	設定禁止
0	1	設定禁止
1	0	96 Kバイト
1	1	128 Kバイト

RAM1	RAM0	内部RAM容量の選択 <sup>注</sup>
0	0	設定禁止
0	1	3584バイト
1	0	設定禁止
1	1	5120バイト

注 内部RAM容量は、周辺RAM容量と高速RAM容量の合計です。

注意1 IMSは、マスクROM製品（ $\mu$ PD784975）にはありません。

また、IMSへの書き込み命令をマスクROM製品で実行しても動作に影響を与えません。

2 インサーキット・エミュレータでエミュレーションCPUを $\mu$ PD78F4976に選択した場合、IMSへFFH以外の書き込み命令を実行しても、常に“FFH”になります。

マスクROM製品の同一のメモリ・マップにするIMSの設定値を表3-3に示します。

表3-3 内部メモリ・サイズ切り替えレジスタ (IMS) の設定値

対象のマスクROM製品	IMSの設定値
μPD784975	EDH

## 3.6 制御レジスタ

制御レジスタには、プログラム・カウンタ (PC)、プログラム・ステータス・ワード (PSW)、スタック・ポインタ (SP) があります。

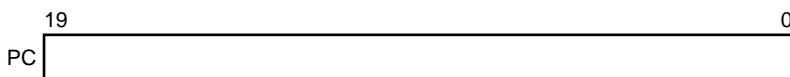
### 3.6.1 プログラム・カウンタ (PC)

次に実行するプログラムのアドレス情報を保持する20ビット・バイナリ・カウンタです (図3-6参照)。

通常、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した場合、イミディエイト・データやレジスタの内容がセットされます。

RESET入力により、0番地と1番地の16ビット・データがPCの下位16ビットに、0000がPCの上位4ビットにセットされます。

図3-6 プログラム・カウンタ (PC) のフォーマット



### 3.6.2 プログラム・ステータス・ワード (PSW)

プログラム・ステータス・ワード (PSW) は、命令の実行の結果によってセット、リセットされる各種フラグで構成される16ビット・レジスタです。

上位8ビット (PSWH)、下位8ビット (PSWL) 単位でリード・アクセス/ライト・アクセスします。また、ビット操作命令により、各フラグの操作ができます。

PSWの内容は、ベクタ割り込み要求の受け付け時、およびBRK命令の実行時に自動的にスタックに退避し、RETI命令またはRETB命令の実行時に自動的に復帰します。また、コンテキスト・スイッチング使用時には、RP3に自動的に退避し、RETCS命令またはRETCSB命令の実行時に自動的に復帰します。

RESET入力により、全ビットがリセット (0) されます。

図3-7で“0”と書かれているビットには、必ず“0”を書き込んでください。また、“-”と書かれているビットの内容は、読み出し時には不定となります。

図3-7 プログラム・ステータス・ワード (PSW) のフォーマット

略号	7	6	5	4	3	2	1	0
PSWH	UF	RBS2	RBS1	RBS0	-	-	-	-
PSWL	S	Z	RSS	AC	IE	P/V	0	CY

各フラグについて、次に示します。

#### (1) キャリー・フラグ (CY)

演算結果のキャリー、ポローを記憶するフラグです。

また、シフト・ローテート命令実行時は、シフト・アウトされた値を記憶し、ビット操作命令実行時は、ビット・アキュムレータとして機能します。

CYフラグの状態は、条件付き分岐命令でテストできます。

#### (2) パリティ/オーバーフロー・フラグ (P/V)

P/Vフラグは、演算命令の実行に伴い、次の2種類の動作をします。

P/Vフラグの状態は、条件付き分岐命令でテストできます。

##### ・パリティ・フラグ動作

論理演算命令、シフト・ローテート命令、CHKL, CHKLA命令の実行の結果、セット(1)されたビット数が偶数のときにはセット(1)されます。奇数のときにはリセット(0)されます。ただし、16ビットのシフト命令の場合、演算結果の下位8ビットのみパリティ・フラグに有効です。

##### ・オーバーフロー・フラグ動作

算術演算命令の実行の結果、2の補数で表現される数値範囲を越えたときのみセット(1)されます。それ以外のときにはリセット(0)されます。具体的には、MSBからのキャリーとMSBへのキャリーの排他的論理和の結果が、このフラグの内容になります。たとえば、8ビットの算術演算では、2の補数の範囲は80H(-128)~7FH(+127)であり、演算結果がこの範囲以外になったときセット(1)され、範囲内のときはリセット(0)されます。



例 8ビット加算命令実行時のオーバーフロー・フラグの動作を次に示します。

78H (+120) と 69H (+105) の加算を行うと、演算結果が E1H (+225) となり、2の補数の上限を越えるため、P/Vフラグがセット(1)されます。また、2の補数表現で、E1Hは-31になります。

$$\begin{array}{r}
 78H (+120) = \quad 0111 \ 1000 \\
 + ) \underline{69H (+105)} = + ) \underline{0110 \ 1001} \\
 \hline
 0 \ 1110 \ 0001 = -31 \ P/V = 1
 \end{array}$$

CY

また、次のような2つの負数の加算は、演算結果が2つの補数の範囲内にあるため、P/Vフラグはリセット(0)されます。

$$\begin{array}{r}
 FBH (-5) = \quad 1111 \ 1011 \\
 + ) \underline{F0H (-16)} = + ) \underline{1111 \ 0000} \\
 \hline
 1 \ 1110 \ 1011 = -21 \ P/V = 0
 \end{array}$$

CY

### (3) 割り込み要求許可フラグ(IE)

CPUの割り込み要求受け付け動作を制御するフラグです。

“0”のときは割り込み禁止となり、ノンマスカブル割り込み、およびマスク解除したマクロ・サービスのみ受け付けが可能となります。それ以外は、すべて禁止されます。

“1”のときは割り込み許可状態となり、割り込み要求受け付けの許可は、各割り込み要求に対応する割り込みマスク・フラグおよび各割り込みの優先順位により制御されます。

EI命令実行によりセット(1)され、DI命令実行または割り込みの受け付けでリセット(0)されます。

### (4) 補助キャリー・フラグ(AC)

演算の結果、ビット3からのキャリーがあったとき、またはビット3へのボローがあったときにセット(1)されます。それ以外のときにはリセット(0)されます。

ADJBA, ADJBS命令実行時に使用されます。

**(5) レジスタ・セット選択フラグ (RSS)**

X, A, C, Bとして機能する汎用レジスタ, およびAX, BCとして機能する汎用レジスタ・ペア (16ビット) を指定するフラグです。

このフラグは, 78K/ シリーズとの互換性を保つために用意されているフラグです。78K/ シリーズ用のプログラムを流用する場合を除いて, 必ず0にしてください。

**(6) ゼロ・フラグ (Z)**

演算の結果が“0”であることを記憶するフラグです。

演算の結果が“0”のときにセット(1)されます。それ以外のときにはリセット(0)されます。

Zフラグの状態は, 条件付き分岐命令でテストできます。

**(7) サイン・フラグ (S)**

演算の結果, MSBが“1”であることを記憶するフラグです。

演算の結果, MSBが“1”のときにセット(1)されます。“0”のときにはリセット(0)されます。Sフラグの状態は, 条件付き分岐命令でテストできます。

**(8) レジスタ・バンク選択フラグ (RBS0-RBS2)**

8個のレジスタ・バンク (レジスタ・バンク0-レジスタ・バンク7) のうち, 1つを選択する3ビットのフラグです (表3-4参照)。

SEL RBn命令の実行などにより選択されたレジスタ・バンクを示す3ビットの情報が格納されています。

表3-4 レジスタ・バンクの選択

RBS2	RBS1	RBS0	指定レジスタ・バンク
0	0	0	レジスタ・バンク0
0	0	1	レジスタ・バンク1
0	1	0	レジスタ・バンク2
0	1	1	レジスタ・バンク3
1	0	0	レジスタ・バンク4
1	0	1	レジスタ・バンク5
1	1	0	レジスタ・バンク6
1	1	1	レジスタ・バンク7

**(9) ユーザ・フラグ (UF)**

ユーザ・プログラム上でセットおよびリセットし, プログラムの制御に利用できるフラグです。

### 3.6.3 RSSビットの使用方法

基本的にRSSビットは、常時、0に固定して使用してください。

次の説明は、78K/ シリーズ用のプログラムを流用する場合で、流用するプログラムがRSSビットを1にしている場合のための説明です。RSSビットを0に固定して使用する場合には読む必要はありません。

RSSビットは、A (R1) , X (R0) , B (R3) , C (R2) , AX (RP0) , BC (RP1) の持っている機能をR4-R7 (RP2, RP3) レジスタでも使用できるようにするためのビットです。このビットを有効に使用することで、プログラム・サイズやプログラムの実行に関して効率の良いプログラムを作成することができます。

しかし、不用意に使用すると思わぬ不具合が発生することがあります。したがって、通常は、RSSビットは0にして使用してください。RSSビットを1にして使用するのは、78K/ シリーズ用のプログラムを流用する場合だけにしてください。

すべてのプログラム中で、RSSビットを0にして使用することにより、プログラム作成およびデバッグ作業の効率が向上します。

RSSビットを1にして使用しているプログラムを流用する場合でも、可能であれば、RSSビットを1にしないプログラムへ修正してから流用することを推奨します。

#### (1) RSSビットの仕様

オペレーション一覧 (18.2 参照) で、オペランド欄にA, X, B, C, AXレジスタが直接記載されている命令で使用するこれらのレジスタ

インプライド・アドレッシングにより、A, AX, B, Cレジスタを使用する命令で、インプライドで指定されるレジスタ

インデクスト・アドレッシング、ベースト・インデクスト・アドレッシングでA, B, Cレジスタを使用する命令で、アドレッシング時に使用するレジスタ

これらの場合に使用するレジスタをRSSビットによって、次のように切り替えます。

・RSS = 0 の場合

A R1, X R0, B R3, C R2, AX RP0, BC RP1

・RSS = 1 の場合

A R5, X R4, B R7, C R6, AX RP2, BC RP3

前述以外で使用するレジスタは、RSSビットの内容にかかわらず常に同一のレジスタになります。NEC製のアセンブラ（RA78K4）では、A, X, B, C, AX, BCレジスタについて、この名前で記述されている場合にどちらのレジスタ用の命令コードを生成するかは、アセンブラのRSS疑似命令で決定されます。

RSSビットをセット、リセットする場合には、その命令の直前（または直後）に、必ずRSS疑似命令を記述してください（下記例参照）。

### プログラム例

- ・RSS = 0 にする場合

```
RSS 0          ; RSS疑似命令
CLR1 PSWL. 5
MOV B, A       ; この記述は、“MOV R3, R1” に該当します。
```

- ・RSS = 1 にする場合

```
RSS 1          ; RSS疑似命令
SET1 PSWL. 5
MOV B, A       ; この記述は、“MOV R7, R5” に該当します。
```

### (2) RA78K4での命令コードの生成方法

- ・RA78K4では、命令のオペレーション一覧のオペランド欄にAまたはAXが直接記載されている命令と同機能の命令がある場合、オペランド欄にAまたはAXが直接記載されている命令コードを優先して生成します。

**例** MOV A, r命令でrをBとした場合と、MOV r, r'命令でrをA, r'をBとした場合では同一の機能です。また、アセンブラのソース・プログラム上では同一の記述（MOV A, B）となります。この場合、RA78K4は、MOV A, r命令に相当するコードを生成します。

- ・オペランド欄にr, r', rp, rp'が指定されている命令で, A, X, B, C, AX, BCが記述された場合, RA78K4のRSS疑似命令のオペランドによって, A, X, B, C, AX, BC命令はそれぞれ次のレジスタを指定する命令コードを生成します。

レジスタ	RSS = 0	RSS = 1
A	R1	R5
X	R0	R4
B	R3	R7
C	R2	R6
AX	RP0	RP2
BC	RP1	RP3

- ・オペランド欄のr, r', rp, rp'にR0-R7, RP0-RP4を記述した場合, その指定どおりの命令コードを出力します (オペランド欄にA, AXが直接記載されている命令コードは出力しません)。
- ・インデクスト・アドレッシング, ベースト・インデクスト・アドレッシングで使用するA, B, Cレジスタは, R1, R3, R2やR5, R7, R6といった記述はできません。

### (3) 使用上の注意

RSSビットを切り替えることで, 2組のレジスタ・セットを持つのと同様の効果が得られます。しかし, プログラムの静的な記述とプログラム実行時の動的なRSSビットの変化を常に一致するように, 十分に注意をしてプログラムを記述しなければなりません。

また, RSS = 1にしたプログラムは, コンテキスト・スイッチング機能を使用しているプログラムでは使用できないので, プログラムの流用性が悪くなります。さらに, 同一の名称で異なるレジスタを使用するため, プログラムの可読性が悪くなり, デバッグ作業が難しくなります。したがって, RSS = 1として使用しなければならない場合は, これらの欠点を十分に考慮のうえ, プログラムを作成してください。

なお, RSSビットで指定されないレジスタは, 絶対名称を記述することによりアクセスできます。

### 3.6.4 スタック・ポインタ (SP)

スタック領域 (LIFO形式 : 00000H-FFFFFFH) の先頭アドレスを保持する24ビット・レジスタです (図3-8参照)。サブルーチン処理や割り込み処理時にスタック領域をアドレスするために使用されます。上位4ビットには、必ず“0”を書き込んでください。

SPの内容は、スタック領域への書き込み前にデクリメントされ、スタック領域からの読み出し後にインクリメントされます (図3-9, 図3-10参照)。

SPは、専用命令によりアクセスします。

SPの内容は、 $\overline{\text{RESET}}$ 入力により不定になりますので、リセット解除直後に (サブルーチン・コールや割り込みを受け付ける前に) 必ず初期化プログラムによりSPをイニシャライズしてください。

#### 例 SPのイニシャライズ

```
MOVG SP, #0FEE0H ; SP 0FEE0H (FEDFHから使用の場合)
```

図3-8 スタック・ポインタ (SP) のフォーマット

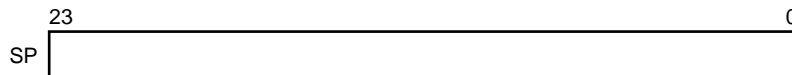


図3 - 9 スタック領域へ退避されるデータ

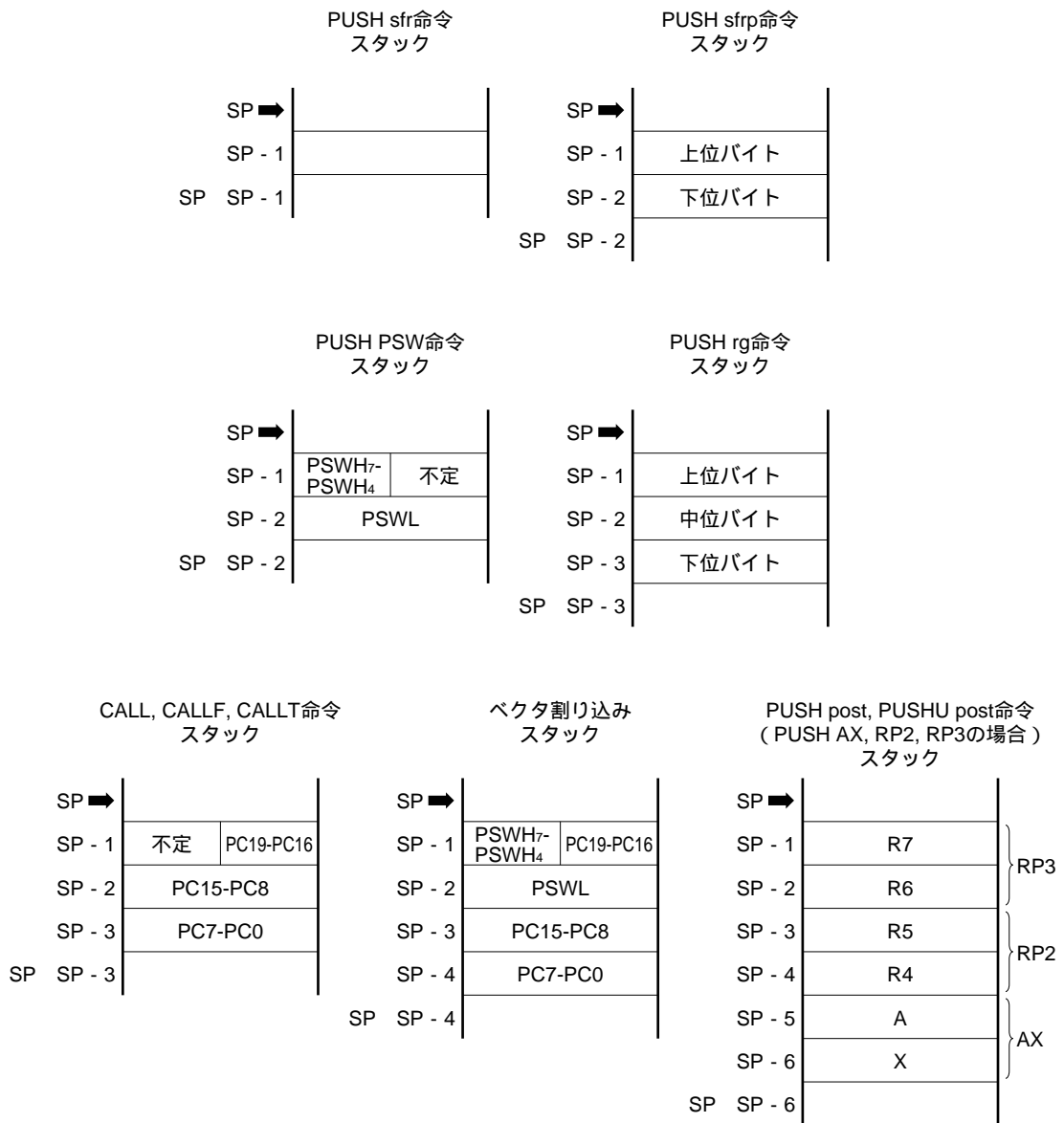
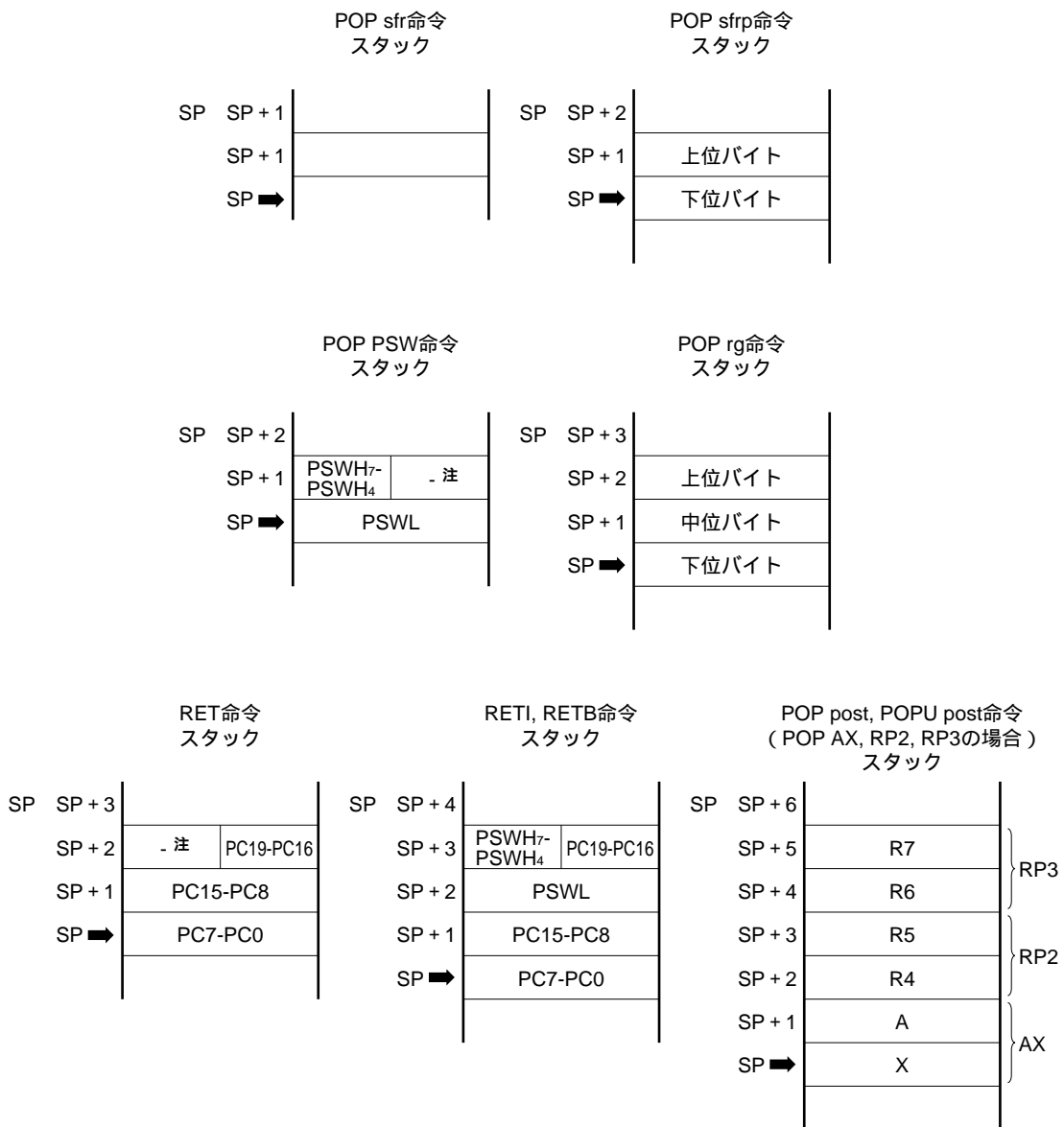


図3 - 10 スタック領域から復帰されるデータ



注 この4ビットのデータは無視されます。



注意1．スタック・アドレッシングでは、1 Mバイトすべての空間でアクセス可能ですが、SFR領域および内部ROM領域にスタック・エリアを確保することはできません。

2．スタック・ポインタ (SP) は、 $\overline{\text{RESET}}$  入力により不定になります。また、SPが不定の状態でもノンマスカブル割り込みは受け付け可能です。したがって、リセット解除直後のSPが不定な状態でノンマスカブル割り込みの要求が発生すると、予期しない動作を行う場合があります。この危険を回避するために、リセット解除後のプログラムは、必ず次のようにしてください。

```
RSTVCT  CSEG  AT 0
          DW   RSTSTRT
          {
INITSEG  CSEG  BASE
RSTSTRT : LOCATION 0H ; or LOCATION 0FH
          MOVG SP, #STKBGN
```

## 3.7 汎用レジスタ

### 3.7.1 構成

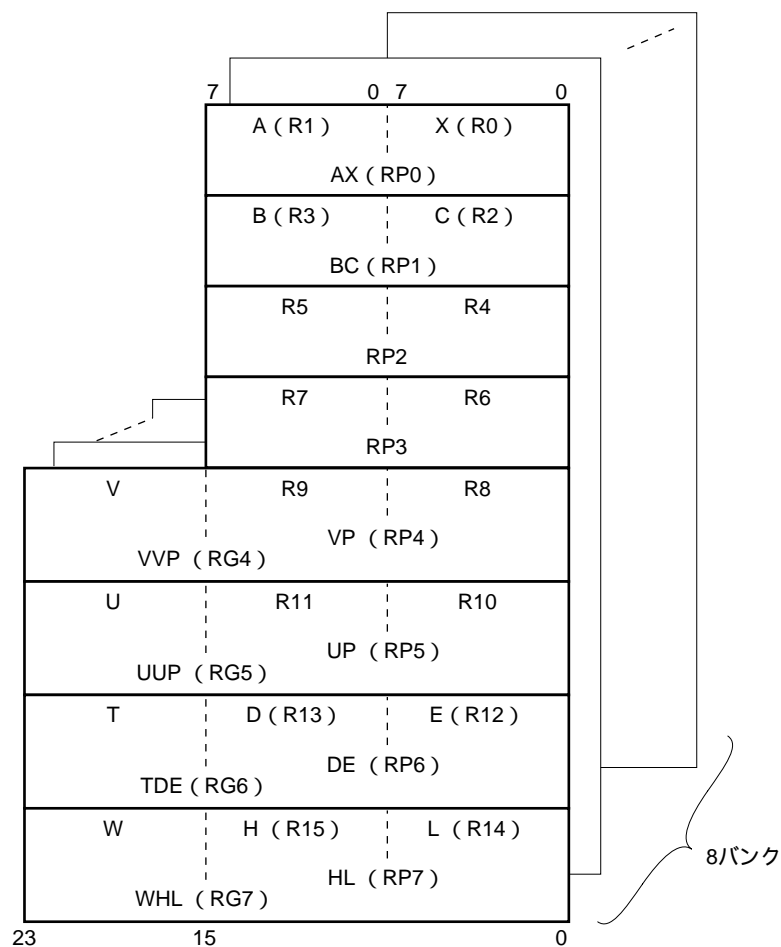
8ビットの汎用レジスタ16本で構成されています。また、8ビット汎用レジスタ2本を組み合わせて、16ビット汎用レジスタとして使用することもできます。さらに、16ビット汎用レジスタのうち4本は、アドレス拡張用の8ビット・レジスタと組み合わせて、24ビット・アドレス指定用レジスタとして使用することができます。

アドレス拡張用のV, U, T, Wレジスタを除いた汎用レジスタは、内部RAMにマッピングされています。

これらのレジスタ・セットは、8バンク用意されており、ソフトウェアまたはコンテキスト・スイッチング機能により切り替えて使用することができます。

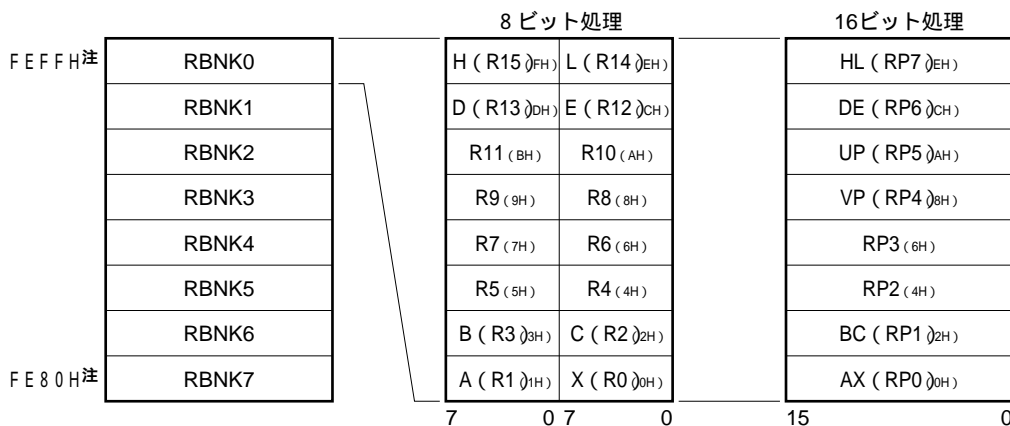
$\overline{\text{RESET}}$ 入力により、レジスタ・バンク0が選択されます。また、実行中のプログラムで使用しているレジスタ・バンクは、PSW内のレジスタ・バンク選択フラグ (RBS0, RBS1, RBS2) を読み出すことによって確認できます。

図3 - 11 汎用レジスタのフォーマット



備考 ( )内は絶対名称です。

図3 - 12 汎用レジスタのアドレス



注 LOCATION 0 命令実行時。LOCATION 0FH命令実行時のアドレスは、上記の値に0F0000Hを加えた値

注意 R4, R5, R6, R7, RP2, RP3は、PSWのRSSビットを1にすることで、それぞれ、X, A, C, B, AX, BCレジスタとして使用することができますが、この機能を使用するのは78K/ シリーズ用のプログラムを流用する場合だけにしてください。

備考 レジスタ・バンクを変更する場合で、元のレジスタ・バンクに戻す必要がある場合は、PUSH PSW命令でPSWをスタックへ退避してからSEL RBn命令を実行してください。元に戻すときは、スタックの位置に変化がなければPOP PSW命令で戻せます。

なお、ベクタ割り込み処理プログラムなどでレジスタ・バンクを変更する場合には、PSWは割り込みの受け付け時に自動的にスタックへ退避され、RETI, RETB命令で復帰されますので、割り込み処理ルーチンで使用するレジスタ・バンクが1つの場合には、単にSEL RBn命令を実行するだけで、PUSH PSWやPOP PSW命令を実行する必要はありません。

例 レジスタ・バンク2を指定する場合

```

...
PUSH PSW
SEL RB2
...
POP PSW
...

```

} レジスタ・バンク2で動作

元のレジスタ・バンクで動作

### 3.7.2 機能

汎用レジスタは、それぞれ8ビット単位で操作できるほか、2個の8ビット・レジスタがペアとなって16ビット単位で操作することもできます。また、16ビット・レジスタのうち4本は、アドレス拡張用の8ビット・レジスタと組み合わせて24ビット単位で操作できます。

また、各レジスタは、演算結果の一時保管や、レジスタ間演算命令のオペランドとして汎用的に使用することが可能です。

0FE80H-0FEFFH (LOCATION 0 命令実行時。LOCATION 0FH命令実行時は0FFE80H-0FFEFFH) の領域は、汎用レジスタ・エリアとして使用する、しないにかかわらず、通常のデータ・メモリとしてアドレス指定し、アクセスすることができます。

78K/ シリーズでは、8つのレジスタ・バンクを持っているので、通常の処理と割り込み時の処理でレジスタ・バンクを使い分けることにより、効率のよいプログラムを作成できます。

各レジスタは、それぞれ次に示す固有機能を持っています。

#### A (R1) :

- ・ 8ビット・データの転送や演算処理の中心となるレジスタです。8ビット・データに対するすべてのアドレッシング・モードと組み合わせて使用できます。
- ・ ビット・データの格納にも使用できます。
- ・ インデクスト・アドレッシング、ベースト・インデクスト・アドレッシング時のオフセット値を格納するレジスタとして使用できます。

#### X (R0) :

- ・ ビット・データの格納に使用できます。

#### AX (RP0) :

- ・ 16ビット・データの転送や演算処理の中心となるレジスタです。16ビット・データに対するすべてのアドレッシング・モードと組み合わせて使用できます。

#### AXDE :

- ・ DIVUX, MACW, MACSW命令実行時に、32ビット・データの格納用として使用されます。

#### B (R3) :

- ・ ループ・カウンタとしての機能を持っており、DBNZ命令で使用できます。
- ・ インデクスト・アドレッシング、ベースト・インデクスト・アドレッシング時のオフセット値を格納するレジスタとして使用できます。
- ・ MACW, MACSW命令のデータ・ポインタとして使用します。

## C (R2) :

- ・ループ・カウンタとしての機能を持っており、DBNZ命令で使用できます。
- ・ベースト・インデクスト・アドレッシング時のオフセット値を格納するレジスタとして使用できます。
- ・ストリング命令、SACW命令のカウンタとして使用します。
- ・MACW、MACSW命令のデータ・ポインタとして使用します。

## RP2 :

- ・コンテキスト・スイッチング使用時に、プログラム・カウンタ (PC) の下位16ビットを退避するために使用します。

## RP3 :

- ・コンテキスト・スイッチング使用時に、プログラム・カウンタ (PC) の上位4ビットおよびプログラム・ステータス・ワード (PSW) (PSWHのビット0-3を除く) を退避するために使用します。

## VVP (RG4) :

- ・ポインタとしての機能を持っており、レジスタ・インダイレクト・アドレッシング、ベースト・アドレッシング、ベースト・インデクスト・アドレッシング時に、ベース・アドレスを指定するレジスタとして動作します。

## UUP (RG5) :

- ・ユーザ・スタック・ポインタとしての機能を持っており、PUSHU、POPU命令により、システム・スタックとは別のスタックを実現することができます。
- ・ポインタとしての機能を持っており、レジスタ・インダイレクト・アドレッシング、ベースト・アドレッシング時に、ベース・アドレスを指定するレジスタとして動作します。

## DE (RP6), HL (RP7) :

- ・インデクスト・アドレッシング、ベースト・インデクスト・アドレッシング時には、オフセット値を格納するレジスタとして動作します。

## TDE (RG6) :

- ・ポインタとしての機能を持っており、レジスタ・インダイレクト・アドレッシング、ベースト・アドレッシング時に、ベース・アドレスを指定するレジスタとして動作します。
- ・ストリング命令、SACW命令のポインタとして動作します。

WHL (RG7) :

- ・24ビット・データの転送や演算処理の中心となるレジスタです。
- ・ポインタとしての機能を持っており、レジスタ・インダイレクト・アドレッシング、ベースト・アドレッシング時に、ベース・アドレスを指定するレジスタとして動作します。
- ・ストリング命令、SACW命令のポインタとして動作します。

各レジスタは、それぞれの固有機能を重視した機能名称 (X, A, C, B, E, D, L, H, AX, BC, VP, UP, DE, HL, VVP, UUP, TDE, WHL) のほか、絶対名称 (R0-R15, RP0-RP7, RG4-RG7) でも記述することができます。対応は、表3 - 5を参照してください。

表3 - 5 機能名称-絶対名称の対応

(a) 8ビット・レジスタ

絶対名称	機能名称	
	RSS = 0	RSS = 1 <sup>注</sup>
R0	X	
R1	A	
R2	C	
R3	B	
R4		X
R5		A
R6		C
R7		B
R8		
R9		
R10		
R11		
R12	E	E
R13	D	D
R14	L	L
R15	H	H

(b) 16ビット・レジスタ

絶対名称	機能名称	
	RSS = 0	RSS = 1 <sup>注</sup>
RP0	AX	
RP1	BC	
RP2		AX
RP3		BC
RP4	VP	VP
RP5	UP	UP
RP6	DE	DE
RP7	HL	HL

(c) 24ビット・レジスタ

絶対名称	機能名称
RG4	VVP
RG5	UUP
RG6	TDE
RG7	WHL

注 RSS = 1として使用するのは、78K/ シリーズ用のプログラムを流用する場合だけにしてください。

備考 R8-R11には機能名称はありません。

## 3.8 特殊機能レジスタ (SFR)

内蔵周辺ハードウェアのモード・レジスタ、コントロール・レジスタなどの特別な機能が割り付けられたレジスタで、0FF00H-0FFFFH<sup>注</sup>の256バイトの空間にマッピングされています。

**注** LOCATION 0 命令実行時。LOCATION 0FH命令実行時はFFF00H-FFFFFH

**注意** この領域内で、SFRの割り付けられていないアドレスをアクセスしないでください。誤ってアクセスすると、 $\mu$ PD784975がデッドロック状態となることがあります。デッドロック状態は、リセット入力によってのみ解除されます。

表3 - 6 に特殊機能レジスタ (SFR) の一覧を示します。表中の項目の意味は次のようになります。

- ・略号..... 内蔵されたSFRを示す記号。NEC製アセンブラ (RA78K4) では予約語になっています。Cコンパイラ (CC78K4) では#pragma sfr指令により、sfr変数として使用できます。
- ・R/W..... 該当するSFRが読み出し / 書き込みが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
  - W : 書き込み (Write) のみ
- ・操作可能ビット単位... 該当するSFRを操作する場合に、適応可能な操作ビット単位を示します。16ビット操作可能なSFRはオペランドのsfrpに記述でき、アドレスで指定する場合は偶数アドレスを記述します。
  - 1ビット操作可能なSFRは、ビット操作命令に記述できます。
- ・リセット時..... $\overline{\text{RESET}}$ 入力時の各レジスタの状態を示します。

表3-6 特殊機能レジスタ(SFR)一覧(1/2)

アドレス <sup>注1</sup>	特殊機能レジスタ(SFR)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	
0FF00H	ポート0	P0		R			-	不定
0FF01H	ポート1	P1					-	
0FF02H	ポート2	P2		R/W			-	
0FF04H	ポート4	P4					-	
0FF05H	ポート5	P5					-	
0FF06H	ポート6	P6					-	
0FF07H	ポート7	P7					-	
0FF08H	ポート8	P8					-	
0FF09H	ポート9	P9					-	
0FF0AH	ポート10	P10					-	
0FF0BH	ポート・リード7	PLR7		R			-	不定
0FF0CH	ポート・リード8	PLR8					-	
0FF0DH	ポート・リード9	PLR9					-	
0FF10H	16ビット・タイマ/カウンタ0	TM0			-	-		0000H
0FF12H	16ビット・キャプチャ/コンペア・レジスタ00(16ビット・タイマ/カウンタ)	CR00		R/W	-	-		
0FF14H	16ビット・キャプチャ/コンペア・レジスタ01(16ビット・タイマ/カウンタ)	CR01			-	-		00H
0FF16H	キャプチャ/コンペア・コントロール・レジスタ0	CRC0					-	
0FF18H	16ビット・タイマ・モード・コントロール・レジスタ0	TMC0					-	
0FF1CH	プリスケラ・モード・レジスタ0	PRM0					-	
0FF22H	ポート2モード・レジスタ	PM2					-	
0FF24H	ポート4モード・レジスタ	PM4					-	
0FF25H	ポート5モード・レジスタ	PM5					-	
0FF26H	ポート6モード・レジスタ	PM6					-	
0FF32H	プルアップ抵抗オプション・レジスタ2	PU2					-	00H
0FF4EH	プルアップ抵抗オプション・レジスタ	PUO					-	
0FF50H	8ビット・タイマ/カウンタ50	TM50	TM5	R	-			04H
0FF51H	8ビット・タイマ/カウンタ51	TM51			-		-	
0FF52H	8ビット・コンペア・レジスタ50	CR50	CR5	R/W	-			
0FF53H	8ビット・コンペア・レジスタ51	CR51			-		-	
0FF54H	8ビット・タイマ・モード・コントロール・レジスタ50	TMC50	TMC5				-	
0FF55H	8ビット・タイマ・モード・コントロール・レジスタ51	TMC51					-	
0FF56H	タイマ・クロック選択レジスタ50	TCL50	TCL5				-	
0FF57H	タイマ・クロック選択レジスタ51	TCL51					-	
0FF7AH	発振モード選択レジスタ	CC					-	
0FF80H	A/Dコンバータ・モード・レジスタ	ADM					-	
0FF81H	A/Dコンバータ入力選択レジスタ	ADIS					-	
0FF83H	A/D変換結果レジスタ	ADCR		R	-		-	不定
0FF90H	シリアル動作モード・レジスタ0	CSIM0		R/W			-	00H
0FF91H	シリアル動作モード・レジスタ1	CSIM1					-	
0FF94H	シリアルI/Oシフト・レジスタ0	SIO0			-		-	

注1 . LOCATION 0命令を実行した場合。LOCATION 0FH命令を実行した場合は、この値に“F0000H”を加えます。  
 2 . リセットにより各ポートは入力モードに初期化されるので、実際に“00H”が読み出されるわけではありません。出力ラッチは“0”に初期化されます。



表3-6 特殊機能レジスタ(SFR)一覧(2/2)

アドレス <sup>注</sup>	特殊機能レジスタ(SFR)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	
0FF95H	シリアルI/Oシフト・レジスタ1	SIO1		R/W	-		-	00H
0FFA0H	外部割り込み立ち上がりエッジ許可レジスタ0	EGP0					-	
0FFA2H	外部割り込み立ち下がりエッジ許可レジスタ0	EGN0					-	
0FFA8H	インサースビス・プライオリティ・レジスタ	ISPR		R			-	
0FFA9H	割り込み選択コントロール・レジスタ	SNMI		R/W			-	
0FFAAH	割り込みモード・コントロール・レジスタ	IMC					-	80H
0FFACH	割り込みマスク・レジスタ0L	MK0L	MK0					FFH
0FFADH	割り込みマスク・レジスタ0H	MK0H					-	
0FFB0H	表示モード・レジスタ0	DSPM0					-	10H
0FFB2H	表示モード・レジスタ1	DSPM1					-	01H
0FFB4H	表示モード・レジスタ2	DSPM2					-	00H
0FFC0H	スタンバイ・コントロール・レジスタ	STBC			-		-	30H
0FFC2H	ウォッチドッグ・タイマ・モード・レジスタ	WDM			-		-	00H
0FFC4H	メモリ拡張モード・レジスタ	MM					-	20H
0FFCFH	発振安定時間指定レジスタ	OSTS					-	00H
0FFD0H- 0FFDFH	外部SFR領域	-					-	-
0FFE0H	割り込み制御レジスタ(INTWDT)	WDTIC					-	43H
0FFE1H	" (INTP0)	PIC0					-	
0FFE2H	" (INTP1)	PIC1					-	
0FFE3H	" (INTP2)	PIC2					-	
0FFE4H	" (INTTM00)	TMIC00					-	
0FFE5H	" (INTTM01)	TMIC01					-	
0FFE6H	" (INTKS)	KSIC					-	
0FFE7H	" (INTCSI0)	CSIIC0					-	
0FFE8H	" (INTCSI1)	CSIIC1					-	
0FFE9H	" (INTTM50)	TMIC50					-	
0FFEAH	" (INTTM51)	TMIC51					-	
0FFEBH	" (INTAD)	ADIC					-	

注 LOCATION 0命令を実行した場合。LOCATION 0FH命令を実行した場合は、この値に“F000H”を加えます。

### 3.9 注意事項

(1) 内部高速RAM領域 (LOCATION 0 命令実行時: 0FD00H-0FEFFH, LOCATION 0FH 命令実行時: FFD00H-FFEFFH) からのプログラム・フェッチはできません。

(2) 特殊機能レジスタ (SFR)

0FF00H-0FFFFH<sup>注</sup>の領域内で、SFRの割り付けられていないアドレスをアクセスしないでください。誤ってアクセスすると、μPD784975がデッドロック状態となることがあります。デッドロック状態は、リセット入力によってのみ解除されます。

注 LOCATION 0 命令実行時。LOCATION 0FH 命令実行時はFFF00H-FFFFFFH

(3) スタック・ポインタ (SP) の動作

スタック・アドレッシングでは、1Mバイトすべての空間でアクセス可能ですが、SFR領域および内部ROM領域にスタック・エリアを確保することはできません。

(4) スタック・ポインタ (SP) の初期化

SPIは、 $\overline{\text{RESET}}$ 入力により不定になります。一方、リセット解除直後でもノンマスカブル割り込みは受け付け可能です。したがって、リセット解除直後のSPが不定な状態でノンマスカブル割り込みの要求が発生すると、予期しない動作を行う場合があります。これらの危険を最小限にするために、リセット解除後のプログラムは必ず次のようにしてください。

```
RSTVCT  CSEG AT 0
        DW  RSTSTRT
        }
INITSEG  CSEG BASE
RSTSTRT : LOCATION 0H ; or LOCATION 0FH
        MOVG SP, #STKBGN
```

# 第4章 ポート機能

## 4.1 ポートの機能

μPD784976サブシリーズは、12本の入力ポート、8本の出力ポートと52本の入出力ポートを内蔵しています。図4-1にポートの構成を示します。いずれのポートも1ビット操作、8ビット操作が可能で、きわめて多様な制御が行えます。また、ポートとしての機能のほかに、内蔵ハードウェアの入出力端子としての機能などを持っています。

図4-1 ポートの種類

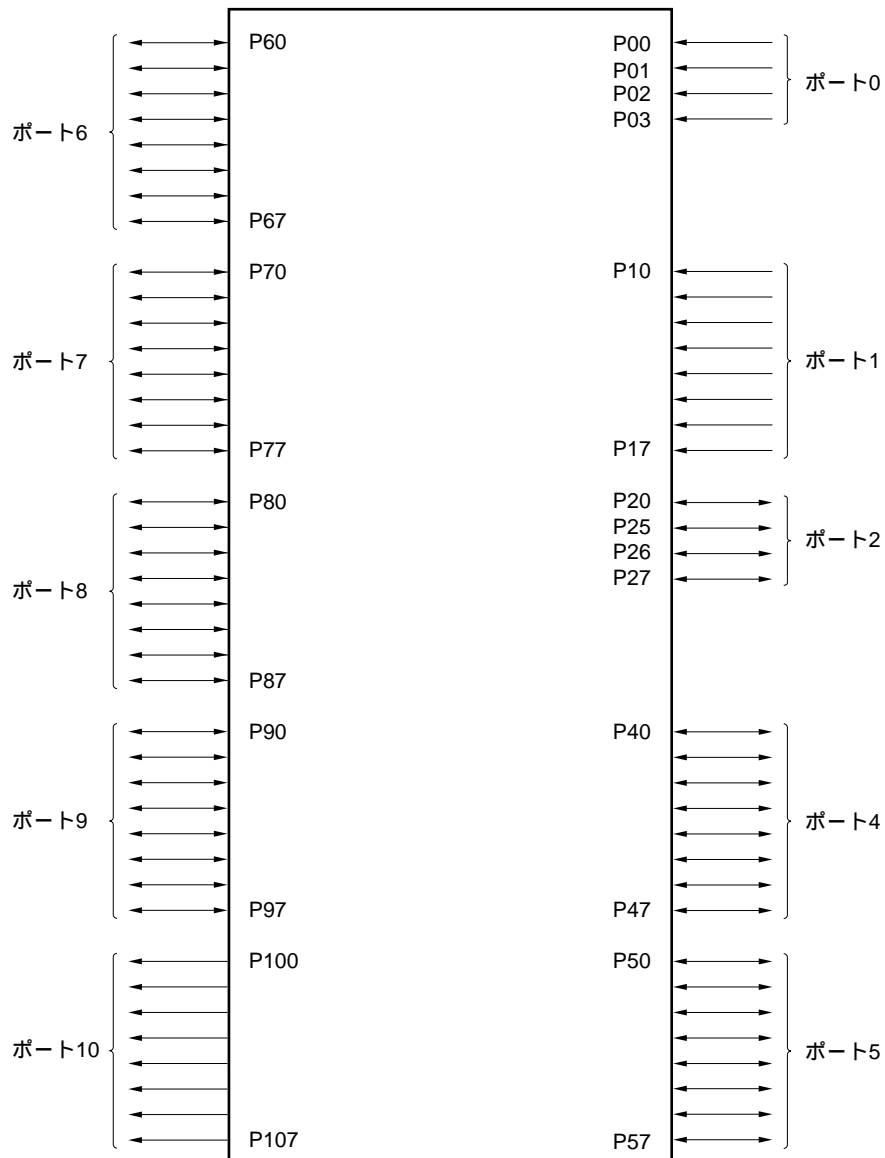


表4 - 1 ポートの機能 (1/2)

端子名称	機 能	兼用端子
P00-P03	ポート0。 4ビット入力専用ポート。	ANI8-ANI11
P10-P17	ポート1。 8ビット入力専用ポート。	ANI0-ANI7
P20	ポート2。	TI00
P25	4ビット入出力ポート。	SI0
P26	1ビット単位で入力/出力の指定可能。	SO0
P27	ソフトウェアにより、1ビット単位または8ビット単位で内蔵プルアップ抵抗が使用可能。	SCK0
P40-P47	ポート4。 8ビット入出力ポート。 1ビット単位で入力/出力の指定可能。 LEDを直接駆動可能。 入力ポートとして使用する場合のみ、ソフトウェアにより、8ビット単位で内蔵プルアップ抵抗が使用可能。	-
P50-P57	ポート5。 N-chオープン・ドレイン8ビット中耐圧入出力ポート。 1ビット単位で入力/出力の指定可能。 LEDを直接駆動可能。 マスクROM製品は、マスク・オプションにより、1ビット単位でプルアップ抵抗の内蔵が可能。μPD78F4976は、プルアップ抵抗を内蔵していません。	-
P60	ポート6。	SI1
P61	8ビット入出力ポート。	SO1
P62	1ビット単位で入力/出力の指定可能。	SCK1
P63	入力ポートとして使用する場合のみ、ソフトウェアにより、8ビット単位で内蔵プルアップ抵抗が使用可能。	TIO50
P64		INTP0
P65		INTP1
P66		TIO51
P67		INTP2
P70-P77	ポート7。 P-chオープン・ドレイン8ビット高耐圧入出力ポート。 1ビット単位で入力/出力の指定可能。 マスクROM製品は、マスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵が可能。μPD78F4976は、プルダウン抵抗を内蔵していません。	FIP16-FIP23

表4 - 1 ポートの機能 (2/2)

端子名称	機 能	兼用端子
P80-P87	<p>ポート 8。</p> <p>P-chオープン・ドレイン 8 ビット高耐圧入出力ポート。</p> <p>1 ビット単位で入力 / 出力の指定可能。</p> <p>マスクROM製品は、マスク・オプションにより、1 ビット単位でプルダウン抵抗の内蔵が可能。μPD78F4976は、プルダウン抵抗を内蔵していません。</p>	FIP24-FIP31
P90-P97	<p>ポート 9。</p> <p>P-chオープン・ドレイン 8 ビット高耐圧入出力ポート。</p> <p>1 ビット単位で入力 / 出力の指定可能。</p> <p>マスクROM製品は、マスク・オプションにより、1 ビット単位でプルダウン抵抗の内蔵が可能。μPD78F4976は、プルダウン抵抗を内蔵していません。</p>	FIP32-FIP39
P100-P107	<p>ポート 10。</p> <p>P-chオープン・ドレイン 8 ビット高耐圧出力ポート。</p> <p>マスクROM製品は、マスク・オプションにより、1 ビット単位でプルダウン抵抗の内蔵が可能。μPD78F4976は、プルダウン抵抗を内蔵していません。</p>	FIP40-FIP47

## 4.2 ポートの構成

ポートは、次のハードウェアで構成しています。

表4 - 2 ポートの構成

項目	構成
制御レジスタ	ポート・モード・レジスタ (PMm : m = 2, 4-6) プルアップ抵抗オプション・レジスタ (PUO, PU2)
ポート	合計 : 72本 (入力 : 12本, 出力 : 8本, 入出力 : 52本)
プルアップ抵抗	・マスクROM製品 合計 : 28本 (ソフトウェア制御 : 20本, マスク・オプション制御 : 8本) ・μPD78F4976 合計 : 20本
プルダウン抵抗	・マスクROM製品 合計 : 32本 (マスク・オプション制御 : 32本) ・μPD78F4976 なし

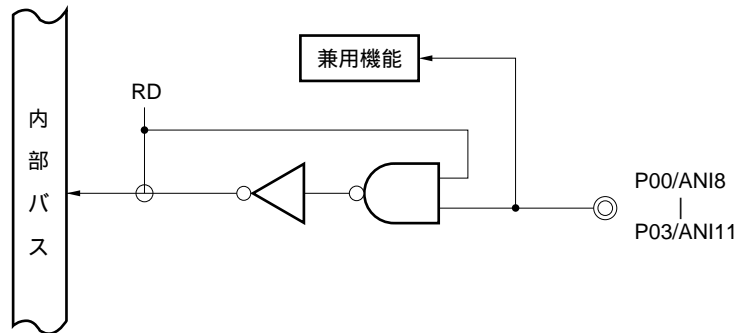
### 4.2.1 ポート0

4ビット入力専用ポートです。

また、兼用機能としてA/Dコンバータのアナログ入力があります。

図4 - 2にポート0のブロック図を示します。

図4 - 2 P00-P03のブロック図



RD : ポート0のリード信号

**注意** ポート0はA/Dコンバータのアナログ入力と兼用になっているため、ポート0をアナログ入力端子として使用しているときは、ポートの読み出し命令を行わないでください(ビット操作を含む)。

ポートの読み出し操作の際、アナログ入力端子に中間電位が入力されていると、中間電圧を読み出すことになるため、チップの信頼性を損ねる可能性があります。

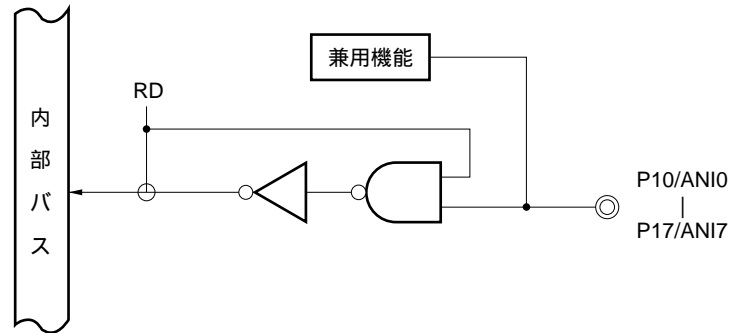
### 4.2.2 ポート1

8ビット入力専用ポートです。

また、兼用機能としてA/Dコンバータのアナログ入力があります。

図4-3にポート1のブロック図を示します。

図4-3 P10-P17のブロック図



RD : ポート1のリード信号

**注意** ポート1はA/Dコンバータのアナログ入力と兼用になっているため、ポート1をアナログ入力端子として使用しているときは、ポートの読み出し命令を行わないでください（ビット操作命令を含む）。

ポートの読み出し操作の際、アナログ入力端子に中間電位が入力されていると、中間電圧を読み出すことになるため、チップの信頼性を損ねる可能性があります。

### 4.2.3 ポート2

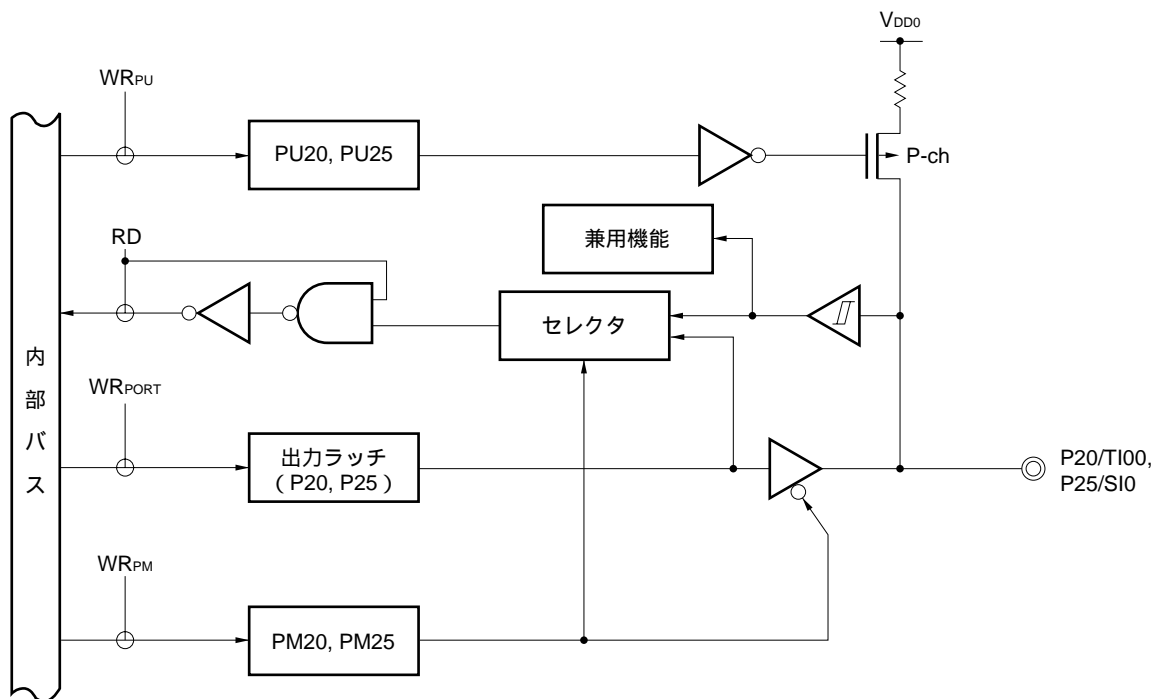
出力ラッチ付き4ビット入出力ポートです。P20, P25-P27端子は、ポート2モード・レジスタ (PM2) により、1ビット単位で入力モード/出力モードの指定ができます。P20, P25-P27端子は、プルアップ抵抗オプション・レジスタ2 (PU2) により、1ビット単位または8ビット単位で内蔵プルアップ抵抗を使用できます。

また、兼用機能としてシリアル・インタフェースのデータ入出力、シリアル・クロック入出力、タイマの入力があります。

$\overline{\text{RESET}}$ 入力により、入力モードになります。

図4-4~図4-6にポート2のブロック図を示します。

図4-4 P20, P25のブロック図



PU2 : プルアップ抵抗オプション・レジスタ2

PM2 : ポート2モード・レジスタ

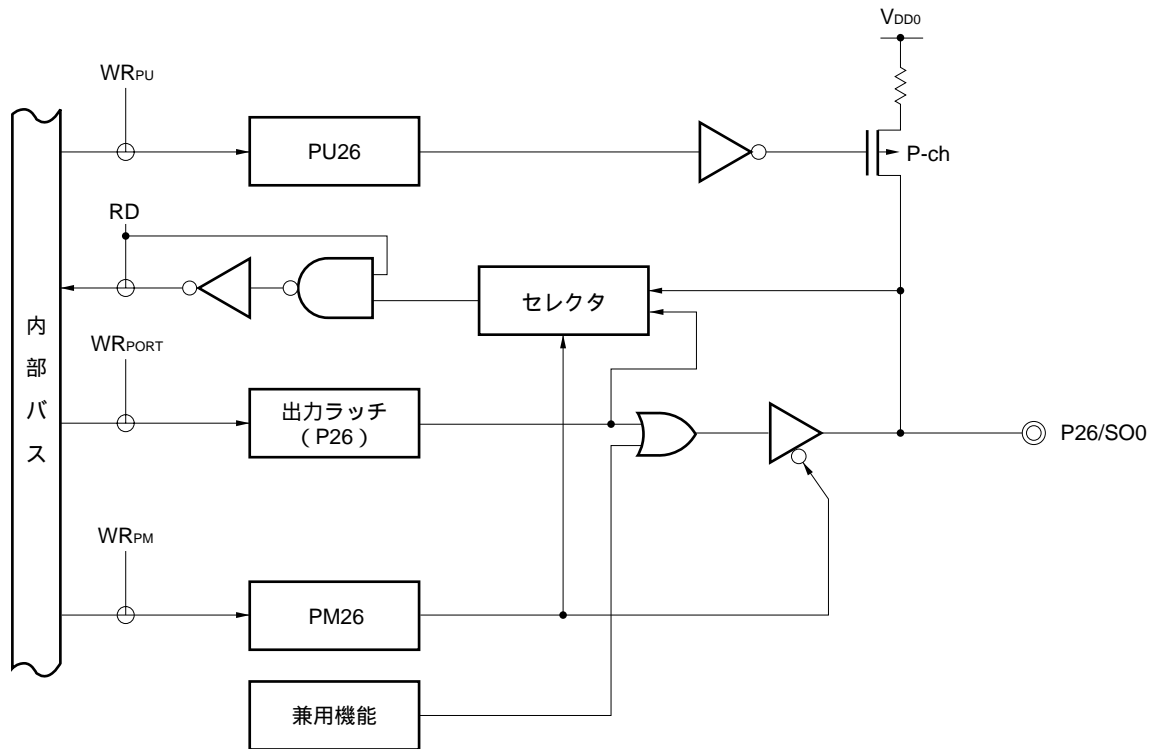
RD : ポート2のリード信号

WR : ポート2のライト信号

**注意** ポート2で出力モードに指定した端子に対しては、プルアップ抵抗を接続しないでください。



図4 - 5 P26のブロック図



PU2 : プルアップ抵抗オプション・レジスタ 2

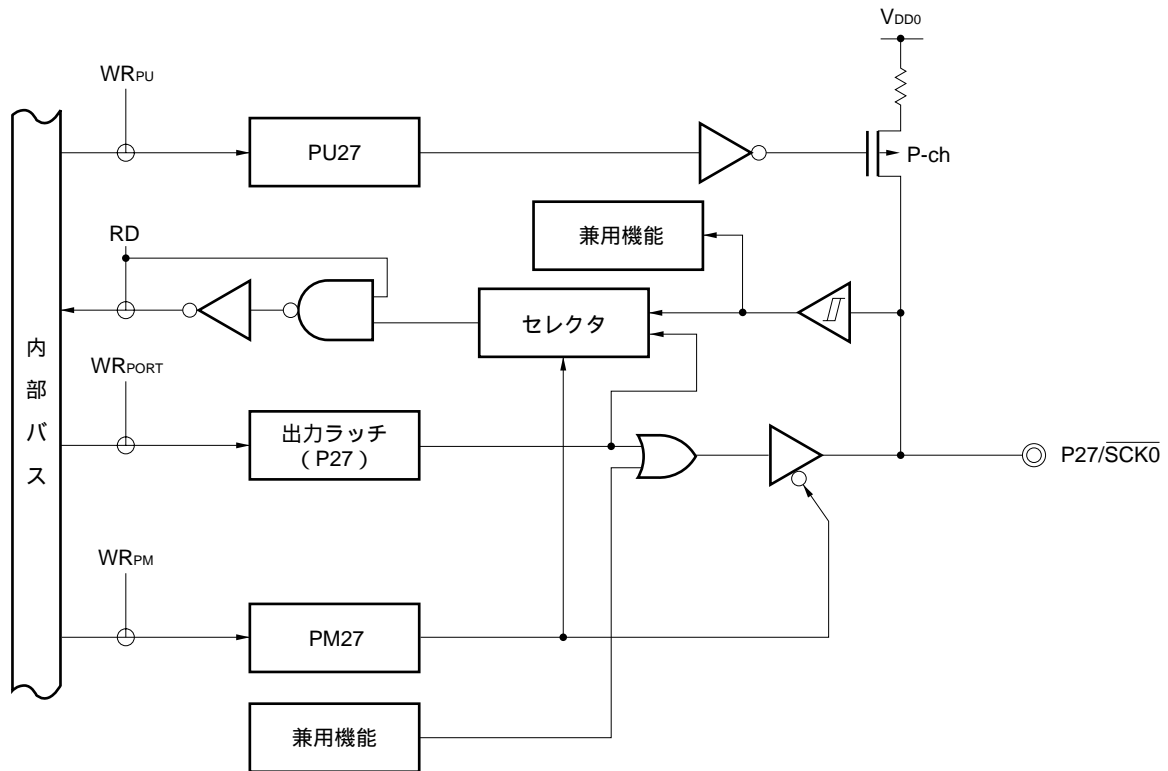
PM2 : ポート 2 モード・レジスタ

RD : ポート 2 のリード信号

WR : ポート 2 のライト信号

**注意** ポート 2 で出力モードに指定した端子に対しては、プルアップ抵抗を接続しないでください。

図4-6 P27のブロック図



PU2 : プルアップ抵抗オプション・レジスタ2

PM2 : ポート2モード・レジスタ

RD : ポート2のリード信号

WR : ポート2のライト信号

**注意** ポート2で出力モードに指定した端子に対しては、プルアップ抵抗を接続しないでください。

#### 4.2.4 ポート4

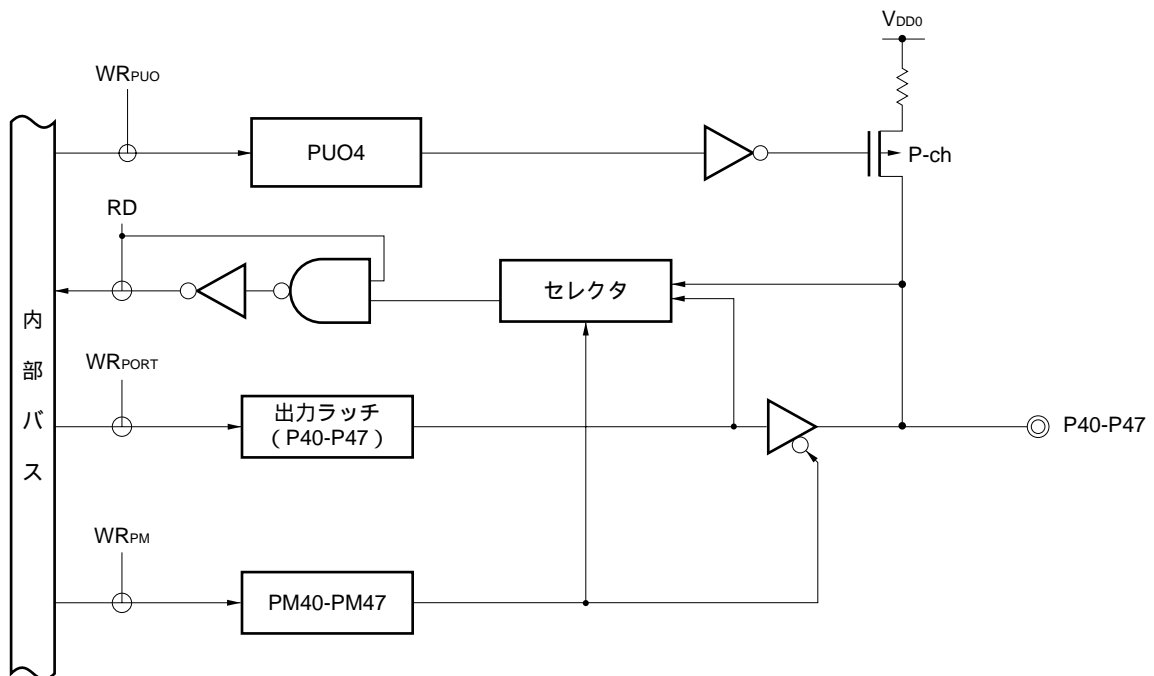
出力ラッチ付き8ビット入出力ポートです。P40-P47端子は、ポート4モード・レジスタ（PM4）により、1ビット単位で入力モード/出力モードの指定ができます。P40-P47端子は入力ポートとして使用する時のみ、プルアップ抵抗オプション・レジスタ（PUO）のビット4（PUO4）により、ポート単位で内蔵プルアップ抵抗を使用できます。

ポート4はLEDを直接駆動可能です。

$\overline{\text{RESET}}$ 入力により、入力モードになります。

図4-7にポート4のブロック図を示します。

図4-7 P40-P47のブロック図



PUO : プルアップ抵抗オプション・レジスタ

PM4 : ポート4モード・レジスタ

RD : ポート4のリード信号

WR : ポート4のライト信号

### 4.2.5 ポート5

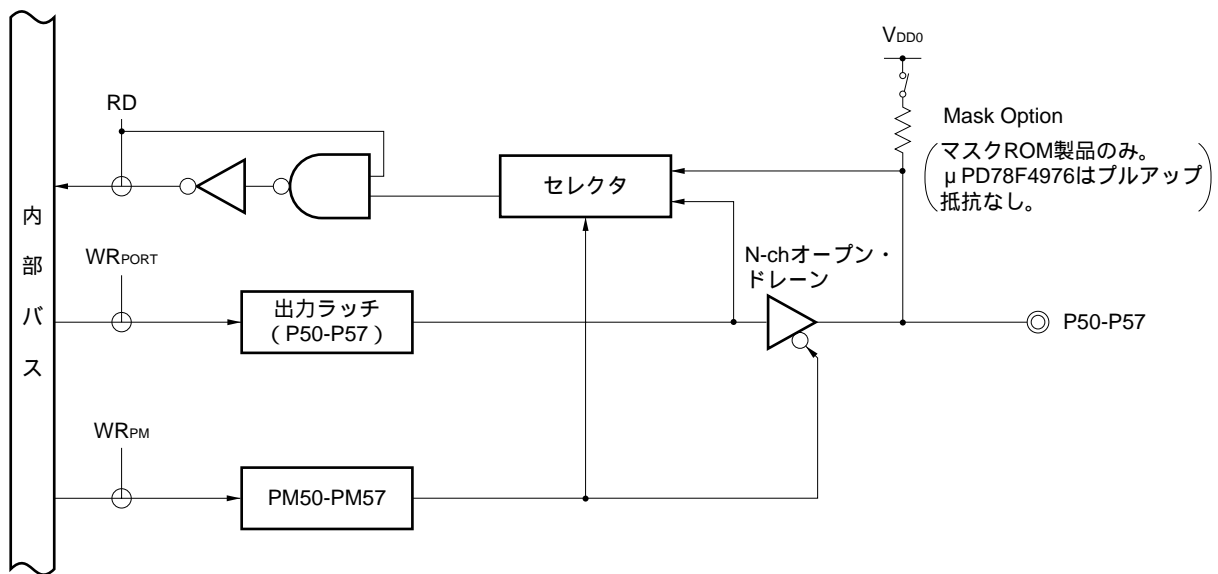
出力ラッチ付き 8 ビット入出力ポートです。P50-P57端子は、ポート 5 モード・レジスタ (PM5) により、1 ビット単位で入力モード/出力モードの指定ができます。マスクROM製品は、マスク・オプションにより、1 ビット単位でプルアップ抵抗の内蔵ができます。μPD78F4976は、プルアップ抵抗を内蔵していません。

ポート 5 はLEDを直接駆動可能です。

$\overline{\text{RESET}}$ 入力により、入力モードになります。

図 4 - 8 にポート 5 のブロック図を示します。

図 4 - 8 P50-P57のブロック図



PM5 : ポート 5 モード・レジスタ

RD : ポート 5 のリード信号

WR : ポート 5 のライト信号

### 4.2.6 ポート6

出力ラッチ付き 8 ビット入出力ポートです。P60-P67端子は、ポート 6 モード・レジスタ (PM6) により、1 ビット単位で入力モード / 出力モードの指定ができます。P60-P67端子は入力ポートとして使用する  
 ときのみ、プルアップ抵抗オプション・レジスタ (PUO) のビット 6 (PUO6) により、ポート単位で内蔵  
 プルアップ抵抗を使用できます。

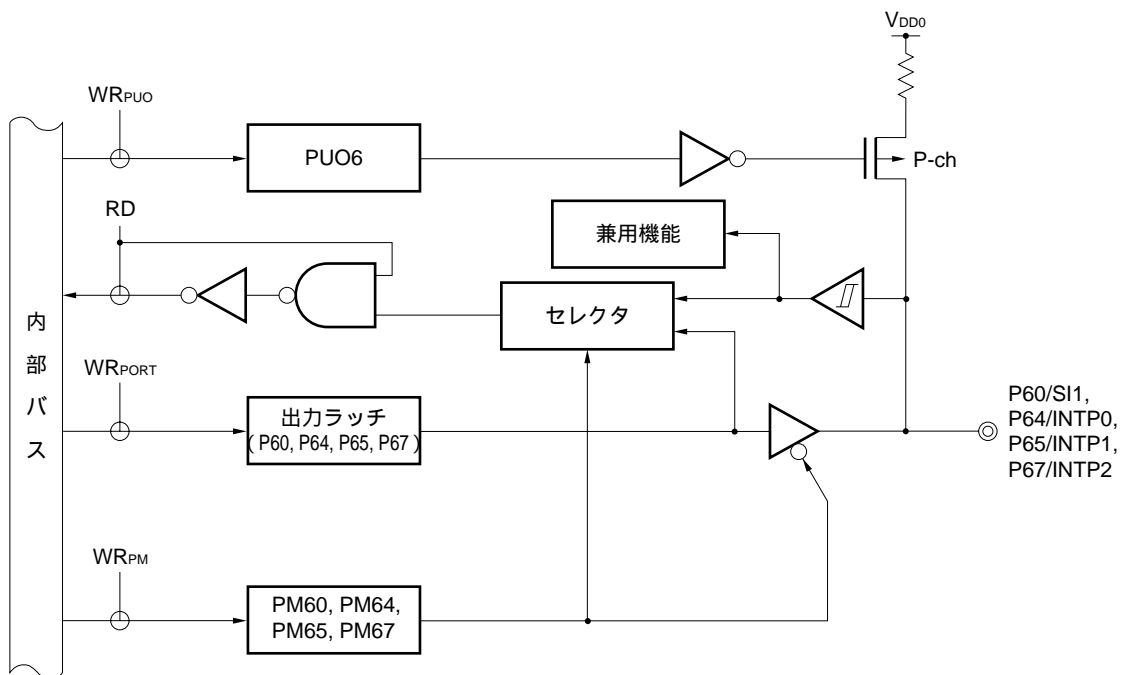
また、兼用機能としてシリアル・インタフェースのデータ入出力、シリアル・クロック入出力、タイマの  
 入出力、外部割り込み要求入力があります。

$\overline{\text{RESET}}$ 入力により、入力モードになります。

図 4 - 9 , 図 4 - 10 , 図 4 - 11にポート 6 のブロック図を示します。

**注意** P64, P65, P67は外部割り込み要求入力と兼用となっていますが、割り込み入力端子として使  
 用しない場合は、外部割り込み立ち上がりエッジ許可レジスタ (EGP0) , 外部割り込み立ち  
 下がりエッジ許可レジスタ (EGN0) で「割り込み禁止」を設定するか、割り込みマスク・フ  
 ラグ (PMKn : n = 0-2) に 1 を設定してください。そうしないとポート機能を出力モードに指  
 定し出力レベルを変化させたとき、割り込み要求フラグがセットされ、不用意な割り込み処理  
 が実行されることがあります。

図 4 - 9 P60, P64, P65, P67のブロック図



PUO : プルアップ抵抗オプション・レジスタ

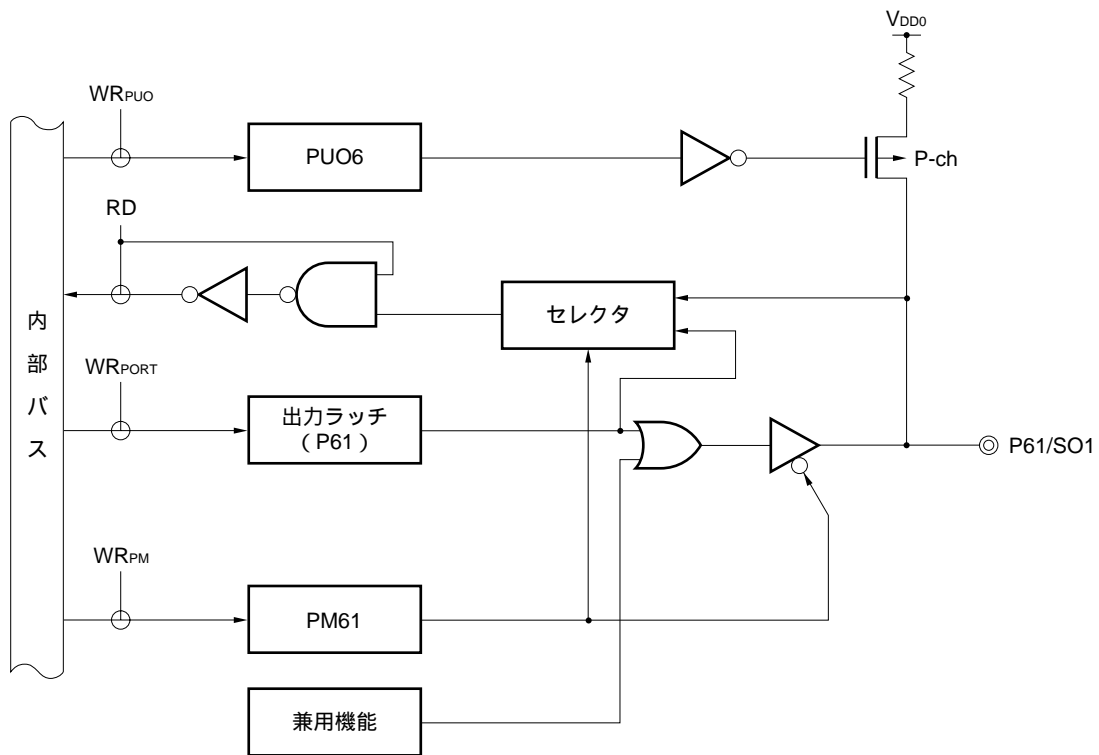
PM6n : ポート 6 モード・レジスタ

RD : ポート 6 のリード信号

WR : ポート 6 のライト信号

**備考** n = 0, 4, 5, 7

図4 - 10 P61のブロック図



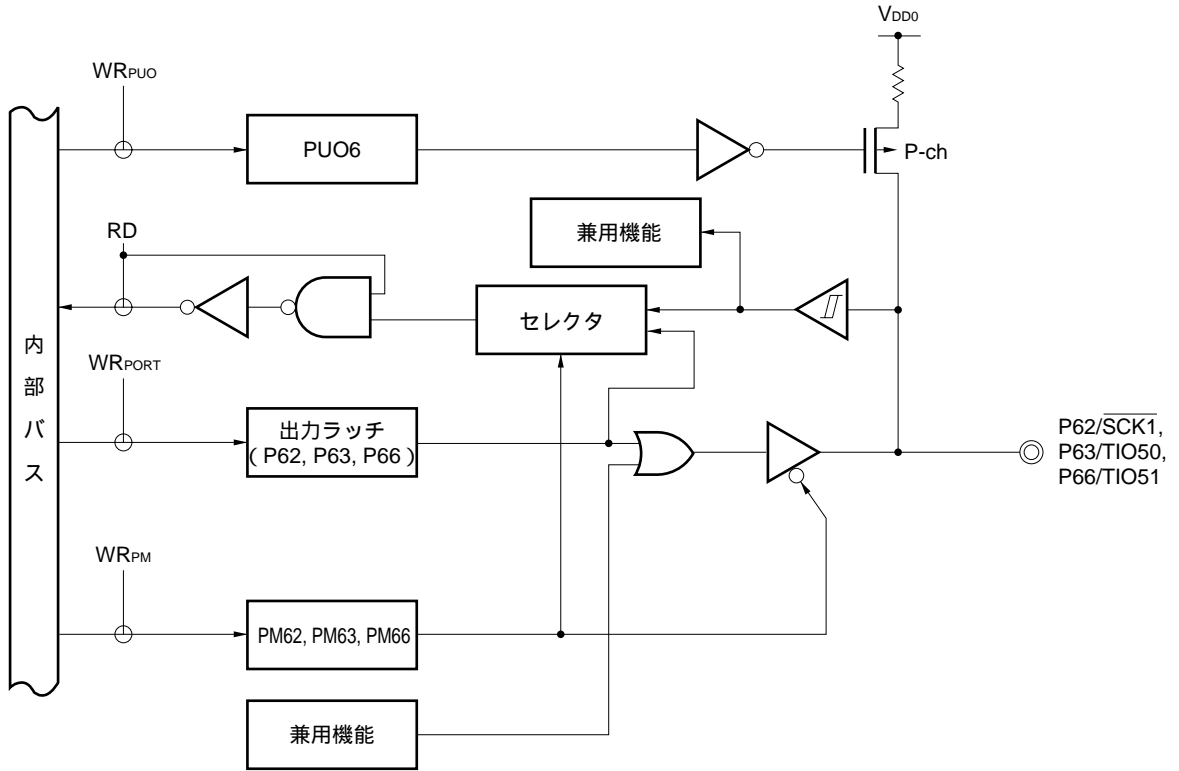
PUO : プルアップ抵抗オプション・レジスタ

PM61 : ポート6モード・レジスタ

RD : ポート6のリード信号

WR : ポート6のライト信号

図4 - 11 P62, P63, P66のブロック図



PUO : プルアップ抵抗オプション・レジスタ

PM6 : ポート6モード・レジスタ

RD : ポート6のリード信号

WR : ポート6のライト信号

### 4.2.7 ポート7

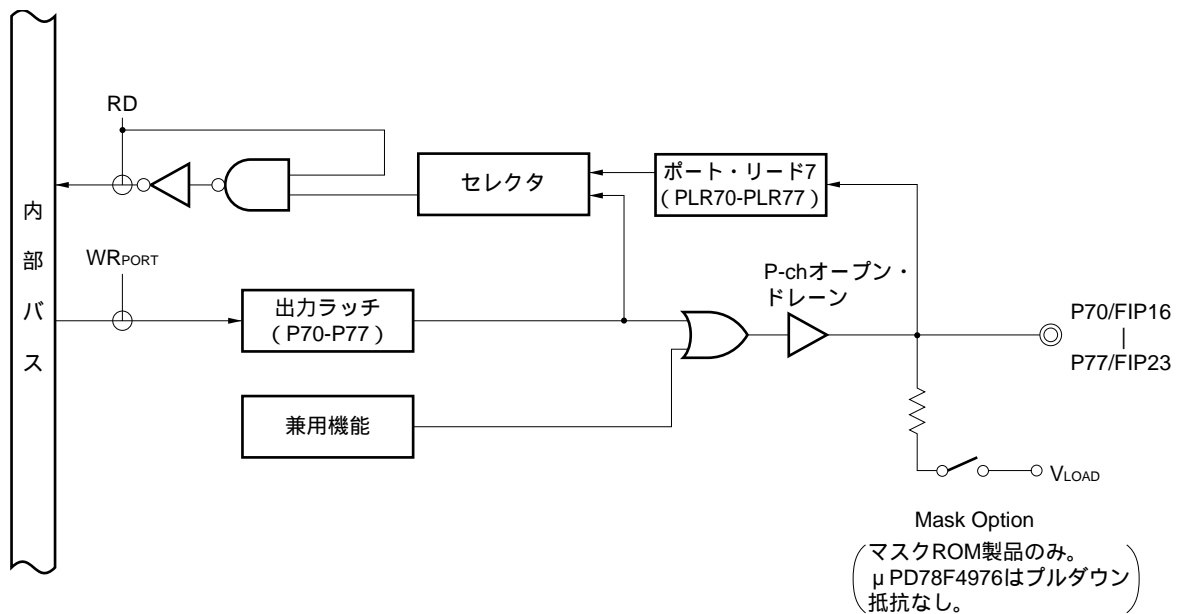
出力ラッチ付き8ビット入出力ポートです。出力ポートとして使用する場合には、出力ラッチ（P70-P77）に設定した値が出力されます。また、入力ポートとして使用する場合には、出力ラッチ（P70-P77）に“0”を設定し、ポート・リード7（PLR70-PLR77）をリードしてください。また、出力ラッチ（P70-P77）に“1”を設定すると、出力ラッチそのものの値がリードされます。マスクROM製品は、マスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵ができます。μPD78F4976は、プルダウン抵抗を内蔵していません。

また、兼用機能としてFIPコントローラ/ドライバの出力があります。

$\overline{\text{RESET}}$ 入力により、入力モードになります。

図4-12にポート7のブロック図を示します。

図4-12 P70-P77のブロック図



RD : ポート7のリード信号

WR : ポート7のライト信号



### 4.2.8 ポート8

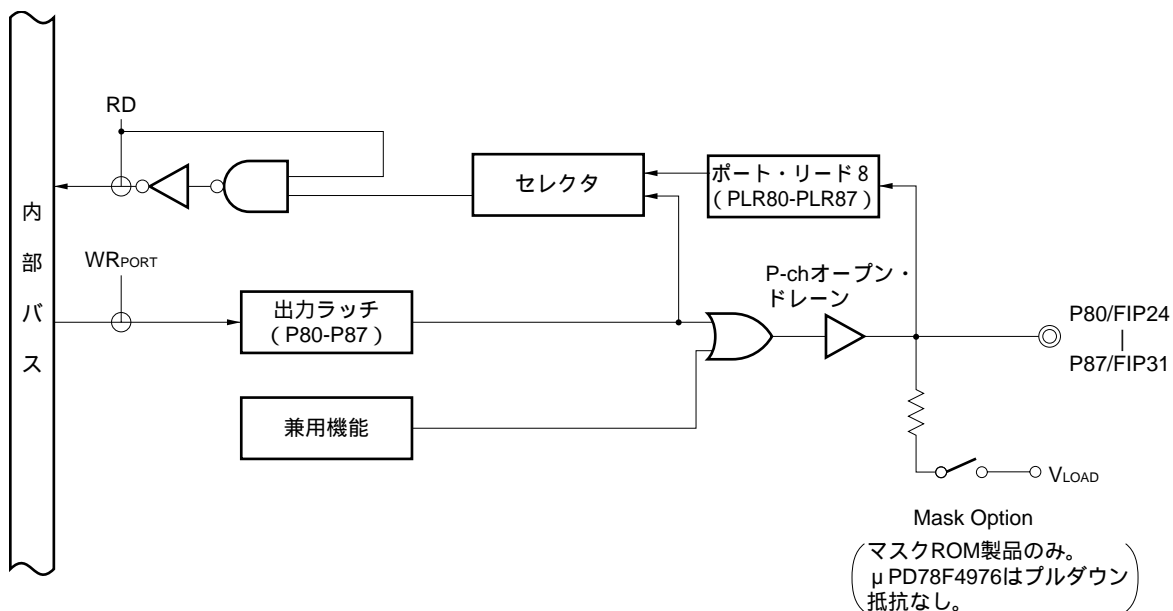
出力ラッチ付き8ビット入出力ポートです。出力ポートとして使用する場合には、出力ラッチ（P80-P87）に設定した値が出力されます。また、入力ポートとして使用する場合には、出力ラッチ（P80-P87）に“0”を設定し、ポート・リード8（PLR80-PLR87）をリードしてください。また、出力ラッチ（P80-P87）に“1”を設定すると、出力ラッチそのものの値がリードされます。マスクROM製品は、マスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵ができます。μPD78F4976は、プルダウン抵抗を内蔵していません。

また、兼用機能としてFIPコントローラ/ドライバの出力があります。

$\overline{\text{RESET}}$ 入力により、入力モードになります。

図4-13にポート8のブロック図を示します。

図4-13 P80-P87のブロック図



RD : ポート8のリード信号

WR : ポート8のライト信号

### 4.2.9 ポート9

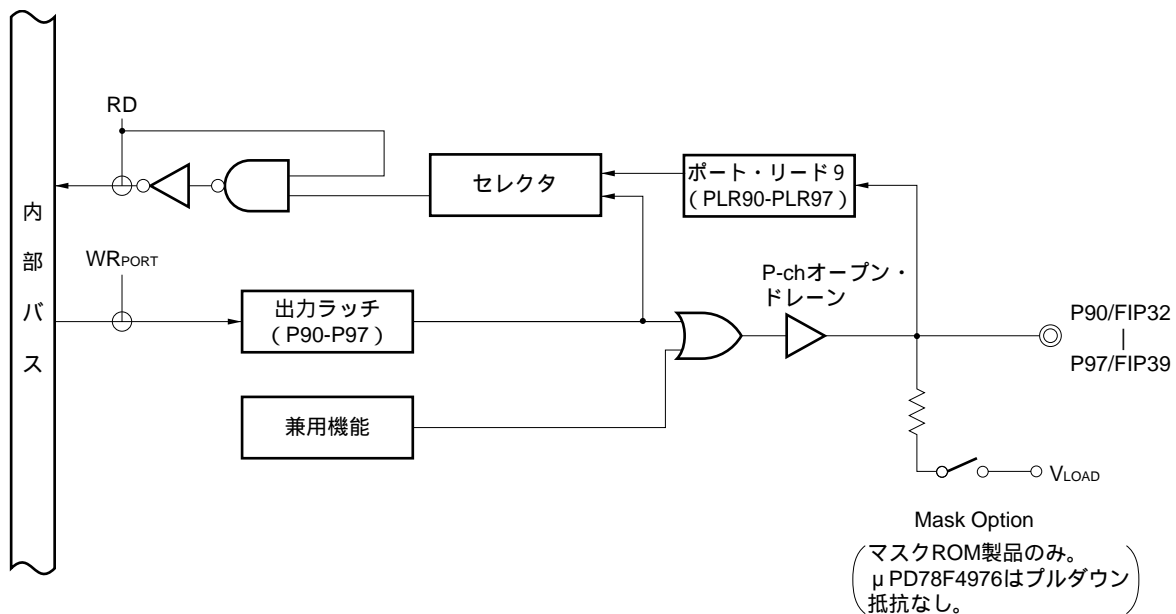
出力ラッチ付き8ビット入出力ポートです。出力ポートとして使用する場合には、出力ラッチ（P90-P97）に設定した値が出力されます。また、入力ポートとして使用する場合には、出力ラッチ（P90-P97）に“0”を設定し、ポート・リード9（PLR90-PLR97）をリードしてください。また、出力ラッチ（P90-P97）に“1”を設定すると、出力ラッチそのものの値がリードされます。マスクROM製品は、マスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵ができます。μPD78F4976は、プルダウン抵抗を内蔵していません。

また、兼用機能としてFIPコントローラ/ドライバの出力があります。

$\overline{\text{RESET}}$ 入力により、入力モードになります。

図4-14にポート9のブロック図を示します。

図4-14 P90-P97のブロック図



RD : ポート9のリード信号

WR : ポート9のライト信号

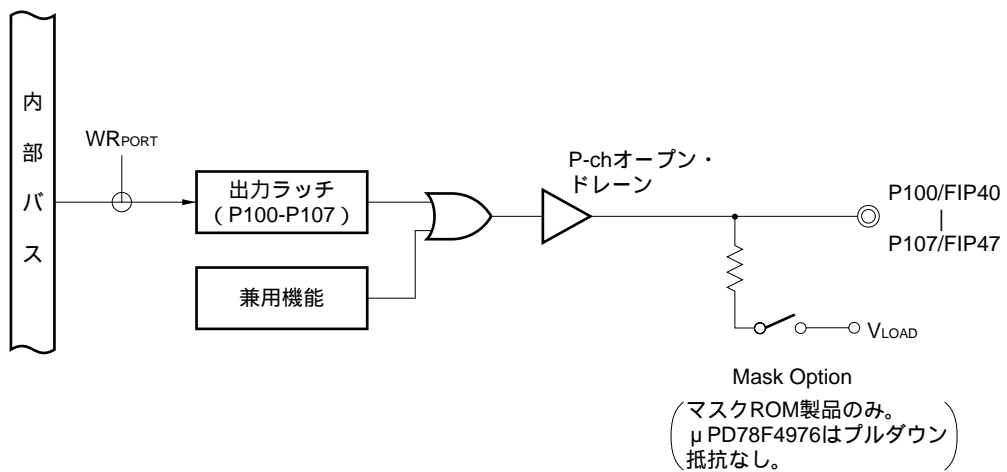
### 4.2.10 ポート10

8ビット出力専用ポートです。マスクROM製品は、マスク・オプションにより、1ビット単位でプルダウン抵抗の内蔵ができます。μPD78F4976は、プルダウン抵抗を内蔵していません。

また、兼用機能としてFIPコントローラ/ドライバの出力があります。

図4 - 15にポート10のブロック図を示します。

図4 - 15 P100-P107のブロック図



WR : ポート10のライト信号

### 4.3 ポート機能を制御するレジスタ

ポートを制御するレジスタには、次の2種類があります。

- ・ポート・モード・レジスタ (PM2, PM4-PM6) ・プルアップ抵抗オプション・レジスタ (PUO, PU2)

#### (1) ポート・モード・レジスタ (PM2, PM4-PM6)

ポートの入力/出力を1ビット単位で設定するレジスタです。

PM2, PM4-PM6は、それぞれ1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、FFHになります。

ポート端子を兼用機能の端子として使用する場合、ポート・モード・レジスタ、出力ラッチを表4-3のように設定してください。

**注意1** . P00-P03, P10-P17は入力専用端子です。

2 . P100-P107は出力専用端子です。

3 . P64, P65, P67は外部割り込み要求入力と兼用となっていますが、割り込み入力端子として使用しない場合は、外部割り込み立ち上がりエッジ許可レジスタ (EGP0) , 外部割り込み立ち下がりエッジ許可レジスタ (EGN0) で「割り込み禁止」を設定するか、割り込みマスク・フラグ (PMKn : n=0-2) に1を設定してください。そうしないとポート機能を出力モードに指定し出力レベルを変化させたとき、割り込み要求フラグがセットされ、不用意な割り込み処理が実行されることがあります。

表4-3 兼用機能使用時のポート・モード・レジスタ、出力ラッチの設定

端子名称	兼用機能		PM x x	P x x	端子名称	兼用機能		PM x x	P x x
	名称	入出力				名称	入出力		
P20	TI00	入力	1	x	P62	$\overline{\text{SCK1}}$	入出力	1/0	x/0
P25	SI0	入力	1	x	P63	TIO50	入出力	1/0	x/0
P26	SO0	出力	0	0	P64	INTP0	入力	1	x
P27	$\overline{\text{SCK0}}$	入出力	1/0	x/0	P65	INTP1	入力	1	x
P60	SI1	入力	1	x	P66	TIO51	入出力	1/0	x/0
P61	SO1	出力	0	0	P67	INTP2	入力	1	x

**注意1** . シリアル動作モード・レジスタn (CSIMn) のビット1, 0 (SCLn1, SCLn0) で選択するクロックにより、PM27, PM62の設定は異なります。

内部クロック (SCLn1, SCLn0 = 0, 0) のとき : 0

外部クロック (SCLn1, SCLn0 = 0, 0) のとき : 1

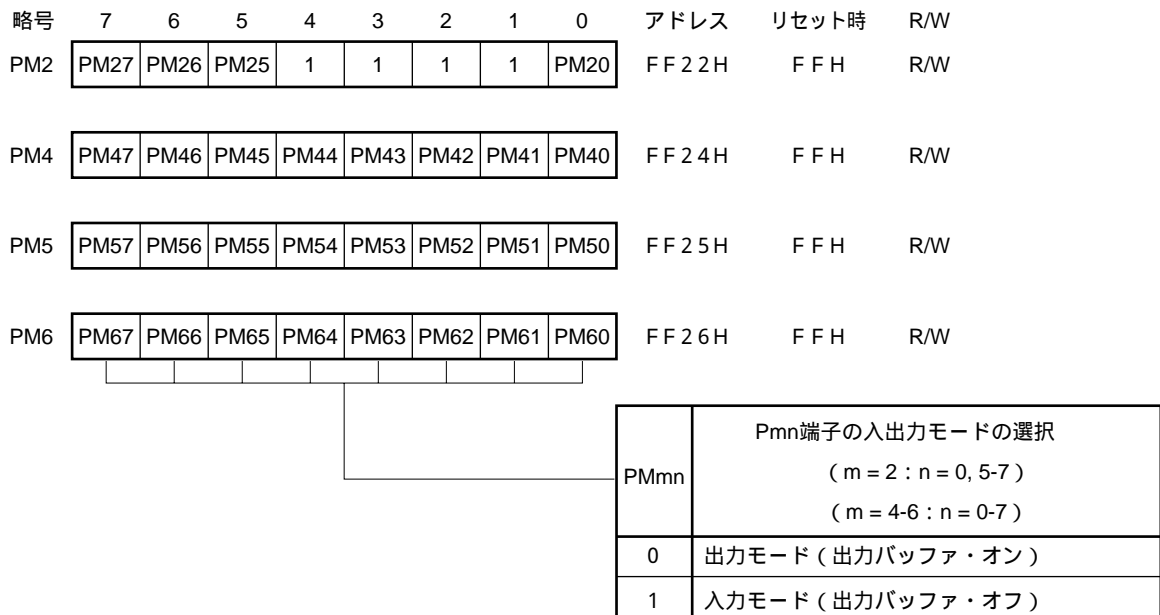
2 . 兼用機能の端子として使用しているときに、これらのポートに対して読み出し命令を実行した場合、読み出したデータは不定になります。

**備考** x : don't care (設定の必要はありません)

PM x x : ポート・モード・レジスタ

P x x : ポートの出力ラッチ

図4 - 16 ポート・モード・レジスタのフォーマット



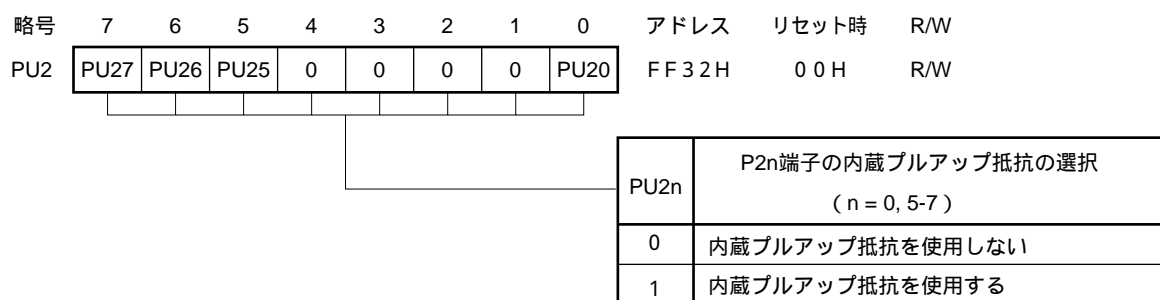
(2) プルアップ抵抗オプション・レジスタ2 (PU2)

ポート2の各端子の内蔵プルアップ抵抗を使用するか、使用しないかを1ビット単位または8ビット単位で設定するレジスタです。PU2で内蔵プルアップ抵抗の使用を指定したビットで、入力モード/出力モードの指定にかかわらず、内部でプルアップ抵抗が使用できます。ポート2で出力モードに指定した端子に対しては、プルアップ抵抗を接続しないでください。

PU2は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図4 - 17 プルアップ抵抗オプション・レジスタ2 (PU2) のフォーマット



(3) プルアップ抵抗オプション・レジスタ (PUO)

ポート4, 6のプルアップ抵抗を使用するか, 使用しないかを設定するレジスタです。

PUOは, ポート4, 6をポートごとに内蔵プルアップ抵抗の接続を指定できます。

PUOは, 1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

RESET入力により, 00Hになります。

図4 - 18 プルアップ抵抗オプション・レジスタ (PUO) のフォーマット

アドレス : 0FF4EH リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
PUO	0	PUO6	0	PUO4	0	0	0	0

PUOn	Pn端子の内蔵プルアップ抵抗の選択 (n = 4, 6)
0	内蔵プルアップ抵抗を使用しない
1	内蔵プルアップ抵抗を使用する

**注意** ポート4, 6のうち出力モードに指定した端子に対しては, PUOn = 1に設定してもプルアップ抵抗は使用できません。入力モードに指定した端子に対してのみ, プルアップ抵抗は使用できます。

## 4.4 ポート機能の動作

ポートの動作は、次に示すように入出力モードの設定によって異なります。

### 4.4.1 入出力ポートへの書き込み

#### (1) 出力ポートの場合

転送命令により、出力ラッチに値を書き込みます。また、出力ラッチの内容が端子より出力されません。

一度出力ラッチに書き込まれたデータは、もう一度出力ラッチにデータを書き込むまで保持されません。

#### (2) 入力ポートの場合

転送命令により、出力ラッチに値を書き込みます。しかし、出力バッファがオフしていますので、端子の状態は変化しません。

一度出力ラッチに書き込まれたデータは、もう一度出力ラッチにデータを書き込むまで保持されません。

**注意** 1ビット・メモリ操作命令の場合、操作対象は1ビットですが、ポートを8ビット単位でアクセスします。したがって、入力/出力が混在しているポートでは、操作対象のビット以外でも入力に指定されている端子の出力ラッチの内容が不定になります。

### 4.4.2 入出力ポートからの読み出し

#### (1) 出力ポートの場合

転送命令により、出力ラッチの内容が読み出せます。出力ラッチの内容は変化しません。

#### (2) 入力ポートの場合

転送命令により、端子の状態が読み出せます。出力ラッチの内容は変化しません。

### 4.4.3 入出力ポートでの演算

#### (1) 出力ポートの場合

出力ラッチの内容と演算を行い、結果を出力ラッチに書き込みます。また、出力ラッチの内容が端子より出力されます。

一度出力ラッチに書き込まれたデータは、もう一度出力ラッチにデータを書き込むまで保持されません。

#### (2) 入力ポートの場合

出力ラッチの内容が不定になります。しかし、出力バッファがオフしていますので、端子の状態は変化しません。

**注意** 1ビット・メモリ操作命令の場合、操作対象は1ビットですが、ポートを8ビット単位でアクセスします。したがって、入力/出力が混在しているポートでは、操作対象のビット以外でも入力が指定されている端子の出力ラッチの内容が不定になります。

## 4.5 マスク・オプションの選択

マスクROM製品には、次のマスク・オプションがあります。μPD78F4976には、マスク・オプションはありません。

表4-4 マスクROM製品のマスク・オプションとμPD78F4976との比較

端子名	マスクROM製品のマスク・オプション	μPD78F4976
P50-P57	1ビット単位でプルアップ抵抗を内蔵できます。	プルアップ抵抗を内蔵していません。
P70/FIP16-P77/FIP23, P80/FIP24-P87/FIP31, P90/FIP32-P97/FIP39, P100/FIP40-P107/FIP47	1ビット単位でプルダウン抵抗を内蔵できます。	プルダウン抵抗を内蔵していません。



# 第5章 クロック発生回路

## 5.1 クロック発生回路の機能

クロック発生回路は、CPUおよび周辺ハードウェアに供給するクロックを発生する回路です。  
システム・クロック発振回路には、次の1種類があります。

- ・メイン・システム・クロック発振回路  
4 ~ 12.5 MHzの周波数を発振します。スタンバイ・コントロール・レジスタ (STBC) によるSTOPモードへの設定 (STPビット=1) およびRESET入力で、発振を停止できます。

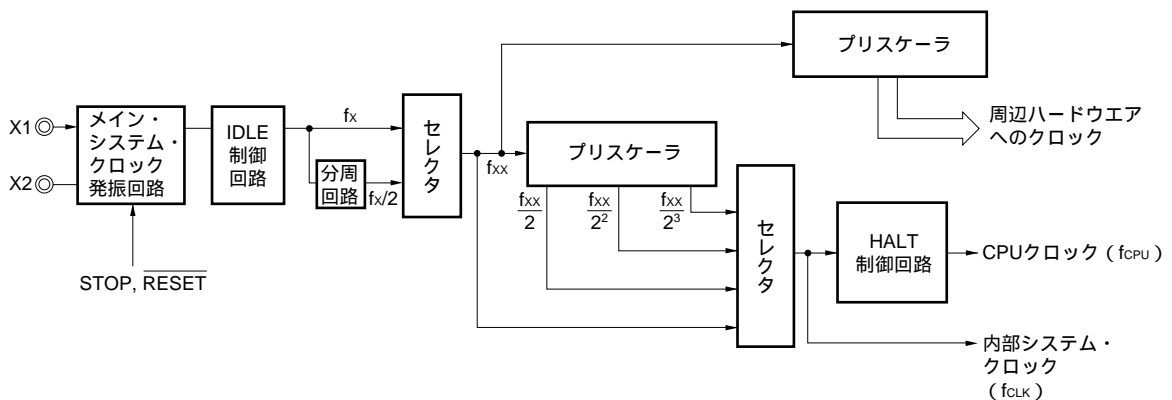
## 5.2 クロック発生回路の構成

クロック発生回路は、次のハードウェアで構成しています。

表5 - 1 クロック発生回路の構成

項 目	構 成
制御レジスタ	スタンバイ・コントロール・レジスタ (STBC) 発振モード選択レジスタ (CC) 発振安定時間指定レジスタ (OSTS)
発振回路	メイン・システム・クロック発振回路

図5 - 1 クロック発生回路のブロック図



- 備考1 . fx : メイン・システム・クロック発振周波数  
2 . fxx : メイン・システム・クロック周波数

## 5.3 制御レジスタ

### (1) スタンバイ・コントロール・レジスタ (STBC)

STBCは、スタンバイ・モードの設定と、内部システム・クロックを選択するためのレジスタです。スタンバイ・モードの詳細については、第15章 **スタンバイ機能**を参照してください。

プログラムの暴走によって誤ってスタンバイ・モードに入らないように、専用命令によってのみ書き込みができます。この専用命令は、MOV STBC, #byte命令で特殊なコード構成(4バイト)になっています。3バイト目と4バイト目のオペコードが互いに1の補数の場合のみ、書き込みが行われます。3バイト目と4バイト目のオペコードが互いに1の補数になっていない場合は、書き込みが行われず、オペランド・エラー割り込みが発生します。この場合、スタック領域に退避されるリターン・アドレスは、エラーの原因となった命令のアドレスです。したがって、スタック領域に退避されたリターン・アドレスにより、エラーの原因となったアドレスを知ることができます。

なお、単純にオペランド・エラーからRETB命令で復帰を行うと、無限ループとなります。

オペランド・エラー割り込みは、プログラムが暴走している場合にのみ発生する(NEC製アセンブラRA78K4では、MOV STBC, #byteと記述された場合、正しい専用命令しか生成しません)ので、オペランド・エラー割り込みの処理プログラムではシステムの初期化を行うようにしてください。

他の書き込み命令(MOV STBC, AやAND STBC, #byteやSET1 STBC.7など)は無視され、何の動作も行いません。すなわち、STBCへの書き込みは行われず、また、オペランド・エラー割り込みのような割り込みも発生しません。STBCの読み出しは、データ転送命令によりいつでもできます。

$\overline{\text{RESET}}$ 入力により、30H になります。

図5-2に、STBCのフォーマットを示します。

スタンバイ・コントロール・レジスタ (STBC) は、CPUクロックの選択、分周比、モード (通常動作 / HALT / IDLE / STOP) を設定するレジスタです。

STBCは、8ビット・メモリ操作命令で設定します。

RESET入力により、30Hになります。

図5 - 2 スタンバイ・コントロール・レジスタ (STBC) のフォーマット

アドレス : 0FFC0H リセット時 : 30H R/W

略号	7	6	5	4	3	2	1	0
STBC	0	0	CK1	CK0	0	0	STP	HLT

CK1	CK0	CPUクロックの選択注 (スルーレート・クロック・モード時、発振分周モード時)
0	0	$f_{xx}$ ( $f_x, f_x/2$ )
0	1	$f_{xx}/2$ ( $f_x/2, f_x/2^2$ )
1	0	$f_{xx}/2^2$ ( $f_x/2^2, f_x/2^3$ )
1	1	$f_{xx}/2^3$ ( $f_x/2^3, f_x/2^4$ )

STP	HLT	動作指定フラグ
0	0	通常動作モード
0	1	HALTモード (HALTモードが解除されると自動的にクリア)
1	0	STOPモード (STOPモードが解除されると自動的にクリア)
1	1	IDLEモード (IDLEモードが解除されると自動的にクリア)

注意1 . 外部クロック入力時にSTOPモードを使用する場合は、発振安定時間指定レジスタ (OSTS) のEXTCビットを必ずセット (1) してからSTOPモードを設定してください。外部クロック入力時にOSTSのEXTCビットがクリア (0) されている状態でSTOPモードを使用すると必ず  $\mu$ PD784975の破壊または信頼性の低下をまねく可能性があります。なお、OSTSのEXTCビットをセット (1) する場合には、必ずX2端子に、X1端子に入力しているクロックの逆相のクロックを入力してください。

2 . スタンバイ命令のあと (スタンバイ解除後) は、NOP命令を3回実行してください。そうしないと、スタンバイ命令の実行と割り込み要求が競合した場合に、スタンバイ命令を実行せず、スタンバイ命令に続く複数の命令を実行後に割り込み要求を受け付けます。割り込み受け付け前に実行する命令は、スタンバイ命令実行後の最大6クロック以内に実行を開始する命令です。

注 CPUクロックの選択は、発振モード選択レジスタ (CC) でも設定できます。

```

例 MOV STBC, #byte
    NOP
    NOP
    NOP
    ⋮
    
```

**備考** f<sub>xx</sub> : メイン・システム・クロック周波数( f<sub>x</sub>またはf<sub>x</sub>/2 )

f<sub>x</sub> : メイン・システム・クロック発振周波数

( 2 ) 発振モード選択レジスタ ( CC )

メイン・システム・クロック発振回路から出力したクロックを，外部クロックと同じ周波数にして内部回路を動作させる( スルーレート・クロック・モード )か，分周回路を通して原発振周波数の1/2で内部回路を動作させるかを設定するレジスタです。

CCは，1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により，00Hになります。

図5 - 3 発振モード選択レジスタ(CC)のフォーマット

アドレス : 0FF7AH リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
CC	ENMP	0	0	0	0	0	0	0

ENMP	CPUクロックの選択
0	原発振周波数の1/2
1	スルーレート・クロック・モード

**注意** ENMPビットは，ソフトウェアではリセットできません。リセットは，システム・リセットにより行われます。

(3) 発振安定時間指定レジスタ (OSTS)

OSTSは、発振回路の動作を指定するレジスタです。OSTSのEXTCビットにクリスタル/セラミック発振または外部クロックのどちらを使用しているかを設定します。EXTCビットをセット(1)している場合だけ、外部クロック入力時にもSTOPモードを設定することができます。

OSTSは、1ビット転送命令または8ビット転送命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図5-4 発振安定時間指定レジスタ(OSTS)のフォーマット

アドレス: 0FFCFH リセット時: 00H R/W

略号	7	6	5	4	3	2	1	0
OSTS	EXTC	0	0	0	0	OSTS2	OSTS1	OSTS0

EXTC	外部クロックの選択
0	クリスタル/セラミック発振使用時
1	外部クロック使用時

EXTC	OSTS2	OSTS1	OSTS0	発振安定時間の選択
0	0	0	0	$2^{19}/f_{xx}$ ( 41.9 ms )
0	0	0	1	$2^{18}/f_{xx}$ ( 21.0 ms )
0	0	1	0	$2^{17}/f_{xx}$ ( 10.5 ms )
0	0	1	1	$2^{16}/f_{xx}$ ( 5.2 ms )
0	1	0	0	$2^{15}/f_{xx}$ ( 2.6 ms )
0	1	0	1	$2^{14}/f_{xx}$ ( 1.3 ms )
0	1	1	0	$2^{13}/f_{xx}$ ( 655 $\mu$ s )
0	1	1	1	$2^{12}/f_{xx}$ ( 328 $\mu$ s )
1	x	x	x	$512/f_{xx}$ ( 41.0 $\mu$ s )

- 注意1. クリスタル/セラミック発振使用時には、EXTCビットを必ずクリア(0)して使用してください。EXTCビットをセット(1)すると、発振が停止してしまいます。
2. 外部クロック入力時にSTOPモードを使用する場合は、EXTCビットを必ずセット(1)してからSTOPモードを設定してください。外部クロック入力時にEXTCビットがクリア(0)されている状態でSTOPモードを使用すると、 $\mu$ PD784975の破壊または信頼性の低下を招く可能性があります。
3. 外部クロック入力時でEXTCビットをセット(1)する場合には、必ずX2端子に、X1端子に入力しているクロックの逆相のクロックを入力してください。EXTCビットをセット(1)すると、 $\mu$ PD784975はX2端子に入力されているクロックでのみ動作するようになります。

備考1.( )内は、 $f_{xx} = 12.5$  MHz動作時

2. x : don't care

## 5.4 メイン・システム・クロック発振回路

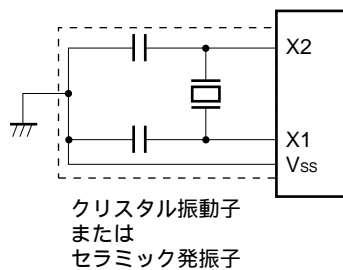
メイン・システム・クロック発振回路はX1, X2端子に接続されたクリスタル振動子またはセラミック発振子(標準: 12.5 MHz)によって発振します。

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端子には, その反転した信号を入力してください。

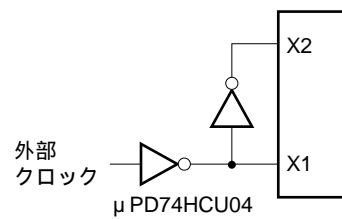
図5 - 5 にメイン・システム・クロック発振回路の外付け回路を示します。

図5 - 5 メイン・システム・クロック発振回路の外付け回路

(a) クリスタル, セラミック発振



(b) 外部クロック



注意1．メイン・システム・クロック発振回路を使用する場合は、配線容量などの影響を避けるために、図5-5の破線の部分を次のように配線してください。

配線は極力短くする。

他の信号線と交差させない。また、変化する大電流が流れる線と接近させない。

発振回路のコンデンサの接地点は、常に $V_{SS}$ と同電位となるようにする。大電流が流れるグランド・パターンに接地しない。

発振回路から信号を取り出さない。

図5-6に発振子の接続の悪い例を示します。

図5-6 発振子の接続の悪い例(1/2)

(a) 接続回路の配線が長い

(b) 信号線が交差している

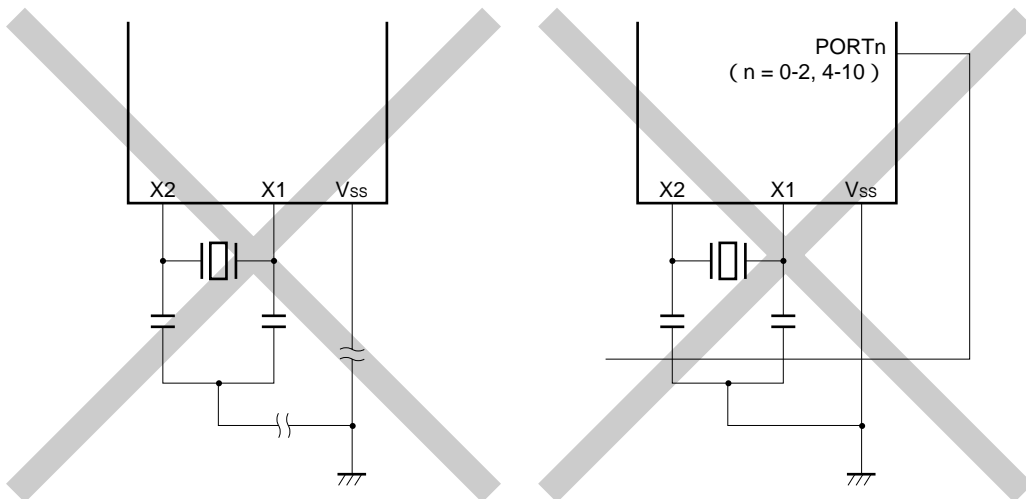
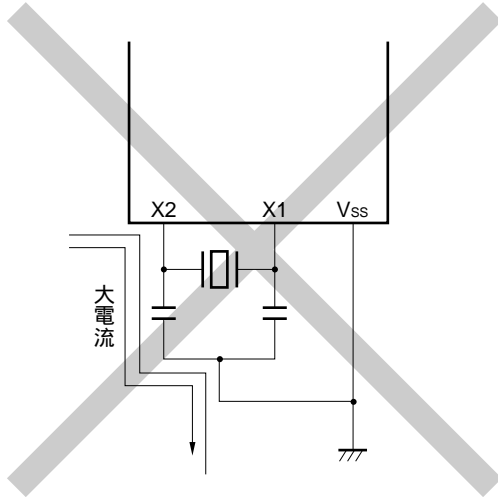
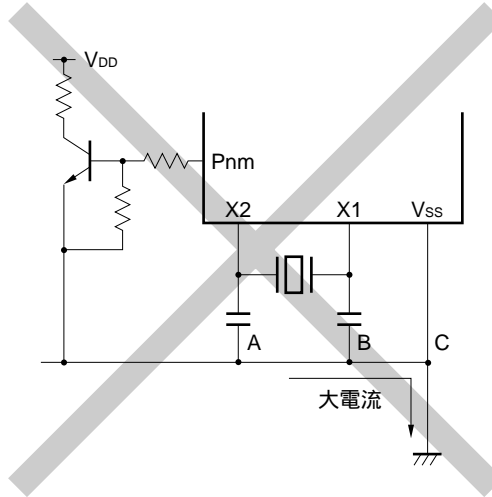


図5 - 6 発振子の接続の悪い例(2/2)

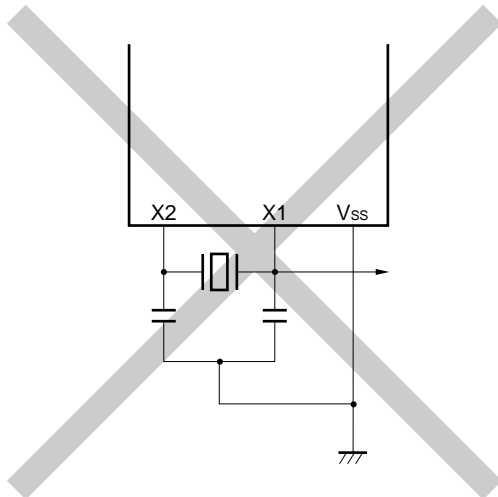
(c) 変化する大電流が信号線に  
近接している



(d) 発振回路部のグランド・ライン上に電流が流れる  
(A点, B点, C点の電位が変動する)



(e) 信号を取り出している



### 5.4.1 分周回路

分周回路は、メイン・システム・クロック発振回路出力 ( $f_{xx}$ ) を分周して、各種クロックを生成します。



## 5.5 クロック発生回路の動作

クロック発生回路は次に示す各種クロックを発生し、かつ、スタンバイ・モードなどのCPUの動作モードを制御します。

- ・メイン・システム・クロック  $f_{XX}$
- ・CPUクロック  $f_{CPU}$
- ・周辺ハードウェアへのクロック
- ・内部システム・クロック  $f_{CLK}$

クロック発生回路の動作はスタンバイ・コントロール・レジスタ(STBC)、発振モード選択レジスタ(CC)により決定され、次のような機能、動作となります。

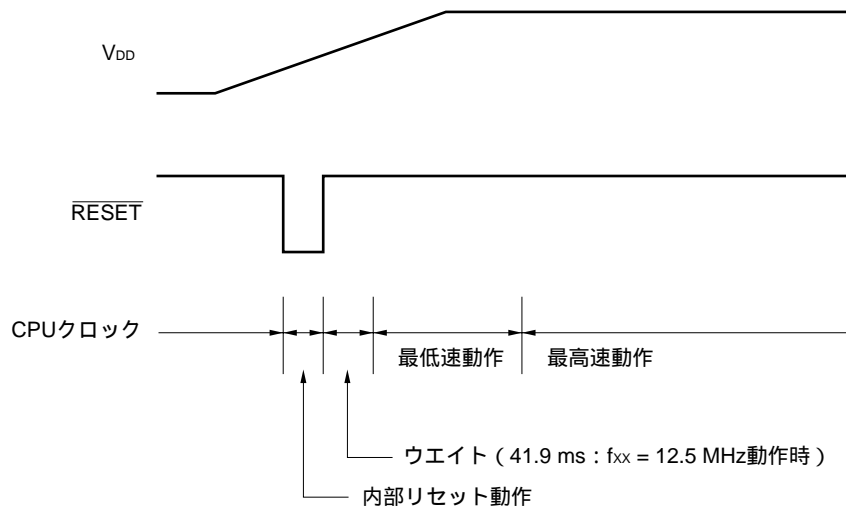
- RESET信号発生によりメイン・システム・クロックの最低速モード(1280 ns :  $f_{XX} = 12.5$  MHz動作時)が選択されます(STBC = 30H, CC = 00H)。なお、RESET端子にロウ・レベルを入力している間、メイン・システム・クロックの発振は停止します。
- メイン・システム・クロックを選択した状態でSTBC, CCの設定により5段階のCPUクロック(80 ns, 160 ns, 320 ns, 640 ns, 1280 ns :  $f_{XX} = 12.5$  MHz発振時)を選択することができます。
- メイン・システム・クロックを選択した状態でSTOPモード、HALTモード、IDLEモードの2つのスタンバイ・モードが使用できます。
- 周辺ハードウェアへのクロックはメイン・システム・クロックを分周して供給されます。このため、メイン・システム・クロックを停止させたときは周辺ハードウェアも停止します(ただし、外部からの入力クロック動作は除く)。

## 5.6 CPUクロックの切り替え

CPUクロックは、スタンバイ・コントロール・レジスタ(STBC)のビット4, 5(CK0, CK1)により切り替えることができます。

CPUクロックの切り替えについて説明します。

図5 - 7 CPUクロックの切り替え



電源投入後、 $\overline{\text{RESET}}$ 端子をロウ・レベルにすることでCPUにリセットがかかります。その後、 $\overline{\text{RESET}}$ 端子をハイ・レベルにするとリセットが解除され、メイン・システム・クロックが発振開始します。このとき、自動的に発振安定時間 ( $2^{19}/f_x$ ) を確保します。

その後、CPUはメイン・システム・クロックの最低速 (1280 ns :  $f_{xx} = 12.5$  MHz発振時) で命令の実行を開始します。

V<sub>DD</sub>電圧が最高速で動作できる電圧まで上昇するのに十分な時間経過後、スタンバイ・コントロール・レジスタ (STBC)、発振モード選択レジスタ (CC) を書き換えて最高速動作を行います。

## 第6章 タイマ/カウンタの概説

16ビット・タイマ/イベント・カウンタを1ユニット、8ビットPWMタイマを2ユニット内蔵しています。

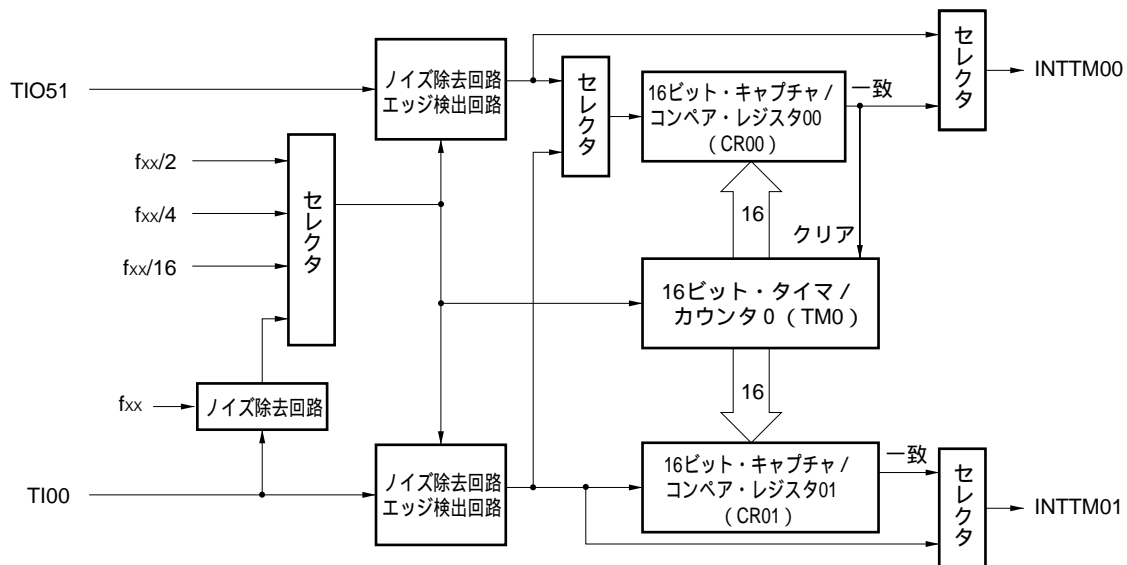
また、計4つの割り込み要求をサポートしていますので、4ユニットのタイマ/カウンタとして機能させることができます。

表6-1 タイマ・カウンタの動作

項目		名称	16ビット・タイマ/ イベント・カウンタ	8ビットPWM タイマ50	8ビットPWM タイマ51
カウント幅	8ビット				
	16ビット				
動作モード	インターバル・タイマ		1ch	1ch	1ch
	外部イベント・カウンタ				
機能	タイマ出力			1ch	1ch
	PWM出力				
	方形波出力				
	パルス幅測定		2入力		
	割り込み要求数		2	1	1

図6-1 タイマ/カウンタのブロック図(1/2)

### 16ビット・タイマ/イベント・カウンタ (TM0)

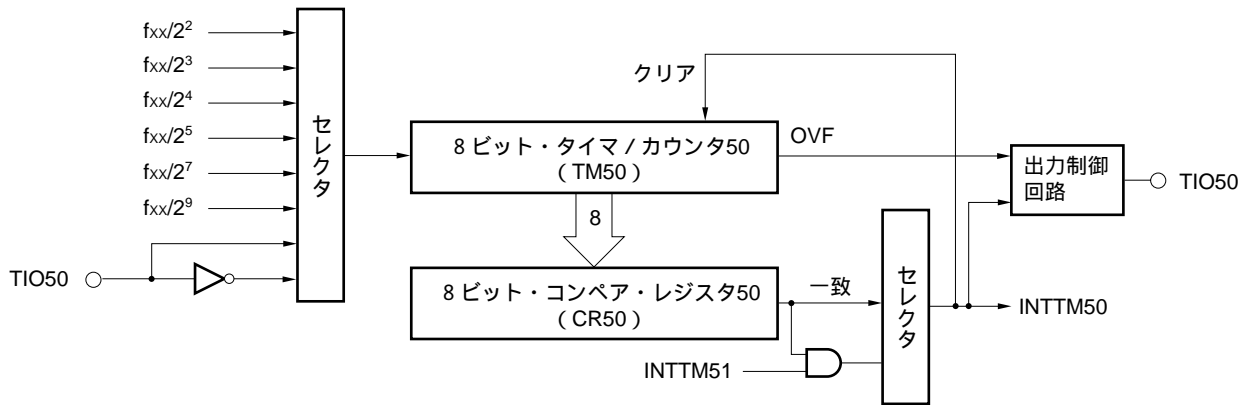


備考1 . fxx : システム・クロック周波数

2 . TIO51端子は、TM51への外部クロック入力機能/タイマ出力機能を兼用しています。

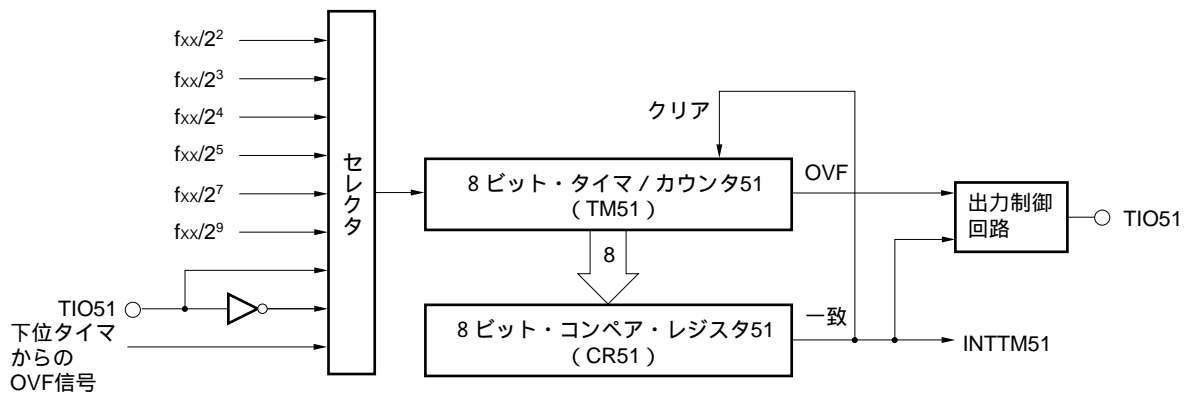
図6-1 タイマ/カウンタのブロック図(2/2)

8ビットPWMタイマ50 (TM50)



- 備考1 . fxx : システム・クロック周波数  
 2 . OVF : オーバフロー・フラグ

8ビットPWMタイマ51 (TM51)



- 備考1 . fxx : システム・クロック周波数  
 2 . OVF : オーバフロー・フラグ  
 3 . TIO51端子は, TM0へのキャプチャ入力機能と兼用しています。

## 第7章 16ビット・タイマ/イベント・カウンタ

### 7.1 機能

16ビット・タイマ/イベント・カウンタ (TM0) には、次のような機能があります。

- ・インターバル・タイマ
- ・パルス幅測定
- ・外部イベント・カウンタ

#### (1) インターバル・タイマ

あらかじめ設定した任意の時間間隔で割り込み要求を発生します。

#### (2) パルス幅測定

外部から入力される信号のパルス幅を測定できます。

#### (3) 外部イベント・カウンタ

外部から入力される信号のパルス数を測定できます。

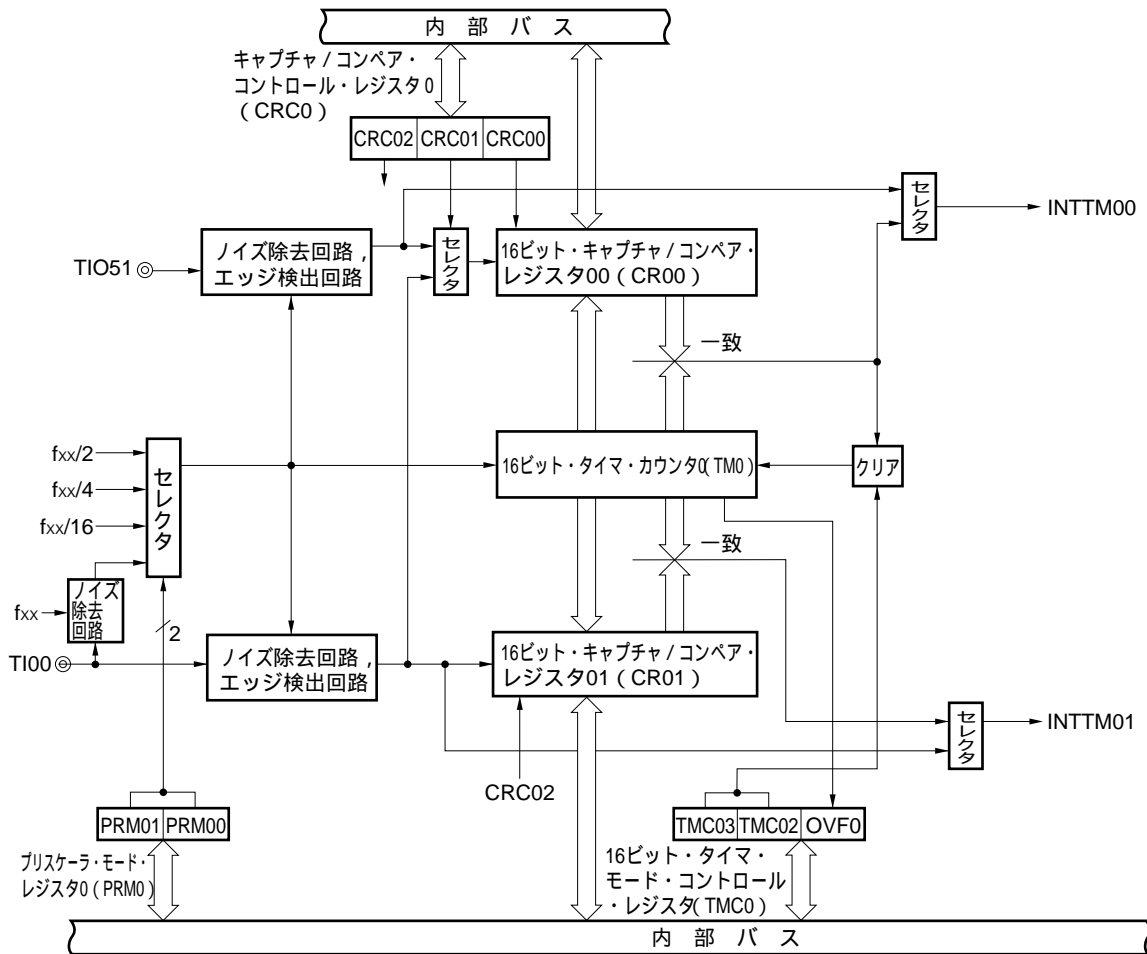
## 7.2 構成

16ビット・タイマ/イベント・カウンタ (TM0) は、次のハードウェアで構成されています。

表7-1 16ビット・タイマ/イベント・カウンタ (TM0) の構成

項目	構成
タイマ・レジスタ	16ビット×1本 (TM0)
レジスタ	16ビット・キャプチャ/コンペア・レジスタ : 16ビット×2本 (CR00, CR01)
外部クロック入力	1本 (TIO0)
制御レジスタ	16ビット・タイマ・モード・コントロール・レジスタ0 (TMC0) キャプチャ/コンペア・コントロール・レジスタ0 (CRC0) プリスケアラ・モード・レジスタ0 (PRM0)

図7-1 16ビット・タイマ/イベント・カウンタ (TM0) のブロック図



**(1) 16ビット・タイマ・カウンタ0 (TM0)**

TM0は、カウント・パルスをカウントする16ビットのリード専用レジスタです。

入力クロックの立ち上がりに同期して、カウンタをインクリメントします。また、動作中にカウント値を読み出した場合、カウント・クロックの入力を一時停止し、その時点でのカウント値を読み出します。次の場合、カウント値は0000Hになります。

$\overline{\text{RESET}}$ 入力

TMC03, TMC02をクリア

TI00有効エッジ入力でクリア&スタート・モード時のTI00有効エッジが入力されたとき

CR00の一致でクリア&スタート・モード時のTM0とCR00の一致

(2) 16ビット・キャプチャ/コンペア・レジスタ00 (CR00)

CR00は、キャプチャ・レジスタとコンペア・レジスタの機能をあわせ持った16ビットのレジスタです。キャプチャ/コンペア・コントロール・レジスタ0のビット0 (CRC00)により、キャプチャ・レジスタとして使用するのか、コンペア・レジスタとして使用するのかを設定します。

・CR00をコンペア・レジスタとして使用するとき

CR00に設定した値と16ビット・タイマ・カウンタ0 (TM0)のカウント値を常に比較し、一致したときに割り込み要求 (INTTM00)を発生します。TM0をインターバル・タイマ動作に設定したとき、インターバル時間を保持するレジスタとしても使用できます。

・CR00をキャプチャ・レジスタとして使用するとき

キャプチャ・トリガとしてTI00端子、またはTIO51端子の有効エッジが選択できます。TI00, TIO51の有効エッジは、プリスケラ・モード・レジスタ0 (PRM0)で設定します。キャプチャ・トリガをTI00端子の有効エッジに指定したときは表7 - 2、キャプチャ・トリガをTIO51端子の有効エッジに指定したときは表7 - 3のようになります。

表7 - 2 TI00端子の有効エッジとCR00のキャプチャ・トリガ

ES01	ES00	TI00端子の有効エッジ	CR00のキャプチャ・トリガ
0	0	立ち下がりエッジ	立ち上がりエッジ
0	1	立ち上がりエッジ	立ち下がりエッジ
1	0	設定禁止	設定禁止
1	1	立ち上がり、立ち下がりの両エッジ	キャプチャ動作しない

表7 - 3 TIO51端子の有効エッジとCR00のキャプチャ・トリガ

ES11	ES10	TIO51端子の有効エッジ	CR00のキャプチャ・トリガ
0	0	立ち下がりエッジ	立ち下がりエッジ
0	1	立ち上がりエッジ	立ち上がりエッジ
1	0	設定禁止	設定禁止
1	1	立ち上がり、立ち下がりの両エッジ	立ち上がり、立ち下がりの両エッジ

CR00は、16ビット・メモリ操作命令で設定します。

RESET入力により、0000Hになります。

**注意** CR00には、0000H以外の値を設定してください。したがって、イベント・カウンタとしての使用時、1パルスのカウント動作はできません。



## (3) 16ビット・キャプチャ/コンペア・レジスタ01 (CR01)

キャプチャ・レジスタとコンペア・レジスタの機能をあわせ持った16ビットのレジスタです。キャプチャ/コンペア・コントロール・レジスタ0のビット2 (CRC02) により、キャプチャ・レジスタとして使用するのか、コンペア・レジスタとして使用するのかを設定します。

- ・CR01をコンペア・レジスタとして使用するとき

CR01に設定した値と16ビット・タイマ・カウンタ0 (TM0) のカウント値を常に比較し、一致したときに割り込み要求 (INTTM01) を発生します。

- ・CR01をキャプチャ・レジスタとして使用するとき

キャプチャ・トリガとしてTI00端子の有効エッジが選択できます。TI00の有効エッジは、プリスケラ・モード・レジスタ0 (PRM0) で設定します。

キャプチャ・トリガをTI00端子の有効エッジに指定したときは表7 - 4のようになります。

表7 - 4 TI00端子の有効エッジとCR01のキャプチャ・トリガ

ES01	ES00	TI00端子の有効エッジ	CR01のキャプチャ・トリガ
0	0	立ち下がりエッジ	立ち下がりエッジ
0	1	立ち上がりエッジ	立ち上がりエッジ
1	0	設定禁止	設定禁止
1	1	立ち上がり、立ち下がりの両エッジ	立ち上がり、立ち下がりの両エッジ

CR01は、16ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、0000Hになります。

**注意** CR01には、0000H以外の値を設定してください。したがって、イベント・カウンタとしての使用時、1パルスのカウント動作はできません。

## 7.3 制御レジスタ

16ビット・タイマ/イベント・カウンタ (TM0) を制御するレジスタには、次の4種類があります。

- ・16ビット・タイマ・モード・コントロール・レジスタ0 (TMC0)
- ・キャプチャ/コンペア・コントロール・レジスタ0 (CRC0)
- ・プリスケラ・モード・レジスタ0 (PRM0)

### (1) 16ビット・タイマ・モード・コントロール・レジスタ0 (TMC0)

16ビット・タイマの動作モード、16ビット・タイマ・カウンタ0 (TM0) のクリア・モード、およびオーバーフローを検出するレジスタです。

TMC0は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

**注意** 16ビット・タイマ・カウンタ0 (TM0) は、TMC02, TMC03に0, 0 (動作停止モード) 以外の値を設定した時点で動作を開始します。動作を停止させるには、TMC02, TMC03に0, 0を設定してください。

図7 - 2 16ビット・タイマ・モード・コントロール・レジスタ0 (TMC0) のフォーマット

アドレス : 0FF18H リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
TMC0	0	0	0	0	TMC03	TMC02	0	OVF0

TMC03	TMC02	TM0の動作モードの指定
0	0	動作停止 (TM0は0にクリア)
0	1	フリーランニング・モード
1	0	TI00有効エッジの入力でクリア&スタート・モード
1	1	TM0とCR00の一致でクリア&スタート・モード

OVF0	16ビット・タイマ・カウンタ0 (TM0) のオーバーフロー検出
0	オーバーフローなし
1	オーバーフローあり

**注意 1** . TI00端子の有効エッジの設定は、プリスケラ・モード・レジスタ0 (PRM0) で行います。

**2** . TM0とCR00の一致でクリア&スタートするモードを選択した場合、CR00の設定値がFFFFHで、TM0の値がFFFFHから0000Hに変化するとき、OVF0フラグが1に設定されません。

**備考 1** . OVF0は、“0”を書き込むことで、どのタイミングでもクリアされます。

**2** . TI00 : 16ビット・タイマ/イベント・カウンタ0 (TM0) の入力端子

TM0 : 16ビット・タイマ・カウンタ0

CR00 : コンペア・レジスタ00

CR01 : コンペア・レジスタ01

(2) キャプチャ/コンペア・コントロール・レジスタ0 (CRC0)

キャプチャ/コンペア・レジスタ (CR00, CR01) の動作を制御するレジスタです。  
 CRC0は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。  
 $\overline{\text{RESET}}$ 入力により、00Hになります。

図7-3 キャプチャ/コンペア・コントロール・レジスタ0 (CRC0) のフォーマット

アドレス：0FF16H リセット時：00H R/W

略号	7	6	5	4	3	2	1	0
CRC0	0	0	0	0	0	CRC02	CRC01	CRC00

CRC02	CR01の動作モードの選択
0	コンペア・レジスタとして動作
1	キャプチャ・レジスタとして動作

CRC01	CR00のキャプチャ・トリガの選択
0	TIO51の有効エッジでキャプチャする
1	TI00の有効エッジの逆相でキャプチャする

CRC00	CR00の動作モードの選択
0	コンペア・レジスタとして動作
1	キャプチャ・レジスタとして動作

- 注意1 . CRC0の設定は、必ずタイマ動作を停止させてから行ってください。
- 16ビット・タイマ・モード・コントロール・レジスタ (TMC0) で、TM0とCR00の一致でクリア&スタート・モードを選択したとき、CRC00をキャプチャ・レジスタに指定しないでください。
  - プリスケアラ・モード・レジスタ0 (PRM0) で、TI00端子の有効エッジを立ち上がり、立ち下りの両エッジに指定した場合、CRC00はキャプチャ動作を行えません。
  - キャプチャを確実に行うために、キャプチャ・トリガはPRM0で選択したカウント・クロックの2回分より長いパルスが必要とします。

(3) プリスケアラ・モード・レジスタ0 (PRM0)

16ビット・タイマ/イベント・カウンタ (TM0) のカウント・クロックおよびTIO0, TIO51入力の有効エッジを設定するレジスタです。

PRM0は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図7-4 プリスケアラ・モード・レジスタ0 (PRM0) のフォーマット

アドレス：0FF1CH リセット時：00H R/W

略号	7	6	5	4	3	2	1	0
PRM0	ES11	ES10	ES01	ES00	0	0	PRM01	PRM00

ES11	ES10	TIO51有効エッジの選択
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、立ち下がりの両エッジ

ES01	ES00	TIO0有効エッジの選択
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、立ち下がりの両エッジ

PRM01	PRM00	カウント・クロックの選択
0	0	$f_{xx}/2$ (6.25 MHz)
0	1	$f_{xx}/4$ (3.13 MHz)
1	0	$f_{xx}/16$ (781 kHz)
1	1	TIO0有効エッジ

**注意** カウント・クロックにTIO0の有効エッジを設定する場合、TIO0有効エッジでクリア&スタート・モードおよびキャプチャ・トリガに設定しないでください。

**備考** ( )内は、 $f_{xx} = 12.5$  MHz動作時

## 7.4 動作

### 7.4.1 インターバル・タイマ (16ビット) としての動作

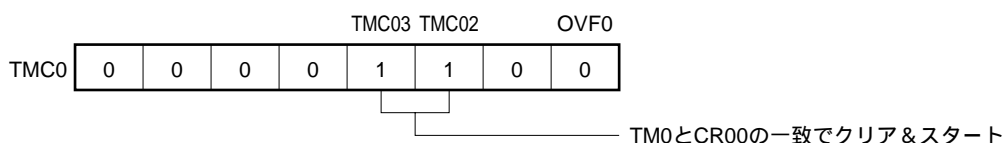
16ビット・タイマ・モード・コントロール・レジスタ0(TMC0)と、キャプチャ/コンペア・コントロール・レジスタ0(CRC0)を図7-5のように設定することにより、インターバル・タイマとして動作します。16ビット・キャプチャ/コンペア・レジスタ0(CR00)にあらかじめ設定したカウント値をインターバルとし、繰り返し割り込み要求を発生します。

16ビット・タイマ・カウンタ0(TM0)のカウント値がCR00に設定した値と一致したとき、TM0の値を0にクリアしてカウントを継続するとともに割り込み要求信号(INTTM0)を発生します。

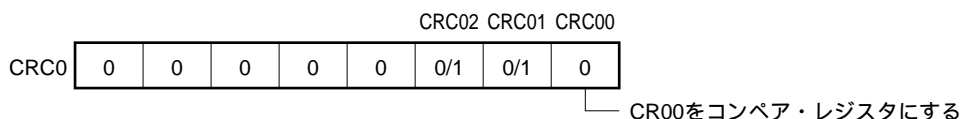
プリスケアラ・モード・レジスタ0(PRM0)のビット0,1(PRM00, PRM01)で16ビット・タイマ/カウンタのカウント・クロックを選択できます。

図7-5 インターバル・タイマ動作時の制御レジスタ設定内容

(a) 16ビット・タイマ・モード・コントロール・レジスタ0(TMC0)



(b) キャプチャ/コンペア・コントロール・レジスタ0(CRC0)



**備考** 0/1: 0または1を設定することにより、インターバル・タイマと同時にほかの機能を使用することができます。詳細は、図7-2, 7-3を参照してください。

図7-6 インターバル・タイマの構成図

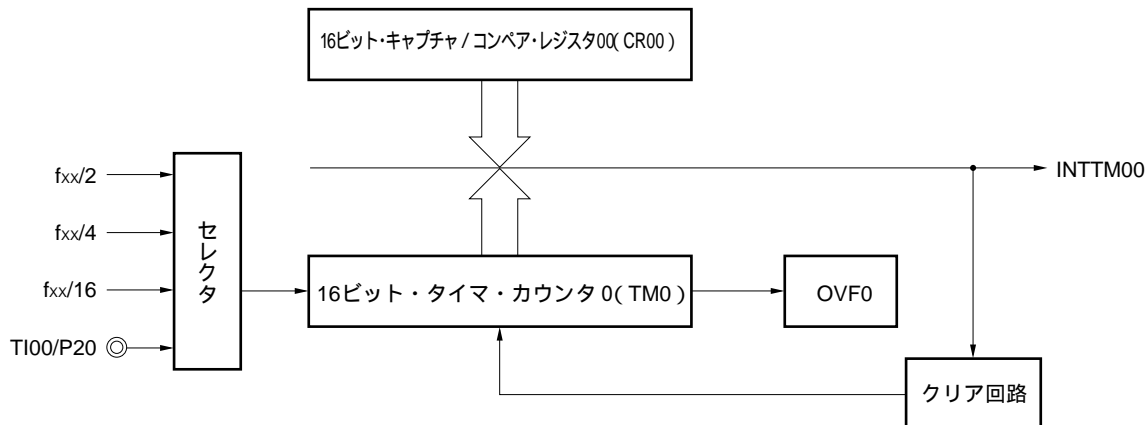
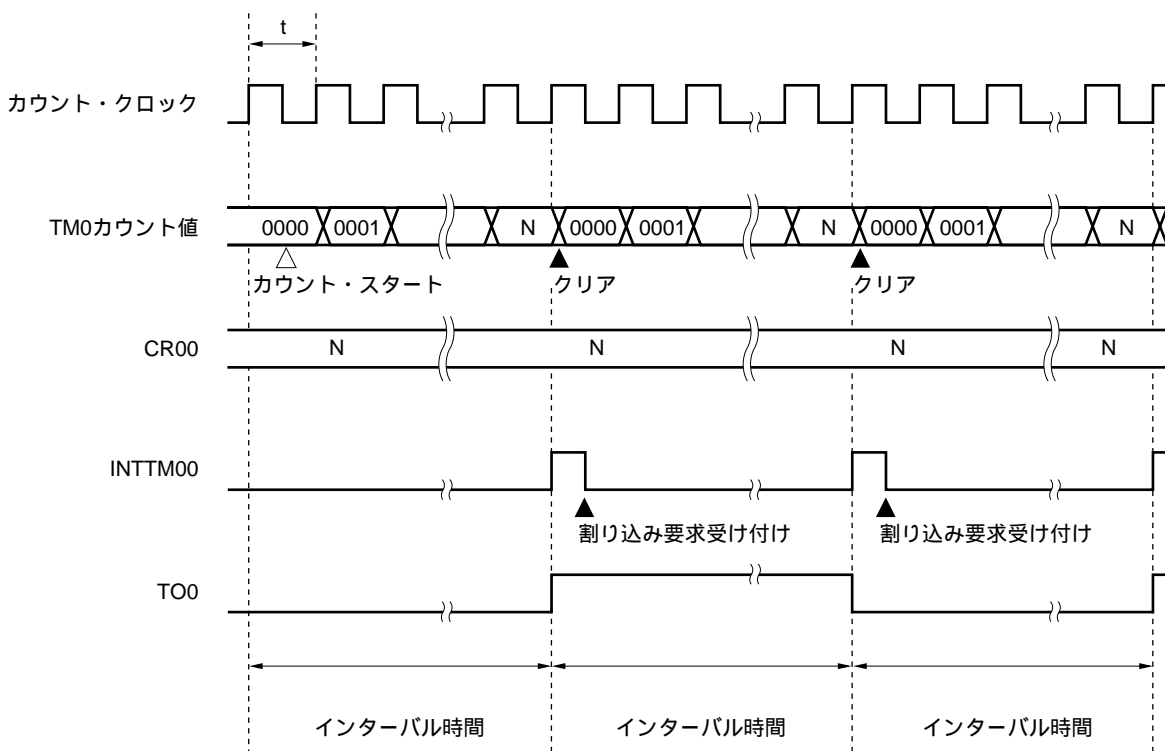


図7-7 インターバル・タイマ動作のタイミング



備考 インターバル時間 = (N + 1) × t : N = 0001H-FFFFH

## 7.4.2 パルス幅測定としての動作

16ビット・タイマ・カウンタ0(TM0)を使用し、TI00/P20端子およびTIO51/P66端子に入力される信号のパルス幅を測定できます。

測定方法は、TM0をフリーランニングさせて測定する方法とTI00/P20端子に入力される信号のエッジに同期してタイマをリスタートさせて測定する方法があります。

### (1) フリーランニング・カウンタとキャプチャ・レジスタ1本によるパルス幅測定

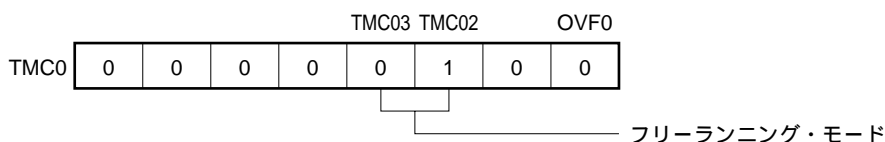
16ビット・タイマ・カウンタ0(TM0)をフリーランニングで動作させているとき(図7-8のレジスタの設定参照)、TI00/P20端子にプリスケアラ・モード・レジスタ0(PRM0)で指定したエッジが入力されるとTM0の値を16ビット・キャプチャ/コンペア・レジスタ01(CR01)に取り込み、外部割り込み要求信号(INTTM01)をセットします。

エッジ指定はPRM0のビット4, 5(ES00, ES01)で行い、立ち上がり、立ち下がり、両エッジの3種類の選択ができます。

PRM0で選択したカウント・クロックでサンプリングを行い、TI00端子の有効レベルを2回検出することではじめてキャプチャ動作を行うため、短いパルス幅のノイズを除去できます。

図7-8 フリーランニング・カウンタとキャプチャ・レジスタ1本によるパルス幅測定時の制御レジスタ設定内容

#### (a) 16ビット・タイマ・モード・コントロール・レジスタ(TMC0)



#### (b) キャプチャ/コンペア・コントロール・レジスタ0(CRC0)

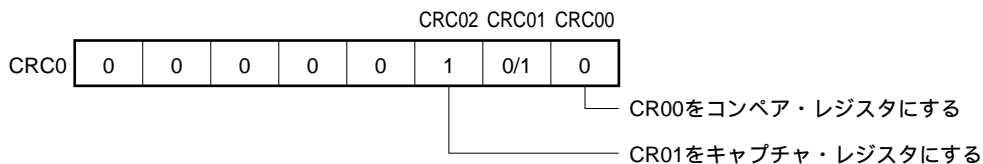




図7-9 フリーランニング・カウンタによるパルス幅測定の構成図

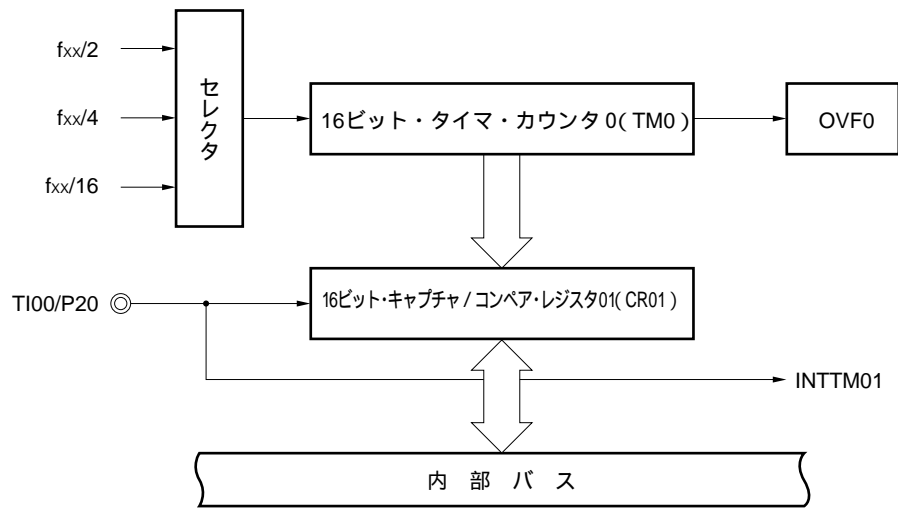
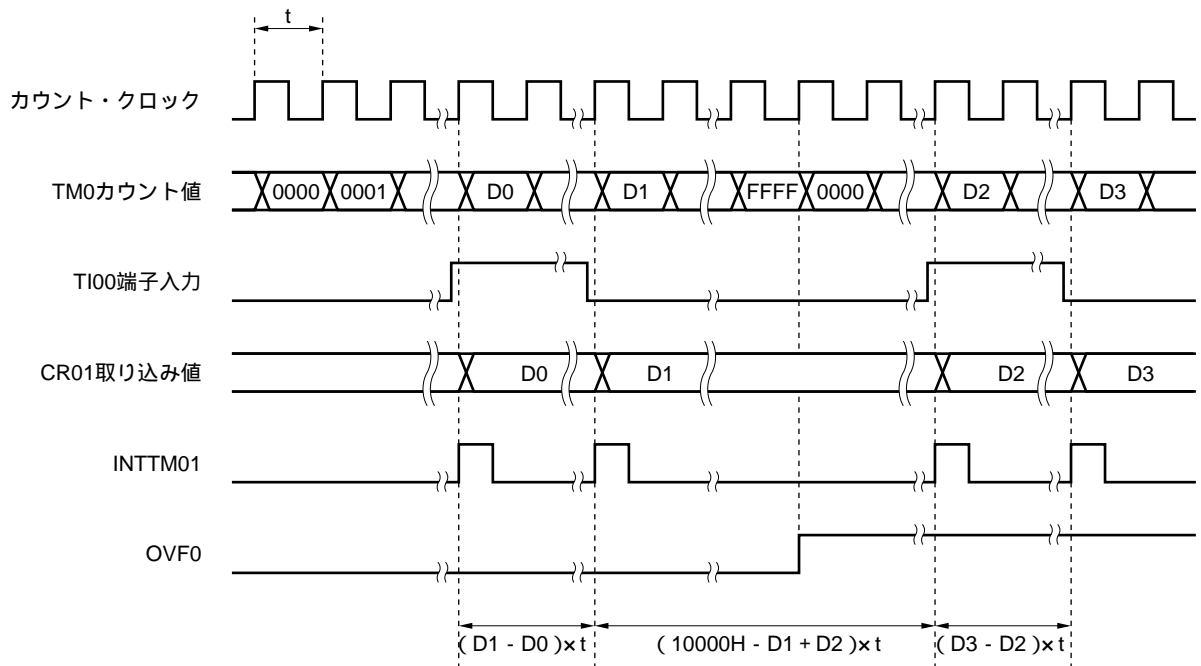


図7-10 フリーランニング・カウンタとキャプチャ・レジスタ1本によるパルス幅測定動作のタイミング(両エッジ指定時)



(2) フリーランニング・カウンタによる2つのパルス幅測定

16ビット・タイマ・カウンタ0(TM0)をフリーランニングで動作させているとき(図7-11のレジスタの設定参照), TIO0/P20端子およびTIO51/P66端子に入力される2つの信号のパルス幅を同時に測定できます。

TIO0/P20端子にプリスケアラ・モード・レジスタ0(PRM0)のビット4, 5(ES00, ES01)で指定したエッジが入力されるとTM0の値を16ビット・キャプチャ/コンペア・レジスタ01(CR01)に取り込み, 外部割り込み要求信号(INTTM01)をセットします。

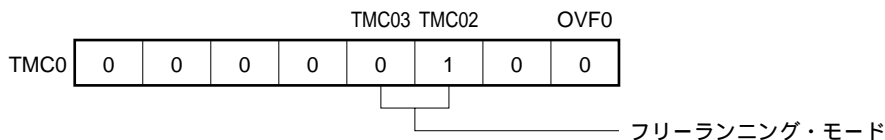
また, TIO51/P66端子にPRM0のビット6, 7(ES10, ES11)で指定したエッジが入力されるとTM0の値を16ビット・キャプチャ/コンペア・レジスタ00(CR00)に取り込み, 外部割り込み要求信号(INTTM00)をセットします。

TIO0/P20端子とTIO51/P66端子のエッジ指定は, 立ち上がり, 立ち下がり, 両エッジの3種類の選択ができます。

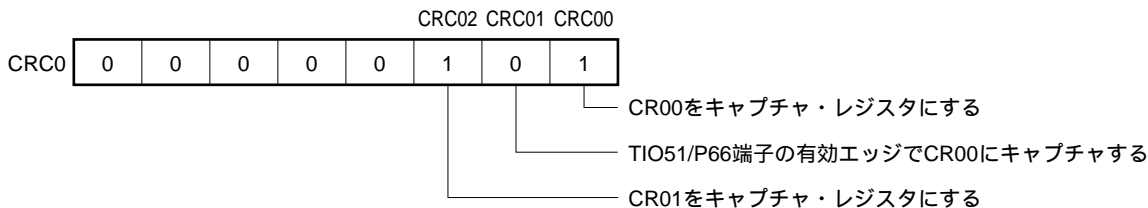
プリスケアラ・モード・レジスタ0(PRM0)で選択したカウント・クロックでサンプリングを行い, TIO0/P20端子またはTIO51/P66端子の有効レベルを2回検出することではじめてキャプチャ動作を行うため, 短いパルス幅のノイズを除去できます。

図7-11 フリーランニング・カウンタによる2つのパルス幅測定時の制御レジスタ設定内容

(a) 16ビット・タイマ・モード・コントロール・レジスタ0(TMC0)



(b) キャプチャ/コンペア・コントロール・レジスタ0(CRC0)



・キャプチャ動作について(フリーランニング・モード)

キャプチャ・トリガが入力されたときのキャプチャ・レジスタの動作を示します。

図7 - 12 立ち上がりエッジ指定時のCR01キャプチャ動作

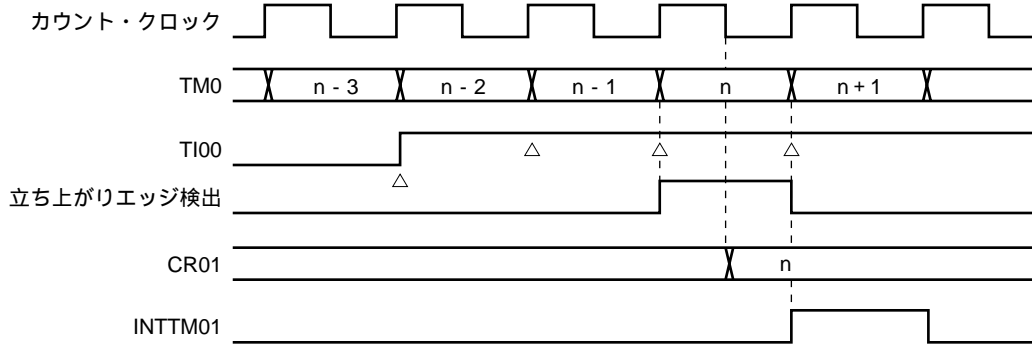
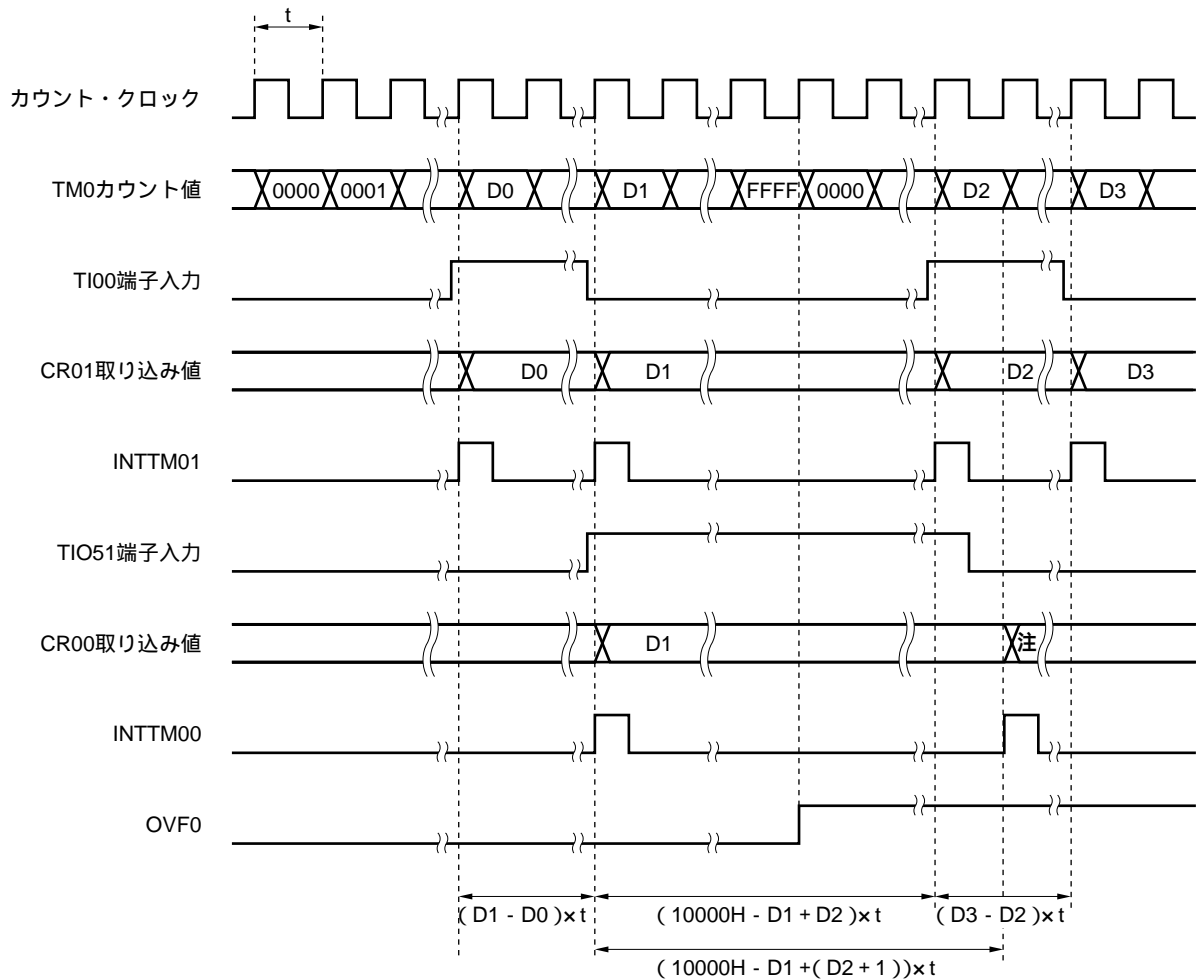


図7 - 13 フリーランニング・カウンタによるパルス幅測定動作のタイミング(両エッジ指定時)



注  $D2 + 1$

**注意** 図7 - 13のTI00端子入力およびTI01端子入力によるキャプチャ動作，割り込み要求発生タイミングには，簡略化のためノイズ除去による遅延は考慮していません。正確には，図7 - 12立ち上がりエッジ指定時のCR01キャプチャ動作を参照してください。

(3) フリーランニング・カウンタとキャプチャ・レジスタ2本によるパルス幅測定

16ビット・タイマ・カウンタ0(TM0)をフリーランニングで動作させているとき(図7 - 14のレジスタの設定参照)，TI00/P20端子に入力する信号のパルス幅を測定できます。

TI00/P20端子にプリスケアラ・モード・レジスタ0(PRM0)のビット4, 5(ES00, ES01)で指定したエッジが入力されるとTM0の値を16ビット・キャプチャ/コンペア・レジスタ01(CR01)に取り込み，外部割り込み要求信号(INTTM01)をセットします。

また，CR01へのキャプチャ動作と逆のエッジ入力でTM0の値を16ビット・キャプチャ/コンペア・レジスタ00(CR00)に取り込みます。

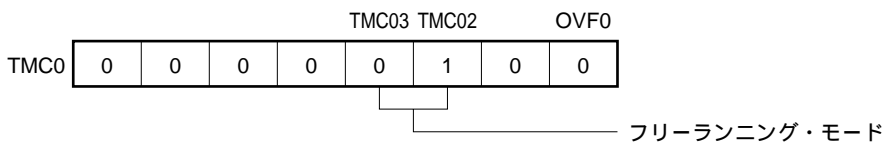
TI00/P20端子のエッジ指定は，立ち上がり，立ち下がりエッジの2種類の選択ができます。

PRM0で選択したカウント・クロックでサンプリングを行い，TI00/P20端子の有効レベルを2回検出することではじめてキャプチャ動作を行うため，短いパルス幅のノイズを除去できます。

**注意** TI00/P20端子の有効エッジを，立ち上がり，立ち下がりの両エッジに指定した場合，CR00はキャプチャ動作を行えません。

図7 - 14 フリーランニング・カウンタとキャプチャ・レジスタ2本によるパルス幅測定時の制御レジスタ設定内容

(a) 16ビット・タイマ・モード・コントロール・レジスタ0(TMC0)



(b) キャプチャ/コンペア・コントロール・レジスタ0(CRC0)

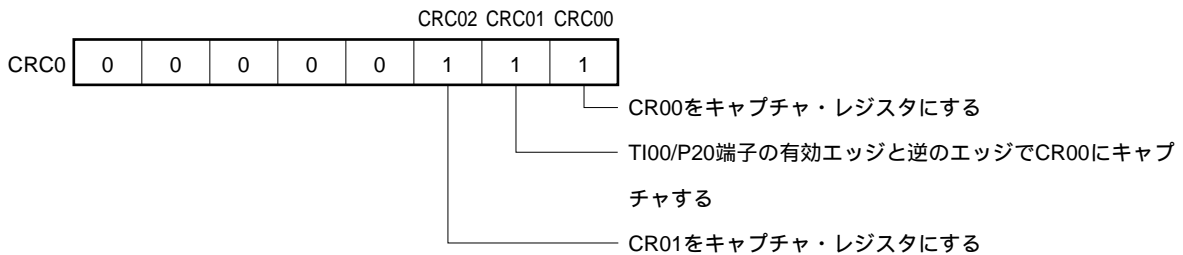
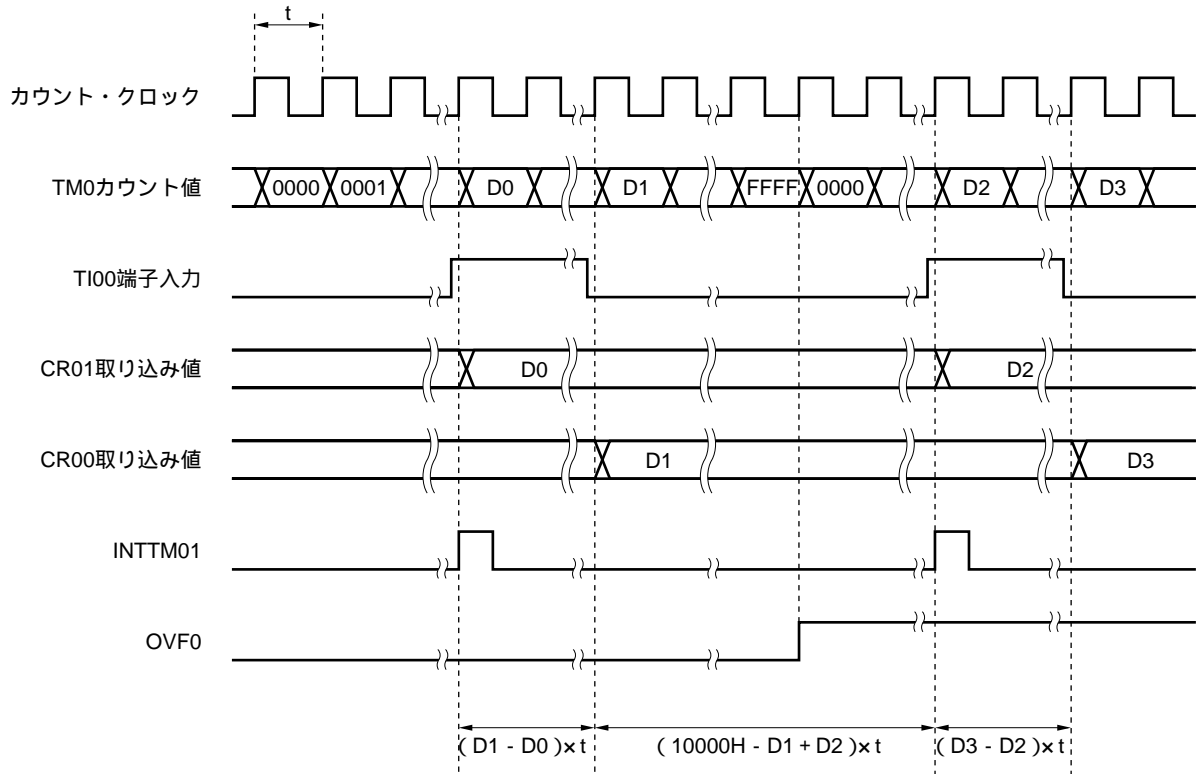


図7 - 15 フリーランニング・カウンタとキャプチャ・レジスタ2本による  
パルス幅測定動作のタイミング(立ち上がりエッジ指定時)



**注意** 図7 - 15のTI00端子入力によるキャプチャ動作，割り込み要求発生タイミングには，簡略化のためノイズ除去による遅延は考慮していません。正確には，図7 - 12 立ち上がりエッジ指定時のCR01キャプチャ動作を参照してください。

(4) リスタートによるパルス幅測定

TI00/P20端子への有効エッジを検出したとき、16ビット・タイマ・カウンタ0(TM0)のカウント値を16ビット・キャプチャ/コンペア・レジスタ0(CR01)に取り込んだのち、TM0をクリアしてカウントを再開することによりTI00/P20端子に入力された信号のパルス幅を測定します(図7-16のレジスタの設定参照)。

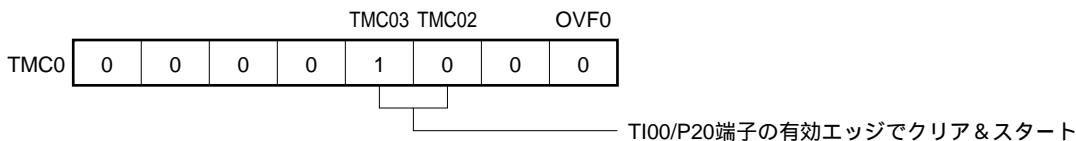
エッジ指定はプリスケアラ・モード・レジスタ0(PRM0)のビット4, 5(ES00, ES01)により、立ち上がり、立ち下がりエッジの2種類から選択できます。

PRM0で選択したカウント・クロックでサンプリングを行い、TI00/P20端子の有効レベルを2回検出することではじめてキャプチャ動作を行うため、短いパルス幅のノイズを除去できます。

**注意** TI00/P20端子の有効エッジを、立ち上がり、立ち下がりの両エッジに指定した場合、16ビット・キャプチャ/コンペア・レジスタ00(CR00)はキャプチャ動作を行えません。

図7-16 リスタートによるパルス幅測定時の制御レジスタ設定内容

(a) 16ビット・タイマ・モード・コントロール・レジスタ0(TMC0)



(b) キャプチャ/コンペア・コントロール・レジスタ0(CRC0)

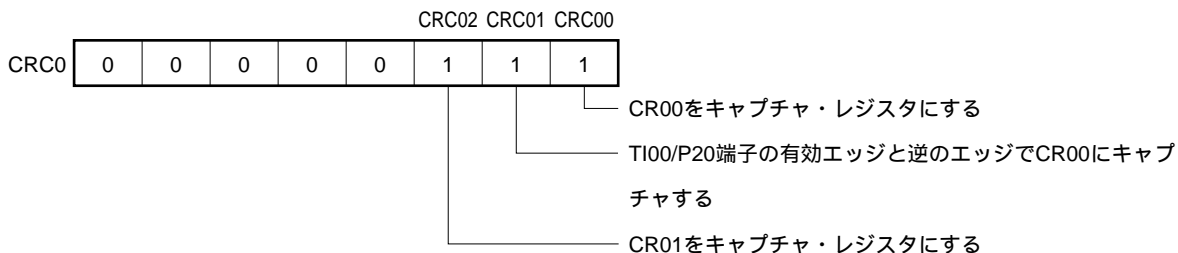
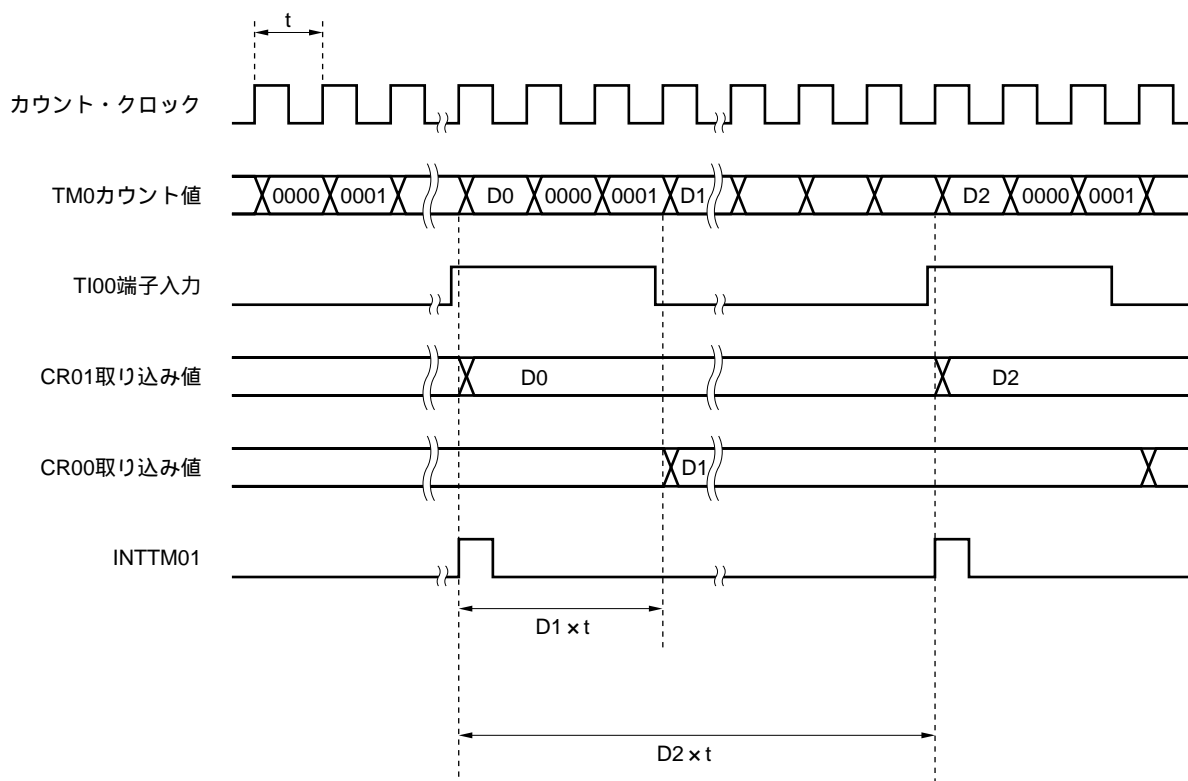


図7-17 リスタートによるパルス幅測定動作のタイミング(立ち上がりエッジ指定時)



**注意** 図7-17のTI00端子入力によるキャプチャ動作, 割り込み要求発生タイミングには, 簡略化のためノイズ除去による遅延は考慮していません。正確には, 図7-12 立ち上がりエッジ指定時のCR01キャプチャ動作を参照してください。

### 7.4.3 外部イベント・カウンタとしての動作

外部イベント・カウンタは, TI00/P20端子に入力される外部からのクロック・パルス数を16ビット・タイマ・カウンタ0(TM0)でカウントするものです。

プリスケアラ・モード・レジスタ0(PRM0)で指定した有効エッジが入力されるたびに, TM0がインクリメントされます。

TM0の計数値が16ビット・キャプチャ/コンペア・レジスタ0(CR00)の値と一致すると, TM0は0にクリアされ, 割り込み要求信号(INTTM00)が発生します。

なお, CR00には0000H以外の値を入れてください(1パルスのカウント動作はできません)。

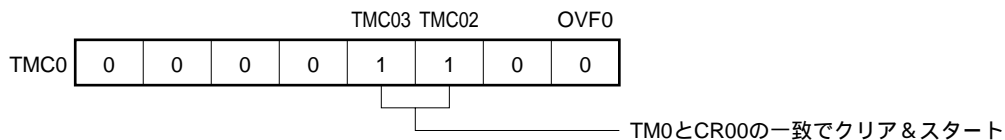
TI00/P20端子の入力クロックでカウント動作を行うには, PRM0のビット0, 1(PRM00, PRM01)でTI00の有効エッジを指定します。

エッジ指定はPRM0のビット4, 5(ES00, ES01)により, 立ち上がり, 立ち下がり, 両エッジの3種類から選択できます。

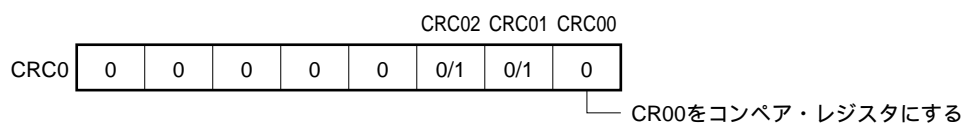
PRM0で選択したカウント・クロックでサンプリングを行い, TI00端子の有効レベルを2回検出することではじめてキャプチャ動作を行うため, 短いパルス幅のノイズを除去できます。

図7 - 18 外部イベント・カウンタ・モード時の制御レジスタ設定内容

(a) 16ビット・タイマ・モード・コントロール・レジスタ0(TMC0)



(b) キャプチャ/コンペア・コントロール・レジスタ0 (CRC0)



備考 0/1 : 0または1を設定することにより、外部イベント・カウンタと同時にほかの機能を使用できます。詳細は、図7 - 2 , 7 - 3を参照してください。

図7 - 19 外部イベント・カウンタの構成図

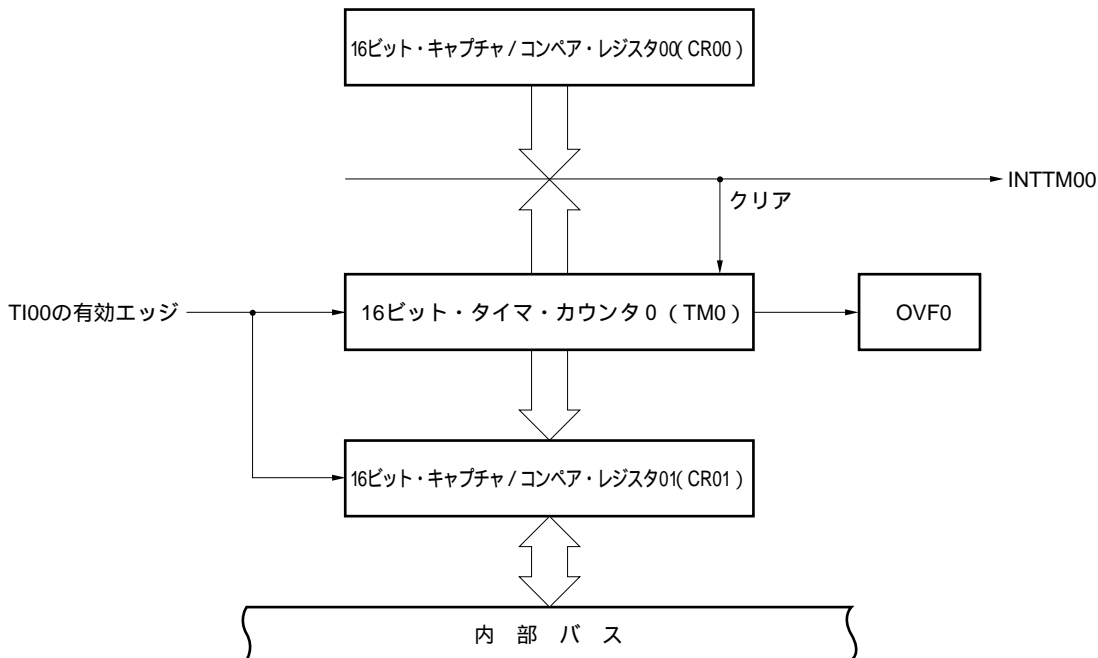
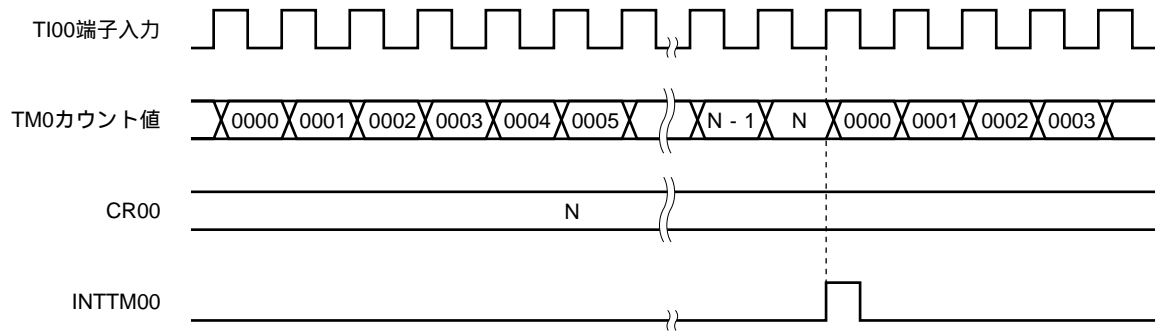




図7 - 20 外部イベント・カウンタ動作のタイミング(立ち上がりエッジ指定時)



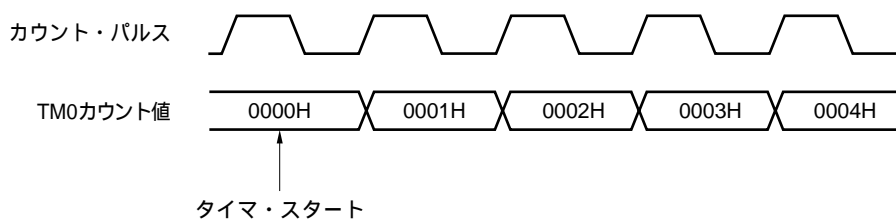
**注意** 外部イベント・カウンタのカウント値を読み出す場合は、TM0を読み出してください。

## 7.5 注意事項

### (1) タイマ・スタート時の誤差

タイマ・スタート後、一致信号が発生するまでの時間は、最大で1クロック分の誤差が生じます。これはカウント・パルスに対して16ビット・タイマ・カウンタ0(TM0)のスタートが非同期で行われるためです。

図7-21 16ビット・タイマ・カウンタ0のスタート・タイミング



### (2) 16ビット・コンペア・レジスタの設定

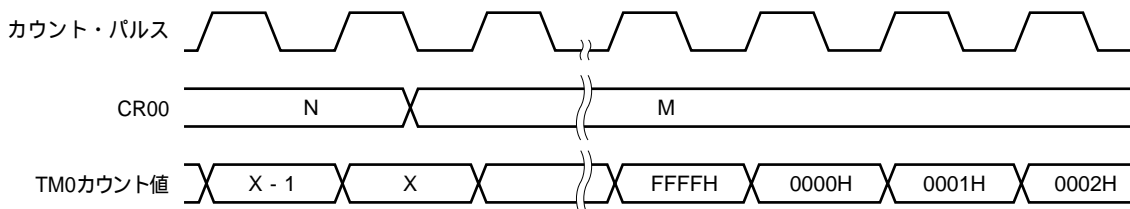
16ビット・キャプチャ/コンペア・レジスタ00, 01(CR00, CR01)には、0000H以外の値を設定してください。

したがって、イベント・カウンタとして使用時、1パルスのカウント動作はできません。

### (3) タイマ・カウント動作中のコンペア・レジスタの変更後の動作

16ビット・キャプチャ/コンペア・レジスタ0(CR00)の変更後の値が、16ビット・タイマ・カウンタ0(TM0)の値よりも小さいとき、TM0はカウントを継続しオーバフローして0から再カウントします。したがって、CR00の変更後の値(M)が変更前の値(N)よりも小さいときは、CR00を変更後、タイマを再スタートさせる必要があります。

図7-22 タイマ・カウント動作中のコンペア・レジスタの変更後のタイミング

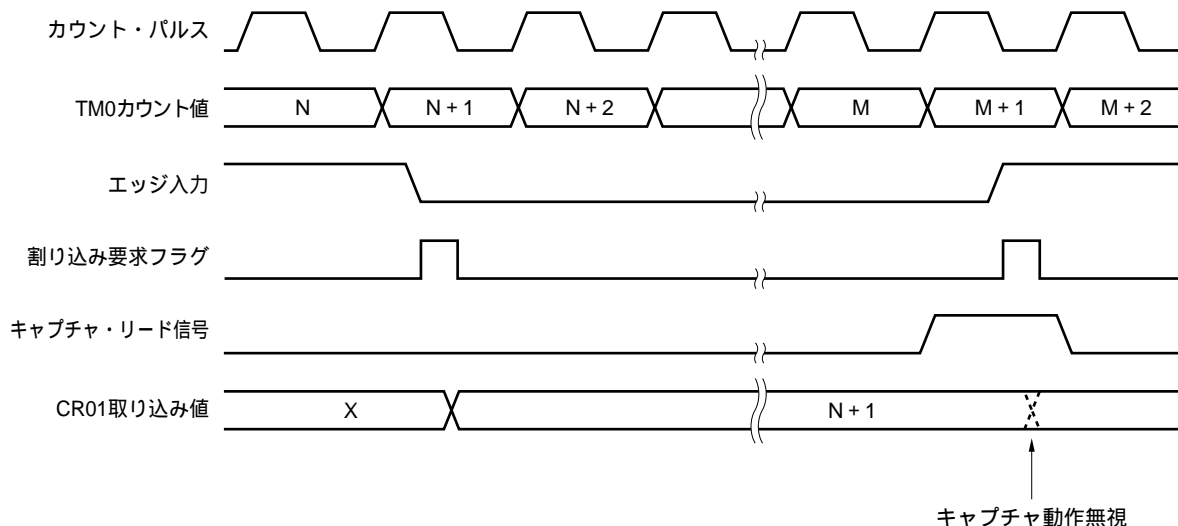


備考  $N > X > M$

(4) キャプチャ・レジスタのデータ保持タイミング

16ビット・キャプチャ/コンペア・レジスタ01(CR01)の読み出し中にTI00/P20端子の有効エッジが  
入力されたとき、CR01はキャプチャ動作を行わず、データを保持します。ただし、有効エッジの検出  
による割り込み要求フラグ(INTTM01)はセットされます。

図7 - 23 キャプチャ・レジスタのデータ保持タイミング



(5) 有効エッジの設定

TI00/P20端子の有効エッジの設定は、16ビット・タイマ・モード・コントロール・レジスタ(TMC0)  
のビット2, 3(TMC02, TMC03)に0, 0を設定し、タイマ動作を停止させたのちに行ってください。  
有効エッジの設定は、プリスケアラ・モード・レジスタ0のビット4, 5(ES00, ES01)で行います。

(6) OVF0フラグの動作

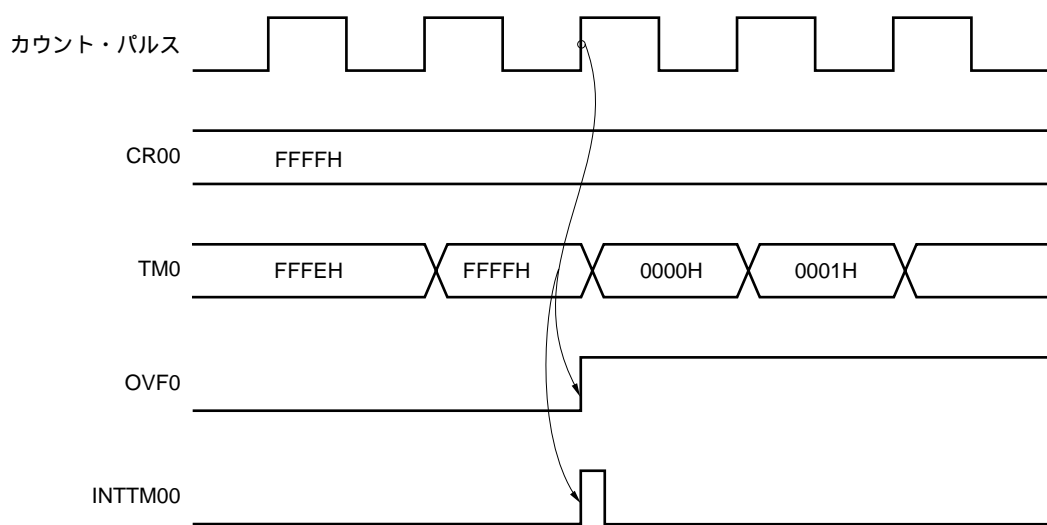
OVF0フラグは次のとき、1に設定されます。

TM0とCR00の一致でクリア&スタートするモードを選択

CR00をFFFFHに設定

TM0がFFFFHから0000Hにカウント・アップするとき

図7-24 OVF0フラグの動作のタイミング



(7) 競合動作について

16ビット・キャプチャ/コンペア・レジスタ(CR00, CR01)のリード期間とキャプチャ・トリガ入力の競合(CR00, CR01は、キャプチャ・レジスタとして使用)

キャプチャ・トリガ入力が優先されます。CR00, CR01のリード・データは不定となります。

16ビット・キャプチャ/コンペア・レジスタ(CR00, CR01)のライト期間と16ビット・タイマ・カウンタ0(TM0)との一致タイミングの競合(CR00, CR01はコンペア・レジスタとして使用)

一致判別は正常に行われません。一致タイミング付近で、CR00, CR01のライト動作は、行わないでください。

## 第8章 8ビットPWMタイマ

### 8.1 8ビットPWMタイマの機能

8ビットPWMタイマには、次の2つのモードがあります。

- ・ 8ビット・タイマ・カウンタ (TM5n : n=0, 1) を単体で使用するモード (単体モード)
- ・ カスケード接続して使用するモード (16ビット分解能 : カスケード接続モード)

次に、これら2つのモードについて説明します。

#### (1) TM5n (n=0, 1) を単体で使用するモード (単体モード)

8ビットのタイマ/イベント・カウンタとして動作します。

次のような機能として使用できます。

- ・ インターバル・タイマ
- ・ 外部イベント・カウンタ
- ・ 方形波出力
- ・ PWM出力

#### (2) カスケード接続して使用するモード (16ビット分解能 : カスケード接続モード)

カスケード接続することにより、16ビットのタイマ/イベント・カウンタとして動作します。

次のような機能として使用できます。

- ・ 16ビット分解能のインターバル・タイマ
- ・ 16ビット分解能の外部イベント・カウンタ
- ・ 16ビット分解能の方形波出力

## 8.2 8ビットPWMタイマの構成

8ビットPWMタイマは、次のハードウェアで構成されています。

表8-1 8ビットPWMタイマの構成

項目	構成
タイマ・カウンタ	8ビット・タイマ・カウンタ5n (TM5n)
レジスタ	8ビット・コンペア・レジスタ5n (CR5n)
タイマ出力 / 外部 クロック入力	TIO5n
制御レジスタ	タイマ・クロック選択レジスタ5n (TCL5n) 8ビット・タイマ・モード・コントロール・レジスタ5n (TMC5n)

n = 0, 1

図8-1 8ビットPWMタイマ50 (TM50) のブロック図

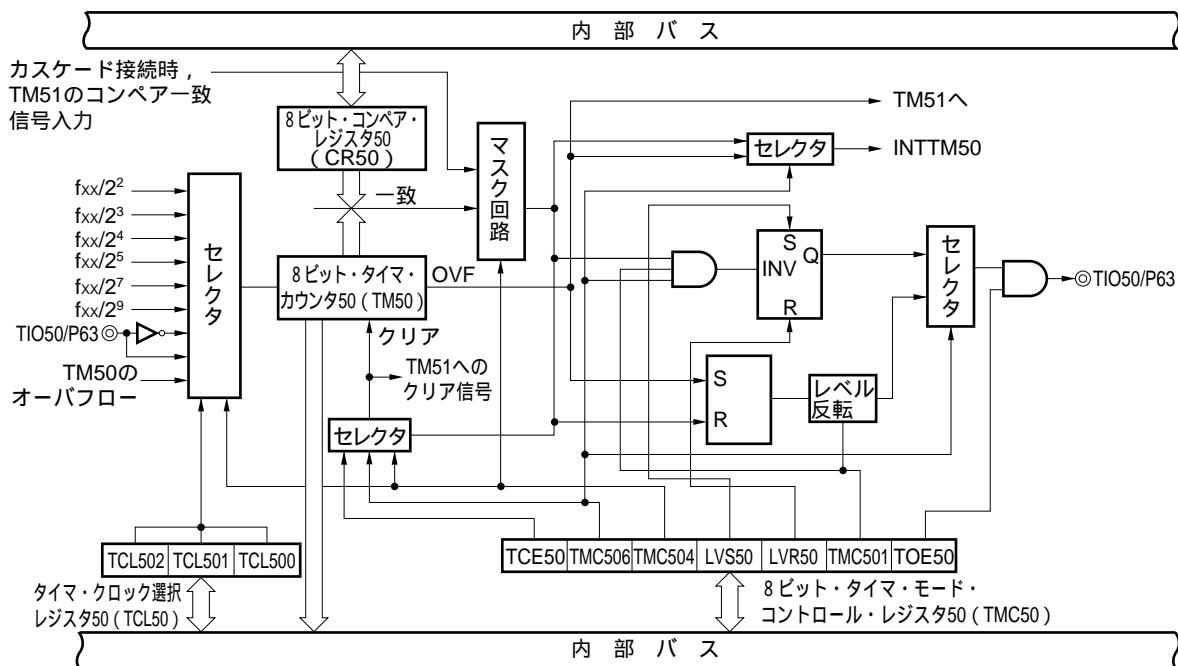
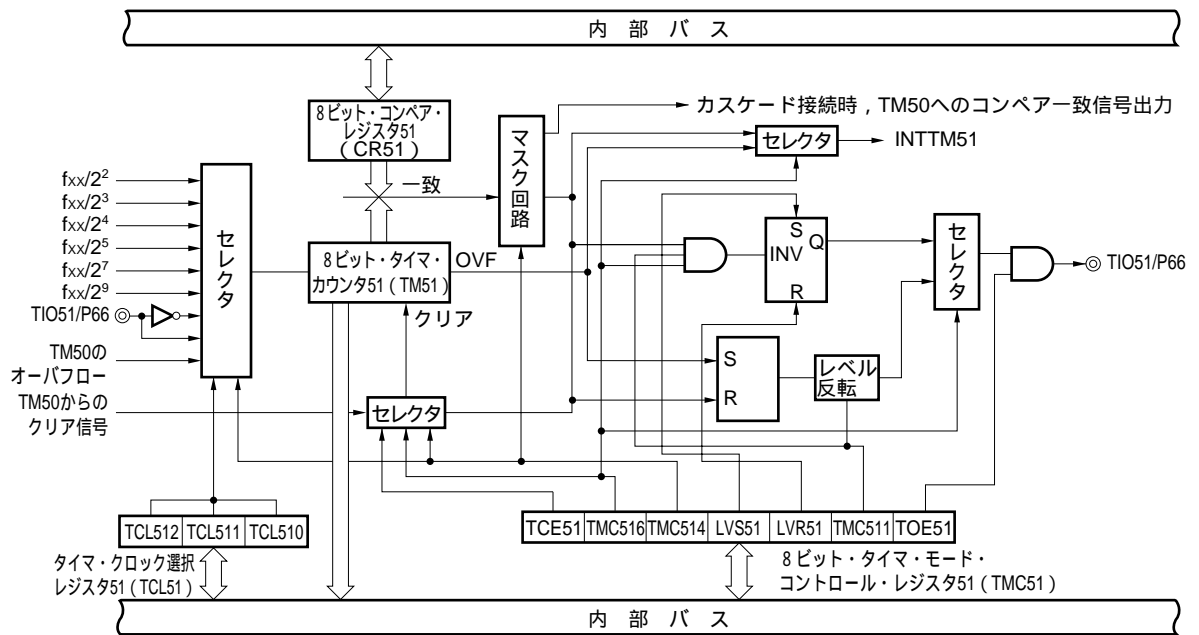


図8-2 8ビットPWMタイマ51 (TM51) のブロック図



**(1) 8ビット・タイマ・カウンタ<sub>5n</sub> (TM<sub>5n</sub> : n = 0, 1)**

TM<sub>5n</sub>は、カウント・パルスをカウントする8ビットのリード専用レジスタです。

カウント・クロックの立ち上がり同期して、カウンタをインクリメントします。また、動作中にカウント値を読み出した場合、カウント・クロックの入力を一時停止し、その時点でのカウント値を読み出します。次の場合、カウント値は00Hになります。

$\overline{\text{RESET}}$ 入力

TCE<sub>5n</sub>をクリア

TM<sub>5n</sub>とCR<sub>5n</sub>の一致でクリア & スタート・モード時のTM<sub>5n</sub>とCR<sub>5n</sub>の一致

**注意** カスケード接続時は、TM<sub>50</sub>のTCE<sub>50</sub>をクリアしても00Hとなります。

**備考** n = 0, 1

**(2) 8ビット・コンペア・レジスタ<sub>5n</sub> (CR<sub>5n</sub> : n = 0, 1)**

CR<sub>5n</sub>に設定した値と、8ビット・タイマ・カウンタ<sub>5n</sub> (TM<sub>5n</sub>) のカウント値を常に比較し、その2つの値が一致したときに、割り込み要求 (INTTM<sub>5n</sub>) を発生します (PWMモード以外)。

CR<sub>5n</sub>の値は、00H-FFHの範囲で設定でき、カウント動作中の書き換えが可能です。

**注意** カスケード接続時にデータを設定するときは、必ずタイマ動作を停止させてから行ってください。タイマ動作は、8ビット・タイマ・モード・コントロール・レジスタ<sub>50</sub> (TMC<sub>50</sub>) のビット7 (TCE<sub>50</sub>) とTMC<sub>51</sub>のビット7 (TCE<sub>51</sub>) の両方をクリアすることで停止します。

**備考** n = 0, 1



### 8.3 8ビットPWMタイマを制御するレジスタ

8ビットPWMタイマを制御するレジスタには、次の2種類があります。

- ・ タイマ・クロック選択レジスタ5n (TCL5n : n = 0, 1)
- ・ 8ビット・タイマ・モード・コントロール・レジスタ5n (TMC5n : n = 0, 1)

#### (1) タイマ・クロック選択レジスタ5n (TCL5n : n = 0, 1)

8ビット・タイマ・カウンタ5n (TM5n : n = 0, 1) のカウント・クロックおよびTIO5n入力の有効エッジを設定するレジスタです。

TCL5nは、1ビット・メモリ操作命令または、8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図8-3 タイマ・クロック選択レジスタ5n (TCL5n) のフォーマット

略号	7	6	5	4	3	2	1	0	アドレス	リセット時	R/W
TCL5n	0	0	0	0	0	TCL5n2	TCL5n1	TCL5n0	FF56H(TCL50), FF57H(TCL51)	00H	R/W

TCL5n2	TCL5n1	TCL5n0	カウント・クロックの選択
0	0	0	TIO5nの立ち下がりエッジ
0	0	1	TIO5nの立ち上がりエッジ
0	1	0	$f_{xx}/2^2$ (3.125 MHz)
0	1	1	$f_{xx}/2^3$ (1.56 MHz)
1	0	0	$f_{xx}/2^4$ (781 kHz)
1	0	1	$f_{xx}/2^5$ (391 kHz)
1	1	0	$f_{xx}/2^7$ (98 kHz)
1	1	1	$f_{xx}/2^9$ (24 kHz)

注意1 . TCL5nを同一データ以外に書き換える場合は、いったんタイマ動作を停止させてから行ってください。

2 . ビット3-ビット7には必ず“0”を設定してください。

備考1 . カスケード接続時、TM50以外はTCL512-TCL510の設定は無効になります。

2 . n = 0, 1

3 . f<sub>xx</sub> : メイン・システム・クロック周波数

4 . ( ) 内は、f<sub>xx</sub> = 12.5 MHz動作時。

(2) 8ビット・タイマ・モード・コントロール・レジスタ $5n$  (TMC $5n$  :  $n = 0, 1$ )

TMC $5n$ は、次の6種類の設定を行うレジスタです。

8ビット・タイマ・カウンタ $5n$  (TM $5n$  :  $n = 0, 1$ ) のカウント動作制御

8ビット・タイマ・カウンタ $5n$  (TM $5n$  :  $n = 0, 1$ ) の動作モードの選択

単体モード / カスケード接続モードの選択 (TMC51のみ)

タイマ出力F/F (フリップフロップ) の状態設定

タイマF/Fの制御またはPWM (フリーランニング) モード時のアクティブ・レベルの選択

タイマ出力の制御

TMC $5n$ は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、04Hになります。

図8-4 8ビット・タイマ・モード・コントロール・レジスタ5n (TMC5n) のフォーマット

略号	⑦	6	5	4	③	②	1	0	アドレス	リセット時	R/W
TMC5n	TCE5n	TMC5n6	0	TMC5n4	LVS5n	LVR5n	TMC5n1	TOE5n	FF54H(TMC50), FF55H(TMC51)	04H	R/W

TCE5n	TM5nのカウンタ動作制御
0	カウンタを0にクリア後, カウンタ動作禁止(プリスケアラ禁止)
1	カウンタ動作開始

TMC5n6	TM5nの動作モード選択
0	TM5nとCR5nの一致でクリア&スタート・モード
1	PWM(フリーランニング)モード

TMC5n4 <sup>1,2</sup>	単体モード/カスケード接続モードの選択
0	単体モード(8ビット・タイマで使用)
1	カスケード接続モード(TM50と接続し, 16ビット・タイマとして使用)

LVS5n	LVR5n	タイマ出力F/Fの状態設定
0	0	変化しない
0	1	タイマ出力F/Fをリセット(0)
1	0	タイマ出力F/Fをセット(1)
1	1	設定禁止

TMC5n1	PWMモード以外(TMC5n6=0)	PWMモード(TMC5n6=1)
	タイマF/Fの制御	アクティブ・レベルの選択
0	反転動作禁止	ハイ・アクティブ
1	反転動作許可	ロウ・アクティブ

TOE5n	タイマ出力の制御
0	出力禁止(ポート・モード)
1	出力許可

注1. TMC504は, 必ず0を書き込んでください。

2. TMC514は, フォーマットどおりにしたってください。

注意 TMC5n6によりTM5nの動作モード選択を行うとき, またはTMC514により単体モード/カスケード接続モードの選択を行うときは, タイマ動作を停止させてから行ってください。

備考1. PWMモード時は, TCE5n=0により, PWM出力はインアクティブ・レベルになります。

2. データ設定後にLVS5n, LVR5nを読み出すと, 0が読み出せます。

3. n=0, 1

## 8.4 8ビットPWMタイマの動作

### 8.4.1 インターバル・タイマ（8ビット動作）としての動作

8ビット・コンペア・レジスタ5n (CR5n) にあらかじめ設定したカウント値をインターバルとし、繰り返し割り込み要求を発生するインターバル・タイマとして動作します。

8ビット・タイマ・カウンタ5n (TM5n) のカウント値がCR5nに設定した値と一致したとき、TM5nの値を0にクリアしてカウントを継続すると同時に、割り込み要求信号 (INTTM5n) を発生します。

タイマ・クロック選択レジスタ5n (TCL5n) のビット0-ビット2 (TCL5n0-TCL5n2) でTM5nのカウント・クロックを選択できます。

**備考** n = 0, 1

#### [ 設定方法 ]

各レジスタの設定を行います。

- ・TCL5n : カウント・クロックの選択
- ・CR5n : コンペア値
- ・TMC5n : TM5nとCR5nの一致でクリア&スタート・モードを選択  
( TMC5n = 0000 x x x 0B x = don't care )

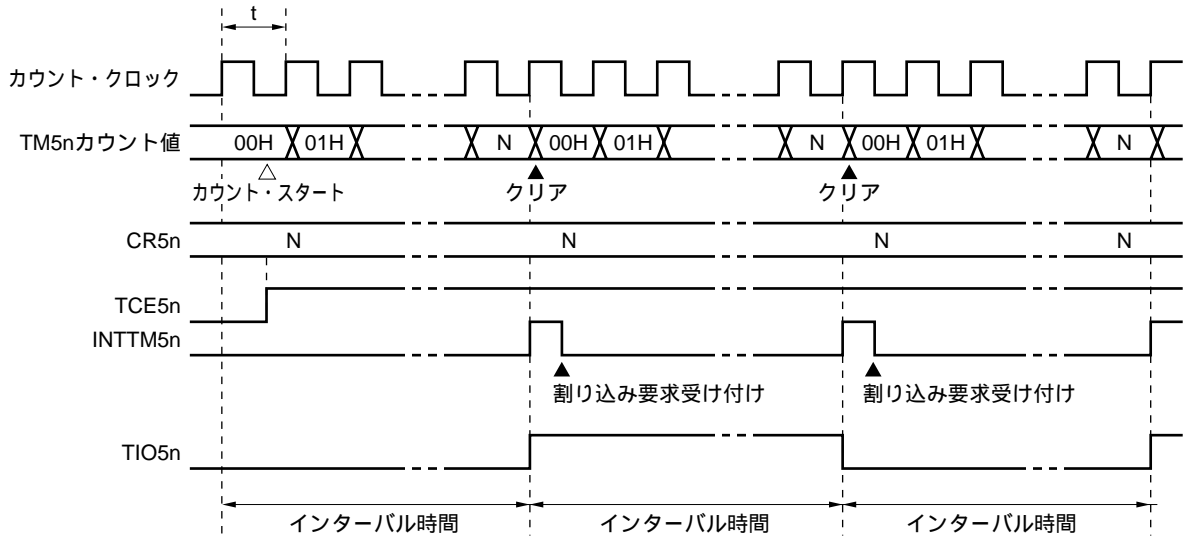
TCE5n = 1を設定すると、カウント動作を開始します。

TM5nとCR5nの値が一致すると、INTTM5nが発生します (TM5nは00Hにクリアされます)。以後、同一間隔でINTTM5nが繰り返し発生します。カウント動作を停止するときは、TCE5n = 0にしてください。

n = 0, 1

図8 - 5 インターバル・タイマ動作のタイミング (1/3)

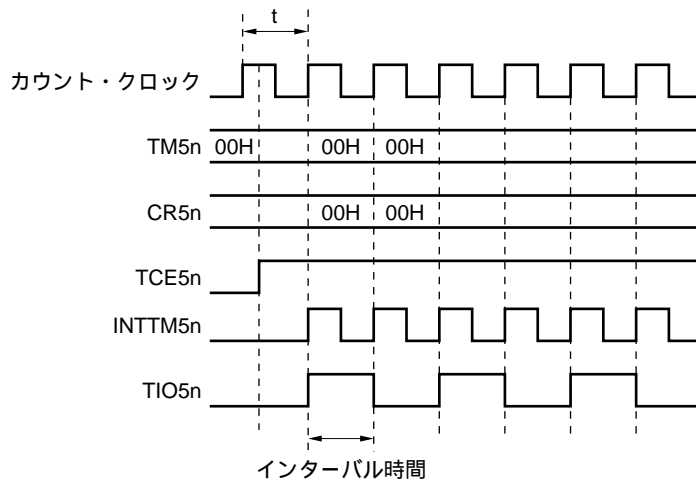
(a) 基本動作



備考1 . インターバル時間 =  $(n + 1) \times t$  : N = 00H-FFH

2 . n = 0, 1

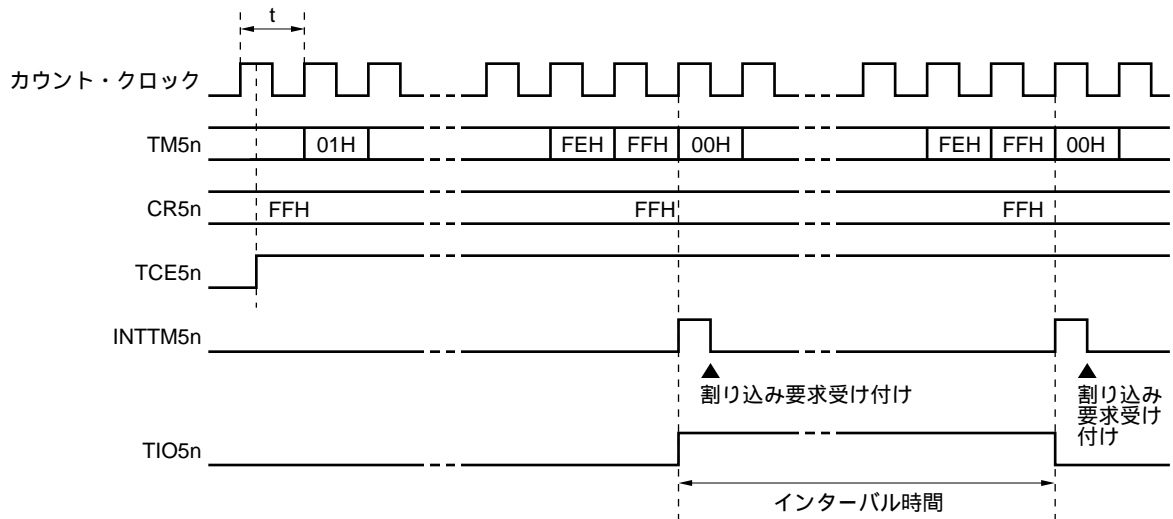
(b) CR5n = 00Hの場合



n = 0, 1

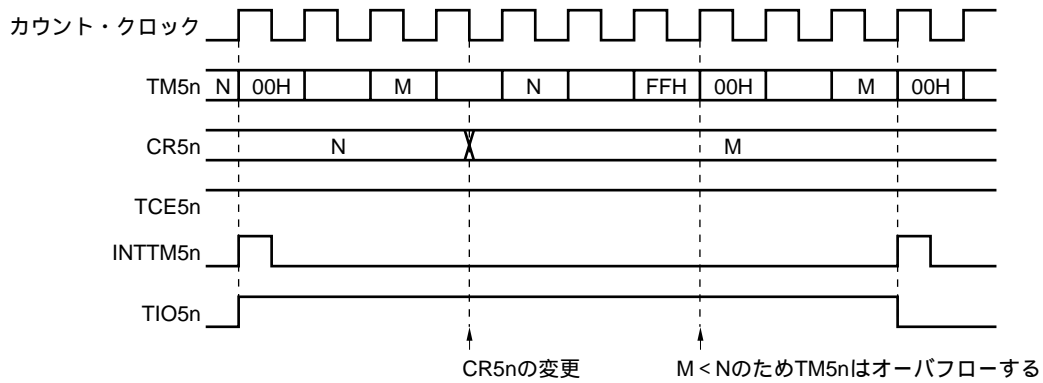
図8-5 インターバル・タイマ動作のタイミング(2/3)

(c) CR5n = FFHの場合



n = 0, 1

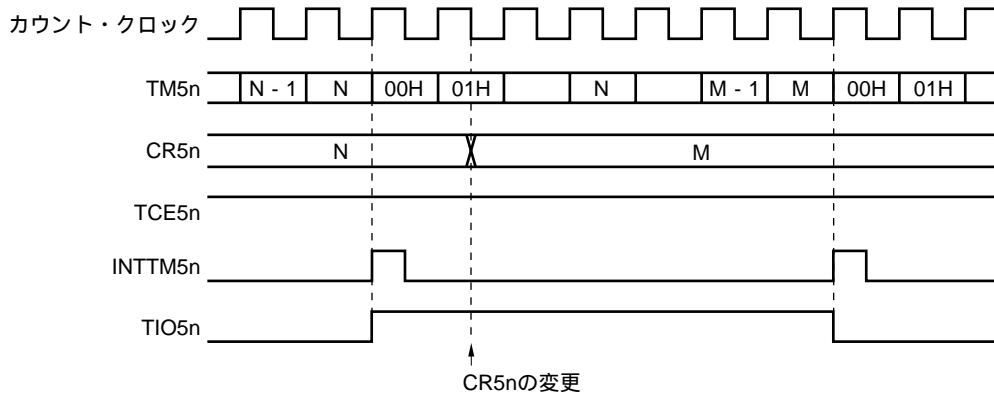
(d) CR5n変更による動作 (M < N)



n = 0, 1

図8 - 5 インターバル・タイマ動作のタイミング (3/3)

(e) CR5n変更による動作 (M > N)



n = 0, 1

### 8.4.2 外部イベント・カウンタとしての動作

外部イベント・カウンタは、TIO5nに入力される外部からのクロック・パルス数をカウントするものです。

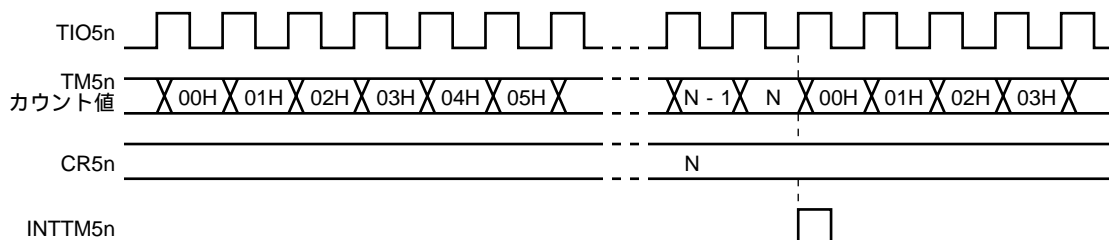
タイマ・クロック選択レジスタ5n (TCL5n) で指定した有効エッジが入力されるたびに、TM5nがインクリメントされます。エッジ指定は、立ち上がりまたは立ち下がりのいずれかを選択できます。

TM5nの計数値が8ビット・コンペア・レジスタ5n (CR5n) の値と一致すると、TM5nは0にクリアされ、割り込み要求信号 (INTTM5n) が発生します。

以後、TM5nの値とCR5nの値が一致するたびに、INTTM5nが発生します。

備考 n = 0, 1

図8 - 6 外部イベント・カウンタ動作のタイミング (立ち上がりエッジ指定時)



n = 0, 1

### 8.4.3 方形波出力（8ビット分解能）としての動作

8ビット・コンペア・レジスタ5n (CR5n) にあらかじめ設定した値をインターバルとする、任意の周波数の方形波出力として動作します。

8ビット・タイマ・モード・コントロール・レジスタ5n (TMC5n) のビット0 (TOE5n) に1を設定することにより、CR5nにあらかじめ設定したカウント値をインターバルとしてTIO5nの出力状態が反転します。これにより、任意の周波数の方形波出力（デューティ = 50 %）が可能です。

**備考** n = 0, 1

#### [ 設定方法 ]

各レジスタの設定を行います。

- ・タイマ出力端子と兼用しているポートのポート・ラッチ，ポート・モード・レジスタに “ 0 ” を設定
- ・TCL5n : カウント・クロックの選択
- ・CR5n : コンペア値
- ・TMC5n : TM5nとCR5nの一致でクリア&スタート・モード

LVS5n	LVR5n	タイマ出力F/Fの状態設定
1	0	ハイ・レベル出力
0	1	ロウ・レベル出力

タイマ出力F/Fの反転許可

タイマ出力許可 TOE5n = 1

TCE5n = 1を設定すると、カウント動作を開始します。

TM5nとCR5nの値が一致すると、タイマ出力F/Fが反転します。

また、INTTM5nが発生し、TM5nは00Hにクリアされます。

以後、同一間隔でタイマ出力F/Fが反転し、TIO5nから方形波が出力されます。

n = 0, 1



#### 8.4.4 8ビットPWM出力としての動作

8ビット・タイマ・モード・コントロール・レジスタ5n (TMC5n) のビット6 (TMC5n6) を“1”に設定することにより、PWM出力として動作します。

8ビット・コンペア・レジスタ5n (CR5n) に設定した値で決まるデューティ比のパルスを、TIO5nから出力します。

PWMパルスのアクティブ・レベルの幅は、CR5nに設定してください。また、アクティブ・レベルは、TMC5nのビット1 (TMC5n1) により選択できます。

カウント・クロックは、タイマ・クロック選択レジスタ5n (TCL5n) のビット0-ビット2 (TCL5n0-TCL5n2) で選択できます。

TMC5nのビット0 (TOE5n) により、PWM出力の許可/禁止が選択できます。

**注意** PWMモード時のCR5nの書き換えは、1周期に1回のみ可能です。

**備考** n = 0, 1

##### (1) PWM出力の基本動作

###### [ 設定方法 ]

タイマ出力端子と兼用しているポートのポート・ラッチ、ポート・モード・レジスタに“0”を設定します。

8ビット・コンペア・レジスタ5n (CR5n) でアクティブ・レベル幅を設定します。

タイマ・クロック選択レジスタ5n (TCL5n) で、カウント・クロックを選択します。

TMC5nのビット1 (TMC5n1) で、アクティブ・レベルを設定します。

TMC5nのビット0 (TOE5n) に“1”を設定し、タイマ出力を許可します。

TMC5nのビット7 (TCE5n) に“1”を設定すると、カウント動作を開始します。

カウント動作を停止するときは、TCE5nに“0”を設定してください。

n = 0, 1

###### [ PWM出力の動作 ]

カウント動作を開始すると、PWM出力 (TIO5nからの出力) はオーバフローが発生するまでインアクティブ・レベルを出力します。

オーバフローが発生すると、アクティブ・レベルを出力します。アクティブ・レベルは、CR5nと8ビット・カウンタ5n (TM5n) のカウント値が一致するまで出力されます。

CR5nとカウント値が一致したあとのPWM出力は、再度オーバフローが発生するまでインアクティブ・レベルを出力します。

以後、カウント動作が停止されるまで、を繰り返します。

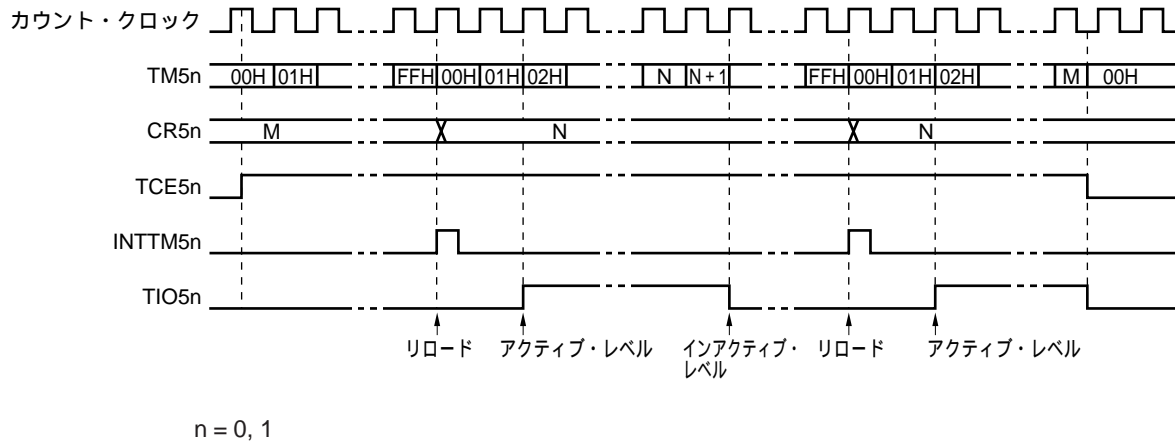
TCE5n = 0によりカウント動作を停止すると、PWM出力はインアクティブ・レベルになります。

n = 0, 1

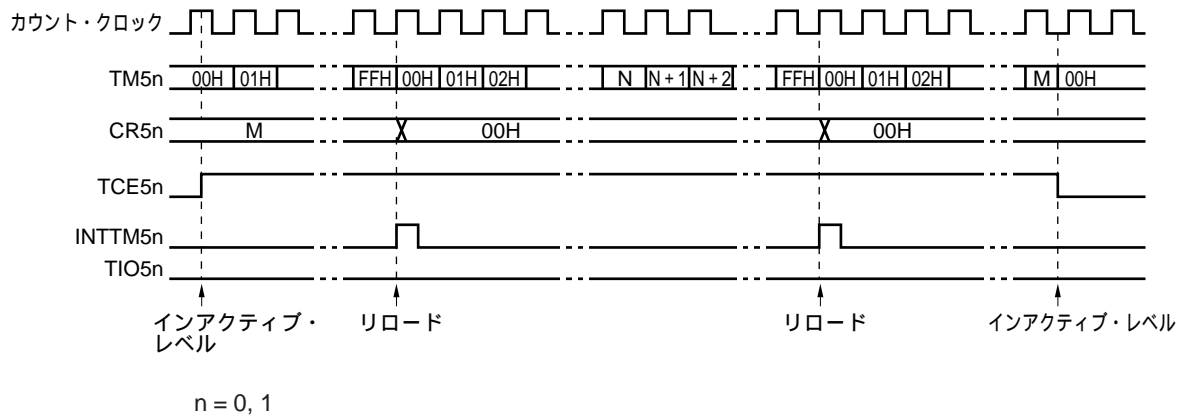
(a) PWM出力の基本動作

図8-7 PWM出力の動作タイミング

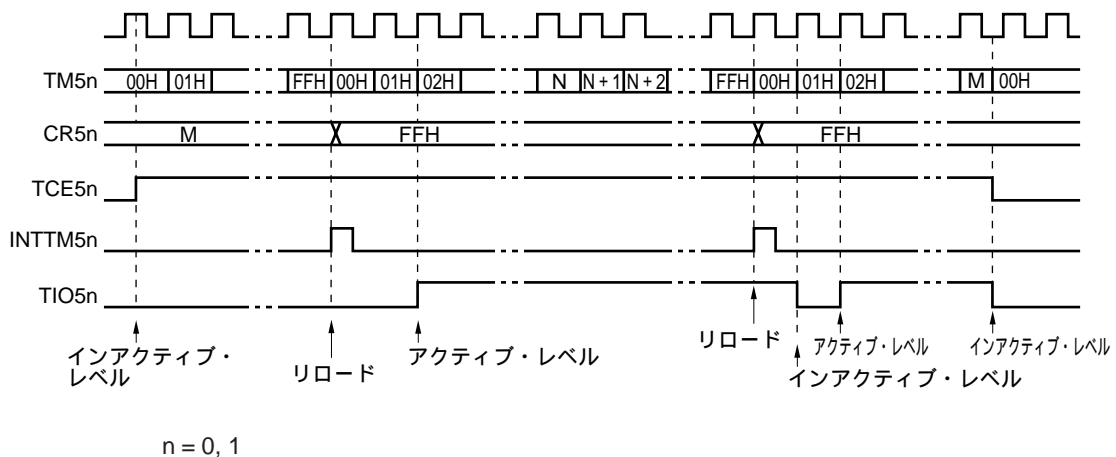
(i) 基本動作 (アクティブ・レベル=Hのとき)



(ii) CR5n = 0の場合



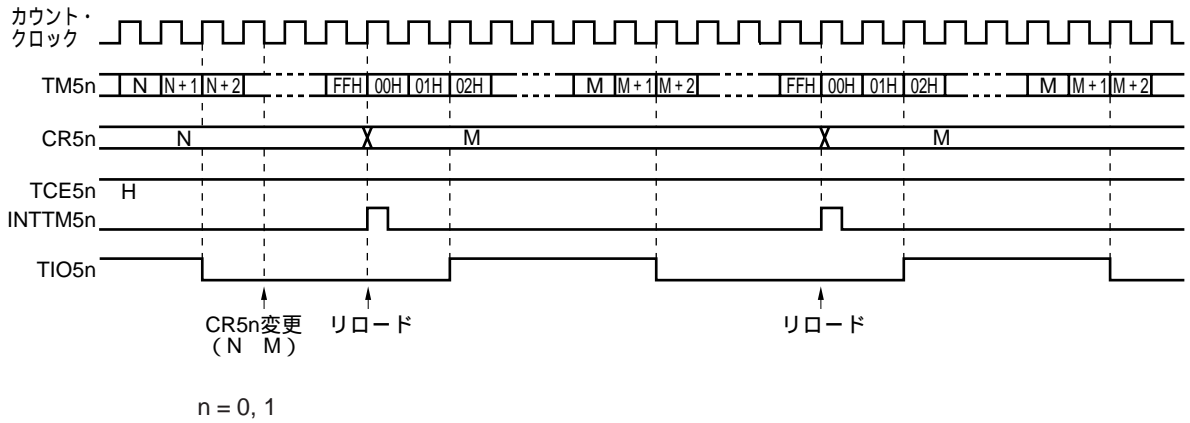
(iii) CR5n = FFHの場合



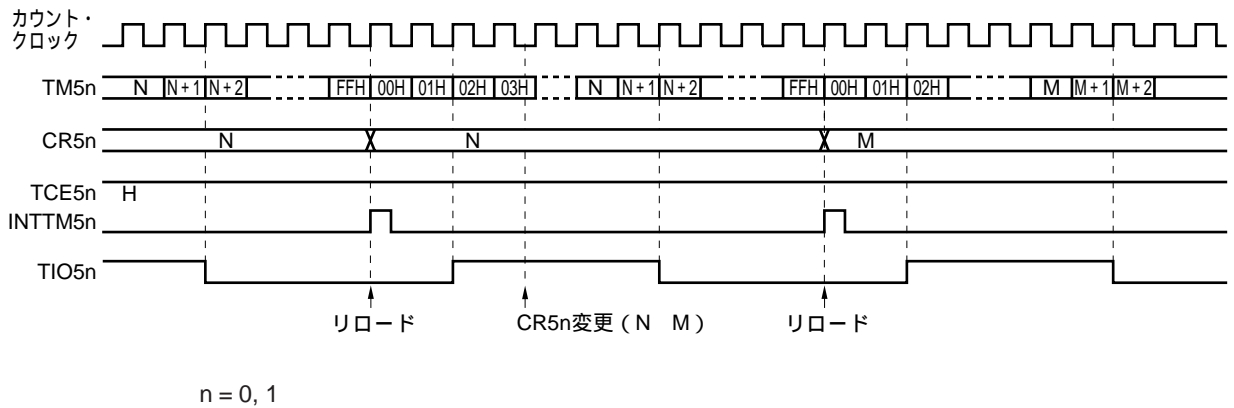
(b) CR5n変更による動作について

図8-8 CR5n変更による動作のタイミング

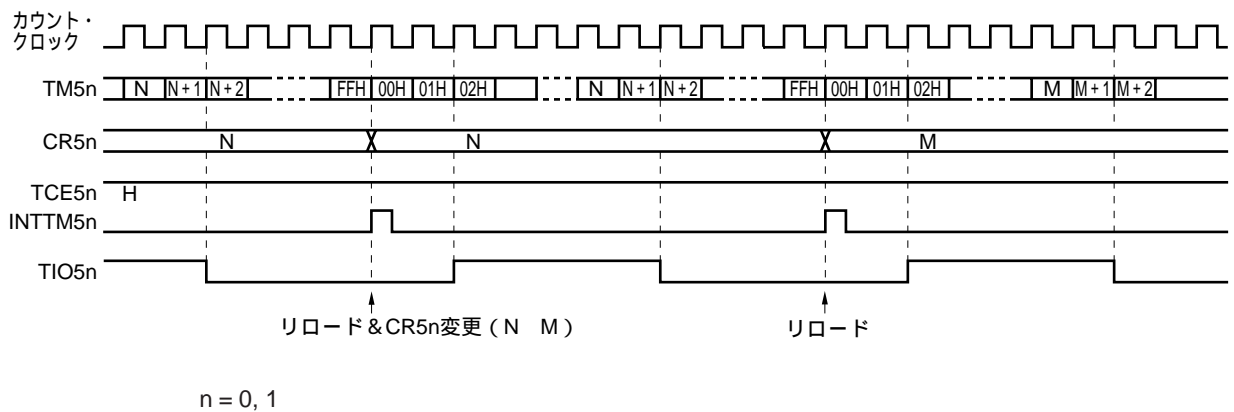
(i) CR5nの値をTM5nのオーバーフロー前にN Mに変更した場合



(ii) CR5nの値をTM5nのオーバーフロー後にN Mに変更した場合



(iii) CR5nの値をTM5nのオーバーフロー直後の2クロック間(00H, 01H)にN Mに変更した場合



## (2) カスケード接続(16ビット・タイマ)モード

## ・インターバル・タイマ(16ビット分解能)としての動作

8ビット・タイマ・モード・コントロール・レジスタ51(TMC51)のビット4(TMC514)に“1”を設定することにより、16ビット分解能のタイマ/カウンタ・モードになります。

8ビット・コンペア・レジスタ51(CR51)にあらかじめ設定したカウント値をインターバルとし、繰り返し割り込み要求を発生するインターバル・タイマとして動作します。

## [設定方法]

各レジスタの設定を行います。

- ・TCL5n : TM50はカウント・クロックの選択  
カスケード接続するTM51は設定不要
- ・CR5n : コンペア値(各コンペア値とも00H-FFHの設定が可能)
- ・TMC5n : TM5nとCR5nの一致でクリア&スタート・モードを選択

$$\left[ \begin{array}{ll} \text{TM50} & \text{TMC50} = 0000 \times \times \times 0\text{B} \quad \times : \text{don't care} \\ \text{TM51} & \text{TMC51} = 0001 \times \times \times 0\text{B} \quad \times : \text{don't care} \end{array} \right]$$

TM51からTCE51 = 1に設定し、最後にTM50のTCE50 = 1に設定することにより、カウント動作を開始します。

カスケード接続された全タイマのTM5nとCR5nの値が一致すると、TM50のINTTM50が発生します(すべてのTM5nは00Hにクリアされます)。

以後、同一間隔でINTTM50が繰り返し発生します。

**注意1** . 8ビット・コンペア・レジスタ5n(CR5n)の設定は、必ずタイマ動作を停止させてから行ってください。

2 . カスケード接続で使用している場合でも、TM51のカウント値がCR51と一致すると、TM51のINTTM51が発生してしまいます。TM51は、割り込み禁止のため必ずマスクしてください。

3 . TCE5nのセットは、TM51のTCE51から行い、最後にTM50のTCE50をセットしてください。

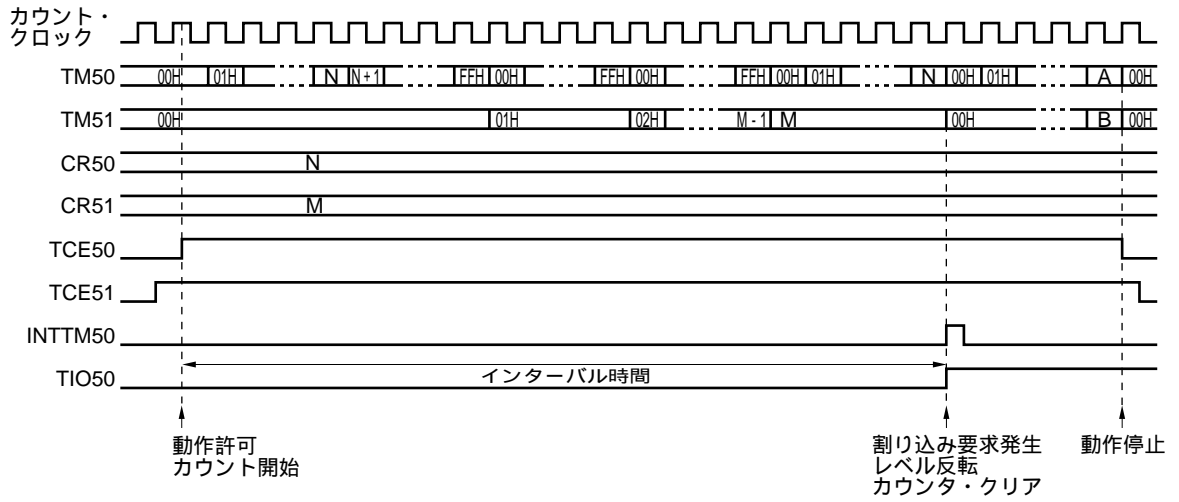
4 . カウントの再スタート/ストップは、TM50のTCE50のみ1/0に設定することにより、動作/停止させることができます。

ただし、8ビット・コンペア・レジスタ5n(CR5n)の設定を行う場合は、必ずTM50のビット7(TCE50)とTM51のビット7(TCE51)をクリアしてください。

備考 n = 0, 1

図8 - 9に、16ビット分解能カスケード接続モードのタイミング例を示します。

図8 - 9 16ビット分解能カスケード接続モード

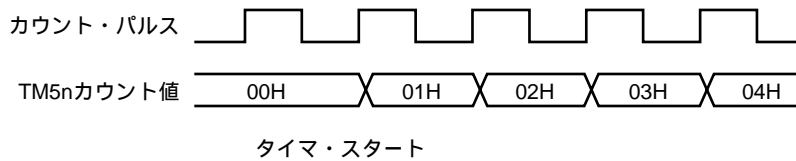


## 8.5 8ビットPWMタイマの注意事項

### (1) タイマ・スタート時の誤差

タイマ・スタート後，一致信号が発生するまでの時間は，最大で1クロック分の誤差が生じます。これは，カウント・パルスに対して8ビット・カウンタ $n$  (TM5 $n$  :  $n=0, 1$ ) のスタートが非同期で行われるためです。

図8 - 10 8ビット・カウンタ $5n$  (TM5 $n$ ) のスタート・タイミング



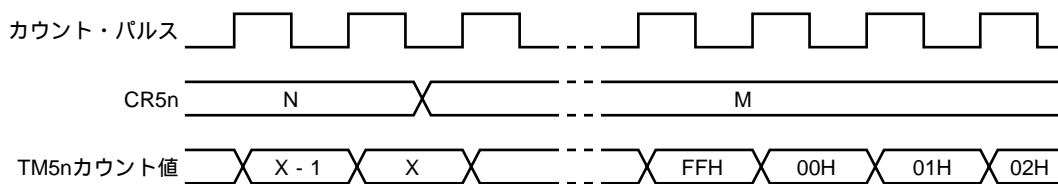
$n=0, 1$

### (2) タイマ・カウント動作中のコンペア・レジスタの変更後の動作

8ビット・コンペア・レジスタ $5n$  (CR5 $n$ ) の変更後の値が，8ビット・タイマ・カウンタ $5n$  (TM5 $n$ ) の値よりも小さいときはカウントを継続し，オーバーフローして0から再カウントします。したがって，CR5 $n$ の変更後の値 ( $M$ ) が，変更前の値 ( $N$ ) より小さいときは，CR5 $n$ を変更したあと，タイマを再スタートさせる必要があります。

備考  $n=0, 1$

図8 - 11 タイマ・カウント動作中のコンペア・レジスタの変更後のタイミング



$N > X > M$

$n=0, 1$

**注意** TIO5 $n$ 入力を選択している場合を除き，ストップ状態に設定する前は必ずTCE5 $n=0$ にしてください ( $n=0, 1$ )。

(3) タイマ動作中のTM5n読み出しについて

動作中のTM5n ( $n=0, 1$ ) の読み出しは、選択クロックを一時停止して読み出すため、選択クロックはより長いハイ/ロウ・レベルのある波形を選択してください。

また、カスケード接続時にTM5n ( $n=0, 1$ ) の読み出しを行うときは、カウント変化中の読み出しを考慮しソフトウェアで2度読み、一致などの対策を行ってください。

〔メモ〕



## 第9章 ウォッチドッグ・タイマ

ウォッチドッグ・タイマは、プログラムの暴走を検出するためのタイマです。

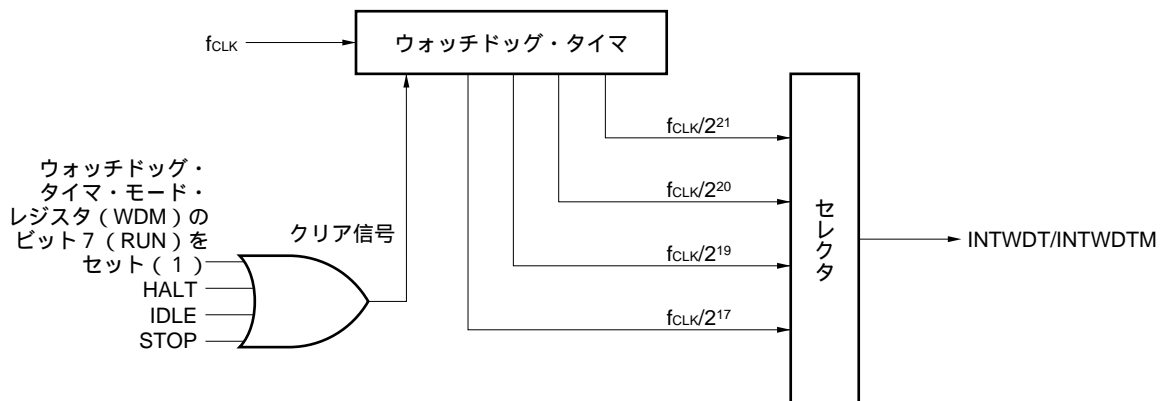
ウォッチドッグ・タイマ割り込みの発生で、プログラムまたはシステムが異常であることを検出します。そのため、プログラムの各所に、一定時間以内にウォッチドッグ・タイマをクリア（カウント開始）する命令を入れておきます。

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・タイマがオーパフローすると、ウォッチドッグ・タイマ割り込み（INTWDT）が発生し、プログラムの異常を知らせます。

### 9.1 構成

図9 - 1に、ウォッチドッグ・タイマのブロック図を示します。

図9 - 1 ウォッチドッグ・タイマのブロック図



**注意 1.** ウォッチドッグ・タイマ動作中に、スタンバイ・モード (HALT/STOP/IDLE) に設定した場合、ウォッチドッグ・タイマはクリアされ停止します。スタンバイ・モード解除要求が発生すると、HALT/IDLEモードは解除要求発生後にウォッチドッグ・タイマ動作を開始します。STOPモードは、解除要求発生後に発振安定時間を待ってウォッチドッグ・タイマ動作を開始します。

**2.** INTWDTはノンマスクابل割り込み、INTWDTMはマスクابل割り込みです。ウォッチドッグ・タイマの割り込みをノンマスクابل割り込みとして使用するかマスクابل割り込みとして使用するかは、割り込み選択コントロール・レジスタ (SNMI) のビット 1 (SWDT) の設定で選択することができます。SNMIに関しては、第12章 割り込み機能12.4 (6) を参照してください。

**備考** fCLK : 内部システム・クロック ( $f_{xx} \sim f_{xx}/8$ )

## 9.2 制御レジスタ

### ウォッチドッグ・タイマ・モード・レジスタ (WDM)

WDMは、ウォッチドッグ・タイマの動作を制御する8ビット・レジスタです。

プログラムの暴走によってウォッチドッグ・タイマが誤ってクリアされないように、専用命令によってのみ書き込みができます。この専用命令は、MOV WDM, byte命令で特殊なコード構成(4バイト)になっています。3バイト目と4バイト目のオペコードが互いに1の補数の場合のみ、書き込みが行われます。

また、3バイト目と4バイト目のオペコードが互いに1の補数でなければ書き込みが行われず、オペランド・エラー割り込みが発生します。この場合、スタック領域に退避されるリターン・アドレスは、エラーの原因となった命令のアドレスです。したがって、スタック領域に退避されたリターン・アドレスにより、エラーの原因となったアドレスを知ることができます。

なお、単純にオペランド・エラーからRETB命令で復帰を行うと、無限ループとなります。

オペランド・エラー割り込みは、プログラムが暴走している場合にのみ発生する(NEC製アセンブラRA78K4では、MOV WDM, byteと記述された場合、正しい専用命令しか生成しません)ので、システムの初期化をプログラムで行うようにしてください。

他の書き込み命令(MOV WDM, AやAND WDM, byteやSET1 WDM.7など)は無視され、何の動作も行いません。すなわち、WDMへの書き込みは行われず、また、オペランド・エラー割り込みのような割り込みも発生しません。

システム・リセット( $\overline{\text{RESET}}$ 入力)後、いったんウォッチドッグ・タイマをスタートさせると(RUNビットをセット(1)すると)、WDMの内容を変更できません。ウォッチドッグ・タイマを停止させることができるのはリセットのみです。ウォッチドッグ・タイマのクリアは、専用命令によりいつでもできます。

WDMの読み出しは、8ビット・データ転送命令によりいつでもできます。

$\overline{\text{RESET}}$ 入力により、WDMは00Hになります。

図9-2に、WDMのフォーマットを示します。

図9 - 2 ウォッチドッグ・タイマ・モード・レジスタ (WDM) のフォーマット

アドレス : 0FFC2H リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
WDM	RUN	0	0	0	0	WDT2	WDT1	0

RUN	ウォッチドッグ・タイマの動作の指定
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

WDT2	WDT1	カウント・クロック	オーバフロー時間 [ms] ( $f_{CLK} = 12.5 \text{ MHz}$ )
0	0	$f_{CLK}/2^{17}$	10.5
0	1	$f_{CLK}/2^{19}$	41.9
1	0	$f_{CLK}/2^{20}$	83.9
1	1	$f_{CLK}/2^{21}$	167.8

- 注意 1 .** ウォッチドッグ・タイマ・モード・レジスタ (WDM) は、専用命令 (MOV WDM, #byte) でのみ書き込みが可能です。
- 2 .** RUNビットをセット (1) するためのWDMへの書き込みには、毎回、同じ値を書き込んでください。異なる値を書いても1回目に書き込んだ内容を変更できません。
- 3 .** RUNビットは、いったんセット (1) するとソフトウェアではリセット (0) できません。

**備考**  $f_{CLK}$  : 内部システム・クロック ( $f_{xx} \sim f_{xx}/8$ )  
 $f_{xx}$  : メイン・システム・クロック周波数

## 9.3 動作

ウォッチドッグ・タイマは、ウォッチドッグ・タイマ・モード・レジスタ (WDM) のRUNビットをセット (1) することによりクリアされ、カウント動作を開始します。RUNビットをセット (1) したあと、WDM のWDT2, WDT1ビットで指定されたオーバフロー時間が経過すると、ノンマスクابلな割り込みである割り込み (INTWDT) を発生します。

オーバフロー時間が経過する前にRUNビットを再度セット (1) すると、ウォッチドッグ・タイマはクリアされ、再度カウント動作を開始します。

## 9.4 注意事項

### 9.4.1 ウォッチドッグ・タイマ使用時の一般的な注意事項

(1) ウォッチドッグ・タイマは、暴走を検出するための手段の1つですが、すべての暴走を検出できるわけではありません。したがって、特に信頼性の要求される装置では、内蔵のウォッチドッグ・タイマだけでなく、外付けの回路により暴走を早期に検出し、正常状態に復帰、または安全な状態にして動作を停止させるなどの処理ができるようにする必要があります。

(2) 次のような場合、ウォッチドッグ・タイマは暴走を検出できません。

ウォッチドッグ・タイマのクリアを、タイマ割り込み処理プログラム内で行っている場合  
 割り込み要求およびマクロ・サービスが一時的に保留される場合 (14.9 割り込み要求およびマクロ・サービスが一時的に保留される場合参照) が連続して発生している場合  
 プログラムの論理的な誤りにより暴走している場合 (プログラムの各モジュールは正常に動作しているが、全体として正常に動作していない場合) で、ウォッチドッグ・タイマを定期的クリアしている場合  
 暴走時に実行している命令群で、定期的にウォッチドッグ・タイマをクリアしている場合  
 暴走の結果、STOPモード、HALTモードまたはIDLEモードになってしまった場合  
 CPUが外来ノイズで暴走したときにウォッチドッグ・タイマも暴走してしまった場合

、 の場合は、プログラムを修正することで検出を可能にすることができます。

の場合は、ウォッチドッグ・タイマがクリアできるのは4バイトの専用命令だけです。 も同様に4バイトの専用命令でなければSTOPモード、HALTモードまたはIDLEモードにすることができません。また、暴走の結果、 の状態になるためには、3バイト以上の連続したデータが特定パターン (例 BT PSWL. bit, \$\$など) になっている必要があります。したがって、 および暴走の結果、 の状態になることが発生するのはきわめてまれであると考えられます。

## 9.4.2 $\mu$ PD784976サブシリーズのウォッチドッグ・タイマに関する 注意事項

- (1) ウォッチドッグ・タイマ・モード・レジスタ (WDM) は、専用命令 (MOV WDM, byte) でのみ書き込みが可能です。
- (2) RUNビットをセット (1) するためのウォッチドッグ・タイマ・モード・レジスタ (WDM) への書き込みには、毎回、同じ値を書き込んでください。異なる値を書いても1回目に書き込んだ内容を変更できません。
- (3) RUNビットは、いったんセット (1) するとソフトウェアではリセット (0) できません。

〔メモ〕

## 第10章 A/Dコンバータ

### 10.1 A/Dコンバータの機能

A/Dコンバータは、アナログ入力をデジタル値に変換する8ビット分解能コンバータで、12チャンネル (ANI0-ANI11) のアナログ入力を制御できる構成になっています。

A/D変換動作の起動方法は、ソフトウェア・スタートのみです。

アナログ入力をANI0-ANI11から1チャンネル選択し、A/D変換を行ってください。A/D変換の動作は繰り返し行い、A/D変換を1回終了するたびに割り込み要求 (INTAD) が発生されます。

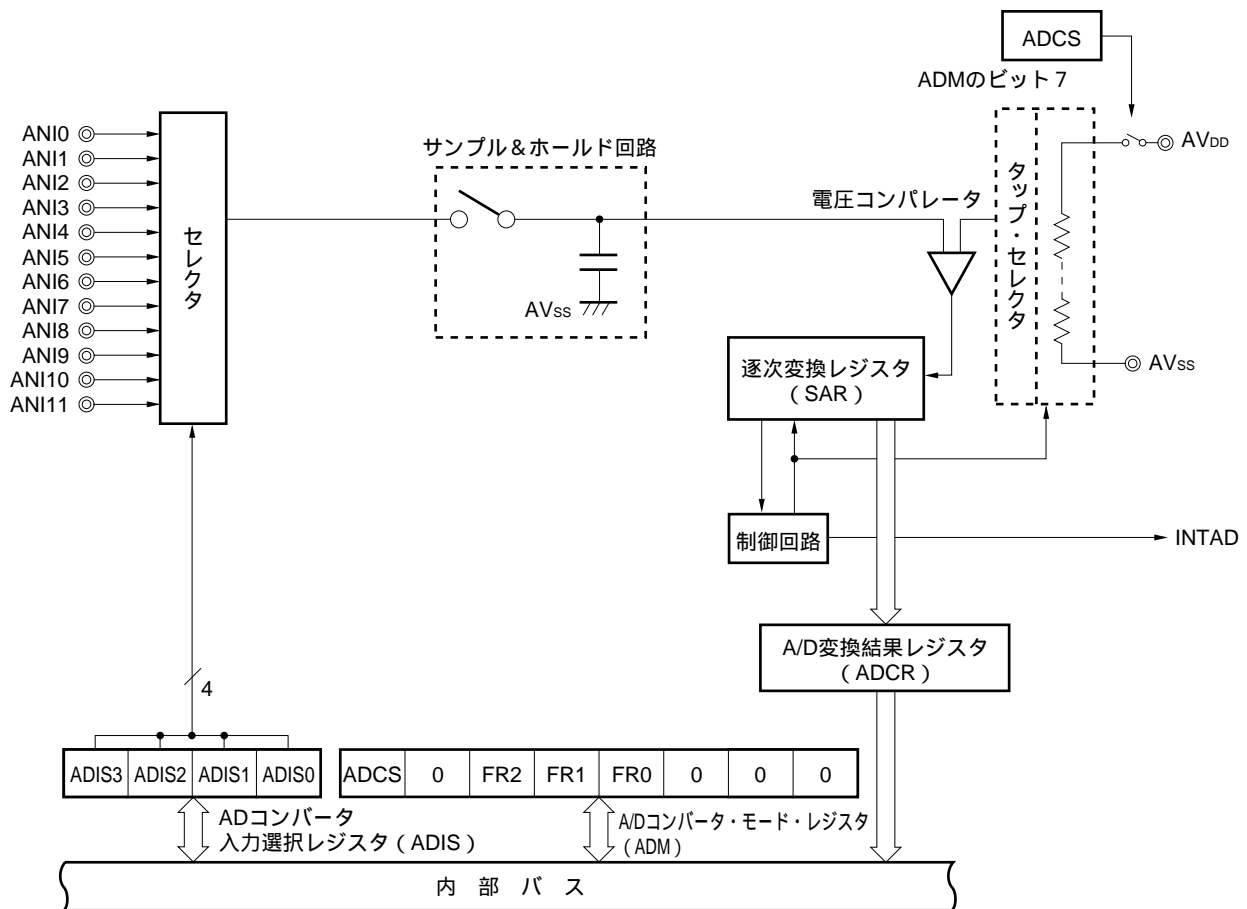
### 10.2 A/Dコンバータの構成

A/Dコンバータは、次のハードウェアで構成しています。

表10 - 1 A/Dコンバータの構成

項 目	構 成
アナログ入力	12チャンネル (ANI0-ANI11)
レジスタ	逐次変換レジスタ (SAR) A/D変換結果レジスタ (ADCR)
制御レジスタ	A/Dコンバータ・モード・レジスタ (ADM) A/Dコンバータ入力選択レジスタ (ADIS)

図10 - 1 A/Dコンバータのブロック図



**(1) 逐次変換レジスタ (SAR)**

アナログ入力の電圧値と直列抵抗ストリングからの電圧タップ (比較電圧) の値を比較し、その結果を最上位ビット (MSB) から保持するレジスタです。

最下位ビット (LSB) まで保持すると (A/D変換終了)、SARの内容はA/D変換結果レジスタ (ADCR) に転送されます。

**(2) A/D変換結果レジスタ (ADCR)**

A/D変換結果を保持します。A/D変換が終了するたびに、逐次変換レジスタ (SAR) から変換結果がロードされます。

ADCRは、8ビット・メモリ操作命令で読み出します。

RESET入力により、不定になります。

**(3) サンプル&ホールド回路**

サンプル&ホールド回路は、入力回路から順次送られてくるアナログ入力信号を1つ1つサンプリングし、電圧コンパレータに送ります。また、そのサンプリングしたアナログ入力電圧値をA/D変換中は保持します。



**(4) 電圧コンパレータ**

電圧コンパレータは、アナログ入力と直列抵抗ストリングの出力電圧を比較します。

**(5) 直列抵抗ストリング**

直列抵抗ストリングは、 $AV_{DD}$ - $AV_{SS}$ 間に接続されており、アナログ入力と比較する電圧を発生します。

**(6) ANI0-ANI11端子**

A/Dコンバータへの12チャンネルのアナログ入力端子です。A/D変換するアナログ信号を入力します。

**注意** ANI0-ANI11入力電圧は規格の範囲内でご使用ください。特に $AV_{DD}$ 以上、 $AV_{SS}$ 以下（絶対最大定格の範囲内でも）の電圧が入力されると、そのチャンネルの変換値が不定となり、またほかのチャンネルの変換値にも影響を与えることがあります。

**(7)  $AV_{SS}$ 端子**

A/Dコンバータのグランド電位端子です。A/Dコンバータを使用しないときでも、常に $V_{SS1}$ 端子と同電位で使用してください。

**(8)  $AV_{DD}$ 端子**

A/Dコンバータのアナログ電源端子です。A/Dコンバータを使用しないときでも、常に $V_{DD1}$ 端子と同電位で使用してください。

スタンバイ・モード時には、変換動作停止（A/Dコンバータ・モード・レジスタ（ADM）のビット7（ADCS）= 0）にすることにより、直列抵抗ストリングに流れる電流を低減できます。

### 10.3 A/Dコンバータを制御するレジスタ

A/Dコンバータを制御するレジスタには、次の2種類があります。

- ・ A/Dコンバータ・モード・レジスタ (ADM)
- ・ A/Dコンバータ入力選択レジスタ (ADIS)

#### (1) A/Dコンバータ・モード・レジスタ (ADM)

A/D変換するアナログ入力の変換時間、変換動作の開始 / 停止を設定するレジスタです。

ADMは、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

RESET入力により、00Hになります。

図10 - 2 A/Dコンバータ・モード・レジスタのフォーマット

略号	⑦	6	5	4	3	2	1	0	アドレス	リセット時	R/W
ADM	ADCS	0	FR2	FR1	FR0	0	0	0	FF80H	00H	R/W

ADCS	A/D変換動作の制御	
0	変換動作停止	
1	変換動作許可	

FR2	FR1	FR0	A/D変換時間の選択 <sup>注1</sup>		
			クロック数	f <sub>xx</sub> = 12.5 MHz動作時	f <sub>xx</sub> = 6.2 MHz動作時
0	0	0	144/f <sub>xx</sub>	設定禁止	23.0 μs
0	0	1	120/f <sub>xx</sub>		19.2 μs
0	1	0	96/f <sub>xx</sub>		15.4 μs
1	0	0	288/f <sub>xx</sub>	23.0 μs	46.1 μs
1	0	1	240/f <sub>xx</sub>	19.2 μs	38.4 μs
1	1	0	192/f <sub>xx</sub>	15.4 μs	30.7 μs
上記以外			-	設定禁止	

注1 . A/D変換時間が14 μs以上になるように設定してください。

- 2 . FR0-FR2を同一データ以外に書き換える場合は、いったんA/D変換動作を停止させたのちに行ってください。

**注意** ビット7 (ADCS) をセット直後の変換結果は不定になります。

**備考** f<sub>xx</sub> : メイン・システム・クロック周波数

(2) A/Dコンバータ入力選択レジスタ (ADIS)

A/D変換するアナログ電圧の入力ポートを指定するレジスタです。

ADS0は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図10 - 3 A/Dコンバータ入力選択レジスタのフォーマット

略号	7	6	5	4	3	2	1	0	アドレス	リセット時	R/W
ADIS	0	0	0	0	ADIS3	ADIS2	ADIS1	ADIS0	FF81H	00H	R/W

ADIS3	ADIS2	ADIS1	ADIS0	アナログ入力チャネルの指定
0	0	0	0	ANI0
0	0	0	1	ANI1
0	0	1	0	ANI2
0	0	1	1	ANI3
0	1	0	0	ANI4
0	1	0	1	ANI5
0	1	1	0	ANI6
0	1	1	1	ANI7
1	0	0	0	ANI8
1	0	0	1	ANI9
1	0	1	0	ANI10
1	0	1	1	ANI11
上記以外				設定禁止

## 10.4 A/Dコンバータの動作

### 10.4.1 A/Dコンバータの基本動作

A/D変換するチャンネルをA/Dコンバータ入力選択レジスタ (ADIS) で1チャンネル選択してください。

選択されたアナログ入力チャンネルに入力されている電圧を、サンプル&ホールド回路がサンプリングします。

一定時間サンプリングを行うとサンプル&ホールド回路はホールド状態となり、入力されたアナログ電圧をA/D変換が終了するまで保持します。

逐次変換レジスタ (SAR) のビット7がセットされます。タップ・セレクトにより直列抵抗ストリングの電圧タップが  $(1/2) AV_{DD}$  にされます。

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が電圧コンパレータで比較されます。もし、アナログ入力が  $(1/2) AV_{DD}$  よりも大きければ、SARのMSBはセットされたままです。また、 $(1/2) AV_{DD}$  よりも小さければMSBはリセットされます。

次にSARのビット6が自動的にセットされ、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 :  $(3/4) AV_{DD}$
- ・ビット7 = 0 :  $(1/4) AV_{DD}$

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のビット6が次のように操作されず。

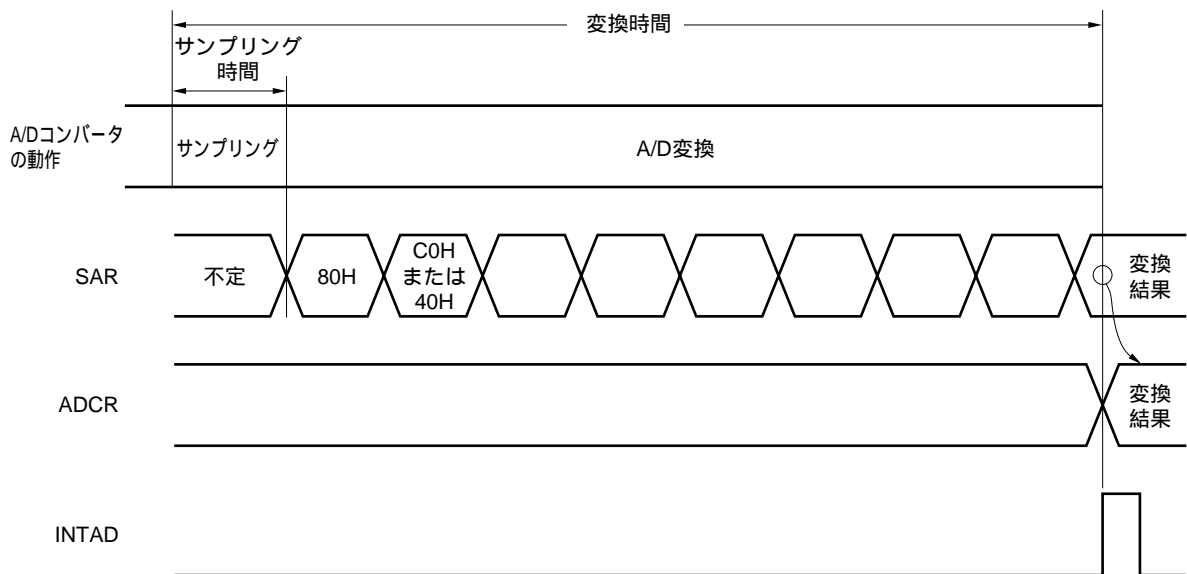
- ・アナログ入力電圧 電圧タップ : ビット6 = 1
- ・アナログ入力電圧 電圧タップ : ビット6 = 0

このような比較をSARのビット0まで続けます。

8ビットの比較が終了したとき、SARには有効なデジタルの結果が残り、その値がA/D変換結果レジスタ (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) を発生させることができます。

図10-4 A/Dコンバータの基本動作



A/D変換動作は、ソフトウェアによりA/Dコンバータ・モード・レジスタ (ADM) のビット7 (ADCS) をリセット (0) するまで連続的に行われます。

A/D変換動作中に、ADM、A/Dコンバータ入力選択レジスタ (ADIS) に対する書き込み操作を行うと変換動作は初期化され、ADCSがセット (1) されていれば、最初から変換を開始します。

A/D変換結果レジスタ (ADCR) は、 $\overline{\text{RESET}}$ により不定となります。

### 10.4.2 入力電圧と変換結果

アナログ入力端子 ( ANI0-ANI11 ) に入力されたアナログ入力電圧とA/D変換結果 ( A/D変換結果レジスタ ( ADCR ) ) には次式に示す関係があります。

$$ADCR = \text{INT} \left( \frac{V_{IN}}{AV_{DD}} \times 256 + 0.5 \right)$$

または,

$$( ADCR - 0.5 ) \times \frac{AV_{DD}}{256} < V_{IN} < ( ADCR + 0.5 ) \times \frac{AV_{DD}}{256}$$

INT ( ) : ( ) 内の値の整数部を返す関数

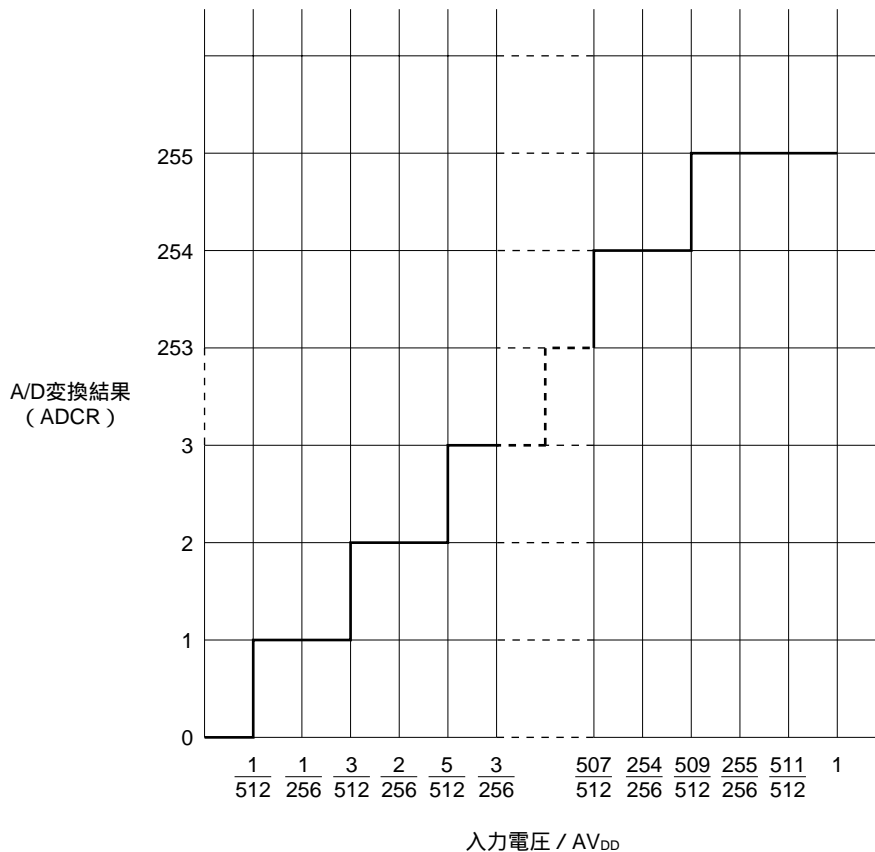
$V_{IN}$  : アナログ入力電圧

$AV_{DD}$  : A/Dコンバータの電源電圧入力

ADCR : A/D変換結果レジスタ ( ADCR ) の値

図10 - 5 にアナログ入力電圧とA/D変換結果の関係を示します。

図10 - 5 アナログ入力電圧とA/D変換結果の関係



### 10.4.3 A/Dコンバータの動作モード

A/Dコンバータ入力選択レジスタ (ADIS) によってANI0-ANI11からアナログ入力を1チャンネル選択し、A/D変換を開始させてください。

A/D変換動作の起動方法は、ソフトウェア・スタート (A/Dコンバータ・モード・レジスタ (ADM) を設定することにより開始) のみです。

また、A/D変換結果は、A/D変換結果レジスタ (ADCR) に格納され、同時に割り込み要求信号 (INTAD) が発生されます。

#### ・ソフトウェア・スタートによるA/D変換動作

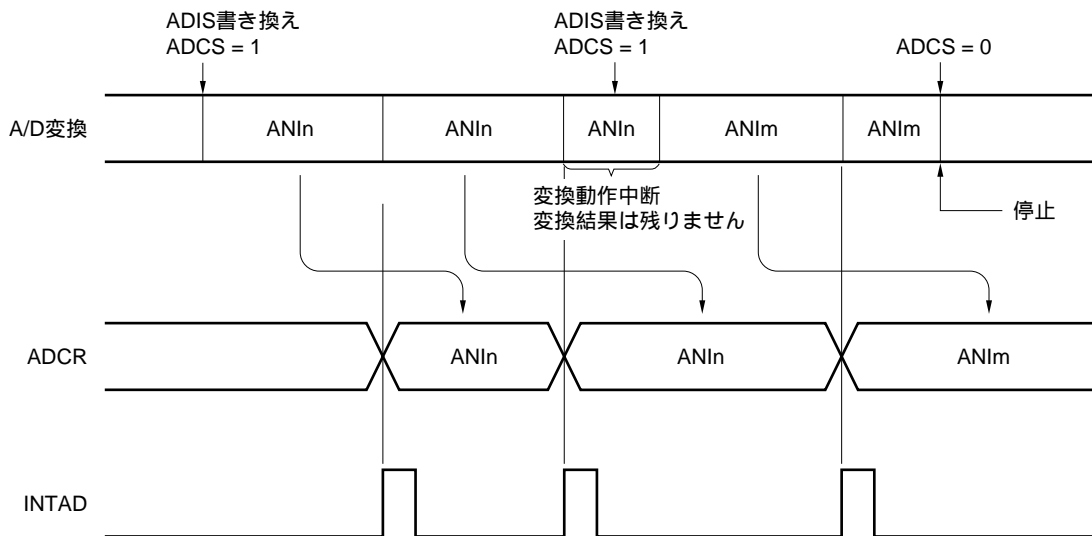
A/Dコンバータ・モード・レジスタ (ADM) のビット7 (ADCS) に1を設定することにより、A/Dコンバータ入力選択レジスタ (ADIS) で指定したアナログ入力端子に印加されている電圧のA/D変換動作を開始します。

A/D変換動作が終了すると、変換結果をA/D変換結果レジスタ (ADCR) に格納し、割り込み要求信号 (INTAD) が発生します。A/D変換動作が一度起動し、1回のA/D変換が終了すると、ただちに次のA/D変換動作を開始します。新たなデータをADISに書き込むまで繰り返しA/D変換動作を行います。

A/D変換動作中に、ADISを書き換えると、そのとき行っていたA/D変換動作を中断し、新たに選択したアナログ入力チャンネルのA/D変換動作を開始します。

また、A/D変換動作中にADCSが0であるデータをADMに書き込むと、ただちにA/D変換動作を停止します。

図10 - 6 ソフトウェア・スタートによるA/D変換動作



備考 n = 0, 1, ……………, 11  
 m = 0, 1, ……………, 11

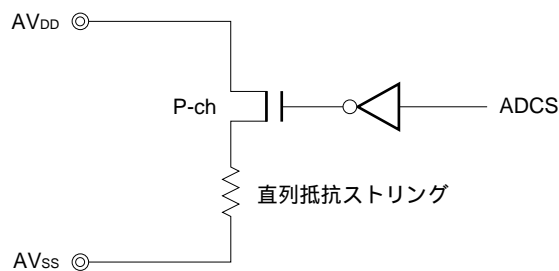
## 10.5 A/Dコンバータの注意事項

### (1) スタンバイ・モード時の消費電流について

A/Dコンバータは、スタンバイ・モード時には動作が停止します。このとき変換動作停止 (A/Dコンバータ・モード・レジスタ (ADM) のビット7 (ADCS) = 0) にすることにより、消費電流を低減させることができます。

スタンバイ・モード時の消費電流を低減させる方法例を図10 - 7 に示します。

図10 - 7 スタンバイ・モード時の消費電流を低減させる方法例



### (2) ANI0-ANI11入力範囲について

ANI0-ANI11入力電圧は規格の範囲内でご使用ください。特にAVDD以上、AVSS以下 (絶対最大定格の範囲内でも) の電圧が入力されると、そのチャンネルの変換値が不定となります。また、ほかのチャンネルの変換値にも影響を与えることがあります。

### (3) 競合動作について

変換終了時のA/D変換結果レジスタ (ADCR) ライトと命令によるADCRリードとの競合  
ADCRリードが優先されます。リードしたあと、新しい変換結果がADCRにライトされます。

変換終了時のADCRライトと外部トリガ信号入力の競合  
A/D変換中の外部トリガ信号は受け付けません。したがってADCRライト中の外部トリガ信号も受け付けません。

変換終了時のADCRライトとA/Dコンバータ・モード・レジスタ (ADM) ライト、またはA/Dコンバータ入力選択レジスタ (ADIS) ライトの競合

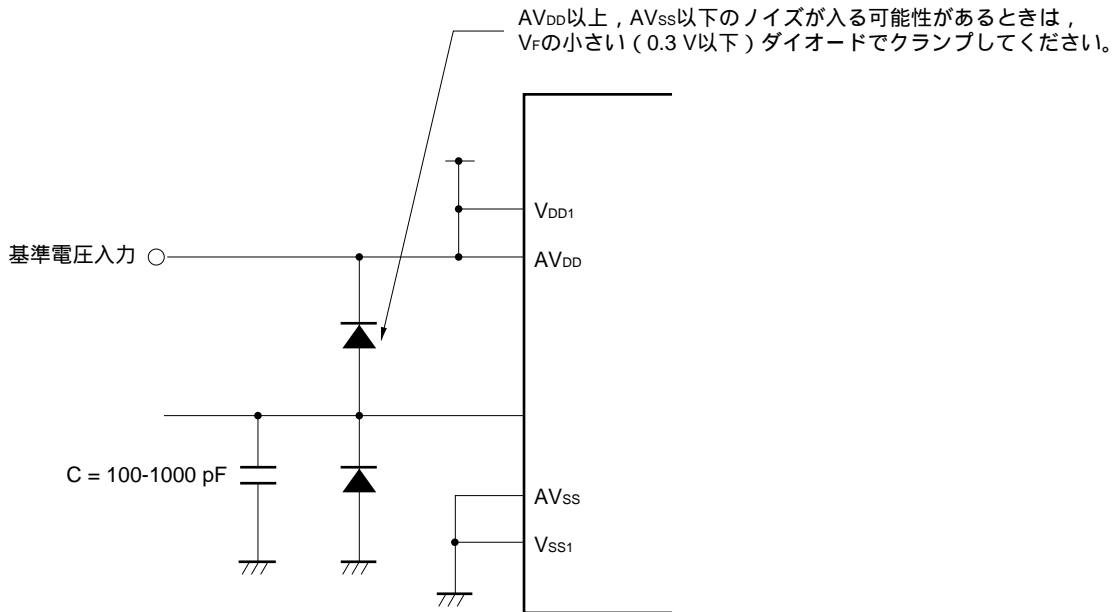
ADMまたはADISへのライトが優先されます。ADCRへのライトはされません。また、変換終了割り込み要求信号 (INTAD) も発生しません。



**(4) ノイズ対策について**

8ビット分解能を保つためには、 $AV_{DD}$ 、 $ANI0$ - $ANI11$ 端子へのノイズに注意する必要があります。アナログ入力源の出力インピーダンスが高いほど影響が大きくなりますので、ノイズを低減するために図10-8のようにCを外付けすることを推奨します。

図10-8 アナログ入力端子の処理

**(5) ANI0-ANI11**

アナログ入力 ( $ANI0$ - $ANI11$ ) 端子はポート端子 ( $P00$ - $P03$ ,  $P11$ - $P17$ ) と兼用になっています。

$ANI0$ - $ANI11$ のいずれかを選択してA/D変換をする場合、変換中にポートの入力命令は実行しないでください。変換分解能が低下することがあります。

また、A/D変換中の端子に隣接する端子へデジタル・パルス印加すると、カップリング・ノイズによってA/D変換値が期待どおりに得られないこともあります。したがって、A/D変換中の端子に隣接する端子へのパルス印加はしないようにしてください。

**(6) AV<sub>DD</sub>端子の入力インピーダンスについて**

$AV_{DD}$ 端子は基準電圧源の端子を兼用しており、 $AV_{DD}$ 端子と $AV_{SS}$ 端子の間には約21.4kΩの直列抵抗ストリングが接続されています。

したがって、基準電圧源の出力インピーダンスが高い場合、 $AV_{DD}$ 端子と $AV_{SS}$ 端子の間の直列抵抗ストリングと並列接続することになり、基準電圧の誤差が大きくなります。

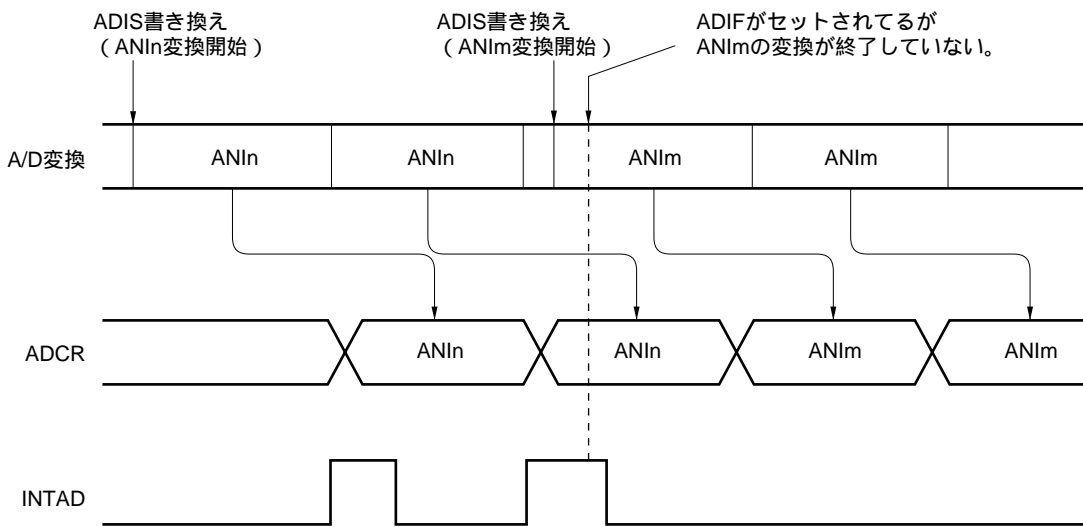
**(7) 割り込み要求フラグ (ADIF) について**

A/Dコンバータ入力選択レジスタ (ADIS) を変更しても割り込み要求フラグ (ADIF) はクリアされません。

したがって、A/D変換中にアナログ入力端子の変更を行った場合、ADIS書き換え直前に変更前のアナログ入力に対するA/D変換結果および変換終了割り込み要求フラグがセットされる場合があります、ADIS書き換え直後にADIFを読み出すと、変更後のアナログ入力に対するA/D変換が終了していないにもかかわらずADIFがセットされている場合がありますので注意してください。

また、A/D変換を一度停止させて再開する場合は、再開する前にADIFをクリアしてください。

図10 - 9 A/D変換終了割り込み要求発生タイミング



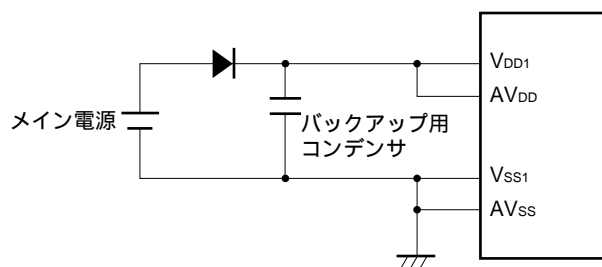
備考 n = 0, 1, ……………, 11  
 m = 0, 1, ……………, 11

(8) AV<sub>DD</sub>端子について

AV<sub>DD</sub>端子はアナログ回路の電源端子であり、ANI0-ANI11の入力回路にも電源を供給しています。

したがって、バックアップ電源に切り替えるようなアプリケーションにおいても、図10 - 10のように必ずV<sub>DD1</sub>端子と同レベルの電位を印加してください。

図10 - 10 AV<sub>DD</sub>端子の処理



**(9) A/D変換スタート直後の変換結果について**

A/D変換動作をスタートした直後の最初のA/D変換値は不定になります。A/D変換終了割り込み要求 (INTAD) をポーリングし、最初の変換結果を廃棄するなどの対策を行ってください。

〔メモ〕

# 第11章 シリアル・インタフェース

## 11.1 シリアル・インタフェースの機能

シリアル・インタフェースには、次の2種類のモードがあります。

- ・動作停止モード
- ・3線式シリアルI/Oモード

### (1) 動作停止モード

シリアル転送を行わないときに使用するモードです。

### (2) 3線式シリアルI/Oモード (MSB先頭固定)

シリアル・クロック ( $\overline{\text{SCK}}$ )、シリアル出力 (SO)、シリアル入力 (SI) の3本のラインにより、8ビット・データ転送を行うモードです。

3線式シリアルI/Oモードは、同時送受信動作が可能なので、データ転送の処理時間が短くなります。

シリアル転送する8ビット・データの先頭ビットは、MSB固定です。

3線式シリアルI/Oモードは、クロック同期式シリアル・インタフェースを内蔵する周辺I/Oや表示コントローラなどを接続するときに有効です。

## 11.2 シリアル・インタフェースの構成

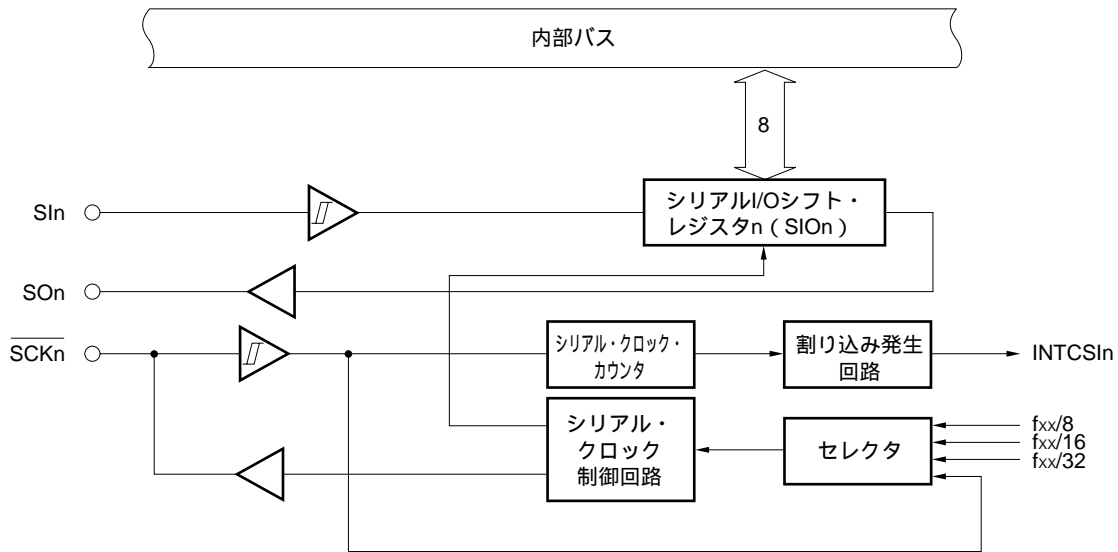
シリアル・インタフェースは、次のハードウェアで構成されています。

表11 - 1 シリアル・インタフェースの構成

項目	構成
レジスタ	シリアルI/Oシフト・レジスタn (SIOn)
制御レジスタ	シリアル動作モード・レジスタn (CSIMn)

備考 n = 0, 1

図11 - 1 シリアル・インタフェースのブロック図



備考 n = 0, 1

(1) シリアル/Oシフト・レジスタ n (SIO\_n)

パラレル-シリアルの変換を行い、シリアル・クロックに同期してシリアル送受信（シフト動作）を行う8ビット・レジスタです。

SIO\_nは、8ビット・メモリ操作命令で設定します。

シリアル動作モード・レジスタ n (CSIM\_n) のビット7 (CSIE\_n) が1のとき、SIO\_nにデータを書き込むか、または読み出すことによりシリアル動作が開始されます。

送信時は、SIO\_nに書き込まれたデータが、シリアル出力 (SO\_n) に出力されます。

受信時は、データがシリアル入力 (SIn) からSIO\_nに読み込まれます。

RESET入力により、不定になります。

**注意** 転送動作中のSIO\_nアクセスは、転送起動トリガとなるアクセス以外は実行しないでください (CSIM\_nのビット2 (MODE\_n) = 0のときはリードが、MODE\_n = 1のときはライトが禁止となります)。

備考 n = 0, 1

(2) シリアル・クロック・カウンタ

送受信動作時に出力されるシリアル・クロック、および入力されるシリアル・クロックをカウントし、8ビット・データの送受信が行われたことを調べます。

### 11.3 シリアル・インタフェースを制御するレジスタ

シリアル・インタフェースを制御するレジスタには、シリアル動作モード・レジスタ  $n$  (CSIMn) があります。

・シリアル動作モード・レジスタ  $n$  (CSIMn)

シリアル・インタフェースのシリアル・クロック、動作モード、動作の許可/停止を設定するレジスタです。

CSIMnは、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図11-2 シリアル動作モード・レジスタ  $n$  のフォーマット

略号	⑦	6	5	4	3	2	1	0	アドレス	リセット時	R/W
CSIMn	CSIE <sub>n</sub>	0	0	0	0	MODE <sub>n</sub>	SCL <sub>n1</sub>	SCL <sub>n0</sub>	FF90H, FF91H	00H	R/W

CSIE <sub>n</sub>	SIO <sub>n</sub> の動作許可/禁止の指定		
	シフト・レジスタ動作	シリアル・カウンタ	ポート
0	動作禁止	クリア	ポート機能 <sup>注</sup>
1	動作許可	カウント動作許可	シリアル機能 + ポート機能

MODE <sub>n</sub>	転送動作モード・フラグ		
	動作モード	転送起動トリガ	SO <sub>n</sub> 出力
0	送信/送受信モード	SIO <sub>n</sub> ライト	通常出力
1	受信専用モード	SIO <sub>n</sub> リード	ロウ・レベル固定

SCL <sub>n1</sub>	SCL <sub>n0</sub>	クロックの選択
0	0	$\overline{\text{SCKn}}$ 端子への外部クロック入力
0	1	$f_{xx}/8$ (1.56 MHz)
1	0	$f_{xx}/16$ (781 kHz)
1	1	$f_{xx}/32$ (391 kHz)

注 CSIE<sub>n</sub> = 0 (SIO<sub>n</sub>動作停止状態) のときは、SIO<sub>n</sub>, SO<sub>n</sub>,  $\overline{\text{SCKn}}$ と接続された端子は、ポート機能として使用できます。

備考1 .  $f_{xx}$  : メイン・システム・クロック周波数

2 . ( ) 内は、 $f_{xx} = 12.5$  MHz動作時。

3 .  $n = 0, 1$

## 11.4 シリアル・インタフェースの動作

シリアル・インタフェースには、次の2種類の動作モードがあります。

- ・動作停止モード
- ・3線式シリアルI/Oモード

### 11.4.1 動作停止モード

動作停止モードではシリアル転送を行いませんので、消費電力を低減することができます。

また、シリアルI/Oシフト・レジスタ  $n$  (SIO $n$ ) もシフト動作を行いませんので、通常の8ビット・レジスタとして使用することができます。

また、動作停止モードでは、SIn, SO $n$ ,  $\overline{\text{SCK}}n$ 端子を通常の入出力ポートとして使用できます。

#### (1) レジスタの設定

動作停止モードの設定は、シリアル動作モード・レジスタ  $n$  (CSIM $n$ ) で行います。

CSIM $n$ は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

略号	⑦	6	5	4	3	2	1	0	アドレス	リセット時	R/W
CSIM $n$	CSIE $n$	0	0	0	0	MODE $n$	SCL $n$ 1	SCL $n$ 0	FF90H, FF91H	00H	R/W

CSIE $n$	SIO $n$ の動作許可/禁止の指定		
	シフト・レジスタ動作	シリアル・カウンタ	ポート
0	動作禁止	クリア	ポート機能 <sup>注</sup>
1	動作許可	カウンタ動作許可	シリアル機能 + ポート機能

注 CSIE $n$  = 0 (SIO $n$ 動作停止状態) のときは、SIn, SO $n$ ,  $\overline{\text{SCK}}n$ と接続された端子は、ポート機能として使用できます。

備考  $n = 0, 1$



### 11.4.2 3線式シリアルI/Oモード

3線式シリアルI/Oモードは、クロック同期式シリアル・インタフェースを内蔵する周辺I/Oや表示コントローラなどを接続するときに有効です。

シリアル・クロック ( $\overline{\text{SCKn}}$ )、シリアル出力 (SO<sub>n</sub>)、シリアル入力 (SI<sub>n</sub>) の3本のラインで通信を行います。

#### (1) レジスタの設定

3線式シリアルI/Oモードの設定は、シリアル動作モード・レジスタ n (CSIM<sub>n</sub>) で行います。

CSIM<sub>n</sub>は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、00Hになります。

略号	⑦	6	5	4	3	2	1	0	アドレス	リセット時	R/W
CSIM <sub>n</sub>	CSIE <sub>n</sub>	0	0	0	0	MODE <sub>n</sub>	SCL <sub>n1</sub>	SCL <sub>n0</sub>	FF90H, FF91H	00H	R/W

CSIE <sub>n</sub>	SI <sub>n</sub> の動作許可 / 禁止の指定		
	シフト・レジスタ動作	シリアル・カウンタ	ポート
0	動作禁止	クリア	ポート機能 <sup>注</sup>
1	動作許可	カウント動作許可	シリアル機能 + ポート機能

MODE <sub>n</sub>	転送動作モード・フラグ		
	動作モード	転送起動トリガ	SO <sub>n</sub> 出力
0	送信 / 送受信モード	SI <sub>n</sub> ライト	通常出力
1	受信専用モード	SI <sub>n</sub> リード	ロウ・レベル固定

SCL <sub>n1</sub>	SCL <sub>n0</sub>	クロックの選択
0	0	$\overline{\text{SCKn}}$ 端子への外部クロック入力
0	1	f <sub>xx</sub> /8 (1.56 MHz)
1	0	f <sub>xx</sub> /16 (781 kHz)
1	1	f <sub>xx</sub> /32 (391 kHz)

注 CSIE<sub>n</sub> = 0 (SI<sub>n</sub>動作停止状態) のときは、SI<sub>n</sub>、SO<sub>n</sub>、 $\overline{\text{SCKn}}$ と接続された端子は、ポート機能として使用できます。

備考1 . f<sub>xx</sub> : メイン・システム・クロック周波数

2 . ( ) 内は、f<sub>xx</sub> = 12.5 MHz動作時。

3 . n = 0, 1

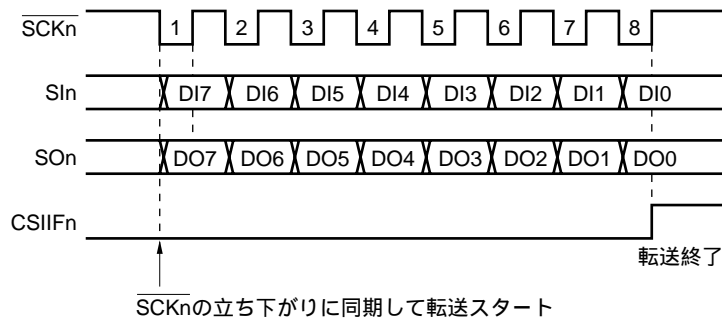
(2) 通信動作

3線式シリアルI/Oモードは、8ビット単位でデータの送受信を行います。データは、シリアル・クロックに同期して1ビットごとに送受信されます。

シリアルI/Oシフト・レジスタn (SIO<sub>n</sub>)のシフト動作は、シリアル・クロック ( $\overline{\text{SCKn}}$ )の立ち下がりに同期して行われます。そして、送信データがSO<sub>n</sub>ラッチに保持され、SO<sub>n</sub>端子から出力されます。また、 $\overline{\text{SCKn}}$ の立ち上がりで、SI<sub>n</sub>端子に入力された受信データがSIO<sub>n</sub>にラッチされます。

8ビット転送終了により、SIO<sub>n</sub>の動作は自動的に停止し、割り込み要求フラグ (CSIF<sub>n</sub>) がセットされます。

図11 - 3 3線式シリアルI/Oモードのタイミング



(3) 転送スタート

シリアル転送は、次の2つの条件を満たしたとき、シリアルI/Oシフト・レジスタn (SIO<sub>n</sub>)に転送データをセットする(またはリードする)ことで開始します。

- ・ SIO<sub>n</sub>の動作制御ビット (CSIE<sub>n</sub>) = 1
- ・ 8ビット・シリアル転送後、内部のシリアル・クロックが停止した状態か、または $\overline{\text{SCKn}}$ がハイ・レベルの状態
- ・ 送信/送受信モード
  - CSIE<sub>n</sub> = 1, MODE<sub>n</sub> = 0のとき、SIO<sub>n</sub>ライトで転送スタート
- ・ 受信専用モード
  - CSIE<sub>n</sub> = 1, MODE<sub>n</sub> = 1のとき、SIO<sub>n</sub>のリードで転送スタート

**注意** SIO<sub>n</sub>にデータを書き込んだあと、CSIE<sub>n</sub>を“1”にしても転送はスタートしません。

8ビット転送終了により、シリアル転送は自動的に停止し、割り込み要求フラグ (CSIF<sub>n</sub>) をセットします。

# 第12章 FIPコントローラ/ドライバ

## 12.1 FIPコントローラ/ドライバの機能

μPD784976サブシリーズに内蔵しているFIPコントローラ/ドライバの機能を示します。

- (1) 表示データの自動読み出しによる表示信号出力 (DMA動作) が可能。
- (2) FIP表示に使用しない端子は、入出力ポートおよび出力ポートとして使用可能 (FIP16-FIP47端子のみ)。
- (3) 表示モード・レジスタ1 (DSPM1) により8段階の輝度調節が可能。
- (4) キー・スキャン応用を考慮したハードウェアを内蔵。
  - ・キー・スキャン・タイミングを示す割り込み要求信号 (INTKS) を発生。
  - ・キー・スキャン・フラグ (KSF) により、キー・スキャン・データを出力しているタイミングの検出が可能。
  - ・キースキャン・タイミングの挿入/非挿入を選択可能。
- (5) FIPを直接駆動可能な高耐圧出力バッファを内蔵。
- (6) FIP0-FIP15端子は、プルダウン抵抗を内蔵。FIP16-FIP47端子は、マスクROM製品のみマスク・オプションによりプルダウン抵抗の内蔵が可能 (μPD78F4976は、プルダウン抵抗を内蔵していません)。

μPD784976サブシリーズが内蔵する48本のFIP出力端子のうち、FIP16-FIP47はポート機能との兼用端子になっています。なお、FIP0-FIP15はFIP出力専用端子です。

表示モード・レジスタ0 (DSPM0) のビット7 (DSPEN) によりFIP表示禁止に設定したとき、FIP16-FIP47はポート端子として使用できます。また、FIP表示許可状態であっても、表示信号出力として使用しないFIP出力端子についてはポート端子として使用できます。

表12 - 1 FIP出力端子とポートの兼用端子対応表

FIP端子名称	兼用ポート名称	入出力
FIP16-FIP23	P70-P77	入出力ポート
FIP24-FIP31	P80-P87	入出力ポート
FIP32-FIP39	P90-P97	入出力ポート
FIP40-FIP47	P100-P107	出力専用ポート

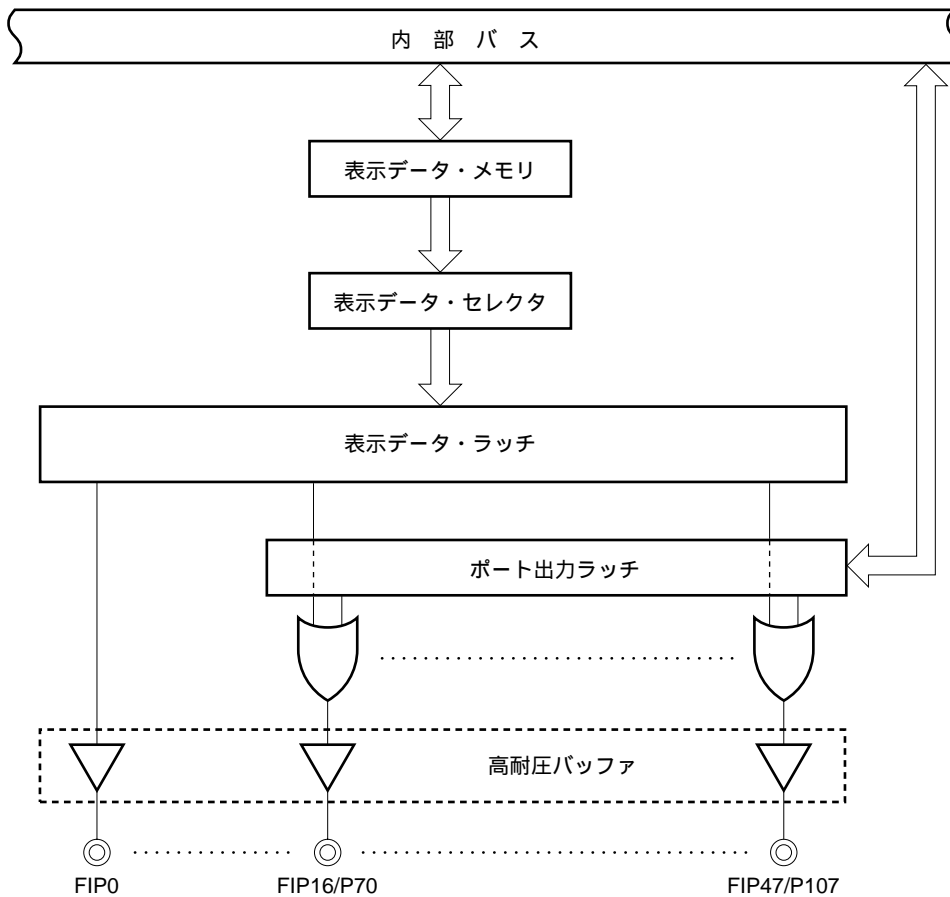
## 12.2 FIPコントローラ/ドライバの構成

FIPコントローラ/ドライバは、次のハードウェアで構成されています。

表12 - 2 FIPコントローラ/ドライバの構成

項目	構成
表示出力	48本
制御レジスタ	表示モード・レジスタ 0 (DSPM0) 表示モード・レジスタ 1 (DSPM1) 表示モード・レジスタ 2 (DSPM2)

図12 - 1 FIPコントローラ/ドライバのブロック図



## 12.3 FIPコントローラ/ドライバを制御するレジスタ

### 12.3.1 制御レジスタ

FIPコントローラ/ドライバを制御するレジスタには、次の3種類があります。

- ・表示モード・レジスタ0 (DSPM0)
- ・表示モード・レジスタ1 (DSPM1)
- ・表示モード・レジスタ2 (DSPM2)

#### (1) 表示モード・レジスタ0 (DSPM0)

DSPM0は次の設定をするレジスタです。

- ・表示許可/禁止
- ・FIP出力本数

DSPM0は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 入力により、10Hになります。

図12 - 2 表示モード・レジスタ0のフォーマット

略号	⑦	6	5	4	3	2	1	0	アドレス	リセット時	R/W
DSPM0	DSPEN	0	FOUT5	FOUT4	FOUT3	FOUT2	FOUT1	FOUT0	FFB0H	10H	R/W

DSPEN	FIP表示の許可 / 禁止の指定
0	禁止
1	許可

FOUT5	FOUT4	FOUT3	FOUT2	FOUT1	FOUT0	FIP出力本数
0	1	0	1	1	1	17-24
0	1	1	1	1	1	25-32
1	0	0	1	1	1	33-40
1	0	1	1	1	1	41-48
上記以外						設定禁止

- 注意 1 . ビット 6 には、必ず “ 0 ” を設定してください。
- 2 . ビット 7 (DSPEN) = 1 のときは、DSPEN 以外のビットにデータを書き込まないでください。
- 3 . FIP 出力に使用する端子の兼用ポートの出力ラッチには、必ず “ 0 ” を設定してください。

(2) 表示モード・レジスタ1 (DSPM1)

DSPM1は次の設定をするレジスタです。

- ・ FIP出力信号のブランキング幅
- ・ 表示パターン数

DSPM1は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。  
 $\overline{\text{RESET}}$ 入力により、01Hになります。

図12 - 3 表示モード・レジスタ1のフォーマット

略号	7	6	5	4	3	2	1	0	アドレス	リセット時	R/W
DSPM1	FBLK2	FBLK1	FBLK0	FPAT4	FPAT3	FPAT2	FPAT1	FPAT0	FFB2H	01H	R/W

FBLK2	FBLK1	FBLK0	FIP出力信号のブランキング幅
0	0	0	1/16
0	0	1	2/16
0	1	0	4/16
0	1	1	6/16
1	0	0	8/16
1	0	1	10/16
1	1	0	12/16
1	1	1	14/16

FPAT4	FPAT3	FPAT2	FPAT1	FPAT0	表示パターン数
0	0	0	0	1	2
0	0	0	1	0	3
0	0	0	1	1	4
0	0	1	0	0	5
0	0	1	0	1	6
0	0	1	1	0	7
0	0	1	1	1	8
0	1	0	0	0	9
0	1	0	0	1	10
0	1	0	1	0	11
0	1	0	1	1	12
0	1	1	0	0	13
0	1	1	0	1	14
0	1	1	1	0	15
0	1	1	1	1	16
上記以外					設定禁止

**注意** 表示モード・レジスタ0 (DSPM0) のビット7 (DSPEN) = 1のときは、表示モード・レジスタ1 (DSPM1) にデータを書き込まないでください。



(3) 表示モード・レジスタ2 (DSPM2)

DSPM2は次の設定をするレジスタです。また、表示タイミング/キー・スキャンの状態を表示します。

- ・キー・スキャン・タイミングの挿入/非挿入
- ・表示サイクル ( $T_{DSP}$ )

DSPM2は、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。ただし、ビット7 (KSF) のみ、1ビット・メモリ操作命令で読み出すことができます。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図12-4 表示モード・レジスタ2のフォーマット

略号	⑦	6	5	4	3	2	1	0	アドレス	リセット時	R/W
DSPM2	KSF	KSM	0	0	0	0	FCYC1	FCYC0	FFB4H	00H	R/W

KSF	キー・スキャン・サイクルの状態
0	キー・スキャン・サイクル以外
1	キー・スキャン・サイクル中

KSM	キー・スキャン・サイクル挿入の選択
0	挿入しない
1	挿入する

FCYC1	FCYC0	表示サイクル
0	0	$16 \times 2^9 / f_{xx}$ (655.36 $\mu$ s)
0	1	$16 \times 2^8 / f_{xx}$ (327.68 $\mu$ s)
1	0	$16 \times 2^7 / f_{xx}$ (163.84 $\mu$ s)
1	1	$16 \times 2^6 / f_{xx}$ (81.92 $\mu$ s)

注意1. ビット2-ビット5には、必ず“0”を設定してください。

2. 表示モード・レジスタ0 (DSPM0)のビット7 (DSPEN) = 1のときは、表示モード・レジスタ2 (DSPM2) にデータを書き込まないでください。

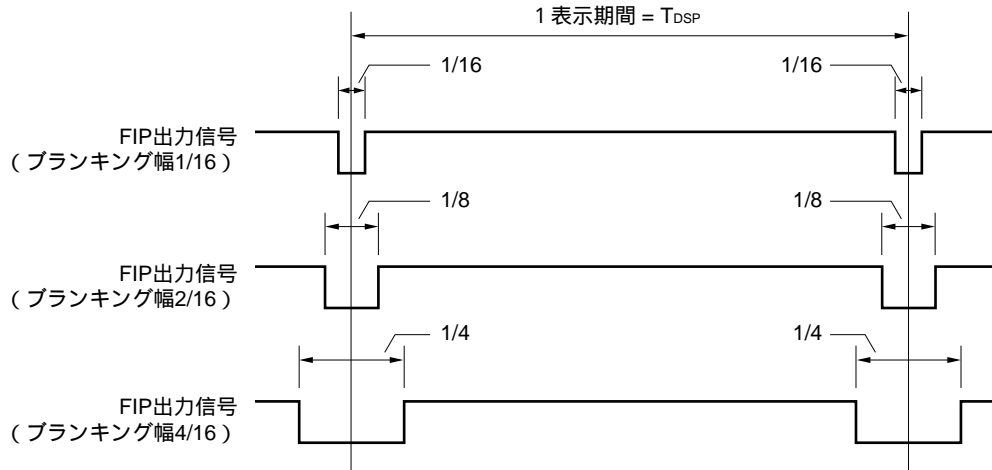
備考1.  $f_{xx}$ : メイン・システム・クロック周波数

2. ( ) 内は、 $f_{xx} = 12.5$  MHz動作時。

### 12.3.2 1表示期間とブランキング幅

表示モード・レジスタ1 (DSPM1) のビット5-ビット7 (FBLK0-FBLK2) で設定されるブランキング幅によって、表示期間の始まりと終わりに2等分されるかたちでブランキングされます。

図12 - 5 FIP出力信号のブランキング幅



## 12.4 表示データ・メモリ

表示データ・メモリは、表示する表示データを格納する96バイトのRAM領域です。

表示データ・メモリは、EA00H-EA5FH番地にマッピングしています。

FIPコントローラはFIP表示のために、表示データ・メモリに格納したデータをCPUの動作とは無関係に読み出します（DMA動作）。

また、表示に使用しない領域は、通常のRAMとして使用できます。

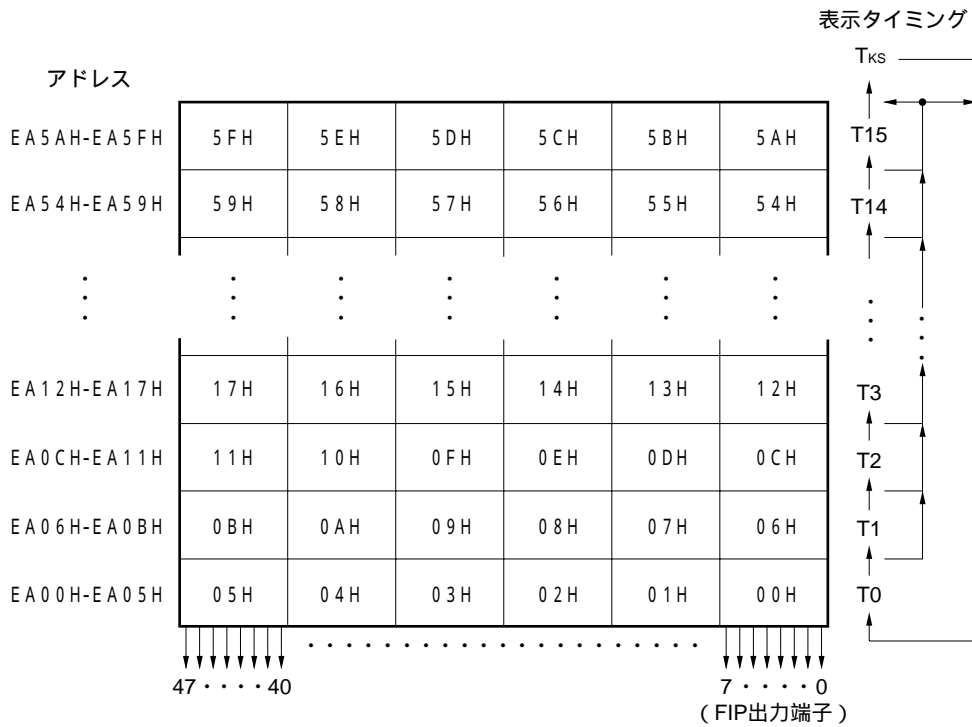
キー・スキャン・タイミング（ $T_{ks}$ ）には、FIP出力はオール“0”出力となり、ポート7-ポート10の出力ラッチのデータがFIP16/P70-FIP47/P107に出力されます。

表示データ・メモリのアドレス配置は次のようになっています。

### ・FIP出力48本，16パターンの場合

各表示タイミング（ $T_0$ - $T_{15}$ ）で出力されるデータに対応する表示データ・メモリのアドレスは図12-6（たとえば、 $T_0 = EA00H-EA05H$ ,  $T_1 = EA06H-EA0BH$ ）のようになっており、FIP出力が48本（FIP0-FIP47）の場合は6バイトで1表示データを構成します。この1表示データの最下位ビットから最上位ビットの順にFIP出力端子0（FIP0）からFIP出力端子47（FIP47）に対応しています。

図12-6 表示データ・メモリのアドレス配置とFIP出力の関係  
（FIP出力48本，16パターンの場合）



## 12.5 キー・スキャン・フラグとキー・スキャン・データ

### 12.5.1 キー・スキャン・フラグ

キー・スキャン・フラグ (KSF) はキー・スキャン・タイミング中に1にセットされ、表示タイミングでは自動的に0にリセットされるフラグです。

KSFは表示モード・レジスタ2 (DSPM2) のビット7にマッピングされており、1ビット単位でのテストが可能です。書き込みはできません。

KSFをテストすることにより、キー・スキャン・タイミング中であるかどうかを判断でき、キー入力データが正しいかどうかを決めることができます。

また、キー・スキャン・タイミング挿入指定フラグ (KSM) (表示モード・レジスタ2 (DSPM2) のビット6) でキー・スキャン・タイミングの挿入/非挿入の選択ができます。

### 12.5.2 キー・スキャン・データ

ポート7-ポート10に格納したデータがキー・スキャン・タイミング中にFIP16-FIP47端子から出力されます。

**注意** キー・スキャン・タイミング中にセグメントと桁の両方がオンになるようなスキャンを行った場合、表示がちらつくことがあります。

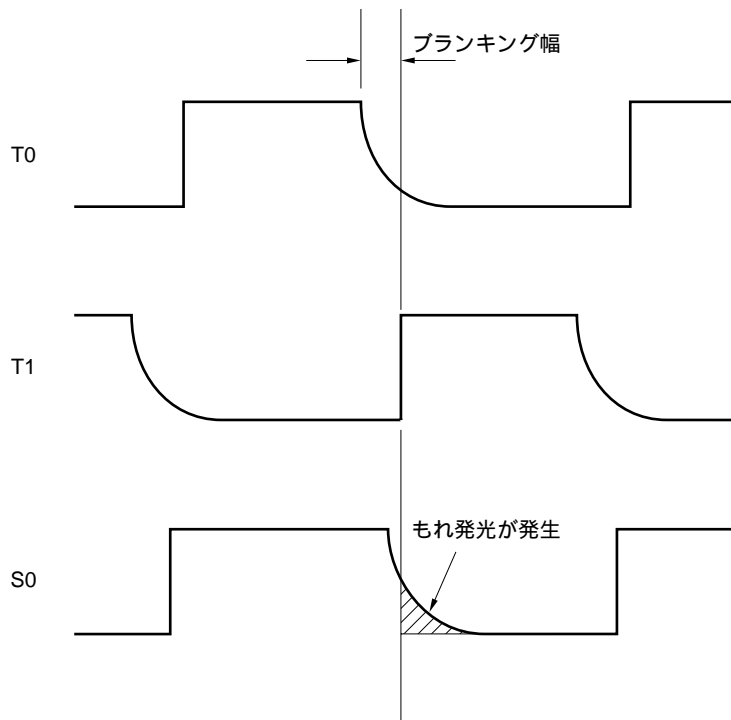
## 12.6 蛍光表示管のもれ発光について

$\mu$ PD784976サブシリーズを用いて蛍光表示管を駆動するときに、もれ発光が生じる場合があります。この原因としては主に次の2つが考えられます。

### (1) ブランキング時間が短いことによるもれ発光

図12-7は2桁表示で1桁目T0は点灯し、2桁目T1は非点灯とした場合の信号波形を示しています。この図のようにブランキング時間が短いとセグメント信号がなくなる前にT1信号が立ち上がってしまうため、もれ発光が生じます。一般的にはブランキング時間は20  $\mu$ s程度必要となりますので、表示モード・レジスタ1 (DSPM1) の設定値を考慮する必要があります。

図12-7 ブランキング時間が短いことによるもれ発光の概念図



### (2) 蛍光表示管のセグメント-グリッド間容量によるもれ発光

図12-9のように十分なブランキング時間をとったときでも、もれ発光が生じる場合があります。これは図12-8で $C_{SG}$ として示されるように蛍光表示管のグリッド-セグメント間に容量があるため、セグメント信号がオンしたとき、 $C_{SG}$ を通してタイミング信号端子が持ち上げられてしまいます。この電圧が図12-9のようにカット・オフ電圧 ( $E_K$ ) 以上となると、もれ発光が生じます。

このヒゲ状の電圧は、 $C_{SG}$ の大きさと内蔵プルダウン抵抗 ( $R_L$ ) の値によって変化します。 $C_{SG}$ の値が大きいほど、また $R_L$ の値が大きいほどこの電圧は大きくなり、もれ発光は発生しやすくなります。

この $C_{SG}$ の値は、蛍光表示管の表示面積によって異なり、面積が大きいほど $C_{SG}$ は大きくなります。

したがって、蛍光表示管の大きさによってもれ発光を発生させないためのプルダウン抵抗値が異なってきます。

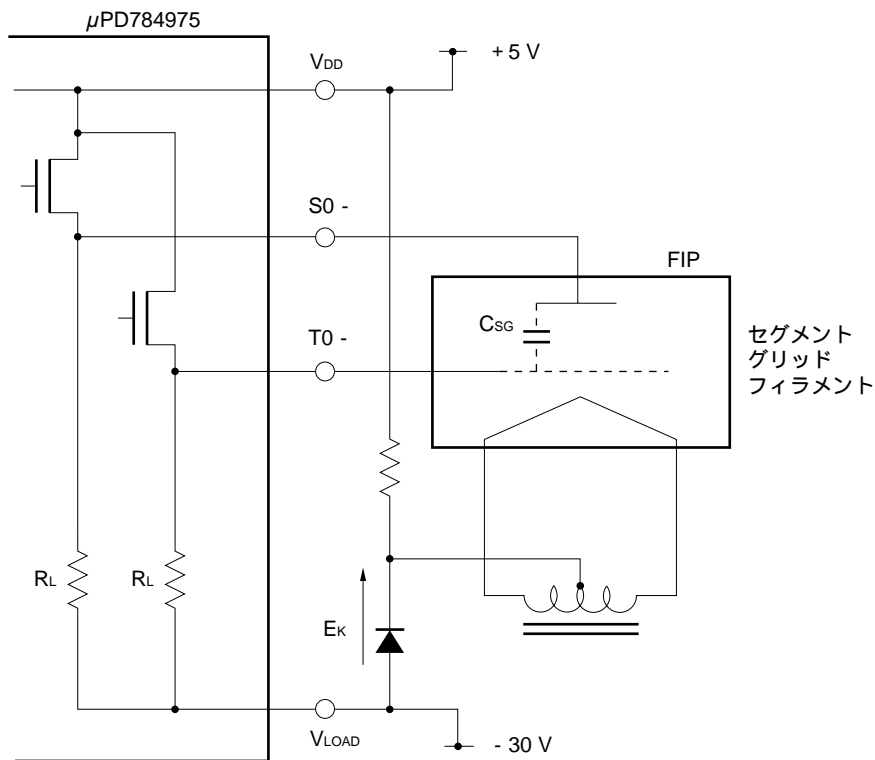
一方、マスク・オプションで内蔵できるプルダウン抵抗値は比較的大きくなっていますので、この内蔵プルダウン抵抗だけではもれ発光を抑えることができない場合が生じます。

十分な表示品質が得られない場合には、バック・バイアスを深くする（ $E_k$ を上げる）、蛍光表示管にフィルタをかける、あるいはタイミング信号端子に数10k のプルダウン抵抗を外付けするなどの対策を検討してください。

この $C_{SG}$ に起因するもれ発光は、全体の表示周期に対するヒゲ状電圧のデューティ・サイクルによって発生しやすさが変化しますので、表示桁数が少ないほど発生しやすいことになります。

また、表示輝度を下げても効果があります。

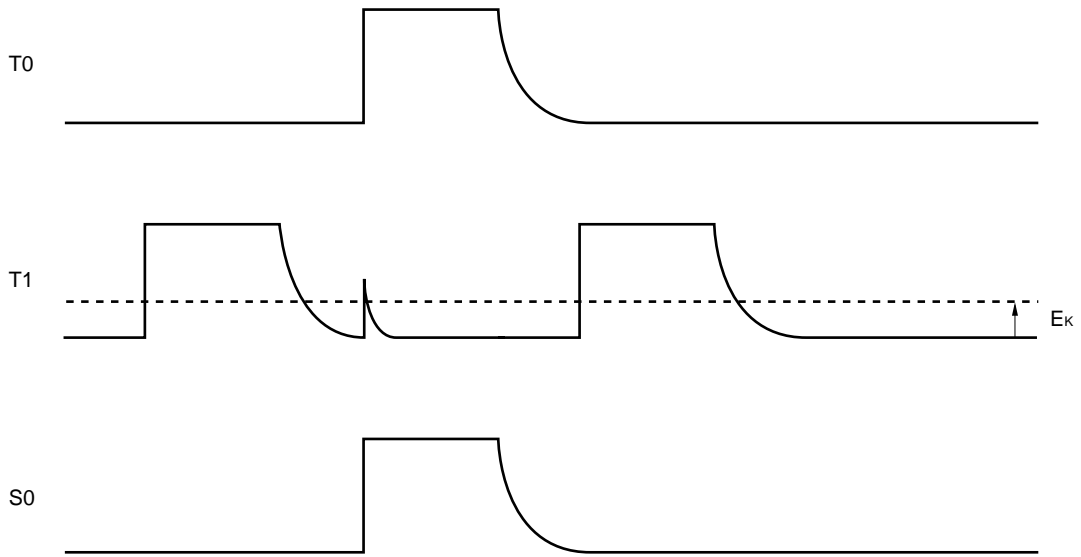
図12 - 8  $C_{SG}$ によるもれ発光の説明図



$E_k$  : カット・オフ電圧

$R_L$  : 内蔵プルダウン抵抗

図12 - 9 C<sub>SG</sub>によるもれ発光の概念図



## 12.7 全損失の計算方法

μPD784976サブシリーズの消費電力には次の3つがあります。この3つの消費電力の和が全損失 $P_T$  (図12-10参照) 以下となるように設計してください (定格の80%以下での使用を推奨します)。

CPUの消費電力： $V_{DD} (MAX.) \times I_{DD} (MAX.)$  で計算される消費電力です。

出力端子の消費電力：FIP出力端子に最大の電流を流した場合の消費電力です。

プルダウン抵抗の消費電力：FIP出力端子に内蔵するプルダウン抵抗による消費電力です。

図12 - 10 許容全損失 $P_T$  ( $T_A = -40 \sim +85$  )

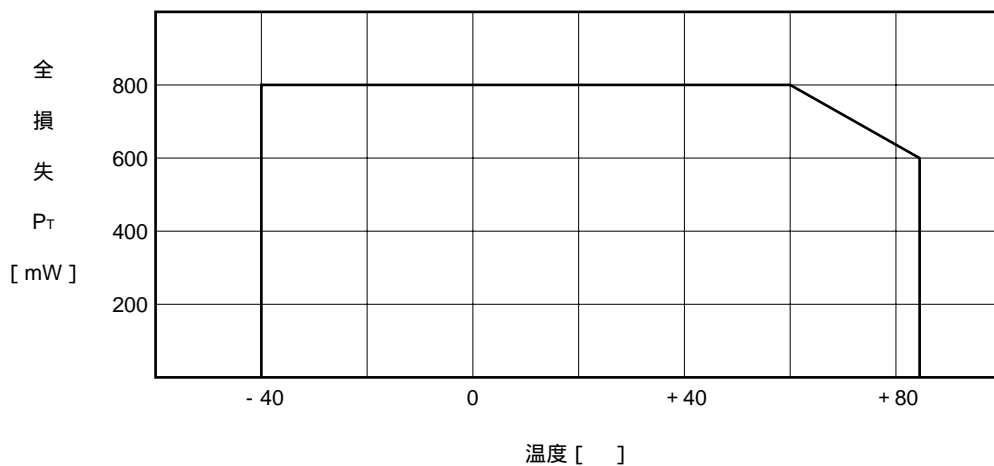


図12 - 11で示す表示例における全消費電力の計算方法を示します。

例 次のような条件を仮定します。

$V_{DD} = 5.5\text{ V}$  , 5.0 MHz発振

電源電流 ( $I_{DD}$ ) = 21.0 mA

FIP出力 : 11グリッド × 10セグメント (ブランキング幅 = 1/16 : FBLK0-FBLK2 = 000Bのとき)

グリッド端子には最大10 mA流れるものとします。

セグメント端子には最大3 mA流れるものとします。

また、キー・スキャン・タイミングではFIP出力端子はオフしているものとします。

FIP出力電圧 : グリッド  $V_{OD} = V_{DD} - 2\text{ V}$  (2 Vの電圧降下があるものとします。)

セグメント  $V_{OD} = V_{DD} - 0.5\text{ V}$  (0.5 Vの電圧降下があるものとします。)

蛍光表示管の電圧 ( $V_{LOAD}$ ) = - 35 V

マスク・オプション・プルダウン抵抗 = 25 k

以上のような条件を計算方法 ~ にあてはめ、全損失を計算します。

CPUの消費電力 :  $5.5\text{ V} \times 21.0\text{ mA} = 115.5\text{ mW}$

出力端子の消費電力 :

$$\begin{aligned} \text{グリッド} \quad (V_{DD} - V_{OD}) \times \frac{\text{各グリッドの電流値合計}}{\text{グリッド数} + 1} \times (1 - \text{ブランキング幅}) = \\ 2\text{ V} \times \frac{10\text{ mA} \times 11\text{グリッド}}{11\text{グリッド} + 1} \times (1 - \frac{1}{16}) = 17.2\text{ mW} \end{aligned}$$

$$\begin{aligned} \text{セグメント} (V_{DD} - V_{OD}) \times \frac{\text{点灯ドットのセグメント電流値合計}}{\text{グリッド数} + 1} \times (1 - \text{ブランキング幅}) = \\ 0.5\text{ V} \times \frac{3\text{ mA} \times 31\text{ドット}}{11\text{グリッド} + 1} \times (1 - \frac{1}{16}) = 3.6\text{ mW} \end{aligned}$$

プルダウン抵抗の消費電力 :

$$\begin{aligned} \text{グリッド} \quad \frac{(V_{OD} - V_{LOAD})^2}{\text{プルダウン抵抗値}} \times \frac{\text{グリッド数}}{\text{グリッド数} + 1} \times (1 - \text{ブランキング幅}) = \\ \frac{(5.5\text{ V} - 2\text{ V} - (-35\text{ V}))^2}{25\text{ k}} \times \frac{11\text{グリッド}}{11\text{グリッド} + 1} \times (1 - \frac{1}{16}) = 50.9\text{ mW} \end{aligned}$$

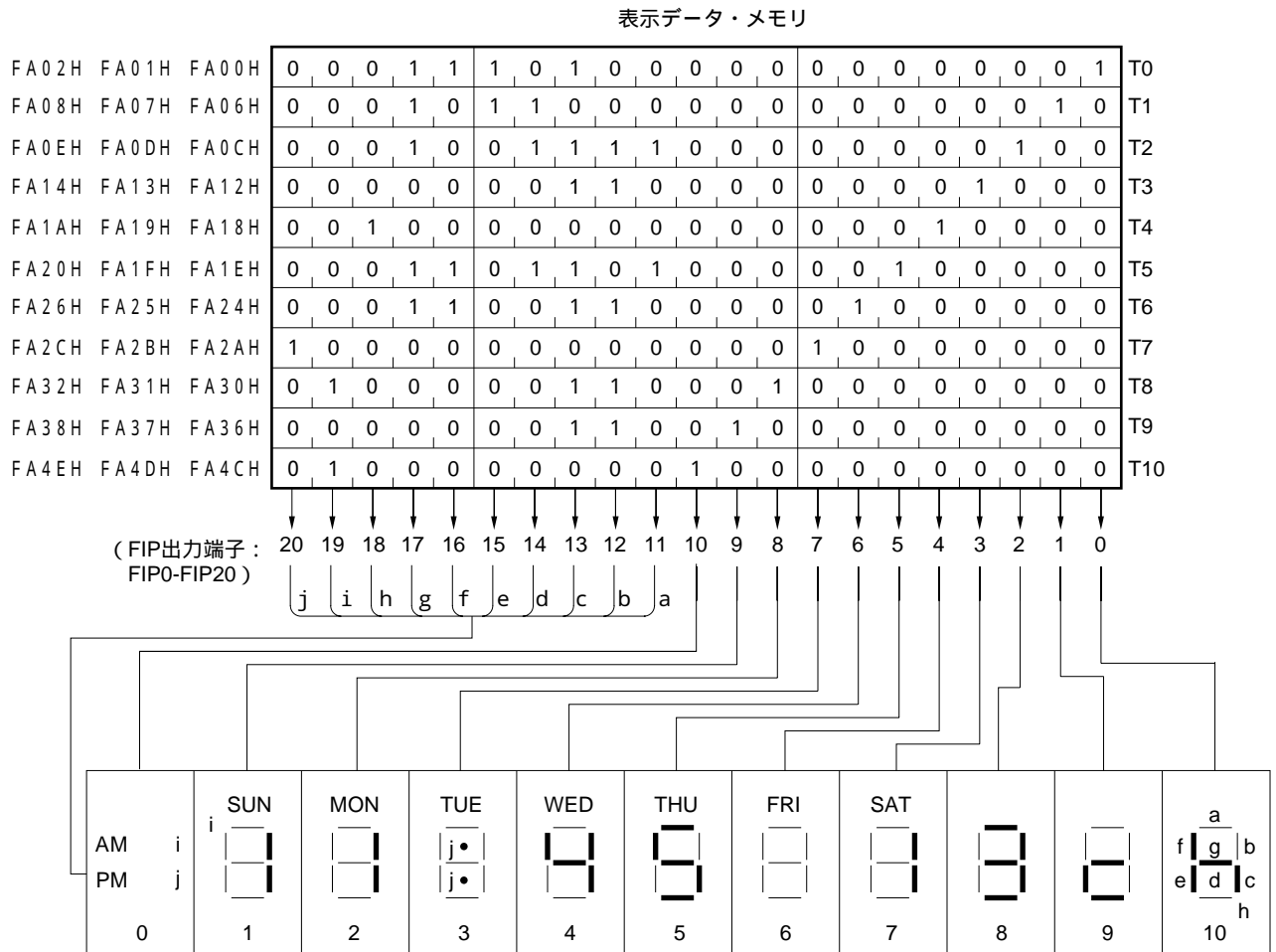
$$\begin{aligned} \text{セグメント} \quad \frac{(V_{OD} - V_{LOAD})^2}{\text{プルダウン抵抗値}} \times \frac{\text{点灯ドット数}}{\text{グリッド数} + 1} \times (1 - \text{ブランキング幅}) = \\ \frac{(5.5\text{ V} - 0.5\text{ V} - (-35\text{ V}))^2}{25\text{ k}} \times \frac{31\text{ドット}}{11\text{グリッド} + 1} \times (1 - \frac{1}{16}) = 155.0\text{ mW} \end{aligned}$$



$$\text{全消費電力} = \quad + \quad + \quad = 115.5 + 17.2 + 3.6 + 50.9 + 155.0 = 342.2 \text{ mW}$$

この例では、全消費電力が図12 - 10に示す許容全損失の定格を越えないので、消費電力は問題ありません。  
 全消費電力が許容全損失の定格を越えた場合には、消費電力を下げる必要があります。消費電力を下げるには、内蔵するプルダウン抵抗の本数を少なくします。

図12 - 11 10セグメント-11桁表示時の表示データ・メモリの内容とFIP出力の関係



〔メモ〕

## 第13章 エッジ検出機能

P64, P65, P67は、立ち上がりエッジ / 立ち下がりエッジをプログラマブルに指定できるエッジ検出機能を持っており、検出したエッジを内部の各ハードウェアへ送っています。

エッジ検出機能は、STOPモード、IDLEモード時を含み、常時動作しています。

### 13.1 制御レジスタ

外部割り込み立ち上がりエッジ許可レジスタ (EGP0)、外部割り込み立ち下がりエッジ許可レジスタ (EGN0)

EGP0, EGN0は、P64, P65, P67端子で検出するエッジの有効エッジを指定するレジスタです。

EGP0, EGN0は、8ビット操作命令または1ビット操作命令で読み出し / 書き込みが可能です。

$\overline{\text{RESET}}$ 入力により00Hになります。

図13 - 1 外部割り込み立ち上がりエッジ許可レジスタ (EGP0)、外部割り込み立ち下がりエッジ許可レジスタ (EGN0) のフォーマット

アドレス : 0FFA0H リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
EGP0	EGP7	0	EGP5	EGP4	0	0	0	0

アドレス : 0FFA2H リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
EGN0	EGN7	0	EGN5	EGN4	0	0	0	0

EGPn	EGNn	INTPm端子の有効エッジ (n = 4, 5, 7, m = 0-2)
0	0	割り込み禁止
0	1	立ち下がりエッジ
1	0	立ち上がりエッジ
1	1	立ち上がり、立ち下がりの両エッジ

**備考** EGP0およびEGN0のビット4, 5, 7は、それぞれINTP0-INTP2端子を制御しています。

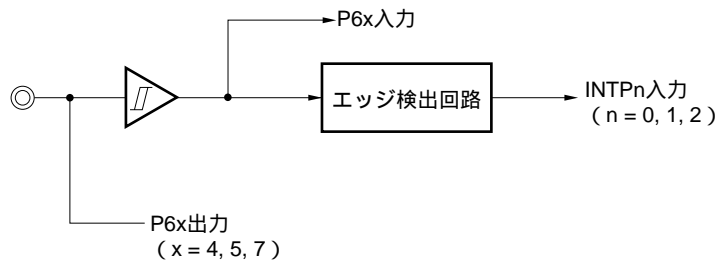
制御される端子	INTP0	INTP1	INTP2
EGP0のビット	EGP4	EGP5	EGP7
EGN0のビット	EGN4	EGN5	EGN7

## 13.2 P64, P65, P67端子のエッジ検出

P64, P65, P67端子のエッジ検出回路構成を図13 - 2 に示します。

P64, P65, P67端子には、アナログ・ディレイによるノイズ除去回路を内蔵しておりません。したがって、端子に有効エッジが入力し、ヒステリシスタイプの入力バッファを通過後直ちにエッジ検出（認識）を行います。

図13 - 2 P64, P65, P67端子のエッジ検出



## 第14章 割り込み機能

μPD784975は、割り込み要求の処理としてベクタ割り込み、コンテキスト・スイッチング、マクロ・サービスの3つのモードを備えています(表14-1参照)。これら3つの処理モードは、プログラムで任意に設定できます。ただし、マクロ・サービスによる割り込み処理の選択は表14-2で示すマクロ・サービス処理モードを備えている割り込み要求ソースに対してのみ選択できます。コンテキスト・スイッチングは、ノンマスカブル割り込みおよびオペランド・エラー割り込みでは選択できません。

また、マスク可能なベクタ割り込みについては、4レベルの優先順位を持った多重処理制御を容易に行うことが可能です。

表14-1 割り込み要求の処理形態

割り込み要求処理モード	処理の主体	PC, PSWの内容	処 理 形 態
ベクタ割り込み	ソフトウェア	スタックへの退避 / 復帰動作を行う	ベクタ・テーブルで指定されたアドレス <sup>注</sup> のサービス・プログラムに分岐し実行
コンテキスト・スイッチング		レジスタ・バンク中の固定エリアへの退避 / 復帰を行う	ベクタ・テーブルで指定されたレジスタ・バンクへの切り替えを自動的に行い、レジスタ・バンク中の固定エリアで指定されたアドレス <sup>注</sup> のサービス・プログラムに分岐し実行
マクロ・サービス	ハードウェア (ファームウェア)	保持	メモリとI/Oとのデータ転送などあらかじめ設定した処理を実行

**注** すべての割り込みサービス・プログラムの先頭アドレスは、ベース領域にある必要があります。ベース領域内にサービス・プログラムの本体が配置できない場合は、ベース領域内にサービス・プログラムへの分岐命令を記述してください。

## 14.1 割り込み要求ソース

μPD784975には、表14-2に示すように16種類の割り込み要求ソースがあり、それぞれに割り込みベクタ・テーブルが割り付けられています。

表14-2 割り込み要求ソースの種類

割り込み要求タイプ	デフォルト優先順位	割り込み要求発生ソース	発生ユニット	割り込み制御レジスタ名	コンテキスト・スイッチング	マクロ・サービス	マクロ・サービス・コントロール・ワード・アドレス	ベクタ・テーブル・アドレス
ソフトウェア	なし	BRK命令の実行	-	-	不可	不可	-	003EH
		BRKCS命令の実行	-	-	可	不可	-	-
オペランド・エラー	なし	MOV STBC, byte命令またはMOV WDM, byte命令, LOCATION命令のオペランド不正	-	-	不可	不可	-	003CH
ノンマスクابل	なし	INTWDT (ウォッチドッグ・タイマのオーバーフロー)	ウォッチドッグ・タイマ	-	不可	不可	-	0004H
マスクابل	0	INTWDTM (ウォッチドッグ・タイマのオーバーフロー)		WDTIC	可	可	0FE06H	0006H
	1	INTP0 (端子入力エッジ検出)	エッジ検出	PIC0			0FE08H	0008H
	2	INTP1 ( " )		PIC1			0FE0AH	000AH
	3	INTP2 ( " )		PIC2			0FE0CH	000CH
	4	INTTM00 (16ビット・タイマ / カウンタ (TM0) とキャプチャ・コンペア・レジスタ (CR00) の一致信号発生)	16ビット・タイマ / イベント・カウンタ	TMIC00			0FE0EH	000EH
	5	INTTM01 (16ビット・タイマ / カウンタ (TM0) とキャプチャ・コンペア・レジスタ (CR01) の一致信号発生)		TMIC01			0FE10H	0010H
	6	INTKS (FIPコントローラ / ドライバからのキー・スキャン・タイミング)	FIPコントローラ / ドライバ	KSIC			0FE12H	0012H
	7	INTCSI0 (CSI0の3線式転送終了)	シリアル・インタフェース	CSIIC0			0FE14H	0014H
	8	INTCSI1 (CSI1の3線式転送終了)		CSIIC1			0FE16H	0016H
	9	INTTM50 (8ビット・タイマ / カウンタ (TM50) と8ビット・コンペア・レジスタ (CR50) の一致)	8ビット PWM タイマ (TM50)	TMIC50			0FE18H	0018H
	10	INTTM51 (8ビット・タイマ / カウンタ (TM51) と8ビット・コンペア・レジスタ (CR51) の一致)	8ビット PWM タイマ (TM51)	TMIC51			0FE1AH	001AH
11	INTAD (A/Dコンバータの変換終了)	A/Dコンバータ	ADIC			0FE1CH	001CH	

備考1. デフォルト優先順位は、固定されている値です。同一優先順位に指定された複数の割り込み要求が、同時に発生した場合に優先される順位を示します。

2. CSI: クロック同期式シリアル・インタフェース

### 14.1.1 ソフトウェア割り込み

ソフトウェアによる割り込みには、ベクタ割り込みを発生するBRK命令とコンテキスト・スイッチングを行うBRKCS命令があります。

ソフトウェアによる割り込みは、割り込み禁止状態でも受け付けられます。割り込み優先順位制御の対象になりません。

### 14.1.2 オペランド・エラー割り込み

MOV STBC, byte命令, MOV WDM, byte命令およびLOCATION命令のオペランドが不正の場合に発生する割り込みです。

オペランド・エラー割り込みは、割り込み禁止状態でも受け付けられます。割り込み優先順位制御の対象になりません。

### 14.1.3 ノンマスカブル割り込み

ノンマスカブル割り込みはウォッチドッグ・タイマによって発生します。

ノンマスカブル割り込みは、割り込み禁止状態であっても無条件<sup>注</sup>に受け付けられます。また、割り込み優先順位制御の対象にならず、すべての割り込みに対して最優先の割り込み要求です。

**注** 同一のノンマスカブル割り込みのサービス・プログラムの実行中と、優先順位の高いノンマスカブル割り込みのサービス・プログラムの実行中を除く

### 14.1.4 マスカブル割り込み

マスカブル割り込みは、割り込みマスク・フラグの設定によってマスク制御を受ける割り込みです。また、プログラム・ステータス・ワード (PSW) のIEフラグによって、マスカブル割り込み全体に対して受け付けの許可/禁止の指定ができます。

マスカブル割り込みは、通常のベクタ割り込みのほかに、コンテキスト・スイッチングやマクロ・サービスによって受け付けることが可能です (マクロ・サービスについては一部の割り込みを除く。表14-2参照)。

マスカブル割り込みは、表14-2のように同一優先順位を持つ複数の割り込み要求が同時に発生している場合の優先順位が決められています (デフォルト優先順位)。また、割り込み優先順位を4レベルのグループに分け、多重処理の制御を行うことができます。ただしマクロ・サービスは、優先順位制御やIEフラグとは無関係に受け付けられます。

## 14.2 割り込み処理モード

μPD784975の割り込み処理モードには、次の3つのモードがあります。

- ・ベクタ割り込み処理
- ・マクロ・サービス
- ・コンテキスト・スイッチング

### 14.2.1 ベクタ割り込み処理

割り込みが受け付けられると、プログラム・カウンタ（PC）、プログラム・ステータス・ワード（PSW）を自動的にスタック領域に退避させ、ベクタ・テーブルに格納されているデータで示されるアドレスに分岐し、割り込み処理ルーチンを実行します。

### 14.2.2 マクロ・サービス

割り込みが受け付けられると、CPUの実行を一時中断し、ハードウェア的にデータ転送を行います。マクロ・サービスは、CPUを介さずに行われるため、プログラム・カウンタ（PC）、プログラム・ステータス・ワード（PSW）などのCPUステータスを退避/復帰する必要がありません。したがって、CPUのサービス時間を向上させる大きな効果があります（14.8 マクロ・サービス機能参照）。

### 14.2.3 コンテキスト・スイッチング

割り込みが受け付けられると、ハードウェアにより所定のレジスタ・バンクを選択し、レジスタ・バンク内にあらかじめ設定しておいたベクタ・アドレスに分岐すると同時に、現在のプログラム・カウンタ（PC）、プログラム・ステータス・ワード（PSW）をレジスタ・バンク内に退避します（14.4.2 BRKCS命令によるソフトウェア割り込み（ソフトウェア・コンテキスト・スイッチング）の受け付け動作、14.7.2 コンテキスト・スイッチング参照）。

**備考** コンテキストとは、プログラムの実行において、そのプログラムからアクセス可能なCPUのレジスタのことです。このレジスタには、汎用レジスタ、プログラム・カウンタ（PC）、プログラム・ステータス・ワード（PSW）、スタック・ポインタ（SP）が含まれます。



### 14.3 割り込み処理制御レジスタ

μPD784975の割り込み処理は、割り込み処理の指定を行う各種制御レジスタにより、各割り込み要求ごとに制御されます。表14 - 3に、割り込みの制御レジスタ一覧を示します。

表14 - 3 制御レジスタ一覧

レジスタ名	略号	機能
割り込み制御レジスタ	WDTIC, PIC0, PIC1, PIC2, CSIIC0, CSIIC1, TMIC00, TMIC01, KSIC, TMIC50, TMIC51, ADIC	各割り込み要求の発生の記憶, マスク制御, ベクタ割り込み処理またはマクロ・サービス処理の指定, コンテキスト・スイッチング機能の許可/禁止, 優先順位の指定を行うレジスタ
割り込みマスク・レジスタ	MK0 (MK0L, MK0H)	マスカブル割り込み要求のマスク制御 割り込み制御レジスタ内のマスク制御フラグと連動ワード・アクセスまたはバイト・アクセスが可能
インサービス・プライオリティ・レジスタ	ISPR	現在受け付け中の割り込み要求について優先順位を記憶
割り込みモード・コントロール・レジスタ	IMC	優先順位を最下位レベル(レベル3)に指定したマスカブル割り込みのネスティングを制御
割り込み選択コントロール・レジスタ	SNMI	ウォッチドッグ・タイマからの割り込み信号を, マスカブル割り込みとして使用するか, ノンマスカブル割り込みとして使用するかの選択を行うレジスタ
ウォッチドッグ・タイマ・モード・レジスタ	WDM	ウォッチドッグ・タイマの動作を制御するレジスタ

割り込み制御レジスタは、おのおのの割り込み要因ごとに制御レジスタが割り当てられています。各レジスタのフラグは、それぞれのレジスタ内のビット位置に対応した内容の制御を行います。

表14 - 4に、各割り込み要求信号に対応する割り込み制御レジスタのフラグ名称を示します。

表14 - 4 割り込み要求ソースに対する割り込み制御レジスタのフラグー覧

デフォルト 優先順位	割り込み 要求信号	割り込み制御レジスタ					
			割り込み 要求フラグ	割り込み マスク・フラグ	マクロ・サービス 許可フラグ	優先順位 指定フラグ	コンテキスト・スウィ ッチング・フラグ
0	INTWDT	WDTIC	WDTIF	WDTMK	WDTISM	WDTPR0 WDTPR1	WDCSE
1	INTP0	PIC0	PIF0	PMK0	PISM0	PPR00 PPR01	PCSE0
2	INTP1	PIC1	PIF1	PMK1	PISM1	PPR10 PPR11	PCSE1
3	INTP2	PIC2	PIF2	PMK2	PISM2	PPR20 PPR21	PCSE2
4	INTTM00	TMIC00	TMIF00	TMMK00	TMISM00	TMPR000 TMPR001	TMCSE00
5	INTTM01	TMIC01	TMIF01	TMMK01	TMISM01	TMPR010 TMPR011	TMCSE01
6	INTKS	KSIC	KSIF	KSMK	KSISM	KSPR0 KSPR1	KSCSE
7	INTCSI0	CSIIC0	CSIF0	CSIMK0	CSIISM0	CSIPR00 CSIPR01	CSICSE0
8	INTCSI1	CSIIC1	CSIF1	CSIMK1	CSIISM1	CSIPR10 CSIPR11	CSICSE1
9	INTTM50	TMIC50	TMIF50	TMMK50	TMISM50	TMPR500 TMPR501	TMCSE50
10	INTTM51	TMIC51	TMIF51	TMMK51	TMISM51	TMPR510 TMPR511	TMCSE51
11	INTAD	ADIC	ADIF	ADMK	ADISM	ADPR0 ADPR1	ADCSE

### 14.3.1 割り込み制御レジスタ

割り込み制御レジスタは、各割り込み要因ごとに割り当てられており、対応する割り込み要求の優先順位制御や、マスク制御などを行うレジスタです。図14 - 1に、割り込み制御レジスタのフォーマットを示します。

#### (1) 優先順位指定フラグ (××PR1, ××PR0)

優先順位指定フラグは、27種類のマスカブル割り込みに対して、割り込み要因ごとの優先順位を指定するフラグです。

優先順位レベルは、4レベルまで指定でき、複数の割り込み要因を同じレベルに指定することができます。マスカブル割り込み要因のうち、レベル0が最も優先順位が高くなります。

優先順位レベルの等しい割り込み要因同士内で、複数の割り込み要求が同時に発生した場合は、デフォルト優先順位の順に受け付けられます。

ソフトウェアにより、ビット単位で操作ができます。

$\overline{\text{RESET}}$ 入力により、全ビットが“1”になります。

#### (2) コンテキスト・スイッチング許可フラグ (××CSE)

コンテキスト・スイッチング許可フラグは、マスカブル割り込み要求をコンテキスト・スイッチングにより処理することを指定するフラグです。

コンテキスト・スイッチングは、あらかじめ指定してあるレジスタ・バンクをハードウェア的に選択し、レジスタ・バンク内にあらかじめストアしておいたベクタ・アドレスへ分岐すると同時に、現在のプログラム・カウンタ (PC)、プログラム・ステータス・ワード (PSW) の内容をレジスタ・バンクに退避する機能です。

通常のベクタ割り込み処理よりも高速に割り込み処理の実行を開始できるので、リアルタイム処理に適しています。

ソフトウェアにより、ビット単位で操作ができます。

$\overline{\text{RESET}}$ 入力により、全ビットが“0”になります。

#### (3) マクロ・サービス許可フラグ (××ISM)

マクロ・サービス許可フラグは、各フラグに対応する割り込み要求について、ベクタ割り込みまたはコンテキスト・スイッチングで処理するか、マクロ・サービスで処理するかを指定するフラグです。

マクロ・サービス処理を選択していた場合に、マクロ・サービスが終了すると (マクロ・サービス・カウンタが0になると)、マクロ・サービス許可フラグはハードウェアで自動的にクリア (0) されます (ベクタ割り込み処理 / コンテキスト・スイッチング処理)。

ソフトウェアにより、ビット単位で操作ができます。

$\overline{\text{RESET}}$ 入力により、全ビットが“0”になります。

**(4) 割り込みマスク・フラグ (××MK)**

割り込みマスク・フラグは、各フラグに対応する割り込み要求について、ベクタ割り込み処理、マクロ・サービス処理の許可/禁止を指定するフラグです。

割り込みマスク・フラグの内容は、割り込み処理の起動などにより変化することはありません。また、割り込みマスク・フラグの内容と、割り込みマスク・レジスタとは同一の内容となります(14.3.2 **割り込みマスク・レジスタ (MK0, MK1)** 参照)。

マクロ・サービス処理要求もマスク制御の対象で、マクロ・サービス要求もこのフラグによりマスクできます。

ソフトウェアにより、操作ができます。

$\overline{\text{RESET}}$ 入力により、全ビットが“1”になります。

**(5) 割り込み要求フラグ (××IF)**

割り込み要求フラグは、各フラグに対応する割り込み要求の発生でセット(1)されます。割り込みが受け付けられたときにハードウェアにより自動的にクリア(0)されます。

ソフトウェアにより、操作ができます。

$\overline{\text{RESET}}$ 入力により、全ビットが“0”になります。

図14 - 1 割り込み制御レジスタ (x x ICn) (1/2)

アドレス : 0FFE0H-0FFE8H リセット時 : 43H R/W

略号	⑦	⑥	⑤	④	3	2	①	①
WDTIC	WDTIF	WDTMK	WDTISM	WDCSE	0	0	WDTPR1	WDTPR0
PIC0	PIF0	PMK0	PISM0	PCSE0	0	0	PPR01	PPR00
PIC1	PIF1	PMK1	PISM1	PCSE1	0	0	PPR11	PPR10
PIC2	PIF2	PMK2	PISM2	PCSE2	0	0	PPR21	PPR20
TMIC00	TMIF00	TMIK00	TMISM00	TMCSE00	0	0	TMPR001	TMPR000
TMIC01	TMIF01	TMMK01	TMISM01	TMCSE01	0	0	TMPR011	TMPR010
KSIC	KSIF	KSMK	KSISM	KSCSE	0	0	KSPR1	KSPR0
CSII00	CSIIF0	CSIMK0	CSIISM0	CSICSE0	0	0	CSIPR01	CSIPR00
CSII01	CSIIF1	CSIMK1	CSIISM1	CSICSE1	0	0	CSIPR11	CSIPR10

x x IFn	割り込み要求発生の有無
0	割り込み要求なし (割り込み信号が発生していない)
1	割り込み要求状態 (割り込み信号が発生)

x x MKn	割り込み処理の許可 / 禁止
0	割り込み処理を許可
1	割り込み処理を禁止

x x ISMn	割り込み処理形態の指定
0	ベクタ割り込み処理 / コンテキスト・スイッチング処理
1	マクロ・サービス処理

x x CSEn	コンテキスト・スイッチング処理の指定
0	ベクタ割り込みで処理
1	コンテキスト・スイッチングで処理

x x PRn1	x x PRn0	割り込み要求の優先順位指定
0	0	優先順位 0 (最優先順位)
0	1	優先順位 1
1	0	優先順位 2
1	1	優先順位 3

図14 - 1 割り込み制御レジスタ (x x ICn) (2/2)

アドレス : 0FFE9H-0FFEBH リセット時 : 43H R/W

略号	⑦	⑥	⑤	④	3	2	①	①
TMIC50	TMIF50	TMMK50	TMISM50	TMCSE50	0	0	TMPR501	TMPR500
TMIC51	TMIF51	TMMK51	TMISM51	TMCSE51	0	0	TMPR511	TMPR510
ADIC	ADIF	ADMK	ADISM	ADCSE	0	0	ADPR1	ADPR0

x x IFn	割り込み要求発生の有無
0	割り込み要求なし (割り込み信号が発生していない)
1	割り込み要求状態 (割り込み信号が発生)

x x MKn	割り込み処理の許可 / 禁止
0	割り込み処理を許可
1	割り込み処理を禁止

x x ISMn	割り込み処理形態の指定
0	ベクタ割り込み処理 / コンテキスト・スイッチング処理
1	マクロ・サービス処理

x x CSEn	コンテキスト・スイッチング処理の指定
0	ベクタ割り込みで処理
1	コンテキスト・スイッチングで処理

x x PRn1	x x PRn0	割り込み要求の優先順位指定
0	0	優先順位 0 (最優先順位)
0	1	優先順位 1
1	0	優先順位 2
1	1	優先順位 3

### 14.3.2 割り込みマスク・レジスタ (MK0)

MK0は、割り込みマスク・フラグで構成されるレジスタです。MK0は16ビット・レジスタで、16ビット単位で操作できるほか、MK0はMK0L、MK0Hとして、8ビット単位でも操作できます。

また、MK0の各ビットは、ビット操作命令により1ビット単位で操作ができます。各割り込みマスク・フラグは、対応する割り込み要求の許可/禁止を制御します。

割り込みマスク・フラグがセット(1)されていると、該当する割り込み要求の受け付けは禁止されます。

割り込みマスク・フラグがクリア(0)されていると、該当する割り込み要求は、ベクタ割り込みまたはマクロ・サービスとして受け付けが可能な状態になります。

MK0中の各割り込みマスク・フラグは、割り込み制御レジスタ中の割り込みマスク・フラグと同一のフラグです。割り込みのマスクに関する制御を一括して行うために、MK0を用意しています。

$\overline{\text{RESET}}$ 入力によりMK0はFFFFHになり、すべてのマスクブル割り込みは禁止されます。

図14 - 2 割り込みマスク・レジスタ (MK0) のフォーマット

バイト・アクセス時

アドレス : 0FFACH, 0FFADH リセット時 : FFH R/W

略号	7	6	5	4	3	2	1	0
MK0L	CSIMK0	KSMK	TMMK01	TMMK00	PMK2	PMK1	PMK0	WDTMK
MK0H	1	1	1	1	ADMK	TMMK51	TMMK50	CSIMK1

x x MKn	割り込み要求の許可 / 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

ワード・アクセス時

アドレス : 0FFACH リセット時 : 0FFFFH R/W

略号	15	14	13	12	11	10	9	8
MK0	1	1	1	1	ADMK	TMMK51	TMMK50	CSIMK1
	7	6	5	4	3	2	1	0
	CSIMK0	KSMK	TMMK01	TMMK00	PMK2	PMK1	PMK0	WDTMK

x x MKn	割り込み要求の許可 / 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



### 14.3.3 インサース・プライオリティ・レジスタ (ISPR)

ISPRは、現在処理中のマスカブル割り込みの優先順位レベルと処理中のノンマスカブル割り込みを示すレジスタです。マスカブル割り込み要求が受け付けられると、その割り込み要求の優先順位レベルに対応するビットがセット ( 1 ) され、サービス・プログラムが終了するまで保持されます。ノンマスカブル割り込みが受け付けられると、そのノンマスカブル割り込みに対応したビットがセット ( 1 ) され、サービス・プログラムが終了するまで保持されます。

RETI命令またはRETCS命令の実行の際、ISPR内でセット ( 1 ) されているビットのうち、最も優先順位の高い割り込み要求に対応するビットが、ハードウェアにより自動的にクリア ( 0 ) されます。

RETB命令およびRETCSB命令の実行では、ISPRの内容は変化しません。

RESET入力により、00Hになります。

図14 - 3 インサース・プライオリティ・レジスタ (ISPR) のフォーマット

アドレス : 0FFA8H    リセット時 : 00H    R

略号	7	6	5	4	3	2	1	0
ISPR	0	WDTS	0	0	ISPR3	ISPR2	ISPR1	ISPR0

WDTS	ウォッチドッグ・タイマ割り込み処理状態
0	ウォッチドッグ・タイマ割り込み ( ノンマスカブル割り込み : INTWDT ) を受け付けていない
1	ウォッチドッグ・タイマ割り込み ( ノンマスカブル割り込み : INTWDT ) を受け付け中

ISPRn	優先順位のレベル ( n = 0-3 )
0	優先順位nの割り込みを受け付けていない
1	優先順位nの割り込みを受け付け中

**注意** インサース・プライオリティ・レジスタ ( ISPR ) は、読み出しのみ可能です。書き込みを行うと、誤動作する恐れがあります。

### 14.3.4 割り込みモード・コントロール・レジスタ (IMC)

IMCは、PRSLフラグで構成されるレジスタです。PRSLフラグは、優先順位が最下位レベル（レベル3）に指定されているマスクブル割り込みのネスティングの許可/禁止を指定します。

IMCを操作する場合は、誤動作を防ぐために割り込み禁止状態（DI状態）に設定してから行ってください。

8ビット操作命令とビット操作命令で、読み出し/書き込みが可能です。

$\overline{\text{RESET}}$ 入力により、80Hになります。

図14 - 4 割り込みモード・コントロール・レジスタ (IMC) のフォーマット

アドレス：0FFAAH    リセット時：80H    R/W

略号	7	6	5	4	3	2	1	0
IMC	PRSL	0	0	0	0	0	0	0

PRSL	マスクブル割り込み（最下位レベル）についてのネスティング動作の制御
0	レベル3（最下位レベル）に設定された割り込み間でのネスティング可能
1	レベル3（最下位レベル）に設定された割り込み間でのネスティング禁止

### 14.3.5 ウォッチドッグ・タイマ・モード・レジスタ (WDM)

WDMは、専用命令によってのみ書き込みが可能です。この専用命令は、MOV WDM, byte命令で特殊なコード構成(4バイト)になっています。3バイト目と4バイト目のオペコードが互いに1の補数の場合のみ、書き込みが行われます。

また、3バイト目と4バイト目のオペコードが互いに1の補数でなければ書き込みが行われず、オペランド・エラー割り込みが発生します。この場合、スタック領域に退避されるリターン・アドレスは、エラーの原因となった命令のアドレスです。したがって、スタック領域に退避されたリターン・アドレスにより、エラーの原因となったアドレスを知ることができます。

なお、単純にオペランド・エラーからRETB命令で復帰を行うと、無限ループとなります。

オペランド・エラー割り込みは、プログラムが暴走している場合にのみ発生する(NEC製アセンブラRA78K4では、MOV WDM, byteと記述された場合、正しい専用命令しか生成しません)ので、システムの初期化をプログラムで行うようにしてください。

他の書き込み命令(MOV WDM, AやAND WDM, byteやSET1 WDM.7など)は無視され、何の動作も行いません。すなわち、WDMへの書き込みは行われず、また、オペランド・エラー割り込みのような割り込みも発生しません。

WDMの読み出しは、データ転送命令によりいつでもできます。

$\overline{\text{RESET}}$ 入力により、00Hになります。

図14-5 ウォッチドッグ・タイマ・モード・レジスタ (WDM) のフォーマット



**注意** ウォッチドッグ・タイマ・モード・レジスタ (WDM) は、専用命令 (MOV WDM, byte) でのみ書き込みが可能です。

### 14.3.6 割り込み選択コントロール・レジスタ (SNMI)

SNMIはウォッチドッグ・タイマからの割り込み要求信号を、マスカブル割り込みとして使用するか、ノンマスカブル割り込みとして使用するかを選択するレジスタです。

このレジスタのビットは、リセット後一度しかセット (1) することができず、ビットのクリア (0) は、リセットで行います。

SNMIは、1ビット・メモリ操作命令または8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$  入力により、00Hになります。

図14 - 6 割り込み選択コントロール・レジスタ (SNMI) のフォーマット

アドレス : 0FFA9H リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
SNMI	0	0	0	0	0	0	SWDT	0

SWDT	ウォッチドッグ・タイマ割り込みの選択
0	ノンマスカブル割り込み (INTWDT) として使用。 割り込みマスク・レジスタによる割り込み処理の禁止はできません。
1	マスカブル割り込み (INTWDTM) として使用。 ベクタ割り込み、マクロ・サービスが利用できます。割り込みマスク・レジスタにより、割り込み処理を禁止することができます。

### 14.3.7 プログラム・ステータス・ワード (PSW)

PSWは、命令の実行結果や割り込み要求に対する現在の状態を保持するレジスタです。マスカブル割り込みの許可/禁止を設定するIEフラグがPSWの下位8ビット (PSWL) にマッピングされています。

PSWLは、8ビット単位で読み出し/書き込み操作ができるほか、ビット操作命令や専用命令 (EI, DI) により操作ができます。

ベクタ割り込み受け付け時、BRK命令実行時にはスタックに退避され、IEフラグはクリア (0) されます。また、PUSH PSW命令によってもスタックに退避されます。RETI, RETB, POP PSW命令により、スタックから復帰します。

コンテキスト・スイッチングおよびBRKCS命令実行時には、レジスタ・バンクの固定エリアに退避され、IEフラグはクリア (0) されます。また、RETCSI, RETCSB命令により、レジスタ・バンク中の固定エリアから復帰します。

$\overline{\text{RESET}}$ 入力により、PSWLは00Hとなります。

図14-7 プログラム・ステータス・ワード (PSWL) のフォーマット

リセット時: 00H

略号	7	6	5	4	3	2	1	0
PSWL	S	Z	RSS	AC	IE	P/V	0	CY

S	通常の命令実行時に使用
Z	
RSS	
AC	

IE	割り込み受け付けの許可/禁止
0	禁止
1	許可

P/V	通常の命令実行時に使用
CY	

## 14.4 ソフトウェア割り込みの受け付け動作

ソフトウェア割り込みはBRK, BRKCS命令の実行により受け付けられます。ソフトウェア割り込みは禁止することができません。

### 14.4.1 BRK命令によるソフトウェア割り込みの受け付け動作

BRK命令を実行すると、プログラム・ステータス・ワード (PSW), プログラム・カウンタ (PC) の順にスタックに退避し、IEフラグをクリア (0) し、ベクタ・テーブル (003EH, 003FH) の内容をPCの下位

16ビットに、0000BをPCの上位4ビットにロードして分岐します（サービス・プログラムの先頭は、ベース領域になければいけません）。

BRK命令によるソフトウェア割り込みからの復帰にはRETB命令を使用します。

**注意** BRK命令によるソフトウェア割り込みからの復帰にRETI命令を使用してはいけません。  
RETB命令を使用してください。

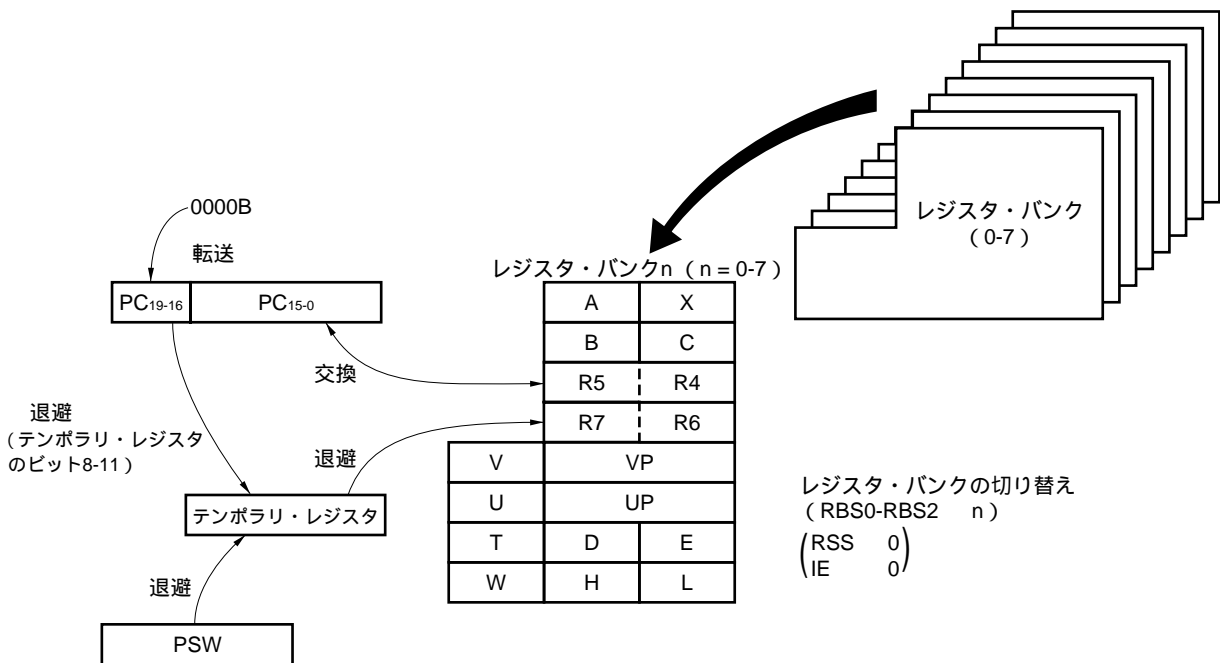
### 14.4.2 BRKCS命令によるソフトウェア割り込み（ソフトウェア・コンテキスト・スイッチング）の受け付け動作

BRKCS命令の実行により、コンテキスト・スイッチング機能を起動できます。

BRKCS命令のオペランドで、コンテキスト・スイッチング後のレジスタ・バンクを指定します。

BRKCS命令を実行すると、指定したレジスタ・バンク内にあらかじめストアしておいた割り込みサービス・プログラムの先頭アドレス（ベース領域内に限ります）に分岐すると同時に、それまでのプログラム・カウンタ（PC）、プログラム・ステータス・ワード（PSW）の内容をレジスタ・バンク内に退避します。

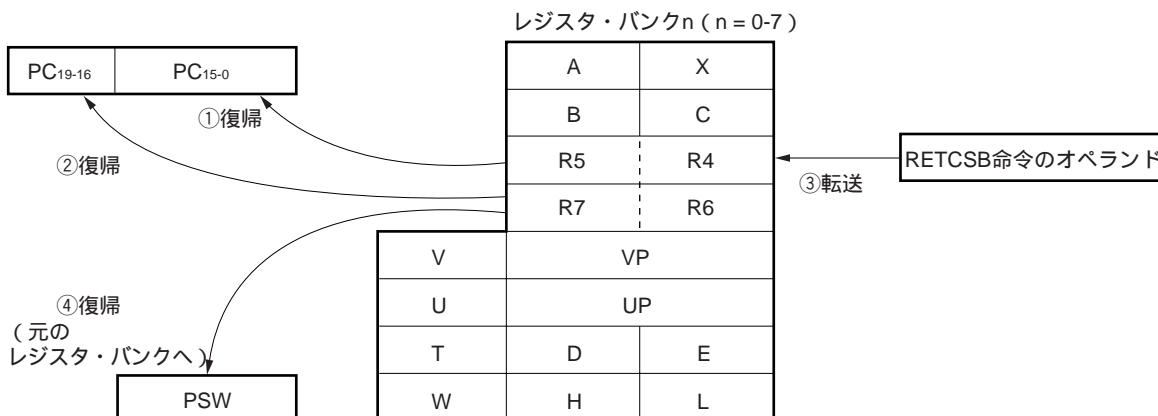
図14-8 BRKCS命令の実行によるコンテキスト・スイッチング動作



BRKCS命令によるソフトウェア割り込みからの復帰には、RETCSB命令を使用します。RETCSB命令では、次にBRKCS命令でコンテキスト・スイッチングを行うときの割り込みサービス・プログラムの先頭アドレスを指定する必要があります。この割り込みサービス・プログラムの先頭アドレスは、ベース領域内になければいけません。

**注意** BRKCS命令によるソフトウェア割り込みからの復帰にRETCS命令を使用してはいけません。  
RETCSB命令を使用してください。

図14 - 9 BRKCS命令によるソフトウェア割り込みからの復帰動作 (RETCSB命令の動作)



## 14.5 オペランド・エラー割り込みの受け付け動作

オペランド・エラー割り込みは、MOV STBC, byte命令およびMOV WDM, byte命令、LOCATION命令のオペランドの3バイト目の全ビットを反転させたデータが、オペランドの4バイト目と一致しなかった場合に発生します。オペランド・エラー割り込みは禁止することができません。

オペランド・エラー割り込みが発生すると、プログラム・ステータス・ワード (PSW) とエラーが発生した命令の先頭アドレスをスタックに退避し、IEフラグをクリア (0) し、ベクタ・テーブルの値をプログラム・カウンタ (PC) にロードし、分岐します (ベース領域にかぎる)。

スタックに退避されるアドレスが、エラーの発生した命令の先頭アドレスとなっているので、単にオペランド・エラー割り込みサービス・プログラムの最後にRETB命令を記述しただけでは、再度オペランド・エラー割り込みが発生してしまいます。したがって、スタック中のアドレスを加工するか、14.12 割り込み機能を初期状態に戻す方法を参考にプログラムを初期状態に戻してください。

## 14.6 ノンマスカブル割り込みの受け付け動作

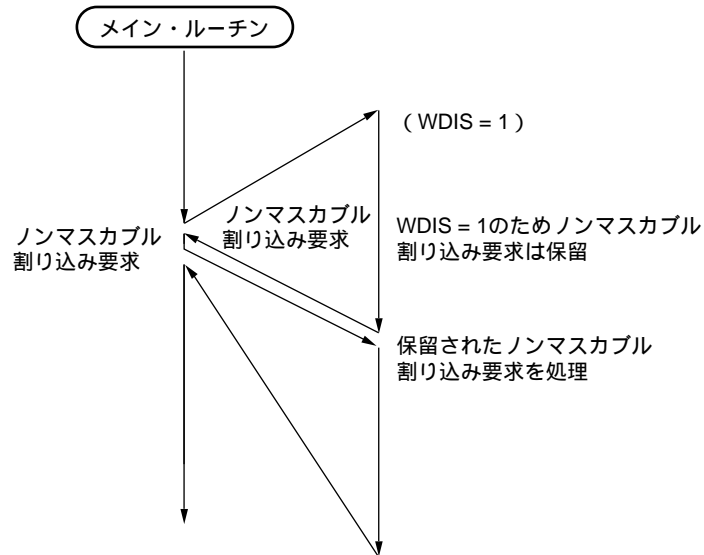
ノンマスカブル割り込みは、割り込み禁止状態であっても受け付けられます。

ノンマスカブル割り込み要求は、14.9 割り込み要求およびマクロ・サービスが一時的に保留される場合に記述されている状態でなければ、ただちに受け付けられます。ノンマスカブル割り込み要求が受け付けられると、プログラム・ステータス・ワード (PSW)、プログラム・カウンタ (PC) の順にスタックへの退避を行い、PSWのIEフラグをクリア (0) し、受け付けたノンマスカブル割り込みに対応するインサービス・プライオリティ・レジスタ (ISPR) のビットをセット (1) し、ベクタ・テーブルの内容をPCへロードし、分岐します。セット (1) されるISPRのビットは、WDTSビットになります。なお、ノンマスカブル割り込みサービス・プログラム実行中に、同一のノンマスカブル割り込み要求が2回以上発生しても、ノンマスカブル割り込みサービス・プログラムの終了後に受け付けられるノンマスカブル割り込みは1回だけになります。

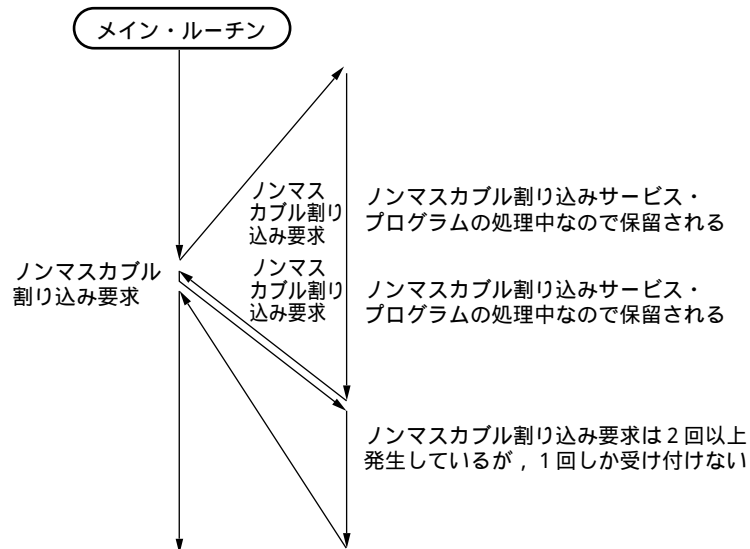


図14 - 10 ノンマスカブル割り込み要求の受け付け動作

( a ) ノンマスカブル割り込みサービス・プログラム実行中に新たなノンマスカブル割り込み要求が発生した場合



( b ) ノンマスカブル割り込みサービス・プログラム実行中に新たに2回のノンマスカブル割り込み要求が発生した場合



- 注意1. ノンマスクابل割り込みサービス・プログラム中でもマクロ・サービスの要求は受け付けられ、処理されます。ノンマスクابل割り込みサービス・プログラム中でマクロ・サービス処理を行いたくない場合は、ノンマスクابل割り込みサービス・プログラム中で割り込みマスク・レジスタを操作して、マクロ・サービスが発生しないようにしてください。
2. ノンマスクابل割り込みからの復帰には、必ずRETI命令を使用してください。他の命令では、以降の割り込みの受け付けが正常に行われません。ノンマスクابل割り込み受け付け後、初期状態からプログラムを再開したい場合は、14.12 割り込み機能を初期状態に戻す方法を参照してください。
3. ノンマスクابل割り込みは、ノンマスクابل割り込みサービス・プログラム実行中（優先順位の低いノンマスクابل割り込みサービス・プログラム実行中に、優先順位の高いノンマスクابل割り込みの要求が発生した場合を除く）および14.9に示す特定命令の実行後の一定期間を除いては必ず受け付けられます。したがって、特にリセット解除後などのスタック・ポインタ（SP）の値が不定の場合でもノンマスクابل割り込みを受け付けます。このときのSPの値によっては特殊機能レジスタ（SFR）の書き込みを禁止しているアドレス（3.2.3 特殊機能レジスタ（SFR）の表3-3参照）へプログラム・カウンタ（PC）およびプログラム・ステータス・ワード（PSW）を書き込んでしまい、CPUがデッドロックしてしまったり、端子から予期しない信号を出力したり、RAMが実装されていないアドレスへPCやPSWを書き込むことにより、ノンマスクابل割り込み処理プログラムからメイン・ルーチンへ正常に戻れず暴走したりする場合があります。
- したがって、 $\overline{\text{RESET}}$ 解除後のプログラムは必ず次のようにしてください。

```
CSEG AT 0
DW  STRT
CSEG BASE
STRT :
LOCATION 0FH ; or LOCATION 0
MOVG SP, imm24
```

## 14.7 マスカブル割り込みの受け付け動作

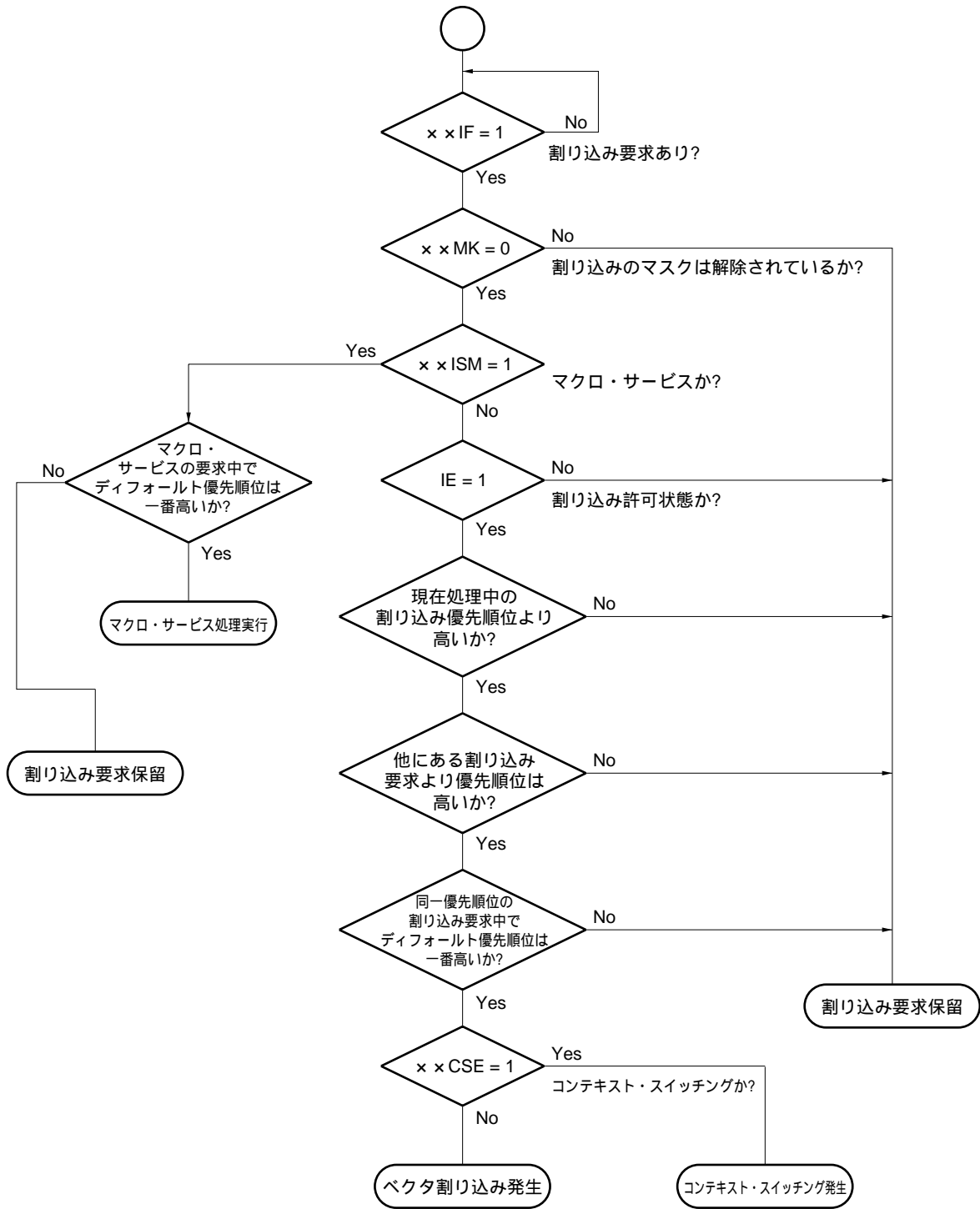
マスカブル割り込みは、割り込み要求フラグがセット（1）され、その割り込みのマスク・フラグがクリア（0）されていると受け付けが可能な状態になります。マクロ・サービスで処理を行う場合は、ただちに受け付けられ、マクロ・サービスによる処理が行われます。ベクタ割り込みまたはコンテキスト・スイッチングの場合は、割り込み許可状態（IEフラグがセット（1）されているとき）で、その割り込みの優先順位が、受け付け可能な優先順位であれば受け付けます。

マスカブル割り込みの要求が同時に発生した場合は、優先順位指定フラグで高優先順位に指定されているものから受け付けられます。また、同一優先順に指定されている場合はデフォルト優先順位に従います。

保留された割り込みは受け付け可能な状態になると受け付けられます。

図14 - 11に、割り込み受け付けのアルゴリズムを示します。

図14 - 11 割り込み要求受け付け処理アルゴリズム



### 14.7.1 ベクタ割り込み

ベクタ割り込みによるマスカブル割り込みの要求が受け付けられると、プログラム・ステータス・ワード (PSW)、プログラム・カウンタ (PC) の順にスタックに退避し、IEフラグをクリア (0) (割り込み禁止状態) にし、受け付けた割り込みの優先順位に該当するインサービス・プライオリティ・レジスタ (ISPR) のビットをセット (1) します。さらに、割り込み要求ごとに決められたベクタ・テーブル中のデータをPCへロードし、分岐します。ベクタ割り込みからの復帰は、RETI命令で行います。

**注意** マスカブル割り込みをベクタ割り込みで受け付けた場合は、必ずRETI命令で復帰してください。他の命令では、以降の割り込みに関する動作が正常に行われません。

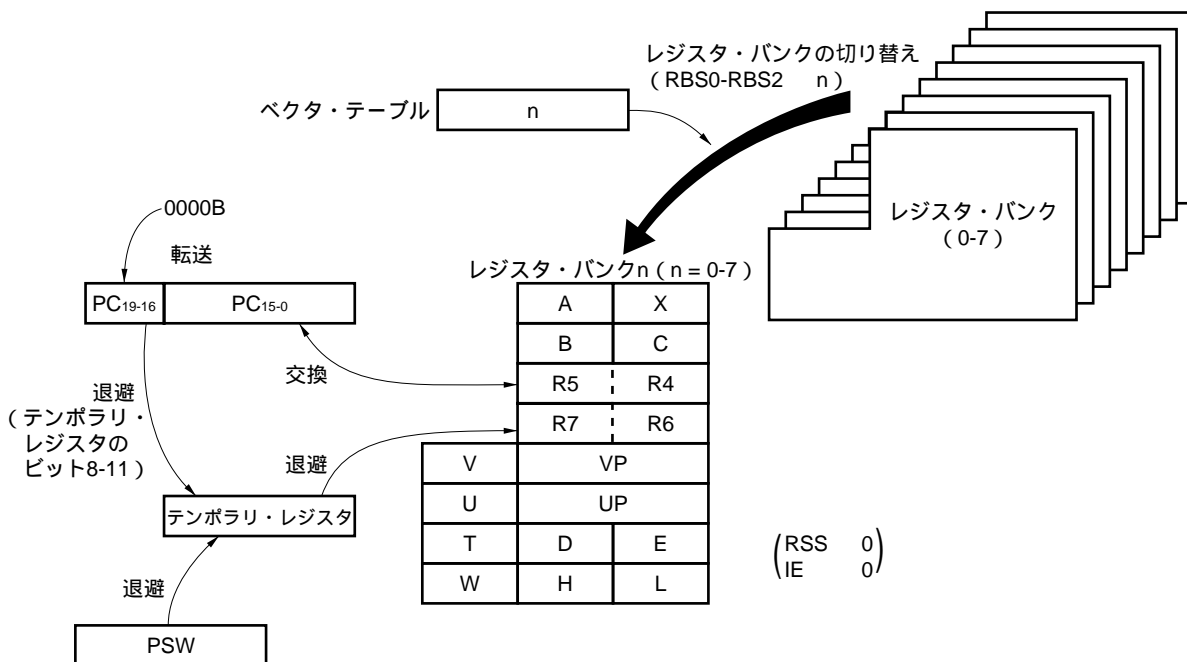
### 14.7.2 コンテキスト・スイッチング

割り込み制御レジスタのコンテキスト・スイッチング許可フラグをセット (1) することにより、コンテキスト・スイッチング機能の起動を許可します。

コンテキスト・スイッチング機能が許可されている割り込み要求が受け付けられると、対応するベクタ・テーブル・アドレスの下位アドレス (偶数アドレス) の下位3ビットで指定されるレジスタ・バンクが選択されます。

選択されたレジスタ・バンク内にあらかじめストアしておいたベクタ・アドレスをプログラム・カウンタ (PC) に転送すると同時に、それまでのPC、プログラム・ステータス・ワード (PSW) の内容をレジスタ・バンク内に退避し、割り込みサービス・プログラムに分岐します。

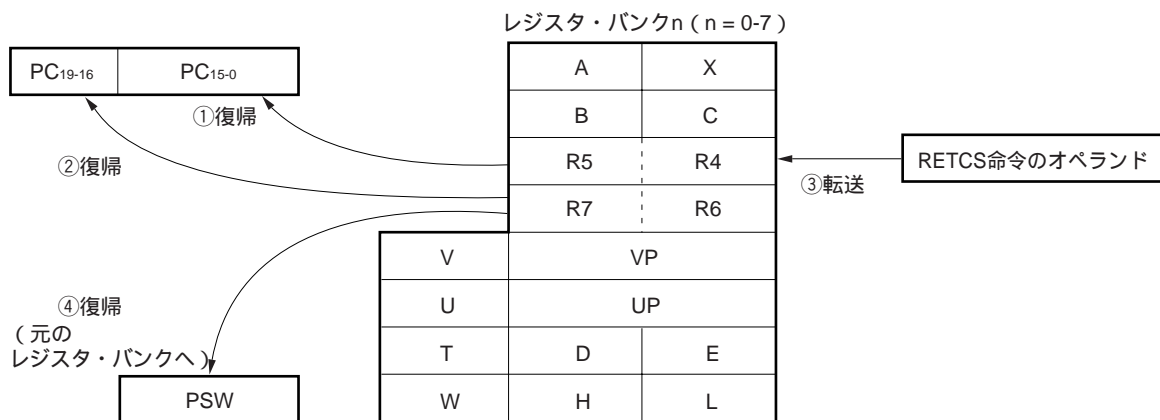
図14 - 12 割り込み要求の発生によるコンテキスト・スイッチング動作



コンテキスト・スイッチング機能を使用した割り込みからの復帰には、RETCS命令を使用します。RETCS命令では、次にその割り込みを受け付けたときに実行する割り込みサービス・プログラムの先頭アドレスを指定する必要があります。この割り込みサービス・プログラムの先頭アドレスは、ベース領域内になければいけません。

**注意** コンテキスト・スイッチングによる割り込みからの復帰には、必ずRETCS命令を使用してください。他の命令では、以降の割り込みに関する動作が正常に行われません。

図14 - 13 RETCS命令によるコンテキスト・スイッチング機能を使用した割り込みからの復帰動作



### 14.7.3 マスカブル割り込みの優先順位

μPD784975は、割り込み処理中にさらに別の割り込みを受け付ける多重割り込みの処理を行います。多重割り込みは、優先順位によって制御できます。

優先順位制御には、デフォルト優先順位による制御と、優先順位指定フラグの設定によるプログラマブル優先順位制御があります。デフォルト優先順位による優先順位制御は、複数の割り込みが同時に発生している場合、各割り込み要求にあらかじめ割り付けてある優先順位（デフォルト優先順位）に従って割り込み処理を行います（表14 - 2参照）。プログラマブル優先順位制御は、各割り込み要求を優先順位指定フラグの設定によって、4レベルに分けます。多重割り込み可能な割り込み要求を表14 - 5に示します。

なお、割り込みを受け付けるとIEフラグが自動的にクリア（0）されますので、多重割り込みを使用する場合は、割り込み処理プログラム中でEI命令を実行するなどして、IEフラグをセット（1）し、割り込み許可状態にしてください。

表14 - 5 多重割り込み処理

現在受け付け中の割り込みの優先順位	ISPRの値	PSWのIEフラグ	IMCのPRSLフラグ	受け付け可能なマスカブル割り込み
受け付け中の割り込みなし	00000000	0	x	・すべてのマクロ・サービスのみ
		1	x	・すべてのマスカブル割り込み
3	00001000	0	x	・すべてのマクロ・サービスのみ
		1	0	・すべてのマスカブル割り込み
		1	1	・すべてのマクロ・サービス ・優先順位を0, 1, 2に指定したマスカブル割り込み
2	0000 x 100	0	x	・すべてのマクロ・サービスのみ
		1	x	・すべてのマクロ・サービス ・優先順位を0, 1に指定したマスカブル割り込み
1	0000 x x 10	0	x	・すべてのマクロ・サービスのみ
		1	x	・すべてのマクロ・サービス ・優先順位を0に指定したマスカブル割り込み
0	0000 x x x 1	x	x	・すべてのマクロ・サービスのみ
ノンマスカブル割り込み	0100 x x x x	x	x	・すべてのマクロ・サービスのみ

図14 - 14 割り込み処理中に他の割り込み要求が発生した場合の処理例 (1/3)

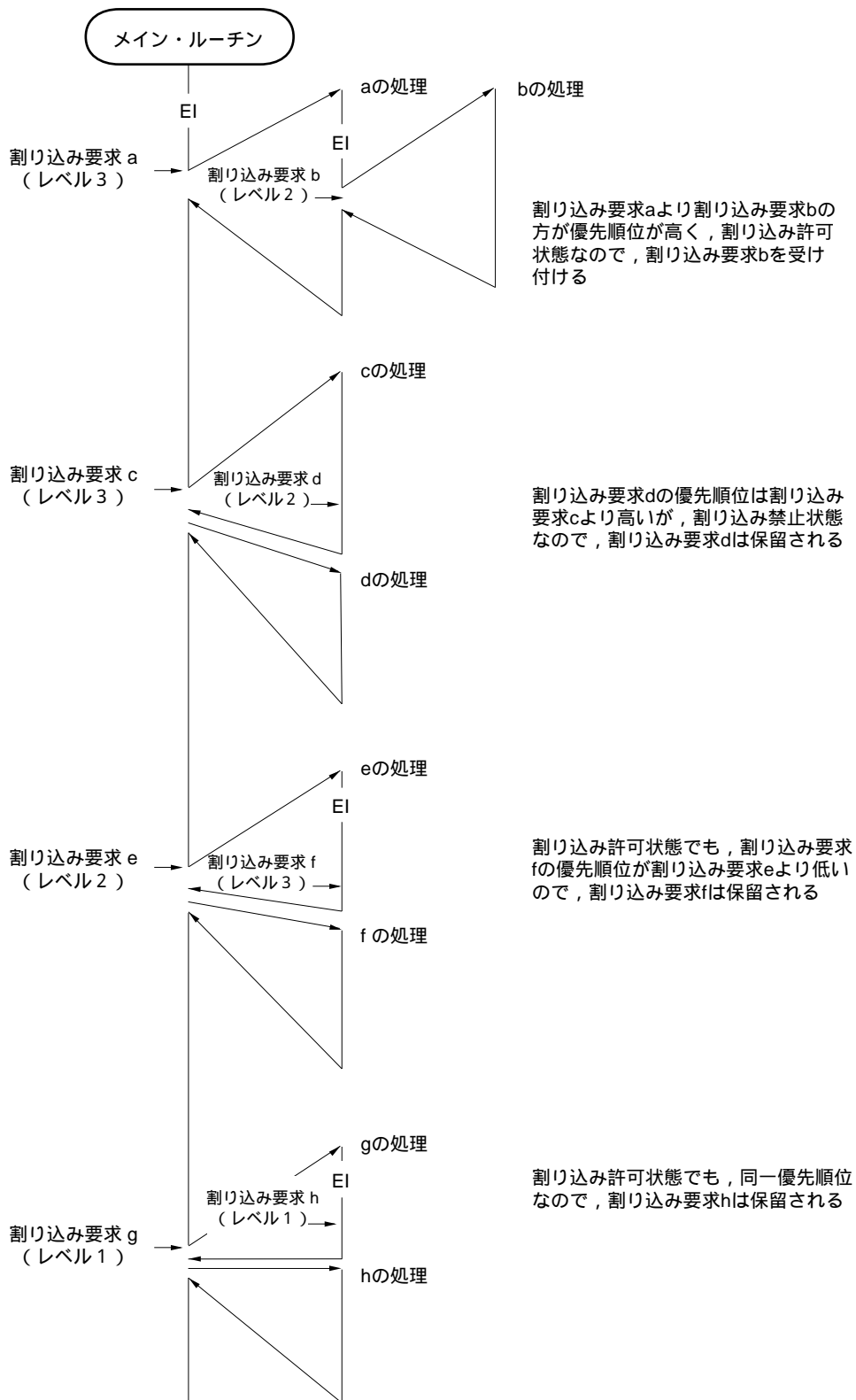




図14 - 14 割り込み処理中に他の割り込み要求が発生した場合の処理例 (2/3)

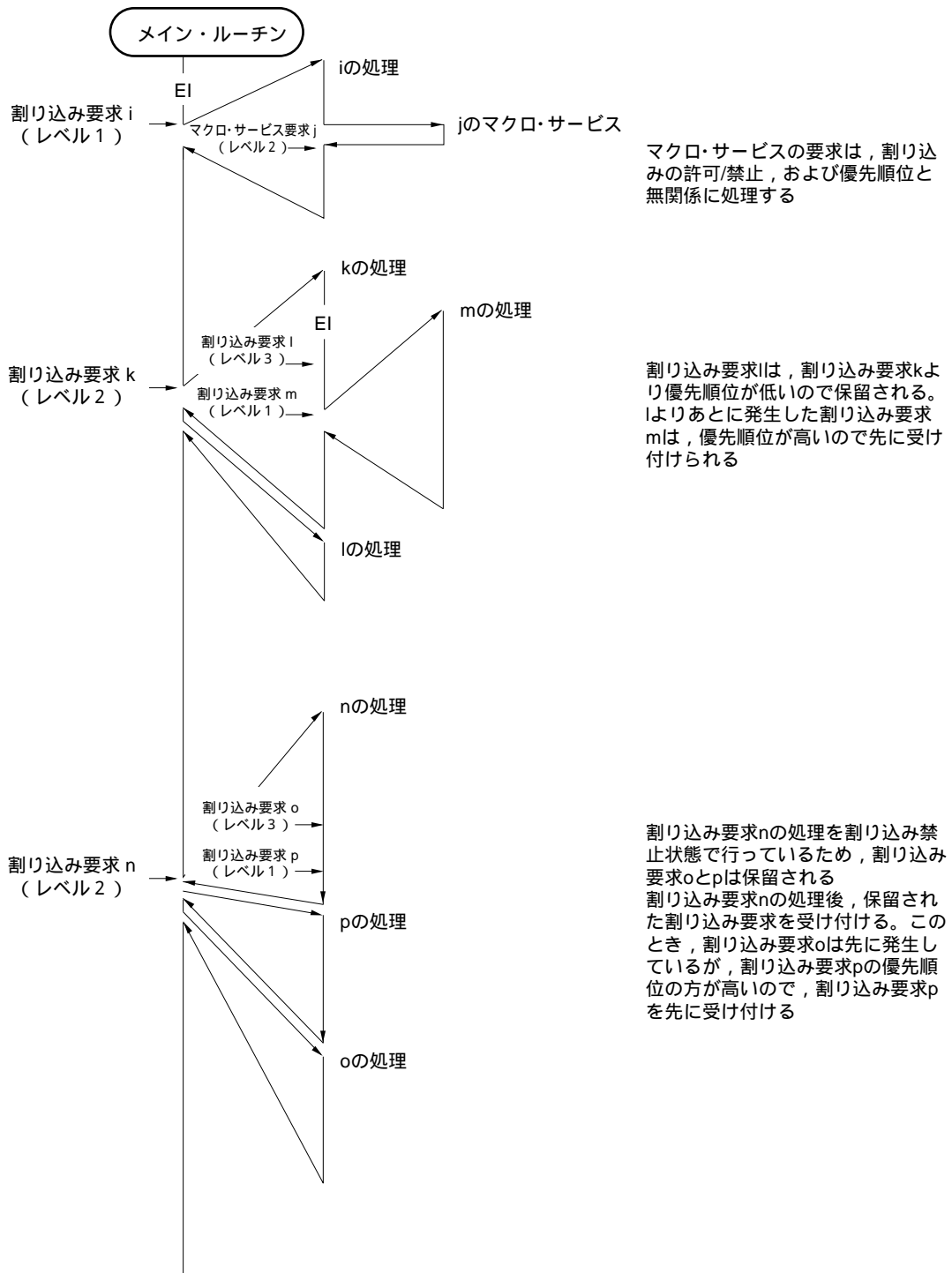
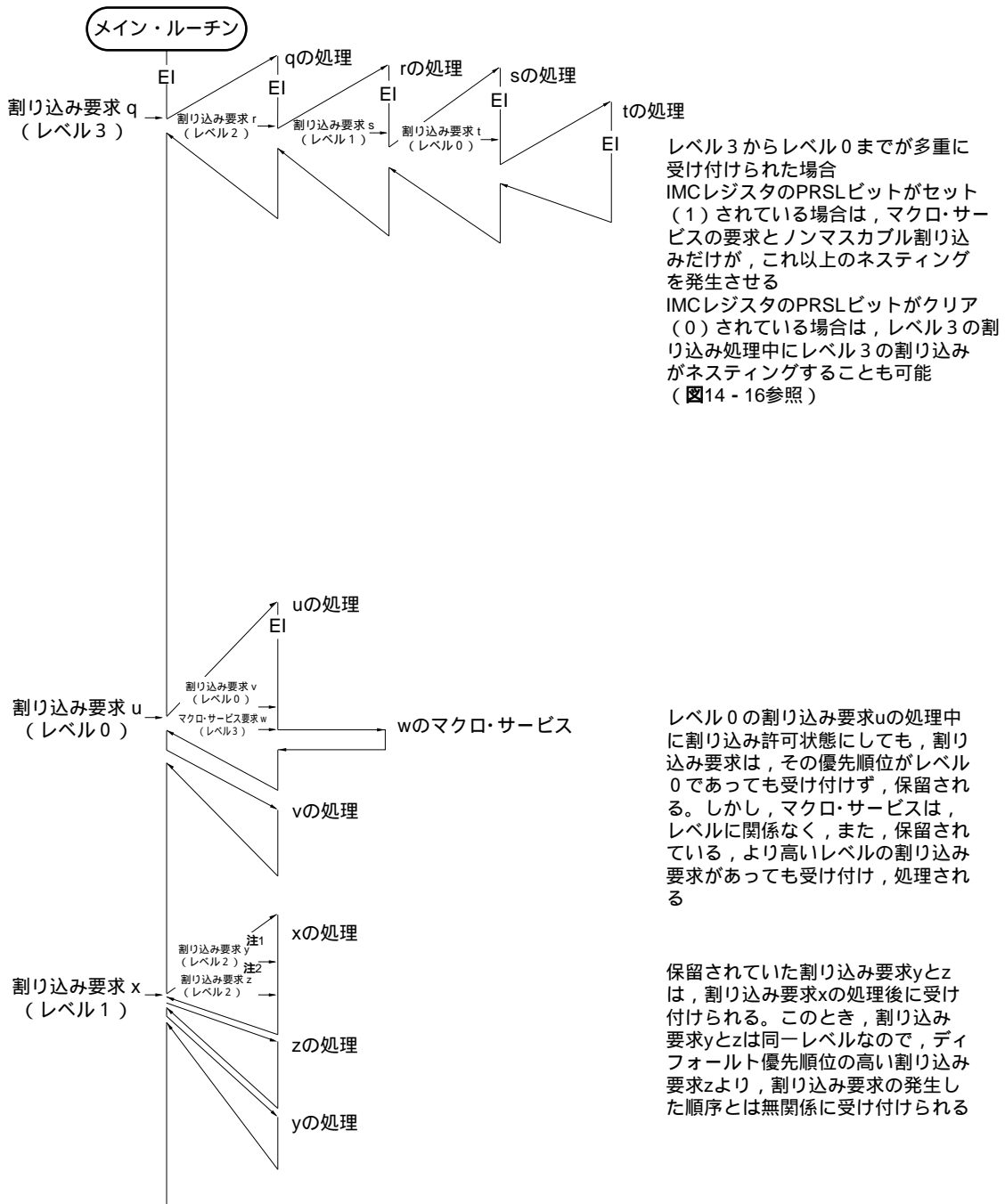


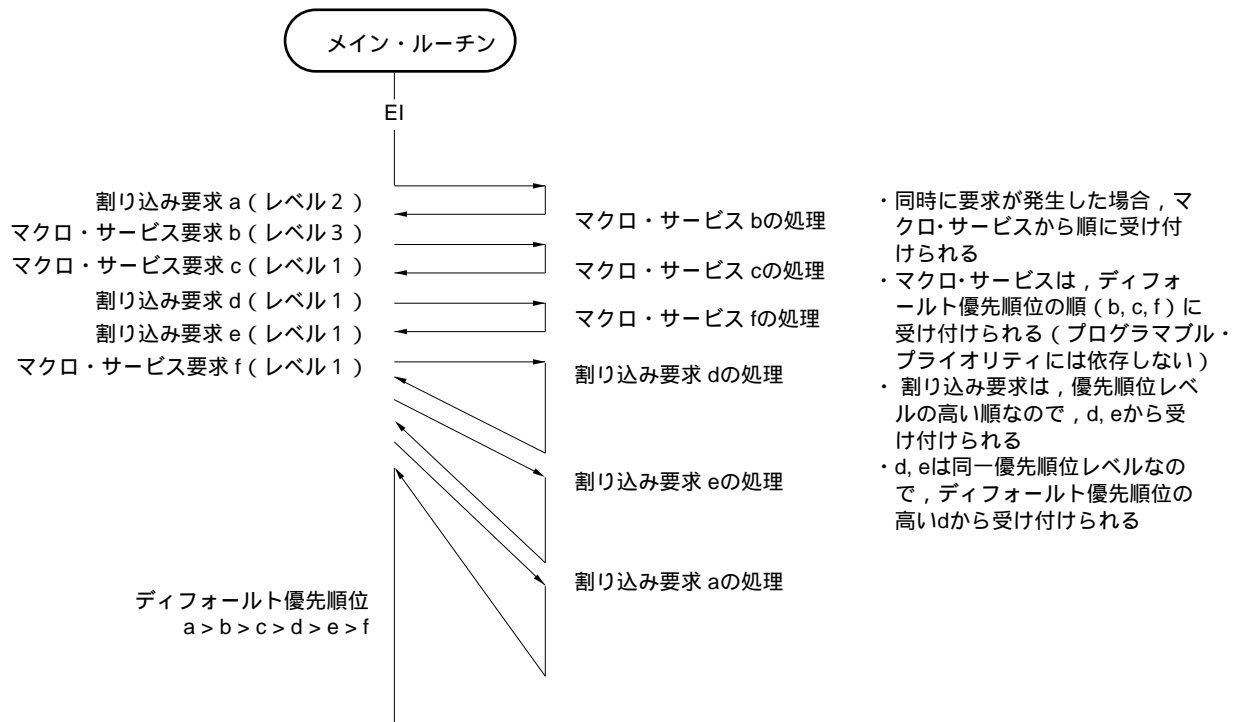
図14 - 14 割り込み処理中に他の割り込み要求が発生した場合の処理例 (3/3)



- 注1 . デフォルト優先順位が低い
- 2 . デフォルト優先順位が高い

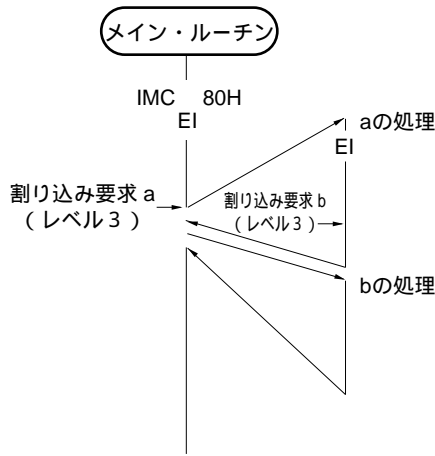
- 備考1 . 図中のa-zは、割り込み要求およびマクロ・サービス要求を区別するためにつけた仮の名称です。
- 2 . 図中のデフォルト優先順位の高い/低いは、2つの割り込み要求間の相対的な優先順位の高さを示します。

図14 - 15 同時発生した割り込み要求の処理例



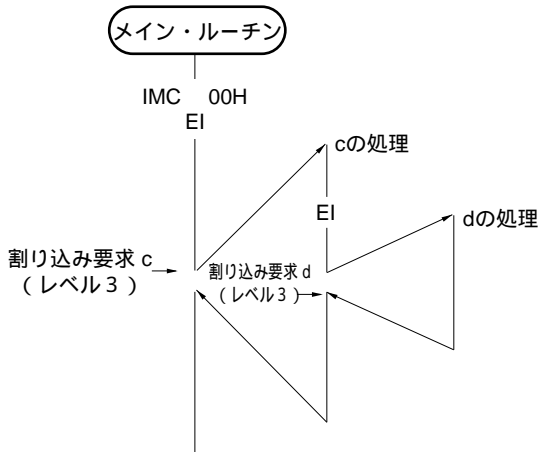
備考 図中のa-fは、割り込み要求およびマクロ・サービス要求を区別するためにつけた仮の名称です。

図14 - 16 IMCの設定によるレベル3の割り込みの受け付け動作の違い



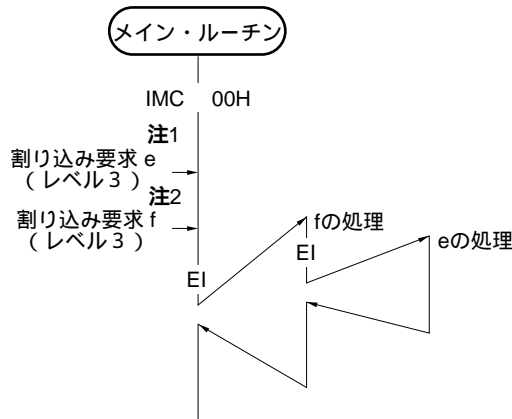
IMCのPRSL = 1として、レベル3の割り込み間でのネスティングを禁止

割り込み許可状態であっても同一優先順位なので、割り込み要求bは保留される



IMCのPRSL = 0として、レベル3の割り込み処理中でも、レベル3の割り込みを受け付けられる（ネスティングできる）ように設定

レベル3の割り込み要求cが割り込み許可状態で処理されていて、PRSL = 0なので、同一のレベル3の割り込み要求dを受け付ける



割り込み要求eとfは同一レベルなので、デフォルト優先順位の高い割り込み要求fが先に受け付けられる  
割り込み要求fの処理中に割り込み許可状態にすると、保留されていた割り込み要求eがPRSL = 0なので受け付けられる

- 注1 . デフォルト優先順位が低い
- 2 . デフォルト優先順位が高い

- 備考1 . 図中のa-fは、割り込み要求を区別するためにつけた仮の名称です。
- 2 . 図中のデフォルト優先順位の高い / 低い は、2つの割り込み要求間の相対的な優先順位の高さを示します。

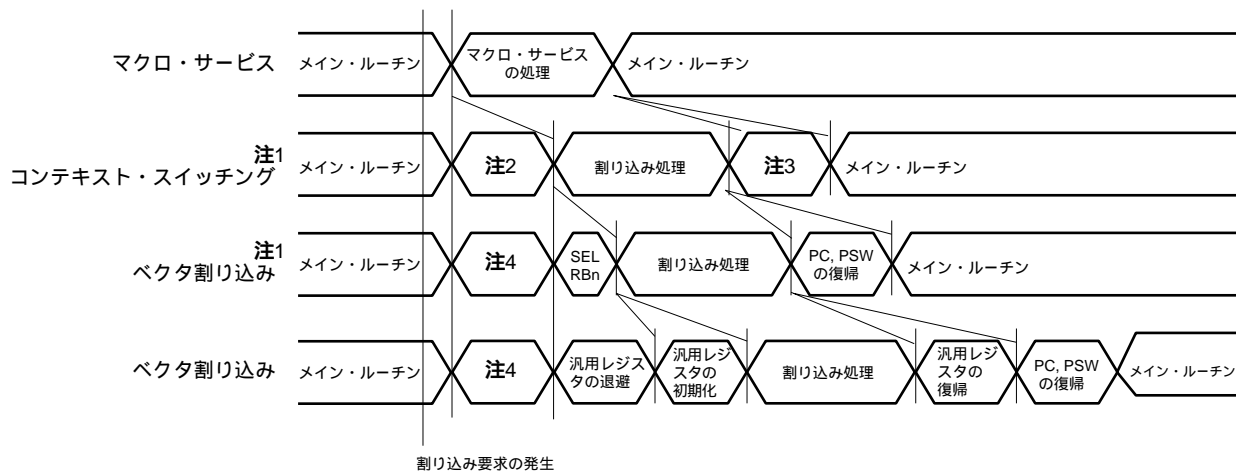
## 14.8 マクロ・サービス機能

### 14.8.1 マクロ・サービスの概要

マクロ・サービスは、割り込みの処理方法の一つです。通常の割り込みではプログラム・カウンタ (PC) やプログラム・ステータス・ワード (PSW) を退避し、PCへ割り込みサービス・プログラムの先頭アドレスをロードしますが、これらの処理の代わりに別の処理 (主にデータ転送) を行います。したがって、割り込みの要求に対して高速に応答することができます。さらに、プログラムで行うより高速に転送処理が可能なので、処理時間も短縮することができます。

また、指定された回数の処理を行ったあと、ベクタ割り込みを発生しますので、ベクタ割り込みのプログラムが簡略化できるなどの効果もあります。

図14-17 ベクタ割り込みとマクロ・サービスの処理の違い



注1．レジスタ・バンク切り替えを使用した場合で、レジスタにはあらかじめ初期値が設定されているとき

- 2．コンテキスト・スイッチングによるレジスタ・バンクの切り替え，PC，PSWの退避
- 3．コンテキスト・スイッチングによるレジスタ・バンク，PC，PSWの復帰
- 4．PC，PSWをスタックへ退避，ベクタ・アドレスをPCへロード

### 14.8.2 マクロ・サービスの種類

マクロ・サービスは表14-6に示す12種類の割り込みで使用することができます。また、3種類の動作タイプがあり、用途により使い分けが可能です。

表14 - 6 マクロ・サービスが使用可能な割り込み

デフォルト 優先順位	割り込み要求発生ソース	発生ユニット	マクロ・サービス・コント ロール・ワード・アドレス
0	INTWDTM (ウォッチドッグ・タイマのオーバフロー (インターバル・タイマを選択時))	ウォッチドッグ・ タイマ	0FE06H
1	INTP0 (端子入力エッジ検出)	エッジ検出	0FE08H
2	INTP1 ( " )		0FE0AH
3	INTP2 ( " )		0FE0CH
4	INTTM00 (16ビット・タイマ/カウンタ (TM0) と キャプチャ/コンペア・レジスタ (CR00) の一致信号 発生)	16ビット・タイマ/ イベント・カウンタ (TM0)	0FE0EH
5	INTTM01 (16ビット・タイマ/カウンタ (TM0) と キャプチャ/コンペア・レジスタ (CR01) の一致信号 発生)		0FE10H
6	INTKS (FIPコントローラ/ドライバからのキー・ス キャン・タイミング)	FIPコントローラ/ ドライバ	0FE12H
7	INTCSI0 (CSI0の3線式転送終了)	シリアル・インタ フェース	0FE14H
8	INTCSI1 (CSI1の3線式転送終了)		0FE16H
9	INTTM50 (8ビット・タイマ/カウンタ (TM50) と8 ビット・コンペア・レジスタ (CR50) の一致)	8ビットPWMタイ マ (TM50)	0FE18H
10	INTTM51 (8ビット・タイマ/カウンタ (TM51) と8 ビット・コンペア・レジスタ (CR51) の一致)	8ビットPWMタイ マ (TM51)	0FE1AH
11	INTAD (A/Dコンバータの変換終了)	A/Dコンバータ	0FE1CH

備考 1 . デフォルト優先順位は、固定されている値です。マクロ・サービス要求が同時に発生した場合に優先される順位を示します。

2 . CSI : クロック同期式シリアル・インタフェース

マクロ・サービスの種類は次の4タイプです。

#### (1) タイプA

割り込み要求の発生ごとに1バイトまたは1ワードのデータを特殊機能レジスタ(SFR)とメモリとの間で行い、指定された回数のデータ転送を行うとベクタ割り込みの要求を発生します。

転送の対象となるメモリは、LOCATION 0 命令実行時は0FE06H-0FE1DH、LOCATION 0FH命令実行時は0FFE06H-0FFE1DHの内部RAMに限定されます。

指定の方法が簡単で小容量の高速のデータ転送に向いています。

#### (2) タイプB

タイプAと同様に、割り込み要求の発生ごとに、1バイトまたは1ワードのデータを特殊機能レジスタ(SFR)とメモリとの間で行い、指定された回数のデータ転送を行うとベクタ割り込みの要求を発生します。

転送の対象となるSFRとメモリはマクロ・サービス・チャンネルで指定します(メモリは、1 Mバイトの全空間)。

タイプAの汎用タイプで、転送データ量が多い場合に向いています。

#### (3) タイプC

割り込み要求の発生ごとに、2つの特殊機能レジスタ(SFR)へメモリからデータ転送します。指定された回数のデータ転送を行うと、ベクタ割り込みを発生します。

タイプCのマクロ・サービスは一度の割り込み要求で2箇所へのデータ転送を行うことのほかに、出力データのリング制御やコンペア・レジスタとデータを自動的に加算する機能などを付加して使用することもできます。メモリについては、1 Mバイトの全空間を使用できます。

#### (4) カウンタ・モード

割り込み発生によって、マクロ・サービス・カウンタ(MSC)をデクリメントしていくモードで、割り込みの分周動作や、割り込み発生回路のカウント用として使用することができます。

MSCが0になるとベクタ割り込みを発生させることができます。

マクロ・サービスを再起動するときは、MSCの再設定が必要です。

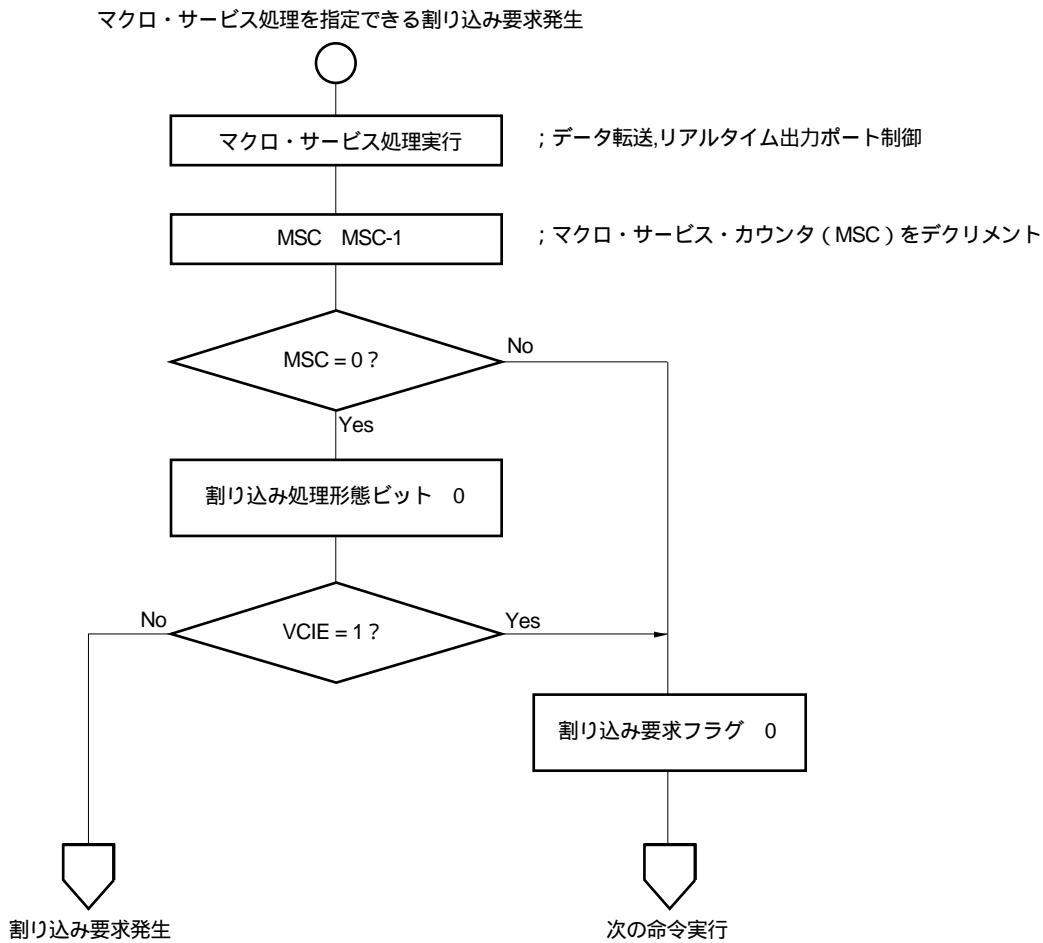
MSCは16ビットに固定で、8ビット・カウンタにすることはできません。

### 14.8.3 マクロ・サービスの基本動作

図14 - 10で示すアルゴリズムで発生したマクロ・サービス処理を指定できる割り込み要求は、基本的には図14 - 18に示すシーケンスで処理されます。

マクロ・サービス処理を指定できる割り込み要求は、IEフラグの状態には影響されません。割り込みマスク・レジスタ (MK0) の割り込みマスク・フラグのセット (1) によってのみ禁止されます。割り込み禁止状態、また割り込み処理プログラム実行中でもマクロ・サービス処理は実行できます。

図14 - 18 マクロ・サービス処理シーケンス



マクロ・サービスは、マクロ・サービス・コントロール・ワードのモード・レジスタに設定された値により、マクロ・サービスのタイプや転送方向などを決定します。その後、チャンネル・ポインタで指定されるマクロ・サービス・チャンネルをマクロ・サービスのタイプに従って使用して転送処理を行います。

マクロ・サービス・チャンネルは、転送回数を記憶するマクロ・サービス・カウンタや転送先や転送元のポインタやデータ・バッファが配置されているメモリで、LOCATION 0 命令実行時にはFE06H-FE1DH, LOCATION 0FH 命令実行時にはFFE06H-FFE1DHの中の任意のアドレスに配置することが可能です。



#### 14.8.4 マクロ・サービス終了時の動作

マクロ・サービスは、他のプログラムの実行中に指定された回数の処理を行います。指定された回数の処理を行うと（マクロ・サービス・カウンタ（MSC）が0になると）、マクロ・サービスは終了します。このときの動作には2通りあり、マクロ・サービスごとのマクロ・サービス・モード・レジスタのビット7にあるVCIEビットでその動作を指定します。

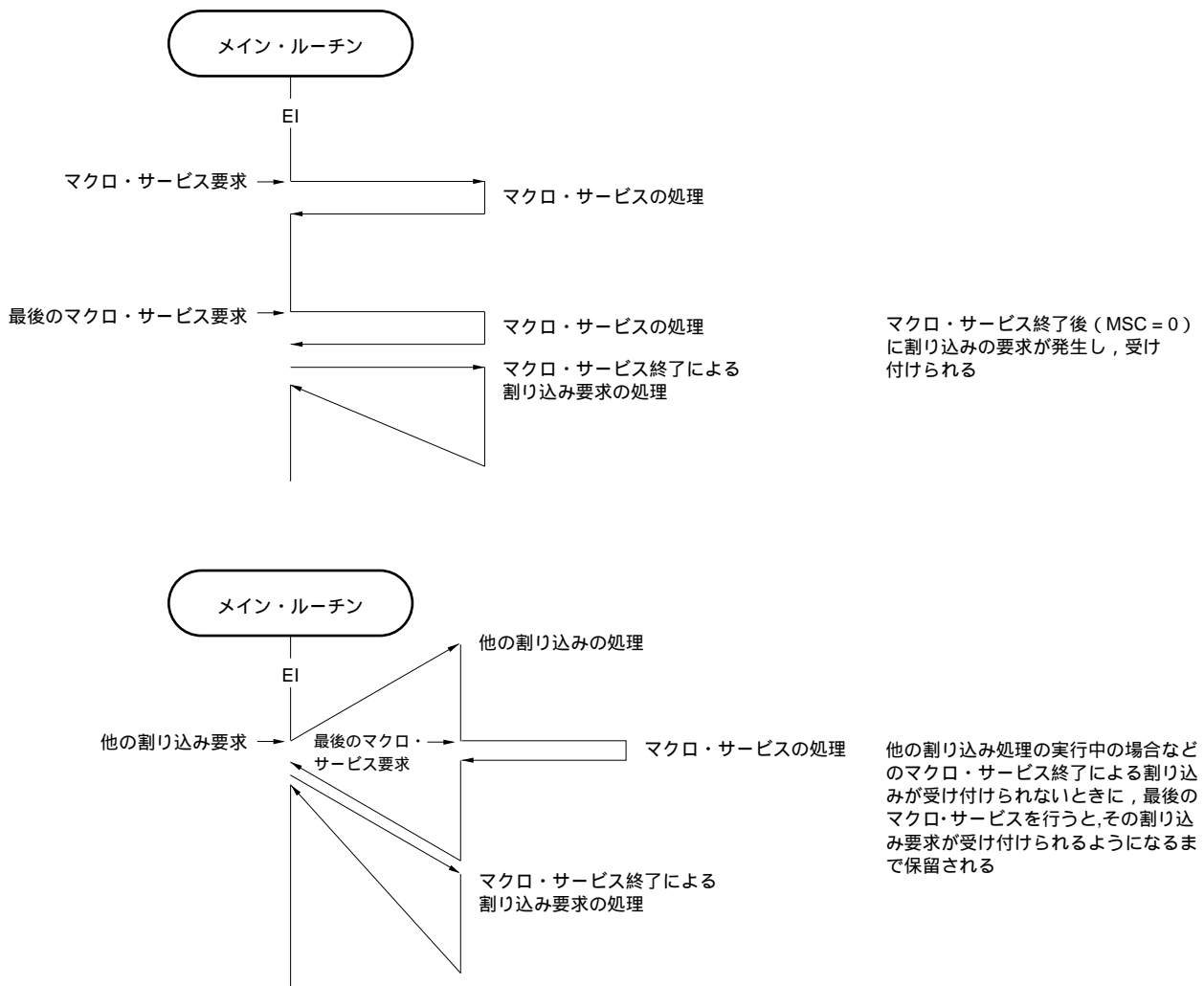
##### （1）VCIEビットが0の場合

マクロ・サービスの終了後、ただちに割り込みの要求を発生するモードです。図14 - 19に、VCIEビットが0の場合のマクロ・サービスおよび割り込みの受け付けの動作例を示します。

このモードは、最後に行ったマクロ・サービスの処理によって、一連の動作が終了する場合などに使用します。主な使用例を次に示します。

- ・ A/D変換結果の取り込み（INTAD）
- ・ タイマ/カウンタとコンペア・レジスタの一致によるコンペア・レジスタの更新時（INTTM00, INTTM01, INTTM50, INTTM51）
- ・ INTP<sub>n</sub>端子へのエッジ入力によるタイマ/カウンタのキャプチャ・レジスタの読み込み時（INTP0, INTP1, INTP2）

図14 - 19 VCIE = 0 ときのマクロ・サービス終了時の動作



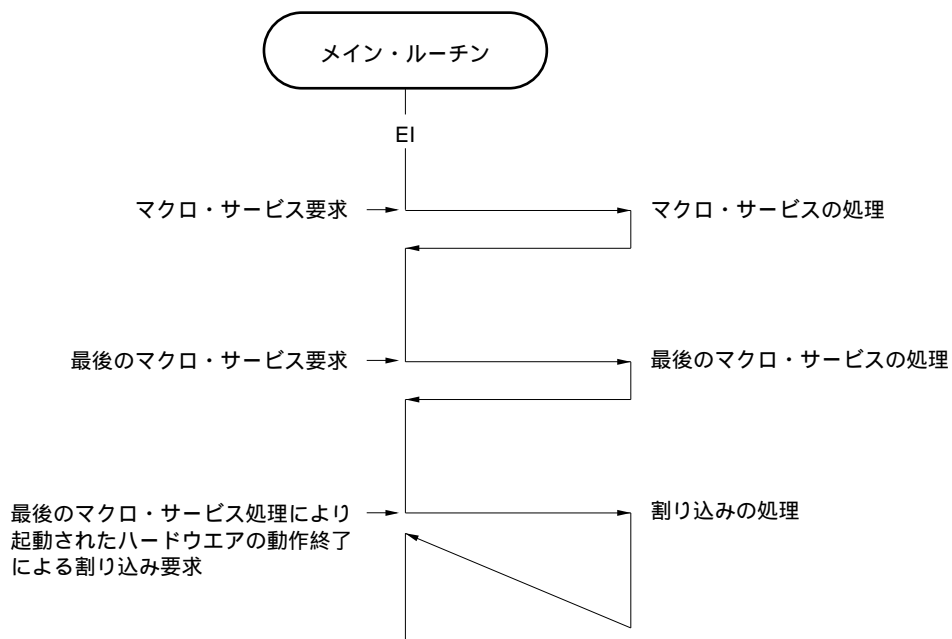
(2) VCIEビットが1の場合

マクロ・サービスの終了後、割り込みの要求を発生しないモードです。図14 - 20に、VCIEビットが1の場合のマクロ・サービスおよび割り込みの受け付けの動作例を示します。

このモードは、最後に行ったマクロ・サービスの処理によって、最後の動作を開始させる場合などに使用します。主な使用例を次に示します。

- ・クロック同期式シリアル・インタフェースによるデータ転送 (INTCSI0, INTCSI1)

図14 - 20 VCIE = 1 のときマクロ・サービス終了時の動作



### 14.8.5 マクロ・サービス制御レジスタ

#### (1) マクロ・サービス・コントロール・ワード

μPD784975のマクロ・サービス機能は、マクロ・サービス・モード・レジスタとマクロ・サービス・チャンネル・ポインタで制御されます。マクロ・サービス・モード・レジスタによってマクロ・サービス処理モードを設定し、マクロ・サービス・チャンネル・ポインタによってマクロ・サービス・チャンネルのアドレスを指し示します。

マクロ・サービス・モード・レジスタとマクロ・サービス・チャンネル・ポインタは、マクロ・サービス・コントロール・ワードとして、各マクロ・サービスごと、図14-21に示すように内部RAMの一部にマッピングされています。

マクロ・サービス処理を行うとき、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・レジスタとチャンネル・ポインタの値を設定しておく必要があります。

図14-21 マクロ・サービス・コントロール・ワードのフォーマット

予約語	アドレス		要因
ADCHP	0FE1DH	チャンネル・ポインタ	INTAD
ADMMD	0FE1CH	モード・レジスタ	
TMCHP51	0FE1BH	チャンネル・ポインタ	INTTM51
TMMMD51	0FE1AH	モード・レジスタ	
TMCHP50	0FE19H	チャンネル・ポインタ	INTTM50
TMMMD50	0FE18H	モード・レジスタ	
CSICHP1	0FE17H	チャンネル・ポインタ	INTCSI1
CSIMMD1	0FE16H	モード・レジスタ	
CSICHP0	0FE15H	チャンネル・ポインタ	INTCSI0
CSIMMD0	0FE14H	モード・レジスタ	
KSCHP	0FE13H	チャンネル・ポインタ	INTKS
KSMMD	0FE12H	モード・レジスタ	
TMCHP01	0FE11H	チャンネル・ポインタ	INTTM01
TMMMD01	0FE10H	モード・レジスタ	
TMCHP00	0FE0FH	チャンネル・ポインタ	INTTM00
TMMMD00	0FE0EH	モード・レジスタ	
PCHP2	0FE0DH	チャンネル・ポインタ	INTP2
PMMD2	0FE0CH	モード・レジスタ	
PCHP1	0FE0BH	チャンネル・ポインタ	INTP1
PMMD1	0FE0AH	モード・レジスタ	
PCHP0	0FE09H	チャンネル・ポインタ	INTP0
PMMD0	0FE08H	モード・レジスタ	
WDTCHP	0FE07H	チャンネル・ポインタ	INTWDTM
WDTMMD	0FE06H	モード・レジスタ	

(2) 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内部RAM上に書いておきます(図14-21参照)。

図14-22に、マクロ・サービス・モード・レジスタのフォーマットを示します。

図14-22 マクロ・サービス・モード・レジスタのフォーマット(1/2)

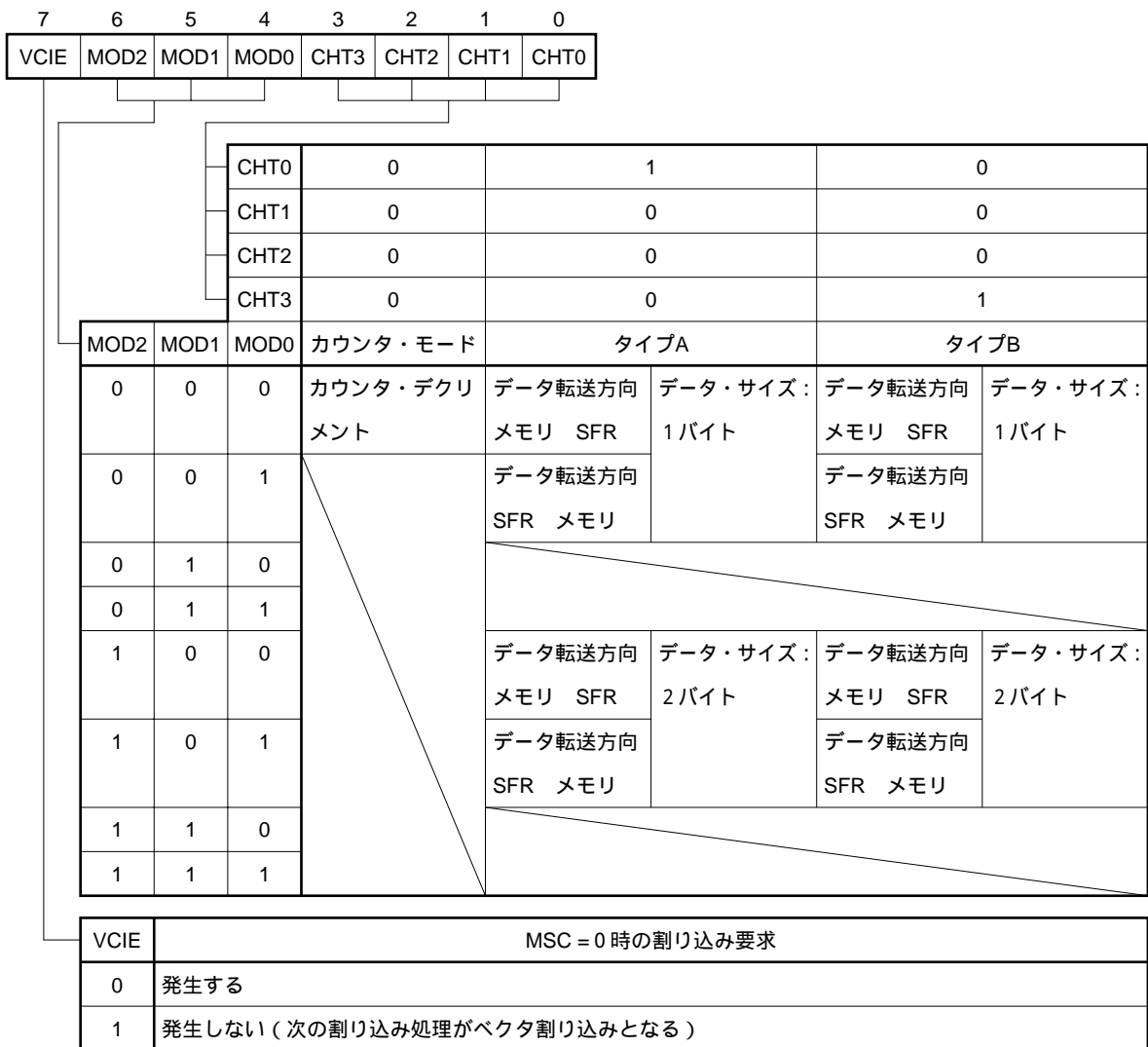


図14 - 22 マクロ・サービス・モード・レジスタのフォーマット (2/2)



(3) マクロ・サービス・チャンネル・ポインタ

マクロ・サービス・チャンネル・ポインタは、マクロ・サービス・チャンネルのアドレスを指定するポインタです。マクロ・サービス・チャンネルはLOCATION 0 命令実行時はFE06H-FE1DH, LOCATION 0FH 命令実行時はFFE06H-FFE1DHの256バイトの空間に配置可能で、アドレスの上位16ビットが固定です。したがって、マクロ・サービス・チャンネル・ポインタには、マクロ・サービス・チャンネルの最上位アドレスに格納されるデータの低位8ビットを設定します。

## 14.8.6 マクロ・サービス・タイプA

### (1) 動作

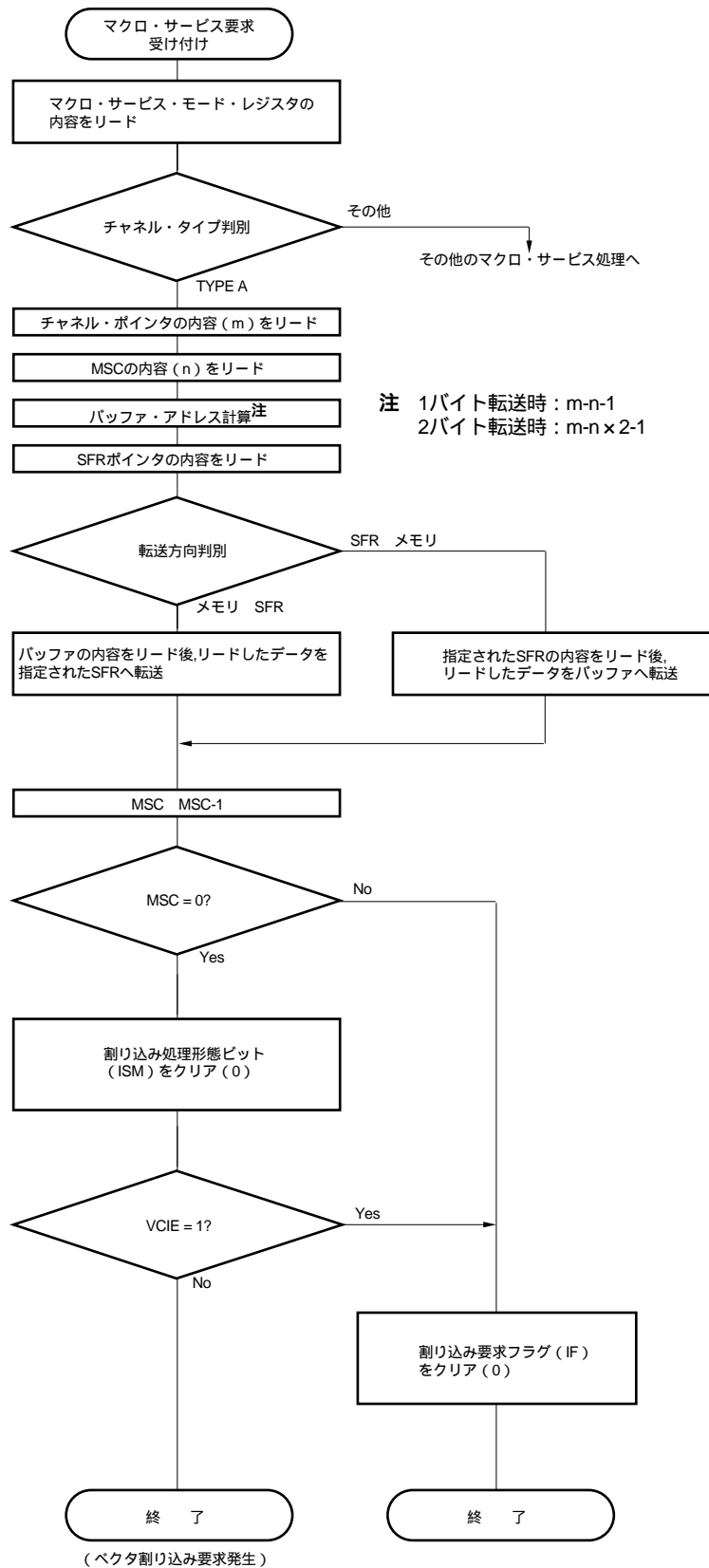
マクロ・サービス・チャンネル内のバッファ・メモリとマクロ・サービス・チャンネル内で指定されるSFRとの間でデータの転送を行います。

タイプAは、データの転送方向としてメモリからSFR、またはSFRからメモリを選択することができます。

あらかじめマクロ・サービス・カウンタに設定した回数だけ、データ転送を行います。なお、1回のマクロ・サービス処理で8ビットまたは16ビットのデータが転送されます。

転送するデータ量が少ないときに、高速でデータ転送ができるので有効です。

図14 - 23 マクロ・サービス・データ転送処理フロー（タイプA）





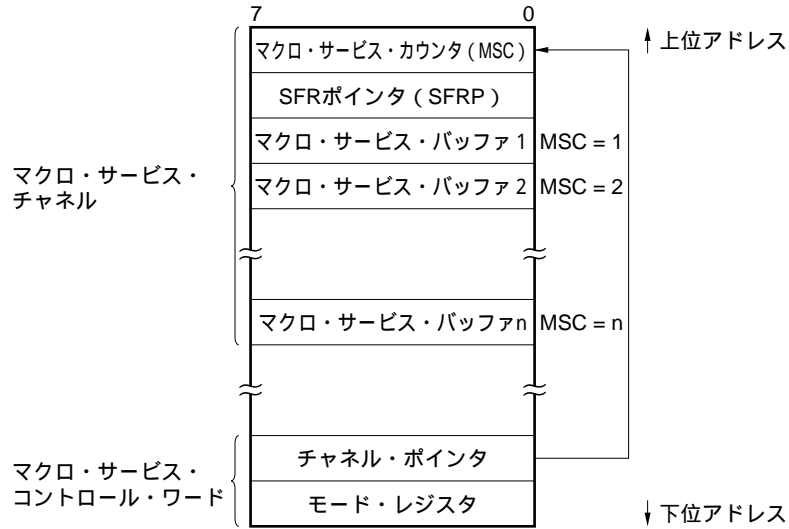
## (2) マクロ・サービス・チャンネルの構成

チャンネル・ポインタと8ビットのマクロ・サービス・カウンタ(MSC)とで、転送元もしくは転送先となる内部RAM(LOCATION 0命令実行時: FE06H-FE1DH, LOCATION 0FH命令実行時: FFE06H-FFE1DH)上のバッファ・アドレスを示します(図14-24参照)。チャンネル・ポインタには、マクロ・サービス・チャンネル内にあるマクロ・サービス・カウンタのアドレスの下位8ビットを書き込みます。

アクセスの対象となるSFRは、SFRポインタ(SFRP)で指定します。SFRPには、SFRアドレスの下位8ビットを書き込みます。

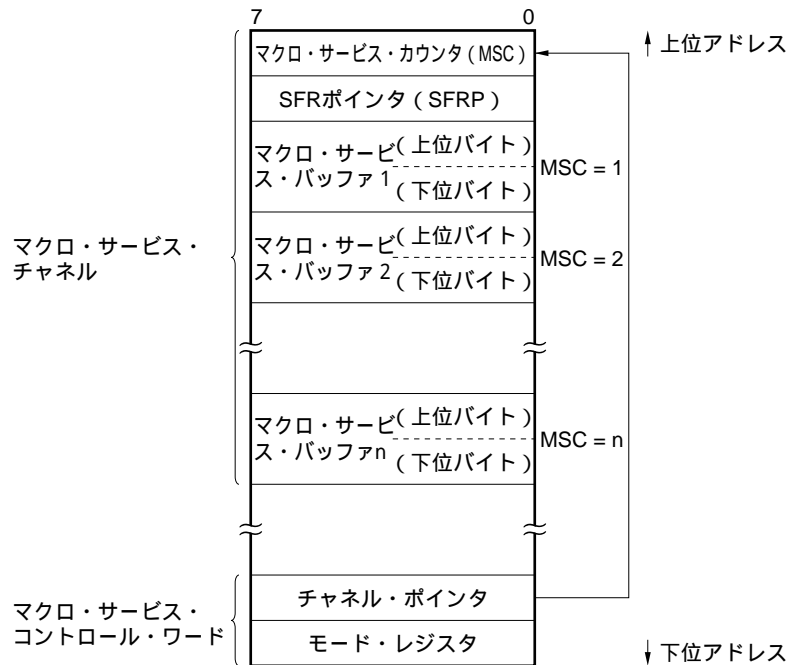
図14 - 24 タイプAのマクロ・サービス・チャンネル

( a ) 1バイト転送時



$$\text{マクロ・サービス・バッファ・アドレス} = (\text{チャンネル・ポインタ}) - (\text{マクロ・サービス・カウンタ}) - 1$$

( b ) 2バイト転送時



$$\text{マクロ・サービス・バッファ・アドレス} = (\text{チャンネル・ポインタ}) - (\text{マクロ・サービス・カウンタ}) \times 2 - 1$$

### 14.8.7 マクロ・サービス・タイプB

#### (1) 動作

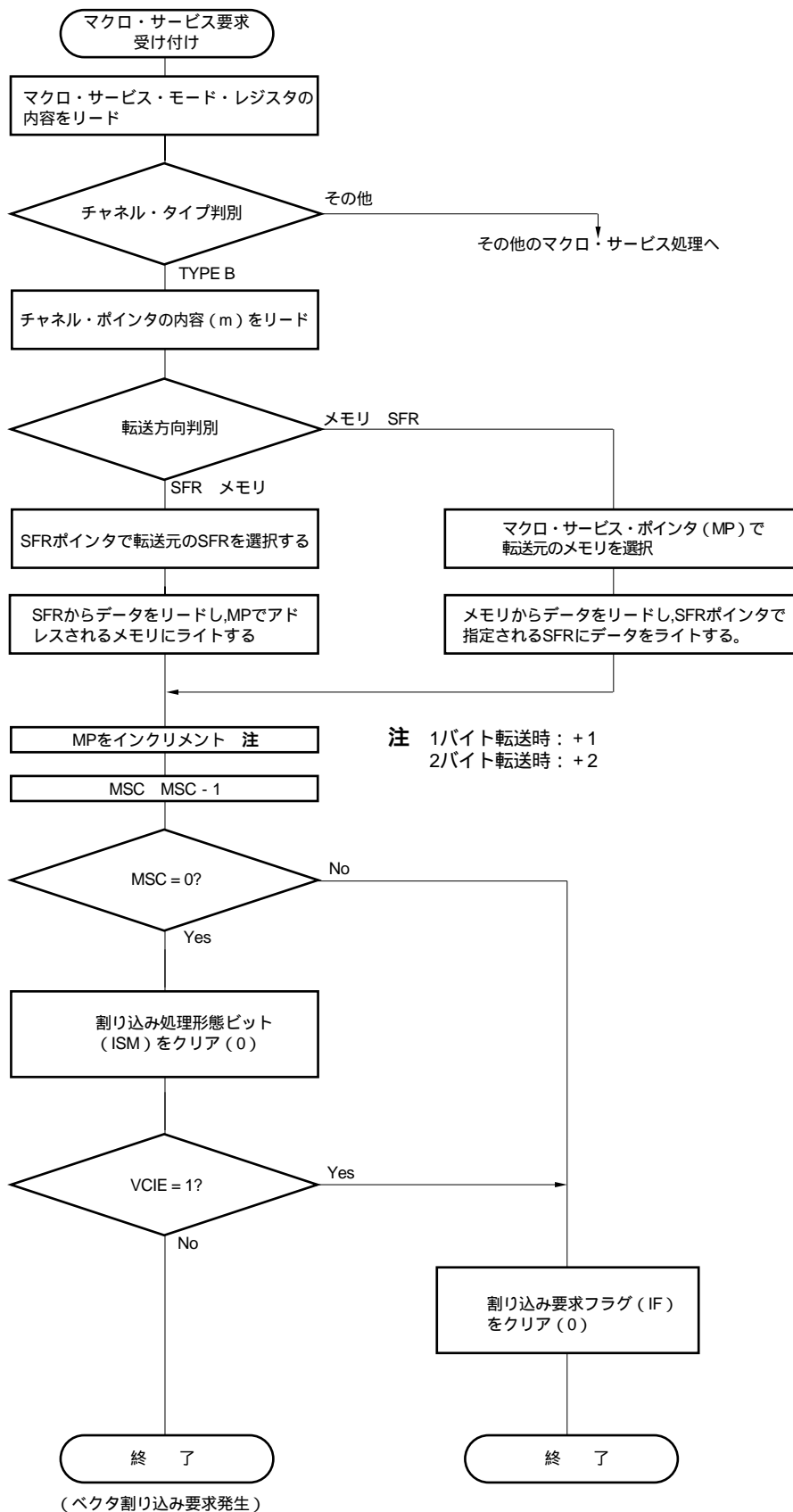
マクロ・サービス・チャンネルで指定されるメモリ内のデータ領域とSFRとの間でデータの転送を行います。

タイプBは、データの転送方向としてメモリからSFR、またはSFRからメモリを選択することができます。

あらかじめマクロ・サービス・カウンタに設定した回数だけ、データ転送を行います。なお、1回のマクロ・サービス処理で8ビットまたは16ビットのデータが転送されます。

タイプAのマクロ・サービスを汎用的にしたタイプで、データ・バッファ領域として8ビット・データ転送時は最大64 Kバイト、16ビット・データ転送時は最大128 Kバイトまでの領域を1 Mバイトの任意のアドレス空間に持たせることができるため、大容量データの処理に最適です。

図14 - 25 マクロ・サービス・データ転送処理フロー（タイプB）



(2) マクロ・サービス・チャンネルの構成

マクロ・サービス・ポインタ (MP) は、転送先または転送元となる 1 Mメモリ空間内のデータ・バッファ領域を示します。

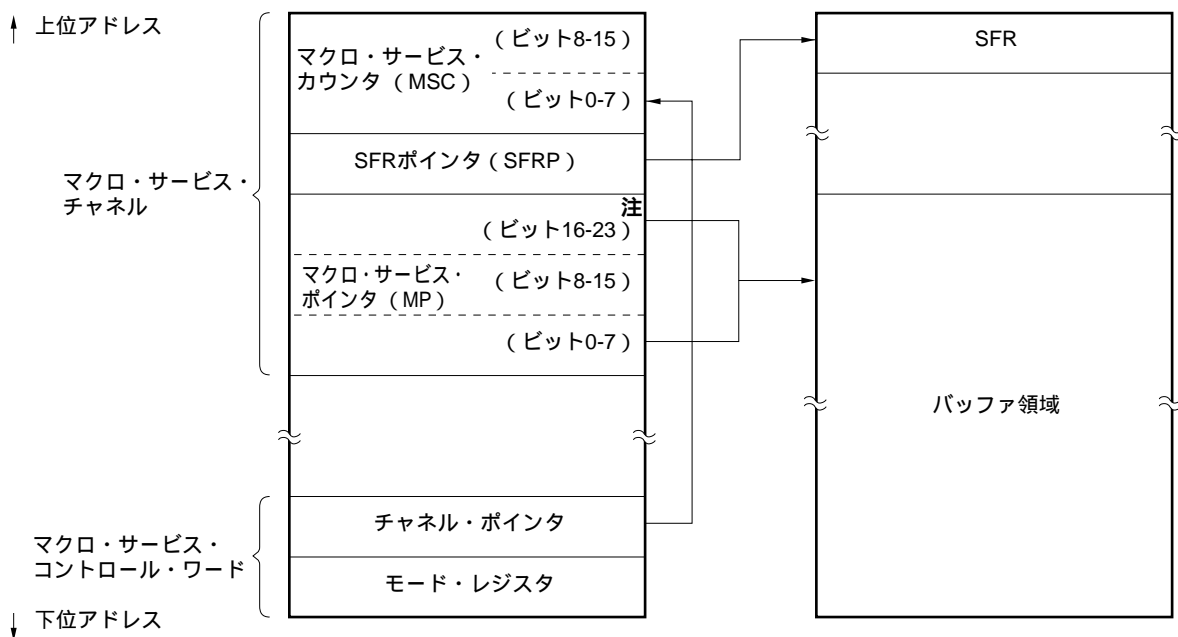
SFRポインタ (SFRP) には、転送先または転送元となるSFRのアドレスの下位 8 ビットを書き込みます。

マクロ・サービス・カウンタ (MSC) は、16ビットのカウンタでデータ転送の回数を指定します。

MP, SFRPおよびMSCを格納するマクロ・サービス・チャンネルは、LOCATION 0 命令実行時は、内部RAM空間の0FE06H-0FE1DH, LOCATION 0FH命令実行時は0FFE06H-0FFE1DHに配置します。

マクロ・サービス・チャンネルは、図14 - 26で示すようにチャンネル・ポインタで示します。チャンネル・ポインタには、マクロ・サービス・チャンネル内にあるマクロ・サービス・カウンタへアドレスの下位 8 ビットを書き込みます。

図14 - 26 タイプBのマクロ・サービス・チャンネル



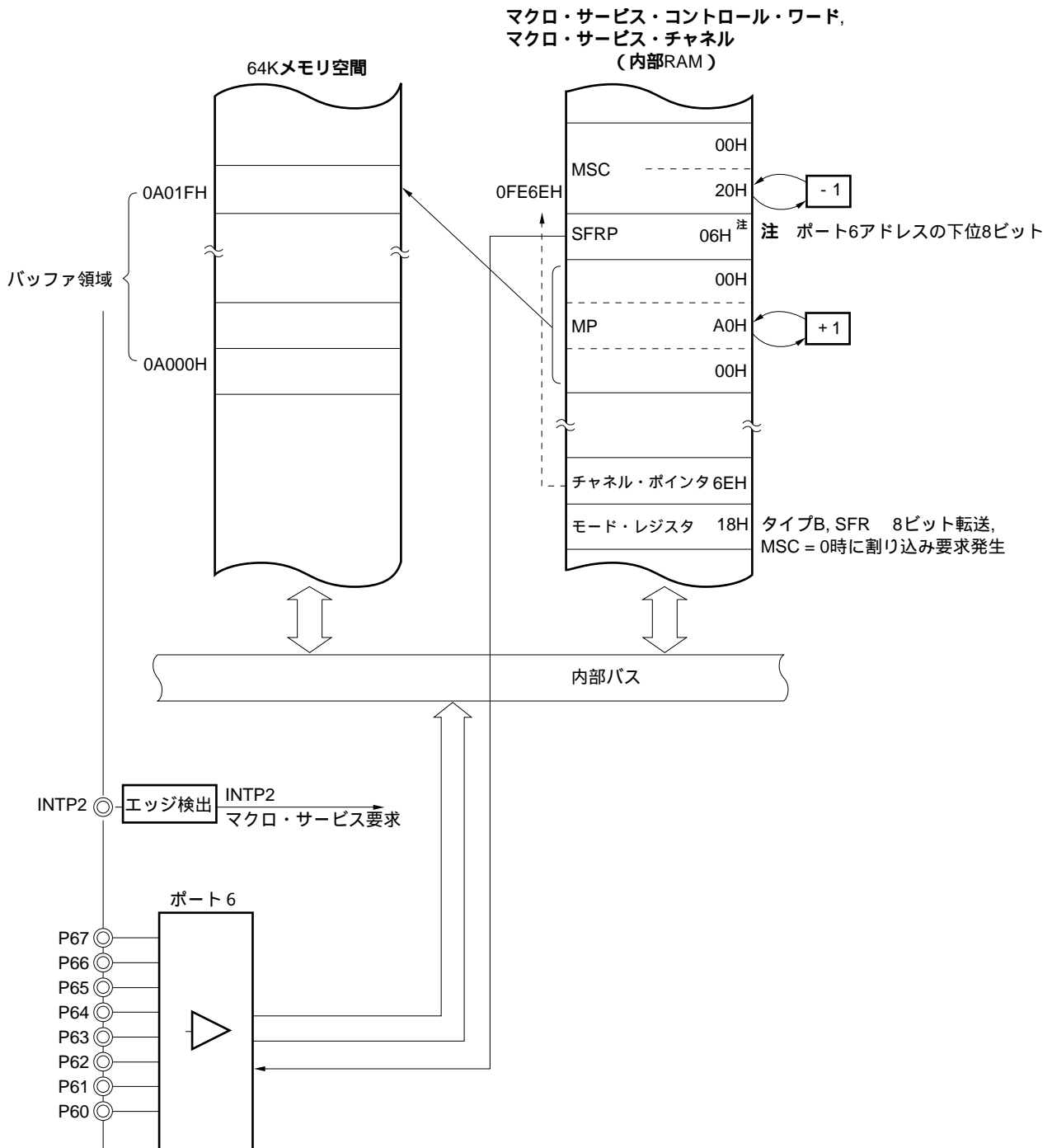
マクロ・サービス・バッファ・アドレス = マクロ・サービス・ポインタ

注 ビット20-23は必ず0にしてください。

(3) タイプB使用例

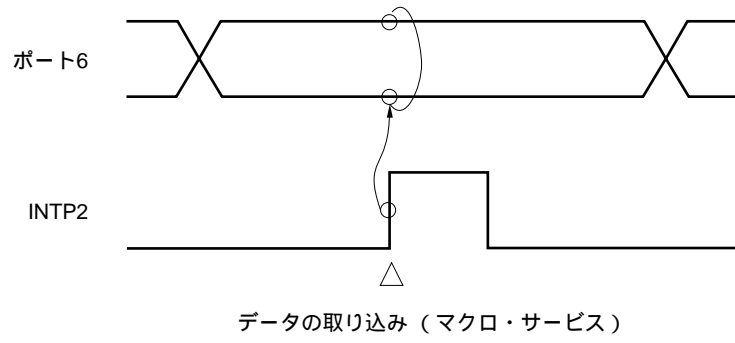
外部信号に同期してポート6から平行・データを入力する例を示します。外部信号との同期は外部割り込み端子 (INTP2) 入力を用います。

図14 - 27 外部割り込みに同期した平行・データ入力



**備考** 図中のマクロ・サービス・チャンネルのアドレスは、LOCATION 0 命令実行時の値です。  
LOCATION 0FH命令実行時には、図中の値に0F0000Hを加えてください。

図14 - 28 パラレル・データ入力のタイミング



## 14.8.8 マクロ・サービス・タイプC

### (1) 動作

タイプCのマクロ・サービスは、1回の割り込みの要求でマクロ・サービス・チャンネルで指定されるタイマ用とデータ用の2つのSFRへ、マクロ・サービス・チャンネルで指定されるメモリ内のデータを転送します（SFRは自由に選択できます）。タイマ用のSFRは、8ビットまたは16ビットの選択が可能です。

タイプCのマクロ・サービスは、上記の基本的なデータ転送に加えて、バッファ領域の圧縮とソフトウェアの負担を軽減するために以下の機能を付加することができます。

これらの指定は、マクロ・サービス・コントロール・ワードのモード・レジスタで指定します。

#### (a) タイマ用マクロ・サービス・ポイントの更新

タイマ用のマクロ・サービス・ポイント（MPT）を保持するか、インクリメント/デクリメントするかを選択できます。なお、インクリメント/デクリメントの方向は、データ用マクロ・サービス・ポイント（MPD）と同じ方向になります。

#### (b) データ用マクロ・サービス・ポイントの更新

データ用のマクロ・サービス・ポイント（MPD）をインクリメント/デクリメントするかを選択できます。

#### (c) 自動加算

タイマ用のマクロ・サービス・ポイント（MPT）でアドレスされるデータと現在のコンペア・レジスタの値を加算して、加算結果をコンペア・レジスタへ転送します。

自動加算を指定しない場合は、MPTでアドレスされるデータをコンペア・レジスタに転送するだけです。

#### (d) リング制御

あらかじめ指定された長さの出力データ・パターンを自動的に繰り返し出力します。



図14 - 29 マクロ・サービス・データ転送処理フロー（タイプC）（1/2）

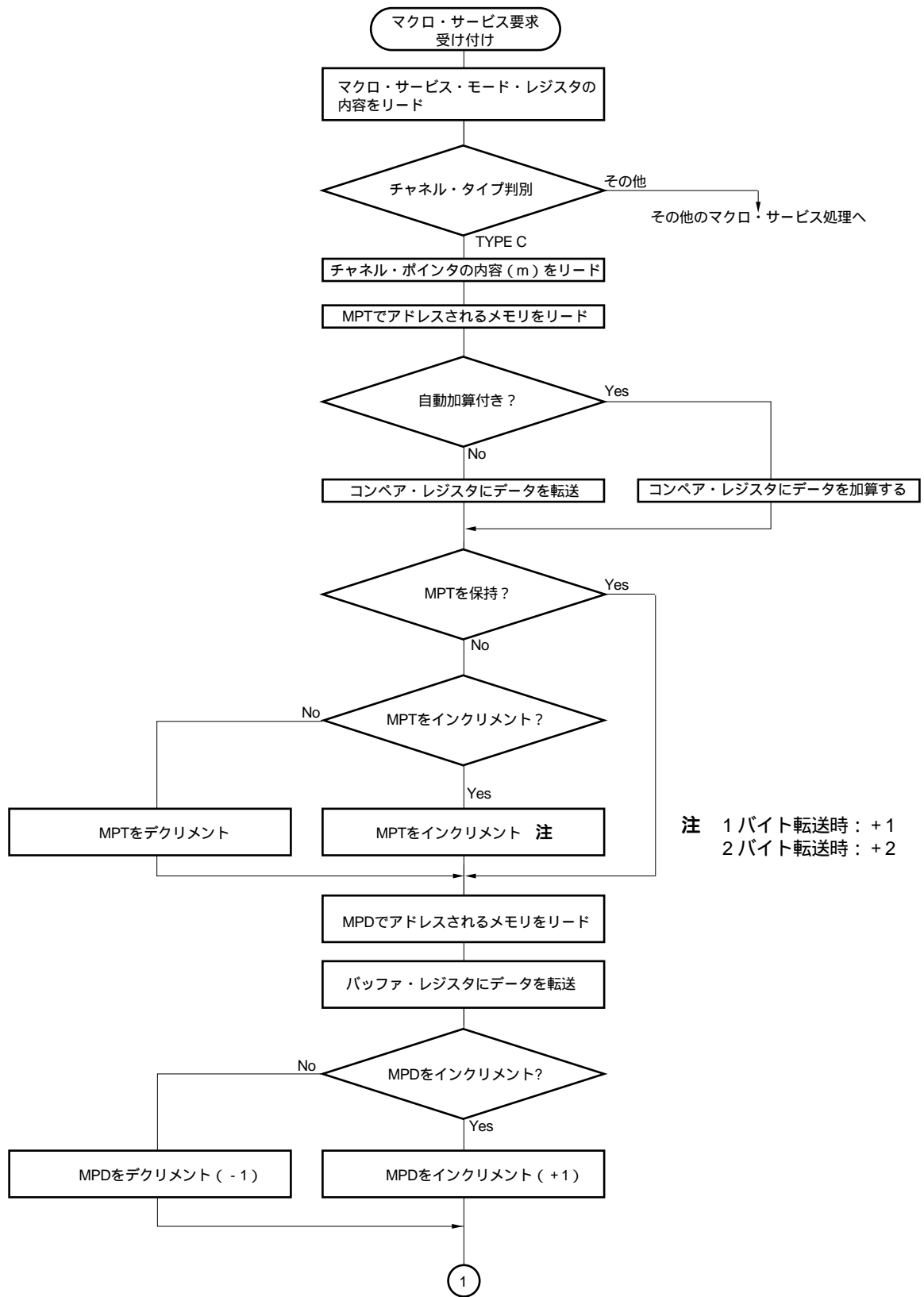
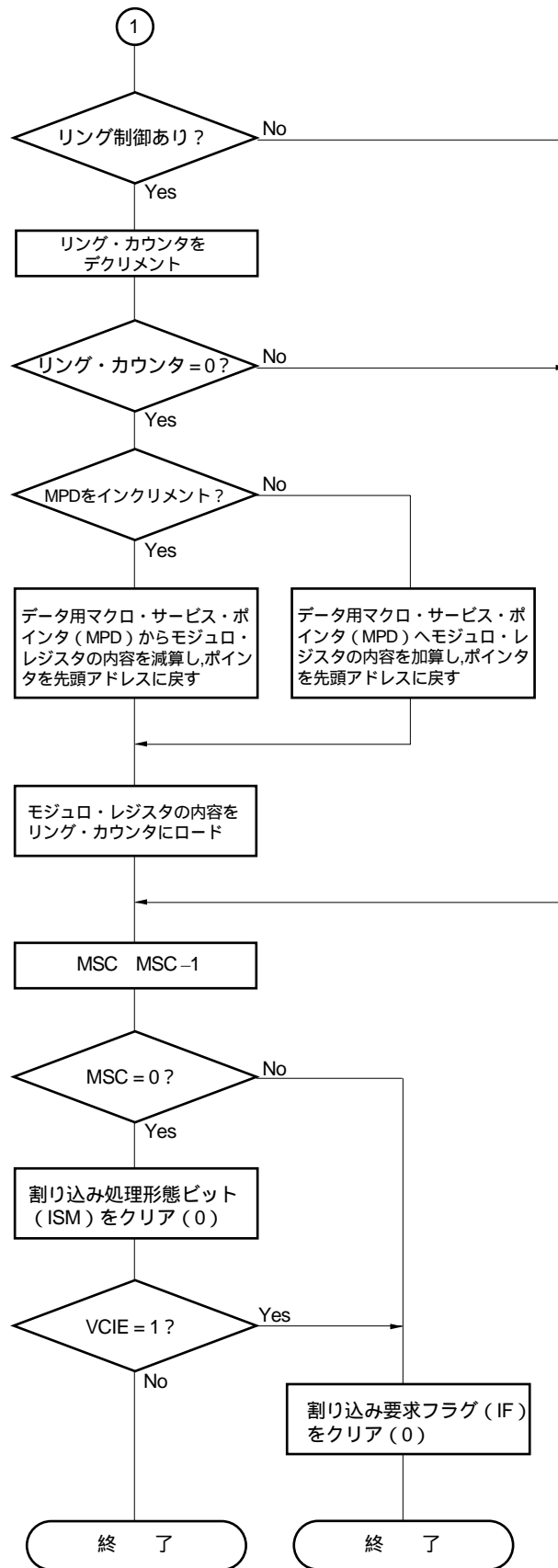


図14 - 29 マクロ・サービス・データ転送処理フロー（タイプC）（2/2）



(ベクタ割り込み要求発生)

## (2) マクロ・サービス・チャンネルの構成

タイプCのマクロ・サービス・チャンネルには図14 - 30に示す2種類があります。

タイマ用マクロ・サービス・ポインタ (MPT) はおもにタイマ/カウンタのコンペア・レジスタに転送または加算する1Mメモリ空間内のデータ・バッファ領域を示します。

モジュロ・レジスタ (MR) は、リング制御を使用するとき繰り返しのパターン数を指定します。

リング・カウンタ (RC) は、リング制御を使用するときのパターン内のステップを保持します。通常初期設定時は、MRと同じ値に設定します。

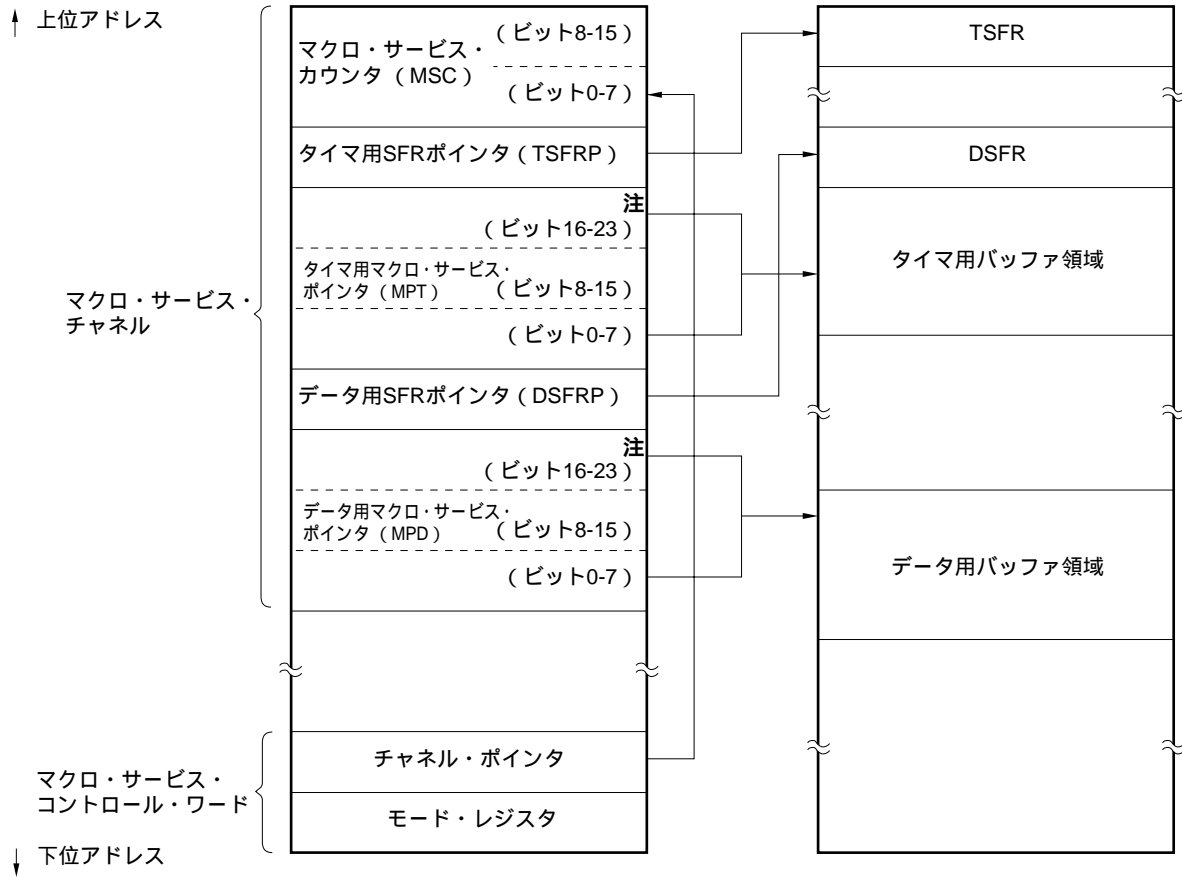
マクロ・サービス・カウンタ (MSC) は、16ビットのカウンタでデータの転送回数を指定します。

タイマ用SFRポインタ (TSFRP)、およびデータ用SFRポインタ (DSFRP) には、転送先となるSFRのアドレスの下位8ビットを書き込みます。

これらのポインタやカウンタを格納するマクロ・サービス・チャンネルは、LOCATION 0命令実行時は0FE06H-0FE1DHの内部RAM空間に、LOCATION 0FH命令実行時は0FFE06H-0FFE1DHの内部RAM空間に配置します。マクロ・サービス・チャンネルは、図14 - 30に示すようにチャンネル・ポインタで示します。チャンネル・ポインタには、マクロ・サービス・チャンネル内にあるマクロ・サービス・カウンタへアドレスの下位8ビットを書き込みます。

図14 - 30 タイプCのマクロ・サービス・チャンネル (1/2)

( a ) リング制御なし

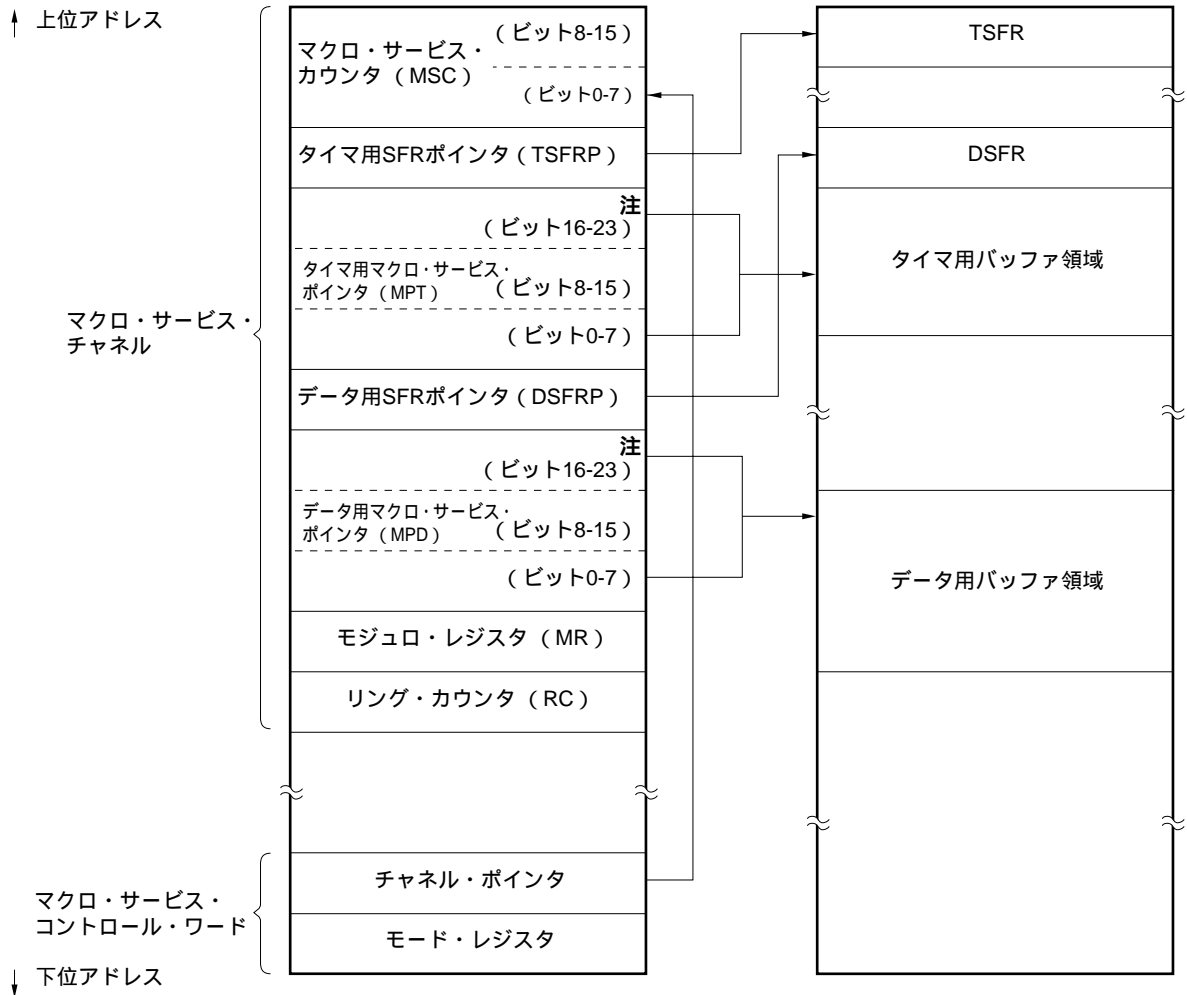


マクロ・サービス・バッファ・アドレス = マクロ・サービス・ポインタ

注 ビット20-23は必ず0にしてください。

図14 - 30 タイプCのマクロ・サービス・チャンネル (2/2)

(b) リング制御あり



マクロ・サービス・バッファ・アドレス = マクロ・サービス・ポインタ

注 ビット20-23は必ず0にしてください。

**(b) 自動加算制御およびリング制御の使用例****(i) 自動加算制御**

マクロ・サービス・ポインタ (MPT) で指定された出力タイミング・データ ( t) をコンペア・レジスタの内容に加算し、加算結果をコンペア・レジスタに書き戻します。

この自動加算制御を利用することにより、コンペア・レジスタの設定値をその都度プログラム中で計算させる必要がなくなります。

**(ii) リング制御**

リング制御は、あらかじめ決まる出力データ・パターンを1サイクル分だけ用意しておいて、1サイクル分の出力パターンをリング形式で繰り返し順に出力するものです。

リング制御を用いる場合、出力するデータ・パターンとして1サイクル分だけ準備すればよいので、データROM領域を小さくすることができます。

マクロ・サービス・カウンタ (MSC) のデクリメントは、1回のデータ転送ごとに行います。

リング制御の場合も、MSC = 0 で割り込み要求を発生します。

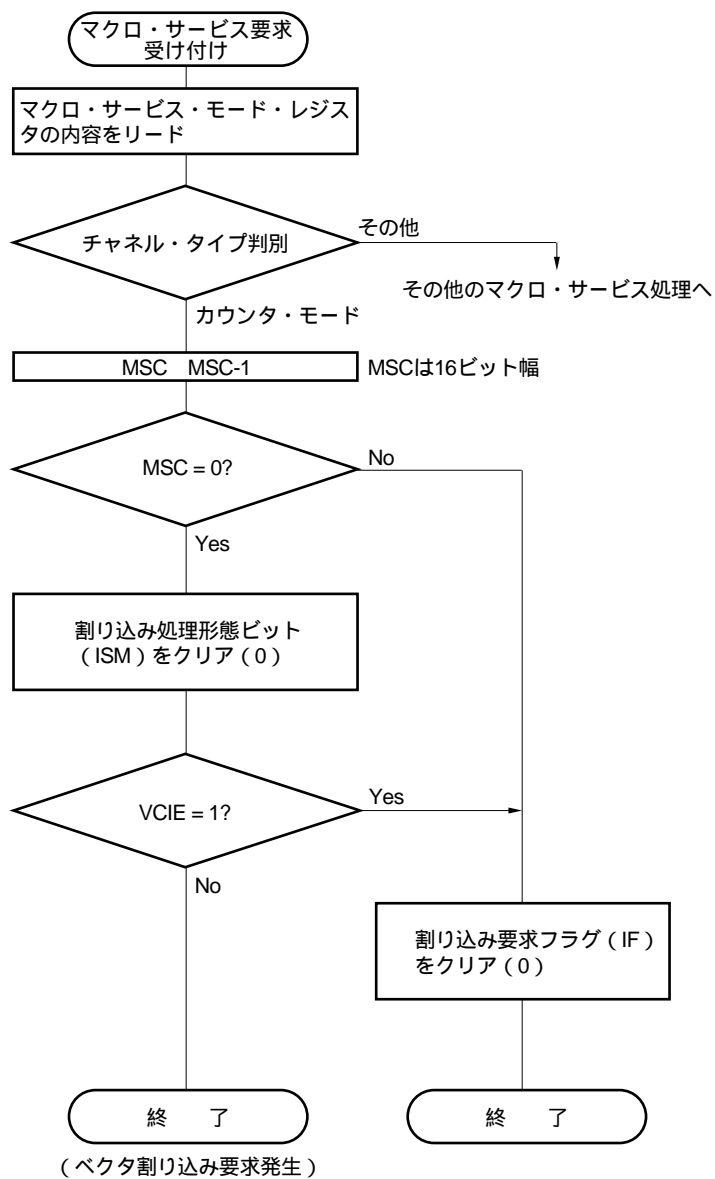
### 14.8.9 カウンタ・モード

#### (1) 動作

あらかじめマクロ・サービス・カウンタ (MSC) に設定した回数だけ、MSCのデクリメントを行います。

割り込みの発生回数を数えることができるので、割り込み発生周期が長いときに、イベント・カウンタとして使用することができます。

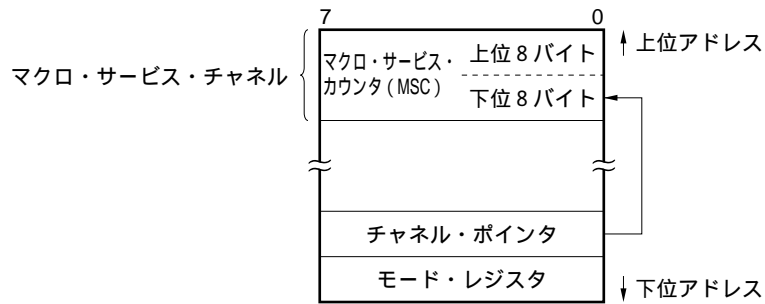
図14 - 31 マクロ・サービス・データ転送処理フロー (カウンタ・モード)



(2) マクロ・サービス・チャンネルの構成

16ビットのマクロ・サービス・カウンタ (MSC) のみでマクロ・サービス・チャンネルは構成されま  
す。チャンネル・ポインタには, MSCのアドレスの下位 8 ビットを書き込みます。

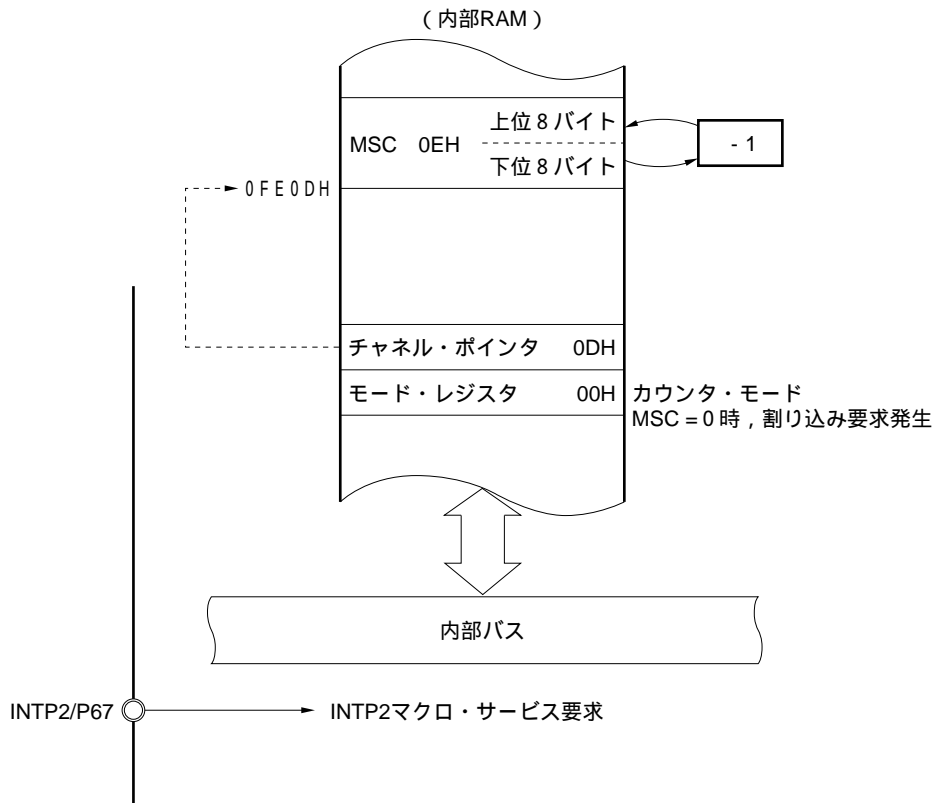
図14 - 32 カウンタ・モード



(3) カウンタ・モード使用例

外部割り込み端子 INTP2に入力されたエッジの回数をカウントする例を示します。

図14 - 33 エッジの回数のカウント



**備考** 図中の内部RAMのアドレスは, LOCATION 0 命令実行時の値です。  
LOCATION 0FH命令実行時には, 図中の値に0F0000Hを加えてください。



## 14.9 割り込み要求およびマクロ・サービスが一時的に保留される場合

次に示す命令を実行すると、8システム・クロックの間、割り込みの受け付けおよびマクロ・サービスの処理が一時的に保留されます。ただし、ソフトウェア割り込みは保留されません。

EI

DI

BRK

BRKCS

RETCS

RETCSB !addr16

RETI

RETB

LOCATION 0HまたはLOCATION 0FH

POP PSW

POPU post

MOV PSWL, A

MOV PSWL, byte

MOVG SP, #imm24

割り込み制御レジスタ<sup>注</sup>、MK0、IMC、ISPRの各レジスタに対する書き込み命令およびビット操作命令  
(BT, BF命令を除く)

PSWのビット操作命令

(ただし、BT PSWL, bit, \$ addr20, BF PSWL. bit, \$ addr20, BT PSWH.bit, \$addr20,  
BF PSWH.bit, \$addr20, SET1 CY, NOT1 CY, CLR1 CY命令を除く)。

**注** 割り込み制御レジスタ：WDTIC, PIC0, PIC1, PIC2, TMIC00, TMIC01, KSIC, CSIC0, CSIC1,  
TMIC50, TMIC51, ADIF

**注意** 該当する命令群を連続して使用する場合で、割り込みやマクロ・サービスが保留されている期間が長くなつては困る場合には、NOP命令などを途中に挿入して割り込みやマクロ・サービスが受け付けられるタイミングを作ってください。

## 14.10 割り込みおよびマクロ・サービスで一時的に実行が中断される命令

次に示す命令は、受け付け可能な割り込み要求およびマクロ・サービスの要求により、一時的にその命令の実行を中断し、割り込みおよびマクロ・サービスを受け付けます。中断された命令は、割り込みサービス・プログラムの終了後、またはマクロ・サービス処理の終了後に再開します。

### 一時的に中断される命令

MOVM, XCHM, MOVBK, XCHBK  
 CMPME, CMPMNE, CMPMC, CMPMNC  
 CMPBKE, CMPBKNE, CMPBKC, CMPBKNC  
 SACW

## 14.11 割り込みおよびマクロ・サービスの動作タイミング

割り込み要求は、各ハードウェアで発生します。発生した割り込み要求は、割り込み要求フラグをセット（1）します。

割り込み要求フラグがセット（1）されると、優先順位などの判定を行うために8クロック（ $0.64 \mu\text{s}$ ,  $f_{\text{CLK}} = 12.5 \text{ MHz}$ ）かかります。

その後、実行中の命令が終了した時点で、その割り込みまたはマクロ・サービスの受け付けが許可されていれば、その割り込み要求の受け付け処理を行います。なお、実行中の命令が割り込みおよびマクロ・サービスを一時的に保留する命令であれば、その次の命令終了後に受け付けられます（保留する命令については、14.9 割り込み要求およびマクロ・サービスが一時的に保留される場合参照）。

図14 - 34 割り込み要求の発生と受け付け（単位：クロック =  $1/f_{\text{CLK}}$ ）



### 14.11.1 割り込みの受け付け処理時間

割り込み要求の受け付けには、表14 - 7のような時間がかかります。表14 - 7の時間後に、割り込み処理プログラムの実行を開始します。

表14 - 7 割り込み受け付け処理時間

(単位：クロック =  $1/f_{CLK}$ )

ベクタ・テーブル	IROM						EMEM					
	IROM, PRAM			EMEM			PRAM			EMEM		
スタック	IRAM	PRAM	EMEM	IRAM	PRAM	EMEM	IRAM	PRAM	EMEM	IRAM	PRAM	EMEM
ベクタ割り込み	26	29	$37 + 4n$	27	30	$38 + 4n$	30	33	$41 + 4n$	31	34	$42 + 4n$
コンテキスト・スイッチング	22	-	-	23	-	-	22	-	-	23	-	-

備考1．IROM：内部ROM（ただし、高速フェッチ指定時）。

PRAM：内部RAMの周辺RAM（ただし、分岐先の場合はLOCATION 0 命令実行時のみ）。

IRAM：内部高速RAM。

EMEM：外部メモリおよび高速フェッチを指定していないときの内部ROM。

- 2．nは、スタックへの書き込み時に必要となる1バイト当たりのウェイト数（ウェイト数はアドレス・ウェイトとアクセス・ウェイトの合計です）。
- 3．ベクタ・テーブルがEMEMの場合で、ベクタ・テーブルの読み出し時にウェイトを挿入しているときは、表中の値に、ベクタ割り込み時は2m、コンテキスト・スイッチング時はmを加えてください。ただし、mはベクタ・テーブルの読み出し時に必要となる1バイト当たりのウェイト数です。
- 4．分岐先がEMEMの場合で、分岐先の命令の読み込み時にウェイトを挿入しているときは、そのウェイト数を加えてください。
- 5．スタックがPRAMにとられている場合で、スタック・ポインタ（SP）の値が奇数の場合は、表中の値に4を加えてください。
- 6．ウェイト数は、アドレス・ウェイトとアクセス・ウェイトの合計数です。

### 14.11.2 マクロ・サービスの処理時間

マクロ・サービス処理は、マクロ・サービスのタイプなどにより異なり、表14 - 8のような時間となります。

表14 - 8 マクロ・サービス処理時間

(単位：クロック =  $1/f_{CLK}$ )

マクロ・サービスの処理タイプ			データ領域	
			IRAM	その他
タイプA	SFR メモリ	1バイト	24	-
		2バイト	25	-
	メモリ SFR	1バイト	24	-
		2バイト	26	-
タイプB	SFR メモリ		33	35
	メモリ SFR		34	36
タイプC			49	53
カウンタ・モード	MSC = 0		17	-
	MSC = 0		25	-

備考1．IRAM：内部高速RAM。

2．その他のデータ領域で次の場合には、各場合ごとに指定されたクロック数を加算してください。

- ・ IROM, PRAMでデータのサイズが2バイトで、奇数番地にデータが配置されている場合：4クロック。
- ・ EMEMでデータ・サイズが1バイトの場合：データ・アクセス時のウエイト数。
- ・ EMEMでデータ・サイズが2バイトの場合： $4 + 2n$  (ただし、 $n$ は1バイト当たりのウエイト数)。

3．タイプA, タイプB, タイプCで、MSC = 0となる場合は、1クロックを加算してください。

4．タイプCの場合、使用する機能やそのときの状態に応じて次の値を加算してください。

- ・ リング制御：4クロック。ただし、リング制御時にリング・カウンタが0になった場合は、さらに7クロックを加算。

## 14.12 割り込み機能を初期状態に戻す方法

プログラムの暴走や、システムの異常をオペランド・エラー割り込みやウォッチドッグ・タイマなどで検出した場合、システム全体を初期状態に戻す必要があります。μPD784975は、割り込みの受け付けに関する優先順位の制御をハードウェアによって行います。この割り込みの受け付けに関するハードウェアも、初期状態にしないと以降の割り込みの受け付けに関する制御が異常になる場合があります。

次に、割り込みの受け付けに関するハードウェアをプログラムで初期化する方法を示します。なお、ハードウェアで初期化する方法は、 $\overline{\text{RESET}}$ 入力しかありません。

```

例          MOVW MK0,  0FFFFH ; マスカブル割り込みはすべてマスクする
          MOV  MK1L,  0FFH

IRESL  :
          CMP  ISPR,  0          ; すべての処理中の割り込みサービス・プログラムがなくなったか?
          BZ   $NEXT
          MOVG SP,  RETVAL      ; SPの位置を強制的に変更
          RETI                   ; 処理中の割り込みサービス・プログラムを強制終了, 戻り番地はIRESL

RETVAL  :
          DW   LOWW (IRESL)     ; RETI命令でIRESLへ戻るためのスタックのデータ
          DB   0
          DB   HIGHW (IRESL)    ; LOWW, HIGHWは、それぞれシンボルの下位16ビット, 上位16ビットの値を計算するアセンブラの演算子

NEXT    :
          ・この後、内蔵周辺ハードウェアの初期化および割り込み制御レジスタの初期化を行う
          ・割り込み制御レジスタの初期化時には、割り込み要求フラグを必ずクリア(0)する

```

## 14.13 注意事項

- (1) インサースervice・プライオリティ・レジスタ (ISPR) は、読み出しのみ可能です。書き込みを行うと、誤動作する恐れがあります。
- (2) ウォッチドッグ・タイマ・モード・レジスタ (WDM) は、専用命令 (MOV WDM, byte) でのみ書き込みが可能です。
- (3) BRK命令によるソフトウェア割り込みからの復帰に、RETI命令を使用してはいけません。RETB命令を使用してください。
- (4) BRKCS命令によるソフトウェア割り込みからの復帰に、RETCS命令を使用してはいけません。RETCSB命令を使用してください。
- (5) マスカブル割り込みをベクタ割り込みで受け付けた場合は、必ずRETI命令で復帰してください。他の命令では、以降の割り込みに関する動作が正常に行われません。
- (6) コンテキスト・スイッチングによる割り込みからの復帰には、必ずRETCS命令を使用してください。他の命令では、以降の割り込みに関する動作が正常に行われません。
- (7) ノンマスカブル割り込みサービス・プログラム中でもマクロ・サービスの要求は受け付けられ、処理されます。ノンマスカブル割り込みサービス・プログラム中でマクロ・サービスの処理を行いたくない場合は、ノンマスカブル割り込みサービス・プログラム中で割り込みマスク・レジスタを操作して、マクロ・サービスが発生しないようにしてください。
- (8) ノンマスカブル割り込みからの復帰には、必ずRETI命令を使用してください。他の命令では、以降の割り込みの受け付けが正常に行われません。ノンマスカブル割り込み受け付け後、初期状態からプログラムを再開したい場合は、14.12 **割り込み機能を初期状態に戻す方法**を参照してください。
- (9) ノンマスカブル割り込みは、ノンマスカブル割り込みサービス・プログラム実行中 (優先順位の低いノンマスカブル割り込みサービス・プログラム実行中に、優先順位の高いノンマスカブル割り込みの要求が発生した場合を除く) および14.9に示す特定命令の実行後の一定期間を除いては必ず受け付けられます。したがって、特にリセット解除後などのスタック・ポインタ (SP) の値が不定の場合でもノンマスカブル割り込みを受け付けます。このときのSPの値によっては特殊機能レジスタ (SFR) の書き込みを禁止しているアドレス (3.8 **特殊機能レジスタ (SFR) の表3 - 6**参照)へプログラム・カウンタ (PC) およびプログラム・ステータス・ワード (PSW) を書き込んでしまい、CPUがデッドロックしてしまったり、端子から予期しない信号を出力したり、RAMが実装されていないアドレスへPCやPSWを書き込むことにより、ノンマスカブル割り込み処理プログラムからメイン・ルーチンへ正常に戻れず暴走したりする場合があります。

したがって、RESET解除後のプログラムは必ず次のようにしてください。

```
CSEG AT 0
DW   STRT
CSEG BASE

STRT :
LOCATION 0FH ; or LOCATION 0
MOVG SP, imm24
```

- (10) 該当する命令群を連続して使用する場合で、割り込みやマクロ・サービスが保留されている期間が長くなつては困る場合には、NOP命令などを途中に挿入して割り込みやマクロ・サービスが受け付けられるタイミングを作ってください。

〔メモ〕



# 第15章 スタンバイ機能

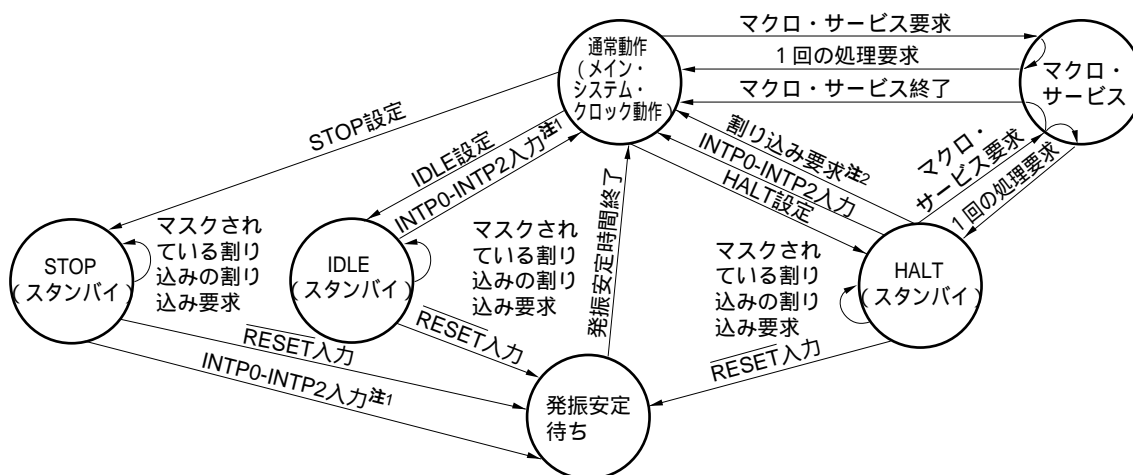
## 15.1 構成と機能

μPD784975は、システムの消費電力を低減させることのできるスタンバイ機能を持っています。スタンバイ機能には次のような3つのモードがあります。

- ・HALTモード ..... CPUの動作クロックを停止させるモードです。通常動作モードとの組み合わせによる間欠動作により、システムのトータル消費電力を低下させることができます。
- ・IDLEモード ..... 発振回路の動作を継続したまま、それ以外のシステム全体が停止するモードです。STOPモードに近い低消費電力と、HALTモードと同等の時間で、通常のプログラム動作に復帰できます。
- ・STOPモード ..... 発振回路を停止させ、システム全体が停止するモードです。リーク電流だけの超低消費電力にすることができます。

各モードにはソフトウェアによって設定します。図15 - 1に、スタンバイ・モード (STOP/IDLE/HALTモード) の遷移図を示します。

図15 - 1 スタンバイ・モードの遷移図



注1 . INTP0-INTP2はマスクされていない場合

2 . マスクされていない割り込み要求のみ

備考 ウォッチドッグ・タイマは、スタンバイの解除 (STOPモード / HALTモード / IDLEモード) には使用できません。

## 15.2 制御レジスタ

### 15.2.1 スタンバイ・コントロール・レジスタ (STBC)

STBCは、STOPモードの設定と、内部システム・クロックを選択するためのレジスタです。

プログラムの暴走によって誤ってスタンバイ・モードに入らないように、専用命令によってのみ書き込みができます。この専用命令は、MOV STBC, byte命令で、特殊なコード構成(4バイト)になっています。3バイト目と4バイト目のオペコードが互いに1の補数の場合のみ、書き込みが行われます。

また、3バイト目と4バイト目のオペコードが互いに1の補数でなければ書き込みが行われず、オペランド・エラー割り込みが発生します。この場合、スタック領域に退避されるリターン・アドレスは、エラーの原因となった命令のアドレスです。したがって、スタック領域に退避されたリターン・アドレスにより、エラーの原因となったアドレスを知ることができます。

なお、単純にオペランド・エラーからRETB命令で復帰を行うと、無限ループとなります。

オペランド・エラー割り込みは、プログラムが暴走している場合のみ発生する(NEC製アセンブラRA78K4では、MOV STBC, byteと記述された場合、正しい専用命令しか生成しません)ので、システムの初期化をプログラムで行うようにしてください。

他の書き込み命令(MOV STBC, AやAND STBC, byteやSET1 STBC.7など)は無視され、何の動作も行いません。つまり、STBCへの書き込みは行われず、また、オペランド・エラー割り込みのような割り込みも発生しません。

STBCの読み出しは、データ転送命令によりいつでもできます。

$\overline{\text{RESET}}$ 入力により30Hになります。

図15-2に、STBCのフォーマットを示します。

図15 - 2 スタンバイ・コントロール・レジスタ (STBC) のフォーマット

アドレス : 0FFC0H リセット時 : 30H R/W

略号	7	6	5	4	3	2	1	0
STBC	0	0	CK1	CK0	0	0	STP	HLT

CK1	CK0	CPUクロックの選択 <sup>注</sup> (スループット・クロック・モード時, 発振分周モード時)
0	0	$f_{xx}$ ( $f_x, f_x/2$ )
0	1	$f_{xx}/2$ ( $f_x/2, f_x/2^2$ )
1	0	$f_{xx}/2^2$ ( $f_x/2^2, f_x/2^3$ )
1	1	$f_{xx}/2^3$ ( $f_x/2^3, f_x/2^4$ )

STP	HLT	動作指定フラグ
0	0	通常動作モード
0	1	HALTモード (HALTモードが解除されると自動的にクリア)
1	0	STOPモード (STOPモードが解除されると自動的にクリア)
1	1	IDLEモード (IDLEモードが解除されると自動的にクリア)

注 CPUクロックの選択は、発振モード選択レジスタ (CC) でも設定できます。

注意 1 . 外部クロック入力時にSTOPモードを使用する場合は、発振安定時間指定レジスタ (OSTS) のEXTCビットを必ずセット (1) してからSTOPモードを設定してください。外部クロック入力時にOSTSのEXTCビットがクリア (0) されている状態でSTOPモードを使用すると必ず  $\mu$ PD784975の破壊または信頼性の低下を招く可能性があります。なお、OSTSのEXTCビットをセット (1) する場合には、必ずX2端子に、X1端子に入力しているクロックの逆相のクロックを入力してください。

2 . スタンバイ命令のあと (スタンバイ解除後) は、NOP命令を3回実行してください。そうしないと、スタンバイ命令の実行と割り込み要求が競合した場合に、スタンバイ命令を実行せず、スタンバイ命令に続く複数の命令を実行後に割り込み要求を受け付けます。割り込み受け付け前に実行する命令は、スタンバイ命令実行後の最大6クロック以内に実行を開始する命令です。

例 MOVE STBC, byte  
NOP  
NOP  
NOP

**備考**  $f_{xx}$  : メイン・システム・クロック周波数 ( $f_x$ または $f_x/2$ )

$f_x$  : メイン・システム・クロック発振周波数

## 15.2.2 発振安定時間指定レジスタ (OSTS)

OSTSは、発振回路の動作およびSTOPモード解除時の発振安定時間を指定するレジスタです。OSTSのEXTCビットにクリスタル/セラミック発振または外部クロックのどちらを使用しているかを設定します。EXTCビットをセット(1)している場合だけ、外部クロック入力時にもSTOPモードを設定することができます。

また、OSTSのOSTS0-OSTS2ビットで、STOPモード解除時の発振安定時間を選択します。一般的に、水晶振動子を使用する場合は40 ms以上、セラミック発振子を使用する場合は4 ms以上の発振安定時間を選択するようにしてください。

なお、発振安定にかかる時間は、使用する水晶振動子/セラミック発振子および、接続するコンデンサの容量などにより影響を受けます。したがって、発振安定用の時間を短かめに設定したい場合には、水晶振動子/セラミック発振子メーカーと相談してくださいますようお願いいたします。

OSTSは、1ビット転送命令または8ビット転送命令で設定します。

$\overline{\text{RESET}}$ 入力により00Hにクリアされます。

図15-3に、OSTSのフォーマットを示します。

図15 - 3 発振安定時間指定レジスタ (OSTS) のフォーマット

アドレス : 0FFCFH リセット時 : 00H R/W

略号	7	6	5	4	3	2	1	0
OSTS	EXTC	0	0	0	0	OSTS2	OSTS1	OSTS0

EXTC	外部クロックの選択
0	クリスタル/セラミック発振使用時
1	外部クロック使用時

EXTC	OSTS2	OSTS1	OSTS0	発振安定時間の選択
0	0	0	0	$2^{19}/f_{xx}$ ( 41.9 ms )
0	0	0	1	$2^{18}/f_{xx}$ ( 21.0 ms )
0	0	1	0	$2^{17}/f_{xx}$ ( 10.5 ms )
0	0	1	1	$2^{16}/f_{xx}$ ( 5.2 ms )
0	1	0	0	$2^{15}/f_{xx}$ ( 2.6 ms )
0	1	0	1	$2^{14}/f_{xx}$ ( 1.3 ms )
0	1	1	0	$2^{13}/f_{xx}$ ( 655 $\mu$ s )
0	1	1	1	$2^{12}/f_{xx}$ ( 328 $\mu$ s )
1	x	x	x	$512/f_{xx}$ ( 41.0 $\mu$ s )

- 注意 1 . クリスタル/セラミック発振使用時には、EXTCビットを必ずクリア ( 0 ) して使用してください。EXTCビットをセット ( 1 ) すると、発振が停止してしまいます。
- 2 . 外部クロック入力時にSTOPモードを使用する場合は、EXTCビットを必ずセット ( 1 ) してからSTOPモードを設定してください。外部クロック入力時にEXTCビットがクリア ( 0 ) されている状態でSTOPモードを使用すると、 $\mu$ PD784975の破壊または信頼性の低下を招く可能性があります。
- 3 . 外部クロック入力時でEXTCビットをセット ( 1 ) する場合には、必ずX2端子に、X1端子に入力しているクロックの逆相のクロックを入力してください。EXTCビットをセット ( 1 ) すると、 $\mu$ PD784975はX2端子に入力されているクロックでのみ動作するようになります。

備考 1 . ( ) 内は、 $f_{xx} = 12.5$  MHz動作時

2 . x : don't care

## 15.3 HALTモード

### 15.3.1 HALTモードの設定および動作状態

HALTモードには、スタンバイ・コントロール・レジスタ (STBC) のHLTビットをセット (1) することにより設定されます。

STBCの書き込みは、専用命令による8ビット・データ書き込みのみ可能です。したがって、HALTモードの設定は、“MOV STBC, byte”命令で行います。

割り込み許可 (プログラム・ステータス・ワード (PSW) のIEビットがセット (1)) に設定している場合、HALTモード設定の命令のあと (HALTモード解除後) にはNOP命令を3回記述してください。そうしないとHALTモード解除後、割り込みを受け付ける前に複数の命令を実行してしまう可能性があります。これにより、割り込み処理と命令の実行の順序関係が変化してしまうことがあります。実行順序が変化することによる不具合を防止するために、前述の処理が必要です。

**注意** HALTモードを解除する条件が成立しているときにHALTモードの設定を行った場合、HALTモードに入らずに次の命令の実行、またはベクタ割り込みサービス・プログラムへの分岐を行います。確実にHALTモードの設定をしたい場合は、HALTモードに入る前に割り込み要求をクリアするなどしてください。

表15 - 1 HALTモード時の動作状態

クロック発振回路	動作
内部システム・クロック	動作
CPU	動作停止 <sup>注</sup>
I/Oライン	HALTモード設定前の状態を保持
各周辺機能	動作継続
内部RAM	保持

注 マクロ・サービス処理は実行されます。

### 15.3.2 HALTモードの解除

HALTモードは、次の2つのソースによって解除することができます。

- ・マスカブル割り込み要求 (ベクタ割り込み、コンテキスト・スイッチング、マクロ・サービス)
- ・ $\overline{\text{RESET}}$  入力

表15 - 2に、解除ソースと解除後の動作の概要を示します。また、図15 - 4に、HALTモード解除後の動作を示します。

表15 - 2 HALTモードの解除と解除後の動作

解除ソース	MK <sup>注1</sup>	IE <sup>注2</sup>	解除時の状態	解除後の動作
RESET入力	x	x	-	通常のリセット動作
マスカブル割り込み要求 (マクロ・サービスの要求を除く)	0	1	・ 割り込みサービス・プログラムの実行中でない ・ 優先順位の低いマスカブル割り込みサービス・プログラムの実行中 ・ 優先順位レベル3の割り込みサービス・プログラムの実行中で、PRSLビット <sup>注4</sup> がクリア(0)されている	割り込み要求の受け付け
			・ 同一優先順位のマスカブル割り込みサービス・プログラム実行中 (PRSLビット <sup>注4</sup> がクリア(0)されている場合で、優先順位レベル3の割り込みサービス・プログラム実行中を除く) ・ 優先順位の高い割り込みサービス・プログラム実行中	
	0	0	-	
	1	x	-	HALTモード保持
マクロ・サービス要求	0	x	-	マクロ・サービス処理実行 終了条件不成立 再びHALTモード 終了条件成立 VCIE <sup>注5</sup> =1のとき：再びHALTモード VCIE <sup>注5</sup> =0のとき：マスカブル割り込み要求による解除と同じ
				1

注1．各割り込み要求ソースにある割り込みマスク・ビット

2．プログラム・ステータス・ワード (PSW) 中にある割り込み許可フラグ

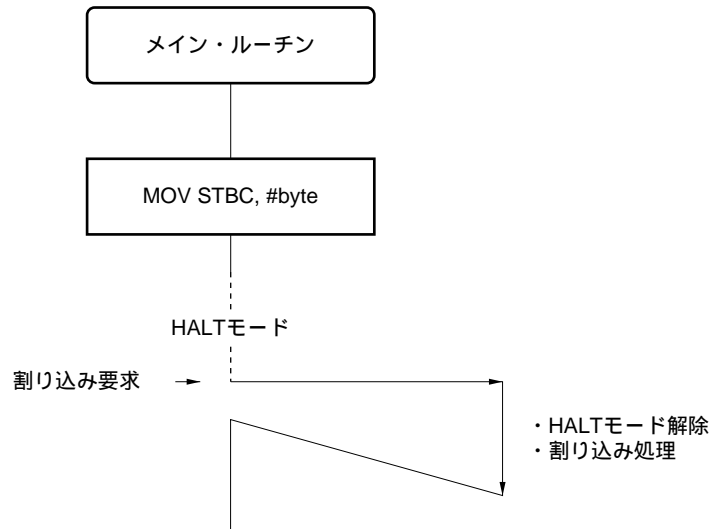
3．保留された割り込み要求は、受け付け可能になった時点で受け付けられる

4．割り込みモード・コントロール・レジスタ (IMC) 中のビット

5．各マクロ・サービス要求ソースにあるマクロ・サービス・コントロール・ワードのマクロ・サービス・モード・レジスタ中のビット

図15 - 4 HALTモード解除後の動作 (1/4)

(1) HALTモードに入ったあと、割り込みが入った場合



(2) HALTモードに入ったあとリセット

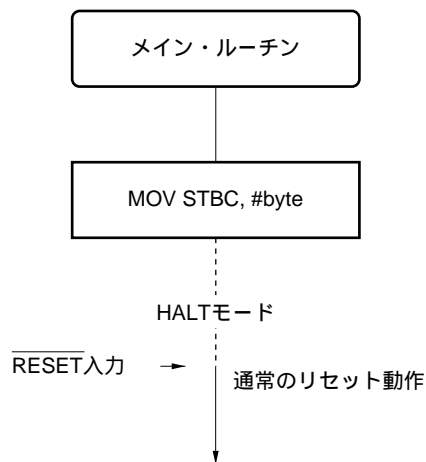
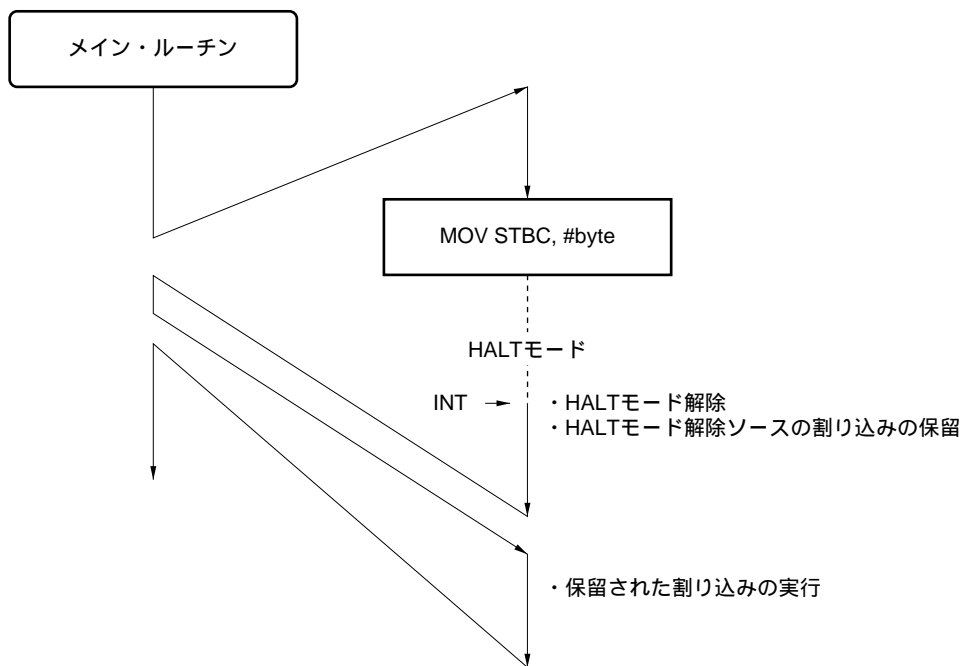




図15 - 4 HALTモード解除後の動作 (2/4)

(3) 解除ソースの割り込みよりも優先順位の高い、あるいは同じ割り込み処理ルーチン中にHALTモードに入った場合



(4) 解除ソースの割り込みよりも優先順位の高い割り込み処理ルーチン中にHALTモードに入った場合

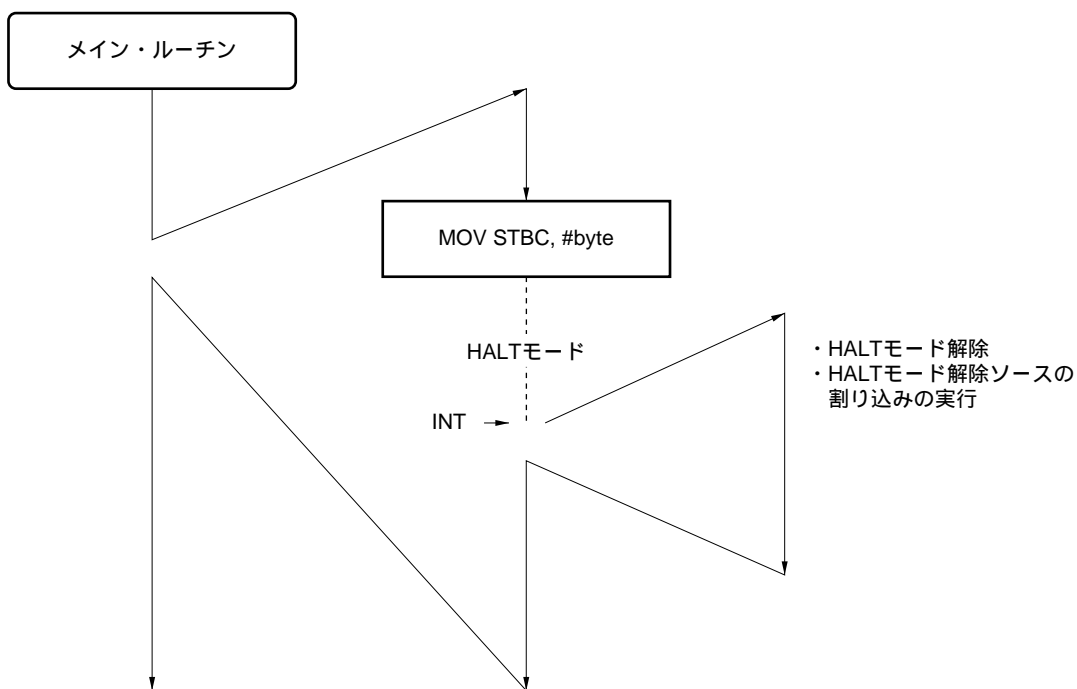
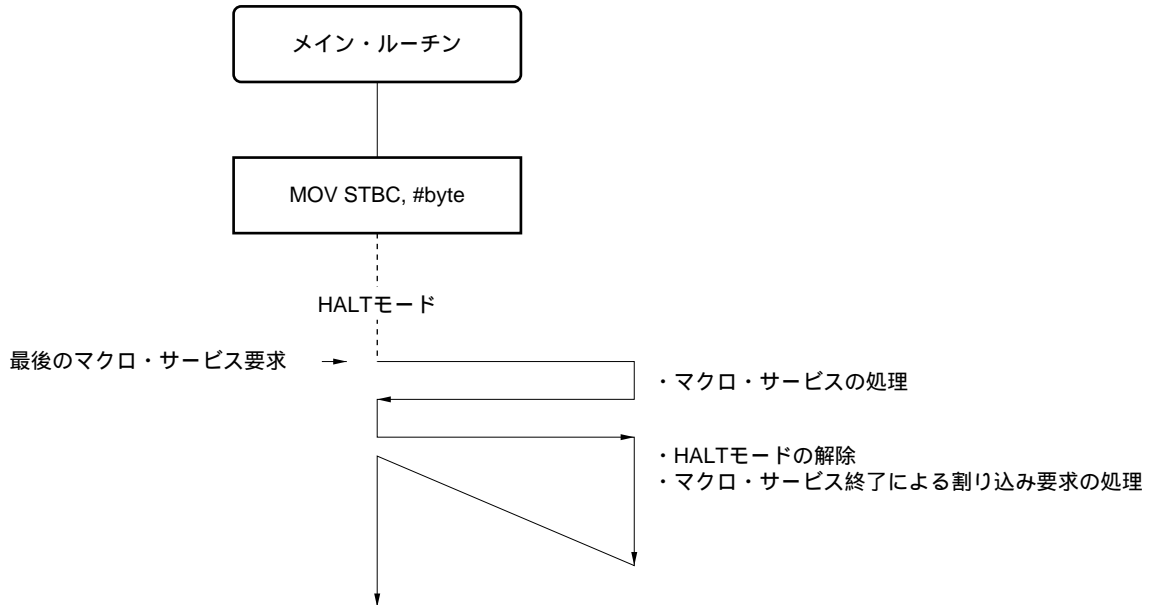


図15 - 4 HALTモード解除後の動作 (3/4)

(5) HALTモード中にマクロ・サービスの要求があった場合

(a) マクロ・サービスの終了条件が成立し、成立後、ただちに割り込み要求を発生する (VCIE = 0) のとき



(b) マクロ・サービスの終了条件が成立しないとき、またはマクロ・サービスの終了条件が成立し、成立後、割り込み要求を発生しない (VCIE = 1) のとき

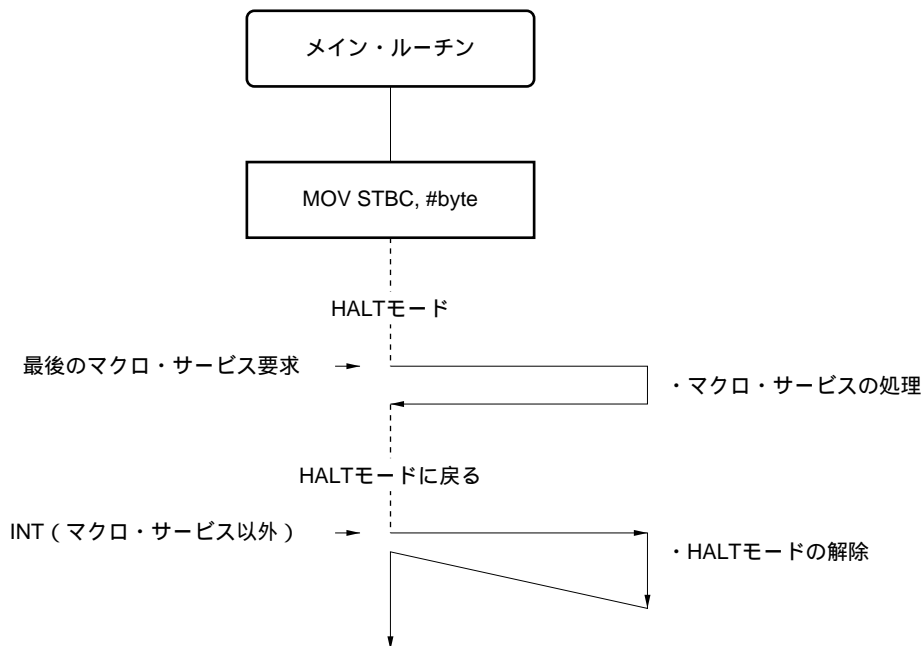
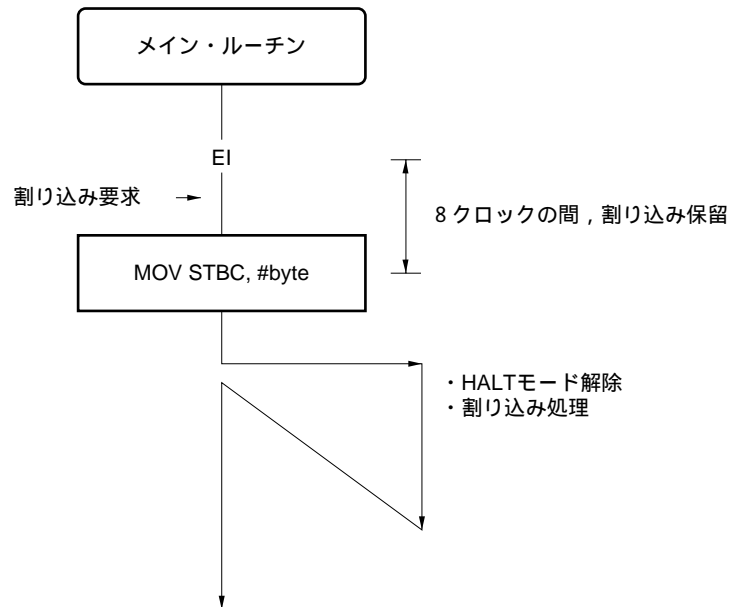
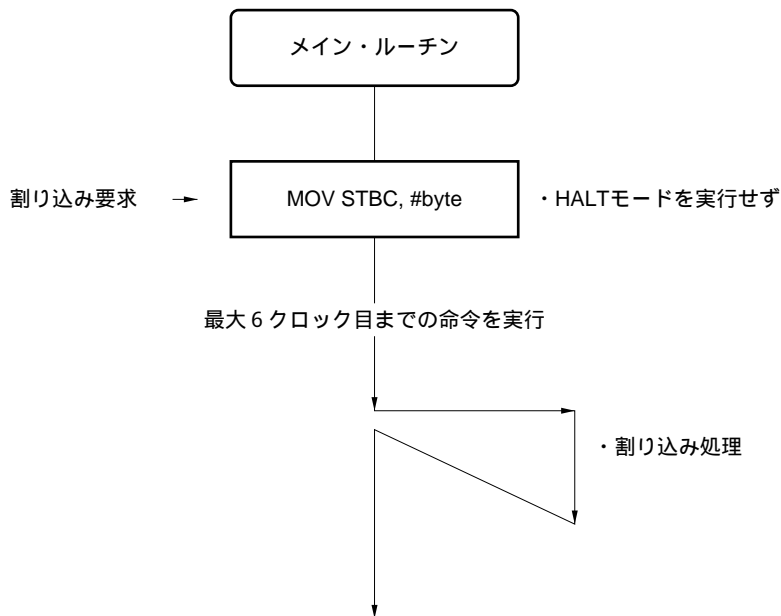


図15 - 4 HALTモード解除後の動作 (4/4)

(6) 割り込み要求が一時的に保留される命令において割り込みが入り、その割り込みが保留中にHALTモードに入った場合



(7) HALTモード設定命令と割り込みが競合する場合



### (1) マスカブル割り込み要求による解除

マスカブル割り込み要求によるHALTモードの解除は、割り込みマスク・フラグが0の割り込みによつてのみ可能です。

HALTモードが解除されると、割り込み要求許可フラグ(IE)がセット(1)されている場合は、その割り込みが受け付け可能であれば、割り込みサービス・プログラムへ分岐します。受け付けが不可能な場合とIEフラグがクリア(0)されている場合は、HALTモードを設定した次の命令から実行を再開します。割り込みの受け付けについての詳細は、14.6 **マスカブル割り込みの受け付け動作**を参照してください。

マクロ・サービスは、HALTモードを一時的に解除して、1回の処理を行い、再びHALTモードに戻ります。マクロ・サービスが指定回数だけ行われると、マクロ・サービス・コントロール・ワードのマクロ・サービス・モード・レジスタ中にあるVCIEビットがクリア(0)されている場合は、HALTモードを解除します。この解除後の動作は、前述したマスカブル割り込みによる解除と同一です。また、VCIEビットがセット(1)されている場合は、再びHALTモードとなり、次の割り込みの要求でHALTモードを解除します。

表15 - 3 マスカブル割り込み要求によるHALTモードの解除

解除ソース	MK <sup>注1</sup>	IE <sup>注2</sup>	解除時の状態	解除後の動作
マスカブル割り込み要求 (マクロ・サービスの要求を除く)	0	1	・割り込みサービス・プログラムの実行中でない	割り込み要求の受け付け
			・優先順位の低いマスカブル割り込みサービス・プログラムの実行中	
	・優先順位レベル3の割り込みサービス・プログラムの実行中で、PRSLビット <sup>注4</sup> がクリア(0)されている	MOV STBC, byte命令の次の命令を実行 (HALTモードを解除した割り込み要求は、保留される <sup>注3</sup> )		
	0	0	-	
	1	x	-	HALTモード保持
マクロ・サービス要求	0	x	-	マクロ・サービス処理実行 終了条件不成立 再びHALTモード 終了条件成立 VCIE <sup>注5</sup> = 1 のとき：再びHALTモード VCIE <sup>注5</sup> = 0 のとき：マスカブル割り込み要求による解除と同じ
			1	x

注1．各割り込み要求ソースにある割り込みマスク・ビット

- 2．プログラム・ステータス・ワード (PSW) 中にある割り込み許可フラグ
- 3．保留された割り込み要求は、受け付け可能になった時点で受け付けられる
- 4．割り込みモード・コントロール・レジスタ (IMC) 中のビット
- 5．各マクロ・サービス要求ソースにあるマクロ・サービス・コントロール・ワードのマクロ・サービス・モード・レジスタ中のビット

(2) RESET入力による解除

通常のリセット動作と同様にリセット・ベクタ・アドレスに分岐した後、プログラムを実行します。ただし、内部RAMの内容は、HALTモード設定直前の値を保持しています。

## 15.4 STOPモード

### 15.4.1 STOPモードの設定および動作状態

STOPモードには、スタンバイ・コントロール・レジスタ (STBC) のSTPビットをセット (1) することにより設定されます。

STBCの書き込みは、専用命令による8ビット・データ書き込みのみ可能です。したがって、STOPモードの設定は、“MOV STBC, byte” 命令で行います。

割り込み許可 (プログラム・ステータス・ワード (PSW) のIEビットがセット (1)) に設定している場合、STOPモード設定の命令のあと (STOPモード解除後) にはNOP命令を3回記述してください。そうしないとSTOPモード解除後、割り込みを受け付ける前に複数の命令を実行してしまう可能性があります。これにより、割り込み処理と命令の実行の順序関係が変化してしまうことがあります。実行順序が変化することによる不具合を防止するために、前述の処理が必要です。

**注意** HALTモードを解除する条件 (15.3.2 HALTモードの解除参照) が成立しているときにSTOPモードの設定を行った場合、STOPモードには入らずに次の命令の実行、またはベクタ割り込みサービス・プログラムへの分岐を行います。確実にSTOPモードの設定をしたい場合は、STOPモードに入る前に割り込み要求をクリアするなどしてください。

表15 - 4 STOPモード時の動作状態

クロック発振回路	発振停止
内部システム・クロック	停止
CPU	動作停止
I/Oライン	STOPモード設定前の状態を保持
各周辺機能	すべて動作停止 <sup>注</sup>
内部RAM	保持

**注** A/Dコンバータは動作を停止しませんが、A/Dコンバータ・モード・レジスタ (ADM) のCSビットがセット (1) されていると、消費電流は減りません。

**注意 1** . 外部クロックを使用するシステムでSTOPモードを使用する場合は、OSTSのEXTCビットを必ずセット (1) してください。OSTSのEXTCビットがクリア (0) されているときに、外部クロックを入力するシステムでSTOPモードの設定を行うと、消費電流が増加します。

なお、OSTSのEXTCビットをセット (1) する場合には、必ずX2端子に、X1端子に入力しているクロックの逆相のクロックを入力する必要があります (5.4 システム・クロック発振回路参照)。

**2** . A/Dコンバータ・モード・レジスタ (ADM) のCSビットをクリア (0) してください。

### 15.4.2 STOPモードの解除

STOPモードは、INTP0-INTP2入力およびRESET入力により解除されます。

表15 - 5 に、解除ソースと解除後の動作の概要を示します。また、図15 - 5 に、STOPモード解除後の動作を示します。

表15 - 5 STOPモードの解除と解除後の動作

解除ソース	MK <sup>注1</sup>	ISM <sup>注2</sup>	IE <sup>注3</sup>	解除時の状態	解除後の動作
RESET入力	x	x	x	-	通常のリセット動作
INTP0-INTP2 端子入力	0	0	1	・割り込みサービス・プログラムの実行 中でない	割り込み要求の受け付け
				・優先順位の低いマスク割込み サービス・プログラムの実行中	
	0	0	0	・優先順位レベル3の割り込みサー ビス・プログラムの実行中で、PRSL ビット <sup>注5</sup> がクリア(0)されている	MOV STBC, byte命令の次の命令を実 行 (STOPモードを解除した割り込み要 求は、保留される <sup>注4</sup> )
				・同一優先順位のマスク割込み サービス・プログラム実行中 (PRSLビット <sup>注5</sup> がクリア(0)されて いる場合で、優先順位レベル3の割 り込みサービス・プログラム実行中を除 く)	
x	1	x	・優先順位の高い割り込みサービス・プ ログラム実行中	STOPモード保持	

注1 . 各割り込み要求ソースにある割り込みマスク・ビット

2 . 各割り込み要求ソースにあるマクロ・サービス許可フラグ

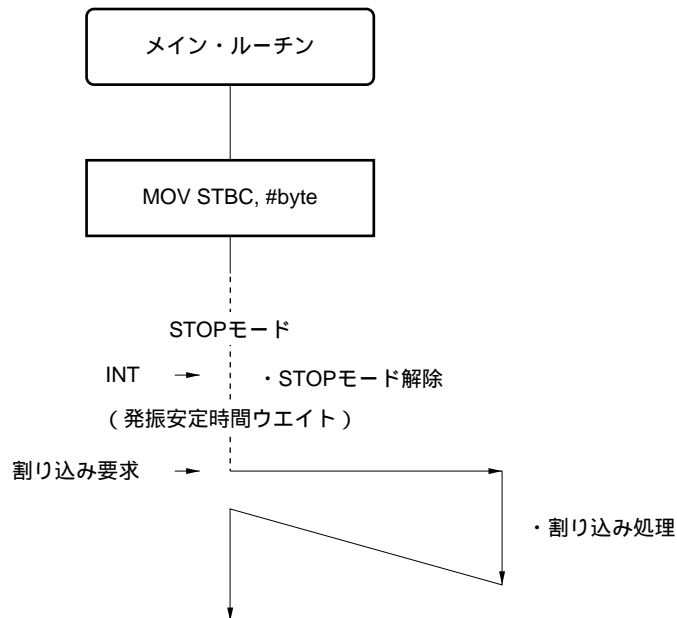
3 . プログラム・ステータス・ワード (PSW) 中にある割り込み許可フラグ

4 . 保留された割り込み要求は、受け付け可能になった時点で受け付けられる

5 . 割り込みモード・コントロール・レジスタ (IMC) 中のビット

図15 - 5 STOPモード解除後の動作 (1/3)

(1) STOPモードに入ったあと、割り込みが入った場合



(2) STOPモードに入ったあとリセット

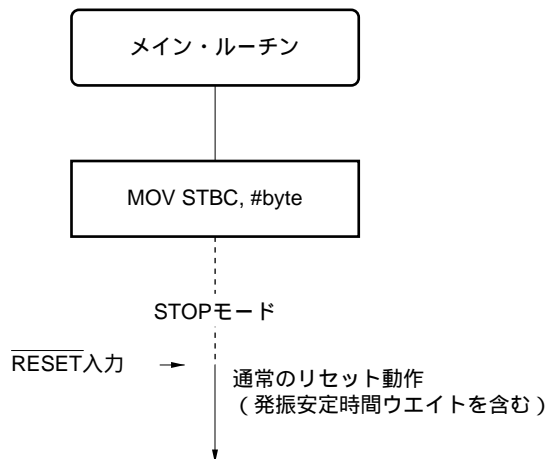
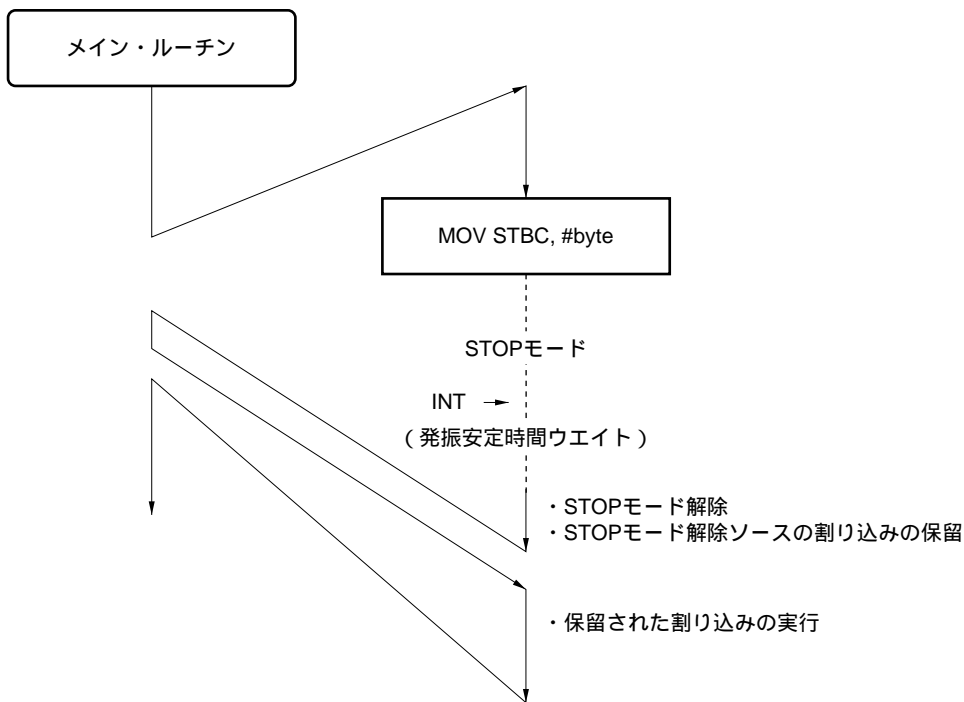




図15 - 5 STOPモード解除後の動作 (2/3)

(3) 解除ソースの割り込みよりも優先順位の高い、あるいは同じ割り込み処理ルーチン中にSTOPモードに入った場合



(4) 解除ソースの割り込みよりも優先順位の高い割り込み処理ルーチン中にSTOPモードに入った場合

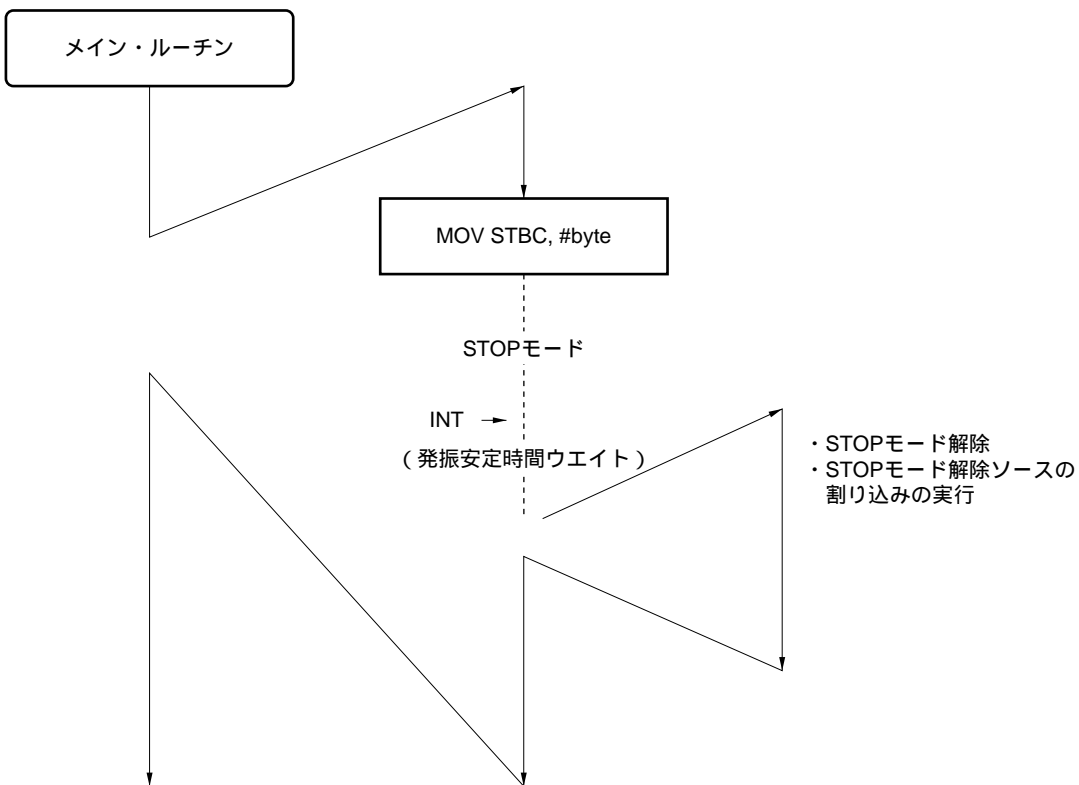
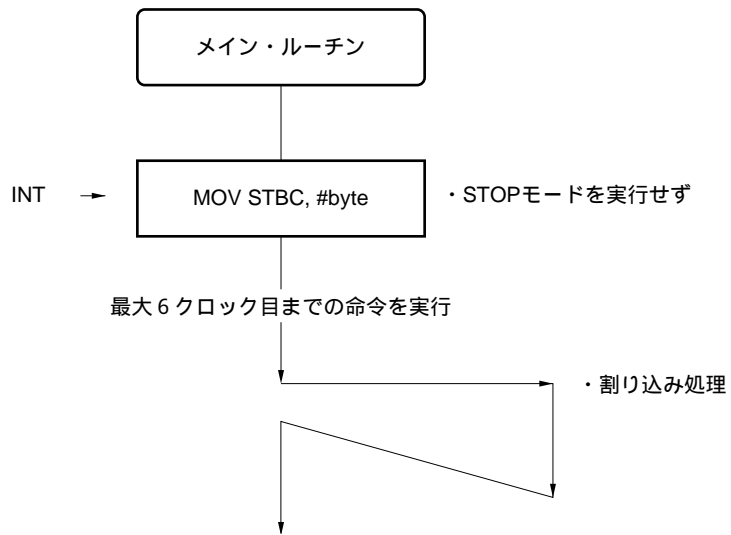


図15 - 5 STOPモード解除後の動作 (3/3)

(5) STOPモード設定命令と割り込みが競合する場合



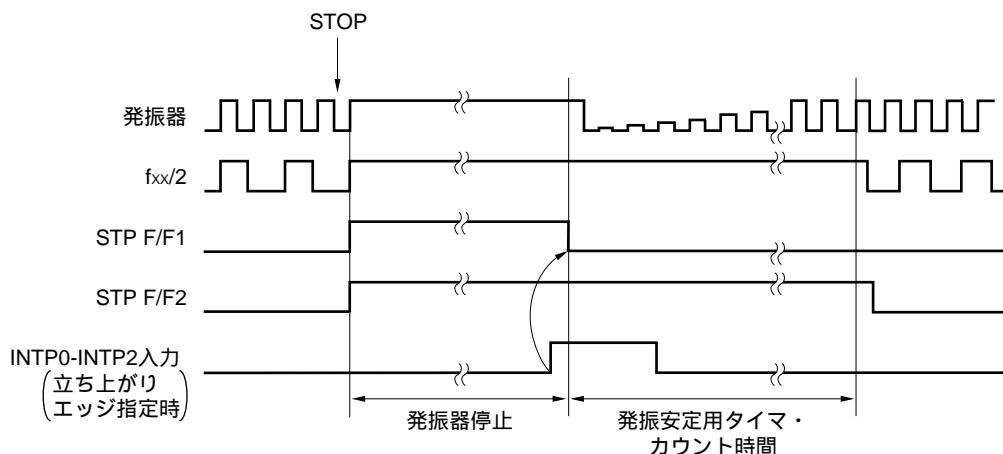
### (1) INTP0-INTP2入力によるSTOPモードの解除

INTP0-INTP2入力による割り込みのマスクが解除されていて、マクロ・サービスが禁止されている場合に、INTP0-INTP2入力に外部割り込みモード・レジスタ1 (INTM1) で指定した有効エッジが入力されると発振器が発振を再開します。その後、発振安定時間指定レジスタ (OSTS) で指定された発振安定時間後にSTOPモードは解除されます。

STOPモードから解除されると、割り込み許可フラグ (IE) がセット (1) されている場合は、その割り込みが受け付け可能であれば、割り込みサービス・プログラムに分岐します。受け付け不可能な場合とIEフラグがクリア (0) されている場合は、STOPモードを設定した命令の次の命令から実行を再開します。

割り込みの受け付けについての詳細は、14.6 マスカブル割り込みの受け付け動作を参照してください。

図15 - 6 INTP0-INTP2入力によるSTOPモードの解除



### (2) RESET入力によるSTOPモードの解除

$\overline{\text{RESET}}$ 入力をハイからロウに下げるとリセット状態になります。 $\overline{\text{RESET}}$ の立ち上がりでクロックの発振を開始し、発振安定用タイマがオーバーフローすると通常動作をスタートします。このとき、内蔵データ・メモリはSTOPモード設定前の内容を保持しています。

## 15.5 IDLEモード

### 15.5.1 IDLEモードの設定および動作状態

IDLEモードには、スタンバイ・コントロール・レジスタ（STBC）のSTPビットとHLTビットの両方をセット（1）することにより設定されます。

STBCの書き込みは、専用命令による8ビット・データ書き込みのみ可能です。したがって、IDLEモードの設定は、“MOV STBC, byte”命令で行います。

割り込み許可（プログラム・ステータス・ワード（PSW）のIEビットがセット（1））に設定している場合、IDLEモード設定の命令のあと（IDLEモード解除後）にはNOP命令を3回記述してください。そうしないとIDLEモード解除後、割り込みを受け付ける前に複数の命令を実行してしまう可能性があります。これにより、割り込み処理と命令の実行の順序関係が変化してしまうことがあります。実行順序が変化することによる不具合を防止するために、前述の処理が必要です。

**注意** HALTモードを解除する条件（15.3.2 HALTモードの解除参照）が成立しているときにIDLEモードの設定を行った場合、IDLEモードには入らずに次の命令の実行、またはベクタ割り込みサービス・プログラムへの分岐を行います。確実にIDLEモードの設定をしたい場合は、STOPモードに入る前に割り込み要求をクリアするなどしてください。

表15 - 6 IDLEモード時の動作状態

クロック発振回路	発振継続
内部システム・クロック	停止
CPU	動作停止
I/Oライン	IDLEモード設定前の状態を保持
各周辺機能	すべて動作停止 <sup>注</sup>
内部RAM	保持

**注** A/Dコンバータは動作を停止しませんが、A/Dコンバータ・モード・レジスタ（ADM）のCSビットがセットされていると、消費電流は減りません。

**注意** A/Dコンバータ・モード・レジスタ（ADM）のCSビットをリセットしてください。

### 15.5.2 IDLEモードの解除

IDLEモードは、INTP0-INTP2入力または、 $\overline{\text{RESET}}$ 入力により解除されます。

表15 - 7に、解除ソースと解除後の動作の概要を示します。また、図15 - 7に、IDLEモード解除後の動作を示します。

表15 - 7 IDLEモードの解除と解除後の動作

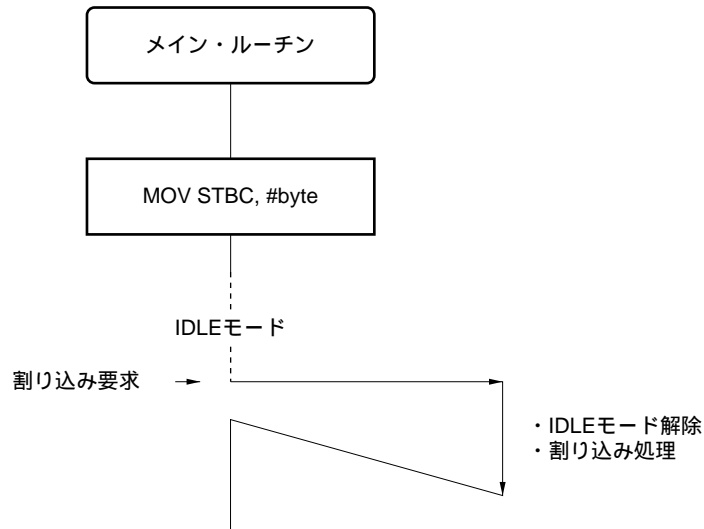
解除ソース	MK <sup>注1</sup>	ISM <sup>注2</sup>	IE <sup>注3</sup>	解除時の状態	解除後の動作
$\overline{\text{RESET}}$ 入力	x	x	x	-	通常のリセット動作
INTP0-INTP2 端子入力	0	0	1	<ul style="list-style-type: none"> <li>・割り込みサービス・プログラムの実行中でない</li> <li>・優先順位の低いマスク割り込みサービス・プログラムの実行中</li> <li>・優先順位レベル3の割り込みサービス・プログラムの実行中で、PRSLビット<sup>注5</sup>がクリア(0)されている</li> </ul>	割り込み要求の受け付け  MOV STBC, byte命令の次の命令を実行(IDLEモードを解除した割り込み要求は、保留される <sup>注4</sup> )
	0	0	0	-	
	1	0	x	-	IDLEモード保持
	x	1	x	-	

注1 . 各割り込み要求ソースにある割り込みマスク・ビット

- 2 . 各割り込み要求ソースにあるマクロ・サービス許可フラグ
- 3 . プログラム・ステータス・ワード (PSW) 中にある割り込み許可フラグ
- 4 . 保留された割り込み要求は、受け付け可能になった時点で受け付けられる
- 5 . 割り込みモード・コントロール・レジスタ (IMC) 中のビット

図15 - 7 IDLEモード解除後の動作 (1/3)

(1) IDLEモードに入ったあと、割り込みが入った場合



(2) IDLEモードに入ったあとリセット

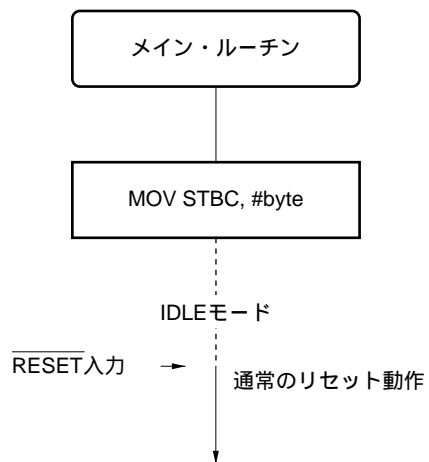
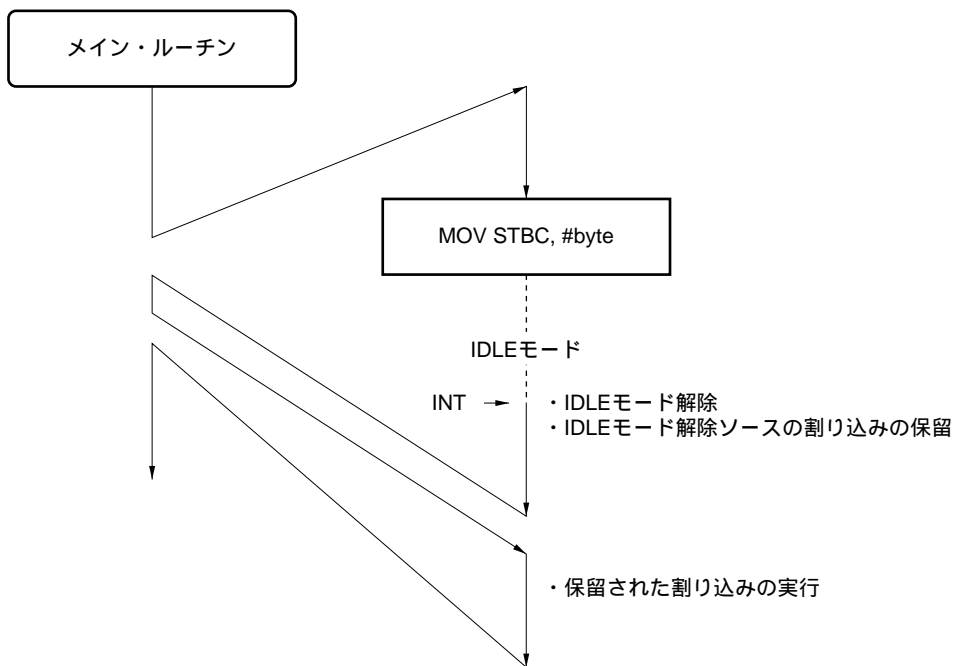


図15 - 7 IDLEモード解除後の動作 (2/3)

(3) 解除ソースの割り込みよりも優先順位の高い、あるいは同じ割り込み処理ルーチン中にIDLEモードに入った場合



(4) 解除ソースの割り込みよりも優先順位の高い割り込み処理ルーチン中にIDLEモードに入った場合

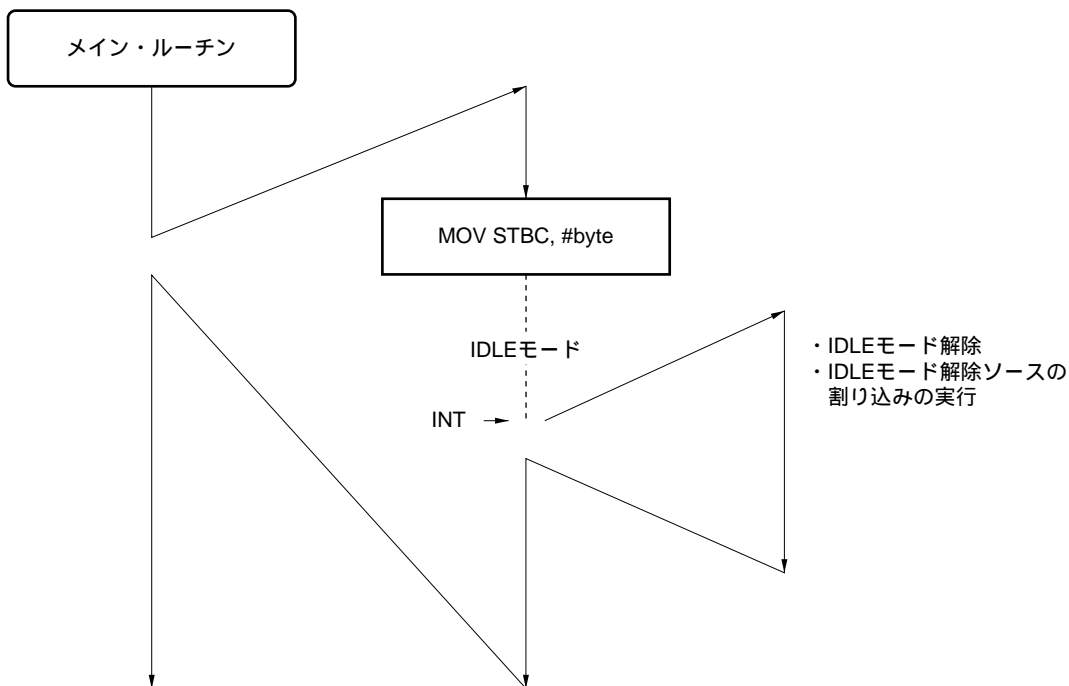
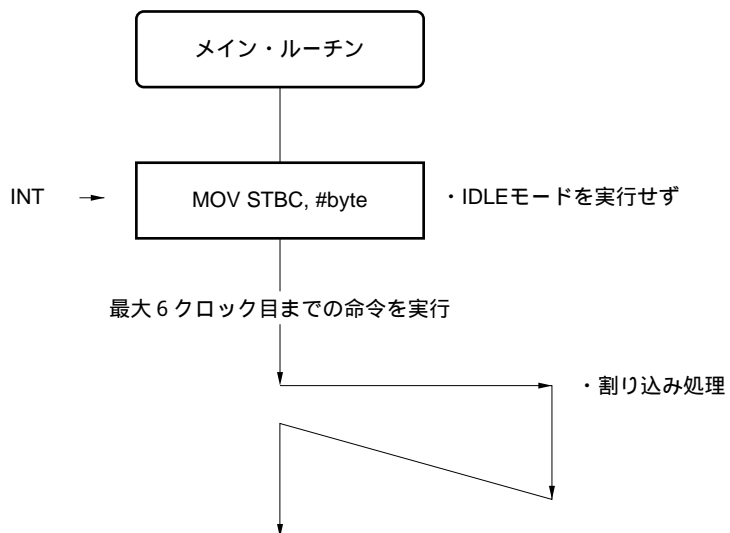


図15 - 7 IDLEモード解除後の動作 (3/3)

(5) IDLEモード設定命令と割り込みが競合する場合





**(1) INTP0-INTP2入力によるIDLEモードの解除**

INTP0-INTP2入力による割り込みのマスクが解除されていて、マクロ・サービスが禁止されている場合に、INTP0-INTP2入力に外部割り込みモード・レジスタ1 (INTM1) で指定した有効エッジが入力されると、IDLEモードは解除されます。

IDLEモードから解除されると、割り込み許可フラグ (IE) がセット (1) されている場合は、その割り込みが受け付け可能であれば、割り込みサービス・プログラムに分岐します。受け付けが不可能な場合とIEフラグがクリア (0) されている場合は、IDLEモードを設定した命令の次の命令から実行を再開します。

割り込みの受け付けについての詳細は、14.7 **マスカブル割り込みの受け付け動作**を参照してください。

**(2)  $\overline{\text{RESET}}$ 入力によるIDLEモードの解除**

$\overline{\text{RESET}}$ 入力をハイからロウに下げるとリセット状態になります。 $\overline{\text{RESET}}$ の立ち上がりでクロックの発振を開始し、発振安定用タイマがオーバフローすると通常動作をスタートします。このとき、内蔵データ・メモリはIDLEモード設定前の内容を保持しています。

## 15.6 STOPモード/IDLEモード使用時のチェック項目

STOPモード/IDLEモード使用時の消費電流を下げるために必要なチェック項目を示します。

### (1) 各出力端子の出力レベルは適正か？

各端子の適正な出力レベルは、次段の回路によって異なります。最も消費電流が少なくなるような出力レベルを選んでください。

- ・次段の回路の入カインピーダンスが低い場合にハイ・レベルを出力していると、電源からポートへ電流が流れ出し、消費電流が増えてしまいます。次段の回路がCMOS ICなどの場合がこれにあたります。CMOS ICは、電源OFF時には、入カインピーダンスが低くなります。消費電流を抑えるため、またCMOS ICの信頼性に悪影響を与えないため、ロウ・レベルを出力してください。ハイ・レベルを出力していると、電源の再投入時にラッチアップの原因となります。
- ・次段の回路によっては、ロウ・レベルを入力すると消費電流が増えてしまう場合があります。このような場合は、ハイ・レベルまたはハイ・インピーダンスを出力し、消費電流が少なくなるようにしてください。
- ・次段の回路がCMOS ICの場合で、そのCMOS ICに電源が供給されているときに出力をハイ・インピーダンスにすると、CMOS ICの消費電流が増える場合があります（このとき、CMOS ICが過熱し、破壊する場合があります）。このような場合には、適正なレベルを出力するか、抵抗によりプルアップまたはプルダウンするなどしてください。

出力レベルの設定方法は、ポートのモードによって異なります。

- ・ポートがコントロール・モードのときには、内蔵ハードウェアの状態によって出力レベルが決定されますので、内蔵ハードウェアの状態を考慮して出力レベルを設定する必要があります。
- ・ポート・モードのときには、ソフトウェアでポートの出力ラッチとポート・モード・レジスタへ書き込みをすることにより、出力レベルを設定することができます。

ポートがコントロール・モードになっているときには、ポート・モードへ変更することで、出力レベルの設定が簡単になります。

**(2) 各入力端子の入力レベルは適正か？**

各端子に入力される電圧レベルは、 $V_{SS}$ 電位から $V_{DD}$ 電位の範囲内にしてください。この範囲外の電圧を印加すると、消費電流が増えるだけでなく、 $\mu$ PD784975の信頼性にも悪影響を与えます。

また、中間電位が加わらないようにしてください。

**(3) 内蔵プルアップ抵抗は必要か？**

不要なプルアップ抵抗は、消費電流を増やしたり、他のデバイスのラッチアップの原因となったりします。プルアップ抵抗は、必要な部分だけを使用するモードに指定してください。

必要な部分と不要な部分が混在する場合は、必要な部分については外付けにプルアップ抵抗を接続し、内蔵プルアップ抵抗を使用しないモードに指定するなどしてください。

**(4) A/Dコンバータ**

A/Dコンバータ・モード・レジスタ (ADM) のビット7にあるCSビットをクリア (0) することにより、 $AV_{DD}$ 、 $AV_{REF1}$ 端子に流れる電流を減らすことが可能です。さらに、電流を減らしたい場合には、 $AV_{REF1}$ 端子への電流供給を外付け回路で切断してください。

なお、 $AV_{DD}$ 端子は必ず $V_{DD}$ 端子と同電位にしてください。STOPモード時に $AV_{DD}$ 端子に電源が供給されていないと、消費電流が増えるだけでなく、信頼性にも悪影響を与えます。

## 15.7 注意事項

- (1) HALTモードを解除する条件(15.3.2 HALTモードの解除参照)が成立しているときにHALTモード/STOPモード/IDLEモード(以下スタンバイ・モード)の設定を行った場合、スタンバイ・モードには入らずに次の命令の実行、またはベクタ割り込みサービス・プログラムへの分岐を行います。確実にスタンバイ・モードの設定をしたい場合は、スタンバイ・モードに入る前に割り込み要求をクリアするなどしてください。
- (2) クリスタル/セラミック発振使用時には、EXTCビットを必ずクリア(0)して使用してください。EXTCビットをセット(1)すると、発振が停止してしまいます。
- (3) 外部クロックを使用するシステムでSTOPモードを使用する場合は、OSTSのEXTCビットを必ずセット(1)してください。OSTSのEXTCビットがクリア(0)されているときに、外部クロックを入力するシステムでSTOPモードの設定を行うと、消費電流が増加します。  
なお、OSTSのEXTCビットをセット(1)する場合には、必ずX2端子に、X1端子に入力しているクロックの逆相のクロックを入力する必要があります(5.4 システム・クロック発振回路参照)。
- (4) STOPモード/IDLEモード設定時には、A/Dコンバータ・モード・レジスタ(ADM)のCSビットをクリア(0)してください。
- (5) スタンバイ命令のあと(スタンバイ解除後)は、NOP命令を3回実行させてください。そうしないとスタンバイ命令の実行と割り込み要求が競合した場合に、スタンバイ命令を実行せず、スタンバイ命令に続く複数の命令を実行後に割り込みを受け付けます。割り込み受け付け前に実行する命令は、スタンバイ命令実行後の最大6クロック以内に実行を開始する命令です。

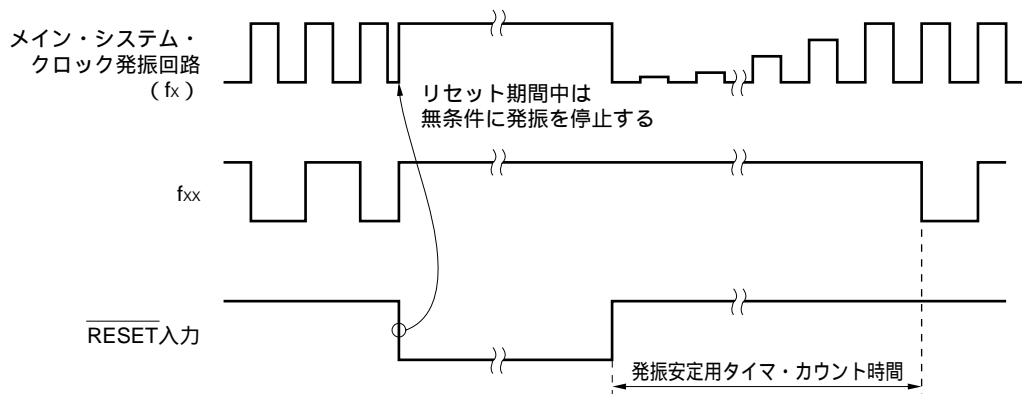
例 MOV STBC, byte  
NOP  
NOP  
NOP  
⋮

## 第16章 リセット機能

$\overline{\text{RESET}}$ 入力端子にロウ・レベルが入力されると、システム・リセットがかかり、各ハードウェアは、表16 - 1 に示すような状態になります。また、リセット期間中は、メイン・システム・クロックの発振を無条件に停止しますので、システム全体の消費電流を抑えることができます。

$\overline{\text{RESET}}$ 入力がロウ・レベルからハイ・レベルになると、リセット状態が解除され、発振安定用タイマのカウント時間（41.9 ms : 12.5 MHz動作時）のあと、リセット・ベクタ・テーブルの内容がプログラム・カウンタ（PC）にセットされ、PCにセットされたアドレスに分岐し、その分岐先のアドレスからプログラムの実行を開始します。したがって、任意のアドレスからリセット・スタートできます。

図16 - 1 リセット期間中のメイン・システム・クロックの発振



$\overline{\text{RESET}}$ 入力端子は、ノイズによる誤動作を防ぐため、アナログ・ディレイによるノイズ除去回路を内蔵しています。

図16 - 2 リセット信号の受け付け

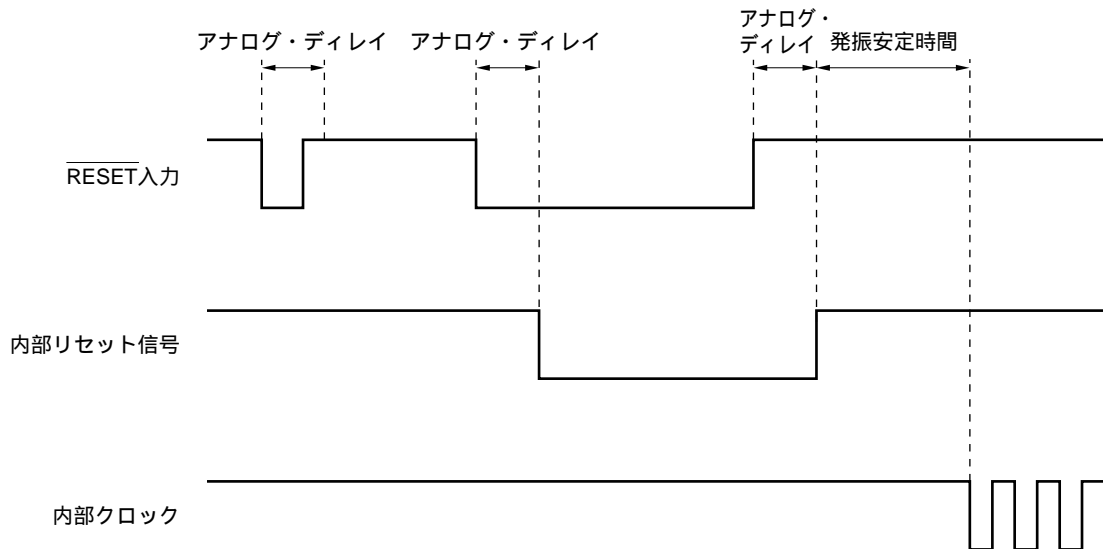


表16 - 1 各ハードウェアのリセット中，リセット後の状態

ハードウェア	リセット中の状態 ( $\overline{\text{RESET}} = \text{L}$ )	リセット後の状態 ( $\overline{\text{RESET}} = \text{H}$ )
メイン・システム・クロック発振回路	発振を停止する	発振を開始する
プログラム・カウンタ (PC)	不 定	リセット・ベクタ・テーブルの値をセット
スタック・ポインタ (SP)	不 定	
プログラム・ステータス・ワード (PSW)	0000Hにイニシャライズ	
内蔵RAM	不定。ただし，スタンバイ状態をリセットで解除した場合には，スタンバイ設定前の値を保持する。	
I/Oライン	入力バッファ，出力バッファをOFFにする。	ハイ・インピーダンス
その他のハードウェア	所定の状態に初期化する <sup>注</sup>	

注 表3 - 6 特殊機能レジスタ (SFR) 一覧のリセット時を参照。

## 第17章 $\mu$ PD78F4976のプログラミング

フラッシュ・メモリへの書き込みは、ターゲット・システムに実装した状態（オンボード）で行うことができます。専用フラッシュ・ライター（Flashpro（型番FL-PR3, PG-FP3））をホスト・マシンおよびターゲット・システムに接続して書き込みます。

**備考** FL-PR3は、株式会社内藤電誠町田製作所の製品です。

### 17.1 通信方式の選択

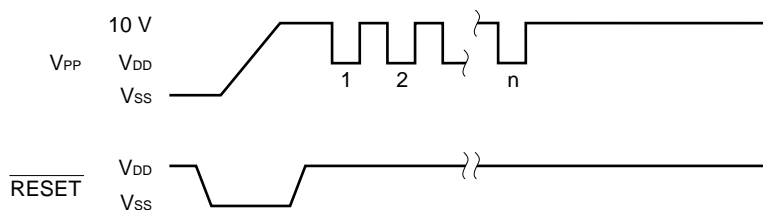
フラッシュ・メモリへの書き込みは、Flashpro を使用し、シリアル通信で行います。表17 - 1 に示す通信方式から選択して書き込みを行います。この通信方式の選択は、図17 - 1 に示すようなフォーマットを用います。表17 - 1 に示すV<sub>PP</sub>パルス数で、それぞれの通信方式が選択されます。

表17 - 1 通信方式一覧

通信方式	チャンネル数	使用端子	V <sub>PP</sub> パルス数
3線式シリアルI/O	2	$\overline{\text{SCK0}}$ /P27 SO0/P26 SI0/P25	0
		$\overline{\text{SCK1}}$ /P62 SO1/P61 SI1/P60	1

**注意** 通信方式は、必ず表17 - 1 に示すV<sub>PP</sub>パルス数で選択してください。

図17 - 1 通信方式選択フォーマット



## 17.2 フラッシュ・メモリ・プログラミングの機能

選択された通信方式による各種コマンド/データ送受信により、フラッシュ・メモリの書き込みなどの動作を行います。主な機能を表17 - 2 に示します。

表17 - 2 フラッシュ・メモリ・プログラミングの主な機能

機 能	説 明
一括消去	全メモリの内容を消去します。
ブロック消去	16 Kバイトを1メモリ・ブロックとし、指定したメモリ・ブロックの内容を消去します。
一括ブランク・チェック	全メモリの消去状態を確認します。
ブロック・ブランク・チェック	指定したブロックの消去状態を確認します。
データ・ライト	書き込み開始アドレスおよび書き込みデータ数(バイト数)をもとに、フラッシュ・メモリに書き込みを行います。
一括ベリファイ	全メモリの内容と入力したデータを比較します。
ブロック・ベリファイ	指定したメモリ・ブロックの内容と入力したデータを比較します。

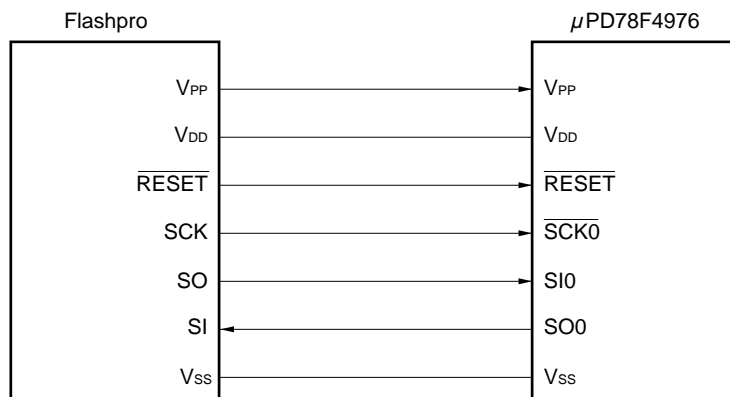
なお、フラッシュ・メモリのベリファイは、ベリファイを行うデータをシリアル・インタフェースを介して外部から供給し、ブロックまたは全データの照合が終了したあとに不一致データの有無を外部へ出力するようになっています。したがって、フラッシュ・メモリの読み出し機能はなく、このベリファイ方式によって、フラッシュ・メモリの内容を他者に読まれることはありません。



### 17.3 Flashpro の接続

Flashpro と  $\mu$ PD78F4976との接続は、通信方式（3線式シリアルI/O）によって異なります。接続図を図17 - 2 に示します。

図17 - 2 3線式シリアルI/O方式でのFlashpro の接続（3線式シリアルI/O 0を使用する場合）



〔メモ〕

# 第18章 命令のオペレーション

## 18.1 凡 例

### (1) オペランドの表現形式と記述方法 (1/2)

表現形式	記 述 方 法
r, r' <sup>注1</sup>	X(R0), A(R1), C(R2), B(R3), R4, R5, R6, R7, R8, R9, R10, R11, E(R12), D(R13), L(R14), H(R15)
r1 <sup>注1</sup>	X(R0), A(R1), C(R2), B(R3), R4, R5, R6, R7
r2	R8, R9, R10, R11, E(R12), D(R13), L(R14), H(R15)
r3	V, U, T, W
rp, rp' <sup>注2</sup>	AX(RP0), BC(RP1), RP2, RP3, VR(RP4), UR(RP5), DE(RP6), HL(RP7)
rp1 <sup>注2</sup>	AX(RP0), BC(RP1), RP2, RP3
rp2	VR(RP4), UR(RP5), DE(RP6), HL(RP7)
rg, rg'	VVR(RG4), UUR(RG5), TDE(RG6), WHL(RG7)
sfr	特殊機能レジスタ略号 (表3-6 特殊機能レジスタ(SFR)一覽参照)
sfrp	特殊機能レジスタ略号 (16ビット操作可能レジスタ: 表3-6 特殊機能レジスタ(SFR)一覽参照)
post <sup>注2</sup>	AX(RP0), BC(RP1), RP2, RP3, VR(RP4), UR(RP5)PSW, DE(RP6), HL(RP7) 複数記述可能。ただし, UPIはPUSH/POP命令, PSWIはPUSHU/POPU命令にかぎる
mem	[TDE][WHL][TDE+][WHL+][TDE-][WHL-][VVP][UUP]: レジスタ・インダイ レクト・アドレッシング [TDE+byte][WHL+byte][SP+byte][UUP+byte][VVP+byte]: ベースト・アドレッシング imm24[A] imm24[B] imm24[DE] imm24[HL]: インデクスト・アドレッシング [TDE+A][TDE+B][TDE+C][WHL+A][WHL+B][WHL+C][VVP+DE] [VVP+HL]: ベースト・インデクスト・アドレッシング
mem1	memから[WHL+][WHL-]を除いたすべて
mem2	[TDE][WHL]
mem3	[AX][BC][RP2][RP3][VVP][UUP][TDE][WHL]

注1 . RSSビットを1とすることで, R4-R7をX, A, C, Bとして使用することができますが, この機能を使用するのは, 78K/IIIシリーズ用のプログラムを流用する場合だけにしてください。

2 . RSSビットを1とすることで, RP2, RP3をAX, BCとして使用することができますが, この機能を使用するのは, 78K/IIIシリーズ用のプログラムを流用する場合だけにしてください。

(1) オペランドの表現形式と記述方法 (2/2)

表現形式	記述方法
注	
saddr, saddr'	FD20H-FF1FH イミーディエト・データまたはレーベル
saddr1	FE00H-FEFFFH イミーディエト・データまたはレーベル
saddr2	FD20H-FDFFFH, FF00H-FF1FH イミーディエト・データまたはレーベル
saddrp	FD20H-FF1EH イミーディエト・データまたはレーベル (16ビット操作時)
saddrp1	FE00H-FEFFFH イミーディエト・データまたはレーベル (16ビット操作時)
saddrp2	FD20H-FDFFFH, FF00H-FF1EH イミーディエト・データまたはレーベル (16ビット操作時)
saddrg	FD20H-FEFDH イミーディエト・データまたはレーベル (24ビット操作時)
saddrg1	FE00H-FEFDH イミーディエト・データまたはレーベル (24ビット操作時)
saddrg2	FD20H-FDFFFH イミーディエト・データまたはレーベル (24ビット操作時)
addr24	0H-FFFFFFH イミーディエト・データまたはレーベル
addr20	0H-FFFFFFH イミーディエト・データまたはレーベル
addr16	0H-FFFFH イミーディエト・データまたはレーベル
addr11	800H-FFFH イミーディエト・データまたはレーベル
addr8	0FE00H-0FEFFF <sup>注</sup> イミーディエト・データまたはレーベル
addr5	40H-7EH イミーディエト・データまたはレーベル
imm24	24ビット・イミーディエト・データまたはレーベル
word	16ビット・イミーディエト・データまたはレーベル
byte	8ビット・イミーディエト・データまたはレーベル
bit	3ビット・イミーディエト・データまたはレーベル
n	3ビット・イミーディエト・データ
locaddr	00Hまたは0FH

注 LOCATION命令で00Hを指定した場合は、ここに示したアドレスになります。

LOCATION命令で0FHを指定した場合は、ここに示したアドレスにF0000Hを加えた値がアドレスになります。

(2) オペランド欄の記号

記号	説明
+	オートインクリメント
-	オートデクリメント
	イミディエイト・データ
!	16ビット絶対アドレス
!!	24ビット/20ビット絶対アドレス
\$	8ビット相対アドレス
\$!	16ビット相対アドレス
/	ビット反転
[ ]	インダイレクト・アドレッシング
[ % ]	24ビット・インダイレクト・アドレッシング

(3) フラグ欄の記号

記号	説明
(ブランク)	変化なし
0	0にクリアされる
1	1にセットされる
x	結果に従ってセット/クリアされる
P	P/Vフラグがパリティ・フラグとして動作する
V	P/Vフラグがオーバフロー・フラグとして動作する
R	以前に退避した値がリストアされる

(4) オペレーション欄の記号

記号	説明
jdisp8	次の命令の先頭アドレスと分岐先アドレスとの相対アドレス距離の符号付き2の補数データ (8ビット)
jdisp16	次の命令の先頭アドレスと分岐先アドレスとの相対アドレス距離の符号付き2の補数データ (16ビット)
PC <sub>HW</sub>	PCのビット16-19
PC <sub>LW</sub>	PCのビット0-15

(5) オペランドにmemを含む命令のバイト数

memのモード	レジスタ・インダイレクト・アドレッシング		ベースト・アドレッシング	インデクスト・アドレッシング	ベースト・インデクスト・アドレッシング
バイト数	1	2注	3	5	2

注 MOV命令でmemに [ TDE ] [ WHL ] [ TDE + ] [ TDE - ] [ WHL + ] [ WHL - ] を記述した場合のみ、1バイト命令になります。

(6) オペランドにsaddr, saddrp, r, rpを含む命令のバイト数

オペランドにsaddr, saddrp, r, rpを含む命令の中には、バイト数をスラッシュ “ / ” で2つに分けて記述しているものがあります。どちらのバイト数になるかは下表によります。

表現形式	バイト数の左側	バイト数の右側
saddr	saddr2	saddr1
saddrp	saddrp2	saddrp1
r	r1	r2
rp	rp1	rp2

(7) オペランドにmemを含む命令、ストリング命令の記述

オペランドのTDE, WHL, VVP, UUP (24ビット・レジスタ) は、それぞれDE, HL, VP, UPと記述することもできます。ただし、DE, HL, VP, UPと記述した場合でもTDE, WHL, VVP, UUP (24ビット・レジスタ) として扱われます。

## 18.2 オペレーション一覧

### (1) 8ビット・データ転送命令：MOV

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
MOV	r, byte	2/3	r byte					
	saddr, byte	3/4	(saddr) byte					
	sfr, byte	3	sfr byte					
	!addr16, byte	5	(saddr16) byte					
	!!addr24, byte	6	(addr24) byte					
	r, r	2/3	r r					
	A, r	1/2	A r					
	A, saddr2	2	A (saddr2)					
	r, saddr	3	r (saddr)					
	saddr2, A	2	(saddr2) A					
	saddr, r	3	(saddr) r					
	A, sfr	2	A sfr					
	r, sfr	3	r sfr					
	sfr, A	2	sfr A					
	sfr, r	3	sfr r					
	saddr, saddr'	4	(saddr) (saddr')					
	r, !addr16	4	r (addr16)					
	!addr16, r	4	(addr16) r					
	r, !!addr24	5	r (addr24)					
	!!addr24, r	5	(addr24) r					
	A, [saddrp]	2/3	A ((saddrp))					
	A, [%saddrg]	3/4	A ((saddrg))					
	A, mem	1-5	A (mem)					
	[saddrp] A	2/3	((saddrp)) A					
	[%saddrg] A	3/4	((saddrg)) A					
	mem, A	1-5	(mem) A					
	PSWL, byte	3	PSW <sub>L</sub> byte				x	x
	PSWH, byte	3	PSW <sub>H</sub> byte					
	PSWL, A	2	PSW <sub>L</sub> A				x	x
	PSWH, A	2	PSW <sub>H</sub> A					
	A, PSWL	2	A PSW <sub>L</sub>					
	A, PSWH	2	A PSW <sub>H</sub>					
	r3, byte	3	r3 byte					
A, r3	2	A r3						
r3, A	2	r3 A						

(2) 16ビット・データ転送命令 : MOVW

二モニック	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
MOVW	rp, word	3	rp word						
	saddrp, word	4/5	(saddrp) word						
	sfrp, word	4	sfrp word						
	!addr16, word	6	(addr16) word						
	!!addr24, word	7	(addr24) word						
	rp, rp'	2	rp rp'						
	AX, saddrp2	2	AX (saddrp2)						
	rp, saddrp	3	rp (saddrp)						
	saddrp2, AX	2	(saddrp2) AX						
	saddrp, rp	3	(saddrp) rp						
	AX, sfrp	2	AX sfrp						
	rp, sfrp	3	rp sfrp						
	sfrp, AX	2	sfrp AX						
	sfrp, rp	3	sfrp rp						
	saddrp, saddrp'	4	(saddrp) (saddrp')						
	rp, !addr16	4	rp (addr16)						
	!addr16, rp	4	(addr16) rp						
	rp, !!addr24	5	rp (addr24)						
	!!addr24, rp	5	(addr24) rp						
	AX, [saddrp]	3/4	AX ((saddrp))						
	AX, [%saddrg]	3/4	AX ((saddrg))						
	AX, mem	2-5	AX (mem)						
	[saddrp] AX	3/4	((saddrp)) AX						
[%saddrg] AX	3/4	((saddrg)) AX							
mem, AX	2-5	(mem) AX							



(3) 24ビット・データ転送命令：MOVG

二モニック	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
MOVG	rg, imm24	5	rg imm24						
	rg, rg'	2	rg rg'						
	rg, !addr24	5	rg (addr24)						
	!addr24, rg	5	(addr24) rg						
	rg, saddrg	3	rg (saddrg)						
	saddrg, rg	3	(saddrg) rg						
	WHL, [%saddrg]	3/4	WHL ((saddrg))						
	[%saddrg] WHL	3/4	((saddrg) WHL)						
	WHL, mem1	2-5	WHL (mem1)						
	mem1, WHL	2-5	(mem1) WHL						

(4) 8ビット・データ交換命令：XCH

二モニック	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
XCH	r, r'	2/3	r r'						
	A, r	1/2	A r						
	A, saddr2	2	A (saddr2)						
	r, saddr	3	r (saddr)						
	r, sfr	3	r sfr						
	saddr, saddr'	4	(saddr) (saddr')						
	r, !addr16	4	r (addr16)						
	r, !addr24	5	r (addr24)						
	A, [saddrp]	2/3	A ((saddrp))						
	A, [%saddrg]	3/4	A ((saddrg))						
	A, mem	2-5	A (mem)						

(5) 16ビット・データ交換命令 : XCHW

二モニック	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
XCHW	rp, rp'	2	rp rp'						
	AX, saddrp2	2	AX (saddrp2)						
	rp, saddrp	3	rp (saddrp)						
	rp, sfrp	3	rp sfrp						
	AX, [saddrp]	3/4	AX ((saddrp))						
	AX, [%saddrg]	3/4	AX ((saddrg))						
	AX, !addr16	4	AX (addr16)						
	AX, !!addr24	5	AX (addr24)						
	saddrp, saddrp'	4	(saddrp) (saddrp')						
	AX, mem	2-5	AX (mem)						

(6) 8ビット演算命令 : ADD, ADDC, SUB, SUBC, CMP, AND, OR, XOR

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
ADD	A, byte	2	A, CY A + byte	x	x	x	V	x
	r, byte	3	r, CY r + byte	x	x	x	V	x
	saddr, byte	3/4	(saddr), CY (saddr) + byte	x	x	x	V	x
	sfr, byte	4	sfr, CY sfr + byte	x	x	x	V	x
	r, r'	2/3	r, CY r + r'	x	x	x	V	x
	A, saddr2	2	A, CY A + (saddr2)	x	x	x	V	x
	r, saddr	3	r, CY r + (saddr)	x	x	x	V	x
	saddr, r	3	(saddr), CY (saddr) + r	x	x	x	V	x
	r, sfr	3	r, CY r + sfr	x	x	x	V	x
	sfr, r	3	sfr, CY sfr + r	x	x	x	V	x
	saddr, saddr'	4	(saddr), CY (saddr) + (saddr')	x	x	x	V	x
	A, [saddrp]	3/4	A, CY A + ((saddrp))	x	x	x	V	x
	A, [%saddrg]	3/4	A, CY A + ((saddrg))	x	x	x	V	x
	[saddrp] A	3/4	((saddrp)), CY ((saddrp)) + A	x	x	x	V	x
	[%saddrg] A	3/4	((saddrg)), CY ((saddrg)) + A	x	x	x	V	x
	A, !addr16	4	A, CY A + (addr16)	x	x	x	V	x
	A, !!addr24	5	A, CY A + (addr24)	x	x	x	V	x
	!addr16, A	4	(addr16), CY (addr16) + A	x	x	x	V	x
	!!addr24, A	5	(addr24), CY (addr24) + A	x	x	x	V	x
	A, mem	2-5	A, CY A + (mem)	x	x	x	V	x
mem, A	2-5	(mem), CY (mem) + A	x	x	x	V	x	

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
ADDC	A, byte	2	A, CY A + byte + CY	x	x	x	V	x
	r, byte	3	r, CY r + byte + CY	x	x	x	V	x
	saddr, byte	3/4	( saddr ), CY ( saddr ) + byte + CY	x	x	x	V	x
	sfr, byte	4	sfr, CY sfr + byte + CY	x	x	x	V	x
	r, r'	2/3	r, CY r + r' + CY	x	x	x	V	x
	A, saddr2	2	A, CY A + ( saddr2 ) + CY	x	x	x	V	x
	r, saddr	3	r, CY r + ( saddr ) + CY	x	x	x	V	x
	saddr, r	3	( saddr ), CY ( saddr ) + r + CY	x	x	x	V	x
	r, sfr	3	r, CY r + sfr + CY	x	x	x	V	x
	sfr, r	3	sfr, CY sfr + r + CY	x	x	x	V	x
	saddr, saddr'	4	( saddr ), CY ( saddr ) + ( saddr' ) + CY	x	x	x	V	x
	A, [ saddrp ]	3/4	A, CY A + (( saddrp )) + CY	x	x	x	V	x
	A, [ %saddrg ]	3/4	A, CY A + (( saddrg )) + CY	x	x	x	V	x
	[ saddrp ] A	3/4	(( saddrp )), CY (( saddrp )) + A + CY	x	x	x	V	x
	[ %saddrg ] A	3/4	(( saddrg )), CY (( saddrg )) + A + CY	x	x	x	V	x
	A, !addr16	4	A, CY A + ( addr16 ) + CY	x	x	x	V	x
	A, !!addr24	5	A, CY A + ( addr24 ) + CY	x	x	x	V	x
	!addr16, A	4	( addr16 ), CY ( addr16 ) + A + CY	x	x	x	V	x
	!!addr24, A	5	( addr24 ), CY ( addr24 ) + A + CY	x	x	x	V	x
	A, mem	2-5	A, CY A + ( mem ) + CY	x	x	x	V	x
mem, A	2-5	( mem ), CY ( mem ) + A + CY	x	x	x	V	x	

ニモニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
SUB	A, byte	2	A, CY A - byte	x	x	x	V	x
	r, byte	3	r, CY r - byte	x	x	x	V	x
	saddr, byte	3/4	( saddr ), CY ( saddr ) - byte	x	x	x	V	x
	sfr, byte	4	sfr, CY sfr - byte	x	x	x	V	x
	r, r'	2/3	r, CY r - r'	x	x	x	V	x
	A, saddr2	2	A, CY A - ( saddr2 )	x	x	x	V	x
	r, saddr	3	r, CY r - ( saddr )	x	x	x	V	x
	saddr, r	3	( saddr ), CY ( saddr ) - r	x	x	x	V	x
	r, sfr	3	r, CY r - sfr	x	x	x	V	x
	sfr, r	3	sfr, CY sfr - r	x	x	x	V	x
	saddr, saddr'	4	( saddr ), CY ( saddr ) - ( saddr' )	x	x	x	V	x
	A, [ saddrp ]	3/4	A, CY A - (( saddrp ))	x	x	x	V	x
	A, [ %saddrg ]	3/4	A, CY A - (( saddrg ))	x	x	x	V	x
	[ saddrp ] A	3/4	(( saddrp )), CY (( saddrp )) - A	x	x	x	V	x
	[ %saddrg ] A	3/4	(( saddrg )), CY (( saddrg )) - A	x	x	x	V	x
	A, !addr16	4	A, CY A - ( addr16 )	x	x	x	V	x
	A, !!addr24	5	A, CY A - ( addr24 )	x	x	x	V	x
	!addr16, A	4	( addr16 ), CY ( addr16 ) - A	x	x	x	V	x
	!!addr24, A	5	( addr24 ), CY ( addr24 ) - A	x	x	x	V	x
	A, mem	2-5	A, CY A - ( mem )	x	x	x	V	x
mem, A	2-5	( mem ), CY ( mem ) - A	x	x	x	V	x	

ニモニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
SUBC	A, byte	2	A, CY A - byte - CY	x	x	x	V	x
	r, byte	3	r, CY r - byte - CY	x	x	x	V	x
	saddr, byte	3/4	( saddr ), CY ( saddr ) - byte - CY	x	x	x	V	x
	sfr, byte	4	sfr, CY sfr - byte - CY	x	x	x	V	x
	r, r'	2/3	r, CY r - r' - CY	x	x	x	V	x
	A, saddr2	2	A, CY A - ( saddr2 ) - CY	x	x	x	V	x
	r, saddr	3	r, CY r - ( saddr ) - CY	x	x	x	V	x
	saddr, r	3	( saddr ), CY ( saddr ) - r - CY	x	x	x	V	x
	r, sfr	3	r, CY r - sfr - CY	x	x	x	V	x
	sfr, r	3	sfr, CY sfr - r - CY	x	x	x	V	x
	saddr, saddr'	4	( saddr ), CY ( saddr ) - ( saddr' ) - CY	x	x	x	V	x
	A, [ saddrp ]	3/4	A, CY A - (( saddrp )) - CY	x	x	x	V	x
	A, [ %saddrg ]	3/4	A, CY A - (( saddrg )) - CY	x	x	x	V	x
	[ saddrp ] A	3/4	(( saddrp )), CY (( saddrp )) - A - CY	x	x	x	V	x
	[ %saddrg ] A	3/4	(( saddrg )), CY (( saddrg )) - A - CY	x	x	x	V	x
	A, !addr16	4	A, CY A - ( addr16 ) - CY	x	x	x	V	x
	A, !!addr24	5	A, CY A - ( addr24 ) - CY	x	x	x	V	x
	!addr16, A	4	( addr16 ), CY ( addr16 ) - A - CY	x	x	x	V	x
	!!addr24, A	5	( addr24 ), CY ( addr24 ) - A - CY	x	x	x	V	x
	A, mem	2-5	A, CY A - ( mem ) - CY	x	x	x	V	x
mem, A	2-5	( mem ), CY ( mem ) - A - CY	x	x	x	V	x	

ニモニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
CMP	A, byte	2	A - byte	x	x	x	V	x
	r, byte	3	r - byte	x	x	x	V	x
	saddr, byte	3/4	( saddr ) - byte	x	x	x	V	x
	sfr, byte	4	sfr - byte	x	x	x	V	x
	r, r'	2/3	r - r'	x	x	x	V	x
	A, saddr2	2	A - ( saddr2 )	x	x	x	V	x
	r, saddr	3	r - ( saddr )	x	x	x	V	x
	saddr, r	3	( saddr ) - r	x	x	x	V	x
	r, sfr	3	r - sfr	x	x	x	V	x
	sfr, r	3	sfr - r	x	x	x	V	x
	saddr, saddr'	4	( saddr ) - ( saddr' )	x	x	x	V	x
	A, [ saddrp ]	3/4	A - (( saddrp ))	x	x	x	V	x
	A, [ %saddrg ]	3/4	A - (( saddrg ))	x	x	x	V	x
	[ saddrp ] A	3/4	(( saddrp )) - A	x	x	x	V	x
	[ %saddrg ] A	3/4	(( saddrg )) - A	x	x	x	V	x
	A, !addr16	4	A - ( addr16 )	x	x	x	V	x
	A, !!addr24	5	A - ( addr24 )	x	x	x	V	x
	!addr16, A	4	( addr16 ) - A	x	x	x	V	x
	!!addr24, A	5	( addr24 ) - A	x	x	x	V	x
	A, mem	2-5	A - ( mem )	x	x	x	V	x
mem, A	2-5	( mem ) - A	x	x	x	V	x	

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
AND	A, byte	2	A A byte	x	x			P
	r, byte	3	r r byte	x	x			P
	saddr, byte	3/4	( saddr ) ( saddr ) byte	x	x			P
	sfr, byte	4	sfr sfr byte	x	x			P
	r, r'	2/3	r r r'	x	x			P
	A, saddr2	2	A A ( saddr2 )	x	x			P
	r, saddr	3	r r ( saddr )	x	x			P
	saddr, r	3	( saddr ) ( saddr ) r	x	x			P
	r, sfr	3	r r sfr	x	x			P
	sfr, r	3	sfr sfr r	x	x			P
	saddr, saddr'	4	( saddr ) ( saddr ) ( saddr' )	x	x			P
	A, [ saddrp ]	3/4	A A (( saddrp ))	x	x			P
	A, [ %saddrg ]	3/4	A A (( saddrg ))	x	x			P
	[ saddrp ] A	3/4	(( saddrp )) (( saddrp )) A	x	x			P
	[ %saddrg ] A	3/4	(( saddrg )) (( saddrg )) A	x	x			P
	A, !addr16	4	A A ( addr16 )	x	x			P
	A, !!addr24	5	A A ( addr24 )	x	x			P
	!addr16, A	4	( addr16 ) ( addr16 ) A	x	x			P
	!!addr24, A	5	( addr24 ) ( addr24 ) A	x	x			P
	A, mem	2-5	A A ( mem )	x	x			P
mem, A	2-5	( mem ) ( mem ) A	x	x			P	

ニモニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
OR	A, byte	2	A A byte	x	x		P	
	r, byte	3	r r byte	x	x		P	
	saddr, byte	3/4	(saddr) (saddr) byte	x	x		P	
	sfr, byte	4	sfr sfr byte	x	x		P	
	r, r'	2/3	r r r'	x	x		P	
	A, saddr2	2	A A (saddr2)	x	x		P	
	r, saddr	3	r r (saddr)	x	x		P	
	saddr, r	3	(saddr) (saddr) r	x	x		P	
	r, sfr	3	r r sfr	x	x		P	
	sfr, r	3	sfr sfr r	x	x		P	
	saddr, saddr'	4	(saddr) (saddr) (saddr')	x	x		P	
	A, [ saddrp ]	3/4	A A ((saddrp))	x	x		P	
	A, [ %saddrg ]	3/4	A A ((saddrg))	x	x		P	
	[ saddrp ] A	3/4	((saddrp)) ((saddrp)) A	x	x		P	
	[ %saddrg ] A	3/4	((saddrg)) ((saddrg)) A	x	x		P	
	A, !addr16	4	A A (saddr16)	x	x		P	
	A, !!addr24	5	A A (saddr24)	x	x		P	
	!addr16, A	4	(addr16) (addr16) A	x	x		P	
	!!addr24, A	5	(addr24) (addr24) A	x	x		P	
	A, mem	2-5	A A (mem)	x	x		P	
mem, A	2-5	(mem) (mem) A	x	x		P		



二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
XOR	A, byte	2	A A ∨ byte	x	x			P
	r, byte	3	r r ∨ byte	x	x			P
	saddr, byte	3/4	( saddr ) ( saddr ) ∨ byte	x	x			P
	sfr, byte	4	sfr sfr ∨ byte	x	x			P
	r, r'	2/3	r r ∨ r'	x	x			P
	A, saddr2	2	A A ∨ ( saddr2 )	x	x			P
	r, saddr	3	r r ∨ ( saddr )	x	x			P
	saddr, r	3	( saddr ) ( saddr ) ∨ r	x	x			P
	r, sfr	3	r r ∨ sfr	x	x			P
	sfr, r	3	sfr sfr ∨ r	x	x			P
	saddr, saddr'	4	( saddr ) ( saddr ) ∨ ( saddr' )	x	x			P
	A, [ saddrp ]	3/4	A A ∨ (( saddrp ))	x	x			P
	A, [ %saddrg ]	3/4	A A ∨ (( saddrg ))	x	x			P
	[ saddrp ] A	3/4	(( saddrp )) (( saddrp )) ∨ A	x	x			P
	[ %saddrg ] A	3/4	(( saddrg )) (( saddrg )) ∨ A	x	x			P
	A, !addr16	4	A A ∨ ( addr16 )	x	x			P
	A, !!addr24	5	A A ∨ ( addr24 )	x	x			P
	!addr16, A	4	( addr16 ) ( addr16 ) ∨ A	x	x			P
	!!addr24, A	5	( addr24 ) ( addr24 ) ∨ A	x	x			P
	A, mem	2-5	A A ∨ ( mem )	x	x			P
mem, A	2-5	( mem ) ( mem ) ∨ A	x	x			P	

(7) 16ビット演算命令: ADDW, SUBW, CMPW

ニモニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
ADDW	AX, word	3	AX, CY AX + word	x	x	x	V	x
	rp, word	4	rp, CY rp + word	x	x	x	V	x
	rp, rp'	2	rp, CY rp + rp'	x	x	x	V	x
	AX, saddrp2	2	AX, CY AX + (saddrp2)	x	x	x	V	x
	rp, saddrp	3	rp, CY rp + (saddrp)	x	x	x	V	x
	saddrp, rp	3	(saddrp), CY (saddrp) + rp	x	x	x	V	x
	rp, sfrp	3	rp, CY rp + sfrp	x	x	x	V	x
	sfrp, rp	3	sfrp, CY sfrp + rp	x	x	x	V	x
	saddrp, word	4/5	(saddrp), CY (saddrp) + word	x	x	x	V	x
	sfrp, word	5	sfrp, CY sfrp + word	x	x	x	V	x
	saddrp, saddrp'	4	(saddrp), CY (saddrp) + (saddrp')	x	x	x	V	x
SUBW	AX, word	3	AX, CY AX - word	x	x	x	V	x
	rp, word	4	rp, CY rp - word	x	x	x	V	x
	rp, rp'	2	rp, CY rp - rp'	x	x	x	V	x
	AX, saddrp2	2	AX, CY AX - (saddrp2)	x	x	x	V	x
	rp, saddrp	3	rp, CY rp - (saddrp)	x	x	x	V	x
	saddrp, rp	3	(saddrp), CY (saddrp) - rp	x	x	x	V	x
	rp, sfrp	3	rp, CY rp - sfrp	x	x	x	V	x
	sfrp, rp	3	sfrp, CY sfrp - rp	x	x	x	V	x
	saddrp, word	4/5	(saddrp), CY (saddrp) - word	x	x	x	V	x
	sfrp, word	5	sfrp, CY sfrp - word	x	x	x	V	x
	saddrp, saddrp'	4	(saddrp), CY (saddrp) - (saddrp')	x	x	x	V	x
CMPW	AX, word	3	AX - word	x	x	x	V	x
	rp, word	4	rp - word	x	x	x	V	x
	rp, rp'	2	rp - rp'	x	x	x	V	x
	AX, saddrp2	2	AX - (saddrp2)	x	x	x	V	x
	rp, saddrp	3	rp - (saddrp)	x	x	x	V	x
	saddrp, rp	3	(saddrp) - rp	x	x	x	V	x
	rp, sfrp	3	rp - sfrp	x	x	x	V	x
	sfrp, rp	3	sfrp - rp	x	x	x	V	x
	saddrp, word	4/5	(saddrp) - word	x	x	x	V	x
	sfrp, word	5	sfrp - word	x	x	x	V	x
	saddrp, saddrp'	4	(saddrp) - (saddrp')	x	x	x	V	x

( 8 ) 24ビット演算命令 : ADDG, SUBG

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
ADDG	rg, rg'	2	rg, CY rg + rg'	x	x	x	V	x
	rg, imm24	5	rg, CY rg + imm24	x	x	x	V	x
	WHL, saddrg	3	WHL, CY WHL + ( saddrg )	x	x	x	V	x
SUBG	rg, rg'	2	rg, CY rg - rg'	x	x	x	V	x
	rg, imm24	5	rg, CY rg - imm24	x	x	x	V	x
	WHL, saddrg	3	WHL, CY WHL - ( saddrg )	x	x	x	V	x

( 9 ) 乗除算命令 : MULU, MULUW, MULW, DIVUW, DIVUX

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
MULU	r	2/3	AX A × r					
MULUW	rp	2	AX(上位), rp(下位) AX × rp					
MULW	rp	2	AX(上位), rp(下位) AX × rp					
DIVUW	r	2/3	AX(商), r(余り) AX ÷ r <sup>注1</sup>					
DIVUX	rp	2	AXDE(商), rp(余り) AXDE ÷ rp <sup>注2</sup>					

注1 . r = 0 の場合 , r X, AX FFFFH

2 . rp = 0 の場合 , rp DE, AXDE FFFFFFFFH

( 10 ) 特殊演算命令 : MACW, MACSW, SACW

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
MACW	byte	3	AXDE (B) × (C) + AXDE, B B + 2 , C C + 2 , byte byte - 1 End if( byte = 0 or P/V = 1 )	x	x	x	V	x
MACSW	byte	3	AXDE (B) × (C) + AXDE, B B + 2 , C C + 2 , byte byte - 1 if byte = 0 then End if P/V = 1 then if overflow AXDE 7FFFFFFFH, End if underflow AXDE 80000000H, End	x	x	x	V	x
SACW	[ TDE + ] [ WHL + ]	4	AX  (TDE) - (WHL)  + AX, TDE TDE + 2, WHL WHL + 2 C C - 1 End if( C = 0 or CY = 1 )	x	x	x	V	x

(11) 増減命令 : INC, DEC, INCW, DECW, INCG, DECG

モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
INC	r	1/2	r r + 1	x	x	x	V	
	saddr	2/3	(saddr) (saddr) + 1	x	x	x	V	
DEC	r	1/2	r r - 1	x	x	x	V	
	saddr	2/3	(saddr) (saddr) - 1	x	x	x	V	
INCW	rp	2/1	rp rp + 1					
	saddrp	3/4	(saddrp) (saddrp) + 1					
DECW	rp	2/1	rp rp - 1					
	saddrp	3/4	(saddrp) (saddrp) - 1					
INCG	rg	2	rg rg + 1					
DECG	rg	2	rg rg - 1					

(12) 補正命令 : ADJBA, ADJBS, CVTBW

モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
ADJBA		2	Decimal Adjust Accumulator after Addition	x	x	x	P	x
ADJBS		2	Decimal Adjust Accumulator after Subtract	x	x	x	P	x
CVTBW		1	X A, A 00H if A <sub>7</sub> = 0					
			X A, A FFH if A <sub>7</sub> = 1					

(13) シフト・ローテート命令 : ROR, ROL, RORC, ROLC, SHR, SHL, SHRW, SHLW, ROR4, ROL4

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
ROR	r, n	2/3	(CY, r7 r0, r <sub>m-1</sub> r <sub>m</sub> ) × n回 n = 0 - 7				P	x
ROL	r, n	2/3	(CY, r0 r7, r <sub>m+1</sub> r <sub>m</sub> ) × n回 n = 0 - 7				P	x
RORC	r, n	2/3	(CY r0, r7 CY, r <sub>m-1</sub> r <sub>m</sub> ) × n回 n = 0 - 7				P	x
ROLC	r, n	2/3	(CY r7, r0 CY, r <sub>m+1</sub> r <sub>m</sub> ) × n回 n = 0 - 7				P	x
SHR	r, n	2/3	(CY r0, r7 0, r <sub>m-1</sub> r <sub>m</sub> ) × n回 n = 0 - 7	x	x	0	P	x
SHL	r, n	2/3	(CY r7, r0 0, r <sub>m+1</sub> r <sub>m</sub> ) × n回 n = 0 - 7	x	x	0	P	x
SHRW	rp, n	2	(CY rp0, rp15 0, rp <sub>m-1</sub> rp <sub>m</sub> ) × n回 n = 0 - 7	x	x	0	P	x
SHLW	rp, n	2	(CY rp15, rp0 0, rp <sub>m+1</sub> rp <sub>m</sub> ) × n回 n = 0 - 7	x	x	0	P	x
ROR4	mem3	2	A <sub>3-0</sub> (mem3) <sub>3-0</sub> , (mem3) <sub>7-4</sub> A <sub>3-0</sub> , (mem3) <sub>3-0</sub> (mem3) <sub>7-4</sub>					
ROL4	mem3	2	A <sub>3-0</sub> (mem3) <sub>7-4</sub> , (mem3) <sub>3-0</sub> A <sub>3-0</sub> , (mem3) <sub>7-4</sub> (mem3) <sub>3-0</sub>					

(14) ビット操作命令 : MOV1, AND1, OR1, XOR1, NOT1, SET1, CLR1

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
MOV1	CY, saddr. bit	3/4	CY (saddr. bit)					x
	CY, sfr. bit	3	CY sfr. bit					x
	CY, X. bit	2	CY X. bit					x
	CY, A. bit	2	CY A. bit					x
	CY, PSWL. bit	2	CY PSWL. bit					x
	CY, PSWH. bit	2	CY PSWH. bit					x
	CY, !addr16. bit	5	CY !addr16. bit					x
	CY, !!addr24. bit	2	CY !!addr24. bit					x
	CY, mem2. bit	2	CY mem2. bit					x
	saddr. bit, CY	3/4	(saddr. bit) CY					
	sfr. bit, CY	3	sfr. bit CY					
	X. bit, CY	2	X. bit CY					
	A. bit, CY	2	A. bit CY					
	PSWL. bit, CY	2	PSWL. bit CY	x	x	x	x	x
	PSWH. bit, CY	2	PSWH. bit CY					
	!addr16. bit, CY	5	!addr16. bit CY					
	!!addr24. bit, CY	6	!!addr24. bit CY					
	mem2. bit, CY	2	mem2. bit CY					

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
AND1	CY, saddr. bit	3/4	CY CY ( saddr. bit )					x
	CY, /saddr. bit	3/4	CY CY ( $\overline{\text{saddr. bit}}$ )					x
	CY, sfr. bit	3	CY CY sfr. bit					x
	CY, /sfr. bit	3	CY CY $\overline{\text{sfr. bit}}$					x
	CY, X. bit	2	CY CY X. bit					x
	CY, /X. bit	2	CY CY $\overline{\text{X. bit}}$					x
	CY, A. bit	2	CY CY A. bit					x
	CY, /A. bit	2	CY CY $\overline{\text{A. bit}}$					x
	CY, PSWL. bit	2	CY CY PSWL. bit					x
	CY, /PSWL. bit	2	CY CY $\overline{\text{PSWL. bit}}$					x
	CY, PSWH. bit	2	CY CY PSWH. bit					x
	CY, /PSWH. bit	2	CY CY $\overline{\text{PSWH. bit}}$					x
	CY, !addr16. bit	5	CY CY !addr16. bit					x
	CY, /!addr16. bit	5	CY CY $\overline{\text{!addr16. bit}}$					x
	CY, !!addr24. bit	2	CY CY !!addr24. bit					x
	CY, /!!addr24. bit	6	CY CY $\overline{\text{!!addr24. bit}}$					x
	CY, mem2. bit	2	CY CY mem2. bit					x
	CY, /mem2. bit	2	CY CY $\overline{\text{mem2. bit}}$					x
OR1	CY, saddr. bit	3/4	CY CY ( saddr. bit )					x
	CY, /saddr. bit	3/4	CY CY ( $\overline{\text{saddr. bit}}$ )					x
	CY, sfr. bit	3	CY CY sfr. bit					x
	CY, /sfr. bit	3	CY CY $\overline{\text{sfr. bit}}$					x
	CY, X. bit	2	CY CY X. bit					x
	CY, /X. bit	2	CY CY $\overline{\text{X. bit}}$					x
	CY, A. bit	2	CY CY A. bit					x
	CY, /A. bit	2	CY CY $\overline{\text{A. bit}}$					x
	CY, PSWL. bit	2	CY CY PSWL. bit					x
	CY, /PSWL. bit	2	CY CY $\overline{\text{PSWL. bit}}$					x
	CY, PSWH. bit	2	CY CY PSWH. bit					x
	CY, /PSWH. bit	2	CY CY $\overline{\text{PSWH. bit}}$					x
	CY, !addr16. bit	5	CY CY !addr16. bit					x
	CY, /!addr16. bit	5	CY CY $\overline{\text{!addr16. bit}}$					x
	CY, !!addr24. bit	2	CY CY !!addr24. bit					x
	CY, /!!addr24. bit	6	CY CY $\overline{\text{!!addr24. bit}}$					x
	CY, mem2. bit	2	CY CY mem2. bit					x
	CY, /mem2. bit	2	CY CY $\overline{\text{mem2. bit}}$					x

二モニック	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
XOR1	CY, saddr. bit	3/4	CY CY ∨ ( saddr. bit )						x
	CY, sfr. bit	3	CY CY ∨ sfr. bit						x
	CY, X. bit	2	CY CY ∨ X. bit						x
	CY, A. bit	2	CY CY ∨ A. bit						x
	CY, PSWL. bit	2	CY CY ∨ PSWL. bit						x
	CY, PSWH. bit	2	CY CY ∨ PSWH. bit						x
	CY, !addr16. bit	5	CY CY ∨ !addr16. bit						x
	CY, !!addr24. bit	2	CY CY ∨ !!addr24. bit						x
	CY, mem2. bit	2	CY CY ∨ mem2. bit						x
NOT1	saddr. bit	3/4	( saddr. bit ) ( $\overline{\text{saddr. bit}}$ )						
	sfr. bit	3	sfr. bit $\overline{\text{sfr. bit}}$						
	X. bit	2	X. bit $\overline{\text{X. bit}}$						
	A. bit	2	A. bit $\overline{\text{A. bit}}$						
	PSWL. bit	2	PSWL. bit $\overline{\text{PSWL. bit}}$	x	x	x	x	x	
	PSWH. bit	2	PSWH. bit $\overline{\text{PSWH. bit}}$						
	!addr16. bit	5	!addr16. bit $\overline{\text{!addr16. bit}}$						
	!!addr24. bit	2	!!addr24. bit $\overline{\text{!!addr24. bit}}$						
	mem2. bit	2	mem2. bit $\overline{\text{mem2. bit}}$						
	CY	1	CY $\overline{\text{CY}}$						x
SET1	saddr. bit	2/3	( saddr. bit ) 1						
	sfr. bit	3	sfr. bit 1						
	X. bit	2	X. bit 1						
	A. bit	2	A. bit 1						
	PSWL. bit	2	PSWL. bit 1	x	x	x	x	x	
	PSWH. bit	2	PSWH. bit 1						
	!addr16. bit	5	!addr16. bit 1						
	!!addr24. bit	2	!!addr24. bit 1						
	mem2. bit	2	mem2. bit 1						
	CY	1	CY 1						1
CLR1	saddr. bit	2/3	( saddr. bit ) 0						
	sfr. bit	3	sfr. bit 0						
	X. bit	2	X. bit 0						
	A. bit	2	A. bit 0						
	PSWL. bit	2	PSWL. bit 0	x	x	x	x	x	
	PSWH. bit	2	PSWH. bit 0						
	!addr16. bit	5	!addr16. bit 0						
	!!addr24. bit	2	!!addr24. bit 0						
	mem2. bit	2	mem2. bit 0						
	CY	1	CY 0						0

(15) スタック操作命令 : PUSH, PUSHU, POP, POPU, MOVG, ADDWG, SUBWG, INCG, DECG

ニモニック	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
PUSH	PSW	1	( SP - 2 ) PSW, SP SP - 2						
	sfrp	3	( SP - 2 ) sfrp, SP SP - 2						
	sfr	3	( SP - 1 ) sfr, SP SP - 1						
	post	2	{( SP - 2 ) post, SP SP - 2 }x m回 <sup>注</sup>						
	rg	2	( SP - 3 ) rg, SP SP - 3						
PUSHU	post	2	{( UUP - 2 ) post, UUP UUP - 2 }x m回 <sup>注</sup>						
POP	PSW	1	PSW ( SP ), SP SP + 2	R	R	R	R	R	
	sfrp	3	sfrp ( SP ), SP SP + 2						
	sfr	3	sfr ( SP ), SP SP + 1						
	post	2	{ post ( SP ), SP SP + 2 }x m回 <sup>注</sup>						
	rg	2	rg ( SP ), SP SP + 3						
POPU	post	2	{ post ( UUP ), UUP UUP + 2 }x m回 <sup>注</sup>						
MOVG	SP, imm24	5	SP imm24						
	SP, WHL	2	SP WHL						
	WHL, SP	2	WHL SP						
ADDWG	SP, word	4	SP SP + word						
SUBWG	SP, word	4	SP SP - word						
INCG	SP	2	SP SP + 1						
DECG	SP	2	SP SP - 1						

注 mは, postで指定されたレジスタ数



(16) コール・リターン命令 : CALL, CALLF, CALLT, BRK, BRKCS, RET, RETI, RETB, RETCS, RETCSB

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
CALL	!addr16	3	(SP - 3) (PC + 3) SP SP - 3, PC <sub>HW</sub> 0, PC <sub>LW</sub> addr16					
	!!addr20	4	(SP - 3) (PC + 4) SP SP - 3, PC addr20					
	rp	2	(SP - 3) (PC + 2) SP SP - 3, PC <sub>HW</sub> 0, PC <sub>LW</sub> rp					
	rg	2	(SP - 3) (PC + 2) SP SP - 3, PC rg					
	[ rp ]	2	(SP - 3) (PC + 2) SP SP - 3, PC <sub>HW</sub> 0, PC <sub>LW</sub> (rp)					
	[ rg ]	2	(SP - 3) (PC + 2) SP SP - 3, PC (rg)					
	\$!addr20	3	(SP - 3) (PC + 3) SP SP - 3, PC PC + 3 + jdisp16					
CALLF	!addr11	2	(SP - 3) (PC + 2) SP SP - 3, PC <sub>19-12</sub> 0, PC <sub>11</sub> 1, PC <sub>10-0</sub> addr11					
CALLT	[ addr5 ]	1	(SP - 3) (PC + 1) SP SP - 3, PC <sub>HW</sub> 0, PC <sub>LW</sub> (addr5)					
BRK		1	(SP - 2) PSW, (SP - 1) <sub>0-3</sub> , (PC + 1) <sub>HW</sub> , (SP - 4) (PC + 1) <sub>LW</sub> , SP SP - 4 PC <sub>HW</sub> 0, PC <sub>LW</sub> (003EH)					
BRKCS	RBn	2	PC <sub>LW</sub> RP2, RP3 PSW, RBS2 - 0 n, RSS 0, IE 0, RP3 <sub>8-11</sub> PC <sub>HW</sub> , PC <sub>HW</sub> 0					
RET		1	PC (SP) SP SP + 3					
RETI		1	PC <sub>LW</sub> (SP), PC <sub>HW</sub> (SP + 3) <sub>0-3</sub> , PSW (SP + 2), SP SP + 4 ISPR中でセット(1)されている最も優先順位の 高いフラグをクリア(0)	R	R	R	R	R
RETB		1	PC <sub>LW</sub> (SP), PC <sub>HW</sub> (SP + 3) <sub>0-3</sub> , PSW (SP + 2), SP SP + 4	R	R	R	R	R
RETCS	!addr16	3	PSW RP3, PC <sub>LW</sub> RP2, RP2 addr16, PC <sub>HW</sub> RP3 <sub>8-11</sub> ISPR中でセット(1)されている最も優先順位の 高いフラグをクリア(0)	R	R	R	R	R
RETCSB	!addr16	4	PSW RP3, PC <sub>LW</sub> RP2, RP2 addr16, PC <sub>HW</sub> RP3 <sub>8-11</sub>	R	R	R	R	R

(17) 無条件分岐命令 : BR

ニモニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
BR	!addr16	3	PC <sub>HW</sub> 0, PC <sub>LW</sub> addr16					
	!!addr20	4	PC addr20					
	rp	2	PC <sub>HW</sub> 0, PC <sub>LW</sub> rp					
	rg	2	PC rg					
	[ rp ]	2	PC <sub>HW</sub> 0, PC <sub>LW</sub> ( rp )					
	[ rg ]	2	PC ( rg )					
	\$ addr20	2	PC PC + 2 + jdisp8					
	\$ !addr20	3	PC PC + 3 + jdisp16					

(18) 条件付き分岐命令 : BNZ, BNE, BZ, BE, BNC, BNL, BC, BL, BNV, BPO, BV, BPE, BP, BN, BLT, BGE, BLE, BGT, BNH, BH, BF, BT, BTCLR, BFSET, DBNZ

二モニク	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
BNZ	\$ addr20	2	PC PC + 2 + jdisp8 if Z = 0					
BNE								
BZ	\$ addr20	2	PC PC + 2 + jdisp8 if Z = 1					
BE								
BNC	\$ addr20	2	PC PC + 2 + jdisp8 if CY = 0					
BNL								
BC	\$ addr20	2	PC PC + 2 + jdisp8 if CY = 1					
BL								
BNV	\$ addr20	2	PC PC + 2 + jdisp8 if P/V = 0					
BPO								
BV	\$ addr20	2	PC PC + 2 + jdisp8 if P/V = 1					
BPE								
BP	\$ addr20	2	PC PC + 2 + jdisp8 if S = 0					
BN	\$ addr20	2	PC PC + 2 + jdisp8 if S = 1					
BLT	\$ addr20	3	PC PC + 3 + jdisp8 if P/V $\nabla$ S = 1					
BGE	\$ addr20	3	PC PC + 3 + jdisp8 if P/V $\nabla$ S = 0					
BLE	\$ addr20	3	PC PC + 3 + jdisp8 if (P/V $\nabla$ S) Z = 1					
BGT	\$ addr20	3	PC PC + 3 + jdisp8 if (P/V $\nabla$ S) Z = 0					
BNH	\$ addr20	3	PC PC + 3 + jdisp8 if Z CY = 1					
BH	\$ addr20	3	PC PC + 3 + jdisp8 if Z CY = 0					
BF	saddr. bit, \$ addr20	4/5	PC PC + 4 <sup>注</sup> + jdisp8 if (saddr. bit) = 0					
	sfr. bit, \$ addr20	4	PC PC + 4 + jdisp8 if sfr. bit = 0					
	X. bit, \$ addr20	3	PC PC + 3 + jdisp8 if X. bit = 0					
	A. bit, \$ addr20	3	PC PC + 3 + jdisp8 if A. bit = 0					
	PSWL. bit, \$ addr20	3	PC PC + 3 + jdisp8 if PSWL. bit = 0					
	PSWH. bit, \$ addr20	3	PC PC + 3 + jdisp8 if PSWH. bit = 0					
	!addr16. bit, \$ addr20	6	PC PC + 3 + jdisp8 if !addr16. bit = 0					
	!!addr24. bit, \$ addr20	3	PC PC + 3 + jdisp8 if !!addr24. bit = 0					
mem2. bit, \$ addr20	3	PC PC + 3 + jdisp8 if mem2. bit = 0						

注 バイト数が4のとき。5のときはPC PC + 5 + jdisp8になります。

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
BT	saddr. bit, \$ addr20	3/4	PC PC + 3 <sup>注1</sup> + jdisp8 if ( saddr. bit )= 1					
	sfr. bit, \$ addr20	4	PC PC + 4 + jdisp8 if sfr. bit = 1					
	X. bit, \$ addr20	3	PC PC + 3 + jdisp8 if X. bit = 1					
	A. bit, \$ addr20	3	PC PC + 3 + jdisp8 if A. bit = 1					
	PSWL. bit, \$ addr20	3	PC PC + 3 + jdisp8 if PSWL. bit = 1					
	PSWH. bit, \$ addr20	3	PC PC + 3 + jdisp8 if PSWH. bit = 1					
	!addr16. bit, \$ addr20	6	PC PC + 3 + jdisp8 if !addr16. bit = 1					
	!!addr24. bit, \$ addr20	3	PC PC + 3 + jdisp8 if !!addr24. bit = 1					
mem2. bit, \$ addr20	3	PC PC + 3 + jdisp8 if mem2. bit = 1						
BTCLR	saddr. bit, \$ addr20	4/5	{ PC PC + 4 <sup>注2</sup> + jdisp8, ( saddr. bit ) 0 } if ( saddr. bit )= 1					
	sfr. bit, \$ addr20	4	{ PC PC + 4 + jdisp8, sfr. bit 0 } if sfr. bit = 1					
	X. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, X. bit 0 }if X. bit = 1					
	A. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, A. bit 0 }if A. bit = 1					
	PSWL. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, PSWL. bit 0 } if PSWL. bit = 1	x	x	x	x	x
	PSWH. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, PSWH. bit 0 } if PSWH. bit = 1					
	!addr16. bit, \$ addr20	6	{ PC PC + 3 + jdisp8, !addr16. bit 0 } if !addr16. bit = 1					
	!!addr24. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, !!addr24. bit 0 } if !!addr24. bit = 1					
mem2. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, mem2. bit 0 } if mem2. bit = 1						

注1 . バイト数が3のとき。4のときはPC PC + 4 + jdisp8になります。

2 . バイト数が4のとき。5のときはPC PC + 5 + jdisp8になります。

二モニク	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
BFSET	saddr. bit, \$ addr20	4/5	{ PC PC + 4 <sup>注2</sup> + jdisp8, ( saddr. bit ) 1 } if ( saddr. bit ) = 0						
	sfr. bit, \$ addr20	4	{ PC PC + 4 + jdisp8, sfr. bit 1 } if sfr. bit = 0						
	X. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, X. bit 1 } if X. bit = 0						
	A. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, A. bit 1 } if A. bit = 0						
	PSWL. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, PSWL. bit 1 } if PSWL. bit = 0	x	x	x	x	x	
	PSWH. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, PSWH. bit 1 } if PSWH. bit = 0						
	!addr16. bit, \$ addr20	6	{ PC PC + 3 + jdisp8, !addr16. bit 1 } if !addr16. bit = 0						
	!!addr24. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, !!addr24. bit 1 } if !!addr24. bit = 0						
	mem2. bit, \$ addr20	3	{ PC PC + 3 + jdisp8, mem2. bit 1 } if mem2. bit = 0						
DBNZ	B, \$ addr20	2	B B - 1, PC PC + 2 + jdisp8 if B = 0						
	C, \$ addr20	2	C C - 1, PC PC + 2 + jdisp8 if C = 0						
	saddr, \$ addr20	3/4	( saddr ) ( saddr ) - 1, PC PC + 3 <sup>注1</sup> + jdisp8 if ( saddr ) = 0						

注1 . バイト数が3のとき。4のときはPC PC + 4 + jdisp8になります。

2 . バイト数が4のとき。5のときはPC PC + 5 + jdisp8になります。

(19) CPU制御命令 : MOV, LOCATION, SEL, SWRS, NOP, EI, DI

二モニク	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
MOV	STBC, byte	4	STBC byte						
	WDM, byte	4	WDM byte						
LOCATION	locaddr	4	SFR, 内部データ領域の配置アドレスの上位ワード指定						
SEL	RBn	2	RSS 0, RBS2 - 0 n						
	RBn, ALT	2	RSS 1, RBS2 - 0 n						
SWRS		2	RSS $\overline{\text{RSS}}$						
NOP		1	No Operation						
EI		1	IE 1 ( Enable interrupt )						
DI		1	IE 0 ( Disable interrupt )						

(20) スtring命令 : MOVTLW, MOV, XCHM, MOVBK, XCHBK, CMPME, CMPMNE, CMPMC, CMPMNC, CMPBKE, CMPBKNE, CMPBKC, CMPBKNC

ニモニック	オペランド	バイト	オペレーション	フラグ					
				S	Z	AC	P/V	CY	
MOVTLW	!addr8, byte	4	(addr8 + 2) (addr8), byte byte - 1, addr8 addr8 - 2 End if byte = 0						
MOV	[TDE + ] A	2	(TDE) A, TDE TDE + 1, C C - 1 End if C = 0						
	[TDE - ] A	2	(TDE) A, TDE TDE - 1, C C - 1 End if C = 0						
XCHM	[TDE + ] A	2	(TDE) A, TDE TDE + 1, C C - 1 End if C = 0						
	[TDE - ] A	2	(TDE) A, TDE TDE - 1, C C - 1 End if C = 0						
MOVBK	[TDE + ] [WHL + ]	2	(TDE) (WHL), TDE TDE + 1, WHL WHL + 1, C C - 1 End if C = 0						
	[TDE - ] [WHL - ]	2	(TDE) (WHL), TDE TDE - 1, WHL WHL - 1, C C - 1 End if C = 0						
XCHBK	[TDE + ] [WHL + ]	2	(TDE) (WHL), TDE TDE + 1, WHL WHL + 1, C C - 1 End if C = 0						
	[TDE - ] [WHL - ]	2	(TDE) (WHL), TDE TDE - 1, WHL WHL - 1, C C - 1 End if C = 0						
CMPME	[TDE + ] A	2	(TDE)-A, TDE TDE + 1, C C - 1 End if C = 0 or Z = 0	x	x	x	V	x	
	[TDE - ] A	2	(TDE)-A, TDE TDE - 1, C C - 1 End if C = 0 or Z = 0	x	x	x	V	x	
CMPMNE	[TDE + ] A	2	(TDE)-A, TDE TDE + 1, C C - 1 End if C = 0 or Z = 1	x	x	x	V	x	
	[TDE - ] A	2	(TDE)-A, TDE TDE - 1, C C - 1 End if C = 0 or Z = 1	x	x	x	V	x	
CMPMC	[TDE + ] A	2	(TDE)-A, TDE TDE + 1, C C - 1 End if C = 0 or CY = 0	x	x	x	V	x	
	[TDE - ] A	2	(TDE)-A, TDE TDE - 1, C C - 1 End if C = 0 or CY = 0	x	x	x	V	x	
CMPMNC	[TDE + ] A	2	(TDE)-A, TDE TDE + 1, C C - 1 End if C = 0 or CY = 1	x	x	x	V	x	
	[TDE - ] A	2	(TDE)-A, TDE TDE - 1, C C - 1 End if C = 0 or CY = 1	x	x	x	V	x	
CMPBKE	[TDE + ] [WHL + ]	2	(TDE)-(WHL), TDE TDE + 1, WHL WHL + 1, C C - 1 End if C = 0 or Z = 0	x	x	x	V	x	
	[TDE - ] [WHL - ]	2	(TDE)-(WHL), TDE TDE - 1, WHL WHL - 1, C C - 1 End if C = 0 or Z = 0	x	x	x	V	x	

二モニック	オペランド	バイト	オペレーション	フラグ				
				S	Z	AC	P/V	CY
CMPBKNE	[ TDE + ] [ WHL + ]	2	( TDE )-( WHL ), TDE TDE + 1 , WHL WHL + 1, C C - 1 End if C = 0 or Z = 1	x	x	x	V	x
	[ TDE - ] [ WHL - ]	2	( TDE )-( WHL ), TDE TDE - 1 , WHL WHL - 1, C C - 1 End if C = 0 or Z = 1	x	x	x	V	x
CMPBKC	[ TDE + ] [ WHL + ]	2	( TDE )-( WHL ), TDE TDE + 1 , WHL WHL + 1, C C - 1 End if C = 0 or CY = 0	x	x	x	V	x
	[ TDE - ] [ WHL - ]	2	( TDE )-( WHL ), TDE TDE - 1 , WHL WHL - 1, C C - 1 End if C = 0 or CY = 0	x	x	x	V	x
CMPBKNC	[ TDE + ] [ WHL + ]	2	( TDE )-( WHL ), TDE TDE + 1 , WHL WHL + 1, C C - 1 End if C = 0 or CY = 1	x	x	x	V	x
	[ TDE - ] [ WHL - ]	2	( TDE )-( WHL ), TDE TDE - 1 , WHL WHL - 1, C C - 1 End if C = 0 or CY = 1	x	x	x	V	x

### 18.3 アドレッシング別命令一覧

(1) 8ビット命令( ( ) 内は、rとしてAを記述することで実現している組み合わせです)

MOV, XCH, ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP, MULU, DIVUW, INC, DEC, ROR, ROL, RORC, ROLC, SHR, SHL, ROR4, ROL4, DBNZ, PUSH, POP, MOVVM, XCHM, CMPME, CMPMNE, CMPMNC, CMPMC, MOVBK, XCHBK, CMPBKE, CMPBKNE, CMPBKNC, CMPBKC

表18-1 8ビット・アドレッシング別命令一覧表

第2オペランド 第1オペランド	byte	A	r r'	saddr saddr'	sfr	!addr16 !!addr24	mem [ saddrp ] [ %saddrg ]	r3 PSWL PSWH	[ WHL + ] [ WHL - ]	n	なし <sup>注2</sup>
A	(MOV) ADD <sup>注1</sup>	(MOV) (XCH) (ADD) <sup>注1</sup>	MOV XCH (ADD) <sup>注1</sup>	(MOV) <sup>注6</sup> (XCH) <sup>注6</sup> (ADD) <sup>注1,6</sup>	MOV (XCH) (ADD) <sup>注1</sup>	(MOV) (XCH) ADD <sup>注1</sup>	MOV XCH ADD <sup>注1</sup>	MOV	(MOV) (XCH) (ADD) <sup>注1</sup>		
r	MOV ADD <sup>注1</sup>	(MOV) (XCH) (ADD) <sup>注1</sup>	MOV XCH ADD <sup>注1</sup>	MOV XCH ADD <sup>注1</sup>	MOV XCH ADD <sup>注1</sup>	MOV XCH				ROR <sup>注3</sup>	MULU DIVUW INC DEC
saddr	MOV ADD <sup>注1</sup>	(MOV) <sup>注6</sup> (ADD) <sup>注1</sup>	MOV ADD <sup>注1</sup>	MOV XCH ADD <sup>注1</sup>							INC DEC DBNZ
sfr	MOV ADD <sup>注1</sup>	MOV (ADD) <sup>注1</sup>	MOV ADD <sup>注1</sup>								PUSH POP
!addr16 !!addr24	MOV	MOV ADD <sup>注1</sup>	MOV								
mem [ saddrp ] [ %saddrg ]		MOV ADD <sup>注1</sup>									
mem3											ROR4 ROL4
r3 PSWL PSWH	MOV	MOV									
B, C											DBNZ
STBC, WDM	MOV										
[ TDE + ] [ TDE - ]		(MOV) (ADD) <sup>注1</sup> MOVVM <sup>注4</sup>							MOVBK <sup>注5</sup>		

注1 . ADDC, SUB, SUBC, AND, OR, XOR, CMPIはADDと同じ

2 . 第2オペランドがないか, 第2オペランドがオペランド・アドレスでない

3 . ROL, RORC, ROLC, SHR, SHLはRORと同じ

4 . XCHM, CMPME, CMPMNE, CMPMNC, CMPMCはMOVVMと同じ

5 . XCHBK, CMPBKE, CMPBKNE, CMPBKNC, CMPBKCはMOVBKと同じ

6 . この組み合わせでsaddrがsaddr2の場合, 短いコード長の命令がある



(2) 16ビット命令 ( ( ) 内は, rpとしてAXを記述することで実現している組み合わせです)

MOVM, XCHW, ADDW, SUBW, CMPW, MULW, MULW, DIVUX, INCW, DECW, SHRW,  
SHLW, PUSH, POP, ADDWG, SUBWG, PUSHU, POPU, MOVTBLW, MACW, MACSW, SACW

表18 - 2 16ビット・アドレッシング別命令一覧表

第2オペランド \ 第1オペランド	word	AX	rp rp'	saddrp saddrp'	sfrp	!addr16 !!addr24	mem [ saddrp ] [ %saddrg ]	[ WHL + ]	byte	n	なし <sup>注2</sup>
AX	(MOVW) ADDW <sup>注1</sup>	(MOVW) (XCHW) (ADD) <sup>注1</sup>	(MOVW) (XCHW) (ADD) <sup>注1</sup>	(MOVW) <sup>注3</sup> (XCHW) <sup>注3</sup> (ADD) <sup>注1,3</sup>	MOVW (XCHW) (ADD) <sup>注1</sup>	(MOVW) XCHW	MOVW XCHW	(MOVW) (XCHW)			
rp	MOVW ADDW <sup>注1</sup>	(MOVW) (XCHW) (ADD) <sup>注1</sup>	MOVW XCHW ADDW <sup>注1</sup>	MOVW XCHW ADDW <sup>注1</sup>	MOVW XCHW ADDW <sup>注1</sup>	MOVW				SHRW SHLW	MULW <sup>注4</sup> INCW DECW
saddrp	MOVW ADDW <sup>注1</sup>	(MOVW) <sup>注3</sup> (ADD) <sup>注1</sup>	MOVW ADDW <sup>注1</sup>	MOVW XCHW ADDW <sup>注1</sup>							INCW DECW
sfrp	MOVW ADDW <sup>注1</sup>	MOVW (ADD) <sup>注1</sup>	MOVW ADDW <sup>注1</sup>								PUSH POP
!addr16 !!addr24	MOVW	(MOVW)	MOVW						MOVTBLW		
mem [ saddrp ] [ %saddrg ]		MOVW									
PSW											PUSH POP
SP	ADDWG SUBWG										
post											PUSH POP PUSHU POPU
[ TDE + ]		(MOVW)						SACW			
byte											MACW MACSW

注1 . SUBW, CMPWはADDWと同じ

- 2 . 第2オペランドがないか, 第2オペランドがオペランド・アドレスでない
- 3 . この組み合わせでsaddrpがsaddrp2の場合, 短いコード長の命令がある
- 4 . MULW, DIVUXはMULWと同じ

(3) 24ビット命令( )内は、rgとしてWHLを記述することで実現している組み合わせです)

MOVG, ADDG, SUBG, INCG, DECG, PUSH, POP

表18 - 3 24ビット・アドレッシング別命令一覧表

第2オペランド \ 第1オペランド	imm24	WHL	rg rg'	saddrg	!!addr24	mem1	[ %saddrg ]	SP	なし <sup>注</sup>
WHL	(MOVG) (ADDG) (SUBG)	(MOVG) (ADDG) (SUBG)	(MOVG) (ADDG) (SUBG)	(MOVG) ADDG SUBG	(MOVG)	MOVG	MOVG	MOVG	
rg	MOVG ADDG SUBG	(MOVG) (ADDG) (SUBG)	MOVG ADDG SUBG	MOVG	MOVG				INCG DECG PUSH POP
saddrg		(MOVG)	MOVG						
!!addr24		(MOVG)	MOVG						
mem1		MOVG							
[ %saddrg ]		MOVG							
SP	MOVG	MOVG							INCG DECG

注 第2オペランドがないか、第2オペランドがオペランド・アドレスでない

(4) ビット操作命令

MOV1, AND1, OR1, XOR1, SET1, CLR1, NOT1, BT, BF, BTCLR, BFSET

表18 - 4 ビット操作命令アドレッシング別命令一覧表

第2オペランド \ 第1オペランド	CY	saddr. bit A. bit PSWL. bit mem2. bit !addr16. bit !!addr24. bit	sfr. bit X. bit PSWH. bit	/saddr. bit /A. bit /PSWL. bit /mem2. bit /!addr16. bit /!!addr24. bit	/sfr. bit /X. bit /PSWH. bit	なし <sup>注</sup>
CY		MOV1 AND1 OR1 XOR1		AND1 OR1		NOT1 SET1 CLR1
saddr. bit sfr. bit A. bit X. bit PSWL. bit PSWH. bit mem2. bit !addr16. bit !!addr24. bit	MOV1					NOT1 SET1 CLR1 BF BT BTCLR BFSET

注 第2オペランドがないか、第2オペランドがオペランド・アドレスでない

(5) コール・リターン命令 / 分岐命令

CALL, CALLF, CALLT, BRK, RET, RETI, RETB, RETCS, RETCSB, BRKCS, BR, BNZ, BNE, BZ, BE, BNC, BNL, BC, BL, BNV, BPO, BV, BPE, BP, BN, BLT, BGE, BLE, BGT, BNH, BH, BF, BT, BTCLR, BFSET, DBNZ

表18 - 5 コール・リターン命令 / 分岐命令アドレッシング別命令一覧表

命令アドレスの オペランド	\$addr20	\$!addr20	!addr16	!!addr20	rp	rg	[ rp ]	[ rg ]	!addr11	[ addr5 ]	RBn	なし
基本命令	BC <sup>注</sup> BR	CALL BR	CALL BR RETCS RETCSB	CALL BR	CALL BR	CALL BR	CALL BR	CALL BR	CALLF	CALLT	BRKCS	BRK RET RETI RETB
複合命令	BF BT BTCLR BFSET DBNZ											

注 BNZ, BNE, BZ, BE, BNC, BNL, BL, BNV, BPO, BV, BPE, BP, BN, BLT, BGE, BLE, BGT, BNH, BHはBCと同じ

(6) その他の命令

ADJBA, ADJBS, CVTBW, LOCATION, SEL, NOT, EI, DI, SWRS

〔メモ〕

## 付録A 開発ツール

μPD784976サブシリーズの製品を使用するシステム開発のために必要な開発ツール構成を次頁以降に示します。

### PC98-NXシリーズへの対応について

特に断りのないかぎり，IBM PC/AT™互換機でサポートされている製品については，PC98-NXシリーズでも使用できます。PC98-NXシリーズを使用する場合は，IBM PC/AT互換機の説明を参照してください。

### Windowsについて

特に断りのないかぎり，「Windows」は，次のOSを示しています。

- ・ Windows 3.1
- ・ Windows 95
- ・ WindowsNT™ Ver. 4.0

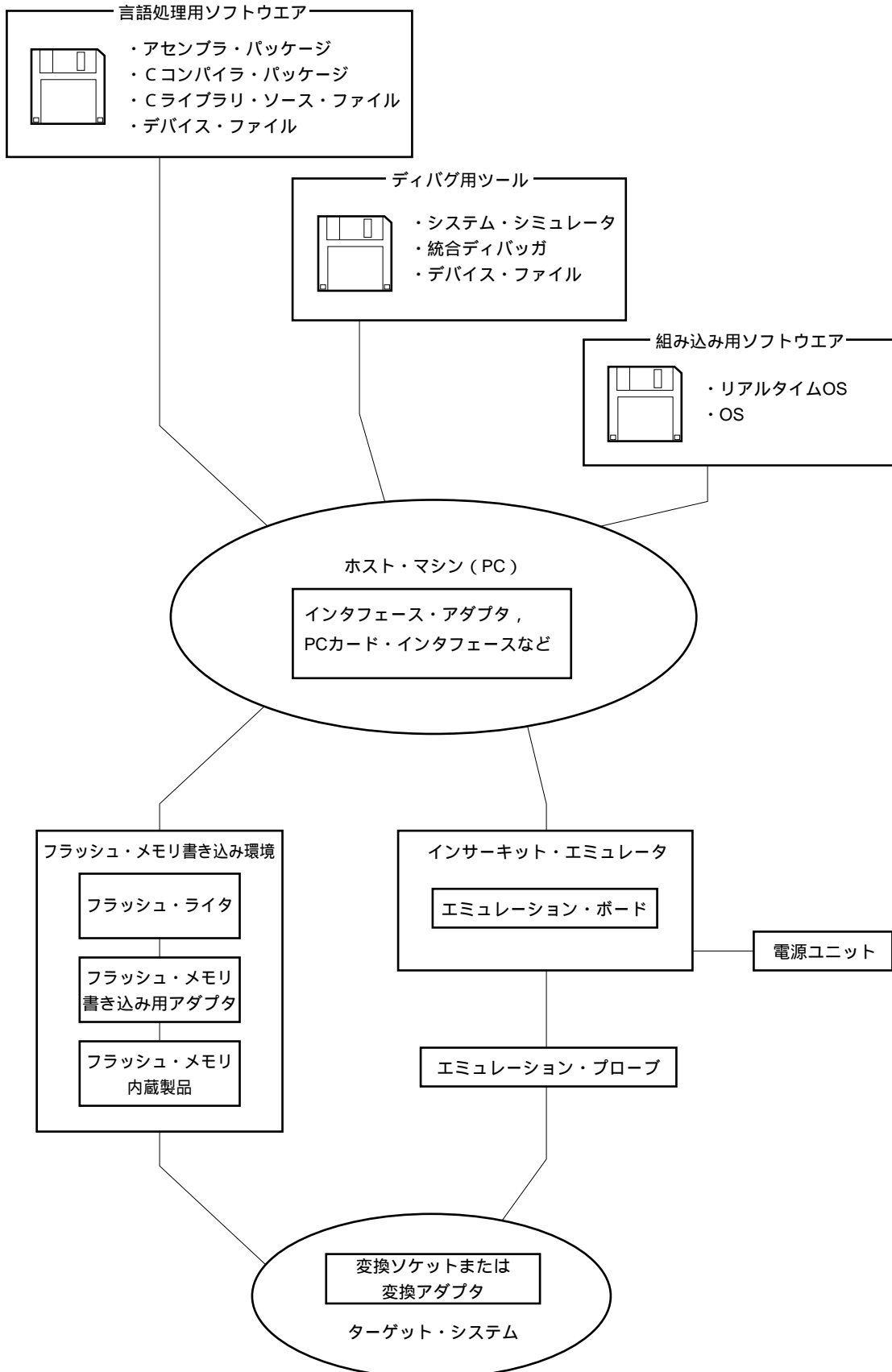
## A.1 言語処理用ソフトウェア

RA78K4 アセンブラ・パッケージ	<p>モニックで書かれたプログラムをマイコンの実行可能なオブジェクト・コードに変換するプログラムです。</p> <p>このほかに、シンボル・テーブルの生成、分岐命令の最適化処理などを自動的に行う機能を備えています。</p> <p>別売のデバイス・ファイル (DF784976) と組み合わせて使用します。</p> <p>&lt;PC環境で使用する場合の注意&gt;</p> <p>アセンブラ・パッケージはDOSベースのアプリケーションですが、Windows上でプロジェクト・マネージャ (アセンブラ・パッケージに含まれています) を使用することにより、Windows環境でも使用できます。</p> <p>オーダー名称: <math>\mu S \times \times \times RA78K4</math></p>
CC78K4 Cコンパイラ・パッケージ	<p>C言語で書かれたプログラムをマイコンの実行可能なオブジェクト・コードに変換するプログラムです。</p> <p>別売のアセンブラ・パッケージおよびデバイス・ファイルと組み合わせて使用します。</p> <p>&lt;PC環境で使用する場合の注意&gt;</p> <p>Cコンパイラ・パッケージはDOSベースのアプリケーションですが、Windows上でプロジェクト・マネージャ (アセンブラ・パッケージに含まれています) を使用することにより、Windows環境でも使用できます。</p> <p>オーダー名称: <math>\mu S \times \times \times CC78K4</math></p>
DF784976 <sup>注</sup> デバイス・ファイル	<p>デバイス固有の情報が入ったファイルです。</p> <p>別売の各ツール (RA78K4, CC78K4, SM78K4, ID78K4-NS, ID78K4) と組み合わせて使用します。</p> <p>対応OS, ホスト・マシンは組み合わせられる各ツールに依存します。</p> <p>オーダー名称: <math>\mu S \times \times \times DF784976</math></p>
CC78K4-L Cライブラリ・ソース・ファイル	<p>Cコンパイラ・パッケージに含まれているオブジェクト・ライブラリを構成する関数のソース・ファイルです。</p> <p>Cコンパイラ・パッケージに含まれているオブジェクト・ライブラリをお客様の仕様にあわせて変更する場合に必要です。</p> <p>ソース・ファイルのため、動作環境はOSに依存しません。</p> <p>オーダー名称: <math>\mu S \times \times \times CC78K4-L</math></p>

注 DF784976は、RA78K4, CC78K4, SM78K4, ID78K4-NS, ID78K4のすべての製品に共通に使用できます。

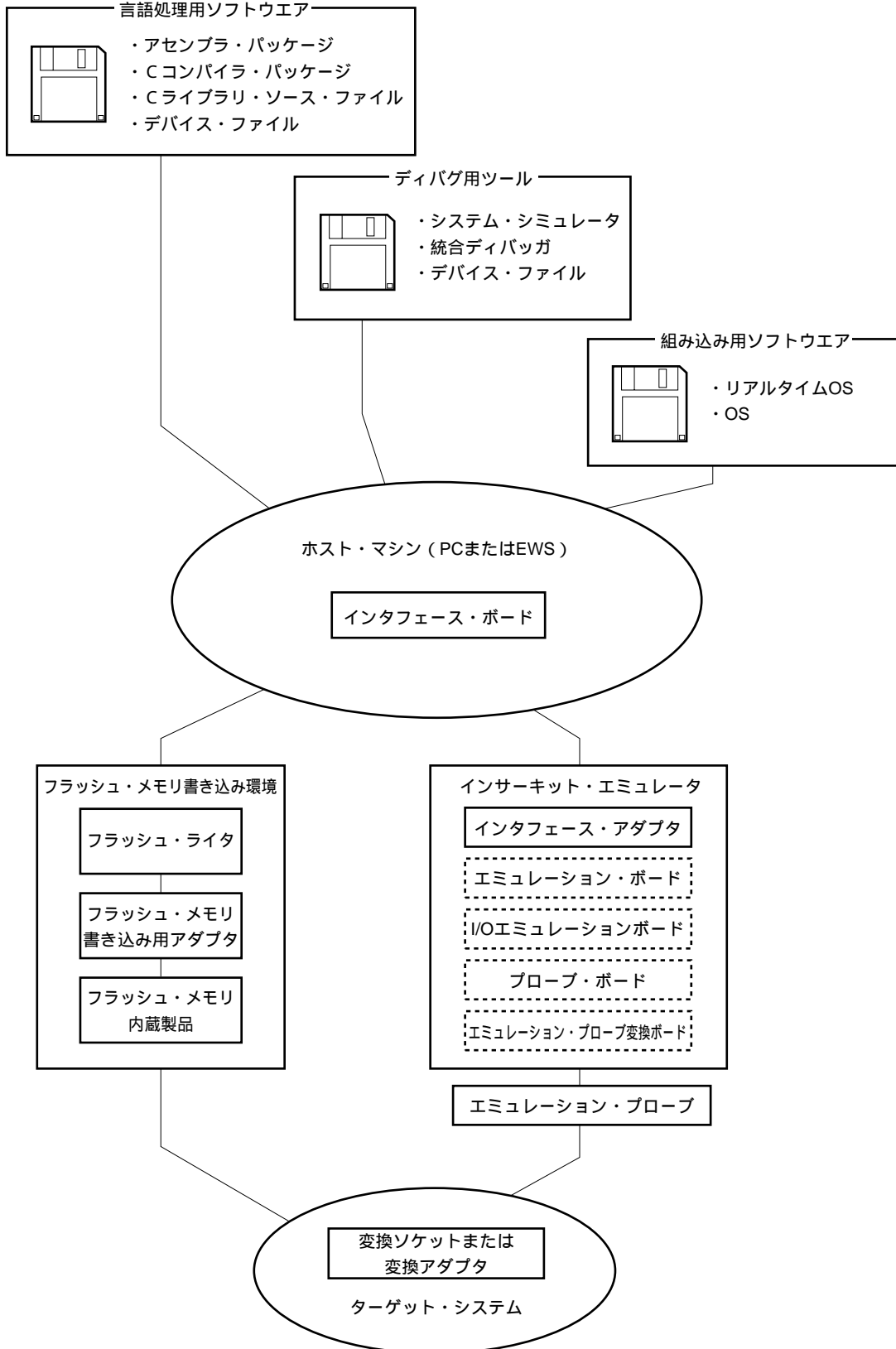
図A - 1 開発ツール構成 (1/2)

(1) インサーキット・エミュレータ IE-78K4-NSを使用する場合



図A - 1 開発ツール構成 (2/2)

(2) インサーキット・エミュレータ IE-784000-Rを使用する場合



備考 破線の部分は開発環境によって異なります。A.3.1 ハードウェアを参照してください。



備考 オーダ名称の××××は、使用するホスト・マシン，OSにより異なります。

- μS××××RA78K4
- μS××××CC78K4
- μS××××DF784976
- μS××××CC78K4-L

××××	ホスト・マシン	OS	供給媒体
AA13	PC-9800シリーズ	日本語Windows <sup>注</sup>	3.5インチ2HD FD
AB13	IBM PC/AT互換機	日本語Windows <sup>注</sup>	3.5インチ2HC FD
BB13		英語Windows <sup>注</sup>	
3P16	HP9000シリーズ700 <sup>TM</sup>	HP-UX ( Rel.10.10 )	DAT ( DDS )
3K13	SPARCstation <sup>TM</sup>	SunOS ( Rel. 4 . 1 . 4 ) ,	3.5インチ2HC FD
3K15		Solaris <sup>TM</sup> ( Rel. 2 . 5 . 1 )	1/4インチCGMT
3R13	NEWS <sup>TM</sup> ( RISC )	NEWS-OS ( Rel. 6 . 1 )	3.5インチ2HC FD

注 DOS環境でも動作します。

## A.2 フラッシュ・メモリ書き込み用ツール

Flashpro (型番 FL-PR3, PG-FP3) フラッシュ・ライター	フラッシュ・メモリ内蔵マイコン専用のフラッシュ・ライターです。
FA-100GF <sup>注</sup> フラッシュ・メモリ書き込み用アダプタ	フラッシュ・メモリ書き込み用アダプタです。Flashpro に接続して使用します。 ・FA-100GF : 100ピン・プラスチックQFP (GF-3BAタイプ) 用

注 開発中

備考 FL-PR2, FL-PR3, FA-100GFは、株式会社内藤電誠町田製作所の製品です。

問い合わせ先：株式会社内藤電誠町田製作所 (TEL (044) 822-3813)

## A.3 デバッグ用ツール

### A.3.1 ハードウェア (1/2)

#### (1) インサーキット・エミュレータ IE-78K4-NSを使用する場合

IE-78K4-NS インサーキット・エミュレータ	78K/ シリーズを使用する応用システムを開発する際に、ハードウェア、ソフトウェアをデバッグするためのインサーキット・エミュレータです。統合ディバッガ (ID78K4-NS) に対応しています。電源ユニット、エミュレーション・プローブおよび、ホスト・マシンと接続するためのインタフェース・アダプタと組み合わせて使用します。
IE-70000-MC-PS-B 電源ユニット	AC100~240 Vのコンセントから電源を供給するためのアダプタです。
IE-70000-98-IF-C インタフェース・アダプタ	IE-78K4-NSのホスト・マシンとしてPC-9800シリーズ(ノート型パソコンを除く)を使用するときに必要なアダプタです(Cバス対応)。
IE-70000-CD-IF-A PCカード・インタフェース	IE-78K4-NSのホスト・マシンとしてノート型パソコンを使用するときに必要なPCカードとインタフェース・ケーブルです(PCMCIAソケット対応)。
IE-70000-PC-IF-C インタフェース・アダプタ	IE-78K4-NSのホスト・マシンとしてIBM PC/AT互換機を使用するときに必要なアダプタです(ISAバス対応)。
IE-70000-PCI-IF インタフェース・アダプタ	IE-78K4-NSのホスト・マシンとしてPCIバスを内蔵したパソコンを使用するときに必要なアダプタです。
IE-784976-NS-EM1 <sup>注</sup> エミュレーション・ボード	デバイスに固有な周辺ハードウェアをエミュレーションするためのボードです。インサーキット・エミュレータと組み合わせて使用します。
NP-100GF エミュレーション・プローブ	インサーキット・エミュレータとターゲット・システムを接続するためのプローブです。100ピン・プラスチックQFP (GF-3BAタイプ) 用です。
EV-9200GF-100 変換ソケット (図A-2, 図A-3参照)	100ピン・プラスチックQFP (GF-3BAタイプ) を実装できるように作られたターゲット・システムの基板と、NP-100GFを接続するための変換ソケットです。

注 開発中

備考1 . NP-100GFは、株式会社内藤電誠町田製作所の製品です。

問い合わせ先：株式会社内藤電誠町田製作所 (TEL (044) 822-3813)

2 . EV-9200GF-100は、5個を1組として、1組単位で販売しています。

## A.3.1 ハードウェア (2/2)

## (2) インサーキット・エミュレータ IE-784000-Rを使用する場合

IE-784000-R インサーキット・エミュレータ	IE-784000-Rは、78K/ シリーズに共通に使用できるインサーキット・エミュレータです。別売のIE-784000-R-EM, IE-784976-NS-EM1と組み合わせて使用します。ホスト・マシンを接続してデバッグを行います。別売の統合デバッグ (ID78K4) とデバイス・ファイルが必要で、これらと組み合わせて、C言語や構造化アセンブリ言語のソース・プログラム・レベルでのデバッグが可能です。C0カバレッジ機能などにより効率の良いデバッグやプログラムの検査が行えます。ホスト・マシンとの接続は、イーサネット™または専用バスで行い、別売のインタフェース・アダプタが必要になります。
IE-70000-98-IF-C インタフェース・アダプタ	IE-784000-Rのホスト・マシンとしてPC-9800シリーズ (ノート型パソコンを除く) を使用するときに必要なアダプタです (Cバス対応)。
IE-70000-PC-IF-C インタフェース・アダプタ	IE-784000-Rのホスト・マシンとしてIBM PC/AT互換機を使用するときに必要なアダプタです (ISAバス対応)。
IE-78000-R-SV3 インタフェース・アダプタ	IE-784000-Rのホスト・マシンとしてEWSを使用するときに必要なアダプタとケーブルです。IE-784000-R内のボードに接続して使用します。 なお、イーサネットとしては10Base-5をサポートしており、他の方式の場合には市販の変換アダプタが必要になります。
IE-784000-R-EM	78K/ シリーズに共通に使用するエミュレーション・ボードです。
IE-784976-NS-EM1 <sup>注</sup> エミュレーション・ボード	デバイスに固有な周辺ハードウェアをエミュレーションするためのボードです。
IE-78K4-R-EX3 <sup>注</sup> エミュレーション・プローブ変換ボード	IE-784976-NS-EM1をIE-784000-R上で使用するときに必要な100ピン用の変換ボードです。
EP-78064GF-R エミュレーション・プローブ	インサーキット・エミュレータとターゲット・システムを接続するためのプローブです。100ピン・プラスチックQFP (GF-3BAタイプ) 用です。
EV-9200GF-100 変換ソケット (図A-2, 図A-3参照)	100ピン・プラスチックQFP (GF-3BAタイプ) を実装できるように作られたターゲット・システムの基板と、EP-78064GF-Rを接続するための変換ソケットです。

注 開発中

備考 EV-9200GF-100は、5個を1組として、1組単位で販売しています。

### A.3.2 ソフトウェア (1/2)

SM78K4 システム・シミュレータ	ホスト・マシン上でターゲット・システムの動作をシミュレーションしながら、Cソース・レベルまたはアセンブラ・レベルでのディバグが可能です。 SM78K4はWindows上で動作します。 SM78K4を使用することにより、インサーキット・エミュレータを使用しなくても、アプリケーションの論理検証、性能検証をハードウェア開発から独立して行えます。開発効率やソフトウェアの品質の向上が図れます。 別売のデバイス・ファイル (DF784976) と組み合わせて使用します。 オーダ名称：μS x x x x SM78K4
-----------------------	---

備考 オーダ名称の x x x x は、使用するホスト・マシン、OSにより異なります。

μS x x x x SM78K4

x x x x	ホスト・マシン	OS	供給媒体
AA13	PC-9800シリーズ	日本語Windows	3.5インチ2HD FD
AB13	IBM PC/AT互換機	日本語Windows	3.5インチ2HC FD
BB13		英語Windows	

### A.3.2 ソフトウェア (2/2)

ID78K4-NS <sup>注</sup> 統合ディバッガ (インサーキット・エミュレータIE-78K4-NS対応)	78K/ シリーズをディバグするためのコントロール・プログラムです。 グラフィカル・ユーザ・インタフェースとして、パソコン上ではWindows, EWS上ではOSF/Motif™を採用し、それらに準拠した外観と操作性を提供しています。また、C言語対応のディバグ機能を強化しており、ソース・プログラムや逆アセンブル表示、メモリ表示をトレース結果に連動させるウインドウ統合機能を使用することにより、トレース結果をC言語レベルで表示させることも可能です。その他、タスク・ディバッガやシステム・パフォーマンス・アナライザなどの機能拡張モジュールを取り込むことにより、リアルタイムOSを使用したプログラムのディバグ効率を向上させることができます。
ID78K4 統合ディバッガ (インサーキット・エミュレータIE-784000-R対応)	別売のデバイス・ファイル (DF784976) と組み合わせて使用します。 オータ名称: $\mu S \times \times \times ID78K4-NS$ , $\mu S \times \times \times ID78K4$

注 開発中

備考 オータ名称の  $\times \times \times$  は、使用するホスト・マシン, OSにより異なります。

$\mu S \times \times \times ID78K4-NS$

$\times \times \times$	ホスト・マシン	OS	供給媒体
AA13	PC-9800シリーズ	日本語Windows	3.5インチ2HD FD
AB13	IBM PC/AT互換機	日本語Windows	3.5インチ2HC FD
BB13		英語Windows	

$\mu S \times \times \times ID78K4$

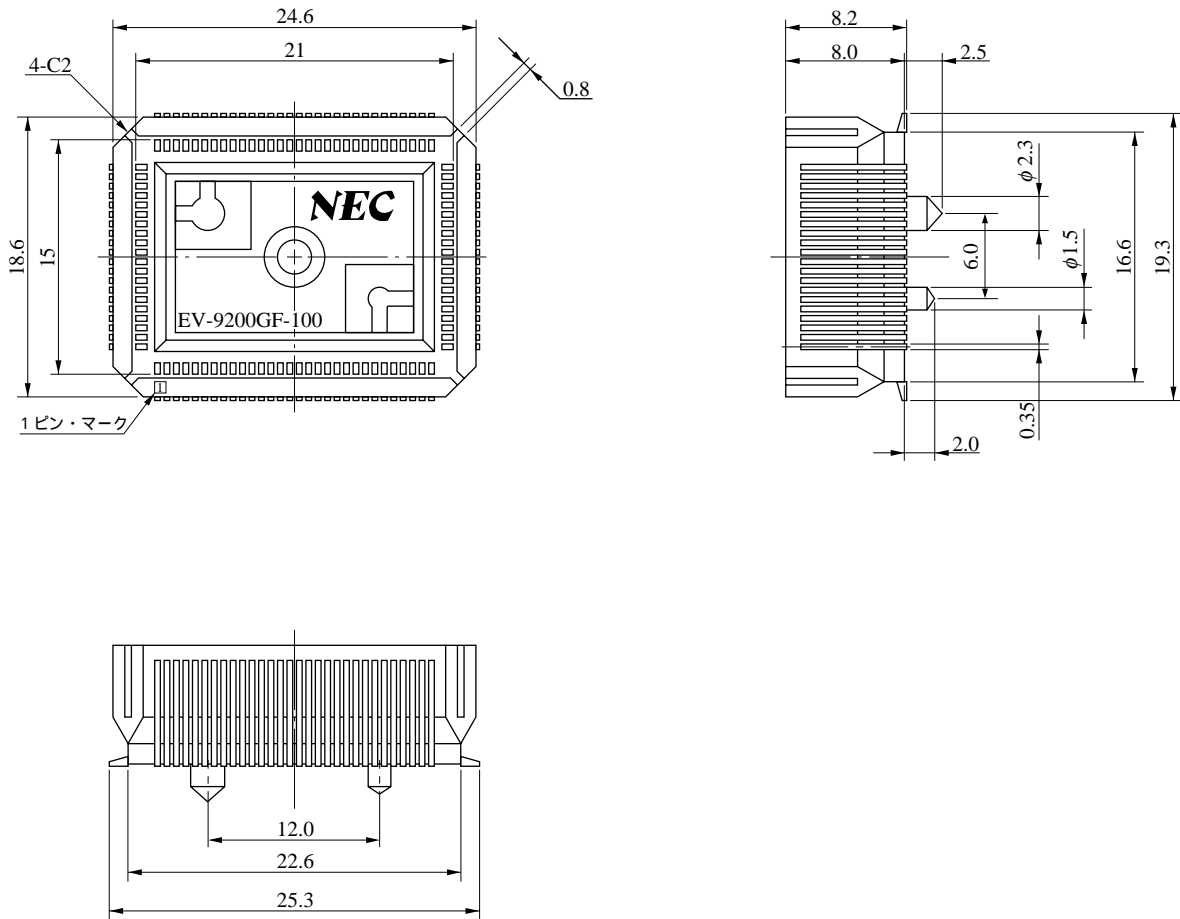
$\times \times \times$	ホスト・マシン	OS	供給媒体
AA13	PC-9800シリーズ	日本語Windows	3.5インチ2HD FD
AB13	IBM PC/AT互換機	日本語Windows	3.5インチ2HC FD
BB13		英語Windows	
3P16	HP9000シリーズ700	HP-UX ( Rel.10.10)	DAT ( DDS )
3K13	SPARCstation	SunOS ( Rel. 4 . 1 . 4 ) ,	3.5インチ2HC FD
3K15		Solaris ( Rel. 2 . 5 . 1 )	1/4インチCGMT
3R13	NEWS ( RISC )	NEWS-OS ( Rel. 6 . 1 )	3.5インチ2HC FD

## A.4 変換ソケット(EV-9200GF-100)

### (1) 変換ソケット(EV-9200GF-100)の外形図と基板取り付け推奨パターン

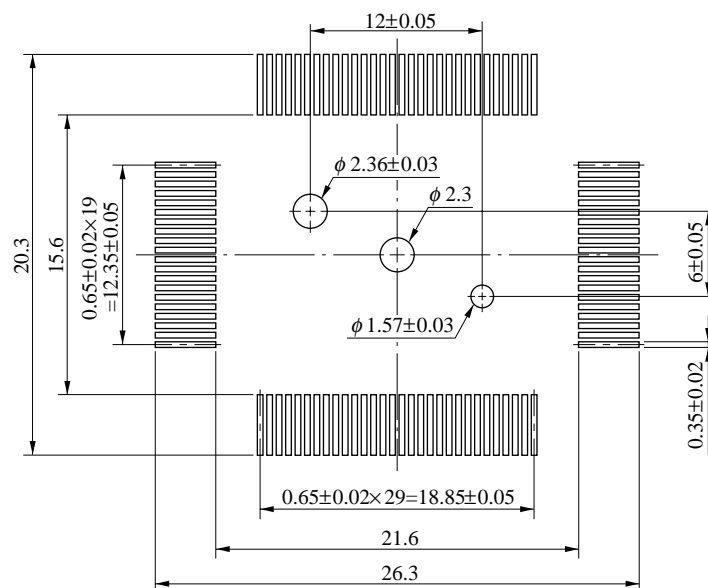
NP-100GFまたはEP-78064GF-Rと組み合わせて基板に実装します。

図A - 2 EV-9200GF-100 外形図(参考)(単位:mm)



EV-9200GF-100-G0

図A - 3 EV-9200GF-100 基板取り付け推奨パターン（参考）（単位：mm）



EV-9200GF-100-P1

注意 EV-9200用のマウント・パッド寸法と、対象製品のマウント・パッド寸法（QFP用）は、その一部が異なる場合があります。QFP用の推奨マウント・パッド寸法は、「半導体デバイス 実装マニュアル，C10535J」をご参照ください。

(メモ)



## 付録 B 組み込み用ソフトウェア

μ PD784976サブシリーズのプログラム開発やメンテナンスをより効率的に行うために、次の組み込み用ソフトウェアを用意しています。

### リアルタイムOS (1/2)

RX78K/ リアルタイムOS	<p>μ ITRON仕様に準拠したリアルタイムOSです。</p> <p>RX78K/ のニュークリアスと複数の情報テーブルを作成するためのツール（コンフィギュレータ）を添付しています。</p> <p>別売のアセンブラ・パッケージ（RA78K4）およびデバイス・ファイル（DF784976）と組み合わせて使用します。</p> <p>&lt;PC環境で使用する場合の注意&gt;</p> <p>リアルタイムOSはDOSベースのアプリケーションです。Windows上ではDOSプロンプトで使用してください。</p> <p>オーダ名称：μ S x x x RX78K4-</p>
--------------------	---

**注意** RX78K/ を購入する際、事前に購入申込書にご記入のうえ、使用許諾契約書を締結してください。

**備考** オーダ名称の x x x x および は、使用するホスト・マシン、OSなどにより異なります。

μ S x x x RX78K4-

	製品概要	量産時使用数量の上限
001	評価用オブジェクト	量産品には使用しないでください。
100K	量産用オブジェクト	10万個
001M		100万個
010M		1000万個
S01	ソース・プログラム	量産用オブジェクトのソース・プログラム

x x x x	ホスト・マシン	OS	供給媒体
AA13	PC-9800シリーズ	日本語Windows <sup>注</sup>	3.5インチ2HD FD
AB13	IBM PC/AT互換機	日本語Windows <sup>注</sup>	3.5インチ2HC FD
BB13		英語Windows <sup>注</sup>	
3P16	HP9000シリーズ700	HP-UX ( Rel.10.10 )	DAT ( DDS )
3K13	SPARCstation	SunOS ( Rel. 4 .1.4 ) ,	3.5インチ2HC FD
3K15		Solaris ( Rel. 2 . 5 . 1 )	1/4インチCGMT
3R13	NEWS ( RISC )	NEWS-OS ( Rel. 6 . 1 )	3.5インチ2HC FD

**注** DOS環境でも動作します。

(メモ)

## 付録C レジスタ索引

### C.1 レジスタ索引 (50音順)

#### 【あ行】

- インサースビス・プライオリティ・レジスタ (ISPR) ... 215
- ウォッチドッグ・タイマ・モード・レジスタ (WDM) ... 160, 217
- A/Dコンバータ・モード・レジスタ (ADM) ... 168
- A/Dコンバータ入力選択レジスタ (ADIS) ... 169
- A/D変換結果レジスタ (ADCR) ... 166

#### 【か行】

- 外部割り込み立ち上がりエッジ許可レジスタ0 (EGP0) ... 201
- 外部割り込み立ち下がりエッジ許可レジスタ0 (EGN0) ... 201
- キャプチャ/コンペア・コントロール・レジスタ0 (CRC0) ... 122

#### 【さ行】

- 16ビット・タイマ・カウンタ0 (TM0) ... 117
- 16ビット・タイマ・キャプチャ/コンペア・レジスタ00 (CR00) ... 118
- 16ビット・タイマ・キャプチャ/コンペア・レジスタ01 (CR01) ... 119
- 16ビット・タイマ・モード・コントロール・レジスタ0 (TMC0) ... 120
- シリアルI/Oシフト・レジスタ0 (SIO0) ... 180
- シリアルI/Oシフト・レジスタ1 (SIO1) ... 180
- シリアル動作モード・レジスタ0 (CSIM0) ... 181
- シリアル動作モード・レジスタ1 (CSIM1) ... 181
- スタンバイ・コントロール・レジスタ (STBC) ... 104, 272

#### 【た行】

- タイマ・クロック選択レジスタ50 (TCL50) ... 143
- タイマ・クロック選択レジスタ51 (TCL51) ... 143

#### 【な行】

- 内部メモリ・サイズ切り替えレジスタ (IMS) ... 60

#### 【は行】

- 発振安定時間指定レジスタ (OSTS) ... 107, 274
- 発振モード選択レジスタ (CC) ... 106
- 8ビット・タイマ・カウンタ50 (TM50) ... 142
- 8ビット・タイマ・カウンタ51 (TM51) ... 142
- 8ビット・コンペア・レジスタ50 (CR50) ... 142
- 8ビット・コンペア・レジスタ51 (CR51) ... 142

8ビット・タイマ・モード・コントロール・レジスタ50 (TMC50) ...	144
8ビット・タイマ・モード・コントロール・レジスタ51 (TMC51) ...	144
表示モード・レジスタ0 (DSPM0) ...	187
表示モード・レジスタ1 (DSPM1) ...	189
表示モード・レジスタ2 (DSPM2) ...	191
プリスケアラ・モード・レジスタ0 (PRM0) ...	123
プルアップ抵抗オプション・レジスタ (PUO) ...	100
プルアップ抵抗オプション・レジスタ2 (PU2) ...	99
プログラム・ステータス・ワード (PSW) ...	61, 219
ポート0 (P0) ...	84
ポート1 (P1) ...	85
ポート2 (P2) ...	86
ポート4 (P4) ...	89
ポート5 (P5) ...	90
ポート6 (P6) ...	91
ポート7 (P7) ...	94
ポート8 (P8) ...	95
ポート9 (P9) ...	96
ポート10 (P10) ...	97
ポート2モード・レジスタ (PM2) ...	98
ポート4モード・レジスタ (PM4) ...	98
ポート5モード・レジスタ (PM5) ...	98
ポート6モード・レジスタ (PM6) ...	98
ポート・リード7 (PLR7) ...	94
ポート・リード8 (PLR8) ...	95
ポート・リード9 (PLR9) ...	96

**【ま行】**

メモリ拡張モード・レジスタ (MM) ...	47
------------------------	----

**【わ行】**

割り込み制御レジスタ ...	211
割り込み選択コントロール・レジスタ (SNMI) ...	218
割り込みマスク・レジスタ0H (MK0H) ...	214
割り込みマスク・レジスタ0L (MK0L) ...	214
割り込みモード・コントロール・レジスタ (IMC) ...	216

## C.2 レジスタ索引 (アルファベット順)

### 【A】

- ADCR : A/D変換結果レジスタ ... 166
- ADIC : 割り込み制御レジスタ ... 212
- ADIS : A/Dコンバータ入力選択レジスタ ... 169
- ADM : A/Dコンバータ・モード・レジスタ ... 168

### 【C】

- CC : 発振モード選択レジスタ ... 106
- CR00 : 16ビット・タイマ・キャプチャ/コンペア・レジスタ00 ... 118
- CR01 : 16ビット・タイマ・キャプチャ/コンペア・レジスタ01 ... 119
- CR50 : 8ビット・コンペア・レジスタ50 ... 142
- CR51 : 8ビット・コンペア・レジスタ51 ... 142
- CRC0 : キャプチャ/コンペア・コントロール・レジスタ0 ... 122
- CSIIC0 : 割り込み制御レジスタ ... 211
- CSIIC1 : 割り込み制御レジスタ ... 211
- CSIM0 : シリアル動作モード・レジスタ0 ... 181
- CSIM1 : シリアル動作モード・レジスタ1 ... 181

### 【D】

- DSPM0 : 表示モード・レジスタ0 ... 187
- DSPM1 : 表示モード・レジスタ1 ... 189
- DSPM2 : 表示モード・レジスタ2 ... 191

### 【E】

- EGP0 : 外部割り込み立ち上がりエッジ許可レジスタ0 ... 201
- EGN0 : 外部割り込み立ち下がりエッジ許可レジスタ0 ... 201

### 【I】

- IMC : 割り込みモード・コントロール・レジスタ ... 216
- IMS : 内部メモリ・サイズ切り替えレジスタ ... 60
- ISPR : インサースビス・プライオリティ・レジスタ ... 215

### 【K】

- KSIC : 割り込み制御レジスタ ... 211

### 【M】

- MK0H : 割り込みマスク・レジスタ0H ... 214
- MK0L : 割り込みマスク・レジスタ0L ... 214
- MM : メモリ拡張モード・レジスタ ... 44

**【O】**

OSTS : 発振安定時間指定レジスタ ... 107, 274

**【P】**

P0 : ポート0 ... 84  
P1 : ポート1 ... 85  
P3 : ポート3 ... 86  
P4 : ポート4 ... 89  
P5 : ポート5 ... 90  
P6 : ポート6 ... 91  
P7 : ポート7 ... 94  
P8 : ポート8 ... 95  
P9 : ポート9 ... 96  
P10 : ポート10 ... 97  
PIC0 : 割り込み制御レジスタ ... 211  
PIC1 : 割り込み制御レジスタ ... 211  
PIC2 : 割り込み制御レジスタ ... 211  
PLR7 : ポート・リード7 ... 94  
PLR8 : ポート・リード8 ... 95  
PLR9 : ポート・リード9 ... 96  
PM2 : ポート2モード・レジスタ ... 96  
PM4 : ポート4モード・レジスタ ... 96  
PM5 : ポート5モード・レジスタ ... 96  
PM6 : ポート6モード・レジスタ ... 96  
PRM0 : プリスケラ・モード・レジスタ0 ... 123  
PSW : プログラム・ステータス・ワード ... 61, 219  
PUO : プルアップ抵抗オプション・レジスタ ... 100  
PU2 : プルアップ抵抗オプション・レジスタ2 ... 99

**【S】**

SIO0 : シリアルI/Oシフト・レジスタ0 ... 180  
SIO1 : シリアルI/Oシフト・レジスタ1 ... 180  
SNMI : 割り込み選択コントロール・レジスタ ... 218  
STBC : スタンバイ・コントロール・レジスタ ... 104, 272

**【T】**

TCL50 : タイマ・クロック選択レジスタ50 ... 143  
TCL51 : タイマ・クロック選択レジスタ51 ... 143  
TM0 : 16ビット・タイマ・カウンタ0 ... 117  
TM50 : 8ビット・タイマ・カウンタ50 ... 142  
TM51 : 8ビット・タイマ・カウンタ51 ... 142  
TMC0 : 16ビット・タイマ・モード・コントロール・レジスタ0 ... 120  
TMC50 : 8ビット・タイマ・コントロール・レジスタ50 ... 144

TMC51 : 8ビット・タイマ・コントロール・レジスタ51	...	144
TMIC00 : 割り込み制御レジスタ	...	211
TMIC01 : 割り込み制御レジスタ	...	211
TMIC50 : 割り込み制御レジスタ	...	212
TMIC51 : 割り込み制御レジスタ	...	212

**【W】**

WDTIC : 割り込み制御レジスタ	...	211
WDTM : ウォッチドッグ・タイマ・モード・レジスタ	...	160, 217

— お問い合わせ先 —

【技術的なお問い合わせ先】

N E C 半導体テクニカルホットライン (インフォメーションセンター)  
 (電話: 午前 9:00 ~ 12:00, 午後 1:00 ~ 5:00)

電話 : 044-548-8899  
 FAX : 044-548-7900  
 E-mail : s-info@saed.tmg.nec.co.jp

【営業関係お問い合わせ先】

半導体第一販売事業部								
半導体第二販売事業部	〒108-8001	東京都港区芝5-7-1	(日本電気本社ビル)					(03)3454-1111
半導体第三販売事業部								
中部支社 半導体第一販売部	〒460-8525	愛知県名古屋市中区錦1-17-1	(日本電気中部ビル)					(052)222-2170
中部支社 半導体第二販売部								(052)222-2190
関西支社 半導体第一販売部	〒540-8551	大阪府大阪市中央区城見1-4-24	(日本電気関西ビル)					(06)6945-3178
関西支社 半導体第二販売部								(06)6945-3200
関西支社 半導体第三販売部								(06)6945-3208
北海道支社	札幌	(011)231-0163	甲府支店	甲府	(055)224-4141	京都支社	京都	(075)344-7824
東北支社	仙台	(022)267-8740	長野支店	松本	(0263)35-1662	神戸支社	神戸	(078)333-3854
岩手支店	盛岡	(019)651-4344	静岡支店	静岡	(054)254-4794	中国支社	広島	(082)242-5504
郡山支店	郡山	(024)923-5511	立川支店	立川	(042)526-5981,6167	鳥取支店	鳥取	(0857)27-5311
長岡支店	長岡	(0258)36-2155	埼玉支店	大宮	(048)649-1415	岡山支店	岡山	(086)225-4455
水戸支店	水戸	(029)226-1717	千葉支店	千葉	(043)238-8116	四国支社	松山	(089)945-4149
群馬支店	高崎	(027)326-1255	神奈川支店	横浜	(045)682-4524	九州支社	福岡	(092)261-2806
太田支店	太田	(0276)46-4011	三重支店	津	(059)225-7341			
宇都宮支店	宇都宮	(028)621-2281	北陸支店	金沢	(076)232-7303			



## 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] μPD784976サブシリーズ ユーザーズ・マニュアル(暫定)  
(U14119JJ1V0UM00(第1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半導体テクニカルホットライン

FAX: (044) 548-7900